***Module 2***

**1. What is Exploratory Testing?**

1. Exploratory testing is a concurrent process where.
2. Test design execution and logging happen simultaneously.
3. Makes use of experience, heuristics and test patterns.
4. Though the current trend in testing is to push for automation, exploratory testing is a new way of thinking. Automation has its limits.

**2. What is traceability matrix?**

1. To protect against changes you should be able to trace back from every system component to the original requirement that caused its presence.
2. A software process should help you keeping the virtual table up-to-date.
3. Simple technique may be quite valuable.

**3. What is Boundary value testing?**

1. Boundary value analysis is a methodology for designing test cases that concentrates software testing effort on cases near the limits of valid ranges.
2. Boundary value analysis generates test cases that highlight errors better than equivalence partitioning.
3. At those points when input values change from valid to invalid errors are most likely to occur.

**4. What is Equivalence partitioning testing?**

1. The numbers fall into a partition where each would have the same, or equivalent, result i.e. an Equivalence Partition (EP) or Equivalence Class.
2. In EP we must identify Valid Equivalence partitions and Invalid Equivalence partitions where applicable.
3. **Plus there are 2 Invalid partitions,**

**5. What is Integration testing?**

1. System Integration Testing is testing between the **System and Acceptance** phases.
2. The System has already proven to be functionally correct, what remains to be tested is how the system reacts to other systems and/or organisations.
3. In terms of the V Model, Systems Integration Testing corresponds to the Functional and Technical Specification phases of the software development lifecycle.
4. Black Box testing techniques used.

**6. What determines the level of risk?**

1. A properly designed test that passes, reduces the overall level of Risk in a system.
2. Risk:- **A factor that could result in future negative consequences usually expressed as impact and likelihood.**
3. When testing does find defects, the Quality of the software system increases when those defects are fixed.
4. The Quality of systems can be improved through Lessons learned from previous projects.

**7. What is Alpha testing?**

1. It is always performed by the developers at the software development site.
2. Sometimes it is also performed by Independent Testing Team.
3. It is conducted for the software application and project.
4. Alpha Testing is definitely performed and carried out at the developing organizations location with the involvement of developers.

**8. What is beta testing?**

1. It is always performed by the customers at their own site.
2. It is not performed by Independent Testing Team.
3. Beta Testing is always open to the market and public.
4. It is usually conducted for software product.

**9. What is component testing?**

Component(Unit) A minimal software item that can be tested in isolation. It means **A unit is the smallest testable part of software.**

Unit testing is the first level of testing and is performed prior to Integration Testing.

**10. What is functional system testing?**

1. Functional Testing: Testing based on an analysis of the specification of the functionality of a component or system.
2. **Specification:-** E.g. Requirements specification, Use Cases, Functional specification or maybe undocumented.
3. **Function:-** what the system does.

**11. What is Non-Functional Testing?**

1. Non-Functional Testing: Testing the attributes of a component or system that do not relate to functionality, e.g. reliability, efficiency, usability, interoperability, maintainability and portability.
2. Non-functional testing includes, but is not limited to, performance testing, load testing, stress testing, usability testing, maintainability testing, reliability testing and portability testing.

**12. What is GUI Testing?**

1. Graphical User Interface (GUI) testing is the process of testing the system’s GUI of the System under Test. GUI testing involves checking the screens with the controls like menus, buttons, icons, and all types of bars – tool bar, menu bar, dialog boxes and windows etc.
2. Check all the GUI elements for size, position, width, length and acceptance of characters or numbers. For instance, you must be able to provide inputs to the input fields.
3. Check you can execute the intended functionality of the application using the GUI.

**13. What is Adhoc testing?**

1. Adhoc testing is an informal testing type with an aim to break the system.
2. This testing is primarily performed if the knowledge of testers in the system under test is very high.
3. Adhoc Testing does not follow any structured way of testing and it is randomly done on any part of application.

**14. What is load testing?**

1. Its a performance testing to check system behavior under load. Testing an application under heavy loads, such as testing of a web site under a range of loads to determine at what point the system’s response time degrades or fails.
2. Load testing is a kind of performance testing which determines a system’s performance under real-life load conditions. This testing helps determine how the application behaves when multiple users access it simultaneously.

**15. What is stress Testing?**

Stress Testing is done in order to check when the application fails by reducing the system resources such as RAM, HDD etc. and keeping the number of users as constant.

**16. What is white box testing and list the types of white box testing?**

1. Testing based on an analysis of the internal structure of the component or system.
2. Structure-based testing technique is also known as ‘white-box’ or ‘glass-box’ testing technique because here the testers require knowledge of how the software is implemented, how it works.

* **Types**

1. **Major (High):** The defect that results in the termination of the complete system or one or more component of the system and causes extensive corruption of the data. The failed function is unusable but there exists an acceptable alternative method to achieve the required results then the severity will be stated as major.
2. **Moderate (Medium):** The defect that does not result in the termination, but causes the system to produce incorrect, incomplete or inconsistent results then the severity will be stated as moderate.
3. **Minor (Low):** The defect that does not result in the termination and does not damage the usability of the system and the desired results can be easily obtained by working around the defects then the severity is stated as minor.
4. **Cosmetic:** The defect that is related to the enhancement of the system where the changes are related to the look and field of the application then the severity is stated as cosmetic.

**17. What is black box testing? What are the different black box testing techniques?**

1. **Black-box testing:** Testing, either functional or non-functional, without reference to the internal structure of the component or system.
2. Specification-based testing technique is also known as ‘black-box’ or input/output driven testing techniques because they view the software as a black-box with inputs and outputs.

* Equivalence partitioning
* Boundary value analysis
* Decision tables
* State transition testing
* Use-case Testing
* Other Black Box Testing

**18. Mention what are the categories of defects?**

1. **Error of Commission:-** Commission means instruction or some kind of command given. Now the error in commission means the error in made in command or instruction. For example, suppose I wrote a loop which I was trying to run 10 times but I command it to run more than 10 times by mistake this is the error of commission.
2. **Errors of Omissions:-** As name is already describing error of omission is some thing which happens accidentally. Omission word means something left out or executed. Practical most common example of this error is suppose we make a function in programming open its bracket but forget to close at the end.
3. **Error of Clarity:-** The most common error in the natural languages. This error happens due to miss understanding between the developer and client. It travels most of the time from the requirements to the software.
4. **Error of Speed or Capacity:-** The name of the error is itself enough i think to tell about it this error. Your software is working fine but not working in the required time this is the error of speed. When it comes to capacity it can be relevant to memory. For example, a small integer is declared where the long integer was required.

**19. Mention what bigbang testing is?**

Integration testing is a type of testing that is used to check the functionality of integrated components of a software system. It is usually performed after unit testing and before validation testing. In integration testing, individual software components are combined and tested as a group. The purpose of this testing is to check if the components work together as expected. Integration testing can be either bottom-up or top-down.

**20. What is the purpose of exit criteria?**

1. Align your teams on a common definition of test completion
2. Ensure your product meets completion standards before entering the next stage, which avoids costly project delays
3. Create clear parameters for test engineers to evaluate software

**21. When should "Regression Testing" be performed?**

1. Change in requirements and code is modified according to the requirement.
2. New feature is added to the software.
3. Defect fixing.
4. Performance issue fix.

**22. What is 7 key principles? Explain in detail?**

1. **Testing shows presence of Defects:-** Testing can show that defects are present, but cannot prove that there are no defects.

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. However Testing cannot prove that there are no defects present.

1. **Exhaustive Testing is Impossible:-** Testing everything including all combinations of inputs and preconditions is not possible.

So, instead of doing the exhaustive testing we can use risks and priorities to focus testing efforts.

This is very unlikely that the project timescales would allow for this number of tests.

1. **Early Testing:-** Testing activities should start as early as possible in the software or system development life cycle, and should be focused on defined objectives.

Testing activities should start as early as possible in the development life cycle

1. **Defect** **Clustering:-** A small number of modules contain most of the defects discovered during pre-release testing, or are responsible for the most operational failures.

Defects are not evenly spread in a system

1. **The Pesticide Paradox:-** If the same tests are repeated over and over again, eventually the same set of test cases will no longer find any new defects.

To overcome this “pesticide paradox”, the test cases need to be regularly reviewed and revised, and new and different tests need to be written to exercise different parts of the software or system to potentially find more defects.

1. **Testing is Context Dependent:-** Testing is basically context dependent.

Testing is done differently in different contexts.

Different kinds of sites are tested differently.

1. **Absence of Errors Fallacy:-** If the system built is unusable and does not fulfill the user’s needs and expectations then finding and fixing defects does not help.

Even after defects have been resolved it may still be unusable and/or does not fulfil the users’ needs and expectations.

**23. Difference between QA v/s QC v/s Tester**

|  |  |  |  |
| --- | --- | --- | --- |
| No. | QA | QC | Tester |
| 1 | Activities which ensure the implementation of 1 2 3 4 5 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. | QC can be considered as the subset of Quality Assurance. |
|  |  |  |  |
|  |  |  |  |

**24. Difference between Smoke and Sanity?**

|  |  |  |
| --- | --- | --- |
| **No** | **Smoke testing** | **Sanity testing** |
| **1** | Smoke Testing is performed to ascertain that the critical functionalities of the Sanity Testing is done to check the new functionality / bugs have been fixed program is working fine | Sanity Testing is done to check the new functionality / bugs have been fixed |
| **2** | The objective of this testing is to verify stability of the system in order to with more rigorous testing | The objective of this testing is to verify the rationality of system in order proceed to proceed with more rigorous testing |
| **3** | This testing is performed by the developers or testers | Sanity testing is usually performed by testers |
| **4** | Smoke testing is usually documented or scripted is unscripted | Sanity testing is usually not documented. |
| **5** | Smoke testing is a subset of Regression testing | Sanity testing is a subset of Acceptance testing |

**25. Difference between verification and Validation**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Verificaton** | **Validation** |
| **Definaton** | The process of evaluating work-products (not the actual f inal product) of a development phase to determine whether they meet the specified requirements for that phase. | The process of evaluating software during or at the end of the development process to determine whether it satisfies specified business requirements |
| **Objective** | To ensure that the product is being built according to the requirements and design specifications. In other words, to ensure that work products meet their specified requirements. | To ensure that the product actually meets the user’s needs, and that the specifications were correct in the first place. In other words, to demonstrate that the product fulfills its intended use |
| **Question** | Are we building the product right? | Are we building the right product? |
| **Evaluation**  **Items** | Plans, Requirement Specs, Design Specs, Code, Test Cases | The actual product/software. |
| **Activities** | * Reviews * Walkthroughts * Inspections | * Testing |

**26. Explain types of Performance testing.**

* Load Testing
* Stress Testing
* Endurance Testing
* Spike Testing
* Valume Testing
* Scalability Testing

**27. What is Error, Defect, Bug and failure?**

* **Error:-** A discrepancy between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition. This can be a misunderstanding of the internal state of the software, an oversight in terms of memory management, confusion about the proper way to calculate a value, etc.
* **Defect:-** Commonly refers to several troubles with the software products, with its external behavior or with its internal features
* **Bug:-** A fault in a program which causes the program to perform in an unintended or unanticipated manner. See: anomaly, defect, error, exception, and fault. Bug is terminology of Tester
* **Failure:-** The inability of a system or component to perform its required functions within specified performance requirements. See: bug, crash, exception, and fault.

**28. Difference between Priority and Severity**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Priority** | **Severity** |
| **Defination** | Priority is a parameter to decide the order in which defects should be fixed. | Severity is a parameter to denote the impact of a particular defect on the software. |
| **Purpose** | Priority means how fast the defect has to be fixed. | Severity means how severe the defect is affecting the functionality. |
| **Value** | Its value is subjective. | Its value is objective. |
| **Categories** | Priority is divided into 3 categories:   1. Low 2. Medium 3. High | Severity is divided into 4 categories:   1. Critical 2. Major 3. Medium 4. Low |

**29. What is Bug Life Cycle?**

* As you can see from above diagram, a defect‘s state can be divided into Open or Closed.
* When a bug reaches one of the Closed or Terminal states, its lifecycle ends. Each state has one or more valid states to move to.
* In a typical scenario, as soon as a bug is identified, it is logged into the bug tracking system with status as Submitted. After ascertaining the validity of the defect, it is given the “Open” Status.

**30.Explain the difference between Functional testing and NonFunctional testing**

|  |  |
| --- | --- |
| **Fuctional** | **Nonfuctional** |
| Functional testing is performed using the functional specification provided by the client and verifies the system against function requiremants. | Nonfuctional testing checks the Performance, reliability, scalability and other non-functional aspects of the software system. |
| Functional testing is executed first | Non functional testing should be performed after functional testing |
| Manual testing or automation tools can be used for functional testing | Using tools will be effective for this testing |
| Business requirements are the inputs to functional testing | Performance parameters like speed , scalability are inputs to non fuctional testing. |
| Easy to do manual testing | Tough to do manual testing |

**31. To create HLR& Test cases of instsgram and facebook only first page.**

<https://1drv.ms/x/c/b6a791daab688a94/Eb_hns7EP4lNmQexHQ5fCMgBXESFB4gMc3aa7QoQVCIGXg?e=PoheYv>

**32. What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development Life Cycle)?**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **SDLC** | **STLC** |
| **Domain** | SDLC is mainly related to software development | STLC is mainly related to software testing. |
| **Focus** | Besides development other phases like testing is also included. | It focuses only on testing the software. |
| **Phases** | SDLC involves total six phases or steps. | STLC involves only five phases or steps. |
| **Number of members** | In SDLC, more number of members (developers) are required for the whole process. | In STLC, less number of members (testers) are needed. |
| **objective** | In STLC, less number of members (testers) are needed. | Goal of STLC is to complete successful testing of software. |

**33. What is the difference between test scenarios, test cases, and test script?**

|  |  |  |
| --- | --- | --- |
| **Test scenarios** | **Test cases** | **Test script** |
| A test scenario is a high-level document that describes end-to-end functionality to be tested. | Test cases contain definite test steps, data, and expected outcomes for testing all the features of the software under test. | Test Script is set of instructions or a short program to test any functionality of software application/product. |
| Test scenarios are focused on what to test. | Test Case is a manual approach of software testing | Test Script is an automatic approach of software testing. |
| It will take less time as compared to test cases. | It takes more time in comparison to trying circumstances. | Test Script are characterized as manual test script and automation test scripts. |
| Requires less time for test scenarios | Requires more resources and time. | Requires less time for testing scripts. |
|  |  |  |

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

1. **Test plan:-** In [software testing](https://www.geeksforgeeks.org/software-testing-basics/), documentation is very important. Testing should be documented to provide efficient resource control monitoring. For successful testing, a test plan plays a very important role.
2. **Information:**-

Test strategy

Objectives

Test schedule

Required resources (human resources, software, and hardware)

Test estimation

Test deliverables

Test approach

**35. What is priority?**

Priority is Relative and Business-Focused. Priority defines the order in which we should resolve a defect. Should we fix it now, or can it wait? This priority status is set by the tester to the developer mentioning the time frame to f ix the defect. If high priority is mentioned then the developer has to fix it at the earliest. The priority status is set based on the customer requirements.

**36. What is severity?**

Severity is absolute and Customer-Focused. It is the extent to which the defect can affect the software. In other words it defines the impact that a given defect has on the system.

**37. Bug categories are…**

1. Functional bugs
2. Logical bugs
3. Workflow bugs
4. Unit level bugs
5. System level integration bugs
6. Out of bonds bugs
7. Security bugs

**38. Advantage of Bugzila .**

* Key features of Bugzilla includes
* Advanced search capabilities
* E-mail Notifications
* Modify/file Bugs by e-mail
* Time tracking
* Strong security
* Customization
* Localizatio

**39. Difference between priority and severity**

|  |  |  |
| --- | --- | --- |
| **Features** | **Priority** | **Severity** |
| **Definition** | Difference between priority and severity | Severity is a parameter to denote the impact of a particular defect on the software. |
| **Purpose** | Priority means how fast the defect has to be fixed. | Severity means how severe the defect is affecting the functionality. |
| **Relation** | Priority is related to scheduling to resolve the problem. | Severity is related to the quality standard. |
| **Categories** | Priority is divided into 3 categories:   * Low * Medium * High | Severity is divided into 4 categories:   * Critical * Major * Medium * Low |
| **value** | Its value is subjective. | Its value is objective. |

**40. What are the different Methodologies in Agile Development Model?**

* Scrum
* Lean
* Kanban
* Crystal
* XP (Extreme Programming)
* DSDM (Dynamic Software Development Method)
* FDD (Feature Driven Development)

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

|  |  |
| --- | --- |
| **Authentication** | **Authorization** |
| In the authentication process, the identity of users are checked for providing the access to the system. | While in authorization process, a the person’s or user’s authorities are checked for accessing the resources. |
| In the authentication process, users or persons are verified. | While in this process, users or persons are validated. |
| It is done before the authorization process. | While this process is done after the authentication process |
| It needs usually the user’s login details. | While it needs the user’s privilege or security levels |
| Authentication determines whether the person is user or not. | While it determines **What permission does the user have?** |

**What are the common problems faced in Web testing?**

* **Ceoss browser compatibility:-** Earlier, when internet explorer was the only browser available, just unit testing would have done the job. But, currently, with hundreds of browsers along with their different versions available for desktop and mobile, cross-browser compatibility is a common issue. It is ideal for a tester to use a cloud testing platform like lambdest for testing whether the application is compatible across different browsers.
* **Responsiveness:**- The one thing to look out for while testing is whether the application fits properly in the device resolution. A tester must check if there are any horizontal scrolling, alignment or padding issues, and sizes of font and buttons in different devices.
* **Cross device compatibility:**-

Nowadays, people mostly use mobile devices to access websites. Although there are a limited number of devices in iOS, the count increases tenfold when it comes to Android. It is important for a tester to target the devices where the application is meant to run and start testing in each of them.

**42. To create HLR & TestCase of WebBased (WhatsApp web , Instagram) 1. WhatsApp Web**

<https://1drv.ms/x/c/b6a791daab688a94/ESJTzJyS6QdFjTpvCjsuHigBxrnrHkfwRTRvumNgHYBFfA?e=8DBGTL>

**43. All tets scenario:-**

**ANS:-**

[**https://1drv.ms/x/c/b6a791daab688a94/ESW4Kr8QAilDul-fMEIO5Z4BlOuEet06A4W006wLfTFRsw?e=0ugNIe**](https://1drv.ms/x/c/b6a791daab688a94/ESW4Kr8QAilDul-fMEIO5Z4BlOuEet06A4W006wLfTFRsw?e=0ugNIe)