

A Survey on Software Process Models

Sonali Thopate^{#1}, Snehal Desai^{#2}, Aparna Jadhav^{#3}, Rasika Mundhe^{#4}, Vishal Soni^{#5}

Vishwatmak Om Gurudev College of engineering

at-mohili, post-Aghai, tal-shahapur

Dist-thane.

¹2519sonali@gmail.com

²snehaldesai1994@gmail.com

³apps1104@gmail.com

⁴rasika.mundhe@gmail.com

⁵Soni.sonivishal.vishal8@gmail.com

Abstract—Software process model is a step by step highly structured technique for designing and development of any software. There are various process models widely used for developing software. Process models give a guidelines regarding development of the software.

Process models are very important for developing the software in a systematic manner, such that it will be delivered within the time deadline and should also have proper quality. There are four major steps which includes Elicitation, Designing, Coding and Testing.

In this paper we develop a new model for software development that lays special emphasis on highly structured lifecycle and defining an output with each stage and also tries to fulfill the objective of the Software Engineering of developing high quality product within schedule and budget. The new proposed model is designed in such a way that it allows client and developer to interact freely with each other in order to understand and implement requirements in a better way using the concept of process model.

Keywords—Software Engineering, software process models, Free-Flow Model, Requirement collection, Testing, Activities involved in process models, Comparative analysis of models.

I. INTRODUCTION

Programming process model is a procedure which gives another method for planning of any product inside less time and gives high caliber of programming item according to the necessity. Programming process gives the better execution in Implementation and building up the product. It disentangled the multifaceted nature of the product and upgrades the nature of item. It incorporates the essential objective to make quality programming as indicated by individuals who execute activity by utilizing already set client prerequisites.

The procedure model gives an arrangement of exercises to framework architects and engineers to take after. The thoughts regarding the product procedure model have been around for quite a while and numerous varieties exist, for example, the Waterfall model, Iterative model, Prototype model, Incremental improvement model, Spiral model. These varieties have numerous renditions fluctuating from those which are simply directing standards, to static frameworks of advancement.

1.1 Requirement Analysis Phase:

Following are the objective of this phase-

To distinguish what sorts of elements are required. This is the procedure of finding, dissecting, characterizing, and archiving the prerequisites. It includes measuring every last utilitarian necessity, non-practical, prerequisite, specialized, operational, and transitional necessities. Necessities examination includes consistent collaboration with customer to decide their desires, giving visual representations, making mappings between last details characterizing limits of the framework. It can be portrayed as a formal assentment between the customer and the organization which unmistakably states as what will be in degree and what is out of extension furthermore pen down Understanding of every gathering over the final result. Major activities-

i) Problem understanding; ii) required specification

Phase ends with a SRS document...

Design & Specification Phase

1st step from problem domain to solution domain

2 stages-i) System Design ii) Detailed Design

System design- defines the general structure, major modules, functions of modules, interface between modules, output formats of the system.

Detailed design- focuses on how each component can be implemented, internal logic to be used, the algorithms to be applied etc.

A Process Model is a set of activities. It creates relationship between activities performed in a manner that satisfies the ordering relationship that will produce desired product. A Process Model is broken down into different activities and specifies how these activities are organized in the entire software development process.

II. WHAT IS PROCESS MODEL

A process model is either a descriptive or prescriptive characterization of how software is to be developed. A process model defines the history of how a required software system was developed. Process models used for the understanding and improving skill of software development processes.

The characteristics of process model are as follows:-

- It gives the outline for what documents produce for delivery to client.
- It gives basic information about cost and overall quality of software product.
- It gives the guideline to organizer to organize, plan, budget, schedule and manage software project work.

Software processmodel can represent multiple interconnected task chains [1]. It includes the following task action:-

- It develop representation of a non-linear sequence of actions and transform available resources into intermediate product.
- It identifies attributes and operations on the objects.
- It creates interfaces between objects, attributes, or operations.

III. DIFFERENT SOFTWARE PROCESS MODELS

3.1. WATERFALL PROCESS MODEL:

The Waterfall Process Model is also called as classical life cycle model [2] was the first process model to present sequential framework, describing basic stages for successful software development model. It states that the phases are organized in a linear order.

The sequence of activities-

- Requirement analysis, Project planning, System designs, Details design, Coding, Unit testing, System integration & testing.
- To clearly identify the beginning and end of the phases verification & validation activities are required.
- Output of the phase should be consistent with its input (which is the output of another phase) and with the overall requirements of the system.
- Work products or intermediate products are produced as outputs, which are in the form of documents.

- It is the most widely used process model.
- It is well suited for projects where the problem domain and the requirement specific are well understood...

Two basic assumptions-

- All phases listed in the waterfall model should be performed anyway.
- Any different ordering of the phases will result in a less successful software product.

Project outputs-

Requirement document, Project plan, System design document, detailed design document, Test plan & test reports, Finalcode, Softwaremanuals, Review reports.

Advantages of waterfall model-

- Simple goal and easy to understand and use.
- Easy to understood milestones.
- Easy to understand all stages and implement product.

Limitations-

- It assumes that the requirements can be frozen(baseline) before the design begins.(not practical)
- Freezing the requirements requires choosing the hardware, which on completion of the project may become obsolete.
- It requires that the requirement be completely specified before the rest of the development can proceed, but in some systems it can be desirable to develop a part of it, and later enhance the system.
- Document driven process, not suitable for interactive applications.
- Cost/schedule.
- Customers/End users already know about it.

3.2. PROTOTYPE MODEL:

A disposable model is worked to comprehend the necessities. It is produced in light of the as of now known prerequisites. The customer can get a genuine feel of the framework, on the grounds that the connections can empower better comprehension of the necessity. This outcome into more steady prerequisites that can change less every now and again. Can be utilized for confused and extensive frameworks where there are no manual procedures or existing frameworks accessible. The model needs to experience improvement stages which is done casually. A model is produced, after sensibly understanding the framework, customers communicate with it, give input (adjustments, missing elements).Prototype is altered to consolidate the progressions, again customers can collaborate; criticism; model changed. The cycle rehashes itself; until time and cost of changing the framework and getting the food input are surpassed Prototype incorporates prerequisites which are not well understood. For minimizing cost, fewer docs ought to be included, trying ought to be

reduced. At times the expense of advancement without model is more than improvement with a model.

Advantages of prototype model-

- Provides a good way to determine requirements.

Limitations of prototype model-

- The developers make implementation to make the prototype work in fewer periods by using the operating system or programming language [3].

3.3. INCREMENTAL DEVELOPMENT MODEL:

Rather than deliver the system as a single delivery, the development and delivery is broken down into increments with each increment delivering part of the required functionality. User requirements are prioritized and the highest priority requirements are included in early increments. Once the development of an increment is started, the requirements are frozen though requirements for later increments can continue to evolve

Advantages:

- Customer value can be delivered with each increment so system functionality is available.
- Early increments act as a prototype to help elicit requirements for later increments.
- Lower risk of overall project failure.
- The highest priority system services tend to receive the most testing

Limitation of incremental model-

- In this model is rapid context switching between various activities very complex [4].

3.4. SPIRAL MODEL:

Activities in this model are arranged in spiral that has many cycles. The spiral model can be divided into 4 quadrants.

1st Quadrant: Determining objectives, alternatives & constraints.

2nd Quadrant: Evaluation of these alternatives, identifying & resolving the risks involved.

3rd Quadrant: Develop strategies and verify.

4th Quadrant: next phase is planned.

A cycle begins with identifying the objective, different alternatives and the associated constraints. The process from the center of the spiral proceeding in the clockwise direction. There are various tasks regions in this model. The radial dimension represents the cost incurred; the angular dimension represents the progress in each activity of the cycle. Each cycle is completed by the review of the activities of that cycle including the next phase. This is the preferred model for high-risk projects. This model can be used for development as well as enhancement projects. In this model [5], presenting a sequence of activities with some techniques from one activity to the other.

Advantages of spiral model-

- Requirements can be checked more perfectly.

Limitation of spiral model-

- It is difficult to contract deadlines using this model.

3.5. ITERATIVE ENHANCEMENT MODEL:

It combines the benefits of the prototype model and the waterfall model. The software is developed in increments, each increment adds a functionality to the system until the entire system is built. Implementation of the subset of the overall problem, which capable of forming a usable system is the 1st step. Project Control List is maintained, which has list of all the tasks reqd for final implementation. It helps in minimizing the chances of errors, and reduces the redesigning of work.

Advantages-

- It results in better testing; testing increments is easier than testing the system as a whole.
- These increments can also provide feedbacks from the clients.
- Early increments can be implemented with fewer people, later additional staff can be included.
- Increments can be planned so as to manage technical risks.

IV. CONCLUSION

The paper has demonstrated interrelationship between software products and production process. It is probably the most time-consuming project management activity and also includes continuous activity from initial concept through to system delivery. The above study gives a clear understanding that various process models when employed for developing different software then they may generate successful results. There are many process models such as Waterfall, prototype, incremental, spiral, iterative enhancement model. All these different software process models have their own advantages and disadvantages

REFERENCES

- [1] Kling, R., and W. Scacchi, The Web of Computing: Computer Technology as Social Organization, Advances in Computers, 21, 1-90, Academic Press, New York, 1982.
- [2] W.W. Royce, "Managing the Development of Large Software Systems: Concepts and Techniques", IEEE, IEEE Computer Society, August 1970, pp. 1-9.
- [3] Ian Sommerville, "Software Engineering", 8th Edition, 2006, pp. 89.
- [4] R.S. Pressman, "Software Engineering, A Practitioner's Approach", 5th ed. New York: McGraw-Hill, 2001, pp. 32.
- [5] R.J. Madachy, "Software Process Dynamics", New Jersey: Wiley Inter science, 2007, pp. 33.