Digital Egypt Pioneers Initiative

React Front-End Development

ReservQ Restaurant website

by React

Supervised by:

Eng. Basma Abdel Halim

Team members:

Ahmed Eid

Fathey Khaled

Lojien Fathey Elfayoumy

Logina yasser noureldeen

Mariam elsaid abozaid

Abstract

The **ReservQ** Restaurant Website is a comprehensive digital platform designed to provide a seamless and convenient dining experience for customers, whether they prefer dining in, takeout, or delivery. The platform features an intuitive user interface that allows customers to browse menus, book tables, and place orders efficiently.

Additionally, the website includes an advanced administrative system that enables restaurant owners and managers to oversee reservations, track orders, update menus, and interact with customers effortlessly. A dedicated dashboard provides reports and analytics to help optimize performance and enhance user experience.

The platform also supports a secure login system, ensuring users have access to personalized services such as order history, reservation management, and promotional offers. On the administrative side, managers can control user roles and staff permissions through a flexible and efficient management system.

This documentation outlines all the technical and functional aspects of the website, including the main interface, user roles, reservation and payment mechanisms, order management, and security protocols that safeguard user data and privacy. **ReservQ** is designed to be a complete restaurant management solution, enhancing both customer satisfaction and operational efficiency.

Table of Contents

Introduction

Welcome to the ReservQ Restaurant Website documentation. This platform is designed to revolutionize the dining experience by offering a seamless and efficient way for customers to enjoy their meals, whether they prefer dining in, takeout, or delivery. With a user-friendly interface, advanced management tools, and a wide range of functionalities, the website ensures convenience and accessibility for all users, including customers, restaurant staff, and administrators.

The platform is built to enhance efficiency at every level, quick online ordering to real-time order tracking and secure payment processing. Customers can browse restaurant menus, customize their orders, and manage their bookings effortlessly. Meanwhile, restaurant owners and administrators can oversee operations through an intuitive dashboard that provides real-time insights, order management tools, and menu customization options.

Security and personalization are key aspects of the ReservQ experience. The website features a secure login system that allows users to access their accounts, track their orders, and receive personalized offers. Additionally, administrators can manage user roles, monitor system activity, and ensure smooth operations through comprehensive control settings.

This documentation provides a detailed overview of the platform's core components, key features, and usage guidelines for both customers and administrators. It covers essential functionalities such as user authentication, order placement, reservation management, and system security, ensuring that all stakeholders can navigate the platform efficiently and make the most of its capabilities.

1. Project Planning & Management :

1.1.Project Proposal

- **Overview of the project**

The **ReservQ Restaurant Website** is a modern and efficient digital solution designed to enhance the dining experience for customers while streamlining restaurant operations. This platform provides an all-in-one system for table reservations, online ordering, and delivery management, ensuring convenience for both customers and restaurant staff.

The website offers a seamless and intuitive user interface, enabling customers to browse menus, place orders, and manage reservations with ease. Whether dining in, ordering takeout, or opting for delivery, users can enjoy a hassle-free experience tailored to their preferences. Secure payment processing and order tracking further enhance customer satisfaction.

For restaurant administrators and staff, **ReservQ** includes a powerful backend system that simplifies menu management, order tracking, and customer interactions. A dynamic dashboard provides real-time analytics, helping restaurant owners optimize operations, improve service efficiency, and boost customer engagement.

Security and personalization are integral to the platform, featuring secure login authentication and personalized user accounts that allow customers to access their order history, manage bookings, and receive customized promotions. Administrators can efficiently manage user roles, monitor system activity, and ensure smooth workflow through comprehensive control settings.

This documentation serves as a guide to understanding the **ReservQ Restaurant Website**, covering its core functionalities, user roles, and system architecture. It provides detailed insights into navigation, user authentication, order processing, and administrative tools, ensuring that both customers and restaurant operators can fully leverage the platform’s capabilities for an optimized dining experience.

• objectives

The primary objective of the ReservQ Restaurant Website is to provide a seamless and efficient digital solution for both customers and restaurant management. The platform is designed to enhance the overall dining experience by offering a range of essential features, including an online menu, reservations, order management, and customer engagement tools.

Key Goals:

1. Online Menu Accessibility – Allow customers to browse updated restaurant menus with detailed descriptions, pricing, and customization options.
2. Seamless Online Ordering – Facilitate easy ordering for dine-in, takeout, and delivery, with secure payment options and real-time order tracking.
3. Enhanced Customer Engagement – Offer personalized promotions, discounts, and loyalty rewards to encourage repeat business and customer satisfaction.
4. Efficient Restaurant Management – Provide administrators with a user-friendly dashboard to manage orders, reservations, customer interactions, and inventory.
5. Secure User Authentication – Ensure a safe and personalized experience through a robust login system, allowing customers to track their orders and manage bookings.
6. Real-Time Notifications – Keep customers informed about order status, reservation confirmations, and special offers through instant updates.

By integrating these features, ReservQ aims to modernize restaurant management, improve customer satisfaction, and drive business growth through an intuitive and functional digital platform.

- **scope of the Project.**

1. Customer Features

- Online Menu – Customers can browse an up-to-date menu with detailed descriptions, images, and pricing.
- Online Ordering – Enables customers to place orders for dine-in, takeout, or delivery.
- **User Authentication** – Customers can create accounts, manage orders, track reservations, and receive personalized promotions.

2. Restaurant Management Features

- Admin Dashboard – A centralized interface for managing orders, reservations, and customer interactions.
- Menu Management – Enables restaurant administrators to update menus, pricing, and availability dynamically.
- Order & Reservation Management – Allows staff to track, process, and manage incoming orders and bookings efficiently.
- Promotions & Discounts – Admins can create and manage special offers and loyalty programs.

3. Technical Aspects

- Responsive Design – The website will be optimized for desktops, tablets, and mobile devices.
- Security – Implements user authentication, secure transactions, and data encryption.
- Notifications & Alerts – Sends email or SMS updates for order confirmations, promotions, and important alerts.

The **ReservQ Restaurant Website** aims to bridge the gap between customers and restaurant management, ensuring a smooth and modernized dining experience while improving operational efficiency.

1.2.Project Plan

- **Project Timeline & Milestones**

Phase	Tasks	Duration	Milestones
1. Planning & Requirement Analysis	- Define project scope - Identify key features & functionalities - Gather requirements from stakeholders	**1Week**	✅ Project Scope Finalized
2. UI/UX Design	- Wireframing & Prototyping - Designing User Interface (UI) - Feedback & Iterations	**1 Week**	✅ UI/UX Design Completed
3. Development – Frontend & Backend	- Develop website frontend (React, HTML, CSS, JavaScript) - Backend development - API integration for payments & notifications	**3 Weeks**	✅ Core Functionalities Developed
4. Testing & Quality Assurance	- Unit Testing - User Acceptance Testing (UAT) - Performance & Security Testing	**2 Week**	✅ System Testing Completed
5. Deployment & Launch	- Server setup & cloud deployment - Beta Testing with real users - Final bug fixes & optimization	**1Week**	✅ Website Launched
6. Maintenance & Future Enhancements	- Continuous monitoring - Feature updates based on feedback - Performance improvements	**Ongoing**	✅ Post-Launch Support

- **Deliverables**

Phase 1 – Planning & Analysis

- Project Requirement Document
- Project Plan & Timeline

Phase 2 – UI/UX Design

- Wireframes & Prototypes
- UI Design Mockups

Phase 3 – Development

- Website Frontend & Backend Code

- API Integrations (Payments, Authentication, Notifications)

Phase 4 – Testing & QA

- Bug Reports & Fixes
- Test Cases & Results

Phase 5 – Deployment & Launch

- Live Website Deployment
- Final System Documentation

Phase 6 – Maintenance & Enhancements

- User Feedback Reports , Feature updates

1.3Task Assignment & Roles

Team Member	Role	Responsibilities
Member 1	**Team leader** & **Frontend Developer**	- Oversee project timeline and deliverables - Manage team coordination and task assignments - Ensure documentation is up to date
Member 2	**Frontend Developer**	- Design wireframes, prototypes, and final UI - Implement frontend components (React, HTML, CSS) - Ensure responsive and user-friendly design - Assist in frontend testing
Member 3	**Frontend Developer**	- Develop interactive features for the website - Implement dynamic UI elements and navigation - Ensure cross-browser compatibility
Member 4	**Frontend Developer &Backend Developer**	- Develop server-side logic - Ensure smooth data flow between frontend and backend
Member 5	**Frontend Developer**	- Perform testing (user testing) - Implement frontend components (React, HTML, CSS) - Ensure responsive and user-friendly design - Monitor site performance post-launch

1.4. Risk Assessment & Mitigation Plan

The ReservQ Restaurant Website project faces several risks, categorized into project, technical, operational, and post-launch risks.

1. Project Risks include scope creep, time mismanagement, resource limitations, and communication gaps. To mitigate these, we will define a clear scope, set realistic deadlines, and ensure regular team meetings
2. Technical Risks involve bugs, performance issues. We will conduct rigorous testing, optimize performance, implement strong security protocols
3. Operational & User Risks such as low user engagement, high traffic will be managed by designing an intuitive UI, using scalable hosting
4. Post-Launch Risks include customer support overload, frequent maintenance needs, and market competition. To address these, we will continuously improve the platform based on user feedback.

By actively monitoring and addressing these risks, the project will remain stable, secure, and user-friendly.

1.5.Key Performance Indicators (KPIs) – Metrics for Project Success

1. System Performance & Reliability
   - Response Time: Page load time for a smooth user experience.
   - System Uptime: Ensure 99.9% uptime to minimize service disruptions.
2. User Engagement & Adoption
   - User Retention Rate: Monitor how many users continue using the platform over time.
3. Operational Efficiency
   - Order Processing Time: Time taken from order placement to confirmation should be fast and seamless.
4. Security & Compliance
   - Number of Security Incidents: Track data breaches or unauthorized access attempts.
5. Customer Satisfaction : Customer Reviews & Ratings

By monitoring these KPIs, we can continuously improve the platform rerformance

2. Literature Review:

2.1. Feedback & Evaluation

The lecturer's assessment will focus on several aspects, including the functionality, usability, technical implementation, and adherence to project requirements. Key areas of evaluation include:

- Project Scope & Execution: Whether the website meets its intended objectives (reservations, ordering, customer engagement).
- Code Quality & Efficiency: Well-structured, optimized, and secure coding practices.
- User Interface (UI) & User Experience (UX): Simplicity, responsiveness, and ease of use.
- System Reliability & Performance: The website's speed, uptime, and ability to handle multiple users.
- Innovation & Creativity: Unique features that enhance user experience.

2.2. Suggested Improvements

Based on feedback, possible areas of enhancement include:

- Enhancing UI/UX Design: Improve visual aesthetics, animations, and user interactions.
- Advanced Features: Implement AI-based recommendations, loyalty programs, or chatbot support.
- Security Upgrades: Strengthen authentication, encryption, and protection against cyber threats.
- Performance Optimization: Improve load times, database queries, and overall system efficiency.
- Cross-Platform Compatibility: Ensure smooth operation across different browsers and devices.

2.3.Final Grading Criteria

The grading system will be structured as follows:

1. Documentation (20%) – Clear project description, technical documentation, and proper citations.
2. Implementation (40%) – Code quality, functionality, and completeness of the website.
3. Testing & Debugging (20%) – Performance, security, and usability testing with bug fixes.
4. Presentation & Demonstration (20%) – Clarity, professionalism, and ability to explain project components effectively.

3. Requirements Gathering:

The ReservQ Restaurant Website project requires careful analysis and planning to ensure that it meets the needs of all stakeholders. This section outlines the key stakeholders, user stories, functional and non-functional requirements necessary for the system's success.

3.1. Stakeholder Analysis

Key stakeholders and their needs include:

- Customers – Need an easy-to-use platform for browsing menus, making reservations, and placing orders
- Restaurant Owners & Managers – Require a dashboard to manage menus, reservations, orders, and customer feedback efficiently.
- Staff (Chefs, Waiters, Delivery Personnel) – Need access to real-time order updates and reservation details.
- Administrators – Must have full control over the system, including user management, reports, and security settings.

3.2. User Stories & Use Cases

- Customer Use Case: A user visits the website, browses the menu, , or places an order for delivery.
- Restaurant Manager Use Case: A manager logs in to update the menu , and track daily sales.
- Staff Use Case: A chef or waiter receives real-time order notifications and updates order status.
- Administrator Use Case: An admin manages user access, reviews system performance, and ensures security compliance.

3.3. Functional Requirements

The website must support the following key functionalities:

1. User Authentication – Secure login, registration, and profile management.
2. Menu Browsing & Search – Display restaurant menus with categories and filters.
3. Reservations & Order Management – Allow customers to book tables or place online orders.
4. Payment Integration – Support multiple payment methods (credit card, PayPal, cash-on-delivery).
5. Real-time Notifications – Inform users about order status, reservation confirmations, and promotions.
6. Admin Dashboard – Provide a control panel for restaurant owners and staff.
7. Reviews & Ratings – Enable customers to leave feedback on their dining experience.
8. Multi-Device Compatibility – Ensure the website works on desktops, tablets, and mobile devices.

3.4. Non-Functional Requirements

- Performance: Website should load within 2 seconds and handle high traffic efficiently.
- Security: data protection to prevent cyber threats.
- Usability: Ensure an intuitive user experience with responsive design for all devices.
- Reliability: Maintain 99.9% uptime, with backup systems to prevent downtime.

By gathering and addressing these requirements, the ReservQ Restaurant Website will ensure a seamless and user-friendly experience for all stakeholders.

4. System Analysis & Design :

4.1. Problem Statement & Objectives :

- **Problem Statement**

Traditional restaurant reservation and ordering systems often suffer from inefficiencies such as long wait times, miscommunication in orders, and lack of real-time tracking. Customers experience frustration when booking tables, placing orders, or managing their dining preferences, while restaurant staff struggle with operational bottlenecks and outdated management tools.

- **Project Objectives**

The ReservQ Restaurant Website aims to:

Provide a seamless online reservation system to reduce wait times.

Enable quick and easy online ordering for takeout and delivery.

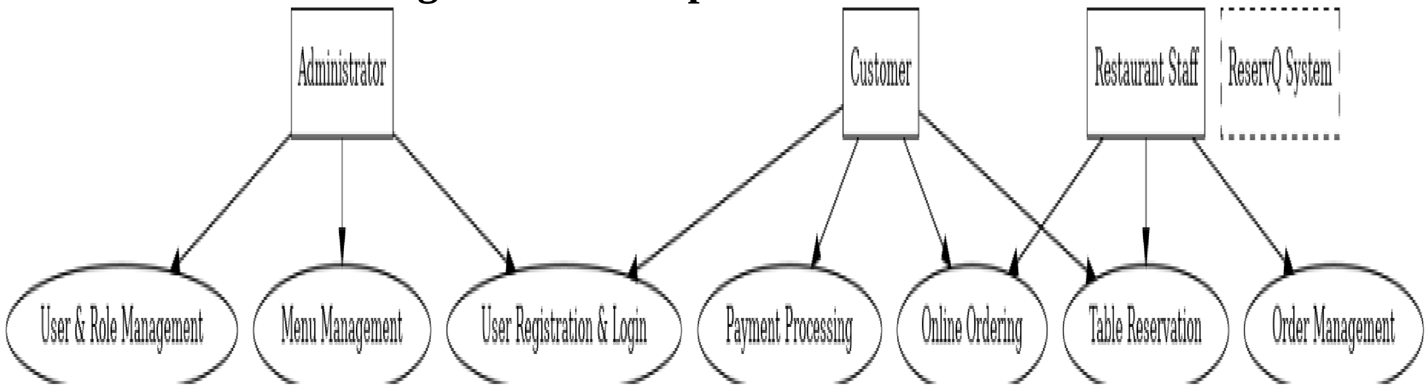
Offer real-time order tracking for customers.

Implement a user-friendly interface for smooth navigation.

Empower restaurant administrators with an intuitive dashboard for menu management, order handling, and analytics.

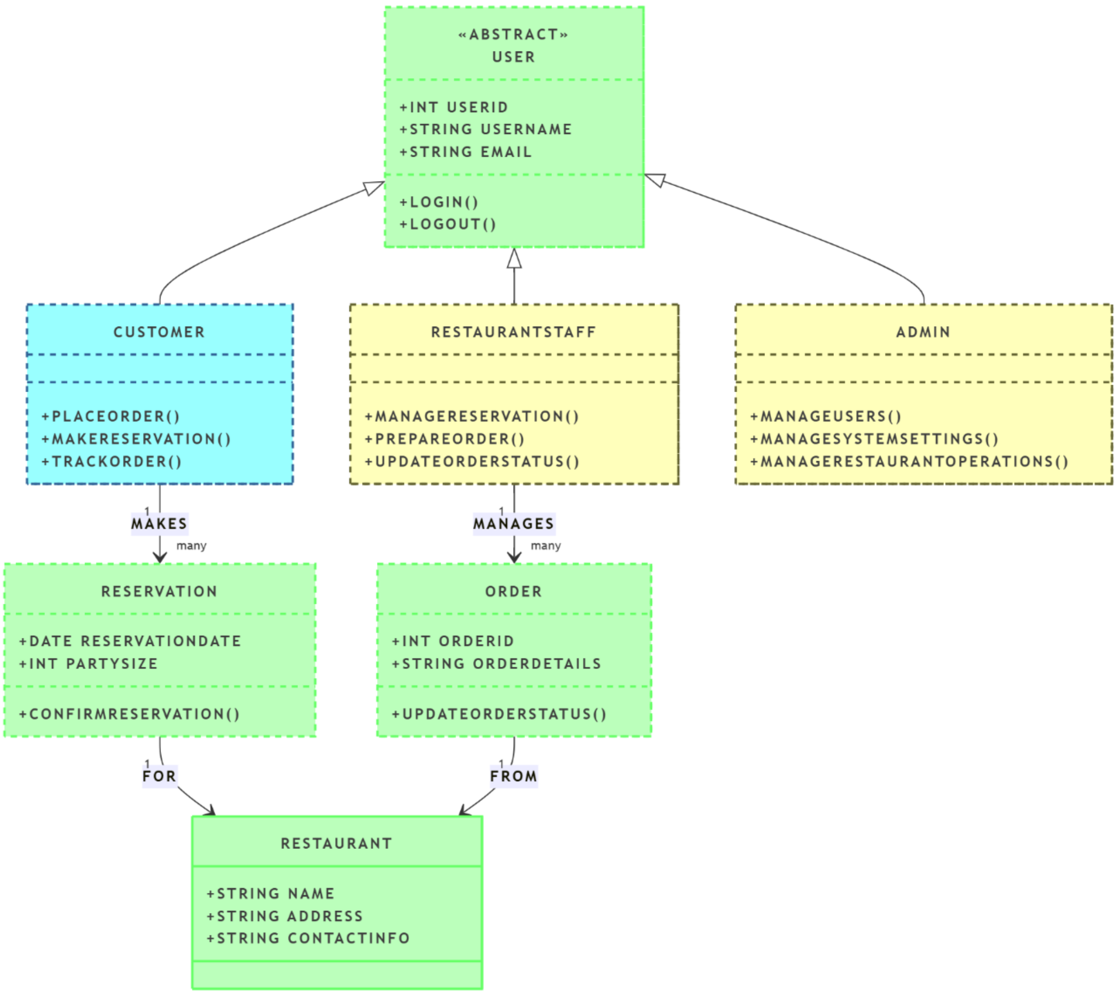
Ensure secure payments and user authentication for data protection.

- **Use Case Diagram & Descriptions**



The system consists of three primary actors:

- Customer – Places orders, makes reservations, and tracks orders.
- Restaurant Staff – Manages reservations, prepares orders, and updates order statuses.
- Administrator – Manages users, system settings, and restaurant operations.



- **Use Case Descriptions**

Use Case ID	**Use Case Name**	**Actors**	**Description**	**Steps**
UC1	User Registration & Login	Customer, Administrator	Users can register and log in securely using email, phone, or social authentication.	1. User enters credentials. 2. System verifies details. 3. User gains access to the platform.
UC2	Table Reservation	Customer, Restaurant Staff	Customers can book a table, and staff can confirm or modify reservations.	1. Customer selects a date, time, and number of guests. 2. System checks availability. 3. Reservation is confirmed or modified by staff if needed.
UC3	Online Ordering	Customer, Restaurant Staff	Customers can place food orders for takeout or delivery, and staff processes them.	1. Customer selects menu items. 2. Order is submitted to the kitchen. 3. Staff prepares and updates the status.
UC4	Order Management	Restaurant Staff	Staff can update, process, and track food orders.	1. Staff views pending orders. 2. Orders are prepared and updated. 3. Customers receive notifications.
UC5	Payment Processing	Customer	Customers can securely pay for their orders using online payment gateways.	1. Customer selects a payment method. 2. System processes the payment securely. 3. Confirmation receipt is sent.
UC6	Menu Management	Administrator	The admin can add, update, or remove menu items.	1. Admin logs in. 2. Updates menu details. 3. Changes reflect on the website instantly.
UC7	User & Role Management	Administrator	The admin can manage users, assign roles, and monitor activity.	1. Admin views user list. 2. Updates user permissions if needed. 3. System updates roles accordingly.
UC8	Real-Time Order Tracking	Customer	Customers can track the status of their order in real-time.	1. Customer places an order. 2. System updates order status dynamically. 3. Customer receives notifications.

- ## Functional & Non-Functional Requirements

Functional Requirements :

User Authentication: Secure login for customers, staff, and administrators.

Reservations: Customers can book tables with date and time selection.

Online Ordering: Customers can browse menus, customize meals, and place orders.

Order Tracking: Real-time status updates on orders for customers.

Payment Integration: Secure payment gateway for online transactions.

Admin Dashboard: Manage users, reservations, menu items, and system settings.

Non-Functional Requirements :

Scalability: The system must support high traffic during peak dining hours.

Security: Implement SSL encryption, secure authentication, and data protection.

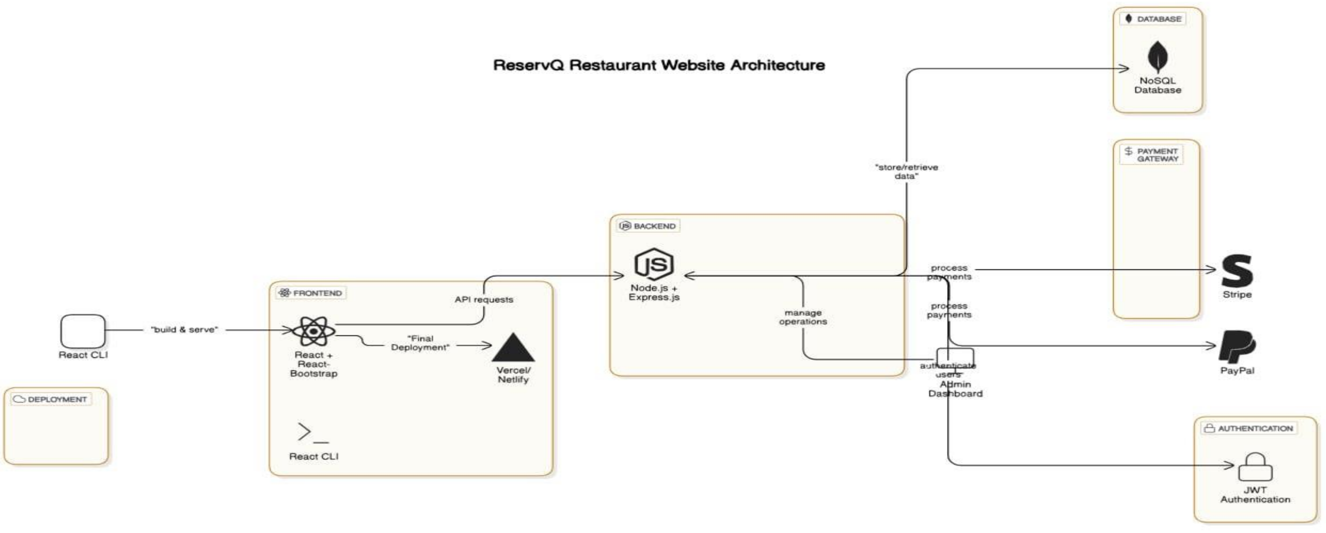
Performance: Pages should load within 2 seconds for optimal user experience.

Usability: Intuitive UI/UX for smooth navigation across all devices.

Reliability: Ensure 99.9% uptime for critical restaurant operations.

- ## Software Architecture:

  The ReservQ Restaurant Website follows a Model-View-Controller (MVC) architecture to ensure modularity, scalability, and maintainability. The system is designed to support real-time ordering, reservations, user authentication, and secure payments, ensuring smooth interactions between customers, restaurant staff, and administrators.



- **High-Level System Components & Interactions**

  - **Frontend (Client-Side) – View**
    Technology Used: React.js

User Interface (UI): Provides a responsive and interactive experience.

Order & Reservation Forms: Enables users to book tables and place orders.

Real-Time Order Tracking: Displays live updates using WebSockets or Firebase.

  - **Backend (Server-Side) – Controller & Model**
    Technology Used: Node.js (Express.js)

Authentication Service: Handles secure login, registration,

Order Processing Service: Manages order placement, modifications, and updates.

Reservation System: Processes table bookings and manages availability.

Payment Gateway Integration: Connects with third-party payment providers like Stripe or PayPal.

Admin Dashboard: Allows restaurant owners and staff to manage orders, menus, and customers.

  - **Database Layer – Model**
    Technology Used: MongoDB

Users Database: Stores customer, staff, and administrator information.

Orders Database: Tracks order details, statuses, and transaction records.

Reservations Database: Manages table bookings and availability.

Menu Database: Stores menu items, pricing, and customizations.

- **System Interactions:**

1- User Registration & Authentication:

Customers, staff, and admins register/login using email, phone, or social

Secure authentication using JWT tokens.

Menu Browsing & Order Placement:

2- Customers browse restaurant menus and customize their orders.

The system validates order details and forwards them for processing.

Order Processing & Payment:

3- Order details are sent to the kitchen/staff dashboard.

Secure online payments are processed via third-party gateways.

Reservation Management:

4- Customers can select a date, time, and preferred table.

The system checks availability and confirms reservations.

Real-Time Order Tracking:

5- Customers receive updates on order status (Pending → Preparing → Out for Delivery).

Notifications are sent via WebSockets or Firebase.

Admin & Staff Management:

6- Admins can manage users, monitor system activity, and update menu items.

Staff can update order statuses and manage reservations.

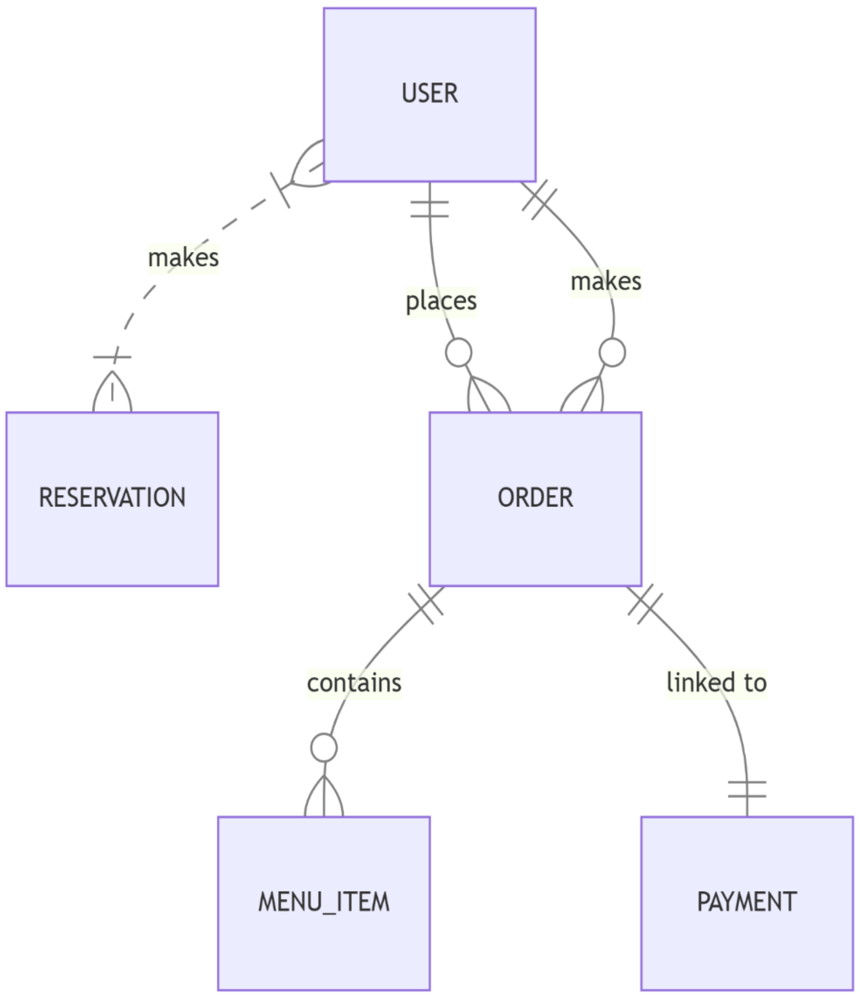
Architecture Style: Model-View-Controller (MVC)

The **MVC** architecture is chosen due to its structured separation of concerns:

- **Model (M):** Handles business logic, data storage, and interactions with the database.
- **View (V):** Manages the user interface and user interactions.
- **Controller (C):** Processes user requests, updates models, and determines appropriate responses.

4.2 Database Design & Data Modeling :

- **ER Diagram (Entity-Relationship Diagram) :**
  The **Entity-Relationship Diagram (ERD)** provides a structured representation of the restaurant website's database. It defines how different entities (tables) interact with each other and ensures efficient data storage, retrieval, and management.



- **Entities & Attributes**

  The database consists of six main entities: Users, Menu_Items, Orders, Order_Items, Reservations, and Payments.

  1) Users (Customers & Admins)

  Stores user details for customers and administrators.

- Primary Key: user_id
- Attributes:
  - name (VARCHAR) – Customer’s full name
  - email (VARCHAR, UNIQUE) – Login identifier
  - phone (VARCHAR) – Contact number
  - password (VARCHAR) – Encrypted password
  - role (ENUM: 'customer', 'admin') – User type

  Relationships:
- A User can place multiple Orders (One-to-Many).
- A User can make multiple Reservations (One-to-Many).

  2) Menu_Items (Restaurant’s Food & Drinks)

  Stores details of available food items.
- Primary Key: item_id
- Attributes:
  - name (VARCHAR) – Name of the dish
  - description (TEXT) – Short description of the item
  - price (DECIMAL) – Cost per item
  - category (VARCHAR) – Dish category (e.g., appetizer, main course)

  Relationships:
- A Menu Item can be part of multiple Orders (Many-to-Many through Order_Items).

  3) Orders (Customer Orders)

  Stores order details, linked to users and payments.
- Primary Key: order_id
- Foreign Key: user_id (References Users)
- Attributes:
  - order_date (DATETIME) – Timestamp of order placement
  - total_price (DECIMAL) – Total cost of the order
  - status (ENUM: 'Pending', 'Completed', 'Cancelled') – Order status

Relationships:

- An Order is placed by a User (One-to-Many).
- An Order contains multiple Menu Items (Many-to-Many through Order_Items).
- An Order has one Payment (One-to-One).

4) Order_Items (Connecting Orders & Menu_Items)

Handles the many-to-many relationship between Orders and Menu_Items.

- Primary Key: order_item_id
- Foreign Keys:
  - order_id (References Orders)
  - item_id (References Menu_Items)
- Attributes:
  - quantity (INT) – Number of times an item is ordered

Relationships:

- Connects Orders and Menu Items in a Many-to-Many relationship.

5) Reservations (Table Bookings)

Stores customer table reservations.

- Primary Key: reservation_id
- Foreign Key: user_id (References Users)
- Attributes:
  - date (DATE) – Reservation date
  - time (TIME) – Reservation time
  - guests (INT) – Number of guests

Relationships:

- A User can make multiple Reservations (One-to-Many).

6) Payments (Transaction Records)

Stores payment details for orders.

- Primary Key: payment_id
- Foreign Key: order_id (References Orders)
- Attributes:
  - payment_method (ENUM: 'Credit Card', 'PayPal', 'Cash') – Payment type
  - amount (DECIMAL) – Total amount paid
  - status (ENUM: 'Paid', 'Failed', 'Refunded') – Payment status

Relationships:

- A Payment is linked to one Order (One-to-One).

- **Logical & Physical Schema**

Logical Schema (Tables, Attributes, and Keys)

Below are the **main database tables** with **primary keys (PK)** and **foreign keys (FK)**.

Table Name	Attributes
User	user_id (PK), name, email, password, phone, role (customer/staff/admin)
Restaurant	restaurant_id (PK), name, location, contact, opening_hours
Menu	menu_id (PK), restaurant_id (FK), item_name, description, price, availability
Reservation	reservation_id (PK), user_id (FK), restaurant_id (FK), date, time, guests, status
Order	order_id (PK), user_id (FK), restaurant_id (FK), order_date, status, total_amount
Order_Item	order_item_id (PK), order_id (FK), menu_id (FK), quantity, subtotal
Payment	payment_id (PK), order_id (FK), payment_method, payment_status, transaction_date
Review	review_id (PK), user_id (FK), restaurant_id (FK), rating, comment, review_date

Relationships in Logical Schema

- **Users** → **Orders** (One-to-Many) → A user can place multiple orders.
- **Users** → **Reservations** (One-to-Many) → A user can make multiple reservations.
- **Orders** → **Order_Items** → **Menu_Items** (Many-to-Many) → An order contains multiple menu items.
- **Orders** → **Payments** (One-to-One) → Each order has one payment transaction.

- **Normalization Considerations**

To ensure data integrity and avoid redundancy, the database follows 3rd Normal Form (3NF) principles:

1. **1NF (First Normal Form)**
    - Each table has a **unique primary key**.
    - Data is **atomic** (each field has only one value).
2. **2NF (Second Normal Form)**
    - No **partial dependencies** (all non-key attributes depend on the **whole primary key**).
    - Example: `Order_Item` ensures each **order contains multiple items** but does not duplicate **menu details**.
3. **3NF (Third Normal Form)**
    - No **transitive dependencies** (no attribute depends on another non-key attribute).
    - Example: `Payment` table only stores **payment-specific details**, not order information.

4.3.Data Flow & System Behavior:

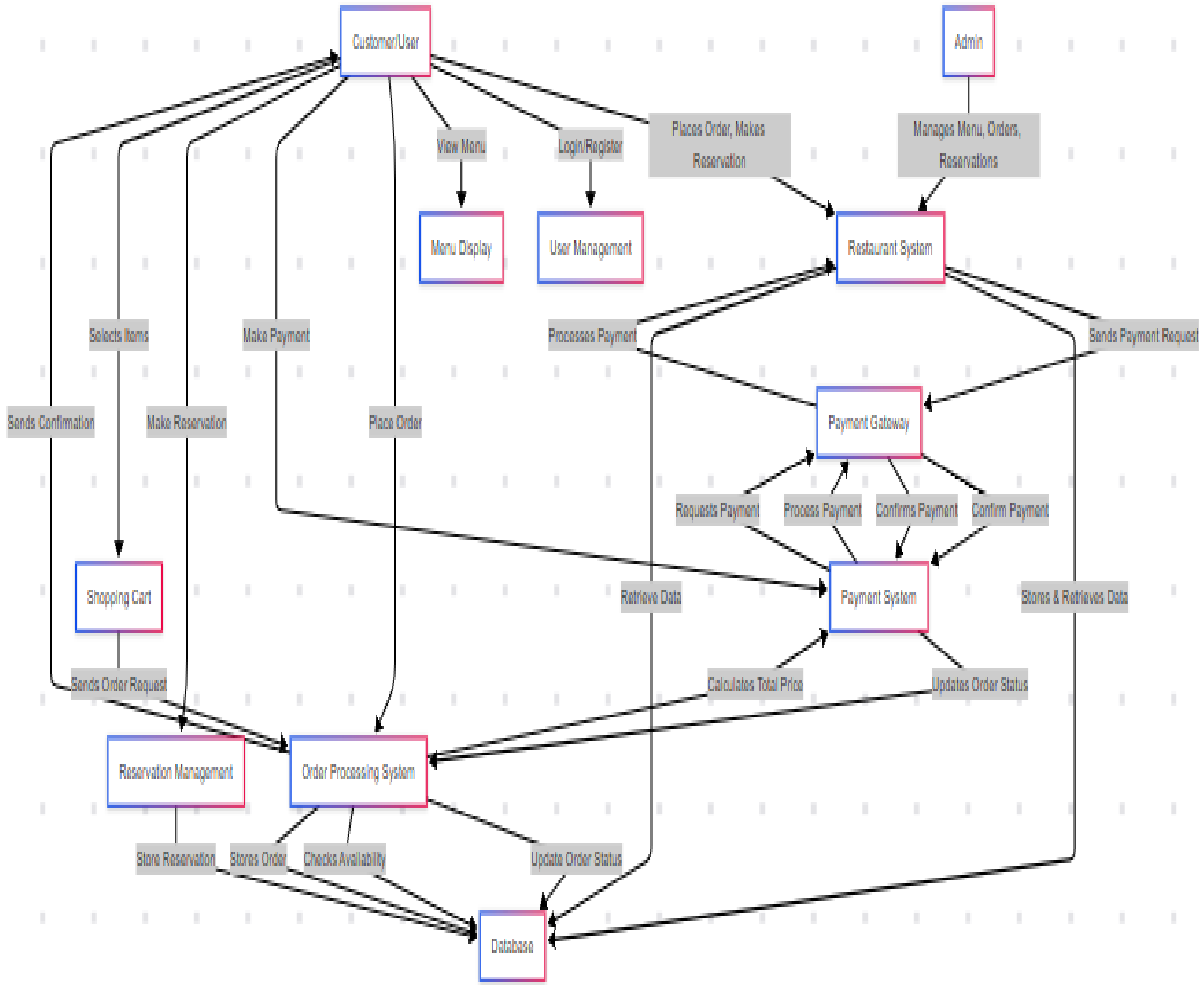
• DFD (Data Flow Diagram)

A **Data Flow Diagram (DFD)** is used to **visualize how data moves through the system**. It helps in understanding:

-Data sources and destinations

-How data is processed at each stage

-Interactions between different components



DFD Level 0 (Context-Level Diagram)

Purpose: Represents the entire Restaurant System as a single entity interacting with users and external systems.

Entities & Their Roles

- **Customer (User)** → Browses menu, places orders, makes reservations, and makes payments.
- **Admin** → Manages menu, orders, and reservations.
- **Payment Gateway** → Processes payments securely.
- **Database** → Stores user details, menu items, orders, and reservations.

Data Flow Description

1-**Customers** interact with the **Restaurant System** to place orders and make reservations.
2-The **Restaurant System** retrieves and updates data from the **Database**.
3-The **Admin** manages the **menu, orders, and reservations** via the **Restaurant System**.
4-The **Payment Gateway** handles **payment transactions** and updates order statuses.

DFD Level 1 (Detailed Breakdown of System Components)

Purpose: Breaks down the Restaurant System into subsystems such as User Management, Order Processing, and Payment Handling.

Subsystems & Their Roles

User Management → Handles user registration, login, and authentication.
Menu Management → Retrieves and displays menu items from the database.
Order Processing → Handles order placement, updates order status, and interacts with the payment system.

Reservation Handling → Allows customers to book tables and updates reservations in the database.

Payment Processing → Sends payment details to the **Payment Gateway** and confirms transactions.

Data Flow Description

1-Customers log in or register via the User Management System.
2-They browse the Menu and select items for ordering.
3-The Order Processing System stores order details in the Database.
4-The Payment System interacts with the Payment Gateway to process payments.
5-The Reservation System updates reservation details in the Database.

DFD Level 2 (Detailed Process Flow for Order Processing & Payment Handling)

Purpose: Further breakdown of the Order Processing and Payment Handling System to show step-by-step data flow.

Entities & Processes

-**Customer** places an order → Sends request to **Order System**.
-**Order System** verifies stock and calculates total → Updates **Database**.
-**Payment System** requests payment from **Payment Gateway** → Gets confirmation.
-**Order System** updates order status → Notifies the **Customer**.

• Sequence Diagrams

A **Sequence Diagram** shows how **different entities interact** in a step-by-step manner. It includes:

Actors (Users, Admins, System, Payment Gateway, Database, etc.) Objects (Restaurant System, Order Processing, Payment Processing, etc.) Messages exchanged between components

• Key Processes & Sequence Diagrams

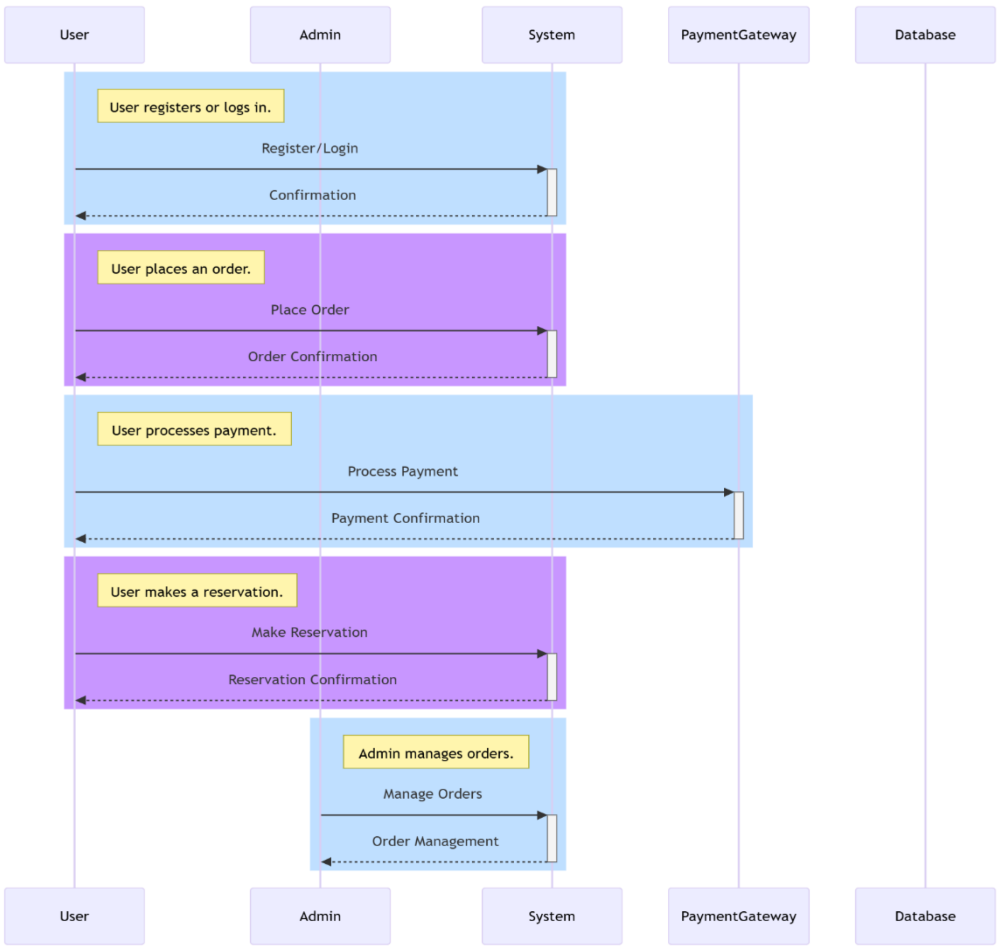
Below are the key interactions that happen in the restaurant system:
1-User Registration & Login

2-Placing an Order

3-Payment Processing

4-Making a Reservation

5-Admin Managing Orders



• Activity Diagram

An Activity Diagram represents the flow of control in a system. It shows: User interactions with the system.

Step-by-step processes like ordering food, making reservations, and payments.
Decision points and alternate paths.

• Key Workflows & Activity Diagrams

Below are the main workflows of the system:

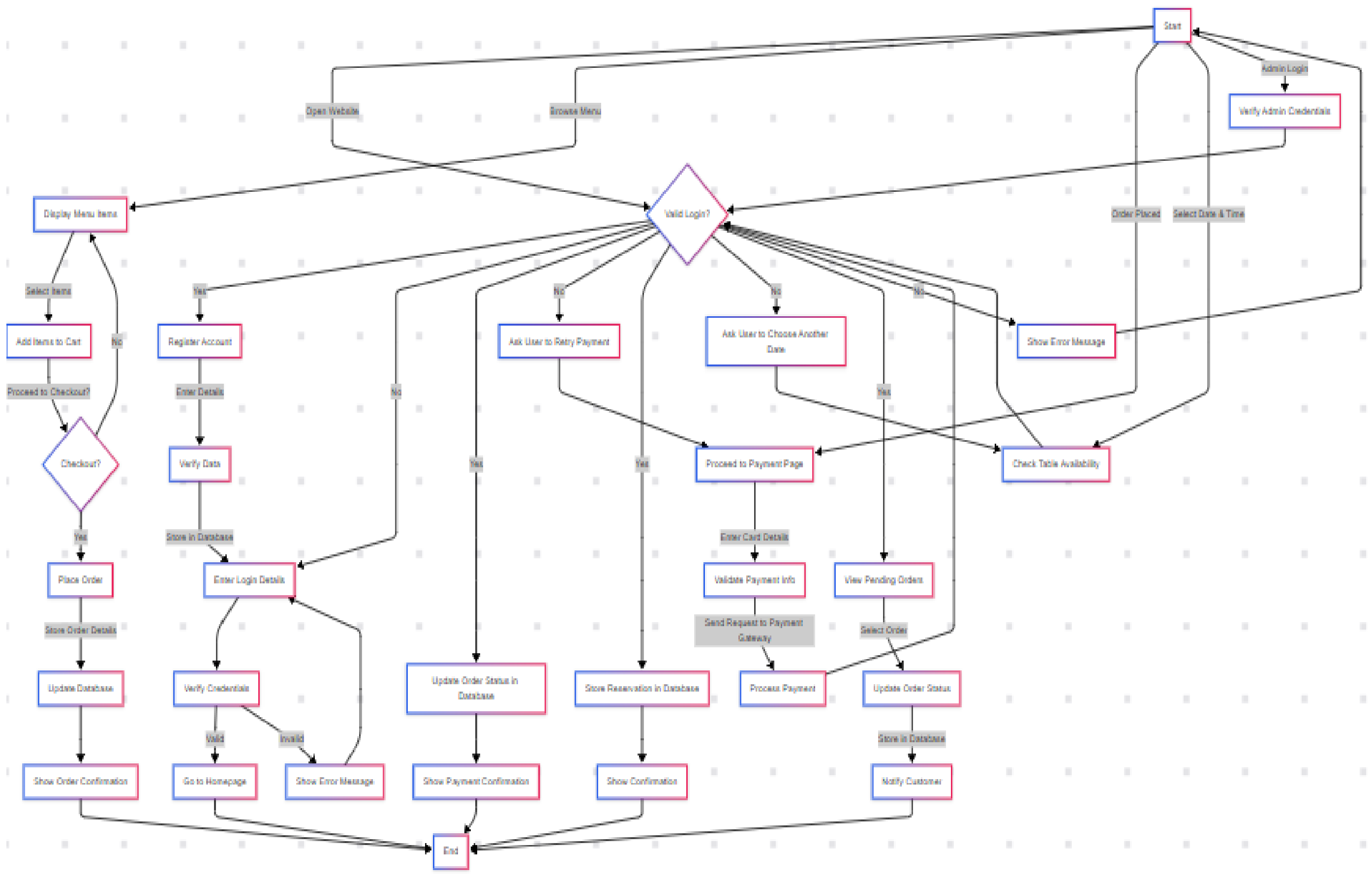
1-User Registration & Login

2-Placing an Order

3-Payment Processing

4-Making a Reservation

5-Admin Managing Orders



• State Diagram

A State Diagram represents how an object (e.g., Order, User, Payment, or Reservation) transitions between states based on different actions or events.

Objects in the system change states (e.g., an order moves from "Pending" to "Completed").
Arrows show state transitions caused by events or actions.
Useful for understanding business logic in the system.

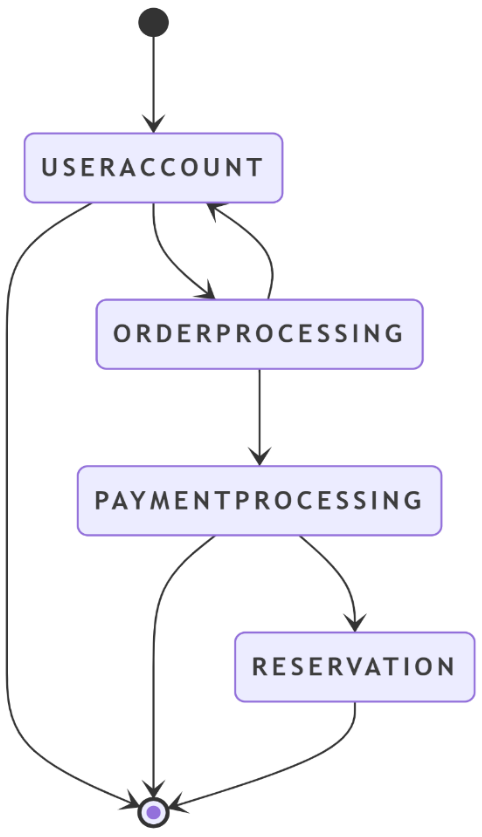
- **Key State Diagrams**

Below are the main state transitions in the restaurant system:
1-User Account Lifecycle

2-Order Processing Lifecycle

3-Payment Processing Lifecycle

4-Reservation Lifecycle



• Class Diagram

A Class Diagram represents**:**
Classes (entities) in the system
Attributes (data fields) and Methods (functions) for each class
Relationships (e.g., one-to-one, one-to-many, many-to-many) between classes

- ### Key Classes & Relationships

Below are the main classes in the restaurant system:

1-User – Customers and Admins who interact with the system

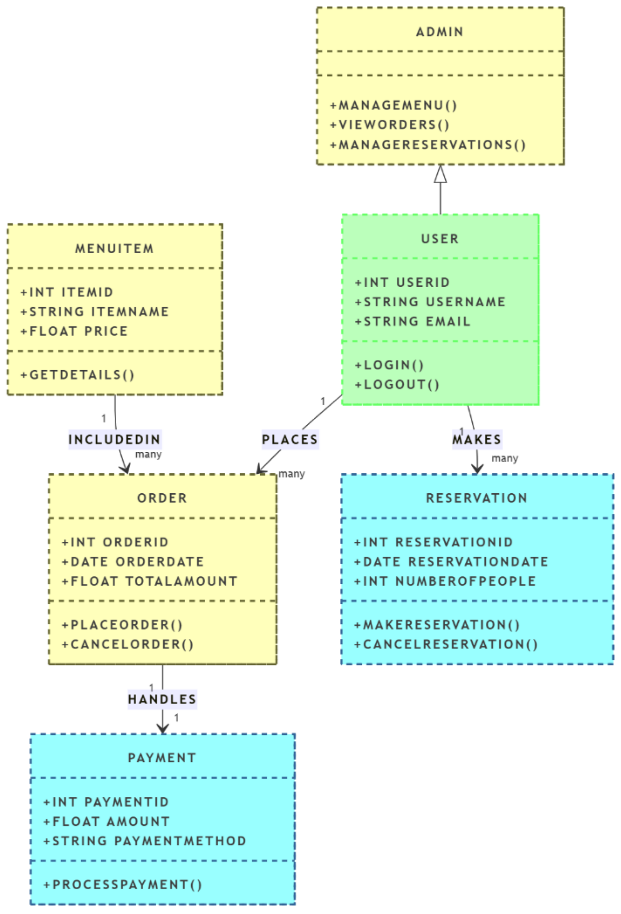
2-Order – Represents a food order placed by a user

3-MenuItem – Represents a food item available for ordering

4-Reservation – Represents a table booking by a user

5-Payment – Handles transactions for food orders

6-Admin – Manages menu, orders, and reservations



4.4 UI/UX Design & Prototyping

Our Idea Based On : Here

Git Hub Repository Here

• Wireframes & Mockups

Key Pages & Wireframes

📌 Home Page

- Header: Logo, navigation menu (Home, Menu, About, Reservations, Contact).
- Hero Section: Restaurant image, CTA button (Book a Table / Order Online).
- Featured Dishes: Display popular menu items.
- Footer: Social media links, contact details.

📌 Menu Page

- Food Categories: Starters, Main Course, Desserts, Beverages.
- Add to Cart: Users can add items to order.

📌 Reservations Page

- Date & Time Picker: Users select a reservation date & time.
- Guest Count Selection: Number of guests.
- Confirmation & Email Notification.

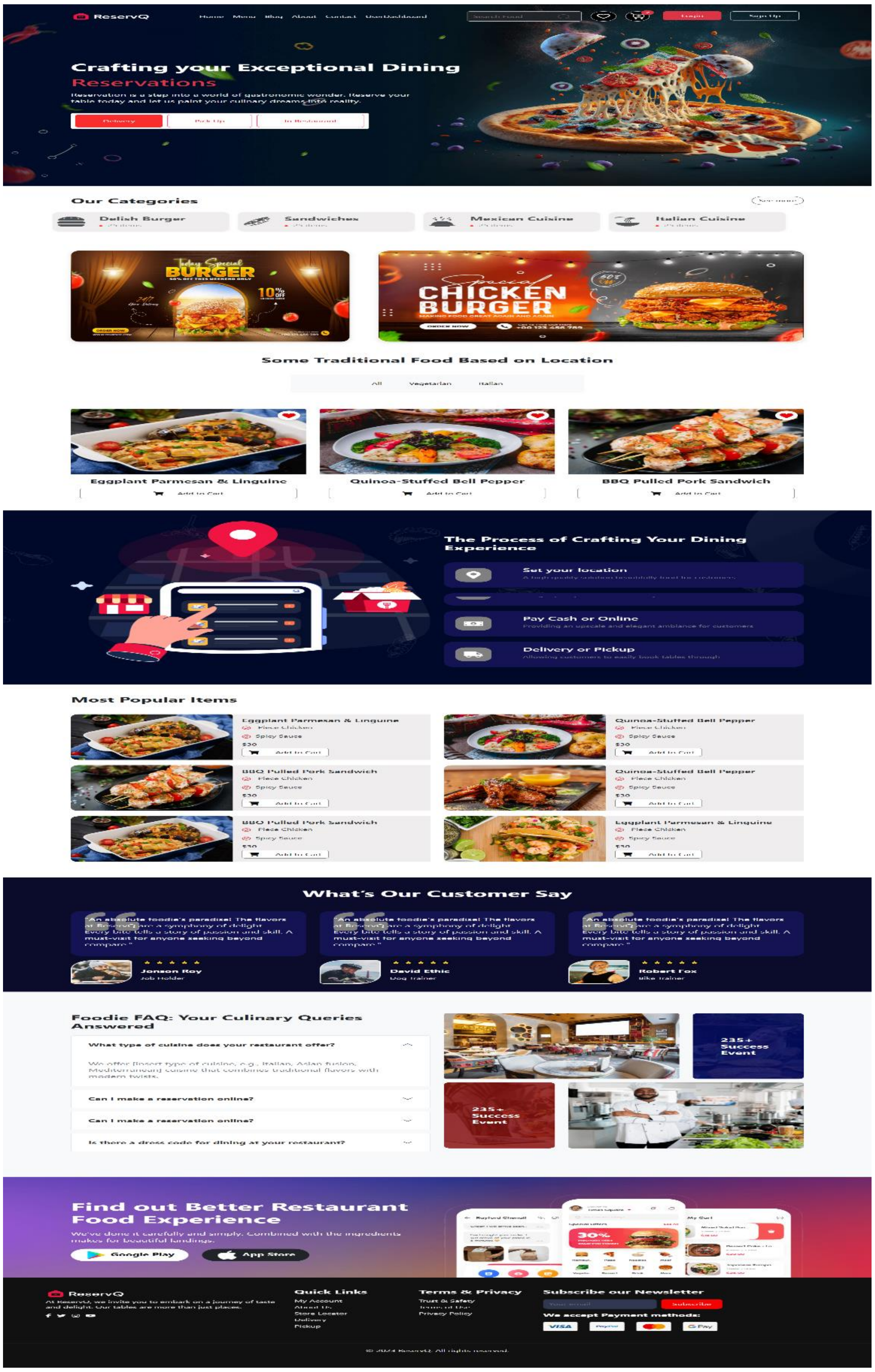
📌 Order Checkout Page

- Cart Summary: Lists items added to the order.
- Delivery/Pickup Option: Choose between dine-in, takeaway, or delivery.
- Payment Gateway Integration: Card, PayPal, Cash on Delivery.

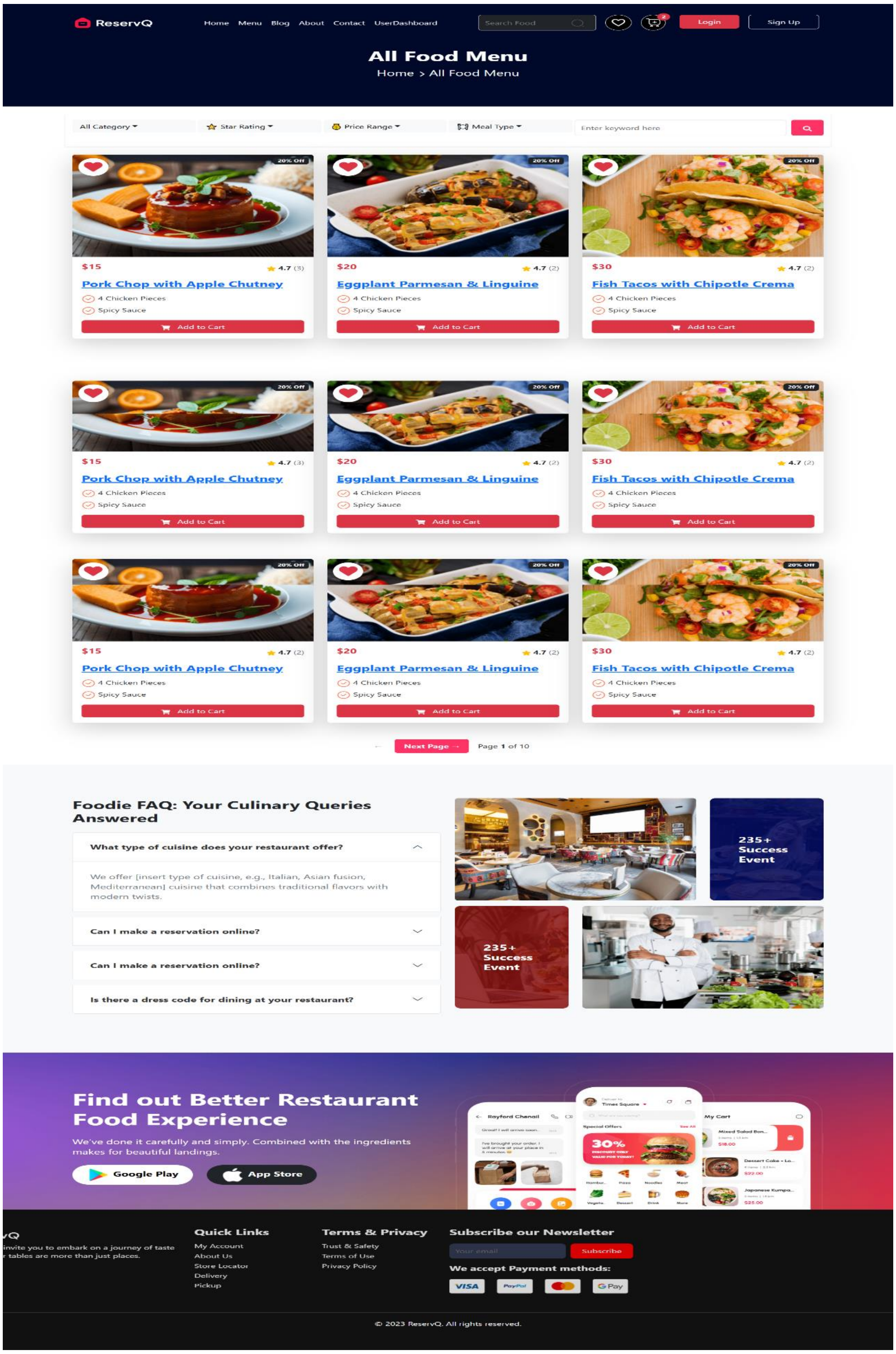
📌 Contact Page

- Restaurant Location Map.
- Phone Number, Email, and Social Links.

This is the home page of our website:


ReservQ
Home Menu Blog About Contact UserDashboard
Login
Sign Up
Crafting your Exceptional Dining
Reservations
Reservation is a step into a world of gastronomic wonder. Reserve your table today and let us paint your culinary dreams into reality.
Delivery
Pick Up
In Restaurant
Our Categories
Delish Burger
Sandwiches
Mexican Cuisine
Italian Cuisine
Today Special BURGER
Special CHICKEN BURGER
Some Traditional Food Based on Location
All
Vegetarian
Italian
Eggplant Parmesan & Linguine
Quinoa-Stuffed Bell Pepper
BBQ Pulled Pork Sandwich
Add to Cart
The Process of Crafting Your Dining Experience
Set your location
Pay Cash or Online
Delivery or Pickup
Most Popular Items
Eggplant Parmesan & Linguine
Quinoa-Stuffed Bell Pepper
BBQ Pulled Pork Sandwich
Piece Chicken
Spicy Sauce
$30
What's Our Customer Say
"An absolute foodie's paradise! The flavors at ReservQ are a symphony of delight. Every bite tells a story of passion and skill. A must-visit for anyone seeking beyond compare."
Jonson Roy
David Ethic
Robert Fox
Foodie FAQ: Your Culinary Queries Answered
What type of cuisine does your restaurant offer?
Can I make a reservation online?
Is there a dress code for dining at your restaurant?
235+ Success Event
Find out Better Restaurant Food Experience
Google Play
App Store
ReservQ
Quick Links
Terms & Privacy
Subscribe our Newsletter
We accept Payment methods:
VISA
© 2024 ReservQ. All rights reserved.

This is the menu page of our website:



ReservQ

Home Menu Blog About Contact UserDashboard

Search Food

Login Sign Up

All Food Menu

Home > All Food Menu

All Category ▾ Star Rating ▾ Price Range ▾ Meal Type ▾ Enter keyword here

20% Off
$15 ★ 4.7 (3)
Pork Chop with Apple Chutney
4 Chicken Pieces
Spicy Sauce
Add to Cart

20% Off
$20 ★ 4.7 (2)
Eggplant Parmesan & Linguine
4 Chicken Pieces
Spicy Sauce
Add to Cart

20% Off
$30 ★ 4.7 (2)
Fish Tacos with Chipotle Crema
4 Chicken Pieces
Spicy Sauce
Add to Cart

20% Off
$15 ★ 4.7 (3)
Pork Chop with Apple Chutney
4 Chicken Pieces
Spicy Sauce
Add to Cart

20% Off
$20 ★ 4.7 (2)
Eggplant Parmesan & Linguine
4 Chicken Pieces
Spicy Sauce
Add to Cart

20% Off
$30 ★ 4.7 (2)
Fish Tacos with Chipotle Crema
4 Chicken Pieces
Spicy Sauce
Add to Cart

20% Off
$15 ★ 4.7 (2)
Pork Chop with Apple Chutney
4 Chicken Pieces
Spicy Sauce
Add to Cart

20% Off
$20 ★ 4.7 (2)
Eggplant Parmesan & Linguine
4 Chicken Pieces
Spicy Sauce
Add to Cart

20% Off
$30 ★ 4.7 (2)
Fish Tacos with Chipotle Crema
4 Chicken Pieces
Spicy Sauce
Add to Cart

← Next Page → Page 1 of 10

Foodie FAQ: Your Culinary Queries Answered

What type of cuisine does your restaurant offer?

We offer [insert type of cuisine, e.g., Italian, Asian fusion, Mediterranean] cuisine that combines traditional flavors with modern twists.

Can I make a reservation online?

Can I make a reservation online?

Is there a dress code for dining at your restaurant?

235+ Success Event

235+ Success Event

Find out Better Restaurant Food Experience

We've done it carefully and simply. Combined with the ingredients makes for beautiful landings.

Google Play App Store

vQ
invite you to embark on a journey of taste
r tables are more than just places.

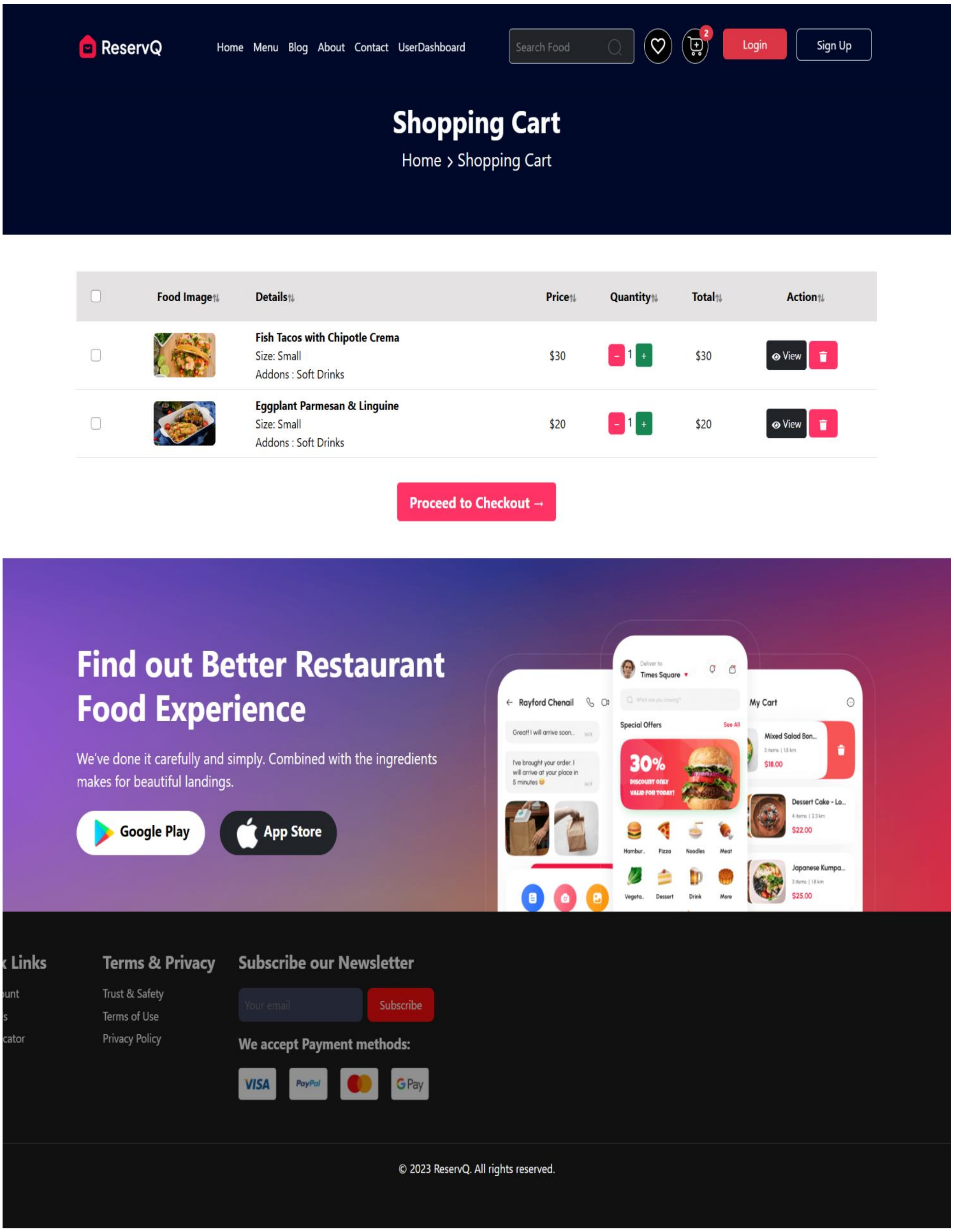
Quick Links
My Account
About Us
Store Locator
Delivery
Pickup

Terms & Privacy
Trust & Safety
Terms of Use
Privacy Policy

Subscribe our Newsletter
Your email Subscribe
We accept Payment methods:
VISA PayPal G Pay

© 2023 ReservQ. All rights reserved.

This is the shopping cart of our website:



ReservQ Home Menu Blog About Contact UserDashboard Search Food Login Sign Up

Shopping Cart

Home › Shopping Cart

	Food Image	Details	Price	Quantity	Total	Action
☐		Fish Tacos with Chipotle Crema Size: Small Addons : Soft Drinks	$30	- 1 +	$30	View
☐		Eggplant Parmesan & Linguine Size: Small Addons : Soft Drinks	$20	- 1 +	$20	View

Proceed to Checkout →

Find out Better Restaurant Food Experience

We've done it carefully and simply. Combined with the ingredients makes for beautiful landings.

Google Play App Store

k Links

unt

s

cator

Terms & Privacy

Trust & Safety

Terms of Use

Privacy Policy

Subscribe our Newsletter

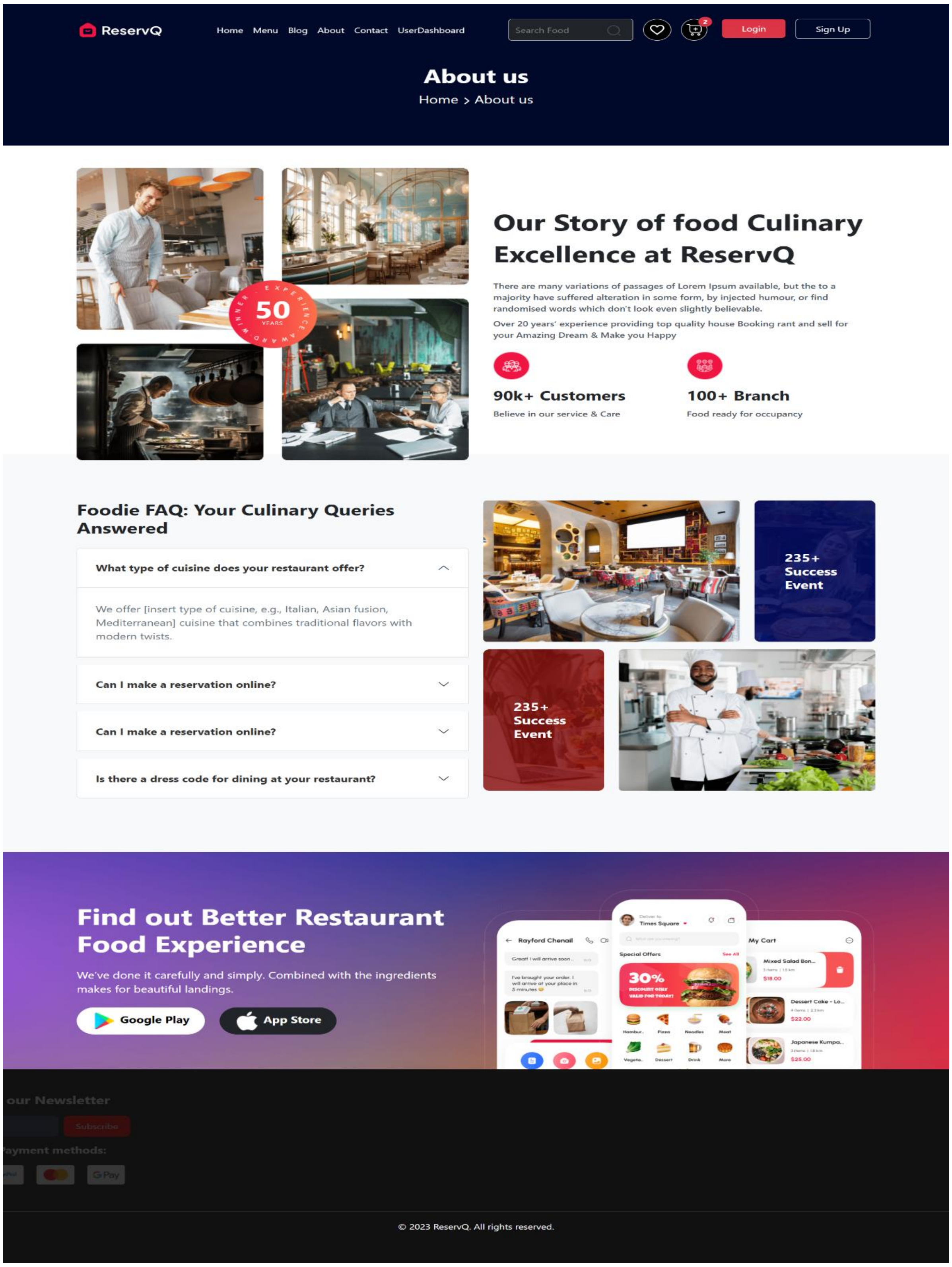
Your email Subscribe

We accept Payment methods:

VISA PayPal GPay

© 2023 ReservQ. All rights reserved.

This is the about us page of our website:



About us

Home > About us

Our Story of food Culinary Excellence at ReservQ

There are many variations of passages of Lorem Ipsum available, but the to a majority have suffered alteration in some form, by injected humour, or find randomised words which don't look even slightly believable.

Over 20 years' experience providing top quality house Booking rant and sell for your Amazing Dream & Make you Happy

90k+ Customers
Believe in our service & Care

100+ Branch
Food ready for occupancy

Foodie FAQ: Your Culinary Queries Answered

What type of cuisine does your restaurant offer?

We offer [insert type of cuisine, e.g., Italian, Asian fusion, Mediterranean] cuisine that combines traditional flavors with modern twists.

Can I make a reservation online?

Can I make a reservation online?

Is there a dress code for dining at your restaurant?

Find out Better Restaurant Food Experience

We've done it carefully and simply. Combined with the ingredients makes for beautiful landings.

our Newsletter

Subscribe

Payment methods:

© 2023 ReservQ. All rights reserved.

4.5. System Deployment & Integration

- **Technology Stack**

The technology stack includes the backend, frontend, and database used to develop the restaurant website.

-Frontend (Client-Side)

Technology: React.js (ReservQ Template)
React.js – Fast, component-based UI rendering.
Tailwind CSS – Modern styling with flexibility.
Redux – State management for user authentication & cart.

-Backend (Server-Side)

Technology: Node.js
Node.js – Scalable and non-blocking runtime for backend logic.

-Database

Technology: MongoDB (NoSQL Database)
Stores menu items, orders, users, and reservations.
Fast retrieval with flexible schema for menu updates.

-Hosting & Deployment

Technology: Vercel (Frontend) & AWS (Backend & Database)
Frontend: Deployed on Vercel for seamless React hosting.
Backend: Hosted on AWS EC2 or DigitalOcean.
Database: Hosted on MongoDB Atlas (Cloud-based NoSQL).

- **Deployment Diagram**

A Deployment Diagram illustrates how software components (frontend, backend, database) are distributed across different hardware.

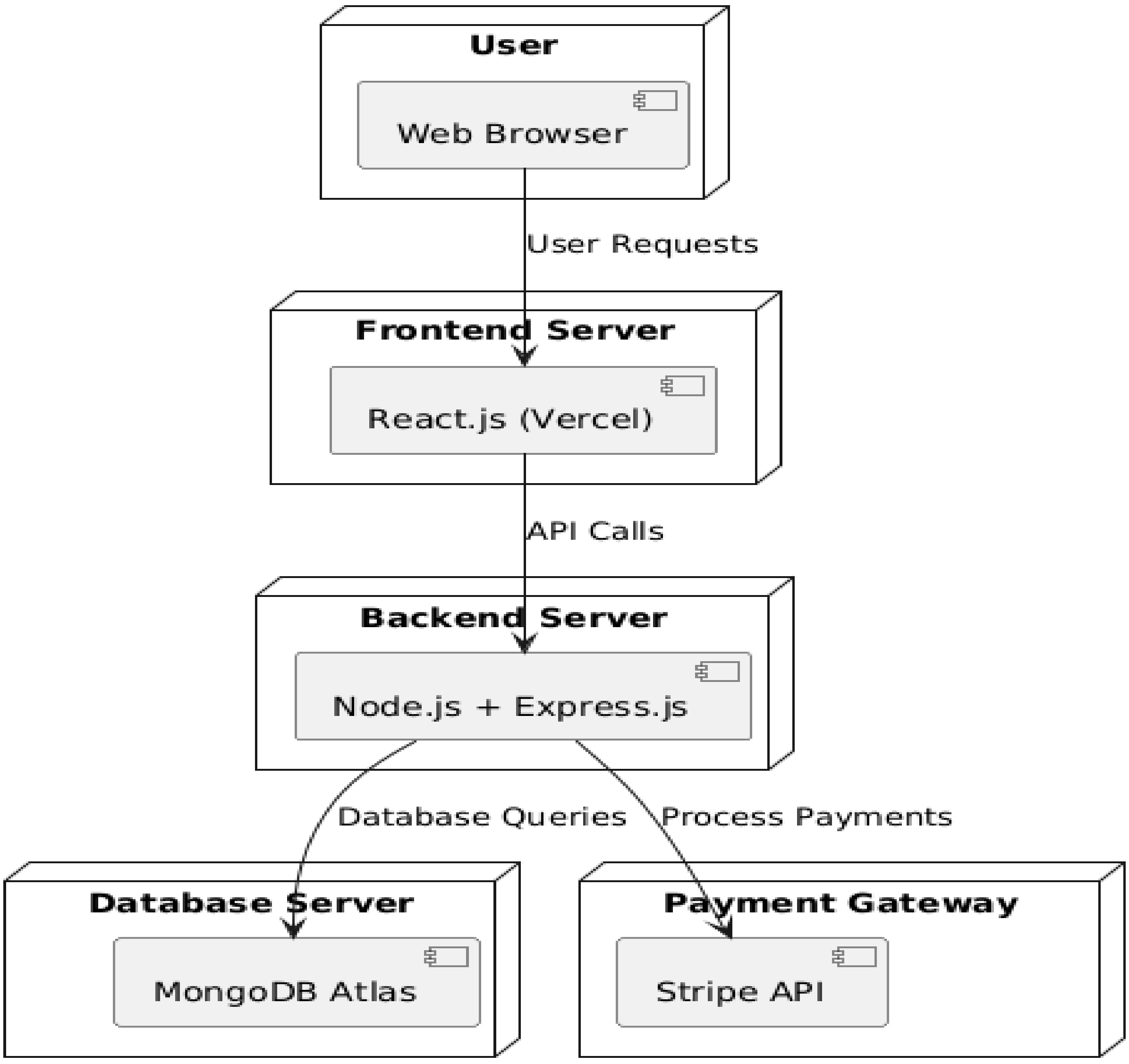
Explanation:

1-Users access the restaurant website from a browser.

2-The frontend (React.js) sends API requests to the backend.

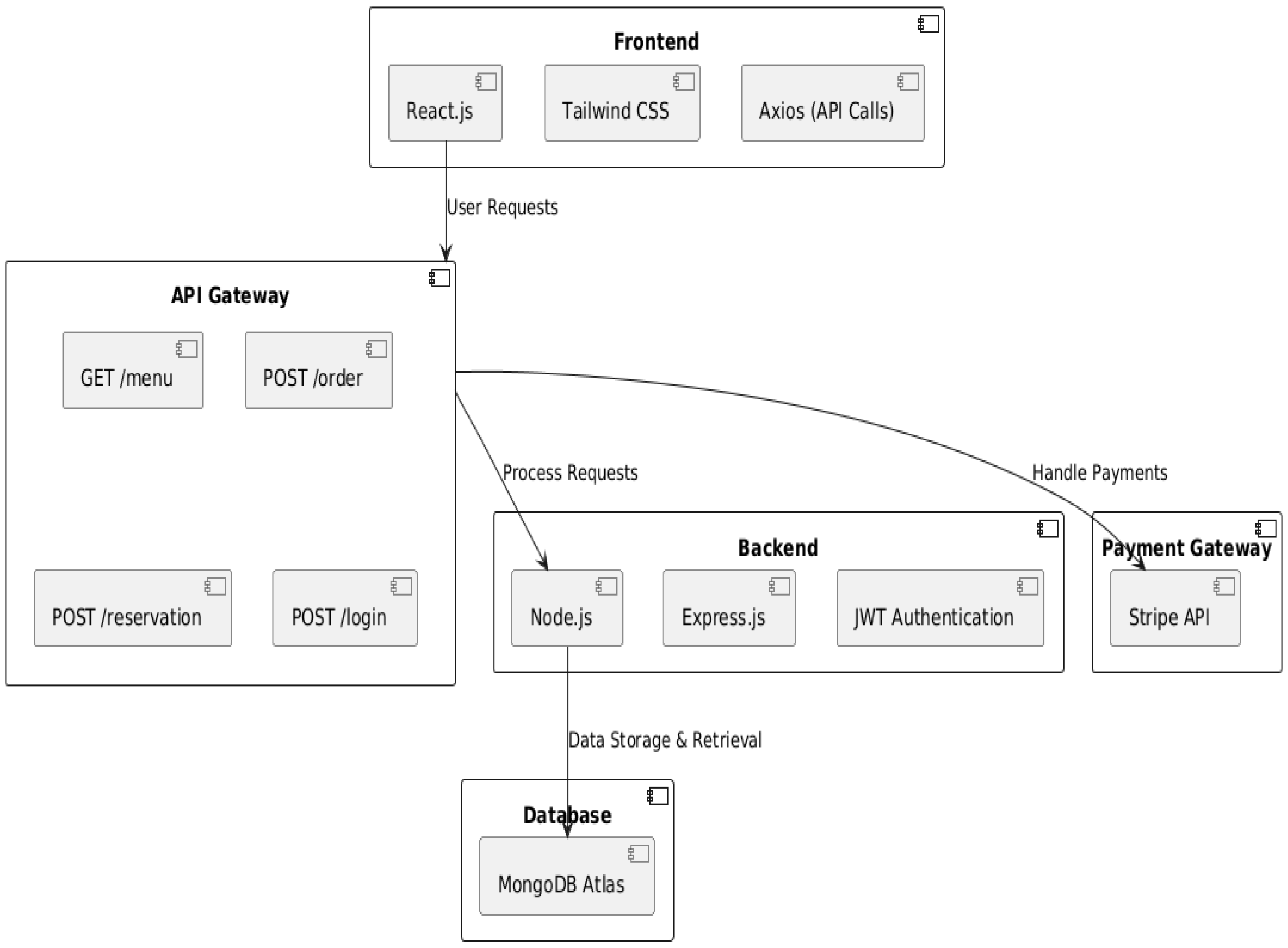
3-The backend (Node.js & Express.js) processes requests (menu retrieval, orders, reservations).

4-The MongoDB Atlas database stores and retrieves data.



- **Component Diagram**

A Component Diagram shows the high-level structure of system components and their dependencies.



Explanation of Components & Dependencies:

1-**Frontend (React.js)**

- Fetches menu items and order history from the backend.
- Sends new orders and reservations via API requests.

2-**Backend (Node.js & Express.js)**

- Handles business logic (authentication, orders, reservations).
- Interacts with MongoDB Atlas for data storage.

3-**Database (MongoDB)**

- Stores Users, Menu Items, Orders, and Reservations.

4-**API Gateway**

- Provides endpoints:
  GET /menu – Fetch menu items.
  POST /order – Place an order.
  POST /reservation – Book a table.
  POST /login – Authenticate users.

5-**Payment Gateway (Stripe)**

- Processes payments when an order is placed.