Product Requirements Document

Module 3: Menu & Item Selection
Project: QuickBites – Food Ordering & Delivery Platform
Prepared by: Varshini (Business Analyst)
Version 1.0

1. Feature Overview

Users should be able to view restaurant menus and add items to their cart with customizations (like add-ons, spice level, quantity, etc.).

2. Key Requirements

- Show menu categories (Starters, Main Course, Beverages)
- Show item details (name, price, veg/non-veg, rating)
- Allow item customization (add-ons, spice level)
- Allow add/remove items
- Update cart count instantly

3. User Stories

US1 – As a user, I want to browse a restaurant menu so I can choose what to order.
US2 – As a user, I want to customize my item with add-ons.
US3 – As a user, I want to increase or decrease item quantities.
US4 – As a user, I want to view item details before ordering.

4. Acceptance Criteria

- Menu loads correctly under each category
- Add-ons display only for relevant items
- Quantity updates the cart in real-time
- User sees “Item Added” confirmation

5. Functional Requirements

- FR1: Menu must be fetched from the restaurant database
- FR2: Items must show correct availability (Available/Out of Stock)
- FR3: Add-ons must update final price
- FR4: Cart counter must update instantly

6. Non-Functional Requirements

- Menu load time < 2.5 seconds
- Image optimization for fast loading
- Accurate pricing & tax calculations

7. Out of Scope

- AI-based food recommendations
- Allergy warnings

Module 4: Cart & Checkout

1. Feature Overview

Users should be able to review their selected food items, modify quantities, view total price, add delivery instructions, and proceed to payment.

2. Key Requirements

- Display all selected items
- Allow users to increase/decrease quantity
- Allow item removal
- Display taxes, delivery charges, packaging fees
- Allow applying promo codes
- Allow selection of delivery address

3. User Stories

US1 – As a user, I want to modify item quantities so I can adjust my order.
US2 – As a user, I want to add/remove items from the cart.
US3 – As a user, I want to see total cost breakdown before payment.
US4 – As a user, I want to add delivery instructions (e.g., "Leave at door").
US5 – As a user, I want to apply a promo code to reduce the order amount.

4. Acceptance Criteria

- Price recalculates correctly after changes
- Promo codes validate correctly
- User cannot proceed without selecting an address
- Taxes and delivery charges are displayed accurately

5. Functional Requirements

- FR1: Update item quantity dynamically
- FR2: Recalculate cart totals instantly
- FR3: Allow entry of promo codes & display discount
- FR4: Allow selection of saved or new address
- FR5: Validate minimum cart amount before checkout

6. Non-Functional Requirements

- Cart page load < 2 seconds
- Calculations accurate to 2 decimal places
- Smooth UI updates without lag

7. Out of Scope

- Wallet recharge
- Detailed tax breakdown per item

Project: QuickBites – Food Ordering & Delivery Platform

Module 5: Payment Flow

Version 1.0

1. Feature Overview

Users complete their order through a secure and fast payment process that supports multiple payment methods.

2. Key Requirements

- Provide secure payment options
- Show final payable amount
- Payment success/failure screen
- COD (Cash on Delivery) option
- Retry option in case of failed payment

3. User Stories

US1 – As a user, I want to choose my preferred payment method.
US2 – As a user, I want a smooth and secure payment experience.
US3 – As a user, I want to retry payment if it fails.
US4 – As a user, I want to pay via UPI for convenience.

4. Acceptance Criteria

- User sees UPI, Card, and COD options
- Payment gateway must return success/failure status
- Failed payment triggers a retry option
- Successful payment navigates to "Order Confirmed" screen

5. Functional Requirements

- FR1: Integrate Payment Gateway (UPI/Card/Netbanking)
- FR2: Support COD with backend flag
- FR3: Handle payment failure callbacks
- FR4: Display order summary before payment
- FR5: Generate transaction ID for each payment

6. Non-Functional Requirements

- Payment processing < 10 seconds
- 99% payment gateway uptime
- Secure encryption for transaction data

7. Out of Scope

- EMIs / Buy Now Pay Later
- Cryptocurrency payments

Project: QuickBites – Food Ordering & Delivery Platform

Module 6: Order Placement & Confirmation

Version 1.0

1. Feature Overview

After successful payment or COD selection, the user's order is officially created, confirmed, and sent to the restaurant for preparation.

2. Key Requirements

- Generate order ID
- Show order confirmation screen
- Send order details to restaurant dashboard
- Send confirmation notification to user
- Estimate delivery time

3. User Stories

US1 – As a user, I want to know my order is confirmed after payment.

US2 – As a user, I want to see the order ID for tracking.

US3 – As a user, I want to receive notifications for status updates.

4. Acceptance Criteria

- Order ID generated automatically
- Confirmation page shows: items, total amount, ETA
- Restaurant receives order instantly
- User receives push notification/ SMS

5. Functional Requirements

- FR1: Create order entry in database
- FR2: Notify restaurant dashboard
- FR3: Calculate estimated delivery time
- FR4: Send confirmation notification to user
- FR5: Store order details for tracking module

6. Non-Functional Requirements

- Order creation < 2 seconds
- 99.5% reliability
- Notifications must be delivered in real-time

7. Out of Scope

- Order cancellation by restaurant
- Scheduled delivery (covered separately)

Project: QuickBites – Food Ordering & Delivery Platform

Module 7: Order Tracking

Version 1.0

1. Feature Overview

Order Tracking allows users to monitor the live status of their order—from restaurant preparation to rider pickup and delivery.

2. Key Requirements

- Show order status updates
- Show rider location on a map (basic conceptual)
- Show estimated delivery time
- Allow contacting the delivery partner
- Allow contacting restaurant (optional)

3. User Stories

US1 – As a user, I want to track my order in real time.
US2 – As a user, I want to know the current order stage (Accepted → Preparing → Out for Delivery).
US3 – As a user, I want to see the rider's location on the map.
US4 – As a user, I want to contact the rider if needed.

4. Acceptance Criteria

- Status updates must display correctly
- Map shows rider movement
- ETA updates dynamically
- Contact buttons visible when rider is assigned

5. Functional Requirements

- FR1: Fetch live status from backend every 10 seconds
- FR2: Display 4 key statuses:
    1. Order Accepted
    2. Preparing
    3. Out for Delivery
    4. Delivered
- FR3: Map integration (Google Maps / API)
- FR4: Show rider details once assigned
- FR5: Notification alerts for status changes

6. Non-Functional Requirements

- Real-time updates with < 5-second delay
- Map loading < 2 seconds
- 99% location accuracy for rider

7. Out of Scope

- Advanced route optimization
- Multi-rider order splitting

Project: QuickBites – Food Ordering & Delivery Platform

Module 8: Scheduled Delivery

Version 1.0

1. Feature Overview

Scheduled Delivery allows users to choose a future date and time slot for their order instead of immediate delivery.

2. Key Requirements

- Show available delivery time slots
- Allow users to pick a future date
- Validate restaurant operating hours
- Show expected delivery window
- Store schedule details with the order

3. User Stories

US1 – As a user, I want to schedule my delivery for later.
US2 – As a user, I want to choose from available time slots.
US3 – As a user, I want clear confirmation of my selected delivery time.

4. Acceptance Criteria

- Only valid future dates/slots appear
- User cannot pick past time slots
- Confirmation page shows scheduled time
- Restaurant receives scheduled time data

5. Functional Requirements

- FR1: Show next 7 days of availability
- FR2: Display time slots (ex: 30-min intervals)
- FR3: Validate restaurant open/close hours
- FR4: Store scheduled date/time in order database
- FR5: Send reminder notification before delivery

6. Non-Functional Requirements

- Slot loading < 1.5 seconds
- Accurate time conversion (user timezone)
- No duplicate slot booking

7. Out of Scope

- Subscription-based recurring deliveries
- Multi-schedule orders

Project: QuickBites – Food Ordering & Delivery Platform

Module 9: Reviews & Ratings

Version 1.0

1. Feature Overview

This module allows users to rate restaurants and delivery partners and provide written feedback after an order is completed.

2. Key Requirements

- Rate restaurant (1–5 stars)
- Rate delivery partner
- Write a text review (optional)
- Post-order review prompt
- Display average ratings on restaurant listing

3. User Stories

US1 – As a user, I want to rate my order experience.
US2 – As a user, I want to leave feedback for other customers.
US3 – As a user, I want to view restaurant ratings before ordering.

4. Acceptance Criteria

- User can submit both star ratings and optional comments
- Restaurant average rating updates correctly
- Delivery partner rating is stored separately
- User is prompted to review only after order completion

5. Functional Requirements

- FR1: Allow rating submission (1–5 stars)
- FR2: Store ratings in the database
- FR3: Update average rating dynamically
- FR4: Display ratings on restaurant profile
- FR5: Trigger review prompt after a completed order

6. Non-Functional Requirements

- Ratings load < 1 second
- Reviews moderated to prevent abuse
- Scalable system for large numbers of reviews

7. Out of Scope

- AI review summarization
- Image/video-based reviews

Project: QuickBites – Food Ordering & Delivery Platform
Module 10: Offers & Coupons
Prepared by: Varshini (Business Analyst)
Version 1.0

1. Feature Overview

Users should be able to view, apply, and redeem promotional offers or coupons during checkout to get discounts on their orders. The system should validate coupons and reflect discounts in real-time.

2. Key Requirements

- Display available offers and coupons on the cart/checkout page
- Allow users to apply promo codes manually
- Validate coupon eligibility (e.g., minimum order value, first-time user, specific restaurants/items)
- Show discounted amount and final payable amount
- Notify users of invalid/expired coupons
- Allow multiple types of offers: percentage-based, flat discount, free delivery, or item-specific deals

3. User Stories

US1 – As a user, I want to see all applicable offers before checkout so I can save money.
US2 – As a user, I want to enter a promo code to get a discount.
US3 – As a user, I want to know if a coupon is invalid or expired.
US4 – As a user, I want the discount to update my total instantly.
US5 – As a user, I want to apply item-specific offers when ordering particular dishes.

4. Acceptance Criteria

- Available offers are displayed correctly based on cart contents and user eligibility
- Promo codes apply valid discounts accurately
- Invalid or expired coupons show an error message
- Total amount reflects discount immediately after applying the offer
- System prevents checkout if a coupon is applied incorrectly

5. Functional Requirements

- **FR1:** Fetch active offers and coupons from the backend
- **FR2:** Validate coupon code rules (min order value, user eligibility, item applicability)
- **FR3:** Apply discount to total order amount and display updated price
- **FR4:** Show error messages for expired, invalid, or inapplicable coupons
- **FR5:** Allow backend configuration for different types of promotions (percentage, flat, free delivery, item-specific)

6. Non-Functional Requirements

- Offer/coupon validation should happen in < 1 second
- Discounts must calculate accurately to 2 decimal places
- System should scale to handle high traffic during promotions
- UI should display applied coupons clearly and update totals dynamically

7. Out of Scope

- Automated AI-based offer suggestions (future enhancement)
- Third-party coupon integration outside QuickBites platform
- Loyalty points or wallet-based discounts