HAAGA-HELIA
University of Applied Sciences

A framework for evaluating the viability of business ideas for web and mobile products; Case study 1: Unhidden Case study 2: Opas

Mikael Vankalo

Bachelor's Thesis

Degree Programme in Experience and Wellness Management 2016

HAAGA-HELIA
University of Applied Sciences

Author	Year of entry
Mikael Vankalo	2011
Title of report A framework for evaluating the viability of business ideas for web and mobile products; Case study 1: Unhidden, Case Study 2: Opas	**Number of report and attachment pages** 76 + 8
Advisors Tommo Koivusalo Mario Ascenção	

Abstract

One definition of a Startup is that it is a temporary organization in search of a scalable, repeatable and profitable business model (Blank 2012, 5). More than 90% of technology Startups fail within 3 years (Marmer & al., 2011) with 70% caused by premature scaling, additionally 42% by building something with no market need (CBInsights 2014). This portfolio-based thesis aims to critically examine methodologies of thought leaders in web and mobile based Startup entrepreneurship and attempt to compile a custom-made theoretical framework that could be used to experiment with web / mobile business ideas and improve the resulting Startup's likelihood to succeed. The resulting framework is applied into two separate experiments in web and mobile products starting from only a business idea and resulting to a Minimum Viable Product (MVP) that is tested for market need.

Keywords: Startup, Entrepreneurship, Lean Startup, Agile, Business models, MVP.

Table of Contents

1. Introduction

"The entrepreneurial function does not essentially consist in either inventing anything or otherwise creating the condition which the enterprise exploits. It consists in getting things done" (Schumpeter 1934, 132). The earliest research on entrepreneurship can be traced back to Joseph Schumpeter. According to Schumpeter (1934), the entrepreneur who spreads innovation is the single most important factor in economic development and the key driver to keep the economics in motion.

The entrepreneurs' mindset today especially in the field of technology Startups has supposedly changed by relatively new methodologies such as the lean Startup. While the previous generations' purpose was to build established companies that last, the current generation is accused of hacking together technology Startups that simply build features rather than products with the purpose of a fast exit and being acquired by large, established corporations in technology such as Apple, Google, Microsoft or Facebook. (McGinn 2012).

This is a portfolio thesis, combined from studying several methodologies associated with web and mobile Startups during the last year of my studies, while aiming to build a web/mobile Startup that meets a market need. The following methodologies are studied: The Lean Startup Methodology, Agile Software Development and User Experience Design. The approaches from thought leaders in these fields are researched following an empirical part based on the learning from these approaches by conducting two experiments. The thesis also chooses to use analytics of previous Startup failures, such as The Startup Genome Report. It is currently the most comprehensive research and analytics focused only on Startups and the components that cause success or failure for Startups. The sample size consisted of questionnaires from 663 global high-growth Startups and the analysed data from 3200 Startups. According to their analysis, up to 92% of global high-growth technology Startups fail, with failure being defined as ceasing to exist after 3 years (Marmer, Herrmann, Dogrultan & Berman 2011). In the majority of the cases, failure was more likely to be caused by self-destruction rather than competition or external issues. This indicates how important it is for the founder(s) to focus on getting internal matters on track right from the start of a new business idea.

The Genome Report concludes that the single, largest issue that caused 70% of the Startup failures, is premature scaling. The wide concept of premature scaling simply means that the Startup is trying to expand too fast, dimensions of customers, product, team, business model or funding that could not handle the expansion. (Marmer & al., 2011)

To provide another perspective on what causes Startup failure or success, data is applied also from CBinsights, a venture capital database. Their research included 101 Startups that failed. According to their study, the number one biggest reason for failure is simply building something that does not have a market need, which accounted to 42% of the Startup failures (CBInsights 2014). Additionally, two out of three Startups that survive, report having changed their plans drastically during the way (Mullins & Komisar 2009).

Based on this knowledge, it seems reasonable to suggest that it makes more sense for new Startups to start with a different approach compared to making a traditional business plan, especially so if the Startup is still confused about what the product is, who the customers will be and how it will make money (McClure 2013). More specifically, there is a clear need for a dynamic and updatable model that provides the most use also as a learning tool for someone starting with an only an idea that also supports an evolutionary process as the founder(s) and company grows. The development process should also support fast alterations in the business environment, rather than firmly follow a plan that is based on assumptions and planned 5 years ahead. Equally, this dynamic business model should not foreclose the importance of planning, strategy and vision. The aim is to follow a vision connected to a business idea and persevere long enough until validated learning about the market by performing experiments until elements in the business model are proven right or wrong. This activity should lead to actual progress by confirming what works or showing the need to change something that is proven wrong. This is also meant to minimize more expensive risks for failure in the future and knowing how to things correctly before scaling up. This thesis project aims to focus on recognizing those risks early on the Startup journey, so they can be experimented with and prepared for while the Startup is still in search mode of its business model.

Eric Ries, a Silicon Valley entrepreneur and author compiled together a toolset for Startup entrepreneurs in his book 'The Lean Startup' in 2011. It has its roots derived from the lean manufacturing system at Toyota between 1948 and 1975 (Ries 2011, 6). He is accredited for being a pioneer in the lean Startup movement, which is a strategy for optimal allocation of resources to reduce waste in the process. Instead of sumptuous planning and "build it and they will come" mentality, it claims to be based on scientific experimentation, analytics and actions based on validated learning. It has gained momentum and hype among consultants and in the Startup world since 2011, coining terms like *pivoting*, *Minimum Viable Product* and the *build-measure-learn* cycle. The main idea behind it is, that the entrepreneur turns his vision into hypotheses for a business model. These hypotheses then needs to be validated through gathering feedback from building a minimum set of features for the product or service. Measuring the feedback from the Minimum Viable Product (MVP), learning takes place and elements in the business model are either; persevered, pivoted or dumped (Blank 2012, 67)

1.1. Objectives and scope

1. The primary objective of this portfolio based thesis is to compile a suitable theoretical structure that combines business development with software development and adding design aspects for the early stages in web and mobile based entrepreneurship.

2. The secondary objective is to use the theory to build a Minimum Viable Product (MVP), using the resources available to test if it is something that the customer wants in order to decrease risks of later Startup failure caused by no market need.

The scope consists of theory compiled from studying relevant publications of thought leaders related to Startups. To back up the theory, the study aims to find trustworthy analytics and statistics related to Startups. The chosen theory and analytics are combined to create a theoretical framework that is adapted and modified to reflect my own position as the researcher and future Startup founder, giving direction for the study and to be used as a foundation to test future business ideas for market need.

The following methodologies are studied more closely as the basis of a theoretical foundation: Lean Startup, Agile Software Development and User Experience Design. The two experiments that are conducted contain tools and methods that are adapted from these frameworks as seen fit and within the scope of the project.

To create an overview of the business idea and to describe the different elements in the business model the thesis compares the Business Model Canvas based on the thesis work by Alexander Osterwalder (2010) with the Lean Canvas, created by Ash Maurya (2012) and published in his book 'Running Lean', which has gained traction particularly in early stage internet Startups that are still in the idea validation stage.

Entrepreneurship in itself, is claimed to be a collection of a wide range of sciences such as decision making, economics, management, sociology and psychology (Amit, Glosten, & Muller 1993, 25). The thesis project will not attempt to categorize entrepreneurship as any science or art, or a mixture of both, but simply to find out if the two experiments would reveal some parts of science in it, i.e. something that could be repeated later with known variables. The aim is to include the theoretical frameworks and find out through experimenting and learning from many failures, if a more optimal path towards success in building web / mobile based products could be reached. The statistics about likely causes of failure from the Startup Genome Project and CBinsights is used as a group of underwater rocks to be spotted early on and proactively steer the direction to avoid crashing into those common reasons of failure when the Startup is still in its idea stage.

The practical implementations start by coming up with a business idea that is aimed to be a scalable solution for a real problem and is validated on a segment of people that would be the first users or paying customers (Early adopters). The familiarity with these three elements: problem, solution, users/customers are used as the starting criteria when selecting a business idea. The idea is incrementally evolved by learning from experimenting and increasingly taking into account additional elements that a Startup needs to succeed. For the software development part, this thesis examines two common approaches: The Waterfall Model versus Agile Development, aiming to find an applied development methodology that is suitable for the current project.

The primary criteria for choosing the right methodologies lies predominantly in simplicity, which in turn allows speed in development and eases the understanding of entire the entire project (Highsmith 2009, 40).The project, as well as the experiments' scope are constrained by the resources available: close to zero budget, resources for expertise are limited to myself as a novice web developer and the resources and skills of any team members that can be assembled for the projects with a deadline for a Minimum Viable Product on 16th of December 2015. This project intends to use mostly free, open source software and find options of starting a business without funding (Bootstrapping) by ideal and inventive use of resources available. This documentation of the process is aimed to compile a structural body and hopefully provide a helpful guide for starting entrepreneurs with a Startup idea for building web or mobile products.

1.2. Research problem(s)

The foremost problem this research is trying to solve is: where should the focus lie in the early stages of building a web Startup to avoid a costlier failure later on? The current situation does provide a wealth of research, information and examples of the great Startup success stories and simply trying to copy those models might give a sense of clearness and security compared to an almost endless range of possibilities. Studying also Startup failures, offers an alternative way to learn, especially for an unexperienced entrepreneur by showing to not do the same mistakes that someone has already done. It also helps the Startup founder(s) to build a learning culture early on that is based on finding opportunities for experimenting in what works. (Edmondson 2011).

This research aims to examine that approach of reaching success from the perspective of experiments and learning from executing tiny failures and incrementally building a Startup on top on those learnings rather than attempting to replicate success stories. This approach brings forth challenges in form of research quality about failed Startups, especially in the local Finnish Startup ecosystem. For many logical reasons, Startups tend to keep low profile before they reach success and the early stages of failed Startups are very often only in the mind of the people involved and left unpublished. There is also clearly a lack of reliable statistics, analytics and documented history about local Startups in Finland. National statistic providers are also not collecting statistics for a specific group of Startups such as web/mobile Startups per se, or even Startups at all, but rather focusing on all new enterprises (Statistics Finland 2015).

As a result, this thesis chooses to rely on the analytics picked from the Startup Genome report and CBinsights, which are both operating from Silicon Valley, but have analysed Startups globally in their research. This is seen fit for the study, as in the context of this thesis a web / mobile Startup is defined to be scalable for the global market. The challenges about conducting research about Startups also repeatedly involve issues of the terms not being properly defined. A fundamental example of this is that an accurate, globally accepted definition of a Startup is still absent that would allow more reliable comparison of research from different sources. A local example of this issue is how the Finnish media ended up in an unresolved debate in 2014, whether or not the tax-payer supported mining company that ran into environmental problems named Talvivaara should be called a Startup (Rämö 2014).

The secondary goal is to achieve a project management process where flexibility and stability are in balance. Related to defining success, a very common issue is found in most publications about managing in the way they measure project success. Traditionally, project success is measured how well the project output matches the plan for scope, cost and schedule to determine whether it is a success or not. This if found highly inconsistent with the lean Startup mind-set as well as agile projects, where value for the customer and quality are of the highest priorities. For that reason, the experiments conducted aim to primarily measure value and quality to its users above the projects scope, cost and schedule. This makes measuring more difficult and less accurate, but is eventually considered better to measure somewhat fuzzy information that is important about actual project success than precise measures of unimportant matters that does not indicate much about the projects potential to become a successful Startup.

This thesis aims to also examine two viable business canvas models for a lean Startup (Business Model Canvas and Lean Canvas) and choosing one to start working with. Due to the vast possibilities, the research will only discuss solutions that are viable with the very limited resources available. Since this thesis is also highly interdisciplinary; it is compiling a multitude of study fields to achieve somewhat fuzzy goals, the research has a potential to overgeneralize, stay on an abstract level and not go too much into detail. Consequently, the focus is purposefully on finding simplicity in processes to get things done in an entrepreneurial spirit rather than forming extensive and overly detailed documentation.

1.3. Concepts and definitions

MVP	Minimum Viable Product. A version of a new product which allows a team to collect the maximum amount of validated learning about customers with the least effort (Ries 2009)
Pivot	In relation to entrepreneurship, means a shift in strategy, usually driven by specific customer feedback. (Blank 2012, 25)
Customer development	A structured process for testing the business model hypotheses- markets, customers, channels, pricing- and turning those into facts. (Blank 2012, 22)
Bootstrapping	To finance your company's Startup and growth with the assistance of or input from others. (Small Business Encyclopedia 2016)
Innovation accounting	How to Define, Measure, and Communicate Progress. (Ries 2011, 25)
Startup	A temporary organization designed to search a repeatable and scalable business model. (Blank 2012, xvii).
Business model	Describes how an organization creates, delivers and captures value (Blank 2012, 10)
Web Startup	The Startups' product or service is acquired or consumed via the web or a website (The great Startup wiki 2016)
LSM	Lean Startup Methodology. (Ries 2011, 5)
BMC	Business Model Canvas. A visual canvas developed by Alexander Osterwalder that contains the elements of a business model (Osterwalder 2010)

Agile	An approach to software development and project management where development teams work more regularly with business users in short iterations and teams are empowered and self-organized (Wikipedia 2016)
Scrum	One of the most common agile processes that has proven to be successful where teams of seven to nine people work interactively with business users within short sprints (iteration) to regularly deliver software increments. (Schwaber & Sunderland 2013)
SaaS	Software as a service. A software product that is hosted remotely, usually over the internet (Wikipedia 2016)
UX	User Experience. Requirements and recommendations for human-centric design principles and activities throughout the life cycle of computer-based interactive systems. (Wikipedia 2016)

2. Theoretical Frameworks

2.1. History of business planning and product development in brief

Dutch East India Company is considered the first ever company, founded in 1602. From there it took a while before the first organization chart, "Principles of Management", was written in 1856 by Brian McCallum to manage corporations. The first Business administrators were graduated in 1908 from the Harvard MBA program. These MBA curriculums were the first to provide managers and administrators with a set of tools such as accounting, human resources and strategy that they needed to run existing companies. This set of tools that was taught around the world in other MBA's for a roughly a hundred years, but was missing one key set of tools. This tool set was how to start a business and survive the first years of uncertainty. Below is an illustration of the traditional product development model that relies on that the Startup founders know exactly how the product will solve a problem for the customer and what features it must have right from the very beginning. (Blank & Dorff 2012, 6)

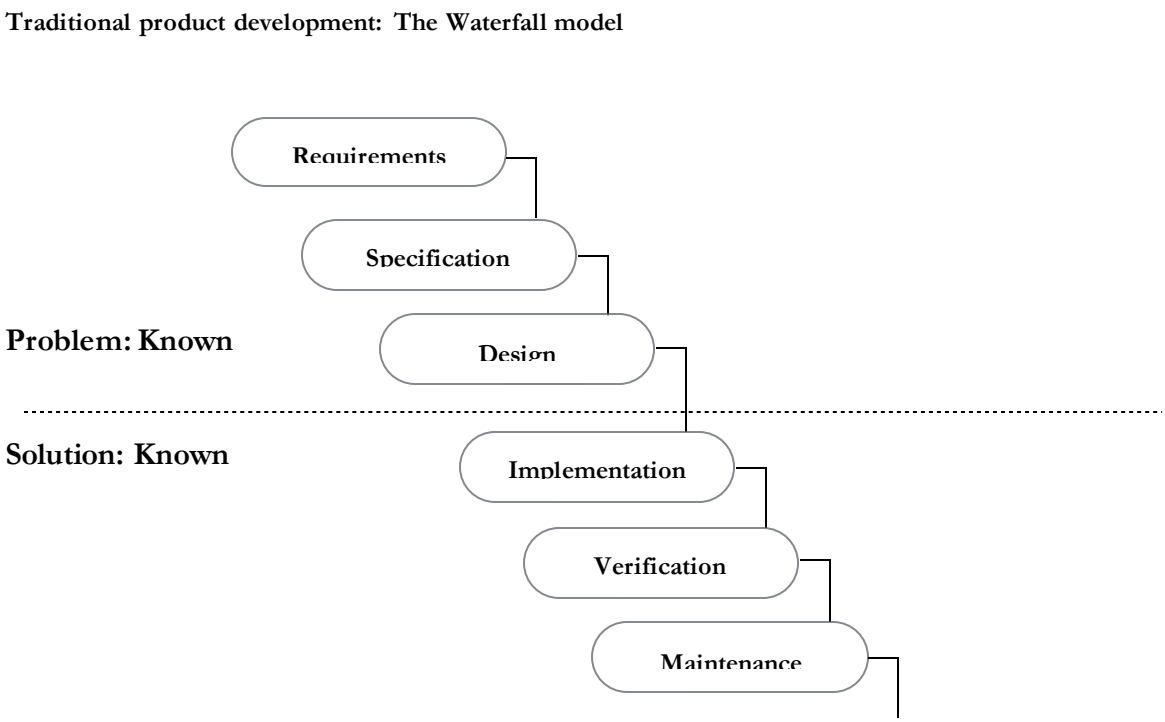


Figure 1. Illustration of Traditional waterfall development (Ries 2009).

One of the greatest flaws with the development model is that you do not really know if you are wrong until you go out of business. This traditional business plan approach is generally represented in software development by the "Waterfall" model that relies on the problem and solution being known before the development process starts (Blank & Dorff 2012, 15).

Eric Ries (2011) points out that Startups are in in fact building experiments that needs to be validated through measuring and learning, rather than building products and usually do not know what the exact problem is when starting with a business idea and how they're going to solve it. Ries, suggests therefore that Startups should instead use an iterative and adaptable process of focusing on focusing early on to what customers wants using a structured process, called Customer Development instead of following traditional product development processes such as the waterfall model. (Ries 2011, 61)

2.2. Origins of the lean Startup approach

In his book 'The lean Startup' Eric Ries describes the foundation for the framework, which has led to form the Lean Startup Methodology (LSM). The ideas originate from the manufacturing industry, more specifically from Japan and the Toyota Production system which Ries began to study. Eric Ries combined the ideas from the Toyota manufacturing process with his own entrepreneurial challenges in software development and the framework for the lean Startup methodology started to conceptualize. Meanwhile, in the beginning of the 21st century many mobile and web Startups in Silicon Valley began to develop their own management tools that suited better for a starting their business, when the existing models were mostly for established companies that had little uncertainty of who their customers were, how to make money and what their product should be. The book 'Four steps to the epiphany' (2006), by Steve Blank was one the first major publications to offer tools specifically for Startups such as Customer Development. The core idea of it is that 'products that are developed by founders who get in front of customers early and often, win'. He defined a Startup to be 'a temporary organization in the search of a scalable, repeatable and profitable business model' (Blank & Dorff 2012, 15).

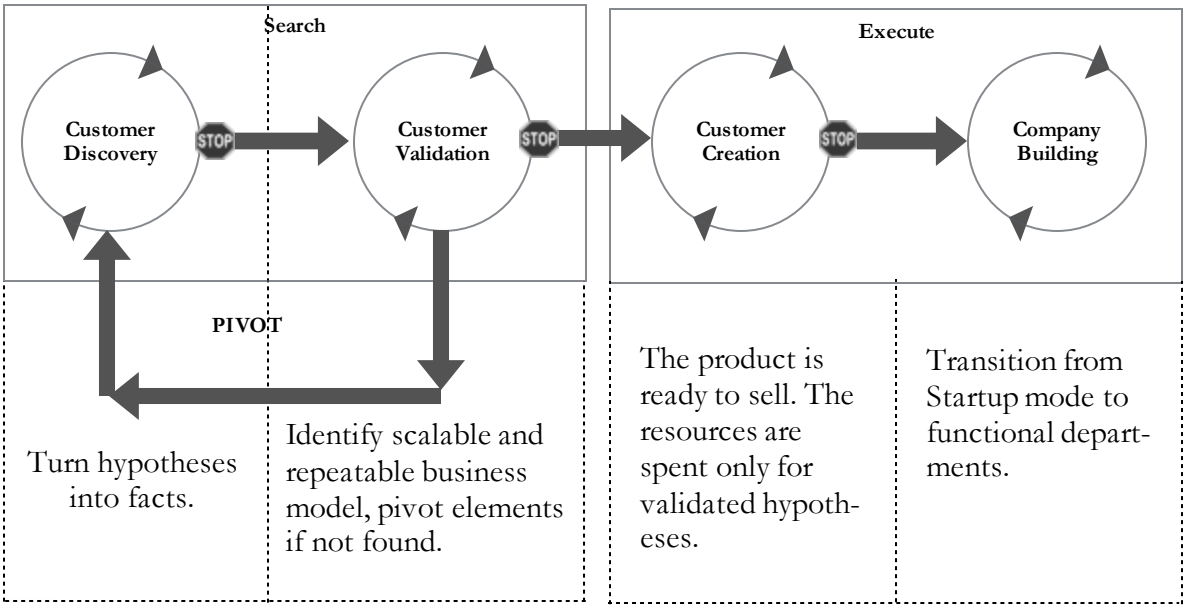


Figure 2. Illustration of the four stages of a Startup (Blank & Dorff 2012, 53; Blank 2013).

This model also sums up the early stages involved for a Startup to reduce the risks of failure later on by testing its assumptions for making a business profitable on real potential customers and only proceed with what is known to be a fact rather than trust on assumptions.

In his book, the Startup Owner's Manual, Steve Blank indicates that the first ever Startup built using the modern lean Startup methodology would be IMVU, a 3D instant message chat, where the user could create customizable avatars and meet new people. IMVU was co-founded by Eric Ries, who was a student of Steve Blank. It used a combination from the learnings of customer development and agile software development principles and iterated its way to a successful company while other similar companies did not succeed. (Blank & Dorff 2012, 16),

Eric Ries combined two methods, agile development in engineering to solve the issue of "unknown features" and customer development for "unknown customer needs" to form a methodology called 'The Lean Startup'. Probably the most unusual thing about the lean Startup approach was a rapid development cycle that consisted of adding newly written code up to 50 times a day into the production version. (Ries 2011, 64)

The lean Startup founds its core idea to a Startup being "a catalyst that transforms ideas into products" (Ries 2011, 75). When customers again come in touch with the product, data is created from feedback that can and should be acted upon. There is also a heavy emphasis on that Startups are actually building experiments instead of products and the valuable part for a Startup is to learn from those experiments by measuring in a cycle consisting of small batches where the aim to minimize the time spent on the total loop of the cycle called the "Build-Measure-Learn Loop". (Ries 2011, 75).

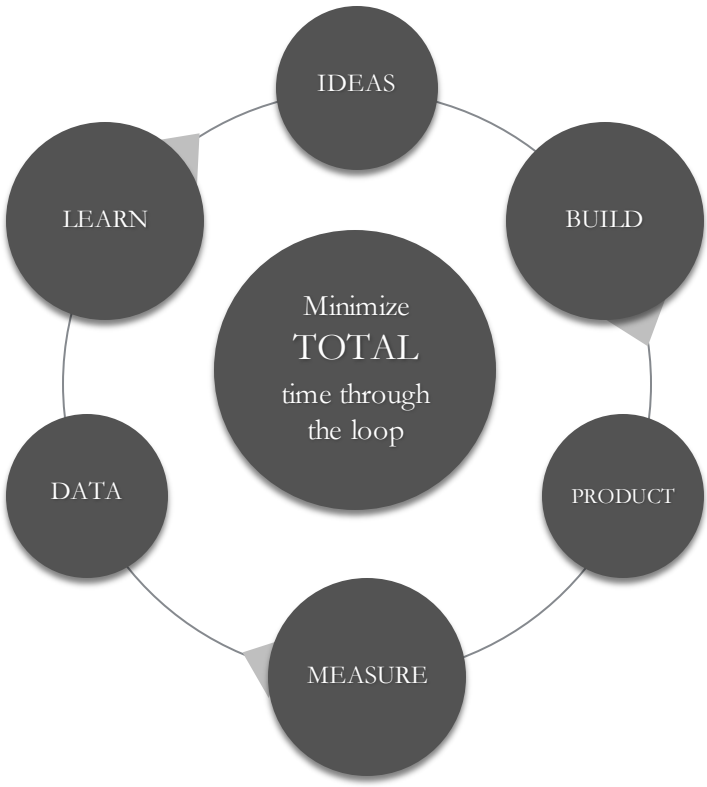


Figure 3. Illustration of the Build-Measure-Learn Loop (Ries 2011, 75).

2.3. The Minimum Viable Product

The Minimum Viable Product is a minimum set of features of a product that can go through the whole Build-Measure-Learn loop with the smallest amount of effort and development time (Ries 2011, 77). Progress is thereafter measured by validated learning that is applied to a full turn around the loop. A Lean Startup is the embodiment of its founder's vision and strategy, which is conveyed through a business model using customer development and agile development methodology. The business model consists initially of assumptions that need to be validated or changed after a full cycle is completed in the build-measure-loop. A change in the strategy and a change in the business model is one of the fundamental aspects in the lean Startup methodology, called a pivot. (Ries 2011, 78)

2.3.1 Build

The MVP should be considered as the fastest way to gain knowledge if there's a market need (Blank 2012). It should be a risk reduction tool to test if the assumed business model works. However, the MVP should not simply a product that is barely functional. Firstly, if it doesn't offer any value for the customer beyond functionality, it cannot be considered reliable and usable by the customer (Blagojevic 2013).

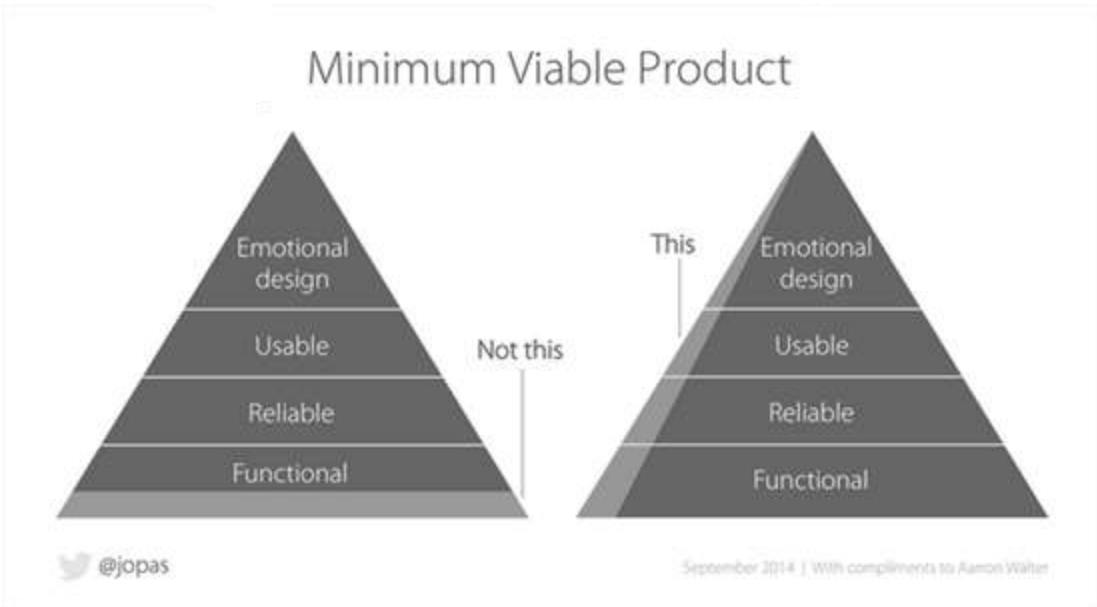


Figure 4. Illustration of what the Minimum Viable Product should be (Pasanen 2 Jan 2015).

Secondly, the most common misconception of the MVP is most likely that it even has to be a tangible product (Bank 2014). According to Ries (2009), it is more about learning as much as possible with the least amount of effort. As examples, some common types of Minimum Viable Products are:

1. Explainer video

A classic example of this would be what Dropbox used to validate the problem that their idea was based on. It was a simple 3-minute demo of how the product worked, combined with smart share ability and marketing to gain visibility. With the video, they tested their riskiest assumption by showing how superior file synchronization worked on a video, people realized this was a problem for them. It proved that the assumption was right and that there was a demand by getting over 70 000 email sign ups overnight through its landing page and thus confirming the founders to start building a Startup based on the MVP. (Bank 2014).

2. Landing page

The most common type consisting of a rather simple website called a landing page that contains by minimum the information about the offering (value proposition) and calls the visitor to perform an action e.g. sign up or place an order. The activity is usually measured with analytics tools with the most common metric being to measure the conversion rate, i.e. what percentage of visitors sign up by giving their email address. (Bank 2014)

3. Concierge MVP

A manual process that contains all the same steps as the product would, except that the steps are performed manually rather than automated. (Bank 2014).

4. Wizard of Oz MVP

As an example, the founder of Zappos started this way by taking photos of shoes in traditional shoe stores and placed the pictures online. When people wanted to order the shoes online, the founder bought them from the shoe store and shipped the orders. This validated that there was a need for an online shopping experience for shoes with little investment. (Bank 2014).

2.3.2 Measure

Reliable measuring of feedback creates initially a lot of extra work compared to traditional product development, but should become easier over time as the process integrates as a natural stage in development. Ries (2011, 114), has coined the term *Innovation accounting,* which focuses on how progress could be defined, measured or communicated.

The following techniques are common in Lean Startups to measure progress:
Split testing or A/B testing – Comparison of two versions by subject response (Wikipedia 2016)
Cohort analysis – Data categorized into related groups for analysis (Wikipedia 2016)
Conversion funnels – Describing a customer journey on a site converting into revenue (Wikipedia 2016)
Net promoter score (NPS) – E.g. "How likely is it that you would recommend our company/product/service to a friend or colleague?" (Wikipedia 2016)
Usability testing – observation under controlled conditions how well the users can use the product (Wikipedia 2016)

Dave McClure (2007), coined the term "pirate metrics", a conversion funnel used by many Startups as a general layout and provides a good guideline to start with before better understanding better what to specifically measure.

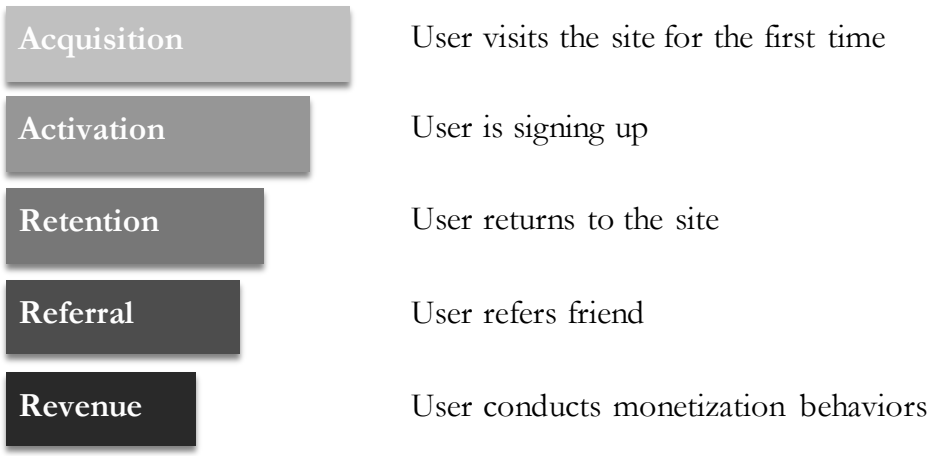


Figure 5. Illustration of Pirate metrics (McClure 2007).

2.3.3 Learn

The final and undoubtedly most important thing is to learn by interacting with the customers which parts consist of waste and which parts are value of the MVP in order to get further ideas how to improve it. According to Ries (2011, 47), many products or features cannot be tested without being built first, but a good reminder of the Build-Measure-Learn mindset is that anything that isn't tested with the customer remains a potential form of waste and it is better to find out sooner than later. When the build-measure-loop is finished the entrepreneur faces the option of either pivot for a shift in strategy and or persevere with the original strategy based on the learning to continue with the next iteration. (Ries 2011, 47).

2.4. Customer development vs Design thinking

Another widely used process for coming up with solutions for problems is Design Thinking, used mainly to create general innovations instead of high-tech innovations for Startups. The process is also an iterative one like the lean Startup process, differencing mostly in the urge for speed to create a new product that matches the customer needs (Blank 2014). According to Blank (2014), the first major difference is the starting point, design thinking relies on finding out the customer need through extensive market research, whereas customer development has its starting point in the founders' vision of the product and creating a customer. Design thinking uses more structured and widespread methods and tools in multiple phases, which can be time consuming and expensive for a Startup with limited resources. Consequently, Blank (2014) suggests that the best bet for a small tech Startup with limited resources is to build something that is 'good enough' (a Minimum Viable Product) and continuously improve it from there on using the lean Startup methodology. (Blank 2014)

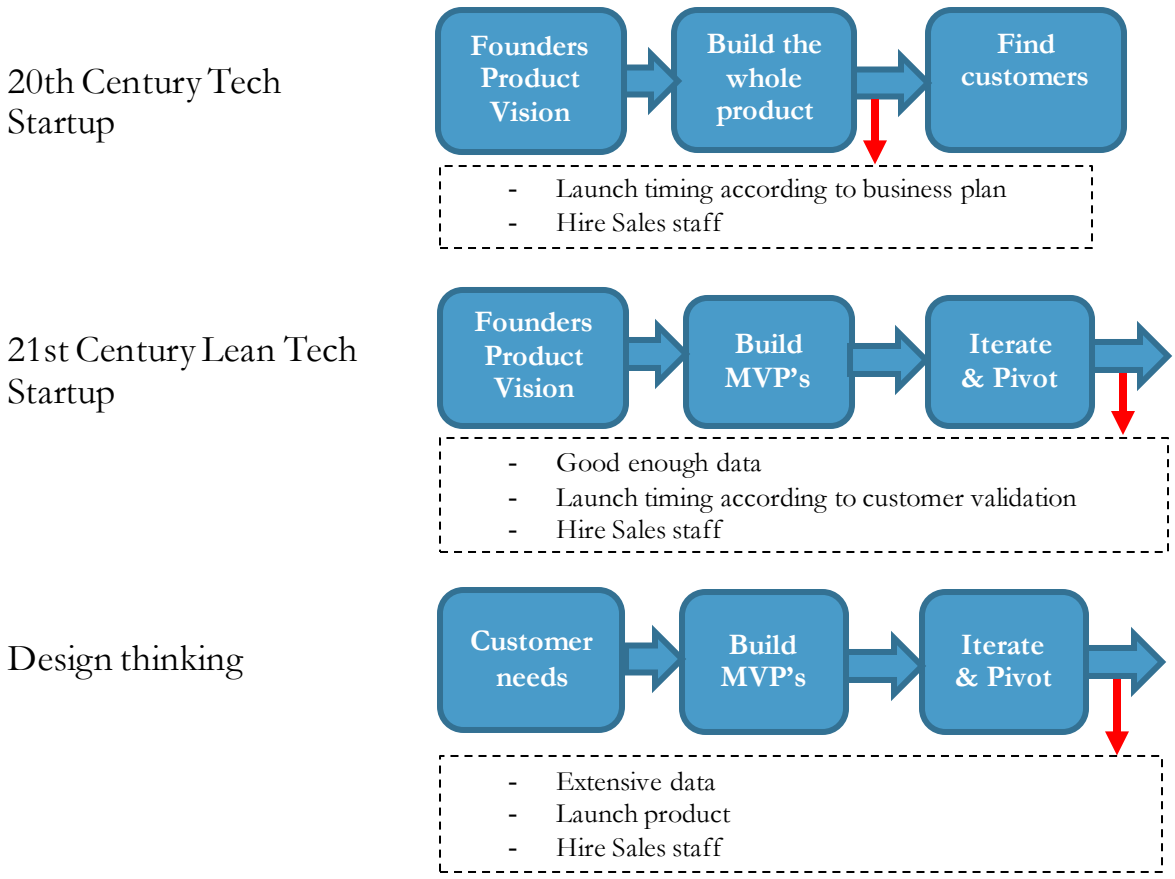


Figure 6. Illustration of the comparison between the three frameworks (Blank 2014).

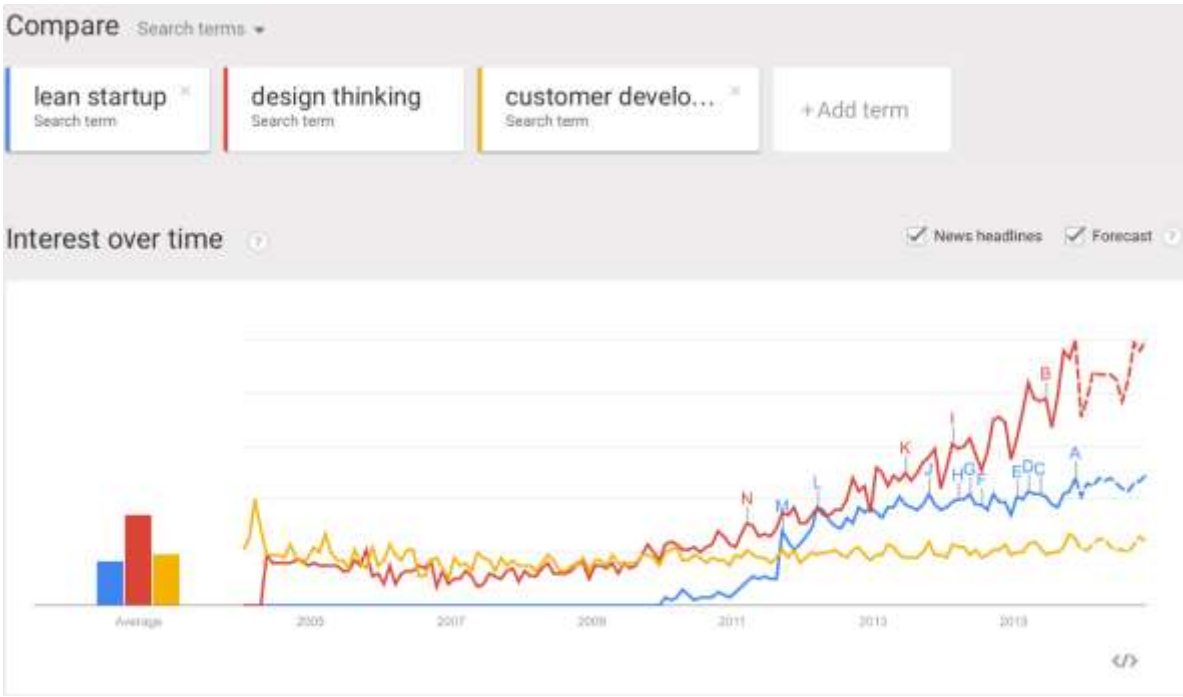


Figure 7. Illustration of the popularity of the three frameworks by search terms at Google Trends (Developed by the author)

2.5. Business model canvases suitable for the lean Startup approach

Currently, there are two popular business model canvases associated with the lean Startup approach. The first is the Business Model Canvas (BMC) proposed by Alexander Osterwalder (2010). The second one is the Lean Canvas (LC), by Ash Maurya (2012), which is an adoption from the Business Model Canvas. Osterwalder's publication, Business Model Generation (2010) is a comprehensive guide to business modelling on a single canvas based on the design thinking approach but has a lot in common with the lean methodology. The main difference is the amount of time and resources required to validate the model using real customers, as described by Steve Blank in the previous chapter. The Business Model Canvas has been largely adopted by large corporations such as IBM and Ericsson in their new ventures and internal Startups, where the resources are more abundantly available. (Osterwalder 2010)

The process briefly behind completing a BMC starts with the founder's vision translated into hypotheses about each component of the business model and requires a set of experiments to test them. Questions such as who are your customers, what problems they need to have solved, what features would solve them and how much customers would pay to solve them. Only after these hypotheses are carefully tested to be true, by being in close contact with your customers, you have a valid idea that can be started up to build on. (Osterwalder 2010, 15, 28)

2.5.1 The business model canvas

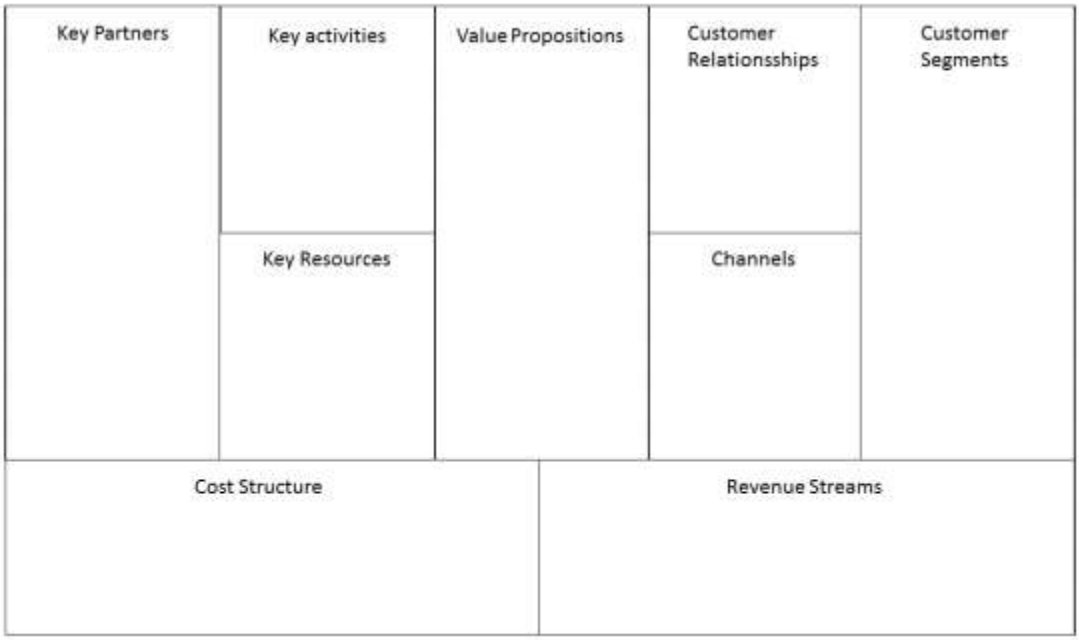


Figure 8. Illustration of the Business Model Canvas (BMC) (Osterwalder 2010).

The Business Model Canvas (BMC) was developed for a need of a model that everyone would understand. Osterwalder (2010, 15) suggests there was a need for a model that is "simple, relevant, and intuitively understandable, while not oversimplifying the complexities how enterprises function".

There is, however a general consensus reached of which business model is suitable for a specific target audience on the official website of the Business Model Canvas. The site compares the Business model canvas with the lean canvas. The advice is to use the Business Model Canvas for new and existing businesses, whereas the lean canvas is purely targeted towards Startup entrepreneurs and a simpler problem-solution oriented approach that allows entrepreneurs to learn gradually how to improve as an entrepreneur (Canvanizer 2014).

2.5.2 The lean canvas

The Lean Canvas (LC) is adopted by Ash Maurya from the Business Model canvas by Alexander Osterwalder. The changes made are specifically designed to match the needs of an early stage software technology Startup.

The reason Ash Maurya started to derive his own customized model was that he thought the Business Model Canvas was made using established companies such as Apple and Skype with a vast amount of resources as examples and not focusing enough on the really early stages of a Startup as shown in the illustration that follows. (Maurya 2012, 6).

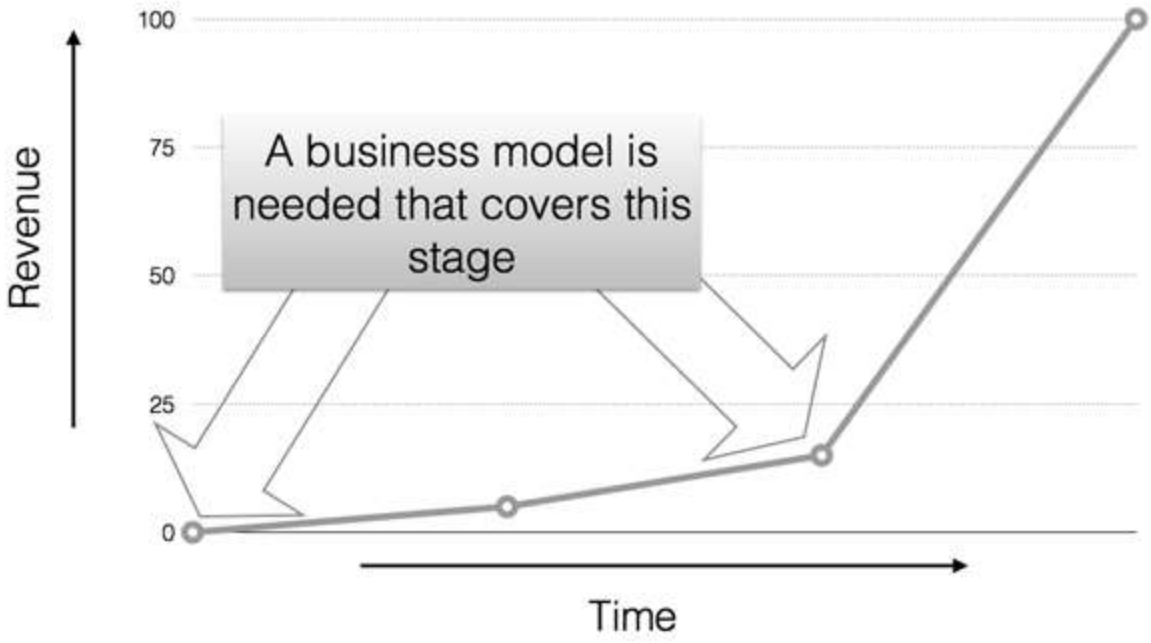


Figure 9. Illustration of the suitability of Lean Canvas (Maurya 2012, 6)

The benefits of the lean canvas include that it is fast to create, concise because you have to select your words carefully and portable so you can iterate it easily (Maurya 2012, 5-6). The lean canvas is meant to be filled by writing down the initial vision for the business and to show it to at least one person. The initial idea behind a constant iteration of a business plan is that it is highly unlikely to get it right the first time (Maurya 2012, 6). Steve Blank suggests a similar strategy (Blank 2012, 34) to continuously "make itera-

tions and pivots" to the business model that are tested through a stream of pass/fail experiments, though he recommends using the Business Model Canvas for writing down the first assumptions for elements involved.

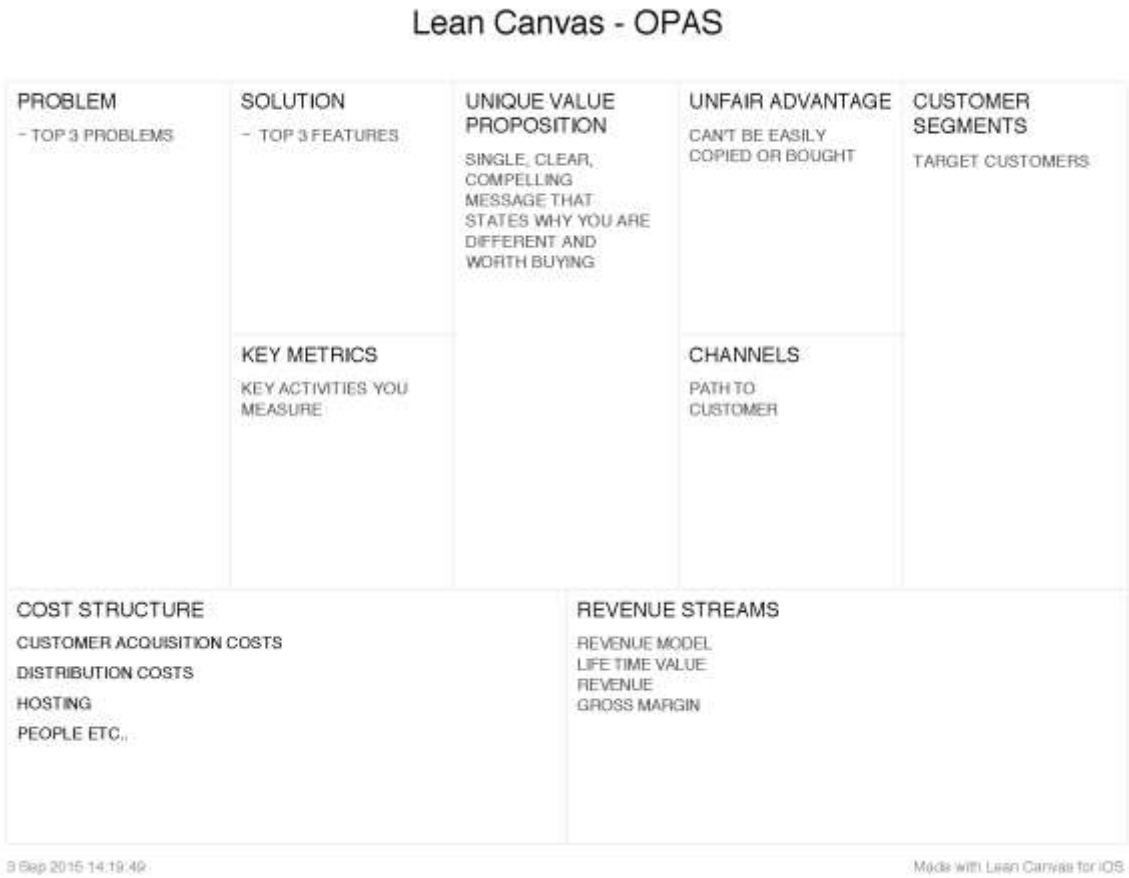


Figure 10. Illustration of the Lean Canvas guide (Maurya 2012, 5)

Maurya (2012, 15) suggests to start with the defining the problem you are trying to solve and then fill out the customer segment that has that problem. After the problem is defined a solution is drafted on the canvas, which is used as a basis for the Minimum Viable Product (MVP).

2.6. Development frameworks

2.6.1. User Experience Design

The broad term "user experience design" (UX) was invented and coined by Don Norman, because he wanted to broaden the aspects covering the user's experience of the product or service beyond traditional human interface and usability design (Bard 2014). According to his view the definition "User experience encompasses all aspects of the end-user's interaction with the company, its services, and its products" (Norman & Nielsen 2015)

User experience design strives to make sure that:
"the user's needs, limitations, goals, desires and expectations are served"
As well as: "The organizations objectives are served as a result of serving the users" (Bard 2014)

The downside of allowing user involvement is that it is an additional required skill to the vast range of skills needed for development teams. It involves knowledge about how to collaborate with users on features and function trade-offs (Bard, 2013). User experience design practices offers a variety of techniques designed to specifically finding out what the user wants and thus bring value, with a relatively small amount of resources compared to traditional and extensive market research. These provide a large amount of information about the user by creating personas, processes for testing interfaces such as mock-ups and wireframes, user flows and testing them on real users before starting with the production code (UXMastery 2015)

These help to show in a cost-effective manner how the user would like to use the application and what are the most wanted features even before development starts. It is however somewhat of a case of debate if a person's experience can truly be designed beforehand, since people are highly personalized and what works for one might not work for another. Cummings (2014) mentions that "we can design interactive systems, not people". The core of UX is to although to design the interactive systems so that it evokes responses for the desired users that can be predicted statistically and effect the level of enjoyment and behaviour. (Cummings 2014).

According to Cummings (2014), the process consists of five phases that are, based on iterative steps and communicating the design steps effectively to the project stakeholders. Similarities to the lean Startup and agile development are substantial, therefore making user experience design a noteworthy methodology in the development team of web Startups for user centric design that is something that the user wants.

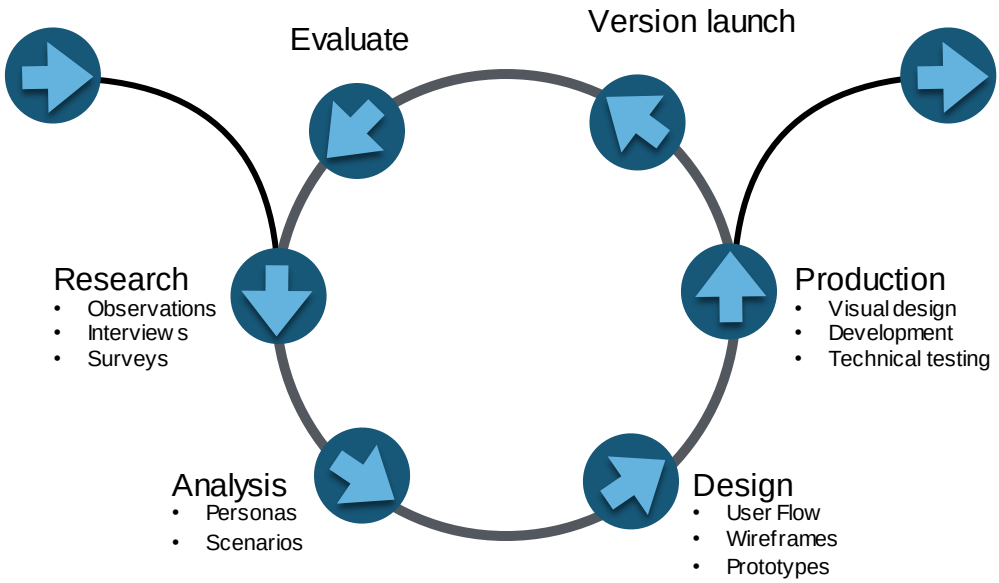


Figure 12. Illustration of the basics in the UX process (Cummings 2014).

2.6.2 Agile Software Development

The agile principles can be summarized into key values that reflect what agile project leaders should hold as their guidelines, this is called the agile manifesto.

> **Individuals and interactions** over processes and tools
> **Working software** over comprehensive documentation
> **Customer collaboration** over contract negotiation
> **Responding to change** over following a plan
> (Beck & al., 2001)

Agile development, however hyped it might be at the moment, is not suitable for every software project. Jim Highsmith (2009, 13), a founding member of the team that created the agile manifesto, points out that certain projects have very certain and unchanged specifications, where traditional waterfall might be more suitable. Software projects of Startups rarely belong to this category since there usually isn't any proof that the developed features will appeal to the customers without first trying them out (Blank & Dorff 2012, 46).

According to Highsmith (2010) there are three values that summarize the agile mindset:

> "Delivering value over meeting constraints"
> "Leading the team over managing tasks"
> "Adapting to change over conforming to plans"
> (Highsmith 2010, 17)

Highsmith (2010), states that being agile is principally a mindset, and less so about practices. The difficulty of it lies in balancing between chaos and order, creating just the right circumstances for innovation. The difficulty lies also in measuring the success of agile projects. Traditionally, projects have been measured according to scope, schedule and cost. The difference between traditional development and agile development emerges as most visible in the way they measure success. Highsmith (2010, 19) points out that traditional project management plans the whole project based on scope, schedule and cost. These are also the terms it is measured with. This is known as the "iron triangle" of project management. Highsmith (2010) suggests that agile projects should use another method to measure success specifically designed for agile projects.
This option measures Quality, Value and Constraints and puts more emphasis on measuring Value and Quality than the project Constraints.

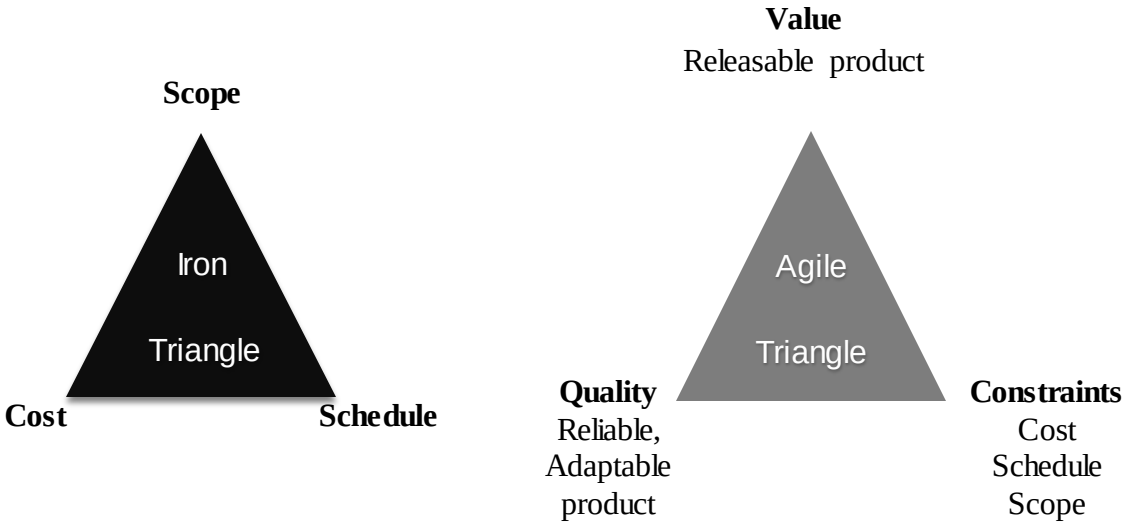


Figure 13. Illustration of the traditional project management triangles' evolution to an agile triangle (Highsmith 2009, 21).

According to Highsmith (2009), measuring agile projects with cost, scope and schedule is widely misused, even so by large research groups such as the Standish Group, although their publications are used widely for measuring success in agile projects. If experienced professionals struggle with measuring project success by value and quality, it suggests that measuring project success by these terms is potentially very difficult. Concentrating on staying in line with the planned scope, schedule and cost seems like the easy choice for a single project but provides only vague information to determine later Startup success.

As an example, if the measurements of scope, cost and schedule were used for the movie *Titanic,* it would be have been a huge failure by all dimensions in the traditional measurements - it exceeded its budget, schedule and scope by far, yet it was the first movie ever to create over $1 Billion in revenue (Highsmith 2010). An opposite example to show how measuring scope, cost and schedule fails to be the right attributes is the project *Iridium* in 1998, where Motorola attempted to bring wireless phone service to consumers through satellite phones. In terms of scope, cost and schedule the project stayed on track but in the end they only managed to get 10 000 subscribers instead of the projected 500 000. The project scope consisted of 66 satellites launched to orbit and

had a budget of $5 Billion. The consumers did not see the value in buying a $3000 satellite phone that didn't work inside buildings, while the 2G cellular phone got more popular and the increasing network infrastructure of cell sites solved this problem for the consumer at a fraction of the cost (24/7Wall St. 2009).

Based on this knowledge, there seems to be a clear need for aiming the measurement of success and failure in projects to prioritize value and quality right from the start of a new project as a Startup idea. The statistics also backs this notion, according to CBInsights that studied 146 failed Startups, the nr.1 reason Startups fail is that they build something that does not have a market need (CBInsights 2014). To conclude, the one single most effective way to improve the success of Startups seems to be having the users/customers involved with the development process early on and build a systematic feedback loop that prioritizes building the value and quality of the product for the customer, above project scope, schedule and cost.

2.6.3 Project Management in Agile Software development

The agile principles can be shortened effectively into what it takes for leadership for an agile project: "A traditional project manager focuses on following the plan with minimal changes, whereas an agile leader focuses on adapting successfully to inevitable changes" (Highsmith 2009, 17)

There are currently more than 10 methodologies that can be nested under the agile umbrella. The most popular agile methodology at the moment is Scrum, with a majority of about 56% using the practice, according to Version One (2015). The definition of Scrum is that "it's a process framework used since the 1990's within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value" (Schwaber & Sutherland 2013)

Highsmith (51, 2010) suggests that the single most important thing for a successful agile project is a self-organizing Development Team.

It is up to the Startup founder to lead the team for a mindset based on the following principles:

1. All work done should be based on a continuous flow of customer value
2. Execution should be focused on done rather than perfect
3. Delivery should be iterative and feature based in small batches
4. The aim should be for technical excellence for building quality for customers
5. The aim should be for simplicity to allow speed in delivery

(Highsmith, 2009, 28, 32, 34, 37, 40).

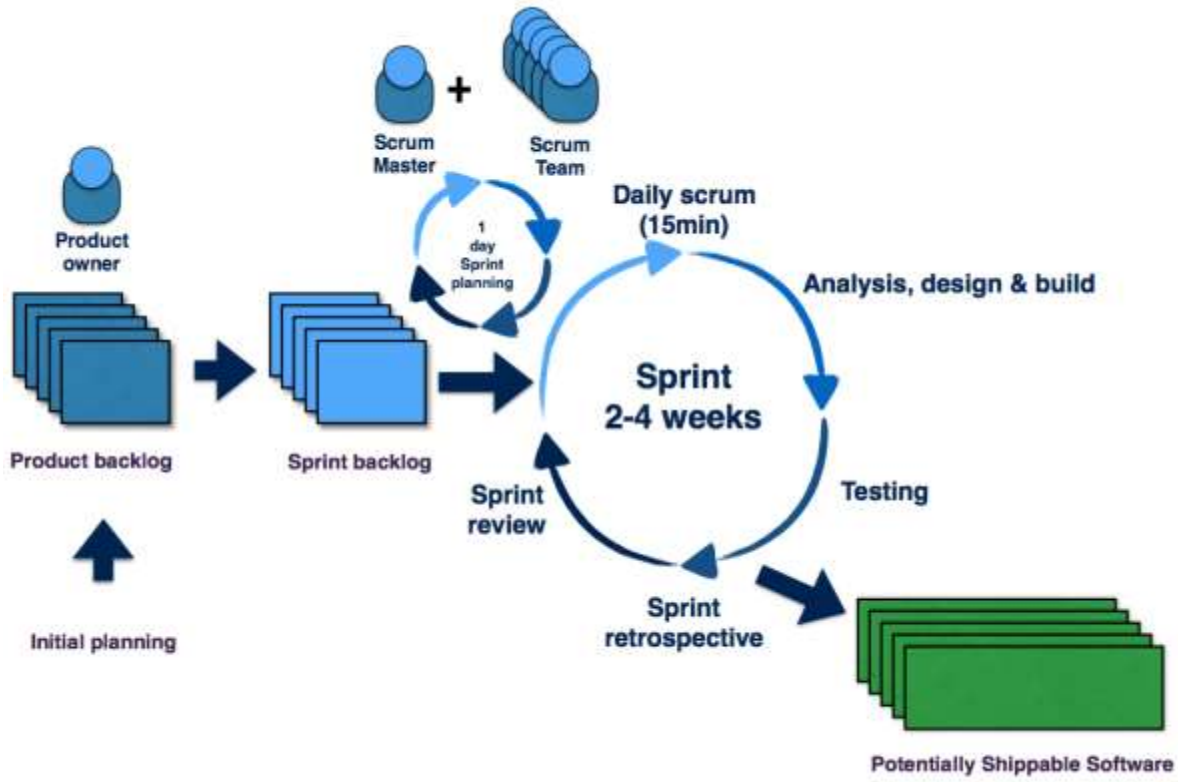


Figure 14. An illustration of a high-level overview of Scrum (Schwaber & Sutherland 2013).

Schwaber and Sutherland (2013) states that the roles of the project team should consist of a Product Owner that is responsible of maximizing the value of the product and the work done by the Development team. The Product Owner takes sole responsibility of the product backlog that consists of a curated list of requirements of the project.

The Development Team consists of a Scrum Master and the Development Team. The Development Team is self-organizing, which means they use the product backlog to produce releasable functionalities. The Development Team should help each other out so no member would become a blockage in the delivery of work. The scrum guide suggests a team of 5 to7 persons with a maximum Development Team size of 9 persons, before coordination becomes too difficult. (Schwaber & Sutherland 2013)

2.7 Startup failure and success

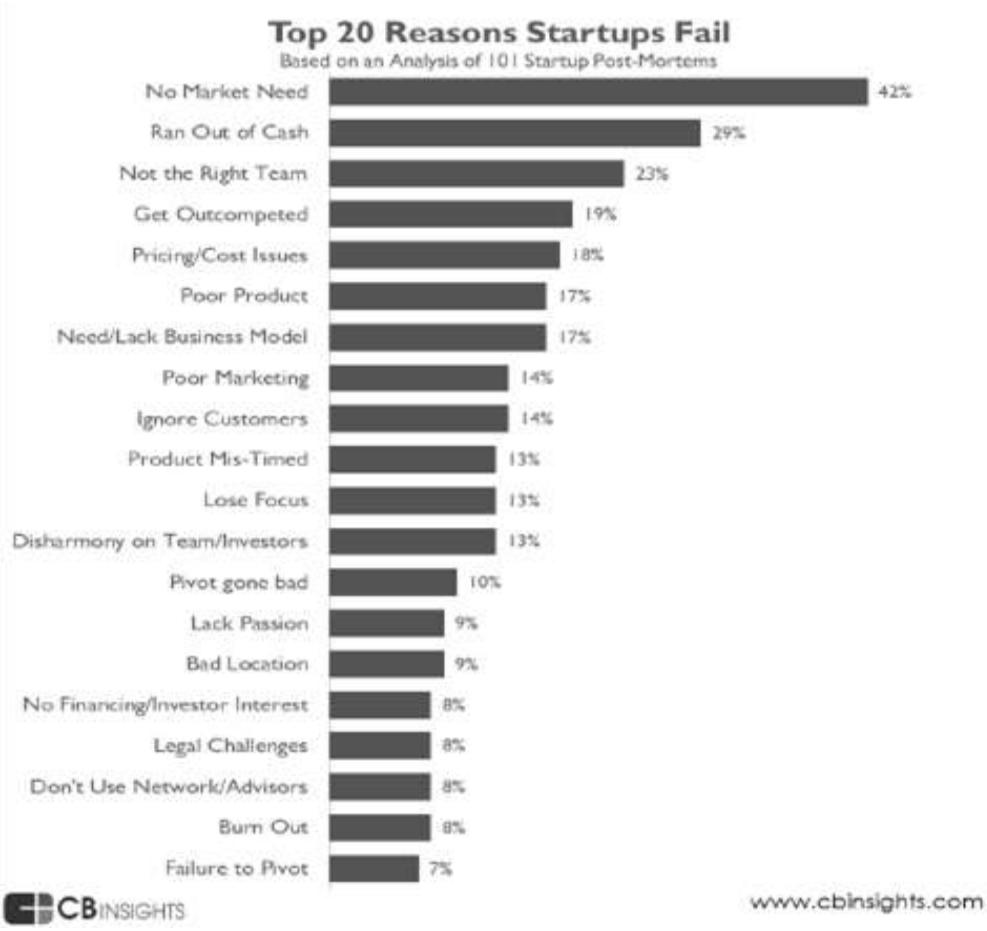


Figure 15. Illustration of technology Startups most common reasons for failure (CBinsights 2014).

To provide another perspective, Bill Gross conducted research of 100 successful technology Startups, based on Startups choosing the most important major factors that led to success.

1. **Timing: 42%**
2. **Team: 32%**
3. **Idea: 28%**
4. **Business model: 24%**
5. **Funding 14%**

(Gross 2015)

Summarized, in order of importance, timing as the most important success factor suggests that the markets are evolving quickly and with the proper insight, preparing for a market change early on could be a better strategy than trying to fit into the existing space. The team should also be skilled enough to be able to execute an idea until a product/market fit and a business model that is fundable in the eyes of the investor.

2.9 Summary of the theoretical frameworks

The following thought leaders on Startups are studied to form a tailor-made framework for the empirical study: Eric Ries suggests for a lean Startup to pair up Customer Development with Agile Engineering (Ries, 2012). Steve Blank suggests to write down the different aspect on a Business Model Canvas using Customer Development to validate the assumptions, whereas Ash Maurya proposes to use the Lean Canvas for early stage Startups. In my opinion, the importance of design in the early stages seems to be discreetly neglected, especially when the mentioned frameworks emphasize measuring success by quality and value for the customer. Consequently, the thesis suggests adding processes known from the field of User Experience Design (UX) to provide tools and methods that are used to find out how to improve the user experience, product value and quality from the customer viewpoint.

The following figure represents my view of the suitable theoretical frameworks involved in building the foundations of a Startup developing products that are consumed via web browsers or mobile devices. All the different approaches consist of iterative and incremental processes and are therefore assumed to complement each other, providing a unified workflow for the aspects in business, development and design.

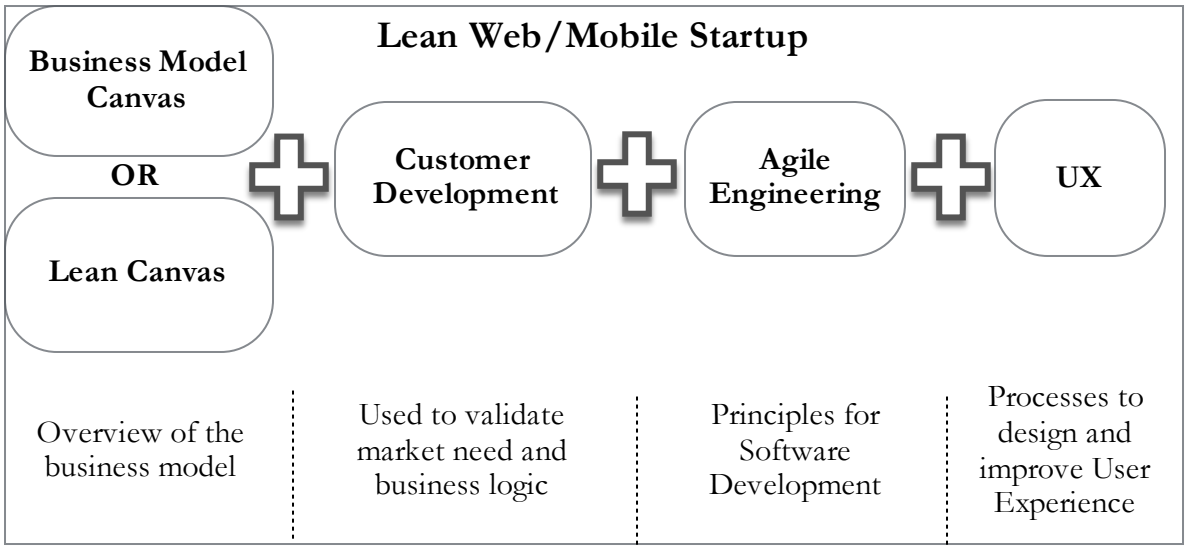


Figure 16. Illustration of a compilation of theoretical frameworks by different authors that are associated with building a Lean Startup that develops Web and Mobile products. (Osterwalder 2010; Maurya 2012; Blank 2012; Ries 2011).

3. Product experiment 1 - Unhidden

3.1. Selection

The empirical part of the project started with coming up with an idea for a project to focus on as the basis of a Startup. I find that Bill Aulet (2013, 17) nails a big part of this dilemma by seeking an answer to the following question: "what can I do well that I would love to do for an extended period of time"? This was used for the selection of ideas, combined with the phrase by Paul Graham (2012), who suggests the following: "The way to get good Startup ideas is not to try to think of Startup ideas. It's to look for problems, preferably problems you have yourself".

3.2. Learning from failure

"Success is a lousy teacher, it seduces smart people into thinking they can't lose". (Gates 1995, 15) Many entrepreneurial thought leaders and the Silicon Valley tech community praise failure as the best way for self-improvement and as one of the best ways to learn. I do agree with the notion, until the point until that learning from failure becomes an excuse for not trying one hundred percent or giving up too soon. Validating ideas based on failing and removing the wrong, building an improved version and continuing to iterate does require a certain mindset of the founder that is not easily tipped after bad feedback. This mindset also understands that learning quickly what does not work is as valuable as learning what works. Some sort of failure with any dimension of the business seems to be inevitable at some stage, so better let it happen sooner rather than later with spending as little resources as possible to learn what works and what not.

3.3. Product experiment 1 – Execution

The first idea was for a mobile app – a two-sided marketplace for "secret" locations that would let the user take a photo, geotag the location and sell the location information. The project was named Unhidden. The idea was greatly motivated with building a solution using the second generation of micropayment systems that was introduced in 2010s rather than focusing on solving a market need.

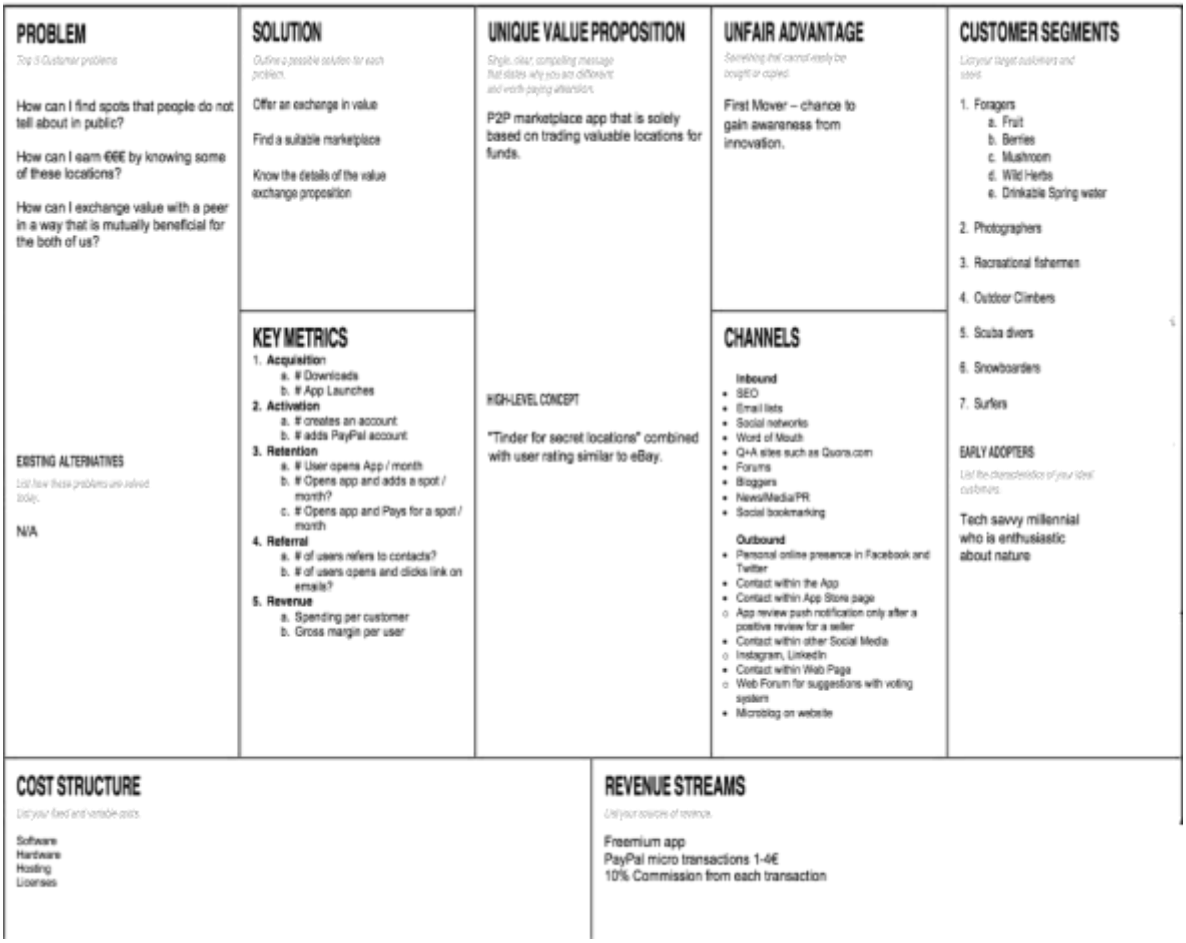


Figure 17. Illustration of a first draft of Unhidden Lean Canvas (Developed by the author)

According to CBInsights (2014), the biggest reason (42%) for Startups to fail was because there was no real market need. Ash Maurya (2012, 81) suggests to build a demo of the solution to test if it is something that people really want. He also suggests to test out if the problem really exists before building a solution, which I ignored at the time of

the project. I jumped right in to see if people liked the solution that I had in mind, which could be counted as case of premature optimization.

Figure 18. Illustration of screen mock-ups for Unhidden. (Developed by the author)

For demoing purposes on how the app would work, I created an interactive prototype with Invision prototyping tool that acted somewhat like a native app. I also created a landing page to describing the idea and sharing screenshots on dedicated Finnish communities in Facebook groups and forums dedicated for fishing, foraging and outdoor extreme sports.

3.3.1. Measuring User feedback

I quickly realized from the feedback that was given that it was not really a solution for a problem and the assumed users did not really want to share their secret spots even if they would get paid for it. I set up Google Analytics to measure the visitors of the landing page (unhidden.co) and a free Mailchimp e-mail list to collect e-mails of people that signed up for a beta testing list. Ultimately, the landing page had 552 unique visitors in 2 weeks and only 8 persons gave their e-mails by signing up. On the other hand, roughly 10% of the feedback and comments on the forums and Facebook groups was surprisingly positive and encouraging, including one online publisher (luontoon.fi) that asked for more details to write a publication about the app.

Engaging in conversations directly with people that belonged to Facebook groups that consisted of assumed early adopters proved to be the most effective way of getting feedback that could be used for a problem / solution fit. The feedback especially in foraging and some outdoor sports communities were at times even downright hostile about keeping their spots secret. Many people also directly stated that this was a not a problem for them and finding secret spots with the help of an app or putting a price tag on secret spots was not the solution they were after. Some also commented that it took away the excitement of actually finding a secret spot. This told me enough about being off track with actually solving a real problem and being too excited about building a solution and just for the fun of building a mobile app. The increasing complexity to execute the idea made it also more difficult for people to understand its value proposition. Eventually, the whole idea started to sound to me as an oxymoron - an app that is based on revealing secret spots. Turned out that a great majority of the people giving feedback also wanted secret spots to remain as a secret and not on sale by some app, hence making the core of the idea invalid.

Guy Kawasakis' (2012) presentation on Slideshare suggests that if a Startup wants to build a community, the first thing it needs is a great product and some "early evangelists" that care enough to build a cause around your product or service. Based on the feedback from different forums, it became very soon obvious that this idea did not have those attributes. The amount of negative feedback would be very hard to overcome and building an initial fan base would surely prove to be very difficult.

Additionally, it became apparent that the app would also have a high technology risk that required complex security features firstly for mobile payments and secondly to avoid fraud in manipulating the metadata for the locations or the photos. Also, the app would need to have a very sophisticated business logic to not allow users to sell locations for potentially harmful purposes.

I decided to suspend the idea after a month mostly due to not reaching a problem solution fit, but also because of the technical complexity to build a Minimum Viable Product for the app outreached the initial expectations and became too difficult to pursue in a sensible timeframe. In the end, that failure could be considered a success for personal development though, by focusing all my time in learning more about web development, mobile app prototyping, app marketing, UX Design and iOS app development for and a wide range of core skills in mobile based business development and app development. Ultimately the failure only cost 10€ that I paid for the landing page domain name to perform the smoke test.

4. Product Experiment 2 – OPAS

I decided to start with a completely different idea, in a more structured manner, applying the frameworks described in the theoretical part of the thesis. I also started with examining the problem more carefully before jumping into the solution. I personally had a general problem of finding people with certain skills within my university that are interested in collaboration with projects but found it difficult mainly because the current solutions were very fragmented and slow processes. It seemed to me there was need for an on-demand web service for finding people based on skills within the organization that did not yet exist.

4.1. Planning the idea

The planning started with creating a lean canvas to get a better overview of the whole business model and the various parts that included into it. Ash Maurya (2012, 6), suggests to quickly draft a lean canvas from your hypotheses of the various elements in the Business Model.

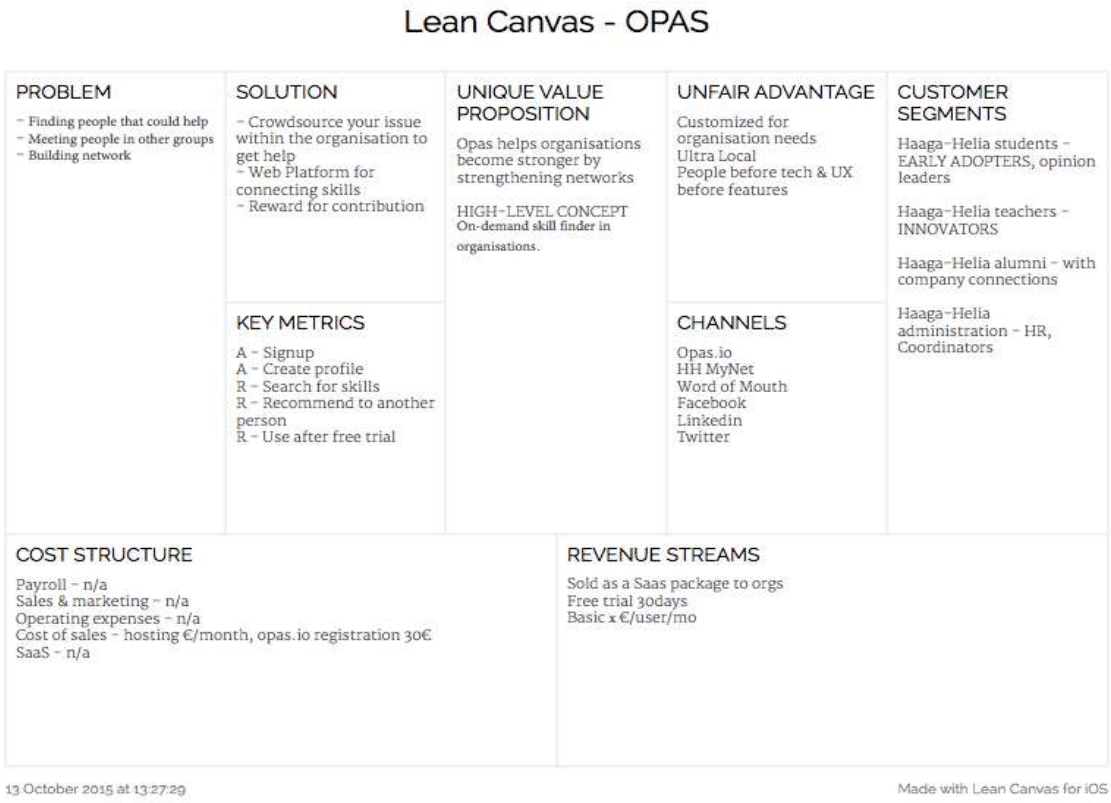


Figure 19. Opas first draft of the Lean Canvas (Developed by the author)

4.2. Implementation

The implementation for Experiment 2 started with joining in a course as a product owner in a collaboration course between the Haaga-Helia Startup School and a number of Business Information Technology students. The project started with envisioning the project and setting up goals and scope for development and defining how project success would be measured. The project scope and end-result was a Minimum Viable Product that would be tested for a market need.

Completing a quick market segmentation forced to research the potential size of the idea and a general overview of the market involved.

Industry	Institutions of higher education in Finland	IT Companies in Finland	Public sector in Finland	Global Enterprises
End user	Student Teacher	Developer Designer Sales Marketing	Employees	Employees
Application	Collaborative learning Internal Communication	Internal communication Employee collaboration	Employee collaboration	Employee collaboration
Benefits	Access to expertise Self development Alternative learning methods	Innovate faster Improved work efficiency Improved job satisfaction	Increased collaboration & stronger network	Increased collaboration & stronger network
Lead Customers	1. University of Helsinki 36k 2. University of Turku 20k 3. Aalto University 19k	1. Nokia 61k 2. TeliaSonera 28k 3. Tieto 14k	Itella 17K VR 10K Valtion työmarkkinalaitos 78K etc.. The state: total 520K	Walmart, 2,2M McDonalds 1,7M Sinopec, China 1M Hon Hai 1M
Market Characteristics	Institutionalized Government funded	Tech enthusiasts	Institutionalized	Mainstream
Partners / Players	Opetushallitus OAJ Opetusministeriö Microsoft IBM LinkedIn	Microsoft IBM Nokia	Microsoft IBM Tieto	Microsoft IBM
Size of market (potential customers)	Uni students - 163k Polytechnic students - 110k	> 300k	> 600k	~1B
Competition	LinkedIn Facebook groups Intranets Opinder	LinkedIn Yammer TIBBR Intranet	LinkedIn Own internal networks	Own Internal networks
Platform	Mostly Windows	Win/Mac/Linux	Mostly Windows	Win/Mac/Linux

Figure 20. Illustration of Opas initial Market Segmentation chart. (Developed by the author)

4.3. Project management for the Opas Project

The project had the following fixed resources and constraints: no assigned budget, 3 months to produce an output, a development team of two product owners, 12 Information Technology students with a very high variety of skills and background.

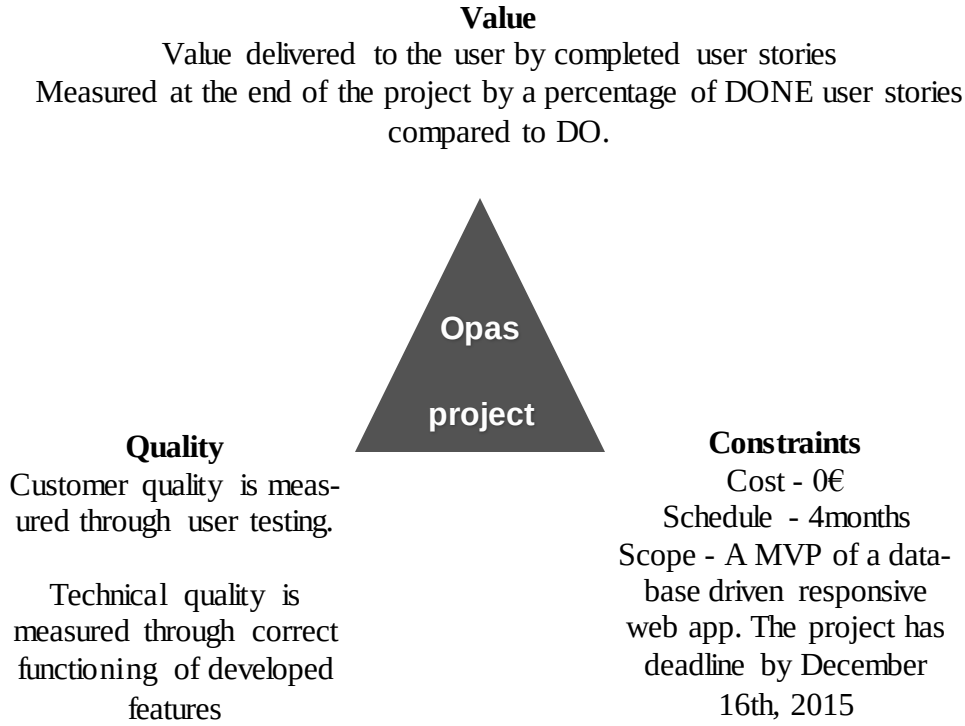


Figure 21. Agile triangle for the Opas project. (Developed by the author)

The project management was also constrained by the team having to use their own computer for development that meant two different Operating systems (OS X & Windows). Additional constraints was zero budget and the schedule, which was 4 months. The team could only meet only once a week and mostly work remotely.

4.3.1. Setting up an agile project

The Project management for Opas became an iterative process as well, as strictly following a Scrum process proved to be impossible, due to the size and inexperience of the team. Especially estimating how long developing would take turned out to be difficult. We held a feedback session after each iteration and improved the process according to the feedback.

Eventually, the best form for the team turned out to be a simplistic but effective to-do list with TO DO, DOING, TESTED & DONE boards to keep things simple so the focus could be on development. Everyone in the team could post tasks on the TO DO list.

Figure 22. Evolution of iteration/sprint structure using Trello. (Developed by the author)

This turned out to be the right way to achieve progress for this team while keeping the project somewhat agile and the team still self-organizing around the tasks.

After the development team members first got to know each other and a good enough remote communication level was reached using the Slack communication environment, we focused on understanding the users' problems. Learning from the first experiment, the center of attention was on finding out if there actually was a problem worth solving. According to Ash Maurya (2012, 71), a founders first instinct is to conduct a bunch of surveys or focus groups using quantitative research methods. This is not considered a

good idea by Maurya because of the information collected is very likely to be worthless i.e. formulating a good survey is nearly impossible when the problem is still not verified. The first business model was based on assumptions and used to start validating the different aspects and change them if needed, when learning took place. The first riskiest assumption about this plan was to find out if the problems that I assumed in the business model actually was a problem. Ash Maurya (2012, 82) suggests to find out:

- How the users rank their top 3 problems?
- How the users currently solve these problems?
- Is this the right customer segment?

This was achieved with performing in-depth interviews for the assumed early adopter segment and learn more about their worldview. In this case it was other students in my own university and as Steve Blank (2012, 288) figuratively suggests founders to "get out of the building" to find people for customer validation, in this case though it made more sense to stay in the university building where those people could be found. Therefore, we started with a qualitative method of an interviewing that was conducted on 26 students by asking about the problems they have related to learning.

Setting up the stage	Testing the problem
Introductions - tell briefly about the idea 2min	Rank the problems of the interviewee 2min
Collect customer segment info 1min	Explore user worldview, workflow, needs 10min

Figure 23. Summary of a simplified problem interview (Developed by the author)

USER	Description of user	Problem rank	When? Where?	User needs	USE CASE	Feature List
STUDENT 13	Independent learner, Don't like a public profile	1. Build network 2. Find skills 3. Meeting new people	When getting ideas about a project	Experienced insight when learning new topics	Contact a person who completed a similar project	Get information from my other accounts like LinkedIn.

Figure 24. Example of collected information (Developed by the author)

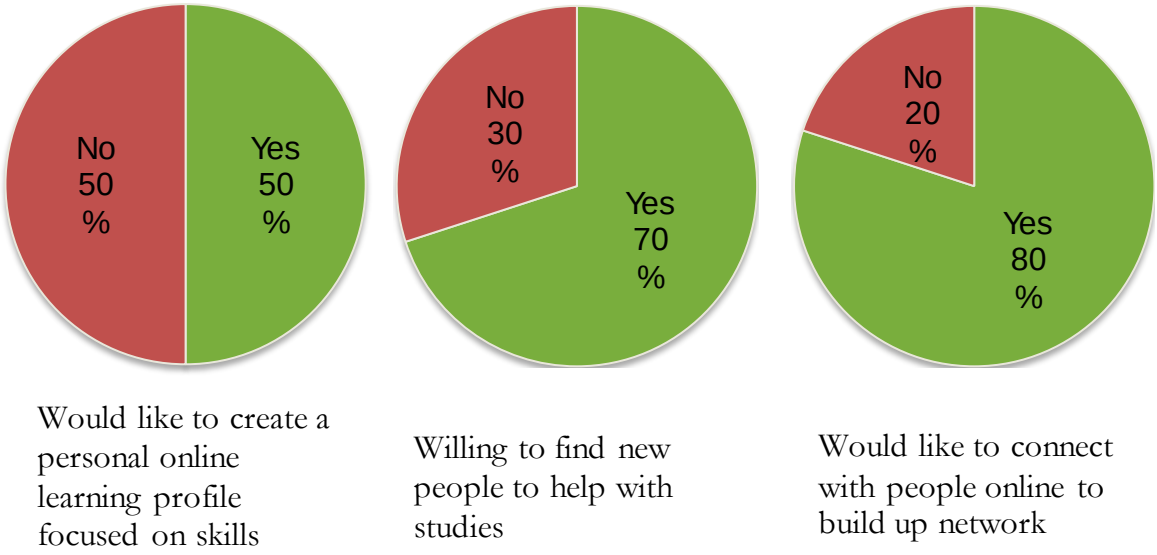


Figure 25. Most occurring problems compiled from the qualitative interviews related to the solution. (Developed by the author)

Figure 26. Most occurring problems turned into user stories. (Developed by the author)

These large user stories were called epics and functioned as the foundation of all development. They were used to build the features needed so that the user could complete a certain story.

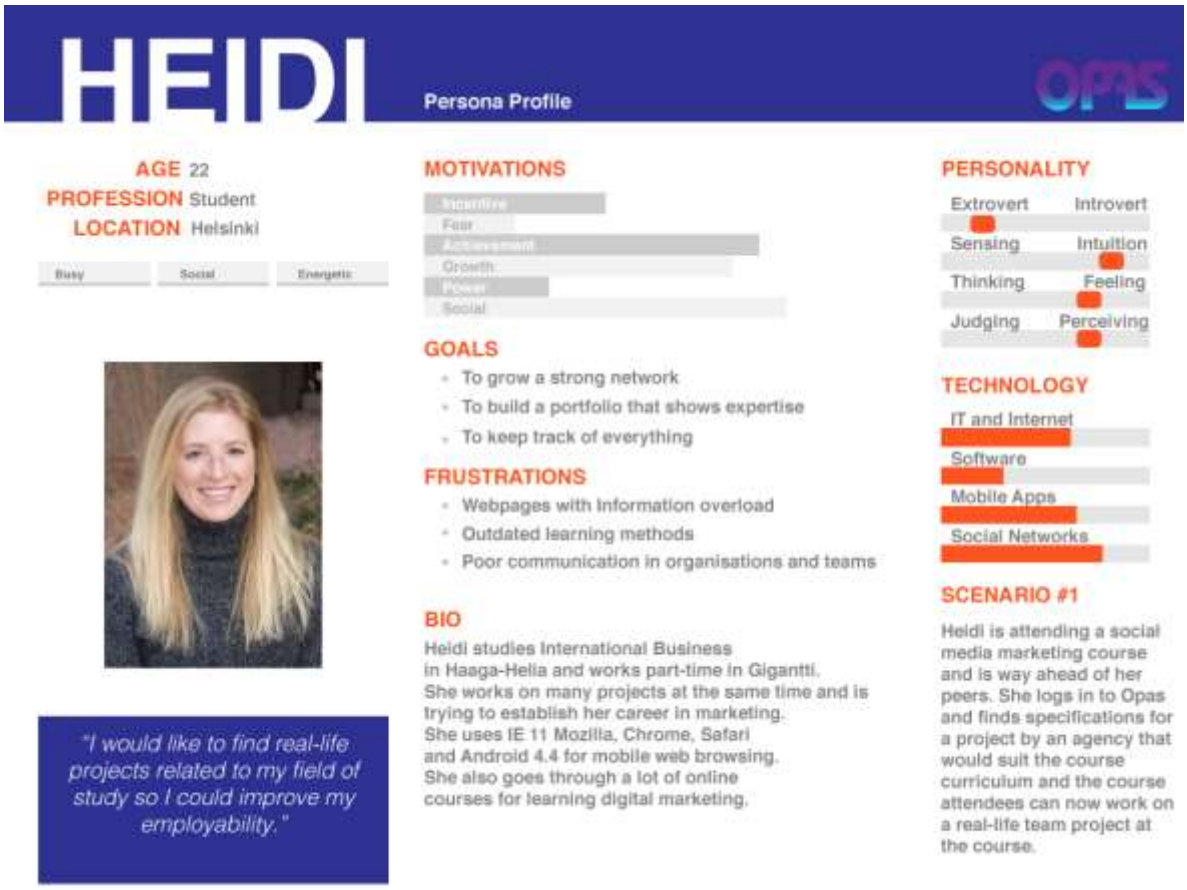


Figure 27. Illustration of the user persona. (Developed by the author)

This user persona was created to provide a concrete example of a user that belonged to the largest user group of the Opas product. It was meant to help to create the look and feel of the design of the site and a reference point for user behaviour.

4.4. Information architecture

After the user was specified different site maps and user flow diagrams were created. This was used for determining a desired path for the user in browsing the site and to provide help in forming an initial information architecture for the web app.

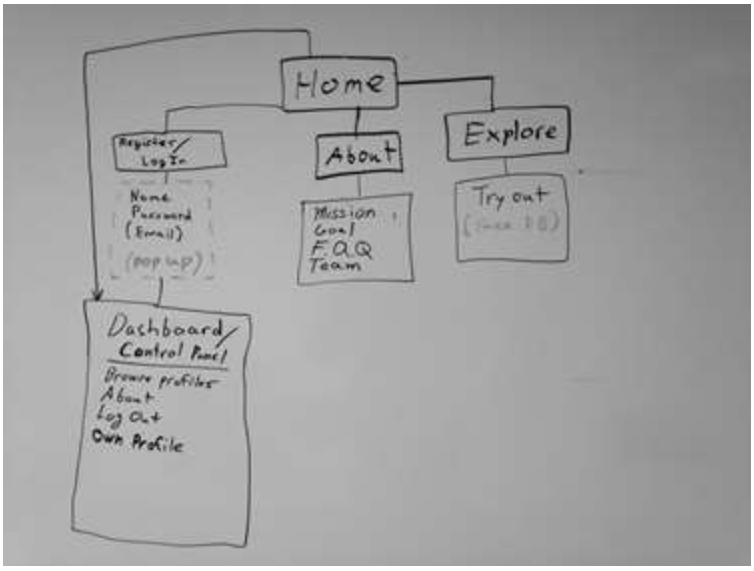


Figure 28. First draft of the site map (Developed by the author)

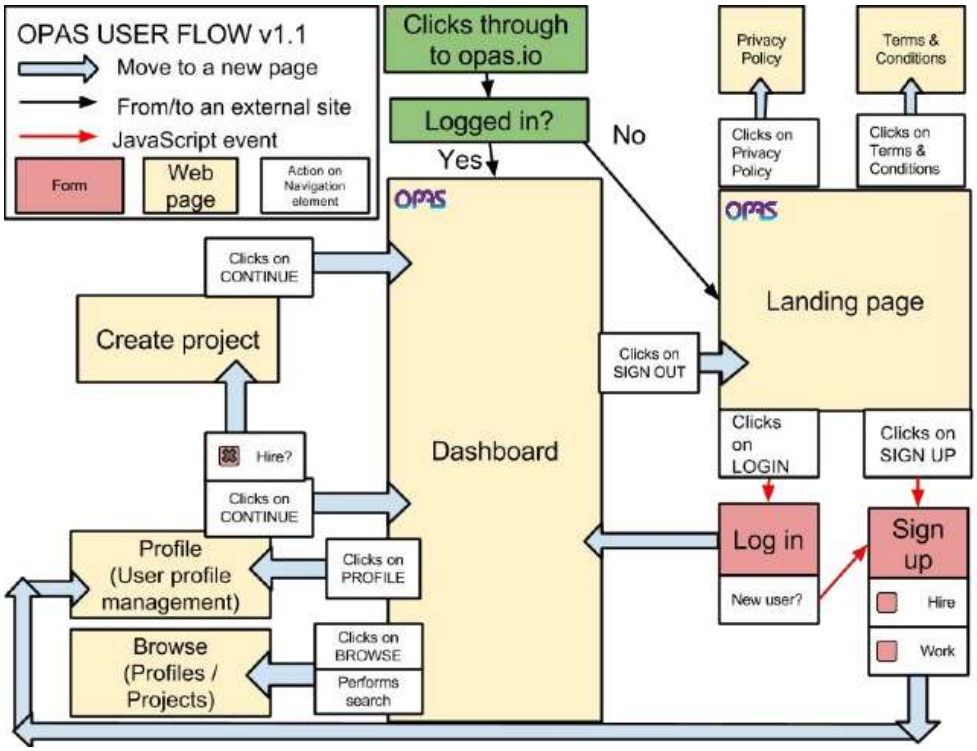


Figure 29. User flow for the OPAS MVP (Developed by the author)

4.5. Interaction & Visual design

This step involved to quickly wireframe how the application would work. Due to time constraints, the focus was on a working process of interaction rather than highly detailed visual design. Even so, I also created a logo and preliminary colour palette for Opas early on to provide a guiding point for visual direction.

Figure 30. First paper mock-ups of Opas screens on mobile (Developed by the author)

Figure 31. Opas logo (Developed by the author).

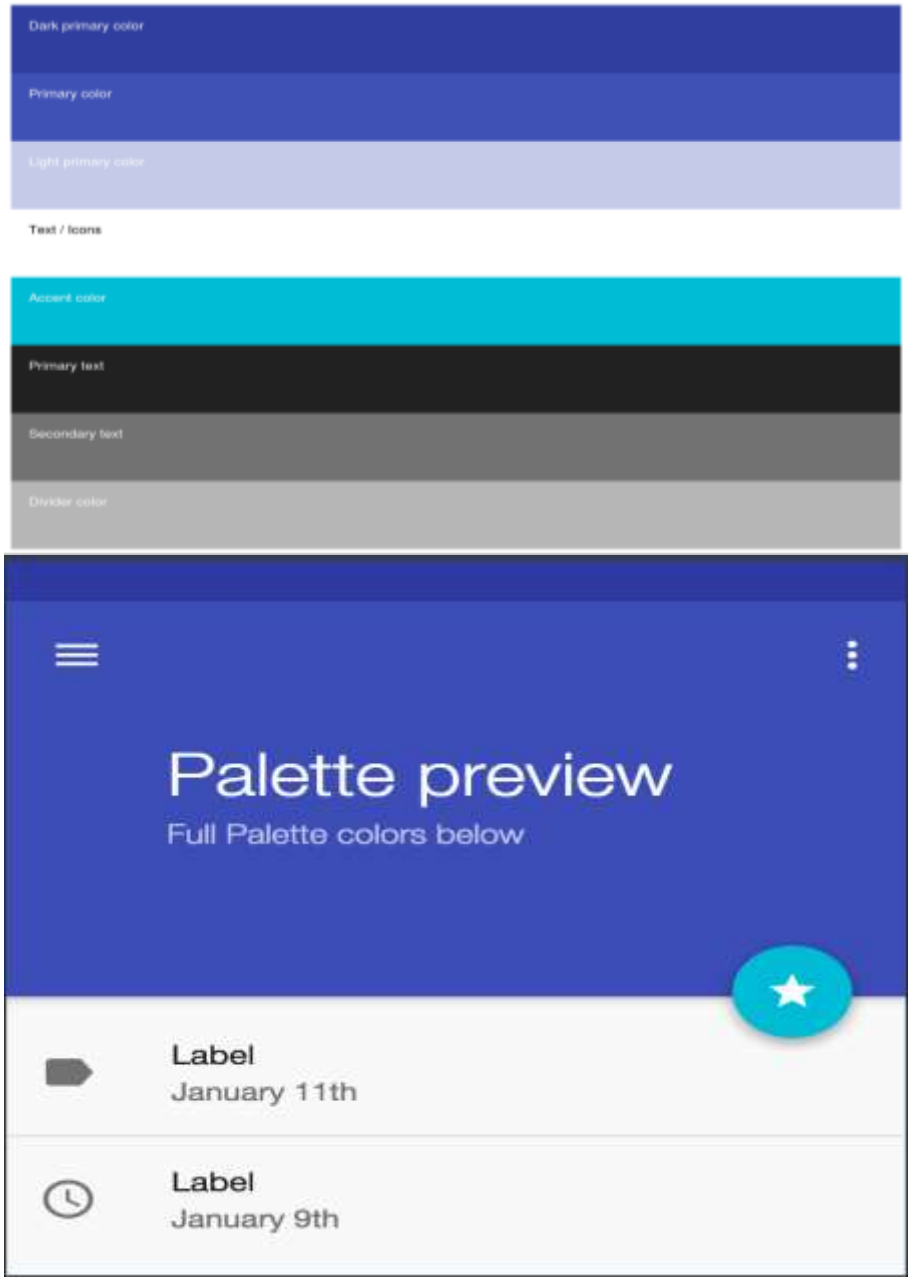


Figure 32. Opas colour palette (Developed by the author)

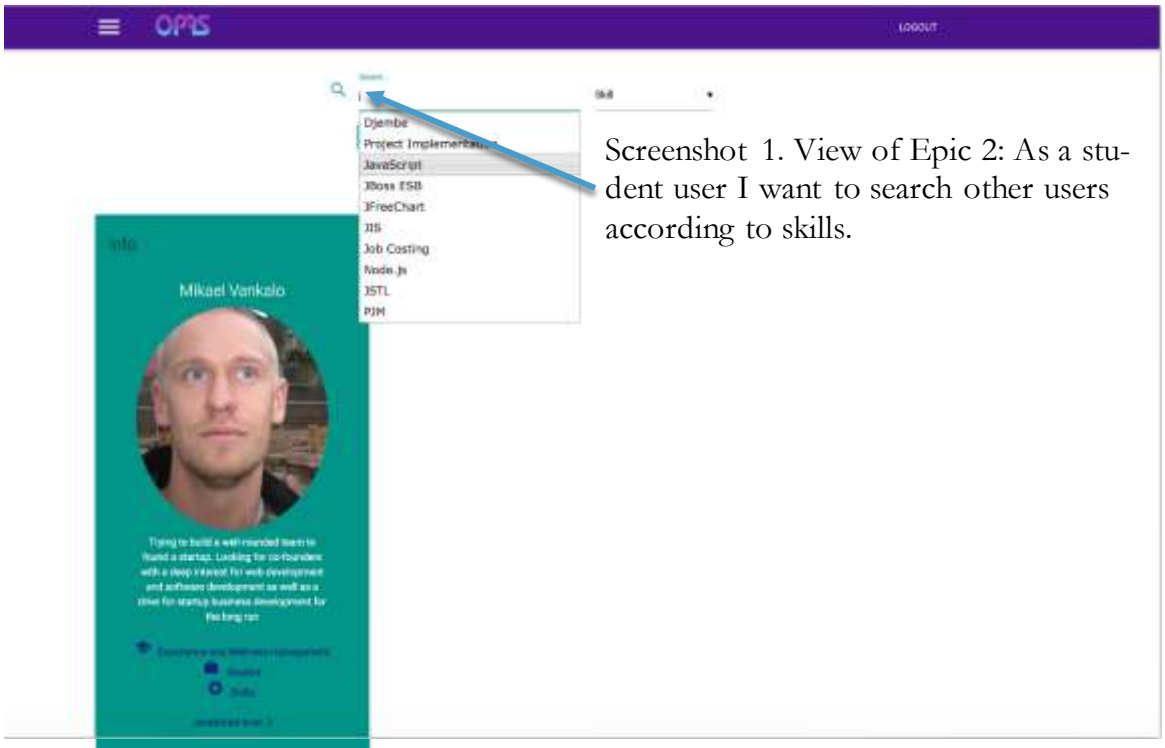
The design aspect were decided by Development team voting early on in the project as well to provide consistency and a visual guide to ease development by giving specifications for design for the Development Team.

4.6. The product

The site of Opas.io was located on a prototype server (proto474.haaga-helia.fi) provided by Haaga-Helia UAS and is now archived to the following address: http://web.archive.org/web/20160110160414/http://proto474.haaga-helia.fi/

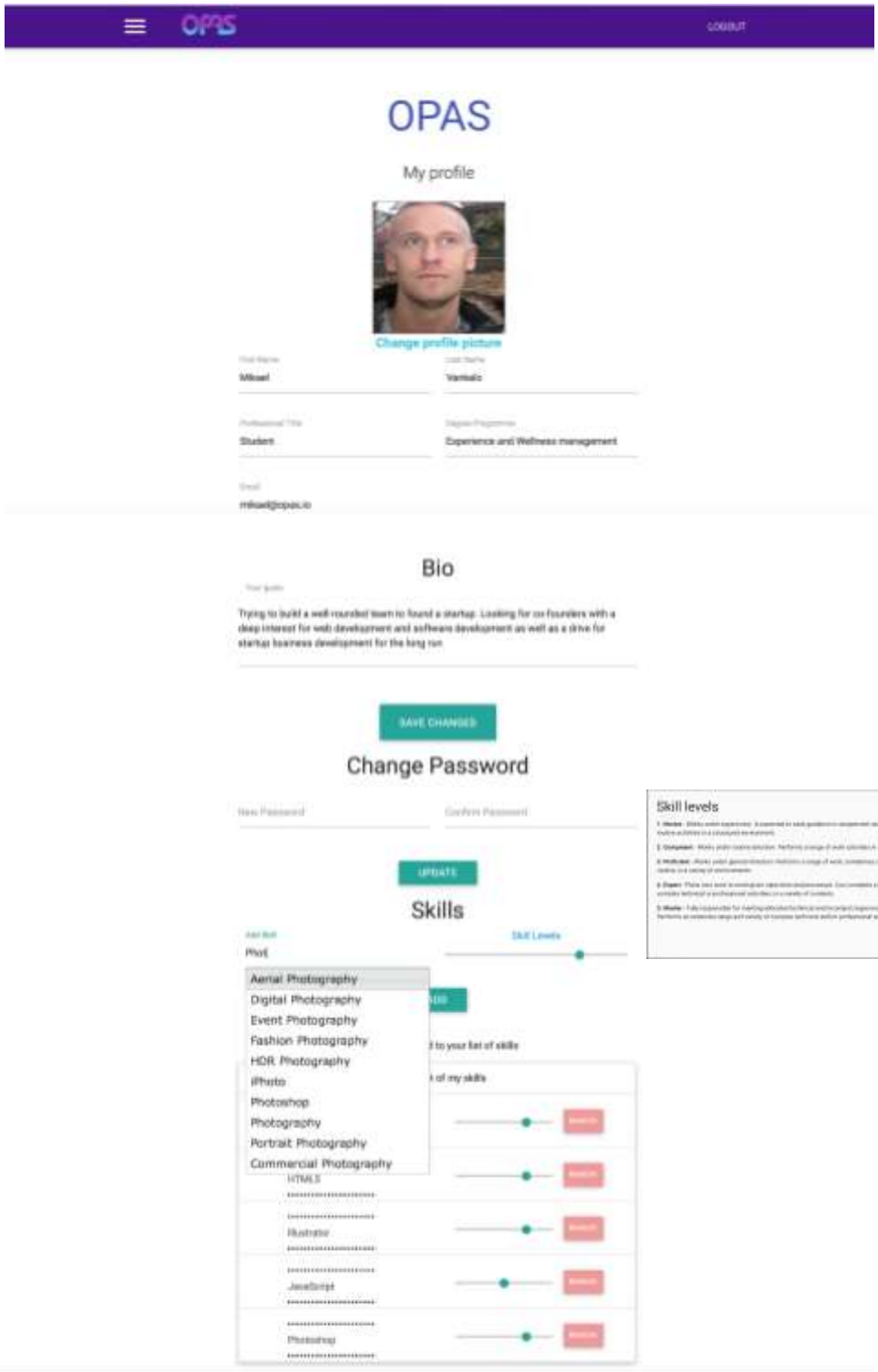
4.6.1. Screenshots

All screenshots are taken in desktop mode (1366x766px).

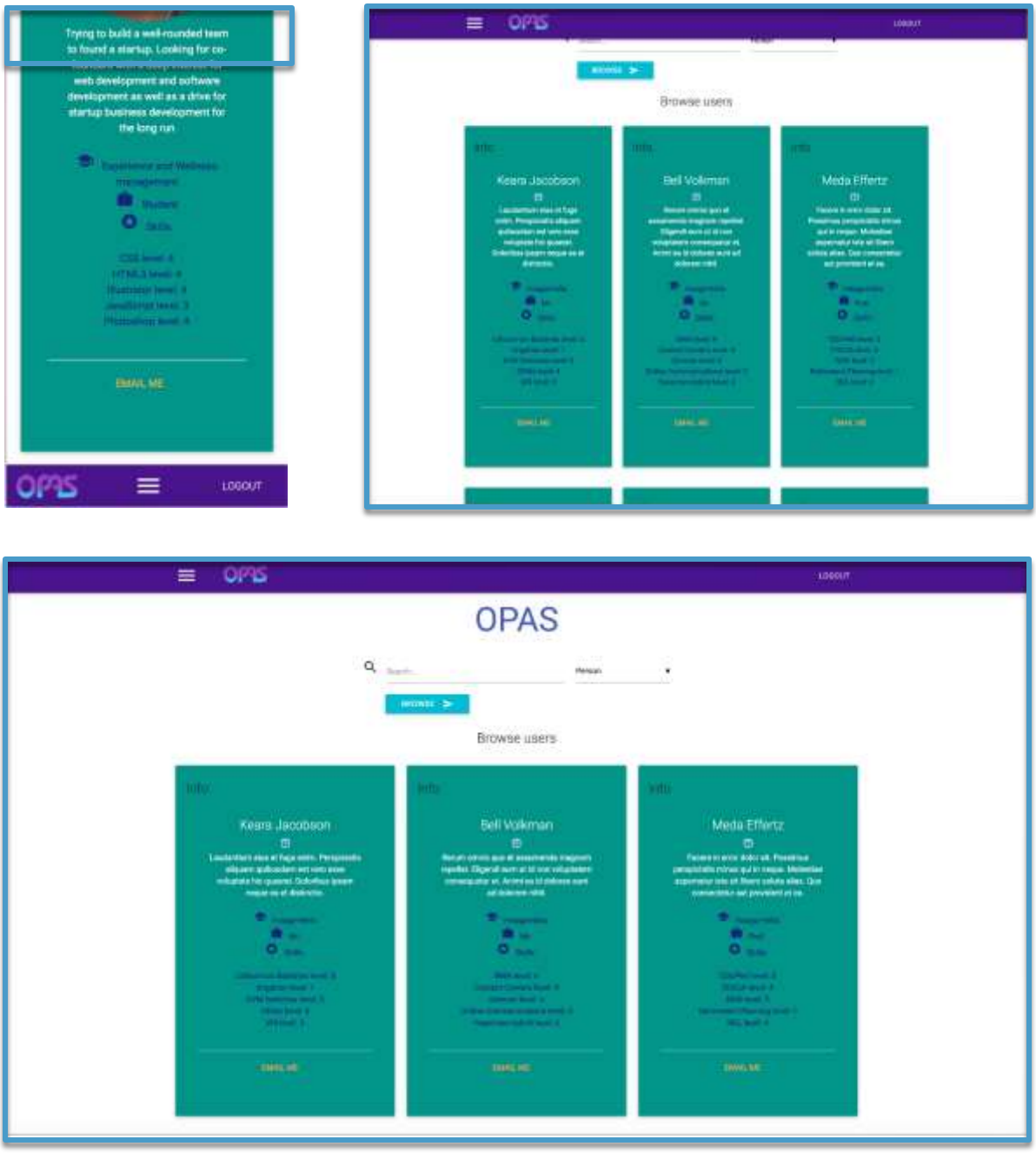


Screenshot 1. View of Epic 2: As a student user I want to search other users according to skills.

Screenshot 2. View of Epic 3: As a student user I want to connect with other users to build up my network.



Screenshot 3. Epic 1: As a student user a want to create a profile that shows my skills and goals.



Screenshot 4. View of browse.html page in responsive layouts with breakpoints for mobile, tablet and desktop screen sizes. (Developed by the Software Engineering Project team SYS4TF222-1.)

A lot of development effort was spent to create only one responsive version of the web app that would work on all devices. The major differences between mobile and desktop screens is the positioning of the navigation bar to the bottom of the screen to ease the

usability even on with one handed use on mobile screens. This proved to be more difficult than expected, since customizing the code of the Frontend framework Materialize based on Google material design caused several conflicts later on.

4.7. Usability testing

4.7.1. Usability testing on location

The purpose of the usability testing was to see what made the users confused or what they did not understand about the product. The aim was additionally to measure how well users could complete the larger user stories known as epics that were the foundation for the development of the application. If they could not be completed, the reason behind failure was noted down and used as the basis for further improvements in the development of the Opas web app.

The following factors were used as metrics to measure early progress:
A. User visits landing page
B. User signs up
C. User browses other users based on skills

The sample size for the first usability test needed to exceed a minimum of 5 participants. It was purely qualitative research based on finding user insight. Adding more test users does not significantly add to the findings in usability problems and that with a number of 5 persons over 80% of the usability issues can be found (U.S Department of Health & Human Services, 2016). The biggest usability problems were almost all related to design issues rather than functional ones.

The amount of severe usability issues found clearly showed that the MVP was not ready and most likely caused the user to abandon the site on several occasions if the site was visited by remote and not in a test environment.
The usability issues were divided into two categories, severe and minor. The severe issues being the ones that cause the visitor to not know how to proceed, therefore most likely to abandon the site.

1. Registration form closes after invalid password (e.g. password was too short). 2 out of 7 failed to sign up because of this problem. Sign up with OAuth was additionally requested by test users (i.e. social login with Facebook, Linkedin and Google+ etc...).

2. Log on form also closes when credentials don't match records. 2 out of 7 test users failed initially to sign up because of this issue.

3. Registration modal text boxes were too small to be selected, especially on mobile screens. 3 out of 7 had issues with selecting the right box to type in.

4. After profile creation, a button was needed to continue to browsing other profiles. 3 out of 7 test users failed to find the way to continue to browse profiles without guidance.

5. Contact form on the landing page was unclear for the user of where the message would go or what should be typed into it. The users did not mind to leave their email though. 2 out 7 did not understand the logic behind the contact form.

6. Profile pictures take too much space on the browse profiles page and dominate the focus of attention. 2 out of 7 did not note the skills of the profiles.

7. The landing page did not seem to be intuitively scrollable. 3 out of 7 did not scroll down on the first page without guidance to find out more about Opas.

Some minor usability issues mentioned by test users:

1. Link for skill level could be easier to find.
2. The other pages were not found to be intuitively scrollable.
3. Location could not be added to profiles.
4. The burger menu is confusing for and unnecessary for desktop users when there is more screen space available to place a normal navigation bar.

4.7.2. Remote usability testing

Additionally, automated remote testing was set up to conduct two weeks of quantitative studies in the form of website analytics. The aim was not gain as much visitor as possible but to focus on getting mostly the visitors that belonged to the assumed early adopter segment. This segment consisted of was a fairly narrow group of students in Haaga-Helia. This activity involved keeping the site online and tracking down the visitors with Google analytics and Hotjar analytics tools that are currently both free of cost to use. They allow to track the behaviour of those website visitors that enable tracking on their browsers. For quantitative studies a suggested sample size is more than 20 users to get any data that could be named as statistically valid (U.S Department of Health & Human Services 2016)

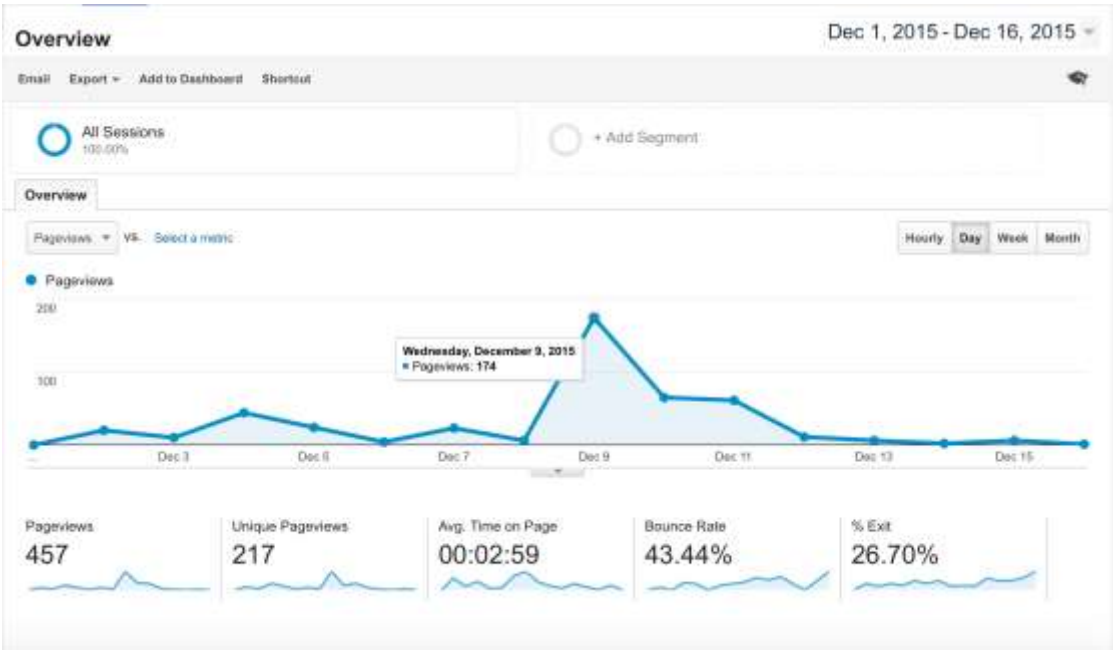


Figure 33. Google analytics statistics Overview from the opas.io landing page. (Developed by the author)

The site www.opas.io was very moderately marketed on December 9th on relevant social media channels such as the Haaga-Helia Facebook page and Twitter account. A profile for Opas was created on Facebook and Twitter with a couple of posts and linked to the site opas.io. The reach through the channels that were tested consisted only of a couple of hundred users and it was aimed to increase the sample size to be big enough for the quantitative usability study.

The Google analytics overview data consists mostly vanity metrics at this stage, but can however, be used to draw some conclusions and subjects for future improvements. Simple posts to relevant Facebook groups resulted in a spike in visitors for the site on December 9th, so Facebook marketing seemed to work with little effort. Most of the visitors on the site were not tracked or did not allow to be tracked from their browsers and the reliability of the Hotjar tool for analytics was compromised. This led to only 15% of the visitors being tracked by the Hotjar tool for their user flow during the two weeks.

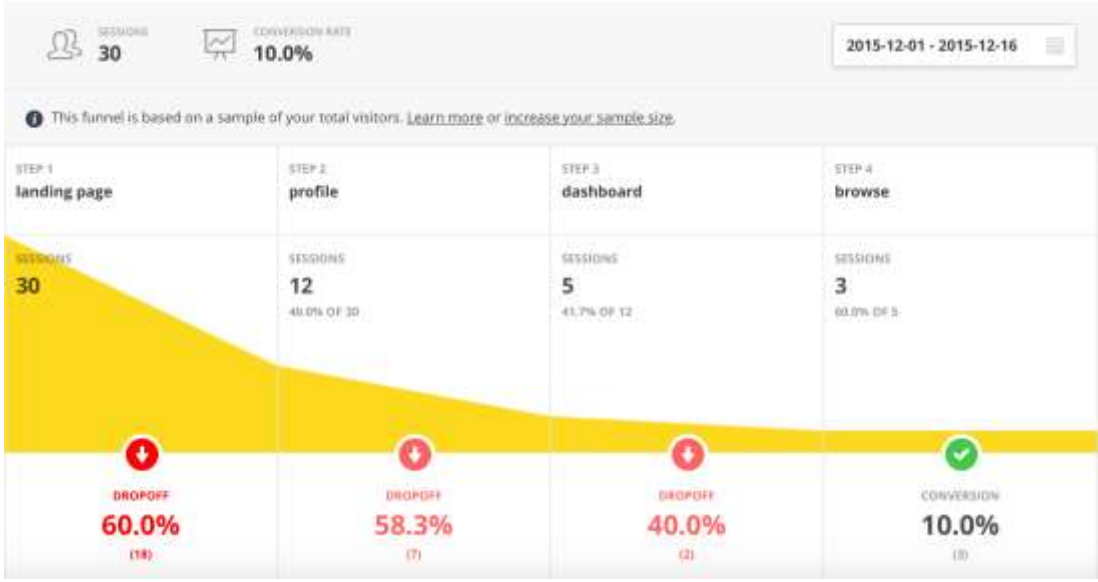


Figure 34. Hotjar analytics funnel measuring user flow for all pages (Developed by the author)

The conversion rate measured how many visitors end up on the page where it is possible to browse other profiles. Even though the sample size is barely over the minimum limit of 20 users, a 10% conversion rate could be considered a success considering how untested and raw the first version of the site was when it was released. This data also shows that 40% of the visitors for the landing page gave their email address to sign up, which might somewhat prove the effectiveness of directed marketing on social media channels and that the sign up as a call to action was somehow working.

Social Network	Acquisition			Behavior		
	Sessions ↓	% New Sessions	New Users	Bounce Rate	Pages / Session	Avg. Session Duration
	34 % of Total: 15.32% (222)	47.06% Avg for View: 57.21% (-17.74%)	16 % of Total: 12.60% (127)	52.94% Avg for View: 65.32% (-18.95%)	3.41 Avg for View: 2.66 (28.37%)	00:06:30 Avg for View: 00:04:37 (40.96%)
1. Facebook	28 (82.35%)	42.86%	12 (75.00%)	42.86%	3.93	00:07:54
2. Twitter	5 (14.71%)	60.00%	3 (18.75%)	100.00%	1.00	00:00:00
3. LinkedIn	1 (2.94%)	100.00%	1 (6.25%)	100.00%	1.00	00:00:00

Figure 35. Effectiveness of social media channels on December 9th. (Developed by the author)

The analysis of resulting traffic after posting an identical post to test web app launch to three different social media channels on one day during December 9th resulted in Facebook becoming the most effective traffic source.

The results from the quantitative studies could not be taken into too much consideration at this very early stage since the sample size was very small. This is closely related to the issues found with qualitative studies for signing up and due to time constraints the corrections could not be made before the course ended on December 16th. The conclusion only confirms that the Minimum Viable Product was not tested for functionality, usability and reliability before taken in front of some potential users. However, the qualitative usability tests revealed the issues with little effort and gave good insights for improvement.

4.8 Overview of the Project schedule

Phase	Days (estimated)	Days (spent)	Days (left)	1	2	3	4	5	6	7	8	9	10	11	12
Envision: Estimate & Plan project	7	7	0	7											
Concept: User & Domain research, technical exploration	7	11	0		7	4									
Project Framing: Product Direction, Mockups, Wireframes, Design	10	12	0			7	5								
Development environment setup for the team (Version Control, Server, Localhost, Frameworks)	7	20	0				7	7	6						
Sprint 1: First version of the site (HTML / CSS / Javascript / PHP)	14	14	0					7	7						
Sprint 2: Database, BackEnd & FrontEnd integration	10	15	0						7	7	1				
Usability Testing 1 (Among Team)	1	1	0							1					
Bug fixes, optimization	2	1	1								1				
Content	7	2	5									2		-	
Usability testing 2 (MVP & Real users)	7	1	6										1		
Measure & Learn from feedback	2	0	2											0	
2nd version of product	10	0	10												0
TOTAL DAYS	84	84	24	7	7	11	12	14	20	8	2	2	1	0	0
Weekly Burnout	30			5	4	5	4	5	4	5	4	5	4	4	5
Total time left (from estimate)			84	79	75	70	66	61	57	52	48	43	39	35	30
Total time left (from spent)			84	77	70	59	47	33	13	5	3	1	0	0	0

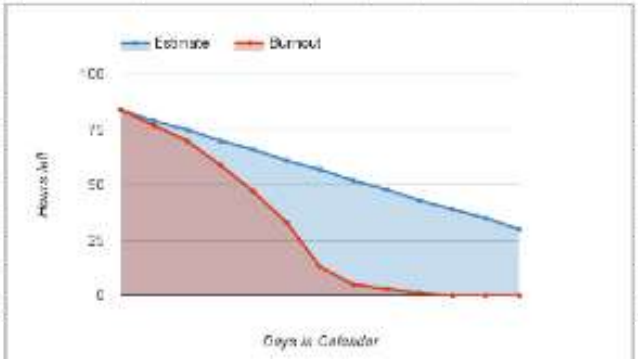


Figure 36. Opas project schedule in days; estimate and time spent (Developed by the author)

The project schedule was used to manage the progress in product experiment 2, Opas. It shows the challenges of the project in terms of staying in schedule. The project suffered greatly from compatibility issues in having to use two different operating systems (Windows and Mac OS X). Setting up version control (Git) and the backend framework (Laravel) took a lot longer than was planned for as well as Integration of database, backend and frontend. According to these rough estimates the project would have needed and additional 24 days to complete an improved 2nd version of the product to complete the user stories that were the project goals.

5. Results

No two projects are the same. However, the two, very different experiments did offer enough experience for a comparison. The primary criteria for me was to find out how to minimize the amount of resources spent to give a maximum amount of learning with a goal of ultimately achieving validation or invalidation of the idea. A secondary criteria for me was to find out if the resources that were reasonably available would be enough to build something that people want. The most important resource in this case were skills and time. Bootstrapping, without even considering paid options was the default way of proceeding, to the exception of buying what was considered mandatory, such as domain names.

The market need proved to be more effortlessly invalidated with the process in experiment 1, i.e. the Unhidden project. It quickly showed that the vast majority of the assumed early adopters were not too enthusiastic about the idea thus could not become much needed early evangelists of the early product that would promote the product forward. More so, the app would have required complex features for the business logic to work such as a separate back end or even a content management system. With common reasoning, it became all too obvious to not spend time to start transferring the solution into an MVP simply by engaging in a qualitative discussion about the problem though Facebook groups and forums with the thought early adopters. This eliminated a whole lot of potential wasted time that would have otherwise been spent on a project without an initial market need.

On the other hand, Steve Blank (2014) mentions that no business plan survives first contact with the customers. It could also be considered too early to cancel the Unhidden project because of the aspect of a big enough problem not found right away after a first contact with the assumed early adopters. This could have been further tested by changing the segment of assumed early adopters and by running the landing page test again. By having to switch the customer segments to find out which customer segment would need the solution it made it more obvious that it was only a solution looking for a problem, rather than the other way.
Consequently, the main reason for the cancellation of the first project became that it originally was an idea for the sake of coming up with a Startup idea and trying out a solution, rather than aiming to solve a real problem that I would personally care enough

about to carry it on for the next 5 years or so. Combined to high technical complexity for an MVP it simply did not seem reasonable to execute. In this case the Lean Startup methodology helped somewhat with the reasoning to move on to another project rather than to persevere and not feel too regretful about cancelling the project.

Even though, Experiment 1 fulfilled the criteria of going as quickly as possible in building a demo and getting feedback from assumed customers proved to validate or invalidate the idea more efficiently, the end-results from user feedback were more promising in Experiment 2 to further continue developing the idea.

The strongest argument on behalf of agile and iterative methodologies used in the thesis was the ability to experiment with the approach itself, creating suitable processes for the situation at hand. Another major advantage showed itself as providing a unified iterative workflow for all the major aspects in developing the foundations for an internet Startup. Particularly so, in getting also the business side and logic to join in on becoming an iterative process rather than having it all perfectly figured out beforehand as the traditional business plans would require. The design community has worked in iterative and incremental processes for decades now and the agile manifesto for software development was created already in 2001 (Beck & al.) Recently, with the help from the popularity of the lean Startup movement, an iterative approach has also reached business development offering a compatible workflow with the design and developer aspects allowing complimentary workflows.

The Lean Startup by Eric Ries claims that the lean Startup methodology offers a scientific approach of creating and managing a successful Startup (Ries 2012). That is a bold statement and for me, it suggests that a Startup strictly following the methodology would lead to success. However, following the methodology revealed to me that the lean Startup methodology is very much focused mostly on perfecting the development process to build a perfect product. The methodology simply does not take nearly enough into account other aspects in entrepreneurship besides product development to be considered a guide to Startup success. This also leads to my biggest criticism against the lean Startup. Its customer centric view suits well for today's challenges but not tomorrow's highly innovative solutions or new inventions that would create new markets.

Judging from even a small amount of users part-taking in the usability testing, showed that it is very hard for the user to foresee next big leaps or request innovative features. The harsh reality was that they seemed to mostly want features that were familiar from other similar products and for a cheaper price. The usability testing also revealed that most of the feedback given by the users mostly focused on minor issues in web site design rather than functionality of the app or did not offer anything yet to improve the logic of the business idea.

I also found the theory of the methodology by Eric Ries to leave too much room for interpretations, such as not providing any real examples for the innovation accounting measurement method mentioned, where now the pirate metrics by Dave McClure has somewhat become the default in the Startup world.

The challenges with agile software development were found to be caused mostly due to human factors in adopting the principles, rather than constitutional issues of the principles. This was likely caused due to agile projects being highly fluid, making them more difficult to manage, giving more responsibility for decision making for the Development team, hence requiring a more experienced team to gain progress compared to the waterfall approach. The original team size of 12 students in the development team also proved all too large for an agile project and created excessive coordination issues. The coordination issues transformed into pressure on the product owners to change towards a traditional waterfall approach rather than rely on team self-organizing to produce working software. The following were the main observations related specifically to improving success in agile projects from the Opas project.

1. Success in Agile projects is dependent on experienced team members.

Agile projects involve a lot more flexibility to make decisions to the Development team and involve constantly changing needs compared to the waterfall approach. It also demands a lot more and without experienced members the much needed measuring feedback and constant planning and predicting is very difficult. Many aspects in agile projects seemed to be tailored to rely on the technical expertise of the Development team and their adaptability to work together. Overcoming technical uncertainty for the project as well as uncertain requirements required a lot of added time for experimenting and further research.

2. Forming a cross-functional and self-organizing team does not come naturally.

The Opas project consisted of a collaboration with a relatively large team of 12 IT students with a wide range of backgrounds and skills. To make coordination easier, the Development Team divided into subgroups according to their expertise and preferences of Database, Backend and Frontend. This proved to be a management mistake and prevented the project to be truly agile later on, since the mindset of the development team already formed heavily to functional departments that only did their specified part of the tasks rather than completing feature that required crossing borders between Backend, Frontend and Database.

3. Integrated, continuous testing takes a lot of effort

Maintaining extensive testing practices after changes made to the code, proved to be extremely time consuming and was probably not automated enough to ease the burden for the developers making changes in the production codebase. Any reasonable software solutions for automated testing were found to be very expensive. Reaching to a definition of what is DONE early on is a common pitfall in agile projects. This proved to be the case the Opas project as well, even if it was anticipated for and defined. The main reason behind this issue seemed simply to be the amount of time and energy needed for continuous testing when pushing new bits of code to the production version. Constantly being behind the project schedule also did not allow enough testing.

5.1 Ethical viewpoints

I believe a major influence for today's general business ethics was formed during the recession in the 1970s by one influential economist named Milton Friedman, who also received a Nobel Prize in economics 1976. He can be tracked back to a publication in the New York Times that became a popular phrase for many decades: "the social responsibility of business is to increase its profits" (Friedman 1970). This could very well be the single most powerful idea that could be tracked and that has had the biggest impact to prioritizing profit over ethical viewpoints in business and social responsibility for companies in the recent history of market economics.

Even though, the idea of the ultimate purpose of business being to raise stakeholder value is now largely agreed only to be a result, not a strategy, some estimates state that this school of thought will continue to persist in major businesses and business schools until 2020 (Denning 2013)

Another ethical school of thought for businesses, represented initially by Peter Drucker, presents a different view: "The purpose of business is to create a customer" (Drucker, 1954, 62). This view seems to be the more accurate to how lean and agile Startups form their strategy and has the potential of driving even more profit in the long run when a complete and sustainable business model is created based on the customer needs rather than focusing merely driving of profit that could be considered short sighted. Peter Drucker's definition seems like it was ahead of its time, because globalization and the Internet needed to happen before consumers could very quickly and conveniently choose between other providers if the product or service does not fulfil their needs.

The Lean Startup Methodology succeeds in capturing the purpose of a business in this view, by perfecting the product through experiments and constantly testing how much value the idea/solution/product has the customer. The learning is used to direct future investments only on validated aspects and therefore concentrating on keeping the customer happy for the long run and the Startup behind the product more ethical than merely focusing on making profit.

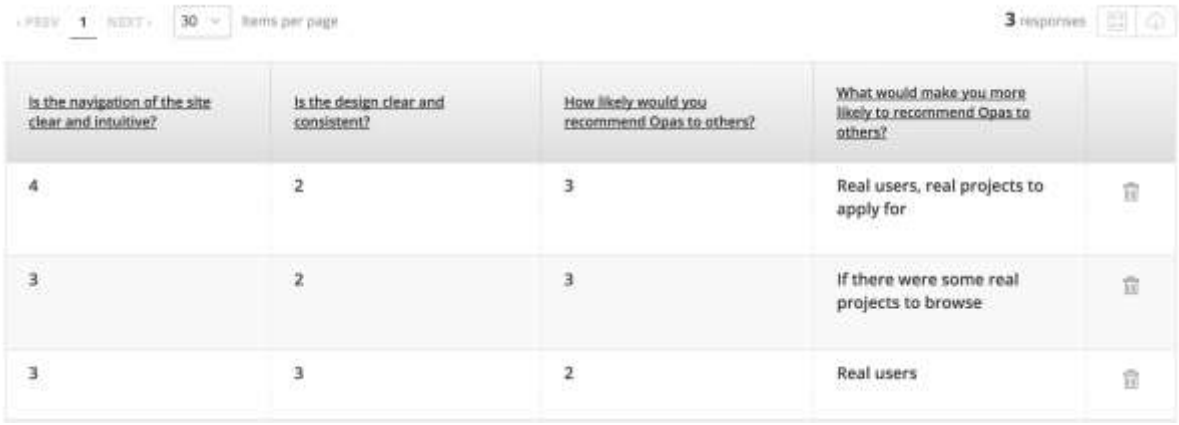
The lean Startup methodology does also have some of its own inbuilt ethical viewpoints that needs to be recognized early on when validating the problem/solution fit. One highly popular experiment to test customer demand related to the Lean Startup Methodology is the landing page test or the so called "smoke test" that was also carried out in experiment 1 and also as slightly modified in experiment 2. For internet/mobile Startups, the intention of the landing page test is to find out quickly if there is demand for the product by setting up a webpage and asking for some sort of payment, i.e. in the form of money or personal data, such as an e-mail address. As this has somewhat become the norm to test initial interest, it also involves selling a statement to the customer that the product is very much under development even if it's more likely that the landing page consists of simply fake screenshots, fake user activity and fake user testimonials and not a single line of code has been written yet.

A common phrase the web Startup community: "fake it until you make it" somewhat describes this activity. This seems to have become the typical path, especially when there's two sided markets involved where the content needs to exist before it can provide value to the user. As an example, the founders of the social news networking site Reddit, populated in the initial stages the site's content with a large amount of content from the founders own accounts with different names (Mead, 2012). For social sites, this creates the opportunity to set the initial tone in of the site and make it feel alive from day 1. The founders of Reddit managed to build a whole community around their own fake accounts without getting caught and created an initial momentum for the site to make it look lively. Without this activity, the site would most likely not have provided enough value for the randomly visiting user to be interested about the product.

With the second product experiment of Opas, this ethical question of having fake users popped up during the first user testing. On one day, we did two types of user testing, one was on location user testing and the other one consisted of remote testing using Hotjar screen recordings together with Google Analytics. Turns out the only 3 comments that were left by the remote testers were about apparent frustration in anticipating real users and real projects for the site rather than fake ones.

The fake users were seeded purely for development purposes to test out that the site worked as should and this was explained in the face to face user testing but not for the remote testers.

We made the mistake of leaving the fake users still on the site as we did not think it would raise any severe issues if the fake users were among the real ones on the site during the remote usability testing. Apparently though, based on the comments below, the presence of fake users seemed to be the only thing worth commenting about for the real users that expected a working product and greatly affecting the value of the product. Clearly, that made the MVP of Opas to not reach a level of reliability that was assumed by the user and therefore the whole product was not considered to be reliable enough by the users.



« PREV 1 NEXT » 30 Items per page 3 responses

Is the navigation of the site clear and intuitive?	Is the design clear and consistent?	How likely would you recommend Opas to others?	What would make you more likely to recommend Opas to others?	
4	2	3	Real users, real projects to apply for	
3	2	3	If there were some real projects to browse	
3	3	2	Real users	

Figure 37. Net promoter score (2,66 out of 10) and Feedback from remote users. (Developed by the author)

5.2 Conclusions

One major objective of this thesis was to reach a structure of testing business ideas for a market need. The thesis focused on combining frameworks from the following aspects: business development, software development and design aspects for the early stages in web and mobile based entrepreneurship. This was reached with the second experiment that was also tested for initial market need. The results that followed proved that there is somewhat a market need for the product of Opas, but the problem remains still too fragmented for a start and needs to be further refined for the second iteration of the business model as well as the MVP.

The study did not directly result in a scientific or optimal way to validate an idea for a web or mobile Startup. It did however provide a push toward that direction and showed that finding an optimal way is more dependent of properly understanding the context of the idea and then finding the right tools and approaches for each situation rather than following a single methodology such as the lean Startup as a process for success. In a nutshell, judging from only two experiments, the simplest and most effective structure of validating an idea for a web or mobile based product was found to be setting up a web site, a landing page that describes the product or service with a sign up form. To direct traffic on the site, which consisted of assumed early adopters, social media channels such as Facebook groups provided enough directed traffic for the site to test the conversion rate, combined with some organic traffic from Google.

The Lean Startup methodology is somewhat easy to understand, but difficult to master and most likely requires years of practice, especially for an unexperienced entrepreneur. It did show its value as a flexible framework especially in web / mobile Startups, more specifically for in the Startups' product development. It also showed some of its limitations; there needs to exist a clearly identifiable current market that could divided into segments that can be used as a testing ground for a demo or a MVP. Also, the methodology is easier to start with if the type of the idea is something that can be easily turned into an MVP. Consequently, Startup ideas that have a high cost in failed iterations such as many related to the pharmaceutical industry, healthcare and financial industries can

be expected to have difficulties in accepting the Lean Startup approach in using learning from experimenting as a basic structure. The two experiments were found suitable for a lean approach since the products consisted of affordable web hosting and code that was easy change for the next iteration after the issues were spotted by test users. As a result, the study suggests that Startup founder's definitely should research first which approach is suitable for the idea and the product and try to avoid being biased by the popularity and hype by consultants of the Lean Startup or Agile Software development approaches.

My biggest criticism against the lean Startup methodology lies in Startup ideas that are based on predicted needs that are emerging from new inventions and technological advancements. In this category falls the ideas that rely on the founders vision about new, emerging markets and they can be really hard to validate using the methodology since the starting point in validating a problem requires that it needs to be present today rather than tomorrow before a solution can be validated. Instead, the Lean Startup methodology does provide a good process for perfecting product development and reducing resources spent that do not offer additional value for the customer.

In the fast changing technology industry that is known for quickly jumping in on the bandwagon of the next hyped thing, relying on this aspect of validation creates a wide range of lost opportunities for entrepreneurs and in my opinion directs entrepreneurs to re-invent the wheel businesses and copycat other success stories simply by reducing costs and added features when the aim is in reducing risks by validation. Solutions, that are meant to be started as preparing to solve future problems, that could be somewhat predicted by a visionary or inventive entrepreneur and do not yet find a market for an existing problem are really hard to validate using the lean Startup methodology.

To conclude, the Lean Startup does not by itself provide enough range for successful business development as stated by Eric Ries, but succeeds in offering a good framework for technology Startups for fine tuning the product or service by finding out current customer needs. Consequently, those that practice lean Startup seems to have divided roughly into two camps. The first group use it as a step-by-step process to follow

for building a Startup and the other as a set of tools for specific purposes when needed. The two experiments changed my opinion to be very much in favour of the latter. I believe that successful entrepreneurship consists of a great deal of different sciences and methodologies used so creatively that it could be considered art, combined with preparing for certain opportunities that requires a great deal of insight that can be used for the factor of right timing. A methodology such as the lean Startup revealed itself to be one toolbox among many that are needed by an entrepreneur and not always offering the best tool for every situation. Specifically, for a project consisting of inexperienced team members, being lean can prove to be too flexible for reaching optimal progress, especially when team members are still learning the about the development or technical aspects.

After the experiments, any scientific methods of reaching Startup success still remains a bit less complex and interdependent, in my view being a subject of slightly improved guesses drawn from experience, timing and being prepared for opportunities rather than an exact science that could be used to build a repeatable path to Startup success. What was left after studying the Lean Startup methodology and conducting two experiments is a useful tool set of methods to use in confirming that the product is on the right track once the right customers are found and not moving forward before it is confirmed.

As an aim to decrease the amount of art of entrepreneurship in the start, Ash Maurya introduces a more practical way with a step-by-step guide how to start with a set of scripted interviews to start validating first if the problem exists, by using qualitative interviews to test if the assumptions for a business model is correct. This can be harder than it seems at first as it can lead to a never-ending path of wasted time conducting more and more complicated interviews. It proved to give significantly more useful responses when presenting the idea for the assumed user in the form of having something tangible built or demoed that could be called the MVP. Even if only consisted of mock-up screens to demo the solution for the assumed customers, the difference in gaining more accurate feedback proved to be crucial. This was especially true for web

and mobile products, where the MVP is possible to be built very rapidly and a wide range of prototyping opportunities exist for free.

Psychologically, becoming a founder for a lean Startup seems to heavily demand having a mindset of focusing on constant observation and learning that should overcome the inevitable ego trips caused by failure or success in the experiments when turning a business ideas into a profitable companies. The methodologies of Lean Startup, Agile Engineering and User Experience Design are all more or less focused on understanding much of the customer worldview first and then provide value in the form of more effective solutions. Learning about what people actually think about the product involves a lot of interaction with the assumed customers' right from the start, and later often so with a product that has its early flaws. It proved not to be anywhere near a natural process for the Development Team that participated in the second experiment. For many, the reason was most likely pride in their own work and unwillingness to present a product that is not perfected before getting feedback about it. However, the findings from real users testing the early product and giving feedback only fully confirmed that real users get a completely different type of User Experience and usability issues than the developer who has stared at the more complex details of same application for the last months.

The methodologies such as the Lean Startup approach offer good focus points early on for the Startup and its team by focusing to gain facts about actual progress in business development from the perspective of the value provided to the customer rather than in terms of profit, funding received or founder ego boosts or knockdowns. In the early stages, the ability to experiment, listen, observe to find a clearly defined problem as well as the ability to be able to make changes based on facts from feedback might very be the most important factors leading to success later on. This is by no means an easy process and definitely takes more than two experiments to be good at it.

6. Evaluation of the thesis process

After the two experiments the most rewarding aspect became the amount of personal development and becoming more able to understand deeper patterns in development processes as well as human behaviour of the Development team as well as the Users of the product, nevertheless needed to build a successful Startup or to be a successful entrepreneur.

The most useful aspect I learned from the thesis process was the mindset of getting things done in slight improvements by prototyping and continuous iterations rather than attempting to produce something perfect right away. A passion for strategical thinking and problem solving became the key driver for progress and the ability to clearly define problems became a must to find the right channels for answers and insights. The thesis process also forced to become more efficient in self-educating by using online resources such as Massive Online Open Courses (MOOC) or Question & Answer sites such as Stack Overflow, StackExchange or Quora. The ability to network with the people who had answers and insight was also of great help for the thesis process. To self-educate efficiently for solving problems proved to be the most needed aspect to reach results and progress. The thesis process also verified to me personally that something as complex as building a Startup that involves a lot of different sciences and numerous changing and interdependent variables, a more hands on approach using experimentation as a primary tool for learning provided a much more rewarding way of learning and gaining actual progress in personal development than getting immersed into studying theories, methodologies and best practices.

The material for the thesis consisted from a timeframe of 11 months and consisted of numerous updates in the mindset of optimizing the path from an idea to finding a market need. Looking back on the early material in the first experiment now recognizes several beginner mistakes and I am assured that time will also change some of the findings presented in this thesis as more learning takes place. Ultimately, the thesis process helped to understand the value of continuous learning.

7. List of references

24/7Wall St. 2009. The Biggest 10 tech failures of the last decade. URL: http://content.time.com/time/specials/packages/article/0,28804,1898610_1898625_1898640,00.html. Accessed: 15 January 2015.

Amit, R., Glosten, L., & Muller, E. 1993. Challenges to Theory Development in Entrepreneurship research. Journal of Management Studies. URL: http://onlinelibrary.wiley.com/doi/10.1111/j.1467-6486.1993.tb00327.x/pdf. Published 5 May 2007. Accessed: 22 September 2015.

Bank, C. 2014. Minimum Viable Products – Defined by the experts. URL: http://www.onextrapixel.com/2014/10/13/minimum-viable-products-defined-by-the-experts/. Accessed 15 January 2016.

Bard J. 2014. UX Design Defined. URL: http://uxdesign.com/ux-defined. Accessed: 14 January 2016.

Beck, K., Beedle, M., Bennekum, A., Cockburn, A., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R., Mellor, S., Schwaber, K., Sutherland, J., Thomas, D. 2001. The agile manifesto. URL: http://www.agilemanifesto.org/. Accessed: 14 February 2016.

Blagojevic, V. 2013. The ultimate guide to minimum viable products. URL: http://scalemybusiness.com/the-ultimate-guide-to-minimum-viable-products/. Accessed 16 July 2015.

Blank, S. & Dorff, B. 2012. The Startup owner's manual. The step-by-step guide for building a great company. K&S Ranch. Pescadero.

Blank, S. 2013. Why the Lean Startup changes everything. Harvard Business Review. URL: https://hbr.org/2013/05/why-the-lean-start-up-changes-everything. Accessed 18 December 2015.

Blank, S. 2014. Driving Corporate Innovation: Design Thinking vs. Customer Development. URL: http://steveblank.com/2014/07/30/driving-corporate-innovation-design-thinking-customer-development/. Accessed: 14 January 2016.

Canvanizer. 2014. Business Model Canvas vs. Lean Canvas. URL: https://canvanizer.com/how-to-use/business-model-canvas-vs-lean-canvas. Accessed: 20 December 2015.

CBInsights. 2014. Top reasons for Startup failure. URL: https://www.cbinsights.com/blog/Startup-failure-reasons-top/ Accessed: 20 November 2015.

Cummings, M. 2014. UX Design – Humanizing interaction. URL: http://uxdesign.com/ux-defined-2. Accessed: 14 January 2016.

Denning, S. 2013. Forbes online publication. URL: http://www.forbes.com/sites/stevedenning/2013/07/05/the-overunder-on-when-the-worlds-dumbest-idea-will-die/#5a12c1492ba5. Accessed: 5 February.

Drucker, P. 1954. The practice of management. Harper Business. New York.

Edmondson A. 2011. Harvard Business Review. URL: https://hbr.org/2011/04/strategies-for-learning-from-failure. Accessed 5 August.

Friedman, M. 1970. New York Times. The social responsibility of business is to increase its profits. New York.

Gates, William H. 1995. The Road Ahead. Penguin books. New York.

Google Trends. 2015. Comparison of three search keywords. https://www.google.com/trends/explore#q=lean%2520Startup%252C%2520design%2520thinking%252C%2520customer%2520development&cmpt=q&tz=Etc%252FGMT-2. Accessed: 20 December 2015.

Graham, P. 2012. How to get Startup ideas. URL: http://paulgraham.com/Startupideas.html. Accessed: 16 March 2016.

Gross, B. 2015. The single biggest reason why Startups succeed. TED Talk. URL: http://www.ted.com/talks/bill_gross_the_single_biggest_reason_why_Startups_succeed. Accessed: 25 January 2016.

Highsmith J. 2010. Agile project management: Creating innovative products. 2nd ed. Pearson Education. Boston.

Kawasaki, G. 2012. Tips for Start Up Success: The Art of Community Building. Slideshare presentation. URL: http://www.slideshare.net/GuyKawasaki/tips-for-start-up-success-the-art-of-community-building/7. Accessed: 15 January 2016.

Marmer, M., Herrmann, B., Dogrultan, E., Berman, R. 2011. The Startup Genome report. URL: https://s3.amazonaws.com/Startupcompass-public/StartupGenomeReport2_Why_Startups_Fail_v2.pdf. Accessed: 20 August 2015

Maurya, A. 2012. Running Lean. Iterate from plan A to a plan that works. O'Reilly. Sebastopol.

McClure, D. 2007. URL: http://www.slideshare.net/dmc500hats/Startup-metrics-for-pirates-long-version. Accessed 23 May 2015.

McClure, D. 2013. Quora post. What is the proper definition of a Startup? URL: https://www.quora.com/What-is-the-proper-definition-of-a-Startup. Accessed 15 January 2016.

McGinn, D. 2012. Interviewed by Sarah Green. Harvard Business Review. URL: https://hbr.org/2012/08/whats-wrong-with-todays-entrep/. Accessed: 22 November 2015.

Mead, D. 2012. Motherboard. How Reddit Got Huge: Tons of Fake Accounts. URL: http://motherboard.vice.com/read/how-reddit-got-huge-tons-of-fake-accounts--2. Accessed: 10 February 2016.

Mullins, J. & Komisar, R. 2009. Getting to plan B. Harvard Business Review Press. Boston.

Norman, D. & Nielsen, J. 2014. The Definition of User Experience. URL: https://www.nngroup.com/articles/definition-user-experience/. Accessed: 15 January 2016.

Osterwalder, A. & Pigneur, Y. 2010. Business Model Generation. Wiley. New Jersey

Pasanen J. 2015. # 'How to start small' https://medium.com/@atroyn/how-to-start-small-efe1bf831aaf … by @atroyn /HT @ggonweb thanks. Tweet @jopas. URL: https://twitter.com/jopas/status/551188242934935553. Accessed 10 August 2015.

Ries, E. 2009. Minimum Viable Product: a guide. URL: http://www.Startuples-sonslearned.com/2009/08/minimum-viable-product-guide.html. Accessed 12 January 2016.

Ries, E. 2009. Slideshare presentation. URL: http://www.slideshare.net/Startuples-sonslearned/2009-05-01-how-to-build-a-lean-Startup-step-by-step/10-TradiJonal-ProductDevelopment_UnitofprogressAdvancetoNextStage_Waterfall_Require-ments_Design. Accessed 17 September 2015.

Ries, E. 2011. The Lean Startup 2011. Crown Business. New York.

Rämö, K. 2014. Taloussanomat. URL: http://www.taloussanomat.fi/yrityk-set/2014/11/07/talvivaara-on-Startup-onko-tassa-jarkea/201415536/12. Accessed 12 July 2015.

Schumpeter, Joseph A. 1934. The Theory of Economic Development: An Inquiry into Profits, Capital, Credit, Interest, and the Business Cycle. Cambridge.

Schwaber, K. & Sutherland, J. 2013. The Scrum Guide. URL: http://www.scrum-guides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf#zoom=100. Accessed: 21 January 2016.

Small Business Encyclopedia. 2016. URL: http://www.entrepreneur.com/encyclopedia/bootstrapping. Accessed: 15 March 2016.

Statistics Finland 2014. New enterprises. URL: www.stat.fi/til/aly/2014/aly_2014_2015-10-29_tie_001_en.html. Accessed: 19 September 2015.

The great Startup wiki 2014. Web Startup. URL: http://thegreatStartupwiki.com/web-Startups/. Accessed: 24 February 2016.

U.S Department of Health & Human Services 2016. Digital Communications Division. URL: http://www.usability.gov/how-to-and-tools/methods/recruiting-usability-test-participants.html. Accessed: 12 January 2016.

UXMastery. 2015. UX techniques. URL: http://uxmastery.com/resources/techniques. Accessed 18 January 2016.

VersionOne. 2015. State of agile survey. URL: https://www.ver-sionone.com/pdf/state-of-agile-development-survey-ninth.pdf. Accessed 20 January 2016.

Wikipedia. 2016. Agile Software Development. URL: https://en.wikipedia.org/wiki/Agile_software_development. Accessed: 24 February 2016.

Wikipedia. 2016. A/B testing. URL: https://en.wikipedia.org/wiki/A/B_testing. Accessed 15 March 2016.

Wikipedia. 2016. Cohort analysis. URL: https://en.wikipedia.org/wiki/Cohort_analysis. Accessed 15 March 2016.

Wikipedia. 2016. Conversion funnel. URL: https://en.wikipedia.org/wiki/Conversion_funnel. Accessed 15 March 2016.

Wikipedia. 2016. Net promoter score. URL: https://en.wikipedia.org/wiki/Net_Promoter. Accessed 15 March 2016.

Wikipedia. 2016. Usability testing. URL: https://en.wikipedia.org/wiki/Usability_testing. Accessed 15 March 2016.

Wikipedia. 2016. UX – User Experience. URL: https://en.wikipedia.org/wiki/User_experience. Accessed 15 March 2016.

Wikipedia. 2016. SaaS – Software as a Service. URL: https://en.wikipedia.org/wiki/Software_as_a_service. Accessed: 20 February 2016.

Wilcox, J. 2014. The sales pitch is dead. Long Live Solution Interviews! URL: http://customerdevlabs.com/2014/08/05/problem-solution-interviews-b2b-sales-pitch/. Accessed: 15 February 2016.

8. Attachments

8.1. User interview

1. What is the hardest part about finding people that could provide help within your organization?
2. Can you tell a bit about the last time this happened?
3. What made it difficult?
4. Have you done anything to solve that problem?
5. What don't you like about the solutions you tried?

Applied from Wilcox (2014)(Developed by the author)

8.2 Usability checklist for Opas

- Navigation
    - Is the navigation clear and intuitive?
    - Are there any unnecessary levels or all the pages are simple?
- Functionality
    - Does the functionality meet your expectations?
    - Something missing/unnecessary features?
- Control
    - Are there all the necessary controls: add, update, delete, cancel, etc.?
- Consistency and design
    - Did you pay attention to design/colours (if not - perfect)?
    - Were you confused with buttons names/icons meaning?
- How likely you would recommend this to others?
    - rate (1 - 5) where '1' is highly unlikely and '5' is very likely
- What would make you more likely to recommend OPAS to others?

(Pasanen, Vankalo, Volkova, 2015)

8.3. Database plan for Opas

14.10.2015	Get general feedback and ideas Get a list of use cases Values domain Friends/connections table etc. Values types (raw - name, categorical - gender, identifier - id, relational - FK) Strong and weak entities (FK position) Candidate keys, primary keys, alternate keys Check redundancy – redundant relations, redundant entities Design derived attributes (total profile number) Check if current entities and attributes provide all the information for every use case Document everything
21.10.2015	Normalization up to 3NF Validate relations with normalization Integrity constraints – null or not null data, domain (allowed values), multiplicity (one to one, many to one, many to many), FK constraints (can they be null, what happens when inserting, updating and deleting both parent and child tables) General constraints (business rules) User rights Check if current design has growth potential. Check if current design can be easily updated during further product iterations Document everything
28.10.2015	Decide upon dbms and db engine (mariaDB and InnoDB) Estimate system scale, disk space requirements Implement relations, constraints (create table statements) Implement sample data (insert statements)

	Indexes? File organizations? Even bother? User views design (what data each user sees and updates) Concurrency control? (many transactions at the same time) Document everything
04.11.2015	Do transaction queries according to views design Check queries correctness (is semantics correct, are correct joins used, etc) Discuss security measures (system security and data security) Analyze threats - theft and fraud, loss of confidentiality (secrecy), loss of privacy, loss of integrity, and loss of availability. What security controls we need based on threats? (authorization, views, encryption, raid backup, etc) Discuss backup options Test first prototype version
11.11.2015	Add credentials
18.11.2015	Backend merging
25.11.2015	
02.12.2015	
09.12.2015	Final testing and identifying missed milestones Validating against requirements
16.12.2015	Database administration and support options Scale of the future project, is relational db the right choice Further goals and targets Know performance issues and improvements suggestions Measure transaction throughput, response time (how to improve), disk storage requirements (how to reduce) Analyze current and optimal system resources (server hardware) Final documentation

8.3.1. Database table for Opas

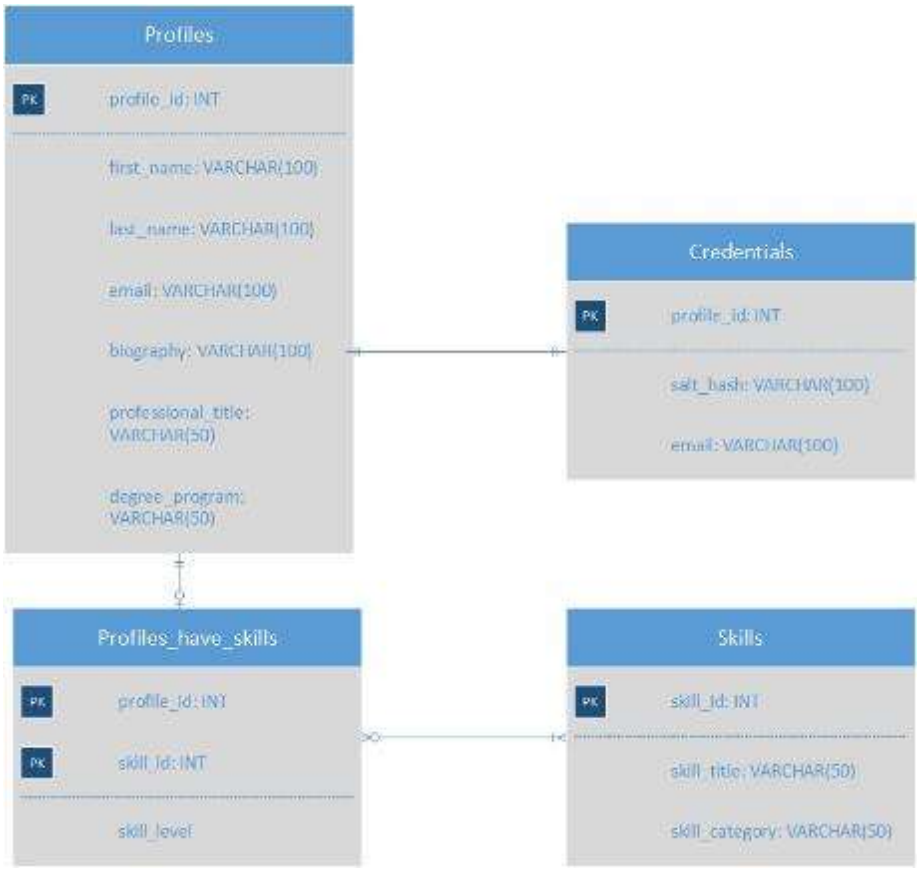


Table 1. Database table. (Berclaz, Dubuis, Ivanov, Volkova 2015)

Table descriptions

Table name	Description	Occurrence	Medium / Max amount	Volatility
Credentials	*Logins and passwords*	*Each user registers and uses his unique credentials to access the system*	*5/ 1000*	*+/- 5-10 a month*
Skills	*Any studies-related subject/ language, etc. Skill that a user might potentially have*	*Predefined in DB for using as a drop-down for FK or for search by skill option*	*10 / 20*	• *(fixed)*
Profiles	*Basic information about a user (student, teacher, alumni or staff)*	*New profiles/ members are added any time, approximately 10 more each month*	*5 / 1000*	*+/- 5-10 a month*
Profiles_have_skills	*The skills that refer to a particular profile*	*The user chooses the skills that he has*	*1 / 20*	*+/- 5-10 a week (depending on profiles)*

Table 1.1 Data Dictionary. Table descriptions. (Berclaz, Dubuis, Ivanov, Volkova 2015)

Column descriptions with the data types of the chosen technical storage (Relational Database, MS SQL Server as the DBMS):

Table name	Column	Description	Data type and Length	NULL / UNIQUE	Primary / Alternate key	Special value domain?
Profiles	*profile_id*	*(foreign ke =>)*	*=> Credentia (profileId)*	*NOT NULL*	*PK*	*(foreign key =>)*
	first_name	*first name of a user*	*VARCHAR(100)*	*NOT NULL*	-	-
	last_name	*last name of a user*	*VARCHAR(100)*	*NOT NULL*	-	-
	email	*(foreign ke =>)*	*=> Credentia (email)*	*NOT NULL*	-	*(foreign key =>)*
	biography	*comments*	*VARCHAR(100)*	*NULL*	*-tra*	-
	professional_titl	*job title if any*	*VARCHAR(50)*	*NULL*	-	-
	degree_program	*name of th program*	*VARCHAR(50)*	*NOT NULL*	-	-
Skills	*skill_id*	*(Surrogate key)*	*INTEGER*	*NOT NULL*	*PK*	*100-999999*
	skill_title	*Name of the skill. E.g. "French language"*	*VARCHAR(50)*	*NOT NULL*	*AK*	-
	skill_categories	*name of th category*	*VARCHAR(50)*	*NOT NULL*	-	-

Pro-files_have_skills	*profile_id*	*(foreign ke =>)*	*=> Profiles (profile_id)*	*NOT NULL*	*-*	*(foreign key =>)*
	skill_id	*(foreign ke =>)*	*=> Skills (skill_id)*	*NOT NULL*	*-*	*(foreign key =>)*
	skill_level	*rank of th skill from lowest to hishest*	*SMALLINT*	*NOT NULL*	*-*	*1-5*
Credentials	*email*	*-*	*VAR-CHAR(100)*	*NOT NULL*	*AK*	*ASP.NET Email vali-dator regula expression*
	salt_hash	*Generated salt value for pass-word, and hash for salt+pass-word strin*	*VAR-CHAR(100)*	*NOT NULL*	*UNIQUE KEY*	*-*
	profile_id	*(Surrogate key)*	*INTEGER*	*NOT NULL*	*PK*	*10-999999*

Table 1.2 Data Dictionary. Column descriptions. (Berclaz, Dubuis, Ivanov, Volkova 2015)

8.5. Permit to use course output in thesis: Software Engineering Project SYS4TF222-1

Attachment 1.

HAAGA-HELIA
University of Applied Sciences

Permit to Use Course Outputs in Thesis

In HAAGA-HELIA a student may write a portfolio thesis consisting of prior course products or assignments. This form aims to ensure that permits are given by students co-authoring or co-producing a course product whatever the format. The permit covers modifying, editing, and using the course outputs in the thesis and related publications. The permit also applies to transferring the course outputs to a third party.

STUDENT INFORMATION: STUDENT PLANNING PORTFOLIO THESIS

First and last name	Mikael Vankalo
Degree programme	Experience and Wellness Management
Campus	Haaga campus
Email address	mikael.vankalo@myy.haaga-helia.fi

STUDENTS ISSUING PERMISSION TO UTILIZE CO-AUTHORED COURSE PRODUCTS

Date	Name	Signature
02.12.2015	Pavel Ivanov	
2.12.2015	Jaana Ahonen	
2.12.2015	AshkanYaldaie	Ashkan
2.12.2015	Victor Stepanchikov	
02.12.2015	Anastasia Volkova	
2.12.2015	Mari Pasanen	
02.12.2015	Antoine Dubuis	
02.12.2015	Valentin Berclaz	
02.12.2015	Vincent Schiré	

Figure 38. Permit to use course output from Software Engineering Project Team SYS4TF222-1.