**03-02-2025**

**Testing:**

The process of identifying the products is satisfying the needs/requirements of the client or not.

**Objectives of Testing:**

* Identify defects, errors, or missing requirements to the products.
* Ensuring the correctness of project/software development.
* Reducing maintenance costs.
* Ensuring the product is bug free before shipment/release end user.
* Ensuring the reliability of the software.
* Identify usability issues and improve user satisfaction.

**Why Testing?**

Testing is essential in software development and other fields to ensure that to meet all standards before deployment.

**Quality:**

Quality is a continuous process throughout the software development life cycle.

**Software Testing:**

* Software Testing is a continuous process.
* High-quality software is delivered quickly and consistently at every stage of the development.
* Software Testing is used to detect and identify the defect present in the developed

Software.

**Need for Software Testing:**

* Continuous testing reduces the risk of deployment failures.
* Check whether the developed software is a user friendly or not.
* Security improvement.
* Check whether the customer is satisfying or not.
* Reduces cost of fixing bugs.

**Quality Software:**

* Continuous Testing.
* User friendly.
* When we develop the software in time with no bugs.
* Testing starts early in the development cycle.

**Project:**

If the software is developed based on the requirement of single customer, then it is known as project.

**Product:**

If the software is developed based on the requirement of multiple customers in the market, then it is known as product.

**Error:**

* Incorrect logic.
* Human mistake during development.

**Bug/Defect:**

* Incorrect implementation.
* Performance.

**Failure:**

* A bug that was not detecting during testing.
* Complete deviation from the system.

**04-02-2025**

**Software Testing:**

Software Testing is used to detect and identify the defect present in the developed software.

**They are two types of Software Testing**

1. Manual Testing
2. Automation Testing
3. **Manual Testing**

Manual Testing is a process where testing execute test cases without using automation tools.

**Advantages**

* More flexible for small projects.
* Less cost.
* It gives quick feedback.
* Easy to understand

**Disadvantages**

* Time consuming.
* Difficult to scale for large applications.

1. **Automation Testing**

Automation Testing involves using scripts and testing tools to execute test cases automatically.

**Advantages**

* Reduce human errors.
* Fast execution.

**Disadvantage**

* Maintenance cost high.
* Requires programming knowledge.

**Categories/Techniques of Software Testing**

1. Static testing
2. Dynamic Testing
3. **Static Testing**

Static Testing is a software testing technique that involves reviewing and analyzing code without executing.

1. **Dynamic Testing**

Dynamic Testing is a software testing technique that involves executing the code

1. White Box Testing
2. Black Box Testing
3. **White Box Testing**

White Box Testing is focuses on internal logic and code structure.

1. Unit Testing/Component Testing
2. Integration Testing
3. **Unit Testing/Component Testing**

Unit Testing is a software testing technique where individual components are tested.

1. **Integration Testing**

Integration Testing is a type of software testing where multiple components are tested together.

1. **Black Box Testing**

The funcationally of application are tested with out knowing its internal code.

1. System Testing
2. User Acceptance Testing
3. **System Testing**

System Testing is a type of software testing is to test whole applications.

1. **User Acceptance Testing**

User Acceptance Testing is a level of software testing in which software is tested for user acceptance.

1. Alpha Testing
2. Beta Testing
3. **Alpha Testing**

Testers will test the application software in the presence of customer.

1. **Beta Testing**

Testing is done by the customer to check whether software is working properly or not.

**Smoke Testing**

Basic test to check if the applications is stable for further testing.

**Difference between Bug & Defect**

|  |  |
| --- | --- |
| **Bug** | **Defect** |
| Identified by the testers | Identified by the developer |
| Fault in the software | Variance of the output |
| Bugs are caused by incorrect logic | Defect are caused by misunderstanding |

**Difference between Error & Failure**

|  |  |
| --- | --- |
| **Error** | **Failure** |
| If a program can’t run | If an end-user discover an issue with the s/w |
| Occurs during coding or designing | Occurs during runtime or testing |
| Mistake made by the developer | Software doesn’t meet the expected output |

**05-02-2025**

**Software development life cycle**

Software Development Life Cycle is a structured process used by software developers to design, develop, test, and deploy software.Software development life cycle is to ensure that the final product meets user requirements and is delivered efficiently.

1. **Planning**

* Planing is the foundation for the entire project.
* Identify risks and allocate resources.
* Develop a high-level project.
* Define project goals and objectives.

1. **Requirement Analysis**

* Define functional and non-functional requirements.
* It is the process of identifying, gathering, analyzing, and documenting the needs and expectations of stakeholder for a software.
* In this phase all the gathered requirements are documented as software requirement specification(SRS).

1. **Design**

* Design in the software development life cycle is the phase where the database structure, technical specifications are planned before coding begins.
* Select technologies, tools, and frameworks.

1. **Development**

* Development in the software development life cycle is the phase where the actual coding of the software takes place.
* Perform unit testing on individual components.

1. **Testing**

* Testing in the software development life cycle is the process of identifying the bugs.
* The process of checking whether the project is stratifying the requirements of the client.

1. **Deployment**

* Deployment in the software development life cycle is the phase where the software is released into the production environment.

1. **Maintenance**

* Monitor software performance
* Maintenance in the software development life cycle is the phase that occurs after the software has been developed

**Waterfall model**

Waterfall model follows a linear and sequential approach.It is one of the earliest software development life cycle.

* It is mainly used in small projects.
* Testing will be done only after completely the software was developed.
* Difficult to measure the progress of the project.
* Client involvement is very less.

**Phases of waterfall model**

1. **Requirement Gathering**

It is the process of identifying, gathering, analyzing, and documenting the needs and expectations of stakeholder for a software.

1. **System design**

Based on the gathering information planning is done by architects

1. **Implementation**

After the designing the plane the developer develop the code based on the blue print

1. **Testing**

Testers will perform different types of testing and produces relevant report

1. **Deployment**

Software application must be available for the end user to make use of it.

1. **Maintenance**

Maintenance in the software development life cycle is the phase that occurs after the software has been developed.

**Advantages of the Waterfall Model**

* Simple and easy to understand
* Clear documentation
* Suitable for smaller projects
* Early planning and design
* Easy to maintain

**Disadvantages of the Waterfall Model**

* Lack of flexibility
* Not ideal for large or complex projects
* Testing is done after coding

**Agile Methodology**

* Agile Methodology is a modern approach to software development
* Requirement of client is taken throughout the process
* Changes can be made at any stage
* Agile Methodology is delivering the piece of software which was develop
* Testing is done on piece of software developed
* Mostly used to develop large projects

**Agile Methodology has 3 basic principle**

* Customer no need to wait till the whole software is developed
* Delivering the piece of software which contains small functionalities which is developed and tested
* We can adopt/ Accept the requirements changes from the customer at any point of process

**Advantages of the Agile Methodology**

* Flexibility and Adaptability
* Release will be very fast
* Improved product quality
* Customer satification

**Disadvantages of the Agile Methodology**

* High dependency on team collaboration
* Difficult to measure progress
* Lack of documentation

**Agile Scrum meeting**

Scrum is a specific Agile Framework designed to help teams work together to deliver

Quality software

**Agile Sprint meeting**

A Sprint is a cycle within the scrum process where the team work on a subset of tasks to create a product increment.

**Agile review meeting**

The agile review meeting is focused on demonstrating the work done during the spirit and gathering feedback from stockholder.

**Agile Retrospective Meeting**

The Agile Retrospective Meeting often called the sprint retrospective is focused on reflecting on the process used during the sprint.

**06-02-2025**

**Sprial Model**

The Sprial Model is a risk-driven software development approach that combines elements of both Waterfall and Iterative Model.

* It is best suited for large, complex, high-risk projects.
* This model follows a cyclic approach.
* This model overcomes the drawback of waterfall model.

**Phases of Sprial Model**

1. **Planning**

* It includes estimating the cost,schedule and resources for the iteration.
* It also involves understanding the system requirements for communication between the system and the customer

1. **Risk Analysis**

* Risk Analysis includes identifying, estimating and monitoring and technical feasibility and management risk.
* Risk Analysis develops a prototypes.

1. **Engineering and Execution**

* Develop the code based on the client requirement.
* All the implemented features are then verified with through testing
* Testing can be done by unit testing , integration testing and system testing.

1. **Evaluate**

* The software is evaluated by the customer.
* Testing can be done by software testing and user acceptance testing

**Advantages**

* Sprial Model is suitable for large and complex projects.
* It handles the risk.
* It changes the implementation.
* Testing can done in every stage.

**Disadvantages**

* It is expensive and time taking process.
* It is not suitable for a simpler and smaller projects.
* It require more documentation.

**Incremental Model**

* Incremental Model is a process of software development where the requirements divided into multiple modules of the software development life cycle.
* In this module goes through the requirements, design, implementation and testing phases.

**Phases of Incremental Model**

1. **Requirement Analysis**

* It is the process of identifying, gathering, analyzing, and documenting the needs and expectations of stakeholder for a software.
* In this phase all the gathered requirements are documented as software requirement specification

1. **Design & Development**

* Design in the software development life cycle is the phase where the database structure, technical specifications are planned before coding begins.
* Develop the code based on the client requirement.

1. **Testing**

* All the implemented features are then verified with through testing.
* The process of checking whether the project is stratifying the requirements of the client.
* In this testing phase the various methods are used to test the behaviour of each task.

**Advantages**

* Errors are easy to be recognized
* Easier to test and debug
* More flexible
* Simple to manage risk because it handled during its iteration.

**Disadvantages**

* Need for good planning.
* Cost is high.

**Prototyping Model**

* Prototyping is a blue print of the software
* A Prototyping is a model or sample of a product made for the client evaluation and feedback regarding the desired finished product.
  + - 1. **Requirement Gathering and Analysis**
* It is the process of identifying, gathering, analyzing, and documenting the needs and expectations of stakeholder for a software.
  + - 1. **Quick Design**
* Design in the software development life cycle is the phase where the database structure, technical specifications are planned before coding begins.
  + - 1. **Build a Prototype**
* A Prototyping is a model or sample of a product made for the client evaluation and feedback regarding the desired finished product.
  + - 1. **User Evaluation & Feedback**
* This step describes the performance testing.The customer will tells the strengths and weaknesses of the design which was sent to the developer
  + - 1. **Refining & Iteration**
* Based on feedback. The prototype is modified,improved or redesigned.
  + - 1. **Final Product Development**
* Once the prototype is approved, full-scale development begins.

**Advantages**

* Early feedback
* Better understanding of requirements
* Improved communication
* Error detection
* User Involvement
* Cost saving

**Disadvantages**

* Time consuming
* Prototyping tools are expensive
* It is slow process because it takes more time to development

**Cloud**

Cloud is a network of remote servers that stores,manage, the process data over the

Internet.Cloud services are hosted and maintained by third-party providers

**Server**

A physical or virtual machine that stores data,run applications,and proceses request.

Server can exist locally or in the cloud.

**Cloud computing**

Cloud computing is a transformative pallandrom that involves delivering the applications

And servies through the internet.

**Two Types of Cloud Computing**

**Service Model**

**Deployment Model**

**Deployment Model**

1. **Public cloud**

It delivers the resources such computer,storage,networking,development,deployment

through the internet.They owned and run by the third party.

1. **Private cloud**

We can built run even we can access the services for a single organization,affering more control

1. **Hybrid cloud**

Combination of public and private cloud

1. **Community cloud**

Using multiple cloud providers to avoid reliance on a single vendor.