

# Collaborative Practices for Software Requirements Gathering in Software Startups

Rafael Chanin      Leandro Pompermaier      Afonso Sales      Rafael Prikkladnicki  
PUCRS, School of Technology    PUCRS, School of Technology    PUCRS, School of Technology    PUCRS, School of Technology  
Porto Alegre, RS, Brazil      Porto Alegre, RS, Brazil      Porto Alegre, RS, Brazil      Porto Alegre, RS, Brazil  
rafael.chanin@puers.br      leandro.pompermaier@puers.br      afonso.sales@puers.br      rafaelp@puers.br

**Abstract**—Capturing requirements during the software development process has always been a challenge. Usually, there is a customer with a defined problem or a problem to be explored. However, when we talk about software startups, the requirements gathering process changes. The entrepreneurs need to systematically work with hypotheses and experimentations, and test them as quickly as possible, trying to understand whether these assumptions can become requirements for a system or not. In this context, entrepreneurship training programs, specially for technical students, help these entrepreneurs to understand this new way of gathering systems requirements. This paper describes how some of these practices were performed in a program for new entrepreneurs that took place in a science and technology park. Our preliminary results indicate that several collaborative practices can foster the understanding of the software requirement process for software startups.

**Index Terms**—Software Startups, Software Requirements, Collaboration, MVP

## I. INTRODUCTION

Requirements gathering is a critical phase of the requirements engineering process [1]. Startups [2] follow a specific dynamism in which the software development process constantly evolves and changes, specially in relation to software which works with market uncertainties that force it to make rapid and constant changes. Software startups constantly deal with the idea of an unknown customers or user. In order to mitigate this issue, Steve Blank [3] developed a process as a way to provide entrepreneurs with guidelines into managing the customers.

In this study, we present practices commonly used in training programs for new entrepreneurs that are used in the design of the Minimum Viable Product (MVP). Therefore, the main contributions can be summarized as: (i) identification of collaborative practices used in customer identification; and (ii) discussion of the results of applying these techniques with engineering requirements.

The remainder of this paper is organized as follows: Section II presents our case study - *Startup Garage* - and Section III explores the proposed collaborative practices. Finally, we conclude this study with our final remarks in Section IV.

This work is partially funded by FAPERGS (17/2551-0001/205-4).

## II. STARTUP GARAGE

*Startup Garage* is a program developed by the Tecnopuc<sup>1</sup>, a scientific and technology park located inside Pontifical Catholic University of Rio Grande do Sul, in Porto Alegre, Brazil. The goal of this program is to help entrepreneurs with their first steps into the startup world. The program was design to deliver tools and methodologies that participants can use to create and validate their business hypotheses. In addition, it also focuses on developing entrepreneurial skills, business modelling and digital marketing.

Since 2014, we have performed the program once a year, and since then we have had 400 participants, 105 startups, from which 70 are software startups. The program is divided into three stages: (i) problem definition and validation; (ii) business model development; and (iii) operational issues, such as marketing.

## III. COLLABORATION PRACTICES

For this research, we used a field study approach [4] by interviewing 60 software startup teams (4 members per team on average) from the *Startup Garage* program. The instrument used was a semi-structured interview (with both mentors and entrepreneurs) in addition to observing teams working on their project. The interview script was designed to capture the effectiveness of the techniques defined and used by the mentors during the program. Moreover, the interviews helped us to understand the context of each startup and also their maturity stage in regards to their product or service.

One of the biggest challenges mentors faced was to deconstruct students focus on implementation. All teams come with a preconceived idea about their solution. Mentors' role is to "convince" them to take some steps back in order to understand that before developing a solution, they must know who the customer is and what problem they are facing.

The following sections describe some of these collaborative approaches, along with a reflection of the results obtained with each one in regards to requirements gathering. All related concepts, such as *customer development*, *hypothesis definition*, *running experiments*, etc., were presented and discussed with students in the beginning of the program.

<sup>1</sup><http://www.pucrs.br/en/innovation/science-and-technology-park/>

### A. Hypothesis Definition

The hypothesis definition process starts by each team defining the first hypothesis they want to test. In the *Startup Garage* Program, a hypothesis was always formed by a customer (or user) and a problem. These two elements have to be connected, *i.e.*, the problem must be related to the chosen customer/user.

Once hypotheses were defined by all teams, each one had to present their own to the group. Each team had 2 minutes to present their idea and the group had 10 minutes to discuss and give feedback. This is a great opportunity to reshape or even to completely change a hypothesis. The worst thing that can happen to a startup is to spend resources on issues that will not help its development.

### B. Experiment Planning

Once hypotheses were defined and agreed upon, teams needed to design their plan on how to validate them. At this moment, teams could use tools/methods such as: **interview** (talking to potential customers in order to understand their problems); **social media page** (a page that generates content in order to drive traffic); **blogging** (similar to the social media page, but as a blog); **landing page** (a web page that explains the value proposition, and tries to collect some type or currency, such as email); **concierge method** (a manual service for a group of customers); **explainer video** (a short animation explaining how the idea will solve the customer problem).

The definition of which tool/method to use was done by each team. In this stage, teams had to present not only the chosen tool/method, but also the strategy behind it.

### C. Experiment Execution

After planning their experiments, startups had to put it into practice. This activity was performed by each team on their own. Since each experiment could take different timespans, the whole group met weekly in order to evaluate the evolution of each project.

Participants were extremely happy with the way the *Startup Garage* program was organized. One of the participants said: “if we have not had the opportunity to share our hypothesis and get feedback, we would have build the wrong solution”. She continues: “We were confident we knew what the market wanted, and so we thought we knew what had to be built”.

### D. Pitch Sessions

Pitch sessions are meetings where entrepreneurs present their startup ideas to several stakeholders, such as investors, other peers, and mentors in order to get feedback. These feedbacks can be related to the presentation itself, but the biggest focus is on the startup idea itself. The goal is to see if every dot is connected and if the idea makes sense. For instance, an investor could evaluate a startup business model and understand that the proposed model is not viable. In this sense, the startup might need to change not only the business model, but also the solution, since a change in the business model could have an impact on the system requirements.

Pitch sessions were opened to anyone from the community. In addition, there was always opportunities for the audience to collaborate by asking questions or by giving advices. These sessions can be seen as a requirement validation technique called *walk-through* [5]. While a walk-through session is a meeting between stakeholders and developers that walk-through the requirements documentation, page by page, line by line, to ensure that the document represents the complete understanding of everyone about what is to be developed in the specific project [5], in a pitch session, developers (entrepreneurs) seeks, through collaboration with peers, for feedback in order to refine their systems' requirements.

From the participants standpoint, the pitch sessions were really valuable. First of all, there is the challenge of presenting an idea in public in just a few minutes. The presenter must not only practice the speech a lot in order to look confident and not to go over time, but she/he also needs to work on the presentation in such a way that it make sense for anyone in the audience. Moreover, the feedback they received from all stakeholders involved brought a lot of insight on how to move further with their projects.

## IV. CONCLUSIONS

In this study, we presented some collaborative practices that can be used by new entrepreneurs to design their software solution in a software startup. We analyzed the path of 60 software startups that participated in a training program called *Startup Garage*. Interviews and day-to-day observation were used as our research method in order to gather the data about these startups.

Our preliminary results indicate that, in a software startup context, collaborative practices may enhance the requirements gathering process since entrepreneurs rarely envision all aspects that should be taken into account themselves. By sharing and giving and receiving feedback, software startups can increase the chances of aiming at the right target. In this scenario, the target is understanding who the customer is, what problems they have, and what is the best solution.

As future work, we intend to compare software startups that went through a collaborative process and others that did not used these practices. By doing so, we would be able to really measure the impact of collaboration in software requirement gathering for software startups.

## REFERENCES

- [1] F. Anwar and R. Razali, “Stakeholders selection model for software requirements elicitation,” *American Journal of Applied Sciences*, vol. 13, no. 6, pp. 726–738, 2016.
- [2] D. Smagalla, “The truth about software startups: it’s not the size of the budget but how it is used that determines success or failure of the enterprise,” *MIT Sloan Management Review*, vol. 45, no. 2, pp. 7–8, 2004.
- [3] S. Blank, *The startup owner’s manual: The step-by-step guide for building a great company*. BookBaby, 2012.
- [4] A. H. Van de Ven and M. S. Poole, “Field research methods,” *The Blackwell companion to organizations*, pp. 867–888, 2002.
- [5] I. Sommerville and P. Sawyer, *Requirements engineering: a good practice guide*. John Wiley & Sons, Inc., 1997.