Empirical Software Engineering (2023) 28:115
https://doi.org/10.1007/s10664-023-10331-w

An empirically based model of software prototyping: a mapping study and a multi-case study

Elizabeth Bjarnason[1] · Franz Lang[1] · Alexander Mjöberg[1]

Accepted: 19 April 2023 / Published online: 30 August 2023

Abstract

Context Prototyping is an established practice within product and user interface design that is also used as a requirements engineering (RE) practice within agile development. Even so, there is a lack of theory on prototyping.

Aims Our main research objective is to support practitioners in improving on their prototyping practices.

Method We have designed a model that describes key aspects of the practice of prototyping. The model is based on a systematic mapping study consisting of thirty-three primary studies and on empirical data from twelve case companies. We validate and demonstrate the applicability of our model through a focus group at one company and through semi-structured interviews at eleven (other) startup companies.

Results Our prototyping aspects model (PAM) consists of five aspects of prototyping, namely purpose, prototype scope, prototype media, prototype use, and exploration strategy. This model has enabled practitioners to discuss their prototyping practices in terms of the concepts provided by our model.

Conclusions The model can be used to categorise prototyping instances and can thereby support practitioners in reflecting and improving on their prototyping practices.

Keywords Prototyping · Agile · Requirements engineering · Systematic mapping study · Case study · Practices

Communicated by Maria Teresa Baldassarre, Markos Kalinowski.

This article belongs to the Topical Collection: *Special Issue on International Symposium on Empirical Software Engineering and Measurement (ESEM)*

✉ Elizabeth Bjarnason
elizabeth.bjarnason@cs.lth.se

Franz Lang
franz.langg@icloud.com

Alexander Mjöberg
alexander.mjoberg@gmail.com

[1] Department of Computer Science, Lund University, Lund, Sweden

1 Introduction

Prototyping is a creative practice commonly applied within product design (Tronvoll et al. 2017), usability and user-interface design (Hakim and Spitzer 2000; Lauesen 2005; Hartson 2019), and software development (Acosta et al. 1994, 1993; Goldman and Narayanaswamy 1992) to explore the problem and/or the solution domain through use of a prototype, i.e. an early sample, model, or release of a product that simulates one or many dimensions of the (future) product. A prototype can range from a simple paper sketch through a computer-generated mock-up to an (incomplete) version of the production software, e.g. a minimum viable product (MVP), and can be used to learn about user problems and to evaluate solution ideas (Sergio 2015). Thus, prototyping enables validating a solution proposal before developing the full product through cost effective testing with real users (Nielsen 1993). In our research, we focus on the practice of prototyping, namely the use of prototypes to obtain learning, rather than on the construction of prototypes, meaning that any representation that aids in exploring and learning about a feature or dimension of a new product or business model can be used for prototyping. This broad definition of a prototype, also includes entities such as PowerPoint sketches and videos (Karras et al. 2017), e.g. used as marketing material, and interview questions (Batova et al. 2016), e.g. used to explore the customer and user domain.

Within human–computer interaction (HCI) and design, the practice of prototyping is well established, and used to explore and test ideas and solutions for user interface designs. However, researchers within HCI pointed out a lack of knowledge about the fundamental nature and characteristics of prototyping, already in 2008 (Lim et al. 2008). While there is more recent research on prototyping, we find no comprehensive theory of the practice or its methodology, e.g. overarching principles or procedures for achieving a certain outcome, and only one related literature review of prototyping concerning the definition of MVP (Lenarduzzi et al. 2016). For these reasons, we were interested in exploring prototyping methodology with the aim of supporting practitioners in describing and discussing their prototyping practices, and thereby enable them to improve on these through reflection-based learning (Bjarnason et al. 2014).

The overall objective of our study was to explore how to categorise prototyping practices from the perspective of prototyping methodology. The initial part of our study was performed in collaboration with our case company Telavox, who were interested in further improving their use of prototyping. We based our research on the current body of knowledge and defined the following four research questions to guide our research:

RQ1: What are the main aspects of prototyping methodology?
RQ2: What fields of research have previously investigated the main aspects of prototyping methodology (RQ1)?
RQ3: What types of research have investigated the main aspects of prototyping methodology (RQ1)?
RQ4: How can the initial version of our prototyping aspects model be improved to better support practitioners?

In this paper, we extend on previously published results for RQ1 (Bjarnason et al. 2021a) by presenting a revised version of our *prototyping aspects model* (PAM) including improvements to the initial version of the model (RQ4). These improvements are based on additional empirical data from a multi-case study of eleven startup companies. The initial version of PAM was based on a systematic mapping study (RQ2 and RQ3) and was validated through a focus group at one case company. Herein, we also present results for RQ4 based on semi-structured interviews with twelve practitioners where PAM was used to discuss and categorise forty-three prototyping instances applied within the case companies. This additional empirical material and analysis enables us to improve on PAM and provide a revised answer to RQ1. Initial results on the prototyping practices of startups are published in Bjarnason (2021) based on four of the twelve interviews. This article is based on the full set of interviews and focuses on validating and improving PAM rather than on describing the overall prototyping practices of the companies, as in Bjarnason (2021).

Our prototyping aspects model (PAM) may be used to characterise and compare prototyping instances, e.g. regarding the scope and media of a prototype, how it is reviewed with users, and how this affects the cost–benefit balance of using the practice of prototyping for exploring the problem domain and developing software products. We believe that the additional refinements of the model presented in this paper, further improve the applicability of PAM and the model's usefulness for practitioners.

The rest of this article is organised as follows. We describe related work on prototyping in Section 2. In Section 3, we outline our research method, and in Section 4, we describe the case companies. The model of prototyping (PAM), which is our main contribution, is presented in Section 5, in response to RQ1, based on previous work on prototyping and on our findings from the multi-case study. Section 6 contains results on RQ2-RQ4 that are discussed in Section 7, before concluding in Sect. 8.

2 Related work

While our point of departure and main area of competence is requirements engineering (RE), the research presented in this paper is based on previous work on prototyping within software engineering, RE, as well as, within human factors, usability and user-interaction design. Furthermore, prototyping is commonly used within software startups and in the early stages of software development to explore and validate user needs and requirements. In this section, we provide a brief introduction to related work on prototyping within human–computer interaction and design, agile RE, and for software startups. More detailed references are provided in Section 5, as part of describing our prototyping aspects model.

2.1 Prototyping within human–computer interaction and design

Within product- and human–computer interaction design, prototyping is commonly used to design and to evaluate the user interface by "*producing or building a model or mockup of a design that can be manipulated and used … to simulate a user experience*" and thereby test this experience without having *"to build the real thing"* (Hartson 2019). While prototyping is mainly used to support design and evaluation within usability and user interaction design, several other advantages of the practice are described including its role in supporting communication and creativity. Prototypes can facilitate communication of design ideas by providing *"a concrete basis for discussions"* between developers, designers, and users (Budde and Zullighoven 1990). As such, prototypes "*serve as a vehicle that enables users and designers to develop a common language*" (Hakim and Spitzer 2000). Thus, the use of prototypes can stimulate user involvement, and support marketing and selling new product ideas to customers and to (internal) management. Many of these benefits and uses of prototyping are covered in our model by the aspect of *Purpose* (see Section 5.1*)*.

Prototyping can be performed throughout the design process starting with simple sketches that evolve into wireframes and into gradually more complete representations of the design. The variation in prototype richness and scope was characterized by Nielsen using the terms breadth and depth (or detailing) of functionality represented by the prototype, where a *horizontal* prototype provides a broad set of features but with a low degree of refinement (depth), while a *vertical* prototype can provide detailed (deep) representation of one feature only, i.e. a narrow breadth (Nielsen 1993). Other variants of prototype scope include local prototypes that focus on specific issues and are thus both shallow and narrow, and T prototypes that represent a broad set of features but only details (depth) for a few parts (Hartson 2019). We base one of the aspects of our model (*Scope*) on this terminology and include the dimension of breadth, while expanding the dimension of depth to distinguish between the refinement of one or more facets, namely functional, visual, interactive and data, see Section 5.2.

The term *fidelity* is used to indicate how close to the final product the prototype is with regards to appearance and interaction (Tullis 1990). Low- versus high-fidelity prototypes are often used for different purposes within the design community. Low-fidelity prototypes can convey a general look of an interface and are often used to communicate, educate and inform, while high-fidelity prototypes are more expensive to build they can be used to test and evaluate further details of the design, and even serve as a basis for development of the product under design (Rudd et al. 1996). In our model, fidelity is represented by a combination of the two aspects Scope (see Section 5.2) and Prototype Media (see Section 5.3).

Research on the use of prototyping in generating new user interaction designs found that prototyping multiple design in parallel and sharing these led to more creative and better final designs (Dow et al. 2011). In our previous research on the use of prototyping in startups, we found that very few startups work with parallel exploration and prototyping, and that those who did work with parallel prototypes tended to have a background in usability and human-interaction design (Bjarnason 2021). This dimension of the practice of prototyping is covered in our model by the aspect of *Exploration Strategy*.

Finally, prototypes can be used as a means to present/demonstrate a product idea, or to evaluate a design through allowing users to interact with the prototype. Such interactions often occur in a meeting or lab settings, e.g. during usability testing, either through free testing or by providing scenarios or other protocols for the user to follow (Tronvoll et al. 2017). Evaluating prototypes in “the wild”, i.e. in the environment of the final product, provides a more realistic setting for evaluating a prototype both in terms of the actual physical environment and w.r.t. the people available (Hendry et al. 2005). These dimensions are covered in our model by the aspect of Prototype Use, see Section 5.4.

2.2 Prototyping in agile requirements engineering

Prototyping has been identified as a practice commonly applied within agile software development to manage rapidly evolving requirements by the practices’ ability to support customer communication, and for validating and refining requirements (Ramesh et al. 2010). Käpayaho and Kauppinen report from a case study of using prototyping for user interface development at a large retail company applying an agile approach. Their findings indicate that prototyping addresses several challenges within agile but also identify a need to complement prototyping with additional practices. They observed that prototyping helped with challenges related to managing with very little requirements documentation, such as intangible and unaligned views and plans on what to develop. Furthermore, the quality of stakeholder communication at the case company was increased through prototyping and mutual understanding between the development team and the customers was reached faster. Prototyping also increased the motivation for requirements work since updating a prototype was considered more motivating than writing requirements documents (Käpyaho and Kauppinen 2015).

The benefits of using prototypes in the customer communication have also been observed by several other researchers including Ramesh et al. and Zink et al. Illustrating and communicating product requirements through a prototype reduces risks related to low requirements specification quality (Ramesh et al. 2010). As the prototype representation of the (future) product becomes more concrete with each iteration, the product requirements thereby gradually become more specific (Zink et al. 2017). In particular, an executable prototype provides a rich means of communicating requirements details, and reduces the risk of ambiguity, incompleteness and inconsistency in the requirements communication (Acosta et al. 1994).

While prototyping provides benefits to agile development, the practice also imposes risks. In particular, research shows that the use of production software in prototyping (rather than throw-away prototypes) incurs risks related to product quality such as scalability, security, and robustness (Ramesh et al. 2010). In this case, demonstrating a fully functioning (but early) version of production software may convey an overly positive view of development status to customers and other stakeholders, that in turn may lead to a push to release software prematurely, before sufficient quality has been achieved. Agile projects need to be aware of these risks and apply other practices to mitigate these. Käpayaho and Kauppinen suggest complementary practices such as clearly stating customer responsibilities, management of quality requirements, and consideration of the bigger picture (Käpyaho and Kauppinen 2015).

2.3 Prototyping in software startups

Software startups are new companies that develop novel software-based products or services, and that commonly apply prototyping (Bjarnason 2021) for exploring and evaluating new ideas and technology in a quick and relatively cheap way (Lauesen 2005), and thus, enables cost effective testing with users (Shepherd and Gruber 2020). This cost–benefit aspects is especially important to software startups who operate under very uncertain and resource-constraint conditions with the aim of exploring new business opportunities and develop innovative products (Giardino et al. 2014; Paternoster et al. 2014). While availability of open-source software and pay-as-you-go services provide software-based business opportunities, startups struggle to define solutions for which there is product-market fit and risk wasting time and resources on developing unsuccessful features (Giardino et al. 2015). One important success factor is to test the business idea early on to validate its viability in the market (Block and MacMillan 1985).

Software startups typically perform light-weight, informal and ad-hoc requirements engineering, in particular during the early stages of the startup venture due to limited resources (Giardino et al. 2014; Klotins et al. 2019; Nguyen-Duc et al. 2017; Terho et al. 2016). Requirements are initially elicited and prioritised mainly based on internal sources and on problems experienced by founders. The source of requirements gradually shifts as potential customers are identified, and prototypes are produced that can be used to elicit new ideas and priorities also from customers and other external parties (Nguyen-Duc et al. 2017; Terho et al. 2016). Tripathi et al. provide similar findings albeit with some more details and based on a broad coverage of literature and cases (Klotins et al. 2019).

Within startups, an early version of the final product is often used to validate product ideas with users (Alves et al. 2020; Tripathi et al. 2018), e.g. by demonstrating mock-ups to customers, as a cost-effective way to obtain market feedback. Thus, prototyping is used as a means to learn about the user and the market (Giardino et al. 2015; Paternoster et al. 2014), and to increase the chances of business success. However, the use of a live product version as a prototype poses conflicts between the need for quick feedback and a focus on product quality (Ciriello et al. 2017). One case study of prototyping in twenty Norwegian startups, by Nguyen-Duc et al., identified factors that affect the speed of prototyping, including the choice of prototype tools and components, varying competences, and the communication and involvement of customers and external stakeholders in the prototyping. They observed that the purpose of prototyping and how the prototype is used regarding customer involvement are factors that need to be considered when selecting prototyping practices (Nguyen-Duc et al. 2017).

In an earlier publication, we reported on the prototyping practices of four early-stage startups (Companies A-D, of this article) to understand how they use prototyping to elicit, prioritise, validate, and communicate requirements. In that study (Bjarnason 2021), we found that *prototyping is commonly used* within early-stage startups as a natural part of developing and validating a product, but also that prototyping is implicitly required to ensure funding of the startup venture. Internally, prototyping is used to *explore* and *communicate detailed product requirements*, while prototypes are used externally to *communicate* and *validate product-market fit*. Thus, for startups, prototyping plays a vital role in market validation and in obtaining paying customers. This validation is required by most investors and prototyping thus plays a critical part in securing the funding needed for startup ventures. Prototyping instances tend to be gradually refined from sketches and interactive mock-ups to fully functioning software versions, often MVPs. The more refined prototype versions (realistic mock-ups and early versions of production software) are more costly to produce and require software engineering expertise. Despite this, our case study found that many startups prefer using these more refined prototypes instead of prototyping using simpler media such as paper sketches or mock-ups. This preference, though more costly, may be due to that a more refined prototype appears inductive to a higher degree of customer trust. Thus, software startups face the challenge of balancing the cost of prototype scope and media against the gains and value that can be obtained from more refined prototypes.

3 Research method

We have addressed our four research questions (RQ1-RQ4) by designing a model of prototyping (PAM) using a combination of a systematic mapping study, a focus group at one case company, and a multi-case study of eleven additional case companies, see Fig. 1. The mapping study provided a broad base of scientific knowledge that supported our design of the model. The focus group with practitioners provided an initial validation of our model. The model and its practical applicability and usefulness were further validated through semi-structured interviews at the other eleven case companies by using the model with practitioners to discuss and describe their prototyping practices.

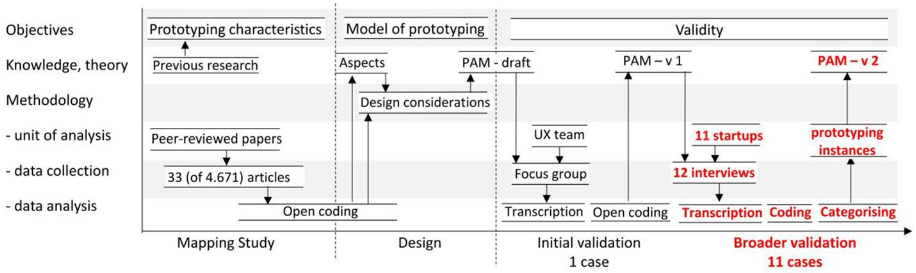

Fig. 1 Overview of the research method used to design and validate the Prototyping Aspects Model (PAM)

The model was designed iteratively in multiple steps. The initial design was performed by the last two authors resulting in a draft of the model. This draft was then revised by the first author after re-analysis of the literature and after using the model with the initial case companies resulting in the first version of the model (published in Bjarnason et al. (2021a)). The first version was then further validated by the first author through semi-structured interviews with practitioners from eleven startup companies and revised based on analysis of the prototyping instances described by the interviewees. The changes to the model were discussed and agreed among all three authors resulting in the second version of our model, which is presented in this paper.

3.1 Systematic mapping study

We performed a systematic mapping study based on guidelines by Petersen et al. (Petersen et al. 2015) to explore and draw from the current body of knowledge on prototyping to compile and provide an overarching view of prototyping methodology based on current knowledge. The literature review was guided by our *research questions* RQ1-RQ3 and consisted of *searches*, *study selection*, *data extraction*, and *data analysis*. The list of scanned articles and our categorisation of the ones included are provided on-line (Bjarnason et al. 2021b) to enable other researchers to validate and to facilitate replicating the analysis of our systematic map.

Searches were defined iteratively in-line with RQ1 through test searches and by consulting with two experts in prototyping. The initial searches were combinations of “software prototype”, “software prototyping”, “prototype”, and “prototyping” that yielded large amounts of hits (almost 80.000). In a second iteration the test searches were limited to the terms “prototyping agile”, “requirements engineering prototyping”, “agile requirement engineering prototyping”. To further guide our study and help focus our searches, we consulted two experts in user experience and design, i.e. one senior manager at the initial case company (Telavox) and one senior researcher in user experience at Lund University. These experts provided insights into how to extend the searches beyond software engineering, thereby increasing the quality of our searches.

Test searches were performed in three search engines, of which two were selected for our review. We selected Lund University Libraries search engine (LUBsearch) and ACM digital library and excluded Google Scholar. LUBsearch provides a broad base since it includes other search engines such as Scopus, IEEE Explore, and ScienceDirect. ACM was selected due to providing a set that suited the scope of our mapping study. ACM provides good coverage of software engineering in general, while complementing the content provided by LUBsearch. In addition, our early test searches with ACM resulted in identifying several articles that are highly relevant to our review.

The **search strings** were specified through an exploratory process with test searches and customised for each of the two selected search engines depending on their specific search facilities. For LUBsearch, we derived keywords from a smaller set of matching papers. The final search consisted of the search string “Prototyping AND (Fidelity OR Software Prototype OR Agile)” for the subject term, with the options “Peer reviewed” and in English. For ACM, we found that many articles lacked keywords and settled on the search string “prototype OR prototyping*”* for the title. The final searches were performed in February 2020.

Study selection was performed through a gradually extended scan of title, abstract, and full text using *inclusion* and *exclusion* criteria to guide the selection decisions (see below), which resulted in identifying 33 primary studies. The two last authors each performed this selection on the set from ACM and LUBsearch respectively. They aligned their selection by comparing and reaching consensus on inclusion and exclusion for a random set of 10 publications. We included articles on prototyping from all fields and excluded articles that do not explicitly discuss prototyping methodology or prototype dimensions. We defined the **inclusion criteria** as articles published before February 2020 that cover meta-level or methodological aspects of prototyping or prototype use, since this is the aim of our main research question RQ1. The **exclusion criteria** were defined as articles that merely describe the use of prototypes without considering methodological perspectives such as principles and procedures for the prototyping, articles that are not peer reviewed or written in English, and duplicates of studies already included in the systematic map. We provide a demographic overview of our map in Section 6.1 including the number of articles per search engine and for each selection step.

Data extraction, data analysis and classification of general and specific items of the primary studies was performed by the last two authors, and then validated by the first author. The general items extracted were publication year and research field (RQ2), and the primary studies were classified according to these and according to research type (RQ3, Wieringa et al. 2006). The resulting classification of the primary studies is reported through describing the demographics of the systematic map in Section 6.1. Furthermore, specific items related to our main research question (RQ1) were extracted. These specific items consisted of the aspects of prototyping covered in the primary studies. We extracted the information specific to our enquiry on prototyping through reading the full text of each primary study and wrote a short summary of how it relates to our study. Initial categories, or aspects of prototyping, emerged and were identified based on these summaries, similar to open coding of grounded theory. When these aspects had been established, each primary study was classified according to these aspects.

3.2 Design of prototyping aspects model including initial validation

Our model was designed iteratively based on the open coding of our systematic mapping study. First, the commonly occurring aspects of *purpose* (why) and *prototype scope* (what) were included, followed by *exploration strategy* and *prototype use* (how). Each aspect was detailed into further facets through analysis of the primary studies. An additional aspect that was considered, but at this point excluded, was the method used to produce a prototype, e.g., paper prototyping or computer simulation. Since our focus was prototyping methodology, and since computer-simulated prototypes can achieve similar effects as paper prototypes, we decided to exclude this aspect from the first version of our model. However, a similar fifth aspect of *prototype media* was added in the second version of our model based on the insights gained from our multi-case study. For the first version of the model, an additional design iteration was performed to increase internal validity of our model.

We performed an initial validation of the draft version of the prototyping aspects model by re-analysing the articles in our systematic map (see below). This internal validation mainly led to renaming aspects, adding, and restructuring some facets. The aspect of strategy was renamed *exploration strategy* and the aspect of review method was renamed *prototype use* to more clearly reflect what these aspects represent. The facets related to *validation & testing* were grouped to provide a better overview of the purposes of prototyping. Two additional facets were added to *prototype scope*, namely *interactive & haptic behaviour*, and *realistic data*, both of which were observed in the original analysis, but initially not included. Finally, the facet of usage environment was added to *prototype use* since the feedback that can be obtained from demonstrating or testing a prototype may vary for, e.g., a loud or a dark environment compared to in a meeting setting.

The design of our model was further refined after subsequent validation with practitioners at Telavox and through a multi-case study of ten software startup companies. These validation steps are described below. The first and second version of the model are described in Section 5, and the differences between the two versions are described and motivated in Section 6.3 based on our subsequent multi-case study.

3.3 Validation of draft version: Focus Group with Telavox and Reanalysis

The initial draft version of our prototyping aspects model was validated through reanalysis of the primary studies of our systematic map and through a focus group at a case company.[1] The reanalysis was performed to improve reliability and internal validation of our prototyping aspects model, while the focus group was performed to validate the relevance and usefulness of our model with practitioners. The first author performed triangulation of the model through an independent re-analysis of the primary studies of our systematic map including (re)coding the full text of the articles in NVivo. The differences were then discussed, agreed with the other two authors, and the model updated as described in this article.

The focus group was prepared by designing a focus group protocol structured according to the main activities or stages of a requirements process, namely Concept exploration, Eliciting customer needs, Identifying system scope & requirements, Validate and improve system scope & requirements, and Confirm system scope & requirements. These stages were inspired by Lauesen and correspond to the main activities of requirements engineering common to all projects, both traditional and agile (Lauesen 2002). For each stage, we considered the main RE goals and suitable prototyping instances, or scenarios based on two of the aspects of our model, namely purpose and prototype scope. These prototyping scenarios were discussed at the focus group, supported by a set of questions. The protocol was iteratively designed by the last two authors, reviewed with the first author and with a contact person at the company, and then improved upon. The focus group protocol is available in Appendix A.

[1] The focus group was performed prior to the reanalysis and thus on initial draft of PAM, as depicted in Fig. 1.

The focus group was conducted with five practitioners from Telavox' user experience team. This team elicits and details product requirements through prototyping of the user interface. The participants had degrees in either industrial economics or interaction design; two B.Sc. and three M.Sc. The focus group was managed, led, and moderated by the last two authors. To ensure equal airtime, the participants were given turns to initiate the discussion for each talking point. The focus group was performed online due to the ongoing Covid-19 pandemic. The meeting was recorded, transcribed, and analysed using open coding to identify information related to the different stages of requirements, and to the different aspects of the model. The empirical data from the focus group it is treated as confidential since it may contain company information and the participants are promised anonymity to encourage them to speak freely. Furthermore, the description of the case company and the case-specific results have been reviewed by the case company.

3.4 Broader validation: multi-case study

We performed a multi-case study of software startups using semi-structured interviews to gain insights into current prototyping practices. Software startups were selected as our object of study due to these companies frequent use of prototyping to explore and shape their business models and product offerings, as part of the Lean startup method (Ries 2011) where new products are designed through continuously building, measuring, and learning what the customers want and are willing to pay for (Olsen 2015). Through the case study, we explored the practical applicability of our model by using it to discuss and categorise prototyping instances, thereby validating the model further and identifying improvements to it (RQ4). The case study consisted of four stages, namely *preparations*, *data collection, data analysis,* and *validation.*

In the *preparation stage*, a case study protocol and an interview guide were designed (available in Appendix B), and an initial set of case companies and interviewees were recruited. The interview guide is based on the first version of our prototyping aspects model (Bjarnason et al. 2021a) and on previous research on software startups (progression model (Klotins et al. 2019), typical characteristics (Berg et al. 2018; Giardino et al. 2015)). The interviews were designed to investigate how startup companies work with requirements and prototyping, and support a broad exploration of (possibly) influencing factors by covering questions about the interviewee and the company. After the first two interviews, it became clear that the terms prototyping and prototype are not uniformly understood, which caused some initial confusions at these interviews. The interview guide was then extended with a question about what prototyping means to the interviewee. Case companies and interviewees were selected through convenience sampling and consisted of startups recruited through local business incubators; Minc, VentureLab, SmiLe, and Ideon Innovation, of which some were previously involved in our RE course. The main criteria for selecting startups to include in our study, was that they use or plan to use prototyping for developing their business and/or software. In total, eleven startups (Company A-K) were investigated through twelve interviews with thirteen interviewees. The companies and interviewees are described in Section 4.2. In general, one interview was held per company, except for Company D where the founder and the technical lead were interviewed separately. In addition, the interview at Company H was held with two founders of different profiles (at their request).

The *data collection* consisted of semi-structured interviews with ample opportunity for interviewees to speak freely and to ask follow-up questions. The interviews were held and recorded in a video conferencing system (Zoom), and each lasted for about one hour. At the interviews, we presented our prototyping aspects model (version 1) and opened up for a discussion of how their prototyping practices relate to the different aspects (purpose, scope, use, and exploration strategy of prototyping). At the beginning of each interview, the participants were informed that their participation is voluntary and that they and their companies can remain anonymise, if they so wish, and that the interview data is treated as confidential since it may contain company information. Furthermore, each participant was sent a draft of the resulting articles for the parts derived from their interview for validation purposes.

In the *data analysis stage*, the audio recordings were transcribed and analysed by applying thematic coding in several iterations using NVivo. In the *first analysis iteration*, codes representing the different parts of the interview were used, such as interviewee roles and background, company and product characteristics, startup life-cycle maturity and challenges, RE and prototyping practices. In the *second analysis iteration*, each interview transcript was re-read and prototyping instances mentioned by the interviewees were identified. The relevant parts of the transcripts for each prototyping instance were coded using codes denoting the case company and a sequence number for each prototyping instance. For example, for Company B for which three prototyping instances were identified, the codes B1-B3 were used to denote and tag the relevant parts of the interview transcripts for each of these instances. In the *third analysis iteration*, the interview data per prototyping instance was analysed and each prototyping instance was categorized according to the prototyping aspects model (see table in Section 6.3). Observations about how well the model corresponded to the described prototyping instances were noted in memos, together with descriptions of each prototyping instance, and illustrative quotes found in the transcripts. These memos were then used to report the results and to adjust the prototyping aspects model as described in Section 6.3.

Finally, in the *validation* stage of our case study, the interviewees were asked to validate the descriptions related to their companies and the co-authors discussed and agreed to the revised version of the prototyping aspects model. All interviewees were contacted and asked to validate the information relevant to their companies, and to indicate if they wished their company name to be included in the article or not. The interviewees were provided with the prototyping instances identified for their startup, the memos related to these including the quotes extracted from the transcripts, and the company and interviewee characteristics reported in Section 4.2. The interviewees responded, mainly by agreeing to the descriptions and by highlighting some minor misunderstandings and additional information, after which the manuscript was revised accordingly.

4 Case companies

The initial draft of the prototyping aspects model was validated through a focus group at Telavox. The first published version of the model was then further validated through the multi-case study of eleven software startup companies where twelve practitioners were interviewed about their use of prototyping.

Table 1 Main characteristics of the eleven startups included in the multi-case study. Companies for which no name is given prefer to remain anonymous

Case company		Age	#Employees	Product domain
A	Pinteg AB	2 years	1	Data privacy management and personal integrity
B	Smart Green Stations AB	3 years	1	Sustainability
C		< 1 year	1	Personal communication
D	AgVital	3 years	4	Diet precision health
E	SiB Solutions	4 years	20	Logistics technology
F		2 years	8	Health
G	Homepal AB	4 years	7	Facilities management
H	Animals AI	2 years	6	Agriculture
I	nobiz	6 years	2	Consulting, Marketing, User experience
J		11 years	40	Consumer applications
K		2 years	2	Software engineering tools

4.1 Initial case company: Telavox

Telavox offers cloud-based Private Branch Exchange solutions. The company was founded in 2002 and currently has around 250 employees. The initial part of this study, i.e. the mapping study and initial design of the model, was carried out in 2020 at the company's site in Malmö, Sweden, by the 2nd and 3rd author. At this site there are about 15 development teams for areas such as app development for Android and iOS, user experience, and web development. Product support is coordinated and provided via key account managers that work closely with customers. The company applies Scrum, and thus uses an agile development model. New and improved product ideas are explored and communicated through prototyping and general product statements. Product owners coordinate product development and prioritise product requirements. When an idea is ready for development, the appropriate teams are assigned user stories. The teams apply test-driven development and continuous delivery. Product scope is validated with key account managers and customers before release.

4.2 Case companies A-K: software startups

Eleven software startups (Company A-K) were investigated in our multi-case study, see Table 1. Most of these companies were in the inception or stabilisation phase (Klotins et al. 2019) of their business ventures, while Companies E and J were in the growth stage (Klotins et al. 2019). In total, we interviewed thirteen practitioners at the case companies, mostly founders and co-founders, but also some product owners and technical leads. An overview description of our interviewees is provided in Table 2.

5 Results: Prototyping aspects model (RQ1)

Our model characterizes the methodology of prototyping by five aspects: *purpose of prototyping*, *prototype scope*, *prototype media, use* of *prototype,* and *exploration strategy*, and is here described based on related work. Table 3 provides an overview of the model

Table 2 Main characteristics of the interviewees for each startup

Company	Interviewee's role	Main competences	Experience (years)
A	Founder and CEO	Sales & Marketing, Business development	20+
B	Founder	Electrician	10+
C	Founder	Economics	10+
D	Founder	Engineering, Business management	30+
	Technical lead	Software engineering, Business management	20+
E	Chief product officer	Software engineering, product & project manager	10+
F	Product owner	Business and marketing	20+
G	Founder	Engineering	10+
H	Founder, CTO	IT, Economics	10+
	Founder, Head of Marketing	Marketing, communication	10+
I	Founder	Usability, interaction design	20+
J	Founder	Engineering	20+
K	Founder	Engineering, Project management	20+

and the primary studies from our initial mapping study for each aspect. In this article, a revised version of the model is presented based on a multi-case study where the model was used to characterise prototyping instances. The changes to the first version of the model are described and motivated in Section 6.3. Table 3 contains both the first version of the model (published in Bjarnason et al. (2021a)) and the revised (second) version (presented herein) to facilitate comparison.

5.1 Purpose of prototyping – why prototype?

The practice of prototyping can achieve many different purposes that often vary throughout the life cycle of a project. We have identified eight purposes, namely *exploration*, *communication*, *incremental development*, *quality improvement*, and validation & testing of *problem–solution fit, product-market fit, technical feasibility,* and *usability*. When prototyping, several purposes may be satisfied simultaneously, e.g., communicating the product idea to potential customers while also validating its market desirability. A project's purpose of prototyping often evolves from exploration and communication to validation & testing (Ratcliff 1988).

5.1.1 Exploration & learning

Prototyping is commonly used to explore the solution space (Dow et al. 2011; Lim et al. 2008; Rahman et al. 2017; Wiberg and Stolterman 2014) and learn by experimenting with ideas (Budde and Zullighoven 1990; Chen et al. 1994; Tronvoll et al. 2017), and is a foundational purpose for any prototyping. Problems and new solution directions can be discovered and explored through prototyping and can lead to new ideas and direct further exploration (Lim et al. 2008). Such exploration of multiple solutions can mitigate the risk of overinvesting in a single concept (Dow et al. 2011). However, Schneider found that when users are left out of the process and prototype use is purely internal, only the developers learn (Schneider

Table 3 Overview of our prototyping aspects model (PAM) with supporting references from our systematic mapping study. The first version of the model was published in Bjarnason et al. (2021a) and is shown in the right-most column

Prototyping aspects model	Supporting references	*First version of model*
Purpose of Prototyping		
Exploration & learning	Arano et al. 1993; Bellomo et al. 2013; Budde and Zullighoven 1990; Chen et al. 1994; Ciriello et al. 2017; Derboven et al. 2010; Dow et al. 2011; Fairley and Willshire 2005; Fern and Donaldson 1989; Hakim and Spitzer 2000; Jaskiewicz and Helm 2018; Kordon 1994; Lichter et al. 1994; Lim et al. 2008; McCurdy et al. 2006; Rahman et al. 2017; Raja 2009; Ratcliff 1988; Schneider 1996; Toffolon and Dakhli 2008; Tronvoll et al. 2017; Wiberg and Stolterman 2014; Zink et al. 2017)	Exploration & learning
Communication: sales, alignment		Communication & alignment
Incremental development		Incremental development
Quality improvement		Quality improvement
Validation & testing: Problem–solution fit, product-market fit, technical feasibility, usability testing		Validation & testing: Business viability, market desirability, technical feasibility, usability testing
Scope of Prototype		
Breadth of functionality	Bruegger et al. 2009; Budde and Zullighoven 1990; Derboven et al. 2010; Goldman and Narayanaswamy 1992; Hakim and Spitzer 2000; Heisler et al. 1989; Hendry et al. 2005; Jaskiewicz and Helm 2018; Kordon 1994; Lim et al. 2008; Liu and Khooshabeh 2003; McCurdy et al. 2006; Sefelin et al. 2003; Tronvoll et al. 2017; Yasar 2007; Zainuddin and Liu 2012)	Breadth and depth of functionality
Functional refinement		
Visual appearance		Refinement of visual appearance, interactive & haptic behaviour, data realism
Interactive & haptic behaviour		
Data realism		
Prototype Media		
Sketch: paper or computer-based	Hendry et al. 2005; Liu and Khooshabeh 2003; McCurdy et al. 2006; Sefelin et al.2003)	
Wireframe: paper or computer-based		
Mock-up: paper or computer-based		
Source-code software		
Other: video, interview		

Table 3 (continued)

Prototyping aspects model	Supporting references	*First version of model*
Use of Prototype		
Reviewers: internal, FFF, or external	Budde and Zullighoven 1990; Cafer and Misra 2009; Ciriello et al. 2017; Dow et al. 2011; Grevet and Gilbert 2015; Heisler et al. 1989; Hendry et al. 2005; Lichter et al. 1994; Liu and Khooshabeh 2003; Schneider 1996; Tronvoll et al. 2017; Zainuddin and Liu 2012)	Review method:
Prototype interaction: yes, no (demo)		Demo, scenario testing, free testing, internal use
Review approach: scenario-based, free		
Usage environment		Usage environment
Exploration Strategy		
Single vs parallel exploration	Budde and Zullighoven 1990; Dow et al. 2011; Jaskiewicz and Helm 2018; Tronvoll et al. 2017)	Point-based exploration
Iteration focus: Business, product, feature, optimisation		Parallel exploration
		Optimisation exploration
Iteration size		Flexible exploration

1996). Lichter et al. suggest combining the purpose of exploration with communication and testing of product-market fit to clarify requirements (Lichter et al. 1994).

5.1.2 Communication: sales, alignment of requirements

Visions and ideas about a product can be communicated through prototyping, which provides a common language between developers and stakeholders (Budde and Zullighoven 1990; Ciriello et al. 2017; Rahman et al. 2017; Zink et al. 2017) and acts as an anchor for group communication (Dow et al. 2011). Prototyping thus facilitates presenting, discussing, and evaluating a product with external parties, such as customers and investors, and internally within a project, thereby supporting decision making (Budde and Zullighoven 1990; Raja 2009*)*. Ciriello et al. note that prototypes can support requirements elicitation by clarifying problems early on (Ciriello et al. 2017). However, a prototype may convey an overly positive impression of the current status that can lead to unrealistic customer expectations and subsequent requests to evolve the prototype into the final system (Lichter et al. 1994).

5.1.3 Incremental development

One purpose of prototyping may be to evolve the prototype into a final product based on user feedback and priorities (Budde and Zullighoven 1990; Lichter et al. 1994; Schneider 1996; Toffolon and Dakhli 2008). In these cases, the prototype may be a pilot system or a partial product version such as alpha or beta, or an MVP, that is developed with the expressed intention of exploring or validating a solution option (according to our definition of prototype). In agile development, prototyping is often an integral part of the development process with regular feedback from users and other stakeholders (Fairley and Willshire 2005), e.g. through validation of software at end-of-sprint demonstrations. Thus, prototyping is often used within agile development to detail and validate requirements, and to reduce uncertainty (Bellomo et al. 2013). Fern et al. propose a prototyping methodology that covers throw-away prototypes as well as prototypes that will be developed further into the deliverable system (Fern and Donaldson 1989).

5.1.4 Quality improvement

Prototyping can be used to optimise quality aspects, e.g. by focusing on response times while all other behaviour is retained. Kordon observed that care needs to be taken to avoid measurement overhead and to ensure accuracy in the evaluation (Kordon 1994). Arano et al. propose the use of hybrid prototyping for improving quality aspects since this can enable measuring quality in early development stages prior to full implementation (Arano et al. 1993).

5.1.5 Validation & testing

One of the main purposes of prototyping is to validate requirements by testing a solution option with internal and/or external stakeholders (Ciriello et al. 2017; Tronvoll et al. 2017) for perspectives, such as problem–solution fit, product-market fit, technical feasibility, and usability. *Problem–solution fit* concerns the degree to which the envisioned product solves an actual customer or end-user problem. Prototyping can be used to investigate customer needs, validate and clarify customer requirements and tasks (Budde and Zullighoven 1990;

Ciriello et al. 2017*),* and thus support a company in designing a solution to address customer problems. *Product-market fit* is a premise for business viability. Prototyping can be used to explores a product's business potential from the customer perspective, and to assess the value of the product and likeliness of purchase (McCurdy et al. 2006; Zink et al. 2017) and the product's ability to fit within time and budget constraints (Zink et al. 2017*).* The insights gained from such prototyping can be used to support business-related decision making (Ciriello et al. 2017). *Technical feasibility* concerns a system's technical capabilities and the feasibility of realising requirements in the solution space (Budde and Zullighoven 1990; Zink et al. 2017), e.g. to operate at scale, to resolve structural uncertainties, or to fulfill security requirements. Proof-of-concept prototypes are built for this purpose (Tronvoll et al. 2017) and can be used to evaluate novel technical approaches (Fern and Donaldson 1989). Similarly, breadboard prototypes are used to investigate technical aspects, e.g. in circuit design, and to support system specification and coding (Budde and Zullighoven 1990; Ciriello et al. 2017; Lichter et al. 1994; Lim et al. 2008). Furthermore, Lichter et al. found that a prototype built to demonstrate feasibility can also support project acquisition (Lichter et al. 1994). Finally, Zink et al. found that feasibility testing requires a high share of the time invested in a project (Zink et al. 2017). *Usability testing* was reported by Zink et al. as the least common singular purpose for prototyping though it is often combined with other purposes (Zink et al. 2017). User interface design can be validated through prototyping (Derboven et al. 2010; Hakim and Spitzer 2000; Zink et al. 2017) and uncover various usability issues (Lim et al. 2008). When prototyping for usability testing, Hakim et al. recommends ensuring that metrics such as task completion time, can be captured and considering the protocol for use (e.g. associated instructions and data) since this may affect the results of the testing (Hakim and Spitzer 2000).

5.2 Scope of prototype – what to prototype?

The scope of a prototype describes the extent to which a prototype resembles the final product and is, in our model, represented by the **breadth** of the prototype's functionality (Bruegger et al. 2009; Budde and Zullighoven 1990; Goldman and Narayanaswamy 1992; Hakim and Spitzer 2000; Lim et al. 2008; McCurdy et al. 2006; Tronvoll et al. 2017), and the facets of **functional refinement** (Bruegger et al. 2009; Budde and Zullighoven 1990; Goldman and Narayanaswamy 1992; Hakim and Spitzer 2000; Lim et al. 2008; McCurdy et al. 2006; Tronvoll et al. 2017), **visual appearance** (Budde and Zullighoven 1990; Goldman and Narayanaswamy 1992; Hakim and Spitzer 2000; Hendry et al. 2005; Jaskiewicz and Helm 2018; Lim et al. 2008; Liu and Khooshabeh 2003; McCurdy et al. 2006; Zainuddin and Liu 2012), **interactive & haptic behaviour** (Budde and Zullighoven 1990; Goldman and Narayanaswamy 1992; Hakim and Spitzer 2000; Hendry et al. 2005; Lim et al. 2008; Liu and Khooshabeh 2003; McCurdy et al. 2006) and **data realism** (Lim et al. 2008; McCurdy et al. 2006). *Breadth* represents the extent to which a prototype covers a product's full functionality, e.g., all or only one product features (broad or narrow prototype), while the other facet represents the degree of detailing of this (deep or shallow) regarding functionality, visual appearance, interactive behaviour, and data realism. *Functional refinement* represents the degree of detailing for each function or feature. The *visual appearance* of a prototype concerns aesthetics, e.g., layouts, fonts, and user interface elements, and is interrelated to functional refinement since visuals are represented by functionality. The user's experience is also affected by how well a prototype mimics a product's *interactive &*

(sometimes) *haptic behaviour,* and to what degree the available *data* can simulate normal and realistic use.

The *scope* affects the cost of producing a prototype and the feedback that can be obtained (Derboven et al. 2010; Lim et al. 2008; Liu and Khooshabeh 2003; McCurdy et al. 2006; Tronvoll et al. 2017). Thus, prototype scope relates to the purpose of prototyping. A rough paper prototype of one product feature is an example of a prototype with a small functional breadth and a low degree of refinement, i.e. a narrow and shallow prototype. In contrast, a minimal viable product may have a broad functional scope by representing all menu options, but varying degrees of refinement (and depth) depending on the implementation status for each feature and for the overall product, e.g., the user interface, and for the amount of user data provided in the prototype.

Yasar describes three perspective to consider when prototyping: role (of product in a user's life), look & feel, and implementation (Yasar 2007). These perspectives can help pinpoint the purpose of prototyping and the scope of a prototype. In our model, look & feel is captured by the facets of visual and interactive & haptic behaviour. The facet of realistic data relates to look & feel, but primarily to the perspective of role since the degree to which provided data simulates actual use affects the user's ability to relate to their situation.

There are different views on how the degree of prototype refinement, e.g., for visuals, affects the type and amount of feedback that can be obtained. Sefelin et al. reports on a study on usability testing with prototypes where the number of suggestions and critiques do not significantly differ between prototypes with a high vs. low degree of refinement, i.e. between deep or shallow prototypes. However, the users preferred testing with a prototype with a high degree of interactive refinement since this provided freedom to explore the system by themselves (Sefelin et al. 2003). Several researchers report that some usability issues can not be discovered unless the prototype provides a broad functional scope with a high degree of refinement, in particular for visual appearance and interactive behaviour (Liu and Khooshabeh 2003; McCurdy et al. 2006). However, a mix of low and high degrees of refinement is suggested as more economical, e.g., for usability testing. Yasar even reports that simpler prototypes with narrow functionality scope and a low degree of refinement are cost effective and useful for validation purposes and to capture major issues (Yasar 207 Similarly, Bellomo et al. describe how a company consciously selects the breadth and degree of refinement of prototype scope to match the prototyping purpose, and to validate the concept in focus (Bellomo et al. 2013).

The terms hi and lo-fidelity are often used to describe resemblance to the final product. Our terminology is more fine-grained and provides a more precise way of categorising prototypes by their breadth and degree of refinement w.r.t. different facets. We believe that our model can support informed decisions of which prototype scope that is required to meet the desired purposes of prototyping.

5.3 Prototype media – what media to use for prototype?

A prototype can be constructed using a range of techniques and media, including sketching on paper (Hendry et al. 2005) or in PowerPoint, using a prototyping tool to produce a wireframe or a mock-up (Liu and Khooshabeh 2003), or using an early version of product software as your prototype (McCurdy et al. 2006; Sefelin et al. 2003). Even a video or a physical model is included in our definition of a prototype, and can be used to, e.g., validate product-market fit, or to communicate with stakeholders.

While there is a correspondence between different kinds of prototype media and the scope of these, the cost of producing prototypes, the gains and types of learning that can be obtain from these varies with the media (McCurdy et al. 2006; Zainuddin and Liu 2012). The choice of media may also affect the ease with which the prototype can be evaluated in some environments (Hendry et al. 2005). Sefelin et al. compared the use of paper sketches to that of computer-based prototypes of identical breadth and level of refinement, i.e. different media was used to represent the same prototype scope. These prototypes were used for usability testing, and the results showed no major differences in the obtained feedback depending on prototype media. However, the computer-based prototypes tended to yield more comments on graphical details, while the paper sketches stimulated participant to "*a greater willingness to draw their suggestions*" (Sefelin et al. 2003).

Liu and Khooshabeh made a similar comparison between paper sketches and computer-based (interactive) mock-ups (Zainuddin and Liu 2012). They found that while paper sketches provide more flexibility for designers in early stages, this kind of prototype media requires more effort to use with larger sets of users, e.g., to manually simulate interaction. Furthermore, Liu and Khooshabeh also found that interactive computer-based mock-ups yielded more feedback when used for user testing.

While paper sketches are often cheaper and faster to construct, Sefelin et al. suggest selecting prototype media based on a number of factors, including the competence of those constructing the prototype, the abilities of the available prototyping tools, and the envisioned continuation of the prototyping. Finally, Hendry et al. found that paper sketches were a good media for eliciting and validating requirements in actual environments and with some stakeholder groups, such as in public locations where people of all age categories can be accessed (Hendry et al. 2005).

5.4 Prototype use – how to use the prototype?

This aspect concerns how a prototype is used to achieve a purpose and covers who the **reviewers** are (internal, external, or with family-friends-and-foes FFF), if direct **prototype interact** is used (or not), what **review approach** that is used (scenario based or free), and in what **environment** the prototype is presented and reviewed. With these four facets of prototype use the main uses of prototype identified in our literature review can be represented, namely internal prototype use without any user presentation (Budde and Zullighoven 1990; Dow et al. 2011; Schneider 1996), prototype demonstrations (Bellomo et al. 2013; Heisler et al. 1989), scenario testing (Tronvoll et al. 2017), and free testing (Heisler et al. 1989; Tronvoll et al. 2017). Lichter et al. emphasise the importance of user involvement in prototyping (Lichter et al. 1994) and several researchers address challenges with this (Ciriello et al. 2017; Lichter et al. 1994; Zainuddin and Liu 2012). Zainuddin and Liu propose a systematic approach to building prototypes with user involvement and capturing user feedback during prototype use (Zainuddin and Liu 2012).

The choice of **reviewers** affects the learnings that can be obtained from prototyping. Purely *internal* use of a prototype can support brainstorming and allow designers and engineers to generate and organize ideas. While internal use allows a project to focus on ideas and possibilities rather than on external expectations, there is a risk that knowledge remains with, e.g., the developers. Schneider suggests better capitalization of pure internal prototyping by capturing and documenting this knowledge (Schneider 1996). Using a prototype with *external* parties often increases the cost due to increased expectations and demands on quality. In addition, the pre-knowledge of customers and users affect the feedback that can be obtained, and Cafer and Misra suggest adapting the review method based on the customers' cognitive abilities (Cafer and Misra 2009). Several of the startups included in our multi-case study described showing prototypes to family-friends-and-foes (FFF) before going to potential customers, as a means to obtain friendly feedback.

There is usually no direct reviewer **interaction** with the prototype during demonstrations. Instead, the presenter shows the prototype by operating it live or by using prepared media such as videos or photos. For more refined prototypes, e.g. computer-simulated mock-ups with a high degree of interactivity reviewers may be encouraged to interact and directly use the prototype, and thereby provide richer insights and user feedback. Such interaction can be achieved also with simpler prototypes, additional human effort is required to simulate, e.g. button presses, when using prototypes with a low degree of refinement for interactive & haptic behaviour (Liu and Khooshabeh 2003).

Different **approaches** can be used when evaluating a prototype, e.g. a scenario-based approach which can support reviewers in relating the prototype to their own problems and usage scenarios. When using prototyping for testing purposes, *scenarios* can be used to guide users through instructions and steps. Similarly, Ciriello et al. suggests using storytelling to increase customer involvement (Ciriello et al. 2017). An alternative approach when testing prototypes with external parties or FFFs, is to encourage free testing as for example for beta testing when minimal instructions are provided.

The **usage environment** for a prototype can affect the outcome (Grevet and Gilbert 2015; Lichter et al. 1994; Tronvoll et al. 2017) both in the possible feedback and the representation of subjects in the review. If a product's natural habitat is very different from a lab or meeting room environment, this can affect the feedback, e.g., for a system intended for a loud environment. Using a prototype in conditions similar to that of the final product increases the chance of uncovering new setting-specific requirements (Lichter et al. 1994). Hendry et al. performed usability testing among home-less people and concluded that using a prototype in the streets enabled them to reach otherwise inaccessible user segments (Hendry et al. 2005). Thus, the environment of use may result in some future users being underrepresented in the prototyping, further limiting the feedback.

5.5 Exploration strategy – how to traverse the solution space?

This aspect concerns strategies used to traverse the solution space over time. The exploration strategy determines which instances to pursue, how resources and decisions are organised, and how uncertainties are managed. The needs and goals may change for each iteration, which can enable focusing on specific product aspects or parts. Within one iteration, several concurrent solution paths can be pursued in parallel, which can stimulate innovation (Dow et al. 2011) and avoid imposing unnecessary limitations (Budde and Zullighoven 1990), but also cause contradictions between prototypes and complicate decision making (Jaskiewicz and Helm 2018). When iterating over multiple parallel prototyping instances, the length of the iterations may impact the extent and quality of the prototyped solution option and thus the prototyping effectiveness (Tronvoll et al. 2017). Jaskiewicz et al. found that while longer iterations promote focusing on certain aspects, multiple short iterations can lead to a more diverse but superficial set of solution options (Jaskiewicz and Helm 2018).

Table 4 Number of articles in each step of the selection process

Search engine	*Articles found*	After scan of		
		Title	*Abstract*	*Full article*
LUBsearch	2.713	84	30	19
ACM	1.958	62	24	14
Total	4.671	146	54	33

We model this aspect using the three facets: **single vs parallel exploration**, **iteration focus**, and **iteration size** These facets are partly based on the four strategies identified by Tronvoll et al., namely *point-based*, *parallel* (or set-based solution arrays), *optimisation* (or performance set investigations), and *flexible exploration* (Tronvoll et al. 2017). Our facet of **single exploration** corresponds to Tronvoll's *point-based strategy* where resources are focused on one single solution path and where alternatives are considered before being cemented. If the solution appears unsuitable it may be necessary to discard the prototype and redo the process. In contrast, several potential solution options are pursued simultaneously when using **parallel exploration**. Decisions are managed by merging prototype variants either continuously or during stage-wise iterations.

Tronvoll's other two strategies, namely *optimisation* and *flexible exploration* can be viewed as either **single or parallel exploration**, though the **focus** and **size** of the iterations varies, which is captured by these two facets of our model. Tronvoll's optimisation exploration uses *parallel exploration* with a focus on *feature or product level,* and with gradually decreasing iteration *sizes*. For such exploration, the number of simultaneously investigated options are kept to a minimum to reduce cross-contamination and over time the solution converges by applying a systematic performance evaluation of the optimised characteristics. Decisions surrounding the solution options are postponed until they can be validated and when encountering several alternatives, only the most promising one is pursued. Solution options are judged by performance, which affects the evaluation. Such prototypes should be evaluated against a range (weak-strong-good quality) and may involve additional factors that affect each other. For example, a certain design-level requirement may provide an efficient way of saving text, but drastically increase the size of the saved file.

Finally, Tronvoll's *flexible exploration* strategy corresponds to a **single exploration strategy** where the solution options are based on best-guesses and on facilitating the change of quality requirements as the work progresses. This approach is well suitable for agile development where the solution option is iterated, evaluated, and changed as required.

6 Underpinning results: map demographics, focus group and multi-case study

We will now present the results used to design the prototyping model aspects presented in the previous section. In this section, we report on the demographics of our systematic map, the results from the focus group and from the interviews of our multi-case study. The model was initially designed based on previous literature, then validated and adjusted through the empirical data of the focus group and the multi-case study.

Table 5 Number of articles published per 5-year period for each main discipline

	SE	RE	HF/ Design	Total
1988–1992	4	1		5
1993–1997	5			5
1998–2002			1	1
2003–2007	2		4	6
2008–2012	2	3	3	8
2013–2018	4	1	3	8
Total	17	5	11	33

6.1 Demographics of Map (RQ2 and RQ3)

Our systematic map consists of thirty-three primary studies published between 1988 and 2018, see Table 4. The majority of these (17) were within software engineering (in general), 11 within human factors (HF) & design, and 5 within requirements engineering (RE), see Table 5. The number of articles appear to have increased somewhat over the years from on average 1 to 2 articles a year, except around the year 2000. The increase is mainly within the areas of HF & Design and RE. The research type for each paper was classified according to the categories by Wieringa et al. (Wieringa et al. 2006): (1) *Evaluation research* investigates a problem or technique in practice and provides new knowledge of causal or logical relationships, (2) *Solution proposals* present a solution without a full-blown validation, (3) *Validation research* presents a solution proposal validated outside of industrial practice. (4) *Philosophical papers* sketch new theories or frameworks, (5) *Experience papers* describe personal experiences and may contain anecdotal evidence.

Around 40% of the articles (14 of 33) are based on *in-vivo* research and empirical data from industry (*Evaluation)*, see Table 6. Another 40% (12 of 33) propose solutions based on in vitro (only) validation or theoretical reasoning (*Solution proposal* and *Validation* type). Our systematic map indicates an increase of in-vivo research concerning prototyping over the past 10–15 years.

6.2 Focus group results: Validation of Initial Version of Model[2]

Practitioners discussed the use of prototyping to perform requirements-related tasks such as elicitation, specification, and validation at the focus group by reflecting on scenarios based on our prototyping aspects model. In general, the participants agreed to the scenarios, that they correspond to a typical way of working and to their prototyping practices, which they see as a good way of working with agile software development. There was some disagreement about prototyping in early project stages. Several participants said that prototyping is time consuming and difficult, and rarely used during early phases and at the first meetings with customers. Instead, these participants preferred stakeholder interviews and competitor analysis to understand the problem domain. In contrast, two participants said that early prototyping, e.g., through sketching, can be useful in understanding the problem. Another participant said that early prototypes can be beneficial in loosely defined

[2] The aspect of prototype media was not included in the model at this stage of our study and thus not covered in the focus group.

Table 6 Types of research of articles in our systematic map.

	1988-	1993-	1998-	2003-	2008-	2013–2018	**SUM**
Evaluation	1	1		3	4	5	14
Experience	1		1			1	3
Philosophical	2			2	1		5
Solution proposal	1	3			3	2	9
Validation		1		1			2

projects and facilitate discussions about user expectations. They had used simple prototypes in the form of sketches, which helped designers and users understand what they were discussing and assisted in framing the role of the system.

6.2.1 Purpose

The participants described several purposes for prototyping. Its value in *exploration & learning* was described as "*the more you work with design [of the prototype] the more you know what works for you*".[3] Several participants described the value of prototyping for internal use and the importance of "*building understanding before you start thinking about solutions*." One participant described creating personal prototypes to get an overview of the future system, essentially working as a *specification*. Another participant said that prototypes are more useful than a formal specification to *communicate* requirements. Two participants described the importance of exposing developers to the prototype as a way of *testing the technical feasibility* and, if possible, adjust the requirements to "*the easiest way … to implement the solution*". Another participant described the importance of communicating and involving developers by testing prototypes and discussing requirements, although some developers just want information on "*the components I need to build*."

6.2.2 Prototype scope

The focus group provided validation for this aspect of our model. They recognised the relevance of considering breadth and depth of functionality, and refinement of all our identified facets (of the first version of our model) except for data richness, which was not mentioned during the focus group. The participants described that in early project stages, they prefer to use simpler prototypes, either paper sketches with shallow functionality and low refinement or a prototyping tool that supports producing prototypes with shallow functionality and mid to high degree of visual refinement. These simpler prototypes provide "*a way to understand the problem*" and to "*organize and explore …* [the solution] *space*." Another participant implied that feedback on visual style could be avoided by using prototypes with a low degree of visual and interactive refinement such as wireframes that encourage feedback on functionality. In contrast, for more complex solutions "*you get value out of being able to click around*" and a high degree of interactive refinement is preferred. Some participants said that prototyping must involve interactivity and that they "*wouldn't do a good job …* [without] *interactive prototypes*".

[3] Direct quotes from focus group participants are noted within citations and italicized.

This illustrates the need for a common definition of prototyping. The fact that prototype scope was spontaneously discussed by our participants demonstrates that our model can provide support for this.

6.2.3 Prototype use

The participants confirmed all four review methods included in the first version of our model, i.e., internal use, demonstrations, scenario-based, and free testing. Several participants described using prototyping to test new ideas with colleagues and that they first show the prototype "*within the team and not with users*". This was especially the case for early prototypes that one participant rarely showed to customers, despite experiencing challenges in discussing with users without any visuals. When prototypes are shown to external stakeholders it is often as a demonstration that takes place after first having discussed their problem and current situation. One participant said that users are only occasionally involved in interacting with prototypes while this is done "*all the time*" within the project. Instead, users are involved in beta testing "*all the time*", which some, but not all, of the participants see as prototyping. One participant described the use of scenario testing a few times a year, e.g., with new employees or students.

Several participants mentioned the importance of structuring feedback sessions to "*get feedback on the right thing*", i.e. relevant facets. One participant said that it is easier to get feedback on what is good rather than what needs improving. Another participant described that "*talking with just one person isn't enough*" since stakeholders may have preconceived solution ideas. One participant suggested using solution-agnostic questions to encourage users to consider what problems the system should solve and to discuss requirements at domain and product level.

6.2.4 Exploration strategy

Our participants validated the four exploration strategies included in the first version model and described that the strategy is varied throughout development stages and for different prototyping purposes. One participant described the importance of creating several parallel prototyping variants to avoid getting stuck in a single solution too soon. She said that using multiple prototypes helps users' express the direction the system should take, and thereby involves them in determining product scope and requirements. Similarly, another participant described that they "*throw a wide net with many different options*" in initial project stages. In addition, several participants said that building several variants of a prototype is useful for exploring alternative solutions and demonstrating these to stakeholders to "*collect ideas about how to proceed and what path to choose*" and thereby optimise the solution. Furthermore, one participant described that in a later stage, an incremental and *flexible* exploration strategy is needed to "*fit in* [customer requests] *with the concept you have in mind*".

6.3 Multi-case study results: broader validation and adjustments (RQ4)[4]

We validated the prototyping aspects model through our multi-case study with eleven startups. The model was presented to practitioners in the interviews and used in the analysis of

[4] Direct quotes from interviewees are noted within citations and italicized.

Table 7 Overview of the 43 prototyping instances (Nn) described by case companies (N = A-K), categorized by our prototyping aspects model. Described lacks of a facet are denoted with –. Blank cells indicate lack of information in the empirical data

#	Purpose *x when relevant*								Prototype media *interview, mockup, other, sketch, video, soft-Ware*	Prototype scope *low, medium, high*	
	Exploration	Communication	Incremental dev	Quality impr	Product-Solution	Product-Market	Techn. feasability	Usability testing		Breadth	Func. refinment
A1	x								s		-
A2	x	x				x			m	h	l
A3	x		x			x			W	h	h
A4		x			x		x		W	h	h
B1						x			s		-
B2						x			m		-
B3						x	x		W		lh
C1		x							W	h	h
C2	x	x			x		x		m	h	-
C3		x	x						W	h	h
D1	x								m		-
D2		x							s		
D3		x	x			x			W	m	h
D4		x							s		-
E1	x						x		m	-	h
E2			x			x			W	h	h
F1								x	m		-
F2	x	x						x	m	h	-

Table 7 (continued)

#	Purpose *x when relevant*								Prototype media *interview, mockup, other, sketch, video, soft-Ware*	Prototype scope *low, medium, high*	
	Exploration	Communication	Incremental dev	Quality impr	Product-Solution	Product-Market	Techn. feasability	Usability testing		Breadth	Func. refinment
F3								x	W	h	h
F4		x							W	h	h
G1	x				x				s		-
G2	x	x			x				m	h	
G3						x			W		
H1		x			x				v		m
H2			x						W	h	h
H3		x				x			W		
I1						x			vsm		-
I2	x								vsm		
I3	x								i	-	-
I4						x			m	h	l
I5							x		W	l	h
J1	x						x		i	-	-
J2	x						x		W	l	l
J3	x				x				W		
J4	x		x					x	o	h	-
J5	x				x				i	h	-
J6		x							v		

Table 7 (continued)

#	Purpose *x when relevant*								Prototype media *interview, mockup, other, sketch, video, soft-Ware*	Prototype scope *low, medium, high*	
	Exploration	Communication	Incremental dev	Quality impr	Product-Solution	Product-Market	Techn. feasability	Usability testing		Breadth	Func. refinment
J7	x	x					x	x	W	l	m
K1	x	x	x						W	h	m
K2		x	x		x				W	h	h
K3		x			x		x		W	h	h
K4		x			x				o	h	h
K5		x			x				o	h	-

Table 7 (continued)

#	Prototype scope *low, medium, high*			Prototype use					Exploration strategy	
	Visual appear.	Interactive beh.	Data realism	Reviewers *internal, external*	Interact *yes, no*	Appr. *scenario, free*	Environ. *meeting, actual*	single / parallel	Iteration focus: *business, product, feature, optimisation*	Iteration size: *small, medium, large*
A1	h	-		if	yn				p	
A2	h	m	m	ie	n	s	m		bp	
A3	h	h		ie	y	f	a		b	
A4	l	l		i		s			bpf	
B1	l	l	l	e	n		m		b	
B2	m	-	l	e	n		m		bp	
B3	h	lh	-h	ie	y	f	a		pf	
C1	h	h	h	e	yn		ma		b	
C2	m	-		ie	yn		m		bp	
C3		h		ie	y	f			pf	
D1	m			i	n				p	
D2	-	-	-	i	y				f	
D3	h	h		ie	y	f			bp	
D4	m			i	y				p	
E1	l	l		i	y			s	bo	
E2	h	h		i	y				f	s
F1	m	-		ie	y	s	m	p	f	
F2	m	m		ie	yn	s	m		pf	
F3	h	h		e	y	f	a		p	
F4	h	h		e	n		m		b	
G1	l	-		i	y		m		f	
G2	m	m	m	ie	yn	s	m		bp	
G3	lm			e	y	f			bp	

Table 7 (continued)

#	Prototype scope *low, medium, high*			Prototype use					Exploration strategy	
	Visual appear.	Interactive beh.	Data realism	Reviewers *internal, external*	Interact *yes, no*	Appr. *scenario, free*	Environ. *meeting, actual*	single / parallel	Iteration focus: *business, product, feature, optimisation*	Iteration size: *small, medium, large*
H1	m	-	m	e	n	s		s	b	
H2	-	-	h	i	y			s	b	
H3			h	e	yn	f		s	o	
I1	h	-		ie	yn				b	
I2	h			ie	yn				b	
I3	-	-	-	e	n				b	
I4	mh	l		ie	y	f			bf	
I5				i	y				p	
J1	-	-		e	n				bp	
J2	l	l		i	y				f	
J3				if	y	s			p	
J4	h	-	h	if	n		m		bp	
J5	-	-	m	e	n				b	
J6				e	n				b	
J7	h	l		i	y				o	
K1	m	m		i	y				pf	
K2	h	h		ie	y	f			f	
K3	h	h		ie	y	f			bo	
K4	h	h		e	n	s			bp	
K5	l	-		e	n	s			b	

the interview data to categorize the prototyping instances described by our interviewees. In total, we identified forty-three prototyping instances among our startups, see Table 7. Prototyping in these startups ranges from using simple sketches through mock-ups, to using source code to explore and to obtain feedback on early software versions. We have categorised the identified prototyping instances using our prototyping aspects model. In this section, we describe and motivate changes to our model based on our empirical data. The first and the revised (second) version of the model are shown side-by-side in a table in Section 5. We recommend that the reader consults that table while reading this section. Initial results of the prototyping practices of startup Companies A-D are available in a separate publication (Bjarnason 2021).

6.3.1 Purpose

This aspect mostly worked well in discussing and categorising how the startups use prototyping, and only two minor revisions are made to this aspect, namely for the purposes *Communication* and *Validation*.

One of the main purposes of prototyping is "*as a tool for selling products to customers*" (Company B) and is thus used in communicating "*about sales… to make them understand that there is a need*" (Company A). For this reason, Company A had invested "quite a lot of work" in building a high-fidelity interactive mockup (prototyping instance A2) which "*is fake, but quite good… communicates that we know and builds trust… Based on that we get investors.*" (Company A). For these reasons, we adjust the detailing of the aspect of Purpose in our model to highlight that communication includes sales and marketing, thus *Communication & Alignment* is changed to *Communication: Sales, Alignment.*

For *Validation & Testing*, the boundaries between *Market viability* and *Business viability* were hard to determine for the prototyping instances, and the difference between these two is unclear. Instead, we replace these with the terms *Problem–Solution fit* and *Product-Market fit* (Osterwalder et al. 2014). This will more clearly separate between prototyping to ensure that the product matches the needs of users and customers by provides a solution that addressing their problem, and prototyping to validate that customers are willing to pay for your product and thus the existence of a viable market and business opportunities. Different prototyping instances may be used for each dimension. For example, the mock-up C2 was used to "explore what is required [by the market]", i.e. product-market fit, while the beta release C3 "to 2–3 [customers] is used to tune [the implementation] further" (Company C) and thus improve the problem–solution fit. Product-Market Fit is especially important for startups that "*must produce a prototype to be able to confirm that you have buyers, and thus revenue… [before] incurring more costs without knowing that it is sellable*" (Company A).

Furthermore, while we observe that prototyping is not directly used within our startups for the purpose of improving quality, we retain the purpose of *Quality Improvement* in our model. The lack of support for this prototyping option in our case study is likely due to the nature of early startups, rather than lack of relevance in general of this purpose. In startups, the main focus is on establishing a viable business model with a product solution that matches customer needs and problems, and developing initial product version to realise their solution ideas. Thus, prototyping is primarily used for sales & marketing (Communication), internal exploration of the solution domain, feasibility testing of technical aspects of the solution design, and to validate the product-solution and product-market fit of their business model.

6.3.2 Prototype media

When discussing and analysing the prototyping in the startups it became clear that the media used to represent a prototype, e.g. paper or PowerPoint sketches, computer-generated mock-ups, or source code software, was an important aspect to consider. A similar aspect, related to paper prototyping, was discussed by the authors during our original design of the model, but discarded at that point in time since the aspect of prototype scope appeared to be sufficient. However, we reintroduce this aspect in our model since the multi-case study indicates that the choice of prototype media affects the costs and benefits of prototyping. Our interviewees described that they "*have chosen to produce a mock-up due to cost and time aspects*" (Company A). We believe that this is an important aspect of prototyping that can further support practitioners in making informed decisions about the kind of prototype media to use. In particular for early stages of product development, we want to highlight the cheaper and easier kinds of prototype media such as videos and interviewing, which some of our startups find very useful. For example, in prototyping instances H1, J6, I1 and I2, videos are used for sales & marketing (communication) and for validating product-market fit. One interviewee described videos as a way "*to present products far before they are ready*" (Company I). This interviewee also described prototyping instance I3 and interviews as a means of prototyping "*even without anything to show*" by "*talking to people about how they solve this problem today*" (Company I). Similarly, the now growing startup Company J used videos in social media channels for sales & marketing purposes; "*funny videos that became a bit viral*" (Company J). This startup also uses interviewing for exploring the problem domain. In the early prototyping instance J1, they "*asked people* [users]... [and] *that became an important signal*" that their solution idea was viable, and later used a similar approach in prototyping instance J5 to validate their revenue model.

6.3.3 Prototype scope

When discussing the scope of a prototype, the degree of refinement of the visual appearance, interactive behaviour, and functionality worked well. For example, one interviewee described that "*we do not focus that much on interactivity at the beginning*" (Company F). In addition, five interviewees expressed the importance of realistic data, e.g. to capture behaviour around "*errors and bad data*" (Company I). Also, "*it is really important to use that* [realistic data] *otherwise your [customers] probably get stuck on that"* (Company F) and that when "*the data is fairly realistic…* [it] *can be shown to customer without any problems*" (Company G). Another startup that markets advanced AI algorithms described that they "*need accurate data to back up* [our algorithm]", and thus demonstrate their solution by using "*actual* [customer] *data*" (Company H).

The aggregated dimension of broad vs narrow functional scope was harder to convey, though we saw examples of both broad and narrow prototypes covering many vs a few features. For this reason, we revise this aspect to include *Breadth* as a stand-alone dimension, alongside the dimensions that can be refined, i.e. *functionality*, *visual appearance*, *interactive & haptic behaviour*, and *data realism*. We believe this provides a clearer and more easily comprehensible model for describing the scope of a prototype.

6.3.4 Prototype use

This is the aspect where the previous structure was less aligned with how the interviewees described their use of prototypes. The dimension of *usage environment* worked well, even though many of our startups only tested their prototypes in meeting settings. In part, this is due to challenges in accessing the actual usage environment. For example, for Company F where testing in a live environment is "*difficult since our product targets clinics... varies a lot how open they are with this*." Other reasons for this, are likely cost and awareness of when testing in a live environment is important. One interviewee believed that "[environment] *is an important aspect of mobile solutions, to test them in their correct context*" (Company I).

We find that the (previous) dimension of *review method* is not fine-grained enough to categories the identified prototyping instances. Instead, we replace this dimension in our model with the three dimensions *reviewers* (who receives or gives feedback on the prototype), *prototype interaction* (directly with prototype or not, e.g. when demoing), and *review approach* (scenario-based or free). This corresponds better to how the interviewees describe their use of prototypes.

All interviewees described with whom prototypes were used, i.e. *Reviewers,* and often the same prototype is used with multiple kinds of reviewers. For example, the sketches of prototyping instance A1 were "*tried out on ourselves and on people in our proximity, then we started developing*" (Company A). The later more refined "*mockup [A2] then becomes a blueprint for the product to be built by developers*" (Company A). Thus, reviewers are often internal within the startups and development team, as well as, external such as customers, funders, or product users. In addition, some startups described using family and friends (so called *FFF*, Family, Friends, and Foes) for obtaining feedback on prototypes. For example, for Company J, the prototyping instance J3 used "*FFF when we have made something new. What do they think?*" and for J4 they "*grabbed people in the corridors and ask them what they think... [and then] realised that we had missed several things*"(Company J).

There were also variations in how these reviewers could *interact* with the prototype. When demoing, e.g. of sketches or early mock-ups, there is no reviewer interaction with the prototype. There are also examples of choosing not to allow reviewer interaction with the prototype since "[customer] *awareness is too low*" (Company A) and "*the market is still not that aware of their needs...to have the level of understanding to be active* [in giving constructive feedback on premature mock-ups]" (Company E). For more refined prototypes w.r.t. functionality, visual appearance, and interactivity, the users may be encouraged to try the prototypes out for themselves. For example, the fully functioning product prototype of B3 was available to the general public and "*customers get to use our stuff... [allowing the startup to] follow up how they are used in the field*" (Company B). Similarly, users are encouraged to interact directly with the mock-up of prototyping instance F2 "*to see reactions.... how they reason, how they select and react to things*" (Company F), which provides the startup with rich feedback. We have added a dimension of *Interactivity* to our model to cover this dimension.

Finally, while *scenario testing* was included in our initial version of the model, our interviewees primarily described the use of scenarios as an approach that could be applied both when demoing a prototype, or when allowing users to interact directly with the prototype. For example, when demonstrating the paper mock-ups of prototyping instance F1, the startup would "*try to get them [customers] to think about what they do then [in that scenario]* (Company F). Similarly, a startup that provides a technical solution uses scenarios to help non-techie customers "*see the value* [in the solution] …[through] *presenting these cases to them… and avoid just clicking us through*" (Company G). A similar approach is seen for prototyping instances K4 and K5, where the startup "*goes through a simple scenario, explains the challenges, and shows the solution*" (Company K). Thus, we added *Review approach* as a dimension with the options scenario-based or free, where free, covers both free testing, e.g. for beta releases, and demos without any clear scenarios.

6.3.5 Exploration strategy

This final aspect was the hardest to discuss with our interviewees, and the one with the least number of responses with the exception of *parallel exploration*. Using multiple prototype versions in parallel was a clear and understandable strategy, though most of our startups stated that due to resource constraints they tend to focus on one version at a time. As one interviewee said: "*we go with one* [option] *first since we are a small team*" (Company H). Another interviewee mentioned that "*we always try to sketch different [options]*" (Company F) though without giving any concrete example of this (and therefor not seen in any of the reported prototyping instances.) A third interviewee said: "*focus one thing at a time and learn about. But, in some cases* [parallel exploration is useful] … *e.g. for a pricing model*" (Company I). Thus, we have modified the aspect of exploration strategy to include three facets: *single or parallel exploration*, *iteration focus* (*business*, *product*, *feature* or *optimisation* level), and *iteration size*. The two facets related to prototyping iterations were indicated by a few of our interviewees. For example, one interviewed technical lead advocated the "*need to iterate slowly… [to ensure] what the stakeholder originally wished for*" (Company D). Initially, startups often iterate at the level of the product, as was described by the interviewee from Company E: "*we have an idea and want to see if it works.*" And, then move towards "*more gradual development … and more structured feature growth from the very basic experimental ideas up to the ready-for-market products*" (Company E).

The dimension of iteration size replaces the other three exploration options (of the initial version of our model), namely *point-based*, *optimisation*, and *flexible exploration*. The difference between these previous strategies mainly connects to the size of the change between prototype versions. Prototyping with an *optimisation* strategy, and thus with small changes between versions, could be conveyed through giving the Purpose of Quality improvement. Similarly, the difference between point-based and flexible could also be deferred to the size of changes and connected to the stage and maturity of a product or an idea. For example, in the early stages of a startup prototyping could be used for broad exploration of a suitable solution approach, in which case flexible exploration appears suitable. As a startup and their solution approach matures, they are more likely to want to explore more fine-grained variations, e.g. in what features or what user-interface design to implement, in which case point-based exploration, or even optimisation exploration is more relevant.

7 Discussion

We have investigated the current body of knowledge regarding prototyping methodology through a systematic mapping study and a multi-case study of eleven startups. Our research identifies *five main aspects of prototyping* (RQ1) that cover the Why? and How? and What? of prototyping. These aspects can be used to characterize prototyping instances and thereby provide practitioners and researchers with a model (see Section 5) that can help them to reflect on, analyse, and improve their prototyping practices. This was the case at our focus group and in the twelve interviews of our multi-case study where the four aspects included in the initial version of our model were covered, namely *purpose*, *prototype scope*, *prototype use*, and *exploration strategy*. When categorising the prototyping instances described in the interviews of the multi-case study, a fifth aspect emerged, namely the kind of *prototype media* used, e.g. sketch, mock-up, or source code software.

The systematic map on which our model is based consists of thirty-three primary studies and represents a wide set of papers from the past thirty years from the areas of human factors and design, requirements engineering, and agile software development (RQ2). We observe an increase in publication rate during the past decade and of empirically based research (RQ3).

7.1 The aspects of prototyping (RQ1 and RQ4)

7.1.1 Purpose of prototyping

The *purpose* of prototyping can vary and may consist of a combination of reasons. At the heart of prototyping, lies *exploration* of the solution space through experimenting with ideas, gathering feedback, and iteratively detailing, validating, and communicating product requirements. Within agile, prototyping is a known RE practice where requirements are gradually defined, validated, and communicated through prototyping as part of the incremental development process (Ramesh et al. 2010). With a prototype, new requirements can be *elicited* through exploration & learning and *validated* by testing business viability, market desirability, technical feasibility, or usability.

Prototyping also provides a powerful tool for *communicating* with customers and for creating a good communication climate within a development team (Dow et al. 2011) by showing rather than just telling. In addition, for startups prototypes play a vital role in sales & marketing and in obtaining funding for their venture (Bjarnason 2021). However, there is also a risk that prototypes can convey an inaccurate perception of development status and create unrealistic expectations (Lichter et al. 1994). This risk should be considered when prototyping to validate product-market fit to avoid making unrealistic business decisions regarding budget and development plans (Ciriello et al. 2017; Zink et al. 2017).

Prototyping can also support exploring the problem at hand. An additional purpose of prototyping is to specify requirements and to act as a requirements specification, as was mentioned by several of our case companies. This is a topic that requires further research to understand in what contexts a prototype can be used as a specification and what is required of a prototype to fulfil this purpose.

7.1.2 Prototype media

The kind of *prototype media* used affects the cost of constructing it, but also the learnings and benefits that can obtained from using it. Simpler and cheaper media such as paper or interviewing, or computer-based mock-ups can be used with good benefits, especially in the early stages of ideation and product design and development, such as in software startups (Nguyen-Duc et al. 2017). However, our previous research indicates that some startups tend to prefer and strive to use source code software for prototyping since they believe this enables them to demonstrate ability and build trust with customers and investors (Bjarnason 2021). This preference for certain types of prototype media is often due to the skills and previous experiences of prototyping technologies and tools of those involved in the startup, which play an important role when selecting the media used for prototyping (Gupta et al. 2021; Nguyen-Duc et al. 2017). This connection between skill set and prototype media also found in our interview material. For example, while UX designers can quickly produce mock-ups using tools, e.g. Figma, software developers often prefer sketching and exploring their ideas directly in source code software.

While we advocate considering current experience and competence, we also encourage practitioners to consider other factors when selecting the kind of media to use for prototyping, such as what kind of feedback that is sought and the risks involved in using source code software for prototyping. While choosing source code as the prototype media enables quickly getting started with actual development, this also comes with risks related to product quality and to cementing solution ideas too early on. While there may be stakeholders, e.g. sponsors, that are eager to quickly move on to realising and delivering a novel idea, moving too quickly to production source code comes with a risk of having to cancel the project later on (Ciriello et al. 2017). For these reasons, throw-away prototyping is often advocated since companies are then forced to separate between identifying the ‘right’ requirements (through prototyping) and implementing the requirements ‘right’.

7.1.3 Scope of a prototype

The *prototype scope* also affects the cost of prototyping and the type of feedback that can be obtained. Thus, the scope should be selected to match the intended purposes of the prototyping effort. The breadth of a prototype and its degree of refinement for functionality, visual appearance, interactive behaviour, and data realism, can be varied. For example, a high-quality mock-up that covers all the features of a future system with toy examples, has a broad prototype scope covering all major features but with a low degree of functional refinement of these, but with a high degree of refinement for their visual appearance and interactive behaviour, while it provides a low degree of refinement for realistic data (only toy examples). The two dimensions of breadth and refinement (in general) have been described in previous literature using the terms horizontal versus vertical prototyping (Budde and Zullighoven 1990), where horizontal prototyping explores the breadth and a vertical prototyping the refinement of the scope of the software being development. Compared to previous literature, our model also distinguishes between the type of facet that is refined, e.g. visuals or functionality, and provides a more fine-grained terminology for describing the scope of a prototype.

While our focus group participants actively related to three of the facets of refinement, namely functionality, visual appearance and interactive behaviour, they did not mention data richness. However, five of our interviewees in the startups could relate to the importance of using realistic data, mainly to convey realistic scenarios and trust to customers, but also for validating error cases connected to missing or badly formatted data. We interpret the omission of mentioning this facet at the focus group and among the remaining seven interviewees as an indication of low awareness of the importance of the facet of data for prototyping. Since it is also the facet with the least number of supporting references in our mapping study, we interpret this as an indication that further knowledge and research on the role of data richness in prototyping is needed. For example, to what degree does the use of realistic data, e.g., in a demonstration, affect what feedback that can be obtained?

Furthermore, we note that studies report contradicting results on the relationship between prototype scope and the feedback that can be obtained, and thus the purpose that can be fulfilled. For example, for usability testing, one study shows no significant difference in feedback for prototyping with a low versus a high degree of refinement (Sefelin et al. 2003), while other studies suggest that the most cost-effective scope is either a mix of low and high refinement (Liu and Khooshabeh 2003; McCurdy et al. 2006), or to use simple prototypes with a low degree of refinement (Yasar 2007). These conflicting findings indicate that the matter is complex, and that further empirical research is needed to identify the relevant factors of prototyping practice and the environment, and the relations between these.

7.1.4 Use of a prototype

The aspect of *prototype use* covers *with whom* the prototype is presented and reviewed (internal, external, or with family-friends-and-foes), *how* (with or without user interaction, and based on scenarios or more freely), and in which *environment* this takes place, e.g., in a lab setting or in the actual environment in which the actual product is to be used. At our initial case company, prototypes are frequently used internally to try out new ideas. Our focus group participants described using prototypes with customers either through a demonstration without any direct user interaction, or as free testing of a beta version of the software. In general, our startup companies describe using prototypes primarily internally in the early stages of their business venture, and gradually extending the use to family-friends-and-foes and then to external stakeholders such as potential customers and investors.

Direct interaction with prototypes is common for internal use and when the prototype is slightly more refined. Several startups also describe using scenarios both when demonstrating a prototype and when asking users to interact with it as a means to enhance understandability and communicating how their product solution can address user problems. Who and how a prototype is used affects the type of knowledge that can be obtained through the interactions that takes place both with the actual prototype and between the people involved in, e.g. a prototype demonstration.

The obtained feedback can be steered to specific aspects by structuring the prototype demo sessions. For example, the focus group participants described using solution-agnostic questions to focus a prototype demonstration on the problem description, rather than on details in the solution. The interactions around prototypes relate to the cognitive abilities (Cafer and Misra 2009) and communication skills (Ciriello et al. 2017) of those involved and is a complex and interesting area for further research. For example, to investigate how to optimise the feedback obtained from users through communication techniques such as storytelling, and if a smaller, less refined, and thus cheaper, prototype scope can be compensated for by boosting the prototype use through designing extensive and highly realistic usage scenarios.

The environment in which a prototype is used is another facet of use that affects the feedback that can be obtained. Previous research describes the impact of the physical environment (Lichter et al. 1994). We suggest that the digital environment also may play an important role here and is a facet to consider for use of a prototype. In our previous research on digital work environments, we have identified systems interplay and work interplay as two important factors to consider in RE for systems and tools intended to be used in the work place, i.e. the interaction with other systems and with current work practices (Håkansson and Bjarnason 2020). Further research is needed to investigate in which situations a prototype should be used in the targeted digital and physical environment, and in which situations a lab or meeting setting is sufficient to gain the knowledge needed at a specific stage in the development process and for a specific product domain, type of feature, and prototyping purpose.

7.1.5 The exploration strategy

The *strategy* used to explore the solution space affects which prototyping instances that are pursued and how resources are utilised. Previous research identifies four main strategies, namely point-based, parallel, optimisation and flexible exploration (Tronvoll et al. 2017). While all these strategies are relevant to prototyping, we found that when discussing exploration strategy with practitioners a different set of facets is more suitable. This set of facets can also be used to describe all four strategies included in the first version of the model. For that reason, we modify our model to cover *single or parallel exploration* where multiple solution paths are explored at once, the ***size*** and the *focus of the iteration*, e.g. business, product, feature, or optimisation level.

Exploring multiple options in parallel enables optimising, e.g. a detail in the design, or different parameters in a revenue model. While this approach requires more resources, it also allows for delaying decisions on alternative requirement options until more knowledge has been obtained for these. Exploring one solution path or quality aspect at a time through prototyping allows for freely selecting solution options based on current knowledge and as requirements changes. Single option exploration is the most common approach among our startups since it is more cost-effective. The interviewees that did mention using a parallel exploration approach tended to have a background in user-interaction design. This is similar to our focus group participants who described only using single exploration in later development stages, while preferring a strategy of parallel exploration in early stages to keep an open mind to alternative solutions and avoid getting locked into one solution option at an early stage. This corresponds to previous research that found that parallel exploration stimulates innovation (Dow et al. 2011).

We note that previous research identifies a connection between the length of an iteration and the number and quality of solution options, or requirement possibilities, that are obtained (Jaskiewicz and Helm 2018). This highlights an interesting avenue for further research in considering the perspective of time in relation to prototyping scope and considering the costs and benefits of taking multiple short prototyping steps covering only a few requirements for each step, compared to performing fewer iterations on a larger set of requirements for each step.

Furthermore, we note a parallel between prototyping in small optimising iterations with how to manage quality (non-functional) requirements (Berntsson Svensson and Regnell 2015). In both cases, the outcome focuses on evaluating against a range of values, as opposed to a simple pass/fail, and the increased complexity of needing to considering influencing factors (Tronvoll et al. 2017). This complexity in managing quality requirements has also been observed in the context of another agile RE practice, namely the one of using test cases as requirements (Bjarnason et al. 2016). It would be interesting to investigate if this is due to the same underlying characteristics of quality requirements, or if there are other factors in common between prototyping and testing, and, if so, how these two practices relate and can be aligned.

7.2 Threats to validity

We discuss the validity of our study and the presented model in view of descriptive and theoretical validity, and generalizability.

Interpretative and **descriptive validity** concern how reasonable the conclusions are given the data and the extent to which these are objectively described. We judge that both these aspects are high for our study. Several steps were taken to mitigate the risks of misunderstanding and misinterpreting the literature on which our model is based, the focus group participants, and the interviewees. The primary studies were analysed twice, first as part of the initial design by the 2nd and 3rd author who calibrated their views, and then independently by the 1st author, as part of the initial validation (after the focus group). We interpret the fact that this re-analysis only led to minor modifications and improvements of the presented model (see Section 3.3) as an indication of high descriptive validity although further research is required to further strengthen the evidence for our model. The risk of misinterpreting the focus group participants was mitigated using the same kind of independent re-analysis of the transcript. In addition, the results from the literature study and the focus group were presented at the case company, who also reviewed an earlier version of this paper describing the first version of our model without raising any concerns about misinterpretation. Furthermore, the risk of the (single) first researcher mis-interpreting the interview material was partly mitigated by asking the interviewees to read through the summarising memos with selected quotes from the material and the set of identified prototyping instances. The feedback from the interviewees mainly concerned additions, e.g. mentioning more prototyping instances, which were then incorporated in the paper.

Theoretical validity is determined by our ability to capture what was intended. While we believe our model provides a good representation of the existing research on prototyping methodology, there is a risk that we have missed relevant previous research and that there are additional primary studies relevant to the topic of our systematic mapping study. This remains an open threat that could be addressed in future research through triangulating the search results, e.g. through snowballing. Furthermore, there is also a risk that our model does not fully align with current prototyping practices in industry, even though we have taken initial steps to explore and validate this. Thus, there is a risk that our model is not complete and that there are additional aspects of prototyping, and in particular additional facets, that should be included in our model. This is especially relevant to the aspect that were modified in this second version of the model, primarily i.e. *Prototype media*, *Prototype use*, and *Exploration strategy*. While these modified aspects were used in our analysis of the interview material, they were not presented to the interviewees. Thus, further research, in particular of the practical applicability of our model, is needed to strengthen the completeness of our model, e.g. through case studies and other empirical investigations of industrial prototyping practices. Furthermore, our research indicates a number of potential relationships within the entities of the model, e.g. between prototype scope and prototyping purpose, that appear to affect the learnings that can be obtained. Exploring these relationships, e.g. through case studies, quasi-controlled experiments, and extended literature reviews, poses an interesting area for further research that can provide insights into how to optimise prototyping practices from a cost–benefit perspective.

Generalizability of our model beyond our case companies and the area of RE is believed to be medium. Since our model is based on previous research within HF, RE and agile software development, we believe that our model may be valid and relevant beyond the case companies involved in our study. However, further research is required to validate this. In particular, prototyping practices for large and established software product companies need to be investigated since their prototyping practice may differ greatly from those of the smaller and newer companies investigated so far in our research. Furthermore, additional research is needed to understand the lack of empirical data in our case studies on some of the aspects and facets of our model, in particular exploration strategy and quality improvement. The lack of evidence for these in our case studies may be due to contextual difference in how the practice of prototyping is used, but may also be an indication of areas for which increased insight may support practitioners in improving their prototyping practices. For these reasons, we suggest further case studies where our prototyping model is applied to additional companies and organisational contexts to further improve generalizability.

8 Conclusions and future work

While there is a host of research on prototyping within user interaction design, software engineering and agile development, we find only some research on the use of prototyping as a requirement engineering (RE) practice. Also, most of the research on the methodological aspects of the practice of prototyping concerns the scope of a prototype (e.g. horizontal vs. vertical, hi- vs lo-fidelity, paper prototyping, sketching, mockups etc.) rather than considering the overall practice of prototyping (that also includes how and with whom a prototype is used and for what purposes). To address this lack, we have designed a model of prototyping. Our *prototyping aspects model* (PAM) is based on a systematic mapping study of previous research and has been iteratively validated and improved upon through a focus group at one case company and through interviews at eleven startup companies (RQ4). We have identified five main aspects of prototyping (RQ1) that are included in our model, namely the *purpose* of prototyping, the *scope* of a prototype, the *prototype media,* the method of *prototype use*, and the *exploration strategy* used to explore the solution space. In this paper, we provide a description of each aspect and their more detailed facets based on previous research and on empirical data from our case studies. We conclude that research on prototyping methodology has mainly been performed within software engineering (in general) and within human factors & design (RQ2), and that there is roughly the same amount of in-vivo and in-vitro research on this topic (RQ3), although there appears to be an increase of in-vivo research during the past decade.

We believe that our model can support agile development teams in reflecting on their prototyping practices and in making conscious choices regarding how to explore the solution space in an effective way considering their goals and resources. Practitioners are encouraged to consider the following:

- *Purpose of prototyping*: What is to be achieved with prototyping, in general and for a prototyping instance, e.g. learning or validation, communication or optimisation? Select the *prototype scope* and the method of *prototype use* to match the intended purpose based on existing knowledge from research and your own experience.
- *Prototype scope*: To what extent does the prototype need to represent the final product to achieve the intended the purpose? Is a broad representation of all future product features needed, or is a narrower scope sufficient? What level of refinement is needed w.r.t. functionality, visual appearance, interactive behaviour, and data? Depending on the aim, focus on detailing a specific feature (narrow and refined prototype scope) or providing an overall system view (broad prototype scope). Balance the cost of a broader and more refined prototype scope against the possible benefits.

- *Prototype media*: Given the *purpose* of the prototyping and the selected *scope*, consider what *media* that will yield the best learnings and benefits with the least amount of effort. If the purpose is to explore an early idea within the development team, consider using simpler media such as *sketches* either in paper or in digital form. If the purpose is to test the *technical feasibility* of a new component, this will likely be best achieved by prototyping in actual *source code software*. We encourage practitioners to consider simpler forms of media, such as paper sketches, interviewing, and videos that enable exploring ideas very early on in the design and development process. Also, consider the available resources and competences within your team and select the prototype media accordingly. If someone is skilled in producing *computer-based mock-ups*, this is a fast and cheap way to explore and validate product ideas involving user interaction. In contrast, it is quicker for an experienced software developer to validate, e.g. a new algorithm in a technically complex product through prototyping in source code software. Furthermore, consider the overall cost of development from a long-term perspective. In particular, if considering prototyping in source code software, also consider if and how this prototype is to be thrown out or carried on into subsequent development stages with the associated risks to product quality and cementing ideas too early on.
- *Prototype use*: Which stakeholders and user categories can provide the feedback needed to fulfil the purpose? Should they interact directly with the prototype or is a pure demo sufficient? Should a scenarios-based approach be used to increase understandability? Design the review to align with the purpose; to focus on problem or on solution understanding, and on the relevant facets of scope. Consider the usage environment, both physical and digital, and adapt to yield the desired type of feedback, both regarding content and stakeholder representation.
- *Exploration strategy*: How broad is our current potential solution space? Is the main focus currently on product or feature level? What sized changes are suitable to manage between iterations? If in the initial stages of development, consider using a parallel exploration strategy to avoid fixating on a single solution option too early on. Switch to a single exploration strategy as more certainty is gained.

The presented model poses a starting point for further research into prototyping in specific organisational contexts, e.g., for startups, and to explore the relationships between prototyping aspect. The impact of different factors can be studied, and different prototyping practices compared, by categorising prototyping instances using the five aspects of our model and by comparing to other contextual factors. Areas for future research on prototyping practices include the use of prototyping as a specification practice, the effect of realistic data, the interplay between how a prototype is used/reviewed and the obtained feedback, the communication around prototypes, and the influence of cognitive abilities. Through evidence-based guidelines and insight into prototyping, practitioners may be supported in selecting their prototyping practices from a cost–benefit perspective, and thus improve their abilities to effectively elicit, specify, validate, and communicate novel business ideas and requirements. We believe that effective prototyping can help software development organisations to optimise their use of resources to pinpoint and develop successful products.

Finally, since prototyping is used in several areas, such as user interaction design, requirements engineering, software design and development, the practice has the potential to bridge and integrate different development activities throughout the development life-cycle. As such, it is an interesting practice for further research, and in particular, to investigate how prototyping can facilitate a better integration and alignment between different software development activities such as requirements engineering, user interface and software design, implementation, and testing.

APPENDIX A: Focus Group Protocol

The following protocol was used to perform an initial validation of our model at a focus group with practitioners at our initial case company Telavox. The aim of the focus group was to discuss prototyping scenarios (categorised by the initial draft of our model) for five stages of RE with generic and stage-specific questions. The following questions were used for all stages:

- What is a good/bad outcome for this stage?
- How do you ensure a good outcome for this stage?
- What is clear/unclear with this stage?
- What knowledge is required for a good outcome of this stage?
- Which people should be involved in this stage?

Stage 1 – Concept exploration.
Purpose: exploration of problem domain and solutions
Scope: shallow functionality, low degree of refinement
Use: mainly internal usage
Strategy: parallel exploration

Domain- and product-level requirements are elicited by exploring ideas and solution space. Prototypes of shallow functional scope and low refinement are tested with low-cost methods such a paper prototyping. The main focus is on internal learning. Sharing knowledge externally is optional.

Questions:

- Which are the best ways to brainstorm ideas for a new system? What are the pros and cons with these?
- Pros and cons of not having thought of a new idea prior to talking to a customer about a future system?

Stage 2 - Eliciting customer needs.
Purpose: Testing market desirability, Exploration
Scope: narrow and shallow functionality, low degree of refinement
Use: any review method with users/customers.
Strategy: flexible exploration

Domain-level requirements are elicited and market desirability tested. The aim is to understand customer needs with a focus on the role of the system from the users' perspective. Simple prototypes of selected system concepts are designed and presented to users.

Questions

- In this stage, a simple paper prototype can be of use. What can you vs. can you not learn from such a prototype?
- It can feel difficult to present early prototypes for discussions. Do you agree, what pros and cons do you see with this?

Stage 3 – Identify system scope & requirements.

Purpose: Testing market desirability, Exploration (external), Communication (internal)
Scope: shallow functionality, low degree of refinement
Use: any review method, internal and external use
Strategy: point-based exploration

Prototyping is used to identify system scope and requirements. A simple prototype with shallow functional scope and a low degree of refinement is used externally to pinpoint requirements that will satisfy customer needs, and internally to communicate and align regarding requirements.

Question

- Requirements are identified in traditional, as well as in agile projects, but noted in different ways and forms. In traditional RE, requirements are documented in an SRS, often kept in a spreadsheet. Do you experience a need for such documentation in agile projects? How do you achieve this?

Stage 4 – Test and improve system scope & requirements.

Purpose: Usability testing, Test market desirability, Communication (internal & external)
Scope: broad and shallow functionality, low degree of refinement
Use: any review method, external and internal use
Strategy: flexible exploration

Product scope and requirements are communicated, and usability and market desirability is validated. Communication and alignment of requirements between customer and development, and within a project is facilitated by prototypes that act as requirements specifications. Simple prototypes (broad and shallow functionality with low refinement) represent the current understanding. User feedback is captured by demonstrations, scenario testing, or free testing. A flexible exploration strategy is used to develop prototypes based on feedback.

Question

- How early on is it good to test prototypes?

Stage 5 – Confirm system scope & requirements.

Purpose: Communication, Validation of Market desirability and Usability, Optimisation, Incremental development
Scope: broad and mid/deep functionality, low/mid visual refinement
Use: any review method
Strategy: point-based exploration

Prototyping is used to communicate with customers and to agree on system scope and requirements. The prototype is broader and more refined than in previous stage, particular for functionality, and can be a true (throw-away) prototype or an early version of the system.

Question

- When do you need to perform a more formal validation of the requirements and user-interface design for a product?
- What is the difference between performing a formal validation and having a colleague perform the validation?

APPENDIX B: Interview Guide for Multi-Case Study of Startups

The following interview guide was used with entrepreneurs in our multi-case study of software startup companies. The main aim of the interviews was to explore prototyping practices of software startups and contextual factors that may influence these. The interviews were also designed to validate the prototyping aspects identified in our theoretical model (PAM) and explore to the model's usefulness in supporting practitioners to describing their prototyping practices.

Interview introduction – 10 min

1) **Present the study** (purpose and time frame), main researcher, policy for NDA & confidentiality. GDPR paper, recording etc
2) **Interviewee presentation**: current role, main area of expertise, #years at startup/in field, current and previous experience of startup ventures

Contextual characteristics [Business & Tech] – 10 min

4) **Company/startup**: company origin/history, age (years), size (employees, teams)
5) **Product**: domain, type of VP [SW-based product, content, service, experiences, user data Teece 2010)]
5) **[B]**[5] **Business model**: customer type (market/bespoke, B2B), revenue model, channels
6) [B] Show model of **start-up life-cycle maturity** (Klotins et al. 2019)

   a) What stage and status are you currently in?
   b) Describe current main focus and goals for

      i) product development (ideation, building, variations)
      ii) marketing/growth
      iii) operations & customer support

   c) What stages & status have you been through?

7) [B] **Startup challenges & characteristics** (based on (Berg et al. 2018) and (Giardino et al. 2015), grouped by Time & Resources, Business vs Technology focus, Organisation)

   a) How does your startup relate to these, for each category?

[5] Questions primarily for interviewees with the business perspective

RE practices – 10 min

For your current stage:

8) **What are your main requirements sources**: internal/external, Tech/Business focus?
9) **Do you currently have any software development?** Development model (agile, traditional, hybrid), Size (#engineers & teams)
10) **How do you do handle the following?** Techniques based on (Klotins et al. 2019) and (Lauesen 2002) as checklist

    a) Elicitation
    b) Validation
    c) Communication of ideas & requirements (primarily externally to customers and sponsors/investors)

Prototyping – 25 min

11) **What does prototyping mean to you?** Simple sketches, mock-ups, MVPs?
12) **Describe how you use prototyping,** in what stages, for what purposes, with what scope and how.
13) **Present prototyping aspects model. How do you relate to each aspect** in your prototyping practices**:**

    a) Purpose
    b) Scope of prototype
    c) Use of prototype: review method, environment.
    d) Strategy for handling uncertainties

14) **How do you reason concerning the cost-benefit balance for prototyping?**
15) **Do you use any tools for prototyping?**
16) **Does your prototyping approach vary,** if so how and why?

    a) for different purposes, such as eliciting, validating, and communicating?
    b) for different stakeholders?
    c) due to different points-in time, e.g. as your start-up matures?

Future work – 5 min

17) What would you like to improve around prototyping in your startup?
18) What topics/areas/questions/problems within prototyping would you as a startup want research to address & investigate?

Acknowledgements We thank the participants at Telavox and the interviewees at the startups for good collaboration, and for investing their time and engagement in this study. This work was partly funded by the Swedish strategic research environment ELLIIT.

Funding Open access funding provided by Lund University. The research presented in this article was partly funded by the Swedish strategic research environment ELLIIT.

Data availability The dataset for the systematic map of literature and the protocols for the focus group and the case study including interview guide are available on-line (Bjarnason 2021b). Data collected during the focus group and the interviews are not publicly available due to reasons of confidentiality.

Declarations

Conflicts of interests Bjarnason's contribution to this work was funded by the Swedish strategic research environment ELLIIT.
She has no other financial or non-financial interests, or connection to any of the involved case companies, to disclose. Lang's and Mjöberg's initial contribution to this work (the initial mapping study) was part of their MSc project, which was performed at the case company Telavox during 2020. Since completing their MSc project, they have no financial or non-financial interests, or connections to Telavox or any of the other involved case companies, to disclose.

References

Acosta, R.D., Burns, C.L., Rzepka, W.E. and Sidoran, J.L. (1994).A case study of applying rapid prototyping techniques in the Requirements Engineering Environment. *Proceedings of IEEE International Conference on Requirements Engineering* 66–73

Alves, C., Cunha, J. and Araujo, J. (2020). On the Pragmatics of Requirements Engineering Practices in a Startup Ecosystem. 311–321

Arano, T., Chang, C.K., Mongkolwat, P., Liu, Y. and Shu, X. (1993). An object-oriented prototyping approach to system development. *Proc. of 17th Int. Computer Softw. and Applications Conf.* 56–62

Batova, T., Clark, D. and Card, D. (2016). Challenges of lean customer discovery as invention. *2016 IEEE International Professional Communication Conference (IPCC)* 1–5

Bellomo, S., Nord, R.L. and Ozkaya, I. (2013). Elaboration on an integrated architecture and requirement practice: Prototyping with quality attribute focus. *2nd Int Workshop on TwinPeaks* 8–13

Berg V, Birkeland J, Nguyen-Duc A, Pappas IO, Jaccheri L (2018) Software startup engineering: A systematic mapping study. J Syst Softw 144(2018):255–274

BerntssonSvensson R, Regnell B (2015) A Case Study Evaluation of the Guideline-Supported QUPER Model for Elicitation of Quality Requirements. REFSQ 15:230–246

Bjarnason E, Hess A, BerntssonSvensson R, Regnell B, Doerr J (2014) Reflecting on Evidence-Based Timelines. IEEE Software 31(4):37–43. https://doi.org/10.1109/MS.2014.26

Bjarnason E, Unterkalmsteiner M, Borg M, Engström E (2016) A multi-case study of agile requirements engineering and the use of test cases as requirements. Inf Softw Technol 77(2016):61–79

Bjarnason E, Lang F, Mjöberg A (2021a) A Model of Software Prototyping based on a Systematic Map. ESEM 2021:1–11

Bjarnason, E., Lang, F., Mjöberg, A. (2021b) Data set for systematic mapping study on prototyping aspects and protocol for focus group and case study. https://serg.cs.lth.se/experiment-packages/pam/. Accessed: 2023-08-14

Bjarnason, E. (2021). Prototyping Practices in Software Startups: Initial Case Study Results. *29th International Requirements Engineering Conference Workshops (REW)* 206–211

Block, Z. and MacMillan, I.C. (1985) Milestones for Successful Venture Planning. *Harvard Business Review, Sept 1985.*

Bruegger, P., Lalanne, D., Lisowska, A. and Hirsbrunner, B. (2009). Tools for designing and prototyping activity-based pervasive application. *Proc of 7th Int Conf on Mobile Comp and Multim* 129–136

Budde R, Zullighoven H (1990) Prototyping revisited. Proc of COMPEURO 90:418–427

Cafer, F. and Misra, S. (2009). A cognitive requirement specification model. *24th Int Symp on Computer and Information Sciences* 518–521

Chen Chen, Porter, A. and Purtilo, J. (1994) Tool support for tailored software prototyping.*3rd Symp on Assessm of Quality Softw Dev Tools* 171–181

Ciriello, R.F., Richter, A. and Schwabe, G. (2017). When Prototyping Meets Storytelling: Practices and Malpractices in Innovating Software Firms. *Proc 39th Int Conf on Softw Engineering in Practice Track* 163–172.

Derboven J, De Roeck D, Verstraete M, Geerts D, Schneider-Barnes J, Luyten K (2010) Comparing user interaction with low and high fidelity prototypes of tabletop surfaces. Proc NordCHI 2010:148–157

Dow, S., Fortuna, J., Schwartz, D., Altringer, B., Schwartz, D. and Klemmer, S. (2011) Prototyping dynamics: sharing multiple designs improves exploration, group rapport, and results. *Conf on HFs in Computing Systems* 2807–2816

Fairley RE, Willshire MJ (2005) Iterative rework: the good, the bad, and the ugly. Computer. 38(9):34–41

Fern, D.A. and Donaldson, S.E. (1989). Tri-Cycle: a prototype methodology for advanced software development. *Proc of 22nd HICSS* 377–386

Giardino C, Unterkalmsteiner M, Paternoster N, Gorschek T, Abrahamsson P (2014) What Do We Know about Software Development in Startups? IEEE Software. 31(5):28–32

Giardino C, Bajwa SS, Wang X, Abrahamsson P (2015) Key Challenges in Early-Stage Software Startups. Agile Processes in Software Engineering and Extreme Programming 2015:52–63

Goldman N, Narayanaswamy K (1992) Software Evolution through Iterative Prototyping. Int Conf on Software Engineering 1992:158–172

Grevet, C. and Gilbert, E. (2015). Piggyback Prototyping. *33rd Annual ACM Conf on Human Factors in Computing Systems* 4047–4056

Gupta V, Rubalcaba L, Gupta C (2021) Multimedia Prototyping for Early-Stage Startups Endurance: Stage for New Normal? IEEE MultiMedia. 28(4):107–116. https://doi.org/10.1109/MMUL.2021.3122539

Håkansson, E. and Bjarnason, E. (2020) Including Human Factors and Ergonomics in Requirements Engineering for Digital Work Environments. *IEEE First International Workshop on Requirements Engineering for Well-Being, Aging, and Health (REWBAH)* 57–66

Hakim, J. and Spitzer, T. (2000).Effective prototyping for usability. *18th Conf on Computer Documentation* 47–54

Rex Hartson (2019). *The UX Book.* Morgan Kaufmann Publishers

Heisler, K.G., Tsai, W.T. and Ramamoorthy, C.V. (1989). Integrating the role of requirements specification into the process of prototyping: the protospec. *22nd Annual Hawaii Int Conf on System Sciences* 348–357

Hendry DG, Mackenzie S, Kurth A, Spielberg F, Larkin J (2005) Evaluating paper prototypes on the street. CHI 05:1447–1450

Jaskiewicz, T. and van der Helm, A. (2018) Unlocking the Interactive Office: Concurrent Prototyping Approach. *DIS 18* 547–558

Käpyaho, M. and Kauppinen, M. (2015) Agile requirements engineering with prototyping: A case study. *23rd Int Requirements Engineering Conf* 334–343

Karras, O., Unger-Windeler, C., Glauer, L. and Schneider, K. (2017). Video as a By-Product of Digital Prototyping: Capturing the Dynamic Aspect of Interaction. *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)* 118–124

Klotins E, Unterkalmsteiner M, Chatzipetrou P, Gorschek T, Prikladnicki R, Tripathi N, Pompermaier L (2019) A progression model of software engineering goals, challenges, and practices in start-ups. IEEE TOSEM 2019:1–1

Kordon F (1994) Proposal for a generic prototyping approach. ETFA 94:396–403

Lauesen S (2002) Software Requirements. Addison Wesley, Styles and Techniques

Lauesen, S. (2005). *User interface design : a software engineering perspective*. Pearson/Addison-Wesley

Lenarduzzi, V. and Taibi, D. MVP (*2016)* Explained: A Systematic Mapping Study on the Definitions of Minimal Viable Product. *42th Euromicro Conf on Softw Engineering and Adv Applications (SEAA)* 112–119

Lichter H, Schneider-Hufschmidt M, Zullighoven H (1994) Prototyping in industrial software projects-bridging the gap between theory and practice. IEEE Trans on Software Engineering. 20(11):825–832

Lim, Y.-K., Stolterman, E. and Tenenberg, J. (2008). The anatomy of prototypes: Prototypes as filters, prototypes as manifestations of design ideas. *ACM Trans on Computer-Human Interaction*. 15, 2 7:1–7:27

Liu, L. and Khooshabeh, P (2003) Paper or interactive? : A study on prototyping techniques for ubiquitous computing environments. *CHI '03 Extended Abstracts on Human Factors in Computing Systems* 1030–1031

McCurdy, M., Connors, C., Pyrzak, G., Kanefsky, B. and Vera, A. (2006).Breaking the fidelity barrier: an examination of our current characterization of prototypes and an example of a mixed-fidelity success. *Proc of SIGCHI Conf on HFs in Computing Systems* 1233–1242

Nguyen-Duc A, Wang X, Abrahamsson P (2017) What Influences the Speed of Prototyping? An Empirical Investigation of Twenty Software Startups. Agile Proc in Softw Eng 2017:20–36

Nielsen, J. (1993). *Usability engineering*. AP Professional

Olsen D (2015) The Lean Product Playbook - How to Innovate with Minimum Viable Products and Rapid Customer Feedback. Wiley

Osterwalder A, Pigneur Y, Bernarda G, Smith A, Papadakos T (2014) Value Proposition Design. Wiley

Paternoster N, Giardino C, Unterkalmsteiner M, Gorschek T, Abrahamsson P (2014) Software development in startup companies: A systematic mapping study. Inf & Softw Techn. 56(10):1200–1218

Petersen K, Vakkalanka S, Kuzniarz L (2015) Guidelines for conducting systematic mapping studies in software engineering: An update. Inf Softw Technol 64(2015):1–18

Rahman, A., Razek, A. and van Husen, C. (2017) Innovation by service prototyping design dimensions and attributes, key design aspects, and toolbox. *Int Conf on Engin, Techn and Innovation (ICE/ITMC)* 571–576

Raja, U.A. (2009). Empirical studies of requirements validation techniques. *2nd Int Conf on Computer Control & Comm*) 1–9

Ramesh, B., Cao, L. and Baskerville, R. (2010). Agile requirements engineering practices and challenges: an empirical study. *information Systems Journal*. 20 5

Ratcliff B (1988) Early and not-so-early prototyping-rationale and tool support. Proc COMPSAC 88(1988):127–134

Ries, E. (2011). The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses. Currency

Rudd J, Stern K, Isensee S (1996) Low vs. high-fidelity prototyping debate. Interactions. 3(1):76–85. https://doi.org/10.1145/223500.223514

Schneider K (1996) Prototypes as assets, not toys. Why and how to extract knowledge from prototypes. (Experience report). ICSE 96:522–531

Sefelin, R., Tscheligi, M. and Giller, V. (2003) Paper prototyping - what is it good for? a comparison of paper- and computer-based low-fidelity prototyping. *CHI '03 Extended Abstracts on HFs in Comp. Systems* 778–779

Sergio, L.M. (2015) Why is important prototyping? in Human-Computer Interaction (course material), Universitat d'Alacant: http://desarrolloweb.dlsi.ua.es/cursos/2015/hci/why-is-important-prototyping. Accessed: 2023-08-14

Shepherd, D.A. and Gruber, M. (2020). The Lean Startup Framework: Closing the Academic–Practitioner Divide - Dean A. Shepherd, Marc Gruber,. *Entrepreneurship Theory and Practice*

Teece DJ (2010) Business Models, Business Strategy and Innovation. Long Range Planning. 43(2):172–194

Terho H, Suonsyrjä S, Systä K (2016) The Developers Dilemma: Perfect Product Development or Fast Business Validation? Product-Focused Software Process Improvement 2016:571–579

Toffolon, C. and Dakhli, S. (2008). An Iterative Meta-Lifecycle for Software Development, Evolution and Maintenance. *3rd Int Conference on Software Engineering Advances* 284–289

Tripathi N, Klotins E, Prikladnicki R, Oivo M, Pompermaier LB, Kudakacheril AS, Unterkalmsteiner M, Liukkunen K, Gorschek T (2018) An anatomy of requirements engineering in software startups using multi-vocal literature and case survey. J Syst Softw 146(2018):130–151

Tronvoll SA, Elverum CW, Welo T (2017) Prototype Experiments: Strategies and Trade-offs. Procedia CIRP 2017:554–559

Tullis, T.S. (1990). High-fidelity prototyping throughout the design process. *Proceedings of the Human Factors Sciety 34th Annual Meeting* 266

Wiberg, M. and Stolterman, E. (2014). What makes a prototype novel? | Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational. *Proc of 8th Nordic Conf on HCI* 531–540

Wieringa R, Maiden N, Mead N, Rolland C (2006) Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. Requirements Engineering. 11(1):102–107

Yasar, A.-U.-H. (2007) Enhancing experience prototyping by the help of mixed-fidelity prototypes. *Proc of 4th Int Conf on Mobile Technology* 468–473

Zainuddin, F.B. and Liu, S. (2012). An Approach to Low-fidelity Prototyping Based on SOFL Informal Specification. *2012 19th Asia-Pacific Software Engineering Conference* 654–663

Zink, L., Hostetter, R., Böhmer, A.I., Lindemann, U. and Knoll, A. (2017).The use of prototypes within agile product development. *2017 Int. Conf. on Engin, Techn and Innovation (ICE/ITMC)* 68–77