***TEST DATA MANAGEMENT:***

Test Data Management (TDM) is the process for providing controlled data access to modern teams throughout the Software Development Lifecycle (SDLC).

Test data management is a type of operation that ensures accurate data is ready for the testers to help them in their swanky work process. Test data management (TDM) accomplishes compatible test data properly and flawlessly. TDM provides and creates a brief automated test that can benefit users without any difficulty in the middle of their work.

**Why is TDM important?**

Modern Test Data Management solutions help organizations accelerate application development speed, code quality, data compliance, and sustainability initiatives by providing timely access to fresh relevant data downstream for code development, automated tests, troubleshooting, and validation.

**Features of Test Data Management:**

1. Provides required test data
2. Provides inbuilt data library
3. Allows sensitive data masking and encryption
4. Identifies data sources
5. Prepares data generation rules
6. Protection of sensitive data and many more.

**Importance of Test Data Management:**

* TDM Provides high-quality software that will work effectively.
* It will obstruct unusual bug fixes.
* Stored the primitive data successfully
* Decreases the risks of misplacement of the information.
* Provides test data to application members in an accurate time.
* Avoids higher costs.
* Maintains security for sensitive information.
* Keeps data always in the right position.
* Provides easy access to the testers.
* No extra steps for the test data by a lot of team members.

**What can Test Data Management tools do?**

Test Data Management involves the synchronization of multiple data sources from production, versioning copies, sensitive data discovery, compliance masking data, and multicloud distribution of test data to support agile development and automated testing.

**Some of the tools are:**

* Informatica
* LA Test Data Manager
* LISA Solutions
* Compuware
* Delphix
* Microfocus Data Express
* IBM InfoSphere Optim

**Challenges of TDM:**

* Proper availability of test data
* Maintaining data quality
* Providing proper storage to test data
* Lack of tools and skills for data extraction
* Identifying corrupted data at an earlier stage
* Lack of domain knowledge

**Solutions to Challenges of TDM:**

* Availing sufficient amount of required data before testing.
* Properly checking the quality of test data before using that.
* Providing sufficient storage for storing test data.
* Acquiring required domain knowledge.
* Knowing properly how to use different tools.
* Removing the corrupted data from earlier stage.
* Focusing on processing cost and storage and manage accordingly.

***TEST ENVIRONMENT MANAGEMENT:***

Test Environment Management deals with the maintenance and upkeep of the test bed.

List of activities by the Test environment management function include,

* Maintenance of a central repository with all the updated version of test environments.
* Test environment management as per the test team demands.
* As per the new requirements creating new environments
* Monitoring of the environments
* Updating/deleting outdated test-environments
* Investigation of issues on the environment
* Co-ordination till an issue resolution.

A testing environment is a setup of software and hardware for the testing teams to execute test cases. In other words, it supports test execution with hardware, software and network configured.

Test bed or test environment is configured as per the need of the Application Under Test. On a few occasion, test bed could be the combination of the test environment and the test data it operates.

Setting up a right test environment ensures software testing success. Any flaws in this process may lead to extra cost and time to the client.

**Challenges in setting up Test Environment Management:**

**1. Proper planning on resource usage:** Ineffective planning for resource usage can affect the actual output. Also, it may lead to conflict between teams.

**2. Remote environment:** It is possible that a Test environment is located geographically apart. In such a case, the testing team has to rely on the support team for various test assets. (Software, hardware, and other issues).

**3. Elaborate setup time:** Sometimes test set up gets too elaborated in cases of Integration Testing.

**4. Shared usage by teams:** If the testing environment is used by development & testing team simultaneously, test results will be corrupted.

**5. Complex test configuration:** Certain test requires complex test environment configuration. It may pose a challenge to the test team.