

Integrating Service Design Prototyping into Software Development

Tanja Sauvola¹, Simo Rontti², Laura Laivamaa³, Markku Oivo¹, Pasi Kuvaja¹

¹M3S research unit, University of Oulu
e-mail: firstname.lastname@oulu.fi

²University of Lapland
e-mail: firstname.lastname@ulapland.fi

³e-mail: lsmeds@ulapland.fi

Abstract—Customer-driven service design is becoming an integral part of continuous software development. The fulfilment of needs is manifested through customer behaviour patterns that are often difficult to identify and validate for R&D. This paper investigates how customer involvement in software development can be achieved through experience prototyping. First, participatory action research with four cases is presented. As a result, the benefits, challenges and critical factors for successful service prototyping are identified. Second, a practical model is proposed for integrating service design as sprints within the software development process. Based on the study, the deployment of these methods can be adopted through an organisational culture that invests in the needed mindset, expertise, timing and placement. Contextual and motivating user involvement is important throughout the software development process. A number of important subjects that need further studies, such as service design performance measurement and customer data management, were also identified.

Keywords—customer involvement; service design; software development; experience prototyping

I. INTRODUCTION

Today, software is transforming almost all industries and is the main driver for innovation [1]. Elbert stated that all companies are in the software business either directly with IT solutions or indirectly with products in which software is critical for value creation, such as embedded systems, or value delivery, such as services. In today's highly competitive and fast-changing markets, software intensive industry is evolving towards a value driven and adaptive real-time business paradigm [2]. Hence, we live in a world of data overload, where any argument in product development can find supporting data. It is easy to find information to support our assumptions, but testing them with customers and then taking corrective actions is still hard [3]. Customer involvement and understanding customer needs are essential in software development in order to build successful products and services. According to Humble, after failing to deliver value for customers, the second largest risk in product development is building the wrong thing the right way and overinvesting in unproven opportunities. For this reason software companies need to continuously collect customer feedback and validate assumptions during the development process in order to build a product that is the best fit for customer needs [3][4][5]. However, customer

needs are often difficult to identify and they can change rapidly. Obtaining tacit and complex knowledge from customers is hard, as interacting and talking to customers may often be misleading, e.g., asking customers for information that they are not able to provide, such as what product to develop or technologies it should contain [6]. Agile methods [7][8] and Lean Startup [9] philosophy are addressing these issues by offering a range of techniques for more flexible ways of working. The aim is to produce a definition of a new service concept or the 'minimum viable product', which is to be implemented in markets as fast as possible with minimum effort, allowing us to measure how customers react and then validate findings.

Service design (SD) is an ascendant field of research where cultural, social and human interaction are connected [10]. The customer-driven SD approach aims at products and services that are useful, usable and desirable from the user's point of view and efficient from the service or product producer's point of view [11]. Service design has already taken place in the business-to-consumer (B2C) context, but it is also recognised as a useful approach in the business-to-business (B2B) context as well as in the internal development of organisations' processes [12]. More recently, a few process models and working practices, such as Lean UX [13], user story mapping [14] and design sprint models [15][16], have been introduced under both service design and user experience (UX) design titles and attempt to synthesise service design thinking, agile software development and lean start-up philosophies.

In this study, using four case projects from Finnish companies, we examine the role and impact of the experience prototyping methods of service design in software development contexts. Therefore, the main research question is:

RQ: How can service design be integrated into software development through a collaborative experience prototyping approach?

In exploring this we also ask:

- What are the benefits, challenges and critical factors of collaborative experience prototyping in software development?
- What is the position and role of service design in software development?

The contribution of the paper is twofold. First, we present a participatory action research study with four cases, where we identify the role of SD and benefits, challenges and

critical factors of experience prototyping in the software development context. Second, we outline a practical model for integrating SD in the software development process in order to increase customer insights and solve problems that are relevant for customers and thereby deliver value.

The rest of the paper is structured as follows. Section II presents the background and related work for this study. Section III presents study design and research method. In Section IV, we present and discuss our results from the empirical study and outline the findings in a practical model for integrating service design in software development projects. Finally, Section V concludes this paper and suggests topics for future research.

II. BACKGROUND AND RELATED WORK

A. Customer involvement in software development

Today, software is developed in rapidly changing and unpredictable markets, with complex and constantly changing customer requirements and the added pressure to shorten time-to-market [17]. Agile methods, which are widely adopted in the software industry [18], facilitate more flexible ways of organising software development activities in order to better meet the dynamic and unpredictable conditions in the business environment. Many different agile methodologies have been devised, such as Extreme Programming, Scrum, Kanban and Lean software development, which to some extent share the underlying mindset but use different implementations [2]. As stated by Nerur et al., agile methods are people-centric, recognising the value that competent people and their relationships bring to software development. In addition, agile methods focus on improving customer satisfaction through collaboration, active participation of relevant stakeholders and embracing change. The value of agile methods depends largely on an organisation's ability to learn. [19]

In the Agile software engineering literature [8][19], customer involvement is seen as the direction software companies should take to transform their practices throughout the development process. Typically, this is addressed by having a product owner represent the customer point of view [4]. However, recent studies show that even though ways of learning about customers are increasing, software companies often find it challenging to obtain timely and accurate feedback from customers to support research and development (R&D) decision-making processes continuously [4][5]. Customer involvement is studied widely in areas such as participatory design, user-centric design, usability engineering, human aspects of software engineering and requirements engineering [20].

Customer involvement is an abstract concept that refers to ways in which customers play roles in the software development process and the extent of their participation. Customer involvement is referred to as direct interaction using techniques based on active customer participation [21]. Through the years, a long list of practices and methods has been introduced to enable user participation and involvement. Although user participation seems to be a beneficial and well-understood approach in product

development, direct user involvement may not always be feasible. The situation is especially difficult in business markets when a wide physical or cultural gap exists between suppliers and customers and there can be multiple organisations and management layers between developers and users. The SD approach introduced in the next section offers a method to bridge this gap.

B. Service design approach

SD is a methodological approach, which can be used for customer involvement during the software development process. It is a holistic, multidisciplinary field that helps to innovate and improve existing products and services as well as make them more useful and desirable for customers [22]. Service design provides methods and tools for concretising and understanding user motivations and emotions during the development process for all the stakeholders involved. In B2B markets in particular, a large number of internal and external stakeholders must be considered, such as users, decision makers, developers, etc. The SD approach integrates the themes of a customer's emotions and experiences in the innovation process and concretises them for the benefit of value co-creation efforts [23].

Service design offers an outside-in-development approach, where products and services are developed holistically from customers' and end-users' point of view. In the B2B context, it means studying both the client business' and the end-users' processes, needs and wants. SD views the entire customer journey before, during and after the actual service in order to design the process fluently and support customers' goals [19]. Another key concept is the touchpoint through which the product or service is experienced; touchpoints are anything that can be designed in order to direct user experience in the desired direction [24]. This includes not only software user interfaces but also phone service, face-to-face communication, social media, signs, service premises, prints, physical tokens and their interconnection from the customer point of view. The touchpoint concept can also be looked at from the software development point of view (inside-out), as a link to customers through traditional product development phases: requirements, development, verification and validation and post-deployment where qualitative and quantitative data is collected [5].

The concept of co-creating value is defined by Srivastava and Verma as systematic and structured process based on collaboration with external stakeholders to generate value for the company as well as for the customers [25]. In the SD process, customers are not considered merely as feedback informants but as active participants from the beginning to the end of the process. In the process, customers may be targets of study via qualitative methods, such as interviews and observation, or customers can be asked to produce the customer data themselves using self-documentation methods, such as design probes [26]. Co-design is emphasised in SD, which refers to the process in which stakeholders are involved in concrete productive design tasks. These workshop sessions typically include collaborative prototyping and other means of expressing the information

needed in the design process facilitated by design professionals [27]. Recently, the workshop-based SD process has led to the compression of service design into a short but efficient design sprint as the pre-development phase in a lean and agile software development process [15][16].

According to Buchenau and Fulton Suri, experience prototyping is a key method in co-design sessions, serving as an efficient medium for concretising and empathising with customer insight. The aim of experience prototyping is to represent and prototype different design concepts and ideas. Prototypes are defined as 'representations of a design made before final artefacts exist'. To them the term 'experience design' consists of methods that make it possible for designers, customers and users to reach a common understanding of the forthcoming results (products or services) of the ongoing project. Experience prototypes may include suggestive staging; 'quick and dirty' card mockups, taking the roles of both customer and service provider and enacting service situations. Experience prototyping has three roles in the design process: understanding customer insight findings, exploring new ideas and communicating concepts to others. [28]

Pinheiro lists three main goals for how early experience prototypes can be used in a project: '(1) set the context for users to participate in idea generation and co-design, (2) service enactment, or role-play, to explain or learn from a complex concept and (3) test to validate specific service interactions or the entire service journey'. The experience prototype can serve as an earlier, more inexpensive and efficient version of the minimum viable product (MVP) emphasised in Lean Startup philosophy. [15]

Several methods and facilities have been developed in order to enhance the facilitation of co-design and holistic experience prototyping. The Service Innovation Corner (SINCO) is a service prototyping environment concept developed at the University of Lapland [29]. SINCO consists of the environment and a set of tools for co-design and experience prototyping. In SINCO, technological equipment and digital material, such as photos, videos, and sounds, are used to create the atmosphere of actual service moments for prototyping and re-enactment. The SINCO set-up has two

117' background projection screens perpendicular to each other to provide the background scenery and enable partial yet immersive spatiality [30]. SINCO methods and tools were applied in this research when conducting co-creation workshops with the case companies. Fig. 1 summarises the research setting and the interrelations of the key concepts of this paper.

III. RESEARCH DESIGN

In this section, research methods and settings, data collection activities and analysis are presented.

A. Research method

The aim of this paper is to study the role and impacts of experience prototyping methods of service design in the software development context. We apply the multiple case study approach, which adopts an interpretive approach [31]. The multiple-case study is suitable for this study because it allows the researchers to study the phenomenon in a real life setting as well as a cross-case analysis of the data. The general research framework for this study can be characterised as participatory action research (PAR), as described by Whyte, with the research activities taking place in empirical context advancing both science and practice [32]. PAR is a social process involving practitioners in the research from the initial design of the project through data gathering and analysis to final conclusions and actions arising out of the research. Methods to achieve the research goal include end-user interviews, role plays and data and information visualisations. They also include experience prototyping in the SINCO environment where both customers' and end-users' tangible and intangible needs and wants were examined with scenarios including images, video and audio material. The approach to these methods is also PAR, in that it studies a situation and a set of problems to determine what the facts through self-reflective iterative cycles. This can also be described as a co-learning process between researchers and workshop participants resulting to organisational learning and change [32][33].

B. Case companies

The study uses empirical data from four case companies in Finland. We refer to the companies as Company A, B, C and D. The four companies were selected using convenience sampling from a group of Finnish leading-edge companies participating in two large national research programmes. Company A is developing embedded software solutions for specialised markets in the wireless and automotive industry. It also provides B2B product development services and customised solutions for wireless communications. The focus of the case was to increase user insights of special devices used in specialised market segments, such as public safety. Company B is an SME operating in metal industry, manufacturing hydraulic cylinders and offering cylinder maintenance services for big industry clients. The focus of the project in the company was to develop a digital maintenance service. Company C operates in the software industry, offering a variety of IT solutions focusing on data security operations, mainly in B2B and B2C markets.

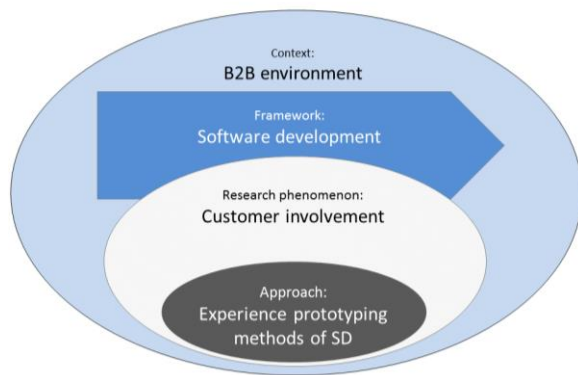


Figure 1. They key concepts and the research setting of this paper.



Figure 2. Example of the SD workshops: Experience prototyping public safety communication use cases in the SINCO.

Company D is a multinational bank operating on both B2B and B2C markets. The focus of the case project was to analyse and improve online banking services.

C. Data collection and analysis

Empirical data was collected from December 2013 to March 2014 (Companies B and D) and from November 2014 to September 2015 (Companies A and C), using semi-structured interviews with open ended questions [n=11], workshops [n=12], group discussions [n=10], field diaries by the service design team during workshops [n=13] and secondary data [n=29]. The interviewees and workshop attendees were selected by a key contact person from each company who was asked to nominate experts from various departments, such as product management, R&D, validation and verification, sales and marketing and, in some cases, contact centre and customer counter functions. The interview guide allowed us to conduct the interviews in the form of a discussion with each interviewee that lasted approximately 1–2 hours.

The workshops (Fig. 2), besides being an integral part of the service design sprints, can be considered as participative action research cycles, producing data about the research phenomenon [31]. In the workshops, SD was turned into action through service design sprints affiliated to the company's existing R&D, business development or marketing processes. All workshop participants had a lot of experience of working in the company on multiple projects. Workshops lasted approximately between 3–5 hours and were facilitated by 1–3 service designers. Workshops and interviews were video and audio recorded and transcribed for

analysis. In addition, participatory observation was used as a research method in this study. These field diary notes were important, as the emerged emotions during the workshops, such as frustration, anger or laughter, were observed and documented carefully. In addition, secondary data was collected from the case companies: (1) process models and other documents, (2) both individual and group interviews of the potential end-users and (3) material from workshops and presentations.

In our study, we assess three aspects of validity, i.e. construct validity, external validity and reliability, as identified by Runeson et al. [30]. Prior to data collection, the research design that also included the data collection process was carefully considered. The activity involved selecting appropriate companies and roles for the interviews and providing all interviewees with introductory materials (e.g., study objectives, the structure of the workshops and interviews, etc.). Threats to the reliability of the study findings were mitigated by three researchers involved in all the phases of the research process. In particular, data collection and analysis was performed in continuous collaboration following the general techniques for case study analysis and used the QSR NVivo tool¹. During the analysis, all materials, including transcripts, field notes, audio and video files and other related material, were stored in NVivo. All transcribed interviews were carefully read and coded by themes. The results were produced by looking for themes related to the research question.

¹ www.qsrinternational.com

IV. RESULT ANALYSIS AND DISCUSSION

Based on the empirical findings, this section presents and discuss the identified benefits, challenges and critical factors of service prototyping in a software development context (Tables I, II and III). Additionally, a practical model for integrating service design sprints into the software development process is outlined and discussed (Fig. 3).

A. Experience prototyping in software development

The use of SD experience prototyping methods has benefits and challenges. The findings from each case study company show, that at their best, these methods can nourish and support innovation and development culture. The most important offering of the workshops, according to many of the participants, was the liberalisation of mindsets. Experience prototyping methods also allow for the efficient constitution of a complete understanding of all stakeholders' viewpoints. These methods were seen as pleasant and motivating but also challenging because participants had to step out of their comfort zone. According to workshop participants, they were able to develop significant knowledge, skills, and emotions by experience prototyping in an emergent process that empowered people to engage in discovery, reflection and even action (Fig. 2). Testing of business hypotheses and assumptions in product development could be then turned into faster and more accurate decision-making and value creation. Table I categorises and describes the identified benefits and Table II categorises and describes the identified challenges of experience prototyping based on the research data. In Table III, we also identified critical factors that need to be considered in order to successfully implement experience prototyping within company processes.

TABLE I. IDENTIFIED BENEFITS OF EXPERIENCE PROTOTYPING

Benefit	Description
Improved Communication	SD methods help to improve communication between all the stakeholders (e.g., management, sales, development team, customer and end-users). Collaborative prototyping sessions also increase transparency in the organisation by uncovering grassroots knowledge to be exploited in development even on strategic and business level.
Instant feedback	Presents an opportunity to get instant and direct feedback from the end-users.
Increased motivation and innovation	SD methods motivate development teams, customers and end-users to innovate, co-develop and more actively participate in the development process. SD methods also support and sustain innovation process.
Mindset change	It brings out the user oriented mindset and changes the viewpoint from insight-out to outside-in. Customer journey walkthrough and empathising the customer role inherently leads to customer-centric development approach (outside-in) instead of just pushing new technology based features (inside-out).
Learning and decision making	This accelerates the decision-making, e.g., via more efficient distribution and understanding of the

Benefit	Description
	information. In the process, tacit knowledge is converted to tangible knowledge. This is an opportunity to find behaviours and patterns about which users are not aware. Learning process with different levels: individual, team and organisational.
Identification and prioritisation of features or potential market segments	SD methods help stakeholders to identify and prioritise features as well as avoid building unnecessary functions based on deeper understanding of end-user needs. It may also help to identify new potential products/services or market segments and even reduce time to market.
Value creation	Value can be created from intangible end-user experiences and emotions. Quick and cost effective way to test new ideas before any development work is done.

TABLE II. IDENTIFIED CHALLENGES OF EXPERIENCE PROTOTYPING

Challenge	Description
Facilitation	Special skills and environment are needed to facilitate SINCO workshops. Companies do not have the premises nor facilitation capabilities.
Stakeholder availability	Due to increasing demands and hectic business schedules, it may be challenging to find suitable time for all stakeholders to participate in a workshop at the same time. In the B2B context, involving end-users may be challenging.
Measurement and data management	Lack of systematic ways to collect customer data and identify metrics for how to measure customer value and no practices for documenting and integrating workshop results back to the existing processes. Video was identified as an effective medium to document and communicate experience prototypes but companies' IT systems don't necessarily offer suitable ways to store and tag videos.
Timing and placement of the workshop	Proper timing and placement of the workshop with the company's process. In two of the cases, the involvement of end users and the whole SD sprint took place too late in the development process.

TABLE III. IDENTIFIED CRITICAL FACTORS OF EXPERIENCE PROTOTYPING.

Critical factor	Description
Service design expertise	In-house and outsourced service design expertise is utilised. In-house service design expertise is needed to pursue the co-creation model forward and to be able to facilitate and/or purchase facilitation on demand. The outsider view was needed to help companies apply an outside-in customer-centric development approach.
Preparation	Understanding the business challenge and the context is critical. Preparations phase consists of brief from the case company, collecting information (benchmarking, observations, mystery shopping etc.) and information analysis for scenarios formation. Defining which information will be collected as the result of the workshop session, and how.

Critical factor	Description
Facilitation	The facilitator must be able to direct the participants and experience prototyping session as well as rapidly build mock-ups during the session. Facilitator defines the point of view through which matters are analysed. Also, proper rhythm of different collaboration modes: presenting background information, enacting and gathering and analysing insights unveiled in the drama (role play) are important.
Involvement of real customers and end users	Involving real users to both the customer insight phase (in workshop and during the preceding field study) and concept testing is critical in order to validate the findings.

Our results indicate that, typically in the B2B context, there is no direct interface between the development team and users. Often this is due to the intermediaries in the supply chain. We also noticed that even if the link between the development team and the users does exist, this opportunity is rarely fully utilised, and typically users are involved too late in the process. For development teams, this may lead to a situation where it is difficult to understand the reasons behind the requirements and validate which features bring real value. Experience prototyping workshops acted as a tool for innovation, communication and interpreter emphasising stakeholder experiences, thus allowing different points of views to be discussed. It provided instant and detailed insight about end-user motivations in different situations and the possibility to test various different service or development options. In experience prototyping workshops, involving real users is necessary in order to enable deep customer insight efficiently as well as test and validate the findings. By observing the workshop sessions, we noticed that sessions that were too long are challenging and may become a tiresome activity, especially when the methods used were unfamiliar to most of the participants. Therefore, it is important to balance the time between the

workshops and discussion activities and provide clear instructions to the attendees.

Essentially, the challenge for companies today is no longer how to solve technical issues but rather how to solve problems that are relevant for customers and end users. This requires changing the company culture and mind-set from 'insight-out' technology and features first thinking towards more customer-centric 'outside-in' approach.

B. Integrating service design sprint into software development processes

The role of SD varied with to R&D phase and the lifecycle point of the software being developed. The roles and the positioning of the service design sprints against software development process in the case companies is presented in Table IV.

Through the findings from the company cases, we identified that SD experience prototyping could to be considered as a new development strategy. In all cases, the companies' company culture was identified as a key factor to support change and encourage constant learning. SD methods provide one set of tools for gaining this lacking user knowledge. As a result of our study, we present a practical model of how to introduce SD experience prototyping into Agile Scrum process as sprints (Fig. 3.). In the model, user knowledge is used for continuous learning during the Scrum process, which can be used to test, validate and prioritise features, update the product roadmap, improve the product or service and ultimately result in better customer satisfaction. The model builds on the possibility of learning and executing small tasks that are delivered as an MVP to customers.

In the proposed model, the SD experience prototyping sprints take place in three phases of the agile software development process. The first SD sprint focuses on customer insight and analysing the customer journey through the holistic service experience in which the software product is part. The involvement of real customers and end-users is crucial during this first SD sprint. The first SD sprint results in the product backlog of the MVP or script for the minimum

TABLE IV. THE ROLES AND THE POSITIONING OF THE SERVICE DESIGN SPRINTS AGAINST THE SOFTWARE DEVELOPMENT PROCESS

Company	Role of SD sprint	Positioning of SD sprint
A	Novelty customer involvement platform to validate the holistic user experience of a communication device and its ecosystem	Indirect link to ongoing software development, complementing product backlog, validating overall product/service quality (software development being one sub problem of the design, manufacturing and delivery of the electronic device/platform under development)
B	Pre-development tool for a new maintenance service concept. Providing a service vision and come up with a roadmap for IT service development.	SD sprint as a pre-development phase of software development, online-software concept/backlog as a core deliverable of the SD sprint
C	Make corporate internal operations and knowledge visible through gamification and prototyping.	Pre-development phase, online service portal concept/mockup being a central deliverable of SD sprint.
D	Validate and finish the user experience of new service concept consisting of new online channels already being in implementation phase.	Indirect link to already ongoing software development, complementing product backlog, validating overall product/service quality

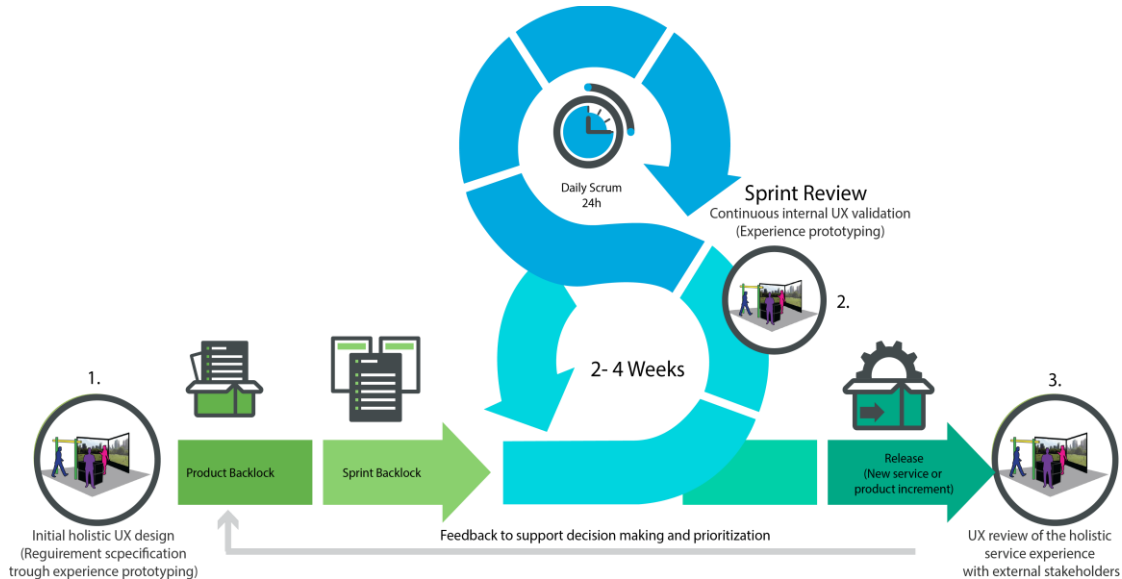


Figure 3. Service design sprint as a part of the Agile Scrum process

valuable service. The second position, where holistic user experience prototyping takes place, happens during the software development sprint as company's internal holistic user experience (UX) check point tool. The aim is to validate and integrate the developed individual software features into a common vision of the holistic outside-in service experience at regular intervals. The third experience prototyping sprint takes place before actual commercial launch of a service. Depending on the case, this sprint may have different foci. The first is evaluating the product or service concept with customers and other external stakeholders in order to accept and refine the critical points of the customer journey before deployment. The second purpose is to educate relevant stakeholders (e.g., sales, marketing and support) of the software use cases and customer experience related sales arguments. The responsibility of coordinating the SD sprints falls naturally to the product owner. The actual facilitation of the experience prototyping sessions requires hiring or purchasing special service design expertise, or it may fall under UX designers' responsibility as new expertise with the need for education and training.

V. CONCLUSION

Users expect nothing less than great products that are easy to use and bring value regardless of technology, platform or context. Delivering value in constantly changing markets and meeting customer needs are key success factors for any business. The objective of this paper was to study the role and impact of experience prototyping in software development projects, including its benefits, challenges and critical factors.

Our study shows that during the case projects, the role of SD was to concretise customers' and end-users' needs to internal stakeholders and to innovate, validate and create new product or service concepts holistically. The methods to achieve the aim included end-user interviews, followed by workshops and role plays, data and information

visualisations as well as experience prototyping during the workshops in the SINCO environment. In general, SD methods enhance the software development process and benefit both the developers and users by enabling companies to develop customer-centric products and services that are useful, desirable and competitive in the market. We identified a number of benefits of experience prototyping in the software development context, such as instant feedback, faster and more accurate decision-making, continuous learning as well as focusing the development effort on things that bring value to users are some of the perceived benefits. Furthermore, we identified challenges and critical factors that could block the use of SD methods during development activities, such as stakeholder availability, special skills and the environment needed for workshop facilitation, etc. Based on the results, we present a practical model of how to integrate SD prototyping sprints in software development processes. The model builds on existing software development practices with short learning and iteration cycles, where customer experience can be improved by arcing out the situation, making quick improvements and prototyping the experience again. In the proposed model, the SD experience prototyping sprints take place in three phases (before, during and after) of the agile software development process.

It is important to note that due to the methodological nature of our research, generalisation based on the results is inherently limited. However, our research results offer a fruitful ground for future studies on using experience prototyping or other SD methods in software development practices. Our future research will aim to validate the proposed model in an empirical context. For future research, it could be important to identify mechanisms that can be used to analyse and incorporate workshop results into software development processes and identify metrics for analysing customer value.

ACKNOWLEDGMENT

This work was supported by TEKES (Finnish Fund for Technology and Innovation) as part of the 'Need for Speed' project (<http://www.n4s.fi/>) and 'Value through Emotion' project. This work has been done in co-operation with research group (M3S) from the University of Oulu and University of Lapland.

REFERENCES

- [1] C. Elbert, 'Looking into the future'. *IEEE Software*, vol. 32, no. 6, pp. 92–97, 2015.
- [2] J. Järvinen, T. Huomo, T. Mikkonen and P. Tyrväinen, 'From Agile software development to Mercury business' In *Software Business. Towards Continuous Value Delivery*, Cyprus: Springer, vol. 182, pp. 58–71, 2014.
- [3] J. Humble and J. Molesky, *Lean Enterprise: How High Performance Organizations Innovate at Scale*, CA, USA: O'Reilly, 2015.
- [4] H. H. Olsson and J. Bosch, 'Towards continuous customer validation: A conceptual model for combining qualitative customer feedback with quantitative customer observation', in *Software Business: 6th International Conference, ICSOB*, Portugal, pp 154-166, 2015.
- [5] T. Sauvola, L. E. Lwakatare, T. Karvonen, P. Kuvaja, H. H. Olsson and J. Bosch, 'Towards customer-centric software development, a multiple-case study', in *41st Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, Portugal, pp 9-17, 2015.
- [6] A. Griffin, 'Obtaining customers' needs for product development', in K.B. Kahn, *The PDMA Handbook of New Product Development*, 3rd Ed., Hoboken, NJ: Wiley, pp. 213–230, 2013.
- [7] J. Highsmith, *Agile Project Management: Creating Innovative Products*. New York, NY: Addison-Wesley, 2009.
- [8] T. Dybå and T. Dingsøyr, 'Empirical studies of agile software development: A systematic review', *Information and Software Technology*. vol. 50, pp. 833–859, 2008.
- [9] E. Ries, *The Lean Startup. How Constant Innovation Creates Radically Successful Businesses*. UK: Penguin, 2011.
- [10] S. Miettinen and A. Valtonen, (Eds), *Service Design with Theory. Discussions on Change, Value and Methods*. Rovaniemi, Finland: Lapland University Press, 2012.
- [11] E. Kuure and S. Miettinen, 'Learning through action: Introducing the innovative simulation and learning environment Service Innovation Corner (SINCO)', presented at World Conference on E-Learning, 21-24 October, Las Vegas, NV, USA, pp. 1536-1545, 2013.
- [12] B. Reason, 'Mind your own business. Service Design in a B2B'. *Touchpoint Journal, Service Design Network*, vol. 1, no. 3, January 2010.
- [13] J. Gothelf and J. Seiden, *Lean UX. Applying Lean Principles to Improve User Experience*, USA: O'Reilly, 2013.
- [14] J. Patton, *User story mapping. Discover the whole story, build the right product*. Sebastopol, CA, USA: O'Reilly, 2014.
- [15] T. Pinheiro, *The Service Startup. Design Gets Lean*. USA: Hayakawa, Altabooks and Createspace, 2014.
- [16] J. Knapp, J. Zeratsky and B. Kowitz, *Sprint. How to solve big problems and test new ideas in just five days*. New York, NY, USA: Simon & Schuster, 2016.
- [17] G. G. Claps, R. B. Svensson and A. Aurum, 'On the journey to continuous deployment: Technical and social challenges along the way', *Information and Software Technology*, vol. 57, pp. 21–31, 2015.
- [18] P. Rodríguez, J. Partanen, P. Kuvaja and M. Oivo, 'Combining lean thinking and agile methods for software development: a case study of a Finnish provider of wireless embedded systems detailed', in *47th Hawaii International Conference on System Sciences, IEEE*, pp 4770 – 4779, 2014.
- [19] S. Nerur and V. Balijepally, 'Theoretical reflections on agile development methodologies'. *Communications of the ACM—Emergency Response Information Systems: Emerging Trends and Technologies*, vol. 50, no. 3, pp 79-83, 2007.
- [20] U. Abelein and B. Peach, 'Understanding the influence of user participation and involvement on system success—A systematic mapping study,' *J. Empirical Software Engineering*, vol. 20, pp. 28–81, 2013.
- [21] S. G. Yaman, T. Sauvola, L. Riungu-Kalliosaari, L. Hokkanen, P. Kuvaja, M. Oivo and T. Männistö, 'Customer involvement in continuous deployment: a systematic literature review', in *Requirements Engineering: Foundation for Software Quality, 22nd International Working Conference, REFSQ*, Sweden, pp 249–265, 2016.
- [22] M. Stickdorn and J. Schneider, *This Is Service Design Thinking: Basics, Tools, Cases*. New York, NY, USA: Wiley, 2012.
- [23] S. Miettinen, S. Rontti and J. Jeminen, 'Co-prototyping emotional value', in *19th DMI: Academic Design Management Conference. Design Management in an Era of Disruption*, London, UK, pp. 2-4, 2014.
- [24] J. Howard, 'On the origin of touchpoints. design for service research, patterns and observation'. [Online]. Available: <https://designforservice.wordpress.com/2007/11/07/on-the-origin-of-touchpoints/>
- [25] R. M. Srivastava and S. Verma, *Strategic Management. Concepts, Skills and Practices*, New Delhi: PHI Learning, 2012.
- [26] T. Mattelmäki, *Design Probes*. Finland: University of Art and Design Helsinki, 2006.
- [27] Y. Lee, 'Design participation tactics: The challenges and new roles for designers in the co-design process'. *CoDesign—Internal Journal of CoCreation in Design and the Arts*, vol. 4, no. 1, pp. 31–50, March 2008.
- [28] M. Buchenau and J. Fulton Suri, 'Experience prototyping', in *Proc. 3rd Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques (DIS '00)*, ACM, USA, pp. 424-433, 2000.
- [29] S. Rontti, S. Miettinen, E. Kuure and A. Lindström, 'A Laboratory Concept for Service Prototyping—Service Innovation Corner (SINCO)', in *Service Design and Innovation Conference (ServDes 2012)*, Helsinki, Finland, pp. 229-241, 2012.
- [30] S. Rontti, S. Miettinen, E. Kuure and A. Lindström, 'Agile techniques in service prototyping', in *Service Design with Theory. Discussions on Change, Value and Methods*, S. Miettinen and A. Valtonen, Eds. Rovaniemi, Finland: Lapland University Press, pp. 189–196, 2012.
- [31] P. Runeson, M. Höst, A. Rainer and B. Regnell, *Case Study Research in Software Engineering: Guidelines and Examples*. NJ: Wiley, 2012.
- [32] W. F. Whyte, *Participatory Action Research*, USA: Sage, 1991.
- [33] J. M. Chevalier and D. J. Buckles, *Handbook for Participatory Action Research, Planning and Evaluation*. Ottawa: SAS2 Dialogue. [Online]. Available: http://www.participatoryactionresearch.net/sites/default/files/sites/all/files/manager/Toolkit_En_March7_2013-S.pdf, 2013