Features List

1. Schedule Menu

* The owner can schedule the menu and card design

1. Online Ordering System through table

* Customers can order food from the website and pay with cash/UPI

1. Menu Editing By Owner (The owner can add or remove items and edit prices)

* Enable, add, disable, edit, and delete current items from the menu

1. Analytics (How much order through the day)

* Total Sales Today / This Week / This Month
* Number of Orders (Dine-in, Takeaway, Delivery)
* Menu edit
* Average Order Value (AOV)
* Customer Reviews
* Table Occupancy Rate (for dine-in)
* Inventory Status (Low Stock Alerts)
* Coupon dashboard
* Bill/invoice printing

 Top-Selling Dishes

 Least-Selling Dishes

 Sales by Category (Drinks, Appetizers, Mains, Desserts)

 Payment Methods Breakdown (Cash, Card, UPI, etc.)

1. Hindi, English & Bengali languages toggle

* Toggle between languages

1. Inventory management

* Add edit delete and update inventory items

1. Customer Feedback

* Add review after the payment process

1. AI Chabot

* Estimate order delivery time
* Ask AI what to eat
* AI-based games in the middle of the order (QUIZ to get discounts)

*API Details*

1. Login

Take email id and password then start session with JWT token

1. Register

Take all data in m\_restaurant table as request and return true/false in response

1. CRU for m\_menu, m\_order, *M-CustomerReview,*M-Inventory, T-Bill, T-Coupon, T-Alert
2. Add back details

Update/add m\_restaurant table under the mentioned company ID

1. Audit table for transaction, T-transactionAudit (later)
2. API to calculate total sales from m\_order table. Date range should be provided as request(today, week, and month)
3. API to calculate AOV
4. API to calculate table occupancy rate (later)
5. API to display top/least selling dish
6. Online payment gateway through API (paste QR from front end)

*API Requests*

Will update it in a parallel manner while coding

Login

[http://localhost:8080/restaurant/auth/login](http://localhost:8080/restaurant/login)

{

"email":"fergusdaw2000@gmail.com",

"pswd":"123456"

}

{ "Status": "Success", "Message": "Login Success" }

Register

http://localhost:8080/restaurant/auth/register

{

"userName": "Adarsh",

"pswd": "69",

"companyId": "3",

"email": "adarsh@example.com",

"totalTables": "10",

"ifscCode": "ABC123456",

"gstNumber": "GSTIN12345XYZ",

"address": "123, Sample Street, City, Country",

"phoneNumber": "1234567890"

}

readMenuData

<http://localhost:8080/restaurant/readMenuData>

{

"companyId" : "1"

}

CreateMenu

<http://localhost:8080/restaurant/createMenu>

{

"itemName": "Dal",

"itemCode": "3-0002",

"itemStatus": "Y",

"itemPrice": "99",

"category": "Sides",

"companyId": "3",

"effectiveFromDate": "16-02-2025"

}

UpdateMenu

<http://localhost:8080/restaurant/updateMenu>

{

"id":"5",

"itemName": "Dal",

"itemCode": "3-0003",

"itemStatus": "Y",

"itemPrice": "99",

"category": "Sides",

"companyId": "3",

"delFlag": "N",

"effectiveFromDate": "16-02-2025"

}

readOrder

<http://localhost:8080/restaurant/readOrder>

{

"companyID" : "2"

}

createOrder

<http://localhost:8080/restaurant/createOrder>

{

"id": 1,

"tableNumber": "12",

"companyID": "3",

"itemsName": "Burger Combo",

"itemsCode": "BGC123",

"itemsPrice": "250",

"paymentStatus": "Paid",

"paymentMode": "Credit Card",

"processStatus": "Completed"

}

updateOrder

{

"id":"5",

"delFlag": "Y"

}

Update data related to user

<http://localhost:8080/restaurant/updateUserProfile>

{

"id":"3",

"bankAccountNumber": "123456789012",

"ifscCode": "HDFC0001234"

}

Update customer review

<http://localhost:8080/restaurant/updateCustomerReview>

{

"id": "2",

"delFlag": "Y"

}

create

<http://localhost:8080/restaurant/createCustomerReview>

{

"tableNumber": "21",

"companyID": "3",

"delFlag": "N",

"message": "Excellent ambiance and quick service!",

"stars": "5"

}

Read customer review

<http://localhost:8080/restaurant/readCustomerReview>

{

"companyID":"1"

}

Read inventory data

<http://localhost:8080/restaurant/readInventory>

{

"companyID":"1"

}

Create Inventory

<http://localhost:8080/restaurant/createInventory>

{

"itemName": "Cheese Burger",

"price": "150",

"quantity": "50",

"itemCode": "CBG123",

"companyID": "2"

}

Update inventory

<http://localhost:8080/restaurant/updateInventory>

{

"id": "2",

"delFlag": "Y"

}

Read bill

<http://localhost:8080/restaurant/readBill>

{

"companyID":"1"

}

Create a bill

<http://localhost:8080/restaurant/createBill>

{

"userName": "John Doe",

"companyID": "2",

"address": "123 Main Street, Cityville",

"phone": "+91 9876543210",

"email": "johndoe@example.com",

"gstNumber": "GST123456XYZ",

"paymentMode": "Credit Card",

"items": "Pizza, Pasta, Coke",

"price": "850",

"billNo": "BILL20250216",

"persons": "3"

}

Read coupon data

<http://localhost:8080/restaurant/readCoupon>

{

"companyID":"1"

}

Create coupon

<http://localhost:8080/restaurant/createCoupon>

{

"id": 101,

"description": "10% off on all orders",

"amount": "10",

"companyID": "2",

"expDate": "2025-12-31",

"delFlag": "N"

}

Update coupon

<http://localhost:8080/restaurant/updateCoupon>

{

"id": "101",

"delFlag": "Y"

}

Read alert data

<http://localhost:8080/restaurant/readAlert>

{

"companyID" : "1"

}

Create alert data

<http://localhost:8080/restaurant/createAlert>

{

"alertName": "System Update",

"description": "Scheduled maintenance at midnight",

"status": "Active",

"priority": "High",

"companyID": "2"

}

Update alert data

<http://localhost:8080/restaurant/updateAlert>

{

"id": "4",

"delFlag": "Y"

}

Total Sales

<http://localhost:8080/restaurant/totalSales>

{

"companyId": "1",

"fromDateAndTime": "2025-02-01 16:40",

"toDateAndTime": "2025-02-15 14:39"

}

Aov

<http://localhost:8080/restaurant/aov>

{

"companyId": "1",

"fromDateAndTime": "2025-02-01 16:40",

"toDateAndTime": "2025-02-15 14:39"

}

Best and least selling dish

<http://localhost:8080/restaurant/bestLeastSellingDish>

{

"companyId": "1",

"fromDateAndTime": "2025-02-01 16:38",

"toDateAndTime": "2025-02-17 14:39"

}