American International University-Bangladesh (AIUB)

Department of Computer Science

Faculty of Science & Technology (FST)

Software Engineering

Project Title

Meat Bazar

Submit to

MD. RAIHAN MAHMUD

Submit by:

Name	Id
MD.Nazmus Sadat Numan	22-48497-3
MD Doad	22-48506-3
Hasibul Islam Hasib	22-48507-3
Redwanul Haque Mazumder	22-48481-3

Project Name: Meat Bazar	Test Designed by:
Test Case ID: DD_01	Test Designed date:
Test Priority (Low, Medium, High): High	Test Executed by:
Module Name: Order Overview	Test Execution date:

Test Title: Verify that distributor can view pending, in-progress, and completed orders

Description: See pending, in-progress, and completed orders

Precondition (If any): Distributor need to login and go to dashboard

Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the Order Overview page		Distributor should see lists of orders as Pending, In-Progress, and Completed		

Project Name: Meat Bazar	Test Designed by:
Test Case ID: DD_02	Test Designed date:
Test Priority (Low, Medium, High): High	Test Executed by:
Module Name: Meat Stock Levels	Test Execution date:

Test Title: Verify that meat stock levels and expected deliveries are displayed

Description: Test meat inventory tracking

Precondition (If any): Distributor need to login and go to dashboard

Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)

1. Go to the Meat Stock Levels page		Distributor should see available meat quantities and upcoming deliveries		

Project Name: Meat Bazar		Test Designed by:		
Test Case ID: DD_03		Test Designed date:		
Test Priority (Low, Medium, High): High		Test Executed by:		
Module Name: Customer Orders		Test Execution date:		
Test Title: Verify that user can view new and pending customer orders in real time				
Description: : View new and pending customer orders				
Precondition (If any): Distributor need to login and go to dashboard				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the Customer Orders page		Distributor should see real-time list of new and pending orders		

Project Name: Meat Bazar	Test Designed by:
Test Case ID: DD_04	Test Designed date:
Test Priority (Low, Medium, High): High	Test Executed by:
Module Name: Payment Tracking	Test Execution date:
Test Title: Verify that user can track payments made to collectors and received from customers	

Description: Track payments made to collectors and received from customers.				
Precondition (If any): Distributor need to login and go to dashboard				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the Payment Tracking page		Distributor should see records of payments made and received		

Project Name: Meat bazar		Test Designed by:		
Test Case ID: DD_05		Test Designed date:		
Test Priority (Low, Medium, High): High		Test Executed by:		
Module Name: Notification and alerts		Test Execution date:		
Test Title: Verify that distributor receives alerts for new orders, payment confirmations, and inventory updates				
Description: Test alert/notification system				
Precondition (If any): Distributor need to login and go to dashboard				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
Login to the system		Notification icon or section becomes visible		
Trigger a new order/payment/inventory update from another account	New order placed	Notification is received instantly by the user		
Click on the notification		User is redirected to relevant order/payment/inventory page		

Project Name: Meat Bazar		Test Designed by:		
Test Case ID: DD_06		Test Designed date:		
Test Priority (Low, Medium, High): High		Test Executed by:		
Module Name: Sales & Performance Reports		Test Execution date:		
Test Title: Verify that sales and performance reports display correctly				
Description: Test insights including total sales, top items, and trends				
Precondition (If any): Distributor need to login and go to dashboard				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
Go to the Reports section		User sees a dashboard with sales insights		
View total sales and revenue		Correct total values are shown		
Check most ordered items		A list of top-selling items is displayed		
View revenue trends		Graph/chart shows revenue over time		

Project Name: Meat Distribution System	Test Designed by: Hasib
Test Case ID: HH_01	Test Designed date: 5/5/25
Test Priority (Low, Medium, High): High	Test Executed by: Hasib
Module Name: Collector Management Dashboard	Test Execution date: 5/5/25
Test Title: Verify Functionality of the Collector Management Dashboard	
Description: This test validates the core functionalities of the Collector Management Dashboard, including tracking incoming distributor requests, managing pending orders, checking payment statuses, reviewing meat supply history, receiving real-time notifications, and viewing financial overviews.	
Precondition (If any): 1. The collector management system is set up and connected to a live database.	

2. Test data should be available for distributors, orders, and payments. 3. Test users (distributors, admins) have been set up in the system.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
Login to the Dashboard Verify Distributor Requests Check Pending Orders Verify Payment Status Logout from the Dashboard	Distributor User: test User Valid login credentials: test User Distributors: Distributor A, Distributor B, Distributor C Sample Requests: Meat request for 10kg beef, 5kg chicken Orders: place an order.	The login page accepts valid credentials and grants access to the dashboard. The Distributor Requests section displays incoming meat requests with correct details. Pending Orders are displayed with accurate order ID, details, and status. The Payment Status section shows accurate details for all payments (paid/pending).		
Post Condition: Collector Dashboard tests verify accurate display of distributor requests, pending orders, payment status, meat supply history, real-time notifications, and financial overview. UI is responsive, data matches backend, and no errors occur.				

Project Name: Meat Distribution System			Test Designed by: Hasib	
Test Case ID: HH_02			Test Designed date: 5/5/25	
Test Priority (Low, Medium, High): High			Test Executed by: Hasib	
Module Name: Order Management (Functionality)			Test Execution date: 5/5/25	
Test Title: Verify Collector Can View Distributor Requests				
Description: Ensure that a collector can view all supply requests submitted by distributors in the system dashboard.				
Precondition (If any): 1. Collector must be logged into the system. 2. At least one distributor request must exist.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)

Login as Collector Navigate to Distributor Requests' section Select a request from the list	Collector Credential. N/A A request ID.	Collector dashboard loads successfully Distributor request list is displayed Request details (quantity, distributor, date) are visible.	AS EXPECTED,	PASS
Post Condition: 1. The collector remains logged in and can proceed to process the request. 2. The system returns to the collector dashboard, ready for the next action.				

Project Name: Meat Distribution System			Test Designed by: Hasib	
Test Case ID: HH_03			Test Designed date: 5/5/25	
Test Priority (Low, Medium, High): Medium			Test Executed by: Hasib	
Module Name: Order Management (Non-Functionality)			Test Execution date: 5/5/25	
Test Title: Verify Instant Notification for New Distributor Orders				
Description: To validate that collector, receive immediate alerts when a new order is placed by any distributor.				
Precondition (If any): 1. The collector is logged in and the notification system is active. 2. A distributor has access to place an order.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
Login as Collector In parallel, as Distributor and place a new order Observe collector's interface.	Collector credentials Place an Order ID. N/A	Dashboard loads Order submitted successfully Collector receives real-time notification popup with order details	AS EXPECTED,	PASS
Post Condition: 1. Collector receives real-time alert/notification popup with order details				

Project Name: Meat Distribution System	Test Designed by: Hasib
Test Case ID: HH_04	Test Designed date: 5/5/25
Test Priority (Low, Medium, High): High	Test Executed by: Hasib
Module Name: Order Management (PDC)	Test Execution date: 5/5/25

Test Title: Verify Secure Transaction Data Handling via Encryption				
Description: To ensure that financial transaction data (payment slips, confirmations) is encrypted and securely transmitted/stored.				
Precondition (If any): 1. Collector and Distributor are authenticated users. 2. Payment system integration is active. 3. A valid order has been placed and is ready for payment.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
Login as Distributor Place an order and proceed to payment Submit payment Use network traffic analyzer Access backend database with admin role	Valid credentials Place an Order ID. Payment info: card/bank transfer. N/A Payment table	Distributor dashboard loads. Order placed and payment initiated. Payment is processed successfully. Payment data is encrypted during transmission. Financial records are encrypted at rest.	AS EXPECTED,	PASS
Post Condition: 1. Payment is securely logged and available for reconciliation by the collector.				

Project Name: Meat Distribution System			Test Designed by: Hasib	
Test Case ID: HH_05			Test Designed date: 5/5/25	
Test Priority (Low, Medium, High): High			Test Executed by: Hasib	
Module Name: Signup			Test Execution date: 5/5/25	
Test Title: Validate Signup Performance, Security, UI/UX, and Role Compliance				
Description: Ensure the signup process is fast, secure, responsive, intuitive, and meets compliance and integration constraints.				
Precondition (If any): 1. Signup page is live and accessible. 2. Backend services (OTP, email, encryption, and role APIs) are operational. 3. Test environment includes desktops, tablets, and mobile devices.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)

Open signup page on desktop, tablet, and mobile browsers. Fill in the signup form with valid data. Submit signup form. Review interface navigation and usability. Verify privacy policy and data terms presence.	N/A Input role and password. Keep waiting. Manual review N/A	Page loads correctly. Form accepts input. Signup complete. Form clearly labeled. Data protection law confirmed.	AS EXPECTED,	PASSED
Post Condition: 1. User is registered, assigned a valid role, securely stored in the system, and redirected to their respective dashboard.				

Project Name: Meat Distribution System			Test Designed by: Hasib	
Test Case ID: HH_06			Test Designed date: 5/5/25	
Test Priority (Low, Medium, High): High			Test Executed by: Hasib	
Module Name: Forgot Password			Test Execution date: 5/5/25	
Test Title: Validate Complete Forgot Password Flow with Security and Performance				
Description: To verify that the password reset process is secure, functional, user-friendly, and meets performance expectations across all platforms.				
Precondition (If any): 1. The user must be previously registered in the system. 2. The email/SMS gateway (e.g., Firebase or Twilio) must be active and configured.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
Navigate to the "Forgot Password" page on desktop, tablet, and mobile. Enter registered email or phone number.	N/A Enter valid email	Page loads correctly. Input is accepted.	AS EXPECTED	PASS

Submit request and start timer. Enter received OTP and set a new password. Log in with new password.	N/A Enter OTP N/A	OTP or reset link is delivered within 30 seconds. Password field validates strength and format Success message appears.		
Post Condition: 1. User's password is successfully updated, and login with the new password is verified.				

Project Name: User Requirements Dashboard Automation Testing			Test Designed by: Redwanul	
Test Case ID: MB-LOGN-01			Test Designed date: 5/5/25	
Test Priority (Low, Medium, High): Medium			Test Executed by: Redwanul	
Module Name: Login Session (or User Authentication)			Test Execution date: 5/5/25	
Test Title: Verify login with valid username and password				
Description: Test website login page				
Precondition (If any): 1. User must have valid username and password (if any). 2. The website must be accessible and operational. 3. The user must be on the login page of the website.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Go to the website. 2. Enter username: testuser@gmail.com 3.Enter password: Test@1234 4. Click Submit.	Username: testuser@gmail.com Password: Test@1234	User should login into the application (as expected).	Loaded in 3 sec	Pass
Post Condition: User is validated with database and successfully logs into account.				

Project Name: Order History (Functionality)	Test Designed by: Redwanul
Test Case ID: MB-OH_02	Test Designed date: 5/5/25
Test Priority (Low, Medium, High): Medium	Test Executed by: Redwanul
Module Name: Functional Requirements for Order History	Test Execution date: 5/5/25

Test Title: Verify Viewing of Order History with Valid User Credentials Rationale				
Description: Test the functionality of the Order History feature on the Meat Bazar dashboard to ensure users can successfully view their purchase history, including details such as type of meat, quantity, and total payment, and navigate back to the dashboard using the "Back" button.				
Precondition (If any): 1. The user must be logged into the Meat Bazar platform with valid credentials (e.g., email/phone and password). 2. The user must have at least one completed order in their purchase history stored in the database. 3. The dashboard must be accessible and operational, with the Order History feature enabled.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1.Log into the Meat Bazar platform using valid user credentials. 2.Navigate to the dashboard. Click on the "Order History" option to access the purchase history. 3.Verify that the purchase history details are displayed (type of meat, quantity, total payment). 4.Click the "Back" button to return to the dashboard.	Username: testuser@gmail.com (or phone: 1234567890, depending on login setup) Password: Test@1234	1.The user successfully logs into the Meat Bazar platform and lands on the dashboard. 2. The "Order History" option is accessible and clickable on the dashboard. 3. Clicking the "Back" button returns the user to the dashboard.	Pass (assuming the system behaves as expected based on the requirements and test data provided).	PASS
Post Condition: 1. The user successfully views their purchase history, including details such as type of meat, quantity, and total payment, and returns to the dashboard after clicking the "Back" button.				

Project Name: Order Tracking (Functionality)	Test Designed by: Redwanul
Test Case ID: MB-OT-03	Test Designed date: 5/5/25
Test Priority (Low, Medium, High): Medium	Test Executed by: Redwanul

Module Name: functionality of the Order Tracking feature			Test Execution date: 5/5/25	
Test Title: Verify Order Tracking with Valid Order ID				
Description: Test the functionality of the Order Tracking feature on the Order Tracking dashboard to ensure users can successfully track the status of their order (e.g., Processing, Pending, Picked Up) using a valid order ID, view rider details if the order is picked up, and navigate back to the dashboard using the "Back" button.				
Precondition (If any): • The user must be logged into the Order Tracking platform with valid credentials (e.g., email/phone and password). • The user must have at least one active or recently placed order in the system with a valid order ID. • The dashboard must be accessible and operational, with the Order Tracking feature enabled.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1.Log into the Order Tracking platform using valid user credentials. 2.Navigate to the dashboard. 3. Click on the "Order Tracking" option to access the order tracking interface. 4. Enter or select a valid order ID to track the order. 5.Verify that the order status is displayed (e.g., Processing, Pending, Picked Up). 6. If the order status is "Picked Up," verify that the rider's name and phone number are displayed. □ Click the "Back" button to return to the dashboard.	1.Username: testuser@gmail.com 2.Password: Test@1234 3.Order ID: ORDER12345 4.Sample Order Details in Database: • Order ID: ORDER12345, Status: Picked Up, Rider Name: (Ayon) and phone number (01303651303) • (Alternative scenario: Order ID: ORDER67890,	The user successfully logs into the Order Tracking platform and lands on the dashboard. The "Order Tracking" option is accessible and clickable on the dashboard. After entering or selecting the order ID (ORDER12345), the order status "Picked Up" is displayed. Since the status is "Picked Up," the rider's name (Ayon) and phone number (01303651303)are displayed.	Pass (assuming the system behaves as expected based on the requirements and test data).	PASS

	Status: Processing, Rider Details: None)	Clicking the "Back" button returns the user to the dashboard.		

Post Condition:

1. The user successfully tracks the order status using the valid order ID, views the rider's name and phone number if the order is picked up, and returns to the dashboard after clicking the "Back" button.

Project Name: Meat Ordering	Test Designed by: Redwanul
Test Case ID: MB_MO-04	Test Designed date: 5/5/25
Test Priority (Low, Medium, High): Medium	Test Executed by: Redwanul
Module Name: functionality of the Meat Ordering feature	Test Execution date: 5/5/25
Test Title: Verify Meat Ordering with Valid Selection and Payment Initiation	
Description: Test the functionality of the Meat Ordering feature on the Meat Ordering dashboard to ensure users can successfully select a meat type (Beef, Mutton, or Chicken), specify a quantity (in kg or full animal), view pricing, initiate payment, and navigate back to the dashboard using the "Back" button.	
Precondition (If any): • The user must be logged into the Meat Ordering platform with valid credentials (e.g., email/phone and password). • The user must have sufficient funds, or a valid payment method configured to initiate payment. • The dashboard must be accessible and operational, with the Meat Ordering feature enabled.	

Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1. Log into the Meat Ordering platform using valid user credentials. 2.Navigate to the dashboard. 3. Click on the "Meat Ordering" option to access the ordering interface. 4. Select a meat type (e.g., Beef). 5.Specify a quantity (e.g., 2 kg).	1.Username**:** testuser@gmail.com 2. Password: Test@1234 3. Meat Type: Beef 4. Quantity: 2 kg 5. Pricing: 500 BDT (assuming 250 BDT per kg for Beef) 6. Payment Method: bKash (configured and funded)	1.The user successfully logs into the Meat Ordering platform and lands on the dashboard. 2. The "Meat Ordering" option is accessible and clickable on the dashboard.	Pass (assuming the system behaves as expected based on the requirements and test data).	PASS

6. Verify that the pricing is displayed based on the selected quantity. 7. Click the "Payment" button to initiate the payment process. 8. Click the "Back" button to return to the dashboard.		3. After selecting Beef and specifying 2 kg, the pricing (500 BDT) is displayed. 4. Clicking the "Payment" button initiates the payment process and displays the payment interface with bKash selected. 5. Clicking the "Back" button returns the user to the dashboard.		
Post Condition: 1. The user successfully selects a meat type, specifies a quantity, views the pricing, initiates the payment process, and returns to the dashboard after clicking the "Back" button.				

Project Name: Security			Test Designed by: Redwanul	
Test Case ID: MB_Payment-05			Test Designed date: 5/5/25	
Test Priority (Low, Medium, High): Medium			Test Executed by: Redwanul	
Module Name: Login Security			Test Execution date: 5/5/25	
Test Title: Verify Secure Login with Encrypted Credentials				
Description: This test case verifies that the login page of the Meat Bazar application securely authenticates users by encrypting credentials during transmission and properly handles valid/invalid login attempts. The system should prevent unauthorized access and display appropriate error messages for incorrect credentials, ensuring compliance with security requirements.				
Precondition (If any): • The Meat Bazar application is deployed and accessible on a web browser. • A test user account exists in the Firebase Authentication database with the following credentials: • Email: testuser@gmail.com • Password: Test@1234 • The browser's developer tools are open to monitor network traffic for encryption (e.g., HTTPS usage). • The system is configured to use Firebase Authentication as specified in the project constraints.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)

1.Navigate to the Meat Bazar login page. 2.Enter the test user's email: 3.Enter the test user's password: testuser@gmail.com Secure Pass Test@1234 4.Use browser developer tools to confirm that the data transmission uses HTTPS. 5.Click the "Login" button. 6.Attempt a second login with incorrect credentials (Email: testuser@gmail.com , Password: WrongPass123). 7.Observe the system's response to the incorrect login attempt.	Valid Credentials: Email: testuser@gmail.com Password: Test@1234 Invalid Credentials: Email: testuser@gmail.com Password: WrongPass123	1.The login page loads within 3 seconds, as per non-functional requirements. 2.Data transmission is encrypted (HTTPS is used, no plain text credentials visible in network traffic). 3.For valid credentials, the user is redirected to their respective dashboard (e.g., Customer Dashboard). 4.For invalid credentials, an error message is displayed: "Incorrect email or password. Please try again." 5.The system does not allow access to the dashboard with invalid credentials.	Pass (assuming the system behaves as expected based on the requirements and test data).	PASS
Post Condition: • The test user is logged into the system with valid credentials and can access the Customer Dashboard. • The system remains secure, with no unauthorized access granted during testing. • Network traffic logs confirm that all sensitive data (email and password) was transmitted securely via HTTPS.				

Project Name: Responsiveness	Test Designed by: Redwanul
Test Case ID: MB-Responsiveness-06	Test Designed date: 5/5/25
Test Priority (Low, Medium, High): Medium	Test Executed by: Redwanul
Module Name: Responsive Login Interface	Test Execution date: 5/5/25
Test Title: Verify Responsive Design of Login Page Across Devices	
Description: This test case verifies that the login page of the Meat Bazar application maintains a responsive design across various devices (desktop, tablet, and mobile), ensuring it loads within 5 seconds and adapts seamlessly to different screen sizes. The test focuses on usability and performance to confirm a consistent, responsive user experience as per the project's non-functional requirements.	
Precondition (If any): 1. The Meat Bazar application is deployed and accessible on a web browser. 2. A test user account exists in the Firebase Authentication database with the following credentials: a. Email: testuser@gmail.com b. Password: Test@1234 3. The browser's developer tools are open to monitor network traffic for encryption (e.g., HTTPS usage). 4. The system is configured to use Firebase Authentication, ensuring secure data transmission as specified in the project constraints. 5. Testing devices are prepared: a desktop (1920x1080 resolution), a tablet (iPad, 768x1024 resolution), and a mobile phone (iPhone, 375x667 resolution).	

Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1.Open the Meat Bazar login page on the desktop browser and measure the load time. 2.Resize the browser to tablet resolution (768x1024) and measure the load time. 3.Resize the browser to mobile resolution (375x667) and measure the load time. 4.Enter valid credentials (Email: testuser@meatbazar.com, Password:	• Valid Credentials: Email: testuser@gmail.com Password: WrongPass123 • Device Resolutions: Desktop: 1920x1080 Tablet: 768x1024 Mobile: 375x667	• The login page loads within 3 seconds on all devices (desktop, tablet, mobile). • The interface adjusts responsively to each resolution, with no overlapping elements, excessive scrolling, or usability issues.	Pass (assuming the system behaves as expected based on the requirements and test data).	Pass

SecurePass123!) on each device and verify navigation to the dashboard. 5.Rotate the mobile view (portrait to landscape) and confirm the interface adjusts without errors. 6.Use browser developer tools to ensure the layout adapts responsively (no overlapping elements or excessive scrolling).		• Navigation to the dashboard occurs smoothly after entering valid credentials on all devices. • The mobile view rotates from portrait to landscape seamlessly, maintaining a responsive layout.		
Post Condition: • The login page remains responsive across all tested devices, with no changes to its adaptive layout. • Users can access the dashboard seamlessly on any device, ensuring consistent and responsive experience. • The interface continues to adjust dynamically to screen size changes without affecting usability.				

Project Name: Meat Bazar			Test Designed by:	
Test Case ID: SN_01			Test Designed date:	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Update Meat Inventory			Test Execution date:	
Test Title: Update the availability of meat type (Beef, Mutton, Chicken), set Price of meat.				
Description: Update the availabity of meat type (Beef, Mutton, Chicken) in kg or full size (Cow, Goat, Chicken), set and update daily meat prices for customers.				
Precondition (If any): Check the (Cow, Goat, Chicken) if it is in good health condition or not and check the market and government price for daily market then update or set the price.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
Go to the admin dashboard Go to Meat Inventory Update price Update availability	100 kg of Beef Price per kg 600 tk. 60 kg of mutton price per kg 800 tk. 200 kg of chicken price per kg 180 tk.	The availability of meat and price should be updated. The customer can see this update.	As expected,	Pass
Post Condition: The admin can successfully update the price and meat inventory, and the data must be added to the database.				

Project Name: Meat Bazar			Test Designed by:	
Test Case ID: SN_02			Test Designed date:	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Manage Users			Test Execution date:	
Test Title: View, Add, Remove users; View order history; Modify customer orders				
Description: Admin can manage all users (Collectors, Distributors, Customers). This includes viewing all users, adding new users, removing existing users, viewing customer order history, and modifying customer orders.				
Precondition (If any): Admin must be logged in with proper authorization. Users being added must provide valid details.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1.Go to the Admin Dashboard 2.Navigate to "Manage Users" 3.View the list of all users (Collectors, Distributors, Customers) 4. Click "Add New User" and enter valid details 5. Click "Remove" on a specific user 6.Click on a customer's name to view their order history 7.Select an order and choose "Cancel" or "Keep" 8.Confirm the update	New Collector: Name – Akash, Role – Collector, Phone – 017xxxxxxx, Region – Gazipur Remove User: Distributor – Jamal Hossain View Order History: Customer – Sumi Akter Modify Order: Order ID – 4567, Status – Cancel	Admin should be able to see all registered users Admin can successfully add new users and see them in the list Admin can remove users and they no longer appear in the system Admin can view customer order histories Admin can cancel or keep orders successfully and status should update accordingly	As expected	Pass
Post Condition: User data and order modifications are updated in the database. Removed users no longer have access.				

Project Name: Meat Bazar	Test Designed by:
Test Case ID: SN_03	Test Designed date:
Test Priority (Low, Medium, High): High	Test Executed by:

Module Name: Allocate Meat to Collectors			Test Execution date:	
Test Title: Assign specific quantities of meat (kg or full animal) to collectors				
Description: Admin can allocate specific quantities of meat (in kg or full animals) to collectors. Admin selects a collector and assigns meat based on availability and distribution requirements.				
Precondition (If any): Admin must be logged in. Meat inventory must be updated and available in the system. Collectors must be already registered.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1.Go to the Admin Dashboard. 2.Navigate to the "Meat Allocation" section. 3.View available meat stock from the inventory. 4.Select a registered collector from the list. 5.Enter the quantity and type of meat to allocate (in kg or full animal). 6.Click the "Assign" or "Allocate" button to confirm allocation. 7.Receive confirmation message of successful allocation.	1.Collector Name: Akash Rahman 2.Meat Type: Beef Allocation Quantity: 150 kg 3.Meat Type: Goat Allocation Quantity: 2 full animals	1.Admin should be able to view the current meat stock. 2.Admin can select a collector and assign meat allocations successfully. 3.The allocation should be recorded and visible in the collector's dashboard. 4.The meat inventory should reflect the reduced quantity based on the allocation. 5.A success confirmation message should appear after assignment.	As expected	Pass
Post Condition: The meat allocation data is updated in the database. Collector can view their allocated quantities. Inventory is updated accordingly.				

Project Name: Meat Bazar	Test Designed by:
Test Case ID: SN_04	Test Designed date:
Test Priority (Low, Medium, High): High	Test Executed by:
Module Name: Manage Distributors	Test Execution date:
Test Title: View distributor activity and modify distributor records	

Description: Admin can manage distributor-related operations, including viewing their activity in order fulfillment and meat distribution, and modifying their records (such as contact information, assigned areas, or performance logs).				
Precondition (If any): Admin must be logged in. Distributors must already be registered in the system.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)
1.Go to the Admin Dashboard. 2.Navigate to the "Manage Distributors" section. 3.View the list of all registered distributors. 4.Click on a distributor's name to view their order fulfillment and meat distribution activity logs. 5.Click the "Edit" button to modify distributor information such as phone number, region, or status. 6.Submit the updates. Receive confirmation message of a successful update.	1.Distributor Name: Jamal Hossain 2.Previous Phone: 018XXXXXXXX 3.Updated Phone: 019XXXXXXXX 4.Region: Dhaka (no change) 5.Activity Log: Order ID 3421 – Delivered, Order ID 3510 – Pending	1.Admin should be able to access the list of all distributors. 2.Distributor activity logs related to order fulfillment and meat delivery should be visible. 3.Admin can successfully edit and update distributor details. 4.Updated information should be saved in the database and reflected in the system. 5.A success message should appear after updates are submitted.	As expected	Pass
Post Condition: Distributor records are updated in the system database, and activity logs remain accessible to the admin.				

Project Name: Meat Bazar			Test Designed by:	
Test Case ID: SN_05			Test Designed date:	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Payment History			Test Execution date:	
Test Title: View all customer transactions and track profits from Collectors and Distributors				
Description: Admin can monitor and review all payment transactions made by customers. The system also allows the admin to track profits generated from Collectors and Distributors, giving an overview of revenue flow.				
Precondition (If any): Admin must be logged in. Transactions must already exist in the system database.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)

1.Go to the Admin Dashboard. 2.Navigate to the "Payment History" section. 3.View the list of all payment transactions made by customers including date, amount, method, and order ID. 4.Switch to the "Profit Tracking" tab to view collected profit details from Collectors and Distributors. 5.Select a date range or filter by role (Collector/Distributor) to view profit summaries. 6.Review and analyze the total revenue and individual transaction details.	Customer Transaction: • Customer: Sumi Akter • Order ID: ORD_5609 • Amount: 4,800 BDT • Payment Method: bKash • Date: 02-05-2025 Profit Tracking: • Collector: Akash Rahman _ Collected: 35,000 BDT • Distributor: Jamal Hossain _ Delivered: 27,000 BDT • Date Range: 01-04-2025 to 02-05-2025	1.Admin should be able to view all customer payment records accurately. 2.Each transaction should include complete information such as amount, method, and date. 3.Admin can filter and view profit statistics for Collectors and Distributors within a given time period. 4.Profit summaries should display accurate totals, based on successful deliveries and collections.	As expected	Pass
Post Condition: Admin successfully views detailed financial records. Data remains saved in the system and available for future review and reporting.				

Project Name: Meat Bazar			Test Designed by:	
Test Case ID: SN_06			Test Designed date:	
Test Priority (Low, Medium, High): Medium			Test Executed by:	
Module Name: Admin Profile & Logout			Test Execution date:	
Test Title: View Admin Personal Info and Perform Logout				
Description: Admin can view their personal information, navigate back to the dashboard, and log out securely. Includes "Back" button functionality in both views.				
Precondition (If any): Admin must be logged in with valid credentials.				
Test Steps	Test Data	Expected Results	Actual Results	Status (Pass/Fail)

1.Go to the Admin Dashboard 2.Click on “Personal Info” 3.Click the “Back” button from the Personal Info page 4.Click on “Logout” from the dashboard 5.Click the “Back” button from the Logout page	1.Admin credentials: • Username: admin01 • Password: ******** 2.Expected session: Active session when accessing profile or logout.	1.Admin dashboard should be displayed 2.Admin's personal info should be shown (Name, Email, Phone, etc.) 3.Back button should return to dashboard 4.Logout should redirect to the login page 5.Back button from logout should return to dashboard if session is active	As expected	Pass
Post Condition: Admin can securely view personal info and log out. System ends session on logout and prevents access without re-authentication.				