

# Analysis of Software Engineering Models For Software Development and Quality Management

Sandeep kaur  
 Sandeep\_multani@yahoo.com  
 Student of M.tech (cse)  
 Lovely Professional University,phagwara,India

**Abstract.** *This paper presents the comparison of two models of software engineering for developing a software product and quality management. It is deal with the software development through the development models, which are known as software development life cycle (SDLC). The main objectives of this paper to show process comparative analysis between two models water fall model and iterative models of software engineering by showing their flexibility to develop a good software product..*

**Keywords-** software development life cycle (SDLC), water fall model, iterative model, process comparative analysis, quality management.

## 1. Introduction

The software development life cycle is the well defined process in which the software designers, database designers, database developers and users collaborate to build the application software. It involves designing the application from scratch documenting everything adding the improvements and fixing the bugs that occur in SDLC.[3]

A quality of software system is based on the three principles:

1. Software development process.
2. Measurement of product
3. Management and human interactions.

The software development life cycle may be divided into the following steps:

1. Original idea from customer requirements.
2. Creation of feature lists based on customer requirements.
3. Input /output design based on technical design.
4. Implementation of code
5. Test the software product
6. Releases the beta version of software product and fix the bugs.
7. Releases the final version of software product.

The success of the software largely depends on proper analysis, estimation, design and testing before the same is implementation. All the information relevant to the system and its scenarios, the occurrence and the behaviour is gathered in first phase original idea from customer

requirement of SDLC. All these things are considered for refinery process.[3] This process varies from company to company depending on the standardization and the methodology that they adopt. It lists all the details of the proposed items in an organized manner. Detailed design of the software in terms of interfaces that will provide the relevant functionalities have to be implemented. The source code based on the required functionality by adhering the coding standards, code optimization etc. It is important to ensure that the functioning of it is perfect, software testing involves a mechanism that is used to report the bugs /undesired results that may occur in SDLC. The bugs that are occurred earlier in the cycle are rectified and ready to released and delivered to customer in last phase of SDLC.

Software Development Life Cycle seven basic steps:

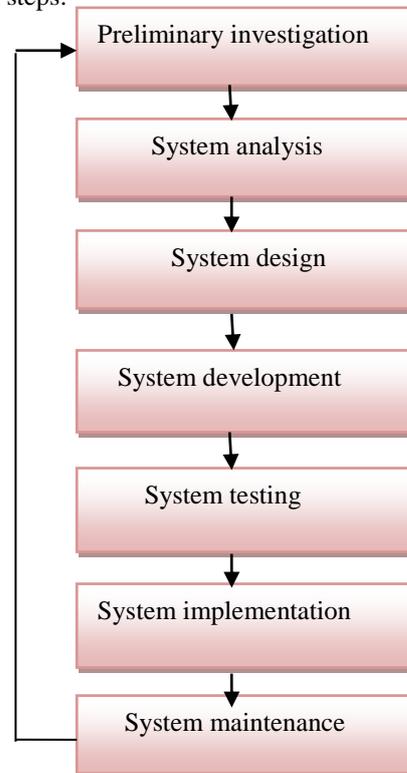


Figure1. Software Development Life Cycle[3]

## 2. Software development models

A software development model is abstract representation of a process.[1] It presents a description of a process from some particular prospective as:

1. Specification.
2. Design.
3. Validation.
4. Evaluation.

### General Software Process Models are:

1. Waterfall model.
2. Prototype model.
3. Rapid application development model.
4. Evaluation development
5. Incremental model
6. Iterative model
7. Spiral model
8. Component based software engineering

## 3. Software Development and Quality management models

There are numbers of general models for software process .but this research will view the Waterfall model, Iterative models, total quality management model.

### 3.1 The Waterfall Model

This model places a lot of emphasis on documentation, Requirements, Specification and Design Document. In waterfall model each phase must be completed sequentially in its entirety before the immediate next phase can begin. If the requirements are of the project are static so this model is suitable. The output of each phase is the input to the immediate next phase in this model, It also is referred to as a classic life cycle model .

The emergence of the Waterfall process to help tackle the growing complexity of development project is logical event. This model encourages the development team to specify what the software is supposed to do before developing the system that lead to final product. It is consist of ETVX Entry-Task-Validation-Exit. This paradigm is a key characteristic of the waterfall process. It enables more accurate tracking of the project progress and early identification of possible slippages.[2] This structural approach is very important for large organizations with large, complex development projects. Many major developers ,especially those who are involved with system development ,adopt this process.

### Requirement Definition:

The requirements analysis includes determining interaction needed with other applications and databases performance

requirements, user interface requirements, and so on

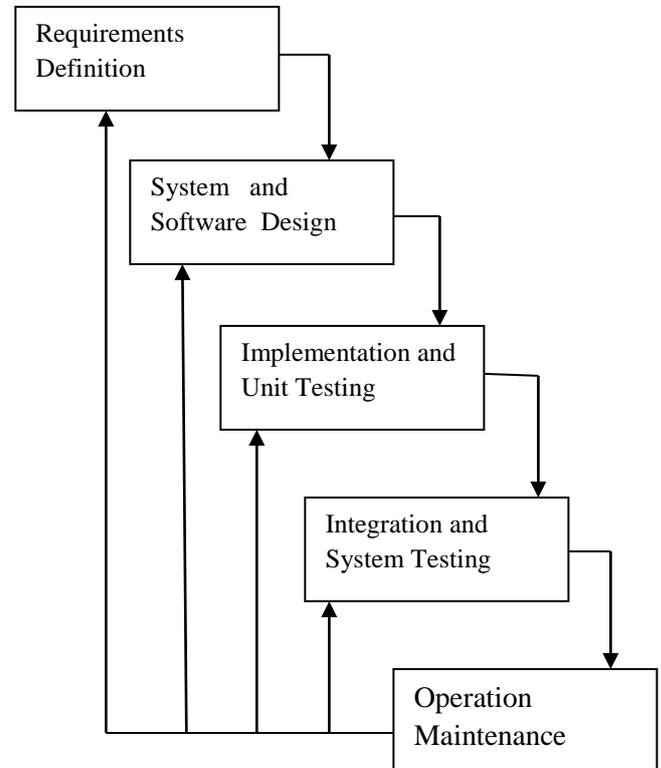


Figure 3.1 Waterfall model[1]

### System and Software Design:

This design defines the major components and interaction of those components but it does not define the structure of each component and also consist of tools used in project can be determined by the designer.

### Implementing and Unit Testing:

The software components defines architecture design stage and produce a specification for how each component is implemented.

### Integrated and System Testing:

Determined whether the software meets the specified requirements and finds any errors present in the code.

### Operation and Maintenance:

Addresses problems and enhancement requests after the software releases.

The change control board in some organizations maintain the quality of the product by reviewing each change made in the in the maintenance cycle model when correcting problem or implementing these enhancement requests.

**Advantages of Waterfall model**

- It is very easy to understand and well suited for small projects.
- Inherent to each phase of this model.
- A review process immediately m.sc after the particular phase is over.
- Identifies deliverables and milestones.
- Reinforces good habits: define –before-design, design-before-code.[1]

**Disadvantages of Waterfall model**

- It is high risk.
- It cannot be guaranteed that one phase of this model is perfect before we move on to the immediate next phase in the model.[3]
- This is not suited for long projects or projects where the requirements can change.
- Difficult to integrate risk management.[1]
- Difficult and expensive to make changes to documents swimming upstream.

**3.2 Iterative model**

Iterative development process model provides a learning experience based on the analysis of each intermediate product, the design and requirements are modified over series of iteration and provides a system to the users that meets evolving customer needs with improved design based on feedback and learning.[2] This process contains eight major steps:

- 1) Domain analysis
- 2) Requirements definition
- 3) Software architecture
- 4) Risk analysis
- 5) Prototype
- 6) test suit and environment development
- 7) Integration with previous iterations.
- 8) Release of iteration.[2]

The iteration model addresses many problems associated with waterfall model. With the iteration model, the project is divided into small parts. This allows the development team to demonstrate the results earlier on in the process and they can obtain valuable feedback from the system users. Each iteration is actually a mini – waterfall process with feedback from one phase. It provides vital information for design of next phase. In a variation of this model, the software products produced at the end of each step can go into production immediately as incremental releases.

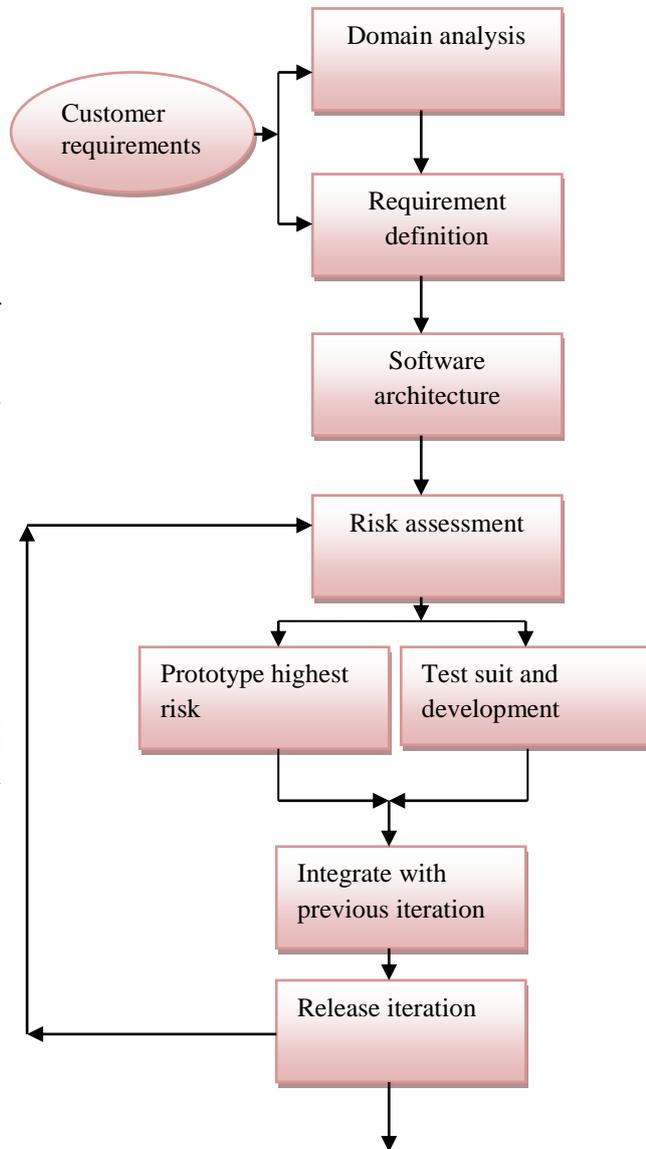


Figure 3.2 Example of the iteration development process model[2]

**Advantages of Iteration model**

- Coding is faster.
- Testing is faster than water fall process.
- Process is increasing in design phase.
- This model facilitates the support for changes within the life cycle.

**Disadvantages of Iteration model**

- More time spent in review and analysis.
- A lot of steps that need to be followed in this model.
- Delay in one phase can have detrimental effects on the software as a whole.

**3.3 Total quality management**

This term represents a style of management aimed to achieving long term success by linking quality and customer satisfaction. The key elements of a TQM system are:

- Customer focus
- Process
- Human side of quality
- Measurement and analysis

This sigma is applied to produce the better quality of software. The good quality product can only produce by focuses on customer requirements, analysis, high risk management.[2]

#### **4. Comparative study of two models of software engineering for best and suitable for software development and quality and risk management.**

The software consists of documents and programs that contain a collection that has been established to be a part of software engineering procedures. The aim of software engineering is to create a suitable work that construct program of high quality. In the study of two models in waterfall model doesn't match reality well. The changes in [1] waterfall process is difficult as compare to iterative model, the iteration process consist of high risk management and analysis. Once the analysis is over, Each requirements is categorized based on their priority [3].The risk management process produces the high quality software product at the end of software development process. Waterfall model costly for small teams and projects. Only the final phase produces a non-documentation deliverable in waterfall model[1].The iterative model provides the facility to change high risk management by changing within the life cycle.

**Conclusion and future work:** This study consist of comparison between two models of software engineering. The aim of this study to examine better and suitable model for risk management, Software development. By the study of water fall model and iterative model the risk is high in waterfall and risk management and analysis in iteration process because it facilitates for changes within the life cycle.[3] Iterations is to provide a system to the users that meets evolving customer needs with improved design based on feedback and learning.[2].In future iteration process can be compared with other software engineering models more time spent in review and analysis.

#### **REFERENCES**

- [1] Nubil Mohammed Ali Munassar and A. Govardhan,"*A comparison between five models of software engineering*" published in September 2010,vol 7,issue 5.IJCSI international journal of computer science.
- [2] Stephen H.Kan,"*Metrics and models in software quality engineering*" Pearson publication Edition 2<sup>nd</sup>.
- [3] Jitendra M. jagtap "*System analysis & design*" Tech-Max publication, pune.