**Coursework Test Plan -5COSC19C Object-Oriented Programming**

Student Name – Dumindu Induwara Gamage

IIT ID -20221168

UOW ID -w1953846

**Test plan for the console Application**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test No | Test Input | Expected Result | Actual Result | Pass/Fail |
| Menu Item 1 – Show Menu (“M” or “m”) | | | | |
| 01 | Enter “M” to the main prompt | Show menu | Show menu | Pass |
| 02 | Enter “m” to the main prompt | Show menu | Show Menu | Pass |
| 03 | Enter any other character or Character combination | Show an error. | Show an error.  Output:  (SEEMS LIKE, YOU ARE ENTERING A WRONG MENU KEY. PLEASE CHECK IT AND TRY AGAIN) | Pass |
| Menu item 2 -Add product to the List | | | | |
| 01 | Enter “A” to the main prompt | Request product details. | Request product details. | Pass |
| 02 | Enter “a” to the main prompt | Request product details. | Request product details. | Pass |
| 03 | Valid inputs for product type, product ID, name, and price | Product added successfully | Product added successfully | Pass |
| 04 | Attempt to add a product when the maximum limit is reached | "Cannot add more products. Maximum products limit reached." printed | "Cannot add more products. Maximum products limit reached." printed | Pass |
| 05 | Invalid product type (e.g., a string instead of a number) | "Error: Invalid input format. Please enter a valid number for price or product type." printed | "Error: Invalid input format. Please enter a valid number for price or product type." printed | Pass |
| 06 | Invalid price (e.g., a non-numeric value) | "Error: Invalid input format. Please enter a valid number for price or product type." printed | "Error: Invalid input format. Please enter a valid number for price or product type." printed | Pass |
| 07 | Exception during product creation | Proper error message printed | Proper error message printed | Pass |
| Menu item 3 -Show Product Details | | | | |
| 01 | Products added to the list. | Display a formatted table with product details | Display a formatted table with product details | Pass |
| 02 | No products in the list | "No products available. Please add the products and check." printed | "No products available. Please add the products and check." printed | Pass |
| 03 | Different product types in the list | Correct mapping of numeric product types to string names in the table | Correct mapping of numeric product types to string names in the table | Pass |
| 04 | Products with different quantities and prices | Display correct quantities and prices in the table | Display correct quantities and prices in the table | Pass |
| 05 | Check formatting and alignment | Ensure that the table is formatted correctly, and columns are aligned | Ensure that the table is formatted correctly, and columns are aligned | Pass |
| Menu item 4-deleteProduct | | | | |
| 01 | Product ID of an existing product | "Product deleted successfully." printed, specified product removed from the list | "Product deleted successfully." printed, specified product removed from the list | Pass |
| 02 | Product ID of a non-existing product | "Product not found with ID: [Product ID]" printed | "Product not found with ID: [Product ID]" printed | Pass |
| 03 | Invalid Input (e.g., non-numeric Product ID) | "Invalid input. Please enter a valid Product ID." printed | "Invalid input. Please enter a valid Product ID." printed | Pass |
| Menu item 5-SavetoFile | | | | |
| 01 | List of Electronics and Clothing products | CSV file created with correct headers and product details | CSV file created with correct headers and product details | Pass |
| 02 | Empty product list | "Product data saved to CSV: [filename]" printed, CSV file contains only headers | "Product data saved to CSV: [filename]" printed, CSV file contains only headers | Pass |
| 03 | IOException during file writing | "Error while saving data: [Error Message]" printed | "Error while saving data: [Error Message]" printed | Pass |
| 04 | Mix of valid and invalid products | CSV file created with valid products, invalid products cause exceptions but are caught and printed | CSV file created with valid products, invalid products cause exceptions but are caught and printed | Pass |
| Menu item 5-loadtoFile | | | | |
| 01 | Valid CSV file with Electronics and Clothing products | The products list contains correct instances of Electronics and Clothing | The products list contains correct instances of Electronics and Clothing | Pass |
| 02 | Empty CSV file | The products list is empty | The products list is empty | Pass |
| 03 | CSV file with invalid data (e.g., missing fields) | Invalid data skipped, appropriate error message printed | Invalid data skipped, appropriate error message printed | Pass |

**GUI Test Cases**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case | Test Object | Test Steps | | Expected Result | Status |
| 1 | Initialization and Display | 1. Instantiate WestminsterShoppingManager. 2. Create an instance of MainFrame, passing the WestminsterShoppingManager. | | MainFrame is created without errors, and the GUI components are displayed | Passed |
| 2 | Product Category Selection | 1. Open the MainFrame. 2. 2. Select different categories (All, Electronics, Clothings) from the combo box. | | The displayed table updates according to the selected category. | Passed |
| 3 | Shopping Cart Button | 1. Open the MainFrame. 2. Click the "Shopping Cart" button | | The ShoppingCartFrame is displayed without errors. | Passed |
| 4 | Add to Shopping Cart Button | 1. Open the MainFrame. 2. Select a product from the table. 3. Click the "Add to Shopping Cart" button. | | The selected product is added to the shopping cart in the ShoppingCartFrame. | Passed |
| 5 | Table Display and Selection | 1. Open the MainFrame. 2. Check if the table displays product information. 3. Select different rows in the table. | | Product information is displayed in the table, and selecting a row triggers the display of product details. | Passed |
| 6 | Product Details Display | 1. Open the MainFrame. 2. Select a product from the table. | | Product details (Product ID, Name, Category, Price, Info) are displayed accurately below the table | Passed |
| 7 | Shopping Cart Integration | 1. Open the MainFrame. 2. Select a product from the table. 3. Click the "Add to Shopping Cart" button. 4. Verify the product is added to the ShoppingCartFrame. | | The selected product is added to the shopping cart, and the shopping cart updates accordingly. | Passed |
| 8 | Category Filtering | | 1. Open the MainFrame. 2. 2. Select "All," "Electronics," and "Clothings" from the combo box. | The displayed table updates based on the selected category, showing only relevant products. | Passed |