

AN INVESTIGATION OF SOFTWARE REQUIREMENTS PRACTICES AMONG SOFTWARE PRACTITIONERS: A STUDY IN JEDDAH

Azida Zainol¹ and Haneen Hassan Al-Ahmadi²

Department of Computer Science, King Abdulaziz University, Jeddah, Kingdom of Saudi Arabia

ABSTRACT

This paper presents the result of software requirements practices survey among software practitioners, a study in a city of Jeddah, Saudi Arabia. As software requirements are important and they lead to the successful of a software development project, it becomes interesting to investigate the current software requirements practices in the kingdom of Saudi Arabia. As an initial work, a survey was conducted in Jeddah as a study before we conduct it in the Kingdom of Saudi Arabia software industry. The survey is conducted by distributing a set of questionnaire to the software practitioners. There are 17 respondents completed the questionnaire out of 50 distributed questionnaire, which is 34% of response rate. The result of this survey is promising and it has shown that requirements management area should be focused for future improvement. In the future, the survey will focus on software engineering and requirements engineering practices over the entire Kingdom of Saudi Arabia software industry.

KEYWORDS

Software engineering practice, Requirements engineering practice, Software Requirements practice, Requirements management

1. INTRODUCTION

Nowadays software plays an important role in our lives. Software can be found anywhere across variety of domain, such as in education, banking, medical and many others. Thus, developing high quality software is becoming increasing and demanding in these days [1]. In order to ensure producing high quality software, it is essential to have and manage requirements appropriately over the entire software development. Hence, it is important to understand how the software practitioners conducting software requirements during the software development. It has been a huge issue over a decade in the industry that many organizations are struggling to understand, document and manage the software requirements [2].

This paper presents the state of the art software requirements practices in the city of Jeddah. Although this paper presents only a city, it helps us to understand and benchmark how we practice software requirements against the current state of the art in the software requirements research area. It shows areas that software practitioners have adequately performed as well as it shows certain areas that need improvement urgently. Thus, it aids in identifying the energy and resources to the targeted areas for improvement efficiently.

A survey was conducted in order to understand how the software practitioners conducted the software requirements and at the same time to get their consent regarding employing an open source tool during software development. The result of this survey is very promising and it has shown the software engineering domains that have been conducted comprehensively as well as domains that need urgent improvement in the future.

2. REVIEW

Requirements is defined as a statement of what the system must do or what characteristic it must have [3]. A complete definition from the IEEE software engineering glossary [4] stated that a requirements is defined as :

“(1) A condition or capability needed by a user to solve a problem or achieve an objective; (2) a condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents; (3) a documented representation of a condition or capability as in (1) or (2).”

Moreover, the Guide to the Software Engineering Body of Knowledge – SWEBOK [5] has defined software requirements as:

“A property that must be exhibited by something in order to solve some problem in the real world. It may aim to automate part of a task for someone to support the business processes of an organization, to correct shortcomings of existing software, or to control a device—to name just a few of the many problems for which software solutions are possible.”

Theoretically, requirements are important during software development because they describe what the user needs and why the user needs it [6], [7]. It is essential to identify the correct requirements in the early stage of software development because the cost of correcting a defect that is found late in the project is far more than to fix it shortly after its creation [8]. One of the major consequences of requirements problem is rework, rework can consume 30 to 50 percent of total development cost [9] and requirements error account for 79 to 85 percent of the rework cost [10]. Preventing requirements errors and catching them early therefore has a huge effect on reducing rework. These days, developing software projects is becoming difficult and challenging. The majority of software development projects in the USA will take longer, cost more than planned and result in "out of specification" products that fail to meet user requirements [11]. Moreover, a report on a survey of over 3800 organizations in 17 European countries concluded that more than 50% of the perceived software problems were in the area of requirements specification and requirements management [12]. Based on this evidence, it can be stated that requirements are very important and it should be given extra attention because it leads to the successful of a software development project.

As software requirements are related to software engineering and requirements engineering, we reviewed several surveys regarding software engineering and requirements engineering in the literature. During this literature reviews, we collected the related variables that have been used to measure the software engineering and requirements management practices.

A survey was conducted among Australian software practitioners regarding their software development practices during recent software project and the survey helped them to understand the requirements issues and their relationship with project success [13].

Moreover, a survey of requirements management practices in the Malaysian software industry was conducted in 2008 [14]. The result of this survey helped them to understand how the software practitioners conducted their requirements management practices and therefore the areas which were lack of practices have been identified for improvement.

In 2015, a survey of software engineering practices has been conducted in Turkey [15]. The result of this survey became interested to software engineering professional in Turkey and world-wide. It has shown the areas of strength and weakness that would encourage the industry and academia to collaborate in those areas.

Based on these related survey of software engineering generally and specifically for software requirements, we can constructed the survey based on the measured variables as discuss in section 3. We also confident that the software engineering practices survey is able to be conducted in the kingdom of Saudi Arabia software industry in order for us to continue with empirical study. As an initial work, we focus on a survey in a city of Jeddah in order to understand the process of investigation in the larger area.

3. METHOD

In this investigation we used a survey to collect the software requirements practices among the software practitioners who involved in small and medium software development projects in Jeddah, Kingdom of Saudi Arabia. In this survey, we used questionnaire that contains 19 questions. The questionnaire has two objectives which are to collect the current practices during software development process and to study how they are dealing with their software requirement. The questionnaire is designed based on CMMI that focus on requirements management process area (REQM) [16].

The questionnaire takes about 15 minutes to complete and should be completed by software engineers or someone who has software development background purposely for small and medium software projects. It is divided into three main sections, namely demographic profile, software development process and software requirements practices.

In the demographic section, we are interested to know the background of the respondents so that the results collected are promising. Hence, we collected the following variables; type of organization, how many years of experience in software development and the numbers of employees that involved in software development.

Moreover, in software development section, we are interested to investigate how the software practitioners conducted software development. Thus, we gathered these variables [1], [13], [14], [15],:

- The frequency of employing a methodology in every software development project
- The use of specific methodology in software development. The methodologies are : object-oriented development, structured development, information engineering, prototyping, formal specification or others.
- The use of any requirements management tools that help the respondents in managing their requirements. The tools are : Rational RequisitePro, RaQuest, Cradle, PACE, SpeedReq, Caliber-RM, AnalystPro, Tracer, others or never used any tools.
- The need to employ open source requirement management tool that is freely available in the market.
- The features that the respondents are looking for in a requirements management tool. The features are [17], [18], [19] :
 - The tool should be easy to use. Not too much training and administration needed. The tool should not create additional tasks and deployment should not require extensive customization
 - The tool must have access control whereby each participants as appropriate access to the data. (role-based project-based and task based access control)

- The tool must be adaptable and extensible to need of the organization or project
- The tool should be free licencing that allow the user to use the tool in full version without limitation
- The tool should be database centric, but also support document management
- Others, please specify

The questionnaire is designed to collect the software requirements management practices. Thus, the following variables are designed based on CMMI – Requirements Management (REQM) process area [16] :

- Do you uniquely identify each requirements?
- Do you defined policies for requirements management?
- Do you record requirements traceability from original sources?
- Do you define traceability policies?
- Do you maintain a traceability manual?
- Do you use a database to manage requirements?
- Do you define change management policies?
- Do you identify global system requirements?
- Do you identify volatile requirements?
- Do you record rejected requirements?
- Do you reuse requirements over different projects?

The questionnaire was administered and distributed using SurveyMonkey. The respondents are contacted via email and WhatsApp application to take part in the survey by clicking the link provided.

4. RESULT AND DISCUSSIONS

This section describes the finding of this survey arranging as demographic profile, software development process and software requirements practices. We distributed the questionnaire to 50 respondents and we managed to collect the complete responses from 17 respondents, given as 34% of complete responses. Although the responses are low, yet, it can show to us the actual current state of software requirements practices.

4.1. Demographic profile

The first section of the survey is about collecting the demographic profile of the respondents. The respondents were asked to identify the type of company, namely government, semi government, private and small and medium enterprise. The result has revealed that majority of the respondents are working in the private sector with 52.54%, government with 23.53%, small and medium enterprise 17.65% and semi government with 5.88% as shown in Figure 1.

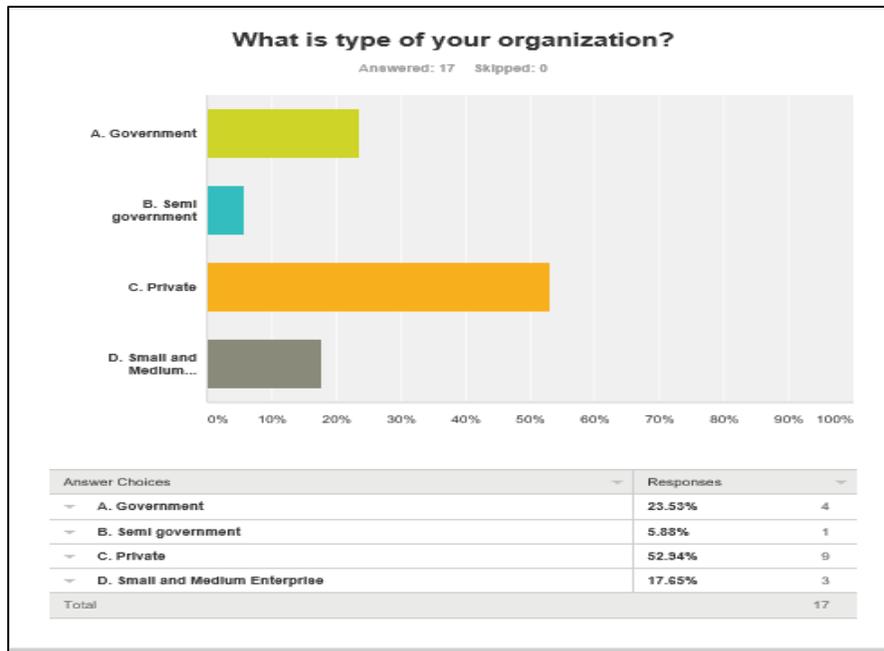


Figure 1. Type of organizations

Subsequently, Figure 2 below presents the experience of the respondents in software development. Most of the respondents have 3 to 5 year experience (29.41%), 5 to 10 years (23.41%), more than 10 years (17.65%), less than 1 year (17.65%) and 1 to 3 years (11.76%). Moreover, Figure 3 depicts the number of employees that involved during software development. The result shows that 41.18% of the respondents claimed that 11 to 25 employees are involved in the software development, 17.65% claimed 1 to 5 employees and 26 to 50 employees. In addition, 11.76% of the respondents claimed that 5 to 10 employees and more than 100 employees are involved in software development. Thus, we can ascertain that all of the respondents have experienced working in a team during software development.

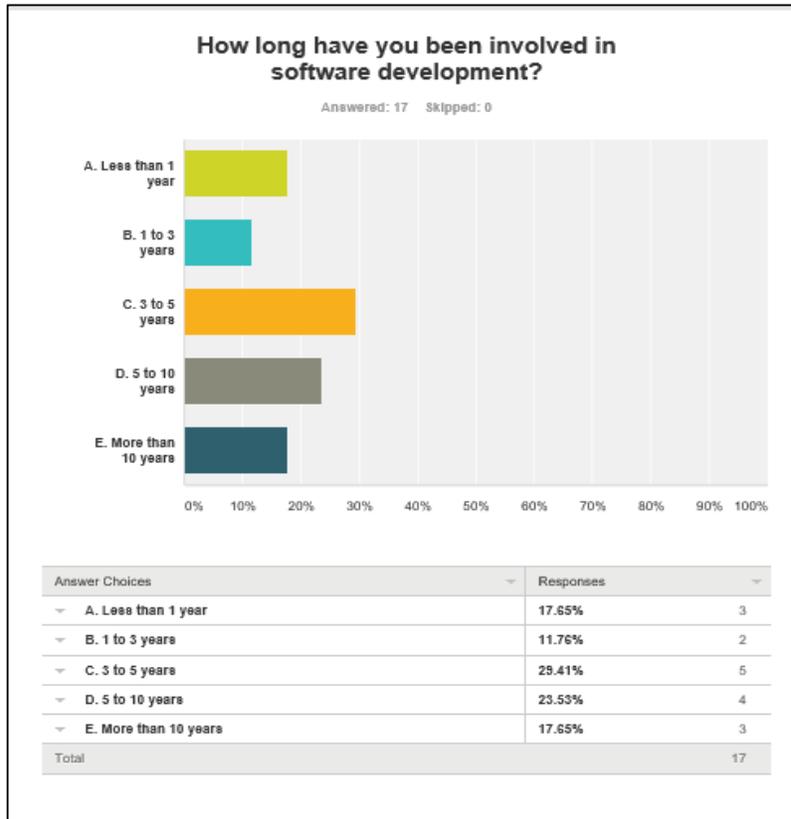


Figure 2. Experience

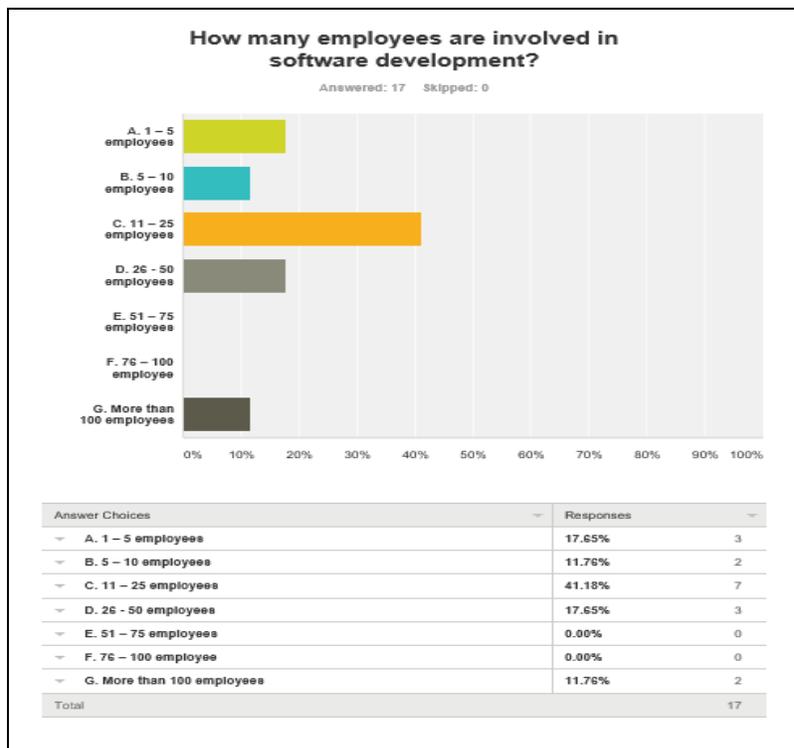


Figure 3. The number of employees

4.2. Software development process

In this survey, we are interested to understand the methodology involved during software development and what the respondents need toward open source tools. In this survey, we asked the respondents if they always use proper methodology during software development process. The result has shown that respondents are equally use and sometime use the proper methodology during software development (41.18%) and 17.65% never use any proper methodology as shown in Figure 4. Thus, we can suggest that there is a need to encourage the software practitioners to use proper methodology in the future.

Furthermore, the survey asked the respondents to identify the most commonly used methodology during software development. The result has shown that most of the respondents always used structured development and object oriented is regularly used during software development as presented in Figure 5.

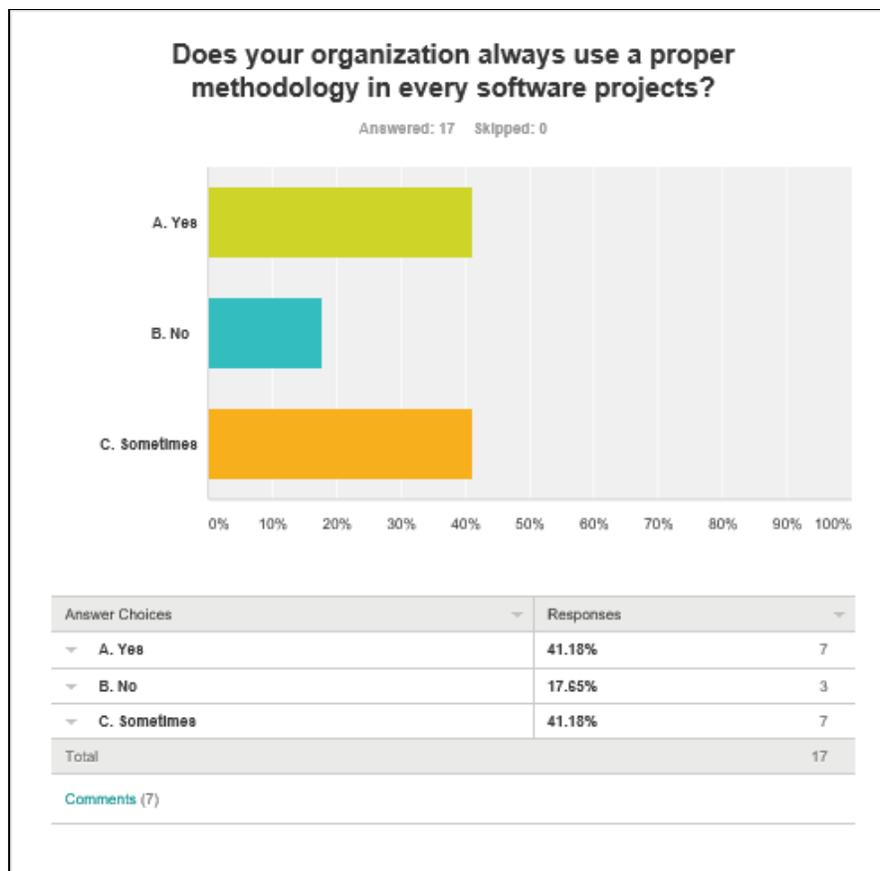


Figure 4. Proper methodology used

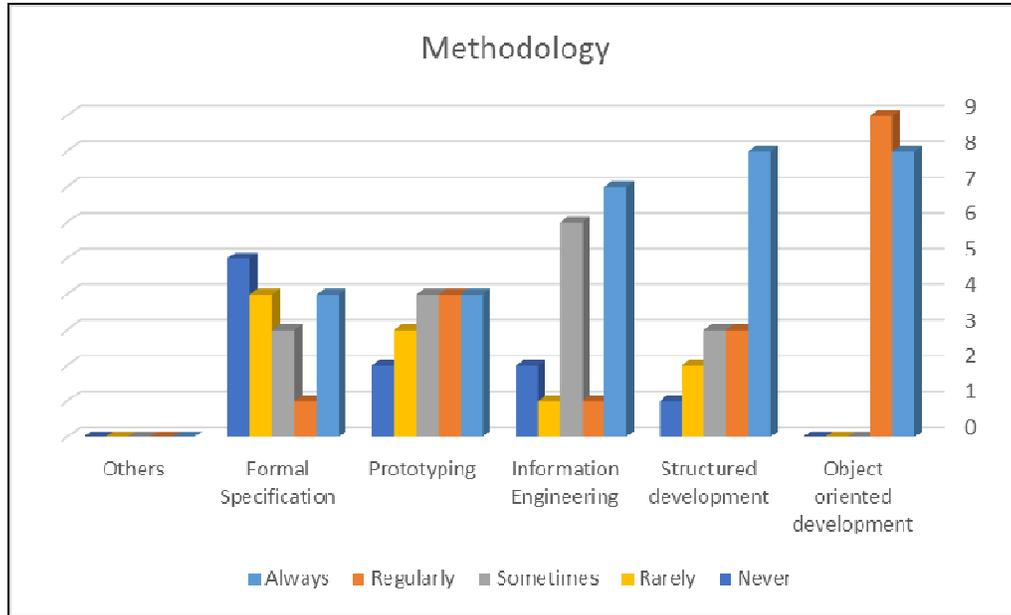


Figure 5. Methodology

In this survey, we also interested to know the requirements management tool that the respondents used in managing their requirements during software development. The result has revealed that most of the respondents never used any requirements management tool. However, during software development 5 respondents used Tracer to manager their requirements as shown in Figure 6.

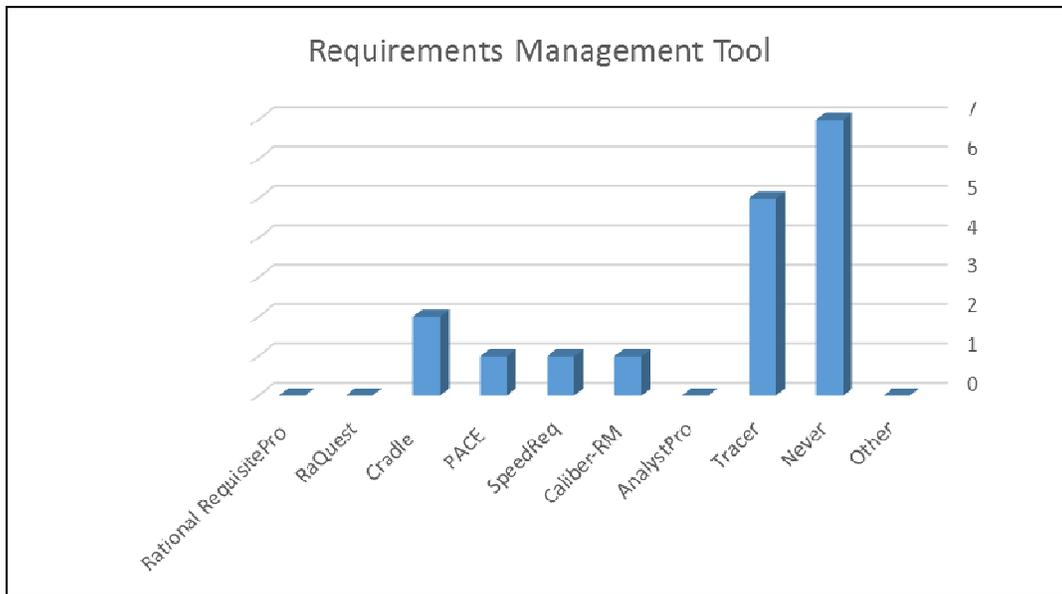


Figure 6. Requirements management tool

Furthermore, the respondents are asked to give their consent regarding the use of an open source requirements management tool in the market. The result has shown that 16 out of 17 respondents agree to use the open source requirements management tool as presented in Figure 7. This is a

good indicator to affirm that an open source requirements management tool is needed in the market. Moreover, the respondents are asked to select features that they are looking forward in an open source requirements management tool. The features are:

- A. The tool should be easy to use. Not too much training and administration needed. The tool should not create additional tasks and deployment should not require extensive customization.
- B. The tool must have tight access control whereby each participant has appropriate access to the data. (Role-based, project-based and task based access control.)
- C. The tool must be adaptable and extensible to the needs of the organization or project.
- D. The tool should be free licensing that allows the user to use the tool in full version without limitation.
- E. The tool should be database centric, but also support document management.

The result is very promising (Figure 8) as all the respondents decided that the tool should have the above features in order for the respondents to employ the tool during software development in the future.

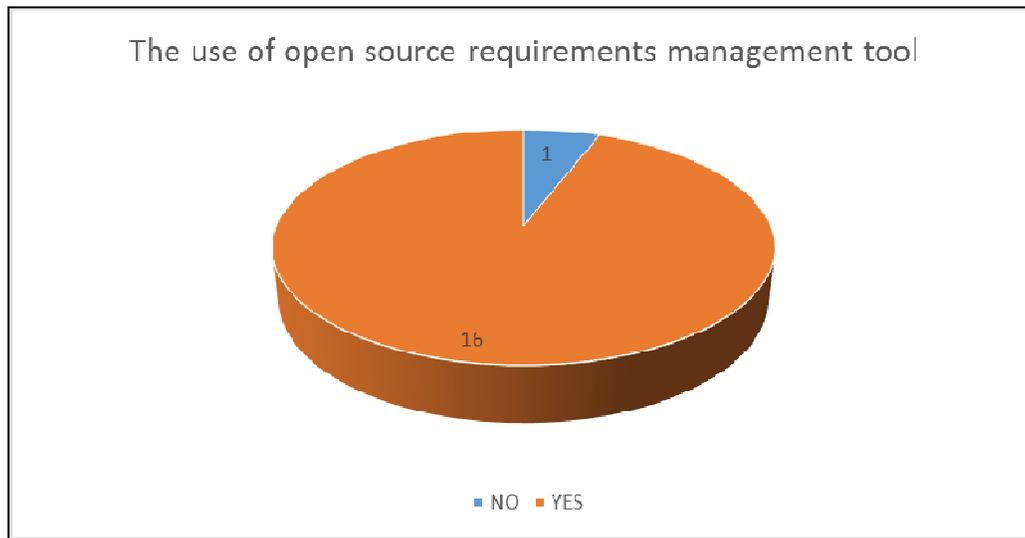


Figure 7. The use of open source requirements management tool

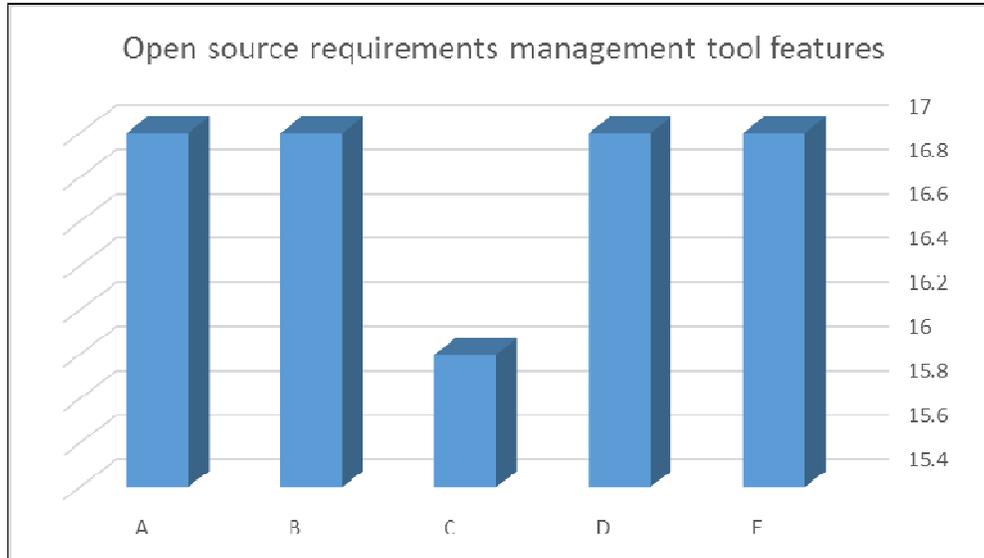


Figure 8. Open source requirements management tool features

4.3. Software requirements practices

In this survey we would like to investigate how the respondents practices their requirements over the entire software development. Thus, we constructed the questionnaire based on the following practices, as described in Table 1.

The result as described in Figure 9 shown that most of the respondents highly practiced the following:

- Uniquely identify each requirements
- Record requirements traceability from original sources
- Define traceability policies
- Maintain traceability manual
- Use a database to manage requirements
- Define change management policies
- Identify global system requirements
- Identify volatile requirements
- Record rejected requirements
- Reuse requirements over different projects

However, the respondents claimed that they are low in defining policies for requirements management policies practice. Thus, it can be seen that the respondents are practising software requirements partially and there is a room for improvement in this area. It is important to encourage the respondents to perform software requirements practices entirely over the software development in order to produce high quality software within the allocation budget and time.

Table 1. Software requirements practices.

No	Questions	Not Applicable	Unknown	Very Low	Low	Average	High	Very High
1	Do you uniquely identify each requirement?	1	2	0	0	0	10	4
2	Do you have defined policies for requirements management?	0	3	0	6	2	1	5
3	Do you record requirements traceability from original sources?	1	1	0	4	5	6	0
4	Do you define traceability policies?	0	1	6	0	3	7	0
5	Do you maintain a traceability manual?	0	0	5	4	1	7	0
6	Do you use a database to manage requirements?	0	4	0	1	2	5	4
7	Do you define change management policies?	0	3	3	2	0	7	0
8	Do you identify global system requirements?	0	2	3	0	3	7	2
9	Do you identify volatile requirements?	0	2	0	1	4	5	5
10	Do you record rejected requirements?	0	2	4	0	5	6	0
11	Do you reuse requirements over different projects?	0	2	2	2	4	7	0

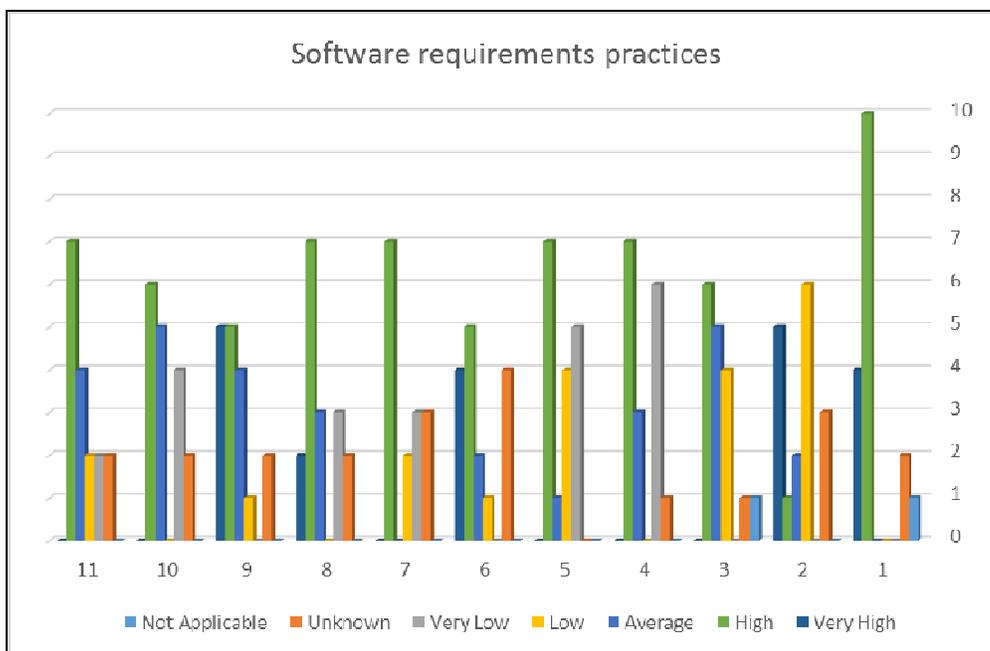


Figure 9. Software requirements practices

3. CONCLUSIONS

As a conclusion, this paper has described the software requirements practices among the software engineers in a study of a city. The result is very promising and it shows that software practitioners are aware as well as practicing good software requirements partially. We have identified that extra attention should be given to requirements management area. The software practitioners are willing to employ open source tool during software development. We are also able to determine the features of an open source tool.

This survey was conducted only in a city of Jeddah and therefore the result of this survey cannot be generalized to represent the current software requirements practices in the Kingdom of Saudi Arabia. However, the result is promising and it is a good indicator for us to conduct a survey in the Kingdom of Saudi Arabia in the future.

In the future, we will conduct a survey of software engineering and requirements engineering practices as well as the software projects success factors in the Kingdom of Saudi Arabia. Thus, this work is an initial work towards the survey of software engineering practices. Generally, this work has contributed to the software engineering body of knowledge and specifically contributed to the requirements engineering research area.

ACKNOWLEDGEMENTS

The authors would like to thank King Abdulaziz University for giving us the chances to complete this work and all the person who have help us to complete this work.

REFERENCES

- [1] A. Zainol and S. Mansoor, "A Survey of Software Engineering Practices in the Software Industry," in *Proceeding of The IASTED International Conference on Software Engineering*, 2009.
- [2] K. Wiegers and J. Beatty, *Software Requirements*. Pearson Education, 2013.
- [3] A. Dennis, H. B. Wixom, and D. Tegarden, *System Analysis and Design with UML Version 2.0: An Object Oriented Approach*, Third. USA: John Wiley & Sons Ltd., 2005.
- [4] IEEE, "IEEE Standard Glossary of Software Engineering Terminology (ANSI)," (1-55937-0(SH13748-NYF)), 1990.
- [5] A. Abran, P. Bourque, R. Dupuis, and J. W. Moore, *Guide to the Software Engineering Body of Knowledge - SWEBOK*. Piscataway, NJ, USA: IEEE Press, 2001.
- [6] P. Roger and B. Maxim, *Software Engineering: A Practitioner's Approach*, 8th Editio. McGraw Hill Education, 2015.
- [7] I. Sommerville, *Software Engineering*, 10th Editi. Edinburgh: Pearson Education, 2015.
- [8] R. B. Grady, "An economic release decision model: Insights into software project management.," in *Proceedings of the Applications of Software measurement Conference*, 1999.
- [9] B. W. Boehm and P. N. Papaccio, "Understanding and Controlling Software Cost," *IEEE Trans. Softw. Eng.*, vol. 14, no. 10, pp. 1462–1476, 1988.
- [10] D. Leffingwell, "Calculating your return on investment from more effective requirements management," *Am. Program.*, vol. 10, no. 4, pp. 13–16, 1997.
- [11] L. J. May, "Major causes of software project failures," *CrossTalk J. Def. Softw. Eng.*, vol. 11, no. 6, pp. 9–12, 1998.
- [12] Javed, Talha, and Q. S. Durrani, "A study to investigate the impact of requirements instability on software defects," *ACM SIGSOFT Softw. Eng. Notes*, vol. 29, no. 3, pp. 1–7, 2004.
- [13] J. Verner, A. Park, K. Cox, A. Park, S. Bleistein, C. Science, and N. Cerpa, "AJIS Vol 13, No. 1 September 2005 REQUIREMENTS ENGINEERING AND SOFTWARE PROJECT SUCCESS: AN INDUSTRIAL SURVEY IN AUSTRALIA AND THE U.S.," *Australas. J. Inf. Syst.*, vol. 13, no. 1, pp. 225–238, 2005.
- [14] A. Zainol and S. Mansoor, "Investigation into requirements management practices in the malaysian software industry," in *Proceedings - International Conference on Computer Science and Software Engineering, CSSE 2008*, 2008, vol. 2, pp. 292–295.
- [15] V. Garousi, A. Co, A. Betin-can, and O. Demirörs, "A Survey of Software Engineering Practices in Turkey," *J. Syst. Softw.*, vol. 108, pp. 148–177, 2015.
- [16] C. P. Team, "CMMI ® for Development , Version 1 . 2," no. August, 2006.
- [17] M. Hoffmann, N. Kuhn, M. Weber, and M. Bittner, "Requirements for requirements management tools," in *Proceedings. 12th Requirements Engineering IEEE International*, 2004.
- [18] D. M. Alghazzawi, S. T. Siddiqui, M. U. Bokhari, and H. S. A. Hamatta, "Selecting Appropriate Requirements Management Tool for Developing Secure Enterprises Software," *Int. J. Inf. Technol. Comput. Sci.*, vol. 4, no. March, pp. 49–55, 2014.
- [19] M. Lang and J. Duggan, "A Tool to Support Collaborative Software Requirements Management," *Requir. Eng.*, vol. 6, no. 3, pp. 161–172, 2001.

Authors

Azida Zainol is an Assistant Professor at Department of Computer Science, King Abdulaziz University, Jeddah. Her research interests are software engineering, requirements engineering, and software modelling and software quality assurance.



Haneen Hassan Al-Ahmadi graduated from Universiti Putra Malaysia in year 2005 with a Master's Degree and PhD in 2016. Her research interests are software engineering and software quality

