Reliable Requirement Specification: Defect Analysis Perspective

Sandeep Kumar Nayak[1,*], Raees Ahmad Khan[2], and Md. Rizwan Beg[3]

[1] Department of Computer Science and Application, Integral University, Lucknow – 226 026, UP, India
[2] Department of Information Technology, Babasaheb Bhimrao Ambedkar University (A Central University), Lucknow, India
[3] Department of Computer Science and Engineering, Integral University, Lucknow, India
nayak_sam@rediffmail.com, khanraees@yahoo.com, rizwanbeg@gmail.com

Abstract. Reliability is a primary concern for the successful software development organizations. There is various threats point in the requirement phase that causes for requirement defects and so defect occurring in the further phases. A key aspect of delivering and improving the software reliability it is necessary to be confident that the requirement delivered to the further phases of SDLC must be reliable. A reliable requirement can be produce only after removing or resolving all types of requirement defects. In this paper we are proposing a Reliable Requirement Specification Procedure and concrete analysis process which will help out in producing the Reliable Requirement Specification for the designer.

Keywords: Reliable Requirement Specification, Reliability Assessment, Defect Analysis.

1 Introduction

Ineffectiveness of software Reliability Management in Software Life Cycle is the main cause of reliability faults in the completed software system. Stringent analysis, testing and managing of software reliability should be carried out at the initial stage of System Development Life Cycle (SDLC) [1]. According to Roger S. Pressman and Robert B. Grady the cost and effort incurred in finding and fixing the defects are 1% at requirement phase which is much more less than to fixing at test and deployment phase i.e. 15% and 80% respectively [2]. Both the authors surveyed various industries for elaborating defects they found that more than 50% defects are related to Requirement Phase that means

Requirement Defects = (> 0.5) * Total No. of Defects [2]

For proper defect identification the requirement Document must be written in either form:

1) Textual Form 2) Matrix Form 3) Diagram Form 4) Models Form.

* Corresponding author.

P.V. Krishna, M.R. Babu, and E. Ariwa (Eds.): ObCom 2011, Part II, CCIS 270, pp. 740–751, 2012.

There are some specific characteristics which must be fulfilled at the earlier phase of the requirement gathering and preparation so that the defects may be identify at very starting point. The characteristics are as follows:

i. Accurate: Accurate requirement data is free from error.
ii. Complete: Complete requirement data contains all of the important facts.
iii. Economical: Requirement data should be relatively inexpensive to produce.
iv. Flexible: Flexible requirement data can be used for a variety of purposes, not for just one.
v. Relevant: Relevant requirement data is important to the decision-maker.
vi. Simple: Requirement data should be simple to find and understand.
vii. Timely: Timely requirement data is readily available when needed.

Verifiable: Verifiable requirement data can be checked to make sure it is accurate.

2 Accessible Exertion

There are several studies conducted by different researchers for producing reliable software through error removal in code lines and software testing. But there are only few researchers who have given time in defect detection and removal in the requirement phase for delivering the reliable requirement specification. Few authors have given four ways to detect defects a) Checklist Based Detection b) Scenario Based Detection c) Perspective Based Detection d) Traceability Based Detection by [3], some authors depend upon "Defect Density" Model and Design Phase Analysis for defect detection [4], some emphasis on classify the defect similarities and their patterns[5], some researcher narrates to detect the defects phase wise as a) Elaboration b) Inception c) Construction d) Transition[6] and also detected defect through identification of risk item in the requirement document, establishing relationship between defects and their causes and by recording the requirement defects[7].

3 Proposed Approach

In this paper we are proposing a model (Fig.1) for proper analysis of requirement for producing Reliable Requirement document and also performing a review process for rechecking the analyzed document. In this paper we are stepping towards a concrete refinement process i.e. Requirement Analysis and Document Review for reaching maximum probable reliable requirement document. Through this approach designer will feel easiness to construct the most fitted and operable design for the further phases of the SDLC.

Generally Requirements are initiated by the customer or clients for the proposed project. Requirement analysts will be responsible for the preparation of proper methodology for gathering the requirements so that the requirement can be prepared. There are some tips for the requirement preparation:

i. Examine the Project Need or Opportunity
ii. Project Objectives Statement
iii. Categorized the requirement into actual Needs and Wants.
iv. Negotiate the Definition of Requirements in accordance with Customer

v. Requirement validation
vi. Requirement Gathering Techniques

Processed Requirements must be describe under some of the vital and specific heads such as: Budget, Risk, Project description, Data requirements, Functional process requirements, Security requirements, Recoverability requirements, System availability requirements, Performance requirements, Capacity requirements, Glossary/ Metadata, Overall product Functionality, Interface, Attributes, Barriers.

The proposed model for Requirement Analysis contains some of the essential steps through which gathered requirement will be passing and generates Reliable Requirement. Reliable Requirement will produce through assessing the reliability of the requirement which will be further delivered at the designer's end for proper structuring of the project. Actually, this model refers to a process of generating such defect less requirement specification (Reliable Requirement) prior to deliver for the design that:-

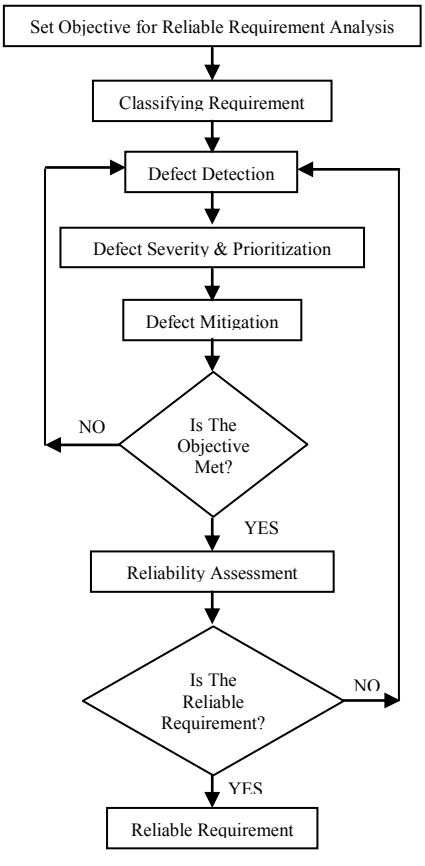

Fig. 1. Proposed Model for Reliable Requirement Analysis

i. Satisfies the user's needs with respect to the proposed system
ii. Provides sufficient information to construct the proposed system

Generally the results of requirement analysis are not directly presentable to a customer, who is usually an expert in his domain area but has little understanding of the completed documentation. Requirement analysis process is a concrete approach for analyzing the described and structured requirement properly.

3.1 Objective of Requirement Approach

Prior to start analyzing requirements there must be some objectives to be achieved by the Requirement Analysis Process are as:

i. Improve Project planning for test Activities
ii. Increase Defect Identification and Risk Assessment
iii. Increase Requirement Reliability
iv. Decrease Requirement Defect Density
v. Reduce Unconventional Requirement
vi. Increase Requirement productivity
vii. Measurement of Reliability of Requirement
viii. Effective Defect Mitigation
ix. Define and Refine all types of requirements in reference to reliability parameter
x. Introduce absolute process to define functional parts of the software
xi. Filter the top level architecture and requirements artifacts
xii. Produce practically analyzed structured documentation

Generate approval criterion to analyzed documentation

3.2 Requirement Classification

After setting the objective for the requirement analysis initiate the classification of requirement where requirements must be decomposed under some specific heads:

1) Requirement Scope, 2) Feasibility, 3) Verifiability, 4) Input Section 5) Problems/Issues, 6) Proper Solution, 7) Execution Plan, 8) Functional issue, 9) Human Resource/Team, 10) Output (Expected and Actual) and 11) Requirement Boundaries

Instead of these specific heads some sorts of tips for classifying the Requirements are given below:

i. Classify the progressed requirement at granularity level so that the team members may also infer of what is happening.
ii. Classify requirement data and their operations.

iii. Structuring the relationship and dependency among the Requirement.
iv. Designing and identification of metadata, data dictionary and data objects.
v. Define Requirement Category, Relationship and Association Category
vi. Transitions among the different Classified Requirement

3.3 Defect Detection

Requirement Defect should be identify under each of the classified requirements. There are some of the causes for unwanted emergence of defects:-

1) Loosely Explained Requirement 2) Erroneous user's Requirement 3) Improper Recording of Requirement 4) Improper Requirement Structuring 5) Lack of proper technique to requirement processing 6) Less involvement of organizational vision

There are some guidelines for narrating the defects.

i. Defect Recognition in classified requirement
ii. Defect Investigation Technique
    - Prototyping
    - Checklist Matching
    - Inspection Technique
iii. Defect Type
    - Team oriented (Communication, Participation, Domain Knowledge, Process Execution)
    - Process Oriented (Improper Methodology to achieve Objective, Mismanagement in selection of proper Process, Poor Elicitation)
iv. Defect Location and Description
v. Defect Severity

3.4 Defect Severity and Prioritization

Defect Prioritization can be defined in different ways:

i. Defect Prioritization is associated with schedule to resolve the defects e.g. out of many issues to be tackled, which one should be addressed first by the order of its importance or urgency.
ii. It is a pointer towards the importance of the Requirement Defect.
iii. Priority refers to how soon the requirement defect should be fixed.
iv. Priority to fix a requirement defect is decided in consultation with the client

In Table 1. the levels of defect priority, their impact and the possible measures are given. The below given chart (Table 2) mention the defect severity leveling and prioritization allotted to each and every severe defect and conclusion reflects all possible combinations and resultants of this phase in Requirement Analysis.

Table 1. Defect Priority with their Impacts and Measures

Defect Priority Level	Impact	Measures
Urgent(P1)	• Core functionality fails • Test execution is completely blocked.	• Fix the defect urgently
High(P2)	• Some of the functionality fails • Lower the execution process	• Fix the defect as soon as possible
Modest(P3)	• An important functionality fails but we don't need to test • it right away	• Fix the defect soon
Low(P4)	• Label missing, spelling or non-logical mistake	• Don't fix this defect before the high and medium defects are fixed. • Fix the defect any time

3.5 Defect Mitigation

Defect priority indicates the impact on the test team or test planning. If the defect blocks or greatly slows down test execution then these defects should have highest grade for the defect priority. Some of the key aspects must be considered before fixing the defects:-

i. Software Product Type
ii. Associated Risk
iii. Defect Investigation Process
iv. Cost of Fixing the Defect
   - Number of Defect which are Fixed
   - Number of team members involved in fixing the Defect
   - Time taken by the Defect Fixing Process

Table 3 represents expected defects under some classified requirement which is labeled by severity and priority. On the basis of defect severity and priority defects may be fixing or not or action may be taken later depends on the Preventive Measures Guidelines for requirement defect mitigation are:-

i. Complete and clear communicated data must be processed.
ii. Apparent dissemination of all functions and information flow with expected operational input and output.
iii. Feasible and obvious vision of expected system.

iv. All types of text, images, graphics, quantitative operations and expected functional characteristics must be noticeable.
v. The scope of specific operational input must be defined well.
vi. A "Defect Sink" must be formed to contain all types of requirement defect so that it may not affect the rest of the requirement.
vii. For defect mitigation there must be a proper checklist or prototype created by requirement analyst for identifying and resolving the defect.
viii. Periodic scheduling must be there for preventing future defect.
ix. Requirement changes must be traceable.
x. Cross-functional requirement data must be identified separately.
xi. Interdependent requirement defect must have the highest priority for mitigation.
xii. Non-dependent requirement defect must have conditional priority among their domain.
xiii. Requirement must be classify at granule level

3.6 Requirement Classification

Since in this paper we are discussing Requirement Defect Analysis Perspective so there will be some aspects on which the reliability of the Requirement may assess.

DD -- Number of Defects Detected
DM -- Number of Defects Mitigated
DR -- Number of Defects Unhealed
T1 -- Total time taken in defect detection
T2 --Total Time taken in Defect Mitigation/Rework

Failure rate = Total no of Defects Detected/Total time taken in defect detection (DD/T1)
Repair rate = Total no of Defects Mitigated/Total time in Defect Mitigation/Rework (DM/T2)
Mean Time to Failure (MTTF) = 1/ Failure rate
Mean Time to Repair (MTTR) = 1/ Repair rate

With the quantification of MTTF and MTTR we may identify at what extent the requirement defects have mitigated and number of defects as unhealed. Through this we may also assess the percentage of unhealed requirement and so the reliability of the requirement.

3.7 Reliable Requirement

This is the final outcome of the proposed model for Reliable Requirement Analysis. This phase will deliver a Reliable Requirement at the early stage of software development life cycle. Maximum of the Requirement defects which may create problems in structuring the operational parts of the design are removed or fixed for delivering the Reliable Requirement Specification.

4 Review

This is the second level of refinement process of the reliable requirement specification in terms of technical review (Fig 2). This review process is related with floating invigilation to detect deficiencies and risks such as:

4.1 Document Classification

The review process is executed by minimum of three to five requirement analysts or two or more group of minimum two experts. Whole document will be classified under different article for simplification of the review process by the experts. They will classify the reliability requirement specification document on the basis of their analytical and management skills.

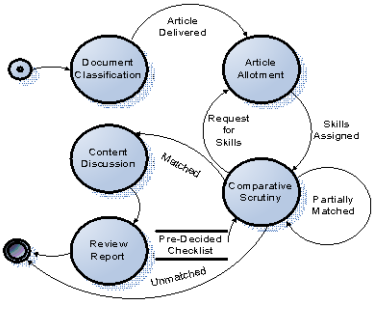

Fig. 2. Second Level Review Process

4.2 Article Allotment

Classified articles are allotted to each skilled member of the review committee to review the article. And also distribute the pre decided checklists along with the concerned article.

4.3 Comparative Scrutiny

Although some of reliable requirement specifications have been discussed in my earlier publication [8]. After receiving a copy of article along with the pre-decided checklists, reviewer starts scrutiny on the basis of the checklist. The reviewers individually inspect the objects (budget, risk and user requirements), classes and their features on the basis of the pre-decided checklist. They must write down their observations through considering the details that are needed for the review report. The allocation of articles and the pre-decided checklist must be included in the review report.

This comparative scrutiny based on Pre-Decided Checklist will be treated as surface review of overall reliable requirement document:-

i. Suitable standard requirement documentation, fact & figures, tables, and diagrams labeled and referenced
ii. Consistent, adequate and appropriate detailing of requirement
iii. Reliability Characteristics of Requirement is mentioned
iv. Confidence Level in terms of reliability
v. Acceptable reference link and dependencies among the requirement and user characteristics description
vi. All functional & non-functional requirement and their input, output, functions & features specified
vii. Required hardware, software, environment and conditions for operation are specified
viii. Instructions for all phases of System Development Life Cycle specified
ix. All types of requirement features included
x. Is there any constraints or barriers in implementing the software defined as:- 1) Regulatory policies 2) Hardware limitations 3) Interfaces to other applications 4) Parallel operation 5) Audit functions 6) Control functions 7) Higher-order language requirements 8) Criticality of the application
xi. Quality attributes (efficiency, flexibility, interoperability, maintainability, portability, usability, availability) correctly specified
xii. Time-critical functions and timing criteria identified for reliable requirement

4.4 Content Discussion

Whenever the Reliable Requirement Specification Document are reviewed with the help of provided checklist then this document must need to integrated at this stage where one to one or face to face discussion are needed to summarize the findings of scrutinized document which is studied by different reviewers. They also determine the contents which must be included in the review report.

4.5 Final Reliable Requirement Specification

The approved output of the review process in the form of review report must be delivering for finalizing the Reliable Requirement Document. This reviewed document may be treated as the final Reliable Requirement Specification Document which is further deliver to the design phase of the System Development Life Cycle. Now the designer may feel easiness in handling this document for constructing the actual design for developing the software.

We would like to draw your attention to the fact that it is not possible to modify a paper in any way, once it has been published. This applies to both the printed book and the online version of the publication. Every detail, including the order of the names of the authors, should be checked before the paper is sent to the Volume Editors.

Table 2. Defect Severity & Priority with Conclusion

Defect Severity	Definition	Prioritization Based on Defect Severity	Conclusion
Critical (S_1)	This type of requirement defect is extremely severe and may capable to halt the entire system. If this requirement is implemented then the software application will refuse to execute or crooked of data.	• Urgent (P_1) • Low (P_4)	$S_1 + P_1$ If the requirement defect for short term application (Daily or weekly) and may halt the system then it should be fix urgently. $S_1 + P_4$ If the requirement defects for long term application (yearly or more) then fixation process may tolerate till the next release or enhancement.
Major (S_2)	This type of requirement defect is also a severe and may capable to halt some functions of the system with serious degradation in performance. Although this is also a failure, but some operations carry on at a lower rate of performance.	• High (P_2) • Urgent (P_1)	$S_2 + P_2$ If logical issue of requirement and lower rate performance defect but not halted the system defect may be fix in very second time (but not urgently) $S_2 + P_1$ If major severity of requirement and may halt the system then it should be fix urgently.
Minor (S_3)	This type of requirement defect is not so severe but caused a low-level disruption of the system or malfunctioning of some operations. This kind of disruption or non-availability of some functionality can be acceptable for a limited period. This type of disruption may cause for failure, but it continues to operate for short span of time at a lower rate of performance.	• Modest(P3)	$S_3 + P_3$ If some operational requirement disrupted which will not affect the core business logic then these defect may resolve in next release
Cosmetic (S_4)	This type of defect is one that mainly related to the appearance or the layout of the data which has no risk of corrupting the statistics and incorrect values. Since it converse about the headline, flag, marker and Colors so it has no actual effect on the operations and may carry on operating with system without any degradation in performance.	• Low (P4)	$S_4 + P_4$ There is no risk of corrupting the operational requirement (headline, labeling, tagging etc.) then the requirements are permissible with no degradation in performance, but need some improvement further.

Table 3. Defects under Classified Requirement with Severity & Priority and Preventive Action

Requirement Classification	Requirement Scope	Input Section	Output Section	Requirement Boundaries	Functional Issues
Identified Defect	• Incorrect portrayal for Product • Business Case, behavior or Mission not defined • Unclear needs, goals, objectives • Operational Concepts lacks • Interface Description missing and others	• Input data repository missing • Inconsistency of data naming • Precondition detail missing • Input data dictionary not defined and others	• Output data sink missing • Precondition detail missing • Input data dictionary not defined • Missing detail of expected results and others	• Condition and environment detail missing • Limited cross functional data link • Improper partitioning of modules • Unclear Resource allocation/reallocations and others	• Actor and its role missing • Operational component missing • Component name & State missing • Casual defects of operations and others
Defect Severity and Priority to Defect	(S1) (P1) (P4)	(S1) (P1) (P4)	(S1) (P1) (P4)	(S1) (P1) (P4)	(S1) (P1) (P4)
	(S2) (P2) (P1)	(S2) (P2) (P1)	(S2) (P2) (P1)	(S2) (P2) (P1)	(S2) (P2) (P1)
	(S3) (P3)	(S3) (P3)	(S3) (P3)	(S3) (P3)	(S3) (P3)
	(S4) (P4)	(S4) (P4)	(S4) (P4)	(S4) (P4)	(S4) (P4)
Preventive Action	○ Fix Now ○ Fix later ○ No Action	○ Fix Now ○ Fix later ○ No Action	○ Fix Now ○ Fix later ○ No Action	○ Fix Now ○ Fix later ○ No Action	○ Fix Now ○ Fix later ○ No Action

5 Conclusion

In this paper we proposed a concrete Requirement Analysis process for producing the Reliable Requirement software. The benefits of using this process are-

i. The main purpose of using this Analysis procedure is “To reduce rework”, “Shortening the requirement document Length” and “Make Requirement reliable”.
ii. Defect Location and Description will shorten the search time for defect identification.

Projected preventive measure guidelines provides an approach for early detection of requirement defect and also assist for considering all perspective requirement corner.
The direction of further study is to be developing a framework to Reliable Requirement Life Cycle (A defect Perspective) for early identification and mitigation of requirement defects and a technique for mitigating requirement defect one by one as defect identified.

References

1. Chao, B., Zhu, X.D., Li, Q., Huang, A.C.: Reliability Management in Software Requirement Analysis. In: International Conference on Management of Innovation and Technology, pp. 1104–1107. IEEE Explore (2006)
2. Pressman, R.S.: Risk Mitigation, Monitoring and Management. In: Software Engineering – A Practitioner's Approach, 5th edn., p. 156. McGraw-Hill (2001)
3. Stefan, B., Michael, H.: Software Product Improvement With Inspection: A Large Scale Experiment On The Influence Of Inspection Process On Defect Detection In Software Requirement Documents. In: Proceedings of the 26th Euromicro Conference, vol. 2, pp. 262–269. IEEE Explore (2000)
4. Singh, L.K., Tripathi, A.K., Vinod, G.: Software Reliability Early prediction in Architectural Design Phase: Overview and Limitations. Journal on Software Engineering and Applications 4, 181–186 (2011)
5. Fuping, Z., Aizhen, C., Xin, T.: Study on Software Reliability Design Criteria Based On Defect Patterns. In: 8th International Conference on Reliability, Maintainability and Safety (ICRMS), pp. 723–727. IEEE Explore (2009)
6. Mohan, K.K., Verma, A.K., Srividya, A., Rao, G.V., Gedela, R.K.: Early Quantitative Software Reliability Prediction Using Petri-nets. In: Third International Conference on Industrial and Information Systems, pp. 1–6. IEEE Explore (2008)
7. Lee, E.S., Bae, J.M.: Design Opportunity Tree for Requirement Management and Software Process Improvement. In: International Conference on Multimedia and Ubiquitous Engineering, pp. 395–400. IEEE Explore (2007)
8. Nayak, S.K., Khan, R.A., Beg, R.: A Comparative Template for Reliable Requirement Specification. International Journal of Computer Applications 14(2), 27–30 (2011)