



Challenges & Solutions of Requirement Change Management

Muhammad Ahkam Ahmad and Waqas Mehmood

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

January 22, 2020

Challenges & Solutions of Requirement Change Management

Abstract:

For the achievement of business goals and survive in current software market it is necessary to respond quickly and efficiently to requirement changes. But this is a universal fact that when changes comes in software development life cycle. This become a cause to study about the challenges and their possible solutions of requirement change management. Which includes some of the practice methods use in industries to tackle those challenges. Challenges that have come across include change management automation, connectivity with software artifacts, change active management, change anticipation and reusability. Also there are equal solutions which includes best practices and some frameworks to handle discussed challenges in requirement change management. In this paper we have tried to discuss the huddles that come across the production of a software system and their possible solutions or processes.

Introduction:

In current era of software development life cycle the changes in client's requirement and addition of new and innovative features is a constant process during the development of a specific system [1]. The changes are normally occurred due to updating hierarchy level, for instance change in stakeholders and business rule or it can be a change in user class. Normally the unclear vision for the product lead to the constant changes demands from client side.

There are a number of reasons for a software requirement change, one would obviously be lack of clear vision and scope of the system that how far and vast implementation of the system could be possible at initial stage as requirements are not clear at first [2] [3]. Which can be result in total wrong development of the software system. More factors of changes could be universal one that is affecting a larger dimension like introduction of new rule or constitution form government side that is affecting many businesses. In addition market competencies also show effects on frequent requirement changes [4].

Moreover changes in requirements bring up the flood of bugs and errors that results in lack of client satisfaction and create a block in the software flow which results in delay in the software development life. In this new era of constantly change in requirements, build a system which would sustain in continuous requirement change is critical [5] [6]. Which directly impacts the need for which the system is developing that is to achieve the premium business goal. Hence it is notable that to obtain the great success of the developing project, it is extremely important that changes of requirements significantly managed because it affects the success criteria [7] [8].

A bulk of cases has been recognized where the climax of the product was not satisfactory as development team failed to manage requirement changes [9]. Which means that inefficient requirement change management will lead to the project failure and play a vital role for heavy business losses [10] [11]. Apparently, it is an integral thing to analyze the latest challenges of requirement change management and solution of those challenges to achieve maximum success in developing project.

Background:

Integral part of requirement engineering is requirement change management and complete understanding of its activities is essential for get a success in implementation of changes in software system. Requirement change management can be defined as an exercise of handling modification in software requirement during whole software and requirement engineering process [7]. In this exercise, flood of new requirements and modification in current building software system observed which required proper process to tackle significantly.

The term management can be elaborate as requirement addition, updating requirement, requirement deletion and error fixing [12]. The main activities perform in requirement change management process are get early change request, develop a proposal document, quantify the effect of change, make choice either to make a change or not and finally apply the modification to the system [7]. Requirement change management has many issues to resolve. One of them is inadequate requirements, which means the problem in definition or description of changes which need to be done [9] [13]. Clearly, adequate information about what is required is the basic need of both requirement management and requirement development as well to achieve the system goal.

Moreover, some of the requirements are not oblivious and easy to understand even for the stakeholders [15]. Which results the development of wrong system or system failure because important and adequate description was not present [16]. The second issue is ambiguous requirements which means that idea is so unclear and creeping the scope that it becomes ambiguous [17]. It generally come when proper methods does followed in requirement gathering process. The third issue is requirement traceability which means to provide a framework or method by which each change can be traceable [9] [18] [19]. The four issue is cost and time, many researchers have a perception that consumption of cost and time is a key problem in requirement change management [8] [11] [18] [23]. These are the issues to be address and solve in requirement change management process.

Literature Review:

REQUIREMENT CHANGE MANAGEMENT CHALLENGES:

Requirement Change Management has following challenges according to key findings.

A. Reusability:

Reusability means that requirements must be in a form that is useful in future [25] [26].

It has a huge role in requirement development phase and when we move to requirement change management phase, it importance extends. In this era of fast modification in requirements and need of adaptation of these requirements signify the need of reusability and reuse of tested features of present information system [25]. It is clear that reusability reduces time consumption and keep a record of monotonous works in constantly changing surroundings for dealing with latest requirements at fast pace. Reusability should be adapt at initial level of requirement development so that it can also be mirror in requirement change management phase. It should also have to be extendable to attain high adaptation on modification.

B. Change Activity Measurement:

Use of framework to record change histories is a great habit to adopt in requirement change management. Examining change exercise is a work to anticipate requirement's adherence and seek for adjustment to enhance process that might result in less changes in future [3]. Moreover, change activity measurements help to describe and keep a record of changes revision history [23]. It would also help to obtain a deep insight of future decision making to attain an ideal and optimal implementation.

C. Connectivity with Software Artefacts:

The process of Requirement change management and software artifacts has a significant relationship with each other so connectivity between both processes is a must. This connectivity would solve many conflicts and save rework, for example modification according to requirement and then do coding means that two times change impact analysis for the maintainer one for requirement occur, second when actually coding work being done [22]. In this manner the life cycle of software development is affected by involvement of changes in each phase in the shape of cost and time [8]. Customer's intellectual level of demands are the big problems because code work has many design patterns and mirror multiple implementation therefore it is difficult to match requirement and actual code work [22]. In general, most of the requirement given informally and cannot be tracked because unrecorded [21]. Which lead to the fact that requirements should connect with software artifact programmatically and in a formal way. As the actual functionality of the system should be clear and communicate via formal specification [27].

D. Change Anticipation:

A challenging task in software development is to anticipate correct changes in requirement change management process. As some requirements are not known at the time of development of and become known in future [11]. In framework of website application anticipating market and customer need and take action according to that will be appreciated [28]. It is better to identify changes at an early stage rather to do it at production stage [29].

E. Change Management Automation:

So many reliance on human is a key problem in requirement change management [32] [33]. Although many organizations and IT companies are currently spending their time and money in the maintenance of already build system and the do not have further extension capability to develop a new system [34]. Therefore to obtain a life cycle of software development that will automate all routine tasks is required [35]. Use of tool to achieve this task is successful and enhanced start [3]. But in literature usage of agent-based process is recommended to enhance the productivity of software process [36] [37] [38] [39] [40] [41].

Methodology:**SOLUTION OF REQUIREMENT CHANGE MANAGEMENT CHALLENGES:**

Solution of Challenges of found challenges of Requirement Change Management are discussed as.

A. Attaining Reusability:

The process to reuse the code of different module of different project can be term as reusability [15]. The modules that can be reused between different projects are the module of document of similar projects. the design patterns that have been used, the test case documents, the project management strategy documents, the architecture followed and tested which also give the software a high level of maintainability, different diagrams like ERD, UML and class diagrams and third party libraries. This specific method of reusability not only reduces the effort and time but it also enhance the quality of the software. Most of the organization have the database to backup reusability. There are a lot of methods available to attain reusability. There are standards that have been followed in industries, one of them is COBRA (**C**OST Estimation **B**enchmarking and **R**isk **A**ssessment) that is developed to create interaction between components. There are many more methods and techniques that are being used to achieve reusability [16].

B. Effective Change Activity Measurement:

The need of measuring the change is to keep a record of every input received from any side during software development life cycle and effective project management. To do this effectively every possible detail about the change that is going to be made and the details of the person suggesting change have to be recorded precisely. The details would contain the identity of person, its designation, date of change suggestion, attachments and descriptions about changes, line of codes that have been changed, details of persons that made actual changes, reported bugs against change, test cases applied and bug fix history. This information can be obtained by using software tools like SVN, GIT etc. [10]

C. Connection of Software Artefacts:

To achieve better communication between software artefacts that are the patterns documentation, code and designs, it is essential to take a broad look of the project rather than to just focus on developing application [21]. The researchers give the theory that keeping the development environment and the software artifacts that are reusable separate is the primary feature of the model and framework of reuse-orientation, which is called experience factory. So it is recommended to keep both things separated and synced.

D. Reliable Change Anticipation:

To obtain the effective change management in future it is better to build the architecture of the system extendable. Refactoring is a key solution to provide reusability, extendibility and flexibility type quality to the system [5]. In this manner, usage of design patterns in any system enhance maintainability and reusability. In regard of reusability design pattern refactoring have a stable future. In order to find better approaches and new techniques research is in progress. In the new era of programming based on objects, designed pattern have earned a lot of attractions. The refactoring pattern means that it is better to use a good design pattern at an early stage of software development rather to adopt when changes come on head. It is not possible to directly transform the specific design into code. It describes a way and set a road map that defines that there can be multiple approaches for the solution of a specific problem in different criteria. Particularly design patterns of object oriented approach defines interaction and relationship between objects or more precisely interaction or relationship between classes. FIM architectures [19] or patterns with architectural reusability provides reusable architectures [18]. Which function on all level of reusability.

E. Automate The System:

Requirement change management tool are available in market that provides data type of requirements in a simple format or schema. There are research environments available like TOOR that gives us the functionality to link software artifacts and other relationships and show in object oriented format, also provide searching facility through requirement related data and provide requirement support as well [17]. There is another tool named IBM Rational that provide us track ability of models in a sophisticated manner. It is a useful tool for optimizing, collaborating, and verifying requirements across the organization. This RM tool allows to capture, trace, analyze and manage changes to information.

Analysis:

In search of solution of current challenges of requirement change management it is found that formal specification has occurred as avoided method which results in inaccurate requirements, untraceable changes history and repetitive work. Formal specification keeps software artifacts connected and helps in automation of change management by using approach which is agent oriented. The benefits which it gives includes a vision of current business and problem to be solved by developing the system and at what point the system can be exceeded to achieve the business goals. Which give all functional requirements and an anticipation of changes which can occur in future. It also give a map to reuse the already developed module in systematic way and avoid gold plating. By doing the formal specification via an automated system, automated requirement change management can be achieve and traceability increases. Moreover by adopting sustainable design patterns in software basic architecture the system get the flexibility to extend cause less rework. In addition frameworks like GIT or SVN can also be used to track the requirement changes and requirement change history. By considering all the techniques and methods effective requirement change management can be achieved and scope of the system will be balanced.

Conclusion:

The work that has been done, shows the current provocations and drawbacks of Requirement Change Management. The achievement of challenges required a hierarchy that has been followed as creating the business arrangement, developing the formal stipulation, change management computerization through intermediary approach and change activity measurement. Reclaimable and relatedness of requirements with software commodity will result from achieving formal stipulation and change management computerization. These challenges and solution can be vary from time to time and adaptation of new techniques and further improvement is the constant need of sustainability. At the end use of automation system for both requirement gathering and requirement change management would be a beneficial step.

Reference

- [1] N. Ali and R. Lai, "Managing requirements change in Global Software Development," in *2014 International Conference on Data and Software Engineering (ICODSE)*, Bandung, Indonesia, 2014.
- [2] K. Weigers and B. Joy, *Software Requirements*, 3rd Edition, Microsoft Pres, 2013.

- [3] H. Zedan and M. A. Khan, "Alignment strategies and frameworks in co-evolution of business and information technology," in *International Conference on Information, Networking and Automation (ICINA)*, Kunming, China, 2010.
- [4] T. Rusinaitė, D. Kalibatienė and O. Vasilecas, "Requirements of dynamic business processes - a survey," in *IEEE 3rd Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE)*, Riga, Latvia, 2015.
- [5] S. Schiffner, T. Rothschild and N. Meyer, "Towards a Subject-Oriented Evolutionary Business Information System," in *IEEE 18th International Enterprise Distributed Object Computing Conference Workshops and Demonstrations*, Ulm, Germany, 2014.
- [6] A.-L. Lamprecht, S. Naujokat and I. Schaefer, "Variability Management beyond Feature Models," *Computer*, vol. 46, no. IEEE, p. 54, 2013.
- [7] H. Naz, Y. H. Motla, S. Asghar, M. A. Abbas and A. Khatoon, "Effective usage of AI technique for requirement change management practices," in *5th International Conference on Computer Science and Information Technology*, Amman, Jordan, 2013.
- [8] M. Bano, S. Imtiaz, N. Ikram, M. Niazi and M. Usman, "Causes of requirement change - a systematic literature review," in *16th International Conference on Evaluation & Assessment in Software Engineering*, Ciudad Real, Spain, 2012.
- [9] S. Jayatilleke and R. Lai, "A Method of Specifying and Classifying Requirements Change," in *22nd Australian Software Engineering Conference*, Melbourne, VIC, Australia, 2013.
- [10] A. A. kalbani and K. Nguyen, "Designing flexible business information system for modern-day business requirement changes," in *2nd International Conference on Software Technology and Engineering*, San Juan, PR, USA, 2012.
- [11] A. A. Khan, S. Basri, P. D. D. Dominic and Fazal-E-Amin, "A process model for Requirements Change Management in collocated software development," in *IEEE Symposium on E-Learning, E-Management and E-Services*, Kuala Lumpur, Malaysia, 2012.
- [12] S. Hassan, U. Qamar and M. A. Idris, "Purification of requirement engineering model for rapid application development," in *6th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, Beijing, China, 2015.
- [13] M. Hanafiah and R. Abdullah, "An evaluation on components of experience based factory model in requirement engineering process: A preliminary study," in *Proceedings of the 6th International Conference on Information Technology and Multimedia*, Putrajaya, Malaysia, 2014.
- [14] B. Suranto, "Software prototypes: Enhancing the quality of requirements engineering process," in *International Symposium on Technology Management and Emerging Technologies (ISTMET)*, Langkawai Island, Malaysia, 2015.

- [15] S. K. Saha, M. Selvi, G. Büyükcan and M. Mohymen, "A systematic review on creativity techniques for requirements engineering," in *International Conference on Informatics, Electronics & Vision (ICIEV)*, Dhaka, Bangladesh, 2012.
- [16] M. W. Grenn, S. Sarkani and T. Mazzuchi, "A Theory of Information Quality and its Implementation in Systems Engineering," *IEEE Systems Journal*, vol. 9, no. 4, pp. 1129 - 1138, 2013.
- [17] C. K. Chang, "Situation Analytics: A Foundation for a New Software Engineering Paradigm," *Computer*, vol. 49, no. 1, pp. 24 - 33, 2016.
- [18] A. Akhtar, Y. H. Motla, H. Aslam and M. Jamal, "Role of requirement change in software architecture using Twin Peaks Model," in *IEEE 5th International Conference on Software Engineering and Service Science*, Beijing, China, 2014.
- [19] S. Ramzan and N. Ikram, "Making Decision in Requirement Change Management," in *International Conference on Information and Communication Technologies*, Karachi, Pakistan, 2005.
- [20] J. Cleland-Huang, C. Chang and M. Christensen, "Event-based traceability for managing evolutionary change," *IEEE Transactions on Software Engineering*, vol. 29, no. 9, pp. 796 - 810, 2003.
- [21] B. Burgstaller and A. Egyed, "Understanding where requirements are implemented," in 2010, Timisoara, Romania, IEEE International Conference on Software Maintenance.
- [22] E. B. Charrada, "Updating requirements from tests during maintenance and evolution," in *Proceedings of the eighteenth ACM SIGSOFT international symposium on Foundations of software engineering*, Santa Fe, New Mexico, USA, 2010.
- [23] S. Hussain, N. Ehsan and S. Nauman, "A strategic framework for requirements change in technical projects: Case study of a R&D project," in *3rd International Conference on Computer Science and Information Technology*, 2010.
- [24] B. B. Chua, "Rework Requirement Changes in Software Maintenance," in *Fifth International Conference on Software Engineering Advances*, Nice, France, 2010.
- [25] M. Radgui, R. Saidi and S. Mouline, "Business models alignment with reuse approach," in *IEEE Eighth International Conference on Research Challenges in Information Science (RCIS)*, Marrakech, Morocco, 2014.
- [26] S. Banerjee, A. Sarkar and N. C. Debnath, "Quality evaluation of requirement engineering framework: Business object based approach," in *International Conference on Computing, Management and Telecommunications (ComManTel)*, Ho Chi Minh City, Vietnam, 2013.
- [27] M. M. Ali and T. K. Saha, "A proposed framework for full automation of software testing process," in *International Conference on Informatics, Electronics & Vision (ICIEV)*, Dhaka, Bangladesh, 2012.
- [28] H. Demirkan, R. R. Harmon and M. Goul, "A Service-Oriented Web Application Framework," *IT Professional*, vol. 13, no. 5, pp. 15 - 21, 2011.

- [29] L. Shi, Q. Wang and M. Li, "Learning from evolution history to predict future requirement changes," in *21st IEEE International Requirements Engineering Conference (RE)*, Rio de Janeiro, Brazil, 2013.
- [30] B. Xu, J. Wu and H. Cai, "Business process driven ontology discovery method from distributed data environment," in *Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, Shanghai, China, 2011.
- [31] M. U. Bokhari and S. Tabrez, "A Comparative Study of Software Requirements Tools For Secure Software Development," in *3rd National Conference; INDIACOM*, New Delhi, 2009.
- [32] N. Kushwaha, S. Sahu and R. K. Tyagi, "Evolving intelligent agents for hospital management system," in *3rd IEEE International Advance Computing Conference (IACC)*, Ghaziabad, India, 2013.
- [33] A. Shaban-Nejad and H. Volker, "Towards a framework for requirement change management in healthcare software applications," in *22nd ACM SIGPLAN conference on Object-oriented programming systems and applications companion*, 2007.
- [34] S. Konno, "Service oriented organizational management system for an information systems business," in *ICSSSM12*, Shanghai, China, 2012.
- [35] P. H. Feiler and W. S. Humphrey, "Software process development and enactment: concepts and definitions," in *Proceedings of the Second International Conference on the Software Process-Continuous Software Process Improvement*, Berlin, Germany, 1993.
- [36] V. Gaur, A. Soni and P. Bedi, "An Application of Multi-Person Decision-Making Model for Negotiating and Prioritizing Requirements in Agent-Oriented Paradigm," in *International Conference on Data Storage and Data Engineering*, Bangalore, India, 2010.
- [37] F. Golpayegani, K. Azadbakht and R. Ramsin, "Towards process lines for agent-oriented requirements engineering," in *Eurocon*, Zagreb, Croatia, 2013.
- [38] A. Fleischmann, U. Kannengiesser, W. Schmidt and C. Stary, "Subject-Oriented Modeling and Execution of Multi-agent Business Processes," in *IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, Atlanta, GA, USA, 2013.
- [39] C.-e. Lin, K. M. Kavi, F. T. Sheldon, K. M. Daley and R. K. Abercrombie, "A Methodology to Evaluate Agent Oriented Software Engineering Techniques," in *40th Annual Hawaii International Conference on System Sciences (HICSS'07)*, Waikoloa, HI, USA, 2007.
- [40] S. Ali, B. Soh and T. Torabi, "A novel approach toward integration of rules into business processes using an agent-oriented framework," *IEEE Transactions on Industrial Informatics*, vol. 2, no. 3, pp. 145 - 154, 2006.
- [41] Z. Hua, Z.-C. Wang, J. Hua, G.-X. Yang and Y.-W. Liu, "The Framework of Agent-Oriented Programming," in *International Conference on Machine Learning and Cybernetics*, Guangzhou, China, China, 2005.