SOFTWARE ENGINEERING MODELS
CONSEQUENCES AND ALTERNATIVES

Nitin Mishra¹, Shantanu Chowdhary², Arunendra Singh³, Anil Sharma⁴

¹Associate professor, School of Engineering, Sangam University, Bhilwara
²M.tech Student, School of Engineering, Sangam University, Bhilwara
³Asst. professor, PSIT Kanpur
⁴M.tech Student, ITM Bhilwara

ABSTRACT
There are lots of models which are used in software development Life cycle. This research deals with a vital and important issue in computer world. It is concerned with the software management processes that examine the area of software development through the Development models, which are known as software development life cycle. It represents five of the development models namely, waterfall, Iteration, spiral, win-win spiral. These models have advantages and disadvantages as well. Therefore, the main objective of this research is to represent different models of software development and make a comparison between two of them waterfall model and spiral model. To show the features and defects of each model. And proposed a Different Alternative model.

Keywords: Software Management Processes, Software Development, Development Models, Software Development Life Cycle, Comparison between five models of Software Engineering, Software engineering Models Consequences and Alternatives.

1. INTRODUCTION
Now a day’s computer is everywhere. In fact, computer has become indispensible in today's life as it is used in many fields of life such as industry, medicine, commerce, education and even agriculture.

Now days, organizations become more Dependent on computer in their works as a result of computer technology. [1]Computer is considered a time-saving device and its progress helps in executing complex, long, repeated processes in a very short time with a high speed. In addition to using computer for work, people use it for fun and entertainment. Noticeably, the number of companies that produce software programs for the purpose of facilitating works of offices, administrations, banks, etc., has increased recently which results in the difficulty of enumerating such companies. During the previous four decades, software has been developed from a tool used for analyzing information or solving a problem to a product in itself. However, the early programming stages have created a number of problems turning software an obstacle to software development particularly those relying on computers. Software consists of documents and programs that contain a collection that has been established to be a part of software engineering procedures. Moreover, the aim of software engineering is to create a suitable work that constructs programs of high quality.

![Diagram](image-url) Fig. 1 Explanation of software engineering conception.
2. SOFTWARE DEVELOPMENT MODELS

There are five models present to develop a software system but our emphasis on two among them [5]:
- Waterfall Model.
- Spiral Model.

Now we discuss it one by one

**WATERFALL MODEL**
It is called waterfall because of the way it falls down [3]. It is a linear sequential model because if one stage is not complete another cannot be started. Stages are defined below.

- **Decide**: In this stage we build the Business case, User Requirement and System Specification.
- **Design**: In this stage we make an Overall structure and Detail structure of the software.
- **Develop**: In this stage we implement the designing phase by building the each and every component.
- **Demonstrate**: In this stage we apply the test on the builded product generally white box and black box testing is applied in it.

![Waterfall Model Diagram](image)

**Fig 2 Water Fall Model**

**Strengths**
- Minimizes planning overhead since it can be done up front.
- Structure minimizes wasted effort, so it works well for technically weak or inexperienced staff.

**Weaknesses**
- Inflexible
- Only the final phase produces a non-documentation deliverable.
- Backing up to address mistakes is difficult.

**SPIRAL MODEL**:
This model was not the first model [4] to discuss iterative development, but it was the first model to explain why the iteration matters. As originally envisioned, the iterations were typically 6 months to 2 years long. Each phase starts with a design goal and ends with the client (who may be internal) reviewing the progress thus far. Analysis and engineering efforts are applied at each phase of the project, with an eye toward the end goal of the project.

The process begins at the center position. From there it moves clockwise in traversals. Each traversal of the spiral usually results in a deliverable [6]. It is not clearly defined what this deliverable is. This changes from traversal to traversal. For example, the first traversals may result in a requirement specification. The second will result in a prototype, and the next one will result in another prototype or sample of a product, until the last traversal leads to a...
product which is suitable to be sold. Consequently the related activities and their documentation will also mature towards the outer traversals. E.g. a formal design and testing session would be placed into the last traversal.

![Spiral model]

**CONSEQUENCES OF WATERFALL & SPIRAL MODEL**

<table>
<thead>
<tr>
<th>STRENGTHS</th>
<th>WEAKNESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>High amount of risk analysis.</td>
<td>Can be a costly model to use.</td>
</tr>
<tr>
<td>Good for large and mission-critical projects.</td>
<td>Risk analysis requires highly specific expertise.</td>
</tr>
<tr>
<td>Software is produced early in the software life cycle.</td>
<td>Project’s success is highly dependent on the risk analysis phase.</td>
</tr>
</tbody>
</table>

**ALTERNATIVE OR PROPOSED MODEL**

Above two models are lac in certain functionality so for this reasons I proposed a software model which is very much suitable for the development of the software’s and gives a suitable frame work to which assure the customer satisfaction and also the best realization for vendor of the software. The proposed model has both the properties of waterfall and the spiral model.

Stages of Proposed model:
1) Requirement Analysis
2) Design generation
3) Prototype Generation
4) Implementation
5) Testing
6) Handover product
7) Customer Block

Now we discuss all stages one by one

1) **Requirement Analysis:**
The next stage is to define a set of user requirements. These define for the preferred solution strategy what the software system needs to achieve in order to meet the business opportunity. Areas which need to be included are Function Requirement and Non-Functional Requirement.

2) Design Generation:
In this stage the data Flow diagrams are built to find out the way or road map to develop the software.

After this bloke the developer move to the **customer Block** in customer Block there are three stages
a) Customer Evolution
b) Alteration
c) Approved

3) Prototype Generation:
In this stage the working model of each module is generated and then the result of this stage is fed to customer Block again for customer Approval.

4) Implementation: In this stage the developer implement the prototype model by using Back Hand and front Hand Languages.

5) Testing:
Now after Product is developed the final testing is done by the developer by using Testing Techniques ex white box or black box or any other technique.

6) Handover Product:
Finally the ready product is hand over or installed in the customer specified place.

7) Customer Block:
The customer Block is the main attraction of this model because it decides the whole project cost for the customer as well as the developer [6]. In this stage the developer discussed the developed Design & developed Prototype with the customer if customer proposed any changes to it, developer changes the project according to customer or negotiates it with possible changes. Otherwise if customer approved the project they will proceed to next stage.

**ADVANTAGES OF PROPOSED MODEL**

a) Fewer complexes than any other model.
b) Cost is controlled by developer in each phase.
c) Customer satisfaction is assured.
d) Stages are easy to implement and completely practical for any product development.
e) Applicable on both small and big projects.
f) Need less maintenance after delivery because it developed according to customer.

Prototype model gives better idea to the customer and developer how the final product looks like.
3. CONCLUSION AND FUTURE WORK

After completing this research, it is concluded that:
1. There are many existing models for developing Systems for different sizes of projects and Requirements.
2. These models were established between 1970 and 1999.
3. Waterfall model and spiral model are used commonly in developing systems.
4. Each model has advantages and disadvantages for the development of systems, so each model tries to eliminate the disadvantages of the previous model.

Finally, some topics can be suggested for future works:
1. Suggesting a model to simulate advantages that are found in different models to software process Management.
2. Making a comparison between the suggested model and the previous software processes management models.
3. Applying the suggested model to many projects to ensure of its suitability and documentation to explain its mechanical work.

REFERENCES

[6] www.ccs.neu.edu/home/matthias/670-s05/Lectures/2.html

AUTHORS

Nitin Mishra received B.Tech ,MTech NIT Allahabad and did his PHD under guidance of Dr. Anurag Mishra of Virgennia Tech University USA. He is currently working as Associate Professor at Sangan University Bhilwara.

Shantanu Chowdhary Received B.E. Degree from Rajasthan University Rajasthan, After B.E. He also done MBA from Rajasthan Technical university Rajasthan. Currently Pursuing M.Tech from Sangan University Bhilwara Rajasthan.