SECTION - A

1) _____ leads to deviation from the expected results.

Error

2) SDLC stands for : Software Development Lifecycle

3) The primary goal of ____ testing to ensure that the entire software system meets specified requirements.

System

4) TQM stands for : Total Quality Management

5) Give any two verification method in software development.

- Codereview
- Code walkthroughs

6) _____ testing involves deliberately providing invalid, unexpected, or erroneous inputs to evaluate how well the the software can handle unexpected conditions.

Negative testing

7) _____ automated testing tool is used to test applications, measuring system behaviour, and performance under load.

performance

8) whitebox testing is classified as Static testing and _____ testing

Structural

9) _____ Coverage refers to writing test cases that execute each of the program statements.

Statement

10) Equivalence Partitioning comes in _____ testing

Blackbox.

SECTION-B

11) What is Unit testing?

Unit testing is a type of software testing where individual components or functions of a program are tested in isolation to ensure they work as intended. It focuses on verifying the correctness of small, isolated place of code, typically at the function or method level.

12) What is code review?

code review refers to a process where developers systematically examine and evaluate each other's source code to identify potential bugs, improve code quality and ensure adherence to coding standards.

13) What is Usability testing?

Usability testing is a software testing method that evaluates how easy it is for users to interact with a product.

14, what do you mean by quality assurance?

Quality assurance to a systematic process that ensures a software product meets specified quality standards and customer expectations by monitoring all stages of development, identifying potential issues early on, and implementing measures to prevent defects, ultimately delivering a reliable and functional software application to the user.

15) what is Six Sigma.

Six sigma is a set of methodologies and tools used to improve business process by reducing defects and errors, minimizing variation and increasing quality and efficiency

16) Define Debugging.

Debugging is the process of finding and fixing errors or bugs in the source code of any software, when software doesn't work as expected, computer programmer study the code to determine why any errors occured.

17) Differentiate structural testing and static testing.

Structural testing

Structural testing involves examining the internal code structure to find defects based on how the code is written.

Static testing

Static testing analyzes software artifeat like code, documents, and requirements without actually executing the code.

Techniques : Code review
Code walk-through
Static analysis tools.

18) Mention the four types of coverages in code coverage testing.

- Statement Coverage
- Branch Coverage
- Condition Coverage
- path Coverage

19) What are the coloumns in an equivalence partition table.

A equivalence partition table in software testing usually includes coloumns for ; Testcase Id, Equivalence class description, Input data, Expected output and sometimes an additional coloumn for the Actual output.

20) Mention any two situations where state based testing is useful.

- Finite state Machines
- user authentication systems.

21) What you mean by test Management ?

Test Management in Software testing refers to the structured process of planning, organizing, executing, Monitoring and reporting on all testing activities throughout the software development life cycle, ensuring that the testing process is efficient, effective and aligned with project goals to deliver high quality software by identifying and managing.

22) what is test plan.

Test plan is software testing is a document that outputlines that strategy resources and schedule for testing a software application.

SECTION-C

23) Explain the purpose of software testing.

The purpose of software testing is to ensure that software application works as expected and meets its requirements. It helps identify effects or bugs in the system, ensuring the software's functionality, reliability and performance. Testing also verifies the softwar compatiability with different environments and ensure that meets security standards ultimately, it helps deliver a high-quality product, providing confidence to users and stakeholders that the software is safe

24) Explain top-down testing techniques with its advantages and disadvantags.

Top-down testing is a software testing techniques where testing starts from the top-level modules or components of the system and gradually moves down to the lower-level modules. The higher-level modules that are yet to be developed.

Advantages

- Early detection of major design issues: Since testing begins at the top, it helps identify high-level design or architectural flaws early in the development cycle.

Disadvantages

- Incomplete testing of lower-level modules: The lower-level modules may be simulated with stubs or not fully tested until later, which could lead to delays or missed bugs in those components.

25) Explain the advantage and disadvantages of Blackbox testing

Advantages

- focuses on functionality: Blackbox testing examines the software from the user's perpective, ensuring that it behaves as expected without needing knowledge of the internal code.

• Unbiased testing : Testers don't need to be familiar with the implementation, so it eliminates the risk of tester bias, ensuring that all parts of the software are tested based purely on inputs and outputs.

Disadvantages

• Limited Coverage : Since testers don't have access to the internal structure of the system. certain edge cases or paths may be overlooked, leading to Incomplete testing.

26) write notes on regression testing

Regression testing is the process of re-testing a software application after changes such as bug fixes, enchancement or new features additions to ensure that the existing functionality is not affected. The goal is to be identify any defects introduced by these changes and confirm that previously working parts of the system still function as intended.

Advantages

• Ensure that Software remains stable after modifications.
• Helps maintain high-quality standards as the software evolves.

Disadvantage

• Can be time-consuming, especially for large applications.
• Requires frequent updates to test cases as applications changes.

27) Explain the General characteristics of software testing.

① purpose activity : software testing aims to identify defects, verify functionality and ensure that the software meets its requirements and user expectations

② systematic process : Testing follows a struct approach, involving test planning, executio and documentation to ensure through coverage and traceability.

③ Defect Identification : The primary goal is to uncover errors, inconsistencies, or issue within the software to improve quality and reliability.

④ verification and validation : Testing ensurin the software works as ~~coded~~ intended and fullfills the specified requirements and user needs (validations)

28) write notes on boundary value analysis

Bandary value analysis is a software testing techniques used to identify errors of the boundaries of input ranges, rather than focusing solely on the center It is based on the principle that defects are more likely to occur at the edges of input values.

Advantages

- effective in finding boundary - related errors, Improving test coverage with fewer test cases

Disadvantages

- It may not cover errors related to data within the valid range, as it mainly focuses on boundary conditions

29) Explain user documentation testing?

User documentation testing in software testing involves verifying the accuracy clarity, and completeness of the user manuals, help guides and other documentation associated with a software product. The goal is to ensure and effectively use the software by referring to the the documentation.

1. Accuracy: Ensuring the documentation correctly reflects the software's functionality and features.
2. clarity: verifying that the language and instructions are clear simple and easy to follow for the target audience.
3. Completness: Confirming that all necessary information, procedure and use cases included in the documentation.
4. Consistency: Ensuring terminology, formatting and style are consistent throughout the documents.

Effective user documentation testing helps users to quickly get up to speed with the software and minimize the likelihood of errors caused by misunderstanding or lack of information.

30) Explain the tasks to be covered in test automation.

Test automation in software testing involves automating repetitive tasks to improve efficiency and accuracy in the testing process. The tasks typically covered in test automation include.

① Regression testing: Automating the execution of test cases to ensure that new code changes to not break existing functionality.

② Smoke testing: Automating the Intial test to quickly check whether the build is stable enough for more detailed testing.

③ Load and performance testing: Automating test to evaluates how the System perform under varying levels of load and stress.

④ Respetitive functional testing: Automating tests for frequently used feature to ensure they continue to work as expected after updates or changes.

31) Describe condition coverage in whitebox testing.

Condition coverage in white-box testing is a type of code coverage criterian that ensures each individual condition in a decision like an "if" or "while" statement is evaluated to both true and false atleast once during testing.

Consider the following code

```
Java
if(a>0 && b<a)
{
  // do Something
}
```

To achieve condition coverage, you would need to test:

- a>0 is true, b<10 is true.
- a>0, is true, b<10 is false
- a>0, is false, b<10 is true
- a>0, is false, b<10 is false

Condition Coverage ensures that each condition in the code is independently verified for both true and false, outcomes improving test thoroughness.

SECTION D

32) Explain various approaches to Software testing?

Software testing are classified into two.

① Manual testing

Testing any software or an application according to the clients need without using any automation tool.

② Automation testing

Automation testing is a software testing techn where test cases are executed automatically using special software tool.

- whitebox testing

whitebox testing also called "clearbox testing is a type of software testing where the tester knowns the internal wooking of the application.

- Blackbox testing

Blackbox testing is a software testing techniques where the tester focus on testing the functionality of the application without knowing its internal code.

- Greybox testing

Greybox testing is a type of software testing techniques that combines elements of both. whitebox and black box testing, where the tester has partial knowledge of the internal working of the application while focusing on its functionality.

functional testing

Functional testing is a type of software testing that evaluates whether the software functions as expected, based on its requirements and specifications without considering the internal code.

② Adhoc testing

Testing the application whithout any plan, test case or documents.

Security testing

It is the path of non-functional testing it is used to test the security of the application like authentication, authorization and confidentiality.

- Accessability testing

It is the part of usability testing. testing the application to check how user friendly the application is for disable people.

- Localisation testing

It is the part of non-functional testing testing the localised variation of the application for its local languages local culture and local settings.

- performance testing

performance testing is the process of accessing the speed, responsive and stability of a system under load to ensure it meets specified performance

- Usability testing

Usability is the process of evaluating a product or system by testing it with real users to ensure it is easy to use, efficient and provides a satisfactory user experience.

pair testing

It is the path of pair testing, testing is done by two testers is one system without any testcase or documents.

② Negative testing

Testing the application with negative input condition

③ Exploratory testing

It is simultaneous process of testing planning testcase preparation test case execution and Bug reporting all done at same time

④ Smoke testing

Testing the major functionalities each release to ensure the stability of the application this testing is done at the first stage.

Non - functional testing

Non - functional testing is a type of software testing that evaluates the performance usability, security and other non - functional aspects of the software.

- Database testing

Testing the database using sql for creating, listing, updating, deleting particular data in database.

33) Discuss several methods to achieve static testing by humans.

Static testing is the method of software testing where the code or documentation is reviewed without executing the program. The goal is to find defects early in the development lifecycle by reviewing, inspecting or analyzing artifacts such as source code, requirements, design documents and specifications.

1. Code Reviews

code review are a type of static testing where peers or team members review each other's code to find defects, ensure adherence to coding standards and improve overall quality.

2. walkthrough

A walkthrough is an informal review process where the author of a document, code or design leads a group of reviews through the material. The goal is to understand the content and identify potential issue or improvements.

3. Inspections

Inspections are formal and highly structured static testing methods where team systematically examines the code or documentation. A formal checklist is often used, and the goal is to defect compilance issues, and inconsistencies. Inspections are typically led by a trained moderator

4. Static Analysis

Static Analysis refers to using automated tool to examine the source code for potential defects, security vulnerabilities and adherence to coding standards. Though tool performance the analysis, human involvements is still required to interpet the results and make decisions based on the finding.

5. Style checking

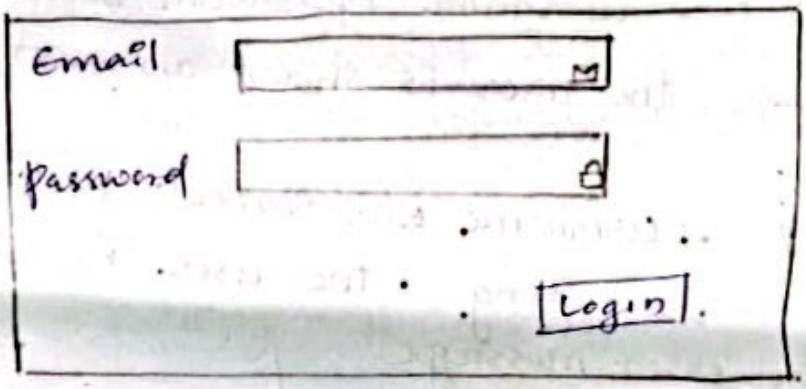
Style checking involves reviewing the software requirements documentation to ensure that they complete clear, consistent and feasible. this method helps identify any ambigu or inconsistane.

34) Decision table

- It is one of the techniques on black box testing
- Decision table testing is a software testing technique used to test system behaviour for different input Componation.
- This is a systematic approach where the different input combination and their corrospond system behaviour. (output) are captured on a tabular form. It is also called as a cause - Effect table.

- A decision table is a tabular representation of inputs versus rules / cases / test conditions.
- A decision table helps to check all possible combinations of conditions for testing and testers can also identify missed conditions easily. The condition are indicated as True (T) and false (F) values.

Ex: Decision table for a login screen



- The condition is simple if the user provide the correct username and password the user will be redirected to the homepage. If any of the input is wrong, an error message will be displayed.

Conditions	rule 1	Rule 2	Rule 3	Rule 4
username(T/F)	F	T	T	F
Password (T/F)	F	T	F	T
Output (E/H)	E	H	E	E

T - Correct username / password

F - wrong username / password

H - Home screen displayed

E - Error message displayed

Interpretation

1) case 1 - username & password both were wrong the user is shown an error message

2) case 2 - username was correct but password was wrong. the user is shown an error message.

3) case 3 - username was wrong but the password was correct. the user is shown an error message.

4) case 4 - username and password were both correct and the user is navigated to the homepage.

35) Software testing is a systematic process designed to evaluate and validate the functional performance and reliability of a software application. The test process ensure that the software meets the specified requirement and behave as expected under various condition. The process typically involves multiple stages from planning and design to execution and closure.

1. Requirement Analysis

The first step is testing process is analysing the requirements to understand the software functionality the business needs, and the testing objectivies. This phase involves reviewing the project requirements user stories specification and user case.

2. Test planning

Test planning is the next phase, where the testing strategy and approach are defined. It involves setting objectivies, determing the resource needed defining the scope and establishing schedules the test plan also outlines the test environment and tools required.

3. Test design

In the test design phase, the test cases are created based on the requirement and test plan. This step focuses on designing

the actual test that will be executed to validate the software functionality.

4) Test Environment Setup

This phase involves setting up the test environment where the test will be executed. It ensures that the software is tested in a stable and conditions that mimics real-world conditions.

5) Test execution

Test execution involves running the designed test cases in the test environment to validate the software functionality during execution, testers compared actual results with expected results to identify defects.

6) Defect reporting and Tracking

During the test execution phase, testcase often discover defects which are reported and tracked for resolution. This involves logging, through to resolution.

7) Test Reporting

After executing the test and tracking defect a test report is generated to communicate the outcomes to stakeholders. The test rep summarizes the testing activities, result a overall quality of the software.

8) Test closure

The test closure phase is the final step in the testing process. It involves completing the test activities, evaluating the overall test process and preparing the final deliverables.

9) Regression Testing

After the development team has fixed the defects regression testing is conducted to ensure that the changes made do not affect the existing functionality of the software. it is important to test the modified areas and any affected parts of the appli. to confirm that no new defect introduced.