Design Thinking and Scrum in Software Requirements Elicitation: A Case Study

Rafael dos Santos Braz(✉), José Reinaldo Merlin, Daniela Freitas Guilhermino Trindade, Carlos Eduardo Ribeiro, Ederson Marcos Sgarbi, and Fabio de Sordi Junior

Centro de Ciências Tecnológicas (CCT), Universidade Estadual do Norte do Paraná (UENP), Bandeirantes, Paraná, Brazil
rafaelsantosbrazpfi@gmail.com
{merlin,danielaf,biluka,sgarbi,fabiodsj}@uenp.edu.br
https://uenp.edu.br

Abstract. Design Thinking is an innovative human-centered methodology that has gained visibility and importance for its great efficacy and efficiency in generating and testing innovative ideas. The present work seeks the application of this method in software development, more specifically in Software Requirements Elicitation. To this aim, the Design Thinking method was adapted by the team through the concepts of the Scrum framework for application in a case study that was conducted within a Brazilian state university. This study has verified this method to design a system for allocating and reserving resources for this university. Through the results obtained, this model can be considered positive, since it was possible to model a solution with 95% of average completeness and 100% of stakeholders' satisfaction.

Keywords: Design Thinking · Scrum · Software Requirements Elicitation · Software Engineering · Prototypes

1 Introduction

In the business environment, it is necessary a different way to be ahead of the competitive and dynamic market. Generally, the organizations chose the innovation processes as its differential. In this context, the innovation aims solutions that meet the customers' needs. An innovative solution looks forward to maximizing the products or services quality and to reduce the development time [19]. However, reaching this goal is a complex task because it is necessary to apply multidisciplinary processes and a deep understanding of the client and its working field [1].

This analysis is not limited to the business area since it is possible to find the same problem and definition in other areas, either academical and organizational. One of these areas is software development, which encompasses considerable aspects of creating systems for commercial or non-commercial purposes.

A. Marcus and W. Wang (Eds.): HCII 2019, LNCS 11583, pp. 179–194, 2019.
https://doi.org/10.1007/978-3-030-23570-3_14

In sectors like that, Software Engineering has ruled and concerned about all steps of building systems, from the specification to the maintenance after the deployment [17].

Software Engineering leads to the use of approaches, models, processes, and methods to design and build technological solutions for a specific group of people. Both developing entity and customers long for some innovation. Exactly to attend it, innovative methodologies have been built and used in software processes [1,17].

The Design Thinking (DT) nowadays practiced has begun in business to improve business innovative processes. DT is a multidisciplinary and human-centered methodology that centers on a target group's needs. It is possible because this approach is design-oriented. Consequently, DT looks forward to the people's experience and well-being [1,19].

Just as innovation itself, the DT approach can be applied in multiple sectors because of its great effectiveness and adaptability and also its high potential [1]. However, only in the current decade, this methodology is being widely employed in software products developing in the worldwide scope [13]. While the same fact is not a completely valid affirmation in Brazil since initiatives in this area have been slowly promoted in universities and market [14].

As indicated by [1], Design Thinking has high adaptability, and Software Engineering can seize it especially on the Software Requirements Elicitation subarea, which is part of a software lifetime and it is like DT process. The elicitation stage is responsible for collecting, documenting, and checking the system requirements. Abstractly, these requirements describe what the software must do [17].

As mentioned above, a software lifetime corresponds to the steps from beginning to after employing. That whole period is covered by a software project that rules and policies the development of the proposed solution. There are some approaches to manage that type of project, and one of them is Agile Software Development – Agile Methods. These methods have grown significantly in importance and popularity since the ending of the twenty century and, in particular, since the beginning of the twenty-first century because of the Agile Manifest ("philosophical soul" of the Agile Movement) in 2001 [10]. Methods like that aim to deliver software solutions more quickly and also ensuring they meet the costumers' often volatile needs [12].

Within the Agile Context, it is worth mentioning the Scrum framework. Scrum is a mature generalized model within which it is possible to creatively and productively manage the conception of a product, besides maximizing its value by applying the guiding principles of Agile Methods [16].

This work provides an application of DT in software development through the application of a unified Design Thinking and Scrum approach in a case study that has aimed to verify this model's applicability to elicit, analyze and manage the stakeholders and users' needs within a Brazilian state university.

The remainder of this paper is organized as follows. Section 2 presents a bibliographic review and related works. Section 3 describes the Design Thinking

method and the Scrum framework. Section 4 describes the research realized, the method adopted and also all its phases. Section 5 presents the whole Case Study conducted, detailing activities and artifacts. Section 6 presents the main results of the study. At last, Sect. 7 presents the conclusions.

2 Background and Related Works

Design as a creative method has been a valuable resource since it was settled as an important factor for organizational differentiation in the second half of the twenty century [18].

Brown indicated in his paper [3] a great first example of applying the Designer's thinking (the design method or Design Thinking): the brilliant inventor and entrepreneur Thomas Edison (1847–1931). Edison was able to create new markets and trends from his ability to imagine how people would long and use his products. That way he could devise products oriented to the customers' needs.

Some decades after Edison, the first generation of DT was started by Bruce Archer (Systematic method for designers [2]) and John Christopher Jones (Design Methods [11]) in 1965 and 1970, respectively. Such studies have been considered the first references to interactive design through multidisciplinary thinking, which is a Design Thinking key requirement [7].

Subsequently, the current DT was born. This approach aims business innovation by creatively soluting problems [18]. Therefore, it is based on the way that designers think during their products conception. This line of reasoning has driven the innovative process, as indicated by Brown [3, p. 85] *"Thinking like a designer can transform the way you develop products, services, processes – and even strategy."* This transformation is due to the fact designers consider a problem everything that impedes the people's experience and well-being. Hence, these professionals identify problems and generate effective solutions [19].

In their paper, Ferreira et al. [8] demonstrate DT in a different context from the business area. The authors indicate this method has been widely used (directly or indirectly) in the health area, frequently in the prevention or diseases treatment stages. In this case, Design Thinking concerns about the interaction between health professionals and their patients, besides checking their real needs and guiding the communication process.

In Grossman-Kahn and Rosensweig's study [10], the authors reported a multidisciplinary design-oriented approach that integrates different innovative methods. It is intentionally directed to Computer Science by employing three different methodologies. The first one is Agile Software Development because it can rapidly build and improve prototypes based on customers' needs during multiple iterations. Furthermore, it deploys the Lean Startup method that predicts importance equivalence between finding the problem and building the solution. The last one is DT for developing the client relationship and efficiently identifying their needs.

According to this train of thought, the authors Alves et al. [1] conducted a case study, in which they directly applied DT in a software project to urban

mobility. In their work, they concluded DT is worth for software development. In another study [5], the writers reaffirm this conclusion through a case study for renewing two Brazilian Army's computerized systems.

Paula [14] and Cavalcanti [6] show academic visions on utilizing DT to educational processes in Brazilian's scope. Palacin-Silva et al. [13] write about a similar practice, which DT is placed in international educational context. In both cases, the authors display positive views of employing DT educationally. However, a review on Brazilian's bibliography suggests that DT has not been widely used yet, either academically or entrepreneurially. Initiatives in this area are considerably more recent than internationally. Cavalcanti [6] indicates Brazilian universities began to deeper study the DT applications only around 2013.

At last, the present paper aims to contribute to the above-described context by displaying the results of an applicability case study of Design Thinking and Agile Development Methods unified carried out in a Brazilian state university. The methodology employed during this study is described in the next section.

3 Design Thinking and Scrum

Design Thinking is an innovative methodology to build solutions based on designers' thinking and three main pillars: (1) Empathy, comprehension of other people's feelings and reactions by imagining yourself in similar situations; (2) Collaboration, work-groups realizing collective activities or searching for a common result; and (3) Experimentation, drawing conclusions through different conditions [8].

The basic process used during the case study conducted has been proposed by the consulting American firm IDEO[1]. That process has been broadly accepted as the contemporary DT approach. IDEO has been providing an already consolidated DT method and also presents a number of success stories [1,3].

Generally, the DT process has four non-linear and high adaptable phases. However, there are various ways to apply the designer's thinking since Design Thinking itself is a methodology that has a significant historical path of ideas evolution [3]. Because of it, DT can be applied to different projects, both in its nature and problem [19]. Figure 1 presents the layout of the DT model created by IDEO.

The immersion seeks to approximate the team and the context of the problem from the point of view of the client and users. For this purpose, a reframing and analysis of the problem are done. After that, there is the Analysis and Synthesis stage to examine the information collected. This moment is dedicated to recognizing who are the fundamental people to the problem. Ideation is the phase in which innovative ideas are generated. For this, some tools are employed to stimulate the agents' creativity and participation. Finally, in Prototyping, these ideas are validated trough prototypes that reduce the abstraction of the ideal solution. The way the prototype will be created depends on the project and the available tools [19].

[1] https://www.ideo.com/.

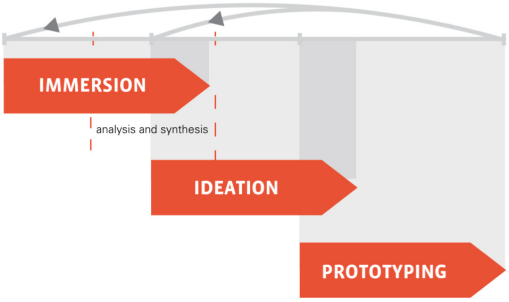

Fig. 1. Stages of IDEO's Design Thinking [19, p. 18].

Agile Methods follow different principles to traditional software development processes, as they have a greater focus on people, and can be satisfactorily adapted to requirements changes [12]. Among the various Agile Methodologies existing, Scrum is a framework that aims to creatively develop products with the highest possible value besides solving complex problems. This methodology has been widely accepted and used both inside and outside universities [16].

Scrum focuses on how the team should interact so that work can be accomplished effectively and efficiently. Through these characteristics of easy comprehension, focus on the agents involved, easy adaptation on requirements, and great management of the software project, the concepts of DT and Scrum were unified in the method adopted during the case study conducted [16].

4 The Method Adopted

According to [9], researches should be classified over its goals and technical procedures. This study can be classified as "Exploratory research" in its goals because it has conceptually detailed Design Thinking and its applications in software development. Besides, it is possible to classify this study as "Case Study" in its technical procedures since it is oriented to explore DT and Agile Methods in the academic context by describing and checking DT applicability to elicit requirements for a real problem in allocating, booking, and organizing university's materials and rooms resources.

Therefore, DT was adapted to the software development context through the Scrum framework. Subsequently, the method adopted was applied to a case study to check if it could satisfactorily specify requirements during the Requirement Elicitation phase.

There are eight different phases in the method adopted, each one of them aiming and manipulating its own artifacts. For the representation of this process[2], the Business Process Model and Notation (BPMN) was used because it

[2] Documentation on https://rafaelsantosbraz.github.io/ModeloDT.github.io/.

is a graphic representation that simplifies and standardizes processes modeling and description that is universally recognized in the literature on Software Engineering.[3]

Figure 2 presents the BPMN representation of the process applied, showing all the artifacts and the steps sequence to be followed during a software project. It is plausible to subdivide the steps of the general model into two larger groups: Preparation and Construction. The first one covers all phases from the initial contact to the end of the Prototyping. Construction takes care of the subsequent steps until the end of the project cycle. This separation is important because the Preparation group is responsible for outlining, understanding, analyzing, generating ideas and modeling the solution. The Construction group focuses on coding the specified solution. Because of that, the activities of the preparation group were fully applied in the case study of this research. The sections below describe these activities.

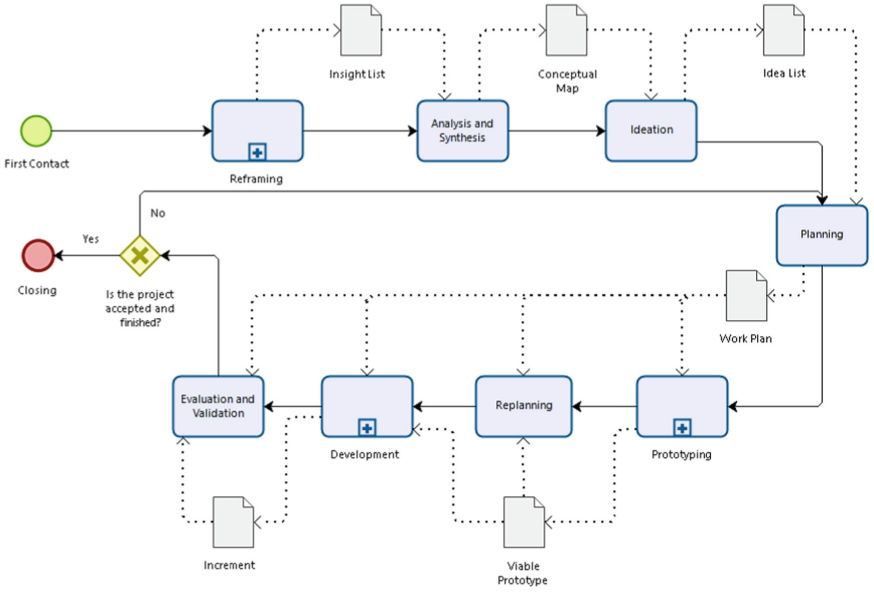

Fig. 2. Graph representation of the method adopted.

4.1 Reframing

The Reframing phase aims to place the work team into the client's work environment and approaches the end users' (client's clients) problem to identify their needs and the opportunities and boundaries of the context studied. This phase

[3] http://www.bpmn.org/.

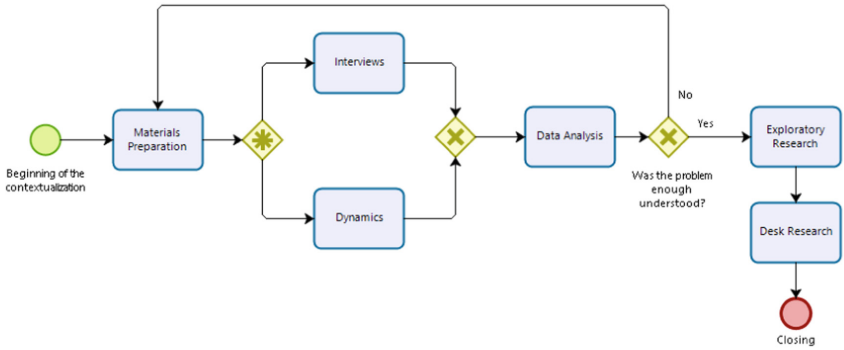

Fig. 3. Activities sequence of the Reframing phase.

is a mix of Preliminary and In-Depth immersions from the first stage of IDEO's DT that searches for different prospects to comprehend the problem identified, besides defining the project boundary.

For this to take place, a number of activities can be done, as shown in Fig. 3. In general, interviews and group or individual dynamics. Besides that, it is necessary to do in-depth research by observing the daily routine of the actors involved. External research for similar cases and their solutions are recommended.

4.2 Analysis and Synthesis

The second phase of the process aims to find patterns, relationships, connections, and challenges that involve the comprehension of the problem established during the Reframing stage. At this moment, it is suggested to synthesize the results obtained. Therefore, a graph representation is selected to ease the understanding of all agents involved. A common way to represent it, it is to create a Conceptual Map that gathers all information generated until the current phase [19].

4.3 Ideation

The Ideation phase is the moment when a large number of new ideas are creatively and innovative generated to solve the problem identified. For this purpose, meetings between the work team and the main actors involved, that can contribute to the ideas generation, is a fundamental activity. Usually, some practices like Brainstorming sessions (all participants are allowed to unrestrainedly contribute new ideas) or Co-Creation Workshops (group dynamics to generate, develop, and present innovative ideas) [1].

4.4 Planning

This is a short-lived phase that acts as a transition moment from all data collected and analyzed to the prototyping process. Then and there, the team has to

define which ideas suggested will be taken forward and implemented. Then, it is necessary to create the initial Work Plan that only concern about the prototyping tasks. This document has a time grid for each team member and indicates what should be done during every day of the present stage.

4.5 Prototyping

During Prototyping, the design team has to develop prototypes (models or schemes that represent the ideas in real life) that ease visually and practically demonstrating the features of the expected final product. Thereby, the team can validate the ideas selected and the way they were implemented in the prototypes, checking if they completely meet the users' expectations and if they are applicable. In accordance with the principles of Design Thinking, it is worth highlighting the prototypes can be elaborated in several different and creative ways, therefore, the team has to choose which way is the best one for meeting its own goals. Figure 4 shows the internal activities of Prototyping which follow the creation-validation precept.

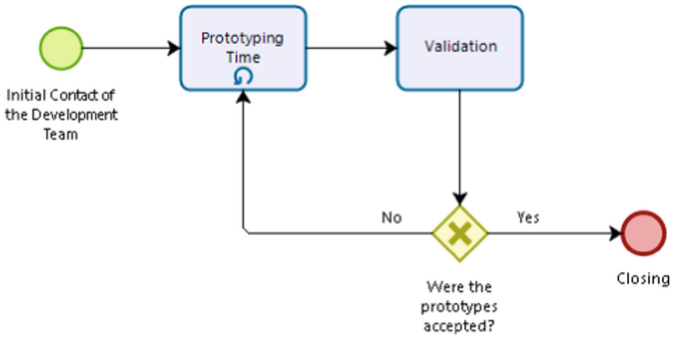

Fig. 4. Representation of the flow of prototyping activities.

4.6 Replanning

This stage is similar to Planning. At this moment, the team has to review and refresh the Work Plan since just now the Viable Prototypes have been made to guide the construction of the real solution.

4.7 Development

The Development phase is a grouping of activities that have as their common purpose the construction of the specified solution. To that aim, the development is based on Viable Prototypes specifications and the Work Plan. At the end of this, an Increment Done will be obtained. This artifact must be a real, concrete and functional application. Figure 5 details the flow of development activities that follow the same precept of Prototyping (Construction-Validation).

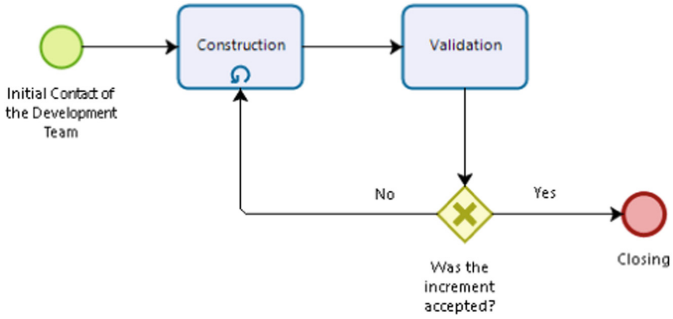

Fig. 5. Representation of the flow of development activities.

4.8 Evaluation and Validation

The current phase is the last one in the Software Project cycle adopted. At this moment, the Increment Done is given to the client and users. Besides this, it is worth pointing a meeting to evaluative review all the main events and actions that influenced the work team or the process. The reviewing could be, for example, a self-assessing and discussion moment to check what actions and behaviors were beneficial to the team and project as well as which ones had better be avoided in the future to optimize the work. The Scrum framework enhance the importance of the review sessions during the whole project [16].

After adapting the Design Thinking process to software development through the Scrum framework, the renewed model was employed in a real situation. The next sections describe the whole case study.

5 Case Study

The case study conducted has covered the Preparation activities group (Sect. 4), which has aimed to verify if the method adopted could be efficiently applied to Software Requirements Elicitation.

The case study was attended by volunteer agents directly or indirectly involved in the problem studied, among them, it is worth citing the participation of professors, students, directors, coordinators, secretaries, trainees, and maintenance, organization and cleaning staff. This multi-stakeholder base follows the Design Thinking precept of observing the problem from the point of view of different people. In this case, the participants were in different hierarchy levels of a Brazilian state university and they acted as clients or users.

At the university in question, the whole process of allocation, reservation, use, organization and control of its resources (material objects or locals as rooms, classes or computer laboratories) was done manually. It generated countless conflicts and a low level of users' satisfaction. Because of that, the team has intended to design a technological solution to help to reduce this problem.

The subsections below expose what activities performed and the artifacts generated during the case study, following the flow presented in Fig. 2.

5.1 Reframing

It was then initiated by the Reframing phase. For this purpose, both individual and collective interviews were applied. The interviews aimed to identify how the manual process of allocation, reservation, and control of the university's resources was realized and what were the main difficulties faced during this process. During the interviews, a simple process of quick conversations guided by a previously prepared questionnaire was followed.

The dynamics applied consisted of conducting activities to better understand the thoughts and the people's point of view about certain aspects as their desires, perceptions, and attitudes. The material used for this was based on the one provided by [15], but translated and adapted by the team. That material[4] is relevant because it creatively stimulates the participants to think about their needs, especially if applied collectively and quickly.

This material is named Hero Profile or Empathy Map and it was chosen because it respects one of the three main pillars of Design Thinking: the Empathy. The work of [4] reaffirms the importance of seeking empathy for clients and users because using this, the team will be able to analyze and understand emotionally and cognitively the needs of its target group. Figure 6 contains a model of the Empathy Map used in the dynamics.

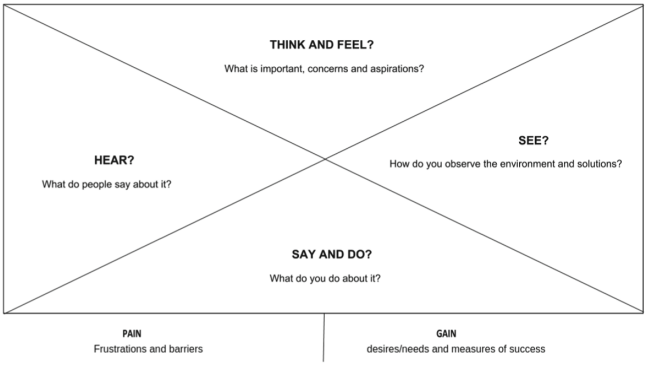

Fig. 6. The Empathy Map model (adapted from [15]).

Subsequently, the team analyzed the data collected during the previous activities. At this point, it was important to extract information that contemplates the needs, opportunities and barriers to the problem studied. This useful information is named insights and the List of Insights is the document that organizes and gathers all those important aspects observed (optimized from [19]).

After the first analysis of the data, the List of Insights drawn by the team had more than thirty items, which were grouped by several themes such as

[4] The original material is available on http://www.theservicestartup.com/.

functionalities, acceptance criteria, and norms. The list below contains some summarized examples of these items:

1. Allow dynamic resource allocation.
2. Solve resource allocation conflicts.
3. Efficiently improve the resource allocation process.
4. Seek to decrease the wasting of resources (resources reserved, but without effective use).
5. Allow resources allocation for maintenance and cleaning periods.
6. Pay attention to fixed periods/times for resource allocation.

Consecutively, an Exploratory Research was conducted to understand the client's work context through a Participant Observation, when the daily routine of the main agents of the process was observed. In the end, some more insights were added to the original list. Among them, these two ones are more important:

1. Increase the communication efficiency among the agents involved.
2. Allow all people that participate in the process to be able to use the generated solution.

After this, Desk Research was necessary to broaden the perspectives and better delineate the project boundaries by finding similar solutions in different sources on the Internet. This research is the last one performed in the Reframing phase. At this point, some more Insights were added to norms and opportunities topics.

5.2 Analysis and Synthesis

After Reframing, the work team had to analyze and synthesize the List of Insights drawn, aiming to find patterns and correlations between its items. The results of this stage are usually arranged in a Conceptual Map that is a graphical visualization (simplified, direct, organized and representative) of the information generated and collected during the previous phase. It is possible to build this map starting it from the central problem and branching the information by the main topics. Therefore, this type of representation is useful to discuss the problem with the stakeholders.

5.3 Ideation

That is a significant phase for the remainder of the study since it is the moment to generate innovative ideas to solve the particular problem. As suggested by the original DT, Brainstorming sessions have been realized with clients and users who could participate.

After the sessions, the Idea List was elaborated, which is a catalog of all the ideas generated since the beginning of the process that will be really implemented. That way, this artifact is responsible for guiding the construction of the

solution and the Planning and Replanning stages. Generally, this list resembles a restaurant menu or a set of playing cards.

It is worth highlighting this artifact is compared to the Product Backlog because it contemplates the insights and ideas generated in a way closer to the description of software requirements, as mentioned above, but they still maintain their own characteristics.

Finally, the Idea List of the case study has been summarized in the following list:

1. Allow the reservation to be altered, performed or canceled in advance and without direct and personal communication.
2. Allow all users to access the solution through an access validation.
3. Allow viewing of resources in a time grid.
4. Ask users to agree to the term of commitment before the allocation.
5. Enable dynamic inclusion and maintenance of resources and users by users themselves.
6. Allow directly accessing the solution on computers and mobile devices that have a connection to the Internet.

5.4 Planning

For the Planning stage, the team has met with the stakeholders to decide which ideas from the Idea List would be covered in the first cycle of Prototyping. The fundamental items were selected and the Work Plan for the Prototyping cycle was written.

5.5 Prototyping

This step is subdivided into a loop of two main tasks to build a Viable Prototype that has been validated as a concrete application to sufficiently meet the stakeholders' needs and expectations. The first task is the Prototyping Time itself, which is a prototype design cycle. As indicated by the Scrum framework [16], this task always begins with a small meeting to review the Work Plan for the specific day.

During Prototyping, several activities were performed. First of all, paper prototypes were created to represent system wireframes and, subsequently, the digital versions of them were made, aiming to improve the functionalities visualization. The focus of this stage was the needs identified and how to improve the usability.

After the Prototyping Time, it was necessary the prototypes created were validated by the stakeholders, thus, applicable tests were applied to evaluate them by the agents involved in the process. If the prototype was not positively evaluated, it would not be considered feasible and the Prototyping step would be repeated.

In this study, the prototyping and validation sequence has been performed twice until the prototypes presented have been considered viable and reached the ideal satisfaction level. The subsequent section contains the analysis of the satisfaction and the results of the study.

6 Analysis of Results

As mentioned above, some interviews and dynamics were applied to understand the problem and to identify needs. During these activities, participants' satisfaction data were collected to compare the manual process to the solution proposed. In the Design Thinking of IDEO, satisfaction has a strongly qualitative perspective and the dynamics applied to the study through the Empathy Map are also qualitative [19]. For that reason, it is important to emphasize here the predominant qualitative items identified in the process of resource allocation that is summarized in the following list:

1. The existence of general problems and conflicts during the entire resource allocation process.
2. The manual process has been not considered effective as communication, agility, and ease.
3. Participants have indicated a negative view of the environment (the entire context of the allocation method).
4. There is no full organization and control in this manual process, and the participants long for improving it.
5. The method is considered too fixed and difficult to make changes on the time grid.

However, it is possible and preferable to unite these qualitative points with the quantitative ones from the interview questionnaires to outline a general scope of the participants' satisfaction. For this purpose, the Likert scale was employed, which normally has five possible answer options. This pattern was adopted due to the fact it is frequently found and recommended in the literature. In this case, the scale values were enumerated and defined in this way: 1–completely dissatisfied; 2–partially dissatisfied; 3–neutral (neither satisfied or dissatisfied); 4–partially satisfied; and 5–completely satisfied.

Following the Likert scale and statistically analyzing the data collected, the satisfaction level of the participants with the manual process is 2 (partially dissatisfied). This value demonstrates that the current process has not met their needs and expectations. The graph in Fig. 7 indicates the complete distribution of satisfaction that was between values 1 and 3.

After the prototypes validation sessions, more specifically when the prototypes were classified as viable, 100% of the participants indicated the value 5 for their satisfaction level with the prototypes presented. To better specify this satisfaction, the completeness of the prototypes was also analyzed. In this case, this value quantitatively represents how much the prototypes meet the users' needs and defined specifications. This completeness and the level of satisfaction were important for the team to evaluate the prototypes as viable.

The completeness calculation was performed through a mathematical relationship between the fulfilled items and the total number of items previously listed together with the new users' requests. Each value found was classified in one of these three categories of completeness: A (95% to 100%); B (70% to 94%); and C (less than 70%). The evaluation team established the prototypes

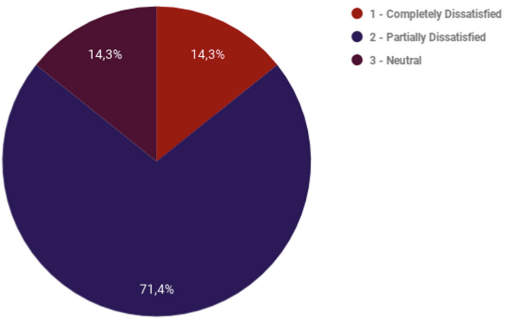

Fig. 7. Participants' satisfaction level with the manual allocation process.

completeness must be in A or B categories (completeness greater than or equal to 70%) to be considered viable, and also at least 80% of the validating agents must be fully satisfied with the prototypes. Figure 8 shows the completeness calculated at the end of the last validation session. As shown, most of the items are above 95% and none of them is below 70%.

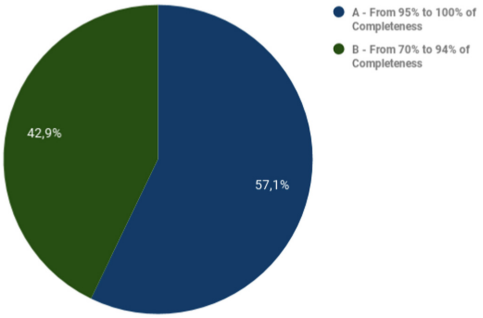

Fig. 8. The final completeness distribution by categories.

7 Conclusion

Design Thinking is an innovative methodology based on identifying people's needs in order to generate innovative solutions. This methodology has a long track of ideas, from the principles of Design to the current method. However, its application outside the business context (the origin of the current DT), more specifically, in software development, became effectively considered only in the late 2000s and at the beginning of the present decade.

DT has been applied in several sectors because this method has a great efficacy and adaptability, and also has a set of well-defined, multidisciplinary and

human-centered tasks. Regarding the application of DT in Software Projects, it is worth mentioning its combination to some principles of the Agile Methods because of their apparent compatibility and common objectives.

This work fits in with this context since it presents and discusses the results of a case study conducted to apply the DT method, adapted through the Scrum framework principles, in the Software Requirements Elicitation to model a software product to solute the problem of allocating resources of a Brazilian university.

Through the quantitative and qualitative results obtained, it is possible to indicate Design Thinking as a positive model to the software development area. The unified approach of DT and Scrum has met the team's expectations by successfully identifying needs, opportunities and project barriers, and also modeling the requirements to design prototypes capable of achieving 100% of users' satisfaction and average completeness greater than 95%. Nevertheless, the results of this case study are preliminary ones and it is necessary to keep developing the model employed and reviewing new cases such as complete systems for larger and more complex problems.

Acknowledgments. The authors would like to thank the Universidade Estadual do Norte do Paraná (UENP) and the Fundação Auraucária for allowing the development of the case study and providing financial assistance.

References

1. Alves, A.T., Lima, A.M., de Oliveira Sales, E., da Costa, A.J.S.: Relato da aplicação da metodologia design thinking no projeto de um software para mobilidade urbana. In: SBC (ed.) SBSI. vol. X, pp. 333–344. Londrina (2014)
2. Archer, L.B.: Systematic method for designers. Council of Industrial Design, London (1965)
3. Brown, T.: Design thinking. Harvard Bus. Rev. **52**(6), 84–92 (2008)
4. Callahan, E.: Cross-cultural empathy: learning about diverse users in design thinking process. In: Stephanidis, C. (ed.) HCI 2018. CCIS, vol. 850, pp. 236–240. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-92270-6_32
5. Canedo, E.D., Parente da Costa, R.: The use of design thinking in agile software requirements survey: a case study. In: Marcus, A., Wang, W. (eds.) DUXU 2018. LNCS, vol. 10918, pp. 642–657. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91797-9_45
6. Cavalcanti, C.M.C.: Contribuições do Design Thinking para concepção de interfaces de ambientes virtuais de aprendizagem centradas no ser humano. Ph.D. thesis, Universidade de São Paulo (USP), São Paulo-SP (2015)
7. Eickhoff, F.L., McGrath, M.L., Mayer, C., Bieswanger, A., Wojciak, P.A.: Large-scale application of IBM design thinking and agile development for IBM z14. IBM J. Res. Dev. **62**(1), 1–14 (2018)
8. Ferreira, F.K., Song, E.H., Gomes, H., Garcia, E.B., Ferreira, L.M.: New mindset in scientific method in the health field: design thinking. Clinics **70**(12), 770–772 (2015)
9. Gil, A.C.: Como Elaborar Projetos de Pesquisa, 4th edn. Atlas, São Paulo (2002)

10. Grossman-kahn, B., Rosensweig, R.: Skip the silver bullet: driving innovation through small bets and diverse practices. In: Leading Innovation through Design: Proceedings of the DMI 2012 International Research Conference. The Design Management Institute (DMI), Boston, MA, USA, August 2012
11. Jones, J.C.: Design Methods, 2nd edn. Van Nostrand Reinhold, New York (1992)
12. Paetsch, F., Eberlein, A., Maurer, F.: Requirements engineering and agile software development. In: International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, pp. 1–6. IEEE Computer Society (2003)
13. Palacin-Silva, M., Khakurel, J., Happonen, A., Hynninen, T., Porras, J.: Infusing design thinking into a software engineering capstone course. In: The 30th IEEE Conference on Software Engineering Education and Training, pp. 212–221. IEEE, November 2017
14. Paula, D.F.O.: Model for the Innovation Teaching (MoIT): um modelo baseado em Design Thinking, Lean Startup e Ágil para estudantes de graduação em computação. Master's thesis, Universidade Federal de Pernambuco (UFPE), Recife-PE, March 2015
15. Pinheiro, T.: The Service Startup: Design Thinking gets Lean: A practical guide to Service Design Sprint, 4th edn. Altabooks, CreateSpace, Hayakawa (2014)
16. Schwaber, K., Sutherland, J.: Guia do Scrum, November 2017
17. Sommerville, I.: Software Engineering, 9th edn. Pearson, London (2011)
18. Valentim, N.M.C., Silva, W., Conte, T.: The students' perspectives on applying design thinking for the design of mobile applications. In: International Conference on Software Engineering (ICSE), vol. 39, pp. 77–86. IEEE/ACM (2017)
19. Vianna, M., Vianna, Y., Adler, I.K., Lucena, B., Russo, B.: Design Thinking: Inovação em Negócios, 1st edn. MJV Press, Rio de Janeiro (2012)