

Modeling of Nonfunctional Requirements for Agile Development Processes

**Trupti Suryawanshi,
MTECH, Computer Engineering,
BVDUCOE, Pune, India**

**Gauri Rao,
Associate Professor, Computer Engineering,
BVDUCOE,Pune, India**

Abstract: Agile method means deliver quickly, modify quickly and change frequently. Main purpose of designing agile software development process is to solve the difficulty of delivering high quality software on time and in under continuously changing requirements as per business need. Now days various agile software development process (Scrum, Extreme Programming (XP), Feature Driven Development (FDD) etc.), used to deliver quality functional requirements (FR). In software development process engineers are mainly focused only on functional requirements under the stress of deploying the software within a time. But Non Functional Requirement (NFR) is also important element of the development process. To develop quality software NFRs are equally responsible as FRs. NFR plays very sensitive role to deliver quality software products. Most requirement modeling methods are mainly concentrated on FRs where as NFRs received less attention. Discovering NFR is not easy job. There is no proper NFR elicitation method is available. In this paper we design a tool which is used to generate NFRs such as security, performance, scalability etc. This tool helps to the project manager and scrum team in risk evaluations, planning and visualization of the proposed plan.

Keywords- FR, NFR, Requirement elicitation, Scrum, Sprint

I. INTRODUCTION

Requirement Engineering is one of the most important areas of software engineering. Each software is successful if and only if it can fulfill all the requirements of customer. To develop any software first stage is requirement gathering and analyzing which plays important role. Requirements are classify into two types. First is the Functional Requirements and second one is Non Functional Requirements. Functional Requirements gives expected functioning of the system such as internal and external visible output of the system. On other side Non Functional Requirements gives detail constraints and control to the system [7]. NFRs are also known as quality attributes of the system.

Agile software development is becoming more and more popular in successful implementing and quickly delivering quality functional requirements. But academic and industrial researchers agree that NFRs have been ignored in software development process and mainly in agile methodologies [3]. FRs is easily collected from stakeholders but discovering NFRs is not an easy and straightforward job. There is no appropriate NFR generation method is available. Few methods have been proposed with some advantages and its disadvantages. But still there is no any standard method defined by requirement engineering [13].

Now a day's most of the researchers are attracted towards the developing NFR generating tools. In this paper we propose a tool which easily focused on NFR elicitation. This tool helps agile development team modeling NFRs early on during requirement gathering as well as in analysis phases. This tool integrates FRs and NFRs modeling under one tool.

This paper presents a tool of modeling nonfunctional requirements for agile development process. Modeling of NFRs with FRs produces effective software. The paper organized as follows. Section II presents a background of our concept. Section III gives the overview and objectives of tool. Section IV describes brief analysis of the tool and how the system is designed with proper snapshots of our results. Section V discusses the conclusion and observation.

II. BACKGROUND

A. Agile Development Process:

Agile development software is allows to project team and to project developer for easily handle changes in requirements, at any stage of the schedule in development process [12]. Agile software development methods are very popular due to its features. Quickly delivery, Simplicity, Face to face communication between stakeholders, Continuous awareness to technical superiority, constantly focus on excellent plan and his own selected team is some well-known features of agile development process [3].

Most common agile methods [6] are listed below:

- Extreme Programming (XP)
- Agile modelling (AM)
- Scrum
- Feature Driven Development (FDD)
- Dynamic Systems Development Method (DSDM)
- Adaptive Software Development (ASD)
- Crystal Methodologies (CM)

B. Non Functional Requirements:

NFR is defined as the software requirement which does not describe what actually software will do but how the software will do it [13]. NFR is the non behavioral requirement of the system. It defines quality and constraints of the system.

Examples of NFRs:

Performance requirement is related with resource required, response time, throughput and anything else connected with performance.

Security is related with protection of the system or software such as authentication, authorization etc.

Portability is nothing but the ability to run software on different platforms.

Reliability is related with the ability of system to execute its proper functions and operations in its defined environment without experiencing failure or system crash.

Usability means ease of use or user friendly software.

Scalability requirement is the capability of software to change with the business environment.

III. PROPOSED WORK

Requirement Elicitation [14] is nothing but requirement gathering or discovering. Requirement elicitation is most important knowledge intensive activity in software development. This is one important job of requirement engineering. FR elicitation is easily handles by considering the requirements of stakeholders. But elicitation of NFR is not that much easy task. NFRs are prioritized in stockholders point of view so the elicitation process is closely related with stakeholder's consideration.

NFR elicitation is crucial issue in software development process [7]. There is no standard method is available for NFR elicitation. Recently several elicitation techniques are introduced with a few strengths and drawbacks. But yet not any method is agreed by requirement engineering team. There are standard definition defined for FR but no proper definition is available for NFR. To properly define NFR is complex because NFRs show inconsistency in different conditions. Sometimes integrating NFRs with FRs creates conflicts among stakeholders.

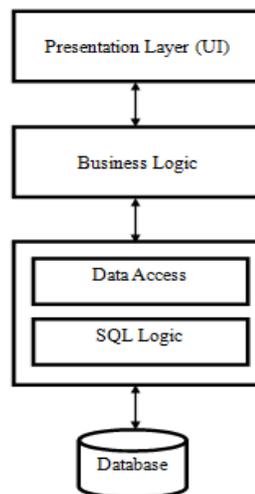
The proposed system has many features which help agile team in modeling NFRs. Rapid identification and linking with FRs will turn out effective software.

IV. ANALYSIS

A. Design

Our model is based on three tier architecture. Three tier architecture means separating project in three layers User Interface, Business Logic and Data Layer where we separate logic, data and user interface in three partition. It means if we replace SQL server database with oracle database then we need to change only data layer Or if we change user screen from windows application to website then we need to change only our user interface layer rest the whole things remain the same.

Fig.1 Three Tier System Architecture



Presentation Layer/ UI Layer: This layer of the architecture gives user interface where the user performs their activity. For example, login form, sign up or any other form fill by the users. Objective of this layer is the user's input validation and the rules for processing.

Business Logic: Most of the business operations are done here. For example, after collecting data, use our business laws and do the validation. Main purpose of this layer is defining classes and business components.

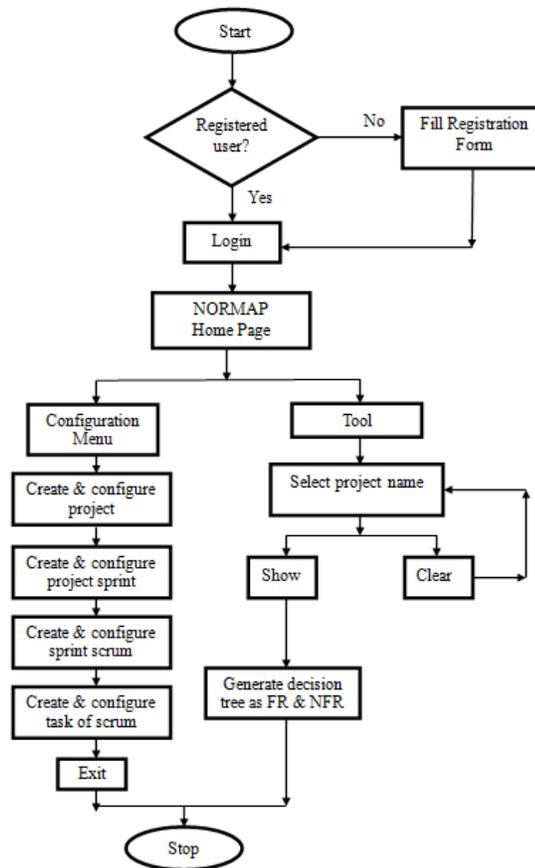
Data Access Layer: Each and every code which is connected with database and the things related with database done at the Data Access Layer. At this layer we define methods which gives help to the Business Layer for connection of the data and execute required actions, whether to return data or to control data (insert, update, delete and so on).

B. Features

1. Combine FRs with NFRs under single tool
2. Helps in risk evaluation process
3. Scheduling, designing and visualization of the future plan.

C. System Workflow

Fig.2 Flowchart of proposed system



D. System Description

The proposed approach gives two phases.

1. Configuration phase
2. Generate Decision Tree
 1. Configuration phase

In this phase we provide different configurations. This configuration process is going through four different configuration phases.

- a) *Project configuration:* At this stage we can add, delete or modify any information of project. This information includes project code, project name, description, time required

to complete project, start date, end date, release date etc. This information is required for decision tree generation.

- b) *Task configuration*: Here we can configure different task of project. It includes project name, type, task name, priority, code, duration, start and end date etc. This information is used in scrum and sprint configuration.
- c) *Sprint configuration*: At this stage we can configure sprint details. Sprint details include same fields as in task configuration. Each sprint has duration like ½ month. The sprint is used for scrum creation.
- d) *Scrum configuration*: Now here we can configure the scrum under sprint. Each scrum has duration like ½ weeks. The scrum is used for task tracking.

2. Generate Decision Tree:

For decision tree generation user needs to select the project. All sprint, scrum, task are categorized in two sections as FRs and NFRs.

Each node of decision tree holds below listed information:

- Project/sprint/scrum/task code
- Number of days complete
- Number of hours complete

Generated result helps to project manager for risk driven evaluations. It also useful for scrum and project manager to visualize and analyze proposed plan of project.

E. Algorithm

Input: Configuration details

Output: Decision tree of FR and NFR

Algorithm:

Begin

Registered user will login.

Click on Configuration and select Project Configuration.

Set project configuration details

Click on Configuration and select Task Configuration

Set task configuration details

Click on Configuration and select Sprint Configuration

Set sprint configuration details

Click on Configuration and select Scrum Configuration

Set scrum configuration details

Click on tools and select generate tree

Select Project Name

Click on Show button

Display project, sprint, scrum and task categorized into FR and NFR

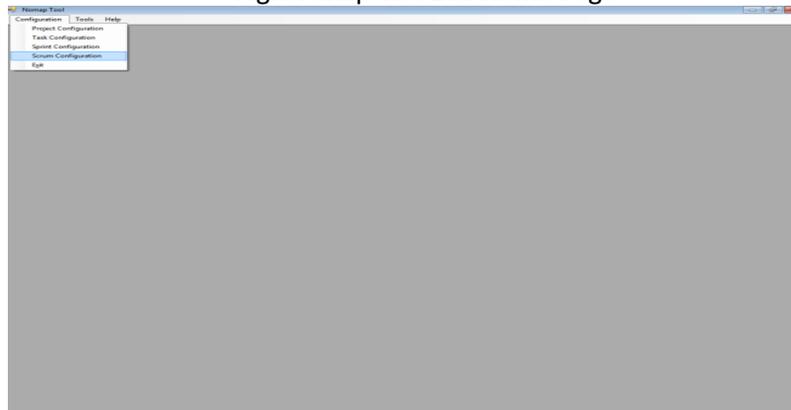
Select exit

End

F. Results

1. User will login with its username and password.
2. After successfully login user will get home page of application.

Fig. 3: Snapshot 1 of Home Page



3. User can add , delete or modify the Project configuration

Fig.4: Snapshot 2 configure project details

Project Code	Project Name	Description	EffortsHours	Estimated Start Date	Estimated End Date	Actual Start Date	Actual End Date	IsCompleted
ERP	ERP	ERP	450.00	2013-10-10	2013-10-10	2013-10-10	2013-10-10	0
CRM	CRM	Customer Relationship m...	1500.00	2013-10-10	2015-10-10	2013-10-10	2015-10-10	0

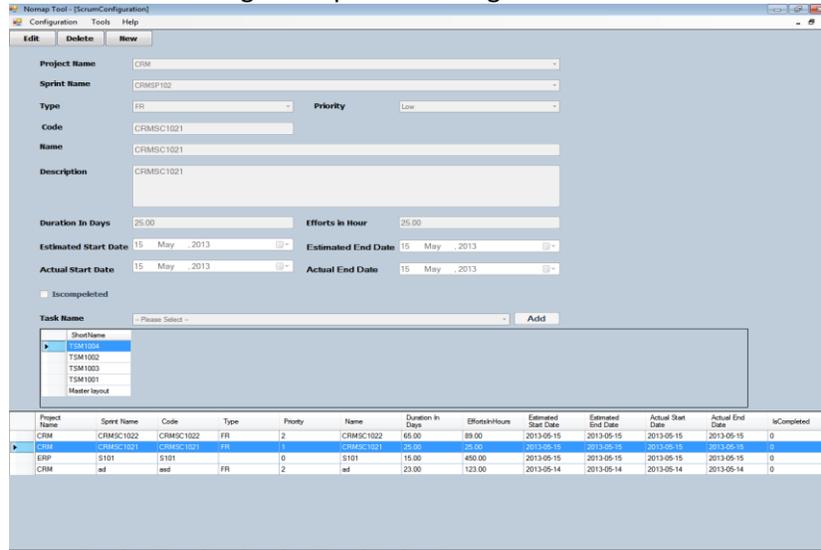
4. User can add, delete or modify sprint configuration

Fig.5: Snapshot 3 configure sprint details

Project Name	Code	Type	Priority	Name	Description	Duration In Days	EffortsHours	Estimated Start Date	Estimated End Date	Actual Start Date	Actual End Date	IsCompleted
ERP	SP101	SP	Medium	ScmAdmin	ScmAdmin	45.00	1600.00	2013-05-13	2013-05-13	2013-05-13	2013-05-13	0

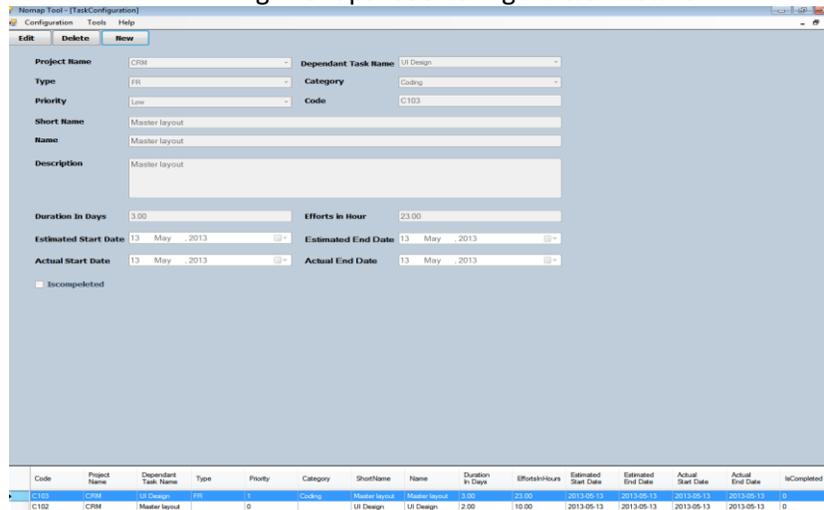
5. User can add, delete or modify scrum configuration

Fig.6: Snapshot 4 configure scrum details



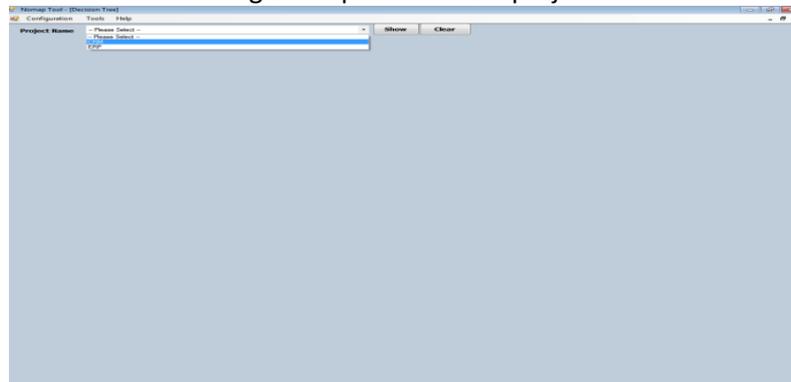
6. User can add, delete or modify task configuration

Fig.7: Snapshot 5 configure task details



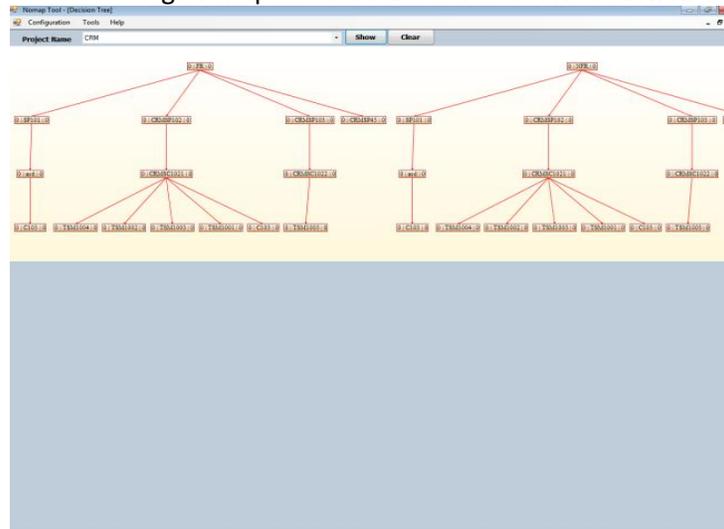
7. Select project name.

Fig 8: Snapshot 6 select project name



8. Click on show button to generate decision tree of FRs and NFRs.

Fig.8: Snapshot 7 decision tree of FRs and NFRs



V. CONCLUSION

At the end, we conclude that appropriate managing of NFR has very significant role in design stage of software development. It can provide help to get better the features and happiness of customers. It also helps the developer to improve the development time as well as cost which is used by the developer to revisit those requirements in further development stages. Failure of Projects, prolonged delay or significant increase in cost are some drawbacks of ignoring NFRs. Success of any software depends on the customer satisfaction which is basically depends on maximizing NFR elicitation and integration in the software product. Lack of support to NFR is responsible for the failure of system.

In this paper we have proposed NFRs elicitation technique. This paper proposes a design of tool to develop NFRs by considering that NFR plays sensitive role in the success of the software system.

References

- [1] Andrea De Lucia and AbdallahQusef "Requirements Engineering in Agile Software Development", Journal of Emerging Technologies in Web Intelligence, Vol. 2, No 3, August 2010.
- [2] JayakanthSeinivas and Kristina Lundqvist, "Using agile methods in software product development: A case study", Information Technology: New Generations, Third International Conference, vol.0,1415-1420,2009.
- [3] Weam M. Farid / Frank J. Mitropoulos "NORMATIC: A Visual Tool for Modeling Non-functional Requirements in Agile Processes" , 2012.
- [4] A.Sidky and J.Arthur, "Determining the Applicability of Agile Practices to Mission and Life-Critical Systems",Proceedings of the 31st IEEE Software Engineering Workshop, IEEE Computer Society,2007,pp.
- [5] Kaushal Pathak an AnjuSaha "Review of Agile Software Development Methodologies", International Journal of Advannced Research in Computer Science an Software Engineering,Vol.3,Issue 2,February 2013.
- [6] Malik Qasaimeh, HosseinMehrfard, AbdelwahabHamou-Lhadj, "Comparing Agile Software Processes Based on the Software Development Project Requirements", CIMCA 2008, IAWTIC 2008, and ISE 2008.

- [7] Md. Mijanur Rahman and Shamim Ripon, "Elicitation and Modeling Non-Functional Requirements – A POS Case Study", December 20, 2012.
- [8] FraukePaetsch, Dr. Armin Eberlein, Dr. Frank Maurer, "Requirements Engineering and Agile Software Development".
- [9] PratimaSingh and Anil Kumar Tripathi, "Issues In Testing of software with NFR", International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.4, July 2012.
- [10]David Ameller, OriolCollell, Xavier Franch, "ArchiTech: Tool Support for NFR- Guided Architectural Decision-Making", June 25, 2012.
- [11]NupurChugh and Aditya Dev Mishra, "Assimilation of Four Layered Approach to NFR in Agile Requirement Engineering", International Journal of Computer Applications (0975 – 8887) Volume 78 – No.5, September 2013.
- [12]GayaneAzizyan, MiganoushKatrinMagarian, Mira Kajko-Mattson, "Survey of Agile Tool Usage and Needs", Agile Conference,,IEEE,2011.
- [13]A. Matoussi, R. Laleau. "A Survey of Non-Functional Requirements in Software Development Process"October 2008.
- [14]Chian Wen Too, Sa'adah Hassan, Jamilah Din, and Abdul AzimAbd. Ghani, "Towards Improving NFR Elicitation in Software Development", International Journal of Information Technology & Computer Science(IJITCS) (ISSN No : 2091-1610) Volume 7 : No : 1 : Issue on January / February, 2013.
- [15]G. Goth, "Agile Tool Market Growing with the Philosophy", 2009, IEEE Software, 26(2), pp. 88-91
- [16]G.Sousa, G.Silva, and J.Castro, "Adapting the NFR Framework to aspect Oriented Requirements Engineering", in the 17thBrazilian Symposium on Software Engineering (SBES), 2003, Manuas, Brazil.
- [17]Aburub F., Odeh M. and Beeson I., "Modeling Non-Functional Requirements for Business Process",Information and Software Technology Journal, vol.49, no. 11-12, pp. 1162-1171, 2007.
- [18]S. Ambler, "Agile Requirements Modeling", 2012 available at : [http://www.agilemodeling.com/essays/agile Requirements.htm](http://www.agilemodeling.com/essays/agile%20Requirements.htm)
- [19]WaleedHelmy, AmrKamel and Osman Hegazy, "Requirements Engineering Methodology in Agile Environment", International Journal of Computer Science Issues, Vol. 9, Issue 5, No 3, September 2012
- [20]A. Ananda Rao and M.Gopichand, "Four Layered Approach to Non-Functional Requirements Analysis", International Journal of Computer Science Issues, Vol. 8, Issue 6, No 2, November 2011
- [21]Saravana. K.M, G. N. Basavaraj, Rajkumar, Dr. A. Kovalan, " Case Study On Agile User Stories Prioritization Using Imaginative Standard", International Journal of Engineering Research and Applications (IJERA), Vol. 2, Issue 5, September- October 2012