**Test Case completeness of Frontend requirements**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr .No** | **Test case Name** | **Steps to execute** | **Expected Result** | **Result** |
| 1 | Verify component at the top of the page. | 1.Login to the application  2.Observe the component at the top of the page | Expect that component should display Name, Description, Price, and Add New item button.  Expect that component should display a Image Upload button. |  |
| 2 | Verify the Name text box | 1. Login to the application.  2.Enter the valid item name i.e. text value for e.g. Mobile  3. Enter the item name more than the allowed value.  4. Enter the item name less than the allowed value i.e. check for blank value 5.Enter the invalid item name i.e. enter the Number, special character, decimal number. | 1. Expect that user should login successfully.  2. Expect that user should be able to enter the valid item name. It should accept only alphabets.  3.Expect that UI should not be allow user to enter the name more than the allowed value  4. Expect that name should be mandatory field. UI should not allow user to enter/keep the name field blank.  3. Expect that user should receive error message to enter invalid item name. It should not accept number, special character, decimal number etc. |  |
| 3 | Verify the Description text box | 1. Login to the application.  2. Enter the valid item Description containing only alphabets.  3. Enter the item Description more than the allowed value.  4. Enter the item Description less than the allowed value i.e. check for blank value 5.Enter the invalid item Description i.e. enter the Number, special character, decimal number | 1. Expect that user should login successfully.  2. Expect that user should be able to enter the valid item Description with only alphabets.  3.Expect that UI should not be allow user to enter the Description more than the allowed value  4. Expect that UI should not allow user to enter/keep the Description. field blank.  3. Expect that description textbox should not accept number, special character, decimal number etc. |  |
| 4 | Verify the Price text box | 1. Login to the application.  2. Enter the valid item Price i.e. number, decimal number.3. Enter the –ve value in price text box.  4. Enter the item Price text box for blank value 5.Enter the invalid item Price i.e. enter the text value | 1. Expect that user should login successfully.  2. Expect that user should be able to enter item Price i.e. number, decimal number.  3.Expect that UI should not be allow user to enter the -ve value  4. Expect that Price text field should be mandatory. UI should not allow user to enter/keep blank value in Price text box..  3. Expect that description textbox should not accept text value. |  |
| 5 | Verify the functionality of Add New Item button | 1. Login to the application.  2. Enter valid value in Name, Description, and Price text box.3.Click ‘Add New Item’ button. | 1. Expect that user should login successfully 2.Expect that New item should be added to the inventory.  3.Expect that newly added item should display in list containing the name and price of the item |  |
| 6 | Verify the functionality of upload a picture of the item. | 1. Login to the application.  2. Enter valid value in Name, Description, and Price text box. Click ‘Upload Item image’ button. Upload valid image file with extension PNG, Tiff, JPEG, Gif,PDF.  3. Click ‘Upload Item image’ button. Upload word, excel, ppt file  4. Click ‘Add new item’ button. | 1. Expect that user should login successfully.2.Expect that user should be able to add new item with valid image file. Expect that user should receive green banner message saying “Item added successfully”  3. Expect that user should not be able to upload word, excel, ppt file.Expect error to upload invalid image.  4.Expect that newly added item should display in list containing the name ,price and thumbnail of the item |  |
| 7 | Verify delete button functionality | 1. Login to the application.  2. Observe the Delete button location.  3. Click delete button.  4. Click ‘Ok’ button.  5. Click Delete button. Click ‘Cancel’ button on the warning dialog box. | 1. Expect that user should login successfully.2. Each item in the list should have a delete button.  3. Expect that warning message should display saying “Do you want to delete item” with ‘Ok’ and ‘Cancel’ button.4.Expect that item should be deleted successfully. Expect green banner message saying “Item deleted successfully” 5.Expect that on click of cancel button, item delete should be cancelled. Item should not be deleted. |  |
| 8 | Verify full page reload | 1. Login to the application.  2.Navigate between the two pages of the app. | 1. Expect that user should login successfully.  2.Expect that full page reload when user navigate between the two pages of the app. |  |
| 9 | Verify the navigation on click of the item. | 1. Login to the application.  2. Click on the item display in the list. | 1. Expect that user should login successfully.  2.Expect that click on an item should bring the user to a new page with all the information about the selected item (name, description, price, and optionally photo). |  |
| 10 | Verify the UI of both pages | 1. Login to the application.  2.Observe the UI of both pages | 1. Expect that user should login successfully.  2. Expect that content of both pages should be both horizontally and vertically centered. |  |

**Test Case completeness of Backend requirements :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case Name** | **Steps to execute** | **Expected Result** | **Result** |
| Verify the Item table structure | 1. Connect to the DB.  2.Observe the Item table column and data size and data type | 1.Expect that user should be able to connect DB successfully.2.Expect below columns in Item table : 1.Item ID int,  2.ItemName varchar(20),  3.Description varchar(40),  4.Price float |  |
| Verify table constraint | 1. Connect to the DB.  2.Observe constraint in the Item table | Expect constraint applied on column are :  1.Primary key constraint on Item ID 2.Not null constraint on all column  3.Unique key constraint on Item name |  |
| Verify record count in Item table | 1. Connect to the DB.  2.Run query : Select count(\*) from Item | Expect the number of records in db should match with the no of records in UI in list |  |

**Identify and define Test Cases for automation :**

Below testcases can be automated :

1. Verify component at the top of the page
2. Verify the Name text box
3. Verify the Description text box
4. Verify the Price text box
5. Verify the functionality of Add New Item button
6. Verify the functionality of upload a picture of the item.
7. Verify delete button functionality
8. Verify the navigation on click of the item