THE SOFTWARE REQUIREMENT PROCESS FOR THE SOFTWARE INDUSTRIES

Taimoor Hassan, Muhammad Zulkifl Hasan, Raja Zaka Ullah

Department of Computer Science, LG University, Lahore, Pakistan

KEYWORDS

Requirement Engineering, Natural Language, Requirement Management, Software System

ABSTRACT

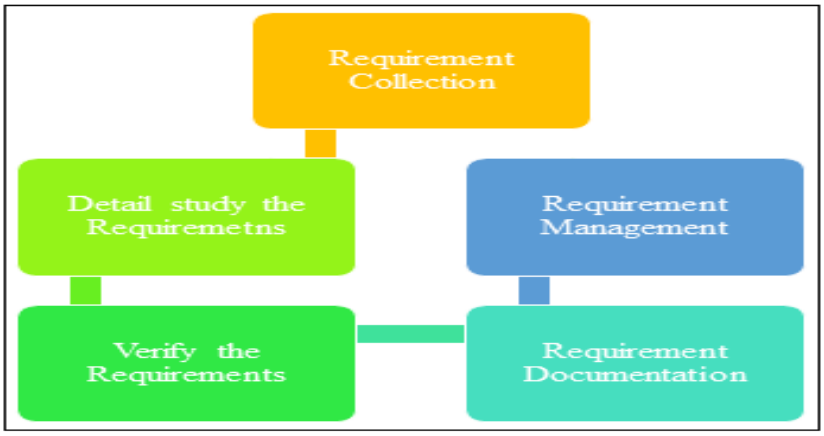
In the software system development, requirement engineering plays a great role because system successfulness depends on it. If the requirements are clear and unambiguous in every aspect, then the system will be developed error-free. A few clients that are not educated and illiterate, then in this scenario, the given requirements are not clear and full of ambiguous. In this paper, we proposed a method to resolve this problem in shape of break the requirements, filter it (separate necessary and un-necessary requirements), break into little bits, numbering them and prioritizing, after it documented the requirement that is very helpful for current and future changing in the software system and finally the shift to the development team for developing project. The researcher aimed to apply this proposed method on case study as experimental to find out the definite and relevant results. Researcher implement this proposed methodology on case study as discussed in results section.

INTRODUCTION

In the classical approach of software development; in the product development of design phase a few requirements are used as input. Within software engineering, the software requirement engineering is a field that maintains the requirements of company holders and solved it with the help of software. A process optimization and development of product (Leveson & Weiss, 2009) for need of specific design, development process and product, functional and physical requirement is an individual document that must be able to the perform. For the design, service or product, material, several requirements specifications must be satisfied to complete the task. A term requirement has been in use in the software engineering (Anwer & Ikram, 2006). Occasionally, the individuals face un-educate clients and customers then we will negotiate with them to understand the requirements in the meaningful manner (Chakraborty, Baowaly, Arefin & Bahar, 2012). Because we know very well that without the clarity of requirements, we will not develop a successful or error-free software/system (Mazón, Pardillo & Trujillo, 2007).

In software development, requirement engineering works like a bridge that is helpful for the construct the project. It explains what customer wants, its needs about the developing software system. In requirement management, many of activities and steps are performed that are very helpful for the project team to identify the requirements their control on proposed software system their track performance and changing that is most possible throughout the project proceeding (Pandey, Suman & Ramani, 2010). Analyzing different methods of software requirement engineering like OMT method, VORD, and Domain Based Approach. Describe the requirement engineering process, starts from the feasibility study to necessitate the documentation (Sadraei, Aurum, Beydoun & Paech, 2007). The framework highlighted the major challenges that include the technology issues and their crises, economical issue, external and internal issue, stakeholders' conflicts with one and another. Overall, requirement management and requirement development are depending on the requirement engineering. If you think that you will develop the software system without the requirement engineering, then the project will lead to the failure (Anwer & Ikram, 2006).

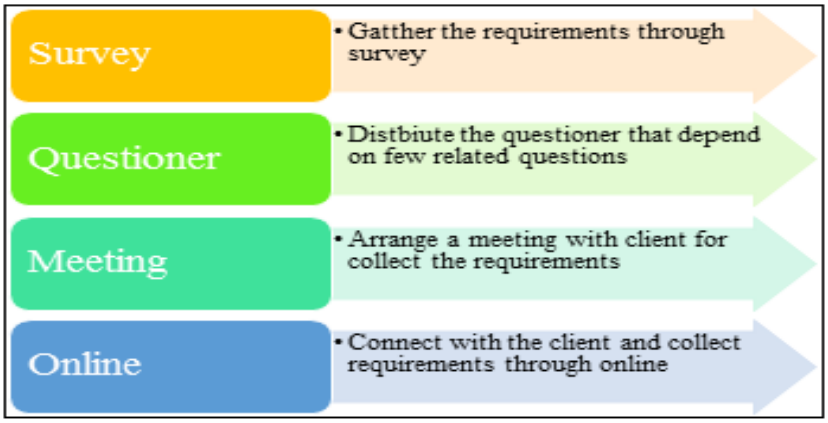
Figure 1 Requirement Engineering Process



Requirement Collection

In the first step of the requirement engineering process is the requirement collection or gathering that is collected from the client in different manners. It is the basic and most important step for the development of the software system. Software companies gather the requirements from different techniques that are described in below figure 2, but these techniques are not limited, many more techniques also use to get the requirements from the phone, courier, fax, email and many more (Lee & Chang, 2006)

Figure 1 Requirement Gathering Techniques



Detail Study the Requirements

After gathering the requirements, in this phase, analyses the requirement in detail form and will try best to fetch out the relevant and accurate requirements. (Mich & Garigliano, 2000) This phase is most important because on this level we understand the requirements that are much helpful to develop the software system.

Verify the Requirements: After Analyzing, verify the requirements and match the actual requirements that come from the client. Validate it with different functionality to ensure that requirements are fully validated and error-free because software system development is based on this validated requirement (Song, 2017) otherwise a few problems arise on the testing and implementation phase.

Requirement Documentation: The final phase of the requirement engineering, all the requirements would be documented that is very beneficial for the software development team. (Jiao & Chen, 2006) If the requirements are not documented, then there are many errors arises in the development phase. Documented requirements also helpful for future changing in the software system.

Requirement Management: It is a set of different activities that help the project team to control, identify and track the requirements because throughout (Roger, 2010) the project proceeding, requirements changing up to compilation of software system, Requirement management overall manage the requirement of the proposed software system.

LITERATURE REVIEW

Several years ago, different scholars and researchers apply different approaches in field of software requirement engineering. Currently, no accurate approach gives exact results. Every approach has different results and their meaning that is totally different from one and other. Use of UML (unified modeling language) diagram, will efficient requirement engineering and their process. Apply the approach (Saeed, Sarwar & Bilal, 2016) on case study (file sharing system) fetch the requirements, divide into the functional and non-functional requirements, drafting it with description and give the priority (high, medium, low), mapping the requirements in pre and post condition in use case scenarios. With the help of this technique, we will achieve the efficient requirements for small and low level of project of diverse companies. In software development life cycle, requirement engineering is most important phase (Chakraborty et., 2012) related to requirement engineering with the help of complex system. Thus, analyzing different methods of software requirement engineering like OMT method, VORD and Domain Based Approach.

Describe the requirement engineering process, starts from the feasibility study to require documentation. LORD method of requirement analysis on the domain component of the hospital (patient, bill, nurse, medicine, doctor) after this evaluate the results in the light of the software development life cycle. In the development of software applications and selection of COTS (customer-off-the-shelf) components face the different challenges of the requirement engineering. Some researchers discuss several problems in the domain of the requirement engineering (Asghar & Umar, 2010). The framework highlighted the major challenges that include technology issues and their crises, economical issue, external and internal issue, stakeholders' conflicts with one and another (Saeed, Sarwar & Bilal, 2016). Also describe the requirement engineering process (specification, validation, analyzing, documentation) functional and non-functional requirements and their usage level in the proposed software system. So, finally, we can say that a lot of research work done in the field of the software requirement engineering and their process.

Problem Statement

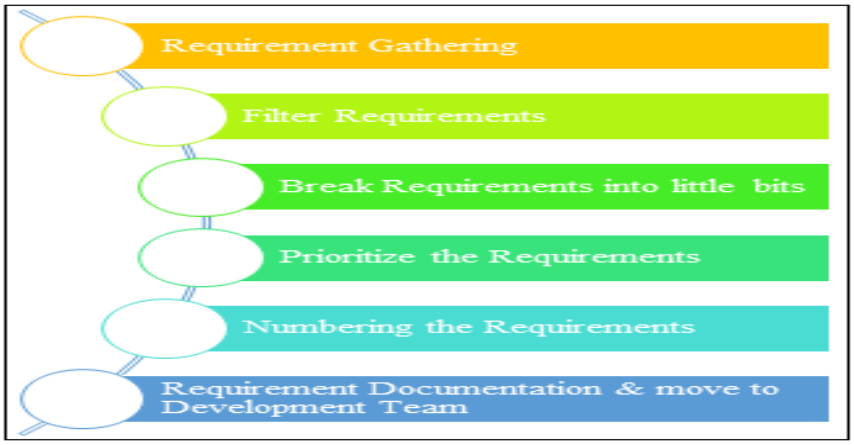
In our software industry, several software projects develop day by day. That is good for industries because it is big investment in our country. Nowadays, almost all organizations use different software system for performing routine office tasks and transactions. So, we can say that we face many problems if we don't use and install the software system in any organization and company (Fung, Ren & Xie, 1996). Software industry are big investors for country and contribute their part in development of our industries. With the passage of time, in the development of software systems, there are many problems that is faced by the company in shape of software requirements. The main issue with this is too illiterate of clients. So, they do not say properly and express requirements about proposed system. Such clients found everywhere to develop the software from the software company. As a result, proposed software will not be very good in every aspect due to the client's unclear requirements. There are many fluctuations that arise in software development because there is much gap discussion and communication gap between the client and the project team. The main reason behind this is the illiterate the client as well as not properly clear the requirements (Damian et al., 2004).

PROPOSED SOLUTION

In this section, the proposed solution of a given problem that is described in section II. Overall description of the proposed solution that how we will develop the software system

error-free and understandable for the client as well as the company's end-user in below figure 3.

Figure 2 Used Approach software system



Requirement Gathering

In first stage of given proposed solution, start with requirement gathering. Requirement gather from different sources or techniques that is describe in section I. We know that the client is not well skill or well educated so, we observe the client's company system as well as their employees and very important the end-user who directly use the proposed system. Also, we meet the company stakeholders and the other employees to exactly find out the requirements of the proposed system (Sadraei et al., 2007).

Filter the Requirements: After requirement gathering, filter requirements. Requirements divide in functional and non-functional form, sorting them in a sequential way and filtered it from fake, ambiguous and irrelevant data. *Break the Requirements*: After filtering the requirements, break requirements in a little bit form, break it in different parts. Due to this requirement specification and analyzing so easy and understandable for the software team. *Prioritize the Requirements*: In this phase, the prioritize requirements based on the functionality. Essential and basic requirements are on top priority and non-essential requirements are on down priority (Svensson & Malmqvist, 2001). Develop software functionality on a priority basis. Client satisfaction is more due to early complete priority requirements.

Numbering the Requirements: The most important step in this approach, allot numbering to little bit prioritize requirements. Normally numbering will be in numbers form like 1,2,3,4,5,6 ……n. With the help of this, we will easily find out the requirements against their number and work on it in the shape of development (Loconsole, 2004). *Requirement Documentation*: In this phase, all requirements will be documented. Requirements will be written that is very helpful to understand the system, track the requirements, all the functionality and their usage and for future (in the case of changing requirement) After this, document forward to the development team for developing the software system. All processing must be documented because, after the project complete, the documented file sends to the client for understanding all the stages and its usage and future purpose (Fuggetta, A., 2000).

RESULTS OF STUDY

We use the case study (Hotel Management System) to implement the proposed solution to find out the relevant results. *Requirement Gathering*: First, in this case, study, we collect the total 12 requirements that are available in given below table 1.

Table 1 HMS Requirement Gathering

Hotel Management System		Hotel Management System	
Requirements	User login and password	Requirements	Time scheduling
	Book room on ID card		Room cleaning staff
	Check authentication with NADRA		Booking confirmation message/email
	Payment through cash/credit card		Automated generated slip
	Bank validation		Food facility record
	External reservation guest		Check out/ Check-in

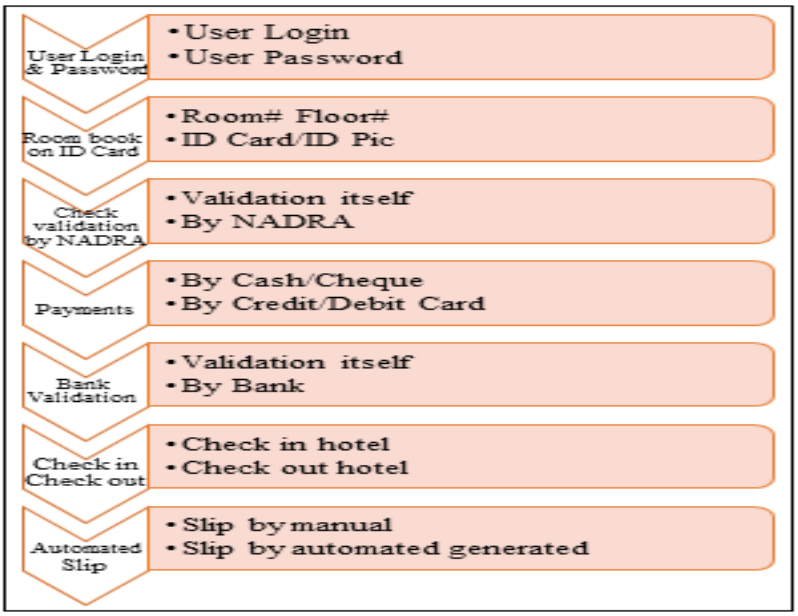
Filter Requirements: We have a total of 12 requirements, but after filtering requirements we erase irrelevant data and divide it into functional and non-functional requirements of the Hotel Management System that is given below in table 2.

Table 2 HMS Functional/Non-Functional

Functional Requirements	Non-Functional Requirements
User login and password	External reservation guest
Book room on ID card	Booking confirmation message/email
Check authentication with NADRA	Room cleaning staff
Payment through cash/credit card	Time scheduling
Bank validation	Food facility record
Check out/ Check-in	-----
Automated generated slip	-----

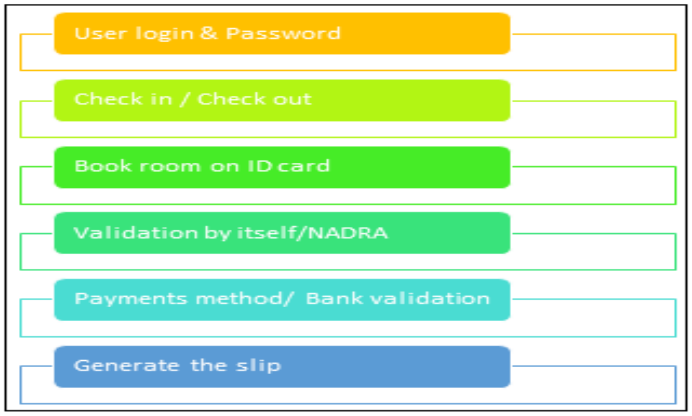
Break Requirements: In this step, we break all the requirements into little bits that are shown in the above table 2. But our major focus on the functional requirements (Miron & Formoso, 2003).

Figure 3 Break HMS Requirements



Prioritize the Requirements: In this stage of proposed methodology, we will be prioritizing the 7 functional requirements of HMS (Hotel Management System) that is given above table 2.

Figure 4 Prioritize Requirements



Numbering the Requirements: In the almost last stage of the proposed system that we are numbering the functional requirements of the Hotel Management System that is given below in table 3.

Table 3: HMS Numbering the Requirements

Number Allocation	Requirements
1	User Login/Password
2	Check-in/ Check out
3	Book the room
4	Validation by NADRA
5	Payments/Bank checking
6	Generate Slip

Requirement Documentation: At this level, finally, we will document all the requirements that are given in table 3 and shift to development team of the Hotel Management software system project.

CONCLUSION

In this paper, we discuss the overall requirement engineering process and different levels. We see in our society, several software companies working to develop a different software system that is very helpful for organizations. In some cases, clients are not well-educated, well-skilled, and illiterate. So, they don't express clear requirements in front of software team because they don't know what they want with the system. We present a definite requirement base approach, that describe the clarity of the understandable requirements. Firstly, we are gathering the requirements, filtering the requirements, break it into little relevant bits, prioritize it, numbering it and at end all requirements would be documented and shift to the development team of software system for the construct the project. We implement this proposed methodology on case study (Hotel Management System) that is discussed in results section.

REFERENCES

Anwer, S., & Ikram, N. (2006). Goal oriented requirement engineering: A critical study of techniques. In 2006 13th Asia Pacific Software Engineering Conference (APSEC'06) (pp. 121-130). IEEE.

Asghar, S., & Umar, M. (2010). Requirement engineering challenges in development of software applications and selection of customer-off-the-shelf components. *International Journal of Software Engineering*, 1(1), 32-50.

Chakraborty, A., Baowaly, M. K., Arefin, A., & Bahar, A. N. (2012). The role of requirement engineering in software development life cycle. *Journal of emerging trends in computing and information sciences*, 3(5), 723-729.

Damian, D., Zowghi, D., Vaidyanathasamy, L., & Pal, Y. (2004). An industrial case study of immediate benefits of requirements engineering process improvement at the Australian Center for Unisys Software. Empirical Software Engineering, 9(1-2), 45-75.

Fuggetta, A. (2000). Software process: a roadmap. In Proceedings of the Conference on the Future of Software Engineering (pp. 25-34). ACM.

Fung, R. Y., Ren, S., & Xie, J. (1996). The prioritisation of attributes in customer requirement management. In 1996 IEEE International Conference on Systems, Man and Cybernetics. *Information Intelligence and Systems*, 2, 953-958).

Jiao, J., & Chen, C. H. (2006). Customer requirement management in product development: a review of research issues. *Concurrent Engineering*, 14(3), 173-185.

Lee, M. C., & Chang, T. (2006). Applying TQM, CMM and ISO 9001 in knowledge management for software development process improvement. *IJSS*, 2(1), 101-115.

Leveson, N. G., & Weiss, K. A. (2009). Software System Safety. In Safety Design for Space Systems (pp. 475-505). Butterworth-Heinemann.

Loconsole, A. (2004). The empirical studies on requirement management measures. In Proceedings. 26th International Conference on Software Engineering (pp. 42-44). IEEE.

Mazón, J. N., Pardillo, J., & Trujillo, J. (2007). A model-driven goal-oriented requirement engineering approach for data warehouses. In International Conference on Conceptual Modeling (pp. 255-264). Springer, Berlin, Heidelberg.

Mich, L., & Garigliano, R. (2000). The ambiguity measures in requirement engineering. *In International Conference on Software Theory and Practice*. ICS.

Miron, L. I., & Formoso, C. T. (2003). Client requirement management in building projects. In 11th Annual Conference of the International Group for Lean Construction, Blacksburg, Virginia, USA, Virginia Polytechnic Institute and State University.

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

Pressman, Roger. S. (2010). The Software Engineering. A Practitioner's Approach, 120-124.

Sadraei, E., Aurum, A., Beydoun, G., & Paech, B. (2007). A field study of the requirements engineering practice in Australian software industry. Requirements Engineering, 12(3), 145-162.

Saeed, M. S., Sarwar, N., & Bilal, M. (2016). Efficient requirement engineering for small scale project by using UML. In 2016 Sixth International Conference on Innovative Computing Technology (INTECH) (pp. 662-666). IEEE.

Song, W. (2017). Requirement management for product-service systems: Status review and future trends. *Computers in Industry*, 85, 11-22.

Svensson, D., & Malmqvist, J. (2001). Integration of requirement management and product data management systems. *DETC*, Pittsburgh, PA.