**Manual Testing**

**What is software testing?**

Software testing is the part of software development we use testing to send a quality product to the client and we testing to identify the bugs in the code.

**Software Quality**

Software quality is defined as a the product is according to the need of the customer

It should be delivered on time

It should in the budget

Easily maintainable

Bug free

**Product vs Project**

Project is defined as the application developed according to the needs of the customer is called as project

Product is defined as the application developed according to the needs of the various people or according to the market needs.

**Why do we need testing?**

We do testing to identify the bug in the code and to check it meets the end user requirements and to check application is according to the requirements or not and if we identify the bug after the deployment it will be more expensive

**Error:** The problem caused due to incorrect human action is called error

**Defect:** Deviation from the expected output to the actual output

**Failure:** if any bug raised after the deployment can be called as a failure

**Why software has bugs normally?**

Due to mis communication and Lack of experienced testers

Programming errors

Changing requirements

**Software development life cycle**

Software development lifecycle undergoes following steps:

1. Requirement analysis
2. Design
3. Implementation
4. Testing
5. Deployment
6. Support/Maintenance

**Waterfall model advantages and disadvantages**

Advantages of waterfall model:

Quality of product will be good

Initial investment will be less because testers are hired at the last stage of development

Rasing of bugs will be less because no changing of requirements in the middle of the project

It is best for small projects

Disadvantages of waterfall model:

Testing starts only after the development process

Development cost is high because if any error raises in the middle of the development process we have do the project from beginning and if any error raises in the middle of the code it continues in the next phase also and testing will be done at the end of the development

**Why testing is necessary?**

If any level of testing cannot declare that there is no defect in the product, then Why is it required at all?

**Development people assume**

Developers assume that they have developed the correct output id will not raise any errors in the code but coming to actual-scenario when we undertake actual execution part we don’t know whether it will work or not.

Developers may have excellent skills of coding but integration issues

Sometimes in a team each one developer some different parts of code while integrating the code may raise errors and the primary role of the software testing is not to demonstrate the correct ness of the code to ensure no failure after the deployment of the code.

Testing is a process of demonstrating there is no errors in the code

This process of testing is done in the acceptance testing if the application meets the required criteria, then it must be accepted by the customer

According to the number of defects it says about the quality of the software

If there is a greater number of defects in the code, we can say it is a bad software and bad process of development.

**software bug occurs when**

when developer write the code other than mentioned in the requirements

**The bugs are caused for many reasons but the main reason is specification**

Specification is the largest bug producer

If it is not maintained well or not communicated properly to the entire team the bugs will raise

Planning software is very important if it not done properly bugs will raise

**Next largest source of bugs is design**

If designing of software is not done properly bugs will raise

The cost of fixing bus will be more if not find the bugs in the early stage if we find the bugs in the specification stage the cost will be less as the stages increases gradually cost of fixing bugs will also increase.

**Software Testing**

It is a process of testing the application is bug free

Not only checking the software should find out the errors in the code.

**Types of testing:**

Testing is divided into two types

1. Manual testing

Manual testing includes testing the software manually without any automated tools is called manual testing

1. Automation testing

Automation testing includes testing the product my writing the scripts and by using other software to test the product.

The Automation testing is used to quickly rerun the test scenarios which are tested manually tested



**Static Testing Techniques**

In the phase of these static testing, we will not test the code but we test the documentation part as we know most of the errors raise at the designing phase only so the goal of the static testing is test the documentation part and reduce defect by reducing defect in the documentation from which the code is being developed.

**Review**: it is also a type of static testing

Review is process conducting meeting with the customers, managers, users or other interested people to know about their need before the implementation starts this helps to reduce the errors

**Walkthrough Review:**

It is not a formal review

It is led by authors In this process author guides participants his or her thought process or according to the document to archive common understanding and take feedback this process is very help full the people who are new to software discipline, or the people who cannot understand software development process.

**Inspection Review:**

It is the most formal review type it is led by trained moderators before this review documentation will be done and checked thoroughly by the reviewers before the meeting

**Informal review:**

Is unplanned and undocumented

**Technical review:**

Is documented fault detection process includes technical experts.

**Dynamic Testing Techniques**

The process of evaluating the system based upon its behaviour during execution.

**Levels of testing**

**Unit Testing:**

This testing is done by the developer only small units of code will be tested in this unit testing

**Integration testing:**

In this testing all the individual parts of code is integrated into one and check it is according to the data flow diagram in the TDD

**System testing:**

In this testing complete software will be tested by the tester

**Acceptance Testing**

In this level software is been tested for UAT at the client location where the software is being used.

**Alpha Testing**

This testing is done at the company in front of the customer

**Beta Testing:**

This testing is done by the customer to check if the software is running properly or not.

**Testing Types:**

**Functional Testing:**

In this functional testing QA determine if a piece of code is working according to the predetermined requirements it uses black box testing so the tester does not have any knowledge about in-built logic of the code only, we provide input and check whether we got expected out put or not.

**1.Nonfunctional testing:**

Load testing

Reliability

Readiness

**2.Black box testing:**

In black box testing is a type of software testing we test without any knowledge on internal logics of the code.

**3.White box testing:**

In white box testing we check the internal logics of the code white box testing is done by the developer

**4.Smoke testing:**

This test will be done on initial build it is compulsory to do smoke testing to check whether it is stable or not

**5.Sanity testing:**

Sanity testing will be done on stable build after the smoke testing if any test fails, we stop the further testing

Sanity testing is measurement to know whether we should perform further test or not.

**6.Retesting:**

If any test case fails, we send it to the QC and perform testing again.

**7.Regression testing:**

We perform regression testing when ever if any changes occur in the code we do both functional and on functional testing this type of testing to make sure the code works properly even after adding the new components to the complete program

**Smoke testing vs sanity testing:**

Smoke testing will be done in initial build it testcase fails reject the build if it passes send it to system testing or integration testing

Sanity test will be done after the smoke test and only on stable build if any test case fails, we stop doing further testing.

**Static testing:**

we perform static testing before implementation of the code to avoid bugs as we know 85% of bugs raise before the implementation phase only so by doing static testing, we can reduce the bugs in later stages

we have some static techniques:

Review

Walkthrough

Inspection

**Review**:

Review will be done before the development after the designing phase

In this phase we check whether the document is complete or not in this phase we do requirement review design review test plan review and the review can be conducted by anyone in the like manager tester coworker developer.

**Walkthrough:**

It is informal and not planed we can conduct when ever we required the author of the document will explain to their team.

**Inspection:**

It is the model formal type review prior intimation will be there for the team documentation part will be completed before this phase there will be 3-8 people in this meeting.

**Dynamic testing:**

Dynamic testing will be done after the implementation of the code in this phase we check the behaviour of the software

Dynamic testing requires the code to be executed

**Alpha testing:**

Alpha testing is the first testing which we done after the implementation of the software

Advantage: immediate solution after the defect is found

**Beta testing:**

Beta testing is done 1st time from the client side this testing is also called UAT testing.

Disadvantages: no immediate solution will be found after the defect is found

**Installation testing:**

In this phase we provide all the required recourses at the client location

In this type of testing test engineer check deployment process is according to the user manual or not user manual will be provided by the project manager.

**Usability testing:**

This kind of testing will be done for user friendly ness

**Monkey testing:**

It is used for game testing to check whether the system will crash

**Portability Testing:**

Developed application Should support multiple environment

**Forced error Testing:**

To check valid error message will display

**Debugging:**

executing program line by line for finding errors.

**Security Testing:**

Checking Security of application

**Inspection:**

It is formal evolution of software

**Concurrency Testing:**

Multiuser Testing

Some of the most popular SDLC models are

* Waterfall model
* Agile model
* V model
* Incremental model
* Spiral model

**1.Waterfall model**

In water fall model we follow some steps:

* Requirement and analysis
* Design
* Implementation
* Testing
* Deployment
* Maintenance

**2.V Shaped Model**

We use v shaped model in small and medium size projects. V model is the extension of waterfall model in this model testing will be done after every step.

**3.Agile model**

Agile is the commination of both iterative and incremental process in this agile model we can add our requirements at any stage when ever we require in agile model we can perform collaboration work this is the main advantage of agile model and also we can expect a piece of software so that customer can have good satisfaction.

**4.Incremental model**

In incremental model requirements are divided into modules each module undergoes all the phases like analysis,designing,testing,maintaince

We use this incremental model when project has lengthy development schedule and in this model customer interaction will be more.

**5.Spiral model**

Spiral model is also called as version control model

Spiral model overcome the drawbacks in the waterfall model

Spiral model is very expensive.

In every cycle new software will be released.

It releases new version after every cycle

Advantages:

Testing will be done after every cycle

Changes are allowed after every new cycle.

Disadvantages:

Testing will done after the end of each cycle

Requirement changes are not allowed in between the cycle

**QA (Quality Assurance)**

Responsible for highest possible quality

He is responsible for finding defect

**QC (Quality Control)**

He is responsible for finding defects

**QE (Quality Engineer)**

He responsible for writing code for testing Quality engineer comes under Automation engineer

**System testing:**

**1.GUI Testing:**

Testing GUI application, user interface testing

such as menus, check boxes, icon, images

not functional, just look and feel

check size and position of element

image quality, spelling check, alignment

Fonts are understanding or not

**2.Usability Testing:**

check the easiness of application

helping messages are display if user confuse

check user friendly application or not?

**3.Functional Testing:**

In functional testing we check the behaviour of the software in function testing we check the software is running according to the expected out put or not functional testing is like a black box testing

Some common functional testings are

Smoke testing

Sanity testing

Integration testing

System testing

UAT testing

**Non-functional testing:**

once functional testing done i.e. s/w work user requirement then do non

functional testing

performance testing

Load testing – gradually increase the load

Stress testing – suddenly increase the load (Eg: Online filling form)

Volume testing – how much data handle

security of software

recovery of application

Compatibility testing – work with all platform

**End to End testing techniques:**

Testing overall application

adding customer delete and edit customer logout

**Test case Design techniques:**

1.ECP Equivalence Class Partition

2.BVA value analysis

3.Design Table

4.State transition

5.Error guessing

**1.ECP Equivalence class partition:**

Equivalence partition is used to reduce the number of test cases and time we Divide the data and I we get the values in the range then we can deicide it is valid if we don’t get in the expected range we can decide it is invalid. And we divide the input data into partitions

**2.BVA value analysis:**

Boundary value Analysis

It focuses mainly on boundary values only where more errors or likely to occur it mainly focus on the edge range only it misses the errors in the middle of the range BVA is a black box testing technique mainly it checks the boundary values where the more test cases may fail

**3.Design table technique:**

We perform this condition when we have more conditions and based on the condition we have to perform the action.

**4.State transition technique:**

Take action depends on the state

For example in login page if test cases failed for 3-4 times then we block the account.