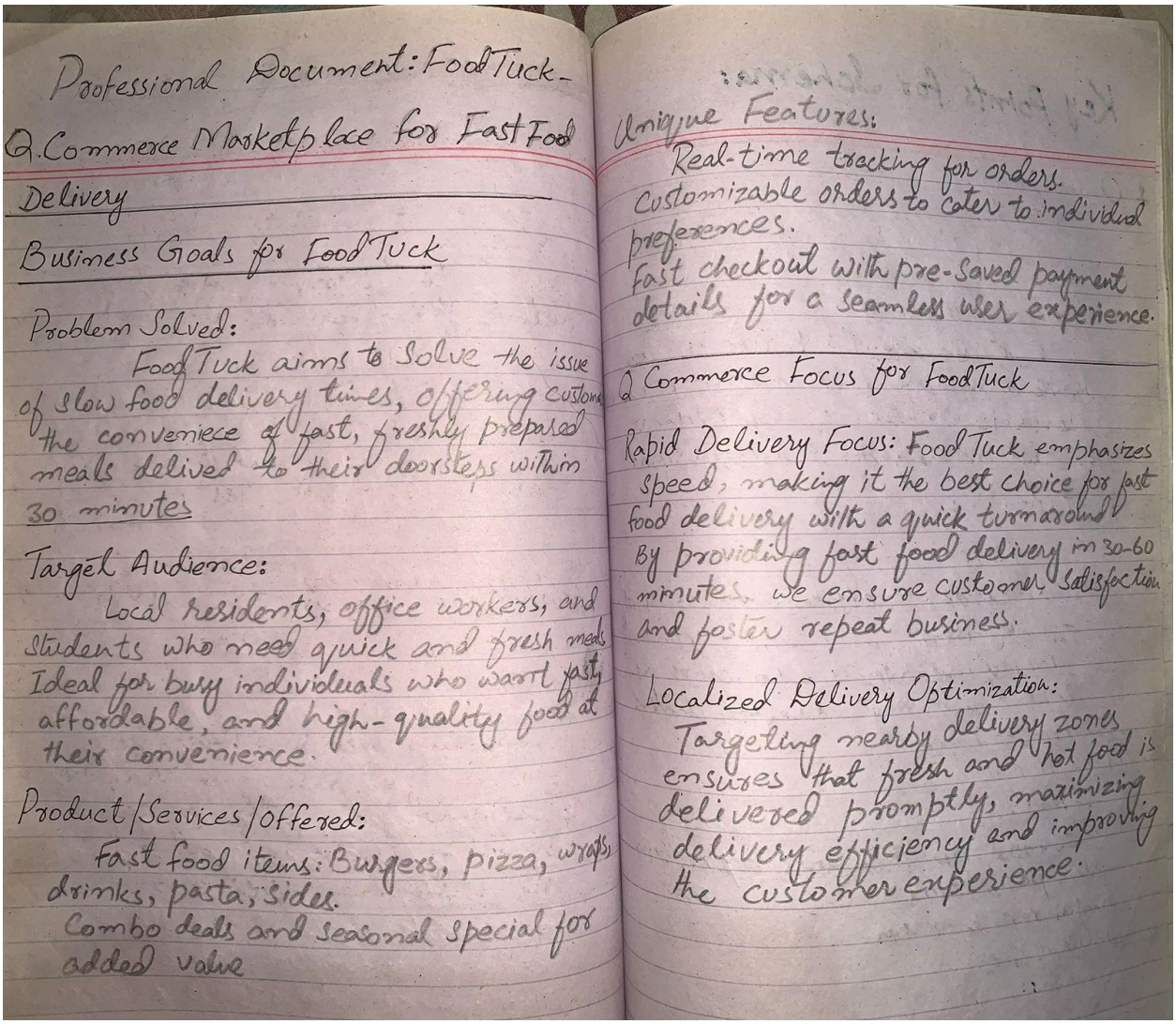
Day 01: Foundation of My Marketplace Journey



Professional Document: FoodTuck-
Q.Commerce Marketplace for Fast Food Delivery

Business Goals for FoodTuck

Problem Solved:
FoodTuck aims to solve the issue of slow food delivery times, offering customers the convenience of fast, freshly prepared meals delived to their doorsteps within 30 minutes

Target Audience:
Local residents, office workers, and students who need quick and fresh meals. Ideal for busy individuals who want fast, affordable, and high-quality food at their convenience.

Product/Services/offered:
Fast food items: Burgers, pizza, wraps, drinks, pasta, sides.
Combo deals and seasonal special for added value

Unique Features:
Real-time tracking for orders.
Customizable orders to cater to individual preferences.
Fast checkout with pre-saved payment details for a seamless user experience.

Q Commerce Focus for FoodTuck

Rapid Delivery Focus: FoodTuck emphasizes speed, making it the best choice for fast food delivery with a quick turnaround. By providing fast food delivery in 30-60 minutes, we ensure customer satisfaction and foster repeat business.

Localized Delivery Optimization:
Targeting nearby delivery zones ensures that fresh and hot food is delivered promptly, maximizing delivery efficiency and improving the customer experience.

Key Points for Schema:

1 Order Items:

Fields:

id, order id, product id, quantity, price

Relationships: Connects Product and Order. Each order can have multiple items, and each items is linked to a product (menu item) from a restaurant.

2. Restaurant:

Fields:

id, name, location, cuisine type, menu items

The restaurant will be directly linked to menu items (the product offered) and other fulfillment.

. Menu Items:

Fields: id, name, description, price category, prep time. This represents the food items available in the restaurant's menu

Delivery

Fields: delivery id, order id delivery status, estimated delivery time, Delivery details, like delivery status and time are recorded for each order

[user] ------> [order] ------> [Restaurant]

[user]	[order]	[Restaurant]
-ID	-OrderID	-ID
-Name	-UserID	-Name
-Email	-Product IDs	-Location
-Phone No	-Quantity	-Cuisine Type
-Address	-Total Price	-Menu Items

- Order Status

[Menu Item]

- ID
- Name
- Price
- Description

[Delivery]

- Delivery ID
- Order ID
- Delivery Status
- Estimated Delivery Time
- Delivery Address

Steps to Implement

1. Define Models in Next.js Backend:
   * Create models for products, orders, customers, and deliveries.
   * Use API to interact with the database for fetching and updating data.
2. Dynamic Routing:
   * Use src/app/orders/[orderId] for detailed order pages.
   * Use src/app/products/[productId] for individual product pages.
3. Frontend Features:
   * Implement reusable components in src/components to ensure code reusability.
   * Use TailwindCSS for responsive design and mobile-friendly layouts.
4. Real-Time Tracking:
   * Integrate WebSockets or similar technology for live updates on order status (e.g., "Preparing", "Out of Delivery", "Delivered").
5. Testing:
   Validate schema with sample data, test end-to-end order flows from product selection to order completion.