

Lean Product Development in Early Stage Startups

Jens Björk, Jens Ljungblad and Jan Bosch

Software Engineering Division
Chalmers University of Technology
Göteborg, Sweden
[jensbjljenslj@student.chalmers.se, jan.bosch@chalmers.se]

Abstract. Software startups are more popular than ever and growing in numbers. They operate under conditions of extreme uncertainty and face plenty of challenges, underlined by their high failure rate. Using Design Science Research, these challenges were investigated. A literature study showed that in recent years, several authors have suggested ways to increase the odds of succeeding as a startup, such as customer focused development, fact based decision making, pivoting and agile/lean thinking. Interviews with industry professionals showed that few used these Lean Startup practices: many found them difficult to implement in practice. In response, we developed the Early Stage Software Startup Development Model (ESSSDM) for managing early stage software startups by applying Lean Startup principles. The model is novel in that it supports managing a portfolio of product ideas and provides clear criteria for when to move forward with product ideas, when to abandon product ideas as well as recommends what concrete techniques that can be used and when, in order to achieve this. The process was instantiated and evaluated on a startup project in an incubator setting.

Keywords: startup companies, lean product development, software process.

1 Introduction

New software companies are started each day, and emerging technologies such as smartphones, cloud infrastructure platforms and enhanced web development tools have made it even quicker and easier to get started. However, contrary to what the media portrays, far from all startups succeed, i.e. only one in 58 turn out successful [13]. Several authors [1][11][15] argue that this is not only attributable to fierce competition, but to how startups are typically run.

During the early 2000's, agile software development methods [4] gained traction in the software development community. This was followed by increased attention towards metrics-driven development [10] where techniques such as A/B testing are used to base decisions on data instead of opinions. During the same time, Lean Startup concepts gained popularity in the startup community [1][5][13][15]. However, our interviews with nine startups shows that applying these principles in practice proves to be difficult. In this paper, we present the Early Stage Software Startup Development Model (ESSSDM) as a solution. The model has been validated in a startup setting.

The contribution of this paper is twofold. First, it presents a validated process model that manages a portfolio of ideas, whereas existing approaches focus on one idea. Second, the model provides a detailed approach to handling individual ideas with clear stage gates and exit criteria that provides much more guidance than existing approaches.

This paper is structured as follows. First, we outline the research method, followed by background and related work with respect to product development in early stage startups. Then we present the findings from interviews with nine existing startups and the problem statement, followed by the section that presents the ESSSDM model that presents the key contributions of the paper. Next we present the validation of the model using a case study, followed by the conclusion of the paper.

2 Research Method

Design Science Research (DSR) was chosen as the primary framework for the research. DSR differs from traditional research in that it focuses on learning through design, i.e. the construction of artefacts. The act of designing is, within DSR, used as a research method or technique [17].

Takeda, et al. [17] describes a model of the iterative design cycle. It comprises five phases. (1) Awareness of problem. Research proposal and research questions are formed. (2) Suggestion. Abductive reasoning, drawing from existing knowledge and theory within the field, leads to a suggestion of how to solve the problem. (3) Development. The suggested solution is realized in the form of an artefact. (4) Evaluation. The artefact is evaluated according to defined criteria. (5) Conclusion. When the artefact performs to satisfaction according to evaluation criteria, iteration stops and conclusions are drawn.

Being iterative, the model allows for moving back and forth between phases. If new information emerges during the development of the artefact, phase one and two can be revisited, and a new or modified suggestion formed. Similarly, during phase four, if evaluation criteria are not met, phase three is revisited and the artefact improved.

DSR was deemed a good fit due to the context of the research project. With the authors taking part in the forming of a startup, the design of an artefact aimed at mitigating typical challenges and problems were seen as highly relevant. Furthermore, the close proximity to a real-world startup meant the artifact could be rapidly iterated over/evaluated.

2.1 Research methodology instantiation

Awareness of problem. The research questions investigated were (1) What are the typical challenges and problems in early stage software startups? (2) What solution would serve to mitigate the identified challenges and problems?

Suggestion. A generic literature review was conducted, focusing initially on agile practices and in later iterations on Lean Startup theory. In addition, semi-structured interviews with industry professionals were carried out. Although an interview guide with template questions was written, structure was kept loose so that discussions were free to go in new and interesting directions. Abductive reasoning based on the literature and the interviews led to a set of problems (see chapter 5) and a suggested solution in the form of a process.

Development and evaluation. During the deductive stages, data was gathered mainly through participatory observation and reflective journals. The process was built for and evaluated on the aforementioned startup project. Revisits to the suggestion phase were frequent. In total, the process saw three major revisions, and multiple minor ones.

3 Background and Related Work

Emerging technologies such as smartphones, cloud infrastructure platforms and enhanced web development tools have made starting a company very easy. However, only one in 58 newly started companies is successful [13]. To understand these challenges, we need to understand what constitutes a software startup. A popular definition, by Eric Ries [15], states that "a startup is a human institution designed to deliver a new product or service under conditions of extreme uncertainty." For the purpose of this paper, we narrow the definition by including the following characteristics: startups have limited resources in terms of people and funding. Consequently, startups run on tight schedules and are exploratory in nature, initially lacking clear requirements, customers and even business models.

3.1 Agile software development

Over the last decade, several agile software development methods have been developed, but Scrum [16] is one of the most popular agile development processes, and is founded on empirical process control theory. Empiricism states that knowledge comes from experience and that decisions should be made based on what is known, not on what is believed. Empirical process control theory is a way to deal with "imperfect processes that generate unpredictable and unrepeatable outputs" by

prescribing frequent inspection and adaptation. In Scrum, inspection and adaptation is applied not only to the software product in development, but to the process as well.

3.2 The Lean Startup movement

Agile development processes are solution focused. That is, they are mainly applied in situations where the problem is well known/understood but the solution is not. In a startup context, however, uncertainty is even greater: both problem and solution are typically not well understood. For the software engineer working in a startup, however, being focused on the solution is often not enough. A product is more than a solution, it is a business model, and in a startup the software engineer is often involved in both business and technical development efforts.

This customer and problem focused thinking has been advocated in the past by people such as Steve Blank [1], John Mullins and Randy Komisar [13], but has in recent years gained traction because of Eric Ries [15] and the Lean Startup movement. Ries noticed that, because of solution focused thinking, a lot of software startups were failing, including his own. Turns out, many were spending time and money developing products that people were not interested in. He calls this achieving failure: successfully executing a bad plan. While projects may have been delivered on time and on budget, and with good design to boot, nobody wanted the product. This underscores the importance of understanding the problem before the solution. While Ries can be credited with coining the term Lean Startup and bringing the word to the masses, his work is heavily influenced by, in particular, that of Steve Blank, who outlined the Customer Development Model in 2005 [1]. Others, such as John Mullins and Randy Komisar, contributed greatly to the field before Ries with *Getting to Plan B* in 2009 [13]. Likewise, Jason Fried and David Heinemeier Hansson touched upon many similar concepts with the book *Getting Real* [5].

3.3 Customer Development

In his book *The Four Steps to the Epiphany* [1], Steve Blank presents the Customer Development Model, which is further explained in his 2012 follow-up *The Startup Owner's Manual* [2]. Blank argues that the highest risk in building a business is not building the product, but finding people to pay for it. Startups generally do not lack products; they lack customers. Therefore, the traditional product centric development model, where a product is thought of, developed, beta tested then launched is flawed because it ignores customers up until product launch. The Customer Development model, on the other hand, considers customers from the start. It is a structured process for testing business model assumptions (or hypotheses) about markets, customers, channels and pricing. The model consists of four steps, where the first two mark the search for the business model, and the last two its execution.

3.4 The Lean Startup

Ries published *The Lean Startup* in 2011 [15], wherein he states that entrepreneurship is a form of management. It is fundamentally different from traditional management in that the unit of progress is learning about customers and what they want. And because agile methodologies are not enough for this purpose, Ries brings in Steve Blank's Customer Development Model to fill the gap. Another central concept within *The Lean Startup* is The Pivot, which is the term Ries uses for when a startup changes direction, but stay grounded in what they have learned (about customers) so far. He claims having pivoted is the most frequently occurring commonality among successful startups: they seldom end up doing what they initially set out to do. By reducing the time between pivots, it is possible to increase the odds of success, before running out of money.

The Build-Measure-Learn (BML) loop describes the concept of validated learning. Ideas are turned into products by building them, data is gathered by measuring how products are used by customers using various techniques, and new ideas can then be formed from what is learned by analyzing the data. One major iteration through the feedback loop constitutes a potential pivot. By reducing the time it takes to get through the BML loop, time between pivots can be reduced, and the odds of success increases.

Ries suggests treating this process of validated learning as if one were a scientist: by applying the scientific method and thinking in terms of learning experiments. By formulating falsifiable hypotheses (statements that can be proven wrong by empirical data) learning objectives can be defined up front. By running experiments, hypotheses are validated (proved valid/invalid), by analyzing the data typically leading to the formulation of new hypotheses.

The Lean Startup suggests many techniques for speeding up the BML loop time. One of them is building a Minimum Viable Product (MVP). An MVP is typically the first version of a product released to customers, and should contain only the absolute minimum in terms of features and design for it to become viable to the customer, i.e. it solves the customer's problem.

3.5 Running Lean

Running Lean [11] is a rigorous process and handbook for creating Lean Startups based on [1][15]. The process is divided into three steps: (1) document Plan A, (2) identify the riskiest parts of the plan and (3) systematically test the plan. Documenting the initial plan is done in the form of a Lean Canvas, which is Maurya's version of Osterwalder's Business Model Canvas [14]. The Lean Canvas captures and focuses on the entire business model, not only the product/solution. The canvas is a living document, and is continuously updated as the plan iterates from Plan A to a plan that works.

4 Interview Results

Nine software startups in the Gothenburg region were interviewed using a semi-structured format, both from a business and a software perspective. The purpose of these interviews was to get a good understanding of how software startups typically work in the early stages, and if any patterns, processes or best practices could be observed.

For each startup, the following will be discussed: (1) **Context**. Size of company, area of business, technology platform, type of product. Where the idea originated from. (2) **Development practices**. Business and software development practices. How the company conducts its operations. (3) **Problems/challenges**. Things that are viewed either as problematic or challenging when running a startup.

From a software development perspective, all companies used agile practices, especially Scrum and Kanban. From a business development perspective, a few companies were aware of Lean Startup methodologies and worked in that manner, but most were either unaware or found it difficult to apply in their situation. Some did, however, follow principles similar to Lean, without necessarily labeling it so. That includes working closely with customers and pivoting towards product/market fit.

Of those not following Lean Startup practices, few worked actively with validating product concepts early and often with customers (trying to pinpoint underlying problems) before building and scaling a solution. In some cases this was due to products and business models having been copied from existing ones, to be applied in different contexts/countries, thereby reducing uncertainty and the need for extensive validation. Also, the opinion was voiced that Lean Startup is difficult to apply in situations where the product is depending on a network effect. In such cases, scaling before reaching product/market fit might be necessary.

Startups that did put a lot of effort into understanding underlying problems either followed Lean Startup or created new products, i.e. not copying existing ones. Those same startups had also pivoted the most.

Many proclaimed to be data-driven to some extent, keeping close track of various metrics. Even so, most strategic decisions were based on intuition and gut feeling. Many dabbled in A/B testing of their user interfaces, but this was mostly viewed as an optimization technique. No one A/B tested features. Those following Lean Startup did perform experiments according to the scientific method but admitted it was difficult to base strategic decisions on data alone. Thus, there is still ways to go before startups are truly data-driven, or apply fact-based decision-making.

It became apparent that there is an early-stage process not heavily discussed in the literature, where different product ideas are weighed against each other before a decision is made on what product to develop. This often happens prior to the forming of the company. A structured approach to tackle this task seemed to be lacking. Some startups brought this early-stage idea selection process further, by actively investigating multiple ideas in parallel even after forming the company.

Startups are run in many different ways and there are many different types of startups. On the software development side, all companies adhere to agile methods,

but on the business development side, few agreed upon best practices could be observed. What all can agree upon, however, is that it is difficult to work in an organized and structured manner in an early stage software startup.

5 Problem Statement

Lean Startup principles significantly reduce the uncertainty that surrounds startups and increasing their success rate. These principles, however, are rather philosophical in nature and hard to put into practice. This was confirmed by the interviews: entrepreneurs either were not familiar with the concepts or had a hard time implementing them in their companies. Although some authors, e.g. [11], claim to provide a guide for implementing Lean Startup principles in practice, we have identified some key areas where improvements are needed:

1. Existing processes and theories do not adequately support working on, or investigating, multiple product ideas in parallel, as part of an idea portfolio.
2. Existing processes and theories provide insufficient validation criteria for moving product ideas forward through process stages.
3. Existing processes and theories give no clear guidance on when to abandon a product idea.
4. Existing processes and theories provide insufficient suggestions of what techniques to use and when, while validating product ideas.

6 ESSSDM

In response to the identified challenges, we have developed the Early Stage Startup Software Development Model (ESSSDM). The model extends existing Lean Startup approaches, incorporates the results from interviews with entrepreneurs as well as is based on earlier experiences with startups by the authors. The process is defined in a clear step-by-step fashion with clear exit criteria for each stage. In addition, the model presents guidance concerning the techniques and practices to employ during the different stages. Moreover, the process supports multiple product ideas, constituting a product idea portfolio, being investigated in parallel by a team.

The overall process is comprised of two major levels. On the topmost level, managing product idea portfolio, one develops hypotheses concerning potential problems to solve, drafts up rough business models and documents these in a prioritized product ideas backlog. On the second level, a product idea is picked from the backlog and is validated systematically using the Build- Measure-Learn (BML) loop until the product is deemed scalable, put on hold or abandoned.

6.1 Level 1: Managing product idea portfolio

Typically when creating a startup, a product domain is selected and frequently the team has at least one product idea. Independent of the specific initial product idea, the first task is to generate additional promising product ideas. Ideas can be crude; the rest of the process is designed to iterate over an initial (crude) idea and improve upon it. Often, it is worthwhile to investigate multiple ideas in parallel as it allows for more than one person involved in the startup and because it decreases the risk of an overly emotional connection with a specific idea. This increases the odds of finding an idea worth pursuing.

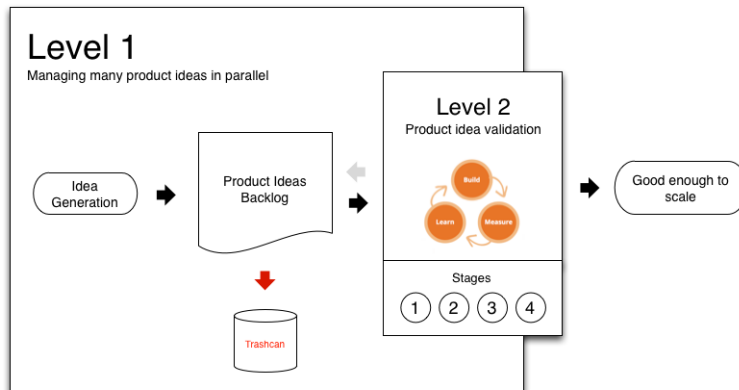


Figure 1: Overview of ESSSDM

6.2 Step 1: Product idea generation

Typically, ideas spawn instantly or emerge over time. A product idea contains, as a minimum, a problem or collection of problems that need solving. One of the primary techniques is to organize exploratory interviews with potential customers in the domain. The selection of interviewees should be influenced by:

- Areas where the entrepreneur has domain knowledge
- Copying and tweaking an existing (successful) product
- Problems experience by the entrepreneur (scratch your own itch) [5]

6.3 Step 2: Documenting product ideas as business models

In order to compare product ideas, it is important to document these in a comparable format, e.g. Lean Canvas [11]. Working on multiple product ideas (as part of an idea portfolio) in parallel provides several advantages, one being that there is always something in the pipeline to work on when one idea is on hold: waiting for interview session dates to arrive or waiting for feedback or other data. However, if working on multiple ideas in parallel, it is important to enforce a limit on how many ideas can be worked on simultaneously. Our validation showed that the number of concurrent product ideas pursued should be below 50% of team size. The product ideas backlog is prioritized continuously, with the most promising ideas actively worked on. Eventually, as one idea gains traction and demands more resources, other ideas should be put on ice until resources become available.

6.4 Level 2: Product idea validation

When an idea is picked from the product ideas backlog, it undergoes systematic validation. This process can be described as a feedback loop comprising risk prioritization and BML looping. The product idea moves through four stages, each with its own activities and defined exit criteria. The four stages are: (1) understanding the problem; (2) defining the solution; (3) qualitative validation; (4) quantitative validation. Each stage is associated with different sets of risks. Risks are prioritized and put on a risk backlog. With risks identified, assumptions (falsifiable hypotheses) can be formed and tested using the BML loop. The learning that is gained from validating the assumptions is fed back into the product idea and the risk backlog. By doing this, risks can be dealt with one by one, through the stages, until the product is deemed scalable, or until a risk becomes blocking and the product idea invalidated.

6.5 Stage 1: Understand the problem

During the first stage of testing the plan, focus lies on gaining a deep understanding about the problem and who experiences it. The key risks during this stage are: (1) do we have a problem worth solving?; (2) who experiences this problem?; (3) what competition is there? Before proceeding to the next stage, at least half of potential customers should give a positive indication of the product idea. Also, the team should identify a promising customer segment, find at least one problem that customers want solved, as well as build understanding of how customers currently solve the problem.

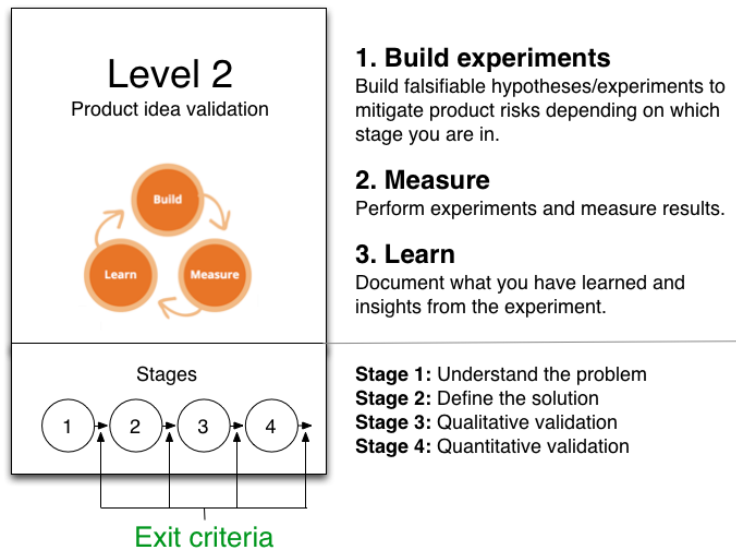


Figure 2: ESSSDM Level 2 process

6.6 Stage 2: Define the solution

In this stage, the purpose is to define a potential solution and communicate this to potential prospects. When defining the solution, the team focuses on the simplest possible solution to solve the problem: the MVP. The key risks during this stage are: (1) what is the minimum feature set for the MVP?; (2) who are the early adopters?; (3) what would customers pay for the solution?

In order to move to the next stage, the team interviews a minimum amount of potential customers, has identified a typical early adopter, has defined a MVP, has confirmed that customers are willing to pay for the product, has verified that the solution is feasible to implement within a realistic time horizon, as well as has secured a test user base for the product.

6.7 Stage 3: Qualitative validation

The purpose of the third stage is to develop an MVP and launch it to early adopters in order to verify that it solves their problem and that they are willing to pay for it. Key risks are: (1) Does the MVP demonstrate a Unique Value Proposition (UVP)?; (2) Are early adopters willing to pay?; (3) How to sustain an influx of early adopters?

6.8 Stage 4: Quantitative validation

During the fourth and final stage the product is launched to a larger group of people, including non-early adopters. The key risks are: (1) Do people want the product?; (2) How to reach customers through inbound channels?; (3) Will the business model hold? This stage marks the beginning of the scaling phase.

7 Validation

ESSSDM was evaluated as part of a startup project at the School of Entrepreneurship at Chalmers University of technology. The following evaluation criteria were considered:

- The process must be perceived as practical by project members
- The process must support working on, or investigating, multiple product ideas in parallel, as part of an idea portfolio
- The process must provide clear guidance on when to move product ideas forward through process stages
- The process must provide clear guidance on when to abandon a product idea
- The process must provide clear guidance on what techniques to use and when, while validating product ideas

7.1 Project context

The project group consisted of five students: three business developers and two software engineers, working in an incubator setting. The incubator provides office space and business advisors. As of this writing, the project has been running for five months, and will continue to run for four more. The focus of the project was to find a promising product idea in the small business segment and turn it into a company.

7.2 Process evaluation

At the beginning of the project, no ideas existed. Doing exploratory interviews with potential prospects made the team both knowledgeable with how small businesses operated (the targeted market segment) and provided many ideas. As more exploratory interviews were conducted, a picture of how small businesses operated in general began to emerge and the team began to focus on certain promising leads. When 40+ exploratory interviews had been conducted, the value of each new interview became less and less, with no new learnings gained, and focus shifted from problem understanding to solution defining.

Project members perceived the process as practical. Based on the amount of people in the team, the team always had ideas to work with and never lost its momentum. Having a cap of three simultaneous ideas worked out well for the size of the team due to the fact that one business developer was responsible for one idea each. When dividing responsibility this way, the importance of having a comparable format (Lean Canvas) became apparent during idea prioritization sessions. Furthermore, using canvases made the team think of the product not as a solution, but as a business model. This gave the team a broader perspective and made it easier to spot the potential in each product idea.

The process gave clear guidance on when to move product ideas forward through process stages. Using well-defined exit criteria for each stage contributed to the team's good momentum and allowed each business developer to work independently.

The process gave clear guidance on when to abandon product ideas. The team constantly evaluated whether exit criteria had been reached or not. When additional interviews resulted in no more learning, and there was no clear path towards fulfilling the criteria, the team took a decision: pivot, persevere or abandon. If there was no obvious way to pivot, the team usually opted to abandon the idea in favor of another one from the product ideas backlog.

The process gave clear guidance on what techniques to use and when for validating product ideas. The proposed techniques proved efficient; no technique took more than two working days to apply. This was valuable in order to push the project forward and get fast feedback from customers.

Concluding, initial validation in the project context suggests that ESSSDM overcomes the challenges discussed in the problem statement.

8 Conclusions

Software startups are more popular than ever and growing in numbers. They operate under conditions of extreme uncertainty, and face plenty of challenges, underlined by their high failure rate. In this paper, these challenges were investigated through a literature study of the Lean Startup community as well as through interviews with industry professionals. The result of this investigation showed that few practitioners apply Lean Startup methods because these were found too vague and imprecise to implement in practice.

In response to the investigation, we developed the Early Stage Software Startup Development Model (ESSSDM) that addresses the identified challenges:

- The process supports working on, or investigating, multiple product ideas in parallel, as part of an idea portfolio
- The process provides clear guidance on when to move product ideas forward through process stages
- The process provides clear guidance on when to abandon a product idea

- The process provides clear guidance on what techniques to use and when, while validating product ideas

In future work, we intend to provide more validation especially to stages 3 and 4 of level 2 of ESSSDM and to apply to the process to additional startups.

References

- [1] Blank, S., Dorf, B. (2006) *The Four Steps to the Epiphany: Successful Strategies for Products that Win* (3rd edition), Cafepress.com
- [2] Blank, S. (2012) *The Startup Owner's Manual: The Step-by-Step Guide for Building a Great Company*, K&S Ranch, Inc.
- [3] Campbell, D. (2010) *Software as a Service: spend and payment solution*. Summit: Canada's magazine on public sector purchasing. <http://www.summitconnects.com> (June 2010).
- [4] Cockburn, A. (2006) *Agile Software Development: The Cooperative Game* (2nd edition), Addison-Wesley Professional.
- [5] Fried, J., Hansson, D.H., Linderman, M. (2009) *Getting Real: The smarter, faster, easier way to build a successful web application*, 37signals.
- [6] Furr, N., Ahlstrom, P. (2011) *Nail it then Scale it: The Entrepreneur's Guide to Creating and Managing Breakthrough Innovation*, NISI Institute, USA.
- [7] Gartner, Inc. (2011) *Forecast: Software as a Service, All Regions, 2010- 2015*. <http://www.gartner.com/it/page.jsp?id=1791514> (14 Sept. 2011).
- [8] Gustafsson, A., Qvillberg, J. (2012) *Implementing Lean Startup Methodology: An Evaluation*, Chalmers University of Technology.
- [9] Holson, L.M. (2009) *Putting a Bolder Face on Google*. The New York Times, February 28. <http://www.nytimes.com/2009/03/01/business/01marissa.html>
- [10] Klasaviius, M. (2012) *Metrics-Driven Development*. InfoQ, November 30. <http://www.infoq.com/articles/metrics-driven-development>
- [11] Maurya, A. (2012) *Running Lean: Iterate from Plan A to a Plan That Works* (2nd edition), O'Reilly Media.
- [12] McClure, D. (2007) *Startup Metrics for Pirates: AARRR! 500 Hats*. <http://500hats.typepad.com/500blogs/2007/09/startup-metrics.html> (26 Sept. 2007). 56
- [13] Mullins, J., Komisar, R. (2009) *Getting to Plan B: Breaking Through to a Better Business Model*, Harvard Business Review Press
- [14] Osterwalder, A. and Pigneur, Y. (2010) *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers*, Wiley.
- [15] Ries, E. (2011) *The Lean Startup: How Constant Innovation Creates Radically Successful Businesses*, Penguin Group, London.
- [16] Schwaber, K., Sutherland, J. (2011) *The Scrum Guide*, Scrum.org
- [17] Vaishnavi, V. and Kuechler, W. (2004). *Design Science Research in Information Systems*. January 20, 2004, last updated September 30, 2011. <http://www.desrist.org/design-research-in-information-systems/>

