|  |  |  |
| --- | --- | --- |
| Test Case Id | Test\_001 | |
| Test Scenario | Verify the adding functionality with a valid barcode, supplier, product name, wholesale price, and retail price. | |
| Test Steps | | |
| 1. Open the application and navigate to the Product page, and locate the Add Product option. 2. Click on Add Product. 3. Enter a valid barcode in the “Barcode” field. 4. Select a valid and appropriate supplier in the “Supplier” field. 5. Enter a valid product name in the “Product Name” field. 6. Enter a valid wholesale price in the “Wholesale Price” field. 7. Enter a valid retail price in the “Retail Price” field. 8. Click the “Add” button. | | |
| Prerequisites | | |
| * Users should be logged into the application. * The supplier of the product must be added to the system before adding the product of that supplier. | | |
| Test data | | |
| Barcode: 17  Supplier: Munchee  Product name: Cream cracker  Wholesale price: 110  Retail price: 120 | | |
| Expected/ Intended Results | | |
| * Display a success message: “Product added successfully.” * Refresh the table with newly added data. | | |
| Actual Results | | * Displayed success message “Product added successfully.” * Refresh the table with newly added data. |
| Test status – Pass/ Fail | | Pass |

# Add product

# Edit Product

|  |  |  |
| --- | --- | --- |
| Test Case Id | Test\_002 | |
| Test Scenario | Verify the Update Product functionality with a valid barcode, supplier, product name, wholesale price, and retail price. | |
| Test Steps | | |
| 1. Open the application and navigate to the Product page and locate the Edit Product option. 2. Click on the Edit Product icon located in the table. 3. Enter a valid barcode in the “Barcode” field (if needed). 4. Select a new valid and appropriate supplier in the “Supplier” field (if needed). 5. Enter a new valid product name in the “Product Name” field (if needed). 6. Enter a new valid wholesale price in the “Wholesale Price” field (if needed). 7. Enter a new valid retail price in the “Retail Price” field (if needed). 8. Click the “Add” button. | | |
| Prerequisites | | |
| * Users should be logged into the application. * Add the updating supplier to the system if the supplier is not added to the database. | | |
| Test data | | |
| Barcode: 20  Supplier: Maliban  Product name: Cream cracker  Wholesale price: 150  Retail price: 180 | | |
| Expected/ Intended Results | | |
| * Display a success message: “Product updated successfully.” * Refresh the table with updated data. | | |
| Actual Results | | * Displayed success message “Product updated successfully.” * Refresh the table with updated data. |
| Test status – Pass/ Fail | | Pass |

# Delete Product

|  |  |  |
| --- | --- | --- |
| Test Case Id | Test\_003 | |
| Test Scenario | Verify the Delete Product functionality. | |
| Test Steps | | |
| 1. Open the application and navigate to the Product page, and locate the Delete Product option in the table under the “Actions” column. 2. Click on the Delete Product icon. 3. Deletion Confirmation alert occurs. 4. Click the “delete” option in the confirmation alert. | | |
| Prerequisites | | |
| * Users should be logged into the application. | | |
| Test data | | |
| A specific products that has been added to the database. | | |
| Expected/ Intended Results | | |
| * Display a success message: “Product deleted successfully.” * Refresh the Table with existing product records. | | |
| Actual Results | | * Displayed success message “Product updated successfully.” * Refresh the Table with existing product records. |
| Test status – Pass/ Fail | | Pass |

# Product Report Generation

|  |  |  |
| --- | --- | --- |
| Test Case Id | Test\_004 | |
| Test Scenario | Verify the Product Generation report by using the added product data in the database using the “Add Product” function. | |
| Test Steps | | |
| 1. Open the application and navigate to the Report Generation Page 2. Click on the “Product Report” option. | | |
| Prerequisites | | |
| * Users should be logged into the application. | | |
| Test data | | |
| Existing product data records in the database. | | |
| Expected/ Intended Results | | |
| * Display a success message: “Report generated successfully at Document/SmartPOS.” * Generate and save the report locally at “Document/SmartPOS” | | |
| Actual Results | | * Displayed a success message: “Report generated successfully at Document/SmartPOS.” * Generated and save the report locally at “Document/SmartPOS” |
| Test status – Pass/ Fail | | Pass |

# Search Product

|  |  |  |
| --- | --- | --- |
| Test Case Id | Test\_005 | |
| Test Scenario | Verify that the Searching function works properly. | |
| Test Steps | | |
| 1. Open the application and navigate to the Product Page 2. Click on the “search” field. 3. Search for a Product | | |
| Prerequisites | | |
| * Users should be logged into the application. * Product records must exist. | | |
| Test data | | |
| Existing products “Product Name” | | |
| Expected/ Intended Results | | |
| * Display filtered data records on the product table. | | |
| Actual Results | | * Displayed filtered data records on the product table. |
| Test status – Pass/ Fail | | Pass |