**Module-3** (Testing on Live Application)

**1)What is RDBMS ?**

* **Relational database management system**. While a relational database organizes data based off a relational data model, a relational database management system (RDBMS) is a more specific reference to the underlying database software that enables users to maintain it.

**2) What is SQL**?

* Structured Query Language SQL is **a database tool that is used to create and access the database to support software applications**.

**3) Write SQL Commands**

* SELECT - extracts data from a database.
* UPDATE - updates data in a database.
* DELETE - deletes data from a database.
* INSERT INTO - inserts new data into a database.
* CREATE DATABASE - creates a new database.
* ALTER DATABASE - modifies a database.
* CREATE TABLE - creates a new table.

**4) What is join?**

* JOIN is **an SQL clause used to query and access data from multiple tables, based on logical relationships between those tables**. In other words, JOINS indicate how SQL Server should use data from one table to select the rows from another table.

**5) Write types of join?**

* Four types of joins: **left, right, inner, and outer**.
* Cross join. A cross join returns all possible combinations of rows of two tables (also called a Cartesian product).
* Join/inner join. An inner join, also known as a simple join, returns rows from joined tables that have matching rows. ...
* Left outer join/left join.
* Right outer join/right join.
* Full outer join.

**6) How many constraints and describe ?**

* NOT NULL Constraint.
* UNIQUE Constraint.
* DEFAULT Constraint.
* CHECK Constraint.
* PRIMARY KEY Constraint.
* FOREIGN KEY Constraint.

**7) Difference between DBMS and RDBMS?**

* In DBMS, data stores either in either a navigational or hierarchical form. RDBMS uses tabular structures to store data. In table headers are the column names and the rows contains corresponding values. NoSQL uses to store data in structured, semi-structured and unstructured forms.

**8) What is API Testing**

* API testing is **a type of software testing that analyzes an application program interface (API) to verify that it fulfills its expected functionality, security, performance and reliability**. The tests are performed either directly on the API or as part of integration testing.

**9) Types of API Testing**

**Some of the common types of API Testing include:**

[**Functional Testing**](https://www.zucisystems.com/software-testing/functional-testing/)

This testing is a broad-level examination of the specific functions of the program. It evaluates the responses in terms of accuracy of output, whether it lies within expected parameters, and how errors are handled. Apart from regular operational scenarios, Functional Testing of APIs also considers edge cases for boundary conditions.

**Security Testing**

Since the API provides access to all external applications to access the internals of the software product, it is usually considered the most exposed or vulnerable part of the system. Hence Security Testing is critical to ensure the safety of the system, as a single vulnerability or bug could jeopardize the entire operations of an enterprise. Penetration Testing and Fuzz/Noise Testing are subsets of Security Testing. While Penetration Testing, as the name suggests, attempts to breach an application, Fuzz/Noise Testing is a kind of legitimate DOS (Denial of Service) attempt to flood the system with digital noise, e.g., massive amounts of dummy data, and thereby check whether it leads to a system crash.

**Load Testing**

Load Testing validates whether the API operates under massive and/or sustained loads, e.g., by progressively increasing user requests from 1k to 10k and 100k and so on. Loads are typically baseline or regular loads, theoretical maximum loads, and overloads which exceed the theoretical maximum by 15-20%. Load Testing of APIs focuses on how these progressive loads are handled and the failure rate is measured for each load level.

**Runtime Error Detection**

While most tests are related to the implementation of the API and its functionalities, the Runtime Error Detection is concerned solely with the actual operation of the API. It examines the results of utilizing the API codebase and monitors the system for execution errors, memory leakage, and tests its error handling capabilities.

**User Interface Testing**

As may be expected, UI Testing evaluates the User Interface of the application. It is an indirect test of the API in the sense that it does not test the API directly but rather tests the UI that is connected with the API. It however gives testers an overview of the performance, efficiency, and usability of the system.

**Validation Testing**

This is usually performed at the very end of Software Development Life Cycle but it is one of the important tests to be conducted. Validation Testing examines such issues as the appropriateness of the API for the application, code bloat, and API behavior as well as accuracy, efficiency, and optimization level of the program.

Web APIs broadly fall into two established classes of web services: SOAP or Simple Object Access Protocol, REST or Representational State Transfer, and now, GraphQL, which is a far more recent development in the web services arena. Unlike SOAP, which uses XML, REST APIs are URL-based (using HTTP) and can provide output data in a variety of formats JSON, CSV, or RSS, among others.

Due to its flexibility, REST APIs are the logical choice for web services. They comprise four main access methods, viz., GET, POST, PUT, and DELETE.

[**REST API Testing**](https://www.zucisystems.com/blog/api-testing-with-rest-assured/) essentially checks the correctness of the HTTP status codes, verifies response headers and payload, examines the overall performance time, and occasionally also the application state. The common scenarios that are considered for REST API Testing include basic positive tests with and without optional parameters, negative tests with valid and invalid inputs, as well as security/authorization checks.

**10) What is responsive testing**

* Most responsive design tests start with visual testing. This testing **ensures that the UI of the app/website looks the same across devices and browsers**. For example, the colour of a banner on an Android app may look different across various Samsung mobile phones, possibly due to differences in colour codes.

**11) Which Types of tool are available for responsive testing?**

* Bootstrap. Bootstrap is a free, open source tool for building mobile-first responsive websites. ...
* Wirefy. Wirefy is a free tool for easily creating responsive wireframes. ...
* FitVids. ...
* Adaptive Images. ...
* FitText. ...
* Webflow. ...
* Invision. ...
* Blueberry.

**12) what is full form of IPA and APK**

* **iPA: iOS APP Store Package**. **APK: Android Application Package file**. exe: Executable File

**13) how to create step for open the developer option mode on**

1. Open the Settings app, scroll down to the bottom of the list and select About phone or About device. ...
2. Scroll down the About phone menu to find the Build Number.
3. Tap the Build Number seven times. ...
4. Enter your PIN or pattern to enable the Developer Options menu.