

# Effective Requirement Management Through Version Control Data centric Approach

Imran Iqbal<sup>1</sup>, Naveed Abbas Mirani<sup>2</sup> and Ali Javed<sup>3</sup>

<sup>1</sup> Software Engineering, UET Taxila, SCOTMANN  
Islamabad, Punjab, Pakistan

<sup>2</sup> Software Engineering, UET Taxila, POF  
Wah Cantt, Punjab, Pakistan

<sup>3</sup> Software Engineering, UET,  
Taxila, Punjab, Pakistan

## Abstract

Due to the rapid modern advancements in new era, development of new products became necessary. Whether it's the PLC's or the microprocessor that manipulates data, requirement of well managed, highly accurate and effective software's are needed to provide the desired results.

The first step for the successful software development process is the effective requirement management. As the competition in the software development industry grows day by day it is necessary that one provides the cost effective and timely solutions.

The paper is organized in a way to design a Data Model of requirement management version control mechanism.

## 1. Introduction

As per Carnegie Mellon Software Engineering Institute, "Data signify that 60 – 80 percent of the expense of software development lies in rework". [1] Which refers to the job that ought to be carried out the 2nd time since it could not be done perfectly the 1st time. One major cause of alteration is attributed to inadequate requirement management. Also, rework retards the timely delivery that multiplies the perceptible production expenditures as well as tarnishes the repute of the corporate a great deal.

Requirement management is continuous phenomenon that runs hand-in-hand with the development process that guarantees the whole set of settled requirements and irregularities in the planning are dealt appropriately. It's worthwhile to identify and record the connection among requirements in addition to the discernment of any discrepancies studied by the technical team in their entirety. Errors detected subsequently in the development process may inflict an enterprise up to twice 100 folds loss greater than if they were checked in the requirements' analysis point. Modifications are desirable if they are fruitful; rapid and nonstop changes break the flow of development process. It

is also better to have a change control mechanism, where all the changes with their impacts and status are dominantly documented. Changes should be adequate, practically sound, and simple otherwise complex changing requirements creates complexity in development process, hence consuming more resources.

Effective Requirement Management needs specialized Tracking and Tractability tools, Most of the organizations are not willing to spend at the start of the projects for such ever demanding tools. That sort of approach cost more in the testing and maintainability phase makes the system complex and process draw many a time more resources then what required for the purchase of these analysis tools in the requirement phase.

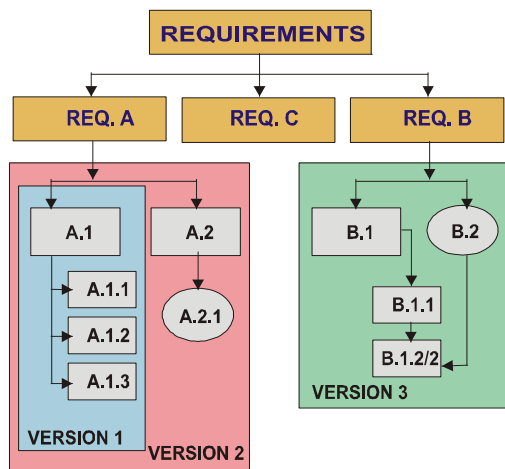
Requirement management repository is a shared library, which allows multiple business units to contribute and reuse requirements assets. Requirements can be reused "as is" or modified by creating a new version for each project. This flexibility reduces duplication, and supports compliance requirements by mandating that specific requirements be used to enable regulatory or process compliance [3], hence minimized the resources required for the future projects.

## 2. Literature Review

There are two basic types of approaches followed for effective requirement management, Database Centric Approach and Document centric Approach. Companies how have the luxury of higher budget usually go for automated tools provided by many vendors where as in smaller scope projects with limited resources, requirements are managed by document centric (Microsoft Word, Visio) approach.

We have designed a model based on database centric approach, which uses a version control requirement management technique. Most of the available tools are either very expensive or have the limited functionality, unable to meet the increasing demands of Effective Requirement Management. Our idea of version control requirement based on the theory, that application can be divided into smaller versions for incremental delivery. Hence requirements should be in accordance with these versions rather than making a heap of requirements which are difficult to traceable.

**3. Proposed System Design**



General System Layout

In our general proposed layout all distinct major requirements are represented at the first layer, as we are going deeper in the layout list of more detailed requirements are presented. Versions consisting of at least one or group of requirements are created based on the priority of the outcome required. In the above scenario priority of version 1 consisting of set of A.1.1, A.1.2, A.1.3 is assumed to be higher, hence required to be delivered first. Similarly version 1 and version 2 combines to fulfill the REQ.A (Requirement A).

The layouts like one above is automatically generated by our Version Control Requirement Software, the user only had to provide the relevant data. The set of data required is shown in the figure below.

Requirement Name	Description	Status	Priority	Effort	Approved (date)
------------------	-------------	--------	----------	--------	-----------------

Fig 2 Dataset Required Model

Detail functionality of these key factors can be illustrated as;

S.#	Information	Description
1.	Requirement Name	It must be the logical name of the requirement which uniquely distinguish it from other requirements.
2.	Description	Brief detail of functionality of a specific Requirement.
3.	Status	Describes the status of a specific Requirement whether it is Approved, Proposed or Deferred.
4.	Priority	It is very critical as Versions are based on the priority of the requirements.
5.	Effort	It is the estimated amount of work required to fulfill the requirement. Effort may be in the No. of Hours required to complete a task.
6.	Approved (date)	Date of the Approval for development.

Table-1

**3.1 Example: Model Implementation on Flight Management System**

The proposed system is implemented on a real world scenario to have a better understanding of the architecture and to emphasize on the unique benefits of it. We have taken the Flight Management System Application and applied all our version control proposed methodologies with the particular emphasis on the Effective Requirement Management.

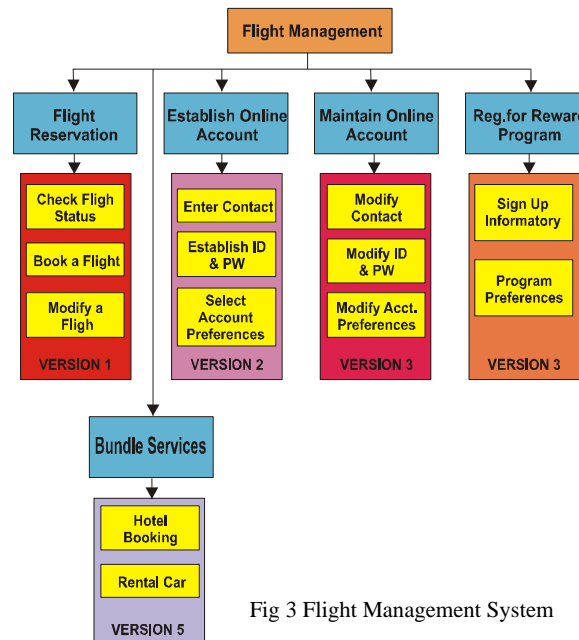


Fig 3 Flight Management System

#### 4. Advantages of Version Approach

This Version based architecture provides several benefits over the techniques used now a day for effective requirement management.

##### 4.1 East to build Test Cases

As requirements are listed with much more detail, Test cases are easy to develop hence ensuring that each requirement is fulfilled. Moreover, versions are tested against the documented Standards and Procedures individually before integrating with other versions. Error location and correction consumes lot of resources, by using iterative version approach error tractability becomes easier.

##### 4.2 Greater in-depth Knowledge

As each requirement is listed in the database, with its priority, functionality along with the amount of effort required to complete that requirement and current status ,every one even the stakeholders have the in-depth knowledge of the development process. Also, because of a central repository, development process is much faster as reducing the need of paper driven intimations and documentations.

##### 4.3 Completely Tested Releases

Most often when working on the unmanaged heap of requirements, development process greatly suffers, result is we are running out of our deadlines and delivered the product without testing it properly. By using version architecture, testing is going on in parallel with version development, making sure that even the smallest of the edge is get refined before the actual delivery.

##### 4.4 Enhance Tractability & Version Comparison

Changes and version comparisons are easy. Future reuse due to central repository has a bigger impact on profit margins.

#### Conclusion & Future

Keeping in view the ever growing and rapidly changing requirements we have proposed a Version Control Requirement Management Model, which is not only efficient but also facilitates in iterative software development and testing techniques. As a result making sure accurate and on time delivery of the application, which not only reduces the visible development cost but also reduces intangible cost.

In the future, we are going to develop a fully automated Version Requirement tool, which is capable of generating run time Analysis and reports based on the proposed methodology. Several efforts are needed in the integration process of these iterative versions as well, because of a several small chunks lot of effective and efficient integration is required.

#### Acknowledgments

I would like to thanks my supervisor Mr. Ali Javed for his excellent guidance, untiring help and endless support throughout this result oriented and well directed research work.

#### References

- [1] Reducing rework through effective requirements management December 2009 Requirements management White paper, IBM.
- [2] CMMI Requirements Management <http://www.software-quality-assurance.org/cmmi-requirements-management.html>
- [3] HP Requirements Management, <http://h20195.www2.hp.com/V2/GetPDF.aspx/4AA2-3219ENW.pdf>
- [4] Dividing Requirements into Iterations for a Staged Delivery by Joy Matthews [www.piersonrequirementsgroup.com](http://www.piersonrequirementsgroup.com)
- [5] Requirements Honesty Francisco A.C. Pinheiro, Universidade de Brasilia
- [6] Requirements Documents that Win the Race Kirstin Kohler & Barbara Paech, Fraunhofer IESE, Germany
- [7] Engineering of Unstable Requirements using Agile Methods James E. Tomayko, Carnegie Mellon University
- [8] Complementing XP with Requirements Negotiation Paul Grunbacher & Christian Hofer, Johannes Kepler University, Austria
- [9] Just in Time Requirements Analysis - The Engine that Drives the Planning Game Michael Lee, Kuvera Enterprise Solutions Inc., Boulder, CO.
- [10] Why Use a Requirements Matrix? Article By Darrel Raynor
- [11] Requirements Management in Global Software Development Rafael Prikladnicki, Jorge Audy, Roberto Evaristo, School of Computer Science, United States.

## **Biography**

### **First Author**



Engr. Imran Iqbal has been working as a Network & Database Developer, since August 2008, in Scotmann Pharmaceuticals, Islamabad, Pakistan. He is currently doing M.Sc in Software Engineering from University of Engineering and Technology Taxila, Pakistan. He has received his B.Sc Degree in the field of Computer engineering from COMSATS Institute of Information Technology, Pakistan. His areas of interest are Software Application Development, Advanced Databases, Digital Image Processing, Wireless and Adhoc Networking, System Engineering and Software Requirements Analysis.

### **Second Author**



Engr. Naveed Abbas Mirani has been working as a Database Administrator, since January 2003, in POF Wah Cantt. Pakistan. He is currently doing M.Sc in Software Engineering from University of Engineering and Technology Taxila, Pakistan. He has received his B.E (Computer Systems) from MUET Jamshoro, Pakistan. His areas of interest are Advanced Databases, Software Quality Assurance, Computer Vision, Software Requirements Analysis and Adhoc Mobile Networks.

### **Third Author**



Engr. Ali Javed has been a lecturer since April 2008 in the Department of Software Engineering, University of Engineering and Technology Taxila, Pakistan. He accomplished his M.Sc in Computer engineering from university of Engineering and Technology Taxila, Pakistan in February, 2010. His areas of interest are Video Summarization, Digital Image Processing, Computer vision, Software Quality Assurance, Software testing and Software Requirements Analysis.