Use Case Document: Online Grocery Store

Domain: Retail - E-Commerce

Use Case Name: Online Grocery Shopping App

Use Case Description:

This use case describes the process of purchasing groceries and household items through an online grocery store application. The customer can browse available products, search by name/category/brand, add items to cart, customize quantities, and proceed with payment for delivery or pickup. The system enables users to place, track, and manage orders in a seamless manner.

Use Case Goal:

To allow customers to browse products, search for specific items, manage their shopping cart, complete purchases, and track order status efficiently within the application.

Actors:

- Customer
- Admin/Store Manager
- Delivery Personnel

Pre-Conditions:

- Customer has internet access and a compatible device.
- Customer is registered and logged into the app (for checkout process).
- Customer has a valid payment method linked to their account.
- Customer is in an operational delivery zone.
- The store has sufficient inventory of requested items.

Post-Conditions:

- Customer's grocery order is successfully placed, confirmed, and sent to the store for processing.
- Customer receives order confirmation along with an estimated delivery time.
- Order details are updated in the customer's order history.
- Order details are sent to the delivery personnel (for delivery orders).
- Inventory is updated to reflect the purchased items.

Main Flow (Basic Path):

1. Browse: The customer opens the grocery app and views the home page with featured items.
2. Search: The customer searches for products by name, category, or brand using the search function.
3. Product Selection: The customer views product details and adds items to their shopping cart, specifying quantities as needed.
4. Cart Management: The customer reviews items in the cart, making adjustments to quantities or removing items as necessary.
5. Delivery/Pickup Option: The customer chooses delivery or pickup option, entering or confirming delivery address if required.
6. Payment: The customer proceeds to checkout, enters or confirms billing information, and selects a payment method.
7. Confirmation: The system confirms the order, provides an estimated delivery/pickup time, and displays an order reference number.
8. Processing: The store begins gathering the ordered items for delivery or pickup.
9. Status Updates: The system sends real-time notifications about the order status (e.g., processing, ready for pickup, out for delivery).
10. Completion: The customer receives the order at the specified address or picks it up from the store.

Alternative Flows:

1. Invalid Payment: If the payment fails (e.g., insufficient funds or card issues), the system displays an error message and prompts the customer to select another payment method or fix the issue.
2. Out-of-Stock Items: If a requested item is out of stock, the system notifies the customer and provides options to remove the item, choose a substitute, or continue with the remaining items.
3. Delivery Area Unavailable: If the customer's delivery address is out of range, the system notifies them and offers alternatives, such as pickup or changing delivery location.
4. Account Registration: If an unregistered customer attempts to checkout, the system prompts them to create an account or continue as a guest.
5. Order Tracking: Customer can view order history and track current order status from their account page.

Exception Handling:

1. Network Connectivity: If there is a network issue, the app should display an error message and allow customers to retry their action or save their cart for later.
2. Store Unavailability: If the store becomes unavailable for orders (e.g., closed, system maintenance), the customer should be notified with estimated time when service will resume.

3. Order Modifications: If a customer tries to modify an order after it's been processed, the system should notify them whether the modification is possible (before preparation) or not (once the order is already in preparation).
4. Payment Processing Errors: If there are issues with payment processing, the system should provide clear error messages and alternative payment options.

Possible Test Cases:

Test Case ID	Description	Input Data	Expected Outcome
TC01	Register a new account	Valid user information	Account created successfully with confirmation message
TC02	Search for products by name	Product name "Apples"	Display of all apple products available in store
TC03	Add items to the cart	Valid product selections	Items added to cart with correct quantities and total price
TC04	Proceed to checkout and complete purchase	Valid payment and shipping details	Order placed successfully with an order reference number
TC05	View order history	User login credentials	Display of past orders with status information
TC06	Track current order status	Order reference number	Real-time order status updates

TC07	Modify quantity in shopping cart	Update quantity from 1 to 3	Cart updated with new quantity and price
TC08	Remove item from shopping cart	Select remove option	Item removed and cart total updated
TC09	Fail payment due to insufficient funds	Invalid card details	Error message and prompt for another payment method
TC10	Search for non-existent product	Product not in inventory	"No results found" message with suggested alternatives
TC11	Select delivery to an out-of-range address	Address outside delivery zone	Error message with pickup option offered
TC12	Access order history without authorization	Unauthorized access attempt	Redirect to login page

Expected Outcome:

- Customers can easily browse products, search by various criteria, and place orders.
- The system provides a responsive experience across desktop and mobile devices.
- The application ensures smooth payment and delivery processes.
- Error handling is well defined, and the customer experience is seamless.
- The app provides clear communication on the status of orders.
- Customers can efficiently manage their accounts and view order history.
- The system maintains accurate inventory and order records.
-