UX Requirements Matters: Guidelines to Support Software Teams on the Writing of Acceptance Criteria

Suéllen Martinelli
Federal University of São Carlos (UFSCar)
São Carlos, Brazil
suellen.martinelli@estudante.ufscar.br

Nicolas Nascimento
Pontifical Catholic University of Rio Grande do Sul (PUCRS)
Porto Alegre, Brazil
nicolas.nascimento@pucrs.br

Jonathan Souza
Federal University of São Carlos (UFSCar)
São Carlos, Brazil
jonathanhjs@ufscar.br

Afonso Sales
Pontifical Catholic University of Rio Grande do Sul (PUCRS)
Porto Alegre, Brazil
afonso.sales@pucrs.br

Luciana Zaina
Federal University of São Carlos (UFSCar)
Sorocaba, Brazil
lzaina@ufscar.br

ABSTRACT

User Stories (USs) are artifacts that define requirements in agile software development and that are usually complemented by Acceptance Criteria (AC) which provide details about what should be implemented. These details often attempt to integrate User Experience (UX) requirements to their descriptions. This paper aims to introduce a set of guidelines that supports software teams to write UX requirements during the elaboration of AC. The guidelines are separated into Interaction Design (ID) and Visual Elements (VE). We conducted a case study with 10 agile teams that develop mobile applications that used the guidelines during a sprint. After that, we carried out a qualitative analysis of the UX requirements described in the USs/AC to evaluate the usage of the guidelines. We analyzed the guidelines' acceptance regarding usefulness and ease of use based on feedback from the teams. We analyzed 242 AC and identified that 99 AC had UX requirements related to ID and 87 to VE. We identified that only 7 developers, spread over four teams, reported negative feedback about the guidelines. We conclude that teams were helped by the guidelines that respectively address details about the user's interaction with the product, as well as the definition of the most adequate visual elements to enable this interaction.

CCS CONCEPTS

• **Software and its engineering** → **Designing software**; Agile software development.

KEYWORDS

Acceptance Criteria, User Stories, User Experience, Mobile Application Development

SBES 2022, October 5–7, 2022, Virtual Event, Brazil

ACM ISBN 978-1-4503-9735-3/22/10…$15.00
https://doi.org/10.1145/3555228.3555230

ACM Reference Format:
Suéllen Martinelli, Nicolas Nascimento, Jonathan Souza, Afonso Sales, and Luciana Zaina. 2022. UX Requirements Matters: Guidelines to Support Software Teams on the Writing of Acceptance Criteria. In *XXXVI Brazilian Symposium on Software Engineering (SBES 2022), October 5–7, 2022, Virtual Event, Brazil.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3555228.3555230

1 INTRODUCTION

User Stories (USs) are valuable artifacts for agile teams in the management of software development activities which includes requirements specification [17, 27]. USs contain brief descriptions of features from the user perspective and are created with contributions collected from users and/or customers. The structure (grammar) of Cohn [3] for writing USs is the most used by developers [18, 27]. It follows a simple template, *i.e*, "as a *type of user*, I want *some goal* so that *some reason*", which points out the user, its needs and the reason of having such need [3]. Three aspects compose USs: (i) a description of the story, (ii) conversations about the story to drill down into the details, and (iii) tests that convey and document details that can be used to determine when a story is completed [3].

Furthermore, Acceptance Criteria (AC) are present in the structure of the USs to complement them by describing critical points in the development of the US [3]. In addition to providing the details that impact implementation, AC also support developers during the testing phase [3]. Unlike the USs, for the AC there is no best consensual structure. North [22] proposes a structure for the elaboration of AC, which is composed of three parts: *given*; *when*; and *then*. In the "given" statement, initial context is presented, in the "when" statement, an event or action is described, and in the "then" statement, expected results are defined.

Moreover, the development of modern software products requires companies to consider User Experience (UX). UX is related to the experience that a product provides for users when using it [7]. Therefore, designing for UX is about understanding expectations, and also considering the actions of users [7]. A framework is proposed by Garrett [7] to support the development of products that provide a better user experience. This structure is divided by the author into five layers (*i.e.*, strategy, scope, structure, skeleton, and surface) so that in each layer the UX elements are presented. As each layer is contemplated the product becomes more concrete.

Industry professionals report that the main interest in UX is to design better products [15]. Therefore, the need to constantly improve the UX of products is recognized as a method to meet the expectations of users [12]. However, studies show that users' needs are often treated informally or neglected when it comes to requirements related to UX [12, 15], even though professionals are aware of the importance of dealing UX in the early stages of software product development [15]. Despite the recognized relevance, many software teams struggle in specifying UX-related requirements [12]. These struggles can be related to the fact that UX information is scattered in different artifacts (e.g., project documents, UX-specific artifacts) [33] or even that many companies do not have UX experts working in them [13]. Therefore, in some cases, the development team is responsible for specifying the UX requirements. In addition, as US/AC is a valuable artifact to the teams, they can use them to manage UX requirements [27, 33].

This paper aims to introduce *Acceptance Criteria of User Stories* (ACUX) a set of guidelines to support software teams to the writing of UX requirements during the elaboration of ACs. ACUX is classified into two aspects of UX design, Interaction Design (ID) and Visual Elements (VE). Thus, our study goal is to understand how ACUX is used in the elaboration of AC considering UX requirements. Hence, we have sought to evaluate ACUX through a case study with 10 agile teams in a Research, Development and Innovation environment that develops mobile applications. Our two data sources were the AC elaborated by the 10 agile teams in which ACUX was applied and the results of a *Technology Acceptance Model* questionnaire which was applied to extract insights from participants regarding the guide itself. Our preliminary results indicate that teams were helped by guidelines that address details about the user's interaction with the product, besides the definition of the most adequate visual elements to enable this interaction. Moreover, ACUX had a high acceptance, but some teams reported issues regarding to ease of use.

2 ACUX

To elaborate ACUX, we followed the proposal of Rusu et al. [26] which is composed of six stages (see next sections). The literature review that supports ACUX elaboration can be seen in [30] (see Chapter 3). The current ACUX version focuses on the description of UX requirements considering concrete UX elements related to interaction design and visual interface.

2.1 ACUX Elaboration

In the **exploratory stage**, we followed the steps of a systematic mapping study for the execution of search and screening of relevant papers in the literature [24]. Our main goal was to identify what the literature has reported about the writing of UX requirements in USs and AC. Firstly, we conducted an ad-hoc exploration by looking for seminal papers which focus on our general research topic. As results showed us there were a few papers on that topic, we decided to use a more open string: *"UX" OR "user center-design" OR "user experience design") AND ("user story" OR "user stories") AND ("acceptance criteria" OR "acceptance test"))"*. Our searchings were conducted in four well-known bases: *ACM Digital Libray*, *SpringerLink*, *IEEE Explorer* and *Science Direct* which resulted in a total of 257 papers. After applying our inclusion (the paper had to discuss approaches to the US and AC elaboration) and exclusion criteria (not written in English; short papers, *i.e.*, less than 4 pages; not published in conferences or journals), we ended up with 9 papers. As we had a small number of papers, we performed a snowballing forward process to complement our literature review [32]. We selected Lucassen et al. [17]'s work as our seed as it is the work with the largest number of citations in Google Scholar which discussed the elaboration of US. We carried out three levels of snowballing forward from which we got 247 papers. After that, we applied the inclusion and exclusion criteria (the same described above) and select 17 papers. At the end, we had 26 papers which were carefully analyzed. We verified that most studies do not present many details about the elaboration of AC. From the 26 papers, 8 consider UX in their proposal, and from those only 3 discuss issues with regards to UX in AC elaboration; however, they do not propose approaches to support the writing of UX requirements from AC [2, 16, 19]. Our review also showed that Cohn [3] and North [22] are the most used grammar to write USs and AC, respectively.

In the **descriptive stage**, the authors point out that the main concepts regarding the context of the guidelines should be formalized. In our case, we worked on conceptualizing the most important characteristics in the context of USs/AC elaboration. As our literature review did not present relevant findings on AC elaboration, we decided to conduct an exploratory study in which four researchers from the UX and Software Engineering (SE) area carried out a qualitative analysis in 261 AC of 79 USs that 30 developers created. During the USs/AC writing, the developers were supported by UX artifacts that informed the main user aspects. To conducted the analysis, the researchers considered Garrett's Framework [7] (see Figure 1). In this analysis, we sought to relate the description of each USs/AC with the UX elements presented in Garrett's framework (see Figure 1). Garrett's framework was chosen because the proposal elements are closely related to the practice with real examples of designing products that provide a better experience for users. The results showed that even the developers were motivated to include UX requirements in the USs and AC (by using the UX artifacts), most of the AC did not report any UX requirement (63%). Among those that reported UX, most presented elements from the structure layer (see Figure 1) in a little abstract description.

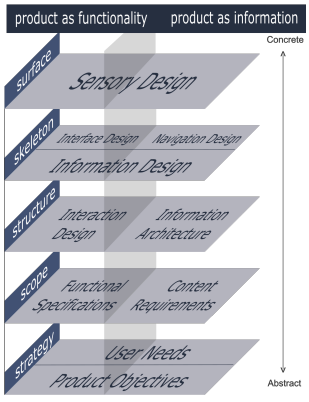

Figure 1: Garrett's framework [7]

Rusu et al. [26] state that guidelines' characteristics have to be defined in the **correlational stage**. Due to Garrett [7]'s framework UX-related practical elements and examples (see Figure 1) we used it as an underpinning to ACUX elaboration. To build ACUX, the three upper layers of the framework of (*structure, skeleton and surface*) were considered, as they work the steps that directly interfere in the user's interaction with the product. Taking into account these three layers, the UX elements of the "structure" and "skeleton" layers were managed in a group called "Interaction Design" while the elements of the "surface" layer composed the "Visual Elements" group. We considered that these divisions could improve the communication of the guidelines meanings to the software teams.

In the **explanatory stage**, we consolidated the findings of the previous results and defined the guidelines. We constructed ACUX considering the results of the *Systematic Literature Mapping* (SLM), the exploratory study and the UX elements of Garrett's framework. The SLM provided positive feedback for using the Cohn [3] and North [22] grammar. The exploratory study highlighted the main usual mistakes in the writing of AC and the main artifacts that support the insertion of UX requirements. Garrett's framework [7] supported us in describing practical ways to describing UX requirements. It is worth noting that ACUX is not platform-specific. It provides a general description that can be applied on different projects and platforms.

In the **validation stage**, we evaluated ACUX with 4 experts who had practical experience in the software industry. They provided feedback about the guidelines improvement and ways of how to use ACUX from a developer's perspective. First, 2 experts evaluated ACUX and made suggestions on ACUX visual presentation format and the need of having practical examples for each guideline. For the visual format, the experts suggested the use of an interactive way to present ACUX. We thus produced a second version which two other experts evaluated and they suggested only refinements in the examples. In the **refinement stage**, we took into account the suggestions to produce the final version of ACUX.

2.2 ACUX Guidelines

ACUX contains three components. First, ACUX presents the guidelines split into two groups, the Interaction Design (ID) and the Visual Elements (VE), with 6 and 9 guidelines respectively. For each guideline, ACUX shows a practical example of how to apply that guideline. ACUX also provides 9 tips regarding "Usual Mistakes" that alerts to what should be avoided during the UX requirements description. Finally, ACUX points out some artifacts that can help developers to get insights regarding UX requirements and consequently add them in AC. Table 1 presents the two groups with their respective guidelines, and Table 2 lists the usual mistakes.

We built an interactive version of ACUX[1] using Figma, a platform where users can see the examples by clicking on the respective guideline. In ACUX, all the examples are presented using the grammar of Cohn [3] for writing the USs and North [22] for writing the AC. See an example of the guideline ID-01 application below.

> **US**: *As a <online customer> I want <search for products> so that <I find the ones I want to buy>.*
>
> **AC**: *Given <that there are products related to the search term> when <**I click the search button**> then <the products found are returned>.*

[1] ACUX guidelines available in Portuguese and online.

Table 1: Guidelines of the ACUX

INTERACTION DESIGN	
ID-01	Specify how the user interacts with system functionality.
ID-02	Specify how to get to a particular screen, or the paths the user can take when on it.
ID-03	Specify organization details and presentation of content, such as grouping and sorting.
ID-04	Specify details about how the information is arranged, and how one information is linked to another.
ID-05	Specify how the interface elements available on the screen allow the user to navigate.
ID-06	Specify the sequence in which information should be presented to facilitate interaction.
VISUAL ELEMENTS	
VE-01	Specify the most suitable visual elements so that the user can carry out his tasks.
VE-02	Specify the organization of elements on the screen so that they are readily understood and easily used by users.
VE-03	Specify details about how to present information so that the user can understand it more easily, such as graphics and images.
VE-04	Specify elements that make it possible to go from one point to another in the system.
VE-05	Specify style details.
VE-06	Consider color palette, typography, style/visual identity guide.
VE-07	Specify details about fonts, colors and shapes that relate to the style adopted in the product/project.
VE-08	Specify contrast details, highlighting what users really need to see.
VE-09	Specify details to maintain design uniformity (keep size of elements uniform).

Table 2: Usual Mistakes in AC

Description of Usual Mistakes
Do not specify details about user interaction;
Do not specify details about the organization of the information;
Do not point out user actions with the product;
Do not provide details about the organization of the content;
Do not specify the visual elements that allow for interaction or navigation;
Do not specify the visual elements of the interface;
Do not specify style details;
Specify details of fonts, colors and shapes that are not related to the style adopted in the project;
Specify details based on personal insights.

3 CASE STUDY

We have conducted an exploratory case study with the participation of 10 agile teams that develop mobile applications to evaluate ACUX. Our study was approved by the human research ethics committee at UFSCar under process number 37663220.5.0000.5504. However, as the study was carried out in a Research and Development and Innovation (R&D&I), we can not make the raw data available. We followed the case study guidelines proposed by Runeson and Höst [25] to the study design. We aimed to investigate how the agile teams use the guidelines to the elaboration of AC for USs which includes UX requirements. Our unit of analysis was the software teams. We defined the following Research Questions (RQ) to guide our study: **RQ1**: *"How are the guidelines applied in the elaboration of*

Acceptance Criteria by agile teams?" and **RQ2**: *"What is the teams' feedback in relation to acceptance of the guidelines about the usefulness and ease-of-use?"*

3.1 Case Study Context

The study was conducted in a R&D&I that develops Apple mobile apps. In the center, trainees are introduced to the development ecosystem of Apple platforms and acquire experience by working together in agile teams. These trainees are supported by tech, design and business mentors who have solid experience in the software industry. We will refer to these mentors as leadership team hereinafter. The teams work on the development of apps that supports users to overcome real-world problems. The R&D&I environment motivates the team members to cooperate in all software development stages from requirements elicitation to deployment steps.

Considering our study, the teams employed an adaption of Scrum by applying the concept of a Sprint 0 [29]. In the environment, all sprints lasted for 1 week. Specifically in Sprint 0, the teams search for evidence on innovative features to be implemented on the apps by conducting, for instance, user interviews, market research, and literature reviews. These investigations also provided insights to the team regarding user characteristics and needs. Considering these findings, the teams are able to consolidate an initial product backlog. After that, the teams proceeded to the sprint planning and defined a set of USs which were prioritized and then moved to the sprint backlog. Taking into account the sprint backlog, the teams discussed and defined a set of AC for each US of the sprint backlog. In the environment, the sprint planning lasts for 2 hours on average, in line with the Scrum guide [28]. During the writing of the AC, the teams used our guidelines. The R&D&I teams are used to adopting US in their projects, but they have little experience with US specifications. The teams did not take any approach for specifying UX requirements in US before using our guidelines. As the AC of all USs were finished, the teams started the Scrum sprint. The teams performed the usual Scrum ceremonies during the sprint (*i.e.*, daily meetings, sprint review, sprint retrospective). Sprints were conducted by every team independently, in parallel with others and in sync with the defined 1 week sprints.

In our study, each team (N=10) was composed of 5 members plus one leader. From these 5 members, at least one was a dedicated designer (see Table 3). The Scrum Master (SM) role was performed by a team leader (a member of the leadership team) and there were no Product Owners (POs). Each team worked on developing mobile apps which addressed different scopes (see Table 3). The trainees dedicated 20 hours/week to software development activities.

Table 3: Team characteristics

Team	Project Scope	Size	Developers	Designers	Lead
1	Businesses	5	4	1	A
2	Mental Health	5	4	1	A
3	Emotional Health	5	3	2	B
4	Donation	5	4	1	B
5	Lifestyle	5	4	1	C
6	Education	5	4	1	C
7	Supermarkets	5	4	1	D
8	Gaming	5	4	1	D
9	Beer	5	4	1	E
10	Productivity	5	4	1	E

Considering the trainees who participated in the case study, the agile teams were composed of 17 participants that self-identify as female and 33 that self-identify as male. Further, 11 trainees were up to 20 years old, 29 are between 21 and 25 and 10 were over 25, with the oldest trainee being 37. Most trainees were pursuing an undergraduate' degree with a single exception pursuing a master's degree. In general, all 50 trainees had at least 6 months of experience working with mobile application development, with 28 of those having had some prior experience in some level with mobile application development. In general, the teams also had around 4 months of experience with UX. All the projects considered in the case study involved graphical interface design.

3.2 Data Collection

We collected 2 data sources: the AC elaborated by the 10 agile teams and the participants answers about the usage of the guidelines. Initially, we conducted two sessions to present the guidelines, first to the leadership team and then to the development teams. As the team members had little experience with UX, we considered that there would be a time for the teams to learn how to use the guidelines. In order to mitigate the impact on the participants' learning curve, we conducted the training in online meetings that lasted 2 hours each.

Besides, we consider that the guidelines were presented in a visual and easy-to-understand approach, which contributed to the good use of ACUX. The presentations were conducted by a researcher with 3 years of experience in UX Research and SE. In the two sessions, the researcher introduced ACUX and also clarified any doubts regarding the use of the guidelines. As the leaders had daily contact with the trainees, the leaders could help the teams clarify doubts about the use of the guidelines during the writing of the acceptance criteria. The leaders took part in the second session to put the researcher in contact with the trainees. Before the presentation of ACUX, teams had already started Sprint 0.

After that, each team began working on their projects (see Table 3). The planning of Sprint 1 was carried out by each team which focused on the writing of the AC by using the guidelines. All the AC and USs were written in a collaborative way where all the teams' members discussed the AC and then described them in the *Airtable*[2] platform. The teams' leaders and the researchers were made available to the teams in case of doubts when applying the guidelines. During the sprint (a week), the teams could refine the AC of their respective projects. At the end of the sprint, we collected a total of 286 AC associated with 155 USs from the 10 teams. The teams shared with us all the USs and AC in a spreadsheet.

After Sprint 1, we invited the trainees to answer an online questionnaire voluntarily to collect feedback regarding ACUX. The questionnaire was based on *Technology Acceptance Model* version 1 (TAM1), which focuses on analyzing the usefulness and the ease of use of a technology or artifact under user perspective [4]. We adopted TAM1 keeping the original categories (i.e., perceived usefulness and ease of use) and their respective questions [4]. We only changed the answers from 7 to 4-point scale without the neutral point. According to [6] and [10], a scale without a neutral point favors a more accurate response and prevents the choice of a neutral point to avoid a conflict of opinion with the researcher. We

[2]Airtable plataform online: https://airtable.com/.

created the questionnaire in Airtable with 10 mandatory questions. We added one optional open question from which the participants could describe their individual opinion and make suggestions to the guidelines. Our TAM instrument also was reviewed by a Ph.D. student and a researcher with experience in using TAM in SE.

3.3 Data Analysis

We considered two sources in the data analysis. To answer the RQ1 (*"How are the guidelines applied in the elaboration of Acceptance Criteria by agile teams?"*), we carried out a qualitative analysis in 286 AC of 155 US (see Section 3.3.1). To explore the feedback of the teams and answer the RQ2 (*"What is the teams' feedback in relation to acceptance of the guidelines about the usefulness and ease-of-use?"*), we analyzed the trainees' answers to the TAM questionnaire and the open question about the guidelines (see Section 3.3.2).

Three researchers participated in the data analysis. The AC analysis was conducted by two researchers, a Ph.D. candidate and a Master, both in the Computer Science area with extensive knowledge, *i.e.*, 4+ years, in SE and UX in industrial settings. The feedback analysis was conducted by a Ph.D. candidate with experience of 3+ years in SE and UX. The two steps of analysis were reviewed by two senior researchers, 15+ years of experience, in SE and UX in industrial settings.

3.3.1 Acceptance Criteria Analysis. We transcribed all the data collected, *i.e.*, US and AC, in online spreadsheets using the platform *Airtable* to enable collaboration between researchers. We conducted an analysis in four steps: selection of AC that fulfills quality criteria, warm-up analysis, selection of the AC that specified UX requirements, and analysis of which UX requirements were applied. Before exploring the use of the guidelines, we applied exclusion criteria (EC) into 155 USs and 286 AC to improve the quality of our sample. The ECs and the respective number of US and AC excluded from the sample are described as follow. EC1 - USs that did not have AC,*i.e.*, epics, (removed 8 USs and 0 AC); EC-02 - AC description did not have a clear relationship with the US, *i.e.*, the US implementation was not necessarily affected by the AC (removed 2 USs and 2 AC); EC-03 - USs that did not follow the user's product perspective,*i.e.*, the USs described a developer's activity (removed 28 USs and 33 AC); EC-04 - USs with a user perspective but not related to product features,*e.g.*, US described the app's download from the play store (removed 2 USs and 3 AC); and EC-05 - duplicated USs and AC (removed 2 USs and 5 AC). At the end, 42 USs and 44 AC were eliminated from our sample.

After applying the exclusion criteria, we got a sample of 242 AC associated with 113 USs to be analyzed. We decided to conduct a warm-up analysis to align the strategy for analysis and the understanding of two researchers about the guidelines. To the warm-up, we selected 10 USs/AC from the sample randomly and each researcher individually analyzed them. The analysis goal was to identify whether there were UX requirements (*i.e.*, IDs or VEs) in the AC descriptions taking into account the guidelines. The researchers read each US and the respective AC and answered the question: "*Does AC have ID or VE requirements?*". The answers were counted to *yes* or *no* classification. Then a meeting was conducted to see the warm-up results and align their points of view.

Subsequently, the two researchers thus conducted the analysis of 242 AC /113 USs following the steps described above. To check the reliability of the results from the sample, we applied the Kappa coefficient [5]. Kappa represents the agreement relationship between raters in the classification of items, so that the results can vary between -1 and 1, being classified as follows: no agreement (-1 - 0), slight agreement (0 - 0.20) , fair agreement (0.21 - 0.40), moderate agreement (0.41 - 0.60), substantial agreement (0.61 - 0.80) and almost perfect agreement (0.81 - 1) [5]. To calculate the Kappa index, we selected a sample of 73 AC from 242, *i.e.*, 30% of the total AC analyzed, randomly. These 73 AC were analyzed by two researchers (R1 and R2). R1 and R2 agree that 36 AC had some ID or VE elements (as *yes* classification), whereas both researchers disagree on 27 AC (as *no* classification). R1 disagrees that 5 AC does not present ID or VE elements, and R2 disagrees about others 5 AC. Therefore, we have achieved a score of 0.72 in the Kappa index and a relative acceptance rate of 86%. The value obtained places the AC analysis process (on whether or not UX requirements were present in the AC descriptions) at a substantial level of agreement [5] which is considered a good result [14].

Considering the AC that contained UX requirements, *i.e.*, 85 USs and 123 AC, the researchers carried out a closed coding process separately [8]. The researchers conducted the closed coding individually by using the ID and VE guidelines as the codebook. The researcher assigned the codes during the analysis, *i.e.*, the ID and VE guidelines, to the AC (see Table 4). An individual acceptance criterion could have more than one code assigned to it. We also picked up whether there were mistakes in the AC specification, the ones we called "Usual Mistakes" in ACUX (see Table 4). After the closed coding process, the researchers conducted a meeting of agreement in which they discussed the AC one by one to check agreements and disagreements. In case of disagreement, a new discussion round was conducted to reach consensus on the best coding.

3.3.2 Feedback Analysis. We analyzed the teams' guidelines acceptance over the dimensions of usefulness and ease of use. In the questionnaire, participants selected their answers from a 4-point Likert scale. To calculate the answers for each question, we considered weights for each answer as follow: Totally Agree (4), Partially Agree (3), Partially Disagree (2), and Strongly Disagree (1). In our analysis, the trainees' responses in the TAM questionnaire were grouped by teams as we wished to explore the teams perspective about the guidelines. We calculated averages for each TAM question by the team since we had teams with a single member answering (Team 5 and 8) and the others with two or more members.

We performed a qualitative analysis of the responses to the open question by following a closed coding procedure [8]. We assigned the *Positive Aspects* code to characterize the trainees' positive feedback about the use of the guidelines, while the *Negative Aspects* code to the negative feedback. The *Opinions and Suggestions* code was assigned to represent feedback that did not present a value judgment about the guidelines. Finally, we performed a comparison between the quotes gathered in each code with the TAM answers.

4 RESULTS

Our findings demonstrate how ACUX was used in practice by the teams, highlighting the main UX requirements included in the AC

Table 4: Analysis Example

User Story	Acceptance Criteria	Does AC have ID or VE element?	ID	VE	Usual Mistakes
"As a user, I would like to visualize my data in a graphical way to understand it more easily."	*"GIVEN the registered moods, WHEN I click on the calendar page, THEN the period graph and the distribution of moods in the period are displayed."*	Yes	*"ID-01: "click""*	*"VE-03: "graph""*	-
	"GIVEN the graphics page WHEN they are displayed, THEN they appear colorful and visual graphics."	Yes	-	*"VE-03: "graphic"", "VE-07: "colorful and visual""*	*"Do not specify style details: "colorful and visual""*

(Section 4.1) and we also present the acceptance of ACUX by the teams (Section 4.2).

4.1 Guidelines Usage (RQ1)

The first point we noticed from the result was that most of the AC analyzed reported some kind of UX requirement. Among the 242 AC analyzed, UX requirements were identified in 123 AC (*i.e.*, approximately 51%). Our results pointed out that many of the AC in which UX requirements were not found deal with aspects more related to Functional Specifications and Content Requirements [7]. As an example: *"It should be possible to access quick game settings"*. In this AC, we have not identified any UX requirements based on the guidelines.

We have also identified that the UX requirements present in both groups of guidelines were reported. Specifically, 99 AC had UX requirements linked to the ID group, and 87 to the VE group. Furthermore, we observed that 63 AC reported UX requirements simultaneously linked to both groups (ID and VE). Figure 2 shows the relationship between the teams, the guidelines used and the UX dimensions present in the AC. Note that all teams reported some UX requirement at some point. In addition, all teams used the guidelines of both groups (ID and VE). Even developing applications that address different contexts, we identified that the teams addressed both UX requirements related to the ID and VE dimensions. We see that most teams have added more requirements related to the ID group. Only Teams 2, 3 and 9 addressed more requirements from VE. However, we consider the usage of the guidelines among the UX dimensions by teams was balanced. In Teams 7, 8 and 10, the ID group had greater prominence (>= twice as many occurrences), and in Team 9 the VE group stood out (>= twice as many occurrences).

We identified that the guideline "ID-01" and "VE-01" are the most recurrent (present in 73 and 55 AC, respectively). The "ID-03" and "VE-03" guidelines, respectively with 43 and 29 occurrences, complete the four main guidelines used by the participants. We have also identified that, in addition to being the most recurrent, "ID-01" was the only guideline in which it reported UX requirements across all teams. With this we conclude that regardless of the context of the application under development, detailing *"how the user interacts (ID-01)"* is an important factor for the teams. Observing these results, we realized that there is a relationship between guidelines "ID-01" and "VE-01", which deal respectively with the form of user interaction, and most adequate visual elements for users to carry out their tasks. Focusing on the 63 AC that considered both groups of guidelines synchronously, 34 (equivalent to 53.97%) reported "ID-01" and "VE-01" simultaneously, for example: *"[...] when I **click** (ID-01) the **button** (VE-01) [...]"*.

The only guideline in which its content was not found in any AC was "VE-08", which addresses issues related to contrast, to highlight what needs to be noticed by users. "VE-09" and "VE-07" were rarely used, found respectively in only 01 (*"must be responsive and maintain the design on all devices"*) and 02 (*"colorful and visual"*; *"one color in each tag"*) AC. Thus, we observe that the least found guidelines are all related to the VE. This result indicates that the UX requirements related to the more concrete steps of product development (*i.e.*, visual elements) were less addressed by the teams. While the more abstract issues (*e.g.*, before what is visual) were worked on more by the study teams.

In summary, the guidelines mainly helped in the inclusion of UX requirements related to details about how the user's interaction with the product need to occur. Then, the teams detailed the most suitable visual elements to provide the interaction. We noticed that teams also frequently did not provide details about how content should be organized (ID-03), and how information should be presented (VE-03).

We observed in our analysis the presence of usual mistakes in the AC descriptions, that is, we judged whether the guidelines were used correctly. The result of this check was 23 usual mistakes (see Table 5). The most recurrent usual mistake was related to the lack of details about the user interaction. This fact showed us that participants often failed to specify how the end user's interaction with the system will take place. That is, despite specifying in the AC that at a given moment there is user interaction with the functionality or product, how this interaction should occur was not clear (click? gesture? touch?). For example *"[...] I interact [...]"*; and *"[...] make notes [...]"*, in these two examples extracted from two AC present in our analysis, it was not specified how the interaction should occur. The second most common mistake found was related to the presence of information based on personal perceptions (for example: *"[...] clear and concise [...]"* and *"[...] easy access [...]"*). Information that the guidelines points out as a detrimental factor to the inclusion of UX in products.

4.2 Teams' Feedback on the Guidelines (RQ2)

Our results are consistent with a sample of 27 out of the 50 trainees (54%) who participated in the case study and answered the online TAM questionnaire voluntarily (see questions in Table 6). Figure 3 shows the distribution of teams' answers about their perception of *usefulness* and *ease-of-use* on ACUX. Team 5 and 8 provided

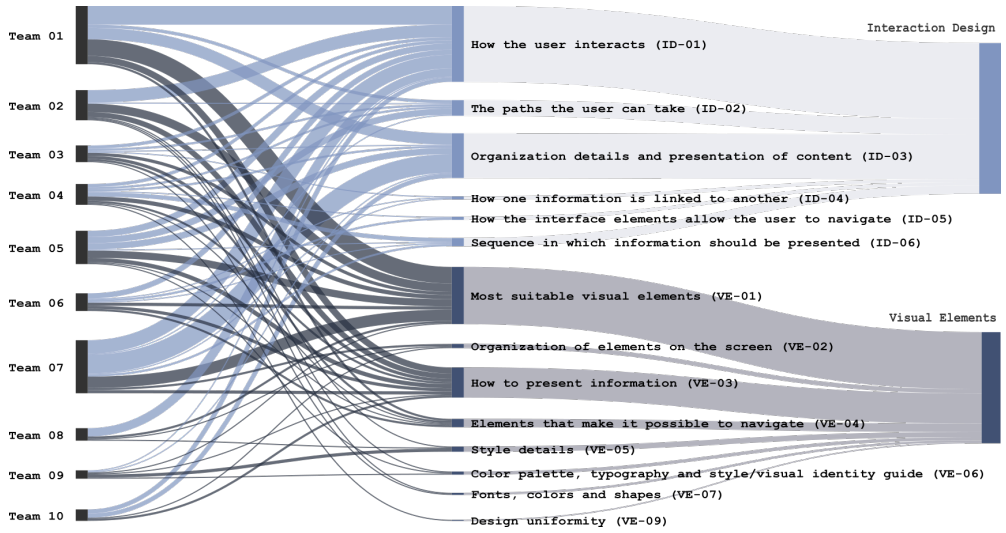

Figure 2: Guidelines usage

Table 5: Usual mistakes found in AC

Description of the usual mistake	Quantity
Not specify details about user interaction	14
Specify details based on personal perceptions	05
Not specify style details	02
Not specify the visual elements of the interface	01
Not specify the visual elements that allow interaction or navigation	01
Total	**23**

feedback from only one trainee by team. In other teams, two or more trainees reported feedback on ACUX.

Table 6: TAM questions used in the questionnaire

Dimension	ID*	Question
Perceived of Usefulness	**U1**	Using the guidelines allowed me to include UX aspects when writing ACs into USs more quickly.
	U2	Using the guidelines improved my perception of good practices for developing ACs in USs that meet the UX aspects.
	U3	Using guidelines is important and adds value to my work.
	U4	Guidelines made the result of the USs more interesting.
	U5	I find guidelines for developing ACs in USs that meet the UX aspects.
Perceived of Ease-of-use	**F1**	It was easy to learn how to use guidelines.
	F2	I find the guidelines easy to understand.
	F3	I find it easy to apply the guidelines.
	F4	Using guidelines makes my work easier.
	F5	Guidelines allow flexibility for the development of ACs in USs.

**The "ID" here relates to the identifier to each question present in the TAM questionnaire.*

We identified that only 4 out of 10 teams provided disagreement responses on the guidelines (Teams 2, 7, 8, and 10). Looking at these four teams, questions U1, F2, and F3 (see Table 6) received the most negative feedback, adding up to three disagreements each. Aside from Team 8, when we consider the other teams (Teams 2, 7, and 10) individually, each team provided at most two disagreement answers on ACUX per question. The negative feedback located among these four teams is consistent with the opinion of seven trainees. We observe that 74% (20 out of 27) of the trainees provided only positive feedback on the guidelines. This percentage is consistent with the

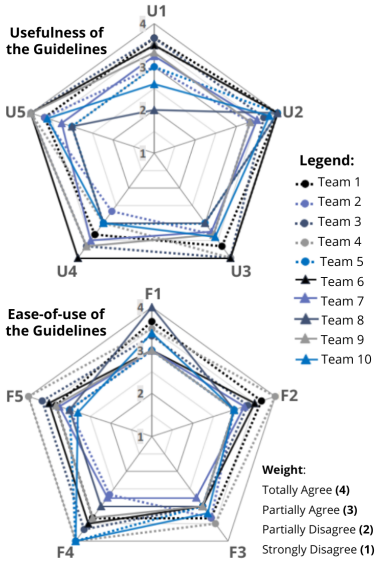

Figure 3: Trainees' feedback on the perceived usefulness and ease-of-use of the guidelines by team - TAM answers

averages for each team per TAM question, centered between 3 and 4 points (see Figure 3). Mean scores of 3.5 or higher were obtained in all TAM questions, especially in U2 (in eight teams), U5 (in seven teams), and F4 (in five teams) questions. From these results, we understand that the ACUX guidelines were well accepted by the teams, in relation to usefulness and ease-of-use.

We also coded the answers received from 9 trainees in the open question. We extracted line-by-line quotes [8] from each response using three closed codes. Therefore, we captured 9 quotes to *Positive Aspects* (from Teams 1, 2, 3, 6, and 7), 6 quotes to *Negative Aspects* (from Teams 1, 2, 5, and 7), and others 3 quotes to *Opinions and Suggestions* (from Teams 6, 7, and 10). Figure 4 shows some quotes and their respective code, that reflects chunks of the trainees'

feedback. The qualitative analysis allowed us to confirm the teams' acceptance of ACUX, and to identify the benefits of the guidelines in the trainees' vision. In relation to the guidelines' usefulness, it is shown to be useful for assisting in the implementation of AC (see Trainee A in Figure 4) and was pointed out as a tool capable of optimizing the trainees' work time (see Trainee B). Trainees recognized that the guidelines provided concrete insight on how to implement features with respect to UX (see Trainee C), and provided useful guidelines when documenting USs (see Trainee D). The trainees with less than one year of experience in software development and agile methods were the trainees who provided the most positive feedback about ACUX's usefulness in the open question.

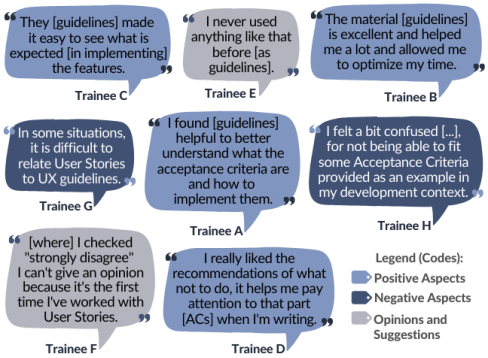

Figure 4: Quotes extracted from trainees' feedback

Taking into account the guidelines' ease-of-use, we understand that the results represent a reflection of the trainees' first contact with a structured and UX-oriented material, since one of the participants admitted to never having used something similar (see Trainee E in Figure 4). We infer that some trainees may not have previous experiences in writing USs, as one of them justified this as the reason for providing feedback as *Strongly Disagree* (see Trainee F). This lack of experience in designing USs may make it difficult for trainees to recognize the relationship between AC and UX aspects, which meet the particularities of their projects (see Trainee G and H). These variables can influence ACUX's ease-of-use. Looking at the negative feedback of Teams 2, 7, 8, and 10 in TAM, we see that two of them (Teams 2 and 7) were also present in *Negative Aspects* quotes. However, other trainees from these same teams reported only positive feedback in the open question. Trainees from Teams 1 and 5 also reported quotes of *Negative Aspects*, but did not report disagreement responses in the TAM.

5 RELATED WORK

The literature stresses the benefits and difficulties to UX integrate in the agile development [9, 12]. Interviews carried out with practitioners presented that USs are used in agile environments [9, 11], but limitations regarding writing and understanding of the USs are identified [1, 11]. Additionally, to align the team's members expectations in the elaborating of USs is also a challenge [1]. Thus, we trust it is necessary the development of guidelines that directs practitioners to writing USs, mainly to organize the UX information during software development [33]. In practitioners' opinion of software development, to use solutions as guidelines during elaborating USs presents benefits for Requirements Engineering [18].

Lucassen et al. [17] present the framework *Quality User Story* (QUS) framework that evaluates the textual structure of USs. The study evaluated USs, where 25% of them violate some quality criteria. The framework can be useful to developers writing more robust USs. Wautelet et al. [31] compared USs elaborated with the QUS framework [17] (1st group) between USs developed in free format (2nd group). The authors identified that the 1st group showed improved problem understanding by creating USs guided by a framework. Although the solutions are relevant, the development of AC and UX requirements are not identified. Based on case studies, Pereira et al. [23] propose a set of design patterns to organize UX information into online tools (*e.g.*, Jira). One of the patterns provides information to guide developers to include UX information in the AC.

From the studies that investigate AC in USs without excluding UX requirements, we identified a mapping of usability mechanisms that suggests how to describe these mechanisms from US or AC [19]. The usability mechanisms describe ways of providing feedback to the users, i.e., System Status Feedback (warning, status). The UserX Story is a grammar to USs, which incorporates the Personas [21] and the Nielsen Heuristics [20] to write AC. Professionals who used UserX Story in agile projects returned positive feedback about the proposal, besides to facilitate UX integration. Further study investigated how UX artifacts assist agile teams in writing AC [16]. Persona and Nielsen's Heuristics were the most influential techniques for developing USs between developers. Both UX artifacts contributed in the writing AC.

The main difference of our work was to provide a set of guidelines with practical examples to practitioners which are focus on the content that is writing in the AC. We did not identify studies that were directly similar to ours. Pereira et al. [23], for instance, support developers on identifying UX requirements that can be described in AC. Choma et al. [2], Lopes et al. [16] explore the writing of AC; however, their proposals are tight related to the use of UX-specific artifacts. Moreno and Yagüe [19] focus on the description of feedback messages in AC. Our proposal differs by providing guidelines that directly point out ways on how to write UX requirements linked to interaction design and visual element of the interface.

6 DISCUSSION

Considering our RQ1 (*How are the guidelines applied in the elaboration of Acceptance Criteria of User Stories by agile teams?*), we notice the usage of the guidelines was concentrated in four main directions, two of which were related to the Interaction Design UX dimension, and two were related to Visual Elements. However the fact that the ID group presented a higher number of occurrences when compared to the VE group did not provide sufficient evidence to relate this result to the usage of guidelines. It is important to remark that the main support of ACUX was to help teams at describing on the AC details on how the user should interact with the product (such as clicking, writing, pressing). Moreover, it has also made possible to include in the AC information about which

VE need to be present in the interface to enable users to be able to carry out their tasks (such as button, icon).

We concluded that teams also frequently described in the AC (*e.g.*, "grouping" and "lists") UX requirements on how content should be presented and organized (related to the ID group). In addition, ACUX helped teams to include information about how information should be presented (related to the VE group), such as "graphics" and "images" for example. Differently from Moreno and Yagüe [19]'s proposal that presents how to provide feedback to the users (i.e., warning, status), our guidelines provide a more broadly perspective by giving details of how to deal with different UX elements. The usage of the guidelines by the teams also showed us that there is a relationship between the guideline that describes the form of interaction (ID-01), and the one that suggests the description of the most adequate visual elements for carrying out the tasks (VE-01). This fact shows us that the use of ACUX may have helped teams to organize UX requirements and link both UX dimensions (ID and VE) in the same AC, reinforcing the statement in [33] that suggests guidelines for linking and organizing UX information.

The two main types of usual mistakes found are related, respectively, to the lack of details about a user's interaction with the product and the presence of information that allows for different interpretations. In other words information whose perception or understanding may be different for each person (such as *"clearly"* and *"easy to access"*). We report this as usual mistakes and error in the use of guidelines, since they are points that hinder the inclusion of UX requirements by allowing different interpretations, interfering directly in the final product. Kamei et al. [11] and Buchan et al. [1] argued that this type of error during the elaboration of AC is related to the difficulty of professionals with agile approaches having difficulties in writing and editing USs.

By answering our RQ2 (*What is the teams' feedback in relation to acceptance of the guidelines about the usefulness and ease-of-use?*), we could restate that the teams' acceptance was positive in the dimensions of usefulness and ease-of-use in relation to the guidelines. This acceptance is verified by the high averages obtained by teams, consolidated between 3 to 4 in all TAM questions (see Figure 3). Therefore, we identified more positive than negative feedback in relation to use ACUX by teams to developing AC in USs, especially in the questions as U2, U5, and F4. Consequently, the frequency of trainees' quotes related to the *Positive Aspects* was more expressive when compared to quotes of the *Negative Aspects* (see Figure 4). Overall, the teams classified ACUX as useful and ease-to-use material in elaborating AC and USs that consider UX requirements.

We found out that three out of six quotes gathered in *Negative Aspects* code are from the same participant. For the trainee who was acting as a developer, the guidelines was a complex solution to use. We identified that this trainee had less than 2 years of professional experience with agile methods (what is a common characteristic among the participants). We did not recognize a pattern of this type of feedback among trainees with the same time of experience in agile. Therefore, we infer that the developers did not receive guidelines in a friendly way, and possible causes for this are given in the literature. Professionals with technical purpose roles (*e.g.*, developers) are less positive about using templates and guidelines that guide the development of USs [18]. As the participant is experiencing a phase of building an agile mindset [11], when exploring new tools and practices, this may also be one of the causes. Promoting agile developers' contact with solutions that meet UX requirements in developing USs can enhance the description of requirements and promote an awareness of the importance of UX [16].

The quotes from Trainee G and H (see Figure 4), we infer that the difficulties perceived by the trainees in the guidelines' usage can be connected to the project scope for which each team was working (Trainee G was working on a business oriented project, and H was working on an app dedicated to mental health). Additionally, the teams that presented disagreements on the guidelines in some TAM questions also were working with apps about supermarket (Team 7), gaming (Team 8) and productivity (Team 10). Some trainees may feel that writing USs requires more initial work when breaking down a requirement into smaller parts [11, 18], in addition to low clarification of requirements by trainees [1] about area of the app.

The way teams recognized and defined UX requirements also could have influenced in the acceptance of the guidelines. In the R&D&I environments, during Sprint 0, the trainees carried out benchmarks and research with potential users (*e.g.*, interviews). Although teams were oriented on good user research practices, we infer that they would have found (*e.g.*, users' recruitment and define users' characteristics [9]) when performing these activities. These difficulties are common in software startups [9], and could have influenced in the usage of the guidelines. The literature shows that integrating UX with agile software development is a complex activity [12, 33]. Therefore, even with these difficulties, we observed that the guidelines were useful to teams' work for elaborating AC that focus on UX. It is worth noting that ACUX can be used not only to document but also to stimulate the team conversation and reflection about UX, since USs are built by the team.

7 THREATS TO VALIDITY

Even though the researchers actively engaged to reduce biases and external influences, we discussed the construct, internal, external and reliability validity based on Runeson and Höst [25]'s work.

In the construct validity, the presentation and explanation of ACUX to study participants might not have been sufficient to ensure a proper usage of ACUX. We minimize this both by introducing ACUX to the leadership team, who could act to remove any impediments teams were having, and by ensuring a researcher was available to answer trainees' doubts about the use of ACUX. In terms of internal validity, there is a possibility that researchers might have been influenced to find UX concepts in the analysed AC. To mitigate this aspect, two researchers have conducted the analysis steps separately and independently first. After, they discussed their findings and thus reached consensus.

External validity-wise, our results reflect a context where real-world mobile apps are developed and where majority of the teams are composed of trainees. However, we understand that the generalization of our results is not limited to this setting. We believe that difficulties such as gathering UX information dispersed in different artifacts [33] and not having specialized UX professionals [13] are obstacles faced by software industry professionals, whether trainees or experts when working in front-end, back-end, or designer roles.

Therefore, given UX requirements specification is a team responsibility and US/AC support professionals in this practice, we can generalize our results to other contexts of the software industry. Finally, reliability in the study was addressed by discussing and reaching consensus towards the steps and analysis to be conducted. Every analysis was performed independently and separately by one researcher. After that, a alignment meeting was held to ensure all findings and conclusions are agreed upon by the researchers.

8 CONCLUSION

This paper presented ACUX, a set of guidelines that supports software teams to the writing of UX requirements during the elaboration of AC. To evaluate our proposal we conducted a case study in a R&D&I center that develops mobile applications. We analyzed 242 AC produced by 10 agile teams and 27 individual responses about the usage of the ACUX guidelines. We conclude that teams were helped by the guidelines that respectively address details about the user's interaction with the product, as well as the definition of the most adequate VE to enable this interaction. Specifically the guidelines about user interactions with functionalities and the use of suitable elements (see ID-01 and VE-01 in Table 1) were relevant for the teams. Moreover, ACUX had a high acceptance. However, one participant suggests the need of having more context-related examples. We understood that further investigation should be conducted to see whether these results can be seen as a challenge of the trainees in applying ACUX in their project' context.

We understood that our work has academic and practical contributions. From academic perspective, we present a new proposal to support software teams in the writing of UX requirements from the use of a valuable artifact that is the US/AC. In the practical perspective, we consider that there is an educational aspect coming out of our research that brings significance for the practice. Our proposal can help teams that operate with a low number of individuals and that are overwhelmed by UX requirements.

As future works we intend to repeat this case study later in the same environment, since the experience level of participants will have increased by then, allowing us to further investigate whether our findings are concise and to assess to what extent the level of experience of developers and designers impacts our findings. We also intend to carry out a study to explore the use of ACUX in different types of projects or platforms. In our case study, all of the projects involved graphical interface design. Once the guidelines are independent and allow the selection of the ones that make sense for the project, future works can explore projects that do not have a graphical interface.

ARTIFACT AVAILABILITY

Due to our case study was conducted in an R&D&I environment, we are not authorized to make US/AC available so they present sensible information related to the innovation aspects.

ACKNOWLEDGMENTS

We thank the support of the grants #2020/00615-9 and #2020/11441-1, São Paulo Research Foundation (FAPESP), and the support of the CNPq - process number: 313312/2019-2.

REFERENCES

[1] Jim Buchan, Muneera Bano, Didar Zowghi, Stephen MacDonell, and Amrita Shinde. 2017. Alignment of Stakeholder Expectations about User Involvement in Agile Software Development. In *Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering*. ACM, New York, NY, USA, 334–343. https://doi.org/10.1145/3084226.3084251

[2] Joelma Choma, Luciana A. M. Zaina, and Daniela Beraldo. 2016. UserX Story: Incorporating UX Aspects into User Stories Elaboration. In *Human-Computer Interaction. Theory, Design, Development and Practice*, Masaaki Kurosu (Ed.). Springer International Publishing, Cham, 131–140. https://doi.org/10.1007/978-3-319-39510-4_13

[3] Mike Cohn. 2004. *User Stories Applied: For Agile Software Development.* Addison Wesley Longman Publishing Co., Inc., USA.

[4] Fred D. Davis. 1989. Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly* 13, 3 (1989), 319–340. https://doi.org/10.2307/249008

[5] Joseph L. Fleiss. 1981. *Statistical Methods for Rates and Proportions.* Wiley, London.

[6] Ron Garland. 1991. The mid-point on a rating scale: Is it desirable. *Marketing bulletin* 2, 1 (1991), 66–70. https://www.rangevoting.org/MB_V2_N3_Garland.pdf

[7] Jesse James Garrett. 2011. *The elements of user experience: user-centered design for the web and beyond, Second Edition.* New Riders, Berkeley, CA, USA.

[8] Graham R Gibbs. 2007. *Analyzing Qualitative Data* (1st ed.). SAGE Publications Ltd, London.

[9] Laura Hokkanen and Kaisa Väänänen-Vainio-Mattila. 2015. UX Work in Startups: Current Practices and Future Needs. In *Agile Processes in Software Engineering and Extreme Programming*. Springer International Publishing, Cham, 81–92. https://doi.org/10.1007/978-3-319-18612-2_7

[10] Robert Johns. 2005. One Size Doesn't Fit All: Selecting Response Scales For Attitude Items. *Journal of Elections, Public Opinion and Parties* 15, 2 (2005), 237–264. https://doi.org/10.1080/13689880500178849

[11] Fernando Kamei, Gustavo Pinto, Bruno Cartaxo, and Alexandre Vasconcelos. 2017. On the Benefits/Limitations of Agile Software Development: An Interview Study with Brazilian Companies. In *Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering*. ACM, New York, NY, USA, 1–6. https://doi.org/10.1145/3084226.3084278

[12] Pariya Kashfi, Robert Feldt, and Agneta Nilsson. 2019. Integrating UX principles and practices into software development organizations: A case study of influencing events. *Journal of Systems and Software* 154 (2019), 37–58. https://doi.org/10.1016/j.jss.2019.03.066

[13] Kati Kuusinen. 2015. Task allocation between UX specialists and developers in agile software development projects. In *IFIP Conference on Human-Computer Interaction*. Lecture Notes in Computer Science, Springer, ., 27–44. https://doi.org/10.1007/978-3-319-22698-9_3

[14] J. Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *Wiley, International Biometric Society* 33 (1977), 159–174. https://doi.org/10.2307/2529310

[15] Effie Lai-Chong Law, Virpi Roto, Marc Hassenzahl, Arnold P.O.S. Vermeeren, and Joke Kort. 2009. Understanding, scoping and defining user experience: a survey approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. ACM, New York, NY, USA, 719–728. https://doi.org/10.1145/1518701.1518813

[16] Larissa A. Lopes, Eduardo G. Pinheiro, Tiago S. da Silva, and Luciana A. M. Zaina. 2018. Using UxD Artefacts to Support the Writing of User Stories: Findings of an Empirical Study with Agile Developers. In *Proceedings of the 19th International Conference on Agile Software Development: Companion* (Porto, Portugal) *(XP '18)*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3234152.3234158

[17] Garm Lucassen, Fabiano Dalpiaz, Jan Martijn E.M. van der Werf, and Sjaak Brinkkemper. 2015. Forging high-quality User Stories: Towards a discipline for Agile Requirements. In *IEEE 23rd International Requirements Engineering Conference (RE)*. Ottawa, Canada, 126–135. https://doi.org/10.1109/RE.2015.7320415

[18] Garm Lucassen, Fabiano Dalpiaz, Jan Martijn E. M. van der Werf, and Sjaak Brinkkemper. 2016. The Use and Effectiveness of User Stories in Practice. In *Requirements Engineering: Foundation for Software Quality*, Maya Daneva and Oscar Pastor (Eds.). Springer International Publishing, Cham, 205–222. https://doi.org/10.1007/978-3-319-30282-9_14

[19] Ana M. Moreno and Agustín Yagüe. 2012. Agile User Stories Enriched with Usability. In *Agile Processes in Software Engineering and Extreme Programming*, Claes Wohlin (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 168–176. https://doi.org/10.1007/978-3-642-30350-0_12

[20] Jakob Nielsen. 2020. Usability Heuristics for User Interface Design. Nielsen Normamn Group (NNGroup). https://www.nngroup.com/articles/ten-usability-heuristics/ July 28, 2021.

[21] Don Norman. 2018. Ad-Hoc Personas & Empathetic Focus. jnd.org. https://jnd.org/ad-hoc_personas_empathetic_focus/ July 28, 2021.

[22] Dan North. 2006. Introducing behaviour driven development. *Better Software Magazine* (2006).

[23] Anathan Pereira, Abner Cleto Filho, Eduardo Guerra, and Luciana A. M. Zaina. 2021. Towards a Pattern Language to Embed UX Information in Agile Software Requirements. In *EuroPLoP'21: European Conference on Pattern Languages of Programs 2021, Graz, Austria, July 7 - 11, 2021*. ACM, 18:1–18:8. https://doi.org/10.1145/3489449.3489991

[24] Kai Petersen, Robert Feldt, Shahid Mujtaba, and Michael Mattsson. 2008. Systematic Mapping Studies in Software Engineering. In *12th International Conference on Evaluation and Assessment in Software Engineering (EASE)*. Bari, Italy, 68–77. https://doi.org/10.14236/ewic/EASE2008.8

[25] Per Runeson and Martin Höst. 2009. Guidelines for conducting and reporting case study research in software engineering, In . *Empirical software engineering* 14, 2, 131–164. https://doi.org/10.1007/s10664-008-9102-8

[26] Cristian Rusu, Silvana Roncagliolo, Virginica Rusu, and Cesar Collazos. 2011. A Methodology to Establish Usability Heuristics. In *The 4th Int. Conf. on Advances in Computer-Human Interactions (ACHI 2011)*. Gosier, Guadeloupe, France, 59–62.

[27] Eva-Maria Schön, Jörg Thomaschewski, and María José Escalona. 2017. Agile Requirements Engineering: A systematic literature review. *Computer Standards & Interfaces* 49 (2017), 79–91. https://doi.org/10.1016/j.csi.2016.08.011

[28] Ken Schwaber and Jeff Sutherland. 2011. The scrum guide. *Scrum Alliance* 21, 19 (2011), 1. https://billlewistraining.com/wp-content/uploads/2017/02/PMP-Agile-Study-Materials.pdf

[29] Tiago Silva da Silva, Milene Selbach Silveira, and Frank Maurer. 2013. Ten Lessons Learned from Integrating Interaction Design and Agile Development. In *2013 Agile Conference*. Agile Conference 2013, Nashville, TN, USA, 42–49. https://doi.org/10.1109/AGILE.2013.11

[30] Jonathan H. J. Souza. 2021. *ACUX: um guia para escrita de aspectos de UX em Critérios de Aceitação de User Stories - In Portuguese*. Master's thesis. Universidade Federal de São Carlos, Brazil. https://repositorio.ufscar.br/handle/ufscar/14347

[31] Yves Wautelet, Dries Gielis, Stephan Poelmans, and Samedi Heng. 2019. Evaluating the Impact of User Stories Quality on the Ability to Understand and Structure Requirements. In *The Practice of Enterprise Modeling*, Jaap Gordijn, Wided Guédria, and Henderik A. Proper (Eds.). Springer, Cham, 3–19. https://doi.org/10.1007/978-3-030-35151-9_1

[32] Claes Wohlin, Per Runeson, Martin Höst, Magnus C. Ohlsson, Björn Regnell, and Anders Wesslén. 2012. Experimentation in software engineering. In *The Kluwer International Series in Software Engineering*. Springer Science & Business Media.

[33] Luciana AM Zaina, Helen Sharp, and Leonor Barroca. 2021. UX information in the daily work of an agile team: A distributed cognition analysis. *International Journal of Human-Computer Studies* 147 (2021), 102574. https://doi.org/10.1016/j.ijhcs.2020.102574