

# Requirements Engineering in Scrum Framework

Nagy Ramadan Darwish  
Department of Information  
Systems and Technology  
Institute of Statistical  
Studies and Research

Salwa Megahed  
Department of Information  
Systems and Technology  
Institute of Statistical  
Studies and Research

## ABSTRACT

Requirement Engineering (RE) plays an important role in the success of software development life cycle. As RE is the starting point of the life cycle, any changes in requirements will be costly and time consuming. Failure in determining accurate requirements leads to errors in specifications and therefore to a mal system architecture. In addition, most of software development environments are characterized by user requests to change some requirements. Scrum as one of agile development methods that gained a great attention because of its ability to deal with the changing environments. This paper presents and discusses the current situation of RE activities in Scrum, how Scrum benefits from RE techniques and future challenges in this respect.

## Keywords

Agile, Requirement Engineering, Software Development, Scrums.

## 1 INTRODUCTION

RE is a branch of the system engineering, concerned with constrains and goals to be achieved [1]. As many projects ran over budget and schedule, some projects caused property damage and a few projects caused loss of life. From its beginnings in the 1960s, writing software has evolved into a profession concerned with how best to maximize the quality of software and of how to create it. By early 90th, a radical shift emerged as independent field of study. IEEE sponsored conferences and symposiums for RE. By late 90th, the field had grown and supported a large number of projects [16]. RE is a process of determining the purpose of the system by identifying users and their needs, documenting these in way that helps in analysis, communication and implementation [16].

Scrum is one of the agile lightweight methods that used in software development. It is a simple set of roles, responsibilities, and meetings that never change. Scrum process benefits the organization by helping it to:

- Ensure a high quality of products,
- Scalable from single process to entire project,
- Provide better estimates to time and cost,
- Be more in control of the project schedule and state [10].

Although RE and agile methodology seem to be incompatible as RE is heavyweight in documentation while agile methods are people oriented, still agile in general and scrum in specific can get benefit from RE techniques. Examples of RE techniques in scrum approach: interviewing, JAD, modeling, prototyping, and documentation. This paper aims to highlight the current situation of scrum approach and RE, the use of RE techniques in scrum, and future challenges facing scrum and RE techniques.

The rest of this paper is structured as follows: Section 2 provides related work, Section 3 introduces requirements engineering framework, Section 4 provides the fundamental concepts related to Scrum, Section 5 shows the common RE techniques that are used in Scrum framework, Section 6 presents the conclusion and future work.

## 2 RELATED WORK

In the last years, there has been a growing awareness of the RE importance for increasing the quality of software projects and lot of research has been conducted in this area. This section introduces some important related work in RE and agile methods and scrum framework in addition, it introduces the current situation of scrum approach and RE.

- The authors in [12] provide their research that was conducted using qualitative methods. Respondents who play different job roles from nine organizations were interviewed in order to collect data. Majority of the respondents successfully practices the scrum RE practices, face to face communication, requirements prioritization, iterative requirements engineering and managing requirements change. Some organizations were not comfortable with the test driven development.
- The author in [4] discusses the requirements traceability problem in agile software development and the relationships between the traceability and refactoring processes and their impact on each other.
- The author in [3], provides a comparison between RE and agile development approaches. In addition, the author presents commonalities and differences between both of them and the ways of how agile software development can benefit from requirements engineering methods.
- The author in [13] describes an agile requirements engineering approach for re-engineering and changes in existing Brownfield adaptive system. The approach has few modifications that can be used as a part of Scrum development process for re-engineering and changes. The approach illustrates the re-engineering and changes requirements through introduction of GAP analysis & requirements structuring & prioritization by creating AS-IS & TO-BE models with 80 / 20 rule. An attempt to close the gap between requirements engineering & agile methods in form of this approach is provided for practical implementation.
- The author in [15] explores requirements changes in an agile-scrum software development process. The goal of the study was two folds:

- To empirically investigate the claims of agile proponents that Agile methodology allows changes to requirements even late into the project with minimal impact on software functionality and quality of the delivered product.
- To investigate the impact of requirements changes on the development productivity, if there exist a correlation between the development effort and requirements changes. Measurement data were collected from the development teams. Productivity of multiple project teams was evaluated using both traditional and new measures. Passed test cases is one of the measures investigated in this research. The findings of the experiments demonstrate a high correlation between the number of test cases passed and the productivity.

Current situation of scrum approach and RE is as follow:

- During requirement elicitation: product owner formulates the product backlog. Different stockholders can participate in product backlog. A user story is used to describe a feature that delivers a value to the customer. Story is written on the face of the card while it's acceptance test on the back [2].
- In RE Analysis stage: at this stage, requirements are checked for completeness, consistency, essentiality and feasibility. Product backlog refinement meetings are held to discuss them. Product owner prioritizes the product backlog and analyze the feasibility of requirements. Prioritization in Scrum takes place before each development iteration which is unlike the traditional requirement engineering where prioritization is done only once during the life cycle [2].
- Requirement documentation: it is almost unfeasible. In Scrum approach, documentation is done face to face. A lack of documentation can lead to a long term problems for the team. A new team member will have a lot of questions regarding the project [2, 3].
- Requirement validation: is done through review meetings. Agile in general, and Scrum in specific do not tend to use documentation as in RE. The validation in scrum is done through sprint review at the end of each cycle. This meeting puts both of stockholders and developers on the right track [2, 5].
- Requirement management: generally, requirements are managed through documentation that captures stores and traces each requirement from elicitation till implementation is completed. In scrum approach this can be done through sprint planning meetings, and items in product backlog which are used for tracking. Also changes in requirements are added or deleted to and from product backlog [4].

### 3 REQUIRMENTS ENGINEERING

Requirements engineering is the branch of software engineering concerned with the real-world goals for, functions of, and constraints on software systems. It is also concerned with the

relationship of these factors to precise specifications of software behavior [16]. Requirements are classified into process and product. Process is concerned with cost, lead time and organization, while requirement product is broken down to functional and non-functional requirements. Functional requirements are viewed from two sides: user side (user requirements) and developer side i.e. testability, maintainability [1]. Non-functional requirements i.e. (availability, reliability, maintainability, reusability) are considered the responsibility of developers.

#### 3.1 Requirements Engineering Framework

RE includes many activities that can be framed in framework. One of the most common frameworks for RE, is as shown in fig (1). RE includes five main activities that can be: Elicitation, Analysis and Negotiation, Documentation, Validation, and Management.

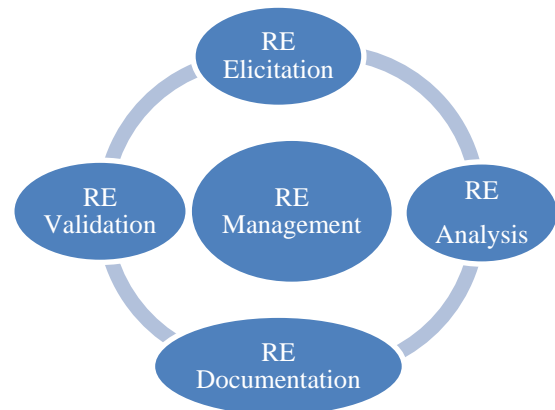


Fig. (1) Requirement Engineering Framework

#### 3.2 Requirements Elicitation

It is the process of collecting requirements from stakeholders. System constrains, boundaries, identification of problems are stated in this stage. This stage is also known as “Requirements gathering”. Some questions should be answered in this stage [4]:

- Does the system contribute to high level objectives?
- Can the system be implemented within given budget?
- Is the system feasible?

#### 3.3 Requirements Analysis

It checks requirements for necessity, consistency, completeness and feasibility. Conflicts in requirements are resolved through prioritization negotiation with stakeholders. Solutions to requirements problems are identified and a compromise set of requirements is agreed [3].

#### 3.4 Requirements Documentation

The main purpose of the documentation is the communication between stakeholders and developers. A lack of documentation can lead to problems on long term [4]. Requirement document should be clear, consistent, concise and feasible [3].

#### 3.5 Requirement Validation

Validation checks consistency, completeness and realism of the requirements [2]. In figure 2, inputs to validation process are: organizational knowledge and organizational standards. The output is a list of reported problems with action to solve these problems [3].

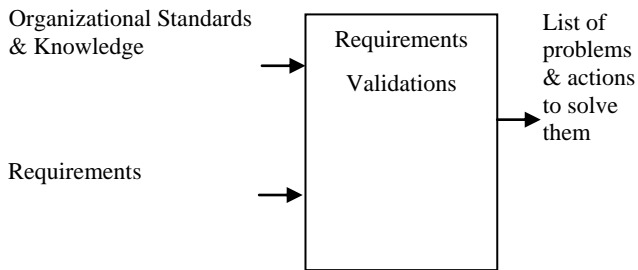


Fig. (2) Validation process

It helps in highlighting problems at early stages of the project life cycle to avoid project delay or failure. The main goal of this stage is to ensure that requirements reflect the real user needs [4, 17].

### 3.6 Requirement Management

This process concerned with all activities associated with changing the requirements of the system from change/version control to requirements tracing [2]. Tools and techniques are used to simplify the process of tracing. Analysis is done to determine the stability of requirements and manage future changes.

## 4 SCRUM

Scrum approach was initiated by Ken Swaber in 1995 and included in agile methodology. Since it contains the same concepts of agile, it is defined as an agile framework for completing complex projects. Scrum approach is originally formalized for software development projects, but works well for any complex, innovative scope of work [5].

### 4.1 Scrum Framework

Scrum approach is based on the principles and values of the agile manifesto, which proposes a different style for managing software development work, encouraging people over processes orientation, working software over documentation, collaboration over contract negotiation, and responding to change over following a plan.

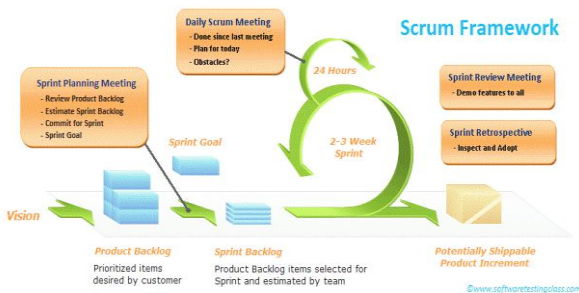


Fig. (3) Scrum Framework [6]

- A product owner creates a prioritized list called a product backlog.
- During sprint planning, the team pulls a small chunk from the top of that list, a sprint backlog, and decides how to implement those pieces.
- The team has a certain amount of time a sprint (usually two to four weeks) to complete its work, but it meets each day to assess its progress (daily Scrum).
- Along the way, the Scrum Master keeps the team focused on its goal.

- At the end of the sprint, the work should be potentially shippable: ready to hand to a customer, put on a store shelf, or show to a stakeholder.
- The sprint ends with a sprint review and retrospective.
- As the next sprint begins, the team chooses another chunk of the product backlog and begins working again [5].

### 4.2 Roles in Scrum

A scrum presents a team that act cooperatively to deliver a project within time and with minimal cost [5]:

- Product Owner: is responsible for setting product backlog, decide about the business value and ROI, prioritize the product backlog.
- Scrum Master: responsible for the implementation of the project, trouble shooter, protecting the team from external influences.
- Scrum Team: self-organizing team, no titles, cross-function [9].

### 4.3 Scrum Artifacts

Scrum approach is an agile software development framework used for incremental software development. The three basic artifacts of scrum are [8]:

- Product Backlog: list of ordered and prioritized backlog items.
- Sprint Backlog: selected backlog items, unchangeable and committed items.
- Increment (Burn chart): to show how much work left against the time.

### 4.4 Events in Scrum

These events enable transparency on the project progress to all who are involved in the project. The vital events of scrum are:

- Sprint: is an iterative, time boxed duration with fixed time (from two weeks to month). Each sprint starts with planning and ends with review.
- Sprint planning: communicates the scope of work likely during that sprint
  - Selects product backlog items that likely can be done.
  - Prepares the sprint backlog that details the work needed to finish the selected product backlog items, with the entire team.
  - Sets a four-hour time planning event limit for a two-week sprint [9].
- Daily Scrum: Each day during a sprint, the team holds a daily scrum (or stand-up) meeting to discuss:
  - What have finished yesterday?
  - What will be done today?
  - Any obstacles!.
- Sprint Review: At the end of a sprint, the team holds two events: the sprint review and retrospective. At sprint review, the team:

- Reviews the work that was completed and the planned work that was not completed.
- Presents the completed work to the stakeholders [9].
- Sprint retrospective:
  - Two main questions are asked in the sprint retrospective: What went well during the sprint? What could be improved in the next sprint?
  - The recommended duration is one and a half hours for a two-week sprint (pro-rata for other sprint durations).
  - This event is facilitated by the scrum master.

## 5 RE TECHNIQUES USED IN SCRUM APPROACH

RE in Scrum approach is iterative, not final and predefined but evolves in each Sprint. In each Sprint, customer meets the development team to provide with the next set of requirements to be implemented during that Sprint. An advantage of this practice is that since the customers are not clear about the exact requirements at an initial stage, when they see the evolving system they can be more clear and specific about the requirements. The development team also has the benefit of immediate access to the customer in order to understand the requirements better [12]. The following requirements engineering techniques help those scrum techniques as product backlog, sprints, and daily scrums to be achieved.

### 5.1 RE Elicitation

At this stage, the team has to obtain all information about the application domain, services that the system should provide [4]. The choice of the suitable technique depends on what kind of information is required, time, and cost [16]. Most important techniques used in this stage are:

- Interviews: the important point in agile methods is “customer involvement”. Interviewing is the process of transferring unfiltered knowledge. In order to avoid misunderstanding, interviewing the customer is important for development of the system [3]. As per CHAOS report, the customer involvement by this technique (interviewing) and user on site was found number one in projects success [1]. Using RE techniques which based on user involvement is important in Scrum approach [3, 4].
- Prototyping: it is considered an effective visualized tool which helps the user to illustrate the proposed system. There are three types of prototyping: low-fidelity, high-fidelity and wizard-of-oz prototypes. The first type is a sketch to show how user interface will look. It can be manual drawing on papers or by using a computer based tool. The high-fidelity type of prototype is done using web-page. This type provides more realistic illustration of the proposed system. The wizard-of-oz, is a type of prototype which simulates the responses of the system in response to some user inputs [2, 18].
- User stories: one of Scrum approach practices used to record the user requirements. Features and values of the product are written by the user on front of the paper and the results on the back. User stories are also used a type of planning and documenting the requirements. The team then writes the user stories in proper business language to understand [2].

### 5.2 Requirements Analysis

The main purpose of this stage is to ensure that requirements decided in elicitation stage are clear, complete, and consistent. Conflicts are resolved by prioritization with users. The common techniques used for requirements analysis in scrum approach are [4]:

- JAD sessions: these sessions encourage customer involvement which is the base line in agile methodology in general and scrum in specific. Sessions are important at the beginning of the project. The results of the sessions are for documentation. Documentation in agile methods is relaxed. Those sessions have to be held frequently during the project [2, 3,4].
- Prioritization: in scrum approach, in beginning of each iteration, requirements are collected and prioritized in stack [4]. Only product owners prioritize them according to the business value, customer needs and ROI. The frequent communication between the team and stockholders helps to distinguish between “must have” requirements and “nice to have” requirements [4]. At the following diagram, if requirements are valid, they are included in prioritized list with the new ones. Then a new list is prepared to identify the features that will be implemented.

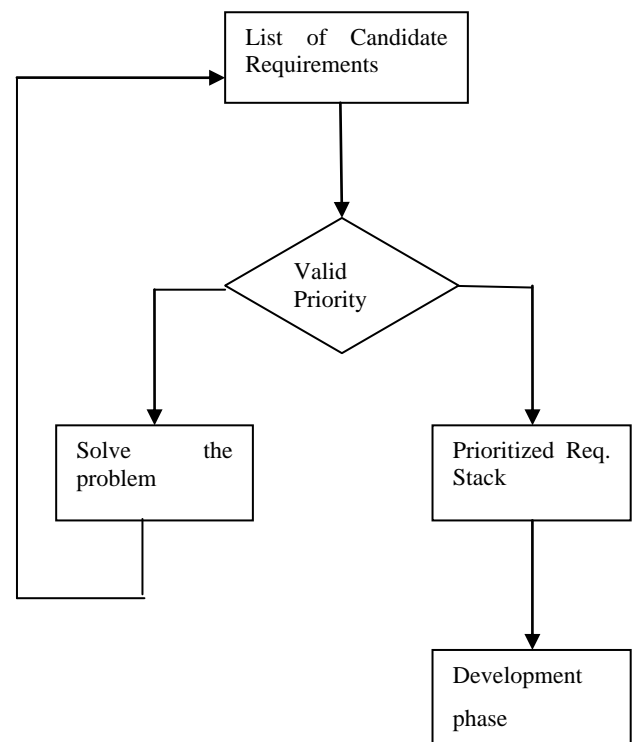


Fig. (4) Requirement prioritization process [4]

### 5.3 Requirements Validation

This process helps to improve the software quality. The main practices used for requirements validation in scrum are:

- Review meetings are held frequently for the purpose of validating the requirements. During review meetings, customers and scrum team checking requirements against organizational standards, organizational knowledge and user requirements to know about the strength and weakness of the design, technology limitations.

Conflicts and errors in requirements have to be solved [1, 2].

- Testing in scrum approach is done after each review to check if the system reacts in expected way. If not, then a clarify needs to be presented [3, 18]. There are different kinds of tests. Acceptance test, test driven development and acceptance test driven development. Acceptance test is a satisfaction conditions that determines if product features are implemented. Test driven development is a kind of test that is developed prior to the code while acceptance test-driven development helps in building right product [4].

#### 5.4 Requirements Documentation

The RE process is not only for finding out information about the system but also essential way of communication between all parties involved in the software development process. Requirements documentation helps in analyzing, re-writing, and validating those requirements. Although modeling and documentation in agile methods less used than RE, they are used as a way of communication with the user [16]. Scrum master assigns some members to produce modeling for the purpose of documentation in parallel. Use Cases describe the interaction between users and the system and considered as documentation [3,4]. Developers use computer-based and project management tools for high level description of the system. Also, a reverse engineering process can be used to reverse the code to documentation.

#### 5.5 Requirements Management

Scrum approach believes that changes in the system are inevitable. Problems of this stage are summarized in:

- Few people are experienced in requirements management.
- People don't distinguish between users and stakeholders requirements of the system.
- The way in which requirement management problems are managed and solved [19].

Managing requirements changes is fulfilled by main practices such as:

- Iterative RE: the main source of requirements information is analysis. Prior to each sprint, information is needed for sprint planning and implementing. Prioritize of requirements is done according to their business value decided by the customer.
- Short releases: in order to meet the user requirements faster, software is delivered in short releases. Release is the output of the development process. Changes will be required if output is odd. No doubt that each release confirms the relationship between users and development team in a way of communication.
- Customers' feed-back: before the beginning of each sprint, a planning meeting is held to ensure that development team aware of the product details. This meeting helps in minimizing changes after the sprint when product is delivered. At the end of each sprint, a sprint review meeting is held to discuss and share feed-back with development team if the product is not meeting the customer expectations [2, 16]. Both pre- and post- meeting are essential to obtain a product that meets user requirements.

## 6 CONCLUSION AND FUTURE WORK

Although scrum is lightweight method used in small projects, it can still benefit from requirement engineering techniques as shown in section (V). Many challenges of requirements engineering in scrum framework are presented as follow:

A lack of requirements techniques in scrum that practitioners can choose from.

Non-functional requirements: no accepted techniques for managing non-functional requirements. They are ill defined.

Scalability: doesn't mean growth of size in S/W. It also refers to complexity, variability in software line and a degree of heterogeneity [2].

In this paper, a survey research in requirements engineering stages and scrum framework is presented. Also, an attempt of answering how scrum framework can benefit from RE techniques is introduced. A further research is targeted to solve the problems of using RE techniques in scrum framework.

## 7 REFERENCES

- [1] Aybüke A., and Claes W., 2005, Engineering and Managing Software Requirements, Chapter 20, pp. 453-476.
- [2] Elshandidy, H., and Mazen, S., 2013, Agile and Traditional Requirements Engineering: A Survey, International Journal of Scientific & Engineering Research, Vol. 4, Issue 9.
- [3] Eberlein, A., and Maurer, F., 2003, Requirements Engineering and Agile Software Development, University of Calgary.
- [4] Quesf, A., 2010, Requirements Engineering in Agile Software Development, Journal of Emerging Technologies in Web Intelligence, Vol.2, No.3.
- [5] Mahalakshmi, M., et al, 2013, Traditional SDLC Vs Scrum Methodology – A Comparative Study, India, International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 6.
- [6] Scrum Framework  
image: [https://search.yahoo.com/yhs/search?hspart=avg&hsimp=yhsfh\\_lsonswrow&type=ch.49.xp.nt.04.03.eg.avg...1215tb&param2=new\\_tab\\_search&param3=ch.49](https://search.yahoo.com/yhs/search?hspart=avg&hsimp=yhsfh_lsonswrow&type=ch.49.xp.nt.04.03.eg.avg...1215tb&param2=new_tab_search&param3=ch.49)
- [7] Helmy, W., and Hegazy, O., 2012, Requirements Engineering Methodology in Agile Environment, Faculty of Computers & Information Systems, Cairo University.
- [8] Nagy Ramadan Darwish and Abdel Wahab, E., 2016, A Security Testing Framework for Scrum Based Projects, International Journal of Computer Applications, Vol. 138, No.7.
- [9] Cho, J., 2008, Issues and Challenges of Agile Software Development with Scrum, Journal of Management Information Systems, Vol. IX, No. 2, pp. 188-195.
- [10] Marchesi, M., et al, 2007, Distributed Scrum in Research Project Management, Conference: Agile Processes in Software Engineering and Extreme Programming, 8th International Conference, XP, Como, Italy, June 18-22.
- [11] Dick Carlson, 2007, Practical Agile Requirements Engineering, Presented to the 13th Annual Systems Engineering Conference, Hyatt Regency Mission Bay,

- San Diego, CA.  
[http://dtic.mil/ndia/2010systemengr/WednesdayTrack1\\_11106Matuzik.pdf](http://dtic.mil/ndia/2010systemengr/WednesdayTrack1_11106Matuzik.pdf).
- [12] Vithana,N., 2015, Scrum Requirements Engineering Practices and Challenges in Offshore Software Development, Sri Lanka Institute of IT Sri Lanka, Singapore.
- [13] Bin Masood, A., et al, Applying Agile Requirements Engineering Approach for Reengineering & Changes in existing Brownfield Adaptive Systems, Mohammad Ali Jinnah University.
- [14] Fahd, M., 2014, Customer Oriented Requirement Engineering By Using Scrum, Dept. of Computer & Information System, Islamia University, Pakistan.
- [15] Mueller, C., 2011, Requirements Management in an Agile-Scrum, Department of Computer Science San Marcos, TX.
- [16] Nuseibeh, B., et al, Requirements Engineering: A Roadmap, Department of Computer Science, Imperial College, London, U.K.
- [17] Szekely, P., 1994, User Interface Prototyping: Tools and Techniques, USC/Information Sciences Institute, Marina del Rey, CA., USA.
- [18] Sommerville, I., 1997, Requirements Engineering, John Wiley & Sons.
- [19] Hull, E., et al, 2005, Requirements Engineering, Springer London Berlin Heidelberg, Second Edition, Chapter 8, pp. 153-154.