



## SOFTWARE DEVELOPMENT LIFE CYCLE MODELS

Danda Swathi

M.Tech Computer Science and Engineering

**ABSTRACT** This paper 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 six of the software development models namely, waterfall, Prototyping, Incremental, RAD, Spiral and Extreme programming. This paper provides a framework which serve as a basis for analyzing the similarities and differences among different life cycle models. Therefore, the main objective of this research is to represent different models of software development and make a comparison between them to show the features and defects of each model.

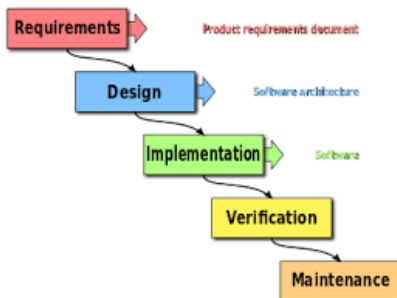
**KEYWORDS :** Software Project, Software Process Models, Software Development Life Cycle, SDLC phases, Comparison between five models of Software Engineering.

**Introduction:**

No one can deny the importance of computer in our life, especially during the present time. In fact, computer has become indispensable in today's life as it is used in many fields of life such as industry, medicine, commerce, education and even agriculture. It has become an important element in the industry and technology of advanced as well as developing countries. Now a days, organizations become more dependent on computer in their works as a result of computer technology. 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.

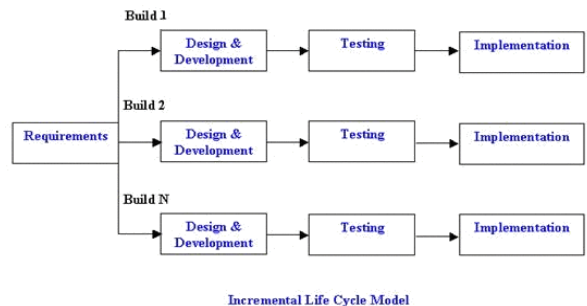
**Waterfall Model:**

The waterfall model was introduced by Royce in (1970), specifically in the context of spacecraft mission software design, and is one of the most popular methods of assessing the evolution of a product or system. Essentially, it is a step-by-step sequential description of the product's life cycle that spans 7 different stages, originally denominated "system requirements, software requirements, analysis, program design, coding, testing and operations" (Royce 1970).

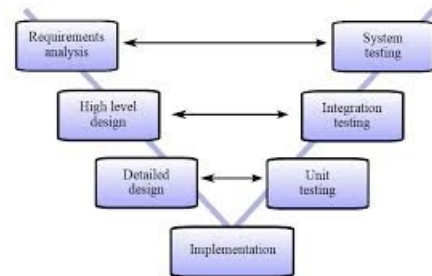
**Waterfall Life Cycle Model****Incremental Model:**

The incremental model is a particular evolution of the waterfall model that attempts to address its more prominent shortcoming, which is the slowness of the cycle. It also aims at outlining a more flexible process that requires less extensive planning up-front. According to this approach, instead of dividing the SDLC into static, isolated steps, the whole process can instead be designed, tested, and implemented one

fraction at a time, in successive stages, so that with each stage (or increment), there can be at least some feedback from the client. This feedback will provide valuable assistance in the next increment of the process and so forth. With each ongoing increment, the product is extensively tested and improved, according to objectives and expectations from the client, which facilitates its eventual success

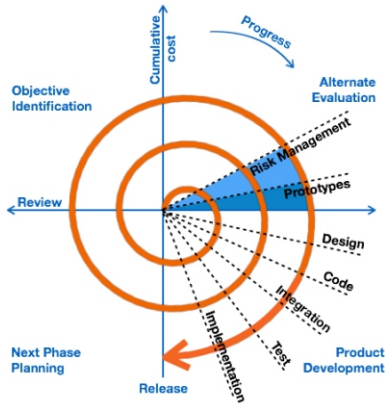
**V-shaped Model:**

V-Shaped life cycle is a sequential path of execution of processes. Each phase must be completed before the next phase begins. The testing procedures are developed early in the life cycle before any coding is done, during each of the phases preceding implementation. Requirements begin the life cycle model just like the waterfall model. Before development is started, a system test plan is created. The test plan focuses on meeting the functionality specified in the requirements gathering. The high-level design phase focuses on system architecture and design. An integration test plan is created in this phase as well in order to test the pieces of the software systems ability to work together. The low-level design phase is where the actual software components are designed, and unit tests are created in this phase as well. The implementation phase is, again, where all coding takes place. Once coding is complete, the path of execution continues up the right side of the V where the test plans developed earlier are now put to use.

**V-Shaped Life Cycle Model****Spiral Model:**

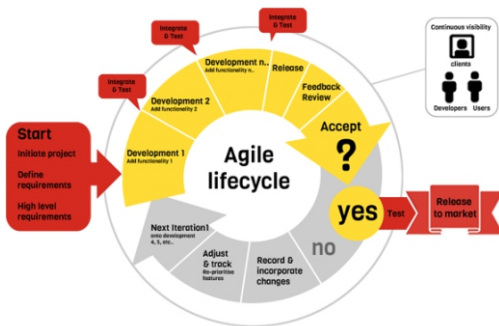
The spiral model dates back to the end of the 1980s, when it was outlined by Barry Boehm, and introduces something that other models did not take into account, which is risk analysis. In essence, the spiral model attempts to bring together key aspects of some other prominent

models (namely the waterfall, incremental, and evolutionary prototyping), in an attempt to gather the most appropriate traits from each one, because specific projects might be more or less adaptable to specific models



**Agile model:**

Agile software development is a style of software development that emphasizes customer satisfaction through continuous delivery of functional software”.



The code is tested more frequently. Sometime a dedicated “Continuous Integration” Server/Software may be used to ease the integration testing of the code. 8. Depending on the feedback received, the code is refactor. Refactoring does not impact the external behavior of the application but the internal structure may be changed to provide better design, maintainability. Some ways of refactoring may be add interface, use super class, move the class etc.

**Conclusion:**

There are many existing models for developing systems for different sizes of projects and requirements and these models were established between 1970 and 1999. Waterfall model and spiral model are used commonly in developing systems. Each model has advantages and disadvantages for the development of systems, so each model tries to eliminate the disadvantages of the previous model.

**References:**

1. Ian Sommerville, "Software Engineering", Addison Wesley, 7th edition, 2004.
2. CTG. MFA – 003, "A Survey of System Development Process Models", Models for Action Project: Developing Practical Approaches to Electronic Records Management and Preservation, Center for Technology in Government University at Albany / Suny, 1998.
3. Steve Easterbrook, "Software Lifecycles", University of Toronto Department of Computer Science, 2001. National Instruments Corporation, "Lifecycle Models", 2006
4. JJ Kuhl, "Project Lifecycle Models: How They Differ and When to Use Them" 2002 , www.businessolutions.com.
5. Karlm, "Software Lifecycle Models", KTH, 2006.
6. Rlewallen, "Software Development Life Cycle Models", 2005 ,http://codebeter.com.
7. Barry Boehm, "Spiral Development: Experience, Principles, and Refinements", edited by Wilfred J. Hansen, .