Aalto University
School of Science
Master's Programme in Information Networks

Emmi Linkola

Inspired by The Lean Startup – Introducing Experimentation to Improve the New Product Development Process of an IT Service Company in the Social and Healthcare Sector

Master's Thesis
Espoo, February 10, 2019

Supervisor: Marjo Kauppinen, Professor
Thesis advisor: Tuomas Järvensivu, MS.c. (Tech.)

AALTO UNIVERSITY
School of Science
Master's Programme in Information Networks

ABSTRACT OF THE
MASTER'S THESIS

Author: Emmi Linkola

Title of the thesis:

Inspired by The Lean Startup – Introducing Experimentation to Improve the New Product Development Process for an IT Service Company in the Social and Healthcare Sector

Number of pages: v + 96

Date: 10. February 2019

Major: ICT in Business

Supervisor: Marjo Kauppinen

Thesis advisor: Tuomas Järvensivu

Managing risks and ensuring that development time is spent on value-creating activities are critical concerns for companies operating in unpredictable environments. Experimentation is a method that can help such companies make data-driven decisions instead of relying on guesswork. The case company of this thesis was inspired by the method introduced in the book The Lean Startup. However, they found implementing it challenging due to the environment they operate in, where customer acquisition costs are high and developing solutions requires deep understanding of customer processes.

The research problem that this thesis addresses is: how can experimentation be utilized to improve the new product development process in the case company? The research method utilized was action research, which consisted of mapping out the challenges in the current process through interviews and iteratively developing a proposal for an experimentation process.

The most notable challenges in the current new product development process were related to how the project in question was sold, what the perceived goals of the project were, how work was estimated, and a lack of user research. The proposed experimentation process aims to help the team achieve a shared understanding of the goals of each experimentation project and to align them with company-wide goals. It also addresses the lack of user research by incorporating it into the process.

This thesis provides implementation details on how to experiment at the case company. The case company should start experimentation with background research, acquiring a pilot customer and conducting user research before developing possible solutions.

Keywords: Lean Startup, experimentation, new product development

Publishing language: English

AALTO-YLIOPISTO
Perustieteiden korkeakoulu DIPLOMITYÖN
Informaatioverkostojen koulutusohjelma TIIVISTELMÄ

Tekijä: Emmi Linkola	
Työn nimi: Eksperimentaation käyttöönotto The Lean Startupin innoittamana uusien tuotteiden kehitysprosessin parantamiseksi IT-palveluyrityksessä sosiaali- ja terveysalalla	
Sivumäärä: v + 96	**Päiväys**: 10. Helmikuuta 2019
Pääaine: ICT in Business	
Valvoja: Marjo Kauppinen **Työn ohjaaja**: Tuomas Järvensivu	

Riskien hallinta ja kehitystyön keskittäminen lisäarvoa tuottaviin tehtäviin ovat äärimmäisen tärkeitä tavoitteita yrityksille, jotka toimivat ennalta-arvaamattomissa ympäristöissä. Eksperimentaatio on menetelmä, joka voi auttaa tämänlaisia yrityksiä tekemään päätöksiä kerättyyn tietoon pohjautuen arvailun sijaan. Yritys, jota tämä diplomityö käsittelee, kiinnostui The Lean Startup-kirjassa esitellystä menetelmästä. Yritys kuitenkin huomasi menetelmän toteuttamisen olevan haastavaa omassa toimintaympäristössään, jossa asiakkaiden hankinta on kallista ja ratkaisujen kehittäminen vaatii asiakkaan prosessien syvää ymmärrystä.

Tämän diplomityön tutkimusongelma on: miten kyseenomaisen yrityksen uusien tuotteiden kehitysprosessia voidaan parantaa eksperimentaatiota hyödyntämällä? Tutkimusmenetelmänä oli toimintatutkimus, joka koostui yrityksen nykyisen uusien tuotteiden kehitysprosessin haasteiden kartoittamisesta haastattelujen avulla ja ehdotetun eksperimentaatioprosessin kehittämisestä.

Merkittävimmät haasteet yrityksen nykyisessä uusien tuotteiden kehitysprosessissa liittyivät kyseisen projektin myyntiin, projektin miellettyihin tavoitteisiin, työmäärän arviointiin, ja käyttäjätutkimuksen puutteeseen. Ehdotettu eksperimentaatioprosessi pyrkii auttamaan työryhmää saavuttamaan yhteisen ymmärryksen kunkin eksperimentaatioprojektin tavoitteista ja siitä, miten ne ovat yhteydessä koko yrityksen tavoitteisiin. Se myös lisää käyttäjätutkimusta prosessiin.

Tämä diplomityö tarjoaa käytännön neuvoja eksperimentointiin kyseisessä yrityksessä. Tämän yrityksen tulisi aloittaa eksperimentaatio taustatutkimuksella, pilottiasiakkaan hankinnalla ja käyttäjätutkimuksen teolla ennen mahdollisten ratkaisujen kehittämistä.

Asiasanat: Lean Startup, eksperimentaatio, tuotekehitys	
Kieli: Englanti	

Contents

1 Introduction

1.1 Motivation

Operating in a highly uncertain business environment introduces a set of challenges that cannot be solved through traditional management and planning. Regardless of how rigorous the planning is and how well executed the plans are, success will not follow unless the planned product has a good product-market fit – and this is exceptionally rarely the case on the first try. Thus, uncertain business environments require an iterative and experimentation-oriented manner of working in order to determine what works in the target market (e.g. Sarasvathy, 2001). Continuous experimentation is a practice aimed at addressing such uncertainty. The Lean Startup, introduced by Eric Ries (2011), is one popular experimentation method from business literature. It aims to make the odds of succeeding better through continuous, validated learning, iterating with the product and the business model, and measuring long-term progress. It helps companies deal with uncertainty through validating the underlying hypotheses of the business early and cheaply. The case company that this thesis focuses on was inspired by Ries' book to test experimentation methods in their own operations. Ries (2011) defines a startup as any company that operates in an environment characterized by high uncertainty. The business of the case company that this research paper focuses on indeed operates in an uncertain domain: the development of IT services, especially – but not limited to – payment services, for the Finnish social and healthcare sector. As there is no proven path to follow, the case company can benefit from investigating whether utilizing experimentation methods such as The Lean Startup could improve their new product development process. Validating the most important hypotheses that the products are based on ensures that the products evolve into the right direction and time is spent developing the things that take the company toward sustainable growth. Making the process data-driven and measurable through an experimentation process could help the case company follow whether the business is developing in a sustainable manner.

This thesis will explore the possibilities of utilizing The Lean Startup to ensure that the new product development team at the case company is making real progress with new products. It attempts to make the case company's product development process more efficient and reliable through being data-driven. The Lean Startup explains how to experiment in environments where new customers can easily be acquired. However, different new product development strategies are appropriate for different situations, whereas Ries only presents a supposedly one-size-fits-all method (Hauser et al., 2006, cited in Frederiksen and Brem, 2017). Symptoms of the lack of implementation details for different situations can be found by looking at the current state of the practice in the industry: experimentation in practice is often not systematic and continuous (Lindgren

and Münch, 2016). This thesis aims to help the case company avoid the known pitfalls and build a systematic, continuous experimentation process. It also attempts to provide guidance on how experimentation can best be practiced in the particular situation of the case company, which operates in a business-to-government domain, where customer acquisition costs are high and the customer processes that the case company supports are complex.

1.2 Research problem and questions

The research problem in this thesis is as follows:

How can experimentation be utilized to improve the new product development process in the case company?

Furthermore, the research problem is complemented by the following research questions:

RQ1: What are the challenges in the current new product development process of the case company?

RQ2: What experimentation process can be formulated to improve the current new product development process?

RQ3: What suggestions can be given to the case company in order to systematically keep improving the experimentation process?

1.3 Scope of the thesis

The case company is first and foremost an IT service development company, and therefore the literature review section of this thesis is focused on the context of software development. In addition, this thesis focuses on experimentation processes, leaving other themes that The Lean Startup explores – such as the entrepreneurial mindset of doing instead of planning – on lesser focus. While related, this is another topic entirely in scientific literature, related to e.g. effectuation and systems thinking. To keep the focus concise, this paper keeps the spotlight solely on the process of experimentation.

Business analytics are also a topic that has been written about extensively. The metrics that this thesis explores in more detail are related to growth and innovation. Other types of business analytics will not be explained in depth.

1.4 Structure of the thesis

Figure 1.1 illustrates how the thesis is structured. It shows, which chapters address which of the research questions. The first two chapters, introduction and research methods,

address all of the research questions. The literature study is focused on experimentation, and therefore addresses research questions two and three. The first part of the empirical study introduces results related to research question one, while the second chapter examines results related to research question two. The third part of the empirical study mainly reviews results related to research question three, but also adds to the answer to research question two. The discussion chapter is divided into three parts, each of which address one of the research questions, and a fourth chapter on limitations related to all of them. Finally, the conclusions bring together findings related to all of the research questions.

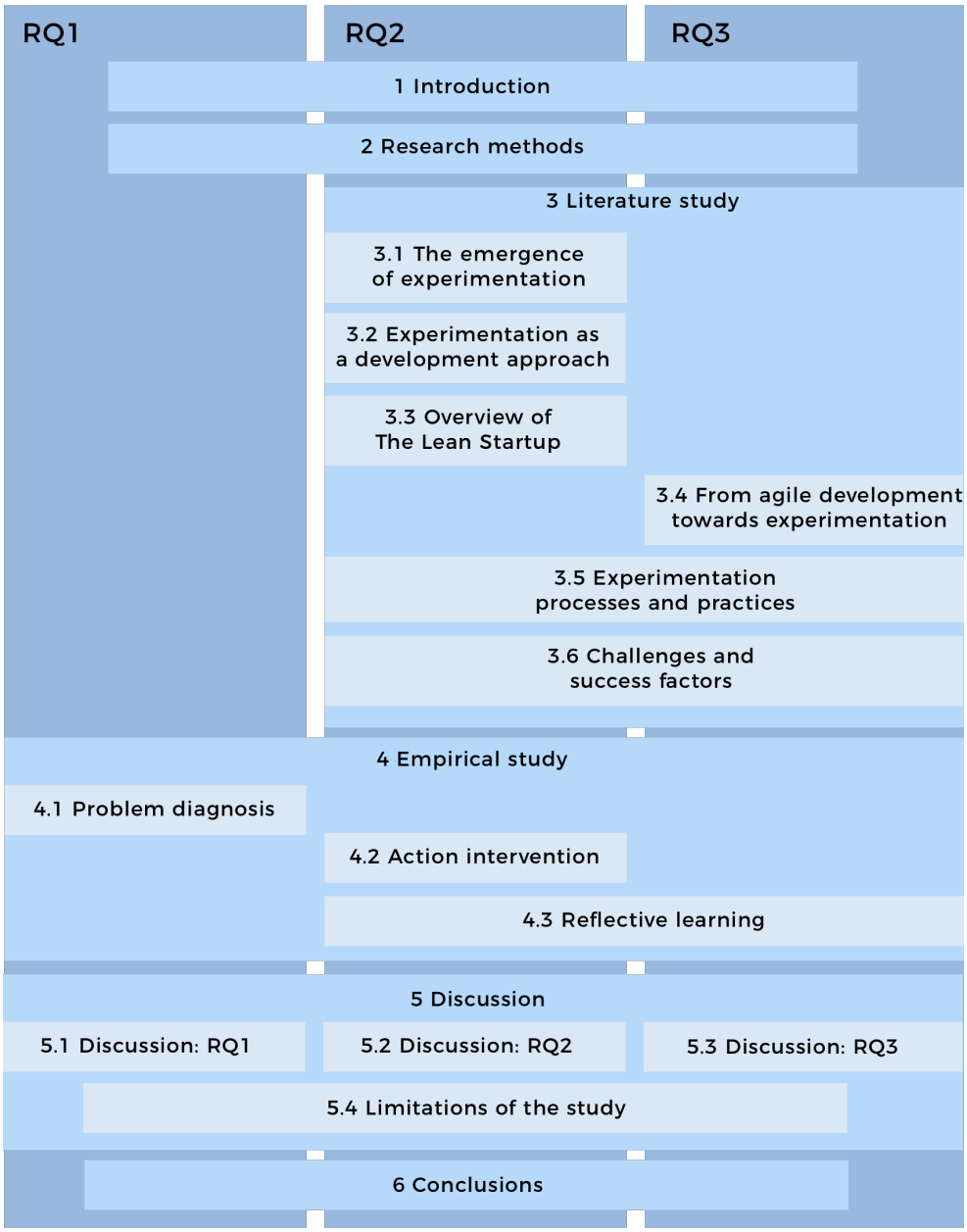


Figure 1.1. The structure of the thesis.

2 Research methods

2.1 Literature review

The author started the literature review with getting to know The Lean Startup and The Startup Way by Eric Ries. Next, she searched for evaluations of The Lean Startup in scientific source material. Based on what she had learned so far, she broke down The Lean Startup into themes and practices and delved deeper into each of them. This revealed that there was a plethora of literature on many of the topics, such as startup performance measurement and continuous experimentation. Furthermore, there is much more literature on continuous experimentation than there is literature that explicitly addresses The Lean Startup. For the case company, many of the guidelines that are not directly included in The Lean Startup proved to be very interesting. For example, Ries (2011) does not provide guidelines to transitioning towards a continuous experimentation process or what kind of infrastructure to build for an experimentation system. In addition, Ries (2011) focuses mainly on practices that are optimal for business-to-consumer companies, which led to a gap between practices that Ries recommends and practices that are useful for the case company, as the main customers of the case company are municipalities. Therefore, the author decided to keep The Lean Startup as the inspiration of the research but not limit the practices considered for implementation to those included in Ries' (2011) book.

2.2 Empirical study

2.2.1 Case description

The case company is a small Finnish company that makes IT services for the social and healthcare sector. The company has less than 20 employees, whose tasks range from customer service to software development. The main business of the company is making different payment services for the social and welfare sector, but they also make IT services for the healthcare reform in Finland. The case company was formed as a subsidiary in 2017, and their existing payment services were developed while the company was still part of the parent company. Therefore, the first own new product development project of the case company was developing services used in a pilot for the social and healthcare reform.

The payment services are an already established business for the case company. However, the case company wants to explore extending their services to cover different payment models, as it currently only covers one payment model. In addition, the case company is interested in determining whether they could operate as a platform for a network to which other companies and parties operating in the social and healthcare

sector could connect to. The case company is also considering utilizing new technologies to make their solutions more secure.

The case company has also done a project where they developed IT services for a pilot regarding the social and healthcare reform in Finland for a federation of municipalities. The project is still ongoing, but the work is mostly maintaining the system and doing small additional development. Since this project, the case company has focused more on building own products, rather than custom solutions for customers.

2.2.2 Research process

The empirical research was conducted as action research. Action research consists of a problem diagnosis, after which follows acting to the problems that were found by formulating a solution in a phase called action intervention, and evaluating and improving the solution in a phase called reflective learning.

Action research is a qualitative research method in which research and practice are particularly intertwined (Avison et al., 1999). It consists of researchers and practitioners working together to diagnose problems, act on them, and learn from the results (Avison et al., 1999). The research process itself is often the same as its outcome in action research (Patton, 2002).

In action research, the researcher is involved in the work with the company they are researching. This helps the researcher modify theories in order to make them more appropriate based on the feedback from the practical implementation (Avison et al., 1999). It gives researchers the chance to observe not only what people say they do, but also how they act (Avison et al., 1999). In this case, the researcher was working as a consultant in the software development team at the case company, which gave them the opportunity to understand the processes of the case company through participation and modify the suggested experimentation process based on planning the practical implementation of the process with the case company.

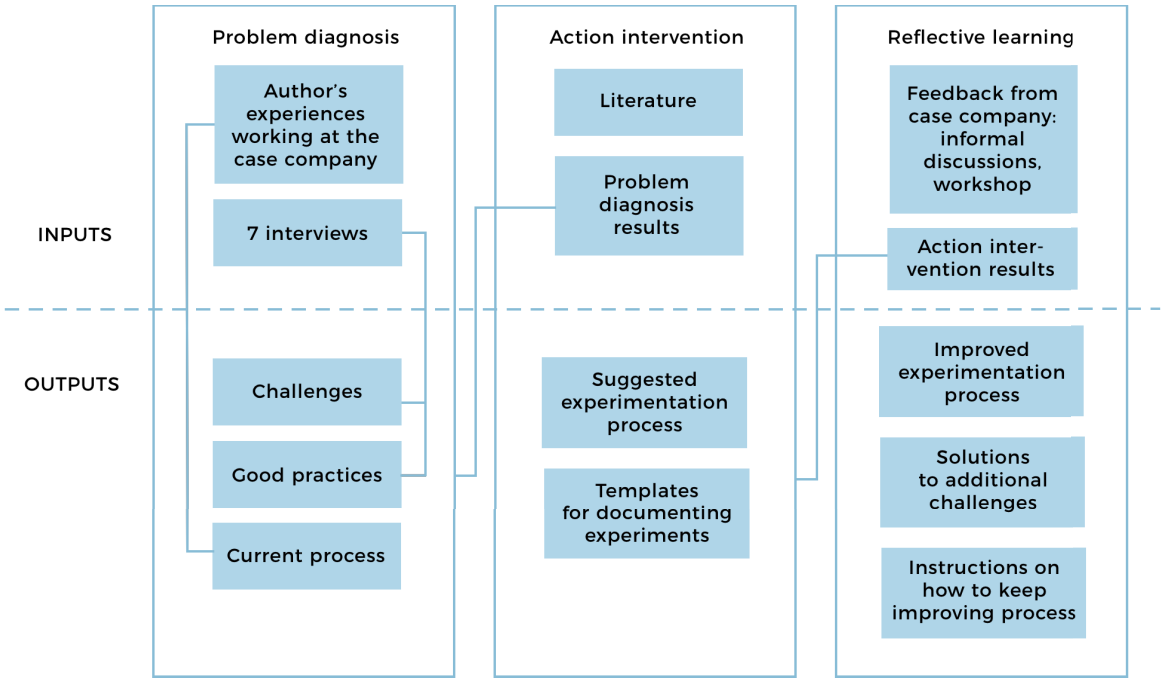


Figure 2.1. The empirical research process used in this thesis in the context of the action research framework.

The empirical research process in this thesis consisted of several parts. First, interviews were conducted to identify problems in the current new product development process. Based on the results of these interviews, an experimentation process was formulated utilizing the findings from literature to best answer the specific needs of the case company. The process was introduced to the case company and iterated on based on feedback from the case company to better suit the particular needs of the case company.

Figure 3.1 presents the action research process that was used in this thesis. It displays the sources and outcomes of each action research phase. The sources for the problem diagnosis were the author's own experiences working in the case company, as well as interviews of 7 people including both employees and customers of the case company. The outcomes were a general view of the current new product development process of the case company, as well as challenges and good practices related to it. These findings were combined with literature in the action intervention phase in order to formulate an experimentation process that can be suggested to the case company, as well as templates that guide and document the process. These, in turn, were utilized in the reflective learning part, where they were evaluated with the case company and improvements were formulated, along with instructions for how to keep systematically improving the process. Since the experimentation process can not solve all the challenges found in the problem diagnosis phase, solutions to them were also formulated separately.

2.2.3 Data collection and analysis

This chapter is divided into subchapters based on the phases of the action research that was conducted. First, we examine the data collection and analysis methods in the problem diagnosis phase, then in the action intervention phase, and finally the reflective learning phase.

Problem diagnosis

The problem diagnosis was implemented as a set of semi-structured interviews of employees of the case company that had taken part in a previous new product development project, as well as three representatives of the customer. The internal and external interviewees were given a slightly different set of questions. The interview templates are included in appendices A and B. The internal interview included a set of 19 open-ended questions, while the external interview had 18 questions. The pre-defined questions were complemented with possible additional questions on the same themes where deemed suitable by the researcher. This semi-structured interview approach allows for understanding the interviewee better and helps interpret what they say (Järvinen, 2004), and was chosen in order to allow the responses to affect the direction of the interviews. Especially the last question was formatted to bring up other topics the respondents felt important. As recommended by Patton (2002), leading questions were avoided, and emphasis was on letting the interviewee describe what, how and why things had happened during the previous project. The questions were worded to be truly open-ended, not providing the respondent with e.g. an implied scale of feeling or thought (Patton, 2002). The own experiences or expectations of the researcher can also influence the results of an interview (Patton, 2002), especially since she had participated in the project that the interviews were about. In order to avoid influencing the answers, the questions were worded as neutrally as possible. The interviews were also recorded and transcribed.

The topic of the interviews was chosen to be specifically the previous new product development project, since that was the first new product development project that the case company had completed. Since no explicitly defined process for new product development exists in the case company, it makes sense to utilize the experiences from the previous project to determine what the process was like and what challenges it faced. A total of 8 people were interviewed. Their roles in the company included CTO, a developer, a product manager/UX designer, and the CEO. In addition, the customer's product owner was interviewed, as well as two employees of the customer, who worked both as end users and as customer service for consumers who used the service.

The interviews included questions about the new product development process from several viewpoints: the roles of the people involved, the business benefits the project aimed for, the scope of the project, prioritization of the work, communication internally,

with the customer, and with end users, and the feedback loop all the way from the end users to the case company. The interviews included direct questions about how areas of the project – such as roles, prioritization, and utilization of feedback – succeeded, as well as requests for the interviewees to describe how things happened, which were aimed at finding challenges. In addition to these questions, the interviewees were encouraged to bring up any challenges that they could think of that did not come up through the questions.

The interviews were analyzed through finding themes and categories in the transcribed interviews, as recommended by Patton (2002). First, all challenges and good practices were identified. Since there were many challenges, recurring themes were identified and grouped inside categories, each of which contains related themes. Finally, these categories were summarized into a table. Good practices that are already employed by the case company were also summarized into a table, although they were not categorized, since there was a smaller number of them. In addition, a high-level process model of the new product development process that the case company had utilized so far was developed based on the learning from the interviews and the own experiences of the author, who was part of the project team. The findings are presented in chapter 4.1.

Action intervention

In the action intervention phase, the author introduced the findings from the literature study and the interviews that were conducted in the problem diagnosis phase to the case company. In the same session, the author proposed a process for the case company to utilize when experimentation is implemented in practice at the case company.

The proposed experimentation process was formed by combining different sources from literature, while keeping in mind the particular challenges that the new product development process had previously faced at the case company, which were uncovered by the interviews. The summary of the literature review (chapter 3.7) includes the chosen sources from literature. All of the challenges that were found in the new product development process of the case company are not affected by the experimentation process. The challenges that the experimentation process is based on include challenges that the process could mitigate, and challenges that the process aims to take into account because they may be emphasized when experimentation is implemented. The challenges that the process could help mitigate were chosen based on findings from literature, but the benefit of introducing experimentation remains hypothetical for each challenge, since these were not verified in practice. The same holds true for the challenges that could be emphasized, and the whether the process succeeds in avoiding emphasizing them.

The process also took into account the current practices of the case company, by utilizing tools that the company already has in place, such as Confluence. The process

recommends using Confluence for documentation purposes, since it is a familiar tool that the company already had in use.

Reflective learning

The reflective learning part consisted of evaluating the outcomes of the action intervention phase and formulating next steps. The process that was created in the action intervention phase was adjusted to better answer the particular needs of the case company. This was done through analysis of informal conversations with the case company and their feedback of the process.

The informal discussions between the case company and the researcher included considering which projects the formulated experimentation process would be suitable for. These discussions led to the realization that none of the potential projects could be started right away with the proposed project, but instead some additional steps, including user research, were needed in order to be able to start the process. The case company held a workshop, where the project was planned that would be most suitable for experimentation, and the steps that needed to be taken before the product could be developed with experimentation were outlined together. These included finding a pilot customer with whom user research could be conducted in order to understand their processes and problems better before developing a solution. Subsequently, the author extended the proposed process to include the outlined steps that were needed before an actual experiment could be done. The projects that the case company undertakes all have similar aspects, e.g. that the customers are from the public sector, and developing solutions for them typically will require the same kind of background work to understand their processes, and hence these steps should be included in the experimentation process itself. Including the additional steps makes the process more suitable for the projects that the case company works on and makes it clearer to the case company when and how they can start experimenting.

The experimentation process does not aim to solve all of the challenges uncovered by the interviews in the problem diagnosis phase. Therefore, solutions were also formulated for the challenges that are not taken into account in the experimentation process. The author held a retrospective for the case company, which most of the employees who had had a large role in the project attended. The author introduced the challenges by category, and action points were formulated to answer the challenges which the case company perceived to need concrete actions to improve on.

3 Literature review

This section is structured as follows. First, we introduce the different factors that have enabled experimentation to emerge and affected its popularity in the software development industry. Next, we briefly explain the concept of experimentation and motivate its usage in uncertain market conditions. Then, we introduce The Lean Startup methodology in more detail. In the following chapter, we move on to implementation details and describe how a company can transition towards utilizing an experimentation system. After that we describe different experimentation processes that can be the end goal of the transformation. Once we have established what the process is like, we assess the benefits of the experimentation system. Finally, we determine some challenges and success factors that companies face with regard to experimentation.

3.1 The emergence of experimentation

The Lean Startup, written by Eric Ries in 2011, built on existing ideas in scientific literature to popularize experimentation in startups. He encouraged companies of all sizes and ages that operate in uncertain markets, to conduct experiments in order to find product-market fit and the most successful business models. However, the trend of moving toward experimentation as a product development approach has been present in the industry for far longer, especially in the context of software development organizations.

Several recent technological trends have supported the emergence of the experimentation trend (Lindgren & Münch, 2016). The ability to deploy rapidly enables companies to put their products in front of customers and users quickly and gather feedback fast. Technologies that support the instrumentation of products in order to collect passive and active customer feedback have redefined what data can be collected and utilized in development. Observations from operations can be directly fed back to developers as a result of the technological advancements in DevOps (Lindgren & Münch, 2016). Another supporting trend has been the closer integration of product discovery, validation, and delivery activities, in e.g. the form of design sprints (Lindgren & Münch, 2016).

Each technological trend that supports experimentation builds on the foundation of previous trends and achievements. For example, the ability to deploy rapidly is made possible through continuous integration and test automation (Lindgren & Münch, 2016). This trend in the industry is mirrored in the model of the transformation process of a single company becoming experiment-driven. In this model, a company climbs up metaphorical stairs to become first agile, then achieve continuous integration, then continuous development, and finally be able to implement an Innovation Experiment

System to test ideas systematically and make strategic decisions based on data (Olsson et al, 2013).

Innovation Experimentation Systems have been pioneered in cloud computing and SaaS technologies, due to them enabling continuous delivery and the possibility to observe customer behavior directly and in real time (Bosch, 2012; Lindgren & Münch, 2016). However, they are applicable anywhere where it is possible to collect usage data from the product to steer the decision-making of the organization (Bosch, 2012). Despite the popularity of experimentation in recent literature, Lindgren and Münch (2016) find that the current state of experimentation practiced in the industry is often not systematic and continuous. Companies understand the importance of validating business ideas through experimentation, and yet the practice is not widely adopted in the industry.

3.2 Experimentation as a development approach

Experimentation is an activity fundamentally linked to innovation (Thomke, 1998). Strategic innovations are innovations that are linked to all the different aspects of an offering to a customer: both physical goods and intangible aspects, such as services and brand, as well as how the production and realization of the offering is conducted (Hassi & Tuulenmäki, 2012). Strategic innovations result in significant improvements to the product in terms of customer satisfaction and market growth, but they "cannot be studied or analysed into existence" (Hassi & Tuulenmäki, 2012, p. 2). Therefore, the data needed to realize strategic innovations cannot be collected, but it has to be created (Hassi & Tuulenmäki, 2012). Experimentation is a method with which to create the data that shows where to steer the business. Experimentation with business is a well-established concept, although it is not practiced in the industry broadly and consistently (Davenport, 2009, cited in Bosch, 2012).

The goal of experimentation is to make sure that effort is spent on activities that create customer value. Trying to identify the right things to build through guessing is exceedingly difficult, if not impossible, in situations where one cannot know for sure. In these uncertain environments, it is risky for companies to rely only on the experience and intuition of its employees (Lindgren & Münch, 2016). Experimentation helps identify what is actually valuable to the customers and to prioritize development activities on these things. It gives companies a way to avoid taking large risks and instead base decisions on data, as in the experiment-driven approach, assumptions about the product are systematically identified, prioritized and validated (Lindgren & Münch, 2016). While experimentation can save companies from making expensive mistakes, it can also make it more likely that great innovations are realized (Davenport, 2009; cited in Hassi and Tuulenmäki, 2012). This is enabled by the better understanding of the needs, priorities, and behaviors of customers that experimentation makes possible (Gutbrod, Münch, & Tichy, 2017). In addition, getting customer feedback earlier and quicker is one of the

biggest benefits perceived by practitioners (Gutbrod et al., 2017; Karvonen et al., 2015). Experimentation enables companies to minimize R&D investment between customer proof points (Bosch, 2012). The process is built around ensuring that there is customer demand for the product early on (Gutbrod et al., 2017). Ries (2011) also emphasizes the need to fail early and cheaply in The Lean Startup.

In its simplest form, experimentation consists of trial and error, guided by insight for the direction in which solutions could be found (Baron 1988; Marples 1961; Allen 1966; cited in Thomke, 1998). Ries (2011) presented in The Lean Startup one fairly detailed process for conducting experimentation, but several other processes and guidelines can be found in scientific literature, such as those presented by Thomke (1998) and Fagerholm et al. (2014).

Experimentation as a development approach (also known as an Innovation Experiment System) has several characteristics. It emphasizes continuously evolving the software through frequent deployments, gives customers and users a central role in the process, and at the core of the process, focuses on innovating and testing as many ideas as possible to ensure customer satisfaction and revenue growth (Bosch, 2012). In order for a company to practice development with an IES, some prerequisites, such as continuous integration and continuous deployment, need to be in place to ensure that the company has all the necessary capabilities and can avoid common pitfalls (Olsson et al, 2013).

3.3 Overview of The Lean Startup

The Lean Startup is not a scientific publication and contains largely anecdotal evidence. However, many of its themes are also recognizable in scientific literature. The Lean Startup has taken inspiration from several sources, e.g. Steve Blank's (2013) Customer Development Process and the Business Model Canvas by Osterwalder and Pigneur (Grossman, 2016). In addition, experimentation in new product development was not invented by Ries and has been written about for more than two decades (e.g. in Thomke, 1998). The name Lean Startup pays homage to the well-established practice of Lean Manufacturing, which has been adapted from the automobile industry to numerous other fields, including software development. At the core of Lean is the idea of eliminating everything that does not create value for the end user, as those activities are therefore seen as wasteful (Poppendieck & Poppendieck, 2007). Lean includes seven principles that summarize its most important points. These principles are as follows: optimize the whole, eliminate waste, build quality in, learn constantly, deliver fast, engage everyone, and keep getting better (Fagerholm et al., 2014).

It is evident that the ideas The Lean Startup presents are not unique to it and can be found in other literature as well. Perhaps what has made The Lean Startup (and its successor, The Startup Way) remarkably successful is that it presents practical guidelines

for entrepreneurs on what to do and how to do it, not to forget the tantalizing success stories of companies that have implemented its practices.

In his book, Ries (2011) introduces the five principles that are the basis of The Lean Startup:

1. Entrepreneurs are everywhere
2. Entrepreneurship is management
3. Validated learning
4. Build-measure-learn
5. Innovation accounting

The first principle is related to who can apply the Lean Startup methodology. Ries (2011) stretches the definition of a startup to include any institution that creates new products or services, and operates under conditions of extreme uncertainty. Anyone that works in such a startup, should according to Ries (2011) be considered an entrepreneur. Consequently, he recommends established companies to make "entrepreneur" a job title. He argues that if innovation is seen as everyone's job, then no one will do it properly. Ries (2011) recommends that innovative rule-breakers should be given freedom to do what they do best – try out something radically different. However, they should also be held accountable for that their experiments lead to actual, valuable learning about the customers and the market. Respecting people is also one of the basic principles of Lean software development (Poppendieck & Poppendieck, 2007). This means giving them the tools and the freedom to use their specialized knowledge in the way that they perceive to be best for the advancement of their company and their product or service. The knowledge and creativity of each employee is seen as valuable and worth harnessing (Ries, 2011).

The second principle, "entrepreneurship is management", is related to why Ries is proposing such a rigorous process for innovation, which is often considered a creative task. The process aims to ensure that the market demand for every idea is validated in order to not waste time on developing products or features that no customer wants. The management aspect makes sure that people are held accountable for being able to demonstrate progress. According to Ries (2011), the measure for progress for a startup is learning. This brings us to the third principle of The Lean Startup: validated learning. In the uncertain environments that startups operate in, achieving product-market fit is not possible through extensive market research, as the data needed to validate the business simply does not yet exist. Therefore, it needs to be created by continuously conducting experiments that steer the direction of the business. Ries (2011) emphasizes that learning facts about the users and target market is more valuable than planning and executing: in order to create value for the end user, the most important question in new product development is not how to build the product, but what should be built in the first place. While e.g. agile development methods focus on defining how to build, and the vision of

the startup should answer the question of why, experimentation helps with determining what to spend development time on.

Ries' (2011) proposal for how to implement experimentation is by cyclically executing loops of validated learning called Build-Measure-Learn, which is also his principle number four. The loops start by identifying hypotheses related to the product. The riskiest hypotheses that are vital to get right in order for the business to survive, are what Ries (2011) calls leap-of-faith assumptions. During each Build-Measure-Learn loop, the riskiest hypothesis is picked to be tested and a minimum viable product (MVP) is built with the purpose of validating or invalidating the hypothesis.

The Lean Startup process aims to make sure that a startup is continuously making progress in its new product development efforts. As previously established, the measure of progress is learning. While learning is the desired outcome from experiments, the learning must be validated and measurable in order to ensure that progress is made. The practice that Ries (2011) proposes for measuring learning is innovation accounting, the fifth principle of The Lean Startup. Innovation accounting uses actionable metrics, which describe clear cause and effect, to monitor how experiments are helping fulfil business goals.

In scientific literature the themes of The Lean Startup have been generalized to wider topics. Frederiksen and Brem (2017) connect Ries' (2011) ideas to existing literature. They identified the main principles of the method as follows:

1. User and customer involvement in product and business development
2. An iterative approach to new product development
3. Experimentation in new product development
4. The minimum viable product
5. Entrepreneurial thinking – planning versus doing

Ries (2011) mentions e.g. testing with real customers and shifting the mindset from planning to doing throughout the book, although he does not explicitly state them as principles. Frederiksen and Brem (2017) find similarities between Ries' thinking about the entrepreneurial mindset and effectuation logic, as compared to causation logic. Effectual thinking has been addressed in more detail in chapter X.X. Frederiksen and Brem (2017) also stress the importance of agile, incremental and iterative product development; Ries (2011) does not identify this as a principle, but he recommends using the Build-Measure-Learn loop, which is an iterative way of working. When done right, experimentation leads to validated learning, according to Ries (2011). However, validated learning cannot be achieved without user and customer involvement in the process; after all, it is their behavior, or the lack of thereof that validates or invalidates our assumptions.

The Lean Startup has taken inspiration from e.g. Blank's (2013) ideas on Customer Development. Blank (2013b) himself sees the Lean Startup themes as connected to both

his own research and other existing literature, such as the Business Model Canvas developed by Osterwalder and Pigneur. He divides the methodology into three principles:

1. Intricate business plans are based on inaccurate, untested hypotheses, and therefore should be replaced with a business model canvas which summarizes how the company will create value
2. A "get out of the building" approach called customer development is used to validate hypotheses
3. Lean startups practice agile development

Blank's (2013b) take on the Lean Startup principles is clearly influenced by his own work on the topic. He includes his own customer development methodology as one cornerstone of the approach. It puts much focus on constant contact with real customers in order to achieve validated learning. In addition, he recommends using the Business Model Canvas. While Ries (2011) also echoes the need for a change in mindset, he does not specify one method that one should use to achieve this.

3.4 Transitioning from an agile company towards continuous experimentation

There are many agile methods that focus on up-front testing, but they lack a framework that would take into account the concerns of stakeholders with the goal of creating value to the customer (Fagerholm, Guinea, Mäenpää, & Münch, 2014). Agile methods incorporate a product backlog where new features and requirements are catalogued, but they do not guide on the prioritization of the items on the backlog, as they are development-focused (Lindgren & Münch, 2016). They provide input on how the product or service should be built, but not on what is valuable to build (Lindgren & Münch, 2016).

The differences between agile and experimentation are especially pronounced for products where customers expect the product to evolve continuously. When the business model is based on customers paying a subscription fee, they can leave any time, which makes it dangerous not to update the product often enough. This is especially true in e.g. products utilizing SaaS technologies and continuous deployment. Deploying these systems is less costly than with traditional technologies, as there are no installation costs and less risks. They also enable the collection of customer feedback and usage data, ideally in real time.

Companies typically cannot evolve straight from using only agile methods to innovation experiment systems. Several capabilities need to be developed in order to introduce successful, systematic experimentation; for example, customers need to be involved in the validation of new products early on before a business model has been selected, the product needs to be instrumented to collect usage data, and the architecture should enable running automated tests (Bosch 2012).

The transformation process roughly follows a certain set of steps towards a better development approach. Olsson, Bosch and Alahyari (2013) introduce a framework called "Stairway to Heaven" to imitate this process, which is illustrated in figure 3.1.

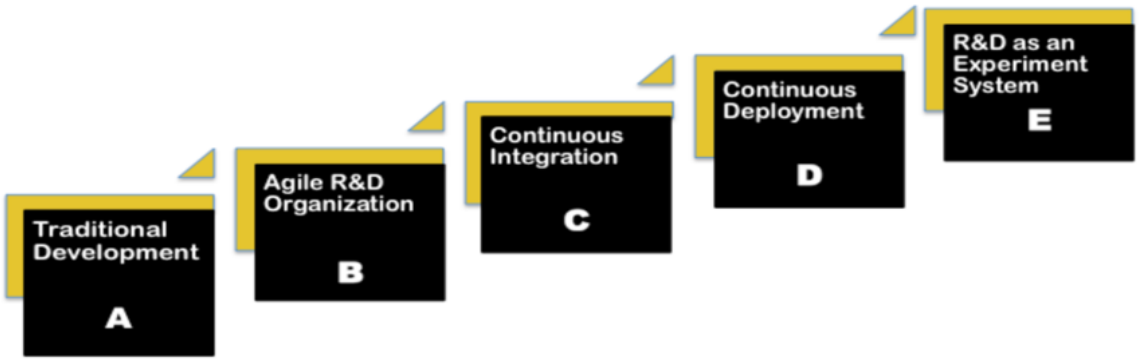


Figure 3.1. The Stairway to Heaven model, as illustrated in Olsson, Bosch and Alahyari (2013)

Companies do not necessarily have to start the process from traditional development, although that might be the starting point for older companies that still practice e.g. waterfall development. The first transition that needs to happen is the transition to an agile R&D organization. Larger companies usually begin with a few teams doing agile, before the entire organization makes the leap (Olsson et al., 2013). While having only a few agile teams can pose difficulties, the benefits of agile are greater than the challenges in implementation. Agile enables shorter cycle times, which lead to being more flexible with changing requirements. However, to adhere to short cycle times, suppliers also need to implement them. The management should be committed to the transformation and provide a clear vision. Technologically, feature teams can benefit from the support of an architect, although teams should primarily be allowed to be self-formed and self-directed. (Olsson et al., 2013)

To move from agile development to continuous integration, the product needs to be tested systematically and automatically. This introduces a variety of new tools, each of which can bring its own difficulties. However, it enables continuous quality improvement and the benefit of always having a shippable product. Transitioning to continuous integration involves external suppliers as well, as integrations to their systems need to be adjusted to the new way of working. Internally, a shift in the perspective of thinking is needed, from milestone-focused to viewing delivery and release as continuous activities. This is all enabled by the technology: increased modularization to decrease dependencies between systems, utilizing test-driven development, and automated tests. (Olsson et al., 2013)

Transitioning from continuous integration to continuous delivery means customers get incremental deliveries faster, and the development organization can learn fast from customer feedback (Karvonen et al., 2015; Olsson et al., 2013). This enables cheaper development, as time is spent on the activities that customers value most (Olsson et al., 2013), and customers are also able to perform their own testing and business activities on top of the deliveries (Karvonen et al., 2015). However, shortening cycle times can be complex and expensive (Karvonen et al., 2015). Continuous delivery of independent parts of the system requires changes to the technical architecture (Karvonen et al., 2015), as well as the capability to roll back unsuccessful deployments (Olsson et al., 2013). Developers can have a lack of trust in the quality of the software, if they are not aware of where problems are and what the status of the quality each feature is (Olsson et al., 2013). Continuous, incremental deliveries also require changes to the business model, as it should not promote fixed releases and price models that assume the requirements are frozen upfront (Olsson et al., 2013). It is not enough for only development and R&D to work in short cycles, but business development must also adopt this approach (Karvonen et al., 2015). It can be useful to identify lead customers and work closely with them (Karvonen et al., 2015; Olsson et al., 2013).

The last step on the Stairway to Heaven model is the transition from continuous delivery to R&D as an innovation experiment system. In this phase, active and passive customer feedback is continuously collected and effectively used as input for development. This requires a close relationship with the customer; for some, getting partially ready functionality that is used to experiment can be an issue (Olsson et al., 2013). It also requires some adjusting from the development organization, as they have to accept that a rigorous testing and validation phase is not possible before the functionality is released to customers (Olsson et al., 2013). The process has to be fine-tuned to establish a short customer feedback loop and decision-making based on the data (Karvonen et al., 2015). To successfully run experiments, technological architecture needs to enable run-time variation of functionality and collection of usage data (Karvonen et al., 2015; Olsson et al., 2013).

Lindgren and Münch (2016) do not identify clear steps that a transition would follow, but they also propose an evolutionary approach for moving towards experiment-driven development. They suggest that experimentation could be started from small, simple experiments, and iteratively move towards experimentation with more complex functionality, while simultaneously developing the necessary capabilities and resources, as illustrated in figure 3.2. In the first experiment, only basic skills are required, as well as appropriate product management in order to define and prioritize relevant hypotheses in a systematic way. For example, a vision board (that summarizes the motivation to create the product and some of its most important aspects, such as target group and the user need it answers to) and a validation board (that summarizes the most important hypotheses and whether they were validated or invalidated) can be utilized for this purpose. In the next experiment, technical infrastructure should be improved as it enables more complex

experiments through e.g. feedback channels, data analysis tools and data storage capabilities. In the subsequent experiments, evolving the organization's ways of working is necessary to ensure that product decisions are made based on data and the organizational structures support experimentation well. (Lindgren & Münch, 2016)

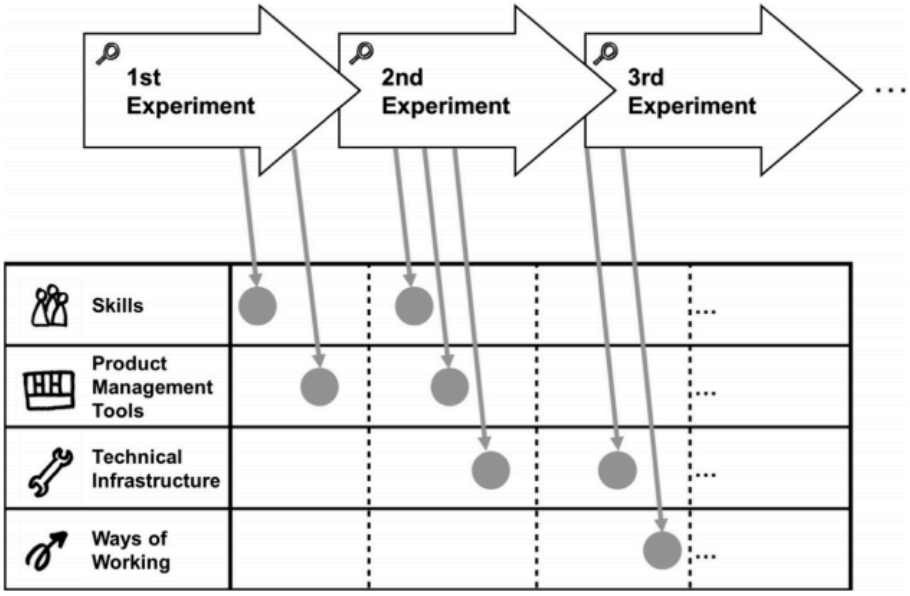


Figure 3.2. An evolutionary approach to the transition to being experiment-driven, as illustrated in Lindgren and Münch (2016).

3.5 Experimentation processes and practices

This chapter introduces first the process of Ries' (2011) The Lean Startup and associated practices. It also compares the practices to similar ones in scientific literature, in cases where such were found. Then, other experimentation processes from scientific literature are introduced.

3.5.1 The Lean Startup process and associated practices

Build-measure-learn

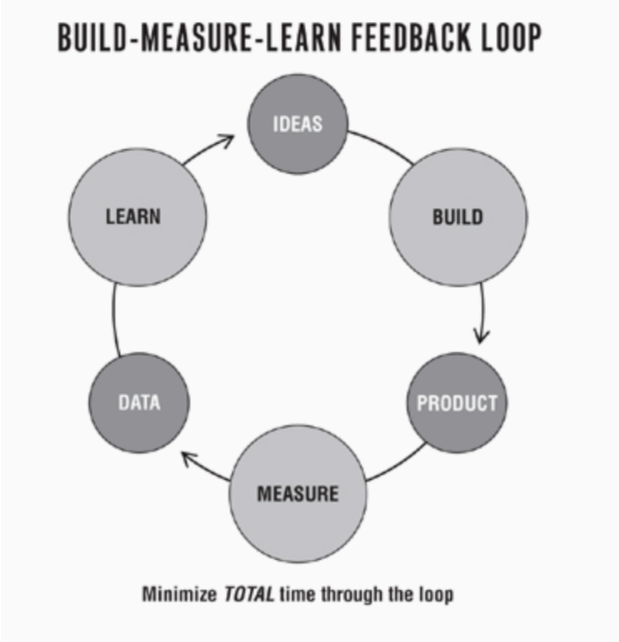


Figure 3.3. The Build-Measure-Learn loop, as presented in Ries (2011).

The Build-Measure-Learn (BML) loop, illustrated in figure 3.3, describes how the experimentation process is conducted on a high level and how the different actions in the process affect each other. In The Lean Startup, the focus is on minimizing the total time for completing the loop (Ries, 2011).

The loop begins with identifying business-critical hypotheses and deciding which ones of them to test. Generally, the riskiest assumptions should be chosen for testing. Once a hypothesis has been selected, the loop continues to the Build phase. In this phase, a minimum viable product is built. Its purpose is to validate or invalidate the chosen hypothesis with minimal effort while still producing reliable results and enabling a full turn of the Build-Measure-Learn loop. To be able to measure the impact of the MVP, it must be put in the hands of customers. In the Measure phase of the loop, the focus shifts to determining whether the product development efforts have led to real progress. The method for measuring progress that Ries (2011) recommends is innovation accounting, which utilizes actionable metrics to learn how the MVP has affected customer behavior. Finally, based on the analysis that the collected metrics have enabled, the entrepreneur must decide whether to pivot or persevere. If the learning has led to the discovery that the hypothesis is false, the startup needs to change course and compose a new strategy, with

new hypotheses to verify. Throughout the process, and even when the company decides to pivot, a startup's overarching vision should remain constant. (Ries, 2011)

Planning a Build-Measure-Learn loop should happen in reverse order. Evaluating what one needs to learn comes first, and the appropriate metrics for measuring the learning should be selected accordingly. After the metrics have been chosen, the experiment is designed, keeping in mind the desired learning outcome and what metrics will lead to achieving that learning. (Ries, 2011)

In the end notes of The Lean Startup, Ries (2011) mentions that the Build-Measure-Learn loop was heavily inspired by John Boyd's Observe-Orient-Decide-Act (OODA) loop. Originally developed for the military as a strategy for air-to-air combat, it has since been adapted for the modern business world. The strategy consists of four steps: observing the variables of the situation, orienting the aircraft into the best possible position relative to the opponent, deciding on the best course of action, and rapidly acting on that decision (Boncheck & Fussell, 2013). Boncheck and Fussell (2013) find similarities in the need for rapid decision-making in the face of uncertain situations between air combat and the business world. In both environments, there is a need to make better decisions, faster than your opponent (Boncheck & Fussell, 2013). The abstracted process of acquiring data, turning that data into insight, and then acting on that insight is similar between the environments (Boncheck & Fussell, 2013). However, one key difference is that company leaders must enable entire organizations to have this capacity (Boncheck & Fussell, 2013). This process bears distinct resemblance to the Build-Measure-Learn loop, where data is acquired through experimentation, analyzed, and used to guide business decisions, such as whether to pivot or persevere.

Scientific literature also mentions the Build-Measure-Learn loop, although it is not the first experimentation process that has been proposed. There are a few rather similar processes for experimentation presented in literature. Thomke (1998) views the execution of an experiment as a four-step iterative cycle, consisting of designing, building, running and analyzing. The main difference between Thomke's experimentation cycle and the BML loop is that Thomke sees the process as something that can be stopped once a solution is reached, whereas Ries recognizes that the process is never really finished and must be continued in order for the company to remain current (Frederiksen and Brem, 2017). Even before Thomke, Sykes and Dunham proposed in 1995 a circular process called "Critical Assumption Planning", in which assumptions are found, their criticality is assessed, tests are designed and ran, and finally the findings are evaluated (cited in Frederiksen and Brem, 2017). In contrast to the Build-Measure-Learn loop, this process also has an explicit funding request stage (Frederiksen and Brem, 2017).

Hassi and Tuulenmäki (2012) specify in their model, how the learning from experimentation leads to a strategic innovation. The strategic innovation is a collection of a number of ideas verified and adjusted through experimentation, which makes it difficult for competitors to copy (Hassi and Tuulenmäki, 2012). Hassi and Tuulenmäki (2012) call

these execution ideas, defining them as both larger ideas about e.g. what need the business will serve and smaller, more detailed ideas.

Scientific literature also reveals a larger context in which the Build-Measure-Learn loop can be used as a part of an Innovation Experiment System. In Fagerholm, Guinea and Mäenpää's (2014) model for continuous experimentation, the Build-Measure-Learn loop is repeated to test business strategy assumptions. In addition, the model specifies different roles and artifacts.

Leap-of-faith assumptions

Business plans are based on a set of assumptions and show a path to achieve the company's vision (Ries, 2011). Being aware of this is important, as knowing that the operation is based on assumptions makes entrepreneurs more likely to revise the business plan when new information becomes available (Mullins and Komisar 2009, cited in Gruber 2010). In the beginning, the assumptions have not proven to be true, which is why a startup's early efforts should focus on testing them as quickly as possible (Mullins and Komisar, 2009, cited in Gruber, 2010). According to Ries (2011), many of the assumptions are unexceptional, based on industry experience or well-established facts. The riskier assumptions, however, have a higher impact as the entire success of the startup depends on them; that is exactly why they are called leaps of faith (Ries, 2011). In order to determine which practices are riskier and which are unexceptional, entrepreneurs can identify analogs and antilogs (Mullins and Komisar, 2009, cited in Gruber, 2010). This means finding comparable experiences from the industry and establishing both best and worst practices in order to learn from them. While some lessons can be learned from predecessors, others are always specific to the startup at hand. Two of the most important and riskiest assumptions are the value hypothesis – that customers find the product valuable – and growth hypothesis – that the business will grow at the expected pace (Ries, 2011).

Identifying testable assumptions is not necessarily an easy feat. Typically, startups have a solution in mind, or an idea of what the opportunity at hand is. Hassi and Tuulenmäki (2012) find that it is challenging to break down this opportunity idea into smaller elements. Usually, teams tend to start by designing experiments for ideas that are too complex to test, and break the idea further down when they realize that it is impossible to test reliably in its current form (Hassi and Tuulenmäki, 2012). It can also be challenging to make assumptions explicit, as they are often implicit thoughts (Hassi and Tuulenmäki, 2012). To make the process easier, Hassi and Tuulenmäki (2012) recommend thinking of the hypothesis as a claim that states the action that the team assumes the user group to take, as well as the reason why they would do so.

Once the leap-of-faith assumptions have been identified, the first step is to confirm that they are realistic, and the customer really has a significant problem worth

solving. To translate intuition-based strategic plans into data, entrepreneurs must "get out of the building" to start learning, a practice Steve Blank introduced and Ries (2011) recommends in The Lean Startup. The practice is based on the understanding that the data can only be created with real customers and not thought into being inside the office. A similar practice at Toyota is called *genchi gembutsu*, "go and see for yourself", which means that strategic decisions should be based on an employee's firsthand understanding of customers, their problems and their behavior (Ries, 2011). At Toyota, this practice literally means that to understand a problem a manager can go witness it firsthand. For startups, early contact with customers only reveals which assumptions require urgent testing (Ries, 2011).

Minimum Viable Product

In The Lean Startup, Ries (2011) defines the minimum viable product (MVP) as the fastest way to get through the Build-Measure-Learn loop with the minimum amount of effort. Its purpose is to begin the process of learning by testing fundamental business hypotheses. Ries (2011) emphasizes the "minimum" of an MVP, stating that everything beyond what was required in order to start learning is waste. The first MVP of a new product is targeted towards early adopters, who do not expect or event want a finished product; they care about having a product that others have not yet discovered (Ries, 2011). Thus, the product does not need to have a full feature set or be fully functional in the first iterations. Ries (2011) claims that most people overestimate how many features an MVP needs, and recommends to always simplify the MVP when in doubt. The purpose of the MVP is to validate or invalidate hypotheses, and for that purpose it does not need to be software: there are various implementations for an MVP. Some of the most common implementations are as follows:

- **Landing page**. A landing page for a product or service that does not yet exist to measure interest in it.
- **Mockup.** The functionality is not implemented, only mocked.
- **Single feature.** Only one feature is implemented, so that it can be tested in isolation.
- **Concierge MVP.** No automation is implemented, even though the final implementation is planned to be automated. Instead, the automation is replaced with a human doing the work the service would do automatically.
- **Explainer video.** Instead of building the product, making a video that explains how it would work. For example, Dropbox used this approach to demonstrate their idea that was hard to explain in words. (Duc & Abrahamsson, 2016)

Building something fast and cheap to achieve fast learning is more valuable than having a high-quality product early on when that product might solve a problem not worth solving or solve it in a way that customers do not value.

The concept of the minimum viable product was not invented by Eric Ries. The term itself was coined by Frank Robinson in 2001, also not in scientific literature. He defined the MVP from an economic point of view, calling it a "strategy and process directed toward making and selling a product" (Robinson, 2001; cited in Lenarduzzi and Taibi, 2016). Ries popularized the term in The Lean Startup, and his definition became the most frequently reused one (Lenarduzzi & Taibi, 2016). However, others have since defined the MVP in a multitude of ways. The term "minimum" has been understood to mean different things: e.g. effort, functionalities, requirements, or implementation (Lenarduzzi & Taibi, 2016). Steve Blank's definition of "minimum" referring to a minimum feature set is, however, reused the most frequently (Lenarduzzi & Taibi, 2016). Despite the differences in definitions, there are some common key factors that characterize the MVP in its definitions. It is widely accepted to require minimal effort, result in a minimal set of features, which will be tested with early adopters, who give feedback in order for the development organization to achieve maximum validated learning (Lenarduzzi & Taibi, 2016).

There have not been many scientific studies on the MVP (Frederiksen & Brem, 2017). However, it could be a less studied topic since creating an MVP is similar to creating a scientific experiment, which is a large and complex subject. Most mentions of the MVP in scientific literature do not question the efficacy or the use of the MVP (Frederiksen & Brem, 2017). Ries has received some criticism for a lack of implementation details in The Lean Startup, focusing on the more abstract bigger picture; the MVP is one of the topics that lack a practical definition (Frederiksen and Brem, 2017). It is, however, an important consideration with regards to the scientific viability of The Lean Startup, as it is the main channel for user input (Frederiksen & Brem, 2017).

The MVP serves more roles and purposes than only supporting validated learning. It can also be used to facilitate product design, bridge communication gaps, and to facilitate cost-effective product development activities (Duc & Abrahamsson, 2016). Duc and Abrahamsson (2016) identify different roles for the MVP as a design artifact, a boundary-spanning object, and a reusable artifact. As a design artifact it can be used in the visualization of ideas, which also lets non-technical people get feedback about the technical feasibility of their ideas, as a reflection on architectural design, and it facilitates realizing mismatches between the prototype and user expectations (Duc & Abrahamsson, 2016). As a boundary-spanning object, the MVP helps in communicating between different stakeholders and mindsets; the business and the technical, the ideator and the end users, and the entrepreneur team and the investors (Duc & Abrahamsson, 2016). As a reusable artifact, the MVP serves as documentation, as a bootstrapping mechanism that is demonstrable to stakeholders and can be used as basis for the final product, and as an initial offering that enables growth hacking (Duc & Abrahamsson, 2016).

Ries (2011) notes that startups tend to delay releasing their products because they fear having their idea stolen. However, this fear is most often unnecessary, as competitors

are unlikely to notice the idea or have the time to implement a competing idea. The MVP can also be seen as a branding risk, since it's lower-quality than the final product, but this is rarely a problem as there is typically no PR hype around the first release.

One of the biggest challenges with the MVP is knowing what the appropriate fidelity for it is. A balance needs to be found between business goals and technical quality. Terho, Suonsyrjä and Systä (2016) call this the "developer's dilemma": having to choose between quality and speed, which would achieve fast product iteration. Typically, prototypes are created to either be thrown away (throwaway prototypes) or to be used as basis for the final product (evolutionary prototypes) (Nguyen-Duc, Wang, & Abrahamsson, 2017). However, in the case of building an MVP, the outcome is usually not known beforehand (Terho et al., 2016). If building the MVP results in deciding to pivot, the MVP will be thrown away. On the other hand, if the MVP is a success and validates the hypotheses that it was built to test, it can be used as an evolutionary prototype for the final product (Terho et al., 2016). It is important to understand that even if the MVP is thrown away, the work put into it is not waste if it resulted in validated learning. However, it can frustrate developers if high-quality work that they have put their talent into making is eventually not used, as the benefit gained from a throwaway MVP can in the worst case be invisible to the developers (Terho et al., 2016). For many of them, writing elegant code is a matter of professional pride and aligns with their professional values, but in aiming for minimum viability, polishing the work to a level that the developer can be proud of is often not sensible (Terho et al., 2016). This is also an issue when determining why an MVP has failed. It can be a challenging task, for which different stakeholders typically have different biases. Developers tend to think that it is the lack of refining and missing features that make an MVP fail (Terho et al., 2016). This can be dangerous if it results in the MVP being refined indefinitely, while no one understands why the customers do not want the product (Terho et al., 2016). However, the MVP should also not be thrown away before it is developed to the minimum viability. It has to be refined to a certain extent in order to produce reliable results for learning and to assess whether a business problem is worth solving. In order to mitigate the risk of over-quality or under-quality MVPs, both the business and technical side of the development organization need to have a similar understanding of the goals of the MVP and use similar measures for assessing product success (Terho et al., 2016).

Innovation accounting

The Lean Startup calls for data-driven decision-making. This data is created through experiments and measured and analyzed through innovation accounting. Measuring performance helps startups face hard truths about where they are right now and move towards their goals with the help of experiments (Ries, 2011). Performance measurement can also be useful for the investors, to whom the data can show whether the startup is working in accordance with their interests (Rompho, 2018).

There are several approaches to traditional accounting and software performance measurement, but they are not applicable in the uncertain environment of innovation measurement, and many lack a connection between the measured data and business value (e.g. Basili et al., 2007). Ries (2011) writes that startups are too unpredictable for the forecasts and milestones used in traditional accounting to be accurate. In the scientific field, research about performance measurement in small companies is very limited (Rompho, 2018). Furthermore, the performance of startups should be measured differently than that of small or medium-sized companies (Rompho, 2018). Traditional financial accounting measures, such as return on investment or earnings per share can give misleading signals for continuous improvement and innovation (Kaplan & Norton, 1992). This kind of measures are misaligned with the competencies startups aim to develop (Kaplan & Norton, 1992). However, startups should still measure their progress, because it helps them monitor whether they are reaching business goals. It gives them an accurate view of the path they are currently on and insight to whether that is a good path to keep following. This enables startups to change course in order to find better opportunities: "if a new venture does succeed, more often than not it is in a market other than the one it was originally intended to serve, with products and services not quite those with which it had set out, bought in large part by customers it did not even think of when it started, and used for a host of purposes besides the ones for which the products were first designed" (Drucker, 1985, p. 185; cited in Ripsas, Schaper and Tröger, 2015).

The method that Ries (2011) recommends for measuring progress in disruptive innovation is innovation accounting. He defines it as "a way of evaluating progress when all the metrics typically used in. an established company (revenue, customers, ROI, market share) are effectively zero" (Ries, 2017, p. 269). It gives performance assessment a better base than simply talking to a few customers and monitoring product milestones, helps startups make product prioritization decisions and decide which customers to target or listen to (Ries, 2011). Moreover, it provides chained indicators of predicted success, gives teams a focus on the most important leap-of-faith assumptions, and provides a common vocabulary to be used in negotiations about resource allocation (Ries, 2017). Innovation accounting is done by turning leap-of-faith assumptions into a quantitative financial model (Ries, 2011). In addition, it requires identifying the real drivers of growth, such as the cost of acquiring customers, network effects, or repeat purchase rate (Ries, 2011). In The Lean Startup, Ries (2011) describes innovation accounting as working in three steps. In step one, an MVP is used to establish the baseline data of the current status of the company. This data is filled into the growth model of the company. In step two, the company attempts to find the best way to achieve sustainable growth; Ries (2011) calls this "tuning the engine of growth". At this point, every initiative by the startup should aim to improve the drivers of its growth. Finally, the startup faces the question of whether to change course or stay on the current path. Now, it can see from the collected data, whether the drivers of the business model have made real progress. (Ries, 2011)

The “engine of growth” is designed to give startups a small set of metrics that they especially need to focus on, depending on what their growth hypothesis is. Ries (2011) identifies three different types of engines of growth: the sticky, the viral, and the paid engine of growth. Companies that employ the sticky engine of growth, aim to attract and retain customers in the long term. They focus all their energy on keeping their existing customers and grow when the rate of new customer acquisition exceeds the rate at which existing customers stop engaging with the product. Therefore, they can track the growth by subtracting the churn rate from the customer acquisition rate. On the other hand, in companies that use the viral engine of growth, growth happens as a natural side effect of customers using the product, as new customers will be brought by existing customers. These companies can measure how many customers bring another new customer to the product to determine whether they are on a path of sustainable growth. Finally, the paid engine of growth refers to companies that make their revenue from the profit from each customer. That is determined by the cost of acquiring each customer relative to the revenue gained from them. Consequently, growth is created by either increasing the revenue from each customer or lowering the cost of acquiring new customers. (Ries, 2011)

LEVEL 1 DASHBOARD

HLS MVP 1: Street corner lemonade stand / tables and chairs

Milestones:			Launch of Instagram Campaign	Drop in Prices	Introduction of Super-food Line	Hiring of Marketing Intern	New Location	
	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
# Passersby	100	100	125	150	175	200	400	450
# Customers	0	0	5	20	35	45	60	75
Conversion rate	0%	0%	4%	13%	20%	23%	15%	17%
Price per lemonade	9	9	9	5	6.5	6.5	7	7
# Orders per customer	0	0	1	1	1	2	2	2

Figure 3.4. Example of a level one innovation accounting dashboard (Ries, 2017)

Ries describes innovation accounting on a fairly general level in The Lean Startup and does not give many implementation details. However, the lack of these is made up for in his subsequent book, The Startup Way, where Ries gives detailed instructions for how to incrementally take innovation accounting into use and what the growth model can look like. He writes that there is too much complexity in the innovation accounting framework for it to be taken into use all at once. Instead, it should be taken into use in three different levels, each suitable for different situations and different teams. Small teams and

initiatives in their early stages can use level one, which Ries (2017) calls “dashboard”. It involves creating a basic dashboard of metrics that have been monitored. This dashboard gives the team a sense of which actions are working well, and which ones are not. An example is presented in figure 3.4. At this stage, the startup has to start demonstrating validated learning through showing how customer behavior has changed between experiments. According to Ries this is best measured with per-customer input, as that can be measured in any sample size, however small. Some per-customer metrics are conversion rates, revenue per customer, and referral rate. Good metrics should, as a whole, give answers to four perspectives: the execution perspective (did the company do what it said it would), behavior change perspective (are the employees working differently), customer impact perspective (do customers perceive an improvement), and financial impact perspective (have new sources of growth been unlocked). Ries recommends starting with manageable metrics, the collection of which is not too cumbersome. At this stage, the main aim should be to set up a regular cadence of releases and customer contact. The amount of customers reached does not need to be large, but it should never be zero. The dashboard helps get the experiments flowing with regular customer feedback, instead of always completing a product and then showing it to a number of customers. (Ries, 2017)

After the development organization has gotten familiar with innovation accounting on level one, it can move on to level two, which Ries (2017) calls “business case”. On this level, a business plan should exist, and the company should have identified and prioritized leap-of-faith assumptions. While the level one dashboard only contained metrics related to incoming revenue, it excluded costs and long-term retention. These should be displayed by the level two innovation accounting dashboard, as it aims to give a comprehensive understanding of what is happening in the business. The metrics should be picked carefully, avoiding selecting only metrics that make the team look successful. In addition, each metric should correspond to a specific leap-of-faith assumption, and no metric that does not should be included. The dashboard should enable validating or invalidating the value hypothesis and the growth hypothesis; they need to be translated into quantitative metrics. For the value hypothesis, this could mean e.g. determining what behavior indicates customer delight. In the case of the growth hypothesis, the metrics should focus on the engine of growth, e.g. for the sticky engine of growth, the referral rate; for the paid engine of growth, the revenue per customer; and for the viral engine of growth, whether new customers come to the product as a side effect of normal usage. (Ries, 2017)

Ries (2017) calls the final level of innovation accounting “Net Present Value”. At this level, innovation accounting is a rigorous process of translating learning into dollars after each new data point. This involves rerunning the full business case each time, and assumes that the company has an initial business model spreadsheet filled with predictions (or goals). As this spreadsheet is updated with the new numbers generated in the experiments, it gradually gets more accurate at predicting future numbers. When these

are translated to their net present value, the value and impact of everything that has been learned becomes visible instantly. It should be noted that this level of innovation accounting may be too much overhead for a small startup, which is why it is recommended for a startup that has already found moderate growth. (Ries, 2017)

Other implementation details that Ries (2011) recommends are cohort analysis and split testing. In cohort analysis, the focus shifts from looking at gross numbers to splitting up the customers into groups based on e.g. the week that they have come into contact with the product and observing how the behavior of each group is different (Ries, 2011). This ensures the metrics are actionable, as they focus on the behavior change of users, and helps analyze whether initiatives have led to success or not (Ripsas et al., 2015). Figure 3.5 illustrates cohort analysis through an example, in which the retention rate of customers is tracked. The customers are divided into cohorts based on the week that they have started using the service.

Cohorts Analysis

(Joined in)	Month 1	Month 2	Month 3	Month 4	Month 5	Month 6	Month 7
Week 1	100%	10%	9%	9%	7%	7%	7%
Week 2	100%	12%	10%	10%	8%	7%	?
Week 3	100%	16%	14%	13%	12%	?	
Week 4	100%	17%	15%	14%	?		
Week 5	100%	20%	19%	?			
Week 6	100%	22%	?				
...	...	...	...	...	...	...	...

Figure 3.5. Example of cohort analysis of customer retention rates (*Maurya, 2013; cited in Ripsas, Schaper and Tröger, 2015)*

Choosing the right metrics is paramount in order to produce reliable data for its intended use. Gross metrics can be misleading, as they do not emphasize current progress over past decisions (Ries, 2011). The numbers can look good even if the company has recently stood still. Ries (2011) calls this type of metrics vanity metrics and cautions against using them, since they do not accurately predict sustainable growth. Instead, he recommends using metrics that are actionable, accessible and auditable. Actionable metrics demonstrate clear cause and effect and are replicable as well as reliable. Accessibility for metrics means that they are well-defined and simple enough that everyone is able to understand the reports. Finally, auditability is important as the reliability of the data is often questioned when it demonstrates that a project is failing. Therefore, there should be

a way to confirm that the data is consistent with reality, e.g. by tracing the data back to a specific customer and confirming what the data shows with them. (Ries, 2011) However, this brings with it concerns about the privacy of users.

In the beginning phases, startups often use unrefined metrics such as revenue, since these are easy to collect (Terho et al., 2015). More detailed metrics usually require some infrastructure to collect the data from e.g. product usage (Terho et al., 2015). However, having a data collection system in place within the current product also makes it harder to pivot away from the current path and abandon the product as it is (Terho et al., 2015).

Croll and Yoskovitz (2013; cited in Rompho, 2018) have written in detail about metrics that startups can use. They have identified metrics that six types of startups (e-commerce, SaaS, mobile apps, media sites, user-generated content and two-sided marketplace) can use. However, there seems to be a lack of research regarding startup performance metrics in business-to-business, and especially business-to-government, contexts.

Ripsas, Schaper and Tröger (2015) also present guidelines for metrics with which to monitor startup performance. They call their measurement tool a "startup cockpit", which is a metaphor for how the metrics are used to steer the business. This cockpit is different for each startup, and it has to be adjusted depending on the development stage, the business model, the strategy, and the financing structure of the startup in question, as these influence what clusters of metrics are useful for the startup. It combines Ries' accounting idea with Dave McClure's "AARRR" (acquisition, activation, retention, revenue, referral) metrics, which provide indicators of revenue before revenue is realized. Just as Ries does, the authors behind the startup cockpit emphasize that not all numbers that the business generates are key performance indicators, and focus should be on actionable numbers that have a proven impact on future results. They propose that there are three relevant clusters of metrics for measuring startup performance: the customer activity, the financial perspective, and the process perspective. Customer activity metrics answer the question of how customers perceive the delivered benefit. Example metrics are e.g. the percentage of recurring customers and customer satisfaction metrics. The financial perspective focuses on the economic survival of the startup and the profit it makes. In the beginning, metrics can be e.g. the time the startup has left before it runs out of money and burn rate, and later margin analysis and return on investment can be included. The process perspective measures efficiency of the development organization. Indicators of progress include customer lifetime value and acquisition costs, as well as metrics related to quality and learning. In addition to the different perspectives, startups must consider how e.g. their growth strategy and the interests of the investors influence what are interesting metrics for them. (Ripsas, Schaper and Tröger, 2015)

The startup cockpit has clearly drawn inspiration from the Balanced Scorecard, which is a performance measurement method for startups that Kaplan and Norton

introduced much earlier in 1992. It includes roughly the same perspectives, with slightly different names, and what the startup cockpit calls the "process perspective" is split into an innovation and learning perspective and an internal business perspective in the Balanced Scorecard. Examples of measures for the perspectives that the startup cockpit and the Balanced Scorecard recommend are summarized in Table 3.1. The purpose of the Balanced Scorecard at the time it was introduced was to provide a holistic understanding for managers of how the different areas and competencies in the company are progressing. Simultaneously, it emphasizes the need to focus on a few key metrics instead of collecting too many numbers (Kaplan & Norton, 1992). Similarly to the Lean Startup method, the Balanced Scorecard also aims to help avoid optimizing parts of the company at the expense of the whole.

Perspective	Measures in the Balanced Scorecard	Measures in the startup cockpit
Customer perspective	Time it takes to meet the customer's needs, quality, service	Customer satisfaction measures, percentage of recurring customers
Internal business perspective	Cycle time, employee skills, cost, productivity	-
Process perspective	-	Customer lifetime value, customer acquisition costs, learning curve within the organization, quality
Innovation and learning perspective	Percentage of new products sales compared to total sales	-
Financial perspective	Cash flow, sales growth, market share, return on equity	Time out of cash, burn rate, margin analysis

Table 3.1. Measures for the different perspectives included in the comprehensive view of a startup's business, as stated in Kaplan and Norton (1992) and Ripsas, Schaper and Tröger (2015).

Pivot or persevere

The Lean Startup aims to determine whether a startup is making sufficient progress with its current strategy or whether a change in course is necessary (Ries, 2011). All the other parts of the methodology serve as the basis of learning that informs the decisions whether to pivot or persevere. Thus, the pivot is an outcome of validated learning (Bajwa, Wang, Duc, & Abrahamsson, 2016). The method strives to provide the necessary tools and data

to realize the company is on a failing path and make a move to correct it. Ries (2011, p. 165) defines a pivot as a "structured course correction designed to test a new fundamental hypothesis about the product, strategy, and engine of growth". Bajwa et al. (2017) link the pivot to the business aspect by defining it as a strategic change to a business concept, product or the business model. Terho et al. (2015) also state that the reason to pivot is linked to the business model. The business model consists of several hypotheses, and when one of them proves to be false, the company must consider a change of course (Terho et al., 2015).

Pivoting is considered vital for software startups in order to survive, achieve growth, and find a business model that works (Bajwa et al., 2016). When a startup pivots, it makes a new strategic hypothesis, which will have to be tested with a new MVP (Ries, 2011). Ries (2011) points out that it is not necessary to throw out all the work that has been done when pivoting; it can be repurposed into something that aligns better with the new strategic direction of the company. Ries (2011) emphasizes that there is no rigid formula to follow with pivot or persevere decisions. There is always a human element to the decisions and it may be an emotionally charged decision to abandon the previous plan. Thus, Ries (2011) recommends addressing pivoting in a structured manner. He recommends regular pivot or persevere meetings, less frequently than every few weeks and more often than every few months (Ries, 2011). In those meetings, metrics collected over time should be analyzed and compared to expectations that have been set over time (Ries, 2011).

Using the wrong kind of metrics can cause entrepreneurs to live in their own reality, not noticing the need to change. If the metrics are not accurately measuring sustainable growth, they can paint a false picture of success. Companies tend to wait too long before pivoting or avoid it altogether and neglect the validated learning process (Bajwa et al., 2016; Ries 2011). Most entrepreneurs are reluctant to abandon their vision before being sure it has gotten a real chance to prove itself, resulting in dragging on too long before pivoting (Ries, 2011). This can cause companies to get stuck in a situation where they make no real progress despite working hard. Having an unclear hypothesis also makes pivoting harder, as it makes recognizing a failure more difficult, which results in less motivation to make radical changes (Ries, 2011).

Startups pivot in order to survive. Ries (2011) writes about the "runway" with which startups measure the time they have left. Usually the runway is defined as the remaining cash in the bank divided by the monthly burn rate, but Ries (2011) states that it could also be measured as the number of pivots the startup can still make. In this metaphor the runway could be extended by finding a way to get to the next pivot faster; that is, to achieve the same amount of learning in a shorter time (Ries, 2011). If a startup fails to anticipate their failure, one failure may be enough to put them out of business, but failing early can also help avoid final, fatal failures (Bajwa et al., 2017). An inexpensive failure provides an opportunity to learn and adjust the operations accordingly.

Knowing when to pivot is also essential. Ries (2011) identifies decreasing effectiveness of experiments and a general feeling that product development should be more productive as signs that it is time to pivot. Bajwa et al. (2017) also identify factors that lead to pivots. They are mostly external factors that occur outside the control of the company. This may mean that companies usually pivot as a reaction to an outside event, instead of making a purposeful strategy re-adjustment as Ries described. The most common triggering factor is negative customer reaction. Slow user acquisition, low user retention rate, and slow growth are all signs that customers do not love the product as much as anticipated. It is extremely important to first be able to identify the right problem in order to create something valuable for customers. Unfortunately, it is very probable that the original assumptions towards what customers want will prove to be wrong, which brings the need to pivot toward a problem that is an actual pain for the customers. Competition is another common trigger for pivots. Larger competitors have more resources, and thus they can operate faster. Other triggers include – but are not limited to – technology changes, ideas brought by investors, and fixing flaws in the business model. However, multiple factors probably collectively form the true trigger for a pivot. Particular for software companies is the tendency of users to use a product beyond its intended use, so software startups should pay attention to whether this would become a cause for a pivot. (Bajwa et al., 2017)

Just as there can be many causes for a pivot, the exact form that a pivot takes also varies. Table 3.2 presents some of the pivot types that have been identified in literature.

Pivot type	Description	Source
Zoom-in	What was previously a single feature of the product becomes the whole product	Ries (2011)
Zoom-out	What was the whole product becomes one feature in a larger product	Ries (2011)
Customer segment	The product solves a real problem, but for different customers than the company originally planned to serve	Ries (2011)
Customer need	It is found that the current problem is not very important, or a more important problem is found	Ries (2011)
Platform	An application becomes a platform or vice versa	Ries (2011)
Business architecture	The company's business architecture changes from high margin, low volume to high volume, low margin or vice versa. E.g. B2C to B2B	Ries (2011)
Value capture	How a product captures value (including how it makes revenue) is changed. Can have consequences for the rest of the business, product and marketing strategies	Ries (2011)
Engine of growth	The growth strategy is changed. Ries (2011) identifies different types of engines of growth as the viral, sticky, and paid growth models	Ries (2011)
Channel	The sales or distribution channel or process is changed. The requirements of the channel may determine the price, features and competition of the product	Ries (2011)
Technology	A sustaining innovation, incremental improvement designed to appeal to an existing customer base. Can provide e.g. a better price or performance.	Ries (2011)
Market zoom-in	A better defined target group is found within the existing target group	Bajwa et al. (2017)
Side project	A side project unrelated to the main business outshines the main project	Bajwa et al. (2017)
Complete	Everything other than the team changes	Bajwa et al. (2017)

Table 3.2. Different pivot types.

Pivoting is not a widely studied subject, and there are few scientific studies available about it. Evidence for all the pivot types identified in table X has not been found in empirical studies. In particular, Bajwa et al. (2017) studied pivoting in 49 software startups, and found no evidence of pivots related to business architecture, value capture, engine of growth or social aspects. These types of pivots may be rarer, but that remains to be proven in a scientific manner in future studies. In addition, a pivot often causes another pivot (Bajwa et al., 2016). Terho et al. (2015) find that comprehensive pivots tend to happen early in a startup's life, whereas once the business model is clarified and set, the changes are smaller and focus on more detailed implementation aspects. They state that in the beginning, "when key metrics are unrefined and informal the pivots are also wide and have a tendency to contain several other pivots" (Terho et al., 2015, p. 557). Better metrics often require a system to collect data about the product (Terho et al., 2015). With the help of such data, improvement points are easier to identify, and more focused pivots can be made while measuring the effects of the pivots (Terho et al., 2015).

3.5.2 Experimentation processes in scientific literature

In the Lean Startup, Ries (2011) presents the Build-Measure-Learn loop as a way to organize the experimentation process. However, this is not an entirely new concept: for example, Thomke (1998) views the execution of an experiment in the context of new product development as a four-step iterative cycle. The steps are as follows: designing the experiment, building the necessary instrument to conduct the experiment, running the experiment, and analyzing its results (Thomke, 1998). Thomke (1998) also points out that optimal results are achieved when switching experimentation modes, i.e. how the experiment is conducted and what instrument is utilized in the experiment. Different modes can uncover different errors in the product design, and therefore it increases the efficiency of experimentation to e.g. use a low-fidelity prototype to find the largest design flaws in the beginning, but move to a more advanced prototype later that will help discover errors of a different magnitude (Thomke, 1998). This also helps reduce the investments needed before the product has been validated.

The research conducted by Thomke (1998) is more focused on how the design of a product is validated, while Ries (2011) gives great emphasis to validating novel business ideas. It is worth noting that Ries recommends experimenting with real, paying customers from the very beginning, to ensure that the business is viable and the customers are willing to pay for the product. However, there are differing approaches regarding when customers should be charged for an incomplete product. Bosch (2012) presents a model that separates the stages when the product is offered for free and when customers are charged, as well as takes into account how the validation of the business and the product differ and how the experimentation techniques should differ between them. This

model is on a higher level than e.g. the Build-Measure-Learn loop, and looks at a longer timeline, during which many iterations of BML can be executed. The model aims to keep investments in the product as low as possible before anything has been deployed to customers. The first goal is to produce an MVP as quickly as possible. Once customers can use the MVP, the initial value is often not monetizable, but the development organization can benefit from customer feedback. Once the product reaches sufficient customer value, it can be commercially deployed using the selected business model. Continuing to collect customer feedback is still valuable to evolve the product in the right direction. (Bosch 2012)

The three stages included in Bosch's (2012) innovation experiment system model are called pre-deployment, non-commercial deployment and commercial deployment. There are distinct practices to adopt in each of the stages. In the pre-deployment stage, only active customer feedback can be collected, as there is no working implementation of the product yet. Therefore, the more customers can be involved in the process, the better. Some useful techniques to utilize are BASES testing, advertising and solution jams. In BASES testing, the product concept and information such as pricing and differentiating features are presented to a panel of customers together with competing products. The customers are then asked to rate the product relative to the competitors. This approach works especially well for new products aiming for established markets and is considered a good predictor of future success. Advertising refers here to placing adverts online for products that do not yet exist and measuring interest in the product, a technique that Ries (2011) recommends as well. Solution jams are events, where employees work on turning new ideas into concepts while getting feedback from customers. At the end of the event, the most promising ideas can be selected for development. (Bosch 2012)

In the non-commercial deployment stage, development has lead to an MVP of a new product or feature, and it is ready for customer testing. The product should be instrumented in order to collect data. This can be done e.g. in the form of in-product surveys in strategic locations within the product. For drawing attention to a new product, ads for it can be placed into a product that is already in use. Many companies choose to market their new products through a "labs" website, where the new product is offered free of charge to the most interested customers for the purpose of collecting feedback before the official release. (Bosch 2012)

Once the commercial deployment stage is reached, the product has a model of monetization, although the business model is now experimented with. The focus of the experiments shifts to, in addition to validating the monetization model, optimizing the use of the system and its existing features. Real-time observation of customer behavior through passive customer feedback, and continuous alignment between business goals and metrics collected from the product is possible at this stage. A/B testing between several versions of the same functionality and ethnographic studies in a real context,

instead of a testing environment, are recommended techniques for this stage. (Bosch 2012)

Lindgren and Münch (2016) present some additional practices and guidelines for the experimentation process. As possible experiment implementations, they suggest prototypes, surveys and usability testing. However, these implementations focus on getting direct feedback from customers, although that presents the challenge of being able to see the problem that customers are presenting instead of solutions that they provide. Lindgren and Münch (2016) found that few companies have implemented sophisticated instrumentation for collecting usage data, although it is seen as valuable. However, quantitative usage data can only point to where problems are, and not why or what the actual problem is. Therefore, they recommend using both quantitative and qualitative data collection methods and data analysis. Different stakeholders should be involved in the analysis of the customer feedback: this also helps in establishing the necessary competences for everyone involved and enforcing a fast learning, experimental mindset. (Lindgren and Münch, 2016)

Hassi and Tuulenmäki (2012) break down the experimentation process into smaller pieces. They suggest that business development teams need three different idea types to make progress: an opportunity idea, an experimentation idea, and an execution idea. The process is iterative, and teams repeatedly return to the opportunity idea to start the cycle again. The opportunity idea is the first thing that is needed. A problem has been detected, and there is an initial idea of how to solve it. Next, this opportunity idea is broken down into smaller elements and the assumptions that can be tested are identified. In the study conducted by Hassi and Tuulenmäki (2012), teams would typically start designing experiments for ideas that are still too complex to test and see finding an assumption that is testable as a challenge. Teams also found it challenging to make implicit thoughts explicit for testing purposes. Once a testable assumption has been identified, it is turned into an experiment with experiment ideas. These should be small and not too ambitious or cumbersome to build. The question in creating experimentation ideas is “how to create the right setting to communicate the value you are proposing to a potential customer, in a manner that” – “allows you to learn whether your solution works or not” (Hassi and Tuulenmäki, 2012, p.10). The experiment idea should be designed keeping in mind what type of prototypes might be needed, what arrangements need to be carried out and how data is collected from the experiment. Execution ideas are concerned with how to make the product or service that was experimented with a sustainable business. These can be both larger ideas, e.g. what needs the business will serve, and smaller, more detailed ideas. A large number of execution ideas form a strategic innovation that is difficult for competitors to copy. (Hassi & Tuulenmäki, 2012)

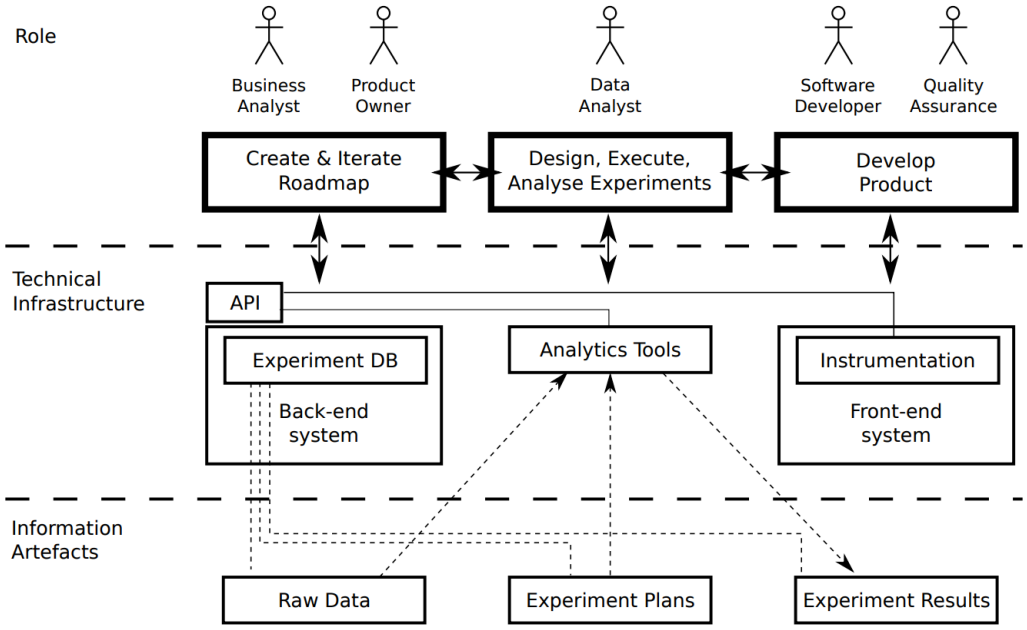


Figure 3.6. The system that Fagerholm et al. (2014) propose for continuous experimentation.

Fagerholm et al. (2014) present a framework for the organizational context in which experimentation can be conducted systematically, which is illustrated in figure 3.7. The framework involves different roles, technical infrastructure, and information artefacts. The system needs to include at least the abilities to release minimum viable products with suitable instrumentation, to design and manage experiment plans, to link experiment results with a product roadmap, and to manage a flexible business strategy. In this system, the experimentation is conducted in repeated Build-Measure-Learn blocks where assumptions identified from the business strategy are tested, much in the same way as Ries (2011) describes. However, Fagerholm et al. (2014) extend the system by defining the organizational context in which such a process can best operate. The roles in the experimentation system are business analyst, product owner, data analyst, software developer and quality assurance. In a small company, one person can handle multiple roles. The business analyst and the product owner handle creating and iteratively updating a product roadmap. In their work, they utilize a back-end system that stores existing experimental plans and results, learning from past experiences and storing new knowledge there so it can later be reused. The data analyst designs, runs and analyses experiments. They access raw data to perform analysis on it and provide developers with instructions for how the product should be instrumented. Software developers work with quality assurance to develop minimum viable products and the final product. In addition to these roles, additional roles such as a user experience designer are common. The technical infrastructure in an innovation experiment system consists of instrumentation of

the product, analytics tools for the data collected from the product, and a system that stores experiment plans and results as well as raw data collected from the product. Though detailed, this model is a generalization and many variations are possible. For example, several experiments can run in parallel, and e.g. in B2B environments, experiments may be deployed to customers in a test environment to alleviate reluctance towards changes. (Fagerholm et al., 2014)

3.6 Challenges and success factors in experimentation

3.6.1 Challenges

Lindgren and Münch (2016) found that experimentation is rarely practiced systematically and continuously in the industry. In order to be considered systematic, experimentation should fulfill the following criteria:

1. Assumptions are related to business considerations and prioritized based on them
2. The assumptions are formulated into falsifiable hypotheses
3. Experiments are designed and conducted appropriately
4. The experiments are analyzed and interpreted in the context of the business considerations
5. If the results are unexpected, an analysis of the reasons is conducted
6. The results are used in decision-making and follow-up action

(Lindgren and Münch, 2016) Each criterion has challenges related to it, but Lindgren and Münch (2016) find that the most significant challenges with respect to experimentation are on an organizational level and relate to organizational culture, product management, and resourcing. Issues such as time constraints, funding, release cycle speed, and an organizational culture that facilitates experimentation pose challenges to many companies. Release cycles can be slowed down due to bottlenecks in the development process, but that process is hard to do faster. One interviewee in Lindgren and Münch's (2016, p. 86) study says, "I don't think you can accelerate anything. What you can do is do less". Hassi and Tuulenmäki (2012) note that the team's sense of ownership and responsibility tend to increase after the first experiment iteration, which increases the speed of the process. Similarly, Yaman et al. (2016) find that while first experiments can be seen to take much effort, the subsequent experiments will be able to reuse experiment designs and guidelines.

There are several challenges related to the different aspects of systematic experimentation. The challenges are summarized in table 3.3.

Characteristic of systematic experimentation	Challenges
Assumptions are related to business considerations and prioritized based on them	Selecting which ideas to test and when (Bosch, 2012; Fagerholm et al., 2014); Avoiding risks and making prioritization decisions based on previous experience and beliefs (Bosch, 2012)
The assumptions are formulated into falsifiable hypotheses	Transforming an idea into a falsifiable hypothesis (Hassi & Tuulenmäki, 2012)
Experiments are designed and conducted appropriately	Moving quickly from conceptual thinking into action (Hassi & Tuulenmäki, 2012); Sufficient instrumentation of the product (Lindgren & Münch, 2016); Identifying a viable MVP (Lindgren and Münch, 2016); Getting permission to collect user data in B2B settings (Fagerholm et al., 2014) Lack of customer interview skills (Gutbrod et al., 2017); Lack of resources/staff for experimentation (Gutbrod et al., 2017);
The experiments are analyzed and interpreted in the context of the business considerations	Identifying metrics (Lindgren & Münch, 2016); Sharing data to everyone involved ((Lindgren & Münch, 2016); Validating the problem and the customer segments, not the particular solution (Gutbrod et al., 2017)
If the results are unexpected, an analysis of the reasons is conducted	Lack of data analysis (Lindgren & Münch, 2016)
The results are used in decision-making and follow-up action	Defining decision criteria (Fagerholm et al., 2014); Not utilizing experiment data in decision-making (Lindgren & Münch, 2016)

Table 3.3. Challenges with Innovation Experiment Systems.

There are also some additional challenges specific to the environments of businesses that are not in direct contact with their end users like business-to-consumer companies are. These challenges include e.g. customers' organizational culture, gaining access to end users, and getting the customer's consent to collect product usage data (Lindgren & Münch, 2016). Many of the challenges are especially customer-related. In the business-to-business domain, end users of the product are often not the same as the customer, but instead the customer's customer (Rissanen and Münch, 2015; cited in Yaman et al., 2016). Accessing customers can be equally challenging as accessing end users, as it may not be possible to interrupt their normal work to involve them in experimentation (Rissanen and Münch, 2015; cited in Yaman et al., 2016).

3.6.2 Success factors

Success factor	Source
Technical tools and technical competence	Lindgren and Münch, 2016
Release cycle speed	Lindgren and Münch, 2016
Good organizational culture where work is done near the customer interface	Lindgren and Münch, 2016
Finding the optimal time to switch between different experimentation methods	Thomke, 1998
Active lead users involved in experimentation	Rissanen and Münch, 2015; cited in Yaman et al., 2016

Table 3.4. Success factors in experimentation.

While challenges with experimentation are easy to identify, there are less known success factors presented in literature. The identified success factors are presented in table 3.4. Technology does not pose large difficulties for companies utilizing experimentation systems, but without it, experimentation would be much more difficult (Lindgren & Münch, 2016). Both technical tools and competence are attributing factors (Lindgren & Münch, 2016). Besides technology, many challenges are also success factors; for example, release cycle speed and organizational culture can have a positive effect (Lindgren & Münch, 2016). The organizational culture should make sure the process involves the people doing the work near the customer interface, e.g. developers (Lindgren & Münch, 2016). This also ensures that the information reaches everyone effectively. In addition, Thomke (1998) recommends switching between different types of

experimentation to achieve as efficient a process as possible. The amount of errors one can uncover with a certain type of prototype or experiment decreases the more time passes, so switching to another can aid in saving time (Thomke, 1998).

Since accessing customers can be difficult in business-to-business contexts, internal test subjects can in some cases be used instead (Yaman et al., 2016). Lead customers that are actively involved in the experimentation process can be very valuable but are also challenging to acquire (Rissanen and Münch, 2015; cited in Yaman et al., 2016).

3.7 Summary

Experimentation process and practices

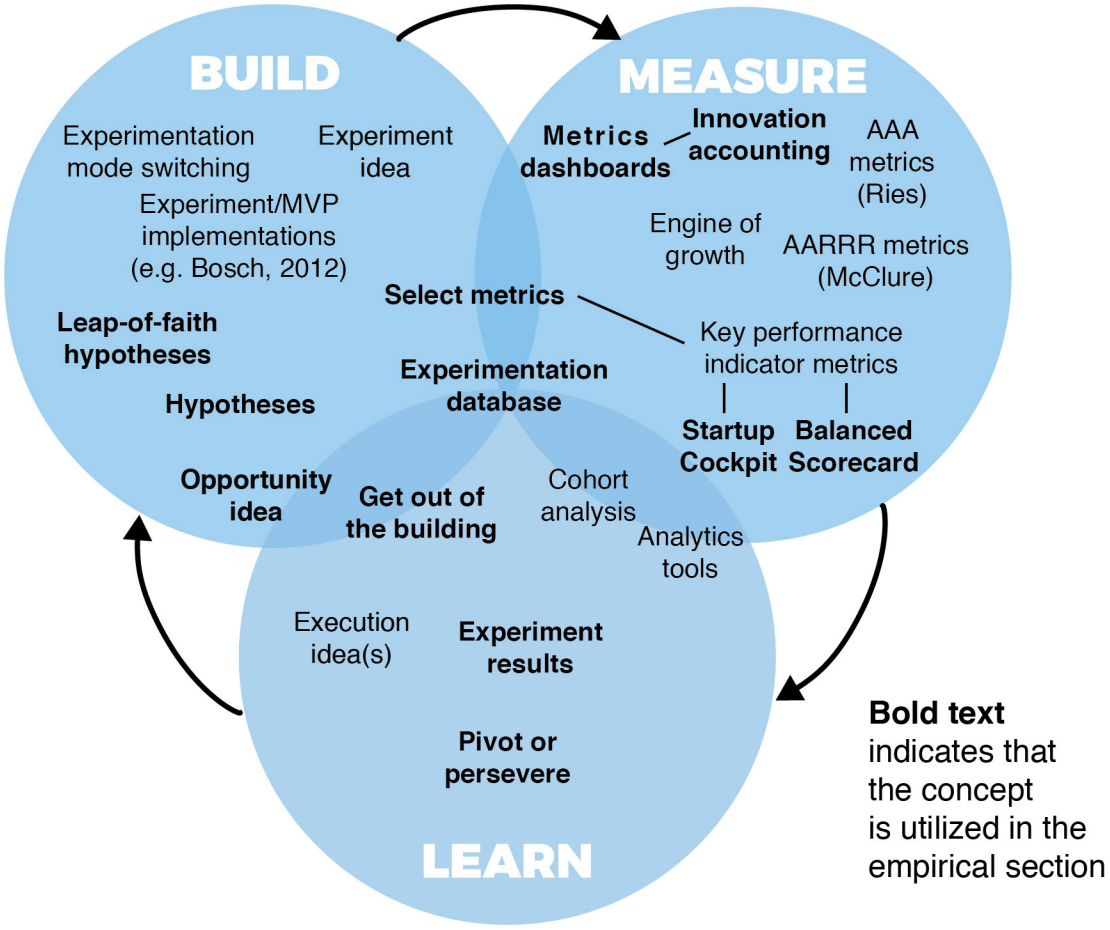


Figure 3.7. Overview of the experimentation process and related practices and artifacts found in literature.

Figure 3.7 illustrates how the different practices, methods and activities of experimentation relate to each other. The concepts indicated with a bold font in figure 3.7 are utilized in the process proposed in the empirical study.

The Build-Measure-Learn loop is central for both the Lean Startup method and other, scientific experimentation methods. It is used in figure 3.7 to encompass all the other practices, which are done in different parts of the loop.

When the process is executed for the first time, it begins somewhere before the "build" phase. The first task is to identify what is the most urgent learning that is needed in order to proceed with peace of mind, knowing that effort is spent on the most critical thing to do first. There is not yet data to learn from; but there needs to be an **opportunity idea**. The opportunity idea describes a problem and a potential solution to it (Hassi and Tuulenmäki, 2012). If no opportunity idea has been identified, one way to generate ideas is to utilize Steve Blank's (2013) recommendation to "**get out of the building**" and observe people doing their tasks in real life. The opportunity idea can be broken down into smaller components that are easier to test (Hassi and Tuulenmäki, 2012). These components that make up the opportunity idea are also often called **hypotheses**. The most important hypotheses to validate are what Ries (2011) calls **leap-of-faith hypotheses**. They are the assumptions that are most critical for the success of the project, and if they are proven invalid, the product needs a change in course. The riskiest hypothesis should be chosen to be the first that is validated with experimentation.

In the "**build**" phase, an experiment is designed and built. The purpose of the experiment is to either validate or invalidate the chosen hypothesis. To build an experiment, an **experiment idea** is needed that outlines how the experiment can be implemented (Hassi and Tuulenmäki, 2012). Bosch (2012) presents some suitable **implementations for the experiment** based on how far the product has been developed so far. The phases of product development that Bosch (2012) refers to are pre-development, non-commercial deployment and commercial deployment. Ries (2011) also lists in The Lean Startup some types of experiments. Ries (2011) recommends that the experiment is carried out with the help of some type of **Minimum Viable Product (MVP)**. The MVP can be instrumented for data collection, and instrumentation infrastructure that can be utilized in multiple experiments can also be built during this phase. Therefore, **selecting metrics** is done during the "build" phase. However, the MVP is not necessarily a product; it can be e.g. a prototype, a landing page, or an explainer video. In any case, Ries (2011) assumes that at this stage, the company has a proposal for what the product – the solution to the user's problem – would be like. However, some of the experiments Bosch (2012) considers possible in the pre-development phase do not yet include building any type of artifact to be developed in order to run the experiment. He recommends ethnographic studies and solution jams as these kinds of experiment implementations. Thomke (1998) points out that optimal results are achieved through

experimentation mode switching; i.e. changing the MVP implementations and metrics collection methods in order to gain as much new knowledge as possible.

The "**measure**" phase includes monitoring progress through selected metrics. The metrics should be chosen so that they lead to the learning that validates or invalidates the chosen hypothesis. The metrics should also be chosen before the "build" phase, in order to design the experiment for the purpose of gathering the correct data (Ries, 2011). The metrics should be tied to the company's "**engine of growth**" – its strategy for sustainable growth (Ries, 2011). Ries (2011) recommends **innovation accounting** as the practice to utilize for selecting and monitoring metrics. He extends his explanation of innovation accounting in the sequel to The Lean Startup, The Startup Way, in which he introduces three different levels of **dashboards** for keeping up with how key metrics are progressing. The key metrics should, according to Ries (2011), be **actionable, accessible and auditable**. There are additional guidelines for metrics in both business and scientific literature. In business literature, one notable example is Dave McClure's so-called pirate metrics: **AARRR** (acquisition, activation, retention, revenue, referral) **metrics**. However, these metrics are not as relevant to the business-to-government field that the case company operates on, as they are to business-to-consumer markets. In the business-to-government field, retention of customers is not as challenging, because switching costs are high, and there are very rarely cases where acquisition would have happened (in the case of the case company, this would mean that the customer has signed a contract) but activation would not have happened (this would mean that the customer has started to use the product). In addition, acquisition of customers is much more cumbersome than with other types of products. Therefore, AARRR metrics are not good indicators of progress for the case company. However, the **Balanced Scorecard** and the **Startup Cockpit** include **key performance indicator metrics** that the case company can utilize. Both of them consist of different clusters of metrics which to consider when choosing appropriate ways to measure progress and innovation.

In the "**learn**" phase, the focus is on analyzing the results of the experiment. This can involve **analysis tools** and e.g. **cohort analysis**, in which users are grouped into categories based on for example when they have signed up for a service, and the change in the behavior between different user groups is analyzed (Ries, 2011). This phase also involves making decisions about how to proceed based on the learning from the experiment. The most important decision is whether to **pivot or persevere**; whether the current path still seems to be taking the product in the right direction, or whether a change in course is necessary. Hassi and Tuulenmäki (2012) call ideas that stem from the learning from the experiment **execution ideas**; they are related to how the product should evolve in light of the new knowledge.

Figure 3.7 also includes artifacts and infrastructure created and used in the different parts of the experimentation process, as recommended by Fagerholm et al. (2014). The **experimentation database** is part of the infrastructure. Inside of it are stored

the experimentation artifacts (the experiment plan, the experiment results and the raw data). The technical infrastructure also includes analytics tools and instrumentation infrastructure used to collect data about the product usage.

Achieving systematic experimentation

Figure 3.7 does not include how a company can gradually move towards systematic experimentation. According to the Stairway to Heaven model, having an agile organization with first continuous integration and then continuous deployment practices in place are prerequisites for an innovation experiment system (Olsson, Bosch and Alahyari 2013). Lindgren and Münch (2016) recommend that the first experiments that a company makes are lightweight, and that companies move towards more complex experimentation systems gradually. In the first experiment, only skills and product management tools are needed; once the company has accumulated some experience from experimentation, they should also evolve their technical infrastructure and ways of working.

Implementing experimentation systematically in practice poses several challenges for companies. The most significant challenges are on an organizational level and relate to organizational culture, product management, and resourcing (Lindgren and Münch, 2016). Time constraints, funding, release cycle speed, and an organizational culture that facilitates experimentation are some of the most common challenges (Lindgren and Münch, 2016).

The case company operates mostly in a business-to-government environment. However, there are not many studies that explore how to implement experimentation processes for such businesses. However, the challenges that business-to-business companies face are interesting for the case company, since there are similarities to how they e.g. access end users or acquire new customers. Some typical challenges for experimentation in business-to-business environments are gaining access to end users and getting the customer's consent to collect product usage data (Lindgren and Münch, 2016). Accessing the customers can also be challenging if they have to interrupt their normal work to take part in experimentation (Rissanen and Münch, 2015; cited in Yaman et al., 2016).

Answering the typical challenges well can be enough to succeed in systematical experimentation. Technological tools and competence are also an attributing factor, even though they do usually not pose a significant challenge for companies (Lindgren and Münch, 2016). To answer the challenges specific to business-to-business contexts, internal subjects (Yaman et al., 2016) and lead customers can be utilized (Rissanen and Münch, 2015; cited in Yaman et al., 2016).

4 Empirical study

This section summarizes key findings from the empirical study. It is structured according to the different phases of the chosen research method, action research: problem diagnosis, action intervention, and reflective learning.

4.1 Problem diagnosis: the current process

This section analyses the new product development process of the case company through examining the previous – and only – new product development project of the case company through interviews of the case company and the customer. This section maps out the current new product development process utilized by the company and identifies current good practices and challenges.

4.1.1 Current process

Figure 4.1 presents the current process. It consists of building multiple minimum viable features (MVF) that together form the final product. The project idea came from the (then potential) customer. Based on their idea of what needed to be done, the first step was to plan the entire project and which features to include in the scope on a high level together with the customer. Then, each feature was built one by one. The building of each minimum viable feature consisted of a feature planning meeting, in which specifications were developed for the feature, design of the feature, in which a mockup was typically produced, and development, in which the actual MVF was developed. The specifications and mockups were documented in the documentation tool the company uses (Confluence). The customer would give feedback about the mockup and the MVF, resulting in possible changes. Once each feature were finished, they were taken to production and together, the features formed the final product.

In real life, the progression of the process was not quite as linear as figure 4.1 might make it seem. In reality, more high-level planning and prioritization had to be done during the project as well, when it was deemed necessary due to new requests from the customer. Sometimes, the process of building an MVF was not completed at once due to re-prioritizing features or a more urgent feature or different project needing attention. In addition, ad-hoc design was often necessary during development, to decide how the details of each functionality would work.

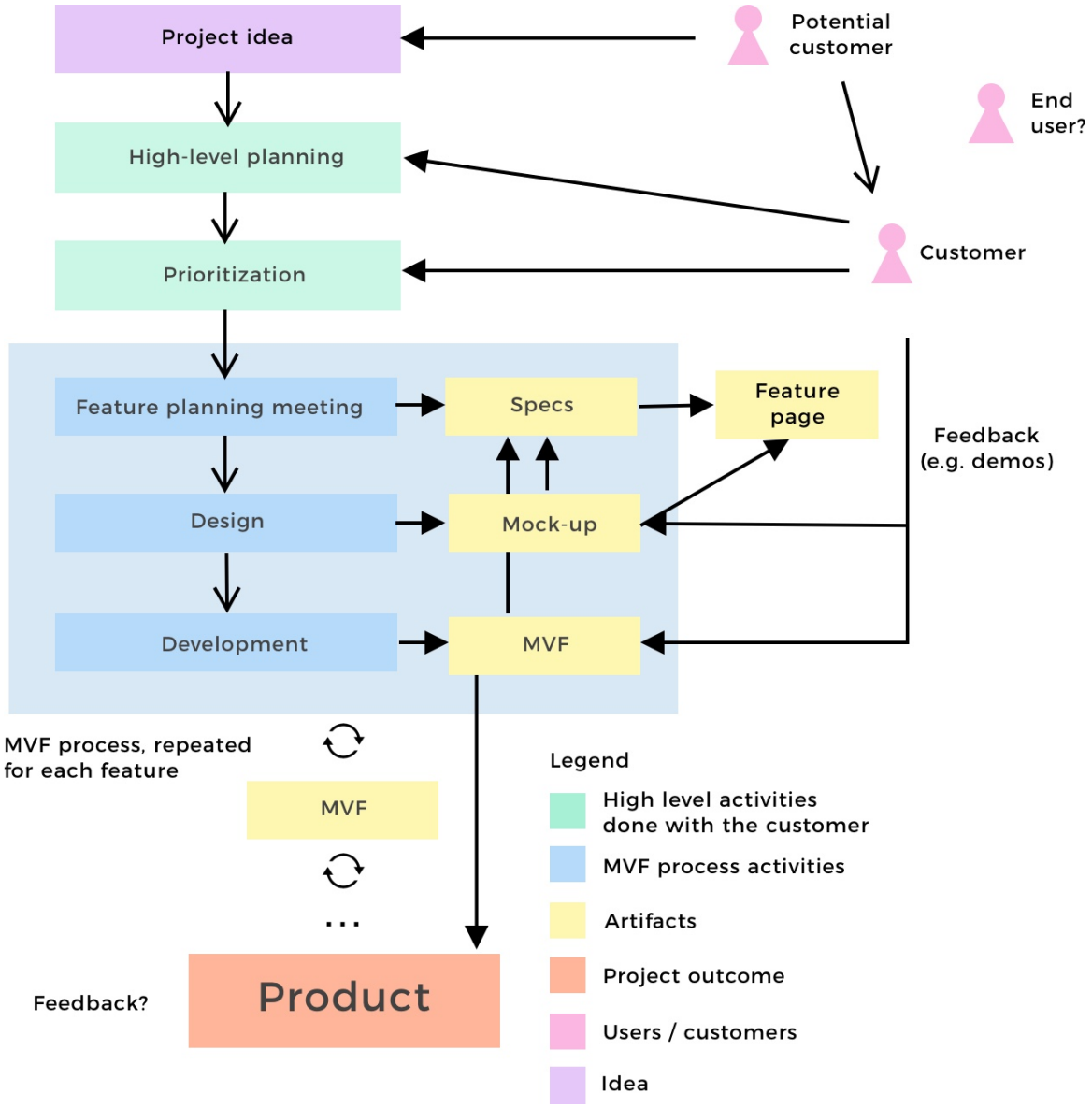


Figure 4.1. The current process employed by the case company.

In the current process, the project idea came solely from the customer, and the case company was only the implementing party. The project was planned very loosely before a contract was written between the customer and the case company. More detailed planning usually happened while planning each feature. Prioritization of features was done together with the customer before the features were developed, but also during the project as needed. The MVF building process started with planning the feature, which resulted in a specification of what the minimum viable feature feature should be like. Sometimes this specification would be either only stored in the heads of team members, or alternatively, documented in text in Confluence, which is the documentation tool used by the case company. This understanding of what the feature includes was used by the UX designer

to design a mock-up of the interface (in the cases where the feature would have an interface – if the feature did not need a UI, then this phase would instead be a technical design phase done by the developers). The developers broke down work based on the design and their understanding of what the feature requires technically and started development work. Typically, the design would be showed to the customer already before development started to get feedback. Additionally, development progress would be shown to the customer during the development in weekly demo meetings over Google Hangouts. Customer feedback would sometimes lead to repeating the design and development steps to include the requested changes. Once the customer was satisfied with the feature, it would be released to production.

It is worth noting that while customer feedback about features was regularly collected, end user feedback is missing from the process. In addition, the design of the feature was not based on user research, but instead on specifications formulated together with the customer. These specifications would be based on guesses of what users need and want, and naturally also the expertise and experience of the people involved, but the guesses would not be validated in real life.

4.1.2 Current good practices

Good practice
The role of the product owner as a filter between developers and the customer
Building MVPs with minimal functionality
Showing the customer demo versions of work in progress every week
Enabling faster communication with the customer through Slack
Building trust by visiting the client in the beginning of the project

Table 4.1. Good practices that the interviews uncovered in the previous new product development project.

Table 4.1 presents recurring good practices that the case company already has. Noticeably fewer good practices than challenges were identifiable in the interviews; however, one explaining factor could be that the interview questions were especially formulated to find challenges.

The development team at the case company especially appreciated that the product owner acted as a filter between them and the customer. This resulted in less interruptions to development work and enabled the developers to focus on value-creating work, whereas the product owner handled requests for which developer expertise was not necessary. Requests for changes and additional work were prioritized by the product

owner and added to appropriate places in the work queue, rarely resulting in immediate interruptions in the work of the developers.

Throughout the project, new functionality or features were built as MVPs. This had its own challenges: the chunks of work were sometimes perceived as too large, which was also de-motivating, they were often larger than estimated, and sometimes too much time was spent on making the MVP technologically robust. However, the team persisted with the principle of producing MVPs with minimal functionality in order to deliver results to the customer as fast as possible.

Despite challenges in the communication with the customer, good practices were also established. The customer was happy that the team had visited them often in the beginning of the project, as that gave both sides the opportunity to get to know each other personally. Many interviewees mentioned that the established trust was a great help in keeping the customer patient even when challenges were faced. One team member reflected, “I think it was good that we spent a few days there in the beginning, and that they kind of developed a feeling that we probably know what we are doing, I don’t think they would have been so calm and patiently waited during all delays if they hadn’t trusted that we know better: that if we want a high-quality end result, then that will take time”. Additionally, the customer liked having Slack as a channel that enabled informal and fairly quick communication.

The interviewees found that even more important than the economic benefit from the project was the learning it helped acquire. As it was the first project that the case company did since it was formed, it contributed very much both to the domain knowledge at the case company, and to forming own practices in e.g. project management and development.

4.1.3 Current challenges

In order to improve the new product development process at the case company, the author searched for challenges in the current process. These challenges are used later in the research process to determine which experimentation practices would be especially beneficial for the case company. Since the interviews specifically addressed the previous new product development project of the case company, some of the identified challenges are not generalizable as problems in the process but are rather problems in the specific project.

The project had many challenges that can be attributed to it being the first project that the case company worked on since being formed. In addition, the company had acquired a startup and the two were in the process of merging together, so the people within the company were not familiar with each other yet when the project was sold and

started. Some things were gradually learned through the project, while other challenges that arose will have to be addressed in subsequent projects.

The challenges in the project can be divided into six main clusters or problem areas. They are as follows:

1. Challenges related to how the project was sold;
2. Lack of a shared vision of the goals of the project and the alignment of the project relative to company-wide goals;
3. Work estimates much smaller than the realized outcome;
4. Lack of user research and feedback collection;
5. Challenges related to prioritization decisions;
6. And communication issues

The clusters are explained below in more detail, with tables that show all challenges related to each cluster. Some challenges were related to multiple clusters, in which case they are included in multiple clusters.

Challenges related to how the project was sold
The project was sold with a fixed price and ill-defined scope, instead of e.g. selling iterations
The client was originally promised a scope larger than the case company could deliver
Developers were not involved enough in the discussion during the selling of the project
The case company did not properly understand what the items the client wanted in the scope meant

Table 4.2. Challenges related to how the project was sold.

Table 4.2 lists the challenges that arose from how the project was sold. The project was sold at a fixed price, and the scope was only loosely defined. Agile software projects are typically not sold with a fixed price and scope, partly due to how difficult it is to estimate in advance the amount of time and effort that will be needed for a project. While there were no surprises regarding the client's expectations of what is to be delivered, the team did not have a good understanding of what the features that the client wanted meant on a technical and use case level. Nevertheless, the case company agreed before the start of the project to develop these features. The feature list was later, after the start of the project, reduced to the most important items that formed a more realistic and better-defined scope. One team member described the process of defining the scope as follows: "At first we said that everything was possible and promised all kinds of things. And the developers were not included in the discussion, they could have said that it makes no sense".

Lack of a shared vision of the goals of the project and the alignment of the project relative to company-wide goals
It was not clear whether the product will be a proof-of-concept MVP or later be used as a longer-term production product. Therefore, much time was possibly unnecessarily spent on making the product technically robust
The project did not align clearly with the case company's business goals; the company wants to focus more on building own products rather than custom solutions for clients
The business impact of the project may be smaller than anticipated due to being less reusable than originally intended

Table 4.3. Challenges related to the lack of a shared understanding of the business goals of the project and the alignment of the project relative to company-wide goals

Table 4.3 presents the individual challenges related to the lack of a shared vision of the goals of the project and the alignment of the project relative to company-wide goals. The team members had differing views on the goals of the project from the perspective of the case company, and it was unclear internally within the company, what the intended outcome of the project would be. Some team members viewed it as almost a proof-of-concept MVP, intended for use for a maximum of one year, while others thought of it as an exercise in writing production-grade software, while also possibly being a product that could be sold to other companies as well. Multiple team members pondered in their interviews whether the product could have been delivered faster had it been less technically robust and had existing tools been used efficiently. Lighter technological choices that enable faster building would have been preferred by some team members.

The project was also not clearly aligned with the business goals of the company. The likely cause for this was that the business goals were not clearly formulated in the first place at the time when the project was agreed upon, as the company was still very new. The team also did not share an understanding of the common business goals. Although everyone agreed that the project had achieved some business benefits, many questioned whether the effort spent on the project could have been better spent on something else. As one team member put it, "We have learned a lot about this domain and scene. How we can position ourselves now with regards to new opportunities." — "These doors would not have opened if we hadn't done this. Then of course some other doors would have opened". One of the expected business benefits was being able to sell similar projects to other clients. While the company may now be in a better position to be considered as a partner for these projects, the outcome of this project is not directly reusable as a basis for future projects. Moreover, the current strategy of the company has

evolved to moving towards developing own products instead of always making custom solutions for every client.

Work estimates much smaller than the realized outcome
The case company did not properly understand what the items the client wanted in the scope meant before making work effort estimates
It was not clear whether the product will be a proof-of-concept MVP or later be used as a longer-term production product. This led to a mismatch between the work estimates and the realized outcome
The case company did not properly understand the real-life use cases of features and the problems users will face using the features before estimating the effort required to build them
Existing services and assets were not re-used as much as they could have been
The progress felt slow due to the work being done in too large chunks, which also affected motivation in the case company and kept the customer waiting a long time before each release

Table 4.4. Challenges related to work estimates being much smaller than the realized outcome.

Table 4.4 lists the challenges related to the size of work estimates continuously being much smaller than the actual outcome. Some of the challenges regarding work estimation arose from the previous challenge cluster: the way the project was sold. The price for the project was determined based on work effort estimates that were made before the project was started. These estimates were made hastily, based on only a partial understanding of what the items the client wanted to include in the scope meant and what they meant in terms of effort for the technical implementation. In the course of the project, features such as strong authentication caused more work than was initially expected. There had also been little user research to help understand the real-life use case of each feature. The estimates were also made with the assumption that existing services and assets would be utilized in order to speed up the process, but later it was decided to start implementation without them. Perhaps due to the size of the initial estimates, the size of the chunks of work done before each release became large. This resulted both in frustration for the client, as they had to wait a long time between releases, but also frustration internally as progress felt slow. Based on the challenges faced by the case company, one way to mitigate the challenge of inaccurate work estimates would be aiming to sell and plan projects so that the chunks of work being planned or estimated are well understood and not too large.

Lack of user research and feedback collection
Little user research with only one user group; no knowledge on what other types of users need or how the service should work for them
No customer feedback was collected; no clear picture of what the customer thought of the end result and how the project went
No end user feedback was collected; no knowledge on what the end users thought of the outcome of the project (the service)
The case company did not properly understand the real-life use cases of features and the problems users will face using the features, before the features were built

Table 4.5. Challenges related to the lack of user research, user testing and feedback collection

The challenges related to the lack of user research and feedback collection are summarized in table 4.5. Minimal user research was done before the project was started in order to make a list of features included in the project. This user research was conducted only with one user group, which meant there was no knowledge on what other types of users would need from the service. The lack of user research and user testing both before the project and during development was reflected into many aspects. It meant that the team did not fully understand the use cases it was building the product for; it made prioritizing harder; there was no knowledge on what the users wanted or what they thought of the outcome; and there was no clear, coherent picture of how satisfied the customer was with the project. The team was unanimous that there should have been more contact with end users, both during and after the project. However, as the end users are not only the client but also the customers of the client, they are harder to reach.

Challenges related to prioritization decisions
There was no data to help understand which items are a priority for users
Not building some features led to excessive manual work for both the case company and the client
The development of some features was stopped in the middle of the work, then finished some time later after a pause due to more pressing matters needing acute attention

Table 4.6. Challenges related to prioritization decisions.

The team identified no problems in prioritizing the order in which to implement features. However, the priority of different items as perceived by users was never determined due to the lack of user research. As table 4.6 shows, there were also challenges regarding which features to build in the first place as well as prioritizing this project among other projects within the case company. Not building the feature of letting customers update their submitted information in the service was a decision – made primarily by the customer – that led to excessive manual work for both the case company and the customer, as the information had to be changed manually. Eventually, the case company built a script to automate the work. However, the functionality could have been built in to the user interface without making it a separate feature that needs approval from the customer, had it been clear that it was much needed. This could possibly have been achieved through user testing. The prioritization issue of this project relative to other projects stems from the small size of the development team at the case company. When something urgent requires attention, there are not enough developers to work on the other projects. That is why there were situations where the developers working on the project in question had to stop the work in the middle of developing a feature and continue once the more urgent work had been done.

Communication issues
The customer feels that responses to requests have sometimes been very slow or there has been no reaction at all from the case company
There was less communication with the client in the middle of the project, which made them frustrated
Client had trouble with providing customer service to the end users, due to the case company holding the technical knowledge needed to help some customers
Customer sometimes received angry calls from end users who were not getting responses from the case company's support email
Development and marketing have little contact, and development projects are under-utilized in marketing and sales
The customer announced the publication date of the service without the knowledge of the case company, which the case company found out about only a week before the service was supposed to be published

Table 4.7. Challenges related to communication.

Table 4.7 summarizes challenges related to communication. Most of the communication issues were between the case company and its customer. The customer felt that sometimes it took longer than it should for the case company to answer questions and

requests. One of the people involved in the project on the customer's side explained, "we have mainly been in contact with [the case company] through Slack, and I have to say that sometimes answering has been slow and taken an unnecessary amount of time considering that there's things that we need to find out". In the beginning of the project, this could have been caused by that the role of the customer responsible fluctuated between different people. There was no one person who would take care that every request or question from the client is answered. However, it does not explain slow communication in the middle of the project. Many team members noticed that communication decreased compared to the beginning, and that this caused some frustration for the client. In addition, the client felt that providing customer service to end users was difficult, as they themselves could not answer technical questions or see the same views as the end users in the service. They often had to forward the end users to the team at the case company, so that they could provide better customer service. There were cases where the case company took too long to answer end user messages, and they called the client to complain. Internal communication within the case company was not without issues either: the development efforts were not utilized in marketing and sales as much as they could have been, due to that the development and marketing roles had little contact.

It should be noted that it was evident from the interviews that some personnel at the customer did not quite understand the agile and iterative nature of software development projects. This manifested as e.g. a belief that the only way to solve the problems would be more careful planning beforehand. However, the case company also agreed to sell the project at a fixed price and scope, although it is rare to be able to set the schedule, scope and budget in advance for any agile development project that is not very small. As the client expected to be delivered a finished product that adheres to their specification on a predefined schedule, they announced the publication date without the knowledge of the case company, who only found out a week before the supposed publication. This shows that the case company should have taught and educated the client better on what to expect. This is especially necessary in the case of experimentation projects, since they require even more understanding and co-operation from the client in business-to-business (and in the case of the case company, business-to-government) projects. In addition, clients in the domain of the case company typically have no previous experience from software projects.

4.2 Action intervention: introducing experimentation

This section introduces the experimentation process suggested for the case company, illustrates it with an example, and reviews which challenges it aims to address in the current process.

4.2.1 Process proposal

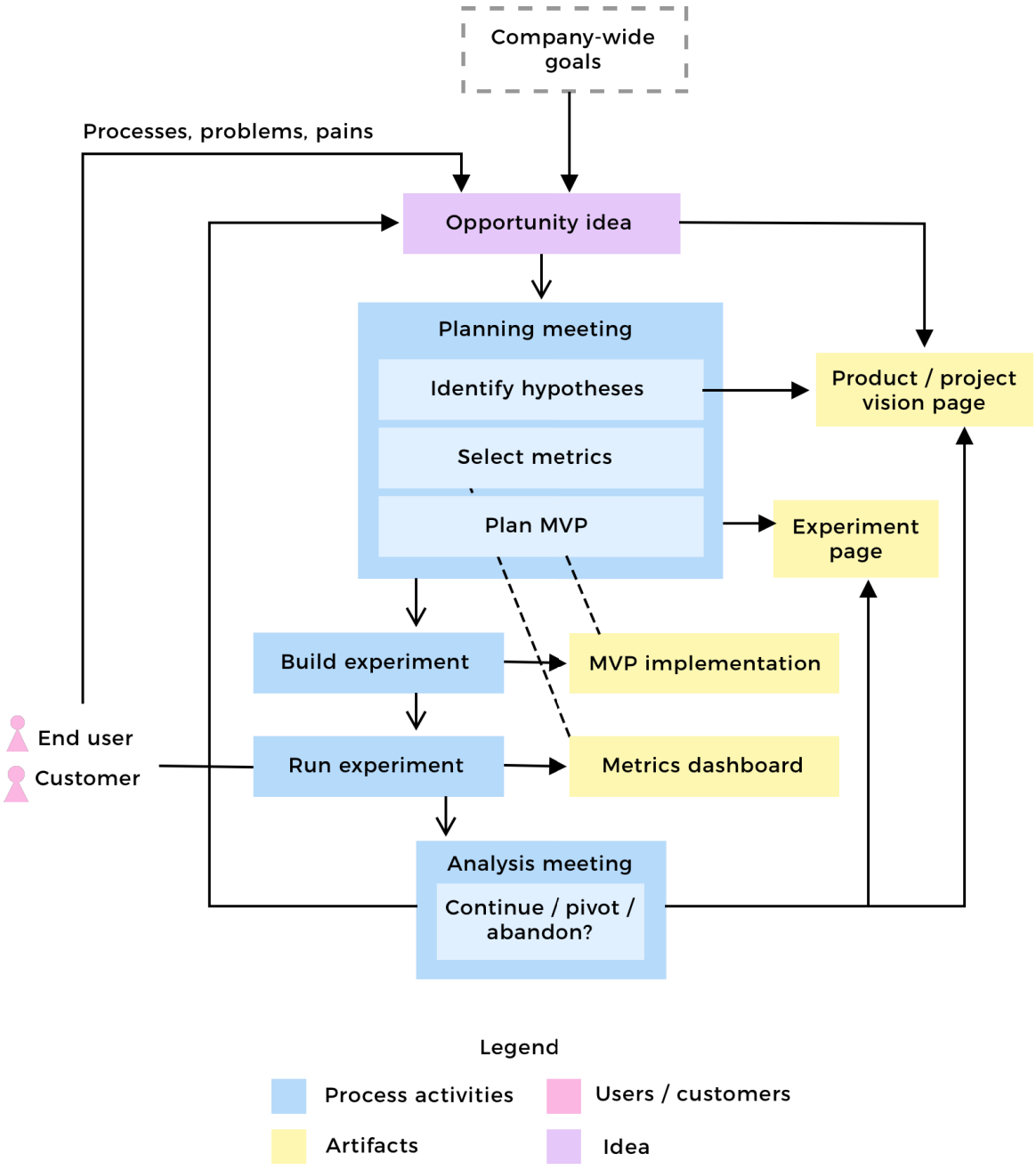


Figure 4.2. The proposed experimentation process for the case company.

Figure 4.2 illustrates the experimentation process suggested for the case company. The experimentation process starts with an opportunity idea that can come from e.g. the company-wide goals, the processes, problems and pains that the customers or end users

have, or through analysis of previous experiment results. The opportunity idea is an idea of a problem and a potential solution to it. It is turned into an experiment in the planning meeting, where it is broken down into smaller assumptions that can be tested. Both the opportunity idea and the assumptions it contains are documented on a product/project vision page in the case company's documentation tool of choice, Confluence. The metrics that test the chosen assumptions are chosen, as well as the suitable implementation for the MVP that enables the collection of the necessary metrics. These are documented on an experiment page, which is also in Confluence as a sub-page for the product/project vision page. After the planning meeting, the experiment is built and ran with real customers or users, while the metrics are monitored on a metrics dashboard. Finally, the results of the experiment are analyzed in an analysis meeting, where the results are again tied to the product/project vision and it is decided how to proceed: whether to continue in the current direction, pivot, or abandon the product or project.

One of the key differences to figure 4.1, which illustrated the previous product development process of the case company, is that the experimentation process is not the entire product development process. Instead, it can be repeated at any stage in the product development process where it is seen useful. Consequently, the outcome of the experimentation process is also not a product. The outcome is the learning that was gained, and its effect on the product vision. The process may still result in a minimum viable product (MVP) that is used as a basis for the full product, but that depends on the chosen implementation for the experiment. The MVP could also take a different form than a product similar to the intended final product, in which case it might not be directly usable for the purpose of building the final product on top of it.

Another key difference: the exp process is more driven by the case company; the opportunity idea is theirs and not the customers, and therefore they are working more towards their own goals and not just implementing the wishes of a customer.

Opportunity idea

The opportunity idea is an idea of a problem that needs to be solved, and a preliminary idea of a potential solution to it. It can be hard to come up with opportunity ideas, which is why the process includes some sources for opportunity ideas. They can come from understanding the processes, problems or pains of the customer or the end user, in which case they are ideas of how to improve the current solutions or which new solutions to develop. They can also be based on company-wide goals or the learning acquired through previous experiments.

Planning meeting

In the planning meeting, an experiment that tests part of the opportunity idea is designed. People with relevant skills should be involved in the planning meeting; e.g. if the experiment is likely to require technical implementation, developers should be involved. The first task in the planning meeting is to identify what needs to be learned in order to

know whether the solution of the opportunity idea should be implemented or not. The opportunity ideas are typically too large to be tested in their entirety in one experiment, and therefore they are broken down into multiple **hypotheses** that they consist of. The hypotheses are assumptions of what will happen and why: for example, what actions users are assumed to take and why. Among the most important hypotheses include the value hypothesis – how the product will create value for its users – and the growth hypothesis – how the company expects the product to acquire new customers. The most critical hypotheses should be chosen as the first ones to validate with experiments. One or more hypotheses can be chosen, as long as they can all be validated with the same minimum viable product.

After the hypothesis which to test has been selected, the focus shifts to deciding what data would help validate or invalidate the hypothesis. This affects the choice of appropriate **metrics**. In addition to selecting the metrics, it should be decided what the success criteria are for the metrics: what results in the metrics indicate that the hypothesis was valid, and what results indicate that the hypothesis was invalid. In addition, a person should be chosen to be responsible for making a metrics dashboard to keep everyone up to date on how the metrics are progressing while the experiment is running. Once the metrics have been planned, the appropriate data collection method and implementation for the **minimum viable product (MVP)** can be chosen. The timetable for the experiment should also be set in the planning meeting, including when to conclude the experiment in an analysis meeting and make decisions about how to proceed.

Building the experiment

After the planning meeting, the experiment is built. This phase can take many forms, but its output should be some **implementation for an MVP**. It is not necessarily a product or even a prototype that is built – it can also take the form of e.g. a solution jam or an advertisement for a product that does not exist yet. However, this phase includes preparing all necessary settings for running the experiment. If the MVP needs to be instrumented for passive data collection from users, it should be done in the building phase.

Running the experiment

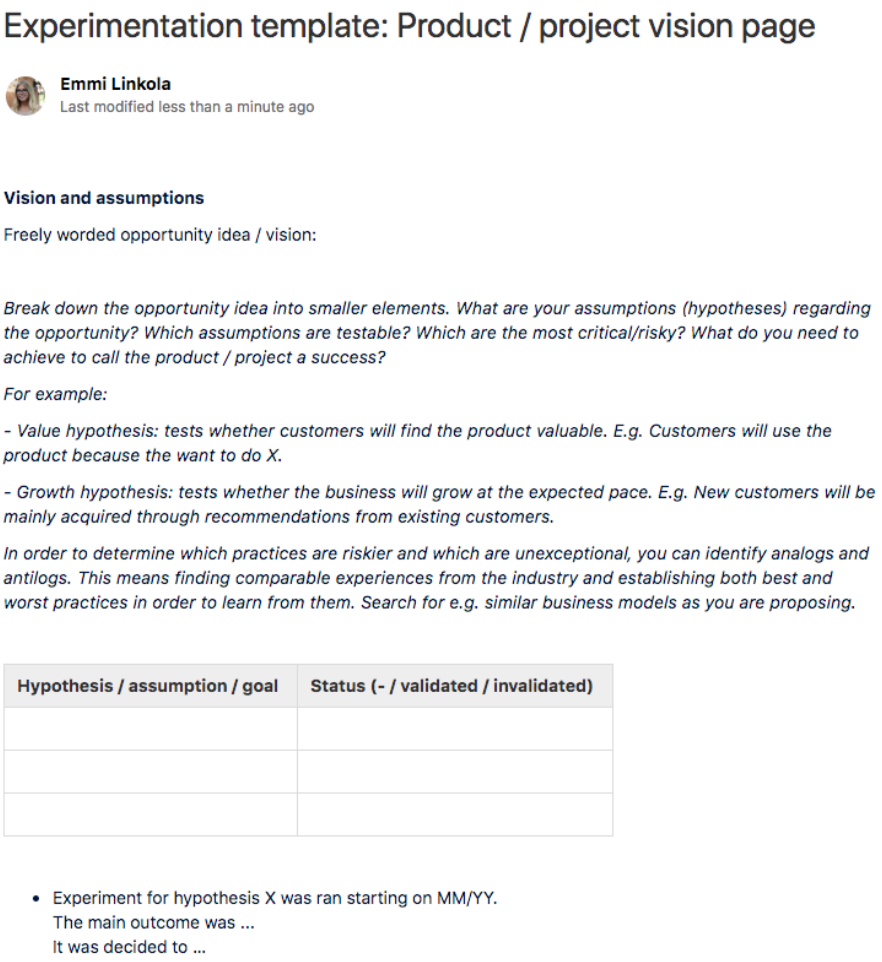
Once the experiment is ready, it is introduced to users and ran, while metrics are displayed, monitored and kept up-to-date on the **metrics dashboard**. The metrics dashboard includes the most important metrics related to the progress of the product or project. These can be metrics from measuring e.g. the value hypothesis and the growth hypothesis, which determine whether the product is valuable to its users and whether it will grow at the expected pace. However, the metrics dashboard is utilized for a longer time than only one experiment; therefore, it should contain metrics that are relevant for the longer-term monitoring of the product. The dashboard follows the change of the metrics on e.g. a weekly basis, to see long-term trends and the effect that experiments

have on the numbers. The dashboard is utilized in order to keep the metrics as accessible as possible even to employees who are not directly involved in the experiment. This also aims to help with keeping the progress visible and thus motivating employees through the feeling that progress is being made.

Analysis meeting

Finally, the results of the experiment are analyzed in an analysis meeting. The results are considered with regards to the overall vision for the product and relative to the opportunity idea, and it is decided how to proceed; whether there is need for a change in course, whether the product or project should be abandoned completely, or if the current direction of the product is performing sufficiently well. The same people who attended the planning meeting and were involved in the building and running of the experiment should take part in the analysis meeting.

Product/project vision page



Experimentation template: Product / project vision page

Emmi Linkola
Last modified less than a minute ago

Vision and assumptions

Freely worded opportunity idea / vision:

Break down the opportunity idea into smaller elements. What are your assumptions (hypotheses) regarding the opportunity? Which assumptions are testable? Which are the most critical/risky? What do you need to achieve to call the product / project a success?

For example:

- Value hypothesis: tests whether customers will find the product valuable. E.g. Customers will use the product because the want to do X.

- Growth hypothesis: tests whether the business will grow at the expected pace. E.g. New customers will be mainly acquired through recommendations from existing customers.

In order to determine which practices are riskier and which are unexceptional, you can identify analogs and antilogs. This means finding comparable experiences from the industry and establishing both best and worst practices in order to learn from them. Search for e.g. similar business models as you are proposing.

Hypothesis / assumption / goal	Status (- / validated / invalidated)

- Experiment for hypothesis X was ran starting on MM/YY.
  The main outcome was ...
  It was decided to ...

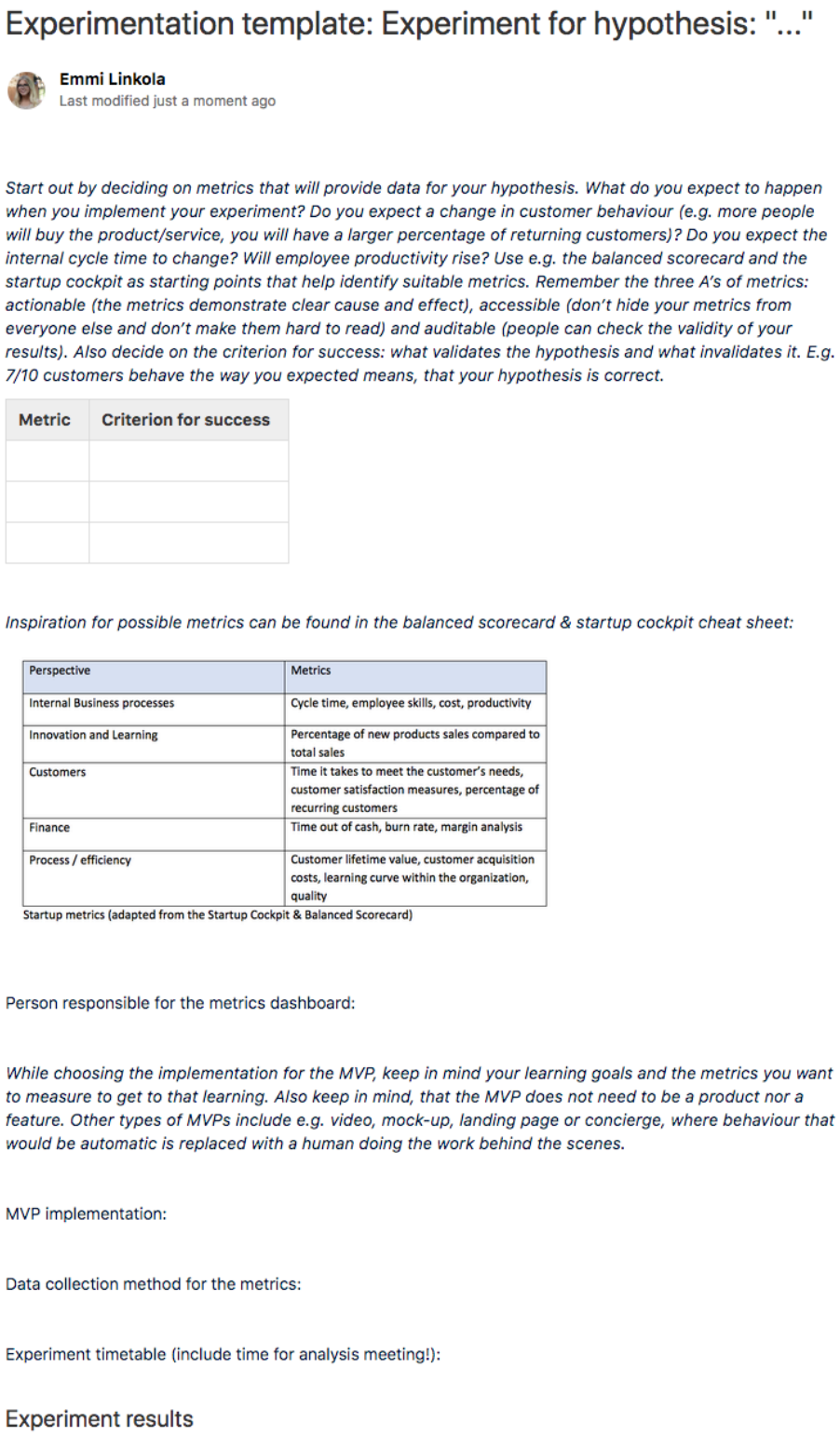
Figure 4.3. The template for the product/project vision page to be utilized in the experimentation process.

The purpose of the product/project vision page is to help with the planning of experiments and connect single experiments with the overall vision of the product or project. It serves as documentation of decisions that have been made and actions that have been tried, as well as the opportunity idea and hypotheses related to it. The product vision page is also meant to ensure that experiment results are utilized in decision-making regarding the product or project vision and its subsequent direction. The product/project vision page is stored in the case company's documentation tool of choice (Confluence). A copy of it should be made for each product or project that is experimented on, and the page should be filled accordingly.

Figure 4.3 illustrates a template that the author made for the product or project vision page. This template should be used as a basis for the page. The template page is mainly filled out during the planning and analysis meetings. It includes descriptions of the things to be filled in, as well as some additional tips that can help understand what to do during the first times that the experimentation process is implemented. The opportunity idea is filled in, when the first experiment for a product or project is planned. However, the opportunity idea is meant to evolve, and thus new versions of it should be added to the top of the page whenever the idea is adjusted or sharpened. The hypotheses related to the idea are included in the page as a table that also shows the current status of each hypothesis – whether they are validated, invalidated, or have not been addressed yet.

The product/project vision page is a main page, which includes a sub-page for each individual experiment. However, the vision page includes a list that summarizes all finished individual experiments, information about when they were ran, their main outcomes, and what actions were decided on based on the outcomes. These should be filled in after analysis meetings with a summary of what was decided there. When the development of the product progresses, the vision page and experiment sub-pages can be used to track and document the progress that has been made.

Experiment page



Experimentation template: Experiment for hypothesis: "..."

Emmi Linkola
Last modified just a moment ago

Start out by deciding on metrics that will provide data for your hypothesis. What do you expect to happen when you implement your experiment? Do you expect a change in customer behaviour (e.g. more people will buy the product/service, you will have a larger percentage of returning customers)? Do you expect the internal cycle time to change? Will employee productivity rise? Use e.g. the balanced scorecard and the startup cockpit as starting points that help identify suitable metrics. Remember the three A's of metrics: actionable (the metrics demonstrate clear cause and effect), accessible (don't hide your metrics from everyone else and don't make them hard to read) and auditable (people can check the validity of your results). Also decide on the criterion for success: what validates the hypothesis and what invalidates it. E.g. 7/10 customers behave the way you expected means, that your hypothesis is correct.

Metric	Criterion for success

Inspiration for possible metrics can be found in the balanced scorecard & startup cockpit cheat sheet:

Perspective	Metrics
Internal Business processes	Cycle time, employee skills, cost, productivity
Innovation and Learning	Percentage of new products sales compared to total sales
Customers	Time it takes to meet the customer's needs, customer satisfaction measures, percentage of recurring customers
Finance	Time out of cash, burn rate, margin analysis
Process / efficiency	Customer lifetime value, customer acquisition costs, learning curve within the organization, quality

Startup metrics (adapted from the Startup Cockpit & Balanced Scorecard)

Person responsible for the metrics dashboard:

While choosing the implementation for the MVP, keep in mind your learning goals and the metrics you want to measure to get to that learning. Also keep in mind, that the MVP does not need to be a product nor a feature. Other types of MVPs include e.g. video, mock-up, landing page or concierge, where behaviour that would be automatic is replaced with a human doing the work behind the scenes.

MVP implementation:

Data collection method for the metrics:

Experiment timetable (include time for analysis meeting!):

Experiment results

Figure 4.4. The template for the experiment page to be utilized in the experimentation process.

Experiments should each be documented on their own sub-pages under the product vision page. The experiment pages include more detailed information on the design of the experiments.

Figure 4.4 illustrates the template for an experiment page that should be filled in during the planning meeting. The heading of the page includes the chosen hypothesis, or if the experiment is meant to validate multiple hypotheses, a name that summarizes the purpose of the experiment. In the latter case, the hypotheses that the experiment tests should be listed at the top of the page. The template includes a table to which metrics and their criteria for success should be filled in, along with some instructions and examples for how to choose metrics. The template also has a "cheat sheet" that combines example metrics from the Balanced Scorecard and the Startup Cockpit to give some ideas of what the metrics could be from different perspectives. After the metrics have been chosen and filled in into the table, someone is chosen to be responsible for making (if it is the first experiment for the product or project in question) or maintaining (if it is not the first experiment) the metrics dashboard. The chosen MVP implementation is summarized into the page, along with data collection methods with which the data related to the chosen metrics will be obtained. This includes deciding whether the MVP will be instrumented, or whether data will be collected in another way. The experiment page template includes a place to fill in the timetable for the experiment in order to ensure that it is not forgotten to agree on a timebox for running the experiment and the date on which to analyze the results. In the analysis meeting, the results of the experiment should be filled in. They can be documented here in more detail, as a short summary will be made for the product/project vision page.

4.2.2 Example on implementing the process

This chapter will present an example process in order to illustrate the experimentation process in practice. The example focuses on the concept of personal budget. Personal budget is a model, in which municipalities make a care plan for a citizen together with them, which includes certain types of care and a certain amount of money to be used for that care. The citizen can freely decide which service provider they want to utilise to provide the care. Currently, service advisors in municipalities help citizens decide on a service provider by looking with them at a list of service providers who provide the services in question, which also displays the prices and locations for each service provider.

Opportunity idea

The opportunity idea in this example is to make a service that eases the process of selecting a service provider. Instead of having to look at a list of service providers to

choose from, the service would suggest a service provider and even handle the booking. The opportunity idea is based on the company-wide goal to differentiate from the competition by providing services that are easier and more effortless to use than other competing services.

Planning meeting

The planning meeting and its outcomes are documented on the product/project vision page, and the experiment design is documented in more detail on an experiment page created for this particular experiment.

Identify hypotheses

In the planning meeting, the opportunity idea is broken down into hypotheses based on the current understanding of the opportunity. This example will focus on the value hypothesis. The value hypothesis is the hypothesis which states how the solution will create value for customers. In this case, the value hypothesis could be that "the most important criteria when choosing a service provider are its price, availability and location". The new service would automatically choose a service provider by these criteria. Therefore, it does not allow for the user to use other criteria, such as reading reviews or visiting the websites of different providers, to decide on a service provider. If the price, availability and location are indeed the criteria that the citizen wants to base their decision on, then the service creates value by doing that work automatically for the citizen. Otherwise, the service will worsen the experience of the citizen, since they are not allowed to choose freely. Another hypothesis is that "personal budget recipients would rather have the service provider chosen for them than select it themselves from a list". These will both be tested in the experiment detailed in this example.

Select metrics

To measure whether the personal budget recipients would rather have the service provider selected for them, a number can be collected of how many personal budget recipients prefer the new system to the old way of selecting a provider from a list. In order to measure which criteria are the most important, the experimenters will collect a list of all mentioned criteria while the citizen is observed and interviewed making a selection, and the numbers of times each criterion is mentioned are tallied.

Plan MVP

We have already established that to collect the metrics, user observation and interviewing is needed. In addition, the citizens need to be able to compare the new system with the old in order to be able to say which they prefer. Therefore, the MVP in this example could be a digital mock-up of the user interface of the service, where one can input all the data the service would need to make a suggestion - the budget, the timeslot when the service is

needed and the location of the citizen - but that does not really do the calculation behind the scenes, but comes up with a mock result.

Build experiment

Building the experiment consists of building the mock-up of the user interface of the new system that suggests a service provider. To save time, this could be done with an online service, so that no development time is necessary. In addition, the preparations include finding and contacting a suitable number of test users, agreeing on a time and place for running the experiment and planning in more detail the questions that will be asked from the test users.

Run experiment

The experiment could be ran as follows:

1. The citizen is asked to search for a service provider with the existing system that lists out all services providers along with their prices and locations. They are asked to speak out loud while they do it, and especially explain what criteria they look at when they assess the suitability of a service provider. The observer asks questions to find out more if the citizen speaks too little or is too vague. The observer marks down each criterion that is mentioned.
2. Next, the citizen is presented with the mock-up of the service that will suggest only one service provider. They are asked to make a booking for themselves for the care that they need. Again, they are asked to speak out loud while they do it, so that the observer is able to take note if the citizen faces any frustrations in the process.
3. The citizen is asked which way they prefer to choose a service provider, and to motivate their choice.
4. Later, the experimenters tally the amount of times each criterion was mentioned, also taking into account if the citizen said anything about whether the criterion is very important to them or not so important, in order to determine a generalized picture of which criteria the citizens usually base their service provider choices on.

Metrics dashboard

The metrics dashboard reflects the change of metrics over time and through multiple MVPs. Therefore, it should include metrics that are relevant for a longer time than one experiment. Since the importance of different criteria when selecting a service provider is not likely to change, it is sufficient to measure it once and not add it to the metrics dashboard. However, the percentage of citizens who prefer the new suggestion system to the old list system can likely be improved on in subsequent experiments. Therefore, it can be added to the metrics dashboard. Later, if the system is developed and tried out in production, the same metric can be measured through measuring in real use how many citizens will choose to have the service provider chosen for them if given both options. In

addition, it can be tested how new changes that are made, such as adding more criteria based on which the service provider is chosen, affect this percentage.

Analysis meeting
Before the analysis meeting, the results are written down onto the experiment page in Confluence. In the analysis meeting, a short summary of the results is also written onto the product/project vision page. These results are discussed, and it is decided how they affect the strategy of the product and if changes in course are necessary. The key decisions are also documented on the product/project vision page. In the example case, if the results point to that the most important criteria for choosing a service provider are indeed price, availability and location, and if more than half of the test group prefer to have the service provider chosen for them based on these criteria rather than choosing from a list by themselves, the hypotheses can be considered validated. If the metrics show otherwise, however, a pivot may be considered.

4.2.3 Considerations for answering the current challenges

The suggested experimentation process was made keeping in mind the current challenges and mitigating them where possible, as well as aiming not to emphasize them. However, it does not help with every challenge that the interviews uncovered. The rest of this chapter will discuss in more detail the challenges that the experimentation process does aim to help mitigate, and which challenges need to be given special attention in order to avoid emphasizing them.

Challenge category	Hypothetical benefit from the experimentation process
Lack of a shared vision of the goals of the project and the alignment of the project relative to company-wide goals	The product/project vision is linked to each experiment during the planning and analysis meetings and documented in the product/project vision page
Little user research and lack of feedback collection	The experimentation process includes running the experiment with real customers and end users

Table 4.8. The possible benefits specific to the case company as based on its current new product development process challenges.

Table 4.8 presents the challenge categories that experimentation could help the company solve. It also details how experimentation would be beneficial with regards to each challenge category.

The product/project vision page is aimed toward creating a shared understanding of the vision and goals for each project or product, as well as each experiment and their effect on the direction of the project. Writing down the vision and goals also helps later, when experiment results are tied to decisions about subsequent actions. However, aligning projects relative to company-wide goals is something that has to be addressed already when deciding whether to pursue new projects or not. Company-wide goals should, though, be considered when making pivot or persevere -decisions.

Experimentation necessitates user contact and testing ideas with real users. In order to start doing this, the case company first needs to work on establishing contact with users who can participate in testing. Some experiments can be conducted through simply instrumenting products with metrics and following the progression of those metrics, without actually talking to the users. However, this is not always the case, and ideally these two methods should be combined to produce both quantitative and qualitative data. Experimentation is likely to increase the amount of user research where the purpose is to test the validity of a solution. The case company can be recommended to ensure that also the problem that the users have is well understood through user research methods that aim not just to test a solution but deepen the understanding of the problem.

The understanding that the experimentation process brings about the use cases of the product can help in making work estimates for the final product. However, estimating the work needed for building each experiment can remain as difficult as before. Clients may also want higher-level work estimates before any experimentation is done. Therefore, work estimates will likely continue to pose a challenge, although it is a known problem which the case company is actively working on.

Adopting the experimentation process is not likely to solve prioritization or communication issues. It might make communication issues more evident, however, since in customer projects the consent to conduct experiments, reasons for experimenting, and how the experimentation might affect the customer all need to be taken into account and communicated appropriately in order to not cause frustration for the customer. Selling experimentation projects also requires developing different skills than are needed in selling fixed-price, fixed-scope projects like the case company has done before.

Challenge category	Hypothesis of why experimentation would emphasize the challenge
Communication issues	Experimentation processes require close communication with business-to-business and business-to-government clients. They might need to interrupt their normal work to participate in the experimentation process.
	The data that is collected in an experimentation process must come from real end users. Reaching them might be harder since they are the customers of the client, and it is likely to require communication and co-operation with the client.
Work estimates much smaller than the realized outcome	Experimentation aims to find out what to build with minimal effort. If too much effort is spent on every experiment, the process will not be cost-efficient. The challenge being emphasized is not really making work estimates, but that the outcome of effort that was spent is consistently larger than anticipated.
Challenges related to how the project was sold	Experimentation processes aim to create knowledge of what to build. Since what will be built is not known beforehand, it is not possible to know the exact schedule and budget of projects either. Therefore, selling projects in iterations is even more crucial and requires the company to develop new capabilities.

Table 4.9. Challenges that the case company already faces that are especially important to acknowledge and address because an experimentation process may emphasize them

4.3 Reflective learning: continuously improving the process

This section builds on top of the action intervention phase and summarizes learnings from it. In this chapter, an improved version of the proposed process is introduced, illustrated with an example, and next steps are suggested for the case company.

4.3.1 Improved process

After the author had introduced the process proposed in the action intervention phase to the case company, potential projects for implementing the process were considered. The most potential project in which to implement the process was a project that would be starting in the future; there was only an initial idea of what the product would be, but the idea was not properly fleshed out. Reviewing the process in the context of this project led to acknowledging that the process did not yet cover the entire product development process but only a part of it. The experimentation process could be repeated many times during a project, but there were steps that needed to be done before the experimentation could be started. The author iterated on the process based on this understanding and made an improved proposal. Figure 4.5 illustrates the improved process.

The improved process presents the new product development process from the beginning, before actual experimentation is started. The process begins with an original high-level opportunity idea that should be linked to the company-wide goals, but that does not describe the problem it solves or the solution it offers on a sufficient level. Therefore, it needs to be refined so that it includes both a problem and a solution on such a level that they can be pitched to a potential customer. Once a pilot customer has been acquired, some user research is likely to be necessary to gain a proper understanding of the problem. This can now be done in cooperation with the pilot customer. Once the problem is understood well enough to start developing minimum viable products, the first experiment can be started as described in chapter 4.2.

The rest of this chapter will discuss in more detail the differences between the improved process and the original process proposal. The main difference is the steps added before the first planning meeting.

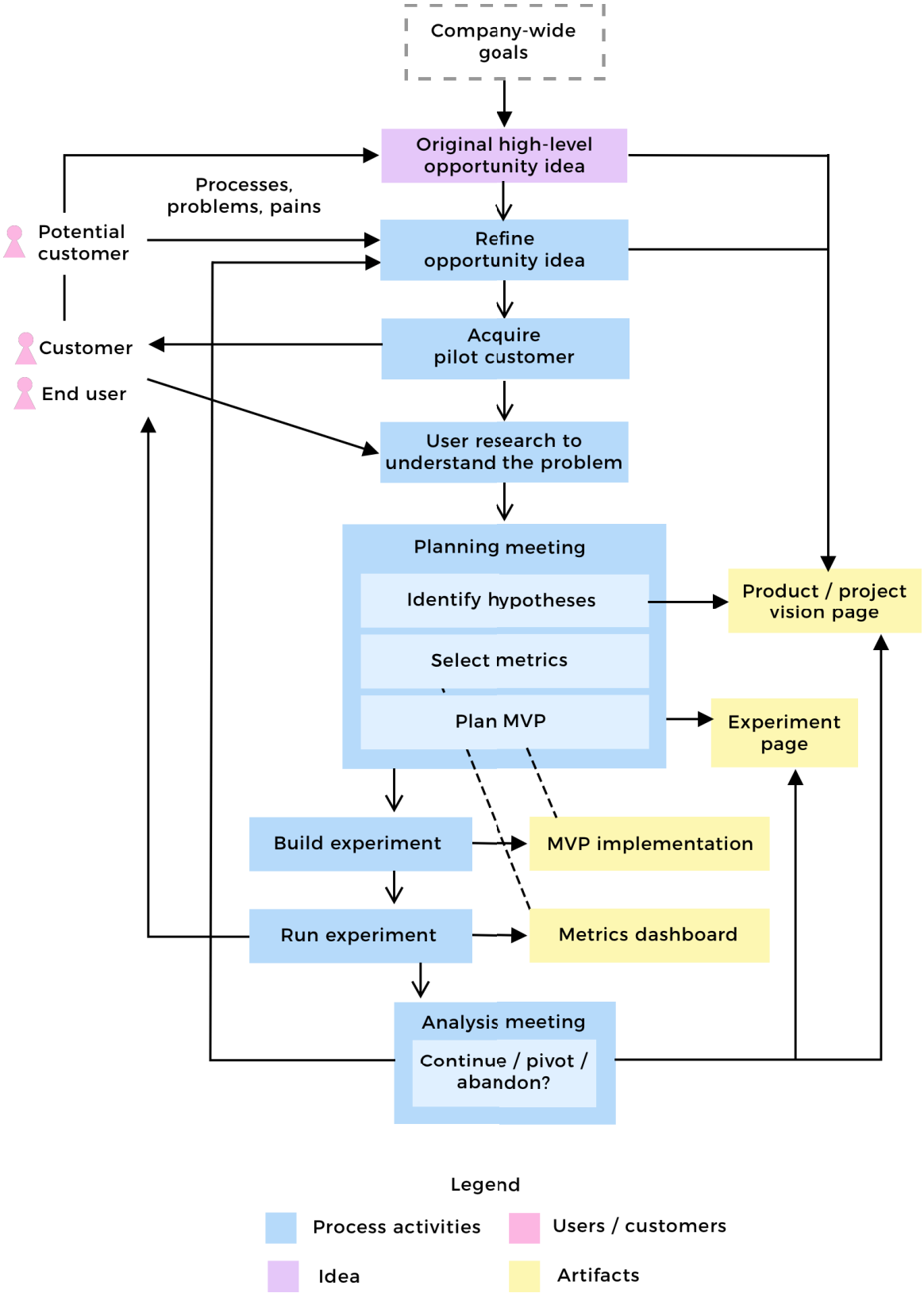


Figure 4.5. An improved version of the experimentation process.

Original high-level opportunity idea

The improved process recognizes that the opportunity idea is in the beginning often first an original high-level opportunity idea that is a vague idea which needs to be sharpened into an idea that is clear enough to explain the opportunity at hand in more detail. An opportunity idea should describe an existing (though hypothetical) problem, as well as a potential solution to it. However, the original high-level opportunity idea lacks depth in describing both the problem it solves and the solution it offers. Still, it provides an idea that is interesting for the case company to explore further. For example, the idea could be to solve a problem – or multiple problems – that customers might have, but that are currently not understood well enough to describe how to solve them. Alternatively, it could be an idea for a product, without an understanding of what problem it actually would solve.

Refining the opportunity idea

Since the original high-level opportunity idea describes the opportunity vaguely, it needs to be refined through some background research. The case company feels this needs to be done, because in order to understand problems in their domain, they need to have a customer with whom they can do user research and subsequently experiment with solutions. However, the case company suspects that their potential customers may not be willing to e.g. let themselves be observed by a user researcher while working, unless the company already has a contract with them. In the domain of the case company, acquiring a customer is a long process, and the case company wants to have an idea of what they are offering when they start selling pilot projects to potential customers. Therefore, in order to be able to acquire a pilot customer, the case company wants to have a clearer view of what they can offer to them, and this is why the opportunity idea is refined. This can for example mean doing some market research to connect the original high-level opportunity idea into the specific problems that potential customers have.

Acquiring a pilot customer

The case company presumes that to conduct user research in their domain, they need to co-operate with a pilot customer municipality with whom they have a contract. Without a contract, customers might be reluctant to help the company understand their processes and e.g. let the case company observe them while they go through their payment processes. Therefore, having a customer for a pilot project enables the case company to really understand the processes their product needs to support, through user research and later also testing the viability of their solutions with the customer.

Acquiring a pilot customer is essential for two reasons. First, the case company needs to be able to do some user research that will help them understand the entire problem. The problems that the case company solves are not simple enough that solutions could be formulated without the help of a customer who knows the processes. Second, experimentation is not possible without access to real users. Having agreed with a

customer beforehand that the solution will be developed in iterations utilizing experimentation may help in co-operating with them. Additionally, end users can be easier to reach through the customer, since they are the customer's customers, although in some cases the customer is also the end user.

A pilot customer could be found e.g. through the research done while refining the opportunity idea. The pilot customer should be a customer that is especially interested in gaining a solution to the problem in question and is willing to put in some effort in order to help with the process.

User research to understand the problem better

Before starting to develop a solution through experimenting, a better understanding of the problem, its entire context, and the processes of the customer is needed. The processes and problems related to them tend to be so complex that a potential – even partial – solution cannot be built and tested until a better understanding is acquired. It should be noted that the processes for doing similar things may vary between customers, but the case company wants to start with developing a solution that fits one customer, while also aiming to make it as generalizable as possible so that it fits as many customers as possible.

4.3.2 Example on implementing the improved process

This example is based partly on some work that the case company and the author have already done. The original high-level idea that the example uses is a real idea that the case company has for a project. However, the process has not been completed with this idea; the case company is still refining the opportunity idea. This section offers an overview of how the first steps have been conducted and a hypothetical example of how the process could proceed.

The case company currently offers a solution to the service voucher payment model for Finnish municipalities, as well as solutions for implementing freedom of choice tools with which citizens can choose private healthcare providers as the providers for their public healthcare. However, the goal of the company is to become the go-to solution for healthcare payments from the public sector to the private and support multiple payment models instead of just one. This is an area where the case company wants to try experimentation: with new solutions to other payment models than it previously has offered. Therefore, the high-level opportunity idea of the case company is that it wants to offer solutions to more payment models for public-to-private payments in social and healthcare than it currently does. However, the company does not yet know, which payment models municipalities would be the most interested in, or for which ones they currently have the most lacking solutions. In addition, the case company cannot start

developing solutions without understanding the processes that municipalities have related to each payment model.

While refining the opportunity idea, the case company decided it would be best to focus first on developing one additional solution to one additional payment model. Even though the aim is to ultimately offer solutions to several of them, the models can require quite different solutions, and therefore only one solution should be developed at a time.

In order to refine the opportunity idea, the case company wants to know which payment models it should first focus on. Therefore, it needs to know what payment models their customers, i.e. municipalities, are most interested in. The case company already has some hypotheses about which payment models are challenging to arrange for municipalities. However, there are more factors at play. Since the legislation will soon be changing regarding some models, the changes may mean that municipalities need to arrange the payments differently. Therefore, the case company also wants to know if the municipalities think this will affect the billing of payments: whether the new legislation makes it more difficult – which is what the case company suspects – or whether municipalities think it makes no difference or even eases the billing of payments.

The author worked on refining the opportunity idea by making a survey for social and healthcare leaders in municipalities, in which they were asked to evaluate how interested their municipalities would be in implementing different payment models in certain scenarios. The survey questions are included in Appendix C. The questionnaire was divided into three parts: background questions about the person answering and their municipality, questions regarding a scenario where the social and healthcare reform that is currently being considered in Finland does not actualize, and questions about one payment model in particular and how new legislation would affect it. The purpose behind the questions about the scenario where the social and healthcare reform does not happen was to gouge interest that municipalities have for implementing or testing different payment models even if they are not required by law to do so. The case company wanted to ask about one payment model in particular: customer fees. They had a hypothesis that billing customer fees is currently the hardest challenge to municipalities and will become even more challenging if new legislation passes.

26 social and healthcare leaders answered the survey. Interestingly, the results are not in line with the assumptions of the case company. The municipalities do not seem to find that new legislation would make billing customer fees more difficult. In general, the results indicate that the most interesting payment model that municipalities would like to implement in the near future is personal budget, in which citizens are given a certain sum of money by the municipality to spend on certain private social and healthcare services. There was some variation between the answers of municipalities that currently use the service voucher payment model, and companies that currently do not use it. Municipalities that currently do not use service vouchers, were most interested in implementing a service voucher model which has a bonus/sanction element related to it.

However, only three of the respondents' municipalities did not yet implement payment services with service vouchers, and therefore these results may not be generalizable.

In order to arrive at a refined opportunity idea, the case company should next choose how to proceed. The survey had a possibility for respondents to opt in to being contacted by the case company, and a few municipalities opted in. The case company can call these municipalities and ask more about their answers and their thoughts regarding the payment models. The case company could even find a pilot customer among these municipalities. The case company should decide which payment model to pursue based on the survey results – they should indicate interest in the model in more than just a couple municipalities – and also based on what potential pilot customers are most interested in, since the experiments cannot be conducted without a pilot customer. This is their refined opportunity idea. Acquiring the pilot customer can be done at the same time even though it is a separate step in the process. The pilot customer could be an existing customer that wants to try a new payment model, one of the respondents of the survey, or acquired through other means.

Once the opportunity idea is refined, i.e. the payment model for which to develop a solution has been selected, and the company has a pilot customer to work with, the case company can start conducting user research to understand the processes that the customer has related to the payment model. The user research can for example take the form of observing the employees of the pilot customer as they work through their usual process of handling the payments and asking clarifying questions. It is also likely that the customer needs to actively participate in outlining and visualizing the payment process to help the case company gain a better understanding. Going to the premises of the customer was regarded as a good practice in a previous product development project, so the case company should keep doing this. This way, the work of the customer is interrupted as little as possible, while communication can still be direct and trust is easier to establish between the parties.

After the user research, the case company should be able to start experimenting with potential solutions. This can be done in a similar manner as the example in chapter 4.2.2 illustrated.

4.3.3 Next steps

The very first next step for the case company is naturally to try out the process in practice. The first rounds of experimentation should be implemented in a fairly lightweight way, since there is a lot to learn and practice already in the implementation of the process itself. Once the case company has tried out experimenting for the first time, it is advisable to review the experimentation process and make improvements based on the gained learnings. This will enable greater customization of the process to suit the particular needs and processes of the case company. In subsequent experiments, the process can be

gradually refined and more effort can be invested once practices that work well have been established. The proposed next steps for the case company are outlined below:

1. Choose an experimentation topic that is related to whether value is delivered to customers, rather than validating market demand
2. Implement the process in a lightweight manner
3. Iteratively improve the process through experiment retrospectives, where the team assesses what worked well about the process and what could be improved
4. Develop more refined experimentation practices regarding e.g. tools to use for instrumenting products or analyzing collected metrics

The topics for the first experiments should be concerned with whether value can be created to customers, rather than whether many customers want to buy the solution. The value that a business creates refers to the worth of the offerings of the company as perceived by its customers. By focusing on improving their value creation, the case company can ensure that they are improving their product in the direction where the product better matches the needs of their customers. The case company should experiment on validating value creation rather than market demand because of the environment they operate in. The environment the case company operates in is characterized by very high customer acquisition costs, and new customers are acquired quite seldom. Validating market demand would not produce reliable results in such an environment, since the acquisition rate grows at a very slow pace. Any type of experiment that measures the interest of potential buyers is likely to be flawed, since the potential buyers are hard to reach and they need much information to be able to make a real purchase decision. The case company could measure interest in a manner that does not require the making of a purchase decision, but the results of such an experiment could be misleading and not indicative of actual interest to buy the solution if it is developed. If experimenting through measuring the customer acquisition rate, the number of potential buyers would be so small that any differences in the numbers between different time periods could simply be caused by chance and not by the actions of the case company. Therefore, the case company is recommended to conduct at least their first experiments with topics that are related to whether they can deliver value to customers, rather than whether there is market demand for the solution – in other words, to concentrate on the value hypotheses more than the growth hypotheses. Once the case company has learned how to experiment in practice, they may find a way to reliably experiment with growth as well.

Once the case company has found a suitable topic for experimentation, they should implement the proposed process in a lightweight manner. This means that e.g. the collection of metrics should not involve significant work done beforehand to build infrastructure for instrumenting projects. In the beginning, metrics that are less cumbersome to collect can be selected, or the data collection method can be adjusted to

collect the metrics in a way that involves less work. For example, collecting feedback through active usage during user testing can take less effort than developing a reusable infrastructure for collecting metrics from passive usage, in which the users are not aware that they are participating in a test and the metrics are collected solely through the instrumented product. However, the case company has been developing a system for collecting business analytics; this can be reused in experiments as well.

When the experimentation process is applied in real life, the case company is likely to learn more about which practices work well for them, and which parts of the process are not so suitable. Therefore, it can be suggested that the case company introduces retrospectives for the project team. Currently, the case company only has retrospectives for the development team; however, experiment retrospectives could help streamline the work of the entire team that is involved in experimentation. The retrospectives can be held e.g. after each experiment. It should be kept in mind that the experiments should not run for overly long durations. The aim of the retrospectives is to improve the experimentation process through analyzing what went well and what needed improvement. The retrospectives should include formulating action points assigned to specific people to implement possible improvements, which is a practice that the case company has in the development retrospectives, and that is widely accepted in the industry. It helps ensure that actions are taken in order to improve, and improvements are not only talked about.

After the first few experiments, the case company can make the process more systematic by investing more in the experimentation process through e.g. developing practices and infrastructure for instrumenting products and analyzing results. The case company is already working on some infrastructure for collecting business analytics; once it is ready, it can be utilized in experimentation as well to collect measurements. The proposed process does not yet include any analysis tools to be used in analyzing the collected data. This is because the author recommends the case company to start with a lightweight process and metrics that are easy to collect; therefore, the data analysis should also be straightforward enough to not need new tools for analysis. As the company gets more comfortable with the experimentation process, however, and depending on what kinds of metrics are collected, analysis tools could become very beneficial later. Google Analytics is a service that the case company is already familiar with, and therefore it can be utilized for metrics collection and analysis without too much effort spent getting familiar with a new tool. Other data collection and analytics tools that can be recommended for the case company are Amplitude or MixPanel, both of which are tools specialized in collecting and analyzing data from web-based services and producing insights to e.g. what users are doing on the site and how different functions on a site are performing (compared to each other).

If the case company finds it useful, they can also consider dividing the roles in the experimentation process to different people, which would allow people to specialize in

one of the roles. The different tasks of creating and iterating the product/project vision, designing, executing, and analyzing experiments, and developing the minimum viable product can be divided to different roles. For the case company, having too many people involved in the process may not be beneficial considering the small size of the company; however, some of the roles can be combined and handled by the same person.

Next steps for answering the current challenges

In order to take actions for the challenges that could be improved upon outside the experimentation process, the author held a retrospective session for the case company where the author presented the six clusters of challenges found in the interviews and each individual challenge related to them. The challenges were discussed together and some action points, best practices and "not to do's" were formulated.

The case company had already made note about the need to sell projects in iterations. This was discussed again in the retrospective. It came up that selling projects in iterations to clients who have never bought projects like this before can be challenging. The case company recognized the need to develop relevant capabilities for this, although no concrete action points were formed. The communication and involvement of different functions within the company in the sales process was discussed as well. It was decided to both involve development in sales discussions more actively, as well as to involve sales (or more generally the business side) more consistently in development projects. One concrete way to do this was discussed: to have product owner pairs, where one person is from the business side and the other more focused on development.

Selling iterations was also seen as useful with regards to estimating work. Making project proposals in small chunks could enable the case company to estimate work on shorter timespans. In addition, the case company thought that it could be a good idea to measure task completion times in a lightweight manner in order to achieve better accuracy for work estimates. Some of the "not to do's" formulated included estimating work effort for long time periods and making technological choices in a hurry.

User research was discussed at length. Establishing systematic user contact instead of only ad-hoc communication with end users was seen as a valuable effort to undertake. Systematic user contact would include establishing contact to multiple user groups, rewarding lead users for their participation, having a process in place about how and when to contact users, and utilizing the results also in sales and marketing. An action point was formulated that the UX designer of the case company would make a list of all users that she knows can be contacted for user research purposes to the documentation tool that the case company uses (Confluence). New ways of acquiring test users were also discussed; e.g. through the company's weekly newsletter by offering a reward for those who participate in testing and through user trainings. Analytics also came up as a way to get passive feedback from end users.

As a process improvement, the case company came up with the idea to make an initial project checklist to be utilized in the beginning of new projects. It would include deciding e.g. how customer service is handled, who the product owner pair is, how end user contact will be handled, etc. This idea was based on the discussions on e.g. communication with customers and end users, as a way to make sure the problem areas are all taken into account in the process in the future. The need for such a checklist also arose from the realization that when taking on new projects, it is not always possible to simply adjust the current processes to fit them. All functions of the company may need to implement new processes in order to handle the projects as efficiently as possible.

5 Discussion

The study investigates utilizing experimentation to improve the new product development process at the case company. The research questions address the challenges in the current process, what an experimentation process that improves the process should look like, and what suggestions can be given to the case company in order to achieve systematic experimentation.

5.1 Challenges in the current process

Six different clusters of challenges were found in the new product development process of the case company through interviews that addressed a specific previous project. Each of the clusters consisted of related smaller challenges. Challenges related to how the project was sold formed one of the clusters. This cluster was closely related to another one, which consisted of the challenges posed by that work estimates were much smaller than the realized outcome. In addition, there was a cluster that consisted of problems caused by the lack of a shared vision within the team for the project, and the alignment of the project relative to company-wide goals. Other clusters included the lack of user research and feedback collection, challenges related to prioritization decisions, and communication issues.

The previous project, which the interviews addressed, was the case company's first new product development project. Thus, it provided a valuable learning experience, and many of the identified challenges arose from the inexperience with similar projects. Since the case company had only recently been formed, its strategy was not initially completely clear and still evolved during the project, which was a root cause for many of the problems faced in the project.

One of the challenge clusters most affected by the newness of the company was **challenges related to how the project was sold**. The company was merging with a startup at the time of the selling of the project, and the people were not familiar with each other and the roles of each other yet. This was one reason why the project was sold without proper involvement of developers in the discussions. Selling the project with a fixed price and ill-defined scope aggravated the problem. As the team had an incomplete understanding of the items that the client wanted to include in the scope, the client was originally promised a scope larger than the case company could deliver. Once this was realized, the scope was re-negotiated. At the end of the project, the case company was well aware of the challenges related to selling projects. To alleviate them, it decided to aim for selling projects in iterations instead of fixed price and scope. In addition, the case company decided to enforce closer co-operation between sales and development through e.g. involving developers in all sales discussions for projects that are likely to involve

development. Another measure that was decided on was to make product owner pairs in which one person is a developer and the other focuses on business considerations.

Work estimates formed another challenge cluster closely related to the problems that arose from how the project was sold. The initial estimates were made with an incomplete understanding of what the items in the scope meant, and without a proper understanding of their real-life use cases and the problems users might face in those cases. Another cause for the large difference between the estimates and the outcome was that the role of the product fluctuated between a quick proof-of-concept MVP used only for a short time and a longer-term product. The estimates were done thinking the product would be developed very quickly, sacrificing technical robustness for development speed. The estimates also took into account that there were some existing services and assets that could have been re-used, but in reality they were not. The biggest challenge, however, was posed by the initial work estimates, and therefore the case company decided not to try to estimate work for overly long periods of time anymore. Estimating the work effort for each iteration was easier and more accurate, although the case company decided to aim to improve the shorter-term estimates as well through measuring development task completion times in a lightweight way.

The lack of a shared vision of the goals of the project and the alignment of the project relative to company-wide goals affected several aspects of the project. Since the goals of the project were not clearly understood by everybody and fluctuated during the project, the product became less reusable than originally was intended. The unclear purpose of the project led to confusion of whether the role of the product was to serve as a lightweight, quick proof-of-concept MVP, or a longer-term product. In addition, as the company matured and set business goals for the company as a whole, focus shifted from making solutions to customer projects to building own products. While the change in focus was not a problem per se, it meant that the previous project was a good learning experience but not an exercise in building the own products that the company wants to build from now on.

One of the most persistent challenges in the new product development process of the case company was contact with end users and the **lack of user research and feedback collection**. In the previous project, user research was only conducted with one user group, while feedback collection was not done at all. Therefore, there was no clear understanding of how the end users and customers perceived the final product, or what the real-life use cases of the product were. Understanding the use cases could have helped alleviate some problems that end users eventually faced. Based on these findings, the case company decided to establish user research and testing practices, build relationships with lead customers and reward them for participating in user tests. Collecting analytics is also something the case company wants to explore to get indirect feedback from users.

User feedback could have greatly improved the **prioritization challenges**. The biggest challenges that prioritization posed were decisions of whether or not to build

some features. Eventually, not building certain features led to excessive manual work for both the case company and the client. Prioritization was also a challenge relative to other projects that the case company was working on. As the development team was small, only four people, work on the project in question needed to be paused in the middle of the work when some other project needed acute attention.

The project also had some **communication issues**. These are important to be aware of, since experimentation can bring additional challenges to communication. Experimentation projects typically require even more understanding and co-operation from the customer in business-to-business projects (Lindgren and Münch, 2016). In the current process, the customer would often have wished for faster response times. The customer also provided customer service to the end users, which they found difficult, since they did not have the same knowledge about the system as the case company. Sometimes the customer needed to advise end users to contact the case company instead. Some end users grew frustrated when they did not get a quick enough response to their queries from the case company. According to the case company, this was likely caused by that it was not always clear who was responsible for answering the end users. To alleviate this challenge, the case company thought it might be a good idea to develop a project checklist which to use in the future to ensure that the responsibilities have been agreed on. The checklist would include things such as how to handle customer service, who the product owner is, and how end user contact is ensured. It was also recognized that in the public healthcare domain many of the customers are likely to be inexperienced with software projects, and clear communication from the case company is especially crucial to prevent problems.

5.2 Proposal for an experimentation process

The author proposed an experimentation process to the case company that could help improve their current new product development process. This process was formulated based on the challenges found in the interviews, the literature review, and keeping in mind the current process in order to make the experimentation process relatively easy to adopt.

The experimentation process aims to alleviate two of the challenges currently faced by the company. It gives user research a large emphasis, since that is what validates or invalidates assumptions about user behavior. This would help the case company understand better what its end users value and how to develop products to a direction where they are even more valuable to their users. The proposed experimentation process also includes consistently documenting the vision for a product and the goals of each experiment, which is meant to ensure that everyone shares the same understanding of the goals. Writing the vision and goals down also aids in tying experiment results to decisions about future actions.

The proposed experimentation process starts with an opportunity idea. The opportunity idea is a concept presented by Hassi and Tuulenmäki (2012), which refers to an idea of a problem and a possible solution for how to solve it. In the beginning, this idea is on a high level and might not be very refined. In The Lean Startup, Ries (2011) makes the assumption that entrepreneurs have an opportunity idea – although he does not use this word – that they can start with. He also does not offer solutions for how to come up with opportunity ideas. However, this could be where to apply Steve Blank's (2013) idea of "getting out of the building", i.e. the opportunity idea could come from user research or other customer contact. In addition, new ideas could arise from the knowledge gained from completed experiments. The proposed experimentation process takes into account the possibility of the opportunity idea being vague in the beginning of the product development process; in the beginning, the idea does not necessarily describe the problem it solves or the solution it offers on a sufficient level.

The opportunity idea is refined in the next step of the process. The goal of this is to make the opportunity idea clearly describe a problem and, on a general level, how it would be solved. The opportunity idea needs to be refined enough so that a potential customer can clearly see how they would benefit from the solution and what problem of theirs it solves. In addition, the case company wants to preliminarily know whether pursuing the idea makes sense. The refining of the idea could for example take the form of market research, which is conducted to understand what the most pressing problems are that customers face related to a certain topic.

As a result of refining the original high-level opportunity idea, the case company should have a clearer picture of the problem that they will be solving for customers. However, in the domain of the case company, even initial, low-fidelity solution proposals cannot be developed without a deep understanding of the processes that the customers have and the problems they face. Therefore, some user research is necessary. In order to conduct user research, the case company needs to partner with a pilot customer that will help the case company properly understand the problem and the related processes. The case company suspects that they cannot conduct user research without a pilot customer, since the research requires time away from normal work for the user who participates. Having a pilot customer who is especially eager to acquire a solution for their problem and is willing to spend time in participating in the product development process can be very valuable, similarly to how lead customers can be very helpful in experimentation in business-to-business contexts (Rissanen and Münch, 2015; cited in Yaman et al., 2016). However, it should be noted that finding such customers can be difficult (Rissanen and Münch, 2015; cited in Yaman et al., 2016). Another known challenge is persuading the customers to spend their time participating in the process or getting access to end users through customers, in cases where the customers are not end users themselves (Lindgren and Münch, 2016).

Once the problem area is well understood, the next step is to have a planning meeting, in which people with relevant skills are involved. The outline of this planning meeting is stored on a product/project vision page for which a template was developed by the author. The template is stored in the case company's documentation tool of choice, Confluence. In the meeting, the product/project vision page template should be filled out to serve as documentation for both the vision for the product or project and all associated experiments that have been conducted. This practice is based on a combination of sources from literature and the existing ways of working at the case company: the case company already uses Confluence for documentation, and Fagerholm et al. (2014) stress the importance of documenting experiments in order to reuse the gained knowledge. The product/project vision page also aims to help the company get through the experimentation process for the first few times and serve as a guideline for what needs to be decided, planned and done. This can help lessen the amount of effort the case company needs to spend on learning the experimentation process; Yaman et al. (2016) found that first experiments are usually seen to take much effort. Subsequent experiments can be easier since they are able to reuse experiment designs and guidelines (Yaman et al., 2016), which is why the process is documented so that it can easily be reused.

In the planning meeting, the build-measure-learn loop is planned in reverse order, as recommended by Ries (2011). The first task is to break down the opportunity idea into smaller hypotheses that can be tested. To make formulating a hypothesis easier, it can be thought of as the action that the team assumes the user group to take, as well as the reason why they would do so (Hassi and Tuulenmäki, 2012). The most risky and critical hypothesis should be selected as the basis on which the experiment is designed (Ries, 2011). Multiple hypotheses can also be chosen, if they can easily be tested with the same minimum viable product. The experiment design begins with identifying what learning needs to be achieved to validate or invalidate the chosen hypothesis and choosing metrics that represent that learning. Appropriate success criteria should also be defined for the chosen metrics. Then, the implementation for the minimum viable product is decided on based on how to best capture the necessary metrics. Inspiration for potential MVP implementations can be sought from e.g. Bosch's (2012) suggestions for suitable different techniques for gathering customer feedback depending on how far the product has been developed. Bosch (2012) suggest different techniques depending on whether the product is in the pre-development, pre-commercial deployment or commercial deployment phase. Some suitable methods are advertising for products in the pre-development phase, product alphas in the non-commercial deployment phase, and performance metrics or surveys in the commercial deployment phase. To conclude the planning meeting, the responsibilities for building the minimum viable product and for making a metrics dashboard to be utilized while the experiment is running are assigned to people. A timetable for the experiment is also agreed on, including a timeslot for another meeting where results will be analyzed.

Subsequently, the experiment is built and ran with real users, and the plan is adjusted as necessary. Lindgren and Münch (2016) recommend utilizing some project management tools already in the first experiments but keeping them simple. In the proposed project, this is implemented in the form of a metrics dashboard, which is a tool that Ries (2011) recommends to help monitor how experiments are affecting key numbers over time. The metrics dashboard should be kept simple in the first experiments and be updated regularly, e.g. once a week.

Once the results of the experiment have been gathered, they are discussed in the analysis meeting. If much analysis is needed to interpret the data, then that should be done before the analysis meeting. However, the collected data should in the first experiments be simple enough to not need extensive analysis. In the analysis meeting, the team again takes a new look at the product/project vision page and assesses how the results of the experiment affect the product vision. In this way, the product/project vision page is meant to ensure that experiment results are linked to decision-making regarding the product vision and strategy, since linking experiments with decision-making is one of the characteristics of systematic experimentation (Lindgren and Münch, 2015). Based on the analyzed results and knowledge gained from the experiment, the decision is made how to proceed with the product: whether to continue on the current course, adjust the direction – make a pivot – or to abandon the product.

Compared to the process presented by Ries (2011) in The Lean Startup, the proposed experimentation process includes more practical guidelines. Additionally, it gives an opinion on where opportunity ideas can come from and presents some steps to take before the first MVP is developed for companies, for which jumping straight to developing solutions is not feasible due to the complexity of the problems they intend to solve. In summary, the proposed experimentation process shows where and how the case company can start the Build-Measure-Learn cycle.

5.3 Suggestions for the future

Companies that start experimentation can in the beginning find the experimentation process cumbersome (Yaman et al. 2016). Therefore, the case company can be suggested to **start implementing the experimentation process in a lightweight manner**. This means e.g. choosing metrics that are easy to collect, not using new tools for data analysis, and not running overly long or ambitious experiments. In general, the company should learn more about what works well in the experimentation process and what does not, before investing more in the process, and this can only be done through experience. Experience will also help make the process quicker, as Hassi and Tuulenmäki (2012) note. This is a side effect of the increasing sense of ownership and responsibility that teams tend to have over the experiments (Hassi and Tuulenmäki, 2012).

The case company should choose to **conduct their first experiments on topics related to the value that they deliver to customers rather than market demand**. Market demand is very hard to accurately and reliably measure in the environment of the case company, where customer acquisition costs are high and customer acquisition happens slowly. This is because customers require much information and time before they can make a purchasing decision. Utilizing metrics related to whether customers would buy solutions would lead to the problem that the metrics could not be trusted. The potential buyers would either have to know that they are not really making a purchase decision, which could skew the numbers, or alternatively, the number of buyers would be so small that any differences in this number between different time periods could simply be caused by chance and not by the actions of the case company. Therefore, the case company should choose the topics of their first experiments so that they are related to validating value hypotheses rather than growth hypotheses. Once the case company has learned how to experiment well, they can try experimenting through measuring market demand, if they find a reliable manner to do so.

In order to ensure that the case company continually improves the experimentation process, the author suggests that they **introduce regular experiment retrospectives**, where the team assesses which practices in the process worked well and which should be improved. The entire team that was involved in the experiment should attend the retrospectives. The case company already has bi-weekly retrospectives for the development team, where the developers discuss how to improve their ways of working. However, the experimentation process is not only a development practice, so it is beneficial to involve other perspectives from people specialized in e.g. business or user experience. The retrospectives should include formulating action points assigned to specific people to implement possible improvements, which is a typical practice in the industry, and in line with how the case company conducts its development retrospectives.

After the case company has in practice conducted an experiment utilizing the experimentation process, it can think about investing more into the process by **involving more advanced practices**. Lindgren and Münch (2016) also recommend an evolutionary approach to experimentation, in which technical infrastructure and ways of working are invested more in after the company has been able to develop the necessary skills and product management tools. In later experiments, building reusable infrastructure can be considered in order to instrument products, i.e. to automatically collect usage data from them. In addition, tools for analyzing the gathered data can be beneficial. The case company could utilize e.g. Google Analytics, which they already have a little experience with, if they wish to spend the time learning a new tool minimal, or services like Amplitude or MixPanel to collect and analyze usage data.

If the case company grows or wants to involve more of their people in the experimentation, they can also consider dividing the roles in the experimentation process to different people, which would allow people to specialize in one of the roles. Fagerholm

et al. (2014) include in their model of systematic experimentation the roles of business analyst and product owner, who create and iterate the product roadmap, data analyst, who designs, executes, and analyses experiments, and developers and quality assurance who are responsible for the actual implementation of the minimum viable product. For the case company, this number of roles may be too many considering the small size of the company; in their case, multiple roles can likely be handled by the same person.

5.4 Limitations of the study

The study did not include testing the proposed experimentation process in practice. Therefore, the suitability of the proposed process still needs to be validated. This is why the author proposed that the case company improves the process once they have accumulated learning from running experiments using the process in real life. The lack of learnings from practical implementation is a limitation of the study and could also be a subject for further research.

The study utilized action research, which is typically best implemented as an iterative process (Avison et al., 1999). Iterative research enables the research to participate in and observe the change in the organization and the actions of its members. However, this study did not include more iterations than going through the cycle of problem diagnosis, action intervention and reflective learning once. Had the proposed experimentation process been implemented in practice, the study would have greatly benefited from going through the action research cycle more than once. In addition, the reflective learning phase of this study only focused on iterating on the proposed experimentation process. The suggestions for the next steps that the case company should take were not iterated on in this study. To make the suggestions better fitting for the case company, they could also have been assessed together with the case company and modified accordingly.

The benefits that experimentation could bring to the case company have also not been validated in practice. Therefore, they remain hypothetical until the company actually implements the experimentation process. The benefits and challenges of implementing this particular process pose an interesting subject for further research.

The study focuses on one case company and the experimentation process is tailored to suit the current processes at the company in question. Therefore, the proposed experimentation process may not be directly usable for other companies.

6 Conclusions

This thesis was motivated by an interest to improve the new product development process of the case company through experimentation. Experimentation enforces data-driven decision making, which could make the product development process of the case company more reliable. The aim of the case company is to be able to steer their business to sustainable growth by verifying with real customers which of their ideas are worth pursuing. As experimentation is especially helpful in managing risks in uncertain environments, it is most interesting for the case company to implement when developing new products instead of supporting their stable existing main product. Therefore, this thesis especially focused on the new product development process in the case company. The findings of the thesis are specific to the situation and environment of the case company; the case company operates in a business-to-government environment, where customer acquisition costs are high, as are switching costs for customers. This is a different environment than where typical examples of companies implementing experimentation operate. The classic examples of experimentation in The Lean Startup generally address companies that can easily acquire new customers and analyze their behavior through methods such as cohort analysis. This is, however, not possible for the case company, since they do not continuously acquire new customers at a high pace. Therefore, the case company needs different strategies for implementing an experimentation process. The research problem this thesis addressed is to find out what those strategies are for this particular case: how the case company can utilize experimentation in their new product development process.

Due to the environment and domain they operate in, **the case company should start experimentation with background research, acquiring a pilot customer and conducting user research instead of jumping straight to developing possible solutions**. This conclusion was arrived at through iterating on the proposed experimentation process and discussions with the case company about how the proposed process should be implemented. There was a clear gap between the current opportunity idea that the case company had and being able to break down the opportunity idea into testable hypotheses. The opportunity idea was not clear and refined enough for the experimentation process to be started. Therefore, background research was necessary in order to clarify what problem the new product would solve. In addition, potential solutions for hypothetical problems were impossible to suggest, because the case company did not have a sufficient understanding of customer processes outside of their established business. This understanding can only be acquired through user research, because the processes can vary from customer to customer and are not documented publicly anywhere in sufficient detail. However, reaching users willing to participate in user research is in some cases challenging for the case company. For most of the products that the case company works on, one of the primary target groups is municipality employees. The case company suspects that in order to find users that are willing to

participate in e.g. user observation studies, they need to work together with a municipality that is their pilot customer. This way, the customers have incentive to participate in the user research, as they will be getting a product built to suit their needs in return for their effort. For these reasons, the case company should start the new product development process with making the opportunity idea clear enough to be explained to potential customers, acquiring a pilot customer with whom experimentation can be conducted, and researching the processes and problems that the customer currently has related to the opportunity idea.

As market demand is especially difficult to validate in environments with high customer acquisition costs, **the case company should start experimenting through validating whether value is created for customers**. Measuring how the customers perceive the worth of the products of the case company and how the worth of the offerings can be increased can produce reliable results for the case company. On the other hand, measuring the market demand for solutions would be unreliable for the case company as the lifetime value of each of their customers is high and correspondingly, customer acquisition is time-consuming. Measuring whether the acquisition rate climbs higher is not viable, since the acquisition rate grows at a very slow pace. If the rate is a little higher in one quarter than the other, a small difference cannot necessarily be attributed to anything other than chance. If the case company measured interest in a solution without measuring whether there are actual buyers for it, the results could be misleading, since making a purchasing decision can be much more complicated for the customers than indicating interest in a product. Therefore, it is best for the case company to start with validating whether they can deliver value to customers, since that will produce more reliable results. Starting with an experimentation topic that is too difficult to reliably experiment on, such as one related to the market demand of a product, would also not help the case company learn how to experiment properly. In addition, the case company might use unreliable experimentation results as a basis for their decision-making if this was the case.

Since introducing experimentation is a significant change to the current process of the case company, **the case company should strive for systematically linking the experimentation results with decision-making through applying a combination of new and existing practices**. Existing practices are used in order to lower the threshold of implementing the process in practice and to ensure that the process fits in with the ways of working of the case company. Experimentation processes are also typically considered to take large amounts of effort by companies that have just started implementing them, which is why lowering the amount of effort needed is crucial. This thesis mapped out the process that the case company currently uses in new product development, along with its challenges and good practices, in order to define which practices to keep and which to avoid. The proposed experimentation process includes documenting the product/project vision along with the related hypotheses and experiments in the documentation tool that the case company already uses. This is however not only done for documentation

purposes; the documented vision for the product/project is utilized later in the process in the analysis meeting, when the experimentation results are used as a basis for decisions for how to proceed with the project or product. In this manner, the product/project vision page is meant to aid in ensuring that the experiment results are systematically linked to decision-making regarding the product or project. The guidelines of the proposed process encourage the team to take a new look at the vision and strategy of the product/project and assess how the experiment results affect it, before making decisions on next steps. Inevitably, most of the steps in the experimentation process will be new to the case company. However, using the documentation of previous meetings as a basis for discussion of how things have evolved is a practice that the case company currently utilizes e.g. in bi-weekly development team retrospectives. In addition, utilizing experiment retrospectives in order to improve the experimentation process is a new practice, adapted from the existing practice of having retrospectives for the development team.

A large part of the scientific literature on experimentation focuses on companies, for which customer acquisition is much easier and faster than for the case company of this thesis. Therefore, one suggestion for further research is how experimentation processes differ according to the different types of customers that companies have. The solution that this thesis proposes for companies like the case company is to start with background research, to acquire a pilot customer and to conduct user research before developing solutions. Further studies are needed to examine the validity of this approach, both for the case company and potentially for other, similar companies. Assessing how the proposed experimentation process works in practice remains a subject for further research, as it has not been implemented by the case company yet but only been developed and improved through discussions.

References

Avison, D., Lau, F., Myers, M., & Nielsen, P. A. (1999). Action Research. *Communications of the ACM*, *42*(1), pp. 94–97.

Bajwa, S. S., Wang, X., Duc, A. N., & Abrahamsson, P. (2016). How Do Software Startups Pivot? Empirical Results from a Multiple Case Study. In: Maglyas A., Lamprecht AL. (eds) *Proceedings of ICSOB 2016,* pp. 169-176.

Bajwa, S. S., Wang, X., Nguyen Duc, A., & Abrahamsson, P. (2017). "Failures" to be celebrated: an analysis of major pivots of software startups. *Empirical Software Engineering*, *22*(5), pp. 2373–2408.

Basili, V. R., Heidrich, J., Lindvall, M., Münch, J., Regardie, M., Rombach, D., Trendowicz, A. (2007). GQM+Strategies: A Comprehensive Methodology for Aligning Business Strategies with Software Measurement. In *Proceedings of the DASMA Software Metric Congress*, pp. 253–266.

Blank, S. (2013a). *The Four Steps to the Epiphany: Successful Strategies for Products That Win*. K S Ranch.

Blank, S. (2013b). Why the Lean Start-Up Changes Everything. *Harvard Business Review*, (May), [online] Available at: <https://hbr.org/2013/05/why-the-lean-start-up-changes-everything> [Accessed 13. January 2019].

Boncheck, M., & Fussell, C. (2013). Decision Making, Top Gun Style. *Harvard Business Review,* (September) [online] Available at: <https://hbr.org/2013/09/decision-making-top-gun-style> [Accessed 13. January 2019].

Bosch J. (2012) Building Products as Innovation Experiment Systems. In: Cusumano M.A., Iyer B., Venkatraman N. (eds) *Proceedings of ICSOB 2012,* pp. 27-39.

Duc A.N., & Abrahamsson P. (2016) Minimum Viable Product or Multiple Facet Product? The Role of MVP in Software Startups. In: Sharp H., Hall T. (eds) *Proceedings of XP 2016,* pp. 118-130.

Fagerholm, F., Guinea, A. S., Mäenpää, H., & Münch, J. (2014). Building blocks for continuous experimentation. In *Proceedings of the 1st International Workshop on Rapid Continuous Software Engineering - RCoSE 2014*, pp. 26–35.

Frederiksen, D. L., & Brem, A. (2017). How do entrepreneurs think they create value? A scientific reflection of Eric Ries' Lean Startup approach. *International Entrepreneurship and Management Journal*, *13*(1), 169–189. https://doi.org/10.1007/s11365-016-0411-x

Grossman, E. (2016). Weighing in: Reflections on a steady diet of Lean Startup. In: M. H. Morris & E. Liguori, Eds. *Annals of entrepreneurship education and pedagogy - 2016*. Edward Elgar Publishing. Ch. 5.

Gruber, M. (2010). Getting to Plan B: Breaking through to a better business model. *Journal of Commercial Biotechnology*, *16*(2), 93–96.

Gutbrod M., Münch J., Tichy M. (2017) How Do Software Startups Approach Experimentation? Empirical Results from a Qualitative Interview Study. In: Felderer

M., Méndez Fernández D., Turhan B., Kalinowski M., Sarro F., Winkler D. (eds) *Proceedings of PROFES 2017,* pp. 297-304.

Hassi, L., & Tuulenmäki, A. (2012). Experimentation-driven approach to innovation: developing novel offerings through experiments. In *ISPIM Conference Proceedings; Manchester*, pp. 1–13.

Järvinen, P. (2004). *On Research Methods*. Tampere: Opinpajan kirja.

Kaplan, R. S., & Norton, D. P. (1992). The Balanced Scorecard-Measures That Drive Performance. *Harvard Business Review*, (January-February), 71–79. Retrieved from https://umei007-fall10.wikispaces.com/file/view/Kaplan%26Nortonbalanced+scorecard.pdf

Karvonen, T., Lwakatare, L. E., Sauvola, T., Bosch, J., Olsson, H. H., Kuvaja, P., & Oivo, M. (2015). Hitting the Target: Practices for Moving Toward Innovation Experiment Systems. In *Proceedings of ICSOB 2015*, pp. 117–131. https://doi.org/10.1007/978-3-319-19593-3_10

Lenarduzzi, V., & Taibi, D. (2016). MVP Explained: A Systematic Mapping Study on the Definitions of Minimal Viable Product. In *Proceedings - 42nd Euromicro Conference on Software Engineering and Advanced Applications*, pp. 112–119.

Lindgren, E., & Münch, J. (2016). Raising the odds of success: the current state of experimentation in product development. *Information and Software Technology*, *77*, 80–91. https://doi.org/10.1016/J.INFSOF.2016.04.008

Nguyen-Duc A., Wang X., Abrahamsson P. (2017) What Influences the Speed of Prototyping? An Empirical Investigation of Twenty Software Startups. In: Baumeister H., Lichter H., Riebisch M. (eds) *Proceedings of XP 2017,* pp. 20-36. https://doi.org/10.1007/978-3-319-57633-6_2

Olsson, H. H., Bosch, J., & Alahyari, H. (2013). Towards R&D as Innovation Experiment Systems: A Framework for Moving Beyond Agile Software Development. In: *Proceedings of the IASTED International Conference on Software Engineering*, pp. 798–805. https://doi.org/10.2316/P.2013.796-008

Patton, M. Q. (2002). Qualitative Interviewing. In *Qualitative Research & Evaluation Methods* (3rd ed., pp. 339–429). Thousand Oaks: Sage Publications.

Poppendieck, M., & Poppendieck, T. (2007). *Implementing Lean Software Development: From Concept to Cash*. Addison-Wesley.

Ries, E. (2011). *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. New York, New York, USA: Crown Business.

Ries, E. (2017). *The Startup Way: How Entrepreneurial Management Transforms Culture and Drives Growth*. Portfolio Penguin.

Ripsas, S., Schaper, B., & Tröger, S. (2015). A Startup Cockpit for the Proof-of-Concept. In: Faltin G. (eds) *Handbuch Entrepreneurship* (pp. 1–16). Wiesbaden: Springer Fachmedien Wiesbaden. https://doi.org/10.1007/978-3-658-05263-8_21-1

Rompho, N. (2018). Operational performance measures for startups. *Measuring Business*

Excellence, *22*(1), 31–41. https://doi.org/10.1108/MBE-06-2017-0028

Sarasvathy, S. D. (2001). Causation and Effectuation: Toward a Theoretical Shift from Economic Inevitability to Entrepreneurial Contingency. *Academy of Management Review*, *26*(2), 243–263. https://doi.org/10.5465/AMR.2001.4378020

Terho, H., Suonsyrjä, S., Karisalo, A., & Mikkonen, T. (2015). Ways to Cross the Rubicon: Pivoting in Software Startups. In: Abrahamsson P., Corral L., Oivo M., Russo B. (eds) *Proceedings of PROFES 2015*, pp. 555-568. https://doi.org/10.1007/978-3-319-26844-6_41

Terho, H., Suonsyrjä, S., & Systä, K. (2016). The Developers Dilemma: Perfect Product Development or Fast Business Validation? In In: Abrahamsson P., Jedlitschka A., Nguyen Duc A., Felderer M., Amasaki S., Mikkonen T. (eds) *Proceedings of PROFES 2016,* pp. 571–579.

Thomke, S. H. (1998). Managing experimentation in the design of new products. *Management Science*, *44*(6), 743–762.

Yaman, S. G., Fagerholm, F., Munezero, M., Münch, J., Aaltola, M., Palmu, C., & Männistö, T. (2016). Transitioning Towards Continuous Experimentation in a Large Software Product and Service Development Organisation – A Case Study. In: Abrahamsson P., Jedlitschka A., Nguyen Duc A., Felderer M., Amasaki S., Mikkonen T. (eds) *Proceedings of PROFES 2016,* pp. 344-359.

Appendices

A Interview structure for employees of the case company

1. What is your position in the company?
2. How long have you been in this job?
3. What was your role in the project?
4. What were the roles of the rest of the people involved in the project?
5. How should the roles in the project have been organized or divided to optimize the process?
6. What were the targeted business benefits of the project?
7. How was the scope of the project determined?
8. How was it determined what is the problem that the customer needs solved?
9. How was it determined what is the problem that the end user needs solved?
10. How was the work prioritized?
11. Looking back, how successful was the prioritization?
12. What information would be needed when prioritizing?
13. Describe how you stayed up-to-date on the status of the parts of the project that were not your responsibility.
14. How did the customers and end users perceive the outcome?
15. Describe how customer interaction happened in the project.
16. How was customer feedback collected?
17. How was end user feedback collected?
18. How was the customer and end user feedback used?
19. Were there any challenges in the process or project that did not come up in previous questions that you would like to bring up?

B Interview structure for the customers of the case company

1. What was your role in the project?
2. When did you start working on this project?

3. What were the roles of the rest of the people involved in the project?
4. How should the roles and responsibilities in the project have been organized or divided to optimize the process?
5. What were the targeted benefits of the project?
6. How was the scope of the project determined?
7. How was it determined what is the problem that the you needed solved?
8. How was it determined what is the problem that the end user needs solved?
9. How was the work prioritized?
10. Looking back, how successful was the prioritization?
11. What information would be needed when prioritizing?
12. Describe, how you stayed up to date on how the work of others was progressing in the project.
13. Describe how interaction between you and the case company happened in the project.
14. How was end user feedback collected?
15. How was end user feedback utilized?
16. How well did the case company utilize your feedback?
17. How did the end users perceive the outcome?
18. How well did the outcome of the project answer your expectations?
19. Were there any challenges in the process that did not come up in previous questions that you would like to bring up?

C Survey for social and healthcare leaders in municipalities

Sosiaali- ja terveydenhuollon palvelusetelimallit maksuissa julkisten ja yksityisten toimijoiden välillä
Tämä kysely on osa Aalto-yliopistoon tehtävää diplomityötä sosiaali- ja terveysalalla toimivien IT-palveluiden tuotekehitysprosessin parantamisesta. Kyselyn tavoitteena on selvittää kuntien ja kuntayhtymien kiinnostusta eri palvelusetelimalleihin, joilla voidaan toteuttaa sosiaali- ja terveyspalveluiden julkiselta puolelta yksityisille toimijoille suuntautuvia maksuja.

Kyselyyn vastaaminen kestää noin 10 minuuttia. Kysely on jaettu kolmeen osaan: taustatietoihin kunnastasi, asiakasmaksumalliin liittyviin kysymyksiin, ja kysymyksiin palvelusetelimalleista sote-uudistuksen kariutuessa. Voit halutessasi jättää osaan kysymyksistä vastaamatta ja siirtyä eteenpäin painamalla enter-nappia.

1. Taustatiedot vastaajan kunnasta tai kuntayhtymästä
    a. Kuntani tai kuntayhtymäni:
    b. Nimikkeeni / tittelini:
    c. Järjestääkö kuntasi / kuntayhtymäsi tällä hetkellä sosiaali- ja terveyspalveluita palvelusetelimallilla? Palvelusetelimallissa kunta myöntää kuntalaiselle maksusitoumuksen, palvelusetelin, jolla tämä voi hankkia tarvitsemansa palvelun kunnan hyväksymältä yksityiseltä palveluntuottajalta, kuten kotihoitoyritykseltä tai yksityiseltä lääkäriasemalta.
        - Kyllä
        - Ei
        - En osaa sanoa

        Jos vastaus on kyllä, siirrytään kysymykseen 2
        Kaikissa muissa tapauksissa siirrytään kysymykseen 3

2. Mikäli sote-uudistus ei toteudu
    Vastaa tämän osion kysymyksiin kuvitellen, että on varmaa ettei sote-uudistus tule toteutumaan. Jos et osaa tai halua vastata kysymyksiin, voit jättää vastaamatta ja edetä enter-napilla tai vierittämällä hiirellä alaspäin.
    a. Mikäli sote-uudistus ei toteudu, kuinka houkuttelevana kuntasi/kuntayhtymäsi pitää siirtymistä sote-palveluiden järjestämiseen osittain tai kokonaan **kapitaatiomallilla**? Kapitaatiomallissa kunnan asukas voi valita itselleen esimerkiksi terveydenhuollon tai suun terveydenhuollon palveluntuottajan, joka ottaa vastuun asiakkaan perusterveydenhuollosta tai suun terveydenhuollosta. Mallissa yksityiselle palveluntuottajalle maksetaan jokaisesta julkiselta siirtyneestä asiakkasta kiinteä korvaus tiettynä ajanjaksona (esim. kuukausittain).
        Likert-asteikko 1-5, missä 1 = Ei yhtään houkuttelevana, 2 = Erittäin houkuttelevana
        Jos vastaus on 4 tai 5, siirrytään kysymykseen 2b, kaikissa muissa tapauksissa siirrytään kysymykseen 2c
    b. Missä palveluissa ensisijaisesti soveltaisit kapitaatiomallia?
    c. Mikäli sote-uudistus ei toteudu, kuinka houkuttelevana kuntasi/kuntayhtymäsi pitää siirtymistä osittain tai kokonaan malliin, jossa **palvelusetelin arvo** määräytyy osittain **kannustin-sanktio -mallin** mukaisesti? Tässä mallissa palvelusetelin lopullinen arvo lasketaan palveluntuottajien tuottamien palveluiden laadun perusteella: heikko palvelun laatu vaikuttaa arvoon negatiivisesti, hyvä laatu taas positiivisesti.
        Likert-asteikko 1-5, missä 1 = Ei yhtään houkuttelevana, 2 = Erittäin houkuttelevana
        Jos vastaus on 4 tai 5, siirrytään kysymykseen 2d, kaikissa muissa tapauksissa siirrytään kysymykseen 2e
    d. Missä palveluissa ensisijaisesti soveltaisit palveluseteliä, jonka arvo määräytyy kannustin-sanktio -mallin mukaisesti?
        Avoin tekstikenttä

e. Mikäli sote-uudistus ei toteudu, kuinka houkuttelevana kuntasi/kuntayhtymäsi pitää siirtymistä sote-palveluiden järjestämiseen osittain tai kokonaan **henkilökohtaisen budjetin mallilla**? Henkilökohtaisen budjetin mallissa asiakkaalle laaditaan suunnitelma asiakkaan tarvitsemista palveluista ja esim. kuukausittainen tai vuosittainen budjetti, jonka hän saa käyttää osittain tai kokonaan vapaasti suunnitelman mukaisiin palveluihin. Tarkemman kuvauksen löydät osoitteesta *https://goo.gl/pXBPEX.*
   Likert-asteikko 1-5, missä 1 = Ei yhtään houkuttelevana, 2 = Erittäin houkuttelevana
   Jos vastaus on 4 tai 5, siirrytään kysymykseen 2f, kaikissa muissa tapauksissa siirrytään kysymykseen 4
f. Missä palveluissa ensisijaisesti soveltaisit henkilökohtaisen budjetin mallia?
   Avoin tekstikenttä

   Siirrytään kysymykseen 4

3. Mikäli sote-uudistus ei toteudu
   Vastaa tämän osion kysymyksiin kuvitellen, että on varmaa ettei sote-uudistus tule toteutumaan. Jos et osaa tai halua vastata kysymyksiin, voit jättää vastaamatta ja edetä enter-napilla tai vierittämällä hiirellä alaspäin.
   a. Mikäli sote-uudistus ei toteudu, kuinka houkuttelevana kuntasi/kuntayhtymäsi pitää sote-palveluiden järjestämisen kokeilemista tai pilotoimista **kapitaatiomallilla**? Kapitaatiomallissa kunnan asukas voi valita itselleen esimerkiksi terveydenhuollon tai suun terveydenhuollon palveluntuottajan, joka ottaa vastuun asiakkaan perusterveydenhuollosta tai suun terveydenhuollosta. Mallissa yksityiselle palveluntuottajalle maksetaan jokaisesta julkiselta siirtyneestä asiakkasta kiinteä korvaus tiettynä ajanjaksona (esim. kuukausittain).
      Likert-asteikko 1-5, missä 1 = Ei yhtään houkuttelevana, 2 = Erittäin houkuttelevana
      Jos vastaus on 4 tai 5, siirrytään kysymykseen 2b, kaikissa muissa tapauksissa siirrytään kysymykseen 2c
   b. Missä palveluissa ensisijaisesti soveltaisit kapitaatiomallia?
   c. Mikäli sote-uudistus ei toteudu, kuinka houkuttelevana kuntasi/kuntayhtymäsi pitää sellaisen palvelusetelimallin kokeilemista tai pilotoimista, missä **palvelusetelin arvo** määräytyy osittain **kannustin-sanktio -mallin** mukaisesti? Tässä mallissa palvelusetelin lopullinen arvo lasketaan palveluntuottajien tuottamien palveluiden laadun perusteella: heikko palvelun laatu vaikuttaa arvoon negatiivisesti, hyvä laatu taas positiivisesti.
      Likert-asteikko 1-5, missä 1 = Ei yhtään houkuttelevana, 2 = Erittäin houkuttelevana
      Jos vastaus on 4 tai 5, siirrytään kysymykseen 2d, kaikissa muissa tapauksissa siirrytään kysymykseen 2e

d. Missä palveluissa ensisijaisesti soveltaisit palveluseteliä, jonka arvo määräytyy kannustin-sanktio -mallin mukaisesti?
   Avoin tekstikenttä
e. Mikäli sote-uudistus ei toteudu, kuinka houkuttelevana kuntasi/kuntayhtymäsi pitää sote-palveluiden järjestämisen kokeilemista tai pilotoimista **henkilökohtaisen budjetin mallilla**? Henkilökohtaisen budjetin mallissa asiakkaalle laaditaan suunnitelma asiakkaan tarvitsemista palveluista ja esim. kuukausittainen tai vuosittainen budjetti, jonka hän saa käyttää osittain tai kokonaan vapaasti suunnitelman mukaisiin palveluihin. Tarkemman kuvauksen löydät osoitteesta *https://goo.gl/pXBPEX.*
   Likert-asteikko 1-5, missä 1 = Ei yhtään houkuttelevana, 2 = Erittäin houkuttelevana
   Jos vastaus on 4 tai 5, siirrytään kysymykseen 2f, kaikissa muissa tapauksissa siirrytään kysymykseen 4
f. Missä palveluissa ensisijaisesti soveltaisit henkilökohtaisen budjetin mallia?
   Avoin tekstikenttä

4. Asiakasmaksut
   Asiakasmaksuilla tarkoitetaan asiakkaan itse maksamia maksuja julkisten sosiaali- ja terveyspalveluiden käytöstä. Lisätietoa asiakasmaksuista löytyy sosiaali- ja terveysministeriön sivuilta: *https://stm.fi/sotehuollon-asiakasmaksut*
   a. Kuinka vaikeaa tai työlästä asiakasmaksujen laskuttaminen on tällä hetkellä kunnassasi tai kuntayhtymässäsi? Mikäli et osaa sanoa, jätä vastaamatta tähän kysymykseen.
      Likert-asteikko 1-5, missä 1 = Ei ollenkaan vaikeaa, 5 = Erittäin vaikeaa
   b. Miten uusi asiakasmaksulaki mielestäsi toteutuessaan vaikuttaa asiakasmaksujen laskuttamiseen? Uudessa asiakasmaksulaissa on tavoitteena mm. määrittää kaikki maksuperusteet lain tasolle ja yhdenmukaistaa maksut tuotantotavasta (julkinen/yksityinen) riippumatta, silloin kun palvelu on tuotettu julkisella rahalla. Lisätietoa uudesta asiakasmaksulaista: *https://stm.fi/asiakasmaksulain-uudistus*
      - Se helpottaa laskuttamista
      - Se vaikeuttaa laskuttamista
      - Se ei vaikuta mitenkään
      - En osaa sanoa
   c. Miten sote-uudistus mielestäsi toteutuessaan vaikuttaa asiakasmaksujen laskuttamiseen? Sote-uudistuksessa on tavoitteena, että kaikki asiakasmaksut perisi maakunta riippumatta siitä, onko palvelun tuottaja julkinen/yksityinen.
      - Se helpottaa laskuttamista
      - Se vaikeuttaa laskuttamista
      - Se ei vaikuta mitenkään
      - En osaa sanoa
   d. Onko asiakasmaksujen laskuttamiseen olemassa tällä hetkellä kuntasi tarpeisiin soveltuvaa tietojärjestelmää?
      - Kyllä
      - Ei

- En osaa sanoa

5. Haluaisitko keskustella Vaanan asiantuntijoiden kanssa eri palvelusetelimallien toteuttamisesta?
   - Kyllä
   - Ei

6. Tässä voit vielä antaa palautetta kyselystä tai halutessasi perustella antamiasi vastauksia. Voit myös jättää vastaamatta kysymykseen.
   Avoin tekstikenttä

Kiitos ajastasi ja vastauksistasi!