

Name:- Pranav Mishra Registration No:- 12114762
Roll no:- RDOCO9A55 Course Code:- CAP437

Teacher:- Dr. Yasir Iqbal Sir Course Title:- Software Engineering

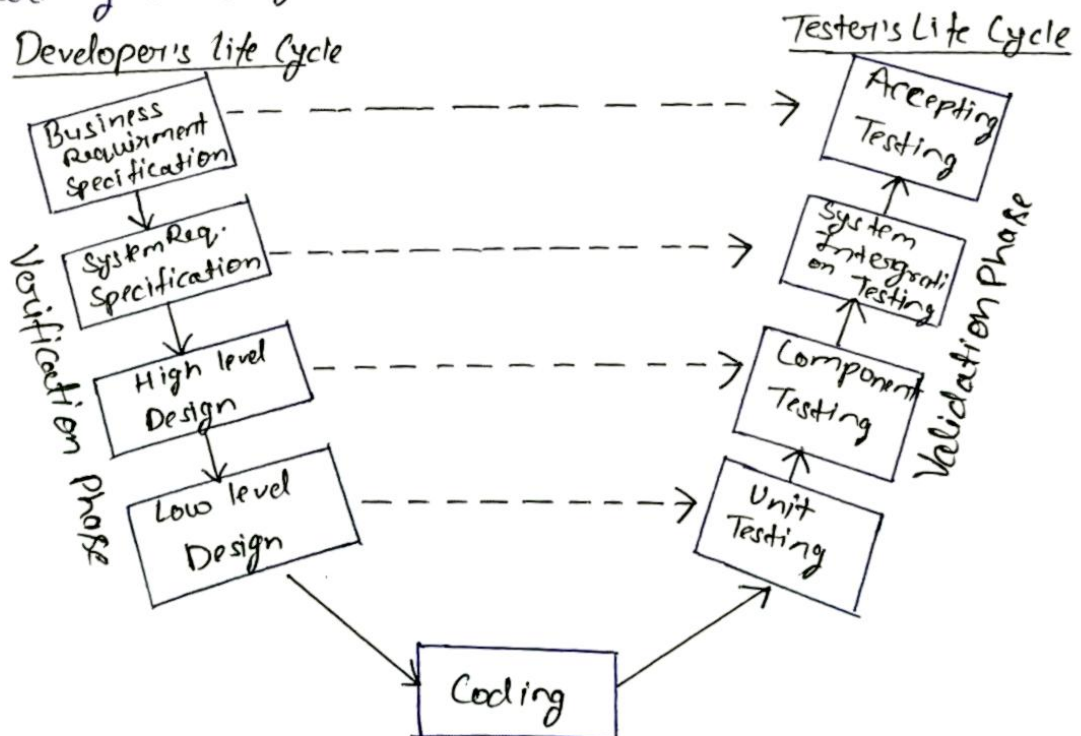
Q. Ans:- 1:-

The V-model is an SDLC model where execution of processes happens in a sequential manner in a V-shape. It is also known as Verification and Validation model.

The V-model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage.

Under the V-model, the corresponding testing phase of the development phase is planned in parallel.

So, there are Verification phases on one side of the 'V' and Validation phases on the other side. The Coding Phase joins the two sides of the V-model.



V-Model

Verification:- It involves a static analysis method (review) done without executing code. It is the process of evaluation of the product development process to find whether specified requirements meet.

Validation:- It involves dynamic analysis method (functional, non-functional), testing is done by executing code. Validation is the process to classify the software after the completion of the development process to determine whether the software meets the customer expectations and requirements.

There are the various phase of Verification stage of V-Model:-

- 1:- Business Requirement Analysis:- This stage contains communication to understand customer's expectations and exact requirement.
2. system Design: In this stage system engineers analyze and interpret the business of the proposed system by studying the user requirement document.
- 3:- Architecture Design: Architectural specifications are understood and designed in this phase. The system design is broken down further into modules taking up different functionality. This is also referred to as High Level Design (HLD).
- 4: Module Design:- In the module design stage, the system breaks down into small modules. The detailed design of the module is specified, which is known as Low level Design.
- 5: Coding Phase: After designing, the coding phase is started. Based on the requirements, a suitable programming language is decided. There are some guideline and standards for coding.

There are the various stages of Validation Phase of V-Model.

1:- Unit Testing:- In the V-Model, Unit Test Plans (UTPs) are developed during the module design phase.

These UTPs are executed to eliminate errors at code level or unit level.

2:- Integration Testing:- Integration Test Plans are developed during the Architectural Design Phase. These test verify that groups created and tested independently can coexist and communicate among themselves.

3:- System Testing:- System Test Plans are developed during System Design phase. Unlike Unit and Integration Test Plans, System Test Plans are composed by the client's business team.

4:- Acceptance Testing:- Acceptance testing is related to the business requirement analysis part.

It includes testing the software product in user atmosphere. Acceptance tests reveal the compatibility problems with the different systems, which is available within the user atmosphere.

Ans: 2 Advantages and Disadvantages of Waterfall Model:-

Advantages of Waterfall Model:-

→

What is Waterfall Model:-

The waterfall Model was the first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

Advantages:-

- This model is simple to implement also the number of resources that are required for it is minimal.
- The requirements are simple and explicitly declared; they remain unchanged during the entire project development.
- The start and end points for each phase is fixed, which makes it easy to cover progress.
- The release date for the complete product, as well as its final cost, can be determined before development.
- It gives easy to control and clarity for the customer due to a strict reporting system.
- Process and result are well documented.
- Works well for smaller projects where requirements are very well understood.

Disadvantages:-

- 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. So, risk and uncertainty is high with this process model.
- It is difficult to measure progress within stages.
- Cannot accommodate changing requirements.
- Adjusting scope during the life cycle can end a project.
- This model cannot accept the changes in requirement during development.
- It becomes tough to go back to the phase.
- Since the testing done at a later stage, it does not allow identifying the challenges and risks in the earlier phase, so the risk reduction strategy is difficult to prepare.

Ans:-3: Requirement Analysis:-

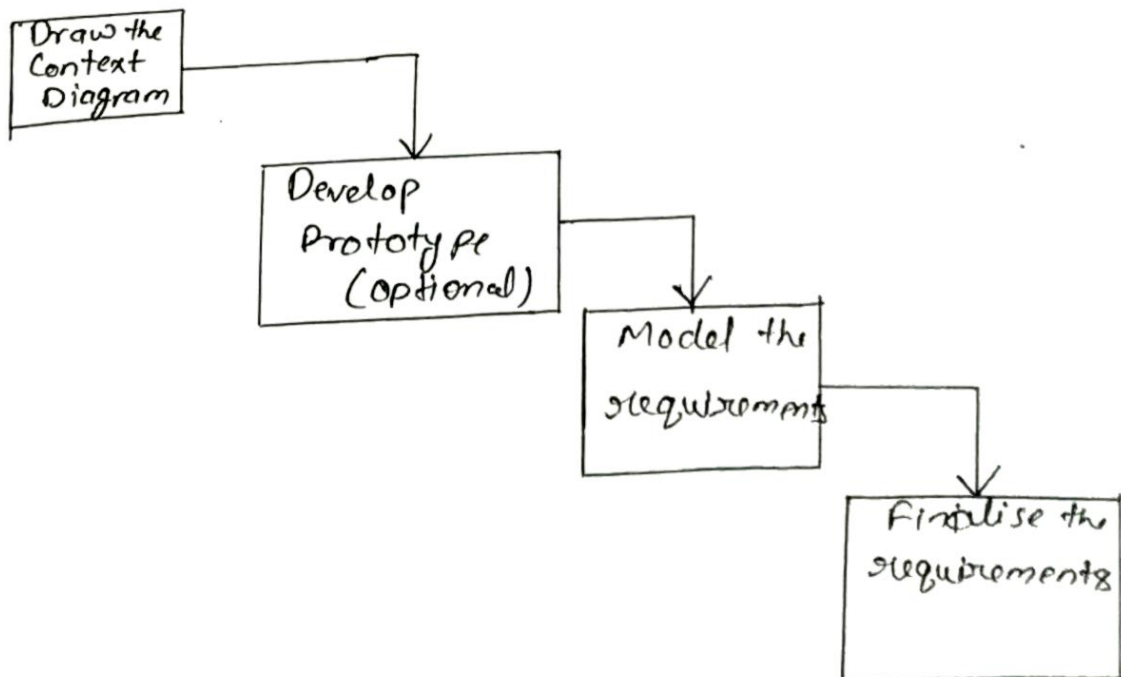
Requirement Analysis helps to understand, interpret, classify, and organize the software requirements in order to assess the feasibility, completeness, and consistency of the requirements.

Requirements Analysis is the process of defining the expectations of the users for an application that is to be built or modified.

Requirement analysis is significant and essential activity after elicitation.

Requirements are typically gathered from stakeholders such as owners, users and subject matter experts. This is always done in the early phase of any project to ensure that the final product conforms to all the requirements.

Steps of Requirements Analysis:-



Draw the Context diagram:-

The context diagram is a simple model that defines the boundaries and interfaces of the proposed systems with the external world. It identifies the entities outside the proposed system that interact with the system.

Development of a Prototype:-

One effective way to find out what the customer wants is to construct a prototype, something that looks and preferably acts as part of the system they say they want.

We can use their feedback to modify the prototype until the customer is satisfied continuously.

Model the requirements:-

The process usually consists of various graphical representations of the functions, data entities, external entities, and the relationships between them. The graphical view may help to find incorrect, inconsistent, missing requirements.

Finalise the requirements:-

After modelling the requirements, we will have a better understanding of the system behaviour. Now we finalize the analysed requirements, and the next step is to document these requirements in a prescribed format.