**Assignment of software testing**

1. **What is sdlc**

**Ans-SDLC stands for Software Development Life Cycle. It is a structured process used by software developers and project teams to design, develop, test, and deploy high-quality software systems efficiently.**

1. **What is software testing?**

**Ans-Software testing is the process of evaluating a software application or system to identify defects, ensure quality, and verify that it meets the specified requirements.**

1. **What is srs?**

**Ans-SRS stands for Software Requirements Specification.**

**It is a detailed document that describes what a software system should do, how it should behave, and the constraints it must operate under. It serves as a bridge between the client/stakeholders and the development team.**

1. **What is oops?**

**Ans-OOPs stands for Object-Oriented Programming System. It is a programming paradigm based on the concept of "objects", which can contain data (fields or attributes) and code (methods or functions).**

1. **Write basic concepts of oops**

**Ans- Class,Object,Encapsulation, Inheritance,Polymorphism ,Abstraction**

1. **What is object?**

**Ans-An object is a real-world entity or instance of a class in object-oriented programming. It contains both data (attributes) and behavior (methods/functions) that operate on the data.**

1. **What is class?**

**Ans-In Object-Oriented Programming (OOPs), a class is a blueprint or template for creating objects. It defines the structure and behavior that the objects (instances) will have.**

1. **What is encapsulation?**

**Ans-Encapsulation is one of the four core principles of Object-Oriented Programming (OOPs). It refers to the binding of data (variables) and the methods (functions) that operate on that data into a single unit, typically a class, and restricting direct access to some of the object's components.**

1. **What is inheritance?**

**Ans-Inheritance is a fundamental concept in Object-Oriented Programming where a new class (child or subclass) derives properties and behaviors (attributes and methods) from an existing class (parent or superclass).**

**10)What is polymorphism?**

**Ans-Polymorphism means "many forms". In Object-Oriented Programming, it allows methods or functions to behave differently based on the object or context that calls them.**

**11) Write sdlc phases with basic introduction**

**Ans-SDLC stands for Software Development Life Cycle. It is a systematic process used by software developers to design, develop, test, and deliver high-quality software efficiently. SDLC helps ensure that the software meets user requirements and is delivered on time and within budget.**

**🛠️ Phases of SDLC:**

1. **Planning**
   * **Define the project scope, goals, and feasibility.**
   * **Allocate resources and set timelines.**
2. **Requirements Analysis**
   * **Gather and document the detailed functional and non-functional requirements.**
   * **Engage stakeholders to understand their needs.**
3. **Design**
   * **Create the system architecture and detailed design based on requirements.**
   * **Includes UI/UX, database, and system design.**
4. **Implementation (Coding)**
   * **Write the actual code according to the design documents.**
   * **Follow coding standards and best practices.**
5. **Testing**
   * **Verify and validate the software to find and fix defects.**
   * **Types of testing include unit, integration, system, and acceptance testing.**
6. **Deployment**
   * **Release the software to the production environment.**
   * **Prepare users and provide documentation and training.**
7. **Maintenance**
   * **Provide ongoing support, fix bugs, and implement enhancements.**
   * **Ensure the software remains useful over time.**

**12) What is Integration testing?**

**Ans-Integration Testing is a software testing phase where individual modules or components are combined and tested together as a group. The goal is to verify that these modules work correctly when integrated and communicate properly with each other.**

**13) What is Alpha testing?**

**Ans-Alpha Testing is a type of internal acceptance testing performed by the software development team (developers and QA) before the product is released to external users or clients.**

**14) What is beta testing?**

**Ans-Beta Testing is a type of external user acceptance testing where the software is released to a limited number of real users outside the development team to use and evaluate in a real-world environment before the final release.**

**15) What is GUI Testing?**

**Ans-GUI Testing (Graphical User Interface Testing) is a type of software testing that focuses on testing the visual elements of an application — such as buttons, menus, forms, icons, and other interactive elements — to ensure they function correctly and look as expected.**

**16) What is load testing?**

**Ans-Load Testing is a type of performance testing that checks how a software application behaves under a specific expected load — such as a number of users, transactions, or data volume.**

**17) What is stress Testing?**

**Ans-Stress Testing is a type of performance testing that evaluates how a software system behaves under extreme or beyond-normal conditions — such as heavy user traffic, limited resources, or peak data load — to determine its stability, reliability, and failure recovery.**

**18) What is white box testing**

**Ans-White Box Testing (also known as Clear Box Testing, Glass Box Testing, or Structural Testing) is a software testing technique where the tester has full knowledge of the internal code, structure, and logic of the application.**

**19) What is black box testing?**

**Ans-Black Box Testing is a software testing technique where the tester does not know the internal structure or code of the application. Instead, testing is done by providing inputs and observing outputs to verify the system behaves as expected.**

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

**Ans-Exit Criteria in software testing define the conditions that must be met before testing can be considered complete. They help ensure that the product has been tested thoroughly and is ready for release or the next phase of development.**

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

**Ans-Regression Testing should be performed whenever changes are made to the software, to ensure that existing functionality still works and no new bugs have been introduced.**

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

**Ans- 1. Testing Shows Presence of Defects**

**Explanation:**

* **Testing can reveal the presence of bugs, but it can’t prove their absence.**
* **Even after extensive testing, you can’t be 100% sure the software is bug-free.**
* **The goal is to reduce the risk of defects, not eliminate it completely.**

**2. Exhaustive Testing is Impossible**

**Explanation:**

* **You cannot test all possible combinations of inputs, paths, and data due to time and cost limitations.**
* **Instead, risk-based testing and prioritization are used to focus on the most critical areas.**

**3. Early Testing Saves Time and Money**

**Explanation:**

* **Testing should start early in the software development life cycle (SDLC), ideally from the requirement or design phase.**
* **The earlier a defect is found, the cheaper it is to fix.**

**4. Defect Clustering**

**Explanation:**

* **Most defects are usually found in a small number of modules.**

**5. Pesticide Paradox**

**Explanation:**

* **Running the same set of test cases repeatedly will no longer find new bugs.**
* **Test cases need to be regularly reviewed and updated to find fresh issues.**

**6. Testing is Context Dependent**

**Explanation:**

* **The type and approach to testing depend on the context of the application.**
* **Safety-critical software (e.g., for airplanes) needs more rigorous testing than a mobile game.**

**7. Absence-of-Errors Fallacy**

**Explanation:**

* **Just because the software is bug-free, doesn’t mean it’s useful or correct.**
* **A product that meets technical requirements but fails to meet user needs is still a failed product.**

**23) What is functional system testing?**

**Ans- Functional System Testing is a type of black-box testing that verifies whether the entire system behaves according to its functional requirements. It is performed after integration testing and focuses on testing the complete and integrated software system.**

**24) What is Non-Functional Testing?**

**Ans- Non-Functional Testing is a type of software testing that checks *how well* the system performs, rather than what it does. It evaluates the quality attributes of a software application like performance, usability, reliability, security, and scalability.**

**25) Difference between QA v/s QC v/s Tester**

**Ans-Quality Assurance is process oriented.Quality Assurance makes test plan,review test cases and guide tester and quality control engineer.**

**Quality control is product oriented.Quality control engineer makes test cases ,bug report and finds bugs.**

**26) Difference between Smoke and Sanity?**

**Ans- Smoke Testing:**

**Smoke testing is often referred to as a "build verification test." It is performed after a new build is deployed to verify that the critical functionalities of the application are working correctly.**

**Sanity Testing:**

**Sanity testing, on the other hand, is a focused testing approach. It is usually performed after receiving a minor code change or bug fix to verify that the specific functionality or issue has been resolved, and that no related functionality is broken.**

**27) Explain types of Performance testing.**

**Ans- 1. Load Testing**

* **Purpose: To check how the system behaves under expected load conditions.**

**2. Stress Testing**

* **Purpose: To determine the system’s breaking point by pushing it beyond normal limits.**

**3. Volume Testing (Flood Testing)**

* **Purpose: To test the system’s ability to handle large volumes of data.**

**28) Explain the difference between Functional testing and NonFunctional testing**

**Ans- What is Functional Testing?**

* **Focus: What the system does (its features and functions).**
* **Purpose: To check if the application works correctly based on requirements.**
* **Tested by: QA/Testers using test cases.**
* **Type: Black-box testing (no need to know the code).**

**What is Non-Functional Testing?**

* **Focus: How the system performs (speed, usability, reliability, etc.).**
* **Purpose: To check the quality of the system under different conditions.**
* **Tested by: Performance testers, security testers, or automation engineers.**
* **Type: Also black-box testing, but focused on behavior and performance.**

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

**Ans-**  **SDLC is the process of developing software, from planning to delivery and maintenance.**

**It covers everything: requirement gathering, design, coding, testing, deployment, and support.**

**STLC is the process of testing the software to make sure it works correctly and meets requirements.**

**It is a part of SDLC, focused only on testing activities.**

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

**Ans-** **A Test Plan is a formal document that outlines the strategy, scope, objectives, resources, schedule, and approach for testing a software application.**

**1. ✅ Test Plan ID**

* **A unique identifier for the test plan (e.g., TP-001)**

**2. 🖥️ Scope of Testing**

* **In-Scope: What will be tested (e.g., login, registration)**
* **Out-of-Scope: What won’t be tested (e.g., payment module)**

**3. 🧾 Entry and Exit Criteria**

* **Entry: Conditions required to begin testing (e.g., code freeze, test data ready)**
* **Exit: Conditions to end testing (e.g., all critical bugs fixed)**

**31) What is join?**

**Ans-** **A JOIN is used in SQL to combine data from two or more tables based on a related column (like a common ID).**

**32) Write type of joins.**

**Ans-**

| **JOIN Type** | **Description** |
| --- | --- |
| **INNER JOIN** | **Returns records that have matching values in both tables** |
| **LEFT JOIN** | **Returns all records from the left table + matched from right** |
| **RIGHT JOIN** | **Returns all records from the right table + matched from left** |
| **FULL JOIN** | **Returns all records when there's a match in either table** |

**33) What is API Testing**

**Ans-** **API Testing checks whether the data sent and received between different software systems is correct, secure, and efficient — without using a GUI.**

**34) What is Responsive Testing?**

**Ans- Responsive Testing ensures that your website automatically adjusts and remains usable on all devices, no matter what size the screen is.**

**35) Write a query to create the table in Structured Query Language.**

**Ans-** **CREATE TABLE Employees (**

**EmployeeID INT AUTO\_INCREMENT PRIMARY KEY,**

**FirstName VARCHAR(50) NOT NULL,**

**LastName VARCHAR(50) NOT NULL,**

**36) Write a query to insert data into table.**

**Ans-** **INSERT INTO Employees (FirstName, LastName)**

**VALUES**

**('Alice', 'Smith'),**

**('Bob', 'Johnson'),**

**('Charlie', 'Brown');**

**37) Create two tables named Seller and Product apply foreign key in product table Fetch data from both table using different joins.**

**Ans-** **-- Creating Seller Table**

**CREATE TABLE Seller (**

**SellerID INT AUTO\_INCREMENT PRIMARY KEY,**

**SellerName VARCHAR(100) NOT NULL,**

**City VARCHAR(50)**

**);**

**-- Creating Product Table with Foreign Key referencing Seller**

**CREATE TABLE Product (**

**ProductID INT AUTO\_INCREMENT PRIMARY KEY,**

**ProductName VARCHAR(100) NOT NULL,**

**Price DECIMAL(10, 2),**

**SellerID INT,**

**FOREIGN KEY (SellerID) REFERENCES Seller(SellerID)**

**);**

**-- Insert data into Seller**

**INSERT INTO Seller (SellerName, City) VALUES**

**('Amazon', 'Seattle'),**

**('Flipkart', 'Bangalore'),**

**('Meesho', 'Delhi');**

**-- Insert data into Product**

**INSERT INTO Product (ProductName, Price, SellerID) VALUES**

**('Laptop', 55000.00, 1),**

**('Mobile', 15000.00, 1),**

**('Shoes', 2000.00, 2),**

**('T-shirt', 500.00, 3),**

**('Bag', 1200.00, NULL); -- product with no seller**

**SELECT Product.ProductName, Product.Price, Seller.SellerName**

**FROM Product**

**INNER JOIN Seller ON Product.SellerID = Seller.SellerID;**

**SELECT Product.ProductName, Product.Price, Seller.SellerName**

**FROM Product**

**LEFT JOIN Seller ON Product.SellerID = Seller.SellerID;**

**SELECT Product.ProductName, Product.Price, Seller.SellerName**

**FROM Product**

**RIGHT JOIN Seller ON Product.SellerID = Seller.SellerID;**