Integrating Agile and User-Centered Design
A Systematic Mapping and Review of Evaluation and Validation Studies of Agile-UX

Gabriela Jurca, Theodore D. Hellmann, and Frank Maurer
Department of Computer Science
University of Calgary
Calgary, Alberta, Canada
{gajurca, tdhellma, frank.maurer}@ucalgary.ca

Abstract—Agile software engineering and user-centered design are two important development processes for ensuring that an application has good user experience. However, integrating these two different processes into a single Agile-UX approach remains difficult. We performed a systematic mapping study to identify relevant research and understand what the field of Agile-UX looks like at present. This mapping discovered that there were only a few evaluation and validation papers published to date, so we performed a review of those papers to better understand the recommendations of these types of papers. Based on this, we are able both to provide a discussion of common trends in these papers that should be of interest to practitioners and to identify gaps in existing literature that indicate strong opportunities for future work.

Keywords—systematic mapping; literature review; agile; user experience; user-centered design; empirical studies

I. INTRODUCTION

The many benefits of Agile Software Engineering have led to it becoming a mainstream development methodology. However, Agile on its own does not necessarily address the usability of the software product. At the same time, the need for a good User Experience (UX) has become more evident, and so efforts have been made to integrate usability practices from User Centered Design (UCD) into Agile.

The goal of UCD is to increase usability – the extent to which a user finds an interface easy to navigate – and the term can also refer to the methods that can be used to improve the design of an application’s interface [1]. Usability is an important factor to consider for any product. For example, when users encounter a difficulty on a website, they are likely to respond by leaving the website [1]. Also, a lack of usability on a company’s internal applications will reduce an employee’s work efficiency. Therefore, many practitioners have, since the beginning of Agile, been motivated to find the best ways to integrate usability practices into products which are developed through Agile [2].

In this paper, we present a systematic mapping and literature review of publications discussing the integration of Agile and UCD (Agile-UX) which should be of interest to both practitioners and researchers. For practitioners, this paper provides an overview of recommendations on Agile-UX and identifies venues where publications on this topic can be found. For researchers, this paper provides a literature review that includes more sources than existing reviews and further identifies gaps in existing research that should be addressed. The research questions this review addresses are:

RQ1. Is the rate of publication increasing over time?
RQ2. What venues are most important for this field?
RQ3. What types of papers are most prevalent?
RQ4. Are the types of studies changing over time?
RQ5. What are the recommendations of existing work?

To answer RQ1-4, we did a systematic mapping of all papers discussing Agile-UX. In order to answer RQ5, we performed a literature review of only evaluation and validation papers identified through RQ3. This was done in order to focus on the more strongly empirical papers within Agile-UX.

II. BACKGROUND

Agile and UCD are quite different processes, which makes it difficult to imagine how they can be integrated. However, Agile and UCD share some common goals, which serve as good starting points for an integration of the two. These differences and commonalities are discussed in this section.

A. Agile Software Engineering

The priority of Agile is customer satisfaction through the frequent delivery to the customer of small sets of working features every iteration [3]. This focus on functionality can come at the cost of usability, as Agile emphasizes minimal upfront design work. However, because Agile emphasizes the delivery of vertical slices of functionality, this means that working features can be delivered to customers in order to get frequent feedback, which is a key concept in UCD [4]. This means that customers have the opportunity to accept features or request changes to their design early on. However, it should be noted that the customer may have a different understanding of what the product should be from users.

Also in Agile, there is no notion of a user interface specialist. Systematic practices from UCD that can improve the usability of the product are not employed. Whenever a product has good usability, it is most likely due to coincidence [5].

B. User-Centered Design / User eXperience

As opposed to working features, the priority of UCD and UX is user satisfaction. Significant resources are allocated to extensive user research at the beginning of the project to support this [3]. This is followed by iterations which consist of
development and evaluation, but the iterations are longer than a typical Agile iteration.

UX design emphasizes specialized methods of end-user research before the product is made [4]. Some of the methods used in user research to gather and understand design requirements are focus groups, usability testing, card sorting, participatory design, questionnaires, heuristics, and interviews. To analyze design requirements, UX makes use of Personas, Scenarios, or Essential Use Cases:

- **Persona**: a fictional character which represents the characteristics of a typical user. Personas make it easier for the UX designer to base their measurements around.
- **Scenario**: a fictional story about the sequence of events that a persona may encounter. The scenario may represent a problem that the persona is likely to encounter in orienting the design process around the thought process that a typical user might have.
- **Essential Use Case**: a detailed description of an interaction that a user might have with the system. These help to break up the problem of design into manageable pieces.

At the end of each cycle, UX designers perform usability evaluations on the design with end-users. This process gives feedback about whether usability goals have been met, and the feedback may be used to adjust the product accordingly.

C. Integrating UX and Agile

Agile and UX share the goal of producing high-quality software even though they approach this goal from different perspectives. Fundamentally, both approaches are human-centered and cyclical.

UCD practices can improve Agile by providing systematic way to examine end-user needs [3]. On the other hand, Agile can improve UCD by providing more frequent iterations, which leads to more frequent usability evaluations. The early feedback can then be incorporated into the product more quickly.

One of the problems of integrating Agile and UX design is the synchronization of their activities. For example, one challenge of the integration is to synchronize usability evaluations from UCD with the unit testing or acceptance testing of Agile developers [2].

Another problem is that the collaboration between UX designers, Agile developers, and other teams (such as marketing) needs to be enhanced through a large amount of communication. Communication between UX designers and developers is very important because each group will have different goals, priorities, and processes [6].

III. RELATED WORK

A previous systematic review conducted by Silva da Silva in 2011 found some interesting trends about the papers in this field [7]. For example, interest in Agile-UX has generally increased over time since the creation of Agile. The increasing interest in Agile-UX has provided support for the need of our systematic review to provide updates and new insights into the field.

Silva da Silva found some other trends of the field related to the content of the studies. For example, the majority of the studies were classified into a Specialist approach, which means that the UX design work for the Agile team was conducted by a specialized UX designer [8]. This approach was more common than the Generalist approach, in which all members of the Agile team participate in design, and the Generalist/Specialist approach, a mix between the two.

Little Design Up Front (LDUF) was the most common practice used in Agile-UX primary studies which Silva da Silva collected [3]. This practice reduces – but does not eliminate – the large amount of design work done through UCD at the beginning of the project so that more effort can be spent on functionality.

The second most common practice was close collaboration between the Agile team and the UX team [3]. This can increase the success of a project by improving understanding of what the project is supposed to be between the two teams. Sohaib [9] recommended this practice after reviewing literature in Agile-UX and additionally suggests that collaboration should also include customers, users, product managers, and business analysts.

It has been recommended that the Agile-UX team should include a Sprint 0 during which the UX team performs initial user research for UX design [3]. During this initial iteration, the Agile team should work on creating user stories. Sohaib has also recommended that user stories be used in the scenario-based design of UCD.

Previous literature reviews seem to agree that iterative development is necessary throughout the project and that multidisciplinary teams should be used because they ensure complete expertise [9] [7].

Other recommendations often suggest that UX design should simply be performed in parallel to Agile development [6]. The problem with this approach is that it undermines collaboration between developers and designers and makes communication between the parties more difficult.

Previous reviews on the integration of Agile and UCD note that there is a lack of sound and controlled studies, as well as a lack of studies in general [9] [10]. Our systematic review addresses this in that we include a larger number of papers in our set and we focus on evaluation and validation papers specifically.

IV. METHODOLOGY

A previous systematic review also categorized the primary studies based on their type, but the authors noted that a limitation was that the classification system used in that paper was not established [3]. Our aim was to use an established classification system to categorize the papers in order to see what trends could be discovered and if they were different from that of previous work, or if they confirmed previous trends. We also performed some additional analyses on the papers than previous literature reviews [3] [9].

A. Research Questions

Our goal was to gather all existing primary studies on Agile-UX and analyze the papers by type, so that we could understand where this field is headed. Our research questions were:

RQ1. Is the rate of publication increasing over time?
RQ2. What venues are most important for this field?
RQ3. What types of papers are most prevalent?
RQ4. Are the types of studies changing over time?
RQ5. What are the recommendations of existing work?
B. Search for Papers

The search for papers was initiated through snowball sampling by one author, beginning from the existing systematic literature review done by Silva da Silva [7]. That is, we assume that the authors of that paper did a proper search for papers, so we built our paper set by looking at what papers were cited by each citation in [7] and so on. Sampling was continued until every citation in the network we were building was already included in our paper set.

In order to identify papers newer than [7], we examined the keywords in the paper set after the previous step and performed an automated search for them using: ACM Digital Library\(^1\), SpringerLink\(^2\), Scopus\(^3\), and IEEE Xplore\(^4\). Since the papers that had already been gathered were from 2011 or earlier, this search was limited to papers which had been published between 2011 and 2014.

The following search string was used to gather papers on the topic of Agile-UX: (agile or xp or scrum) and (usability or user-centered or ucd or user-experience)

C. Categorizing Papers

The papers were read by one author and categorized into the following different categories, according to the classification system defined by Wieringa [11]:

- **Evaluation**: The investigation of a problem which is in practice. The investigation should support a conclusion about a new casual or logical relationships. Examples of evaluation studies are: case studies, field studies, field experiments, surveys, logic, or math.
- **Validation**: The investigation of a solution proposal which is not in practice yet. The solution may have been proposed by the author or it may come from a different source. Examples of validation studies are: experiments, simulation, and prototyping.
- **Solution**: A solution technique is proposed and its relevance is argued for. The technique should be novel, but not yet validated. A small proof-of-concept may be offered.
- **Experience**: The author’s anecdotal experience is presented in order to express what has been done in a project. These papers usually contain a list of lessons learned. The most common authors for this type of paper are practitioners.
- **Philosophical**: A new conceptual framework is presented, which should be sound, original, and insightful.
- **Opinion**: The author’s opinion is presented about how something should be done. The opinion should ideally provoke a discussion.

This classification system was used because the categories were easy to interpret for papers about Agile-UX. The Evaluation and Validation papers were further examined in more detail in order to summarize the recommendations that had been given for Agile-UX. We chose to analyze these paper types in more detail because they are generally more rigorous than other paper types.

From the Solution and Validation paper set, the papers were also categorized by the type of research which was conducted by the authors. The categories for the types of research were:

- Case study
- Interview
- Action Research
- Question-based Survey
- Focus Group

V. Analysis

To answer RQ1-4, we performed several high-level analyses on all of the papers in our paper set. This section provides a series of visualizations in order to answer these questions.

A. Is the rate of publication increasing over time?

The first of the 75 papers in our overall paper set to be published appeared in 2002 [12]. This is interesting given that the field of agile software engineering wasn’t described until the creation of the Manifesto for Agile Software Development in 2001. Immediately after the definition of agile, people were already working on figuring out how to integrate UX into agile, which underscores the significance of work in this field.

The number of publications per year in this field increased significantly from 2007 onwards, with no year since then having fewer than 6 publications. 2008 and 2009 saw the highest number of publications, with over half of those (8 of 12 and 7 of 13, respectively) being experience reports. It’s possible that the increase in publications in this field may be due to an increase in interest in this kind of practitioner-reported view of how agile and UX can be integrated in practice.

After a decrease in publications from this peak between 2010 and 2012, the rate of publication recovered significantly in 2013, as can be seen in Figure 1.

B. What venues are most important for this field?

As can be seen in Figure 2, more publications have come from the Agile Conference (20) than from any other venue – in fact, there are almost double the number of publications at that conference as there are at the next most frequently-occurring venue (CHI, with 10). Taken together, all venues with more than two publications have 44 publications overall – over half of the total. This is somewhat concerning as it could imply that research in this field is only being made known to a narrow group of researchers. On the other hand, research published in some of conferences under the “Other” heading in Figure 2 run the opposite risk – their research is not reaching the community of researchers interested in these subjects. In the

---

\(^1\) dl.acm.org
\(^2\) link.springer.com
\(^3\) www.scopus.com
\(^4\) ieeexplore.ieee.org

**Figure 1**: Number of publications on Agile-UX per year.
future, researchers should take care to balance these concerns when selecting venues to submit their work to.

C. What types of papers are most prevalent?

We also looked into the types of papers published in order to give an indication of the general maturity of the field of Agile-UX and to better understand what sorts of research are necessary to drive the field forward in the future. Additionally, in order to make the current paper of higher value to future researchers, we provide direct citations based on paper type in TABLE 1.

Out of the 75 papers included in our paperset, 28 – over a third – are experience reports, and only 22 are evaluation or validation papers. This confirms the work of previous publications [7], who note that there are few rigorously-conducted studies in Agile-UX. While experience reports, opinion papers, philosophical papers, and solution proposals may be quite relevant to problems in practice, they are hard to classify in terms of rigor, making it difficult to draw clear conclusions from them. This is the reason the Results section of this paper focuses on the remaining two paper types. It is also an indication that the most important thing researchers can do to drive Agile-UX forward is to conduct more validation and evaluation studies.

D. Are the types of studies changing over time?

Specifically for the evaluation and validation studies we identified, we additionally looked into the types of empirical studies used to investigate Agile-UX. It’s important to note here that a single study may report results of more than one study. 28 studies were identified across the 22 evaluation and validation papers we identified, with over half (16) of these coming from the last two years we studied. This can be interpreted as a strongly positive sign that the field is moving in the right direction – towards more empirical work. In future work, it would be useful for more controlled experiments to be conducted, as very few of these have been done to date, but it will likely be difficult to design these experiments while ensuring that the setup is relevant to real-world conditions.

In Table 2, it is clear that the variation of types of studies has increased over time in this paper set. Before 2013, at least half of the different types of studies listed were not all performed. As time increases, the researchers also seem to use an increasing amount of different types of studies within a single paper. Using a mixed method approach is an attractive way to approach qualitative research, because it focuses the attention on the research problem [13].

More research is still needed which uses focus groups or action research to derive its results. The availability of different kinds of research results may provide a more rounded view of the research problem and its solution [13]. Creswell recommends mixed methods for the study of social sciences in order to derive knowledge from the situation, and the study of Agile-UX has some components which are very much based in social science [13]. As will be discussed in the next sections, the social interactions between Agile-UX team members have a large effect on the success of the integration.

VI. RESULTS

In this section, we discuss the major trends discovered in the 22 experience and validation papers in our paperset.

A. Comparison to Related Work

One paper stated that the previous systematic study had established that practices like LDUF, Sprint 0, and One Sprint Ahead have already been accepted as necessary for Agile-UX integration [14]. For example, Raison stated that LDUF had already been accepted for Agile-UX because 31 out of 58 studies had recommended LDUF in their systematic review.

However, our systematic review could not conclude that LDUF, One Sprint Ahead, and Sprint 0 were highly used in practice or recommended, as determined by validation or evaluation studies.

It was found that 11 of the 22 studies did not comment on LDUF, four studies stated that LDUF was used to some degree by the companies that they studied, and three studies found that LDUF had not been successfully implemented in the companies they studied.

The studies which stated that LDUF had not been implemented in the companies they studied also emphasized the high-degree of organizational failures related to Agile-UX.
integration which were present in the companies. In these companies, the intention was to adopt LDUF, but this could not be achieved for various contextual reasons.

For example, LDUF and Sprint 0 could not be implemented because the UX designers were too overworked with too many projects in order to have time to complete an initial design for the developers on time, or to participate in Sprint 0 [15] [16] [17]. The UX designers could not stay one sprint ahead, and were usually at the same sprint as the developers or sometimes even one sprint behind.

It seems that organizational problems may act as a significant hindrance to Agile-UX implementation in industry. These problems also hinder academic researchers from studying the effectiveness of Agile-UX methodologies that require careful coordination in industry.

The reason that Silva da Silva [15] may have found greater support for LDUF, One Sprint Ahead, and Sprint 0 could be that they examined papers from all categories, not just validations and evaluations. Since we found that most of the papers in Agile-UX are experience reports, it might be the case that the industry practitioners who have contributed experience were more likely to have had a positive experience with implementing Agile-UX and have therefore successfully implemented practices like LDUF, One Sprint Ahead, and Sprint 0.

B. Organizational Pitfalls of Agile-UX Integration

I will discuss some of the organizational problems which can hinder Agile-UX integration. Practitioners may find this information useful for when they try to implement Agile-UX in the industry, or academic researchers can use this information for selecting appropriate research subjects in order to effectively study Agile-UX methodologies in action.

One issue which was common between most validation and evaluation studies, was the power struggle between UX designers and developers. As Raison et al. [14] observed in their case studies, UCD was often perceived as optional or peripheral to the development work. For example, numerous organizations did not consider UX designers as a full-time member of the Agile team [3].

Furthermore, there are often not enough UX designers involved in the Agile projects, as da Silva [3] commented. As an extension, the UX designer has to work on more than one project at a time. This can be a problem, because UX designers often have to perform multiple roles on a project, such as interaction design, or UI development, performing such tasks as: user research, market research, user-centered design, prototyping, usability inspection, user testing, visual design, feedback, and coding [3]. In some cases, the integration with Agile has led to cancellation of the software product, because the UX designers have not been able to stay one sprint ahead [17].

Another problem occurs when developers, UX designers, and stakeholders lack a UX vision for the product, the project is slowed down [18]. In this case, decision-making becomes very difficult. Kollmann [18] has found that Sprint 0 does not compensate for a lack of product vision, because the “Big Picture” of the product should already be known at that point.

C. Organizational Recommendations

1) General: Although there are pitfalls, there is overwhelming evidence which supports that Agile and UX design are a strong fit for each other. The majority of literature has found that a successful integration will be beneficial to the business and user. There are a number of factors and techniques that can be considered to increase the chances of integration success.

One recommendation is to make the UX designers full time members of the Agile team, so that they can focus on accomplishing all of their assigned tasks which may be very numerous, as mentioned earlier [3]. This would also improve the UX vision for the product, since the UX designers would not be fragmented among different projects [18]. Kollmann has suggested that a beneficial approach may be to have satellite UX designers which permanently work alongside the developer team.

Another factor which improves collaboration is to have a shared workspace between UX designers and developers. The collaborative culture will encourage the team to see UCD as an equal part of the project. In fact, close collaboration between the whole team, including product managers, UX designers, and developers is very helpful against encountering conflicts later on [17]. Chamberlain [16] suggests that some balancing mechanism needs to be put in place, so that each discipline can equally contribute to the team.

However, perhaps the most essential aspect of the integration is also the contextual value that the team is embedded in, and may be more important to consider than the practices themselves [19]. For example, Cajander observed that Usability Specialists need to be given more organizational support in order to make a difference regarding usability, which they are responsible for [20]. As Raison sees it, a significant cultural change, ongoing sponsorship, and a holistic approach needs to be applied in order to ensure that the success of the integration is long-term.

2) XP-focused recommendations: Although XP is a branch of Agile which is characterized very close and frequent collaboration with customers, this does not guarantee good usability of the product [5]. However, good usability can be

---

**TABLE 2: TYPES OF STUDIES REPORTED IN EVALUATION AND VALIDATION PAPERS ON AGILE-UX.**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Interview</td>
<td>[63]</td>
<td>[10]</td>
<td>[8]</td>
<td>[23]</td>
<td>[18]</td>
<td>[21]</td>
<td>[17]</td>
<td>[3]</td>
<td>[20]</td>
<td>9</td>
</tr>
<tr>
<td>Case Study</td>
<td>[16]</td>
<td>[22]</td>
<td>[19]</td>
<td>[46]</td>
<td>[17]</td>
<td>[14]</td>
<td>[3]</td>
<td>[7]</td>
<td>[8]</td>
<td>8</td>
</tr>
<tr>
<td>Survey</td>
<td></td>
<td>[17]</td>
<td>[25]</td>
<td>[27]</td>
<td>[79]</td>
<td>[4]</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>Experiment</td>
<td>[5]</td>
<td></td>
<td>[81]</td>
<td></td>
<td>[24]</td>
<td>[79]</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>Action Research</td>
<td>[26]</td>
<td></td>
<td></td>
<td>[27]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>[3]</td>
<td>1</td>
</tr>
<tr>
<td>Focus Group</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Total (Unique)</strong></td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td>11</td>
<td>7</td>
</tr>
</tbody>
</table>
ensured by adding systematic usability practices to the XP process. It is known that XP gathers customer requirements continuously, so therefore discount usability is a good fit due to its low cost and fast implementation [21]. Discount usability includes techniques like Scenarios, Simplified thinking aloud, Heuristic evaluation and Card sorting.

D. Artifacts and Practices

Other recommendations stem from Brown’s [22] finding that artefacts such as sketches, lists, and stories are fundamental to the collaboration between Agile developers and UX designers. It appears that most artifacts and practices have a great need to be light-weight in Agile-UX, so that the practices can adapted to the schedules of all participants, and the artifacts are easily accessible to everybody involved. In this section, I will list some of the artifacts and practices which were suggested by various authors.

For example, Brown observed that an important artefact of design meetings was the white board, since it was the focus of everybody of all disciplines in the meeting room. Brown has recommended that a shared repository of whiteboard contents would aid the interactions which follow the meeting.

Another artefact that has been perceived to facilitate team communication is a Concept Map [23]. A Concept Map depicts relationships between user stories and scenarios, and can be used to mitigate team conflict. This approach is advantageous, because it is a light-weight approach that can fit nicely within the Agile-UX iterations.

As has been mentioned, the collaboration of team members of different disciplines has been shown to be vital to Agile-UX. The Informal Cognitive Walkthrough (ICW) is a two-step user experience research methodology which has been adapted to be more light weight so that it can be applied in an Agile environment and between the different disciplines [24]. According to Grigoreanu, ICW is based off of Cognitive Walkthroughs, but it only asks two straightforward questions at each step, which makes it easier to accommodate to different people’s schedules. ICW is first performed with UX designers and developers, and then it is done with representative customers in order to get product feedback.

Some of the most highly used practices and artifacts used in the Agile-UX industry were found to be Workshops, Lo-Fi Prototyping, Interviews, Meetings with Users and Scenarios [25]. These are not necessarily the most recommended, but this list gives an impression of which artifacts and practices are most feasible in the industry.

E. Integration Frameworks

Lee [26] proposes the XSBD framework, which helps to integrate usability engineers into an agile team. It is drawn from Scrum, XP, and Scenario–Based Design. This framework helps to synchronize the usability and development activities, and also helps the usability engineer perform their usability evaluations. The framework was shown as useful, because it helps to improve the accuracy of gathered customer requirements. One of the key artifacts of this framework is the Central Design Record, which holds the prioritized goals which helps drive design.

Abdallah [27] has built upon XSBD to propose QXSBD, which has the following key addition: explicit metrics to measure the usability of the software. This process involves a one-day workshop to create the scenarios, instead of an iteration long process like in XSBD.

VII. LIMITATIONS

The most obvious limitation of this work is the fact that we used snowball sampling in order to collect all papers from before 2011. We assume that the study on which we base the present work found all relevant papers; however, if that study missed any papers, this study would also miss those same papers.

The second potential weakness with this study is that the majority of the search for papers and interpretation of results was done by the first author. However, to guard against researcher bias, the results were discussed frequently with the second author.

Next, we included workshop summaries, extended abstracts, and the like in our paper set. A common practice of systematic studies is to consider only full conference papers and journal articles based on the assumption that these sources are of higher quality than other types of papers. However, due to the low number of evaluation and validation papers in the field of Agile-UX, we chose to include these additional papers in order to be able to consider results from more studies overall.

Finally, a major consideration about the present work is that we excluded experience reports from our review. This was done so that we could focus on more rigorous studies. However, many experience reports on Agile-UX explain how companies have been able to successfully integrate Agile and UCD. By excluding these studies, our paper may present a more negative view of the field than if they had been included.

VIII. FUTURE WORK

One major direction for future work would be to extend this paper to consider papers based not on their type or on type of study that was conducted but based on the rigorousness of the research methodology and its relevance to practice. Recent research on this topic [28] has shown that, when these factors are taken into account, the results of a systematic mapping can be changed dramatically. For example, past systematic reviews of test-driven development have indicated that the effects of this practice are unclear; however, when considering only rigorous, relevant studies, the effect of test-driven development becomes overwhelmingly positive. This study was the inspiration for our consideration of only evaluation and validation papers in the present work.

Additionally, due to the high number of experience reports collected in this study, it would be useful in future work to find a way of performing a review of these results also and perhaps integrating those findings with the ones presented here. However, it’s difficult to understand how to evaluate these papers given that there is no formal experimental setup. This makes it difficult to understand whether results that worked for one company would be generalizable to other organizations.

IX. CONCLUSIONS

One of the major recommendations that this study was that researchers need to consider the organizational context of the industry subjects they are studying. This is because Agile-UX methodologies may not be implemented as expected in organizations where there is a lack of support for the integration
of Agile and UX. For example, in order to provide support for LDUF, Sprint 0, and One Sprint Ahead in practice, these practices need to be given a fair shot at being implemented correctly.

Industry practitioners may also want to consider what type of support they offer the Agile and UX team before they implement the suggested practices from literature. The studies of this review have confirmed that there is a wide variety of how well the practices and artifacts which were described in literature can be applied. One of the major problems which was identified with Agile-UX integration among different studies was that the UX-designer was over worked and overly distributed among the Agile teams. This also contributed to a lack of the “big picture,” which was identified as very important to the success of integration.

Some of the recommended practices and artifacts that were identified from evaluation and validation studies were: concept maps, cognitive walkthrough variants, workshops, lo-fi prototypes, interviews, scenarios, and meetings with users. For academics, this study suggests two major ways to advance the field of Agile-UX research. First, academics should endeavor to conduct more evaluation and validation studies, in partnership with practitioners, to evaluate how well existing approaches to Agile-UX are working. Specifically, we note that there are very few papers reporting the results of focus group, action research, experiment, and survey studies.

Second, this study demonstrates that a significant amount of existing literature on Agile-UX comes from practitioner-created experience reports. These studies are difficult to compare to more rigorous work, like evaluation papers, making it difficult to use systematic mapping and review approaches to understand what the state of the field as a whole is. The investigation of new techniques for investigating existing literature in this field and others represents a large, unaddressed area for future work.

REFERENCES


