**Test case designing.**

**What is test case and test case designing?**

* A test case is a set of actions performed on a system to determine if it satisfies software requirements and functions correctly.
* The purpose of a test case is to determine if different features within a system are performing as expected also meets the customer requirements.

**Test case design consists of the below parameters,**

**Test cases - Login page**

**Test Case ID:** SIT\_TC001

**Test Scenario\Description:** To authenticate a successful user login on the Gmail.

**Test Steps:**

Step1 - The user navigates to Gmail.com.

Step2 - The user enters a registered email address in the ’email’ field.

Step3 - The user clicks the ‘Next’ button.

Step4 - The user enters the registered password.

Step5 - The user clicks ‘Sign In.’

**Prerequisites:** User should have a registered Gmail ID with a unique username and password.

**Test Data:** Username and password.

**Expected Results:** Once username and password are entered, the web page redirects to the user’s inbox, displaying and highlighting new emails at the top.

**Actual Results:** As Expected, User able to login to the application successfully.

**Test Status** – Pass/Fail: Pass

**What a good test case should Posses?**

* Test Case should be consistent, the testing flow or sequence should be good.
* Test Case should consist of both positive and negative scenarios.
* Test cases should be able to catch the defects.
* Test cases should not be redundant, means no duplicates.
* Test case should cover various scenarios to ensure end to end testing.
* A test case should provide a roadmap for verifying specific functionalities and business scenarios of the software.
* It highlights the exact steps, the data to be used, the actual and the expected results.

**Design Test – Objective**

**The main objective of test case design is to,**

* **Meets the entire business requirements so that the product is good.**
* **Covers all the functional, non-functional business scenarios.**
* **Early identification of errors\bugs before the product is released to users.**
* **To achieve the quality or the Bug free product**
* **Each test case should focus on a specific business scenario or functionality.**

**Design Test – Workflow**

* I**dentify the test objectives.**
* **Identify the test scenarios.**
* **Specify the test data and expected results.**
* **Create the test cases.**
* **Decide on the test schedule.**

**Test case design activities (Expected Result) –**

* **Based on the test execution, the actual and the expected results are verified.**
* **The expected results are the results as per the requirements.**

**Test case format major elements.**

**With the below formats, the test cases can be defined,**

* **Verify  
  Using [tool name]  
  With [conditions]  
  To [what is returned, shown, demonstrated]**
* **Verify: Used as the first word of the test statement.  
  Using: To identify what is being tested.**

**“To verify whether user is able to access\connect to the database using the user credentials”.**

**Guidelines for writing test cases**

* **Prioritize which test cases to write based on project timelines and the risk factors of the system or application.**
* **Be clear, concise, and assertive in describing what the tester needs to do and what results they should ideally get.**
* **Create unique test cases and avoid irrelevant or duplicate test cases.**
* **Confirm that the test suite checks all specified requirements mentioned in the specification document.**
* **Write test cases that are transparent and straightforward. The title of each test case should be short.**
* **Test case steps should be broken into the smallest possible segments to avoid confusion when executing.**
* **Test cases should be written in a way that allows others to easily understand them and modify the document when necessary.**
* **Keep the end user in mind whenever a test case is created.**
* **Do not assume the features and functionality of the system.**
* **Each test case should be easily identifiable.**
* **Descriptions should be clear and concise.**

|  |
| --- |
| **Topics covered** |
| **What is Performance Testing**  **Types of Performance Testing**  **What is Volume Testing**  **What is Security Testing**  **System Security**  **ETL Security**  **Lack of Awareness**  **Absence of Tools**  **Lack of Standard Approach/Methodology** |

**Performance Testing**

* **Performance Testing** is a software testing process used for testing the speed, response time, stability, reliability, scalability, and resource usage of a software application under a particular workload.

**Performance Testing Attributes:**

* **Speed:**   
  It determines whether the software product responds quickly.
* **Scalability:**   
  It determines the amount of load the software product can handle at a time.
* **Stability:**   
  It determines whether the software product is stable in case of varying workloads.
* **Reliability:**   
  It determines whether the software product is secure or not.

## Types of Performance Testing

Below are the types of performance testing,

**Load testing –**

* Checks the application’s ability to perform under anticipated user loads.
* The objective is to identify performance bottlenecks before the software application goes live.

[**Stress testing**](https://www.guru99.com/stress-testing-tutorial.html)**–**

* Involves testing an application under extreme workloads to see how it handles high traffic or data processing.
* The objective is to identify the breaking point of an application.

**Endurance testing –**

* This testing is done to make sure the software can handle the expected load over a long period of time.

**Volume testing** –

* Under Volume Testing large no. of. Data is populated in a database, and the overall software system’s behaviour is monitored.
* The objective is to check software application’s performance under varying database volumes.

**Scalability testing**–

* The objective of scalability testing is to determine the software application’s effectiveness in “scaling up” to support an increase in user load.
* It helps in planning capacity addition to the software system.

**Performance Testing Tools:**

* [**Jmeter**](http://jmeter.apache.org/)
* [**Open STA**](http://opensta.org/)
* [**Load Runner**](http://www.hp.com/)
* [**Web Load**](http://www.radview.com/)

**Volume Testing**

* **Volume Testing is done when the software is subjected to a huge volume of data.**
* **To analyse the system performance by increasing the volume of data in the database.**

**The objective of performing the volume testing is to**

* **Check system performance by increasing volume of data in the database.**
* **To identify the problem that occur with large amount of data.**
* **To find out the point at which the systems are in stability.**
* **Volume Testing will help to identify the capacity of the system or application – normal and heavy volume.**

|  |  |
| --- | --- |
| **Volume Testing Vs Load Testing** | |
| **Volume Testing** | [**Load Testing**](https://www.guru99.com/load-testing-tutorial.html) |
| **Volume testing is testing of an application with large amount of data in database** | **While in load testing, application is subjected to certain level of load to analyse the behaviour of the application** |
| **Volume testing verifies if the system responds as expected for a certain volume of data.** | **Load testing checks the performance of the system when the user load is increased.** |
| **By increasing size of the file., volume testing can be performed.** | **By increasing number of the file, load testing can be performed.** |

**Security Testing**

* Security testing is a type of software testing that focuses on evaluating the security of a system or application.
* Security Testing is done to identify the vulnerabilities and potential threats and the risks in a software application.
* Also to ensure that the system is protected against unauthorized access, data breaches, and other security-related issues.

Sample Examples -

* A password should be in encrypted format.
* Application or System should not allow invalid users.
* Check cookies and session time for application.
* For financial sites, the Browser back button should not work.

### **Principles of Security Testing:**

Below are the basic principles of security testing:

* Confidentiality
* Integrity
* Authentication
* Authorization
* System Availability

**ETL Security**

* ETL Security refers to the security measures taken to safeguard data during the ETL process.
* ETL (extract, transform, load) is a data integration method that combines information from a variety of disparate sources and loads it into a centralized data warehouse.
* Good data security is important for ETL processes, as they involve converting the information into a coded form using encryption algorithms to ensure that the data can only be accessed by authorized individuals who possess the encryption key.

**Lack of Awareness**

* Data leakage happens when data is unintentionally or accidentally exposed or disclosed during the ETL process.
* Data leakage can result from human errors, miscommunication, or lack of awareness.

**Absence of Tools:**

Data security is a critical aspect of managing and processing data. When **ETL (Extract, Transform, Load)** tools are absent, several implications arise:

**Manual Processes and Human Error**:

* Without ETL tools, data extraction, transformation, and loading tasks are performed manually.
* This increases the risk of **human error**, leading to data inconsistencies, inaccuracies, and security issues.

**Lack of Encryption and Secure Transfer**:

* ETL tools often provide **built-in mechanisms for data encryption** during data transfer and load.
* Without these tools, data transfer may occur over insecure channels, exposing sensitive information to potential threats.

**Data Consistency and Quality**:

* ETL processes enforce data consistency by transforming raw data into a consistent format.
* Without ETL, data quality may suffer, affecting decision-making and potentially leading to security gaps.

**Data Silos**:

* Without ETL, data remains scattered across various systems, creating data silos.
* These silos hit the centralized security management system and increase the risk of unauthorized access.

**Lack of Standard Approach/Methodology:**

when standard approaches or methods are not followed in ETL security, several consequences can arise:

**Data Breaches**:

* Ignoring security best practices can expose sensitive data to unauthorized access.
* [Data breaches can result in financial losses, legal penalties, and damage to an organization’s reputation](https://hevodata.com/learn/factors-to-ensure-etl-security/).

**Non-Compliance**:

* Failing to adhere to security standards and regulations can lead to non-compliance.
* [This may result in fines, legal actions, and loss of trust from customers and stakeholders.](https://hevodata.com/learn/factors-to-ensure-etl-security/)

**Data Violations**:

* Neglecting data requirements can lead to legal complications.
* [Data must be stored and processed in compliance with regional laws and regulations](https://hevodata.com/learn/factors-to-ensure-etl-security/).

**Inadequate Encryption and Hashing**:

* If encryption and hashing mechanisms are not properly implemented, data in transit (during ETL processes) may be vulnerable(Exposed).
* [Proper encryption ensures data confidentiality and integrity](https://hevodata.com/learn/factors-to-ensure-etl-security/).

**Insufficient Data Masking**:

* Sensitive information (such as login credentials, financial details, and protected health information) should be masked to prevent unauthorized exposure.
* [Failure to do so can lead to privacy violations](https://hevodata.com/learn/factors-to-ensure-etl-security/)