**SOFTWARE QUALITY ENGINEERING**

Logo

Description automatically generated with medium confidence

**Test Plan Document**

**Submitted By:**

Muhammad Memoon (20-SE-030)

Sabha Rani(20-SE-032)

Muhammad Huzaifa(20-SE-045)

**Submitted To:**

**Ma’am Farwa Aizaz**

HITEC UNIVERSITY, TAXILA

**Test Plan Document**

Contents

[1. Scope of the Tests 2](#_Toc137760434)

[2. Testing Environment 3](#_Toc137760435)

[3. Test Details 3](#_Toc137760436)

[4. Test Schedule 5](#_Toc137760437)

[6. Test Cases 6](#_Toc137760438)

**Test Plan**

# 1. Scope of the Tests

**1.1 The software package to be tested:**

**- Name:** Online Cafeteria Ordering System

**- Version:** 2.0

**- Revision:** N/A

**1.2 Documents providing the basis for the planned tests:**

- System Requirements Specification (Version 2.5)

- Design Document (Version 3.0)

- User Manual (Version 2.2)

# 2. Testing Environment

**2.1 Testing Sites:**

- Testing will be conducted in a dedicated testing environment.

**2.2 Required Hardware and Software Configuration:**

**- Server:** Minimum 16GB RAM, 500GB HDD, Intel Xeon Processor.

**- Client Devices:** Minimum 8GB RAM, 250GB HDD, Latest Web Browser.

- **Operating System:** Windows 10, macOS, Linux.

**2.3 Participating Organizations:**

**- Testing Team:** QA Department, XYZ Software Solutions.

- **Development Team:** ABC Software Solutions.

**2.4 Manpower Requirements:**

**- Test Lead:** 1 person

**- Testers:** 3 persons

**- Developers:** 2 persons (for error correction)

**2.5 Preparation and Training Required for the Test Team:**

- Test team will be provided with training on the system functionality and test procedures.

- Testers will familiarize themselves with the system by referring to the user manual.

- Developers will receive a briefing on identified issues and necessary fixes.

# 3. Test Details

**3.1 Test Identification: Menu Selection and Order Placement**

**3.2 Test Objective:**

- To verify that users can browse the menu, select items accurately, and place orders successfully, and the system handles valid and invalid inputs appropriately.

**3.3 Cross-reference:**

- Design Document: Section 4.1 - Menu Structure, Section 4.2 - Order Placement.

- System Requirements Specification: Requirement ID - REQ-2.1.1, REQ-2.3.2.

**3.4 Test Case Requirements:**

**- Test case 1:** Successful menu browsing and selection.

**- Test case 2:** Invalid menu selection with out-of-stock items.

- **Test case 3:** Order placement with valid inputs.

**- Test case 4:** Order placement with missing required fields.

**3.5 Special Requirements:**

- Integration with a payment gateway for testing online payments.

- Test environment should include a variety of menu items with different attributes (e.g., vegetarian, gluten-free).

**3.6 Data to be Recorded:**

- Test case ID

- Test case description

- Steps performed

- Expected results

- Actual results

- Pass/Fail status

# 4. Test Schedule

**4.1 Preparation:**

- Review system requirements and design documents: 2 days.

- Test environment setup: 1 day.

- Test case preparation: 3 days.

**4.2 Testing:**

**- Test case 1:** Successful menu browsing and selection: 1 day.

**- Test case 2:** Invalid menu selection with out-of-stock items: 0.5 days.

**- Test case 3:** Order placement with valid inputs: 1 day.

**- Test case 4:** Order placement with missing required fields: 0.5 days.

**4.3 Error Correction:**

- Developers fixing issues found during testing: 5 days.

**4.4 Regression Tests:**

- Repeating all test cases after error correction: 2 days.

**5. Test Deliverables**

**5.1 Test Plan:** This document outlines the scope, objectives, and schedule of the testing activities.

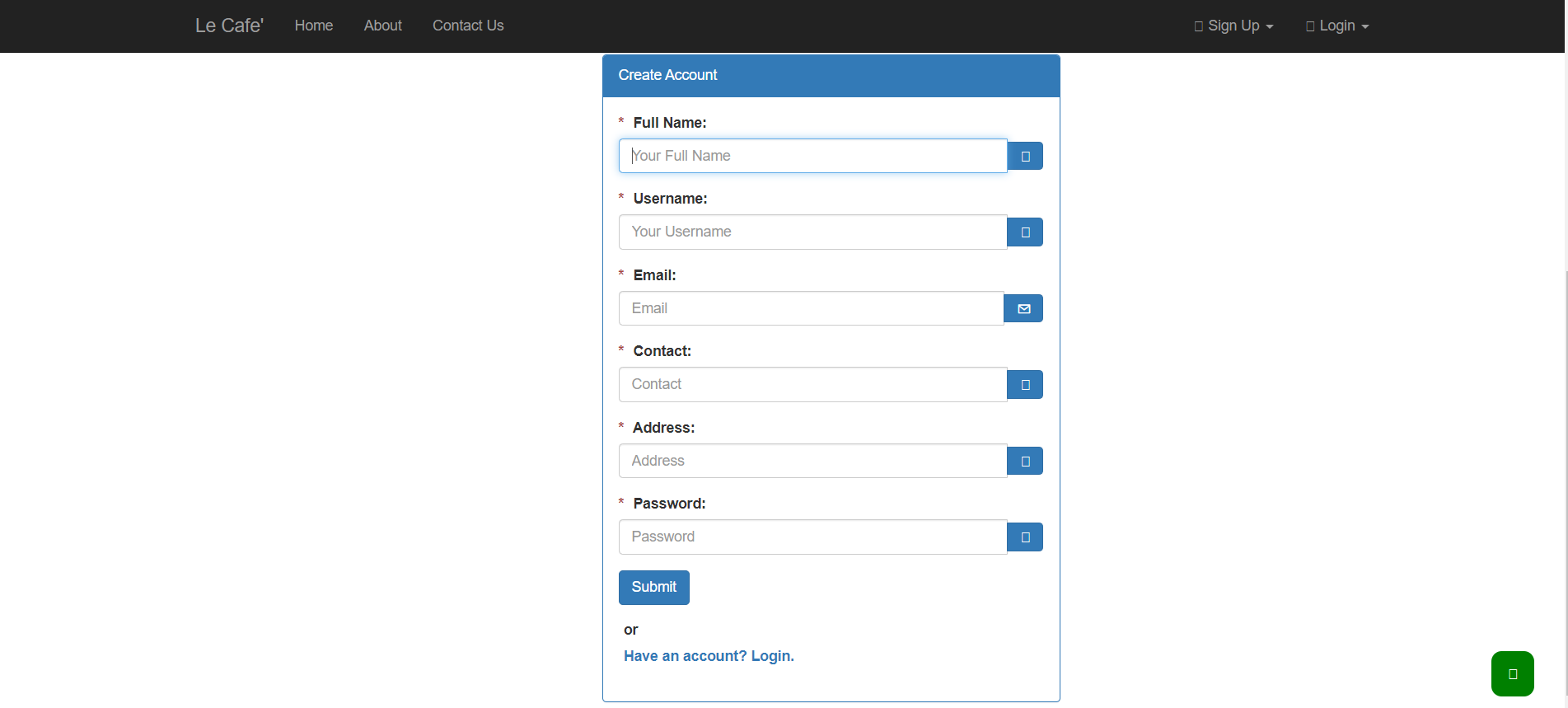
**5.2 Test Cases:** Detailed test cases with step-by-step instructions and expected results.

**5.3 Test Data:** Sample data for testing various scenarios.

# 6. Test Cases

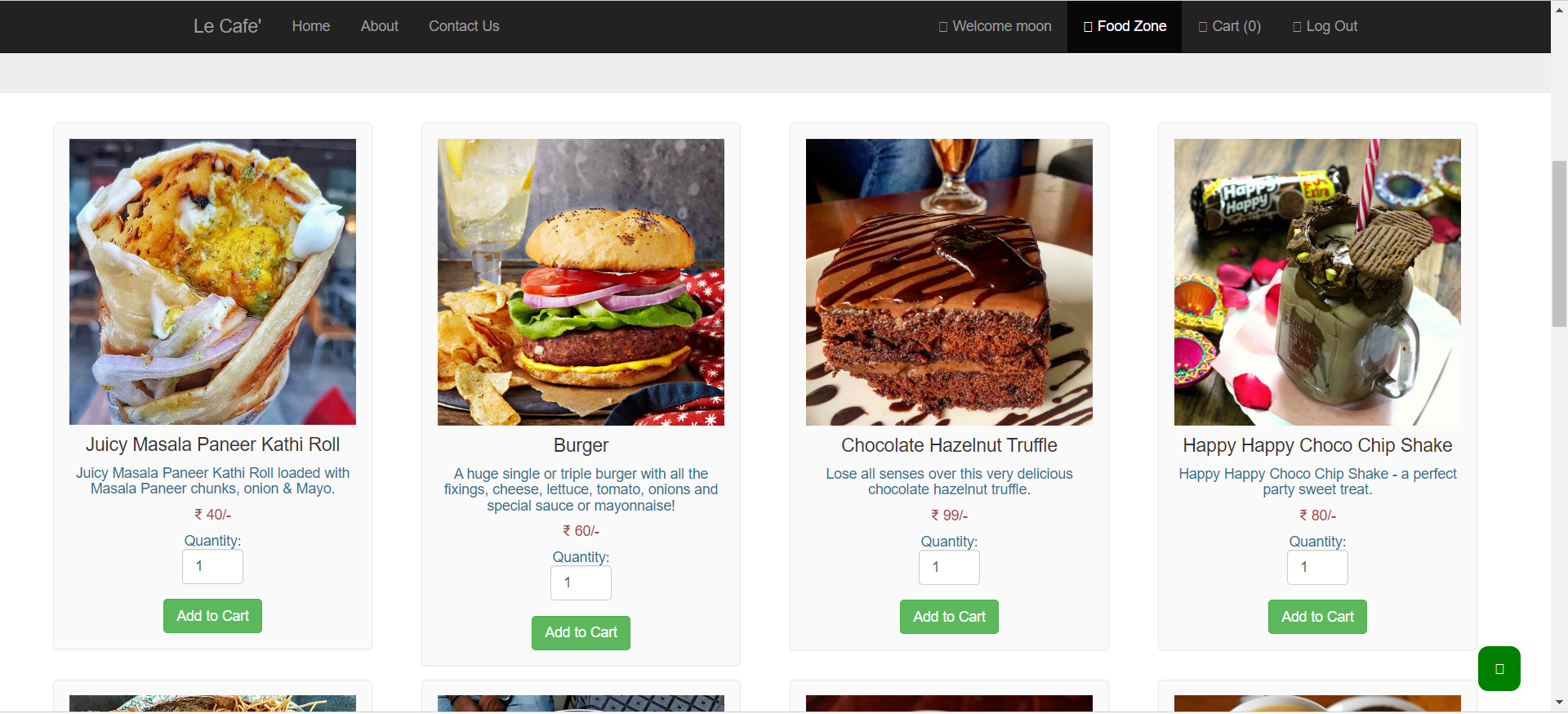
**Feature:** Account creation

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Valid username and password | Sign up successful | Pass | Sign up successful |
| 2 | Username with special characters | Sign up successful | Pass | Sign up successful |
| 3 | Short password | Password should be at least 8 characters long | Fail | Sign-up failed - Password should be at least 8 characters long |
| 4 | Long password | Password exceeds the maximum character limit | Fail | Sign-up failed - Password exceeds the maximum character limit |
| 5 | Duplicate username | Username already exists | Fail | Sign-up failed - Username already exists |



**Feature: Home Screen**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Display home screen | The system should display a home screen with specified components | Pass | Home screen displayed with customize a meal, most rated recipes, account activity, meal plan, and search bar |

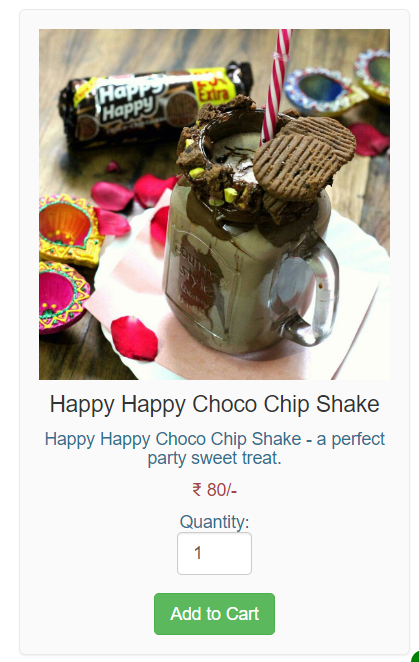


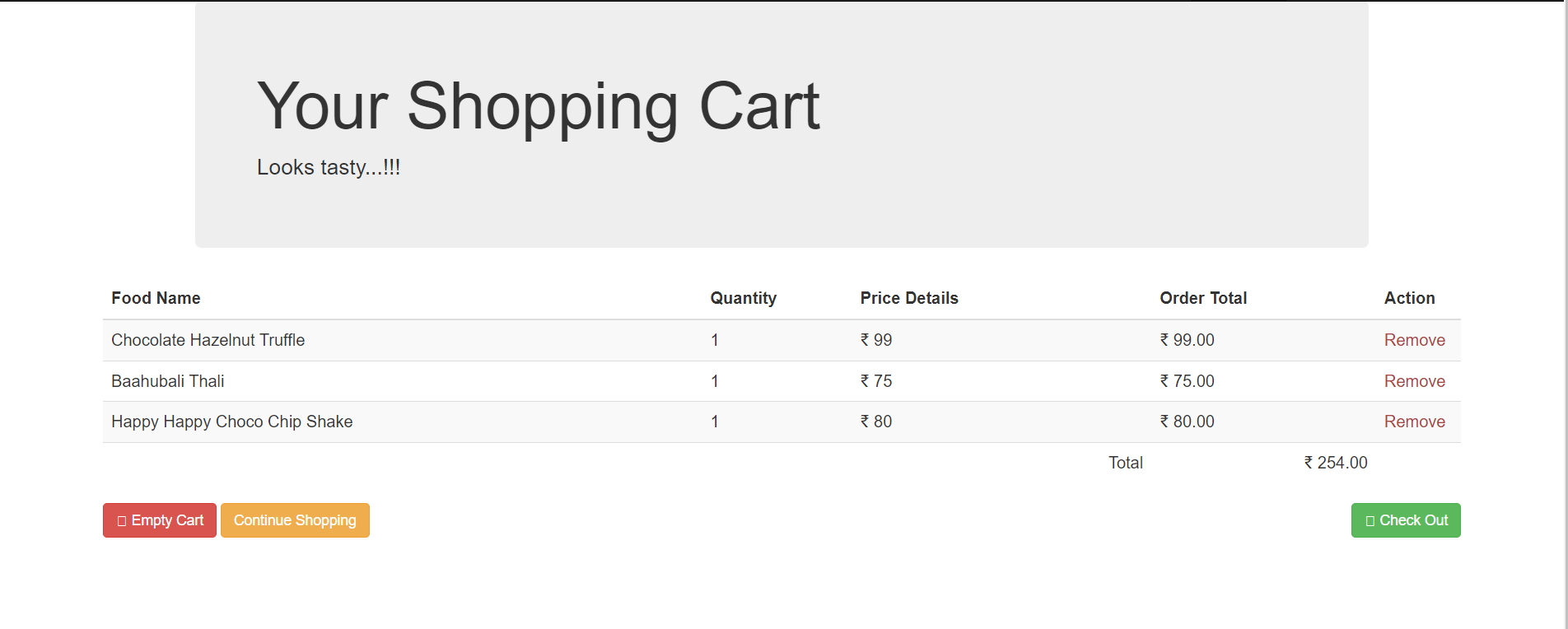
**Feature: Search Bar**

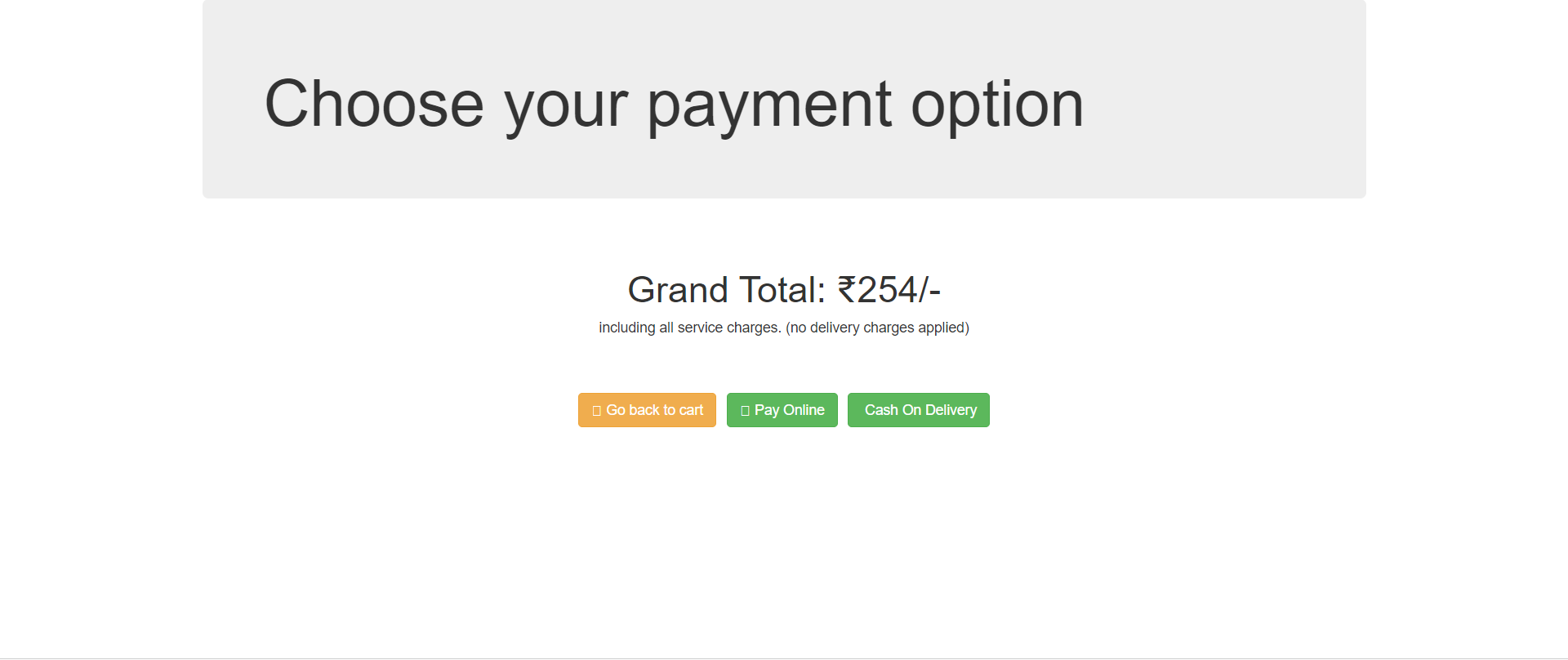
| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Search for keywords | User should be able to enter meal keywords and view search results | Pass | Entered keywords in search bar, received list of available options |

**Feature: Place Order**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Confirm customer registration | System should confirm customer registration if already registered | Pass | Customer registered, order placement process initiated |
| 2 | Register option for new customers | System should provide registration options for new customers | Pass | New customer received registration options and can continue placing an order |
| 3 | Select order type | Customer should be able to specify whether the order is for pick-up or delivery | Pass | Customer selected order type successfully |
| 4 | Provide delivery location | If order is for delivery, customer should provide a valid delivery location | Pass | Customer provided a valid delivery location |
| 5 | Display menu for specified date | System should display the menu for the specified date | Pass | Menu for the specified date displayed |
| 6 | Display available food items | Menu should display only available food items for that date | Pass | Menu displayed with only available food items |
| 7 | Specify number of units | Customer should be able to indicate the number of units for each menu item | Pass | Customer specified the desired number of units for each menu item |
| 8 | Order multiple identical meals | System should permit ordering multiple identical meals | Pass | Customer successfully ordered multiple identical meals |
| 9 | Maximum order limit | System should inform the customer of the maximum order limit for each food item | Pass | Customer notified of the maximum order limit for each food item |
| 10 | Edit or cancel order | If order cannot be fulfilled, customer should have options to edit or cancel the order | Pass | Customer able to edit or cancel the order as required |
| 11 | Display order details | System should display the ordered food items, prices, and payment amount | Pass | Order details displayed accurately |
| 12 | Prompt order confirmation | System should prompt the customer to confirm the meal order | Pass | Customer prompted to confirm the order |
| 13 | Order not confirmed | If customer does not confirm the meal order, they can edit or cancel the order | Pass | Customer able to edit or cancel the order |

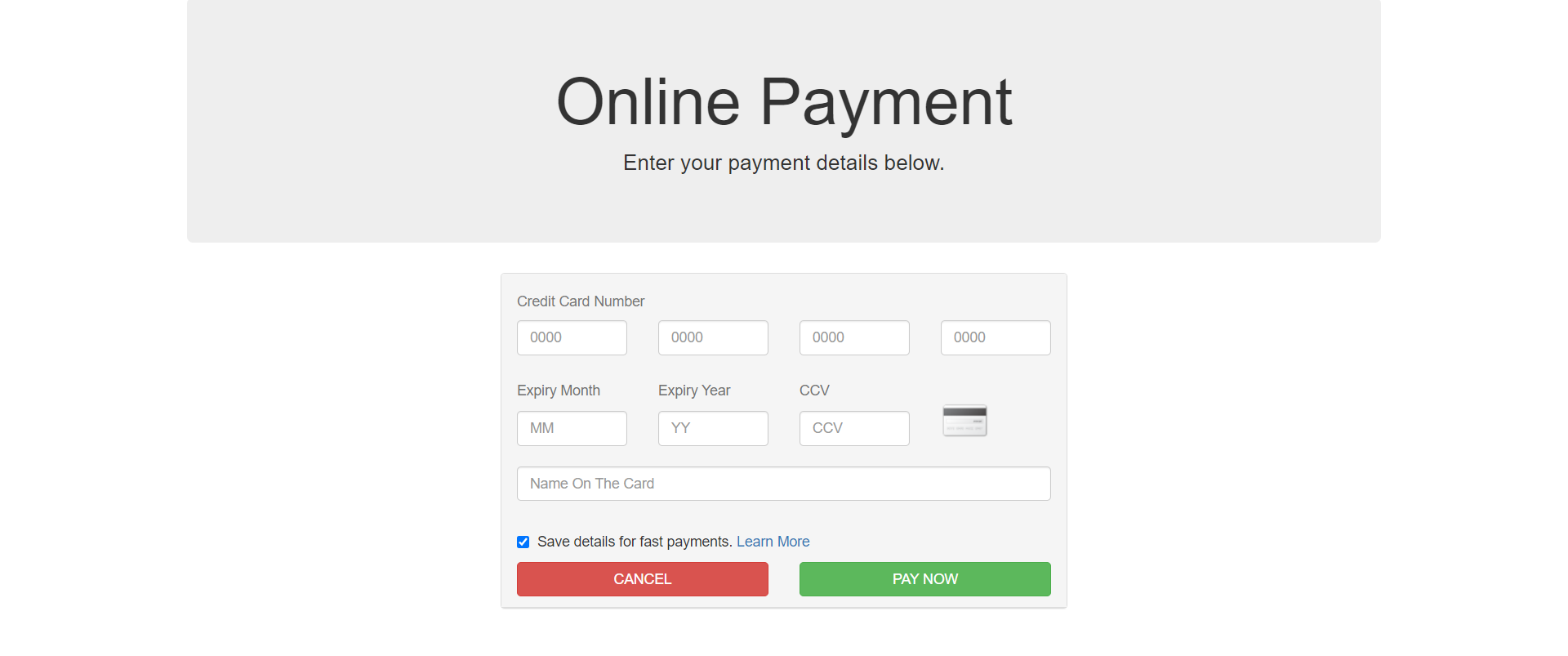






**Feature: Payment Option**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Select payment method | System should prompt the user to select a payment method | Pass | User prompted to select a payment method |
| 2 | Credit card payment option | If meal is for pick-up, user should be able to choose credit card payment | Pass | User successfully selected credit card payment option |
| 3 | Cash payment option | If meal is for pick-up, user should be able to choose cash payment | Pass | User successfully selected cash payment option |

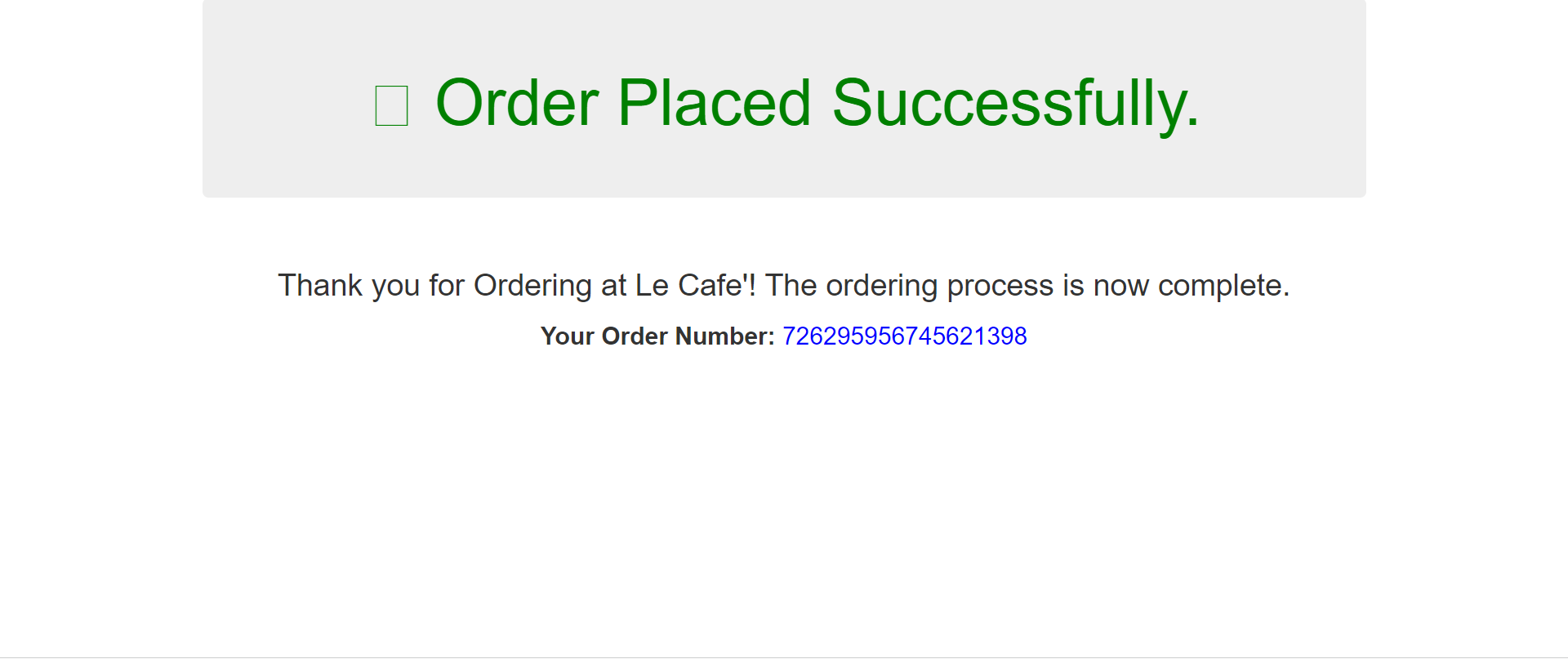


**Feature: Order Payment Deduct**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Issue payment request | If user confirms the order and selects credit card payment, system should issue a payment request to the Bank system | Pass | Payment request successfully issued to the Bank system |

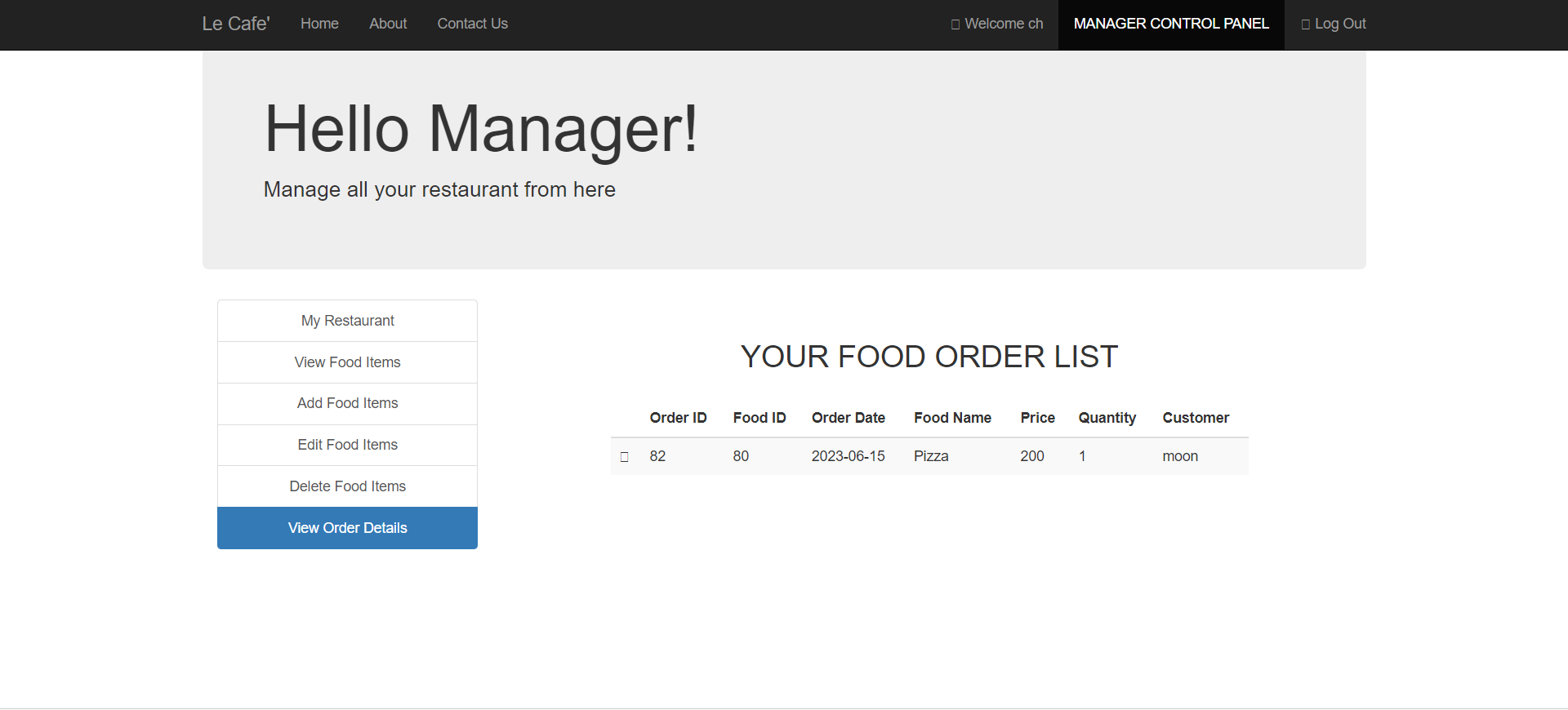
**Feature: Order Payment Confirm**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Confirm payment acceptance | If payment request is accepted, system should display a confirmation message with the payment deduction transaction number | Pass | Payment acceptance confirmed with transaction number |



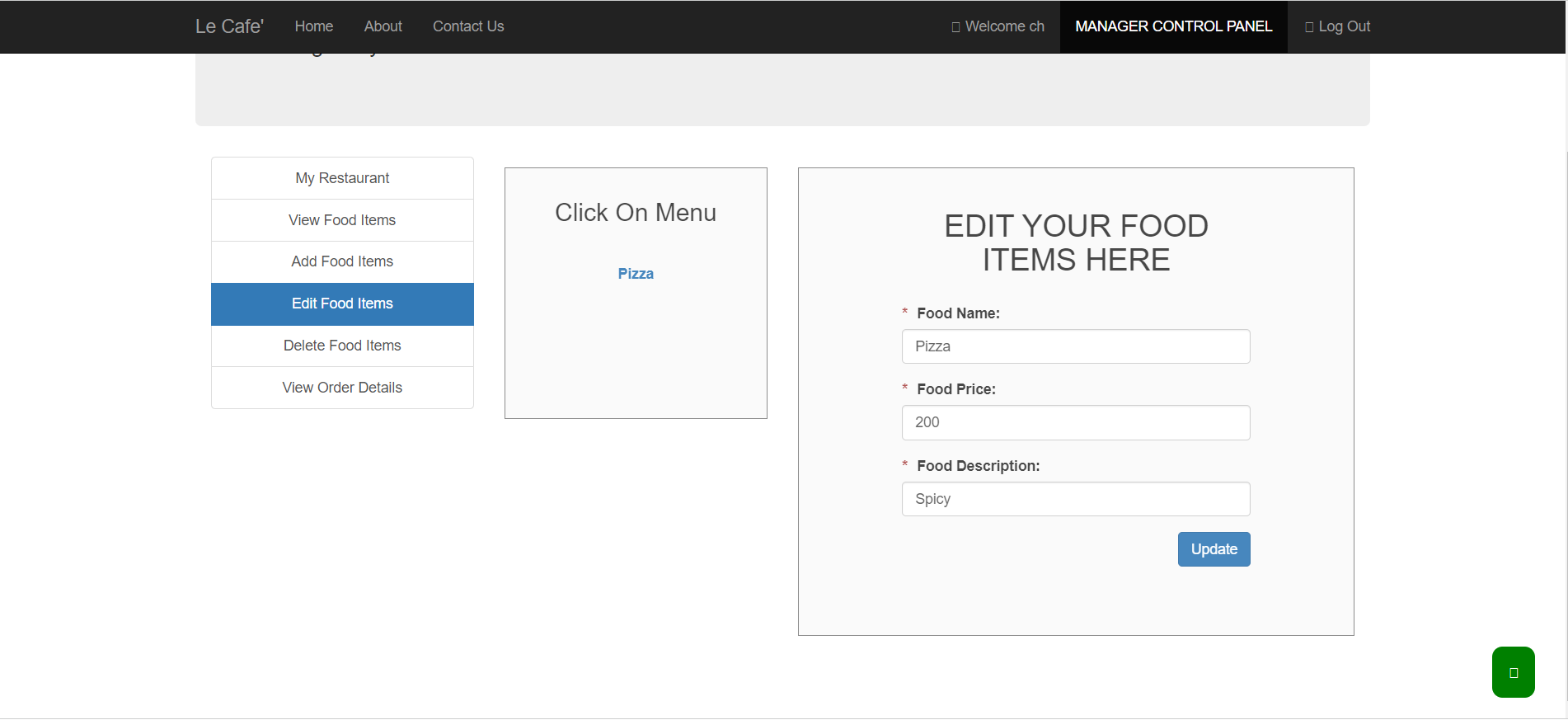
**Feature: Order Message to Cafeteria Staff**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Send order message | System should send a message to the Cafeteria Inventory System and cafeteria chef with the number of units of each food item in the order | Pass | Order message successfully sent to the Cafeteria Inventory System and cafeteria chef |



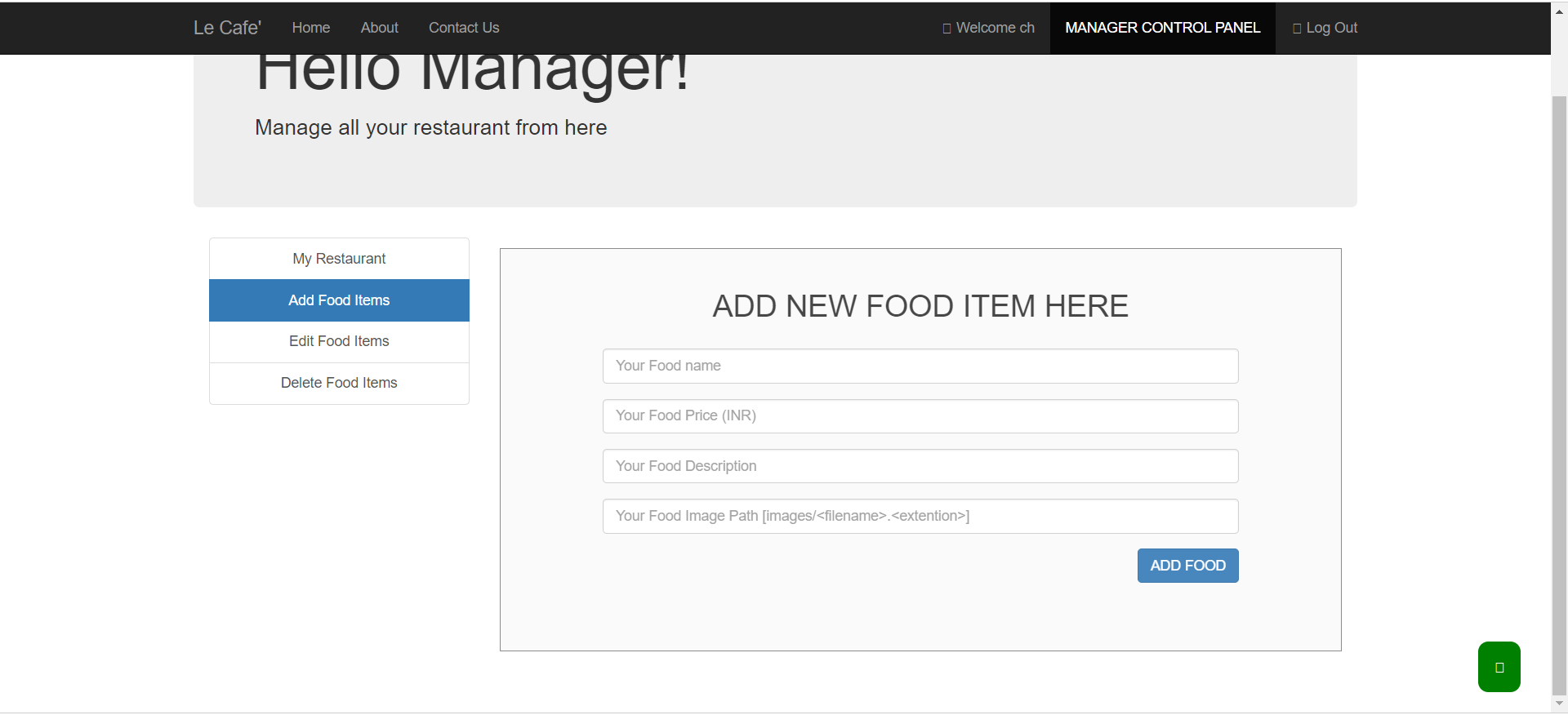
**Feature: Update Menu and Inventory**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Update menu and inventory | System should update the menu for the current order's date to reflect any items that are now out of stock in the cafeteria inventory | Pass | Menu and inventory successfully updated |



**Feature: Create Menu**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Create menu option | System should provide an option for the manager to create a menu | Pass | Create menu option available |
| 2 | Add menu details | Manager should be able to add menu details while creating the menu | Pass | Menu details added successfully |

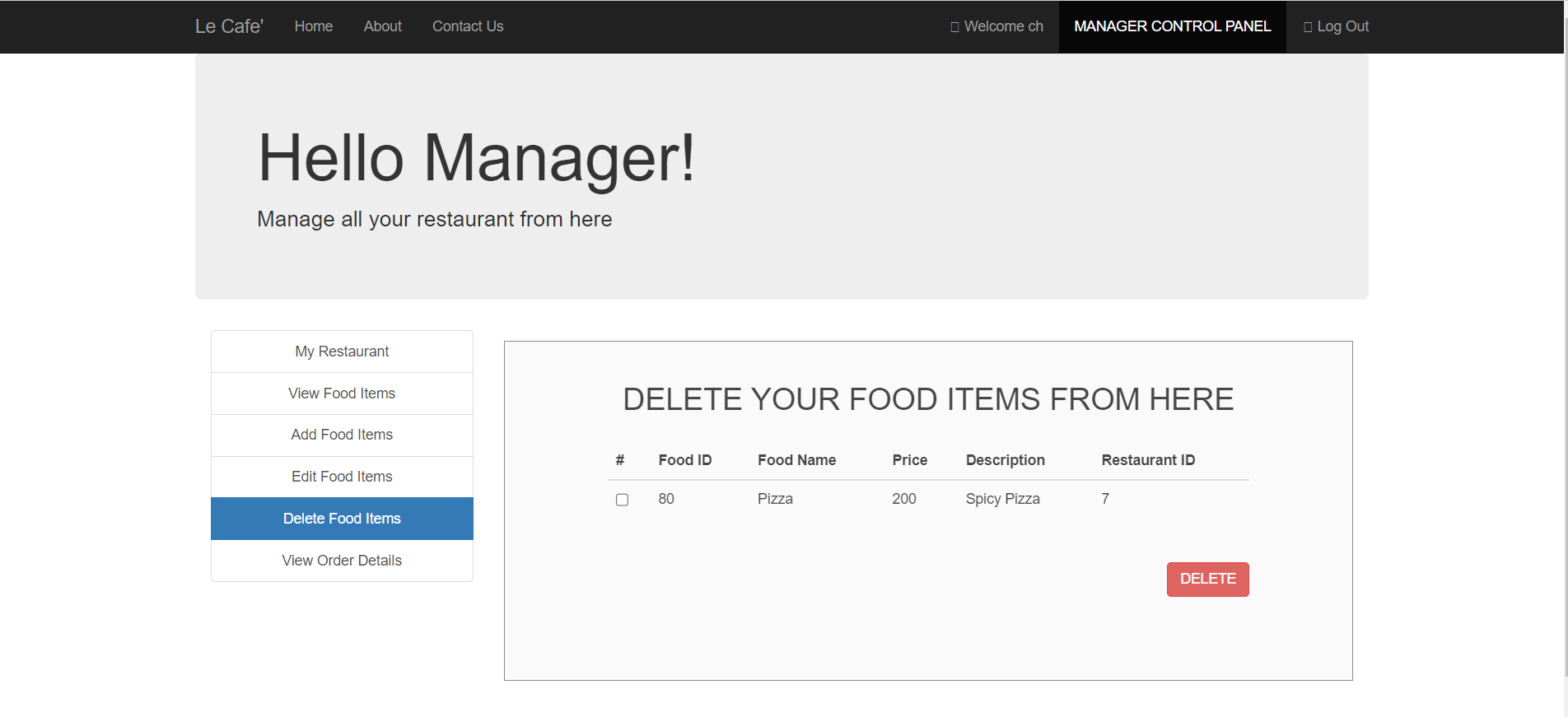


**Feature: Modify Menu**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Modify menu option | System should provide an option to modify an existing menu | Pass | Modify menu option available |
| 2 | Update menu details | Manager should be able to update the menu details | Pass | Menu details updated successfully |

**Feature: Delete Menu**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Delete menu option | System should provide an option to delete an existing menu | Pass | Delete menu option available |
| 2 | Confirm menu deletion | Manager should be able to confirm the deletion of the menu | Pass | Menu deletion confirmed successfully |



**Feature: Menu View**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | View modified menu | After updating the menu, user should be able to view the modified menu in the application | Pass | Modified menu displayed successfully |

**Feature: Menu Save**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Save menu after update | If the menu is updated and not saved, the updated menu should not be displayed and the user should be informed to re-update the details | Pass | User prompted to save the updated menu before viewing it |

**Feature: Display all orders**

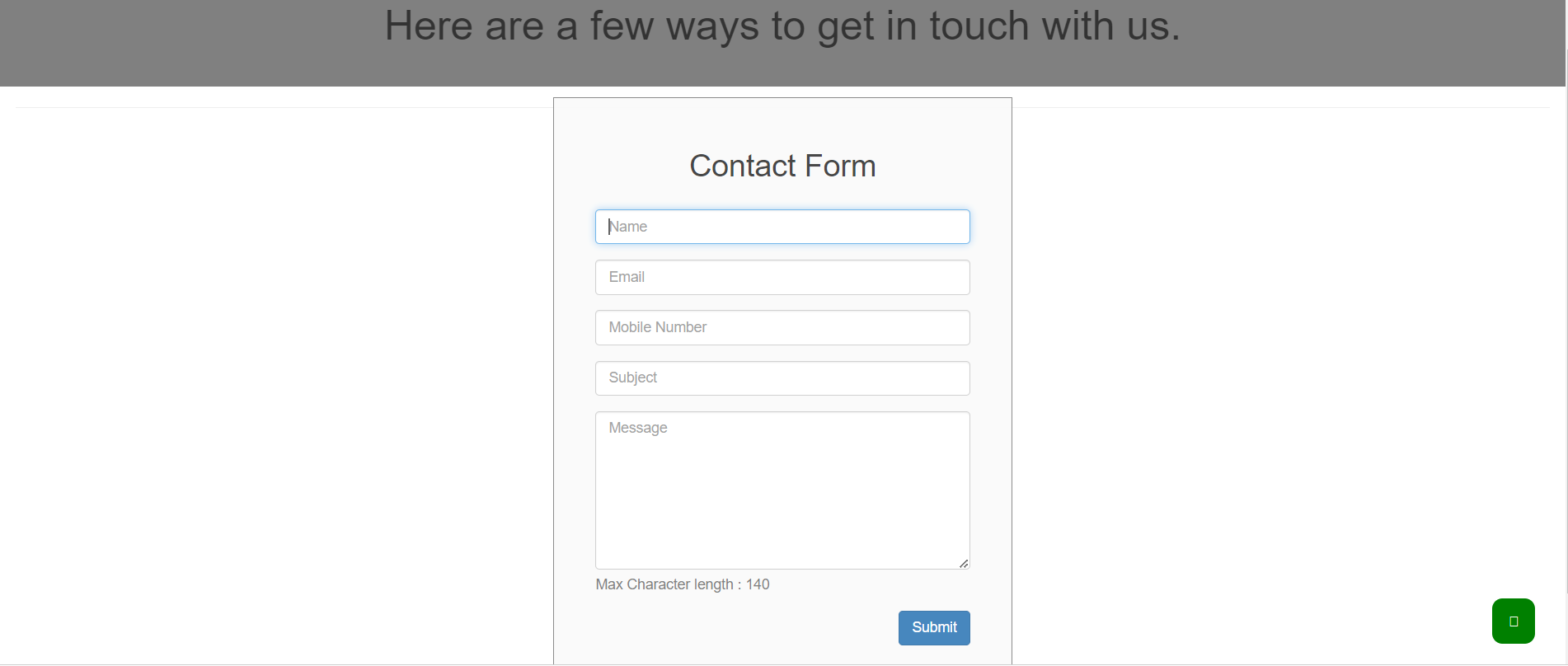
| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Display all orders | The system should display all the orders to the chef | Pass | All orders are displayed to the chef |

**Feature: Display Status**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Display order status | The system should enable the chef to display the status of each order, either "Preparing order" or "Ready to pick up" | Pass | Order status is displayed correctly for each order |

**Feature: Feedback Submission**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Submit feedback | Registered user should be able to submit feedback on the app | Pass | Feedback successfully submitted by the registered user |
| 2 | Include detailed explanation | Feedback should allow the user to provide a detailed explanation of their problem, if any | Pass | Detailed explanation included in the submitted feedback |



**Feature: Display Feedback to Admin**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Display submitted feedback | Admin should be able to view the submitted feedback | Pass | Admin can view the submitted feedback |
| 2 | Differentiate open and closed feedback | Admin should be able to identify whether a feedback is open or closed | Pass | Open and closed feedback is properly differentiated |

**Feature: Admin Response**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Post reply to feedback | Admin should be able to post a reply to the feedback | Pass | Admin successfully posted a reply to the feedback |

**Feature: Login With Facebook**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Register with Facebook | User should have the option to register with COS application through Facebook | Pass | User successfully registers with COS using Facebook |
| 2 | Share activities on Facebook | User should be able to share activities with COS on Facebook, such as "Ordered and Enjoyed Chicken Biriyani from Cafeteria - Simply Delicious" | Pass | User's activities are successfully shared on Facebook |

**Feature: Order Custom Meals**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Order custom meal | User should be able to order a new meal that is not available in the cafeteria menu | Pass | User successfully places an order for a customized meal |
| 2 | Provide style and ingredients | User should be able to provide the style of the food and the ingredients to customize the order | Pass | User includes the desired style and ingredients for the custom order |

**Feature: Track Order Status**

| **Test Case** | **Name** | **Requirement** | **Status** | **Actual Result** |
| --- | --- | --- | --- | --- |
| 1 | Track order status | User should be able to track the status of their meal order | Pass | User successfully tracks the status of their meal order |

---------------------------------------------------------------------------------------------------------------------