ISSN: 2278 – 7798
International Journal of Science, Engineering and Technology Research (IJSETR)
Volume 5, Issue 9, September 2016

Requirements Verification & Validation in Bee-Hive Model

Pooja S Bhandary [1] **Kishore PV[2]** **Swarnalatha K S [3]**

***Abstract*— Requirements engineering is the process of recursively gathering and analyzing requirements. The goal of the validation phase is to ensure that the requirements are clear and consistent which in turn leads to a sense of clarity about the end product while the verification step ensures that the developed systemmeets the needs of the stakeholders. In this paper we describe the Verification & Validation step as applied in the Bee-Hive Model.**

Keywords—Software Engineering; Requirements Elicitation andAnalysis; Requirements Specification; RequirementsValidation; Requirements Engineering Process Model,

I. INTRODUCTION

The success of any software project is determined by its ability to meet the goal for which it was intended. Requirements Engineering is a phase which enables the developers to identify the needs of the stakeholder and document them so that they can be analyzed and implemented efficiently. Well defined and effective requirements are really important as they form the basis of the framework of a system. The quality of the software product being developed is directly related to the quality of its requirements. The main task for software engineers is to recognize the important requirements from the unimportant ones and also to classify the multifarious requirements into distinct domains. There is a need to identify the relationships among numerous requirements as it gives a clear understanding about them also provides sufficient knowledge in prioritizing the requirements.[1]

II. REQUIREMENTS ENGINEERING

Requirement engineering is the most effective phase of software development process. It aims to collect good requirements from stakeholders in the right way. It is important for every organization to develop quality software products that can satisfy user's needs. It is a complex exercise that considers product demands from a vast number of viewpoints, roles, responsibilities, and objectives. Therefore, it becomes necessary to apply requirement engineering practices in every phase of software development process.
Requirements engineering is a systematic process of developing requirements through an iterative cooperative process of analyzing the problem, documenting the resulting observations in a variety of representational formats, and checking the accuracy of the understanding gained. It calls for the involvement of requirements engineers, customers who commission the system, users who interact with the system and the people who introduce the system in the enterprise. A requirement is defined as a condition or capability that must be met or fulfilled by a system to satisfy a contract, standard, specification, or other formally imposed documents [8]. The requirements defined for a system should be: correct, consistent, verifiable and traceable. It identifies the technological restrictions under which the application should be constructed and run. Whenever a software application is built, be it for the Web or not, the development team has to acquire certain knowledge about the problem domain and the application's requirements. The elicitation and specification of these requirements is a complex process as it is necessary to identify the functionality that the system has to fulfill in order to satisfy the users' and customers' needs. Although there is a lack of a standardized process supporting handling of requirements and guaranteeing the quality of the results, best practice in the development of general software applications provide a set of techniques[3].

III. REQUIREMENTS ENGINEERING PHASES

The Software Engineering Community recognizes major activities with the Requirements Generation process:
The main phases of Requirements Engineering are:

A. Background Research
B. Requirements Elicitation and Analysis
C. Requirements Prototyping
D. Requirements Verification and Validation
E. Requirement specification

A.Background Research

This is the first phase in the requirements engineering model. This can be compared to the feasibility step in the conventional models. But unlike feasibility study which is one dimensional and only focuses on whether the proposed system is cost-effective or not and if it lies within the required budget the background research is carried out in specific areas related to the project [9] . The areas of background research considered here are generic and can be applied to most of the project undertaking.The areas of background research are 1.Applicaton Domain 2.Scope of evolution 3.Organizational Factors 4.Market Trend 5.Scale/Product analysis 6. Safety and security

B.Requirements Elicitation and Analysis

This is a process of deriving the systems requirements through the observation of existing systems, discussions with potential users and procurers, task analysis and so on. This may involve development of one or more system models and prototypes. These help the analyst understand the system to be specified [2**].**

C.Requirements prototyping.A prototype is developed to evaluate the performance of the current system and find out the requirements that are satisfied by the system. This constitutes the have requirements

D.**Requirement Verification and validation**

Here a Venn diagram is drawn between the haveand the want requirements as specified above.Based on the level of intersection the success of that particular area of research is determined. Thelevel of intersection indicates the extent to whichthe prototype satisfies the user's needs andexpectations .To indicate a successful model a certain idealpercentage of intersection is assigned to each areabased on its grade. If the ideal percentage criteria ismet in all the areas then it can be considered acomplete model.

E.Requirements Specification

The activity of translating the information gathered during the analysis activity into a document that defines a set of requirements. Two types of requirements may be included in this document. They are User Requirement and System Requirement. User Requirements are abstract statements of the system requirements for the customer and the end-user of the system. System

Requirements are a more detailed description if the functionality is to be provided.

IV. RESEARCH METHOD

One of the areas where the Bee-Hive model can prove to be more efficient than the other conventional models is that the extent to which the user requirements are met by the software development process is inbuilt into the model. The requirement elicitation process generates the "have" requirements and from prototyping we obtain the "want" requirements. Based on the mathematical Venn diagram model and probabilistic statistics we deduce the success factor which in turn depends on the number of user requirements that are satisfied. With reference to the Bee-Hive paper, consider the validation process. The intersection between the Have and the Want requirements specify indirectly the probability of success.

Here a Venn diagram is drawn between the have and the want requirements as specified above. Based on the level of intersection the success of that particular area of research is determined. The level of intersection indicates the extent to which the prototype satisfies the user's needs and expectations.

To indicate a successful model a certain ideal percentage of intersection is assigned to each area based on its grade. If the ideal percentage criteria are met in all the areas then it can be considered a complete model.

Advantages:

1. If a particular area falls back on the intersection criteria and does not meet it then work needs to be done only in that particular area and the entire process need not be repeated.

2. Based on the grade of a particular area the intersection criteria is allotted. Hence too much time need not be invested in areas where more efficiency is not required.

RANK ALLOTMENT TO THE 6 PHASES OFFEASIBILITY STUDY

There are two ways of allotting rank to one of the 6 phases of feasibility study based on

1. The inherent importance of a particular phase related to the application
2. The mathematical deduction of the rank of each phase dependent on the number on "want" requirements generated.
3. Combination of the above two ways.

    1. Inherent importance rank allotment

       If based on the application to be applied to, the 6 phases can directly be ordered on the basis of their relative importance then this method is used.For instance consider a military application related to nuclear activities, here safety and security takes precedence hence it is allocated the rank 1 so on and so forth.

    2. The mathematical deduction

       The requirement elicitation for each phase generates the number of "want" requirements.Total number of want requirements are calculated:NOTE: The want requirements should be discrete with no requirement common to two or more phases.

       n(W) indicates total want requirements
       n(W)=n(W(Background Research))+n(W(Scope of Evolution))+n(W(Organisation Factor))+n(W(Market))+n(W(Scale Check))+n((Safety and Security))

       The rank is allotted by calculating:

       R(*Field)=n(W(*Field))/n(w)

       Where *Field indicates one of the six phases such that

       ∑R(*Field)=1

In this method the phase that generates the maximum number of "want" requirements has the highest rank.

    3. The combination of the above two methods are used.

    If there are phases corresponding to an application that can be relatively more important than the others then they are given higher ranks, for other phased method 2 is applied.

DETERMINATION OF THE INTERSECTION OF HAVE AND WANT REQUIREMENTS

Consider two sets have and want for each phase where

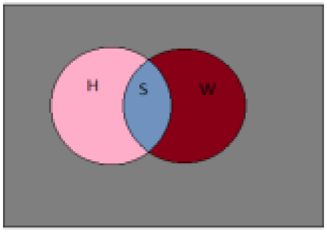
H={x|x is a set of all requirements that the prototype can satisfy}

W={x|x is a set of all requirements that the user has demanded}

n(H) and n(W) denotes the cardinality number of the two sets which indicates the number of elements present in the two sets respectively.

S =H ∩ W={x|x is a set of have requirement that satisy want requirement}

n(S)=Numberof requirements belonging to H ∩ W



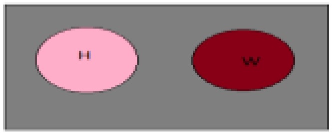
The probability of success can be indicated by P(NR) and is given by:

$$P(NR) = n(S)/n(W)$$

Hence, probability of risk P(R)= 1 –P(NR)

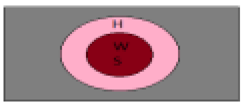
Different cases:

1) Highest Risk Process i.e P(NR)=0 or P(R)=1



2) No Risk process i.e P(NR)=1 or P(R)=0

i.e n(H)>=n(W)



3.) In the rest of the cases depending on the application the success rate should be set i.e n(S) should be set. Some of the apllications require a large number of requirements to be satisfied while some applications don't require a very large number of n(S). So depending on whether n(S) has been achieved or not the validation is either successful ornot.

DECIDING WHETHER THE SOFTWARE DEVELOPMENT PROCESS IS SUCCESSFUL OR NOT

Based on the rank allocated to each phase as discussed above, its P(NR) is calculated as shown above.

A cutoff P(NR) is allocated depending on the rank.

If P(NR) as above the cutoff then the process is successful else modifications should be made to improve the P(NR) ratio of the phase so that it exceeds the cutoff P(NR) for that particular phase.

The cutoff P(NR) is allocated by domain experts.

IF the cutoff P(NR) of a high rank phase is not met then carrying on without modification is very risky and almost guaranteed to fail.

But if the cutoff P(NR) of a low ranked phase is not met then it is more feasible to proceed without medication rather than to go back and repeat the entire process.

Conclusion: The V&V techniques that we proposed is proven to be efficient in terms of verifying and validating requirements based on the type of the application domain.

VIII. REFERENCES

[1] Swarnalatha K S, G.N Srinivasan, Pooja S Bhandary "a constructive and dynamic framework for requirement engineering process model – bee hive model" International Journal of Computer Engineering and Technology (IJCET), ISSN 0976-6367(Print),ISSN 0976 - 6375(Online), Volume 5, Issue 7, July (2014), pp. 48-54 © IAEME.

[2] Swarnalatha K S, G N Srinivasan, Rakesh R, Vipin Dwivedi "software requirement collection enhancement using sampling technique and applying t-distribution" International Journal of Computer Engineering and Technology (IJCET), ISSN 0976-6367(Print), ISSN 0976 - 6375(Online), Volume 5, Issue 7, July (2014), pp. 67-72 © IAEME.

[3]Swarnalatha K S , G.N Srinivasan ,Harshvardhan V Rawoot, Rakesh R, Nishkarsh Singh" Different approaches for Dynamic Requirement Engineering Process Models: Cone and Hour Glass Model" Paper got accepted in IEEE conference 2014(ICAECC-14).

[4] AzlenaHaron and ShamsulSahibuddin, "TheStrength and Weakness of RequirementEngineering (RE) Process," presented at theICCTD 2012: 2012 the 2nd InternationalConference on Computer Technology andDevelopment, Cairo, Egypt, 2012.

[5] B. Mutschler, M. Reichert, and J. Bumiller,"Unleashing the Effectiveness of Process-Oriented Information Systems: Problem Analysis, Critical Success Factors, andImplications," IEEE Transactions on Systems,Man, and Cybernetics, Part C: Applicationsand Reviews, vol. 38, no. 3, pp. 280-291,2008.

[6]Liu, L. and L. Lin, 2012. ARED-CK: Anautomated requirements elicitation approachbased on decision-making with completeknowledge. Proceeding of the 1st IEEEInternational Workshop on ManagingRequirements Knowledge, Sch. of Software,Tsinghua Univ., Beijing, pp: 47-52.

[7]. Mishra, D., A. Mishra and A. Yazici,2012. Successful requirement elicitation bycombining requirement engineering

techniques. Proceeding of the 1st IEEEInternational Conference on the Applicationsof Digital Information and Web Technologies, Dept. of Comput. Eng., Atilim Uni., Ankara,pp: 258-263.

[8]. Ramsin, R. and R.F. Paige, 2011. Iterativecriteria-based approach to engineering therequirements of software development Methodologies. IET Software, 4(2): 91-104.

[9]. Sharp, H., C.W.A. Finkelstein and H. Galal,1999. Stakeholder identification in theRequirements engineering process. Proceeding of the 10th IEEE International Workshop onDatabase and Expert Systems Applications,Centre for HCI Design, City Univ., London, UK,pp: 387-391.

[10]. Singh, Y., P.K. Bhatia and O. Sangwan,2007. A review of studies on machine learningTechniques. Int. J. Comp. Sci. Secur., 1(1): 70-84.

[11]. Wongthongtham, P., E. Chang, T.S.Dillon and I. Sommerville, 2009. Developmentof software engineering ontology formultisite software development. IEEE Trans. Knowl. Data Eng., 21(8): 1205-1217.