Functional & Non-Functional Requirement Elicitation and Risk Assessment for Agile Processes

Pooja Sharma* and **Saru Dhir***

ABSTRACT

Requirement analyses have a deep impact on software engineering. Requirement as the initial phase in the software development is a continuous client-centric process for the successful Agile software implementation. System Requirements for implementation are classified as Functional and Non-functional requirement. This paper represents different Function and Non-functional requirement elicitation processes which defines the importance of non-functional requirement as prior as functional requirement for enhanced system quality and customer satisfaction. It is also defined that how various requirements are elicited and non-functional requirements are added to the system. A survey is also conducted which tries to identify the impact of requirement elicitation risks in the system implementation.

Keywords: Functional requirements, Non-functional requirements, Requirement elicitation, Requirement risks, Risk impact.

I. INTRODUCTION

Requirement gathering is a generic phase in any software development. In the development of a software project, it is necessary to define that what developer plan to develop. Organizations are adapting different methodologies for the successful implementation of project [3]. Different stakeholders may have different view about the software to be developed and their requirements may vary for the software. There exist various ways under requirement elicitation method to gather and analyze the requirement needs [1]. While gathering requirements for development team members assumptions cannot made that what is to be developed, the requirements of the software and its development will be in accordance to customer's need, end user's necessity and interest of all the stakeholder's. Requirement analysis and specification may appear in a simple task, but this appearance is very deceiving. In software development requirement has to be very clear. The nature of requirement has to be unambiguous, understandable and very much feasible, for successful implementation of any software project [6].

The initial phase of requirement gathering has to be successfully implemented. It is also possible that requirement may change with time, so it is necessary to follow such a development method that can easily adapt the changing needs of the customer irrespective of the development. Because of adapting the customer's changing needs Agile software development methodology have gained relevant impact in the software industry [9] in the recent past. Agile methodology follows an iterative and incremental approach of development and due to it non-linear development approach changes are easily incorporated even late in the development cycle. This method also allows to backtrack and make changes in the system. While gathering requirements elicitation techniques should be properly practices from all the viewpoints of software requirement. For a successful project implementation, there should not be any requirement issued left

---

*   Amity University Uttar Pradesh, India, *E-mail: poosharma7448@gmal.com*

**   Amity University Uttar Pradesh, India, *E-mail: sdhir@amity.edu*

without discussion. This also helps in gaining market value and customer satisfaction. Fig 1. describes the various requirements elicitation techniques through which requirements for the software can be gathered. The collected requirements are analyzed from user point of view. These requirements are on the basis of what the end user needs and market demand.
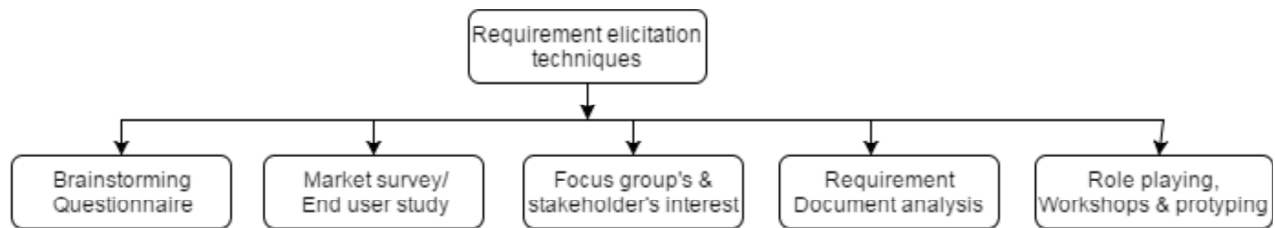

Figure 1: Requirement elicitation Techniques

In this phase, gathering and analysis phase collecting requirements are very essential part. This is achieved by market survey, understanding the end user's need and interviewing all the stakeholders (people who have an interest in the project). The rest of the paper is as follows: Section-II describes functional vs non-functional requirements. Section-III gives an overview of the existing Literature review. Section IV represents the methodology for requirement gathering. Section V elucidates the risk in requirement elicitation.

II. FUNCTIONAL VS. NON-FUNCTIONAL REQUIREMENT

Requirements gives the idea about 'what to do' in the software. During the requirements elicitation phase in agile development methodology while the development team is busy in gathering it is necessary that the functional requirements and the non-functional requirements are adequately identified. Functional requirements are defined as the basic needs that are actually needed for the on the ground implementation of the software project. These requirements are the legal or regulatory requirements of software which includes the Business requirement, external interfaces, certification requirement, etc. Functional requirements define the behavior with their functionality in the system. These requirements define the working of the system, that 'what a system supposes to do.

Non-functional defines the quality attributes of the system. These NFR represents that 'how a system supposes to do'. Though such requirements are not documented for the system need, yet it is necessary to have NFR's for successful implementation of the system. But it is found that while functional working of the system is captured in the system non-functional requirements are not properly incorporated. Non-functional requirements are often defined as the ilities like usability, reliability, availability, reusability and interoperability. For the implementation of a successful software project, both functional and non-functional requirement has to be given equal importance. Functional and non-functional requirement is complementary to each other. While developing a software project, ignoring either of them may result in faulty system implementation. During the development frequent change requirement is also a challenging task [11].

III. LITERATURE REVIEW

There exists different work done by the researchers on requirement elicitation, prioritization and analysis. Traditionally Software Requirement Specification (SRS) document was prepared in the industry, but in Agile methodology, there is no comprehensive documentation is prepared as requirements keep on changing with time. The researcher has used various requirement analysis techniques. Table 1. describes some of the studies conducted by different researchers on software requirement from the digital library IEEE. The ground study represents the idea of requiring prioritization and elicitation of the requirement to maintain the user stories in the number iteration (Sprint cycle). The study also encounters the proposed articrafts that how functional and non-functional requirements can be modeled adequately in the system development.

Table 1
Related studies

Year	Author	Work done
2005	Ronit Ankori [10]	This study shows automatic way of retrieving functional requirement from stakeholders through a machine learning system.
2008	Malik Qasaimeh *et al.* [2]	The paper analyzes various processes under the umbrella of agile based upon software project development requirement.
2010	Zornitza Racheva *et al.* [12]	This study focuses more on the client and carries out an on-ground study for agile requirement prioritization.
2012	Donna D. Gregorio [4]	This study uses a business analyst to elicitate requirement and uses a sprint cycle to maintain user stories.
2012	Weam M. Farid *et al.* [5]	The study proposes three fundamental artifacts (AUCs), (ALCs) and (ACCs) for adequate modeling of Functional and Non-functional requirement in the system.
2012	Weam M. Farid *et al.* [13]	This research proposes a java-based NORMATIC tool for incorporation of the non-functional requirement in the system.
2013	Weam M. Farid *et al.* [7]	The study proposes a quality metrics NORPLAN (Non-functional requirements planning) that will use risk-driven algorithm for improved requirement planning.
2015	Darshan Domah *et al.* [8]	This study proposes NERV (Non-functional Elicitation Reasoning and Validation) methodology for defining Non-functional requirements in Agile software development.

IV. METHODOLOGY

In software development there are different phases to pass through the complete process of achieving a deliverable product. Initially the client defines the requirement of the required software, these requirements are documented and then, according to the achieved requirements software is successfully implemented and delivered [13]. In an Agile methodology during the whole implementation, there is a continuous interaction between client and development team members. There is no such requirement document is prepared. In Agile the developed method is adaptable and it regularly updates the changing requirements of the customer into the system. It is usually a critical task to set and update the changing requirement. In such case requirement engineering has to be properly carried out and managed.
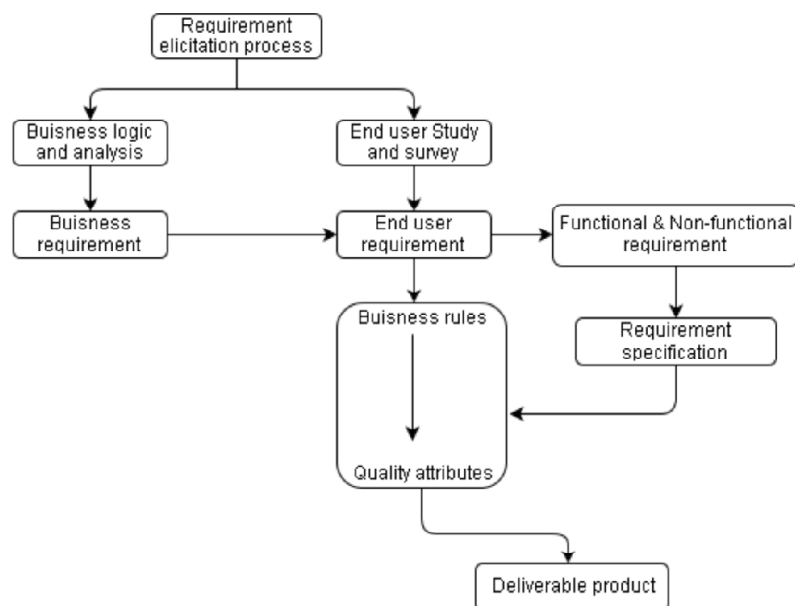

Figure 2: Requirement levels for final deliverable product

Traditionally for the development of any software project requirements are elicited from the respective end user and client. In the current trend of Agile software development requirements are gathered and stored in the form of user stories and requirement backlogs are prepared. From these user stories requirements can be taken from the backlogs if any similar project is encountered in the fore coming future. The main goal of the software development is to deliver a successfully well implemented software project/product.

Here, Fig 2. defines a complete process from requirement elicitation to a final deliverable product. It describes that how business requirements are obtained from business logic and end user requirements are collected with the help of surveys and studies conducted on the end user's necessity. Functional and Non-functional requirements are combined with these requirements on which business rules are applied so as to meet standards in the business market. Which along with quality attributes together delivers a successfully implemented product.

Requirement Trigger card
Requirement Elicitation
1.   Who are the end users?
2.   What an end user needs
3.   What are the business logics?
4.   What is requirements specification available?
FR and NFR specification
1.   How NFR can be incorporated in the project?
2.   Where NFR are needed to importantly addressed?
3.   What quality attributes are needed in the system?
Requirement Validation
1.   What are validation criteria for the requirements?
2.   How qualitative Specified requirements are?
3.   When customer should accept the requirement after verification?
Risk Planning
1.   What is a possible risk in the requirements?
2.   How risks are identified and addressed in risk identification?
3.   What are the major risks?
4.   How risk degrade system quality?

Figure 3: Requirement Trigger Card

The requirements cannot directly taken to the implementation of the software. It is likely to occur some uncertainty in the requirement elicitation process. There are steps to be followed in requirement before implementation Fig 3. (Requirement Trigger Card) defines the entire process requirement elicitation to planning which leads to the implementation. In elicitation target user is identified and their needs are gathered. It also tries to under the market with respect to the software that is supposed to be developed. The next step in the requirement specifies the functional and non-functional requirement so that NFR's can be correctly addressed at right phase in the development process.

V. RISK

Risk in software development and requirement elicitation can be defined as the uncertainty in the process of requirement gathering that may cause potential failure in software development. It is possible that some of the requirements are being misunderstood due to which wrong documentation is made which finally results in a faulty system implementation. A successful system implementation ensures customer satisfaction and high degree of risk management so as to prevent faulty system implementation. Though the Agile methodology starts testing at a very early stage yet some of the risks went undiscovered while gathering requirement.

Our study tries to identify the most likely occurred risks and their impact in the successful system implementation. Table II lists identified risks in requirement gathering phase.

Table 2
Identified Risks

| S.No	Notations	Identified Risks
1.	R1	Requirements being misunderstood
2.	R2	Insufficient time allocated to requirement gathering
3.	R3	Poorly articulated requirements
4.	R4	Lack of communication
5.	R5	Wrong identification of functional and non-functional requirements
6.	R6	No formal requirement documentation
7.	R7	Wrong elicitation technique followed
8.	R8	Lack of knowledge about business needs
9.	R9	Wrong target user
10.	R10	Lack of knowledge about requirement elicitation techniques
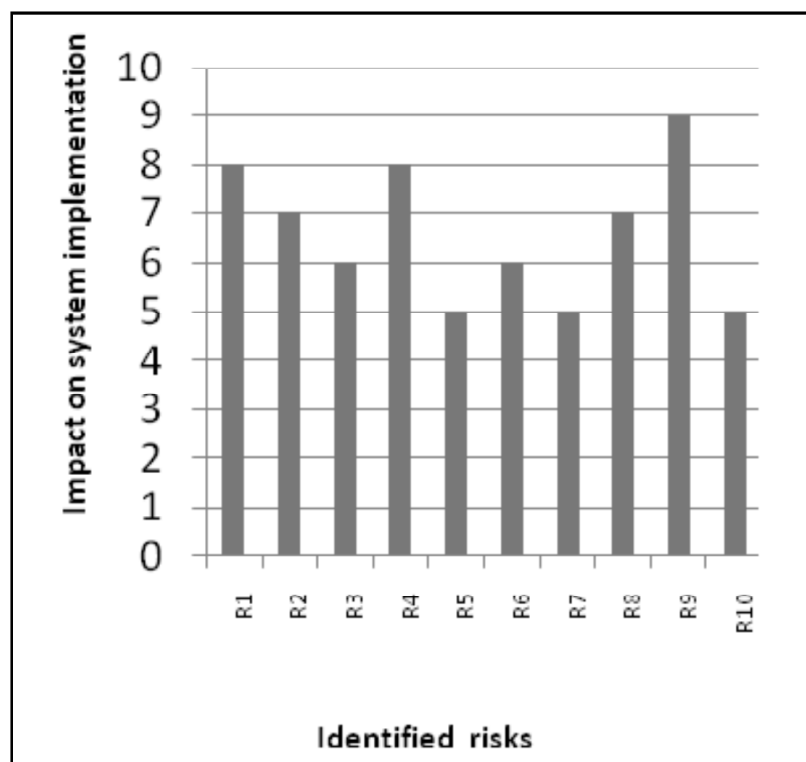
Figure 4: Impact risks in system implementation

In requirement elicitation various risks are identified. These risks are then surveyed to find the impact of these on a successful system implementation. Here, Fig 4. defines the impact of the identified risks on the successful system implementation. The rankings for the impact of risks are identified on the basis of the survey and expert advice is also incorporated in the same.

VI. CONCLUSION

Requirement is a critical phase in software development. The paper concludes the importance of the Functional and Non-functional in Agile implementation which also helps in the successful system

implementation. Here, various requirements elicitation techniques are studied and how requirements are elicited from various stakeholders, end user and client.

The study also identifies the risks in the requirement phase and these are ranked on a scale of 1-10 upto extent they leaves the impact on the system implementation. These rankings for the identified risks are obtained from the survey conducted from the teams which are into the industry for requirement gathering and thoroughly studies the requirement elicitation techniques.

REFERENCE

[1] Dr. Deepak Kumar, Saru Dhir, "A Role of Non-Functional computing in software Engineering", ICRITO 2013 (Co-sponsored by IEEE Computer Society), at Amity Institute of Information and Technology on 29th -30th Jan.,2013, page 133.

[2] Malik Qasaimeh, Hossein Mehrfard, Abdelwahab Hamou-Lhadj, "Comparing Agile Software Processes Based on the Software Development Project Requirements", CIMCA, IAWTIC, ISE, IEEE, 2008.

[3] Preeti Rai, Saru Dhir, "Impact of Different Methodologies in Software Development Process", in (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (2), Mar-April 2014, ISSN: 0975-9646, page: 1112-1116.

[4] Donna D. Gregorio "How the Business Analyst Supports and Encourages Collaboration on Agile Projects", IEEE, 2012.

[5] Weam M. Farid, Frank J. Mitropoulos, "Novel Lightweight Engineering Artifacts for Modeling Non-functional Requirements in Agile Processes", IEEE, 2012.

[6] Saru Dhir, Deepak Kumar, V.B. Singh, "Success and Failure Factors that Impact on Project Implementation using Agile Software Development Methodology", CSI 2015.

[7] Weam M. Farid, Frank J. Mitropoulos, "NORPLAN: Non-functional Requirements Planning for Agile Processes", IEEE, 2013.

[8] Darshan Domah, Frank J. Mitropoulos, "The NERV Methodology: A Lightweight Process for Addressing Non-functional Requirements in Agile Software Development", IEEE, 2015.

[9] Heinecke, H., Noak, C., Schweizer, D., "Constructing Agile Software Processes", Third International Conference on extreme Programming and Agile Processes in Software Engineering, 2002.

[10] Ronit Ankori, "Automatic Requirements Elicitation in Agile Processes", Proceedings of the IEEE International Conference on Software - Science, Technology & Engineering, IEEE, 2005.

[11] Chandrika Sikka, Saru Dhir, Madhurima Hooda, "A survey on challenges in Software Development during the Adoption of Agile Environment", Innovations in Computer Science and Engineering, published proceeding in Advances in Intelligent systems and computing, Springer journal volume 413, Page No. 219.

[12] Zomitza Racheva, Maya Daneva,, Andrea Herrmann, Roel J. Wieringa, "A Conceptual Model and Process for Client-driven Agile Requirements Prioritization", IEEE, 2010.

[13] Weam M. Farid, Frank J. Mitropoulos, "NORMATIC: A Visual Tool for Modeling Non-functional Requirements in Agile Processes", IEEE, 2012.

[14] Saru Dhir, Deepak Kumar, "Agile Software Development in Defiance of Customary Software Development Process: A Valuation of Prevalence's and Challenges", Advance Science letters, vol. 21, pp: 3554-3558.

[15] Saru Dhir, Deepak Kumar and V.B. Singh, "An Estimation Technique in Agile Archetype using Story Points and Function Point Analysis", International Journal of Process Management and Benchmarking, 2016.

[16] Saru Dhir, Deepak Kumar and V.B. Singh, "Success and Failure Factors that Impact on Project Implementation using Agile Software Development Methodology", CSI 2015.