On the Interaction between Socio-Technical Dimensions in the Next Generation of Requirements Engineering

Pankaj Kamthan
Concordia University, Montreal, Canada
kamthan@cse.concordia.ca

Abstract

The practice of human-centered methodologies for software development, such as agile methodologies, is impacted by the surrounding technological environment. This paper analyzes the potential of the Mobile Social Web in facilitating the social dimension of agile requirements engineering. In doing so, an agile requirements engineering process (AREP) is presented. AREP consists of a nonlinear sequence of technical activities, each of which is supported by a number of social activities. The interplay between activities underlying AREP and Social Web applications is illustrated by means of representative examples. The relationships to other software project artifacts, and to open source software (OSS), are highlighted.

Keywords: Agile Methodology, Collective Intelligence, Knowledge Engineering, Mobility, Social Dynamics, Web 2.0.

1. Introduction

The discipline of software engineering advocates a systematic and disciplined approach towards the development and evolution of software systems. There are a number of concerns in software development, including requirements engineering [1]. In requirements engineering, the attention is on the problem so as to devise a desirable and viable solution, namely the software system, which can satisfy the identified stakeholders. In software engineering, significant emphasis is placed on requirements engineering in the light of its impact on later phases of software development. It is therefore natural to seek means for effective requirements engineering processes.

It is evident that the ecosystem of software engineering is constantly changing, and influences the practice of software engineering. In the past decade, there have been a number of notable strategic changes in industrial software engineering, including the adoption of agile methodologies [2]. In this period, there have also been noteworthy technological changes, such as the emergence of the Social Web [3], and the proliferation of mobile devices. The extension of the Web to many-to-many communication paradigm, accessible via a variety of mobile devices, has made the Social Web a prospective choice for attending to the socio-technical aspect of software engineering [4], especially for geographically-dispersed software project teams.

In this paper, the interest is primarily in attending to the social dimension of agile requirements engineering from the perspective of the Mobile Social Web, although some of the arguments apply to other developmental contexts.

The rest of the paper is organized as follows. In Section 2, the necessary background for later discussion is provided and related work is presented. This is followed with the implications of Mobile Social Web Applications (MSWAs) for salient activities in the development of agile requirements in Section 3, and the limitations of a commitment to
the Mobile Social Web in Section 4. In Section 5, directions for future research are outlined. Finally, in Section 6, concluding remarks are given.

2. Background and Related Work

This section examines the elements of a conceptual model for the evolution of requirements engineering, a variation of which was presented earlier [5]. This is followed by a brief analysis of related work. For brevity and ease of reference, the model is termed as ADMS, and given in Figure 1.

![Diagram of ADMS model](image)

**Figure 1. The quad depicting the perceived directions of evolution of requirements engineering**

The possibility that the directions in ADMS are not necessarily mutually exclusive is indicated by grey dashed lines in Figure 1. These directions are also not necessarily (temporally) sequential. Indeed, as implied by later sections, they have influenced the evolution of each other, and it is these interrelationships that bring richness to ADMS.

2.1. Agile

In the 1990s, a number of limitations of rigidity in approaches for development of certain types of software systems were realized. The drive to cope with these limitations led to the inception of agility.

The Agile Manifesto characterizes the term “agile” and provides a vision for agile software development. It is motivated by the need for organizations to adequately respond to variability in the market, and to improve the relationship between technical and non-technical stakeholders, thereby reducing risk and increasing the likelihood of success of software projects. The manifesto constitutes the basis for a number of agile methodologies, including Agile Experience Design (AXD), Crystal Clear, Extreme Programming (XP), OpenUP, Scrum, and User-Centered Agile Process (UCAP). In certain cases, such as the Discipline Agile Delivery (DAD) process framework [6], elements of multiple agile methodologies have been included.

The agile methodologies have garnered support in national and international standards, as indicated by the ISO/IEC/IEEE 26511 Standard, the ISO/IEC/IEEE 26515 Standard, and the ISO/IEC/IEEE 29148 Standard. They have been applied to the
development of interactive software systems of different sizes in a variety of domains, such as computer games, healthcare, and real-time streaming media.

There are different types of agile requirement, and their adoption depends on the underlying agile methodology [1]. For example, an agile requirement can take the form of a use case (in Crystal Clear and OpenUP) or a user story (in AXD, XP, Scrum, and UCAP). The notions of use case and user story have evolved since their inception to better serve the current practice of agile methodologies.

There are a number of commercial tools aimed for conventional requirements engineering, such as Borland CaliberRM, IBM/Rational DOORS, and IBM/Rational RequisitePro. However, these tools are usually not deployed in agile requirements engineering. The use of tools in agile requirements engineering has been less common as compared to conventional requirements engineering. However, the situation is improving, and the use of tools is being seen as a necessity. There are tools specific to user story management, such as ThoughtWorks Mingle, that have built-in support for Wiki.

2.2. Digital

Traditionally, agile requirements engineering has relied on conversations among the stakeholders, on implicit knowledge of the stakeholders, and on paper. These approaches have a number of advantages. For example, they provide familiar channels for human-to-human interaction, maintain competitive advantage, and help reduce duplication.

However, it is being increasingly acknowledged that these non-rigorous means do not scale in the current environment of industrial software development [7]. For example, it is known that humans can have short-term memory; even otherwise, people may not be able to, or may not like being compelled to, recall crucial details. Thus, a tangible means for expressing and communicating agile requirements becomes necessary.

The now-superseded IEEE Standard 830, and the literature it is influenced by, is based on the premise that software requirements are expressed on and communicated via paper. Even though numerous advantages of paper have been reinforced by its ubiquitous use over millennia, there are also certain disadvantages that become apparent in the current software development environment. Indeed, drawbacks of paper have motivated calls, at the time when the outreach of agile methodologies was beginning to take shape, to “modernize” the way software requirements, in general, are expressed and communicated [8]. The recurring theme, express or implied, in these calls is digitization.

2.3. Mobile

The ubiquity of inexpensive and highly capable mobile devices, along with broad availability and access to wireless local area networks (LANs), has changed the landscape of computing. It has alleviated certain restrictions on space and time, and opened new vistas for personal communication and social interaction essential in the practice of software engineering, in general, and requirements engineering, in particular.

The interest in this paper is particularly in those mobile devices that possess adequate hardware and software capabilities for accessing and interacting with MSWAs. The devices with such capabilities currently in the market include a variety of smartphones, tablets, and netbooks.
For the aforementioned mobile devices, there are general-purpose, as well as special-purpose user agents, a detailed discussion of which is beyond the scope of this paper. For example, Opera Mobile is a general-purpose user agent, and WikiMobile is a special-purpose user agent.

2.4. Social

The stakeholders of agile requirements include, among others, software engineers, customers, and users. These roles can, if necessary, be specialized further, say, to provide a better understanding of the roles and/or to delegate some of the responsibilities elsewhere [9].

The principles and practices of the Agile Manifesto underscore the inherent social dimension of requirements engineering [10]. In particular, explicit involvement and active participation of non-technical stakeholders is crucial to agile methodologies [11].

The impact of information technology over the years has led to behavioral changes in these stakeholders, especially those that are digital natives [12]. This trend is likely to continue with the proliferation of mobile devices and the advent of the Social Web, and is illustrated in Figure 2.

![Figure 2. The new generation of stakeholders of requirements is owner of mobile devices, as well as avid user of the Social Web](image)

The Social Web supports the social dimension of agile requirements engineering [13] by congregating multiple, possibly disparate, mediums. In doing so, it provides a unified platform for the realization of (1) the relationships among people, and (2) the relationships between people and agile project artifacts, such as, agile requirements.

2.5. Related Work

There have been so far relatively few initiatives, in academia and in industry, towards utilizing the potential of the Mobile Social Web in software engineering, in general, and requirements engineering, in particular. This paper builds upon and/or extends these efforts.
From a perspective of stakeholders, the uses of Wiki in managing agile requirements have been outlined [14]. In it, the need for requirements engineering-specific Semantic Wiki for ameliorating some of the challenges in a conventional Wiki is also emphasized. The usability of Wikis in managing software requirements in course projects has been evaluated, with mixed results [15]. It has been shown that MSWAs enable various types of collaborations among teachers and students of software requirements [16]. Finally, the implications of the Social Web for agile requirements engineering have been explored [13].

3. Use of Mobile Social Web Applications in an Agile Requirements Engineering Process

To be systematic, the development of agile requirements is based on an agile requirements engineering process (AREP). AREP, by its very nature, is usually iterative as well as incremental.

3.1. Activities of AREP

In practice, an execution of AREP involves carrying out certain activities. In general, AREP consists of a number of technical, as well as individual and social activities, irrespective of the kind of agile methodology, type of requirement, or underlying technology [13]:

- **Technical Activities.** The common types of technical activities are Planning, Meeting, Authoring, Reviewing, and Publishing.
- **Social Activities.** The common types of social activities are Communicating, Collaborating, Negotiating, Disseminating, Syndicating, and Educating.

The aforementioned activities are common, although there is no claim that the list is exhaustive. The list of activities in AREP is also in alignment with the communication patterns that have been found in agile requirements engineering [17].

The technical activities in AREP typically follow a nonlinear sequence. Furthermore, the social activities occur, by necessity, as part of conducting technical activities. For example, Meeting involves Communicating, Collaborating, and Negotiating; Reviewing involves Educating; and Publishing involves Disseminating and Syndicating.

3.1.1. Mobile Social Web Applications: There are a number of SWAs available for use, including those collated and organized into various categories at Go2Web20 (http://www.go2web20.net/). The activities in AREP can be realized by one or more SWAs. The interest in this paper is especially in those SWAs that can be accessed via a mobile device of a type mentioned previously. MSWAs vary in a number of ways, including functionality, maturity, and terms of use. The selection of appropriate MSWAs is significant as the underlying choices affect AREP; however, is beyond the scope of this paper.

3.2. Planning

The purpose of planning is to identify, select, and allocate resources, including personnel, knowledge on the quality of agile requirements, and tools for authoring and managing agile requirements. It also involves deciding upon the schedules for meeting, authoring, and reviewing.
3.2.1. Illustrative Use of MSWA: There can be obstacles in deploying conventional means of setting-up a meeting among stakeholders. For example, the responsibility of finding common date(s) and time(s) often rests solely on the person chairing or moderating the meeting, accommodating the needs and preferences of each participant can be difficult, and the process can be unnecessarily time-consuming. The use of MSWAs such as Doodle and Google Calendar can facilitate the arrangement of such meeting(s). These applications have a number of features, including the ability by the participants to share and update the calendar at any time.

3.3. Meeting

The purpose of a meeting is to elicit necessary knowledge from customers and/or users in order to author the initial set of agile requirements. In doing so, a meeting can also help create shared understanding and build mutual trust.

3.3.1. Illustrative Use of MSWA: For building shared conceptualizations, for creating a ‘team memory’, and for reflection, taking notes during a meeting and/or recording minutes of the meeting can be useful. Usually, taking notes during a meeting is an individual activity. However, it can be useful for the participants to share and compare their notes at the conclusion of a meeting. For taking and sharing notes, there are a number of MSWAs, including Evernote, Microsoft Office OneNote, and UberNote.

3.3.2. Illustrative Use of MSWA: The parameters (such as locations and schedules) of meetings are prone to change, at times, abruptly. There can also be participants who, for some reason such as illness or travel schedule, may be unsure of their attendance, and may wish to periodically relay their state of readiness. In such an event, other participants need to be informed in a timely manner. The use of a microblogging service makes this possible. There are several microblogging services available, including Twitter and identi.ca. A microblog post is asynchronous and inexpensive (as per space occupied and time consumed). For example, the size of a Twitter post (or Tweet) is proportional to the size of the subject of an e-mail message. The use of microblogging is illustrated in Figure 3.

![Twitter microblog post](image_url)

@kamthan: John has not showed up with the slides yet. I am going to see if he is in the cafeteria.

Figure 3. A moment (state) in the preparation for a meeting captured by a microblog post

3.4. Authoring

The purpose of authoring is make agile requirements explicit.

3.4.1. Illustrative Use of MSWA: In a collaborative approach, brainstorming is often a prerequisite to authoring agile requirements, and for which there are a number of approaches. For example, chatting is a text-, audio-, or video-based approach, and mind mapping is a graphic-based approach, for brainstorming. There are a number of chat applications, such as Google Talk and micromobs. A chat session could be one-to-one, or many-to-many. In some cases, it is possible to log the proceedings of a chat session.
For creating mind maps, there are a number of MSWAs, including bubbl.us and Mindomo. The agile project team members can share these mind maps over the Web and, depending on the permissions, modify others’ maps.

3.4.2. Illustrative Use of MSWA: There are a number of possible approaches for authoring agile requirements, each with their own advantages and disadvantages. In one of the approaches, the agile project team selects a specific actor, user role, or persona, depending on the type of agile requirement, and authors all requirements related to it. This process is then repeated, and the agile requirements are subsequently annotated with metainformation. The result is a collection of agile requirements for the current iteration. There are a number of general-purpose MSWAs, such as Google Docs and Zoho that can assist in authoring agile requirements.

3.5. Reviewing

The purpose of reviewing agile requirements is to improve and to learn.

A review of agile requirements involves an assessment of their quality, and solicited feedback from certain selected stakeholders, including customers and users. For example, an agile requirement deemed relatively large for the current iteration needs to be split, it may need to be clarified, it may need to be reprioritized, and so on.

3.5.1. Illustrative Use of MSWA: A prototype design corresponding to an agile requirement may also suggest that the requirement is unacceptable by a user, or is unimplementable, and must either be revised or rejected. It is therefore this step that can trigger iteration in AREP. There are a number of MSWAs, such as Creately, Gliffy, and MockFlow that can assist in collaboratively creating both low- and high-fidelity prototypes. These prototypes can also be shared with others.

3.6. Publishing

The purpose of publishing agile requirements is their subsequent use by others in the agile project team, and beyond. For example, an agile requirement is used by those in the roles of project managers, designers, implementers, testers, customers, and users. (It could also be (re)used by others in future, similar projects.)

3.6.1. Illustrative Use of MSWA: There are a number of general-purpose MSWAs, such as Wiki, that can be used for publishing agile requirements. The possibility for hypertext, classification, and versioning makes a Wiki also useful for traceability. A Wiki is run by a Wiki system. There are several open source implementations of Wiki systems available. There are services such as WikiMatrix (http://www.wikimatrix.org/) that provide a regularly updated list of Wiki systems, as well as facilitate their comparison and appropriate selection. The commonly-used Wiki systems include MediaWiki (and its extension for the Semantic Web, namely Semantic MediaWiki) and TWiki, that can be acquired, installed, and administered under a variety of computing platforms.

3.6.2. Illustrative Use of MSWA: A MSWA, such as a Wiki, can also be used for associating, with an agile requirement, metainformation from other social activities. (The stakeholders of metainformation and information, that is, the agile requirement, need not be the same [18].) For example, social tagging of domain-specific terms can be used to classify requirements informally and to help locate relevant information in a collection of agile requirement descriptions; microblogging can be used for unplanned,
brief, announcements; a syndication mechanism can be used to asynchronously inform others of any changes to an agile requirement description; and a social network can be used to bookmark and share an agile requirement description. (There can also be provisions for commenting, rating, and voting.) This is illustrated in Figure 4 in a relatively abstract manner.

![Diagram of services provided by the Social Web](image)

**Figure 4. An assortment of services provided by the Social Web lends support to an agile requirement description**

In sociology, a *boundary object* describes information used in different ways by different communities. There are a number of characteristics of artifacts to act as boundary objects [19]: Modularization, Abstraction, Accommodation, and Standardization. From a sociological perspective, an appropriate agile requirement description, along with the metainformation garnered from MSWAs, acts as a boundary object.

### 3.7. Agile Requirements and Other Artifacts

The agile requirements are usually part of an ensemble of ‘lightweight’ agile project artifacts that can be developed in parallel, and allowed to co-evolve. These artifacts are interrelated (as well as, related, to other kinds of information), and examining the nature of relationships is crucial in understanding the context in which these artifacts co-exist.

#### 3.7.1. Illustrative Use of MSWA:

Indeed, agile requirements can be enriched by association with other, essential, agile project artifacts, such as user model and software project glossary, co-located or dispersed, each accessible via some mobile device. For example, pointing to a user model (say, user role or persona) can help demonstrate the relevance of an agile requirement to a user, and pointing to a software project glossary can help disambiguate and clarify terms, including abbreviations, in a natural language-based agile requirement description. This, for a specific user story, is illustrated in Figure 5.
There are other agile project artifacts, such as domain model, context-of-use model, and software design description, which can be pointed to similarly. These agile project artifacts can reside on the same MSWA (say, Wiki) or, take advantage of the distributed nature of the Social Web, and reside on different MSWAs.

3.7.2. Illustrative Use of MSWA: There can be multiple authors of an agile requirement description. These authors can have profiles, say, on the organizational social network, or on an organizational Wiki, and can be related dyadically in some relevant manner. This, inspired by the RELATIONSHIP ontology (http://vocab.org/relationship/), is illustrated in Figure 6 in a relatively abstract manner. (An analysis of the ensuing social dynamics between the authors or, more generally, between the stakeholders of agile requirements, due to their presence on a social network, is beyond the scope of this paper.)
4. Limitations of the Mobile Social Web for AREP

For an effective use of MSWAs in AREP, there are certain organizational, social, and technical limitations that need to be acknowledged and addressed. In some cases, the limitations are essential; in other cases, they are ephemeral.

4.1. Organizational Limitations

The nature of how a software project is executed is often a manifestation, even reflection, of an organization’s culture. The adoption of an agile methodology by an organization is not automatic. For example, if an agile methodology X is followed as-is, then an organization Y can face difficulties in retaining the ISO 9000 certification, or seeking such a certification. This, in turn, can discourage adoption of X by Y. In an agile project, there are several concerns (variables), with different priorities, competing for the same resources (constants). A transition to agile methodologies, in general, and agile requirements engineering, in particular, comes with benefits, as well as associated costs [20]. A commitment to Mobile Social Web depends intrinsically on an organization’s software development process maturity, in general, and requirements engineering process maturity, in particular [21].

There is currently no single MSWA that can be used for all the activities in AREP. This situation is unlikely to improve in the foreseeable future. Therefore, a commitment to Mobile Social Web invariably means the use of multiple MSWAs, not all of which are free-of-cost for commercial use. This, in turn, has non-positive implications for an organization’s administration and learning [22]. For example, an appropriate use of Wiki is not automatic, and may require training [23].

4.2. Social Limitations

The degree and depth of (voluntary) participation of stakeholders in the activities underlying AREP equipped with MSWAs depends on a number of factors, including individual personality characteristics, as well as customs and cultural traditions. For example, certain stakeholders may be indifferent towards contributing asynchronously or otherwise may be hesitant in expressing their views publicly in writing, for example, if the language of communication is not their first language, or certain members of a geographically-dispersed team may not be accustomed to communicating synchronously and remotely, across starkly different time zones. Therefore, participation of stakeholders is not necessarily automatic.

The potential for inactivity can have adverse consequences. For example, an insufficient contribution by stakeholders can unfavorably impact the quality of information during microblogging or social tagging.

Finally, (agile project) artifacts can transcend the spatiotemporal limitations inherent to people. However, they are not a replacement for people, for a number of reasons, including the fact that tacit knowledge cannot be expressed in form of artifacts, the issues related to affect cannot be understood by consumption of artifacts only, and trust cannot be established by communication of artifacts only.

4.3. Technical Limitations

There are a number of indirections that emanate from a technological commitment to an agile project. A connection to a wireless network can be constrained by space or time: (1) it may not be available at a certain location, or (2) it may not be available
during certain duration, or, if it is, then the connection could be prohibitively slow. The issues (1) and (2) can be perceived as disruptive, especially if there is expectation of constant connectivity.

For a positive user experience, it is crucial that all the technical elements involved work in unison. However, this may not be the case. For example, consider the following scenario. A given mobile device, say M, a given user agent, say U, and a given MSWA, say A, may not be pairwise compatible. In particular, it is not automatic that A has necessary support for M, or that either M or U have necessary support for A. This is, more likely than not, the case when M, U, and A, do not originate from the same organization and, evidently, different organizations tend to have different goals for their products.

Finally, to a certain extent, MSWAs also inherit the limitations of conventional Web Applications, especially pertaining to accessibility, interoperability, privacy, security, and usability [16]. There is currently no national or international standard for MSWAs, particularly that for interface metaphors. This can adversely impact the learning curve. MSWAs, like other globally distributed systems, do not, in general, come with an a priori guarantee of longevity or persistence. A MSWA available at one time may be discontinued at a later time, or its functionality may be significantly altered without prior notice. The discontinuation of the Google Notebook serves as an example.

5. Directions for Future Research

This section briefly highlights directions for future research that arise from the discussions in previous sections, and thereby suggest proneness to evolution of the ADMS model.

5.1. Empirical Assessment

The transition by an organization towards a methodology or technology is contingent on that organization’s culture. It is only recently that organizations have started considering the business potential of agile methodologies, as well as that of the Mobile Social Web, with varying outcomes. For a commercial entity, a consideration of the viability of an option in light of a financial investment is a crucial part of decision-making [24].

Therefore, an investigation of the analytics, metrics, and return on investment (ROI) of the adoption of MSWAs in software development organizations using agile methodologies is of research interest. For that, established qualitative and quantitative approaches, such as grounded theory and the Technology Acceptance Model (TAM), could be useful.

5.2. Mobile Social Web for OSS Requirements Engineering

There are apparent similarities between elicitation and management of requirements in agile projects and open source software (OSS) projects [25]. For example, OSS requirements engineering process has a significant social dimension. In OSS development, there is no notion of a “team” per se, but there is a collection of developers that are self-organized and usually geographically-dispersed. OSS requirements originate from these developers (who are also the users), and are expressed informally [26]. OpenUP is an open source process. The interest in agile methodologies, in general, has also spawned the development of agile project management systems, available as OSS.
The notion of openness, in general, and OSS, in particular, has also been instrumental in the development of the Social Web. For example, many MSWAs, including Wiki systems, are available as OSS. Indeed, OSS foundries such as SourceForge (http://sourceforge.net/) host a variety of, and increasing number of, MSWAs.

The increasing adoption of OSS in academia and industry motivates the study of the impact of information technology, in general, and the Mobile Social Web, in particular, on OSS requirements engineering.

5.3. Agile Requirements Engineering of MSWA

The transition by an organization towards a methodology or technology is contingent on that organization’s culture. It is only recently that organizations have started considering the business potential of agile methodologies, as well as that of the Mobile Social Web, with varying outcomes. For a commercial entity, a consideration of the viability of an option in light of a financial investment is a crucial part of decision-making [24].

The relationship between agile requirements and the Mobile Social Web is symbiotic, one direction of which is explored in this paper. The other direction, namely the development of agile requirements for MSWAs, is also of research interest.

A desirable candidate for such MSWA is a standards-based, user-centered, model-driven, and pattern-oriented agile requirements management system, available as OSS and accessible via a suitable mobile device. This MSWA could rely on AREP, as well as serve as a proof-of-concept for AREP.

6. Conclusion

The discipline of software engineering, in general, and the sub-discipline of requirements engineering, in particular, need to strike a balance between theory and practice. In theory, requirements engineering relies on computer science and conventional, relatively more established, engineering disciplines. In practice, requirements engineering, to a large extent, is social, as well as cultural, engineering. This reality must be made explicit in requirements engineering processes, and products resulting from those processes.

The technological environment in which software engineering resides and thrives is constantly changing. The Internet and the Web have fundamentally changed software engineering, in general, and requirements engineering, in particular, and the Mobile Social Web is envisaged to do the same. The Mobile Social Web nurtures connectivity, serendipity, and creativity, all of which are indispensable in the current and foreseeable future software development environment. It can help an agile project team to be more productive. In the long-term, the Mobile Social Web may lead to cost savings for the organization, and even reduce waste. The ADMS model along with the examples presented earlier are a step in that direction.
However, this paper does not advocate methodological or technological determinism. The extent of commitment to agility by an organization, or that of the acceptance of the Mobile Social Web by its people, can vary broadly, for a number of reasons, including preference for rigidity and unclear ROI. The Mobile Social Web is relatively new, and its technological infrastructure is yet to mature. The use of the Mobile Social Web, as with its predecessors, can have inadvertent, possibly unacceptable, side-effects. There are open, non-trivial, issues with respect to the quality of MSWAs. Indeed, for reasons stated earlier, care must be exercised by an organization in the adoption of MSWAs. Therefore, a commitment of the Mobile Social Web to agile methodologies, in general, and AREP, in particular, obligates prudence.

References


Author

Pankaj Kamthan

The author has a Ph.D. in Computer Science, has been teaching in academia and industry for the past two decades, and has served on several program committees of conferences and editorial boards of journals related to e-learning, information technology, and resource management.