

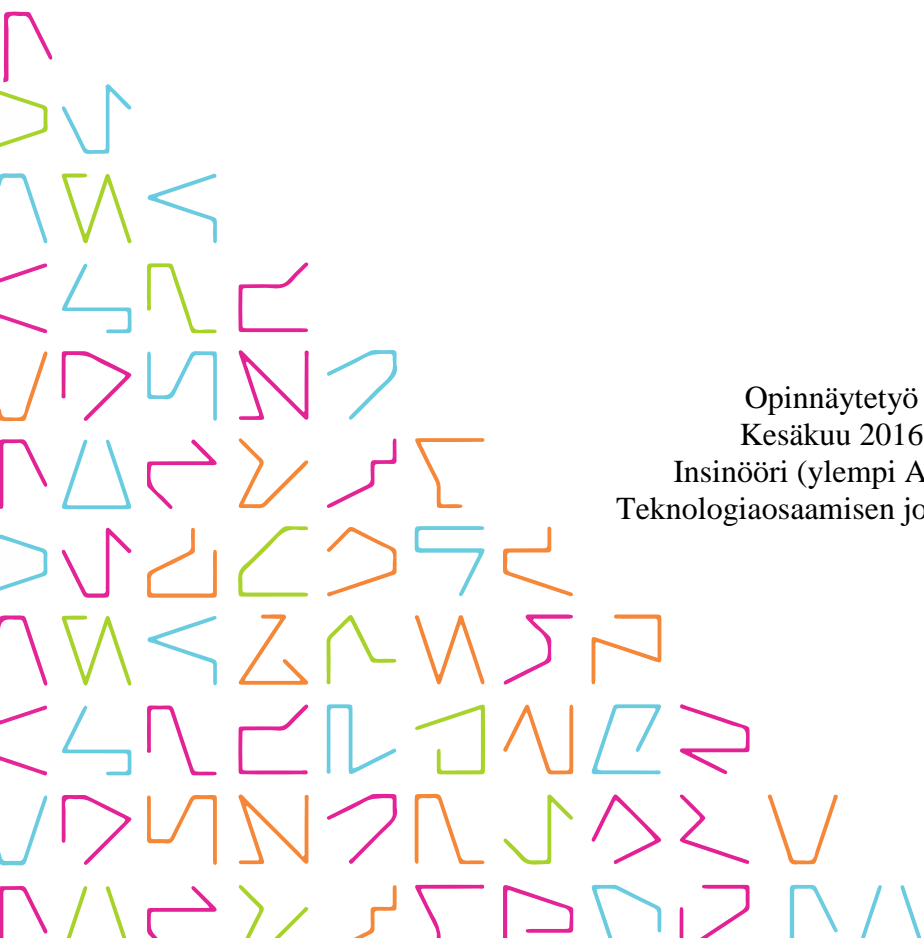


TAMPEREEN
AMMATTIKORKEAKOULU

VAATIMUSTEN KÄSITTELY LOISTEHON KOMPENSOINTILAITTEISTOJEN SUUNNIT- TELUSSA

Mika Lammi

Opinnäytetyö
Kesäkuu 2016
Insinööri (ylempi AMK)
Teknologiaosaamisen johtaminen



TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Insinööri (ylempi AMK)
Teknologiaosaamisen johtaminen

LAMMI MIKA:

Vaatimusten käsittely loistehon kompensointilaitteistojen suunnittelussa

Opinnäytetyö 71 sivua, joista liitteitä 5 sivua

Toukokuu 2016

Kehitystehtävän päätavoitteeksi asetettiin tutkia vaatimusten käsittelyä. Tutkimusmenetelmänä käytettiin pääosin kvalitatiivista kirjallisuustutkimusta. Kirjallisuustutkimusta haettiin tukea avoimilla haastatteluilta, joiden tuloksilla syvennettiin kirjallisuustutkimuksessa löydettyjä seikkoja. Osatavoitteena oli löytää vastauksia siihen, millaisista elementeistä vaatimusten käsitteleminen koostuu. Lisäksi tarkoituksena oli löytää käytäntöjä, joiden varaan voidaan työntilajalle perustaa vaatimuksien käsittelyä hyödyntävä suunnitteluprosessi.

Kehitystehtävän tulisi auttaa hahmottamaan, kuinka projektien vetäjien tulisi ohjata järjestelmien suunnittelua vaatimusten käsittelyä hyödyntäen. Päämääräksi asetettiin määritellä millaiseksi loistehon kompensointilaitteiston suunnitteluprosessin tulisi muuttua, jotta se palvelisi tehokkaasti sekä vaatimusten määrittelyä että vaatimusten hallintaa.

Kehitystehtävän tuloksena saatiin muodostettua selkeä ymmärrys siitä millaisista tekijöistä vaatimusten määrittely ja hallinta muodostuu. Kirjallisuustutkimuksen avulla pystyttiin hahmottamaan reunaehdot vaatimusten kartoittamiselle. Kirjallisuuden perusteella ja haastatteluiden tukemana vaatimusten hallinnalle luotiin toimiva prosessimalli.

Työntilajalla käytettävissä olevaan Jira-järjestelmään on luotavissa monipuolisesti yrityksen vaatimuksien käsittelyä palveleva ympäristö. Kehitystehtävän aikana tehdyn tutkimustyön perusteella pystyttiin määrittelemään millaisilla perusominaisuuksilla Jira-järjestelmällä voidaan aloittaa vaatimusten käsittely työntilajalla.

Tutkimusta ei jatkettu käytännön tasolla niin pitkälle, että vaatimuksien käsittelyä olisi toteutettu oikeassa toimitusprojektissa. Tutkimuksen seuraavana askeleena tulee olemaan sopivan aloitusprojektin löytäminen ja kehitystehtävässä luotujen prosessien käytännön ongelmien kartoittaminen.

Asiasanat: vaatimus, vaatimusmäärittely, vaatimusten hallinta, jira

ABSTRACT

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

LAMMI MIKA:

Requirement Definition Methods in the Reactive Power Compensation System's Design Process

Master's thesis 71 pages, appendices 5 pages
May 2016

Within this thesis qualitative literature research was exploited to study the requirement engineering and the requirement management principles. Literature research was completed with an open interviews to receive practical information from the requirement engineering process. Interviews were concentrated to other fields of industry. Main target for the research was to investigate the factors of which the requirement engineering and the requirement management are based on. A further objective was also set to support the main target. The purpose for it was to produce a process model for the company to be implemented in its current design process.

Parallel target for the thesis was to study how technical requirement changes in the project execution should be managed. Obviously it is project management role, but the purpose was to define the rules and working methods to support the project management. For the requirement changes it should be noted that the most extra expenses for the project usually comes from changes which are unexpected or are poorly implemented in the final product.

As a result for this study a new design process model was created based on the boundaries found out from the literature and interviews. The process model included requirement engineering and requirement management aspects, meanwhile rest of the reactive power compensation design process remained untouched.

In final part of the thesis Jira-system was taken under investigation. Since the system has a wide capabilities for modifications, the knowledge gathered during literature study was exploited to produce efficient working methods with Jira. Main focus was to produce simple guide lines to support first steps in commissioning of the program.

Practical portion of this thesis work does not extend to evaluation or testing of the new processes. However it should be noted that even though requirement engineering and management process are working in the desktop it is most likely that some practical problems needs to be overcome before process are in efficient use.

Key words: requirement, requirement engineering, requirement management, jira

SISÄLLYS

1	JOHDANTO.....	7
2	TYÖNTILAAJA	8
	2.1 GE Energy Connections.....	8
	2.2 Projektin sidosryhmät	9
	2.3 Suunnittelun ongelmakohdat	10
3	KEHITYSTEHTÄVÄ	14
	3.1 Kehittämistehtävän tavoite	14
	3.2 Tutkimusmenetelmät	15
4	VAATIMUSMÄÄRITTELYN TAUSTATEKIJÄT	16
	4.1 Vaatimusten käsittelyn yhdistäminen kompensointijärjestelmiin	16
	4.2 Vaatimusten käsittelyn tärkeys	17
	4.3 Suunnittelun sidosryhmät	19
	4.3.1 Sidoryhmien analysointi.....	20
	4.4 Prosessimallit	23
	4.4.1 Vesiputousmalli	24
	4.4.2 Spiraalimalli	26
	4.4.3 V-malli	27
	4.4.4 Ketterät menetelmät	28
	4.4.5 Prosessimallien teorian liittäminen tutkimukseen.....	29
	4.5 Järjestelmän ja systeemin erottaminen	31
	4.5.1 Loistehon kompensointijärjestelmien systeemin ja järjestelmän tunnistaminen	32
	4.6 Vaatimusten erilaiset tyypit	33
5	SUUNNITTELUPROSESSIIN PEILAAMINEN HAASTATTELUIDEN KAUTTA.....	36
	5.1 Haastattelut	36
	5.2 Päätelmät haastatteluista	39
6	VAATIMUSTEN KÄSITTELYN KOMPONENTIT	42
	6.1 Vaatimusmäärittely	44
	6.1.1 Vaatimustenkartoittaminen	45
	6.1.2 Vaatimusten analysointi.....	46
	6.1.3 Vaatimusten dokumentointi	50
	6.1.4 Vaatimusten validointi	51
	6.2 Vaatimustenhallinta	52
7	VAATIMUSKÄSITTELYN TOTEUTUS	54
	7.1 Vaatimuksien dokumentointi Jirassa	55

7.1.1	Vaatimusten työkierto Jirassa	57
7.1.2	Vaatimusdokumentti Jirassa.....	59
8	TUTKIMUKSEN TULOKSIEN YHTEENVETO	62
	LÄHTEET	64
	LIITTEET	66
	Liite 1. Haastattelulomake (tiivistetyt vastaukset)	66
	Liite 2. Vaatimuksen dokumentointilomake Jira-järjestelmässä (Jira 6.4 ohjelmisto 2016).	68
	Liite 3. Vaatimusmäärittelyn ja prosessimallin liittäminen toisiinsa	69
	Liite 4. Haastattelujen mind-map kaavio.....	70
	Liite 5. Vapaasti kirjoitun vaatimuksen ohjenuorat	71

LYHENTEET JA TERMIT

GE	General Electric
HW	Hardware
MS	Mekaniikkasuunnittelu
O&S	Ohjaus- & suojausjärjestelmäsuunnittelu
P&C	Protection & control
PET	Power electronics Tampere
PJ	Projektin johto
PM	Project manager (projektipäällikkö)
PEM	Project engineering manager (tekninen projektipäällikkö)
SEI	Software engineering institute
SS	Systeemisuunnittelu
SW	Software
SWOT	Strengths, Weaknesses, Opportunities, Threats - analyysi

1 JOHDANTO

Vaatimusten käsitteleminen on perinteinen työskentelytapa menestyneissä ohjelmistoprojekteissa. Sen sijaan vaatimusten määrittelyn ja vaatimusten hallinnoinnin soveltaminen perinteisen teollisuuden käyttöön on harvemmin tutkittu aihe. Tämän kehitystehtävän tavoitteena on valottaa oleellisimpia seikkoja, mitä kuuluu vaatimusten käsittelyn taustalle ja millaisista elementeistä kehitystehtävän tilaajalle saadaan vaatimuksien käsittely rakennettua. Työn tilaajana on GE Grid Solutionin Tampereella sijaitseva PET-yksikkö. PET toimittaa räätälöityjä loistehon kompensointijärjestelmiä kantaverkkoyhtiöiden tarpeisiin.

Voidaan todeta, että vaatimusmäärittely on yksinkertaisimmillaan muistilista suunnittelijan mielessä. PETillä on kuitenkin havaittu, että projektin parissa työskentelevien suunnittelijoiden määrän lisääntyessä tai suunnittelijoiden vaihtuessa projektin aikana, ajautetaan suunnittelutehtävissä helposti vaikeuksiin. Ongelman syvyyttä lisäävät vielä monimutkaiset projektit, joissa suunnittelu joudutaan jakamaan useisiin eri osa-alueisiin.

Tässä kehitystehtävässä yritetään löytää ratkaisuja työntilaajan suunnitteluprosessin kehittämiseen, jotta vaatimusmäärittely saataisiin tehokkaasti sisällytettyä osaksi projektitoteutusta. Vaatimusten käsittelyn perusteiden hahmottamiseen haetaan tukea kirjallisuudesta. Lisäksi yritetään avoimien haastattelujen avulla kartoittaa kuinka vaatimusmäärittely on toteutettu muissa perinteisen teollisuuden alalla toimivissa yrityksissä. Tehtävän ulkopuolelle rajattiin modernien, niin kutsuttujen ketterien menetelmien syvällisempi tutkiminen. Työn käytännönsuutta rajattiin myös sitten, että työssä kehitettäviä uusia prosesseja ei kehitystehtävän puitteissa testata tai arvioida.

Työn tuloksena syntyy myös katsaus siitä miten työn tilaajayritykselle käyttöön tarjotusta vaatimuksienhallintatyökalusta saataisiin, aikaisemmin kehitystehtävässä tutkittuja vaatimusten hallinta tarpeita, parhaiten palveleva hyötykokonaisuus. Tarjottu vaatimuksienhallintajärjestelmä on vapaasti muokattava, verkossa toimiva palvelu. Täten sen säätäminen yrityksen tarpeisiin parhaiten sopivaksi vaatii ymmärrystä niistä seikoista joita vaatimuksienmäärittelyltä ja hallinnalta perustapauksissakin edellytetään.

2 TYÖNTILAAJA

2.1 GE Energy Connections

Kehitystehtävän tilaaja on osa GE Energy connection konsernia. Työn tilaaja tunnetaan konsernissa PETtinä. PETin tuoteportfolioon kuuluu räätälöityjä loistehon kompensointijärjestelmien toimittaminen kantaverkkoyhtiöiden sovelluksiin. Räätälöity loistehon kompensointijärjestelmä koostuu ulos asennettavasta suurjännitelaitteistosta ja sitä ohjaavasta, sisätilaan asennettavasta, ohjaus- ja suojausjärjestelmästä (P&C). Suurjännitelaitteiston ja P&C-järjestelmän liittää toisiinsa tehoelektronikalla toteutettu tyristoriventtiili. Kuvassa 1 on esitettyä tyypillinen loistehon kompensointijärjestelmä. Siinä suurjännitelaitteet näkyvät vasemmassa reunassa. P&C-järjestelmä ja tehoelektronikka ovat sijoitettuna oikeassa reunassa olevaan rakennukseen.



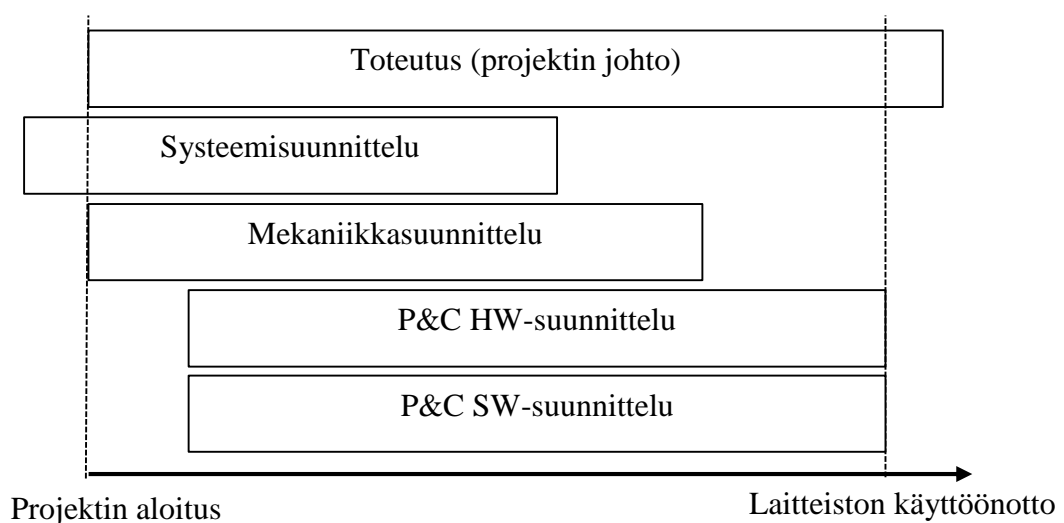
KUVA 1. Loistehon kompensointijärjestelmä (Kuva: Sisäinen kuva-arkisto, Carlo Gasparri 2015)

Kantaverkkoyhtiöiden ollessa kriittisiä toimijoita valtioiden infrastruktuurissa on niillä useimmissa tapauksissa erittäin tiukat toimintaohjeet ja määritelmät sallituista teknologioista. Lisäksi loistehon kompensointijärjestelmän toiminnollisuudet vaativat jokaisessa projektissa asiakaskohtaisia erikoispiirteitä toteuttavia muutoksia. Näiden kahden seikan johdosta PET on pakotettu muokkaamaan järjestelmätoimituksia vastaamaan asiakkaiden

vaatimuksia. Tyypillisiä muuttuvia vaatimuksia kompensointilaitteistolle on toimintaa määrittelevien suunnitteluparametrien muutokset, komponenttien toimittajien rajaaminen, mekaanisten asennuksien erityisohjeet ja P&C:ää koskevat toiminnollisuusvaatimukset. Yleisesti voidaan todeta, että järjestelmän räätälöinti tarpeet kohdistuvat laajasti kaikkiin laitteiston osa-alueisiin.

2.2 Projektin sidosryhmät

Loistehon kompensointilaitteistot on jaettu viiteen PETin sisäiseen sidosryhmään: projektitoteutus, systeemisuunnittelu, mekaniikkasuunnittelu ja P&C-suunnittelu. P&C-suunnittelu on vielä jaettuna SW- ja HW-suunnitteluryhmiin. Toteutus kostuu projektipäälliköstä ja projektin teknisestä projektipäälliköstä. Projektin johto on mukana projektin alusta alkaen ja jatkaa projektissa kunnes se on luovutettu jälkimarkkinoinnille. Monessa tapauksessa systeemisuunnittelu vastaa laitteiston sähköisestäämitoituksessa ja on mukana projektissa jo tarjousvaiheen esisuunnittelussa. Mekaniikkasuunnittelun vastuulle kuuluu järjestelmän mekaanisten rakenteiden suunnittelu sekä suurjännitekomponenttien mitoitust ja määrittely. P&C-ryhmä on jaettu karkeasti sähkösuunnitteluun ja ohjelmasuunnitteluun. Kuvassa 2 on esitettyä projektitoteutuksen sidosryhmien osallistuminen suunnitteluun projektin aikana. Kuvassa projektinjohto on piirretty ylimmälle riville, sen toimissa projektin jokaista osa-aluetta koordinoivana tekijänä.

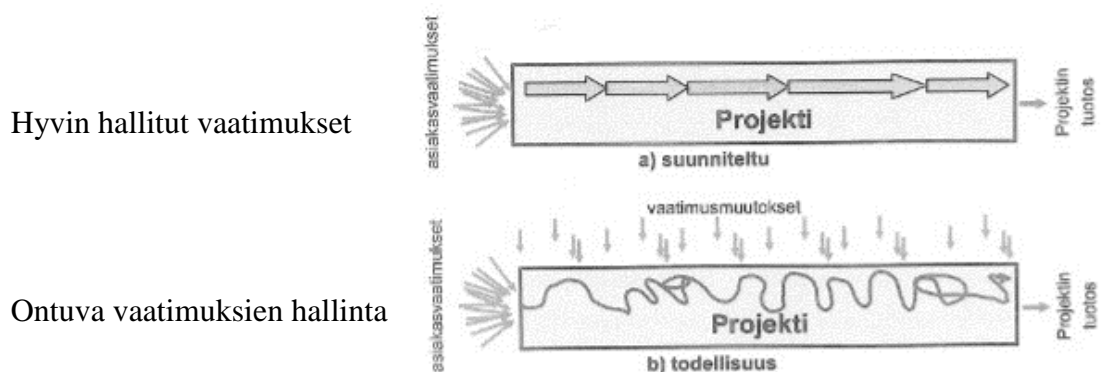


KUVA 2. Loistehon kompensointijärjestelmän suunnitteluryhmät

2.3 Suunnittelun ongelmakohtat

PETin toimittamien projektien laajuuden kasvaminen sekä asiakkaiden profiilin muuttuminen on korostanut suunnittelun ongelmakohtia. Monimutkaisen järjestelmän suunnittelun koordinointiin tulisi saada vakiinnutettua toimintamalleja, joiden avulla projektin teknisestä suunnittelusta saataisiin joustavampaa ja samanaikaisesta leikattua suunnitteluvirheitä. Seurattaessa ISO9001:2008 laatustandardia voidaan sieltä poimia yhdeksi yrityksen johtoryhmän laatuksiteriksi asiakasvaatimuksiin vastaaminen. Standardi määrittelee, että yrityksessä tulisi olla määriteltyä toimintamalli minkä perusteella voidaan varmistaa täyttykö asiakkaan laitteistolle asettamat vaatimukset (Charles, Joseph, John 2009, 40). Laatustandardien ollessa asiakkaille tärkeitä ohjenuoria toimittajien valinnassa, tulisi laatustandardien luomat raja-aidat jalkauttaa käytäntöön myös suunnittelun tehokkaammassa ohjaamisessa.

PETin projekteille on tyypillistä, että jo tarjousvaiheessa niihin saatetaan sopia niin sanottuja poikkeamia asiakkaan virallisiin määrittelydokumentteihin. Projektin ensimmäisissä asiakastapaamisissa projektipäällikkö sopii suunnittelun kannalta oleellisia linjanvetoja. Suunnittelijoiden kommunikoidessa suoraan asiakkaan asiantuntijoille, saatetaan tunnistaa uusia teknisiä yksityiskohtia. Näiden kaikkien elementtien yhdistyessä on projektien teknisentoteutuksen seuraaminen muodostunut ongelmalliseksi ja monesti toteutuksen loppuvaiheessa törmätään yhteensopivuus ongelmiin. Haikala & Mikkonen kuvaillee kirjassaan Ohjelmistotuotannon käytännöt kuinka laitteiston toimittajan näkökulmasta suunnittelun lähtökohtana tulisi olla selvät vaatimukset. Kuvassa 3 on esitettyä kuinka Haikala & Mikkonen näkee projekti toteutuksen, kun vaatimukset ovat selvästi hallinnoituja tai vaatimuksienhallinta on ontuvaa. (Haikala & Mikkonen 2011, 22)



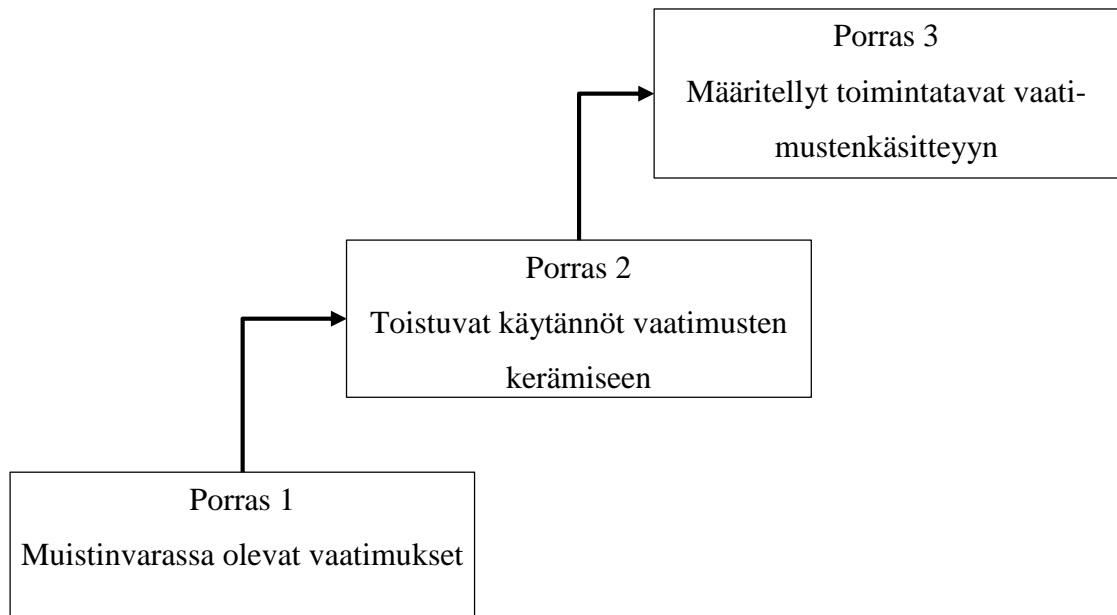
KUVA 3. Projektitoteutuksen ero, kun vaatimukset ovat alussa määritelty ja projektin aikana määritelty (Haikala & Mikkonen 2011, 22) .

Toisena ongelmana on havaittu projektien kehittyminen erittäin monimutkaisiksi. Monimutkainen projekti saattaa sisältää yli tuhat sivua asiakkaan määrittely dokumentaatiota. Projektien ollessa näin laajoja tarvitsee niihin osallistua suunnittelijoita useassa eri projektin vaiheessa. Käytännössä tämä tarkoittaa, että suunnittelija joutuu tutustumaan koko projektin erityispiirteisiin ennen kuin hän voi tehokkaasti alkaa toteuttamaan omaa osaluettuaan. Tämä johtaa siihen, että pirstaloituneita määrittelyjä joudutaan tutkimaan useaan kertaan eri suunnittelijoiden toimesta. Tällöin suunnittelijoille syntyy helposti poikkeava ymmärrys valmistettavasta kokonaisuudesta. Projektipäällikön ollessa raskaasti kuormitettu hänelle ei välttämättä jää aikaa katselmoida kaikkia projektin suunnitteludokumentteja sillä tarkkuudella, että yksittäisen suunnittelijan ajatusvirheet saataisiin korjattua. Kirjassa *Guide to the project management body of knowledge* on selvästi kuvattuna kuinka vaatimusmäärittelyn tulisi olla omana kokonaisuutenaan, hyvin hoidetussa projektissa. Vaatimusmäärittelyllä pystytään kirjan mukaan kanavoimaan projektin toteutuksen kannalta kriittistä tietoa projektin sidosryhmille. (Project management institute, 106)

Vaatimusmäärittely on sisällytettyä myös ohjelmistotekniikan laatustandardeihin. Vaikka loistehon kompensointilaitteisto ei kokonaisuudessaan ole ohjelmistoprojekti, voidaan vaatimusmäärittelyn tärkeyttä perustella ohjelmistotekniikan laatustandardien kautta. Kirjassa *Software Process Quality : Management and Control* kuvataan miten SEI-järjestö on määritellyt ohjelmistotekniikan kypsyyttä kuvaavat portaavat (Kenett & Baker 1999, 58).

SEIn standardissa ohjelmistotuotannon kypsyyttä on jaettu viiteen eri portaaseen. Kun peilataan portaikkoon perinteisemmän teollisuuden käyttöön, voidaan todeta sen olevan liiaksi ohjelmistotekniikkaan nojautuva. Kirjassa *Requirment engineering* on kuitenkin kyseinen asteikko paremmin virtaviivaistettuna. Loistehon kompensointilaitteistojen vaatimusmäärittelyn kannalta on todenperäisempää verrata suunnittelun kypsyyttä kyseisen kirjan asteikkoon.

Kuvassa 4 on esitettyä SEIn määrittelemät ohjelmistotekniikan kypsyyttä kuvaavat portaavat vaatimusmäärittelyyn kohdistettuna. (Kotonya & Sommerville 1997, 47)



KUVA 4. Ohjelmistotekniikan kypsyyttä kuvaavat askeleet (Kotonya & Sommerville 1997, 47. Muokattu)

Vaatimusmäärittelyn kypsyyttä kuvaavan asteikon portaavat voidaan tarkemmin kuvata seuraavilla tavoilla (Kotonya & Sommerville 1997, 47; Kenett & Baker 1999, 58):

PORRAS 1

Vaatimusmäärittelyn kypsyyden ensimmäiseen portaaseen yletytään, kun yrityksessä tapahtuu suunniteltavan laitteiston vaatimusten keräämistä. Useimmissa tapauksissa vaatimukset ovat suunnittelijoiden itseänsä varten kerättyjä muistiinpanoja. Portaalla yksi organisaatiolla ei kuitenkaan ole millään tasolla määriteltynä vaatimusten määrittämistä tai hallintaa. Tyypillisiä ongelmia ensimmäisellä tasolla olevissa organisaatioissa ovat väärinymmärrykset eri suunnittelun sidosryhmien kanssa ja iso lukumäärää suunnittelun iterointia.

PORRAS 2

Toisella portaalla olevat organisaatiot ovat tunnistaneet tarpeen vaatimusten hallinnalle ja niiden määrittämiselle. Organisaatiossa on tällöin luotuna selvät käytännöt ja ohjeet siitä kuinka vaatimuksia kerätään ja hallinnoidaan. Tasolla kaksi vaatimuksien sisällön tulee myös olla laadultaan päteviä.

PORRAS 3

Korkeimmalla tasolla vaaditaan, että vaatimusten keräämisessä ja hallinnassa käytetään niin sanottuja kehittyneitä menetelmiä. Organisaatiossa tulee käyttää vaatimusten keräämisessä ja hallinnoimisessa yhteisiä prosesseja. Tärkein seikka tasolle kolme yltämisessä on, että organisaatiolla on selvät tavoitteet siitä kuinka vaatimusten hallintaa ja keräämistä tulisi kehittää.

Peilattaessa PETin tämän hetkistä toimintamallia, kirjallisuudesta havaittuihin vaatimusmäärittelyn kypsyyttä kuvaaviin portaisiin, voidaan arvioida PETin toiminnan täyttävän osan ensimmäiselle portaalle asetettavista määreistä. Toiminnasta jää kuitenkin puuttumaan kriittisiä osia, kuten vaatimusten käsittelyprosessit. Analyysin tuloksena voidaan todeta, että organisaation toiminnan kannalta vaatimusmäärittelyn nostaminen ensimmäiselle tasolle tulisi olla mahdollista tämän kehitystehtävän tuloksien pohjalta.

3 KEHITYSTEHTÄVÄ

3.1 Kehittämistehtävän tavoite

Räätälöityjen projektien toteutuksessa ongelmaksi on havaittu, että jokaisesta projektista muodostuvan yhä uudelleen kehitettävä tuote. Projektin johtamisessa yhdistyy silloin sekä normaali toimitusprojekti että monimutkainen uuden tuotteen kehitysprojekti. Kummankin tapahtuessa samanaikaisesti, tarvitaan projektin menestykselliseen läpiviemiseen erilaisia suunnittelumenetelmiä.

Kehittämistehtävän tavoitteeksi asetettiin tutkia miten loistehon kompensointijärjestelmän suunnitteluprosessiin saadaan sisällytettyä vaatimusten määrittely ja kuinka toteutetaan vaatimusten hallinta. Vaatimusmäärittely on perinteisesti tärkeässä roolissa muun muassa tuotekehityshankkeessa tai ohjelmistoprojektintoteutuksessa, mutta nyt se tulisi saada sovellettua perinteisen teollisuuden apuvälineeksi. Tavoite rajattiin pelkästään laitteiston teknisiä vaatimuksia määrittelevän prosessin hahmottamiseen. Työntuloksena tulisi pystyä aloittamaan vaatimusten määrittelemisen ja projektin suunnittelun katselmoiminen, luotuja vaatimusmäärittelyprosessia hyödyntäen.

Työntilaaja halusi kehitystehtävän sisältävän riittävän määrän vaatimusmäärittelyn perusteiden tutkimista ja arvioimista kuinka vaatimusmäärittelystä saadaan koko laitteiston suunnittelua saumattomasti tukeva työkalu. Pelkän teorialatkimuksen lisäksi asetettiin tavoitteeksi kokemusten ja parhaiden käytäntöjen kerääminen haastattelemalla käytettävissä olevia asiantuntijoita. Haasteltavia asiantuntijoita pyrittiin hankkimaan sekä omasta organisaatiosta että sen ulkopuolelta.

PETillä on käytettävissä GE-konsernin tarjoama vaatimustenmäärittelyohjelmisto, Jira. Kehitystehtävä haluttiin rajata siten, että vaatimusmäärittelyprosessi tulisi rakentua kyseistä ohjelmistoa hyödyntäväksi. Jira-ohjelmisto on suunniteltu tukemaan ohjelmistoprojekteissa käytettäviä tekniikoita, joten yhtenä haasteena on saada riittävästi ymmärrystä vaatimusmäärittelystä tukemaan Jiran käyttöä kaikissa suunnittelun osa-alueissa.

3.2 Tutkimusmenetelmät

Opinnäytetyön tutkimusasetelma ohjasi työssä käytettäväksi kvalitatiivista tutkimusmenetelmää. Vaatimusmäärittelyn tutkimisen ollessa ilmiöön keskittyvää, soveltuu kvalitatiivisten tutkimusmenetelmien käyttäminen parhaiten tavoitteiden saavuttamiseen. Valittua tutkimusotetta tukee myös työntilaajan tarve kehittää toimintaansa tutkimuksen tuloksien avulla. (Eskola & Suoranta 2005) Tutkittavaksi kirjallisuudeksi valikoituivat vaatimusmäärittelyä ja projektien johtamista ohjeistavat kirjat sekä aihepiiriä käsittelevät teollisuuden lehtijulkaisut.

Kirjallisuustutkimuksen tueksi valittiin tutkimusaineistoa täydentävät asiantuntijahaastattelut. Haastatteluihin valittiin asiantuntijoita sekä omasta organisaatiosta että muissa teollisuuden yrityksissä toimivista henkilöistä. Haastatteluihin osallistuneilta vaadittiin aikaisempaa kokemusta vaatimusmäärittelyprosessista ja suunnittelu taustaa. Näillä vaatimuksilla saatiin haastattelut sidottua paremmin kehitystehtävän käytännöllisyystavoitteen saavuttamiseen.

Asiantuntijahaastatteluiden ja kirjallisuustutkimuksen pohjalta pystytään konstruktiiivista tutkimusmenetelmää hyödyntäen rakentamaan työntilaajalle täysin uusi toimintamalli ja sitä tukevat projektin eri vaiheisiin sidotut dokumentit ja aktiviteetit. Konstruktiiivisen tutkimuksen perusolettamana on, että tutkimuksen tuloksena luodaan jotain uutta. Konstruktiiivisessa tutkimuksessa kuvailtu tavoite keskittyä käytännön läheiseen haasteiden ratkaisemiseen tukee myös osaltaan tämän kehitystehtävän tavoitteeseen pääsemistä. (Metodix: Metoditietämystä kaikille 2016)

4 VAATIMUSMÄÄRITTELYN TAUSTATEKIJÄT

4.1 Vaatimusten käsittelyn yhdistäminen kompensointijärjestelmiin

Vaatimusten käsittelyn hahmottaminen on hyvä aloittaa tutustumalla systeemyöhön. Systeemyö on perinteisesti yhdistetty ICT-alan termistöön, kuitenkin sen parhaita puolia hyödyntäen voidaan systeemyöskentelystä saada perinteisemmän teollisuuden käyttöön hyödyllisiä toimintamalleja. Vaatimusten hallinnan ja määrittelyn ollessa yksi systeemyön kategorioista, tulee systeemyöhön kiinnittää siten huomiota. Tutkimusta tehdessä systeemyö ei varsinaisesti noussut terminä esille kirjallisuudessa tai haastatteluissa. Se kuitenkin tunnistettiin kummassakin edellä mainitussa kategoriassa.

Systeemyö on määritelty teoksessa Tietojärjestelmien kehittäminen tietojärjestelmiin liitettäväksi termiksi (Pohjonen 2002, 15). Tietojärjestelmätyö tarkoittaa muun muassa yritysten tietojärjestelmien määrittelyä, kehittämistä ja käyttöönottamista. Kun tätä peilataan niihin tuloksiin mitä tässä tutkimuksessa vaatimusten käsittelystä saatiin, voidaan todeta että tietojärjestelmien ja ohjelmistotekniikan teoksissa esitetyt teoriat ja käytännöt vaatimusten käsittelystä on sovellettavissa myös loistehon kompensointijärjestelmien vaatimusten käsittelyä tutkittaessa.

Systeemyö voidaan ymmärtää sisältävän kaikki perinteisen ”ATK:n” piiriin kuuluvat käsitteet. Toisin sanoen se sisältää perinteisemmän teollisuuden näkökulmasta katsottuna täysin erilaisia prosesseja ja työskentelymenetelmiä. Kuinka loistehonkompensointi voidaan sitten liittää systeemyöhön? Perehdyttäessä enemmän ohjelmistotekniikan pääpiirteisiin ja PETin laitteiston suunnitteluun voidaan tunnistaa jonkin verran yhtäläisyyksiä prosessi tasolla. Luonnillisesti suunnitteluohjelmat ja tekniikat ovat erilaisia, mutta loistehon kompensointilaitteiston ollessa pitkälle sulautettujärjestelmä on muun muassa muutostarpeet laitteistoalustaan yksi yhdistävä tekijä. Toinen tekijä on projektitoteutus, ohjelmistotekniikassa toimitetaan useasti ratkaisuja asiakkaalle projekti muotoisesti, niin myös loistehon kompensointilaitteistoissa.

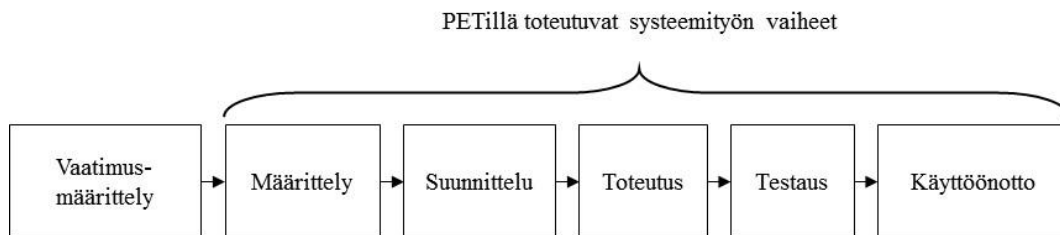
Modernissa systeemyössä painotetaan monesti niin sanottuja ketteriä työskentelymenetelmiä. Ketterienmenetelmien taustalla on tarve nopeisiin muutoksiin toteutuksessa. Muutokset vaativat hyvää ja säännöllistä kommunikointia, mikä onkin sisällytetty ketterien työmenetelmien perustoimintaan. Pentti Virtanen havainnollistaa artikkelissaan kuinka ohjelmistokehitysprojektit ovat aluksi epäonnistuneet niiden nojatessa perinteiseen projektinjohtamiseen, missä kaikkien vaiheiden tulisi olla tarkkaan ennustettavissa (Virtanen 2011, 15). Peilattaessa Virtasen kirjoitusta PETin toimitusprojekteihin voidaan löytää monia yhtäläisyyksiä ohjelmistotuotannon kanssa. Todennäköisesti yksi suurimmista yhteneväisyyksistä on tarve nopeaan reagointiin ja suunnanmuutoksiin järjestelmän suunnittelun aikana. Tutkimusta ei kuitenkaan ulotettu ketterien menetelmien tutkimiseen, koska tarkoituksena oli tutkia vaatimusten käsittelyn lähtökohtia.

Systeemyön perusajatuksena voidaan pitää toimintojen eri vaikutusasteisia kytkentöjä toisiinsa. Kuten aikaisemmin tässä kappaleessa todettiin, loistehon kompensointilaitteiston suunnittelu voidaan liittää systeemyön käsitteeseen. Suunnittelun lopputulokseen vaikuttavat: asiakas, projektia toteuttavat osapuolet ja monesti tuotekehityksellä on uusia ominaisuuksia, joita tulisi pystyä implementoimaan laitteistoon. Monessa projektissa järjestelmä tulee lopulta olemaan huomattavan erilainen siihen nähden mitä alkuperäisessä myynnin projektisuunnitelmassa on kuvattu. Toiminta vastaa Virtasen kuvailemaa systeemyötä, missä eri tahojen ja niin sanottujen palautekytkentöjen ohjaamana syntyy alkuperäisestä suunnitelmasta poikkeava ohjelmisto. (Virtanen 2011, 15)

4.2 Vaatimusten käsittelyn tärkeys

Useissa projektien hallinnan käsikirjoissa vaatimusmäärittely on todettu tärkeäksi kulmakiveksi projektin onnistumisen kannalta. Aikaisemmassa systeemyökappaleessa todettiin, että vaatimusmäärittelyä voidaan pitää yhtenä toimintaperiaatteena systeemyössä. Vaatimusmäärittelyn tarkoituksena on tarjota projektinjohdolle työkalu, millä voidaan hallita monimutkaiseksi kehittyneitä projektien vaatimuksia. Projektin hallinnoinnin kannalta on oleellista, että projektin aloituksesta lähtien ruvetaan suunniteltavalle laitteistolle keräämään teknisiä vaatimuksia. (Project management institute, 105)

Paula Männistö on luonut artikkeliinsa mielikuvituksellisen tapauksen, missä vaatimusmäärittelyn puuttuminen johtaa katastrofaaliseen lopputulokseen. Tarinan opetuksena Männistö pitää sitä seikkaa, että riittävän tarkalla vaatimusmäärittelyllä olisi selvitty myös artikkelin katastrofitapauksesta. Artikkelissa Männistö nostaa vaatimusmäärittelyn tarkoittamaan koko liiketoiminnan tarpeiden kartoittamista. Hän kuvaa kuinka vaatimusmäärittely on itseasiassa yksi osa niin kutsutusta systeemyöstä. (Männistö 2011, 59) Kuvassa 5 on esitettyä Paula Männistön artikkelissa esitetyt systeemyön vaiheet. Kuvaan on myös merkittynä tutkimustuloksien valossa löydetty, PETillä käytössä olevat systeemyön vaiheet.



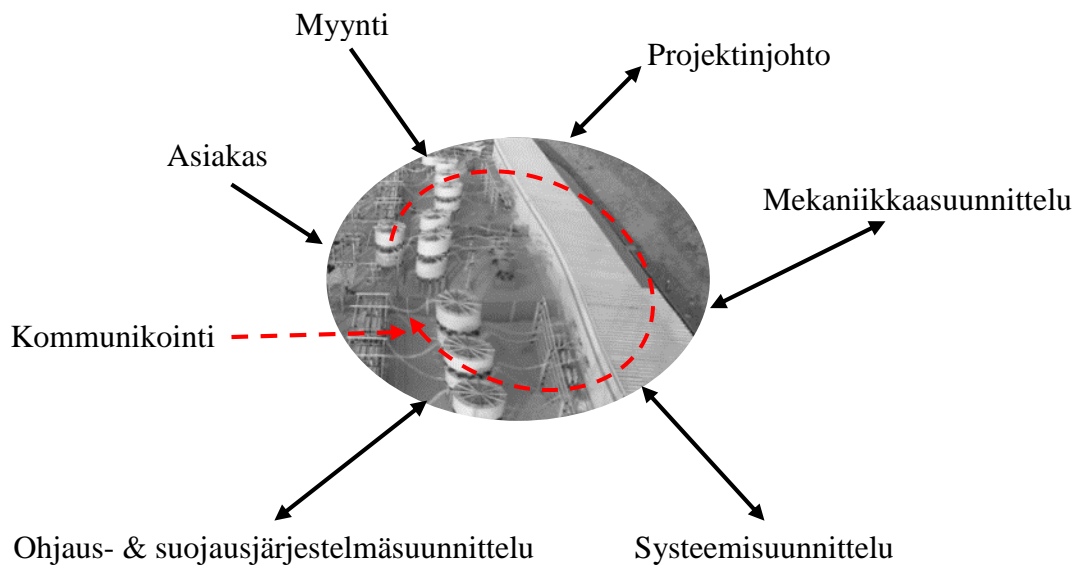
KUVA 5. PETillä toteutuvat systeemyön vaiheet (Männistö 2011, 59. Muokattu)

Artikkelissa esiintyneen kuvaajan perusteella (kuva 5) voidaan todeta kuinka vaatimusmäärittelyn tulisi olla ensimmäinen työvaihe ennen laitteiston määrittelydokumentaation kirjoittamista. Jätettäessä vaatimusmäärittely pois kaaviosta huomataan helposti suunnitteluprosessin vaillinaisuus. Jokaisessa suoritetussa haastattelussa tunnistettiin vaatimusten määrittelyn sijoittuminen suunnitteluprosessin ensimmäiseksi vaiheeksi, tämä tukee Männistö näkemystä. Useiden eri lähteiden viitatessa siihen, että vaatimusten määrittely tulee upottaa osaksi järjestelmien suunnittelua, voidaan todeta tämän kehitystehtävän aihe tärkeäksi elementiksi PETin suunnitteluprosessissa.

4.3 Suunnittelun sidosryhmät

Vaatimusmäärittelyyn perehdyttäessä tulee osata huomioida vaatimusmäärittelyn merkityksellisyys suunnittelun sidosryhmiin. Kirjassa *Model-Based Requirements* sidosryhmien merkitys nostetaan peräti yhdeksi tärkeimmistä vaatimusmäärittelyn toimivuutta selittävästä tekijöistä (Holt, Bronsword, Perry 2012, 9). Suunnittelunsidosryhmien tunnistaminen on merkittävää, sillä ne saattavat muovata laitteiston teknisien vaatimuksien sisältöä ja käyttöä. Vaatimuksien ei tarvitse olla kohdennettuja pelkästään laitteiston suunnittelijoille, sen sijaan niitä voidaan määritellä esimerkiksi ostolle ja tuotekehitykselle. (Holt, Bronsword, Perry 2012, 11). Ottaessa huomioon kehitystehtävän rajauksen, jätettiin kuitenkin osto ja tuotekehitys vaatimusmäärittelyn kehittämisen alkuvaiheen ulkopuolelle.

Sidosryhmien analysointi voidaan aloittaa ensiksi tunnistamalla eri sidosryhmät. Kuvassa 6 on määriteltynä millaisia sidosryhmiä PETillä työskentelee loistehon kompensointijärjestelmien suunnittelun parissa. Kuvassa sisällä oleva nuoli indikoi kuinka eritahojen välisen vuorovaikutuksen tuloksena kokonaisuus syntyy.



KUVA 6. PETin vaatimusmäärittelyn piirissä olevat sidosryhmät

Sidosryhmien merkitys PETin projektille on joko yksi- tai kaksisuuntainen. Yksisuuntaisessa kommunikaatiossa tulee ryhmältä suunnittelupiiriin pelkästään tietoa. Kaksisuuntainen sidosryhmä puolestaan tuottaa dataa muille sidosryhmille ja käyttää laitteiston suunnittelussa muiden ryhmien muodostamaa dataa hyväkseen.

4.3.1 Sidosryhmien analysointi

Sidosryhmien tunnistamisen lisäksi on oleellista suorittaa sidosryhmille riittävän tarkka sidosryhmäanalyysi. Analysoitaessa sidosryhmiä suunnittelun näkökulmasta, saadaan myös vaatimusmäärittelyn kannalta tarkempaa tietoa siitä millä tasoilla sidosryhmä vaikuttaa vaatimusmäärittelyyn. Model-Based requirements teos kirjaa tekniselle vaatimukselle minimi parametrit. Näihin parametreihin kuuluu muun muassa vaatimuksen omistajan kirjaaminen. (Holt, Bronsword, Perry 2012, 103) Sidosryhmienanalysoimisen avulla voidaan laajentaa ymmärrystä siitä millaisia omistajia vaatimuksille tulisi määritellä.

ASIAKAS

Loistehon kompensointilaitteistojen vaatimusmäärittelyn kannalta asiakkaan roolina on antaa järjestelmän suunnittelulle lähtötietoja. Asiakkaan toimittamasta määrittelydokumentaatiosta pyritään muodostamaan näkemys laitteiston vaatimuksista.

Asiakas näkee määrittelemänsä vaatimukset, kun suunnittelun ensimmäisessä vaiheessa laitteiston määrittelydokumentaation ensimmäinen versio lähetetään asiakkaalle hyväksyttäväksi. Asiakkaan roolina on myös antaa laitteiston suunnittelulle tarvittavat hyväksynnät, jotta projekti pääsee etenemään seuraaviin toteutusvaiheisiin. Asiakkaalle ei tunnisteta tarvetta määrittää varsinaisia teknisiä vaatimuksia.

MYYNТИ

Myynnin suhde suunnitteluun on pelkästään lähtödatan tarjoaminen. Projektin toimittussopimuksen synnyttyä on myynti organisaation tehtävänä järjestää niin kutsuttu ”transfer meeting”. Kokouksessa myynti luovuttaa laitteiston projektinnille. Projektin siirto vaiheessa käydään läpi kaikki esisuunnitteluvaiheessa muodostettu materiaali. Lisäksi selvitetään projektipäällikölle tarjousvaiheessa sovitut poikkeamat asiakkaan dokumentaatioon.

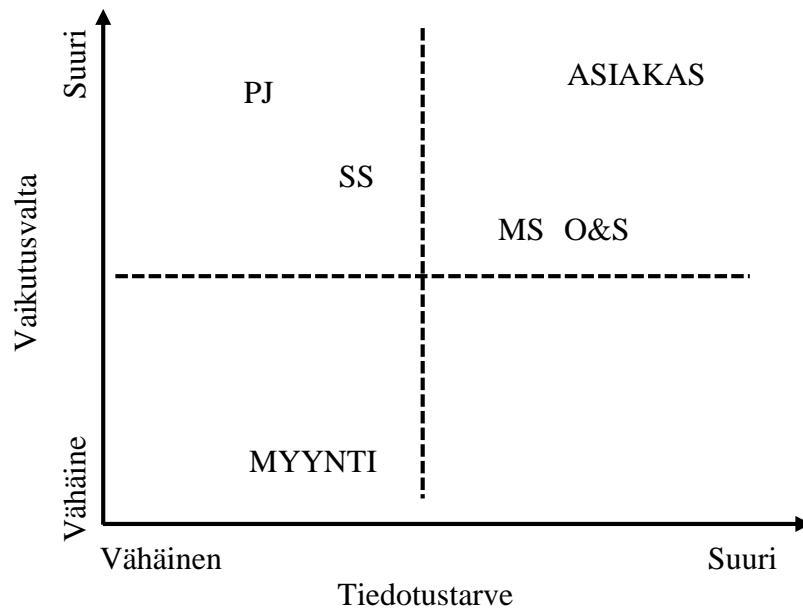
PROJEKTINJOHTO

Projektinjohto on laitteiston suunnittelun kannalta kriittisessä roolissa. Sen tehtävänä on koordinoida jokaista suunnittelun sidosryhmää ja koota yksittäisistä suunnittelun elementeistä toimiva kompensointijärjestelmä. Laitteiston määrittelyn näkökulmasta tulee projektinjohdon pysytä tuottamaan suunnittelun sidosryhmille vaatimuksia, mutta sen tulee myös pystyä ymmärtämään muiden sidosryhmien määrittelemiä vaatimuksia ja huomata vaatimusten aiheuttamat ristikkäisvaikutukset omaan suunnitteluun ja sen ulkopuolelle.

SUUNNITTELURYHMÄT

Suunnitteluryhmät asettuvat laitteistonsuunnittelun kannalta kaikista monipuolisempaan ryhmään. Niiden on pystyttävä tuottamaan suunnittelun määrittelydokumentaatiota ja sen pohjalta luomaan varsinaiset suunnitteludokumentit. Näitä kumpaakin toimintoa ohjaavat asiakkaan toimittamat määritysdocumentit ja vielä mahdolliset poikkeamat niihin. Suunnitteluryhmällä on myös tavoitteena pystyä adaptoimaan omaan määrittely- ja suunnitteludokumentaatioonsa toisien suunnitteluryhmien luomia vaatimuksia. Viimeisenä seikkana suunnitteluryhmille lasketaan laitteiston varsinaisten toiminnollisuuksien toteuttaminen. Toiminnollisuuksien toteuttaminen vaatii erilaista lähestymistä dokumentaation luomiseen verrattuna, suunnitteluryhmät ovatkin sidosryhmistä eniten kosketuksissa tuotantoon ja laitteiston loppukäyttäjiin. Näin voidaan todeta suunnitteluryhmien olevan kaikkien eniten verkostoituneita laitteistonsuunnittelun kannalta.

Kuvassa 7 on esitettyinä matriisimuodossa eri sidosryhmien suhtautuminen laitteiston suunnitteluun. Matriisissa Y-akselilla on kuvattuna sidosryhmän vaikutusvalta projektin suunnittelua ajatellen. X-akselilla on puolestaan kuvattuna kuinka paljon kyseinen sidosryhmä tarvitsee tiedotusta projektin suunnittelun eri osa-alueista. Kuvassa suunnitteluryhmät ovat vielä pilkottuna pienempiin osa-alueisiin.



KUVA 7. Sidosryhmämatriisi

Kuvassa 7 MS tarkoittaa mekaniikkasuunnittelua ja O&S ohjaus- ja suojaussuunnittelua. PJ on projektinjohto ja SS on systeemisuunnittelu.

PÄÄTELMÄT SIDOSRYHMIEN ANALYSOINNISTA

Sidosryhmäanalyysi sitoo yhteen käsityksen kohderyhmästä, jolle vaatimusten käsittely järjestelmää tulee kehittää. Suunnitteluprosessiin tulee ottaa mukaan kaikki PETin suunnitteluryhmät, koska ne ovat analyysin perusteella vaikutusvallan keskitason yläpuolella. Myynti on luokiteltu vaikutusvallaltaan vähäisimmäksi. Tämä johtuu siitä, että analyysin tavoitteena oli löytää teknisien asioiden osalta vaikutusvaltaisia ryhmiä. Analyysi ohjaa vaatimusten käsittelyn painotusta suunnitteluryhmien välillä seuraavan laiseksi:

- Systemisuunnittelu on enemmän määrittelemässä vaatimuksia
- Mekaniikka ja ohjaus- ja suojaussuunnittelu ovat pääpainoltaan toteuttamassa vaatimuksia.
- Projektijohto on määrittelemässä vaatimuksia ja seuraamassa niiden toteutusta.

Painotukset eivät suoranaisesti korostu kun tutkittiin vaatimusten käsittelyn edellytyksiä. Painotuksien ja sidosryhmien analyysin merkitys huomattiin tarpeelliseksi vaatimusten käsittelyn käytännön toteutusta tutkittaessa.

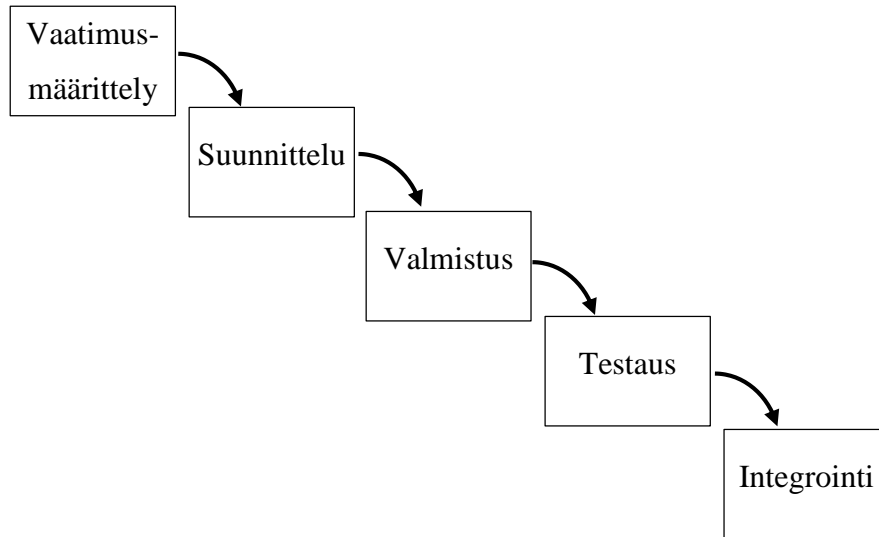
4.4 Prosessimallit

Kuten aikaisemmin todettiin, voidaan lähes jokaista loistehon kompensointilaitteistojen asiakas kohtaista räätälöinti projektia pitää lähestulkoon omana tuotekehitys projektinaan. Ohjelmistosuunnittelun parista voidaan helposti tunnistaa yhtäläisyyksiä PETin suunnitteluprosessin kanssa, nopeat muutokset suunnittelussa ovat muiden elementtien ohella ohjelmistosuunnittelusta muistuttava piirre. Vaatimusmäärittelyn kokonaisprosessin hahmottamisen kannalta on tarpeellista tutkia millaisia suunnitteluprosesseja perinteisissä ohjelmistoprojekteissa on totuttu käyttämään.

Ohjelmistotekniikassa perinteisesti käytetyt prosessimallit ovat vesiputousmalli ja spiraalimalli. Kirjassa *Essentials of Software Engineering* on kuvattuna kuinka nämä kaksi perinteistä prosessimallia on tyypillisesti muodostettu. Kirjassa kuvaillaan miksi prosessimallin hahmottaminen on laitteiston suunnittelun kannalta oleellista ja pureudutaan prosessimallien toimintaan. (Tsui & Karam 2007, 74) Pohdittaessa prosessimalleja loistehon kompensointilaitteiston suunnitteluun kannalta, voidaan todeta suunnitteluprosessin kuvaamisen ja vaatimusmäärittelyn upottamisen kyseiseen kuvaukseen olevan tärkeää. Hahmottamalla kokonaisuus pystytään sulauttamaan vaatimusmäärittelyjärjestelmä paremmin projektista toiseen toistuvaksi, eikä projektikohtaisille toimintatavoille jää tulkinvaraisuutta. Seuraavissa kappaleissa on tutkittu yleisimmiksi oletettuja prosessimalleja vaatimusmäärittelyprosessin muodostamisen tueksi.

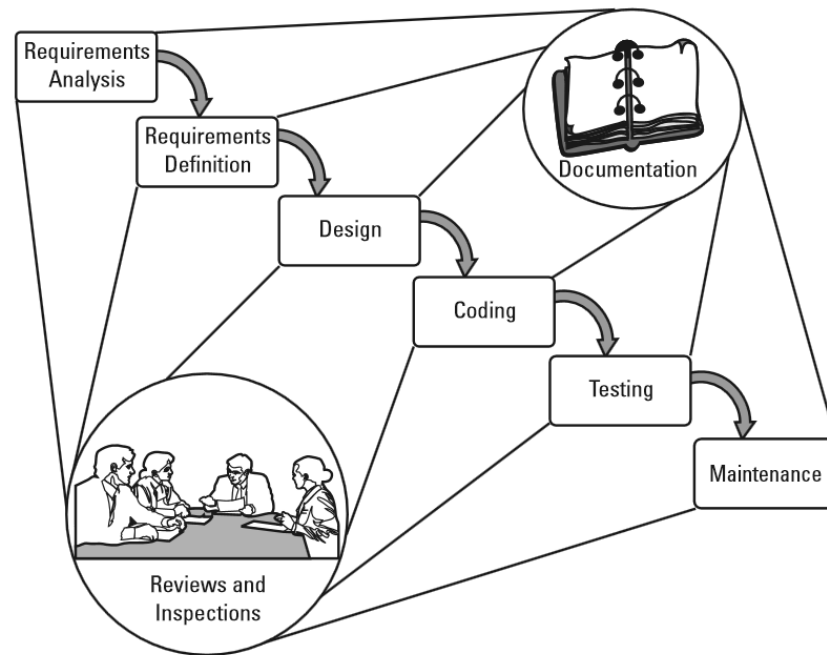
4.4.1 Vesiputousmalli

Vesiputousmalli on erittäin perinteinen tapa kuvata suunnitteluprosessia. Kuvassa 8 on esitettyä Essentials of Software Engineering kirjan mukaan millainen on perinteinen vesiputousmalli ohjelmistotekniikan alueella.



KUVA 8. Vesiputousmallinen prosessimalli (Tsui & Karam 2007, 77. Muokattu)

Vesiputousmalli on erittäin tyypillinen kuvaus suunnittelun vaiheista. Sen toimintaperiaatteena on, että toteutusaskelissa ei voida edetä ennen käynnissä olevan askeleen valmistumista. Vesiputousmalli mahdollistaa järjestelmän suunnittelutilanteen hahmottamisen erittäin helposti. Kun suunnittelu on jaettu riittävän pieniin ja realistisiin osiin, on kaaviosta helposti hahmotettavissa mikä on suunnittelun todellinen tilanne. Perinteisen vesiputousmallin heikkous on ollut, että siinä ei ole otettu huomioon mahdollista tarvetta palata suunnitteluportaisissa takaisin päin. Näin ollen mallista onkin muodostettu kehittyneempiä versioita, joissa on mahdollisuus palata portaisiin takaisinpäin suunnittelukatselmointien perusteella. Portaisissa voidaan palata alkua kohti useampia askeleita kerrallaan, mutta kaaviossa edetään aina askel kerrallaan. (Tsui & Karam 2007, 78) Kuvassa 9 on esitettyä kuvan 8 pohjalta millaiseksi vesiputousmallin kehittyneempi versio on kehitetty.



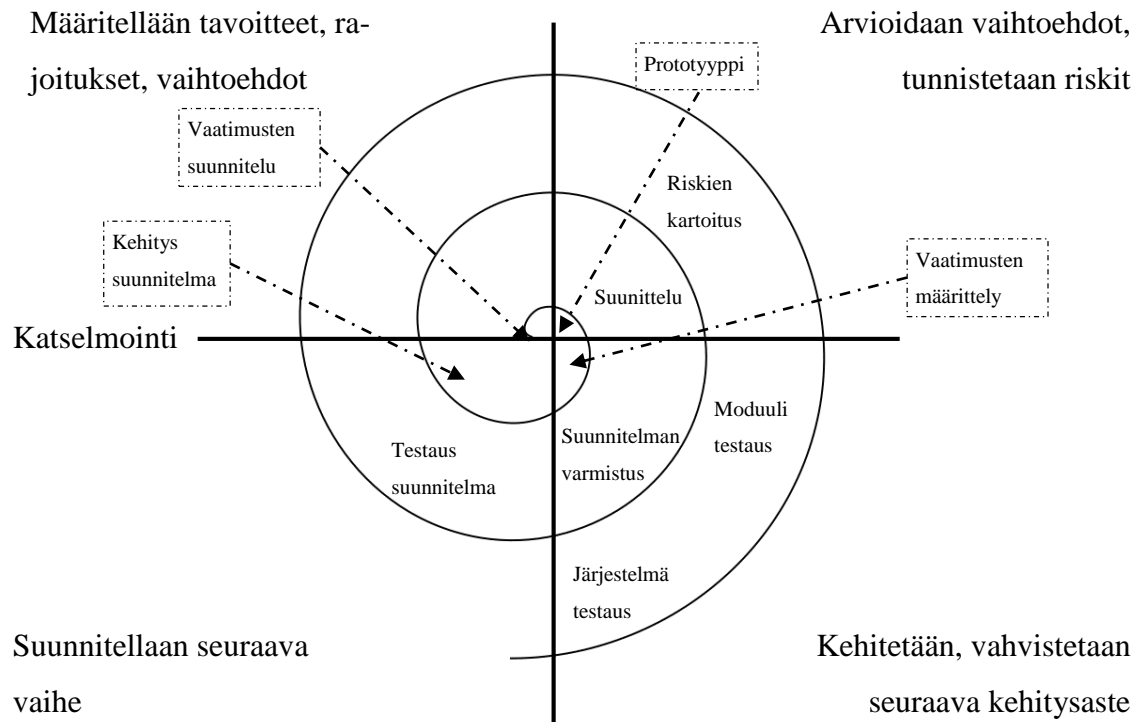
KUVA 9. Kehittyneempi vesiputousmalli (Rakitin & Steven 2001, 19)

Kehittyneestä vesiputousmallista voidaan huomioida, että projektin jokaisen askeleen kohdalla tehdään erillinen katselmointi, minkä perusteella suunnittelu voidaan palauttaa takaisin aikaisempiin portaisiin. Kuvan 9 malliin on myös upotettuna dokumentointi jokaiseen vaiheeseen. Laajoissa ohjelmistoprojekteissa dokumentointityö tulisivin saada upotettua varsinaisen työsuorituksen rinnalla kulkeväksi tehtäväksi.

Vesiputousmallin kiistattomia heikkouksia on, että se ei kuitenkaan näytä ohjelmistotuotannossa tyypillistä ohjelmistotuotannon ja ohjelmistotestauksen välistä jatkuvaa iterointia. Projektitoteutuksen kannalta oleelliset riskit jäävät myös vesiputousmallissa kirjauttamatta. (Rakitin & Steven 2001, 19) Kolmas huomattava haitta kuvan 9 vesiputousmallissa on, että vaatimukset oletetaan kerättynä heti projektin alussa eikä niitä sen jälkeen tarvitse tarkastella. Ottaessa huomioon PETin projektitoteutuksen saattaa vaatimuksia kerääntyä projektille vielä pitkään suunnitteluvaiheessakin.

4.4.2 Spiraalimalli

Spiraalimalli on toinen perinteisistä ohjelmistotekniikan prosessimalleista. Spiraalimallin lähtökohdista on vesiputousmallia joustavampi eteneminen ja sisäänrakennettu mekanismi riskien kartoittamiseen. Kuvassa 10 on kirjan *Essentials of Software Engineering* esitys spiraalisesta prosessista.

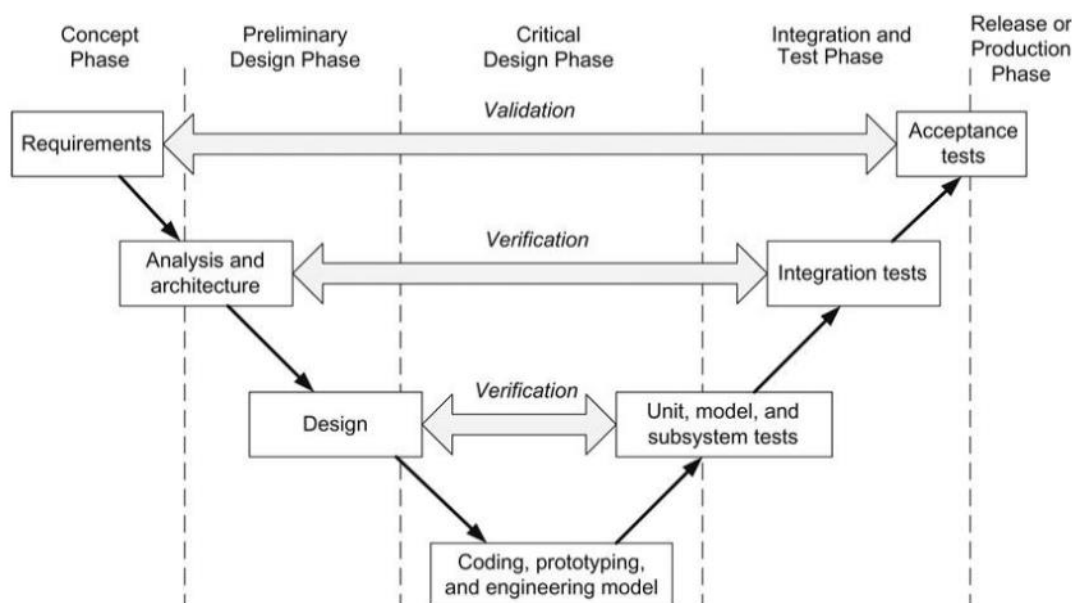


KUVA 10. Spiraalimallinen prosessi (Tsui & Karam 2007, 81. Muokattu)

Tutkittaessa kuvassa 10 esitettyä spiraalimallista suunnitteluprosessia huomataan sen olevan monimutkaisempi kuin aikaisemmin esitetty vesiputousmalli. Sen kehittämisen lähtökohdista on ollut ohjelmistotuotannon riskien parempi huomioiminen. Spiraalimalliin kuuluu alkuperäisessäkin versiossa jaksoittain toistuva katselmointi tärkeiden suunnitteluportaiden jälkeen. Spiraalimallin vahvuutena on, että mallia noudatettaessa muodostetaan prototyyppi, jonka avulla riskien määrittäminen on helppoa (Tsui & Karam 2007, 81). Toisaalta kirjassa *Software Verification and Validation for Practitioners and Managers* on esitetty, kuinka spiraalimallin noudattaminen saattaa olla riskien kartoitusmielessä erittäin haastavaa (Rakitin & Steven 2001, 22).

4.4.3 V-malli

V-malli on myös tunnettu yhtenä ohjelmistotekniikan prosessimallina. Sen vahvuutena on mallin yksinkertainen sovellettavuus muun muassa sulautettujen järjestelmien suunnitteluun. Kuvassa 11 on esitettyä perinteinen V-malli kirjan *Developing and managing Embedded Systems and products* (Fowler 2014, V-model).



KUVA 11. V-muotoinen prosessimalli (Fowler 2014, V-model).

Prosessimalli voidaan hahmottaa helposti V-muotoiseksi, kuten kuvassa 11 on tehty. V-muoto tukee myös toimintamallia käytännössä. Mallissa lähdetään korkeimmalta tasolta, missä määritellään vaatimukset joita systeemin tulee noudattaa. Järjestelmän suunnittelu ja valmistus etenee aina vain yksityiskohtaisempiin suunnitteluportaisiin. Yksityiskohtaisimmat seikat kerääntyvät V-mallin pohjalle. Aloitettaessa testaamaan ensiksi pienempiä osia ja sitten myöhemmin integroimaan kokonaisuutta, ruvetaan jälleen nousemaan projektissa korkeammalle tasolle. Lopuksi projektista poistutaan asiakashyväksynnän kautta. Mallissa toimintavaiheet ovat vielä jaettuna omiin pysty segmentteihin joiden tarkoituksena on kuvata sen hetkisen toteutusvaiheen kohtaa koko toteutuksen kannalta.

Kirjassa *Developing and managing Embedded Systems and products* esitetään V-mallin heikkoutena yksinkertaista vaatimuksien huomioimista. Mallin mukaan vaatimuksia ei toteutusvaiheessa enää päästä päivittämään (Fowler 2014, V-model). Vaatimuksien päivittäminen on monessa toimitusprojektissa erittäin oleellisessa roolissa. Valmistettavan

laitteiston ollessa monimutkainen ja mahdollisesti asiakkaallekin uusi saattaa vaatimukset päivittää sitä mukaan kun asiakas pääsee katselmoimaan järjestelmän suunnitteludokumentaatiota.

4.4.4 Ketterät menetelmät

Pentti Virtanen on kirjoittanut projektiyhdistyksen julkaisussa siitä kuinka monimutkaisissa ja monikansallisissa projekteissa projektien ohjaamisen tulisi muuttua automaattisen, vaistonvaraiseen toimintaan. Vertauskuvana hän käyttää villieläinlaumaa, joka ilman erilisiä käskyjä osaa muuttaa toimintatapoja rauhallisesta syömisestä vaaraan väistämiseen. Hän peräänkuuluttaa artikkelissaan itseohjautuvuutta tiimeiltä. (Virtanen 2011, 15).

Aikaisemmissa kohdissa esitettyjen perinteisten prosessimallien rinnalle on viime vuosina noussut uusia menetelmiä, joita monesti kutsutaan ”agile” menetelmiksi. Näitä voidaan kutsua niin sanotuiksi ketteriksi prosessimalleiksi. Menetelmissä perus lähtökohtana on, että ne eivät suoranaisesti sisällä tiukasti rajattua toimintamallia. Kyse on paremminkin toimintatavoista ja kehitysprojektin automaattisesta kommunikointityylistä. (Stellman & Greene 2015, Chapter 1. Learning Agile)

Ketterissä malleissa on yhdistävänä tekijänä nopeasti muuttuva suunnittelu ympäristö. Kirjassa Agile Project Management : How to Succeed in the Face of Changing Project Requirements kuvataan kuinka perinteinen projektin johtaminen ei yksinkertaisesti sovellu nopeasti muuttuvaan suunnittelukenttään. Kirjan mukaan perinteinen tiukasti aikataulutettu projektinjohtaminen tarvitsee tuekseen soveltavia menetelmiä, nopeiden muutoksien hallintaan. (Chin 2003, 3)

Seuraavat seikat voidaan laskea ketterien prosessimallien yleisiksi eduiksi (Tsui & Karam 2007, 108).

1. Tehtävät voidaan jakaa pieniin osiin, jolloin kokonaisuudesta saadaan yksinkertaisemman oloinen. Pienempiä osakokonaisuuksia on helpompi hallita.
2. Lähtötietojen ollessa puutteellisia voidaan ketterillä menetelmillä aloittaa järjestelmän suunnittelu. Menetelmien avulla voidaan palata takaisin jatkamaan osa-alueiden suunnittelua kunnes tietoa kokonaisuudesta on riittävästi kertynyt

3. Loppukäyttäjät tuodaan tyypillisesti mukaan järjestelmän suunnitteluun, tällöin lopputuloksesta saadaan parhaiten asiakkaan vaatimuksia palveleva.
4. Ohjelmistoprojekteissa ketterillä menetelmillä voidaan korvata raskasta dokumentointitarvetta.
5. Tiedonsiirtyminen ketterissä menetelmissä on tehokasta, mutta kevyttä. Menetelmät eivät vaadi toimiakseen raskasta dokumenttien hyväksymisprosesseja.
6. Muutoksien hallinta on ketterillä menetelmillä helppoa. Kaikkien ylempien summana saadaan nopea vasteaika muuttuviin ja tarkentuviin vaatimuksiin.

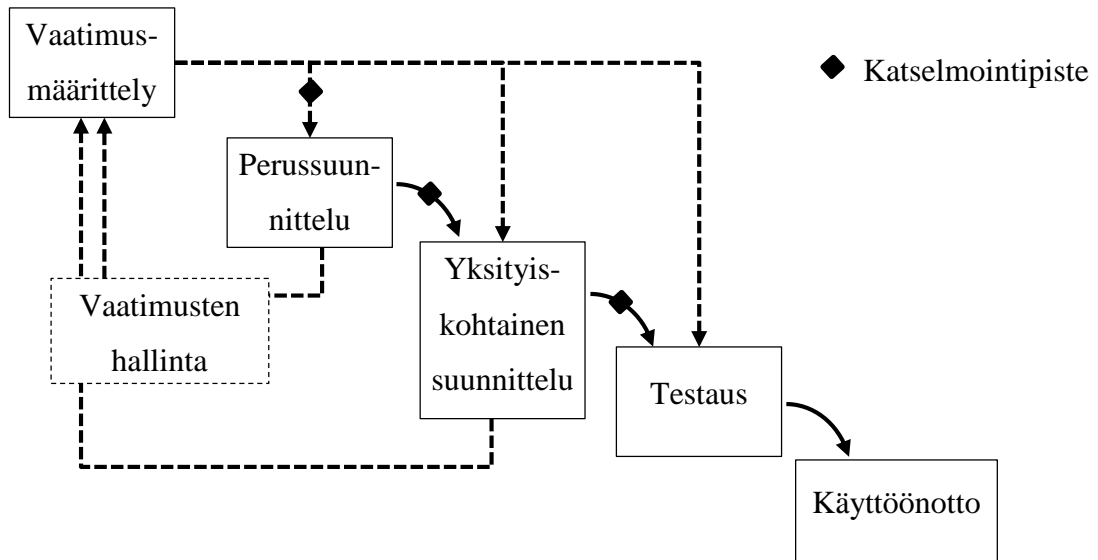
Vaikka ketterille menetelmille löydettiin kirjallisuudesta ja lehdistä paljon positiivisia tekijöitä jätettiin ketterien menetelmien käsitteleminen tämän tutkimuksen ulkopuolelle. Ketterien menetelmien pois jättämistä voidaan perustella kahdesta eri näkökulmasta. Ensimmäiseksi se voidaan lukea kehittyneeksi suunnitteluprosessiksi. Koettiin, että perusteiden puuttuessa ei kehittyneeseen menetelmään voida hypätä ilman, että ensin luodaan perusteet. Toinen ketterien menetelmien poisjättämistä perusteleva tekijä oli haastattelujen sisältö. Enemmistön haastateltavista ei tunnistanut ketterien suunnittelumenetelmien käyttöä teollisuuden projekteissa.

4.4.5 Prosessimallien teorian liittäminen tutkimukseen

Vaikka perinteistä vesiputousmallia pidetään monessa kirjallisuuslähteessä vanhanaikaisena, saadaan siitä rakennettua PETille toimiva pohja vaatimusten keräämiselle sekä niiden hyödyntämiselle projekteissa. Kuten kirjassa *Software Verification and Validation for Practitioners and Managers* todettiin, voidaan perinteiseen malliin lisätä muun muassa katselmointia jokaiseen prosessin askeleeseen (Rakitin & Steven 2001, 19). Jotta mallista saataisiin vielä PETin toimintaa paremmin palveleva, tulee siihen vielä lisätä riittävät iteroitukierrokset suunnittelumuutoksia varten.

Loistehon kompensointilaitteiston suunnitteluun kohdistuvat muutokset ovat usein yllättäviä ja saattavat vaikuttaa merkittävästi jo tehtyyn suunnitteluun. Aikaisemmassa kapaleessa, missä käsiteltiin ketteriä kehitysmenetelmiä, kerrottiin ketteryyden soveltuvan parhaiten muutosriikkaaseen ympäristöön (Stellman & Greene 2015, Chapter 1. Learning Agile). Ketteryyttä on vaikea kuvata perinteisessä vesiputousmallissa, silti niistä tuttua tehokasta kommunikaatiota tulisi pystyä lisäämään perinteiseenkin vesiputousmalliin.

Kuvassa 12 on esitettyä millainen PETin suunnitteluprosessista tulisi saada, jotta vaatimusmäärittelyn tuomat edut saataisiin tehokkaasti sulautettua toimintaa tukemaan.



KUVA 12. Prosessimalli sovellettuna PETin suunnitteluportilla

PETille sovellettavassa prosessimallissa on kuvattuna loistehon kompensointilaitteiston toimittamiseen soveltuvat askeleet. Kuten lähes kaikissa lähteissä painotettiin, portaisiin lisätään uudessa prosessissa vaatimusmäärittely ensimmäiseksi tehtäväksi. Vaatimusmäärittelyporras PETin projekteissa tulee tarkoittamaan, että perussuunnitteluun ei tulisi edetä ennen kuin vaatimusmäärittelyn avulla on kuvattuna riittävän tarkasti suunniteltava systeemi. Matti Vuori kirjoittaa artikkelissaan vaatimusmäärittelyn huonoista käytännöistä. Artikkelin mukaan vaatimusmäärittelyn taustaprosessien suunnittelu ja kuvaaminen ovat avaintekijänä vaatimusmäärittelyn lopullisessa onnistumisessa (Vuori 2009, 20).

Prosessiin on haettu ketteristä menetelmistä poimittua joustavaa kommunikaatiota suunnittelun katselmointipisteillä. Katselmointipisteen tarkoituksena on kutsua koolle projektin suunnittelussa olevat suunnitteluryhmät ja katselmoida sen hetkinen suunnittelun tilanne vaatimuksia hyödyntäen. Katselmointipisteellä kontrolloidaan myös suunnittelun osa-alueiden hyväksymistä seuraavaan vaiheeseen etenemistä varten.

Ohjelmistotekniikassa esiintyvää suunnittelun iterointia, on uudessa prosessimallissa hahmotettu takaisin kytkentä nuolilla. Tarkoituksena on, että suunnittelun edetessä laitteiston vaatimukset tarkentuvat. Kun uusia vaatimuksia tarvitsee luoda tai jo olemassa olevia vaatimuksia tarvitsee tähdentää, on suunnitteluprosessissa käytössä vaatimusten

hallinta. Hallinnalla tarkoitetaan, että projektin teknisestä toteutuksesta vastaava ottaa vastuun muutoksien kirjaamisesta ja niiden jalkauttamisesta takaisin suunnitteluun. Tämä ei tarkoita kuitenkaan sitä, että projektin vetäjä tai teknisestä toteutuksesta vastuussa olevat kirjaa muuttuneet vaatimukset.

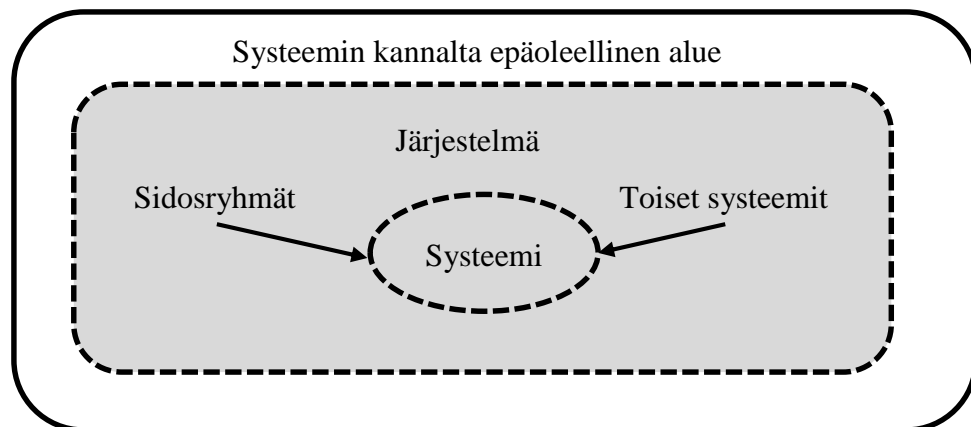
4.5 Järjestelmän ja systeemin erottaminen

Kirjassa Requirements Engineering Fundamentals on esitettyä kuinka vaatimuksien määrittämisen kannalta on oleellista pystyä tunnistamaan määriteltävä laitteisto kaikesta muusta sitä ympäröivästä järjestelmästä. Jos suunniteltavan järjestelmän rajojen hahmottamiseen ei kiinnitetä huomiota, saattaa kerätyt vaatimukset olla irrelevantteja ja täten johtaa suunnittelua kohti vääriä päämääriä. (Pohl & Rupp 2015, System and Context Boundaries)

Requirements Engineering Fundamentals selittää millaisilla kriteereillä voidaan järjestelmän rajoja tunnistaa. Alettaessa hahmoittamaan suunniteltavaa järjestelmää, voidaan ottaa käyttöön kaksi eri termiä: system boundary (systeeminrajat) ja context boundary (järjestelmänrajat). Systeeminrajat voidaan määrittää siten, että systeemi on se laite tai kokonaisuus mitä ollaan suunnittelemassa. Tunnistamisessa merkittävä seikka on, että systeemin sisäpuolella olevaan asiaan pystyttään vaikuttamaan.

Järjestelmänrajojen puolestaan voidaan kuvata olevan se ympäristö, mistä vaatimuksia systeemille muodostetaan. Toisin sanoen järjestelmäalueella olevat seikat eivät ole muutettavissa, niitä voidaan pitää suunnittelun rajoituksina. (Pohl & Rupp 2015, System and Context Boundaries) Järjestelmä on siis se alue, mikä vaikuttaa systeemin suunnitteluun, järjestelmä alueella toimivat kaikki tunnistetut suunnittelunsidosryhmät. Järjestelmä alueelle voidaan kuitenkin poikkeuksena määrittää toiset systeemit (Pohl & Rupp 2015, System and Context Boundaries). Suunniteltavalla systeemillä saattaa olla rajoittavana tekijänä toinen kehitettävä systeemi, mikä on saman järjestelmän alaisena. Tällöin tulee tunnistaa näiden systeemien välillä olevat rajapinnat ja hahmottaa kahden systeemin toisilleen muodostamat rajoitukset.

Kolmanneksi tulee vielä tunnistaa mikä on se alue, joka voidaan täysin jättää pois systeemin vaatimuksia mietittäessä. Todellisuudessa rajoja tuskin pystytään määrittämään kovin kiinteästi. Vaatimusten ja projektien edetessä saatetaan joutua muutamaa systeemin ja järjestelmän välistä rajausta. Ei siis ole täysin pois suljettua, että suunnitteluprosessin aikana epäoleellisena pidetyltä alueelta joudutaan nostamaan tekijöitä järjestelmä alueelle. Tällaisia tekijöitä voivat olla esimerkiksi säädösten ja lakien muuttuminen (Pohl & Rupp 2015, System and Context Boundaries). Kuvassa 13 on visualisoitu mitä systeemin rajoja kuvaavilla termeillä on tarkoitettu.



KUVA 13. Systeemin ja järjestelmän välinen raja (Pohl & Rupp 2015, System and Context Boundaries. Muokattu)

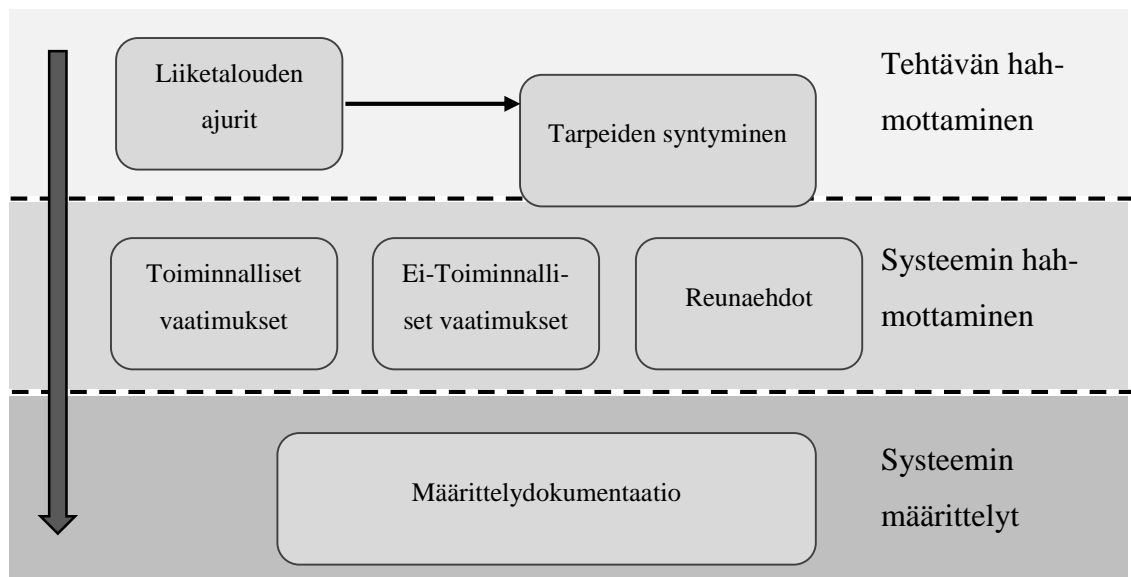
4.5.1 Loistehon kompensointijärjestelmien systeemin ja järjestelmän tunnistaminen

Loistehon kompensointijärjestelmien vaatimustenmäärittelyn kannalta voidaan tunnistaa haasteelliseksi epäoleellisen alueen määrittäminen. Monessa PETin projektissa laitteisto suunnitellaan yhteistyössä kohdemaan yksikön kanssa. Tällöin projektissa joudutaan rajaamaan vastuualueita erilaisin lausekkein ja taulukoin. Tämä ei kuitenkaan tarkoita, sitä että PETin toimitusrajan ulkopuolelle rajattuja tekijöitä voitaisiin suunnittelussa täysin unohtaa. Vaatimusten kartoittamisessa ei siten voida rajoittua pelkästään tunnetun järjestelmän sisäpuolelle. Tämän lisäksi joudutaan järjestelmän vaatimuksiin useissa tapauksissa sisällyttämään saman järjestelmän sisällä olevan toisen systeemin vaatimuksia. Kohdemaan yksikön ollessa monesti tiiviimmin yhteistyössä loppuasiakkaan kanssa, on heillä tiedossa kirjoittamattomia toimintatapoja. Nämä toimintatavat eivät ole kirjattuna asiakkaan toimittamiin määrittelydokumentteihin. Täten muodostuu

linkki toimitettavan järjestelmän sisälle, joka vaikuttaa kumpaankin systeemiin. Projektipäällikön toimii erittäin kriittisessä roolissa tietojen jakajana, jotta vaatimusten määrittäminen näiden kahden systeemin välillä onnistuu.

4.6 Vaatimusten erilaiset tyypit

Teos Ohjelmistotuotannon käytännöt määrittelee vaatimuksen seuraavalla tavalla: ”Vaatimus (requirement) on jotain, mitä tuotteella pystyy tekemään, tai ominaisuus, joka tuotteella tulee olla.” (Haikala & Mikkonen 2011, 61) Ennen tarkempaa perehtymistä vaatimusten käsittelyyn tarvitsee hahmottaa vaatimusten luokittelu perusteet. Vaatimuksien tyyppien tunnistamisen tärkeys korostuu, kun vaatimuksia kerätään isoja määriä ja niiden kategorioiminen on välttämätöntä. Erittäin isoihin projekteihin tulisikin luoda aina oma vaatimus hierarkia, minkä toimisi vaatimuksien hallinnan runkona. Kuvassa 14 on havainnollistettu vaatimuksien luokittelu tyyppihin ja vaatimusten sijoittuminen suunnittelu prosessissa.



KUVA 14. Vaatimuksen kehitysmalli (Robertson & Robertson 2010, 11. Muokattu)

Useat vaatimusmäärittelyä käsittelevät lähteet jakavat vaatimukset samoilla perusteilla toiminnallisiin ja ei-toiminnallisiin vaatimuksiin. Kolmannen kategorian vaatimusten luokittelulle muodostaa niin kutsutut reunaehdot. Alla olevassa listauksessa on selitettyä Mastering the Requirement Process kirjan mukaisesti eri vaatimustyyppit loistehon kompensointilaitteistolle (Robertson & Robertson 2010, 9).

TOIMINNALLINEN VAATIMUS

Toiminnalliset vaatimukset toteuttavat asioita, joita suunniteltavan laitteiston tulee toteuttaa. Tyypillisesti toiminnallinen vaatimus on se lopputulos mitä asiakas toivoo suunniteltavan laitteiston tekevän. Loistehon kompensointijärjestelmän tasolla toiminnalliseksi vaatimuksiksi voi olla määrittelyt erilaisista säätötavoista tai käsikäyttöpaneelille määrittelyt toiminnot.

EI- TOIMINNALLINEN VAATIMUS

Ei-toiminnalliset vaatimukset kuvaavat millaisia suunniteltavan systeemin ominaisuuksien tulee olla. Ne sisältävät esimerkiksi määritteitä reagointiajoista, ulkonäköseikkoja tai rajaavat suunnittelussa käytettäviä standardeja. Ei-toiminnalliset vaatimukset muodostavat vaatimusryhmän, jotka sanelevat systeemin toiminnan kannalta oleellisia attribuutteja. Tarkasteltaessa ei-toiminnallisia vaatimuksia loistehon kompensointijärjestelmän tasolla voidaan ei-toiminnallisiin vaatimuksiin listata esimerkiksi loistehon säädön askelvasteen asettumisaika tai järjestelmän tapahtumalokin minimi aikaresoluutio.

REUNAEHDOT

Reunaehdoiksi luettavat vaatimukset rajoittavat systeemin suunnittelua kokonaisvaltaisesti. Reunaehtoina voidaan pitää enemmän yleismaailmallisia tekijöitä, kuten esimerkiksi lait ja säädökset joita tulee noudattaa. Reunaehdoksi määritellään myös systeemin toimitusaika. Peilattaessa reunaehtojen määrittelyä loistehon kompensointijärjestelmiin nousee yhdeksi merkittävimmäksi tekijäksi kansainväliset IEC- ja IEEE-standardit. Asiakkaiden määrittelydokumentaatioissa määritellään useassa tapauksessa kompensointijärjestelmän toiminta siten, että sen tulee täyttää kansainvälisien standardien määrittelemät minimi vaatimukset.

LUOKITUKSEN MERKITYS TUTKIMUSTULOKSIEN VALOSSA

Kirjallisuutta tutkittaessa luokkien käyttämiselle ei löydetty aluksi selviä hyötynäkökulmia. Tutkimuksen aikana syventynyt ymmärrys vaatimusten käsittelystä auttoi kuitenkin hahmottamaan luokittelun hyötynäkökulmaa. Tutkittaessa ja pohdittaessa vaatimusten luokittelua voitiin todeta, että PETin toimintamallissa kaikki luokat ovat tarpeellisia. Kuvassa 14 esitetty jaottelu toteutuu lähes kuvatun mukaisesti PETin toiminnassa. Tämän tutkimuksen kannalta oleelliseksi kohdaksi tunnistettiin systeemin hahmottamisvaiheen erilaiset vaatimusluokat (keskimmäinen kohta kuvassa 14).

Kun tutkittiin vaatimusten käytännön toteutusta, huomattiin vaatimusten luokilla olevan oma merkittävä rooli. Niiden merkitys korostuu, kun työkiertoja ja tarkastuspisteitä vaatimuksille ollaan määrittelemässä. Vaatimuksen luokan perusteella voidaan päätellä tarvitseeko sille määritellä työkiertoon erityisiä testausvaiheita. Myös haasteluissa Heikkinen painotti vaatimusten luokkajaon tärkeyttä (Heikkinen 2016)

Vaatimusten jakamisen eri luokkiin voidaan olettaa myös vaikuttavan resurssien suunnitteluun. Tutkimuksen perusteella voidaan nähdä, että osa suunnittelijoista keskittyy pelkästään teoriatasolla tai niin sanotusti työpöytätasolla laitteiston suunnitteluun. Heille voidaan ohjata ratkaistavaksi vaatimukset joissa keskitytään reunaehtoihin tai vaatimukset ovat ei-toiminnallisia. Toiminnalliset vaatimukset voidaan puolestaan ohjata sellaisille suunnittelijoille, jotka osallistuvat toimintojen suunnitteluun ja testaamiseen. Peilattaessa tätä ajatusta Jira-järjestelmän toiminnan tutkimista saatuihin tuloksiin, voidaan todeta, että Jira-järjestelmän luoma mahdollisuus osoittaa vaatimuksia eri ihmisille ratkaistavaksi tukee luotua ymmärrystä luokkien käyttämisestä.

5 SUUNNITTELUPROSESSIIN PEILAAMINEN HAASTATTELUIDEN KAUTTA

Tässä kappaleessa puretaan haastattelujen tulokset ja pohditaan niiden perusteella vaatimusten käsittelyä tukevien seikkojen merkitystä PETin suunnittelun kannalta. Kirjallisuustutkimuksen tueksi haluttiin löytää hyviä kokemuksia vaatimusten käsittelyn parissa työskenteleviltä. Haastattelut pidettiin avoimina haastatteluina, jotta ajatuksia ja kokemuksia jaettaisiin mahdollisimman monesta eri näkökulmasta. Haasteltavien taustoja analysoitaessa huomattiin, että he edustivat vaatimusten käsittelyä monesta eri näkökulmasta. Lisäksi haastateltavilla huomattiin olevan usean vuoden verran kokemusta vaatimusten hallinnasta ja vaatimusten keräämisestä. Tätä voidaan pitää positiivisena tekijänä haastatteluiden tuloksien merkityksellisyyttä mietittäessä. Haastateltavat edustivat konepajateollisuutta, tietoliikennealaa ja automaatioalaa. Haastattelujen jäsentelyyn ja tuloksien analysoinnin tukena käytettiin mind-map tekniikkaa. Liitteessä 4 on esitettyä analysoinnin tukena käytetty mind-map kaavio.

5.1 Haastattelut

MARKO KARJALAINEN

Marko Karjalainen edusti konepajateollisuuden näkökulmaa vaatimusten hallinnasta. Karjalainen työskentelee tiiviisti vaatimusten käsittelyn parissa. Hän ei ole suoraan tekemisissä ulkoisien asiakkaiden parissa, vaan työtehtävät sijoittuvat ratkaisujen etsimiseen sisäisen asiakkaan tilauksiin. Karjalaisen edustamassa yrityksessä valmistetaan tuotteita, joille on jo valmiina niin sanottu tuotealusta. Käytännössä tämä tarkoittaa jo olemassa olevaan tuotealustaan tehtäviä muutoksia. Karjalaisen tapauksessa sisäiset asiakkaat tilaavat Karjalaisen edustamalta suunnitteluryhmältä muutoksia tuotealustaan. Muutoksilla pyritään vastaamaan ulkoisien asiakkaiden vaatimuksiin. Karjalainen kommentoi, että vaatimusten hallinnan perimmäisenä tarkoituksena on vastata siitä, että oikeat sidosryhmät reagoivat muuttuneeseen ympäristöön (Karjalainen 2016).

Vaatimusmäärittelyprosessia voidaan soveltaa niin yrityksen ulkoisien asiakkaiden projekteihin kuin sisäisienkin asiakkaiden tilaamiin töihin, tämä tekee vaatimusmäärittelystä monipuolisesti hyväksikäytettävän suunnittelun tukiteknikan. Vaatimusmäärittelyn kuuluessa sisäisien asiakkaiden projekteihin on useasti kyseessä kehitysprojekti. Karjalaisen edustamassa yrityksessä sisäiset asiakkaat liittyvät nimenomaan kehitysprojekteihin. Niissä vaatimuksienmäärittelyllä kartoitetaan ensin millaista ominaisuutta ollaan toteuttamassa. Tarvittavat uudet ominaisuudet kirjataan vaatimuksiksi. Tämän jälkeen vaatimuksien kustannusta ja toteutusta arvioidaan tuotepäälliköiden ja suunnittelijoiden toimesta ennen kuin niitä aloitetaan varsinaisesti toteuttamaan. (Karjalainen 2016)

Tuotteen vaatimuksien analysoinnin ja hyväksymisen jälkeen aloitetaan vaatimusten avulla määrittelydokumentaation laatiminen. Vaatimuksia tullaan suunnittelun myöhäisemmässä vaiheessa hyödyntämään ali-tehtävien teettämisessä. Ali-tehtävät voidaan jakaa jopa suoraan alihankinnan tehtäväksi, kunhan ne ovat riittävän selviä kokonaisuuksia.

Kerätyt vaatimukset seuraavat Karjalaisen kokemuksissa kehitysprojektia aina valmiin kokonaisuuden testaukseen asti. Silloin vaatimuksia käytetään yhtenä testattavan laitteen arviointikriteerinä. (Karjalainen 2016) Karjalaisen haastattelun perusteella voidaan tunnistaa, että myös PETillä on sisäinen tilaaja ja toimittaja suhde projektisuunnittelun ja tuotekehityksen välillä. Rajapinta ei tosin ole aktiivisesti käytössä ja sitä on vaikea hahmottaa. Viitaten aikaisempaan kappaleeseen 4.4.5 (prosessimallien teorian liittäminen tutkimukseen) projektitoteutuksen vaatimustenkeräämisvaiheessa tulisikin pystyä tunnistamaan erilleen projektitoteutuksen ja tuotekehityksen vaatimukset. Näin ollen tulisi kytä haaromaan projektin vaatimukset ja tuotekehitysvaatimukset erilleen.

MIKA KOLEHMAINEN

Mika Kolehmaisien kokemukset perustuvat automaatioprojektien vaatimusten käsittelyyn. Hänen haastattelussa selvisi, että niin sanottua vesiputousmallia sovellettiin myös hänen edustamassa yrityksessä. Vesiputousmallinen vaatimusten käsittely tuki Kolehmaisien aikaisemmissa kokemuksissa koko suunnitteluprosessin elinkaarta. Vaatimukset projektille alkoivat hahmottua jo järjestelmän myyntivaiheessa. Verrattaessa tätä PETin toimintaan oli Kolehmaisella kokemusta niin sanotun kyselykaavakkeen käyttämisestä jo myyntivaiheessa. Kyselylomakkeella pystyttiin automaatioprojekteissa rajaamaan ja valitsemaan

perusominaisuudet järjestelmälle. Kyselylomake toimii silloin ensimmäisenä projektin vaatimuksia kokoavan järjestelmänä.

Projektit olivat Kolehmaisien kokemuksissa hieman pienempiä kuin mitä PETillä, näin ollen myös hänen aikaisemmat kokemukset perustuivat taulukkopohjaiseen vaatimusten käsittelyjärjestelmään. Myyntivaiheessa kyselylomakkeella kartoitetut vaatimukset siirtyivät taulukkopohjaiseen vaatimusten hallintajärjestelmään.

Kolehmainen kommentoi, että vaatimusten kartoittamista ei voida aloittaa ennen kuin kaikki projektiin liittyvä materiaali on tiedossa ja toimittajalla käytettävissä. Materiaalin selvittäminen ja lajittelu voidaan lukea projektipäällikön vastuualueelle. Kaikki kerätty materiaali ja sen sijainti tulisi olla jokaiselle projektiin osallistuvalla selvillä. Projektin aloituskokousta voidaan pitää sopivana tilaisuutena tiedon jakamiseen. (Kolehmainen 2016)

Kolehmaisien haastattelussa korostui standardoidun tuotealustan tarpeellisuus vaatimusten käsittelyssä. Kolehmaisien mukaan standardialustan avulla vaatimusten kartoittaminen on huomattavasti jäsennellympää. Jos standardi konseptia ei ole määriteltyinä on erittäin vaikeaa vetää rajaa siihen mitä kaikkea merkataan projektin tekniseksi vaatimukseksi.

JUHO HEIKKINEN

Juho Heikkisen kokemukset perustuivat tietoliikenne- ja ohjelmisto alalle. Heikkisellä oli laaja näkemys järjestelmien elinkaaren hallinnasta ja siitä kuinka tuotealustaan tulisi pysyä osoittamaan muutokset, sen sijaan että niitä tuotaisiin jokaiseen projektiin erikseen. Heikkisen kokemukset perustuivat enemmän tuotteiden vaatimusten hallintaan kuin projektien vaatimusten hallintaan. Tämä on hieman poikkeava ajatusmalli siihen miten PETin laitteistotoimitukset toteutetaan tällä hetkellä.

Heikkinen kommentoi vahvasti, sitä kuinka alustaan kerätään vaatimuksia erillisellä prosessilla, mikä sisältää muun muassa tarvittavat katselmointipisteet tuotekehityksen puolella. Vasta sen jälkeen järjestelmän muutokset tuodaan myytävään laitteeseen tai projektin omaisesti toimitettavaan järjestelmään. Heikkisen tuoma näkemys haastattelussa oli

koko tuotteen elinkaaren hallinta. Vaatimukset saattavat tuotemaisessa sovelluksessa kerrääntyä aina tuotekehityksen alkulanseerauksesta, tuotteen ”ramp-down” vaiheen suunnitteluun. (Heikkinen 2016).

Heikkinen oli myös työskennellyt vaatimusten käsittelyjärjestelmän kehittämisen ja sen soveltamisen parissa. Heikkinen piti seuraavia ominaisuuksia järjestelmälle tärkeinä:

Priorisointi – Vaatimukset tulisi pystyä priorisoimaan ja luokittelemaan. Tämä helpottaa vaatimusten hallinnointia ja vaatimusten analysointia.

Tunnisteet – Jokaiselle vaatimukselle tulee pystyä luomaan yksilöllinen tunniste. Sen avulla vaatimukseen pystytään viittaamaan tuotedokumentaatioissa. Yksilöllisen tunnisteiden avulla pystytään myös luomaan helposti listoja, suunnittelun edistymisen seuraamiseksi.

Luokittelu – Vaatimusmäärittelyssä on tärkeää, että vaatimuksia pystytään luokittelemaan erilaisilla perusteilla. Se lisää ison vaatimusmassan hallinnoinnin helpoutta, kun vaatimusten parissa työskentelee paljon ihmisiä.

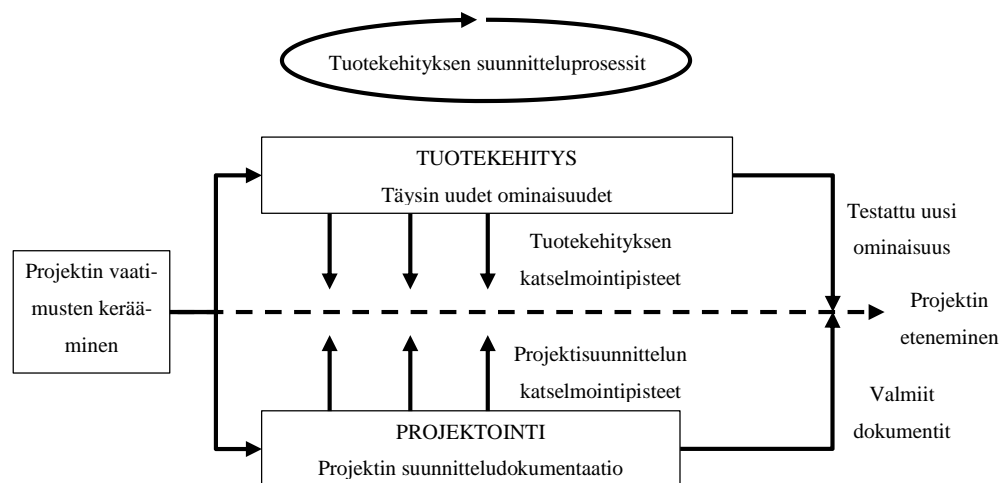
5.2 Päätelmät haastatteluista

Karjalaisen kommenttien perusteella voidaan todeta, että hänen edustamassa suunnitteluosastossa noudatetaan selvästi vesiputous mallista suunnittelun elinkaarta. Kolehmaisesta kokemuksesta voitiin myös todeta hänen edustaman automaatioprojektien toteutuksen seuraavan vesiputoustyyppistä suunnitteluprosessia. Tässä työssä suoritettua kirjallisuustutkimuksen perusteella valittiin jo aikaisemmin PETille toimivaksi suunnitteluprosessiksi vesiputousmallinen elinkaarimalli (kappale 4.4.5). Suoritettujen haastattelujen perusteella voidaan edelleen pitää vesiputousmallista ratkaisua PETin projektitoimituksiin parhaiten soveltuvana.

Haastattelujen perusteella kappaleessa 4.4.5 esitettyyn prosessimalliin on kuitenkin otettava vielä huomioon Heikkisen mainitsemat seikat koko toimitettavan järjestelmän elinkaareen liittyen. Kuvaajaa tulisi päivittää siten, että se ottaisi huomioon projektiin kohdistuvat tuotekehitystarpeet ja mahdollisesti projektin siirtäminen huolto-organisaatiolle.

Tämän työn puitteissa ei enempää pohdittu sitä, millainen prosessi tulee tuotekehitykselle osoitettujen ominaisuuksien katselmointiin luoda.

Lisäämällä vesiputousmalliin tuotekehitykselle erillisen haaran, pystyisivät projektin pääsuunnittelijat ja projektipäällikkö paremmin seuramaan erillisinä osastona toimivaa tuotekehitystä. Projektin suunnittelun kannalta on oleellista, että projektin aloitusvaiheessa tunnistetaan tuotekehitykselle kuuluvat tehtävät. Jos tuotekehityksen tarve huomioidaan liian myöhään, ei tuotekehitykselle jää aikaa toteuttaa omia suunnittelukatselmointejaan ja raportointeja etenemisestä. Tuloksena saattaa olla, että uusi ominaisuus ei sovellu projektiin luotuun kompensointijärjestelmään. Riittävällä tarkkuudella vaatimusten kartoitus- ja analysointivaiheessa saataisiin vähennettyä tällä hetkellä esiintyviä rajapinta ongelmia tuotekehityksen ja projektionnin välillä. Kuvassa 15 on vielä havainnollistettuna miten tuotekehitys huomioitaisiin PETin projektien toteutuksessa.



Kuva 15. Tuotekehityksen liittyminen projektien toteutukseen.

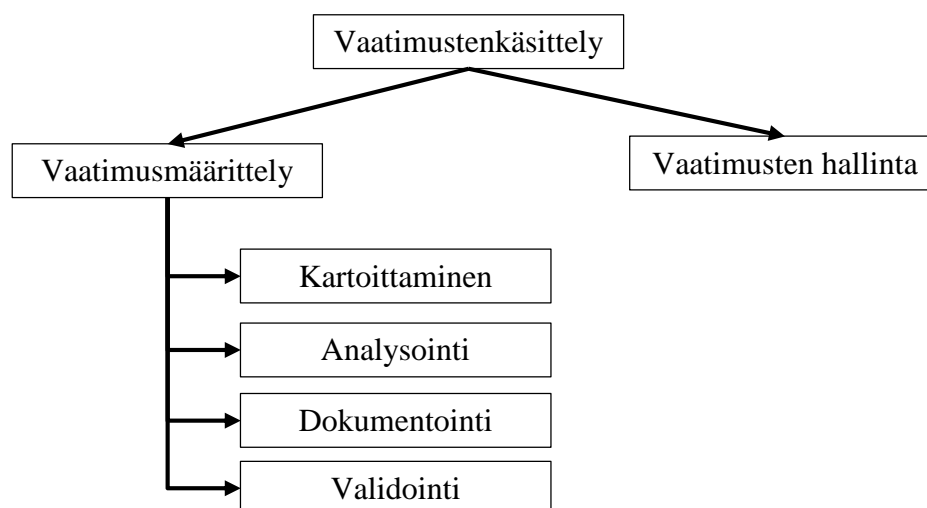
Kuvassa 15 on esitettyä kuinka tuotekehitys tarpeet huomioidaan heti projektin aloitusvaiheessa. Projektin suunnitteluryhmä aloittaa suunnitteludokumentaation tuottamisen ja tuotekehitykselle pidetään aloituspalaveri. Aloituksessa sovitaan aikataulu ja tarkistuspisteet tuotekehityksen uudelle ominaisuudelle. Tuotekehityksen katselmoineissa voi olla mukana myös tekniikkaa tuntevia suunnittelijoita antamassa näkemyksen ja kommentteja onko tuotekehityksen esitys käytännön kannalta toimiva.

Jokaisen haastateltavan kohdalla mainittiin standardi tuotealusta tai peruskonsepti. PETin tämän hetkinen tuotealusta ei sisällä peruskuvausta kompensointijärjestelmä. Yksittäiset komponentit ovat dokumentoituna ja tekijät tietävät oman osuutensa peruskokoonpanon. Haastattelujen perusteella tämän voidaan todeta olevan erityinen ongelma vaatimusten käsittelyn kannalta. Esimerkiksi järjestelmän myyntivaiheessa on vaikea arvioida muutoksien kustannusvaikutusta. Karjalaisen ja Kolehmaisen haastattelujen perusteella voidaan todeta, että vaatimusten kartoittamisessa luodaan vaatimus, kun huomataan että perusominaisuuteen tarvitaan poikkeava ominaisuus. Jos perusominaisuuksia ei ole valmiiksi määriteltynä on vaikea arvioida mitkä asiat oletetaan tiedettävän ja mitkä lopulta pitäisi vaatimusten käsittelyjärjestelmään kirjata. Standardi tuotekuvauksen laatiminen tulee olla tehtynä ennen kuin vaatimusten määrittelyä voidaan täydellä teholla hyödyntää projektin aloituksesta lähtien.

Kolehmaisen ja Karjalaisen kokemuksissa vaatimuksia kirjataan ja seurataan taulukkopohjaisella ratkaisulla. Tämä on varmasti toimiva sellaisessa sovelluksessa, missä vaatimuksia hallinnoi ja seuraa vain muutamat ihmiset. Vaatimusten käsittelyn laajentuessa kymmenille ihmisillä saattaa taulukkopohjainen vaatimusten käsittelyjärjestelmä muodostua jopa pullonkaulaksi. Aikaisemmin tässä työssä tehdyn sidosryhmäanalyysin perusteella (kappale 4.3.1) perusteella voidaan todeta, että suunnittelun vaikutusvallan määrä on lähes kaikilla sidosryhmillä keskitason yläpuolella. Näin ollen voidaan päätellä, että vaatimusten käsittelyjärjestelmään tulee PETin tapauksessa käyttämään useat ihmiset. Tämä johtaa siihen, että taulukkolaskenta pohjainen ratkaisu ei toimi. Yhdistämällä Heikkisen haastattelu ja kirjallisuudesta poimitut tiedot voidaan tehdä päätelmä PETille sopivasta järjestelmästä. Palvelin pohjainen, erillinen vaatimusten käsittelyjärjestelmä on näihin kahteen seikkaan vedoten parhaiten PETin tarkoitusta palveleva.

6 VAATIMUSTEN KÄSITTELYN KOMPONENTIT

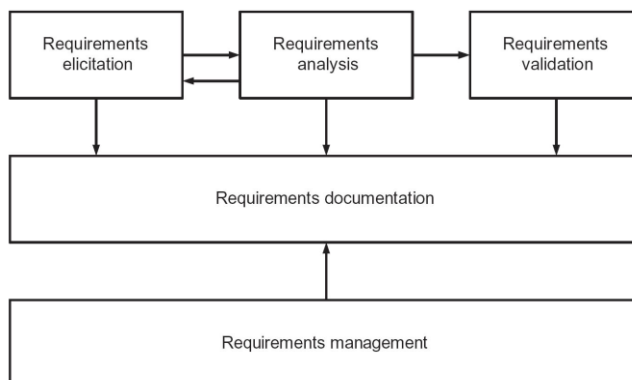
Kuvassa 16 on esitettyä miten Developing and Managing Embedded Systems and Product havainnollistaa vaatimus määrittelyn ja vaatimusten hallinnan eron (Fowler 2014, Developing and managing requirements).



KUVA 16. Vaatimuksien käsittelyn rakenne (Fowler 2014, Developing and managing requirements)

Vaatimusmäärittelyprosessi noudattaa kirjallisuudessa hyvin pitkälle samanlaisia periaatteita, kuin mitä kuvassa 16 on esitettyä. Essentials of software engineering kuvaa, että vaatimusmäärittelyprosessi alkaa vaatimusten kartoittamisella. Kun vaatimukset on kartoitettu, aloitetaan niiden analysointi. Kirjallisuudessa dokumentointi portaaseen yhdistyy monessa tapauksessa prototyyppien luominen (Tsui & Karam 2007, 142). Ohjelmistotuotannon alueella prototyyppien ja simulointimallien tekeminen on huomattavasti sujuvampaa, kuin mitä se on valmistavan teollisuuden toimitusprojekteissa. Kun vielä vaatimusmäärittelyä halutaan soveltaa työn tilaajan tarpeisiin, ei dokumentaatiota voida toteuttaa suoraan siten kuin se on ohjelmistotuotannon kirjallisuudessa monesti kuvattu. Validointi vaiheella tarkoitetaan ohjelmistotekniikassa yleisesti vaatimuksien hyväksyttämistä ja katselmointia, jotta asiakkaan tilaama laitteisto tai ohjelmisto vastaa kaikkia odotuksia. ISO/IEEE 15288 määrittelee, että vaatimuksien määrittelyn tuloksena syntyy systeemille vaatimusmassa. Vaatimuksien ei ole tarkoituksena suoraa määrittellä miten systeemi tulisi valmistaa tai suunnitella. Sen sijaan niiden tarkoituksena on määrittää ne raja-arvot joiden sisällä systeemin suunnittelun tulee tapahtua. (ISO/IEEE 15288:2015 2015, 54)

Kuvassa 17 on esitettyä kuinka Business analysis kirjassa on vaatimusten käsittely rakennettu.



KUVA 17. Vaatimusten käsittelemisen toimintamalli (Paul, Cadle & Yeates 2014, a frame work for requirements engineering)

Kuvan 17 mallin huomataan olevan hieman poikkeava verrattuna Fowlerin esitykseen kuvassa 16. Suurin ero malleissa on, että kuvan 17 mallissa on vaatimusten dokumentointi korostetussa kohdassa. Kaaviossa se on sijoitettuna vaatimusten käsittelyn keskipisteeksi. Vaatimusten hallinta vaikuttaa omalta osaltaan vaatimusdokumentaatioon ja toisaalta vaatimusten määrittely vaikuttaa myös vaatimus dokumentaatioon. Malli ei täten niin hiarkinen kuin Fowlerin esittämä malli. Teoksessa Business analysis kuvailaan jopa, että projekteja on epäonnistunut pelkästään huonon vaatimus dokumentoinnin johdosta (Paul, Cadle & Yeates 2014, Documenting and Managing Requirements).

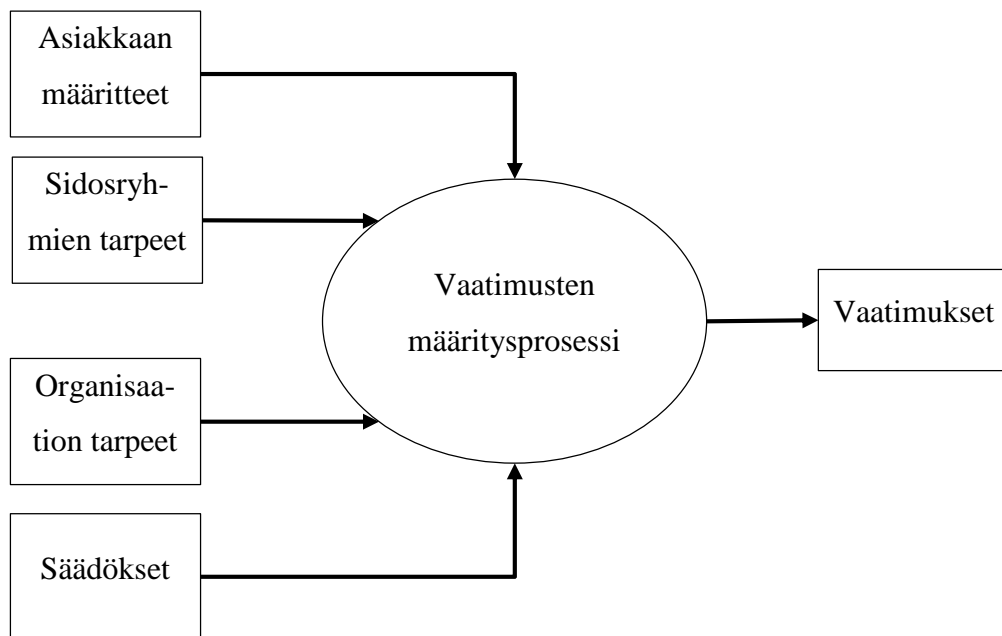
VAATIMUSTEN KÄSITTELYN POHDINTA PETIN KANNALTA

Kummassakin mallissa voidaan todeta olevan tärkeitä tekijöitä vaatimusten käsittelyn ylemmän kuvan muodostamisessa. Kokonaisuutena PETin toimintaan paremmin soveltuvana mallina voidaan pitää kuvassa 16 esitettyä rakennetta vaatimusten käsittelystä. Kuvan 16 mallissa kaikki vaatimus käsittelyyn liittyvät ali-osat tulevat systemaattisesti mukaan. Malli sopii myös paremmin aikaisemmin perusteltuun vesiputous malliseen suunnitteluprosessiin. Siinä toteutuu kaikki kuvassa 16 esitetyt vaatimusten käsittelyn kohdat.

6.1 Vaatimusmäärittely

Kirjassa Requirements Engineering on kuvattuna lohkokaaavioesityksenä vaatimusmäärittelyprosessin koostumus. Karkeimmassa esityksessä prosessiin kuuluu lähtödatan antajia ja prosessin lopputuloksesta hyötyjiä. Prosessiin dataa syöttävinä tekijöinä voidaan pitää seuraavia tekijöitä: asiakkaan määrittelydokumentaatio, projektin sidosryhmien tarpeet, laitteistoa koskevat standardit, säädökset sekä lait ja organisaation toimintastrategian luomat tarpeet. Prosessista saadaan tuloksena järjestelmän toimintaa kuvaavat vaatimukset, järjestelmän määritelmädokumentaatio ja järjestelmän toimintaa kuvaavat lohkokaaviot. (Kotonya & Sommerville 1997, 28)

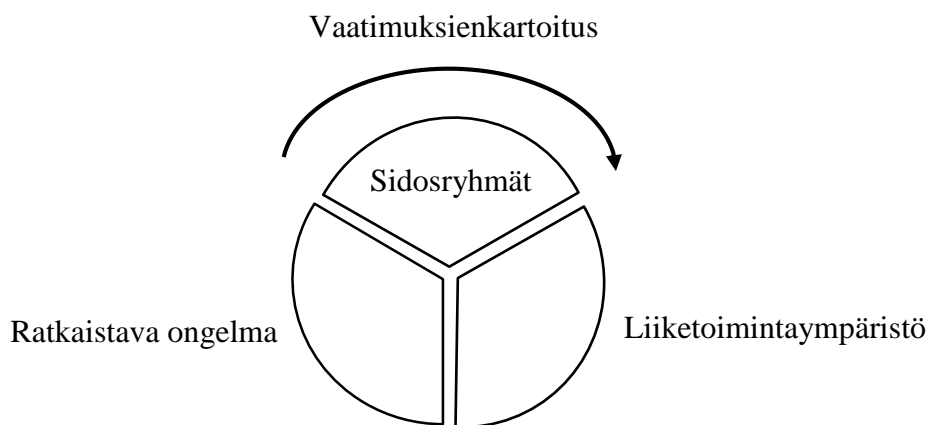
Kuvassa 18 on esitettyä aikaisempia tietoja soveltamalla vaatimusten määrittelyn kuvaus, joka on laadittu PETin loistehon kompensointilaitteistojen suunnittelun kannalta. Kuvassa on esitettyä miten vaatimuksienmäärittelyssä tarvitaan eri lähteistä tietoa, joka voidaan jalostaa vaatimuksien määrittely prosessin avulla monikäyttöisiksi vaatimuksiksi.



KUVA 18. Vaatimusmäärittelyprosessin lähteet ja tulokset PETille sovellettuna (Kotonya & Sommerville 1997, 28. Muokattu)

6.1.1 Vaatimustenkartoittaminen

Vaatimusmäärittelyprosessi aloitetaan vaatimusten kartoittamisella, kartoittamisen ollessa määrittelyprosessin ensimmäinen askel voidaan se nähdä jo prosessinkin näkökulmasta yhtenä vaatimusmäärittelyprosessin oleellisimpana kohtana. Jos vaatimuksia ei saada kattavasti kartoitettua, saattaa jotain järjestelmän tilaajan kannalta oleellista jäädä toteuttamatta. (Kotonya & Sommerville 1997, 55) Kuvassa 19 on esitettyä millaisista ulottuvuuksista vaatimuksien kartoittaminen rakentuu PETin toimintaympäristössä.



KUVA 19. Vaatimuksien kartoittamisen osatekijät (Kotonya & Sommerville 1997, 55. Muokattu)

Sidosryhmien tuomat tarpeet ja rajoitukset vaatimusmäärittely tulee olla tunnistettuna, aikaisemmassa kappaleessa: suunnittelun sidosryhmät, on tarkemmin kuvattuna PETin suunnittelun sidosryhmien merkitys.

Eri ryhmillä saattaa olla toisistaan poikkeavia tarpeita ja näkemyksiä siitä millainen systeemi ollaan toimittamassa. Vaatimuksien muodostaminen palvelemaan pelkästään yhtä sidosryhmää ajaa toteutuksen vinoutuneeseen lopputulokseen.

Liiketoimintaympäristö tuo vaatimusten kartoittamiseen yrityksen toiminnan strategisen näkökulman. Jos yrityksellä on tavoitteena lisätä esimerkiksi palvelujen tuottamista, tulee vaatimuksien kartoittamisessa ottaa huomioon nämä yrityksen strategiset linjaukset. Ratkaistavalla ongelmalla kuvataan sitä korkeimman tason vaatimusta, mitä suunniteltavan laitteiston tulee toteuttaa, toisin sanoen se voi olla projektin tavoite. PETin tapauksessa tämä olisi esimerkiksi kantaverkon kompensointi vaatimusten täyttäminen.

Siirryttäessä vaatimusten kartoittamisen teoreettisesta osuudesta lähemmäksi käytännön osuutta huomattiin, että vaatimusten kartoittamiseen on olemassa useita erilaisia tekniikoita. Kirjallisuutta tutkimalla päästiin PETin tapauksessa siihen tulokseen, että vaatimusten kartoittamisessa voidaan käyttää vapaasti kaikkia tekniikoita tiedonkeräämiseen. Kirjallisuustutkimuksen perusteella voidaan todeta, että kartoitustekniikan valitseminen tarvitsee aina arvioida tapauskohtaisesti.

Jotta päästäisiin mahdollisimman lähelle vaatimusten kartoittamistekniikoiden perusteita, lähdeaineistoksi valittiin kansainvälinen ISO/IEC/IEEE 29148:2011 standardi. Standardissa mainitaan seuraavat tekniikat vaatimusten kartoittamiseen (ISO/IEC/IEEE 29148:2011 2011, 22) :

- Työpajat ja aivoriihitekniikat
- Haastattelut ja kyselyt
- Teknisen dokumentaation katselmointi
- Kilpailevien järjestelmien analysointi
- Simulaatiot, prototyypit, mallinnukset
- Järjestelmän toimintakyvyn mittaaminen kilpailevia järjestelmiä vastaan
- Analysointimenetelmät kuten esimerkiksi SWOT-analyysi

Tutkimuksen ja kokemuksen perusteella voidaan todeta, että PETin projekteissa vaatimusten kartoittamisessa käytännöllisin tekniikka olisi varmasti vapaat keskustelut tilaisuudet. Niiden sisältöä voisi rikastuttaa käyttämällä aivoriihi tekniikkaa.

6.1.2 Vaatimusten analysointi

Vaatimuksien kartoittamisen jälkeen tulee vaatimusmassalle tehdä analysointi, se voidaan tehdä esimerkiksi erillisessä analysointikatselmuksessa. Katselmoinnissa on tarkoituksena tarkistaa yksitellen kaikki vaatimukset ja arvioidaan niiden soveltuvuus toteutettavaksi. Analysoinnin haasteena on, että siihen tulisi pystyä resursoimaan kokeneita suunnittelijoita, joilla on laaja ymmärrys ympäröivästä järjestelmästä. Vaatimuksien analysointivaiheen ollessa hidas prosessi on parhaita osaajia monesti vaikea sitoa siihen riittävän pitkäksi ajaksi (Kotonya & Sommerville 1997, 77).

ISO/IEC/IEEE 29148:2011 standardi määrittelee vaatimuksien analysoinnin seuraavalla tavalla:

The purpose of the Requirement Analysis Process is to transform the stakeholder, requirement-driven view of desired services into a technical view of a required product that could deliver those services.

Tämän tueksi määrittelee Mikkola & Märijärvi kirjassaan Ohjelmistotuotanto vaatimuksien analysoinnin tavoitteeksi seuraavat tekijät (Mikkola & Märijärvi 2004 ,95):

- Selvittää kunkin vaatimuksen tarve. Vaatimuksen tarpeen selvittämisen nyrkkisääntönä voidaan pitää vastauksen etsimistä kysymykseen ”miksi?”. Tarpeen selvittämisen tuloksena tulisi tietää perimmäisimmät syyt asiakkaan tarpeista.
- Arvioida kunkin vaatimuksen tärkeys (priorisointi). Priorisoinnin merkityksenä on valikoida koko vaatimusmassasta missä järjestyksessä vaatimuksia tulisi ruveta työstämään. Esimerkiksi erittäin pitkään kestävät toteutukset tulisi merkitä korkealla prioriteetillä.
- Sovittaa yhteen ristiriitaiset vaatimukset. Vaatimuksien analysoinnissa törmätään varmasti ristiriitaisiin vaatimuksiin. Ristiriidat saattavat nousta esiin niin systeemin suunnittelijoita kuin asiakkaankin osalta. Vaatimuksien sovitamisen voidaankin siis todeta muodostavan selkärangan koko suunniteltavalle systeemille.

Vaatimusten analysoinnin tukena voidaan pitää vaatimuksien kategorioimista ja priorisointia. Vaatimuksien kategorioiminen omiin kokonaisuuksiinsa helpottaa vaatimusten käsittelyä ja niiden merkityksellisyyden tunnistamista. Vaatimusten kategorioimisessa tulee valita järjestelmän valmistajan kannalta parhaiten toimiva kategoriointi tapa. Vaatimuksia voidaan ryhmitellä esimerkiksi suunnittelun vaiheiden mukaisesti, mallipohjaisen suunnittelun mallien mukaan tai sidosryhmien mukaan. (Tsui & Karam 2007, 151-160).

Analysointi prosessiin liittyy myös vaatimusten prioriteettien arvioiminen. Priorisointiin voidaan kehittää yrityskohtaisia laskentakaavoja tai kehittyneitä vertailu matriiseja (Tsui & Karam 2007, 151-160). Vaatimuksia voidaan myös priorisoida pelkästään arvioimalla niitä.

Kirjassa *Mastering the Requirement Process* neuvotaan arvioimaan seuraavia asioita vaatimusten priorisoinnissa (Robertson & Robertson 2010, 333):

- Toteutuksen kustannukset
- Vaatimuksen tarve asiakkaan näkökulmasta katsottuna
- Paljonka aikaa tarvitaan toteutukseen suhteessa käytettävissä olevaan aikaan
- Kuinka helppo vaatimus on teknisesti toteuttaa
- Kuinka tärkeä vaatimus on liiketoiminnan strategian toteutumisen kannalta
- Hyödyllisyys liiketoiminnalle
- Sääteleeekö säädökset tai lait vaatimusta

Analysoitaessa vaatimuksia saatetaan tulla ristiriitatilanteisiin, jolloin kaksi vaatimusta kumoavat toisensa. Muita ongelmia, mihin vaatimuksien analysoinnissa saatetaan törmätä ovat: resurssien puute, ajan puute tai tekniset rajoitukset. Törmätessä edellä esitettyihin ongelmiin, tulisi Requirements Engineering kirjan mukaan käydä niin sanottu vaatimusneuvottelu. Neuvottelun tavoitteena on käydä suunnittelun sidosryhmien kanssa keskustelua siitä millaisilla vaatimuksilla pystytään pääsemään kaikkia tyydyttävään ratkaisuun. Kirjan mukaan neuvotteluista saadaan paras tulos, kun sidosryhmien edustajat tapaavat toisensa. (Kotonya & Sommerville 1997, 81)

VAATIMUSTEN ANALYSOINTI TUTKIMUKSEN LIITTÄMINEN KEHITYSTEHTÄVÄÄN

Vaatimusten analysoimisessa tarvitsee huomioida kirjallisuustutkimuksessa löydettyjä tekijöitä. Vaikka vaatimusten analysoinnista löydettiin paljon materiaalia, on analysointi aina riippuvainen yrityksen teknologioista ja analysoijien ammattitaidoista. Näin ollen suoria ohjeita ei kirjallisuudesta voida kopioida. Kirjallisuustutkimuksen perusteella voidaan ehdottaa taulukon 1 mukaisia suuntaviivoja noudatettavaksi PETin vaatimusten analysointi vaiheessa.

TAULUKKO 1. Vaatimusten analysoinnin näkökulmat PETille sovitettuna

Tarkistuskohta	Selite
Esisuunnittelun tarve	Onko vaatimus suoraan toteutettavissa vai tarvitaanko tueksi tuotekehityksen panosta? Onko vaatimuksen toteuttamiseen tarvittava lähdemateriaali saatavilla?
Vaatimuksen laajuus	Onko vaatimus riittävän yksinkertainen vai tulisiko se jakaa pienempiin osiin seurannan ja testaamisen helpottamiseksi
Tarpeettomat vaatimukset	Onko vaatimus todella tarpeellinen projektin tavoitteiden täyttymiselle? Suunnittelun minimi periaatetta tulee pystyä tähdentämään myös vaatimusten hyväksymisessä.
Vaatimuksen toteuttaminen	Onko vaatimuksen toteuttamiseen tarvittavat laitteistot ja järjestelmät PETille saatavissa. Millaisiksi arvioidaan uusien järjestelmien hankintakustannukset?
Liiketoimintastrategia	Toteuttaako vaatimus niitä strategisia linjauksia mitä PET-yksikölle on asetettu?
Vaatimuksen yksiselitteisyys	Ymmärtävätkö kaikki vaatimuksen parissa työskentelevät sidosryhmät vaatimuksen samalla tavalla?
Realismi	Onko vaatimus realistisesti ajateltuna toteutettavissa projektibudjetin puitteissa? Jos havaitaan, että vaatimus on tarpeellinen, mutta ei ole realistisesti toteutettavissa tulee projektin johdon aloittaa tarvittavat toimenpiteet budjetin tai aikataulun korjaamiseksi.
Vaatimuksen testattavuus	Onko toteutus vaatimuksen toteutuminen testattavissa (lisääminen testisuunnitelmaan)? Kuinka vaatimuksen toteutuminen validoidaan?

6.1.3 Vaatimusten dokumentointi

Requirements Engineerign Fundamentals pitää vaatimusten dokumentointia erittäin tärkeänä osana koko suunnittelu prosessia. Kirjan mukaan dokumentaation tarvitsee olla systemaattista ja se tulisi ulottaa myös vaatimusten muutoksien hallintaan. Dokumentoinnista ei ole säädetty tiukkoja rajoja, mutta sen tulee pystyä palvelemaan kaikkia sidosryhmiä. (Pohl & Rupp 2015, Document Design) Haikala & Mikkonen (2011) kuvaa teoksessaan, että viime vuosina ovat yleistyneet verkkopalvelut, joihin vaatimukset voidaan dokumentoida ja luoda niille vaatimusmäärittelyprosesseja noudattavat työprosessit.

Vaatimuksien dokumentointiin on käytössä paljon erilaisia tekniikoita. Requirements Engineerign Fundamentals listaa normaalin kirjoitetun tekstin olevan yksi tapa dokumentoida vaatimuksia. Sen vahvimpana puolena on, että sen tekeminen ei vaadi sidosryhmiltä erillisiä koulutuksia. Lisäksi se on sovellettavissa kaikkiin mahdollisiin vaatimuksiin. Kirjoitetun tekstin haittapuolena on, että siitä saattaa tulla epäselvää. Kirjoitettu teksti sisältää myös monesti mahdollisuuden tulkintaan, joten lopputulema saattaa kirjoitetun vaatimuksen perusteella olla alkuperäisestä suunnitelmasta poikkeava. (Pohl & Rupp 2015, Requirements Documentation using Natural Language)

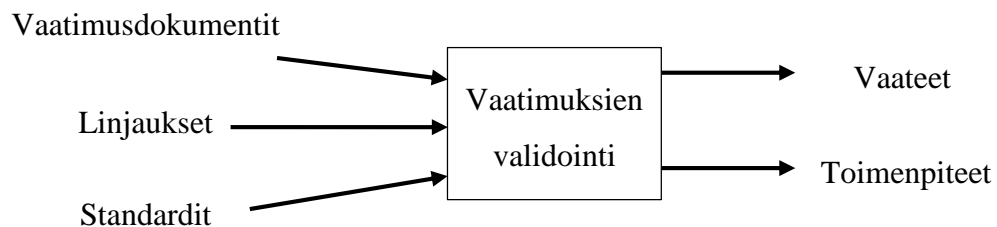
Vapaan kirjoituksen vastakohtana ovat erilaiset mallinnuskielet. Niiden haittana, on että niitä ei pysty muodostamaan ilman koulutusta ja niiden lukeminen vaati ensiksi mallinnuskielen ymmärtämisen. Mallinnuskielet poistavat vaatimuksista tulkinnanvaraisuutta ja nopeuttavat vaatimukset lukemista. (Pohl & Rupp 2015, Documentation using Conceptual Models). Raussi & Virtanen kirjoittavat artikkelissaan mallinnuskielen ja vapaan tekstipohjaisen vaatimusdokumentaation eroista. Artikkelin pohjalta voidaan todeta, että sopivan dokumentointitapa tulisi pystyä määrittelemään jokaisessa vaatimusmäärittelyä hyödyntävässä yrityksessä tapauskohtaisesti (Raussi & Virtanen 2009, 16).

Kun vaatimuksia kirjoitetaan vapaasti, voidaan vaatimus dokumentaation sisällön perusoheistuksena pitää ISO/IEC/IEEE 29148:2011. Peilattaessa vaatimusten dokumentointia PETillä voidaan standardin esittämiä vaatimuksia kirjallisesta vaatimusten dokumentoinnista pitää toimivana ohjenuorana. Liitteessä 5 on esitettyinä standardin listaamat seikat vapaasti määritellylle vaatimukselle.

6.1.4 Vaatimusten validointi

Vaatimusmäärittelyosuuden viimeisenä portaana on vaatimusten validointi. Validoinnissa perusajatuksena voidaan pitää laadittujen vaatimusten hyväksymistä. Validoinnin lopputuloksena tulisi pystyä aloittamaan vaatimusten perusteella systeemin toteutus. Validointivaiheen kantavana kysymyksenä voidaan pitää: ”Onko kaikki vaatimukset kirjattuna?”. (Kotonya & Sommerville 1997, 87)

Vaatimuksien validoinnin haasteena on, että siihen ei ole saatavilla mitään yksittäistä dokumenttia, mitä vastaan vaatimukset hyväksyttäisiin. Validointia ei siis voida perustaa muihin seikkoihin kuin kokemukseen ja tietämykseen suunniteltavasta systeemistä. Vaatimuksien validointi prosessiin tulee sisääntulona laaditut vaatimusdokumentit, organisaatiotason strategialinjaukset ja säädökset sekä standardit. Prosessista ulostulona saadaan listaus järjestelmän vaatimuksesta ja sovitusta toimenpiteistä, joilla suunnitellua voidaan aloittaa. Kuvassa 20 on esitettyä vaatimuksien validointiprosessi. (Kotonya & Sommerville 1997, 89)

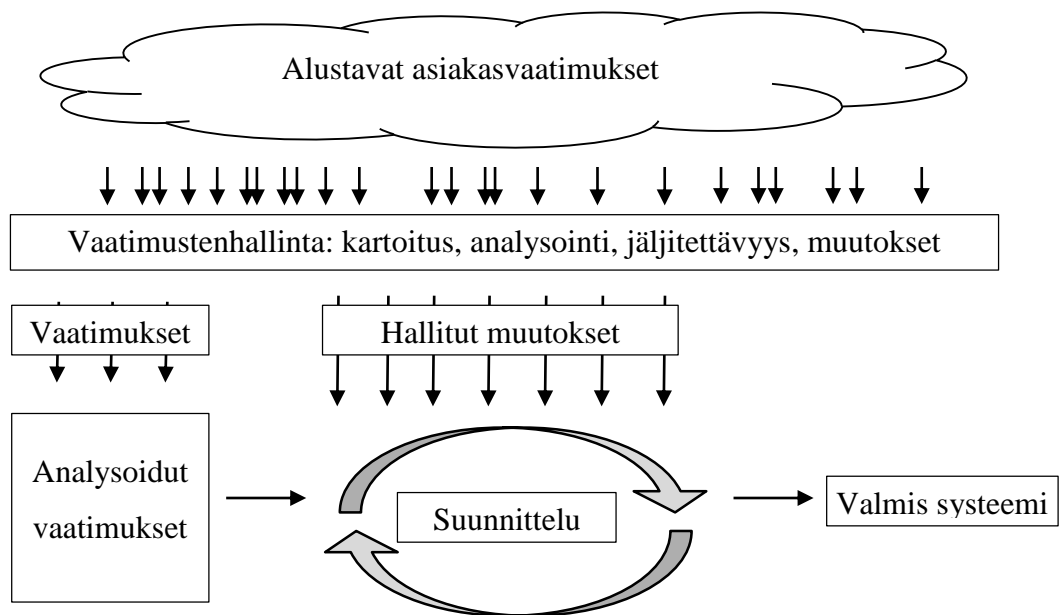


KUVA 20. Vaatimuksien validointiprosessi (Kotonya & Sommerville 1997, 89. Muokattu)

Validointiprosessi saattaa olla ajallisesti erittäin pitkä kestoinen. Tuotettavan systeemin ollessa monimutkainen joudutaan validointi prosessiin ottamaan mukaan useita eri suunnittelun sidosryhmiä. Requirements Engineering teoksessa varoitetaan, siitä että suunnitteluvaiheeseen ei tulisi edetä ennen kuin validointi prosessi on kokonaan käyty lävitse. (Kotonya & Sommerville 1997, 90)

6.2 Vaatimustenhallinta

Requirements Engineerin kirja määrittelee, että vaatimustenhallinta voidaan käsittää prosessina, missä hallitaan vaatimuksien muutosta sitä mukaan, kun asiakkaalla ja systeemin toimittajalla ymmärrys luotavasta systeemistä kehittyy. (Kotonya & Sommerville 1997, 113) Requirements Engineering Fundamentals puolestaan kuvaa, että vaatimusten hallinta käsittää vaatimusten tarkentamisen, priorisoinnin, seuraamisen ja vaatimusten versioinnin. Sen mukaan vaatimusten hallintaan tulee liittää yksittäisien vaatimusten hallinta ja niiden lisäksi kaikkien vaatimuskäytäntöjen hallinta. (Pohl & Rupp 2015, Requirements Management) Haikala & Märijärvi (2004) selittää kirjassaan, että vaatimusten hallinta on tukitoimi vaatimusten määrittelylle. Ilman vaatimusten kunnollista hallintaa, ei päästä lopputulokseen mikä tyydyttäisi systeemin tilaajaa. Kuvassa 21 on esitettyä Haikalan & Märijärven näkemys vaatimusten hallinnasta suunnitteluprosessissa.



KUVA 21. Vaatimustenhallinnan merkitys vaatimusten käsittelyssä (Haikala & Märijärvi 2004, 93. Muokattu)

Haikala & Märijärvi (2004) on kuvannut kuinka vaatimusten hallinnan merkitys on toimia asiakkaan ja toimittajan välillä. Tehtäviin kuuluu koordinoita, että kaikki muutokset joita systeemin kehittyessä huomataan, tulee kirjattua hallintajärjestelmään. Kuvassa 21 on esitettyä suunnitteluprosessi iteroivana prosessina, koska vaatimusten validoinnin

jälkeen joudutaan lähes jokaisessa projektissa täydentämään suunnittelua uusilla vaatimuksilla. Pohdittaessa vaatimustenhallintaa PETin toiminnan kannalta voidaan tulla sellaiseen johtopäätökseen, että vaatimustenhallinnan tulee olla projektin vetäjien vastuulla. Heidän tehtävien rajaukseen tulisi kuulua toimiminen linkkinä suunnittelun osa-alueiden ja asiakkaan välillä. Linkkinä oleminen tarkoittaa tällöin muutoksien kommunikointia prosessissa alaspäin suunnittelulle. Lisäksi se vaatii muutosviestien viemistä takaisin asiakkaalle päin. (Haikala & Märijärvi 2004, 99)

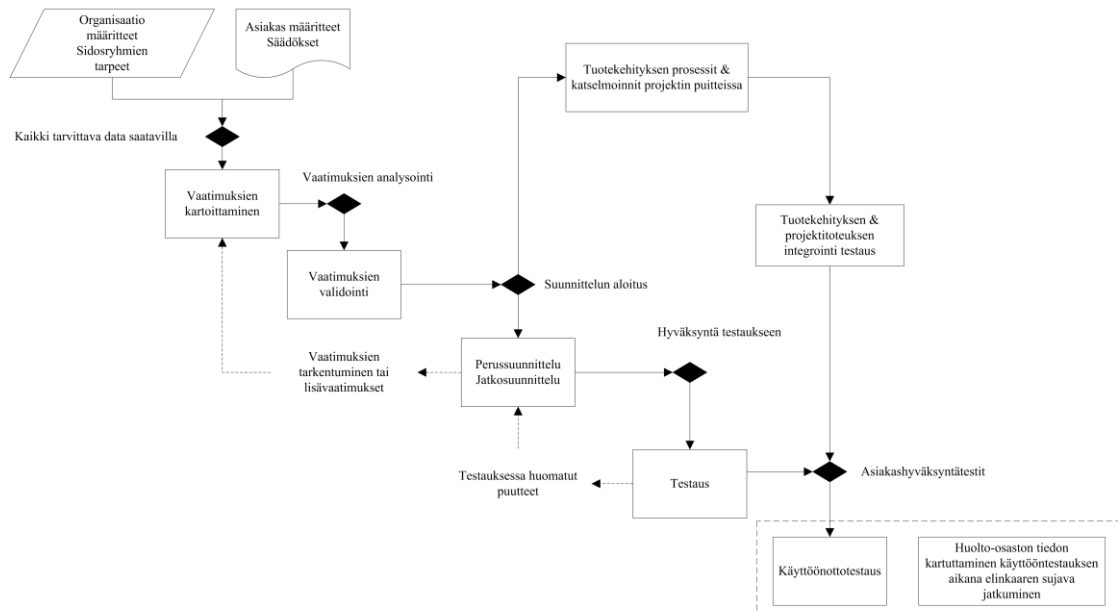
Pelkkä kommunikaatiolinkkinä toimiminen ei vielä ole vaatimusten hallinnan kokonaisuus. Vaatimusten keräämisen ja niiden hyväksymisen koordinointi kuuluu myös vaatimusten hallinnan piiriin. PETin toimintaan peilaten projektin vetäjien ei tarvitse itse määritellä tai hyväksyä vaatimuksia. Sen sijaan heidän tulee pystyä ylläpitämään projektissa järjestelmää, minkä perusteella kyseiset tehtävät projektin osalta tulee täytettyä. Toisin sanoen vaatimustenhallinnan tehtävänä on ylläpitää vaatimustietokantaa sekä ylläpitää siellä pidettävien vaatimuksien tilaa (Kotonya & Sommerville 1997 115).

Haikala & Märijärvi (2004) määrittelevät muutoksien olevan, joko eteenpäin- tai taaksepäin jäljitettäviä. Eteenpäin jäljitys muutoksen kohdalla tarkoittaa, että muutoksesta johtuvia uusia vaatimuksia saatetaan joutua luomaan. Taaksepäin jäljitettävyys puolestaan kuvaa, että muutoksen johdosta jotain jo suunniteltua joudutaan muokkaamaan tai jättämään kokonaan pois systeemistä. (Haikala & Märijärvi 2004, 98-99)

Muutoksien tuovat projektin johtamiselle lisähaasteen työkuormien arvioimiseen. Pienien muutoksien tekeminen ei kuormita suunnitteluprosessia normaalia enempää, mutta perusasioiden muuttaminen systeemin suunnittelun ollessa jo hyvässä vauhdissa saa aikaan helposti projektin työmäärän lisääntymisen. Toimivasta vaatimuksien hallinnasta ei tarvitse tehdä turhan monimutkaista. Yksikertaisuudessaan vaatimuksia voidaan rekisteröidä pelkkään taulukkolaskenta pohjaiseen työkaluun. Tällöin esimerkiksi projektipäällikkö ilmoittaa muutoksista suunnitteluryhmien vetäjille, jotka kirjaavat muutokset taulukkoon. Taulukon avulla voidaan suorittaa esimerkiksi laadittujen suunnitteludokumenttien tarkastusta. Projektin aloituksessa voidaan kerätä niin sanottu esitietokaavake, josta saadaan suoraan poimittua osa vaatimuksista. Vaatimuksena esitietolomakkeen laatimiselle on, että tuotteista on luotuna yksityiskohtaiset standardikuvaukset. (Kolehmainen 2016)

7 VAATIMUSKÄSITTELYN TOTEUTUS

Vaatimuksienmäärittelyprosessista luotiin kuvan 22 mukainen ehdotus, se perustuu kehitystehtävän kappaleissa 4.4.5, 5.2 ja 6.2 tutkittuihin tekijöihin toimivasta vaatimuksien käsittelymallista. Kuva 22 on esitettyä isommassa koossa myös liitteessä 3.



KUVA 22. Ehdotus vaatimuksienkäsittely prosessista

Vaatimusten kerääminen tulisi aloittaa heti projektin aloituskokouksen jälkeen. Teknisessä aloituskokouksessa tulisi ensiksi tutustua projektinjohdon kanssa pääpiirteittäin millainen projekti on kyseessä. Tämä sisältää muun muassa suunnittelua rajoittavien ympäristötekijöiden esittelyn. Tilaisuudessa voidaan myös alustavasti keskustella merkittävimmistä teknisistä rajoituksista suunnittelulle. Projektin suunnittelukeskeisten asioiden lisäksi aloituskokouksessa tulee sopia miten vaatimusmäärittely ja vaatimusten hallinta tullaan projektissa toteuttamaan.

Vaatimusten kartoittamiseen tulisi aina sopia aikataulu tavoitteellisuuden varmistamiseksi. Vaatimusten kartoittamisessa voidaan käyttää vapaasti sellaisia menetelmiä, jotka koetaan tilanteeseen parhaiten sopivaksi. Haastatteluiden ja kirjallisuudesta luotujen mielikuvien perusteella aivoriihi tyyliset kokoukset ovat hyvä lähtökohta loistehon kompensointijärjestelmien vaatimusten kartoittamiselle. Vaatimusten kartoittaminen tulisi päättää projektinjohdon kutsumaan vaatimusten validointi kokoukseen. Kokouksessa analysoidaan kartoitetut vaatimukset ja hyväksytään ne prosessin seuraavaan vaiheeseen.

Kaikkien suunnittelun sidosryhmien ollessa mukana vaatimusten hyväksynnässä voidaan vielä tarkistella ovatko vaatimusten väliset viittaukset suunnitteluosa-alueiden välillä oikein.

Kuvan 22 validointivaiheen jälkeen siirrytään laitteiston suunnitteluvaiheisiin. PETin toimintamallissa systeemin suunnittelu on monessa tapauksessa jaettuna perussuunnitteluun ja yksityiskohtaiseen suunnitteluun. Kompensointi systeemin suunnittelussa ennen suunnitteludokumentointia laatimista luodaan systeemistä määrittelydokumentit. Vaatimusten kartoituksen ei ole tarkoituksena PETin sovelluksessa toimia määrittelydokumenttina, jota voitaisiin esimerkiksi hyväksyttää asiakkaalla. Sen sijaan vaatimusmäärittelyn tarkoituksena on selkiyttää kaikille suunnittelun osa-alueille millaista laitteistoa ollaan suunnittelemassa.

Kuvassa 22 on esitettyä myös suunnitteluiteroitusta kuvaavat takaisinkytkentänuolet, kyseisessä vaiheessa toteutuu aikaisemmin selvitetty vaatimusten jatkuva hallinta. Suunnittelun lomassa todetaan lähes aina uusia vaatimuksia, kuten kappaleessa 6.2 kirjoitettiin. Vaatimusten hallinta voidaan osoittaa projektin johdon tehtäväksi, kuten Kolehmainen ja Karjalainen haastatteluissa kuvasi (Kolehmainen 2016 ja Karjalainen 2016). Projektin johdon tehtävään vaatimusten hallinnassa kuuluu jatkuva suunnittelun muutoksien koordinaatio.

Suunnittelun eri osa-alueet tulee pitää tietoisena muutoksista, joita tapahtuu muissa sidosryhmissä. Projektin vaatimusten hallinta kannattaa rakentaa siten, että projektin vetovastuussa olevat koordinoivat muutoksien kirjaamisen ja muutostiedon antamisen muille suunnittelun osa-alueille. Muutoksien toteutuksen kannalta on hyvä, että suunnittelijoille järjestetään tilaisuus keskustella muutoksen vaikutuksista. Oleellista on kuitenkin muistaa kirjata muutokset vaatimusten hallinta järjestelmään.

7.1 Vaatimusten dokumentointi Jirassa

PETillä on käyttöönotettavissa Atlasian Jira palvelin pohjainen ohjelmisto vaatimustenmäärittelyyn. Kappaleessa 5.2 purettujen haastattelujen perusteella voidaan todeta Jira sisältää niitä ominaisuuksia, mitä PETille soveltuvassa järjestelmässä tulisivatkin olla. Jira

on hyvin monipuolisesti muokattavissa oleva työkalu, mikä mahdollistaa useiden käyttäjien samanaikaisen yhteistyön projektin vaatimusten parissa. Ohjelmiston vahvuutena on, että vaatimusten dokumentoinnista saadaan läpinäkyvää ja visuaalisesti helppoa. Vaatimusten parissa työskentelevien ei tarvitse Jiraa käytettäessä avalla raskaita taulukoita tai muistaa lähettää sähköposti-ilmoituksia vaatimusten muutoksista. Jiran hyvänä puoleena on myös sen tarjoama mahdollisuus yhteistyöalustaan vaatimusten työstämisessä. Jiran tarjoamat yhteistyö mahdollisuudet ovat esimerkiksi vaatimusten kommentointi mahdollisuus, siten että kommentit ovat kaikkien nähtävillä. Suunnittelijoita kiinnostavia vaatimuksia voi lisätä omalle vahtilistalle, näin käyttäjät saavat muutoksista automaattisesti muistutuksen. (Jira User guide 6.4 2016)

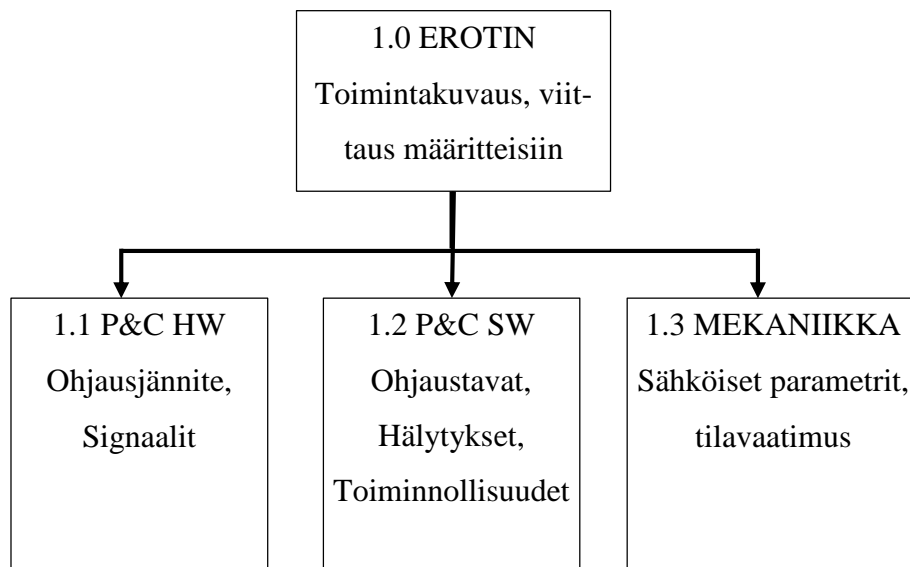
Jiran vahvuus konsernitason työkaluna on sen vapaa muokattavuus. Vapaasti muokattava järjestelmä mahdollistaa tarpeellisten prosessien upottamisen ohjelmaan, esimerkiksi tässä kehitystehtävässä tutkittuja PETille oleellisia vaatimusten käsittelyn kohtia voidaan toteuttaa Jirassa. Toisaalta vapaus luo haasteen prosessien suunnittelulle. Jos mikään seikka ei ole rajoitettu tarvitsee prosesseja suunnittelevien pystyä tarkkaan määrittelemään mitä ohjelmistolla halutaan todellisuudessa saavuttaa. GEellä on Englannissa keskitettynä Jira tukiryhmä. Kehitystehtävän puitteissa käytiin useita puhelinkeskusteluja, jotta saataisiin näkemys millaisia muutoksia PETin Jira näkymään tarvittaisiin tämän kehitystehtävän tuloksen puitteissa.

Jirassa vaatimuksien työkiertoa kutsutaan work flowksi. Työkierron säätäminen PETin tarpeisiin oli Jiraan dokumentoitavien vaatimusten näkyvimpiä muutoksia Jira ohjelmissiin. Sen lisäksi Jiraan dokumentoitavaan vaatimukseen asetettiin tiettyjä attribuutteja pakolliseksi. Näillä muutoksilla Jirasta saadaan vaatimusmäärittelyprosessin toteuttamiseen toimiva alusta ja saadaan toteutettua tässä työssä tutkittuja vaatimusmäärittelyn komponentteja. Samanaikaisesti Jira yritettiin pitää mahdollisimman yksinkertaisena. Yksinkertaisuuden on todettu madaltavan uuden ohjelmiston käyttöönottamisen kynnyksiä. (Puhelin keskustelut Mackowiak & Cooper. 2016).

Teoksessa *Managing Software Requirements* suositellaan, että monimutkaisissa vaatimuksissa voidaan päävaatimus jakaa niin sanottuihin alivaatimuksiin. Alivaatimuksien tarkoituksena on lisätä vaatimuksen työkierron joustavuutta ja helpottaa vaatimuksen käytäntöön soveltamista. (Leffingwell & Widrig 1999, 199-200). Loistehon kompensointi laitteiston osalta vaatimuksen jakaminen ylätasoon vaatimukseen ja sitä seuraaviin

alatason vaatimukseen helpottaisi suunnitteluryhmien yhteistyötä. Esimerkkinä jaottelusta voidaan korostaa ulkokomponentin erottamiseen tarkoitettua kenttälaitteen (erotin) vaatimusten koostamista. Siinä päätason vaatimuksena on kenttälaitte rajoittavine vaatimuksineen ja sille alivaatimuksiksi on määritelty erilaiset suunnitteluryhmä kohtaiset vaatimukset.

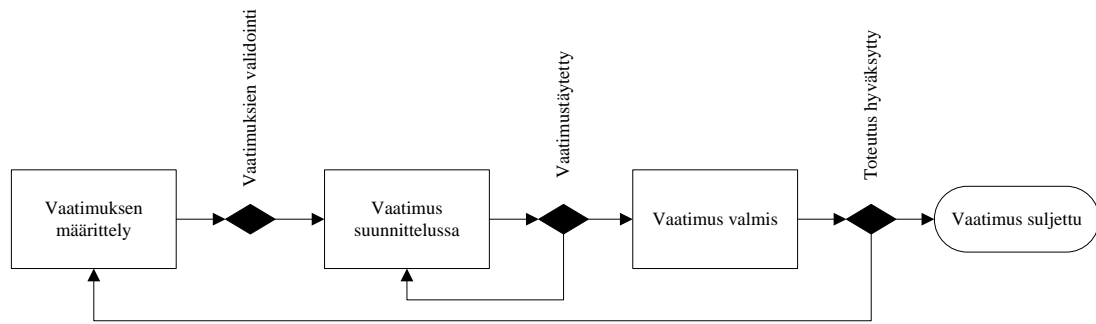
Kuvassa 23 on havainnollistettu kuinka Managing Software Requirements teoksen mukaisesti voidaan esimerkin vaatimukselle luoda alivaatimukset.



KUVA 23. Vaatimusten hiarkinen toteutus

7.1.1 Vaatimusten työkierto Jirassa

Työkierrolla Jirassa määritellään millaisia portaita yhdelle dokumentoidulle vaatimukselle asetetaan. PETin suunnitteluprosessin näkökulmasta olisi järkevää määritellä eri kategorioihin kuuluville vaatimuksille erilaisia työkiertoja, niillä saataisiin vaatimukset paremmin palvelemaan erilaisia suunnitteluprosesseja. Lisäksi erilaiset työkierrat parantaisivat vaatimusten hallintaa Jira-järjestelmässä. Kuvassa 24 on esitetty millainen vaatimuskierro voisi toimia PETin suunnitteluprosessin työkierron lähtökohtana. Työkierto määriteltiin haastatteluista saatujen kommenttien perusteella sekä PETin nykyisen suunnittelun elinkaaren kuvaamisella. Työkierron kuvaamisessa hyödynnettiin myös Jiran tukitimiltä saatuja näkemyksiä Jiran oikeanlaisesta käyttämisestä.



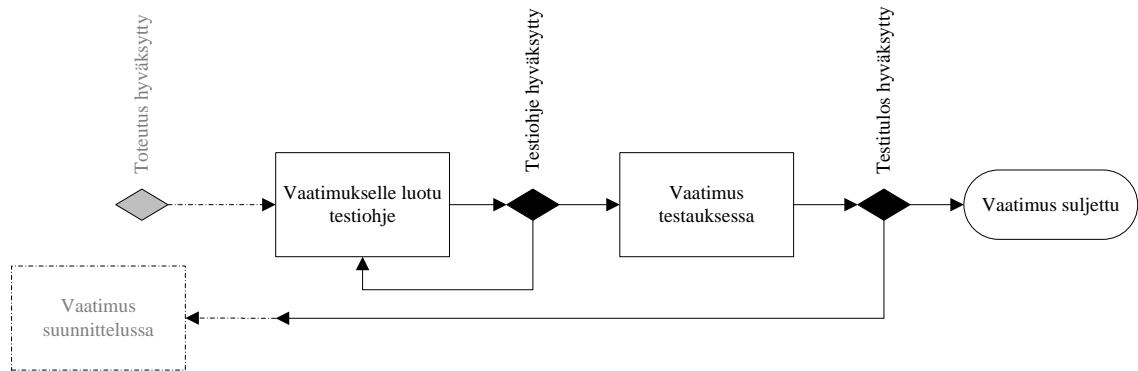
Kuva 24. PETin Jira vaatimuksen yksinkertaisen mallinen työkierto

Kuvan 24 työkierto aloitetaan vaatimusten kartoittamisen ja analysoinnin jälkeen, vaatimusmäärittelyn tulisi siis olla valmis ennen Jira työkierron alkamista. Validointi kokouksen jälkeen vaatimusdokumentti hyväksytään Jira-järjestelmässä suunnitteluvaiheeseen. Kun vaatimusmäärittelyn mukaiset suunnitteludokumentit on laadittu, vaatimusdokumentti jää Jirassa odottamaan hyväksyntää. Hyväksyntä pisteen tarkoituksena on toimia vaatimusten hallinnan apuvaiheena. Jira-järjestelmän avulla projektinjohto pystyy seuraamaan mitkä suunnittelun vaiheet ovat vielä toteuttamatta.

Periaatteellisella tasolla kaikki yhtä kokonaisuutta koskevat vaatimukset tulisi olla hyväksytyinä ennen kuin kokonaisuudelle voidaan antaa valmistuslupa. Kuvasta 24 voidaan nähdä, että jokaisen vaiheen jälkeen voidaan vaatimus joko hyväksyä seuraavaan vaiheeseen tai estää vaatimuksen eteneminen. Hylättäessä vaatimus, siirtyy se kierrossa takaisin päin, odottamaan uutta hyväksyntä kierrosta.

Työkiertoa voidaan vielä syventää lisäämällä siihen uusia kohtia. Monessa tapauksessa suunnitteluprosessiin liittyy myös laitteiston valmistus ja testaus. Testaamisvaiheelle voidaan luoda vaatimusmäärittelyssä erikseen testaamista koskevia erityisvaatimuksia. Jos vaatimukset koskevat systeemin toimintoja, voidaan vaatimusten työkiertoa pidentää suunnitteluvaiheesta. Vaatimuskierroksen loppuun voidaan tällöin lisätä portaita seuraamaan testausvaihetta.

Testausta seuraavilla lisä portaita saadaan loistehon kompensointilaitteistojen vaatimustenhallintaan lisää ulottuvuutta. Kuvassa 25 on esitettyä kuvalla 24 lisäaskeleet, jotka laajentavat toiminnallisen vaatimuksen työkierron testaustuloksien hyväksymiseen.



KUVA 25. Jira työkiertoon lisätyt testauksen verifiointi askeleet

7.1.2 Vaatusdokumentti Jirassa

Jirassa dokumentoitavalle vaatimukselle voidaan asettaa vapaasti attribuutteja, joita vaatimuksen kirjoittajan tulee vähintään määritellä ennen kuin vaatimuksen voi tallentaa tietokantaan. Liitteessä kaksi on esitettyä havainnekuva siitä millainen on Jirassa vaatimuksien dokumentoinnissa täytettävä lomake. Alla olevassa kuvassa 26 on esitelty miten yksittäinen vaatimus, selain pohjaisessa Jira-järjestelmässä, esittäytyy katsojalle. Kuvaan on merkitty numeroin tutkimuksen perusteella PETin sovelluksessa oleelliset kentät.

2014-75-Galoper USVC / PET75GALL-10 5

PLC I/O list 4

Edit Comment Assign More Implement Resolve Resolve & Close

Details

Type: RequirementSW
 Priority: Medium
 Component/s: Control SW
 Labels: None
 Requirement Category: Software
 Verification Method: Test
 Requirement Source: Suokas Ville

Status: 3 OPEN (View Workflow)
 Resolution: Unresolved

Description

In case Ethercat I/O assembly exceeds maximum allowed number of terminals in one power supply, extra power supply is needed. This will require sifting of I/O points in I/O-list 2

KUVA 26. Jiraan luotu vaatimus (Jira 6.4 ohjelmisto 2016)

PETille muodostetun vaatimuskirjeen pakolliset kentät määriteltiin seuraavin perustein:

(1) Vaatimukselle asetettavat attribuutit.

Vaatimukselle asetettavien attribuuttien tehtävänä on toteuttaa hyvälle vaatimukselle esitetyt tavoitteet riittävästä kategorioimisesta. Attribuuteissa määritellään millaista työkiertoa vaatimuksen tulisi noudattaa ja millainen prioriteetti vaatimukselle on asetettu. Attribuutteina syötetään myös mihin suunnittelun osa-alueisiin vaatimus vaikuttaa (component/s) tai kuka on niin sanottu vaatimuksen omistaja. Oleellinen tieto attribuuteissa on myös mistä vaatimus on lähtöisin, näin vaatimuksen lukija pystyy selvittämään epäselvyytilanteessa vaatimuksen lähtökohdat. Lisäksi vaatimusta voi vielä halutessaan kategorisoida kentillä: labels, requirement category ja verification method, mutta niitä ei kehitystehtävän tässä vaiheessa todettu tärkeiksi.

(2) Vaatimukselle määritelty vapaamuotoinen sanallinen kuvaus.

Peilattaessa PETin suunnittelun osa-alueiden toimintaa ohjelmistotuotantoon todettiin, että toimivin vaatimuksen dokumentointitekniikka olisi vapaamuotoinen, sanallinen kuvaus vaatimuksesta. Jirassa vaatimuksen dokumentointilomakkeeseen kirjoitetaankin siis vapaasti kuvaus vaatimuksesta, toteuttaen kuitenkin kappaleessa 5.1.3 esitettyjä vaatimuskirjeen minimivaatimuksia. Lomakkeelle on mahdollista lisätä myös liitetiedostoja. Liitteissä voidaan selventää vaatimusta esimerkiksi lohkokaaviokuvalla tai muulla vastaavalla lisätiedolla vaatimuksesta.

(3) Vaatimuksen työkierron tila

Vaatimuksen työkierron tila ilmoittaa missä vaiheessa työkiertoa kyseinen vaatimus luku hetkellä on. Jira-järjestelmään on rakennettavissa vaatimuksien hallintaa tukemaan erilaisia koontinäyttöjä, joista voidaan seurata listamuotoisesti eri vaatimuksien tiloja. Vaihtoehtoisesti Jiraan voidaan luoda prosentuaalisia näyttöjä, jotka kuvaavat missä vaiheessa vaatimuksien eteneminen kokonaisuutena on.

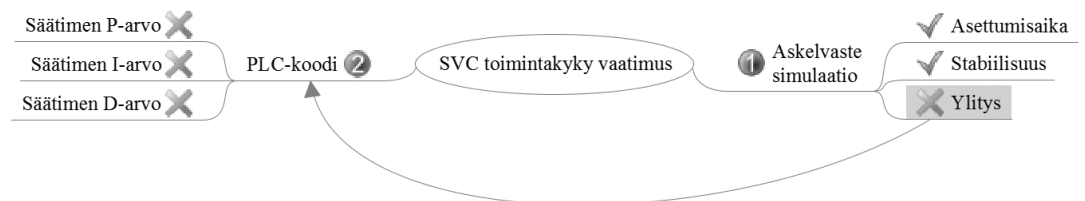
(4) Vaatimuksen hallintaan tarkoitetut painikkeet

Vaatimus näytön ylä-reunassa on yksittäisen vaatimuksen hallintaan tarkoitetut painikkeet. Painikkeilla hyväksytään vaatimuksen senhetkinen työkierto tai lisätään esimerkiksi kommentti vaatimukselle.

(5) Vaatimuksen yksilöivä vaatimusnumero ja vaatimukselle laadittava otsikko

Luotaessa Jiraan vaatimus luo järjestelmä automaattisesti sille yksilöidyn tunniste, kuvan 25 tapauksessa tunniste on PET75GAL-10. Tunnisteen avulla voidaan Jirassa luoda linkityksiä ja riippuvuuksia eri vaatimuksien välille. Näiden tarkoituksena on luoda syvyyttä vaatimuksien toteuttamiselle.

Linkityksiä on Jirassa useita erityyppisiä, esimerkiksi estävällä linkillä voidaan rajoittaa vaatimuksen työkierron etenemistä ennen kuin toisaalla oleva vaatimus on saatu sopivaan työkierron tilaan. Toisaalta linkki voi olla pelkästään informatiivinen, joten sillä ei ole riippuvuussuhdetta kahden eri vaatimuksen välillä. Vaatimukselle on myös mahdollista luoda ali-vaatimuksia. Ali-vaatimuksien tarkoituksena on rakentaa tarvittaessa syvempää hierarkiaa vaatimuksille, kuten kappaleessa 6.1 todettiin. Laajoissa projekteissa hierarkialle voi syntyä tarvetta, kun vaatimushallintaa, luomista ja toteutusta suorittaa isompi suunnittelija joukko. Kuvassa 27 on esitettyä kuvitteellinen esimerkki vaatimuksen hierarkiasta ja linkityksestä.



KUVA 27. Vaatimuksien linkittäminen toisiinsa

Kuvan 27 tapauksessa SVCeen toimintakyvylle on asetettuna askelvastesimulaatiolle ei-toiminnalliset vaatimukset. Simulaatioiden tuloksena järjestelmän suunnittelussa asetetut säätimen parametrit eivät toimi, tuloksena laitteiston ohjauksessa tapahtuu liian suuri ylitys. Tämän seurauksena PLC-ohjelmassa olevien säätimien parametrien asettaminen ei voi alkaa ennen kuin askelvaste simulaatioiden tulokset ovat hyväksytysti toteutettu. Kyseisessä esimerkissä askelvastesimulaation ja PLC-koodin välillä on asetettu etenemistä estävä linkki. PLC-koodi vaatimuksessa ei päästä etenemään ennen kuin askelvastesimulaatio vaatimus on päässyt työkiertonsa loppuun.

8 TUTKIMUKSEN TULOKSIEN YHTEENVETO

Kehitystehtävän tavoitteeksi asetettiin vaatimusten määrittelyn ja hallinnoimisen toteutustapojen kartoittaminen sekä arvioiminen millainen vaatimusten käsittelyjärjestelmä pystytään luomaan työn tilaajan käytössä olevan ohjelmiston varaan. Työn tilaajalla ei ollut aikaisempia kokemuksia vaatimuksien käsittelystä, joten määränpäänä tulee olemaan tehokkaampi projektien toteuttaminen hyödyntämällä vaatimusten käsittelyä.

Työn ensimmäisessä osassa tutustuttiin kirjallisuuslähteiden kautta mitä vaatimusten määrittely ja vaatimusten hallinta todellisuudessa tarkoittaa. Tavoitteena oli tutkia taustoja, joiden päälle vaatimusten käsittely tulisi yrityksissä muodostaa. Vaatimusten käsittelyn perusteita tutkittaessa tehtiin huomio siitä, että vaatimukset eivät ole pelkästään muistilappuja suunnittelijoiden ja projektijohdon välillä. Sen sijaan vaatimusten käsittelyn taustalla on ajatusmalli systemaattisesta ja askeleittain etenevästä prosessista.

Työntilaajalla on pitkä historia räätälöityjen projektien läpiviemisestä ja kompensointilaitteistojen suunnittelusta, joten suunnittelumenetelmiin ei tarvitse tehdä muutoksia. Kehitystehtävän tutkimuksen perusteella suurimmat muutokset tarvitaan suunnittelun aloitukseen ja suunnittelun aikaisien muutoksien käsittelyyn. Näitä muutoksia varten luotiin kehitystehtävässä uusi prosessimalli. Sen tavoitteena on tukea suunnitteluprosessia vaatimusten käsittelyllä ja luoda vaatimusten käsittelylle automaattisesti toistuvat katselmoi-
mointi ja hyväksyntäkäytännöt. Liitteessä 3 on esitettyä ehdotus suunnitteluprosessin vesiputousmallisesta etenemisestä.

Tutkimuksessa tehtyjen haastatteluiden perusteella voidaan todeta, että vaatimusten kartoittamisessa tulisi käyttää selviä rajauksia. Niiden tarkoituksena on helpottaa standarditoteutuksesta poikkeavien asioiden poimimista vaatimusten käsittelyä varten. Suunniteltavasta systeemistä tulee olla ensiksi selvät standardikuvaukset. Kun, vaatimuksia aletaan projektille kartoittamaan pystyvät suunnittelijat nojaamaan standardikuvaukseen laitteesta. Kirjattavat vaatimukset ovat silloin poikkeamia tähän standardiin. PETillä ei ole tällä hetkellä selvää ja yksityiskohtaista kuvausta siitä millainen on yrityksen standardituote. Näin ollen voidaan tutkimuksen pohjalta todeta, että vaatimusten käsittelyä tukisi huomattavasti standardikuvauksien luominen.

Kehitystehtävässä tutkittiin myös kuinka työn tilaajalla käytössä oleva Jira-järjestelmä saadaan tehokkaasti valjastettua tukemaan vaatimusten käsittelyä. Kehitystehtävän tutkimustuloksien pohjalta Jiraan pystyttiin luomaan vaatimusten määrittelemistä varten alustava lomakepohja ja sille kaksi erilaista työkiertoa. Jiralle tehtiin suppeita testejä, joiden perusteella voidaan todeta vaatimusten määrittelemisen ja hallinnoinnin onnistuvan Jiran avulla. Kehitystehtävän aikana ei päästy testaamaan todellisessa projektissa vaatimusten määrittelyyn ja hallintoihin luotuja prosesseja. Täten voidaan epäillä, että prosessien ja järjestelmien laajamittaisemmassa käyttöönotossa tullaan törmäämään käytännön ongelmiin, joita ei ole osattu tämän kehitystehtävän puitteissa ottaa huomioon.

Kehitystehtävän tavoitteena ei ollut Jiran toimintaohjeiden luominen vaatimusten käsittelemiseksi. Ennen vaatimusten käsittelyn laajamittaista käyttöönottamista suositellaan ohjeistusmateriaalin luomista ja koulutustilaisuuden pitämistä. Ohjeistusmateriaalin lisäksi käyttöönottamisen tarvitsee riittävän yksinkertaisen projektin, jotta käytännönongelmien ratkaisemisesta ei muodostuisi liian haastavaa samanaikaisesti, kun uusi järjestelmä on muutenkin käyttäjillä tutustumisasteella.

Kehitystehtävän aikana tutustuttiin niin sanottuihin ketteriin suunnittelumenetelmiin. Niiden vahvana puolena on nopea sopeutuminen muutoksiin ja kommunikoinnin helppous projektin henkilöstön muuttuessa projektin elinkaaren aikana. Ketterien menetelmien ollessa suunniteltu ensisijaisesti ohjelmistotekniikkaa palveleviksi ei niistä suoraan pystytä saamaan toimivia työtapoja PETin käyttöön. Menetelmistä olisi kuitenkin sovellettavissa toimintatapoja, joilla saataisiin edelleen kehitettyä vaatimusten käsittelyä. Jira-järjestelmä on myös rakennettu tukemaan tunnetuimpia ketteriä menetelmiä, kuten scum ja kanban. Jatkokehityksen tavoitteeksi tulisi asettaa menetelmiin tutustuminen ja PETin suunnitteluprosessin hiominen entistä sujuvammaksi.

Kappaleessa 2.3 esiteltiin SEI-järjestön määrittelemät ohjelmistojen kypsyyttä kuvaavat portaat. Kappaleessa todettiin, että PETin suunnitteluprosessissa on edellytyksiä nousta korkeammalle kypsyyssasteelle, mutta tällä hetkellä toiminta ei vastaa kehittyneelle laitteiston suunnittelulle määriteltyjä kypsyyssasteita. Kehitystehtävän tuloksena vaatimusten käsittelystä on saatavissa suunnittelua tukeva prosessi, minkä perusteella suunnittelun laatua saadaan helposti kehitettyä.

LÄHTEET

Charles, A., Joseph, J. & John, E. 2009. ISO 9001:2008 Explained. Third Edition. Milwaukee: ASQ Press. Sähköinen aineisto.

Chin, G. 2003. Agile Project Management: How to Succeed in the Face of Changing Project Requirements. New York: AMACOM Books. Sähköinen aineisto.

Paul., D. Cadle, J. & Yeates, D. 2014. Business Analysis - Third edition. Swindon: BCS Learning & Development Ltd. Sähköinen aineisto.

Eskola, J. & Suoranta, J. 1998. Johdatus laadulliseen tutkimukseen. Tampere: Vastapaino. Sähköinen aineisto.

Fowler, K. 2014. Developing and managing Embedded Systems and products. Oxford: Elsevier. Sähköinen aineisto.

Haikala, I. Mikkonen, T. 2011. Ohjelmistotuotannon käytännöt. Helsinki: Talentum

Haikala, I. Märijärvi, J. 2004. Ohjelmistotuotanto. Helsinki: Talentum

Heikkinen, J. Tarjouspäällikkö. 2016. Haastattelu 21.4.2016. Haastattelija Lammi, M. Tampere

Holt, J., Bronsword M & Perry S. 2012. Model-Based Requirements engineering. Stevenage. The Institution of Engineering and Technology. Sähköinen aineisto.

ISO/IEC/IEEE 29148:2011. 2011. System and software engineering – Life cycle processes – Requirements engineering. IEEE Xplore.

ISO/IEEE 15288:2015. 2015. Systems and software engineering – Systems life cycle processes. IEEE Xplore.

Jira User guide 6.4. 2016. Viitattu 29.2.2016. Jira ohjelmiston tukisivusto. <https://confluence.atlassian.com/jira064/jira-user-s-guide-720416011.html>

JIRA 6.4. 2016. Kuvakaappaus otettu 1.3.2016. Kuvakaappaus.

Kenett, R. & Baker, E. 1999. Software Process Quality: Management and Control. New York: CRC Press. Sähköinen aineisto.

Karjalainen, M. Suunnittelija. 2016. Haastattelu 3.3.2016. Haastattelija Lammi, M. Tampere.

Kolehmainen, M. Sähkösuunnitteluryhmän vetäjä. 2016. Haastattelu 28.2.2016. Haastattelija Lammi, M. Tampere.

- Kotonya, G. & Sommerville, I. 1997. Requirements Engineering. Chichester: John Wiley & Sons Ltd.
- Leffingwell, D. & Widrig, D. 1999. Managing Software Requirements. First edition. Addison Wesley.
- Mackowiak, G. & Cooper, M. 2016. Puhelinkeskustelut.
- Metodix: Metoditietämystä kaikille. 2016. Luettu 5.2.2016. <https://metodix.wordpress.com/2014/05/19/lukka-konstruktiivinen-tutkimusote/>
- Männistö, P. 2011. Satu häpäntimestä ja kuinka siitä olisi tullut kunnollinen. Projektitoiminta 1/2011. Helsinki: Projekttyhdistys ry
- Project management institute. 2008. A guide to the project management body of excellence. USA: Project Management Institute, Inc. Sähköinen aineisto.
- Pohl, K. & Rupp, C. 2015. Requirements Engineering Fundamentals. 2nd Edition. Santa Barbara: Rocky Nook Inc. Sähköinen aineisto.
- Pohjonen, R. 2002. Tietojärjestelmien kehittäminen. 2. painos. Jyväskylä: Docendo
- Rakitin, S. & Steven, R. 2001. Software Verification and Validation for Practioners and Managers. Second Edition. Lontoo: Artech house Sähköinen aineisto.
- Raussi, T. & Virtanen, P. 2009. Vaatimusmäärittelyä ketterästi vai perinteisesti ? Systemityö 2/2009. Espoo: Sytyke Ry.
- Robertson, S. & Ronertson, J. 2010. Mastering the Requirements Process. Second edition. Boston: Addison-Wesley.
- Stellman, A. & Greene, J. 2015. Learning Agile. USA: O'Reilly Media, Inc.
- Tsui, F. & Orlando, K. 2007. Essentials of Software Engineering. Massachusset: Jones and Bratlett Publishers. Sähköinen aineisto.
- Viranen, P. 2011. Lauman johtaminen hämärässä –johdatus systeemiajatteluun. Projektitoiminta 1/2011. Helsinki: Projekttyhdistys ry
- Vuori, M. 2009. Vaatimusmäärittelyn huonoimmat käytännöt. Systemityö 2/2009. Espoo: Sytyke Ry.

LIITTEET

Liite 1. Haastattelulomake (tiivistetyt vastaukset)

Kategoria 1.

Miten paljon olette olleet tekemisissä vaatimuksien hallinnan ja keräämisen parissa ?

Karjalainen: Käyttänyt suunnittelun apuvälineenä ja kerännyt muutoksia työkaluun.

Kolehmainen: Käyttänyt suunnittelun apuvälineenä ja kerännyt muutoksia työkaluun.

Heikkinen: Aikaisemmissa työtehtävissä erittäin paljon.

Kategoria 2.

Millaista prosessimallia yrityksessänne noudatetaan / noudatettiin ?

Karjalainen: Vesiputousmallinen, mutta soveltaen.

Kolehmainen: Vesiputousmallinen, mutta soveltaen.

Heikkinen: Gatereview mallit.

Miten näet PETin prosessimallin toimivuuden (kysytään omassa organisaatiossa toimivalta) ?

Karjalainen: - (Organisaation ulkopuolelta).

Kolehmainen: Vesiputousmallinen, askeleittain etenevä prosessimalli on toimiva. Siihen tulisi saada suunnittelun muutokset kuitenkin selvästi kuvattua.

Heikkinen: Kehitettävää on monella saralla. Perus asioiden haltuun ottaminen on oleellista, jotta vaatimusten hallinnasta saadaan koko järjestelmän elinkaarta tukeva.

Kategoria 3.

Kuinka vaatimuksien kerääminen projektin alussa toteutetaan ?

Karjalainen: Projektipäällikkö tunnustelee sisäisen asiakkaan tarpeet

Kolehmainen: Projektipäällikkö vastuussa keräämisen organisoinnista, mutta vaatimuksia keräävät projektin pääsuunnittelija ja oleellisimpien sidosryhmien vetäjät.

Heikkinen: Projektipäällikön lisäksi tarjouspäällikkö on merkittävässä roolissa elinkaarimallia ajateltaessa.

Katselmoidaanko vaatimukset ennen suunnittelun aloittamista ?

Karjalainen: Katselmoidaan.

Kolehmainen: Vaatimukset tulee katselmoida ennen suunnittelun aloittamista.

Heikkinen: Koko organisaation prosessien ollessa oikein viritettynä, vaatimusten katselmoinnit tapahtuvat syklisestä oikeissa tuotealustan julkaisu tai muutos hetkissä.

Kuka kerää projektin vaatimukset ?

Karjalainen: Pääsuunnittelija / suunnittelijat.

Kolehmainen: Pääsuunnittelija ja projektipäällikkö.

Heikkinen: Vaatimusten kerääminen tulisi olla koko järjestelmän elinkaaren ajan toistuvaa. Ajateltaessa koko elinkaarta, ei silloin voida osoittaa yhtä tahoa joka olisi vastuussa vaatimusten keräämisestä.

Kategoria 4.

Kuinka vaatimuksien muuttuminen ja uusien vaatimuksien ilmestyminen toteutusvaiheessa on hallinnoitu ?

Karjalainen: Suunnittelun etenemistä seurataan vaatimuksien avulla. Ensin tiuhempaan tahtiin ja myöhemmin katselmointi väliä pidentäen. Suunnittelija ylläpitää vaatimuksien tilanteen.

Kolehmainen: Projektipäällikkö ja pääsuunnittelija ylläpitää listaa vaatimuksista.

Heikkinen: Oikeanlaiset työryhmät ja muutoksien dokumentointi on tärkeää. Muutoksista tulee myös julkaista muutosdokumentit. Tämä tosin vaatii tuotealusta tyyppistä tuotehallintaa.

Millaisia järjestelmiä on käytössä ?

Kolehmainen: Taulukkolaskenta pohjainen työkalu.

Karjalainen: Taulukkolaskenta pohjainen työkalu.

Heikkinen: Heikkisellä oli laajasti kokemusta ohjelmistopuolen vaatimusten käsittelyjärjestelmien kehittämisestä. Hän suositteli ehdottomasti erillistä järjestelmää PETIn koikoisiin projekteihin.

Liite 2. Vaatimuksen dokumentointilomake Jira-järjestelmässä (Jira 6.4 ohjelmisto 2016).

Create Issue
⚙️ Con

Project* + 2014-75-Gallopier USVC

Issue Type* RequirementSW

Summary*

Priority* Medium

Description*

Labels
Begin typing to find and create labels or press down to select a suggested label.

Requirement Category* None
Requirement Category Description

Verification Method* None
Verification Method Description

Attachment

Drop files here to attach them
or
Select files

Linked Issues [Blocks] blocks +
Begin typing to search for issues to link. If you leave it blank, no link will be made.

Requirement Source*
Requirement Source Description

Affects Version/s **None**

Component/s
Start typing to get a list of possible matches or press down to select.

Derived From

Existing None

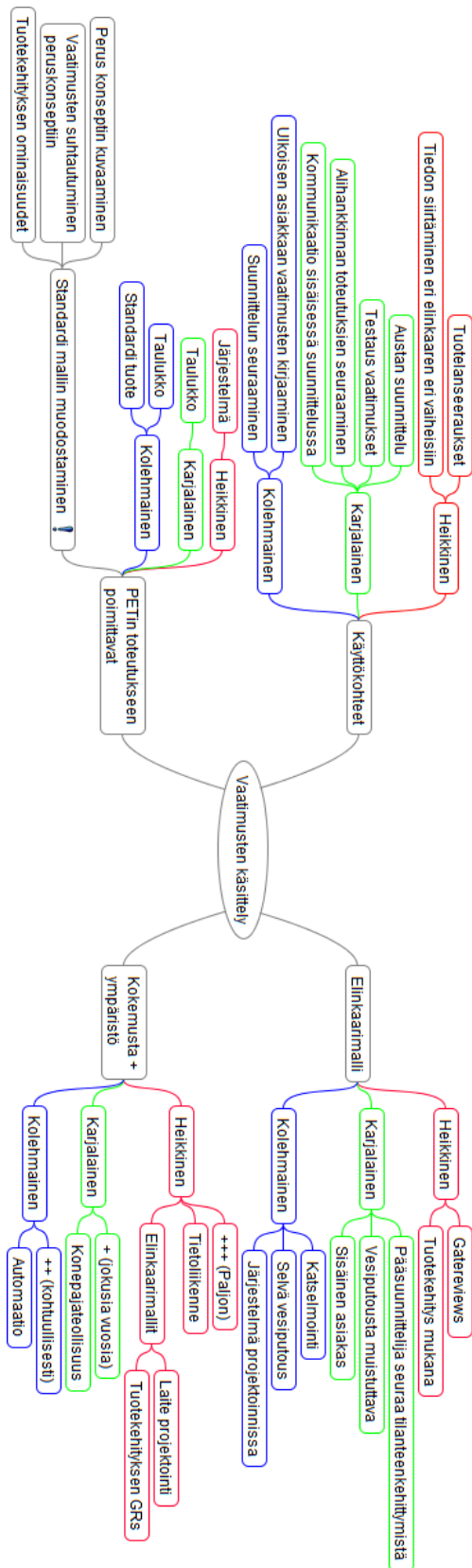
Requirement Id
Original Requirement Id (Used for Data Import)

Requirement Classification None
Requirement Classification

Schedule Notes

Create another
Create

Liite 4. Haastattelujen mind-map kaavio



Liite 5. Vapaasti kirjoitun vaatimuksen ohjenuorat

Necessary. The requirement defines an essential capability, characteristic, constraint, and/or quality factor. If it is removed or deleted, a deficiency will exist, which cannot be fulfilled by other capabilities of the product or process. The requirement is currently applicable and has not been made obsolete by the passage of time. Requirements with planned expiration dates or applicability dates are clearly identified.

Implementation Free. The requirement, while addressing what is necessary and sufficient in the system, avoids placing unnecessary constraints on the architectural design. The objective is to be implementation-independent. The requirement states what is required, not how the requirement should be met.

Unambiguous. The requirement is stated in such a way so that it can be interpreted in only one way. The requirement is stated simply and is easy to understand.

Consistent. The requirement is free of conflicts with other requirements.

Complete. The stated requirement needs no further amplification because it is measurable and sufficiently describes the capability and characteristics to meet the stakeholder's need.

Singular. The requirement statement includes only one requirement with no use of conjunctions.

Feasible. The requirement is technically achievable, does not require major technology advances, and fits within system constraints (e.g., cost, schedule, technical, legal, regulatory) with acceptable risk.

Traceable. The requirement is upwards traceable to specific documented stakeholder statement(s) of need, higher tier requirement, or other source (e.g., a trade or design study). The requirement is also downwards traceable to the specific requirements in the lower tier requirements specification or other system definition artefacts. That is, all parent-child relationships for the requirement are identified in tracing such that the requirement traces to its source and implementation.

Verifiable. The requirement has the means to prove that the system satisfies the specified requirement. Evidence may be collected that proves that the system can satisfy the specified requirement. Verifiability is enhanced when the requirement is measurable.