Scan2Order - Smart Restaurant Solution

A PROJECT REPORT OF

PROJECT – I (PROJ - CSE - 322G)

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

BACHELOR OF TECHNOLOGY

COMPUTER ENGINEERING

SUBMITTED BY	**REGISTRATION NO.**
Puneet Chauhan(22CE015)	2210071214

SUPERVISOR

MS.Ritu Arora

(ASSISTANT PROFESSOR)

DEPARTMENT OF COMPUTER ENGINEERING

THE TECHNOLOGICAL INSTITUTE OF TEXTILE & SCIENCES, BHIWANI

HARYANA– 127021

(2024-2025)

Vision of the Institution

To Become Excellent Knowledge Enterprise

Mission of the Institution

Creation of Centre of Excellence for Learning & Research in Engineering and Technology

Vision of the Department

To Produce Professionally Excellent, Knowledgeable, Globally Competitive and Socially Responsible Computer Engineers and Entrepreneurs.

Mission of the Department

1. To provide quality education to students of computer engineering.
2. To establish an Industry Institute interaction, participation and collaboration to various job-oriented skills in computer engineering.
3. To build human values, social values, entrepreneurship skills and professional ethics among the IT technocrats.
4. To focus on innovative and latest development of technologies by engaging in various research areas.

TABLE OF CONTENTS

DECLARATION

I, **Puneet Chauhan (22CE015)**, hereby declare that the project work entitled **“Scan2Order - QR Based Restaurant Ordering System”**, submitted in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Engineering** to the **Department of Computer Engineering**, **The Technological Institute of Textile & Sciences, Bhiwani**, is a record of my own original work carried out under the valuable guidance and supervision of **MS. Ritu Arora**.

I further declare that this project work has been carried out with utmost sincerity, dedication, and honesty.
The matter presented in this report has not been submitted by me or anyone else, either fully or partially, for the award of any degree, diploma, or any other academic recognition elsewhere.

Wherever contributions of others have been taken, every effort has been made to clearly acknowledge and reference such sources within the report.
I fully understand that in case any part of this work is found to be plagiarized or copied, I shall be responsible for the consequences as per institutional guidelines.

This project reflects my own research, understanding, development, and practical implementation of the Scan2Order system in the real-world scenario.

Signature of Student: **University Roll No:**

Puneet Chauhan (22CE015) Puneet Chauhan (2061019)

CERTIFICATE

This is to certify that the project report titled "Scan2Order - QR Based Restaurant Ordering System", submitted by Puneet Chauhan (22CE015) , is an authentic record of their own work carried out in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Engineering at The Technological Institute of Textile & Sciences, Bhiwani, Haryana.

The project work has been conducted with sincerity and dedication under the valuable supervision and guidance of MS. Ritu Arora, Assistant Professor, Department of Computer Engineering.

It is further certified that the work embodied in this project report has not been submitted, either fully or partially, to any other university or institution for the award of any degree or diploma.

The project reflects the genuine efforts, hard work, innovative thinking, and technical knowledge acquired by the students during their study period.
The contributions made by them towards the successful design, development, and implementation of the Scan2Order system are commendable and meet the standards expected at this level of academic pursuit.

Project Guide:

Ms. Ritu Arora
(Assistant Professor)
Department of Computer Engineering

Dr. Jyoti Chaudhary (Head of Department)
Department of Computer Engineering
The Technological Institute of Textile & Sciences, Bhiwani

ACKNOWLEDGEMENT

I would like to express my profound gratitude and deep regards to my project guide **Ms. Ritu Arora**, Assistant Professor, Department of Computer Engineering, for her invaluable guidance, encouragement, and dedicated support throughout the duration of this project work titled **"Scan2Order - QR Based Restaurant Ordering System."**
Her expert advice, timely feedback, and critical suggestions were pivotal for the successful completion of this project.

I am also extremely thankful to **Dr. Jyoti Chaudhary**, Head of Department, Computer Engineering, for her constant support, motivation, and for providing all necessary resources and facilities required for the successful execution of this work.

I extend my sincere thanks to the entire faculty and staff members of the **Department of Computer Engineering** at **The Technological Institute of Textile & Sciences, Bhiwani**, for their valuable support, encouragement, and assistance at various stages of this project.

I am deeply grateful to my parents and family for their continuous encouragement, moral support, and blessings, which kept me motivated throughout my academic journey.

Lastly, I would like to thank my friends and peers for their constant cooperation, brainstorming sessions, and insightful discussions that contributed immensely to this project's completion.

This project would not have been possible without the collective support of all the above-mentioned individuals.

Signature Of Student

Puneet Chauhan (22CE015)

ABSTRACT

The project titled "Scan2Order - QR Based Restaurant Ordering System" is developed to revolutionize the traditional restaurant ordering process by introducing a contactless, efficient, and modern solution.

The system enables customers to simply scan a QR code placed on their dining table, which leads them to a dynamic digital menu. Customers can browse available food items, customize their orders, add items to the cart, and place their orders directly from their mobile devices without any need for physical interaction or staff dependency.

The project focuses on enhancing customer experience, reducing waiting times, minimizing human errors in order-taking, and improving overall operational efficiency of restaurants.

The Admin Panel built within the system allows restaurant managers to manage menus, tables, and monitor live orders in real-time.

Oracle APEX serves as the core development platform, providing a robust backend, secure authentication mechanisms, and seamless database connectivity.

Additionally, WebSocket communication is planned for real-time order updates, and RESTful APIs integration is designed to extend future capabilities like online payments and notification systems.

The system incorporates strict Authentication and Authorization methods to ensure secure access for different user roles.

Scan2Order is designed to be scalable, user-friendly, and adaptable to the growing needs of modern restaurants.

It not only meets the current demand for hygienic, contactless services but also lays a strong foundation for future enhancements such as mobile payment integration, loyalty rewards, and customer feedback collection.

This project reflects the potential of blending technology with hospitality to create smarter, faster, and more customer-centric dining experiences.

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Expansion
UI	User Interface
UX	User Experience
API	Application Programming Interface
SQL	Structured Query Language
HTML5	HyperText Markup Language
CSS	Cascading Style Sheets
JS	JavaScript
QR	Quick Response
DB	Database
CRUD	Create, Read, Update, Delete
REST	Representational State Transfer
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
JWT	JSON Web Token (optional for token authentication)
OTP	One Time Password (for future secure login)
APEX	Oracle Application Express
ERD	Entity Relationship Diagram
TCP	Transmission Control Protocol
AJAX	Asynchronous JavaScript and XML

CHAPTER 1

1.INTRODUCTION TO SCAN2ORDER

- **1 Introduction**

Scan2Order - QR Based Restaurant Ordering System is a web-based solution designed to simplify and digitize the restaurant ordering process.
The system eliminates the need for physical menus and manual order-taking by allowing customers to scan a QR code placed on their table using their smartphones.

After scanning the QR code, customers are redirected to a dynamic digital menu where they can browse food categories, add items to their cart, and place their order directly from their device.
The placed orders are instantly reflected in the backend system, visible to the kitchen staff and billing counter for processing.

Scan2Order ensures a faster, error-free, and more organized method of handling customer orders while reducing dependency on manual efforts.

The system is built using **Oracle APEX** for rapid backend development, **HTML, CSS, and JavaScript** for the front-end interface, and **SQL Database** for secure data storage.
It also incorporates **Authentication and Authorization** mechanisms to restrict access to admin functionalities and **dynamic QR code generation** for linking specific tables.

Scan2Order is designed to improve the efficiency of restaurant operations, minimize human errors, and deliver a quick and seamless ordering experience to customers

1.2 Key Highlights of Scan2Order

- **Contactless Ordering: Entire menu access and ordering happen digitally via mobile phones.**
- **Dynamic Menu: Items, prices, and availability can be updated in real-time from the admin panel.**
- **Cart Management: Customers can add multiple items, increase/decrease quantities, and review their cart before ordering.**
- **Live Order Updates: Orders are immediately reflected on the kitchen/billing side for quicker preparation.**
- **Admin Dashboard: A secure backend for restaurant staff to manage menu, view orders, and monitor table statuses.**

  Responsive Design: Works on all devices — mobiles, tablets, and desktops without needing an app.
- **Authentication and Authorization: Only authorized users (admins) can manage backend operations securely.**

1.3 Objective of Scan2Order

The objectives of the Scan2Order project are:

- **To digitize the menu browsing and order placement process in restaurants.**
- **To minimize human contact and promote hygienic dining experiences.**
- **To speed up the order processing time.**
- **To reduce manual errors caused by verbal/written orders.**
- **To allow restaurants to dynamically update their menus anytime without reprinting.**
- **To build a secure, easy-to-manage admin panel for restaurant management.**

1.4 Technologies Used

- **Oracle APEX - Application development platform for backend UI, logic, and database operations.**

- **HTML5, CSS3, JavaScript - For front-end development to create mobile-responsive, interactive web pages.**

- **SQL Database - For storing menu items, order records, user authentication information.**

- **QR Code API - For generating unique QR codes for each table.**

- **WebSocket (Optional future upgrade) - For real-time order status updates without refreshing pages.**

- **RESTful APIs (Optional integration) - For future payment gateway and notifications integration.**

1.5 Workflow of Scan2Order

The working of Scan2Order involves the following simple steps:

1. **Customer Scans QR Code at the table.**

2. **Digital Menu Opens on the customer’s mobile browser.**

3. **Customer Browses Menu items by categories.**

4. **Adds Items to Cart and modifies quantity if required.**

5. **Places Order with a single click.**

6. **Order Notification Appears instantly on the restaurant’s kitchen/billing screen.**

7. **Kitchen Prepares Food according to received order.**

8. **Food is Served to the correct table number.**

1.6 Advantages of Scan2Order

Aspect	Traditional System	Scan2Order System
Menu Access	Physical Menus	Digital Menu
Order Placement	Manual/Waiter	Customer's Mobile
Hygiene	Low (shared menus)	High (contactless)
Order Errors	High	Very Low
Menu Updates	Costly Reprinting	Instant Updates
Customer Waiting Time	Longer	Reduced

1.7 Future Enhancements

- Integration with popular **payment gateways** for online payments.
- **Customer Feedback** form after order completion.
- **Loyalty Programs** and rewards for frequent customers
.
- **Real-Time Order Tracking** from customer side.
- **Multi-Branch Support** for restaurant chains.

1.8 Summary

Scan2Order offers a practical, scalable, and futuristic solution to traditional restaurant operations.

It transforms the dining experience by enabling **contactless**, **error-free**, **fast**, and **efficient** order processing.

Designed for today's tech-savvy customers and competitive restaurant environments,

Scan2Order ensures that both customers and restaurant owners benefit from a streamlined, enjoyable service experience.

CHAPTER 2 : TECHNOLOGY USED

HTML5

2.2.1 Overview

HTML5 (HyperText Markup Language version 5) is the standard language for structuring and presenting content on the web.

It introduces semantic elements, multimedia support, and better form controls which make web applications easier to develop and maintain.

2.2.2 Key Features

- **Semantic Tags:** `<section>`, `<article>`, `<nav>`, `<footer>`.
- **Form Enhancements:** New input types like `email`, `date`, `number`.
- **Audio and Video Support:** `<audio>`, `<video>` tags.
- **Local Storage:** For offline web applications.

```
● <!DOCTYPE html>
● <html lang="en">
● <head>
●   <meta charset="UTF-8">
●   <meta name="viewport" content="width=device-width, initial-scale=1.0">
●   <title>Scan2Order - Menu</title>
● </head>
● <body>
●   <header>
●     <h1>Welcome to Scan2Order</h1>
●   </header>
●   <section>
●     <h2>Our Menu</h2>
●     <ul>
●       <li>Margherita Pizza - $10</li>
●       <li>Veggie Burger - $8</li>
●       <li>French Fries - $5</li>
●     </ul>
●   </section>
●   <footer>
●     <p>Thank you for ordering with Scan2Order!</p>
  </footer>
● </body>
  </html>
```

CSS3

2.3.1 Overview

CSS3 (Cascading Style Sheets version 3) is used for styling HTML elements, making web pages visually attractive and responsive to different screen sizes.

2.3.2 Key Features

- **Flexbox and Grid:** For complex layouts.
- **Animations and Transitions:** Smooth visual effects.
- **Media Queries:** For mobile responsiveness.
- **Custom Fonts and Gradients:** For better UI.

CSS3

2.3.1 Theory

CSS3 is responsible for the visual presentation of HTML elements.
It enhances user experience by enabling responsive designs, smooth animations, flexible layouts, and attractive styling.

In **Scan2Order**, CSS3 ensures that the menu and cart pages look clean, modern, and are mobile-responsive.

2.3.2 Features of CSS3

- **Flexbox and Grid Layouts:** For advanced responsive designs.
- **Media Queries:** Adapt designs to different screen sizes.
- **Animations and Transitions:** For smoother user interactions.
- **Box Model:** To manage margins, paddings, and borders efficiently.

Selector

Selectors are used to select the HTML elements that we want to style.

Example:

css

```
h1 {
  color: blue;
  font-size: 28px;
}
```

✅ Styles all `<h1>` headings to appear blue with larger font size.

---

2. Background Property

Used to set background colors or images for elements.

Example:

css

```
body {
  background-color: #f4f4f4;
}
```

✅ Gives a soft grey background to the whole page.

---

3. Font Styling

Controls the appearance of text.

Example:

css

```
p {
  font-family: Arial, sans-serif;
  font-size: 16px;
  color: #333;
}
```

✅ Makes paragraph text clean and professional.

---

4. Box Model Properties

- **Margin:** Space outside the border.
- **Padding:** Space between content and border.
- **Border:** The outer line around elements.

Example:

css

```
div {
  margin: 20px;
  padding: 15px;
  border: 2px solid #000;
}
```

✅ Provides neat spacing and separation between sections.

---

5. Flexbox Layout

Flexbox makes arranging elements much easier and responsive.

Example:

css

```
.container {
  display: flex;
  justify-content: space-around;
  align-items: center;
}
```

✅ Aligns items horizontally and spaces them evenly across the page.

---

6. Grid Layout

Grid helps design structured, two-dimensional layouts easily.

Example:

css

```css
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto;
  gap: 10px;
}
```

✅ Creates a 3-column layout with even spacing — great for menus!

---

7. Media Queries (Responsive Design)

Used to create mobile-friendly web designs.

Example:

css

```css
@media (max-width: 600px) {
  h1 {
    font-size: 20px;
  }
}
```

✅ Automatically adjusts heading size for smaller devices.

---

8. Hover Effects

Enhance interactivity by changing styles when a user hovers over an element.

Example:

css

```css
button:hover {
  background-color: #009688;
  color: white;
}
```

✅ Makes buttons look attractive when users interact.

---

9. Transitions

Create smooth changes when properties are changed.

Example:

css

```
button {
  transition: background-color 0.3s ease;
}
```

1.1JavaScript Introduction

JavaScript is a scripting language used to create interactive web pages. It allows real-time manipulation of content, form validations, dynamic cart operations, and communication with servers using AJAX.

In Scan2Order, JavaScript handles dynamic cart management, item selection, total calculation, and order placement.

1.2 Key Concepts in JavaScript

1.2.1 Variables

Variables store data values.

In modern JavaScript, we use `let`, `const`, and `var` to declare variables.

Example:

javascript

```javascript
let itemName = "Paneer Tikka";
const price = 150;
```

✅ `let` **allows re-assignment;** `const` **makes the value constant.**

1.2.2 Data Types

Common JavaScript data types:

- **String: Text (e.g., "Welcome")**
- **Number: Numeric values (e.g., 100)**
- **Boolean: true or false**
- **Array: Collection of items**
- **Object: Key-value pairs**

Example:

javascript

```javascript
let menuItem = {
  name: "Pizza",
  price: 250,
  available: true
};
```

✅ **Objects are very important in managing food items.**

1.2.3 Functions

Functions are reusable blocks of code.

Example:

javascript

```javascript
function add(a, b) {
  return a + b;
}

let total = add(100, 50);  // 150
```

✅ **Functions help modularize code, like adding to cart or calculating totals.**

1.2.4 Conditional Statements

Used to perform different actions based on different conditions.

Example:

javascript

```javascript
let available = true;

if (available) {
  console.log("Item is available!");
} else {
  console.log("Out of stock!");
}
```

✅ Used in Scan2Order to check item availability dynamically.

1.2.5 Loops

Loops execute a block of code repeatedly.

Example:

javascript

```
let cart = ["Pizza", "Burger", "Pasta"];

for (let i = 0; i < cart.length; i++) {
  console.log(cart[i]);
}
```

✅ Loops are used to display all items in the cart dynamically.

1.2.6 Arrays

Arrays hold multiple values in a single variable.

Example:

javascript

```
let categories = ["Starters", "Main Course", "Desserts"];
console.log(categories[1]); // Output: Main Course
```

✅ Menus are often organized using arrays.

1.2.7 Events

Events are actions performed by users (clicks, mouseover, etc.) to which JavaScript can respond.

Example:

javascript

```
document.getElementById("orderBtn").addEventListener("click",
function() {
  alert("Order Placed Successfully!");
});
```

✅ In Scan2Order, clicking "Place Order" button triggers event handlers.

1.2.8 DOM Manipulation

DOM (Document Object Model) represents the page structure in a tree format.

JavaScript can change, add, or delete HTML elements dynamically.

Example:

javascript

```
document.getElementById("menuTitle").innerText = "Today's
Special Menu";
```

✅ DOM manipulation allows live updates without reloading the page.

1.2.9 AJAX (Asynchronous JavaScript and XML)

AJAX is used to communicate with the server without reloading the page.

Example (Fetch API):

javascript

```
fetch('/placeOrder', {
  method: 'POST',
  body: JSON.stringify(cart),
  headers: { 'Content-Type': 'application/json' }
})
.then(response => response.json())
.then(data => console.log("Order Confirmed", data));
```

✅ AJAX is important for real-time order placement in Scan2Order.

2.1 Introduction

Oracle Application Express (APEX) is a powerful low-code development platform for building scalable, secure, and enterprise-grade web applications.

Although APEX is famous for Large Scale Bussiness Logic But in Scan2Order we utilized the true strength of APEX by writing custom SQL, PL/SQL, dynamic server-side logic, and strong database interaction — beyond basic no-code.

This chapter explains how Oracle APEX was used to design and develop key functionalities in Scan2Order using real coding techniques.

2.2 Core APEX Features Used in Scan2Order

- **Custom SQL Reports**
- **Dynamic PL/SQL Processes**
- **Validations using PL/SQL**
- **Authentication and Authorization Management**
- **Dynamic Actions (without No-Code Wizards)**
- **Custom Procedures and Functions**
- **API Integrations**
- **Secure Session Management**

2.3 Backend Coding in APEX

Oracle APEX supports writing backend logic directly using PL/SQL blocks, which allows more complex and powerful functionality than simple declarative options.

User Authentication (Custom Login)

Purpose: Verify admin login using PL/SQL block (not default APEX login).

Code:

plsql

```
DECLARE
    v_count NUMBER;
BEGIN
    SELECT COUNT(*)
    INTO v_count
    FROM users
    WHERE username = :P101_USERNAME
      AND password = :P101_PASSWORD
      AND role = 'ADMIN';

    IF v_count = 1 THEN
        APEX_UTIL.SET_SESSION_STATE('USER_ROLE', 'ADMIN');
        APEX_AUTHENTICATION.POST_LOGIN (p_uname => :P101_USERNAME);
    ELSE
        raise_application_error(-20001, 'Invalid Username or Password');
    END IF;
END;
```

✅ **Attached To: Login Page Process (After Submit).**

2.2 Insert New Order (Cart to Orders Table)

Purpose: Insert new order along with multiple items (bulk insert using G_F01, G_F02 arrays).

Code:

plsql

```
DECLARE
    v_order_id orders.order_id%TYPE;
BEGIN
    INSERT INTO orders (table_id, order_time, total_amount, status)
    VALUES (:P1_TABLE_ID, SYSDATE, :P1_TOTAL_AMOUNT, 'Placed')
    RETURNING order_id INTO v_order_id;

    FOR i IN 1..APEX_APPLICATION.G_F01.COUNT LOOP
        INSERT INTO order_details (order_id, item_id, quantity)
        VALUES (v_order_id, TO_NUMBER(APEX_APPLICATION.G_F01(i)),
TO_NUMBER(APEX_APPLICATION.G_F02(i)));
    END LOOP;

    COMMIT;
END;
```

✅ **Attached To: Order Placement Button (Dynamic Action → PL/SQL Process).**

2.3 Update Order Status (Admin Dashboard)

Purpose: Change order status dynamically (Placed → Preparing → Ready → Completed).

Code:

plsql

```
BEGIN
    UPDATE orders
    SET status = :P3_NEW_STATUS
    WHERE order_id = :P3_ORDER_ID;

    COMMIT;
END;
```

✅ **Attached To: Status Dropdown Change (Dynamic Action → PL/SQL).**

2.4 Validation: Prevent Duplicate Menu Item

Purpose: Ensure no duplicate food item names when inserting into Menu.

Code:

plsql

```
DECLARE
    v_exist NUMBER;
BEGIN
    SELECT COUNT(*)
    INTO v_exist
    FROM menu_items
    WHERE LOWER(item_name) = LOWER(:P2_ITEM_NAME);

    IF v_exist > 0 THEN
```

```
        raise_application_error(-20002, 'Menu Item already exists!');
    END IF;
END;
```

✅ **Attached To: Validation Section of Menu Add/Edit Form.**

2.5 Generate Dynamic QR Code URLs (Table Wise)

Purpose: Create a dynamic URL per table for QR Code generation.

SQL Query:

sql

```
SELECT table_id,
       'https://scan2order.com/menu?table_id=' || table_id AS qr_code_url
FROM tables;
```

✅ **Use In: Report Region (QR Code Printout Page).**

2.6 Dynamic Action: Refresh Orders Region on Status Update

Purpose: Without full page submit, refresh live orders report after status update.

Action Sequence:

- **Event: Change (on dropdown P3_NEW_STATUS)**
- **True Action: Execute PL/SQL Code (Update)**
- **Then: Refresh Orders Region**

✅ **Dynamic and lightweight refresh!**

2.7 Table Structure Used in Backend (Summary)

Table Name	Columns	Purpose
`menu_items`	item_id, item_name, price, category, available	Store food menu
`orders`	order_id, table_id, order_time, total_amount, status	Main order header
`order_details`	detail_id, order_id, item_id, quantity	Individual items inside orders
`tables`	table_id, qr_code_url	Restaurant tables linked with QR codes
`users`	user_id, username, password, role	Admin users authentication

---

2.8 Example: Send Notification on New Order (Optional Future)

PL/SQL Sample Process:

plsql

```
BEGIN
    APEX_UTIL.SEND_EMAIL(
        p_to => 'kitchen@restaurant.com',
        p_from => 'scan2order@restaurant.com',
        p_subj => 'New Order Received!',
        p_body => 'A new order has been placed. Order ID: ' ||
:P1_ORDER_ID
    );
END;
```

✅ **(Optional future feature — automatic new order notification to kitchen.)**

2.9 Example: Create Order Report (Admin View)

SQL Query For Interactive Report:

sql

```
SELECT o.order_id,
       t.table_id,
       o.order_time,
       o.status,
       o.total_amount
FROM orders o
JOIN tables t ON o.table_id = t.table_id
ORDER BY o.order_time DESC;
```

✅ **Admin can easily view all orders with real-time sorting, filtering.**

2.10 Protect Sensitive Pages (Authorization Scheme)

Protect Admin Pages by Role:

Authorization Scheme Condition:

plsql

```
RETURN APEX_UTIL.GET_SESSION_STATE('USER_ROLE') = 'ADMIN';
```

✅ **Attached To: Admin Menu Management, Admin Orders Pages.**

✅ **If session role is not Admin, access is blocked.**

2.11 Example: Logout Button (Clear Session)

Dynamic Action:

- **Event: Click (Logout Button)**
- **True Action: Execute PL/SQL**

plsql

```
BEGIN

    APEX_UTIL.CLEAR_SESSION;

    APEX_AUTHENTICATION.LOGOUT(p_this_flow => 'f?p=APP_ID:1',
p_next_app_page_sess => 1);

END;
```

✅ **Session securely destroyed and redirected to Login page.**

User Authentication (Custom Login)

Purpose: Verify admin login using PL/SQL block (not default APEX login).

Code:

plsql

```plsql
DECLARE
    v_count NUMBER;
BEGIN
    SELECT COUNT(*)
    INTO v_count
    FROM users
    WHERE username = :P101_USERNAME
      AND password = :P101_PASSWORD
      AND role = 'ADMIN';

    IF v_count = 1 THEN
        APEX_UTIL.SET_SESSION_STATE('USER_ROLE', 'ADMIN');
        APEX_AUTHENTICATION.POST_LOGIN (p_uname => :P101_USERNAME);
    ELSE
        raise_application_error(-20001, 'Invalid Username or Password');
    END IF;
END;
```

✅ **Attached To: Login Page Process (After Submit).**

2.2 Insert New Order (Cart to Orders Table)

Purpose: Insert new order along with multiple items (bulk insert using G_F01, G_F02 arrays).

Code:

plsql

```
DECLARE
    v_order_id orders.order_id%TYPE;
BEGIN
    INSERT INTO orders (table_id, order_time, total_amount, status)
    VALUES (:P1_TABLE_ID, SYSDATE, :P1_TOTAL_AMOUNT, 'Placed')
    RETURNING order_id INTO v_order_id;

    FOR i IN 1..APEX_APPLICATION.G_F01.COUNT LOOP
        INSERT INTO order_details (order_id, item_id, quantity)
        VALUES (v_order_id, TO_NUMBER(APEX_APPLICATION.G_F01(i)),
TO_NUMBER(APEX_APPLICATION.G_F02(i)));
    END LOOP;

    COMMIT;
END;
```

✅ **Attached To: Order Placement Button (Dynamic Action → PL/SQL Process).**

2.3 Update Order Status (Admin Dashboard)

Purpose: Change order status dynamically (Placed → Preparing → Ready → Completed).

Code:

plsql

```
BEGIN
    UPDATE orders
    SET status = :P3_NEW_STATUS
    WHERE order_id = :P3_ORDER_ID;

    COMMIT;
END;
```

✅ **Attached To: Status Dropdown Change (Dynamic Action → PL/SQL).**

2.4 Validation: Prevent Duplicate Menu Item

Purpose: Ensure no duplicate food item names when inserting into Menu.

Code:

plsql

```
DECLARE
    v_exist NUMBER;
BEGIN
```

```
    SELECT COUNT(*)
    INTO v_exist
    FROM menu_items
    WHERE LOWER(item_name) = LOWER(:P2_ITEM_NAME);

    IF v_exist > 0 THEN
        raise_application_error(-20002, 'Menu Item already exists!');
    END IF;
END;
```

✅ **Attached To: Validation Section of Menu Add/Edit Form.**

2.5 Generate Dynamic QR Code URLs (Table Wise)

Purpose: Create a dynamic URL per table for QR Code generation.

SQL Query:

sql

```
SELECT table_id,
       'https://scan2order.com/menu?table_id=' || table_id AS qr_code_url
FROM tables;
```

✅ **Use In: Report Region (QR Code Printout Page).**

2.6 Dynamic Action: Refresh Orders Region on Status Update

Purpose: Without full page submit, refresh live orders report after status update.

Action Sequence:

- **Event: Change (on dropdown P3_NEW_STATUS)**
- **True Action: Execute PL/SQL Code (Update)**
- **Then: Refresh Orders Region**

✅ **Dynamic and lightweight refresh!**

2.7 Table Structure Used in Backend (Summary)

Table Name	Columns	Purpose
`menu_items`	item_id, item_name, price, category, available	Store food menu
`orders`	order_id, table_id, order_time, total_amount, status	Main order header
`order_details`	detail_id, order_id, item_id, quantity	Individual items inside orders
`tables`	table_id, qr_code_url	Restaurant tables linked with QR codes
`users`	user_id, username, password, role	Admin users authentication

---

2.8 Example: Send Notification on New Order (Optional Future)

PL/SQL Sample Process: plsql

```plsql
BEGIN

    APEX_UTIL.SEND_EMAIL(

        p_to => 'kitchen@restaurant.com',

        p_from => 'scan2order@restaurant.com',

        p_subj => 'New Order Received!',

        p_body => 'A new order has been placed. Order ID: ' ||
:P1_ORDER_ID

    );

END;
```

✅ **(Optional future feature — automatic new order notification to kitchen.)**

2.9 Example: Create Order Report (Admin View)

SQL Query For Interactive Report:

sql

```sql
SELECT o.order_id,

       t.table_id,

       o.order_time,

       o.status,

       o.total_amount

FROM orders o
JOIN tables t ON o.table_id = t.table_id
ORDER BY o.order_time DESC;
```

✅ **Admin can easily view all orders with real-time sorting, filtering**

2.10 Protect Sensitive Pages (Authorization Scheme)

Protect Admin Pages by Role:

Authorization Scheme Condition:

plsql

```
RETURN APEX_UTIL.GET_SESSION_STATE('USER_ROLE') = 'ADMIN';
```

✅ **Attached To: Admin Menu Management, Admin Orders Pages.**

✅ **If session role is not Admin, access is blocked.**

2.11 Example: Logout Button (Clear Session)

Dynamic Action:

- **Event: Click (Logout Button)**
- **True Action: Execute PL/SQL**

plsql

```
BEGIN

    APEX_UTIL.CLEAR_SESSION;
    APEX_AUTHENTICATION.LOGOUT(p_this_flow => 'f?p=APP_ID:1',
p_next_app_page_sess => 1);

END;
```

✅ **Session securely destroyed and redirected to Login page.**

CHAPTER 3:PROJECT WORK

3.1 Introduction

The project Scan2Order is built
to solve ordering problems in restaurants.
It replaces traditional menus with QR-based ordering.

Customers scan the code and order directly.
Admins can manage food items, tables, and orders.

This chapter explains the design, flow,
modules, technologies, and working of the system.
Screenshots are inserted where necessary for clarity.

3.2 System Overview

Scan2Order consists of two main modules:

- **Customer Module**
- **Admin Module**

Each module is built carefully to make
ordering smooth, fast, and contactless.

The system has two main parts:

- **Front-end Web Pages (HTML, CSS, JS)**
- **Back-end Server (Oracle APEX, PL/SQL, SQL Database)**

3.3 Workflow of the System

Step 1: Customer scans QR Code at table.

Step 2: Customer views the dynamic menu.

Step 3: Customer adds food items to cart.

Step 4: Customer places the order.

Step 5: Order is sent to kitchen/admin panel.

Step 6: Admin updates the order status.
Step 7: Food is prepared and served.

3.4 Modules of Scan2Order

3.4.1 Customer Module

Customers interact through their mobile browsers.
No app download is required.

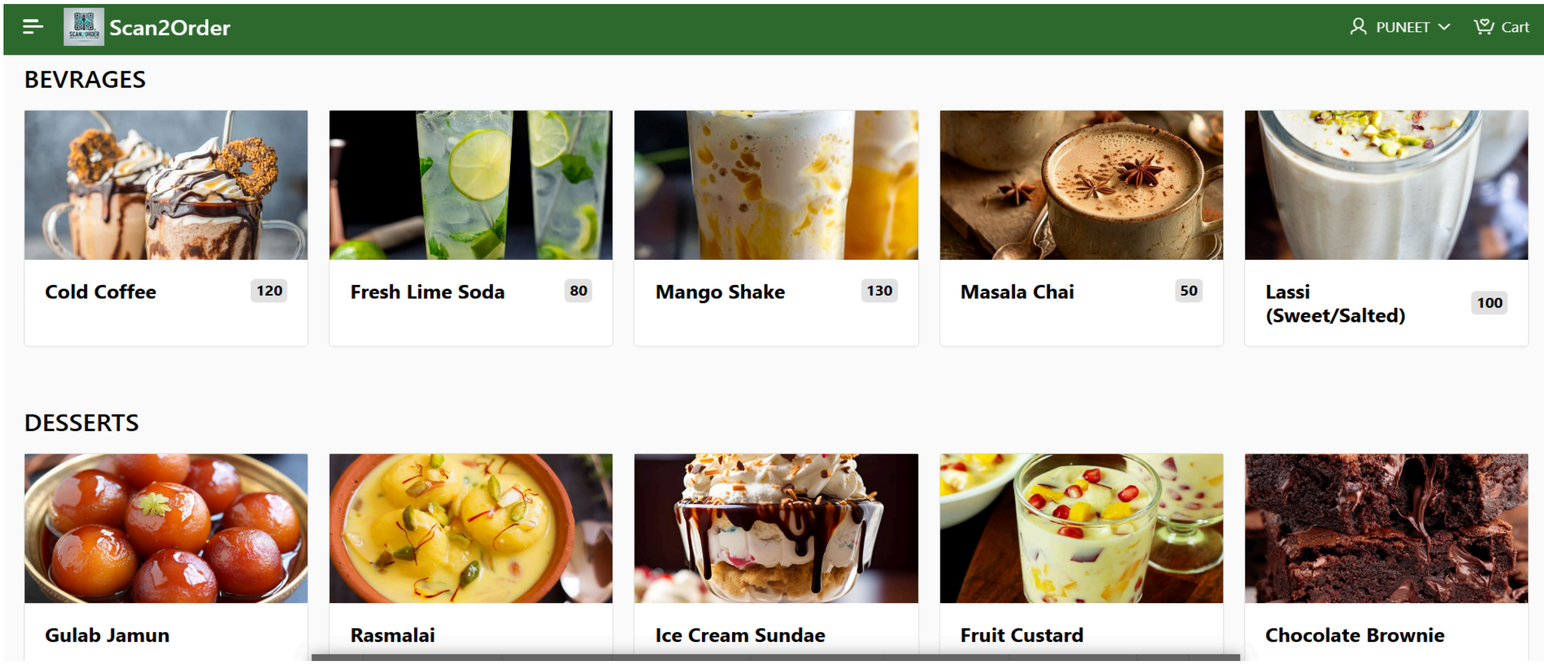
Features:

- **QR Code Scan:**
  Opens menu page directly.
- **Dynamic Menu View:**
  Items are listed under categories.
- **Cart Management:**
  Add, remove, update quantity easily.

- **Order Placement:**
  Submit order with one click.
  Order Confirmation:
  A thank-you message with Order ID is shown.

1..Scan QR Code To Display Dynamic Menu

Step 2: Display Menu To order



BEVRAGES

Cold Coffee	120
Fresh Lime Soda	80
Mango Shake	130
Masala Chai	50
Lassi (Sweet/Salted)	100

DESSERTS

Gulab Jamun

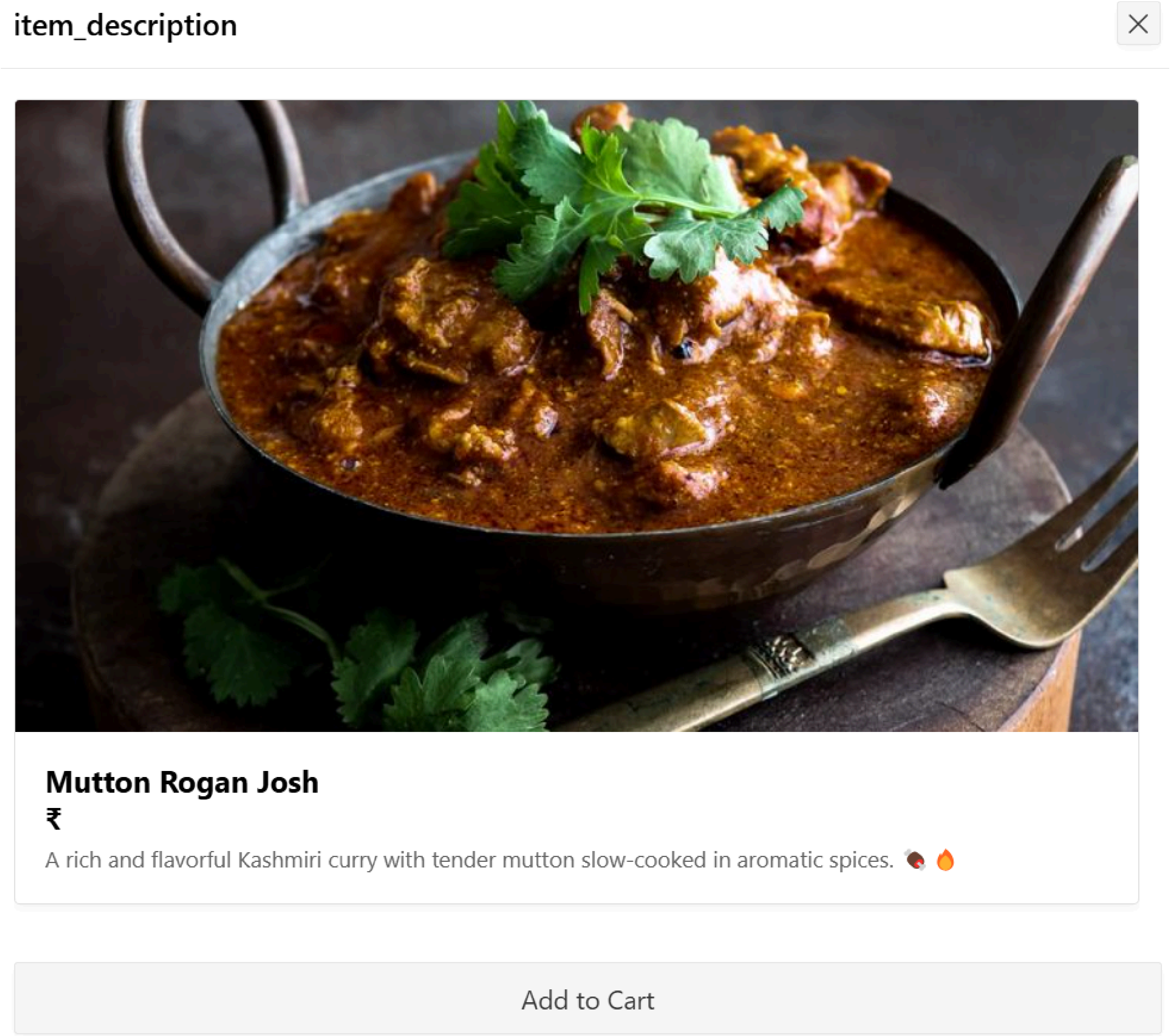
Rasmalai

Ice Cream Sundae

Fruit Custard

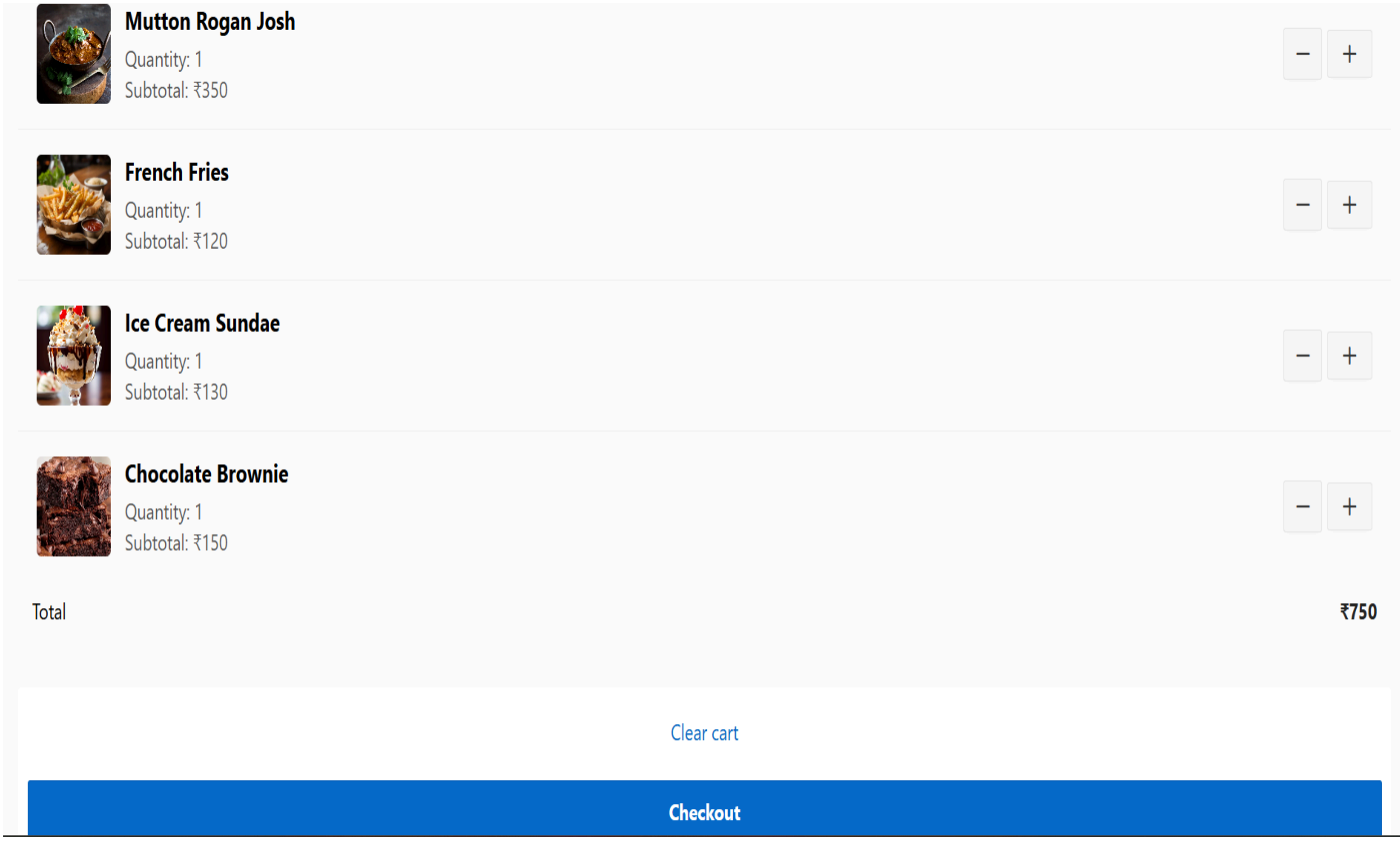
Chocolate Brownie

Item Description



Mutton Rogan Josh
₹
A rich and flavorful Kashmiri curry with tender mutton slow-cooked in aromatic spices. 🍖 🔥

Add To Cart



Mutton Rogan Josh
Quantity: 1
Subtotal: ₹350

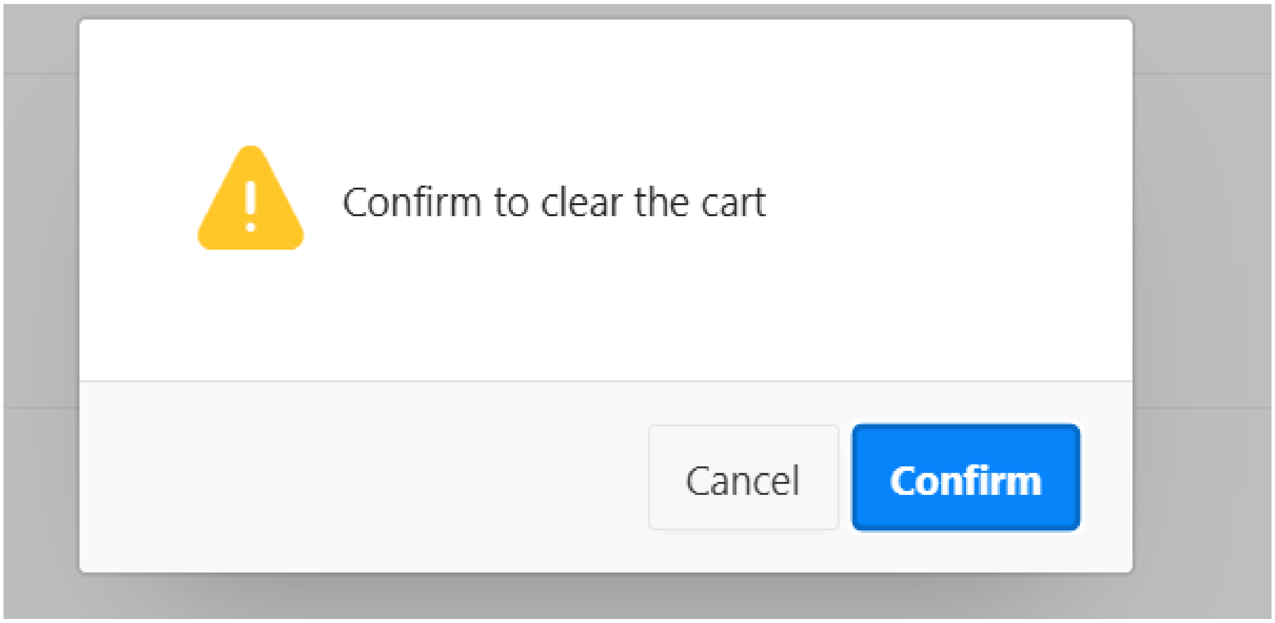
French Fries
Quantity: 1
Subtotal: ₹120

Ice Cream Sundae
Quantity: 1
Subtotal: ₹130

Chocolate Brownie
Quantity: 1
Subtotal: ₹150

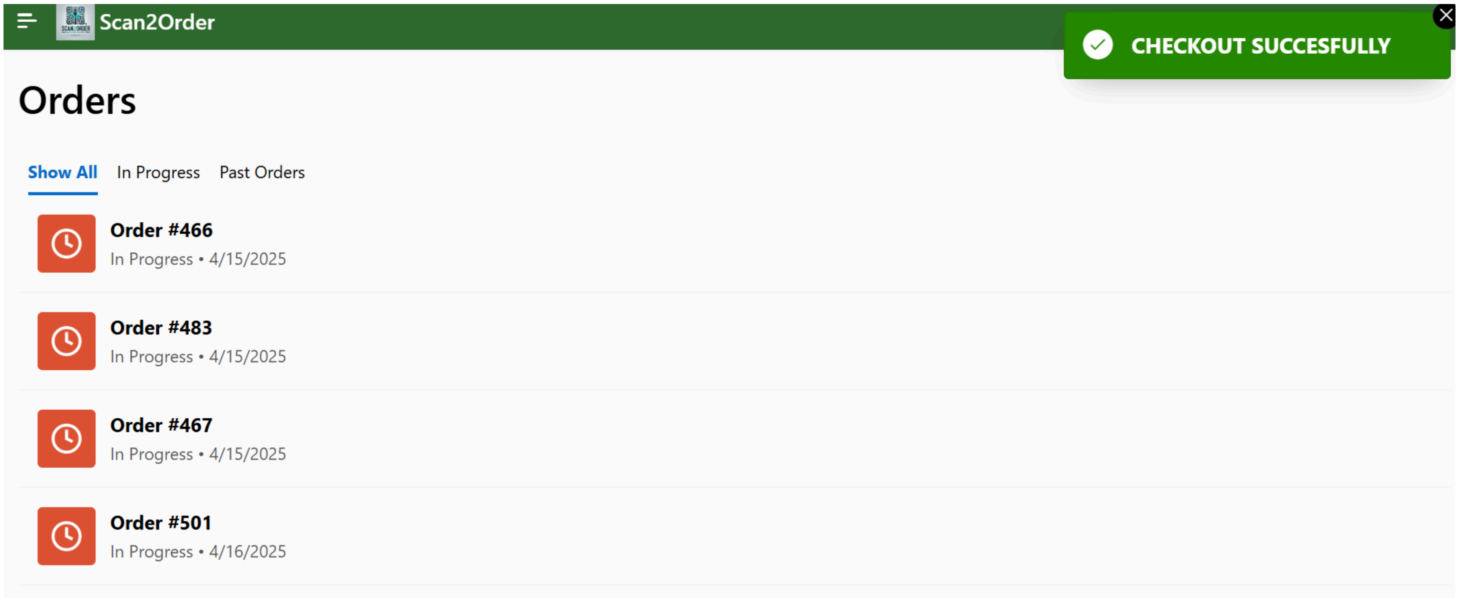
Total **₹750**

Clear Cart



Confirm to clear the cart

Order Confirmation



Orders

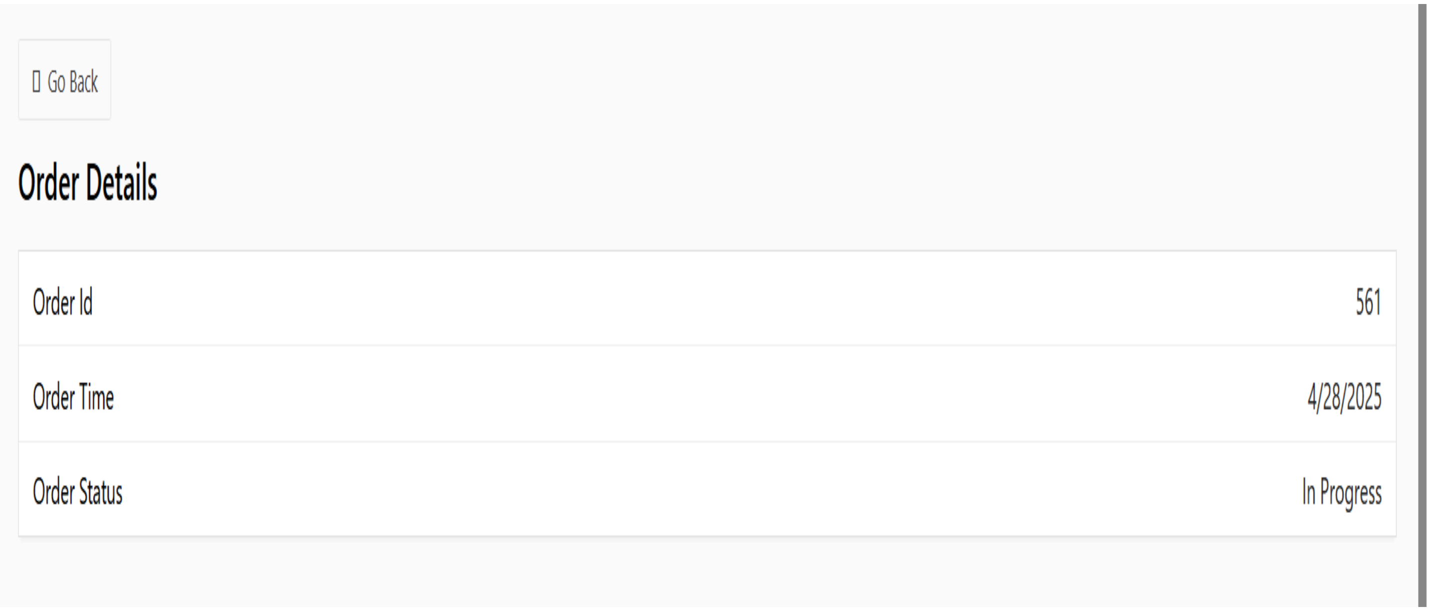
Order #466
In Progress • 4/15/2025

Order #483
In Progress • 4/15/2025

Order #467
In Progress • 4/15/2025

Order #501
In Progress • 4/16/2025

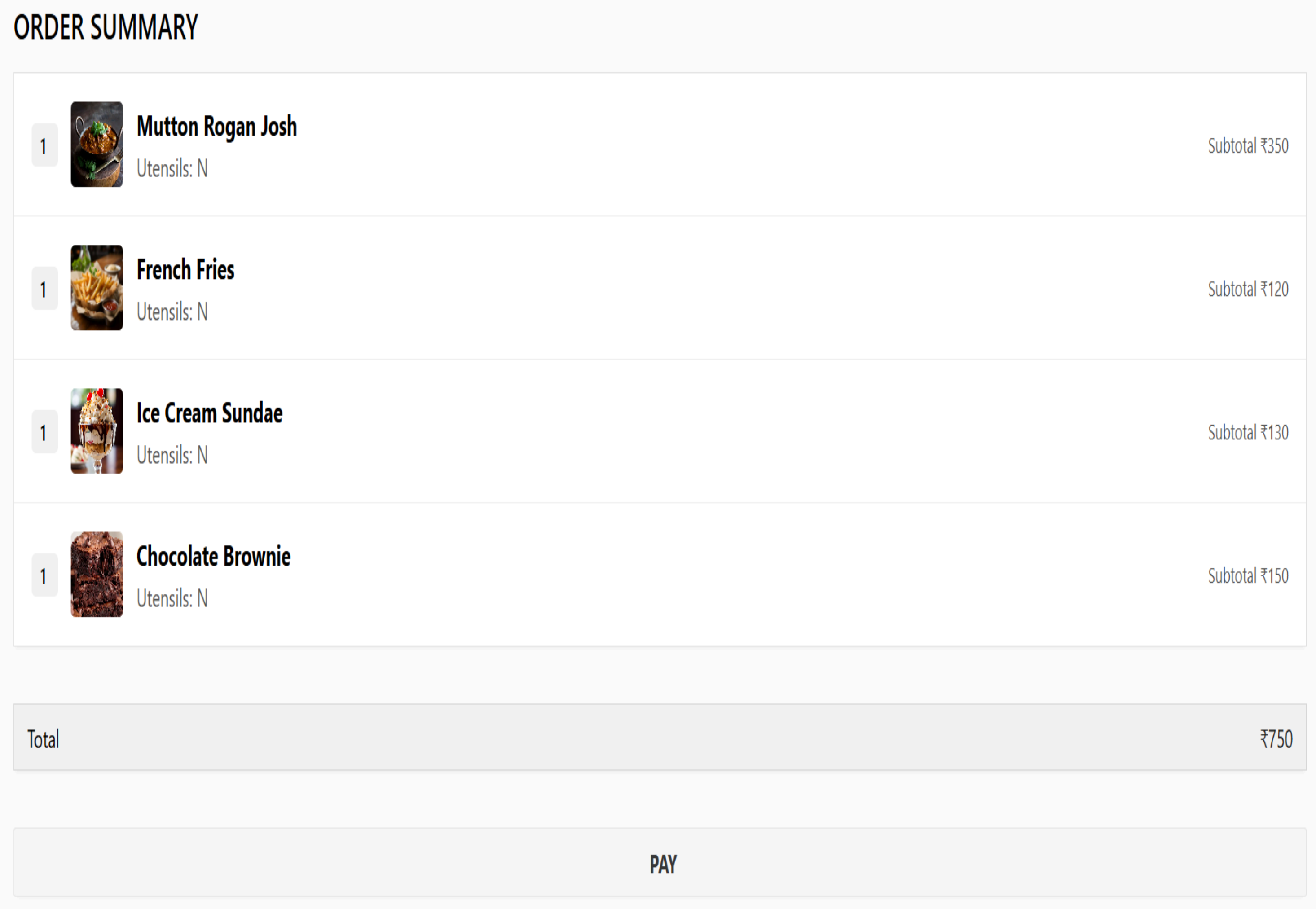
Order Detail



Order Details

Order Id	561
Order Time	4/28/2025
Order Status	In Progress

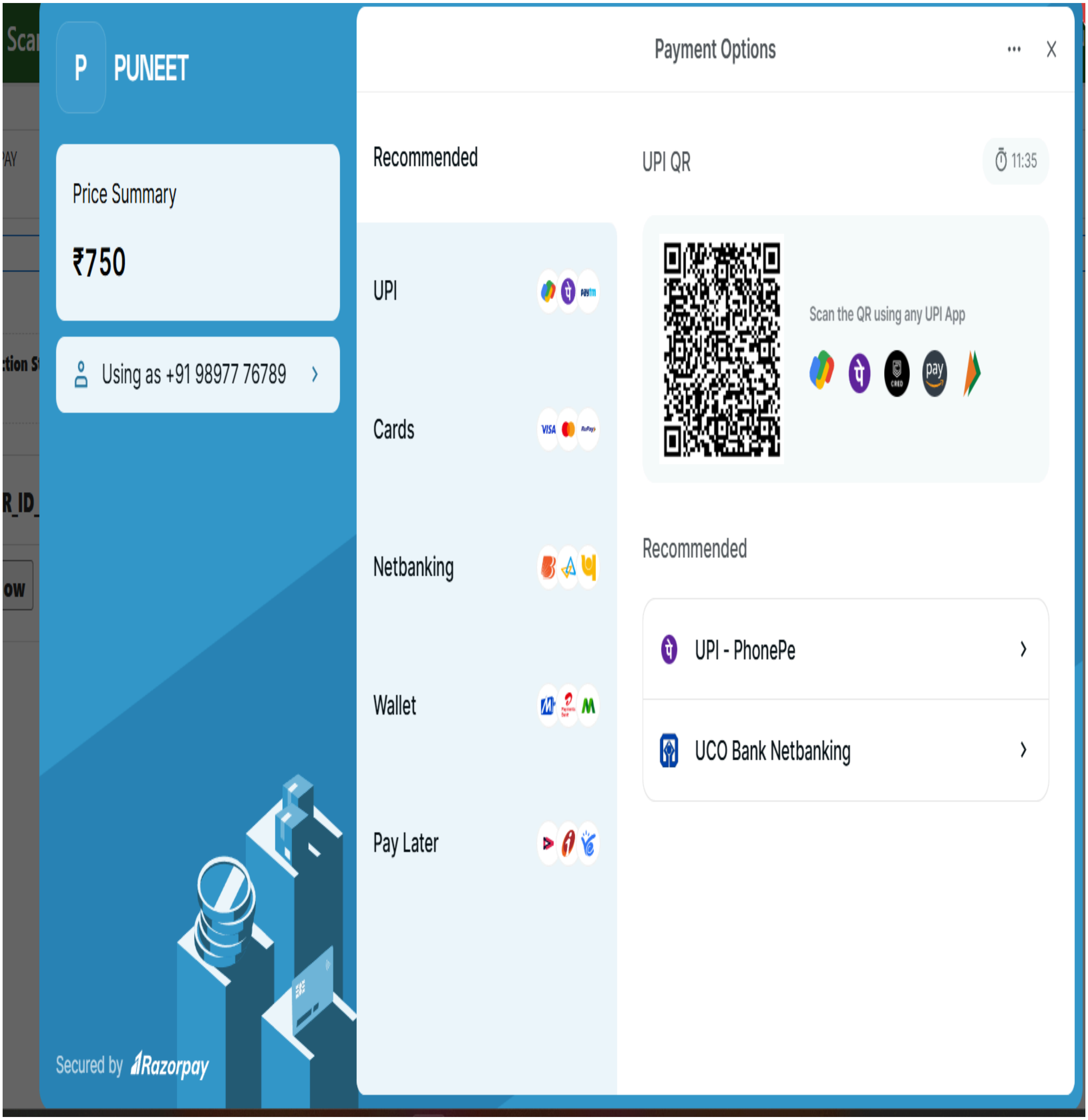
Order Summary



ORDER SUMMARY

1	**Mutton Rogan Josh** Utensils: N	Subtotal ₹350
1	**French Fries** Utensils: N	Subtotal ₹120
1	**Ice Cream Sundae** Utensils: N	Subtotal ₹130
1	**Chocolate Brownie** Utensils: N	Subtotal ₹150

Total	₹750



P PUNEET

Price Summary

₹750

Using as +91 98977 76789

Payment Options

Recommended

UPI QR

11:35

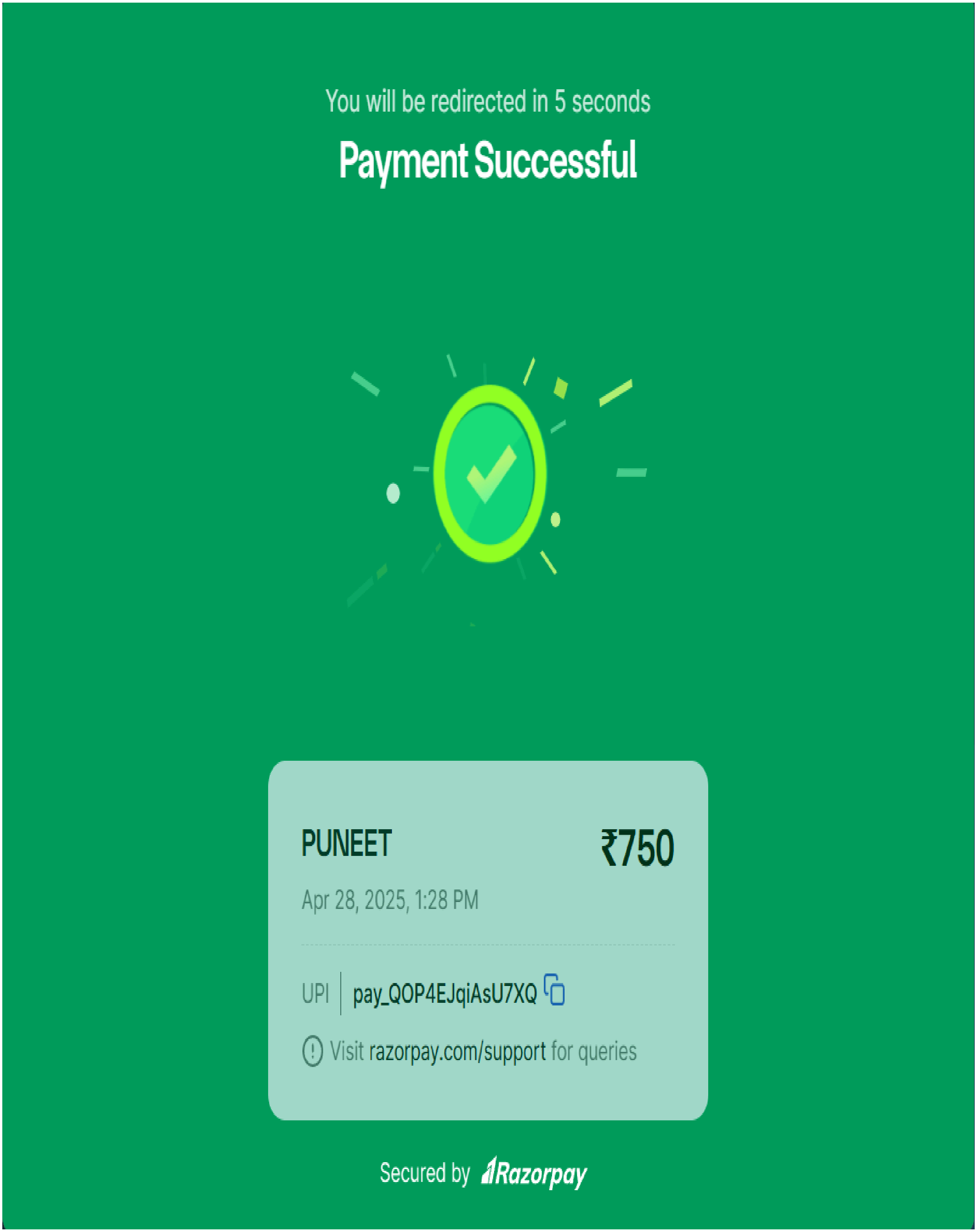
Scan the QR using any UPI App

Recommended

UPI - PhonePe

UCO Bank Netbanking

Secured by Razorpay



You will be redirected in 5 seconds

Payment Successful

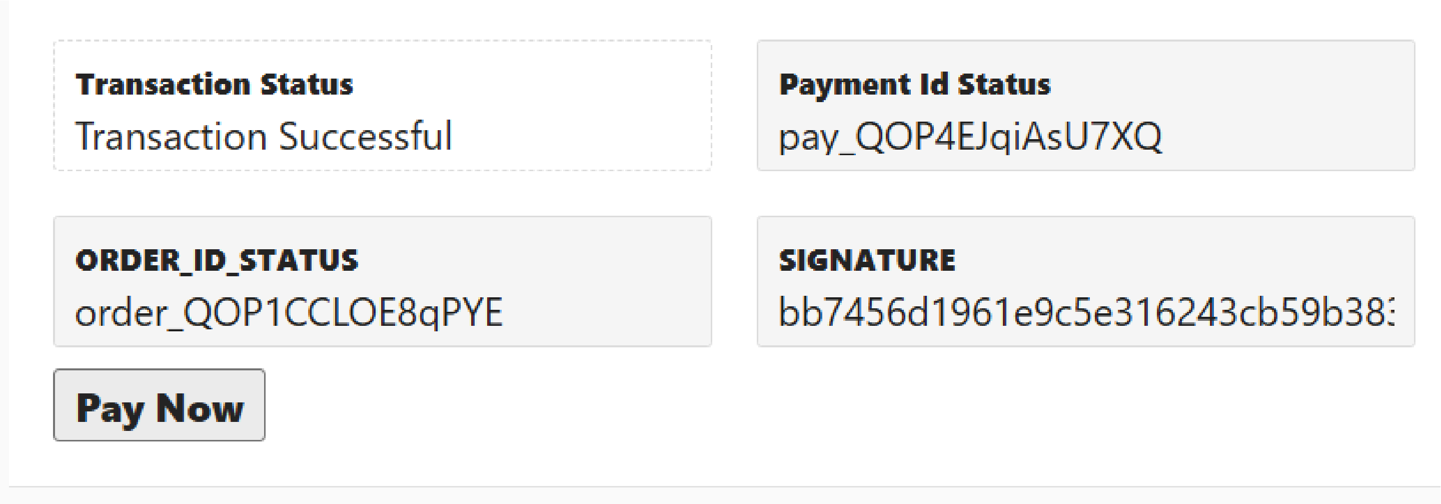
PUNEET ₹750

Apr 28, 2025, 1:28 PM

UPI | pay_QOP4EJqiAsU7XQ

ⓘ Visit razorpay.com/support for queries

Secured by Razorpay



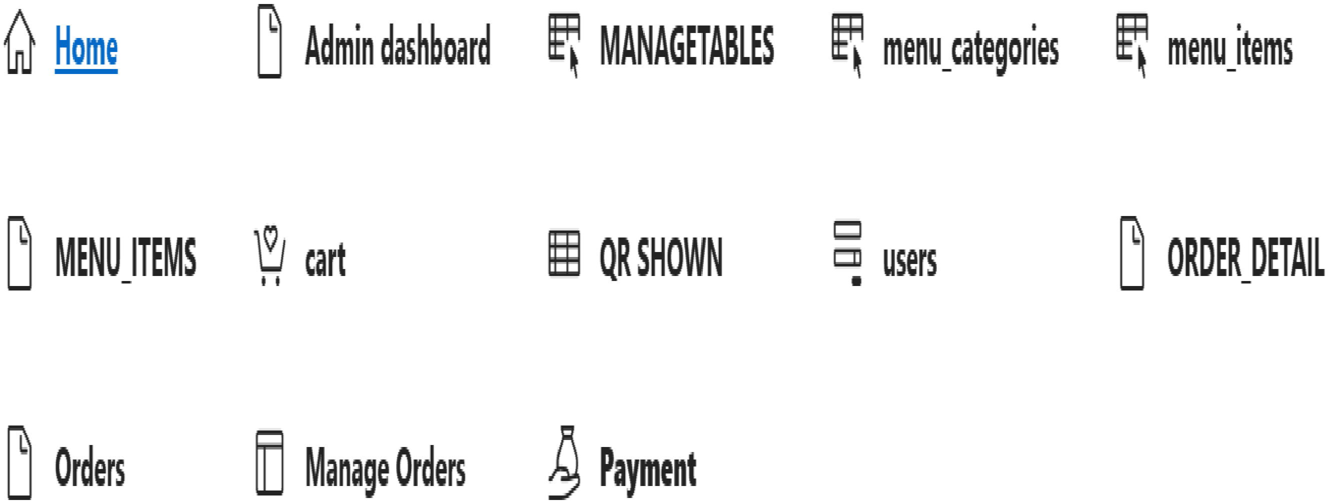
Transaction Status
Transaction Successful

Payment Id Status
pay_QOP4EJqiAsU7XQ

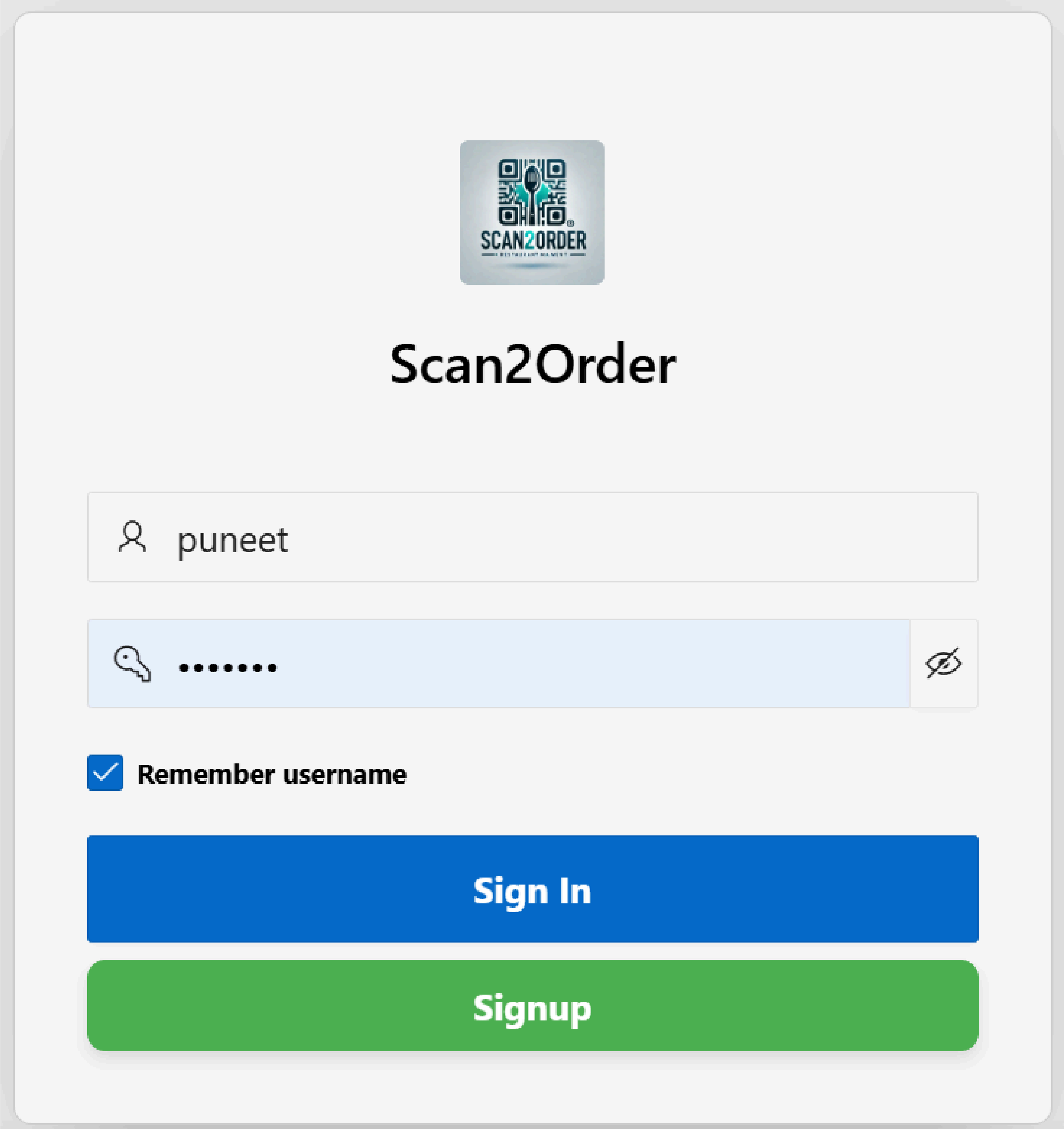
ORDER_ID_STATUS
order_QOP1CCLOE8qPYE

SIGNATURE
bb7456d1961e9c5e316243cb59b38

Admin Panel Functionality



Login page With SignUp



Scan2Order

puneet

•••••••

☑ Remember username

CHAPTER 4 (Result)

The Scan2Order - QR Based Restaurant Ordering System was successfully developed and deployed to address the key challenges faced in traditional restaurant order management.

The system fulfills core requirements such as:

- **Customer QR code scanning,**
- **Menu browsing,**
- **Cart management,**
- **Order placement,**
- **Admin login,**
- **Menu and table management,**
- **Order tracking.**

The web application delivers a clean, fast, and user-friendly interface optimized for both customers and restaurant staff.

Upon testing, the system demonstrated stable and reliable performance.
Customers could easily scan the QR codes, browse dynamic menus, add items to their cart, and place orders seamlessly from their mobile devices without needing to install any app.

The admin panel performed efficiently for restaurant staff, allowing them to:

- **Manage food menu entries,**
- **Track live orders,**
- **Update order statuses (e.g., Preparing, Ready, Served),**
- **Manage restaurant tables and their linked QR codes,**
- **Securely access backend operations using role-based authentication.**

The login system ensured secure access control, maintaining data privacy, separation between users and administrators, and session security.

4.2 Discussion

The implementation of **Scan2Order** significantly improved restaurant operations by:

- **Reducing manual workload** involved in taking and managing orders.

- **Minimizing errors** typically occurring in manual order recording.

- **Speeding up service times** by eliminating the need for waitstaff to take orders

  .

- **Enhancing hygiene and safety** by replacing physical menus with digital menus.

- **Increasing customer satisfaction** through faster service and contactless interactions.

- **Enabling real-time order management** by kitchen and billing staff via the admin dashboard.

The use of **dynamic QR codes** and **web-based interfaces** meant that updates to menus or table setups could be made instantly, without downtime or costly reprinting of menus.

The **interactive reports**, **dynamic actions**, and **PL/SQL processes** developed in Oracle APEX ensured robust backend processing and real-time live updating of the system without full-page reloads.

CHAPTER 5

FUTURE SCOPE AND ENHANCEMENT

5.2.2 Real-Time Kitchen Notifications (WebSocket Integration)

Currently, orders are refreshed manually.
In the future, **WebSocket** can be used to create real-time bi-directional communication between the customer's device, kitchen, and admin dashboard.

Benefits:

- Kitchen staff receive new orders instantly.
- Customers get live updates (e.g., "Order Preparing", "Order Ready").

🛠️ **Technologies Used:** Oracle REST Data Services + WebSocket Server.

5.2.3 Customer Feedback and Rating System

After completing the meal, customers can submit feedback directly from their mobile.

- Star ratings (1 to 5 stars)
- Short review comments
- Feedback on food, service, and overall experience

Collected data can help the restaurant improve services.

🛠️ **Technologies Used:** APEX Forms, REST APIs, Sentiment Analysis (optional future AI).

5.2.4 Loyalty Program and Coupons

To encourage repeat customers, a loyalty program can be introduced.

- Customers earn points on every order.
- Redeem points for discounts or free items.
- Special promotions and coupons for frequent visitors.

🛠 **Technologies Used:** SQL Triggers for Points Update, Coupon Engine APIs.

5.2.5 Table Reservation System

Customers could pre-book tables from the web app itself.

Features:

- Choose table type (e.g., AC/Non-AC, Window Side)
- Select date and time
- Automatic QR code generation for reserved tables

🛠 **Technologies Used:** Calendar Components, APEX Scheduler.

5.2.6 Multi-Branch Support (Chain Restaurants)

Currently, Scan2Order works for single locations.

In future, multi-branch support can be added:

- Multiple restaurants managed under one admin panel
- Branch-specific menus
- Centralized reporting and analytics

🛠 **Technologies Used:** Branch ID Mapping, Multi-Tenant Database Design.

5.2.7 AI-Powered Smart Recommendations

An AI model could recommend dishes to customers based on:

- Past orders
- Popularity trends
- Current weather (e.g., hot soups on rainy days)

🛠 **Technologies Used:** Machine Learning Models, Oracle Machine Learning.

5.2.8 Offline Mode (Progressive Web App - PWA)

Adding offline capabilities will allow customers to browse the menu and place orders even if internet connectivity drops temporarily.

- Data synchronization once connection restores
- LocalStorage caching for menus

🛠️ **Technologies Used:** Service Workers, IndexedDB.

5.2.9 E-Menu in Multiple Languages

Currently, the menu is only available in English.
In future, multilingual support can be added:

- Hindi
- Regional Languages
- English

Customers can choose language preference when they scan the QR.

🛠️ **Technologies Used:** APEX Dynamic Translations, JSON-based Language Files.

5.2.10 Business Analytics and Reports

Advanced dashboards can be added for restaurant owners:

- Daily/Monthly Sales Reports
- Most Sold Items
- Peak Business Hours
- Table Utilization Statistics

Helps in making better business decisions.

🛠️ **Technologies Used:** APEX Charts, BI Publisher Integration, SQL Analytical Functions.

CHAPTER 6

REFERENCES

Oracle Corporation, Oracle APEX Official Documentation, Available at: https://docs.oracle.com/en/database/oracle/application-express

Oracle Corporation, SQL Language Reference, Available at: https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf

Oracle Corporation, PL/SQL User's Guide and Reference, Available at: https://docs.oracle.com/en/database/oracle/oracle-database/19/plsql

Mozilla Developer Network (MDN), HTML5 Reference Guide, Available at: https://developer.mozilla.org/en-US/docs/Web/HTML

Mozilla Developer Network (MDN), CSS3 Reference Guide, Available at: https://developer.mozilla.org/en-US/docs/Web/CSS

Mozilla Developer Network (MDN), JavaScript Documentation, Available at: https://developer.mozilla.org/en-US/docs/Web/JavaScript

WebSocket API Documentation, WebSockets Explained, Available at: https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API

RESTful Web Services Guide, REST API Design, Available at: https://restfulapi.net

Razorpay, Payment Gateway Integration Guide, Available at: https://razorpay.com/docs/payments/payment-gateway

QR Code.com, QR Code Essentials, Available at: https://www.qrcode.com/en/

W3Schools, HTML Forms and Input Types, Available at: https://www.w3schools.com/html/html_forms.asp

W3Schools, CSS Flexbox Layout, Available at: https://www.w3schools.com/css/css3_flexbox.asp

W3Schools, JavaScript Events, Available at: https://www.w3schools.com/js/js_events.asp

Google Charts API, Dynamic QR Code Generation, Available at: https://developers.google.com/chart/infographics/docs/qr_codes

Stripe, Secure Payment Integration Guide, Available at: https://stripe.com/docs/payments

Oracle Blogs, Real-Time Communication with Oracle REST and WebSocket, Available at: https://blogs.oracle.com/apex/post/real-time-oracle-rest-and-websockets

APEX World, Building Dynamic Apps with APEX, Available at: https://apex.world

Research Paper, Smart Ordering System using QR Code Technology, International Journal of Innovative Research in Computer and Communication Engineering, Vol. 5, Issue 4, April 2017.

Research Paper, Digitizing Restaurant Services with Web Applications, International Journal of Computer Applications, Vol. 183, No. 3, 2021.

Medium Blog, How to Create a QR-Based Order Management System, Available at: https://medium.com/@techblogs/qr-based-order-system