Software Process/SDLC Models

Software Process model:

- Abstract representation of a software process
- Each Process model represents a process from different perspective

Following are the most important and popular SDLC models followed in the industry:

- ✦ Waterfall model.
- ✦ V Model
- ✦ Evolutionary development.
- ✦ Component-Based development model (CBSE).
- ✦ Process iteration(incremental and spiral Models)

The other related methodologies are Agile and RAD Model – Rapid Application Development

Waterfall Model

- First published model

- Because of the cascade from one phase to another, this model is known as ‘Waterfall Model’.

- It is also referred to as linear-sequential life cycle model.

- Each phase must be completed before the next phase can begin and there is no overlapping in the phases.

Waterfall Model

The sequential phases in Waterfall model are:

Requirement Gathering and analysis: All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.

- System’s services
- Constraints
- Goals

System Design: The requirement specifications from 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.

Waterfall Model

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

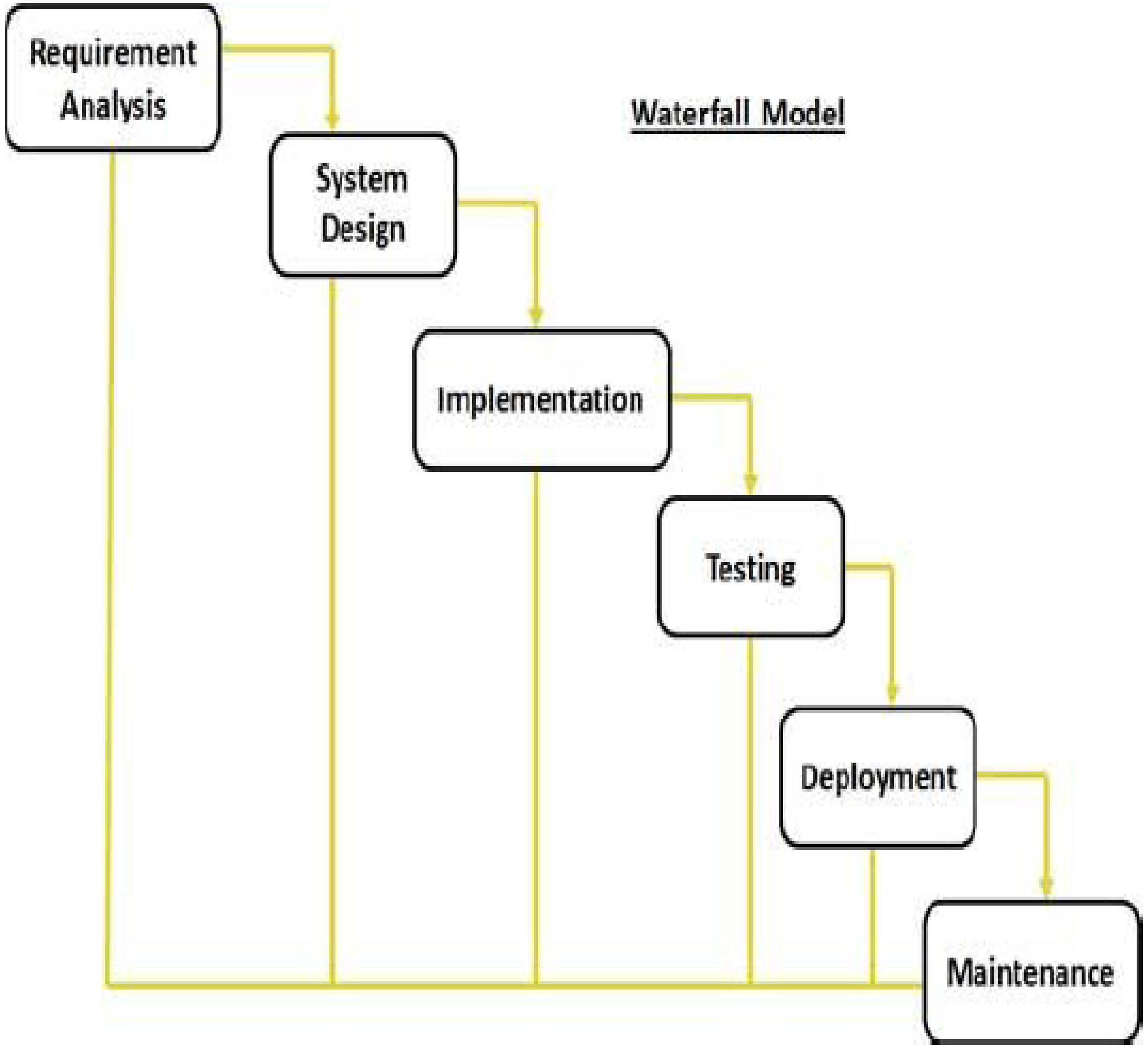
Integration and Testing: All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

Waterfall Model

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: There are some issues which come up in the client environment. To fix those issues patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

Following is a diagrammatic representation of different phases of waterfall model.



Waterfall Model

Applicability:

- Requirements are very well documented, clear and fixed
- Product definition is stable
- Technology is understood

Waterfall Model

Advantages:

- Documentation - at each phase
- Phases are processed and completed one at a time.
- Easy to arrange tasks.

Problems:

- Inflexible partitioning of the project into distinct stages makes it difficult to respond to changing customer requirements.

V Model

- A variation of the waterfall model
- Development phases are presented in a V-shaped graph.

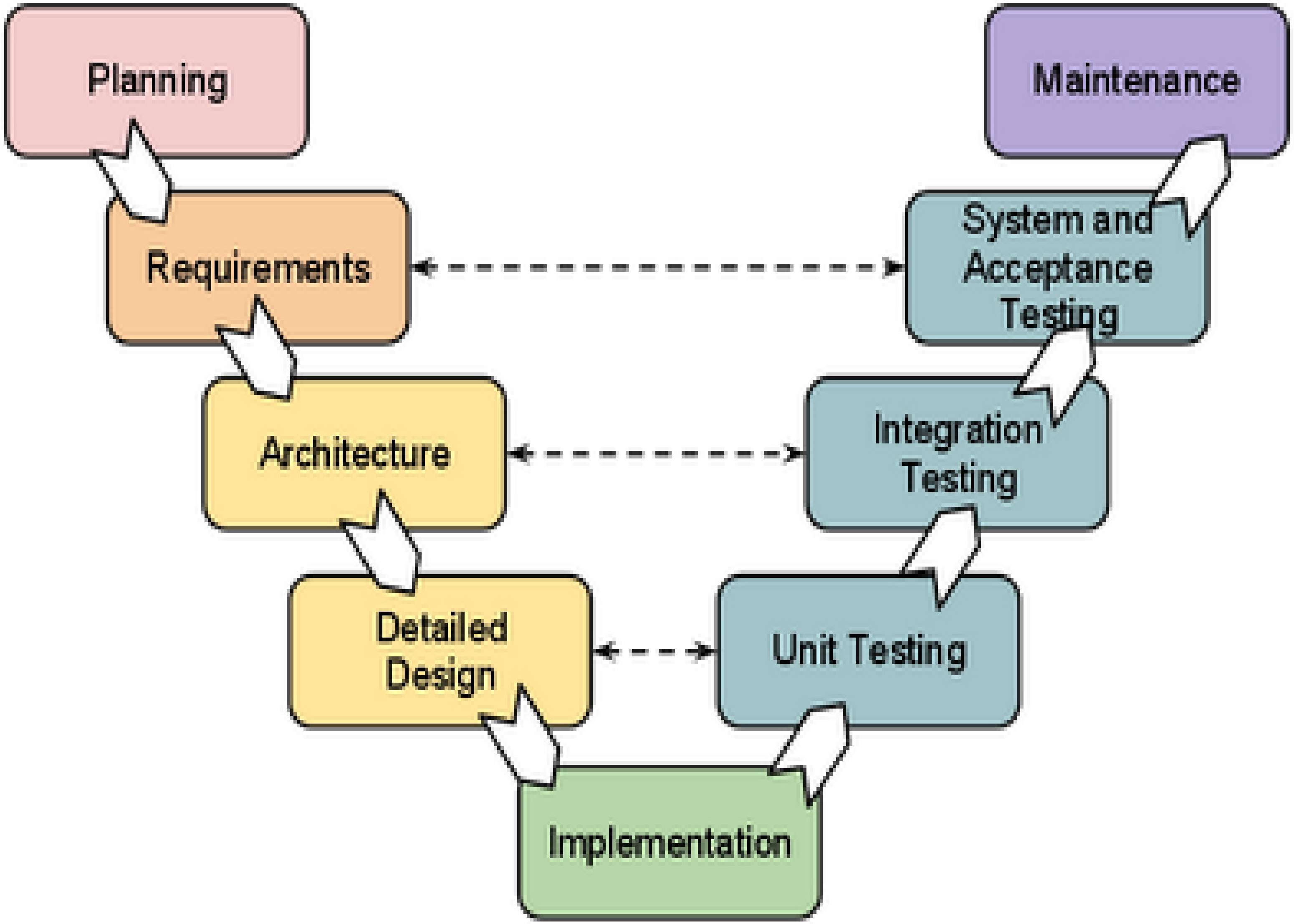
- It is also known as **Verification and Validation model**.

- For every **single** phase in the development cycle there is a directly associated testing phase.

Verification and Validation

- *Verification*
  - Does the product meet system specifications?
  - Have you built the product right?
- *Validation*
  - Does the product meet user expectations?
  - Have you built the right product?

V Model Design



V Model Verification Phases

Following are the Verification phases in V-Model

Business Requirement Analysis:

- This phase involves detailed communication with the customer to understand his expectations and exact requirement.
- The acceptance **test** design planning is done at this stage as business requirements can be used as an input for acceptance testing.

System Design:

- System design would comprise 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.

V Model Verification Phases

Architectural Design:

- System design is broken down further into modules taking up different functionality.
- The data transfer and communication between the internal modules and with the outside world (other systems) is clearly understood and defined in this stage.
- With this information, integration tests can be designed and documented during this stage.

Module Design:

- In this phase the detailed **internal design** for all the system modules is specified
- It is important that the design is compatible with the other modules in the system architecture and the other external systems.
- Unit tests can be designed at this stage

V Model

Coding Phase

- The actual coding of the system modules designed in the design phase is taken up in the Coding phase.
- The best suitable **programming** language is decided based on the system and architectural requirements.
- The coding is performed based on the coding guidelines and standards.
- The code goes through numerous code reviews and is optimized for best performance before the final build is checked into the repository.

V Model Validation Phases

Following are the Validation phases in V-Model:

Unit Testing:

- Unit tests designed in the module design phase
- Unit testing is the testing at code level and helps to eliminate bugs at an early stage

Integration Testing:

- Integration testing is associated with the architectural design phase.
- Integration tests are performed to test the coexistence and communication of the internal modules within the system.

V Model Validation Phases

System Testing:

- System tests check the entire system functionality and the communication of the system under development with external systems.

- Most of the software and hardware compatibility issues can be uncovered during system test execution.

Acceptance Testing:

- Acceptance testing is associated with the business requirement analysis phase and involves testing the product in user environment.

- It also discovers the non functional issues such as load and performance defects in the actual user environment.

V-model

Advantages:

- Testing activities like planning, test designing ,happens well before coding.
- Higher chances of success over the waterfall model.
- Works well for projects where requirements are easily understood.

Problems:

- No iteration of phases
- Difficult to handle changes in requirements throughout the life cycle
- No risk analysis

V-model

Applicability:

- The V-shaped model should be used for projects where requirements are clearly defined and fixed.
- The V-Shaped model should be chosen when technical resources are available with needed technical expertise.
- High confidence of customer is required for choosing the V-Shaped model approach.