
Project Title: *Fast-Food*

Course Title: Software Construction & Development

Department: Computing (*Software Engineering*)

Submitted To: Islam Abbasi

Submitted By:

56154	Umer Farooq
55330	Haider Nawaz Thabal

DESCRIPTIVE USE CASES

Use Case No.	UC-01		
Use Case Name	Sign Up Account		
Actor:	Customer		
Type:	Primary		
Description:	A customer registers a new account in the system to place orders and access services.		
Preconditions:	<ul style="list-style-type: none">Customer must have internet access.Customer has valid email/phone number.		
Postconditions:	<ul style="list-style-type: none">Account created and stored in system.		
Normal Flow			
Actor Action		System Response	
1. Customer selects “Sign Up”.		2. System opens registration form.	
3. Customer enters details (name, email, password).		4. System validates input format.	
5. Customer submits form.		6. System saves data and confirms account creation.	
Alternative Flow			
A3.1:	Invalid details → System highlights error → User re-enters.		
A5.1:	Email/phone already registered → System shows “Account Exists” → User prompted to login.		

Use Case No.	UC-02
Use Case Name	Browse Food Menu
Actor:	Customer
Type:	Primary
Description:	Customer views available food items in the menu.
Preconditions:	<ul style="list-style-type: none">Customer is logged in.
Postconditions:	<ul style="list-style-type: none">Food items displayed to customer.
Normal Flow	
Actor Action	System Response
1. Customer selects “Food Menu”.	2. System loads available items.
3. Customer browses items.	4. System shows item details (price, description, availability).
Alternative Flow	
A2.1:	If menu not available → System shows “Menu unavailable” message.

Use Case No.	UC-03		
Use Case Name	Confirm Order		
Actor:	Customer		
Type:	Primary		
Description:	Customer confirms and places an order after adding items to cart.		
Preconditions:	<ul style="list-style-type: none">• Customer has selected items in cart.• Customer has a valid payment method.		
Postconditions:	<ul style="list-style-type: none">• Order confirmed and saved.• Order ID generated.		
Normal Flow			
Actor Action		System Response	
1. Customer views cart.		2. System displays items with total price.	
3. Customer confirms order.		4. System saves order and generates Order ID.	
5. Customer selects payment.		6. System processes payment and confirms.	
Alternative Flow			
A5.1:	Invalid payment → System shows error → Customer retries.		
A3.1	Order confirmation fails → System shows “Try Again Later”.		

Use Case No.	UC-04		
Use Case Name	Rate Service		
Actor:	Customer		
Type:	Primary		
Description:	Customer provides feedback on food and delivery service.		
Preconditions:	<ul style="list-style-type: none">• Order completed and delivered.		
Postconditions:	<ul style="list-style-type: none">• Rating stored for analytics.		
Normal Flow			
Actor Action		System Response	
1. System prompts customer for rating after order.		2. Customer selects stars/enters comments.	
3. Customer submits rating.		4. System saves rating in database.	
Alternative Flow			
A3.1:	Rating submission fails → System prompts retry.		

Use Case No.	UC-05		
Use Case Name	Track Delivery		
Actor:	Delivery Rider		
Type:	Primary		
Description:	Rider tracks customer address and delivery route.		
Preconditions:	<ul style="list-style-type: none">Order assigned to rider.		
Postconditions:	<ul style="list-style-type: none">Rider navigates to customer location.		
Normal Flow			
Actor Action		System Response	
1. Rider selects “Track Delivery”.		2. System retrieves delivery address.	
3. Rider views map.		4. System shows real-time route navigation.	
Alternative Flow			
A3.1:	Map service unavailable → System shows “Location Error”.		

Use Case No.	UC-06		
Use Case Name	Update Delivery Status		
Actor:	Delivery Rider		
Type:	Primary		
Description:	updates the delivery status during the process.		
Preconditions:	<ul style="list-style-type: none">Rider logged into system.		
Postconditions:	<ul style="list-style-type: none">Delivery status updated.		
Normal Flow			
Actor Action		System Response	
1. Rider selects “Update Status”.		2. System displays status options (Picked, On the Way, Delivered).	
3. Rider updates status.		4. System saves status and notifies customer.	
Alternative Flow			
A3.1:	Update fails → System shows retry option.		

Use Case No.	UC-07		
Use Case Name	Manage Staff Attendance		
Actor:	Branch Manager		
Type:	Primary		
Description:	Manager manages daily staff attendance.		
Preconditions:	<ul style="list-style-type: none">• Staff present at branch.		
Postconditions:	<ul style="list-style-type: none">• Attendance log updated.		
Normal Flow			
Actor Action		System Response	
1. Manager opens attendance module.		2. System displays staff list.	
3. Manager marks attendance.		4. System saves attendance record.	
Alternative Flow			
A3.1:	If attendance not recorded → System prompts retry.		

Use Case No.	UC-08	
Use Case Name	Monitor Sales Report	
Actor:	Branch Manager	
Type:	Primary	
Description:	Manager views daily/weekly/monthly sales reports.	
Preconditions:	<ul style="list-style-type: none">Sales data available in system.	
Postconditions:	<ul style="list-style-type: none">Sales report displayed.	
Normal Flow		
Actor Action		System Response
1. Manager selects “Sales Report”.		2. System fetches report data.
3. Manager views sales summary.		4. System shows charts and totals.
Alternative Flow		
A2.1:	Report data unavailable → System shows “No Data Found”.	

Use Case No.	UC-09		
Use Case Name	View Incoming Order		
Actor:	Restaurant Staff		
Type:	Primary		
Description:	Staff views new orders placed by customers.		
Preconditions:	<ul style="list-style-type: none">• Orders exist in system.		
Postconditions:	<ul style="list-style-type: none">• Order details displayed.		
Normal Flow			
Actor Action		System Response	
1. Staff opens “Incoming Orders”.		2. System displays list of new orders.	
3. Staff selects order.		4. System shows order details.	
Alternative Flow			
A2.1:	No new orders → System shows “No Orders”.		

Use Case No.	UC-10	
Use Case Name	Update Order Status	
Actor:	Restaurant Staff	
Type:	Primary	
Description:	Staff updates the status of orders during preparation.	
Preconditions:	<ul style="list-style-type: none">Order is active.	
Postconditions:	<ul style="list-style-type: none">Status updated for customer and rider.	
Normal Flow		
Actor Action		System Response
1. Staff selects an active order.		2. System displays order details.
3. Staff updates order status (Preparing, Ready, Dispatched).		4. System updates database and notifies customer.
Alternative Flow		
A3.1:	Update fails → System prompts retry.	

Use Case No.	UC-11	
Use Case Name	Manage Payment Gateway	
Actor:	System Administrator	
Type:	Primary	
Description:	Admin configures and maintains payment gateway.	
Preconditions:	<ul style="list-style-type: none">Admin has valid credentials.	
Postconditions:	<ul style="list-style-type: none">Payment gateway settings updated.	
Normal Flow		
Actor Action		System Response
1. Admin logs into system.		2. System verifies credentials.
3. Admin selects “Payment Gateway Settings”.		4. System displays configuration options.
5. Admin updates settings.		6. System saves changes and confirms.
Alternative Flow		
A1.1:	Invalid login → Access denied.	
A5.1:	Update fails → System shows error.	

Use Case No.	UC-12		
Use Case Name	Resolve Refund Issue		
Actor:	Help Center Staff		
Type:	Primary		
Description:	Help staff handles refund requests from customers.		
Preconditions:	<ul style="list-style-type: none">Refund request exists.		
Postconditions:	<ul style="list-style-type: none">Refund processed or rejected.		
Normal Flow			
Actor Action		System Response	
1. Staff opens refund request.		2. System shows refund details.	
3. Staff verifies request.		4. System validates transaction.	
5. Staff approves/refuses refund.		6. System updates refund status and notifies customer.	
Alternative Flow			
A3.1:	Invalid request → System rejects automatically.		
A5.1:	Transaction error → System prompts retry.		