



TAMPEREEN
AMMATTIKORKEAKOULU

Ylläpidettävyyden kehittäminen pitkän elinkaaren johtamisjärjestelmiin

Olli Piipponen

Opinnäytetyö
Toukokuu 2018

Teknologiaosaamisen johtamisen ylempi AMK-tutkinto



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Teknologiaosaamisen johtamisen ylempi AMK-tutkinto

Olli Piipponen
Ylläpidettävyyden kehittäminen pitkän elinkaaren johtamisjärjestelmiin

Opinnäytetyö 53 sivua, joista liitteitä 0 sivua
Toukokuu 2018

Tässä opinnäytetyössä tutkitaan ylläpidettävyyden kehittämistä pitkän elinkaaren johtamisjärjestelmiin. Johtamisjärjestelmät ovat puolustuskäyttöön tarkoitettu tietojärjestelmiä, joilla tuetaan reaaliaikaista taktista johtamista ja taistelun johtamista. Johtamisjärjestelmillä on keskimääräistä pidempi elinkaaren tavoite verrattuna keskimääräisiin ohjelmistojärjestelmiin. Pitkän elinkaaren myötä ylläpidettävyyteen ja sen muodostumiseen pitää kiinnittää huomiota. Ylläpidettävyydellä on merkittäviä vaikutuksia järjestelmän elinkaarikustannuksiin.

Ylläpidettävyydellä tarkoitetaan sitä kuinka nopeaa ja helppoa johtamisjärjestelmään on tehdä erilaisista tarpeista tulevia muutoksia. Ylläpidettävyyden tärkein tavoite on, että johtamisjärjestelmä täyttää operatiivisen tarpeen mahdollisimman pitkään mahdollisimman kustannustehokkaasti.

Opinnäytetyössä lähestytään tutkimusongelmaa laadullisesta näkökulmasta ja sitä pyritään selittämään sisältöanalyysin menetelmillä. Tärkeänä tietolähteenä työssä toimivat asiantuntijahaastatteluista saadut tiedot.

Työn toimeksiantaja on Insta DefSec Oy, jolla on pitkä historia puolustusvoimien käyttöön tarkoitettujen pitkän elinkaaren johtamisjärjestelmien rakentamisesta ja ylläpitämisestä.

Tutkimuksessa havaittiin, että ylläpidettävyyden muodostumiseen vaikuttavat tekijät voidaan jakaa korkealla tasolla kolmeen luokkaan. Näitä ovat johtamisjärjestelmän suunnitteluun ja rakentamiseen liittyvät asiat, järjestelmiä rakentaviin ihmisiin liittyvät asiat ja ylläpidon rahoitukseen liittyvät asiat. Näistä osa-alueista valittiin haastattelun perusteella tärkeimmiksi nostetut asiat ja tutkittiin niitä tarkemmin. Näkökulmana osa-alueissa on ylläpidettävyyden rakentaminen.

Lopputuloksena päädyttiin siihen, että kaikki osa-alueet tulee huomioida vähintään jollakin tasolla. Jos jokin osa-alue jätetään huomioimatta, ei johtamisjärjestelmä ole ylläpidettävä. Ylläpidettävyyden rakentamisella ja säilyttämisellä on selkeät edut, joista hyötyvät järjestelmä, toimittaja, tilaaja ja käyttäjä.

Asiasanat: ylläpidettävyys, johtamisjärjestelmät

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Master's Degree Programme in Strategic Leadership of Technology-Based Business

OLLI PIIPPONEN:

Creating maintainability to command and control systems with long life cycle

Master's thesis 53 pages, appendices 0 pages
May 2018

Purpose of this master's thesis is to study how maintainability affects command and control (C2) systems with a long life cycle. C2 systems are military software systems which are made to support real time tactical management and combat management. C2 systems have a longer life cycle expectancy than average computer based software system. Because of the long life cycle maintainability and how it is formed within the system needs to be taken into account. Maintainability has a significant impact on the C2 systems life cycle costs.

Maintainability is defined as ease of making changes in the C2 system. Needs of the changes can come from different reasons. The main goal of maintainability is to keep the C2 system in service fulfilling its need as long as possible with as minimal effort as possible.

In this master's thesis the subject is approached from the qualitative point of view. Research utilizes content analysis method. One important source of information was the expert interviews conducted as part of the thesis.

Client of this thesis is Insta DefSec Oy which has a long history with building and maintaining military C2 systems with long life cycle.

In the research it was noted that maintainability is divided into three main categories. These categories are planning and building the C2 system, people developing the C2 systems and the financial aspect of maintainability. Within these three categories the most important things based on the interviews were chosen for closer examination.

In conclusion all of the three main categories need to be taken into account at least at some level if maintainability is to be achieved. If one of the aspect in neglected maintaining of the C2 system will not be possible. The value added from maintainability to C2 system itself, supplier and customer are clear.

Key words: maintainability, command and control systems

4	YLLÄPIDETTÄVYYDEN MUODOSTAMINEN	26
4.1	Kehittävä ylläpito.....	26
4.2	Johtamisjärjestelmän toteutustapa ja lopputulos.....	27
4.2.1	Elinkaaren suunnittelu ja hallinta.....	27
4.2.2	Prosessit ja laatu	28
4.2.3	Järjestelmän arkkitehtuuri	30
4.2.4	Teknologiavalinnat.....	33
4.2.5	FOSS- ja yhteiskäyttökomponentit	35
4.2.6	Rajapinnat	36
4.2.7	Laadunvarmistus ja todentaminen.....	37
4.2.8	Dokumentointi	40
4.3	Osaamisen ylläpitäminen	42
4.3.1	Tehokas perehdytys.....	42
4.3.2	Osaamisen johtaminen	43
4.3.3	Motivaatio	45
4.4	Johtamisjärjestelmien ylläpidon rahoitus.....	46
4.4.1	Ylläpidon rahoitus ja erilaiset rahoitusmallit	46
4.4.2	Asiakkaan tunteminen	48
4.4.3	Jatkokehitysmahdollisuuksien tunnistaminen.....	49
5	POHDINTA.....	51
5.1	Johtopäätökset tuloksista	51
5.2	Lisäarvo	52
5.3	Jatkotutkimus	53
	LÄHTEET.....	54

1 JOHDANTO

1.1 Tutkimuksen tausta, tavoitteet ja rajaukset

Tässä opinnäytetyössä on tarkoitus tutkia pitkän elinkaaren johtamisjärjestelmien ylläpidettävyyden muodostumista. Tarkoitus on eritellä, mistä eri kokonaisuuksista ylläpidettävyys muodostuu sekä analysoida näitä osa-alueita kokonaisjärjestelmän ylläpidettävyyden näkökulmasta.

Työn tutkimusongelma on laadullinen ja sen on tarkoitus vastata kysymykseen miten ylläpidettävyys muodostuu pitkän elinkaaren johtamisjärjestelmiin. Määrällistä tutkimusongelmaa ei ole tarkoitus työssä tutkia, koska toimenpiteiden vaikutusten mittaaminen olisi haastavaa.

Tässä työssä ylläpidettävyyttä tutkitaan puolustusvoimien käyttöön tarkoitettujen pitkän elinkaaren johtamisjärjestelmien näkökulmasta. Pitkän elinkaaren johtamisjärjestelmillä tarkoitetaan johtamisjärjestelmiä, joiden tavoiteltava elinkaari on 15 - 25 vuotta. Työ tulee käsittelemään ainoastaan järjestelmien ohjelmisto-osuuksia sekä niihin liittyviä kehitysprosesseja, joten ulkopuolelle suljetaan kaikki ajoympäristöihin ja fyysisiin laitteisiin liittyvä elinkaaritarkastelu. Tämä rajaus on tehty sen vuoksi, että työssä halutaan keskittyä ohjelmistokehityksen elinkaarivaikutuksiin.

1.2 Työn toimeksiantaja

Opinnäytetyön toimeksiantaja on Insta DefSec Oy, joka on osa Insta Group -konsernia. Insta DefSec Oy:n ydinliiketoiminta-alueita ovat kriittiset tietojärjestelmät puolustus- ja koulutuskäyttöön, hätä- ja hälytyskeskusjärjestelmät sekä kyberturvallisuus. Insta DefSec Oy:n historia painottuu maanpuolustuksen johtamisjärjestelmä- ja koulutusjärjestelmäosaamiseen, jota on myöhemmin laajentunut siviilisektorille. Insta DefSec Oy on puolustusvoimien strateginen kumppani. Tämän lisäksi Insta DefSec Oy:n asiakkaita ovat mm. valtionhallinta, julkinen sektori, pankit, turvallisuusviranomaiset ja korkea turvallisuustasoa vaativat yritykset. Tämän opinnäytetyön toimeksiantaja yrityksen sisällä on ilmapuolustuksen johtamis- ja koulutusjärjestelmiä tuottava liiketoimintayksikkö.

1.3 Tutkimusaineisto ja -menetelmä sekä niiden valintaperuste

Tutkimusaineisto työtä varten kerättiin alan kirjallisuudesta ja asiantuntijahaastatteluista. Haastatteluissa hyödynnettiin delfoi-menetelmää. Tutkimusaineisto kerättiin näillä menetelmillä, koska tutkimuksen aihe ei ole kovin yleinen. Yleisesti ohjelmistojärjestelmien ylläpidettävyyttä on tutkittu jonkin verran, mutta maanpuolustukseen tarkoitettujen johtamisjärjestelmien osalta julkisia referenssejä aiheesta on vähän. Tämän vuoksi tärkein tutkimusaineistonkeruutapa työssä oli ehdottomasti asiantuntijoiden haastattelut.

Tutkimusaineiston tulkitsemiseen hyödynnettiin tutkimusmenetelmänä kvalitatiivista sisällönanalyysimenetelmää, jossa ”sisällönanalyysin avulla pyritään muodostamaan tutkittavasta ilmiöstä tiivistetty kuvaus, joka kytkee tulokset ilmiön laajempaan kontekstiin ja aihetta koskeviin muihin tutkimustuloksiin.” (Saaranen-Kauppinen & Puusniekka, 2018).

2 PITKÄN ELINKAAREN JOHTAMISJÄRJESTELMÄT

2.1 Mitä ovat pitkän elinkaaren johtamisjärjestelmät?

Pitkän elinkaaren johtamisjärjestelmillä tarkoitetaan tässä työssä Puolustusvoimilla operatiivisessa käytössä olevia johtamisjärjestelmiä, joiden suunniteltu elinkaari on 15 -25 vuotta. Nämä järjestelmät voivat olla myös muina kuin kriisiaikoina ympärivuorokautisessa käytössä. Vähintäänkin nämä järjestelmät ovat koko niiden elinkaaren ajan koulutuskäytössä, jolla varaudutaan mahdollisiin kriisitilanteisiin.

Johtamisjärjestelmät ovat järjestelmäkokonaisuuksia, jotka koostuvat ohjelmisto-osuksista, alustaratkaisuksista sekä muista verkko- ja laiteratkaisuista. Johtamisjärjestelmien tarkoitus on tukea ja mahdollistaa tietyn osa-alueen taistelun johtamista, taktista johtamista tai ylemmän tason strategista johtamista maanpuolustuksellisissa tarkoituksessa. Johtamisjärjestelmän kerää tietoa eri lähteistä, koostaa tietomassaa ja tarjota sitä helposti ymmärrettävän tilannekuvan muodossa tukemaan päätöksentekoa. Päätöksenteon tuen lisäksi johtamisjärjestelmällä mahdollistetaan päätöksen toimeenpaneminen sekä tästä saatavan vasteen tai palautteen vastaanottaminen.

Erilaisia johtamisjärjestelmiä eri johtamisen tasoille ja tarkoituksiin on useita. Tässä opinnäytetyössä ei ole tarkoitus kartoittaa erilaisten johtamisjärjestelmien ja käyttötarpeiden vaikutuksia, vaan johtamisjärjestelmät käsitellään yhtenä kokonaisuutena. Todellisuudessa näitä eroja on kuitenkin olemassa, mutta koska tämä opinnäytetyö halutaan pitää julkisena niitä ei käsitellä.

2.2 Mitä on ylläpidettävyys johtamisjärjestelmissä?

IEEE Standard Glossary of Software Engineering Terminology määrittelee ylläpidettävyyden seuraavasti: “The ease with which a software system or component can be modified to correct faults, improve performance or other attributes, or adapt to a changed environment.” (IEEE, 2003).

Hieman tarkemmin avattuna sovelluksien ja sovelluksista muodostuvien järjestelmien ylläpidettävyys voidaan määritellä siten, kuinka nopeaa ja helppoa järjestelmään on tehdä alla olevia asioita.

- Kyky korjata vikoja järjestelmästä ilman, että korjauksesta aiheutuu uusia vikoja.
- Kyky tuottaa järjestelmään lisätoiminnallisuutta rikkomatta olemassa olevaa toiminnallisuutta.
- Käytettävyyden parantaminen.
- Suorituskyvyn lisääminen.
- Teknisen velan purkaminen.
- Tietoturvan parantaminen tai olemassa olevien tietoturvaluokkien korjaaminen.
- Kyky päivittää tai vaihtaa järjestelmän ajoympäristöä tai siinä käytettäviä teknologioita.
- Kyky perehdyttää uusia työntekijöitä järjestelmän kehittämiseen.

Tämän opinnäytetyön osana toteutetun haastattelututkimuksen tuloksien pohjalta ylläpidettävyuden muodostuminen järjestelmään voidaan jakaa kolmeen luokkaan.

- Järjestelmään itseensä liittyvät asiat, kuten arkkitehtuuri, tekniset ratkaisut, rajapinnat ja niiden dokumentointi.
- Järjestelmää kehittäviin ihmisiin liittyvät, kuten kyky ymmärtää teknisiä ratkaisuja, kyky perehdyttää ihmisiä, mentorointi ja osaamisen johtaminen.
- Rahoituksen takaaminen ylläolevien asioiden tekemisen mahdollistamiseksi.

Johtamisjärjestelmien ylläpidettävyys muodostuu samoista asioista kuin järjestelmien ylläpidettävyys yleisemmin. Merkittäviä eroja johtamisjärjestelmien ylläpidettävyydellä sisällön kannalta ei ole. Puolustusikäisiin tarkoitetuilla johtamisjärjestelmillä ylläpidettävyys näkökulmaan tulee kuitenkin kiinnittää enemmän huomioita pitkistä elinkaarista johtuen.

Haastatteluissa yksi tärkeä asia, joka nousi esiin useampaan kertaan oli asiakkaan tunteminen ja pitkät asiakassuhteet. Asiakkaan tunteminen koettiin tärkeäksi, koska se mahdollistaa asiakastarpeiden paremman ymmärtämisen. Tällä on merkittävä vaikutus siihen, kuinka hyvin järjestelmä vastaa rakentamisen jälkeen alkuperäiseen tarpeeseen, jota varten se on rakennettu. Mikäli asiakkaan tarvetta ja käyttötapaa ei ole ymmärretty oikein,

vaikuttaa se merkittävästi rakennettavaan järjestelmään ja sitä kautta myös siihen kohdistuviin muutostarpeisiin ja ylläpitoon. Asiakastarpeiden ymmärtäminen vaikuttaa kaikkiin osa-alueisiin.

2.3 Pitkän ja lyhyen elinkaaren johtamisjärjestelmän erot

Lyhyen elinkaaren johtamisjärjestelmät voivat olla huomattavasti lyhyemmän ajan käytössä. Niiden rooli voi olla jonkin teknologian tai konseptin protoilu tai kokeilu. Toisaalta jokin järjestelmä voi olla tietyn lyhyemmän siirtymäajan korvaamassa tiettyä käyttötarkoitusta ennen kuin lopullinen järjestelmä on saatu operatiiviseen käyttöön. Syitä, miksi jokin johtamisjärjestelmä tai sen osa on lyhyen ajan käytössä, voi olla monia. Tärkeintä näissä järjestelmissä on aina tunnistaa etukäteen johtamisjärjestelmän oletettu käyttöikä ja –tarkoitus.

Erot pitkän ja lyhyen johtamisjärjestelmän välillä ovat suuria. Lyhyen elinkaaren johtamisjärjestelmässä ja sen kehittämisessä ei tarvitse huomioida vastaavia asioita kuin pitkän elinkaaren johtamisjärjestelmässä. Lyhyen elinkaaren johtamisjärjestelmän teknisellä toteutustavalla tai arkkitehtuurilla ei ole niin suurta merkitystä. Tämä pätee varsinkin silloin, kun kyseessä on demonstrointitarkoitukseen rakennettu järjestelmä. Osaamisen hallinta menettää merkityksen lyhyen elinkaaren johtamisjärjestelmien kehittämisessä. Järjestelmää kehittävät ihmiset ovat hyvin todennäköisesti käytettävissä koko sen elinkaaren ajan. Jos vaihtuvuutta tapahtuu, on perehdyttäminen helpommin järjestettävissä.

2.4 Muuttuvat vaatimukset ja kyky vastata niihin

Johtamisjärjestelmien vaatimukset muuttuvat ja täydentyvät. Johtamisjärjestelmien vaatimusten muuttumiseen vaikuttavat esimerkiksi seuraavat asiat.

- Uhkakuvien muuttuminen.
- Johtamistavan ja käyttäjien tarpeiden muuttuminen, eli operatiivisten tarpeiden muuttuminen.
- Johtamisjärjestelmiin integroitavien muiden järjestelmien lisääntyminen ja päivityminen.
- Uusien tietoturvaauhkien ilmeneminen.

Yllämainituista kolme ensimmäistä ovat luonteeltaan hitaammin muuttuvia. Näitä muutoksia, niiden vaikutuksia ja seurauksia pystytään suunnittelemaan ja toteuttamaan suunnitelmallisesti. Järjestelmään nämä muutostarpeet yleensä näkyvät uusien toiminnallisuuksien kehittämisenä.

Tietoturvallisuus on kuitenkin elementti, jota on huomattavasti vaikeampi ennakoida. Uusia teknologioihin liittyviä haavoittuvuuksia ja hyödyntämismahdollisuuksia voi nousta hyvinkin nopeasti. Näihin uhkiin reagoiminen tulee olla nopeaa. Ilman etukäteen tehtävää suunnittelua ja varatutumista reagoiminen todennäköisesti ei tapahdu riittävän nopeasti.

2.5 Elinkaarikustannukset

Yksi tärkeä näkökulma ylläpidettävyydessä on kustannukset. Jotta johtamisjärjestelmät pysyvät käyttötarpeita vastaavina, pitää ylläpitomuutoksia tehdä. Näiden muutosten tekeminen maksaa rahaa. Elinkaarikustannuksiin pystytään vaikuttamaan merkittävästi erilaisilla valinnoilla, mutta joka tapauksessa ylläpidon suuruus alkuperäiseen hankintahintaan nähden on merkittävä. Esimerkiksi U.S Army Material Command (1972, 1-1) arvioi, että modernin aseiden tai asejärjestelmän elinkaarikustannukset ovat 3 – 20 kertaiset alkuperäiseen hankintahintaan nähden. Tämä vuonna 1972 kirjoitettu teos ei ole tiedon alkuperäinen lähde, vaan se perustuu koostettuun tietoon aiemmista tutkimuksista. Asiaa on siis tutkittu jo pitkään ja nykyaikana, kun järjestelmät muuttuvat yhä monimutkaisemmiksi, nämä kustannukset vain kasvavat.

Hyvänä esimerkkinä elinkaarikustannuksista voidaan käyttää Suomen puolustusvoimien HX-hanketta, jossa ollaan hankkimassa Suomeen nykyaikaisia monitoimihävittäjiä. Valtioneuvoston HX-hankkeen tarjouspyyntöihin liittyvässä tiedotteessa todetaan, että yksi viidestä valintaan vaikuttavasta kriteeristä on elinjaksokustannukset (2018, Puolustusministeriö). Tämä kuvastaa elinkaarikustannuksien merkitystä ja vaikuttavuutta, josta on käyty myös paljon keskustelua HX-hankkeen tiimoilta mediassa.

3 HAASTATTELUTUTKIMUS JOHTAMISJÄRJESTELMIEN YLLÄPIDETTÄVYYDEN MUODOSTUMISESTA

Yhtenä tiedonkeruumenetelmänä tässä opinnäytetyössä käytettiin asiantuntijahaastatteluita. Kaikilla haastatetuilla henkilöillä on pitkä kokemus puolustusvoimien käyttöön tarkoitetuista johtamisjärjestelmistä ja niiden elinkaaren eri vaiheista. Haastatteluita pidettiin yhteensä 6 kappaletta toukokuussa ja kesäkuussa 2017. Haastateltavat on kirjattu lähdeluetteloon.

3.1 Haastattelututkimus

Haastatteluiden pohjana käytettiin TAULUKKO 1 kysymyslistaa. Haastateltavia ei pyritty haastattelutilanteessa ohjaamaan sisällössä, vaan heidän annettiin kertoa tärkeiksi kokemiaan asioita. Näillä pyrittiin saamaan mahdollisimman luotettavia haastattelutoksia ja kartoittamaan haastatteluissa nousevia teemoja. Näin pystyttiin arvioimaan myös asioiden tärkeyttä ylläpidettävyyden kannalta, jos samoja asioita nousi esiin useammassa haastatteluissa.

TAULUKKO 1 Haastattelukysymykset

Numero	Kysymys
1.	Kuinka pitkä järjestelmän elinkaari pitää olla, että järjestelmän ylläpidettävyyteen kannattaa kiinnittää huomiota?
2.	Mistä osa-alueista sinun mielestä ylläpidettävyyys johtamisjärjestelmille muodostuu?
3.	Kuinka tärkeänä näet ylläpidettävyyden kannalta itse järjestelmään liittyvät asiat verrattuna järjestelmää ympäröiviin asioihin?
4.	Missä vaiheessa järjestelmän elinkaarta tehdyillä toimenpiteillä on suurin vaikutus ylläpidettävyyteen?
5.	Minkälainen vaikutus ohjelmistokehitysprosessilla on ylläpidettävyyteen?
6.	Kuinka suuri vaikutus ylläpidettävyydellä on järjestelmän elinkaarikustannuksiin?
7.	Tunnistetaanko se, että järjestelmä on pitkän elinkaaren järjestelmä, riittävän ajoissa?
8.	Mitkä ovat tärkeimpiä dokumentoitavia asioita järjestelmässä ylläpidettävyyden kannalta?
9.	Kuinka suuren rahoituksen järjestelmä vaatii, jotta ylläpitokyky saadaan säilytettyä?

3.2 Haastatteluiden havainnot

3.2.1 Haastattelu 1

Johtamisjärjestelmän elinkaaren tulee olla haastateltavan mukaan yli 5 vuotta, jotta järjestelmää kehittävien ihmisten näkökulmasta siihen kannattaa kiinnittää huomiota.

Haastateltava nosti esiin ylläpidettävyyden muodostumisessa asiakkaan tuntemista, jatkuvuutta järjestelmän kehittämisessä riittävällä volyymillä ja huoltovarmuutta. Tämän lisäksi osaamiseen liittyvissä asioissa nousivat esiin motivoituneet työntekijät, osaamisen ylläpitäminen, hiljaisen tiedon siirtyminen sekä riittävä työntekijöiden vaihtuvuus. Osaamisen johtamisen hyviksi todettuina menetelminä nostettiin esiin mentorointi ja tiimityöskentely. Näillä menetelmillä pystytään välttämään erikoisosaamisen liiallinen henki-

löityminen. Tämä voidaan nähdä myös osaamiseen liittyvänä riskienhallintana. Jos osaaminen polarisoituu liikaa, ei tiimillä ole varaa menettää erikoisosajaa, koska uuden henkilön osaamisen hankkiminen vie niin paljon aikaa ja resursseja.

Haastateltavan mielenkiintoinen näkökulma osaamisen hallintaan liittyen oli, että asiakas maksaa osaamisen ylläpitämisen tavalla tai toisella. Jos osaamista ylläpidetään tekemisen kautta, se maksaa. Toisaalta, jos osaamista ei ylläpidetä, sen uudelleen hankkiminen maksaa.

Ylläpidettävyyden muodostumisen merkittävydessä itse järjestelmän ja sitä ympäröivien asioiden välillä haastateltava näki, että molempien tulee olla riittävällä tasolla. Tärkeitä hänen mukaansa on tunnistaa, että asiakkaan lisäarvo tulee toimivasta johtamisjärjestelmästä, eikä sen mahdollistavasta ympärillä olevasta infrastruktuurista. Joissakin tapauksissa voidaan myös nähdä, että ylläpitokyky voi olla arvoketjun lopputulos.

Vesiputousmalli ohjelmistokehitysprosessina ei haastateltavan mukaan ole tehokkain johtamisjärjestelmän ylläpidossa. Määrittely ja suunnittelu tulee tehdä riittävällä tasolla, mutta ketteriä menetelmiä kannattaa hyödyntää. Järjestelmän tuntemus helpottaa muutosten vaikutuksen arvioinnissa.

Ylläpidettävyyden vaikutuksista elinkaarikustannuksiin on haastava arvioida, koska ne ovat hyvin riippuvaisia järjestelmästä.

3.2.2 Haastattelu 2

Haastateltavan mukaan ylläpidettävyyteen kannattaa alkaa kiinnittämään huomioita, kun järjestelmän elinkaari kestää useita vuosia. Puolustusvoimien käytössä olevien johtamisjärjestelmien elinkaari on yleensä n. 20 vuotta, joten elinkaareen tulee kiinnittää huomioita.

Ylläpidettävyyden muodostamisessa haastateltava nosti esiin järjestelmän hyvän arkkitehtuurin, dokumentoinnin, sopimusrakenteen ja riittävän vähäisen ja hallitun vaihtuvuuden henkilöstön osalta. Hallitulla vaihtuvuudella tarkoitettiin myös hankkivan organisaation henkilöitä.

Haastateltava näki ylläpidettävyyden muodostamisen kannalta järjestelmän tekniset ominaisuudet ja hyvän arkkitehtuurin huomattavasti tärkeämmiksi. Kiteytettynä kaikki muu on korvattavissa paitsi järjestelmä. Myös järjestelmä on korvattavissa, mutta silloin ei puhuta ylläpidosta.

Ylläpidettävyyteen pystyy haastateltavan mukaan vaikuttamaan parhaiten sen elinkaaren alussa. Näillä toimenpiteillä on merkittävä vaikutus elinkaarikustannuksiin.

Ylläpidettävyyden kannalta tärkeimmiksi dokumentoitaviksi asioiksi haastateltava nosti

- arkkitehtuurin,
- rajapinnat,
- kehitysympäristö ja sen pystytys,
- säännöllisesti tehtävät toimenpiteet,
- asennus- ja konfigurointiohjeet,
- ominaisuuksien kuvaukset,
- käyttöohjeet ja
- testausdokumentaatio riittävän korkealla tasolla.

Dokumentoinnin tasossa ja määrässä kannattaa olla kriittinen. Yleensä asioita tunnutaan haastateltavan mukaan ylidokumentoitavan järjestelmän rakentamisvaiheessa ja ylläpidossa.

Riittävän rahoituksen takaaminen ylläpidettävyyden näkökulmasta tarkoittaa, että järjestelmän kehittävässä ylläpidossa pitäisi pystyä pitämään vähintään 3-4 henkilöä täysipäiväisesti kiinni, jotta ylläpitokyky säilyy. Haastateltava myös korosti puolustusvoimille tärkeää huoltovarmuusnäkökulmaa, joka perustuu osaamiseen eli ihmisiin.

3.2.3 Haastattelu 3

Haastateltavan mukaan järjestelmän elinkaaren pituudella ei ole merkitystä siihen, kannattaako ylläpidettävyyttä erikseen huomioida, vaan sen tulisi olla aina mukana. Ylläpidettävyydellä edistetään myös järjestelmän ja sen komponenttien yleiskäyttöisyyttä ja uudelleenkäytettävyyttä.

Tärkeimpinä asioina ylläpidettävyyden muodostumisen kannalta haastateltava koki riittävän moduloinnin, rahoituksen riittävyyden sekä asiakashyödyn saavuttamisen.

Ohjelmistokehitysprosessin osalta haastateltava suositteli ylläpidon näkökulmasta ketteriä menetelmiä. Tärkeää on tekemisen tavasta riippumatta se, että vastuut on selkeästi määritelty. Vastuu yleensä kannetaan paremmin, kun se ei ole ulkoa annettu vaan itse otettu. Myös selkeät ja toimivat kehitystyönohjaamisen työkalut, kuten vikatietokannat ja tehtävienhallintatyökalut, ovat tärkeitä riippumatta tekemisen tavasta.

Haastateltavan mukaan pitkän elinkaaren tunnistaminen saattaa helposti unohtua ja se saatetaan jättää kehittäjien ammattitaidon varaan. Sisäisen omistajuuden muodostuminen on tärkeää. Järjestelmän tuotepäällikön vastuulla on huomioida ylläpidettävyyttä ja uudelleen käyttöä, koska se saattaa kehityksen aikana unohtua. Tämä saattaa tapahtua varsinkin, kun keskitytään pieniin yksityiskohtiin.

Ylläpidettävyyden näkökulmasta tärkeimmät dokumentoitavat asiat ovat

- toiminnallinen kuvaus,
- tekninen kuvaus,
- rajapintakuvaukset,
- käyttöohje, joka on yleensä myös sovelluksen online-ohje ja
- konfiguraationhallintadokumentaatio.

Dokumentaatiossa tulisi haastateltavan mukaan käyttää lisäarvoajattelua. Esimerkiksi lähdekoodin kommentoinnissa on turha kommentoida itse itsensä selittävät osuudet.

Ylläpidon rahoitus tulisi varmistaa siten, että siinä pystytään pitämään n. 5 hengen tiimi kiinni. Tämän rahoituksen tulisi olla jatkuvaa ja sen näkyvyys riittävän pitkä. Niukka rahoitus yleensä johtaa siihen, että järjestelmään muodostuu teknistä velkaa, kun uusien ominaisuuksien tekeminen priorisoituu korkeammalle.

Osaamisen johtamisen näkökulmasta haastateltava korosti vaihtuvuuden suunnittelua. Henkilöitä tulee pitkissä elinkaarissa vaihtumaan, joten tähän tulee varautua. Näitä varautumisen keinoja ovat esimerkiksi tehokas perehdytys ja mentorointi. Myös hyvät asiakassuhteet ja asiakkaan tunteminen helpottavat järjestelmän ylläpitämisessä.

3.2.4 Haastattelu 4

3 vuotta on haastateltavan mukaan järjestelmän elinkaaren pituus, jolloin ylläpidettävyys vaatii huomiointia. Käytännössä kaikki puolustusvoimien hankkimat johtamisjärjestelmät täyttävät tämän kriteerin.

Ylläpidettävyys muodostuu haastateltavan mukaan seuraavista osa-alueista:

- Henkilöstön osaamisen hallinta
 - Dokumentointi
 - Henkilöstön käytön suunnittelu
 - Perehdytys ja tiedon jakaminen
 - Ristiin käyttö
- Arkkitehtuurilliset ratkaisut
 - Arkkitehtuurin ja rakenteen kuvaus ja dokumentointi
 - Henkilöstön vaihtuvuuden vaikutukset arkkitehtuuriin ja sen kuvauksen tärkeys
 - Käyttötarve ja käyttötapa ja niiden muutokset elinkaaren aikana
- Asiakkaan tunteminen
 - Tavoitteiden tunteminen ja loppukäyttäjien tarpeet
 - Pitkät asiakassuhteet

Haastateltava näki, että ylläpidettävyyden muodostumisen kannalta tärkein asia on organisaation osaaminen, joka muodostuu ihmisten osaamisesta. Tekniset ongelmat tulevat esiin osaamisen kautta ja taustalla on yleensä jokin osaamispuute.

Ylläpidettävyyteen pystytään vaikuttamaan haastateltavan mukaan eniten heti järjestelmän rakentamisen jälkeen ennen kuin henkilöt siirtyvät uusiin projekteihin. Tällöin on tärkeää dokumentoida toteutetut ratkaisut ja lopputulos. Tärkeimmät dokumentoitavat asiat ovat

- arkkitehtuuri, johon kuuluvat myös selitykset siitä, miksi rakenne on tietynlainen,
- konfigurointiohjeet,
- määrittelyt, jossa muuttuva maailma korostuu ja
- komponenttien kuvaukset

Dokumenteissa on haastateltavan mukaan hyvä näkyä jäljitettävyysetketju vaatimuksesta todentamiseen. Dokumentoinnin tasossa pitää huomioida tiedon vastaanottaja, mutta yleisesti elinkaaren alkuvaiheessa asioita on syytä dokumentoida tarkemmin. Dokumentaation suurin hyöty on, että kirjoittaja itse muistaa asian paremmin.

Ylläpidon volyymin tulee haastateltavan mukaan olla vähintään olla yksi henkilötyövuosi vuosittain. Tähän tulisi liittyä myös vuosittainen version toimittaminen. Tässä korostuu myös kehittävän ylläpidon tärkeys.

3.2.5 Haastattelu 5

Puolustusvoimien käyttöön tarkoitettujen johtamisjärjestelmien elinkaaren pituus on tavoitteellisesti haastateltavan mukaan aina yli 10 vuotta. Elinkaari muodostuu neljästä vaiheesta, jotka ovat hankinta-, rakentamis-, käyttöönottovaihe ja operatiivinen käyttö sekä ylläpito.

Ylläpidettävyys johtamisjärjestelmiin muodostuu haastateltavan mukaan osaamisen hallinnasta, rahoituksen takaamisesta, toimittajan tiloista ja toimintaedellytyksistä sekä palvelunäkökulmasta.

Ylläpidettävyyden muodostamisen kannalta järjestelmän arkkitehtuuri ja teknologiavaihto eivät korostu järjestelmän tekemisen mahdollistavista asioista. Nämä asiat elävät yhdessä ja ilman toista ei ole toista. Rahoitus kuitenkin on tässä tärkeä asia, koska ilman sitä ei ole mitään muuta.

Haastateltavan mukaan järjestelmän elinkaaren alkuvaiheessa pystytään eniten vaikuttamaan ylläpidettävyyteen. Se, että tunnistetaan ylläpitotarve riittävän ajoissa, kuitenkin vaihtelee. Tässä nostettiin esiin myös kiireen vaikutus. Jos järjestelmän rakentamisessa on aina kiire, ei välttämättä muisteta tai ehditä ottaa ylläpidettävyyttä huomioon riittävästi.

Käytännön toimenpiteinä ylläpidettävyyden rakentamisessa haastateltava nosti esiin ketterän kehitysmallin, arkkitehtuurin ja rajapintojen säännöllisen katselmoinnin, testauksen

ja testausautomaation tärkeyden, dokumentoinnin ja ylläpidettävyyden näkyvillä pitämisen myös asiakkaan suuntaan.

Tärkeimpinä dokumentoitavina asioina haastateltava nosti arkkitehtuurin, keskeisen suunnitteluratkaisut, toiminnallisen näkökulman ja rajapintakuvaukset. Dokumentaatiossa pitää myös huomioida ylläpidettävyys, eli itse dokumentaation päivittäminen olla helposti päivitettävää.

Osaamisen hallinnassa haastateltava nosti esiin mentoroinnin, riittävän vaihtuvuuden tiimien välillä ja resursoinnin suunnittelun.

3.2.6 Haastattelu 6

Haastateltavan mukaan ylläpidettävyyteen kannattaa kiinnittää huomioita, kun järjestelmän odotettu elinkaari on yli 2 vuotta. Laajoissa järjestelmissä pelkästään rakentamisvaihe saattaa kestää vuosia.

Haastateltava nosti esiin seuraavat osa-alueet ylläpidettävyyden muodostumisessa.

- Dokumentointi
- Lähdekoodin tulee olla rakenteeltaan selkeää ja hyvin kommentoitua
- Koodikatselmoinnit
- Kestävät tekniset ratkaisut
- Standardoidut ratkaisut rajapinnoissa ja esittämistavoissa.
- Edellä mainittujen osaamisnäkökulma, koska käytössä myös monessa muussa paikassa
- Yhteiskäyttöiset komponentit
- Kaupalliset valmiskomponentit ja Open Source -komponenttien elinkaaren ylläpitävyys
- Järjestelmän parametrintimahdollisuus
- Toimittajan ymmärrys loppukäyttäjän käyttöympäristöstä ja -tavasta
- Käyttäjien ideat
- Asiakkaan hankintamallit

Kokonaisuudessa ylläpidettävyyden muodostumisessa haastateltava näki tärkeimpänä teknisesti toimivan järjestelmän. Myös osaamisenäkökulma on kuitenkin tärkeä.

Haastateltava korosti, että ylläpidettävyyden rakentaminen on jatkumo. Käyttöönoton jälkeen tehdyillä toimenpiteillä voidaan kuitenkin vaikuttaa eniten ylläpidettävyyteen. Liika toiminnan ketteryys missä tahansa vaiheessa voi kostautua ylläpidettävyydessä. Liialla ketteryydellä haastateltava tarkoitti, että dokumentointia ei tehdä riittävästi tai, että jälkikäteen ei ymmärretä ratkaisuja ja toiminnallisuuksia.

Tärkeimpinä dokumentoitavina asioina haastateltava nosti toiminnalliset määrittelyt, rajapintakuvaukset ja käyttöohjeet.

Ylläpitokyvyn säilymisen kannalta tulisi tekemisessä olla kiinni 3 -10 henkilöä. Vaikka järjestelmään ei kehitettäisi uusia ominaisuuksia tulisi olla vähintään kolme henkilöä sidottu ylläpitoon.

3.3 Yhteenveto

Tähän lukuun on koostettu yhteenveto haastattelukysymyksien vastauksista. Asian jäsentämiseksi jokaiselle haastattelukysymykselle on kirjattu oma yhteenveto.

Yleisesti vastauksissa nousi samoja teemoja, eikä varsinaisia ristiriitoja asioiden välillä näkynyt muutamia poikkeuksia lukuun ottamatta.

3.3.1 Kuinka pitkä järjestelmän elinkaari pitää olla, että järjestelmän ylläpidettävyyteen kannattaa kiinnittää huomiota?

Vastaukset kysymykseen olivat saman suuntaisia. Yksi haastateltava esitti näkemyksen, että elinkaaren oletetulla pituudella ei ole merkitystä, siihen pitääkö ylläpidettävyyttä huomioida. Keskimäärin vastaukset olivat 2 ja 5 vuoden välillä.

Kun huomioidaan, että järjestelmän elinkaari pitää sisällään suunnittelun, rakentamisen, käyttöönoton ja ylläpidon, tämä aika on todella lyhyt. Käytännössä lopputuloksena voidaan todeta, että kaikissa johtamisjärjestelmissä ylläpidettävyys tulee ottaa huomioon.

3.3.2 Mistä osa-alueista sinun mielestä ylläpidettävyys johtamisjärjestelmille muodostuu?

Vastauksissa nousi esiin erilaisia ylläpidettävyiden muodostumiseen vaikuttavia osa-alueita. Monissa vastauksissa nostettiin esiin hyvin samanlaisia osa-alueita. Näitä olivat esimerkiksi järjestelmän hyvä arkkitehtuuri, selkeät ja kestävät tekniset ratkaisut, dokumentointi, henkilöstön osaamisen hallinta, rahoituksen takaaminen, toimivat rahoitusmallit, tilat sekä muut toimintaedellytykset, asiakkaan ja loppukäyttäjän tunteminen sekä ymmärrys käyttöympäristöstä ja –tavasta.

Vastaukset pystyy ryhmittelemään kolmeen eri luokkaan.

1. Järjestelmään itseensä liittyvät asiat, kuten arkkitehtuuri, tekniset ratkaisut, rajapinnat ja niiden dokumentointi.
2. Järjestelmää kehittäviin ihmisiin liittyvät, kuten kyky ymmärtää teknisiä ratkaisuja, kyky perehdyttää ihmisiä, mentorointi ja osaamisen johtaminen.
3. Rahoituksen takaaminen ylläolevien asioiden tekemisen mahdollistamiseksi.

3.3.3 Kuinka tärkeänä näet ylläpidettävyiden kannalta itse järjestelmään liittyvät asiat verrattuna järjestelmää ympäröiviin asioihin?

Tämä kysymyksen vastauksissa oli eniten hajontaa. Kysymyksen asettelu todennäköisesti on myös vaikuttanut vastauksiin. Kysymyksessä tarjotaan käytännössä kahta eri vaihtoehtoa vastaukseksi. Kahdessa vastauksessa ei annettu suoraan vastausta siitä, kumpi on tärkeämpää, vaan korostettiin asioiden yhteisvaikutusta ja kokonaisuutta. Kahdessa vastauksessa taas korostettiin teknisesti toimivan järjestelmän merkitystä ja yhdessä vastauksessa järjestelmää kehittävien henkilöiden ja organisaation osaaminen nostettiin tärkeimmäksi.

3.3.4 Missä vaiheessa järjestelmän elinkaarta tehdyillä toimenpiteillä on suurin vaikutus ylläpidettävyyteen?

Kaksi ajankohtaa nousi esiin kysymyksen vastauksissa.

- Järjestelmän elinkaaren alkuvaihe, jolloin järjestelmää suunnitellaan ja tehdään teknisiä ratkaisuja.
- Heti rakentamisvaiheen jälkeinen aika, jolloin järjestelmää otetaan käyttöön.

Järjestelmän elinkaaren alkuvaihe on ymmärrettävä ylläpidettävyyden merkittävyyden kannalta. Asioiden suunnittelulla pystytään vaikuttamaan merkittävästi lopputulokseen.

Jälkimmäisen ajankohdan ajatus ylläpidettävyyden kannalta oli se, että järjestelmän rakentamisen jälkeen siihen osallistuneet henkilöt ovat vielä projektin käytössä ja heillä on ylläpidettävyyden kannalta tärkeät asiat vielä tuoreessa muistissa. Tämä on ajankohta, jolloin tulisi dokumentoida ylläpidon kannalta tärkeät asiat.

3.3.5 Minkälainen vaikutus ohjelmistokehitysprosessilla on ylläpidettävyyteen?

Suurin osa haastateltavista ymmärsi kysymyksen siten, että minkälainen ohjelmistokehitysprosessi on hyvä ylläpidettävyyden kannalta ja vastasivat siihen. Asiaa olisi voinut tarkastella hieman laajemminkin siten, että mitenkä erilaiset ohjelmistokehitysprosessit vaikuttavat ylläpidettävyyteen.

Kaikissa vastauksissa ketterät ohjelmistokehitysprosessit nähtiin perinteisiä vesiputousmalleja paremmaksi. Määrittelyn ja suunnittelun tärkeyttä kuitenkin korostettiin. Ketterällä kehittämisellä koettiin saavutettavan nopeammin ratkaisu asiakastarpeisiin.

Oma tulkintani, miksi ketterät ohjelmistokehitysprosessit koettiin paremmiksi ylläpidossa, on, että työn ohjaus tilaajan suunnasta voi olla dynaamisempaa. Uusiin tarpeisiin pystytään vastaamaan nopeammin, koska työjonoa käydään tilaajan kanssa aktiivisesti läpi. Työjonon sisällä asioita voidaan priorisoida siten, että tärkeäksi koettuja ylläpitotoimenpiteitä saadaan tarvittaessa nopeasti toteutukseen.

3.3.6 Kuinka suuri vaikutus ylläpidettävyydellä on järjestelmän elinkaarikustannuksiin?

Ylläpidettävyydellä nähtiin olevan merkittäviä vaikutuksia elinkaarikustannuksiin. Määrällisiä vastauksia kysymykseen ei kukaan antanut. Myöskään sen suurempaa analysointia vaikutuksista ei vastauksissa ilmennyt. Vastauksista ei pysty johtamaan suurempia johtopäätöksiä kuin, että haastateltavat kokivat ylläpidettävyydellä olevan vaikutusta elinkaarikustannuksiin.

3.3.7 Tunnistetaanko se, että järjestelmä on pitkän elinkaaren järjestelmä, riittävän ajoissa?

Vastauksien perusteella pitkän elinkaaren tunnistaminen tehtiin yleensä liian myöhään. Järjestelmän rakentaminen on saatettu jo käynnistää, kun elinkaaren hallintaa, ylläpidettävyyttä ja siihen liittyviä asioita aloitetaan pohtimaan. Perusteluna asialle oli, että ylläpidettävyys tulisi olla osana kaikkea tekemistä. Tämä on sinänsä totta, mutta lopputuloksen kannalta saavutetaan varmasti parempia tuloksia, kun asiaa suunnitellaan. Kun asiaa ei tehdä suunnitelmallisesti se helposti jää yksittäisten henkilöiden varaan, jolloin riskit asian unohtumiseen ja huomiotta jättämiseen ovat suuremmat.

3.3.8 Mitkä ovat tärkeimpiä dokumentoitavia asioita järjestelmässä ylläpidettävyyden kannalta?

Tärkeimpinä dokumentoitavina asioina, jotka nousivat melkein kaikissa vastauksissa, olivat järjestelmän toiminnallinen kuvaus, arkkitehtuuri ja rajapintakuvaukset ulkopuolisiin järjestelmiin. Muita esille nostettuja dokumentoitavia asioita olivat konfiguraationhallintadokumentaatio, testausdokumentaatio, käyttö- ja konfigurointiohjeistus.

Tärkeää on muistaa, että dokumentaatio ei ole ikinä itseisarvo. Dokumentaation tarkoitus on välittää tietoa ja tiedon tarve riippuu vastaanottajasta. Yleensä johtamisjärjestelmän toimittajaa kiinnostaa enemmän, miten ja jokin asia on toteutettu, ja käyttäjää, miten ja tämä asia toimii.

Dokumentaatio toimii myös tärkeänä perehdytysmateriaalina uusille työntekijöille. Jos dokumentaatio on hyvin kirjoitettua ja se sisältää ajantasaista tietoa, on siitä suuri lisäapu perehtymisessä.

Ylläpidettävyyden näkökulmasta yksi tärkeä haastatteluissa nostettu asia oli dokumentaation ylläpidettävyys. Dokumentaatiota kirjoittaessa pitäisi miettiä miten itse dokumentaatio pidetään ajantasaisena. Tämä vaikuttaa myös dokumentaation tarkkuustasoon. Hyvin tarkasti dokumentoidut tekniset kuvaukset pitää päivittää joka kerta, kun järjestelmään tehdään pienikin muutos. Toisaalta taas liian korkealla abstraktiotasolla tehty tai liian vähäinen dokumentaatio ei välttämättä sisällä ylläpitävän toteutustyön kannalta riittävää tietoa tulevaisuudessa tehtäviä ylläpitotöitä varten. Dokumentoinnin tasossa pitää löytää sopiva taso, jossa se kuvaa riittävästi asiaa lisäarvoajattelun perusteella mutta sen ylläpito ei ole vielä liian työlästä.

Haastatteluissa nousi dokumentoinnin tasosta vastakkaisia mielipiteitä esiin. Osa haastattavista korosti dokumentoinnin tärkeyttä ja osa varoitti ylidokumentoinnin riskistä. Uskon, että tähän vaikuttaa haastateltavien historia. Jos on ollut mukana ylläpitämässä järjestelmää, joka on dokumentoitu heikosti tai vastakkaisesti todella tarkasti, vaikuttaa tämä varmasti näkökulmaan aiheesta.

3.3.9 Kuinka suuren rahoituksen järjestelmä vaatii, jotta ylläpitokyky saadaan säilytettyä?

Vastaukset rahoituksen suuruudesta vastattiin kaikissa haastatteluissa henkilömäärän kautta. Kiinteitä kuluja tiloista, laitteista tai vastaavista asioista ei huomioitu missään vastauksessa. Aktiivisesti järjestelmää ylläpitävä henkilömäärä, jolla ylläpitokyky säilytetään, oli keskimäärin 3-5 hengen välillä. Tällä henkilömäärällä nähtiin, että järjestelmää pystytään tekemään kokoajan pientä jatkokehitystä. Tästä jatkokehittävästä ylläpidosta käytettiin termiä kehittävä ylläpito. Ylläpitokyvyssä pitää huomioida henkilöstön osaaminen, kehitys- ja integrointityökalut sekä todentamisympäristöt ja -laitteet. Jos jokin näistä asioista jätetään pitkäksi ajaksi käyttämättä, katoaa niihin liittyvä tieto. Fyysiset laitteet saattavat myös ajan kuluessa rikkoutua.

Rahoituksen suuruuden lisäksi sen ennustettavuus ja tasaisuus nostettiin esiin. Jos vo-lyymi vaihtelee paljon, tulee siitä ongelmia. Työntekijöitä ei saada pidettyä pitkäjänteisesti samassa sisällössä kiinni. Tämä aiheuttaa tiedon katoamista ja ylläpidettävyyden heikkenemistä.

Ylläpitotöiden rahoitusmalliksi suositeltiin tuntilaskutusperusteista laskutusmallia. Tähän tietysti saattaa vaikuttaa se, että kaikki haastateltavat edustavat toimittajaa. Tuntilaskutusperusteisessa työssä riskin kantaa täysin tilaaja. Toisaalta ylläpitoluonteisessa työssä riskit ovat muutenkin hyvin pienet, jonka seurauksena tilaajan ei tarvitse erikseen maksaa toimittajalle riskin kantamisesta. Tällöin tilanteesta hyötyvät molemmat.

4 YLLÄPIDETTÄVYYDEN MUODOSTAMINEN

Tässä luvussa on syvennetty haastatteluiden lopputuloksina saatuihin osa-alueisiin, joiden kautta ylläpidettävyys muodostuu pitkän elinkaaren johtamisjärjestelmiin. Osa-alueet on jaettu teemojen mukaisesti omiin lukuihinsa.

4.1 Kehittävä ylläpito

Haastatteluissa noussut termi kehittävästä ylläpidosta on asia, joka vaikuttaa kaikkiin osa-alueisiin. Kehittävällä ylläpidolla tarkoitetaan ylläpitotyötä, jossa ylläpidettävään järjestelmään kehitetään uutta toiminnallisuutta. Kehittävän ylläpidon hyödyt ovat laajat. Uuden toiminnallisuuden rakentaminen olemassa olevaan järjestelmään vaatii samoja asioita, joista itse ylläpitokyky muodostuu. Myös työntekijöiden motivaatio pidemmällä aikavälillä pysyy parempana, kun järjestelmään kehitetään uusia ominaisuuksia. Jotta järjestelmään pystytään vaivattomasti rakentamaan uusia ominaisuuksia, tarvitaan

- osaavia henkilöitä,
- ymmärrys järjestelmän arkkitehtuurista ja toiminnasta,
- rahoitus työn tekemiselle ja
- kehitykseen tarvittavat työkalut ja todentamisympäristöt.

Kehittävä ylläpito voidaan mieltää ylläpitokyvyn säilyttämiseksi kriittisempiä ylläpito-tarpeita varten. Kehittävän ylläpidon sisältö itsessään ei välttämättä on niin tärkeää, mutta sen kautta voidaan varmistaa tarvittavan osaamisen ja järjestelyiden säilyminen näitä kriittisiä tarpeita varten.

Kehittävä ylläpito on myös tilaajalle varma tapa sitoa riittävä määrä resursseja ylläpitokyvyn säilyttämiseen. Toimittaja tekee resurssoinnin suunnittelun pohjautuen tilauskantaan ja jatkonäkymiin. Kehittävän ylläpidon tekeminen vaatii, että tilaaja hankkii tarvittavan rahoituksen työn tekemiselle.

Yksittäisenä asiana ylläpidettävyiden muodostumisen ja säilymisen kannalta kehittävän ylläpidon rakentaminen on se tärkein asia. Kehittävän ylläpidon kautta tulee huomioitua samat asiat, jotka vaikuttavat ylläpitokyvyn muodostumiseen.

4.2 Johtamisjärjestelmän toteutustapa ja lopputulos

Tässä luvussa ja sen aliluvuissa tutustutaan johtamisjärjestelmien toteutustavan ja lopputuloksien vaikutuksiin ylläpidettävyyteen. Tarkoituksena on esitellä asioita, jotka vaikuttavat ylläpidettävyyden muodostumiseen kokonaisuuden kannalta.

4.2.1 Elinkaaren suunnittelu ja hallinta

Hyvin suunniteltu on puoliksi tehty. Tämä vanha sanonta pätee myös johtamisjärjestelmien elinkaariin. Elinkaaren suunnittelu tulee tehdä ennen kuin järjestelmän rakentaminen aloitetaan. Kun järjestelmän rakentaminen on aloitettu, on jo tehty monta päätöstä, joilla on vaikutusta elinkaareen. Järjestelmän elinkaaren suunnittelussa tulee vastata seuraaviin asioihin.

- Kuinka pitkä on järjestelmän oletettu elinkaari?
- Kuinka muuttuvassa ympäristössä järjestelmää tullaan käyttämään?
- Kuinka todennäköisesti järjestelmään tulee kohdistumaan uusia vaatimuksia rakentamisen jälkeen?
- Kuinka merkittäviä nämä vaatimukset voivat olla?
- Kuinka todennäköisesti järjestelmän käyttötarkoitus tulee muuttumaan?

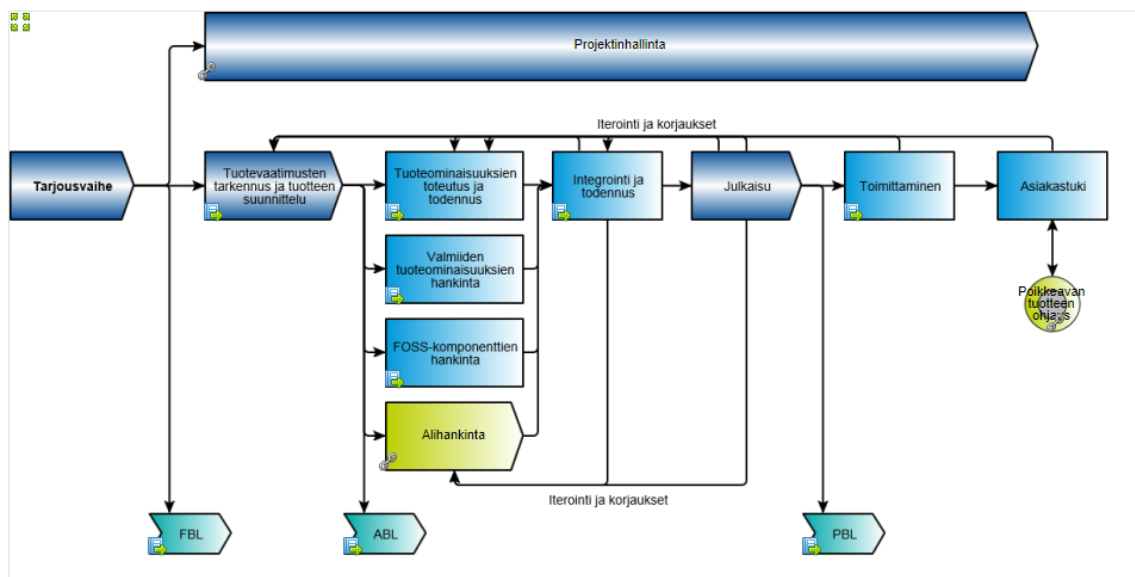
Näihin kysymyksiin vastaamalla saadaan lähtötiedot, jotka elinkaaren suunnittelussa tulee ottaa huomioon. Elinkaarisuunnitelma on hyvä käydä tilaajan kanssa läpi.

Järjestelmän elinkaaren pituus määrittelee monta asiaa ylläpidettävyyden näkökulmasta. Mitä pidempi elinkaari tulee olemaan sitä todennäköisempää on, että järjestelmään tulee kohdistumaan muutostarpeita. Elinkaaren pituus myös vaikuttaa muutostarpeiden toteutukseen helppoutteen teknologisesta näkökulmasta sekä osaamisen näkökulmasta. Lyhyessä elinkaaressa esimerkiksi osaamisen hallintaan ei tarvitse kiinnittää niin paljon huomiota, koska on todennäköisesti rakentamiseen osallistuvat henkilöt ovat käytettävissä kokoajan.

4.2.2 Prosessit ja laatu

Erilaisten prosessien noudattamisen vaikutukset ylläpidettävyyteen riippuvat täysin prosessin sisällöstä ja sen laadusta. Hyvä prosessi tukee ja auttaa muistamaan tärkeät asiat, mutta huono prosessi pakottaa tekemään vääriä asioita. Johtamisjärjestelmien kehityksessä on monia prosesseja. Omia prosesseja voi olla esimerkiksi ohjelmistokehitykselle, vaatimustenhallinnalle, konfiguraationhallinnalle, projektinhallinnalle, asiakirjojenhallinnalle, riskienhallinnalle, alihankinnanhallinnalle, resurssienhallinnalle, myynnille ja erilaisille talouden osa-alueille.

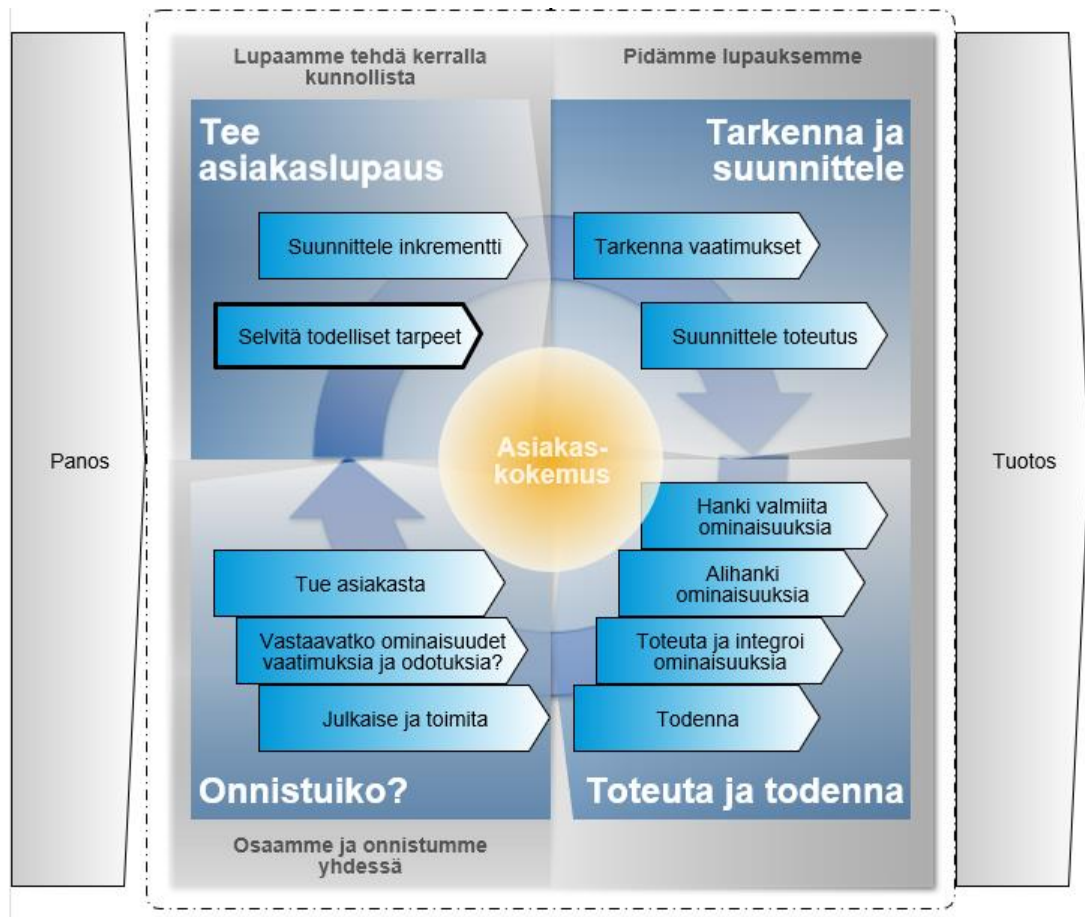
Tekemisen kuvaaminen korkealla tasolla on hyödyllistä. Tästä korkeasta tasosta voidaan johtaa aliprosesseja, joissa voidaan keskittyä yksityiskohtaisesti sisältöön tarkemmin. Insta DefSec Oy:n tietojärjestelmäprojektin läpiviennin prosessi on kuvattu kuvassa Kuva 1.



Kuva 1 Tietojärjestelmäprojektin läpiviennin prosessikuvaus (Insta DefSec Oy, 2018)

Insta DefSec Oy:n tietojärjestelmäprojektin läpiviennin prosessissa on huomioitu puolustusteollisuudelle tärkeiden AQAP2110- ja AQAP2210-laatustandardien vaatimusten noudattaminen. Laatu on aina tavoitteena rakentaa osaksi tekemistä ja sen prosesseja, koska sen jälkikäteen päälle liimaaminen ei ole mahdollista. Laadulla taas on huomattava merkitys ylläpidettävyyden kannalta, koska laatu määritelmänsä mukaisesti mittaa sitä, missä määrin lopputuloksen luontaiset ominaisuudet täyttävät vaatimukset (Anttila & Jussila, 2016).

Yksi ylläpidettävyyteen vaikuttava johtamisjärjestelmän prosessi on ohjelmistokehitysprosessi ja sen aliprosessit. Ohjelmistokehitysprosessilla pystytään vaikuttamana teknisten ratkaisujen toteutustapaan, joilla oli haastatteluidenkin perusteella suuri merkitys johtamisjärjestelmän ylläpidettävyyteen. Insta DefSec Oy:n ohjelmistokehitysprosessi on kuvattu kuvassa Kuva 2.



Kuva 2 Ohjelmistokehitysprosessi korkealla tasolla (Insta DefSec Oy, 2018)

Ohjelmistokehitysprosessi on tärkeä ylläpidettävyyden kannalta, koska niihin tulisi olla rakennettuna sisään tärkeät tukiprosessit. Näitä tukiprosesseja ovat konfiguraationhallinta, vaatimustenhallinta, katselmointikäytännöt ja laadunvarmistus.

Ohjelmistokehitysprosessiin kannattaa määritellä katselmointikäytännöt sisään. TAU-LUKKO 2 esittää katselmointien esimerkkirakenteen. Katselmointien tavoite on tukea kehitystyötä ja varmistaa, että kaikki tarvittavat asiat tulee hoidettua. Näillä tuetaan myös ylläpidettävyyden muodostumista, koska niiden kautta käsitellään samoja asioita, joista ylläpidettävyyden muodostuu teknisestä näkökulmasta muodostuu.

TAULUKKO 2 Esimerkkirakenne katselmoinneista

Katselmus	Tarkoitus	Huomioitavat asiat
Software Requirement Review (SRR)	Katselmoidaan sovelluksen vaatimukset ja niiden soveltuvuus.	Vaatimuksissa tulee huomioida toiminnallinen soveltuvuus, suorituskyky, yhteensopivuus, käytettävyys, luotettavuus, tietoturva, ylläpidettävyys ja siirrettävyys.
Preliminary Design Review (PDR)	Tavoitteena on katselmoida sovelluksen tuoterakenne, arkkitehtuuri, rajapinnat, toiminnalliset kuvaukset ja tekniset ratkaisut.	Katselmus pidetään ennen toteuttamisen aloittamista. Katselmuksen lopputuloksena muodostetaan kokonaisuuden ABL, eli Allocated Baseline.
Critical Design Review (CDR)	Tarkoituksena on käydä läpi, että onko toteutuksessa tullut vastaan asioita, joita ei PDR-katselmuksessa ja suunnitelmissa osattu ennustaa.	Katselmus pidetään, kun toteutus on ollut hetken aikaa käynnissä.
Test Readiness Review (TRR)	Katselmoidaan todentamiseen toteuttamiseen liittyvät asiat, kuten testiresurssit, -ympäristöt ja todentamisen sisältö. Samalla katselmoidaan myös toteutuksen nykytilanne vaatimuksia vasten. Tarkoituksena on varmistaa, että toteutus täyttää kaikki vaatimukset.	Testausvalmiuden katselmus, joka pidetään toteutuksen loppuvaiheessa.
Release Readiness Review (RRR)	Toimitusvalmiuden katselmointi, jossa tarkastetaan, että kaikki sovitut asiat on tehty, laatu järjestelmän mukaiset aiemmat katselmukset on pidetty ja hyväksyntä kriteerit täyttyvät.	Pidetään ennen toimitusta.

4.2.3 Järjestelmän arkkitehtuuri

Järjestelmän tekninen arkkitehtuuri oli yksi asia, joka nousi haastatteluissa esiin yhtenä tärkeimpänä asiana ylläpidettävyyden kannalta. Järjestelmän teknisellä arkkitehtuurilla nähtiin olevan merkittävä vaikutus ylläpidettävyyden muodostumiseen.

Ennen aiheesta jatkamista on hyvä pysähtyä hetkeksi määrittelemään termit, koska arkkitehtuuriin liittyy niitä monta.

- Järjestelmän tekninen arkkitehtuuri kuvaa järjestelmän tekniset ratkaisut korkealla tasolla, suunnittelumallit sekä -tekniikat.
- Ohjelmistoarkkitehtuuri kuvaa järjestelmän erilaiset komponentit, niiden muodostaman rakenteen ja komponenttien väliset rajapinnat. Ohjelmistoarkkitehtuuria voidaan kuvata esimerkiksi kerroksittain. Ohjelmistoarkkitehtuuri ottaa kantaa esimerkiksi tietovarastojen käyttöön sekä käyttöliittymän ja sovelluslogiikan väliseen kommunikointiin.
- Järjestelmäarkkitehtuuri kuvaa korkeammalla tasolla järjestelmän sijoittumista osaksi laajempaa kokonaisuutta. Se kuvaa myös operatiivisesta näkökulmasta liittymistä muihin kokonaisjärjestelmän osiin.

Järjestelmän tekninen arkkitehtuuri ja ohjelmistoarkkitehtuuri suunnitellaan johtamisjärjestelmälle ennen sen rakentamista. Tärkeä näkökulma arkkitehtuureissa on, että myös niitä tulee pystyä ylläpitämään. Kaikkia tulevaisuuden tarpeita ja muutoksia ei pystytä ennustamaan myöskään arkkitehtuurin osalta. Tämän vuoksi järjestelmän arkkitehtuurin pitäisi olla rakenteelta semmoinen, että sen on mahdollista elää ja muuttua järjestelmän elinkaaren mukana. Järjestelmän arkkitehtuurin suunnitteluun kannattaa ottaa myös tilaaja mukaan, koska tilaajalla on todennäköisesti ajatuksia järjestelmän tulevaisuuden muutostarpeista. Nämä ovat tärkeitä lähtötietoja, joita voidaan hyödyntää järjestelmän arkkitehtuurin vaatimuksien määrittelyssä. Arkkitehtuurin laatuattribuutteina seuraavat asiat tulisi aina huomioida

- luotettavuus
- suorituskky,
- skaalautuvuus,
- käytettävyys,
- muunneltavuus ja laajennettavuus,
- saatavuus ja
- integroitavuus.

Järjestelmän arkkitehtuurin kuvaamiseen löytyy monia menetelmiä. Yksi menetelmä on Philippe Kruchtenin 4+1-malli, jossa asiaa katsotaan useammasta rinnakkaisesta näkökulmasta. Mallissa huomioidaan ohjelmistojärjestelmän looginen näkymä, kehitysnäkymä, prosessinäkymä, sijoitusnäkymä ja skenaarionäkymä. (Kruchten, 1995, 42-50). Toinen esimerkki arkkitehtuuriin kuvaamiseen on TOGAF, joka on lyhenne sanoista The Open Group Architecture Framework. TOGAF tarkastelee arkkitehtuuria myös useasta eri näkökulmasta. Tässä mukana tarkastelussa on myös esimerkiksi liiketoiminnan näkökulma.

Avainsanoja järjestelmän ohjelmistoarkkitehtuurissa ylläpidettävyyden kannalta ovat modulaarisuus, osuuksien abstrahointi ja komponenttien tiedonvaihto. Tällä mahdollistetaan muutosten rajaaminen sekä komponenttien lisääminen, poistaminen tai korvaaminen muuttamatta ympärillä olevaa. Modularisoinnilla pystytään tekemään laajoista kokonaisuuksista hallittavia ja mahdollistetaan asioiden tekemisen rinnakkaisuus. Kuvaava termi tälle on Plug In –tyyppinen laajennettava arkkitehtuuri. Ian Gorton määrittelee komponenttien tiedon vaihdon menetelmät termillä Middleware (Gorton, 2011, 39). Arkkitehtuuri on siis eräänlainen kehys, joka määrittelee rakenteen, mekanismit ja työkalut erilaisten komponenttien erotteluun sekä näiden välisen kommunikoinnin mahdollistamiseen.

Tekniikoita komponenttien välisen tiedon vaihtamiseen on erilaisia, joista osa on synkronisia ja osa asynkronisia. Tiedonvaihtomekanismin valintaan vaikuttaa järjestelmän vaatimukset ja tarpeet. Gorton listaa seuraavat yleisimmät tiedonvaihtomekanismit

- distributed objects,
- message-oriented middleware (MOM),
- publish-subscribe,
- application servers ja
- message brokers (Gorton, 2011, 41-60 ja 81-86).

Komponenttien modularisointiin ja sen hallintaan löytyy nykyään paljon valmiita ja toimivia työkaluja. Näitä mekanismeja ei siis kannata rakentaa itse, ellei sille ole perusteltua syytä. Yhtenä esimerkkinä näistä voidaan nostaa OSGi ja sen laajennos DOSGi, joka on lyhenne sanoista Distributed Open Services Gateway Initiative. OSGin kehitys on aloitettu 2000-luvun alussa ja nykyään se on hyvin laajassa käytössä erityisesti Java-kehityskielen kanssa.

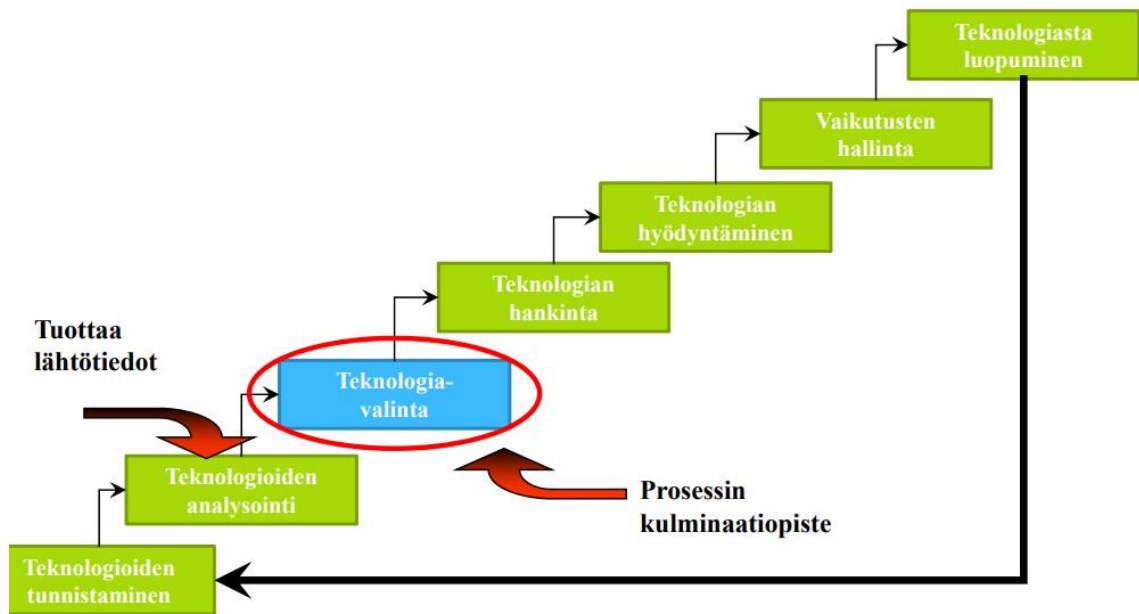
Yksi modularisoinnin hyöty on muutoksien rajaaminen. Kun toteutus on jaettu komponentteihin, niihin liittyy myös rajapinta, joka vastaa komponentin kommunikoinnista muiden komponenttien kanssa. Jos komponentin toimintaan kohdistuu muutostarve, se voidaan ja tavoitellaan toteutettavaksi ilman muutoksia rajapintaan. Tällöin komponenttiin tehty muutos ei näy muille komponentille, eli muutos saadaan rajattua vain tiettyyn paikkaan. Järjestelmän sisäiset rajapinnat tulee olla hyvin verisoitu ja dokumentoitu, jotta niiden käyttö on hallittua.

Modulaarinen ohjelmistoarkkitehtuuri myös mahdollistaa komponenttien irrottamisen järjestelmästä osaksi muita kokonaisuuksia. Tämä liittyy luvussa 4.2.5 käsiteltäviin FOSS- ja yhteiskäyttökomponentteihin. FOSS on lyhenne sanoista Free and Open-Source Software.

Modulaarinen järjestelmän ohjelmistoarkkitehtuuri linkittyy myös järjestelmän konfiguraationhallintaan ja ne tukevat toisiaan. Komponentteja pystytään käsittelemään omina konfiguraatioyksikköinä teknisesti ja konfiguraationhallinnassa. Järjestelmä muodostuu komponenteista, joilla on omat versionsa ja tunnistettu konfiguraatio.

4.2.4 Teknologiavalinnat

Johtamisjärjestelmien ylläpidettävyyden kannalta on suuri merkitys sillä, miten teknologiavalintoja tehdään. Pitkistä elinkaarista johtuen huonojen valintojen kanssa joudutaan pahimmillaan elämään hyvin pitkään. Johtamisjärjestelmäkehityksessä teknologiavalinta voi olla esimerkiksi toteutuskielen, sovellusalustan tai kartta-alustan valinta. Kontekstista riippumatta teknologiavalintojen tekemiseen ja teknologioiden elinkaareen liittyy tiettyjä vaiheita, joita on kuvattu Marko Seppäsen piirtämässä kuvassa Kuva 3.

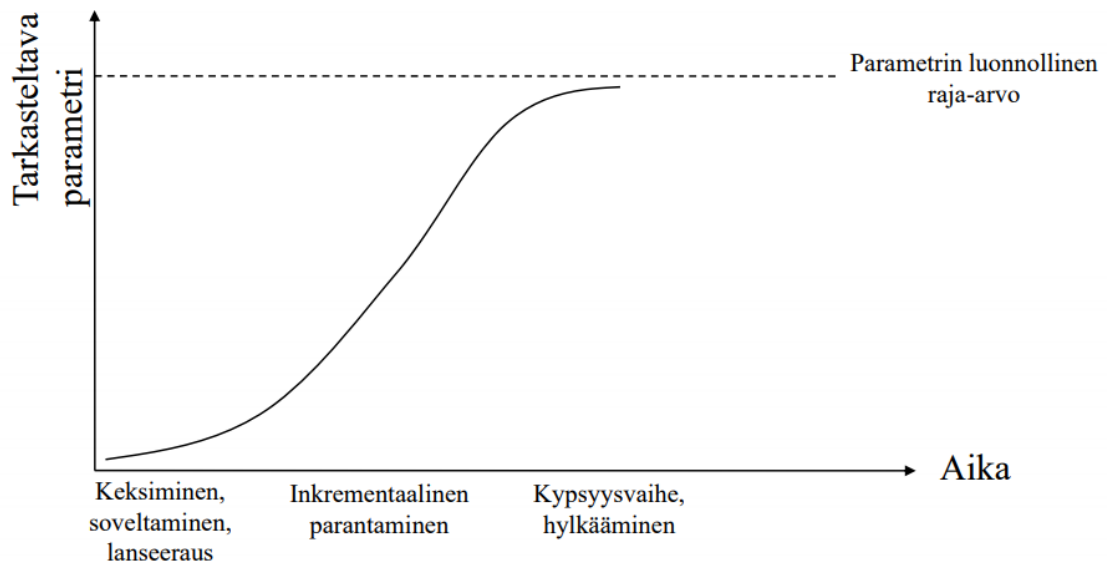


Kuva 3 Teknolojiajohtamisen prosessi (Seppänen, 2016)

Oleellista teknologiavalinnoissa ja niiden elinkaareissa on jatkuvuus. Johtamisjärjestelmien sisällä tämä vastaava jatkuvuus pitää olla olemassa huomioiden, että johtamisjärjestelmän elinkaari jatkuu yksittäisiä teknologioita pidempään.

Johtamisjärjestelmien pitkästä elinkaaresta johtuen järjestelmän arkkitehtuuri tulee rakentaa siten, että yksittäisiä teknologioita on mahdollista vaihtaa myöhemmin. Kaikkein tämä ei ole mahdollista, mutta se on tavoiteltavaa.

Teknologiavalintoihin johtamisjärjestelmien pitkä elinkaari myös vaikuttaa siten, että valittujen teknologioiden pitää olla stabiileja ja pysyä käytössä vielä riittävän pitkään. Stabiiliudella tarkoitetaan sitä, että sillä on riittävän vakiintunut paikka toimialueella ja se ei ole virhealtis. Jos teknologialla ei ole vakiintunutta paikkaa, on riskinä sen katoaminen kokonaan. Tämä on tietojärjestelmäkehityksessä todellinen riski esimerkiksi FOSS-komponenttien kanssa, joiden muutos on nopeaa. Fosterin vuonna 1986 esittelemän teknologioiden elinkaaren S-käyrällä tämä tarkoittaa, että paras hetki valita teknologia pitkän elinkaaren johtamisjärjestelmään, on heti nopean muutoksen jälkeinen kypsyysvaihe. Tällöin teknologia ei enää merkittävästi muutu ja sen asema kentässä on vakiintunut. Elinkaarta teknologialla on kuitenkin vielä jäljellä ennen kuin uusi teknologia korvaa sen. Kuva 4 esittää teknologian elinkaaren geneeristä S-käyrää.



Kuva 4 Teknologian elinkaaren geneerinen S-käyrä (Seppänen, 2016)

Valittujen teknologioiden elinkaaren jatkuminen on tärkeää myös tietoturvanäkökulmasta. Vaikka valittu teknologia täyttäisi kaikki sille asetetut toiminnalliset tarpeet, saattaa syntyä tarve päivittää teknologiaa hyödyntävästä komponentista käyttöön uudempi versio tietoturvan vuoksi. Näitä päivityksiä ei pystytä todennäköisesti itse tekemään, vaan niissä ollaan alkuperäisen toteuttajan varassa. Jos kyseessä on avoimen lähdekoodin komponentti, on näiden päivitysten tekeminen kehityksestä vastaavan yhteisön vastuulla. Avoimen lähdekoodin komponenttien kanssa on tietysti aina mahdollista itse toteuttaa muutokset. Pahimmillaan tietoturvapoikkeaman korjaamatta jättäminen kolmannen osapuolen komponentissa voi tarkoittaa, että koko komponentista joudutaan luopumaan etuajassa. Tämä taas aiheuttaa kustannuksia, joita ei elinkaarisuunnittelussa ole otettu huomioon.

4.2.5 FOSS- ja yhteiskäyttökomentit

Kaikkea ei aina tarvitse tai kannata tehdä alusta loppuun itse. Sovelluskehityksessä on paljon yhteisiä yleisiä tarpeita, joita on ratkaistu aiemmin. FOSS on lyhenne sanoista free and open-source software. Hyviä esimerkkejä yhteisestä tarpeista johtamisjärjestelmäkehityksessä ovat kartta ja tiedon esittäminen kartalla.

FOSS-komponentit ovat avoimia kaikille halukkaille käyttäjille ja kehittäjille. Tämän lisäksi organisaatioilla voi olla sisäisiä yhteiskäyttökomentteja, joita ne hyödyntävät sisäisesti erilaisten järjestelmien ohjelmistokehityksessä.

FOSS- ja yhteiskäyttökomponenttien ensimmäinen hyöty perustuu siihen, että integroimalla valmis komponentti osaksi järjestelmää, säästetään aikaa ja rahaa. Tämä ei kuitenkaan ole ainoa hyöty, vaan näiden komponenttien käytössä on myös ylläpidollinen hyötynäkökulma. Kun samaa komponenttia käytetään monessa erilaisessa järjestelmässä ja sillä on laaja käyttäjäkunta, sen kehitys on myös aktiivista. Käyttäjät tuottavat uusia ominaisuuksia komponenttiin, joita voi hyödyntää omassa järjestelmässä. Komponentin laatu ja tietoturva on todennäköisesti parempi. Kun käyttäjiä on paljon, viat ja tietoturvaongelmat havaitaan nopeammin.

Toimittajan kannalta FOSS- ja yhteiskäyttökomponentteja voidaan hyödyntää uusien järjestelmien rakentamisessa. Olemassa olevista komponenteista pystytään rakentamaan uuteen käyttötarkoitukseen uusi järjestelmä hyvin kustannustehokkaasta.

4.2.6 Rajapinnat

Tässä luvussa rajapinnoilla tarkoitetaan järjestelmän ulkoisia rajapintoja, joilla liitytään ja välitetään tietoa toisten sovellusten tai järjestelmien kanssa. Järjestelmän sisällä on myös erilaisten komponenttien välillä hyvin paljon erilaisia rajapintoja, jotka liittyvät luvussa 4.2.3 kuvattuihin arkkitehtuureihin..

Rajapintojen suunnittelu ja toteuttaminen perustuu aina johonkin tarpeeseen. Tuo tarve ja käyttötapa on tärkeää huomioida rajapinnan valinnassa. Tarve ja käyttötapa vaikuttavat oleellisesti rajapinnan toiminnallisiin ja ei-toiminnallisiin vaatimuksiin. Puolustusvoimien johtamisjärjestelmissä rajapinnan käyttötarkoitus voi olla esimerkiksi hyötytiedon vaihto eri järjestelmien välillä tai järjestelmän tilan seuranta ja lokitiedon hallinta SIEM-järjestelmällä. SIEM on lyhenne sanoista Security Information and Event Management.

Tietosisällön vaihtamiseen tarkoitetuissa rajapinnoissa kannattaa aina suosia standardoituja rajapintoja. Samaa standardoitua rajapintaa pystyy todennäköisesti käyttämään useampaan eri tarpeeseen, eli useampaan eri järjestelmään liittymiseen. Tämä vähentää huomattavasti ylläpitotarvetta, kun jokaista uutta liittyvää järjestelmää varten ei välttämättä tarvitse rakentaa omaa rajapintaa. Standardoitujen rajapintojen ylläpitäminen on kustan-

nustehokkaampaa. Ylläpitotarpeista tehdyistä muutoksista, jotka voivat liittyä toiminnallisuuteen, suorituskykyyn tai virhekorjauksiin, hyötyvät kaikki erilaiset rajapinnan käyttötavat.

Toinen selkeä hyöty standardoitujen rajapintojen käytössä on se, että myös itse rajapintaa jatkokehitetään ja ylläpidetään. Standardirajapintoja käyttää yleensä laaja osajoukko erilaisia järjestelmiä, joilla on yhteisiä tiedonvaihtotarpeita. Erilaisilta järjestelmiltä tulee erilaisia uusia tarpeita tai käytännön tulkintaongelmia rajapinnan käyttötavoista. Käytännön hyöty, joka tästä ylläpidettävyyden kannalta saavutetaan, on kustannustehokkuus. Samoja monimutkaisia ongelmia ei tarvitse aina itse ratkaista, koska joku muu on jo aiemmin ongelmaan törmättyään saattanut ratkaista sen. Toisaalta, jos ongelman ratkaisee itse, hyötyy myös joku muu siitä. Yhteensopivuus, joka on rajapinnan yksi tärkeimmistä ominaisuuksista, on myös selkeä saavutettava etu. Riippuen standardin laatijasta ja ylläpitäjästä, näitä uusia tarpeita tai tulkintaongelmia ratkaistaan erilaisilla tavoilla. Käsittelyta-voissa on eroja. Teollisuuden toimijoiden ylläpitämät teknologiastandardit ja niiden käsittely eroavat merkittävästi esimerkiksi sotilas- tai ilmailustandardeista. Yleinen nyrkkisääntö on, että mitä turvallisuuskriittisempi standardi on kyseessä, sitä hitaampaa sen muuttuminen myös on.

4.2.7 Laadunvarmistus ja todentaminen

Laadunvarmistuksella ja todentamisella ei ole suoraa vaikutusta ylläpidettävyyden muodostumiseen. Aihe liittyy kuitenkin oleellisesti huoltovarmuuteen, joka on käyttäjän näkökulmasta tärkein perustelu ylläpidettävyydelle. Tämän takia aihe on nostettu käsiteltäväksi yhtenä osuutena.

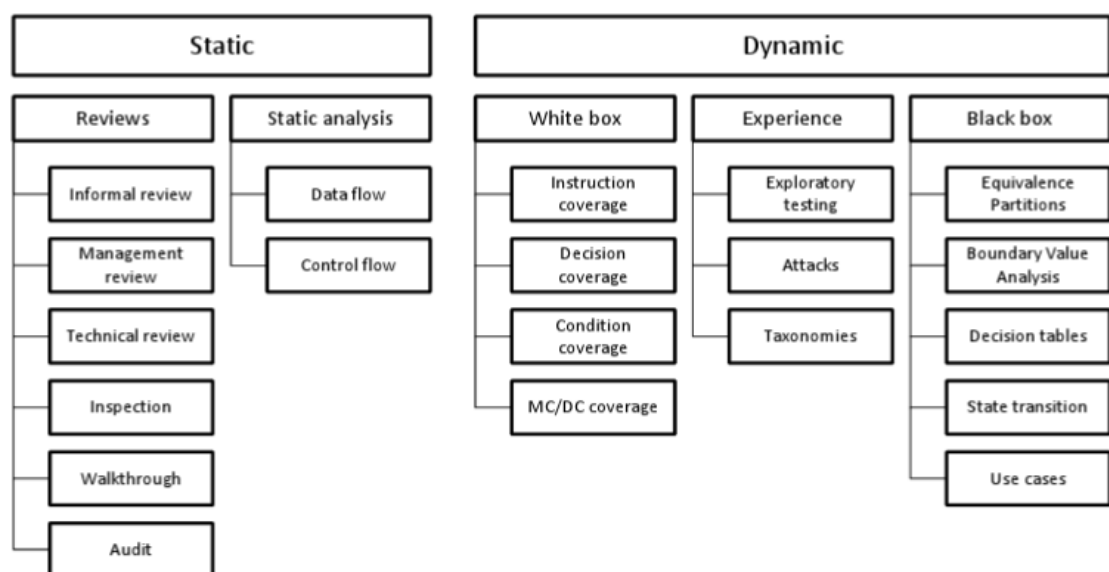
Laadunvarmistuksella ja todentamisella on ylläpidettävyyden kannalta merkitystä sille, kuinka paljon ylläpitotarpeita järjestelmään tulee. Kattavasti rakennusvaiheessa todennetusta johtamisjärjestelmästä löytyy käytössä todennäköisemmin vähemmän virheitä ja sen toiminta on etukäteen paremmin tiedossa. Johtamisjärjestelmien todentaminen ei ole pelkästään ylläpidettävyyden kannalta tärkeää, vaan myös muista näkökulmista. Hyvin todennettu ja toimiva johtamisjärjestelmä tuo oman osansa huoltovarmuuteen, koska ylläpitotarve vähenee myös kriisioloissa. Kriisioloissa on erityisen tärkeää pitää järjestelmä

käytössä ja täyttämässä sille asetettua operatiivista tarvetta. Todentamisella pystytään vähentämään riskiä, että näin ei käy.

Testaamisessa tulee muistaa, että testaaminen ei takaa virheetöntä sovellusta tai järjestelmää, koska kaiken kattavan testaamisen suorittaminen on käytännössä mahdotonta (2013, Homes, 11). Uusia ylläpitotarpeita saattaa siis tulla esiin virheiden muodossa myös kattavasti testatusta järjestelmästä.

Toinen näkökulma ylläpidettävyyteen ja testaamiseen liittyy muutoksien vaikutuksien todentamiseen. Kun järjestelmään toteutetaan jostakin tarpeesta muutos, tulee todentaa, että se ei riko mitään muuta. Tällä tarkoitetaan regressiotestaamista. Regressiotestaus keskittyy järjestelmän kannalta riskialttiisiin ja kriittisiin toiminnallisiin. Regressiotestauksessa tulee huomioida sen helppous ja nopeus. Tästä syystä johtamisjärjestelmien regressiotestauksessa suositellaan erityisesti automaatiotestausta. Regressiotestaus on itseään toistavaa ja samantapaista joka kerta. Tämä sopii siis hyvin koneen tehtäväksi. Lisäksi johtamisjärjestelmien kasvaessa tulee jossain vaiheessa piste vastaan, jolloin manuaalisesti kaiken testaaminen ei ole kustannustehokasta.

Johtamisjärjestelmän ja sen ominaisuuksien todentamisesta voidaan tehdä monella eri tavalla. Todentamisen mallit voidaan jakaa staattisiin ja dynaamisiin, kuten kuvassa Kuva 5.



Kuva 5 Erilaisia testausmenetelmiä (Homes, 2013, 143)

Staattisissa menetelmissä analysoidaan ja katselmoidaan lähdekoodia ja komponentteja. Näihin menetelmiin löytyy nykyään monia automatiikkaa hyödyntäviä työkaluja. Yksi esimerkki on SonarQube, joka tarjoaa kattavan valikoiman erilaisia staattisen analyysin työkaluja. Nämä pystyy liittämään osaksi sovelluksen integrointia, jolloin ne ajetaan automaattisesti, kun sovellus integroidaan.

Dynaamisessa todentamisessa testataan käännettyä ja integroitua ohjelmistojärjestelmää. Dynaamiseen testaamiseen löytyy monia eri tapoja ja menetelmiä. Aiemmin testaus on nähty vaiheena, joka suoritetaan järjestelmän rakentamisen jälkeen. Nykyään testausta kuitenkin tehdään aktiivisesti kehityksen aikana, kuten Kuva 6 kiteyttää. Testauksen tavoitteena on tukea kehitystyötä, eikä pelkästään verifioida lopputuloksia. Tämä perustuu näkemykseen, että mitä aikaisemmin vika havaitaan, sen halvempaa sen korjaaminen on.



Kuva 6 Testaus osana kehitystä (Craig yms. 2002, 11)

Johtamisjärjestelmän todentamiseen liittyy oleellisesti loppukäyttäjien tekemä testaus ja koekäyttö. Toimittajan tekemä todentaminen ei ole riittävää, koska sitä ei lähtökohtaisesti päästä tekemään lopullisessa käyttöympäristössä. Puolustusvoimien tapauksessa johtamisjärjestelmien käyttöönottoon liittyy myös suuri määrä teknisiä hyväksyntöjä ja erilaisia testauksia, joita toimittaja ei voi tehdä. Loppukäyttäjien testaamiseen ja niistä saatuihin havaintoihin liittyvät toimintatavat on syytä suunnitella. Tämän prosessin määrittely ja automatisointi mahdollisimman pitkälle tukee toimittajaa, tilaajaa sekä loppukäyttäjää. Sillä on myös merkittävä vaikutus ylläpidettävyyden kannalta, koska oikein rakennettuna se tukee järjestelmän jatkuvaa kehitystä. Testaukseen yleensä eivät osallistu toimittajan edustajat, koska tavoitteena on samalla verifioida operatiivista käyttötapaa ja harjoitella uusien ominaisuuksien käyttöä. Tästä testaus- ja kehitysprosessista tulisi löytyä seuraavia elementtejä.

- Jos testauksessa havaitaan sovellusvikoja tai käyttötapaan liittyviä vikoja, tulee näistä kerätä tarkat tiedot talteen. Näitä tietoja ovat esimerkiksi vian esiehdot, kuvaus, lopputulos, sovelluslokit ja sessiotallenteet. Näiden tietojen kerääminen kannattaa automatisoida työkalun tehtäväksi.
- Kaikki havaitut viat käsitellään harjoituksen jälkeen toimittajan kanssa. Vikojen esiintyminen ja vaikutukset loppukäyttäjään käydään läpi yhdessä. Tämän käsittelyn perusteella viat priorisoidaan tärkeysjärjestykseen.
- Toimittaja korjaa viat niille asetetussa tärkeysjärjestyksessä ja raportoi etenemisestä tilaajalle.
- Vikojen korjaamisen jälkeen muutokset tulee verifioida samassa ympäristössä, jossa ne havaittiin. Tästä testaamisesta taas vastaavat loppukäyttäjät. Testauksen seurauksena saattaa tulla uusia havaintoja, jotka käsitellään vastaavalla tavalla. Kyseessä on siis jatkuvaa toimintaa.

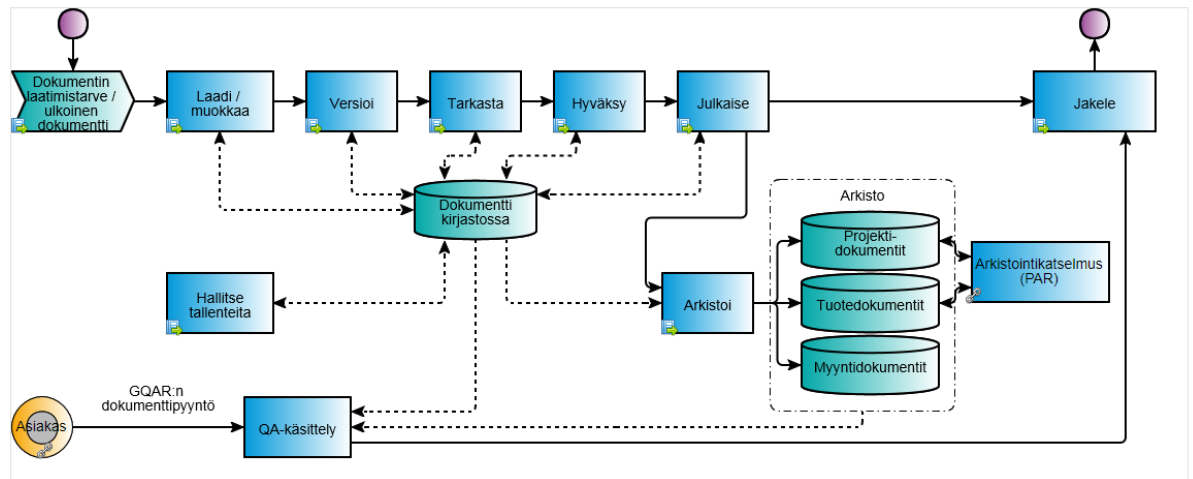
4.2.8 Dokumentointi

Järjestelmän dokumentaatiolla pystytään vaikuttamaan merkittävästi ylläpidettävyyteen. Tässä luvussa dokumentaatiolla käsitetään varsinaiseen järjestelmään liittyvää tuotedokumentaatiota, jossa kuvataan järjestelmän toimintaan ja käyttöön liittyviä asioita. Järjestelmiin yleensä liittyy paljon muuta dokumentaatio, jota on tuotettu rakentamisvaiheessa.

Järjestelmän tuotedokumentaation tulisi pitää sisällään seuraavia asioita.

- Konfiguraationhallintadokumentaatio
- Katselmus- ja auditointipöytäkirjat
- Järjestelmäkuvaus
- Käyttöohjeet
- Järjestelmälle asetetut toiminnalliset ja ei-toiminnalliset vaatimukset
- Toiminnallinen kuvaus järjestelmän käyttämisestä
- Tekniset kuvaukset arkkitehtuurista ja toteutusratkaisuksista
- Rajapintakuvaukset ulkoisista ja sisäistä rajapinnoista
- Järjestelmän vaatimuksien todentamisen toteutustapa ja todentamisen lopputulokset
- Tunnetut poikkeamat ja rajoitteet järjestelmän käytössä

Dokumentaation eli asiakirjojen hallintaan kannattaa määrittää oma prosessi, jotta siihen liittyvät asiat tehdään aina samalla tavalla. Tällä pystytään vaikuttamaan dokumentaation laatuun, eli lopputuloksen ennustettavuuteen. Kuvassa Kuva 7 on esitelty Insta DefSec Oy:n asiakirjahallinnan prosessi.



Kuva 7 Asiakirjahallinnan prosessi (Insta DefSec Oy, 2018)

Asiakirjahallinnan prosessin tärkein tehtävä on varmistaa, että asiakirjoja käsitellään, muokataan ja arkistoidaan oikein. Tällä pyritään varmistamaan, että dokumentaation sisältö on validia, katselmoitua ja arkistoitu sovitulla tavalla. Ylläpidettävyyden kannalta asiakirjahallinnalla on merkitystä, koska se on yksi tapa varmistaa tiedon saatavuus jälkikäteen.

Järjestelmän dokumentaatio linkittyy ylläpidettävyyden näkökulmasta myös osaamisen hallintaan. Dokumentaatio on yleensä asia, johon henkilö ensimmäisenä tutustuu, kun hän haluaa perehtyä järjestelmään. Lukijoina dokumentaatiolle voi käyttäjien lisäksi olla järjestelmän ylläpitämiseen osallistuva henkilö.

Yleinen kysymys, joka dokumentaatioon liittyy, on kenelle sitä kirjoitetaan. Riippuen, kuinka dokumentaation kirjoittaja tähän kysymykseen vastaa, vaikuttaa se dokumentaation sisältöön. Itselle kirjoitettu dokumentaatio yleensä ei ole niin tarkkaa. Kirjoittajalla ja tässä tapauksessa lukijalla on laaja pohjatietämys asiasta ja hän pystyy tekemään paljon oletuksia pohjatiedon perusteella. Kun dokumentaatiota kirjoitetaan jollekin muulle, pitää kirjoittajan ensimmäisenä tehdä oletus lukijan pohjatiedoista. Tämä oletus vaikuttaa siihen, kuinka tarkalla tasolla asioita pitää kuvata.

Liian tarkan ja yksityiskohtaisen dokumentaation ylläpitäminen on työlästä. Jos dokumentaation ylläpitäminen on liian työlästä, on suurempi riskin, että sinne jää vanhaa tai virheellistä tietoa. Dokumentaation ei siis kannata olla liian tarkkaa.

4.3 Osaamisen ylläpitäminen

Johtamisjärjestelmien ylläpitämiseen tarvitaan osaavia ihmisiä. Näillä ihmisillä tulee olla riittävä teknologinen osaaminen, toimialaosaaaminen sekä ymmärrys järjestelmän toteutustavasta ja arkkitehtuurista. Kun ylläpidettävyys määritellään tehokkuutena tehdä tarvittavia muutoksia järjestelmään, korostuu näiden muutoksien tekijöiden kyky tehdä muutoksia. Osaamisen ylläpitäminen on näin ollen pakollinen asia, jotta ylläpitokyvystä voidaan puhua.

Osaamisen johtaminen ja siihen liittyvät asiat korostuvat, kun johtamisjärjestelmien oletettu elinkaari on pitkä. Tällöin tulee väistämättä henkilöstön vaihtuvuutta vastaan. Joissakin harvoissa tapauksissa saattaa olla tilanne, että yksittäinen henkilö on koko pitkän elinkaaren ajan mukana järjestelmän kehittämisessä ja ylläpitämisessä. Useimmiten näin ei kuitenkaan ole. Henkilöstön vaihtumiseen tulee varautua.

4.3.1 Tehokas perehdytys

Vaihtuvuuden seurauksena tulee olla tehokkaat menetelmät uusien työntekijöiden perehdyttämiselle. Mitä nopeammin työntekijä pystytään perehdyttämään järjestelmän kehittämiseen tai testaamiseen sitä helpompaa myös resurssoinnin suunnittelu on. Jos järjestelmän kehittämiseen perehdyttäminen on nopeaa ja helppoa, pystytään uusia työntekijöitä tarvittaessa ottamaan lyhyemmällä varoitusaajalla. Tällä helpotetaan toimittajan organisaation sisällä henkilöstön liikkuvuutta ja se mahdollistaa kehityksen volyymin skaalauksen tarpeen mukaan. Perehdytyskyky vaikuttaa siis suoraan myös ylläpidettävyuteen.

Perehdyttämisen tehokkuutta on helppo mitata. Kuinka kauan kestää henkilön perehdytyksen aloittamisesta siihen, kun henkilö kykenee itsenäisesti toteuttamaan uusia ominaisuuksia tai korjaamaan virheitä itsenäisesti?

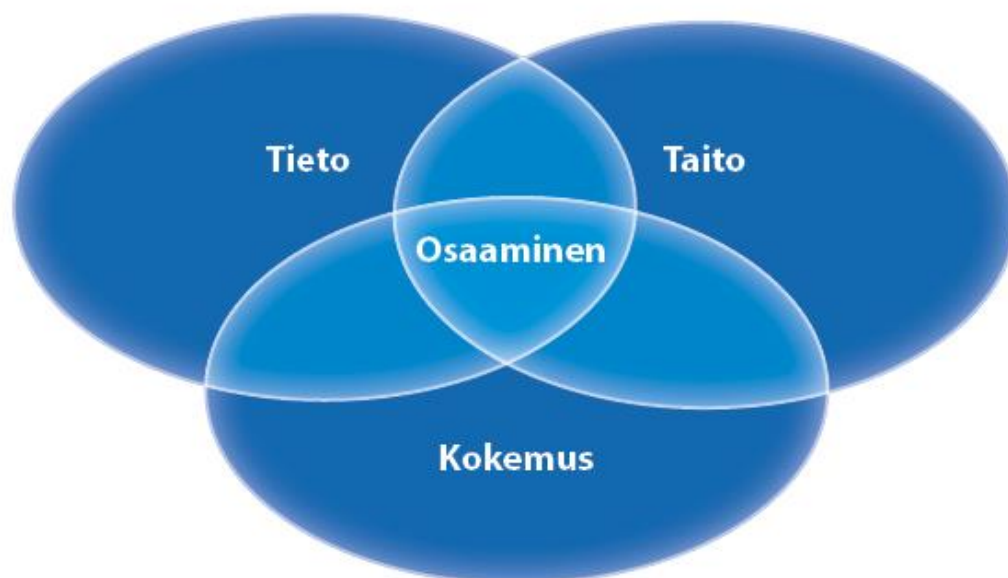
Tehokas perehdytys vaatii valmistautumista. Perehdytykseen pystyy varautumaan

- ylläpitämällä ajantasaisia perehdytysmateriaaleja,
- automatisoimalla kehitysympäristöjen rakentamisen ja
- hyödyntämällä mentorointia.

Mentorointi on tehokas tapa nopeuttaa perehdytystä. Puhtaalla perehdytysmateriaalilla kukaan ei opi tuntemaan järjestelmää, vaan se vaatii omatoimista tutustumista ja tekemistä. Helpoiten tämä onnistuu tavoitteiden kautta. Tällöin on hyvä, että on nimetty kokenempi henkilö, joka pystyy auttamaan ongelmatilanteissa.

4.3.2 Osaamisen johtaminen

Osaaminen muodostuu tiedosta, taidosta ja kokemuksesta (Kuva 8). Sumkin ja Tuomi korostavat osaamisen johtamisessa strategian mukaisia toimia. Osaamisen johtamisen tulisi aina toteuttaa strategiaa, ja se on esimiehen tärkein tehtävä. (Sumkin & Tuomi, 2012).



Kuva 8 Osaamisen osa-alueet (Sumkin & Tuomi, 2012)

Kuvassa Kuva 9 esitetään, kuinka organisaation osaaminen muodostuu sen yksilöiden ja ryhmien osaamisen yhdistelmästä (Sumkin & Tuomi, 2012). Organisaatio osaaminen on tärkeä käsite ylläpidettävyyden näkökulmasta. Osaamisen ei voi olla pelkästään yksilöllä, jotta se tukee ylläpidettävyyden muodostumista.



Kuva 9 Organisaation osaaminen (Sumkin & Tuomi, 2012)

Liika osaamisen keskittyminen yksiköille on ylläpidettävyyden kannalta huono asia riskienhallinnan näkökulmasta. Jos kyseinen henkilö jää pois tiimin käytöstä jostakin syystä, katoaa kaikki hänellä ollut osaaminen samalla. Tämän takia ylläpidettävyyden näkökulmasta pienin osaamisen yksikkö tulisi olla tiimi. Käytännössä tämä tarkoittaa, että tiimin sisällä osaamista kannattaa jakaa useammalle henkilölle. Osaamisen jakaminen onnistuu helpoiten tekemisen kautta. Tämä pätee uuteen ja vanhaan osaamiseen. Kun yhdellä henkilöllä on jokin osaaminen, sen jakaminen toisille onnistuu tekemällä töitä osaavan henkilön kanssa. Kokeneempi henkilö pystyy työn ohella opastamaan ja neuvomaan asiassa. Tärkeä huomioitava asia on hiljaisen tiedon välittyminen. Hiljaista tietoa ei pysty välittämään millään muulla tavalla kuin kanssakäymisen kautta. Täysin uuden osaamisen hankkiminen onnistuu myös tehokkaammin, kun henkilöitä on useampi kuin yksi. Henkilöt pystyvät opetteluun yhteydessä käymään asiaa keskenään läpi ja vaihtamaan ajatuksia toistensa kanssa. Tämä tukee kaikkien oppimista oikein toteutettuna. Yhdessä tekeminen asettaa vaatimuksia ylläpidosta vastaavan tiimin kokoon. Tiimin pitää olla riittävän iso, jotta tämä on mahdollista.

Sumkin ja Tuomi kirjoittavat, että ”Organisaation kehittämisen kannalta on keskeistä, että vuorovaikutus ja viestintä on yksilöiden, ryhmien ja organisaation osien välillä riittävää ja luonteeltaan kannustavaa.” (Sumkin & Tuomi, 2012). Tämä on erityisen tärkeää tiedonvaihtumisen kannalta. Ilman vuorovaikutusta osaaminen ei siirry tiimin sisällä tai

tiimien välillä. Tästä näkökulmasta töiden organisointi kannattaa tehdä siten, että kukaan ei työskentele liian pitkään yksin minkään osa-alueen kanssa. Kun osuuksia toteuttaa useampi henkilö, tarttuu jo tekemisen aikana osaaminen useammalle. Tähän, kun yhdistetään vuorovaikutus ja keskustelu henkilöiden välillä, on lopputulos parempi.

4.3.3 Motivaatio

Työntekijöiden motivointi on monesta syystä tärkeää. Ylläpidettävyyden kannalta motiivoinnin merkitys korostuu siinä, että motivoituneet työntekijät pysyvät todennäköisemmin käytettävissä eivätkä esimerkiksi vaihda työpaikkaa. Motivoitunut työntekijä tekee myös työnsä tehokkaammin ja muodostaa ympärilleen kannustavaa työilmapiiriä

Edward Decin itseohjautuvuusteorian mukaan ihmisten motivaation voi jakaa ulkoiseen ja sisäiseen motivaatioon. Ulkoinen motivaatio muodostuu ulkoisista tekijöistä, kuten muiden odotuksista, uskosta palkitsemiseen tai rankaisemiseen. Perinteinen ulkoisen motivaation esimerkki on työstä saatu palkkio, eli raha. Sisäinen motivaatio on taas ihmisestä itsestään lähtöisin. Asiasta innostunut ihminen on motivoitunut tekemään asioita, koska esimerkiksi nauttii niiden tekemisestä. (Deci, 2000).

Ylläpidettävyyden kannalta sisäisen motivaation luominen on avainasemassa. Pelkällä ulkoisilla motiivointikeinolla lopputulokset eivät ole kestäviä ja ylläpidettävyyden kannalta nimenomaan nämä pitkäkestoiset vaikutukset ovat merkittäviä. Decin mukaan tärkeimmät sisäsyntyiset motivaation lähteet ovat autonomisuus (autonomy), taito (mastery) ja merkittävyys (purpose) (Deci, 2000). Näistä sisäisen motivaation lähteistä johdettuna johtamisjärjestelmien ylläpidettävyyden kannalta seuraavat asiat vaikuttavat motivaation syntymiseen ja säilymiseen.

- Työn merkittävyys on helppo perustella. Johtamisjärjestelmiä käytetään olemassa olevan yhteiskuntarakenteen turvaamiseen ja säilyttämiseen. Niiden avulla pystytään kriisiaikoina pelastamaan ihmishenkiä, koska ne tarjoavat arvokasta lisätietoa ja mahdollistavat reaaliaikaisen johtamisen epävarmoissa olosuhteissa.
- Autonomisuuden saavuttaminen osaksi johtamisjärjestelmäkehittämistä ja – ylläpitoa onnistuu helpoiten tavoitteiden kautta johtamisella. Selkeiden tavoitteiden määrittäminen ja vapauden antaminen riittävässä määrin toteutustapaan motivoi.

Tavoitteiden kautta johtamiseen liittyy oleellisesti myös palautteenanto ja palkitseminen, jotka ovat pakollisia toiminnassa.

- Yksilön, tiimin ja organisaation osaamisen ja taidon korostaminen on myös hyvä motivointitapa. Toimialue ei ole kovin yleinen tai laaja, joten sen uniikkiutta pystyy korostamaan. ”Harva pystyy siihen mitä me teemme ja missä me olemme hyviä”.

4.4 Johtamisjärjestelmien ylläpidon rahoitus

Jotta johtamisjärjestelmien ylläpitäminen on toimittajan puolesta mahdollista, tulee sillä olla riittävä rahoitus. Yleisimmin puolustusvoimien käytössä olevat johtamisjärjestelmät ovat teollisten toimittajien rakentamia. Yritysten perimmäinen tavoite on tehdä kannattavaa liiketoimintaa ja tuottaa omistajilleen lisäarvoa. Toimittajan näkökulmasta riittävän suuri, tasainen ja ennustettava rahoitus järjestelmän ylläpidolle on toivottava, koska se helpottaa resurssoinnin suunnittelussa ja toiminnan ennustamisessa. Ilman riittävää rahoitusta ylläpitokyky heikkenee, koska toimittaja ei välttämättä sido riittävästi resursseja työhön.

4.4.1 Ylläpidon rahoitus ja erilaiset rahoitusmallit

Järjestelmäkehityksessä on perinteisesti kolme erilaista rahoitusmallia, jotka ovat kiinteähintainen, tuntilaskutusperusteinen sekä tavoitehintamalli. Näitä malleja voidaan käyttää erilaisissa hankinnoissa. Hankinnat voivat olla kilpailtuja tai suorahankittuja.

Kiinteähintaisessa hankintamallissa tilaaja määrittelee järjestelmälle vaatimukset. Toimittaja sitoutuu täyttämään nämä vaatimukset kiinteähintaisesta korvauksesta. Kiinteän hinnan lisäksi toimittamiselle yleensä annetaan kiinteä toimitusaika. Tämän jälkeen rakennettavaa järjestelmää verrataan vaatimuksiin. Vaatimuksien tulee täytyä riippumatta siihen käytettävästä työmäärästä. Kiinteässä hankintamallissa riskin kantaa toimittaja. Toimittajan virhearvio esimerkiksi laajuudesta ei näy tilaajalle, vaan hän olettaa saavansa vaatimukset täyttävän järjestelmän. Kun toimittaja kantaa riskin, se näkyy myös hinnassa. Tilaajan kannalta hän maksaa siitä, että toimittaja kantaa riskin.

Tuntilaskutusperusteinen hankinta on nimensä mukaisesti tuntilaskutusperusteinen. Tilaaja tilaa toimittajalta työtä, jota toimittaja tekee ja laskuttaa siihen käytetyn ajan perusteella. Riskin tässä hankintamallissa kantaa tilaaja, koska vaatimuksen täyttymiseen käytettävä aika ei välttämättä tarkalla tasolla ole etukäteen tiedossa. Toimittajan kannalta tuntilaskutusperusteisen työn resursointi ja ennustaminen on helppoa. Toimittajan näkökulmasta tuntilaskutusperusteisen hinnoittelumallin käyttäminen on huomattavasti turvallisempaa, mutta siinä ei ole myöskään mahdollisuutta suurille positiivisille yllätyksille. Jos työmäärät alitetaan merkittävästi, hyödyn tästä saa asiakas. Ainoa parametri, jolla toimittaja voi vaikuttaa kannattavuuteen, on käytettävä tuntilaskutushinta. Tuntilaskutusperusteinen työ myös vaatii läpinäkyvyyttä etenemisen raportoinnissa ja mahdollisissa vastaan tulevilla haasteilla. (Lehtimäki, 2006).

Tavoitehintamalli on välimuoto tuntilaskutusperusteisin ja kiinteähintaisen hankintamallin väliltä. Tässä mallissa riski on jaettu toimittajan ja tilaajan välillä. Mallissa tilaaja laatii vaatimukset. Toimittaja arvioi työmäärän, joka tarvitaan vaatimusten täyttämiseen. Näiden pohjalta muodostetaan hankinnalle tavoitehinta. Jos toimittaja alittaa tavoitehinnan, laskutetaan tavoitehinnan ja toteuman erosta puolet. Sama pätee myös toiseen suuntaan. Eli jos toimittaja ylittää tavoitehinnan, laskutetaan ainoastaan puolet ylityksestä kustannuksista toimittajalta.

Erilaisissa hankintamalleissa kyse on loppujen lopuksi siitä, että kuka kantaa riskin ja paljon riskin kantamisesta maksetaan. Haastatteluissa ylläpidon kannalta parhaaksi rahoitusmalliksi nostettiin tuntilaskutusperusteinen rahoitusmalli.

Tuntilaskutusperusteisessa rahoitusmallissa kehittävässä ylläpidossa on toimiva, mutta sen ympärille pitää rakentaa mekanismit läpinäkyvyyden saavuttamiselle ja työnohjaamiselle. Tässä esitetään yksi esimerkki, siitä minkälaisia nämä mekanismit voivat olla.

1. Tilaaja tilaa toimittajalta tuntilaskutusperusteista kehittävä ylläpitoa X eurolla useamman vuoden ajaksi. Tarjous- ja tilausvaiheessa määritellään korkean tason reunaehdot siitä, minkä järjestelmän tai järjestelmien ylläpitämiseksi rahoitus on tarkoitettu. Tämän lisäksi kehityksen kohteita on mietitty korkealla teematasolla.
2. Toimittaja suunnittelee resursoinnin tarpeen mukaan ja varmistaa, että riittävä määrä osaavia työntekijöitä on käytettävissä koko ylläpitokaudelle.
3. Töiden tarkempi sisältö suunnitellaan ja sovitaan pienemmissä erissä yhdessä. Töiden käynnistäminen perustuu aina yhdessä allekirjoitettavaan aktivointiin, jossa

määritellään tuntityön sisältö, lopputuloksen vaatimukset, mahdolliset rajaukset, riskit, takuu, työmääräarvio sekä mahdolliset muut kustannukset.

4. Töitä voidaan käynnistää useampia rinnakkain riippuen ylläpitorahoituksen vo-lyymistä ja toimittajan resursoinnista.
5. Toimittaja raportoi säännöllisin väliajoin tilaajalle työn sisällön edistymisen sekä siihen käytetyn ajan. Mahdollisten riskien tai haasteiden raportointi tilaajalle tehdään heti niiden havaitsemisen jälkeen. Töihin käytetty aika laskutetaan kertymän mukaan säännöllisin väliajoin.
6. Töiden lopputulokset käsitellään yhteydessä osuuden valmistuttua. Tilaaja pääsee todentamaan lopputuloksen aktivoinnissa sovittuja vaatimuksia vasten.

Hankintoja voidaan tehdä kilpailutuksen kautta tai suoraankintoina riippumatta rahoitusmallista. Suoraankintojen hyödyntäminen, kun se hankintalain puitteissa on mahdollista, säästää tilaajan ja toimittajan aikaa tarjousvaiheessa. Todennäköisesti myös sisältö saadaan kuvattua enemmän tarpeet täyttäväksi jo tarjousvaiheessa. Suoraankintojen käyttö on ylläpidossa ja kehittävässä ylläpidossa on järkevää. Kun ylläpito hankitaan samalta toimittajalta, joka on järjestelmän toteuttanut, no se huomattavasti kustannustehokkaampaa.

4.4.2 Asiakkaan tunteminen

Asiakkaan tunteminen nousi useammassa haastattelussa yhtenä tärkeänä asiana. Asiakkaalla tässä yhteydessä tarkoitetaan johtamisjärjestelmän tilaajaorganisaatiota, hankinnasta vastaavaa asianhoitajaa, käyttäjäorganisaatiota ja varsinaisia loppukäyttäjiä. Asiakkaan tunteminen on siis laaja käsite, jonka takana on monia asioita. Tähän lukuun on jaettu näitä erilaisia näkökulmia ja hyötyjä ylläpidettävyyden kannalta.

Tilaajaorganisaatio ja sen edustajat eli hankinnasta vastaavat asianhoitajat tulevat puolustusvoimissa eri organisaatiosta kuin loppukäyttäjät. Tilaajaorganisaatio toimii siltana ja hankintojen asiantuntijan uuden operatiivisen suorituskyvyn tarvitsijan ja niitä tarjoavien toimittajien välillä. Asianhoitajat vastaavat hankinnan kilpailuttamisesta, sopimuksista, suorituskyvyn rakentamisen seuraamisesta ja toimittamisesta. Käyttäjäorganisaatio ja loppukäyttäjät kertovat operatiivisen tarpeen ja vaatimuksien muodossa hankintaorganisaatiolle lähtötiedot hankittavalle suorituskyvyille. Tästä eteenpäin hankinnasta vastaa

hankintaorganisaatio siihen saakka, kun suorituskyky luovutetaan käyttäjäorganisaatiolle. Ylläpidettävyyden kannalta hankintaorganisaation toimintatapojen tunteminen liittyy rahoituksen varmistamiseen. Jos toimittaja ei ymmärrä, miten hankkiva organisaatio toimii, on yhteisymmärrykseen pääseminen haastavampaa.

Jotta toimittaja pystyy rakentamaan ja ylläpitämään järjestelmää, joka täyttää sille asetetut loppukäyttäjien vaatimukset, pitää toimittajan ymmärtää minkälaisessa toimintaympäristössä sitä käytetään ja mitenkä loppukäyttäjät sitä oikeasti käyttävät. Pitkän hankintaketjun kautta on olemassa riski, että alkuperäinen tarve ja syy suorituskyvylle katoaa matkan varrella. Tätä riskiä pystytään pienentämään pitämällä loppukäyttäjät mukana koko hankinnan ajan asiantuntijoina hankkivalle organisaatiolle ja mahdollisesti toimittajalle. Toimittajan on syytä ymmärtää loppukäyttäjän tarve, jotta yllämainittu riski ei toteudu. Pahimmillaan tämä saattaa tarkoittaa, että rakennetaan tarpeeton, väärin toimiva tai hyödytön suorituskyky.

4.4.3 Jatkokehitysmahdollisuuksien tunnistaminen

Rahoituksen takaaminen ja ylläpitokyvyn säilyttäminen vaatii myös aktiivisuutta toimittajalta. Varsinkin kehittävässä ylläpidossa jatkokehitysmahdollisuuksien tunnistaminen ja lisämyynnin aikaansaaminen vaikuttavat merkittävästi ylläpitokykyyn. Toimittajan näkökulmasta lisämyynnin tekemisellä on selkeä tavoite liiketoiminnan jatkuvuuden turvaamisessa.

Jatkokehitysmahdollisuuksia tulee kehittävässä ylläpidossa vastaan jatkuvasti. Esimerkiksi uuden suorituskyvyn rakentamisen yhteydessä voidaan havaita asioita, joita ei ole huomattu määrittelyn yhteydessä. Jatkokehitysmahdollisuuksia saattaa myös tulla järjestelmän käyttäjien palautteessa. Nämä jatkokehitysmahdollisuudet voivat olla kokoluokaltaan hyvin erilaisia.

Näiden jatkokehitysmahdollisuuksien tunnistaminen ja esiin tuominen on tärkeää syötettä tilaajalle. Tilaajan on helpompi priorisoida kehityskohteita, kun tarpeita tuodaan esiin useammasta lähteistä. Tilaajan saa näitä syötteitä myös muualta, joten toimittajan esiin tuomat jatkokehitysmahdollisuudet ovat vain yksi lähde.

Toimittajan kannattaa kerätä kaikki jatkokehitysmahdollisuudet johonkin käsiteltävään muotoon. Muodolla ei ole suurta merkitystä kunhan tieto on käytettävissä, kun sitä tarvitaan. Lisäksi työkalun on syytä tukea jatkokehitysmahdollisuuksien historiatietoa sekä tarjota mahdollisuus käsitellä ja kommentoida niitä. Kun työkaluun kerätään jatkokehitysmahdollisuuksia pidemmältä ajalta ja niitä ylläpidetään järjestelmän muuttuessa, se tarjoaa korvaamattoman tietovaraston potentiaalisesta mahdollisuuksista toimittajalle ja tilaajalle. Tätä tietovarastoa tulisi käydä läpi tilaajan kanssa säännöllisesti

5 POHDINTA

5.1 Johtopäätökset tuloksista

Johtamisjärjestelmän ylläpidettävyys on erilaisista tarpeista tulleiden muutoksien toteuttamista järjestelmään. Muutoksien tekemisen helppous ja nopeus on asia, joka mittaa ylläpidettävyyttä johtamisjärjestelmässä ja yleisemmin tietojärjestelmissä. Muutoksia on aina mahdollista tehdä, mutta jos niiden tekeminen vaatii uuden henkilön kouluttamisen ja järjestelmän teknisisiin ratkaisuihin tutustumisen, ei voida puhua ylläpidettävästä järjestelmästä.

Ennakoitavat toimenpiteet, joilla helpotetaan muutoksien tekemistä luovat ylläpitokyvyn. Johtamisjärjestelmien ylläpitokyvyn rakentamiseen pystytään vaikuttamaan merkittävästi eri elinkaaren vaiheessa tehtävillä toimenpiteillä. Elinkaaren hallinnan tulee olla suunnitelmallista ja jatkuvaa, jotta järjestelmän ylläpidettävyys saavutetaan ja pystytään säilyttämään läpi elinkaaren.

Vastauksena tutkimusongelmaan, joka oli miten ylläpidettävyys muodostuu pitkän elinkaaren johtamisjärjestelmään, voidaan todeta, että ylläpidettävyys muodostuu huomioimalla kolme tärkeintä elementtiä.

1. Johtamisjärjestelmän arkkitehtuurin tulee olla modulaarinen. Teknisissä ratkaisuissa ja rajapinnoissa tulisi käyttää standardoituja ratkaisuja ja ne tulee dokumentoida hyvin. Teknologiaavalinnat tulee tehdä siten, että valitut teknologiat ovat vaikiintuneita sekä niillä tulee olla riittävästi elinkaarta jäljellä.
2. Ylläpitoon osallistuvien työntekijöiden koulututtamiseen, perehdyttämiseen, motivointiin ja osaamisen johtamiseen tulee kiinnittää huomiota. Henkilöstön vaihtumiseen tulee valmistautua etukäteen valmistelemalla tehokkaat perehdytykäytännöt. Osaamisen johtamisessa on tärkeää huomioida organisaation osaaminen, joka muodostuu yksilöiden ja tiimien osaamisesta sekä näiden välisestä tiedonvaihdosta. Yksilöitä ja tiimejä pitää kannustaa työskentelemään yhdessä. Motivointia ja erityisesti sisäisen motivaation rakentamista tulee tehdä tasaisesti.
3. Ylläpidolle tulee varmistaa riittävän suuri rahoitus, jotta ylläolevia asioita on mahdollista tehdä. Suositeltavaa ylläpidettävyuden kannalta on käyttää tunti-

ktusperusteista rahoitusmallia, koska siitä saa suurimman hyödyn tilaaja ja toimittaja. Rahoituksen järjestäminen on tilaajan vastuulla, mutta myös toimittaja pystyy aktiivisuudellaan vaikuttamaan asiaan. Tässä auttaa hyvien asiakassuhteiden ylläpitäminen, jatkokehitysmahdollisuuksien tunnistaminen ja aktiivinen myynti.

Eri osa-alueet linkittyvät ylläpidettävyyden näkökulmasta toisiinsa. Kaikki osa-alueet pitää olla jollain tasolla hoidettu, että järjestelmän ylläpitokyky on olemassa. Rahoitus mahdollistaa sen, että muita osa-alueita pystytään tekemään. Ilman osaavia ihmisiä teknisiä ratkaisuja ja muutoksia järjestelmään ei pystytä tekemään. Arvoketjuajattelun mukaisesti operatiivisen tarpeen mahdollisimman pitkään täyttävä käytössä oleva johtamisjärjestelmä on lopputulos, joka pyritään saavuttamaan.

Johtamisjärjestelmän ylläpidettävyyden on toimittajan, tilaajan ja loppukäyttäjien yhteisvastuu. Jos joku näistä toimijoista ei ole sitoutunut ylläpidettävyyden rakentamiseen, on lopputulos huono.

Omassa työssäni tutkimustuloksista eniten vaikuttaa organisaation osaamisen kasvattaminen lisäämällä tiimien välistä työskentelyä ja valmistelemalla uusien työntekijöiden perehdytys paremmin. Näiden kahden asian merkitys ylläpidettävyyden näkökulmasta oli itselleni yllättävin asia. Asioihin on kiinnitetty huomiota aikaisemmin, mutta eri syystä. Parantamalla näitä osa-alueita pystytään lisäämään Insta DefSec Oy:n kehittämien ilma- puolustuksen johtamisjärjestelmien ylläpidettävyyttä.

5.2 Lisäarvo

Lisäarvo, joka ylläpidettävyyden rakentamisella ja ylläpitämisellä saavutetaan, on selkeä. Tätä lisäarvoa voidaan miettiä useammasta eri näkökulmasta. Ylläpidettävyyden tuo lisäarvoa järjestelmälle, toimittajalle sekä tilaajalle. Lisäarvo muodostuu kokonaisuuden tavoitteen täyttymisestä, joka on täyttää johtamisjärjestelmällä jokin operatiivinen tarve mahdollisimman pitkään tekemällä mahdollisimman vähän työtä sen eteen.

Johtamisjärjestelmä, jossa ylläpidettävyys on huomioitu täyttää sille asetetut toiminnalliset ja ei-toiminnalliset vaatimukset pidempään. Lisäarvoa muodostuu siitä, että järjestelmä on toimintavarma, luotettava, suorituskykyinen, yhteensopiva, helposti käytettävä ja tietoturvallinen.

Toimittajan näkökulmasta lisäarvoa muodostuu kustannustehokkaammasta tekemisen tavasta. Kustannustehokkuus on yksi asia, jolla toimittaja luo asiakastyytyväisyyttä ja pitkäkestoisia asiakassuhteita. Ylläpidettävyys tukee näiden muodostumista muutenkin ja se on yksi lisäarvoa tuottava osuus. Ylläpidettävästä järjestelmästä pystytään myös hyödyntämään erilaisia osuuksia muihin käyttötarkoituksiin. Tämä on selkeä lisäarvo jatkomyynnille ja mahdollistaa osuuksien hyödyntämistä muissa käyttötarkoituksissa.

Tilaaajan ja käyttäjien näkökulmasta ylläpidettävyys tuo lisäarvoa, koska johtamisjärjestelmä täyttää varmemmin operatiivisen tarpeen. Tähän varmuuteen liittyy oleellisesti huoltovarmuus, jolla tarkoitetaan varautumista kriisioloihin. Se, että järjestelmää pystytään ylläpitämään ja pitämään käytössä epävarmoissa kriisioloissa, on kriittisen tärkeää. Lisäarvoa myös muodostuu järjestelmän elinkaaren ennakoitavuudesta, joka mahdollistaa suunnitelmallisuuden ja kokonaisuuden hallinnan. Ylläpidettävyys luo myös paremmat perusteet tulevaisuuden tarpeiden suunnittelulle.

5.3 Jatkotutkimus

Tässä opinnäytetyössä ei arvioitu ylläpidettävyyteen vaikuttavien osa-alueiden merkittävyyttä. Tämä on asia, jota tulevaisuudessa kannattaisi tutkia tarkemmin. Tämän pohjalta pystyisi tekemään johtopäätöksiä, siitä mihinkä alueisiin kannattaisi erityisesti panostaa, jos ylläpidettävyyttä halutaan parantaa.

Myös työn rajauksien purkamalla kokonaisuudesta saisi laajemman kuvan. Jos tarkasteluun otettaisiin ylläpidettävyyden näkökulmasta myös järjestelmien laitepuoli ja ajoympäristöt, näistä löytyisi varmasti kokonaisuuteen vaikuttavia asioita.

Asiaa voisi myös lähestyä enemmän tilaaajan näkökulmasta, jolloin mukaan voisi liittää enemmän sidosryhmien hallintaa sekä useampien toimittajien hallinnan vaikutuksia ylläpidettävyyteen.

LÄHTEET

- Anttila, Juhani & Jussila, Kari. 2016. Suomen Standardisointiliitto SFS. Mitä laatu on? Luettu 16.5.2018. https://www.sfs.fi/ajankohtaista/uitiskirjeet/uitiskirjeet_2016/mita_laatu_on_artikkeli
- Craig, Rick D., Jaskiel, Stefan P. 2002. Systematic Software Testing. Boston: Artech House, Inc. 2002.
- Deci, E., & Ryan, R. 2000: The "What" and "Why" of Goal Pursuits: Human Needs and the Self-Determination of Behavior. Psychological Inquiry. 11(4): 227-268.
- Gorton, Ian. 2011. Essential software architecture. Berlin, Heidelberg : Springer 2011.
- Homes, Bernard. 2013. Fundamentals of Software Testing. Hoboken: John Wiley & Sons, Incorporated 2013.
- IEEE Std 610.12-1990. 2003. IEEE Standard Glossary of Software Engineering Terminology.
- Kruchten, Philippe. 1995.. Architectural Blueprints — The “4+1” View Model of Software Architecture. IEEE Software 12 (6), pp. 42-50.
- Lehtimäki, Timo. 2006. Ohjelmistoprojektit käytännössä. Helsinki: Readme.fi 2006.
- Puolustusministeriö. 2018. HX-hankkeen tarjouspyyntö lähetettiin. Luettu 13.5.2018. http://valtioneuvosto.fi/artikkeli/-/asset_publisher/hx-hankkeen-tarjouspyynto-lahetettiin
- Tuula Sumkin, Lauri Tuomi. 2012. Osaamisen ja työn johtaminen: organisaation oppimisen oivalluksia. Helsinki: SanomaPro 2012.
- Saaranen-Kauppinen & Puusniekka. 2018. Luettu 15.5.2018. http://www.fsd.uta.fi/menetelmaopetus/kvali/L7_3_2.html
- US. Army Materiel Command. 1972. Engineering Design Handbook - Maintainability Guide for Design: (AMCP 706-134).

Haastateltavat

Asiala Antti, Development Manager, Insta DefSec Oy
 Einola Pasi, Director, Insta DefSec Oy
 Hartikainen Jarkko, Program Manager, Insta DefSec Oy
 Kirves-Lassila Sirkku, Product Manager, Insta DefSec Oy
 Ruokojärvi Marko, Product Manager, Insta DefSec Oy
 Salli Karri-Tuomas, Director, Insta DefSec Oy