**CHAPTER-6**

**SYSTEM IMPLEMENTATION AND TESTING**

* 1. **CODING STANDARDS**
* **Client side Coding Standards**
* Using proper comments while making HTML and CSS pages e.g.: <!—Header goes here--> and /\* Styles goes here\*/.
* Separate component for each module.
* Proper naming convention followed.
* Method, property name, variable are in camel case.
* Property names, class names are in pascal case.
* Each method has proper comment which explains function detail.
* Unused spaces and variables are must be removed from code.
* **Server side Coding Standards**
* Class name, method name, property name must be in pascal case.
* Variable, constant are in camel case.
* All database models must be in model folder.
* All business logic must goes into repository folder.
* All controller action must be in controller folder
* All code which is useful in multiple classes should goes into utility folder.
* All string constants and global constant must be in utility folder.
* All client side code must be in root folder.
  1. **TESTING METHODS:**

There are different methods that can be used for software testing. Here, we used black-box testing and white-box testing methods to test our software.

**6.2.1 Black Box Testing**

The technique of testing without having any knowledge of the interior workings of the application is called black-box testing. The tester is oblivious to the system architecture and does not have access to the source code. Typically, while performing a black-box test, a tester will interact with the system's user interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

**The main advantages of black box testing is**

* Code access is not required.
* Large numbers of moderately skilled testers can test the application with no knowledge of implementation, programming language, or operating systems.

**The main disadvantages of black box testing is**

* Inefficient testing, due to the fact that the tester only has limited knowledge about an application.

**6.2.2 White-Box Testing:**

White-box testing is the detailed investigation of internal logic and structure of the code. White-box testing is also called glass testing or open-box testing. In order to perform white-box testing on an application, a tester needs to know the internal workings of the code.

The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.

**The advantages of white-box testing is**

* The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately.
* It helps in optimizing the code.
* Extra lines of code can be removed which can bring in hidden defects.

**The disadvantages of white-box testing is**

* Due to the fact that a skilled tester is needed to perform white-box testing, the costs are increased.
* Sometimes it is impossible to find out hidden errors that may create problems, as many paths will go untested.
  1. **TEST SUITES DESIGN:**

Test suite is a container that has a set of tests which helps testers in executing and reporting the test execution status. It can take any of the three states namely Active, in progress and completed.

A Test case can be added to multiple test suites and test plans. After creating a test plan, test suites are created which in turn can have any number of tests.

Test suites are created based on the cycle or based on the scope. It can contain any type of tests, functional or Non-Functional.

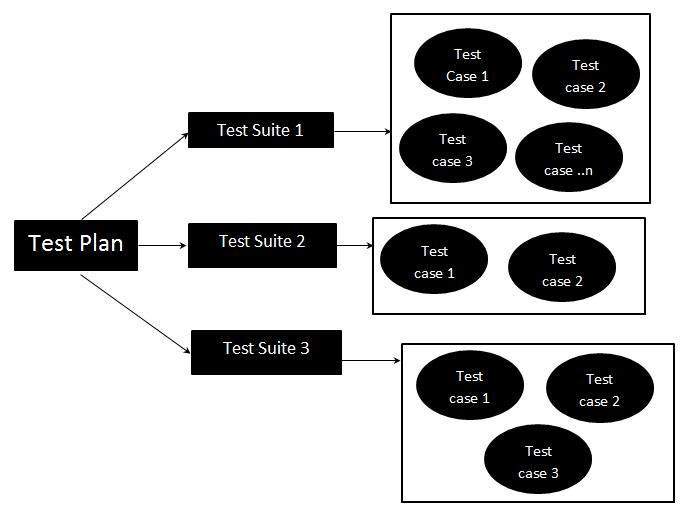


Fig 6.1 Test Suite Design

* 1. **TEST CASES:**

1. **Login**

Table 6.1 Test Case 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr No.** | **Description** | **Test Steps** | **Test data** | **Expected Results** | **Actual Results** | **Pass/**  **Fail** |
| 1. | Check Customer Login with valid Data | 1. Go to Site 2. Enter email 3. Enter password 4. Click Login | email: [abc@xyz.com](mailto:abc@xyz.com)  password:  abc2468@xyz | Login should be successful | As expected | Pass |
| 2. | Check Customer Login with invalid Data | 1. Go to Site 2. Enter email 3. Enter password 4. Click Login | email: [acd@xyz.com](mailto:acd@xyz.com)  password:  abc1357@xyz | User should not Login into an application  Message:  Invalid email or password. | As expected | Pass |

1. **Registration**

Table 6.2 Test Case 2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr No.** | **Description** | **Test Steps** | **Test data** | **Expected Results** | **Actual Results** | **Pass/**  **Fail** |
| 1. | Check Customer Registration data with validation | 1. Go to Site 2. Enter required details. 3. Click Create Account | Name: abc  Email: [abc@xyz.com](mailto:abc@xyz.com)  Country:  India(+91)  Phone number:  9978695768  Password:  abc2468@xyz  Confirm Password | Password and Confirm Password should be match.  Phone number according to country code should be match.  Email must be valid. | As expected | Pass |
| 2. | Check invalid format of email | Follow above steps with invalid email | email: [acd@xy.com](mailto:acd@xy.com) | User should not register. Message: Invalid email. | As  expected | Pass |
| 3. | Check different Phone number digits with country code. | Follow above steps with invalid Phone number digits | Country:  India(+91)  Phone number:  997869576 | User should not register. Message: Invalid phone number. | As  expected | Pass |
| 4. | Enter password which does not match with regex. | Follow above steps with wrong password format | Password:  abdhsgj | User should not register.  Message:  Password must be alphanumeric including at least 1 uppercase letter,1 lowercase letter with minimum 8 characters. | As  expected | Pass |
| 5. | Enter different entries in password and confirm password fields. | Follow above steps with different entries in given fields | Password:  asdf1357@asd  Confirm password:  asdf145@asgh | User should not register. Message: ‘Confirm Password' and 'Password' do not match. | As  expected | Pass |

1. **Create Job**

Table 6.3 Test Case 3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr No.** | **Description** | **Test Steps** | **Test data** | **Expected Results** | **Actual Results** | **Pass/**  **Fail** |
| 1. | Check Required Field | 1. Go to Site 2. Select Create job tab 3. Add data in text fields. 4. Save | Enter Proper Data In all Text fields | System should not display required field error message | As expected | Pass |
| 2. | Check Required fields with empty text | 1. Go to Site 2. Select Create job tab 3. Do not fill details in text field 4. Save | Empty text field | System should display required field error message | As expected | Pass |
| 3. | Add attachments | 1. Go to Site 2. Select Create job tab 3. Attach file from computer to add in job. 4. Save | File in given format | System should properly attach given file | As  expected | Pass |
| 4. | Create Task/Asset Form | 1. Go to Site 2. Select Create job tab 3. Select Add/Create Form in task or assets tab 4. OK | Select existing form or create new form and save it | System should add selected form in Job | As expected | Pass |

1. **Save/Update/Delete Job**

Table 6.4 Test Case 4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr No.** | **Description** | **Test Steps** | **Test data** | **Expected Results** | **Actual Results** | **Pass/**  **Fail** |
| 1. | Save Job | 1. Go to site 2. Create job with proper validation 3. Click Save | All proper job details | System should save all given data for given job | As expected | Pass |
| 2. | Update Job | 1. Go to site 2. Update job details 3. Update | Change job details | System should properly add newly updated job details | As expected | Pass |
| 3. | Delete Job | 1. Go to site 2. Click on delete job icon for specific job 3. Confirm delete | Delete button click | System should delete Job from database as well as from user interface | As expected | Pass |

1. **Generate job details report**

Table 6.5 Test Case 5

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sr No.** | **Description** | **Test Steps** | **Test data** | **Expected Results** | **Actual Results** | **Pass/**  **Fail** |
| 1. | View Job | 1. Go to site 2. Select Job tab 3. Select job title to view details | Click job title | System should display All details of job in required format with correct data | As expected | Pass |
| 2. | PDF | 1. Go to site 2. Select PDF option from view job UI 3. Select appropriate layout options 4. OK | Add customization options as per requirement | System should properly add layout options selected by user. | As expected | Pass |
| 3. | Get Report | 1. Go to site 2. Select PDF option from view job UI 3. Click get report after adding layout options | PDF request | System should  Send PDF download request to the user via mail | As expected | Pass |