**Project iii – Software development**

Test Cases

Girish Bhuteja

Student ID: 8845688

Bachelor of Computer Science (Honors)

***Integration Test Cases***

***1.***

|  |  |
| --- | --- |
| Test Case ID | Increment\_Button\_Test |
| Author | Girish Bhuteja |
| Test Case Description | Test the IncreaseQuantity function, ensuring that the item quantity is correctly incremented and sent when the “Send” button is pressed. |
| Testing Prerequisites | A working GUI and all the required buttons like “Send” and “+” are present on the main menu. |
| Testing Steps | 1. Open the SCADA/HMI interface. 2. Click on the + button associated with an item. 3. Click on the Send button. |
| Input Data | Click on the button. |
| Expected Result | The item quantity is updated, and a window showing that the order has been sent should open. |
| Actual Result & Traceability | The quantity of the order increases when clicking the button, and a window saying your order has been sent opens. |
| Pass/Fail | Pass |

***2.***

|  |  |
| --- | --- |
| Test Case ID | Decrement\_Button\_test |
| Author | Girish Bhuteja |
| Test Case Description | Test the DecreaseQuantity function, ensuring that the item quantity is correctly incremented and sent when the “Send” button is pressed. |
| Testing Prerequisites | A working GUI and all the required buttons like “Send” and “-” are present on the main menu. |
| Testing Steps | 1. Open the SCADA/HMI interface. 2. Click on the - button associated with an item. 3. Click on the button. |
| Input Data | Click on the button. |
| Expected Result | The item quantity is updated, and a window showing that the order has been sent should open. |
| Actual Result & Traceability | The quantity of the order decreases when clicking the button, and a window saying your order has been sent opens. |
| Pass/Fail | Pass |

***3.***

|  |  |
| --- | --- |
| Test Case ID | Update\_Status\_Test |
| Author | Girish Bhuteja |
| Test Case Description | Test the UpdateStatus function, verifying that the dish status transitions correctly to ready and from ready to serve. |
| Testing Prerequisites | Item has been ordered. |
| Testing Steps | 1. Open the SCADA/HMI interface. 2. Go to the Chef’s view. 3. Click on the Ready or Served button. |
| Input Data | Click on the button. |
| Expected Result | The dish's status is updated to 'Ready,' reflecting the correct transition. |
| Actual Result & Traceability | When clicking on the “Ready” button in Chef’s view, there is a notification in the Waiter’s view, and the status is changed to “Ready.” Moreover, when clicking on the “Served” button in the Waiter’s view, it shows a notification in Chef’s view, and the status is updated to “Served.” |
| Pass/Fail | Pass |

***4.***

|  |  |
| --- | --- |
| Test Case ID | New\_Window\_Test |
| Author | Girish Bhuteja |
| Test Case Description | Validate that the new window is successfully opened when the button is pressed. |
| Testing Prerequisites | A functional SCADA/HMI system. |
| Testing Steps | 1. Open the SCADA/HMI interface. 2. Click on Customer to login. |
| Input Data | Click on the button. |
| Expected Result | A new window is opened to confirm that do you want to login as customer. |
| Actual Result & Traceability | When the Customer button is pressed to log in, a new window opens to confirm and ask if you want to log in as a customer. |
| Pass/Fail | Pass |

***5.***

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Update\_Textbox\_Test |  |  |
| Author | Girish Bhuteja |  |  |
| Test Case Description | Test the UpdateTextbox function, ensuring that it successfully updates the textbox value when provided with valid input. |  |  |
| Testing Prerequisites | A textbox in the SCADA/HMI interface. |  |  |
| Testing Steps | 1. Open the SCADA/HMI interface. 2. Enter a valid input in the textbox. |  |  |
| Input Data | Valid Text: [Any valid text] |  |  |
| Expected Result | The textbox value is updated with the provided input. |  |  |
| Actual Result & Traceability | When clicking on the served in the waiter’s view when a dish is being served, it changed the textbox of the status to serve. |  |  |
| Pass/Fail | Pass |  |  |

***6.***

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Update\_Label\_Test |  |  |
| Author | Girish Bhuteja |  |  |
| Test Case Description | Validate the UpdateLabel function, checking if it manages label text correctly based on the provided input. |  |  |
| Testing Prerequisites | A label in the SCADA/HMI interface. |  |  |
| Testing Steps | 1. Open the SCADA/HMI interface. 2. Enter a valid input in the label. |  |  |
| Input Data | Valid Text: [Any valid text] |  |  |
| Expected Result | The label text is updated with the provided input. |  |  |
| Actual Result & Traceability | When clicking on the Ready button from the Chef’s view, the table view of the Waiter shows a notification. |  |  |
| Pass/Fail | Pass |  |  |

***Usability Test Cases***

***1.***

|  |  |
| --- | --- |
| Test Case ID | Request\_Help\_Test |
| Author | Girish Bhuteja |
| Test Case Description | Test if the user can request help by using the "Call Server" button in each table view. |
| Testing Prerequisites | Functional table view with the "Call Server" button. |
| Testing Steps | 1. Open the SCADA/HMI interface. 2. Open the table view interface. 3. Locate and Click on the Call Server button in the table view. |
| Input Data | Click on the button “Call Server”. |
| Expected Result | The system should register the help request, and the waiter should be notified. |
| Actual Result & Traceability | When clicking on the Call server button on the Main menu, it shows a notification to the Waiter’s view showing “!” beside the table that wants help. |
| Pass/Fail | Pass |

***2.***

|  |  |
| --- | --- |
| Test Case ID | View\_Request\_Help\_Test |
| Author | Girish Bhuteja |
| Test Case Description | Test if the waiter can see the notification when a customer from a particular table requests help in the Waiter's view. |
| Testing Prerequisites | Functional Waiter's view with table status notifications. |
| Testing Steps | 1. Open the SCADA/HMI interface. 2. Open the Waiter's view. 3. Check for an exclamation mark beside a table indicating a help request. |
| Input Data | None |
| Expected Result | The waiter should see the notification for the help request from the specific table. |
| Actual Result & Traceability | It shows a notification to the Waiter’s view. |
| Pass/Fail | Pass |

***3.***

|  |  |
| --- | --- |
| Test Case ID | Status\_from\_Waiter\_Test |
| Author | Girish Bhuteja |
| Test Case Description | Test if the Chef can see the notification when the Waiter updates the status of an item in the Waiter's view. |
| Testing Prerequisites | Functional Chefs and Waiters view with table status notifications. |
| Testing Steps | 1. Open the SCADA/HMI interface. 2. Open the Waiter's view. 3. Check for an exclamation mark beside a table indicating a status update from the Chef. |
| Input Data | None |
| Expected Result | The Chef should see the notification for the status update from the Waiter for a specific table. |
| Actual Result & Traceability | The notification is shown in the Chef’s view at that particular table. Moreover, when clicked on that, it shows the initial status to be “Served” in Chef’s view. |
| Pass/Fail | Pass |