Module - 2

(manual testing)

- What is Exploratory Testing?

> Exploratory Testing is a type of software testing where Test cases are not created in advance but testers check system on the fly. They may note down ideas about what to test before test execution. The focus of exploratory testing is more on testing as a "thinking" activity.

- What is traceability matrix?

> A Traceability Matrix is a document that co-relates any two-baseline documents that require a many-to-many relationship to check the completeness of the relationship. It is used to track the requirements and to check the current project requirements are met.

- What is Boundary value testing?

>The testing is an analysis testing process that does testing practices at the boundaries of the partition covering all the testing defects where equivalence testing alone was difficult to handle those defects.

- What is Equivalence partitioning testing?

> Equivalence class testing (Equivalence class Partitioning) is a black-box testing technique used in software testing as a major step in the Software development life cycle (SDLC).

- What is Integration testing?

> Integration Testing is defined as a type of testing where software modules

are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. The purpose of this level of testing is to expose defects in the interaction between these software modules when they are integrated.

- What determines the level of risk?

> Risks should be prioritized according to their level, which is obtained by assessing the likelihood of the event occurring and the impact of that event.

- What is Alpha testing?

> This testing got its name alpha because it is performed at an early stage, close to the end of the software development life cycle but before beta testing. This kind of testing is performed by internal employees of the company which includes testers and Quality Analysts (QA) people.

- What is beta testing?

> Beta testing can be considered "pre-release testing."Beta testing is also sometimes referred to as user acceptance testing (UAT) or end-user testing. In this phase of software development, applications are subjected to real-world testing by the intended audience for the software.

- What is component testing?

> Component testing is defined as a software testing type, in which the testing is performed on each individual component separately without integrating with other components. It's also referred to as Module Testing when it is viewed from an architectural perspective. Component Testing is also referred to as Unit Testing, Program Testing, or Module Testing**.**

- What is functional system testing?

> functional testing of a system includes tests that evaluate the functions

that the system must perform. Functional requirements may be described in work products (requirements, specification, business need, user story, use case) and in the functional specification**.**

- What is Non-Functional Testing?

> Non-Functional Testing is defined as a type of Software testing to check non-functional aspects (performance, usability, reliability, etc) of a software application. It is designed to test the readiness of a system as per nonfunctional parameters which are never addressed by functional testing.

- What is GUI Testing?

> GUI Testing is a software testing type that checks the Graphical User Interface of the Software. The purpose of Graphical User Interface (GUI) Testing is to ensure the functionalities of software application work as per specifications by checking screens and controls like menus, buttons, icons, etc.GUI is what the user sees.

- What is Adhoc testing?

> Ad hoc Testing is an informal or unstructured software testing type that aims to break the testing process in order to find possible defects or errors at an early possible stage. Ad hoc testing is done randomly and it is usually an unplanned activity that does not follow any documentation and test design techniques to create test cases.

- What is load testing?

> Load Testing is a non-functional software testing process in which the performance of software application is tested under a specific expected load. It determines how the software application behaves while being accessed by multiple users simultaneously. The goal of Load Testing is to improve performance bottlenecks and to ensure stability and smooth functioning of software application before deployment.

- What is stress Testing?

> Stress Testing is a type of software testing that verifies stability & reliability of software application. The goal of Stress testing is measuring software on its robustness and error handling capabilities under extremely heavy load conditions and ensuring that software doesn't crash under crunch situations. It even tests beyond normal operating points and evaluates how software works under extreme conditions.

- What is white box testing and list the types of white box testing?

> White Box Testing is a testing technique in which software's internal structure, design, and coding are tested to verify input-output flow and improve design, usability, and security. In white box testing, code is visible to testers, so it is also called Clear box testing, Open box testing, Transparent box testing, Code-based testing, and Glass box testing.

Unit Testing: It is often the first type of testing done on an application. Unit Testing is performed on each unit or block of code as it is developed. Unit Testing is essentially done by the programmer. As a software developer, you develop a few lines of code, a single function or an object and test it to make sure it works before continuing Unit Testing helps identify a majority of bugs, early in the software development lifecycle. Bugs identified in this stage are cheaper and easy to fix.

Testing for Memory Leaks: Memory leaks are leading causes of slower running applications. A QA specialist who is experienced at detecting memory leaks is essential in cases where you have a slow running software application.

- What is black box testing? What are the different black box testing techniques?

> Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal

code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing.

Equivalence Class Testing: It is used to minimize the number of possible test cases to an optimum level while maintains reasonable test coverage.

Boundary Value Testing: Boundary value testing is focused on the values at boundaries. This technique determines whether a certain range of values are acceptable by the system or not. It is very useful in reducing the number of test cases. It is most suitable for the systems where an input is within certain ranges.

Decision Table Testing: A decision table puts causes and their effects in a matrix. There is a unique combination in each column.

- Mention what are the categories of defects?

> Arithmetic Defects.

Logical Defects.

Syntax Defects.

Multithreading Defects.

Interface Defects.

Performance Defects.

- Mention what bigbang testing is?

> Big Bang Integration Testing is an integration testing strategy wherein all units are linked at once, resulting in a complete system. When this type of testing strategy is adopted, it is difficult to isolate any errors found, because attention is not paid to verifying the interfaces across individual units.

- What is the purpose of exit criteria?

> Exit criterion is used to determine whether a given test activity has been completed or NOT. Exit criteria can be defined for all of the test activities right from planning, specification and execution. Exit criterion should be part of test plan and decided in the planning stage.

- When should "Regression Testing" be performed?

> Regression testing can be performed on a new build when there is a significant change in the original functionality. It ensures that the code still works even when the changes are occurring. Regression means Re-test those parts of the application, which are unchanged.

- What is 7 key principles? Explain in detail?

> Testing shows presence of defects

Exhaustive testing is not possible

Early testing

Defect clustering

Pesticide paradox

Testing is context dependent

Absence of errors fallacy

1) Exhaustive testing is not possible

Yes! Exhaustive testing is not possible. Instead, we need the optimal amount of testing based on the risk assessment of the application.

And the million dollar question is, how do you determine this risk?

To answer this let’s do an exercise

In your opinion, Which operation is most likely to cause your Operating system to fail?

I am sure most of you would have guessed, Opening 10 different application all at the same time.

So if you were testing this Operating system, you would realize that defects are likely to be found in multi-tasking activity and need to be tested thoroughly which brings us to our next principle Defect Clustering

2) Defect Clustering

Defect Clustering which states that a small number of modules contain most of the defects detected. This is the application of the Pareto Principle to software testing: approximately 80% of the problems are found in 20% of the modules.

By experience, you can identify such risky modules. But this approach has its own problems

If the same tests are repeated over and over again, eventually the same test cases will no longer find new bugs.

3) Pesticide Paradox

Repetitive use of the same pesticide mix to eradicate insects during farming will over time lead to the insects developing resistance to the pesticide Thereby ineffective of pesticides on insects. The same applies to software testing. If the same set of repetitive tests are conducted, the method will be useless for discovering new defects.

To overcome this, the test cases need to be regularly reviewed & revised, adding new & different test cases to help find more defects.

Testers cannot simply depend on existing test techniques. He must look out continually to improve the existing methods to make testing more effective. But even after all this sweat & hard work in testing, you can never

claim your product is bug-free. To drive home this point, let's see this video of the public launch of Windows 98

4) Testing shows a presence of defects

Hence, testing principle states that – Testing talks about the presence of defects and don't talk about the absence of defects. i.e. Software Testing reduces the probability of undiscovered defects remaining in the software but even if no defects are found, it is not a proof of correctness.

But what if, you work extra hard, taking all precautions & make your software product 99% bug-free. And the software does not meet the needs & requirements of the clients.

5) Absence of Error – fallacy

It is possible that software which is 99% bug-free is still unusable. This can be the case if the system is tested thoroughly for the wrong requirement. Software testing is not mere finding defects, but also to check that software addresses the business needs. The absence of Error is a Fallacy i.e. Finding and fixing defects does not help if the system build is unusable and does not fulfill the user's needs & requirements.

6) Early Testing

Early Testing – Testing should start as early as possible in the Software Development Life Cycle. So that any defects in the requirements or design phase are captured in early stages. It is much cheaper to fix a Defect in the early stages of testing. But how early one should start testing? It is recommended that you start finding the bug the moment the requirements are defined. More on this principle in a later training tutorial.

7) Testing is context dependent

Testing is context dependent which basically means that the way you test an e-commerce site will be different from the way you test a commercial

off the shelf application. All the developed software's are not identical. You might use a different approach, methodologies, techniques, and types of testing depending upon the application type. For instance testing, any POS system at a retail store will be different than testing an ATM machine.

- Difference between QA v/s QC v/s Tester

>

S.N.	Quality Assurance	Quality Control	Testing
1	Activities which ensure the implementation of processes, procedures and standards in context to verification of developed software and intended requirements.	Activities which ensure the verification of developed software with respect to documented (or not in some cases) requirements.	Activities which ensure the identification of bugs/error/defects in the Software.
2	Focuses on processes and procedures rather than conducting actual testing on the system.	Focuses on actual testing by executing Software with intend to identify bug/defect through implementation of procedures and process.	Focuses on actual testing.
3	Process oriented activities.	Product oriented activities.	Product oriented activities.
4	Preventive activities.	It is a corrective process.	It is a preventive process.
5	It is a subset of Software Test Life Cycle (STLC).	QC can be considered as the subset of Quality Assurance.	Testing is the subset of Quality Control.

- Difference between Smoke and Sanity?

> Smoke Testing is performed to ascertain that the critical functionalities of the program are working fine. Sanity testing is done at random to verify that each functionality is working as expected.

- Difference between verification and Validation

> Verification is a process of determining if the software is designed and developed as per the specified requirements. Validation is the process of

checking if the software (end product) has met the client's true needs and expectations.

- Explain types of Performance testing.

> Stress Testing.

Spike Testing.

Load Testing.

Endurance Testing.

Volume Testing.

Scalability Testing.

- What is Error, Defect, Bug and failure?

> ERROR:

An error is a mistake, misconception, or misunderstanding on the part of a software developer. In the category of the developer, we include software engineers, programmers, analysts, and testers. For example, a developer may misunderstand a de-sign notation, or a programmer might type a variable name incorrectly – leads to an Error. It is the one that is generated because of the wrong login, loop or syntax. The error normally arises in software; it leads to a change in the functionality of the program.

BUG:

A bug is the result of a coding error. An Error found in the development environment before the product is shipped to the customer. A programming error that causes a program to work poorly, produce incorrect results or crash. An error in software or hardware that causes a program to malfunction. A bug is the terminology of Tester.

DEFECT:

It can be simply defined as a variance between expected and actual. The defect is an error found AFTER the application goes into production. It commonly refers to several troubles with the software products, with their external behavior or with its internal features. In other words, a Defect is a difference between expected and actual results in the context of testing. It is the deviation of the customer requirement.

FAILURE:

A failure is the inability of a software system or component to perform its required functions within specified performance requirements. When a defect reaches the end customer it is called a Failure. During development, Failures are usually observed by testers.

- Difference between Priority and Severity

> Severity means the seriousness of the defect in the product functionality. Priority means how soon the bug should be fixed. The test engineer determines the severity level of the defect.

- What is Bug Life Cycle?

> Defect life cycle, also known as Bug Life cycle is the journey of a defect cycle, which a defect goes through during its lifetime. It varies from organization to organization and also from project to project as it is governed by the software testing process and also depends upon the tools used.

- Explain the difference between Functional testing and NonFunctional testing

> Functional testing ensures that the functions and features of the application work properly. Non-functional testing examines other aspects of how well the application works. Functional testing tests the functionality

of an app. Non-functional testing tests the performance of these functions.

- To create HLR & TestCase of
- 1)(Instagram , Facebook) only first page

> **https://docs.google.com/spreadsheets/d/1_9IqmT4aQvbZsj16fz_GZIZYQBuNX0Pqurg2BRQTxMw/edit#gid=0**

- What is the difference between the STLC (Software Testing Life Cycle) and SDLC?

> Software Development Life Cycle (SDLC) is a sequence of different activities performed during the software development process. Software Testing Life vsoftware testing process. SDLC is mainly related to software development.

- What is the difference between test scenarios, test cases, and test script?

> A test scenario is any functionality that a software testing company can examine. It is also called a Test Condition or Test Possibility. A test case is a document that lists the steps a QA engineer needs to execute. A test script is a short program written in a programming language.

- Explain what Test Plan is? What is the information that should be covered.

> The test plan conveys how the test will be performed. This includes defining test objectives, test approach, test tools, test environment, test schedules and team responsibilities and composition.

- What is priority?

> something that is more important than other things and that needs to be done or dealt with first. Reorganizing the sales force will be a top priority

for the new president.

- What is severity?

> the quality or state of being severe : the condition of being very bad, serious, unpleasant, or harsh. the severity of the climate. the severity of the punishment. Medication can help shorten the illness and lessen its severity.

- Bug categories are…

> Performance Bugs: ...

Security Bugs: ...

Unit Level Bugs: ...

Functional Bugs: ...

Usability Bugs: ...

Syntax Errors: ...

Compatibility Errors: ...

Logic Bugs:

- Advantage of Bugzila .

> it is an open-source widely used bug tracker;

it is easy in usage and its user interface is understandable for people without technical knowledge;

it easily integrates with test management instruments;

it integrates with an e-mailing system;

it automates documentation.

- Difference between priority and severity.

> Severity is basically a parameter that denotes the total impact of a given defect on any software. Priority is basically a parameter that decides the order in which we should fix the defects. Severity relates to the standards of quality. Priority relates to the scheduling of defects to resolve them in software.

- What are the different Methodologies in Agile Development Model?

> There are 5 main Agile methodologies: Scrum, Kanban, Extreme Programming (XP), Lean Development e Crystal.

- Explain the difference between Authorization and Authentication in Web testing.What are the common problems faced in Web testing?

> Simply put, authentication is the process of verifying who someone is, whereas authorization is the process of verifying what specific applications, files, and data a user has access to. The situation is like that of an airline that needs to determine which people can come on board.

- To create HLR & TestCase of WebBased (WhatsApp web , Instagram)

> https://docs.google.com/spreadsheets/d/1_9IqmT4aQvbZsj16fz_GZIZYQBuNX0Pqurg2BRQTxMw/edit#gid=0

- Write a scenario of only Whatsapp chat messages

> https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- Write a Scenario of Pen

> https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- Write a Scenario of Pen Stand

> https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- Write a Scenario of Door

> https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- Write a Scenario of ATM

> https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- When to used Usablity Testing?

> If possible, usability testing can and should be conducted on the current iteration of a product before beginning any new design work, after you've begun the strategy work around a brand new site or app.

- What is the procedure for GUI Testing?

> We can perform the GUI testing with the help of Automation tools, which can be completed in two types. Throughout the record part, the test steps are encapsulated by the automation tool. And in the playback, these recorded test steps are implemented on the application under test. For example: QTP.

- Write a scenario of Microwave Owen

> https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PX

hJSVvck4B9EYepU/edit#gid=0

- Write a scenario of Coffee vending Machine

>
https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- Write a scenario of chair

>
https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- Write a Scenario of Wrist Watch

>
https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- Write a Scenario of Lift(Elevator)

>
https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- Write a Scenario of whatsapp Group (generate group)

>
https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- Write a Scenario of instagram (video call with chat)

>
https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PX

hJSVvck4B9EYepU/edit#gid=0

- To Create Scenario (Positive & Negative) 2. facebook Chat on Mobile

>
https://docs.google.com/spreadsheets/d/1uKuIPE0vWRYCkQlI1ElxosIju9PXhJSVvck4B9EYepU/edit#gid=0

- Write a Scenario of Whatsapp payment

>
https://docs.google.com/spreadsheets/d/1VIS0VdapU8vxSDhLZTpp_G_O2uSm6XsLfuP0kHvT04A/edit#gid=0