



# A Detailed Software Model Analysis

Navya Anna Sunny<sup>1</sup>, Nisha Yaday<sup>2</sup>

Student, Department of Information Technology, Maharishi Dayanand University, New Delhi, Delhi, India

navya sunny3@yahoo.co.in&yadavnisha651@gmail.com

#### Abstract:-

Software's move along a series of transitions involving their commencement, early development, productive operations, maintenance and repairs as well as retirement. For this, there are a multiple techniques that are applied which are known as software process models or software life cycle models. This paper represents five most widely implemented models namely Waterfall, Incremental, Spiral, V-Shaped and Agile. The main objective of this paper is to elaborate these models and also to list their advantages, disadvantages and when to use which one.

#### **Keywords:**

Comparative Analysis of Software Models, Software Development, Software Development Models, Software Life Cycle Models.

#### I. **INTRODUCTION**

Software is engineered or developed, it is not manufactured. Unlike hardware it is logical rather than physical.

A software development project is a structure imposed on the development of a software product. A software development life cycle is typically a framework that describes the activities performed at each stage of a software development project. Explicit models of software development date back the earliest projects to developing large software systems in the 1950's. The purpose of these software development life cycle models was to

provide a theoretical plan for realistically managing the development of software systems. For this, there are several model each describing approaches to a variety of tasks or activities that take place during the process of software development. These models aim to be the standard that defines all the tasks necessary for developing and maintaining software's. The models that we will take into consideration are:-

- 1. Waterfall Model.
- 2. Incremental Model.
- 3. Spiral Model.
- 4. V-Shaped Model.

Before analyzing these models, we will first study various activities related to software development process.

#### II. **SOFTWARE DEVELOPMENT ACTIVITIES**

Software's undergo a lot of problems during their development. Problem solving in software's consisting of the following activities:-

- 1. Understanding the problem.
- 2. Deciding a plan for a solution.
- 3. Coding the planned solution.
- 4. Testing the actual program.

Each of these activities can be very large



International Journal of Research (IJR) Vol-1, Issue-10 November 2014 ISSN 2348-6848

and handling of them is not possible in a single step. Hence, there is a need to break them into many smaller units so as to manage them efficiently. Every model gives us a unique way to perform these activities and to make software development much easier.

#### III. SOFTWARE PROCESS MODELS

#### A. Waterfall Model

This model is named waterfall model because its diagrammatic view displays a cascade of waterfall. It is a sequential development approach in which development is seen flowing through all analysis the phases such as of implementation, requirements, design, approaches testing. integration and maintenance. This model is easy to understand and follows the notion of "define before design" and "design before code".



#### Fig. 1 Waterfall Model

The advantages of waterfall model are:-

- Easy to understand, easy to use.
- Provides structure to inefficient staff.
- Every stages objective is well understood.
- Sets requirement stability.
- Works well on mature products and weak teams.
- Good for management control (planning, staffing, tracking).
- All requirements must be known upfront.
- · Milestone created for each stage is

considered frozen – inhibition of flexibility.

- May give a false impression of progress.
- Does not reflect the problem solving nature of software development.
- Little opportunity for the customer to preview the system.
- Real projects rarely follow this approach.
- Costly for small teams and projects.
- Difficult to integrate it with risk management.
- Moving a step backwards is very difficult and creates a lot of problems, i.e., if something goes wrong with the design phase then there may be serious problems with the implementation phase.

#### B. Incremental Model

incremental model the whole In requirement is divided into various constructs. Multiple development cycles take place, which makes the incremental life cycle transform into a "multi waterfall" Systems has been cycle. developed through incremental releases requires the provision of essential operating functions, which then provides system users with more improved and more skilled versions of a system at regular intervals. Every cycle is divided into smaller, more easily manageable module. Each module passes through the requirements, analysis of design, implementation and testing phases. A



version of software is produced which is working at the end of the first module, so the consumer has working software in the beginning of the development process. Each subsequent release of the module adds function to the previous release. The process continues untill the complete version of the system is achieved.



#### Fig. 2 Incremental Model

The advantages of incremental model are:-

- · Generates working software quickly.
- · More flexible.
- Easier to test and debug.
- Customer can respond to each built.
- Lowers initial delivery cost.
- Easier to manage risk because risky pieces are identified and handled during ith iteration.

The disadvantages of incremental model are:-

- Needs good planning and design.
- Needs a clear and complete definition of the whole system before it can be broken down and built incrementally.
- •
- Total cost is higher than waterfall model.

#### C. V-Shaped Model

The V – Shaped model is a sequential path of execut ion of processes. It is a variant of

waterfall model that emphasizes on verification and validation of the product. Each phase must be completed before the next phase begins. Product testing is planned in parallel with the corresponding phase of development. Requirements are the beginning of the life cycle model just like the waterfall model. The high-level focuses design phase on system architecture for which an integration test plan is created in order to test the pieces of the software systems ability to work together. However, the low-level design phase lies where the actual software components are designed, and the unit tests are carried out in this phase as well.



## Fig. 3 V - Shaped Model

The advantages of v - shaped model are:-

- Emphasize planning for verification and validation of the product in early stages of product development.
- Each deliverable must be testable.
- Project management can track progress by milestones.
- Easy to use and implement.
- Higher chance of success over the waterfall model due to the early development of test plans during the life cycle.

International Journal of Research (IJR) Vol-1, Issue-10 November 2014 ISSN 2348-6848

The disadvantages of v - shaped model are:-

- Does not easily handle concurrent events.
- •
- Does not handle iterations or phases.
- •
- Does not easily handle dynamic changes in requirements.
- Does not contain risk analysis activities.
- Little flexibility.

#### D. Spiral Model

The spiral model of software development represents a risk driven approach to software process analysis. This approach incorporates the elements of specificationdriven, prototype-driven process methods, together with the classic software life cycle. It has two main distinctive features. First is its cyclic approach for incrementally growing a system's degree of definition and the second one is the implementation while decreasing the degree of risk. The requirements are gathered during the planning phase. In the risk analysis phase, a process is undertaken to identify risk and alternate solutions. A prototype is produced at the end of the risk analysis phase. Software is produced in the engineering phase, along with testing at the end of the phase. The evaluation phase allows the customer to evaluate the output of the project before the project continues to the next spiral.



#### Fig. 4 Spiral Model

The advantages of spiral model are:-

- Provides early indication of insurmountable risks without much cost.
- Users see the system early because of rapid prototyping tools.
- · Critical high-risk functions are developed first.
- The design does not have to be perfect.
- Users can be closely tied to all lifecycle steps.
- Early and frequent feedback from users.
- Cumulative costs assessed frequently.

#### **IV. CONCLUSIONS**

After conducting the analysis of the various process models that are being implemented for more than 4 decades, one can conclude that each and every model has its advantages and disadvantages and are used according to the situation. The best part about these models is that one may use them the way the development team wants. One can create new process models as per its requirement. The waterfall and spiral models are still the most commonly used models. But with



International Journal of Research (IJR) Vol-1, Issue-10 November 2014 ISSN 2348-6848

increased urgency among customers, the agile model is also being used a lot. Hence, it can be said that all the models are effective enough to be implemented as per the requirement. It depends on the developer which one to use.

### V. ACKNOWLEDGMENTS

We would like to thank the entire faculty of IT Department, Dronacharya College of Engineering, Gurgaon, Haryana for their support, encouragement and patience.

#### VI. **REFERENCES**

- [1]Nabil Mohammed Ali Munassar and A. Govardhan, "A Comparison Between Five Models Of Software Engineering", IJCSI International Journal of Comput er Science Issues, Vol. 7, Issue 5, September 2010, pp. 94 – 101.
- [2]Walt Scacchi, Process Model in Software Engineering, New York: John Wiley and Sons, Inc., 2001.
- [3]Dr. Deepshikha Jamwal, "Analysis of software De velopment Models", International Journal of Compute r Science and Technology, Vol. 1, Issue 2, December 2010, pp. 61-64
- [4]Ashwini Majumdar, Gayatri Masiwal,
  P.M.Chawan, "Analysis of Various Software Process Models"
  International Journal of Engineering Research and Applications, Vol. 2, No. 3, 2012, pp. 2015-2021.

[5]http://www.istqbexamcertification.com [6]http://www.executivebrief.com