A Systematic Review on Requirements Documentation Challenges in Agile Methodology

Kaiass Elghariani, Nazri Kama, Nurulhuda Firdaus

Advanced Informatics School, Universiti Teknologi Malaysia, Jalan Sultan Yahya Petra, 54100 Kuala Lumpur, Malaysia.

Abstract

Agile methodology has become commonly used since it offers further collaboration with clients. Requirements engineering is one of the software development process in agile approaches. Agile has some practices, which unlike other traditional software development methods e.g. face to face communication. Software developers are using the definition of agile methodology to improve the agile way of activities. Besides, the challenge of dealing with requirements engineering practices is not much known in agile way or model. However, many literature works have been done in agile requirements engineering and challenges faced by software developers. The aim of this paper is to present the challenges of agile requirements engineering and to fill the gap by presenting the evidence about the challenge of requirements documentation activity faced by agile team members. To gain better understanding about the challenge of traditional requirements engineering which have been solved by adopting agile approaches. The conducted systematic literature reviews are recently published and more than 15 research papers are identified. The research findings can be references to all researchers who are interested in this research area because the paper’s findings clarify that requirements documentation in agile methodology need to have further attention by researchers to better understand the impact of agile requirements engineering documentation challenges.

Keywords: Agile Methodology, Requirements Engineering, Implicit Thinking Documentation

1. Introduction

Agile Methodology (AM) has been improved a lot in software engineering best practices. Recently, developers are looking for more flexibility to develop software systems, which can lead to provide efficient services to their customers [1]. However, agile approaches are mostly having the same tasks in terms of the agility and end-user involvement. Undoubtedly, many software development models are designed to assist the developers to build their software effectively. Meanwhile, huge volume of professional knowledge is usually communicated as part of the software team deliberations. Usually part of this knowledge is explicitly documented in the form of meeting minutes, modeling diagrams, test cases, code, etc.

* Corresponding author. E-mail address: aekaiss2@utm.my
This explicitly documented knowledge can be organized and shared easily. But, substantial experience knowledge remains undocumented and implicitly kept in software engineers’ minds. This experience knowledge is classified as tacit knowledge, which is usually communicated orally or through observation. Though it is important capturing tacit knowledge has twofold challenge. Firstly it is unseen and secondly knowledge experts usually unconsciously exploit it. In other words it is hardly explicated. This characteristic is best reflected by Polanyi’s theory of personal knowledge “we know more than we can tell” [2].

2. Related Works

Zhu [3] has conducted a literature review on Requirements Engineering (RE) practices such as observations, interviews; workshops and strong team collaboration are embedded in iteration-based agile methods. Likewise, Cao & Ramesh [4] stated that RE practices such as customer involvement, requirements prioritization; Boness & Harrison [5] has also conducted a review on requirements modeling [6], Kitchenham & Charters [7] has mentioned that requirements documentation have also been suggested to be used with agile methods.

Boness & Harrison [5] Conducted a review mentioned the lack of software developers’ implicit thinking documentation has become a very critical practice that has not been given any concentration on it. While developing a software project, comments, assumptions and opinions are made about the environment in which it will operate. Often, the implicit thinking is not explicitly recorded and documented, but is built into the system. Such assumptions are unlimited in number, and can fail at any time, causing the software to fail to fulfill its purpose.

3. Research Method

This research process has adopted a strategy provided by Kitchenham & Charters [7]. Therefore, the research described about the main steps of the systematic review, namely planning, conducting the review and reporting the review results [7].

3.1 Planning the review

The first step of planning this review is proposing a research questions related to the research goals. The search strategy has been defined by providing a search string and the inclusion/exclusion criteria. Consequently, the sections below describe with details about them.

3.1.1 Review Objectives and Research Questions

Since the use of agile methods in software development is being increased, it is very significance to study the role of requirement engineering documentation in agile approaches. The following research questions are designed to attain the research objectives.
RQ1. How is requirements documenting in agile methodology?

RQ2. Is there any implicit thinking Documentation in agile methodology?

3.1.2 Search strategy

The guidelines [7] of the study are used for this research. When the research questions and objectives have been conducted, we have started preparing the search strategy to study and review all the empirical studies particular to the research objectives. The plan included the definition of the search space, which included electronic databases and online search engine.

3.1.3 Research Criteria

The criteria of this research review consist of two part criteria 1 and criteria 2, as explained below: Criteria 1 (C1) is a research keywords related to agile software development methods such as agility, agile, Scrum, XP (Extreme Programming) etc.

Inclusion and exclusion criteria aimed to make sure that the study is included. Therefore, Table 1 summarizes the inclusion and exclusion criteria have been used in this research review.

<table>
<thead>
<tr>
<th>No.</th>
<th>Inclusion</th>
<th>Exclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Study is a peer-reviewed publication.</td>
<td>Studies that do not focus explicitly on agile methods</td>
</tr>
<tr>
<td>2</td>
<td>Study is in English.</td>
<td>Studies that do not discuss RE Documentation in agile methods.</td>
</tr>
<tr>
<td>3</td>
<td>Relevant to the search terms defined in the above sections.</td>
<td>Studies that do not meet inclusion criteria.</td>
</tr>
<tr>
<td>4</td>
<td>An empirical research paper, an experience report, or workshop paper.</td>
<td>Opinion, viewpoint, keynote, discussions, editorials, comments, etc.</td>
</tr>
<tr>
<td>5</td>
<td>Study is published between 2000 and 2016.</td>
<td></td>
</tr>
</tbody>
</table>

3.2 Conducting the Review

Conducting the review means that presenting the findings of this research and extraction of information from relevant sources and databases as summarized in Table 1.

By following the search strategy, it is important to note that we only selected databases
that publish peer-reviewed papers (I1). We have done an extensive inspection of the studies’ titles and abstracts by applying the inclusion criteria. Most of the retrieved studies fell within the inclusion criteria I2, I3 and I5.

Then, in the second task, we have reviewed the pre-selected studies in order to apply the exclusion criteria (E1, E2, E3, and E4). We have been read the entire paper and then excluded the studies based on the defined exclusion criteria.

<table>
<thead>
<tr>
<th>Database</th>
<th>Retrieved</th>
<th>Included</th>
<th>Excluded</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE</td>
<td>7</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>ACM</td>
<td>5</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Science Direct</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>Total</td>
<td>15</td>
<td>11</td>
<td>4</td>
</tr>
</tbody>
</table>

3.3 Writing the Review

Work by Kitchenham & Charters [7] provided a guideline that is adopted in this research; the review was being written and described based on the research questions.

(RQ1) How is Requirements Documenting in AM?

Agile methodology is mostly focusing on tacit knowledge and frequent face-to-face communication in place of documentation for explicitly supporting knowledge sharing and decisions [4]. Unreliable and lengthy documentation of requirements specifications is another challenge of traditional requirements engineering [10]. Furthermore, lengthy specification documents are difficult to compile and consume excessive amount of time, which can cause delays. Agile methods discourage lengthy documentation and support face-to-face communication with constant feedback and reviews. Often, the only documentation involved in agile methods includes user stories, product backlogs, burn down charts, etc. [8].

(RQ2) Is there any Implicit Thinking Documentation in AM?

According to Polanyi [9], all knowledge has tacit components and there is nothing between tacit and explicit. He contended that there is not something amenable for conversion [9]. However, a later focus of implicit thinking on some topics is called tacit knowledge [10]. Implicit thinking describes a knowledge that is not explicit [11]. WordWeb online dictionary defines Implicit Knowledge as “Implied though not directly expressed; inherent within the nature of something.” This definition gives the view that implicit thinking is a form of tacit knowledge, which is not expressed promptly but observed from the nature of something. However, there is an indirect difference between Implicit Thinking and Tacit Knowledge. In other words, implicit thinking is that which
has not been put together either by expressions, concept development and assumptions, which leads to principles. Since [9], the literature of implicit thinking is not widely mentioned, even though [10] defends and argues that Polanyi never talked about implicit thinking and that his work is vastly misinterpreted. Implicit thinking is more about thinking “how” to do something, but some as this may be hard to describe explicitly. However, the definitions given above can be said as that, with a careful observation of something or someone, implicit thinking can be extracted but it remains implicit as it could not possibly be explicated.

4. Conclusion

The research paper shows a systematic review on challenges of agile requirements engineering documentation. The guideline of conducting a systematic literature review provided by [7] is adopted in this review to study the available researches about agile methodology and its practices. About 32 research studies stored in common electronic databases, 12 research papers are included to our research topic. Moreover, further analyzing and categorizing have been made into these researches according to the research questions: (i) Agile requirement engineering documentation; and (ii) implicit thinking documentation. The findings in this research provide future dimensions to industry and research experts for further work on requirements engineering documentation in agile approaches.

References