

UNIT - I

LECTURE 1

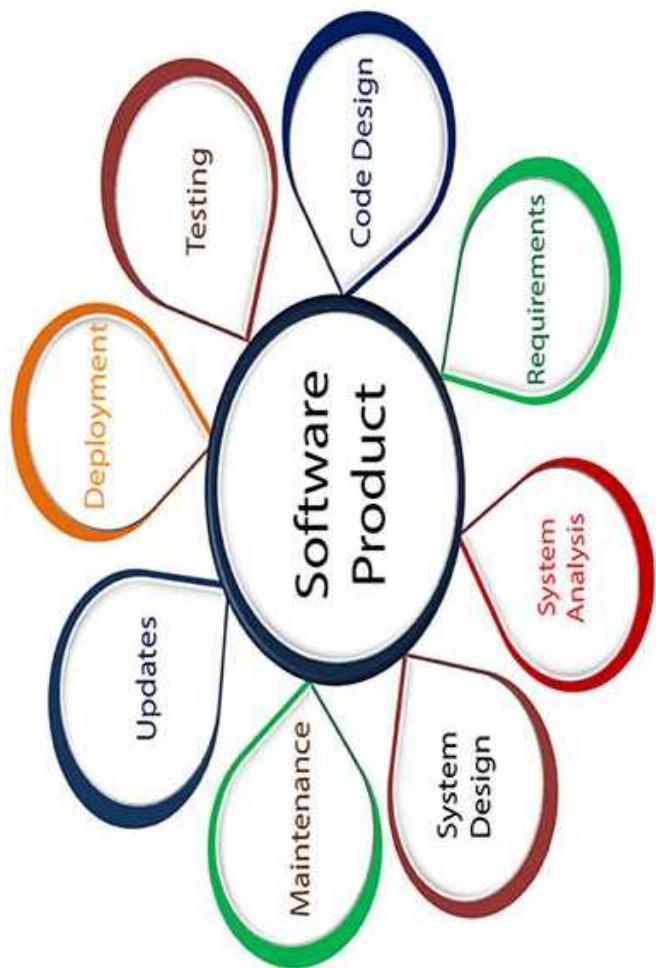


INTRODUCTION



- Software Testing is the process of identifying the correctness and quality of software program. The purpose is to check whether the software satisfies the specific requirements, needs and expectations of the customer.
- In other words, testing is executing a system or application in order to find software bugs, defects or errors.

SOFTWARE ENGINEERING

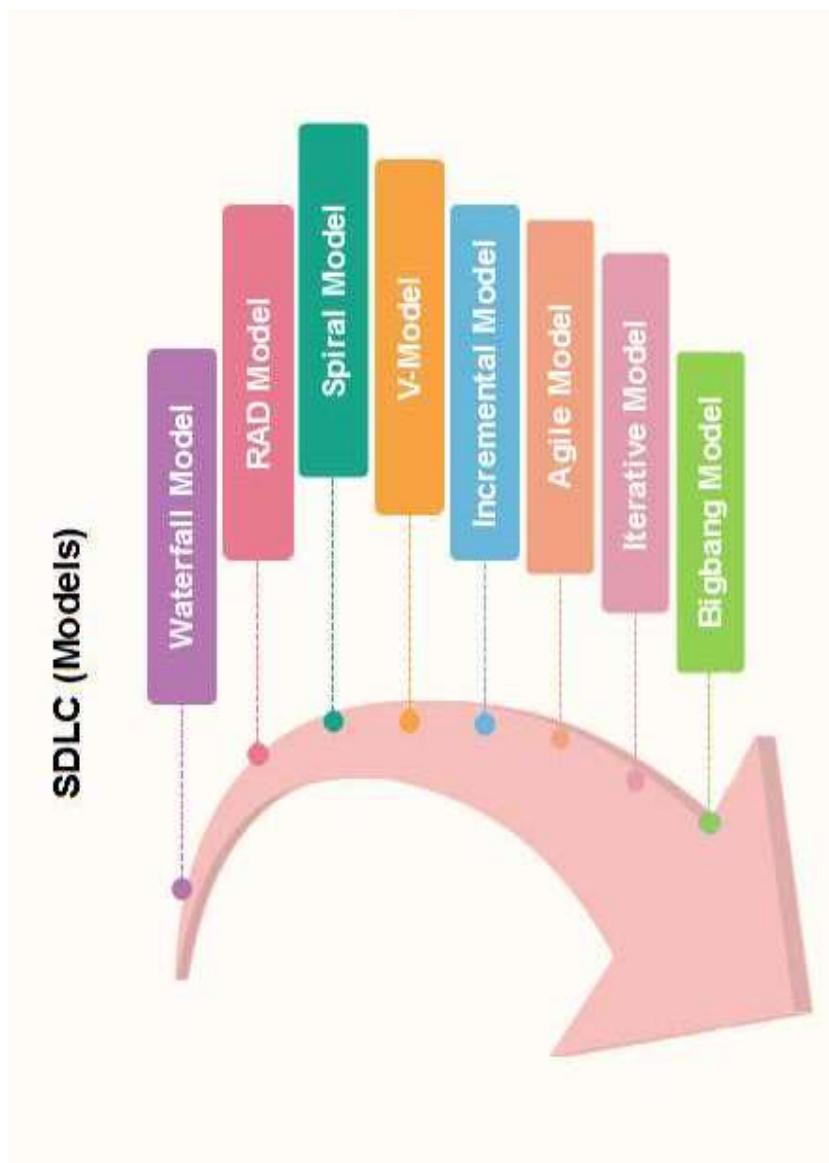


- The application of a systematic, disciplined, quantifiable approach to the development, operation and maintenance of software; that is, the application of engineering to software.
- The study of approaches as in the above statement.
- Software engineering is the establishment and use of sound engineering principles in order to obtain economically software that is reliable and work efficiently on real machines.



SOFTWARE DEVELOPMENT LIFE CYCLE

- Software Development Life Cycle is a systematic approach to develop software. It creates a structure for the developer to design, create and deliver high quality software according to the requirements of customer or end user.



WATERFALL MODEL

- The whole process of software development is divided into separate phases. The outcome of one phase acts as the input for the next phase sequentially. This means that any phase in the development process begins only if the previous phase is complete.



DIFFERENT PHASES OF THE WATERFALL MODEL

- **Requirements:**

- The first phase involves understanding what needs to design and what is its function, purpose, etc. Here, the specifications of the input and output or the final product are studied and marked.

- **System Design:**

- The requirement specifications from the first phase are studied in this phase and system design is prepared. System Design helps in specifying hardware and system requirements and also helps in defining overall system architecture. The software code to be written in the next stage is created now.

- **Implementation:**

- With inputs from system design, the system is first developed in small programs called units, which are integrated into the next phase. Each unit is developed and tested for its functionality which is referred to as Unit Testing.



- **Integration and Testing:**

- All the units developed in the implementation phase are integrated into a system after testing of each unit. The software designed, needs to go through constant Software testing to find out if there are any flaws and errors. Testing is done so that the client does not face any problem during the installation of the software.

- **Deployment of System:**

- Once the functional and non functional testing is done, the product is deployed in the customer environment or released into the market.

- **Maintenance:**

- This step occurs after installation, and involves making modifications to the system or an individual component to alter attributes or improve performance. These modifications arise either due to change requests initiated by the customer, or defects uncovered during live use of the system. The client is provided with regular maintenance and support for the developed software.



ADVANTAGES OF WATERFALL MODEL

- This model is simple and easy to understand and use.
- It is easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.
- In this model phases are processed and completed one at a time. Phases do not overlap.
- Waterfall model works well for smaller projects where requirements are clearly defined and very well understood.

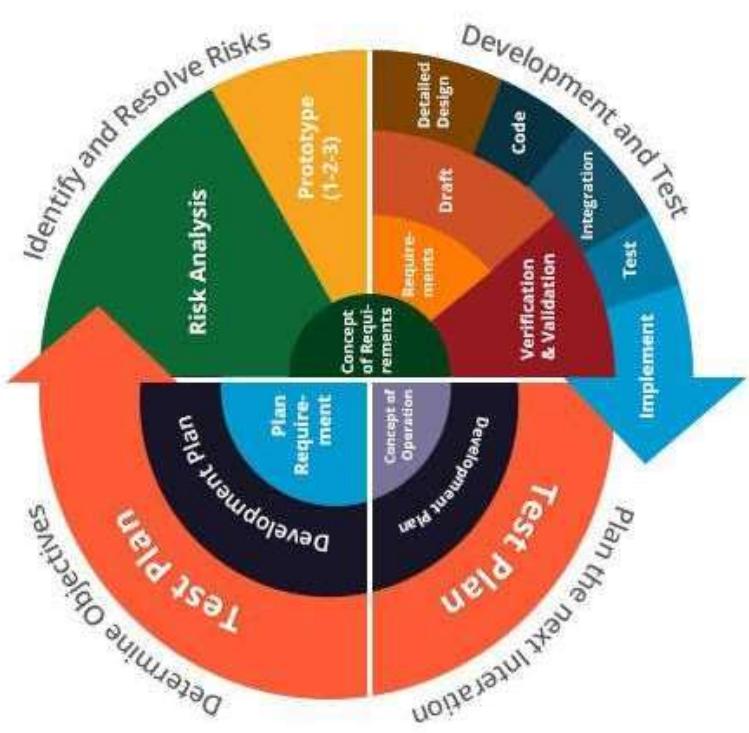
DISADVANTAGES OF WATERFALL MODEL

- Once an application is in the testing stage, it is very difficult to go back and change something that was not well-thought out in the concept stage.
- No working software is produced until late during the life cycle.
- High amounts of risk and uncertainty.
- Not a good model for complex and object-oriented projects.
- Poor model for long and ongoing projects.
- Not suitable for the projects where requirements are at a moderate to high risk of changing.



SPIRAL MODEL

- Spiral Model is a combination of Iterative Development Model and Waterfall Model with very high emphasis on risk analysis. It allows for incremental releases of the product, or incremental refinement through each iteration around the spiral.



DIFFERENT PHASES OF SPIRAL MODEL

- **Planning:**

- Requirements are studied and gathered in this phase. It includes estimating the cost, schedule and resources for the spirals. As the product matures, identification of system requirements and unit requirements are all done in this phase.

- **Risk Analysis:**

- Risk Analysis includes identifying, estimating, and monitoring technical feasibility and management risks, such as schedule slippage and cost overrun. Once the risks are identified, risk mitigation strategy is planned and finalized.



- **Engineering:**

- This phase refers to production of the actual software product at every spiral. Actual development and testing of the software takes place in this phase. It includes testing, coding and deploying software at the customer site. A POC (Proof of Concept) is developed in this phase to get customer feedback.

- **Evaluation:**

- After testing the build, at the first phase, the customer evaluates the product and provides feedback.



ADVANTAGES OF SPIRAL MODEL

- High amount of risk analysis hence, avoidance of Risk is enhanced.
- Good for large and mission-critical projects.
- Strong approval and documentation control.
- Additional Functionality can be added at a later date.
- Software is produced early in the software life cycle.

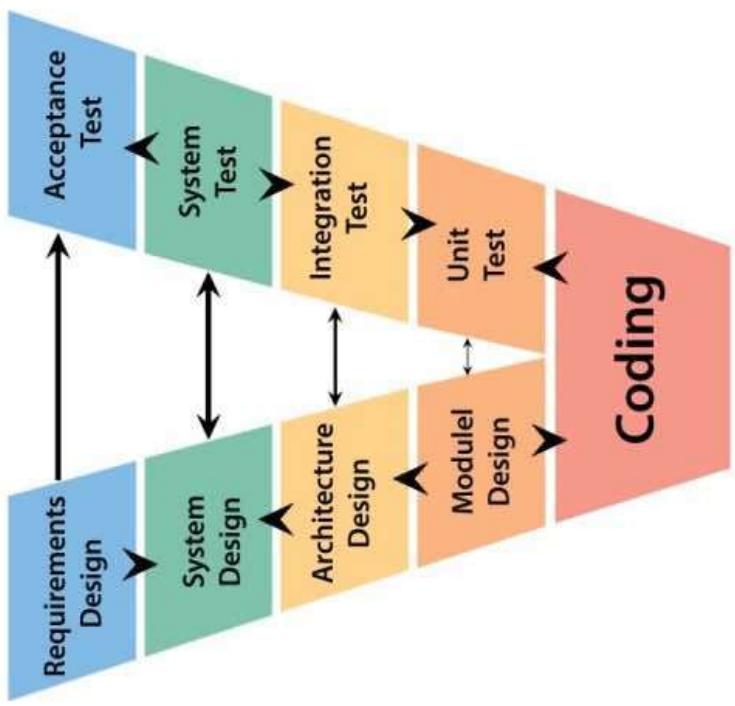
DISADVANTAGES OF SPIRAL MODEL

- Can be a costly model to use.
- Risk analysis requires highly specific expertise.
- Project's success is highly dependent on the risk analysis phase.
- Doesn't work well for smaller projects.



V-MODEL

- In this execution of processes happens in a sequential manner in V-shape. It is also known as Verification and Validation Model. V-Model is an extension of the Waterfall Model and is based on association of a testing phase for each corresponding development stage.



DIFFERENT PHASES OF V-MODEL

- **Business Requirement Analysis**

- In this first phase, the product requirements are understood from the customer perspective. This phase involves detailed communication with the customer to understand their expectations and exact requirement. The users are interviewed and a document called the
- **User Requirements Document**
- The user requirements document will typically describe the system's functional, interface, performance, data, security and other requirements as expected by the user. The user's carefully review this document as this document would serve as the guideline for the system designers in the system design phase.



• **System Design:**

- In the phase, system developers analyze and understand the business of the proposed system by studying the user requirements document. They figure out possibilities and techniques by which the user requirements can be implemented. System design comprises of understanding and detailing the complete hardware and communication setup for the product under development. System test plan is developed based on the system design

• **Architecture Design:**

- This is also referred to as **High Level Design (HLD)**. This phase focuses on system architecture and design. It provides overview of solution, platform, system, product and service/process. Usually more than one technical approach is proposed and based on the technical and financial feasibility the final decision is taken. An integration test plan is created in order to test the pieces of the software systems ability to work together.



- **Module Design:**

- The module design phase can also be referred to as low-level design. In this phase the actual software components are designed. It defines the actual logic for each and every component of the system. The designed system is broken up into smaller units or modules and each of them is explained so that the programmer can start coding directly. It is important that the design is compatible with the other modules in the system architecture and the other external systems.

- **Validation Phases:**

- In the V-Model, each stage of verification phase has a corresponding stage in the validation phase. The following are the typical phases of validation in the V-Model.



- **Unit Testing:**

- Unit tests designed in the module design phase are executed on the code during this validation phase. Unit testing is the testing at code level and helps eliminate bugs at an early stage. A unit is the smallest entity which can independently exist, e.g. a program module. Unit testing verifies that the smallest entity can function correctly when isolated from the rest of the codes/units.

- **Integration Testing:**

- Integration testing is associated with the architectural design phase. These tests verify that units created and tested independently can coexist and communicate among themselves within the system.



- **System Testing:**

- System Tests Plans are developed during System Design Phase.
 - System Test Plans are composed by client's business team.
- System Test ensures that expectations from application developed are met. The whole application is tested for its functionality, interdependency and communication. User acceptance testing: Acceptance testing is associated with the business requirement analysis phase and involves testing the product in user environment. UAT verifies that delivered system meets user's requirement and system is ready for use in real time.



ADVANTAGES OF THE SPIRAL MODEL

- Changing requirements can be accommodated.
- Allows extensive use of prototypes.
- Requirements can be captured more accurately.
- Users see the system early.
- Development can be divided into smaller parts and the risky parts can be developed earlier which helps in better risk management.

DISADVANTAGES OF SPIRAL MODEL

- Management is more complex.
- End of the project may not be known early.
- Not suitable for small or low risk projects and could be expensive for small projects.
- Process is complex
- Spiral may go on indefinitely.
- Large number of intermediate stages requires excessive documentation.

