Evolving SPIDe Towards the Integration of Requirements Elicitation in Interaction Design

Jean C. S. Rosa [ITI/LARSyS | *jean.rosa@iti.larsys.pt*]

Beatriz B. do Rêgo [SPIDeLab/IC/UFBA | *beatrizbr@ufba.br*]

Filipe A. Garrido [SPIDeLab/IC/UFBA | *filipe.garrido@ufba.br*]

Pedro D. Valente [ITI/LARSyS & UMa | *pedro.valente@staff.uma.pt*]

Nuno J. Nunes [ITI/LARSyS & Tecnico/ULisbon | *nunojnunes@tecnico.ulisbon.pt*]

Ecivaldo S. Matos [SPIDeLab/IC/UFBA | *ecivaldo@ufba.br*]

✉ *Interactive Tecnologies Institute, Caminho da Penteada, piso -2, 9020-105, Funchal, Portugal.*

Abstract Among the various interaction (re)design processes and approaches, SPIDe is a semio-participatory methodological process inspired by communication-centered design. However, the development of computational solutions is not limited to interaction design. Requirements elicitation is also an integral part of this process. Some SPIDe studies indicated that it is also possible to raise requirements through its application due to its participatory characteristics. This article presents an investigation on the feasibility of SPIDe when applied to requirements elicitation integrated with interaction design, presenting an exploratory case study. From the perspective of different experts, we explain the strengths and needs of SPIDe in supporting requirements elicitation integrated into interaction design. Data were collected through logbooks, semi-structured interviews, and the TAM questionnaire and then analyzed through thematic analysis. The results show that the SPIDe use for requirements elicitation integrated into the interaction design is feasible. Furthermore, they indicated that possible improvements in SPIDe could benefit the development of the computational solutions considering a single application of SPIDe to obtain data for interaction design and requirements elicitation integrated.

Keywords: Participatory Design, Semiotic Engineering, User Participation, Human Factors

1 Introduction

One of the minimum requirements for a good software is providing excellent interaction to its users [Norman, 2013]. Since people will use the software, the designer or developer must know the potential user profile. Software Engineering (SE) and Human-Computer Interaction (HCI) are Computer Science areas that aim to guarantee user satisfaction and software success and even should be considered complementary. However, they are still not well integrated [Seffah *et al.*, 2005].

Some researchers have pointed out that user participation in software design and development can contribute significantly to its success [Abelein and Paech, 2015; Bano *et al.*, 2016]. Among many benefits, participation can favor communication about requirements (concerning Requirements Elicitation) and validate interaction design options (concerning HCI) [Bano *et al.*, 2016]. However, it is not enough, concerning user participation, to put him/her close to the designer or conduct interviews or questionnaires. Instead, user participation facilitates communication between engineers, designers, and users [Muller and Druin, 2003]. In this sense, Participatory Design (PD) has techniques that can help participation and, consequently, potentialize communication [Muller and Druin, 2003].

Participatory techniques have been widely used in HCI for interaction design. However, few participatory techniques are used to elicit requirements [Pacheco *et al.*, 2018]. This may be due to the existence of many techniques, and choosing one, or a set of them, can be a complicated process for designers and engineers [Ogunyemi and Lamas, 2014].

In order to favor the communication between designers and users during interaction (re)design (initially for educational software) and provide an effective methodological process, we conceived SPIDe [Rosa and Matos, 2016]. SPIDe is composed of the association of PD and is theoretically grounded in Semiotic Engineering (SemEng) [Rosa and Matos, 2016]. This association between PD and SemEng gives SPIDe a semio-participatory design characteristic [Baranauskas, 2013; Rosa and Matos, 2016].

Some studies [Rosa *et al.*, 2017, 2018] have indicated that SPIDe could also provide requirements elicitation. Therefore, this paper presents an exploratory case study to *investigate the feasibility of SPIDe when applied to requirements elicitation integrated with interaction design*. The following question was used to guide this research: *is SPIDe effective for integrated interaction design and requirements elicitation?* The presented case study occurred during the software design for a music band. We collected the data through logbooks [McAlpine *et al.*, 2017], semi-structured interviews [Lazar *et al.*, 2017], and the Technology Acceptance Model (TAM) [Davis, 1989] questionnaire. Data analysis was performed through the thematic analysis [Braun and Clarke, 2006] method. This study's main contribution is to provide evidence of SPIDe's effectiveness in eliciting requirements when integrated with interaction design.

We organized this paper into seven more sections. The following section presents the theoretical foundation based on Requirements Elicitation, Participatory Aspects, Semiotic Engineering, and Interaction Design. In the third section, we present SPIDe and research about it. Section 4 is for related work, and Section 5 is for the research methodology. In turn, we present the research results in Section 6, while Section 7 discusses the results, their limitations, and threats to validity. Finally, the research conclusions are presented in Section 8.

2 Theoretical Foundation

Considering that software is built and used by humans, Norman [2013] suggests that humans who build software[1] should understand and therefore satisfy the humans who use the software. In this sense, in this section, we present the theoretical foundation used in the research. We divided this section into the following subsections: in subsection 2.1, we present a short introduction on requirements engineering, followed by subsection 2.2, which deals with participatory aspects that guide our research; the subsection 2.3 is dedicated to present the semiotic engineering theory; finally, in the subsection 2.4 we present our perspective of semio-participatory interaction design.

2.1 Requirements Engineering

Requirements Engineering (RE) is a strictly human activity [Bourque and Fairley, 2014; Wagner *et al.*, 2019]. As it is a human activity, RE is entirely influenced by the subjectivity of both humans that apply it and that participate in it [De Souza, 2006]. The user's needs, desires, and constraints about a real-world problem to solve are called requirements [Bourque and Fairley, 2014]. RE is the term used to characterize systematized requirements handling, which involves elicitation, analysis, specification, validation, and management of software requirements [Bourque and Fairley, 2014]. The requirements are the basis for all software development by driving the expected results with the development of the system/application.

According to Wagner *et al.* [2019], there is no generally accepted theory for RE; therefore, empirical research in RE becomes a critical and challenging task. It is worth noting that the SWEBOK (Guide to the Software Engineering Body of Knowledge) [Bourque and Fairley, 2014] is not an empirically validated theory about RE but was created from the consensus of the people who participated in its production [Wagner *et al.*, 2019].

Identifying who the user is, their activities, desires, and environment and understanding these characteristics are issues that are related, studied, and treated by RE [Nuseibeh and Easterbrook, 2000; Bourque and Fairley, 2014] but not only. Requirements elicitation is the process/moment of finding the problem that needs to be solved [Nuseibeh and Easterbrook, 2000]. This requires excellent communication be-

tween users and engineers, which makes communication the key to eliciting requirements. Therefore, requirements elicitation is also a strictly human process in which collaboration between interested agents must exist.

For Apshvalka *et al.* [2009], the success of requirements elicitation is related to collaboration between different stakeholders and requirements engineers. How this collaboration can happen depends on the techniques, methods, or approaches used by the requirements engineer to facilitate communication between participants [Pacheco *et al.*, 2018; Bourque and Fairley, 2014]. To effectively elicit requirements, the engineer must seek appropriate techniques that facilitate his/her communication with the user because, in some situations, the user does not know how to communicate what (s)he needs or wants. Some authors [Pacheco *et al.*, 2018; Bourque and Fairley, 2014] report that direct contact with users is preferred over indirect contact to avoid noise and distortion in communication. Therefore, participation becomes an essential aspect of the requirements elicitation.

2.2 Participatory Aspects

In this subsection, we present about collaborative aspects of RE and also about PD. These themes guide the participatory aspects that seek to favor collaboration between requirements engineers, interaction designers, and users.

2.2.1 Collaboration & RE

The requirements engineer's role is to elicit, represent, and manage the perspectives of the most varied subjects interested in the software project. Requirement elicitation is not a passive activity, but it should encourage collaboration between users and the requirements engineer [Bourque and Fairley, 2014; Oran *et al.*, 2017]. For Baranauskas *et al.* [2005], RE is not limited to eliciting, representing, and managing requirements, but it is also a process to facilitate communication between requirements engineers and users. In this sense, *communication* becomes one of the essential principles for the effectiveness of requirements elicitation [Bourque and Fairley, 2014; Oran *et al.*, 2017].

To address the relationship between communication and collaboration between users and requirements engineers, we consider the 3c Collaboration Model, presented by Fuks *et al.* [2007]. In their model, the authors indicate that collaboration is the relationship between communication, coordination, and cooperation. For them, communication consists of exchanging messages and negotiating between people; coordination is related to the way people who collaborate are managed; cooperation is about working together to achieve a particular goal.

Considering requirements elicitation, the engineer needs to look for tools that *coordinate* the activities' *communication* between users and engineers to *cooperate* to solve a problem. In this sense, according to da Costa and Pimentel [2017], it is necessary to use appropriate techniques that facilitate communication between the engineer and the user because, in some situations, the user does not know how to communicate what (s)he needs or wants clearly. Therefore, the techniques used must manage the dynamics of communication

---

[1]In this paper, we consider the humans who build software as designers or engineers. Designers when it comes to Interaction Design and engineers when it comes to SE or Requirements Engineering. Humans who use the software, problem owners, stakeholders, or end-users are called users.
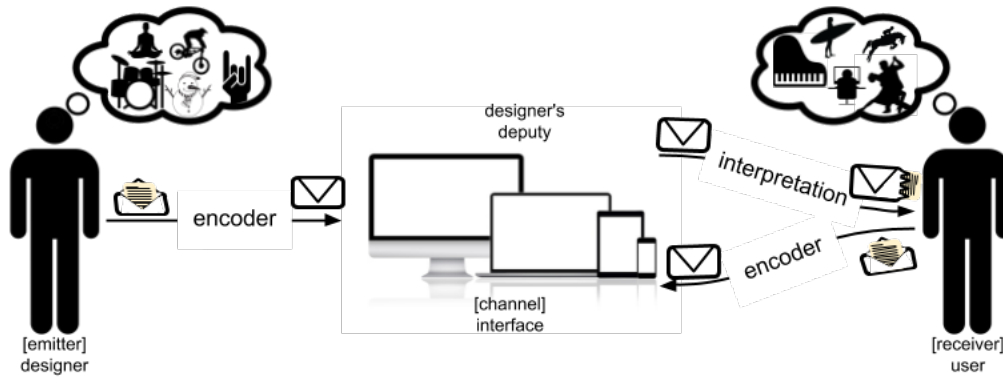
Figure 1. Designer-user communication mediated by interface De Souza [2005]; Rosa and Matos [2016].

(i.e., coordination) to organize the tasks so that there is work together (i.e., cooperation), which requires a conversation so that actions can be taken to resolve problems (i.e., communication). Figure 2 illustrates our perception of RE collaboration considering the 3c collaboration model.
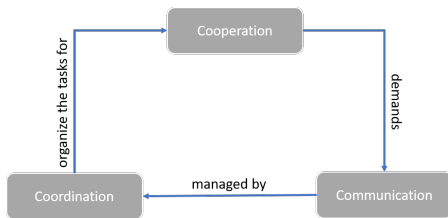

Figure 2. Our perception of RE collaboration considering Fuks *et al.* [2007, 2008].

According to Das [2007], it is indisputable that users' participation in eliciting requirements has positive effects. Das also warns that collaboration between users and engineers in eliciting requirements makes it possible to decrease errors in the elicited requirements. However, we recognize that: *"having a seat at the table is not the same as having a voice; having a voice is not the same as being heard; being heard is not the same as having influence on the outcomes"* [Bano *et al.*, 2018, p. 6].

2.2.2 Participatory Design

Participatory Design (PD) is a way to give users a voice for aroused problem' solutions so that these may be structured according to their perception and collaboration. PD advances User-Centered Design (UCD), in the sense that UCD's ideology is about *make for*, while PD's goes towards *make with* [Luck, 2003; Vilarinho *et al.*, 2019]. So, user participation is not restricted to questionnaires or interviews, but the user should act as co-author of the produced solution.

Six characteristics/concepts as the essence of PD, namely: *(i) equalizing power relations, (ii) situation-based actions, (iii) mutual learning, (iv) tools and techniques, (v) alternative visions about technology,* and *(vi) democratic practices* [Luck, 2018]. According to Luck, PD enables the verbal and nonverbal exchange of ideas that allows emerging knowledge about the user, and their characteristics [Luck, 2003]. It is also important to note that PD implements a democratic philosophy for decision-making since all participants (users,

engineers, designers, etc.) have an equal voice during the solution design. Therefore, with PD, the design is a *social process* [Luck, 2003; Baranauskas, 2013], as the diversity of views expressed by humans during the conception and decision process can influence the development and the project's final results.

It is possible to establish four participatory roles for the user: *(i) user* - the user performs his/her tasks while the engineer seeks to understand the activities that are going on; *(ii) tester* - the user tests the developed artifact and gives feedback; *(iii) informant* - the user is asked about the use of artifacts they already have; *(iv) design partner* - the user can assume the roles of user, tester and informant, and besides, collaborate assuming the role of co-designer and should participate in decision making with the same relevance as a designer [Druin, 2002]. The PD core is genuine participation, which goes far beyond when participants are informants, but they are recognized in the design process, thereby collaborating as design partners [Luck, 2018].

PD has an extensive collection of techniques (*cf.* [Muller *et al.*, 1997]) that can support collaboration through user participation in building a solution. PD and its techniques are widely accepted and used in interaction design and HCI [Ogunyemi and Lamas, 2014]. However, from Pacheco *et al.* [2018] 's systematic literature review, it is possible to identify that few participatory techniques are used for RE. Ogunyemi and Lamas [2014] suggest that one reason for the lack of use of PD in RE is the existence of many techniques, which can take a long time for the engineer and designer to select a suitable one to use in project.

2.3 Semiotic Engineering

Semiotic Engineering (SemEng) is an HCI theory, presented by De Souza [2005], that considers the human-computer interaction as an communication between designer and user through the interface [De Souza, 2005]. However, as the designer cannot be in direct contact with the user at the interaction time, (s)he uses the interface as a deputy. In this sense, the designer designs, encodes, and organizes interface messages. In turn, the user decodes, interprets, and responds as needed. Figure 1 illustrates the designer-user communication mediated by the interface.

According to De Souza, the interface transmits the following generic interaction message:

"Here is my understanding of who you are, what I've learned you want or need to do, in which preferred ways, and why. This is the system that I have therefore designed for you, and this is the way you can or should use it in order to fulfill a range of purposes that fall within this vision." [De Souza, 2005, p. 25].

It is possible to identify four stages of interaction by decomposing this message: (i) the designer analyzes the users and their activities, needs, desires, and environment; (ii) the designer designs the software from his/her interpretation of how users and their activities, needs, wants and environment can or should change (because users want it); (iii) users seek to interpret the designer's message through interaction with the system; and (iv) users finally understand the designer's message and respond to it according to their needs [De Souza, 2005; Leitão, 2017].

The first two stages of interaction can be related to a time that must occur before the interaction, which is the software conception and development. The last two stages are related to the user's contact with the software. Primarily dealing with the first two stages, the designer must know the user in the first one. In turn, the second reflects the moment when the designer and other professionals develop the software from the previous stage's results.

Analyzing the first two stages, it is assumed that the interface should reflect what was developed in the software and, at the same time, the software logic should be aligned with the interaction model that users want, their activities, needs, and environment [De Souza, 2005; Leitão, 2017]. Therefore, HCI can influence RE and vice versa.

2.4 Semio-Participatory Interaction Design

Interaction design models digital things for people's use **?**. Therefore, interaction design is also related to the first two stages of the generic communication message mentioned above. From this perspective, Rosa and Matos [2016] consider that from the communicative perspective presented by SemEng for (human-computer) interaction, interaction design is the construction/manipulation of communication messages. Rosa and Matos [2016] corroborate that Baranauskas [2013] considers interaction design as a social process that designers should not only carry out but also include the users' participation.

In this sense, Rosa and Matos [2016] suggested that software conception and development are not restricted to designers and engineers. However, they also include the user in the role of design partner, turning this semiotic process into *semio-participatory*. The term semio-participatory was presented by Baranauskas [2013] to describe the study of interaction/communication (user-designer or human-computer) through user participation in the design process, associating semiotic approaches with participatory approaches. The expression comes from the combination of particles "semio", which refers to semiotic approaches, and "participatory", referring to participatory approaches. From this semio-participatory perspective, the paradigm of human-computer interaction (or user-designer interaction, according to Se-

mEng) is modified since the user is the co-designer of what (s)he will use when the product is ready. Figure 3 illustrates this paradigm transformation.
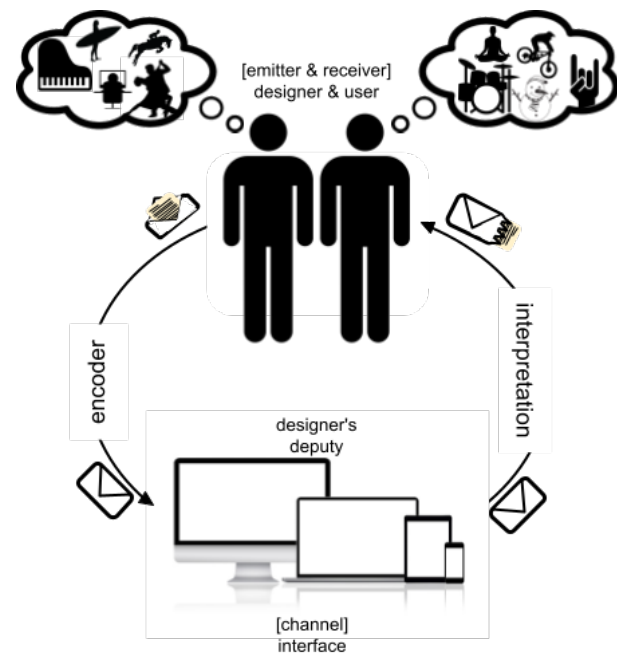

Figure 3. New paradigm of human-computer interaction considering the semio-participatory interaction design Rosa and Matos [2016].

In turn, Rosa and Matos [2016] suggest using SemEng to understand interpretations regarding human-computer interaction. In this sense, the generic interaction message can be a process model for interaction design; the interaction design must correspond to the generic communication message's construction. However, SemEng has no tools for interaction design [De Souza and Leitão, 2009]. So, considering this limitation, the authors proposed SPIDe, the theme of the next section.

3 SPIDe

Considering the association between SemEng and PD, Rosa and Matos [2016] conceived a semio-participatory methodological process for interaction design that has been developed in other studies[2], called SPIDe (**S**emio-**P**articipatory **I**nteraction **De**sign). SPIDe uses SemEng as its theoretical basis, based on the communication-centered design (CCD). In this sense, the process has three stages, as suggested by CCD: (i) contextual analysis, (ii) interface engineering, and (iii) evaluation. Each step comprises participatory techniques, as shown in Figure 4.

Contextual analysis is the first stage of SPIDe and consists of three participatory techniques: contextual inquiry, storytelling, and brainstorming. This stage aims to know the users, identify their contexts, characteristics, problems, wants, and needs, and understand how to solve them. Besides, this is when the designer must understand the solution's impact on the users and their environment. At this point, the designer

---

[2]*cf.* Rosa and Matos [2016]; Pita *et al.* [2017]; Rosa *et al.* [2017, 2018]
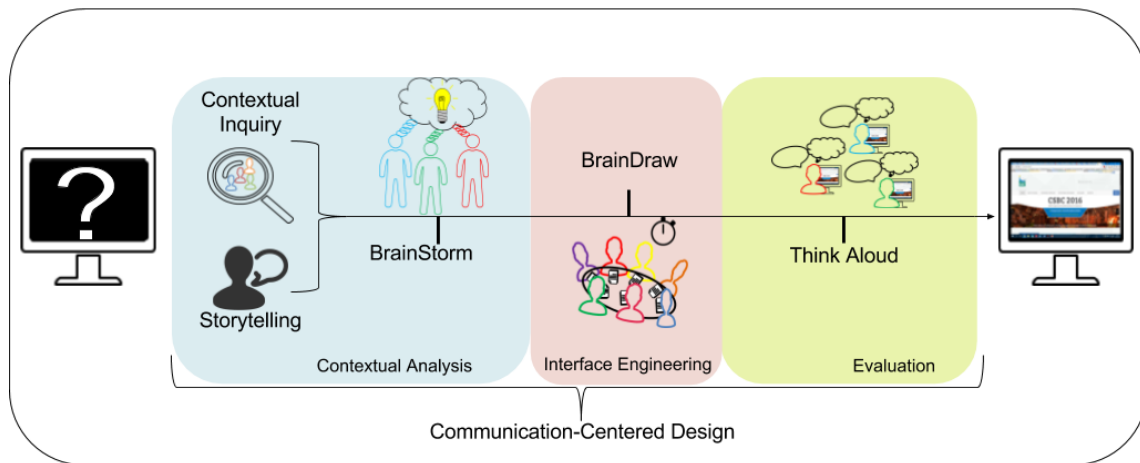
Figure 4. SPIDe process Rosa and Matos [2016]; Pita *et al*. [2017].

also knows the socio-cultural context of users and explores problems and possible solutions with them.

Contextual inquiry is a participatory technique based on ethnography where designers/researchers obtain data by observing how the user is, how (s)he works, and its context [Muller *et al*., 1997; Rosa and Matos, 2016]. In turn, storytelling identifies and clarifies users' problems, desires, and needs by sharing positive and negative stories regarding the proposed theme [Rocha and Baranauskas, 2003; Pita *et al*., 2017]. Finally, brainstorming is a participatory technique that allows participants to share their needs, desires, and vision on improving the activity they perform [Faste *et al*., 2013; Rosa and Matos, 2016].

Analyzing the user's context makes it possible to identify the need for both design and redesign of interaction from the designer's and the user's perspective. Regarding interaction design, the problem is identified and observed during the user's daily life. The designer's role in the design is, after analyzing the data collected in the *contextual inquiry* and *storytelling*, and together with the user, in the *brainstorm*, to discuss the computational solution to be developed. In turn, in the redesign, the designer analyzes the use and the context of the user's interaction to collect data that, after being analyzed, make it possible to improve the quality of the interaction.

The next stage is interface engineering. This stage aims to produce prototypes for the solution provided in the previous stage. For this, the braindraw technique is used [Rosa and Matos, 2016]. In the braindraw, each participant receives a sheet of paper and must draw an interface proposal for a specific time interval and (re)pass it on to the next participant. With this, the participants design the interface of the proposed solution collaboratively. The drawing is done collaboratively, providing the fusion of ideas since all participants are collaborators; that is, the drawing contains characteristics of the meaning systems of all its participants/users/designers [Muller *et al*., 1997; Rosa and Matos, 2016]. In this sense, participants use their signification systems to design an appropriate interface for their use and solution. With braindraw sketching, designers produce mid-fi prototypes[3].

Finally, the evaluation stage aims at evaluating the produced prototype. At this time, users use their signification systems to interact with the solution they have designed through designer mediation. This stage is composed of the think-aloud technique. Think-aloud is a participatory technique of interaction evaluation that enables the user to evaluate the created prototype [Markopoulos *et al*., 2008; Rosa and Matos, 2016]. In think-aloud conduction, the designer prepares an interaction protocol that the participant must try to follow, always verbally expressing their criticisms, suggestions, sensations, and emotions. The interaction evaluation with the prototype through think-aloud makes it possible to identify possible (human-computer) interaction problems before developing the final version.

By associating the SPIDe conduction with the communication message composition of SemEng, we can relate contextual analysis to the first stage, interface engineering to the second stage, and evaluate the last two stages. However, the first two stages are not performed in a *modus operandi make for*, but through PD, with the user as a design partner, in *modus operandi make with* [Rosa and Matos, 2016; Vilarinho *et al*., 2019].

From the associations of SemEng and PD caused by SPIDe, it is possible to establish a collaborative process during interaction design and redesign with users' participation. SemEng's concepts serve as a basis for analyzing the results of each of the PD techniques. The designer's participation in the user's context allows the investigation of the user's daily life and profile.

As a practical guide, the CCD provides interaction design through manipulating messages. These messages are formed by the signs arranged in the interface. The introduction of PD techniques in the CCD process can allow the user to be the manipulator of the interaction messages, allowing for the evaluation and use of a straightforward interpretation of the signs since these signs come from their signification systems and, consequently, from their subjectivity.

SPIDe was conceived as part of the Rosa [2017] master's project. The objective was to understand how the interaction design process of digital educational technology can consider the children's cultural aspects of a school environment. Therefore, according to Rosa [2017], the initial choice of con-

---

[3]*i.e.*, medium-fidelity prototypes.

textual inquiry, brainstorm, braindraw, and think-aloud techniques was due to their uses with children.

Later, Pita *et al.* [2017] conducted a study to improve SPIDe, considering the participation of visually impaired people in the interaction design of software for urban mobility. In carrying out their research, the authors understood that conducting the contextual inquiry technique left participants feeling embarrassed and feeling that the designers were evaluating them. In this sense, Pita *et al.* [2017] suggested adding the storytelling technique so that instead of the participants being observed in their context and problem, they can tell stories about their context and problem. Thus, it is possible to choose the application of contextual inquiry or storytelling during an interactive digital artifact's interaction (re)design.

4   Related Works

Considering theoretical and conceptual aspects and the case study presented in this paper, relations were established with some studies present in the literature [Pereira and Baranauskas, 2015; Arantes, 2013; Mendes and Furtado, 2014]. Figure 5 presents the relationship between related works and SPIDe about semiotic, RE, PD, and HCI through a Venn diagram.
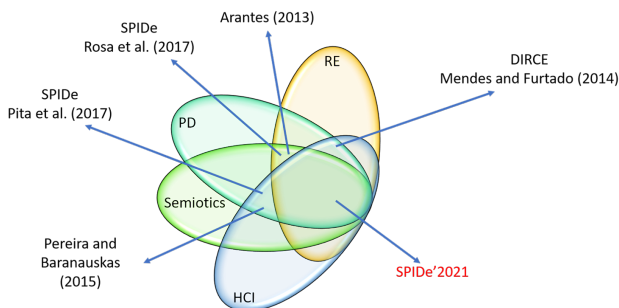

Figure 5. SPIDe and related works.

Pereira and Baranauskas' paper presents a value-oriented and culturally informed approach to interaction design from PD, and Organizational Semiotics from a Socially Aware Design perspective [Pereira and Baranauskas, 2015]. In the approach, the authors make use of five artifacts to support interaction design, namely: (i) the stakeholder identification diagram, (ii) the value identification framework, (iii) the value comparison table, (iv) the culturally aware requirements framework and (v) *eValue*. To fill in the artifacts, the authors use workshops with participatory techniques: brainstorming, interviews, and braindraw.

Despite presenting thematic (culture and values) and innovative artifacts, the application of Pereira and Baranauskas' approach [Pereira and Baranauskas, 2015] is restricted to interaction design, disregarding requirements elicitation and not including a RE model. Also, Pereira and Baranauskas use the semiotic theory of Information Systems applied to HCI, i.e., Organizational Semiotics, like a theoretical basis rather than an HCI theory .

In turn, Arantes [2013] also uses Organizational Semiotics and participatory techniques to elicit requirements of a web-site. In her paper, she characterizes the methodology used as semio-participatory. According to the author, having used a multidisciplinary and human-centered process, the approach facilitated discussion among stakeholders during the workshops, leading to a better understanding of the system's context and social implications. Thus, it was possible to establish a requirements document based on stakeholders' characteristics. However, the semio-participatory methodology is restricted to requirements elicitation, disregarding interaction design.

Mendes and Furtado's paper presents DIRCE (Design of Interaction and Elicitation of Requirements focusing on the Communication and Exploration of ideas) approach [Mendes and Furtado, 2014]. The approach is for requirements elicitation and interaction design. DIRCE also uses PD from brainstorming, questionnaires, interviews, observation, and prototyping techniques. Using these techniques, users participate in the role of user and informant.

SPIDe, in its presentation paper [Rosa and Matos, 2016], sought to value the cultural aspects of users during interaction design, as well as being one of the objectives of Pereira and Baranauskas [2015] approach. SPIDe differs from the approach of Pereira and Baranauskas in having a theoretical-conceptual basis in SemEng, a theory with epistemological limits in HCI, which also differentiates it from the approaches presented by Arantes [2013], and Mendes and Furtado [2014].

SemEng as a theoretical-conceptual basis for semio-participatory interaction design is an unprecedented step for theory since it has no epistemological tools for interaction design. From the generic interaction message proposed by SemEng, we seek to establish a theoretical-methodological relationship between interaction design and requirements elicitation. Finally, SPIDe enables the association between SemEng and PD to favor interaction design and requirements elicitation, leading to epistemological and methodological expansion.

Furthermore, considering that the SPIDe version presented in 2017 by Pita *et al.* [2017] is restricted to interaction design, new researches have been carried out, one of them researching the breadth of SPIDe's scope for requirements elicitation integrated into interaction design (the research subject presented in this manuscript). Initially, Rosa *et al.* [2017] investigated using SPIDe exclusively for requirements elicitation, which concluded that using SPIDe for requirements elicitation is possible. Therefore, expanding the scope of research on SPIDe, in this article, we present the feasibility of SPIDe in interaction design and requirements elicitation integrated. We illustrate in Figure 5 this researches about the expansion in the SPIDe scope.

4.1   Researches around SPIDe

Researches carried out by Rosa and Matos [2016], Pita *et al.* [2017] present applications and modifications of SPIDe under (multi)cultural aspects of schoolchildren and the participation of visually impaired people in the process of interaction design, respectively. The expansion of the scope of SPIDe has conjectured the hypothesis about how SPIDe can elicit software requirements. Therefore, Rosa *et al.* [2017]

presents a systematic literature review to identify whether there are studies that deal with the techniques that make up SPIDe for eliciting requirements. The researchers found works that apply brainstorm and storytelling to elicit software requirements. Therefore, these results indicate the existence of gaps in contextual inquiry, braindraw, and think-aloud to elicit requirements.

In this sense, a study was conducted to experiment with whether SPIDe can elicit requirements (*cf.* [Rosa *et al.*, 2018]). The researchers applied storytelling, brainstorming, braindraw, and think-aloud techniques to elicit requirements for a virtual learning environment. The results show positive evidence of the SPIDe viability for requirements elicitation.

Another research conducted through SPIDe is the design of interaction with/by children with hearing problems (*cf.* [Zabot *et al.*, 2019]). This research had the main objective of adapting a process of interaction codesign to include deaf or hard-of-hearing children (D/HH). The starting point was the analysis of SPIDe in order to identify, adapt and evaluate design solutions for the participation of D/HH in the whole process. Zabot *et al.* [2019] conducted a case study in a public primary school in Salvador/Brazil, with students between six and nine years old. The results of this research were educational games made in partnership with children and indications for modification in SPIDe with an initial step called empathy. This version was called SPIDe Kids.

Finally, another highlight among the studies reports experiences involving long-lived older women in the process of interaction to codesign (*cf.* [Rosa and Matos, 2020, 2019]). This research considered empathy an essential element of the design process, permeating all its stages. Enabled the creation of strategies that favored engagement with these women's effective participation, awakening feelings of empowerment and belonging and contributing to socio-digital inclusion. Like studies with long-lived elderly, these studies broaden the scope of SPIDe by considering design for/by all.

5 Methodology

We performed an exploratory case study [Lazar *et al.*, 2017; Wohlin *et al.*, 2012] to investigate the feasibility of SPIDe when applied to requirements elicitation integrated with interaction design. This research was approved by the School of Nursing's research ethics committee at the Federal University of Bahia[4]. This section details the research planning and execution.

5.1 Planning

Given that this article presents an investigation into the feasibility of SPIDe, the following question guided the research: *is SPIDe effective for integrated requirements elicitation and interaction design?* Effectiveness is understood as doing the right things [Bourque and Fairley, 2014], so it is investigated whether SPIDe meets the proposed goal [Shull *et al.*, 2001] using the logbook [McAlpine *et al.*, 2017], TAM question-

naire [Davis, 1989], semi-structured interviews [Lazar *et al.*, 2017], and thematic analysis [Braun and Clarke, 2006].

The case study of this research was a software project for a gospel music band from a Christian church in Salvador/Brazil. The band leader asked the researchers to create software to help the band's musicians carry out their search and storage activities for songs and chords already played by the band or rehearsal. Consequently, we chose this case for convenience.

This case study is characterized as: *exploratory*, as it studied the research object in its natural context in order to discover what was happening and produce new insights for new work on the theme [Lazar *et al.*, 2017; Shull *et al.*, 2001]; *single*, as it is the study of a single-case studied [Lazar *et al.*, 2017]; *holistic*, given that the case as a whole will be studied [Lazar *et al.*, 2017]; and *instrumental* because the objective is not inherent to the case, but to the research object of the case, as it seeks a broader understanding of the research object and if the case is changed we can conduct the study in the same way [Lazar *et al.*, 2017].

Seven subjects participated in the research, classified into two groups: (a) users and (b) developers. The user group consisted of band members: A (lead and singer), D (guitarist) and E (drummer), and the pastor of the church, J. In turn, the developers' group consisted of a SPIDe applicator (B - Ph.D. student in Computer Science, HCI researcher, SE professor, and RE professional), an interaction designer (Y - Ph.D. student in Computer Science, SE researcher, and HCI professor) and a requirements engineer (N - MSc student in Computer Science, RE professional with over 30 years of experience and researcher in the field). Y and N acted as participants-observers, while B applied SPIDe. It is essential to highlight that only B had previous contact with SPIDe, acting as an observer. The interaction designer and requirements engineer had no prior contact with SPIDe. Table 1 describes the names, groups, and roles of research participants.

Table 1. Participants names, group and role.

Name	Group	Role
A	user	design partner
D	user	design partner
E	user	design partner
J	user	design partner
B	developer	SPIDe applicator
Y	developer	interaction designer
N	developer	requirements engineer

We use three procedures/artifacts to collect data: logbook, TAM questionnaire, and semi-structured interview. The procedures/artifacts are detailed below.

(a) logbook – is an artifact in which participants can write notes, sketches, calculations, feelings, ideas, criticisms, suggestions, observations, and other subjective information that supports the research. Therefore, the logbook is an informal information artifact that can contribute to this research by (i) facilitating cognition and the creation of written results that support participants' reflective practice; (ii) supporting the grouping, synthesis, and interpretation of different kinds of information; and (iii) act as a means of representing information before transformation from informal to

---

[4]Presentation Certificate for Ethical Appreciation number: #22418419.2.0000.5531.

formal [McAlpine *et al.*, 2017].

The seven participants received a sketchbook to serve as a logbook. The sketchbook was delivered at the first meeting and collected at last. The participants used the artifact to collect data about the participants' experiences during SPIDe conduction.

(b) TAM questionnaire – was based on the Technology Acceptance Model that serves to identify subjects' perceptions of usefulness and ease of use of technologies [Davis, 1989]. TAM questionnaire was applied only to developers to identify their perception around the usefulness and ease of use of SPIDe in a four-point scale[5], between totally agree and totally disagree [Laitenberger and Dreyer, 1998]. We do not use a scale with an intermediate point because it does not identify the participant's tendency to agree or disagree with the statements of TAM [Laitenberger and Dreyer, 1998; Marques *et al.*, 2016]. Six affirmations were used to identify the perception of usefulness and six more to identify the ease of use of SPIDe.

In addition to TAM affirmations, the following questions were added to the questionnaire: (i) comment on positive and negative aspects of using SPIDe for interaction design or requirements elicitation with user participation; (ii) was there an interaction design or requirements elicitation aspect that you could not identify through SPIDe? (iii) would you recommend this process to professionals working with interaction design or requirements elicitation? Moreover, (iv) do you think SPIDe helps design interaction and elicit requirements integrated with user participation? These questions were based on Marques *et al.* [2016].

(c) Semi-structured interviews - interviews are valuable procedures for understanding the interviewees' perceptions, concerns, needs, and other reactions [Lazar *et al.*, 2017]. In this sense, an interview was performed individually with each of the users. The interview consisted of questions about (i) difficulties of participation; (ii) collaboration between participants; (iii) the use of subjective aspects such as culture, values, and previous knowledge; (iv) authorship and co-authorship; (v) wants and needs; and (vi) the perception about SPIDe. During the interviews, the audio was recorded for later transcription and analysis.

Qualitative data analysis was performed using the thematic analysis [Braun and Clarke, 2006]. It is possible to organize and describe a dataset coded and categorized into themes through thematic analysis. For this, the analysis is divided into six steps. Figure 6 presents the thematic analysis process. We analyzed the data collected through open questions from the TAM questionnaire, logbooks, and interviews through thematic analysis.

Figure 6. Thematic analysis process Braun and Clarke [2006].

5.2 Execution

SPIDe was conducted on four non-consecutive days. For this conduction, participant B, in agreement with the researchers, decided to apply storytelling instead of contextual inquiry[6], as the contextual inquiry could trigger discomfort among users and developers because the developers should observe users during church services and rehearsals.

Initially, it was planned that each step would be applied in one day. The developers attended every day of the SPIDe conduction. However, on the first day, when storytelling and brainstorming were applied, participants D and A attended, while on the second day, only participants J and E attended. Therefore, due to different participants' presence on the first and second days of SPIDe, B decided to reapply storytelling and brainstorming on the second day. This impacted SPIDe's conduction schedule, which added another day of conduction. In those days, the sketchbooks were distributed to the logbook.

Storytelling was conducted as follows: B asked each participant to think of a positive and a negative experience related to their problems. Afterward, each was asked to share these experiences, and comments could be made during the storytelling. After completing this, B conducted the brainstorm, where the participants were asked to describe, in sticky notes, excerpts of the problems and possible solutions in thirty minutes. All notes were pasted onto a blackboard, and at the end of the given time interval, B started a discussion about each sticky note.

It was defined in context analysis, among other things, that the solution would be a smartphone app. There were discussions about what was thought by users and developers on both days, and the application lasted about one hour. From the results of the contextual analysis, B elicited functional and non-functional requirements. Table 2 presents some elicited requirements.

Table 2. Some requirements elicited by B.

id	Requirements
FR1	The app must make the chorded songs available
FR2	The app must make available songs to be selected for rehearsal and service
FR3	The app must provide music categorization
FR4	The app must make it possible to search for songs by keywords
FR5	The app must provide a discussion place for songs selection for rehearsal and service
NFR1	The app must be available for smartphones, but can also be used in tablets
NFR2	The app must be used by various music bands
NFR3	The app must allow discussion and suggestion of music only to people registered in a particular music band

B performed interface engineering on the third day. Before running braindraw, she informed the users and the other developers about the identified requirements. In turn, users requested that the requirements be written on a board accessible while drawing. Users were given an A4 size sheet with a

---

[5]Totally disagree = -2; partially disagree = -1; partially agree = 1; totally agree = 2.

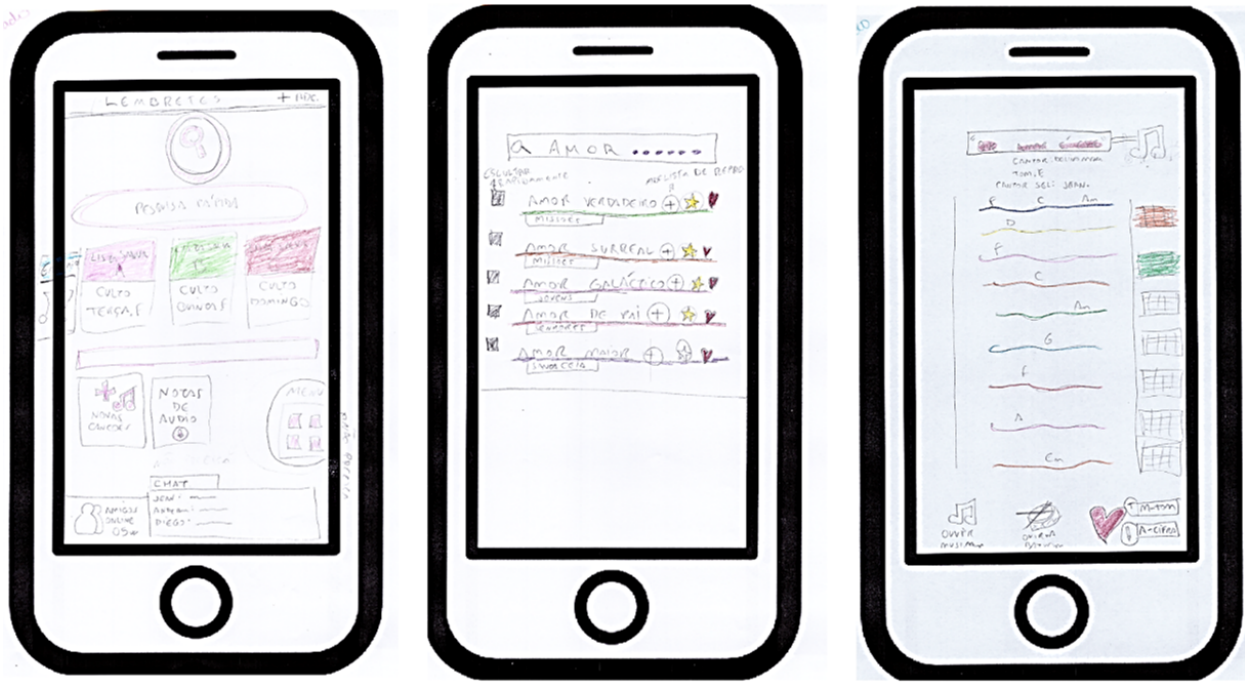[6]Possible change from the paper by Pita *et al.* [2017].

Figure 7. Braindraw draws.

smartphone layout in which they drew the interfaces defined in the previous step: homepage, playlist, and chords songs. Users A, D, and E attended that day. Two minutes was the time determined by B for users to (re)pass the sheets. After the drawings were completed, users chose draws that would be turned into mid-fi prototypes[7]. Figure 7 presents the drawings chosen. This meeting lasted about one hour and thirty minutes.

The fourth day of SPIDe conduction was destined to evaluate the prototypes generated by B and the principal researcher's cooperation from the braindraw drawings. Therefore, only users A and E attended and evaluated the prototypes built by B via think-aloud. A and E were assessed for four and ten minutes, respectively. According to A, *" [the product] is very good [...] with a few clicks you already understand how it works "*. Participants A and B used the following protocol to evaluate the prototypes through think-aloud:

1. view the recommendations and select the recommended song from singer called PG;
2. add the selected song to the song list for the next rehearsal;
3. go back to the home page;
4. view the keyboard and then give up typing;
5. search with the keyword "universo";
6. check the impact level of the song already rehearsed;
7. include the rehearsed song in the song list for Thursday;
8. delete the song "Sou humano" from the list of songs to sing on Thursday;
9. go back to the home page;
10. enter the song list for the rehearsal;

---
[7]*cf. http://bit.ly/2Nci2Ba*

11. mark the song "Meu Universo" as rehearsed;
12. return to home page.

After applying the think-aloud technique, the developers answered the TAM questionnaire and handed the sketchbooks to the researchers. In turn, the researchers contacted the users, scheduled the date and time for the interview, and gathered sketchbooks. However, J did not attend; and E did not write in his sketchbook.

6 Results

After SPIDe's conduction and data collection, the analysis began. The TAM questionnaire result allowed us to identify qualitatively discussed evidence with the thematic analysis.

6.1 TAM Questionnaire Results

As described earlier, the TAM questionnaire affirms that developers should indicate whether they totally agree or totally disagree on a four-point scale about the usefulness and ease of use of SPIDe.

Regarding developers' perception of SPIDe's usefulness, the objective data from the TAM questionnaire show that developers (SPIDe applicator, requirements engineer, and interaction designer) converge to agree about the usefulness of SPIDe for designing interaction and eliciting requirements in an integrated mode, as presented in Table 3. Even developers are unanimous in totally agreeing that using SPIDe was easier to design interaction or elicit requirements with user participation, the affirmation U5. However, the requirements engineer has indicated that it partially disagrees that its effectiveness has increased in U4 (this disagreement will be treated qualitatively from the thematic analysis).
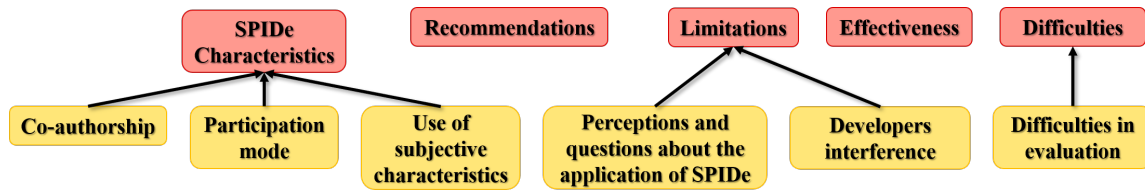
Figure 8. Themes and subthemes. Themes are highlighted in red and the subthemes in yellow.

Table 3. Developers' perception of the SPIDe usefulness.

	-2	-1	1	2
U1. Using SPIDe I was able to design interaction or elicit requirements quickly with user participation.			1	2
U2. Using SPIDe my performance in interaction design or requirements elicitation with user participation has improved.			1	2
U3. Using SPIDe my productivity in interaction design or requirements elicitation with user participation has increased.			1	2
U4. Using SPIDe my effectiveness in interaction design or requirements elicitation with user participation has increased.		1		2
U5. Using SPIDe was easier to design interaction or elicit requirements with user participation.				3
U6. I believe SPIDe is useful for interaction design or requirements elicitation with user participation.			1	2

Table 4. Developers' perception of the SPIDe ease of use.

	-2	-1	1	2
F1. Learning how to design interaction or elicit requirements with SPIDe was easy for me.			1	2
F2. I found it easy to use SPIDe the way I would like.			1	2
F3. I understood what happened while SPIDe was applied.				3
F4. I find it easy to gain skill in interaction design or requirements elicit with user participation using SPIDe.			1	2
F5. I find it easy to remember how to design interaction or elicit requirements with user participation using SPIDe.			2	1
F6. I find the SPIDe easy to use.				3

In turn, the result of developers' perception of SPIDe's ease of use converges to the agreement that SPIDe is easy to use. This convergence points to the developers' unanimity on statement F6, in which they totally agree that SPIDe is easy to use. These results are presented in Table 4.

6.2 Thematic Analysis Results

Regarding the thematic analysis, five themes and six subthemes were identified, as shown in Figure 8, and presented in the following subsections.

6.2.1 SPIDe Characteristics

One of the results of the thematic analysis was the identification of some SPIDe characteristics. Initially, we identify that the **co-authorship** is perceived by users and developers. In response to the interview, user A answered as follows: *"I do not consider myself an author because there are many things that are not mine either; they are from colleagues [...]. The author no, collaborator yes [...] have my opinions, have my drawings, ideas that maybe I suggested and that may end up helping the project"*. In turn, D corroborates with A by declaring: *"I believe it was based on our problems and our solutions. [...] the ideas were mainly our"*. According to Y,

collaboration and communication between users were effective. Thus, it is possible to show that through SPIDe, users assume co-designers or design partners' roles, as indicated by the PD.

In addition to user co-authorship, it was also possible to identify the **participation mode**. SPIDe conduction meetings stimulated users' creativity because, according to user D, *"as the meetings went, it became clear what we needed"*. D also said that after they identified the problems, together with the developers, *"it was easier to find a solution"* because the solution was identified *"debating problems and questioning"*. In turn, A believes this is due to the time granted and because the participants had *"freedom to say whatever we think"*.

These results indicate that users are free to expose stories pertinent to identify problems, needs, and desires (regarding storytelling). Also, they can discuss these requirements to find and propose an appropriate solution (regarding brainstorming), this being performed in a *"[...] natural way"*, according to A. This result reaffirms what is presented by Rosa and Matos [2016] regarding the objective of contextual analysis and the collaborative mode that occurs.

Regarding interface engineering, the braindraw technique was the most positive for users and developers. N declares *"I found the technique of drawing exchange among the participants interesting"*. In turn, B believes that the technique *"[...] integrates users in a fun way, supporting both design and requirements"*. Through braindraw, user A informs that he could have in the interface *"things that I think are cool, that in my opinion work very well within the application proposal"* and that everyone could *"participate directly in the*

creation [of the application]". In this sense, it is possible to evidence that the funny mode of braindraw enables users not only to be testers or informants but also to act directly in how the solution will be built, considering that the interface influences and is influenced by the system logic [Seffah *et al.*, 2005].

Users' significant creativity comes with the **use of subjective characteristics**, a subtheme evidenced in the thematic analysis. Two features were highlighted in this subtheme: cultural aspects and previous knowledge. According to E, his cultural characteristics were introduced to the prototyped application due to braindraw draws. It was also stated by participants A and D. User D, for example, points out that he introduced to drawings *"my way of seeing things, in the creation of the application, principally"*.

In this conduction of SPIDe, culture not only influenced braindraw. B expresses concern about the culture by writing the following in her logbook: *"Does using regional words in participants' speech get in the way? Are developers aware of cultural terms?"* However, the other developers do not report any difficulties in understanding due to regional/cultural expressions (but this does not imply that it could not have happened).

Besides, previous knowledge was used to design and evaluate prototypes. D mentions the following when designing the playlist interface: *"I associated one thing with another, and I found it interesting to put in the created app [...] was from an app that has Spotify*[8]*, which before you listened to music or downloaded the music, you can listen to the best part of the song. Then you press, play for 5 seconds, and immediately get familiar with the music. [...] I put some practicalities that I had observed that some apps did not, and others did"*. The previous knowledge associated with the cultural characteristics of the users is elements that form the signification systems, studied by SemEng regarding human-computer interaction [De Souza, 2005], as well as characteristics associated with PD [Muller and Druin, 2003; Luck, 2003] since the users produce drawings that represent appropriate interfaces according to their previous knowledge and cultural characteristics.

Moreover, the use of users' subjective characteristics and their co-authorship, **other SPIDe characteristics**, was evidenced during the thematic analysis. According to N, SPIDe provides *"quick requirements elicitation and evaluation [...] gain time"*. Also, according to Y, *"[SPIDe] activities have high flexibility and require few resources to perform"*. D corroborates Y by stating that *"I thought the shape of the dynamics you made quite interesting."*. In this sense, users agreed that identifying functionality and *"create screens"* are satisfactory.

6.2.2 Difficulties

Despite the characteristics presented, it was possible to identify some difficulties regarding the SPIDe conduction through thematic analysis. Initially, it is highlighted that there were **difficulties in evaluation** of the prototype. The difficulties reported by developers related to users talking

---

[8]https://www.spotify.com/br/

while interacting with the prototype, which sometimes they have no idea of what users thought or were doing. Y points out in the logbook that *"the user spent more than 2 minutes without pronouncing what he was doing"*. B indicated that E *"did not speak during the interaction"* and did not follow the proposed protocol.

In turn, through the interview, E reported the following: *"I felt it was a little difficult because I do not use these devices much, but then I caught the rhythm [...] I do not use these things very much, and I do not know how to identify each little point. However, with time, we are able to [do]"*. E's report corroborates with Y, which suggested that the user had difficulty recognizing some signs in the interface. However, it is also possible to relate this difficulty to using previous knowledge. As the user has no practice using smartphones, he had difficulties performing the evaluation.

In addition to the difficulty of E, user A also states that he had *"difficulty [...] at the beginning of the application evaluation [...], but after two clicks, I understood how it worked. It was just the first impression"*. Due to these difficulties, B points out that the interaction evaluation was not fully satisfied.

Other **difficulties** were characterized by users and developers. For Y, *"synchronizing participants attendance"* is one of the most severe difficulties and impacts the entire process. In this regard, B questions whether different participants' presence on the first and second day of application may negatively reflect the other stages of SPIDe.

Regarding the braindraw, A states that he had difficulties in creation due to the *"very little time"* given by B. According to the user, the two minutes given made it difficult because he had to understand and continue the drawing that was being passed, so this restricts creativity to "*think about a cool organization*". This challenge with time has been reported by Rosa and Matos [2016].

6.2.3 Limitations, Recommendations & Effectiveness

The developers also characterized limitations. One of the subthemes about limitations is the **perceptions and questions about the application of SPIDe**. In this subtheme, developers' questions were grouped as follows: (i) is reading aloud the think-aloud protocol a standard technique? (ii) which HCI quality attribute is being evaluated? (iii) should the evaluation cover all requirements? The papers about SPIDe [Rosa and Matos, 2016; Pita *et al.*, 2017] do not deal with the reading of the think-aloud protocol and do not address the HCI quality attribute evaluated using the technique; we believe that the evaluation should cover all possible requirements.

Still, regarding interaction evaluation, B pointed out that think-aloud data collection is similar to the Communication Evaluation Method (CEM). CEM is a method proposed by SemEng to evaluate the communicability of interactive computational artifacts with user participation [De Souza and Leitão, 2009]. Therefore *"wanted to tag"* the phenomena that happened during the CEM interaction evaluation.

In addition to the perceptions and questions, it was identified that there was **developers interference** in SPIDe conduction. N described in the open questions of the TAM ques-

tionnaire that *"some requirements were elicited with the help of the developers (N and Y) after the moderator [B] applied the storytelling and brainstorming technique"*. In the logbook, N reports that *"they asked some questions at the end and these questions helped identify requirements"*. B mentioned that the *"developers discussion highlighted important points"*. Because of this interference, it is hypothesized that N partially disagrees (in TAM's affirmation U4) that using SPIDe has increased its effectiveness in interaction design or eliciting requirements with user participation, as the developers held a "debate" that was not predicted in SPIDe.

Other limitations were also identified in the analysis. According to Y, braindraw designs tend to be similar. N pointed out that the process is *"interesting for small projects and few users"* because, according to her, it would be challenging to manage braindraw activities in a medium or large project.

In addition to the themes already presented, it was also possible to identify some **recommendations** made by the developers. One recommendation is to create an artifact for users to reflect on and define the requirements that are being elicited, which need to be in the drawings, and that need to be validated during the evaluation.

Another recommendation is that the user should be advised on how think-aloud works and establish a standard that the protocol is also read aloud during the evaluation. Y even suggested training users before the evaluation.

Some data collected in the case study provides evidence that **SPIDe is effective** for interaction design and requirements elicitation implemented in an integrated mode. E and D's answers corroborate with A. According to A, it was possible to spell out their wants and needs, and the prototyped product meets those wants and needs. User E further described: *"I got it from the drawings I made when I saw in the app that they were"*. D reports that *"the application was created with our ideas [...] supplied the needs of what we had there at the time created"*.

Y confirms that SPIDe delivers on its promise, and *"has plenty of potentials"*. The developers also agree with the users, as B reports that *"SPIDe meets interaction design"* and that *"requirements were easy to identify"*. N expresses *"I found the model very interesting and useful. With this model, it was possible to elicit the requirements and design a prototype solution that was fully adherent to the needs and expectations of users"*.

7 Discussion, Limitations, and Threats to Validity

This investigation's findings explain that developers and users agreed with the effectiveness of the semio-participatory design process. The results show that SPIDe effectively elicits requirements while executing the interaction design process. Hence, it is inferred that its execution can optimize software development, as the process occurs in an integrated way. However, the requirements engineer did not indicate that SPIDe contributed to increased efficiency like others.

The emphasis on stating that SPIDe can optimize software development lies in the confluence of the objective of interaction design (in its initial phases) and the requirements

elicitation, which is to identify/raise data for software design with the highest adequacy in quality aspects, whether in use or interaction. Therefore, SPIDe is a helpful tool for the requirements engineer and the developer (designer) to carry out their activities, including a multi-professional team or from the same working area.

SPIDe can positively impact software development projects. For example, concerning the requirements elicitation, we indicate that one of the main gains of SPIDe is its non-hierarchical participatory character. It allowed the (intended) user to effectively assume the designer's role without being inhibited due to technical limitations to the software development process since their contributions are essential to identify the software and interaction requirements.

SPIDe is a process that is constantly evolving since its conception (see subsection *Researches around SPIDe*), so in this execution, we identified aspects that need attention. Through thematic analysis, we detected which characteristics, limitations, and recommendations should be incorporated or modified for/in SPIDe.

The case study presented in this manuscript is characterized as a feasibility study and is therefore limited to providing sufficient evidence to justify more studies about SPIDe [Shull *et al*., 2001]. That the semio-participatory design process should evolve into a framework; thus, this study's goal is not to obtain a definitive answer to the research question but to allow the construction of an acceptable body of knowledge for project continuation. In this sense, the study also stands out as exploratory, seeking to recognize some phenomena and identify new research questions and hypotheses to improve SPIDe [Lazar *et al*., 2017; Wohlin *et al*., 2012].

In addition to the questions that emerge as a research opportunity for/about SPIDe, through data analysis, we can reiterate that SPIDe is related to the six characteristics/concepts [Luck, 2018]. SPIDe, as a semio-participatory methodological process, uses techniques that seek to give participants an active voice and resources fairly and democratically, with the realization of activities in their natural context, according to thematic analysis results, allowing the mutual construction of knowledge. With this, it is also possible to establish that SPIDe favors genuine participation as users assume the design partners' role.

An essential aspect of SPIDe is that it is noticeable or engaged by the participants throughout its execution. The techniques employed in the contextual analysis and interface engineering phases connect everyone involved in a common objective, the resolution of the proposed problem, or software creation, for example. We can see this with the participants' efforts to contribute to the braindraw, including complementing the other participants' drawings. During braindraw, this engagement is also observed when it is discussed which interface will be converted into a prototype.

The thematic analysis carried out within the scope of this investigation provided an opportunity to identify the characteristics, limitations, and recommendations of/for SPIDe. As a result of the qualitative analysis, research opportunities on SPIDe were presented: (i) how does co-authorship influence the quality of requirements? (ii) to what extent do the subjective characteristics of users influence the developers' understanding of their activities? (iii) what is the ideal amount of

time for users to go through the worksheets during the brain-draw? (iv) is it more beneficial to use think-aloud or CEM to assess the interaction? how to motivate users to speak during think-aloud? (v) is it necessary to add a technique that meets this need in case of interference from the developers? (vi) is SPIDe more effective on small projects, as suggested by the requirements engineer? (vii) what kind of artifact is suitable for defining and consulting requirements? (viii) can SPIDe favor reducing execution time with the team working in a single process to collect data with all stakeholders' contributions, including the customer? (ix) can the overall project costs be reduced using SPIDe?

The research has some well-known areas for improvement, especially in qualitative studies, such as the difficulty in replication, making it difficult to generalize the thematic analysis results. Also, in the presented study, some threats may influence the results, such as the interpersonal aspects of the participants. Some of these threats have been mitigated when possible. The followings are some of these threats and some actions to mitigate them:

Construct Validity: in the TAM questionnaire, words such as satisfaction, effectiveness, and efficiency can be ambiguous. To mitigate this threat, the researchers placed notes on the questions explaining what they considered for each word.

Internal Validity: during SPIDe conduction, some of the users could be prevented from participating (for several reasons unrelated to the research). For example, activities were repeated in the initial days because the users who attended were different.

External Validity: one of the threats was the tiredness of the participants. Both developers and users came directly from their work activities to participate in the research. Researchers proposed meetings of no more than one hour and a half to mitigate this threat, and a snack was served before the activities began. Another threat is that the developers were graduate students (masters and doctorates). However, everyone has proven professional experience in their activities.

Conclusion Validity: in this study, only seven subjects participated due to the size of the project and the characteristics of the study (exploratory and feasibility). The results are considered evidence, as there are limitations to generalization. It is planned to replicate the case study and conduct more research to strengthen the evidence found.

8   Conclusion

This article has presented an exploratory case study that investigated the feasibility of applying SPIDe to requirements elicitation integrated with interaction design. The thematic analysis allowed the identification of SPIDe characteristics due to user participation as a design partner, limitations, recommendations, difficulties, and indications that SPIDe effectively realizes interaction design integrated with requirements elicitation, thus demonstrating the research viability.

This study's evidence can contribute to HCI and RE, as it explored integrating those complementary areas through a semio-participatory design process. Through thematic analysis, it was also possible to identify that the product generated through SPIDe satisfies the wishes and needs of users. Also,

developers find SPIDe useful and easy to use.

Future works are intended to: (i) investigate an artifact that can facilitate access and manipulation of requirements during context analysis, interface engineering, and evaluation; (ii) identify how to mitigate difficulties in the SPIDe evaluation stage; (iii) investigate whether there is a need for a change in SPIDe to systematize the interference of developers making questions to users; and (iv) investigate the use of SPIDe with large teams and projects with a broader scope.

Declarations

Authors' Contributions

Jean C. S. Rosa: Conceptualization, Investigation, Methodology, Data curation, Formal analysis, Visualization, Writing – original draft, Writing – review & editing. **Beatriz B. do Rêgo**: Data curation, Formal analysis, Visualization, Writing – original draft, Writing – review & editing. **Filipe A. Garrido**: Formal analysis, Visualization, Writing – original draft, Writing – review & editing. **Pedro D. Valente**: Formal analysis, Validation, Visualization, Writing – original draft, Writing – review & editing. **Nuno J. Nunes**: Visualization, Supervision, Writing – review & editing. **Ecivaldo S. Matos**: Investigation, Methodology, Formal analysis, Validation, Visualization, Supervision, Writing – review & editing.

Competing interests

The authors declare that they have no competing interests.

References

Abelein, U. and Paech, B. (2015). Understanding the Influence of User Participation and Involvement on System Success - a Systematic Mapping Study. *Empirical Software Engineering*, 20:28–81. DOI: 10.1007/s10664-013-9278-4.

Apshvalka, D., Donina, D., and Kirikova, M. (2009). Understanding the Problems of Requirements Elicitation Process: A Human Perspective. In *Information Systems*

Development, volume 4, pages 211–223. Springer US, Boston, MA. DOI: 10.1007/978-0-387-68772-8_17.

Arantes, F. L. (2013). Requirements Engineering of a Web Portal Using Organizational Semiotics Artifacts and Participatory Practices. *International Journal of Computer Science and Information Technology*, 5(2):131–146. DOI: 10.5121/ijcsit.2013.5212.

Bano, M., Zowghi, D., and da Rimini, F. (2016). User satisfaction and system success: an empirical exploration of user involvement in software development. *Empirical Software Engineering*, pages 1–34. DOI: 10.1007/s10664-016-9465-1.

Bano, M., Zowghi, D., and da Rimini, F. (2018). Power and Politics of User Involvement in Software Development. In *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018 - EASE'18*, pages 157–162, New York, New York, USA. ACM Press. DOI: 10.1145/3210459.3210477.

Baranauskas, M. C. C. (2013). O modelo semioparticipativo de design. In Baranauskas, M. C. C., Martins, M. C., and Valente, J. A., editors, *Codesign de Redes Digitais: tecnologia a serviço da inclusão social*, chapter 2. Penso, Porto Alegre.

Baranauskas, M. C. C., Schimiguel, J., Simoni, C. A. C., and Medeiros, C. B. (2005). Guiding the Process of Requirements Elicitation with a Semiotic-based Approach - A Case Study. In *Proceedings of the 11th International Conference on Human-Computer Interaction*, pages 100–111, Las Vegas.

Bourque, P. and Fairley, R. E. (2014). *Guide to the Software Engineering - Body of Knowledge.* IEEE Computer Society, Washington, D.C., USA.

Braun, V. and Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2):77–101. DOI: 10.1191/1478088706qp063oa.

da Costa, H. F. D. and Pimentel, J. H. C. (2017). O Papel da Comunicação na Elicitação de Requisitos - Entrevistas e Brainstorming. *Revista de Engenharia e Pesquisa Aplicada*, 2(4):97–109. DOI: 10.25286/repa.v2i4.845.

Das, V. V. (2007). Involvement of users in software requirement engineering. In *Proceedings - 10th International Conference on Information Technology, ICIT 2007*, pages 230–233. DOI: 10.1109/ICOIT.2007.4418302.

Davis, F. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3):319–340. DOI: 10.1016/S0305-0483(98)00028-0.

De Souza, C. S. (2005). *The Semiotic Engineering of Human-Computer Interaction*. MIT Press, Cambridge, Massachusetts.

De Souza, C. S. (2006). Da(s) subjetividade(s) na produção de tecnologia. In Nicolaci-da Costa, A. M., editor, *Cabeças Digitais: O cotidiano na era da informação*, pages 81–106. PUC-Rio/Loyola, Rio de Janeiro.

De Souza, C. S. and Leitão, C. F. (2009). Semiotic Engineering Methods for Scientific Research in HCI. *Synthesis Lectures on Human-Centered Informatics*, 2(1):1–122. DOI: 10.2200/S00173ED1V01Y200901HCI002.

Druin, A. (2002). The role of children in the design of new technology. *Behaviour & Information Technology*, 21(1):1–25. DOI: 10.1080/01449290110108659.

Faste, H., Rachmel, N., Essary, R., and Sheehan, E. (2013). Brainstorm, Chainstorm, Cheatstorm, Tweetstorm. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*, pages 1343–1352, New York, New York, USA. ACM Press. DOI: 10.1145/2470654.2466177.

Fuks, H., Raposo, A., Gerosa, M. A., Pimentel, M., Filippo, D., and Lucena, C. (2008). Inter- and intra-relationships between communication coordination and cooperation in the scope of the 3C Collaboration Model. In *2008 12th International Conference on Computer Supported Cooperative Work in Design*, volume 1, pages 148–153. DOI: 10.1109/CSCWD.2008.4536971.

Fuks, H., Raposo, A., Gerosa, M. A., Pimentel, M., and Lucena, C. J. P. (2007). The 3C Collaboration Model. In *Encyclopedia of E-Collaboration*, pages 637–644. IGI Global, Cham. DOI: 10.4018/978-1-59904-000-4.ch097.

Laitenberger, O. and Dreyer, H. M. (1998). Evaluating the usefulness and the ease of use of a Web-based inspection data collection tool. In *Proceedings Fifth International Software Metrics Symposium*, pages 122–132, Bethesda, USA. IEEE Computer Society. DOI: 10.1109/METRIC.1998.731237.

Lazar, J., Feng, J. H., and Hochheiser, H. (2017). *Research Methods in Human-Computer Interaction*. Morgan Kaufmann, Cambridge, USA, 2nd edition.

Leitão, C. F. (2017). Semiotic Engineering as a Reflexive, Transdisciplinary and Humanistic Theory in and Beyond HCI. In Barbosa, S. D. J. and Breitman, K., editors, *Conversations Around Semiotic Engineering*, chapter 10, pages 101–113. Springer International Publishing, Cham. DOI: 10.1007/978-3-319-56291-9_10.

Luck, R. (2003). Dialogue in participatory design. *Design Studies*, 24(6):523–535. DOI: 10.1016/S0142-694X(03)00040-1.

Luck, R. (2018). What is it that makes participation in design participatory design? *Design Studies*, 59:1–8. DOI: 10.1016/j.destud.2018.10.002.

Markopoulos, P., Read, J. C., MacFarlane, S., and Höysniemi, J. (2008). *Evaluating Children's Interactive Products: Principles and Practices for Interaction Designers*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. DOI: 10.1007/s13398-014-0173-7.2.

Marques, A. B., Conte, T., and Barbosa, S. D. J. (2016). Representing the interaction and navigation of interactive systems through a usability-oriented model. In *Proceedings of the 15th Brazilian Symposium on Human Factors in Computer Systems - IHC '16*, pages 1–10, Nova York, EUA. ACM Press. DOI: 10.1145/3033701.3033716.

McAlpine, H., Cash, P., and Hicks, B. (2017). The role of logbooks as mediators of engineering design work. *Design Studies*, 48:1–29. DOI: 10.1016/j.destud.2016.10.003.

Mendes, M. S. and Furtado, E. (2014). The DIRCE approach for Requirements Elicitation and Design of Interaction in innovative systems. In *2014 XL Latin American Computing Conference (CLEI)*, pages 1–10, Montevideo, Uruguay. IEEE. DOI: 10.1109/CLEI.2014.6965150.

Muller, M. J. and Druin, A. (2003). Participatory design: the third space in HCI. In *The human-computer interaction handbook*, pages 1051–1068. L. Erlbaum Associates Inc., Hillsdale, NJ, USA.

Muller, M. J., Haslwanter, J. H., and Dayton, T. (1997). Participatory Practices in the Software Lifecycle. In Helander, M. G., Landauer, T. K., and Prabhu, P. V., editors, *Handbook of Human-Computer Interaction*, chapter 11, pages 256–269. Elsevier, Amsterdam, 2 edition.

Norman, D. (2013). *The Design of Everyday Things*. Basic Books, New York, New York, USA.

Nuseibeh, B. and Easterbrook, S. (2000). Requirements engineering. In *Proceedings of the conference on The future of Software engineering - ICSE '00*, volume 1, pages 35–46, New York, New York, USA. ACM Press. DOI: 10.1145/336512.336523.

Ogunyemi, A. and Lamas, D. (2014). Interplay between human-computer interaction and software engineering. In *2014 9th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–10, Barcelona, Spain. IEEE. DOI: 10.1109/CISTI.2014.6877024.

Oran, A. C., Nascimento, E., Santos, G., and Conte, T. (2017). Analysing Requirements Communication Using Use Case Specification and User stories. In *Proceedings of the 31st Brazilian Symposium on Software Engineering - SBES'17*, pages 214–223, New York, New York, USA. ACM Press. DOI: 10.1145/3131151.3131166.

Pacheco, C., García, I., and Reyes, M. (2018). Requirements elicitation techniques: a systematic literature review based on the maturity of the techniques. *IET Software*, 12(4):365–378. DOI: 10.1049/iet-sen.2017.0144.

Pereira, R. and Baranauskas, M. C. C. (2015). A value-oriented and culturally informed approach to the design of interactive systems. *International Journal of Human-Computer Studies*, 80:66–82. DOI: 10.1016/j.ijhcs.2015.04.001.

Pita, G. L., Zabot, D., Rosa, J. C. S., and Matos, E. (2017). Adapting the SPIDe to Include Visually Impaired Users in Interaction Design. In *Proceedings of the XVI Brazilian Symposium on Human Factors in Computing Systems - IHC 2017*, pages 1–4, New York, New York, USA. ACM Press. DOI: 10.1145/3160504.3160579.

Rocha, H. V. and Baranauskas, M. C. C. (2003). *Design e avaliação de interfaces humano-computador*. Universidade Estadual de Campinas, Campinas.

Rosa, J. C. S. (2017). *Design de Interacão Multicultural: Um Framework Semioparticipativo para o (re)design da Interacão de Softwares Educacionais*. Mestrado, Universidade Federal da Bahia.

Rosa, J. C. S. and Matos, E. (2016). Semio-Participatory Framework for Interaction Design of Educational Software. In *Proceedings of the 15th Brazilian Symposium on Human Factors in Computer Systems - IHC '16*, pages 1–10, New York, New York, USA. ACM Press. DOI: 10.1145/3033701.3033734.

Rosa, J. C. S., Matos, E., Santos, F., and Silva, G. (2018). Experimentando o SPIDe aplicado à Elicitação de Requisitos. In *Anais do WER18 - Workshop em Engenharia de Requisitos*, pages 1–14, Rio de Janeiro. PUC-Rio. DOI: 10.17771/PUCRio.wer.inf2018-34.

Rosa, J. C. S., Silva, F. S., Silva, G. J. F., and Matos, E. (2017). Applying SPIDe's Techniques in Requirements Engineering: a sistematic review. *Computing and System Journal*, 7(2):290–303.

Rosa, V. and Matos, E. (2019). Codesign de Interação com/por/para o Sujeito Idoso: Identificação e Validação de Requisitos para Engajamento na Produção de Tecnologias Digitais. In *Anais Estendidos do Simpósio Brasileiro de Fatores Humanos em Sistemas Computacionais (IHC)*, pages 154–157, Porto Alegre, Brasil. Sociedade Brasileira de Computação - SBC. DOI: 10.5753/ihc.2019.8420.

Rosa, V. A. and Matos, E. (2020). Strategies to improve engagement of long-lived elderly women (80+) in the interaction co-design process. In *Proceedings of the 19th Brazilian Symposium on Human Factors in Computing Systems*, pages 1–10, New York, NY, USA. ACM. DOI: 10.1145/3424953.3426500.

Seffah, A., Desmarais, M. C., and Metzker, E. (2005). HCI, Usability and Software Engineering Integration: Present and Future. In Seffah, A., Gulliksen, J., and Desmarais, M., editors, *Human-Centered Software Engineering - Integrating Usability in the Software Development Lifecycle*, chapter 3, pages 37–57. Springer Netherlands, Dordrecht. DOI: 10.1007/1-4020-4113-6_3.

Shull, F., Carver, J., and Travassos, G. H. (2001). An empirical methodology for introducing software processes. In *Proceedings of the 8th European software engineering conference held jointly with 9th ACM SIGSOFT international symposium on Foundations of software engineering - ESEC/FSE-9*, page 288, New York, New York, USA. ACM Press. DOI: 10.1145/503209.503248.

Vilarinho, T., Farshchian, B., Floch, J., and Hansen, O. G. (2019). Participatory Ideation for Gamification: Bringing the User at the Heart of the Gamification Design Process. In Bogdan, C., K., K., M., L., P., P., and M., W., editors, *Human-Centered Software Engineering. HCSE 2018. Lecture Notes in Computer Science*, volume 11262, pages 51–61. Springer International Publishing, Cham. DOI: 10.1007/978-3-030-05909-5_4.

Wagner, S., Greer, D., Lassenius, C., Männistö, T., Nayebi, M., Oivo, M., Penzenstadler, B., Prikladnicki, R., Ruhe, G., Schekelmann, A., Sen, S., Fernández, D. M., Spínola, R., Tuzcu, A., Vara, J. L. D. L., Winkler, D., Felderer, M., Vetrò, A., Kalinowski, M., Wieringa, R., Pfahl, D., Conte, T., and Christiansson, M.-T. (2019). Status Quo in Requirements Engineering. *ACM Transactions on Software Engineering and Methodology*, 28(2):1–48. DOI: 10.1145/3306607.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., and Wesslén, A. (2012). *Experimentation in software engineering*. Springer Berlin Heidelberg, Berlin, Heidelberg. DOI: 10.1007/978-3-642-29044-2.

Zabot, D., Andrade, S., and Matos, E. (2019). Game Design participativo com crianças surdas e com deficiência auditiva: uma experiência no ensino fundamental. In *Anais do Workshop sobre Interação e Pesquisa de Usuários no Desenvolvimento de Jogos (WIPlay)*, pages 49–58, Porto Alegre, Brazil. SBC. DOI: 10.5753/wiplay.2019.7839.