



**LAHDEN AMMATTIKORKEAKOULU**  
*Lahti University of Applied Sciences*

# PROJEKTIHALLINTA

Ohjelmiston käyttöönottoprojekti

LAHDEN  
AMMATTIKORKEAKOULU  
Tekniikan ala  
Tietotekniikan koulutusohjelma  
Tietotekniikan suuntautumisvaihtoehto  
Opinnäytetyö  
Kevät 2015  
Tomi Männistö

Lahden ammattikorkeakoulu  
Tietotekniikan koulutusohjelma

MÄNNISTÖ, TOMI:

Projektinhallinta  
Ohjelmiston käyttöönottoprojekti

Tietotekniikan opinnäytetyö, 45 sivua

Kevät 2015

TIIVISTELMÄ

---

Tässä opinnäytetyössä on tarkasteltu yleisiä projektikäytännön malleja. Olemassa olevia projektityömalleja on useita ja ne soveltuvat erilaisiin tarkoituksiin ja erilaisille kohderyhmille. Yhteistä näille malleille on niiden järjestelmällisyys ja ne soveltuvat käytettäväksi erilaisiin projekteihin. Hyvä projektityöskentely vaatii selkeän projektiryhmän kokoonpanon ja selkeät roolit projektiryhmän sisällä.

Teoriaosuudessa on esitelty perinteinen projektimalli, vesiputous-malli, ketterät menetelmät ja Scrum. Lisäksi on tarkasteltu tyypillisimpiä projekteissa esiintyviä ongelmia ja riskienhallintaa. Työssä keskitytään ennen kaikkea ohjelmistoprojekteihin ja niiden eri vaiheisiin. Tarkastelussa ovat projektin valmistelu, suunnittelu, projektin osittaminen, projektisuunnittelun vaihetuotteet, toteutus, testaus, käyttöönotto, projektin päättäminen ja lisäksi mahdollinen projektin keskeyttäminen.

Tutkimusosuudessa tarkastellaan kohdeyrityksessä toteutettua projektia ja sen onnistumista työssä esitellyn teorian pohjalta. Yrityksen sisällä toteutettua projektia tarkastellaan vaiheittain. Projektityön vaiheita ja sen onnistumista on reflektoitu projektiin osallistuneiden kesken.

Työn tulosten pohjalta voidaan todeta, että projektityöskentely vaatii kehittämistä kohdeorganisaatiossa. Projektityöskentely ei noudattanut hyvän projektityöskentelyn mallia. Yrityksen johdon näkökulmasta katsottiin, että asetetut tavoitteet saavutettiin, mutta kustannustehokkuuden ja paremman projektityöskentelyn saavuttamiseksi olisi suositeltavaa, että yrityksessä panostettaisiin kirjattuun projektityökäytäntöön. Lisäksi voidaan todeta, että ilman tarkkaa suunnitelmaa ja väliarviointia tehdyssä projektissa tuhlattiin paljon resursseja, mikä olisi voitu välttää tehokkaammalla suunnittelulla ja väliarvioinneilla.

Asiasanat: Projekti, projektinhallinta

Lahti University of Applied Sciences  
Degree Programme in Information Technology

MÄNNISTÖ, TOMI:

Project management  
Software initialization project

Bachelor's Thesis in information technology, 45 pages

Spring 2015

ABSTRACT

---

The objective of this thesis is to give a review of the project practice models that are commonly used. There are several project work models which can be used for different purposes and for different target groups. These models have in common that they are all systematic and well-organized and they can be used in different kinds of projects. Good project work requires a clear project group setup where everyone has a clear understanding of their own role in the group.

The theoretical part introduces different project models like traditional model, fountain model, agile methods and Scrum, which is one type of agile method. Typical problems, which are often faced in project working, are also examined, and so is risk management. This thesis concentrates especially on software projects and the different stages of those projects. Typically the stages are preparation, planning, dividing the project in parts, project planning and outputs, execution, testing, initialization, ending the project and possibly intermitting the project.

In the practical part one software project that was executed in the target company was under review. The project and its succeeding were evaluated based on the theory.

Based on the results of this survey, it can be stated that project practices need to be improved in the target company. Project working did not follow any good project practice models. The management of the target company says that the goals for the executed project were reached. However, if the company invested in the general project practices inside the company it could reach better cost efficiency and the projects would be managed better. Also, the project was executed without proper planning and interim evaluation during the project, which contributed to wasting resources, which could be avoided by using a proper project practice model.

Key words: Project, project management

## SISÄLLYS

1	JOHDANTO	1
2	PROJEKTIT	3
2.1	Projektiryhmän kokoonpano ja roolit	4
2.2	Projektinhallintamalleja	6
2.2.1	Perinteinen projektimalli	7
2.2.2	Vesiputousmalli	7
2.2.3	Ketterät menetelmät	9
2.2.4	Scrum	10
2.3	Tyypillisiä ongelmia projekteissa	12
2.4	Riskien hallinta	15
3	OHJELMISTOPROJEKTIT	17
3.1	Valmistelu	17
3.2	Suunnittelu	18
3.2.1	Projektin osittaminen ja aikataulus	19
3.2.2	Ganttin kaavio	20
3.2.3	Projektisuunnittelun vaihetuotteet	22
3.3	Toteutus	22
3.4	Testaus	24
3.5	Käyttöönotto	28
3.6	Projektin päättäminen	29
3.7	Projektin keskeyttäminen	29
4	PROJEKTIN TOTEUTUS	31
4.1	Valmistelu	33
4.2	Suunnittelu	34
4.3	Toteutus	34
4.4	Testaus	35
4.5	Käyttöönotto	35
4.6	Päättäminen	36
4.7	Ylläpito	36
5	JOHTOPÄÄTÖKSET JA KEHITYSKOHEET	37
5.1	Olemassa olevat hyvät projektikäytännöt	38
5.2	Kehityskoheet	38

6 YHTEENVETO

41

LÄHTEET

42

# 1 JOHDANTO

Liiketoiminnassa kaikella toiminnalla on viime kädessä vain yksi tarkoitus: tuottaa omistajilleen voittoa. Toimivat ja tarkoituksen mukaiset tietojärjestelmät ja ohjelmistot ovat edellytys kannattavalle liiketoiminnalle. Niiden avulla voidaan säästää aikaa, parantaa resursointia, hallita varastoja ja saada keskeisiä reaaliaikaisia operatiivisia ja strategisia tunnuslukuja. Tietojärjestelmien avulla voidaan tuotannosta löytää kustannustehokkuuteen vaikuttavia seikkoja, joita muuten olisi vaikea tai työläs havaita.

Tietojärjestelmien ja ohjelmistojen käyttöönottoprojektit voivat onnistuessaan tuottaa merkittäviä säästöjä ja taloudellista etua kilpailijoihin nähden, mutta epäonnistuessaan ne voivat aiheuttaa pitkäaikaisia lisäkustannuksia muun muassa määräaikaisten lisenssimaksujen muodossa. Lisäksi harvoin tunnutaan ottavan opiksi epäonnistuneista projekteista ja niiden toteutuksesta.

Työ on tehty sähkö- ja tietoliikenneverkkopalveluiden sekä teollisuuspalveluiden tuottajayritykselle. Yrityksen nimeä ei työssä mainita vaan siitä käytetään nimitystä Yritys Oy.

Yritys Oy:n asema omalla toimialallaan on vahvistunut vuosien ajan. Vuonna 2013 Yrityksen liikevaihto oli noin 80 miljoonaa euroa ja liikevoitto noin 4 miljoonaa euroa. Sen palveluksessa oli keskimäärin 500 henkilöä. Yrityksen ovat perustaneet kaksi toimialansa merkittävää toimijaa. Nykyisin yhtiöllä on vain yksi omistaja. (Yritys Oy 2014.)

Yritys tekee alihankintana sähkö- ja tietoverkkojen verkonrakennus-, viankorjaus- ja kunnossapitotöitä. Lisäksi yhtiö tuottaa verkko-operointi- ja dokumentointipalveluita. Sen asiakkaina on valtakunnallisia sähköyhtiöitä, tietoverkko- ja teleoperaattoreita, kuntia, kaupunkeja sekä yksityishenkilöitä.

Yritys Oy:ssä on otettu vuoden 2014 aikana käyttöön ohjelmisto, jonka avulla pyritään parantamaan yhtiön sisäistä raportointia. Ennen käyttöönottoa lähdetieto on hajallaan erilaisissa järjestelmissä, ja tarkoitus on koota tiedot yhteen ja sitä kautta löytää kustannustehokkuutta heikentäviä seikkoja sekä saada reaaliaikaisia talouden tunnuslukuja.

Työssä esitetään ohjelmistoprojektimalleja ja hyviä käytäntöjä sekä verrataan tehdyn projektin toteutusta näihin malleihin. Tavoitteena on löytää kehitettävää yhtiön ohjelmistoprojektityöskentelyssä ja keinoja, joilla saavutetaan kustannussäästöjä.

## 2 PROJEKTIT

Sana projekti tulee latinan kielen verbistä proicere johdetusta partisiipin perfektistä proiectum. Proicere tarkoittaa ulos- tai eteen heittämistä.

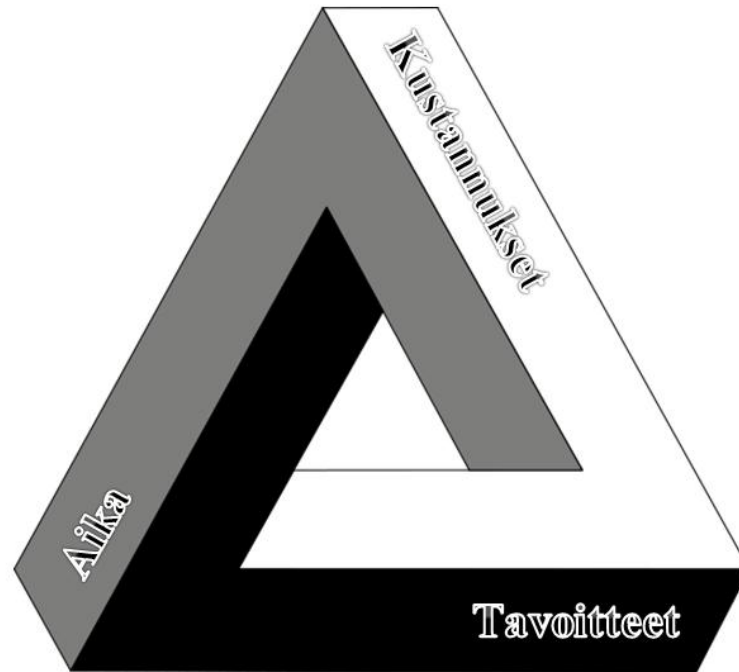
Projekti-instituutti määrittelee projektin kertaluonteiseksi työksi, joka tehdään ainutkertaisen tuotteen, palvelun tai tuloksen aikaansaamiseksi. Vaikka projekti voi olla mitä tahansa vaativan rakennusprojektin tai pienen markkinointiprojektin välillä, on projektille tyypillistä, että sille on määritelty etukäteen aika- ja kustannustavoite sekä hyöty- ja lopputulostavoitteet. (Projekti-instituutti 2014.)

Markkinointiprojekti voi päättyä siten, että dokumentoidaan projektin vaikutus tuotteen myyntiin ja analysoidaan tulokset. Epäonnistunut rakennusprojekti voi päättyä rakennuttajan konkurssiin. Onnistunut IT-projekti voi päättyä siihen, että sovellus otetaan käyttöön suunnitellussa aikataulussa ja sen huomataan vastaavan toiminnallisiin ja liiketaloudellisiin tavoitteisiin.

Projektin onnistumiseen vaikuttaa moni tekijä, kuten tekniset riskit, yrityksen politiikka, tiimin sisäiset ristiriidat ja äkilliset muutokset. Kaikkia tekijöitä ei voi hallita, mutta tiedostamalla riskit voidaan ennaltaehkäistä monia virheitä. (Litke & Kunow 2004, 9).

Projektit ovat monimutkaisia kokonaisuuksia. Niiden tehokas, tuloksellinen ja onnistunut päätökseen saattaminen on erityisen tärkeää yritystoiminnassa. Projektityöskentely on tiimityötä ja yhteen kokoontuu joukko eritaustaisia ihmisiä, joilla on osaamista eri aloilta ja myös erilaiset ajatusmallit. Vaikka projektin kokonaistavoite olisi selvä, tapa ratkaista ongelma ei kuitenkaan yleensä ole selvä. Projekteihin liittyy usein myös suuria investointeja ja on yleistä, että budjetit ylittyvät. Epäonnistuneet projektit merkitsevät siis myös yleensä taloudellisia menetyksiä. Vastauksena näihin ongelmiin on syntynyt projektinhallinnan käsite. (Litke & Kunow 2004, 10–12.)





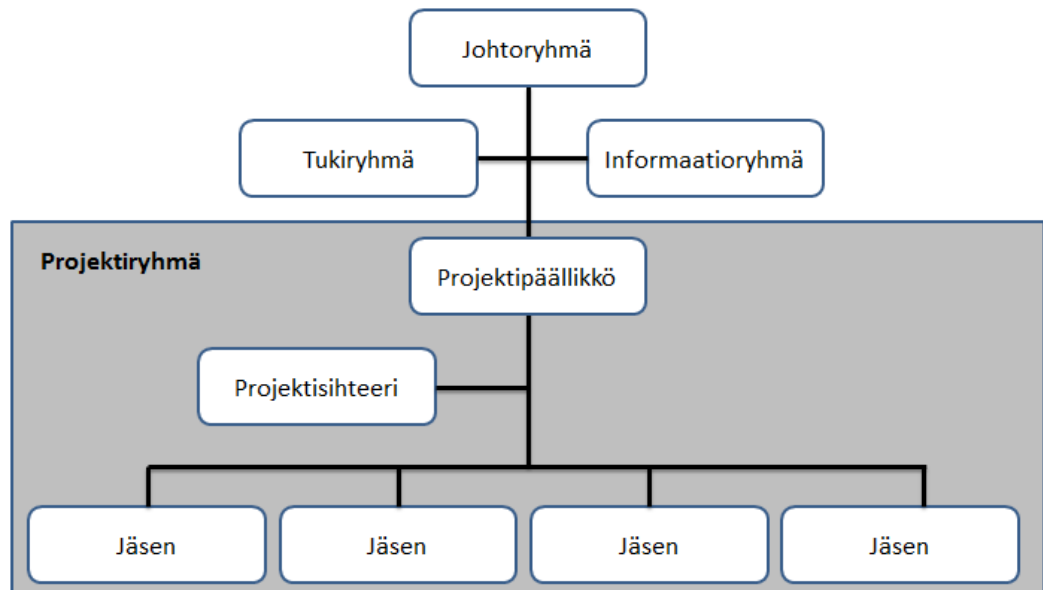
KUVIO 1. Projektin ”pyhä kolminaisuus”

Kuviossa 1 Penrosen kolmio esittää ajan, kustannusten ja tavoitteiden vaikeasti hallittavaa suhdetta. Mitä isommat tavoitteet projektilla on, sitä kauemmaksi keskustasta myös muut kolmion kärjet karkaavat. Jos aikaa kasvatetaan, kasvavat myös projektin kustannukset. Toisaalta kustannuksia tietoisesti lisäämällä voidaan projektin aikataulua kutistaa. Työtuntimäärää kustannuksia lisäämällä ei kuitenkaan voida merkittävästi vähentää. (Litke & Kunow 2004, 16.)

## 2.1 Projektiryhmän kokoonpano ja roolit

Yleensä projekteja varten pitää muodostaa projektiryhmä, jonka tehtävä on suorittaa annettu tehtävä annetussa ajassa (Litke & Kunow 2004, 60).

Projektiryhmän kokoonpano riippuu käytettävästä projektinhallintamenetelmästä.



KUVIO 2. Projektioorganisaatio

Kuvio 2 esittelee yhden projektioorganisaation perusmallin. Varsinainen projektiryhmä koostuu projektipäälliköstä, -sihteeristä ja jäsenistä. Sen ulkopuolelle jää johto-, tuki- ja informaatioryhmät.

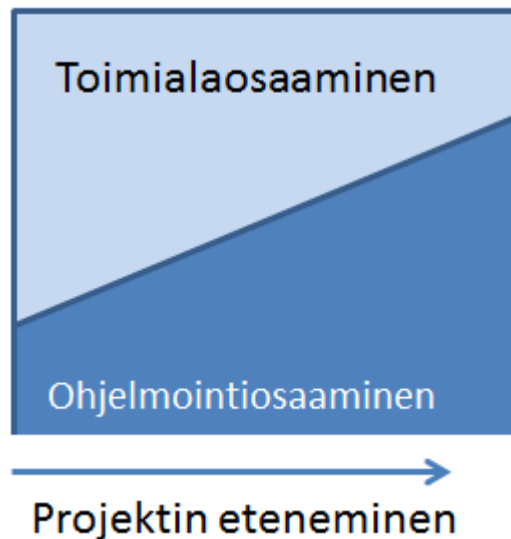
Johtoryhmä on projektioorganisaation korkein päättävä elin, ja siihen kuuluu sekä toimittajan että tilaajan edustajia. Sen tehtäviin kuuluu projektin valvonta, projektisuunnitelmien käsittely ja projektin lopettamisesta päättäminen. (Mikkelin ammattikorkeakoulu 2002.)

Informaatioryhmässä on edustajia kaikista yksiköistä. Sitä käytetään isoissa projekteissa, jolloin johtoryhmän kokoonpanoa saadaan pienennettyä. Tuki- ja informaatioryhmiä voidaan perustaa tarvittaessa. Tukiryhmän tehtävänä on laadun varmistaminen, ja se koostuu yleensä alan asiantuntijoista. (MAMK 2002.)

Tukiryhmällä on tukeva ja neuvoa antava rooli, mutta sillä ei ole päätösvaltaa (Litke & Kunow 2004, 21).

Projektipäälliköllä on kokonaisvastuu projektista. Hän laatii projektisuunnitelman, johtaa projektin läpivientiä vastaten kustannuksista, aikataulusta ja resursseista sekä vastaa projektin tuotosten sisällöstä ja laadusta. Projektipäällikkö raportoi johtoryhmälle. (MAMK 2002.)

Projektiryhmään voi kuulua myös projektisihteeri, joka huolehtii käytännön asioista ja projektin dokumentoinnista. Projektiryhmän jäsenet huolehtivat omista tehtävistään, dokumentoivat ne ja raportoivat projektipäällikölle. (MAMK 2002.)



KUVIO 3. Henkilöresurssien osaamistarve ohjelmistoprojektin edetessä

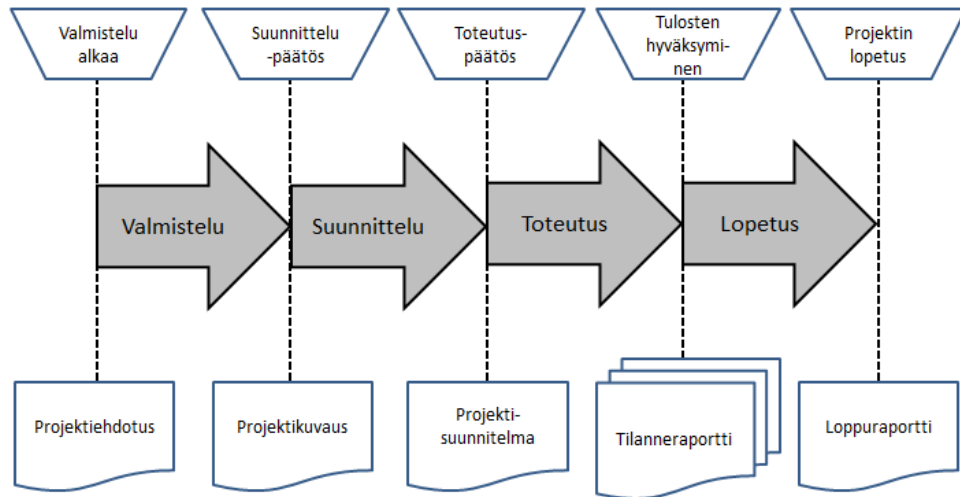
Kuvio 3 esittää toimialaosaamisen ja ohjelmointiosaamisen tarpeen suhdetta projektin edetessä (Ansamäki 2002). Projektin alkuvaiheessa, kun tehdään ohjelmiston vaatimusmäärittelyä, toimialaosaaminen on ohjelmisto-osaamista tärkeämpää. Loppupuolella, kun ohjelmistoa toteutetaan, nousee ohjelmointiosaaminen korkeampaan rooliin.

## 2.2 Projektinhallintamalleja

Projektille tyypillisiä työvaiheita ovat aloitus, suunnittelu, toteutus ja lopetus. Erilaisissa projektimalleissa työvaiheita on erilainen määrä ja niissä eteneminen tapahtuu eri tavoin. Toisissa malleissa voidaan palata aiempaan vaiheeseen, milloin vain, kun taas toisissa malleissa palaaminen on mahdollista vain tietyissä vaiheissa.

### 2.2.1 Perinteinen projektimalli

Perinteinen projektimalli on hyvin yksinkertainen ja suoraviivainen. Projekti etenee valmistelun kautta suunnitteluun, josta se siirtyy toteutukseen ja päättyy lopetukseen. Mallissa ei ole mahdollista palata aikaisempaan vaiheeseen.

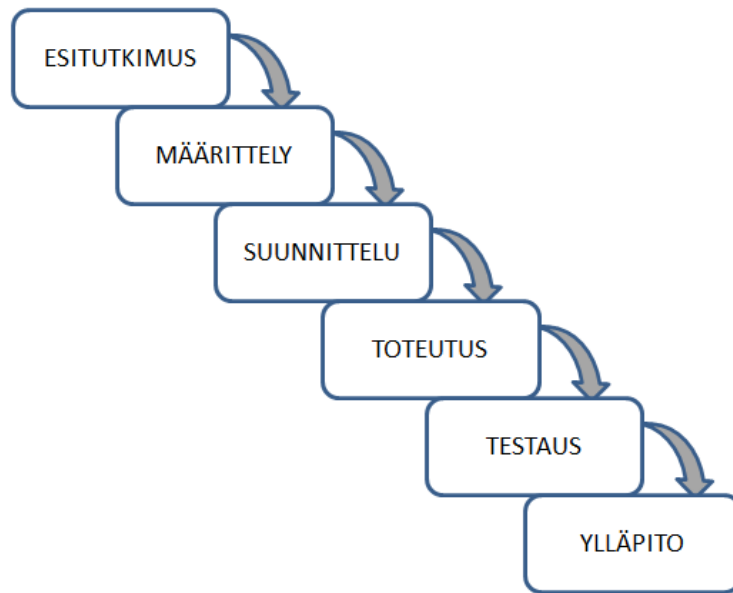


KUVIO 4. Perinteinen projektimalli

Kuviossa 4 on pelkistetty perinteinen projektimalli. Projekti etenee vasemmalta oikealle ja jokaisen kohdan loppuksi arvioidaan, onko tähän mennessä tehdyt suoritteet hyväksyttävällä tasolla ja voidaanko seuraavaan vaiheeseen siirtyä. Lisäksi on hyvä arvioida, onko projektia syytä jatkaa vai onko ilmaantunut sellaisia seikkoja, joiden vuoksi projekti kannattaa lopettaa.

### 2.2.2 Vesiputousmalli

Yksi yleisesti käytetty projektimalli on vesiputousmalli. Jo nimenä vesiputousmalli kuvaa vaiheiden välinen virtaus valuu ylhäältä alaspäin. Vesiputousmallista on tehty erilaisia muunnelmia, mutta peruseriaatteeltaan se on hyvin yksinkertainen.

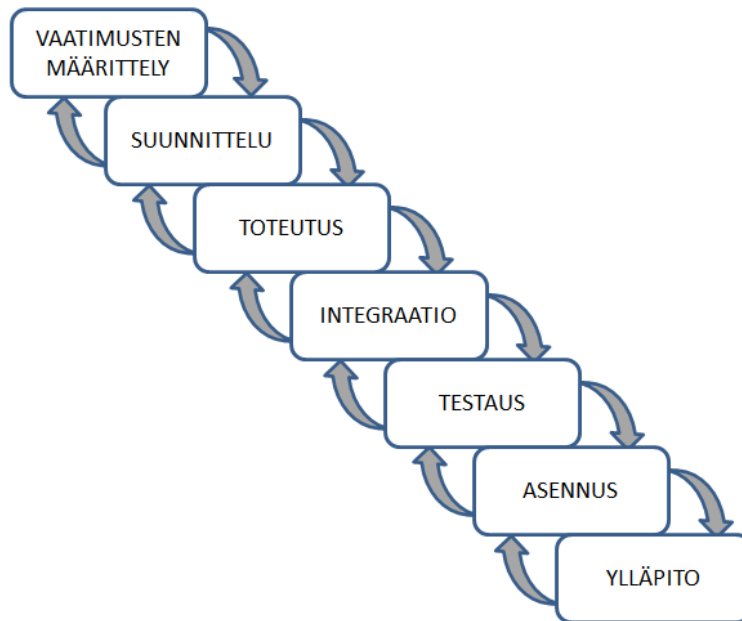


KUVIO 5. Vesiputousmalli

Kuviossa 5 on esitetty yksi esimerkki vesiputousmallista, joka ei eroa kuviossa 4 esitetystä perinteisestä projektimallista. Vesiputousmallin vaatimuksena on se, että yksi vaihe toteutetaan valmiiksi ja arvioidaan tarkastuksin ja katselmuksin ennen kuin aloitetaan seuraava (Tervakari 2008). Edelliseen vaiheeseen palaaminen on yleensä kallista ja joskus jopa mahdotonta.

Käytännössä vesiputousmalli vastaa muiden insinööritieteiden järjestelmäsuunnittelun mallia. Se soveltuu siltojen tekoon, mutta tällaisenaan se ei kelpaa useimpien ohjelmistojen valmistukseen. (Taina 2009, 129.)

Vesiputousmalli korostaa varhaisen suunnittelun merkitystä. Virheiden havaitsemien varhaisessa vaiheessa säästää merkittävästi kustannuksia. (Wikipedia 2015a.)



KUVIO 6. Winston W. Roycen esittelemä vesiputousmalli

Vuonna 1970 Winston W. Royce julkaisi artikkelin *Managing the Development of Large Software Systems*, jossa hän esitteli vesiputousmallin kaltaisen mallin, joka mahdollistaa palaamisen vaihe kerrallaan taaksepäin. Malli on esitetty kuviossa 6. Tämänkaltaisen malli sopii kuvion 5 vesiputousmallia paremmin ohjelmistoprojekteihin, sillä se mahdollistaa iteroinnin (Haikala & Mikkonen 2013, 37).

Vesiputousmallille on ominaista myös kattava dokumentaatio. Jokaisessa vaiheessa dokumentaatio jäädytetään eikä sitä saa muuttaa ilman projektin johtoryhmän lupaa. Dokumenttien jäädyttäminen tarkoittaa käytännössä sitä, että kaikki vaatimukset on kerättävä kerralla, sillä niiden muuttuminen aiheuttaisi muutoksia dokumentteihin. (Taina 2009, 133.)

### 2.2.3 Ketterät menetelmät

Vuonna 2001 joukko ketterien menetelmien puolestapuhujia kirjoitti Ketterän ohjelmistokehityksen julistuksen. Sen mukaan yksilöitä ja kanssakäymistä arvostetaan menetelmiä ja työkaluja enemmän, toimivaa ohjelmistoa arvostetaan enemmän kuin kattavaa dokumentaatiota, asiakasyhteistyötä enemmän kuin

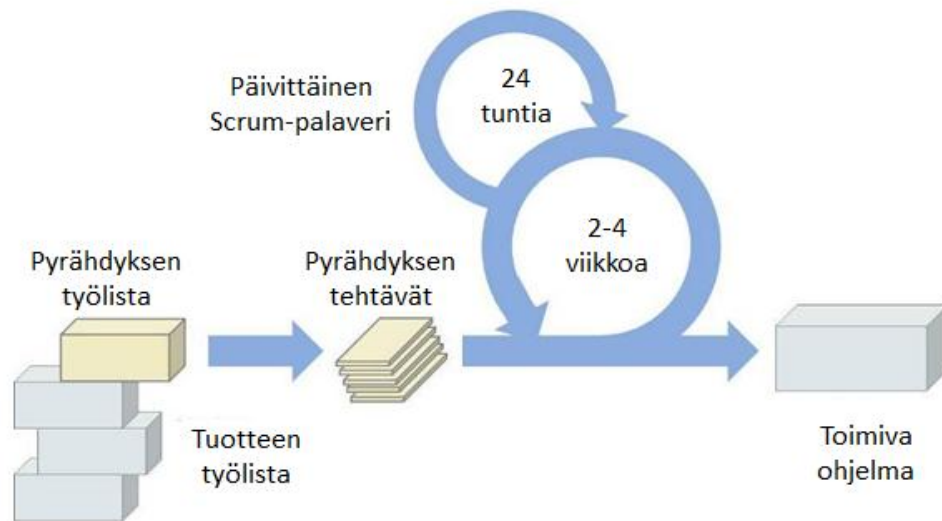
sopimusneuvotteluja sekä muutokseen vastaamista enemmän kuin suunnitelmassa pitäytymistä. (Agilemanifesto 2001a.)

Julistuksen takana on muun muassa periaatteita, joiden mukaan muuttuvia vaatimuksia otetaan vastaan myös kehityksen myöhäisessä vaiheessa ja tekemättä jätettävän työn maksimointi on oleellista (Agilemanifesto 2001b). Ketterien menetelmien ominaispiirteinä on, että ohjelmistosta tuotetaan lyhyin aikavälein toimivia versioita ja näitä toimivia ohjelmistoja pidetään projektin edistymisen mittareina. Juuri nopea reagointi muuttuviin vaatimuksiin on ketterien menetelmien vahvuus. Ketteryyden perusajatus on tuottaa asiakkaan tarpeita paremmin vastaava ohjelmisto nopeammin ja pienemmillä resursseilla. (Auer, Auer, Heinäsmäki, Hölttä, Kalliala, Laanti, Laine, Lekman, Miinalainen, Naski, Piiparinen, Puhakka, Pyhäjärvi, Pääkkönen, Räisänen, Sora, Taipale, Talvio, Tanninen, Toikkanen, Toivola, Toro, Valsta, Väyrynen & von Weissenberg 2013, 43.)

Suurin osa ketteristä menetelmistä jakaa kehityksen muutaman viikon kestäviin iteraatioihin. Ketterin menetelmin pyritään julkaisukelpoiseen ohjelmistoon jokaisen iteraation lopussa siitäkin huolimatta, että julkaisu ei välttämättä ole järkevää. Iteraation lopussa arvioidaan projektin prioriteetit ja päätetään seuraavan iteraation sisällöstä. (Wikipedia 2014a.)

#### 2.2.4 Scrum

Yksi tunnetuimmista projektinhallinnan viitekehyksistä on ketterässä ohjelmistokehityksessä yleisesti käytettävä Scrum. Siinä työtapana on iterative-incremental eli toistuva-lisäävä malli, jossa ohjelmisto kehittyy hiljalleen määrittelynsä mukaiseksi. (Wikipedia 2014b.)



KUVIO 7. Scrum-prosessi

Kuviossa 7 on esitelty Scrum-prosessi. Scrum-prosessi koostuu useista muutaman viikon kestävästä pyrähdysistä, ja jokaisen pyrähdysen lopputuloksena on toimiva tuote. Pyrähdys alkaa suunnittelukokouksella, jossa pyrähdysen työlialle valitaan alkioita tuotteen työlialta ja nämä alkioit pilkotaan pyrähdysen tehtäviksi. Tehtävien kesto on tyypillisesti 4–16 tuntia. Tiimin jäsenet valitsevat itse tehtävät, jotka haluavat tehdä. (Haikala & Mikkonen 2013, 50.)

Tiimi pitää päivittäin Scrum-palaverin (Haikala & Mikkonen 2013, 49). Siinä kaikki tiimin jäsenet vastaavat kolmeen kysymykseen:

- Mitä olet tehnyt edellisen kokouksen jälkeen?
- Mitä aiot tehdä seuraavaksi?
- Onko tiedossa sellaisia esteitä, jotka hidastavat työtäsi?

Palaveriin osallistuu tiimin jäsenten lisäksi Scrum-mestari. (Haikala & Mikkonen 2013, 51.)

Scrumin etuna pidetään yksinkertaisuutta. Sen peruseriaatteet ovat helposti ja nopeasti selitettävissä. Scrumin yksinkertaisuuden takana on se, että se ei ota kantaa ohjelmiston koko elinkaareen liittyviin tehtäviin vaan ennemminkin vain



toteutusvaiheen iteraatioiden organisointiin. Projektin toteuttamiseksi Scrum tarvitsee rinnalleen muun projektinhallinnan välineistön. (Haikala & Mikkonen 2013, 47.)

Scrum-organisaatiomallissa on vain kolme roolia. Yksi niistä on tuotteen omistaja. Tuotteen omistajan tehtävänä on kerätä järjestelmän vaatimukset tuotteen työlistaksi toimimalla rajapintana eri sidosryhmiin. Työlista koostuu alkioista, joilla on arviot toteutuksen kestosta ja liiketoiminta-arvosta. Alkiot voivat olla esimerkiksi käyttötapauksia, käyttäjätarinoita, ominaisuuksia tai vaatimuksia. (Haikala & Mikkonen 2013, 48.)

Toinen rooli on Scrum-mestari, joka vastaa perinteisten projektimallien projektipäällikköä. Suurimpana erona on se, että Scrum-mestarilla ei ole valtaa. Hän voi kuitenkin poistaa tiimiin sopimattoman jäsenen. Tehtäviin kuuluu myös tiimin työtä haittaavien esteiden poistaminen ja tiimin hyvinvoinnista huolehtiminen. Scrum-mestari toimii tuotteen omistajan ja tiimin valmentajana ja vastaa Scrum-prosessin noudattamisesta. Hän myös vastaa pyrähdyn tuloksesta ja varmistaa, ettei tehtäviä merkitä valmiiksi ennen kuin ne täyttävät kaikki niille määritetyt ehdot. (Haikala & Mikkonen 2013, 49.)

Kolmas rooli on tiimi. Optimaalinen Scrum-tiimi koostuu seitsemästä eritaustaisesta kokopäiväisestä henkilöstä. Joukossa voisi olla ohjelmoijia, käyttöliittymäsuunnittelijoita ja testaajia. Scrum-tiimi on itseohjautuva, eikä sillä ole projektipäällikköä. Tiimin tulisi toimia samassa tilassa, ja siellä tulisi olla tehtävätaulu, josta voi seurata tehtävien etenemistä. Tiimi itse päättää työskentelykäytännöistään. (Haikala & Mikkonen 2013, 49.)

### 2.3 Tyypillisiä ongelmia projekteissa

Vuoden 2013 IT-barometri kertoo, että yli puolet IT-projekteista ei pysynyt aikataulussaan ja ylittivät budjetin. Kaksi kolmasosaa projekteista saavuttaa niille asetetut tavoitteet. Kolme neljäsosaa projekteista vastaa lopputulokseltaan suunniteltua. (Tietotekniikan liitto ry, 26.)

Standish Group -niminen IT-alan tutkimusyriety on 2000-luvun alkupuolella selvittänyt, että 31 prosenttia projekteista peruutetaan ennen valmistumista ja 88

prosenttia ylittävät aikataulun tai budjetin. Aikataulun ylitys on keskimäärin 222 prosenttia ja kustannusten ylitys keskimäärin 189 prosenttia. (Phillips 2005, 11.) Aikataulujen puutteellisuus tai niihin liian helposti tehtävät muutokset aiheuttavat projektin venymistä. Liian tiukat aikataulut antavat huonot lähtökohdat projektin onnistuneeseen toteuttamiseen. (MAMK 2013, 60.)

Tyypillinen ongelma projektityöskentelyssä on puutteellisesti tehty määrittely. Projekteja aloitetaan liian helposti eikä kustannusten vaikutusta hyötyyn selvitetä riittävän perusteellisesti. Kunnollisella määrittelyllä voidaan karsia kannattamattomia projekteja pois. (MAMK 2013, 53.)

Huolellisella määrittelyllä projektin tavoitteet ja rajaukset ovat selkeät. Silloin ei ole tarvetta jatkuville muutosehdotuksille. Muutokset vaikuttavat aina kustannuksiin ja sen vuoksi niiden jälkeen pitää kustannukset arvioida uudestaan. (MAMK 2013, 55.)

Ongelmia liittyy myös henkilöstöön tai henkilöstön johtamiseen. Projektin jäsenet eivät välttämättä ole riittävän osaavia tai he eivät ole muilta osin oikeita henkilöitä projektiryhmään. Projektia saatetaan toteuttaa oman työn ohessa, jolloin omistautuminen projektille heikkenee. Projektiryhmä kannattaa pitää pienenä ja kokoaikaisena mieluummin kuin suurena ja osa-aikaisena. (MAMK 2013, 56.)

Projektin riskeihin pitäisi varautua. Riskit tulisi osata arvioida ja tunnistaa suunnitteluvaiheessa. (MAMK 2013, 59.) Apuna voi käyttää esimerkiksi SWOT-analyysiä (Lukkari 204, 33). SWOT on Albert Humbreyn kehittämä nelikenttämenetelmä, jossa arvioidaan organisaation, sen osan, tuotteen, palvelun tai esimerkiksi juuri projektin sisäiset vahvuudet (Strengths), sisäiset heikkoudet (Weaknesses), ulkoiset mahdollisuudet (Opportunities) ja ulkoiset uhat (Threats). SWOT-analyysin pohjalta voidaan tehdä päätelmiä, miten vahvuuksia voidaan hyödyntää, miten heikkoudet saadaan käännettyä vahvuuksiksi, miten tulevaisuuden mahdollisuuksia voidaan hyödyntää ja miten mahdollisia uhkia voidaan välttää ja ennakoita ja näiden pohjalta voidaan saada aikaan toimintasuunnitelma. (Wikipedia 2015.)

SWOT on yleisesti käytetty menetelmä strategisessa suunnittelussa ja sen avulla voidaan määrittää omaa asemaansa ulkomaailmaan ja itseensä nähden.

Menetelmän etuna on sen helppo toteutettavuus eikä se kuluta merkittävästi resursseja. Kuitenkin ohjeistus menetelmän käyttöön pitää olla tarkka, jotta osallistujilta saadaan vastaukset oikeisiin kysymyksiin. (OK-opintokeskus 2015.)

<p><b>VAHVUUDET</b></p> <ul style="list-style-type: none"> <li>• Aikaisemmat referenssit samankaltaisista projekteista</li> <li>• Henkilöstö</li> <li>• Teknologian hyödyntäminen</li> <li>• Tieto-taito</li> <li>• Tekninen osaaminen</li> <li>• Erikoistuminen ja omaan osaamiseen keskittyminen</li> </ul>	<p><b>HEIKKOUEDET</b></p> <ul style="list-style-type: none"> <li>• Vajausta resursseissa</li> <li>• Odotukset ristiriitaisia projektiryhmän sisällä</li> <li>• Projektin kulurakenne ja rahaliikenteen kiertokulku</li> <li>• Henkilöstön vaihtuvuus</li> <li>• Aikataulun mitoitus</li> </ul>
<p><b>MAHDOLLISUUDET</b></p> <ul style="list-style-type: none"> <li>• Maailman raaka-ainehintojen muuttuminen (kupari, öljy)</li> <li>• Toimialan muuttuminen</li> <li>• Asiakkaiden hankintajärjestelmien muuttuminen</li> </ul>	<p><b>UHAT</b></p> <ul style="list-style-type: none"> <li>• Taloustilanteen heikkeneminen</li> <li>• Asiakasyrityksessä käynnissä olevat muutokset</li> <li>• Maailman raaka-ainehintojen muuttuminen (kupari, öljy)</li> <li>• Toimialan muuttuminen</li> </ul>

KUVIO 8. Esimerkki SWOT-mallin mukaisesta riskianalyysistä

Kuvio 8 esittelee arvioitavat asiat. Tämänkaltainen pohdiskelu saattaa paljastaa jo aikaisessa vaiheessa, jos projektilla ei ole edellytyksiä valmistua. Yhtä lailla hyvin tehty analyysi kertoo etukäteen ne asiat, joihin pitää varautua erityisellä huolellisuudella. (Silfverberg 2015, 15.) Mahdollisuuksissa ja uhissa on tarkoituksella samoja asioita, sillä siinä missä esimerkiksi maailman markkinahintojen muutokset voivat olla mahdollisuus, ne voivat olla myös suuri uhka projektille.

## 2.4 Riskien hallinta

Projektinhallinnassa riskillä tarkoitetaan jotain, joka saattaa projektissa muodostua ongelmaksi. Siihen liittyy kaksi keskeistä tekijää: todennäköisyys ja vakavuus. Jos riskin todennäköisyys on 100 %, ei kyseessä ole riski, vaan tosiasia. Riskille ominaista on, että toteutuessaan se aiheuttaa projektille menetyksiä tai mahdollisesti projektin epäonnistumisen. (Haikala & Mikkonen 2013, 164.)

Riskienhallinnan tavoitteena on tunnistaa kaikki riskit ja pyrkiä minimoimaan niiden vaikutus. Muodollisen ja toistettavan hallintaprosessin avulla riskit voidaan tunnistaa ja arvioida. Lisäksi sen avulla voidaan määrittää tehokkaat riskienvähennystoimet sekä seurata ja raportoida riskienhallinnan onnistumista ja edistymistä. (Murch 2002, 163.)

Yleisellä tasolla riskit voidaan jakaa viiteen luokkaan. Ulkoisilla riskeillä tarkoitetaan esimerkiksi muutoksia markkinoilla tai viranomaissäätelystä, luonnonilmiöitä tai ulkoisia häiriöitä esimerkiksi sähkönjakelussa. Näihin on projektissa vaikea puuttua, mutta ne voidaan kuitenkin tunnistaa ja siten niiden vaikutuksia voidaan myös hallita. (Murch 2002, 163.)

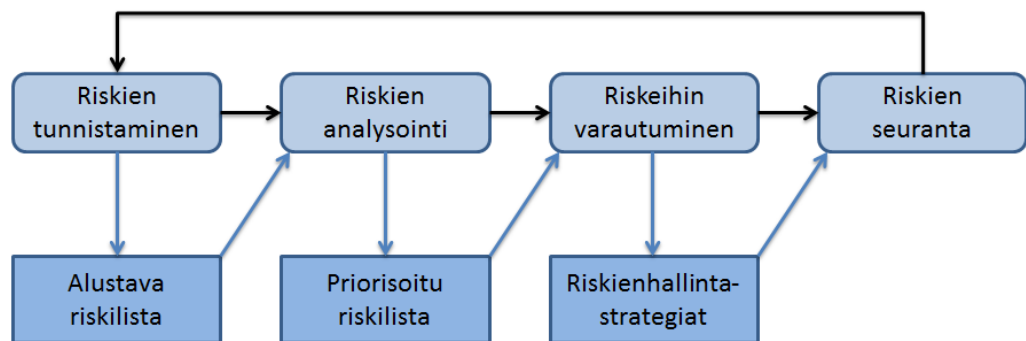
Kustannusriskeillä ymmärretään budjetin ja aikataulun ylitykset, kustannusarvioinnissa tehdyistä virheistä johtuvat ylitykset ja tuotteen tai projektin laajuuden kasvu ja hallitsemattomat muutokset. Kustannusriskit ovat projektipäällikön valvonnassa, ja hänellä on mahdollisuus vaikuttaa niihin. (Murch 2002, 164.)

Aikatauluriskeihin kuuluu epäonnistuneen osittamisen tai aikataulutuksen takia syntyvät viivästykset, odottamattomien ongelmien ratkaisuun kuluva aika sekä resurssimuutokset. Aikatauluriskit voivat johtaa projektin epäonnistumiseen, tai ne voivat aiheuttaa taloudellista vahinkoa. (Murch 2002, 164.)

Väärin työkalujen, epäkypsen tekniikan tai ohjelmistojen käyttö sekä integraatio- ja laitteiston suorituskyvyn ongelmat luetaan tekniikkariskeihin. Niiden seurauksena esimerkiksi kehitettävän järjestelmän suorituskyky tai toiminnallisuus ei vastaa odotuksia. Toiminnan riskeihin, joille on tyypillistä, että muutoksia ei kyetä toteuttamaan tehokkaasti, kuuluu puolestaan muun muassa

riittämätön viestintä, puutteellinen ongelmien ratkaiseminen ja prioriteettien määrittäminen sekä toteutuksen riskit (Murch 2002, 165).

Ohjelmistoprojekteissa eräs hyvin toimiva tapa on luokitella riskit projektiriskeihin, tuoteriskeihin ja liiketoimintariskeihin. Projektirisikin toteutuessa aikataulut ovat vaarassa viivästyä tai jopa koko projektin valmistuminen vaarantuu. Tuoteriskeillä tarkoitetaan sellaisia riskejä, jotka toteutuessaan vaikuttavat tuotteen laatuun. Jokin tuotteen osa saattaa aiheuttaa sen, että koko ohjelmisto ei toimi halutulla tavalla tai nopeudella. Liiketoimintariskeihin luetaan riskit, joiden toteutuessa markkinoilla tapahtuu radikaaleja muutoksia ja projektin budjetoitu tuotto jää merkittävästi pienemmäksi. Kilpailija saattaa ehtiä saada valmiiksi kilpailevan tuotteen ennen omaa tuotetta tai tekee sen paremmin. (Haikala & Mikkonen 2013, 165.)



KUVIO 9. Riskienhallintaprosessi

Kuvio 9 hahmottelee riskienhallintaprosessia. Riskit tunnistetaan kirjaamalla ne alustavaan riskilistaan. Sen jälkeen riskit analysoidaan. Riskiin liitetään sen todennäköisyys ja seurauksen vakavuus. Samalla voidaan miettiä riskin ennusmerkkejä ja mahdollisia seurauksia sidosryhmien kannalta. Analyysin jälkeen riskiin varaudutaan: voidaanko riskiltä välttyä kokonaan tai miten sen seuraukset voidaan minimoida. (Haikala & Mikkonen 2013, 166.)

Ketteriin menetelmiin on osittain sisäänrakennettuna riskienhallintaa. Usein toistuvissa palavereissa riskejä seurataan tiheästi ja ne paljastuvat helposti. Varsinaisen riskienhallintasuunnitelman tarpeellisuutta ketterät menetelmät eivät kuitenkaan poista. (Haikala & Mikkonen 2013, 166.)

### 3 OHJELMISTOPROJEKTIT

Pohjimmiltaan myös ohjelmistoprojektit, kuten kaikki muutkin projektit, ovat liiketaloudellisia projekteja. Vaikka kaikki projektit eivät tähtää suoraan tuottavuuden parantamiseen, niissä raha vaihtaa jollain lailla omistajaa.

Ohjelmistoprojekteille, kuten ohjelmistonkehitys- ja käyttöönottoprojekteille, on tyypillistä jatkuvasti muuttuvat vaatimukset. Tämän vuoksi vähänkään isompaan ohjelmistoprojektiin perinteiset projekti- tai vesiputousmallit eivät ole soveliaita. Muutosten kustannukset ovat liian suuria, ja muutokset aiheuttavat aikataulun venymistä. Kuitenkin vesiputousmallin teoria tunnetaan hyvin ja sille on saatavissa laajalti työkaluja, joten vesiputousmallin variantteja, joissa sallitaan iteratiivisuutta, käytetään suurimmassa osassa ohjelmistoyrityksistä. (Taina 2009, 130–134.)

Yksi onnistuneen ohjelmistoprojektin perusedellytyksistä on kunnolla tehty vaatimusten käsittely. Tämän ovat todenneet useat asiantuntijat. Monien tutkimusten mukaan yli 60-prosenttisesti syyt epäonnistuneeseen ohjelmistoprojektiin ovat huonossa vaatimusten käsittelyssä (Haikala & Mikkonen 2013, 61. mukaan).

Ennen projektin aloittamista on tunnistettu tarve, jonka seurauksena on syntynyt projektialoite tai projektiehdotus. Aloite on saatettu tehdä vain suullisena toteamuksena siitä, että jonkinlaiseen ongelmaan pitäisi löytää ratkaisu.

Joskus on helpompaa, varsinkin suurten projektien kohdalla, jakaa projekti osaprojekteiksi. Tällaisia osaprojekteja voivat olla esimerkiksi esitutkimus, määrittelyprojekti ja toteutusprojekti. (Haikala & Mikkonen 2013, 21.)

#### 3.1 Valmistelu

Valmistelun tavoitteena on kiinnittää projektin tavoitteet ja resurssit. Valmistelu tehdään yleensä kyseisen projektin tulevan projektipäällikön toimesta.

Valmistelun tuloksena saadaan projektiesitys ja alustava projektisuunnitelma, josta selviää vähintään projektin nimi, asiakas, tavoite, alkamis- ja päättymisajat, projektipäällikkö, henkilöresurssit ja budjetti. (Helsingin yliopisto 2014.)

Valmisteluvaiheessa projektiehdotus täydentyy projektikuvaukseksi. Jos valmisteluvaiheen lopussa projektikuvaus hyväksytään, niin siirrytään suunnitteluvaiheeseen. Viimeistään tässä vaiheessa tulee nimetä projektipäällikkö. (Haikala & Mikkonen 2013, 35.)

### 3.2 Suunnittelu

Onnistunut projektisuunnittelu edellyttää toimialan ja menetelmien vahvaa tuntemusta sekä projektiosaamista. Hyvät projektisuunnitelmat ovat tunnistettavissa siitä, että niissä työ on jaettu osiin, joita on helppo hallita. Myös aikataulussa ja budjetissa on varauduttu yllättäviin tilanteisiin. Ne ovat ajantasaisia ja realistisia. Niitä myös tarkistetaan jatkuvasti. (Murch 2002, 27.)

Suunnitteluvaiheessa tuotetaan projektisuunnitelma, jota seuraamalla saavutetaan määritetyt tavoitteet. Projektisuunnitelmasta selviää seuraavat asiat:

- projektin taustat ja lähtökohdat
- tavoitteet ja rajaukset
- tehtävien määrittely ja täsmentäminen
- työmäärän arviointi
- projektiorganisaation ja roolien suunnittelu sekä tehtävien jakaminen
- kustannusten arviointi ja budjetointi
- projektin tuotoksen arvioinnin perusteet
- riskien tunnistaminen, analysointi ja varautuminen.

Projektin suunnittelun, ohjauksen ja seurannan apuvälineenä käytetään projektisuunnitelmaa. Projektisuunnitelman avulla välitetään tietoa eri sidosryhmien välillä. Samalla se toimii eräänlaisena projektisopimuksena. (Haikala & Mikkonen 2013, 166.)

Myös pienistä projekteista on syytä tehdä edes jonkunlainen muistion tapainen projektisuunnitelma, josta voidaan tarkistaa mistä on sovittu (Haikala & Mikkonen 2013, 167). Lisäksi ne ovat keino saavuttaa projektille asetetut tavoitteet ja vaihetuotteet. (Murch 2002, 41.)

### 3.2.1 Projektin osittaminen ja aikataulutus

Projektin osittaminen on yksi perinteisen projektisuunnittelun tärkeimmistä ja keskeisimmistä tehtävistä. Osittamisessa projekti jaetaan sopiviin osiin. Osan pituus voi olla esimerkiksi 2 - 3 päivää. Jaottelu tehdään usein hierarkkisesti. Tehtäväkokonaisuudet jaetaan osatehtäviksi. Liian suurten, esimerkiksi viikkojen pituisten tehtävien todellista aikaa on vaikea arvioida. Vaarana on, että poikkeamia ei huomata ajoissa. Pieniä tehtäviä tulee lukumäärällisesti paljon, ja se aiheuttaa lisääntyntä seuranta ja dokumentointia. Pieniä tehtäviä voidaan yhdistellä isommaksi tehtäväksi. Osittamisen jälkeen työt kohdennetaan tietyn henkilön tehtäväksi ja sijoitetaan kalenteriin. (Haikala & Mikkonen 2013, 157–159).

Ohjelmistoprojektille tyypilliset jatkuvasti muuttuvat vaatimukset hankaloittavat projektin osittamista kovin pitkälle tulevaisuuteen. Pitkissä projekteissa ei ole tarkoituksen mukaista osittaa aluksi koko projektia kovinkaan tarkasti, vaan ositusta tarkennetaan työn edetessä. Ositusta on tarkastettava aikataulutuksen ja tehtävien osalta säännöllisesti kuukauden tai parin välein (Haikala & Mikkonen 2013, 159).

Osittamisessa on syytä huomioida, että henkilön viikkotyömäärä ei ole kokonaisuudessaan käytettävissä projektiin. Viikkotyöajasta on tehtävä realistiset vähennykset henkilön muuhun kuluvaan työajasta, kuten muut työt, matka-aika, kokoukset, projektitarkastukset ja arkipyhät. Yleisesti arvioidaan, että ohjelmistosuunnittelijalla on tehokasta työaikaa käytettävissään 60 - 70 prosenttia viikkotyöajasta. (Haikala & Mikkonen 2013, 161.)

Ketterissä menetelmissä ositusta ja aikataulutusta tehdään vain lähitulevaisuudessa toteutettaville tehtäville. Ketterien menetelmien tehtävät jaetaan vielä pienemmiksi kokonaisuuksiksi kuin perinteisin menetelmin toteutettavissa projekteissa. Tehtävän kooksi suositellaan 4 - 16 tuntia. (Haikala & Mikkonen 2013, 163.)



### 3.2.2 Ganttin kaavio

1800-luvulla amerikkalaisen keksijän ja insinöörin Henry Ganttin kehittämä Ganttin kaavio on projektin hallinnassa suosittu työkalu. Kaaviosta nähdään projektin ja sen työvaiheiden edistyminen suhteessa aikaan. Vaakasuuntainen akseli on aikajana. Tehtävät on sijoitettu allekkain vasempaan reunaan. Jokaisen tehtävän kohdalle merkitään varaus aikajanelle sille kohdalle, jolloin tehtävä on tarkoitus suorittaa. Aikavarauspalkin värityksen avulla voidaan seurata yksittäisen tehtävän valmiusastetta. Kaaviosta on myös helppo nähdä tehtävien keskinäinen toteutusjärjestys; mitä tehtäviä voidaan suorittaa rinnakkain ja minkä tehtävien loppuun saattaminen on edellytys seuraavien tehtävien aloittamiselle. (Wikipedia 2015b.) Kaaviossa voidaan myös seurata etenemistä kriittisellä polulla. Kriittisellä polulla tarkoitetaan joukkoa tehtäviä, jotka ovat toisistaan riippuvaisia ja jotka vaikuttavat projektin suunniteltuihin päivämääriin. Jos kriittisellä polulla oleva tehtävä myöhästyy, uhkaa koko projektia myöhästyminen. (Microsoft 2015.)

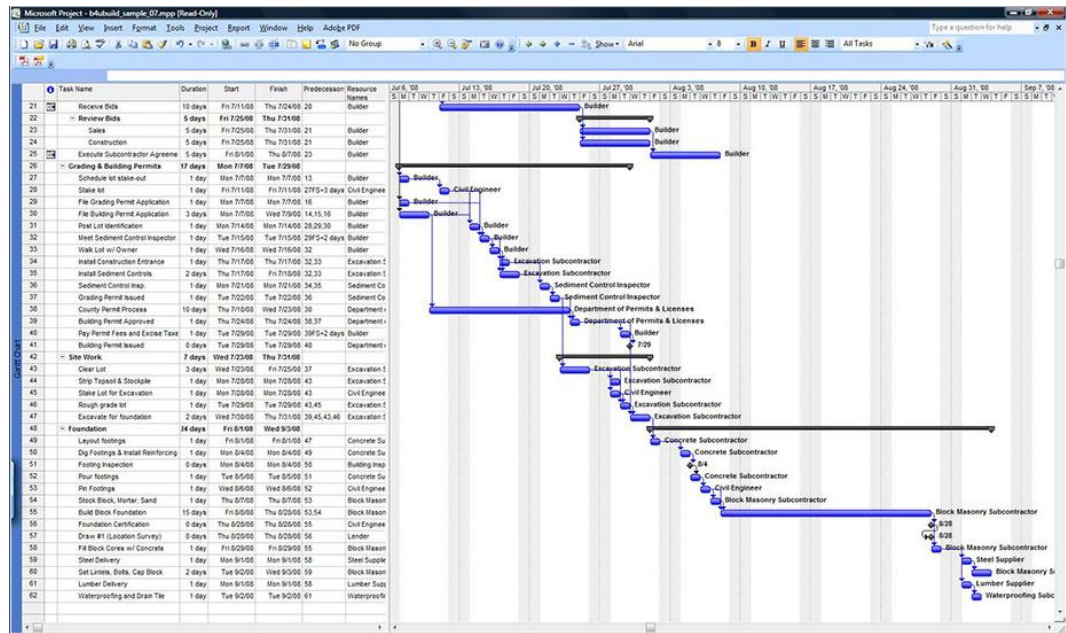
Ganttin kaavion etuna pidetään sen visuaalisuutta. Siitä on helppo hahmottaa kokonaisuus. Kaavion avulla on helppo pitää projektin osalliset ajan tasalla projektin etenemisestä. Ganttin kaavio on myös helppolukuinen raportointiväline. Kaavion heikkoutena pidetään sen yksinkertaisuutta. Se ei ota kantaa työvaiheen vaativuuteen liittyviä asioita. Kaaviossa ei myöskään välttämättä ole helposti tunnistettavissa työvaiheiden välisiä riippuvuuksia. (Wikipedia 2015b).

Tehtävä	Kesto	Viikko 1	Viikko 2	Viikko 3	Viikko 4	Viikko 5
Tehtävä 1	1	■				
Tehtävä 2	5	■ ■ ■ ■ ■				
Tehtävä 3	2		■ ■			
Tehtävä 4	10		■ ■ ■ ■ ■ ■ ■ ■ ■ ■			
Tehtävä 5	2		■ ■			
Tehtävä 6	3			■ ■ ■		
Tehtävä 7	2				■ ■	
Tehtävä 8	1					■
Tehtävä 9	2					■ ■
Tehtävä 10	1					■
Kesto yht.	29	Tehty	19	Tekemättä	10	
Päiviä yht.	25	Päiviä kulunut	14	Päiviä jäljellä	11	0,91

## KUVIO 10. Ganttin kaavio

Kuviossa 10 on esimerkki Ganttin kaaviosta ja sen soveltamisesta. Aikajanalla jokainen ruutu vastaa yhtä työpäivää ja punainen kursori osoittaa, missä kohtaa aikajanaa tällä hetkellä ollaan. Tehtävät on sijoitettu aikajanalle vihrein ruuduin. Tummanvihreät ruudut kuvaavat tehtyä osuutta tehtävästä ja vaalean vihreät tekemätöntä osuutta. Keltaiset nuolet on lisätty Ganttin kaavioon esittämään tehtävien välisiä riippuvuuksia.

Kuviosta havaitaan, että tehtävä 4 on kolme päivää aikataulusta jäljessä. Toisaalta projekti näyttää kaaviossa alkupainotteiselta ja tehtäviä on suoritettu rinnakkain. Projektin lopussa on vielä neljä tyhjää päivää, joilla on varauduttu projektin viivästymiseen. Oikeassa alakulmassa oleva suhdeluku kertoo jäljellä olevan työn määrän suhteessa jäljellä olevaan aikaan. Toisin sanoen alle yhden oleva luku kertoo, että jäljellä olevaa työtä on vähemmän kuin aikaa eli projekti on aikatauluaan edellä.



## KUVIO 11. Ganttin kaavio MS Projectilla tehtynä (b4ubuild.com 2015)

Kuviossa 11 on monimutkaisempi MS Projectilla tehty Ganttin kaavio. Mustat palkit osoittavat päätehtävälle varatun kokonaisajan ja siniset palkit alitehtävälle resursoidun ajan. Kuvion 10 esimerkistä poiketen tässä kaaviossa on tehtävien

välillä ”yhden suhde moneen” -riippuvuuksia. Todellisuudessa suhde voi olla myös ”yhden suhde yhteen” tai ”monen suhde moneen”.

### 3.2.3 Projektisuunnittelun vaihetuotteet

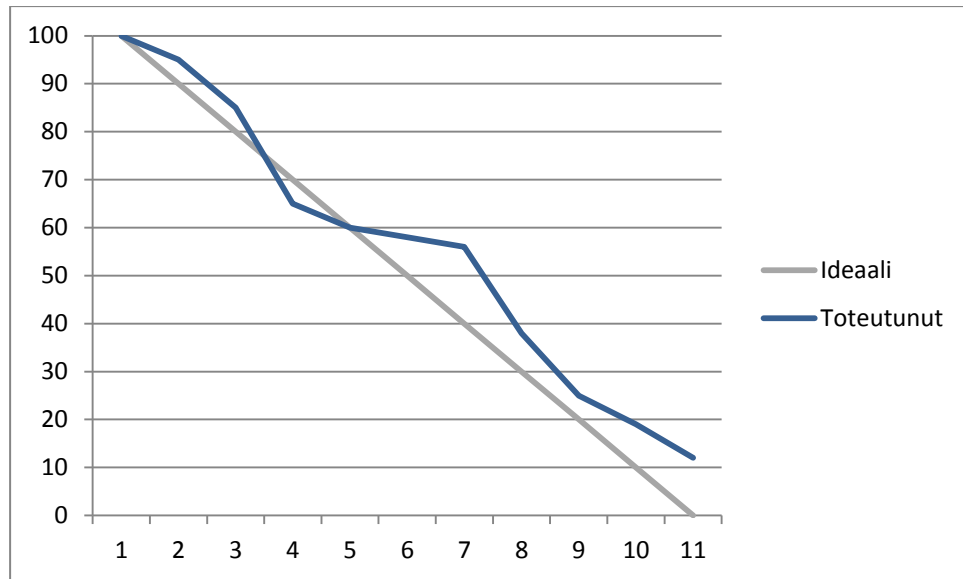
Huonosti määritellyt vaihetuotteet antavat projektille huonot lähtökohdat. Huonosti määritellyissä vaihetuotteissa liikutaan ainoastaan käsitetasolla eikä muisteta mitä hyötyjä projektin avulla saavutetaan. Sen sijaan keskitytään pelkkään prosessiin unohtaen päämäärät. Menestyksellisiä projekteja yhdistää se, että sekä käyttäjät että organisaatio ymmärtävät ja ovat yhtä mieltä siitä, mitkä ovat projektin vaihetuotteet ja niiden hyödyt. (Murch 2002, 28.)

Projektisuunnittelun aikana syntyy vaihetuotteina muun muassa projektin organisaatiokaavio, projektin välietapit, projektistandardit, suunnitteluasiakirjat, kuten resurssisuunnitelma, projektin aikataulu, projektinhallintasuunnitelma ja toteutussuunnitelma sekä riski- ja laaturaportit sekä projektin loppuraportti (Murch 2002, 29).

### 3.3 Toteutus

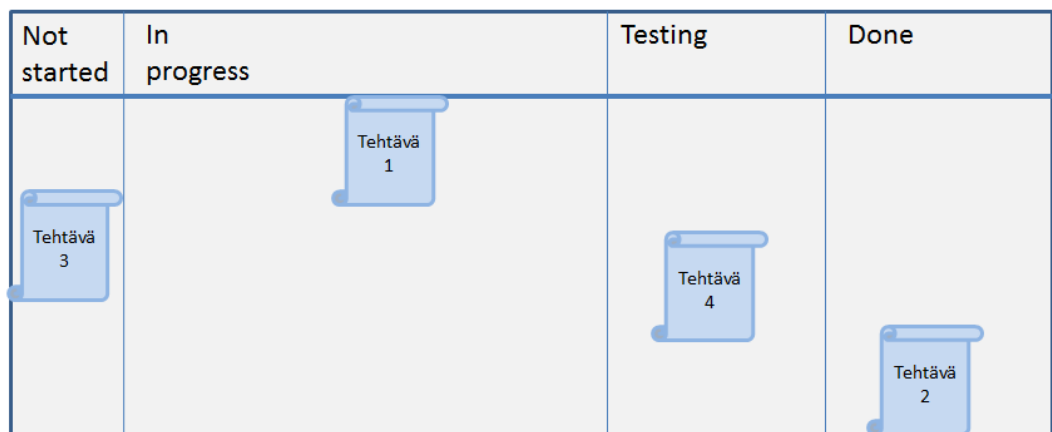
Projektin toteutuksen aikana kaikkiin suunnitelmiin kohdistuu jonkinlaisia vaikeuksia, poikkeuksia tai odottamattomia muutoksia, mikä ei ole mitenkään normaalista poikkeavaa. Jos muutoksia ilmenee paljon, syy voi olla projektisuunnitelmassa, joka ei ollut tarpeeksi perusteellinen. Tarkat suunnitelmat vaativat, että niitä pidetään ajan tasalla. (Murch 2002, 41.)

Toteutusvaiheen aikana yksi tärkeimmistä tehtävistä on raportointi projektin edistymisestä (Murch 2002, 32). Seuranta tehdään ohjausryhmän palaverissa, joissa työn edistymistä verrataan projektisuunnitelmaan. Poikkeamat tulisi huomata mahdollisimman pian, jolloin korjauksiin jää mahdollisimman paljon aikaa. (Haikala & Mikkonen 2013, 164.)



KUVIO 12. Projektin edistyminen ajan funktiona

Kuviossa 12 on esitetty ketterissä menetelmissä käytettävä edistymiskäyrä. Harmaa viiva esittää ideaalia lineaarista edistymistä ja sininen viiva toteutunutta edistymistä. Y-akselilla on jäljellä oleva työmäärä prosentteina ja x-akselilla kuluneet työpäivät. Kuvioista näkee helposti, ollaanko projektissa tai pyrähdyksessä aikataulussa – mitä lähempänä sininen ja harmaa viiva ovat toisiaan, sitä lähempänä ollaan ideaalia edistymistä. (Haikala & Mikkonen 2013, 164.)



KUVIO 13. Scrumin tehtävätaulu

Kuvio 13 esittää Scrumissa käytettävää tehtävätaulua, jota pidetään yhteisessä työtilassa. Tämän kaltaista taulua voi suositella myös muilla malleilla hallittaviin projekteihin. (Haikala & Mikkonen 2013, 164.) Tehtävätaulussa käyttäjätarinat

liikkuvat vaakatasossa vaiheen kerrallaan. Näitä tehtävätauluja voidaan myös korvata soveltuvilla ohjelmistoilla, kuten Agilefant tai Jira (Leppäniemi 2015).

Projektin aikana tulisi pitää säännöllisesti ”mitä tästä opimme” -palavereja. Niiden tarkoituksena on kehitellä uusia työtapoja ja ideoita tulevaisuutta ajatellen. (Murch 2002, 28.) Toteutusvaihe päätetään, kun havaitaan, että asetetut tulostavoitteet on saavutettu (Rannisto 2015).

### 3.4 Testaus

Ohjelmistojen testauksen avulla ei voida osoittaa järjestelmän virheettömyyttä, vaan testauksen tarkoitus on osoittaa, että ohjelmassa on virheitä (Haikala & Mikkonen 2013, 205). Se, että ohjelmasta löytyy paljon virheitä, ei välttämättä tarkoita, että ohjelma olisi erityisen huonosti tehty, vaan se voi olla merkki myös kunnollisesta testauksesta (Haikala & Mikkonen 2013, 143). Kunnollinen testaus vie yli puolet ohjelmistoprojektin resursseista, joten sen läpivientiin kannattaa kiinnittää huomiota (Haikala & Mikkonen 2013, 205).

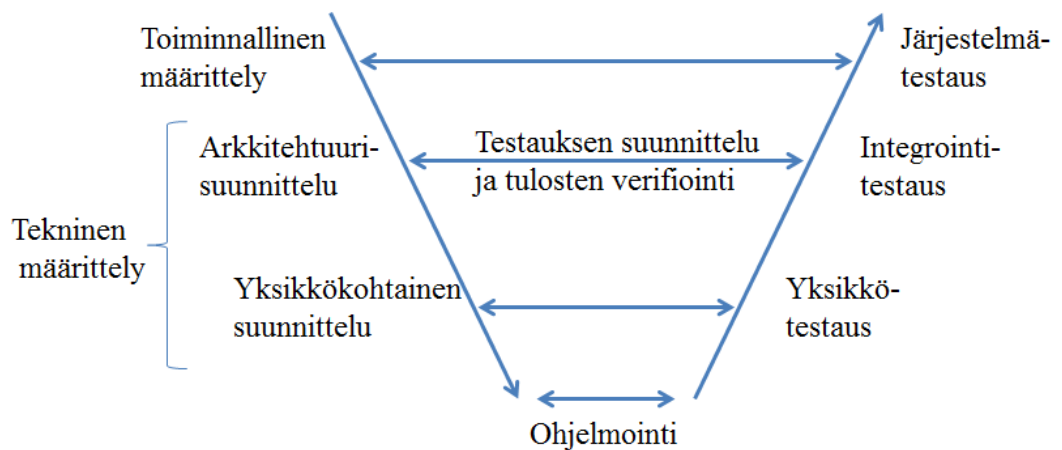
Vaikka testaamalla voidaan käytännössä kattaa vain erittäin pieni osa kaikista ohjelman toiminnoista, ei se tarkoita sitä, ettei testaukseen kannattaisi panostaa. Ennemminkin kannattaa suhtautua varauksella ohjelmiston toimivuuteen hyvistä testaustuloksista huolimatta. (Haikala & Mikkonen 2013, 205.)

Ohjelmistojen testauksessa virheellä tarkoitetaan poikkeamaa määrittelyssä. Ilman määrittelyä ei johdonmukaista testausta voida tehdä, koska lopputuloksen oikeellisuutta ei voida todentaa. Toiminnallinen määrittely ja tekninen määrittely ovat testauksessa yleisimmin käytettävät määrittelydokumentit. Näissä dokumenteissa voi tulkinnan suhteen esiintyä mielipide-eroja. Se, mikä asiakkaan mielestä on virhe, on toimittajan mielestä ominaisuus. (Haikala & Mikkonen 2013, 206.)

Yleisesti on suositeltavaa, että ohjelmiston toteuttajat eivät osallistu testiin. Ulkopuoliset testaajat löytävät enemmän virheitä, koska ohjelmistojen toteuttajat keskittyvät syystä tai toisesta testaamaan niitä ohjelmistojen osia, jotka tiedetään toimiviksi. Käytäntö on kuitenkin osoittanut, että ohjelman kehittäjät osallistuvat yksikkötestaukseen ja joskus myös

integroititestaukseen. Vasta järjestelmätestausvaiheessa ulkopuoliset testaajat pääsevät osallistumaan testaamiseen. (Haikala & Mikkonen 2013, 209.)

Testausprosessissa käytettävä V-malli on varsin yleinen prosessinmallinnuskeino. Testausprosessien lisäksi sitä käytetään myös monissa muissa prosesseissa. Testaus voidaan jakaa osiin, jotka eivät ole riippuvaisia toisistaan. (Myyry 2002, 2.)

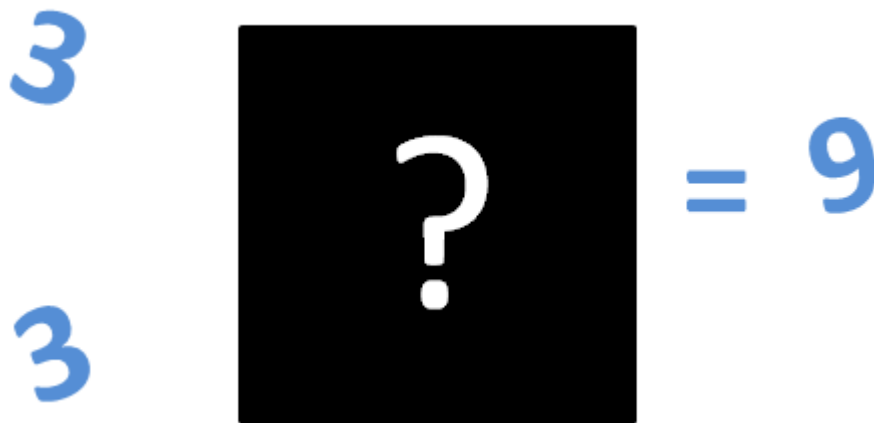


KUVIO 14. Testauksen V-malli

Kuviossa 14 havainnollistetaan V-mallin mukaista kehitystyön ja testauksen suhdetta. Mallin mukaisesti testauksen suunnittelu tapahtuu sen tasoa vastaavalla suunnittelutasolla. Kunkin tason tulokset todetaan oikeaksi vertaamalla niitä vastaaviin sen tason vastaaviin dokumentteihin. (Haikala & Mikkonen 2013, 206.)

Luonnontieteessä ja tekniikassa mustalla laatikolla tarkoitetaan laitetta, oliota tai järjestelmää, jonka sisällöstä ei ole tietoa. Sitä voidaan tarkastella ainoastaan syöttö-, tuloste- ja siirto-ominaisuuksien perusteella. Ohjelmistotuotannossa mustalaatikkotestauksella voidaan kokeilla, saadaanko annetuilla syötteillä odotetun kaltaisia tuloksia. (Wikipedia 2013.)

Mustalaatikkotestauksessa syöteavaruus jaetaan ekvivalenssiluokkiin. Luokat valitaan siten, että testattavan yksikön voidaan olettaa toimivan millä tahansa ekvivalenssiluokan edustajalla, jos se toimii yhdellä kyseisen luokan edustajalla. (Haikala & Mikkonen 2013, 209.)



KUVIO 15. Mustalaatikkotestaus

Kuviossa 15 testattavalle yksikölle annetaan syötetietona kaksi muuttujaa, jotka molemmat ovat arvoltaan 3 ja yksikkö palauttaa tuloksen 9. Jos yksikön palauttama tulos on odotettu, voidaan yksikön olettaa toimivan oikein, vaikka sen toimintaa ei tunneta. Tiedossa ei ole, suorittaako yksikkö kertolaskun  $A*B$  vai yhteenlaskun  $A+A$ , joka toistetaan  $B$  kertaa.

Lasilaatikkotestaus on menetelmä, jossa testattavan yksikön rakenne tunnetaan. Testaajan on tunnettava ohjelmiston toimintaperiaatteet, ja häneltä vaaditaan ohjelmointitaitoja, jotta hän voi tunnistaa ohjelmakoodissa esiintyvät ohjelmapolut. Lasilaatikkotestauksessa valitaan erilaisia syötteitä ja ohjelmakoodin perusteella päätellään ohjelman sisäinen kulku ja edelleen oikeat ulostulot eri syötteille. (Jokela 2007, 6.)

Edellä mainittujen musta- ja lasilaatikkolähestymistapojen lisäksi on olemassa näiden yhdistelmä eli harmaalaatikkotestaus. Mustalaatikkotestauksella ei kyetä löytämään esimerkiksi lähdekooditason tietovirta- tai raja-arvovirheitä. Lasilaatikkotestaus ei puolestaan löydä sellaisia riskejä, jotka liittyvät käyttöjärjestelmään, ohjelmistojen määrittäisiin tai laitteiden yhteensopivuuksiin. (Jokela 2007, 8.)

V-mallin alimmalta tasolta ylöspäin siirryttäessä lähestymistapa muuttuu lasilaatikkotestauksesta yhä enemmän mustalaatikkotestaukseksi.

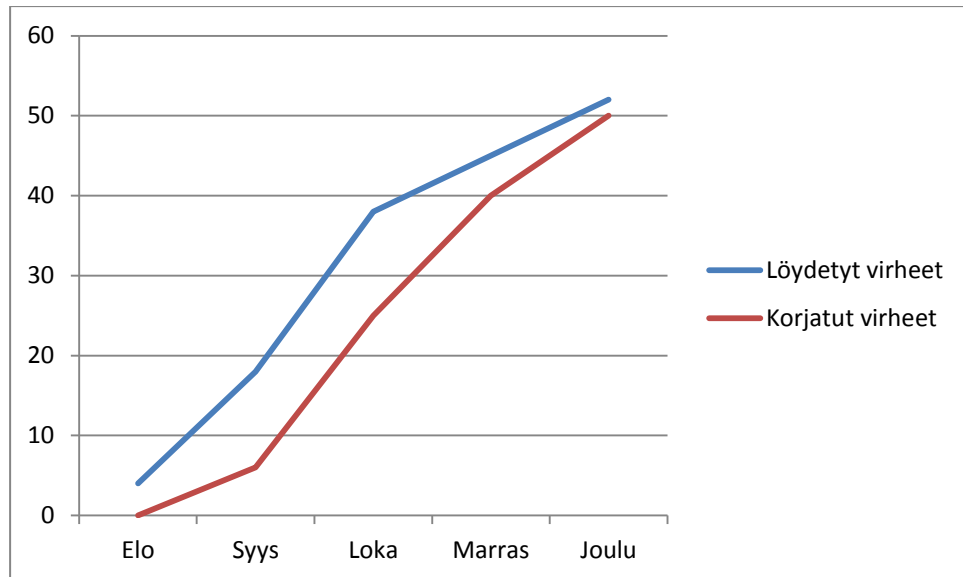
Lähestymistavan voidaan sanoa olevan kytköksissä V-malliin (Haikala & Mikkonen 2013, 209).

Tarvittavan testauksen määrää on mahdotonta arvioida. Siihen vaikuttaa testauksen laatu, käytössä olevat resurssit, kuten aika, raha, henkilöstöresurssit ja välineet. Testauksen määrä on edellä mainittujen muuttujien välillä. Testauksen lopettamisen kriteerinä voi toki olla ajan tai rahan loppuminen, mutta mieluummin lopettamiselle pitäisi asettaa hyväksymiskriteerit, jotka määritellään myös testaussuunnitelmassa. Hyväksymiskriteereinä voidaan pitää kattavuusmittareita. Sen lisäksi, että niillä voidaan varmistaa, että käytetty testiaineisto aiheuttaa testattavan ohjelman kattavan suorittamisen, niitä voidaan käyttää riittävän testauksen toteennäyttämiseksi. (Haikala & Mikkonen 2013, 211.)

Kattavuusmittareilla voidaan mitata muun muassa lausekattavuutta, päätöskattavuutta ja ehtokattavuutta. Testattava yksikkö saavuttaa sadan prosentin kattavuuden, kun sen jokainen lause suoritetaan vähintään kerran. Päätöskattavuus saa sadan prosentin arvon, kun ehtorakenteen päätös saa vähintäänkin kerran molemmat arvot, tosi ja epätosi. Ehtokattavuuden edellytys on, että päätöksen kaikki ehdot saavat kaikki arvonsa. Näiden lisäksi on myös moniehtokattavuus. Sillä tarkoitetaan, että testaus suoritetaan kaikkien ehtojen kaikilla kombinaatioilla. (Haikala & Mikkonen 2013, 211–212.)

Testauksen lopettamispäätöksen perusteena voidaan soveltaa S-käyrää. S-käyrä kuvaa löydettyjen virheiden määrää ajan funktiona. Samaan kuvioon voidaan piirtää myös korjattujen virheiden lukumäärä. Kun uusien löydettyjen virheiden lukumäärä pienenee, toisin sanoen kuvaajan kulmakerroin pienenee sekä löydettyjen ja korjattujen virheiden välinen erotus ei ole suuri, voidaan testaus päättää. (Jääskeläinen, Katara & Vuori 2014, 329.)





KUVIO 16. S-käyrä virheiden testauksessa

Kuviossa 16 on kuvattu löydettyjen ja korjattujen virheiden lukumäärää ajan funktiona. Alussa virheiden määrä on pieni, kun testausta ei ole tehty. Kun testaaminen päästään aloittamaan, alkaa myös virheitä löytyä. Lopulta virheitä on löytynyt hieman yli viisikymmentä, joista lähes kaikki on korjattu. Testaus voidaan lopettaa ja ohjelmisto voidaan laittaa markkinoille.

### 3.5 Käyttöönotto

Käyttöönottovaihe voidaan jakaa suunnittelu- ja toteutusvaiheeseen.

Suunnitteluvaiheen tarkoituksena on varmistaa, että ohjelmisto voidaan ottaa käyttöön. Suunnitteluvaiheessa on laadittava asiaan liittyvät suunnitelmat ja valmistella loppukäyttäjät ohjelmiston käyttöönottoa varten. (Murch 2002, 126.)

Suunnitelmista on selvittävä tiedostojen ja tietokantojen luomisen, päivittämisen ja ylläpidon menetelmät, käyttöönoton valvonnan, varmuuskopioinnin ja käyttöönoton menetelmät sekä varasuunnitelmat mahdollisten riskien toteutuessa. Nämä menetelmät toimivat muun muassa muunnostoimien aikatauluttamisen perustana. (Murch 2002, 127.)

Kun suunnitelmat on hyväksytty, voidaan varsinainen käyttöönotto suorittaa. Käyttöönoton onnistumisen varmistamiseksi siihen yhdistetään käyttäjätyytyväisyyskyselyitä ja järjestelmätarkastuksia (Murch 2002, 131).

Vaikka ketterissä menetelmissä tuotetaan joka pyrähdyksessä toimiva tuote, ei toteutuksia ole pakko ottaa joka pyrähdysten jälkeen käyttöön. Ketterät menetelmät sallivat myös vaiheittaisen käyttöönoton. Käyttöönotto voidaan pilotoida tehtäväksi sopivalla inkrementillä ja siitä saadut muutokset ja käyttäjätarinat sisällytetään kehitysjonoon. (Auer ym. 2013, 93.)

Käyttöönottovaiheeseen sisältyy myös käyttöoppaiden ja menettelyohjeiden tuottaminen sekä henkilöstön kouluttaminen. Nämä pitää tehdä ennen kuin varsinaista käyttöönottoa tehdään, jotta käyttäjät pystyvät osallistumaan käyttöönottoon tiiviimmin. (Murch 2002, 120.)

### 3.6 Projektin päättäminen

Projektin lopetusvaiheessa tuotetaan loppuraportti, johon kerätään ja analysoidaan projektin kulku sekä arvioidaan projektin tulokset. Loppuraportista ilmenee projektin tarkoitus, ja siinä verrataan suunnitelmaa toteutumaan tuotoksen, kustannusten ja aikataulun osalta. Siinä arvioidaan myös organisaation toimivuutta. Loppuraportissa kerrotaan, mitä asioita on opittu, mitä ongelmia tunnistettiin ja miten ne ratkaistiin. Myös toteutuneet riskit kuvataan ja arvioidaan, miten niiltä voitaisiin seuraavalla kerralla välttyä. Loppuraporttiin kirjataan myös hyväksi havaitut työskentelytavat, työvälineet ja menetelmät. (Huotari 2012.)

Projektin päätöskokoukseen osallistuu projektin johtoryhmä. Projektin loppuraportti käsitellään ja hyväksytään. Päätöskokous tekee oman arvionsa projektin onnistumisesta. Kokouksessa vastuu tuotoksesta siirtyy tilaajalle. Kokous myös tekee esityksen projektin päättämisestä sen asettajalle. (Huotari 2012.)

### 3.7 Projektin keskeyttäminen

Joskus on aiheellista keskeyttää tai päättää projekti ilman, että sen tuloksia hyväksytään. Tällaisia tilanteita voivat olla muun muassa:

- projektiin tulevan muutoksen seurauksena ei projektia kannata jatkaa
- projektin heikko kannattavuus

- projektiorganisaation toiminnassa esiintyvät häiriötekijät.

(Kosonen 2014.)

Siitä huolimatta, että projekti keskeytetään, kannattaa se arvioida samoilla menetelmillä kuin valmiiksi saatu projekti. Silloin siitä saadaan oppia tuleviin projekteihin eikä seuraavalla kerralla projektin keskeytyminen johdu samasta syystä.

#### 4 PROJEKTIN TOTEUTUS

Yritys Oy:ssä on otettu vuoden 2014 aikana käyttöön ohjelmisto, jonka avulla tavoitteena on parantaa yhtiön sisäistä raportointia. Ennen käyttöönottoa lähdetieto on ollut hajallaan erilaisissa järjestelmissä, ja projektin tarkoituksena oli koota tiedot yhteen ja sitä kautta löytää kustannustehokkuutta heikentäviä seikkoja ja saada reaaliaikaisia talouden tunnuslukuja. Seuraavaksi analysoidaan toteutettua projektia teorian pohjalta. Tavoite on analysoida toteutettua projektia ja tunnistaa projektityön kehityskohteet, jotta yritys voi kehittää omaa projektityöskentelyään ja siten kartuttaa omaa osaamistaan ja kilpailukykyään.



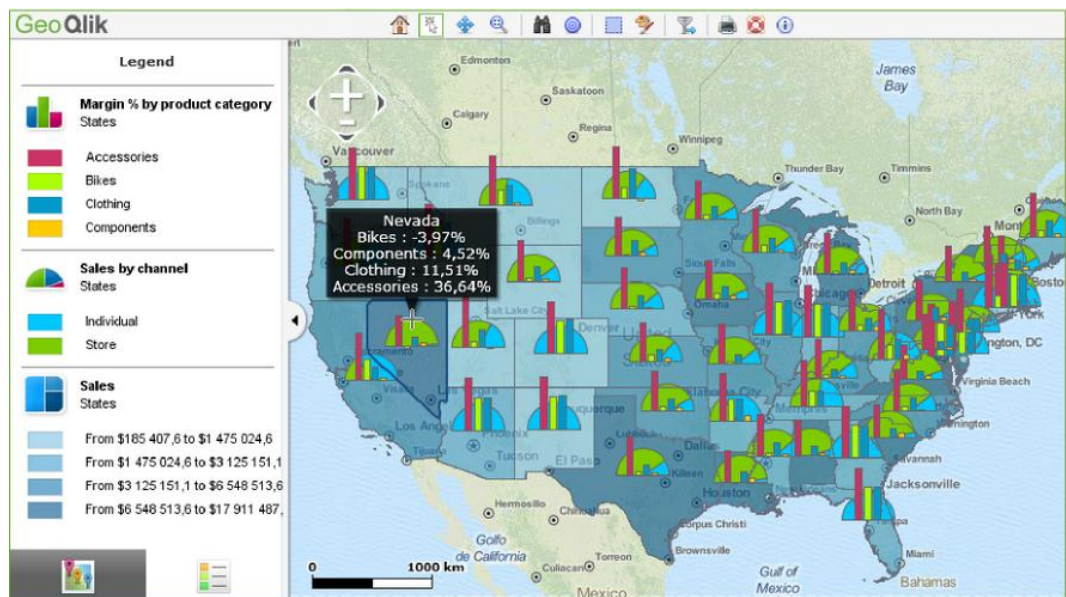
KUVIO 17. Raporttimalli raportointiohjelmasta (MyXcelsius.com 2015)

Kuvio 17 on yhdenlainen esimerkki ohjelmiston tuottamasta raportista. Esimerkki kuvaa tarinoiden määrää kuukausitasolla sekä esittää piirakkadiagrammina tarinoiden suhteellisen määrän aihealueen perusteella.



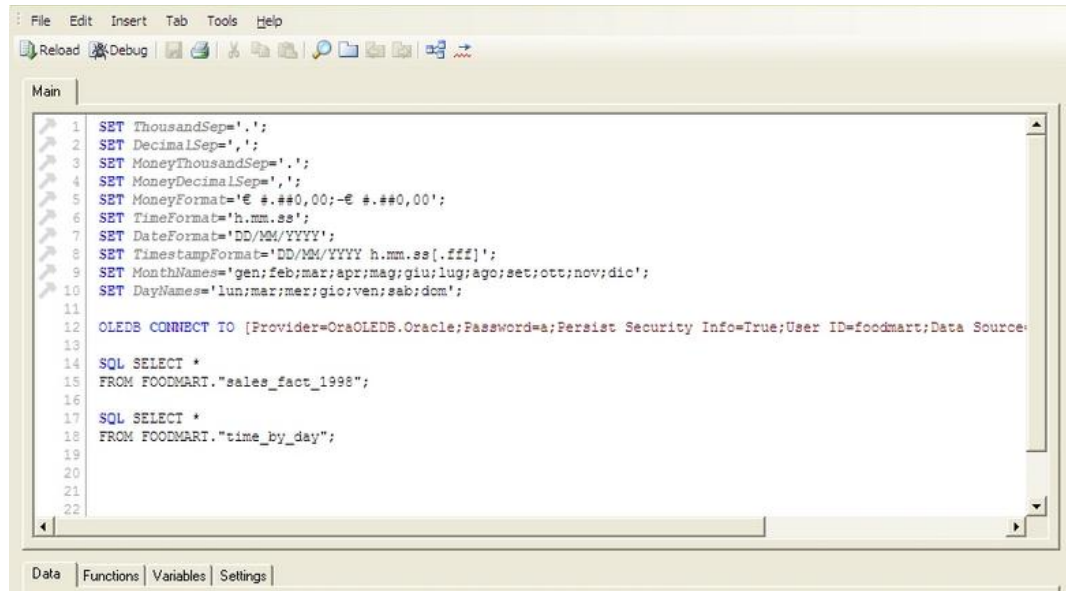
KUVIO 18. Raporttimalli raportointisovelluksesta (Practical 2015)

Kuviossa 18 on kuvattu energian kulutusta erilaisilla mittareilla. Esimerkissä nähdään muun muassa energiankulutus suhteessa rakennuksen tyyppiin sekä kokonaiskulutus.



KUVIO 19. Raportti karttalaajennuksen kanssa (Qlik.com 2015)

Edellisen sivun kuviossa 19 on ohjelmistoon lisätty karttalaajennus. Esimerkissä on esitetty Yhdysvaltojen osavaltiokohtaisia myyntilukuja. Yhtälaillla voitaisiin luoda kaupunki- tai kaupunginosakohtaisia raportteja.



```

1  SET ThousandSep='.';
2  SET DecimalSep=',';
3  SET MoneyThousandSep='.';
4  SET MoneyDecimalSep='.';
5  SET MoneyFormat='€ ###0,00;-€ ###0,00';
6  SET TimeFormat='h.mm.ss';
7  SET DateFormat='DD/MM/YYYY';
8  SET TimestampFormat='DD/MM/YYYY h.mm.ss[.fff]';
9  SET MonthNames='gen;feb;mar;apr;mag;giu;lug;ago;set;ott;nov;dic';
10 SET DayNames='Lun;mar;mer;gio;ven;sab;dom';
11
12 OLEDB CONNECT TO [Provider=OraOLEDB.Oracle;Password=a;Persist Security Info=True;User ID=foodmart;Data Source=
13
14 SQL SELECT *
15 FROM FOODMART."sales_fact_1998";
16
17 SQL SELECT *
18 FROM FOODMART."time_by_day";
19
20
21
22

```

## KUVIO 20. Ohjelmointityökalu

Yllä olevassa kuviossa 20 on kuvakaappaus ohjelmointityökalusta. Esimerkissä asetetaan päivämäärä- ja aikaerottimia sekä kuukaudet ja viikonpäivät italiaksi. Sen jälkeen muodostetaan yhteys tietokantaan ja suoritetaan kaksi SQL-kyselyä.

### 4.1 Valmistelu

Kilpaillulla alalla, jossa tulos tehdään alihankinnalla, on välttämätöntä pystyä seuraamaan taloudellisia tunnuslukuja riittävän nopeasti. Yrityksessä oli tunnistettu tarve hankkia ohjelmisto, joka pystyy keräämään tarvittavat tiedot eri lähteistä ja koostamaan niistä raportit. Valmistelun lopuksi projektiehdotus hyväksyttiin ja päätettiin jatkaa suunnitteluvaiheeseen.

## 4.2 Suunnittelu

Projektivastaavan mukaan suunnittelu oli lähinnä käytäväkeskustelua siitä, millaisia raportteja järjestelmästä halutaan. Näitä tietoja ei kirjattu mihinkään eikä vaatimuksia muutenkaan määritelty.

Ohjelmisto valittiin aiempien kokemusten perusteella ja toimittaja kilpailutettiin. Kilpailutuksessa arvioitiin ohjelmistotoimittajan referenssejä ja hintaa.

Suunnitteluvaiheessa ei tuotettu kirjallista projektisuunnitelmaa, josta olisi selvinnyt tavoitteet, vastuut ja yksityiskohtaiset aikataulut. Projektivastaavan mukaan tavoitteena oli saada yhtiön datamallin rakentaminen eri järjestelmistä nyt hankittuun ohjelmistoon ja johtamisen kannalta olennaiset tunnusluvut näkyviin vuoden 2014 aikana.

Projektiin kiinnitettiin henkilöitä, joiden henkilökohtainen työmäärä verrattuna käytettävissä olevaan työaikaan ylitti 100 %. Sen seurauksena näiden henkilöiden omat työt viivästyivät.

## 4.3 Toteutus

Projektin ohjelmointiosuus tehtiin kolmessa vaiheessa yhtiön pääkonttorilla. Kukin vaihe oli kolmipäiväinen, ja vaiheiden välillä oli noin kahdeksan viikkoa. Ensimmäisen vaiheen jälkeen tehtävänä oli saada raportointiohjelmiston tietokantakäyttöoikeudet kuntoon ja rajapinnat kaikkiin tietolähteisiin toimimaan.

Projektivastaavan mukaan oli ajateltu, että toisessa ja kolmannessa vaiheessa ohjelmiston toimittaja kouluttaa yhtiön omaa väkeä raporttien luontiin. Kaiken kaikkiaan koulutukseen olisi ollut käytettävissä 45 tuntia.

Toisessa vaiheessa yhtiön talousjohtaja ja liiketoimintajohtaja kävivät vuorotellen kertomassa vaatimuksiaan tulostettaville raporteille. Ohjelmistotoimittajan konsultti yhdessä yhtiön kirjanpitäjän kanssa ohjelmoivat raporttinäkymiä. Kirjanpitäjä kertoi, mistä tietokannasta halutut tiedot löytyvät, ja konsultti ohjelmoi skriptit, joilla tiedot saatiin haettua ja tulostettua. Lähdekoodia ei

kommentoitu eikä siitä tehty dokumentteja. Paikalla oli yhtiön omaa henkilökuntaa keskimäärin neljä henkilöä.

Vaikka toisen vaiheen jälkeen projektiryhmän jäsenet kertoivat projektin vastaavalle, että koulutus ei ollut sovitun mukaista, kolmannessa vaiheessa toteutusta jatkettiin samalla tavalla kuin toisessa vaiheessa. Kolmannessa vaiheessa paikalla oli keskimäärin kolme henkilöä yhtiön henkilöstöstä.

Kolmannen vaiheen jälkeen oli saatu tuotettua talous- ja liiketoimintajohtajan haluamat näkymät.

#### 4.4 Testaus

Helposti ajatellaan, että testaamista tehdään, jos yleensäkin tehdään, vain sen vuoksi, että ohjelmisto läpäisee tietyt testit. Testaamisen tarkoitus on löytää virheet ja ehkäistä virheiden aiheuttamat ongelmat. Sen sijaan, että ollaan tyytyväisiä siihen, kun ohjelmisto läpäisee testin, tulisi olla tyytyväinen aina, kun testauksessa ohjelmistosta löytyy virhe.

Yritys ei itse tuota ohjelmistoja, joten testauksessa sen tulisi ehdottomasti panostaa järjestelmän alfa- ja beeta-testaukseen. Kun virheet löydetään ennen hyväksymistestausta, on kustannusten osoittaminen järjestelmän toimittajalle helpompaa. Ammattimaista testausta ei kuitenkaan tule jättää tekemättä yhtiön omien testien vuoksi.

Ohjelmointituotosta testattiin yrityksessä mustalaatikkotestausmenetelmällä. Menetelmän ongelmaksi testauksen aikana muodostui se, että raportit, joihin tuloksia verrattiin, olivat lähtöarvoiltaan erilaisia ohjelmiston käyttämän datan lähtöarvoihin verrattuna. Vaikka useimmiten päästiin todennäköisesti oikeaan tulokseen, ei täyttä varmuutta voitu saavuttaa. Muilla menetelmillä ei ohjelmistoa testattu eikä testituloksia tallennettu.

#### 4.5 Käyttöönotto

Käyttöönottovaiheessa ohjelmisto otettiin tuotantoon suoraan kaikille loppukäyttäjille. Tuotantoon otettu versio ei kuitenkaan ollut sellainen, että se



olisi ollut hyödyllinen kaikille niille, joille pidettiin koulutusta ja annettiin pääsy ohjelmaan. Loppukäyttäjille järjestettiin tunnin koulutus. Ohjelman käyttämiseen ei ole tehty käyttöohjetta.

#### 4.6 Päätäminen

Projektia ei ole vielä päätetty, vaikka projektivastaavan mukaan annetut tavoitteet on saavutettu aikataulussa ja annetussa budjetissa. Ennen päättämistä tehdään käyttöliittymään parannuksia.

#### 4.7 Ylläpito

Ohjelmiston ylläpitoa suoritetaan päivittämällä säännöllisesti ohjelmistoa valmistajalta tulevin päivitystiedostoin. Ennen päivityksiä ajetaan tietyt raportit, jotka sisältävät tietoja kaikista eri lähteistä. Päivitysten jälkeen ajetaan samat raportit ja verrataan tietojen eheyttä.

## 5 JOHTOPÄÄTÖKSET JA KEHITYSKOHTTEET

Projektipäällikön mukaan budjetti ja aikataulu ovat pitäneet. Joidenkin asioiden kohdalla viivästystä on aiheuttanut se, että osaamista on jouduttu ostamaan ohjelmistotoimittajalta. Myös toiminnallisiin tavoitteisiin on päästy. Projektin myötä yhtiöön on tullut osaamista raporttien luomiseen, mutta ei niin paljoa kuin tarkoitus on ollut. Projekti jatkuu vielä käytettävyyteen liittyvin parannuksin.

Ohjelmiston loppukäyttäjien mukaan raportointi ei suurimmilta osin vastaa heidän tarpeitaan. Tietoa on kerralla saatavilla liikaa, osa olennaisista tiedoista puuttuu ja ohjelmiston käytettävyys on heikko.

Puutteellinen tai olematon dokumentointi niin suunnitelman kuin vaatimustenmäärittelyn osalta kasvattaa riskien toteutumisen todennäköisyyttä ja aiheuttaa projektin aikataulun tarpeetonta venymistä. Ohjelmistotuotoksen tarkoituksen- ja tilauksenmukaisuutta on hankala hahmottaa, jos ei ole selvillä, mitä kaikkea siltä odotetaan. Kaikki nämä tekijät lisäävät kustannuksia. Yleisessä tiedossa on, että virheen korjaaminen vaatimusmäärittelyvaiheessa aiheuttaa kustannuksia vain murto-osan siitä, mitä se aiheuttaa toteutusvaiheessa.

Projektsuunnitelman puutteen vuoksi ei resursseja osattu hyödyntää oikein eivätkä projektiryhmän jäsenet aina tienneet, mikä heidän tehtävänsä oli ja mitä heiltä odotettiin. Sen sijaan, että ohjelmistotoimittajan konsultti kirjoitti skriptejä kirjanpitäjän kanssa, olisi se aika pitänyt käyttää projektiryhmän ohjelmoijien kouluttamiseen. Pelkästään henkilöstöresursseja hukattiin 15 päivää, ja lisäkustannuksia syntyi myös matka- ja majoituskuluista. Osaamista ei saatu hankittua toivotulla tavalla.

Tämänkaltaiseen ohjelmiston käyttöönottoprojektiin, jossa itse voidaan tuottaa raportteja yksi tai useampi kerrallaan, olisi voitu soveltaa ketteriä menetelmiä. Käyttäjätapauksia apuna käyttäen olisi voitu määrittää tehtävät, jotka olisi voitu priorisoida tuotteen omistajan toimesta ja projektiryhmä olisi osannut muodostaa ne sellaiseen työjonoon, ettei samaa asiaa ohjelmoida useasti vaan edellisestä skriptistä olisi voitu hyödyntää. Pyrähdyksittäin olisi voitu luoda uusia raporttinäkymiä. Myös käytettävyyteen liittyvät parannukset olisi voitu tehdä pyrähdyksittäin.

Toisaalta käytettävyyteen liittyvät parannukset olisi voitu erottaa omaksi projektikseen. Tekemällä kunnollisen vaatimusmäärittelyn käyttökokemusten perusteella olisi helpompi varmistaa, että tulokset ovat toivotunlaisia.

Tässä projektissa ei noudatettu minkäänlaista projektinhallintamallia, vaikka projekti sinällään täyttikin projektin määritelmät kertaluontoisuutensa, aikataulutuksensa sekä budjetointinsa puolesta. Yrityksessä on aloitettu koulutukset projektityöskentelystä.

### 5.1 Olemassa olevat hyvät projektikäytännöt

Yrityksen johdossa on halu kehittää yhtiötä ja sen toimintamalleja. Erilaisia kehitysprojekteja aloitetaan aktiivisesti, ja hyvin perustellut projektiehdotukset saavat kannatusta. Projektien lopputuloksia analysoidaan ja niistä pyritään löytämään kehitettävää.

### 5.2 Kehityskohteet

Yrityksessä pitäisi ottaa käyttöön projektinhallintamalleja. Tämänhetkisessä tilanteessa, kun aloitetaan paljon projekteja ilman tarkasti määriteltyjä tavoitteita, syntyy erilaisia menetyksiä niin resursseissa, rahassa kuin ajassakin. Myöskään lopputuloksen laatu ei välttämättä vastaa toivottua.

Merkittävimpänä kehityskohteena on vaatimusten määrittely ja dokumentointi. Ilman vaatimusmäärittelyä on lähes mahdotonta saada vertailukelpoisia tarjouksia ohjelmistotoimittajilta. Myöskään määritettyjen vaatimusten toteutumista ei voida seurata ilman vaatimusmäärittelyä.

Toinen tärkeä dokumentaatio on projektisuunnitelma. Ilman projektisuunnitelmaa projektin etenemistä ja onnistumista taloudellisesti, ajallisesti ja tuotannollisesti on hankala mitata. Projektisuunnitelma luo pohjan ohjaukselle ja valvonnalle. Ilman projektisuunnitelmaa ei vaihesuunnitteluakaan voida toteuttaa.

Työskentelyn kulku projektiehdotuksesta valmiiksi projektiksi pitää saada sellaiseksi, että varsinaisen projektin toteuttaminen noudattaa tehtyä projektisuunnitelmaa ja dokumentointi kaikissa vaiheissa olisi riittävän tarkkaa.

Tavoitteena on saada tuotantoon mahdollisimman hyvin suunniteltu ja toimiva projektityöskentelymalli.

Lisäksi alemmalla tasolla projektista vastaavan henkilön tulisi pitää huoli, että

- jokainen projektiryhmän jäsen tuntee projektiryhmän kokoonpanon
- jokainen projektiryhmän jäsen tietää oman tehtävänsä ja muut odotukset
- jokainen projektiryhmän jäsen tietää aikataulun ja resurssit, jotka ovat käytössä.
- jokainen projektiryhmän jäsen tietää, kuinka monta prosenttia kokonaistyöajasta on käytettävissä kyseiseen projektiin.

Projektiin kiinnitettävien henkilöiden työkuormitus ei saa pitkäkestoisesti ylittää 100 %. Projektityöskentelyssä on normaalia, että lyhytaikaisesti työpäivät venyvät, mutta kun venyminen on pitkäkestoista, tulee vastaan jaksaminen sekä työaikalainsäädäntö.

Henkilöstön osaamista voisi yrittää hyödyntää enemmän. Yhtiössä on lähes 500 työntekijää ja varmasti löytyy paljon sellaista osaamista, mitä työntekijät eivät päivittäisessä työssään käytä. Yhtiön toimialan luonteen vuoksi talvi on hiljaisempaa aikaa, joten projektitoimintaan kannattaisi panostaa resursseja silloin.

Projektiin liittyvät riskit tulee kartoittaa ja analysoida sekä tehdä tiettäväksi koko projektiryhmälle. Näin jokainen osaa omalta osaltaan varautua niihin ja pystyy ilmoittamaan, kun riski uhkaa toteutua. Tällöin on mahdollista ehtiä reagoimaan uhkaan, ennen kuin se muuttuu ongelmaksi.

On selvää, että yritys tarvitsee selkeän projektityöskentelymallin, ja olisi hyvä, että se tehtäisiin kirjallisesti ja vaaditut työvaiheet kirjattaisiin. Tekemällä selkeä ohjeistus siitä, miten yrityksen projektit hoidetaan, voidaan mahdollisimman tehokkaasti keskittyä toteutettavaan projektiin ja sen tavoitteiden saavuttamiseen sen sijaan, että hukataan aikaa kulloinkin projektin raamien luomiseen. On myös syytä tiedostaa, että projektimallia tulee jatkuvasti kehittää ja projektimalleja voi olla yrityksessä muutama erilainen, joista voidaan valita kulloiseenkin projektiin parhaiten soveltuva. Positiivista on, että yrityksen tahtotila projektityöskentelyn

kehittämiseen on korkealla. Ensimmäiseksi olisi tärkeää määritellä, mitä yritys tahtoo omalta projektityöskentelyltään, ja kuunnella tarvittaessa työyhteisön jäseniä ja kannustaa avoimeen keskusteluun. Liian helposti nämä perusasiat jäävät tekemättä, kun päivärutiinit vievät kaiken ajan. Yrityksen on kuitenkin syytä tiedostaa, että kun halutaan kustannustehokkuutta, se vaatii myös panostusta.

## 6 YHTEENVETO

Työn tavoitteena oli löytää kehitettävää yhtiön projektityöskentelystä ja löytää keinoja kustannussäästöjen saavuttamiseksi. Yritys Oy:llä ei ollut käytössään projektinhallinta- tai työskentelymalleja. Tässä projektissa työskentely oli organisoimatonta ja jo pelkästään se aiheutti merkittäviä turhia kustannuksia niin suoraan kuin välillisesti. Ottamalla käyttöön projektinhallintamallin Yritys Oy pystyy seuraamaan resursseja, projektin edistymistä ja budjettia. Yleisesti on tiedossa, että jo pelkästään tällä keinolla saadaan kustannussäästöjä monelta projektin eri osa-alueelta.

Yritys Oy on aloittanut projektityöskentelykoulutukset yhtiön esimiehille. Näitä koulutuksia on järjestetty muutaman kerran eri puolilla Suomea. Tulevaisuudessa projektityöskentelytapoja jalkautetaan tuotannon työntekijöille. Tavoitteena on myös tehdä yhtiölle yleinen projektitoimintamalli. Kun projektitoimintamalli on saatu valmiiksi, siitä voi muokata muutaman erilaisen mallin erityyppisiin projekteihin, koska ohjelmistoprojektit luonteeltaan erilaisia kuin verkonrakennusprojektit. Lisäksi yhtiön kannattaa harkita hyvän projektityöskentelyä tukevan ohjelmiston hankintaa ja panostaa sen kunnolliseen koulutukseen ja käyttöönottoon. Hyvä ohjelmisto vähentää käsin tehtävän työskentelyn määrää luomalla muun muassa raportteja, kaavioita ja laskelmia itse.

Hyvä talo rakennetaan hyvin tehdyn perustuksen päälle ja sama periaate koskee myös ohjelmistoprojekteja. Hyvä ohjelmisto rakennetaan hyvän vaatimusmäärittelyn päälle. Sanotaan, että hyvin suunniteltu on puoliksi tehty.

## LÄHTEET

Agilemanifesto. 2001a. Ketterän ohjelmistokehityksen julistus [viitattu 7.11.2014]. Saatavissa: <http://agilemanifesto.org/iso/fi/>

Agilemanifesto. 2001b. Julistuksen takana olevat periaatteet [viitattu 7.11.2014]. Saatavissa: <http://agilemanifesto.org/iso/fi/principles>

Ansamäki, J. 2002. Tietojärjestelmien suunnittelu [viitattu 15.1.2015].

Kymenlaakson ammattikorkeakoulu. Saatavissa:  
[www2.kyamk.fi/~zjaan/kurssit/tjs/OSA1.PPT](http://www2.kyamk.fi/~zjaan/kurssit/tjs/OSA1.PPT)

Auer, A., Auer, L., Heinäsmäki, M., Hölttä, J., Kalliala, E., Laanti, M., Laine, K., Lekman, L., Miinalainen, P., Naski, H., Piiparinen, T., Puhakka, H., Pyhäjärvi, M., Pääkkönen, T., Räisänen, S., Sora, H., Taipale, M., Talvio, J., Tanninen, A., Toikkanen, T., Toivola, T., Toro, K., Valsta, A., Väyrynen, V. & von Weissenberg, M. 2013. Ketterää kehitystä. Vantaa: Hansaprint Oy.

B4ubuild.com. [viitattu 24.2.2015]. Saatavissa:

[http://www.b4ubuild.com/resources/schedule/gant\\_chart\\_02b.jpg](http://www.b4ubuild.com/resources/schedule/gant_chart_02b.jpg)

Haikala, I. & Mikkonen, T. 2013. Ohjelmistotuotannon käytännöt. Hämeenlinna: Kariston Kirjapaino Oy.

Helsingin Yliopisto. 2014. Projektiopas [viitattu 4.11.2014]. Helsingin Yliopisto, Tietotekniikkaosasto. Saatavissa:

[https://notes.helsinki.fi/tietos/yhteiset/palveluluettelo.nsf/f55a48802059d911c22578b5001f5280/cb28b3cfad12c799c22578ee003c0e84/\\$FILE/ATTRPBT6/projektiopas.pdf](https://notes.helsinki.fi/tietos/yhteiset/palveluluettelo.nsf/f55a48802059d911c22578b5001f5280/cb28b3cfad12c799c22578ee003c0e84/$FILE/ATTRPBT6/projektiopas.pdf)

Huotari, Jouni. 2012. Osaaminen kilpailukyvyksi – Projektin lopetus – kalvosarja [viitattu 15.1.2015]. Jyväskylän ammattikorkeakoulu. Saatavissa:

[http://homes.jamk.fi/~huojo/opetus/IIZP2010/IIZP2010\\_ProjektinLopetus.pdf](http://homes.jamk.fi/~huojo/opetus/IIZP2010/IIZP2010_ProjektinLopetus.pdf)

IT-barometri 2013 – maksuton versio. 2013. Tietotekniikan liitto.

- Jokela, J. 2007. Mustalaatikkotestaus mobiilipäätelaitteella [viitattu 25.2.2015]. Tampereen ammattikorkeakoulu. Saatavissa: <https://www.theseus.fi/bitstream/handle/10024/10264/Jokela.Jyri.pdf?sequence=2>
- Jääskeläinen, A., Katara, M. & Vuori, M.. 2014. Ohjelmistojen testaus [viitattu 9.3.2015]. Tampereen teknillinen yliopisto, Tietotekniikan laitos. Saatavissa: [http://www.cs.tut.fi/~testaus/s2014/luennot/TIE-21200\\_2014.pdf](http://www.cs.tut.fi/~testaus/s2014/luennot/TIE-21200_2014.pdf)
- Kosonen, H. Projektinhallinta [viitattu 15.1.2014]. Mikkelin ammattikorkeakoulu. Saatavissa: [http://cna.mikkeliyamk.fi/Public/KosonenH/projekti\\_aikuis\\_2013/Kurssin%20aloi tus/PROJEKTINHALLINTA\\_kokonaisprosessi.pdf](http://cna.mikkeliyamk.fi/Public/KosonenH/projekti_aikuis_2013/Kurssin%20aloi tus/PROJEKTINHALLINTA_kokonaisprosessi.pdf)
- Leppäniemi, P. 2015. Ketterät menetelmät: Scrum [viitattu 15.1.2015]. Saatavissa: <http://ohjelmistotuotanto.panuleppaniemi.com/agile-scrum/>
- Litke, H & Kunow, I. 2004. Projektinhallinta. Masku: Maskun kirjapaino Oy
- Lukkari, U. 2004. Digitaalisen sisältötuotantoprojektin hallinta. IT Press. Helsinki: Edita Prima Oy
- MAMK. 2002. Projektinhallinta [viitattu 7.11.2014]. Mikkelin ammattikorkeakoulu. Saatavissa: [cna.mikkeliyamk.fi/Public/.../Luennot/III.3%20Projektinhallinta.ppt](http://cna.mikkeliyamk.fi/Public/.../Luennot/III.3%20Projektinhallinta.ppt)
- MAMK. 2013. Projektiosaaminen [viitattu 7.11.2014]. Mikkelin ammattikorkeakoulu. Saatavissa: [cna.mikkeliyamk.fi/Public/.../PROJEKTIOSAAMINEN\\_MAMK.pptx](http://cna.mikkeliyamk.fi/Public/.../PROJEKTIOSAAMINEN_MAMK.pptx)
- Microsoft. 2015. Kriittisen polun näyttäminen [viitattu 9.3.2015]. Saatavissa: <https://support.office.com/fi-fi/article/Kriittisen-polun-n%C3%A4ytt%C3%A4minen-10592326-5f31-4506-b7af-c20fc97ab5a3?ui=fi-FI&rs=fi-FI&ad=FI>
- Murch, Richard. 2002. IT-projektinhallinta. IT Press. Helsinki: Edita Prima Oy
- MyXelsius.com [viitattu 24.2.2015]. Saatavissa: <http://myxcelsius.com/wp-content/uploads/2009/02/qlikview.png>



- Myyry, J. 2002. V-mallin mukainen testausmenetelmä [viitattu 24.2.2015].  
Saatavissa: <http://www.soberit.hut.fi/T-76.115/01-02/palautukset/groups/Confuse/t3/docs/vmalli/vmalli.pdf>
- OK-opintokeskus. 2015. Swot-analyysi [viitattu 12.2.2015]. OK-opintokeskus.  
Saatavissa: <http://ok-opintokeskus.fi/swot-analyysi>
- Opetushallitus. 2014. SWOT-analyysi [viitattu 12.11.2014]. Opetushallitus.  
Saatavissa: [http://www.oph.fi/saadokset\\_ja\\_ohjeet/laadunhallinnan\\_tuki/wbl-toi/menetelmia\\_ja\\_tyovalineita/swot-analyysi](http://www.oph.fi/saadokset_ja_ohjeet/laadunhallinnan_tuki/wbl-toi/menetelmia_ja_tyovalineita/swot-analyysi)
- Practical. 2012. [viitattu 24.2.2015]. Saatavissa: <http://cdn.practicaldb.com/wp-content/uploads/2011/05/Energy.jpg>
- Phillips, J. 2005. IT-projektinhallinta-sertifikaatti. IT Press. Helsinki: Edita Prima Oy
- Project Management Institute. 2008. A Guide to the Project Management Body of Knowledge (PMBOK® Guide) — Fourth Edition [viitattu 4.11.2014]. Saatavissa: [ftp://200.57.151.197/pablo/ITILV3/PMBOK\\_4th\\_Edition.pdf](ftp://200.57.151.197/pablo/ITILV3/PMBOK_4th_Edition.pdf)
- Projekti-instituutti. 2014. Projektijohtamisen sanastoa [viitattu 7.11.2014].  
Saatavissa: [http://www.projekti-instituutti.fi/osaamisen\\_kehittaminen/projektijohtamisen\\_sanastoa](http://www.projekti-instituutti.fi/osaamisen_kehittaminen/projektijohtamisen_sanastoa)
- Qlik.com. 2015 GeoQlik [viitattu 24.2.2015]. Saatavissa: <http://market.qlik.com/geoqlik.html>
- Rannisto, A. 2015 Opintomateriaali [viitattu 13.1.2015]. Saatavissa: <http://taam.tampere.fi/opettajat/ari.rannisto/Projekti.htm>
- Silfverberg, P. 2015 Ideasta Projektiksi – Projektinvetäjän käsikirja [viitattu 15.1.2015]. Konsulttitoimisto Planpoint Oy, Työministeriö. Saatavissa: <http://www.mol.fi/esf/ennakointi/raportit/pvopas.pdf>
- Taina, J. 2009. Ohjelmistoprosessit ja ohjelmistojen laatu – luentokalvot [viitattu 6.11.2014]. Saatavissa: [http://www.cs.helsinki.fi/u/taina/opo1/k-2009/pdf/luku-6\\_2.pdf](http://www.cs.helsinki.fi/u/taina/opo1/k-2009/pdf/luku-6_2.pdf)

Tervakari, A. 2008. Ohjelmistotuotannon mallit [viitattu 6.11.2014]. Saatavissa: <https://hlab.ee.tut.fi/hmopetus/vpsist-oppimateriaali/4-menetelmia-ja-malleja/4-3-suunnittelumalleja/4-3-1-ohjelmistotuotannon-malli>

Tietotekniikan liitto ry. 2013 IT-barometri [viitattu 10.11.2014] Saatavissa: [https://ssl.ttlry.fi/sites/ttl.ttlry.mearra.com/files/file-uploads/Tutkimus/IT-barometri/IT-barometri%25202013%2520presentaatio\\_netti.pdf](https://ssl.ttlry.fi/sites/ttl.ttlry.mearra.com/files/file-uploads/Tutkimus/IT-barometri/IT-barometri%25202013%2520presentaatio_netti.pdf)

Wikipedia. 2013. Mustalaatikko (systeemiteoria) [viitattu 24.2.2015]. Saatavissa: [http://fi.wikipedia.org/wiki/Musta\\_laatikko\\_%28systeemiteoria%29](http://fi.wikipedia.org/wiki/Musta_laatikko_%28systeemiteoria%29)

Wikipedia. 2014a. Ketterä ohjelmistokehitys [viitattu 7.11.2014]. Saatavissa: [http://fi.wikipedia.org/wiki/Ketter%C3%A4\\_ohjelmistokehitys](http://fi.wikipedia.org/wiki/Ketter%C3%A4_ohjelmistokehitys)

Wikipedia. 2014b. Scrum [viitattu 7.11.2014]. Saatavissa: <http://fi.wikipedia.org/wiki/Scrum>

Wikipedia. 2015a. Gantt-kaavio [viitattu 24.2.2015]. Saatavissa: <http://fi.wikipedia.org/wiki/Gantt-kaavio>

Wikipedia. 2015b. SWOT-analyysi [viitattu 12.2.2015]. Saatavissa: <http://fi.wikipedia.org/wiki/SWOT-analyysi>

Yritys Oy. 2014. Tilinpäätöstiedote. Julkaisematon lähde.

