**What is test case?**

“**A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements and works correctly.**”

### **GUI and Usability Test Scenarios**

1. All fields on page (e.g. text box, radio options, dropdown lists) should be aligned properly

2. Scroll bar should be enabled only when necessary

3. Description text box should be multi-line

4. Disabled fields should be grayed out and user should not be able to set focus on these fields

5. Upon click of any input text field, mouse arrow pointer should get changed to cursor

6. Information filled by users should remain intact when there is error message on page submit. User should be able to submit the form again by correcting the errors

7. Tab and Shift+Tab order should work properly

8. All buttons on page should be accessible by keyboard shortcuts and user should be able to perform all operations using keyboard

9. Check all pages for broken links

10. All pages should have title

11. Confirmation messages should be displayed before performing any update or delete operation

12. User should be able to select only one radio option and any combination for check boxes.

### **Test Scenarios for Result Grid**

1. Page loading symbol should be displayed when it’s taking more than default time to load the result page

2. Sorted columns should be displayed with sorting icon

3. Result grid values should be sorted by default column

4. Result grids should include all specified columns with correct values

5. Pagination should be enabled when there are more results than the default result count per page

6. Check for Next, Previous, First and Last page pagination functionality

7. Duplicate records should not be displayed in result grid

### **Test Scenarios for a Window**

1. Check if default window size is correct

2. Check if child window size is correct

3. If child window is opened, user should not be able to use or update any field on background or parent window.

4. Check window minimize, maximize and close functionality.

5. Check if window is resizable.

6. Check scroll bar functionality for parent and child windows

7. Check cancel button functionality for child window

### [**Database Testing**](http://www.softwaretestinghelp.com/database-testing-%e2%80%93-practical-tips-and-insight-on-how-to-test-database/) **Test Scenarios**

1. Check if correct data is getting saved in database upon successful page submit

2. Check for data integrity. Data should be stored in single or multiple tables based on design

3. Tables should have primary key column

4. For every database add/update operation log should be added

5. Required table indexes should be created

6. Database logical names should be given according to database name (again this is not standard but helpful for DB maintenance)

7. Check values for table audit columns (like createddate, createdby, update\_date, updatedby, isdeleted, deleted\_date, deleted\_by etc.) are populated properly

8. Check if database fields are designed with correct data type and data length

9. Check if all table constraints like Primary key, Foreign key etc. are implemented correctly

10. Input field leading and trailing spaces should be truncated before committing data to database

11. Null values should not be allowed for Primary key column

### **Test Scenarios for Image Upload Functionality**

1. Check for uploaded image path.

2. Check image upload and change functionality.

3. Check image upload functionality with images having space or any other allowed special character in filename.

4. Check duplicate name image upload.

5. Check if images of specified height and width (if defined) are accepted otherwise rejected.

6. Image upload progress bar should appear for large size images

7. Check multiple images upload functionality

8. Check if user is able to use/view the uploaded images

### **Performance Testing Test Scenarios**

1. Check if page load time is within acceptable range

2. Check page load on slow connections

3. Check response time for any action under light, normal, moderate and heavy load conditions

4. Check database query execution time

5. Check for load testing of application

6. Check CPU and memory usage under peak load condition

### [**Security Testing**](http://www.softwaretestinghelp.com/security-testing-of-web-applications/) **Test Scenarios**

1. Check for SQL injection attacks

2. . Secure pages should use HTTPS protocol

3. Error messages should not reveal any sensitive information

4. Test password security and password policy enforcement

5. Check application logout functionality

6. Sensitive fields like passwords and credit card information should not have auto complete enabled

7. File upload functionality should use file type restrictions and also anti-virus for scanning uploaded files

8. Password and other sensitive fields should be masked while typing

9. Check if forgot password functionality is secured with features like temporary password expiry after specified hours and security question is asked before changing or requesting new password

10. Check if access privileges are implemented correctly

11.

We can test browser for basic functionalists like

1. Test: Open given URL, Homepage

2. Test: Refresh, Stop

3. Test: Multiple tabs

4. Test: History Maintenance

5. Test: Security like blocked sites