International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 10 (2018) pp. 134-142
© MIR Labs, www.mirlabs.net/ijcisim/index.html

Crowdsourcing for Requirements Engineering: A Simplified Review

Sabrina Ahmad, Nurul Atikah Rosmadi, Sharifah Sakinah Syed Ahmad and Siti Azirah Asmai

Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka
Hang Tuah Jaya,
76100 Durian Tunggal Melaka, Malaysia
{sabrinaahmad, sakinah, azirah}@utem.edu.my; atikahrosmadi@gmail.com

***Abstract*: The need to deal with large scale stakeholders to ensure the correctness of software requirements make crowdsourcing technique very useful. It helps to improve the optimal level of requirements quality in terms of breath and save the development cost at the same time. This paper presents and provides evidence of the relevance of crowdsourcing for requirements engineering. A systematic literature review method is adopted here and the literature exploration is based on two specific research questions. The findings from the literature show that many efforts have been done to explore and further improve crowdsourcing for software engineering in general and requirements engineering in specific. This paper provides a foundation to pursue research in improving crowdsourcing technique for the benefit of requirements engineering.**

Keywords: requirements engineering, crowdsourcing, review, elicitation

I. Introduction

Requirements are the basis of software system which provides fundamental information to estimate the project duration and cost, to design the system and eventually evolved into a usable system. Establishing requirements enables us to agree on and to visualize the end software product. The process of obtaining and translating the stakeholders' needs into proper requirements statements is called Requirement Engineering (RE). The process basically mould the shape of the entire software development life cycle as the process is concerned on determining the goals, functions and constraints for software systems [1]. This is also supported by Pandey et al.[2] as he stated in an article that requirement engineering process covers the importance of the entire system and software development life cycle. Even though there are many requirements on RE, they all share the idea that requirements involves finding out what people want from a computer system, and understanding what their needs mean in terms of design.

The challenge is the requirements must be no less than complete but no more than necessary, detailed enough to be verifiable and achievable but free from describing the design decisions. Thus, the dilemma lies between the depth and the breath of the requirements [3]. The depth and the breath of requirements usually come from the stakeholders. However, referring to a study [4], it has been shown that user are seldom involved, despite the common understanding that doing it would result in better quality requirements and higher chance of project success. This is also supported by the Standish CHAOS Report[5] that stated the user involvement seems the most important success factor for IT projects.

Therefore, it is not surprising many methods are introduced to enhance the user involvement. Traditionally, requirements are elicited through conventional techniques such as interview, observation, document analysis, workshops and brainstorming. The requirements elicited from those techniques are then translated into requirements specification and written properly in a document. Subsequently, the documents are checked and confirmed with the stakeholders through inspection and walkthrough to proceed to the next stage; the design. These techniques are able to explore the depth very successfully. However, in order to cover the breadth of the requirements, a large pool of stakeholders is needed.

In any software development project, it is common understanding that the more people involved the more the project need to spend. Therefore, the need of the requirements breadth can become very costly or time consuming when involved a large numbers of stakeholders. As a result, in order to cover the pool of stakeholders over its entire breadth while remaining cost-efficient, smaller numbers of stakeholders are carefully selected to represent the certain group as key stakeholders. By doing that, the breadth is not guaranteed. Figure 1 illustrates the optimal balance between depth and breadth of requirements.

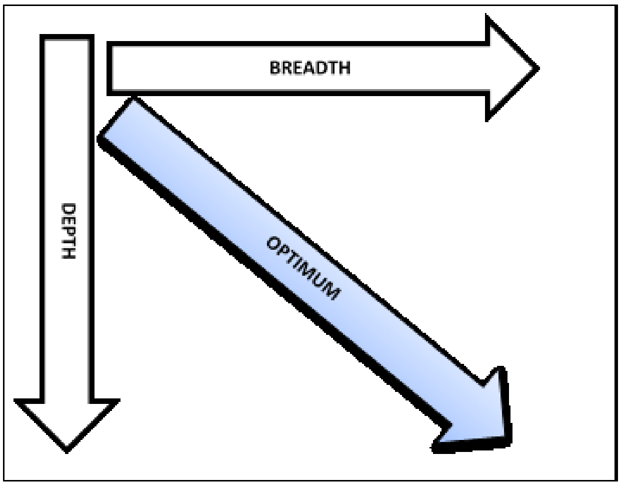


Figure 1. Illustration of the optimal balance between breadth and depth (Taken from [3])

In the era of automation and big data evolution, requirements engineer can elicit and validate requirements with a far broader stakeholders' pool to compliment the traditional techniques. The popular term to that is called crowdsourcing. Crowdsourcing gives the requirements engineer access to a wide diversity of actual and potential stakeholders or users. It has the potential to increase the quality and breadth and even the feasibility of requirements elicitation.

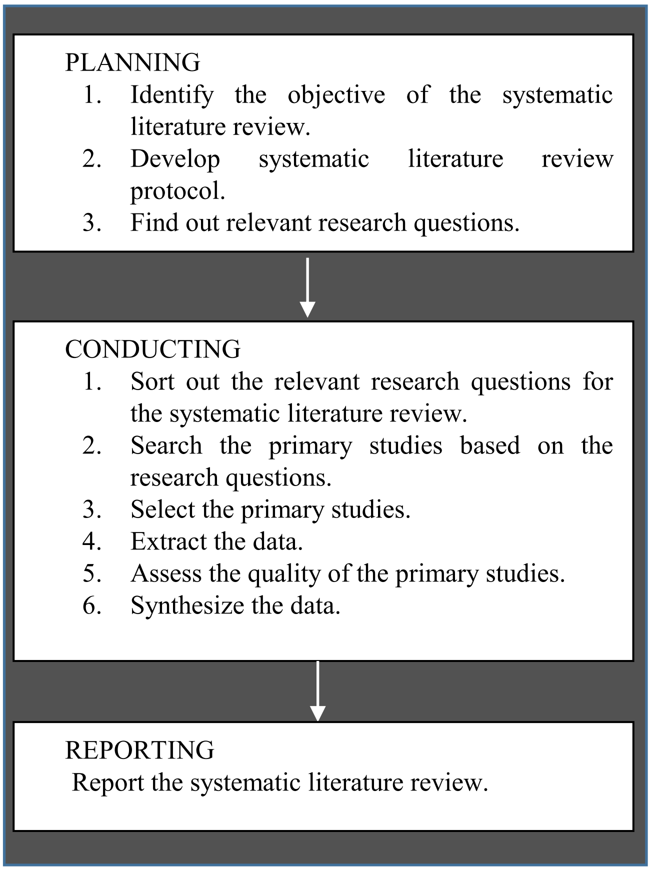
This paper is presenting a simplified literature review to provide an overview of the crowdsourcing technique and its usefulness for requirements engineering. Following Introduction, Section 2 provides the review method. This is followed by the review results in Section 3. Section 4 concludes the paper.

II. Review Method

A. The Systematic Literature Review Method

This paper adopted a systematic literature review method to systematically accumulate, organize, evaluate, and synthesize all existing research evidence related to our research area in order to provide testimony and confidence. It establishes a connection between the existing knowledge and the problem to be solved [6]. The systematic review explores the crowdsourcing technique and how it can benefit requirements engineering. In general, the systematic review is divided into three phases which are planning, conducting and reporting. Figure 2 illustrates the three basic phases.

The systematic literature review started with planning phase in which the objectives are identified to set the purpose on the particular literature review. The objectives basically guide through the searching direction to discover what crowdsourcing is and how it can benefit requirements engineering. In order to achieve the objectives, the review protocol is established. The protocol specified the questions to be addressed, the databases to be searched and the methods to identify, assemble and assess the evidences obtained. Then, relevant research questions are listed down. Following the planning phase, the conducting phase sort out the relevant research questions identified in the planning phase. Next, the primary studies are identified and selected based on the research questions. Then, data is extracted, primary studies quality is assessed and data is synthesized. The systematic literature review is concluded by the reporting phase in which the findings is written properly to report the relevant studies and significant data.



PLANNING

1. Identify the objective of the systematic literature review.
2. Develop systematic literature review protocol.
3. Find out relevant research questions.

CONDUCTING

1. Sort out the relevant research questions for the systematic literature review.
2. Search the primary studies based on the research questions.
3. Select the primary studies.
4. Extract the data.
5. Assess the quality of the primary studies.
6. Synthesize the data.

REPORTING

Report the systematic literature review.

Figure 2. Phases in systematic literature review

B. Review Questions

The questions are focused to develop understanding on crowdsourcing technique and how does it fits in software engineering. It also focused to find the evidence on the strength of crowdsourcing technique and how it benefits the requirements engineering. The research questions are as follows.

1. Research Questions 1: "What is crowdsourcing in Software Engineering?"
2. Research Questions 3: "How does crowdsourcing technique benefits requirements engineering?"

Table 1 shows the relationship between the research questions and the research motivations.

Research Questions		Motivations
RQ1:	What is crowdsourcing in Software Engineering?	Discover and understand the crowdsourcing technique.
RQ2:	How does crowdsourcing technique benefit requirements engineering?	Find out the capability and strength of crowdsourcing to benefit requirements engineering.

Table 1. Research questions and motivations.

C. Search Strategies

This sub-section elaborates on the searching strategies through the digital libraries and databases. The searching is done through search strings and refined search strings.

Listed below are the list of digital databases which happen to be the most popular and familiar databases to ease and broad the set of related literatures:

1) IEEE Xplore (ieeexplore.iee.org)
2) ScienceDirect (sciencedirect.com)
3) Springer (Springerlink.com)
4) Scopus (scopus.com)
5) Google Scholar (scholar.google.com)
6) Elsevier (Elsevier.com)
7) ACM Digital Library (dl.acm.org)
8) Cornell University Library (arXiv.org)

The search strings are based on the research questions and relevant keywords related to the research area such as crowdsourcing, crowd-focus, crowd-centric, crowd-based and requirements engineering. The searching were based on keywords in the title and author names ranging from year 2005 to 2017 covering journal articles, conference proceedings and professional magazine articles. Language for the search was limited to English only.

D. Identification and Selection of Relevant Literature

The search was based on the guideline to write the systematic literature review constructed upon the research questions [7]. The two research questions were answered by the related works searched in order to find out the current finding on the research title. The keywords from the primary studies or keywords that we already known were used to find more articles related to the research. The synonym words and alternative words related to the studies also applied in order to optimize the related work search.

The general keywords used in searching the related articles for this systematic literature review were "“requirements engineering”, and “crowdsourcing”. Other than that we specified the search to “crowd-based”, “crowd-centric”, “crowd-focus,” “requirement elicitation” and “crowd information retrieval”. The searched was resulted in the various reliable journals and conference proceedings covering issues in the crowdsourcing for requirements engineering. In order to obtain as many citations as possible, we also used library facilities as to access full text articles to non-subscribed databases such as Springer.

Upon the completion of the searching process, the findings were filtered and streamlined to related works only. The next phase of review was based on the references of relevant articles to the research questions. Table 2 describes the inclusion and exclusion criteria during the search.

Inclusion criteria	Exclusion criteria
• Journal articles, conference proceedings, professional magazine articles on: ○ Requirements engineering ○ Requirements elicitation ○ Crowdsourcing ○ Crowd-centric ○ Crowd-based ○ Crowd-focused	• Tutorials • Studies not related to research questions • Studies which are unclear

Table 2. The Inclusion and Exclusion Criteria

In this literature review, we have included 32 primary studies that related to crowdsourcing and the relevance in requirements engineering. The distribution over the years is presented to identify the interest of studies in this field.

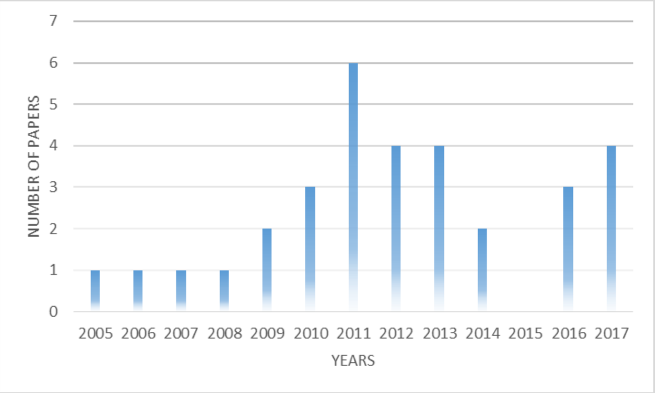


Figure 3. Number of Studies Graph

III. The Review Results

A. Question 1: What is Crowdsourcing in Software Engineering?

We generally review definitions of the term crowdsourcing before going into the role of crowdsourcing in software engineering. Crowdsourcing is a combination of the words ‘crowd’ and ‘outsourcing’. Howe[8] first used the term ‘crowdsourcing’ in Wired magazine article. Estellés-Arolas [9] defines crowdsourcing as a type of participative online activity in which a principal proposes a voluntary undertaking task via a flexible open call.

Crowdsourcing has been used extensively in various disciplines, and the term ‘Crowdsourced Software Engineering’ represents the uses of crowdsourcing technique to support software development. It emphasizes any software engineering activity included, thereby encompassing activities that do not necessarily yield software in themselves, such as requirements elicitation, test case refinement and project planning[10].

An extensive survey has been done and concluded that crowdsourcing software engineering can be define as below[11] :

> *"Crowdsourced Software Engineering is the act of undertaking any external software engineering tasks by an undefined, potentially large group of online workers in an open call format."*

The same survey shows a cumulative growth of crowdsourced software engineering studies published before April 2015 and Figure 4 captured the survey conducted.

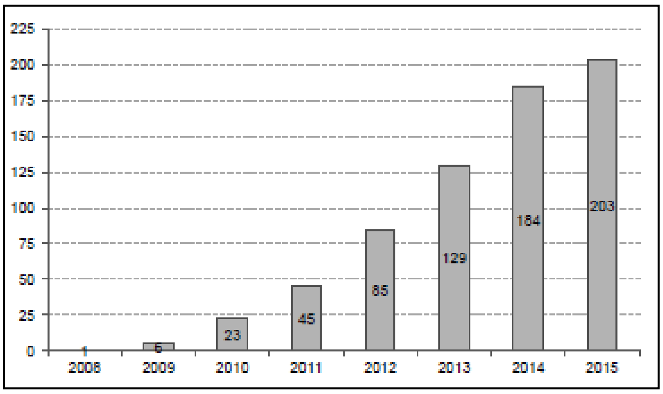


Figure 4: Studies in Crowdsourced Software Engineering (Taken from [19])

The studies span through all software engineering activities from requirements engineering to testing. A survey through Google Scholar shows that more studies on crowdsourced software engineering are emerging in 2016 to date and the pattern still shows a cumulative growth.

Crowdsourcing has high potential to make a better software engineering nowadays in comparison to the traditional method as large scale data is popular nowadays. Crowdsourcing may help software development organizations integrate elastic, external human resources to reduce cost from internal employment, and exploit the distributed production model to expedite the development process. Besides, software engineering can benefit from crowdsourcing as either a source of knowledge needed to develop new software or a source for ideas and feedback on existing software[10].

Many organizations are increasingly using crowdsourcing as a new model for value creation, where new web technologies are used to outsource tasks, which are traditionally performed by specialist or a small group of experts, to unified large group of people [12]. According to Cheon[13], they use regional matchmaking technique at Internet-connected computers in volunteer computing. The crowdsourcing is to take work and outsource it to a crowd of workers as in mass collaboration as mass collaborations have been successful in various business and social activities. Crowdsourcing relies on the intelligence of crowds of people, in cases humans have capabilities that go beyond those available in computers [14], to solve problems that neither could solve well alone.

There is a growing interest in 'engaging the crowd' to identify or develop innovative solutions to public problems. Since the prevalence of crowdsourcing in industry and academia, several surveys about crowdsourcing for the general purpose were published. Howe [8] explains that crowdsourcing is not a single strategy, but "an umbrella term for a highly varied group of approaches." Crowdsourcing as a real-life instance of human collective intelligence is a phenomenon that changes the way organizations use the Internet to collect idea, solve complex cognitive problems and build high-quality repositories (e.g. Wikipedia) by self-organizing agents around data and knowledge [15]. Due to the prevalence of crowdsourcing in industry and academia in recent years, several surveys about crowdsourcing for general purposes were published. Quinn [6] [16] gave an overview of the human computation tasks and taxonomy of crowdsourcing system. According to Kittur et al.[17], crowdsourcing refers to the use of small amounts of time and effort from a large number of individuals to solve large problems. The best part about crowdsourcing is that the problem-solving is done by both amateurs and experts, thus creating a network of such individuals collaborating together.

The advantage of crowdsourcing is that the solutions are often highly relevant to the intended audience because members of the audience are directly involved in the ideation and proposed solutions. For example, the crowdsourcing platforms is Amazon Mechanical Turk, the reCaptcha (for book digitization) and the ESP game (for image labeling).

In this respect, Jeff Howe tracks how the internet has managed to create a highly efficient network of people who can perform tasks exceptionally well. The leveraging upon the internet as a scalable platform for coordinating collaboration has undoubtedly revolutionized the way problems and challenges are approached.

Mutual benefit is an essential component of crowdsourcing and number of project surveys has identified some common reason volunteers participate. These include the size of the challenge, the necessity for volunteer contribution, collaboration with prestigious institutions, contribution to research, education, mental stimulation, being part of a community, personal research interests and enhancing a research from which they will benefit[18]. This technique has long been used in various aspects [9, 14, 19]. Understanding the factors that contribute to project success is important for crowdsourcing's continued adoption, efficient and effective implementation, and maximizing its potential. Website statistics shared by crowdsourcing project teams provide evidence that the potential of the crowd can be significant [16].

1) Crowdsourcing Process

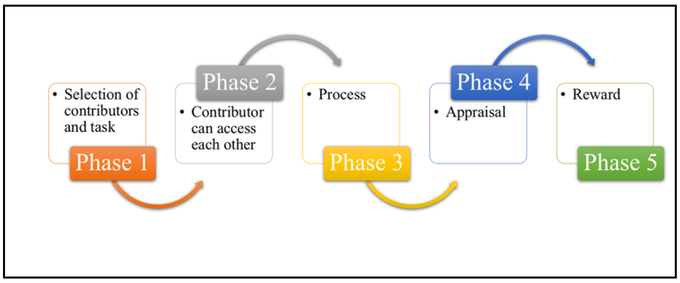
In order to support the crowdsourced software engineering, we survey the crowdsourcing process and presented here. The sequence of crowdsourcing process is independent of the order in which a seeker (an organization or individual) decides on the individual characteristics when it plans the process. Geiger et al. [20] gave details of a taxonomic framework for a crowdsourcing process that identified four dimensions (which is pre-selection of contributors, accessibility of peer contributions, aggregation of contributions and remuneration for contributions) that impacted the process of sourcing and aggregating contributions from the crowd. Meanwhile, Muhdi et al. [21] gave five activities involved in a crowdsourcing process which is Deliberation, Preparation, Execution, Assessment, and Post-processing. Li et al. [22] also provided that general process of crowdsourcing.

From the processes mentioned in the above literature, we can conclude that they present the whole process in crowdsourcing. Nevertheless, some studies only involved a part of the process of crowdsourcing that is the selection of contributors and task [17, 23] . Table 3 categorizes the process of crowdsourcing.

Based on our review, we can summarize the process of crowdsourcing in Figure 5. We can categorize the crowdsourcing process in five processes. It begins with the selection of contributors and task phase; the contributor can access each other phase; process phase; appraisal phase and reward phase.

Selection of contributors and task			**Contributor can access each other**	**Process**	**Appraisal**	**Reward**	**References**
Deliberation Preparation				Execution	Assessment	Post-processing	Muhdi et al., [21]
Pre-selection of contributors			Accessibility of peer contributions	Aggregation of contributions		Remuneration for contributions	Geiger et al. [20]
Release task				Participants submit solution	Choose the best solution	Reward the winner The contractees get the best solution	Li and Hongjuan [22]
-Filter by constraints -Distribute subtask -Refine by profile	-Define goals -Characterize jobs -Profile workers -Identify constraints -Design guided CS algorithm	-Recruitment -Reputation -Payment & incentives -Detecting poor quality work					Lykourentzou et al. [23], Kittur et al.[17]

Table 3: The Crowdsourcing Process



Selection of contributors and task
Phase 1
Phase 2
Contributor can access each other
Process
Phase 3
Phase 4
Appraisal
Reward
Phase 5

Figure 5: Crowdsourcing Process

The first phase is the process of choosing the right contributor for a specific task as outlined in a written job description. Besides that, some studies [17, 23] involved a part of crowdsourcing which is in the selection of contributors and task. Next step is contributor can access each other. In this process, contributors can be precise mechanisms to express their opinion on individual contributions. The third phase is how the crowd contribution within a crowdsourcing process is used by the crowdsourcing organization to achieve the desired outcome. The appraisal phase begins when the submitted ideas are clustered, rated, and best ideas will be rewarded. The final phase is a reward, determines how contributions are paid or otherwise compensated for their work.

Based on crowdsourcing process, we can use this approach in volunteering management. This approach can get more information and faster from the crowd and to be more precise with the requirement from beneficiaries.

B Question 2: How Requirements Engineering can benefit from Crowdsourcing?

Crowdsourcing started to gain popularity ever since high reliability and ever-increasing utilization of web based system and mobile applications by our society. The challenge is obvious as the requirements engineer need to encounter a wider audience of users who basically represents general public; the crowd. Crowdsourcing gives the requirements engineering team access to a wide diversity of actual and potential users. This would allow access to gain a wider and more up to date knowledge of how users perceive the system role in meeting their requirements and to understand how that perception changes over time [24]. While requirements elicitation has been well studied for relatively stable domain like banking or health, the new paradigm like cloud and mobile application rely on the crowd to cater for their needs properly[25]. Therefore, crowdsourcing is deemed useful to accommodate the complexity and the scale of the crowd. At the same time, the crowdsourcing also would help identifying a comprehensive set of stakeholders from an initial sets identified earlier[26].

Through crowdsourcing, the crowd provide inputs upon request and could also be involved in validation [27] and prioritization of requirements [28]. In addition, a proper initiative will be able to keep the crowd repeating the requirement elicitation process by providing useful input to further strengthen the software product in the future. According to Pagano and Maalej[29], users' feedback on software could help requirements engineer to better understand the requirements of the next release of the system.

The crowdsourcing utilizes text and usage mining to derive potential requirements. Written language is a central medium for storing, sharing and communicating content which can be analyse with the help of text mining. Text mining is used to find primarily conscious requirements which can be put in words. The relevant portions of text data are then identified and classified on a per-sentence level using language patterns and considered as requirements. Therefore, reports can be generated by aggregating the data. This is the strength of crowdsourcing technique, unlike conventional requirements elicitation, text mining is neither subject to the interpersonal effects of interviews nor limited to answers to questions asked like questionnaires [3] Beyond that, usage mining helps measure and analyze user behavior to uncover various types of requirements [25] just like traditional ethnography technique but with many users simultaneously. Besides, usage mining can also reveal subconscious requirements which people fail to specifically identify them. In traditional RE, it is called tacit knowledge which usually buried in one's mind with an assumption that the unsaid requirement is understood. In addition, the unconscious requirements can go beyond essential requirements to the innovative ideas[30]. It is great opportunity to receive inputs from the voluntary collective wisdom with diverse expertise instead of limited amount of specified experts through crowdsourcing platform.

In general, Table 4 summarizes the crowdsourcing benefits for requirements engineering.

	Benefits	Source
1.	Conscious requirements identification	• Groen & Koch [3] • Hosseini et al. [24] • Snijders et al. [31] • Adepetu et al. [25]
2.	Unconscious requirements discovery	• Groen & Koch [3]
3.	Innovative ideas encouragement	• Chen et al. [30] • Murukannaiah et al. [32]
4.	Empirical prioritization	• Snijders et al. [31] • Lim et al. [33] • Brabham [28]
5.	Empirical validation	• Ali et al. [34] • Stolee & Elbaum [27] • Brabham [28]
6.	Feedback-based requirements engineering	• Pagano & Maalej [29]
7.	Requirement-Driven Social Adaptation	• Ali et al. [34] • Ali et al. [35]
8.	Stakeholders discovery	• Lim et al. [26]
9.	Wider scale of stakeholders involvement coverage	• Hosseini et al. [24] • Groen & Koch [3]

Table 4. Summary of Crowdsourcing Benefits for Requirements Engineering

Referring to the promising benefits, it is anticipated that much research efforts have been done to further improve crowdsourcing for requirements engineering. Table 5 describes research efforts to improve the process and the quality of the requirements.

	Research efforts	Description	Sources
1.	CrowdRE	An approach to acquire creative requirements from the crowd.	Murukannaiah e. al. [32]
2.	Crowd-centric	Improved RE method to increase user satisfaction through crowdsourcing and gamification	Snijders et al. [31]
3.	Stakesource 2.0	A web-based tool that utilize social networks and collaborative filtering to identify and prioritize requirements.	Lim et al. [33]
4.	StakeRare	A method that uses social networks and collaborative filtering to identify and prioritize requirements in large software project.	Lim & Finkelstein [36]

5.	CRUISE	A tool for Crowdsourcing Requirements Elicitation and Evolution	Sharma & Sureka [37]
6.	Crowdsource and Personas	A collaborative application, and persona builder as a tool to generate personas to represent a specific set of users through real user profiles and data collected through third party services.	Alvertis et al. [38]
7.	REfine	A crowsourced gamified platform to boost and sustain interest in requirements engineering participatory	Snijders et al. [39]

Table 5. Research efforts on Crowdsourcing for Requirements Engineering

Current researches span through approaches, techniques, tools and web-based platform to assist requirements engineering process in many ways while utilizing crowdsourcing benefits. Each of the effort is unique in a way to solve specific problem or to address explicit concern in any of the requirements engineering area. In as early as 2011, the initiative of crowdsourcing yield from utilizing the social network to identify and to prioritize input from the crowd [33] [36]. The effort of just getting requirements from the crowd are then focusing on effort to improve the relationship with the crowd and establishing good rapport by improving the crowd satisfaction. [31, 39]. In 2016, CrowdRE is introduced as an approach to acquire creative requirements from the crowd. Following that, another effort is made to introduce a tool in order to ease the crowdsourcing technique for a smooth process and quality requirements.

I. Conclusion

Crowdsourcing techniques promised broad audiences to provide useful inputs as crowd interacts, generate ideas and discusses them. These can help the requirements engineers understand the wide range of stakeholders in order to provide the right services and products. The crowdsourcing has strength in providing the breadth of knowledge towards the optimal set of requirements. However, it needs support from other techniques in order to dig in the depth of the requirements.

As the nature of system evolve, this paper presents the literature on crowdsourcing technique to compliment conventional requirements elicitation technique to access a wide scale of stakeholders for optimal requirements. The benefits seem prominent and the crowdsourcing appears to be a significant research area to strengthen requirements engineering in the future.

For the benefit of researchers exploring crowdsourcing in requirements engineering, this paper also summarizes crowdsourcing benefits for requirements engineering. Besides, information on related research efforts to date are also presented.

Acknowledgment

We are thankful to Universiti Teknikal Malaysia Melaka for allowing us to leverage the university's resources to conduct the initial stage of this research.

References

[1] Laplante, P.A., Agresti, W.W., Djavanshir, G.R. "Guest Editors' Introduction: IT Quality Enhancement and Process Improvement. IT Professional Magazine", (9), pp. 10. 2007.

[2] Pandey, D., Suman, U., Ramani, A. "An effective requirement engineering process model for software development and requirements management". In *International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom)*, pp. 287-291, 2010.

[3] Groen, E.C. Koch, M. "How Requirements Engineering can benefit from crowds. Requirements Engineering Magazine". 2016.

[4] Kujala, S., Kauppinen, M., Lehtola, L., Kojo, T. "The role of user involvement in requirements quality and project success". In *Proceedings of 13th IEEE International Conference on Requirements Engineering*, pp. 75-84, 2005.

[5] Standish Group. "Chaos Summary 2009*". Technical Report*, 2009.

[6] Salleh, N. "A systematic review of pair programming research-initial results". In *Proc. New Zealand Computer Science Research Student Conference (NZCSRSC08), Christchurch*, 2008.

[7] Kitchenham, B. Charters, S."Guidelines for performing systematic literature reviews in software engineering*". EBSE Technical Report Ver. 2.3(EBSE 2007-001)*, 2007.

[8] Howe, J. "The rise of crowdsourcing. Wired magazine", (14), pp. 1-4. 2006.

[9] Estellés-Arolas, E. González-Ladrón-de-Guevara, F. "Towards an integrated crowdsourcing definition. Journal of Information science", (38), pp. 189-200. 2012.

[10] Stol, K.-J., LaToza, T.D., Bird, C. "Crowdsourcing for Software Engineering. IEEE Software", (34), pp. 30-36. 2017.

[11] Mao, K., Capra, L., Harman, M., Jia, Y. "A survey of the use of crowdsourcing in software engineering. Journal of Systems and Software", (126), pp. 57-84. 2017.

[12] Greengard, S. "Following the crowd. Communications of the ACM", (54), pp. 20-22. 2011.

[13] Cheon, E., Kim, M., Kuk, S., Kim, H.S. "A regional matchmaking technique for improving efficiency in volunteer computing environment". In *First ACIS/JNU*

International Conference on Computers, Networks, Systems and Industrial Engineering (CNSI) pp. 285-289, 2011.

[14] Yuen, M.-C., Chen, L.-J., King, I. "A survey of human computation systems". In *International Conference on Computational Science and Engineering (CSE)*, pp. 723-728, 2009.

[15] Buecheler, T., Lonigro, R., Füchslin, R.M., Pfeifer, R. "Modeling and simulating crowdsourcing as a complex biological system: human crowds manifesting collective intelligence on the internet". In *ECAL*, pp. 109-116, 2011.

[16] Quinn, A.J. Bederson, B.B. "Human computation: a survey and taxonomy of a growing field". In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1403-1412, 2011.

[17] Kittur, A., Nickerson, J.V., Bernstein, M., Gerber, E., Shaw, A., Zimmerman, J., Lease, M., Horton, J. "The future of crowd work". In *Proceedings of the 2013 conference on Computer supported cooperative work*, pp. 1301-1318, 2013.

[18] McKinley, D. "How effectively are crowdsourcing websites supporting volunteer participation and quality contribution. New Zealand". 2013.

[19] Mohan, S., Agarwal, N., Al-Doski, L. "Mobile network-aware social computing applications: a framework, architecture, and analysis. Journal of Ambient Intelligence and Humanized Computing", (4), pp. 43-56. 2013.

[20] Geiger, D., Seedorf, S., Schulze, T., Nickerson, R.C., Schader, M. "Managing the Crowd: Towards a Taxonomy of Crowdsourcing Processes". In *AMCIS*, 2011.

[21] Muhdi, L., Daiber, M., Friesike, S., Boutellier, R. "Crowdsourcing: an alternative idea generation approach in the early innovation process phase of innovation. International Journal of Entrepreneurship and Innovation Management", (14), pp. 315-332. 2011.

[22] Li, Z. Hongjuan, Z. "Research of crowdsourcing model based on case study". In *Service Systems and Service Management (ICSSSM), 2011 8th International Conference on*, pp. 1-5, 2011.

[23] Lykourentzou, I., Vergados, D.J., Papadaki, K., Naudet, Y. "Guided crowdsourcing for collective work coordination in corporate environments". In *International Conference on Computational Collective Intelligence*, pp. 90-99, 2013.

[24] Hosseini, M., Phalp, K.T., Taylor, J., Ali, R. "Towards crowdsourcing for requirements engineering. 2014.

[25] Adepetu, A., Ahmed, K.A., Al Abd, Y., Al Zaabi, A., Svetinovic, D. "CrowdREquire: A Requirements Engineering Crowdsourcing Platform". In *AAAI Spring Symposium: Wisdom of the Crowd*, pp. 2-7, 2012.

[26] Lim, S.L., Quercia, D., Finkelstein, A. "StakeNet: using social networks to analyse the stakeholders of large-scale software projects". In *Proceedings of the 32Nd ACM/IEEE International Conference on Software Engineering-Volume 1*, pp. 295-304, 2010.

[27] Stolee, K.T. Elbaum, S. "Exploring the use of crowdsourcing to support empirical studies in software engineering". In *Proceedings of the 2010 ACM-IEEE international symposium on Empirical software engineering and measurement*, pp. 35, 2010.

[28] Brabham, D.C. "Crowdsourcing as a model for problem solving: An introduction and cases. Convergence", (14), pp. 75-90. 2008.

[29] Pagano, D. Maalej, W. "User feedback in the appstore: An empirical study". In *21st IEEE International Requirements Engineering Conference (RE)*, pp. 125-134, 2013.

[30] Chen, H.-M., Kazman, R., Garbajosa, J., Gonzalez, E. "Big Data Value Engineering for Business Model Innovation". In *Proceedings of the 50th Hawaii International Conference on System Sciences*, 2017.

[31] Snijders, R., Dalpiaz, F., Hosseini, M., Shahri, A., Ali, R. "Crowd-centric requirements engineering". In *Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*, pp. 614-615, 2014.

[32] Murukannaiah, P.K., Ajmeri, N., Singh, M.P. "Acquiring creative requirements from the crowd: Understanding the influences of personality and creative potential in Crowd RE". In *24th International Requirements Engineering Conference (RE)*, pp. 176-185, 2016.

[33] Lim, S.L., Damian, D., Finkelstein, A. "StakeSource2. 0: using social networks of stakeholders to identify and prioritise requirements". In *33rd International Conference on Software Engineering (ICSE)*, pp. 1022-1024, 2011.

[34] Ali, R., Solis, C., Salehie, M., Omoronyia, I., Nuseibeh, B., Maalej, W. "Social sensing: when users become monitors". In *Proceedings of the 19th ACM SIGSOFT symposium and the 13th European conference on Foundations of software engineering*, pp. 476-479, 2011.

[35] Ali, R., Solis, C., Omoronyia, I., Salehie, M., Nuseibeh, B. "Social adaptation: when software gives users a voice". In *Proceedings of 7th International Conference Evaluation of Novel Approaches to Software Engineering*, pp. 28-30, 2012.

[36] Lim, S.L. Finkelstein, A. "StakeRare: using social networks and collaborative filtering for large-scale requirements elicitation. IEEE Transactions on Software Engineering", (38), pp. 707-735. 2012.

[37] Sharma, R. Sureka, A. "CRUISE: A Platform for Crowdsourcing Requirements Elicitation and Evolution. 2017.

[38] Alvertis, I., Papaspyros, D., Koussouris, S., Mouzakitis, S., Askounis, D. "Using crowdsourced and anonymized Personas in the requirements elicitation and software development phases of software engineering". In *11th International Conference on Availability, Reliability and Security (ARES)*, pp. 851-856, 2016.

[39] Snijders, R., Dalpiaz, F., Brinkkemper, S., Hosseini, M., Ali, R. and Ozum, A. "REfine: A gamified platform for participatory requirements engineering". In *IEEE 1st International Workshop on Crowd-Based Requirements Engineering (CrowdRE)*, pp. 1-6, 2015.

Author Biographies

Dr. Sabrina Ahmad was born in Muar, Johor, Malaysia on August, 15 1978. She obtained her first degree in Information Technology (Hons.) from Universiti Utara Malaysia in 2000. In 2002, she obtained Master of Science (Computer Science – Real-time Software Engineering) from Universiti Teknologi Malaysia. Her doctoral degree (PhD in Computer Science) was obtained from The University of Western Australia in 2012. She is specialized in Software Engineering particularly in Requirements Engineering. Dr. Sabrina is currently a Coordinator for Centre of Advanced Computing Technology and a Senior Lecturer at Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka.

Dr. Sharifah Sakinah Syed Ahmad is currently an associate professor in the Department of Intelligent Computing and Analytics, Faculty of Information & Communication Technology, Universiti Teknikal Malaysia Melaka (UTeM). She received her Bachelors and Masters degrees of Applied Mathematics in School of Mathematics from University Science Malaysia. Following this, she received her Ph.D from the University Of Alberta, Canada in 2012 in Intelligent System. Her research in graduate school focused on the granular computing and fuzzy modeling. Her current research interests including evolutionary optimization, fuzzy system, and granular computing. Her current research work is on the granular fuzzy rule-based system, Evolutionary Method, Modeling and Data Science.

Dr. Siti Azirah Asmai is a senior lecturer at Faculty of Information and Communication at Universiti Teknikal Malaysia Melaka (UTeM), Malaysia. She obtained her BSc in Computer Science from Universiti Teknologi Malaysia and MSc. in Information and Communication Technology for Engineer from Conventry University. She received her Ph.D. in Information and Communication Technology from UTeM in 2014. Her current research work focuses predictive analysis, Data analytics and Visualization, Time Series and Intelligent Computing.