

A Review Paper on Software Development Life Cycle Models

Ms. Sindhu madhuri G

Faculty of Engineering and Technology

Jain (Deemed-to-be University), Ramnagar District, Karnataka - 562112

Email Id: g.sindhumadhuri@jainuniversity.ac.in

Abstract

Software development life cycle (SDLC) representation plays vital role for the growth of the software. As we know the software is low cost, easily distributed and shot with a baton of quality. There are tons of models that matter which project needs which model. To meet the need to automate various activities and to meet certain standard requirements and structural processes introduced into industry for transition easy to bring from manual to automatic system. It advances the concept of system development life cycle (SDLC) model came into existence emphasizes the need to follow something structured approach to new or improved construction systems. In SDLC, five most important development models are: Iterative Model, prototype model, linear Sequential Model, V-shaped model, spiral model. All of these models have pronouns as well as pronouns. The major aim of this review paper is to show different representations of development of the software and do relative studies to represent them characteristics and disadvantage of every model of the software. In the number model, here are some important models like waterfall, incremental, spiral, V-shape, rad (rapid application development), agile model. In this paper, want to highlight different models. The aim of this review paper is to give details about different lot of models.

Keywords: *Agile model, Incremental model, Linear sequential model, Prototype model, Rapid application development, Software development model (SDLC), Spiral model, V-shaped model.*

I. INTRODUCTION

Nowadays everyone knows how important computers are these days for our daily lives. Now, computers are used in every area like industry, hospital, medicine, education despite agriculture in the military sector. Computer used for developing countries. Work which is very difficult and time consuming for humans can do that work by computer within a few microseconds[1]. Many companies make software program objective facilities to Offices. Some problems during early software development have happened[2]. So, to avoid these problems or follow some structured flow of constraints software engineering and software development program[3]. We

are breathing in an earth where computers must be used. 75% of a person's life is dependent on the computer. So to deal with the daily increasing needs of the digital world, use computers or laptop or pc. So, for using computers here first of all, development of the software is must require for gain better solution. When computers will be used, then essential to work on the software. Works on everything in the computer the principle of software on which it works. For the development of these software there are a model is required to develop this important software.

A. Software Development Life Cycle (SDLC):

It comprises a complete plan that classify how to maintain, grow, transform and make or boost particular software[3]. The life sequence explain a strategy for improving software standard and the overall process of the development[4]. Software evolution Life Cycle in Information Systems, System engineering and Software engineering is the procedure of creating or converting systems, and models and strategy which is used by the people for the development of these systems [5]. Concept generally refers to a computer or information system. SDLC concept in software engineering underlines the many types of software development methods of. It is an arranged planning and control framework creating an information system software development Process.

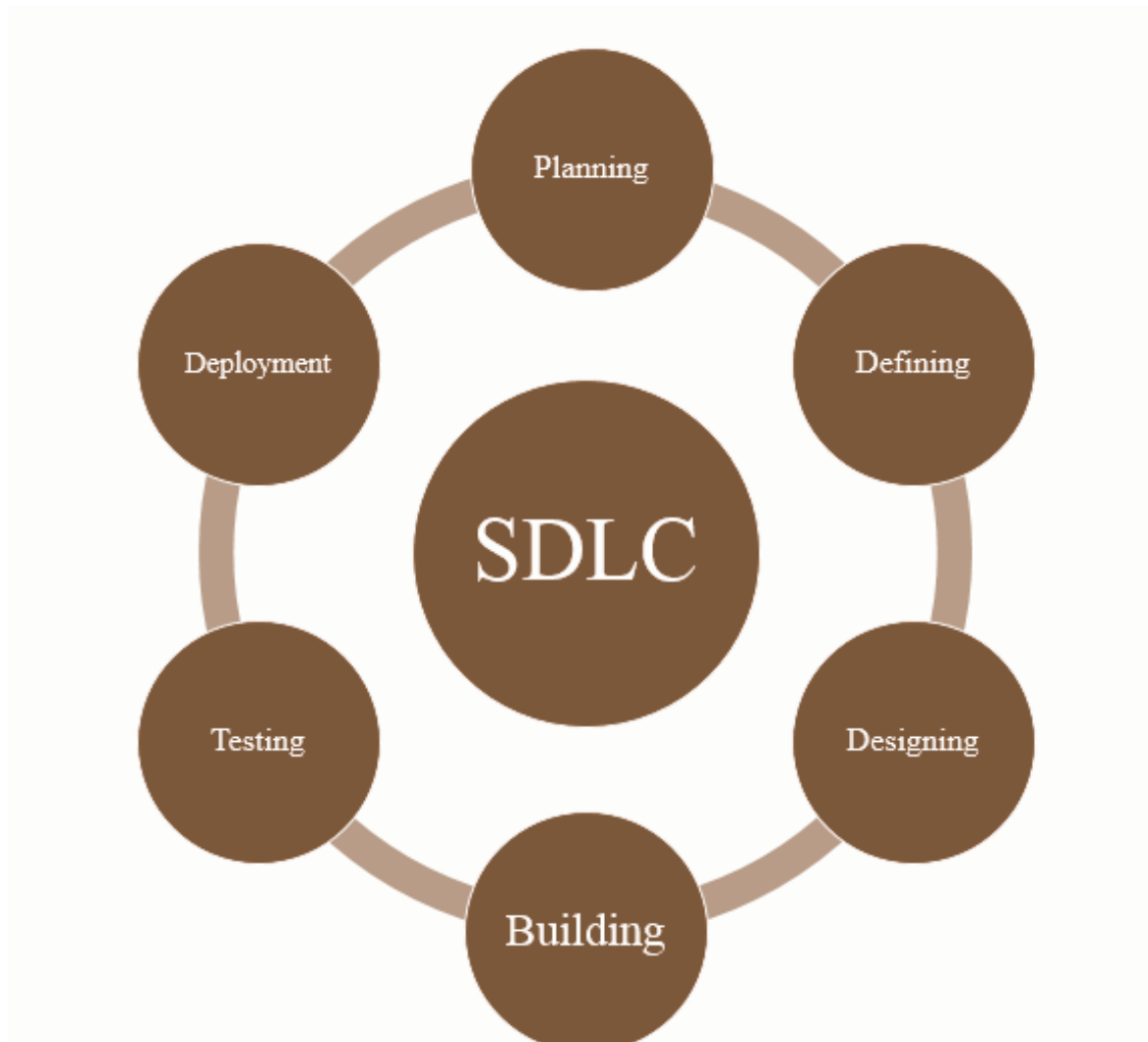


Fig. 1 Phases of Software Development Life Cycle (SDLC)

Understand the project. Do project planning. Understand the requirement. Design according to that coding planned solution. Test the actual program. Every action can be very complicated and essential procedures for execute it effectively and accurately in large software systems. Similarly, these types of activity can be so big that it cannot be controlled and should be broken in a single period and in minor modulus. For example, blueprint of an extensive software system every time divided into several.

The design phase begins at a very high level specifying only components in design system logic where for a detailed design components are specified. Basic activities or take software development steps the systems are; Design Determination of System Requirements Design software development (coding) system testing.

These are models that help develop the desired software. This is followed and rendering is provided. Software life cycle. It comprises all the activities essential to advance a software item. In different words, it plans different activities execute on a software item.

B. Software Development Life Cycle (SDLC) Models:

A software life sequence representation is an illustrative representation development cycle of the software. The SDLC model may have a different approach but the basic phase and activity remain the same for all models[6][7].

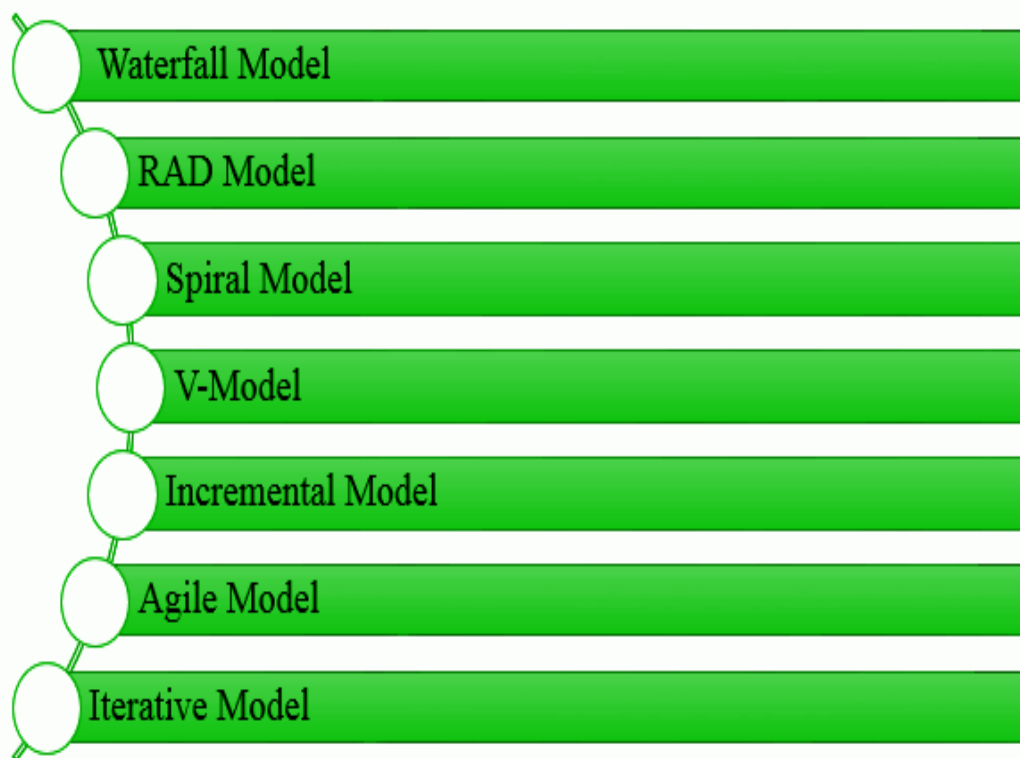


Fig. 2 Models of Software Development Life Cycle (SDLC)

1. Waterfall Model:

This is the kind of representation in which every step is execute one by one. This come after by a series. The output of the initial phase goes to the next phase and these outputs reach to the next phase and so on.

Also each step must be completed to proceed to the next stage. And this process will be repeated until all the phases are not finished, and this is finished after the maintenance phase[8].

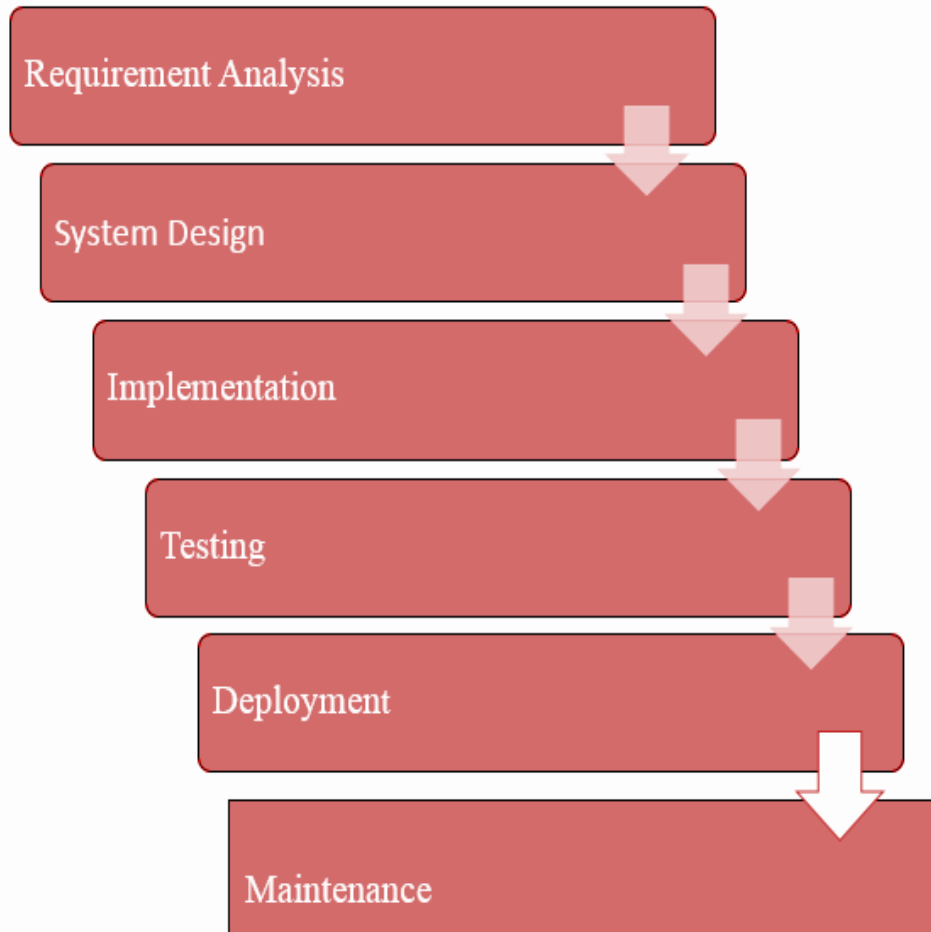


Fig. 3 Waterfall Model

The waterfall model is the first model used in SDLC[9]. It is also recognized as linear sequential model. In this representation, the output of the one phase is the input for the next other phase. The evolution of the next stage starts only when the before the next phase is finalized.

2. RAD Model:

It focuses more on output which is better in quality and in a very short time. Elements in this are generated parallel and distributed rapidly. It allows the customer to use it to focus more on output which is better in quality and in a very short time. Elements in this are generated parallel and distributed rapidly[10].

It allows the customer to use some and ask for changes if necessary. This team is a fully functional short time model for customers. This development (RAD) model describes a process of development of the software that highlight iterative delivery and rapid prototype[11].

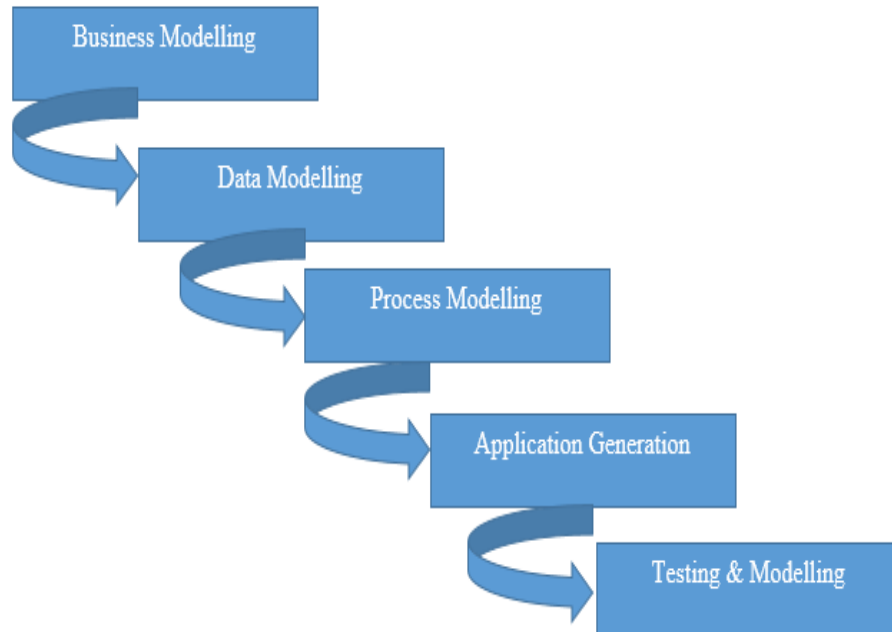


Fig. 4 Phases of RAD Model

Therefore, the RAD model is a faster alternative to the specific waterfall development model, often focusing on large-scale planning and sequential design practices.

3. Spiral Model:

The spiral model, originally proposed by Bohm, is an Evolutionary Software Process model prototype of couples with natural systematic aspects of controlled and linear sequential models[12].

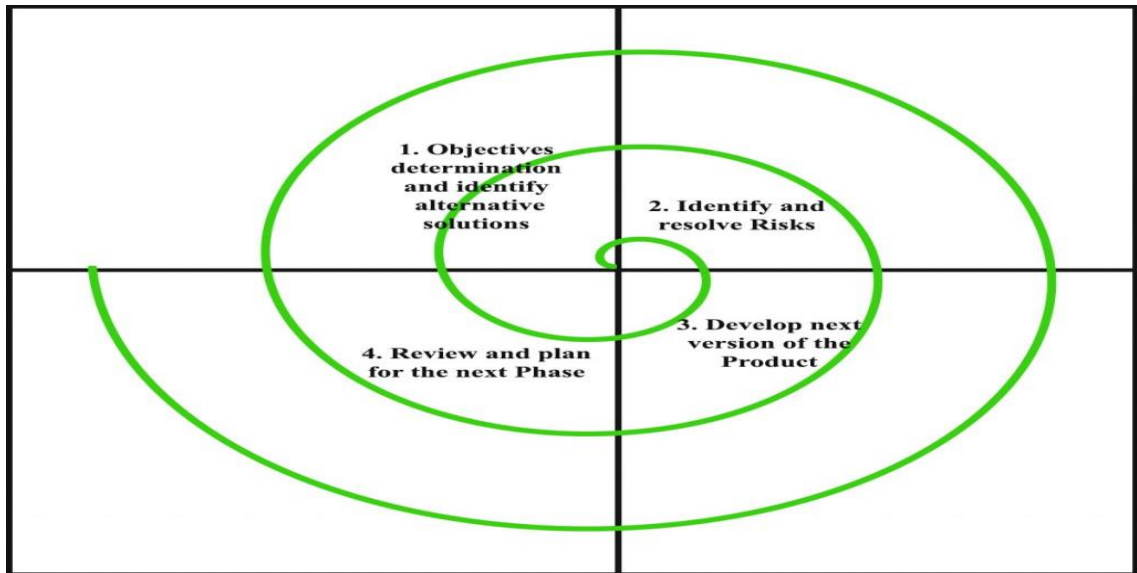


Fig. 5 Spiral Model in SDLC

4. V-Model:

This is called validation and verification model.

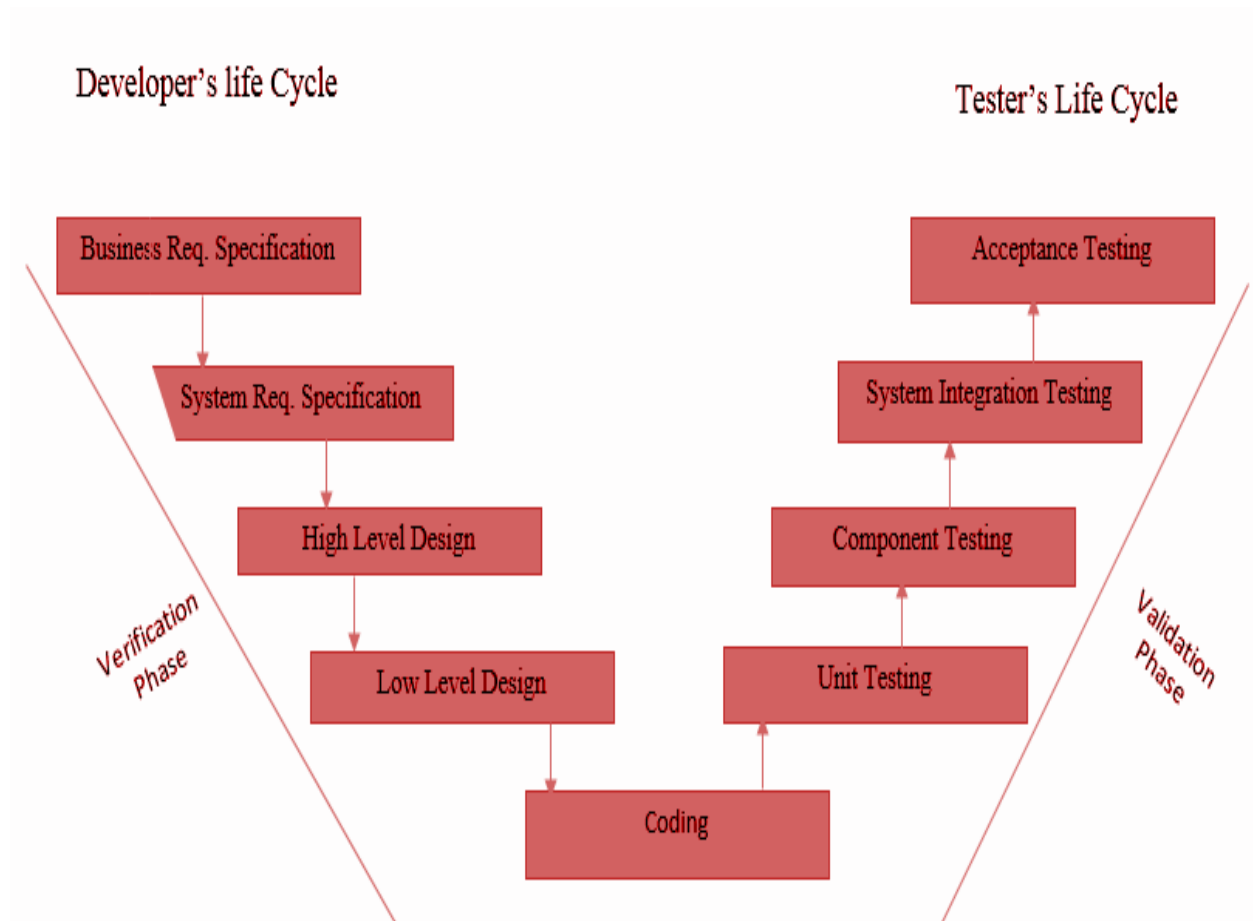


Fig. 5 V-shaped Model in SDLC

The earliest models are also Akin the sequence is as follows and the next phase cannot be started before the phase is completed. This is characterized by the fact that the test is carried out simultaneously with the stage of development, that is, the work done earlier and later verified.

V-model means validation and validation model. V-shaped life, sequential route of execution of processes in a waterfall model[14]. Every part should be finished before the next phase starts. Product testing plan parallel with the same phase of development.

5. Incremental Model:

This model combines components of the linear sequential model with repetitions philosophy of Prototype. The first increment is a major product. This means that the introductory needs are known, but the number of ancillary facilities is not small.

Gist products are used by customers. After this result, the next plan for the next phase is established.

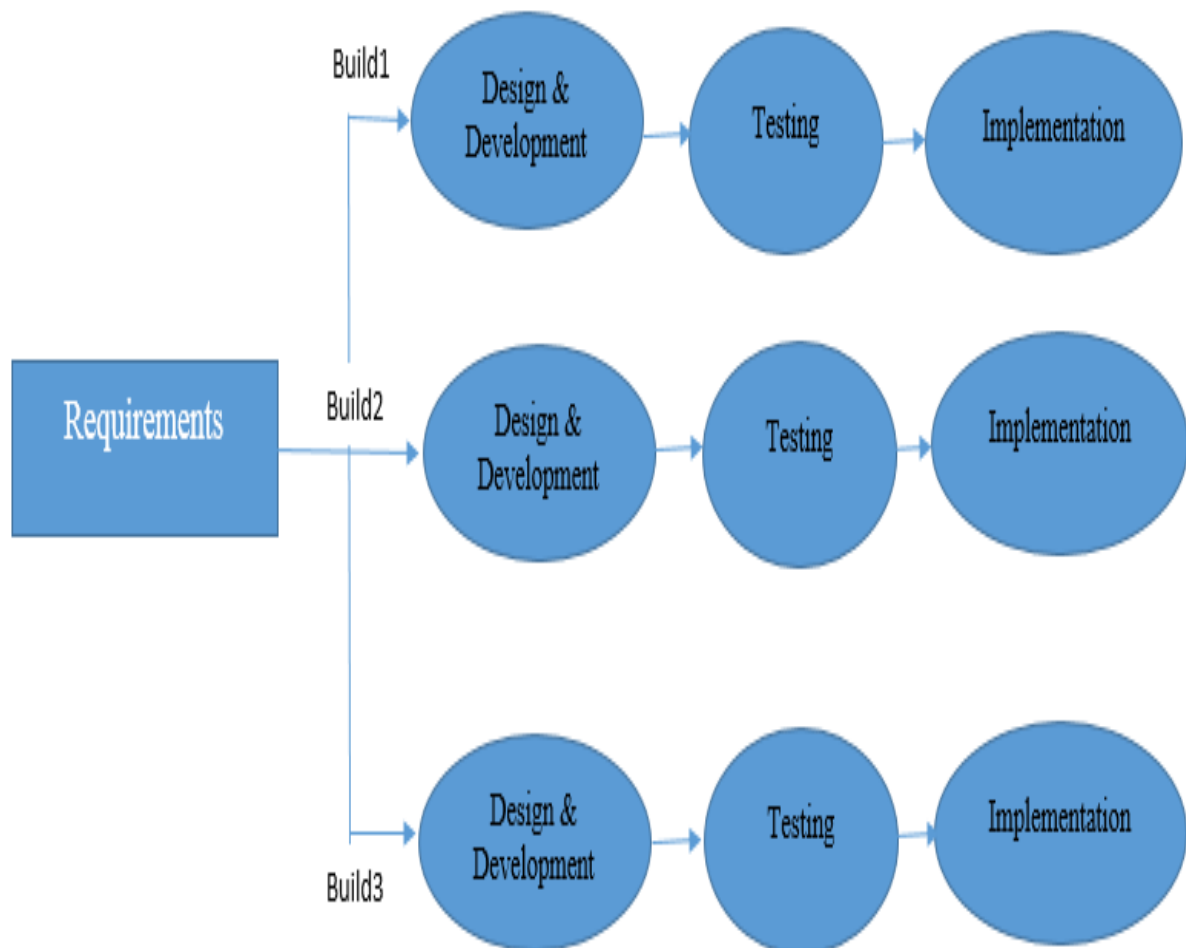


Fig. 6 Incremental Model in SDLC

6. Agile Model:

This model was established by the agile team in the year 2001. Its key objective is advance and continuous delivery of the software to reach customer satisfaction[15].

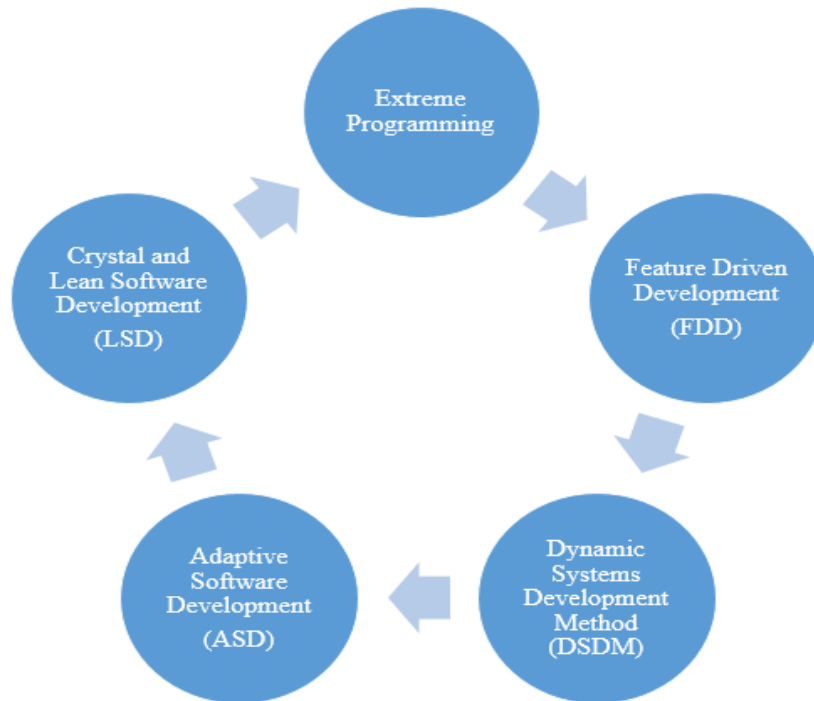


Fig. 7 Agile Model of Software Development Life Cycle

The main drawbacks of this are:

- Incremental - short software is produced, which includes rapid development cycles.
- Cooperative - more client-developer interactions
- Adaptive - Adequate alerts for immediate change.

7. Iterative Model:

In this model, For the purpose of the development of the first version of the software can be starts with some identifications.

If the software requires to be transformed after the first version, a new one version of the software is created with a new iteration. Each release of the iteration model takes an exact and definite time called iteration.

The iterated model allows the first steps to be reached with a different version. The last construction of the project was improved at the last of the software development life cycle (SDLC) process.

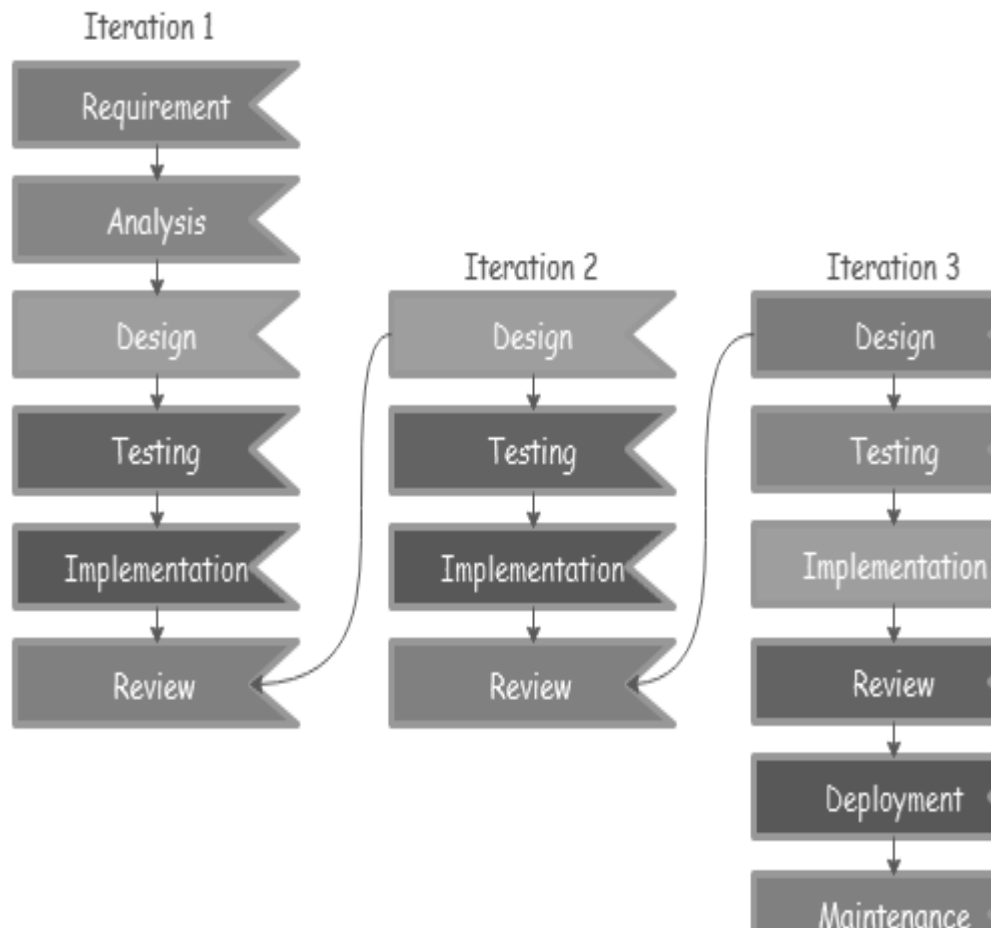


Fig. 8 Iterative Model

II. CONCLUSION

Some strengths and weaknesses having every software. Models are acquiring on the source of essential. Nowadays we are seeing that technology is developing at a very fast pace. Keep in mind that custom developers have to choose a particular model to create the desired software. Choose the accurate one the development model can conduct to quality product and quick delivery. After studying all the models through various factors, Incremental where models are used for new technology the necessary resources are not available.

To develop an internet (online) system for the model web interface, transaction processing i.e. where extremely high volume of movement with end users, for this work here prototype model is used. Since important work is done in a lower capital cost model. For excessive risk projects, long term projects are used in spiral model and where remarkable changes are anticipated to be the lowest cost.

V-shaped models are used for smaller sizes projects where the requirements are clearly defined and definite and when there are all kinds of technical resources available. One model outperforms loss of other models.

III. REFERENCES

- [1] D. G. M. Schouten, I. Pfab, A. H. M. Cremers, B. van Dijk, and M. A. Neerincx, Computers Helping People with Special Needs. 2014.
- [2] S. K. D. and P. Dubey, "SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC) ANALYTICAL COMPARISON AND SURVEY ON TRADITIONAL AND AGILE METHODOLOGY," Natl. Mon. Ref. J. Res. Sci. Technol., 2013.
- [3] Tutorial.com, "Software Development Life Cycle (SDLC)," Softw. Dev. Life Cycle, 2014.
- [4] S. S, "A Study of Software Development Life Cycle Process Models," SSRN Electron. J., 2017, doi: 10.2139/ssrn.2988291.
- [5] V. Rastogi, "Software Development Life Cycle Models- Comparison , Consequences," Int. J. Comput. Sci. Inf. Technol., 2015.
- [6] N. B. Ruparelia, "Software development lifecycle models," ACM SIGSOFT Softw. Eng. Notes, 2010, doi: 10.1145/1764810.1764814.
- [7] V. Osetskyi, "SDLC Models Explained: Agile, Waterfall, V-Shaped, Iterative, Spiral," Existek, 2017. .
- [8] L. Sherrell, "Waterfall Model," in Encyclopedia of Sciences and Religions, 2013.
- [9] M. Bulman, "SDLC - Waterfall Model," Indep., 2017.
- [10] Model RAD, "SDLC - RAD Model," Tutorialspoint, 2018.
- [11] Tutorialspoint, "SDLC Software Prototype Model," Tutorialspoint, 2017.
- [12] B. Boehm, "A Spiral model of software development and enhancement," in Software Management, Seventh Edition, 2007.
- [13] K. Dowding and B. C. Greene, "Spiral Model," in Encyclopedia of Power, 2012.
- [14] Tutorialspoint, "SDLC - V-Model," Tutorialspoint, 2016.
- [15] Tutorials Point (I) Pvt. Ltd., "Sdlc - Agile Model," SDLC - Agil. Model, 2015.