A CONSTRUCTIVE AND DYNAMIC FRAMEWORK FOR REQUIREMENT ENGINEERING PROCESS MODEL – BEE HIVE MODEL

Swarnalatha K S¹, G.N Srinivasan², Pooja S Bhandary³
R V College of Engineering
Mysore Road, R V Vidyanikethan Post, Bangalore, Karnataka 560059

ABSTRACT

Despite of many advances in design of complex software development there remains the problem of highly inadequately specifying the requirements form the stakeholders for any real time application. This paper mainly focuses on BEE-HIVE model based process for requirement process model. BEE-HIVE MODEL driven development fits in any evolutionary and conventional prototyping because design and analysis key parts of any software development process. A BEE-HIVE model is applied to increase speed up and also to check the time required to gather the data from the stakeholders in designing the prototype. The BEEHIVE model may ensure the correctness of the timely generated code, which is another important factor in enabling the software development designing the prototype. The BEEHIVE model may ensure the correctness of the timely generated code, which is another important factor in enabling the software development.

Keywords: Bee-Hive Model, Requirement Specification, Security, Prototyping, Elicitation and Analysis.

1. INTRODUCTION

The Success of any software system is the level of degree to which it meets the purpose of the customers or stake holders. Therefore, requirements engineering plays a vital role for successful completion of any projects. Requirements engineering commonly defined as a part of software engineering which is mainly concerned with real world entities, constraints from stake holders on a complex software systems. Requirements engineering facilitates the converts the informal requirements to formal specifications, which serve stakeholders for any real time application. This paper mainly focuses on BEE-HIVE model based process for requirement process model. BEE-HIVE MODEL driven development fits in any evolutionary and conventional prototyping because design and analysis key parts of any software development process. A BEE-HIVE model is applied
to increase speed up and also to check the time required to gather the data from the stakeholders in the basis for design and development of the software [1]. The proposed The Bee-Hive model attempts to find a compromise between two conventional and tested models of requirements engineering which are waterfall model and evolutionary development model in such a way that the advantages of both the models are effectively incorporated into the system and at the same time trying to reduce and overcome the shortcomings and disadvantages [2]. In other words it tries to maximise the gains while minimizing the disadvantages.

2. CONVENTIONAL APPROACH

Requirements Engineering is the process of understanding and defining what services are required from the system and identifying the constraints on the system’s operation and development [3]. It is an extremely important phase as the success of the subsequent phases directly and indirectly depend on the efficiency with which requirements engineering is carried out.

A strong and robust building needs a sound foundation [4]. This analogy can be extended to the software engineering process where the Requirements engineering process can be considered as the foundation [5]. Hence, it is imperative that novel and efficient techniques of requirements engineering should be developed.

The conventional requirements engineering processes have the following disadvantages:

1) Time consuming [7].

2) Too orthodox i.e. either follow the waterfall or iterative development model.

3) Too much documentation [8].

No clear segregation into system, user and domain requirements.

3. BEE-HIVE MODEL -APPROACH

We have tried to overcome the disadvantages of conventional approach by using a novel and innovative model called the bee-hive model. It is called a bee-hive model because the structure at the centre is in the shape of a bee-hive. The fig 1 below shows the overview of BEE-hive model.

4 PHASES OF BEE-HIVE MODEL

4.1 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. Application Domain

Determines the feasibility of the application platform proposed to implement the project. Research is done based on considering the current platforms and facilities available and analysing whether they are good enough to support the project [10]. If the current resources are found inadequate then it is determined whether developing new and application specific resources and facilities are feasible or not. Criteria for choosing the application domain are listed below.

- Available technology.
- The platform to be selected.
- Required software and hardware.
- Decision whether new technology is to be developed or not.
2. Scope of evolution

This phase is used to predict the kind of future the product or the service developed has. This phase is used to predict the changes it might be subjected to in the future. Depending on the level and the rate of changes it might be subjected either time or quality is set as the critical factor. For example, if a software is changed every 3 months then time is the critical factor here. If on the other hand the software is to be used for a longer lifetime quality is the critical factor. Criteria for choosing the scope of the product are listed below.

- The novelty of the product.
- Shelf-life of the product.
- The kind of acceptance garnered by the product (More popular technologies have a better scope of evolution).
- Future need of such a product/technology. (Try and produce a reliable prediction).

3. Organisational factors

Constitutes the organisational policies, laws, staff, growth plan and resources available in the company in terms of capital and technology owned by it. Criteria for choosing the Organisational factors are listed below.

- Management hierarchy.
- Company policy.
- Proprietary products available to support research and technology.
- The financial condition of the company that affects the budget.

4. Market

This phase used to determine the current competition in the market, the need and aspiration for the product or service being developed. Criteria for choosing market for the product are listed below.

- Current Competition in the market.
- Previous response to similar products/technologies.
- The demand and the need of the product/technology to be released.
- The market trends and conditions.

5. Scale check

This basically determines the scale on which the product or service is to be released. Based on the estimated profit margin the scale is determined and correspondingly the amount of money to be invested into it is determined. Criteria to check scalability of the product are listed below.

- The budget for marketing and distribution.
- The investment on the product.
- The expectations out of the product/technology.
- The brand name and the reputation of the company.

Advantages

- Feasibility study can be carried out parallel in each of the above specified areas.
- It does not hold up the requirement elicitation and analysis of other unrelated areas.
- Talent and skills of people can be mobilized. People who are good at a particular area can be allotted work in that particular area.
Based on the application, different areas are given different precedence’s or importance depending on how critical that aspect is to the application.

Based on the importance of the different areas of feasibility study a rank is given to each area and the area with the higher rank is given more importance and more time and effort is spent on its feasibility study. If in a high grade area of say 1 or 2, project is found unfeasible it is scrapped. Between 3 and 4 grade if found unfeasible then it is manageable but advisable to work on and improve the feasibility. Between 5 and 6 grades if risk involved is not too high then the problems in this area are avoided.

6. Safety and Security
Whether the product to be developed is safe to use and whether it can assure security against data theft and loss [11]. Criteria for choosing the safety and security of product are listed below.

- Vulnerability to loss of data or other damages.
- Preserving data integrity.
- Data backup and back up plans for the facility if an accident occurs.
- Recovery mechanisms if something should go wrong.

4.2 Requirements Elicitation and Analysis
This phase is carried out as a continuation to the feasibility study carried out in different areas. Because the Feasibility study was very specific the requirement elicitation and analysis is simplified to a great extent. As mentioned above requirement elicitation and analysis process is sped up because of greater clarity and non-dependence on areas that do not affect it [12]. This phase generates the want requirements which are the requirements that the customer or user expects and requires.

4.3 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.

4.4 Requirements Verification and Validation
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 [13].

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 is met in all the areas then it can be considered a complete model.

Advantages

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

4.5 Requirements Specification
Only those requirements that lie in the intersection region are documented. Hence the documentation cost and time is greatly reduced as requirements that are irrelevant are not
documented [14]. Also as each area generates its own requirements its management is simpler because the volume of data generated is greatly reduced. Each area of background research generated requirements specific to that area [16]. The application domain produces the domain requirements. The market research produces the user requirements. Safety and security produces critical system requirements. Scope and organisation produces system requirements. Hence an easy segregation of requirements is achieved.

5. BEE-HIVE MODEL – ADVANTAGES

The advantage of the Bee-hive model is listed below:

1. It is a combination of both parallel and serial model. While the phases of feasibility study are carried out parallel, the flow of control within a particular area is sequential.
2. It is a good combination of waterfall and evolutionary development model.
3. Each phase in the feasibility study is largely independent of other areas and if one area fails the negative effects on other areas are reduced to a great extent.
4. Helps us to identify the critical requirements and helps us to focus on the important requirements and not waste time on the others.
5. There is a scope for compromise and imperfections in the system as a few non critical errors cannot directly affect the system.
6. Once the feasibility study is complete the rest of the processes can be completed with relative ease.

Future Enhancement

Critics may argue that feasibility study in the requirements engineering process should be quick and easy. But if the added complexity and time consumption can ensure the efficiency of the subsequent processes at large and reduce the time taken by them then the trade-off is acceptable.

CONCLUSION

It is roughly calculate that to fix an any kind of defect found during requirements engineering process model cost twice the modelling and delivering the new product. This asserts the important role of requirements engineering in any software development process. Requirement prototyping plays vital role in requirements elicitation technique that can help identify any defects at an initial stage and make the project more successful. We have proposed a BEE-Hive model based for requirements engineering.

REFERENCES


