Menu-Q [A solution for efficient hotel operations]

**Project Idea:**

A combined web and mobile app system designed for hotels to manage tables, menus, and orders effectively to Simplify hotel operations, enhance customer experience, and ensure real-time coordination between staff and customers.

**Reason/Motivation behind the Idea:**

**Simplify Operations:** To reduce delays, avoid mistakes, and make managing tables, menus, and orders more efficient.

**Better Customer Experience:** Allow customers to browse the menu, place orders, and track updates easily from their devices.

**Improve Staff Coordination:** Ensure smoother communication between customers and staff for faster service.

**Reducing Dependency on Staff:** Let staff focus on quality service by minimizing the need for frequent order-taking.

**Problem Statement:**

In most hotel restaurants, the food ordering process is primarily manual. Customers call waiters to their tables to take orders, which involves noting down the items and quantities requested. This traditional method presents several challenges, including delays in taking orders, errors in manual note-taking, and inefficient communication between customers and hotel staff.

**Target Audience:**

Hotel and Restaurant Owners who want to modernize their operations, improve customer satisfaction, and enhance efficiency in their food ordering process.

**Use cases for project:**

**Primary Actor:** Customer, Hotel Staff

**Precondition:** The QR code must be placed on the hotel table, and the customer must have a smartphone with internet access.

**Post-Condition:** The order is completed, and the table record is cleared from the system after payment.

**Primary Flow (PF):**

|  |  |
| --- | --- |
| Actor Action | Systems Response |
| 1. Customer scans the QR Code placed on the table using their smartphone. | 1. System detects the table ID through the scanned QR code. |
|  | 1. System displays menu list with images, descriptions, prices and availability on customers device (website) |
| 1. Customer selects the food items and adds them to their order. | 1. System records the selected food items along with table details. |
| 1. Customer confirms and submits the order. | 1. System sends the order details to the hotel staff application. |
| 1. Hotel staff receive the order details on their device. |  |
| 1. Hotel staff reviews the order details, including table number and food items. |  |
| 9. Hotel staff update the order status of the items | 1. System displays the item status to the customer in real time |
| 1. After successfully delivering food to the customers table, hotel staff changes the order status | 1. System updates the order status on user website in real time |
|  | 1. System displays a button on the customer’s device if customer is done |
| 1. Customer decides to reorder additional items. | 1. The system allows the customer to browse the menu and place a new order. |
| 1. Customer places an additional order. | 1. The system sends the new order details to the hotel staff. |
| 1. Hotel staff process the new order. | 1. The system tracks the status of the additional order in real-time. |
| 1. Customer requests the final bill. | 1. The system automatically generates the bill based on all orders placed |
| 1. Hotel staff reviews the final bill and confirms payment. | 1. The system clears the table record once payment is confirmed. |

**Features Overview:**

**Admin Features:**

- Table management: Add tables, generate QR codes, track availability.

- Menu management: Add/delete items, toggle availability.

- Order management: Track orders, update item and order status, generate bills, free tables.

**Customer Features:**

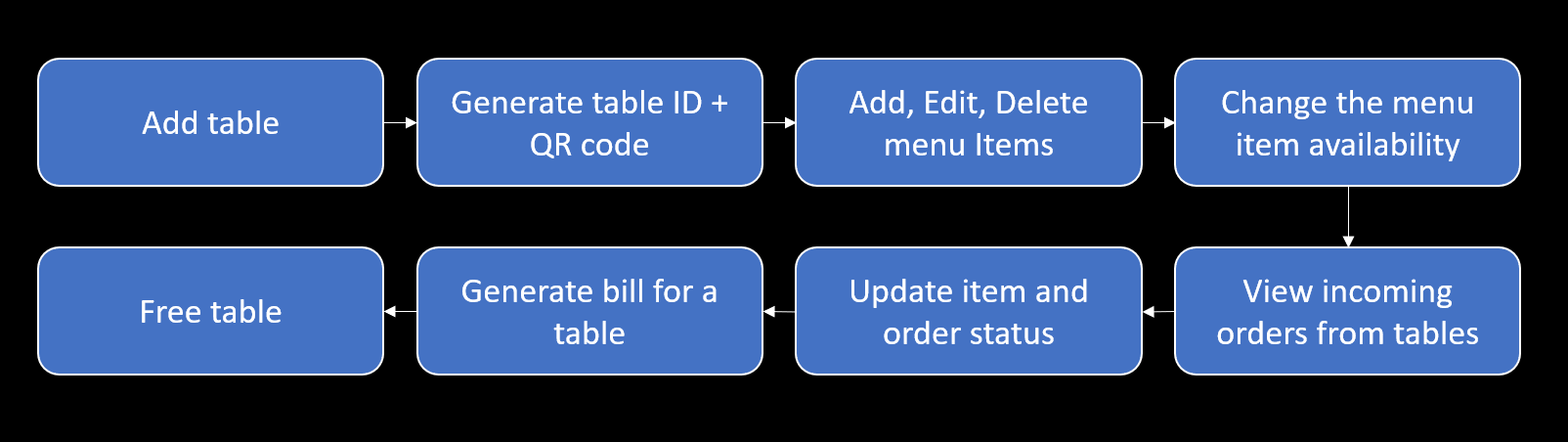
- QR code scanning for table assignment.

- Browse menu, sort by category or type, add or remove items from cart.

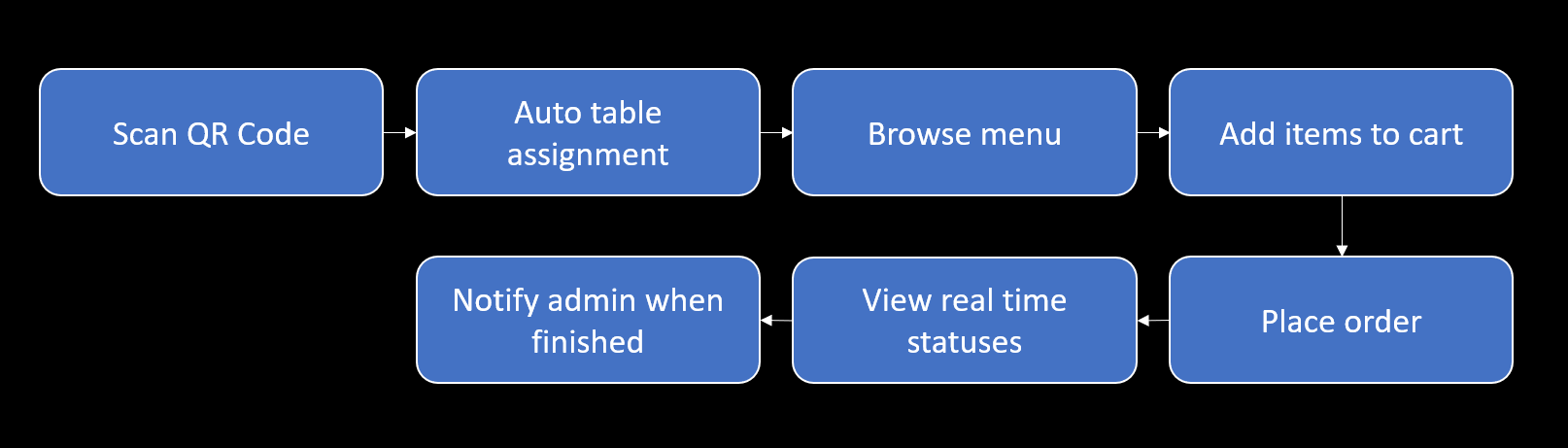
- Place orders, view statuses, and notify the admin after done.

**Application Flow:**

**Admin (Mobile App):**

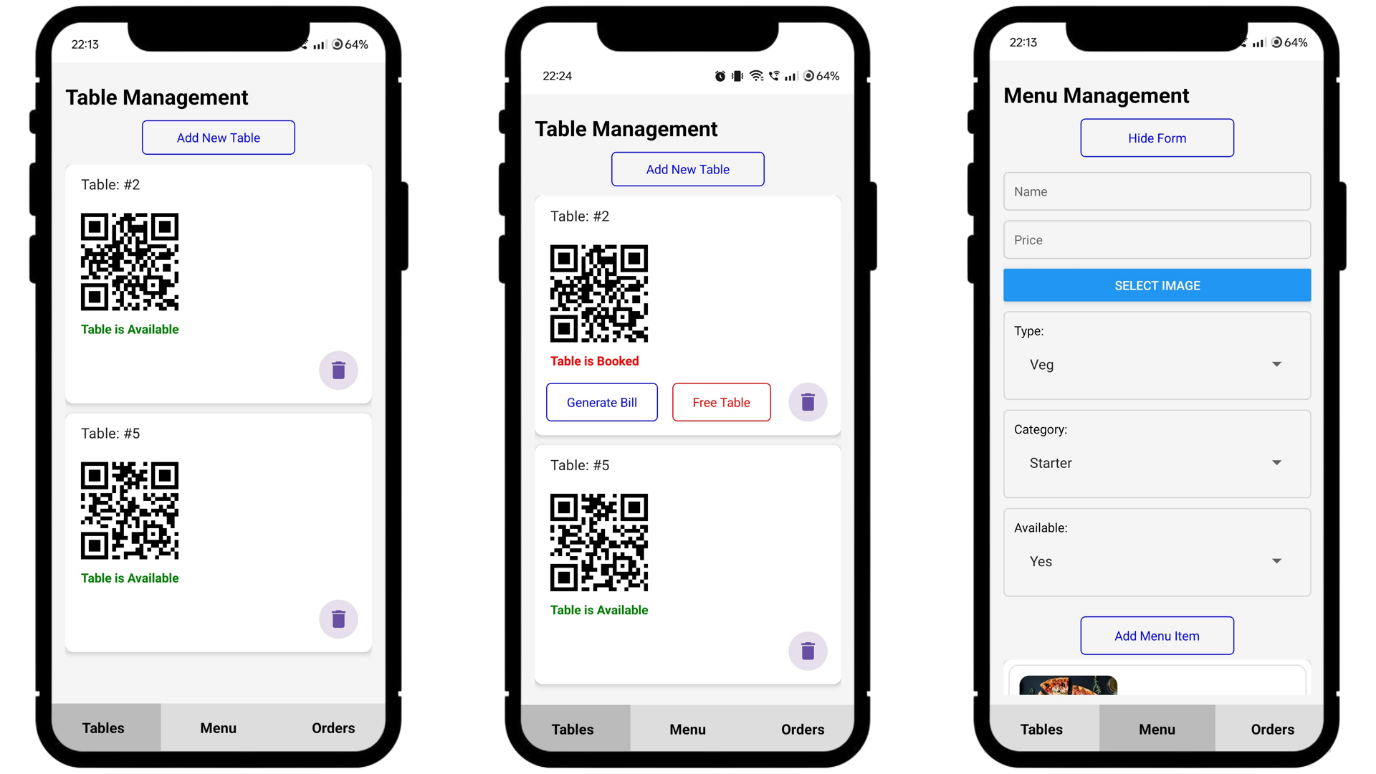


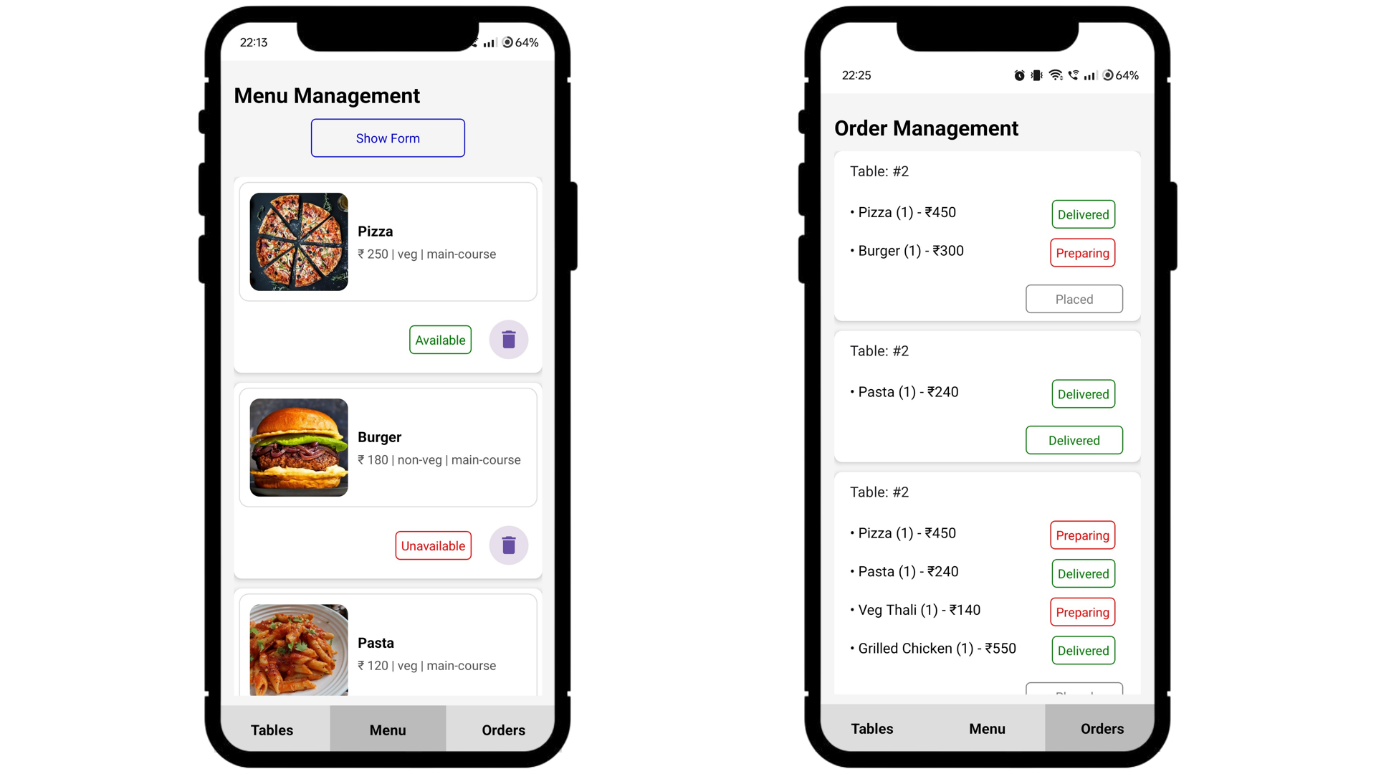
**Customer (Web App):**



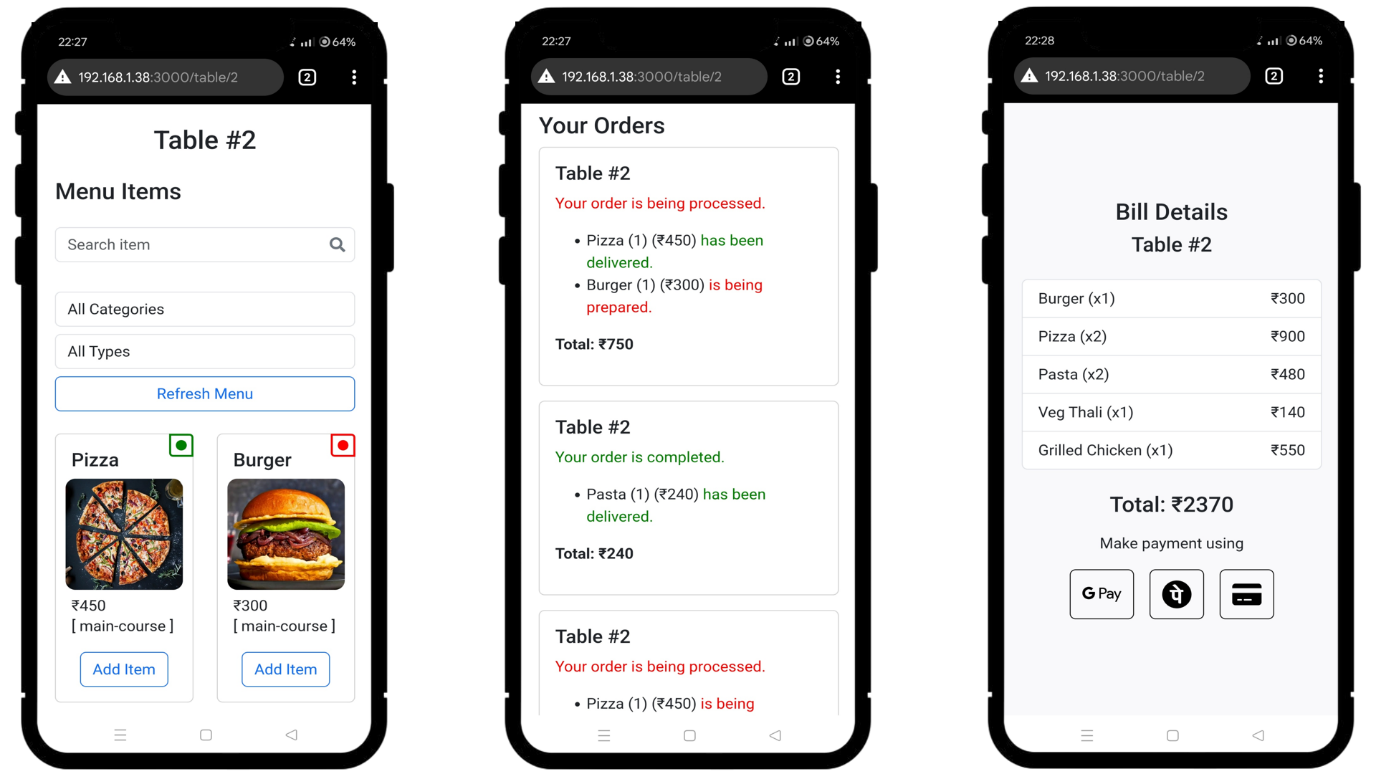
**UI/UX Design of Project:**

**Mobile App:**





**Web App:**



**Backend with APIs:**

**Backend:** Json server with global APIs

**Database:** Json server database

**APIs:**

http://localhost:3000/tables [Add, Delete tables]

http://localhost:3000/menuItems [Add, Edit, Delete menu items]

http://localhost:3000/orders [View orders and change order status]

http://localhost:3000/bills [Bill details for tables]

http://localhost:3000/donerequest [Tables ready for bill and reset]

**Technologies Used:**

**Frontend:**

Mobile App: React Native

Web App: React JS

**Backend:** Json server with global APIs

**Database:** Json server database

**Real-Time Updates:** WebSocket and Polling

**QR Code:** react-native-qrcode-svg

**Security Measures:**

Used a local server so, people outside of the app or webapp can’t access it

**Validation and testing:**

**Validations:**

Done all the form validations

Buttons can’t be pressed with empty input values

Every time checks if the table is available or not to avoid tables which are deleted

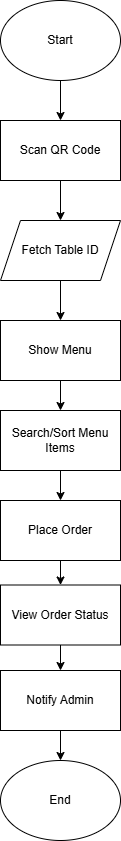
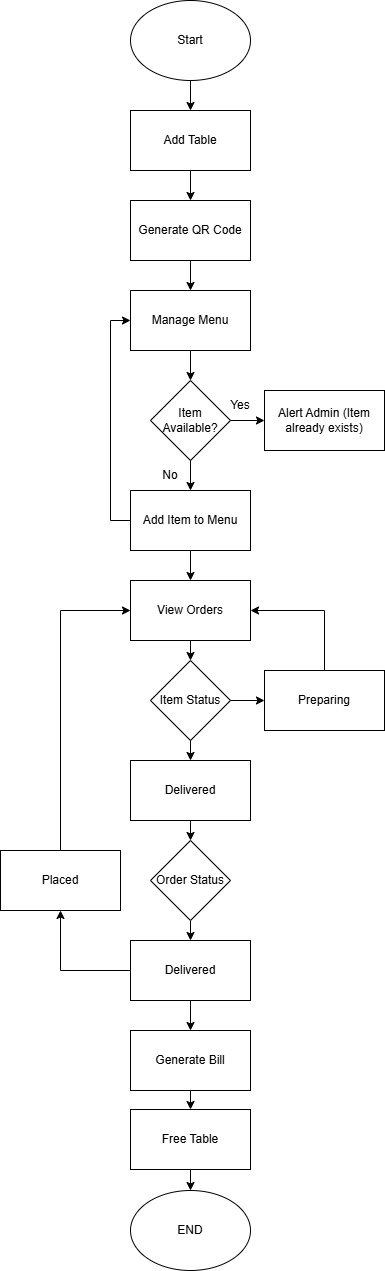
**Testing:**

Done unit testing of each component used in native app and react webapp

Integration testing of all the 3 main parent components

**Process flow diagram:**

Customer and Admin Flow Diagrams

**Challenges faced and solutions:**

**Real time data update from mobile application to web application**

Used polling for less important data and for critical data used socket.io

**QR code generation for table which will return the exact valid URL which is accepted by webapp**

Used react-native-qrcode-svg along with real time navigation and routes checking in webapp to check if the URL is valid or not

**Image uploading functionality**

Used 64-bit conversion from image to binary data passed it through the API and converted the binary data to image by 64-bit unencrypting. Implemented separate API for images to load the text data first and images comes later.

**Future scope:**

Integration of working UPI connectivity to the bill page

Review and rating system for customers

Dashboard for admin to get all the data for further analysis