A Systematic Literature Review for Agile Development Processes and User Centred Design Integration

Dina Salah
Department of Computer Science
University of York
York,UK
dm560@york.ac.uk

Richard F. Paige
Department of Computer Science
University of York
York,UK
richard.paige@york.ac.uk

Paul Cairns
Department of Computer Science
University of York
York,UK
paul.cairns@york.ac.uk

ABSTRACT

Agile development processes and User Centred Design (UCD) integration have been gaining increased interest, in part due to the complementarity of the techniques, the benefits each can apply to the other, and the challenges associated with their combination. This paper describes a Systematic Literature Review (SLR) that was conducted on Agile and UCD integration. The aim of this SLR was to identify various challenging factors that restrict Agile and User Centred Design Integration (AUCDI) and explore the proposed practices to deal with them. The study included a total of 71 papers and excluded 80 papers published from the year 2000 till 2012. AUCDI challenges and their respective proposed practices and success factors were synthesized. A description and taxonomy of AUCDI challenges and its respective success factors and practices were reported. Practitioners can utilise the results in identifying potential AUCDI challenges and practices or success factors to deal with them.

Categories and Subject Descriptors

D.2.10 [**Software Engineering**]: Design

General Terms

Design, Human Factors

Keywords

Agile Software Development Processes; User Centred Design; Agile User Centred Design Integration

1. INTRODUCTION

Agile methods are lightweight software development methods that tackle perceived limitations of plan-driven methods via a compromise between absence of a process and excessive process [29]. Agile processes aim to deal with volatile requirements via discarding upfront, precisely defined plans. They are iterative and are used to develop software incrementally. Different Agile processes implement these ideas in different ways. All Agile processes share common values and principles, defined in the *Agile Manifesto* [5].

User Experience (UX) is defined as the perceptions and responses of users that result from their experience of using a product [31]. User Centred Design is a set of techniques, methods, procedures and processes as well as a philosophy that places the user at the centre of the development process [32, 19]. The goal of applying UCD is to attempt to satisfy users via producing usable and understandable products that meet their needs and interests [19].

Agile and User Centred Design Integration (AUCDI) gained increased interest due to three reasons: first, the reported advantages of UCD on the developed software as it enables developers to understand the needs of the potential users of their software, and how their goals and activities can be best supported by the software thus leading to improved usability and user satisfaction. Second, the Agile community hardly discusses users or User Interfaces (UIs), thus implying either a negligence of UX or focus on less sophisticated UX projects [4]. Moreover, none of the major Agile processes explicitly include guidance for how to develop usable software [49]. In addition, the interaction design role, usability, and UI design in an Agile team is unclear and largely overlooked [15, 6]. Furthermore, principles and practices for understanding and eliciting usability and user requirements and evaluating Agile systems for usability and UX are generally considerably deficient [49, 44, 75]. Third, there exists philosophical and principled differences between Agile methods and UCD in focus, evaluation method, culture and documentation that suggest that their integration will be fundamentally challenging.

This paper provides details of a Systematic Literature Review that was conducted on Agile and UCD integration. This SLR identified various challenging factors that restrict Agile and User Centred Design Integration and explored the proposed practices to deal with them.

The rest of this paper is structured as follows: section 2 discusses the SLR research method. Section 3, discusses the quantitative classification results. Section 4 presents the results of the research questions in regards to AUCDI challenges, practices and success factors. Section 5 discusses the conclusion and future work.

EASE'14 May 13-14 2014, London, England, BC, United Kingdom

http://dx.doi.org/10.1145/2601248.2601276 ...$15.00.

2. SLR PROTOCOL DEVELOPMENT

This SLR had three objectives: to identify AUCDI challenges, to identify AUCDI success factors and practices and to infer relevance of AUCDI success factors and practices to AUCDI challenges.

2.1 Related Work

The AUCDI literature contains only two literature reviews that have been reported so far. The first is a literature review that discusses methods for integrating usability engineering practices into the Agile software development process and identifies the tensions between Agile methods and usability engineering [75]. The second is a SLR that revealed the existence of a common process model for integration and discussed the supporting artifacts for the collaboration between designers and developers [17]. Thus there is an absence of a SLR that provides a comprehensive scrutiny of AUCDI challenges and investigates the success factors and practices that tackle these challenges. The results of this analysis can be used by organisations that aim to achieve the integration to understand the potential integration challenges and the available practices that can be utilised to tackle these challenges.

2.2 Specifying the Research Question(s)

This study aims to address the following questions whose answer can play a significant role in formalizing and structuring AUCDI efforts.

- What are the challenges that could develop during AUCDI adoption process?
- What are the potential success factors for AUCDI?
- What are the potential practices for AUCDI?

2.3 Search Process

This section details the process followed to search for literature and discusses the search resources, search keywords, inclusion and exclusion criteria, data extraction strategy and data synthesis method.

2.3.1 Search Resources

The search included electronic sources, conference proceedings, journal articles and magazines. The search string focused on combining both UCD and Agile keywords and was modified in accordance to the specific search requirements of the different electronic libraries.

Electronic Sources/Digital Libraries: The electronic sources/digital libraries chosen to conduct the search on were: The ACM Digital Library, IEEExplore Digital Library, Google Scholar, Springer, Wiley InterScience, and Citeseer Library.

Conference Proceedings: A number of conferences proceedings were manually searched for research papers and experience reports on the topic. Those conferences included: Agile Conference, XP Conference, XP/Agile Universe, International Conference on Software Engineering (ICSE), Human Factors in Computing Systems Conference (CHI), International Symposium on Empirical Software Engineering and Measurement(ESEM), British HCI, NordicHCI, INTERACT, and European Conference on Cognitive Ergonomics.

Journals: A number of journals were manually searched including: Empirical Software Engineering, Software Practice and Experience, International Journal of Human-Computer Studies, International Journal of Human-Computer Interaction, Behavior and Information Technology, Information and Software Technology, IEEE Transactions on Software Engineering, ACM Transactions on CHI, Human Computer Interaction Journal, and Interacting with Computers.

Magazines: Three magazines were searched; IEEE Software, Communications of the ACM and Interactions. Moreover, the references of primary studies were checked for any relevant studies irrespective of the forum of publication.

2.3.2 Search Keywords

We used the research questions in order to identify the search keywords. Table 1 lists the keywords utilised.

Category	Keywords
UCD	Usability User Experience User Centred Design User Interface User Interaction Usability Engineering Human-Centred
Agile	Agile Method Agile Development Agile Practice Agile Project Scrum Extreme Programming

Table 1: Keywords for Systematic Literature Review Process

2.3.3 Study Selection Criteria

The following section discusses the criteria that were used to assess each paper and decide on whether to include or exclude primary studies.

Inclusion Criteria

To decide on paper inclusion, the following features must exist on the paper

- Peer reviewed to ensure quality of primary study.
- Available on line to ensure paper accessibility.
- Focused on the integration of UCD and Agile to ensure its relevance.
- Focused on Scrum or Extreme Programming or Agile processes in general.
- Not a workshop, panel, tutorial, seminar, interview or poster session to ensure enough details are included in the paper in order to assess its quality and use the paper in answering research questions.
- Published between the year 2000 and 2012.
- Written in English.
- Non redundant since the main focus is to study AUCDI challenges, practices and success factors. All papers written by the same authors that narrate the same AUCDI practices and success factors were excluded.

Exclusion Criteria

Any paper that does not possess any of the inclusion criteria was excluded. The remaining papers were read fully.

A list of included and excluded papers was kept. Results bias was avoided by excluding multiple publications of similar research. This led in some cases to contacting authors directly to verify the most complete and recent publication. The SLR protocol was evaluated via the second and third author to check the internal consistency of the protocol in order to confirm that the research questions derive the search strings, the data extraction forms allow answering the research question(s), and the data analysis procedure is adequate to address the research questions.

2.3.4 Data Extraction and Data Synthesis Method

Data extraction forms were designed to ensure that sufficient and appropriate data is collected to address both the research questions and the quality criteria. Data extraction consistency was checked via two methods: first, data extraction was carried by the first author via randomly selecting a sample of primary studies and subjecting them to data extraction by second and third authors. The results were cross checked and any disagreements were discussed and resolved in meetings. Second, the first author conducted a test-retest process where primary studies were randomly selected and a second extraction was performed to check data extraction consistency.

The data was synthesized via thematic analysis using the iterative thematic synthesis process recommended in [16].

3. RESULTS

This section discusses both the quantitative classification results of the SLR. It starts with an overview of search sources. Then it provides an overview of the studies and discusses the excluded papers and exclusion reasons. Then it discusses the results of classification of the studies.

3.1 Search Sources Overview

A number of digital libraries were searched including: ACM Digital Library, IEEExplore Digital Library, Google Scholar, Springer, Wiley InterScience, and Citeseer Library. The date of the search was between the period of April to September 2012 and covered the years between 2000-2012 since papers on Agile development processes started around the year 2000 whereas papers regarding the integration between Agile and UCD started shortly after that. Table 2 lists the details of the manually searched journals including the start date, start volume number, the end date, and the end volume number searched.

Table 3 lists the details of the conference proceedings that were manually searched including the start and end years searched. Ten conference proceedings were manually searched. The earliest year for included conference proceedings is 2000 and the latest is 2012. Although the aim was to search and include all conferences from the year 2000 till the year 2012, however, some conferences have not started at the year 2000. For example, ESEM started at 2007. In addition, at the time of conducting the search some of the proceedings for the year 2012 were not available on line. Some conferences were Biennial, for example, NordicHCI in even years and INTERACT run in odd years.

3.2 Excluded Papers

The final number of papers that were included for data analysis was 71, and a total of 80 papers were excluded. Paper exclusion was caused by a number of reasons related to format (tutorial, workshop, interview, panel, seminar), lack of peer review, lack of AUCDI focus, lack of focus on XP or Scrum, unavailability on-line, time constraints in case of AUCDI PhD studies, redundancy and lack of quality. A number of papers were excluded due to quality, however quality criteria were not covered in this paper due to space limitations.

Name of Proceedings	Start Search Year	End Search Year
Agile Conference	2003	2011
XP	2003	2012
XP-Agile Universe	2002	2004
International Conference on Software Engineering (ICSE)	2000	2011
Human Factors in Computing Systems Conference (CHI)	2000	2011
International Symposium on Empirical Software Engineering and Measurement(ESEM)	2007	2011
British HCI	2000	2010
NordicHCI	2002	2010
INTERACT	2001	2011
European Conference on Cognitive Ergonomics	2006	2011

Table 3: Conference Proceedings Searched

3.3 Studies Classification

Table 4 shows the results of the second stage of paper classification after taking into consideration the criteria for inclusion and exclusion. A total of 80 papers were excluded for various reasons.

Exclusion Reason	No.	%
Paper Format	16	20%
Lack of Peer Review	14	17.5%
Lack of Focus on AUCDI	17	21.25 %
Lack of Focus on XP or Scrum	2	2.5%
Unavailability On line	4	5 %
Lack of Time	2	2.5 %
Redundant Content	18	22.5%
Lack of Quality	6	7.5%
Total	80	100%

Table 4: Classification Of Excluded Papers

Studies by Year of Publication: Table 5 shows the classification of the different studies according to the publication year; 2007 and 2008 had the largest number of papers. In 2007 there was 11 papers whereas there was 13 papers in 2008.

Studies by Journal: Table 6 shows the list of different journals that included papers on AUCDI and shows that there is a scarcity in publications in journals and all journals included only one occurrence of publication on AUCDI related topics.

Studies by Conference: Table 7 shows the list of different conferences that included papers on AUCDI.

Studies by Magazine: Articles were found in two of the three searched magazines. Interactions included three articles [54, 19, 4] and Communications of the ACM included one article [14].

Studies by Book Chapter: Two book chapters were included [3, 7]

Studies by Masters/PhD: Two Masters theses [69, 45] were included. However, to the best of our knowledge there

Name of Journal	Start Date and volume Number for Years Searched	End Date and Volume Number for Years Searched
Empirical Software Engineering	Volume 5, Number 1, March 2000	Volume 17, Issue 4-5, August 2012
Software Practice and Experience	Volume 30, Issue 1, January 2000	Volume 42, Issue 7, July 2012
International Journal of Human-Computer Studies	Volume 52, Issue 1, January 2000	Volume 70, Issue 9, September 2012
International Journal of Human-Computer Interaction	Volume 12, Issue 1, January 2000	Volume 28, Issue 7, July 2012
Behavior and Information Technology	Volume 19, Issue 1, January 2000	Volume 31, Issue 6, June 2012
Information and Software Technology	Volume 42, Issue 1, January 2000	Volume 54, Issue 9, September 2012
IEEE Transactions on Software Engineering	Volume 26 , Issue 1, January 2000	Volume 38, Issue 3, March 2012
ACM Transactions on CHI	Volume 7 Issue 1, March 2000	Volume 19 Issue 1, March 2012
Human Computer Interaction Journal	Volume 15, Issue 1, 2000	Volume 27, Issue 1-2, April 2012
Interacting with Computers	Volume 12, Issue 3, January 2000	Volume 24, Issue 2, March 2012

Table 2: Manually Searched Journals

Year	Papers	No.	%
2000	0	0	0%
2001	[15]	1	1.4%
2002	[65, 70]	2	2.8%
2003	[44, 36]	2	2.8%
2004	[43, 12, 4, 6, 73]	5	7%
2005	[33, 54, 59, 34, 7]	5	7%
2006	[13, 58, 51, 55]	4	5.6%
2007	[81, 20, 19, 25, 24, 23, 64, 76, 48, 56, 57]	11	15.5%
2008	[8, 74, 62, 77, 22, 30, 80, 50, 45, 63, 52, 3, 9]	13	18.3 %
2009	[1, 40, 21, 11, 18, 67, 42, 49, 72]	9	12.7 %
2010	[2, 26, 78, 46, 75, 35]	6	8.5 %
2011	[14, 69, 17, 53, 27, 47, 37, 10, 71]	9	12.7%
2012	[41, 38, 61, 28]	4	5.6%
Total	—	71	100%

Table 5: Studies by Year of Publication

Name of Journal	No.	%	References
Journal of Systems and Software	1	14.3%	[41]
IFIP Advances in Information and Communication Technology	1	14.3%	[46]
Information Age	1	14.3%	[15]
Journal of Usability Studies	1	14.3%	[76]
Software Practice and Experience	1	14.3%	[27]
The Code 4 Lib Journal	1	14.3%	[52]
Cutter IT Journal	1	14.3%	[36]
Total	7	100%	–

Table 6: Studies by Journal

Name	No.	References
Agile Conference	19	[9, 62, 44, 33, 22, 30, 58, 74, 81, 8, 23, 50, 17, 59, 42, 28, 48, 49, 10]
CHI	6	[77, 11, 80, 51, 63, 47]
XP	6	[13, 25, 2, 26, 78, 61]
HCI	3	[20, 56, 73]
XP/ Agile Universe	1	[6]
ICSE	1	[71]
GI Jahrestagung	1	[12]
OOPSLA	1	[65]
International Conference on Software Engineering Advances	1	[53]
The International Conference on Contemporary Ergonomics	1	[55]
International Conference on Advances in Computer-Human Interactions	1	[38]
International Conference on Computer Design and Applications (ICCDA)	1	[75]
Symposium on Human Interface on Conference Universal Access in Human-Computer Interaction	1	[21]
New Zealand Computer Science Research Student Conference	1	[24]
Product Focused Software Process Improvement	1	[43]
Symposium of the Workgroup Human-Computer Interaction and Usability Engineering	2	[40, 39]
HCII 2009	1	[1]
International Symposium on Intelligent Information Technology Application	1	[67]
Australian Conference on Information Systems	1	[64]
International Computer Software and Applications Conference	1	[34]
International Conference on Human-Centred Software Engineering	1	[37]
Participatory Design Conference	1	[70]
International Conference on Universal Access in Human Computer Interaction	1	[57]
International Conference on Information Technology Interfaces	1	[66]
Irish HCI	1	[72]
Total	56	–

Table 7: Studies by Conference

is also four PhD theses that are focused on AUCDI. All of which were not included due to time constraints or lack of on line availability but papers related to these PhDs were included in the SLR.

Studies by Publication Channel and Occurrence: Table 8 shows the total number of papers published via different publication channels. The table shows that conference papers represent the majority of publications.

Publication Channel	No.	%
Conference	56	79%
Journal	7	10%
Magazine	4	6%
Masters Theses	2	3%
Book Chapter	2	3%
Total	71	100%

Table 8: Studies by Publication Channel and Occurrence

4. AUCDI CHALLENGES AND PRACTICES

This section is focused on reporting the results of the SLR research questions. It reports on AUCDI challenges and the practices that tackle each challenge.

4.1 Lack of Time for Upfront Activities

Agile Methods discourages upfront planning activities since it strives to remain responsive to changing requirements [24, 49]. Moreover, Agile approaches focus on frequently producing deliverable solely in terms of functionality [51, 74]. This has resulted in lack of allocated time for software design planning activities [21, 45], performing user research to discover the problems, work practices and work flows of end users [13, 45, 21, 79, 19] and sketching out a coherent design [45, 21, 79, 51]. Moreover, incremental Agile development is translated into sliced or "feature by feature" development for design, which can result in a UI that is disjoint, piecemeal and lacking a holistic, coherent, and overall structure and vision [55, 63, 2, 49, 51, 74, 60, 54, 58, 4, 45].

Practices and Success Factors: Lack of allocated time for upfront activities was addressed via upfront design. Upfront design is a separate pre-development period that is used in Agile projects for eliciting requirements, understanding users, user goals and context of use and conducting UX design up front and ahead of developers in order to achieve a comprehensive system view [13, 40, 54, 30, 42, 41, 62, 33, 19, 76, 81, 45, 59, 54, 30, 11]. Upfront design can also be used by the development and quality assurance teams to work on back end features such as selecting development environment and system platforms [62] or features with low design cost and high development cost [59]. Upfront design is also referred to as Iteration 0. Upfront design has a positive effect in mitigating poor design judgments, poor task prioritization, costly redesign problems, usability problems and inaccurate work estimates [25].

4.2 Difficulty of Modularization/ Chunking

Design chunking is breaking design into cycle sized pieces called design chunks that incrementally add elements to the overall design and design goals [76]. The incremental nature of Agile processes makes design chunking more critical and challenging [22, 76, 59]. This is due to a number of reasons: first, difficulty in determining the right chunk size and the right amount of interaction design work per iteration [78]. Second, difficulty of maintaining the ordering dependency between design chunks [76]. Third, difficulty in differentiating between user experience design activities that contribute to breadth or depth [33]. Fourth, interaction designers adopt a holistic view to interaction design and as a result it can be difficult for them to grasp and adopt design chunking both as an idea and as a work procedure [76].

Practices and Success Factors: Design chunking was addressed via having well defined design goals [76], using one release to chunk large or complex features [62, 21], chunking design into features [62], time boxing highly creative UX design activities [33] and postponing depth based UX activities to occur later in the development life cycle in order to develop both the functional feature and its related UX design activities in the same iteration [33].

4.3 Difficulty of Prioritizing UCD Activities

Inclusion and prioritization of usability or UX related activities into the different iterations or sprints was reported to be challenging due to developers' focus on accomplishing functionality features rather than usability or UX features [74, 60]. Moreover, although usability tasks could be included on a backlog yet they usually do not get prioritised to be included in the current sprint [74].

Practices and Success Factors: This challenge was handled via assigning this responsibility to the designer or UCD practitioner [51, 60], having a separate UX product backlog [11], Uscrum [74], and assigning a separate UX scrum team to ensure the prioritization of UCD activities [11].

4.4 Optimizing the Work Dynamics Between Developers and UCD Practitioners

Agile development process changed the relationship between developers and UI designers [23] since the UI design became a team effort and required the UI designer to be on call to participate in ad hoc discussions [54]. Moreover, the highly compressed time scales and reliance on team self governance of Agile development processes required more active involvement from UX managers to ensure the regular inclusion of UX activities in team based planning and scheduling [19]. However, the Agile principle "Working software is the primary measure of progress" can pose a challenge on the integration since it can introduce competing goals between developers and UCD practitioners particularly when they work in parallel. Moreover, the Agile principle "Simplicity–the art of maximizing the amount of work not done–is essential" can represent an additional challenge to AUCDI efforts since simplicity in the UI does not always align with simplicity in the implementation [49]. Ongoing and continuous communication need to be maintained between developers and UCD practitioners to avoid the occurrence of delays and bottle necks in the development process [24].

A number of practices were utilised in order to optimize the work dynamics between developers and UCD practitioners. These practices will be discussed in the following subsections

4.4.1 Sharing an Understanding of Users

Understanding the end user and their needs is necessary to design good UX [45]. Investing time to ensure that the entire team understands and agrees on the target audience results

in ease in the collection and utilisation of customer input throughout the development process and allows the UCD practitioners to stay true to their vision and enable them to make decisions on feature sets and design trajectories [59].

4.4.2 Sharing an Understanding of Design Vision

The collaboration between software engineers and usability specialists should be supported via facilitating communication of design intent and rationale [3]. The best UX vision is useless if it is not communicated to the development team [45]. Thus UCD practitioners should effectively share the design vision via communicating it to the development team. This visibility of design vision minimizes rework and illuminates integration issues early on [79] and allow team members to develop and be cognizant of the key design goals of the system in order to ease decision making in case of competing concerns. Moreover, setting the shared design goals allows team members to have a common understanding of important aspects of the system from the customer's perspective. Prioritized goals also allow the team to prioritize fixes [49]. Earlier externalization of design vision to stakeholders is encouraged in order to make the development of usable software more effective, achieve better collaboration, and produce better software faster [57].

Practices and Success Factors: Sharing an understanding of the design vision was achieved via a number of techniques including: the design studio [79], engaging developers in multiple design options [54, 59], developers taking part in UI specifications [2], sharing design artefacts and prototypes [10, 48, 21, 51, 9], and utilising information radiators [45].

4.4.3 Synchronizing Efforts of UCD Practitioners and Developers

Design drift is the occurrence of a difference between the implemented system from the initial design as a result of combining the efforts of developers and UCD practitioners. Synchronization allows the parallel usability and development efforts to proceed relatively smoothly. UI consistency may be undermined as independently empowered teams evolve code in parallel, without coordinating their work [19]. As a result, synchronization points are needed to allow for close collaboration that will keep the information flowing between all parties involved in the project. Moreover, UX practitioners reported that the lack of communicating frequent changes caused a lot of confusion and required an immense effort from the UX team to handle frequent changes in addition to struggling to remain on track with the development team schedule [11].

Practices and Success Factors: Synchronizing the activities of UCD practitioners and developers was addressed via attendance of UX team in daily scrums [62, 11, 10], daily communication of UX designers to clarify design and inform the developers about additions or changes required for the UI [2, 59, 24], and increasing the visibility of UX team's work [47, 19, 8].

4.5 Performing Usability Testing

Usability testing involves measuring typical users' performance on carefully prepared system tasks while watching and recording users performance and logging their software interactions [68, 38]. A number of sub-challenges were related to usability testing within an Agile context including: method of usability testing, scheduling usability testing, accessing users for usability testing, and high cost of running usability sessions. Further details on those sub-challenges is provided in the following subsections:

4.5.1 Method of Usability Testing

Agile time boxed nature poses challenges on conducting usability tests due to the difficulty of scheduling usability tests to evaluate and test prototypes and working builds with representative end users [48, 21, 22, 18, 19]. As a result, some Agile teams resolve to either peer test or do without usability testing, thus jeopardizing the quality of design.

Practices and Success Factors: The effect of tight Agile time lines on conducting usability testing was reportedly overcome via preparation for user research [42], utilising discount usability engineering techniques including: heuristic evaluation [56, 30] and RITE [21, 18], using low fidelity prototypes to conduct usability tests [30, 13, 40, 58], and conducting remote usability testing [19].

4.5.2 Scheduling Usability Testing

Scheduling interaction design evaluations with Agile development iterations was considered as a challenge due to lack of clarity in regards to timing of evaluations as part of the iterative structure of the Agile development process [22]. Moreover, conducting usability testing at the end of the Agile development process could lead to insufficient time and resources to respond to emerging usability issues whereas, if usability tests were done early in the development process this could lead to introducing usability defects in later iterations. Moreover, if usability tests were carried out as frequently as feature acceptance tests this could lead to massive budget increases [44]. Furthermore, the code generated during sprints is often too unstable to be subjected to usability tests even if it was scheduled [19].

Practices and Success Factors: The completion of iterations and releases was perceived as valuable opportunities to frequently test the software usability and declared that usability testing fits well with acceptance testing [23]. Some researchers suggested fitting usability testing in the context of other Agile development tests, for example, acceptance testing sessions (in the case of XP) [22, 38, 12] and demonstration sessions (in the case of scrum) [44] could serve as opportunities for usability feedback on the implemented interaction design. Another suggested technique was introducing a mandatory UI reviews as a gate keeping tool, where two sign offs were set one for code and one for UI[2].

4.5.3 Accessing Users for Usability Testing

The compressed Agile time scale posed difficulties in organising access to the right people at the right time for usability testing [22, 33, 19]. This is due to the need to plan user involvement and schedule appointments with studies' subjects sufficiently in advance and thus may not fit with the Agile development schedule since it may require lead times of weeks.

Practices and Success Factors: The ability to access users for usability testing in Agile teams was maintained via planning in advance for user inclusion [45], utilising an existing user pool to act as development partners or design partners to conduct usability testing [21, 81, 1], using user recruiting firms to frequently schedule for usability sessions [42], conducting remote usability testing [33, 19], and col-

laborative (peer review) UI inspections [56] via designers, developers, end users, graphics designers and usability specialists.

4.5.4 Shorter Time to Iterate Design

The Agile tight time lines allow little time to integrate usability testing feedback into subsequent development cycles. The reporting period for usability testing can be too long and subsequently many changes can occur in the application and as a result many recommendations were obsolete [38]. Agile teams reported lack of time to respond to results of usability evaluations and user feedback [19, 60].

Practices and Success Factors: Shorter time to iterate design with user feedback was handled via dedicating cycles for working on user feedback and incorporating it into the development life cycle [54] and utilising the UX practitioner to act as an Agile customer in order to validate designs that are passed to developers to implement, participate in cycle planning and ensure incorporation of user feedback into the development life cycle [60].

4.6 UCD Practitioner Workload

Although large organisations can afford a dedicated UCD practitioner per team or few teams which can lead to ease in resource allocation and design consistency. Nevertheless, in smaller organisations UCD practitioners are usually either shared among a number of teams or work as part timers. Due to the nature of Agile process this can impose an extra burden on UCD practitioners. A survey that was conducted on 30 teams that transitioned to Agile suggested that only 30% of the usability and UX design team believed that their teams are more effective after transitioning to Agile [50, 21]. This was attributed to sharing UCD practitoners among a number of teams, UCD practitioners attending several lengthy daily Agile meetings, and the tighter Agile time lines that did not provide enough time to finish UCD work [50, 21]. Moreover, the frequent context switching was causing a severe loss of productivity for UCD team members.

Practices and Success Factors: The increased load of UCD practitioners was addressed via conducting mentoring process to developers so as they can perform the role of UCD practitioner [2], office hours [50, 21], decreasing UCD practitioner workload [50, 21] and distributing UCD practitioner workload on two team roles; a UCD researcher and UCD a prototyper [81].

4.7 Lack of Documentation

Agile approaches strive to achieve minimal documentation. However, documentation is crucial for integrating Agile and UCD. The lack of proper requirements documentation was reported to lead to confusion in regards to UX deliverable [11]. A variety of integration related issues need to be documented including; first, documenting design rationale in order to justify and record prior design decisions [54]. Second, recording the source of requirements whether they are customers, users, developers, usability experts or usability elicitation guidelines because this can affect the decision of creating new user stories or modifying existing ones [61]. Third, there is a need to document current designs and their expected delivery date, usability test results, high level progress for late stage design chunks, recommendations and fixes for working versions, user and task information from external users, and the UI design to be implemented [76].

Practices and Success Factors: Lack of documentation was addressed through documenting via wikis [63, 8, 76], documenting via webpages [81], use cases [19], scenarios [63], personas [45], sketches [45], wire frames [45], prototypes [45], design patterns [54, 56], information radiators [45], and tool support [61].

5. CONCLUSIONS AND FUTURE WORK

This SLR aimed to identify and classify various challenging factors that restrict AUCDI and explore the proposed practices and success factors to deal with these challenges. The SLR included a total of 71 papers and excluded 80 papers that were published from the year 2000 till 2012. The findings were quantitatively classified according to year of publication and publication channel. AUCDI challenges were explored and their respective proposed practices and success factors were synthesized and a description and taxonomy of AUCDI challenges and its respective success factors and practices were reported. Industrial practitioners can utilise the description and taxonomy of AUCDI challenges and corresponding practices and success factors in identifying potential challenges of AUCDI and practices or success factors to deal with these anticipated challenges.

To enhance the SLR findings an empirical study will be conducted that investigates current industrial practices for integrating Agile development processes and UCD in order to verify and complement the findings of the SLR. This empirical study aims to identify the common difficulties and concerns that hinder AUCDI attempts and the proposed integration methods.

6. REFERENCES

[1] S. Adikari, C. McDonald, and J. Campbell. Little Design Up-Front: A Design Science Approach to Integrating Usability into Agile Requirements Engineering. In J. Jacko, editor, *Human-Computer Interaction. New Trends*, volume 5610 of *Lecture Notes in Computer Science*, pages 549–558. Springer Berlin / Heidelberg, 2009.

[2] M. Albisetti. Launchpad's Quest for a Better and Agile User Interface. In *11th International Conference on Agile Software Developement, XP2010*, Trondheim, Norway, 2010.

[3] S. Ambler. Tailoring Usability into Agile Software Development Projects. In E. Law, E. Hvannberg, and G. Cockton, editors, *Maturing Usability*, HumanŰComputer Interaction Series, pages 75–95. Springer London, 2008.

[4] J. Armitage. Are Agile Methods Good for Design? *Interactions*, 11(1):14–23, Jan. 2004.

[5] K. Beck. Manifesto for Agile Software Developement, 2000.

[6] H. Beyer, K. Holtzblatt, and L. Baker. An Agile Customer-Centered Method: Rapid Contextual Design. In *XP/AU*, 2004.

[7] S. Blomkvist. Towards a Model for Bridging Agile Development and User-Centered Design. In A. Seffah, J. Gulliksen, and M. Desmarais, editors, *Human-Centered Software Engineering Ů Integrating Usability in the Software Development Lifecycle*, volume 8 of *HumanŰComputer Interaction Series*, pages 219–244. Springer Netherlands, 2005.

[8] D. Broschinsky and L. Baker. Using Persona with XP at LANDesk Software, an Avocent Company. In *Proceedings of the Agile 2008*, AGILE '08, pages 543–548, Washington, DC, USA, 2008. IEEE Computer Society.

[9] J. Brown, G. Lindgaard, and R. Biddle. Stories, Sketches, and Lists: Developers and Interaction Designers Interacting Through Artefacts. In *Proceedings of the Agile 2008*, AGILE '08, pages 39–50, Washington, DC, USA, 2008. IEEE Computer Society.

[10] J. Brown, G. Lindgaard, and R. Biddle. Collaborative Events and Shared Artefacts: Agile Interaction Designers and Developers Working Toward Common Aims. In *Agile Conference (AGILE), 2011*, pages 87 –96, aug. 2011.

[11] M. Budwig, S. Jeong, and K. Kelkar. When User Experience Met Agile: A Case Study. In *Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pages 3075–3084, New York, NY, USA, 2009. ACM.

[12] R. Carbon, J. Dorr, and M. Trapp. Focusing Extreme Programming on Usability. In *GI Jahrestagung*, pages 147–152, 2004.

[13] S. Chamberlain, H. Sharp, and N. Maiden. Towards a Framework for Integrating agile Development and User-Centred Design. In *Proceedings of the 7th International Conference on Extreme Programming and Agile Processes in Software Engineering*, XP'06, pages 143–153, Berlin, Heidelberg, 2006. Springer-Verlag.

[14] T. Coatta and R. Rutter. UX Design and Agile: A Natural Fit? *Communications of the ACM*, 54(1):54–60, 2011.

[15] L. Constantine. Process Agility and Software Usability: Towards Lightweight Usage Centred Design, 2001.

[16] D. Cruzes and T. Dyba. Recommended Steps for Thematic Synthesis in Software Engineering. In *Proceedings of the 2011 International Symposium on Empirical Software Engineering and Measurement*, ESEM '11, pages 275–284, Washington, DC, USA, 2011. IEEE Computer Society.

[17] T. S. da Silva, A. Martin, F. Maurer, and M. Silveira. User-Centered Design and Agile Methods: A Systematic Review. In *Agile Conference (AGILE), 2011*, pages 77–86, aug. 2011.

[18] D. Dayton and C. Barnum. The Impact of Agile on User-Centered Design: Two Surveys Tell the Story. *Technical Communication*, 56(3):219–234, 2009.

[19] M. Detweiler. Managing UCD within Agile Projects. *Interactions*, 14(3):40–42, May 2007.

[20] M. Duchting, D. Zimmermann, and K. Nebe. Incorporating User Centered Requirement Engineering into Agile Software Development. In *Proceedings of the 12th international Conference on Human-Computer Interaction: Interaction Design and Usability*, HCI'07, pages 58–67, Berlin, Heidelberg, 2007. Springer-Verlag.

[21] M. Federoff and C. Courage. Successful User Experience in an Agile Enterprise Environment. In *Proceedings of the Symposium on Human Interface 2009 on Conference Universal Access in Human-Computer Interaction. Part I: Held as Part of HCI International 2009*, pages 233–242, Berlin, Heidelberg, 2009. Springer-Verlag.

[22] J. Ferreira. Interaction Design and Agile Development: Reconciling Iterative and Incremental Approaches. In *Agile Conference*, 2008.

[23] J. Ferreira, J. Noble, and R. Biddle. Agile Development Iterations and UI Design. In *Proceedings of the AGILE 2007*, AGILE '07, pages 50–58, Washington, DC, USA, 2007. IEEE Computer Society.

[24] J. Ferreira, J. Noble, and R. Biddle. Interaction Designers on eXtreme Programming Teams: Two Case Studies from the Real World. In *Proceedings of the Fifth New Zealand Computer Science Research Student Conference (NZCSRSC2007)*, 2007.

[25] J. Ferreira, J. Noble, and R. Biddle. Up-front Interaction Design in Agile Development. In *Proceedings of the 8th International Conference on Agile Processes in Software Engineering and Extreme Programming*, XP'07, pages 9–16, Berlin, Heidelberg, 2007. Springer-Verlag.

[26] J. Ferreira, H. Sharp, and H. Robinson. Values and Assumptions Shaping Agile Development and User Experience Design in Practice. In A. Sillitti, A. Martin, X. Wang, E. Whitworth, W. Aalst, J. Mylopoulos, M. Rosemann, M. J. Shaw, and C. Szyperski, editors, *Agile Processes in Software Engineering and Extreme Programming*, volume 48 of *Lecture Notes in Business Information Processing*, pages 178–183. Springer Berlin Heidelberg, 2010.

[27] J. Ferreira, H. Sharp, and H. Robinson. User Experience Design and Agile Development: Managing Cooperation through Articulation Work. *Software Practice and Experience*, 41(9):963–974, Aug. 2011.

[28] J. Ferreira, H. Sharp, and H. Robinson. Agile Development and User Experience Design Integration as an On-going Achievement in Practice. In *Agile 2012*, 2012.

[29] M. Fowler. The New Methodology, December 2005.

[30] D. Fox, J. Sillito, and F. Maurer. Agile Methods and User-Centered Design: How These Two Methodologies are Being Successfully Integrated in Industry. In *Proceedings of the Agile 2008*, AGILE '08, pages 63–72, Washington, DC, USA, 2008. IEEE Computer Society.

[31] J.-J. Garrett. *The Elements of User Experience:User Centered Design for the Web and Beyond*. New Riders, 2011.

[32] J. Gould and C. Lewis. Designing for Usability: Key Principles and What Designers Think. *Communications of ACM*, 28(3):300–311, Mar. 1985.

[33] P. Hodgetts. Experiences Integrating Sophisticated User Experience Design Practices into Agile Processes. In *Proceedings of the Agile Development Conference*, ADC '05, pages 235–242, Washington, DC, USA, 2005. IEEE Computer Society.

[34] A. Holzinger, M. Errath, G. Searle, B. Thurnher, and W. Slany. From Extreme Programming and Usability Engineering to Extreme Usability in Software Engineering Education (XP+UE→XU). In *Proceedings of the 29th Annual International*

Conference on Computer Software and Applications Conference, COMPSAC-W'05, pages 169–172, Washington, DC, USA, 2005. IEEE Computer Society.

[35] A. Hosseini-Khayat, T. Hellmann, and F. Maurer. Distributed and Automated Usability Testing of Low-Fidelity Prototypes. In *Agile Conference (AGILE)*, pages 59 –66, aug. 2010.

[36] W. Hudson. Adopting User-Centered Design within an Agile Process: A Conversation. *Cutter IT Journal*, 16.10:5–12, 2003.

[37] S. R. Humayoun, Y. Dubinsky, and T. Catarci. A Three-Fold Integration Framework to Incorporate User-Centered Design into Agile Software Development. In *Proceedings of the 2nd International Conference on Human Centered Design*, HCD'11, pages 55–64, Berlin, Heidelberg, 2011. Springer-Verlag.

[38] Z. Hussain, H. Milchrahm, S. Shahzad, W. Slany, M. Umgeher, T. Vlk, C. Koeffel, M. Tscheligi, and P. Wolkerstorfer. Practical Usability in XP Software Development Processes. In *The Fifth International Conference on Advances in computer Human Interactions, ACHI 2012*, 2012.

[39] Z. Hussain, W. Slany, and A. Holzinger. Current State of Agile User-Centered Design: A Survey. In *Proceedings of the 5th Symposium of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society on HCI and Usability for e-Inclusion*, USAB '09, pages 416–427, Berlin, Heidelberg, 2009. Springer-Verlag.

[40] Z. Hussain, W. Slany, and A. Holzinger. Investigating Agile User-Centered Design in Practice: A Grounded Theory Perspective. In *Proceedings of the 5th Symposium of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society on HCI and Usability for e-Inclusion*, USAB '09, pages 279–289, Berlin, Heidelberg, 2009. Springer-Verlag.

[41] Z. Hussain, W. Slany, and A. Holzinger. Agile Software Developement Methods and Usability/ User Centred Design: Prespectives from an Online Survey. *Journal of Systems and Software*, 2012.

[42] T. Illmensee and A. Muff. 5 Users Every Friday: A Case Study in Applied Research. In *Proceedings of the 2009 Agile Conference*, AGILE '09, pages 404–409, Washington, DC, USA, 2009. IEEE Computer Society.

[43] T. Jokela and P. Abrahamsson. Usability Assessment of an Extreme Programming Project: Close Co-operation with the Customer Does Not Equal to Good Usability. In F. Bomarius and H. Iida, editors, *Product Focused Software Process Improvement*, volume 3009 of *Lecture Notes in Computer Science*, pages 393–407. Springer Berlin / Heidelberg, 2004.

[44] D. Kane. Finding a Place for Discount Usability Engineering in Agile Development: Throwing Down the Gauntlet. In *Proceedings of the Conference on Agile Development*, ADC '03, Washington, DC, USA, 2003. IEEE Computer Society.

[45] J. Kollmann. Designing the User Experience in an Agile Context. Master's thesis, University College London, 2008.

[46] M. Larusdottir, E. Bjarnadottir, and J. Gulliksen. The Focus on Usability in Testing Practices in Industry. In P. Forbrig, F. Patern?, and A. Mark Pejtersen, editors, *Human-Computer Interaction*, volume 332 of *IFIP Advances in Information and Communication Technology*, pages 98–109. Springer Boston, 2010.

[47] J. C. Lee, T. Judge, and S. McCrickard. Evaluating EXtreme Scenario-Based Design in a Distributed Agile Team. In *Proceedings of the 2011 Annual Conference on Human factors in Computing Systems*, CHI EA '11, pages 863–877, New York, NY, USA, 2011. ACM.

[48] J. C. Lee and S. McCrickard. Towards Extreme(ly) Usable Software: Exploring Tensions Between Usability and Agile Software Development. In *Proceedings of the AGILE 2007*, AGILE '07, pages 59–71, Washington, DC, USA, 2007. IEEE Computer Society.

[49] J. C. Lee, S. McCrickard, and T. Stevens. Examining the Foundations of Agile Usability with eXtreme Scenario-Based Design. In *Agile Conference*, pages 3 –10, aug. 2009.

[50] A. Leszek and C. Courage. The Doctor is "In" – Using the Office Hours Concept to Make Limited Resources Most Effective. In *AGILE Conference 2008*, pages 196 –201, aug. 2008.

[51] M. Lievesley and J. Yee. The Role of the Interaction Designer in an Agile Software Development Process. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '06, pages 1025–1030, New York, NY, USA, 2006. ACM.

[52] H. Lindstrom and M. Malmsten. User-Centred Design and Agile Development: Rebuilding the Swedish National Union Catalogue. *The Code4Lib Journal*, 5:12–Ű15, 2008.

[53] B. Losada, M. Urretavizcaya, and I. Fernandez-deCastro. Agile Development of Interactive Software by means of User Objectives. In *The Sixth International Conference on Software Engineering Advances*, 2011.

[54] P. McInerney and F. Maurer. UCD in Agile Projects: Dream Team or Odd Couple? *Interactions*, 12(6):19–23, Nov. 2005.

[55] M. Mcneil. Agile User-Centred Design. In *Proceedings of the International Conference on Contemporary Ergonomics (CE2006)*, Cambridge, UK, 2006.

[56] T. Memmel, F. Gundelsweiler, and H. Reiterer. Agile Human-Centered Software Engineering. In *Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI...but not as we know it - Volume 1*, BCS-HCI '07, pages 167–175, Swinton, UK, UK, 2007. British Computer Society.

[57] T. Memmel, H. Reiterer, and A. Holzinger. Agile Methods and Visual Specification in Software Development: A Chance to Ensure Universal Access. In *Proceedings of the 4th International Conference on Universal Access in Human Computer Interaction: Coping with Diversity*, UAHCI'07, pages 453–462, Berlin, Heidelberg, 2007. Springer-Verlag.

[58] G. Meszaros and J. Aston. Adding Usability Testing to an Agile Project. In *Proceedings of the conference on AGILE 2006*, AGILE '06, pages 289–294, Washington, DC, USA, 2006. IEEE Computer Society.

[59] L. Miller. Case Study of Customer Input For a Successful Product. In *Proceedings of the Agile*

Development Conference, ADC '05, pages 225–234, Washington, DC, USA, 2005. IEEE Computer Society.

[60] L. Miller and D. Sy. Agile User Experience SIG. In *Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pages 2751–2754, New York, NY, USA, 2009. ACM.

[61] A. Moreno and A. Yagie. Agile User Stories Enriched with Usability. In C. Wohlin, editor, *Agile Processes in Software Engineering and Extreme Programming*, volume 111 of *Lecture Notes in Business Information Processing*, pages 168–176. Springer Berlin Heidelberg, 2012.

[62] M. Najafi and L. Toyoshiba. Two Case Studies of User Experience Design and Agile Development. In *Proceedings of the Agile 2008*, AGILE '08, pages 531–536, Washington, DC, USA, 2008. IEEE Computer Society.

[63] H. Obendorf and M. Finck. Scenario-Based Usability Engineering Techniques in Agile Development Processes. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '08, pages 2159–2166, New York, NY, USA, 2008. ACM.

[64] D. Parsons, R. Lal, and H. Ryu. Software Developement Methodologies, Agile Developement and Usability Engineering. In *18th Australian Conference on Information Systems*, 2007.

[65] J. Patton. Hitting the Target: Adding Interaction Design to Agile Software Development. In *OOPSLA 2002 Practitioners Reports*, OOPSLA '02, New York, NY, USA, 2002. ACM.

[66] C. S. A. Peixoto. Human-Computer Interface Expert System for Agile Methods. In *International Conference on Information Technology Interfaces*, pages 311–316, june 2009.

[67] C. S. A. Peixoto and A. E. A. da Silva. A Conceptual Knowledge Base Representation for Agile Design of Human-Computer Interface. In *Proceedings of the 3rd international Conference on Intelligent Information Technology Application*, IITA'09, pages 156–160, Piscataway, NJ, USA, 2009. IEEE Press.

[68] J. Preece, Y. Rogers, and H. Sharp. *Interaction Design: Beyond Human Computer Interaction*. John Wiley & Sons, 2002.

[69] P. Rannikko. User Centred Design in Agile Software Developement. Master's thesis, University of Tampere, School of Information Science, April 2011.

[70] M. Rittenbruch, G. McEwan, nigel Ward, T. Mansfield, and D. Bartenstein. Extreme Participation - Moving Extreme Programming Towards Participatory Design. In *Proceedings of the Seventh Biennial Participatory Design Conference*, 2002.

[71] D. Salah. A Framework for the Integration of User Centered Design and Agile Software Development Processes. In *33rd International Conference on Software Engineering (ICSE)*, pages 1132–1133, may 2011.

[72] D. Salah, H. Petrie, and R. Paige. Towards a Framework for Bridging User-Centred Design and Agile Software Development Processes. In *3rd Irish HCI Conference 2009*, 2009.

[73] H. Sharp, H. Robinson, and J. Segal. Integrating User-Centred Design and Software Engineering: a Role for eXtreme Programming? In *BCS-HCI Group's 7th Educators Workshop: Effective Teaching and Training in HCI*, 2004.

[74] M. Singh. U-SCRUM: An Agile Methodology for Promoting Usability. In *Agile Conference*, pages 555 –560, aug. 2008.

[75] O. Sohaib and K. Khan. Integrating Usability Engineering and Agile Software Development: A Literature Review. In *Computer Design and Applications (ICCDA), 2010 International Conference on*, volume 2, pages V2–32 –V2–38, june 2010.

[76] D. Sy. Adapting Usability Investigations for Agile User-Centered Design. *Journal of Usability Studies*, 2(3):112–132, May 2007.

[77] D. Sy and L. Miller. Optimizing Agile User-Centred Design. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '08, pages 3897–3900, New York, NY, USA, 2008. ACM.

[78] K. Tzanidou and J. Ferreira. Design and Development in the ŞAgile RoomŤ: Trialing Scrum at a Digital Agency. In A. Sillitti, A. Martin, X. Wang, and E. Whitworth, editors, *Agile Processes in Software Engineering and Extreme Programming*, volume 48 of *Lecture Notes in Business Information Processing*, pages 372–378. Springer Berlin Heidelberg, 2010.

[79] J. Ungar. The Design Studio: Interface Design for Agile Teams. In *Proceedings of the Agile 2008*, AGILE '08, pages 519–524, Washington, DC, USA, 2008. IEEE Computer Society.

[80] J. Ungar and J. White. Agile User Centered Design: Enter the Design Studio - A Case Study. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '08, pages 2167–2178, New York, NY, USA, 2008. ACM.

[81] H. Williams and A. Ferguson. The UCD Perspective: Before and After Agile. In *Proceedings of the AGILE 2007*, AGILE '07, pages 285–290, Washington, DC, USA, 2007. IEEE Computer Society.