



HAL
open science

From Startup to Scaleup: An Interview Study of the Development of User Experience Work in a Data-Intensive Company

Kati Kuusinen, Martin Kjølbye Sørensen, Nicklas Mandrup Frederiksen,
Niclas Kildahl Laugesen, Søren Holm Juul

► To cite this version:

Kati Kuusinen, Martin Kjølbye Sørensen, Nicklas Mandrup Frederiksen, Niclas Kildahl Laugesen, Søren Holm Juul. From Startup to Scaleup: An Interview Study of the Development of User Experience Work in a Data-Intensive Company. 7th International Conference on Human-Centred Software Engineering (HCSE), Sep 2018, Sophia Antipolis, France. pp.3-14, 10.1007/978-3-030-05909-5_1 . hal-02270708

HAL Id: hal-02270708

<https://inria.hal.science/hal-02270708v1>

Submitted on 26 Aug 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

From Startup to Scaleup: An Interview Study of the Development of User Experience Work in a Data-Intensive Company

Kati Kuusinen¹[0000-0003-3076-1356], Martin Kjølbye Sørensen², Nicklas Mandrup Frederiksen², Niclas Kildahl Laugesen², and Søren Holm Juul²

MMMI Institute, University of Southern Denmark, 5320 Odense M, Denmark
kaku@mmmi.sdu.dk

University of Southern Denmark, 5320 Odense M, Denmark
{maso13, nifre12, nilau12, sojuu12}@student.sdu.dk

Abstract. Small startups often do not have the resources or the skills for upfront qualitative user studies and user experience design. Instead, they operate in market-driven environment where requirements are often invented and validated through frequent releases. The research on how startups do this in practice is scarce. Even less is known about what kind of engineering and user experience practices would help startups to survive and grow into successful businesses. This paper describes how user experience work emerged and grew in a data-intensive startup company operating in the financial sector in Denmark. The paper is based on the interviews of four persons with different roles in the startup. The emerging issues in user experience were found to be in the lack of skills in user experience and in balancing between the use of quantitative and qualitative user data. To conclude, it seems evident that startups would benefit from user experience practices but more research is needed to develop practices that would suit for this particular context.

Keywords: Agile development · User experience · Startup · Scaleup.

1 Introduction

All software development companies balance between process control and flexibility in their ways of working. Small startups tend to remain in the more flexible end of the scale with less defined development processes [21]. Even agile management approaches such as Scrum [20] can be too tedious and limiting for startups to follow with their small resources. However, when they start to scale up, grow, they often face the need for more structured approach for their development work.

Regarding user experience (UX) [15], startups often start with minimal and restricted product versions and with limited advance information from the potential user segment [9], [10]. However, small companies are more sensitive to customer influence than larger ones [21] and they can even fail in validating their business model because of poor UX [9]. Although there have been attempts to

guide startups in organizing their UX effort with their often so scarce UX resources and skills [10], the scientific understanding of the meaning of UX and methods for human-centered development in startups and scaleups is still emerging.

This paper presents an interview study conducted in a Danish company on the edge of the "growth chasm" i.e. a startup turning into a scaleup. We interviewed three out of the ten permanent staff members (chief information officer / IT manager, data scientist, and software developer) and a part-time UX consultant trainee. We present their views on UX and the practices they had experimented with in their software development over the years. Furthermore, we discuss those views and practices in relation to the agile UX and startup literature.

The rest of the paper structure is as follows. Section 2 gives an overview to the related work. Section 3 presents the research method. Section 4 describes the interview findings. Section 5 discusses the findings in relation to previous research. Finally, section 6 is the conclusion of the paper.

2 Background

2.1 Agile UX

Agile UX work in established companies is often based on the integration of human-centered development practices with the agile process model (such as Scrum [20] or Kanban [14]) the company uses. Based on a recent systematic review, frequently recommended practices in agile UX include conducting little UX design work before starting implementation, doing iterative design and development, and having a cohesive product design [3]. Commonly used practices include usability testing, creating user stories, having users directly involved in the development, and using scenarios [3]. There, however, is no evidence of the suitability or applicability of these approaches to UX work in startups.

2.2 Startups and Culture of Experimenting

Startups often comprise of small teams that might have lack in skills and experience [8]. They operate with scarce resources in extremely uncertain, high-risk conditions and therefore have a strong time pressure and urge for short time to market [8]. Startups operate in market-driven environment where requirements are often invented and validated through frequent releases [5] instead of continuous customer involvement as in agile. Software startups do not necessarily follow any software development methodology although there are some directed at them. One of them is the lean startup [19]. It describes the business model hypothesis driven build-measure-learn loop where business or design ideas are built into falsifiable hypotheses that are tested in experiments with real users in actual markets. The outcome is then taken as validated learning to the following loop. This approach is especially aimed for highly dynamic environments

where both customers and the product under ideation are unknown and thus customer value cannot be guaranteed by creating more or better designs; i.e. for the typical startup environments.

Larger established companies have also been adopting lean startup practices in their internal startups [6], [18] and with innovation experiment and continuous experimentation systems [2], [7], [17], [23]. Yaman et al. [23] report that UX team especially was keen on reaching a state where they could make data-informed decisions. Their expectation was that being able to refer to the data would increase both developers and stakeholders buy-in for the design decisions. Another benefit they saw in the approach was being able to focus the improvement effort on the most used features and remove unused features [23]. Being able to remove features will be extremely important in continuous development as otherwise the software will just continue growing.

2.3 Startup Antipatterns

Klotins et al. [12] analyzed 88 startup experience reports revealing three antipatterns Fig. 1, one per each startup phase. The first antipattern can happen during the build of the first version if the introduction to the market fails to happen fast or cheap enough or if the product itself is not competitive enough. The second one can happen if customers are not attracted to the product or the product is not developed further fast enough. Finally, the third one can happen in the phase where the company should grow into new markets if the company fails in keeping the customers happy or the costs down, or because they are not able to get beyond the initial market. Thus, the authors claim that many of the reasons behind startup failures could be mitigated by better engineering practices. Many of the symptoms might also be tackled with better UX practices as they are related to difficulties in attracting people or keeping them satisfied with the product. Thus, it seems clear that startups would benefit from appropriate UX practices and skills.

2.4 UX in Startups

Hokkanen et al. [11] studied UX practices and needs in startups. In their sample of eight Finnish startups, five developed their product having the owners' own personal needs in mind and one developed it for the assumed average user. Feedback was gathered mainly from friends whether or not they belonged to the target user group. Two of the startups had interviewed potential users, three had conducted informal user testing and two had used paper prototypes at some point. All of them used Google Analytics. Most commonly mentioned need towards the scaleup phase was to utilize user data analytics to better understand the user but none of the companies had a clear plan or strategy for the use of analytics or for UX work.

Hokkanen et al. [10] present a UX strategy framework for startups. They suggest that startups concentrate on sellability of their product through its attractiveness, approachability and professionalism.

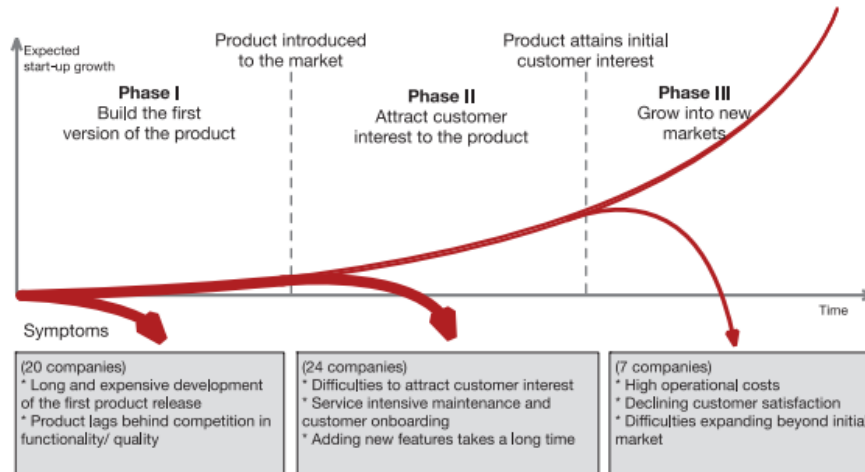


Fig. 1. Startup progression phases and symptoms for anti-patterns [12].

3 Method

The goal of the study presented in this paper was to add to the understanding of how startup companies see UX and working towards improving it in the context of their agile software development. The study was thus explorative in nature and we chose to conduct it as a series of semi-structured interviews in one single startup.

The study was conducted as student work over a 5 ECTS research methodology course aimed at master level software engineering students at the University of Southern Denmark in fall 2017. The first author was in charge over the course and the rest of the authors were students on the course planning, conducting, analyzing and reporting the study in the supervision of the first author. The first author contacted a number of companies with a few topics related to software engineering and the company either picked the topic they were most interested in or refused to participate. The first author then discussed with the company representative to find a relevant research question within the topic to be studied in the company. The companies were selected from a pool of Danish companies that had indicated their interest in working with software engineering students. The first author selected the research approach based on the research question and suitability in the organizational context (interview study in this case) and guided the participating companies by email on how to pick suitable research participants among the staff. The incentive for the participating companies was that companies got the student report of the study they participated in and thus learned more about themselves and perhaps got some suggestions on how to improve from the current state.

The students working in groups of three to four selected a topic they wanted to investigate as their course project. The students created study plans including

short literature reviews, interview guides, and informed consent forms in the supervision of the first author and had to have them approved by her before starting the data gathering. The students handled the communication with the companies in the supervision of the first author from there on. They also chose who they want to interview and agreed on the practicalities directly with the companies.

3.1 Study Plan

The students selected to conduct the interviews in a semi-structured format as follows. Interview was introduced as an informal conversation where the conversation was driven by a series of questions that had been established beforehand. The entire interview was structured through an interview guide that served to ensure that all practicalities were handled and the most essential questions were asked. A semi-structured interview type was chosen based on the varying roles of the interviewees. The interviewer improvised further questions based on the response and investigated interesting topics further.

3.2 Procedure

The interviews were conducted at the company office in November 2017. All four interviewees had different tasks and roles at the company which gave a broader insight in the company's ways of working. The duration for the interview was approximately 45 minutes each and it was recorded. Before the interview took place, the interviewee was asked to sign a consent form which said that they agreed to the recording of the interview and that they had the right to not answer a question and if they felt uncomfortable they were allowed to stop the interview at any time. The recordings were transcribed and deleted when the report was ready, three months after the interviews at the latest.

The purpose of the interview was to establish how the staff work with UX and agile practices. Thus, the main questions were about how user experience is addressed in the company and how to improve UX work in the company. The first participant was the chief information officer (CIO) / IT manager. He organized all the projects the company handles. He was also the person who arranged the weekly Scrum meetings. The second participant was a data scientist who worked with internal systems. He was primarily working with the backend so he extracted all the data from the database that needed to be analyzed. The third participant was a student trainee who studied user experience design at the university and worked with new templates for the company website, where he made some split tests that dealt with new design choices. The fourth participant was a software developer, who was responsible for all the coding on the different platforms that they worked on. See Table 1 for an overview of the interviewees.

3.3 Analysis

We analyzed the transcribed interview data using Qualitative Data Analysis as described in [13]. Before coding the interview data, the students excluded ir-

Table 1. Interviewed roles.

Code	Role	Age (years)
J1	Chief information officer (CIO) / IT manager	48
J2	Data scientist	25
J3	UX consultant	25
J4	Software developer	26

relevant information by first excluding bits with no information and afterwards removing any data that did not have any significance to the topic. When each piece of information was extracted from the transcripts and coded, the students together used physical affinity diagram [1] in which each single piece of information was written down on a post-it note and first grouped by interviewee and then thematically as themes arose from the data. Grouping the information by interviewee gave a clear view of how the different job positions saw the pros and cons and grouping the information by topic gave a quick overview of how much energy the company put into the agile workflow and in user-centered design. Lastly, the affinity wall made it easier to outline the data and to promote discussion.

4 Results

4.1 Studied Company

The studied company had about 10 full-time employees and 15 trainees / student assistants working for them. Maturity-wise the company seemed to be in the so-called scale-up phase [1], [4]: it already had validated its product on the marketplace and had a steady and rather sustainable income. It also had existed for more than five years already. The leading brand of the company was an online consumer-finance service. Their business idea was to focus on online banking services instead of traditional retail services and face-to-face contact with the customer. They utilized online advertising services such as Google AdWords and Facebook Ads to attract new customers.

4.2 User Experience Work in the Company

In the early years, the company had not invested much effort into ensuring good UX of their services. The company did not have a UX designer or team and they did not pay much attention in creating UX designs before implementation. From the beginning, the company had leaned on profitability as an indicator of good UX: If the customers clicked on it, they assumed it worked for them.

Later on, they had added randomized experiments (A/B tests, or split tests) where half of the customer population was randomly exposed to one design version whereas the other half was exposed to another design version. This was done to compare between two design versions to pick the most profitable ones to

be used in the system. They considered that these experiments ensure good user experience as the design choices were validated on real users in the production environment. However, they acknowledged that this method did not give an explanation on why it worked but they considered it enough for the startup company at that point as it guaranteed acceptable revenue. They were able to use this approach because of the high volume of users accessing the site; the company got enough data for each experiment approximately in less than a week. Had they not had such an extensive user base, they think they would have taken another kind of approach to UX.

Recently, the company had realized that they needed to increase their focus on the user experience in their products and they hired a student assistant in a part-time UX consultant role. The assistant's job was to create and conduct both qualitative and quantitative user surveys and report the results to the development teams. Quantitative surveys often combined online user questionnaires with usage data analysis such as clicks or user interaction patterns. They also surveyed users intention and behavior in qualitative online studies. The company still did not conduct user interviews or utilized other user study methods requiring personal contact with the users.

4.3 Scrum vs Kanban in UX Work

The company had utilized Scrum with one week sprints for years. A new sprint normally started each Monday with a sprint planning meeting where all relevant staff members together groomed the backlog and picked tasks for the sprint to begin. Together they chose the tasks that they would commit to as a team. Interviewees considered this as helpful and the developer (J4) told about the calming effect of being aware of their tasks for the sprint already in the beginning of the week. The data analyst (J2) said he liked dividing his tasks into smaller and more manageable pieces for clarity. This practice also allowed them to rearrange tasks to accommodate possible bugs or changes in the environment each week.

Due to the tools they used for managing the development workflow, the interviewees reported on having to do quite a lot of not directly value-adding administrative work because "everything had to be commented and logged in the system". One of the interviewees mentioned that he kept forgetting to update the backlog after completing a task. Another said that he sometimes felt like an accountant because of all the listing and registering tasks.

As the UX consultant worked only part-time, he did not attend to the sprint planning meetings. Instead, he regularly had meetings with the CIO about his progress and results. This also meant that his tasks got assigned outside of the company's Scrum board. As a consequence, the UX consultant mostly worked alone and managed his own work by himself. He and other team members felt that it was problematic to use Scrum for UX work, especially with one week sprints as the UX surveys could not be finished in a week. This made the company to switch to Kanban. The interviewees described how they prioritized and organized tasks in different columns on a digital Kanban tool. The columns had labels indicating the state of the task. When talking about UX design choices,

the programmers explained that they made the design choices by themselves. The developer responsible for the task made UX design decisions alone. The UX consultant was not assigned to the task nor was the CIO interfering in the decisions. The design decisions made in the company were based on general assumptions and/or gut feelings. However, when faced with larger decisions, developers talked to their boss. In this case, they commonly came up with two or more designs and launched them in a split test. The primary goal for the test was to provide information for design choices and the solution that generated more clicks than the other was usually selected for the next version as-is.

The UX consultant was working on a UX design guideline for the company and all the interviewees believed that this guideline would greatly help them in their UX design decisions in the future so that they could base the decisions on something more than general assumptions.

Later on, they decided to abandon Kanban and go back to Scrum. However, all the interviewees agreed that they do not want to use "text-book Scrum". Instead they would tailor Scrum for their needs. They would only take the parts that worked for them, change some of the other things and completely opt not to do some. They believed that going back to the "Scrum inspired" ways of working would provide them more structure and improved collaboration.

The UX consult expressed some concerns in the choice of going back to Scrum fearing of the short sprints and having to deliver finished work within the sprint. This was especially a concern in relation to qualitative user tests. While the Scrum team ran a weekly sprint, it was difficult for the UX consultant to stick to the timeboxed sprint. As he stated, his "work cycle looked more like the rigid waterfall model than fast iterating sprints". His work cycle often consisted of research, analysis, and learning how to improve the the UX. As he put it, to study the customer, analyze the data, and then do some design requires more time than a week. In his opinion it is impossible to adapt and come up with results every week. When asked how he believed it should be done he suggested the dual track model with separate UX and development sprints running in parallel, similarly to [22]. In this way, the UX consultant would be able to start on finding UX solutions for a new problem while the programmer is finishing up with the old problem. When the programmer is ready for a new problem, the designer will already have something ready for him to do so that they can stay effective. The UX consultant also believed that he would have an opportunity to do qualitative user tests within a sprint while the programmer would be working on implementing a solution or testing the code. The interviewees all agreed that they liked working in an agile workflow and they could not imagine that anything else would fit the company better. They had some differences as to how well they believed the current Kanban solution worked compared to Scrum. They all apart from the UX consultant agreed that going back to their own version of Scrum is the right choice. They were not concerned about managing qualitative UX tests while working with Scrum although they did not have an exact plan for how to do it.

5 Discussion

The UX practices in the studied company seem comparable to those Hokkanen et al. [11] observed in the eight studied Finnish startups. The company strongly relied on usage data and split tests but they had not acquired qualitative data from the users to explain their choices. They also relied on the thought that the action of clicking indicates liking or a positive reaction. They had not planned their UX activities and they did not have a UX strategy. Instead their UX awareness had grown little by little, encouraged by the pressure to acquire a larger user base and later also by the urge to keep the acquired users satisfied.

The studied company seemed to have already crossed the first two chasms in the Klotins et al.'s [12] antipattern model. They had successfully brought the product into market and attracted users to the product. Currently, they were on the third chasm where they need to grow into new markets, control the costs, and to keep their current and future users happy. They have started to conduct user surveys. Designing good and effective surveys and being able to transfer the learnings into the design is not, however, an easy task. Perhaps adding a few user interviews in the method palette could help them to better empathize with the user.

The company had used Scrum for years but they tried changing to Kanban soon after hiring a UX person. In their study of the fitness of Scrum and Kanban to UX, Law et al. [16] conclude that Kanban is generally better suited with UX work. Our study supported this. The company chose to go back to Scrum after trying out Kanban although they thought Kanban was better for integrating UX work with other development efforts as it did not force UX work into timeboxes. However, based on research evidence, UX can be integrated to either process model.

The company had adopted practices from both lean startup type build-measure-learn loop and Scrum. They did not follow any process model strictly but picked practices that they felt suitable for them. They could slip from these practices when they, for instance, were in hurry which indicates that the selection of the practices was not a strategic choice but maybe convenience reasons played a significant part in it. The company probably would benefit from a strategic planning of their ways of working such as planning of the experiments more carefully to reflect business or design ideas instead of using them to solve difficulties in making individual design choices.

Startups operating in web and cloud environments often have the benefit of getting a large usage data. This data can offer a vast amount of information if collected and analyzed wisely. It seemed that the studied company did not fully utilize the potential of the data which is in line with the finding of [11]. Thus, the startup, and maybe startups in general, would benefit from improved skills in user-oriented data analytics. However, the dilemma that was experienced also with survey data remains: the challenging task of utilizing the data in successfully guiding the design decisions unfortunately requires great skills and experience in UX research and design.

5.1 Research Quality

The study was conducted by inexperienced students as course work and thus the interview might not have captured all relevant issues. For instance, some nuances and reasons behind the choices the company made remained unanswered. The study would have benefited from a validating round of interviews or discussion over the findings with the company representative.

The study was conducted in a single startup company and thus the results reflect only that particular startup. We however utilized several mitigation strategies such as theory triangulation and interviewing a rather large number of roles in the small startup where the total number of permanent staff members was only ten. Our results can be reflected in the light of previous research which supports the credibility of our results.

UX in startups is an understudied topic with only few studies focusing on it. Thus, our study offers a novel contribution to the academic field by confirming the previous results, offering a qualitative, deeper view in a single representative startup, and by explaining the previous research conducted in larger companies in the context of startups.

6 Conclusion

We conducted an interview study on UX practices and agile ways of working in a single Danish startup. We increased the understanding of the ways of working by describing the practices and challenges the startup has encountered during its lifetime. We conclude that more research is needed to create practices and knowledge on UX that can be applied in small startups operating with scarce resources and UX skills. It is becoming clear that better engineering and UX practices can help startups in their volatile first years as they struggle to acquire users and to keep them happy. These practices should take into account not only the characteristics of startups but also the market-driven environment in which they operate. Therefore, the UX practices ought to arise from experimenting with real users and adding qualitative insight into that. Moreover, the current human-centered software engineering literature in general would benefit from more research on how to conduct UX work in data-driven development environments as it is a trend in software engineering beyond the startup scene.

References

1. Aernoudt, R.: Executive forum: the scale-up gap: and how to address it. *Venture Capital* **19**(4), 361–372 (2017). <https://doi.org/10.1080/13691066.2017.1348724>
2. Bosch, J.: Building products as innovation experiment systems. In: *International Conference of Software Business*. pp. 27–39. Springer (2012)
3. Brhel, M., Meth, H., Maedche, A., Werder, K.: Exploring principles of user-centered agile software development: A literature review. *Information and Software Technology* **61**, 163–181 (2015)
4. Commission, E.: *Europe’s next leaders: The start-up and scale-up initiative* (2016)

5. Dahlstedt, A., Karlsson, L., Persson, A., NattochDag, J., Regnell, B.: Market-driven requirements engineering processes for software products—a report on current practices. In: International Workshop on COTS and Product Software: Why Requirements Are So Important (RECOTS) (2003)
6. Edison, H.: Lean internal startups: empowering software product innovation in large companies (2017)
7. Fagerholm, F., Guinea, A.S., Mäenpää, H., Münch, J.: The right model for continuous experimentation. *Journal of Systems and Software* **123**, 292–305 (2017)
8. Giardino, C., Unterkalmsteiner, M., Paternoster, N., Gorschek, T., Abrahamsson, P.: What do we know about software development in startups? *IEEE software* **31**(5), 28–32 (2014)
9. Hokkanen, L., Kuusinen, K., Väänänen, K.: Early product design in startups: towards a ux strategy. In: International Conference on Product-Focused Software Process Improvement. pp. 217–224. Springer (2015)
10. Hokkanen, L., Kuusinen, K., Väänänen, K.: Minimum viable user experience: A framework for supporting product design in startups. In: International Conference on Agile Software Development. pp. 66–78. Springer (2016)
11. Hokkanen, L., Väänänen-Vainio-Mattila, K.: Ux work in startups: current practices and future needs. In: International Conference on Agile Software Development. pp. 81–92. Springer (2015)
12. Klotins, E., Unterkalmsteiner, M., Gorschek, T.: Software engineering in start-up companies: An analysis of 88 experience reports. *Empirical Software Engineering* pp. 1–35 (2018)
13. Lacey, A., Luff, D.: Qualitative data analysis. Trent Focus Sheffield (2001)
14. Ladas, C.: Scrumban-essays on kanban systems for lean software development. Lulu. com (2009)
15. Law, E., Roto, V., Vermeeren, A.P., Kort, J., Hassenzahl, M.: Towards a shared definition of user experience. In: CHI’08 extended abstracts on Human factors in computing systems. pp. 2395–2398. ACM (2008)
16. Law, E.L.C., Lárusdóttir, M.K.: Whose experience do we care about? analysis of the fitness of scrum and kanban to user experience. *International Journal of Human-Computer Interaction* **31**(9), 584–602 (2015)
17. Olsson, H.H., Bosch, J.: From opinions to data-driven software r&d: A multi-case study on how to close the ‘open loop’ problem. In: Software Engineering and Advanced Applications (SEAA), 2014 40th EUROMICRO Conference on. pp. 9–16. IEEE (2014)
18. Owens, T., Fernandez, O.: The lean enterprise: How corporations can innovate like startups. John Wiley & Sons (2014)
19. Ries, E.: The lean startup: How today’s entrepreneurs use continuous innovation to create radically successful businesses. Crown Books (2011)
20. Schwaber, K., Beedle, M.: Agile software development with Scrum, vol. 1. Prentice Hall Upper Saddle River (2002)
21. Sutton, S.M.: The role of process in software start-up. *IEEE software* **17**(4), 33–39 (2000)
22. Sy, D.: Adapting usability investigations for agile user-centered design. *Journal of usability Studies* **2**(3), 112–132 (2007)
23. Yaman, S.G., Munezero, M., Münch, J., Fagerholm, F., Syd, O., Aaltola, M., Palmu, C., Männistö, T.: Introducing continuous experimentation in large software-intensive product and service organisations. *Journal of Systems and Software* **133**, 195–211 (2017)