10.17625/NKE.2021.001

Doktori (PhD) értekezés

Gerevich János

2020. 08. 26.

NEMZETI KÖZSZOLGÁLATI EGYETEM
Hadtudományi Doktori Iskola

Gerevich János

Agilis szoftverfejlesztési technikák alkalmazási lehetőségei a Magyar Honvédség számára

Doktori (PhD) értekezés

Témavezető:

Dr. habil. Négyesi Imre PhD

……..

Budapest, 2020

TARTALOMJEGYZÉK

BEVEZETÉS

Társadalmunk hatalmas változásokon ment keresztül az első Neumann-elvű számítógép[1] megalkotása óta. A változások egyik fő motorja a számítógépek és számítástudomány hatalmas fejlődése, az informatikai megoldások széleskörű térnyerése volt. Valószínűleg a második világháborút követő években Abraham Maslow sem gondolta, hogy egyszer a számítógép – kis túlzással – az emberi szükségletek[2] minden szintjének biztosításához valamilyen formában kapcsolódni fog. Az azonban minden túlzás nélkül kijelenthető, hogy manapság a biztonságot jelentő második maslowi szinthez és a fölötte elhelyezkedő összes réteghez szervesen kapcsolódik az informatika valamilyen alkalmazása. A piramis csúcsán elhelyezkedő önmegvalósításhoz mára információs rendszerek sokaságának kell helyesen és összehangoltan működniük – ezzel megteremtve a sikeres információs társadalom alapjait. Az egészségügyi, pénzügyi, kereskedelmi, közigazgatási, oktatási és védelmi információs rendszerek mindannyiunk – és így társadalmunk – életminőségét alapvetően befolyásolják.

Klasszikus értelemben a társadalom egészére vonatkoztatott biztonság a Barry Buzan nevéhez köthető biztonsági komplexumok[3] védelmével, a biztonság külön-külön ágazatilag kezelt fokozásával érhető el. A XXI. században az információs komplexum, mindent átölelő dimenzióként teljesedett ki és mára átfogó kapcsolattal rendelkezik a politikai, gazdasági, társadalmi, környezeti és – a jelen értekezés vonatkozásában kiemelkedő fontosságú – katonai biztonsági komplexummal.

Kutatásom célja olyan módszerek és technológiák kidolgozása, amelyek a társadalomtudomány és a számítástudomány szempontrendszereit ötvözve javítani tudják az információs- és katonai biztonságot.

---

[1] Neumann elvű számítógép – a Neumann János által kidolgozott EDVAC számítógép [1]

[2] Emberi szükségletek háromszöge – 5 szintű háromszög, szintjei egymásra épülő emberi szükségleteket reprezentálnak:

1. *„A piramis alján az alapszükségletek, a létfenntartáshoz kapcsolódó szükségletek helyezkednek el.*
2. *A létfenntartás megteremtése magával hozza a biztonsági szükségletek kialakulását: ez a megszerzett javak megóvását, védelmét jelenti.*
3. *A szociális szükségletek az ember társas lény mivoltából fakadnak. A szociális szükséglet kapcsolatteremtési, összetartozási szükséglet. Ennek kielégítése érdekében törekszik jó családi, érdeklődésének, gondolkodásmódjának megfelelő baráti, munkahelyi kapcsolatokra.*
4. *Az ember igyekszik megtalálni helyét a többiek, a társai között, ebből fakad az elismerés iránti szükséglete: igyekszik elfogadtatni magát, elismertetni egyéniségét, képességét, rátermettségét*
5. *A piramis csúcsán az önmegvalósítás szükséglete áll. Az emberek egy része erős késztetést érez arra, hogy képességét, tehetségét maximálisan kihasználja."* [2]

[3] Biztonsági komplexumok – társadalmi, környezeti, katonai, gazdasági, politikai dimenziók alkotják az eredeti modellt. [3]

Napjaink információs társadalmában a katonai célú informatikai rendszerek jelentősége folyamatosan növekedik. Ezen rendszerek a potenciális haderő növeléséhez és a védelmi képességek fenntartásához egyaránt hozzájárulnak. Magyarország védelméért az Alaptörvény [4, 45. cikk] alapján a Magyar Honvédség felel, a feladat ellátásához magas színvonalú informatika rendszerek alkalmazása szükséges mind a stacioner[4] és a tábori rendszerek területén. A honvédelem megfelelő szintű megszervezéséhez elengedhetetlen a magas szinten támogatott elektronikus ügyintézés és a zavartalan külső kapcsolattartás. Ebből a tekintetből a Magyar Honvédség a közigazgatási szervekéhez hasonló ügyviteli folyamatokat valósít meg – szem előtt tartva a hierarchikus szervezeti felépítést és katonai parancs elvű működést. Az imént említett ügyviteli és iratkezelési folyamatok képezik a hivatalos információáramlás alapját a védelmi szektor vérkeringésén belül és kimenő irányba is. A közigazgatási szervekkel, egyéb jogi személyekkel, természetes személyekkel történő elektronikus kapcsolattartás túlnyomórészt már informatikai rendszerekkel támogatva, elektronikusan valósul meg. Jelentős súllyal bírnak azok a módszerek és technológiák, amelyekkel a védelmi szektor infokommunikációs rendszereit kialakítják, mert a kialakított rendszerek biztonsági és megbízhatósági mutatói napi szinten hatással vannak a társadalom átfogó biztonságára is.

A kutatás hatóköre

A kutatás célja az imént vázolt probléma feltárása szoftvertechnológiai és védelemi szempontok szerint, ezt követően olyan szoftverfejlesztési módszerek és tervezési minták kialakítása, amelyek hasznosíthatók védelmi és általános célú infokommunikációs rendszerek fejlesztése során is.

Az elmúlt két évtizedben tapasztalható felhő alapú megoldások elterjedéséből kifolyólag a disszertáció kizárólag virtuális közegben, a kibertérben értelmezett szoftvertechnológiai problémákat tárgyal. A fizikai mérőszámokat, a hálózati és hardver környezethez köthető kérdéseket csak érintőlegesen vizsgálja a dolgozat – mindig az adott szoftverfejlesztési probléma megértéséhez szükséges mértékben.

Szoftvertervezéssel kapcsolatos minőségi és műszaki szempontok meghatározásakor az Európai Unió, az Észak-atlanti Szerződés Szervezete, Magyarország és a Magyar Honvédség által publikált és nyíltan fellelhető informatikai követelmények kerültek feldolgozásra. Ezen követelmények a magánszektorban használt minőségirányítási

[4] Stacioner informatikai rendszer – jelentése: állandó felhasználású informatikai rendszer

rendszerek elvárásait meghaladják. A NATO által tett ajánlások esetében, az ISO szabványok kiterjesztéséről, szigorúbb alkalmazásról beszélünk, ezért az ISO szabványcsalád tárgyalása csak érintőlegesen képezi részét a kutatásnak. Az értekezés kiindulási pontjának számítástudományi oldalról az agilis szoftverfejlesztési módszertanokat tekintettem, amelyeket a kutatásom megkezdésekor világszerte a szoftverfejlesztési tevékenységet folytató cégek 53%-a használt már. [5] A 2019-es statisztikai adatok szerint ez a szám már 97%-ra volt tehető.[6] A friss felmérésekből az is kiderül, hogy a szoftverfejlesztési tevékenységet folytató cégek negyede már több mint öt éve alkalmazza a vizsgált módszereket. Úgy gondolom, hogy az elmúlt négy év statisztikai adatai alapján az agilis szoftverfejlesztési módszertanok kiindulási alapnak tekintése helyes döntés volt a kutatás megkezdésekor.

Az agilis szoftverfejlesztés bemutatása

Még a közelmúltban is a katonai célú alkalmazás volt az egyik inkubátora a különböző műszaki fejlesztéseknek, előbb jelentek meg egy új technológia katonai alkalmazásai és csak utána következhetett a civil használatba vétel. Példa lehet erre a GPS vagy akár az Internet is. Napjainkra ez a trend változni látszik, vannak olyan műszaki területek, amelyek eredményeit a védelmi szektorban már csak kifejlődésüket követően veszik át. Egy ilyen terület a szoftverfejlesztés is, jelenleg rengeteg eszközre és architektúrára készülnek a szoftverek különböző programozási nyelveken szerteágazó technológiai alapokon. A felhasznált programozási nyelvek a hagyományos imperatív nyelvektől a funkcionális programozási nyelveken át jelen vannak.

A számtalan új és gyorsan változó követelményhez az informatikának is alkalmazkodnia kellett, új szoftverfejlesztési módszerek jelentek meg, amelyek gyűjtőnév gyanánt az *agilis szoftverfejlesztési módszertan* nevet kapták. A *Scrum* [7], a *Kanban* [8], a *Lean* [9] különböző válfajai az agilis módszertannak. Ezek alkalmazásában a sok éves múltra visszatekintő nagy szervezetek általában szkeptikusok, legyenek azok telekommunikációs cégek, bankok, államigazgatás és természetesen ide sorolható a hadsereg is. Miért van ez így? Az új módszertan a régitől eltérő szerepköröket, munkafolyamatokat határoz meg, amelyekhez a nagy szervezetek csak nagyon lassan tudnak idomulni.

[5] Az agilis szoftverfejlesztés elterjedését vizsgáló 2016-os nemzetközi felmérés adatai alapján [5]
[6] A StateOfAgile.com amerikai weboldal által megismételt 2019-es felmérés adatai alapján [6]

A *Kiáltvány az agilis szoftverfejlesztésért* [10] című dokumentumot 2001-ben fogalmazták meg korunk vezető szoftvermérnökei, amelyben a szoftverfejlesztés sikerének zálogát új alapokra helyzeték és mellé 12 agilis alapelvet azonosítottak:

- *„Legfontosabbnak azt tartjuk, hogy az ügyfél elégedettségét a működő szoftver mielőbbi és folyamatos szállításával vívjuk ki.*
- *Elfogadjuk, hogy a követelmények változhatnak akár a fejlesztés vége felé is. Az agilis eljárások a változásból versenyelőnyt kovácsolnak az ügyfél számára.*
- *Szállíts működő szoftvert gyakran, azaz néhány hetenként vagy havonként, lehetőség szerint a gyakoribb szállítást választva.*
- *Az üzleti szakértők és a szoftverfejlesztők dolgozzanak együtt minden nap, a projekt teljes időtartamában.*
- *Építsd a projektet sikerorientált egyénekre. Biztosítsd számukra a szükséges környezetet és támogatást, és bízz meg bennük, hogy elvégzik a munkát.*
- *A leghatásosabb és leghatékonyabb módszer az információ átadásának a fejlesztési csapaton belül, a személyes beszélgetés.*
- *A működő szoftver az elsődleges mércéje az előrehaladásnak.*
- *Az agilis eljárások a fenntartható fejlesztést pártolják. Fontos, hogy a szponzorok, a fejlesztők és a felhasználók folytonosan képesek legyenek tartani egy állandó ütemet.*
- *A műszaki kiválóság és a jó terv folyamatos szem előtt tartása fokozza az agilitást.*
- *Elengedhetetlen az egyszerűség, azaz az elvégezetlen munkamennyiség maximalizálásának művészete.*
- *A legjobb architektúrák, követelmények és rendszertervek az önszerveződő csapatoktól származnak.*
- *A csapat rendszeresen mérlegeli, hogy miképpen lehet emelni a hatékonyságot, és ehhez hangolja és igazítja az működését."* [11]

Kutatási célok

Alapvető kutatási célom egy olyan hibrid, ugyanakkor agilis módszertan kialakítása, amely alkalmazható egy katonai szervezet és egy szoftvergyártó cég közös projektmunkájára a legjobb minőségű végeredmény elérése érdekében. Ebbe beletartozik a megfelelő szerepkörök és feladatok azonosítása a tervezési, fejlesztési, tesztelési idő-

szakokban. Érdekes kutatási lehetőség a projekt szakaszaihoz a szükséges dokumentációkra vonatkozó követelmények meghatározása, milyen táblázatokra, sablonokra van szükség az egyes feladatok elvégzéséhez. Célom olyan egyedi technikák és módszerek kialakítása, amelyek segítségével a speciális katonai követelmények (sérülékenységvizsgálat, hibatűrés, számítási kapacitás kiesés, sávszélesség csökkenés, algoritmusok közötti váltás, stb..) a kezdetektől fogva beépíthetők a kifejlesztendő szoftverbe és ez által egy folyamatosan katonai célra fejlesztett rendszer állítható elő már az első lépésektől.

A kutatás további célja a kialakított módszertan alkalmazásakor létrejövő architekturális és szoftvertervezési minták azonosítása és feltárása. Olyan katonai informatikai rendszereknél, ahol a hierarchikus szervezet is része a modellnek, komoly problémákkal találkozhatnak az informatikusok. Az egyes szervezeti elemek felszámolása, áthelyezése kihatással lehet az addig kiadott parancsokra, határidőkre, folyamatokra, felelősökre, hatáskörökre, jogosultságokra, ezekben az esetekben nagyon fontos a jogutódok kiszámítása, modellezése, hogy a folytonosság fenntartható legyen az adott információs rendszeren belül. A modern technológia és a speciális követelmények szintéziséből új tervezési minták, megoldások születhetnek, amelyeket fel lehet használni létező rendszerek modernizálásakor vagy új rendszerek kifejlesztésekor. Az alkalmazott agilis módszerekhez tartozó folyamatok feltárása és megértése után elkészíthetők és kialakíthatók azok a speciális szoftverre vonatkozó közbeszerzési eljárások és belső szabályozók, amelyek alapján a Magyar Honvédség is megfelelő képességekkel rendelkezhet egy agilis szoftverfejlesztési pályázat kiírására és megvalósítására.

Tudományos problémák

- Alkalmazhatók-e az agilis szoftverfejlesztési módszertanok a Magyar Honvédség számára?
- Milyen folyamatok és dokumentum sablonok szükségesek egy agilis szoftverfejlesztési projekt szabályozott végrehajtásához?
- Miben lehet más egy katonai célra agilisan készített egyedi szoftver, mint egy civil alkalmazás?
- Milyen tervezési minták, szoftver architektúrák merülhetnek fel egy agilis szoftverfejlesztési módszertan katonai célú alkalmazása során?

Hipotézisek

- Az agilis szoftverfejlesztési módszertanok alkalmazhatók katonai célra a különböző katonai szakterületek speciális követelményeinek figyelembe vételével.
- Katonai alkalmazás során is azonosíthatók az agilisan végrehajtott projekt fázisai, amelyekhez kidolgozhatók a megfelelő dokumentációs sablonok.
- Egy agilis szoftverfejlesztési módszertan alkalmazása során rugalmasabb és a valóságot jobban modellező katonai célú szoftverek fejleszthetők ki.
- Új szoftvertervezési minták azonosíthatók egy katonai célra agilisan fejlesztett szoftver előállítása során, amelyek a későbbiekben széleskörűen alkalmazhatók.

Kutatási stratégia és módszerek

Az agilis szoftverfejlesztés és a védelmi informatika külföldi és hazai szakirodalmának együttes feldolgozásával, valamint a több mint 15 éves személyes szoftverfejlesztési és tervezési tapasztalataimra támaszkodva egy olyan követelményrendszer kidolgozása a célom, amelynek kielégítéséhez a tipikus tervezési, fejlesztési, tesztelési problémákat széleskörűen fel lehet tárni. Az kutatás során valós problémákból levezetett, az átfogó vizsgálat számára leegyszerűsített, ugyanakkor elégséges példákat fogok vizsgálni a védelmi szempontok és az agilis szoftverfejlesztési irányelvek szem előtt tartásával. A kiválasztott példák az elektronikus ügyintézésből, a katonai iratkezelésből és egyéb védelmi informatikához köthető területekből kerülnek levezetésre, amelyeket eddigi szakmai tevékenységem során már átfogóan volt szerencsém megismerni. A kutatás során kizárólag nyilvános és mindenki számára elérhető irodalom kerül felhasználásra, úgy gondolom, hogy publikus információk alapján is teljes mértékben vizsgálható a probléma.

Kutatói tevékenységem alatt szem előtt fogom tartani a védelmi informatikai elvárásokat és az agilis szoftverfejlesztési elveket egyaránt. Eleinte deduktív módszerrel fogom az általánosan ismert védelmi, biztonsági, informatikai követelmények felhasználásával és szoftverfejlesztésben elfogadott módszerek alapján kialakítani az alap problémát általános és speciális kutatási követelmények formájában.

A követelményrendszer alapját az Európai Unió közleményei és irányelvei, a NATO projektmenedzsment szabványai, Magyarország különböző biztonsági stratégiái, illetve a kutatáshoz kapcsolódó hatályos jogszabályai és a Magyar Honvédség informatikai követelményeket tartalmazó dokumentumai adják. A kutatás által vizsgált irodalom feltárása során általános informatikai, projektmenedzsment, tervezési,

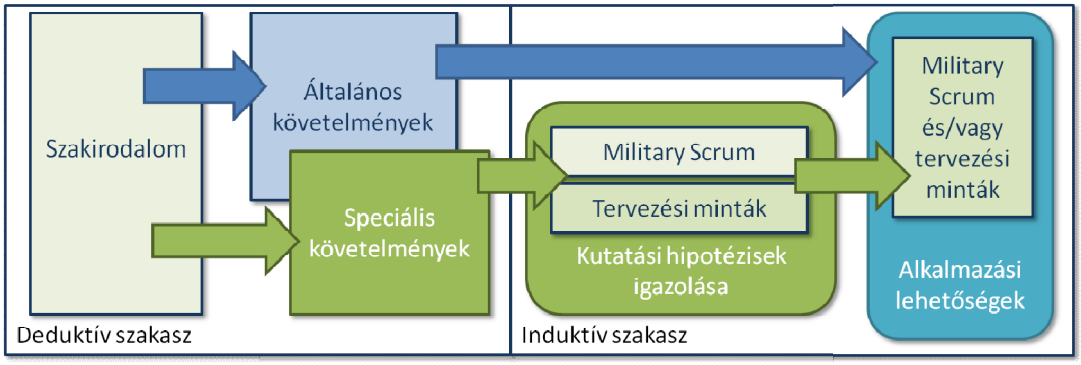
szoftverfejlesztési, szolgáltatási, üzemeltetési, biztonsági, megbízhatósági (minőségi) követelmények azonosítása a cél.

A kialakított szempontrendszer segítségével, érintve a szoftvertechnológiát és a védelmi informatikát, induktív módszerekkel kerül kialakítása a feltárt speciális követelményeket kielégítő szoftverfejlesztési módszertan, a *Military Scrum*. A módszertan részeként előállnak a folyamatleírások, dokumentációs technikák és dokumentum sablonok is, valamint a módszertan által megkívánt szoftverfejlesztési eljárások és üzemeltetési ajánlások is.

A módszertan meghatározását követően az elektronikus ügyintézéshez és a hierarchikus szervezetek folyamataihoz köthető példák segítségével olyan tervezési minták kialakítása a cél, amelyek megfelelnek a kor követelményeinek, valamint üzembiztosan, megbízhatóan és biztonságosan működő szolgáltatások alakíthatók ki felhasználásukkal.

A deduktív szakasz során feltárt kutatási követelményeknek való megfelelés két fázisban történik. A speciális kutatási követelményeknek való megfelelés végeredményei egy katonai célú alkalmazást lehetővé tevő szoftverfejlesztési módszertan és a katonai szervezetek működését támogató szoftvertervezési minták. A két egzakt eredmény segítségével a kutatási hipotézisek igazolása várhatóan megvalósul.

A kutatás során feltárt általános követelmények kielégítése a kutatási eredmények felhasználásával valósulhat meg – a kutatási hipotézisek igazolásán túl –, az eredmények alkalmazási lehetőségeinek bemutatásával.



1. ábra Kutatási stratégia (saját szerkesztés)

1. AZ EURÓPAI UNIÓ ÉS MAGYARORSZÁG LÉTFONTOSSÁGÚ RENDSZEREK VÉDELMÉVEL KAPCSOLATOS JOGSZABÁLYI KÖRNYEZETE

Az értekezésben tárgyalt tudományos problémák vizsgálatához elengedhetetlen a kutatás megfelelő, analízisre alkalmas környezetbe való elhelyezése. A dolgozat vonatkozásában lényeges társadalomtudományi kapcsolódási pontok azonosításához, illetve az általános kutatási követelményrendszer levezetéséhez feltétlenül szükséges feltárni a témához kapcsolódó nemzetközi és hazai szabályozó dokumentumokat.

A feltáró munka alatt az időrendben történő feldolgozás technikáját alkalmazom az érintett terület szakirodalmát tekintve, ugyanis a különböző dokumentumok hatással voltak egymásra megjelenésükkor – és jelenleg is hatással vannak. A NATO és az EU szinten azonosított globális biztonsági kockázatok, valamint a magyar biztonsággal foglalkozó stratégiai dokumentumok szervesen kapcsolódnak egymáshoz, ezért értelmezésük időrendben célszerű a kiberbiztonsági kockázatok vonatkozásában is.

A kutatás témájához igazodva, az általános dokumentumok vizsgálata a kiberbiztonság és a szoftvertechnológia szemszögéből történik. A szakirodalom feldolgozása az ezredforduló után publikált dokumentumokkal veszi kezdetét. Tekintettel arra, hogy a *kritikus infrastruktúra* és a *létfontosságú rendszer* egyaránt használt kifejezések ugyanarra a fogalomra a szakterminológiában, a továbbiakban szinonimaként használom őket. A létfontosságú infokommunikációs rendszerek biztonságának fokozását előirányzó intézkedések a kritikus infrastruktúra védelem szabályozásának további bővítésével jöttek létre, így azok alapvető áttekintése is indokolt.

Az infokommunikációs rendszerek biztonságával legmagasabb szinten elsőként 2001-ben foglalkozott az Európai Közösségek Bizottsága.[7] A *Javaslat egy európai hálózat- és informatikai biztonsági politikára* [12] című dokumentum *"a témakör átfogó megközelítésére épült, amely abból indult ki, hogy a hálózatok, és informatikai rendszerek széles körben alkalmazott támogató infrastruktúrákká, a gazdasági és társadalmi fejlődés kulcstényezőivé váltak, így biztonságuk alapvető prioritás"* [13] – olvasható Munk Sándor kapcsolódó tanulmányában. Az ezt követő években a kritikus infrastruktúrákkal kapcsolatos feladatrendszer kidolgozását tagállami és uniós szinten is megkezdték.

7 Európai Közösségek Bizottsága - 2007-től Európai Bizottság

1.1 EURÓPAI KRITIKUS INFRASTRUKTÚRÁK

2006 decemberében az Európai Közösségek Bizottsága egy 3 tevékenységi irányból álló cselekvési tervet fogalmazott meg a létfontosságú rendszerek azonosítási és kijelölési folyamatának kidolgozásához [14]. Az egyes tevékenységi irányokhoz az alábbi fázisok kapcsolódtak. [14, 4.1]

1) tevékenységi irány – Kritikus Infrastruktúra Védelem Európai Programjának (European Programme of Critical Infrastructure Protection, röviden: EPCIP) stratégiai aspektusai és a létfontosságú rendszerek védelmére horizontálisan alkalmazható intézkedések kidolgozása.
    a) a szabályozás kidolgozására vonatkozó ágazati sorrend, definíciók és fogalmak meghatározása. Létfontosságú rendszerek védelméhez szükséges azonosítási-, együttműködési-, iránymutatási-, adatmegosztási-, kockázatelemzési eszközök és módszerek feltárása az EPCIP megvalósításához;
    b) a cselekvéshez szükséges pénzügyi alapok megteremtése, melyből uniós szintű szakértői csoportok, tevékenységek finanszírozhatók;
    c) harmadik országokkal és nemzetközi szervezetekkel történő együttműködés megteremtése;
2) tevékenységi irány – az európai létfontosságú rendszerek védelme és az ágazati kérdések kezelése.
    a) ágazati követelmények meghatározása az európai szintű létfontosságú rendszerek azonosításához;
    b) a létfontosságú rendszerek azonosítására és vizsgálatára vonatkozó alapok kialakítása ágazatonként. Európai létfontosságú rendszerek kijelölése, sérülékenységek, kockázatok azonosítása. Szabályozások és biztonsági szintek harmonizációja;
    c) európai létfontosságú rendszerek minimális védelmi képességére vonatkozó javaslatok kidolgozása és a megfelelő intézkedések foganatosítása;
3) tevékenységi irány – tagállamok támogatása a saját nemzeti létfontosságú rendszereikkel kapcsolatos tevékenységük során.
    a) nemzeti létfontosságú rendszerek azonosításához használt követelmények megosztása a tagállamok között;

b) a létfontosságú rendszerek azonosítására és vizsgálatára vonatkozó alapok kialakítása ágazati szinten, tagállami szintű létfontosságú rendszerek kijelölése, biztonsági rések elemzése;
c) tagállami szintű létfontosságú rendszerekre vonatkozó védelmi programok kialakítása és fejlesztése, védelmi intézkedések kialakítása, valamint a tulajdonosok, illetve üzemeltetők felügyelete;

A cselekvési terv megjelenése után két évvel, 2008. december 8-án a kritikus infrastruktúrák azonosításával, kijelölésével, valamint védelmük értékelésével és javításával foglalkozó 2008/114/EC számú Európai Tanács által meghatározott irányelv [15] fektette le és kötötte határidőhöz a létfontosságú rendszerek európai szintű nyilvántartásba vételét, melynek végrehajtására 2011. január 12-ét szabta határidőként a tagállamok számára. Első lépésben az energetika és a szállítmányozás [15, (5)] jelent meg kiemelt területként, ugyanakkor az infokommunikációs szektor már itt is említésre került, mint vizsgálandó terület. Az irányelv az alábbiak szerint definiálta a kritikus infrastruktúra védelemmel kapcsolatos fogalmakat: [15, 2. cikk.]

1. Kritikus infrastruktúra – Egy adott Európai Uniós tagállam területén működő létesítmény, rendszer vagy rendszerelem, amely alapvetően szükséges a létfontosságú társadalmi, egészségügyi, biztonsági, védelmi, gazdasági és szociális jóléti funkciók ellátásához. Egy kritikus infrastruktúra sérülése vagy megsemmisítése esetén az adott tagállam nem tudná megfelelően ellátni a funkcióit az érintett területeken.
2. Európai kritikus infrastruktúra (European Critical Infrastructure, röviden: ECI) – olyan kritikus infrastruktúra, melynek sérülése vagy megsemmisülése legalább 2 tagállamra hatással van.
3. Kockázatelemzés – egy kritikus infrastruktúra sérülésének vagy megsemmisülésének hatásvizsgálata.
4. Védelem – olyan tevékenység, amely a fenyegetések megelőzését, enyhítését, illetve semlegesítését célozza meg egy adott létfontosságú rendszer vagy rendszerelem folyamatos fenntartása, működtetése, és integritásának megőrzése céljából.
5. Védelemi tevékenység során keletkező érzékeny információ – olyan tények és adatok, melyek felhasználásával célzott támadás indítható egy adott kritikus infrastruktúra ellen.

6. Európai kritikus infrastruktúra tulajdonosa, illetve üzemeltetője – azok az érintett felek, akik valamilyen formában (anyagi vagy üzemeltetési) felelősséggel tartoznak egy adott kritikus infrastruktúrára vonatkozóan.

A további cikkelyekben [15, 3-9 cikk.] az európai kritikus infrastruktúrák azonosítása, kijelölése, a védelemhez szükséges biztonsági tervek és a biztonsági összekötő tisztviselők[8] szerepe, a védelmi tevékenységről készült beszámolók és a védelmi tevékenység során keletkező érzékeny adatok kezelése került tárgyalásra. Az azonosítási eljárás során az adott ECI sérülése vagy megsemmisülése következtében bekövetkező halálesetek vagy sérülések száma, a gazdasági-, illetve társadalmi hatás került meghatározásra kijelölési szempontként. A kijelölés támogatását a tagállamok közötti együttműködés megszervezésével, két- illetve többoldalú szerződések megkötésével javasolta az Európai Tanács. A kijelölési folyamat tárgyalását követően a biztonsági terveknek megfelelő üzemeltetés fenntartása kapott hangsúlyt.

Elmondható, hogy az Európai Tanács egy átgondolt, jól strukturált és következetes előkészítést követően megalapozta a létfontosságú rendszerek védelmének szabályozását az Európai Unió egészére és a tagállamokra vonatkozóan is. A kialakított szabályozási keretrendszer folyamatait és terminológiáját tekintve a fizikailag létező kritikus infrastruktúra védelemre koncentrált. A kialakított védelmi modell még nem szabályozta az informatikai rendszerek védelmét, ugyanakkor a 2008-as kritikus infrastruktúra védelmet szabályozó EU irányelv jelentősége nagy, mert alapját képezi a későbbiekben tárgyalt létfontosságú infokommunikációs rendszerek védelmét szabályozó dokumentumoknak Uniós és hazai viszonylatban is – ez a dokumentum képezi a létfontosságú rendszerek védelmét előíró magyar szabályozás alapját is, amelyről a következő fejezetben olvashatunk.

Mielőtt a létfontosságú rendszerekkel kapcsolatos magyar szabályozást áttekintenénk, érdemes rövid pillantást vetni a biztonság és ezen belül is a kiberbiztonság szemszögéből a jelenleg érvényes hazai biztonsági stratégiákra. Az alábbiakban az értekezés szempontjából lényeges pontokat, gondolatokat tekintem át, némi magyarázattal kibővítve. A létfontosságú rendszerek védelme alatt a fizikailag létező infrastruktúrák védelmét értjük – tehát a védelmi tevékenység tárgyát egy általános feladatot ellátó infrastruktúra képezi: építmények, hálózatok, gyárak, üzemek, stb.. A fizikailag létező infrastruktúra által nyújtott szolgáltatások létfontosságúak, kritikusak, ha azok kiesése valamilyen társadalmi, gazdasági vagy védelmi funkció ellátását

[8] Biztonsági összekötő tisztviselő – angol kifejezéssel: Security Liaison Officer

egyértelműen gátolják. Néhány példa általános célú kritikus infrastruktúrára: közigazgatás, szállítmányozás, ellátás, hírközlés, védelem (pl.: rendvédelem, katasztrófavédelem), egészségügy, oktatás, stb..

1.2 LÉTFONTOSSÁGÚ RENDSZEREK MAGYARORSZÁGON

Hogy teljes képet kaphassunk a létfontosságú infokommunikációs rendszerek védelméről röviden érdemes néhány szót ejteni a 2012. évi CLXVI. Törvényről [16], amely a létfontosságú rendszerek és létesítmények azonosításáról, kijelöléséről és védelméről rendelkezik, a törvény végrehajtását a 65/2013. (III. 8.) Korm. Rendelet [17] szabályozza. Az alábbi főtevékenységek jelennek meg az imént említett két dokumentumban a létfontosságú rendszerekre vonatkozóan:

1. Azonosítási eljárás – *„az a folyamat, amely során a lehetséges létfontosságú rendszerelemeket kockázatelemzés, valamint az ágazati és horizontális kritériumok alapján meghatározzák;”* [17, 1.1]
2. Kijelölés eljárás – a nemzeti és az európai létfontosságú rendszerelemmé történő nyilvánítás folyamata. Abban az esetben indítható, ha az azonosítási eljárás eredménye pozitív, tehát az adott rendszerelem megfelel valamely ágazati és horizontális kritériumnak.
3. Visszavonási eljárás – a nemzeti és az európai létfontosságú rendszerelem státusz megszüntetésére vonatkozó folyamat, melyet nemzeti szinten a kijelölő hatóság folytat le, pozitív eredmény esetén az adott létfontosságú rendszerelem törlésre kerül a nyilvántartásból. Európai szinten egy EGT tagállam kezdeményezheti, és a Kormány dönt a státusz megszüntetéséről.
4. Nyilvántartás – a kijelölt nemzeti és európai létfontosságú rendszerek nyilvántartása.
5. Üzemeltetői biztonsági terv [17, 2. melléklet] – a létfontosságú rendszerelemek nyilvántartásba vételéhez szükséges dokumentum, melynek tartalmaznia kell az általános-, a környezeti- és a kijelölt rendszerelemre vonatkozó leírást. A biztonsági terv további fejezetei a kockázatelemzés, a kockázatkezelés és megvalósításhoz szükséges eszközrendszer bemutatása.

A bemutatott folyamatos tevékenységeken túl, a Kormány éves jelentést nyújt be az Európai Bizottságnak az európai létfontosságú rendszerekkel kapcsolatos statisztikai adatokról, valamint a kapcsolódó sebezhetőségi pontokról. [16, 13] Magyarországon

a honvédelmi célt szolgáló létfontosságú rendszerek azonosítása, kijelölése és védelme a 359/2015. (XII. 2.) Korm. rendelet [17] által szabályozott, míg a 46/2016. (VIII. 25.) HM utasítás [18] tartalmazza a végrehajtással kapcsolatos feladatokat, utóbbi meghatározza az ágazati javaslattevő, nyilvántartó hatóságot és ellenőrző szervet a HM Védelmi Igazgatási Főosztály személyében, míg a kijelölő hatóság a HM Hatósági Főosztály[9] [18, 2.(1)].

Megállapítható, hogy a bemutatott eljárási rend a 2008/114/EC számú Európai Tanács által meghatározott irányelv magyar leképezése.

Az eddig szemléltetett létfontosságú rendszerekre vonatkozó folyamatok alapvetően fizikai infrastruktúra védelemre vonatkoznak, és nehéz ebben a formában interpretálni őket az infokommunikációs rendszerekre. Ezt a problémát az Európai Unió is külön kezelte, külön szabályozást hozott létre a kibertér védelmére vonatkozóan.

1.3 LÉTFONTOSSÁGÚ INFOKOMMUNIKÁCIÓS RENDSZEREK

Azt, hogy mi a különbség a fizikailag létező kritikus infrastruktúra védelme és létfontosságú infokommunikációs rendszerek védelme között egy megfelelő példa segítségével lehet a legjobban szemléltetni. Az elektronikus ügyintézés és a bizalmi szolgáltatások általános szabályairól szóló 2015. évi CCXXII. törvény [20] értelmében 2018. január 1-től a kormányzati és a gazdasági szereplőknek elektronikus úton kell kapcsolatot tartaniuk egymással. Az 1. táblázatban színes háttérrel jelölt ágazatok és a hozzájuk tartozó alágazatok egyenként érintettek az említett elektronikus kormányzati szolgáltatás megvalósítása során. Bármelyik kiesése a szolgáltatás teljes, illetve részleges kiesését okozhatja, ezzel komoly gazdasági károkat okozva. Az ilyen jellegű összetett szolgáltatások védelméhez a fizikai védelmen túl további szabályozásra és intézkedésekre van szükség az infokommunikációs rendszerek specialitásainak figyelembe vételével. Az említett rendszerek hálózati, hardver- és szoftverkomponensei különböző kockázatot rejthetnek magukban, amelyek biztonsága érdekében integrált rendszerszemléletre van szükség jogszabályi szinten is. Csak így érhetik el céljukat a védelmi képességek növelését előirányzó intézkedések.

[9] A rendelkezések bővebb elemzését Bakos Tamás publikációjában [19] találhatjuk, többek között a fogalmak ismertetését, az azonosításhoz szükséges ágazaton belüli és ágazaton kívüli kritériumokat, illetve a potenciális honvédelmi kritikus infrastruktúrák bemutatását.

	Ágazat	Alágazat
24	Infokommunikációs technológiák	információs rendszerek és hálózatok
25		eszköz-, automatikai és ellenőrzési rendszerek
26		internet-infrastruktúra és hozzáférés
27		vezetékes és mobil távközlési szolgáltatások
28		rádiós távközlés és navigáció
29		műholdas távközlés és navigáció
30		műsorszórás
31		postai szolgáltatások
32		kormányzati informatikai, elektronikus hálózatok
.	.	.
38	Jogrend – Kormányzat	kormányzati rendszerek, létesítmények, eszközök
39		közigazgatási szolgáltatások
40		igazságszolgáltatás
41	Közbiztonság – Védelem	rendvédelmi szervek infrastruktúrái
42		honvédelmi rendszerek és létesítmények

1. táblázat 2012. évi CLXVI. Törvény, 3. melléklet, részlet

A létfontosságú rendszerekre vonatkozó jogszabályok alapvetően fizikai infrastruktúra védelemre vonatkoznak és nehéz interpretálni őket infokommunikációs rendszerekre. Az Európai Unió is felismerte ezt a lényeges különbséget, külön szabályozást hozott létre a kibertérre vonatkozóan, amely célirányosan tárgyalja a bemutatott kérdéskört. A továbbiakban az EU-s és a magyar szabályozás kerül bemutatásra a kutatás szemszögéből releváns információbiztonsági követelmények feltárása mellett.

1.3.1 Létfontosságú infokommunikációs rendszerek védelme az EU-ban

2011-ben az Európai Bizottság a kritikus információs infrastruktúrák védelméről szóló közleményében [21] már kimondottan a kiberbiztonság kérdésével foglalkozott. A közlemény az elért eredmények bemutatásán túl a globális kiberbiztonság elérését tűzte ki célul. A végrehajtásért többek között az ENISA[10] felelt. A meghatározott cselekvési terv 5 pilléren állt az alábbiak szerint [21, 2. old].

1. Felkészültség és megelőzés
   a) tagállamok közötti együttműködés megteremtése hálózatbiztonsági reagáló csoportok segítségével (Computer Emergency Response Team, röviden CERT);
   b) a közszféra és a magánszféra közötti együttműködés erősítése az infokommunikációs infrastruktúrák ellenálló képességének fokozása érdekében;
2. Az alapképességek, szolgáltatások és kapcsolódó szabályok meghatározásával a jól működő tagállami CERT-ek együttesen alkotják az európai informá-

[10] ENISA – European Network and Information Security Agency, azaz Európai Hálózatbiztonsági Ügynökség, a szervezetet 2004-ben hozták létre. 2019. április 17-től Európai Kiberbiztonsági Ügynökségként működik, angolul: European Union Agency for Cybersecurity

ció megosztási és figyelmeztető rendszer (European Information Sharing and Alert System, röviden EISAS) gerincét;

3. Észlelés és reagálás
   a) az EISAS rendszer alapvető működésének megvalósítását tűzték ki célul 2013-ig a tagállami CERT-ekre alapozva, a személyes adatok védelmét határozták meg az egyik célterületnek;
4. Enyhítés és helyreállítás
   a) az ENISA által szervezett nagyszabású hálózati incidensekre történő reagálás és helyreállítás témakörében szervezett gyakorlatok; tagállami szintű iránymutatás;
   b) nagyszabású hálózati incidensekre történő felkészülés páneurópai gyakorlatok segítségével;
5. Nemzetközi együttműködés – az európai alapelvek egyeztetése a különböző nemzetközi szervezetekkel G8, OECD, NATO és partnerekkel, többek között az Egyesült Államokkal; hosszú távon egy nemzetközi keretrendszer kialakítása az ellenálló és biztonságos Internet érdekében;
6. Infokommunikációs szektorra vonatkozó ECI követelmények meghatározása – a tagállamok által meghatározott infokommunikációs szektorra vonatkozó kritérium célterülete a hagyományos- és mobil telefonhálózat, valamint az Internet szolgáltatás volt;

A közlemény 2. pontjában [21, 3-4. o.] a lehetséges fenyegetések számát, hatókörét, kifinomultságát és potenciális hatását vizsgálták az infokommunikációs technológia (röviden: ICT) terjedéséből kifolyólag. Az Európai Bizottság megállapította, hogy az új és technológiailag fejlettebb fenyegetések felbukkanása tisztán mutatja, hogy az ICT segítségével politikai-, gazdasági- és katonai erőfölény teremthető meg. Megállapították, hogy be kell sorolni a lehetséges tevékenységeket az alábbiak szerint.

1. adatlopás – gazdasági, illetve politikai kémkedés. Gazdasági és kormányzati infokommunikációs rendszerek támadása;
2. zavarkeltés – DDoS[11] támadások végrehajtása botnet hálózatok segítségével;

[11] DDoS - Distributed Denial of Service - elosztott szolgáltatásmegtagadással járó támadás, más néven túlterheléses támadás, hatására célba vett informatikai szolgáltatás megbénul vagy helytelenül működik.

3. megsemmisítés – az infokommunikációs technológia terjedésével ez a forgatókönyv is egyre valószínűbbé válhat különböző létfontosságú rendszerek esetében, például: intelligens hálózatok, vízhálózat-irányítási rendszerek

Az Európai Unió által foganatosított intézkedések hatásai megjelennek a magyar szabályozásban is. A kormányzati információbiztonság javítására való törekvések, a megfelelő eseménykezelő központok létrehozása és a jogi háttér megteremtése többek között az EU által kidolgozott és meghatározott követelményeknek köszönhető. A továbbiakban az Unió által megfogalmazott infokommunikációs rendszerek biztonságát taglaló irányelv áttekintése következik információbiztonsági követelmények után kutatva.

1.3.2 Hálózat- és információbiztonság

A kibertér az a virtuális közeg, ahol egyre több és több tevékenységet végezhetünk, ezzel összhangban folyamatosan növekedik az itt elérhető szolgáltatások száma is. Az internet alapú szolgáltatások köre folyamatosan bővül, gondoljunk csak az online vásárlásra, közösségi oldalakra, bankszektorra, e-közigazgatásra, védelmi rendszerekre egyaránt. Ugyanakkor nem szabad elfelejteni a különböző rádiótechnikai, NFC[12] alapú elektronikai megoldásokat sem, amelyek ugyancsak folyamatosan terjednek és részét képezik a kibertérnek. A technológia fejlődésének következménye, hogy a kiberbűnözés elleni harc és a kibervédelem kiemelt hangsúlyt kap a hazai és a nemzetközi szintéren is. A kibertérben tapasztalható támadások célja az információszerzés, információ-manipulálás, álhírek keltése, illetve a megszerzett információ megsemmisítése. A támadások célpontjai lehetnek magánszemélyek, ugyanakkor a magánszektor és a kormányzati szervek sem érezhetik magukat biztonságban, ideértve természetesen a fegyveres erőket is. A védelmi célokat ellátó informatikai rendszerek is lehetnek célpontjai az anyagi haszonból elkövetett kiberbűnözésnek. Az sem példátlan, hogy egy adott ország védelméért felelős katonai szervezet szemben találja magát egy másik ország fegyveres erőihez köthető hacker csoporttal. A külső eredetű kockázatok mellett ki kell emelni az informatikai szolgáltatásokat képező hardver és szoftver környezet sérülékenységeit is, melyek komoly belső kockázatot jelentenek és kezelésükhöz megfelelő módszerekre van szükség.

[12] NFC – Az angol "Near-field communication" kifejezés kezdőbetűiből kialakított rövidítés. A fogalom okostelefonok és hasonló (általában mobil) eszközök közötti kommunikációt leíró szabványgyűjteményt takarja.

A vázolt problémákat is felölelve, 2013-ban jelent meg az Európai Unió kiberbiztonsági stratégiája [22], számos azonosított kiber kihívást tárgyalva. Az egyik megvitatott nehézség a hálózat- és információbiztonság kérdése volt, amelyet a kiberbiztonsági stratégia külön pontban emelt ki, és előirányozta egy Uniós szintű irányelv kialakítását az alapvető informatikai szolgáltatások védelméről.

A kiberbiztonságért folytatott küzdelem alapját képező hatályos EU irányelv a hálózati és információs rendszerek biztonságának az egész Unióban egységesen magas szintjét biztosító intézkedés [23] (továbbiakban: NIS[13] direktíva), amely a kritikus infrastruktúrák védelmére vonatkozó szabályok kiterjesztését tartalmazza az alapvető hálózati és információs rendszerek üzemeltetőire és a digitális szolgáltatókra.

A kritikus rendszerek azonosítása, a kockázatok elemzése, a biztonsági tervek elkészítése, a reagálási intézkedések összeállítása összetett feladat. Egy újonnan kialakítandó kritikus rendszer részét vagy teljes egészét adó információs rendszer tervezése és fejlesztése során alapvető fontosságú a megfelelő módszerek és technológiák alkalmazása. A NIS direktíva 75 pontban foglalja össze azokat az érveket, meglátásokat és teendőket, amelyek szükségessé teszik a probléma megfelelő szintű kezelését. Az elemzés célja olyan követelmények azonosítása, amelyek információbiztonsági és szoftvertechnológiai szempontból is értelmezhetők és így szabványos szoftverfejlesztési módszerekkel lefedhetők.

A szabályozás szükségességét firtató indokok között szerepel a belső piacon működő informatikai rendszerek biztonságának és megbízhatóságának [23, (1)] növelése. Ha a biztonság kérdését vizsgáljuk a tervezés, a fejlesztés és az üzemeltetés vonatkozásában, akkor a biztonsági kérdések az üzemeltetés területét érintik jobban, azonban a megbízhatóság tekintetében a tervezés tűnik fajsúlyosabb területnek. Egy követelményt már azonnal sikerült is azonosítani, miszerint a megbízhatóságra tervezési és fejlesztési oldalról is törekedni kell. A biztonsági események száma folyamatosan növekedik tekintet nélkül arra, hogy szándékosság áll a háttérben vagy hiba okozza a szolgáltatások kiesését. [23, (2)-(3)]. Az említett biztonsági események határokon átívelők lehetnek, ezért fontos olyan módszerek kialakítása, amelyek a határokon átnyúló, felhő alapú üzemeltetési környezetben is megfelelő megbízhatósági és biztonsági paraméterekkel rendelkeznek. A kialakítandó módszerek, technológiák sikerének záloga a tagállamok közötti stratégiai együttműködés [23, (4)], amely magában

[13] NIS – Hálózat- és információbiztonság. „Network and Information Security” kifejezés kezdőbetűiből kialakított rövidítés.

foglalja a magas szintű biztonságot és megbízhatóságot garantáló szabványok kialakítását és alkalmazását is. A tagállamok felkészültségi szintje a kiberbiztonság tekintetében nem egyenlő, az alapvető szolgáltatásokat nyújtó szereplők és a digitális szolgáltatók számára egységes minimum követelmények szükségesek Uniós szinten. [23, (5)-(6)] Az említett követelmények, egyaránt vonatkoznak hálózat-üzemeltetésre és információs rendszerek üzemeltetésére is. Mindkét esetben a biztonsági események detektálása, a logikai hibák feltárása, javítása szabványos eljárásokat és módszereket igényel. Az alapvetések elemzése során sikerült azonosítani 4 követelményt a kialakítandó új módszerek vonatkozásában az alábbiak szerint.

- *A megbízhatóságra tervezési és fejlesztési oldalról is törekedni kell;*
- *felhő alapú üzemeltetési környezetben is megfelelő megbízhatósági és biztonsági paraméterekkel rendelkező szolgáltatásokat kell kialakítani;*
- *magas szintű biztonságot és megbízhatóságot garantáló szabványok kialakítása és alkalmazása a cél;*
- *a biztonsági események detektálása, a logikai hibák feltárása, javítása szabványos eljárásokkal és módszerekkel történjen.*

Az alábbiakban olyan ágazatok bemutatása következik, amelyekre valamilyen formában hatással van a NIS direktíva. Léteznek olyan ágazatok, ahol az információbiztonság már korábban is megfelelően volt szabályozva, de vannak olyan ágazatok is, amelyek biztonsági környezete még formálódik. A vízi közlekedési ágazat számára teljes mértékben alkalmazandó a NIS direktíva. [23, (10)-(11)] A banki és a pénzügyi piaci infrastruktúrára vonatkozó szabályozás sok esetben szigorúbb, mint a NIS direktívában meghatározott, így ezen a területen legfeljebb jogharmonizációra van szükség. [23, (12)-(14)] Az online piacterek, az online keresőprogramok esetében a NIS direktíva az eredeti szolgáltatást nyújtó felekre vonatkozik, a közvetített szolgáltatásokat nyújtó online piacterekre és kereső oldalakra nem. [23, (15)-(16)].

Külön pontban hivatkozik az iránymutatás a felhő alapú szolgáltatásokat biztosító vállalkozásokra, azok tevékenységi körére, mint szabályozandó területre [23, (17)]. A felhő alapú szolgáltatások esetében egyaránt beszélhetünk hálózati és informatikai kockázatokról, ugyanakkor az internetes exchange pontok (IXP) esetében a műszakilag és szervezetileg különálló hálózatok kiesése jelenti a legnagyobb kockázatot [23, (18)].

A végrehajtással kapcsolatos megfontolásokat követően még egy ágazati követelményt találhatunk a közigazgatási szervekre vonatkozóan: *„Az irányelv kizárólag azon közigazgatási szervekre alkalmazandó, amelyeket alapvető szolgáltatásokat nyújtó szereplőként azonosítottak."* [23, (45)] Az ágazati helyzet feltárását követően elmondható, hogy a felhő alapú szolgáltatások kiemelt jelentőséggel bírnak minden szektorban, mert minden olyan esetben, ahol kizárólag számítási teljesítményre épülő szolgáltatásról beszélünk, elviekben lehetőség van privát vagy harmadik félhez tartozó felhő alapú szolgáltatások igénybe vételére.

Biztonsági és megbízhatósági szempontokat figyelembe véve megfontolandó olyan szervezeteknél is figyelemmel kísérni a biztonsági eseményeket, amelyek nem számítanak alapvető szolgáltatónak, de rendelkeznek egyéb informatikai rendszerekkel. Minden ágazat esetében a kockázatkezelés körébe tartozik a tárolt[14], továbbított[15] és kezelt[16] adatok biztonsága. [23, (46)] Az alapvető szolgáltatásokat nyújtó szereplők sok esetben egyéb digitális szolgáltatásokat vesznek igénybe tevékenységük során. A digitális szolgáltatókra nem vonatkozik a NIS direktíva, ugyanakkor olyan biztonsági szintet kell meghatározniuk, amely arányos az általuk nyújtott szolgáltatás mértékével, és biztosítja a folyamatos működést. [23, (47)-(49)] A hardvergyártók és a szoftverfejlesztők szerepe különösen nagy a hálózati és információs rendszerek biztonságának javítása terén, mert a megoldásaikkal fokozni lehet az alapvető szolgáltatók által nyújtott szolgáltatások biztonságát és megbízhatóságát. [23, (50)] *„Az alapvető szolgáltatásokat nyújtó szereplőkre és a digitális szolgáltatókra vonatkozó műszaki és szervezeti intézkedések nem követelhetik meg, hogy egy adott információ- vagy kommunikációtechnológiai kereskedelmi termék tervezése, kialakítása vagy előállítása valamely meghatározott módon történjék."* [23, (51)] Tehát a szóban forgó hardver, hálózati és szoftver termékek előállításával kapcsolatos konkrét követelményeket jogszabályi szinten tilos megfogalmazni, ugyanakkor ajánlást lehet tenni azok előállításának menetére vonatkozóan – meglátásom szerint erre a szoftverek esetében a lehetőség adott, azonban ennek tárgyalása a későbbiekben következik.

Biztonsági és jelentéstételi kötelezettsége az alapvető szolgáltatásokat nyújtó szereplőknek és a digitális szolgáltatóknak is van, ugyanakkor a biztonsági intézkedések vonatkozásában a költségeknek a kockázatokkal arányosnak kell lenniük. Közigazga-

[14] tárolt adatok biztonsága – túlnyomórészt hardver (HW) és szoftver (SW) kérdéseket érintő terület
[15] továbbított adatok biztonsága – túlnyomórészt hardver (HW) és hálózati (Net) kérdéseket érintő terület
[16] kezelt adatok biztonsága – túlnyomórészt szoftver (SW) és hálózati (Net) kérdéseket érintő terület

tási szervek kérhetik további biztonsági intézkedések meghozatalát szerződéses úton. [23, (52)-(54)] A közigazgatási szervek esetén lehetőség van a kiszervezésre, de ebben az esetben az adott közigazgatási szervnek – nem a szolgáltatónak – kell eleget tennie az adott hálózati vagy informatikai szolgáltatásra vonatkozó jogi szabályozásnak. [23, (56)] A különböző ágazatokhoz tartozó, alapvető szolgáltatások üzemeltetése során is lehetséges a megfelelő ajánlások kialakítása. Szoftverek esetében olyan automatizmusokra van szükség, amelyek segítségével az üzembiztonság és a megbízhatóság növelhető. Az alábbi táblázat az egyes ágazatok szoftveres/hardveres/hálózati faktorok szerinti biztonsági kitettségét mutatja be 1-től 5-ig.

Ágazat	Szoftver	Hardver	Hálózat	Adattárolás biztonsága (HW/SW)	Adattovábbítás biztonsága (HW/Network)	Kezelt adatok biztonsága (SW/Netw.)
Vízi közlekedés	3	5	5		Kritikus	
Bankszektor	5	5	5	Kritikus	Kritikus	Kritikus
Felhőalapú szolgáltatás	5	3	5			Kritikus
IXP	1	5	5		Kritikus	
Kormányzati rendszerek	5	3	5			Kritikus

2. táblázat Különböző ágazatok kiberbiztonsági fenyegetettsége (saját szerkesztés)

A 2. táblázatban szereplő értékek szubjektív mérőszámok, céljuk az ágazatok között fennálló diverzitás szemléltetése, azonban így is látható, hogy a különböző ágazatok számára más és más adatbiztonsági területek lehetnek kritikusak. Az is látható, hogy a szoftverkomponensek (SW) jelenléte két esetben is kritikus: az adatkezelési és az adattárolási szabványok létrehozásában. Ezekben az esetekben a szoftvertechnológiának, a szoftverfejlesztési módszereknek jelentős szerepe van. Az iránymutatás alapvető célja a megfelelő módszerek kidolgozása a fizikai védelemtől az informatikai biztonság kérdésköréig, figyelembe véve a *„fizikai és környezeti biztonság, az ellátás biztonsága, a hálózati és információs rendszerek integritása és az e rendszerekhez való hozzáférés ellenőrzése”* [23, (69)] területeket. Ez a követelmény azt jelenti, hogy szabályozott módszerek és folyamatok segítségével történjen a biztonsági tervek előállítása, amelyek segítségével garantálható a felsorolt területeken a biztonság. Minden területhez kapcsolódóan ágazatonként szükséges a következő képességek kialakítása: *„a biztonsági esemény kezelésére szolgáló eljárások, a biztonsági esemény észlelésének képessége, a biztonsági esemény bejelentése és az eseményről való tájékoztatás”.* [23, (69)] A kibertérben is szükségesek azok a rutinok, amelyek segítségével megfelelő tájékoztatás mellett kezelhetők, detektálhatók, bejelenthetők le-

gyenek a biztonsági események. A kibertér vonatkozásában a hálózati és az információs rendszerek térnyerését szem előtt tartva nélkülözhetetlenek *„a szolgáltatások folytonosságát szolgáló stratégiai és vészhelyzeti tervek, katasztrófa elhárítási képességek”.* [23, (69)] Az alapvető szolgáltatások folyamatos fenntartásához szükségesek azon tervek és képességek, amelyek segítségével egy biztonsági esemény bekövetkezésekor is reális cél lehet a szolgáltatás további fenntartása. A cél eléréséhez a biztonsági kockázatok kezelésére alkalmas, folyamatos működést garantáló tervezési módszerekre van szükség. Ezen megfontolásokat szem előtt tartva, a hálózati és információs rendszerek esetében különösen fontosak az üzemeltetés kapcsán *„a monitoringra és a naplózásra vonatkozó előírások, a vészhelyzeti tervek gyakorlása, a hálózati és információs rendszerek tesztelése, biztonsági értékelések és a megfelelés monitoringja”.* [23, (69)] A megfelelő reagálási képesség elősegítéséhez olyan rendszerfelügyeleti megoldások kialakítása szükséges, amelyek azonnal jelzik az alapvető szolgáltatásokban bekövetkező változásokat – biztonsági eseményeket. Mint látható újabb követelményeket sikerült azonosítani a biztonság és a megbízhatóság növelésének érdekében az alapvető szolgáltatást nyújtó infokommunikációs rendszerek tervezésével és üzemeltetésével kapcsolatban. Az elemzés során feltárt újabb követelményeket a következő felsorolás tartalmazza.

- *Szabályozott módszerek és folyamatok segítségével történjen a biztonsági tervek előállítása;*
- *megfelelő tájékoztatás mellett kezelhetők, detektálhatók, bejelenthetők legyenek a biztonsági események;*
- *biztonsági szempontok kezelésére alkalmas, folyamatos működést garantáló tervezési módszerekre van szükség;*
- *a megfelelő reagálási képesség elősegítéséhez olyan rendszerfelügyeleti megoldások kialakítása szükséges, amelyek azonnal jelzik az alapvető szolgáltatásokban bekövetkező változásokat, figyelmeztetnek a potenciális biztonsági eseményekre.*

További módszertani és technológiai meggondolást az iránymutatás nem tartalmaz, ugyanakkor érdemes röviden áttekinteni a direktíva jogi környezetét és végrehajtásának menetét is. Az alapvető szolgáltatásokat nyújtó szereplőkre és a digitális szolgáltatókra alkalmazni kell a NIS direktívában foglaltakat, azonban felmentést kapnak a 2002/21/EK európai parlamenti és tanácsi irányelv [24] hatálya alá tartozó hírközlő

hálózatokat és elektronikus hírközlési szolgáltatásokat nyújtó vállalkozások. A bizalmi szolgáltatók számára a 910/2014/EU európai parlamenti és tanácsi rendeletben [25] meghatározott szabályok vonatkoznak. [23, (7)] Indokolt esetben a NIS direktíva lehetőséget biztosít a tagállamoknak, hogy eltérjenek az irányelvtől, újabb ágazat specifikus szabályozást érvényesítsenek, ezekben az esetekben a Bizottságot tájékoztatniuk szükséges az ún. lex specialis-okról, ezen túl a tagállamoknak lehetőségük van további biztonsági előírások teljesülésének megkövetelésére. [23, (8)-(9); (57)-(65)] Párhuzamosan a speciális szabályozás lehetőségével a tagállamoknak törekedniük kell az előírások és végrehajtásuk során gyűjtött tapasztalatok alapján biztonsági szabványok kialakítására, ezt a tevékenységet az ENISA koordinálja. [23, (66)] A NIS direktíva hatálya alá nem tartozó szervezetek számára is biztosítani kell az önkéntes bejelentés lehetőségét a biztonsági események jelzésére. [23, (67)] A NIS direktívában foglaltak végrehajtásához a Bizottságot végrehajtási hatáskörrel kell felruházni, hogy el tudjon járni a célok megvalósítása érdekében. [23, (68)] A jogi környezet rövid feltárását követően is kijelenthető, hogy Uniós és tagállami szinten is külön foglalkozik az iránymutatás a biztonsági szabványok létrehozásával és a szereplők ösztönzésével.

A kritikus infrastruktúrák kijelölési eljárásával analóg módon, tagállami szinten dől el, hogy egy vállalkozás alapvető szolgáltatásokat nyújtó szereplő-e vagy sem. Az irányelv ágazatonként rendelkezik a besorolás menetéről, az ágazatok listája a szereplők változása miatt bizonyos időközönként frissül. Minden tagállamban rendszeresen el kell végezni az ágazatonkénti felülvizsgálatot a gazdasági és a társadalmi behatások miatt. Azokat a szolgáltatásokat kell vizsgálni, amelyek részét képezik az alapvető szolgáltatások jegyzékének. Egy biztonsági esemény hatásának vizsgálatakor a tagállamoknak figyelembe kell venniük az ágazatközi és az ágazat specifikus tényezőket is.

Tagállami szinten minden azonosított alapvető szolgáltatásnak szerepelnie kell az alapvető szolgáltatások jegyzékében. Ha egy szereplő több tagállamban nyújt alapvető szolgáltatást, akkor erről a tagállamoknak egyeztetniük kell a kockázatkezelés vonatkozásában. A tagállamoknak intézkedniük kell a hálózati és információs rendszerek biztonságát szabályozó kötelezettségek vonatkozásában is, ki kell jelölniük az érintett szolgáltatók körét a követelményeknek történő megfelelés alapján. Ezt követően az Európai Unió számára ágazati lebontásban, anonim módon jelezni kell az azonosított szereplők méretét, piaci részesedését. Az iménti feladatok ellátásához

minden tagállamnak meg kell alkotnia saját nemzeti hálózati és információs rendszerek biztonságával foglalkozó stratégiáját. [23, (19)-(30)] Ezen felül minden tagállamnak ki kell jelölnie egy egyedüli nemzeti kapcsolattartó pontot a határokon átnyúló együttműködés elősegítése végett. [23, (31)]

A NIS direktívában foglaltak végrehajtásáért az ENISA és az Európai Unió terültén működő CSIRT[17] hálózat felelős. Az anonim jelentéseket az adott tagállam egyedüli kapcsolattartó pontja készíti el, és adja át a Bizottságnak. A jelentésnek tartalmaznia kell a biztonsági események jellegét, a biztonsági események megsértésének típusát, súlyosságát, időtartamát. Megfelelő képességekre van szükség tagállami szinten a biztonsági események megelőzéséhez, észleléséhez, kezeléséhez és mérsékléséhez [23, (33)]. A siker eléréséhez ágazatok közötti és nemzetközi együttműködésre van szükség egyaránt. A privát szektorban az alapvető szolgáltatást biztosító szereplőket ösztönözni kell a hálózati és információs rendszerek biztonságának javítása érdekében az együttműködésre, párbeszédre. Az ENISA koordinálja ezt a folyamatot, a meglévő erőforrások felhasználása, a kapacitásépítés, az ismeretgyarapítás a legfontosabb feladat. [23, (34)-(38)] A biztonsági eseményekkel kapcsolatos információ megosztásnak nemzetközinek kell lennie a megfelelő szabályok betartása mellett, központi weboldalt (url: https://cert.europa.eu) működtetve az Unióban bekövetkező biztonsági eseményekről. [23, (39)-(41)] A CyberEurope gyakorlatok hozzásegíthetik a tagállamokat a felkészültség és az együttműködési képesség teszteléséhez, a gyakorlatok segítségével megfelelő kockázatkezelési kultúra alakítható ki. [23, (42)-(44)]. Ezzel az együttműködéssel a hálózati kockázatok jelentősen csökkenthetők, ugyanakkor az adatkezeléssel és az adattárolással kapcsolatos kérdések továbbra is nyitottak – különösen a felhőalapú szolgáltatások esetében.

A jogi környezet és az alapvető szolgáltatásokat biztosító szereplők azonosításának folyamata az eddig feltárt követelményeken túl egyéb technológiai követelményt már nem tartalmaz. Ugyanakkor elmondható, hogy olyan módszerek kialakítása a cél, amely megfelelnek az elemzés során azonosított nyolc követelménynek. A kialakítandó szabványos módszereknek a tervezés, a fejlesztés és az üzemeltetés fázisaiban is előtérbe kell helyezniük az adatkezelési és a kommunikációs kérdéseket. Ezzel a megközelítéssel növelhető a jövőben tervezett rendszerek biztonsága és megbízható-

[17] CSIRT – számítógép-biztonsági és incidenskezelő csoport, az angol **C**omputer **S**ecurity **I**ncident **R**esponse **T**eam kifejezés szavainak kezdőbetűiből kialakított rövidítés.

sága. A biztonsági kockázatok mélyebb megértéséhez szükséges a különböző szolgáltatás típusok alaposabb vizsgálata.

1.3.3 A biztonság és a szolgáltatás típusok kapcsolata

A kritikus infrastruktúrák biztonsága elemi kérdés, például a villamos áramot biztosító erőművek és az elektromos hálózat működéséhez szükséges fizikai eszközök összessége létfontosságú társadalmunk számára. Napjainkra a fizikailag létező rendszerek működtetéséhez is túlnyomórészt szoftverek szükségesek – még abban az esetben is, ha nem infokommunikációs szolgáltatásról beszélünk. Az említett szolgáltatásokban részt vevő elektronikai eszközök vezérlése természetesen szoftveresen történik – tehát az említett szoftverek is részét képezhetik a fizikailag létező kritikus infrastruktúráknak. Ezen a ponton tisztázni kell szoftvertechnológiai szempontból a szoftver és az információs rendszerek fogalmának kapcsolatát. Egyaránt szoftverként tekintendő egy ipari gyártósor PLC[18] vezérlőjének programja, egy nyomtatóhoz vagy webkamerához tartozó telepített alkalmazás, de egy vállalatirányítási rendszer is. A fizikai eszközök működtetéséhez általában „alacsony" szintű szoftverekről, vezérlőkről beszélhetünk.

Korábban az elektronikai eszközök elhanyagolható hányada rendelkezett hálózati kapcsolattal, azonban az új évezredben az újonnan gyártott digitális „kütyük" jelentős része már hálózati kapcsolattal is rendelkezik. Ez a képesség és az Internet széleskörű elterjedése tette lehetővé az IoT[19] kialakulását, ami a kibertér egy meghatározó szegmensévé vált napjainkra. A hálózathoz történő kapcsolódás képessége az ipari felhasználás során a központi felügyelet és a vezérlés miatt jelent meg. A hétköznapi életben az okos elektronikai eszközök és mobil eszközök közötti kommunikáció vált elterjedt megoldássá. Az említett megoldások kifejlesztésével párhuzamosan megjelent a fenyegetés a driverek, a szoftverfrissítések, a potenciális biztonsági rések irányából. Az ipari gépek, az okos eszközök és a vezérlőik között kialakult kapcsolatok már egyfajta elosztott információs rendszernek tekinthetők. A hagyományos információs rendszerek esetei a telekommunikációs, banki, kormányzati rendszerek, amelyek esetében valamilyen üzleti folyamat teljes mértékben szoftverek

[18] PLC – programozható logikai vezérlő, az angol **P**rogrammable **L**ogical **C**ontroller kifejezés szavainak kezdőbetűiből kialakított rövidítés.
[19] IoT – a dolgok internete, az angol **I**nternet **o**f **T**hings kifejezés szavainak kezdőbetűiből kialakított rövidítés.

segítségével megy végbe. Ezekben az esetekben lehetőség nyílik a felhőalapú fejlesztési és üzemeltetési környezet kialakítására.

A felhő alapú technológia elterjedésével különböző típusú szolgáltatás típusok is megjelentek. A különbség a szolgáltatás típusok között az igénybe vett szolgáltatások aránya. A legalapvetőbb szolgáltatás típus az IaaS[20], ilyenkor a szolgáltató által kezelt és biztosított komponensek száma a legkevesebb. A szolgáltató biztosítja a fizikai eszközöket, szervereket, tárhelyet, hálózati komponenseket, és szolgáltatást igénybe vevő kezeli a szoftvereket. Ha egy alapvető szolgáltatás IaaS típusú, akkor a biztonságnak és a megbízhatóságnak két pilléren kell állnia. Megbízható fizikai eszközök alkalmazása mellett, a biztonságot garantáló geo-redundáns adattárolás, tartalék rendszerek kialakítása szükséges. A biztonság és megbízhatóság az IaaS típusú szolgáltatások esetében túlnyomórészt nem szoftvertechnológiai, hanem infrastruktúra tervezési kérdés. A fizikai eszközök által nyújtott szolgáltatásokra épülő következő szolgáltatás típus a PaaS[21], amely alapvető szoftverek jelenlétét feltételezi a felhőben, amelyek segítségével lehetőség van egyedi szoftverek előállítására, ezeket a szoftvereket a szolgáltató biztosítja. Ilyen szoftverkomponensek az operációs rendszerek, webszerverek, adatbázis szerverek, programozási nyelvek. A biztonsági és a megbízhatósági kérdések itt a szoftverek előállításával kapcsolatos folyamatokat érintik. A PaaS típusú szolgáltatások esetén a biztonság növeléséhez a szoftverfejlesztés során használt szoftver komponensek, fejlesztői eszközök, programozási nyelvek megbízhatóságáról kell meggyőződni. A megbízhatóság növeléséhez a szoftverfejlesztés során használt módszerekkel és eszközökkel lehet javítani. Az automatizált tesztelést támogató, a manuális és gépi kódellenőrzést végrehajtó eszközök platformot adnak a jobb minőségű szoftverek előállításához. Amennyiben az alapvető szolgáltatás PaaS típusú, akkor a biztonságot a hiteles, megbízható szoftverfejlesztőtől származó szoftverkomponensek adják, azonban a megbízhatóságot a platformon fejlesztett szoftverekre vonatkozóan a megfelelő szoftverfejlesztést támogató eszközök jelenthetik.

[20] IaaS – Infrastruktúra alapú szolgáltatás, az angol **I**nfrastructure **a**s **a** **S**ervice kifejezés szavainak kezdőbetűiből kialakított rövidítés.

[21] PaaS – Platform alapú szolgáltatás, az angol **P**latform **a**s **a** **S**ervice kifejezés szavainak kezdőbetűiből kialakított rövidítés.

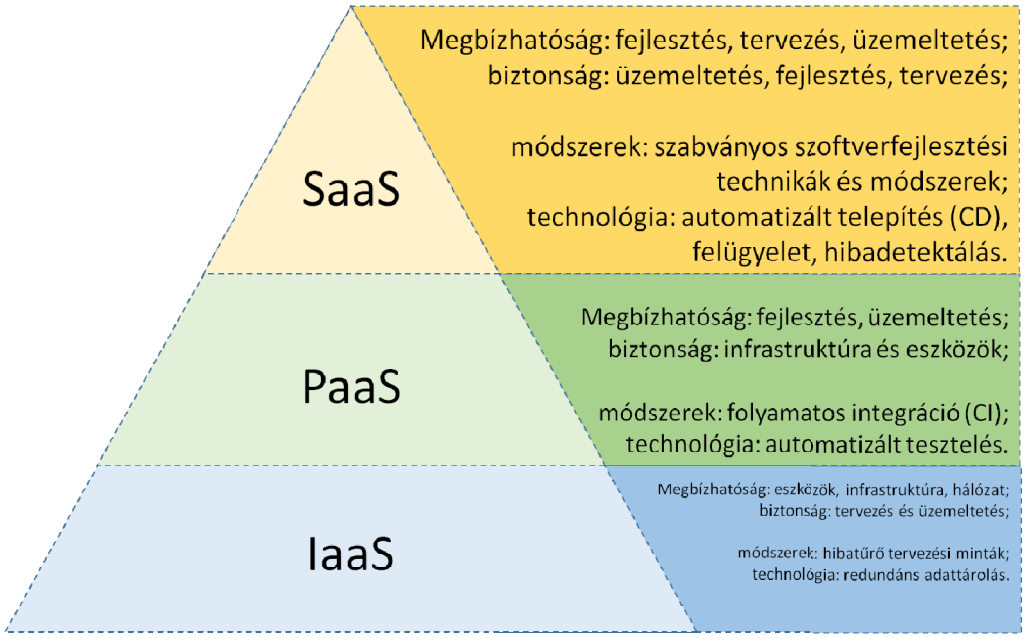
Abban az esetben, ha egy felhőalapú szolgáltatás tárgyát valamilyen szolgáltatásként igénybe vett szoftver képezi, akkor a SaaS[22] szolgáltatás típusról beszélünk. Az ilyen típusú szolgáltatások esetén a szolgáltatást igénybe vevő fél egy célszoftver felületét látja és használja, minden egyéb technológiai kérdésért a szolgáltató felel, beleértve a szolgáltatás alapját képező szoftver fejlesztését és üzemeltetését is. Gyakran fordul elő, hogy különböző vállalatirányítási-, banki- és kormányzati rendszerek ebben a konstrukcióban érhetők el a megrendelők számára. Egy ilyen jellegű szolgáltatás egy kiszervezett képességet takar, amelyért rendszeres díjazás mellett szoftver alapú szolgáltatást kap cserébe a megrendelő. A fizikai hálózatok kiesése, a hardveres sérülés e szolgáltatás típus esetében is kritikus következményekkel járhat, ugyanakkor a szoftveres kockázatok, hibás működés, információlopás, rendszerleállás legalább ekkora kockázatot rejtenek.

A biztonság és a megbízhatóság kérdése különösen érdekes terület a SaaS típusú szolgáltatások esetében. A biztonság vonatkozásában a korábban tárgyalt IaaS és PaaS biztonsági kérdésein túl, szoftvertechnológiai kérdésekkel is találkozhatunk a szolgáltatott szoftverek esetében. Ilyen biztonsági kérdés az, hogy az illetéktelen behatolás, az jogosulatlan adatok kinyerése megfelelően le van-e védve? Az üzemeltetés során elegendő és megfelelő minőségű diagnosztikai adattal rendelkezik az szolgáltatás biztosító? Szabványosan folyamatok segítségével valósul meg az üzemeltetés? Rendszerhiba esetén megoldott-e a megfelelő bejegyzések létrehozása a technikai naplófájlokban? Megbízhatósági oldalról a következő kérdések merülhetnek fel. A szenzitív adatok (például: személyes adatok, pénzügyi adatok) modellezése megfelelő módon történik-e? Lehetséges-e szabályozott módon a szenzitív adatok kinyerésének korlátozása? Rendelkezésre állnak-e anonim adatok, elkódolt adatok a hibakeresések támogatásához? Verziófrissítések során korábban működő funkciók nem tűnnek el, a rendszer sebessége nem lassul, nincsenek időtúllépések a válaszidőkben. Az adott szoftverből elérhető funkciók helyesen működnek, váratlan hibaüzenetektől mentes-e a rendszer? Összegezve a fenti kérdéseket egyetlen kérdésbe az adott szoftver alapú szolgáltatás fejlesztése és üzemeltetése szabványosan történik-e?

Az ENISA által felügyelt CSIRT hálózat védelmet biztosít a rosszindulatú programokkal és vírusokkal szemben, megfelelően támogatva az infrastruktúra alapú szolgáltatási modellt, és részben támogatva a felette lévő platform és szoftver alapú szol-

[22] SaaS – Szoftver alapú szolgáltatás, az angol **S**oftware **a**s **a** **S**ervice kifejezés szavainak kezdőbetűiből kialakított rövidítés.

gáltatási rétegeket. A megfelelő platformok, szabványosított módszerek és technológiák meghatározása a szoftverfejlesztés és üzemeltetés támogatásához előretekintő feladat. Szoftvertechnológiai kihívásként is felfogható a kibertérben elérhető PaaS és SaaS felhő alapú szolgáltatási modellek biztonságának és megbízhatóságának fokozása. A minőségi mutatók garantálásán túlmenően célnak kell lennie a különböző biztonsági események és fenyegetések esetén a gyors reagálási képesség megteremtésének. A 2. ábrán a biztonsági és megbízhatósági tényezők javításához szükséges módszerek és technológiák jelennek meg a szolgáltatási modell háromszög rétegeire értelmezve.



2. ábra Felhő alapú szolgáltatási modell háromszög (saját szerkesztés)

Alulról felfelé haladva csökken a hardver- és a hálózattervezés jelentősége. A biztonsági kérdések tekintetében ugyanebben az irányban a szoftvertechnológia fontossága folyamatosan növekszik. Az IaaS, PaaS és SaaS szolgáltatási modellek számára a háromszög minden rétegében megjelennek az ajánlott módszerek. A platform és a szoftver alapú szolgáltatási rétegekben a megfelelő módszertan és technológia együttes alkalmazása javíthatja a megbízhatóságot és a biztonságot. Mindazonáltal, a platform és az infrastruktúra alapú szolgáltatási rétegek minőségi tényezői egyaránt javíthatók megbízható eszközökkel és szoftverkomponensekkel, hibatűrő fizikai infrastruktúrával.

A SaaS szolgáltatási modellben a megbízhatóság és a biztonság a szoftver tervezéséhez, fejlesztéséhez és működéséhez kapcsolódik. A megbízhatóság szempontjából a szoftverfejlesztéssel kapcsolatos tevékenységek kritikusabbak, míg a biztonság terü-

letén az üzemeltetési környezet a lényegesebb. A minőségi mutatók javítása automatizálással érhető el, amellyel a tervezés, a fejlesztés és az üzemeltetés területeinek együttműködése megfelelően elősegíthető. Az automatizált telepítési folyamatok, a rendszerfelügyelet és a hibák folyamatos észlelése együtt képesek a biztonság és a megbízhatóság technikai hátterének biztosítására. A szabványos szoftverfejlesztési módszerek alkalmazásával a szoftverek előállítása során az üzemeltetési környezet kívánalmainak megfelelő rendszerek kialakítása válik lehetővé.

Az ilyen szoftverfejlesztéssel kapcsolatos tevékenységeket általában PaaS szolgáltatási modellben végzik. A folyamatos integráció (röviden: CI[23]) módszere abban támogatja fejlesztőket, hogy megfelelő visszajelzések alapján folyamatosan tudják bővíteni a fejlesztés alatt álló rendszert. A módszer alkalmazásához automatikus tesztelő eszközök bevonására van szükség. Amennyiben ez megvalósul, akkor a rendszerrel szemben támasztott követelmények automatizált tesztek formájában is létrejönnek, egyfajta réteget képezve az éles üzemben futó forráskód köré. Az iménti módszerek segítségével a felhő alapú szolgáltatási háromszög felső két rétegében lehetővé válik a jobb minőségű szoftverek előállítása.

Az új szoftvertechnológiai szabványok kialakítása során figyelembe kell venni a követelményelemzési technikákat, szoftverfejlesztési módszereket és az üzemeltetési környezetet is. Az új szabványok széles körű használata esetén az információs rendszerek nemcsak hardver és hálózati szemszögből, hanem szoftvertechnológiai oldalról is biztonságosabbá válhatnak. Azonban továbbra is igaz marad, hogy mind a fizikai-, mind a kiberbiztonsági szempontokat is figyelembe kell venni a hálózati és információs rendszerek magasabb szintű biztonságának elérése érdekében.

1.3.4 Intézkedések összegzése

Az elmúlt tizenöt évben az Európai Unió meghatározta az európai kritikus infrastruktúrák védelme érdekében elérendő célkitűzéseit és kidolgozta a kapcsolódó szabályozási mechanizmusokat. Az alapvető szolgáltatások védelmére irányuló intézkedések elengedhetetlenek a digitális társadalom biztonságának fokozásához. Az elemzés során megismertük a NIS irányelvet, amely meghatározza a kritikus infrastruktúrák védelmét a kibertérben az EU kiberbiztonsági stratégiájával összhangban.

Az alapvető szolgáltatások azonosításának, kijelölésének és kezelésének folyamata jól szabályozott, hasonlóan a kritikus infrastruktúrákkal védelme során végrehajtandó

[23] CI – **C**ontinuous **I**ntegration, a funkciók folyamatos integrálása a fejlesztett szoftverbe.

eljárásokhoz. A CSIRT hálózat megfelelően támogatja tagállami szinten a széles körben használt szoftverek sebezhetőségeinek felismerését és a számítógépes hálózatok biztonsági eseményeinek megelőzését. A biztonsági események kármentéséhez elegendő riasztási és reagálási eljárás áll rendelkezésre.

Kijelenthető, hogy az EU által kialakított keretrendszer megfelelően támogatja a számítógépes hálózatok és a fizikai infrastruktúra védelmét, a veszélyek elhárítását, azaz a NIS irányelv kellőképp elősegíti az infrastruktúra alapú szolgáltatási modell védelmét. Azonban az is kiderült az elemzésből, hogy az új szoftvertechnológiai szabványok kidolgozása során figyelembe kell venni az adatátvitel és az adattárolás biztonsági kérdéseit is. A felhőalapú szolgáltatási modellek kockázatainak felülvizsgálata során felszínre került, hogy a biztonsági és megbízhatósági kérdések szemszögéből a platform és a szoftver alapú szolgáltatási modellek esetében a szoftverfejlesztés és az üzemeltetés folyamatát csak részben fedi le a dokumentum.

Az új szabványoknak, mint integrált szoftvertechnológiai megközelítésnek lehetővé kell tenniük az automatizálást a követelmények azonosításától a fejlesztésen át, egészen az üzemeltetésig a felhő alapú szolgáltatások szoftver és platform szolgáltatási rétegeiben.

A NIS irányelvben meghatározott biztonsági és megbízhatósági mutatók javítása mindkét szolgáltatási modell esetében lehetséges. A jövőben kialakítandó szabványos módszerek számára az elemzés során azonosított nyolc információbiztonsági követelmény megfelelő alapvetéseket tartalmaz.

I-1 A megbízhatóságra tervezési és fejlesztési oldalról is törekedni kell. Lásd: 23. old.

I-2 Felhő alapú üzemeltetési környezetben is megfelelő megbízhatósági és biztonsági paraméterekkel rendelkező szolgáltatásokat kell kialakítani. Lásd: 23. old.

I-3 Magas szintű biztonságot és megbízhatóságot garantáló szabványok kialakítása és alkalmazása a cél. Lásd: 23. old.

I-4 A biztonsági események detektálása, a logikai hibák feltárása, javítása szabványos eljárásokkal és módszerekkel történjen. Lásd: 23. old.

I-5 Szabályozott módszerek és folyamatok segítségével történjen a biztonsági tervek előállítása. Lásd: 26. old.

I-6 Megfelelő tájékoztatás mellett kezelhetők, detektálhatók, bejelenthetők legyenek a biztonsági események. Lásd: 26. old.

I-7 Biztonsági szempontok kezelésére alkalmas, folyamatos működést garantáló tervezési módszerekre van szükség. Lásd: 26. old.

I-8 A megfelelő reagálási képesség elősegítéséhez olyan rendszerfelügyeleti megoldások kialakítása szükséges, amelyek azonnal jelzik az alapvető szolgáltatásokban bekövetkező változásokat, figyelmeztetnek a potenciális biztonsági eseményekre. Lásd: 26. old.

1.4 KIBERBIZTONSÁG MAGYARORSZÁGON

A világ nagyhatalmai esetében is megfigyelhetők a különböző biztonsági stratégiák, amelyekben a kockázatok és a fenyegetések mellett potenciális védelmi válaszokat is találhatunk. Az Amerikai Egyesült Államok és Nagy-Britannia esetében egyaránt nemzeti biztonsági stratégiákkal találkozhatunk, míg Franciaország jellemzően Védelmi Fehér Könyv formájában határozza meg potenciális kihívásokat, fenyegetéseket és az azokra adandó válaszokat. Oroszország és a Kínai Népköztársaság is rendelkezik hasonló stratégiai dokumentumokkal, amelyek alapját képezik a kihívásokhoz igazított haderőfejlesztésnek. Az is megfigyelhető, hogy bizonyos időközönként a stratégiai dokumentumok felülvizsgálata is szükséges a megváltozott biztonsági környezet miatt. Egy ilyen új tényező a kiberbiztonság kérdése, amely egyre nagyobb hangsúlyt kap a különböző biztonsággal foglalkozó stratégiai dokumentumokban. A továbbiakban a különböző hazai stratégiai dokumentumok elemzése következik a kiberbiztonság szemszögéből vizsgálódva.

1.4.1 Magyarország Nemzeti Biztonsági Stratégiája

A bemutatott nemzetközi stratégiai dokumentumok magyar megfelelője Magyarország Nemzeti Biztonsági Stratégiája [26], a 2020 áprilisában kihirdetett dokumentum a 2012-ben megjelent [27] biztonsági stratégiát váltotta. A korábbi stratégia négy fejezetben foglalta össze a magyar biztonságpolitika kihívásait és a stratégiai válaszait, amely a dolgozat szempontjából továbbra is lényeges és a hatályos stratégiával összhangban lévő aktuális gondolatokat tartalmaz.

1. fejezet: Magyarország biztonságpolitikai környezete. A dokumentum leszögezi, hogy a *„21. század elején is előfordulhat, hogy a katonai erő kap elsődleges szerepet egy regionális konfliktusban”* [27, 4], ugyanakkor *„a biztonság katonai szegmense is új hangsúlyokkal jelenik meg”* [27, 2]. Az iménti két megállapítás a hagyományos értelemben vett katonai fenyegetéseken túl, a kibertérben megje-

lenő fenyegetésekre is teljes mértékben értelmezhető. A hibrid hadviselés[24] szerves részeként megjelenő kiberhadviselés[25] erre a legjobb példa. Ezen túlmenően a terrorizmus és az illegális bevándorlás napjainkban is valós és folyamatosan jelenlévő fenyegetést jelent, ezért gondolom úgy, hogy ez a megállapítás manapság is megállja a helyét, miszerint *„globalizált világunkban a biztonság nem a határainknál kezdődik”* [27, 5] – legyen az a fizikai valóság vagy a kibertér. A felhő alapú szolgáltatások korában, az információs biztonság vonatkozásában a következő követelményt fogalmazhatjuk meg, *a különböző biztonsági komplexumhoz köthető információs rendszerek kialakítása során egyaránt törekedni kell a biztonságra.*

2. fejezet: Magyarország helye és biztonságpolitikai érdekei a világban. A Biztonsági Stratégia megállapítja, hogy *„Az Észak-atlanti Szerződés 5. cikke, a kollektív védelem Magyarország biztonságának sarokköve”* [27, 13], valamint *„Magyarország NATO- és EU-keretekben folytatott biztonságpolitikai tevékenysége globális és átfogó jelleget ölt”* [27, 18]. A különböző missziós tevékenységen túl a nemzetközi- és különösen a magyar kibertér biztonságának óvása is nemzeti ügy. Még nagyobb jelentőséggel bír a kibertér védelme 2016 júliusát követően, amikor is a NATO a kiberteret hadszíntérré nyilvánította. Így a NATO-nak ugyanolyan hatékonyan meg kell tudnia védenie magát a kibertérben, mint a levegőben, a szárazföldön vagy a tengeren. [28] *Olyan infokommunikációs technológiák és módszerek kialakítása szükséges, amelyek biztosítják a megfelelő védekezési és reagálási képességeket*.

3. fejezet: a Magyarországot érintő biztonsági fenyegetések, kihívások és azok kezelése. A klasszikus fenyegetéseken túl megjelenik a Kiberbiztonság megóvásának kérdése is a Biztonsági Stratégiában. A dokumentum úgy fogalmaz, hogy *"egyre sürgetőbb és összetettebb kihívásokkal kell számolnunk az informatikai- és telekommunikációs hálózatok, valamint a kapcsolódó kritikus infrastruktúra*

[24] Hibrid hadviselés -„a hibrid fenyegetések a hadviselés számos formáját magukban foglalják, beleértve a konvencionális képességeket, irreguláris harceljárásokat és képződményeket, valamint a válogatás nélküli erőszakot alkalmazó terrorista akciókat és bűnözői tevékenységeket. Hibrid háborúkat egyaránt folytathatnak állami és a legkülönfélébb nem állami szereplők. Az egymástól elszigetelten működő egységek, vagy akár ugyanaz a csoport is folytathat „multimodális” tevékenységeket, de ezek általános, műveleti, valamint harcászati irányítása és koordinálása a fő hadszíntéren megy végbe, annak érdekében, hogy a szinergikus hatások bekövetkezzenek a konfliktusok pszichológiai és fizikai dimenzióiban. Ezen hatások a háború valamennyi szintjén jelentkezhetnek.” [29]

[25] Kiberhadviselés - a kiberhadviselés a (kritikus) információs infrastruktúrák bizalmasságának, sértetlenségének és rendelkezésre állásának befolyásolására irányuló tevékenység informatikai, fizikai és emberi eszközökkel. [30]

fizikai és virtuális terében" [27, 31]. Az infokommunikációs hálózatok, informatikai szolgáltatások sérülékenységeit kihasználva ellenségesen fellépő államok, terrorista csoportok jelentős károkat okozhatnak. A dokumentum a hazai kibervédelem megteremtésén túl a nemzetközi védelemben történő részvételt határozza meg elérendő célként, tehát *a számítógépes hálózatokat és a kapcsolódó információs rendszereket fel kell készíteni az összehangolt támadásokkal szemben.*

4. fejezet: a felbukkanó fenyegetések ellen történő hatékony fellépés érdekében *"erősíteni kell a honvédelmi, nemzetbiztonsági, rendvédelmi, igazságszolgáltatási, katasztrófavédelmi és polgári válságkezelési intézmények szoros és hatékony együttműködését"* [27, 43]. Az említett együttműködés magas szintű támogatása többek között a kibertér egy speciális, védelmi célú szektorában valósulhat meg az érintett szereplők közötti magas szintű információcsere révén. Tehát, *szoros együttműködést, hatékony információcserét biztosító infokommunikációs rendszerek kialakítása szükséges a védelmi ágazat szereplői számára.*

A korábbi biztonsági stratégia lényeges kérdéseket tárgyalt a kiberbiztonság témakörében, azonban az alkalmazható eszközrendszerre még nem tért ki megfelelő mélységben. A 2020-as biztonsági stratégia részletesen tárgyalja a regionális és globális biztonsági kihívásokat, külön kiemeli a védelmi ipar fejlesztését [26, 1.2], a forradalmi technológiák alkalmazását a védelmi ágazatban [26, 4.105-107], ezáltal tovább erősíti a korábbi dokumentum alapján feltárt szoftvertechnológiához köthető általános kiberbiztonsági követelményeket.

K-1 A különböző biztonsági komplexumhoz köthető információs rendszerek kialakítása során egyaránt törekedni kell a biztonságra, különös tekintettel a katonai biztonságra. Lásd: 36. old.

K-2 Olyan infokommunikációs technológiák és módszerek kialakítása szükséges, amelyek biztosítják a megfelelő védekezési és reagálási képességeket. Lásd: 36. old.

K-3 A számítógépes hálózatokat és a kapcsolódó információs rendszereket fel kell készíteni az összehangolt támadásokkal szemben.

K-4 Szoros együttműködést, hatékony információcserét biztosító infokommunikációs rendszerek kialakítása szükséges a védelmi ágazat szereplői számára.

1.4.2 Magyarország Nemzeti Katonai Stratégiája

A Katonai Stratégia bővebben tárgyalja Magyarország biztonsági környezetét, hasonlóan a Biztonsági Stratégiához a NATO és az EU tagságot határozza meg Magyarország biztonságának zálogaként, különös tekintettel a NATO Washingtoni Szerződés 5. cikke alapján megvalósuló kollektív védelemre és az EU Lisszaboni Szerződés kölcsönös segítségnyújtási és szolidaritási klauzuláira. A Magyar Honvédség működési környezetén belül is megjelenik a kibertér fogalma, mely új kihívások és potenciális veszélyek forrása lehet [31, 33]. A haderő várható alkalmazási területei között megjelenik a kibertérhez kapcsolódóan a hálózatalapú hadviselés a különböző válságok kezelése során [31, 41]. A technikai területen túlmutatva az információs műveletek súlya drasztikusan növekedik, a megfelelő tájékoztatás, a média, a digitális információáramlás eszközeinek magas szintű felhasználása kulcsfontosságúvá válik [31, 51]. A Katonai Stratégia megállapítja, hogy a digitális eszközök, szolgáltatások, a lakosság vagy akár a védelmi feladatot ellátó erők ellen elkövetett, nem kimondottan fegyveresen – inkább technikai-, digitális eszközökkel – végrehajtott támadások aránya növekedik. Ezt a fajta új hadviselés jelentős anyagi károkat és káoszt okozhat, megközelítve a hagyományos fegyverek potenciálját. [31, 52]

A Magyar Honvédség kialakítandó képességeivel kapcsolatban a stratégia valamenynyi hadrendi elemre vonatkozó legalább alap szintű támadási képesség meglétét célozza meg. [31, 69]. A Katonai Stratégia végén a hálózatalapú hadviselés feltételeinek megteremtése jelenik meg célként, mely a kibertér egy speciális katonai célú alkalmazását jelenti, ahol már nem csak a védelmi feladatok ellátása a cél, hanem különböző jellegű műveletek kibertérben történő támogatása is. Ezen a területen a technikai eszközök beszerzése, fejlesztése, valamint az állomány felkészítése, továbbképzése egyaránt feladat.

A Katonai Stratégiában megjelenik kibertér és a kiberhadviselés fogalma. Napjainkra a védelmi képességeken túl az alapszintű támadási képességek fejlesztése is cél lehet. A megfelelő követelmények megfogalmazása és a technológiai eszközök kiválasztása a hálózat alapú hadviselés, az információs műveletek és a kiberhadviselés területén is kulcsfontosságú kérdésnek tekinthetők. A védelmi tevékenység esetében rendelkezésre áll az a jogszabályi környezet, amely meghatározza a kritikus infrastruktúrák biztosítása érdekében elvégzendő feladatokat.

A biztonság vonatkozásában a kibertérre kiterjesztett stratégiai szintű dokumentum Magyarország Nemzeti Kiberbiztonsági Stratégiája [32] (röviden: NKS). Az NKS

magát a következő módon határozza meg: *„a nemzeti vagyon részét képező nemzeti adatvagyon, valamint a kapcsolódó létfontosságú rendszerek és létesítmények kiberbiztonságának dokumentuma”* [32, 2]. A stratégia deklarálja, hogy igazodik az Európai Unió kiberbiztonsági stratégiájához és a NATO kibervédelmi politikájához [33] és a lisszaboni, valamint a chicagói NATO-csúcsokon megfogalmazott kibervédelmi elvekhez és célokhoz. A stratégia a magyar kiberbiztonsági környezetet a következő módon mutatja be: nagy fenyegetést jelent a kibertérben megvalósuló információs hadviselés, ugyanakkor *„a külső károkozások mellett további kockázatot jelent, hogy a kibertér alkotóelemeiként szolgáló informatikai és hírközlési rendszerek üzembiztonsági szabályozása sem kellően rendezett.” … „Jelen stratégia fő célja annak a döntéshozó politikai és szakmai figyelemnek és képességnek a kiépítése, mely rugalmasan reagálva lehetővé teszi a belátható jövőben a technológiai fejlődésből fakadó új kiberbiztonsági problémák kezelését.”* [32, 4] Az idézett gondolatok alapján azt a következtetést lehet levonni, hogy egyaránt fel kell készülnünk külső és belső eredetű kockázatokra a magyar kibertér biztonságának megteremtése és megóvása érdekében – tehát kijelenthető, stratégiai cél a megfelelő eszközök, módszerek kiválasztása és olyan szabályok alkalmazása, amelyekkel a kívánt biztonsági szint elérhető. A stratégia megállapítja, hogy a biztonságos kibertér megteremtése az egyének, közösségek, gazdasági szereplők, kormányzati szervek és a jövő generációi számára egyaránt stratégiai cél Magyarországon. [32, 8]

A meghatározott célok eléréséhez összkormányzati koordinációra, a civil, a gazdasági és a tudományos területek együttműködésére, valamint a szakosított intézmények (GovCERT) hatékony fellépésre van szükség a Kiberbiztonsági Stratégia alapján. [32, 10. a)-h)] A gazdasági szereplők motivációja [32, 10. i)] jelen tanulmány szemszögéből kiemelten hangsúlyos követelmény, ugyanis ebben a pontban stratégiai célként jelenik meg a kiberbiztonsági követelmények meghatározásához, a kiberbiztonság fokozásához a szükséges módszerek, technológiák alkalmazásának elősegítése.

Az NKS végrehajtásához szükséges eszközrendszerben számos politikai, illetve adminisztratív kormányzati teendő jelenik meg [32, 11. a)-h)], ám ezt követően külön pont foglalkozik a műszaki szempontokkal, konkrétan megjelenik *„a kiberbiztonsági szempontok érvényesítése az állami műszaki fejlesztési feladatok, valamint a kormányzati információs rendszerek fejlesztésével és üzemeltetésével kapcsolatos feladatok ellátása során.”* [32, 11. i)] Az utolsó feladat ellátásához megfelelő módszer-

tani, technológiai háttérre van szükség, ahol már szükségesek a felhasznált technológiákkal szemben alkalmazható követelmények és mérőszámok.

A Nemzeti Kiberbiztonsági Stratégia megfelelően helyezi el a magyar kiberteret a nemzetközi kibertérben és jó alapot teremt a XXI. század kihívásaira való felkészüléshez. Erre a stratégiára épül a 2013. évi L. törvény [34] és a 185/2015. (VII. 13.) Korm. rendelet [35]. Előbbi az állami és önkormányzati szervek elektronikus információbiztonságát, utóbbi a kormányzati eseménykezelő központokkal kapcsolatos feladatokat és eseményeket tárgyalja. A 2013. évi L. törvény 1. fejezetében átfogó fogalomjegyzékkel találkozhatunk, az alábbiakban néhány fontos definíció található.

1. kiberbiztonság: *„a kibertérben létező kockázatok kezelésére alkalmazható politikai, jogi, gazdasági, oktatási és tudatosságnövelő, valamint technikai eszközök folyamatos és tervszerű alkalmazása, amelyek a kibertérben létező kockázatok elfogadható szintjét biztosítva a kiberteret megbízható környezetté alakítják a társadalmi és gazdasági folyamatok zavartalan működéséhez és működtetéséhez”* [34, 1.26]
2. kibervédelem: *„a kibertérből jelentkező fenyegetések elleni védelem, ideértve a saját kibertér képességek megőrzését”* [34, 1.27]
3. logikai védelem: *„az elektronikus információs rendszerben információtechnológiai eszközökkel és eljárásokkal (programokkal, protokollokkal) kialakított védelem;”* [34, 1.34]
4. sérülékenységvizsgálat: *„az elektronikus információs rendszerek gyenge pontjainak (biztonsági rések) és az ezeken keresztül fenyegető biztonsági események feltárása;”* [34, 1.41]

A törvény hatálya alá eső elektronikus információs rendszerek esetében biztosítani kell a kezelt adatok és információk bizalmasságát, sértetlenségét és rendelkezésre állását és az adott rendszer zárt, teljes körű, folytonos és kockázatokkal arányos védelmét [34, 5]. Eme követelmény teljesülését a Kormány által kijelölt hatóság ellenőrzi. Az adott hatóságnak tevékenysége során még ellenőriznie kell egy adott szervezet által használt elektronikus információs rendszer osztályba sorolását és biztonsági szintjét, valamint az adott osztálynak és biztonsági szintnek megfelelő követelmények teljesülését.

A hatóság feladata az ellenőrzés során feltárt biztonsági hiányosság elhárításának elrendelése [34, 14]. A felügyeletet ellátó hatóság elrendelheti egy elektronikus információs rendszer sérülékenységvizsgálatát, valamint egy biztonsági esemény ki-

vizsgálását is [34, 18]. A törvény ezen kívül részletesen tárgyalja a kormányzati eseménykezelő központ hazai- és nemzetközi feladatait, valamint a polgári, nemzetbiztonsági és honvédelmi célú eseménykezelő központok hatáskörét.

A 185/2015. (VII. 13.) Korm. rendelet konkretizálja az egyes eseménykezelő központokat, hatásköröket, valamint a sérülékenységvizsgálatot végezhető gazdasági szervezetek és személyek körét. A tárgyalt kormányrendelet konkrét követelményeket fogalmaz meg a sérülékenységvizsgálatra vonatkozóan.

1) vizsgált területek:
   a) *„külső informatikai biztonsági vizsgálat*
   b) *webes vizsgálat*
   c) *belső informatikai biztonsági vizsgálat, illetve*
   d) *vezeték nélküli hálózat informatikai biztonsági vizsgálata”* [35, 16 (1)]
2) jogosultsági fázisok
   a) „*regisztrált felhasználói jogosultság nélküli vizsgálat*
   b) *regisztrált felhasználói jogosultsággal rendelkező vizsgálat és adminisztrátori jogosultsággal rendelkező vizsgálat”* [35, 16 (2)]

A sérülékenységvizsgálat eredményét az érintett szerv és a hatóság is megkapja, amely alapján mindketten kötelesek eljárni. A tárgyalt két dokumentum a sérülékenység vizsgálat és a biztonsági esemény kivizsgálásának eljárását és a kapcsolódó határidőket írja le, de további technológiai követelményeket nem fogalmaz meg.

A Nemzeti Kiberbiztonsági Stratégia az EU kiberbiztonsági stratégiájának magyar leképezéseként jelent meg, a témakörben áttekintett jogszabályok az NKS-ből kerültek levezetésre. Az Uniós szabályozás a NIS irányelv segítségével fejlődött tovább, amely a tagállamok számára is feladatul szabta a nemzeti hálózat- és információbiztonsági stratégiai dokumentumok megfogalmazását.

A NIS irányelvben foglaltaknak megfelelően Magyarország 2018. december 28-án a 1838/2018 Korm. határozattal hozta nyilvánosságra a hálózati és információs rendszerek biztonságára vonatkozó Stratégiáját [36] (a továbbiakban: HIRbizt. stratégia). A következő elemzésben a HIRBizt. stratégiában fellelhető szoftvertechnológiai szempontból releváns megállapítások és intézkedések vizsgálata következik követelmények után kutatva.

1.4.3 Hálózati- és információs rendszerek biztonsága

A HIRBizt. stratégia előszavából kiderül, hogy a hazai digitális környezet fejlődésével egyidejűleg az állami szervezetek és állampolgárok is egyre szélesebb rétegben használják ki az elektronikus információs rendszerek adta lehetőségeket. A széleskörű felhasználás mellett a különböző kockázatok száma is növekedik a minden napi alkalmazás során. Napjainkban is igaz, hogy az infokommunikációs technológia fejlődése továbbra is hatalmas potenciált rejt magában, ugyanakkor ez a fejlődés teret biztosít a terrorizmus és a kiberbűnözés számára is. A biztonságos kibertér megteremtése és fenntartása közös feladat, melyben részt kell vennie az állami és a piaci szereplőknek, az állampolgároknak és a kiberbiztonsággal foglalkozó szakembereknek egyaránt. Meglátásom szerint minden szoftvertechnológiai vonatkozásban érintett, a szoftverek előállítási folyamata során résztvevő szereplőnek – rendszerszervezőnek, szoftverfejlesztőnek és tesztelőnek – szem előtt kell tartania a kiberbiztonsági kérdéseket, ez a feladatrendszer napjainkra már nem csak a kiberbiztonsági szakértőkre tartozik.

Az értekezés szempontjából a HIRBizt. stratégia különös jelentőséggel bír, mert *„a szabad, biztonságos és innovatív kibertér megteremtése, Magyarország versenyképességének növelése”* mellett további célokat tűz ki az elektronikus közigazgatással kapcsolatosan: *„az innovációk, az új technológiai megoldások biztonságos módon történő bevezetése, illetve adaptálása a digitalizálódott államigazgatási, kormányzati és gazdasági területeken, a biztonságosabb elektronikus közigazgatási rendszer létrehozása, illetve az állami szolgáltatások innovatív fejlesztése”*. [36, bevezetés]

A meghatározott célok eléréséhez szoftvertechnológiai megoldások is szükségesek, amelyek tárgyalása az értekezés induktív szakaszában következik. A HIRBizt. stratégia külön kiemelendő általános célja a kiberbiztonság, az általános felkészültség és tudatosság szintjének fokozása. Kiemelten, külön kapcsolódási pontként említésre kerül az NKS, mint korábban nyilvánosságra hozott biztonsági stratégia, melynek áttekintése korábban megtörtént. A HIRBizt. stratégia leszögezi, hogy a Kiberbiztonsági Stratégiával együtt közösen kívánja javítani a hazai hálózat- és információ biztonságot.

A HIRBizt. stratégia kiemeli a kibertérben tapasztalható fenyegetések erősödését, amelyek negatív hatással vannak a hálózati és az információs rendszerekre egyaránt. A kiberbűnözés és a kiberhadviselés egyaránt potenciális célként kezeli a kormányzati rendszerek gerincét képező számítógépes hálózatokat, valamint a nemzetállamok

infokommunikációs rendszereiben képződő adatvagyont. A létfontosságú infokommunikációs rendszerek által elszenvedett kibertámadások száma folyamatosan növekszik, valamint a támadások előkészítettsége javul, a kivitelezés egyre kifinomultabbá válik.

A kockázati tényezőket két alapvető csoportba lehet besorolni, az első problémakört a technológiai hiányosságok és az ezeken alapuló sérülékenységi pontok jelentik. Ilyen kockázati tényezők az elavult operációs rendszerek és biztonsági frissítéseket nem tartalmazó felhasználói alkalmazások. Az említett szoftverek biztonsági réseket tartalmazhatnak, és ezáltal sebezhetővé válnak a kártevő szoftverekkel szemben. A számítógépes vírussal fertőzött eszközök, az általuk alkotott bot-hálózatok komoly fenyegetést jelentenek egy összehangolt kibertámadás esetén – itt már komoly felelőssége van a szoftverfejlesztőknek. A 2010-es évektől például a világ egyik legnagyobb szoftvergyártó vállalata a Microsoft már ingyen felajánlja felhasználóinak az operációs rendszerek verziófrissítését az elavult rendszerekről az újabb, támogatott rendszerekre.

Általánosságban elmondható, hogy a felgyorsult iparági folyamatok miatt olyan rendszerek kialakítása válik indokolttá, amelyek műszaki háttere folyamatosan frissíthető. A rendszerekben felhalmozott üzleti tudás mennyisége, a költséghatékonyság és a kiberbiztonsági szempontok egyaránt előre vetítik, hogy fel kell készülnie az informatikának az évtizedeken át fejleszthető rendszerek tervezésére és kialakítására. Véleményem szerint a kritikus rendszerek esetében biztonsági és technológiai frissítéseknek kötelező jellegűeknek kellene lenniük, a gyakoriságot kormányzati szinten kellene szabályozni, a megfelelő anyagi források biztosítása mellett. Azt a következtetést lehet levonni a fentiekből, hogy *a gyors változáskezelés képessége alapvető fontosságú a létfontosságú informatikai rendszerek kialakítása során*.

A másik alapvető kockázati csoport az emberi tevékenységből származó biztonsági kérdések kezelése. A PC-ken és a mobil eszközökön kezdeményezett pénzügyi tranzakciók kiemelt célpontjai a kibertámadásoknak. A digitális banki tranzakciókon túl, a különböző kormányzati ügyintézési folyamatok során, az online vásárlások lebonyolításakor is jelentős adatkezelési tevékenység történik a személyes adatok vonatkozásában. Ha az érintettek azonosítása lehetséges a kezelt személyes adatok alapján, akkor az már alapját képezheti a különböző visszaéléseknek. A felvázolt probléma emberi oldalról a tudatos internetezésre való nevelés segítségével orvosol-

ható. Technológiai oldalról azonban a biztonságos azonosítási és kommunikációs folyamatok kialakításával csökkenthetők a kockázatok.

A HIRBizt. stratégia kiemeli, hogy az állami és önkormányzati intézmények jelentős lemaradásban vannak ezen a területen. A dokumentum rávilágít arra, hogy az *"új biztonsági követelmények teljesítése egy meglévő rendszerben lényegesen nagyobb ráfordítást igényel, mint a követelmények figyelembevételének többletigénye egy új rendszer tervezése során"*. [36, 6. o.] 2016-tól a Közigazgatás-fejlesztési Operatív Program (KÖFOP) pályázati felhívásban már a Nemzeti Elektronikus Információbiztonsági Hatóság (a továbbiakban: NEIH) előírta a biztonsági követelmények nagyobb arányú teljesülését. Követelményként fogható fel a jövőre tekintve, hogy *az új rendszerek fejlesztése során fenntartható, folyamatosan frissíthető műszaki alapok kialakítása szükséges.*

A magánszektorban megvalósuló informatikai fejlesztések esetében a kiberbiztonság harmadlagos szempont – különösen a KKV-k esetében. Az alacsony fejlesztési és üzemeltetési költségek elsődleges prioritást élveznek a funkcionális és a teljesítménnyel kapcsolatos követelmények megvalósulásának rovására is. A pénzforgalmi szolgáltatók és a biztosítók számára a jogszabályi környezet szükségessé teszi a kiberbiztonsági tényezők figyelembe vételét, azonban a fennmaradó ágazatokkal kapcsolatban a HIRBizt. stratégia a következő megállapítást teszi *„a hazai digitális fejlesztési projekteknél az informatikai biztonsági irányítás nem éri el a kibertérben azonosított jelenlegi és várható fenyegetések és az implementált rendszerek sérülékenységei – valamint az ezekből eredő kockázatok – miatt szükséges színvonalat"*. [36, 7. o.] Európai Uniós viszonylatban, a privát szektorban a nagyvállalatok 72%-a rendelkezik információbiztonsági politikával, míg a kisvállalatok esetében ez az arány 32%. A magyar mutatók elmaradnak az Uniós átlagtól, mert a 2016-os adatok alapján ezek a számok 53% és 9%.

A közigazgatás és magánszektor kritikus rendszerei esetében *a biztonsági követelményeknek való megfelelés beépítése az újonnan fejlesztett rendszerekbe egyaránt követelmény kell, hogy legyen. A biztonsági követelmények ellenőrzése, a sérülékenységvizsgálat képezze részét a rendszerfejlesztéseknek.* A feltáró munka során a kutatás szemszögéből négy általános szoftvertechnológiai követelményt sikerült azonosítani a HIRBizt. stratégiában, amelyek megválaszolása az értekezés záró fejezetében található.

S-1 Elengedhetetlen a gyors változáskezelési képesség megléte az informatikai rendszerek teljes életciklusa alatt. Lásd: 43. old.

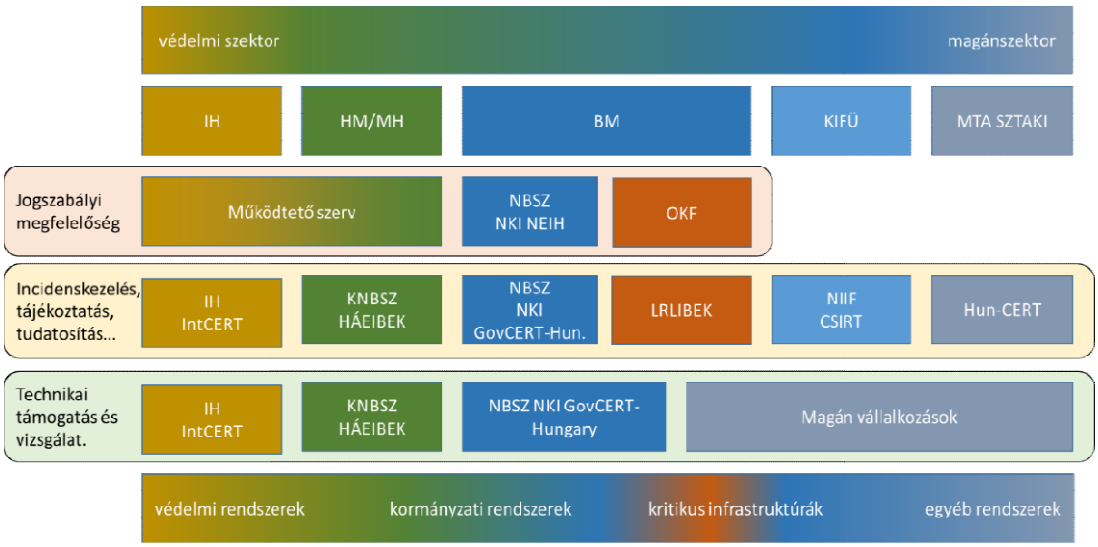
S-2 Az új rendszerek fejlesztése során fenntartható, folyamatosan frissíthető műszaki alapok kialakítása szükséges. Lásd: 44. old.

S-3 Az újonnan kialakított informatikai rendszerek esetén a biztonsági követelmények folyamatos ellenőrzése képezze részét a fejlesztési folyamatnak. Lásd: 44. old

S-4 Kritikus rendszerek esetében a biztonsági követelmények ellenőrzése, a sérülékenységvizsgálat szektor függetlenül képezze részét a rendszerfejlesztéseknek. Lásd: 44. old

1.4.4 Irányítási keretrendszer

Az imént azonosított követelmények szektor függetlenül szem előtt tartandók. A kiberbiztonság kérdésköre nemzetközi viszonylatban is kiemelt hangsúlyt kap. A NATO és az Európai Unió által előírt kötelezettségvállalásoknak való megfeleléshez széleskörű intézményrendszer kialakítása valósult meg kormányzati szinten az elmúlt években. A jelenlegi szervezeti felépítés a 3. ábrán látható.



3. ábra Magyarország kibervédelmi feladatait ellátó szervezetei (saját szerkesztés)

A 3. ábra a 2019. novemberi állapotot tükrözi, amely alapján látható, hogy a Belügyminisztérium (BM) alá rendelt szervezetek végzik a kormányzati rendszerek, illetve a kritikus infrastruktúrákat képező létfontosságú infokommunikációs rendszerek jogszabályi megfelelőségének ellenőrzését. A kritikus rendszerek esetében az Országos Katasztrófavédelmi Igazgatóság (OKF) a felelős, míg a kormányzati rendszerek esetében a Nemzetbiztonsági Szakszolgálathoz (NBSZ) tartozó Nemzeti

Kibervédelmi Intézet (NKI) hatósági feladatokat ellátó intézménye a Nemzeti Elektronikus Információbiztonsági Hatóság (NEIH) végzi a rá háruló feladatokat.

A katonai és a polgári titkosszolgálatok esetében ezeket a feladatokat az adott rendszert működtető szerv látja el. Különböző szektorokon belül külön-külön megtalálhatók a kiberbiztonsági eseménykezelő központok. A polgári hírszerzés esetében az Információs Hivatal keretein belül működő IntCERT látja el az incidenskezelési feladatokat, míg a honvédelmi ágazatban a Katonai Nemzetbiztonsági Szolgálat (KNBSZ) szervezetében működő Honvédelmi Ágazati Elektronikus Információbiztonsági Eseménykezelő Központ (HÁEI BEK) a felelős. A Kormányzati Eseménykezelő Központ (GovCERT-Hungary) az NKI-hoz tartozó információ megosztó és incidenskezelő szervezet, melynek feladata a magyar kormányzat információbiztonságának fokozása és támogatása. A nemzeti létfontosságú rendszerek hálózat- és információbiztonsági feladatait az OKF Létfontosságú Rendszerek és Létesítmények Informatikai Biztonsági Eseménykezelő Központja látja el (LRLIBEK). A Kormányzati Információs Infrastruktúra Fejlesztési Ügynökség (KIFÜ) is rendelkezik számítógépes biztonsági eseménykezelő szervezettel a Nemzeti Információs Infrastruktúra Fejlesztési (NIIF) Program keretén belül, a szervezet röviden a NIIF CSIRT néven ismert. A szervezet célja segíteni az incidenskezelést, minden esetben, ha egy NIIF tagintézménynél történt az.

Minden más esetben, amikor az Infotv. hatálya alá nem tartozó infokommunikációs rendszer esetében következik be biztonsági esemény, akkor a Magyar Tudományos Akadémia (MTA) Számítástechnikai és Automatizálási Kutató Intézet (SZTAKI) által működtetett Hun-CERT az incidenskezelésért felelős szervezet, feladatai közé tartozik a segítségnyújtás az Internet Szolgáltatók Tanácsa (ISZT) tagszervezetei számára.

A hálózati és információs rendszerek sérülékenység-vizsgálatát a védelmi szektoron belül az IH IntCERT, illetve a KNBSZ HÁEIBEK végzi, a kormányzati rendszerek esetében a GovCERT-Hungary a felelős, míg a nemzeti kritikus infrastruktúrák, illetve az egyéb digitális rendszerek esetében a magánvállalatok végzik ezt a tevékenységet.

A bemutatott szereplőkön túl a kormányzat számára a Nemzeti Kiberbiztonsági Koordinációs Tanács (NKKT) nyújt támogatást, véleményező és javaslattevő szervként. Az NKKT valósítja meg a Kiberbiztonsági Stratégiában előírt kormányzati koordinációt, tagjai a kiberbiztonságban érintett szakterületek miniszteri delegáltjai. Az

NKKT munkáját a felkért gazdasági, tudományos és civil szféra képviselőiből álló Kiberbiztonsági Fórum segíti javaslatokkal, véleményekkel. Az NKKT tevékenységének szakmai koordinálását a kiberkoordinátor látja el.

A feltárt irányítási keretrendszerből szoftvertechnológia követelmények ugyan nem vonhatók le, ugyanakkor az látható, hogy Magyarországon már rendelkezésre áll az a szervezeti struktúra, amely a korábban azonosított általános szoftvertechnológiai követelmények megvalósulását ágazati szinten ellenőrizni képes lehet. A továbbiakban a HIRBizt. stratégia szoftverfejlesztéshez kapcsolható célkitűzései kerülnek feldolgozásra.

1.4.5 Célkitűzések

A HIRBizt. stratégia különböző intézkedéseket fogalmaz meg a dokumentumban lefektetett célkitűzések megvalósulásának érdekében. A stratégia három célterületre koncentrálva segítené elő a kiberbiztonság javítását – az érintett területeket az alábbi felsorolás tartalmazza.

1) Digitális környezet iránti bizalom erősítése;
2) digitális infrastruktúra védelem;
3) gazdasági szereplők támogatása.

A továbbiakban a HIRBizt. stratégiában megfogalmazott intézkedések elemzése következik olyan technológiai követelmények után kutatva, amelyek esetében azonosítható valamilyen projektmenedzsmenthez, szoftverfejlesztéshez, informatikai üzemeltetéshez köthető kapcsolódási pont.

Az irányítási keretrendszer bemutatása során láthattuk, hogy jelenleg a kiberbiztonság elősegítését összetett szervezeti rendszer valósítja meg, ahol kormányzati és piaci szereplők egyaránt érintettek. Az első olyan terület, amellyel kapcsolatosan intézkedéseket fogalmaz meg a HIRBizt. stratégia a szakmai együttműködés erősítése. A jelenlegi együttműködési mechanizmusok felülvizsgálata mellett azonosítani kell azokat a területeket, ahol az együttműködés javítandó, olyan fórumokat kell létrehozni, ahol lehetőség nyílik a társadalmi párbeszédre, többek között az etikus hackerek szerepének tisztázására. [36, (1)-(3)] A kölcsönös segítségnyújtáshoz szükséges *„az összehangolt megelőzési, feltárási, mérséklési és reagálási mechanizmusok létrehozása”*. [36, (4)] A védekezési és a reagáló képességek elősegítésének érdekében ismétlődő jelleggel ösztönzendő a privát és állami szereplők bevonásával megtartott gyakorlatok szervezése. A kormányzati és piaci szereplők infor-

matikai rendszereire vonatkozó hibavadászatok feltételeinek tisztázása is sürgető feladat – alapvetően a köz- és magánszféra szerepének és felelősségvállalásának tudatosítása indokolt a kitűzött célok elérésének érdekében [36, (5)-(7)].

A biztonságtudatosság növelésének érdekében jelenleg is tesz Magyarország, résztvevője az ENISA által szervezett évente ismétlődő Európai Kiberbiztonsági Hónap kampányban, a Safer Internet, illetve a Digitális Immunerősítő programokban. Célnak kell lennie a magánszektor és az állami szereplők tudatosságának növelése. A jelentősebb sérülékenységekről, biztonsági eseményekről történő tájékoztatás segítséget nyújthat az érintett szereplők számára. A HIRBizt. stratégia ennek érdekében a hiteles információszerzésre és segítségkérésre alkalmas felületek kialakítását, az általános tájékozottság, tudatosság, felkészültség és fenyegetettség helyzetét reprezentáló adatok gyűjtését és felhasználhatóságát tűzi ki célul. [36, (8)-(9)] A kis- és középvállalatok számára megfelelő ösztönzési módszerek dolgozandók ki, amelyek segítségével az információbiztonsági politikával rendelkező vállalatok aránya növelhető a szektoron belül. [36, (10)-(11)]

A kormányzat által jelenleg működtetett szervezeti struktúra szereplőinek feladatrendszere felülvizsgálatot kíván meg a hatékonyabb együttműködés megvalósítása érdekében. A magánszektor kiberbiztonságának javításához egy olyan szervezet létrehozása a cél, amely képes kibervédelmi megoldásokat ajánlani a magán vállalatok számára. Uniós keretek között a CSIRT és CERT csoportok összehangolt tevékenysége teszi lehetővé a megfelelő hálózatbiztonság megteremtését a köz- és magánszférában. Magyarországnak meg kell tudnia felelni a NIS Direktívában foglaltaknak, a jelenlegi szervezeti rendszer felülvizsgálata mellett létre kell hozni egy nemzeti eseménykezelő központot, amely szélesebb körben nyújt kiberbiztonsági szolgáltatásokat. [36, (13)-(16)] A változásokat ugyancsak indokolja, hogy a kiberbűncselekmények által okozott károk mértéke folyamatosan növekszik, ezért a rendvédelem és az igazságszolgáltatás számára elérendő cél a megfelelő kiberbűnüldözési képességek kialakítása hazai és nemzetközi szinten egyaránt. [36, (11)-(12)] Magyarország tovább kívánja erősíteni az aktív nemzetközi szerepvállalását a NIS Direktívában meghatározott intézményrendszert alkotó szereplők közötti együttműködés erősítésével, többek között nemzetközi a gyakorlatokon történő részvétellel, illetve a kétoldalú és többoldalú nemzetközi kiberbiztonsági kérdések hangsúlyos képviseletével. [36, (28)-(31)]

Úgy gondolom, hogy a hatáskörök és a feladatok tisztázását követően jobb hatásfokkal fognak működni a horizontális és a vertikális együttműködési mechanizmusok. A megfelelő társadalmi párbeszéd kialakítását és a kiberbiztonságot felügyelő szervezeti struktúra átalakítását követően fontos szerepet fognak kapni a NIS Direktívában előirányzott szabványos megoldások, amelyek segítségével az alapvető szolgáltatások biztonsága és megbízhatósága fokozható.

A digitális környezet iránti bizalom és a nemzetközi együttműködés erősítését megcélzó intézkedések elemzése során konkrét szoftvertechnológiai követelményeket nem sikerült azonosítani, azonban az kijelenthető, hogy a különböző tervezési, megvalósítási, ellenőrzési eljárások során szükségesek a megfelelő dokumentációs technikák, műszaki megoldások, amelyekkel a HIRBizt. stratégia is foglalkozik a továbbiakban.

„A kibertámadások elleni eredményes védekezés egyik alapvető feltétele, hogy az informatikai fejlesztésekben már a tervezéskor markáns szerepet kapjon a minőségbiztosítási folyamatok kialakítása, továbbá a kiberbiztonsági kritériumok meghatározása és mérése”. [36, 2.1] Az előző gondolattal nyitja a HIRBizt. stratégia a digitális infrastruktúra védelemmel kapcsolatos fejezetének első bekezdését, amely megbízhatósági követelménynek tekinthető az értekezés szempontjából. Az új informatikai fejlesztések esetében a tervezési fázistól szem előtt kell tartani az alapvető minőségi és kiberbiztonsági követelményeket. A biztonsági követelmények ellenőrzésére ki kell jelölni a kompetens felelőst, elérhető és érthető módszertan segítségével kell támogatni a minőségbiztosítási folyamatot.

A foganatosítandó intézkedések között szerepel *„a könnyen elérhető és használható információs bázis kialakítása”*. [36, (17)] Ezt követően *„modulárisan felépülő módszertani útmutatók kidolgozását”* [36, (18)] tűzi ki célul a HIRBizt. stratégia. Az informatikai fejlesztések minőség-menedzsmentjének javítása érdekében ingyenes belső minőségbiztosítási segédletek, valamint kétnyelvű (angol-magyar), ugyancsak modulárisan felépülő, nyílt kiberbiztonsági minőségmenedzsment tudástár kialakítása szükséges.

Az intézkedésekből levezethető minőségbiztosítási követelményeket a következő lista tartalmazza. Úgy gondolom, hogy szoftvertechnológiai szempontból a modularitás helyett célszerűbb a fázisokra bontható dokumentációs technikák alkalmazása.

Q-1 A minőségbiztosítási folyamatok képezzék részét a kialakítandó szoftverfejlesztési módszertanoknak.

Q-2 A belső minőségbiztosítási eljárások magas szintű támogatása legyen lehetséges a szoftvertechnológiai folyamatok szintjén.

Q-3 A szoftvertechnológiai módszerek rendelkezzenek átlátható dokumentációs technikákkal és eljárásmintákkal.

Q-4 Az IT-fejlesztések különböző fázisaira alkalmazható minőségbiztosítási technikák kerüljenek kialakításra.

Az értekezés szempontjából különös fontossággal bír a HIRBizt. stratégia 2.2 pontja, amely a kormányzati elektronikus szolgáltatások biztonságának növelésével foglalkozik. A megbízható és a biztonságos kormányzati és közigazgatási informatikai háttér kialakítása elsődleges célként jelenik meg a dokumentumban - a meghatározott célterület természetesen tartalmazza a védelmi szektor ágazatait is. A megfogalmazott cél elérésének előfeltétele *„egy olyan kormányzati IT-háttér szisztematikus felépítése, amely mind infrastrukturális, mind üzemeltetési, mind pedig fejlesztési szempontból képes a hagyományos IT-szolgáltatások és a várhatóan egyre több területen elterjedő felhőalapú megoldások, illetve alkalmazás-bérlet (ASP) és szoftverszolgáltatások (SaaS) stabil és megbízható biztosítására”*. [36, 2.2] A NIS Direktíva elemzése során a felhőalapú szolgáltatások szemszögéből azonosított legkritikusabb terület, a szoftver alapú szolgáltatások megbízhatóságának és biztonságosságának fokozása kezelendő stratégiai célként. A foganatosítandó intézkedések elemzése az alábbiakban következik.

„Jöjjön létre és üzembiztosan működjön a stabil és biztonságos kormányzati IT-háttér” [36, (21)] – az intézkedés szoftvertechnológiai követelménnyé formálva a következőként hangzik: *megbízható és stabil rendszerek kerüljenek kialakításra* a kormányzat számára. A sorban a következő intézkedés a nemzetbiztonsági és az elektronikus közigazgatási szolgáltatások szempontjából alapvető szolgáltatást biztosító belső és külső rendszerek maximális védelmét irányozza elő. [36, (22)] Szoftvertechnológiai vonatkozásban ez azt jelenti, hogy *a rendszerek adatkezelési, adatmentési és adattovábbítási képességének maximális védelme alapvető fontosságú*. A NIS direktívával is teljes mértékben összecseng a következő gondolat: *“biztosítani kell a közigazgatás belső rendszereit és külső szolgáltatásait kiszolgáló hálózatok, informatikai infrastruktúra és alkalmazások maximális védelmét”* [36, (23)] – tehát, *a védettség szintjének ellenőrzését támogató szoftverfejlesztési és tesztelési módszerek kiala-*

kítása szükséges az informatikai infrastruktúra nyújtotta szolgáltatások és alkalmazások együttműködésének vizsgálata során. A kormányzati források felhasználásával megvalósított IT fejlesztések teljesülését egységesített biztonsági követelmények teljesítéséhez kellene kötni, ehhez ágazatokon átívelő biztonsági felügyelet létrehozása szükséges. [36, (24)-(25)] Az ellenőrzéshez és a követelmények teljesítéséhez szükséges a biztonsági előírások meghatározása, amennyiben ez adott, akkor a szoftvertechnológia követelményként megfogalmazható, hogy *az alkalmazások biztonsági szintjének ellenőrzését támogató fejlesztési módszerek kialakítása szükséges*.

A fejlesztési környezet ellenőrzését követően a kinyert biztonsági mutatók is ellenőrizhetővé válnak. Az utolsó két intézkedés a meglévő rendszerek védettségének javítását, illetve a jelenlegi szabályozás szigorítását helyezi kilátásba azzal a céllal, hogy egységes biztonsági követelményrendszer álljon elő az informatikai fejlesztések számára.

A következő felsorolás elemei az iménti elemzés során azonosított speciális információbiztonsági követelmények.

I-9 Adatkezelési szempontból megbízható és biztonságos rendszerek kialakítása a cél. Lásd: 50. old.

I-10 Az adatkezelés, az adatmentés és az adattovábbítás maximális védettségének megteremtése alapvető fontosságú. Lásd: 50. old.

I-11 A sérülékenység szintjének ellenőrzését támogató szoftverfejlesztési és tesztelési módszerek kialakítása szükséges az újonnan fejlesztett rendszerek számára. Lásd: 50. old.

I-12 A fejlesztési fázisban lévő alkalmazások biztonsági szintjének ellenőrzését megfelelő szoftverfejlesztési eszközökkel kell támogatni.

A kiberfenyegetések által jelentett kockázatok csökkentését a védelemi, az elhárítási és a reagálási képességek területén végrehajtott kapacitásnöveléssel lehet elérni. A HIRBizt. stratégia célja olyan passzív és aktív eszközök kialakítása, amelyek a felsorolt képességeket erősítik. A passzív kibervédelmi és elhárítási eszközök különböző fenyegetések elleni védelmi funkciót töltenek be, valamint *„biztosítják a fenyegetésekre vonatkozó információk kellő időben történő rendelkezésre állását”*. [36, 2.5] Az informatikai rendszerek fejlesztése során különös fontossággal bír a rosszindulatú infrastruktúrák felfedésére, a *„támadótevékenység nyomon követésére és azonosítására alkalmas”* megoldások kialakítása. Az ellenintézkedések alapját az ilyen formában gyűjtött információk segítségével lehet előkészíteni.

Aktív intézkedés alatt, emberi beavatkozásról beszélünk, *„… informatikai rendszereket érő fenyegetések folyamatos monitorozását, és az azokra való különböző fokozatú reakciókat jelentik”*. [36, 2.5] Olyan észlelési, feldolgozási és felderítési képességek fejlesztésére van szükség, amelyek segítségével detektálhatóvá válnak a fenyegetések és a támadások, illetve a kinyert információk segítségével károkozó tevékenységek forrása azonosíthatóvá a fenyegetettség szintje osztályozhatóvá válik. [36, (39)] Biztonsági követelményként interpretálva, *az informatikai rendszereknek olyan felügyeleti eszközökkel kell rendelkezniük, amelyek lehetővé teszik az észlelési, feldolgozási és felderítési tevékenységek támogatását.* A képességek kialakítását követően az ágazati és ágazatok közötti koordinációra támaszkodva kell kialakítani az átfogó kiberbiztonsági irányítási rendszert. [36, (40)] A felsorolt tevékenységek ágazaton belüli és ágazatok közötti támogatásához megfelelő *módszerekre* és *eszközökre* van szükség, ami követelményként is felfogható az értekezés vonatkozásában. Amint rendelkezésre állnak a megfelelő eljárások és műszaki megoldások, akkor a legkorszerűbb informatika adta lehetőségekkel *„ki kell alakítani a gyors helyzetfelismerés, az értékelés és a kockázatelemzés rendszerét”*. [36, (41)] Az elemzés során további négy általános kiberbiztonsági követelményt sikerült azonosítani az alábbiak szerint.

K-5 Az informatikai rendszereknek olyan felügyeleti eszközökkel kell rendelkezniük, amelyek lehetővé teszik az észlelési, feldolgozási, és felderítési tevékenységek támogatását.

K-6 Az ágazaton belül és ágazatok között is megfelelő módszerekkel kell támogatni a kockázatok kezelését.

K-7 A kiberfenyegetésekre történő megfelelő reagálási eljárások kialakításához és támogatásához a kockázatelemzés és a kockázatkezelés számára megfelelő eszközrendszer biztosítása szükséges.

K-8 Gyors helyzetfelismerést, értékelést és kockázatelemzést támogató módszerek kialakítása szükséges.

A HIRBizt. stratégia további technológiai, módszertani követelményt nem tartalmaz, ugyanakkor intézkedésekkel kívánja előirányozni mérnökök, kutatók képzését, a kutatási stratégiák kialakítását. Célként szerepel a kiberbiztonságot elősegítő eszközök és termékek fejlesztése és alkalmazása. [36, (44)-(45)] Ezen túl megvalósítandó a kutatás-fejlesztési területek azonosítása, támogatása hazai viszonylatban és a nemzetközi jelenlét elősegítése. [36, (46)-(48)].

A KKV szektor információbiztonságának növelése pályázati rendszer segítségével valósulhat meg, illetve államilag támogatott kibervédelmi szolgáltatáscsomagok kialakításával az oktatás, a képzés, a biztonságos üzemeltetés, a megfelelőség és az audit témakörökben. [36, (49)-(50)]. Az utolsó fejezet a hazai versenyképes tudásbázis létrehozását tűzi ki célul, az oktatási rendszer bevonásával, információbiztonsági képzések szervezésével, valamint az alapvető szolgáltatók személyi állományának továbbképzése révén.

Az EU és a kormány által létrehozott szabályozás tárgyalása a HIRB. stratégia feltárásával véget ér. Az elemzés során sikerült azonosítani nyolc kiberbiztonsági, tizenkettő információbiztonsági, négy szoftvertechnológiai és négy minőségbiztosítási követelményt. A továbbiakban a katonai célú szoftverfejlesztés specialitásai után kutatva előbb a Magyar Honvédség, majd a NATO kapcsolódó dokumentumainak feltárása következik.

1.5 ÖSSZEGZÉS

Az 1. fejezet feltáró munkája során áttekintésre került az Európai Uniós és a magyar kritikus infrastruktúra védelemhez köthető szabályozási keretrendszer, amelynek eredményeként az általános azonosítási, kijelölési, felülvizsgálati és nyilvántartásba vételi folyamatok tárgyalása is megtörtént. Végezetül, a Magyar Honvédségre vonatkozó ágazatilag kezelt kritikus infrastruktúra védelem kérdésköre is említésre került – kapcsolódási pontra lelve a létfontosságú rendszerek és a védelmi szektor működése között.

Ezt követően a létfontosságú infokommunikációs rendszerekre vonatkozó speciális szabályozás tárgyalása valósult meg a hálózat- és információbiztonság szemszögéből, amelynek eredményeként nyolc általános információbiztonsági kutatási követelmény azonosítása történt meg a felhő alapú szolgáltatás típusok átfogó tárgyalása mellett.

A fejezet záró szakaszában Magyarország különböző biztonsági stratégiáinak kiberbiztonsághoz köthető aspektusait vizsgálva további speciális és általános kutatási követelmények azonosítása valósult meg. Szám szerint, nyolc általános kiberbiztonsági, négy általános szoftvertechnológiai, négy speciális információbiztonsági és négy minőségbiztosítási kutatási elvárás formájában. A fejezetben azonosított kutatási követelmények felsorolása a következő oldalakon szerepel.

Kiberbiztonság

K-1 A különböző biztonsági komplexumhoz köthető információs rendszerek kialakítása során egyaránt törekedni kell a biztonságra, különös tekintettel a katonai biztonságra.

K-2 Olyan infokommunikációs technológiák és módszerek kialakítása szükséges, amelyek biztosítják a megfelelő védekezési és reagálási képességeket.

K-3 A számítógépes hálózatokat és a kapcsolódó információs rendszereket fel kell készíteni az összehangolt támadásokkal szemben.

K-4 Szoros együttműködést, hatékony információcserét biztosító infokommunikációs rendszerek kialakítása szükséges a védelmi ágazat szereplői számára.

K-5 Az informatikai rendszereknek olyan felügyeleti eszközökkel rendelkezniük, amelyek lehetővé teszik az észlelési, feldolgozási, és felderítési tevékenységek támogatását.

K-6 Az ágazaton belül és ágazatok között is megfelelő módszerekkel kell támogatni a kockázatok kezelését.

K-7 A kiberfenyegetésekre történő megfelelő reagálási eljárások kialakításához és támogatásához a kockázatelemzés és a kockázatkezelés számára megfelelő eszközrendszer biztosítása szükséges.

K-8 Gyors helyzetfelismerést, értékelést és kockázatelemzést támogató módszerek kialakítása szükséges.

Információbiztonság

I-1 A megbízhatóságra tervezési és fejlesztési oldalról is törekedni kell.

I-2 Felhő alapú üzemeltetési környezetben is megfelelő megbízhatósági és biztonsági paraméterekkel rendelkező szolgáltatásokat kell kialakítani.

I-3 Magas szintű biztonságot és megbízhatóságot garantáló szabványok kialakítása és alkalmazása a cél.

I-4 A biztonsági események detektálása, a logikai hibák feltárása, javítása szabványos eljárásokkal és módszerekkel történjen.

I-5 Szabályozott módszerek és folyamatok segítségével történjen a biztonsági tervek előállítása.

I-6 Megfelelő tájékoztatás mellett kezelhetők, detektálhatók, bejelenthetők legyenek a biztonsági események.

I-7 Biztonsági szempontok kezelésére alkalmas, folyamatos működést garantáló tervezési módszerekre van szükség.

I-8 A megfelelő reagálási képesség elősegítéséhez olyan rendszerfelügyeleti megoldások kialakítása szükséges, amelyek azonnal jelzik az alapvető szolgáltatásokban bekövetkező változásokat, figyelmeztetnek a potenciális biztonsági eseményekre.

I-9 Adatkezelési szempontból megbízható és biztonságos rendszerek kialakítása a cél.

I-10 Az adatkezelés, az adatmentés és az adattovábbítás maximális védettségének megteremtése alapvető fontosságú.

I-11 A sérülékenység szintjének ellenőrzését támogató szoftverfejlesztési és tesztelési módszerek kialakítása szükséges az újonnan fejlesztett rendszerek számára.

I-12 A fejlesztési fázisban lévő alkalmazások biztonsági szintjének ellenőrzését megfelelő szoftverfejlesztési eszközökkel kell támogatni.

Szoftvertechnológia

S-1 Elengedhetetlen a gyors változáskezelési képesség megléte az informatikai rendszerek teljes életciklusa alatt.

S-2 Az új rendszerek fejlesztése során fenntartható, folyamatosan frissíthető műszaki alapok kialakítása szükséges

S-3 Az újonnan kialakított informatikai rendszerek esetén a biztonsági követelmények folyamatos ellenőrzése képezze részét a fejlesztési folyamatnak.

S-4 Kritikus rendszerek esetében a biztonsági követelmények ellenőrzése, a sérülékenységvizsgálat szektor függetlenül képezze részét a rendszerfejlesztéseknek.

Minőségbiztosítás

Q-1 A minőségbiztosítási folyamatok képezzék részét a kialakítandó szoftverfejlesztési módszertanoknak.

Q-2 A belső minőségbiztosítási eljárások magas szintű támogatása legyen lehetséges a szoftvertechnológiai folyamatok szintjén.

Q-3 A szoftvertechnológiai módszerek rendelkezzenek átlátható dokumentációs technikákkal és eljárásmintákkal.

Q-4 Az IT-fejlesztések különböző fázisaira alkalmazható minőségbiztosítási technikák kerüljenek kialakításra.

2. A MAGYAR HONVÉDSÉG INFORMATIKAI CÉLKITŰZÉSEI ÉS A SZAKTERÜLETET SZABÁLYOZÓ DOKUMENTUMAI

A hatályos MH Informatikai Stratégiából [37] valamint az érvényes MH Informatikai Szabályzatból [38] és a gyakorlati tapasztalatokból kiindulva is azt láthatjuk, hogy egy igény egy új szolgáltatás kialakítására vagy egy létező rendszer továbbfejlesztésére különböző felhasználói szintről érkezhet. Az igény formálója tartozhat a legmagasabb alkalmazói szinthez vagy a legalacsonyabb felhasználói szinthez is. Minden szereplő esetére a fent hivatkozott két dokumentum alapján levezethető a szolgáltatás fejlesztés indításának, illetve a fejlesztés szükségességére vonatkozó döntéshozatalnak a folyamata. *„Nem kerül azonban egyértelműen tisztázásra, hogy a híradó-informatikai szolgáltatások közé csak a híradó-informatikai*[26] *rendszerek, alkalmazások, eszközök által nyújtott, a szervezeti folyamatokat, tevékenységeket támogató szolgáltatások tartoznak, vagy olyan fontos híradó-informatikai tevékenységek is, mint a fejlesztés, ellátás, technikai kiszolgálás, felkészítés, beszerzés, tanácsadás, stb.*" [39, 150. o.] – derül ki Munk Sándor munkájából.

Ebben a fejezetben a szoftverfejlesztés szemszögéből vizsgálom a jelenlegi szabályozást, különös tekintettel arra, hogy ki és hogyan formálhatja az új és már bevezetett egyedi szoftverekkel szemben támasztott igényeket. Stratégiai szinten legfeljebb a döntés születhet meg egy infokommunikációs rendszer szükségességével kapcsolatban. A magas szintű vezetés azonosíthatja a kapcsolódó szakterületeket, de véglegesnek tekinthető igényeket nem tud formálni. Az egyes szakterületek feladata, hogy meghatározzák saját követelményrendszerüket, azonban a munkafolyamat felügyelete minden esetben a legmagasabb szintű infokommunikációért felelős szervezeté. Elvárható-e, hogy egy új szoftver kifejlesztéséhez első nekifutásra elegendő mértékű és megfelelő minőségű dokumentáció álljon elő, amikor napjaink szoftverei által nyújtott lehetőségek tárháza végtelen?

A katonai célú rendszerfejlesztés nagy múltra tekint vissza, az Egyesült Államokban 1956-ban fogadták el az első konkrét programot a szárazföldi csapatok háborús vezetésének automatizálására vonatkozóan, amelyet egy 10 éves kutatási tevékenység

[26] A honvédségi terminológiában korábban használt híradó-informatika szókapcsolat helyett az infokommunikáció szakkifejezést alkalmazom a továbbiakban, kivételt képeznek ez alól az idézett tartalmak.

követett, amely során kialakult a megfelelő műszaki bázis és fejlesztési gyakorlat is. [40, 140. o.] Az első rendszerek tapasztalatai alapján egy sor újabb rendszerfejlesztés indulhatott el, melyek eredményeit napjainkban is alkalmazzák. Az említett kutatásokhoz szükség erőforrásokat csak a vezető nagyhatalmak tudták megteremteni abban az időben. Napjainkra annyiban változott a helyzet, hogy a korszerű szoftverfejlesztési módszertanok és technológiák alkalmazásával elérhető cél lehet a magas színvonalú egyedi alkalmazásfejlesztés a Magyar Honvédség számára is.

Itt feltétlenül meg kell említeni, hogy az egyedi alkalmazásfejlesztés nem minden esetben a legcélravezetőbb megoldás, mert egy saját fejlesztésű szoftver elkészítése, majd annak fenntartása nagy költségekkel jár. A védelmi kiadások csökkentése érdekében az USA-ban a 90-es évektől megjelent a COTS[27] rendszerek alkalmazása bizonyos területeken. Egy kereskedelmi forgalomban kapható dobozos termék alkalmazása ugyan elsőre alacsonyabb költségeket ígér, azonban ebben az esetben a vásárlás előtt alapos tesztelésre van szükség, hogy a kiválasztott megoldás valóban megoldja-e az alkalmazó eredeti problémáját. A katonai célú COTS rendszerek teszteléséről és bevezethetőségéről Négyesi Imre munkáiban [41][42], találhatunk részletes leírást különös tekintettel a COTS rendszerek értékelési stratégiájáról, kiválasztási módszereiről [42; 113-114 old.], valamint egy módszertant is kapunk egy adott eszköz kiválasztásához. [42; 114 old.] A kutatás további szakaszában feltételezem, hogy olyan speciális infokommunikációs igények jelennek meg alkalmazói oldalon, amelyekhez nem léteznek katonai célú COTS rendszerek. Így az alkalmazói igények kielégítése egyedi szoftverek fejlesztésével valósítható meg, a továbbiakban a Magyar Honvédség szabályozó dokumentumainak ebből a szemszögből történő vizsgálata következik.

2.1 AZ MH INFORMATIKAI STRATÉGIÁJÁNAK ÉS SZABÁLYZATÁNAK SZOFTVERTECHNOLÓGIAI ELEMZÉSE

Az infokommunikációs szolgáltatások és rendszerek fejlesztésével az MH Informatikai Stratégiájának 2. pontjában találkozhatunk először, a dokumentum itt deklarálja, hogy ezeken a területeken is az Informatikai Stratégiában szereplő alapelvek a mérvadók.

[27] COTS - **C**ommercial **o**ff-**t**he-**s**helf angol kifejezés szavainak kezdőbetűiből, jelentése: kereskedelmi úton beszerezhető

A szoftverfejlesztés kapcsán a következő követelmény különös fontossággal bír: *„a szolgáltatások a változó felhasználói igényeknek és alkalmazási körülményeknek megfelelően, minimális erőforrás és idő ráfordításával átalakíthatók, adaptálhatók”*. [37, 4. g)] – Az olvasóban felmerülhet a kérdés, vajon milyen szoftverfejlesztési módszertan képes ezeket az igényeket kielégíteni? Egyáltalán lehetséges-e olcsón, gyorsan, jó minőségű szoftvert előállítani, majd a későbbiekben fenntartani azt? A kérdés megválaszolása a 3. fejezetben található. Lásd: 135. old.

Egy szolgáltatásokra vonatkozó általános alapelv jelenik meg az Informatikai Stratégiában, az alábbiak szerint: *„Az infokommunikációs technológia alkalmazása és fejlesztése a jelen és az elkövetkezendő tíz év meghatározó innovációs és hatékonyságnövelő tényezője, amelynek a honvédelmi ágazatban érvényre kell jutnia”*. [37, 7] Lehetséges infokommunikációs fejlesztés lehet a számítógépes eszközpark fejlesztése, de ezzel összhangban hatékonyságnövelő lehet a szoftverfejlesztésre vonatkozó módszerek fejlesztése is. Új szoftvertechnológiai módszerek alkalmazásával hatékonyabbá tehető a katonai célú egyedi alkalmazásfejlesztés menete is. Általában elmondható, hogy az infokommunikációs fejlesztések modern előkészítése új távlatokat nyithat a fejlesztések megvalósításában és a kialakított szolgáltatások fenntartásában egyaránt.

Irányítás területén az Informatikai Stratégiában még a szoftverfejlesztést illetően a következő elv jelenik meg: *„A híradó informatikai szolgáltatások fejlesztése és biztosítása az alkalmazó szervezetek műveleti követelményei, a híradó-informatikai szervezetek szakmai követelményei, a beszerzés műszaki követelményei, a kialakításra kerülő rendszer rendszerterve, az alkalmazó szervezet alkalmazási terve, valamint a tevékenységet szabályozó jogszabályok, okmányok intézkedések alapján valósuljon meg.”* [37, 72. b)]. Úgy gondolom, hogy szoftvertechnológiai szempontból a műveleti követelmények és az infokommunikációért felelős szervezetek szakmai követelményei egy megfelelő dokumentációs eljárással már tekinthetők egy infokommunikációs fejlesztés követelményrendszerének. A megfelelő dokumentációs módszerekkel nem csak egy agilis implementációt, hanem egy hagyományos mederben folyó megvalósítást is korszerűen elő lehet készíteni.

Az Informatikai Stratégián belül újabb konkrét követelmények szoftverfejlesztéssel kapcsolatban már nem jelennek meg, ugyanakkor a további kérdésekben az MH Informatikai Szabályzata tekintendő mérvadónak. [37, 72. c)]

2.1.1 Infokommunikációs fejlesztések

A már rendszeresített és használatba vett szoftverekkel kapcsolatos szabályozást az Informatikai Szabályzat 3. fejezete tárgyalja, míg az új rendszerek fejlesztésére a 4. fejezet vonatkozik. Az alábbiakban előbb a fejlesztésre vonatkozó követelményeket gyűjtöttem össze, majd a bevezetett rendszerekkel szemben támasztott követelményeket vizsgáltam meg.

Az Informatikai Szabályzat megkülönböztet külső és belső rendszerfejlesztést, ugyanakkor leszögezi azt, hogy mindkét esetben alkalmazni kell a szabályzatban foglaltakat. [38, 4.2.7.] A fejlesztést a szabályzat 3 fázisra bontja az alábbiak szerint:

a) *„a fejlesztés igénylése és előkészítése*
b) *a fejlesztés megvalósítása*
c) *a szolgáltatás bevezetése”* [38, 4.3.1.]

Az első fázis feladata, hogy a fejlesztési igény megfogalmazását követően egy döntés előkészítési folyamaton keresztül az érintett szervezeteket meghatározva döntés szülessen az adott infokommunikációs fejlesztés szükségességéről. Az érintett szervezeteken túlmenően kijelölésre kerül a rendszergazda és a fejlesztésért felelős vezető is. Ha hozzávesszük az adott fejlesztésnek helyt adó hálózat gazdáját, a fejlesztésben érintett követelménytámasztók köre az 1. fázis végére adottnak tekinthető.

A második fázisban, a megvalósítás során találkozhatunk a követelmények meghatározásával, a tervezéssel és a végrehajtással is. A fejlesztés tervezésének szabályozása alapvetően az MH oldali folyamatokra vonatkozik, a megvalósítást végzők feladatait nem részletezi. Ebben a szakaszban kerülnek kijelölésre a fejlesztésben együttműködő, illetve az azt támogató szervezeti elemek. A második fázis szoftvertechnológiai szempontból érdekes része a fejlesztési követelmények meghatározására vonatkozó pont [38, 4.3.3.3.], amely szerint az infokommunikációs fejlesztések végrehajtásához az alábbi igényeket és követelményeket kell meghatározni:

a) *„az alkalmazói igényeket;*
b) *a hadműveleti követelményeket;*
c) *a műszaki követelményeket;*
d) *a működtetési, fenntartási követelményeket.”* [38, 4.3.3.3.1.]

A végrehajtás során az alkalmazói igények, a hadműveleti követelmények, valamint a műszaki követelmények alapján áll össze a fejlesztést meghatározó követelményrendszer, míg a működtetési és fenntartási követelmények a bevezetés és rendszerbe állítást követően kerülnek előtérbe.

Ha szigorúan a szoftverfejlesztés szemszögéből vizsgáljuk az aktuális szabályozást, akkor a szükséges követelmények, igények szabályozás szintjén megjelennek, ezek további strukturálása, rendszerbe szervezése lehetséges, amit a szolgáltatásokra vonatkozó szabályok feldolgozását követően válik lehetségessé.

2.1.2 Informatikai szolgáltatások

A szolgáltatásokra vonatkozó szabályozás széleskörűen és alaposan szabályozza a Magyar Honvédségnél jelenlévő informatikai szolgáltatások menedzsmentjét. Az alábbiakban a bevezetett szolgáltatásokkal kapcsolatos szoftvertechnológiai szempontból érdekes részelteket emeltem ki.

„A szolgáltatásgazda a híradó-informatikai szolgáltatások meghatározott körének előírt feltételek közötti biztosításáért, tervezéséért, fejlesztéséért, felügyeletéért, a felhasználókkal való kapcsolattartás szervezésért a szolgáltatás teljes életciklusa alatt felelős személy." [38, 3.4.2.]

A fejlesztési és az üzemeltetési szempontok feltárását követően megfogalmazható a következő fejlesztési módszertanra vonatkozó követelmény: *az infokommunikációs szolgáltatások biztosítását, tervezését, fejlesztését, felügyeletét támogató, a szolgáltatás teljes életciklusára értelmezhető szoftverfejlesztési módszertan kialakítása a cél.*

További elvárás, hogy az üzemeltető szervezet meghatározott rendszerességgel felülvizsgálja az alkalmazói és támogató szolgáltatásokat az alábbi szempontok szerint:

a) *„a szolgáltatás aktualitása;*
b) *a felhasználók elégedettsége, a felhasználói visszajelzések, javaslatok;*
c) *a rendelkezésre állás;*
d) *a szolgáltatási szint megállapodásban rögzített teljesítménymutatók;*
e) *az üzemeltethetőség;*
f) *a gazdaságosság;*
g) *a technológiai fejlődés lehetősége és megtérülése alapján."* [38, 3.11.4.1.]

Amennyiben a felülvizsgálat során megjelenik valamilyen fejlesztésre vonatkozó igény, akkor *„az üzemeltető szervezet a fejlesztési javaslatot felterjeszti a szolgáltatásgazda részére, aki szükség esetén kezdeményezi a fejlesztést"* [38, 3.11.4.2.] A szolgáltatásokra vonatkozó szabályozásról kijelenthető, hogy megköveteli a fennálló szolgáltatások, rendszerek felülvizsgálatát, jobbá tételét. Ezek a követelmények az infokommunikációs rendszerek továbbfejlesztésére vonatkoznak, amelyek természetesen magukba foglalják a különböző szoftverfejlesztési tevékenységeket is. A vál-

zások kezelésével kapcsolatosan újabb módszertani követelmény fogalmazható meg, miszerint *a fejlesztési javaslatok elemzését, majd szabályozott módon történő változtatási igények meghatározását támogató módszertan kialakítása szükséges.* A következő felsorolás az elemzés során feltárt módszertani követelményeket tartalmazza.

M-1 A szolgáltatások minimális erőforrás és idő ráfordításával legyenek átalakíthatók, adaptálhatók a változó felhasználói igényeknek és alkalmazási körülményeknek megfelelően. Lásd: 58. old.

M-2 Az infokommunikációs szolgáltatások fejlesztése és biztosítása az alkalmazó szervezetek műveleti követelményei, az infokommunikációért felelős szervezetek szakmai követelményei, a beszerzés műszaki követelményei, a kialakításra kerülő rendszer rendszerterve, az alkalmazó szervezet alkalmazási terve, valamint a tevékenységet szabályozó jogszabályok, okmányok intézkedések alapján valósuljon meg. Lásd: 58. old.

M-3 Az infokommunikációs szolgáltatások biztosítását, tervezését, fejlesztését, felügyeletét támogató, a szolgáltatás teljes életciklusára értelmezhető szoftverfejlesztési módszertan kialakítása a cél. Lásd: 60. old.

M-4 A fejlesztési javaslatok elemzését, majd szabályozott módon történő változtatási igények meghatározását támogató módszertan kialakítása szükséges.

2.2 AZ MH HID SZOFTVERTECHNOLÓGIAI ELEMZÉSE

Az előző bekezdésekben láthattuk, hogy a Magyar Honvédség középtávú informatikai törekvéseit a 2014-es Informatikai Stratégia határozza meg, ezen túlmenően az infokommunikációs szolgáltatásokra vonatkozó folyamatokat, eljárásokat a hatályos Informatikai Szabályzat írja le. A szabályzat 3-5 fejezetei foglalkoznak az infokommunikációs szolgáltatásokkal, azok tervezésével, fejlesztésével és a már rendszeresített, illetve használatba vett szolgáltatások üzemeltetésével. Az említett dokumentumok általános követelményeket határoznak meg a stacioner és a tábori rendszerek vonatkozásában, azonban létezik egy gyakorlati követelményekre nagyobb hangsúlyt fektető hatályos dokumentum is, a Magyar Honvédség Összhaderőnemi Híradó és Informatikai Doktrínája [43], a továbbiakban MH HID.

Az MH HID előszavából megtudhatjuk, hogy a dokumentum célja az, hogy *„összefoglalja a katonai híradás és informatika alapelveit, meghatározza a honvédelmi célú híradó és informatikai rendszerek tervezésével, szervezésével és alkalmazásával kap-*

csolatos elveket és követelményeket." [43, 7. o.] A dokumentum az alapelveken túl bemutatja a jelenlegi NATO szabványokat és azok alkalmazásának rendjét is. Az MH HID-ben általános, alkalmazási kérdéseket tárgyaló informatikai követelményrendszer található, amely nem tér ki külön a szoftverfejlesztésre, mint speciális informatikai tevékenységre, ugyanakkor számos párhuzam vonható, a követelmények interpretálhatók.

Az MH HID 1. fejezetében az infokommunikációs rendszerek alapjai kerülnek bemutatásra, ezen belül az informatikai rendszerekkel szemben támasztott tizenhat általános alapelvvel találkozhatunk. [43, 11-14 o.] Az alábbiakban a szoftverfejlesztéshez szervesen kapcsolható tizenkét elv kerül bemutatásra. Az eredeti dokumentumban az egyes alapelvekhez tartozó értelmezések általánosak, ez annak tudható be, hogy híradásra és informatikára együttesen vonatkozó iránymutatásról van szó. A felsorolt pontokban az alapelvek szoftvertechnológiai elemzése következik, a fejlesztett szoftver és a fejlesztési folyamat vonatkozásában is.

S-5 Megbízhatóság – általános informatikai követelmény, szoftverek esetében jó minőségű, hibamentes, folyamatos működésként írható le az adott informatikai rendszer rendeltetésszerű használata mellett. Sajnos, *"a szoftverek megbízhatóságát nem lehet közvetlenül mérni, ezért a szoftverek megbízhatóságának becslését és a különböző termékek összehasonlítását csak szoftvereken értelmezett tényezők (tulajdonságok) vizsgálatával lehet elvégezni."* [44] Azonban ez azt is jelenti egyben, hogy lehetséges olyan releváns mérési tényezők bevezetése, amelyek megbízhatósági szempontból megmutatják a rendszer pillanatnyi állapotát.

S-6 Szabványosság – általános műszaki követelmény, amely értelmezhető COTS rendszerekre, egyedi alkalmazásokra, szoftver modulokra, szoftver komponensekre és szoftverfejlesztési módszerekre is. Azonban a szabványosság túlzott kikényszerítése problémákhoz is vezethet egy szoftverfejlesztési projekt esetében, a legcélravezetőbb megoldás a szabványoknak való megfelelés menedzselése. [45] Ezt a szabványok értékelésével és megfelelő hangsúllyal való kezelésével lehet kivitelezni egy szoftverfejlesztési projekt során.

S-7 Kompatibilitás – azonos technológiák, ezen belül azonos fordítási folyamatok (compilation process), összeépítési módszerek (build methods), tesztelési eszközök, futtatókörnyezetek, azonos szoftververziók felhasználását jelenti. Megfelelő kompatibilitási szintről akkor beszélhetünk, ha a szoftver előállítá-

sa és alkalmazása során a felhasznált komponensek egymással történő interakciói az elvártaknak megfelelően, helyesen mennek végbe. [46, 7.2.8] A kompatibilitási szint vizsgálata – a folyamatok megbízhatóságának javítása – automatizált ellenőrző eljárások segítségével valósítható meg.

S-8 Interoperabilitás – más szoftverekkel történő kétirányú együttműködés kialakításának képessége. Létező modulok, illetve komponensek beágyazhatósága. A megfelelő interoperabilitási szint elérését, jól definiált interfészek, nyílt forráskód, illetve nyílt szabványok segíthetik elő. [47]

S-9 Rugalmasság vagy manőverező képesség – megfelelő jogosultságok mellett ugyanazon képességek, funkciók különböző felhasználókkal történő alkalmazása, illetve ugyanazon funkciók különböző felületekről történő igénybe vétele a rendszer konzisztenciájának megőrzése mellett. A szoftverfejlesztési folyamatok tekintetében is értelmezhető a rugalmasság fogalma, amely a rendszer módosíthatóságának szintjét takarja. Léteznek olyan formális módszerek, amelyek segítségével számszerűsíthető egy adott rendszer módosíthatósági mutatója rugalmassági pontok segítségével. [48]

S-10 Hitelesség – általános informatikai követelmény, mely a rendszerben tárolt adatok megbízhatóságára vonatkozik.

a) *Alap szinten* – az informatikai rendszerben vezetett eseménynapló, mely segítségével kinyerhető a rendszerben tárolt adatok változásának története.
b) *Felső szinten* – valamilyen hitelesítési eszköz alkalmazása, elektronikus aláírás, illetve elektronikus bélyegző integrált alkalmazása az adatok hitelességének garantálása érdekében.

A hiteles környezetben végrehajtott szoftverfejlesztési folyamatok érdekében elektronikusan hitelesített eszközök, hitelesített hardverek és a konfiguráció menedzsment részeként kezelt hozzáférési kulcsok alkalmazása szükséges. [49, 6.4] Olyan technológiák alkalmazása indokolt, amelyek magukban foglalják ezeket a lehetőségeket a katonai célú infokommunikációs rendszerek fejlesztése során.

S-11 Modularitás – általános szoftvertechnológiai követelmény, amely önálló továbbfejlesztési lehetőséget jelent az egyes rendszerelemek, modulok, komponensek esetében. Valamint az egyes modulok cseréjének, verzióváltoztatásának lehetőségét is magában foglalja. A megfelelően kialakított moduláris fel-

építés a fejlesztett rendszerek hosszú távú fejlesztési lehetőségét vetíti előre műszaki és pénzügyi oldalról is. A jól modularizált rendszerek az alkalmazókat piaci – jelen esetben katonai – erőfölényhez juttathatják a bonyolultság és a költségek csökkentésével. [50]

S-12 Skálázhatóság – ugyanazon szoftver vagy szoftverkomponens különböző hardveres környezetben történő alkalmazásának lehetősége.

a) *Alap szinten* – minimális hardverkövetelmények melletti működés, illetve korlátozott funkcionalitású működés képessége.
b) *Felső szinten* – szoftver kapacitásának növelhetősége a hardveres erőforrások növelésével. Például: Egy időben kiszolgálható felhasználók száma.

Az MH HID-ből megismert szinteken túl a szoftver előállítási folyamatokra is értelmezhető a skálázhatóság, amelynek eléréséhez: moduláris architektúra és technológiai platform, automatizált eszközök, hatékony követelményelemzés, kis méretű fejlesztő csapatok, rugalmas szervezeti modell, termékfejlesztési folyamatok és paraméterezhetőségi képesség kialakítása szükséges. [51, 4]

S-13 Biztonság és védettség informatikai rendszerek esetében – az informatikai rendszeren belüli megfelelő azonosítási és jogosultsági rendszer kialakítása. Szoftverekre vonatkoztatva az egyes funkciók elérése csak valamilyen azonosítási eljárást követően megfelelő jogosultságokkal lehetséges. A megfelelő védettségi szint vizsgálatához léteznek különböző sérülékenységvizsgálatot támogató módszerek, ilyen például a Krasznay Csaba által kidolgozott Védelmi Profil eszköztár [52, 1.6] is. Az előállítás/fejlesztés folyamatát támogatandó léteznek olyan statikus forráskód ellenőrzést [53], teszt lefedettség vizsgálatot végző keretrendszerek [54], amelyekkel a biztonsági kockázatok csökkenthetők, bizonyos szektorokban ezek alkalmazása kötelező is. [55]

S-14 Felhasználhatóság – szoftvertechnológiai követelményként értelmezve az adott rendszer grafikus felületeire, más rendszerek számára biztosított interfészeire, illetve a szoftveren belül megvalósított folyamatokra vonatkozó követelmény. A felhasználhatósággal több minőségbiztosítási szabvány is foglalkozik, ilyenek például az ISO 9241-210:2019 [56], valamint az ISO/IEC 25010:2011 [57]. Előbbi az interaktív rendszerek esetében az emberközpontú tervezést támogatja, míg utóbbi a teljes rendszertervezési és szoftverfejlesztési minőségbiztosítási folyamatot írja le. A szabványok mellett léteznek olyan

módszerek is, amelyek a felhasználhatóság számszerűsített mérését támogatják. [58, 3]

S-15 Információ-megosztás – az egyes funkciók, elérhető információk szerepkörökhöz, jogosultságokhoz történő kapcsolása. Speciális jogosultságként kezelhető egy adott felhasználó helye a katonai hierarchiában. Például: a magasabb szinten elhelyezkedő felhasználók több funkciót érhetnek el vagy a megismerhető információ mennyisége növekedhet a magasabb vezetési szinteken. Az információ-megosztáshoz kapcsolódó kommunikációs tevékenységek vizsgálata és összevetése a tudásmegosztással, mint kommunikációs tevékenységgel[28], hasznos és jövőbemutató lehet a katonai célú infokommunikációs rendszerek tervezésekor.

S-16 Adat-konzisztencia – szoftvertechnológiai szemszögből azt jelenti, hogy egy adatot lehetőség szerint egy informatikai rendszeren belül csak egyszer tároljunk. Ezzel a technikával megfeleltethetők a valós életben létező fogalmak és fogalmi-kapcsolatok a szoftverben megjelenő fogalmakkal és a közöttük lévő logikai kapcsolatokkal. Lényeges szempont, hogy az egyes rendszerek megfelelő mutatókkal rendelkezzenek, hogy a jövőben a Big Data[29] technológia segítségével könnyen hasznosítható adatvagyon felhalmozása történjen meg. Már jelenleg is léteznek az adat-konzisztencia szintjének formális értékelésére vonatkozó módszerek, amelyek a jövőben felhasználhatóvá válhatnak akár az infokommunikációs rendszerek tervezése és fejlesztése során is. [62]

A fenti értelmezés az MH HID egy lehetséges kiterjesztése a katonai célú szoftverfejlesztésre vonatkozóan. Az imént felsorolt pontok és a hozzájuk tartozó értelmezések egyaránt felhasználhatók stacioner és tábori célú egyedi alkalmazások fejlesztésekor. Kiemelt alkalmazási terület lehet az infokommunikációs rendszerekkel szemben támasztott követelmények összeállítása, valamint az elkészült szoftverek műszaki és minőségi felülvizsgálata. A felsorolásban szereplő elemek a kutatás szemszögéből a speciális szoftvertechnológiai követelmények kategóriájába tartoznak.

[28] A téma alapos tárgyalása Reijo Savolainen kapcsolódó publikációjában [59] található.

[29] Big Data – *„A Big Data olyan nagy mennyiségű, gyorsan és nagy sebességgel növekvő információs vagyon, amely költséghatékony, innovatív információfeldolgozási technikák segítségével lehetővé teszi a jobb áttekintést, a döntéshozatal támogatását és folyamatok automatizálását."* [60] A technológia társadalmi hatásait *a Big data - Forradalmi módszer, amely megváltoztatja munkánkat, gondolkodásunkat és egész életünket* c. könyv [61] mutatja be.

2.2.1 Infokommunikációs rendszerek tervezése és fejlesztése

Az alábbiakban az MH HID-ben megjelenő informatikai fejlesztésekre [43, 1.3.5.] és szolgáltatásokra [43, 1.3.6.] vonatkozó követelmények kerülnek bemutatásra. A felülről építkező informatikai fejlesztést a következő módon írja le a dokumentum: *„a klasszikus rendszerfejlesztési módszerek alkalmazásával logikai szinten kialakítják a szervezet információs rendszerét, kidolgozzák az információfeldolgozási rendszer, majd az adatfeldolgozási rendszer tervét, meghatározzák a megvalósításhoz szükséges erőforrásokat, majd vagy elkészítik a programtervet és végrehajtják a programozást, vagy a piacról beszerzik a szükséges alkalmazói szoftvert, illetve megvásárolják a szükséges informatikai szolgáltatást.*" [43, 18. o.]

Ahhoz, hogy egy új szoftver kialakítása során vagy egy létező szoftver továbbfejlesztésekor a fejlesztést végző csapat sikeres tudjon lenni, megfelelő minőségű követelményeket kell támasztani az adott szoftverrel, illetve szoftver komponensekkel szemben megrendelői oldalon.

Hasonló megállapítással találkozhatunk az MH HID-ben is: *„a szolgáltatások fejlesztésével kapcsolatos követelmények a honvédségi szervezetekre általánosan jellemző információfeldolgozási folyamatok elemzéséből kerülnek meghatározásra az alkalmazó szervezetek és a híradó-informatikai szakmai szervezetek együttműködése eredményeként…"* [43, 19. o.]. Az informatikai szabályzat ezt még bővebben taglalja, külön kitér arra, hogy az infokommunikációért felelős szakmai szervezetek feladata az együttműködés koordinálása az alkalmazó szervezetekkel. [43, 2.3.4.2.] Szoftvertechnológiai megközelítésből az a kérdés merül fel, hogy az említett *"általánosan jellemző információfeldolgozási folyamatok"* elegendőek-e egy szoftverrel szemben támasztott hagyományos követelményspecifikáció vagy egy agilis követelményrendszer összeállításához valamilyen deklarált és lefektetett módszertan nélkül? A válasz valószínűleg az, hogy nem vagy csak alacsony minőségben.

Amiről még nem esett szó az a rendszeresített, illetve használatba vett szoftverek továbbfejlesztésének, kiváltásának előkészítése. Ilyen esetekben a forrás szoftver műszaki hátterének, funkcionalitásának feltárása, értelmezése komoly feladat elé állíthatja a követelményrendszer összeállítóját. A követelmények kinyerésének miként a szolgáltatásokra vonatkozó irányelvek bemutatását követően kerül tárgyalásra.

2.2.2 Informatikai szolgáltatásokra vonatkozó irányelvek

Az MH kötelékében rendszeresített informatikai szolgáltatásokkal kapcsolatban az alábbi lényeges követelmény jelenik meg: *„a híradó és informatikai szolgáltatás célja a szervezetek eltérő feladatainak végrehajtása érdekében alkalmazható olyan egységes információs képesség biztosítása, amelynek alkalmazásával a feladatokat rövidebb idő alatt, kevesebb erőforrás bevonásával és jobb minőségben hajthatók végre, mint a szolgáltatás igénybe vétele nélkül”*. [43, 19. o.] Itt egy a szolgáltatásokra vonatkozó irányelv jelenik meg, mely az informatikai rendszerek integrált szolgáltatásként történő kialakítását vetíti elő. Ehhez azonban szükséges az egyes felhasználási területek fogalomrendszerének tisztázása és olyan informatikai rendszerek tervezése, melyek letisztultan és konzisztensen képezik le azt a fogalmi rendszert, amelynek a problémáját meg kell oldaniuk. Minderre azért van szükség, mert az egységes szolgáltatásokat a fogalmak számának növekedésével egyre bonyolultabb egyetlen szoftverben, illetve szoftverfejlesztési projektben kezelni. A fejlesztés sebessége túlságosan lelassulhat, a bonyolultság kezelhetetlenné válhat. Ilyen esetben célszerű a modularizáció, az alrendszerekre bontás, mellyel a bonyolultság az alrendszereken belül csökkenthető, az alrendszerek funkciója, felelőssége tisztázható. Új modulok, alrendszerek bevezetésekor, illetve cseréjük során felül kell vizsgálni az egyes rendszerek határait és szükség esetén újra kell gondolni azokat.

A fejlesztések során kompatibilis technológiákat kell kiválasztani, amelyekkel az elkészült alrendszerek könnyedén integrálhatók és együttesen megfelelő minőségű szolgáltatást tudnak képezni. Egybe cseng ezzel a következő MH infokommunikációs szolgáltatásokra vonatkozó követelmény is: *„biztosítsák a honvédségi szervezetek vezetési-irányítási, végrehajtási és együttműködési információs folyamatainak hatékony támogatását, tegyék lehetővé a szervezetek és feladatok széles körében felhasználható híradó és informatikai szolgáltatások biztosítását, képezzék alapját a honvédségi szervezetek integrált, egységes elvek és követelmények alapján kialakított információs rendszerének.”* [43, 1.4.1] Ez egy teljes Magyar Honvédség viszonylatában értelmezhető irányelv, ennek megvalósításához magas szintű követelmény meghatározásra is szükség van.

A feldolgozott irodalom alapján látható, hogy az újonnan fejlesztett szoftverek követelményrendszerének meghatározására számos követelmény áll rendelkezésre. A kutatásom eredeti célja az ilyen jellegű feladatok támogatására használható dokumentációs technikák kidolgozása volt.

A kutatásom alatt azonban nyilvánvalóvá vált, hogy széleskörű követelményekkel találkozhatunk a létező szolgáltatásokra, rendszerekre vonatkozó dokumentumokban is. Ezekben az esetekben egyéb dokumentációs módszereket is be kell vetni ahhoz, hogy megfelelő minőségű követelményrendszer álljon rendelkezésre a szoftverfejlesztési projekt megvalósításának megkezdéséhez a követelmények megfelelő szintű elemzéséhez. Az általam azonosított infokommunikációs rendszerek fejlesztéséhez és fenntartásához kapcsolódó feladattípusok felsorolása az alábbiakban található. Jelen kutatás szemszögéből a felsorolás elemei speciális követelményelemzést támogató eljárásokra vonatkozó kritériumként jelennek meg a kialakítandó szoftverfejlesztési módszertannal szemben.

R-1 Új szoftverek fejlesztése – Még informatikai megoldással nem rendelkező területek lefedése egyedi fejlesztésű szoftverrel. Teljes informatikai megoldás kidolgozása a követelmények elemzésétől, a megvalósításon át, egészen a szoftver használatba vételéig.

R-2 Új műszaki alapokra történő helyezés – az informatika adta lehetőségek bővülésével és az igények növekedésével megeshet, hogy egy sok éven át használt rendszer architektúráját és komponenseit frissíteni kell, mert a jelenlegi formájában már nem fenntartható.

R-3 Szoftver cseréje előző rendszer alapján – elképzelhető, hogy a régi rendszernek van egy kifutási ideje, ebben az esetben a feladat az új rendszer követelményeinek meghatározása és az új szoftver implementálása, nagyon gyakori igény a régi rendszer funkcionalitásának megőrzése, annak jobbá tétele.

R-4 Adatmigrálás[30] *támogatása* – nagyon gyakori igény, hogy a lecserélendő szoftver adatainak átmozgatása is feladat a szoftver cseréje mellett. Az adatok közötti kapcsolatok megértése a forrás és cél rendszerekben egyaránt szükséges a sikeres végrehajtáshoz.

R-5 Rendszerek közötti integráció – két rendszer közötti kommunikáció megteremtése, folyamatok kialakítása az integrált rendszeren belül.

[30] adatmigrálás - áttérés egy alacsonyabb verziójú adatstruktúráról egy magasabb verziójúra vagy egy hasonló feladatot ellátó rendszer más szerkezetű adatstruktúrájára

2.3 ÖSSZEGZÉS

Az értekezés 2. fejezetében a Magyar Honvédség hatályos informatikai dokumentumainak szoftvertechnológiai perspektívából történő áttekintése valósult meg. Az elemző munkával párhuzamosan, az infokommunikációt érintő stratégiai célkitűzések áttekintése is megtörtént.

Az elemzés során négy általános módszertani elvárás került azonosításra az MH Informatikai Szabályzata alapján, majd az MH HID-ben helyet foglaló – szoftvertechnológiai szempontból is értelmezhető – tizenkét speciális informatikai követelmény elemzése következett, egyaránt szem előtt tartva az infokommunikációs rendszerek és a fejlesztési folyamatok perspektíváját.

Befejezésképpen, a szabályozó dokumentumok átfogó vizsgálata után, öt speciális követelményelemzési igény megfogalmazása is lehetővé vált. A továbbiakban a fejezetben feltárt követelmények szerepelnek kategóriákba sorolva.

Módszertan

M-1 A szolgáltatások minimális erőforrás és idő ráfordításával legyenek átalakíthatók, adaptálhatók a változó felhasználói igényeknek és alkalmazási körülményeknek megfelelően.

M-2 Az infokommunikációs szolgáltatások fejlesztése és biztosítása az alkalmazó szervezetek műveleti követelményei, az infokommunikációért felelős szervezetek szakmai követelményei, a beszerzés műszaki követelményei, a kialakításra kerülő rendszer rendszerterve, az alkalmazó szervezet alkalmazási terve, valamint a tevékenységet szabályozó jogszabályok, okmányok intézkedések alapján valósuljon meg.

M-3 Az infokommunikációs szolgáltatások biztosítását, tervezését, fejlesztését, felügyeletét támogató, a szolgáltatás teljes életciklusára értelmezhető szoftverfejlesztési módszertan kialakítása a cél.

M-4 A fejlesztési javaslatok elemzését, majd szabályozott módon történő változtatási igények meghatározását támogató módszertan kialakítása szükséges.

Szoftvertechnológia

S-5 Megbízhatóság

S-6 Szabványosság

S-7 Kompatibilitás

S-8 Interoperabilitás

S-9 Rugalmasság

S-10 Hitelesség

S-11 Modularitás

S-12 Skálázhatóság

S-13 Biztonság

S-14 Felhasználhatóság

S-15 Információ-megosztás

S-16 Adat-konzisztencia

Követelményelemzés

R-1 Új szoftverek fejlesztése

R-2 Új műszaki alapokra történő helyezés

R-3 Szoftver cseréje előző rendszer alapján

R-4 Adatmigrálás támogatása

R-5 Rendszerek közötti integráció

3. SZABVÁNYOS KATONAI KÉPESSÉGFEJLESZTÉS

A katonai képességfejlesztés speciális alesete a katonai célú szoftverfejlesztés, amely alapvető tárgyát képezi jelen disszertációnak. Ugyanakkor azt ki lehet jelenteni, hogy a különböző katonai képességek kialakítása során nagy valószínűséggel megjelennek a rendszer részét képező, illetve támogató funkciókat ellátó szoftver alapú komponensek, informatikai rendszerek. A területhez kapcsolódó kutatási hipotéziseim igazolása során szabványos, a védelmi ágazat számára kialakított folyamatokra támaszkodok a kapcsolódási pontok feltárásával, bemutatásával.

Jelen fejezet célja a kutatás során kialakítandó agilis szoftverfejlesztési technikák szabványos környezetbe történő elhelyezése. Szem előtt tartandó, hogy a kialakítandó dokumentációs technikáknak és módszereknek a későbbiekben a Magyar Honvédség számára is alkalmazható megoldásoknak kell lenniük. Az Észak-atlanti Szerződés Szervezete által kialakított katonai képességfejlesztésekre vonatkozó folyamatok, technikák és dokumentációs ajánlások feltárása megfelelő alapként szolgál a kutatási cél eléréséhez.

A NATO a katonai képességfejlesztés elősegítéséhez megalkotta a védelmi rendszerek életciklus-menedzsmentjét támogató átfogó dokumentációs keretrendszerét. A kialakított keretrendszer általános, ugyanakkor feltételezi a technológiai kihívásokat és alkalmazkodik is hozzájuk. Természetesen a különböző jellegű képességek fejlesztése különböző folyamatokat, így különböző dokumentációs technikákat is jelent. A terület mély megértéséhez az első alaposabb vizsgálatra szoruló dokumentum a NATO rendszerek életciklus menedzsment irányelve (a továbbiakban: SLCM[31] irányelv). [63]

A katonai képességek kialakítása, beszerzése önmagában is nagyon költséges folyamat, az SLCM feladata egy olyan szabványos meder meghatározása, amely segítségével garantálható a katonai képességekkel szemben meghatározott követelményeknek való megfelelés – a befektetés megtérülése. Ezen túlmenően az SLCM irányelv külön kiemeli, hogy azért is szükséges a szabályozott keretek kialakítása, mert a katonai rendszerek fenntartásának költségei a bevezetést követően az induló költségeket nagymértékben meghaladhatják. [63, 2] Az SLCM alapját a nemzetközi civil minőségirányítási szabványok képezik, ezzel elősegítve a közreműködők számára a

[31] SLCM – Az angol **S**ystems **L**ife **C**ycle **M**anagement kifejezés kezdőbetűiből.

határidőn belüli, hatékony és a katonai követelményeknek megfelelő, megfizethető költségű projektek megvalósítását. [63, 3-3.1] Egy NATO katonai képességfejlesztési program (a továbbiakban: NATO program) végrehajtása során minden érintett számára ajánlásként szolgálnak a projektek életciklusát leíró szakaszok[32], az azokon belül meghatározott eljárások és a kapcsolódó dokumentációs sablonok.

A NATO-ra háruló biztonsági feladatok ellátásához, a katonai siker eléréséhez komplex műveleti képességekre van szükség, amelynek záloga a megfelelő katonai képességek rendelkezésre állása. A SLCM irányelv azt is leszögezi, hogy a szabályozás és a felügyelet hiánya a Szövetség katonai képességeinek elégtelenségét eredményezheti. [64, 1.1] A szükséges műveleti képességek eléréséhez, humán és infrastrukturális oldalról egyaránt hatékony harcképességre, kitelepülésre, mobilitásra és túlélő képességre van szükség.

A felsorolt képességek kialakításához olyan katonai rendszerekre – ide értendők a katonai célú szoftverek – van szükség, amelyek az alábbi felsorolásban szereplő tulajdonságokkal rendelkeznek. [64, 1.1] A lista elemeit alkotó tulajdonságok általános katonai képességeket határoznak meg, ugyanakkor azok jelen esetben informatikai követelményekként kerültek interpretálásra. A továbbiakban feltételezzük, hogy tárgyalt katonai képességet egy informatikai rendszer valósítja meg, amely szoftver alapú szolgáltatás (SaaS) formájában kerül kialakításra és felhasználásra.

S-17 Hatékonyság – a funkcionális követelmények ellátása a lehető legalacsonyabb összköltség mellett. Informatikai rendszerek esetében is értelmezhető ez az elvárás. A szoftverfejlesztési projektek hatékonyságának [65] vizsgálatával Aneesh Chinubhai foglalkozik dolgozatában, a cikk az alkalmazott programozási nyelvek, a fejlesztési módszerek, a támogatási eszközök és a projekt menedzsment folyamatok kapcsolatát tárgyalja a hatékonyság szemszögéből. A hatékonyság növelése integrált rendszerszemlélettel lehetséges, a számottevő tényezők együttes vizsgálatával – minden érintett területen a megfelelő eszközök, módszerek alkalmazásával.

S-18 Telepíthetőség – egy informatikai rendszer azon tulajdonsága, amely meghatározza az adott rendszer teljes körű működőképes állapotának eléréséhez szükséges előkészületek, műveletek bonyolultságát. Az adott rendszer akkor telepíthető jól, ha ezek a folyamatok egyszerűek és számtalanszor megismé-

[32] Életciklus szakasz – NATO terminológiával: Lifecycle stage

telhetők, a végeredmény ellenőrizhető. Manapság a *Continuous Deployment* (röviden: CD[33]) technikáját alkalmazzák az informatikai iparágban a legjobb telepítési mutatók elérése érdekében. [66]

S-19 Robusztusság, túlélési képesség – az adott szoftver alapú szolgáltatásra, annak moduljaira, komponenseire értelmezett tulajdonság, amely a teljes rendszer életképességét írja le. Az adott szolgáltatás robusztus, ha a nagy terhelés mellett is el tudja látni a rendszerrel szemben követelményként támasztott funkciókat. Ugyanakkor, az egyes rendszerelemek kiesése sem jelenti a teljes rendszer kiesését. Az automatizált tesztelési technikák alkalmazása a leghasznosabb eszköz a megfelelő szintű robusztusság eléréséhez, illetve a követelmények kielégítésére alkalmas fokozásához. [67]

S-5 Megbízhatóság – ez a tulajdonság az MH HID feldolgozása során is feltárásra került (Lásd: 62. old), értelmezése ebben az esetben is megfelelő: jó minőségű, hibamentes, folyamatos működés az informatikai rendszer rendeltetésszerű használata mellett.

S-20 Karbantarthatóság – egy informatikai rendszer azon tulajdonsága, amely az adott rendszer módosíthatósági, fejleszthetőségi képességét írja le. Egy informatikai rendszer akkor rendelkezik jó karbantarthatósági mutatókkal, ha az új követelményeknek való megfelelés eléréséhez szükséges módosítások alacsony kockázat mellett szabványos folyamatok végrehajtásával elvégezhetők. Napjainkban a szoftverfejlesztésben tapasztalható egyik legnagyobb kihívás a karbantarthatósági képesség megőrzése. Különböző mérési technikák léteznek [68] egy fejlesztett szoftver karbantarthatósági mutatójának kiszámításához a szoftver alapú szolgáltatások esetében. Az alkalmazott módszerek kiválasztásakor azonban figyelembe kell venni az adott rendszer jellemzőit is.

S-21 Fenntarthatóság – az adott informatikai rendszer rendelkezésre állásának hosszú távú garantálása. A környezeti, technológiai, gazdasági, politikai, jogszabályi szempontok változása mellett is az adott rendszer képes a rá vonatkozó követelményrendszerben meghatározott funkciók ellátására. A szoftverfejlesztési projektek fenntarthatóságának meghatározása különféle értelmezéssel bírhat a fejlesztéstől a felhasználásig, különféle hangsúlyokkal, azon-

[33] CD – folyamatos szállítás/telepítés, a rövidítés az angol Continuous Delivery / Deployment kifejezések szavainak kezdőbetűiből származik.

ban minden területen célszerű tenni azért, hogy fenntartható folyamatok kerüljenek kialakításra, ezzel garantálva a projektek sikerét. [69, 4].

S-8 Átjárhatóság, interoperabilitás – ez a tulajdonság az MH HID feldolgozása során is feltárásra került (Lásd: 63. old.), értelmezése ebben az esetben is megfelelő: más rendszerekkel történő kétirányú együttműködés kialakításának képessége. Létező modulok, illetve komponensek beágyazhatósága.

Az SLCM irányelvben a kutatás szempontjából további lényeges és jellegzetes szoftvertechnológiai követelményeket sikerült feltárni a kialakítandó módszerekkel szemben. Az integrált és a költséghatékony védelmi képességfejlesztés eszköze maga az SLCM, amely megköveteli a résztvevőktől a hatékony együttműködést és a felsorolt tulajdonságok megtartását a védelmi célú rendszerek teljes életciklusa során. Az SLCM irányelv a *System-of-Interest* (a továbbiakban: SOI) angol kifejezést használja a katonai képességfejlesztések során kialakítandó rendszerek esetében. Ugyanakkor egy új katonai képességfejlesztési program több rendszer, azaz SOI kialakítását is jelentheti. A Szövetségi rendszerek életciklusa[34] a kialakítandó rendszerekre értelmezett fejlődési folyamatot fedi le a koncepciótól az adott rendszer használatba vételén át egészen a leszereléséig.

Az SLCM irányelv alapvetései között szerepel az együttműködési és az átjárhatósági képesség megteremtése a NATO rendszerek és a tagországok között. A Szövetségi katonai műveletek fenntarthatóságához a nemzeti és a NATO erőforrások hatékony és gazdaságos felhasználása alapvető fontosságú. Ehhez a civil szabványok felhasználása erős pillért ad, mert lehetővé teszi az együttműködést a gazdasági szereplőkkel. A piaci körülmények között bevált technikák és az új technológiák hasznosítása kedvező együttműködési lehetőségeket rejt magában a katonai képességfejlesztésben résztvevők számára. A katonai elvárásoknak megfelelő funkcionális és minőségi mutatók elérése integrált rendszerszemlélettel lehetséges, amelynek elősegítése különböző végrehajtási szinteken alkalmazható dokumentumok formájában valósul meg.

1. NATO Szabványosítási Megállapodás[35] (a továbbiakban: STANAG)
2. NATO Szövetségi Adminisztratív Kiadvány[36] (a továbbiakban: AAP)
3. NATO Szövetségi Minőségbiztosítási Kiadvány[37] (a továbbiakban: AQAP)

[34] Szövetségi rendszerek életciklusa – NATO terminológiával: **S**ystem **L**ife **C**ycle, röviden SLC
[35] NATO Szabványosítási Megállapodás – NATO **Stan**dardisation **Ag**reement (röviden: STANAG)
[36] NATO Szövetségi Adminisztratív Kiadvány – NATO **A**llied **A**dministrative **P**ublication (röviden: AAP)

Az alapvetéseken túl a kutatás szemszögéből ugyancsak alapvető fontosságú katonai képességfejlesztésre vonatkozó követelményeket is találhatunk az SLCM irányelvben. [64, (6)] A dokumentum a következőképp fogalmaz az SLCM rendeltetéséről és céljairól: *„az SLCM célja a NATO képességek hatékony és eredményes átadása, használata és üzemeltetése”*; [64, (6.1)] Az alábbi felsorolásban a kitűzött katonai elvárások elemzése szerepel.

1. *„A műveleti és a logisztikai követelmények, a megfizethetőség, az idő, az ütemezés, a minőség és a kockázati tényezők együttes értelmezése szükséges az SLCM-en belül”*. [64, (6.1.1)] – Ez egy általános katonai rendszerfejlesztéssel szemben támasztott követelmény, ugyanakkor némi átalakítással szoftverfejlesztési projektmenedzsment követelményként is értelmezhető: *a megfizethetőség, az idő, az ütemezés, a minőség és a kockázati tényezők együttes értelmezése szükséges a katonai célú informatikai rendszerek fejlesztése során.*
2. *„Integrált és zökkenőmentes üzletkötési, projektmenedzsment gyakorlatok kialakítása szükséges a koncepció-alkotástól a kivezetésig.”* [64, (6.1.2)] – Az utóbbi követelmény általános elvárásokat fogalmaz meg, azonban ugyanebben a formában is értelmezhető projektmenedzsment követelményként a szoftverfejlesztésre vetítve.
3. *„Teljes életciklusra vonatkozó, minden érdekelt fél között hatékony együttműködés kialakítása a cél, jól definiált felelősségi körökkel”.* [64, (6.1.3)] – Szoftverfejlesztési projektekre változatlanul értelmezhető követelmény.
4. *„Szükségszerű a technológiai bővítés lehetővé tétele, az élettartam alatti módosítások és az elévülés kezelése.”* [64, (6.1.4)] – Szoftvertechnológiai követelményként is értelmezhető elvárás.
5. *„A fejlesztések számára integrált rendszerszemlélet meghatározása szükséges, amely támogatja a felhasználást, elősegíti a követelményeknek való megfelelést, oly módon, hogy minimalizálja a projekt átfutási idejét, maximalizálja a hatékonyságot, valamint minimalizálja költségeket.”* [64, (6.1.5)] – Az elvárás első fele a követelményeknek való megfeleléssel bezárólag reális szoftvertechnológia célkitűzésnek tekinthető, ugyanakkor az átfutási idővel, a hatékonysággal és költségekkel kapcsolatos elvárás együttesen nehezen telje-

[37] NATO Szövetségi Minőségbiztosítási Kiadvány – NATO **A**llied **Q**uality **A**ssurance **P**ublication (röviden: AQAP)

síthető. A követelményhez kapcsolódó közgazdasági probléma külön tárgyalásra kerül a későbbiekben. Lásd: 135. oldal.

6. *„Olyan rendszerek kialakítása a cél, amelyek megfelelnek a hadműveleti és a logisztikai követelményeknek, optimalizálják a belső és a külső interfészeket, kezelik az integrált logisztikát, a szolgáltatásokra vonatkozó támogatást és minimalizálják a környezeti hatásokat a gyártási, működési és leszerelési időszakokban."* [64, (6.1.6)] – Túlnyomórészt fizikailag létrejövő katonai képességekre vonatkozó elvárás, amely a kutatás vonatkozásában nem sorolható be a korábban azonosított csoportokba, így további tárgyalásától el lehet tekinteni.

Az SLCM irányelv elemzése során összesen tíz darab új kutatási követelményt sikerült azonosítani. Először az informatikai fejlesztések vonatkozásban értelmezhető új követelményeket, majd az iménti felsorolásban, a katonai rendszerfejlesztésekre vonatkozó további elvárásokat. Előbbiek a kutatás által kialakított besorolási rendszerben a speciális szoftvertechnológiai követelményekhez sorolhatók. Utóbbiak esetében speciális módszertani és technológiai követelményekről beszélünk.

Módszertani követelmények

M-5 A megfizethetőség, az idő, az ütemezés, a minőség és a kockázati tényezők együttes értelmezése szükséges a katonai célú informatikai rendszerek fejlesztése során. Lásd: 75. old.

M-6 Integrált és zökkenőmentes üzletkötési, projektmenedzsment gyakorlatok kialakítása szükséges a koncepció-alkotástól a kivezetésig. Lásd: 75. old.

M-7 Teljes életciklusra vonatkozó, minden érdekelt fél között hatékony együttműködés kialakítása a cél, jól definiált felelősségi körökkel. Lásd: 75. old.

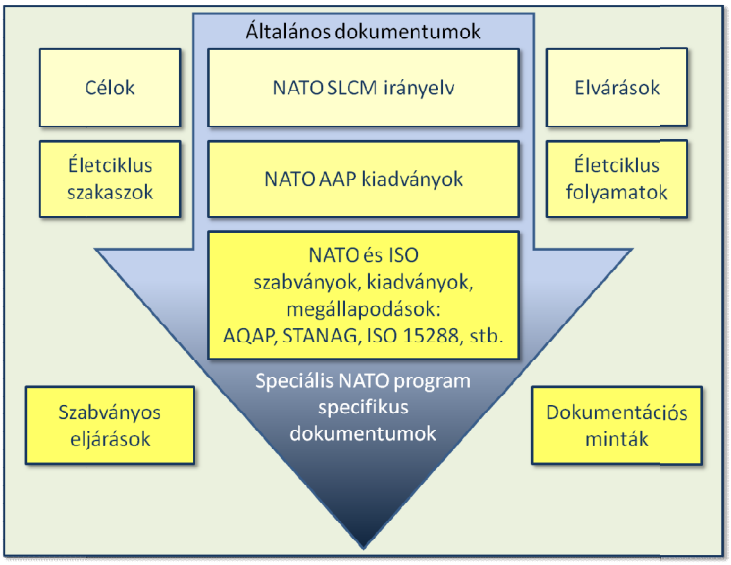
Szoftvertechnológiai követelmények

S-22 Szükségszerű a technológiai bővítés lehetővé tétele, az élettartam alatti módosítások, az elévülés kezelése. Lásd: 75. old.

S-23 A fejlesztések számára integrált rendszerszemlélet meghatározása szükséges, amely támogatja a felhasználást, elősegíti a követelményeknek való megfelelést. Lásd: 75. old.

Az SLCM és a korábban már említett AQAP szabványcsalád közötti kapcsolatokat további dokumentumok írják le AAP kiadványok formájában. Az AAP-20 [70] kiadvány a szakaszokra bontott életciklus modellt definiálja, míg az AAP-48 [71] kiadvány az egyes szakaszokhoz tartozó folyamatokat határozza meg, valamint a folya-

matokhoz kapcsolódó dokumentációs sablonokat és a rendelkezésre álló, alkalmazandó AQAP szabványokat is.



4. ábra A NATO SLCM komponensei (saját szerkesztés)

A 4. ábra alapján látható, hogy a megvalósításához három különböző szinten jelennek meg a támogató dokumentumok. Az első általános szintet az SLCM irányelv alkotja, majd az AAP-20 és az AAP-48 kiadványok együttesen támogatják az életciklus szakaszok és folyamatok végrehajtását, míg végezetül a különböző katonai képességfejlesztési projektekhez az SLCM dokumentumkönyvtár adja speciális dokumentációs eszközöket.

Kutatásom szemszögéből a második szint részletes feltárása is indokolt, az értekezés során létrejövő módszerek és technológiák illesztése a NATO SLCM keretrendszerhez megkívánja a feltáró munka folytatását a szabványos környezetbe történő illesztés elősegítése érdekében.

3.1 NATO AAP-20 ÉLETCIKLUS MODELL

Az SLCM-hez tartozó életciklus szakaszok részletes bemutatása az AAP-20 dokumentumban található meg. A dokumentum általános célja egy szabványos és testre szabható megközelítés lefektetése a NATO katonai képességfejlesztési programok menedzseléséhez. Az általános cél eléréséhez szükséges feladatok három csoportba sorolhatók:

1. a NATO programok életciklusához tartozó szakaszok strukturált meghatározása;
2. szakaszok közötti átmenetekhez szükséges döntési pontok kialakítása;

3. döntéshozatal támogatása minden szinten. 70, 1.2]

Az AAP-20 szoros együttműködést vár el a hadműveleti vezetéstől, a katonai tervezéstől és a NATO programok menedzsereitől. Egy katonai képességfejlesztési tevékenység alapját egy feltárt hiányosság vagy stratégiai cél elérése indokolhatja, amennyiben valamelyik eset előáll, akkor a katonai tervezőkön keresztül kerülhet egy követelmény a program menedzserei elé. Az SLCM-nek támogatnia kell a követelmények meghatározását, a programok végrehajtását, az életciklussal kapcsolatos menedzsmentet, a gyorsított beszerzések és a technológiai bővítés elősegítését is. A tagországokban megvalósuló katonai célú beszerzési eljárások támogatására is alkalmazható a keretrendszer, ugyanakkor a nemzeti beszerzési és fejlesztési gyakorlatok kiváltása nem célja. [70, (1.3)]

A megismert alkalmazási célok tükrében, a továbbiakban az életciklus szakaszok elemzése során a potenciális szoftvertechnológiai kapcsolódási pontok feltárása és rendszerezése a kutatás során követendő irányvonal.

A NATO stratégiai céljainak [72] eléréséhez a katonai képességek folyamatos fejlesztésére van szükség. A hiányzó képességeket megfelelően kiforrott, működő megoldások, illetve felszerelések segítségével kell pótolni. Egy program kiindulási alapját az újonnan kialakítandó katonai képességek képezik, amelyek meghatározását katonai tervezők végzik. Az erőforrások biztosítása és a szükséges döntések meghozatala a NATO különböző szervezetei és a tagországok által valósulnak meg. Amennyiben rendelkezésre állnak a megfelelő erőforrások és döntések, akkor nincs gátja a program végrehajtásának egy adott katonai képesség kialakítása céljából. A program céljaként meghatározott képesség akár több kialakítandó SOI-ból is állhat. A programok menedzseléséhez és a döntéshozatali folyamatok támogatásához a végrehajtás előre meghatározott szakaszok alkalmazásával valósul meg. Minden szakasz valamilyen tevékenység típushoz köthető periódust fed le a NATO rendszerek és a SOI-k vonatkozásában. Egy szakasznak időkorlátokhoz köthetőnek, érthetőnek és behatárolhatónak kell lennie a szakaszon belül végzett tevékenységek szemszögéből. A szakaszok közötti átmenet döntési kapuk segítségével történik. A folyamatot belépési és kilépési feltételekkel lehet végrehajtani. A szakaszok segítenek azonosítani a kockázatokat, támogatják az ütemezést, az általános célokat és a döntéshozatalt. A szakaszokon belül további mérföldkövek is elhelyezhetők, hogy az előrehaladás megfelelőképpen legyen kontrollálható.

5. ábra A NATO programok, a SOI-k és a katonai képességek kapcsolata, forrás: [70, 6. o.]

Az 5. ábra foglalja össze egy kívánt katonai képesség[38] kialakítására indított NATO program főbb folyamatait és lépéseit. A program céljaként meghatározott képességek kialakításához a programokat projektekre lehet bontani. Az SLCM keretrendszerhez kapcsolódó dokumentumok a program- és projekt menedzsment feladatokat egyaránt tárgyalják. A koncepcionális tervezés a kialakítandó képesség teljes egészére vonatkozik, azonban ezt követően további projektek indíthatók különböző rendszerfejlesztési céllal, amelyek produktumai együttesen alkotják majd a kívánt katonai képességet. Természetesen a folyamat bevonhatja a korábban rendelkezésre álló katonai képességeket is. A program végét a leszerelés szakasza zárja.

A projektek céljaként kialakítandó SOI-k között természetesen lehetnek szoftver alapú szolgáltatások is, amelyekre értelmezhető az SLCM keretrendszer, így a továbbiakban az AAP-20 kiadványban lefektetett általános fogalmak és folyamatok tárgyalása következik szoftvertechnológiai aspektusból.

3.1.1 Folyamatos rendszerszemlélet és fenntarthatóság

Egy katonai képesség kialakítása túlmutat az adott felszerelés vagy eszköz egyszeri előállításán. Az adott katonai rendszer akkor tekinthető késznek, érettnek a bevezetésre, ha a katonai képesség ellátása mellett rendelkezik a megfelelő támogató rendszerekkel, illetve megfelelő mutatókkal rendelkezik hadműveleti környezetben a

[38] Kívánt katonai képesség – NATO terminológiával: Required Military Capability

használatba vételhez. Egy autonóm légvédelmi rendszer vagy egy információs rendszer esetében a működési környezet és a támogató rendszerek eltérnek egymástól, ugyanakkor mindkét esetben értelmezettek ezek a fogalmak.

A támogató rendszerek általánosságban logisztikai, személyzeti és infrastrukturális jellegűek. [70, 2.2] Szoftvertechnológiai vonatkozásban is azonosíthatók az analóg rendszerek: fejlesztést támogató szoftverek, a fejlesztést és a rendszerszervezést végző szakemberek, az üzemeltetés támogatására szolgáló szoftverek és az üzemeltető állomány. Mindannyian részét képezik a támogató rendszereknek. Ezek megfelelő szintű érettsége elengedhetetlen az egyes szakaszok sikeres végrehajtásához.

Az AAP-20 kiadványban meghatározott *Rendszer Koncepció* kimondja, hogy egy NATO rendszer esetében a rendszert alkotó hardvereknek, szoftvereknek és szolgáltatásoknak minden alkotóelemükkel együtt folyamatosan megfelelő érettségi szintet kell elérniük egészen az adott rendszer leszereléséig. A dokumentum úgy fogalmaz, hogy a kialakítandó katonai képesség akkor realizálódik, amikor az adott képességet megvalósító rendszerek elérik a hadműveleti érettség meghatározott, elégséges fokát. [70, 2.3] Az előző megállapítás alapján az is előfordulhat, hogy egymástól különböző, független rendszerek érettségi szintjének kell elérniük a katonai képesség realizálásához szükséges megfelelő szintet.

Az AAP-20 kiadványban az alábbi felsorolással találkozhatunk [70, 8. o.], az eredeti szövegkörnyezetben a fogalmak mellett általános értelmezés szerepel, azonban jelen esetben a felsorolás elemei szoftvertechnológia kontextusban szerepelnek.

1. Infrastruktúra – az adott szoftver alapú szolgáltatással szemben támasztott fejlesztési, tesztelési, üzemeltetési követelmények infrastrukturális tényezői.
2. Szervezet – a szolgáltatás kialakításához szükséges személyzeti követelmények összessége fejlesztői, tesztelői, rendszerszervezői kompetenciák vonatkozásában a megfelelő információ technológiai támogatás kialakításával.
3. Oktatás, gyakorlat – a bevezetéséhez szükséges felhasználói, műszaki kézikönyvek, oktatási anyagok, tesztrendszerek.
4. Támogatás – az üzemeltetéséhez szükséges eljárások, karbantartási munkálatok, technikai leírások gyűjteménye.
5. Egyéb – olyan technológiai és módszertani megkötések, amelyek az adott katonai képesség megfelelő érettségi szintjét elő tudják segíteni.

A NATO programok célja azon túl, hogy a kívánt katonai képességek előálljanak, az is, hogy a képességet megvalósító rendszer vagy rendszerek és a kapcsolódó támoga-

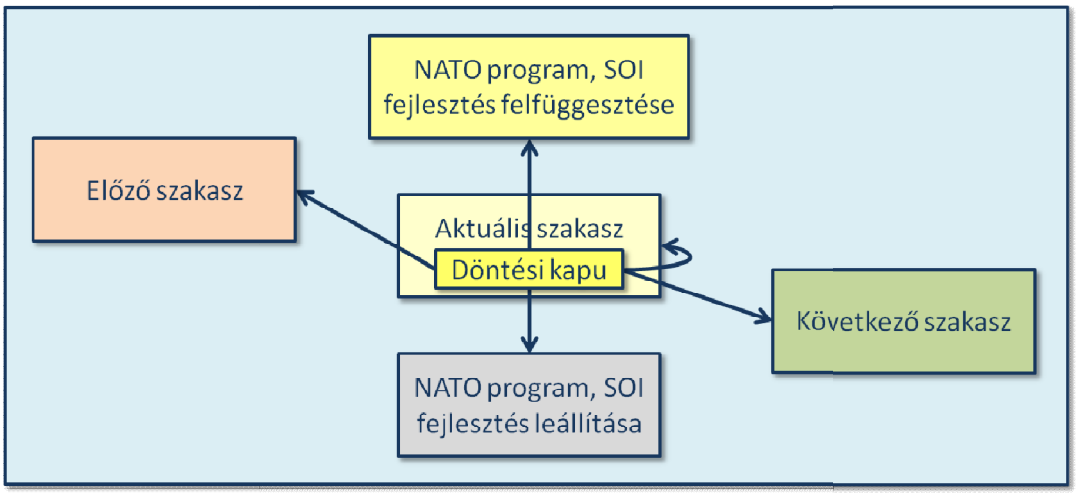
tó rendszerek érettségi szintje folyamatosan elérje a hadműveleti érettség megfelelő szintjét. [70, 2.3] Ha a kialakítandó katonai képesség egy olyan rendszer, amelyhez szoftverfejlesztési feladatok elvégzése is társul, akkor is azonosíthatók a szükséges támogató rendszerek és a működési környezet. Ezt azt jelenti, hogy nemcsak az adott szoftver alapú szolgáltatás érettségének, hanem a kapcsolódó szoftverfejlesztési projekt érettségének is mérhető tényezőnek kell lennie. Technológiai oldalról a rendszert alkotó komponensek és modulok számára az érettségi mutatók mérése önállóan is szükséges. Ugyanez igaz a rendszer egészét illetően is a teljes előállítási folyamat során, így egyfajta szoftverekre értelmezett gyártástechnológiai folyamat alakítható ki. Az érettségi modellnek történő megfelelés eredményeként egy NATO elvárásoknak megfelelő szoftverfejlesztési módszertan határozható meg. Újabb speciális módszertani követelményt sikerült azonosítani, miszerint: *A kialakítandó szoftverfejlesztési módszertan értelmezze az érettségi szint fogalmát és feleljen meg a NATO Rendszer Koncepciónak.*

3.1.2 Döntési pontok és mérföldkövek

Amint eddig is láthattuk az AAP-20 kiadványban kialakított ajánlások átfogó megközelítést adnak a NATO programok végrehajtási szakaszaira vonatkozóan. Célszerű megismerni a szakaszok végrehajtásának és a szakaszok közötti átmenetek alacsonyabb szintű modelljét is, ahhoz hogy a kutatás eredményeként kialakítandó szoftverfejlesztési módszertan megfelelő részletességgel támogassa a NATO rendszerek életciklus modelljét.

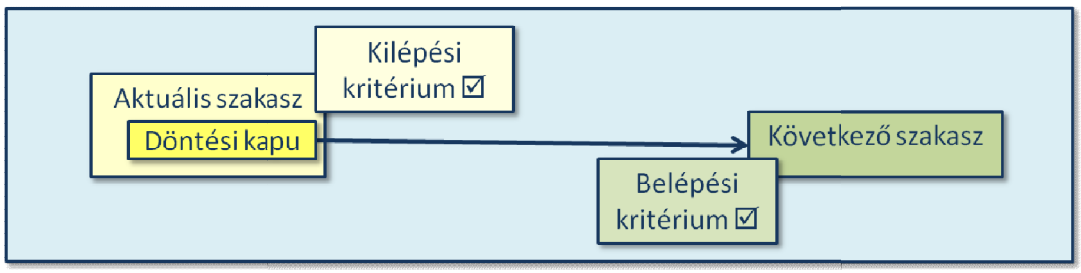
Az kialakítandó katonai képességet alkotó rendszerek érettségének ellenőrzése folyamatosan elvégzendő feladat, az ellenőrzés két szintjét határozza meg az AAP-20. Az első szint a szakaszok között átmeneteket jelenti, míg a második szint a szakaszokon belüli mérföldkövek elérést takarja. Az első szinten, döntési kapukon történő áthaladás segítségével valósul meg a szakaszok közötti átmenet, optimális esetben a következő szakasz megkezdésével.

A továbblépésen túl még további négy döntés születhet a döntési kapun történő áthaladáskor: visszalépés egy előző szakaszba, jelenlegi szakasz folytatása, továbblépés a következő szakaszba, a NATO program/SOI fejlesztés felfüggesztése, illetve a NATO program/SOI fejlesztés leállítása.



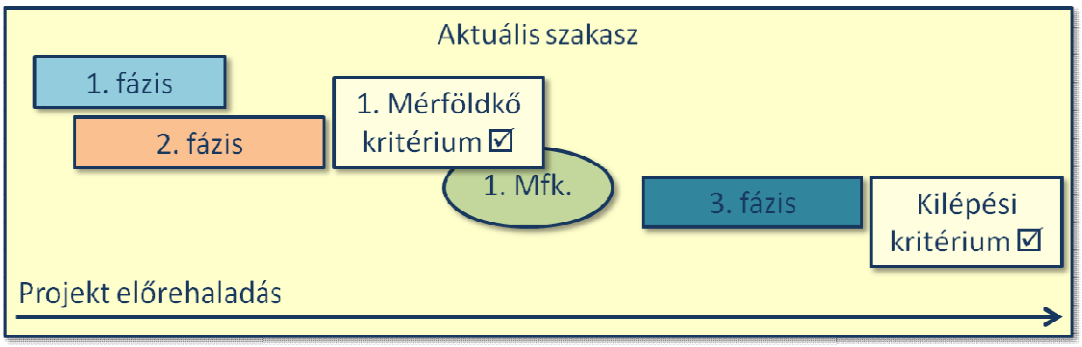
6. ábra Döntési kapu működése (saját szerkesztés)

A döntési kapukon történő áthaladás előfeltétele a megelőző szakasz során keletkezett, a döntéshozatalhoz szükséges kimenetek megléte. Egy szakasz kimenetei lehetnek: egy elvégzett feladat vagy tevékenység végterméke, kapcsolódó dokumentáció, kialakított környezet, stb.. A döntési kapukon történő előre haladáshoz ellenőrizni kell az előző szakaszban kialakított rendszerek érettségét, ugyanakkor a következő szakasz feladatainak is véglegesnek kell lenniük. Az előrehaladási folyamatot szem előtt tartva egy időben két feltételnek is meg kell felelni: az aktuális szakasz kilépési kritériumának, valamint a következő szakasz belépési kritériumának.



7. ábra Szakaszok közötti előrelépés folyamata (saját szerkesztés)

A mérföldkövek az adott szakaszon belüli előrehaladás ellenőrzését teszik lehetővé, a mérföldkövek esetében is lehetőség van a szakaszon belüli ellenőrzések elvégzésére, amely segítségével az előrehaladás nyomon követhető. Az SLCM lehetőséget biztosít az életciklus szakaszokon belüli fázisok definiálására is, amelyek sorban és átfedésekkel is elhelyezhetők az adott szakaszon belül.



8. ábra Szakaszok közötti előrelépés folyamata (saját szerkesztés)

Az eddigiek során sikerült megismerni a NATO programok végrehajtását elősegítő rendszerszemléletet, illetve a projektfolyamatok kialakításakor alkalmazható fogalmakat – *szakaszok, fázisok, mérföldkövek* formájában.

A továbbiakban az AAP-20 kiadványban található életciklus szakaszok rövid elemzése következik a koncepcionális tervezéstől a rendszerek kialakításáig, végezetül a leszerelésükig. Az elemzés célja az egyes életciklus szakaszokra szoftvertechnológiai szempontból értelmezhető folyamatok, eljárások és dokumentumok azonosítása. A feltáró munka alatt azonosított tudásanyag a kutatási célként kitűzött szoftverfejlesztési módszerek meghatározása során kerül majd további felhasználásra, beépítésre.

3.1.3 Koncepcionális előtervezés szakasza[39]

A sorban az első szakasz a koncepcionális előtervezés szakasza, amelynek legfőbb célja a követelmények azonosítása és a kockázatok feltárása. Ebben a szakaszban kell körvonalazni a fejlesztési képességeket, a technológiai korlátokat, az időbeni és pénzügyi lehetőségeket. Ez a szakasz képezi a tagországi- és NATO program menedzsment keretrendszer alkalmazási területének előszobáját. Egy újonnan kialakítandó katonai képesség eléréséhez a Szövetségnek és a tagországoknak is maximális hatékonysággal kell megvalósítaniuk a rájuk háruló feladatokat. Nemzeti vagy nemzetközi katonai képességfejlesztési programok esetében is fel kell készülni a többszereplős, összetett folyamatokat takaró feladatrendszerre.

A NATO védelmi tervezés folyamata[40] azon lépéseket foglalja magába, amelyeket el kell végezni egy NATO program végrehajtásának megkezdése előtt. [70, 15. o.] A következő felsorolás elemei az AAP-20 kiadványban meghatározott védelmi tervezés (NDPP) lépései. A hozzájuk kapcsolódó leírások az eredeti dokumentum kivonatai, bővítve informatika rendszerfejlesztésekre értelmezhető elvárásokkal.

[39] Koncepcionális előtervezés szakasza – NATO terminológiával: Pre-Concept Stage
[40] Védelmi tervezési folyamata – NATO terminológiával: Defence Planning Process (röv.: NDPP)

1. *„Politikai Iránymutatás kialakítása”* avagy a politikai konszenzus megteremtése – a katonai képességfejlesztés igényének felmerülését követően kellő részletességgel tisztázni kell a közös célokat és az elképzeléseket. A dokumentum célja, hogy megalapozza a kívánt képességgel szembeni követelmények meghatározását. Az informatikai fejlesztések esetében is szükséges a követelménytámasztásban érintett szereplők meghatározása, szempontjaik egységesítése, az irányok és a stratégiai célok lefektetése. Egy szoftver alapú szolgáltatás kialakításának megkezdésekor már ki lehet térni a fejlesztéssel, a támogatással, és az üzemeltetéssel kapcsolatos stratégiai követelményekre. Elvárásként megfogalmazható, hogy a stratégiai célok és követelménytámasztásban résztvevő szereplők meghatározása képezze részét a kutatás eredményeként létrejövő szoftverfejlesztési módszertannak.
2. *„Követelmények meghatározása”* – a stratégia célok és az érintett szereplők meghatározását követően lehetségessé válik a célok megvalósításához szükséges képességek azonosítása. Az azonosított képességek – informatikai rendszerek esetében funkciók – lehetővé teszik a kialakítandó rendszerek, valamint a támogató rendszerek koncepcionális tervezését, egyfajta alapot képezhetnek a becslések számára. Magas szintű követelményrendszer állhat elő – korábbi tapasztalatok felhasználásával akár durva erőforrásbecslés is végezhető ebben a nagyon korai szakaszban is. A kutatási célként kitűzött dokumentációs technikáknak támogatniuk kell az előzetes követelményrendszer összeállítását, illetve a durva erőforrásbecslések elvégzését.
3. *„Követelmények arányosítása és célok kitűzése”* – a NATO védelmi tervezés folyamatában a minimum katonai képességre vonatkozó követelmények meghatározása és a tagországi követelmények szintjére történő felosztása a legfontosabb feladat. A fejlesztési célok eléréséhez elengedhetetlen a közös pénzügyi alap megteremtése és a feladatok költségvetési tételekhez rendelése. Ugyanakkor, ha a nemzetközi biztonsági helyzet úgy hozza, időközben is előfordulhat, hogy változás következik be a védelmi fejlesztések tartalmában vagy ütemezésében. Informatikai rendszerek, különösen szoftver alapú szolgáltatások fejlesztése során a változások kezelése folyamatos. A tevékenység támogatását megfelelő eszközökkel és technikákkal kell támogatnia a kialakítandó szoftverfejlesztési módszereknek.

4. *„A végrehajtás elősegítése”* – a stratégiai célok teljesülésének érdekében nemzeti és nemzetközi erőfeszítésre van szükség a NATO programok végrehajtása alatt. Az elégtelen képességek javítását vagy a lassú fejlesztési előrehaladást támogató intézkedések segítségével lehet támogatni. Az intézkedések előkészítésében a kívánt képesség eléréséért tett erőfeszítések értékelése különösen fontos. Az intézkedések eszköze lehet a nemzeti megvalósítás szorgalmazása vagy a nemzetközi megvalósítás elősegítése és támogatása. A végrehajtás elősegítése folyamatos feladatként hárul a NATO-ra. Informatikai fejlesztések esetében a megfelelő projektkörnyezet biztosítása, illetve a rendszer megvalósításának felügyelete jelenti a végrehajtás elősegítését. Olyan módszerek kialakítása kívánatos, amelyek megfelelő visszajelzést adnak a fejlesztői és megrendelői oldal számára is a képesség állapotát és a fejlesztés előrehaladottságát tekintve.
5. *„Eredmények áttekintése”* – a védelmi tervezés részeként a képességek időszakosan történő felülvizsgálatát valósítja meg a NATO. A folyamat során a Szövetség felülvizsgálja és értékeli a tagállamok védelmi és költségvetési terveit. Az összevont nemzeti hadseregek képességeinek értékelésével ellenőrizhető, hogy azok milyen mértékben felelnek meg a stratégiai céloknak. Az eredmények áttekintése lehetőséget biztosít a visszajelzések és a kapcsolódó ajánlások elkészítésére, az eredmények a következő védelmi tervezési ciklus előkészítő dokumentumaiként szolgálhatnak. Szoftverfejlesztési projektek esetében ez a fajta ellenőrzés különös fontos, mert a folyamatos változások miatt előfordulhat, hogy korábban működő funkciók elveszítik hatékonyságukat, vagy akár a követelményeknek megfelelő működési képességüket. Olyan módszerek meghatározása szükséges, amelyek képesek jelezni a rendszerek pazarló vagy helytelen működését.

A feltárt NATO folyamatok alapvetően fizikailag létező haderőtervezést és fejlesztést mutatnak be, ugyanakkor a kibertér esetében elképzelhető, hogy egy szoftver alapú szolgáltatás önmagában katonai képességet reprezentál. A felsorolás elemzése során megfogalmazott informatikai elvárások egy folyamat szintű követelménnyé alakítható a következők szerint. *A kialakítandó szoftverfejlesztési módszer adjon eszközöket a stratégiai célok dokumentálására, a szereplők azonosítására, a követelmények meghatározására és priorizálására, támogassa a változáskezelést, valamint folyamatosan adjon megfelelő visszajelzést a rendszer érettségével kapcsolatban.*

3.1.4 Koncepcióalkotási szakasz[41]

A koncepcionális előtervezés szakaszában meghatározott követelmények további finomításra szolgálnak, többek között erre való a koncepcióalkotási szakasz. Ebben a szakaszban ki kell alakítani az alapvető rendszerkövetelményeket és a megfelelően kivitelezhető tervezési megoldásokat. A követelmények megalapozottságát és a megvalósítás alapját képező megoldások megfelelőségét a koncepcióalkotási szakaszban kell elvégezni. Ez a szakasz két fázisra bomlik, amelyek között egyetlen egy mérföldkő helyezkedik el.

Az SLCM koncepcionális tervezési szakaszának első eleme a *felmérési fázis*[42], amely során az adott NATO program vagy projekt során használható alternatív technikák kiértékelése történik meg a kialakítandó katonai képességekre vonatkozó ismert követelmények tükrében. Az AAP-20 kiadvány külön kiemeli, hogy a szakasz során kidolgozott megoldási javaslatok, tanulmányok formájában iteratív folyamatok során állnak elő. [70, 17. old.] A piacról elérhető COTS rendszerek vizsgálata a fázis során végrehajtandó feladat, az költséges saját fejlesztések kiváltásának érdekében. Ha ez nem lehetséges, akkor a fázis feladata olyan megoldási javaslatok azonosítása, amelyek megfelelnek a rendszerrel szemben támasztott átfogó követelményeknek. Akkor ér véget a fázis, ha megfelelő számú tanulmány állt elő a megvalósíthatósági alternatívák számára. A következő felsorolás a megvalósíthatósági tanulmányok tipikus elemeit tartalmazza:

1. *„Küldetés/Művelet/Képesség szintű elvárások leírása*
2. *Nemzeti és NATO rendszerek elemzése, amelyeknek megfelelhetnek az elvárásoknak*
3. *Alternatív rendszermegoldások és technikai jellemzőik bemutatása*
4. *NATO szintű képességek az alternatív rendszermegoldások esetében*
5. *Időbecslés az alternatív megoldások esetében (K+F, gyártás, beszerzés, készletezés, teljes műveleti képesség, mérföldkövek és ütemezés)*
6. *Gazdasági és vezetői aspektusok*
7. *Logisztikai és szabványosítási követelmények az alternatív megoldások vonatkozásában, beleértve az infrastruktúrát*
8. *Az alternatív megoldásokra vonatkozó kiértékelési követelmények*
9. *SWOT elemzés az alternatív megoldásokra vonatkozóan”* [70, 17-18-old.]

[41] Koncepció alkotási szakasz – NATO terminológiával: Concept Stage
[42] Felmérési fázis – NATO terminológiával: Study Phase

Szoftverfejlesztési projektek esetében ezen a szinten a szóba jöhető technológiák, architekturális megoldások összehasonlítása végezhető el. A fenti felsorolás elemeit figyelembe véve új speciális elvárás támasztható a kutatási célként kitűzött szoftverfejlesztési módszertannal szemben. *Az alternatív megvalósíthatósági tanulmányok kialakítását folyamat szinten is támogassa a kialakítandó szoftverfejlesztési módszertan.*

A felmérés fázisát követően elegendő számú tanulmánynak kell rendelkezésre állnia a kívánt katonai képesség megfelelő szintű kialakításához, így kezdetét veheti a *programindítási fázis,*[43] amely abból indul ki, hogy a politikai és technológiai konszenzus már létrejött, nincs akadálya a továbblépésének. Ebben a fázisban a követelmények további finomítása a közös feladat, amely kidolgozottsága, részletessége lehetővé teszi a NATO program sikeres végrehajtását.

A programindítási fázis során szükséges egy projekt menedzsment iroda[44] kialakítása, amely a program végrehajtásáért felel. Ettől a pillanattól kezdve a program felügyelete kizárólag az érintett résztvevőkre tartozik. A PMO létrejöhet a korábbi fázisokban kialakított szervezetek átalakulásával, valamint új szereplők bevonásával is. A NATO program vezetéséhez a PMO mellett szükséges az irányító bizottság[45] felállítása is, amelynek fő feladatai:

- Fejlesztés előrehaladásának felügyelete
- Követelményrendszer felügyelete
    - a teljes rendszer vonatkozásában
    - az alrendszerek vonatkozásában
- Tervezési kérdések megfontolása

További irányító bizottsági feladatok a különböző projekt-tényezők közötti összefüggések egyeztetése és a kielégítő egyensúly megteremtése közöttük:

- képességekre vonatkozó követelmények;
- projekt végrehajtási idő;
- költségek.

A három tényező között fennálló közgazdasági probléma és az agilis szoftverfejlesztés kapcsolata részletes tárgyalására a következő fejezetben kerül sor. Lásd: 135. old. Miután a kialakítandó rendszerrel és a támogató rendszerekkel szemben támasztott

[43] Programindítási fázis – NATO terminológiával: Programme Establishment Phase.
[44] Projekt menedzsment iroda – NATO terminológiával: **P**roject **M**anagement **O**ffice, röviden: PMO
[45] Irányító bizottság – NATO terminológiával: Steering Committee

követelmények véglegesítésre kerültek, a megfelelő dokumentáció előállítása szükséges a NATO program elindításához. Ilyen dokumentumok a programmal kapcsolatos egyetértési nyilatkozat[46], illetve a szakasz lezárására alkalmas dokumentáció, amely tartalmazza a koncepció alkotási szakasz eredményeit: technikai, pénzügyi, iparági megállapodások. A dokumentáció előállításához javasolt tevékenységek szereplők szerinti csoportosítása a 3. táblázatban található.

Szoftverfejlesztéshez kapcsolható tevékenységek	**Előállított dokumentumok**
Szoftverfejlesztési módszer és projektmódszertan meghatározása (Dev[47]) 1. Életciklus modellek kiértékelése 2. A program/projekt menedzsment tervek elkészítése/módosítása	Szoftverfejlesztési módszertanra vonatkoztatható dokumentáció 1. Kiválasztott életciklus modell 2. Program menedzsment terv 3. Projekt menedzsment terv
Követelményelemzés, specifikáció (RA[48]) 1. Követelmények finomítása 2. Alapvető rendszerkövetelmények előállítása 3. Tervezési megoldások: ábrák, modellek, prototípusok, stb. 4. Megkötések kialakítása elemzése – szoftver intenzív programok esetében. A koncepció alkotási szakasznak figyelembe kell vennie az architektúrát, integrációt és a lehetséges tanúsítási megkötéseket.	Követelményekre vonatkozó dokumentáció 1. A javasolt megoldás bemutatása 2. Az érdekeltek által meghatározott kezdeti rendszerkövetelmények 3. Kezdeti tervezési dokumentáció
Program-, projektindítás (PM[49]) 1. Az életciklusra vonatkozó költségek becslése és a humán erőforrásra vonatkozó követelmények kialakítása 2. Előzetes program ütemezés kialakítása 3. Konfiguráció menedzsment terv kialakítása	Program-, projektindító dokumentáció 1. Életciklusra és HR költségekre vonatkozó dokumentáció 2. Előzetes projektütemezés 3. Kezdeti konfiguráció menedzsment terv
Kockázatelemzés (QA[50]) 1. Megvalósíthatósági tanulmányok értékelése 2. Kockázatelemzés	Kockázatelemzési dokumentáció 1. Megvalósíthatósági értékelés 2. Kezdeti kockázat meghatározás, értékelés és kezelési tervek.

[46] Egyetértési nyilatkozat – NATO terminológiával: **M**emorandum **o**f **U**nderstanding, röviden MOU.
[47] Dev – fejlesztés, az angol **Dev**elopment szó betűiből képzett rövidítés.
[48] RA – követelményelemzés, az angol **R**equirement **A**nalysis szavak kezdőbetűiből képzett rövidítés.
[49] PM – projektmenedzsment, az angol **P**roject **M**anagement szavak kezdőbetűiből képzett rövidítés.
[50] QA – minőségbiztosítás , az angol **Q**uality **A**ssurance szavak kezdőbetűiből képzett rövidítés.

Szoftverfejlesztést kevésbé érintő tevékenységek	Előállított dokumentumok
1. Integrált logisztikai támogatási terv kialakítása	1. Kezdeti integrált logisztikai terv
2. Elévülés menedzsment terv kialakítása	2. Kezdeti elévülés menedzselési stratégia
3. Műveleti koncepció kialakítása	3. Műveleti koncepció
4. Tanulságok összegzése (PM)	4. Tanulságok

3. táblázat A koncepció alkotási szakasz tevékenységeinek és dokumentumainak kapcsolata (saját szerkesztés)

A kialakítandó szoftverfejlesztési módszertannak figyelembe kell vennie a 3. táblázat szoftverfejlesztéshez köthető tevékenységei során előállított dokumentumok tartalmi elemeit. Dokumentációs követelményként megfogalmazva ezt, *a NATO SLCM koncepció alkotási szakasz során keletkező szoftverfejlesztéshez köthető dokumentumoknak helyt kell kapniuk a kialakítandó szoftverfejlesztési módszertan dokumentum sablonjaiban.*

3.1.5 Fejlesztési szakasz[51]

Az alapos tervezést követően, amely egyaránt kitér a kialakítandó rendszerre és a kapcsolódó támogató rendszerekre, lehetővé válik a fejlesztési tevékenység elkezdése. A fejlesztési szakasz és az ezt követő életciklus szakaszok meghatározása fizikailag létrejövő elemeket feltételez, amely eltér a szoftver alapú képességek kialakításától. Ugyanakkor megfigyelhető az is, hogy a szakaszok bemutatása során gyakori az utalás a *szoftver intenzív* katonai képességfejlesztési programokra. Jelen kutatás szemszögéből célszerű feltárni a fejlesztési és a további szakaszok számára előírt tevékenységeket és a kapcsolódó dokumentum típusokat is a kialakítandó módszerekkel szemben támasztott követelmények véglegesítéséhez.

A fejlesztési szakasz nem kerül külön tagolásra fázisok szerint, ugyanakkor nyolc szoftverfejlesztési aspektusból is érdekes, elkülönülő mérföldkövet tartalmaz: *követelmények áttekintése, funkcionális áttekintés, tervezési áttekintés, tesztelésre alkalmasság ellenőrzése, konfigurációs felülvizsgálat, megfelelőségi ellenőrzés, helyességi ellenőrzés, éles üzembe helyezésre alkalmasság ellenőrzése*. *Elvárás, hogy a felsorolt mérföldkövek jelenjenek meg a kutatás céljaként kitűzött szoftverfejlesztési módszer folyamataiban is.*

A fejlesztési szakasz elsődleges feladata az *előállítási szakaszban*[52] alkalmazandó technikai megoldások gyártásra alkalmas állapotra hozása. *„Szoftverek esetében a*

[51] Fejlesztési szakasz – NATO terminológiával: Development Stage
[52] Előállítási szakasz – NATO terminológiával: Production Stage

fejlesztés, a tesztelés, és a tanúsítás garantálja, hogy a használatba vétel megkezdődhessen a már meglévő vagy új hardver eszközökön" [70, 23. o.] Természetesen ez a kijelentés megállja a helyét, ugyanakkor szoftverek esetében a támogató rendszerek megfelelő érettségi szintje is szükséges a következő szakaszok sikeres megvalósításához. Szoftvertechnológiai kontextusban a fejlesztést követő szakaszok klasszikusan az üzemeltetési és a támogatási tevékenységekben manifesztálódnak. A szoftverek esetében is igaz az, hogy a fejlesztési szakasz feladata elérni a korábbi szakaszokban meghatározott követelményeknek való megfelelést. Azonban ezt követően az előállítási- és a *felhasználási szakasz*[53] értelmezése más megközelítést igényel. Célszerűbb ezek együttes tárgyalása a későbbiekben, mert ezek jelentős átfedéssel rendelkeznek a szoftverek életciklusában, ugyanis mindkét szakasz a rendszer első verziójának használatba vétele után következik, ugyanakkor a felmerülő változtatási igények kezelése, a fejlesztések végrehajtása az előállítási és felhasználási szakaszokat egyaránt érinti. Az említett tevékenységek elvégzéséhez szükséges a megfelelő színvonalú támogatási képesség, így kimondható, hogy a szoftverfejlesztés vonatkozásában az előállítási és a támogatási képességeknek folyamatosan rendelkezésre kell állniuk a felhasználási szakasz időtartama alatt. Ezek együttes megléte jelenti azt, hogy az adott szoftver alapú katonai képesség érettségi szintje megfelelő a teljes életciklus alatt – azaz így garantálható a NATO SLCM irányelv elsődleges célja.

A fejlesztési szakasz során nemcsak az adott katonai képességet jelentő szoftvert kell előállítani, hanem el kell végezni a szükséges tesztelési és integrációs feladatokat is, amelyek alapján biztosítható a szoftver érettsége a használatba vételhez. Ezen túl létre kell jönnie a kellőképpen részletes dokumentációnak is. A megfelelő dokumentáltság kérdésköre különösen érdekes az agilis szoftverfejlesztés vonatkozásában, ahol a cél a lényegre törő dokumentációs anyag előállítása. A kialakítandó katonai célú dokumentációs eljárásoknál ezt az elvárást is figyelembe kell venni. A szakasz végrehajtása során meg kell fontolni az érintettek által megfogalmazott követelményeket, azok változtatása kizárólag teljes körű egyetértés mellett lehetséges. *„A kezdeti koncepció finomítására a kialakítandó rendszer vonatkozásában lehetőséget kell biztosítani egészen a fejlesztési szakasz befejezéséig"* [70, 23. o.] – ezt a követelményt hagyományos vízesés modell jellegű fejlesztési módszerek esetén nehézkes kielégíteni. A változtatások kezelése iránti képesség igénye megfogalmazódik az AAP-20 dokumentumban is, a kialakítandó technikáknak választ kell adniuk erre a

[53] Felhasználási szakasz – NATO terminológiával: Utilization Stage

kérdésre is, emellett támogatniuk kell a döntéshozási mechanizmusokat, hogy az érdekeltek kellő mennyiségű és minőségű információval rendelkezzenek az elengedhetetlen közös álláspont kialakításához. Ezzel újabb nagyon fontos speciális folyamat szintű követelményt sikerült azonosítani: *a kialakítandó technikáknak támogatniuk kell a döntéshozási mechanizmusokat, hogy az érdekeltek kellő mennyiségű és minőségű információval rendelkezzenek az elengedhetetlen közös álláspont kialakításához*. A 4. táblázatban szereplők szerint csoportosítva jelennek meg a szakasz során végrehajtandó tevékenységek és az előállított dokumentumok.

Szoftverfejlesztéshez kapcsolható tevékenységek	**Előállított dokumentumok**
Tervezési és fejlesztési és tesztelési feladatok (Dev) 1. A kialakítandó rendszer architektúrájának létrehozása: hardver, szoftver komponensek. Szükséges emberi erőforrások és rendszer működéséhez szükséges belső és külső felületek. 2. A rendszer helyes és hibamentes működésének ellenőrzése 3. Meg kell bizonyosodni arról, hogy a támogató termékek az előállításhoz azonosításra kerültek és rendelkezésre állnak.	Tervezési és fejlesztési és tesztelési feladatok kapcsán létrejövő dokumentáció 1. Megfelelőség- és helyesség vizsgálati dokumentáció (tervek, eljárások, stb.) 2. Megfelelőségi- és helyesség vizsgálati eredmények (összefoglaló jelentések) 3. Szoftvertervezési dokumentáció (architektúra, tervezési dokumentáció) 4. Interfész specifikáció 5. Szoftver/Hardver integrációs tervek és specifikáció
Követelményelemzés, specifikáció (RA, Ops[54]) 1. Rendszerrel szemben támasztott követelmények kiértékelése és finomítása 2. A kialakítandó rendszerrel szemben támasztott követelmények megfelelősségének ellenőrzése és az éles üzembe helyezés feltételeinek ellenőrzése, működőképes, támogatható, visszavonható, költséghatékony. 3. Előállítási szakasz számára szükséges erőforrások meghatározása 4. Támogató rendszerekkel szemben támasztott követelmények meghatározása 5. Karbantartási stratégia kialakítása	Követelményekre vonatkozó dokumentáció 1. Rendszer meghatározás a szükséges tartalmi elemekkel 2. Előállítási (felhasználási) tervek 3. Üzemeltetési tervek 4. Üzemeltetési dokumentáció 5. Karbantartási stratégia/terv 6. Támogatási és karbantartási eljárások 7. Támogató rendszerek és szolgáltatások meghatározása a fejlesztést követő életciklus szakaszra vonatkozóan

[54] Ops – üzemeltetés, az angol **Op**eration**s** szó betűiből képzett rövidítés.

Program-, projektmenedzsment (PM, Ops) 1. Konfiguráció menedzsment terv frissítése 2. Költségvetés, ütemezés és élet ciklus költségekre vonatkozó becslések 3. Programra vonatkozó egyetértési nyilatkozat vázlatának kidolgozása 4. Releváns adatok archiválása	Program- és projektmenedzsmentre vonatkozó dokumentáció 1. Frissített konfiguráció menedzsment terv 2. Frissített költségtervek 3. Programra vonatkozó egyetértési nyilatkozat
Kockázatelemzés (QA) 1. Kockázatok azonosítása és kezelési tevékenységek meghatározása	Kockázatelemzési dokumentáció 1. Frissített kockázat meghatározás, értékelés és kezelési tervek.
Szoftverfejlesztést kevésbé érintő tevékenységek	**Előállított dokumentumok**
1. Leszerelési koncepció kialakítás 2. Integrált logisztikai támogatási terv frissítése 3. Elévülés menedzsment terv frissítése 4. A kialakítandó rendszer architektúrájának létrehozása: hardver komponensek. Szükséges emberi erőforrások és rendszer működéséhez szükséges belső és külső felületek. 5. Tanulságok összegzése (PM)	1. Leszerelési megfontolások 2. Frissített integrált logisztikai terv 3. Frissített elévülés menedzselési stratégia/terv. 4. Hardvertervezési dokumentáció (ábrák, modellek) 5. Tanulságok

4. táblázat A fejlesztési szakasz tevékenységeinek és dokumentumainak kapcsolata (saját szerkesztés)

A kialakítandó szoftverfejlesztési módszertannak figyelembe kell vennie az 4. táblázat szoftverfejlesztéshez köthető tevékenységei során előállított dokumentumok tartalmi elemeit. Ezt elvárásként megfogalmazva: *A NATO SLCM fejlesztési szakasz során keletkező szoftverfejlesztéshez köthető dokumentumok előállítását dokumentum sablonok szintjén támogassa a kialakítandó szoftverfejlesztési módszertan.*

Fizikailag létrejövő egységek esetében a fejlesztési szakasz lezáráshoz a kívánalmaknak megfelelő prototípus előállítása szükséges. Az említett prototípus a szoftverek esetében a szoftver első verziója, amely rendelkezik azokkal a jellemzőkkel, amelyek már lehetővé teszik a szoftver próbaüzembe helyezését az éles vagy egy az éleshez hasonló környezetben. A próbaüzem során feltárt hibák javítását követően kezdődhet meg az éles üzem.

3.1.6 Fejlesztést követő szakaszok

Korábban az 5. ábrán láthattuk már, hogy a fejlesztést követően az előállítás, a felhasználás és a támogatás szakaszai következnek a NATO életciklus modelljében. Mindhárom szakasz értelmezése során feltételezhetjük, hogy az adott projekt célja

egy szoftver alapú szolgáltatás kialakítása – tehát fejlesztés alatt álló SOI egy szoftvert reprezentál.

Az *előállítási szakasz*[55] célja alap esetben a fejlesztési szakasz során előállított SOI gyártásának megkezdése, az előállított eszközök tesztelése, illetve a támogató rendszerek és szolgáltatások kialakítása. Szoftverek esetében a fejlesztői környezetben működő szoftver megfelelő működését az éles felhasználási környezetben bizonyos ellenőrzések elvégzése mellett lehet csak garantálni. Kizárólag szoftver alapú szolgáltatások esetében ezért az előállítási szakasz helyett pontosabb az *üzembe helyezés szakasza* kifejezés használata. A továbbiakban SaaS típusú szolgáltatásokkal kapcsolatos szövegkörnyezetben az üzembe helyezés szakasza kifejezés alatt a NATO SLCM előállítási szakasza értendő.

A *felhasználási szakasz*[56] kezdete a kialakított katonai képesség használatba vétele, ezt követően az esetleges módosítások, frissítések is ebbe az életciklus szakaszba esnek, egészen az adott SOI leszereléséig.

Szoftvertechnológiai tekintetben ezek a szakaszok erősen összeolvadnak, a fejlesztési szakasz záróakkordjaként előállhat a kívánt katonai képességet nyújtó szoftver, ugyanakkor annak gyakorlati tapasztalatait célszerű egy tesztüzemszerű működéssel összegyűjteni, ahol már szükség van támogató képességre is, így gyakorlatilag a *támogatási szakasz*[57] is kezdetét veszi egy időben az előállítási szakasszal. Bármilyen módosítás a szoftver funkcionalitásával kapcsolatosan egyaránt érinti az előállítási és a felhasználási szakaszokat, előbbi szoftverfejlesztési tevékenységet von maga után, utóbbi túlnyomórészt üzemeltetési tevékenységet.

Az előállítás szakaszban elvégzendő tevékenységeket nehézkes szoftverfejlesztésre interpretálni, ugyanis egy fizikailag létrehozandó termék esetén a fejlesztési szakaszban előállt dokumentumok alapján megkezdődhet a gyártás, míg a szoftverek esetében azokat a szoftverfejlesztési módszereket kell képesnek lenni alkalmazni, amelyek az fejlesztési szakaszban felhasználásra kerültek és segítségükkel sikerült előállítani az adott szoftver tesztüzemre alkalmas verzióját. Természetesen ebben a szakaszban is van értelme a kockázatelemzésnek, az egyes funkciók fejlesztési komplexitásának felmérésével, a változtatási igények hatásainak vizsgálatával. Az előállítási szakasz végére teljesülnie kell annak a követelmények, hogy az adott szoftvernek és

[55] Előállítási szakasz – NATO terminológiával Production Stage
[56] Felhasználási szakasz – NATO terminológiával Utilisation Stage
[57] Támogatási szakasz – NATO terminológiával Support Stage

a vele egyidejűleg, párhuzamosan kialakított SOI-knak kompatibiliseknek kell lenniük egymással. Szoftverek esetében ez egy külön kockázati tényező, mert minden változtatás után ez a tulajdonság újra ellenőrizendő.

Ahogy az előállítási szakasz szoftvertechnológiai magyarázatra szolgál, nincs ezzel máshogy a felhasználási szakasz sem. Az előállított szoftver felhasználásának megkezdése attól az időponttól számítható, amikor a szoftver alkalmazója vagy a szoftver alapú szolgáltatás üzemeltetője válik felelőssé a rendszer működtetéséért az erre a célra kialakított üzemeltetési környezetben. Ezt követően a rendszer teljesítménye, a hibákra utaló jelenségek, a helytelen működés detektálása folyamatos rendszerfelügyeletet kíván meg.

A rendszerhibák kijavítása megvalósulhat karbantartási műveletek segítségével, illetve kisebb-nagyobb módosításokkal. Szóba jöhetnek áthidaló és átalakítást igénylő megoldások is. Szoftverek esetében áthidaló megoldás lehet a paraméterezés megváltoztatása vagy funkciók átmeneti letiltása, ha van megkerülő megoldás az adott rendszeren belül.

Az átalakítást igénylő megoldás az adott hiba javítását, új szoftver verzió előállítását, ezt követően a kiadott hibajavító verzió üzembe helyezését jelenti. Ebben az életciklus szakaszban alakulhatnak ki az adott rendszer különböző éles üzemű konfigurációi, amelyeket célszerű a konfiguráció menedzsment tervben rögzíteni, szoftverek esetében erre a célra támogató rendszerek bevezetése is lehetséges.

Nem véletlen, hogy a támogatási szakasz időben erős átfedéssel rendelkezik a felhasználási szakasszal. A támogatási szakasz karbantartási, logisztikai és más SOI-k felé nyújtott támogatási feladatok ellátásával kezdődik meg. Ez a tevékenység magában foglalja a szolgáltatások és a támogató szolgáltatások teljesítményének felügyeletét is.

A rendszerek működésében mutatkozó anomáliák és hibák azonosítása, rögzítése, besorolása és javítása egyaránt a támogatási szakasz feladatrendszeréhez tartozik. A támogatási szakasz a NATO SLCM fogalomrendszerében inkább technikai támogatási feladatokat takar, mintsem üzleti tanácsadást.

A következő táblázatokban az előállítási (üzembe helyezési), a felhasználási és a támogatási szakaszok során végzett tevékenységek és az előállított dokumentumok, termékek láthatók a feladatokat ellátó szereplők szerint csoportosítva. A táblázatok célja a kutatás során elkészítendő dokumentációs sablonokkal szemben támasztott követelményrendszer megalapozása.

Szoftverfejlesztéshez kapcsolható tevékenységek	**Előállított dokumentumok**
Felhasználási szakasz előkészítése (Dev, Ops) 1. Az átvételi tesztek elvégzése. 2. A szükséges szabványosítás megfontolása. 3. Gondoskodni kell a fenntartható felhasználás és támogatás feltételeiről.	Felhasználási szakasz előkészítése során keletkező dokumentáció / termékek 1. Előállított SOI 2. Rendelkezések a fenntartható felhasználási és támogatási szakaszok számára
Minőségbiztosítás (QA) 1. A gyártási folyamatok felügyelete és irányítása (technikai, minőségi és hatékonyság)	Minőségbiztosítással kapcsolatos dokumentáció -
Program-, projektmenedzsment (PM, Ops) 1. Konfiguráció menedzsment terv frissítése 2. A nem anyagi jellegű DOTMLPFI-k kialakítása, illetve azok módosítása. 3. Költségvetés, ütemezés és élet ciklus költségekre vonatkozó becslések. 4. Az előállítási szakaszra vonatkozó egyetértési nyilatkozat frissítése. 5. Programra vonatkozó egyetértési nyilatkozat vázlatának kidolgozása. A felhasználási / támogatási szakaszra vonatkozó egyetértési nyilatkozatok kialakítása. 6. Releváns adatok archiválása.	Program- és projektmenedzsmentre vonatkozó dokumentáció 1. Frissített konfiguráció menedzsment terv 2. Minden nem anyagi jellegű DOTMLPFI frissített verziója 3. Minden terv és rendelkezés kialakításra került a támogatási szakasz számára 4. Frissített költségtervek
Szoftverfejlesztést kevésbé érintő tevékenységek	**Előállított dokumentumok**
1. A rendszer számára szükséges alkotóelemek (katonai felszerelés) előállítása. 2. A rendszer számára szükséges alkotóelemek kialakítása és integrálása a rendszerbe a felhasználási szakasz számára. 3. Az előállítási szakaszban kell kialakítani a fenntartható felhasználás és a kivezetés feltételeit is. 4. Integrált logisztikai támogatási terv frissítése. 5. Elévülés menedzsment terv frissítése. 6. Leszerelési koncepció frissítése. 7. Tanulságok összegzése. (PM)	1. Frissített leszerelési koncepció 2. Frissített integrált logisztikai terv 3. Frissített elévülés menedzselési stratégia/terv

5. táblázat Az előállítási szakasz tevékenységei és dokumentumai (saját szerkesztés)

Az 5. táblázat alapján megállapítható, hogy míg az előállítási szakaszhoz köthető tevékenységek jól definiáltak, úgy az előállított dokumentumok – különösen a szoftvertechnológia vonatkozásban – hiányosnak tűnnek. Így kutatási célként tekinthető az üzembe helyezés szakaszát támogató dokumentumsablonok kialakítása.

Szoftverfejlesztéshez kapcsolható tevékenységek	**Előállított dokumentumok**
Üzemeltetési feladatok (Ops, Dev) 1. A rendszer aktiválása a kialakított üzemeltetési környezetben 2. Az üzemeltetési tervben foglaltak ellenőrzése az üzemeltetés során. 3. Hibaazonosítási eljárások végrehajtása, ha nem megfelelő működés lépett fel a szolgáltatásban. 4. Hibajavítás tevékenység meghatározása, ha lehetséges 5. Tervezési korrekciók kérése, ha lehetséges	Felhasználási szakasz előkészítése során keletkező dokumentáció / termékek 1. Megvalósított képesség
Minőségbiztosítás (QA, Ops) 1. A rendszer működésének ellenőrzése adatok gyűjtésével, a hatékonyság megfelelő. A rendszer megbízható, karbantartható, elérhető. 2. Felhasználói visszajelzések kérése	Minőségbiztosítással kapcsolatos dokumentáció 1. Hiba- és élettartam adatokra vonatkozó dokumentum
Program-, projektmenedzsment (PM, Ops) 1. A támogató termékek beszerzése és a kapcsolódó szolgáltatások kialakítása 2. Képzett és képesített üzemeltetők kijelölése 3. A műszaki változtatások áttekintése és megvalósítása szakaszokra bontott AAP-20 szerinti megközelítéssel történjen. 4. Élettartam növelést előirányzó megfontolások gyűjtése 5. Költségvetés, ütemezés és élet ciklus költségekre vonatkozó becslések.	Program- és projektmenedzsmentre vonatkozó dokumentáció 1. A SOI leszerelésére vonatkozó döntés 2. Leszerelés elfogadása 3. Költségvetés, ütemezés és élet ciklus költségekre vonatkozó becslések
Szoftverfejlesztést kevésbé érintő tevékenységek	**Előállított dokumentumok**
1. Munkahelyi biztonság, környezetvédelmi szabályozás és nemzetközi emberi jogoknak való megfelelősség ellenőrzése a katonai művelet végrehajtása során. 2. Tanulságok összegzése. (PM)	1. Tanulságok összegzése. (PM)

6. táblázat A felhasználási szakasz tevékenységei és dokumentumai (saját szerkesztés)

A felhasználási szakasz vonatkozásában elmondható, hogy az előállítandó dokumentumok mennyisége jelentősen kevesebb, mint az a korábbi szakaszok során volt tapasztalható. Ez megnehezíti a szoftvertechnológiai követelmények azonosítását, ugyanakkor az azonosított Dev és Ops tevékenységek során keletkező dokumentumok beépítése a kialakítandó dokumentációs sablonokba elvárás.

Szoftverfejlesztéshez kapcsolható tevékenységek	**Előállított dokumentumok**
Üzemeltetési feladatok (Ops, Dev) 1. A karbantartási stratégia/terv megvalósítása 2. A támogató rendszerek, rendszerelemek és szolgáltatások rendelkezésre állásának biztosítása a rendszer karbantartása alatt 3. A rendszer képességeinek felügyelete a szolgáltatás biztosításához és hibák rögzítése a későbbi elemzés céljából. 4. Korrekciót, alkalmazkodást, tökéletesítés és megelőzést támogató intézkedések. A helyreállított képességeket megerősítése.	Felhasználási szakasz előkészítése során keletkező dokumentáció / termékek 1. Karbantartási/támogatási adatok (frissített hiba- és élettartam adatokra vonatkozó dokumentum) 2. Tanulságok
Minőségbiztosítás (QA, PM) 1. Az üzemeltetést és karbantartás végző személyzet és más projektek, amelyek készítenek vagy használnak hasonló rendszer elemeket. számára a hibajelentések, korrekciós tevékenységek, trendekről nyújtott információk.	Minőségbiztosítással kapcsolatos dokumentáció -
Szoftverfejlesztést kevésbé érintő tevékenységek	**Előállított dokumentumok**
1. Integrált logisztikai támogatási terv megvalósítása. 2. A kölcsönös logisztikai támogatás lehetőségeinek feltárása. 3. Fogyó eszközök pótlása. 4. Az elévülés menedzsment elvégzése. 5. Tanulságok összegzése. (PM)	1. A SOI leszerelésére vonatkozó döntés 2. Leszerelés elfogadása.

7. táblázat A támogatási szakasz tevékenységei és dokumentumai (saját szerkesztés)

A fenti ábra alapján látható, hogy a támogatási szakasz esetében sincs különösen jelentős dokumentációs elvárás a NATO SLCM oldaláról, a szakasz során elvégzendő támogató tevékenységek bemutatása megfelelő.

Az alapvető szakaszok feltárását követően, elmondható, hogy a NATO életciklus modell különösen átfogó támogatást ad tetszőleges típusú projektek indításához, ugyanakkor a SOI-k sokrétűségéből kifolyólag kevesebb iránymutatással szolgál az előállítási szakasztól kezdődően, ennek okán a három együtt tárgyalt szakasz esetében a következő közös dokumentációs elvárás fogalmazható meg. *A kialakítandó szoftverfejlesztési módszertan részeként kerüljenek meghatározásra azok a dokumentációs sablonok, amelyek megfelelően támogatják a NATO SLCM üzembe helyezési, a felhasználási és a támogatási szakaszok végrehajtását.*

A soron következő életciklus szakasz tárgyalása előtt feltétlenül meg kell említeni a felhasználási szakasz projekt menedzsment feladatai között szereplő egyik lényeges követelményt: *„a műszaki változtatások áttekintése és megvalósítása szakaszokra bontott AAP-20 szerinti megközelítéssel történjen”*. [70, 29. old] Ebben az esetben nem az eddig bemutatott egymást követő szakaszok az irányadók, hanem a *„Módosítási és frissítési eljárás a felhasználási és támogatási szakaszokban”* című 3. sz. melléklet. [73]

3.1.7 Módosítási és frissítési eljárás

A módosítási és frissítési eljárás rövid áttekintése elengedhetetlen a NATO által támogatott iteratív életciklus folyamat megértéséhez, amely nagymértékű hasonlóságot mutat az agilis szoftverfejlesztési módszertanok alkalmazása során megvalósított ismétlődő iterációkkal, így további folyamat szintű követelmények feltárása várható az elemzéstől.

A módosítási és a frissítési eljárások a szoftvertechnológiában szabályozott körülmények között verzióváltást jelentenek a korábban már éles üzemben futó rendszerhez képest. Ez alól valamilyen konfigurációs beállítás változtatása jelenthet csak kivételt, de ez az eset nem érinti az adott szoftver futtatható állományainak módosítását, ezért ezt az ágat nem szükséges a továbbiakban vizsgálni. Az elemzés célja az AAP-20 kiadványban szereplő lépések azonosítása egy adott SOI módosítására vonatkozóan, azonban a dokumentum kiemeli, hogy a folyamat testre szabható és az adott képesség javításához kell alárendelni azt. Ezzel a lehetőséggel élni is szeretnék, a lépések azonosítása után megfogalmazhatók a kutatási célként kitűzött szoftverfejlesztési módszerrel szemben támasztandó folyamat szintű követelmények.

A folyamat alapját az képezi, hogy a felhasználási és a támogatási szakaszokban – amelyek a szoftverek esetében erősen átfedik egymást – felmerülhetnek az adott katonai képességgel szemben módosítási, illetve frissítési igények. Igény alapjául szolgálhat egy képességbeli hiányosság pótlása vagy egy feltárt hibajelenség javítása. Abban az esetben, ha ezek megalapozottnak bizonyulnak – azaz a képesség javulását eredményezik – megkezdődhet a módosítási eljárás. Az eljárás az alap folyamathoz hasonlóan koncepcionális előtervezési és koncepcióalkotási szakaszokkal kezdődik. A tervezési szakaszokat követően, analóg módon az alapértelmezett SLCM folyamatban foglaltakkal megkezdődhet a továbbfejlesztés, később az előállítás. Ez a folyamat egyaránt értelmezhető fizikailag létrejövő termékekre és szoftverekre is,

mindkét esetben el kell végezni a kapcsolódó konfiguráció menedzsmentet érintő feladatokat is.

Ugyanakkor a kapcsolódó SOI-k, támogató rendszerek vonatkozásában hatalmas eltérésekkel szembesülhetünk, ezt a problémakört a dokumentum általánosan tárgyalja. Nem jelennek meg a módosítási eljárás során az előállítandó dokumentumok szakaszonkénti bontásban. Az egyes szakaszok bemutatása során az elvégzendő tevékenységek általánosan kerülnek tárgyalásra, különösebb keretek nélkül. Meglátásom szerint ennek oka, hogy a módosítandó rendszer ezen a ponton már egy használatba vett, fizikailag és/vagy virtuálisan létező rendszer, amely esetében nehézkes direkt követelményeket megfogalmazni.

A módosítási eljárás szakaszaira vonatkozó iránymutatásokban megfogalmazott elérendő célok azonban hasznosíthatók a kutatásom céljaként kitűzött szoftverfejlesztési módszer kialakítása során. Minden szakasz az alap folyamathoz hasonló környezetben kerül végrehajtásra, melyek újbóli bemutatása meghaladja a kutatás kereteit, kizárólag szoftvertechnológiai szemszögéből lényeges tevékenységek kerülnek bemutatásra és tárgyalásra.

A koncepcionális előtervezési szakasz során kell meghatározni módosítandó katonai képességgel és a támogató rendszerekkel szemben támasztott továbbfejlesztési igényeket. A módosítási és frissítési eljárás e szakaszában már az igények priorizálásával is találkozhatunk [73, 4.1] – analógiát mutatva az agilis szoftverfejlesztéssel. Ezen a ponton az alapfolyamathoz hasonlóan a megfelelő konszenzus kialakítása a cél az érintett felek részéről, amelyet a priorizált követelménylistának is tükröznie kell.

Szoftverfejlesztési kontextusba helyezve a szakasz során elvégzendő feladatokat, a következő tevékenységeket kapjuk: duplikált igények azonosítása, irreális követelmények kiszűrése, lehetséges technológia megoldások feltárása. Ezt követően további feladat a valószerű követelmények azonosítása és finomítása, valamint a kiválasztott technológiákkal kapcsolatos különböző kockázatelemzési, kockázatkezelési feladatok elvégzése, hatástanulmányok készítése. Így újabb folyamat szintű követelmény határozható meg a bemutatott tevékenységek elvégzéséhez dokumentációs és technológiai oldalon. *A kialakítandó szoftverfejlesztési módszer támogassa a módosítási eljárás koncepció alkotási szakaszában elvégzendő szoftvertechnológiai mozzanatokat: duplikált igények felismerése, nem teljesíthető követelmények azonosítása, kockázatkezelés.*

Miután megtörtént a módosítási igények pontosítása, megkezdődhet a módosítási és frissítési eljárás koncepció alkotási szakasza, amely az alapfolyamathoz hasonlóan a már ismert két fázist tartalmazza. A felmérési fázisban további megvalósíthatósági tanulmányok elkészítése történik, a módosításhoz szükséges erőforrások felmérésével, a végrehajtási idő meghatározásával. Minden egyes változtatási igény szoftverfejlesztési oldalról is megköveteli a fázis végrehajtását. A becslések pontosságához különböző tényezőket kell figyelembe venni, a következő két követelmény azokat a módosítási és frissítési eljárásban meghatározott tényezőket taglalja, amelyek szoftvertechnológiai környezetben befolyásolhatják az adott módosítási igény erőforrásigényét. *A kialakítandó szoftverfejlesztési módszer támogassa a technológiai lehetőségek kiértékelését, a konfiguráció és rendszer kapcsolatának vizsgálatát, a módosítás hatásának vizsgálatát a rendszer komplexitására, az átvételi követelmények finomítását, a költségbecslést, a megkötések azonosítását, a kockázatelemzést, a megkerülő megoldások feltérképezését, a további érettséget előirányzó módosítások meghatározását.*

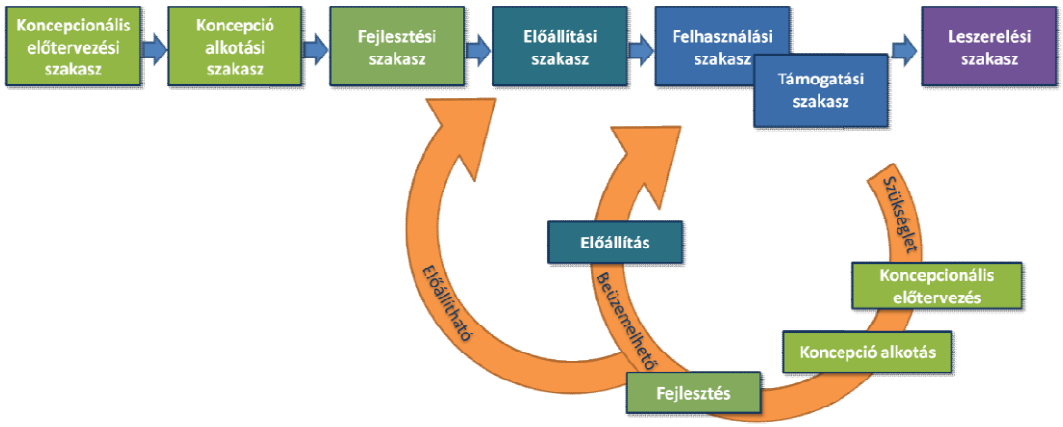
Általános esetben a módosítási eljárás során végrehajtott programindítási fázis feladata a fejlesztési és gyártási szakaszok számára a nyitott kérdések tisztázása, a koncepcionális tervezés lezárása. A fázis során az optimális technológiai megoldás meghatározása a cél. Kutatási követelményként ez a következőképp fest. *A kialakítandó szoftverfejlesztési módszer támogassa az előzetes tervezést és elemzést, a prototípuskészítést, a bemutatókat, a laboratóriumi teszteket, az optimális technológiák meghatározását a hatékonyság, a költségek, az idő és a kockázatok vonatkozásában, megkerülő megoldások meghatározását, a követelményrendszer véglegesítését.*

Szoftverfejlesztési projektek esetében is meg lehet találni a mérföldkövet a két fázis között, az egyik az architekturális megfontolásokat takarja, a másik a teljes megvalósítási elgondolást. A fejlesztési szakasz a módosítási eljárás során a korábban kialakított koncepció alapján valósul meg, ugyan még van lehetőség tervezési döntések meghozatalára, a követelmények és specifikáció finomítására. A módosítási eljárás fejlesztési szakaszára vonatkozó kutatási követelmény a következőképp fogalmazható meg. *A kialakítandó szoftverfejlesztési módszer támogassa a fejlesztést, a tesztelést, az értékelést, a helyesség és a megfelelés ellenőrzését, illetve a követelmények és a specifikáció finomítását, valamint a rendszer verziók korlátlan előállítását.*

A teszt üzem megkezdése a szoftverfejlesztési projektekben az előállítási szakasz belépési pontjának tekinthető, mert ekkor ér véget a módosítások fejlesztése és a

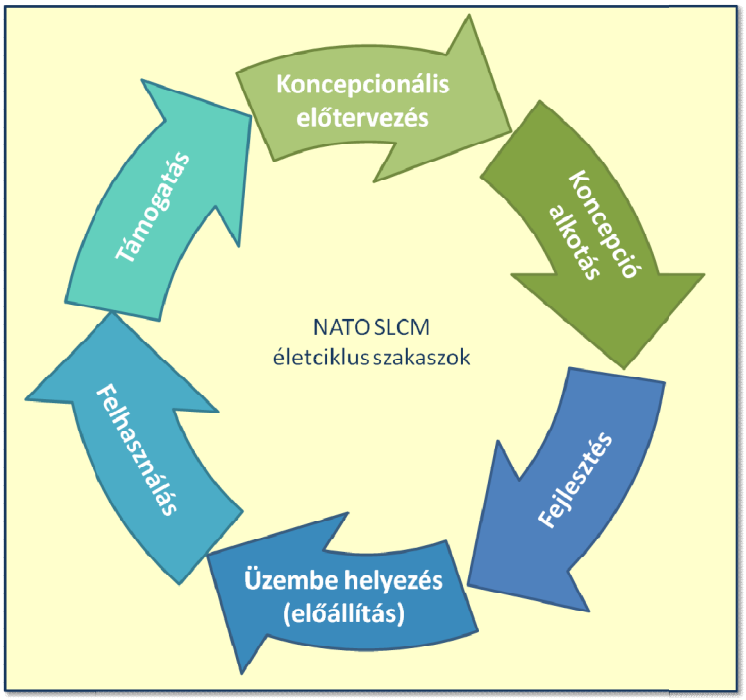
megrendelő itt találkozik a termékkel – szoftverrel – először. Ebben a szakaszban valósul meg az ügyfél által végzett átvételi teszt, amelynek sikeres kimenetele esetén a módosítások éles üzembe helyezhetők és elérhetővé válnak a nagyközönség számára is. A módosítási eljárás előállítási szakaszával lezárul a rendszer módosítására irányuló iterációs folyamat, kutatási követelményként a következő elvárás fogalmazható meg a szakaszban. *A kialakítandó szoftverfejlesztési módszer támogassa a támogató dokumentációk előállítását, a konfigurációkban bekövetkező változtatások követését, az üzemeltetési környezet és támogató rendszerek felkészítését és az átvételi teszt elvégzését.*

Az éles üzembe helyezést követően a frissített dokumentumok alapján kerülnek meghatározásra a felhasználási és üzemeltetési követelmények. Az SLCM folyamata ezt követően az alapfolyamat medrében folytatódik, természetesen indokolt változtatási igény esetében a módosítási és frissítési eljárás bármikor végrehajtható, a folyamat a 9. ábrán látható.



9. ábra Módosítási és frissítési eljárás, forrás: [73, Figure 19]

Szoftverek esetében is megállja a helyét ez a technika, ugyanakkor véleményem szerint célszerű abból kiindulni, hogy a követelmények és a megvalósítást végző funkciók folyamatos változásokon mennek keresztül, ebből a megfontolásból az alapfolyamat szakaszai és a módosítási és frissítési eljárás szakaszai összefésülhetők és az alapfolyamat egyfajta 0. iterációnak tekinthető ekkor. Ez alapján újabb követelmény fogalmazható meg a kialakítandó szoftverfejlesztési módszerrel szemben, miszerint *a kialakítandó folyamatok a koncepcionális előtervezés szakaszával kezdve korlátlanul legyenek megismételhetők.* A 10. ábrán látható a szoftverfejlesztési projektekre értelmezhető, kezdetektől iterációkat előrevetítő életciklus modell.



10. ábra Szoftverekre értelmezett NATO SLCM-nek megfelelő iteratív életciklus (saját szerkesztés)

3.1.8 Leszerelési szakasz[58]

A felhasználást követően az adott katonai képességet jelentő rendszer vagy rendszerek leszerelése, illetve kivezetése a leszerelési szakaszban valósul meg. A leszerelési szakasz nem csak az adott rendszer, hanem a támogató rendszerek felülvizsgálatát és indokolt esetben a megszüntetésüket is jelenti, a leszerelés menetére vonatkozó eljárásoknak az előző szakaszok során már dokumentálásra kell kerülniük a fejlesztési, gyártási, bevezetési tapasztalatok alapján. A leszerelés szakaszában szem előtt kell tartani a biztonsági, védelmi és környezeti szempontokat is. A leszerelési szakasz 4 különböző eredménnyel zárulhat. [70, 33. old]

- *„Redundáns programok konszolidációja.*
- *Működési és karbantartási költségek csökkenése.*
- *A leszerelésből származó haszon maximalizálása.*
- *Felhasználható alkatrészek kinyerése a kivont SOI-ból.”*

Úgy gondolom, hogy a leszerelés esetében a fizikai eszközöktől jelentősen eltérve a szoftverek, illetve szoftver alapú szolgáltatások esetében öt különböző esettel találkozhatunk, amelyeket célszerű megvizsgálni. A leszerelésből származó haszon maximalizálása is lehetséges, ha az újrahasznosítás lehetősége valamilyen formában adott. A következő felsorolás mutatja be a lehetséges kimeneteket.

[58] Leszerelési szakasz – NATO terminológiával Retirement Stage

- Valódi kivezetés
- Szoftver kiváltása másik szoftverrel
    - adatmigrálással
    - adatmigrálás nélkül
- Szoftver funkcionalitásának megvalósítása más rendszerben
    - adatmigrálással
    - adatmigrálás nélkül

Az AAP-20 kiadvány a valódi kivezetés kérdéskörét tárgyalja alaposan, két fázis során történik meg a leszerelés, először a felszámolási fázisban[59] a SOI és a támogató rendszerek által nyújtott szolgáltatások leállítása történik meg. A leszerelés második lépése a likvidációs fázis, amely során az adott SOI és a támogató rendszerek megsemmisítésre kerülnek – amennyiben lehetőség van rá, bizonyos elemeik újrahasznosítása mellett.

Informatikai rendszerek esetében utóbbi gyakori, ugyanis amennyiben az adott rendszer funkcionalitásának egésze vagy annak egy része valamilyen más formában egy másik rendszerben kialakításra kerül, akkor célszerű lehet átmenteni a leszerelt rendszer virtuális tartozékait:

- tárolt adatok
- funkció leírások
- üzemeltetési tapasztalatok

Újabb kutatási követelményhez jutottunk, miszerint:

A *kialakítandó szoftverfejlesztési módszertan támogassa a tárolt adatok migrálhatóságát, a funkciók követelményeinek kinyerését, illetve az üzemeltetési tapasztalatok kinyerését egy szoftver alapú szolgáltatás megszűntetésekor.*

A leszerelési szakaszban bemutatott tevékenységek részben vagy teljes egészében fizikailag létező katonai képességek leszerelését mutatják be, a szakasz során keletkező dokumentációkkal együtt. A leszerelési szakaszban keletkező dokumentumok és tevékenységek összefüggéseinek további tárgyalása már nem indokolt, mert azok alapvetően fizikailag létező képességek megszűntetését és az eljárások során elkészítendő dokumentumokat mutatják be.

Az AAP-20 kiadvány elemzése során számos folyamat szintű követelményt sikerült azonosítani a kialakítandó szoftverfejlesztési módszertannal szemben, ilyenek az

[59] Felszámolási fázis – NATO terminológiával Disengagement Phase

életciklus szakaszokon belül elvégzendő tevékenységek és az elkészítendő dokumentumok. A tanulmányozás alatt, a módosítási és frissítési eljárás mozzanatainak feltárásakor számos párhuzamot sikerült feltárni az agilis szoftverfejlesztési módszertanokkal, ezzel előrevetítve azok kiterjesztési lehetőségét katonai alkalmazási célra. Az alábbiakban az alfejezet során feltárt követelmények következnek.

Módszertani követelmények

M-8 A kialakítandó szoftverfejlesztési módszer értelmezze az érettségi szint fogalmát és feleljen meg a NATO Rendszer Koncepciónak. Lásd: 81. old.

M-9 A kialakítandó szoftverfejlesztési módszer adjon eszközöket a stratégiai célok dokumentálására, a szereplők azonosítására, a követelmények meghatározására és priorizálására, támogassa a változáskezelést, valamint folyamatosan adjon megfelelő visszajelzést a rendszer érettségével kapcsolatban. Lásd: 85. old.

M-10 Az alternatív megvalósíthatósági tanulmányok kialakítását folyamat és dokumentációs szinten is támogassa a kialakítandó szoftverfejlesztési módszertan. Lásd: 87. old.

M-11 A kutatás céljaként kitűzött kialakítandó szoftverfejlesztési módszer folyamataiban jelenjenek meg a felsorolásban szereplő tevékenységek: követelmények áttekintése, funkcionális áttekintés, tervezési áttekintés, tesztelésre alkalmasság ellenőrzése, konfigurációs felülvizsgálat, megfelelőségi ellenőrzés, helyességi ellenőrzés, éles üzembe helyezésre alkalmasság ellenőrzése. Lásd: 89. old.

M-12 A kialakítandó technikáknak támogatniuk kell a döntéshozási mechanizmusokat, hogy az érdekeltek kellő mennyiségű és minőségű információval rendelkezzenek az elengedhetetlen közös álláspont kialakításához. Lásd: 91. old.

M-13 A kialakítandó szoftverfejlesztési módszer támogassa a módosítási eljárás koncepció alkotási szakaszában elvégzendő szoftvertechnológiai mozzanatokat: duplikált igények felismerése, nem teljesíthető követelmények azonosítása, kockázatkezelés. Lásd: 99. old.

M-14 A kialakítandó szoftverfejlesztési módszer támogassa a technológiai lehetőségek kiértékelését, a konfiguráció és rendszer kapcsolatának vizsgálatát, a módosítás hatásának vizsgálatát a rendszer komplexitására, az átvételi követelmények finomítását, a költségbecslést, a megkötések azonosítását, a kockázatelemzést, a megkerülő megoldások biztosítását, a további érettséget előirányzó módosítások meghatározását. Lásd: 100. old.

M-15 A kialakítandó szoftverfejlesztési módszer támogassa az előzetes tervezést és elemzést, a prototípuskészítést, a bemutatókat, a laboratóriumi teszteket, az optimális technológiák meghatározását a hatékonyság, a költségek, az idő és a kockázatok vonatkozásában, megkerülő megoldások meghatározását, a követelményrendszer véglegesítését. Lásd: 100. old.

M-16 A kialakítandó szoftverfejlesztési módszer támogassa a fejlesztést, a tesztelést, az értékelést, a helyesség és a megfelelés ellenőrzését, illetve a követelmények és a specifikáció finomítását, valamint a rendszer verziók korlátlan előállítását. Lásd: 100. old.

M-17 A kialakítandó szoftverfejlesztési módszer támogassa a támogató dokumentációk előállítását, a konfigurációkban bekövetkező változtatások követését, az üzemeltetési környezet és támogató rendszerek felkészítését és az átvételi teszt elvégzését. Lásd: 101. old.

M-18 A kialakítandó folyamatok a koncepcionális előtervezés szakaszával kezdve korlátlanul legyenek megismételhetők. Lásd: 101. old.

M-19 A kialakítandó szoftverfejlesztési módszertan támogassa a tárolt adatok migrálhatóságát, a funkciók követelményeinek kinyerését, illetve az üzemeltetési tapasztalatok kinyerését egy szoftver alapú szolgáltatás megszűntetésekor. Lásd: 103. old.

Dokumentációs követelmények

D-1 A NATO SLCM koncepció alkotási szakasz során keletkező szoftverfejlesztéshez köthető dokumentumoknak helyt kell kapniuk a kialakítandó szoftverfejlesztési módszertan dokumentum sablonjaiban. Lásd: 89. old.

D-2 A NATO SLCM fejlesztési szakasz során keletkező szoftverfejlesztéshez köthető dokumentumok előállítását dokumentum sablonok szintjén támogassa a kialakítandó szoftverfejlesztési módszertan. Lásd: 92. old.

D-3 A kialakítandó szoftverfejlesztési módszertan részeként kerüljenek meghatározásra azok a dokumentációs sablonok, amelyek megfelelően támogatják a NATO SLCM üzembe helyezési, a felhasználási és a támogatási szakaszok végrehajtását. Lásd: 97. old.

3.2 NATO AAP-48 ÉLETCIKLUS FOLYAMATOK

A végrehajtott nemzetközi és NATO programok tapasztalatai alapján három szem előtt tartandó alapelvvel találkozhatunk. A kialakított alapelvek az általános célkitűzésektől a speciális támogató eszközökig adnak iránymutatást. [70, 18-19 old.]

1. *„Valamennyi érintett nemzet megállapodása szükséges a kialakítandó képesség vonatkozásában."* – A cél eléréséhez 3 különböző típusú dokumentum nyújt segítséget egy NATO program végrehajtása során.
    a. NATO SLCM irányelv – jelen kutatás szemszögéből a legfontosabb dokumentum ebben a kategóriában, feldolgozása megtörtént a korábbiakban, a katonai szoftverfejlesztésre értelmezhető követelmények azonosításra kerültek.
    b. Szövetségi Doktrínák[60]– az adott katonai képességekhez kapcsolódó általános elvek leírásai. Az egyes AJP-k elemzése meghaladja az értekezés kereteit, ezért további elemzésüktől eltekintek.
    c. NATO interoperabilitási irányelv [74] – a különböző NATO és nemzeti rendszerek közötti együttműködés kialakítását elősegítő irányelv. Tárgyalása meghaladja az értekezés kereteit, ugyanakkor a szoftvertechnológiai értelemben vett interoperabilitás kutatási követelményeként részét képezi az értekezésnek.
2. *„A közös és a nemzeti értékteremtés érdekében nemzeti szinten megvalósuló együttműködésre van szükség."* – Az elvárt együttműködés kialakításához az első szintnél kisebb hatókörrel rendelkező, ugyanakkor részletesebb iránymutatást adó dokumentumokkal találkozhatunk a második szinten, amelyek segítségével a NATO programok megfelelő dokumentációs, pénzügyi, adminisztratív, kommunikációs és titoktartási keretrendszerben hajthatók végre.
    a. NATO AAP-20 – az elsődleges dokumentum, amely támpontot ad a kívánt együttműködés kialakításához, felhasználása alapvető a kutatási hipotézisek igazolásához, feldolgozása a korábbiakban megtörtént.
    b. NATO AAP-48 – az értekezésben korábban még nem tárgyalt dokumentum – jelen alfejezet célja az AAP-48 kiadvány elemzése és a benne található szoftvertechnológiai követelmények kinyerése.

[60] NATO doktrína – Allied Joint Publication, röviden: AJP

c. A keretrendszer második szintjének további dokumentumai, amelyek elemzése nem képezi célját a kutatásnak, ugyanakkor közülük néhány dokumentum típus említése célszerű, a teljesség igénye nélkül: AAPP[61], AACP[62], ALCCP-1[63], illetve további NATO megállapodások. A felsorolt dokumentumok a beszerzések, a költségek, a tudásmegosztás, kockázatkezelés területeit fedik le, így ezek részletes feltárás nem szükséges.

3. *„Egyetértés és elfogadott elvárások az érintettek részéről a NATO programmal kapcsolatban."* – A NATO programok végrehajtását mikro szinten támogató dokumentumokat találhatjuk a keretrendszer harmadik szintjén.
   a. NATO AAP-20 – a követelmények meghatározásának, finomításának részletes folyamatait leíró dokumentum, amely így teszi lehetővé a sikeres végrehajtás előkészítését. A kapcsolódó szakaszok feltárása és tárgyalás megtörtént.
   b. NATO AQAP szabványcsalád – a különböző katonai képességfejlesztésekhez kapcsolódó minőségbiztosítási szabványok, amelyek elemzése meghaladja a kutatás lehetőségeit. Tekintettel arra, hogy az általam kidolgozandó technikák esetében a NATO SLCM folyamatokhoz történő illeszkedés követelmény, így a minőségbiztosítási szabványok esetükben is értelmezhetők, a megfelelés ellenőrizhető.
   c. a keretrendszer harmadik szintjének további dokumentumai, amelyek elemzése, kiterjesztése nem képezi célját a kutatásnak: AECTP[64], ALP-10[65].

A NATO SLCM sikeres végrehajtását előirányzó alapelvek elemzéséből látható, hogy a kialakítandó agilis szoftverfejlesztési módszerekkel szemben támasztandó követelmények kinyeréséhez az AAP-48 kiadvány feldolgozása maradt hátra.

[61] AAPP – Beszerzési gyakorlattal kapcsolatos kiadványok, a rövidítés az angol **A**llied **A**cquisition **P**ractices **P**ublication kifejezés szavainak kezdőbetűiből származik. Nyilvánosan nem érhető el.
[62] AACP – Nemzetközi együttműködést igénylő megegyezések tárgyalási és előkészítési irányelve, angolul: Guidance for the Negotiation and Drafting of International Co-operative Armaments Arrangements. Nyilvánosan nem érhető el.
[63] ALCCP-1 – Életciklussal kapcsolatos költségekre vonatkozó irányelv, angolul: NATO Guidance on Life Cycle Costs. Nyilvánosan nem érhető el.
[64] AECTP – Környezeti teszteléssel kapcsolatos kiadványok, a rövidítés az angol **A**llied **E**nvironmental **C**onditions and **T**ests **P**ublication kifejezés szavainak kezdőbetűiből származik. Nyilvánosan elérhető kiadvány a sorozatból többek között az AECTP-100. [75]
[65] ALP-10 – Nemzetközi katonai programok integrált logisztikai támogatásának irányelve, angolul: NATO Guidance on Integrated Logistics Support for Multinational Armament Programmes. [76]

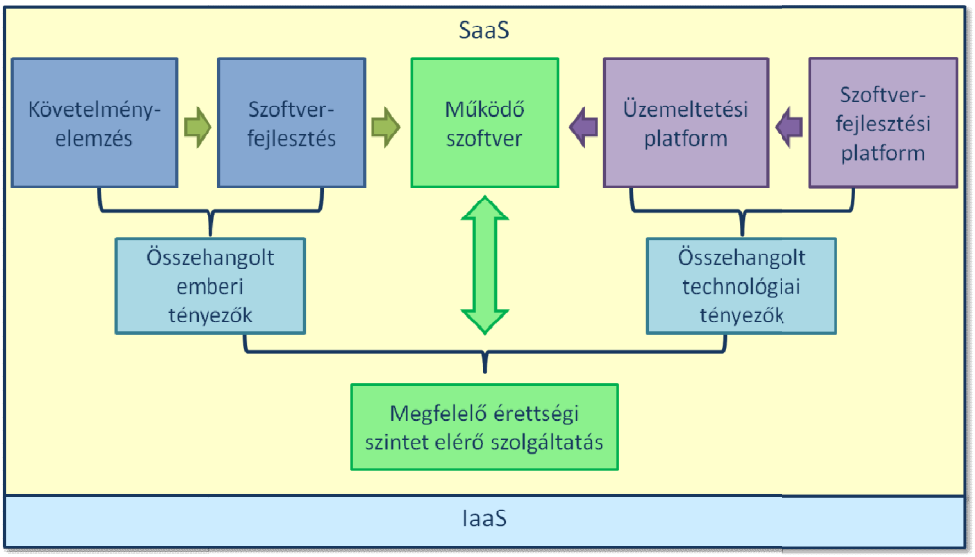
A NATO életciklus folyamatok a katonai rendszerek teljes élettartama alatt lehetővé teszik a felmerülő követelmények megfogalmazását, a követelmények alapján a fejlesztési koncepció kialakítását, végül a továbbfejlesztést. Az AAP-48 kiadványban találhatók meg az életszakaszok végrehajtását támogató folyamatok leírásai, figyelembe véve a katonai célú felhasználás speciális követelményeit. Ahogy a civil életben, úgy a katonai alkalmazás során is egyre összetettebb rendszerekkel találkoznak az érintettek, amelyek kialakítása és fenntartása egyre nagyobb kihívást jelent. Három oka van a jelenségnek: [71, 1.2]

1. *„Eredendő különbségek vannak a rendszereket képező hardver, szoftver és emberi tényezők között."* – más karakterisztikát mutat egy hardver vagy szoftver alapú szolgáltatás, illetve egy személyzetileg megvalósított támogató szolgáltatás. Ezek tervezése, kialakítása, fenntartása merőben eltérő folyamatokat takarhat, ezért célszerű kutatni azok lehetőségeit, e tanulmány a szoftver jellegű aspektusokat vizsgálja.
2. *„Szinten minden napjainkban használatos rendszernek van kapcsolata a számítógépes technológiával: teljes egészében képezi a megvalósítást, tartalmazza azt vagy támogatott általa."* – Ez azt is jelenti, hogy kiemelt szükség van a számítógépes technológiára épülő megoldások kialakításnak módszertani fejlesztésére, azok rendszerintegrációjának fokozására. Ezek a feladatok egybevágnak jelen kutatás alapvető céljaival.
3. *„Nincs megfelelő harmonizáció és integráció az érintett diszciplínák között, ideértve a tudományt, tervezést, menedzsmentet és pénzügyeket."* – Értekezésem esetében a tudományos, a tervezési és a menedzsment területek integrációja a pénzügyi terület erőforrásbecsléshez köthető kérdéseinek érintésével kutatási célként szerepel.

Az AAP-48 kiadvány által meghatározott folyamatok, illetve a kutatási célként kitűzött módszertan alkalmazási területe jelentős átfedést mutat, így a dokumentumban szereplő folyamatok további támpontokat adhatnak. A vizsgálódás hatókörének szűkítése végett a továbbiakban szoftver alapú szolgáltatásokra vetítve kerülnek feldolgozásra a fejezetek folyamat szintű és dokumentációs követelmények után kutatva.

A szoftver alapú szolgáltatási modellben két szignifikáns tényező az ember és a szoftvertechnológia jelenik meg. A szolgáltatás alapját képező szoftver teljesítményének megfelelőnek kell lennie, el kell érnie a megfelelő érettségi mutatókat. Ehhez a követelményhez az üzleti igényeket támasztó szakértőktől a szoftverfejlesztést vég-

ző szakembereken át, az üzemeltetőknek összehangoltan, megfelelően kell együttműködniük, ugyancsak elérve a közösen végrehajtott folyamataik során a megfelelő érettségi szintet. A kialakítandó szoftverfejlesztési technikáknak támogatniuk kell az emberek közötti együttműködést a szerepkörök és a feladatok egyértelmű meghatározásával. Emellett iránymutatást kell adniuk a szoftver előállítási és üzemeltetési folyamatok összehangolására. Valamint, ahhoz hogy szoftver alapú szolgáltatás önmagában is elérje a megfelelő érettségi szintet, a módszertannak elő kell segítenie a két tényező hatékony együttműködését. A folyamatot a következő ábra mutatja be.



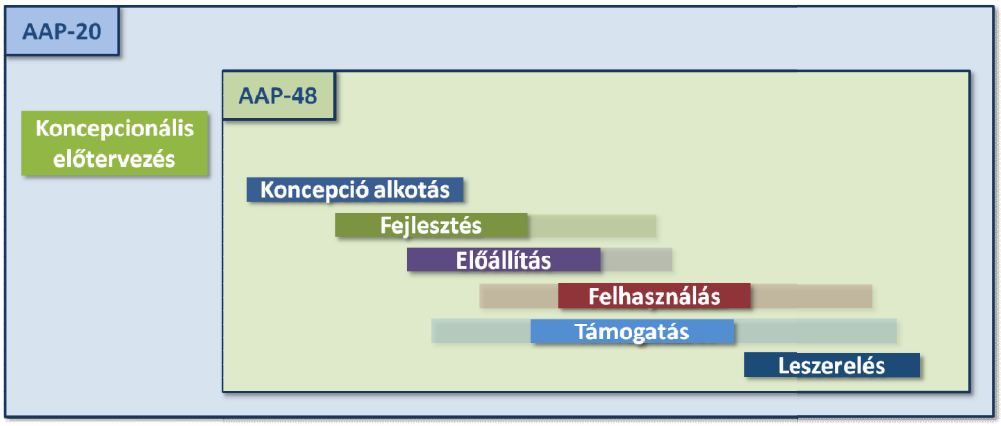
11. ábra Emberi és technológiai tényezők kapcsolata a SaaS modellben (saját szerkesztés)

A NATO Szabványosítási iránymutatása [77] a civil szabványok felhasználását javasolja a probléma kezelésére. Az AAP-48-ban bemutatott folyamatok az ISO/IEC 15288:2008 *„Systems Engineering – System Life Cycle Processes”* [78] szabványból kerültek levezetésre. A kiterjesztés során az alábbi eljárásokat alkalmazták az ISO 15288 folyamatainak tekintetében.

1. Folyamat átvétele teljes mértékben, változtatás nélkül.
2. Folyamat kiterjesztése katonai specialitásokkal, megfelelő AQAP szabvány vagy NATO dokumentumsablon felhasználására történő utalással.
3. Új folyamat meghatározása – a katonai felhasználás számára kialakított speciális folyamat bevezetése az ISO 15288 szabvány szerinti folyamat leíró struktúrában.

A további elemzés során az AAP-48 kiadványban fellelhető 2.-3. típusú pontok tárgyalása következik, azokban az esetekben, ahol van szoftvertechnológiai kapcsolódási pont, így további folyamat szintű követelmények azonosíthatók.

A koncepcionális előtervezési szakasz támogatása nem képezi részét az AAP-48 dokumentumnak. A kiadvány célja a koncepció alkotási szakasz és az azt követő további szakaszok folyamatainak támogatása. Az ISO/IEC 15288:2008 szabvány – ahogy az AAP-48 is – négy területre bontja a műszaki projektek folyamatait, ezek rövid áttekintése következik további folyamat és dokumentációs követelmények után kutatva. Az egyes folyamatokhoz a NATO Szabványosítási Hivatala[66] (a továbbiakban: NSO) által gondozott dokumentumok kapcsolódnak, amelyek a szabályozott végrehajtást segítik elő folyamatleírásokkal és dokumentációs sablonokkal. Ebben az alfejezetben a projekt szó alatt tetszőleges NATO programot, illetve projektet értek.



12. ábra Az AAP-48 és az AAP-20 kapcsolata az NATO SLCM-ben. Forrás: [71, 1-3 o.]

Az AAP-20 és AAP-48 kiadványok kapcsolatát a 12. ábra mutatja be, amely alapján látható, hogy az AAP-20 lefedi a végrehajtás során elvégzendő tevékenységek teljes spektrumát, ezzel szemben az AAP-48 csak a koncepció alkotástól a leszerelésig ad iránymutatást. A halvány színnel meghosszabbított életciklus szakaszokat reprezentáló téglalapok a bizonytalan szakaszok közötti átmenetekre utalnak. A szemléltetett jelenség a programok/projektek végrehajtásában mutatkozó eltérésekből fakad.

3.2.1 Szerződéses folyamatok

A katonai képességfejlesztés területén a szerződésekkel kapcsolatos folyamatok két csoportra különülnek el: megrendelői és beszállítói folyamatok. A megrendelői folyamatok támogatottsága nagyobb hangsúlyt kap, ennek oka, hogy a katonai képességfejlesztések során a NATO, illetve az egyes NATO szervek a megrendelői oldalt képviselik. A beszállítók által készített termékek minőségét a megfelelő minőségbiztosítási szabványok betartása garantálja a védelmi ágazaton belül is. Beszállítói olda-

[66] NATO Szabványosítási Hivatal – NATO Standardization Office, röviden: NSO. 2014 júliusa előtt NATO Standardization Agency (NSA)-ként működött.

lon a civil szabványok alkalmazása elősegítheti a sikeres projektek megvalósítását, azonban a NATO beszerzések esetében ezek nem tekinthetők mérvadónak, a megfelelő AQAP szabványok alkalmazása a követelmény, amelyek *„garantálják a beszállítók folyamataival, létesítményeivel, a szerződések végrehajtásával és a termékekkel szemben támasztott elfogadási követelmények teljesülését."* [71, 6-3 old].

A különböző katonai szakterületek számára különböző AQAP szabványok állnak rendelkezésre. Szoftverfejlesztési projektek esetében az AQAP 2210 [79] szabvány alkalmazása az elvárás. Ezt az elvárást kutatási követelményként is fel lehet fogni. *A kialakítandó szoftverfejlesztési módszertannak összhangban kell lennie az AQAP 2210 szabványban foglaltakkal.*

A szabványok alkalmazásával javítható a fegyveres erők közötti együttműködési programok során az átláthatóság. Szoftvertechnológiai szempontból érdekes, hogy már szerződéses szinten szabályozandó terület a *„szoftverekre vonatkozó szellemi tulajdon, elektronikus adatmegosztás, az adatbázisok létrehozása és használata"* [71, 6-2 old.] Ezekben az esetben az AACP kiadványok használata javasolt. Mindazonáltal, az iménti adatkezelést firtató követelmény felfogható a kialakítandó szoftverfejlesztési módszertan dokumentum sablonjaival szemben támasztott elvárásként is. Az elvárás beépítése a kutatási követelményrendszerben két követelmény segítségével valósítható meg. *A kutatás céljaként kialakítandó szoftverfejlesztési módszertanhoz tartozzon projekt menedzsment sablon.*

A szoftverekre vonatkozó szellemi tulajdon, az elektronikus adatmegosztás, az adatbázisok létrehozása és használata kapjon helyt a projekt menedzsment sablonban.

Amennyiben kormányzati vagy nemzetközi együttműködésre van szükség, akkor a STANAG 4107[67] és az AQAP 2070 [81] dokumentumok nyújthatnak segítséget, azonban ezek további elemzése meghaladja a kutatás hatókörét. Amíg a megrendelői oldal feladatainak áttekintése markáns az életciklus folyamatok tárgyalása során, addig a beszállítói folyamatokkal szemben kizárólag az AQAP szabványcsaládban foglaltak teljesülése az elvárás.

3.2.2 Szervezeti projekt-támogató folyamatok

A szervezeti szinten megvalósuló folyamatok közül a minőségirányítási folyamatok kerültek kiterjesztésre az AAP-48 kiadványban, a területet az AQAP 2000 szabvány

[67] STANAG 4107 – A kormányzati és a szövetségi (AQAP) minőségbiztosítás kölcsönös elfogadásáról szóló szabvány, angolul: Mutual Acceptance of Government Quality Assurance and Usage of the Allied Quality Assurance Publications (AQAP). [80]

szabályozza. Megrendelői szinten kell elvégezni a minőségirányítással kapcsolatos feladatokat szervezeti és projekt szinten egyaránt, a folyamatnak dokumentáltnak, felügyeltnek kell lennie az adott katonai képesség kialakítása és felhasználása során. A minőségirányítási feladatok ellátását a projektmenedzsment részeként kell végrehajtani. A minőségbiztosítási tervnek tartalmaznia kell a folyamatok minőségének mérését és az irányítást elősegítő eljárásokat is. Továbbá, biztosítania kell azt is, hogy a projekt végrehajtása során kialakított megoldások a meghatározott követelményeknek megfeleljenek. Ehhez támogató tervek, szabványok, eljárások, iránymutatások nyújthatnak segítséget. A támogató eljárások feladatai lehetnek az elemzés, a kiszűrés, az áttekintés, a felülvizsgálat és az értékelés. A minőségbiztosítási tervnek indikálnia kell a kapcsolatot a minőségbiztosítás és életciklus során végzett tevékenységek között.

A megfelelően szabályozott szervezeti szintű eljárások mellett a megfelelően szabályozott projekt folyamatokkal lehet elősegíteni a projektek sikeres végrehajtását. Az AAP-48 dokumentumban az ISO/IEC 15288 projekt folyamatok túlnyomó része kiegészítésre kerül a NATO által meghatározott sajátos katonai követelményekkel. A projektek végrehajtásának tervezési folyamata az első terület, ahol megjelennek a speciális katonai megkötések. Az AAP-20 kiadvány 5. számú melléklete tartalmazza az általános projekt terv dokumentáció mintát, amely alapján kialakítható egy kimondottan szoftverfejlesztési projektek támogatására alkalmazható dokumentum sablon. Tekintettel arra, hogy a projektek megvalósítása során el kell végezni a minőségirányítási feladatokat, az is megfogalmazható követelményként, hogy a projekt menedzsment terv tartalmazza a szükséges eljárások lépéseit – többek között a minőségirányítással kapcsolatos teendőket is. A további elemzés során feltárt projekt dokumentációs követelmények az alábbi kutatási követelmény folyamatos bővítésével kerülnek kimondásra. *A kutatás céljaként kialakítandó szoftverfejlesztési módszertanhoz tartozó projekt menedzsment sablon tárgyalja a minőségirányítási feladatokat is.*

3.2.3 Projekt folyamatok

A projektek végrehajtásának értékelését különböző civil technikákkal, illetve a NATO programok esetében az AQAP 2050 [82] szabvány segítségével lehet megvalósítani. Szem előtt tartandó szempont, hogy a belső értékelések képezik alapját a sikeres független értékeléseknek, adott esetben a megrendelő elégedettségnek is. Ab-

ban az esetben, ha kizárólag szoftverfejlesztési tevékenység képezi a projekt tárgyát, akkor reális követelmény lehet a következő. *A szoftverfejlesztési projekt végrehajtása során folyamatos mérőszámok mutassák a fejlesztés alatt álló szoftver pillanatnyi érettségét.*

A folyamatos értékelés és önellenőrzés mellett megvalósítandó a folyamatos kockázatelemzés is a projektek végrehajtása során. A szerződéskötések során azonosított kockázatok kezelésével kapcsolatos kérdéseknek, tevékenységeknek meg kell jelenniük a projektmenedzsment feladatok ellátása során is. A NATO programok végrehajtása során az AQAP-2070 szabvány és az ARAMP-1 [83] iránymutatás nyújt segítséget a kockázatkezelés számára. Szoftverfejlesztési projektek esetében a kockázatkezelés speciális feladatokat takarhat a különböző biztonsági, hardver, környezeti, jogszabályi, stb. követelményeknek történő együttes megfelelés miatt, ezért reális cél lehet a következő követelmény. *A kialakítandó szoftverfejlesztést támogató dokumentációs technikák támogassák a kockázatok elemzését és kezelését a projektek végrehajtása során.*

Egy kialakítandó katonai képesség számára kulcsfontosságú jelentőséggel bír a kapcsoló projekt szintű konfiguráció menedzsment szabályozott végrehajtása. A rendszerek közötti átjárhatóság és a kompatibilitás képessége rejlik a megfelelően kivitelezett konfiguráció menedzsmentben. Ráadásul ez egy olyan tevékenység, amelyet a rendszerek teljes életciklusa alatt megrendelői, gyártói, fejlesztői és felhasználói oldalon is elvégezhetnek. A projektszinten megvalósított konfiguráció menedzsment lényege, hogy az életciklus folyamatok során elvégzett konfiguráció módosítások részét képezik a projektnek és dokumentáltan történnek.

A projektek végrehajtása során keletkező, az adott katonai képességet megvalósító rendszerrel kapcsolatban létrejövő releváns információ megfelelő kezelése szervesen összefügg a konfiguráció menedzsment kérdéskörével. Az információ menedzsment végrehajtása kulcsfontosságú a rendszerek visszahívását, leszerelését követően is. A NATO programok végrehajtása során különböző szintű dokumentumok segíthetnek a szabályozott végrehajtásban: STANAG 4427 [84], ACMP[68] kiadványok. A szoftverfejlesztési projektek vonatkozásában két elkülönülő követelményt is megfogalmazható a konfiguráció menedzsment és az információ menedzsment során keletkező adatok vonatkozásában. *A kialakítandó szoftverfejlesztést támogató dokumentációs tech-*

[68] ACMP – Megrendelői konfiguráció menedzsment terv. Az ***A**cquirers for Life Cycle **CM P**lan* angol kifejezés szavainak betűiből. (ACMP-2009 [85], ACMP-2100 [86] kiadványok állnak rendelkezésre)

nikák támogassák a konfiguráció menedzsmentet érintő tevékenységek során keletkező adatok dokumentálását. A szoftverfejlesztési projekt teljes életciklusa során keletkező információ strukturált formában kerüljön megőrzése és később álljon rendelkezésre.

Az eddig bemutatott követelmények az ISO/IEC 15288 projektfolyamatainak kiterjesztéseként kerültek meghatározásra. Az AAP-48 dokumentum egy teljes élettartam alatt alkalmazható nyomkövetési eljárást is meghatároz a NATO képességet jelentő SOI-k esetében. A cél eléréséhez a SOI-k és azonosítható elemeik esetében kötelező a teljes élettartam alatti egyedi azonosítás alkalmazása. Ez a fajta nyomon követési stratégia lehetővé teszi a legkisebb komponensek és a követelmények között fennálló kölcsönhatások azonosítását, ezzel támogatva a döntések hatásainak nyomon követhetőségét és a folyamatos fejlesztés lehetőségét.

A nyomon követhetőségi képesség megteremtéséhez először meg kell határozni azt az infrastruktúrát és környezetet, amelyen belül az adatok gyűjtése megvalósul. Ezt követően folyamatosan felül kell vizsgálni az infrastruktúra adta lehetőségeket és azokat a projekt igényekhez kell igazítani. Az infrastruktúrának képesnek kell lennie kezelni azokat a kulcsfontosságú adatokat, amelyek az adott SOI-t kellőképp leírják az élettartama alatt. Az adatok gyűjtéséhez ki kell alakítani azt a megfelelően strukturált adatmodellt, amely az érintett szereplők számára is egyértelmű. A projekt végrehajtása során gondoskodni kell az adatok frissítéséről, az adatok minőségének folyamatos biztosításáról, illetve a szabályozott hozzáférésről is. Az adott SOI nyomom követésén kívül lehetőség van a kapcsolódó folyamatok, életciklus szakaszok, eljárások, beszerzett eszközök nyomon követésére is. A folyamatok támogatására szolgáló dokumentumok: STANAG 2290, STANAG 4661 – utóbbiak nyílt forrásból nem érhetők el – és az AUIDP-1[69]. A *nyomom követési folyamatnak* le kell fednie a szerződéskötés, a döntéstámogatás, a konfiguráció menedzsment, az átadás-átvétel, az üzemeltetés, a karbantartás és a leszerelési folyamatokat. Fizikailag létező elemek esetében alkalmazhatók a különböző gyári számok és egyedi azonosítók. A SOI életciklusának meghatározó adatainak is hasonlóképp azonosíthatónak és nyomon követhetőnek kell lenniük, még abban az esetben is, ha fizikailag nem nyilvánulnak meg. A szoftverfejlesztési projektek esetében sem valósítható meg az egyedi azono-

[69] AUDIP-1 – Egyedi azonosítást támogató iránymutatás, angolul: Guidance on Unique Identification of Items [87]

sítás gyári számokkal, ezen a területen a fejlesztési feladatok azonosítói, a szoftververziók és változási jegyzékek képezik a nyomon követhetőség alapját.

13. ábra Projekt szintű feladatok és az SLCM kapcsolata (saját szerkesztés)

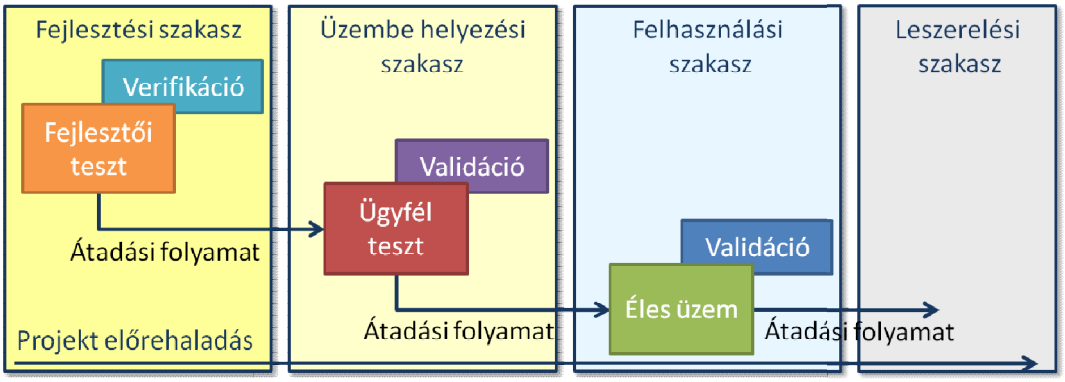
Ez alapján a kialakítandó technikákkal szemben megfogalmazható követelmény a következő. *Szoftverfejlesztésre értelmezett követelményeket, fejlesztési feladatokat, verziókat, konfigurációs beállításokat, üzemeltetési és karbantartási tevékenységeket felölelő nyomkövetési infrastruktúra kialakítása szükséges.*

3.2.4 Technikai folyamatok

A projektek sikeres végrehajtását elősegítő NATO megállapodások, szabványok és kiadványok szabályozott keretet biztosítanak a projektfolyamatok szintjén. A nyomon követési folyamatok az adott katonai képesség érettségi szintjének elérését és fenntartását megfelelő szinten képesek támogatni. Mindazonáltal, a technikai folyamatok támogatásához további megfontolásokat találhatunk az AAP-48 kiadványban. Az architekturális tervezés, a megvalósítás és az integráció folyamatai nem kerültek kiegészítésre, ugyanakkor a verifikáció – a rendszer konzisztenciájának ellenőrzése – már különböző támogató dokumentumok segítségével zajlik. A kialakítás alatt álló katonai képesség helyességének ellenőrzése során az AQAP minőségbiztosítási szabványok alkalmazása szükséges. Azokban az esetekben, ahol a környezeti tényezők ellenőrzése is részét képezi a verifikációnak az AECTP kiadványok adnak iránymutatást.

A verifikáción átesett rendszerek esetében következő lépésként az *átadási folyamat* valósul meg. Az átadási folyamat alapvetően az előállítási szakasz és a felhasználási szakasz közötti átmenetet takarja, ugyanakkor az is elképzelhető, hogy előbb köztes

környezeteken jelenik meg az ügyfél általi tesztelésre szánt, ugyanakkor már használatba vehető SOI. Az átadási folyamat lényeges mozzanata, hogy az átadás tárgyát képező SOI, valamint a hozzá tartozó támogató rendszerek is átadásra kerülnek az átadó és az átvevő szervezetek között, ezáltal az üzemeltetési, fenntartási felelősség is átkerül az átvevői oldalra, hacsak erről másképp nem rendelkeztek a szerződéses feltételekben. Szoftverek esetében a fejlesztési szakaszban előállított szoftververzió átadása a megrendelői tesztek elvégzésére jelenti az első átadási folyamatot.



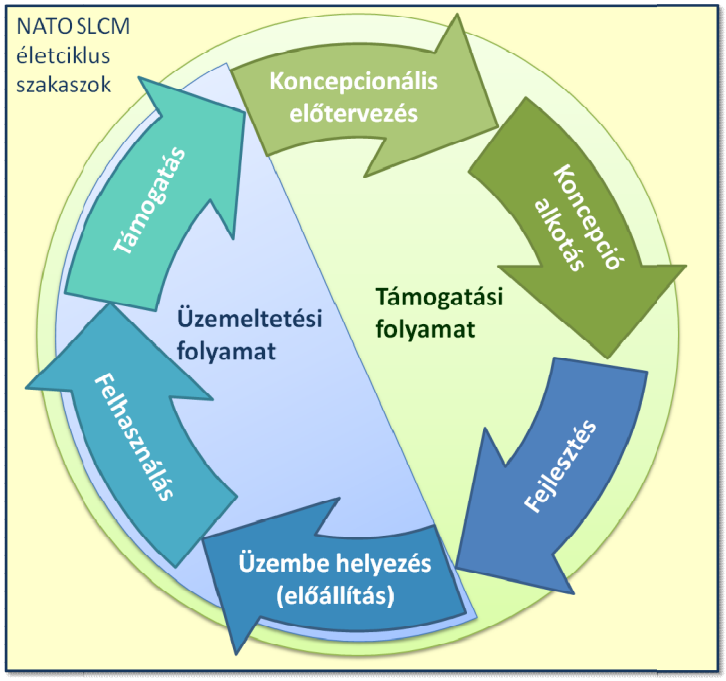
14. ábra Szoftver átadási folyamat bemutatása (saját szerkesztés)

Az átadásokat követően következhet a validáció folyamata – a megfelelősség ellenőrzése. Ilyenkor a felhasználók az adott felhasználási környezetben vizsgálják a rendszer működését, használatra alkalmasságát. A folyamat támogatására szolgálnak a SOI jellegétől függően a különböző AQAP szabványok. A 14. ábrán láthatók az átadások utáni validációs pontok. Abban az esetben, ha egy szoftver nyújtotta szolgáltatás validációjáról beszélünk, akkor azt kell ellenőrizni, hogy az adott informatikai szolgáltatás részeként elérhetők és megfelelően működnek a rendszer nyújtotta funkciók.

A NATO SLCM felhasználási szakaszához kapcsolódó eljárásokat az *üzemeltetési folyamat* fedi le, amely nagyon eltérő lehet a SOI és támogató rendszerek jellegétől függően. A folyamat támogatásához a STANAG 4704 [88], illetve az ISO 17025 [89] dokumentumok adnak iránymutatást. Szoftvertechnológia értelemben az adott szoftver alapú szolgáltatáshoz kapcsolódó verzióváltások, kiterjesztések támogatását takarja az üzemeltetési folyamat a megszakítás nélküli szolgáltatási képesség biztosítása mellett.

A projekt folyamatok esetében a nyomon követéssel került kibővítésre a kiindulási alapnak tekintett ISO 15288 szabvány. Hasonlóan ehhez, a technikai folyamatok is kibővítésre kerültek egy *támogatási folyamattal*, amelynek feladata egy támogató rendszer kialakítása a projekt tárgyát képező SOI számára. A támogatási folyamat

segítségével a követelményeknek megfelelő, használható és fenntartható megoldás alakítható ki. Az eljárás alkalmazásának végeredményeként egy olyan megoldásnak kell létrejönnie, amely képes elvégezni a SOI fenntarthatóságához szükséges feladatokat, alkalmazkodik a SOI tervezési megfontolásaihoz, és a SOI üzemeltetésének támogatását is el tudja látni a rendszer teljes élettartama alatt.



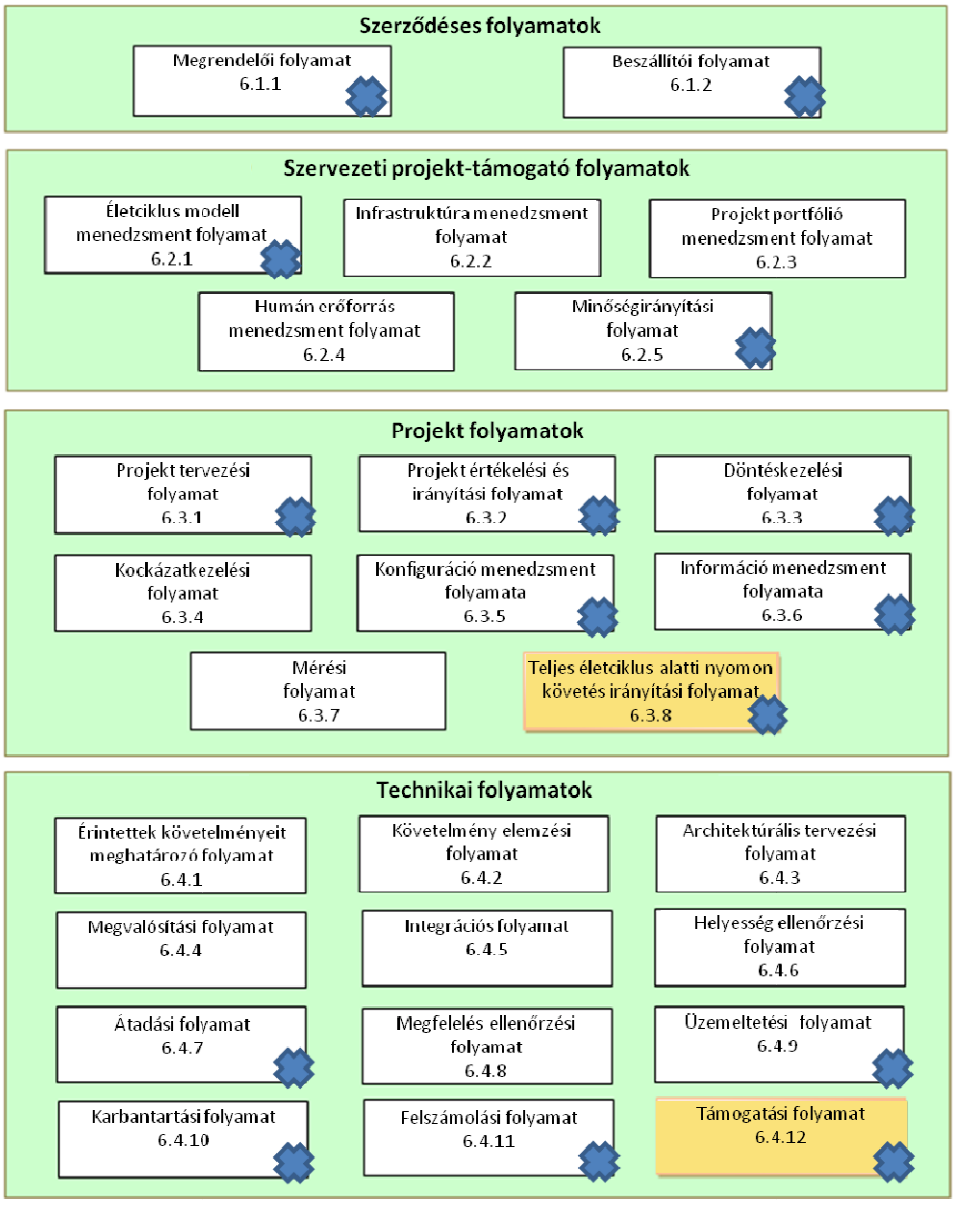
15. ábra Az SLCM és a technikai folyamatok kapcsolata (saját szerkesztés)

A 15. ábrán az AAP-48 kiadványban fellelhető kibővített technikai folyamatokat láthatjuk, illetve azt, hogy az üzemeltetési folyamat, valamint a támogatási folyamat mely területekkel áll kölcsönhatásban a szoftverfejlesztési projektek vonatkozásában. Ezen a ponton érdemes kitérni arra, hogy a tárgyalt SOI jelen esetben egy szoftver vagy egy szoftver alapú szolgáltatás, amelynek támogatási és üzemeltetési feladatai ideális esetben egyaránt támogathatók informatikai rendszerek, speciális szoftverek segítségével.

Figyelembe véve az iménti megfontolást, a technikai folyamatok során feltárt elvárásokra a válasz lehet technológiai is. A kutatási követelmény meghatározásakor figyelembe lehet venni a projekt folyamatok során feltárt eljárásokat is, így egy integrált informatikai megoldás képességének megteremtése is reális céllá válhat.

A projekt- és a technikai folyamatok támogatása integrált megoldás segítségével valósuljon meg, beleértve a verifikáció, az átadás, a validáció, az üzemeltetés, a karbantartás és a leszerelés folyamatait, valamint kielégítve a nyomon követési infrastruktúra számára szükséges adatgyűjtési követelményeket.

A 16. ábrán az AAP-48 kiadványban bemutatott folyamatok szerepelnek, kék színnel jelölve, hogy mely folyamatok feldolgozása során sikerült kutatási követelményeket feltárni.



16. ábra AAP-48 folyamatok. Forrás: [71, 5-3 o.]

A kutatás céljaként kialakítandó módszerek és dokumentációs technikák kiegészíteni kívánják a NATO életciklus modelljének folyamatait és dokumentációs sablonjait a szoftverfejlesztési projektek területén. Emellett elmondható az is, hogy a kutatást sikerült megfelelő kontextusba helyezni katonai oldalról is.

A kutatási hipotézisekben feltételezett katonai célra kialakított agilis szoftverfejlesztési módszernek meg kell felelnie a fejezetben feltárt követelményeknek. A következő oldalon az azonosított követelmények felsorolása szerepel összesítve, megfelelő kérdéskörökhöz besorolva.

Módszertani követelmény

M-20 A kialakítandó szoftverfejlesztési módszertannak összhangban kell lennie az AQAP 2210 szabványban foglaltakkal. Lásd: 111. old.

Szoftvertechnológiai követelmények

S-24 A szoftverfejlesztési projekt végrehajtása során folyamatos mérőszámok mutassák a fejlesztés alatt álló szoftver pillanatnyi érettségét. Lásd: 113. old.

S-25 A projekt- és a technikai folyamatok támogatása integrált megoldás segítségével valósuljon meg, beleértve a verifikáció, az átadás, a validáció, az üzemeltetés, a karbantartás és a leszerelés folyamatait, valamint kielégítve a nyomon követési infrastruktúra számára szükséges adatgyűjtési követelményeket. Lásd: 117. old.

Dokumentációs követelmények

D-4 A kutatás céljaként kialakítandó szoftverfejlesztési módszertanhoz tartozzon projekt menedzsment sablon. Lásd: 111. old.

D-5 A szoftverekre vonatkozó szellemi tulajdon, az elektronikus adatmegosztás, az adatbázisok létrehozása és használata kapjon helyt a projekt menedzsment sablonban. Lásd: 111. old.

D-6 A kutatás céljaként kialakítandó szoftverfejlesztési módszertanhoz tartozó projekt menedzsment sablon tárgyalja a minőségirányítási feladatokat is. Lásd: 112. old.

D-7 A kialakítandó szoftverfejlesztést támogató dokumentációs technikák támogassák a kockázatok elemzését és kezelését a projektek végrehajtása során. Lásd: 113. old.

D-8 A kialakítandó szoftverfejlesztést támogató dokumentációs technikák támogassák a konfiguráció menedzsmentet érintő tevékenységek során keletkező adatok dokumentálását. Lásd: 113. old.

D-9 A szoftverfejlesztési projekt teljes életciklusa során keletkező információ strukturált formában kerüljön megőrzése és később álljon rendelkezésre. Lásd: 114. old.

D-10 Szoftverfejlesztésre értelmezett követelményeket, fejlesztési feladatokat, verziókat, konfigurációs beállításokat, üzemeltetési és karbantartási tevékenységeket felölelő nyomkövetési infrastruktúra kialakítása szükséges. Lásd: 115. old.

3.3 ÖSSZEGZÉS

A kutatás deduktív szakaszának végszavaként elmondható, hogy a NATO katonai képességfejlesztési módszertana – az SLCM – lefedi a katonai rendszerek teljes élettartamát, ehhez megfelelő modellt és folyamatokat biztosít. Az elemzés során láthattuk a szakaszokra bontott életciklust, valamint az egyes szakaszok hatókörét és feladatrendszerét. A kutatási célokhoz igazodva azonosításra kerültek a szoftvertechnológiai aspektusból értelmezhető módszertani, technológiai és dokumentációs követelmények.

A NATO SLCM irányelv elemzése során hét szoftvertechnológiai és három módszertani követelményt meghatározása történt meg. Az életciklus modellt bemutató NATO AAP-20-ban tizenkettő módszertani és három szoftverfejlesztésre értelmezhető dokumentációs elvárás definiálása valósult meg.

Végezetül, a folyamatok áttekintése során egy lényeges AQAP szabványnak való megfelelésre vonatkozó módszertani, ezen túl két szoftvertechnológiai és hét dokumentációs követelmény is megfogalmazódott. A fejezetben feltárt új kutatási követelmények a továbbiakban szerepelnek kategóriánként felsorolva.

Módszertan

M-5 A megfizethetőség, az idő, az ütemezés, a minőség és a kockázati tényezők együttes értelmezése szükséges a katonai célú informatikai rendszerek fejlesztése során.

M-6 Integrált és zökkenőmentes üzletkötési, projektmenedzsment gyakorlatok kialakítása szükséges a koncepció-alkotástól a kivezetésig.

M-7 Teljes életciklusra vonatkozó, minden érdekelt fél között hatékony együttműködés kialakítása a cél, jól definiált felelősségi körökkel.

M-8 A kialakítandó szoftverfejlesztési módszer értelmezze az érettségi szint fogalmát és feleljen meg a NATO Rendszer Koncepciónak.

M-9 A kialakítandó szoftverfejlesztési módszer adjon eszközöket a stratégiai célok dokumentálására, a szereplők azonosítására, a követelmények meghatározására és priorizálására, támogassa a változáskezelést, valamint folyamatosan adjon megfelelő visszajelzést a rendszer érettségével kapcsolatban.

M-10 Az alternatív megvalósíthatósági tanulmányok kialakítását folyamat és dokumentációs szinten is támogassa a kialakítandó szoftverfejlesztési módszertan.

M-11 A kutatás céljaként kitűzött kialakítandó szoftverfejlesztési módszer folyamataiban jelenjenek meg a felsorolásban szereplő tevékenységek: követelmények áttekintése, funkcionális áttekintés, tervezési áttekintés, tesztelésre alkalmasság ellenőrzése, konfigurációs felülvizsgálat, megfelelőségi ellenőrzés, helyességi ellenőrzés, éles üzembe helyezésre alkalmasság ellenőrzése.

M-12 A kialakítandó technikáknak támogatniuk kell a döntéshozási mechanizmusokat, hogy az érdekeltek kellő mennyiségű és minőségű információval rendelkezzenek az elengedhetetlen közös álláspont kialakításához.

M-13 *A kialakítandó szoftverfejlesztési módszer támogassa a módosítási eljárás koncepció alkotási szakaszában elvégzendő szoftvertechnológiai mozzanatokat: duplikált igények felismerése, nem teljesíthető követelmények azonosítása, kockázatkezelés.*

M-14 A kialakítandó szoftverfejlesztési módszer támogassa a technológiai lehetőségek kiértékelését, a konfiguráció és rendszer kapcsolatának vizsgálatát, a módosítás hatásának vizsgálatát a rendszer komplexitására, az átvételi követelmények finomítását, a költségbecslést, a megkötések azonosítását, a kockázatelemzést, a megkerülő megoldások biztosítását, a további érettséget előirányzó módosítások meghatározását.

M-15 A kialakítandó szoftverfejlesztési módszer támogassa az előzetes tervezést és elemzést, a prototípuskészítést, a bemutatókat, a laboratóriumi teszteket, az optimális technológiák meghatározását a hatékonyság, a költségek, az idő és a kockázatok vonatkozásában, megkerülő megoldások meghatározását, a követelményrendszer véglegesítését.

M-16 A kialakítandó szoftverfejlesztési módszer támogassa a fejlesztést, a tesztelést, az értékelést, a helyesség és a megfelelés ellenőrzését, illetve a követelmények és a specifikáció finomítását, valamint a rendszer verziók korlátlan előállítását.

M-17 A kialakítandó szoftverfejlesztési módszer támogassa a támogató dokumentációk előállítását, a konfigurációkban bekövetkező változtatások követését, az üzemeltetési környezet és támogató rendszerek felkészítését és az átvételi teszt elvégzését.

M-18 A kialakítandó folyamatok a koncepcionális előtervezés szakaszával kezdve korlátlanul legyenek megismételhetők.

M-19 A kialakítandó szoftverfejlesztési módszertan támogassa a tárolt adatok migrálhatóságát, a funkciók követelményeinek kinyerését, illetve az üzemeltetési tapasztalatok kinyerését egy szoftver alapú szolgáltatás megszűntetésekor.

M-20 *A kialakítandó szoftverfejlesztési módszertannak összhangban kell lennie az AQAP 2210 szabványban foglaltakkal.*

Szoftvertechnológia

S-17 Hatékonyság

S-18 Telepíthetőség

S-19 Robusztusság

S-20 Karbantarthatóság

S-21 Fenntarthatóság

S-22 Szükségszerű a technológiai bővítés lehetővé tétele, az élettartam alatti módosítások, az elévülés kezelése.

S-23 A fejlesztések számára integrált rendszerszemlélet meghatározása szükséges, amely támogatja a felhasználást, elősegíti a követelményeknek való megfelelést.

S-24 A szoftverfejlesztési projekt végrehajtása során folyamatos mérőszámok mutassák a fejlesztés alatt álló szoftver pillanatnyi érettségét.

S-25 A projekt- és a technikai folyamatok támogatása integrált megoldás segítségével valósuljon meg, beleértve a verifikáció, az átadás, a validáció, az üzemeltetés, a karbantartás és a leszerelés folyamatait, valamint kielégítve a nyomon követési infrastruktúra számára szükséges adatgyűjtési követelményeket.

Dokumentáció

D-1 A NATO SLCM koncepció alkotási szakasz során keletkező szoftverfejlesztéshez köthető dokumentumoknak helyt kell kapniuk a kialakítandó szoftverfejlesztési módszertan dokumentum sablonjaiban.

D-2 A NATO SLCM fejlesztési szakasz során keletkező szoftverfejlesztéshez köthető dokumentumok előállítását dokumentum sablonok szintjén támogassa a kialakítandó szoftverfejlesztési módszertan.

D-3 A kialakítandó szoftverfejlesztési módszertan részeként kerüljenek meghatározásra azok a dokumentációs sablonok, amelyek megfelelően támogatják a

NATO SLCM üzembe helyezési, a felhasználási és a támogatási szakaszok végrehajtását.

D-4 A kutatás céljaként kialakítandó szoftverfejlesztési módszertanhoz tartozzon projekt menedzsment sablon.

D-5 A szoftverekre vonatkozó szellemi tulajdon, az elektronikus adatmegosztás, az adatbázisok létrehozása és használata kapjon helyt a projekt menedzsment sablonban.

D-6 A kutatás céljaként kialakítandó szoftverfejlesztési módszertanhoz tartozó projekt menedzsment sablon tárgyalja a minőségirányítási feladatokat is.

D-7 A kialakítandó szoftverfejlesztést támogató dokumentációs technikák támogassák a kockázatok elemzését és kezelését a projektek végrehajtása során.

D-8 A kialakítandó szoftverfejlesztést támogató dokumentációs technikák támogassák a konfiguráció menedzsmentet érintő tevékenységek során keletkező adatok dokumentálását.

D-9 A szoftverfejlesztési projekt teljes életciklusa során keletkező információ strukturált formában kerüljön megőrzése és később álljon rendelkezésre.

D-10 Szoftverfejlesztésre értelmezett követelményeket, fejlesztési feladatokat, verziókat, konfigurációs beállításokat, üzemeltetési és karbantartási tevékenységeket felölelő nyomkövetési infrastruktúra kialakítása szükséges.

Az első három fejezetben elvégzett feltáró munka során összesen 84 különböző kutatási követelményt sikerült azonosítani az alábbiak szerint.

Kategória (KÓD)	Ált. köv.	Spec. köv.	**Össz.**	Hivatkozás
Kiberbiztonság (K)	8	-	**8**	37, 52. old.
Információbiztonság (I)	8	4	**12**	34, 51. old.
Szoftvertechnológia (S)	4	21	**25**	45, 62, 72, 76, 119. o.
Követelményelemzés (R)	-	5	**5**	68. old.
Minőségbiztosítás (Q)	-	4	**4**	50. old.
Módszertan (M)	4	16	**20**	61, 76, 104, 119. old.
Dokumentáció (D)	-	10	**10**	105, 119. old.
Összesen:	**20**	**64**	**84**	

8. táblázat Az 1-3. fejezetekben feltárt kutatási követelmények számszerűsítése

4. KATONAI CÉLÚ INFORMATIKAI SZOLGÁLTATÁSOK KIALAKÍTÁSA AGILIS SZOFTVERFEJLESZTÉSI TECHNIKÁKKAL

A korábbi fejezetek célja a kutatás megfelelő kiberbiztonsági, információbiztonsági és szoftvertechnológiai környezetbe történő elhelyezése volt, az agilis szoftverfejlesztés katonai célú alkalmazásáról még nem esett szó az eddigiek során. Kutatásom megkezdésekor, 2016-ban a világhálón csak néhány cikk foglalkozott – és azok is csak érintőlegesen – a Scrum szoftverfejlesztési módszertan katonai céllal történő alkalmazásával. Napjainkra ez a tendencia megváltozott, cégek és katonai szervezetek is nyilvánosan vállalják fel, hogy ezt a módszertant alkalmazzák különböző védelmi területeken, többek között Olaszország [90] vagy az Egyesült Államok [91] hadereje is.

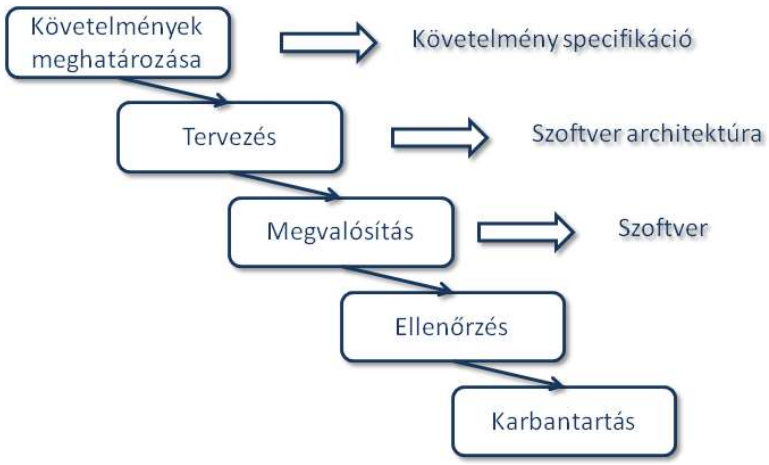
Ugyanakkor a módszertan helyes alkalmazása komoly kérdéseket vet fel a haderőfejlesztés vonatkozásában, mert a fejlesztett rendszerek sok esetben kritikus szerepet töltenek be. A fejlesztésekben résztvevő szereplők, a megrendelő, a felhasználók és a fejlesztők nem feltétlenül tudnak megfelelni egy Scrum módszertan által lefektetett szabályrendszernek, például nincs lehetőségük elegendő mennyiségű verbális kommunikációra a projekten belül – ez a probléma már az USA-ban is a felszínre került. [92] Ennek egyik oka, hogy egy hadsereg, mint megrendelő, nem hasonlítható a piaci szereplőkhöz, még a nagyvállalatokhoz sem igazán – a méreteiből és az összetett feladatrendszeréből kifolyólag sem.

Összességében azért elmondható, hogy már vannak gyakorlati tapasztalatok az agilis szoftverfejlesztési módszertanok katonai céllal történő alkalmazásáról – sőt vannak olyan veteránok is, akik sikeresen tevékenykednek agilis projektek résztvevőiként. [93] A 4. fejezet célja egy olyan szoftverfejlesztési módszertan meghatározása, amely egyértelműen alkalmazható katonai célra és megfelel a korábbi fejezetekben feltárt követelményrendszernek.

A kialakítandó módszertan meghatározása előtt az agilis szoftverfejlesztés, ezen belül a Scrum módszertan rövid áttekintése következik, bemutatva a szoftverfejlesztési projektek támogatásához kialakított modellt, amelyet manapság a különböző szoftverfejlesztő cégek egyre elterjedtebben használnak a megrendelői igények kielégítéséhez, majd ezt követően a katonai alkalmazás lehetőségeinek vizsgálata következik.

Napjainkban az informatikai projektek műszaki projektként jelennek meg a köztudatban, ugyanakkor műszaki projekt egy építészeti, gépészeti és természetesen egy szoftverfejlesztési projekt is. Egy tipikus építészeti projekt lehet egy létesítmény megtervezése majd felépítése. Gépészeti projekt lehet egy fegyverfejlesztés. Az építészeti projektek során statikusok, külső és belső építészek finomítják, vizsgálják át a terveket, míg végül egy elfogadott komplett terv születik, mely alapján felépül a létesítmény. Egy gépészeti projektben hasonlóan először a tervek készülnek el, majd a tervek alapján készül el az eszköz, melynek tulajdonságait laboratóriumi körülmények között vizsgálják. A felmerülő hibákat a tervek finomításával és újabb prototípusok készítésével javítják.

A fenti példákban kézzelfogható, fizikailag létrejövő végeredményt kapunk, az építészet kapcsán egy kész épületet, egy eszközfejlesztés esetében egy prototípust. Az informatikai projektek a kézzelfoghatóság szempontjából szerteágazóak lehetnek. Egy szerverpark beüzemelése karakterisztikáját tekintve építészeti projekthez hasonlítható, tervezés, megvalósítás, átadás (tesztelés). Egy informatikai rendszer üzemeltetése hasonlítható egy gépészeti projekthez, ahol létező fizikai és létező szoftver rendszereket hangolnak össze, szabnak testre, majd az együttes működésüket tesztelik, egyfajta prototípusként. Az informatika egy különleges és izgalmas ágazata a szoftverfejlesztés, mely jellegét tekintve eltér az imént említett informatikai projektektől. Az informatikában is évtizedeken át széleskörűen alkalmazták a vízesés modellt[70] [94], amely az alábbi lépéseket azonosítja a projekt megvalósítása során:



17. ábra Eredeti vízesés modell. Forrás: [94]

A fenti statikus modell feltételezi, hogy minden produktív lépés végterméke egy teljes értékű dokumentum vagy megoldás, amelyet a későbbiekben már nem kell módo-

[70] Vízesés modell – angolul: Waterfall model. Léteznek olyan szakemberek, akik egyenesen tagadják a módszer létezését szoftverek esetében [95] és félreértelmezésként aposztrofálják annak említését is.

sítani. Ez a modell megfelelő lehet egy szoftver alap architektúrájának vagy prototípusának kialakításához, de nehézkesen vagy egyáltalán nem alkalmazható egy változó szoftverre, amellyel kapcsolatban folyamatosan igények merülnek fel a felhasználók, illetve a megrendelő részéről.

Úgy gondolom, hogy itt fontos megjegyezni, hogy minden szoftver esetében két különálló egységet azonosíthatunk: architektúra és funkcionalitás. Az architektúra kialakításhoz a vízesés modell célszerű lehet, azonban manapság ezek a feladatok ritkábban fordulnak elő, mert az informatikai ipar kész architektúrákat biztosít, ezekre példa a .NET Framework [96] vagy a Java Enterprise Edition [97].

Az architektúra kapcsán fontos előre tudni, hogy milyen fizikai eszközökön fog az adott szoftver futni, milyen számítási teljesítménnyel lehet kalkulálni, milyen fizikai környezetben fogják alkalmazni a kifejlesztendő megoldást. Ezen az alkalmazási szinten elengedhetetlen a lépcsőzetes tervezés – megfelelő választás lehet a vízesés modell, azzal a megkötéssel, hogy az elkészülő végtermék egy futtató környezet vagy egy keretrendszer, mely a későbbiekben alapja lehet bármilyen alkalmazásnak a legegyszerűbb oktatói programok és a nagyvállalati szoftverek [98] széles spektrumán. Napjaink alkalmazott szoftverei a két véglet közötti számtalan egyedi alkalmazásra mutatnak példát.

Ha az architektúrát fixnek tekintjük, akkor az adott architektúra által biztosított virtuális térben az egyedi alkalmazások matematikai leképezések halmazaként is tekinthetőek, amelyekről akár matematikai módszerekkel is eldönthető, hogy megoldják-e a kitűzött feladatot. [99] Nagyon fontos megjegyezni, hogy az egyedi alkalmazások implementálása során már csak információval dolgoznak a szoftverfejlesztők, ez egy jelentős különbség más műszaki projektekhez képest. Az iménti felismerésből az következik, hogy valójában hatalmas szabadsággal rendelkeznek a programozók a szoftvert futtató virtuális környezetben. Ezért is különösen fontos a megfelelő módszerek, szabályok, technológiák alkalmazása a szoftverfejlesztési projektek során, és ez alól nem lehet kivétel a Magyar Honvédség sem.

A vízesés modell első ránézésre szimpatikus választás lehet katonai alkalmazási területeken. Követelmények meghatározása, funkciók implementálása, tesztelés és végeredményként létrejön az eredetileg kigondolt szoftver. Ha azonban alaposabban megvizsgáljuk a katonai alkalmazási területeket, legyen az stacioner vagy tábori rendszer, ha felhasználókkal, jogosultságokkal, adatbázisban tárolt adatokkal rendelkezik a kifejlesztendő szoftver, akkor az egy egyedi alkalmazás. A katonai alkalma-

zások esetében különös tekintettel oda kell figyelni a biztonsági, jogosultsági, megbízhatósági, fenntarthatósági, hibatűrési, katasztrófatűrési szempontoknak, azonban az informatika szemszögéből a katonai szoftverfejlesztés a nagyvállalati szoftverfejlesztés, ezen belül az egyedi alkalmazásfejlesztés egy speciális ágazata és ugyanazokkal a problémákkal kell megküzdenie, amelyekkel a civil szoftverfejlesztésben is találkozhatunk.

Napjaink szoftverei iránt egyre több és több igény merül fel minden alkalmazási területen, nem kivétel ez alól a katonai célú alkalmazás sem. Az igények túlnyomó részben a megvalósítandó üzleti funkciókra vonatkoznak. Az igények megnövekedett száma egy statikus projektmódszertan számára elhúzódó specifikációs időszakot jelent, ráadásul nagyon speciális szaktudást igényel.

A vízesés modell esetében a fejlesztés megkezdéséhez a teljes specifikáció előállítása szükséges, mivel a fejlesztés és a tesztelés alapja ez a dokumentum lesz. A legnagyobb probléma a hagyományos statikus specifikációs módszerekkel, hogy kizárólag írott formában emberi nyelven állítanak elő egy szabályrendszert, nincs direkt kapcsolat a szoftver virtuális világával. A követelményrendszert értelmezni kell és az értelmezés folyamata hibákat rejthet, a szoftver bonyolultsága hatványozottan növekszik a funkciók számához viszonyítva.

A modern kor szoftverei esetében fontos szempont a felhasználói élmény, a letisztult grafikus felület, melyekhez előre elkészített képernyőtervekkel kell rendelkezni egy statikus fejlesztési módszertan esetében is. Sajnos a felületek működését nem lehet meghatározni a rendszer által kezelt adatok nélkül, teljes körű követelményrendszerre van szükség, amit megrendelői oldalon elvégezni nehézkes feladat, későn jelennek meg az első eredmények, csak a fejlesztési, tesztelési időszak végén lehet visszacsatolás. Sok éves szoftverfejlesztői, rendszerszervezői tapasztalattal rendelkező szakemberek számára is nehézkes ez a fajta specifikációs módszer.

Ahogy a harcászatban újabb és újabb eszközök jelennek meg, úgy a szoftvernek is gyorsan le kell tudnia követni a változásokat. Ha van egy harcszimuláló szoftverünk, amelyet hagyományos harcászati eszközökre fejlesztettünk ki a vízesés modellt alkalmazva, akkor komoly fejtörést okozhat a drónok bevezetése a rendszerbe, mert a követelményeket külön kell meghatározni. Célszerű, ha a specifikálást azok végzik, akik az eredeti specifikációt létrehozták, holott lehet, hogy már nem is dolgoznak ott, ahol az eredeti dokumentum elkészült. A fejlesztést követően is problémákba ütközünk, újra kell tesztelni manuálisan az egész rendszert.

4.1 AZ AGILIS SZOFTVERFEJLESZTÉS ÉS A SCRUM

Az üzleti szférában, a civil szoftverfejlesztésben lassan húsz éves múltra tekint vissza egy kezdeményezés, mely új alapokra helyezi az informatikai rendszerek, különösen az egyedi szoftverek előállítását. Az a felismerés, hogy a követelményrendszer alacsony szintű meghatározásában a fejlesztő csapat is részt vehet, ha egy megfelelő üzleti tudással – akár katonai területeken is – rendelkező szakember részesévé válik a fejlesztési folyamatnak új dimenziókat nyitott a szoftverek előállításában. Az *Agilis Kiáltvány* [10] tizenkét alapelvet határoz meg ezzel kapcsolatban. Az első négy üzenete az, hogy a megrendelői elégedettséget a folyamatosan jó minőségben előállított szoftverrel lehet elérni, mely megköveteli az aktív megrendelői és beszállítói együttműködést.

Ezt követően a kiegyensúlyozott projektkörnyezet és az ütemezett előrehaladás fontossága kerül külön kiemelésre, ezekre azért van szükség, hogy a feladat végrehajtását végzők nyugodt és előre tervezhető munkakörnyezetben tudjanak dolgozni, így egyfajta plusz motivációt kapjanak.

Az utolsó négy alapelv a kreatív gondolkodást és a mérnöki szabadság fontosságát hangsúlyozza, abból kiindulva, hogy egy motivált közegben, ahol a megvalósítást végzők magukénak érzik a szoftvert, ott lehetőséget kell biztosítani a saját ötleteik, koncepcióik megvalósítására, amennyiben azok a projekt célját szolgálják.

A fenti gondolatokat Ian Sommerville lényegre törően foglalta össze: *„az agilis megközelítés a tervezést és a megvalósítást helyezi középpontba a szoftverfejlesztési folyamat során”* [100, 74. o.], ezzel szemben a terv alapú fejlesztési megközelítés szakaszokat és szakaszonkénti kimeneteket definiál, hasonlóan a korábban látott NATO SLCM folyamatokhoz.

A különböző agilis projektmódszertanok a *Scrum, Lean, Kanban, Scrumban* más és más projektmenedzsmentet igényelnek, ezek különálló tárgyalása meghaladja az értekezés kereteit, az *Agilis gyakorlati útmutató* [101] című projektmenedzsment könyv részletesen tárgyalja őket, a továbbiakban a Scrum módszertan áttekintése következik.

Az agilis kiáltvány nem definiál konkrét módszertant, csak az alapelveket fekteti le, több agilis módszertan is létezik, köztük az egyik legelterjedtebb a *Scrum*, amely az alábbi szerepköröket, szereplőket és feladatokat azonosítja:

Product owner / terméktulajdonos – „A projekt vezetésének központi szereplője, aki döntéshozási jogokkal is fel van ruházva, ezáltal egyszemélyes döntéshozóként felelős a kifejlesztendő funkciókért és azok elkészítésének sorrendjéért." [7, 15. o.]
Ezen felül a terméktulajdonos további feladatai:

1. Kapcsolattartás az ügyféllel
2. Az üzleti és technológiai követelmények összefogása
3. A minőség ellenőrzése a fejlesztés során
4. Együttműködés a Scrum további szereplőivel (ScrumMaster, Fejlesztőcsapat)

ScrumMaster – „Mindenki számára igyekszik bemutatni és felölelni a Scrum értékeit, alapelveit és gyakorlatát. Egyfajta edzőként lép fel, vezetői szintű folyamatokat határoz meg és segíti a Scrum csapatot, valamint a szervezet fennmaradó részét saját teljesítményének javításában" [7, 16. o.]
A ScrumMaster további feladatai:

1. Felmerülő belső és külső problémák megoldása
2. A Scrum hatékonyságának növelése
3. A Scrum-hoz szükséges környezet megszervezése

Fejlesztőcsapat – „A tradicionális szoftverfejlesztési megközelítés különböző szerepköröket tárgyal, például architekt, programozó, tesztelő, adatbázis adminisztrátor, grafikusfelület tervező és így tovább" [7, 16. o.]
A fejlesztőcsapat feladatai a Scrum-on belül:

1. Tervezési feladatok ellátása
2. Fejlesztési feladatok ellátása
3. Tesztelés elvégzése
4. Önszerveződés (agilis megközelítés)

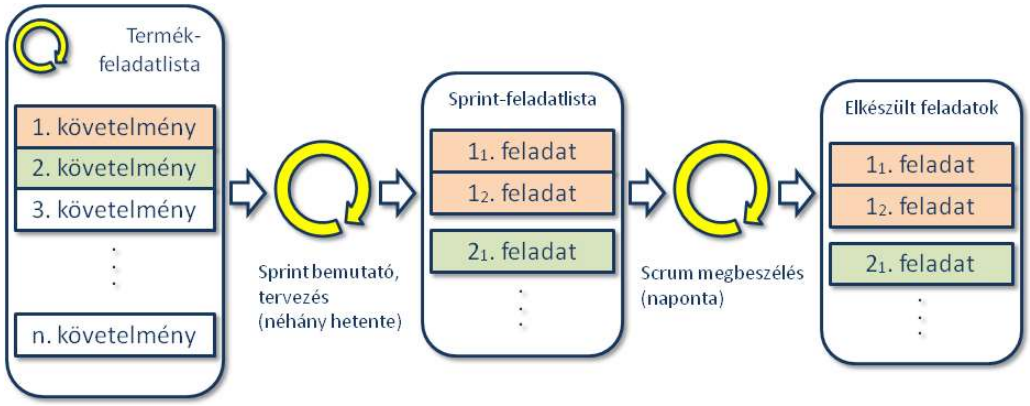
A szoftvertechnológiával foglalkozó irodalomban az ideális szoftverfejlesztő csapat résztvevőinek száma alapvetően eltérő, véleményem szerint egy erős technológia háttérrel rendelkező fejlesztés során ez a szám körülbelül 3-6 fő, tekintettel arra, hogy bizonyos feladatok automatizálhatóak, például: tesztelés, release management. Fontos megemlíteni a Scrum esetében, hogy nem javasolt az alkalmazása a 6-9 főnél nagyobb fejlesztőcsapatok esetében, ilyenkor újabb Scrum csapatok létrehozása ajánlott.
A Scrum egy iteratív szoftverfejlesztési módszertan [102], melynek kiindulási alapja a product backlog avagy a termék-feladatlista, amely a fejlesztés sorrendjében együttesen tartalmazza a termék előállításához szükséges még visszamaradt feladatokat. A

fejlesztés elkezdésének pillanatában tartalmaznia kell az addig azonosított összes követelményt. A módszertan lehetővé teszi a követelmények és a sorrend megváltoztatását, feladatok összevonását vagy szétbontását.

Az agilis kiáltvány 3. pontja szerint minél gyakrabban kell működő szoftvert szállítani, a Scrum értelmezésében ez az alapelv a fejlesztések sprintekbe szervezésében nyilvánul meg. Egy fejlesztési sprint egy rövid fejlesztési időszak (általában 1-3 hét), melynek a végén működő szoftvert kell előállítania a fejlesztőcsapatnak, ahol a sprint feladatai a product backlog tetején szereplő legmagasabb prioritású feladatokból kerülnek ki.

Lényeges különbség egy hagyományos módszertanhoz viszonyítva, hogy a sprint-feladatlista előállítása a terméktulajdonos és a fejlesztőcsapat közös feladata, melyet egy erre a célra szánt körülbelül 4 órás közös tervezésen tesznek meg. A tervezések során, a termék-feladatlista tetején szereplő követelményeket a fejlesztőcsapat értelmezi, majd kisebb feladatokra bontja, egyfajta megvalósíthatósági tervet készítve. A fejlesztési sprint során ezeket a konkrét feladatokat kell megoldania a fejlesztőcsapatnak.

Az agilis kiáltvány 1. pontja szerint az ügyfél elégedettségét működő szoftverrel kell kivívni, ennek módja a Scrum módszertan szerint az előző fejlesztési sprint eredményeinek bemutatása az ügyfél képviselőjének sprintről-sprintre. Amennyiben az ügyfél nem tud részt venni a demonstráción, a bemutató megtartása akkor is hasznos, mert így a fejlesztőcsapat minden tagja láthatja az előző időszak eredményeit.



18. ábra A Scrum folyamata (saját szerkesztés)

Ez a módszer önmagában nem jobb, mint a vízesésmodell és félreértelmezhető, valamint komoly kérdéseket vet fel, hogy valóban az a szoftver fog-e elkészülni, amelyet a megrendelő szeretett volna, mert a követelmények és a megfogalmazott elvárások folyamat közben változhatnak. Ha a megrendelő nem veszi ki a részét a rá háruló feladatokból, nem bocsájt rendelkezésre egy terméktulajdonost a projekt időtartam-

ára, akkor a Scrum csapat tehetetlenné válik megfelelő támogatás és követelményrendszer hiányában.

Természetesen a megfelelő szabályok betartásával a Scrum egy nagyon hatékony módszer is lehet, mellyel kimagasló eredmények érhetők el. A módszer alkalmazásának alapfeltétele a korszerű technológiai háttér, valamint a fejlesztőcsapat, a terméktulajdonos és a megrendelő agilis hozzáállása, ami az addig megszokott szervezeti berendezkedést alárendeli a projekt sikerének.

Példának okáért, ha egy katonai bevetéseket szimuláló szoftvert szeretnénk kifejleszteni, akkor szükséges egy olyan katonai szakértő, aki az adott terület fogalomrendszerét, terminológiáját, szabályait átadja a fejlesztők számára és folyamatosan részt vesz a fejlesztésben. Így az adott szakértő a kifejlesztendő szoftver terméktulajdonosává válik, ezt követően már kapcsolattartóként képviselni tudja a projektet a megrendelő felé. Az sem okoz gondot, ha terméktulajdonos a megrendelő oldaláról kerül a projektbe, de nagyon fontos, hogy a fejlesztés időtartamára a fejlesztőcsapattal napi szinten együtt tudjon dolgozni.

A termékre vonatkozó feladatlista megléte előfeltétele az agilis szoftverfejlesztési projekteknek, azonban a Scrum módszertan összességében annyit mond a tervezésekről, hogy a követelményeket fejlesztési feladatokra kell szétbontani. A kérdés az, hogy egy összetett szoftverfejlesztési projekt esetében elegendő-e csak a termék feladatlistájának karbantartása és frissítése? Minden alkalmazási területen lényeges minden részlet, ezek megértése, felkutatása már a sprint tervezések előtt is szükséges. A statikus módszertanok válasza erre, hogy a fejlesztés előtt körbe kell járni minden területet. A gyakorlat azt mutatja, hogy az implementáció során is nagyon gyakoriak a rendszer teljes egészére vonatkozó, teljes működést érintő kérdések, amelyekre a válaszok fogósak lehetnek még a szakértők számára is.

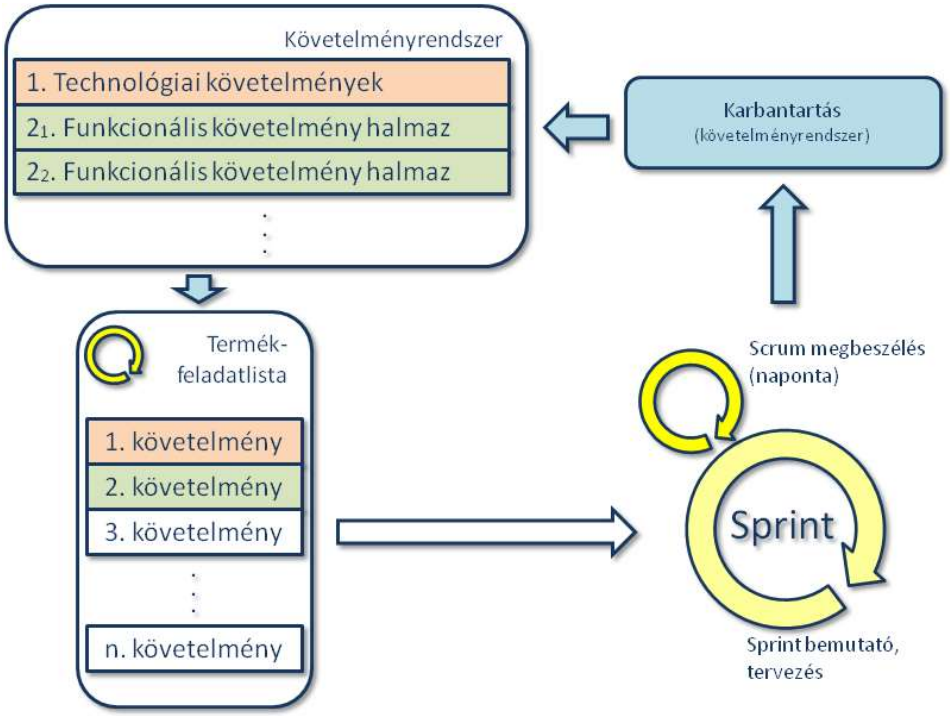
Grafikus felülettel rendelkező szoftverek esetében a képernyőtervek és az azok működését leíró magyarázó szövegek megfelelő kiindulási pontnak tekinthetők egy Scrum fejlesztési sprint tervezéshez. Felülettel nem rendelkező szoftverek esetében is a keletkező leírások elegendőek az implementálás megkezdéséhez.

A rendszeres sprint tervezések során a fejlesztők részéről nagyon hasznos és lényegi észrevételek hangozhatnak el az implementációval kapcsolatban, amelyek a konkrét fejlesztési feladatok meghatározásánál fontosak lehetnek. Ebben a 4 órás tervezésben a cél az, hogy jól átgondolt alacsony szintű követelmények és megvalósítási tervek

születsenek, ám az itt feltárt összefüggések visszahathatnak a teljes követelményrendszerre is.

Nem szabad elfelejteni, hogy a tervezést végző csapat része a terméktulajdonos is, aki szakmai szemmel vesz részt ezeken a megbeszéléseken. Az ő elsődleges feladata a feltárt problémák értelmezése, elemzése. A megoldások hatással lehetnek az egyes követelményekre vagy extrém esetben a teljes követelményrendszerre is. Az is elképzelhető, hogy egyéb szakmai területek bevonása szükséges az adott kérdés tisztázásához.

A gyakorlatban ez a módszer azt jelenti, hogy a sprint időszaka alatt a következő sprint előkészítő tervezése is megvalósulhat, valamint a termékre vonatkozó feladatlista vezetése, karbantartása is ekkor zajlik. A fejlesztéssel párhuzamosan a felkészülés megvalósítható a soron következő sprintre.

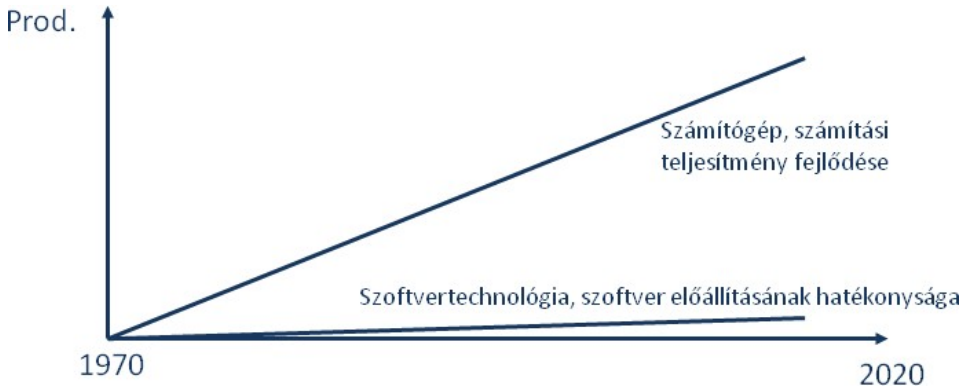


19. ábra Az agilis szoftverfejlesztési sprintek visszahatása az eredeti követelményrendszerre (saját szerkesztés)

A bemutatott fejlesztési folyamat pozitívumainak ellenére, egy katonai szakértő bevonása egy agilis szoftverfejlesztési projektbe önmagában nem elegendő ahhoz, hogy a projekt megfeleljen a speciális katonai minőségi elvárásoknak. A siker eléréséhez olyan szoftverfejlesztési módszertanra van szükség, amely ismeri a védelmi ágazat rendszerfejlesztéseinek jellegzetességeit, illetve ötvözi azokat a korszerű technológiákat, amelyek felhasználása napi szinten megvalósul az informatikai iparágban.

4.1.1 Korszerű technológiai lehetőségek

A folyamatosan gyorsuló számítógépek, a növekedő számítási kapacitás lehetővé teszi olyan szoftverfejlesztési platformok kialakítását, amelyek korábban elképzelhetetlenek lettek volna. A fejlesztett szoftvert tesztesetek tízezreivel, tesztforgatókönyvek százaival fedhetjük le és folyamatosan tesztelhetjük a magas minőség elérése érdekében. Szoftver alapú szolgáltatások esetében az automatizált tesztelés mellett a szimulált átadási és üzembe helyezési folyamatok is kialakíthatók, amelyek segítségével a fejlesztett szoftverek, a konfiguráció és az üzemeltetési környezet kapcsolata is vizsgálhatóvá válik az éles üzembe helyezés előtt. A különböző automatizált folyamatok elkészítése ugyan sok időt igényel a fejlesztők részéről, de ezt követően ezek a korszerű megoldások szerves részévé válnak a szoftver fejlesztői környezetének – új fejlesztési platformokat képezve. Az automatizált tesztelés és az átadási folyamat szimulációja egyaránt javítja a fejlesztett rendszer minőségét, mert a különböző területekről jövő hibák, hamarabb napvilágra kerülhetnek, javításuk még a fejlesztések kora fázisában lehetővé válik, valamint a változtatási igények hatásainak ellenőrzése is analóg módon megvalósítható. A 20. ábra szemlélteti az elmúlt évtizedekben végbement számítási teljesítménynövekedést és a szoftvertechnológia fejlődését.



20. ábra A hardverek sebességének és a szoftvertechnológia hatékonyságának fejlődése (saját szerkesztés)

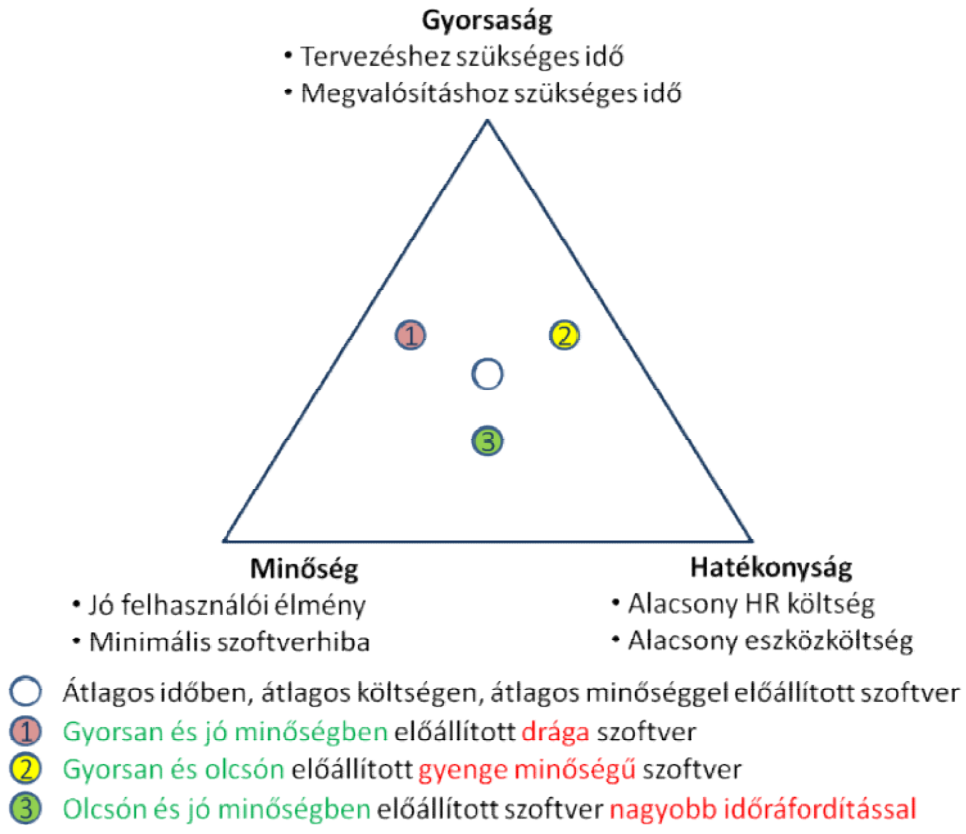
Amíg a gépekről elfogadja a tudományos világ, hogy a Moore törvény [103] alapján az egységárra jutó számítási teljesítmény egyre gyorsabban növekedik, vagyis az integrált áramkörökben a tranzisztorok száma tizennyolc havonta megduplázódik, addig az emberek, konkrétan az emberi agy ugyanolyan kognitív képességekkel rendelkezik, mint ötven évvel ezelőtt. A jövőben sem várható szignifikáns változás a szoftvertechnológia hatékonyságában az emberi tényező oldaláról – a fejlesztők programozási képessége konstansként tekinthető.

A szoftvertechnológia hatékonyságán kizárólag az új projektmódszertanok és a fejlesztést támogató eszközök bevonása javíthat. Olyan megoldásokra kell gondolni, amelyek segítségével kiértékelhetők, ellenőrizhetők, tesztelhetők, szimulálhatók olyan folyamatok, amelyeket korábban a számítógépek szűkös képességei miatt nem lehetett elképzelni. Az agilis szoftverfejlesztési technikák koncepciója az, hogy a működőképesség és a működés automatizált ellenőrzése az egyik tartó pillére a sikeres szoftverfejlesztési projekteknek.

A hatékonyságot gazdasági és projekt oldalról vizsgálva, míg a korszerű eszközök beszerzése egyszeri költségként jelenik meg egy fejlesztési projektre levetítve, addig a fejlesztési projekt szereplőinek bérköltsége folyamatos kiadás. Ebből a megállapításból az következik, hogy hosszú távon luxus a szoftverfejlesztési projektek egy kalap alá vétele az egyéb informatikai projektekhez, mert a leghatékonyabb módszerek alkalmazásához más projektstruktúrára és eszközparkra van szükség.

4.1.2 A Scrum helye a projektmenedzsment háromszögben

A következő ábra a közgazdaságtanban ismert projekt menedzsment háromszög [104] szoftverfejlesztésre vetített képét mutatja be.



21. ábra Szoftvertechnológiai projektmenedzsment háromszög (saját szerkesztés)

1. megközelítés – Az üzleti szférában elképzelhető, hogy egy vállalkozás extra erőforrást biztosít egy szoftverfejlesztésre, azért hogy piaci előnyre tegyen szert. Ebben az esetben az adott megrendelő nagy kockázatot vállal, mert a rövid határidőre bevállalt teljesítés bizonytalansági tényezőket rejthet magában. Állami megrendeléseknél a magas kockázat és a magas költségek együttesen kizárják az ezzel a megközelítéssel elkészített szoftver beszerzésének lehetőségét. A védelmi ágazat esetében elképzelhető, hogy egy kiemelt fontosságú fejlesztés esetében a költségek nem feltétlenül képeznek akadályt, azonban a felmerülő kockázatok mindenképp óvatosságra intenek.
2. megközelítés – Az államigazgatáson kívül, a piaci szférában elképzelhető, hogy egy szereplő gyorsan szeretne megjelenni a piacon egy pilot projekttel és nem szeretne túl sok pénzt kockáztatni, annak reményében, hogy egy gyorsan előállított szoftver további bevételeket hoz és a beérkező forrásokból a létező szoftver hibái javíthatók lesznek vagy esetleg egy új szoftver előállítható a bevételekből. Ez a fajta megközelítés állami szinten nem jöhet számításba, alapvető elvárás a jó minőségű, megbízható szoftver, a katonai alkalmazás esetén ez a követelmény pedig különösen igaz.
3. megközelítés – Piaci szférában is vannak olyan területek, ahol a megbízhatóság és a hatékonyság élvez elsőbbséget. Olyan területekre kell gondolni, melyeknek informatikai támogatása már jelenleg is megoldott, de indokolt a modernizáció, ebben az esetben van idő a fennálló problémák feltárására, különböző tervek kidolgozására. A tervezést követően lehetőség van a továbbfejlesztési lehetőségek priorizálására, megvalósításuk sorba rendezésére. Ha az állami szervezetek működését tekintjük, hosszabb átfutási időket tapasztalhatunk, melynek egyik fő oka, hogy ebben a szférában a döntéshozatalnak megvannak a jogszabályi követelményei, az előkészítési, a jóváhagyási folyamatok sok időt igényelnek. Ha abból indulunk ki, hogy megrendelői oldalon az idő a legkevésbé kritikus tényező – persze ésszerű keretek között – akkor egy ennek megfelelő fejlesztési módszertan választása célszerű lehet. Általában igaz az agilis szoftverfejlesztésre, hogy többlet energiaráfordítást követel meg a megrendelőtől a követelményrendszer összeállítása és a projekt követése során. Ezzel párhuzamosan elmondható a fejlesztőkről is, hogy a megvalósítás és továbbfejlesztés időszakában többlet energiaráfordítással készítik el a rendszereket. A plusz időráfordítás fenntartható és jó minőségű

rendszerek elkészültét jelenti, amelyek rendelkeznek a továbbfejleszthetőség képességével – itt meg kell említeni, hogy a szoftver előállításának folyamata valóban lassabb, mint egy rapid alkalmazás-fejlesztés (RAD) [105] esetén. Úgy gondolom, hogy az agilis szoftverfejlesztés karakterisztikáját tekintve megfelelő helyet foglalja el a projekt menedzsment háromszögben az államigazgatás és ezen belül a Magyar Honvédség számára is, mert a megrendelő és a beszállító tevékenysége párhuzamosan végezhető akár a tervezési és megvalósítási szakaszokban is, valamint megrendelői oldalon jó minőségű szoftver és kiszámítható költségek jelennek meg a fejlesztés során. Beszállítói oldalról a kiszámítható környezet, az alacsony kockázat, összességében a jó együttműködés csökkentheti a fejlesztés költségeit is.

4.2 MILITARY SCRUM – KATONAI CÉLÚ SZOFTVERFEJLESZTÉS

A továbbiakban feltételezem, hogy olyan katonai célú szoftverfejlesztési projekt sikeres végrehajtása a feladat, amely karakterisztikáját tekintve a 3. megközelítésben bemutatott szempontrendszerhez illeszkedik. Ebből a szempontból a Scrum szoftverfejlesztési módszertan alkalmazási lehetősége adott a katonai célú szoftverfejlesztési projektek fejlesztési szakaszai során – ámbár, a Scrum módszertan által nyújtott eszközök nem felelnek meg a kutatás során levezetett katonai követelményeknek.

Ahhoz, hogy az agilis módszerek megjelenhessenek a védelmi ágazat szoftverfejlesztési projektjeiben, a speciális katonai követelményeket kielégítő módszertanra van szükség. A továbbiakban a kutatási célként kitűzött módszertan meghatározása következik a módszertani (M), szoftvertechnológiai (S), dokumentációs (D), minőségbiztosítási (Q), információbiztonsági (I) és követelményelemzési (R) elvárásoknak való megfelelés mentén.

Az egyszerűbb hivatkozhatóság kedvéért a kutatás céljaként kitűzött módszertan *Military Scrum* néven szerepel ezt követően az értekezésben. A Military Scrum nevéhez hűen átveszi a Scrum módszertan szerepköreit: a fejlesztőcsapatot, a termék tulajdonost, illetve a ScrumMaster-t. Emellett a korábban bemutatott iteratív fejlesztési technikát veszi kiindulási alapként, kibővítve azt a katonai fejlesztések számára meghatározott szoftvertechnológiai szempontból is értelmezhető életciklus szakaszokkal.

A Military Scrum folyamatának és dokumentációs sablonjainak meghatározása előtt érdemes egy pillantást vetni a *Kiáltvány az agilis szoftverfejlesztésért* című dokumentumra, amelyben az érintett szoftvermérnökök korszakalkotó gondolatokat fogalmaztak meg 2001-ben. Az általuk lejegyzett gondolatok folyamatos szem előtt tartása teszi lehetővé, hogy a Military Scrum megőrizze az agilis szoftverfejlesztés vívmányait és mégis meg tudjon felelni a szigorú védelmi követelményeknek is.

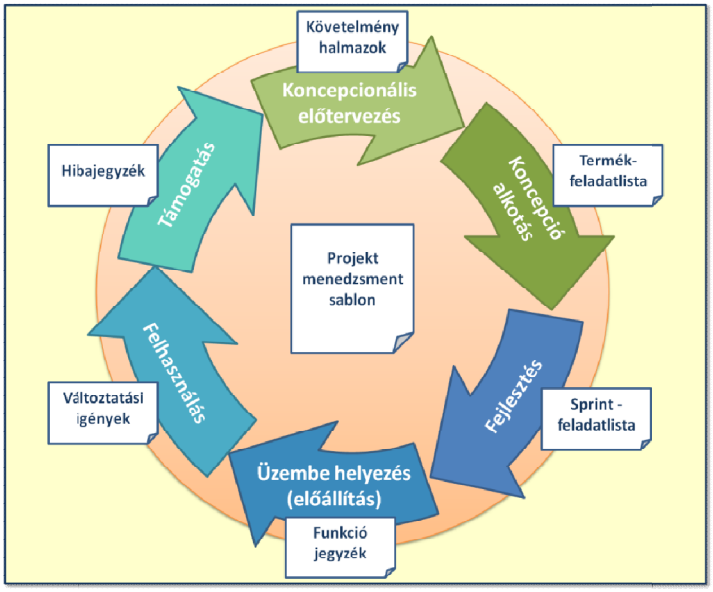
"A szoftverfejlesztés hatékonyabb módját tárjuk fel saját tevékenységünk és a másoknak nyújtott segítség útján. E munka eredményeképpen megtanultuk értékelni:

- ***Az egyéneket és a személyes kommunikációt** a módszertanokkal és eszközökkel szemben*
- ***A működő szoftvert** az átfogó dokumentációval szemben*
- ***A megrendelővel történő együttműködést** a szerződéses egyeztetéssel szemben*
- ***A változás iránti készséget** a tervek szolgai követésével szemben*

Azaz, annak ellenére, hogy a jobb oldalon szereplő tételek is értékkel bírnak, mi többre tartjuk a bal oldalon feltüntetetteket." [10]

A Military Scrum szoftverfejlesztési módszertan meghatározása során egy olyan szabványos meder kialakítása a cél, amely egyensúlyt teremt az agilis megközelítés és a szabályozott projektkörnyezet között. Ez a gondolatmenet azt eredményezi, hogy a Military Scrum célirányosan próbálja csökkenteni a dokumentumok, a folyamatlépések és az előírt adminisztratív lépések számát egy adott szoftverfejlesztési projekt végrehajtása során. Ugyanakkor mégis szem előtt tartja a katonai rendszerekkel szemben támasztott minőségi és dokumentációs elvárásokat a szoftver alapú szolgáltatásokra értelmezhető teljes életciklus alatt. Így az előállítandó dokumentumok száma racionalizálható, tartalmuk a szoftverfejlesztési projektek igényeihez igazítható. A 22. ábrán a Military Scrum szakaszai és a szakaszok során keletkező dokumentumok láthatók. A projekt menedzsment sablon (a továbbiakban: PMS) az 1. sz. mellékletben található, célja a módszertan helyes alkalmazásának támogatása.

A 22. ábra alapján az is látható, hogy a módszertan nem tesz különbséget a teljesen új szoftverfejlesztési projektek, a változtatási igények és a hibák kezelése között. Minden esetben el kell végezni a koncepcionális előtervezési, koncepcióalkotási és fejlesztési feladatokat is, ezek részét képezik az iterációknak. A továbbiakban a Military Scrum hat életciklus szakaszt tartalmazó iteratív folyamata fejlesztési menetként kerül hivatkozásra.



22. ábra A Military Scrum folyamata és kezelt dokumentum típusai (saját szerkesztés)

Ez azt jelenti, hogy a Military Scrum minden esetben hat életciklus szakaszon átmenetelve éri el a fejlesztett szolgáltatás új funkciókat tartalmazó üzembiztos állapotát. A Scrum-ból ismert fejlesztési sprint fogalma kizárólag a fejlesztési szakaszon belül értelmezett.

Ssz.	Szerepkör	Röv.	Leírás
1.	Megrendelő	Cust	A projekt által érintett katonai szakterületek képviselete, döntéshozatali jogosultságokkal.
2.	Termék-tulajdonos	PO	A fejlesztőcsapattal a Military Scrum-on belül együtt tevékenykedő katonai szakértő.
3.	Fejlesztő-csapat	Dev	A szükséges tervezői, fejlesztői képességek birtokában lévő fejlesztést végző csapat.
4.	Üzemeltetés	Ops	A rendszer üzembe helyezésért és éles üzemű működtetéséért felelős személyzet.
5.	Projekt-menedzser	PM	A Military Scrum helyes végrehajtásáért felelős személy, aki ellátja a hagyományos ScrumMaster feladatait és vezeti az 1. számú mellékletben meghatározott projekt menedzsment dokumentációs sablont.
6.	Minőség-biztosító	QA	Folyamatosan ellenőrzi a különböző életciklus szakaszokon belül a projekt menedzsment sablon által előírt tevékenységek végrehajtását, ha hiányosságokat talál, akkor azt jelzi a PM felé.

9. táblázat Military Scrum szerepkörei és azok leírása (saját szerkesztés)

A projektben résztvevő szereplők kijelölése a lehető legkorábban ajánlott, mert a projekt sikere csak abban az esetben garantálható, ha a korai szakaszokban meghatározásra kerülnek a szakmai és a technológiai követelmények. Ez azért fontos, mert így mihamarabb rendelkezésre állhat a fejlesztett szoftvernek legalább egy prototípusa.

A megfelelő személyi állomány gyors delegálásához agilis megközelítésre van szükség minden érintett részéről, ha ez megtörtént, akkor a működést lehetővé tévő rendszerek felállítása is lehetségessé válik, a megfelelő szereplők támogató feladatokhoz történő rendelésével.

A Military Scrum szoftverfejlesztési módszertan célja a katonai felhasználású, szoftver alapú szolgáltatások (SaaS) előállításának és hosszú távú fenntarthatóságának szoftvertechnológiai támogatása. Ezzel a megközelítéssel a Military Scrum a NATO SLCM kontextusában alkalmazhatóvá válik a szoftver intenzív katonai képességek kialakításához. A Military Scrum a szoftver alapú szolgáltatás komponenseit az alábbi ábra szerint definiálja.

Komponens	Szerepkör
Rendszer	
Fejlesztett szoftver	PO, Dev, PM, QA
Működést lehetővé tévő rendszerek	
Projektvezetés	
Projektmenedzsment	PM
Minőségirányítás	QA, PM
Szakértői támogatás	
Technikai támogatás	DevOps, QA
Katonai és fejlesztői szakterületek által nyújtott támogatás	PO, Dev, QA
Műszaki támogató rendszerek	
Fejlesztési és tesztelési eszközök és környezetek	DevOps, QA
Üzemeltetési eszközök és környezetek	Ops, QA

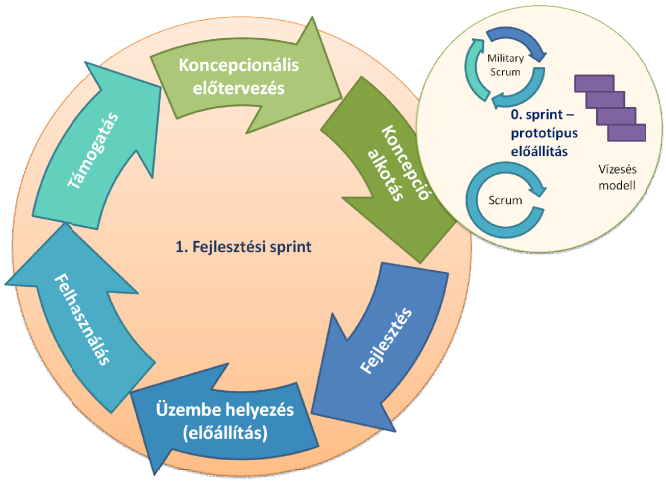
23. ábra Szoftver alapú szolgáltatások fejlesztéséhez szükséges technikai és személyzeti komponensek a Military Scrum szoftverfejlesztési módszertanban (saját szerkesztés)

A klasszikus Scrum módszertan nem ismeri a projektvezetés, a szakértői támogatás és a műszaki támogató rendszerek fogalmait. A Military Scrum a NATO SLCM

Rendszer Koncepciójához igazodva a rendszert egy fejlesztett szoftverként definiálja, amely a működést lehetővé tévő rendszerekkel együtt alkotja a szoftver alapú szolgáltatást – a kialakítandó katonai képességet reprezentáló rendszert, a SOI-t.

A rendszer és a működést lehetővé tévő rendszerek esetében is értelmezett az érettségi szint fogalma a Military Scrum alkalmazása során. A módszertan megköveteli, hogy a minőségbiztosítási eljárások alkalmazásával minden szakaszban álljanak rendelkezésre a megfelelő támogató rendszerek, valamint a hozzájuk tartozó szolgáltatások minden esetben érjék el a megfelelő és egyben elégséges érettségi szintet. Ehhez a Military Scrum előírja a műszaki és a személyzeti komponensek monitorozását. Amennyiben bármelyik komponens kifogásolható, azaz nem felel meg az 1. számú mellékletben megfogalmazott követelményeknek, akkor az a teljes szoftver alapú szolgáltatás érettségi szintjére is kihatással van, azt nem megfelelőre változtatja. Ilyenkor a projektvezetés elsődleges feladata a probléma feltárása és a szükséges lépések megtétele.

A szoftverfejlesztési projekt elindításakor 1. fejlesztési menetről beszélünk, ekkor a koncepcionális előtervezés és a koncepcióalkotás képezi a menetelés legfajsúlyosabb részét. Amennyiben a projekt keretei lehetővé teszik, a koncepció alkotás során legalább egy – akár több prototípus – előállítása is feladat lehet, amelyek egyfajta 0. sprintként jelennek meg a folyamatban. A 0. szoftverfejlesztési sprintek rövidségük és céljuk miatt tetszőleges módszertan alkalmazásával megvalósíthatók: vízesés modell jellegű lépcsőzetes technikák, agilis módszertanok (Scrum, Kanban, stb.), illetve a Military Scrum módszertan által javasolt sprint megvalósításával – különös tekintettel arra, hogy a pályázók, rendelkezésre álló fejlesztőcsapatok, mint potenciális beszállítók, fejlesztők is megvalósíthatják azt.

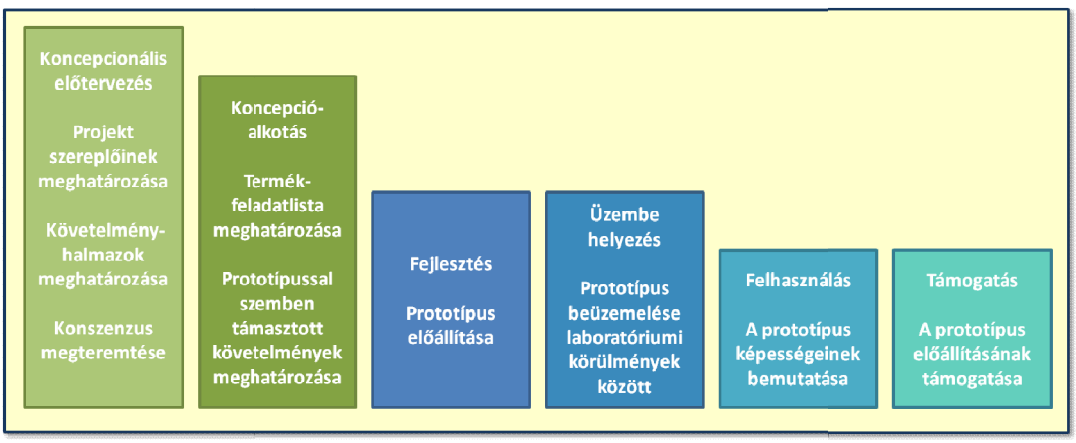


24. ábra Prototípus előállítás a Military Scrum 1. fejlesztési menete során (saját szerkesztés)

Ha a pályázó a prototípus előállítás mellett a Military Scrum módszertan helyes alkalmazását is tudja prezentálni, akkor a megrendelői oldal nagyobb bizonyossággal választhatja ki a nyertest a szoftverfejlesztési projekt teljes megvalósításához, a szoftver alapú szolgáltatás kialakításához.

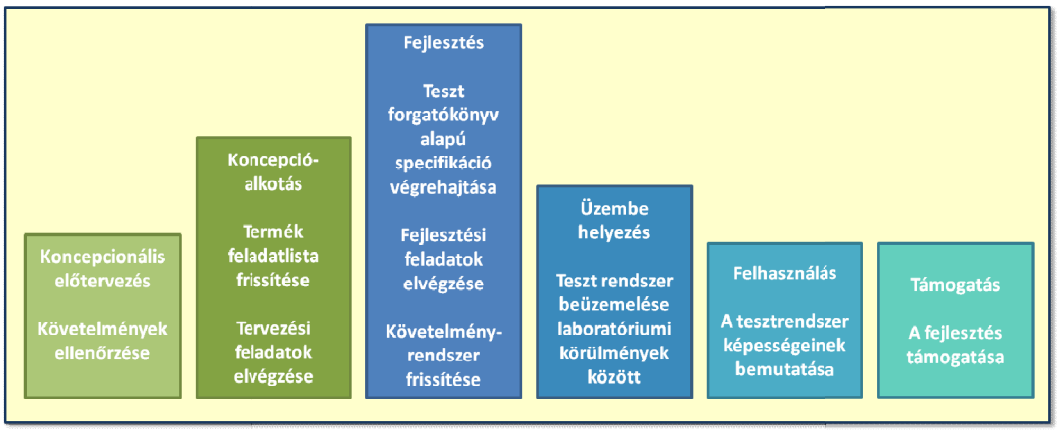
Az 1. koncepcióalkotási szakaszt követően – jó esetben működő prototípus megléte mellett, annak továbbfejlesztésével – folytatódhat az 1. menet fejlesztési szakasza. Abban az esetben, ha az ideális körülmények nem adottak, nem áll rendelkezésre koncepcionális szinten elérhető prototípus, akkor arra kell törekedni, hogy igazoltan működő technológiai alapok kerüljenek meghatározásra műszaki követelményként és a termék feladatlista összeállításához kellő szakértelem társuljon. Ilyenkor elengedhetetlen, hogy olyan szakértői támogatás kerüljön bevonásra a koncepcióalkotási folyamatba, amely valamilyen működő szoftver alapú szolgáltatás fejlesztését sikeresen végzi. Ekkor a koncepcióalkotás műszaki oldalról empirikus módszereken alapszik ugyan, de a prototípus előállítási folyamattal hasonló hatásfokkal rendelkezik. A fenti módszerrel a koncepcióalkotás kimenete egy már ismert alapokon nyugvó műszaki követelményrendszer, amelyhez a szoftverfejlesztési projekt jellegéhez igazodó funkcionális követelmények kapcsolódnak.

Működő prototípus hiányában az első korlátozott funkcionalitással működő szoftver verzió a Military Scrum további szakaszai alatt (üzembe helyezés, támogatás) áll elő és kerül laboratóriumi körülmények között felhasználási (tesztelési) szakaszba. Analóg módon azzal, mintha a prototípus előállítása a Military Scrum módszertan segítségével valósult volna meg. Az alábbi ábrán az 1. menet során végrehajtott tipikus feladatok láthatók, ha a prototípus előállítás is a módszertan segítségével történik. Az oszlopok magassága az egymáshoz viszonyított feladatok nagyságrendjét is szemlélteti. Az 1. menetben a kezdeti követelményrendszer összeállítása az elsődleges cél.



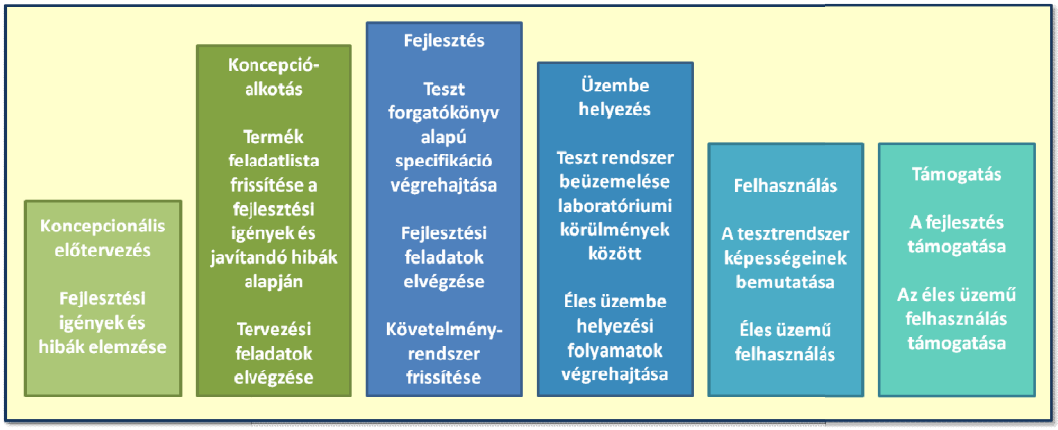
25. ábra A Military Scrum 1. menetének feladatai (saját szerkesztés)

Az 1. menet utáni időszak két részre bontható, aszerint, hogy a fejlesztett szoftver éles üzembe helyezése kezdetét vette-e már vagy sem. Amíg az éles üzembe helyezés nem történik meg, átadás előtti időszakról beszélünk, ekkor a Military Scrum meneteinek feladatai a 26. ábra szerint alakulnak.



26. ábra A Military Scrum átadás előtti meneteinek feladatai (saját szerkesztés)

Az átadás utáni időszakban két tényező befolyásolhatja a sprintek alakulását, ezek a felhasználás során felmerülő igények, illetve a rendszerben tapasztalt hibák. Ekkor is szükséges minden szakasz ismételt végrehajtása a konzisztens követelményrendszer fenntartása végett. A folyamatot a 27. ábra szemlélteti.



27. ábra A Military Scrum átadás utáni sprintjeinek feladatai (saját szerkesztés)

Az 1. menet esetén a koncepcionális feladatok a leghangsúlyosabbak, azonban a rendszer átadását – a szoftver alapú szolgáltatás üzembe helyezését – követően a súlypontok eltolódnak és jelentős feladatok hárulnak a tervezési, fejlesztési és üzemeltetési feladatokat ellátó rendszerekre.

A Military Scrum szoftverfejlesztési módszertan minden esetben kezeli a szakaszokban végrehajtandó tevékenységeket, a mellékelt projekt menedzsment sablon kitér a különböző időszakokban végrehajtott fejlesztési menetek feladataira és a minőségbiztosítási eljárásokra is.

4.3 KÖVETELMÉNYRENDSZER MEGHATÁROZÁS

A modern kor szoftvere bonyolult és egyre bonyolultabbá válik, a régi és az újonnan létrejövő rendszereket össze kell kapcsolni, virtuális világ épül a védelmi ágazatban is. Az implementáció során új módszerekre van szükség, amikor egy újabb területet akarunk leképezni, kell egy kiindulási alap. A Scrum szerint ez a product backlog, avagy termék feladatlista, arról nem szól az eredeti módszertan, hogyan lehet ezt a feladatlistát összeállítani.

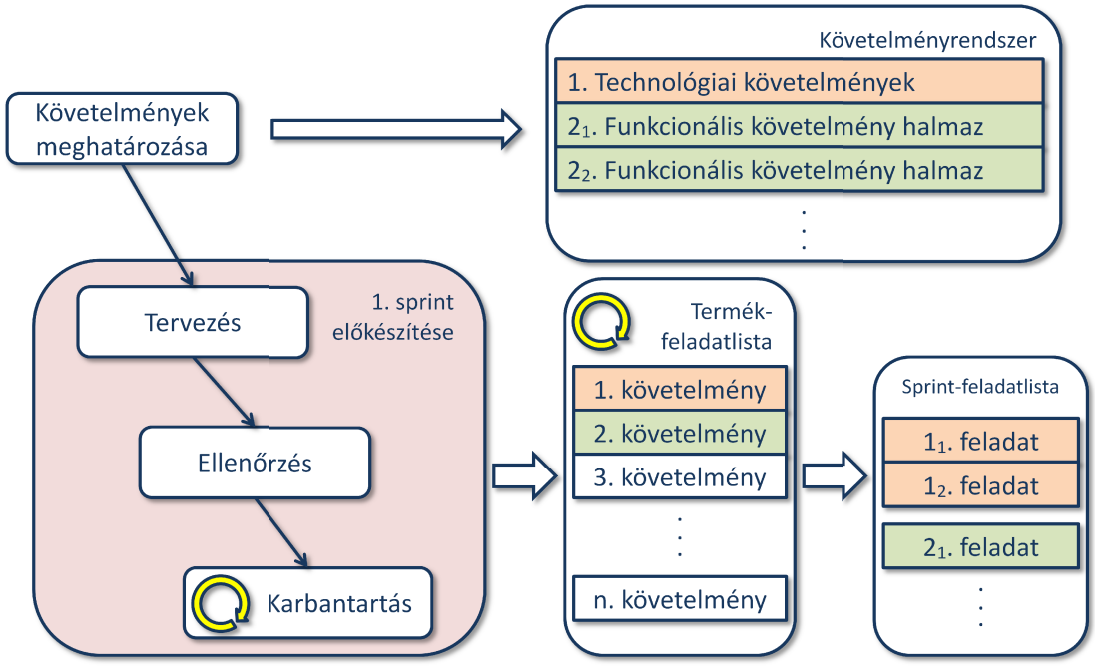
Nyilvánvalóan a szoftverfejlesztési projektek velejárója a specifikáció, a műszaki dokumentáció, úgy megrendelői, mint beszállítói oldalon. Szoftverfejlesztési módszertantól függetlenül elmondható, hogy a megrendelőnek egy fejlesztés megkezdése előtt le kell jegyeznie, hogy mit szeretne. A dokumentum előállítása lehet egy lineáris folyamat végeredménye, de előállhat egy iteratív folyamat végtermékeként is.

Egy dokumentáció esetében nehéz minőségi mutatót meghatározni, egyáltalán hogyan lehet a fejlesztést végző szakterület vagy a beszállító nélkül eldönteni, hogy minden lényeges követelmény meghatározásra került-e? Szem előtt tartandó, hogy a fejlesztési projektet előkészítő specifikációs időszaknak, ahogyan van eleje megrendelői oldalon, úgy vége is kell, hogy legyen. Célszerű ezt a tevékenységet minél rövidebb idő alatt elvégezni, a követelmények finomságát és a határidőt is megszabni.

A vízesés modell egy speciális alkalmazási területe lehet a termékre vonatkozó feladatlista előállítása. Ebben az esetben a cél egy olyan lista elkészítése, amely tartalmazza a leendő szoftver magas szintű áttekintését az azonosított feladatokra lebontva. Az elsődleges feladat a teljes probléma feltérképezése, a kapcsolódó szakmai területek azonosítása, majd a probléma feladathalmazokra és azon belül magas szintű feladatokra bontása. A verifikáció és a tesztelés során ekkor azt kell ellenőrizni, hogy a feladatok szétbontása valóban lefedi-e a teljes problémát. Az esetleges tervezési kérdések, problémák a koncepcionális tervezés szintjén is dokumentálhatók.

A következő ábrán a koncepcionális előkészítés támogatására szolgáló hibrid vízesés modell szerepel. A vízesés modell által nyújtott lineáris folyamat segítségével egy kezdeti követelményrendszer megtervezhető, amelynek ellenőrzését követően, lehetségessé válik annak karbantartása is. Mindez hogyan valósul meg? A karbantartás lépésében már alkalmazhatók a megrendelői konszenzust elősegítő, döntéshozatalt támogató módszerek, amelyek segítségével fel lehet oldani a követelményekben mutatkozó ellentmondásokat. A felvázolt folyamat a Military Scrum 1. sprintje során

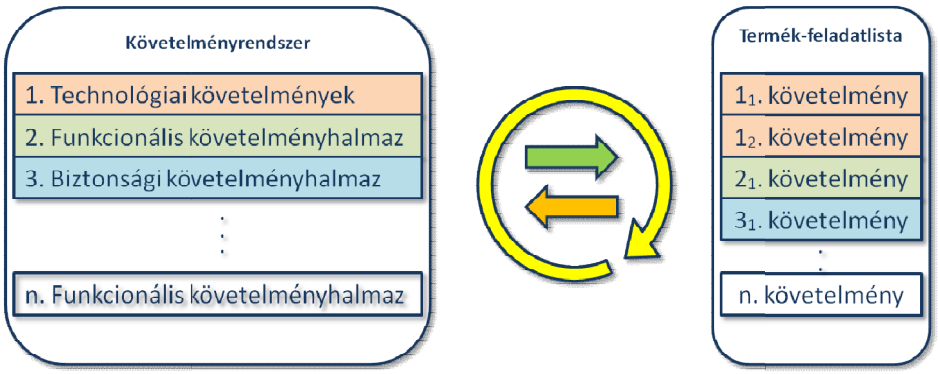
alkalmazható speciális, tervezést támogató módszer, amely mellőzi az implementálást, célja a kezdeti koncepció kialakítása és dokumentálása.



28. ábra A vízesés modell alkalmazásának lehetősége a Military Scrum koncepcionális előtervezési szakaszában (saját szerkesztés)

A fenti ábrán a termék feladatlistához hasonlóan a karbantartás mellett is látható egy iterációt jelző magába forduló nyíl, ami azt jelzi, hogy a dokumentált koncepció karbantartása is folyamatos, a teljes élettartam alatt végzendő tevékenység a vízesés végén, amely az igények és a követelményhalmazok frissítését, majd azok ellenőrzését takarja – így biztosítva a követelményrendszer konzisztenciáját.

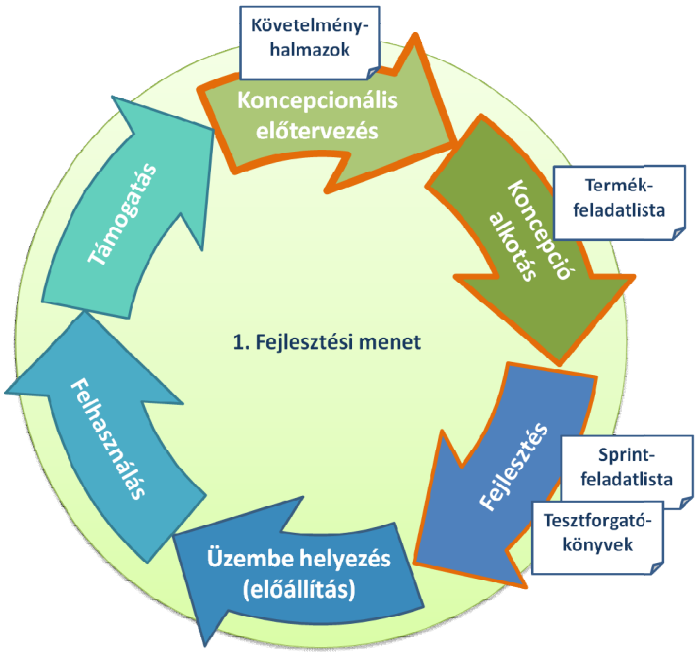
A megrendelői konszenzust elősegítendő, a Military Scrum kiegészíti a Scrum feladatrendszerező és feladatrendező módszerét, azzal hogy a követelmények elemzését a követelményhalmazok összeállításával kezdi meg. A módszertan az egyes követelményhalmazokat hozzárendeli a követelmény formálójához.



29. ábra A Military Scrum követelményhalmazainak és követelményeinek kapcsolata (saját szerkesztés)

A követelmény-karbantartás iteratív folyamatát a 29. ábra mutatja be a Military Scrum sprintjei során – a módszer helyes alkalmazásakor a követelményrendszer ellenőrzése és frissítése rendszeres tevékenység. Az ábrán szereplő eljárás ismétlődő végrehajtásának azért van jelentősége, mert elképzelhető, hogy egy hierarchikus szervezeten belül több szereplőhöz köthető a követelményrendszer és ellentmondásokat tartalmazhat. Az egyes témakörökben keletkező követelmények tisztázása alapvető fontosságú, mert a követelmények minősége jelentősen befolyásolja a fejlesztési projekt kimenetelét.

A Military Scrum 1. fejlesztési menetének koncepcionális előtervezési és koncepcióalkotási szakaszaiban a megrendelő által biztosított katonai szakterületek képviselőinek bevonásával megvalósítható a koncepcionális kérdések tisztázása és a magas szintű tervezés. Ezt követően a kezdeti – ugyanakkor a lehetőségekhez mérten legbővebb – követelményrendszerrel elindulhat a fejlesztés.



30. ábra A Military Scrum specifikációs feladatokat tartalmazó szakaszai (saját szerkesztés)

A fenti ábrán látható, hogy a módszertan alkalmazása során az egyes szakaszokban különböző tervezési dokumentumok előállítása szükséges. A módszertan a fejlesztési feladatok előkészítéséhez 4 szintű tervezési tevékenységet ír elő. A koncepcionális előtervezés során szükséges meghatározni a különböző követelményhalmazokat, míg a koncepció alkotás feladata egy kiterjesztett termék feladatlista előállítása. A fejlesztési szakaszban a Military Scrum sprintek ütemezéséhez igazodva a termék feladatlistából kerülnek kiválasztásra a megvalósítandó követelmények. Azok alapján történik a fejlesztő csapat bevonásával a fejlesztési feladatok részletes meghatározása

– az adott sprint feladatlistájának összeállítása. A folyamat akkor éri el a megfelelő érettségi szintet, ha a feladatok lebontása eléri a követelmények szintjén a megvalósíthatósági tanulmányok finomságát. A lépcsőzetes specifikációs eljárás végén a Military Scrum előírja a fejlesztési feladatokhoz a különböző jellegű tesztforgatókönyvek dokumentálását a forráskód elkészítése előtt. A fejlesztés során a tesztforgatókönyvek és követelményrendszer is tovább finomíthatók a fejlesztési tapasztalatok alapján. Az alábbi táblázatban a Military Scrum szereplői és a négyszintű specifikációs eljárás kapcsolata látható – a megfelelő szakértői támogatás megléte szükséges.

	Köv. rendszer meghatározása	**Termék feladatlista kialakítása**	**Sprint-feladatlista meghatározása**	**Tesztforgatókönyvek elkészítése**
Cust	x	x		
PO	x	x	x	x
Dev			x	x

10. táblázat Tervezési szintek és felelősségi körök (saját szerkesztés)

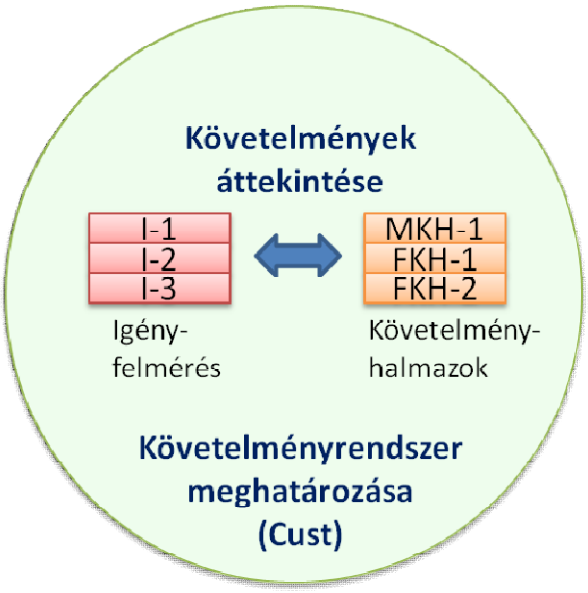
1. *Követelményrendszer meghatározása* – a specifikációs módszer első lépése, amely a megrendelő és a terméktulajdonos közös feladata. Itt történik a magas szintű követelmények meghatározása, feltárása, strukturálása. Ennek a tervezési szakasznak a feladata a megfelelő dokumentációk előállítása, amelyek tartalmazzák a követelményhalmazokat és az azonosított felelősségi köröket.
2. *Termék feladatlista kialakítása* – ez a tevékenység két részre bontható. Az első lépésben a megrendelő teljes körű részvétele is szükséges, mert ekkor határozzák meg a termék-feladatlista első változatát, ezt követően már a terméktulajdonos a megfelelő kommunikáció mellett egyedül is vezetheti azt. Ebben a dokumentumban jelennek meg a megrendelő által megfogalmazott új követelmények, valamint a sprint tervezések és a fejlesztések során azonosított újabb feladatok.
3. *Sprint feladatlista meghatározása* – ez a lépés az ismétlődő tervezések során azonosított megvalósítási tervek létrejöttét jelenti. A termékre vonatkozó egyes feladatok részfeladatokra történő szétbontása ezen a szinten valósul meg. Itt már nem szükséges a direkt megrendelői jelenlét.
4. *Tesztforgatókönyvek elkészítése* – ebben a lépésben a fejlesztők és a terméktulajdonos közösen teszteseteket dolgoznak ki, ez a legalacsonyabb szintű specifikálása a kifejlesztendő szoftvernek. A tesztesetek dokumentálását és jóváhagyását követően – ami visszavonatkoztatható a sprintter-

vezési dokumentációra vagy akár a teljes szoftver követelményrendszerére – elkészíthetők az automatizált tesztek fejlesztőcsapat által.

A Military Scrum által előírt különböző dokumentum típusok részletes meghatározása és bemutatása egyaránt szükséges a módszer helyes alkalmazásának szemléltetéséhez. Igazodva a NATO SLCM nyomon követési elvárásaihoz a Military Scrum egyedi azonosítók képzését írja elő a különböző típusú tervezési dokumentumok számára. A megrendelő felelőssége az egyedi projektazonosítók meghatározása, a feladat megoldható egy megrendelő által vezetett projekt katalógus segítségével. A Military Scrum a következő azonosítási technikát írja elő, ahol a PA az adott projekt egyedi azonosítója.

- Igény azonosítók: *PA-I-1, … , PA-I-n*
- Műszaki követelményhalmaz azonosítók: *PA-MKH-1, … , PA-MKH-n*
- Funkcionális követelményhalmaz azonosítók: *PA-FKH-1, … , PA-FKH-n*

A továbbiakban helytakarékosság miatt a projekt azonosító nem kerül feltüntetésre a példák során, valódi alkalmazási környezetben azonban kötelező a vezetése.



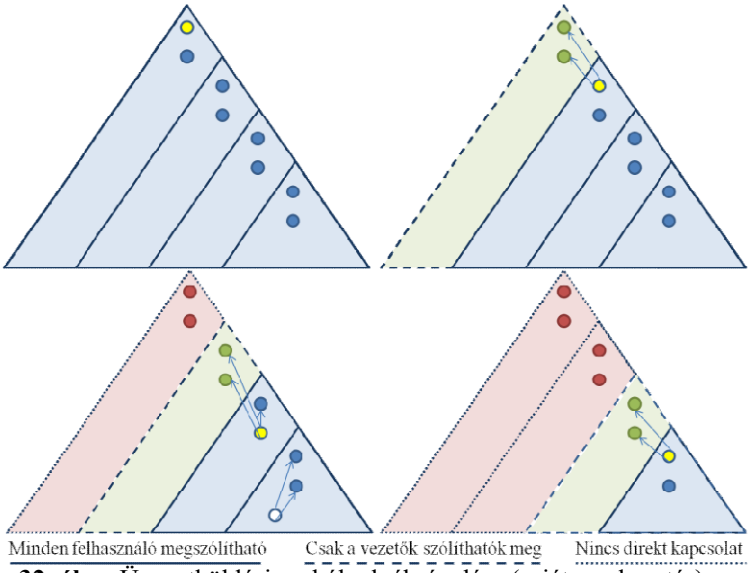
31. ábra Military Scrum 1. tervezési szint – koncepcionális előtervezés (saját szerkesztés)

A Military Scrum projektek során a javasolt azonosító képzési technikától el lehet térni, ilyenkor azt a PMS-ben rögzíteni kell. A technika segítségével biztosítható a szabályozott egyedi azonosító képzés. A követelményrendszer meghatározása akkor ér véget, amikor a feltárt követelményhalmazok következetesen képezik le a megrendelői igényeket, más szóval, a követelményrendszer meghatározása egy ellentmondásmentes eredménnyel záruló követelmény áttekintéssel ér véget. A továbbiakban a termék feladatlista előállításának speciális módszerei következnek.

4.3.1 Új szoftver fejlesztése

A specifikációs módszer bemutatása egy példafeladat megoldásán keresztül valósul meg. A feladat egy állandóan funkcionáló üzenetküldő alkalmazás elkészítése, amely valós időben közvetíti az üzeneteket[71] egy különálló hírközlő hálózaton a Honvédelmi Minisztérium és a Magyar Honvédség munkatársai számára. A kommunikáció során egy speciális titkosító eljárást kell alkalmazni, amely egy egyedi algoritmust használ és mobil eszközök esetén hardverkulcsos titkosítást tételez fel. Az üzenet egy 256 karakter hosszú ASCII szöveges üzenet lehet, tehát legfeljebb 256 byte. Tételezzük fel, hogy az érintett felhasználók száma 700 fő.

A legmagasabb szinten lévő felhasználó akár minden alárendelt szervezet minden egyes felhasználóját megszólíthatja, az alárendelt szervezetek vezetői értelemszerűen csak a saját alárendeltségükben lévő felhasználókat szólíthatják meg egyszerre. Az alábbi ábra az egyes szinteken lévő vezetők, helyetteseik és az alárendeltjeik hírközlési lehetőségeit mutatja be.



32. ábra Üzenetküldési szabályok ábrázolása (saját szerkesztés)

A fenti feladat kapcsán megállapítható, hogy egy teljesen új, központi infokommunikációs szolgáltatás követelményrendszerének kidolgozása a feladat. Ezen a tervezési szinten először is az infrastrukturális és a szoftvert érintő kérdések összegyűjtése, tisztázása a cél. A felmérés elvégzését egy általános dokumentációs sablon alkalmazásával lehet elvégezni, amely a Military Scrum 1. tervezési szintjének, a követelményrendszer meghatározásnak a támogatására szolgál. A módszertan által meghatározott dokumentációs sablonok a 2. számú mellékletben találhatók.

[71] Az USA korai katonai célú üzenettípusaira Négyesi Imre hivatkozott munkájában találhatunk példákat [106, 148-149 o.]

A koncepcionális előtervezés során az igények felmérését a katonai hierarchia minden szintjén el kell végezni. A különböző szinteken szükséges a szakági felelősök igényeinek rögzítése, mert az igényekben lényeges különbségek mutatkozhatnak. A folyamat támogatásához a Military Scrum az alábbi oszlopok meglétét írja elő.

1. *Követelmény támasztója:* az adott követelményhalmazt meghatározó szervezet vagy személy
2. *Érintett szervezeti elemek száma*: az adott követelményhalmazban lévő funkciókat alkalmazó szervezeti elemek száma
3. *Alkalmazás jellege:* stacioner rendszer / tábori rendszer
4. *Felhasználók száma:* a rendszer felhasználóinak száma az adott követelményhalmazhoz tartozó szinten. A követelményhalmazban lévő funkciók leendő felhasználóinak száma.
5. *Felhasználók által használt funkciók köre:* a rendszer felhasználói által használt funkciók az adott alkalmazási szinten
6. *Eszközök*: Infrastrukturális követelmények alkalmazói és szolgáltatói oldalon, például eszközökre (PC/mobil), operációs rendszerre vonatkozó követelmények.

Az alábbiakban egy fiktív négy szinten végrehajtott igényfelmérés eredménye látható a Military Scrum igényfelmérést támogató dokumentációs sablonjának segítségével. A különböző szinteken csak az eszközök vonatkozásában és a potenciális felhasználók számában mutatkoznak eltérések.

Ig.Ssz.	Követelmény támasztója	Érintett szervezeti elemek száma	Alk. jellege	Felhasználók száma	Felhasználók által használt funkciók köre	Eszközök
I-1	HM (Stratégiai, politikai szint)	1	stacioner rendszer	50	Üzenetküldés alárendeltek számára (vezetői funkció) Üzenetküldés felettes számára	Mobil, operációs rendszer: Android
I-2	MHP (Stratégiai, hadászati szint)	1	stacioner rendszer	50	Üzenetküldés alárendeltek számára (vezetői funkció) Üzenetküldés felettes számára	Mobil, operációs rendszer: Android PC, vékony kliens
I-3	MHP IICSF (Hadműveleti szint)	3	stacioner rendszer	100	Üzenetküldés alárendeltek számára (vezetői funkció) Üzenetküldés felettes számára	PC, vékony kliens
I-4	(Harcászati szint követelményei)	20	stacioner rendszer	500	Üzenetküldés alárendeltek számára (vezetői funkció) Üzenetküldés felettes számára	Mobil, operációs rendszer: Android PC, vékony kliens

11. táblázat Követelmények gyűjtése a katonai hierarchia minden szintjén a Military Scrum alkalmazása során (saját szerkesztés)

Kizárólag az igények és követelmények felmérését követően kerülhet sor a műszaki követelmények meghatározására, ugyanis a szolgáltatás kialakításához szükséges infokommunikációs eszközök képességeit és a fejlesztés tárgyát képező szoftver architektúráját is a funkcionális követelmények alapján lehet meghatározni. A beérkezett információk lehetővé teszik a legmagasabb szintű infokommunikációért felelős szervezet számára, hogy elvégezze a műszaki követelmények megállapítását az alkalmazandó hardverekre és szoftverekre egyaránt. Az alábbi felsorolásban találhatjuk a műszaki követelmények elemzésének végeredményét, majd ezt követően a koncepcionális előtervezés végeredményeként azonosított követelményhalmazok szerepelnek.

1. alkalmazás jellege: stacioner rendszer
2. Eszközök típusa: pc, mobil
3. Szükséges sávszélesség: Ha a legmagasabb vezető szólít meg egyszerre mindenkit, akkor 700 * 256 byte ~ 0,18 MByte ~ 1 Mbit/sec.72 Tegyük fel, hogy a hírközlési csatorna adott, nem szükséges további fejlesztés.
4. Szoftverhez szükséges architekturális komponensek:
    a. böngészőben futó vékony kliens
    b. Androidon futó vastag kliens
5. Kliens szoftvereket összehangoló szerver oldali szolgáltatás
6. Szoftverrel kapcsolatos igények
    a. direkt megfogalmazott igény: üzenetküldés
    b. az elemzés során feltárt indirekt igény: a szervezeti felépítés és a felhasználók adminisztrálása a rendszeren belül
7. Biztonsági kérdések: hardveres titkosítás szükséges – tegyük fel, hogy nincs olyan mobil eszköz rendszeresítve, mely képes a hardveres titkosításra, ezért a központi szolgáltatás bevezetéséhez új eszközök beszerzésére van szükség. Hardverkulcsok felhasználókhoz rendelése.
8. Infrastruktúra: központi – tegyük fel, hogy a meglévő hálózati infrastruktúra alkalmazása elegendő. A speciális biztonsági követelmények miatt a stratégiai szint számára legfeljebb 100 új mobil eszközre van szükség.

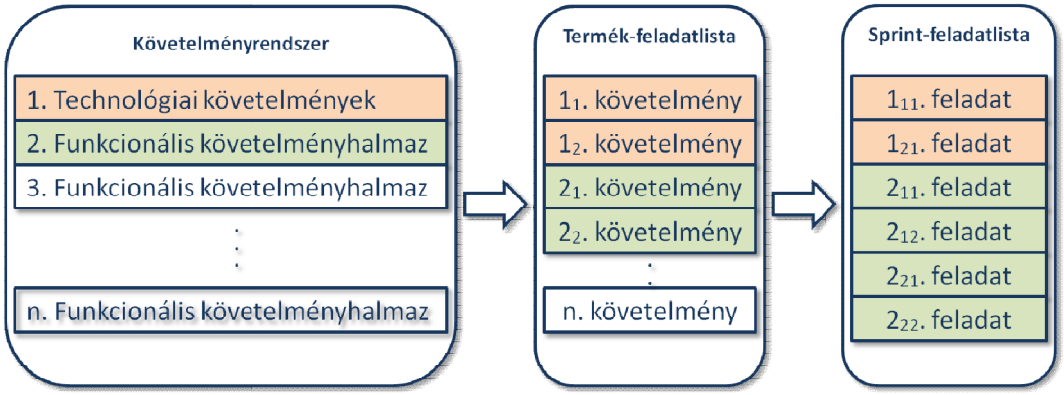
[72] Felhasználók száma összesen 700 fő:
1. Stratégiai szinten: 50 + 50 fő
2. Hadműveleti szinten: 100 fő
3. Harcászati szinten: 500 fő

A 12. táblázatban a koncepcionális előtervezés végére előálló egyedileg azonosított követelményhalmazok láthatók. Az ellentmondások feloldását ebben a szakaszban a megrendelő által bevont szakértőknek kell elvégezniük megrendelői oldalon.

Kh.Ssz.	I. Ssz.	Szakterület	Szervezet	Szakértő	Követelmény
MKH-1	I-1, I-2, I-3	Informatika	MHP IICSF	Lelkes Zoltán ezds.	Architektúra és futtató környezet • Kliens-szerver architektúra kialakítása • Vastag kliens alkalmazás kialakítása Android operációs rendszerre • Böngésző független vékony kliens alkalmazás kialakítása
MKH-2	I-1, I-2, I-3	Biztonság	MHP IICSF	Gyanakvó Péter alez.	Hálózaton történő titkosítási eljárás kialakítása:(mobil-mobil, mobil-PC, PC-PC)
MKH-3	I-1, I-2, I-3	Infrastruktúra	MHP IICSF	Alapos István ezds.	• 100 darab hardverkulcsos titkosításra alkalmas mobil eszköz beszerzése • Hardverkulcsok felhasználókhoz rendelése
FKH-1	I-1, I-2, I-3	Informatika	MHP IICSF	Szakértő Géza alez.	• Üzenetküldés felettes számára • Üzenetküldés alárendeltek számára
FKH-2	I-1, I-2, I-3	Informatika	MHP IICSF	Szakértő Géza alez.	• Szervezeti egységek adminisztrálása • Felhasználók adminisztrálása
FKH-3	.	.	.	.	.
FKH-N					

12. táblázat Szakterületekhez kapcsolt követelményhalmazok a Military Scrum alkalmazása során (saját szerkesztés)

Az iménti példa azt mutatja be, hogyan lehet a Military Scrum dokumentációs technikával felmérni és azonosítani egy katonai célú infokommunikációs rendszerrel szemben támasztott direkt és indirekt igényeket. A példában azonosított követelményrendszer alapja lehet egy belső vagy akár külső megvalósítású infokommunikációs rendszerfejlesztésnek és az előállított dokumentációs struktúra alkalmas egy agilis szoftverfejlesztési projekt előkészítéséhez is. A bemutatott eljárás a Military Scrum megrendelői oldalon végrehajtandó követelményrendszer meghatározó módszerének tekintendő. A bemutatott módszer lehetővé teszi a követelményrendszer strukturált előállítását követelményhalmazok formájában, ezzel megalapozva a koncepcióalkotás szakaszát, amely során a termék feladatlista előállítása a feladat.



33. ábra A Military Scrum szoftverfejlesztési módszertan során keletkező dokumentumok kapcsolata (saját szerkesztés)

A 33. ábra a követelményrendszer és a termékre vonatkozó feladatlista, valamint az agilis megvalósítás során keletkező sprint feladatlista kapcsolatát mutatja be. A termék feladatlista előállítása a Military Scrum szerint a megrendelő és a terméktulajdonos közös feladata a tervezés 2. szintjén. A fejlesztési szakasz megkezdése előtt az addig összegyűjtött igényeket tartalmaznia kell a dokumentumnak, valamint a követelményrendszerből néhány követelményt részletesen ki kell dolgozni addigra.

A következő példa azt szemlélteti, hogyan lehet a fejlesztést egy jól meghatározott követelményrendszer alapján elindítani. A Military Scrum esetében a fejlesztés alapját a termék feladatlista képezi, amely tartalmazza a kifejlesztendő szoftverrel kapcsolatos összes azonosított követelményt a tervezett megvalósítás szerint sorba rendezve. Ez a sorrend határozza meg a fejlesztés menetét, a legmagasabb prioritású feladatok kerülnek a soron következő fejlesztési sprint-be. A Military Scrum termékfeladatlistája az alábbiakban bemutatott oszlopokat tartalmazza:

1. Követelményhalmaz sorszáma: a követelményhalmaz azonosítója, mely alapján az adott feladat meghatározásra került.
2. Sorszám: az adott követelmény sorszáma. A megrendelő által azonosított követelményhalmaz egy külön kezelhető követelményét azonosítja.
3. Követelmény: az adott követelmény leírása.
4. Prioritás: az adott feladat prioritása, egy pozitív egész szám, a termékfeladatlista rendezésének elvét ez az oszlop határozza meg [7, 103-104. o.]
5. Modul: az adott feladatot tartalmazó modul neve, akkor értelmezett, ha az adott rendszer több modulra bontható
6. Állapot: Az adott feladat megvalósításának állapota, mely lehet: Várakozik, Folyamatban, Törölt
7. Storypont: Az adott feladat megvalósítására becsült erőforrás fejlesztői napokban [7, 128-129. o.]
8. Megjegyzés: Az adott feladathoz kapcsolódó megjegyzés
9. Leírás: Az adott feladathoz kapcsolódó rövid leírás
10. Képernyőterv: Ez egy kép vagy képek sorozata, mely az adott feladathoz kapcsolódó felületeket ábrázolja. Az adott követelményhalmazt meghatározó szervezeti elemmel szükséges a felületek tervének egyeztetése a fejlesztés megkezdése előtt. Felhasználói felülettel rendelkező egyedi alkalmazásfejlesztés esetén szükséges.

A 13. táblázatban a követelményhalmazok alapján feltárt termék feladatlista szerepel.

KhSsz	Ssz.	Követelmény	Pri.	Mod.	Állapot	Sp	Megjegyzés	Leírás	Kép. terv.
MKH-1	UF-11	Vékony kliens architektúra kialakítása	100	PC, Mobil	Várakozik	40	Responsive[73] megoldást kell választani	-	-
MKH-2	UF-12	Üzenet titkosítása hardverkulcs segítségével	90	Mobil	Várakozik	40	Az eszközök beszerzése szükséges a feladat megkezdéséhez	-	-
FKH-2	UF-13	Szervezeti egységek adminisztrálása	80	Mobil	Várakozik	30			screen1.jpg, screen2.jpg
FKH-2	UF-14	Felhasználók adminisztrálása	70	Mobil	Várakozik	30			screen3.jpg, screen4.jpg
FKH-1	UF-15	Üzenetküldés felettes számára	60	PC, Mobil	Várakozik	20			screen5.jpg
FKH-1	UF-16	Üzenetküldés alárendeltek számára	60	PC, Mobil	Várakozik	10			screen6.jpg
.	.	.	.	.	.	.	.	.	.
FKH-N	UF-X	-	-	-	-	-	-	-	-

13. táblázat Military Scrum termék feladatlista (saját szerkesztés)

A termék feladatlista tetején csak jól kidolgozott és megvalósítható feladatok állhatnak, a Military Scrum esetében minden követelményről tudható, hogy melyik szakterület formálta, így a problémák tisztázása, a feladatok pontosítása az adott fejlesztési sprint megkezdése előtt is elvégezhető, kérdés esetén a megfelelő katonai szakterület megszólítható.

Ha újabb kérdések merülnek fel a fejlesztések során, akkor az ismétlődő karbantartás folyamatlépésben a feltárt kérdések tisztázhatók és beépíthetők a követelményrendszerbe. A megoldás nagy előnye, hogy egy szűkebb projektcsapat dolgozik a sprintek megvalósításán, ugyanakkor a fejlesztett szoftver teljes élettartama alatt minden követelményhez a kapcsolódó szakterület megszólítható, a kérdések tisztázhatók, így a fejlesztés végére a karbantartott követelményrendszer is rendelkezésre áll. Ha a dokumentációk karbantartása folyamatos, akkor a fejlesztetett szoftver és a hozzá kapcsolódó dokumentumok teljes mértékben lefedik egymást.

Új szoftverek fejlesztése esetén a Military Scrum 1. menetében a bemutatott módszer segítségével valósul meg a termék feladatlista előállítása, azonban létezhetnek további célokkal indított szoftverfejlesztési projektek is. Szoftverek cseréje, frissítése, adatmigrálási feladatok megvalósítása vagy létező rendszerek között interfészek kialakítása az átjárhatóság megteremtésével egyaránt feladata lehet a szoftverfejlesztési projektnek, a Military Scrum megoldást kínál a különböző célok eléréséhez is.

[73] A responsive technológia jelentése a web-fejlesztésben, hogy az oldalt megjelenítő képernyő felbontásától függően más és más elrendezésben jelennek meg a felületi komponensek a felhasználói élmény javítása érdekében.

4.3.2 Szoftver cseréje új szoftver fejlesztésével

A koncepcionális előtervezés úgy is zárulhat, hogy az a konszenzusos döntés születik, hogy egy meglévő szoftver cseréje szükséges a kívánt katonai képesség eléréséhez. Ha a kiváltandó szoftvert korábban agilis technikákkal fejlesztették, akkor akár az is feltételezhető, hogy létezik egy karbantartott termékre vonatkozó feladatlista, amely kiindulási alapként felhasználható a koncepcióalkotás során a Military Scrum 1. fejlesztési sprintjében. Azonban, ha nem áll rendelkezésre vagy nem teljes a termékre vonatkozó követelményrendszer, akkor a módszertan ad egy erre a célra kialakított követelményelemző eljárást.

Egy már létező szoftver működésének feltárásához más eszközökre van szükség, máshonnan indul a követelményelemzési folyamat. Speciális módszer kell ahhoz, hogy ki tudjuk nyerni az újrahasznosítható információt egy kiváltandó rendszerből olyan formában, hogy azt majd fel is tudjuk használni egy új rendszer tervezése során. Az alábbiakban erre a problémára keressük a választ.

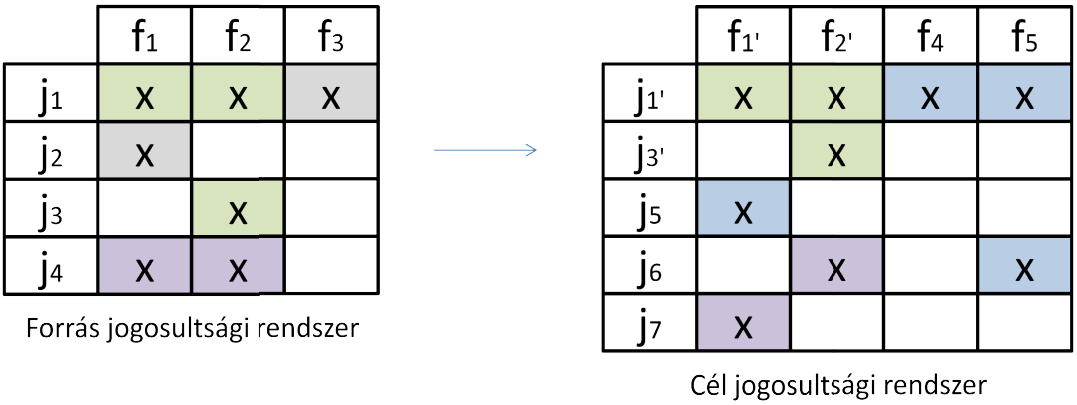
Gyakran tapasztalható az informatikában, hogy már korábban alkalmazásba vett rendszereket az idő elteltével lecserélnek, helyüket új szoftverekkel váltják ki. Ennek oka szerteágazó lehet, egyik eshetőség, hogy műszakilag elévül az alkalmazott rendszer és új technológiai alapokra kell helyezni. Az is elképzelhető, hogy újabb igények jelennek meg a szoftverrel kapcsolatban, amelyeket már nem lehet az eredeti technológiai környezetben kezelni. Alkalmazói oldalon ilyenkor komoly problémát jelenthet az új szoftverrel szemben támasztott követelményrendszer összeállítása. Vegyük kiindulási pontnak az agilis módszerek legalapvetőbb technikáját, hogy először a nagyobb követelmények kerülnek azonosításra, majd azokat bontják apróbb feladatokra és ezt követi a megvalósítás. Hasonló gondolatmenet alkalmazása elképzelhető egy már működő szoftver képességeinek feltérképezésére. A funkció-jogosultsági mátrix[74] már egy régóta ismert szoftvertechnológiai eszköz egy jogosultsági rendszerrel ellátott szoftver működésének leírására.

Az első lépés a használatban lévő rendszer funkcióinak összegyűjtése. Ez a mozzanat megtehető a forráskód elemzésével, valamint a felhasználói felület feltárásával. A legjobb eredmény elérése érdekében célszerű mindkét módszer együttes alkalmazása. Egyszerűbb rendszerek esetében egy lehetséges módszer lehet egy teljes jogosultsági

[74] Funkció-jogosultsági mátrix – egy olyan táblázat, melynek oszlopait egy adott informatikai rendszerben elérhető funkciók alkotják, sorai az adott rendszerben azonosított jogosultságokat reprezentálják. A táblázat egy cellája akkor van kitöltve, ha az adott oszlophoz tartozó funkciót el lehet érni az adott sorhoz tartozó jogosultsággal.

körrel rendelkező tesztelő felhasználó alkalmazásával feltárni a kiváltandó rendszer funkcióit, egyfajta fekete doboz tesztelést végezni. Ha már rendelkezünk egy alapvető rendszerismerettel, akkor a jogosultsági rendszert az egyes jogosultságok elvételével indirekt módon lehet feltárni, jó esetben egy addig elérhető funkció már nem lesz aktív egy adott jogosultság elvételét követően. Ez a módszer erősen függ a szoftver jellegétől, a rendszerben megvalósított folyamatoktól, nem mindig alkalmazható.

Összetettebb rendszekerek esetén a rendelkezésre álló felhasználói és műszaki dokumentáció tanulmányozása is kiindulási alapja lehet a feltáró munkának. Amennyiben sokszereplős és összetett munkafolyamatok jelennek meg egy rendszeren belül, szükségszerű lehet tesztforgatókönyvek kidolgozása az egyes jogosultságok és funkciók kapcsolatának feltárásához. Segítség lehet még a forrásrendszer felhasználói és műszaki dokumentációjának tanulmányozása, valamint olyan tesztforgatókönyvek kialakítása, melyek végrehajtását követően a kiváltandó rendszer adatbázis modelljéből további információk nyerhetőek ki.



34. ábra Jogosultsági rendszerek közötti leképezés (saját szerkesztés)

A feltárt funkciók és a transzformált jogosultsági mátrix alapján előállítható az új szoftverrel szemben támasztott követelményrendszer, a felelősségi-körök meghatározhatók, a Military Scrum fejlesztési szakasza előkészíthető. A függőségek feltárását is el kell végezni, ez lehet a fejlesztési sorrend kialakításának egyik alapja. Az új adminisztrációs felületek kialakítását célszerű már az elején elvégezni. A 33. ábrán azt láthatjuk, hogy a forrásrendszer *f1*, illetve *f2* funkciói leképezésre kerülnek a célrendszer *f1'* és *f2'* funkcióra, ugyanakkor az *f3* funkció megszűnik és új *f4*, *f5* funkciók kerülnek kialakításra. A jogosultságok vonatkozásában azt lehet látni, hogy a *j1*, *j3* jogosultságok leképezésre kerülnek a cél rendszerbe, míg a *j2* jogosultság megszűnik. Új jogosultság a *j5*, míg a *j4* jogosultság két újonnan bevezetett jogosultsággá bomlik szét: *j6*, *j7*, melyekkel az *f1'* és *f2'* funkciók érhetőek el a célrendszerben.

Egy valós projekt esetében is hasonló transzformációs szabályok figyelhetők meg a forrás és célrendszerek között. Természetesen a különbségek mértéke határozza meg a szabályok mennyiségét és a feladat bonyolultságát.

Tegyük fel, hogy a cél rendszerrel szemben támasztott egyéb követelmények elemzése megtörtént. Az iménti példából és a feltételezett egyéb követelményekből kiindulva a célrendszerre vonatkozó feladatlista összeállítható. A 14. táblázat néhány fiktív műszaki követelményt, egy kitalált jogosultsági rendszer kialakításának feladatait és a 34. ábrán bemutatott új funkcionális követelményeket tartalmazza.

A táblázatból a *Megjegyzés* és *Leírás* oszlopok helyén a *Jogosultságok* és a *Korábbi funkció* oszlopok kaptak helyet, eltérve ezzel a *Military Scrum* termékre vonatkozó feladatlistájától. A *Jogosultságok* oszlopban az új funkcióhoz kapcsolódó jogosultságok kerülnek felsorolásra. A *Korábbi funkció* oszlopban a forrásrendszerben azonosított korábbi funkciók és a hozzájuk tartozó jogosultságok szerepelnek.

A forrásrendszer funkcióinak bevezetése a dokumentumban azért szükséges, hogy a fejlesztés során az újonnan fejlesztett funkció összevethető legyen az elődjével, a felmerülő kérdések egyszerűbben tisztázhatók legyenek. A módszer feltételezi, hogy a forrás rendszerben megvalósított funkciók megalapozott követelményekként jelennek meg az új rendszer esetében.

KhSsz	Ssz.	Követelmény	Prior.	Állapot	Sp	Jogosultságok	Korábbi funkció
MKH-1	UK-11	Vékony kliens architektúra kialakítása	100	Várakozik	40	-	-
MKH-2	UK-12	Relációs adatbázis kialakítása	90	Várakozik	40	-	-
FKH-1	UK-13	Szervezeti elemek adminisztrálása	80	Várakozik	20	-	-
FKH-1	UK-14	Felhasználók adminisztrálása	70	Várakozik	20	$j_{1'}$, $j_{3'}$, j_5, j_6, j_7	-
FKH-2	MK-11	$f_{1'}$ funkció kialakítása	60	Várakozik	13	$j_{1'}$, j_5, j_7	f_1 funkció j_1, j_2, j_4 jogosultságokkal
FKH-3	MK-12	$f_{2'}$ funkció kialakítása	60	Várakozik	8	$j_{1'}$, $j_{3'}$, j_6	f_2 funkció j_1, j_3, j_4 jogosultságokkal
FKH-4	MK-13	f_4 funkció kialakítása	50	Várakozik	8	$j_{1'}$	-
FKH-4	MK-14	f_5 funkció kialakítása	50	Várakozik	5	$j_{1'}$, j_6	-
.	.	.	.	.	.	.	.
FKH-N	K-X	-	-	-	-	-	-

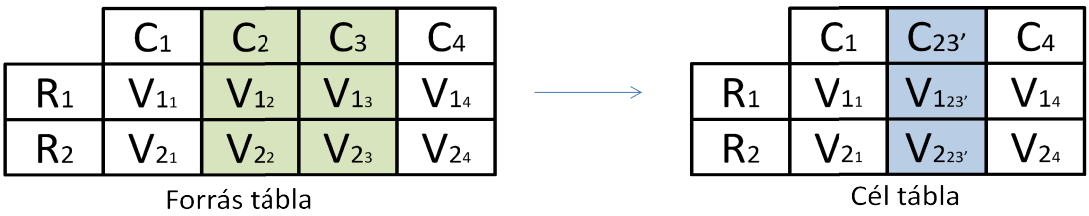
14. táblázat Termékre vonatkozó feladatlista létező rendszer feltárása alapján (saját szerkesztés)

Az imént bemutatott módszerrel a szoftverek kiváltására vonatkozó követelményrendszer összeállítása szabályozott keretek között végezhető el. Az új és az örökölt követelmények egy dokumentumban, strukturált formában kapnak helyet ezzel megalapozva az új szoftver elkészítésének megfelelő minőségű támogatását.

4.3.3 Adatmigrálás

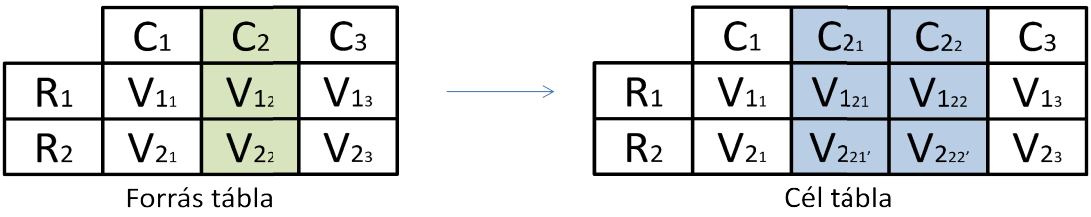
Napjainkban az adatok tárolására leggyakrabban a relációs adatbázis-kezelő rendszereket (RDBMS) [107] használják a szoftverfejlesztők. A technológia több, mint 40 éve jelent meg és mára széleskörűen elterjedt, jelenleg is gyakori választás az új rendszerek kialakításakor. A továbbiakban a relációs adatbázisok közötti adatmigrálás agilis menete kerül bemutatásra. A fogalmak újragondolása is célszerű, mert a módszer nem csak egy kiváltandó rendszer és egy új rendszer közötti adatmigrálás folyamatát mutatja be, természetesen itt is alkalmazható, hanem egy forrás- és célrendszer közötti adatmegfeleltetés szabályainak leírására szolgál.

A relációs adatbázisok az adatok tárolását adattáblákkal, azokon belül különböző típusú oszlopokkal valósítják meg. Az egyes táblák közötti kapcsolatok leírására szolgálnak az idegen kulcsok. Kulcs lehet egyetlen oszlop vagy oszlopok halmaza, amelyekkel egyértelműen azonosítható egy sor egy adattáblában. Egy adatmigrálás során a problémát az okozza, hogy forrás és célrendszer adatstruktúrája, tárolt adatai merőben eltérhetnek egymástól. Elképzelhető, hogy a forrás rendszer több oszlopot használ egy adott funkcióhoz tartozó adatok modellezésére. Tegyük fel, hogy adott egy *f* funkció és a működéséhez szükséges adatokat a forrás rendszer egy tetszőleges táblájában a *C2, C3* oszlopok tartalmazzák. Mindemellett a cél rendszerben az *f* funkció *f'* leképezése a *C23'* oszlopot használja a működéséhez szükséges adatok tárolására.



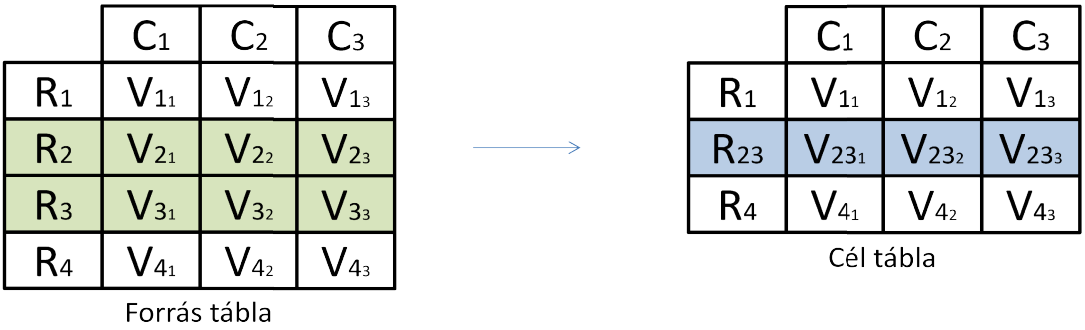
35. ábra Adatmigrálás során oszlopok összevonása (saját szerkesztés)

Az is előfordulhat, hogy az *f* funkció működéséhez szükséges adatokat a forrás rendszer egy tetszőleges táblájában a *C2* oszlop tartalmazza. Ehhez képest a cél rendszerben az *f* funkció *f'* leképezése a *C21* és C*22* oszlopokat használja a működése során.



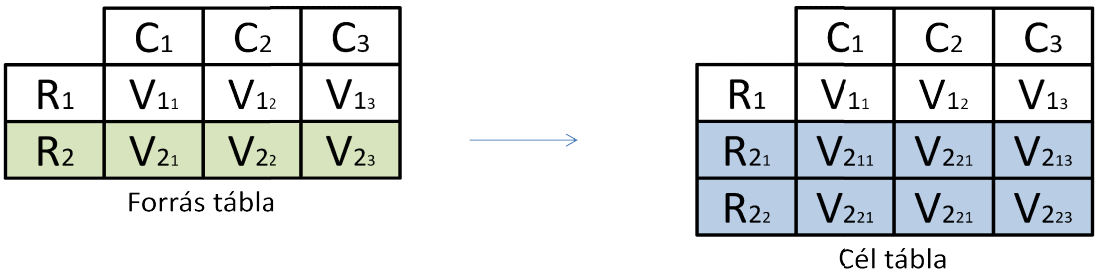
36. ábra Adatmigrálás során oszlopok szétbontása (saját szerkesztés)

Egy adott funkció működése során nem csak a működéshez szükséges oszlopok száma lehet különböző, az is fennállhat, hogy a forrásrendszer több sorban tárolja le az adatokat, mint a célrendszer. Az is lehet, hogy a forrás adatbázisban a funkció működése több adatbázistáblát is érint. Az egyszerűség kedvéért most tételezzük fel, hogy ugyanazon tábla két sora egy funkció meghívásának végeredménye a forrásrendszerben, míg a célrendszerben az eredmény tárolására egyetlen sor szolgál. Tegyük fel, hogy az *f* funkció futásának végeredménye a forrásrendszerben az *R1*, illetve *R2* sorok, míg az *f* funkció *f'* leképezésének meghívása a célrendszerben egy összevont *R23* sorban tárolja ugyanazokat az adatokat más struktúrában más tartalommal.



37. ábra Adatmigrálás során adatsorok összevonása (saját szerkesztés)

Végezetül az is elképzelhető, hogy a forrásrendszer egy adott funkciójának működése során kevesebb sor keletkezik az adott adatbázis táblában, mint a célrendszerben. Hasonlóan az előző példához az is lehet, hogy a céladatbázisban a funkció működése több adatbázistáblát is érint. Tegyük fel, hogy az *f* funkció futásának végeredménye a forrásrendszerben az *R2* rekord, míg az *f* funkció *f'* leképezésének végrehajtása a célrendszerben *R21* és *R22* rekordokban tárolja ugyanazokat az adatokat más struktúrában és más tartalommal.



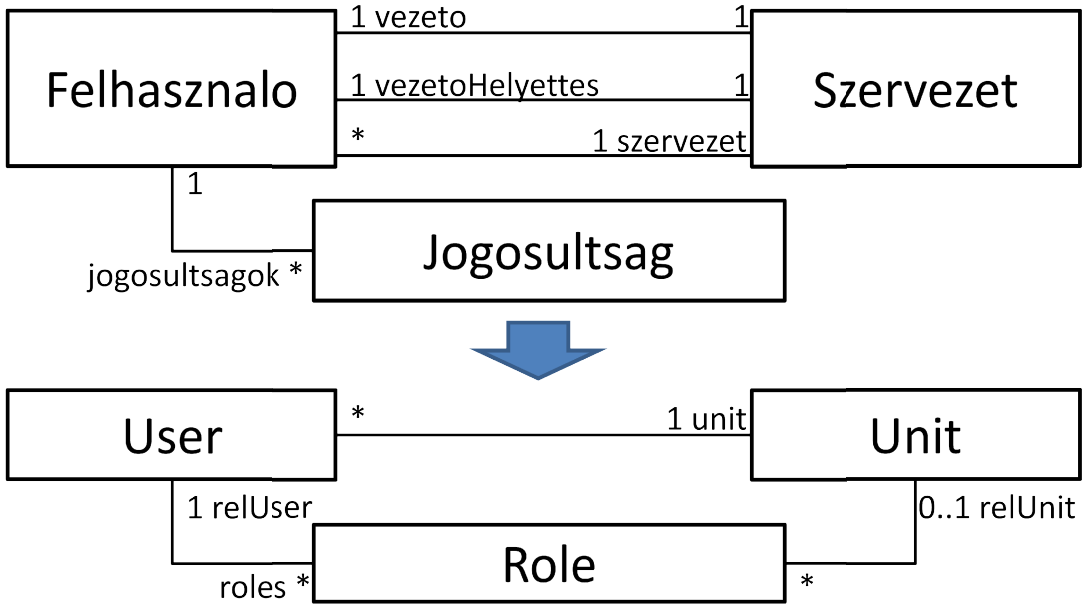
38. ábra Adatmigrálás során adatsor szétbontása (saját szerkesztés)

Az imént bemutatott esetek és ezek különböző kombinációi adják egy adatmigrálási projekt feladatrendszerét. Olyan rendszerek esetében, ahol funkciók százai érhetők el már érezhető a feladat bonyolultsága. A kérdés az, hogyan lehet strukturálni egy

ilyen feladatrendszert, illetve milyen dokumentációs technika bevezetésével lehet egy adatmigrálási feladatot magas színvonalon támogatni.

A probléma szemléltetéséhez tegyük fel, hogy adott egy forrás rendszer, amely három adatbázistáblában modellezi a szervezeti elemeket, a bennük lévő felhasználókat, valamint a felhasználók jogosultságait. Minden felhasználónak pontosan egy szervezeti eleme van. Minden szervezeti elemnek pontosan egy vezetője és egy vezető helyettese van. Ezen felül a felhasználóknak lehetnek egyéb jogosultságai, például: adminisztrátor, ügyintéző.

Tegyük fel, hogy a célrendszer máshogy modellezi a jogosultsági rendszerét. A szemléletesség kedvéért angol elnevezések kerültek bevezetésre a célrendszerben. A *Felhasznalo* adattábla megfelelője a *User* tábla, a *Szervezet* tábla leképezése a *Unit* tábla. A célrendszerben *Role* tábla reprezentálja a jogosultságokat. Lényeges különbség a célrendszerben a forrásrendszerhez képest, hogy a vezető és a vezető helyettes kapcsolat egy-egy jogosultsággal van modellezve.



39. ábra Forrás és cél jogosultsági rendszer bemutatása (saját szerkesztés)

A vázolt feladat megoldása során jól szemléltethető a Military Scrum erre a célra alkalmazható technikája. A módszer bemutatása előtt tegyük fel, hogy az adott forrásrendszer a jogosultsági rendszeren túl más fogalmakat is modellez, amelyeket az egyszerűség kedvéért az *A, B, C*... táblákban tárolja.

A következő felsorolás a Military Scrum adatmigrálást támogató táblázatának oszlopait mutatja be, majd a példa alapján azonosítható követelményeket a 15. táblázat szemlélteti.

1. *Azonosító*: az adott táblát tartalmazó fogalomtér azonosítója
2. *Sorszám*: az adatbázistábla sorszáma
3. *Tábla*: az adatbázistábla neve
4. *Prioritás:* az adott feladat prioritása, egy pozitív egész szám, a termék-feladatlista rendezésének elvét ez az oszlop határozza meg
5. *Állapot:* Az adott feladat megvalósításának állapota, mely lehet: Várakozik, Folyamatban, Kész, Törölt
6. *Storypont:* Az adott feladat megvalósítására becsült erőforrás fejlesztői napokban
7. *Kapcsolatok feldolgozva?:* Igen/Nem – az adott tábla kapcsolataira vonatkozó transzformációs szabályok rendelkezésre állnak-e?
8. *Adatok feldolgozva?:* Igen/Nem – az adott tábla adataira vonatkozó transzformációs szabályok rendelkezésre állnak-e?
9. *Migrációs script kész?:* Igen/Nem – a tervek alapján elkészült-e a leképezéseket megvalósító migrációs script?

Azon.	Ssz.	Tábla	Pr.	Állapot	SP	Kapcsolatok feldolgozva?	Adatok feldolgozva?	Migrációs script kész?
Jog. R.	AKT-11	Felhasznalo	100	Kész	5	Igen	Igen	Igen
Jog. R.	AKT-12	Szervezet	90	Foly.-ban	5	Igen	Igen	Nem
Jog. R.	AKT-13	Jogosultsag	80	Foly.-ban	5	Igen	Nem	Nem
subd.-1	AKT-14	A tábla	70	Várakozik	3	Nem	Nem	Nem
subd.-1	AKT-15	B tábla	70	Várakozik	2	Nem	Nem	Nem
subd.-2	AKT-16	C tábla	60	Várakozik	3	Nem	Nem	Nem
subd.-2	AKT-17	D tábla	60	Várakozik	2	Nem	Nem	Nem
subd.-2	AKT-18	E tábla	60	Várakozik	5	Nem	Nem	Nem
.	.	.	.	.		.	.	

15. táblázat Adatmigrációs feladatlista adattáblák szintjén (saját szerkesztés)

A forrásrendszer fogalomterét célszerű további részekre bontani és így egyfajta altereket (angolul: *subdomain*) kialakítani. Minden altérhez lehetséges két újabb feladatlista kialakítása. Az első táblázat az altéren belüli és esetlegesen kifelé mutató kapcsolatokat gyűjti össze és rendszerezi az alábbiak szerint. Az egyes kapcsolatok feltárása során a megvalósítás sorrendje változhat. Ahhoz, hogy egy kapcsolatot le tudjunk képezni szükséges mindkét tábla kialakítása és csak azt követően lehet beállítani az adott kapcsolatot egy idegen kulcs segítségével. Erre példa a 16. táblázatban az 1. lépéssel illusztrált *Felhasznalo* és *Szervezet* közötti kapcsolat leképezése a cél rendszerbe a *User, Unit* táblákra. A feladat elvégzéséhez azonnal két adatbázistáblával kell műveleteket végezni - az azonosítókat le kell képezni a cél rendszerbe, ezt követően lehet a kapcsolatokat átmozgatni.

Ssz.	Forrás tábla	Forrás oszlop	Kapcsolat	Cél tábl.	Cél oszlop	Cél kap.	Eljá-rás?	Állapot
AKK-11-1	Felhasz.	Id	-	User	id	-	-	Kész
AKK-11-2	Felhasz.	szervezet_id	Szervezet	User	unit_id	Unit	-	Kész
AKK-12-1	Szervezet	id	-	Unit	id	-	-	Kész
AKK-12-2	Szervezet	vezeto_id	Felhasznalo	Role	relUser_id	User	SP-1	Foly.-ban
AKK-12-3	Szervezet	vezeto_id	Felhasznalo	Role	relUnit_id	Unit	SP-1	Foly.-ban
AKK-12-4	Szervezet	helyettes_id	Felhasznalo	Role	relUser_id	User	SP-1	Foly.-ban
AKK-12-5	Szervezet	helyettes_id	Felhasznalo	Role	relUnit_id	Unit	SP-1	Foly.-ban
AKK-12-6	Szervezet	felettes_id	Szervezet	Unit	parent_id	Unit	-	Vár.
AKK-13-1	Jogosult.	id	-	Role	id	-	-	Kész
AKK-13-2	Jogosult.	felhasznalo_ id	Felhasznalo	Role	relUser_id	User	SP-2	Vár.
.	.	.	.	.	.	.	.	.
AKK-XY-1	.	.	.	.	.	.	.	.

16. táblázat Adattáblák közötti kapcsolatokra vonatkozó feladatlista (saját szerkesztés)

Az oszlopok tartalmának leképezése is különböző nehézségű feladat lehet. A legegyszerűbb feladat az oszlopnevek változása, ilyenkor nincs semmilyen transzformációs szabály. Az is előfordulhat, hogy valamilyen szabályrendszer alapján kell összevonni, illetve szétválasztani oszlopok tartalmát, ekkor olyan migrációs scriptekre van szükség, amelyek megvalósítják az adat-transzformációra vonatkozó szabályokat. Az is lehet, hogy valamilyen diszkrét értékkészlet van egy forrás oszlopban és a célrendszer máshogy modellezi ezeket az értékeket, ekkor hasonlóan valamilyen adatmigrációs szabályt kell implementálni. A következő táblázat ezt szemlélteti.

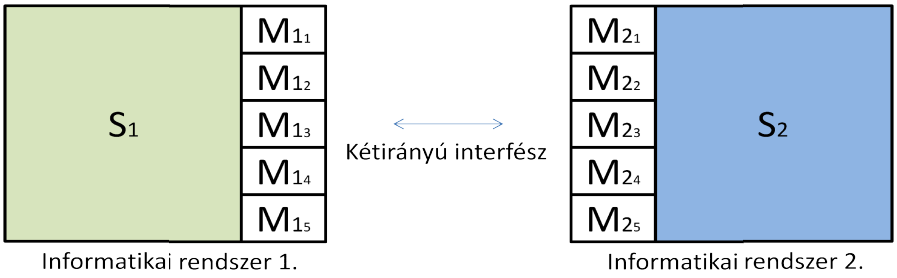
Ssz.	Forrás tábla	Forrás oszlop	Értékkészlet	Cél tábla	Cél oszlop	Érték-készlet	Állapot
AKO-11-1	Felhasznalo	loginNev	-	User	login	-	Kész
AKO-11-2	Felhasznalo	nev	-	User	name	-	Kész
AKO-11-3	Felhasznalo	emailCim	-	User	email	-	Kész
AKO-12-1	Szervezet	nev	-	Unit	name	-	Foly.-ban
AKO-12-2	Szervezet	azonosito	-	Unit	shortName	-	Foly.-ban
AKO-12-3	Szervezet	tipus	-	Unit	type	-	Foly.-ban
AKO-12-4	Szervezet	emailCim	-	Unit	email	-	Foly.-ban
AKO-13-1	Jogosultsag	tipus	ADMIN, UGYINTEZO				Várakozik
.	.						
AMO-XY-1	-						

17. táblázat Adattáblák tartalmára (oszlopaira) vonatkozó feladatlista (saját szerkesztés)

A felvázolt módszer egy alapos eljárást ad a forrásrendszer feltárásához, valamint segítségével a fejlesztés menete is figyelemmel kísérhető, magas szinten támogatható. A bemutatott dokumentációs módszer segítségével a koncepcióalkotás kellőképp támogatható, így széleskörűen alkalmazható adatmigrálást igénylő feladatok során. Az adatmigrálás támogatásához három különböző táblázat került kialakításra, amelyek együttesen képezik a termék feladatlistát. Prioritás oszloppal csak az első táblázat rendelkezik, amely az adatbázis táblákat tartalmazza, a másik két táblázat az adatbázis táblák kulcsainak és egyéb oszlopainak rendszerezésére szolgál, a prioritás öröklődik ezekben az esetekben az első táblázatból.

4.3.4 Rendszerek közötti integráció

A létező rendszerek közötti magas szintű integráció kialakítása napjaink egyik leggyakoribb informatikai kihívása. Az alapszintű műszaki megoldást a problémára az interfész[75] tervezési minta alkalmazása jelenti. Az interfészek fejlesztése során ugyancsak alkalmazható a Military Scrum. A feladat mérete, az alkalmazott technológiák és a fejlesztést végző csapatok fizikai elhelyezkedése együttesen határozzák meg, hogy hány termékre – adott esetben interfészre – vonatkozó feladatlistát kell kialakítani. Az *S1* és *S2* informatikai rendszerek között kialakítandó *M11...M15* és *M21...M25* metódusokat tartalmazó kétirányú interfészt szemlélteti az alábbi ábra.



40. ábra Rendszerek közötti integráció (saját szerkesztés)

Amennyiben nem kivitelezhető az egy időben és kompatibilis eszközökkel történő fejlesztés, akkor külön-külön indított Military Scrum projektekkel lefedhető az *M1* és *M2* interfészek fejlesztése. Ha azonban a megfelelő szoftverfejlesztési környezet adott, akkor az *M1* és *M2* interfészeket, valamint a velük megvalósított folyamatok implementálását egyazon projektbe lehet szervezni. A 18. táblázatban egy közös fejlesztési projektbe szervezett feladatlista néhány első sora jelenik meg.

KhSsz	Ssz.	Követelmény	Prior.	Mod.	Állapot	SP.	Megj.	Leírás
FK-1	IF-11	M11 szolgáltatás kialakítása	100	S1	Várakozik	5		-
FK-1	IF-12	M12 szolgáltatás kialakítása	90	S2	Várakozik	5		-
FK-2	IF-13	M13 és M21 szolgáltatás kialakítása	80	S1, S2	Várakozik	8		-
.	.	.	.	.	.	.	.	.
FK-X	IF-XY	.	.		.	.	.	.

18. táblázat Kétirányú interfész kialakítására vonatkozó (saját szerkesztés)

A bemutatott rövid példa jól szemlélteti azt, hogy a különböző rendszerek közötti integrációs feladatok ilyen módon agilis technikával is támogathatók. A módszer azért különös fontosságú, mert segítségével különböző hálózati végpontokon lévő rendszerek integrációját is támogatni lehet.

[75] Interfész - Az interfész szoftverek esetében két rendszer érintkezési felülete, ahol a kommunikáció menete mindkét rendszer számára ismert. Kétirányú interfészről beszélünk, ha mindkét fél megszólítható a másik által.

4.3.5 Biztonsági kockázatok elemzése

Egy követelményhalmaz által lefedett terület egy vagy több biztonsági követelményhez tartozó funkciót jelenthet a termék feladatlistában, ugyanakkor a termék feladatlista összeállítása során egy biztonsági követelmény alapján szükségessé válhat a követelményhalmazok áttekintése és tisztázása a követelményt megformáló entitások között. A Military Scrum követelményelemzési módszerének sikere a követelményhalmazok és termék feladatlista közötti logikai ellentmondások iteratív feloldásán alapszik. Ha az 1. fejlesztési menet utáni időszakról beszélünk, akkor az automatizált tesztelés is segítséget nyújthat a problémák megértésében, a változtatások hatásainak vizsgálatában, így a biztonsági kérdések tisztázásban is. Ekkor, de egyéb esetekben is a követelményrendszer – a követelményhalmazok és a termékre feladatlista – megértése, feltárása és tisztázása a siker záloga.

Egy önállóan működő, fizikailag létező kritikus rendszer esetében is kiemelt fontossággal bír az általa használt informatikai rendszerek megfelelő működése. Példaként gondolhatunk egy vízszolgáltatást biztosító vízhálózati vezérlő rendszerre. Egy ilyen rendszer esetében a háttérben nagy valószínűséggel megjelenik valamilyen vezérlési feladatokat ellátó célszoftver. Általában ez egy szeparáltan működő informatikai rendszer, amely különböző programozható eszközök vezérlésére alkalmas.

Egy ilyen rendszer tervezésénél, továbbfejlesztésénél, illetve létfontosságú rendszerré nyilvánításakor is alkalmas választás lehet a Military Scrum követelményelemző módszere a szükséges dokumentációk elkészítéséhez, mert a módszer lehetőséget biztosít a követelmények strukturált gyűjtéséhez és az iterációk révén egy kifinomult, minden követelményformáló által teljesnek vélhető feladatlista összeállítását teszi lehetővé. A létfontosságú infokommunikációs rendszerek és az alapvető szoftver alapú szolgáltatások esetében is az üzemeltetői biztonsági tervek kötelező elemei a kockázatelemzés, a kockázatkezelés és megvalósítás eszközeire vonatkozó fejezetek.

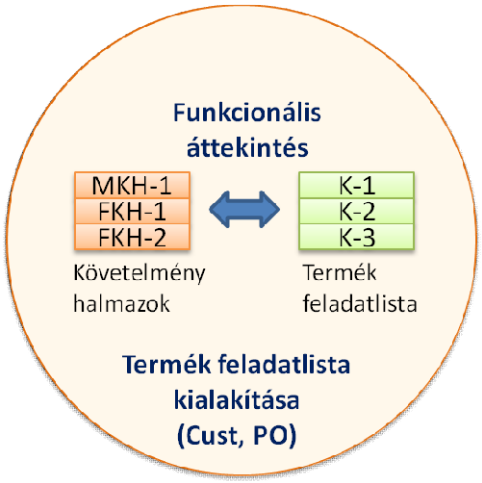
A Military Scrum szakaszai párba állíthatók az üzemeltetői biztonsági terv fejezeteivel az alábbiak szerint.

1. kockázatelemzés → koncepcionális előtervezési szakasz
    a. biztonsági kockázatok követelményhalmazokhoz rendelése
        i. funkcionális követelményhalmazok bővítése
        ii. műszaki követelményhalmazok bővítése
2. kockázatkezelés → koncepcióalkotás szakasza

a. biztonsági követelmények és a reagálás meghatározása a termék feladatlistában

3. eszközrendszer → fejlesztés szakasza
    a. biztonsági aspektusból bővített tesztforgatókönyvek elkészítése
    b. automatizált tesztek kialakítása
    c. szakterület specifikus eszközök alkalmazása

Lényeges szempont a fejlesztett szoftver alapú szolgáltatás leendő üzemeltetési környezetével kapcsolatos biztonsági követelmények beágyazása a műszaki követelményhalmazokba, célszerű megvizsgálni a biztonság és a megbízhatóság, skálázhatóság, robosztusság, karbantarthatóság, fenntarthatóság viszonyát a követelmények megfogalmazásakor. A fejlesztett szoftverrel szemben támasztott minden követelmény a koncepcióalkotási szakaszban kerül rendszerezésre a termék feladatlistában.



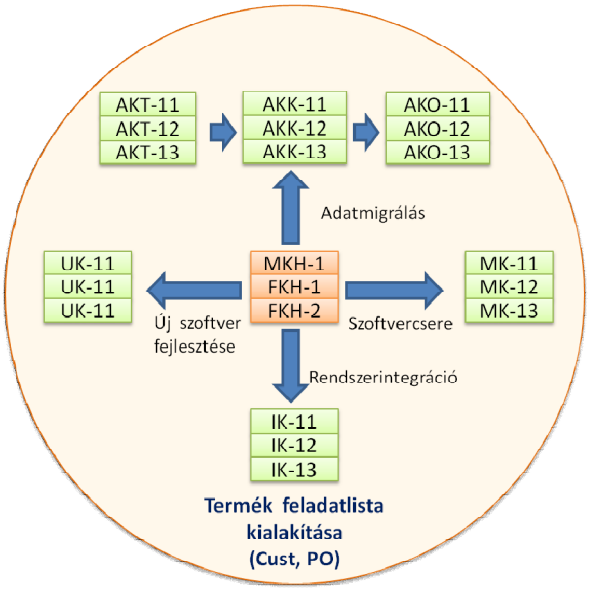
41. ábra A Military Scrum 2. tervezési szintje – koncepcióalkotás (saját szerkesztés)

4.3.6 Nyomon követési infrastruktúra

A Military Scrum a nyomon követési infrastruktúrának való megfelelés érdekében ajánlást tesz a különböző jellegű projektek követelményeinek azonosítására a termék feladatlistában. Az azonosítási rendszertől el lehet térni, ekkor a számképzést a PMS-ben rögzíteni kell.

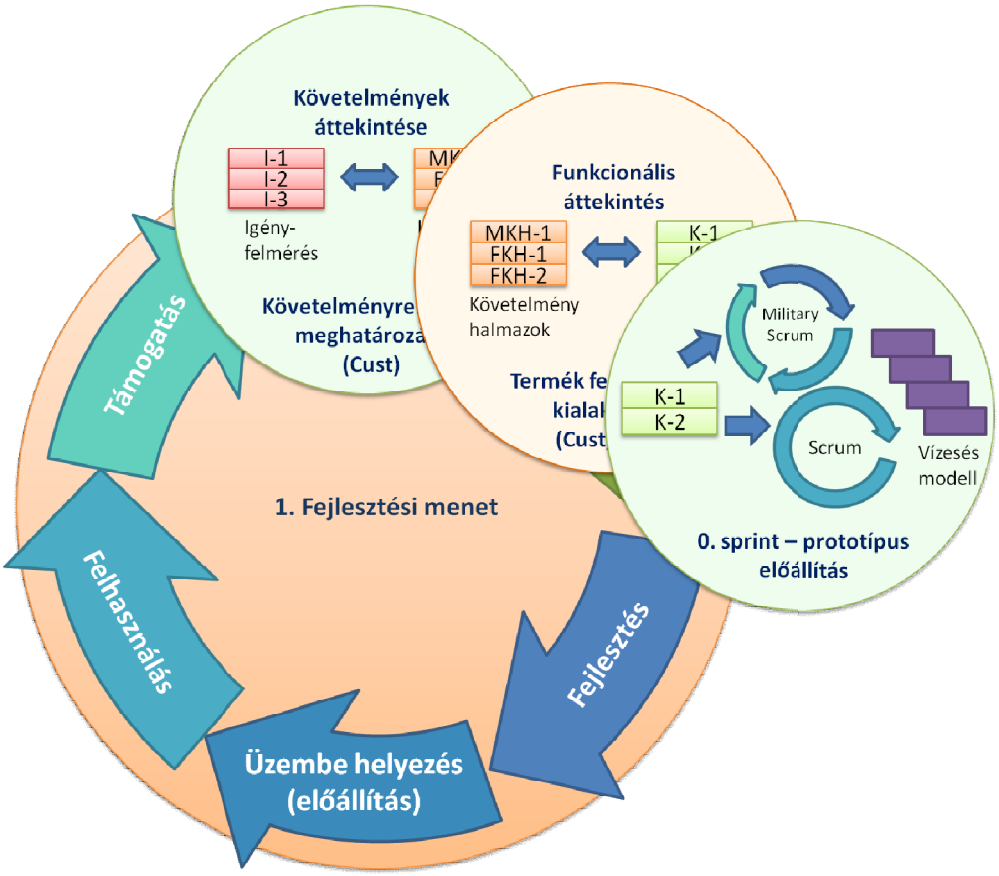
- Új szoftver fejlesztése, új követelmény (UK): *PA-UK-n (n >0)*
- Szoftvercsere, migrált funkciók (MK): *PA-MK-n (n >0)*
- Adatmigrálási követelmény táblákra (AKT): *PA-AKT-n (n >0)*
- Adatmigrálási követelmény kulcsokra (AKK): *PA-AKK-n (n >0)*
- Adatmigrálási követelmény oszlopokra (AKO): *PA-AKO-n (n >0)*
- Rendszerintegráció, integrálási funkció (IK): *PA-IK-n (n >0)*

A 42. ábrán a termék feladatlista kialakítása során javasolt azonosító képzési szabályok láthatók.



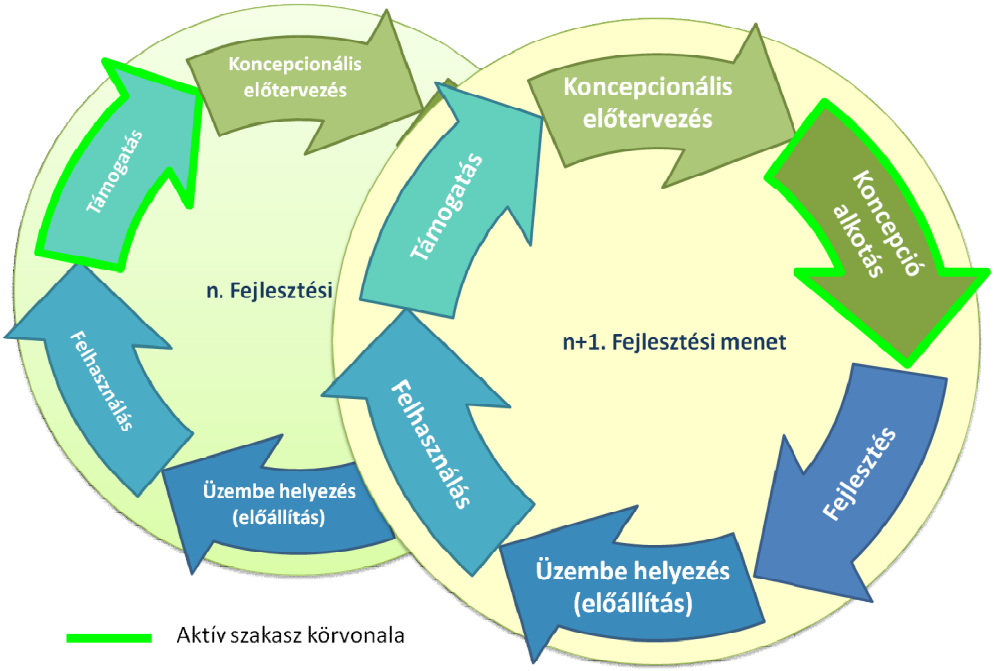
42. ábra Military Scrum termék feladatlista azonosító képzés (saját szerkesztés)

A termék feladatlista előállításban a megrendelőnek és a projekt termékfelelősének kell részt vennie. Az 1. fejlesztési menet koncepcionális szakaszai tovább bővülhetnek a prototípus előállítással, a folyamatot a 43. ábra mutatja be.



43. ábra Military Scrum 1. fejlesztési menetének koncepcionális szakaszai (saját szerkesztés)

Az 1. fejlesztési menetet követő iterációkban a követelmények áttekintése és a funkcionális áttekintés folyamatos tevékenységekként végzendők. A döntéshozatal támogatásához a prototípusok előállítása a későbbiek során is alkalmazható módszer.



44. ábra Military Scrum egymást követő fejlesztési menetek, a zöld szín jelöli az aktív szakaszokat (saját szerkesztés)

A fejlesztett szoftver alapú szolgáltatás teljes élettartama alatt előfordulhat, hogy a fejlesztési menetek időbeni átfedéssel rendelkeznek. Elképzelhető, hogy az előző fejlesztési menet aktív szakaszával párhuzamosan megkezdődik az n+1. menet koncepcióalkotási vagy fejlesztési szakasza is. Az átadás előtti időszakban a laboratóriumi körülmények mellett végzett tesztek tapasztalatai, az azt követő időszakban a felhasználás során felmerült változtatási igények, hibák követelményekké alakítása valósul meg a soron következő menetben a követelményrendszer karbantartásával, a termék feladatlista frissítésével, elemeinek újra priorizálásával. A módszer segítségével a Military Scrum követelményelemzési dokumentumai a szoftver alapú szolgáltatás teljes élettartama alatt konzisztensek maradnak és garantálják a nyomon követési infrastruktúra számára az egyedi azonosító képzést.

A módszertan hatékony alkalmazásához a működést lehetővé tévő rendszerek közül a projektvezetésnek és a szakértői támogatásnak kell rendelkeznie a megfelelő érettségi szinttel, amelynek meghatározását az érintett szakaszokban a PMS tárgyalja a minőségbiztosítási kérdések szekcióban.

4.4 SZOFTVERFEJLESZTÉSI TECHNIKÁK

A követelményhalmazok és a termék feladatlista kialakításán túl természetesen a Military Scrum alkalmazása során is szükség van a hagyományos értelemben vett rendszerszervezésre, ez a tevékenység a fejlesztési szakaszban végrehajtott sprintek részeként valósul meg. A fejlesztési szakasz első lépése a sprint tervezés, amikor a termék feladatlista alapján alacsony szintű fejlesztési feladatok keletkeznek, ekkor a fejlesztőcsapat és a terméktulajdonos együtt vesznek részt a rendszerszervezésben. A másik rendszerszervezési lépés, maga az új funkció implementálása, amit megelőz a tesztesetek azonosítása, ekkor az adott szoftverfejlesztő és a terméktulajdonos együtt végeznek rendszerszervezési feladatokat. Miért van szükség arra, hogy ez a két szerepkör együtt végezzen rendszerszervezési tevékenységet? A részletes tesztforgatókönyvek segítségével ér véget a Military Scrum tervezési folyamata, az utolsó tervezési lépésben nem csak a követelmények megvalósítása a szempont, figyelembe kell venni a rendszer jelenlegi képességeit és mérlegelni kell a változtatások hatásait a teljes fejlesztett szoftver egészét vizsgálva. A módszertan szoftverfejlesztési technikái többek között ezt a fajta szisztematikus rendszeranalízist segítik elő.

A Military Scrum módszertannal a fejlesztés szakaszában több fejlesztési sprintet is meg lehet valósítani. A fejlesztési szakasz bemenete a koncepcióalkotási szakasz eredményeként létrejövő termék feladatlista, amelyben az adott fejlesztési menet vonatkozásában megfelelő részletességgel kidolgozott feladatok szerepelnek. A fejlesztési szakasz végeredményének egy kiadható, legalább tesztüzemre alkalmas verziónak kell lennie. A cél eléréséhez a Military Scrum módszertani és technológiai elvárásokat fogalmaz meg.

4.4.1 Módszertani követelmények

- A termék feladatlistában szereplő feladatok implementálása Scrum módszertan alkalmazásával történjen.
- Az egyes fejlesztési feladatok egyedi azonosító képzése a következőképp valósuljon meg: *PA-K-n-F-m. (PA-K-n, n>0)* a termék feladatlista elemének azonosítója, *m>0* az adott követelményt megvalósító részfeladat vagy funkció sorszáma.
- A fejlesztők végezzék a fejlesztési feladatok részletes meghatározását és a feladatokra vonatozó erőforrásbecslést a sprint tervezéseken.

- A fejlesztési feladatokra vonatkozó erőforrásbecslés a Scrum póker technika alkalmazásával történjen.
- A fejlesztési sprintek zárásakor a fejlesztők prezentálják az általuk készített funkciókat.
- Sikertelen sprintek esetén kötelezően, illetve a fejlesztési szakasz záró elemeként kerüljön sor a Scrum módszertanból ismert retrospektív megbeszélésre.
- A felsorolt módszertani elemek végrehajtásáért a Military Scrum minőségirányítása felelős.

4.4.2 Technológiai követelmények

- A fejlesztett rendszer forráskódja központi verziókövető rendszerben kapjon helyet.
- A fejlesztések a folyamatos integráció (CI) technikájának alkalmazásával valósuljanak meg.
- Minden fejlesztési részfeladat elvégzése után valósuljon meg a rendszer integritásának ellenőrzése és forráskód áttekintése más fejlesztők által is.
- A tesztforgatókönyvek alapján elkészült automata tesztek megléte és a produkciós kód lefedettség ellenőrzése is képezze részét a folyamatos integrációnak.
- Az egység tesztek és az integrációs tesztek kiértékelése is valósuljon meg a CI részeként.
- A fejlesztett szoftver folyamatos integrációja környezetfüggetlen módon legyen megvalósítható megfelelő konfiguráció menedzsment mellett – a szoftver képességeinek a tesztelése a feladat, nem pedig a futtató környezeté.
- A felsorolt technológiai elemek végrehajtásáért a Military Scrum minőségirányítása felelős.

Az imént felsorolt módszertani és technológiai előfeltételek mellett a fejlesztési szakasz során megvalósítandó feladatok mennyiségét a fejlesztőcsapat rendelkezésre álló fejlesztési kapacitása határozza meg, amelynek kiszámítása a Scrum módszertan fókusz faktor technikájával történik. A módszer lényege, hogy a fejlesztők munkanapjai súlyozva képezik részét a fejlesztőcsapat összesített fejlesztési kapacitásának. A becslések folyamatosan végrehajtott implementálási feladatokat feltételeznek nyolc munkaórában (1 SP), nem veszik figyelembe az egyéb munkahelyi teendőket. A projektvezetés feladata a teljes fejlesztési menet nyomon követése és támogatása,

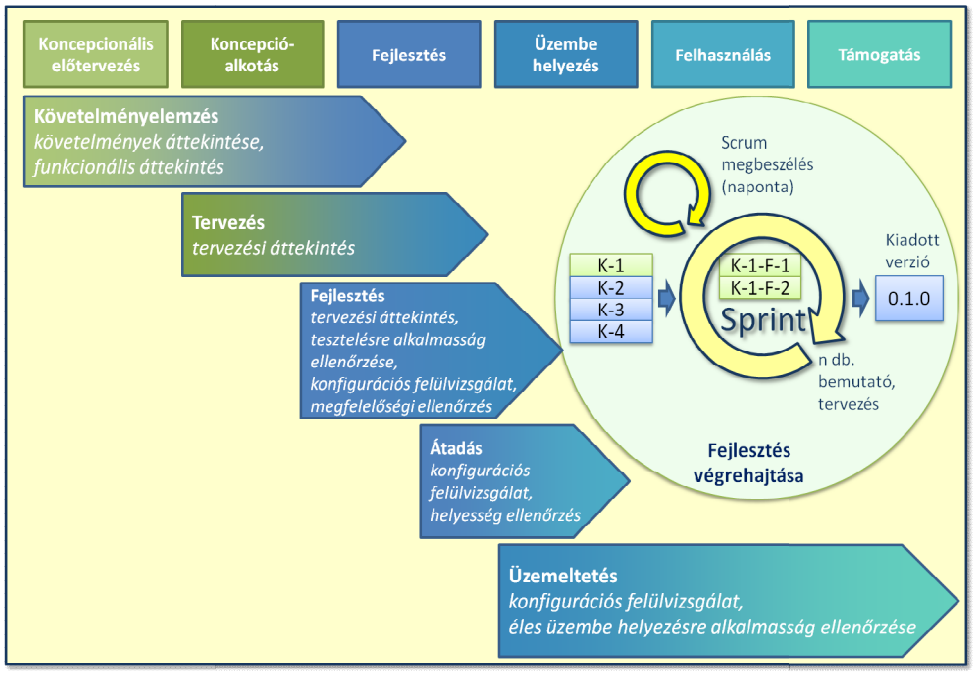
ezért a projektvezetés szereplői nem kerültek feltüntetésre az alábbi táblázatban, amely a fejlesztéshez köthető tevékenységek és projektszereplők kapcsolatát mutatja.

	Tesztforgatókönyvek elkészítése	Fejlesztési feladat elvégzése	Manuális tesztelés	Szoftverbemutató
Cust				x
PO	x		x	x
Dev	x	x		x

19. táblázat A Military Scrum fejlesztési szakaszának feladatai és szereplői (saját szerkesztés)

A szakaszokra bontott életciklus lehetővé teszi a szoftvertechnológiai folyamat akkurátus tárgyalását, ugyanakkor a 45. ábra alapján az látható, hogy a szoftverfejlesztés esetében a követelmények elemzése, megfelelő szintű finomítása három életciklus szakaszt érintő folyamat. A tervezés már megkezdődik a koncepció alkotás szakaszában és még az üzembe helyezés szakasza is adhat visszajelzést a tervezés számára. A fejlesztés a prototípusok előállításával megkezdődhet a koncepció alkotás során és az üzembe helyezési szakasz teljes támogatása is rejthet fejlesztési feladatokat. Az átadási folyamatok előkészítése a fejlesztési szakaszban kezdődik meg és a felhasználással záródik, míg az üzemeltetetési feladatok a fejlesztést követően kapnak nagyobb hangsúlyt.

A fejlesztés szakaszában a tervezési áttekintés a sprintek tervezésekor történik meg, míg a tesztelésre alkalmasság ellenőrzése és a konfigurációs felülvizsgálat a CI által valósul meg a forráskód minden változását követően. A megfelelőségi ellenőrzés a funkciók elkészültét követően a CI és a tesztelést végző szakemberek által zajlik.



45. ábra A Military Scrum és a NATO SLCM folyamatlépéseinek kapcsolata (saját szerkesztés)

4.4.3 Tesztvezérelt fejlesztés

A tesztvezérelt szoftverfejlesztés (a továbbiakban: TDD[76]) egy olyan szoftverfejlesztési technika, amely egy nagyon rövid fejlesztési ciklus ismétlésére támaszkodik. Első lépésben a funkciók követelményeit tesztesetekké alakítják, majd a szoftver olyan formában kerül továbbfejlesztésre, hogy megfeleljen az újonnan definiált teszteseteknek. Ez a módszer megakadályozza, hogy olyan programrészek kerüljenek a szoftverbe, amelyek nem felelnek meg az új és a korábban támasztott követelményeknek. A munkafolyamat lépései az alábbiak

1. *Új teszteset hozzáadása* – a tesztvezérelt fejlesztésben minden új funkció implementálása teszteléssel kezdődik. Ez egy olyan teszt elkészítését jelenti, amely tömören lefedi az új funkció működésére, illetve a létező funkció kibővítésére vonatkozó követelményeket. Az adott funkcióhoz tartozó teszt elkészítéséhez a fejlesztőnek világosan meg kell értenie a specifikációt és a funkcióval szemben támasztott követelményeket. A követelmények lefedéséhez és a kivételek kezeléséhez használati esetekre és felhasználói tesztforgatókönyvekre van szükség, ezek alapján már a tesztek elkészíthetők az adott szoftveres környezethez illeszkedő tetszőleges tesztelési keretrendszerben. A végrehajtott változtatások egy létező teszt módosításai is lehetnek. Ebben az esetben a fejlesztő a követelményekre koncentrál az forráskód elkészítése előtt, ez egy apró, ugyanakkor fontos és meghatározó különbség a tesztvezérelt fejlesztés és a hagyományos fejlesztést követő egység tesztekkel történő teszteléshez (unit testing) képest.
2. *Tesztek újbóli kiértékelése, az új teszt bukásának ellenőrzése* – ez a lépés azt igazolja, hogy az automatizált tesztkörnyezet jól működik, mert ellenőrzi, hogy az új teszt nem integrálható a tesztkörnyezetbe anélkül, hogy a szoftver forráskódján változtatni kellene. Ezzel a módszerrel kizárható, hogy a szükséges viselkedés már létezik vagy az új teszt hibás és mindig átmegy az ellenőrzésen. Jó esetben az új tesztnek az elvárt ok miatt buknia kell, ez a lépés növeli a fejlesztő bizalmát az új tesztben.
3. *Az új funkció implementálása* - a következő lépésben olyan forráskódot kell készíteni, amely az elbukó tesztet megjavítja. Az ebben a fázisban írt új forráskódnak nem kell tökéletesnek lennie, nem kell a legmegfelelőbb megoldást al-

[76] TDD – Teszt vezérelt szoftverfejlesztés [108], az angol Test Driven Development kifejezés szavainak kezdőbetűiből.

kalmaznia, de már a követelményeknek meg kell felelnie – a teszt nem bukhat. Ez a jelenség ebben a lépésben elfogadható, mert a forráskód minősége még javítva lesz az 5. lépésben. Ebben a fázisban a forráskód egyetlen célja, hogy kielégítse az új tesztben ellenőrzött új követelményeket. A programozást végzőnek tilos a teszt által ellenőrzött funkcionalitáson túlmutató forráskódot írnia.

4. *Tesztek újbóli kiértékelése* – ha minden teszt helyes eredményt mutat, akkor a programozó biztos lehet benne, hogy az új forráskód megfelel a követelményeknek, és az új funkció nem sérti a meglévő funkciókat, a rendszer integritását. Amennyiben ez nem teljesül, akkor az új funkció megvalósítását addig kell finomítani, amíg ez meg nem történik.
5. *A forráskód átszervezése (refactor)* - A növekvő kódbázist rendszeresen tisztítani kell a tesztvezérelt fejlesztés során. Az új forráskódot át lehet helyezni arról a helyről, ahol a teszt kielégítésekor helyzetük, oda ahova logikailag jobban illeszkedik. A forráskódban található ismétlődéseket el kell távolítani. Egy alkalmazott objektum-, osztály-, modul-, változó- és metódusnévnek egyértelműnek kell lennie, amik az adott egység céljára és felhasználásra utal, ugyanis extra funkcionalitást került hozzáadásra. A megvalósított funkciók számának növekedésével, a metódusok törzse egyre nagyobbá válhat, ahogy az osztály definíciók is egyre hosszabbak lehetnek. Célszerű őket szétválasztani, megfelelően elnevezni, ezzel növelve az olvashatóság és karbantarthatóság mértékét, ami később a szoftver életciklusában egyre értékesebb lesz. A származtatási hierarchia folyamatos felülvizsgálata is célszerű, elképzelhető, hogy valamilyen ismert tervezési minta kerül azonosításra az áttekintés során. Léteznek speciális és általános iránymutatások a forráskód átszervezéséhez és a tiszta forráskód (clean code) elkészítéséhez [109]. A folyamatos teljes körű tesztelésnek köszönhetően az egyes forráskód átszervezési fázisok során a fejlesztő biztos lehet benne, hogy a folyamat nem módosítja a meglévő funkciókat. A forráskód ismétlődések eltávolításának koncepciója fontos része a szoftverek tervezésének. A tesztek és szoftver forráskódjában található ismétlődések eltávolítására is vonatkozik.
6. *Ismétlés* - a rendszer funkcionalitásának bővítését egy újabb teszt elkészítésével megismételhetjük. Kis lépésekkel célszerű haladni, a tesztfuttatások között legfeljebb 1-10 módosítás végzendő. Ha az új forráskód nem elégíti ki elég gyorsan a követelményeket vagy egyéb tesztek elromlanak, a változtatások ha-

tására a programozónak vissza kell vonnia a változtatásokat és vissza kell állnia egy korábbi állapotra. A folyamatos integráció segíthet egy adott állapotra történő visszaállításban.

A tesztforgatókönyvek elkészítésével a folyamat szintű elvárások azonosítása valósítható meg, azonban a megfelelő szintű szoftvertervezés megvalósításához ez a technika önmagában nem elegendő. A TDD önálló alkalmazása redundáns forráskódot, hasonló viselkedésű, mégis teljesen szeparált forráskódot eredményezhet még a refaktorálási lépések végrehajtása mellett is. A módszer helytelen alkalmazása egyenesen rombolhatja a rendszer integritását.

A Military Scrum ezért a fejlesztési szakaszban előírja a fogalomtér alapú tervezési technika, a DDD[77] alkalmazását is. A módszer ellentéte a TDD-nek, mert itt a folyamat leírásokkal szemben az adott követelményrendszer fogalmainak azonosítása és leképezése a cél.

4.4.4 Fogalomtér vezérelt tervezés

A fogalomtér vezérelt tervezés (DDD) összetett követelmények kezelésére szolgáló módszer, amely a tervezési folyamat során keletkező fogalmak azonosításán alapszik. A fejlődő fogalomtér a fejlesztés előrehaladásának mutatója, a technika helyes alkalmazásával az azonosított és rendszerezett üzleti fogalmak képezik a fejlesztett szoftverben használt csomagok és osztályok elnevezéseit.

- Elsődleges feladat az üzleti fogalmak és a köztük fennálló logikai kapcsolatok azonosítása.
- Második lépésben az azonosított üzleti fogalomtér képezi a tervezés alapját.
- A fejlesztési folyamat során a feltárt tervezési problémák javítását, a modell finomítását először a fogalomtérben kell elvégezni, majd leképezni azt a fejlesztett szoftver szintjére.

A fogalomtér vezérelt tervezés az alábbi modellt határozza meg.

1. *Környezet* – a követelményrendszerben szereplő alapfogalmak azonosítása meghatározása, egységesítése.
2. *Fogalomtér* - az azonosított fogalmak, a fogalmak közötti kapcsolatok, hatásuk egymásra és közösen végzett tevékenységük. A felhasználók a fejlesztett szoftvert, annak fogalomterében használják.

[77] DDD – Fogalomtér vezérelt tervezés, angol Domain Driven Design kifejezés szavainak kezdőbetűiből [110]

3. *Modell* – egy absztrakt rendszer, amely összeköti a fogalomtér bizonyos elemeit, felhasználható a fejlesztett szoftverrel szemben támasztott követelmények kielégítéséhez a folyamatok által érintett fogalmakat modellezve.
4. *Mindenütt jelenlévő nyelv* – az azonosított fogalmak közös fogalomrendszert biztosít a fejlesztőcsapat számára a fejlesztés során. A megrendelő, a termék-tulajdonos és fejlesztőcsapat is egyértelműen tud kommunikálni az adott szoftver képességeit, funkcióit illetően a közös nyelv kialakítását követően.

A technika helytelen alkalmazása túlterhelt modellek kialakításához vezethet, ami a fejlesztett szoftverrel szemben támasztott folyamat szintű elvárások megvalósítását gátolja, lassítja. A megfelelő egyensúly megteremtéséhez mindkét technika együttes alkalmazására van szükség. A sprint tervezések során a termék feladatlistában szereplő követelményekhez tartozó folyamatok 1. szintű tervezése valósul meg. A sprint feladatlista elemei struktúrájukat tekintve megegyeznek a termék feladatlista elemeivel. Azonosító képzésük eltérő, illetve fejlesztési sprintenként külön dokumentumokban vezetendők.

	Sprint tervezés	**Tesztforgatókönyvek elkészítése**	**Fejlesztési feladat elvégzése**	**Kódellenőrzés**
TDD	1. szintű tervezés	2. szintű tervezés	-	felülvizsgálat
DDD	1. szintű tervezés	-	2. szintű tervezés	felülvizsgálat

20. táblázat A fejlesztési szakasz és a TDD, DDD technikák kapcsolata (saját szerkesztés)

A sprint tervezések során a követelményekhez kapcsolódó fogalmak és rájuk épülő folyamatok megfelelő mélységű tisztázása szükséges – a DDD és a TDD együttes 1. szintű alkalmazásával. A fejlesztési feladatokat megelőző konkrét tesztforgatókönyv előállítása képezi a TDD technika alkalmazása során a 2. szintű tervezést. A DDD technika 2. szintű tervezése a fejlesztési feladat elvégzése során valósul meg, amikor az elkészült tesztforgatókönyvek, illetve az aktuális modell összevetése történik meg a modell bővítésének érdekében – ha az szükséges a *refactoring* technika alkalmazása mellett. A gépi és a manuális kódellenőrzés szavatolja a technikák helyes alkalmazását.

A Military Scrum által előírt TDD és DDD technikák több szintű és ellenőrzött alkalmazásával a fejlesztett szoftver fogalomtere és forráskódja is konzisztensen tartható. Az eljárás garantálja a megfelelő érettségi szintű szoftverfejlesztést a rendszer átadása előtt, majd ezt követően az üzembe helyezett rendszer változáskezelését. A PMS tartalmazza a fejlesztés szakaszában vizsgálandó kérdéseket és kérdések eldöntéséhez nyújtott eszközöket is.

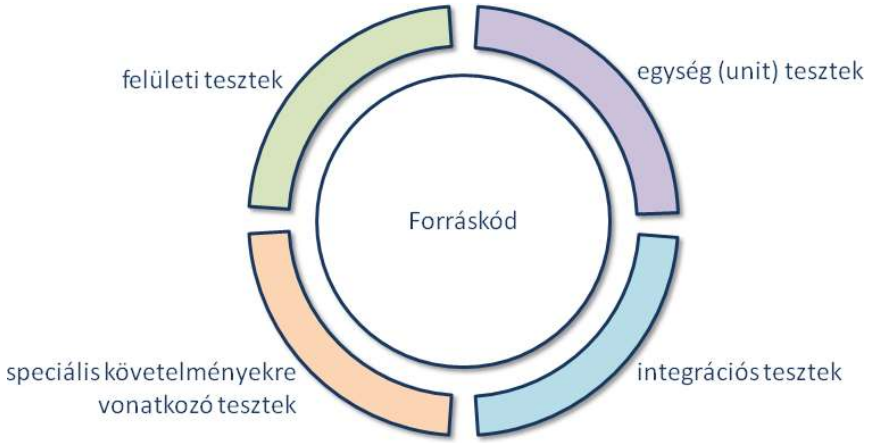
4.4.5 Evolúciós szoftverfejlesztés

Egy Military Scrum által fejlesztett szoftver esetében a módosítások előkészítését és a fejlesztést a terméktulajdonos és a fejlesztők együtt valósítják meg. Egy új követelmény elemzését, részfeladatokra bontását is közösen végzik, meghatározzák az új és megváltoztatandó használati eseteket, majd ezt követően változtathatnak a rendszer forráskódján. Az addigi működést lefedő automatizált tesztek egyértelmű visszajelzést adnak a lefektetett követelmények és az új rendszer különbségeiről, amelyekről már egyszerű eldönteni, hogy elvárt működésről vagy hibáról van szó.

A rendszer forráskódjában történő bármely változtatásról a CI azonnal visszajelez, így a fejlesztett szoftver a virtuális térben egy élő entitássá válik, amelynek pillanatnyi érettsége mérhetővé és meghatározhatóvá válik.

A helyesen megvalósított TDD technika alkalmazásakor a fejlesztők az új követelmények megvalósításához először a futtatható teszteket készítik el és utána értékelik ki a rendszer működését. Ebben az esetben, a virtuális térben a kódnak az a része éli túl a változtatásokat, amely az új követelménynek eddig is megfeleltek és a „rossz" kód kipusztul a rendszerből. A kipusztulást a fejlesztők „okozzák", amikor kibővítik a rendszer működését és az új követelményre helytelenül működő kódot átalakítják. Ez a fejlesztési módszer egy merőben új, szoftverekre értelmezett evolúciós folyamat lehetőségét teremti meg.

Ha az automatizált tesztek kialakításakor a kialakítandó katonai képesség követelményrendszere van szem előtt tartva, akkor az evolúciós szoftverfejlesztés technikájával a katonai alkalmazás egy új szintje érhető el. A kieső számítás kapacitás kezelése, alkalmazott algoritmusok közötti váltás, rejtőzködő behatolások, információlopás detektálása és a reagálási képességek kialakítása a fejlesztett szoftveren belül. A fejlesztett szoftver és az őt körülvevő evolúciós környezet a 46. ábrán látható.

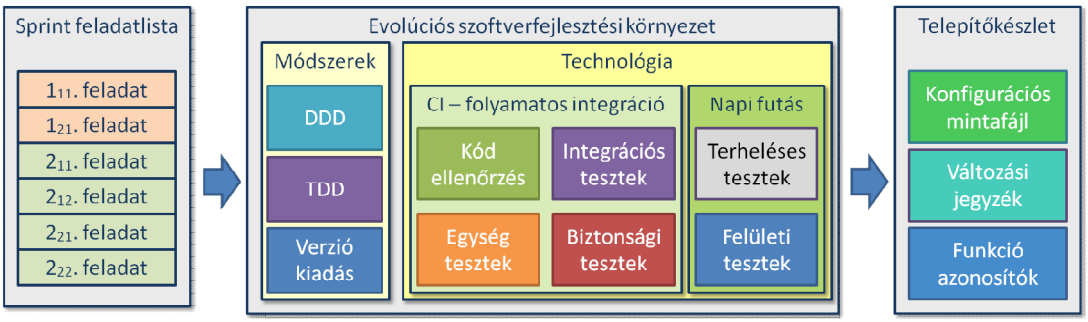


46. ábra Automatizált tesztekkel lefedett forráskód (saját szerkesztés)

Az automatizált és a manuális tesztelés ugyanolyan fontos a sprint időtartama alatt, mert az egyik a konkrét funkciót fedi le, a másik a funkciót és a rendszer integritását vizsgálja együttesen. Utóbbi különösen akkor igaz, ha szakértő végzi a manuális tesztelést. Ez a fajta megközelítése a szoftverek előállításának aktív részvételt követel meg a megrendelőktől is, a probléma mély megértésének folyamata közös tevékenységgé válik. A módszer legnagyobb előnye az, hogy az elkészült termék a technológiai és az üzleti korlátok között is a lehetséges maximumot fogja produkálni, mert folyamatos tesztelés alatt áll üzleti és technológiai oldalról egyaránt. A módszer lehetőséget biztosít arra, hogy a széleskörű funkcionalitás ötvöződjön a megfelelő technológiai háttérrel.

A módszer lehetővé teszi a műszaki alapok frissítését, mert az evolúciós környezet segítségével ellenőrizhetővé válik a fejlesztett szoftver összes képessége. Fontos kiemelni, hogy a technika akkor működik megfelelően, ha kizárólag a fejlesztett szoftver képességeit ellenőrzi – kapcsolódó rendszerek nélkül. Amennyiben más rendszerekkel történő integrációs folyamatok kialakítása a cél, akkor az utánzás (mock) technikájával kell a rendszerek által szolgáltatott adatokat az evolúciós tesztkörnyezet részévé tenni.

A módszertan hatékony alkalmazásához a működést lehetővé tévő rendszerek közül a projektvezetésnek, a katonai és fejlesztői szakterületek által nyújtott támogatásnak, a fejlesztési és tesztelési eszközöknek, környezeteknek kell rendelkezniük a megfelelő érettségi szinttel, amelynek meghatározását az érintett szakaszokban a PMS tárgyalja a minőségbiztosítási kérdések szekcióban.



47. ábra A Military Scrum evolúciós szoftverfejlesztési környezete (saját szerkesztés)

Az evolúciós szoftverfejlesztési környezet egyaránt épít módszerekre és technológiára. A verziókiadás a fejlesztési szakasz végén valósul meg, abban az esetben, ha technológia komponensek helyes működést jeleznek vissza a fejlesztett szoftver pillanatnyi érettségével kapcsolatban, ekkor a verziókiadás módszerét kell alkalmazni. A módszer ismertetése előtt a biztonsági tesztelés alapelvei következnek.

4.4.6 Biztonsági kockázatok kezelése

A fejlesztett szoftver biztonságának fokozása a különböző belső és külső eredetű kockázatok kezelésével valósítható meg. Az alábbi felsorolás néhány fejlesztéshez köthető kockázatot mutat be.

1. Belső kockázatok
    a. inkonzisztens követelményrendszer
    b. helytelen üzleti folyamatok, rosszul értelmezett követelmények
    c. pontatlan vagy lehallgatható kommunikáció az interfésszel illesztett belső rendszerekkel
    d. rendszeren belüli sérülékenységek: a jogosultsági rendszer és az elérhető funkciók hibás összerendelése – nem megfelelően implementált jogosultsági szabályok.
2. Külső kockázatok
    a. megnövekedett számú felhasználó kezelése
    b. pontatlan vagy lehallgatható kommunikáció az interfésszel illesztett külső rendszerekkel
    c. külső sérülékenységek: adatlopás, károkozás, stb.

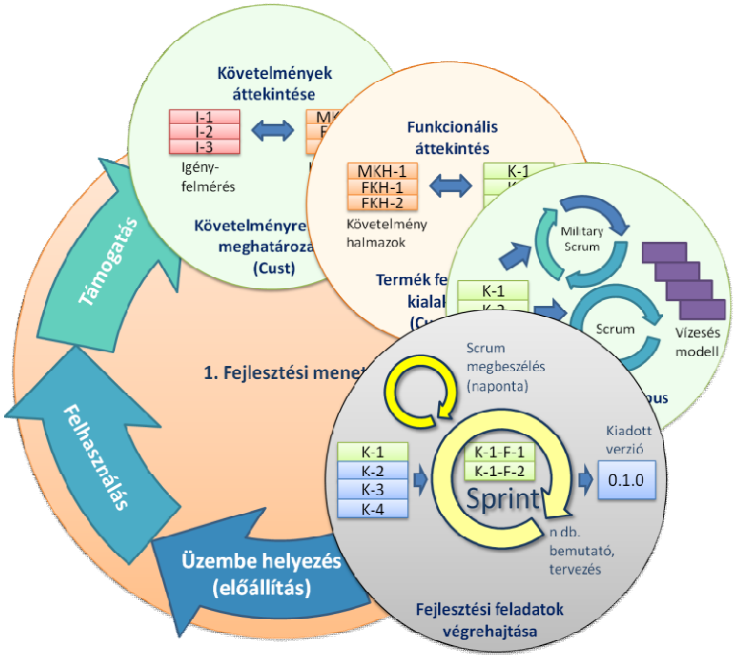
A Military Scrum által meghatározott követelményelemzési módszer strukturált és jól definiált követelményeket biztosít a termék feladatlistában, azonban további kockázatokat rejthetnek magukban a hibásan, helytelenül megvalósított funkciók. A jelenség oka lehet a követelményrendszer alapján nem egyértelműen meghatározott folyamatok bizonytalan megvalósítása. A módszertan a sprint tervezések és tesztforgatókönyvek előállítása során is aktív részvételt vár el a terméktulajdonostól, akinek a szaktudása elengedhetetlen a funkcionális megfelelőség eléréséhez. A teszt forgatókönyv alapú integrációs tesztek helyességét, így a belső kockázatok csökkentését a fejlesztőkkel közösen végzett specifikációs eljárás garantálja.

Amennyiben a projekt koncepcionális szakaszai során megfelelő színvonalú biztonsági kockázatelemzés valósult meg, amelyhez a módszertan dokumentumai megfelelő keretet biztosítanak, akkor a reagálás lehetséges a különböző biztonsági eseményekre a fejlesztett szoftver szintjén is. A módszertan előírja az észlelt biztonsági események naplózását a fejlesztett szoftverben, azok jelzését az üzemeltetői diagnosztikai eszközök számára. A CI részeként megvalósítandó biztonsági tesztek segítségével automatizált sérülékenység-vizsgálat valósítható meg, amely során a fejlesztett szoftver gépi és felhasználói felületei kiértékelhetők.

4.4.7 Verzió kiadás és nyomon követés

A fejlesztési szakasz záró lépése a sprintek során előállított változtatásokat tartalmazó szoftververzió kiadása. A nyomon követési infrastruktúrának és konfiguráció menedzsment való megfelelés miatt célszerű az eljárás lépéseit a módszertan szintjén is meghatározni. Az alábbi felsorolás a verzió előállítással kapcsolatos követelményeket tartalmazza.

- A verziókiadás folyamata automatikus eljárással valósuljon meg a központi verziókövető rendszerben vezetett forráskód alapján.
- A verziószámok képzésének mintája: FOSZAM.ALSZAM
  - FOSZAM – megegyezik az aktuális Military Scrum menet számával.
  - ALSZAM – 0-val indul minden új FOSZAM esetén, a támogatott, üzembe helyezett verzióhoz tartozó sürgős hibajavítások számára fenntartott számláló.
- A kiadott verzió tartalmazza a konfigurációs mintafájlt.
- A kiadott verzióhoz tartozzon változási jegyzék, amelynek előállításáért a terméktulajdonos felel, vezetése a központi verziókövetőben valósuljon meg.
- A kiadott verzióhoz tartozzon egy szöveges állomány, amely tartalmazza a fejlesztési szakasz során fejlesztett funkciókhoz tartozó követelmény azonosítókat – a módszer segítségével a kiadott verzió és implementált követelmények kapcsolata fenntartható a nyomon követési infrastruktúra számára.



48. ábra A Military Scrum 1. fejlesztési menetének fejlesztési szakasza (saját szerkesztés)

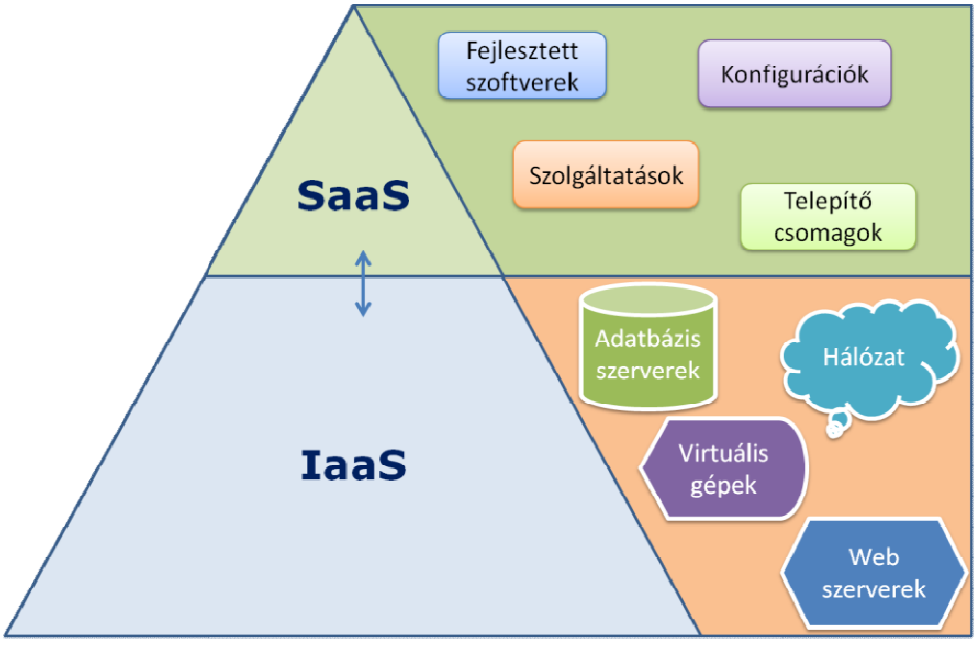
A fejlesztési szakaszban végrehajtandó folyamatokat nem befolyásolja az, hogy átadás előtti vagy átadás utáni időszakban van a módszertannal végrehajtott projekt.

4.5 ÜZEMELTETÉSI FOLYAMATOK TÁMOGATÁSA

A szoftverfejlesztéshez köthető folyamatok nem érnek véget a Military Scrum fejlesztési szakaszában, mert az átadási folyamat, az üzembe helyezés, a felhasználás és a támogatás szakaszai is eredményezhetnek szoftverfejlesztési feladatokat. A módszertan a célja a szoftver alapú szolgáltatások kialakításához és fenntartásához kapcsolódó szoftverfejlesztési folyamatok támogatása, így a módszertan eljárásokat és technológiákat ír elő a fejlesztés utáni szakaszokra is.

A telepítő csomagok szabályozott és nyomon követhető előállításán túl további megkötések szükségesek a szoftver alapú szolgáltatások fenntartható üzemeltetéséhez. A szabályozott verzió kiadás megteremti a fejlesztési és üzemeltetési folyamatok összehangolásának lehetőségét, mert az új igények és a változáskezelés eredményeként létrejövő verziók megegyező üzemeltetési paraméterekkel rendelkeznek.

A probléma megértéséhez látni kell, hogy egy sikertelen telepítés, rendszer leállás, lassulás, bizonytalanul működő funkció mögött rejlő hiba vagy jelenség azonosítása nem feltétlenül egyszerű feladat. Az emberi tényező szerepe jelentős, a problémákat okozhatja mulasztás, tévesztés, hiba, vagy kibertámadás is.

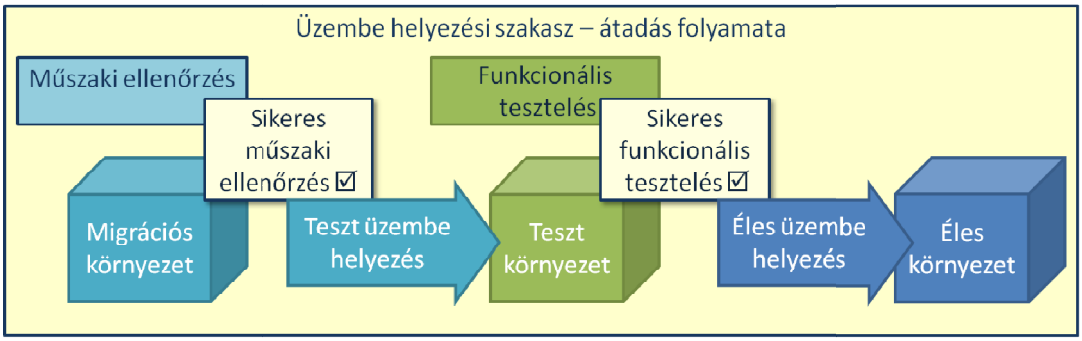


49. ábra SaaS és IaaS szolgáltatási rétegek komponensei (saját szerkesztés)

A 49. ábrán a szoftver és az infrastruktúra alapú szolgáltatások gyakori fogalmai szerepelnek. Egy hiba vagy egy biztonsági incidens esetében az ábrán szereplő komponensek bármelyike érintett lehet, az okok keresése nem egyszerű feladat, ezért a Military Scrum technológiai és módszertani elvárásokat támaszt a fejlesztett szoftver alapú szolgáltatás üzemeltetési és támogatási folyamataival szemben.

4.5.1 Telepítési folyamatok

Helyesen alkalmazott módszertan esetén a fejlesztési szakaszok végén, megfelelőségi ellenőrzésen átesett, laboratóriumi körülmények között működő szoftver verzió kiadása valósul meg. Ha a fejlesztett szoftver képességei és a projekt ütemezése lehetővé teszi, akkor megkezdődhet az átadási folyamat az első próba telepítéssel, ekkor a fejlesztett szoftver már megjelenik a leendő éles üzemeltetési környezetben is, de még nem éles felhasználási céllal. A telepítő csomag publikálása ideális esetben már a kezdetektől a folyamatos szállítás (CD) technikájával valósul meg. Az átadási folyamat a Military Scrum esetén több fázisból áll össze: *műszaki ellenőrzés, teszt üzembe helyezés, funkcionális tesztelés*, *éles üzembe helyezés*.



50. ábra A Military Scrum átadási folyamata (saját szerkesztés)

A műveletek elvégzéséhez szükséges egy migrációs és egy teszt környezet kialakítása az éles környezeten kívül. A *migrációs környezeten* valósul meg az első telepítő csomag próba telepítése, majd verzióváltások során a későbbi telepítőkészletek műszaki ellenőrzése. Az ellenőrzés során az éles szolgáltatás működéséhez szükséges támogató rendszerek működésének ellenőrzése is szükséges: azonosítási, hitelesítési képességek, külső kapcsolati rendszerek elérhetősége, stb. Sikeres műszaki ellenőrzést, majd teszt üzembe helyezés követően nyílik lehetőség a funkcionális tesztelésre, az erre a célra kialakított *teszt környezeten*. Sikeres funkcionális tesztelést követően van lehetőség az éles üzembe helyezés végrehajtására.

Amennyiben a műszaki ellenőrzés vagy a funkcionális tesztelés hiányosságot, hibát fed fel, akkor a megfelelő szakértői támogatással a fejlesztési vagy üzemeltetési feladatok jöhetnek létre a folyamat során, amelyekre az érintett működést lehetővé tévő rendszereknek szabad kapacitást kell biztosítaniuk. A hibajavítást követően a teljes átadási folyamat megismétlése szükséges. A Military Scrum az imént felvázolt módon valósítja meg az átvételi követelmények finomítását, az eljárás akkor ér véget,

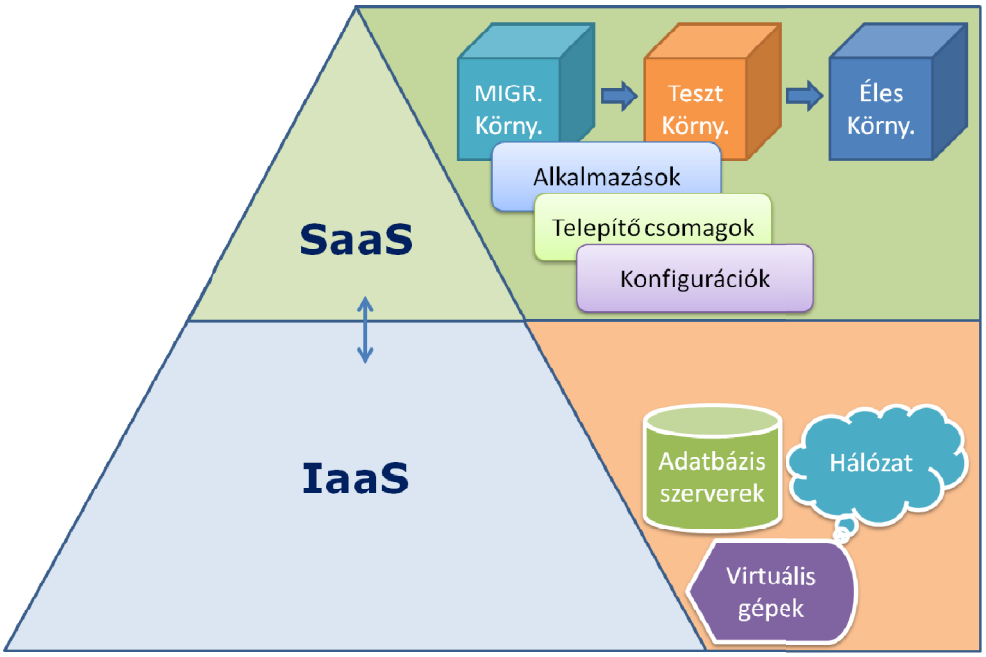
amikor a hibajelenségeket, működésbeli problémákat sikerült orvosolni, a fejlesztett szoftver ekkor éles üzembe telepíthető.

Az átadási folyamat a projekt különböző szereplőinek részvételét követeli meg, a megfelelő érettségi szintet elérő szakértői támogatás nélkül a folyamat nem hajtható végre, az elvárt képességek leírása a PMS-ben található.

	Próba-telepítés	Műszaki ellenőrzés	Teszt üzembe helyezés	Funkcionális tesztelés	Éles üzembe helyezés
DevOps	x	x			
Ops			x		x
Cust, (PO)				x	

21. táblázat A Military Scrum üzembe helyezési szakaszának feladatai és szereplői (saját szerkesztés)

A Military Scrum által előírt átadási folyamat segítségével a szoftver alapú szolgáltatási rétegben szabályozott folyamatok alakíthatók ki az első telepítés és a későbbi verzióváltás támogatásához. A folyamat során az alkalmazások a minőségi mutatók javítását szolgáló kijelölt útvonalon keresztül érik el végső rendeltetési helyüket. A telepítő csomagok és a konfigurációk környezetekhez rendelve külön-külön kezelhetők az éles rendszertől függetlenül, mozgásteret biztosítva a szoftverfejlesztési és üzemeltetési kérdések eldöntésére szolgáló tesztek elvégzéséhez. A különböző környezetek hozzáférési jogosultságai külön szabályozhatók, így az is garantálható, hogy az éles üzembe helyezést csak bizonyos speciális jogokkal rendelkező szakemberek végezhessék, míg az azt megelőző telepítések, tesztek az alacsonyabb jogosultsági szinttel rendelkező szakemberek számára is végrehajthatóvá válnak.



51. ábra A SaaS szolgáltatási réteg és a telepítési folyamatok kapcsolata (saját szerkesztés)

4.5.2 Biztonsági kockázatok kezelése

Az üzembe helyezés és az azt követő szakaszok számos biztonsági kockázatot rejtenek magukban, mert a fejlesztett szoftver alapú szolgáltatás ebben az időszakban válik széles körben elérhetővé. A szoftverfejlesztői oldalon túl, megjelenik az üzemeltetés és a felhasználók is a szereplők között, ezzel növelve a különböző biztonsági események kialakulásának kockázatát.

- Szoftverfejlesztési kockázatok - az átadási folyamat során, ha műszaki vagy funkcionális hiba jelentkezik, akkor a kijavításához a fejlesztőcsapat kapacitásának egy részére szükség lehet hozzá. Amennyiben az egymásra csúszó fejlesztési menetek technikája valósul meg, akkor az aktuális fejlesztési sprint rovására kell elvégezni a hibajavításokat. Kockázatkezelési lehetőségek:
    - Az első telepítési folyamat támogatásához célszerű külön erőforrást biztosítani a fejlesztőcsapat kapacitásának rovására vagy a fejlesztési szakasz átmeneti felfüggesztése segítheti az éles üzembe helyezésig, ha azt lehetővé teszi a projekt ütemezése.
    - Automatizált folyamatok segítségével a fejlesztői támogatás szükségességének csökkentése.
- Üzemeltetési kockázatok – a módszertan által elvárt telepítési folyamatok számos biztonsági kockázatot rejthetnek magukban, mert a felvázolt átadási folyamat számos manuális lépést rejt magában. A telepítő csomagok másolása, mozgatása, a konfigurációs állományok szerkesztése számtalan hibázási lehetőséget rejt magában. A rendszer felhasználása során mutatkozó technikai hibák későn történő detektálása, akár rendszerleálláshoz is vezethet a megfelelő, időben történő reagálás hiányában. Kockázatkezelési lehetőségek:
    - Automatizált telepítési folyamatok megvalósítása megfelelő jogosultsági rendszerhez igazítva.
    - Automatikus naplófájl alapú hibadetektálási technikák alkalmazása.
- Felhasználási kockázatok – az éles üzembe helyezett szoftver alapú szolgáltatás a támadások célpontjává válhat. A szolgáltatást célzó támadások detektálása meghaladja az üzemeltetés védelmi szintjét. Kockázatkezelési lehetőség:
    - A fejlesztett szoftver által felismert biztonsági incidensek elleni reagálási eljárások kialakítása, az események naplózása, azok eljuttatása az üzemeltetési környezet rendszerfelügyeleti szoftveréhez.

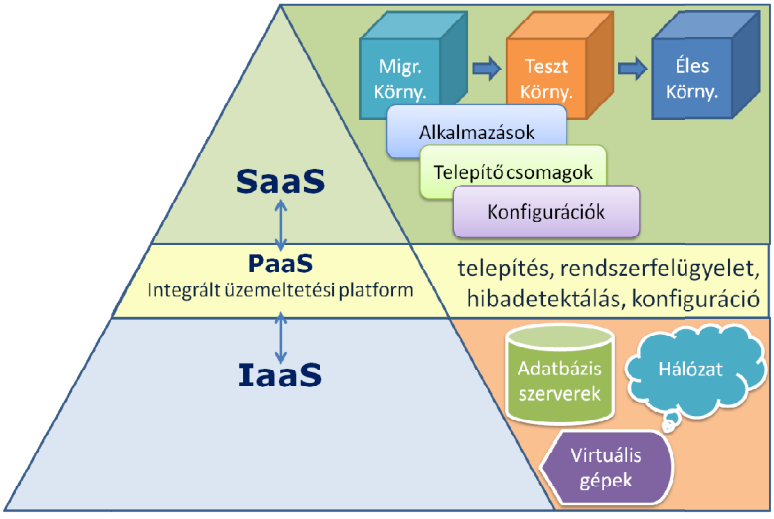
A feltárt kockázatok kezelésére részleges vagy teljes megoldást nyújthat az automatizálás. Az üzembe helyezés, a felhasználás és a támogatás szakaszaiban a kialakított automatizmusok segítségével gyorsabb és átláthatóbb folyamatok, illetve megbízhatóbb szoftver alapú szolgáltatások hozhatók létre. A Military Scrum az iménti gondolatmenet folytatásaként az üzemeltetés támogatási folyamatok számára a működést lehetővé tevő rendszerek számára egy integrált üzemeltetési platform meglétét írja elő. A módszertan a projekt érettségi szintjét a fejlesztést követő szakaszokban az említett üzemeltetési platform meglétéhez és képességeihez köti, a PMS tartalmazza a konkrét elvárásokat.

4.5.3 Integrált üzemeltetési platform

A Military Scrum olyan integrált üzemeltetési platform alkalmazását írja elő, amely segítségével a szoftver alapú szolgáltatási rétegben automatizáltan, integrált támogató rendszer segítségével valósíthatók meg az alábbi folyamatok.

1. Átadási folyamatok támogatása
    - Első telepítést támogató folyamatok
    - Verzióváltást támogató folyamatok
2. Üzemeltetés támogatása
    - Konfiguráció
    - Diagnosztikai funkciók
    - Hibakeresés

Az üzemeltetési platformnak modelleznie kell a fejlesztett szoftver által használt infrastruktúrát, környezeteket, telepítő csomagokat és konfigurációkat, valamint tárolnia kell a rendszerben végbement folyamatok hatásait is.



52. ábra A Military Scrum által előírt integrált üzemeltetési platform feladatai (saját szerkesztés)

Amennyiben az integrált üzemeltetési platform teljesíti a vele szemben támasztott követelményeket, akkor lehetségessé válik a manuális beavatkozások számának jelentős redukálása. A szoftver alapú szolgáltatásban végrehajtott változások minimális, ugyanakkor elégséges jogokkal rendelkező technikai felhasználók segítségével valósulnak meg. A követelményként meghatározott technológia alapot ad a folyamatos telepítés és a rendszerdiagnosztika integrált megvalósításához is.

Az integrált üzemeltetési platform segítségével automatikusan kitelepített, teszt környezeten végzett funkcionális tesztelés felel meg a *helyességi ellenőrzés* folyamatának, amikor a telepítő csomag, a környezete és a funkciók együttes ellenőrzés valósul meg a megrendelő által. A helyességi ellenőrzés egyben az *éles üzembe helyezésre alkalmasság ellenőrzését* is jelenti, ha műszaki ellenőrzés is társul hozzá.

Az integrált megoldás megléte nem csak az emelt szintű folyamat támogatást teszi lehetővé, a rendszerben lévő modell és a folyamatok alapján kinyerhetővé válnak a támogató dokumentációk előállításához szükséges adatok, a rendszer adatbázisa önmagában egyfajta jól strukturált dokumentumként fogható fel.

A megfelelő érettségi szinttel rendelkező üzemeltetés támogató rendszernek biztosítania kell a konfigurációkban bekövetkező változások követését, illetve átfogó képet kell nyújtania az adott szolgáltatás üzemeltetési környezetének és kapcsolódó rendszereinek állapotáról.

A fejlesztés szakaszában a telepítő csomagok számára előírt információk kinyerése az integrált üzemeltetési platform felületein lehetővé teszi a nyomon követés támogatását, mert a kitelepített szoftververziót meghatározó összes követelmény azonosítója kinyerhetővé válik a rendszerből. A változási jegyzékek hasonló elvű kinyerése az üzemeltetés támogató rendszerből jelentősen segítheti a támogatási feladatok ellátását. A platform további lehetőséget biztosíthat a felhasználás szakaszában végzett felhasználói konfiguráció módosítások és a támogatás szakaszában végzett üzemeltetési paraméterezések összefésült megjelenítésére.

Amennyiben a fejlesztett szoftverrel szemben követelmény a saját diagnosztikai adatok publikálása, akkor az integrált üzemeltetési platformmal szemben követelmény lehet azok figyelése, így rendszer szintű diagnosztikai képességeket kialakítva.

További követelmény az üzemeltetés támogató rendszerrel szemben a felügyelt rendszerekben keletkező hibák figyelése, illetve azok jelzése az üzemeltetést végző állomány felé, az eljárásba becsatlakoztathatók a biztonsági események naplófájl bejegyzései is, amelyek így a személyzeti szintű reagálást segítik elő.

4.5.4 Dokumentum sablonok a fejlesztési szakasz után

A felhasználási és támogatási szakaszokat az integrált üzemeltetési platform segítségével támogatja technológia oldalról. Módszertani oldalról különböző dokumentum sablonokat ír elő, amelyek kezelésére alkalmasnak kell lennie az igényeket és követelményeket tartalmazó előírt feladatkezelő rendszernek. Három különböző sémájú táblázat jelenik meg ezekben a szakaszokban a feladatok támogatásához. Az átvételi tesztek elvégzéséhez a Military Scrum a 22. táblázatban lévő oszlopstruktúrát és számképzést írja elő.

Teszt azon.	**Köv. azon.**	**Ellenőrzési feladat**	**Funkció**	**Bemenet**	**Elvárt kimenet**	**Értékelés**	**Hiba-azon.**
T-5.1-1	UK-23	Üzenetküldés alárendeltek számára	Üzenetküldés	Alárendelt felhasználók	Kézbesített üzenet	Sikeres	-
T-5.1-1	UK-24	Üzenetküldés felettes számára	Üzenetküldés	Felettes	Kézbesített üzenet	Sikertelen	H-5.1-1
.	.	.	.	.	.	.	.
T-5.1-n	.	.	.	.	.	.	.

22. táblázat Átvételi tesztek az átadási folyamat során (saját szerkesztés)

Ha hibák kerülnek azonosításra az átadás alatt álló funkciók tesztelése során, akkor a 23. táblázat struktúrájának megfelelő hibabejelentő dokumentációs sablon alkalmazása a követelmény. A Hiba azonosító oszlop második tagja a hibajelenséget produkáló szoftver verziójára utal. Amennyiben a hibát az átvételi teszt folyamán azonosították a Követelmény azonosító és Teszt azonosító oszlop kitöltése is lehetségessé válik a nyomkövetési infrastruktúra támogatásához.

Hiba-azon.	**Köv. azon.**	**Hibajelenség**	**Funkció**	**Bemenet**	**Elvárt kimenet**	**Teszt azon.**
H-5.0-1	-	Helytelenül megjelenő speciális karakterek a kapott üzenetekben	Üzenetküldés	Ékezetes betűk	Helyesen megjelenő kézbesített üzenet	-
H-5.1-1	UK-23	Üzenetküldés felettes számára funkció hibásan működik. A kiválasztható címzettek listája nem teljes.	Üzenet címzettjeinek összeállítása	Felettes felhasználók	Minden felettes címzett kiválasztható	T-5.1-1
.	.	.	.	.	.	.
H-5.x-n	.	.	.	.	.	.

23. táblázat Military Scrum hibajegyzék (saját szerkesztés)

A változtatási igények rögzítése a koncepcionális előtervezési szakasz során megismert táblázattal történik – az igények sorszám képzése egy V karakterrel bővül.

Ig.Ssz.	**Követelmény támasztója**	**Érintett szervezeti elemek száma**	**Alk. jellege**	**Felhasználók száma**	**Igény bemutatása**	**Eszközök**
VI-1	Logisztika (Harcászati szint)	5	tábori rendszer	80	Üzenetküldés funkció bővítése előre meghatározott üzenet típusokkal	Speciális hardver
VI-2	MHP IICSF (Hadműveleti szint)	2	stacioner rendszer	25	Statisztikai adatok kinyer	Android, PC
.	.	.	.	.	.	.
VI-n	.	.	.	.	.	.

24. táblázat Military Scrum változtatási igények (saját szerkesztés)

4.6 ÖSSZEGZÉS

A 4. fejezetben megismerhettük a jelenlegi szoftvertechnológia adta lehetőségeket, áttekintettük az agilis szoftverfejlesztés és a Scrum szoftverfejlesztési módszertan alapvetéseit. Ezt követően a Scrum elhelyezésre került a projekt menedzsment háromszögben, majd láthattuk, hogy a módszertan által elfoglalt hely megfelelő a kormányzati és ezen belül a honvédelmi ágazaton belüli alkalmazásra is.

Ezután, a Military Scrum szoftverfejlesztési módszertan szisztematikus meghatározása valósult meg a NATO SLCM folyamatok és a Scrum módszertan figyelembe vételével, feltételezve, hogy a kialakított módszertan elsődleges felhasználási területe a szoftver alapú szolgáltatások fejlesztése és támogatása, a munkához hozzájárultak a 2. fejezetben feltárt követelményelemzési (R) elvárások is. A fejezet során a Military Scrum által használt életciklus szakaszok, szerepkörök, feladatok és dokumentációs sablonok definiálása is megtörtént – támogatandó az alábbi tevékenységeket:

1) Igények felmérése – koncepcionális előtervezés;
2) követelményhalmazok kialakítása – koncepcionális előtervezés;
3) követelmények meghatározása az alábbi területeken – koncepcióalkotás;
    i) új rendszerek fejlesztése,
    ii) szoftver cseréje,
    iii) adatmigrálás,
    iv) rendszerek közötti integráció;
4) átadás-átvételi tesztelési forgatókönyvek elkészítése – fejlesztés;
5) hibajegyek rögzítése – támogatás;
6) változtatási igények gyűjtése – felhasználás.

A fejlesztési szakasz támogatásához kijelölésre kerültek azok az agilis szoftverfejlesztési technikák, amelyek lehetővé teszik a szoftver alapú szolgáltatások előállításához szükséges telepítőkészletek megfelelőségének garantálását módszertani és technológiai oldalról is egyaránt: TDD, DDD, CI, stb.. A fejlesztést követő szakaszok támogatásához meg lettek határozva az üzemeltetési környezettel szemben támasztott folyamat szintű követelmények, valamint az integrált üzemeltetési platform (IÜP) fogalma is, amelyek együtt lehetővé teszik a megfelelő szintű üzemeltetési mutatók elérését. A módszerek meghatározása során szem előtt volt tartva a biztonsági követelményeknek és a nyomon követési infrastruktúrának való megfelelés is.

5. BIZTONSÁGOS ÉS FENNTARTHATÓ ADATKEZELÉS A KATONAI SZERVEZETEK ÉLETÉBEN

Az előző fejezet tanulságai alapján elmondható, hogy a katonai célú szoftverfejlesztések számára megfelelő módszertani hátteret és ideális technológiai előfeltételeket biztosít a Military Scrum helyes alkalmazása. Az 5. fejezet során feltételezem, hogy a módszertani és a technológiai háttér megléte már adott új tervezési minták meghatározásához és felhasználásához is.

Az eddig még nem tárgyalt kutatási célokhoz igazodva ebben a fejezetben – egyfajta esettanulmányként – a Magyar Honvédség ügyviteli folyamataira vonatkozó hazai jogszabályokból és az általános Uniós adatkezelési szabályozásból kerül levezetésre egy elemzésre alkalmas feladatrendszer. Az elemzés a katonai szervezetekre jellemző ügyviteli folyamatok vizsgálatán alapul az adat-konzisztencia és az információbiztonság szemszögéből.

A létfontosságú infokommunikációs rendszerek védelmére vonatkozó Uniós szabályozás egy átfogó keretet határoz meg a kibertérben jelenlévő különböző fenyegetések kapcsán, azonban a tagállami szintű intézkedések eltérhetnek egymástól a keretrendszer adta lehetőségeken belül.

Magyarországon az információbiztonság kérdéskörével a 2011. évi CXII. törvény foglalkozik az információs önrendelkezési joggal és az információszabadsággal [111] (röviden: Infotv.). Jelenlegi állapotát 2018.07.26-ai módosítását követően nyerte el összhangban az Európai Parlament és Tanács (EU) 2016/679 rendeletével, a GDPR-al [112], amely a személyes adatok védelméről rendelkezik az Európai Unión belül. A szabályozás célja egy országhatárokon átívelő, átlátható és nyomon követhető adatkezelési- és adatfeldolgozási rendszer kialakítása.

Hazai viszonylatban 2018. január 1-én lépett életbe az elektronikus ügyintézés és a bizalmi szolgáltatások általános szabályairól szóló 2015. évi CCXXII. törvény [113], amely lefekteti az elektronikus ügyintézés alapjait és meghatározza azokat a szolgáltatásokat és kötelezettségeket, amelyek a korszerűbb közszolgáltatások nyújtásához szükségesek. Mindkét dokumentumban gyakran jelennek meg az adatok megőrzésével, törlésével kapcsolatos elvárások. A fejezet feltárja az iménti jogszabályokból levezethető informatikai követelményeket, majd az azonosított követelményrendszerre szoftvertervezési mintákon keresztül ad megoldást.

5.1 A GDPR ÁTTEKINTÉSE

A magyar köztudatban is GDPR-ként (angolul: General Data Protection Regulation) jelent meg az Európai Unió Általános Adatvédelmi Rendelete. A dokumentum 2018. május 25-én lépett hatályba, ezzel sok fejfását okozva azoknak az állami és nem állami szervezeteknek, amelyek a hatálya alá tartoznak. Azonnal felmerülhet a kérdés a szakterületet kevésbé ismerőknek, hogy valóban szükség volt-e egy ekkora horderejű rendeletre? Cinikusabban is fel lehet tenni a kérdést: a brüsszeli bürokrácia újra alkotott valamit? A szabályozás létjogosultságát azonban nehéz lenne tagadni, mert ugyan tagállami és Uniós szinten is léteztek már a területet szabályozó dokumentumok, de az EU kiberbiztonságának javításához egy új, átfogó rendeletre volt szükség. Miért leszek attól nagyobb „kiberbiztonságban", ha minden velem kapcsolatban álló jogi személy számára hozzájáruló nyilatkozatot kell kitöltenem a személyes adataim kezelésével kapcsolatban? Erre a kérdésre a válasz maga a GDPR. A rendelet részletes feltárása előtt érdemes szót ejteni a napjainkban egyre nagyobb teret nyerő kiberbűnözésről. Megjelentek a zsaroló és adatlopó vírusok, amelyek alapvetően otthoni notebookokról lophatnak adatokat, ugyanakkor egy feltört vírusos számítógép lehetővé teheti egy teljes informatikai rendszer feltörését is. Célzott támadások mögött lehetnek hacker csoportok, titkosszolgálatok, de az ipari kémkedés is előfordulhat, mint kiváltó tényező. Ha emellett azt is szem előtt tartjuk, hogy az Európai Unió belső piacán működő infokommunikációs rendszerek országhatárokon átnyúló fénysebességű sztrádán terjesztik személyes adatainkat egy-egy online vásárlás, regisztráció, valamilyen szolgáltatás igénybe vétele során, azonnal érthetővé válik, hogy a kiberbiztonsági kockázat jelentékeny biztonsági tényezővé vált napjainkra. Az Általános Adatvédelmi Rendelet célkitűzése, hogy személyes adataink szabályozott módon kerüljenek be a velünk kapcsolatban álló szervezetek informatikai nyilvántartásaiba, adataink továbbadása, feldolgozása követhető módon történjen. A rendeletben külön hangsúlyt kap a személyes adatok törlésének kérdése, csökkentve a biztonsági kockázatot – ha nincs adat, akkor nincs mit ellopni.

Az általános rendelkezésekből [112, 1. fejezet] megtudhatjuk, hogy a rendelet célja alapvetően a természetes személyek adatainak védelme és a személyes adatok szabad áramlásának szabályozása az EU területén belül[78]. Az általános szabályozás alól kivételt képezhetnek az egyes tagállamok védelmi szektoraiban tevékenykedő külön-

[78] A GDPR 1. fejezet 3. cikk foglalkozik a területi hatállyal

böző szervezetek. A precíz megértéshez szükséges a GDPR fogalom-meghatározásaiból [112, 1.4] néhányat felsorolni az alábbiakban.

1. *„„„személyes adat”: azonosított vagy azonosítható természetes személyre („érintett”) vonatkozó bármely információ;*
2. *„adatkezelés”: a személyes adatokon vagy adatállományokon automatizált vagy nem automatizált módon végzett bármely művelet vagy műveletek összessége, így a gyűjtés, rögzítés, rendszerezés, tagolás, tárolás, átalakítás vagy megváltoztatás, lekérdezés, betekintés, felhasználás, közlés, továbbítás, terjesztés vagy egyéb módon történő hozzáférhetővé tétel útján, összehangolás vagy összekapcsolás, korlátozás, törlés, illetve megsemmisítés;*
3. *„adatkezelő”: az a természetes vagy jogi személy, közhatalmi szerv, ügynökség vagy bármely egyéb szerv, amely a személyes adatok kezelésének céljait és eszközeit önállóan vagy másokkal együtt meghatározza;*
4. *„adatfeldolgozó”: az a természetes vagy jogi személy, közhatalmi szerv, ügynökség vagy bármely egyéb szerv, amely az adatkezelő nevében személyes adatokat kezel;”*

A személyes adat definíciójában találkozhatunk az érintett fogalmával, aki az a természetes személy, akinek a személyes adatait az adott nyilvántartás tárolja. Az érintett azonosított, ha direkt módon kinyerhető a tárolt adatokból az érintett kiléte. Azonosítható az érintett, ha egy vagy több adat alapján közvetlen vagy közvetett módon feltárható a természetes személy kiléte.

Az adatkezelés végzése során – amely a személyes adatokon végzett „tetszőleges” műveletet takar – az adatkezelőknek igazolniuk kell a tevékenység jogszerűségét, amelyre felhatalmazást adhat az érintett saját maga vagy valamilyen szerződés, illetve egy tagállami vagy uniós jogszabály is megkívánhatja valamilyen adatkezelési tevékenység végzését. A felsoroltakon túl további kivételes helyzetekben is jogszerű lehet a személyes adatokon végzett adatkezelés, ezeknek a jogosítványoknak harmóniában kell lenniük az uniós joggal – részletesen a második fejezet, hatodik cikk foglalkozik a témakörrel. [112, 2.6.(2)-(4)] Az adatfeldolgozók a GDPR vonatkozásában az adatkezelőktől örökölik meg a felhatalmazást a személyes adatokon végzett tevékenységükre. Mind az adatkezelőknek és az adatfeldolgozóknak is tevékenységüket jogszerű, tisztességes és átlátható eljárás keretein belül kell végezniük. Továbbá, a célhoz kötöttség, az adattakarékosság, a pontosság, a korlátozott tárolhatóság, az

integritás és a bizalmas jelleg sorolhatók azon alapelvek közé, amelyekért az adatkezelő felelősséggel tartozik. [112, 2.5.(1)-(2)].

Attól függetlenül, hogy egy jogszabály, szerződés vagy az érintett hozzájárulása teremt jogalapot az adatkezelésre, az érintett mindenkor jogosult arra, hogy információt kapjon a személyes adataival kapcsolatos adatkezelési folyamatokról, az adatkezelés céljáról és a kezelt személyes adatok kategóriáiról. [112, 2.15] Ha bekövetkezik ez az esemény, akkor a kézi fizikai- vagy digitális nyilvántartást vezető adatkezelőknek is gondot okozhat a fenti információk egy időben történő, pontos összegyűjtése és az adatszolgáltatás. Ennél sokkal összetettebb problémát jelenthet az automatizált módon történő adatkezeléssel kapcsolatos információk kinyerése.

A továbbiakban a digitális alapokon működő automatizált adatkezelés és adatfeldolgozás kérdéskörét vizsgáljuk. A vizsgálat során megpróbáljuk tetten érni az informatikában jól ismert CRUD műveleteket[79], melyek az adatok olvasását, létrehozását, módosítását, valamint törlését fogják össze. Az iménti példában egy létrehozási (CREATE) és olvasási (READ) műveletre vonatkozó követelményre derült fény – az adatkezelési tevékenység feltételezi a személyes adatok mentését, míg a hozzáférési jog azok olvasását. Az Általános Adatvédelmi Rendelet konkrét követelményeket fogalmaz meg a személyes adatok helyesbítését és törlését illetően is [112, 2.16-17], ezáltal a személyes adatokra vonatkozó módosítás (UPDATE) és törlés (DELETE) műveletek is megjelennek a követelményrendszerben. A GDPR és az Infotv. által meghatározott CRUD műveletek problémakörének tárgyalásához nem szükséges az említett dokumentumok további elemzése, mindkét dokumentum olyan speciális jogi eseteket tárgyal, ahol valamelyik CRUD művelet végrehajtása részben, illetve egyáltalán nem végezhető el. A kivételt képező esetek tárgyalása túlmutat az alapvető szoftvertechnológiai problémán, miszerint: Milyen adatmodell képes helyesen és jogszerűen kezelni a működés során felmerülő olvasási, létrehozási, módosítási és törlési műveleteket az érintettek személyes adatainak vonatkozásában?

Az 53. ábrán az érintettek személyes adatai és az adatkezelési tevékenységek során keletkező egyéb információk kapcsolatát láthatjuk. A bemutatott modell feltételezi, hogy egy érintettnek akár több jogalappal is lehetnek különböző adatai egyazon informatika rendszerben, nyilvántartásban. Bizonyos tekintetben az is elképzelhető, hogy az adatkezelési tevékenységek során több *Érintett* adatai is kapcsolódhatnak

[79] CRUD műveletek – mozaikszó a **C**REATE **R**EAD **U**PDATE **D**ELETE angol szavak kezdőbetűiből, magyarul: létrehozás, olvasás, módosítás, törlés

egyazon adatkezelési információhoz. Példa lehet erre egy online rendelés, ahol címzett és a vevő két különböző természetes személy.

53. ábra Az érintett személy adatai és az adatkezelés során keletkező információk közötti kapcsolat primitív modellje. SOK-SOK kapcsolat (saját szerkesztés)

Vizsgáljuk meg ezt a modellt a CRUD műveletek szemszögéből.

1. CREATE – a létrehozás során az érintett minden személyes adata egy helyen kerül tárolásra, ha egy rendszerben több adatkezelési tevékenységet is folytatnak, akkor sérülhet az adattakarékosság elve, mert több adat is bekerülhet a rendszerbe, mint ami feltétlenül szükséges.
2. READ – adatszolgáltatási kötelezettségből vagy az adatkezelési tevékenységből fakadó folyamatok számára szükséges olvasási művelet. Ebben a modellben az érintettnek mindig egy aktív személyes adata van a nyilvántartásban. Ha több adatkezelési eljárás is történt az érintett személyét vonatkozóan, nehézkes egy korábbi cím, tetszőleges adat kinyerése a rendszerből. Eseménynaplók segítségével kezelhető a probléma, de az adatok kinyerése nehézkessé válhat.
3. UPDATE – helyesbítési igény vagy egy adatkezelési tevékenység során végrehajtandó módosítás abban az esetben, ha a nyilvántartásban több aktív adatkezelési tevékenység is zajlik egyazon érintett vonatkozásában inkonzisztens állapotot válthat ki. Teljes egészében vagy részben sérülhetnek az érintett személyes adatai valamelyik folyamatban lévő adatkezelési tevékenységben, ezáltal hibás folyamatok indulhatnak.
4. DELETE – a törlés művelete a GDPR vonatkozásában került igazán előtérbe, addig nem szívesen foglalkoztak a szoftvermérnökök az adatokon végzendő törlés művelet problémájával. Azok a rendszerek, amelyek a GDPR hatálya alá tartoznak már kötelesek a korlátozott tárolhatóság elvét figyelembe venni, így bizonyos időközönként, illetve az érintett kérésére fizikailag törölniük[80] kell az érintett személyes adatait egy-egy adatkezelési tevékenység vonatkozásában. Egy ilyen művelet sértheti az informatikai rendszer integritását, például abban az esetben, ha több párhuzamos adatkezelési tevékenység végrehajtása zajlik egy időben.

[80] Fizikai törlés – egy adatrekord végleges, visszavonhatatlan eltávolítása az adatbázisból

5.2 AZ ÉRINTETTHOLDER TERVEZÉSI MINTA

Évtizedeken át a szükségtelenné vált adatok logikai törlése[81] volt a bevett gyakorlat az informatikai rendszerek tervezése során. Ennek oka főleg az volt, hogy egy programhiba esetén az adatok visszaállítása, visszaépítése sokkal egyszerűbb feladat volt, mint egy fizikai törlés esetén. Ugyanakkor itt azt is meg kell jegyezni, hogy ez a technika vezetett ahhoz is, hogy az ily módon tervezett rendszerekben az adattisztítás szinte lehetetlenné vált az évek múlásával – ezáltal biztonsági kockázatot teremtve a külső és belső kockázatok területén is.

1. Fölöslegesen tárolt adatok – adatlopás;
2. Törölt, inaktív adatok – nehézkes továbbfejlesztés, körülményes adattisztítás;

A problémát tovább tetézi az a tény is, hogy komolyabb jogfolytonosság tanúsítására képes rendszerek esetében egy tetszőleges entitás adatai többször változhatnak az évek során és csak évek múltával kerülhet törlésre az entitáshoz tartozó összes adat.

Az 53. ábrán bemutatott primitív modell önmagában kevésnek bizonyul a fizikai törlés támogatására. Lehetséges választás lehet az 54. ábrán szereplő Proxy tervezési minta alkalmazása, ahol a kliens objektumhoz tartozhat több proxy objektum, melyeken keresztül a kliens üzenetet válthat az ún. proxizott objektumokkal. Az ábrából kiderül, hogy a Proxy mindig a proxizott objektum felé továbbítja az üzeneteket. Valamint egy proxizott objektumhoz egy időben tartozhat több Proxy is. Vizsgáljuk meg ezt az alapvető tervezési mintát a proxizott objektum fizikai törlésének szempontjából. Amennyiben bekövetkezik a DELETE művelet a Proxy nem tudja tovább továbbítani kliens üzeneteit a proxizott objektum felé, mert az már nem létezik – így jó esetben hibaüzenetet, rossz esetben hibát produkál a modell.

54. ábra Proxy tervezési minta (saját szerkesztés)

Létezik a Proxy tervezési mintához nagyon hasonló, ugyanakkor más elven működő tervezési minta a Holder (magyarul: tartó). A Holder felépítését és működési elvét az 55. ábra mutatja be, miszerint különböző kliensekhez, különböző Holder-ek tartozhatnak. A Holder ismeri a megtartott objektumot és a kliens kérésére átadja azt a kliens számára. A modell lényege, hogy nem kell a klienshez folyamatosan betölteni az összes megtartott objektumot, elegendő megtenni azt akkor, amikor a kliens konkrét üzenetküldést kezdeményezne. Utóbbi viselkedés a Proxy esetében is hasonló. A

[81] Logikai törlés – egy adatrekord törölt állapotának jelzése egy igaz/hamis kapcsoló segítségével.

megtartott objektum fizikai törlésekor a Holder egy NULL objektumot ad vissza, amelyet a kliens már kezelhet, természetesen, ha fel van rá készítve.

55. ábra Holder tervezési minta (saját szerkesztés)

A bemutatott Proxy és Holder tervezési minták önmagukban nem képesek kezelni a fizikai törlés problémáját. További finomításra van szükséges a zavartalan adatkezelési tevékenység támogatásához.

Fontos tisztázni, hogy az érintett személyes adatait a megtartott objektum, illetve a proxizott objektum jelentik a fenti modellekben. A Proxy tervezési minta legnagyobb problémája a folyamatban lévő adatkezelési tevékenység szempontjából, hogy a modell állapotmentes, tehát csak továbbítja az üzeneteket. Ha a proxizott objektum törlődik, akkor az esetek jelentős részében az adatkezelési folyamat sérülhet adatvesztés következtében. A Holder tervezési minta esetében minimális állapot már megjelenik, ha a kliens elkéri a Holder-től a megtartott objektumot, akkor annak létezése eldönthető. A GDPR-ban megjelenő követelmények az érintett irányából vizsgálják a problémát, így kézenfekvő a modell irányának megfordítása és a probléma vizsgálata az Érintett irányából, valamint a Holder tervezési minta ötvözése a Proxy tervezési mintával az alábbiak szerint. Az így kialakult tervezési mintát az 55. ábra mutatja be.

56. ábra ÉrintettHolder tervezési minta (saját szerkesztés)

1. CREATE – minden adatkezelési folyamat minden használati esetében az Érintett és az Adatkezelési tevékenységhez köthető információ között létre kell hozni egy ÉrintettHolder típusú objektumot, amely a következő tulajdonságokkal rendelkezik.
2. READ – amíg aktív az adott adatkezelési tevékenység, addig proxyként továbbítja az üzeneteket az Érintett irányába.
3. UPDATE – amennyiben a teljes adatkezelési folyamat vagy egy mozzanata véget ér akkor az ÉrintettHolderben tárolásra kerülnek a konkrét adatkezelési tevékenységhez kapcsolható személyes adatok, amennyiben megváltoznak az Érintett személyes adatai, az ÉrintettHolder az egykor aktív történetileg helyes adatokat fogja tartalmazni.

4. DELETE – ha valamilyen okból törölni kell az Érintettet a teljes informatikai nyilvántartásból vagy bizonyos adatkezelési tevékenységekhez kapcsolódóan, akkor az Érintett fizikai törlése megoldott, ugyanis az ÉrintettHolder tovább tárolhatja egy adott folyamat számára releváns adatokat, ha van valamilyen követelmény erre vonatkozóan. Ha valamilyen okból kifolyólag minden információt törölni kell az Érintettről bele értve minden ÉrintettHolder-t is, akkor egy ÜRES Érintett elem bevezetésével és a kapcsolódó ÉrintettHolder-ek ürítésével és az ÜRES elemre történő átirányítással ez a folyamat is kezelhető. Természetesen ez a folyamat csak megfelelő körülmények mellett indítható el – az informatikai rendszert fel kell készíteni az ÜRES elem létezésére, illetve az Érintetthez kapcsolódó folyamatban lévő adatkezelési folyamatok megszűnését eredményezi nagy valószínűséggel egy ilyen művelet.

A bemutatott *ÉrintettHolder* tervezési minta a természetes személyek adataihoz kapcsolódó CRUD műveletek kezelésére egy átfogó megoldást kínál, de nem ad választ az elektronikus ügyintézéssel kapcsolatos követelményekre. A 2015. évi CCXXII. törvény által felölelt két fő terület közül az elektronikus ügyintézés probléma köre az, amely a GDPR kapcsán tárgyalt problémákkal is találkozik, még akkor is, ha a GDPR korlátozási lehetőséget biztosít a tagállamok számára, hogy bizonyos területeken [112, 3.23] az érintettek személyes adatait véglegesen, illetve hosszabb távon, nagyobb adattartalommal tárolják. A kivételt képező jogszabályokban törekedni kell arra, hogy a természetes személyek alapvető jogai ne sérüljenek, és a szabályozás megfeleljen a demokratikus normáknak.

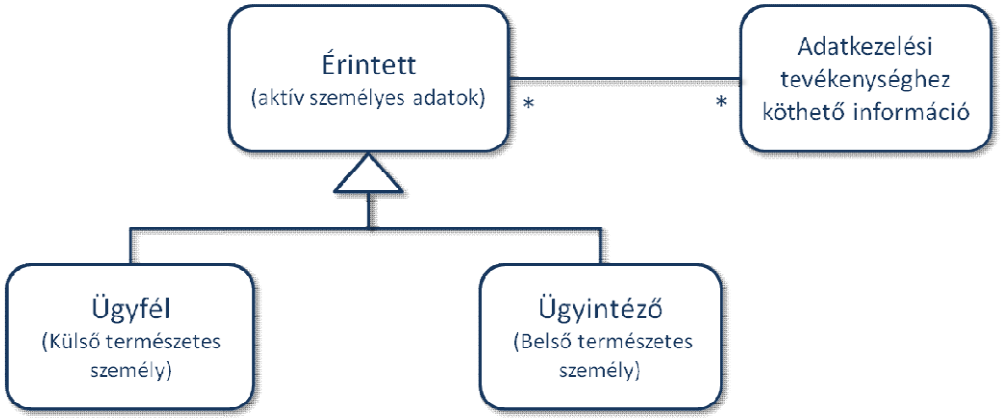
Fontos kiemelni, hogy attól függetlenül, hogy a felhatalmazás adott a kivételek képzésére jogszabályi szinten, a személyes adatokhoz tartozó karbantartási műveletek problémája ezekben a rendszerekben is jelen van, legfeljebb más hatások váltják ki azokat. Egyszerű példával élve: egy online vásárlás ügymenete nagyban hasonlítható egy hatósági igazolás igénylési folyamatához – kérelemre-igazolás; megrendelésre-áruszállítás – egy menet közben megváltozott postázási cím kezelése mindkét esetben elvárható követelmény lehet. A webshop esetében a vásárló kérheti a személyes adatainak azonnali törlését a szállítást követően, míg egy állami szerv felmentéssel rendelkezhet az adatok törlése alól. Érdekes kérdés azonban, hogy meddig őrizheti meg az adott állami szerv az ügyfél adatait. A személyes adatok vonatkozásában értelmezhető egyfajta elévülés, előbb-utóbb a törlés művelet igénye is megjelenhet az informatikai rendszerben. Itt kérdés lehet az, hogy az adott rendszer mennyire van

felkészítve az ilyen jellegű műveletekre. Az alábbiakban az elektronikus ügyintézés és a bizalmi szolgáltatások általános szabályairól szóló törvény elemzése következik a természetes személyek adatain végzett CRUD műveletek szemszögéből.

1. CREATE – az elektronikus ügyintézést biztosító szerv és az ügyfél elektronikus kapcsolatára vonatkozó szabályozás értelmében „Magyarországon az ügyfelet megilleti a jog, hogy az elektronikus ügyintézést biztosító szerv előtti ügyét – az e törvényben meghatározott módon – elektronikusan intézze". [113, 1.3.(1)]
2. READ – a közérdekből nyilvános adatok megismerhetőségének biztosítása, és a személyes adatok védelme megjelenik követelményként a 3. fejezetben [113, 1.7].
3. UPDATE – még ebben a fejezetben találkozhatunk a következő követelménnyel, mely szerint „az elektronikus ügyintézést biztosító szerv az elektronikus ügyintézést támogató és a jogszabályban előírt feltételeket biztosító rendszerfolyamatokat az ügyfél érdekeinek figyelembevételével alakítja ki." [113, 1.6.(2)]. Az ügyfél kényelmét szolgáló funkciókon túl, az „ügyfél igényelheti, hogy az általa meghatározott adatainak változásáról a"…„meghatározott elektronikus ügyintézést biztosító szervet" [113, 4.22.(2).a] automatikusan vagy esetileg elektronikus úton értesítse az illetékes szolgáltató.
4. DELETE – a szabályozott elektronikus ügyintézési szolgáltatások kapcsán megjelenik a törlés műveletre vonatkozó követelmény. Az ügyfélregisztrációs nyilvántartást kezelő szervnek meg kell tudnia szüntetni egy regisztrációt, például abban az esetben, ha a felhasználó kéri azt vagy más hivatalos értesülés alapján. [113, 4.32.(5)] Ez jelenthet fizikai, illetve logikai törlést is az elektronikus ügyintézést biztosító szervek informatikai rendszerében, amikor értesülnek a regisztráció megszűnéséről.

Az összegyűjtött követelmények hatásainak az elektronikus ügyintézését biztosító szervek informatikai rendszereiben is meg kell jelenniük. Ha az elektronikus ügyintézés folyamatát vizsgáljuk, akkor az érintett személy az esetek túlnyomó részében ügyfélként jelenik meg, ugyanakkor nem lehet elmenni amellett sem, hogy az ügy intézéséért felelős személy is lehet érintett, amikor a személyes adatai megjelennek az ügyintézési folyamatban. Utóbbi esetben is megjelennek a CRUD műveletek, amikor a természetes személy felvételre kerül az elektronikus ügyintézést biztosító

szerv (CREATE) informatikai rendszerében. Olvasási művelet (READ) kerül végrehajtásra, amikor az ügyintéző személyes adatai megjelennek az ügyintézés egyes mozzanataihoz köthetően. Ezzel párhuzamosan az ügyintéző személyes adatai módosulhatnak (UPDATE) – akár a név is, például házasságkötés miatt. Illetve az ügyintéző munkaviszonya megszűnhet, ekkor bekövetkezhet a törlés művelete (DELETE) is. Az ügyintézésre úgy is tekintünk, mint egy speciális adatkezelési tevékenységre, ebben az esetben az ügyfél és az ügyintéző is lehet érintett a személyes adatok változásának tekintetében, ezt a gondolatmenetet mutatja be az 57. ábra.



57. ábra Az érintett fogalmának kiterjesztése (saját szerkesztés)

Az 53. ábra tárgyalását követően már láthattuk, hogy a CRUD műveletek egy ilyen primitív modellben problémákba ütközhetnek. A továbbiakban az ÉrintettHolder tervezési minta kiterjesztését mutatjuk be a külső kapcsolattartás és folyamatos ügymenet támogatása érdekében.

5.2.1 Zavartalan külső kapcsolattartás

Az elektronikus ügyintézés és a bizalmi szolgáltatások általános szabályairól szóló 2015. évi CCXXII. törvényből kinyert követelmények alapján látható, hogy az elektronikus ügyintézést biztosító szerv informatikai rendszerének képesnek kell lennie kezelni az ügyfél személyes adataival kapcsolatos CRUD műveleteket. Az 58. ábrán szereplő ÜgyfélHolder tervezési minta felépítését tekintve analóg a már korábbiakban bemutatott ÉrintettHolder-rel.

58. ábra ÜgyfélHolder tervezési minta (saját szerkesztés)

Az alábbiakban az ÜgyfélHolder-en értelmezett elektronikus ügyintézés kapcsán felmerülő CRUD műveletek működése kerül bemutatásra. Az ÜgyfélHolder műkö-

dése az általános adatkezelési folyamatokat kezelő ÉrintettHolder tervezési minta viselkedésének ügyintézési folyamatokra történő leképezéséből vezethető le.

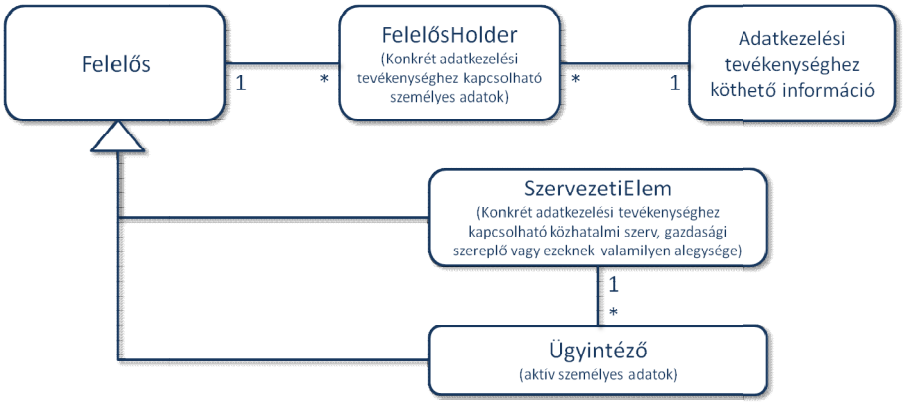
1. CREATE – minden Ügyfél által indított ügyintézési folyamat során létre kell hozni egy ÜgyfélHolder-t is az elektronikus ügyintézést biztosító szerv informatikai rendszerében a párhuzamosan folyó ügyintézési folyamatok végett.
2. READ – Egy folyamatban lévő, illetve lezárt ügy vonatkozásában az Ügyfél aktív adatainak kinyerése történhet az Ügyfélből, illetve egy lezárt ügy esetén az ÜgyfélHolder-ből.
3. UPDATE – Az egyes ügyintézési folyamatok sajátossága lehet, hogy az Ügyfél aktív adatainak módosítása már nincs hatással minden ÜgyfélHolder-re. Az érintett természetes személy – jelen esetben ügyfél – levelezési címének változáskor az ügyfél eredeti postai címét reprezentáló ÜgyfélHolder-t már nem feltétlenül kell megváltoztatni.
4. DELETE – Az ÉrintettHolder tervezési mintával analóg módon a teljes fizikai törlés is támogatható, de ügytípusonként is eljárhat az elektronikus ügyintézést biztosító szerv.

Az ÜgyfélHolder tervezési minta alkalmazása lehetővé teszi a zavartalan külső kapcsolattartást és az ügyfelek személyes adatainak magas szintű kezelését a GDPR alapelveivel összhangban.

5.2.2 Folyamatos ügyintézés fenntartása

Egy elektronikusan keletkezett és intézett ügy életútját összetett folyamatok adják. Az elektronikus ügyintézést biztosító szervnek képesnek kell lennie befogadni az ügyindító dokumentumot, ezt követően szabályozott módon kell párbeszédet folytatni az ügyféllel. Az ügyféllel történő kapcsolattartáson túl nagyon lényegesek ügyintézést támogató belső folyamatok is.

Az ügyintézési tevékenység azon túl, hogy legalább egy ügyintézőhöz – természetes személyhez – köthető még általában kapcsolódik egy szervezeti elemhez is. A kapcsolódó szervezeti elem feladatai közé tartoznak a helyettesítés és az ügyintéző változás problémáinak kezelése. Ebből következik az a felismerés, hogy az ügyintézési tevékenység *SzervezetiElem*-hez és *Ügyintéző*-höz is egyaránt köthető és kötendő.



59. ábra FelelősHolder tervezési minta (saját szerkesztés)

Az 59. ábrán az is látható, hogy a SzervezetiElem és az Ügyintéző között EGY-SOK kapcsolat áll fenn, miszerint egy szervezeti elemhez több ügyintéző is tartozhat. A GDPR terminológiájában az adatkezelési tevékenység ezen a szinten valósul meg. A következő felsorolás a FelelősHolder viselkedését mutatja be.

1. CREATE – Az ügyintézési folyamat indulásakor létre kell hozni egy-egy FelelősHolder-t az ügyintézésért felelős Ügyintéző és SzervezetiElem modellezéséhez.
2. READ – Egy folyamatban lévő, illetve lezárt ügy vonatkozásában a Felelős aktív adatainak kinyerése történhet a SzervezetiElem-ből vagy az Ügyintéző-ből, illetve egy lezárt ügy esetén a FelelősHolder-ből.
3. UPDATE – Az egyes ügyintézési folyamatok sajátossága lehet, hogy a Felelős aktív adatainak módosítása már nincs hatással minden FelelősHolder-re. Az ügyintézésért felelős SzervezetiElem vagy Ügyintéző változásának hatására a FelelősHolder-t már nem feltétlenül kell megváltoztatni.
4. DELETE – Az ÉrintettHolder tervezési mintával analóg módon a teljes fizikai törlés is támogatható, de ügytípusonként is eljárhat az elektronikus ügyintézést biztosító szerv a SzervezetiElem-ek és az Ügyintéző-k vonatkozásában is.

Az FelelősHolder tervezési minta alkalmazása lehetővé teszi az ügyintézési folyamat magas szintű támogatását a GDPR alapelveivel összhangban, különös tekintettel az ügyintézést végző természetes személyek adataira.

Magyarországon az előző évtizedben jelent meg az első elektronikus iratkezelésre vonatkozó kormányrendelet a 24/2006. (IV. 29.) BM-IHM-NKÖM [114] együttes rendelet a közfeladatot ellátó szerveknél alkalmazható iratkezelési szoftverekkel szemben támasztott követelményekről, az eltelt időszakban több jogszabály is foglalkozott a kérdéskörrel, a területet szabályozó dokumentumok közül még mindig

hatályos a 335/2005. (XII. 29.) Korm. rendelet a közfeladatot ellátó szervek iratkezelésének általános követelményeiről [115]. A dokumentum a hagyományos papír alapú iratkezelés és az elektronikus iratkezelés mozzanatait, folyamatait írja le – terjedelmét tekintve nem kirívóan nagyméretű, ugyanakkor fontosságát külön érdemes hangsúlyozni, mert a tárgyalt fogalmak az e-közigazgatás alapját képezik. A rendelet meghatározza az iratkezelés folyamatát, ahol a következő tevékenységekkel találkozhatunk: küldemények átvétele, küldemények felbontása és érkeztetése, iktatás, szignálás, kiadmányozás, expediálás, irattározás, selejtezés és levéltárba adás. [115, 4. fejezet]

A felsorolt fogalmak iratkezelési szoftverbe szervezését a közfeladatot ellátó szerveknél alkalmazható iratkezelési szoftverekkel szemben támasztott követelményekről szóló 3/2018. (II. 21.) BM rendelet [116] szabályozza, melynek értelmében 2018. március 1-ét követően már csak a szabályozásnak megfelelő iratkezelési szoftverek alkalmazása engedélyezett [116, 2.41.2] a közszférában.[82] Az iratkezelési szoftverek folyamattámogatási képességét az iratok[83] és az ügyiratok[84] esetében a következőképp határozza meg a rendelet. *„Az ISZ az ügyirathoz és az iktatott irathoz kapcsolódóan képes támogatni*

1. *„a feladat kiosztását;*
2. *a kiadott feladathoz határidő rendelését;*
3. *több feladat hozzárendelését;*
4. *egy feladat több szereplőhöz történő hozzárendelését;*
5. *a feladat címzett általi megtekintésének naplózását;*
6. *egy feladathoz több kapcsolódó feladat létrehozását;*
7. *azt, hogy a feladat kiosztója bármikor lezárja vagy törölje a feladatot;*
8. *azt, hogy a feladat címzettje rögzíthesse a feladat elintézését;*
9. *a kiadott feladat továbbdelegálásának lehetőségét*
10. *azt, hogy az egy ügyirathoz vagy irathoz tartozó feladatok legyenek listázhatóak és személyenként csoportosíthatóak."* [116, 2.7.14]

[82] A fővárosi és megyei kormányhivatalok esetében 2021. január 1-ét követően kell az iratkezelési szoftverekre vonatkozó rendeletet alkalmazni [116, 2.41.3]

[83] Irat: *„valamely szerv működése vagy személy tevékenysége során keletkezett vagy hozzá érkezett, egy egységként kezelendő rögzített információ, adategyüttes, amely megjelenhet papíron, mikrofilmen, mágneses, elektronikus vagy bármilyen más adathordozón; tartalma lehet szöveg, adat, grafikon, hang, kép, mozgókép vagy bármely más formában lévő információ vagy ezek kombinációja;"* [117, 3.(c)]

[84] Ügyirat: *„egy ügyben keletkezett valamennyi irat;"* [116, 2.36]

A felsorolásban szereplő követelmények a hagyományos iratkezelési tevékenységeken túl [116, 4. fejezet] egyfajta iratkezelési szoftverbe ágyazott feladatkezelő rendszer meglétét határozzák meg. A felsorolt pontok konkrét iratkezelési szoftverekkel szemben támasztott követelményekre lefordítva azt jelentik, hogy egy ügyirathoz, irathoz kapcsolódóan több párhuzamos feladat végrehajtását is lehetővé kell tenni. Az Általános Adatvédelmi Rendelet terminológiájával: egy informatikai nyilvántartáson belül több adatkezelési tevékenység párhuzamosan történő végzését is lehetővé kell tenni. A követelményekből az is kiderül, hogy az informatikai nyilvántartás olvasása, személyekhez kötése a párhuzamosan végzett feladatok kapcsán is megvalósítandó funkció. A továbbiakban a beérkező és kimenő küldemények problémakörét, majd a több szálon futó ügyintézés és a feladatkezelés témáját járjuk körül a hierarchikus felépítésű szervezetek életében. Természetesen a bemutatott tervezési minták figyelembe veszik a GDPR alapelveket.[85]

5.3 A FELELŐSHOLDER TERVEZÉSI MINTA

Az információt hordozó küldemények sok ezer éves múltra tekintenek vissza, a küldemény fogalma a hozzá tartozó küldővel és címzettel az első írott üzenetek kézbesítésekor jelent meg a kőtáblák és a papirusz tekercsek korában. Napjainkban a papír alapú küldemények felépítése nem sokat változott, sőt, az informatikai hálózatok protokolljain is tetten lehet érni a küldemény fogalmát és a megfelelő formátumban – leggyakrabban IP címek segítségével – megadott küldőt és címzettet.

A bevezetésben áttekintett jogszabályok a kormányzati szervek szabályozott iratkezelési folyamatait írják le. Ezen iratkezelési folyamatokat két élesen elkülönülő halmazra lehet bontani. A kormányzati szervhez beérkező, illetve az onnan elküldésre kerülő küldemények folyamatai együttesen alkotják az első halmazt. A második halmazt az ügyintézéshez kapcsolódó iratkezelési folyamatok teszik ki. Az utóbbi halmazhoz tartozó folyamatokat a későbbiekben tárgyaljuk.

Egy papír alapú iratkezelést folytató kormányzati szerv a küldeményekkel kapcsolatos nyilvántartások vezetésére a hagyományos értelemben vett érkeztetőkönyvet, iktatókönyvet, postázási naplót és hasonló társaikat használta és használhatja. Egy informatikai nyilvántartást megvalósító iratkezelési szoftvernek egyaránt képesnek kell lennie a papír alapú és az elektronikus iratok érkeztetésére, postázására és a

[85] GDPR alapelvek: *célhoz kötöttség, az adattakarékosság, a pontosság, a korlátozott tárolhatóság, az integritás és a bizalmas jelleg*

megfelelő nyilvántartások vezetésére. A felsorolt feladatok ellátásához egy átfogó, jól átgondolt adatmodell szükséges.

Az adatmodell kialakítása előtt a 335/2005. (XII. 29.) Korm. rendelet iratkezelési folyamatokat leíró 4. fejezetének bejövő küldeményekkel, érkeztetéssel, iktatással és expediálással foglalkozó paragrafusainak elemzése következik a bejövő és a kimenő küldemények szemszögéből. [115, 18-39, 55-58]

A fizikai küldemények átvételére a címzett, a szervezet vezetője, az iratkezelést felügyelő vezető, a postai meghatalmazással rendelkező személy, az ügyfélszolgálat és az ügyeleti szolgálat jogosult, illetve elektronikus küldemények esetén az erre a célra szolgáló informatikai rendszer. [115, 19] A továbbiakban feltételezzük, hogy egy iratkezelési szoftverben kell megvalósítani a felsorolt követelményeket, azaz a küldemény átvételekor a címzés alapján a legbővebb – akár múltbéli – adatokat is felajánlva kell lehetőséget biztosítani az átvevőnek a címzett megkereséséhez. Erre azért van szükség, mert egy átszervezés esetén elképzelhető, hogy az adott személy vagy szervezeti elem már más adatokkal rendelkezik, mint amelyekről a feladó a küldemény elküldésének pillanatában tudott.

Ha tovább vizsgáljuk a jogszabályban foglaltakat, akkor azt is megtudhatjuk, hogy a küldemény felbontása jogosultsághoz kötött tevékenység, melyet a minősített iratok kivételével a címzett, az arra írásban felhatalmazott személy vagy egy meghatározott szervezeti elem dolgozója végezhet el. A digitális küldemények bontását egy erre a célra alkalmazott informatikai rendszer hajthatja végre, amely része lehet az iratkezelési szoftvernek, vagy akár tőle elkülönülten is működhet. [115, 27] Fizikai iratok esetében a saját kezűleg felbontandó küldeményeket a címzettnek magának kell felbontania, míg egy iratkezelési szoftver esetén elektronikus állomány megismerését a címzett számára kell először lehetővé tenni. [115, 28] Utóbbi esetben egy átszervezés érdekes kérdéseket vethet fel, elképzelhető, hogy a címzett már nem képezi részét az informatikai rendszernek, ekkor a megfelelő jogutódnak kell elvégeznie a postabontást, az elektronikus állomány megismerését.

Az iktatással kapcsolatos követelmények között szerepel, hogy az iktatókönyvnek kötelezően tartalmaznia kell a küldő megnevezését és azonosító adatait [115, 39.(f)], valamint a címzett megnevezését és azonosító adatait. [115, 39.(g)] Fizikai iratkezelés esetén ezek az adatok nehezen tarthatók karban, egy emelt szintű informatikai szolgáltatás keretében akár a múltbeli eredeti adatok feltüntetése mellett az időközben megváltozott, aktuális adatok megjelenítése, esetlegesen kereshetővé tétele

nagyban segítheti az ügyintézési, ügyviteli folyamatokat. Az expediálással – a küldemény feladásával – kapcsolatos követelmények nem térnek ki külön a küldőre és a címzettre [115, 55-58], ugyanakkor a címzés kitöltésekor a küldőnek nagy segítséget nyújthat egy iratkezelési szoftver, ha egy válaszirat esetén az eredeti küldőt fel tudja ajánlani alapértelmezett címzettként, a címzett legfrissebb adataival. A 60. ábrán a küldemény és a hozzá kapcsolódó küldő és címzett primitív adatmodellje szerepel.

60. ábra A küldővel és címzettel rendelkező küldemény primitív adatmodellje (saját szerkesztés)

A fenti modell nem tesz különbséget a kormányzati szervhez érkező kimenő és bejövő küldemények között, a küldő és a címzett általánosan van megjelenítve. Vizsgáljuk meg ezt a modellt a GDPR alapelvek és a küldemény küldőjén és címzettjén végzett CRUD műveletek szemszögéből. Az egyszerűbb tárgyalás kedvéért a küldő legyen minden esetben egy természetes személy, míg a címzett egy felelős, ami lehet egy szervezeti elem vagy egy ügyintéző – közfeladatot ellátó szervnél munkaviszonnyal rendelkező természetes személy.

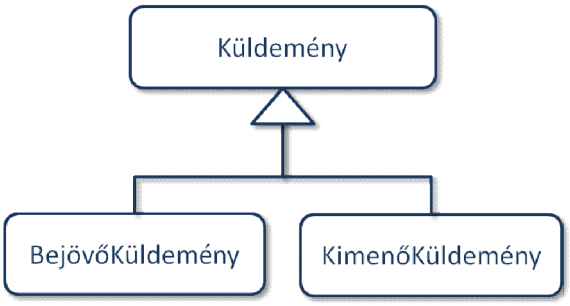
1. CREATE – a beérkező küldemény létrehozása során létrejön a küldő és a hozzá kapcsolódó küldemény, valamint a címzett, ami egy szervezeti elem vagy egy ügyintéző lehet. Abban az esetben, ha a küldő több küldeményt is küld a kormányzati szerv számára már azonnal problémákba ütközhetünk, mert ha a küldő, illetve a címzett adatai módosulnak, akkor minden esetben új küldőt kell létrehozni és a címzettet is duplikálva kell felvenni, így sérül az adattakarékosság elve.
2. READ – ha több küldeményt is küldött már a küldő, akkor ebben a modellben egy adatváltozás problémát jelenthet, mert a küldőhöz tartozó személyes adatok megváltoztatása a korábbi küldemények küldővel kapcsolatos adatait inkonzisztenssé tehetik. A címzettek esetében hasonló problémákba ütközhetünk. Mindig csak az utolsó állapot nyerhető ki az informatikai nyilvántartásból.
3. UPDATE – egy név, postai cím, elérhetőség változás bejelentése vagy egy ügyintézés közben végrehajtott átszervezés inkonzisztens állapotot eredményezhet a küldemény vonatkozásában. Teljes egészében vagy részben sérülhetnek a korábbi küldemények esetében a küldő és a címzettek adatai.

4. DELETE – a küldőre vagy a címzettre vonatkozó törlés művelete ebben a modellben adatvesztéssel jár, inkonzisztens állapotot eredményez, nem támogatott művelet. Elképzelhető, hogy a későbbiekben a GDPR térnyerése kapcsán még a közfeladatot ellátó szervek iratkezelésében is előírássá válhat a törlés művelete bizonyos esetekben.

Az iménti elemzés alapján látható, hogy a küldemény primitív modellje önmagában kevés a probléma megfelelő kezeléséhez, a jogszabályi követelményekből fakadó aktív és történeti értékek tárolását a vázolt modell nem támogatja megfelelően. A vizsgálat során ki sem tértünk a kimenő küldemények kérdésére, amikor a küldő és a címzett szerepet cserél. Ebben az esetben a küldő egy szervezeti elemhez tartozó ügyintéző, míg a címzett egy természetes személy vagy egyéb tetszőleges elérhetőséggel rendelkező külső entitás (gazdasági társaság, kormányzati szerv, stb.).

5.3.1 Bejövő és kimenő küldemény fogalma

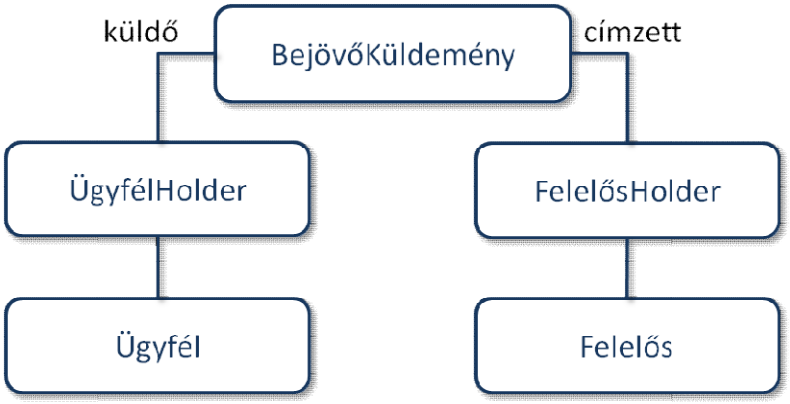
A küldemények vonatkozásában a jogszabályi követelmények kielégítéséhez egy részletesebb modellre van szükség, ahol a bejövő és kimenő küldemények külön kezelhetők. A probléma kezelésére a küldemény fogalmának további bontására van szükség. A 61. ábrán a korábban tárgyalt Küldemény szétbontását láthatjuk egy BejövőKüldemény-re és egy KimenőKüldemény-re, az absztrakció lehetővé teszi a Küldemény hozzákapcsolását tetszőleges adatkezelési tevékenységhez – ügyintézési folyamathoz –, míg a belőle származó két küldeménytípus speciális viselkedése a továbbiakban külön-külön modellezhető, kényelmesen kezelhető.



61. ábra Küldemény tervezési minta (saját szerkesztés)

A továbbiakban a bejövő küldemények problémakörének további vizsgálata következik. Célunk egy olyan modell kialakítása, ahol az aktív és a történeti adatok megfelelően és rugalmasan tárolhatók az ügyfelekhez és a címzettekhez kapcsolódó személyes adatok és egyéb információk vonatkozásában. Az ÜgyfélHolder és FelelősHolder tervezési minták alkalmazása célszerű lehet a bejövő küldeményekhez kapcsolódó szereplők kapcsán. Ebben az esetben a bejövő küldemény küldőjét egy

ÜgyfélHolder segítségével, míg a címzettet egy FelelősHolder-el kapcsolhatjuk a BejövőKüldemény-hez. A kialakított modell a 62. ábrán látható, a továbbiakban el fogjuk végezni a BejövőKüldemény tervezési minta és a kapcsolódó CRUD műveletek elemzését a korábbi vizsgálatokhoz hasonlóan.

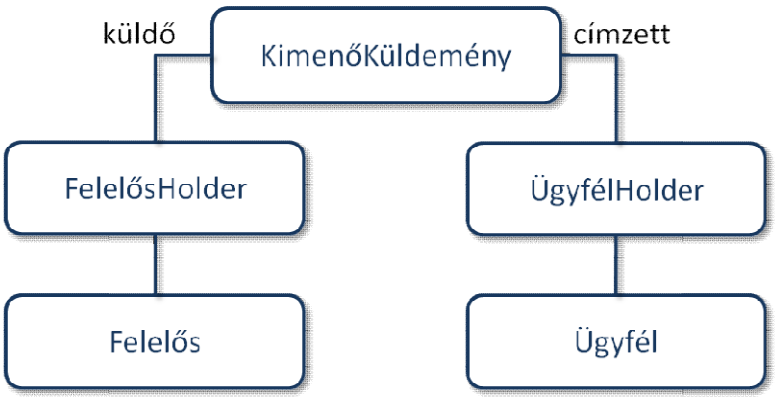


62. ábra BejövőKüldemény tervezési minta (saját szerkesztés)

1. CREATE – egy kormányzati szervhez beérkező küldemény átvételekor létrejön a bejövő küldemény, valamint ha az ügyfél már ismert, akkor az ügyfelet és a bejövő küldeményt összekapcsoló ÜgyfélHolder. Ha egy új ügyfél a küldő, akkor létre kell hozni a megfelelő Ügyfél entitást is. Az ÜgyfélHolder-ek a bejövő küldemények altípusai szerint tovább specializálhatók. A küldemény megjelenési formája (fizikai, elektronikus) és a feladóval kapcsolatos adatok figyelembe vehetők az ÜgyfélHolder kialakításakor. A tárolt adatok az ügyintézési folyamatok számára testre szabhatók. A kormányzati szervhez tartozó felelős szervezeti elemről, illetve ügyintézőről feltételezhető, hogy a bejövő küldemény átvételének pillanatában az adott informatikai rendszerben léteznek – egyébként téves címzésről beszélnénk, így a FelelősHolder tervezési minta segítségével a címzett a bejövő küldeményhez kapcsolható. Ekkor a címzett és bejövő küldemény közötti kapcsolat a használati esetek tükrében az ügyfelek esetével analóg módon megfelelően specializálható.
2. READ – ha a fentiek szerint hozzuk létre a bejövő küldeményekhez kapcsolódó adatmodellt, akkor lehetőség nyílik a történeti adatok kinyerésére a küldő és a címzett vonatkozásában a Holder-ekből, ha küldő vagy a címzett már nem képezi részét az informatikai rendszernek. Ha még aktív ügyintézési folyamatok kapcsolódnak a bejövő küldeményhez, akkor a Holder-eken keresztül elérhető az aktív küldő (Ügyfél) és címzett (Felelős), így a folyamatok helyessége garantálható.

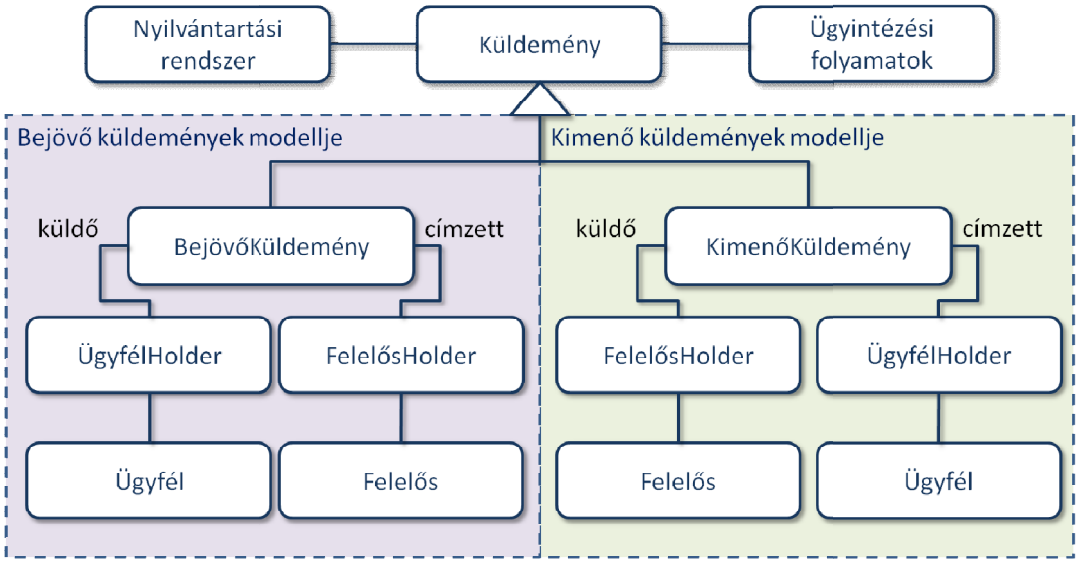
3. UPDATE – a küldők és a címzettek adataiban bekövetkező változásokat a bemutatott tervezési minta segítségével rugalmasan lehet kezelni. A beérkezés pillanatában eltárolt adatokat a Holder-ek a szükséges időtartamig képesek eredeti formájukban tárolni, ugyanakkor az aktív Ügyfél és Felelős entitásokon végzett műveletek ezeket az információkat nem veszélyeztetik.
4. DELETE – a küldők és a címzettek végleges törlését a vázolt modell támogatja, mert a Holder-ek a szükséges időtartamig tárolhatják mindkét fél adatait. A kialakított modell képes kezelni egy átszervezési folyamat következményeit – szervezeti elemek jogutóddal történő felszámolása, személyek törlése – a felelősök esetében a FelelősHolder mutatójának átállításával a jogutód felelősre. Ezzel a beérkező saját kezűleg felbontandó fizikai küldemények és az elektronikus küldemények kézbesítése is garantálható, amely a zavartalan ügyintézés egyik alappillére. A küldők fizikai törlése kezelhető a modell segítségével, akár egy aktív ügyintézési folyamattal párhuzamosan is. Az ügyintéző a válaszküldemény elküldése előtt értesülhet a küldő törléséről az ÜgyfélHolder segítségével és intézkedhet a megfelelő válaszcím lekérdezéséről, az ügyintézési folyamat megfelelő lezárásáról.

A küldemény primitív modelljében tapasztalható rugalmatlanságokra a BejövőKüldemény tervezési minta megfelelő válaszokat ad. Az Általános Adatvédelmi Rendelettel kapcsolatos várható jogharmonizációból fakadó követelmények kezelésére megoldás lehet a bemutatott tervezési minta alkalmazása az iratkezelési szoftverekben. A kimenő küldemények esetében hasonló problémákkal találkozhatunk, mint amelyeket a bejövő küldeményeknél tapasztaltunk, azonban a tervezési minta részletes elemzése meghaladja e tanulmány kereteit. A KimenőKüldemény tervezési minta a 63. ábrán látható, ez esetben a küldő egy felelős – általában egy ügyintéző –, míg a címzett az ügyfél.



63. ábra A KimenőKüldemény tervezési minta (saját szerkesztés)

A Küldemény absztrakció bevezetésével, a BejövőKüldemény és a KimenőKüldemény tervezési minták kialakításával megfelelően kezelhetők a küldőkön és a címzetteken végzett CRUD műveletek. A Küldemény entitás tetszőlegesen hozzákapcsolható az iratkezelési szoftverek nyilvántartási rendszeréhez, az ügyintézési folyamatokhoz, segítségével kezelhetővé válik a visszavárólag elküldött küldemények, a többszöri érkeztetés és postázás problémaköre is.



64. ábra A Küldemény tervezési minta és iratkezelési fogalmak kapcsolata (saját szerkesztés)

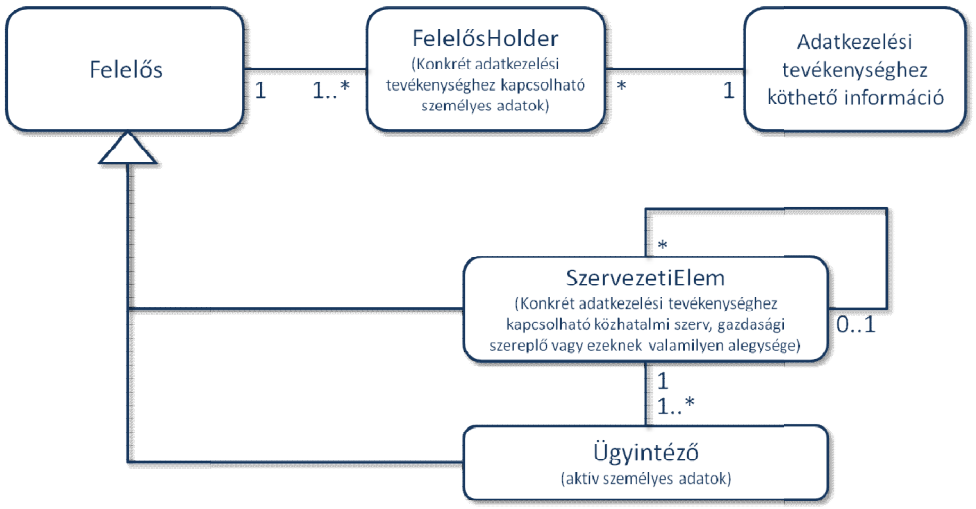
5.3.2 Hierarchikus szervezetek modellezése

Már az ókorban is komoly jelentőséggel bírt, hogy egy hadsereg mennyire volt képes pontosan, összehangoltan végrehajtani a hadvezér utasításait egy csata során, a fegyveres konfliktus végkimenetele nagy részben a parancsvégrehajtás minőségétől függött. Egy kisebb létszámú, de szervezett hadsereggel szemben az ellenségnek minden esetben nehéz dolga volt. Ebből a megfontolásból a rendelkezésre álló haderőt, haderőnemekre és ezen belül jól vezényelhető egységekre és alegységekre formálták, melyek a mindennapokban a fegyverforgatást és a parancsvégrehajtást gyakorolták, így a parancsnokok, ezáltal a vezetési szintek szerepe kulcsfontosságúvá vált. A technikát legsikeresebben a Római Birodalom alkalmazta, amely így egy több évszázadon át regnáló szuperhatalommá válhatott a saját korában.

Ez a fajta hierarchikus szervezeti felépítés napjainkban is megfigyelhető a különböző fegyveres erőknél, rendvédelmi és államigazgatási szerveknél. A katonai és félkatonai szervezetek esetében a parancselvű működés a szervezetek feladatrendszeréből fakad, alapvetően abból, hogy a fegyveres testületek számára parancs alapján akár erőszak alkalmazása is végrehajtandó feladat lehet. A közigazgatási és ezen belül az

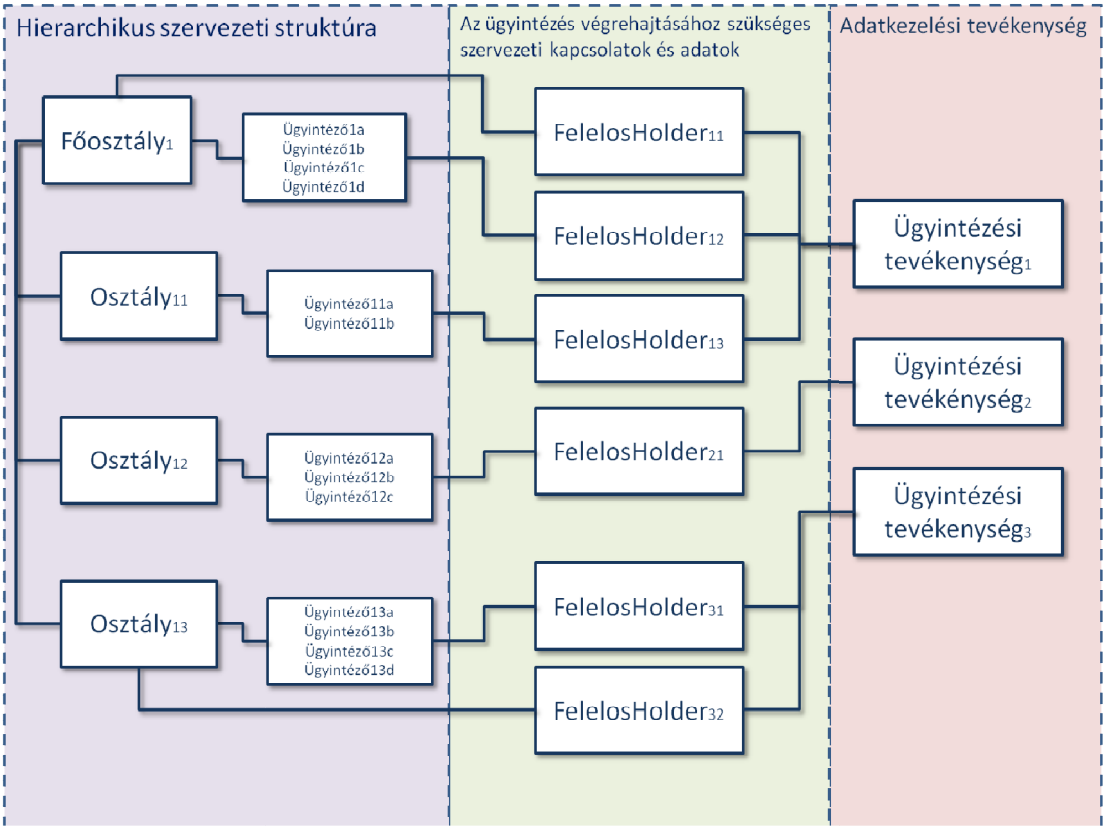
államigazgatási szervek számára a „parancs” az adott ország törvényeiben, jogszabályaiban keresendő, melyek végrehajtása hasonlítható egy katonai szervezet parancsvégrehajtó mechanizmusához – jó esetben kizárólag demokratikus keretek között.
Egy hierarchikus szervezet számára fejlesztett informatikai rendszer esetében a helyesen működő folyamatok garantálásához szükséges a megfelelő informatikai modell alkalmazása. Az e-kormányzás és a hatékony közigazgatás megvalósításában alapvető szerepe lehet a kiválasztott modellnek. Bizonyos szervezeti létszám fölött minden területen megfigyelhető a vezetési szintek megjelenése, ilyenkor az informatikai rendszerekben olyan modell kialakítása szükséges, amely tetszőleges mélységben képes kezelni a különböző szervezetek felépítését.

A FelelősHolder tervezési minta kiterjesztését a 65. ábrán láthatjuk. Az ábra tartalmazza a számosságra utaló 0, 1, * (tetszőleges számú) jelöléseket. A modellre igazak a következő állítások: egy adatkezelési tevékenységhez több FelelősHolder tartozhat, ahol a kapcsolódó felelős lehet egy szervezeti elem vagy egy ügyintéző. Egy felelőshöz akár több FelelősHolder is tartozhat, akár más és más adattartalommal, az adott adatkezelési tevékenységből fakadó használati eset függvényében. Minden szervezeti elemhez legalább egy ügyintézőnek kell tartoznia, ez a kitüntetett ügyintéző reprezentálja a szervezeti elem vezetőjét, hadtudományi megközelítésben a parancsnokot. Magától értetődően további ügyintézők szervezeti elemhez kapcsolása is megengedett. A hierarchikus szervezeti modell a szervezeti elemek egymáshoz kapcsolásával valósul meg, minden szervezeti elemnél lehetőség van egy felettes szervezeti elem hozzákapcsolásához. Egy elektronikus iratkezelési szoftver számára a bemutatott tervezési minta alapot képezhet a hierarchikus szervezeti felépítés modellezésére.



65. ábra A FelelősHolder tervezési minta kiterjesztése hierarchikus szervezetekre (saját szerkesztés)

Az alábbi ábrán a kiterjesztett FelelősHolder tervezési minta egy lehetséges használati esetét láthatjuk. A 66. ábra egy kétszintű szervezetet mutat be, ahol a legmagasabb vezetési szintet egy főosztály és a hozzá tartozó alsó szinteket három osztály alkotják.



66. ábra A hierarchikus szervezetekre kiterjesztett FelelősHolder tervezési minta használati esetei több ügyintézési folyamat esetén (saját szerkesztés)

Három ügyintézési folyamatot láthatunk, ahol az első ügyintézési folyamat a főosztályhoz kapcsolódik és két természetes személy vesz részt az ügyintézési tevékenységben. A második ügyintézési folyamat egyetlen ügyintézőhöz kapcsolódik, míg a harmadik ügyintézési folyamat egy osztályhoz és egy az osztályon dolgozó ügyintézőhöz kapcsolódik. Utóbbi a hagyományos ügyintézés menetét mutatja be, ezzel az esettel lehet a legegyszerűbben szemléltetni a CRUD műveletek és a hierarchikus szervezetekben végmenő folyamatok kapcsolatait. A vizsgálat szempontjából érdekes lehet az ügyintéző változása, szervezeti elem felszámolása és módosítása

1. CREATE – egy ügy keletkezésekor egy-egy kapcsolat jön létre az ügy és az ügyintéző, valamint az ügy és az adott ügyintéző szervezete között. A kapcsolatok az adott ügytípushoz igazított FelelősHolder-ekkel specializálhatók a GDPR alapelveknek megfelelően.

2. READ – Ameddig az ügyintézés folyamatban van, addig a FelelősHolder-en keresztül az aktív szervezeti és belső személyekre vonatkozó adatok érhetők el. Így pontos információhoz juthat az ügyintéző, illetve a folyamatban résztvevő többi természetes személy is a résztvevő szervezeti elemekről, személyekről. Szervezeti elem esetén példa lehet a szervezeti elem neve, nevének rövidítése, azonosítója, stb. Több természetes személy esetén példa lehet az adott felhasználó neve és beosztása, azonosítója.
3. UPDATE – Egy folyamatban lévő ügyintézés során elképzelhető, hogy az ügyintézésért felelős szervezeti elemek, személyek adatai megváltoznak. Ebben az esetben nagyon fontos, hogy milyen típusú adatról van szó: egy történetiség igazolásához szükséges egyszer letárolandó személyes-, illetve szervezeti adatról vagy az ügyintézést támogató folyamatosan frissítendő adathalmazról. A probléma kezelhető különböző viselkedésű FelelősHolder-ek segítségével. Egyik esetben az inicializáláskor egyszer kerül kitöltésre a szervezetek és a személyes adatok lenyomata és az már soha többé nem frissül a FelelősHolder-ben, míg a második esetben csak a folyamat végén kerül véglegesítésre az adattartalom – mindig a konkrét használati esetnek megfelelően.
4. DELETE – a GDPR alapelvekből kiindulva a szervezeti elemeket és az ügyintézésért felelős személyeket is fizikailag törölhetővé kell tenni a rendszerben. Egy folyamatban lévő ügy kapcsán a másodlagos ügyintézési feladatokat ellátó személyek (segítők, bedolgozók, stb.) törlése akár azonnal elvégezhető művelet lehet, míg az elsődleges ügyintéző törlése már nem az. A probléma kezelhető az ügyintézőkre mutató FelelősHolder-ek segítségével, ha valamelyik ügyintéző eltávolításra kerül a rendszerből, akkor az informatikai rendszeren belül továbbra is nyoma marad annak, hogy milyen ügyintézési tevékenységben vett részt az illető, az adat nem tűnik el. Egy szervezeti elem felszámolása is érdekes kérdéseket vethet fel, mert elképzelhető, hogy valahol tárolni kell a jogelőddel kapcsolatos adatokat, erre ugyancsak megoldás lehet egy speciális FelelősHolder, amelyik a régi szervezet adatait tárolja, de már a jogutódra mutat a fizikai törlést követően.

A hierarchikus szervezetekre kiterjesztett FelelősHolder tervezési minta segítségével egyenként lehet kezelni a szervezetben végbemenő strukturális és személyi változásokat a használati esetek tükrében. A megoldás kellő rugalmasságot kölcsönöz a ter-

vezési feladatokat végző szakembereknek – a Scrum agilis szoftverfejlesztési módszertan esetén a terméktulajdonosnak és a fejlesztőcsapat tagjainak. A megoldás támogatja az egyszálú hagyományos ügyintézési folyamatokat (beérkező irat, válasz irat). Ezen túlmenően a modell lehetőséget biztosít a több párhuzamos szálon végrehajtott, digitális alapokon megvalósuló ügyintézési folyamatok támogatásához is. A vázolt modell jó választás lehet a 3/2018 BM rendeletben foglalt folyamattámogatással kapcsolatos követelmények kielégítésére.

5.3.3 Ügyfolytonosság és változó szervezeti környezet

A hagyományos papír alapú iratkezelés során az ügyek kezeléséért, a jogfolytonosság fenntartásáért az ügyviteli irodák voltak a felelősek. Az iktatólapok, előadói ívek kiadása és ellenőrzése manuális lépésként történt meg. Az átszervezések során az ügyviteli irodák segítségével voltak kezelhetők az ügymenetben kialakult problémák – az új felelősök megkeresése ezen a szinten indult újra, ha az átszervezés során a jogutódokról nem rendelkeztek egyértelműen. Az iratkezelési szoftverek világában az elektronikus iratok és elektronikus állományokhoz kapcsolódó folyamatok kezelése már manuálisan nehézkesen vagy egyáltalán nem kezelhető, az iratkezelési szoftvernek kell képesnek lennie a szervezeti változásokból kialakuló folyamatok támogatására.

A 335/2005. (XII. 29.) Korm. rendelet 4. fejezetében az eddig tárgyalt iratkezelési tevékenységeken túl az iktatással [115, 39-50], a szignálással [115, 51], a kiadmányozással [115, 52-54] és az expediálással [115, 55-58] találkozhatunk.

Szignálás során az illetékes szervezeti egység vezetője vagy megbízottja jelöli ki az érkezett irat ügyintézőjét. [115, 51.(a)] Hierarchikus szervezetek esetében elképzelhető, hogy szignáláskor egy szervezeti elemet jelölnek ki felelősnek, ahol egy újabb szignálás következik. A létrejövő szignálások sorozata ún. szignálási láncot képez, amely érzékeny lehet az átszervezési folyamatokra. A szignálási lánc egy közbülső elemének változása kihathat a szabott határidőkre, felelősökre. Ez a folyamat egy nagy létszámú, többszintű hierarchikus szervezet életében szinte mindennapos lehet. A FelelősHolder tervezési minta segítségével a Szignálás-hoz kapcsolhatók azok a szereplők, akiket valamilyen formában érinthet a szervezeti struktúra változása, ilyenek lehetnek a szignáló, a szignálást rögzítő, a felelős szervezeti elem és az ügyintéző is.

A kiadmányozási folyamat során a külső szervhez vagy külső természetes személyhez – érintetthez – küldött iratot egy kiadmányozónak kell aláírnia. [115, 52.(1)] A hierarchikus szervezetek esetében a fizikai iratok esetében nagyon gyakori a többes aláírás intézménye, amikor a kiadmányozás – végső aláírás – előtt, az egyes szakterületi vezetők is aláírják az adott iratot. Az iratkezelési szoftverek esetében megfelelő folyamattámogatás mellett az aláírások száma csökkenthető, ha a többes aláírások helyét az iratkezelési szoftverben végrehajtott jóváhagyásokkal helyettesítik. Ebben az esetben a szervezeti változásoknak nem kell egyértelműen a tervezet visszautasítását eredményezniük. Amennyiben lehetséges a megfelelő jogutódok azonosítása az átszervezés után, akkor ők is elvégezhetik a jóváhagyási feladatokat és az irat kiadmányozható. A folyamat támogatásához a FelelősHolder tervezési minta alkalmazható a jóváhagyók és az aláírók esetében.

Expediáláskor a szervezeti egység iratkezelőjének dokumentálnia kell a nyilvántartással, továbbítással kapcsolatos információkat. [115, 55] A postázás előtti pillanatban az aktuális ügyintéző és felelős szervezeti egység küldeményekhez kapcsolása lényeges mozzanat lehet, mert elképzelhető, hogy a küldemény visszavárólag kerül elküldésre, illetve válaszirat is érkezhet a kimenő küldemény alapján a szervezethez. Egy átszervezés ezeket a folyamatokat is felboríthatja, a FelelősHolder segítségével az eredeti ügyintéző, illetve a felelős szervezeti elem jogutódja a kimenő küldeményhez kapcsolható, tehát a zavartalan iratkezelési folyamat fenntartható a küldemények vissza- és beérkezésekor.

Az iktatókönyvek tartalmával kapcsolatosan egy nagyon lényeges követelmény jelenik meg az ügyintézést illetően, fel kell tűntetni az ügyintéző szervezeti egységet és az ügyintéző személyét is. [115, 39.(j)]. Egy iratkezelési szoftver vonatkozásában a történeti bejegyzéseken túl, a megfelelően kialakított FelelősHolder-ek segítségével az aktuális, illetve a szervezeti átszervezések után a jogutód felelősök is hozzákapcsolhatók a különböző típusú nyilvántartásokhoz, ezen belül az iktatókönyvekhez. A kialakított megoldás segítségével az iktatókönyvek a papír alapú iratkezelés történeti lenyomatai helyett az iratkezelési szoftver nyilvántartási rendszerének nagyon hasznos, valós idejű nézeteivé válhatnak.

5.4 ÖSSZEGZÉS

Az 5. fejezetben bemutatásra került a GDPR hatóköre és az általános adatkezelési tevékenységekhez köthető fogalmak meghatározása is: *érintett, személyes adat, adatkezelés, adatkezelő, adatfeldolgozó*. Megtudhattuk, hogy a tagállamok felmentést kaptak a védelmi szektorban a GDPR végrehajtására vonatkozóan – abból a meggondolásból kiindulva, hogy ebben a szektorban legalább a rendeletben foglalt, vagy annál szigorúbb szabályozások érvényesek.

Ezt követően az érintettek személyes adatainak és a hozzájuk kapcsolódó adatkezelési tevékenységek primitív modelljének vizsgálata következett a CRUD műveletek szemszögéből, amely rávilágított a primitív modell hiányosságaira az adatmódosítás és a fizikai törlés kapcsán is.

A hiányosságok kiküszöbölésére a *Proxy* és a *Holder* tervezési minták tulajdonságait ötvözve meghatározásra került az *ÉrintettHolder* tervezési minta, amely megfelelő viselkedést mutat az érintetteken végrehajtott CRUD műveletek esetében – a párhuzamosan végzett adatkezelési tevékenységek esetén is.

Ugyan a GDPR terminológiájában az *érintett* fogalma természetes személyeket takar, ezzel együtt az *ÉrintettHolder* tervezési minta kiterjeszthető hierarchikus szervezetek modellezésére is az *ÉrintettHolder*-ből származtatott *FelelősHolder* és *ÜgyfélHolder* tervezési minták segítségével.

A tervezési minták működésének meghatározása ügyviteli példák segítségével valósult meg, ám azok felhasználása tetszőleges katonai szakterület számára is alternatívát jelenthet a hierarchia modellezésére és a szakterület specifikus folyamatok megvalósítására.

Ezután, a Magyar Honvédség ügyviteli folyamatainak bemutatása mellett – a vonatkozó hatályos jogszabályi követelmények értelmezésével – további tervezési minták kidolgozása valósult meg az elektronikus ügyintézési folyamatok támogatásához: a *Küldemény* tervezési minta és a belőle származó *BejövőKüldemény* és *KimenőKüldemény* formájában.

A fejezet záró részében a meghatározott tervezési minták felhasználási lehetőségeinek bemutatása történt meg az iratkezelési szoftverek jogszabályi szinten meghatározott folyamatainak támogatásához – ezzel betekintést nyújtva a dolgozat során azonosított tervezési minták speciális szakterületi alkalmazási lehetőségeihez.

6. ÖSSZEGZETT KÖVETKEZTETÉSEK

Kutatásom megkezdésekor négy hipotézist állítottam fel, amelyek igazolásához előbb szoftvertechnológiai aspektusból elemeztem az Európai Unió és Magyarország kiberbiztonságot érintő szabályozási keretrendszerét. Majd ezt követően feltártam a szoftver alapú szolgáltatások fejlesztésének és üzemeltetésének kérdéskörét a NATO katonai képességfejlesztési módszertanának, valamint a Magyar Honvédség informatikai elvárásainak tükrében. A feltáró munka során nyolcvannégy különböző kutatási követelményt azonosítottam és rendszereztem. Az azonosított követelményeket kiberbiztonsági, módszertani, szoftvertechnológiai, információbiztonsági, minőségbiztosítási és dokumentációs kategóriákba, illetve speciális és általános szintekre soroltam be.

A kialakított követelményrendszer hatvan speciális elvárással tette lehetővé számomra a kutatásom induktív szakaszának megkezdését, amelynek elsődleges célja az agilis szoftverfejlesztési technikák és a szabványos katonai képességfejlesztési módszerek szintetizálása volt. Az első két hipotézisem igazolásához célszerűnek láttam egy katonai célú szoftverfejlesztési módszertan meghatározását, amely hibrid megoldásként ötvözi a NATO SLCM folyamatokat és a Scrum agilis szoftverfejlesztési módszertant, valamint kielégíti a rá vonatkozó speciális követelményeket, a módszertanok közötti lényeges különbségeket a 25. táblázat mutatja be.

	Scrum	**NATO SLCM**	**Military Scrum**
Szerepkörök	ScrumMaster, Terméktulajdonos, Fejlesztőcsapat	Pontosan nem definiált, az ISO 15288 szabvány szervezeti folyamatain alapszik	Megrendelő, Projektmenedzser, Terméktulajdonos, Fejlesztő, Üzemeltető, Minőségbiztosító
Felhasználási terület	Szoftverfejlesztés	Általános katonai képességfejlesztés	Szoftver alapú szolgáltatások fejlesztése
Szakaszok/ lépések	Sprint tervezés, fejlesztés	Koncepcionális előtervezés, koncepcióalkotás, fejlesztés, előállítás, felhasználás, támogatás, leszerelés	Koncepcionális előtervezés, koncepcióalkotás, fejlesztés, üzembe helyezés, felhasználás, támogatás
Iterációk kezelése	egy szintű: fejlesztési sprintek	egy szintű: módosítási és frissítési eljárás	kétszintű: Military Scrum fejlesztési menetek, illetve a fejlesztés szakaszában fejlesztési sprintek.
Dokumentáció	Termék feladatlista (Story), fejlesztési feladatlista (Task), egyéb fejlesztés során keletkező tervezési, tesztelési dokumentum	A D-1, D-2, D-3 speciális kutatási követelmények tartalmazzák a szoftverfejlesztés vonatkozásában értelmezhető dokumentumokat.	Igények, követelményhalmazok, termék feladatlista, sprint feladatlista, átvételi tesztforgatókönyv, hibajegyzék, változtatási igények.

25. táblázat A Scrum, NATO SLCM és a Military Scrum módszerek összevetése (saját szerkesztés)

A Military Scrum szoftverfejlesztési módszertan meghatározása sikerrel járt, az előírt dokumentációk számát sikerült csökkenteni a modern technológiák javára, oly módon, hogy a módszertant a képességei továbbra is alkalmassá teszik a katonai célú felhasználásra a NATO követelmények alapján – az egyes követelményeknek való megfelelést a 2. számú melléklet tartalmazza, többek között az SLCM által előírt dokumentációs elvárásoknak való megfelelést is a Military Scrum alkalmazásakor. Így a kialakított módszertan a Magyar Honvédség számára is alternatívát jelent a katonai célú szoftver alapú szolgáltatások kialakításához.

A fennmaradó két hipotézis igazolásához a Magyar Honvédség által végzett ügyviteli folyamatokat és az elektronikus ügyintézés kérdéskörét vizsgáltam az alapvető adatkezelési (CRUD) műveletek végrehajtásának szemszögéből. A műveletek elvégzését a hierarchikus struktúrát alkotó szervezeti elemeken értelmeztem, feltételezve az általuk párhuzamosan folytatott adatkezelési tevékenységeket a GDPR alapelveinek szem előtt tartása mellett: *célhoz kötöttség, adattakarékosság, pontosság, korlátozott tárolhatóság, integritás és bizalmas jelleg*.

A kutatás eredményeként általános tervezési mintákat alakítottam ki, amelyek támogatják a párhuzamosan végzett, a GDPR alapelveknek megfelelő adatkezelési folyamatokat a hierarchikus felépítésű szervezetek számára, így azok különböző katonai szakterületek számára is alkalmazhatók.

Kutatási hipotézisek vizsgálata

A NATO életciklus menedzsment folyamatainak, a Magyar Honvédség informatikai kérdésekben mérvadó dokumentumainak, valamint a magyar és az EU-s kibervédelmi stratégia alkotóelemeinek feltáró elemzésével 58 darab olyan speciális kutatási követelményt sikerült azonosítani, amelyek alapján lehetővé vált az 1. és 2. számú kutatási hipotézisek vizsgálata.

[M-5–M-20] – 16 db módszertani követelmény (NATO SLCM)
[S-5–S-25] – 21 db szoftvertechnológiai követelmény (NATO SLCM és MH HID)
[R-1–R-5] – 5 db követelményelemzési elvárás (MH Informatikai Szabályzat)
[Q-1–Q-4] – 4 db minőségbiztosítási elvárás (HIRBizt. stratégia)
[I-11–I-12] – 2 db információbiztonsági követelmény (EU NIS direktíva)
[D-1–D10] – 10 db dokumentációs követelmény (NATO SLCM)

A Honvédelmi Minisztériumra és a Magyar Honvédségre vonatkozó hatályos jogszabályok, valamint az általános adatbiztonságot előirányzó GDPR rendelet, illetve a NIS direktíva korábbi elemzése során kialakított követelmények alapján vált lehetővé a 3. és 4. számú kutatási hipotézisek vizsgálata. A 2. számú mellékletben található (I-9) és (I-10) speciális információbiztonsági követelmények kielégítését a Military Scrum és a Holder tervezési minták együttes alkalmazása teszi lehetővé.

1. hipotézis: Az agilis szoftverfejlesztési módszertanok alkalmazhatók katonai célra a különböző védelmi szakterületek speciális követelményeinek figyelembe vételével.

A kutatás induktív szakasza során meghatározott Military Scrum szoftverfejlesztési módszertan agilis módszereket és technikákat használ fel a szoftverek és a szoftver alapú szolgáltatások fejlesztésének támogatásához, egyúttal megfelel az azonosított NATO és MH katonai elvárásoknak, így katonai célra alkalmazható. A módszertan által adott konkrét válaszok a kutatási követelményrendszerre a 2. számú mellékletben találhatók.

A kutatási eredmények alapján igazoltnak tekintem az 1. hipotézist.

2. hipotézis: Katonai alkalmazás során is azonosíthatók az agilisan végrehajtott projekt fázisai, amelyekhez kidolgozhatók a megfelelő dokumentációs sablonok.

A Military Scrum hat szakaszból álló iterációt határoz meg a teljes szoftverfejlesztési folyamat számára: koncepcionális előtervezés, koncepcióalkotás, fejlesztés, üzembe helyezés, felhasználás, támogatás. A módszertan a szoftverfejlesztési projektfolyamatok támogatásához projekt menedzsment sablont biztosít, amely hozzárendeli az egyes szakaszokhoz a megfelelő táblázatokat: igények, követelményhalmazok, termék feladatlista, sprint feladatlista, átvételi tesztforgatókönyv, hibajegyzék, változtatási igények.

A kutatási eredmények alapján igazoltnak tekintem a 2. hipotézist.

3. hipotézis: Egy agilis szoftverfejlesztési módszertan alkalmazása során rugalmasabb és a valóságot jobban modellező katonai célú szoftverek fejleszthetők ki.

A hierarchikus katonai szervezetek ügyviteli folyamataira támaszkodva, a természetes személyek és a szervezeti elemek adatain párhuzamosan végrehajtott CRUD műveletek vizsgálatával, rugalmas és valóságot tükröző tervezési minták kerültek kialakításra. A kialakított tervezési minták támogatják a fizikai törlés műveletét, a múltbéli és a naprakész adatok szeparált kezelését. Egyaránt alkalmazhatók a természetes személyek adataira és a katonai hierarchia részét képező egyéb entitásokra – a katonai célú felhasználás lehetőségét a Military Scrum által előírt módszertani és technológiai elvárások teremtik meg.

A kutatási eredmények alapján igazoltnak tekintem a 3. hipotézist.

4. Új szoftvertervezési minták azonosíthatók egy katonai célra agilisan fejlesztett szoftver előállítása során, amelyek a későbbiekben széleskörűen alkalmazhatók

Az *ÉrintettHolder*, az *ÜgyfélHolder*, a *FelelősHolder* és a *Küldemény* tervezési minták azonosítása a természetes személyeken és a szervezeti elemeken végzett CRUD műveletek elemzésével valósult meg a használati esetek tükrében. Az eljárás a Military Scrum szoftverfejlesztési módszertan során előírt TDD és DDD módszerekkel mutat analógiát. A CRUD műveletek leírásai képezik a tesztforgatókönyveket, a tervezési minták pedig a kialakított fogalomteret. Az eredményként létrejött modell univerzális, megfelelő szoftvertechnológia háttér mellett a kormányzat és a magánszektor is alkalmazhatja ügyintézési, kapcsolattartási folyamatok megvalósításához.

A kutatási eredmények alapján igazoltnak tekintem a 4. hipotézist.

6.1 ÚJ TUDOMÁNYOS EREDMÉNYEK

- Elsőként dolgoztam ki a védelmi fejlesztések szempontrendszerét figyelembe vevő, az általános kiberbiztonság fokozását elősegítő, a gyakorlati felhasználhatóságra is hangsúlyt fektető huszonnégy kritériumot.
- Kidolgoztam egy speciális NATO SLCM kompatibilis követelményrendszert a katonai célra alkalmazható szoftverfejlesztési módszertanok kiegészítésére.
- Meghatároztam a Military Scrum szoftverfejlesztési módszertant, amely megfelel a kutatás során kialakított követelményrendszernek.
- A GDPR alapelveknek megfelelő modellbe helyeztem a hierarchikus szervezetekben megvalósuló párhuzamos adatkezelési folyamatokat, majd ez alapján kidolgoztam egy katonai ügyviteli folyamatokon alapuló feladatrendszert.
- Meghatároztam az *ÜgyfélHolder*, a *FelelősHolder* és a *Küldemény* tervezési mintákat, amelyek támogatják a párhuzamosan megvalósuló adatkezelési folyamatokat és alkalmazhatók hierarchikus struktúrában működő szervezetek számára.

6.2 GYAKORLATI FELHASZNÁLHATÓSÁG

Az értekezés első felében feltárt huszonnégy általános kutatási követelmény legalább ugyanennyi gyakorlati felhasználási lehetőséget rejt magában. A továbbiakban a különböző kutatási területekről azonosított általános követelmények és a kutatás tudományos eredményeinek gyakorlati felhasználási lehetőségei következnek.

Kritikus infrastruktúra védelem és kiberbiztonság

A különböző biztonsági komplexumhoz köthető információs rendszerek kialakítása során egyaránt törekedni kell a biztonságra, különös tekintettel a katonai biztonságra. (K-1)

A kutatási hipotézisek igazolása során kialakított Military Scrum szoftverfejlesztési módszertan segítségével olyan szoftver alapú szolgáltatások kialakítása válik lehetővé, amelyek teljes élettartamuk alatt egyenletes biztonsági és megbízhatósági mutatókkal rendelkeznek. A módszertan alkalmazható a katonai biztonsági komplexum információs rendszereinek kialakításához, hisz a szoftver alapú szolgáltatások fejlesztési folyamatainak NATO SLCM-ben való értelmezésével került kialakításra. A biztonsági elvárások már a koncepcionális előtervezés szakaszától integrálhatók a módszertannal megvalósított projektek követelményrendszerébe. A Military Scrum fejlesztést és üzemeltetés támogató eljárásai garantálják a biztonsági követelmények megvalósulását és azok későbbi fenntartását a rendszerek teljes élettartama alatt. Az eljárások alkalmazása nem csak a katonai biztonsági komplexumon belül lehetséges, gazdasági és állami szereplők is egyaránt választhatják a szoftverfejlesztési projektjeik megvalósítása során.

Olyan infokommunikációs technológiák és módszerek kialakítása szükséges, amelyek biztosítják a megfelelő védekezési és reagálási képességeket. (K-2)

A fenti követelménynek való megfelelés nem kizárólag szoftvertechnológiai kérdés, a különböző kritikus infrastruktúrák és létfontosságú rendszerek védelmét az Európai Unió és Magyarország is törvényi szinten tárgyalja, azonban a szoftver alapú szolgáltatások kérdéskörét tekintve a szabályozás még nem tekinthető véglegesnek. Ezen a területen a Military Scrum technológiát és módszereket ad a szolgáltatások tárgyát képező fejlesztett szoftverek tervezéséhez, fejlesztéséhez, üzemeltetéséhez, felhasználásához és támogatásához, értelmezve a biztonsági követelmények fogalmát is. A

módszertan segítségével fejlesztett szoftverek a biztonsági követelmények integrált kezelésének köszönhetően a kezdetektől védekezési és reagálási képességekkel alakíthatók ki.

A számítógépes hálózatokat és a kapcsolódó információs rendszereket fel kell készíteni az összehangolt támadásokkal szemben. (K-3)

A követelmény nem kizárólag szoftvertechnológiai, ugyanakkor a felhőalapú szolgáltatási modell tetején elhelyezkedő szoftver alapú szolgáltatások esetében a Military Scrum alkalmazásával a biztonsági követelmények meghatározásától kezdve a fejlesztés, az üzembe helyezés és a támogatás részeként az automatizált rendszerfelügyelet képessége kialakítható a SaaS rétegben. Amennyiben különböző információs rendszereken belül azonos jellegű incidensek tapasztalhatók az automatizált rendszerfelügyelet által, akkor feltételezhető, hogy összehangolt támadásról van szó és életbe lehet léptetni az egységes kibervédelmi protokollokat.

Szoros együttműködést, hatékony információcserét biztosító infokommunikációs rendszerek kialakítása szükséges a védelmi ágazat szereplői számára. (K-4)

A FelelősHolder és a Küldemény szoftvertervezési minták alkalmazásával a hierarchikus felépítésű szervezeteken belül és a külső információcsere során a GDPR követelményeit kielégítő ügyintézési folyamatok és kétirányú kapcsolattartás valósítható meg. A kutatás során kialakított szoftverfejlesztési folyamatok és az azok alkalmazásán alapuló tervezési minták alternatívaként szolgálnak a jövőben a védelmi ágazat infokommunikációs rendszereinek kifejlesztéséhez és egymáshoz illesztéséhez.

Az informatikai rendszereknek olyan felügyeleti eszközökkel kell rendelkezniük, amelyek lehetővé teszik az észlelési, feldolgozási, és felderítési tevékenységek támogatását.(K-5)

A Military Scrum az első fejlesztési menetben előállított prototípusok esetében már lehetőséget biztosít arra, hogy az észlelési, feldolgozási és felderítési tevékenységek követelményei elvárásként jelenjenek meg a fejlesztett szoftverrel szemben alap szinten. Ezt követően az integráltan végrehajtott fejlesztési, üzemeltetési és felhasználási szakaszokban a képességek fenntartása és növelése lehetséges. Az üzemeltetési környezet kritériumait folyamatosan figyelembe vevő fejlesztés lehetővé teszi, hogy a

szoftver alapú szolgáltatás olyan információkat osszon meg az üzemeltetési környezet rendszerfelügyeleti szoftvereivel, amelyek alapján észlelhetők a szokatlan tevékenységek, azok technikai információi kinyerhetők és feldolgozhatók. Az okok, a külső és belső eredetű tényezők felderítése a feldolgozott információk alapján elvégezhetővé válnak.

Az ágazaton belül és ágazatok között is megfelelő módszerekkel kell támogatni a kockázatok kezelését. (K-6)

A kiberbiztonsági fenyegetések minden ágazatot érintő időközönként megismételt átfogó kockázatelemzésének támogatására alternatíva lehet a Military Scrum biztonsági követelményhalmazok finomítására szolgáló technikája. A technika alkalmazásakor a kockázatelemzési folyamat végére előállnak a minden ágazatot érintő kockázatkezelési elvárások – a létrejövő dokumentum egy biztonsági termék feladatlista. Az egyes ágazatokon belüli speciális kockázatkezelési eljárások kidolgozásához ez a dokumentum szolgálhat kiindulási alapként, amelynek finomításával az ágazatokon belüli speciális kockázatok feltárása és kezelése is megvalósulhat a technika újbóli alkalmazásával. Az eredmények együttes feldolgozásával ágazatokon átívelő integrált kockázatkezelési koncepció alakítható ki.

A kiberfenyegetésekre történő megfelelő reagálási eljárások kialakításához és támogatásához a kockázatelemzés és a kockázatkezelés számára megfelelő eszközrendszer biztosítása szükséges. (K-7)

Amennyiben a kiberfenyegetésekre történő reagálás informatikai rendszerek – szoftver alapú szolgáltatások – segítségével valósul meg, akkor a kockázatelemzés támogatásán túl a kockázatok kezelését megvalósító informatikai rendszerek előállítása is megvalósítható a Military Scrum szoftverfejlesztési módszertan segítségével. Ebben az esetben a módszertan képezi az eszközrendszert, amely segítségével a kockázatelemzéstől a működő reagálási eljárásokig el lehet jutni.

Gyors helyzetfelismerést, értékelést és kockázatelemzést támogató módszerek kialakítása szükséges. (K-8)

A szándékosan szoftver alapú szolgáltatásokkal szemben indított kibertámadások esetében, amikor nem az infrastruktúra működésképtelenné tétele a cél, nagy jelentőségűvé válik az adott informatikai rendszer önálló védekező képessége. A Military

Scrum a biztonsági követelményrendszer értelmezésével, a speciális biztonsági funkciók meghatározásával, majd azok beépítésével a fejlesztetett szoftverbe lehetővé teszi működő infrastruktúra esetén a kibertámadások gyors felismerését, azok jelzését. Amennyiben már rendelkezésre áll a felismert incidensek kezelésének módja, azok rendszeren belül, automatizáltan elvégezhetők. A tapasztalatok beépítését a biztonsági követelményrendszerbe a későbbi kockázatelemzések során a módszertan lehetővé teszi.

Információbiztonság

A megbízhatóságra tervezési és fejlesztési oldalról is törekedni kell (I-1)

A Military Scrum módszertannal megvalósított szoftverfejlesztési projektek koncepcionális előkészítésekor a kiemelt megbízhatósági követelmények a megrendelő és a terméktulajdonos által beépíthetők a különböző követelményhalmazokba. Azok megvalósulásáról a módszertan további szakaszai gondoskodnak. Lényeges továbbá, hogy a módszertan által biztosított követelményelemzési technikák harmadik lépésében, a sprint tervezések során a termék feladatlista elemeinek elemzése és megvalósíthatóságra alkalmas feladatokká történő szétbontása zajlik. A folyamat során a követelmény jelentésével tisztában lévő – akár keletkezésének történetét is ismerő – szakértő (terméktulajdonos) beszélget egy megfelelő szaktudással rendelkező fejlesztőcsapattal, így a tervezés első lépése indukálja a funkcionális megbízhatóságot.

A fejlesztési feladatokhoz kapcsolódó tesztforgatókönyvek előállítása során a terméktulajdonos áttekinti a fejlesztési feladatokhoz tartozó tesztforgatókönyveket, így a funkciók működésére vonatkozó specifikáció ellenőrzése is megvalósul a megvalósítás előtt, ezzel is fokozva a tervezés oldaláról a megbízhatóságot. A sprint tervezések a tervezési tevékenységek félidejét és egyúttal a fejlesztés megkezdését is jelzik, mert ekkor már a fejlesztéseket megvalósító csapat is elkezd dolgozni a követelmények megvalósításán. A terméktulajdonos által jóváhagyott tesztforgatókönyvek elkészítése a TDD és a DDD technikák alkalmazásával lehetővé teszi, hogy a funkciók megfelelő színvonalon kerüljenek implementálásra. Az automatizált tesztelés és az evolúciós szoftverfejlesztési technika jelentősen fokozza a fejlesztett szoftver funkcióinak megbízhatóságát. A Military Scrum által előírt automatizált telepítési folyamatok révén a manuális beavatkozások számát minimalizáló folyamat segítségével kerülnek éles üzembe a kialakított funkciók – megbízható módon.

Felhő alapú üzemeltetési környezetben is megfelelő megbízhatósági és biztonsági paraméterekkel rendelkező szolgáltatásokat kell kialakítani. (I-2)

A felhőalapú üzemeltetési környezet feltételezi, hogy az adott szoftver alapú szolgáltatás infrastrukturális üzemeltetése harmadik fél által valósul meg. Olyan fejlesztési módszertanra van szükség, ahol a fejlesztett szoftver képességeit egyértelműen el lehet különíteni az üzemeltetési környezettől. Így lehetővé válik az üzemeltetési környezettel és a fejlesztett szoftverrel szemben támasztott megbízhatósági és biztonsági paraméterek szeparált kezelése. A Military Scrum szoftverfejlesztési módszertan elkülöníti a fejlesztési és üzembe helyezési szakaszokat, így lehetővé válik a fejlesztett szoftverrel szemben támasztott megbízhatósági és biztonsági követelmények ellenőrzése az üzembe helyezés előtt. Ha az üzembe helyezés egy felhőszolgáltatásba történő telepítést jelent, akkor az előfeltételek ellenőrzése és a tesztelés ezt követően megvalósítható az üzemeltetési környezet biztonságát és megbízhatóságát illetően.

Magas szintű biztonságot és megbízhatóságot garantáló szabványok kialakítása és alkalmazása a cél. (I-3)

Szoftvertechnológiai oldalról a Military Scrum lehetővé teszi a szoftverek, különös tekintettel a szoftver alapú szolgáltatások magas szintű biztonságot és megbízhatóságot garantáló előállítását. A módszertan alapját képezheti a követelményben szereplő szoftvertechnológiai szabványok kialakításának.

A biztonsági események detektálása, a logikai hibák feltárása, javítása szabványos eljárásokkal és módszerekkel történjen. (I-4)

A Military Scrum alkalmazásával saját vagy harmadik fél által biztosított üzemeltetési környezetben is egyaránt kialakíthatók azok az automatizált rendszerfelügyeleti megoldások, amelyek képesek detektálni a szoftver alapú szolgáltatás vonatkozásában a már ismert, releváns biztonsági eseményeket. A szolgáltatás működésében tapasztalható hibák monitorozása az eljárás segítségével ugyancsak megvalósítható. A begyűjtött információk alapján a következő fejlesztési menetben az igazolt logikai hibák javítása és az azonosított biztonsági események kezelésének képessége szabályozott eljárásokkal megvalósítható a fejlesztett szoftveren belül.

Szabályozott módszerek és folyamatok segítségével történjen a biztonsági tervek előállítása. (I-5)

A Military Scrum fejlesztési meneteiben az első két szakaszban a külön kezelhető a biztonsági követelményrendszer és az azokból levezethető biztonsági feladatok.

1. kockázatelemzés → biztonsági követelményhalmazok meghatározása
2. kockázatkezelés → biztonsági követelmények és reagálás meghatározása (a termék feladatlista részét is képezheti)
3. eszközrendszer → automatizált tesztelés, szakterület specifikus eszközök (a fejlesztett szoftvert érintő incidenskezelési képességek kialakítása)

Megfelelő tájékoztatás mellett kezelhetők, detektálhatók, bejelenthetők legyenek a biztonsági események. (I-6)

Amennyiben széles körben kerülne felhasználásra a Military Scrum szoftverfejlesztési módszertan és az üzemeltetési környezetekkel szemben támasztott követelmények szabványként lennének előírva, akkor kialakítható lenne a következő bejelentési modell. Az egyes incidens típusokhoz tartozó megfelelő egyezményes incidenskódok kialakításával – hasonlóan a HTTP hibakódokhoz –a létfontosságú rendszerek esetében lehetővé válna a központi incidensgyűjtés. Így az alapvető szoftver alapú szolgáltatásokat érintő kibertámadások központi kezelése és bejelentése is lehetségessé válna.

Biztonsági szempontok kezelésére alkalmas, folyamatos működést garantáló tervezési módszerekre van szükség (I-7)

A Military Scrum fejlesztési meneteiben lehetőség van a biztonsági szempontok felülvizsgálatára és azok kiegészítésére a kiberbiztonsági tapasztalatok alapján. Amennyiben egy újabb fenyegetés jelenik meg a kibertérben, akkor a koncepcionális szakaszokban kielemezhető annak szoftveres szinten történő kezelése. Amennyiben a felmerült fenyegetések ellen kialakíthatók a fejlesztett szoftveren belüli reagálási eljárások, akkor azok fejlesztése megvalósítható a fejlesztési szakaszon belül, így garantálva az ismert kockázatok tükrében a folyamatos működési képességet.

A megfelelő reagálási képesség elősegítéséhez olyan rendszerfelügyeleti megoldások kialakítása szükséges, amelyek azonnal jelzik az alapvető szolgáltatásokban bekövetkező változásokat, figyelmeztetnek a potenciális biztonsági eseményekre. (I-8)

A digitális infrastruktúra felügyelete, a különböző kibertámadások elleni védekezés Európai Uniós és kormányzati szinten is megvalósul. Ha egy Military Scrum módszertan segítségével fejlesztett szoftver alapvető szolgáltatásként kerülne besorolásra, akkor az üzemeltetési környezettel és a fejlesztett szoftverrel szemben támasztott rendszerfelügyeleti követelmények integrált kezelése biztosított lenne. A módszertan lehetővé teszi a fejlesztett szoftveren belüli megbízható reagálási képességek kialakítását az evolúciós szoftverfejlesztési technika segítségével. A fejlesztett szoftver által jelzett incidensek becsatornázhatók az üzemeltetési környezet rendszerfelügyeleti szoftverébe, így a szoftveres szinten detektált incidensek is láthatóvá válhatnak az infrastrukturális és hálózati biztonsági események mellett.

Szoftvertechnológia és módszertan

Elengedhetetlen a gyors változáskezelési képesség megléte az informatikai rendszerek teljes életciklusa alatt. (S-1)

Egy változtatási igényről a megfelelő szintű szakértői támogatás segítségével gyorsan eldönthető, hogy a megvalósításához elegendő-e az aktuális rendszerkoncepció vagy esetleg annak bővítése, módosítása szükséges. Utóbbi eset ritkának számít egy üzembe helyezett szoftver alapú szolgáltatás esetén. A Military Scrum helyes alkalmazásakor megvalósul az evolúciós szoftverfejlesztés technikája, így a követelmények változtatásának hatásait a tesztfuttató infrastruktúra rövid időn belül képes visszajelezni. Ezt követően már csak a rendszer konzisztenciájának visszaállítása a feladat az új követelmények tükrében. A változtatások kiadhatók a soron következő szoftvertelepítő csomag előállításakor, majd üzembe helyezhetők a Military Scrum által meghatározott automatizált üzembe helyezési eljárásokkal.

Az új rendszerek fejlesztése során fenntartható, folyamatosan frissíthető műszaki alapok kialakítása szükséges. (S-2)

A Military Scrum helyes alkalmazása a fejlesztett szoftver esetében azt jelenti, hogy a rendszer működése automatizált tesztek formájában is dokumentált. Ez a megközelítés lehetővé teszi, hogy amennyiben a szoftver alapú szolgáltatás valamelyik komponense frissítésre szorul, akkor a frissítés kis kockázattal elvégezhető, mert az evolúciós szoftverfejlesztés technikája lehetővé teszi az új komponens és a rendszer konzisztenciájának kiértékelését. A módszertan által meghatározott üzembe helyezé-

si folyamatok visszajelzést adnak a komponens működéséről az éles üzembe helyezés előtt, de már az üzemeltetési környezeten belül.

Az újonnan kialakított informatikai rendszerek esetén a biztonsági követelmények folyamatos ellenőrzése képezze részét a fejlesztési folyamatnak. (S-3)
Az evolúciós szoftverfejlesztés technikája nem tesz különbséget a műszaki, a funkcionális és a biztonsági követelmények között. Ha Military Scrum követelményhalmazai megfelelően tartalmazzák a fejlesztett szoftverrel szemben támasztott biztonsági elvárásokat, akkor azok kiértékelése folyamatos a rendszer teljes élettartama alatt a fejlesztési szakaszokban.

Kritikus rendszerek esetében a biztonsági követelmények ellenőrzése, a sérülékenységvizsgálat szektor függetlenül képezze részét a rendszerfejlesztéseknek. (S-4)
A követelmény megvalósításához természetesen törvényi szabályozás is szükséges. A Military Scrum lehetőséget biztosít a sérülékenységvizsgálat automatizált beépítésére az evolúciós szoftverfejlesztési technika alkalmazása során. Az üzembe helyezés átadási fázisában a migrációs környezetre történő telepítéskor a tesztek ismételt elvégzésével az üzemeltetési környezetben is meg lehet győződni a helyes konfigurációról és működésről a sérülékenységek vonatkozásában. Így kettős védelemmel rendelkezhetnek a módszertannal fejlesztett szoftver alapú szolgáltatások.

A szolgáltatások minimális erőforrás és idő ráfordításával legyenek átalakíthatók, adaptálhatók a változó felhasználói igényeknek és alkalmazási körülményeknek megfelelően. (M-1)
A módszertani követelmény teljesítése akkor lehetséges, ha az adott szoftver alapú szolgáltatás számára szükséges működést lehetővé tévő rendszerek érettségi szintje eléri a megfelelő szintet – a NATO SLCM Rendszer Koncepciójához igazodva. A megfelelően alkalmazott Military Scrum biztosítja a hatékony feladatvégzést és a gyors átfutási időt a felhasználói igények megvalósításakor. Az üzemeltetési környezetben bekövetkező változásokra való felkészülés gyorsan és hatékonyan valósítható meg a módszertan fejlesztési és üzembe helyezési szakaszai során.

Az infokommunikációs szolgáltatások fejlesztése és biztosítása az alkalmazó szervezetek műveleti követelményei, az infokommunikációért felelős szervezetek szakmai követelményei, a beszerzés műszaki követelményei, a kialakításra kerülő rendszer rendszerterve, az alkalmazó szervezet alkalmazási terve, valamint a tevékenységet szabályozó jogszabályok, okmányok intézkedések alapján valósuljon meg. (M-2)
A Military Scrum követelményrendszer meghatározási módszerei lehetővé teszik a felsorolásban szereplő igények összegyűjtését és felelősökhöz rendelését. Ezt követően a követelményelemzési eljárás első két lépésében a követelményhalmazok és a termék feladatlista előállítása során a különböző szintű szervezeti elemek elvárásai beilleszthetők a követelményrendszerbe. Az iteratív finomítás folyamat segítségével a követelményrendszerben mutatkozó anomáliák feloldhatók. Az átadás előtti fejlesztési sprintek során a funkcionális követelményrendszer, a fejlesztett szoftver dokumentációja és az alkalmazási terv finomíthatók.

Az infokommunikációs szolgáltatások biztosítását, tervezését, fejlesztését, felügyeletét támogató, a szolgáltatás teljes életciklusára értelmezhető szoftverfejlesztési módszertan kialakítása a cél. (M-3)
A NATO SLCM-re épülő szoftver alapú szolgáltatások tervezését, fejlesztését, üzemeltetését támogató Military Scrum módszertan megfelel a kutatási követelménynek.

A fejlesztési javaslatok elemzését, majd szabályozott módon történő változtatási igények meghatározását támogató módszertan kialakítása szükséges. (M-4)
A Military Scrum módszertan minden fejlesztési menet során előírja a változtatási igények elemzését a koncepcionális előtervezés és a koncepcióalkotás szakaszában is. A folyamat szabályozott módon történik az 1. fejlesztési menetben, az átadási előtti időszakban és az átadott rendszer esetében is.

6.3 AJÁNLÁSOK

1. Javasolom a Military Scrum szoftverfejlesztési módszertan beépítését a magyar védelmi ágazat informatikai fejlesztéseket szabályozó dokumentumaiba és folyamataiba, ezzel átláthatóságot és átjárhatóságot teremtve a megrendelői és beszállítói folyamatok között.
2. A Military Scrum által meghatározott fejlesztési és üzemeltetési előírások bevezetését javasolom a létfontosságú infokommunikációs rendszerek szoftvereket érintő fejlesztési és üzemeltetési folyamataiba.
3. További kutatásra ajánlom a speciális szoftvertechnológiai (S) követelmények (hatékonyság, rugalmasság, telepíthetőség, stb.) mérési eljárásainak integrálási lehetőségeit a Military Scrum által alkalmazott evolúciós szoftverfejlesztési környezetbe.
4. Ajánlom minden hierarchikus szervezeti felépítést modellező, párhuzamosan végrehajtott adatkezelési folyamatokat megvalósító informatikai rendszerben az *ÜgyfélHolder*, a *FelelősHolder* és a *Küldemény* tervezési minták alkalmazását, beleértve a kormányzati és nagyvállalati felhasználást is.
5. Szorgalmazom az iratkezelési szoftverek fejlesztésével foglalkozó szakemberek számára, hogy a jogszabályi követelményekben meghatározott fogalmak modellezésekor alkalmazzák az értekezésben meghatározott tervezési mintákat.

TÉMAKÖRBŐL KÉSZÜLT PUBLIKÁCIÓK JEGYZÉKE

[1] GEREVICH J.: Az agilis szoftverfejlesztés alkalmazásának lehetőségei a Magyar Honvédség számára. In: Hadmérnök XII. 1. (2017) 170-181. o.

[2] GEREVICH J.: Híradó-Informatikai fejlesztést támogató agilis dokumentációs módszerek. In: Hadmérnök XII. 3. (2017) 210-222. o.

[3] GEREVICH J., NÉGYESI I.: Híradó-Informatikai fejlesztést támogató agilis dokumentációs módszerek - 2. rész. In: Hadmérnök XIII. 1. (2018) 230-244. o.

[4] GEREVICH J., NÉGYESI I.: A Military Scrum követelményelemző módszerének alkalmazása létfontosságú rendszerek fejlesztése során. In: Hadtudományi Szemle XI. 3. (2018) 293-304. o.

[5] GEREVICH J., NÉGYESI I.: A Military Scrum szoftverfejlesztési módszertan alkalmazása létfontosságú infokommunikációs rendszerek fejlesztése során. In: Hadtudományi Szemle XI. 4. (2018) 72-82. o.

[6] GEREVICH J., NÉGYESI I.: A fenntartható és zavartalan elektronikus ügyintézés szoftver-technológiai háttere - 1. rész In: Hadmérnök XIV. 2. (2019) 281-292. o.

[7] GEREVICH J.: Software Technological Interpretation of the NATO Military Capability Improvement Process. In: AMERICAN JOURNAL OF RESEARCH EDUCATION AND DEVELOPMENT IV. Évfolyam 3. szám (2019) 28-35. o.

[8] GEREVICH J., NÉGYESI I.: A fenntartható és zavartalan elektronikus ügyintézés szoftver-technológiai háttere - 2. rész In: Hadmérnök XIV. 4. (2019) 75-89. o.

[9] GEREVICH J., NÉGYESI I.: Network and Information Security of Cloud Computing Services. In: Hadtudományi Szemle XIII. 1. (2020)

IRODALOMJEGYZÉK

[1] NEUMANN J.: First Draft of a Report on EDVAC. University of Pennsylvania, 1945. 06. 30. https://www.wiley.com/legacy/wileychi/wang_archi/supp/appendix_a.pdf (hozzáférés: 2020. 03. 29.)

[2] MASLOW A. H.: A theory of human motivation - In: Psychological Review. 1943. 50. évf. 4. sz. - p. 370–96. http://psychclassics.yorku.ca/Maslow/motivation.htm (hozzáférés: 2020. 03. 29.)

[3] BUZAN B., WÆVER, O, and WILDE J. D.. Security: A New Framework for Analysis. Boulder, Colo: Lynne Rienner Pub, 1998.

[4] Magyarország Alaptörvénye. Magyar Közlöny 2018. 100. sz. http://www.kozlonyok.hu/nkonline/MKPDF/hiteles/MK18100.pdf (hozzáférés: 2020. 03. 29.)

[5] The 10th Annual State of Agile Report. VersionOne Inc. 2016. https://www.stateofagile.com/#ufh-i-338498988-10th-annual-state-of-agile-report/473508 (hozzáférés: 2019. 12. 15.)

[6] The 13th Annual Statue of Agile Report. StateOfAgile.com. 2019. https://www.stateofagile.com/#ufh-i-521251909-13th-annual-state-of-agile-report/473508 (hozzáférés: 2019. 12. 15.)
[7] RUBIN K. S.: Essential Scrum. Ann Arbor, Michigan, USA, Pearson Education, Inc., 2013.
[8] ANDERSON D. J.: Kanban: Successful Evolutionary Change for Your Technology Business. Sequim, WA, USA, Blue Hole Press. 2010.
[9] LIKER J. K.: The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer. NY, USA, McGraw-Hill. 2004.
[10] Kiáltvány az agilis szoftverfejlesztésért http://agilemanifesto.org/iso/hu/manifesto.html (hozzáférés: 2016. 11. 29.)
[11] Az Agilis Kiáltványt alkotó elvek. http://agilemanifesto.org/iso/hu/principles.html (hozzáférés: 2016. 11. 29.)
[12] Network and Information Security: Proposal for a European Policy Approach; Commission of the European Communities, COM(2001) 298, Brussels, 2001. 06. 06.
[13] MUNK S.: Kiberbiztonsági célok, jövőképek, szabályozók az EU-ban és kapcsolatrendszerük az interoperabilitással. - In: Hadmérnök 2018. 13. évf. KÖFOP különszám. - p. 205-217. http://hadmernok.hu/180kofop_12_munk.pdf (hozzáférés: 2018. 04. 05.)
[14] COMMUNICATION FROM THE COMMISSION on a European Programme for Critical Infrastructure Protection 786 final, COM(2006), Brussels, 2006. 12. 12.
[15] COUNCIL DIRECTIVE 2008/114/EC of 8 December 2008 on the identification and designation of European critical infrastructures and the assessment of the need to improve their protection 2012. évi CLXVI. törvény a létfontosságú rendszerek és létesítmények azonosításáról, kijelöléséről és védelméről
[16] 65/2013. (III. 8.) Korm. rendelet a létfontosságú rendszerek és létesítmények azonosításáról, kijelöléséről és védelméről szóló 2012. évi CLXVI. törvény végrehajtásáról
[17] 359/2015. (XII. 2.) Korm. rendelet a honvédelmi létfontosságú rendszerelemek azonosításáról, kijelöléséről és védelméről
[18] 46/2016. (VIII. 25.) HM utasítás a honvédelmi létfontosságú rendszerelemek azonosításáról, kijelöléséről, ellenőrzéséről, valamint az ezzel összefüggő adatok nyilvántartásáról
[19] BAKOS T.: A HONVÉDELMI LÉTFONTOSSÁGÚ RENDSZERELEMEK AZONOSÍTÁSÁRÓL ÉS KIJELÖLÉSÉRŐL I. In.: Hadmérnök 2017. 12. évf. 1. különszám (július) p. 234-240.
[20] Az elektronikus ügyintézés és a bizalmi szolgáltatások általános szabályairól szóló 2015. évi CCXXII. törvény
[21] COMMUNICATION FROM THE COMMISSION TO THE EUROPEAN PARLIAMENT, THE COUNCIL, THE EUROPEAN ECONOMIC AND SOCIAL COMMITTEE AND THE COMMITTEE OF THE REGIONS on Critical Information Infrastructure Protection 'Achievements and next steps: towards global cyber-security' 163 final, COM(2011), Brussels, 2011. 03. 31.

[22] JOINT COMMUNICATION TO THE EUROPEAN PARLIAMENT, THE COUNCIL, THE EUROPEAN ECONOMIC AND SOCIAL COMMITTEE AND THE COMMITTEE OF THE REGIONS Cybersecurity Strategy of the European Union: An Open, Safe and Secure Cyberspace, Brussels, 2013. 02. 07.
[23] DIRECTIVE (EU) 2016/1148 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 6 July 2016 concerning measures for a high common level of security of network and information systems across the Union, 2016. 06. 06.
[24] DIRECTIVE 2002/21/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 7 March 2002 on a common regulatory framework for electronic communications networks and services (Framework Directive)
[25] REGULATION (EU) No 910/2014 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC, 2014. 07. 23
[26] A Kormány 1163/2020. (IV. 21.) Korm. határozata Magyarország Nemzeti Biztonsági Stratégiájáról
[27] 1035/2012. (II. 21.) Korm. határozat Magyarország Nemzeti Biztonsági Stratégiájáról
[28] NATO Cyber Defence Topic https://www.nato.int/cps/en/natohq/topics_78170.htm (hozzáférés: 2018. 03. 20.)
[29] HOFFMAN F. G.: Conflict in the 21st Century: The Rise of Hybrid Wars. p. 8.
[30] KRASZNAY CS.: A Kiberhadviselés elvei és gyakorlata előadás, 2011, p. 2. http://krasznay.hu/presentation/kiberhadviseles2011_krasznay.pptx (hozzáférés: 2018. 03. 20.)
[31] 1656/2012. (XII. 20.) Korm. határozat Magyarország Nemzeti Katonai stratégiájának elfogadásáról
[32] 1139/2013. (III. 21.) Korm. határozat Magyarország Nemzeti Kiberbiztonsági Stratégiájáról
[33] Defending the networks The NATO Policy on Cyber Defence, 2011. 08. 19. https://www.nato.int/nato_static/assets/pdf/pdf_2011_08/20110819_110819-policy-cyberdefence.pdf (hozzáférés: 2018. 03. 22.)
[34] 2013. évi L. törvény az állami és önkormányzati szervek elektronikus információbiztonságáról
[35] 185/2015. (VII. 13.) Korm. rendelet a kormányzati eseménykezelő központ és az eseménykezelő központok feladat- és hatásköréről, valamint a biztonsági események kezelésének, a biztonsági események műszaki vizsgálatának és a sérülékenységvizsgálat lefolytatásának szabályairól
[36] 1838/2018 Korm. határozat Magyarország hálózati és információs rendszerek biztonságára vonatkozó Stratégiájáról.
[37] A honvédelmi miniszter 58/2014. (IX. 10.) HM utasítása a Magyar Honvédség Informatikai Stratégiájának kiadásáról. – Hivatalos Értesítő, 2014. évi 46. szám 5997-6006 o.
[38] A honvédelmi miniszter 39/2014. (V. 30.) HM utasítása a Magyar Honvédség Informatikai Szabályzatának kiadásáról. – Honvédelmi Közlöny, 2014. évi 7. szám 3614-3660 o.

[39] MUNK S.: Híradó-informatikai szolgáltatások alapjai II. Híradó-informatikai szolgáltatások fogalma, értelmezése. In: Hadmérnök X. 4. (2015) 149-165. o. http://hadmernok.hu/154_14_munks.pdf (letöltve: 2017. 04. 15.)

[40] NÉGYESI I.: A csapatvezetési rendszerek automatizálásának első eredményei az USA fegyveres erőinél. In: HADTUDOMÁNY (ONLINE), XXV. E-szám (2015) 139-151. o. ISSN 1588-060 http://mhtt.eu/hadtudomany/2015/2015_elektronikus/14_NEGYESI_IMRE.pdf (hozzáférés: 2017.04.26.)

[41] NÉGYESI I.: Die überprüfung der voraussetzungen von COTS systemen, Hadmérnök VII. 2. (2012) 371-376. o. ISSN: 1788-1919 http://www.hadmernok.hu/2012_2_negyesi.pdf (hozzáférés: 2017.04.26.)

[42] NÉGYESI I.: COTS rendszerek alkalmazási lehetőségeinek vizsgálata, HADTUDOMÁNYI SZEMLE IV. 4. 111-116. o. http://uni-nke.hu/downloads/kutatas/folyoiratok/hadtudomanyi_szemle /szamok/2011/2011_4/2011_4_tt_negyesi_imre_111_116.pdf (hozzáférés: 2017.04.26.)

[43] Magyar Honvédség Összhaderőnemi Híradó és Informatikai Doktrína. 1. kiadás. – Magyar Honvédség, 2013

[44] PAN J.: Software Reliability. Carnegie Mellon University. Dependable Embedded Systems. 1999 https://users.ece.cmu.edu/~koopman/des_s99/sw_reliability (hozzáférés: 2019. 11. 23)

[45] EMMERICH W., FINKELSTEIN A., MONTANGERO C., STEVENS R.: Standards Compliant Software Development. Presented at: ICSE Workshop on Living with Inconsistency, Boston, USA. 1997 https://discovery.ucl.ac.uk/id/eprint/944/1/11.1_compliance.pdf (hozzáférés: 2020. 07. 25.)

[46] CRAIG D. C.: Compatibility of Software Components — Modelling and Verification. PhD thesis, School of Graduate Studies, Department of Computer Science, Memorial University of Newfoundland. 2007. 05. http://www.cs.mun.ca/~donald/phd/thesis.pdf (hozzáférés: 2020. 07. 27.)

[47] ALMEIDA F., OLIVEIRA J., CRUZ J.: Open Standards and Open Source: Enabling Interoperability, International Journal of Software Engineering & Applications (IJSEA), 2011. 2. évf. 1. szám http://airccse.org/journal/ijsea/papers/0111ijsea01.pdf (hozzáférés: 2019. 12. 20)

[48] SHEN L., REN S.: Analysis and measurement of software flexibility based on flexible points. (2006) https://pdfs.semanticscholar.org/587c/48afa6644316fb744d43d27b19cba4afa34f.pdf (hozzáférés: 2020. 07.28.)

[49] DEVANBU P. T., FONG P. W-L., STUBBLEBINE S. G.: Techniques for Trusted Software Engineering https://www.cs.ucdavis.edu/~devanbu/icse98.pdf (hozzáférés: 2020. 07. 30)

[50] BALDWIN C. Y., CLARK K. B.: Modularity in the Design of Complex Engineering Systems. 9. Fejezet https://www.researchgate.net/profile/Kim_Clark7/publication/225351322_Modularity_in_the_Design_of_Complex_Engineering_Systems/links/0deec53be84be1f88d000000.pdf (hozzáférés: 2020.07.28.)

[51] AHOKAS M., KONTIO J., MÄKELÄ M. M., PÖYRY P., LASSILA A.: Effects of Software Engineering Practices on the Scalability of Firms' Software Development Output. Helsinki University of Technology, Software Business Laboratory http://www.jyrkikontio.fi/attachments/File/publications/Ahokas_Scalability_LBP_CRV.pdf (hozzáférés: 2020.07.30.)

[52] KRASZNAY CS.: Magyar Elektronikus Közigazgatási Alkalmazások Információbiztonsági Megoldásai. Doktori (PhD) értekezés. Zrínyi Miklós Nemzetvédelmi Egyetem Hadtudományi Kar Katonai Műszaki Doktori Iskola. 2012

[53] TOP 40 Static Code Analysis Tools (Best Source Code Analysis Tools) https://www.softwaretestinghelp.com/tools/top-40-static-code-analysis-tools/ (hozzáférés: 2020.07.30.)

[54] Top 15 Code Coverage Tools (For Java, JavaScript, C++, C#, PHP) https://www.softwaretestinghelp.com/code-coverage-tools/ (hozzáférés: 2020.07.30.)

[55] ADDAGARRALA K., KINNICUTT P.: Safety critical software ground rules. In: International Journal of Engineering and Technology 2018. 7. évf. 2. szám pp. 344-350. https://www.sciencepubco.com/index.php/ijet/article/view/13209 (hozzáférés: 2020.08.01.)

[56] ISO 9241-210:2019 Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems https://www.iso.org/standard/77520.html (hozzáférés: 2020.08.02.)

[57] ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models https://www.iso.org/standard/35733.html (hozzáférés: 2020.08.02.)

[58] AVOURIS N.M.: An Introduction to Software Usability, Workshop on Software Usability, Proc. 8th Panhellenic Conference on Informatics, vol 2, pp. 514-522, Nicosia, November 2001, Livanis Publ., Athens, 2001 (invited contribution). https://hci.ece.upatras.gr/wp-content/uploads/publications/2001(C44)An%20Introduction%20to%20Software%20Usability.pdf (hozzáférés: 2020.08.02.)

[59] SAVOLAINEN R.: Information sharing and knowledge sharing as communicative activities. In: Information Research 22. évf, 3. szám. 2017. szept. https://files.eric.ed.gov/fulltext/EJ1156371.pdf (hozzáférés: 2020.08.02)

[60] GARTNER, INC.: Big Data. IT Glossary. https://www.gartner.com/en/information-technology/glossary/big-data (hozzáférés: 2020.08.03.)

[61] MAYER-SCHÖNBERGER, V., CUKIER, K. (2018). E-könyv - Big data - Forradalmi módszer, amely megváltoztatja munkánkat, gondolkodásunkat és egész életünket. ISBN: 9789633047033

[62] SHI P., CUI Y., XU K., ZHANG M., DING L.: (2019). Data Consistency Theory and Case Study for Scientific Big Data. Information.

https://www.researchgate.net/publication/332390857_Data_Consistency_Theory_and_Case_Study_for_Scientific_Big_Data (hozzáférés: 2020.08.03.)

[63] NATO POLICY FOR SYSTEMS LIFE CYCLE MANAGEMENT Note by the Secretary General, 2006. 01. 13., North Atlantic Council, p. 1 http://www.army.cz/assets/files/7284/policy.pdf (hozzáférés: 2019.11.15.)

[64] NATO POLICY FOR SYSTEMS LIFE CYCLE MANAGEMENT, ANNEX 1, C-M(2005)0108, 2006. 01. 13., pp. 1-1–1-3 http://www.army.cz/assets/files/7284/policy.pdf (hozzáférés: 2019.11.15.)

[65] CHINUBHAI A.: Efficiency in Software Development Projects. In: International Journal of Software Engineering and Its Applications Vol. 5 No. 4, October, 2011 pp. 171-179 https://pdfs.semanticscholar.org/1d43/513975d8b5d1af932f98ff983b78cfeaaa10.pdf (hozzáférés: 2019. 11. 20)

[66] RODRÍGUEZA P., HAGHIGHATKHAHA A., LWAKATAREA L. E., TEPPOLAB S., SUOMALAINENB T., ESKELIB J., KARVONENA T., KUVAJAA P., VERNERC J. M., OIVOA M.: Continuous Deployment of Software Intensive Products and Services: A Systematic Mapping Study. http://jultika.oulu.fi/files/nbnfi-fe201902185288.pdf (hozzáférés: 2019. 11. 21)

[67] SHAHROKNI A., FELDT R.: A Systematic Review of Software Robustness. Department of Computer Science & Engineering, 2013, Chalmers University of Technology, Gothenburg, Sweden http://www.robertfeldt.net/publications/shahrokni_2013_sysrev_robustness.pdf (hozzáférés: 2019. 11. 23)

[68] MALHOTRA R., CHUG A: Software Maintainability: Systematic Literature Review and Current Trends. International Journal of Software Engineering and Knowledge Engineering. 26. pp. 1221-1253. (2016) https://www.researchgate.net/publication/309522651_Software_Maintainability_Systematic_Literature_Review_and_Current_Trends (hozzáférés: 2019. 11. 27)

[69] PENZENSTADLER B.: What does Sustainability mean in and for Software Engineering? January 2013 Conference: 1st International Conference on ICT for Sustainability (ICT4S) https://www.researchgate.net/publication/255949741_What_does_Sustainability_mean_in_and_for_Software_Engineering/link/00b49520e7541932dd000000/download (hozzáférés: 2019. 12. 02)

[70] NATO STANDARD AAP-20 Edition C Version 1 2015. 10. NATO STANDARDIZATION OFFICE (NSO) https://nso.nato.int/nso/zPublic/ap/PROM/AAP-20 EDC V1 E.pdf (hozzáférés: 2019.11.15.)

[71] NATO STANDARD AAP-48 Edition B Version 1, 2013. 03. NATO STANDARDIZATION AGENCY (NSA) https://nso.nato.int/nso/zPublic/ap/PROM/AAP-48_B_v1ENG.pdf (hozzáférés: 2019.11.15.)

[72] Strategic Concept for the Defence and Security of the Members of the North Atlantic Treaty Organization Adopted by Heads of State and Government at the NATO Summit in Lisbon 19-

20 November 2010
https://www.nato.int/strategic-concept/pdf/Strat_Concept_web_en.pdf
(hozzáférés: 2019. 11. 16.)
[73] NATO STANDARD AAP-20 Edition C Version 1 ANNEX 3 2015. 10. NATO STANDARDIZATION OFFICE (NSO)
https://nso.nato.int/nso/zPublic/ap/PROM/AAP-20_EDC_V1_E.pdf (hozzáférés: 2019.11.15.)
[74] NATO Policy for Interoperability, (2005). NATO, C-M (2005)0016,
[75] NATO STANDARD AECTP-100 Edition E Version 1, 2016. 12.
https://nso.nato.int/nso/zPublic/ap/PROM/AECTP-100 EDE V1 E.pdf
(hozzáférés: 2019.11.17.)
[76] NATO STANDARD ALP-10 Edition C Version 1 2017. 10.
https://nso.nato.int/nso/zPublic/ap/PROM/ALP-10 EDC V1 E.pdf (hozzáférés: 2019.11.18.)
[77] NATO Policy for Standardization. C-M(2010)0063
[78] INTERNATIONAL STANDARD ISO/IEC 15288 IEEE Std 15288-2008 Systems and software engineering — System life cycle processes, Second edition, 2008-02-01, ISBN 0-7381-5666-3 SS95714
[79] NATO STANDARD AQAP-2210 Edition A Version 2 September 2015
https://nso.nato.int/nso/zPublic/ap/AQAP-2210(A)(2).pdf (hozzáférés: 2019.12.01.)
[80] STANDARDIZATION AGREEMENT STANAG 4107 EDITION 11 2019. 01. 15 NSO
https://nso.nato.int/nso/zPublic/stanags/CURRENT/4107EFed11.pdf
(hozzáférés: 2019.12.03.)
[81] NATO STANDARD AQAP-2070 Edition B Version 4, 2019. 10. NSO
https://nso.nato.int/nso/zPublic/ap/PROM/AQAP-2070 EDB V4 E.pdf
(hozzáférés: 2019.12.05.)
[82] NATO AQAP-2050 Edition 1, 2003. 09. NSA.
https://www.difesa.it/SGD-DNA/Staff/DT/NAVARM/Documents/NormeAQAP/aqap2050e.pdf
(hozzáférés: 2019.12.10.)
[83] NATO STANDARD ARAMP-1 Edition 1 Version 1 FEBRUARY 2012 NSA
https://nso.nato.int/nso/zPublic/ap/ARAMP-1E(1).pdf (hozzáférés: 2019. 12. 12)
[84] NATO STANAG 4427 EDITION 3 2014. 12. 18. NSO
https://nso.nato.int/nso/zPublic/stanags/CURRENT/4427Ed03.pdf (hozzáférés: 2019. 12. 15.)
[85] NATO STANDARD ACMP-2009 Edition A Version 2 2017. 03. NSO
https://nso.nato.int/nso/zPublic/ap/PROM/ACMP-2009 EDA V2 E.pdf
(hozzáférés: 2019. 12. 16.)
[86] NATO STANDARD ACMP-2100 Edition A Version 2 2017. 03. NSO
https://nso.nato.int/nso/zPublic/ap/PROM/ACMP-2100 EDA V2 E.pdf
(hozzáférés: 2019. 12. 16.)
[87] NATO STANDARD AUIDP-1 Edition A Version 1 2016 05 NSO
https://nso.nato.int/nso/zPublic/ap/AUIDP-1 EDA V1 E.pdf (hozzáférés: 2019.12.18.)

[88] NATO STANAG 4704 EDITION 2 2017. 12. 1. NSO https://nso.nato.int/nso/zPublic/stanags/CURRENT/4704EFed02.pdf (hozzáférés: 2019.12.19.)

[89] ISO/IEC 17025 TESTING AND CALIBRATION LABORATORIES https://www.iso.org/ISO-IEC-17025-testing-and-calibration-laboratories.html (hozzáférés: 2019. 12. 20.)

[90] BENEDICENTI L., COTUGNO F., CIANCARINI P., MESSINA A., PEDRYCZ W., SILLITTI A., SUCCI G.: Applying scrum to the army: a case study. 2016 IEEE/ACM 38th IEEE International Conference on Software Engineering Companion pp. 725-727 https://www.researchgate.net/publication/303296580_Applying_scrum_to_the_army_a_case_study (hozzáférés: 2020. 08. 05.)

[91] O'Brian S. P., Green-Mack A. D.: TAKING AGILE ALL THE WAY. In: Army ALT Magazine, Science and Technology. April-June 2018 https://asc.army.mil/web/news-alt-amj18-taking-agile-all-the-way/ (hozzáférés: 2020. 08. 05.)

[92] CASSEL D.: The U.S. Department of Defense on How to Detect 'Agile BS' https://thenewstack.io/the-u-s-department-of-defense-on-how-to-detect-agile-bs/ (hozzáférés: 2020. 08. 06.)

[93] FRIEND T.: Why Military Veterans Make Great Scrum Masters http://blog.eliassen.com/why-military-veterans-make-great-scrum-masters (hozzáférés: 2020. 08. 06.)

[94] ROYCE W. W.: MANAGING THE DEVELOPMENT OF LARGE SOFTWARE SYSTEMS In: in: Technical Papers of Western Electronic Show and Convention (WesCon) August 25–28, 1970, LA, USA. http://www-scf.usc.edu/~csci201/lectures/Lecture11/royce1970.pdf (hozzáférés: 2016. 12. 20)

[95] WEISERT C.: There's no such thing as the Waterfall Approach! (and there never was). Information Disciplines, Inc., Chicago. 2003. 02. http://www.idinews.com/waterfall.html (hozzáférés: 2020. 08. 05)

[96] NET Framework https://www.microsoft.com/net (hozzáférés: 2020. 04. 06.)

[97] Java Enterprise Edition http://www.oracle.com/technetwork/java/javaee/overview/index.html (hozzáférés: 2020.04.06)

[98] What is Enterprise Software? https://www.perlmonks.org/?node_id=504043 (hozzáférés: 2020. 05. 17.)

[99] FÓTHI Á.: Bevezetés a programozáshoz, harmadik, javított kiadás (egyetemi jegyzet), Budapest, ELTE IK, (2012) 30-35. o. http://people.inf.elte.hu/bzsr/progmod2/konyv.pdf (hozzáférés: 2016. 12. 20.)

[100] SOMMERVILLE I.: Software Engineering Tenth Edition. Pearson Education Ltd. Edinburgh Gate, Harlow, Essex CM20 2JE, England (2016)

[101] VIDA K.: Agilis gyakorlati útmutató. Akadémiai Kiadó Zrt. 2018

[102] LARMAN C., BASILI V. R.: Iterative and Incremental Development: A Brief History. In: Computer, 36 (2003. 06) pp. 47-56.

[103] MOORE G. E.: Cramming More Components onto Integrated Circuits. In: Electronics Magazine, 38 (1965. 04. 19) No. 8. pp. 114-117.

[104] NEWELL M. W., GRASHINA M. N.: The Project Management Question and Answer Book. NY, USA, AMACOM, 2003. p. 8.

[105] MARTIN J.: Rapid Application Development. Macmillan. (1991) ISBN 0-02-376775-8

[106] NÉGYESI I.: A csapatvezetési rendszerek automatizálásának első eredményei az USA fegyveres erőinél. In: HADTUDOMÁNY (ONLINE), XXV. E-szám (2015) 139-151. o. ISSN 1588-060 http://mhtt.eu/hadtudomany/2015/2015_elektronikus/14_NEGYESI_IMRE.pdf (hozzáférés: 2017.04.26.)

[107] DB-Engines Ranking https://db-engines.com/en/ranking (hozzáférés: 2020.04.30)

[108] BECK K.: Test-Driven Development by Example. Boston, MA, USA, Pearson Education, Inc., 2003.

[109] MARTIN R. C.: Clean Code A Handbook of Agile Software Craftmanship, Boston, MA, USA, Pearson Education, Inc., 2009.

[110] EVANS E.: Domain-Driven Design: Tackling Complexity inthe Heart of Software. Courier in Westford, MA, USA, Pearson Education, Inc., 2003.

[111] 2011. évi CXII. törvény az információs önrendelkezési jogról és az információszabadságról https://www.adatvedelmirendelet.hu/wp-content/uploads/Infotv.módosítás07.27..pdf (hozzáférés: 2018.12.01.)

[112] AZ EURÓPAI PARLAMENT ÉS A TANÁCS (EU) 2016/679 RENDELETE https://eur-lex.europa.eu/legal-content/HU/TXT/HTML/?uri=CELEX:02016R0679-20160504&from=HUA (hozzáférés: 2018.12.03.)

[113] 2015. évi CCXXII. törvény az elektronikus ügyintézés és a bizalmi szolgáltatások általános szabályairól https://net.jogtar.hu/jogszabaly?docid=A1500222.TV (hozzáférés: 2018.12.03.)

[114] 24/2006. (IV. 29.) BM-IHM-NKÖM együttes rendelet a közfeladatot ellátó szerveknél alkalmazható iratkezelési szoftverekkel szemben támasztott követelményekről.

[115] A Kormány 335/2005. (XII. 29.) Korm. rendelete a közfeladatot ellátó szervek iratkezelésének általános követelményeiről – Magyar Közlöny, 2005. évi 172. szám I. köt. 12408-12419 o. https://magyarkozlony.hu/dokumentumok/d49128c85520fcac3628edf1a23fa62db36f4fe9/megtekintes (hozzáférés: 2019. 01. 03.)

[116] A belügyminiszter 3/2018. (II. 21.) BM rendelete a közfeladatot ellátó szerveknél alkalmazható iratkezelési szoftverekkel szemben támasztott követelményekről – Magyar Közlöny, 2018. évi 23. szám 984-1019 o. https://magyarkozlony.hu/dokumentumok/a2d76c6e46db22a7b33f18c6aa67b0597d7c5001/megtekintes (hozzáférés: 2019. 01. 08.)

[117] 1995. évi LXVI. törvény a köziratokról, a közlevéltárakról és a magánlevéltári anyag védelméről https://net.jogtar.hu/jogszabaly?docid=99500066.TV (hozzáférés: 2019. 01. 08.)

ÁBRÁK JEGYZÉKE

TÁBLÁZATOK JEGYZÉKE

RÖVIDÍTÉSEK JEGYZÉKE

AAP	Allied Administrative Publication
AAPP	Allied Acquisition Practices Publication
ACMP	Acquirers for Life Cycle CM Plan
AECTP	Allied Environmental Conditions and Tests Publication
AKK	Adatmigrálási követelmény kulcsokra
AKO	Adatmigrálási követelmény oszlopokra
AKT	Adatmigrálási követelmény táblákra
ALCCP	NATO Guidance on Life Cycle Costs.
AQAP	Allied Quality Assurance Publications
BM	Belügyminisztérium
CD	Continuous Delivery / Deployment
CERT	Computer Emergency Response Team
COTS	Commercial off-the-shelf
CRUD	CREATE-READ-UPDATE-DELETE
CSIRT	Computer Security Incident Response Team
DDD	Domain-Driven Desing
DDoS	Distributed Denial of Service
DEV	Development
EC	European Committee
ECI	European Critical Infrastructure
EDVAC	Electronic Discrete Variable Automatic Computer
EISAS	European Information Sharing and Alert System
ENISA	European Network and Information Security Agency
EPCIP	European Programme of Critical Infrastructure Protection
EU	Európai Unió
GDPR	General Data Protection Regulation
GovCERT-Hungary	Kormányzati Eseménykezelő Központ
GPS	Global Positioning System
HÁEI BEK	Honvédelmi Ágazati Elektronikus Információbiztonsági Eseménykezelő Központ
HIRbizt	Hálózati és információs rendszerek biztonsága
HW	Hardver
IaaS	Infrastructure as a Service
ICT	Information and Communications Technology
IH IntCERT	Információs Hivatal keretein belül működő incidenskezelés
IK	Integrációs követelmény
ISO	International Organization for Standardization
ISZT	Internet Szolgáltatók Tanácsa
IT	Information Technology
IXP	Internet Exchange Point
KIFÜ	Kormányzati Információs Infrastruktúra Fejlesztési Ügynökség
KKV	Kis- és középvállalkozások
KNBSZ	Katonai Nemzetbiztonsági Szolgálat

LRLIBEK	Létfontosságú Rendszerek és Létesítmények Informatikai Biztonsági Eseménykezelő Központja
MH	Magyar Honvédség
MH HID	Magyar Honvédség Összhaderőnemi Informatikai Doktrína
MK	Migrált követelmény
MTA	Magyar Tudományos Akadémia
NATO	North Atlantic Treaty Organisation
NBSZ	Nemzetbiztonsági Szakszolgálat
NDPP	NATO Defence Planning Process
NEIH	Nemzeti Elektronikus Információbiztonsági Hatóság
NET	Network
NFC	Near-field communication
NIIF	Nemzeti Információs Infrastruktúra Fejlesztés
NIS	Network and Information Security
NKI	Nemzeti Kibervédelmi Intézet
NKKT	Nemzeti Kiberbiztonsági Koordinációs Tanács
NKS	Nemzeti Kiberbiztonsági Stratégia
OECD	Organization for Economic Cooperation and Development
OKF	Országos Katasztrófavédelmi Igazgatóság
Ops	Operations
PA	Projekt azonosító
PaaS	Platform as a Service
PC	Personal computer
PLC	Programmable Logical Controller
PM	Projekt menedzser
PMO	Project management office
PMS	Projekt menedzsment sablon
QA	Quality Assurance
RA	Requirements Analysis
RDBMS	relációs adatbázis-kezelő rendszer
SaaS	Software as a Service
SLCM	Systems Life Cycle Management
SOI	System of Interest
STANAG	Standardisation Agreement
SW	szoftver
SZTAKI	Számítástechnikai és Automatizálási Kutatóintézet
TDD	Test-Driven Development
UK	új követelmény
FKH	Funkcionális követelményhalmaz
MKH	Műszaki követelményhalmaz

1. MELLÉKLET

Projekt menedzsment sablon

A Military Scrum projekt menedzsment dokumentációs sablon célja az egyes életciklus szakaszokon belül elvégzendő különböző eljárások egységbe foglalása, ezzel támogatandó a projektek szabályozott végrehajtását.
A módszertan bináris érettségi szintet definiál: megfelel / nem felel meg.
Ha a fejlesztett szoftver vagy valamelyik működést lehetővé tévő rendszer érettségi szintje nem felel meg a dokumentumban meghatározott elvárásoknak, akkor a projektvezetés feladata a megfelelő lépések megtétele a hiányosságok kiküszöböléséhez.

Az első koncepcionális előtervezést bemutató táblázat bemenete a felismert katonai képesség iránti szükséglet.

Alkalmazott rövidítések: Szerepk. – szerepkör, 1M – 1. fejlesztési menet, ÁE – átadás előtt, ÁU – átadás után.

Ssz.	Szerepk.	Kötelező tevékenység	Eszköz / Módszer / Válasz	1M	ÁE	ÁU
KE-1	PM	Projekthez kapcsolódó szakterületek és szereplők azonosítása / frissítése	9. táblázat (138. o.) (PM, QA, PO, Dev, Ops, Cust)	x	x	x
KE-2	PM	Stratégiai célok dokumentálása	-	x	x	x
KE-3	PM	Szerződéskötés támogatása	Kérdések tisztázása Szoftverekre vonatkozó szellemi tulajdon Elektronikus adatmegosztás Adatbázisok létrehozása és használata	x		
KE-4	PM, PO, Cust	Igények azonosítása és szereplőkhöz rendelése	Lásd: 11. táblázat (149. o.)	x	x	x
KE-5	PM	Military Scrum követelményelemzés támogatására alkalmas feladatkezelő rendszer használatba vétele	11. tábl. - 18. tábl. alapján (149, 151, 153, 157, 161, 162, 163. o.)	x		
KE-6	PM	Egyedi PA és követelményhalmaz jelölésrendszer meghatározása	Minta: PA-FKH-1, PA-MHK-1	x		
KE-7	PM	Projekt típusának meghatározása Új fejlesztés Rendszercsere Adatmigrálás Rendszer integráció	Lásd: 149-150. o. igények elemzése	x		
KE-8	PM, PO, Cust	Követelményhalmazok meghatározása az igények alapján	Lásd: 137-138. o. követelmény halmazok meghatározása	x	x	x
KE-9	PO, Cust	Biztonsági kockázatok elemezése	12. táblázat (151. o.)	x	x	x

KE-10	PM, PO, Cust	Követelmények áttekintése	Lásd: 33. ábra (151. o.)	x	x	x
KE-11	PM, PO, Cust	Egyetértés létrejöttének elérése az illetékes döntéshozók szintjén	11., 24., 12. táblázat (149, 184, 149. o.)	x	x	x
KE-12	PM	Szabálytalanságok kezelése, folyamat javítása	Minőségirányítási nyilvántartás alapján	x	x	x
KE-13	QA	Minőségirányítási nyilvántartás vezetése	Táblázat a feltárt hiányosságokról (Sorszám, hiányosság)	x	x	x
KE-14	QA, PM	Döntéshozatalhoz szükséges szereplők kijelölése megtörtént-e?	IGEN/NEM	x	x	x
KE-15	QA, PM	Szakterületek igényei rögzítésre kerültek-e?	IGEN/NEM	x	x	x
KE-16	QA, PM	Igény formálója jogosult-e szervezeti struktúra alapján?	IGEN/NEM	x	x	x
KE-17	QA, PM	Igény formálója jogosult-e szakterület alapján	IGEN/NEM	x	x	x
KE-18	QA, PM	Megfelelő szintű döntéshozói egyetértés adott-e?	IGEN/NEM	x	x	x
KE-19	QA, PM	Szabálytalanságok történtek-e folyamat során?	IGEN/NEM	x	x	x
KE-20	QA, PM	Projektazonosító egyedisége garantált?	IGEN/NEM	x	x	x
KE-21	QA, PM	Projektvezetés érettsége a szakaszban megfelelő-e?	IGEN, ha a KE-15-KE-21 azonosítójú kérdésekre a válasz IGEN	x	x	x
KE-22	QA, PO	Követelményhalmazok összeállítása / frissítése megtörtént-e?	Igen/Nem	x	x	x
KE-23	QA	Szakértői támogatás érettsége a szakaszban megfelelő-e (PO)?	IGEN, ha a K-22 azonosítójú kérdésre a válasz IGEN	x	x	x

1-1. táblázat Koncepcionális előtervezés tevékenységei

Koncepcionális előtervezés kimenete: Döntéshozók által jóváhagyott, karbantartott, minőségbiztosított követelményhalmazok.

Ssz.	**Szerepk.**	**Kötelező tevékenység**	**Eszköz / Módszer / Válasz**	**1M**	**ÁE**	**ÁU**
KA-1	PM	Egyedi követelmény jelölésrendszer meghatározása a projekt típusa alapján	Lásd: 151. o. Minta: UK, MK, AKT, AKK, AKK, IK	x	x	x
KA-2	PO, Cust	Kezdeti termék feladatlista meghatározása új rendszer fejlesztéséhez	Lásd: 152-153. o.	x		
KA-3	PO, Cust	Kezdeti termék feladatlista meghatározása szoftver cseréjéhez	Lásd: 154-157. o.	x		
KA-4	PO, Cust	Kezdeti termék feladatlista meghatározása adatmigráláshoz	Lásd: 157-161. o.	x		
KA-5	PO, Cust	Kezdeti termék feladatlista meghatározása rendszerek közötti integrációhoz	Lásd: 162. o.	x		
KA-6	PO, Cust	Biztonsági kockázatok kezelése	Lásd: 163-164. o.	x	x	x
KA-7	Ops. Cust	Kezdeti üzemeltetési koncepció kialakítása, ha szükséges	-	x		

KA-8	PM, PO	Prototípusok fejlesztésének kezdeményezése az alternatív megvalósíthatósági tanulmányok kialakításához a kezdeti termék feladatlista és üzemeltetési koncepció alapján, ha szükséges	1. A megoldás és a technikai jellemzők bemutatása 2. Elérhető képességek 3. Időbecslés 4. Szabványosítási, rendszeresítési követelmények	x	x	x
KA-10	Ops	Üzemeltetési koncepció frissítése, ha szükséges	-	x	x	x
KA-11	PO	Termék feladatlista frissítése.	13. táblázat (153. o.)	x	x	x
KA-12	PO, DevOps	Funkcionális áttekintés (Duplikált igények felismerése, nem teljesíthető követelmények azonosítása)	Termék feladatlista áttekintése, Prototípus	x	x	x
KA-13	PM	Szabálytalanságok kezelése, folyamat javítása	Minőségirányítási nyilvántartás	x	x	x
KA-14	QA	Minőségirányítási nyilvántartás vezetése	Táblázat a feltárt hiányosságokról (Sorszám, hiányosság)	x	x	x
KA-15	QA, PM	Előálltak-e a megfelelő kezdeti dokumentumok a projekt jellegétől függően?	IGEN/NEM	x		
KA-16	QA, PM	Szabálytalanságok történtek-e folyamat során?	IGEN/NEM	x	x	x
KA-17	QA, PO	Az előállított prototípus megfelel-e a bemeneti igényeknek?	IGEN/NEM	x	x	x
KA-18	QA, PO	Előállt-e a termékre vonatkozó feladatlista, meghatározásra kerültek-e kapcsolatok a követelményhalmazok és követelmények között?	IGEN/NEM	x		
KA-19	QA, PO	Frissítésre került-e a termékre vonatkozó feladatlista,	IGEN/NEM	x	x	x
KA-20	QA, PO	Frissítésre kerültek-e kapcsolatok a követelményhalmazok és követelmények között?	IGEN/NEM	x	x	x
KA-21	QA. DevOps	Funkcionális áttekintés megvalósult-e?	IGEN/NEM	x	x	x
KA-22	QA	Szakértői támogatás érettsége megfelelő-e (PO) ?	IGEN, ha a KA-17-KA-20 kérdésekre a válasz IGEN	x	x	x
KA-23	QA	Szakértői támogatás érettsége megfelelő-e (DevOps) ?	IGEN, ha a KA-21 kérdésre a válasz IGEN	x	x	x
KA-24	QA	Projektvezetés érettsége a szakaszban megfelelő-e?	a KE-16, KE-20 kérdésekre a válasz IGEN	x	x	x

1-2. táblázat Koncepcióalkotás tevékenységei

Koncepcióalkotás kimenete: A szoftverfejlesztési projekt jellegének megfelelő, karbantartott, minőségbiztosított követelményeket tartalmazó termék feladatlista.

Ssz.	Szerepk.	Kötelező tevékenység	Eszköz / Módszer / Válasz	1M	ÁE	ÁU
F-1	PM	Military Scrum fejlesztés támogatására alkalmas feladatkezelő rendszer használatba vétele	13. táblázat	x		
F-2	PM, PO	Fejlesztési sprintek hosszának, az ütemezésnek a megadása.	Scrum módszertan támogatására alkalmas feladatkezelő rendszer	x	x	x
F-3	PO	Soron következő fejlesztési sprint előkészítése, biztonsági követelmények szem előtt tartása	13. táblázat, illetve bizt. kockázatok. Lásd: 176. o.	x	x	x

F-4	PO, Dev	Sprint tervezések végrehajtása	Lásd: 167-168. o.	x	x	x
F-5	PO, Dev	Tervezési áttekintés	Kitöltött sprintre vonatkozó 13. táblázat alapján	x	x	x
F-6	PM, PO	Sprint bemutatók szervezése	Kitöltött sprintre vonatkozó 13. táblázat és feladatkezelő rendszer alapján	x	x	x
F-7	PM	Retrospektív megbeszélések szervezése	Scrum módszertanban leírtak szerint	x	x	x
F-8	Dev Ops	Evolúciós szoftverfejlesztési környezet kialakítása	Lásd: 174-175. o.	x		
F-9	PO, Dev	Teszt forgatókönyvek elkészítése	Módszerek: TDD, DDD Lásd: 170-173. o.	x	x	x
F-10	Dev	Fejlesztési feladatok elvégzése	Lásd: 174-175. o.	x	x	x
F-11	Dev	Tesztelésre alkalmasság ellenőrzése	TDD, CI	x	x	x
F-12	Dev	Konfigurációs felülvizsgálat	CI	x	x	x
F-13	PO	Megfelelőségi ellenőrzés (manuális tesztelés)	Lásd: 175. o.	x	x	x
F-14	PO, Dev	Termék feladatlista frissítése a tapasztalatok alapján	13. táblázat	x	x	x
F-15	PM	Szabálytalanságok kezelése, folyamat javítása	Minőségirányítási nyilvántartás	x	x	x
F-16	QA	Minőségirányítási nyilvántartás vezetése	Táblázat a feltárt hiányosságokról (Sorszám, hiányosság)	x	x	x
F-17	QA, Dev	Verziószámok egyedisége garantált?	Minta: FOSZAM.ALSZAM Lásd: 177. old.	x	x	x
F-18	QA, PM	Feladatazonosítók egyedisége garantált?	IGEN/NEM	x	x	x
F-19	QA, PM	Sprintek szervezése, megrendelő bevonása a Scrum módszertani előírások szerint valósult meg?	IGEN/NEM	x	x	x
F-20	QA, Dev	Becslések alapján a sprintek végrehajtása sikeres-e?	IGEN/NEM	x	x	x
F-21	QA, Dev	A tervezett funkciók fejlesztése megvalósul-e?	IGEN/NEM	x	x	x
F-22	QA, Dev	A fejlesztőcsapat fókusz faktora eléri-e a 70%-ot?	IGEN/NEM		x	x
F-23	QA, Dev	Konfigurációs mintafájl vezetése megvalósul?	IGEN/NEM	x	x	x
F-24	QA, Dev	A tesztforgatókönyvek alapján készülnek a tesztek, majd a funkciók?	IGEN/NEM	x	x	x
F-25	QA, Dev	A fejlesztőcsapat tagjai egyforma terheltség mellett dolgoznak?	IGEN/NEM	x	x	x
F-26	QA, Dev	Alkalmazva van-e a Military Scrum követelményeknek megfelelő evolúciós szoftverfejlesztési környezet?	IGEN/NEM	x	x	x
F-27	QA, Dev	A teljes evolúciós szoftverfejlesztési környezet visszajelzéseinek figyelése megvalósult-e verziókiadás előtt?	IGEN/NEM	x	x	x
F-28	QA, (Dev)	Az evolúciós szoftverfejlesztési környezet időben, helyes válaszokat ad-e a fejlesztett szoftver pillanatnyi érettségével kapcso-	IGEN/NEM	x	x	x

		latban?				
F-29	QA, (Dev)	A CI ellenőrzi-e a minta konfiguráció meglétét?	IGEN/NEM	x	x	x
F-30	QA, PO	Sprint feladatlista alapján a forgatókönyvek elkészültek-e?	IGEN/NEM	x	x	x
F-31	QA, PO	Változási jegyzék elkészült-e?	IGEN/NEM			x
F-32	QA, PO	A fejlesztés során feltárt további követelmények rögzítésre kerültek-?	IGEN/NEM	x	x	x
F-33	QA, PO	Tervezési dokumentumok frissítésre kerültek-e?	IGEN/NEM	x	x	x
F-34	QA, PO	Azonosított biztonsági kockázatok kezelése beépítésre került a rendszerbe?	IGEN/NEM	x	x	x
F-35	QA	A fejlesztési és tesztelési eszközök érettsége megfelelő-e?	IGEN, ha az F-27-F-28 kérdésekre a válasz IGEN	x	x	x
F-36	QA	Szakértői támogatás érettsége megfelelő-e (Dev)?	IGEN, ha az F-20-F-26 kérdésekre a válasz IGEN	x	x	x
F-37	QA	Szakértői támogatás érettsége megfelelő-e (PO)?	IGEN, ha az F-30-F-34 kérdésekre a válasz IGEN	x	x	x
F-38	QA	Projektvezetés érettsége a szakaszban megfelelő-e?	IGEN, ha az F-18 és F-19 kérdésekre a válasz IGEN	x	x	x

1-3. táblázat Fejlesztési szakasz tevékenységei

Fejlesztési szakasz kimenete: Működő funkciókat tartalmazó minőségbiztosított szoftver telepítőkészlet, változási jegyzékkel, funkciólistával, amely megfelel a követelményrendszernek, az üzemeltetési koncepciónak és a nyomon követési elvárásoknak.

Ssz.	**Szerepk.**	**Kötelező tevékenység**	**Eszköz / Módszer / Válasz**	**1M**	**ÁE**	**ÁU**
Ü-1	PM	Military Scrum hibajegyek kezelésére alkalmas feladatkezelő rendszer használatba vétele	11. táblázat - 18. táblázat (149, 151, 153, 157, 161, 162, 163. o.)	x		
Ü-2	PM	Egyedi hibajegy azonosító képzés kialakítása	23. táblázat (184. old)	x		
Ü-3	PM	Integrált üzemeltetési platform (IÜP) meglétének ellenőrzése	-		x	x
Ü-4	PM	Átadási folyamat előkészítése	Lásd: 179-180. o.		x	x
Ü-5	PM	Verziófrissítési folyamat előkészítése	Lásd: 179-180. o.			x
Ü-6	Ops	Telepítő csomag átvétele	-		x	x
Ü-7	DevOps	Próba telepítés	IÜP Lásd: 182-183. o.		x	x
Ü-8	DevOps	Műszaki ellenőrzés	IÜP		x	x
Ü-9	Ops, Cust	Teszt üzembe helyezés	IÜP		x	x
Ü-10	Cust, PO	Funkcionális tesztelés, átvételi tesztforgatókönyv alapján	22. táblázat (184. old)		x	x
Ü-11	Cust	Helyességi ellenőrzés	22. táblázat (184. old)		x	x
Ü-12	Ops, Cust	Éles üzembe helyezésre alkalmasság ellenőrzése	22. táblázat és IÜP		x	x
Ü-13	Ops	Éles üzembe helyezés	IÜP		x	x
Ü-14	Ops	Műszaki támogatás	IÜP		x	x

Ü-15	Cust, PO	Felhasználói támogatás	Változási jegyzék alapján		x	x
Ü-16	Cust, PO	Változtatási igények rögzítése	24. táblázat (184. old)	x	x	x
Ü-17	Cust, PO	Hibák rögzítése	23. táblázat (184. old)		x	x
Ü-18	Ops	Biztonsági kockázatok kezelése	Lásd: 181-182. o.	x	x	x
Ü-19	PM	Szabálytalanságok kezelése, folyamat javítása	Minőségirányítási nyilvántartás	x	x	x
Ü-20	QA	Minőségirányítási nyilvántartás vezetése	Táblázat a feltárt hiányosságokról (Sorszám, hiányosság)	x	x	x
Ü-21	Ops	A szoftver alapú szolgáltatás szintjén ismert biztonsági események detektálása megtörtént-e?	IGEN/NEM	x	x	x
Ü-22	QA, (Ops)	A telepítések az integrált üzemeltetési platform segítségével valósultak meg?	IGEN/NEM	x	x	x
Ü-23	QA, (Ops)	Az IÜP-ből kinyerhető, hogy ki telepített?	IGEN/NEM		x	x
Ü-24	QA, (Ops)	Az IÜP-ből kinyerhető, hogy mikor telepített?	IGEN/NEM		x	x
Ü-25	QA, (Ops)	Az IÜP-ből kinyerhető, hogy ki konfigurált?	IGEN/NEM		x	x
Ü-26	QA, (Ops)	Az IÜP-ből kinyerhető, hogy mikor konfigurált?	IGEN/NEM		x	x
Ü-27	QA, (Ops)	Az IÜP-ből kinyerhető, hogy melyik verziót konfigurálta?	IGEN/NEM		x	x
Ü-28	QA, (Ops)	Az IÜP-ből kinyerhető-e, hogy a konfigurációban bekövetkező változás?	IGEN/NEM		x	x
Ü-29	QA, (Ops)	Az IÜP-ből kinyerhető, hogy melyik környezetre telepített?	IGEN/NEM		x	x
Ü-30	QA, Dev, Ops	Műszaki ellenőrzés megtörtént-e?	IGEN/NEM		x	x
Ü-31	Ops	Hibák rögzítése megtörtént-e?	IGEN/NEM		x	x
Ü-32	QA, PM, PO	Funkcionális tesztelés megtörtént-e?	IGEN/NEM		x	x
Ü-33	Ops, PO	Változási igények rögzítése megtörtént-e?	IGEN/NEM		x	x
Ü-34	PM	Az átadási vagy a verzióváltási folyamat előkészítése megtörtént-e?	IGEN/NEM		x	x
Ü-35	QA	Üzemeltetési eszközök és környezetek érettségi szintje megfelelő-e?	IGEN, ha az Ü-23-Ü-29 kérdésekre a válasz IGEN		x	x
Ü-36	QA	Technikai támogatás érettségi szintje megfelelő-e?	IGEN, ha az Ü-30, Ü-31 kérdésekre a válasz IGEN	x	x	x
Ü-37	QA	Katonai és fejlesztői szakterületek által nyújtott támogatás érettségi szintje megfelelő-e?	IGEN, ha az Ü-32, Ü-33 kérdésekre a válasz IGEN	x	x	x
Ü-38	QA	Projektvezetés érettsége a szakaszban megfelelő-e?	IGEN, ha az Ü-32 és Ü-34 kérdésekre a válasz IGEN	x	x	x

1-4. táblázat Üzembe helyezés és támogatás tevékenységei

Az üzembe helyezési szakasz kimenete: Szabályos körülmények között üzemeltetett és támogatott szoftver alapú szolgáltatás.

2. MELLÉKLET

Speciális kutatási követelmények való megfelelés igazolása

A megfizethetőség, az idő, az ütemezés, a minőség és a kockázati tényezők együttes értelmezése szükséges a katonai célú informatikai rendszerek fejlesztése során. (M-5)
A Military Scrum szoftverfejlesztési módszertan a projekt menedzsment háromszögön belüli elhelyezkedését tekintve a Scrum-al mutat analógiát, így a minőséget és a kockázatok minimalizálását helyezi előtérbe.
A további tényezők közötti összefüggések a fejlesztési sprintek végrehajtásával kapcsolatosak. A fogalmak megfelelő értelmezéséhez szükséges az *ütemezés* fogalmának tisztázása. Jelen esetben egy kiadható, a fejlesztett funkciók vonatkozásában működőképes szoftververzió előállítása egy meghatározott időegység alatt az ütemezés – ez egy ideális sprint időtartamát jelenti. Egy példával elősegítendő a megértést: ha a fejlesztési határidő 2 hónap múlva esedékes, akkor 2 hetes ütemezés mellett 4 fejlesztési sprint alatt lehet elvégezni a fejlesztési feladatokat.

- Költségek – a szakaszok végrehajtásához szükséges kapacitások (tervezői, fejlesztői képességek) hangolásával vagy a határidőkkel szabályozhatók. A kapacitások túlzott csökkentése a sprintek elhúzódását eredményezheti, így ez károsan hat az ütemezési szempontokra. A határidők túlzott kitolása, ugyancsak ütemezési problémákhoz vezethet. A Scrum-mal mutatott hasonlóság miatt a kapacitások túlzott növelése jelentős költségtöbbletet eredményezhet, ugyanakkor az iterációk ideális átfutási idejét különösebben nem tudja növelni, így az ütemezést sem befolyásolja egy Military Scrum-ot alkalmazó szoftverfejlesztési projekten belül. Ebben az esetben további szeparáltan működő fejlesztési projektek indítása javasolt.
- Idő – egy sprint átfutási ideje a szakaszokon belül elvégzendő feladatok mennyiségével vagy a rendelkezésre álló kapacitás változtatásával szabályozható bizonyos határokon belül. Az implementálandó feladatok számának módosításával szabályozható egy iteráció végrehajtási ideje a költségek szinten tartása mellett. Ebben az esetben is van minimum átfutási idő, ami tovább nem csökkenthető. A másik véglet a tervezői, fejlesztői kapacitások túlterhelése, a túl nagy mennyiségű feladat boríthatja az ütemezést. Célszerű az ütemezést tartva, a feladatok

számát a kapacitásokhoz igazítani. A határidők tartása a megfelelően felosztott feladatrendszer segítségével garantálható. A megnövekedett megrendelői igények szeparáltan működő Military Scrum projektek segítségével kezelhetők fenntartható módon, ilyenkor a külön fejlesztett szoftverek integrációs projektjeit is részévé kell tenni fejlesztésnek.

Integrált és zökkenőmentes üzletkötési, projektmenedzsment gyakorlatok kialakítása szükséges a koncepció-alkotástól a kivezetésig. (M-6)

A Military Scrum az 0. fejlesztési sprintben ajánlást teszt a prototípusok előállítására. A prototípusok pályáztatása és kiértékelése egy hosszú távon fenntartandó szoftver alapú szolgáltatás költségeihez viszonyítva elenyésző. Ha a megrendelői oldal a fejlesztett szoftver korlátozott funkciókkal rendelkező prototípusainak előállításába hajlandó invesztálni, akkor a legjobb mutatókkal rendelkező prototípus továbbfejlesztése alapot adhat a folyamatok integrált folytatásához. A módszer lehetőséget teremt a határidők és költségek összefüggéseinek megértésére, így lehetővé teszi a zökkenőmentes üzletkötési folyamatokat a megrendelő és a beszállító között. A szerződéskötés támogatásán túlmenően a PMS utasításainak követésével a projektek teljes élettartamuk alatt integrált projektmenedzsment szisztémával támogathatók.

Teljes életciklusra vonatkozó, minden érdekelt fél között hatékony együttműködés kialakítása a cél, jól definiált felelősségi körökkel. (M-7)

A PMS a projekt legelején a projektmenedzsment tevékenységek részeként megköveteli a résztvevő személyek és szerepük rögzítését a Military Scrum projekten belül. A dokumentum minden életciklus szakasz során meghatározza, hogy az egyes szerepkörök számára milyen tevékenységek elvégzése kötelező, illetve milyen hatáskörrel rendelkeznek a projekten belül. Bármely érintett – Cust, PM, PO, Dev, Ops, QA – változása esetén a változás tényét rögzíteni szükséges. Ezáltal a szerepek és az elvégzendő feladatok folyamatosan egyértelműek, így a rendszer teljes élettartama alatt garantálható a hatékony együttműködés.

A kialakítandó szoftverfejlesztési módszer értelmezze az érettségi szint fogalmát és feleljen meg a NATO Rendszer Koncepciónak. (M-8)

A Military Scrum alkalmazásakor a fejlesztett szoftver és a működést lehetővé tévő rendszerek együttesen megfeleltethetők egy szoftver intenzív NATO SLCM SOI-

nak. Ekkor a katonai képességet reprezentáló rendszer egy szoftver alapú szolgáltatás, amely négy összetevőből áll: fejlesztett szoftver, projektvezetés, szakértői támogatás és támogató rendszerek. Mind a négy összetevő esetén a Military Scrum módszertan értelmezi a bináris érettségi szint (megfelel/ nem felel meg) fogalmát és csak akkor alkalmazható megszakítás nélkül, ha minden komponens megfelel a PMS-ben szakaszonként előírt érettségi szintnek. Ezáltal a módszertan következetes alkalmazása kielégíti a NATO Rendszer Koncepció követelményeit.

A kialakítandó szoftverfejlesztési módszer adjon eszközöket a stratégiai célok dokumentálására, a szereplők azonosítására, a követelmények meghatározására és priorizálására, támogassa a változáskezelést, valamint folyamatosan adjon megfelelő visszajelzést a rendszer érettségével kapcsolatban. (M-9)

- *Stratégiai célok dokumentálása* – a koncepcionális előtervezés szakaszában a PMS előírja a tevékenység elvégzését.
- *Szereplők azonosítása* – a koncepcionális előtervezés szakaszában a PMS előírja a tevékenység elvégzését.
- *Folyamatos visszajelzés a rendszer érettségével kapcsolatban* – a PMS-ben előírt minőségbiztosítási tevékenységek helyes alkalmazásával a fejlesztett szoftver alapú szolgáltatás komponenseinek érettségi szintje ellenőrzött és felügyelt. Az érettségi szinttel kapcsolatos követelmények sérüléséről a projektvezetés a minőségbiztosítón keresztül azonnal értesül – ha szükséges, akkor azonnal megszakítja az adott szakasz végrehajtását – és megkezdi a műszaki vagy a személyzeti működéshez szükséges előfeltételek újbóli megteremtését.
- *Követelmények meghatározása és priorizálása* – a Military Scrum négyszintű specifikációs módszert definiál a fejlesztési feladatok előkészítéséhez. Az 1. fejlesztési menet során a követelményhalmazok és a priorizált termék feladatlista meghatározása megtörténik a koncepcionális előtervezés és a koncepcióalkotás szakaszaiban. A módszertan biztosítja a megfelelő dokumentációs sablonokat és módszereket a tevékenység elvégzéséhez a fejlesztett szoftver teljes élettartama alatt.
- *Változáskezelés* – a Military Scrum által meghatározott dokumentációs infrastruktúra levetővé teszi az új igények beillesztését a követelményrendszerbe ezzel támogatva a változáskezelést. A későbbi fejlesztési menetekben a köve-

telményrendszer frissítését követően a változások kezelése megegyezik az egyéb funkciók fejlesztési és üzembe helyezési folyamataival.

Az alternatív megvalósíthatósági tanulmányok kialakítását folyamat és dokumentációs szinten is támogassa a kialakítandó szoftverfejlesztési módszertan. (M-10)
A termék feladatlista első verziójának előállítását követően lehetőség nyílik az alternatív megvalósíthatósági tanulmányok kialakítására, szoftver prototípusok formájában. A Military Scrum koncepcióalkotási folyamata lehetővé teszi, hogy a kialakítandó képességek szempontjából lényeges feladatok megvalósíthatósága önállóan is vizsgálható legyen a fejlesztési szakasz előtt. A prototípusok előállítása során keletkezett dokumentumok beépítése a kezdeti termék feladatlistába részét képezi a 0. menet projekt menedzsment tevékenységeinek a PMS-ben.

A kutatás céljaként kitűzött kialakítandó szoftverfejlesztési módszer folyamataiban jelenjenek meg a felsorolásban szereplő tevékenységek: követelmények áttekintése, funkcionális áttekintés, tervezési áttekintés, tesztelésre alkalmasság ellenőrzése, konfigurációs felülvizsgálat, megfelelőségi ellenőrzés, helyességi ellenőrzés, éles üzembe helyezésre alkalmasság ellenőrzése. (M-11)

- *Követelmények áttekintése* – a koncepcionális előtervezés szakaszában végrehajtandó tevékenység a Military Scrum szoftverfejlesztési módszertan meghatározása szerint.
- *Funkcionális áttekintés* – a koncepcióalkotás szakaszában végrehajtandó tevékenység a Military Scrum módszertan meghatározása szerint.
- *Tervezési áttekintés* – a fejlesztési sprintek tervezése során végrehajtott tevékenység a Military Scrum módszertan meghatározása szerint.
- *Tesztelésre alkalmasság ellenőrzése* – a CI által megvalósított gépi tevékenység, a fejlesztett szoftver forráskódjának és műszaki paramétereinek változása esetén végrehajtandó tevékenység a módszertan meghatározása szerint.
- *Konfigurációs felülvizsgálat* – analóg módon az előző ponttal.
- *Megfelelőségi ellenőrzés* – az egyes fejlesztési feladatok megvalósítását követően az elkészült funkció automatizált és manuális tesztelése a Military Scrum szoftverfejlesztési módszertan meghatározása szerint.

- *Helyességi ellenőrzés* – az integrált üzemeltetési platform segítségével kitelepített, teszt környezeten végzett funkcionális tesztelés a Military Scrum szoftverfejlesztési módszertan meghatározása szerint.
- *Éles üzembe helyezésre alkalmasság ellenőrzése* – az integrált üzemeltetési platform segítségével sikeresen telepített környezeten a műszaki és a helyességi ellenőrzés elvégzése a Military Scrum szoftverfejlesztési módszertan meghatározása szerint.

A kialakítandó technikáknak támogatniuk kell a döntéshozási mechanizmusokat, hogy az érdekeltek kellő mennyiségű és minőségű információval rendelkezzenek az elengedhetetlen közös álláspont kialakításához. (M-12)

A koncepcionális szakaszokban a követelményrendszerben mutatkozó esetleges anomáliák feloldásához az igények és a megrendelői oldal szakterületeinek összekapcsolása ad támpontot módszertani oldalról. A feltárt ellentmondások feloldása a követelményrendszeren belül ezt követően megrendelői feladat. Műszaki kérdésekben a döntéshozatalt a prototípusok előállítása segítheti elő. Az előállított prototípusok alapján kiértékelhető egy követelményhalmaz vagy a termék feladatlista egy elemének megalapozottsága – a technológia és más tényezők oldaláról is. Az egyeztetések, kutatások végeztével koncepció alkotási szakasz végére a módszerek helyes alkalmazásával előállhat a kellő mennyiségű és minőségű információ a közös álláspont kialakításához.

A fejlesztési szakaszban a Military Scrum által megvalósított evolúciós szoftverfejlesztési technika segítségével a fejlesztési szakaszon belül az érintettek (PO, Dev) a tervezési és fejlesztési feladataik során megfelelő támogatásnak kapnak a rendszer működését érintő kérdésekben. Amennyiben a döntéshozatalhoz további szereplők bevonása is szükséges, akkor a rendelkezésre álló információk alapján a koncepcionális tervezés szintjére is eszkalálhatók a kérdések, ahol a Military Scrum segítségével az igényekig is vissza lehet menni az ellentmondások feloldásának érdekében.

Az üzembe helyezés szakaszában az érintettek számára (DevOps, Cust) az automatizált átadási folyamatok egyértelmű döntési helyzeteket teremtenek az esetlegesen felmerülő hibák esetén. Helyes konfigurációs beállítások mellett a sikertelen telepítés a telepítőkészletek problémájára utalhat. Sikeres telepítés és helytelen működés esetén a fejlesztett szoftver hibája valószínű. Konfigurációs módosításokat követő hatékonysági problémák esetében a konfigurációs beállítások okolhatók. Sikeres műszaki

ellenőrzést és funkcionális tesztelést követően egy éles üzembe helyezésre vonatkozó döntés megalapozottá válik. A felhasználás (PO, DevOps) és a támogatás (DevOps) szakaszában az automatizált rendszerdiagnosztikai eszközök segítségével felderíthetők a rendszerben mutatkozó hibajelenségek és azok javítására vonatkozó követelmény kidolgozása a kinyert adatok alapján megkezdhető. A Military Scrum a fejlesztést követő szakaszok számára is megfelelő döntéshozatalt támogató technikákat biztosít az érintettek részére.

A kialakítandó szoftverfejlesztési módszer támogassa a módosítási eljárás koncepció alkotási szakaszában elvégzendő szoftvertechnológiai mozzanatokat: duplikált igények felismerése, nem teljesíthető követelmények azonosítása, kockázatkezelés.
(M-13)

- *Duplikált igények felismerése* – a koncepcionális előtervezés során az új igények elemzésekor – megfelelő szakértők jelenléte mellett – a duplikátumok azonosíthatók a Military Scrum helyes alkalmazásával.
- *Nem teljesíthető követelmények azonosítása* – a megvalósíthatóság szemszögéből kérdéses igények és megfogalmazott követelmények feltérképezésére nyújt lehetőséget a koncepcióalkotás szakaszában a prototípus előállítási folyamat. Ha a prototípus előállítása komoly akadályokba ütközik vagy egyenesen sikertelen, akkor nem teljesíthető követelményekről beszélünk. Természetesen a kutatási céllal indított prototípus előállítását megfelelő szaktudással rendelkező fejlesztőcsapatnak kell elvégeznie.
- *Kockázatkezelés* – a módszertan előírja a biztonsági kockázatok követelményhalmazokba szervezését, majd az azonosított követelmények és reagáláshoz szükséges funkciók felvételét a termék feladatlistába. A szoftveres kockázatok kezelése ezt követően az egyéb szoftverfejlesztési feladatok megvalósításával együtt történik, amelyhez a módszertan biztosítja a megfelelő eszközöket.

A kialakítandó szoftverfejlesztési módszer támogassa a technológiai lehetőségek kiértékelését, a konfiguráció és rendszer kapcsolatának vizsgálatát, a módosítás hatásának vizsgálatát a rendszer komplexitására, az átvételi követelmények finomítását, a költségbecslést, a megkötések azonosítását, a kockázatértékelést, a megkerülő meg-

oldások biztosítását, a további érettséget előirányzó módosítások meghatározását. (M-14)

- *Technológia lehetőségek kiértékelése* – a prototípus előállítási projektek segítségével támogatható a technológiai lehetőségek kiértékelése.
- *Költségbecslés* – a részletesen kidolgozott termék feladatlista kivonata és a dokumentum alapján elkészített prototípusok előállítási költségei kiindulási alapként szolgálnak a teljes projektre vonatkozó a költségbecsléshez.
- *Kockázatelemzés* – az első fejlesztési menetben és azt követően is a Military Scrum koncepcionális előtervezési szakaszában a biztonsági kockázatok elemzése kötelezően végrehajtandó tevékenység.
- *Megkerülő megoldások biztosítása* – a koncepcióalkotás során lehetőség van a funkcionális megkerülő megoldások beépítésére a termék feladatlistába. A prototípus előállítás révén a megkerülő technikai megoldások biztosítása lehetségessé válik.
- *További érettséget előirányzó módosítások* – a tapasztalatok alapján az adott szoftverfejlesztési projekt érettségi szintekkel szemben támasztott szabályrendszere kibővíthető a PMS bővítésével.
- *Konfiguráció és rendszer kapcsolatának vizsgálata* – az evolúciós szoftverfejlesztési technika lehetővé teszi a fejlesztett szoftver különböző konfigurációk melletti viselkedésének elemzését.
- *Módosítás hatásának vizsgálata a rendszer komplexitására* – a CI technika alkalmazásakor az architekturális és domain szintű módosítások hatásainak vizsgálata szükségszerű, a módosítások hatásáról gyors visszajelzést kapnak a fejlesztők.
- *Megkötések azonosítása* – a CI technika mellett kialakíthatók további speciális automata tesztek, amelyek csak bizonyos időközönként vizsgálják a rendszer képességeit. Például: performancia tesztek segítségével behatárolható az adott rendszer terhelhetősége.
- *Átvételi követelmények finomítása* – a funkcionális tesztelés és a műszaki ellenőrzés fázisaiban az esetlegesen felmerülő hibák esetében lehetőség van az átvételi követelmények finomítására, ha a fejlesztett szoftver kiadott, átadásra szánt verziója áll a feltárt hibajelenség mögött, ebben az esetben a fejlesztőcsapat (Dev) a támogatási szakasz terhére kötelező kijavítani a hiba okát.

A kialakítandó szoftverfejlesztési módszer támogassa az előzetes tervezést és elemzést, a prototípuskészítést, a bemutatókat, a laboratóriumi teszteket, az optimális technológiák meghatározását a hatékonyság, a költségek, az idő és a kockázatok vonatkozásában, megkerülő megoldások meghatározását, a követelményrendszer véglegesítését, további érettséget növelő tevékenységek meghatározását. (M-15)

- *Előzetes tervezés és elemzés* – a koncepcionális szakaszokban a követelményelemzés, míg a fejlesztés szakaszában a sprint tervezés tekinthető az implementációt megelőző előzetes tervezésnek és elemzésnek.
- *Prototípuskészítés* – a prototípus előállítása néhány Military Scrum fejlesztési menet segítségével megvalósítható, a fejlesztési szakaszokban elvégzendő fejlesztési feladatok mennyisége szabályozható.
- *Bemutatók* – a fejlesztési szakaszon belül a fejlesztési sprintek végén lehetséges a bemutatók megszervezése.
- *Laboratóriumi tesztek* – az evolúciós szoftverfejlesztés technikája a szoftverek esetében egyfajta laboratóriumi tesztelésként is felfogható.
- *Optimális technológiák meghatározása* – a módszertan által javasolt prototípus előállítás során cél lehet a különböző technológiai alternatívák kipróbálása is. A funkciók elkészítési ideje, a minőség, a megoldások összetettsége alapján vizsgálhatóvá válnak a következő tényezők közötti összefüggések: hatékonyság, költségek, idő, kockázatok.
- *Követelményrendszer véglegesítése* – a módszertan által javasolt prototípus előállítását követően a koncepcionális szakaszokban lefektetett követelményrendszer pontosítható, az erőforrás becslések finomíthatók, a kiindulási követelményrendszer véglegesíthető.

A kialakítandó szoftverfejlesztési módszer támogassa a fejlesztést, a tesztelést, az értékelést, a helyesség és a megfelelés ellenőrzését, illetve a követelmények és a specifikáció finomítását, valamint a rendszer verziók korlátlan előállítását. (M-16)

A Military Scrum az evolúciós szoftverfejlesztési technikájának segítségével integráltan támogatja a fejlesztést, tesztelést és az értékelést. A tesztfuttató infrastruktúra visszajelzései alapján a követelmények és a specifikáció finomítható. A fejlesztési sprintekhez tartozó bemutatók segítségével a megrendelő rövid időszakonként láthatja a fejlesztett szoftver alakulását. A verziók a fejlesztési szakaszok végén kiadhatók.

A kialakítandó szoftverfejlesztési módszer támogassa a támogató dokumentációk előállítását, a konfigurációkban bekövetkező változtatások követését, az üzemeltetési környezet és támogató rendszerek felkészítését és az átvételi teszt elvégzését. (M-17)

- Támogató dokumentációk előállítása – az integrált üzemeltetési platformból a telepítésekre, telepítő csomagokra, konfigurációkra, változási jegyzékekre, követelményazonosítókra vonatkozó információk kinyerhetők, ezzel támogatva a további dokumentumok előállítását.
- Konfigurációkban bekövetkező változások követése – ha kizárólag a Military Scrum által előírt integrált üzemeltetési platform segítségével történik a konfiguráció, akkor a változások az adott környezet tekintetében kinyerhetők a rendszerből.
- Üzemeltetési környezet felkészítése – az átadási folyamat részeként végrehajtott próbatelepítés elsődleges célja a műszaki ellenőrzés előkészítése, ugyanakkor a tapasztalatok alapján az üzemeltetési környezet felkészítése és támogatottá válik.
- Támogató rendszerek felkészítése – a műszaki ellenőrzés során az éles üzemű működéshez szükséges támogató rendszerek ellenőrzése és a tapasztalatok alapján azok felkészítése támogatott a módszertan által.
- Átvételi teszt elvégzése – a Military Scrum által előírt funkcionális tesztelés feleltethető meg az átvételi teszt elvégzésének, ha annak kimenete a fejlesztett szoftver telepítőkészletének átadás-átvétele.

A kialakítandó folyamatok a koncepcionális előtervezés szakaszával kezdve korlátlanul legyenek megismételhetők. (M-18)

A Military Scrum 20. ábrán látható iteratív folyamata a módszertan meghatározásából kifolyólag korlátlanul megismételhető az adott szoftver alapú szolgáltatás teljes élettartama alatt. A módszertan részletesen tárgyalja az első sprint, az üzembe helyezési előtti és az üzembe helyezett szolgáltatások fejlesztési sprintjeinek teendőit is. Az egyes szakaszok a feladatok függvényében más és más súllyal vannak jelen a sprintekben, azonban a projekt menedzsment dokumentációs sablon részletesen kitér a különböző jellegű sprintek feladatrendszerére. Ennek eredménye, hogy az iteráció minden esetben elvégezhető, így a Military Scrum megfelel a követelménynek.

A kialakítandó szoftverfejlesztési módszertan támogassa a tárolt adatok migrálhatóságát, a funkciók követelményeinek kinyerését, illetve az üzemeltetési tapasztalatok kinyerését egy szoftver alapú szolgáltatás megszűntetésekor. (M-19)

- *Tárolt adatok migrálhatósága* – a Military Scrum módszertan adatmigrálást támogató követelményelemzési módszerével megfelel a kutatási elvárásnak.
- *Funkciók követelményeinek kinyerése* – a Military Scrum módszertan a szoftver cseréje új szoftver fejlesztésével követelményelemzési módszerével megfelel a kutatási elvárásnak.
- Üzemeltetési tapasztalatok kinyerése – a Military Scrum szoftverfejlesztési módszertan által előírt integrált üzemeltetési platformnak képesnek kell lenni a lényeges telepítési és az üzemeltetési adatokat gyűjtésére, így ezek kinyerhetők az üzemeltetés támogatási rendszerből.

A kialakítandó szoftverfejlesztési módszertannak összhangban kell lennie az AQAP 2210 szabványban foglaltakkal. (M-20)

A Military Scrum szoftverfejlesztési módszertan életciklus modellje a NATO SLCM szoftver alapú szolgáltatásokra értelmezett vetületeként került kialakításra. Az AQAP 2210 szabvány részletes tárgyalása ugyan meghaladja az értekezés kereteit, az alábbiakban a szabvány által felügyelt főbb terület és a Military Scrum fogalmainak kapcsolata szerepel.

A Military Scrum által tárgyalt AQAP 2210 fejezetek:

2.2.2 A szoftver projektre vonatkozó minőségirányítási terv – szakaszonkénti lebontásban a PMS részét képezi.

2.2.3 A szoftverrel szemben támasztott követelmények azonosítása és áttekintése – a módszertan koncepcionális szakaszaiban végrehajtott tevékenység során valósul meg.

2.2.4 *Menedzsment* – a Military Scrum projektirányítási feladatokat a PMS írja le.

2.2.4.1 *Szoftverfejlesztési folyamat* – a Military Scrum fejlesztési szakaszában a Scrum szoftverfejlesztési módszertan kiterjesztetett változata a követelmény.

2.2.4.2 *Szervezet* –a projektben résztvevő szereplők megadása és frissítése menet közben kötelező a PMS-ben.

2.2.4.3 *Nem megfelelő szoftver* – a fejlesztési szakaszban a CI és a kapcsolódó technikák segítik a hibák feltárását és javítását. A későbbi szakaszokban az integrált

üzemeltetési platform segítségével detektálhatók a nem megfelelő szoftverek és szoftverkomponensek, a hibajelenségek elhárítása megkezdhető.

2.2.4.4 *Korrekciós tevékenység* – a nem megfelelő működés, a feltárt hibák javítására a Military Scrum változáskezelési folyamata alkalmazható.

2.2.4.6 *Szoftver konfiguráció menedzsment* – a fejlesztési szakaszban a módszertan gondoskodik a konfigurációs lehetőségek dokumentálásáról az előírt példa konfigurációs fájlok megadásával. Az üzembe helyezési és felhasználási szakaszokban a felhasználói és a műszaki paraméterek változtatásának dokumentálásáról az integrált üzemeltetési platform gondoskodik.

2.2.4.9 *Minőségirányítási nyilvántartás* – a Military Scrum szakaszai során a dokumentum kötelezően vezetendő a minőségbiztosító által.

2.2.4.10 *Dokumentáció* – a Military Scrum elsődleges dokumentuma a projekt menedzsment sablon (PMS), amely meghatározza, hogy a különböző jellegű projektekben milyen típusú táblázatokat kell használni a követelményelemzéstől az üzembe helyezésig.

2.2.4.11 *Szoftverhez tartozó média kezelése és tárolása* – a Military Scrum előírja a központi verziókövető rendszer meglétét és használatát a fejlesztett szoftver számára.

2.2.4.12 *Replikáció és szállítás* – a Military Scrum előírja a CI visszajelzéseire épülő automatizált verziókiadás képesség meglétét, amelyet követően az üzemeltetési szakaszban az integrált üzemeltetési platform segítségével megvalósíthatók az automatizált telepítések is.

2.2.5 *Szoftverfejlesztés* – a Military Scrum helyes alkalmazása során megvalósul az evolúciós szoftverfejlesztés technikája, amely megfelelő támogatást és mérőszámokat biztosít a fejlesztők és a minőségbiztosítás számára.

2.2.6 *Kiértékelés, megfelelés és helyesség vizsgálat* – Az evolúciós szoftverfejlesztési technika segítségével a fejlesztett szoftverben megvalósított folyamatok és rendszert alkotó műszaki komponensek együttes és folyamatos értékelése és megfelelőségi vizsgálata megvalósítható. Az automata tesztek által produkált mérőszámok metrikaként szolgálnak a fejlesztett szoftver pillanatnyi minőségét illetően. A helyesség vizsgálat a manuális tesztelés segítségével valósul meg fejlesztői oldalon.

2.2.6.1 *Tesztelés* – a Military Scrum egyaránt előírja a fejlesztési szakaszban az automatizált és a manuális tesztelést. A módszertan megköveteli az átadási folyamat megkezdése előtt a rendszer teljes működésének, különös tekintettel a fejlesztési sza-

kasz feladatainak újra tesztelését. A feltárt hibák javítása kötelező a verziókiadás előtt.

2.2.6.2 *Áttekintés* – a fejlesztési szakaszon belül, minden fejlesztési sprint tervezésekor megvalósul a fejlesztendő feladatok funkcionális áttekintése, ekkor a követelmények és a tervezett megvalósítás párba állítva, egymás mellett szerepel. A fejlesztési sprinteket követően, a retrospektív megbeszéléseken a fejlesztőcsapat értékelheti saját teljesítményét és kötelező dokumentálnia a folyamat javítása érdekében megcélzott változtatásokat.

2.2.7 *Karbantartás* – a Military Scrum a karbantartási feladatok végrehajtását integrált üzemeltetési platform alkalmazásával írja elő, amely gondoskodik a dokumentált folyamatokról és konfiguráció menedzsmentről, illetve kötelezően előírja a funkcionális tesztelést és a műszaki ellenőrzést is az éles üzembe helyezés előtt.

A Military Scrum által nem tárgyalt fejezetek:

Az alvállalkozók kezelése [79, 2.2.4.5], a Dobozos szoftvertermékek [79, 2.2.4.7] és a Nem szállítandó szoftverek [79, 2.2.4.8] fejezeteket a módszertan nem tárgyalja, nem képezik részét a kutatásnak. A felsorolás alapján megállapítható, hogy a Military Scrum szoftverfejlesztési módszertan folyamatai összhangban vannak az AQAP 2210 szabvány szerkezetével. A módszertan által nem tárgyalt pontoknak, illetve a szabványban előírt részletes követelményeknek való konkrét megfelelésről az adott projektet megvalósító szervezetnek kell gondoskodnia – amely a PMS további bővítésével megvalósítható.

Új szoftverek fejlesztése (R-1)

Egy teljesen újonnan kifejlesztendő szoftver alapú szolgáltatás igényeinek felméréstől egy bevezetett rendszer változásigényeinek kezeléséig a Military Scrum biztosítja a megfelelő projekt menedzsment folyamatokat és dokumentációs sablonokat Az igényfelmérés, a követelményhalmazok, majd a termék feladatlista összeállítása elegendő a követelményrendszer meghatározásához a koncepcionális előtervezés és koncepcióalkotás szakaszaiban.

A Military Scrum 1. fejlesztési menetében a kezdeti követelményrendszer alapján egy új szoftver fejlesztése megkezdhető, az átadás előtti menetekben a követelményrendszer finomítása, majd a fejlesztett szoftver igazítása a módosított követelményrendszerhez támogatott. Az átadást követő fejlesztési menetekben a változáskezelés megoldott a követelményelemzés szintjén. Az evolúciós szoftverfejlesztés techniká-

jának alkalmazásával az újonnan fejlesztett és a már üzembe helyezett szoftver alapú szolgáltatások fejlesztése is fenntartható.

Új műszaki alapokra történő helyezés (R-2)

A Military Scrum által megkövetelt szoftverfejlesztési technikák lehetővé teszik, hogy a fejlesztett szoftver architektúrája és a felhasznált technológiák módosíthatók akár a későbbi fejlesztési menetekben. A CI és az egyéb automatizált tesztek védőhálót biztosítanak a fejlesztett szoftver számára, így a műszaki komponensek frissítése, cseréje biztonságosan kivitelezhető az üzleti logikát tartalmazó forráskód megtartása mellett.

Szoftver cseréje előző rendszer alapján (R-3)

Ha a koncepcionális előtervezés szakaszában olyan követelményt azonosítanak az érintettek, amely alapján egy meglévő rendszer cseréje szükséges a kívánt képességek kialakításához, akkor a koncepcióalkotási szakaszban a Military Scrum a szoftver cseréjét támogató dokumentációs sablonjával elősegíti az új rendszer fejlesztéséhez szükséges feladatlista összeállítását.

A kiváltandó szoftver képességei alapján előállított termék feladatlista képezi a Military Scrum fejlesztési szakaszának bemenetét. A menetek során végrehajtott fejlesztési sprintek szemszögéből a feladatlista forrása lényegtelen, ugyanazon szoftverfejlesztési technikák alkalmazása kötelező, mint egyéb esetekben. A helyesen alkalmazott módszertan garantálja a sikeres implementációt.

Adatmigrálás támogatása (R-4)

Amennyiben a koncepcionális előtervezés szakaszában olyan követelményt azonosítanak az érintettek, amely alapján a kívánt képesség eléréséhez valamilyen adatmigrálási feladat elvégzése indokolt, akkor a Military Scrum három komponensből álló dokumentációs eljárást biztosít az adatmigrálási feladatok megfelelő előkészítéséhez.

Fejlesztési oldalról az adatmigrálást megvalósító szoftverek felhasználása eltér az egyéb szoftverekétől, mert ebben az esetben egyet adat-transzformációt megvalósító, „egyszer használatos” szoftver előállítása a feladat. A szoftverfejlesztési környezet kialakítása során a CI feladata az adatmigrálás végrehajtása fejlesztői környezetben, a transzformációs szabályok helyességének ellenőrzése. Az adatmigrálási feladat

megvalósításához különálló Military Scrum projekt indítása javasolt – legfeljebb néhány menettel.

Rendszerek közötti integráció (R-5)

Rendszerek közötti integrációs feladatok esetén a Military Scrum biztosítja azokat a dokumentációs sablonokat, amelyek segítségével elkészíthető az integrációs fejlesztés feladatrendszere. A módszertan által javasolt eljárás külön projekt indítása az integrációs feladatok elvégzésére, amely során minden érintett rendszer oldaláról el kell érni az integrációs feladatokhoz szükséges érettségi szintet.

A rendszerintegráció alapját egy speciális termék feladatlista képezi, amelynek kialakítása az erre a célra indított Military Scrum projekt koncepcióalkotási szakaszának végére előáll a módszertan erre a célra kialakított követelmény elemzési technikája révén. A termék feladatlista modulokra szűrt nézete alapján a rendszerek különálló fejlesztése, felkészítése az integrációra az egyéb szoftverek fejlesztésével analóg módon elvégezhető, a kapcsolódó rendszerek utánzási (*mock*) technikájával a fejlesztői környezetben. Ezt követően a rendszerintegrációs feladat megvalósításához különálló Military Scrum projekt indítása javasolt – több, rövid átfutású fejlesztési menettel, amikor a rendszer folyamatainak javítása érdekében végrehajtott változások kezelése történik.

Megbízhatóság (S-5)

A fejlesztési szakaszban a jó minőségű, hibamentes, folyamatos működést produkáló szoftverek létrejöttét a TDD és a DDD módszerek, valamint a CI által megkövetelt gépi kód és tesztlefedettség ellenőrzés, illetve a bizonyos időközönként (napi, heti) elvégzett automatizált terheléses, felületi, biztonsági tesztfuttatások szavatolják a manuális kódellenőrzés és tesztelés mellett.

A jó minőségű, hibamentes, folyamatos működési környezet kialakítása visszavezethető egészen a követelményelemzési technikákig, a PMS-ben a várható felhasználók száma és felhasználás jellege kötelezően tárgyalandó kérdésként szerepel. A további üzemeltetési környezettel szemben támasztott megbízhatósági követelmények gyűjtését a módszertan támogatja. Amennyiben szempont a katasztrófatűrés vagy a redundáns adatmentés, akkor a laboratóriumi tesztelés során és különösen az üzemeltetési környezetben a követelményeknek megfelelő infrastruktúra kialakítása a követelmény. A módszertan által előírt integrált üzemeltetési platform segítségével a ter-

helés elosztásos vagy a tartalék rendszerre építő megoldások esetén is frissíthetők a különböző környezetek, illetve megoldott azok megfelelő monitorozása.

Szabványosság (S-6)

A különböző műszaki szabványok megjelenhetnek elvárásként a fejlesztett szoftverrel szemben az igényfelmérés közben is – a megfelelést, a szabványos komponensek integrálását a fejlesztési szakasz munkafolyamatai garantálják. A szoftver architektúrájának kialakításakor, a különböző programcsomagok beágyazásakor a fejlesztők felelőssége a szabványok követése és a szabványos megoldások kialakítása. A módszertan a forráskód áttekintését adja eszközként a megoldások felülvizsgálatához ebből az aspektusból.

Kompatibilitás (S-7)

A fejlesztett szoftver kompatibilitását más rendszerekkel, futtató környezetekkel, infrastruktúrával a CI és egyéb automatizált tesztelések segítségével lehet támogatni, ha a fejlesztési szakaszban a kialakított fejlesztési infrastruktúra modellezi a leendő üzemeltetési környezetet. Ha több fejlesztett szoftverből felépülő szoftver alapú szolgáltatás kialakítása a projekt célja, akkor a Military Scrum megköveteli az egységesített műszaki eljárások kialakítását (compile, build, stb.), ami lehetővé teszi a szolgáltatás moduljai számára a közös folyamatok és infrastruktúra alkalmazását.

Interoperabilitás (S-8)

Napjainkban az egyik leggyakoribb követelmény egy fejlesztett szoftverrel szemben a harmadik fél által fejlesztett rendszerekkel való együttműködés kialakítása. Ilyenkor a rendszereken átívelő helyesen működő folyamatok kialakítása a feladat. A nehézséget az okozhatja, hogy az átjárhatóság lehetősége sem adott a fejlesztői környezetben. Az evolúciós szoftverfejlesztési technika lehetőséget biztosít az utánzás (*mock*) technikájának alkalmazására. A kialakítandó szoftver alapú szolgáltatáshoz tartozó automatizált tesztek képesek a másik rendszer viselkedésének szimulálására a szabványos interfészek mögött. Így a fejlesztett szoftverben megvalósítandó folyamatok helyessége, a megbízhatóság garantálható. Amennyiben kétirányú átjárhatóságról van szó, akkor a fejlesztett rendszer a TDD technikájával megbízhatóan működő interfészt publikálhat a kapcsolódni szándékozó rendszerek számára.

Rugalmasság (S-9)

A fejlesztett szoftver jogosultsági mátrixban az egyes funkciókat különböző jogosultságokhoz is hozzárendelve megteremthető a funkcionális rugalmasság. A Military Scrum a TDD technikájának alkalmazásával támogatja a szabályozott, ugyanakkor rugalmas folyamatok kialakítását. Műszaki szinten a rugalmasság az egyes komponensek cseréjét, módosítását, illetve hangolását jelenti, amelyhez az evolúciós szoftverfejlesztés megfelelő eszközöket biztosít. A kitesztelt rendszerkonfigurációk segítségével a későbbi üzemeltetési környezetek hangolása is támogatható.

Hitelesség (S-10)

Amennyiben a követelményelemzés során olyan funkciók kerülnek azonosításra, ahol valamilyen okból kifolyólag a fejlesztett szoftvernek kell gondoskodnia az adatok hitelességéről, akkor alapszinten ez megvalósítható az adott szolgáltatáson belül végrehajtott kriptográfiai algoritmusok végrehajtásával és a képzett *hash*-ek rendszeren belüli tárolásával. A TDD és a DDD technikák együttes alkalmazásával a hitelesítendő modell és titkosított adatok modellje is kialakítható. Emelt szintű hitelesség kialakításához harmadik fél által biztosított, a követelményeknek megfelelő megbízhatósági szinttel rendelkező tanúsítványok integrálása szükséges a fejlesztett szoftver alapú szolgáltatáshoz.

Az éles üzembe helyezett szoftver alapú szolgáltatás által kezelt hitelesítendő adatokat, illetve a szolgáltatás által folytatott kommunikáció hitelességét a megfelelő titkosítási eljárásokkal lehet garantálni. A fejlesztett szoftveren belül, illetve a futtató és a támogató rendszerek által végzett hitelesítési eljárások működőképességének ellenőrzését a módszertan előírja a műszaki ellenőrzés fázisában. Az integrált üzemeltetési platform diagnosztikai képességének ki kell terjednie a szoftver alapú szolgáltatás hitelesítési képességének ellenőrzésére is, ha az részét képezi a követelményeknek.

Modularitás (S-11)

Abban az esetben, ha a kialakítandó képesség megvalósításához több modulból álló szoftver alapú szolgáltatás fejlesztése szükséges, akkor a minden modul esetében külön-külön indított Military Scrum projektek indítása javasolt. Amikor az egyes rendszerek érettség szintje megfelelő, akkor valósítható meg a rendszerintegráció egy erre a célra indított különálló Military Scrum projekttel.

Skálázhatóság (S-12)

Ha a skálázhatóság követelmények a fejlesztett szoftverrel szemben, akkor a Military Scrum által előírt evolúciós szoftverfejlesztési technika segítségével kiértékelhető a fejlesztett szoftver teljesítménye csökkentett, ideális és megnövelt teljesítménymű hardver infrastruktúrán, mint módosított környezeti tényezőn is. A fejlesztés során minden esetben a követelményeknek megfelelő funkcionalitást kell produkálnia a rendszernek. Amennyiben a skálázhatóságot valamilyen terhelés elosztási technika segítségével kell megvalósítani, akkor a kialakított megoldás automatizált tesztelése ajánlott.

Biztonság (S-13)

A Military Scrum a fejlesztési szakaszban a biztonsági kockázatok kezelésének technikájával segíti elő a fejlesztett szoftver biztonsági mutatóinak fokozását. Az evolúciós szoftverfejlesztési technika lehetőséget biztosít a különböző azonosított kockázatok kezelésére, amennyiben a termék feladatlistában megjelennek a reagálási eljárások. A biztonsági szempontok minden szoftver, illetve szoftver alapú szolgáltatás esetében mások lehetnek, ezért ezeknek a tényezőknek a beépítése szoftverfejlesztési folyamatokba projektfüggő. Tipikus ilyen eljárás a sérülékenységvizsgálat. A szoftverfejlesztési folyamatok fenntarthatóságát, az üzemeltetési folyamatok alatt felmerülő biztonsági kockázatokat és a felhasználásból eredendő egyéb kiberbiztonsági fenyegetéseket a Military Scrum a fejlesztett szoftver alapú szolgáltatás számára kialakított integrált üzemeltetési platform segítségével kezeli.

Felhasználhatóság (S-14)

A Military Scrum agilis követelményelemzési technikáival lehetővé válik a megrendelői igények alapos elemzése. Ezt követően a fejlesztési szakaszban megkövetelt folyamatos és aktív megrendelői jelenlét előrevetíti a felhasználhatósággal kapcsolatos visszajelzések beépítését a rendszerbe még a fejlesztés szakaszában. A TDD technika alkalmazásával a megrendelői igények és észrevételek beépíthetők a rendszerbe még a fejlesztési szakasz során, így a felhasználók már egy felhasználhatósági szempontból is áttekintett rendszerrel találkoznak először.

Információ-megosztás (S-15)

A fejlesztett szoftver jogosultsági mátrixához igazítva a különböző jogosultságokkal rendelkező felhasználók a rendszerben tárolt adatok más és más vetületeit láthatják. A követelményelemzés során a különböző láthatósági körök azonosítása, és ezt követően azok megvalósítása lehetséges a TDD technika segítségével. A rendszerben kialakított jogosultságokkal és a hozzájuk kötött láthatósági körökkel megvalósítható a szabályozott információ-megosztás elve a Military Scrum fejlesztési szakaszain belül.

Adat-konzisztencia (S-16)

A TDD módszer során előírt kód újraszervezés során, illetve a DDD módszer szisztematikus fogalomtér modellezése révén a rendszerben tárolt adatok konzisztenciája statikus modell szinten garantálható. A folyamatok során bekövetkező állapotváltozások helyességét a magas tesztlefedettség, valamint az evolúciós szoftverfejlesztés technikája garantálja. Így a Military Scrum szoftverfejlesztési technikái együttesen szavatolja a konzisztens adatmodellt és az adatkezelést is.

Hatékonyság (S-17)

Az evolúciós szoftverfejlesztés segítségével a fejlesztett szoftver hatékonysága folyamatosan figyelemmel kísérhető, amennyiben egy változtatás jelentős lassulást okoz valamely funkcióiban, akkor a CI-hez tartozó folyamatok futási ideje megnövekszik, az eredményekből láthatóvá válik a teljesítményben mutatkozó probléma. Projektmódszertanként a Military Scrum kiegyensúlyozott menetelést tesz lehetővé, azonban a fejlesztési szakaszban, gyorsan és hatékonyan lehetséges az előrehaladás a termék feladatlista magas prioritású, részletesen specifikált feladatai tekintetében. A Military Scrum menetek fejlesztési szakaszának gyors előrehaladási képessége nem befolyásolja negatívan a projekt költségek alakulását.

Telepíthetőség (S-18)

A fejlesztési szakaszban a CI megléte garantálja alapszinten a szoftver telepíthetőségét, mert visszajelzést ad a szoftver fordítási, valamint telepítő csomag előállítási állapotáról. Az erre épülő evolúciós szoftverfejlesztési keretrendszer kibővíthető a telepíthetőségi képesség ellenőrzésével minden CI futás után a fejlesztői környezetben. Ha a fejlesztői környezet az üzemeltetési környezethez hasonló elemekből áll,

akkor ez a technika elősegíti az éles üzemeltetési környezetben a megfelelő telepíthetőségi képességet.
A Military Scrum integrált üzemeltetési platform meglétét írja elő a fejlesztést követő szakaszok támogatásához, amely az átadási folyamat első lépéseként migrációs telepítést valósít meg. A sikeres és helyes telepítést követően automatizált módon teríthető az adott szoftververzió az éles üzemeltetési környezeten belül.

Robusztusság (S-19)
A fejlesztési szakaszban a szoftver alapú szolgáltatások esetében a különböző terheléses tesztelések segítségével fel lehet készíteni a fejlesztett szoftvert a megfelelő „túlélési" képességre. A terheléses tesztek beépíthetők a Military Scrum által előírt evolúciós szoftverfejlesztési keretrendszerbe. Amennyiben a biztonsági követelmények elemzéséből a terhelés-elosztásos üzemeltetési környezetben való működés is elvárás, akkor a fejlesztett szoftver eme képességének ellenőrzését is az evolúciós szoftverfejlesztési környezet részéve kell tenni.
A Military Scrum a szoftver alapú szolgáltatások üzemeltetési biztonsági kockázatainak kezelésére a terhelés-elosztásos technika vagy a tartalék rendszerek kialakítását javasolja. Ha az üzemeltetési környezet alkalmazza a technikák valamelyikét, akkor is a kiadott szoftververzióknak először át kell esniük a műszaki és funkcionális ellenőrzésen az átadási folyamat részeként. Ezt követően az integrált üzemeltetési platform garantálja a friss szoftververzió minden környezetre történő telepítését.

Karbantarthatóság (S-20)
A fejlesztési szakaszban a karbantarthatóság a feltárt hibák javításának, a változtatási igények megvalósításának hatékony képességét jelenti. A Military Scrum módszertan alkalmazásával a szoftver alapú szolgáltatások fejlesztési szemszögből karbantarthatók, köszönhetően a módszertan által előírt szoftverfejlesztési technikáknak: TDD, DDD, CI, evolúciós szoftverfejlesztés és az ezekre épülő verziókiadás.
A Military Scrum által megkövetelt integrált üzemeltetési platform által megvalósított automatizált telepítési folyamatok, a felhasználói és műszaki konfigurációk követése és menedzselése, a folyamatos rendszerdiagnosztika lehetővé teszi a kialakított szoftver alapú szolgáltatás karbantarthatóságát.

Fenntarthatóság (S-21)

A Military Scrum segítségével fejlesztett szoftverek esetében lehetséges a különböző technológiai, gazdasági, politikai, jogszabályi változások hatásainak vizsgálata, kiértékelése és követése az evolúciós szoftverfejlesztési technika alkalmazásával.

Az integrált üzemeltetési platform elvárt képességei között szerepel az üzemeltetési környezetben a fejlesztett szoftvereket érintő környezeti komponensek modell szintű ismerete. A szerkeszthető üzemeltetési modellen értelmezett karbantarthatósági képesség lehetővé teszi a hosszú távú fenntarthatóságot is.

Szükségszerű a technológiai bővítés lehetővé tétele, az élettartam alatti módosítások, az elévülés kezelése. (S-22)

Az evolúciós szoftverfejlesztési technika segítségével a fejlesztett szoftver elévült technológiai komponensei frissíthetők, a meglévők bővíthetők, a rendszer funkciói módosíthatók a módszertan helyes alkalmazása mellett.

A módszertan által előírt integrált üzemeltetési platform segítségével a fejlesztett szoftver technológiailag kibővített, módosított, frissített verziói az éles üzemeltetési környezetben automatizáltan telepíthetők. A Military Scrum helyes alkalmazása garantálja a korszerű technológiai alapokon megvalósított szoftver alapú szolgáltatások fenntartását.

A fejlesztések számára integrált rendszerszemlélet meghatározása szükséges, amely támogatja a felhasználást, elősegíti a követelményeknek való megfelelést. (S-23)

A korábbiakban bemutatott NATO Rendszer Koncepciónak való megfelelés és az érettségi szint fogalmának bevezetése és ellenőrzése a szoftver alapú szolgáltatások teljes életciklusa alatt a Military Scrum-ot képessé teszi a követelmény kielégítésére.

A szoftverfejlesztési projekt végrehajtása során folyamatos mérőszámok mutassák a fejlesztés alatt álló szoftver pillanatnyi érettségét. (S-24)

A folyamatos integráció technikájával a tesztfuttató környezet folyamatos mérőszámokat mutat a fejlesztett rendszer pillanatnyi érettségéről, a helyes és helytelen végeredményt produkáló automatizált tesztek összesített kimutatásával. A rendszer akkor éri el megfelelő érettséget, ha az automatizált tesztek helyes futási eredményeket produkálnak ideális időben.

A projekt- és a technikai folyamatok támogatása integrált megoldások segítségével valósuljon meg, beleértve a verifikáció, az átadás, a validáció, az üzemeltetés, a karbantartás és a leszerelés folyamatait, valamint kielégítve a nyomon követési infrastruktúra számára szükséges adatgyűjtési követelményeket. (S-25)

A Military Scrum szoftverfejlesztési módszertan a különböző életciklus szakaszok támogatásra különböző informatikai megoldások és rendszerek alkalmazását írja elő. A követelményelemzés támogatásához egy erre a célra kialakított feladatkezelő rendszer vagy a szoftverfejlesztés támogatásához előírt rendszer közös használata a módszertani elvárás. Ezt követően az integrált szoftverfejlesztési folyamatokról az evolúciós szoftverfejlesztési környezet, míg az üzemeltetés támogatásról az integrált üzemeltetési platform gondoskodik. A felsorolt eszközök az egyedi azonosító képzési technika segítségével kielégítik a nyomon követési infrastruktúra alapkövetelményeit.

Adatkezelési szempontból megbízható és biztonságos rendszerek kialakítása a cél. (I-9)

A Military Scrum szoftverfejlesztési módszertan helyes alkalmazása lehetővé teszi a speciális igényeknek megfelelő, kifinomult szoftvertervezési minták kialakítását is. Az 5. fejezetben bemutatott különböző tervezési minták a valóságot jobban tükröző, életszerű folyamatok modellezését teszik lehetővé az informatikai rendszereken belül. A módszertan és a tervezési minták együtt támogatják az adatok fizikai törlését a rendszer integritásának megőrzése mellett.

Az adatkezelés, az adatmentés és az adattovábbítás maximális védettségének megteremtése alapvető fontosságú. (I-10)

A Military Scrum módszertan és a Holder tervezési minták együttes alkalmazása a fenti információbiztonsági követelmény szoftvertechnológiai aspektusaira ad választ. Egy adott rendszeren belül a módszertannal és az alkalmazható tervezési mintákkal kifinomult folyamatokat, jól definiált adatkezelési tevékenységeket lehet megvalósítani. Az adatkezelők és az adatfeldolgozók szerepei tisztázhatók. A rögzíthető, a megismerhető, a módosítható, a törölhető adatok külön-külön kezelhetők egy természetes személy vagy más adatkezelési szempontból kritikus entitás esetében. A múltbéli és az aktuális adatok megkülönböztetésével az adatok továbbításához szükséges folyamatok is naprakész információk segítségével valósíthatók meg.

A sérülékenység szintjének ellenőrzését támogató szoftverfejlesztési és tesztelési módszerek kialakítása szükséges az újonnan fejlesztett rendszerek számára. (I-11)
Ha a követelményelemzést követően a termék feladatlistában megjelenik a sérülékenységek szintjének ellenőrzése követelményként, akkor a Military Scrum az evolúciós szoftverfejlesztési környezetben lehetőségét biztosít az automatizált sérülékenységvizsgálatra a CI folyamatainak részeként.

A fejlesztési fázisban lévő alkalmazások biztonsági szintjének ellenőrzését megfelelő szoftverfejlesztési eszközökkel kell támogatni. (I-12)
A Military Scrum biztonsági kockázatok elemzésére szolgáló technikájával a termék feladatlistában megjelennek az adott szoftverrel szemben támasztott biztonsági követelmények direkt vagy indirekt módon. Ezt követően minden követelmény teljes vagy részleges biztonsági vetületét a Military Scrum evolúciós szoftverfejlesztési környezete (CI, egyéb tesztek) képes ellenőrizni a projekt fejlesztési szakaszában.

A minőségbiztosítási folyamatok képezzék részét a kialakítandó szoftverfejlesztési módszertanoknak. (Q-1)
Minden egyes életciklus szakaszokon belül a Military Scrum szoftverfejlesztési módszertan előírja a minőségbiztosító szerepkör betöltését, illetve meghatározza a minőségbiztosítási feladatokat a PMS-ben a szoftver alapú szolgáltatás teljes élettartama alatt.

A belső minőségbiztosítási eljárások magas szintű támogatása legyen lehetséges a szoftvertechnológiai folyamatok szintjén. (Q-2)
A Military Scrum definíciója szerint a minőségirányítás, mint a működést lehetővé tévő rendszerek egyik alkotó eleme a minőségbiztosító és projektmenedzser közös munkájával valósul meg. Ennek köszönhetően a belső minőségbiztosítási eljárások szerves részét képezik a módszertan által lefektetett folyamatoknak. A projektmenedzsment egyik legfontosabb feladata a belső minőségbiztosítási folyamatok végrehajtásának ellenőrzése. A PMS-ben minden életciklus szakaszra vonatkozóan szerepelnek az ellenőrizendő minőségbiztosítási kérdések és a támogató eljárások is, következésképpen a szoftvertechnológiai folyamatok maga szintű minőségbiztosítási eljárásai adottak a Military Scrum alkalmazásával.

A szoftvertechnológiai módszerek rendelkezzenek átlátható dokumentációs technikákkal és eljárásmintákkal. (Q-3)

A Military Scrum a szoftverfejlesztési projektek jellegétől függő dokumentációs sablonokat határoz meg, amelyek helyes alkalmazását a PMS szavatolja. Az életciklus szakaszonként kötelezően végrehajtandó tevékenységek, a módszertan által biztosított dokumentációs eszközök, valamint a minőségbiztosító (QA) által vizsgálandó kérdések és eljárások egyaránt részét képezik a projektvezetés által felügyelt dokumentumnak.

Az IT-fejlesztések különböző fázisaira alkalmazható minőségbiztosítási technikák kerüljenek kialakításra. (Q-4)

A Military Scrum a projekt koncepciójának kialakítása során előállítja az adott fejlesztés jellegéhez igazodó követelményelemzési dokumentumokat. A PMS tartalmazza a minőségbiztosító (QA) által vizsgálandó kérdéseket és a módszertan által nyújtott eszközöket a módszertan minden szakaszára vonatkozóan.

A NATO SLCM koncepció alkotási szakasz során keletkező szoftverfejlesztéshez köthető dokumentumoknak helyt kell kapniuk a kialakítandó szoftverfejlesztési módszertan dokumentum sablonjaiban. (D-1)

- *Kiválasztott életciklus modell* – Military Scrum alkalmazásakor egyértelmű.
- *Program menedzsment terv* – rendszerintegrációs projektek/programok támogatására alkalmas a Military Scrum, ekkor a PMS alapján megvalósítható a program menedzsment.
- *Projekt menedzsment terv* – a Military Scrum esetében PMS alapján megvalósítható a projekt menedzsment.
- *A javasolt megoldás bemutatása* – az 1. fejlesztési menetben indított prototípusok tapasztalati alapján frissített termék feladatlista felel meg a dokumentumnak.
- *Az érdekeltek által meghatározott kezdeti rendszerkövetelmények* – az 1. fejlesztési menet koncepcionális szakaszaiban előálló igények, követelményhalmazok és termék feladatlista
- *Kezdeti tervezési dokumentáció* – az 1. fejlesztési menet termék feladatlistája, optimális esetben bővítve a prototípusok előállítások tapasztalatival.

- *Életciklusra és HR költségekre vonatkozó dokumentáció* – a Military Scrum szoftverfejlesztési módszertan direkt módon nem tárgyalja a költségeket, ugyanakkor a termék feladatlista becsült story pontjai alapján durva becslések előállíthatók a HR és az életciklushoz kapcsolódó költségek vonatkozásában.
- *Előzetes projektütemezés* – a Military Scrum meneteinek dokumentálása során kötelező az ütemezés megadása a fejlesztési szakasz vonatkozásában a PMS-ben.
- *Kezdeti konfiguráció menedzsment terv* – Műszaki követelményhalmazok részeként kezeli a módszertan.
- *Megvalósíthatósági értékelés* – a koncepció alkotás szakaszában készített prototípusok szakértők általi kiértékelése, részét képezi a PMS-nek.
- *Kezdeti kockázat meghatározás, értékelés és kezelési tervek* – a Military Scrum a biztonsági kockázatok elemzésére szolgáló módszerével a követelményhalmazok és a termék feladatlista segítségével kezeli a kockázatokat.

A NATO SLCM fejlesztési szakasz során keletkező szoftverfejlesztéshez köthető dokumentumok előállítását dokumentum sablonok szintjén támogassa a kialakítandó szoftverfejlesztési módszertan. (D-2)

A sprint feladatlista elemeinek rendszerezésére szolgáló lista struktúrája megegyezik a termék feladatlista struktúrájával, azzal a különbséggel. hogy itt a követelmények alapján megtervezett fejlesztési feladatok szerepelnek. A fejlesztési feladatok menedzselésére a módszertan integrált feladatkezelő rendszer használatát írja elő. A Military Scrum szem előtt tartva az agilis szoftverfejlesztési szempontokat, az előírt dokumentumok számát minimalizálja az előírt folyamatok támogatásához. Azonban a termék feladatlista követelményeihez, a sprint feladatlista funkcióihoz és az átvételi tesztek sablonjához hozzá lehet kapcsolni a NATO SLCM-ben előírt különböző dokumentumokat:

- *Megfelelőség- és helyesség vizsgálati dokumentáció (tervek, eljárások, stb.)* – a funkciókhoz tartozó tesztforgatókönyvek és az átvételi teszt támogatására szolgáló sablon alapján előállítható a dokumentáció.
- *Megfelelőségi- és helyesség vizsgálati eredmények (összefoglaló jelentések)* – az integrált feladatkezelő rendszerben a fejlesztési feladatokhoz rögzíthető a

manuális tesztelés eredménye (megfelelőség), majd az átvételi tesztek végrehajtásakor az eredmények vezetendők a kapcsolódó sablonban.

- *Szoftvertervezési dokumentáció (architektúra, tervezési dokumentáció)* – prototípus előállítás során a fejlesztési feladatokhoz tartozó leírások alapján előállítható a dokumentum.
- *Interfész specifikáció* – az interfész fejlesztési feladatokhoz kapcsolódó dokumentumok alapján és az evolúciós szoftverfejlesztési környezet működése alapján előállítható a dokumentum.
- *Szoftver/Hardver integrációs tervek és specifikáció* – az evolúciós szoftverfejlesztési környezet kialakításhoz tartozó feladatok leírásai alkalmasak a dokumentum előállításának támogatásához.
- *Rendszer meghatározás a szükséges tartalmi elemekkel* – a koncepcionális szakaszok során keletkező dokumentumok.
- *Az előállítási (felhasználási) tervek, az üzemeltetési tervek, az üzemeltetési dokumentáció és a karbantartási stratégia/terv* előállítását a Military Scrum követelményelemzési technikái támogatják. Ezt követően a támogatási és karbantartási eljárások az integrált üzemeltetési platform segítségével valósíthatók meg figyelembe véve a módszertan által előírt átadási folyamatot.
- *Frissített kockázat meghatározás, értékelés és kezelési tervek* – az evolúciós szoftverfejlesztési környezetben tapasztalt jelenségek visszavezethetők a koncepcionális tervezési dokumentumok szintjére, így a kockázatkezelési folyamat újra indítható.
- A fejlesztési szakaszok végén előálló példa konfigurációs állományok alapján frissíthető a *konfiguráció menedzsment terv* a rendszer lehetséges paramétereit illetően.
- A frissített költségtervek, a programra vonatkozó egyetértési nyilatkozat támogatásához a fejlesztési menetek, illetve fejlesztési szakaszok erőforrás igényei alapján frissíthetők, az ütemezés és a költségek vonatkozásában a program fenntartásának, folytatását irányzó egyetértési nyilatkozat elkészítése támogatható.

A kialakítandó szoftverfejlesztési módszertan részeként kerüljenek meghatározásra azok a dokumentációs sablonok, amelyek megfelelően támogatják a NATO SLCM üzembe helyezési, a felhasználási és a támogatási szakaszok végrehajtását. (D-3)

- *Karbantartási/támogatási adatok* – a módszertan elvárásai szerint kinyerhetők az integrált üzemeltetési platformból.
- *Frissített hiba- és élettartam adatokra vonatkozó dokumentumok* – a Military Scrum biztosítja a hibajegyek rögzítéséhez a megfelelő sablonokat, az integrált üzemeltetési platform által előírt rendszerdiagnosztikai képességei megfelelő alapot nyújthatnak az élettartammal kapcsolatos dokumentumok előállításához.
- A NATO SLCM dokumentumokon túl a Military Scrum előírja az átadási tesztek támogatását, a hibajegyek vezetését, és a változtatási igények gyűjtését támogató dokumentumok alkalmazását is.

A kutatás céljaként kialakítandó szoftverfejlesztési módszertanhoz tartozzon projekt menedzsment sablon. (D-4)

A projekt menedzsment sablon (PMS) meghatározása szerint a Military Scrum szabályozott végrehajtását támogató dokumentum, amely az értekezés 1. számú mellékletben található, így a követelmény teljesül.

A szoftverekre vonatkozó szellemi tulajdon, az elektronikus adatmegosztás, az adatbázisok létrehozása és használata kapjon helyt a projekt menedzsment dokumentációs sablonban. (D-5)

A Military Scrum a koncepcionális előtervezés szakaszában, a követelményhalmazok meghatározásakor a projektvezetés számára kötelező feladatként írja elő a fenti kérdések tisztázását, azok részét képezik a PMS-nek.

A kutatás céljaként kialakítandó szoftverfejlesztési módszertanhoz tartozó projekt menedzsment sablon tárgyalja a minőségirányítási feladatokat is. (D-6)

A Military Scrum alkalmazásakor a minőségirányítási feladatokat a projektvezetés, mint működést lehetővé tévő rendszer látja el. A PMS minden életciklus szakasz számára meghatározza a minőségbiztosítási eljárásokat és előírja az elvégzendő minőségbiztosítási feladatokat, következésképp a módszertan teljesíti a megfogalmazott dokumentációs követelményt.

A kialakítandó szoftverfejlesztést támogató dokumentációs technikák támogassák a kockázatok elemzését és kezelését a projektek végrehajtása során. (D-7)

A biztonsági követelmények elemzésére, a kockázatok követelményhalmazban történő feltárására és a kockázatkezelési megoldások termék feladatlistába szervezésével a biztonsági kérdéskörbe tartozó feladatok integráltan kezelhetők, szerves részét képezve a módszertannak. A módszertan a követelmények menet közben történő beépítését a dokumentációs keretrendszeren belül támogatja, azok megvalósulását a későbbi szakaszok során bemutatott fejlesztési és üzemeltetési technikák garantálják.

A fejlesztés során az evolúciós szoftverfejlesztés technikájával feltárt, a követelményrendszer részét nem képező kockázatok beépítése lehetséges a következő fejlesztési menet koncepcionális szakaszai során a követelményhalmazokba és a termék feladatlistába, amennyiben a projektvezetés ezt úgy véli a szakértők jelzése alapján, a folyamatot a PMS tartalmazza.

Az üzemeltetés során az integrált üzemeltetési platform által detektált, a követelményrendszer részét eddig nem képező kockázatok beépítése lehetséges a következő fejlesztési menet koncepcionális szakaszai során. A követelményhalmazok és a termék feladatlista frissítése lehetséges, amennyiben a szakértők jelzése alapján a projektvezetés ezt indokoltnak véli. A folyamatot a PMS tartalmazza.

A kialakítandó szoftverfejlesztést támogató dokumentációs technikák támogassák a konfiguráció menedzsmentet érintő tevékenységek során keletkező adatok dokumentálását. (D-8)

Ha megrendelői oldalon konfiguráció menedzsmentet érintő igény merül fel, akkor annak beépítése a szoftver követelményrendszerébe a műszaki követelményhalmazok segítségével megvalósítható. A megrendelői igények és a prototípusok előállítása során keletkező kezdeti konfigurációs beállítások dokumentálását már el kell végezni a koncepcióalkotási szakaszban, a követelményeket a termék feladatlista részévé lehet tenni.

A fejlesztési szakaszban a konfiguráció menedzsment a fejlesztett szoftver paramétereiben bekövetkező változásokat jelenti. A központi verziókövető rendszerből a változások és az ahhoz tartozó fejlesztési feladat azonosítók visszakövethetők. A fejlesztési szakasz végén kiadott telepítő csomagnak tartalmaznia kell az adott szoftververzióhoz tartozó helyes paraméterezést mutató minta konfigurációs állományt.

A módszertan által előírt integrált üzemeltetési platform segítségével végrehajtott konfiguráció módosítások jól dokumentáltak, a szoftver alapú szolgáltatás teljes élettartama alatt kinyerhetők a rendszerből, a kezelt adatok körét a PMS tartalmazza.

A szoftverfejlesztési projekt teljes életciklusa során keletkező információ strukturált formában kerüljön megőrzése és később álljon rendelkezésre. (D-9)

A megfelelően vezetett PMS, illetve a Military Scrum követelményelemzést támogató dokumentumai a módszertan első két szakasza során keletkező igényeket, követelményhalmazokat, követelményeket, erőforrásbecsléseket, kapcsolódó tervezési dokumentumokat megőrzik és folyamatosan rendelkezésre állnak az erre a célra kialakított feladatkezelő rendszerben a módszertan követelményei szerint.

A Military Scrum előírja integrált feladatkezelő megoldás használatát a fejlesztés támogatásához, amelynek helyes alkalmazása mellett a fejlesztési szakasz során megvalósított funkciók és feladatok, valamint a hozzájuk tartozó tesztforgatókönyvek és dokumentációk strukturált formában megőrizhetők és később kinyerhetők.

Az üzemeltetés támogatás számára előírt integrált üzemeltetési platformból a telepítési és egyéb folyamatok során keletkező információk bármikor kinyerhető a módszertani követelmények alapján.

Szoftverfejlesztésre értelmezett követelményeket, fejlesztési feladatokat, verziókat, konfigurációs beállításokat, üzemeltetési és karbantartási tevékenységeket felölelő nyomkövetési infrastruktúra kialakítása szükséges. (D-10)

- *Követelmények* – a projekt koncepciójának kialakítása során a felmért igények, az azonosított követelményhalmazok és a kialakított termék feladatlista elemei egyaránt egyedi azonosítással rendelkeznek, így a Military Scrum első két szakaszában keletkező lényeges információk tekintetében hozzájárulnak az elvárt nyomon követési infrastruktúra kialakításához.
- *Fejlesztési feladatok* – a fejlesztési feladatok azonosító képzése egyedi, tartalmazza a követelményként szolgáló termék feladatlista elem azonosítóját is.
- *Szoftververziók* – a fejlesztett szoftver aktuális verziója a verzió főszám segítségével kapcsolódik az aktuális Military Scrum fejlesztési menethez. A verziókiadás során a termék feladatlista azonosítóinak beágyazása a telepítő csomagba követelmény.

- *Konfigurációs beállítások* – a verziókiadás során a fejlesztett szoftver minta konfigurációjának megadása szükséges, így a konfigurációs paraméterlista szoftver verziószámhoz kapcsolható. Üzemeltetés során bekövetkező változások nyomon követhetőségéről az integrált üzemeltetési platform konfiguráció felügyeleti funkciója gondoskodik. Ha telepítés következtében történt a változás, akkor az adott szoftververzió is kinyerhető a rendszerből.
- *Üzemeltetési és karbantartási tevékenységek* – a fejlesztett szoftver alapú szolgáltatás frissítéséhez és karbantartásához kapcsolódó folyamatokat a módszertan szabályozza, a keletkező adatok kinyerhetők az integrált üzemeltetési platformból, amelyek alapján a tevékenységekhez köthető szoftververziók, követelmények azonosíthatók.