

Lean Startup -metodologian hyödyntäminen startup-yrityksen ohjelmistokehityksessä

TURUN YLIOPISTO
Tietotekniikan laitos
LuK-tutkielma
Tietojenkäsittelytiede
Kesäkuu 2025
Johannes Mensalo

TURUN YLIOPISTO
Tietotekniikan laitos

JOHANNES MENSALO: Lean Startup -metodologian hyödyntäminen startup-yrityksen ohjelmistokehityksessä

LuK-tutkielma, 24 s.
Tietojenkäsittelytiede
Kesäkuu 2025

Ohjelmistoalan startup-yritykset ovat yksi maailman suurimmista talouskasvun vetureista. Siitä huolimatta suurin osa startup-yrityksistä epäonnistuu kahden vuoden sisällä perustamisesta. Yleisimpiä syitä ovat huono markkinakelpoisuus, validoinnin puute tai rahoituksen loppuminen. Samalla ohjelmistokehitys tuo monia haasteita startup-yrittämiseen.

Vakiintuneissa yrityksissä käytetyt ketterän kehityksen menetelmät eivät usein sovellu pienille startup-yrityksille, joille potentiaalisten asiakkaiden ongelma ei ole selvä. Ratkaisuksi tähän on kehitetty monia menetelmiä, joista Lean Startup on noussut yhdeksi suosituimmista viimeaikoina. Sen painopiste on validoidussa oppimisessa, jatkuvassa iteroinnissa, keskittymisessä tärkeisiin ominaisuuksiin sekä hukan minimoinnissa. Ohjelmistokehityksessä edellä mainitut hyödyt ovat tärkeitä nopeuden säilyttämiseksi. Näiden lisäksi startupin menestykseen vaikuttavat tiimin sisäinen dynamiikka, koulutus sekä aikaisemmat johtamis- ja startup-yrityskokemukset.

Tämän kandidaatintutkielman tarkoitus on käsitellä Lean Startup -metodologian (LSM) hyödyntämistä teknologia-startup-yritysten ohjelmistokehityksessä sekä tiimin merkitystä yrityksen menestymisessä. Ohjelmistoalan startup-yritykset ja niiden tiimidynamiikka ovat osa uniikkia ja kiehtovaa ympäristöä, joka on ollut pitkään tutkimusalana aliedustettu. Käsittelen 27 eri tutkimusta ja teosta, jotka tutkivat Lean-menetelmien hyödyntämistä teknologia-startup-yrityksen ohjelmistokehityksessä. Tulokseni osoittavat, että Lean-menetelmästä on hyötyä ohjelmistokehityksessä, kun halutaan validoida liikeideaa ja käyttää resursseja tehokkaasti. Lisäksi tutkimuksista löytyi positiivinen korrelaatio koulutuksen, taustan ja aikaisempien startup-yrittäjäyyskokemusten vaikutuksesta yrityksen menestykseen.

Asiasanat: Lean Startup -metodologia, startup-yritys, pienin toimiva tuote, Lean-ohjelmistokehitys, ketterä kehitys

Sisällys

1 Johdanto	1
1.1 Tutkielman tarkoitus ja rajaukset	2
1.2 Menetelmät ja tiedonhaku	3
1.3 Tutkimuskysymykset	4
1.4 Tutkielman rakenne	5
2 Lean Startup -metodologia	6
2.1 Validoitu oppiminen ja edistyksen mittaaminen	8
2.2 Rakenna-mittaa-opi-palautesykli	9
2.3 Pienin toimiva tuote	11
3 Ohjelmistokehitys startup-yrityksessä	13
3.1 Lean-ohjelmistokehitys	16
3.2 Tiimin merkitys ohjelmisto startup-yrityksen menestymisessä	18
4 Yhteenveto	22
Lähdeluettelo	25

1 Johdanto

Teknologia-startup-yritykset ovat nopeasti eteneviä trendikkäisiin teknologioihin keskittyviä yrityksiä, joilla ei usein ole aikaisempaa toimintahistoriaa tai kokemusta alalta. Ne kehittävät ohjelmistoa äärimmäisessä epävarmuudessa vajavaisilla resursseilla, sekä pyrkivät samalla olemaan mahdollisimman ketteriä ja päihittämään kilpailijat nopeudessa. Teknologia-startup-yritykset ovat yksi suurimmista talouskasvun vetureista maailmanlaajuisesti. Pelkästään Euroopan alueella on yli 140 000 startup-yritystä, joista kolmanneksella on vähintään yksi rahoituskierros hankittuna. Vuonna 2021 näihin yrityksiin oli investoitu 91,1 miljardia euroa pääomaa [1].

Monien teknologia-startup-yritysten tavoitteena on häiritä (engl. *disrupt*) vakiintuneita markkinoita luomalla uusia normeja. Tällainen toimintaympäristö luo haasteita yrityksen johtamiselle, mutta myös ohjelmiston kehitykselle ja ylläpidolle. Startup-yritysten alati muuttuvassa ympäristössä asiat etenevät usein kokeilun ja erehdyksen kautta. Liikeidean validointi ja asiakkaiden tarpeiden kannalta merkityksellisen ongelman ratkaisu vaativat epävarmuuden sietokykyä sekä markkinoiden ja ennen kaikkea asiakkaan tuntemista. Startup-yrityksen tulee selvittää löytää tarpeeksi skaalautuva markkina-alue ja kehittää tuote nopeammin kuin sen kilpailevat yritykset. Merkittävä osa startup-yrityksistä ei pääse tästä vaiheesta skaalautumaan. Suurimmat haasteet ovat sopivien markkinoiden puuttuminen, huono idean validointi ja rahoituksen loppuminen [2].

Yrityksen menestyksen kannalta on tärkeää keskittyä liikeidean validointiin ja al-

lokoida resursseja sellaisten ominaisuuksien kehittämiseen, joista on asiakkaalle eniten arvoa. Usein yrittäjät eivät ole varmoja, mitkä ominaisuudet tuottavat tällaista arvoa, joten he tekevät valistuneita arvauksia kokemustensa pohjalta. Sen sijaan, että he laatisivat monimutkaisen liiketoimintasuunnitelman, he voivat tiivistää hypoteesinsa liiketoimintamallikehykseen (engl. *Business Model Canvas*). Se on kaavio, josta käy ilmi yrityksen arvonluonti itselleen sekä asiakkaille [3]. Startup-yritykset tarvitsevat järjestelmällisen tavan testata hypoteesejaan ja muokata niitä palautteen pohjalta. Lean -metodologia on tuotannon tehostamiseen kehitetty menetelmä, jonka loi 50-luvulla Toyota Motors Company [2], [4]. Monen teollisuusalan iteraation jälkeen siitä on kehittynyt erilleen Lean Startup.

1.1 Tutkielman tarkoitus ja rajaukset

Tämän kandidaatintutkielman tarkoitus on käsitellä Lean Startup -metodologian (LSM) hyödyntämistä teknologia-startup-yritysten ohjelmistokehityksessä. Ohjelmistoalan startup-yritykset ovat osa uniikkia ja kiehtovaa ympäristöä, joka on ollut pitkään tutkimusalana aliedustettu [5], [6].

Oma lähtökohtani tutkimusaiheeseen on peräisin kiinnostuksesta startup-maailmaan. Minulla on henkilökohtaista kokemusta startup-yrityksessä toimimisesta perustajajäsenenä sekä teknologiavastaavana ja ohjelmistokehittäjänä. Näiden kokemusten avulla pystyn syventymään aiheeseen ja tutkimusten havaintoihin paremmin. Olen keskustellut monissa startup-tapahtumissa alalla toimivien henkilöiden kanssa perustajajäsenestä kokeneisiin yrittäjiin ja sijoittajiin asti. Heiltä olen saanut inspiroivia ajatuksia ja arvokkaita neuvoja startup-yrittäjyyteen. Keskusteluisa toistuu usein sama teema, idean validoinnin tärkeys. Se on yksi oleellisimmista asioista startup-yrittäjyydessä, mutta samalla myös yksi helpoimmista aloittelevan yrittäjän laimiinlyödä. Sama idean validoinnin tärkeys on myös Lean Startup -metodologian kulmakiviä.

Kun kolme vuotta sitten perustin oman teknologia-yrityksen neljän hengen tiimillä, monet startup-maailman asiat olivat tuntemattomia. Nuoret startup-yrittäjät ovat usein alkuun kokemattomia ja heidän tietotaitonsa puutteellista. Tekemällä asiota ja osallistumalla aktiivisesti startup-tapahtumiin oppii paljon. Toinen tehokas tapa on lukea alaa käsittelevää kirjallisuutta. Minulle tärkeä aihealue on ollut Lean Startup -metodologia. Se tarjoaa oppeja, joiden mukaan yrittäjät voivat ohjata omaa toimintaansa ja yrityksen suuntaa tehokkaammin ja välttää yleisiä sudenkuoppia. Ohjelmistokehityksessä selkeä tavoite on tärkeää tehokkuuden ja resurssien käytön suhteen. Tarkoitukseni on käsitellä näitä aiheita olemassa olevien tutkimusten ja omien pohdintojen kautta.

Olen valinnut tutkielman lähteeksi artikkeleita, jotka käsittelevät Lean Startapia sekä ohjelmistokehitystä startup-yrityksissä ja pienissä tiimeissä. Tutkimukset sisältävät hakusanoja kuten ”*startup*”, ”*start-up*”, ”*lean*”, ”*MVP*”, tai ”*software development*”. Lyhenne MVP (engl. *minimum viable product*) tarkoittaa pienintä toimivaa tuotetta ja jatkossa käytän tässä tutkielmassa tätä lyhennettä. Tämä on yleisesti käytössä oleva termi alan tutkimuksessa. Lisäksi lähteinä olen käyttänyt tutkimuksia, jotka käsittelevät tiimidynamiikan ja perustajien tärkeyden vaikutusta startupin menestyksessä. Hakusanoilla löytyi myös artikkeleita, jotka käsittelevät Lean-metodologian hyödyntämistä vakiintuneissa yrityksissä ja organisaatioissa. Tällaiset artikkelit olen jättänyt tutkielman ulkopuolelle, sillä tarkoitus on keskittyä pienten alkuvaiheen yritysten ohjelmistokehitykseen sekä yrityksiin, jotka käyttävät pienintä toimivaa tuotetta liikeidean validoimiseen.

1.2 Menetelmät ja tiedonhaku

Tutkimusmenetelmäni on kirjallisuuskatsaus, joka perustuu aiheesta jo olemassa olevaan kirjallisuuteen ja tieteellisiin artikkeleihin. Tutkielmassa lähteinä on käytetty useita eri artikkeleita ja niitä on haettu seuraavista tietokannoista: ScienceDirect,

IEEE, Web of Science, Scopus, SAGE Journals ja INFORMS PubsOnline. Lisäksi merkittävänä kirjallisuuden lähteenä tutkielmassa on toiminut Eric Riesin teos ”The Lean Startup”. Kirja toimii pohjana Lean-metodologian ja sen historian käsittelyssä. Kirjaa on siteerattu laajasta muissa käyttämissäni lähteissä, mikä vahvistaa sen asemaa tärkeänä aineistona alan tutkimuksessa. Alla on listattuna tutkielman lähdemateriaalin tiedonhakuun käytetyt päähakutermit.

- 'startup' AND ('lean' OR 'lean process')
- ('lean startup' OR startup) AND 'software development'
- 'lean software development' AND startup
- 'psychological safety' AND ('software development' OR 'agile')

Hakukriteereillä on pyritty rajaamaan hakutulokset tutkielmaani paremmin tukeviin, ohjelmistokehitystä ja Lean Startupia, käsitteleviin teoksiin. Tutkielman lähteiksi olen valinnut yhteensä 27 artikkelia.

1.3 Tutkimuskysymykset

Edellä mainittujen rajausten pohjalta tutkielmassa tarkastellaan kahta seuraavaa tutkimuskysymystä. Tutkimuskysymys 1 käsittelee tutkielman pääaihetta. Sen avulla käsittelen Lean-metodologian ja ohjelmistokehityksen välistä suhdetta startup-ympäristössä ja miten se eroaa vakiintuneempien yritysten tavasta hallita projekteja. Startup-yritysten projektinhallinta on usein vajaavaista eikä käytössä välttämättä ole tiettyä metodologiaa. Tutkimuskysymys 2 käsittelee tutkielman toista aihetta, eli ryhmädynamiikkaa ohjelmistokehityksessä. Startup-yritysten tiimit voidaan määritellä ryhmiksi, sillä niiden jäsenet työskentelevät yhdessä yhteisen tavoitteen hyväksi. Tutkimuskysymyksen tavoitteena on selvittää, minkälaiset tiimit edistävät ohjelmistokehitystä ja ovat menestyviä.

- TK1: Miten Lean-metodologian avulla voidaan tehostaa startup-yrityksen ohjelmistokehitystä?
- TK2: Miten tiimin kokoonpano vaikuttaa ohjelmistokehitykseen?

1.4 Tutkielman rakenne

Tutkielma koostuu johdannosta, kahdesta pääluvusta sekä yhteenvedosta. Luvussa 2 käydään läpi Lean Startup -metodologian historiaa ja oppeja. Alaluvuissa 2.1 ja 2.2 käsittelen startup-yrityksen oppimisprosessia sekä liiketoimintahypoteesin testaamista. Lisäksi alaluvussa 2.3 käsittelen pienimmän toimivan tuotteen kehitystä ja ominaisuuksien priorisointia ohjelmistokehityksessä. Luvussa 3 käsitellään Lean-metodologian hyödyntämistä startup-yrityksen ohjelmistokehityksessä ja projektinhallinnassa. Alaluvussa 3.2 käyn läpi tiimin merkitystä startupin onnistuneessa ohjelmistokehityksessä ja menestyksessä yleisesti. Viimeisessä luvussa 4 kootaan yhteenvedo tutkielman aiheesta ja tutkimustuloksista. Käsittelen myös mahdollisia jatkotutkimusaiheita ja kysymyksiä, joita en kykene tähän tutkielmaan asettamieni rajojen sisällä käsittelemään.

2 Lean Startup -metodologia

Jotta voimme ymmärtää miten startupit toimivat, meidän tulee ensin tietää mitä ne ovat ja mikä on startup-yrityksen ja vakiintuneen yrityksen ero. Startup-yritykseksi lasketaan mikä tahansa inhimillinen instituutio, joka kehittää uutta tuotetta äärimmäisessä epävarmuudessa [2]. Niiden tavoite on häiritä perinteisiä markkinoita ja korvata vakiintuneita toimijoita uusilla innovatiivisilla tuotteilla. Vakiintuneet yritykset taas tavoittelevat tasaista kasvua, toisin kuin startup-yritykset, joiden kasvu voi olla jopa eksponentiaalista. Kun startup-yritys onnistuu luomaan uuden markkinan, tai häiritsemään olemassa olevia, se pakottaa vakiintuneet yritykset pysymään mukana muutoksessa. Siksi ne eivät voi pysytellä jatkuvasti perinteisissä kustannustehokkuuteen ja laadun parantamiseen keskittyvissä menetelmissään [7].

Lean-metodologia on yksi suosituimmista projektinhallintamenetelmistä, joka on vuosituhaten vaiheen jälkeen omaksuttu muokattuna paikkaamaan ohjelmistotuotannossa suosittua Agilen puutteita. Lean-metodologian juuret juontavat 50-luvun autoteollisuuteen, Toyotan tuotantojärjestelmään (TPS) (engl. *Toyota Production System*) [4]. Toyotan johtajat ymmärsivät jo varhain, että yritys oli vaarassa kasvaa liian isoksi, hitaaksi ja joustamattomaksi. Tämä on luonnollinen seuraus, kun pyritään priorisoimaan toistettavuus organisaatiossa. Tätä ilmiötä kutsutaan ”suuryritysten sairaudeksi”. Monen mutkan kautta Toyota onnistui kehittämään monia ainutlaatuisia oppimisjärjestelmiä, kuten Toyotan tuotantojärjestelmä ja Toyota Product and Process Development System (TPPDS). [8] Alkuperäinen Lean-

metodologian idea Toyotan tehtailla oli antaa kenelle tahansa tuotantolinjan työntekijöistä hierarkiasta riippumatta valta hidastaa linjaston vauhtia ongelman havaitessaan tai pysäyttää se kokonaan tilanteen niin vaatiessa. Linjaston pysähtyessä paikalle saapui esihenkilö tutkimaan tilannetta ja ongelma ratkottiin heti yhdessä. Vasta kun ongelma oli ratkaistu, tuotantolinja jatkoi normaalilla vauhdilla. Tämä mahdollisti virheiden havaitsemisen tarpeeksi aikaisessa vaiheessa ja paransi merkittävästi tuotteiden laatua [4].

Toyotan Lean-metodologia sai laajempaa tunnustusta valtavirtakirjallisuudessa MIT:n (Massachusetts Institute of Technology) viisi vuotta kestäneen autoteollisuustutkimuksen jälkeen. Siinä todettiin Lean-metodologian vaikuttaneen Yhdysvaltojen ja Japanin väliseen suureen tuottavuuseroon. Toyotan lisäksi Lean-metodologia on jo pitkään ollut käytössä monilla muilla auto- ja raskaanteollisuuden aloilla, mutta se on vasta suhteellisen hiljattain löytänyt tiensä ohjelmistokehitykseen [9].

Ries jakaa teoksessaan Lean Startup -metodologian viiteen eri osaan: *Yrittäjiä on kaikkialla*, *Yrittäjyys on johtamista*, *Validoitu oppiminen*, *Rakenna-mittaa-ope* ja *Innovaatiokirjanpito*. Tärkeimmät näistä ohjelmisto-startupin rakentamisessa ovat *validoitu oppiminen* ja *Rakenna-mittaa-ope-palautesykli*, sillä menestyäkseen startup-yrityksen perustajajäsenten on keskityttävä erityisesti niihin.

Ries kertoo teoksessaan Lean Startup -metodologian hyödyntämisestä muissa kuin startup-yrityksissä. Sitä voidaan hyödyntää yrityksen koosta riippumatta ja millä tahansa liikealalla. Yrittäjän tavoin organisaatiossa toimivia henkilöitä kutsutaan termillä *sisäinen yrittäjä* (engl. *intrapreneur*) [2], [10]. Sisäinen yrittäjä hyödyntää startup-yritysten ajattelutapaa ja toimintamalleja luodessaan uusia innovatiivisia tuotteita. He pyrkivät oppimaan virheiden kautta ja testaamaan uusia ideoita nopeaan tahtiin juuri kuten yrittäjät startup-yrityksessä sekä keskittyvät luomaan pitkäaikaista vaikutusta.

2.1 Validoitu oppiminen ja edistyksen mittaaminen

Miten tiimi kykenee erottamaan hyvät ideat huonoista? Tähän kysymykseen vastaamiseen Lean Startup -metodologia hyödyntää validoitua oppimista ja edistyksen mittaamista.

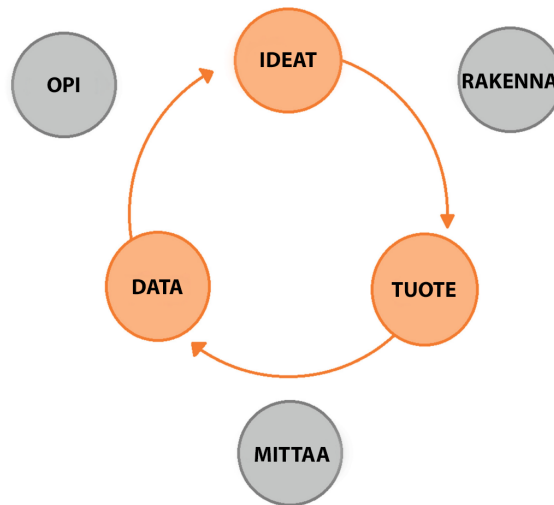
Startupien olemassaolon ehto ei ole pelkästään uusien tuotteiden luonti. Niiden tehtävä on oppia kehittämään kestävä liiketoiminta [2]. Monet aikaisen vaiheen startup-yritykset epäonnistuvat ensimmäisen kahden vuoden sisällä perustamisesta. Jopa 70–80 prosenttia niistä epäonnistuu tuottamaan ennakoitua tuottoa investoinneille [11]. Monessa tapauksessa tämä johtuu ongelman ja ratkaisun välisen yhteyden puuttumisesta ja kyvyttömyydestä oppia virheistä tuote- ja asiakaskehitysvaiheessa [12]. Toinen yleinen syy on huono liikeidea tai tarpeeksi isojen markkinoiden puute [13]. Teknologiataustaisilla yrittäjillä on tapana luoda ensin ratkaisu, ja jälkikäteen pohtia ratkaiseeko se ylipäättänsä sellaista ongelmaa, josta joku olisi valmis maksamaan. Sopivan ongelman löytäminen ja siitä menestyksekkään yrityksen luominen on hyvin haastavaa ja vaatii vahvaa ymmärrystä kohdemarkkinoista ja asiakkaasta. Moni menestystarina on ponnistanut omakohtaisesta ongelmasta, josta on lähdetty ratkaisemaan. Myöhemmin on huomattu, että myös muilla on ollut sama ongelma. Tämäkään ei aina takaa toimivaa liikeideaa. Ratkaisun ongelmaan tulee kohdata asiakkaiden todellinen tarve, jotta sillä on positiivinen muutos. Tässä validoitu oppiminen auttaa erottelemaan hyvät ideat huonoista.

Startupin alkuvaiheessa liikeidean epäonnistumisen myöntäminen voi tuntua vaikealta ja yrittäjä saattaa jäädä loputtomasti hiomaan suunnitelmaansa. Tällaisesta tilanteesta voidaan käyttää termiä analyysihalvaus (engl. *analysis paralysis*) [2]. Puutteellinen tuotteen validointi ja vähäinen kommunikointi asiakkaiden kanssa luovat helposti harhakuvan oikeasta suunnasta. Validoidun oppimisen avulla startup voi testata vision eri osia tiheään [2]. Kunnollisen taustatyön laiminlyönti hankaloittaa liikeidean kokonaiskuvan näkemistä, aiheuttaa viivästyksiä ja lisää kuluja viimeis-

tään tuotteen kehitysvaiheessa. Nämä kaikki ovat asioita, joita tulisi välttää ohjelmistotuotannossa. Suunnitteluvaiheessa aidan matalimmasta kohdasta kulkeminen säästää aikaa ja resursseja, mutta se aiheuttaa myöhemmissä vaiheissa ongelmia.

2.2 Rakenna-mittaa-opi-palautesykli

Rakenna-Mittaa-opi-palautesykli (engl. *Build-Measure-Learn cycle, BML*), on yksi Lean Startup -metodologian tärkeimmistä osista. BML-sykli on laadunparannus- ja ongelman ratkaisumalli, jolla luodaan kokeita liiketoimintamallin testaamiseksi ja tulosten mittaamiseksi. Samalla tavalla toimii William Edwards Demingin Plan-Do-Check-Act -sykli (PDCA), joka on toinen suunnitelutyössä käytetty työkalu. BML-sykliä hyödyntäessä tavoitteena on oppia uutta joko toteamalla liiketoimintahypoteesi toimivaksi tai kumoamalla se [8]. Yritykset voivat BML-syklin avulla päättää, mitä ideoita tulisi jatkaa ja mitkä tulisi hylätä. Jatkuva iterointi on Lean Startupin ydinasioita, ja sen omaksuminen ohjelmistojen kehittämiseksi vaatii teknistä osaamista ja resursseja [14]. Kuvassa 2.1 havainnollistetaan BML-syklin eri vaiheita. Se alkaa ideasta tai hypoteesista, jota lähdetään testaamaan. Rakennusvaiheessa tuotetaan uusi ominaisuus, jota voidaan testata. Asiakkaat testaavat uutta ideaa, ja saatua palautetta käytetään oppimiseen sekä idean iterointiin. Tämän jälkeen sykli alkaa alusta. Sykliä voidaan toistaa kunnes on saavutettu haluttu lopputulos tai päätetään muuttaa ideaa. Tästä tilanteesta käytetään termiä *persevere or pivot*. Pivotoinnilla tarkoitetaan strategista muutosta yrityksen suunnassa, jolla pyritään testaamaan uutta hypoteesia [2], [15].



Kuva 2.1: Rakenna-mittaa-opi-palautesykli. [16]

Startup-yrityksen edistyksen mittaaminen on hankalampaa kuin vakiintuneissa yrityksissä. Toisin kuin niillä, startup-yrityksillä ei ole mitään, minkä pohjalta tehdä ennusteita kasvusta. Siksi BML-sykli on toimiva tapa mitata startup-yrityksen edistystä. Ennen prototyypin rakentamista voidaan tehdä myös *savutesti* (engl. *smoke test*). Sen tarkoituksena on selvittää asiakkaiden kiinnostus tuotetta kohtaan ennen kuin sen kehittämiseen laitetaan resursseja. Ries kuvailee MVP:n ja Rakenna-mittaa-opi-palautesyklin suhdetta seuraavasti ”MVP on se tuoteversio, joka mahdollistaa Rakenna-mittaa-opi-palautesyklin täyskäännöksen mahdollisimman pienellä työmäärällä ja vähällä kehitysjalla.”¹ Siksi BML:stä on suurin hyöty teknologia startup-yrityksille MVP:tä rakentaessa. Sen avulla voidaan konkreettisesti testata, mitkä ominaisuudet tuottavat arvoa palvelussa.

¹”The MVP is that version of the product that enables a full turn of the build-Measure-Learn loop with a minimum amount of effort and the least amount of development time.” Ries 2011, s. 77

2.3 Pienin toimiva tuote

Jokainen liikeidea alkaa joukosta oletuksia, jotka ilmaistaan osana yrityksen strategiaa ja visiota. Startup-yrityksen alkuvaiheen ponnistelujen tavoitteena on näiden oletusten nopea testaaminen. MVP:tä ei ole tarkoitettu pelkästään todisteeksi teknologian toimimisesta. Sen tarkoituksena on testata perustavanlaatuisia liiketoimintahypoteeseja [2]. Etenkin ohjelmistoalalla startupien on tärkeää kehittää markkinoille toimiva MVP. Sen kehittämisessä keskitytään asiakkaiden tarpeiden kannalta tärkeimpiin ominaisuuksiin. Käyttäjiltä saadun palautteen avulla yrittäjä oppii kuinka hyvin MVP suoriutuu. Yrityksen kannattaa jatkaa tuotteen kehitystä, jos saatu palaute tukee alkuperäistä hypoteesia. Muussa tapauksessa yrityksen tulisi pohtia liikeidean pivotointia.

Ohjelmisto startup-yrityksen on tärkeää luoda MVP, ennen kuin tuotekehitykseen laitetaan merkittäviä resursseja. MVP auttaa idean validoinnissa ja oikean tuotteen kehittämisessä [12]. Sitä kehitetään asteittain asiakkaan kanssa sitä mukaa kun tieto halutuista ominaisuuksista tarkentuu [17]. Näin vältetään mahdollisilta riskialtteilta olettamuksilta, joihin perustajajäsenet muuten voivat langeta. *Riskialttein olettamus* (engl. *riskiest assumption*) on hypoteesi, jota ei ole testattu todeksi ja johon koko liikeidea pohjautuu [2]. Kaikkien tiimin jäsenten tulee sitoutua noudattamaan Lean Startupin oppeja, jotta tavoitteet MVP:tä kehitettäessä pysyvät yhtenäisinä. Tiettyihin asioihin, kuten kuinka loppuun mietityltä tuotteen käyttöliittymän tulisi näyttää ja tuntua, on mahdollista käyttää Lean Startupin näkökulmasta turhan paljon aikaa. Yrityksen oppimisen ja menestyksen kannalta arvokkaat *varhaiset omaksijat* (engl. *early adopters*) saattavat jopa vieroksua liian hiottua tuotetta ja vaativat tietynlaista karkeutta alussa [2]. Tuotteen tekninen kankeus annetaan siis alkuvaiheessa anteeksi. Tämä on hyvä uutinen perustajajäsenille, jotka saattavat olla liian huolissaan palvelun visuaalisesta ilmeestä. Tuotteen helppokäyt-

töisyys ja intuitiivisuus ovat tärkeämpiä kuin yksityiskohdat. Startup-yrityksen on mietittävä tarkkaan mihin ominaisuuksiin keskitytään.

3 Ohjelmistokehitys

startup-yrityksessä

Pienet ohjelmistoyritykset ja startupit ovat luovia ja joustavia instituutioita. Ne eivät ole kovin halukkaita ottamaan käyttöön toimintatapoja ja byrokratiaa, joka saattaisi haitata niiden luontaisia ominaisuuksia joustavuudessa tai tuotteen kehitystä [5]. Ne harvoin omaksuvat mitään projektinhallintamenetelmiä ja suosivat yksittäisiä ketterän kehityksen osia. Erityisesti tämä pätee startup-yrityksen elinkaaren alkuvaiheessa [18].

Ohjelmisto-startupit kohtaavat alussa monia haasteita, sillä perustajajäsenten kokemukset johtamisesta ja kehitysprosesseista ovat usein vähäisiä. Heiltä voi puuttua täysin kasvavan yrityksen johtokokemus, eikä tällaista tulevaisuutta osata visioista huolimatta aina ottaa huomioon. Myöhemmin kasvun myötä ilmenevät skaalautuvuusongelmat ja ohjelmistokehityksen kankeus aiheuttavat omia haasteitaan tehokkuuteen ja saattavat vaatia lähdekoodin refaktorointia sekä muita toimenpiteitä. Myös dokumentaation merkitys kasvaa sitä mukaa kun kehitystiimi ja tuote kasvavat. Kattava dokumentaatio helpottaa uusien työntekijöiden perehdytystä ja ennaltaehkäisee tilanteita, joissa kaikki tieto on vain yhdellä henkilöllä.

Startupin näkökulmasta on tärkeää selviytyä ensimmäisistä kuukausista vahvojen ja skaalautuvien ketterän kehityksen menetelmien omaksumisen sijaan. Ketterän kehityksen menetelmissä on myös eräs toinen haaste; niitä ei ole alun perin kehi-

tetty startup-yritysten epävarmaan ympäristöön ja tiukkoihin aikatauluihin. Vaikka alkuun perustajajäsenet olisivat haluttomia hyödyntämään ketterän kehityksen menetelmiä, niitä otetaan todennäköisesti käyttöön myöhemmässä vaiheessa. [6]

Tutkimuksessa *Software development in startup companies: A systematic mapping study* [6] listataan 15 pääteemaa (taulukko 3.1), joita käytetään eniten kuvaamaan käsitettä ”software startup”.

Mapping study – recurrent themes.

ID	Theme	Description	Frequency
T1	Lack of resources	Economical, human, and physical resources are extremely limited.	18
T2	Highly Reactive	Startups are able to quickly react to changes of the underlying market, technologies, and product (compared to more established companies)	17
T3	Innovation	Given the highly competitive ecosystem, startups need to focus on highly innovative segments of the market.	15
T4	Uncertainty	Startups deal with a highly uncertain ecosystem under different perspectives: market, product features, competition, people and finance.	14
T5	Rapidly Evolving	Successful startups aim to grow and scale rapidly.	14
T6	Time-pressure	The environment often forces startups to release fast and to work under constant pressure (terms sheets, demo days, investors' requests)	13
T7	Third party dependency	Due to lack of resources, to build their product, startups heavily rely on external solutions: External APIs, Open Source Software, outsourcing, COTS, etc.	10
T8	Small Team	Startups start with a small numbers of individuals.	9
T9	One product	Company's activities gravitate around one product/service only.	9
T10	Low-experienced team	A good part of the development team is formed by people with less than 5 years of experience and often recently graduated students.	8
T11	New company	The company has been recently created.	7
T12	Flat organization	Startups are usually founders-centric and everyone in the company has big responsibilities, with no need of high-management.	5
T13	Highly Risky	The failure rate of startups is extremely high.	5
T14	Not self-sustained	Especially in the early stage, startups need external funding to sustain their activities (Venture Capitalist, Angel Investments, Personal Funds, etc.).	3
T15	Little working history	The basis of an organizational culture is not present initially.	3

Taulukko 3.1: Teemoja, jotka kuvaavat ohjelmisto-startupia. [6]

Lista vahvistaa monen muun tutkielmani lähteen väitteitä. On havaittu, että ohjelmistoalan startup-yritykset käyttävät mielummin vähemmän aikaa suunnitteluun, jotta kehitysaika ennen tuotteen ensimmäistä julkaisua olisi mahdollisimman lyhyt ja

kilpailuetu kohdemarkkinoilla saavutettaisiin. Näin ollen pääpaino on kehitystyökalujen ja listassa mainittujen kolmansien osapuolten kirjastojen sekä komponenttien käytössä [12]. Tällainen poikkeaa hyvin paljon vakiintuneiden yritysten tavasta kehittää ohjelmistoa, jossa skaalautuvuus ja mahdollisimman pieni riippuvuus kolmansista osapuolista otetaan alusta lähtien huomioon. Aloittavilla startup-yrityksillä ei ole yksinkertaisesti aikaa tai varaa laadultaan tämänlaiseen ohjelmistotuotantoon. Validoinnin ja pääominaisuuksien kehityksen jälkeen startupeilla on enemmän aikaa keskittyä jatkokehitykseen, refaktorointiin ja skaalautuvuuteen. Edellä mainitut syyt tukevat listan havaintoja. Maininnat yrityksen nuoresta iästä, epävarmuudesta ja lyhyestä toimintahistoriasta ovat linjassa muiden tutkimusten kanssa.

Nykyaikainen ohjelmistokehitys on siirtynyt raskaista vesiputous-mallin kaltaisista prosesseista joustaviin ja kevyihin vaihtoehtoihin [17]. Vuonna 2001 julkaistun Ketterän ohjelmistokehityksen manifestin¹ jälkeen Lean-ohjelmisto-kehityksen ja monien muiden ketterän kehityksen menetelmien (esimerkiksi Extreme Programming, Scrum, Feature-Driven Development (FDD) ja Adaptive Software Development (ASD)) hyödyntäminen on kasvanut ohjelmistoalan yritysten ja startupien keskuudessa. Lean-periaatteet jakavat samankaltaisuuksia kahden muun periaatteen, Product Development Flow:n ja Kanbanin kanssa. Näistä Kanban on uusin lisäys Agilen ja Lean-menetelmien väliseen kiistaan. Näitä menetelmiä yhdistävät piirteet kuten lyhyet sprintin elinkaaret, nopea ja tiehä palautteen keruu asiakkaalta ja jatkuva oppiminen [4], [17].

Perinteisessä projektinhallinnassa *projektikolmio* (engl. *the golden triangle of PM*) havainnollistaa kolmen mittarin välisiä suhteita. Sen ulottuvuudet ovat *aika*, *kustannukset* sekä *laajuus*. Yhdessä ne määräävät projektin *laadun*. Projektipäällikön tavoitteena on pitää nämä keskenään kilpailevat ulottuvuudet mahdollisimman tasapainossa. [19] Kuvitellaan tilanne, jossa halutaan säästää projektiin käytettyä

¹<https://agilemanifesto.org/>

aikaa. Tahdin kiristäminen kasvattaisi kustannuksia tai vaihtoehtoisesti vaikuttaisi ominaisuuksien laajuuteen. Projektikolmiolla saadaan hyvin havainnollistettua tilanteita, joita startup-yritykset kohtaavat kehitystyössään. Aikataulusta ei haluta tinkiä, sillä pyrkimys on saattaa tuote markkinoille kilpailijoita nopeammin. Pyrkimys nopeuteen ja ketteryuteen näkyy näin joko laajuuden tai laadun heikentymisenä. Se ei ole pohjimmiltaan huono tilanne startup-yritykselle, sillä startupin tarkoituksena on testata hypoteesia ja oppia virheistä, joita tulee tehtyä matkan varrella. Siksi ominaisuuksia pudotetaan pois tärkeysjärjestyksessä aikaisemmassa luvussa esitetyn BML-syklin avulla.

3.1 Lean-ohjelmistokehitys

Lean-ohjelmistokehitys on lähestymistapa, jossa perinteisiä Lean-periaatteita ja työkaluja sovelletaan ohjelmistokehitykseen. Ketteriä- ja Lean-menetelmiä ei ole helppo erottaa toisistaan. Ketterän kehityksen ja Lean-periaatteiden vertailu paljastaa niiden olevan hyvin samankaltaisia [13]. Lean-ohjelmistokehitys täydentää Agilen oppeja lisäämällä alempana listattuja Lean-menetelmiä sekaan. Siksi se on kasvatanut suosiotaan ohjelmistokehityksen parissa. Nykyään Lean-ohjelmistokehitystä pidetään omana ketterästä kehityksestä eriteltyinä menetelmänä, joskin asiasta on myös eriäviä näkökulmia [13].

Alhaalla on listattuna yhdeksän Lean-metodologian periaatetta, jotka ovat oleellisia ohjelmistokehityksessä. Ne ovat tutkimuksista *Leagile software development: An experience report analysis of the application of lean approaches in agile software development* [4] ja *Using metrics in Agile and Lean Software Development – A systematic literature review of industrial studies* [13].

- minimoi hukka (Eliminate Waste)
- tehosta oppimista (Amplify Learning)

- luo tietoa (Create Knowledge)
- toteuta muutokset nopeasti (Deliver as Fast as Possible)
- anna tiimille valtaa (Empower the Team)
- kannusta laatuun (Build Quality In)
- optimoi kokonaisuus (Optimise the Whole)

Lean-ohjelmistokehityksen painopiste on *hukan* (engl. *waste*) poistamisessa. Työ luokitellaan Lean-ajattelussa arvoa lisääviin toimintoihin, pakollisiin arvoa lisäämättömiin toimintoihin ja arvoa lisäämättömiin toimintoihin. Japanin kielessä hukkaa kuvataan sanalla *muda*. Myös muita termejä, kuten *mura* (engl. *unevenness*) ja *muri* (engl. *overburden*), käytetään kuvaamaan eri tyyppistä hukkaa² [9]. Toiminnot, jotka eivät tuota arvoa on karsittava pois, sillä ne pelkästään kuluttavat resursseja eivätkä vie yritystä tavoitetta kohti [4]. Päätökseen merkata jokin ominaisuus arvoa tuottamattomaksi voivat vaikuttaa monet asiat, kuten tukeeko ominaisuus yrityksen liikeidean hypoteesia, tai mikä on uuden ominaisuuden tuoma hyöty käyttäjälle suhteessa sen kehittämiseen käytettyihin resursseihin. Analysointi on mahdollista viedä vielä pidemmälle pohtien taloudellisesta näkökulmasta, kuinka monta prosenttia kyseinen ominaisuus kasvattaisi yrityksen liikevaihtoa, jos se lisättäisiin tuotteeseen. Lähtökohtaisesti hyvä sääntö on jättää pois MVP:stä kaikki ominaisuudet, prosessit tai panostukset, jotka eivät edistä suoraan tavoiteltua oppimista [2].

Tarkan ja realistisen aika-arvion laatiminen tarvittavista resursseista ja työpanoksesta on haastavaa [9]. Ohjelmistokehityksessä hukkaa syntyy monesta syystä. Yleisiä syitä ovat viivästyminen, turhien tehtävien suorittaminen, suunnitteluvirheet, hyväksynnän odottaminen tai eriävät mielipiteet. Tämän seurauksena asetettuihin määräaikoihin on hankala päästä tai budjettia joudutaan kasvattamaan.

²Alan tutkimuksissa viitataan usein näihin japaninkielisiin termeihin hukasta.

Ohjelmistokehityksessä Lean-metodologian suurin arvo tulee kokonaiskuvan hahmottamisesta. On tärkeää, että yksittäiset ohjelmistokehittäjät ymmärtävät asiakkaiden tarpeita, ja miten he tulevat käyttämään palvelua. Ketterän kehityksen käyttäjätarinat pyrkivät hahmottamaan käyttötilanteita ja käyttäjän tarpeita. Käyttäjätarinoita tehokkaampaa on viettää aikaa asiakkaiden luona ja pyrkiä ymmärtämään heidän tarpeitansa. Startup-yrityksen perustamisvaiheessa työntekijöiden osallistaminen eri asioiden päätöksentekoon työtehtävästä riippumatta on tärkeää. Tämä menettelytapa on vastoin isompien organisaatioiden mallia, jossa jokaisesta sektorista päättää erikseen oma tiimi [20]. Siksi ensimmäiset työntekijät on hyvä valita tarkkaan, sillä heidän ja perustajajäsenten panostus sekä asenne heijastuvat myös tuleviin työntekijöihin.

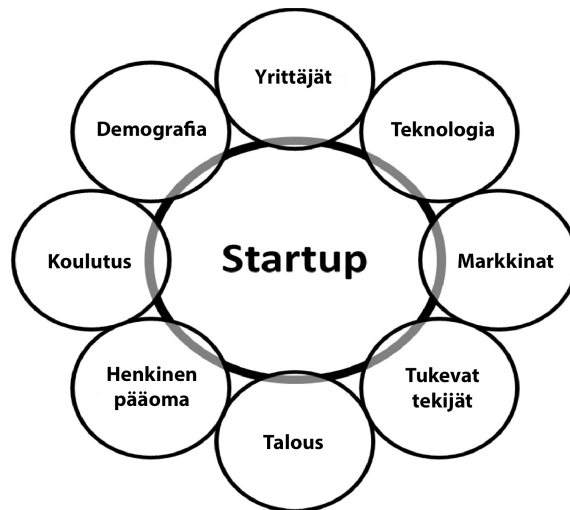
3.2 Tiimin merkitys ohjelmisto startup-yrityksen menestymisessä

Ohjelmistokehitys startup-yrityksessä on tiimin henkilökohtaisten taitojen ja aikaisempien kokemusten sekä jäsenten välisten suhteiden summa. Osaava ja tehokas tiimi voi tehdä huonosta ideasta hyvän tuotteen, mutta täynnä sisäisiä ristiriitoja olevalta tiimiltä tämä ei onnistu.

Alkuvaiheessa startup-yritysten tiimi koostuu usein 2–4 perustajajäsenestä, joilla on yrityksen liikeidean kannalta olennaista ydinosaamista. Usein kuultu mielipide startup-maailmassa toimivien keskuudessa on, että kolme jäsentä olisi ihanteellinen määrä perustajatiimille. Tämä ei sulje pois tiimin kasvua uusien työntekijöiden palkkaamisen myötä. Syy tälle mielipiteelle on perustajajäsenten keskeisen päätöksenteon vaikeutuminen ongelmatilanteissa. Lisäksi osakkuudet ovat jakautuneet useammalle henkilölle, ja on olemassa riski osakkuuksien jäämisestä erotilanteessa yrityksen jättävälle perustajajäsenelle. Tieteellistä tutkimusta, joka käsittelisi tiimin

koon vaikutusta yrityksen menestykseen ei ole saatavilla, joten on mahdollista että tämä on alalla tutkimusaukko, ja täten mahdollinen tutkimuskohde tulevaisuudessa.

Tiimillä on väliä yrityksen menestyksen kannalta. Startup-yrityksen ekosysteemi voidaan jakaa kahdeksaan pääosaan, jotka ovat jaoteltuna kuvassa 3.1 [12]. *Talouden, teknologian, taustatekijöiden ja markkinoiden* lisäksi voidaan havaita, että puolet startup-ekosysteemin elementeistä (*koulutus, demografia, yrittäjät ja henkinen pääoma*) ovat tiimistä riippuvaisia.



Kuva 3.1: Startup-ekosysteemin pääelementit [16]

Perustajajäsenten vaihtelevan koulutustaustan on todettu olevan myönteinen asia myös pääomasijoittajien silmissä [21]. Toinen tutkimus vahvistaa, että aikaisempaa yrittäjyys- ja johtamiskokemusta omaavat tiimit tunnistivat enemmän potentiaalisia liikeideoita. Lisäksi havaittujen liikeideoiden kokonaismäärällä näyttäisi olevan korrelaatio uusien yritysten menestykseen [22]. Koulutukseltaan monimuotoinen tiimi ei ole kuitenkaan kaikissa tilanteissa suotuisin vaihtoehto. On havaittu, että homogeeniset – teknisistä osaajista koostuvat – tiimit ovat paras vaihtoehto deep-tech pohjaisille liikeideoille. Perustajatiimin jäsenten kykyjen moninaisuus muodostaa käänteisen u-käyrän suhteessa startupin suorituskykyyn [23]. Monimuotoisesta tiimistä on hyötyä kilpailevilla markkinoilla, mutta ei niinkään yhteistyöhön

perustuvissa tilanteissa tai hyvin teknisissä tiimeissä [21]. Nuorilla perustajilla on usein ajantasaisin tieto viimeisimmästä teknologiasta ja into oppia uutta, mikä on etu verrattuna vakiintuneisiin yrityksiin.

Tuotteen toteutus lähtee yksittäisistä tiimin jäsenistä, heidän ajallisesta-, tietotaidollisesta- sekä taloudellisesta panostuksestaan. Tiimin jäsenten välisillä sosiaalisilla suhteilla on todettu olevan myönteinen vaikutus startupiin. Luottamus ja sitoutuminen yritykseen vaikuttavat sen mahdollisuuteen menestyä alussa ja myöhemmin kasvuvaiheessa [24]. Psykologinen turvallisuus tiimissä on avainasemassa avointa ilmapiiriä luodessa. Psykologisella turvallisuudella tarkoitetaan tiimin jäsenten jaettua uskomusta siitä, että tiimi on turvallinen henkilöiden väliselle riskinotolle [25]. Korkean psykologisen turvallisuuden omaavan tiimin jäsenet tuntevat olonsa turvaliseksi, pyytävät apua toisilta, tuovat esille epäkohtia ja ottavat herkemmin riskejä. Lisäksi on olemassa tutkimuksia [26], joissa psykologinen turvallisuus liitetään suorituskyykyyn joko suoraan tai epäsuorasti [25]. Tutkimukset osoittavat, että vastaavanlaisti psykologisen turvallisuuden puute tiimissä lisää taipumusta virheiden pimentämiseksi sekä lisää epäröintiä ilmoittaa asioista, jotka saattavat vaikuttaa negatiivisesti tiimin suorituskyykyyn [25]. Ideoiden rakentava haastaminen luo terveen ympäristön, jossa uskalletaan tuoda oma mielipide esille tarpeeksi varhaisessa vaiheessa. On tärkeää ottaa mukaan jokainen tiimin jäsen ajattelutyöhön alusta lähtien. Se sitouttaa jäsenet työskentelemään yrityksen eteen ja tuo näkökulmia eri aloilta ja lähtökohdista.

On yleistä, että perustajajäsenten kokemus keskittyy tiettyyn liiketoiminta-alueeseen [27]. Tämän kääntöpuolena on riski joutua saatavuusharhan alaiseksi. Ohjelmistokehityksen parissa toimiva henkilö havaitsee herkemmin kyseiseen alaan liittyviä liikeideoita kuin yrittäjä, jolla on tausta myynnissä. Demografisesti ja taustoiltaan liian homogeeninen tiimi altistaa ymmärrettävästi vahvistusharhoille [21].

Tästä syystä on hyödyllistä hakea neuvoja ja palautetta myös tiimin ulkopuolelta. Kokeneet yrittäjät ja sijoittajat antavat usein mielellään neuvoja kysyttäessä.

Vinoumista voi siis päästä eroon. Kokemattomat yrittäjät voivat tehdä riskialttiita oletuksia ja potentiaalisille asiakkaille puhuminen on yksi tehokkaimmista keinoista testata hypoteesia. Teoksessaan [2] Ries kuvailee esimerkkinä tilannetta, jossa he antoivat IMVU nimisen palvelunsa testattavaksi kohderyhmään kuuluvalla henkilölle, ja tämä tyrmäsi heidän alkuperäisen hypoteesinsa: ”Meillä oli vuosia vanhentunut ajatusmalli siitä, miten ihmiset käyttivät ohjelmistoja, ja lopulta, tuskaisesti, kymmenien tällaisten kokousten jälkeen meille alkoi valjeta, että IM-lisäosakonsepti oli pohjimmiltaan virheellinen.”³

Startupit toimivat usein hyvin rajallisilla taloudellisilla- ja henkilöstöresursseilla. Tämä pätee etenkin aloittaviin startup-yrityksiin, joiden tiimikoot ovat vielä pieniä. Startup-yritysten johtamis- ja organisaatiokäytännöt keskittyvät olennaiseen, jolloin myös byrokratian määrä on pientä. Tämä antaa yksittäiselle kehittäjälle ja tiimin jäsenelle laajalti valtaa ja kannustaa sopeutumaan useisiin rooleihin ja vastuualueisiin näin vaikuttaen luovasti lopputuotteeseen. [6] Näissä tilanteissa hukan minimointi ja ominaisuuksien priorisointi ovat kriittisiä yrityksen selviämisen kannalta.

Startup-yritykset elävät kyvykkyydestä nopeisiin ja mahdollisimman kevyisiin suunnan muutoksiin. Muussa tapauksessa yritys jää jälkeen kilpailijoistaan ja alun kilpailuetu on menetetty. Osaavilla ohjelmistokehittäjillä on merkittävä rooli teknologiastartupin menestyksessä. Tiimien haasteena on houkutella mukaan näitä tärkeitä kehittäjiä ja saada heidät pysymään yrityksessä [5]. Osaavat ohjelmistokehittäjät tunnistavat arvonsa ja ilman rahoitusta ainoaksi keinoksi saada osaavia tekijöitä mukaan on myydä yrityksen liikeidea tarpeeksi hyvin ja ottaa heidät osakkaiksi.

³IMVU on virtuaalinen 3D-yhteisöpalvelu, jonka Ries perusti vuonna 2004. <https://about.imvu.com/> IM on lyhenne englannin kielen sanalle *instant messaging*.

4 Yhteenveto

Startup-yritykset ja pienet tiimit eroavat merkittävästi vakiintuneiden yritysten toimintamekanismeista. Epävakaa toimintaympäristö ja elinkaari luovat haasteita taasisen ohjelmistokehityksen toteuttamiselle. Monet ketterän kehityksen menetelmät, jotka usein vaativat selkeästi määritetyn ongelman, eivät tuota riittävästi hyötyä pienille startup-tiimeille. Usein niiden ongelma ja ratkaisu ovat molemmat epäselviä. Startup-yritykset ovat toisinaan valmiita hyödyntämään valikoidusti ketterän kehityksen työkaluja ja soveltavat ne osaksi omaa kehitysprosessiaan.

Yksi tutkielman pääkäsitteistä on Lean Startup -metodologia, joka käsittelee tehokkuutta ja ketteryyttä. Se keskittyy varhaisessa vaiheessa startup-yrityksen elinkaarta asiakkaan ongelman validointiin sekä olennaisten ominaisuuksien testaamiseen ja kehittämiseen. Validoitu oppiminen varmistaa jatkuvan oppimisen ja Rakenna-mittaa-opi-palautesykli liiketoimintaoletuksen jakamisen tarpeeksi pieniin testattaviin osiin. Syklin avulla startup-yritys voi validoida liikeidea käyttäjäpalautteen pohjalta ja tehdä päätöksen jatkamisesta eteenpäin tai idean pivotoinnista. Lean Startup -metodologian toinen etu ohjelmistokehityksessä tulee sen pyrkimyksestä vähentää hukkaa. Painopiste on nopeassa oppimisessa, jatkuvassa iteroinnissa ja resurssien allokoinnissa merkityksellisiin ominaisuuksiin.

Tässä tutkielmassa käsitellään Lean Startup -metodologian hyödyntämistä teknologia-startupin ohjelmistokehityksessä. Tutkielmassa on kaksi tutkimuskysymystä, joista ensimmäinen on: ”Miten Lean-metodologian avulla voidaan te-

hostaa startup-yrityksen ohjelmistokehitystä?” Tutkielmasta käy ilmi, että Lean-metodologiasta on hyötyä startup-yrityksen ohjelmistokehityksessä. Idean validointi mahdollistaa ominaisuuksien jaottelun, joko arvoa tuottavaksi tai arvoa tuottamattomaksi nopeuttaen kehitystyötä ja vähentäen hukkaa. Tämän lisäksi Rakennamittaa-opi-palautesyklin käyttö auttaa nimensä mukaisesti ideoiden rakentamisessa ja testaamisen kautta saadun palautteen sisäistämässä. Sykli helpottaa päätöksentekoa tuotteen tai yksittäisen ominaisuuden jatkamisessa tai hylkäämisessä. Toinen tutkimuskysymyksen on: ”Miten tiimin kokoonpano vaikuttaa ohjelmistokehitykseen?” Startupin perustajajäsenten sosiaalisilla suhteilla on merkitystä myöhemmän menestyksen kannalta. [24] Psykologisella turvallisuudella on iso merkitys epäkohtien ja ongelmien esille tuonnissa. Vastakohtana on tiimi, jonka sisällä ongelmia pimitetään ja pyritään tuomaan esille vain hyviä uutisia. Tällaiset startup-yritykset eivät pärjää kovin pitkälle. Molempiin tutkimuskysymyksiin löytyi riittävästi ajankohtaista tutkimusaineistoa ja niihin on vastattu riittävällä tarkkuudella sekä laajuudella.

Ohjelmistoalan startup-yritykset ovat osa uniikkia ja kiehtovaa ympäristöä, joka on ollut pitkään tutkimusalana aliedustettu [5], [6]. Lean-menetelmät ovat kehittyneet ja muuttaneet muotoaan viimeisen 20 vuoden aikana. Tutkielmassani ei käsitellä niiden tulevaisuutta tai mihin alan tutkimus sekä käytännöt ovat menossa. Tämä voisi olla yksi aihe, jolla tutkielmaa jatkettaisiin. Aiheesta julkaistaan jatkuvasti uutta tutkimusta ja startup-maailma on myös yleisesti noussut suuren yleisön tietoisuuteen viime vuosina, niin globaalisti kuin myös Suomessa. Toinen mahdollinen jatkotutkimuksen kohde on tiimin koon vaikutus startupin menestykseen, sillä niiden korrelaatiosta ei löytynyt vahvaa näyttöä.

Lean Startup -metodologian avulla teknologia-startupit voivat hyödyntää resurssejaan tehokkaammin ja nopeuttaa tuotteen saattamista markkinoille. Oikean ongelman ratkaisuun keskittyminen ja hypoteesien testaaminen potentiaalisilla asiakkailla vähentävät merkittävästi hukan määrää. Ohjelmistokehityksessä se tarkoittaa

säästettyä aikaa, henkilöstöressurssien tehokasta käyttöä ja vain tarvittavien ominaisuuksien kehitystä.

Lähdeluettelo

- [1] Atomico ja Slush. ”The State of European Tech 2021”, viitattu 27. toukokuuta 2025. url: <https://2021.stateofeuropeantech.com/chapter/europe-global-tech-force/article/kicking-full-gear/>.
- [2] E. Ries, *The Lean Startup: How Today’s Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Crown/Currency, 2011, ISBN: 978-0-307-88791-7.
- [3] S. Blank, ”Why the Lean Start-Up Changes Everything.”, *Harvard Business Review*, vol. 91, 5 2013, ISSN: 0017-8012. url: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84876555812&partnerID=40&md5=bd42e6abe554e3a5089d7e6cf48ff490>.
- [4] X. Wang, K. Conboy ja O. Cawley, ”“Leagile” software development: An experience report analysis of the application of lean approaches in agile software development”, *Journal of Systems and Software*, vol. 85, s. 1287–1299, 6 2012, ISSN: 0164-1212. DOI: 10.1016/j.jss.2012.01.061.
- [5] S. Sutton, ”Role of process in a software start-up”, *IEEE Software*, vol. 17, nro 4, s. 33–39, 2000. DOI: 10.1109/52.854066.
- [6] N. Paternoster, C. Giardino, M. Unterkalmsteiner, T. Gorschek ja P. Abrahamsson, ”Software development in startup companies: A systematic mapping study”, *Information and Software Technology*, vol. 56, nro 10, s. 1200–1218, 2014, ISSN: 0950-5849. DOI: 10.1016/j.infsof.2014.04.014.

-
- [7] H. Edison, N. M. Smørsgård, X. Wang ja P. Abrahamsson, "Lean Internal Startups for Software Product Innovation in Large Companies: Enablers and Inhibitors", *Journal of Systems and Software*, vol. 135, s. 69–87, 2018, ISSN: 0164-1212. DOI: 10.1016/j.jss.2017.09.034.
- [8] D. Powell ja M. Oliveira, "Insights from a Digital Lean Startup: Co-creating Digital Tools for Cognitive Augmentation of the Worker", *Procedia CIRP*, vol. 104, s. 1384–1388, 2021, ISSN: 2212-8271. DOI: 10.1016/j.procir.2021.11.233.
- [9] A. R ja N. G. Cholli, "Application of Lean Principles in Software Development Processes", teoksessa *2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, vol. 1, 2021, s. 38–42, ISBN: 978-1-6654-0017-6. DOI: 10.1109/CENTCON52345.2021.9687927.
- [10] M. C.W., "The delusion of intrapreneurship", *Long Range Planning*, vol. 19, s. 92–95, 6 1986, ISSN: 00246301. DOI: 10.1016/0024-6301(86)90101-9.
- [11] C. Nobel ja S. Ghost. "Why Companies Fail—and How Their Founders Can Bounce Back". H. B. School, toim., viitattu 22. toukokuuta 2025. url: <https://www.library.hbs.edu/working-knowledge/why-companies-failand-how-their-founders-can-bounce-back>.
- [12] N. Tripathi, M. Oivo, K. Liukkunen ja J. Markkula, "Startup ecosystem effect on minimum viable product development in software startups", *Information and Software Technology*, vol. 114, s. 77–91, 2019, ISSN: 0950-5849. DOI: 10.1016/j.infsof.2019.06.008.
- [13] E. Kupiainen, M. V. Mäntylä ja J. Itkonen, "Using metrics in Agile and Lean Software Development – A systematic literature review of industrial studies",

- Information and Software Technology*, vol. 62, s. 143–163, 2015, ISSN: 0950-5849. DOI: 10.1016/j.infsof.2015.02.005.
- [14] M. Zorzetti, I. Signoretti, L. Salerno, S. Marczak ja R. Bastos, ”Improving Agile Software Development using User-Centered Design and Lean Startup”, *Information and Software Technology*, vol. 141, s. 106718, 2022, ISSN: 0950-5849. DOI: 10.1016/j.infsof.2021.106718.
- [15] S. S. Bajwa, X. Wang, A. N. Duc ja P. Abrahamsson, ”“Failures” to be celebrated: an analysis of major pivots of software startups”, *Empirical Software Engineering*, vol. 22, nro 5, s. 2373–2408, 2017. DOI: 10.1007/s10664-016-9458-0.
- [16] H. Terho, S. Suonsyrjä, A. Karisalo ja T. Mikkonen, ”Ways to Cross the Rubicon: Pivoting in Software Startups”, teoksessa *Product-Focused Software Process Improvement*, P. Abrahamsson, L. Corral, M. Oivo ja B. Russo, toim., vol. 9459, Springer, Cham, 2016, s. 555–568, ISBN: 978-3-319-26844-6. DOI: 10.1007/978-3-319-26844-6_41.
- [17] P. Rodríguez, M. Mäntylä, M. Oivo, L. E. Lwakatare, P. Seppänen ja P. Kuvaja, ”Chapter Four - Advances in Using Agile and Lean Processes for Software Development”, teoksessa *Advances in Computers*, A. M. Memon, toim., vol. 113, 2019, s. 135–224, ISBN: 9780128160701. DOI: 10.1016/bs.adcom.2018.03.014.
- [18] K.-K. Kemell, A. Nguyen-Duc, M. Suoranta ja P. Abrahamsson, ”StartCards — A method for early-stage software startups”, *Information and Software Technology*, vol. 160, s. 107224, 2023, ISSN: 0950-5849. DOI: <https://doi.org/10.1016/j.infsof.2023.107224>.
- [19] M. L. Drury-Grogan, ”Performance on agile teams: Relating iteration objectives and critical decisions to project management success factors”, *Information*

- and Software Technology*, vol. 56, nro 5, s. 506–515, 2014, ISSN: 0950-5849. DOI: 10.1016/j.infsof.2013.11.003.
- [20] G. Coleman ja R. V. O'Connor, "An investigation into software development process formation in software start-ups", *Journal of enterprise information management*, vol. 21, s. 633–648, 6 2008, ISSN: 17410398. DOI: 10.1108/17410390810911221.
- [21] D. A. Shepherd, V. Souitaris ja M. Gruber, "Creating New Ventures: A Review and Research Agenda", *Journal of Management*, vol. 47, s. 11–42, 1 2021. DOI: 10.1177/0149206319900537.
- [22] M. Gruber, I. C. MacMillan ja J. D. Thompson, "From Minds to Markets: How Human Capital Endowments Shape Market Opportunity Identification of Technology Start-Ups", *Journal of Management*, vol. 38, s. 1421–1449, 5 2012, ISSN: 0149-2063. DOI: 10.1177/0149206310386228.
- [23] S. Hoogendoorn, S. C. Parker ja M. van Praag, "Smart or Diverse Start-up Teams? Evidence from a Field Experiment", *Organization Science*, vol. 28, s. 1010–1028, 6 2017. DOI: 10.1287/orsc.2017.1158.
- [24] C.-J. Wang ja L.-Y. Wu, "Team member commitments and start-up competitiveness", *Journal of Business Research*, vol. 65, s. 708–715, 5 2012, ISSN: 0148-2963. DOI: 10.1016/j.jbusres.2011.04.004.
- [25] A. Edmondson, "Psychological Safety and Learning Behavior in Work Teams", *Administrative Science Quarterly*, vol. 44, s. 350–383, 2 1999, ISSN: 2212-8271. DOI: 10.2307/2666999.
- [26] L. Barros, C. Tam ja J. Varajão, "Agile software development projects—Unveiling the human-related critical success factors", *Information and Software Technology*, vol. 170, 6 2024, ISSN: 0950-5849. DOI: 10.1016/j.infsof.2024.107432.

-
- [27] A. Ardichvili, R. Cardozo ja S. Ray, "A theory of entrepreneurial opportunity identification and development", *Journal of Business Venturing*, vol. 18, s. 105–123, 1 2003, ISSN: 0883-9026. DOI: 10.1016/S0883-9026(01)00068-4.