Usability Evaluation Practices within Agile Development

Tiago Silva da Silva
ICT/UNIFESP
Universidade Federal de São Paulo
São Paulo, SP, Brazil
silvadasilva@unifesp.br

Milene Selbach Silveira
FACIN/PUCRS
Pontifícia Universidade Católica do Rio Grande do Sul
Porto Alegre, RS, Brazil
milene.silveira@pucrs.br

Frank Maurer
CPSC/UofC
University of Calgary
Calgary, AB, Canada
frank.maurer@ucalgary.ca

Abstract

Despite having different underlying concepts, agile methods and User-Centered Design aim ultimately at producing high quality software. Aiming at providing better understanding on how usability evaluation is addressed within agile environments, we carried out multiple-case studies in two large IT companies. On the one hand, our studies corroborate the literature with regards to the execution of usability inspection evaluations on lightweight prototypes, taking advantage of the iterative nature of Agile. On the other hand, we observed contrary to the literature that this iterative nature of Agile has not facilitated the execution of user testing sessions. Furthermore, we noticed that it is extremely difficult to perform traditional user testing sessions due to the tight schedules inherent to Agile. Finally, we describe a set of practices that can help improve specific artifacts in different stages of an agile process in order to successfully evaluate software product usability.

1. Introduction

Agile methods have transformed how teams organize the development of a software project. Rather than months of development, followed by months of field test leading up to a release, agile methods organize software development into short iterations with continuous testing and a flexible release date [1]. The agile approach promises to quickly produce more useful and reliable software, and with better control than traditional development methods.

User-centered design (UCD) or Human-centered design is characterised by active involvement of users and a clear understanding of users and their tasks; an appropriate distribution of work between users and technology; the iteration of design solutions; and multi-disciplinary design [2].

Moreover, according to Kumar and Herger [3], UCD is an approach that puts the user and their goals at the center of the design and development process. It strives to develop products that are tightly aligned with the user's needs instead of products that are technology-centered.

By integrating UCD and Agile we can increase the chances that the software produced adds value to the business and is desirable to the end user. There is no point in using software if it does not deliver value [4].

Integrating Agile and UCD is not only possible but it is beneficial to the success of agile teams developing interactive systems. Agile and UCD are a natural fit [5].

Unfortunately, there are still a large number of companies that do not realize the importance of their products' usability. Some companies still focus only on the usability evaluation at the end of development rather than following a user-centered approach from the start.

However, usability evaluations are better than ignoring usability concern completely. They are often the first step for companies that are aiming at adopting a user-centered development cycle and allow companies to determine usability issues of their products.

In order to understand how usability evaluation is addressed in agile projects – which methods are used on, which artifacts are created and at what stage, we carried out multiple case studies in two large Information Technology (IT) companies.

As this paper is part of a longitudinal study that aims at providing a better understanding of the integration between UCD and agile [5] [6], it focuses on usability evaluation within agile processes.

The remainder of this paper is organized as follows: in Section 2, we review related work on agile development and usability evaluation. Section 3 details the research method and describes how the studies were carried out. Section 4 presents our findings with respect to usability evaluation within agile

1530-1605/15 $31.00 © 2015 IEEE
DOI 10.1109/HICSS.2015.607

IEEE computer society

environments. In Section 5, we discuss our results. We conclude the paper in Section 6 with some remarks on what we have achieved as well as suggestions for future work.

2. Related Work

Recently, Silva et al. [5] conducted a systematic literature review on User-Centered Design and agile methods. The review addressed topics like usability evaluation of prototypes and of working software as well as user testing, just to mention a few.

The related work reports a combination of usability evaluation methods depending on the artifacts available and on the stage of the development the product.

Fox et al. [7] indicate that at different stages of the development cycle different UCD methods can be observed, The study describes that once the User Interface (UI) features have been implemented, the UI is passed back to the User eXperience (UX) designer for verification and usability testing. According to their findings, verification consisted of determining if the development team had followed the design rules set out by the designer. If the implemented features are verified as correct and they pass the usability tests, they are marked as finished features and await release to the customer.

Accordingly, Sy [8] reports three stages of usability design: the Early Design Cycle; the Mid-Release Design Cycle; and the Late Design Cycle. According to her, breaking design down into chunks gives them the freedom to mix and match different types of usability investigations into the same session, which enables them to juggle more than one design, and more than one type of usability investigation at the same time. Design chunks are what allow them to elicit more data from fewer users.

Furthermore, UX Designers often evaluate the products from a qualitative viewpoint regardless the name they give to the evaluation method.

Federoff and Courage [9] report that once the UX team has designs ready for feedback, they are rapidly taken into the laboratory for testing with users who represent those who will actually use the product. According to their report, the UX team conducts a tremendous amount of formative testing due to the nature of their business. Formative test or evaluation is a collection of “find and fix” usability engineering methods, focused on identifying usability problems before a product is implemented [10].

Federoff and Courage [9] also reported that, since moving to agile, the UX team at the company had only on rare occasions conducted a standard usability test. According to Federoff and Courage [9], Rapid Iterative Test and Evaluation (RITE) [11] is almost always the method employed to evaluate designs. According to these authors, RITE is based on the principle of iterating the design as each participant test the system, and the problems are fixed before the next participant tests the system. This differs from a traditional usability test by emphasizing extremely rapid changes and verification of the effectiveness of these changes [11]. In a traditional usability test, changes are only made after a full set of participants has evaluated the design [11].

Ferreira et al. [12] report that after the first two-week iteration, UX designers try and test the application with their client. They mentioned that one of the strong themes that emerged was the relationship between iterations and user testing. Although usability was seen as beneficial, participants admitted that sometimes its techniques were traded off against development time.

Other times, usability evaluations are performed on lightweight prototypes rather than on functional software. There are several studies using paper prototypes to validate assumptions and to perform early evaluations with users.

Hodgets [13] reports that designers sketch views on paper and do early usability testing with users. This does not require an expensive usability laboratory or video recordings, just someone who is familiar with the process, plus the paper prototypes, some real users and a notebook. Ideally there is a usability specialist in the team for performing this task, because it is not recommended using staff that have worked on the designs, as they are too familiar with the issues. The author also states that most of the activity is early in the project, with usability testing taking place throughout. Having a user or a representative of the user in the team is said to be better than no contact with users. However, he states that it is not the same as understanding real users’ needs. To do so, he suggests the use of Contextual Inquiry [14] to observe and interview users.

Miller [15] reports that usability tests are used to evaluate a product design by watching the intended users of the product try it – or a prototype of it – for its proposed use, and seeing what problems can be found. The author claims that usability tests are good for discovering issues with learning, discoverability, error rates, and speed of use, and also that they uncover issues with incorrect or omitted feedback. Usability tests can also uncover missing features that are needed to complete a workflow. Along with the usability test data, the author reports that the team gathered customer input from their discussion forums, and visited customers to conduct interviews and collect contextual inquiry data. As can be seen, some reports [15] [13] state that development iterations drive usability testing in agile environments, and the results of user testing sessions from the previous sprint may be an input for the next sprint.

In their experience report, Meszaros and Aston [16] report that usability testing based on paper prototypes and early versions of the software were added to the agile development process for their system's second release resulting in a significant reduction of usability-related rework. The paper prototype became a tangible representation of the project vision that was used in many ways that contributed to the success of the project. According to the authors, conducting usability testing once an application has been built is reasonably obvious even if some of the detailed practices are less than well known to agile practitioners. They claim that usability testing on an application that has not even been built yet might seem foreign to those uninitiated in the ways of User or Usage-Centered Design, but it is in fact quite easy to do.

While some authors report on user testing are part of agile processes, more report on evaluations in which there is no user participation, such as heuristic evaluation and expert reviews.

For instance, Hussain et al. [17] reported that in projects involving XP, expert-based user experience evaluations solve the problem of ad-hoc input. Mock-ups – in early phases – and screens – in later phases – are sent to the Human-Computer Interaction (HCI) experts who then give ad-hoc input by using the different communication channels like instant messaging, email and video conferencing. UCD methods had to be tweaked in a way to be less time and cost intensive by involving less experts or users than recommended by UCD experts. However, in agile processes, these tests are done much more frequently and, therefore, the results can be accumulated.

Chamberlaim et al. [18] report that in their Scrum process user evaluation of the product is encouraged on a monthly basis, as users are ideally present during the sprint. The authors reported only one user test during their observation period and this test was carried out with an internal user within the organization. According to their report, this user was testing a part of the system to ensure it fulfilled the team's requirements before it was released. The Product Owner attended the sprint review where a demo of the work carried out. This allowed the product owner to prioritize fixing of usability issues for the next sprint.

All related work is in line with the statement of Nielsen [19] that agile software development methods force the usability professionals to change and develop new ways of performing usability evaluation.

In order to verify how usability evaluation has changed – which methods are applied to which artifacts in which stages – in agile projects, we carried out multiple case studies. We observed a set of practices – organized according to the different stages of agile development – to allow a smooth integration between usability evaluation and agile methods.

3. Research Method

As aforementioned, multiple case studies were carried out in two large IT companies, two studies in a company in Canada and two in a company in Brazil. Our studies' goal was to investigate how User-Centered Design is addressed within agile environments. Usability evaluation is inherent to this goal and was carefully studied.

We adopted an ethnographically-informed approach [20]. According to Robinson et al. [20], in classic ethnography, the researcher studies practice by immersing themselves in the area under study for several months if not years, documenting what takes place via a range of means. According to these authors [20], it has become quite common to adapt ethnographic traditions in order to conduct shorter studies that fit more easily with a software product development cycle.

In some of our studies, audio and video recording in the setting of practice was not possible because of considerations of time or of commercial confidentiality.

Therefore, we collected our data by observations and interviews with practitioners, but we did not attempt to influence the practice during the study. In order to ensure data confidentiality between the researchers and companies, we signed a non-disclosure agreement with the companies. This restricts our reporting on the details of the companies' products but does not encumber any discussion of their processes.

In the following, we describe the organizational settings, the projects, the participants, as well as the data collection and analysis.

3.1. Organizational Settings

In **Company 1**, the analyzed development team was one of several Scrum teams in the company working on software development[1]. At the time of the study, this company had followed the Scrum [21] methodology for two years. The developers and designers were seated in an open-plan office space located in the same building, though they were not jointly seated.

In **Company 2** there are no separated UX team and Developers team. This company had followed the Scrum methodology for at least one year. Teams did not share developers or UX designers with other teams. The teams were selected for the study because they

[1] The company also has teams working on hardware development.

were the most experienced agile teams in the company. Developers and designers were co-located, seated in an open-plan office space located in the same building.

3.2. Projects

We followed two projects in **Company 1**: *Project X* consists of the development of new features for an existing product of the company; and *Project Y* consists of the development a mobile version for an existing product of the company.

Company 2 is not structured by projects, but by digital products. We studied two different teams developing two different products: *Product X* consists of a web portal about agribusiness in the country; and *Product Y* consists of a web portal of services and opportunities providing addresses and data about companies and services.

3.3. Participants

In **Company** 1, our study involved a team of seven "developers" and one UX designer that worked on both projects aforementioned. The developers were part of the 'Development Team' and the designers were part of the 'UX Team'. The developers had been developing software with Scrum for approximately two years. Even though called developers, different individuals in the team have their own roles according to their area and skills. The roles were Project Manager/Scrum Master, Product Owner, Technical Leader, Developer, and Tester.

Information architects, graphical designers, and interaction designers compose the UX team. Each project has one UX designer, but a UX designer usually work with more than one development team. The same goes for Project Managers, and they are also known as Scrum Masters in the teams.

In **Company 2**, our study involved UX designers and their interactions with an agile team working on the same product. The teams are composed of a Product Leader/Product Owner, a UX Designer, a Developer, a Tester, and one person responsible for Search Engine Optimization (SEO).

One team – *Product X* – has two individuals focused on UX, a UX Designer and a Graphical Designer, whereas the other team – *Product Y* – has just one UX Designer who performs the role of a Graphical Designer as well.

Table 1 summarizes the background and context of the UX designer on the teams.

3.4. Data Collection

We used two first-degree techniques [22] for data collection: observations and interviews.

During observations, we manually took notes of the observations during meetings in both companies. In **Company 1**, we also shadowed a UX designer during his activities and observed meetings in which he was involved. For instance, we observed 4 UX group meetings. During 3 months, we also observed: 2 requirements meetings and 1 retrospective meeting in *Project X*; 1 demo meeting, 3 planning meetings, 3 retrospective meetings and 2 user testing sessions in *Project Y.*

We interviewed 3 members of the UX Team that work in different projects and one Project Manager[2].

In **Company 2**, we observed 5 daily meetings of *Product A*, 5 daily meetings of *Product B* and interviewed the UX Designer and the Product Leader of the two selected teams during 25 working days.

3.5. Data Analysis

Our data analysis was inspired by grounded theory [23]. Qualitative analysis provides a general characterization based on the researchers' coding schemes, instead of quantitative analysis that rely on quantitative measures to describe the data. The way the data is coded will affect its interpretation and the possible courses for its evaluation. Therefore, it is important to ensure that the coding scheme reflects the research goals.

In our research, as suggested by Davison [24], we triangulated the data from different sources and different approaches – observations and interviews were performed in different projects with people with different roles in the company.

Analysis of the qualitative data was conducted in a series of steps. First, we performed an open coding by extracting initial codes – *Use of Prototypes*, *Usability Inspection*, *User Testing* – from the field notes using a grounded theory approach that starts without any codes but generates them from the field notes. When these codes saturated, we started the focused coding and applied the list of codes generated in the systematic literature review (SLR) [5] – *Design Up Front, Prototyping, Usability Evaluation*.

4. Results

In this section, we present the studies' results according to our themes: *the use of prototypes* to perform usability evaluation, *usability inspection* and *user testing*.

4.1. Use of prototypes

In **Company 1**, we noticed that the UX team uses prototypes[3] of different fidelities to perform different

[2] They are also called Scrum Masters or Product Leaders depending on the team.

[3] A prototype can be used to test big or small ideas and it can range from lower fidelity, meaning it is conceptual, unpolished, and spare of detail to higher fidelity, precisely representing part or all of a final user interface [28].

Table 1. Summary of the analyzed organization settings

	Context	UX resources for project	Co-located with developers	Method	Experience with agile (in Years)	Background
UX1	In-house	1 (shared)	No (same building)	Scrum	2	Design
UX2	In-house	1 (shared)	No (same building)	Scrum	2	Psychology
UX3	In-house	2 (dedicated)	Yes (same room)	Scrum	1	Design
UX4	In-house	1 (dedicated)	Yes (same room)	Scrum	1	Design

types of usability evaluations iteratively along the entire process development.

Depending on the project's needs and on the Designer's background, **Company 1** used paper prototypes, low-fidelity prototypes or high-fidelity prototypes to perform both usability inspection evaluation and user testing to validate their ideas.

Inspection evaluation consists of experts using different methods to evaluate a user interface without involving users. In contrast, user – or usability – testing is a technique used to evaluate a product by testing it on users [25].

For instance, UX1 stated: "*We used to use paper prototype, high fidelity prototype, product... some prototyping tools, sometimes high-fi prototypes, sometimes low-fi*" [UX1].

We also noticed that since the members of the UX team have different backgrounds, some of them can code some functional – high-fidelity – prototypes and some of them just cannot do it, as we can see at: "*It's tricky for UX people to code*" [UX2].

In **Company 2**, we noticed that the UX Designers use high-fidelity and low-fidelity prototypes. Concerning low-fidelity prototypes: "*Once the product is defined, I prototype it in two or three weeks. Paper prototype to communicate between us and some HTML to present to directors.*" [UX4].

In this company, UX Designers used low-fidelity prototype to represent and validate some ideas with the Product Leaders. High-fidelity prototypes, in turn, are used to test issues related to graphical design and to verify workflows: "*As we work close (UX Designer and Business Analyst), we validate our ideas together by prototyping and then specifying User Stories*" [UX4]; "*We put an effort to build an HTML version, then the effort to add some links and build a functional prototype is minimal, then we can present them to...*" [UX3]. These quotations also reveal that UX Designers are an integral part of the team, as previously described.

We also observed the use of low and high fidelity prototypes to validate concepts through some usability inspection evaluations.

4.2. Usability inspection

In **Company 1**, we observed that the UX team performs some usability inspection evaluations: a member of the UX team designs a prototype and then another member evaluates it. Even though they do not follow a specific evaluation method, we observed throughout the studies that it works well for them: "*We perform some experts evaluations, peer review*" [UX1]; "*We perform some inspection evaluations, peer review with some UX member*" [UX2].

In **Company 2**, teams conducted some peer reviews. Sometimes with the Graphical Designer and sometimes with the Product Leader, according to the following quotes[4]: "*Me and UX peer review the UIs*" [PL1]; "*We perform a lot of informal evaluations. Me and the Graphical Designer*" [UX3].

4.3. User testing

In **Company 1**, we could observe that user testing sessions with real users are rarely performed, even when the project is in its final stages. The Designers used to perform user testing with internal users, justifying that there is always new and old employees with different profiles and backgrounds: "*Internal studies... new people and old people from inside the Company (...) With real users just at the final stages of the project*" [UX2].

In **Company 2**, although they mention that they understand the importance of user testing, they could not fit them into their process yet. They only mention the use of web analytics tools: "*We used some tracking tools. But we did not perform any User Testing yet*" [UX3].

One of the studied teams in **Company 2** has a database of users that they can call to participate in focus groups: "*As we have a set of users (database of volunteers), we can call them and carry out some focus groups. We have 4 different personas with them*" [UX4]. Having a database of users is an interesting practice that facilitates recruitment for user testing.

[4] Noteworthy that since the study in Company 2 was carried out in Brazil and the meetings as well as the interviews were performed in Portuguese, these were translated to English by the researchers.

However, this team is not performing user testing sessions yet. According to the participants, the company is interested in checking the system's usability with real users, however they could not fit user test sessions into their cycle yet. "*We did not perform any User Testing*" [UX3]; "*We don't perform User Testing yet.*" [UX4].

4.4. Summary of Results

Some results from our studies corroborate the literature review [5]. The use of prototypes to validate assumptions and to perform usability inspection evaluations is a clear example.

Nevertheless, we noticed a practice not mentioned in the literature that is conducting expert peer reviews on prototypes, leading to an evolutionary design. The closest practice observed in the literature was the expert evaluation mentioned by Hussain et al. [17].

The observed teams rarely perform user testing sessions, not even on lightweight prototypes. Only a few user testing sessions were observed in **Company 1**. This issue confirms statements from the literature that report on the difficulty of integrating user testing into short iterations [13].

The main feedback from UX designers is that they do not have time to perform user testing sessions. However, whenever they are able to perform some kind of evaluation, their results are useful and serve as input for planning the next iteration.

We believe that the main similarity between the companies studied is the fact that they perform peer reviews on prototypes to validate ideas and designs.

The main difference between the companies is with regards the dedication of the UX Designers to a single team.

Company 2 has a UX designer as a full-time team member, allowing a greater involvement of the designer with the project, whereas in **Company 1** the UX designers works for multiple teams concurrently. Even though there is a UX team within the company, UX members are not full time members of a specific development team, therefore they are always working on many projects at the same time: "*We usually work on multiple projects, but it's really easier to work on only one product. It's so much easier to get involved; you know what's going on. Your attention is right directed*" [UX2].

In **Company 1, e**ven the Project Managers complain about the results provided by UX Designers: "*UX people should be more committed, not just involved*" [PM]. Looking at this statement from the perspective of usability evaluations, we noticed that sometimes the designers are not able to find the proper time slot to perform usability evaluations and provide useful feedback to the rest of the team. Also, when working on too many projects with limited time for each, UX Designers cannot even plan user testing sessions. This indicates a shortcoming of the actual process as the literature [15] [13] reports that the results of usability tests can guide the development process. Most of the time, designers have barely enough time to perform some expert evaluations. Only rarely they carry out user test sessions, and even these are then sessions with internal users.

Company 2 does not perform user testing sessions. Although this company has a database of users – Design Partners –, they could not find the right time to perform user testing sessions. These users have so far only participated in focus groups.

In the following section, we discuss our findings with regards to usability evaluation according to the different stages of an agile development cycle, considering Early Design stage, Mid-Release Design stage, and Late Design stage, as reported by Sy [8].

5. Observed Practices for Usability Evaluation within Agile Processes

Combining our literature review [5] and multiple case studies carried out in the two companies, we determined a set of practices that have been applied to different artifacts, in the different stages of an agile process, in order to evaluate the usability of a product. These practices are summarized in Table 2. In the next sections, we define and detail each of these stages.

5.1. Early Design Stage

By "Early Design Stage" we mean the first iteration of the agile cycle and a pre-iteration cycle, also referred in the literature as Iteration 0, see [7] and [8].

Iteration 0 is a research stage and provides an opportunity for UX designers to work close with the Business Analyst. Thus, improving their understanding both of the business – by the UX designers – and of the users – by the Business Analysts. Assuming that this is the first release of the project, it is more about setting up the environment, gather requirements, and understand users and their needs. Rapid Contextual Design [14] is a good option to perform these research tasks [13].

Also some usability inspection evaluations are recommended in order to check ideas and assumptions for the design in a peer review, as summarized in the first row of Table 2. We have to keep in mind that these suggested practices are ideal if the companies involved have their main focus on the user.

It is unlikely to have user testing sessions in this stage, even on lightweight prototypes. However, some positive reports [12] suggested early user testing sessions in order to validate design ideas with target users.

Iteration 1 receives as input data gathered in the previous iteration as well as initial UI designs. UX designers may perform usability inspection evaluations on lightweight prototypes – low-fidelity or high-fidelity prototypes – or even on what has been coded in the current iteration to provide feed-back to developers.

At this stage, RITE is a good option to evaluate the usability through user testing sessions [9]. This testing relies mainly on lightweight prototypes, because they are easy to handle, change, and adapt. RITE testing acknowledges that initial designs will be flawed, but that a successful design will be achieved through user input and iteration [11].

RITE and agile development go hand in hand [9]. RITE is fast, iterative, inclusive of the team, collaborative, and in the end produces a prototype that acts as a design specification for development. UX designers often make changes after seeing one participant having a problem. In RITE, the designer is empowered to change the interface if (*i*) the participant is believed to be a representative user for that feature; (*ii*) the problem is believed to be understood and; (*iii*) a solution can be proposed.

In an attempt to map the usability goals described by Rogers et al. [26] – effective to use; efficient to use; safe to use; have good utility; easy to learn; easy to remember how to use –, we believe that we can draw some conclusions.

With regards to early design evaluation, we observed that the teams have used informal methods to validate business ideas and check assumptions for design. Regarding usability goals, at this stage, they were trying to check if the product has good utility and if it is effective to use.

5.2. Mid-Release Design Stage

At this stage of the project, usability inspection can also be performed both on prototypes and on the code produced in this iteration. However, at this stage, UX designers already have some working software from the previous iterations to evaluate. Even when carrying out test sessions with internal users, the team is still able to find and possibly fix potential problems.

UX designers should keep carrying out Contextual Inquiry because they will need a basis for designing the upcoming iterations and releases. Furthermore, the results of usability evaluations performed at this stage will provide information for the next iteration and drive product development.

With regards to the usability goals for the mid release stage, UX designers performed usability inspections of prototypes to check if the product has good utility and if it is safe to use. With regards to usability inspection evaluations of working software, the UX designers were checking if the product is effective and safe to use, and if it is easy to learn. UX designers were using usability tests on prototypes to check if the product is effective to use, easy to learn, and safe to use. However, user test sessions on working software were also used to check if the product is efficient to use and easy to remember.

5.3. Late Design Stage

Considering this stage as the last iteration before the next product release, usability inspection may not be too useful for concerns of the current release. However, this feedback may be useful for the next iteration and release, since there are usually many releases. Moreover, usability inspection methods like guidelines reviews are a good opportunity to check UI standards compliance. Also, UX designers may use RITE to evaluate the code from this current iteration.

According to the literature and based on our results, we observed no complaints about the usability of the product due to a lack of user testing.

Even though we did not observe any of the companies carrying out user testing sessions before the actual release, we hypothesize that the demo meeting is a fair opportunity for gathering feedback on the system's usability. Instead of presenting a demo of the developed features, the agile team could observe the users or users representative using the product by themselves. Developers are usually present during the demos and having them watching a user handling the system during test sessions might provide insights into usability issues.

Before the product is released, evaluations must be focused on identifying issues in the working in order to check if it is efficient to use, safe to use, easy to learn; easy to remember how to use.

6. Discussion

Our studies could not strictly state that there are benefits from inserting the results of user testing into the next iteration because this did not happen in the observed projects due to the tight schedule of agile development. Unlike, we noticed many usability inspection evaluations.

With regards to usability inspection evaluations, our studies corroborate the literature in the sense of that agile development methods have been forcing usability professionals to develop new ways of performing usability evaluations.

As depicted in the studies, some "new usability evaluation methods" emerged within agile. For instance, peer review usability evaluations by pairing with Business Analysts, Product Owners or even with other Designers.

Table 2. Summary of practices - usability evaluation methods -, artifacts and goals to evaluate usability at the different stages of an agile cycle, based on the literature and on the studies' results.

Design Stage	Usability Evaluation Method		Used Artifacts	Goals
Early	*Inspection*	Peer review with Business Analysts	Paper prototypes, sketches	Validate business ideas
		Peer Review with UX Designers	Low-fi prototypes, wireframes	Check assumptions for the design
	User Testing		High-fi prototypes, wireframes, mock-ups	Validate ideas with target users
Mid-Release	*Inspection*	Peer review with UX Designers	Low-fi prototypes, wireframes, mock-ups	Validate ideas for the next iteration
		Peer review with UX Designers	Working software	Check standards compliance
	User Testing		High-fi prototypes	Prepare the system to be released
			Working software	Prepare the system to be released
Late	*Inspection*	Guideline review	Working software	Check standards compliance
	User Testing		Working software	Get the system ready to release

Furthermore, there are methods applied to different artifacts, like paper prototypes, low and high-fidelity prototypes, working software from the previous sprint and from the current one.

This is interesting because UX designers we observed have diverse backgrounds. As previously mentioned, some designers can code and some cannot. Thus, these different designers' background reveals that regardless of the artifact created, designers find a way to evaluate them from the usability perspective by adapting usability evaluation methods to each artifact produced.

In this sense, Ferreira et al. [12] state that even when much UX design has been done up front, interleaving development iterations allows usability testing of working software, i.e., usability testing is done more often at a time when changes can still be incorporated more easily into the software product. However, to make it practical in an agile process, UX designers need to use lightweight prototypes as well as lightweight evaluation methods, for instance, Discount Usability Engineering [25].

RITE is an interesting approach because it is based on the principle of iterating the design as you test each participant. This is in contrast to a traditional usability test, where changes are only made after a full set of participants has evaluated the design [11]. Furthermore, in traditional usability tests, by the third or fourth user, everyone knows what most of the problems are, so there really is little value in continuing to test [27].

Nevertheless, in the analyzed projects, UX designers could not carry out user testing sessions during the iteration. Hence, they used usability inspection evaluations, in order to guide the teams in order for the development of the product. Moreover, based on the results of our studies we hypothesize that their approach – usability inspection evaluations and expert peer reviews iteratively throughout the development process – may reduce the need for expensive traditional usability test sessions at the end of the process.

We believe that the scheduling issues are in part due to the fact that UX designers were working on too many projects at the same time. This limits their ability to properly plan, perform, and report the usability tests. However, as already mentioned, usability inspection evaluations as well as user testing sessions with internal users reveal problems. Real target users might be required in order to reveal additional problems resulting from a misunderstanding of the actual work process.

With regards to evaluating the usability of the product for the next iteration, we believe that the demo meeting is a good opportunity to perform user test sessions. By definition, in a demo meeting, designers and developers demonstrate the product. Rather, they should allow the user or the user representative to experience the product.

It is important to bear in mind that none of the previously- mentioned suggestions – neither those derived from the literature nor those from the case studies – will work if the company does not provide the right environment. For instance, Miller [15] states that not all companies organize their usability resources in ways that allow the previously-described methods and artifacts. Companies should enable their UX and development teams to work together more closely, saving design and development time and effort, and producing a better product for the end-user.

With regards to the limitations of our study, we did not follow the entire development processes of the projects. Therefore, we cannot affirm that usability tests were not necessary, although we did not observe any complaints about the products' usability. Also, we are careful on generalizing from our findings. Although the teams analyzed in these studies are considered not atypical, these studies do not cover all the possibilities and the contexts can vary widely.

7. Conclusion

In recent years several authors have addressed the integration between agile and User-Centered Design and the overall picture is firm enough for researchers and practitioners. For instance, Sy [8] Ferreira et al. [12], Fox et al. [7] and Silva et al. [5] arrived at very similar proposals. Now we need to confirm these studies, for instance, how usability evaluation has been performed in agile environments.

In order to address this issue in real-life projects, we carried out multiple case studies in two large agile companies from Canada and Brazil. As already mentioned, one of our goal was to better understand how usability evaluation occurs within agile environments.

We observed that usability evaluations are mainly inspection evaluations performed by experts and that they are performed on an ongoing basis starting from Iteration 0. UX designers are evaluating low-fidelity prototypes to check design ideas, business goals, effectiveness of use and utility of the product.

The usability evaluation, as well as the UCD process, is a recurrent and iterative process, and the iterative nature of agile methods support these activities that may guide the product development process by improving design and prioritizing tasks.

At the final stage, UX designers focus on what must be delivered, fixing what is possible for the upcoming product release. These short iterations require competencies not much explored of the UX designers, that is, breaking down tasks into smaller pieces and then prioritizing them.

Performing more usability inspection evaluations and not too many user testing sessions helps to drive the product development and is better than having usability evaluation results only at the end of the development.

We hypothesize that the close collaboration plus usability inspections reduce the need for expensive usability studies as the team gets constant feedback on the system and fixes usability issues on ongoing basis.

Our data does not raise any concerns about the usability of the four different systems that we studied, although all four projects did not run large scale usability tests. Maybe they are simply not needed. In a future study, we will try to compare the usability of systems developed with the integrated Agile UCD approach with a more traditional approach where usability tests are conducted late in the development process.

We believe that agile teams will ask for systematic usability and user experience evaluations as they begin to understand the value of the evaluation results as input for the next iterations. This will lead ultimately to improving the quality of the product under development and reducing costs. We see inviting developers to observe usability testing as a means to raise awareness of UX issues. We propose to include a user test to be included in iteration planning meetings as an alternative or addition to a system demo.

It is worthwhile to mention that the challenge is not only which usability evaluation method to use, but also which artifacts are available to perform usability evaluations, how much time the team has and how these evaluations will be helpful for the rest of the product development. This is one of the really interesting topics that are left for future research.

Finally, to operate successfully in the agile environment, UX designers need to understand the agile culture. We believe much of this culture to be helpful and supportive for addressing usability concerns. Conversely, agile developers must understand the importance of having usability evaluation throughout the process.

References

[1] Hugh Beyer, *User-Centered Agile Methods*, John M. Carrol, Ed.: Morgan & Claypool Publishers, 2010.

[2] ISO:13407, "Human-centred design processes for interactive systems," International Organization for Standardization, Standard 1999.

[3] Janaki Mythily Kumar and Mario Herger, *Gamification at Work: Designing Engaging Business Software*. Aarhus, Denmark : The Interaction Design Foundation, 2013.

[4] Lindsay Ratcliffe and Marc McNeill, *Agile Experience Design: A Digital Designer's Guide to Agile, Lean and*

Continuous. Berkeley, CA, United States: New Riders, 2012.

[5] Tiago Silva da Silva, Angela Martin, Frank Maurer, and Milene Silveira, "User-Centered Design and Agile Methods: A Systematic Review," in *AGILE*, Salt Lake City, 2011, pp. 77-86.

[6] Tiago Silva da Silva, M. Silveira, F. Maurer, and T. Hellmann, "User experience design and agile development: From theory to practice," *Journal of Software Engineering and Applications*, vol. 5, no. 10, pp. 743-751, October 2012.

[7] David Fox, Jonathan Sillito, and Frank Maurer, "Agile Methods and User-Centered Design: How These Two Methodologies are Being Successfully Integrated in Industry," in *Agile*, Washington, 2008, pp. 63-72.

[8] Desirée Sy, "Adapting Usability Investigations for Agile User-centered Design," *Journal of Usability Studies*, vol. 2, no. 3, pp. 112-132, May 2007.

[9] Melissa Federoff et al., "Extreme usability: adapting research approaches for agile development," , New York, NY, USA, 2008, pp. 2269-2272.

[10] Janice (Ginny) Redish et al., "Usability in Practice: Formative Usability Evaluations - Evolution and Revolution," in *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, Minneapolis, 2002, pp. 885-890.

[11] M. C. Medlock, D. Wixon, M. McGee, and D. Weish, "The Rapid Iterative Test and Evaluation Method: Better Product in Less Time," in *Cost-Justifying Usability: An Update for the Internet Age*. San Francisco, California, United States: Morgan Kaufmann Publishers, 2005.

[12] Jennifer Ferreira, James Noble, and Robert Biddle, "Agile Development Iterations and UI Design," in *Agile Conference*, Washington, DC, USA, 2007, pp. 50-58.

[13] Paul Hodgetts, "Experiences Integrating Sophisticated User Experience Design Practices into Agile Processes," in *Agile Development Conference - ADC*, Washington, DC, USA, 2005, pp. 235-242.

[14] Hugh Beyer, Karen Holtzblatt, and Lisa Baker, "An Agile Customer-Centered Method: Rapid Contextual Design," in *XP/Agile*, 2004, pp. 527-554.

[15] Lynn Miller, "Case Study of Customer Input For a Successful Product," , Washington, DC, USA, 2005, pp. 225-234.

[16] Gerard Meszaros and Janice Aston, "Adding Usability Testing to an Agile Project," in *Agile Conference*, Washington, DC, USA, 2006, pp. 289-294.

[17] Zahid Hussain et al., "Agile User-Centered Design Applied to a Mobile Multimedia Streaming Application," in *USAB*, Berlin, Heidelberg, 2008, pp. 313-330.

[18] Stephanie Chamberlain, Helen Sharp, and Neil Maiden, "Towards a framework for integrating agile development and user-centred design," in *XP*, 2006, pp. 143-153.

[19] L. Nielsen, "Usability Expert's Fear of Agility – An Empirical Study of Global Trends and Emerging Practices," in *7th Nordic Conference on Human-Computer Interaction: Making Sense T, NordiCHI '12*, Copenhagen, 2012, pp. 261-264.

[20] H. Robinson, J. Segal, and H. Sharp, "Ethnographically-informed empirical studies of software practice," *Information and Software Technology*, vol. 49, no. 6, pp. 540-551, February 2007.

[21] Ken Schwaber and Mike Beedle, *Agile Software Development with Scrum.*: Prentice Hall, 2001, vol. 18.

[22] Tomothy C. Lethbridge, Susan Elliot Sim, and Janice Singer, "Studying Software Engineers: Data Collection Techniques for Software Field Studies," *Empirical Softw. Eng.*, pp. 311-341, 2005.

[23] J. Corbin and A. Strauss, *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. London, United Kingdom: SAGE Publications, 2008.

[24] Robert M. Davison, Maris G. Martinsons, and Ned Kock, "Principles of canonical action research," *Information Systems Journal*, vol. 14, pp. 65-86, 2004.

[25] J. Nielsen, *Usability Engineering*. San Francisco, CA, United States: Morgan Kaufmann Publishers Inc., 1993.

[26] Jenny Preece et al., *Human-Computer Interaction*. Harlow: Addison-Wesley, 1994.

[27] J. Nielsen, "Usability engineering at a discount ," in *Third International Conference on Human-computer Interaction on Designing and Using Human-computer Interfaces and Knowledge Based Systems*, New York, 1989, pp. 394–401.

[28] Jonathan Arnowitz, Michael Arent, and Nevin Berger, *Effective Prototyping for Software Makers*. San Fracisco, CA, United States: Morgan Kaufmann Publishers Inc., 2006.