

Design and Implementation of a Food Ordering Management System

K. Manjula^{1*}, Basavaraj S. Pol², K. R. Varalakshmi³

¹Department of Computer Science and Engineering, R. L. Jalappa Institute of Technology, Kodigehalli, India

Abstract: The Food Order Management System is a web-based application developed using PHP and MySQL to automate and simplify restaurant order management. The system provides a centralized platform for managing customer information, menu items, orders, payments, and order status. It is designed to eliminate the inefficiencies of traditional manual ordering methods by offering a secure, accurate, and user-friendly digital solution. The application enables customers to create accounts, browse the restaurant menu, add food items to a shopping cart, and place orders through an intuitive interface. Users can also view their order history and monitor the status of their orders in real time. On the administrative side, restaurant staff can efficiently manage menu items, process incoming orders, update order status, monitor payment information, and generate sales reports for business analysis. The system also supports storing customer preferences, special requests, and transaction records in a structured relational database. By integrating all restaurant operations into a single platform, the proposed system improves order accuracy, reduces processing time, minimizes human errors, and enhances communication between customers and restaurant staff. Furthermore, the centralized database facilitates efficient inventory monitoring, sales analysis, and decision-making, thereby improving the overall operational performance of the restaurant. The proposed Food Order Management System offers a scalable, reliable, and cost-effective solution for modern restaurant management and contributes to an enhanced customer dining experience.

Keywords: Food Ordering Management System.

1. Introduction

A. Introduction to Food Order Management System

The Food Order Management System is a web-based application developed to automate and optimize the restaurant ordering process using PHP and MySQL technologies. The system provides an integrated platform that enables restaurants, cafés, and other food service establishments to efficiently manage customer orders, menu items, payments, and order status through a centralized database.

The application allows customers to register, browse digital menus, select food items, place orders, and track their order status in real time. Simultaneously, restaurant administrators can manage menu information, monitor incoming orders, update order progress, process payments, and generate business reports through a dedicated administrative interface. By replacing conventional paper-based order management with an

automated digital solution, the system reduces manual errors, improves operational efficiency, and enhances customer satisfaction.

Furthermore, the system supports secure data management, efficient inventory monitoring, and comprehensive sales analysis, making it suitable for restaurants of different sizes. Its modular architecture also allows future enhancements and integration with emerging technologies such as online payment gateways, cloud services, and mobile applications.

B. Aim and Objectives

The primary aim of the proposed Food Order Management System is to develop an efficient, reliable, and user-friendly web application that automates restaurant order processing and management.

The major objectives of the system are: To provide customers with a convenient online platform for ordering food. To automate restaurant order processing and minimize manual intervention.

- To improve order accuracy and reduce processing.
- To maintain a centralized database for customers, menu items, orders, and payments.
- To enable restaurant administrators to efficiently manage menus and customer orders.
- To generate analytical reports for monitoring sales and business performance.
- To enhance customer satisfaction by providing real-time order tracking and faster service.
- To improve overall operational efficiency while reducing paperwork and administrative costs.

C. Reason for the Project

The rapid adoption of digital technologies has significantly transformed the food service industry. Traditional restaurant management methods often involve manual order processing, paper-based record keeping, and inefficient communication between customers and restaurant staff, leading to delays, inaccurate orders, and reduced customer satisfaction.

The proposed Food Order Management System addresses these challenges by providing a centralized web-based platform that automates the complete ordering process. Customers can conveniently browse menus, place orders, and monitor order status from anywhere, while restaurant staff can efficiently

*Corresponding author: manjulakemparaju1506@gmail.com

manage orders, payments, and customer information.

In addition to improving operational efficiency, the system assists restaurant management in monitoring inventory, analyzing sales trends, and generating business reports that support strategic decision-making. The adoption of such a system contributes to increased productivity, reduced operational costs, enhanced service quality, and sustainable business growth.

D. Problem Statement

Many restaurants continue to rely on traditional order management methods, which are often inefficient, time-consuming, and susceptible to human error. Manual order processing may result in incorrect orders, delayed deliveries, poor inventory control, and ineffective communication between customers and restaurant staff. These issues negatively affect customer satisfaction and overall business performance.

The proposed Food Order Management System is designed to overcome these limitations by providing a secure and automated platform for managing food orders and restaurant operations. The system enables customers to place orders online, while administrators can monitor order status, manage menu items, process payments, and maintain customer records through a centralized database.

The implementation of the proposed system improves order accuracy, minimizes operational delays, enhances inventory management, and supports data-driven decision-making through comprehensive reporting features. Consequently, the system provides a reliable and scalable solution for modern restaurant management.

E. Scope

The scope of the proposed Food Order Management System includes the development of a comprehensive web-based platform that supports the complete restaurant ordering lifecycle. The system facilitates customer registration, user authentication, menu management, food ordering, shopping cart management, payment processing, order tracking, and order history maintenance.

For restaurant administrators, the system provides functionalities such as menu management, customer management, order processing, payment monitoring, inventory supervision, and report generation. The centralized database ensures secure storage and efficient retrieval of operational data.

The proposed system is suitable for restaurants, cafés, food courts, and small to medium-sized food service businesses. Furthermore, its modular architecture allows future enhancements, including mobile application support, cloud deployment, online payment gateway integration, QR code-based ordering, artificial intelligence-based food recommendations, and real-time delivery tracking, making it adaptable to future technological advancements.

2. Literature Survey

The development of food ordering and restaurant management systems has gained significant attention due to the

increasing demand for digital platforms that improve operational efficiency and customer convenience. Various researchers have proposed different approaches for automating food ordering, restaurant operations, database management, and customer interaction [1]. A study on online food ordering systems proposed a web-based platform that enables customers to browse menus, select food items, and place orders electronically. The research highlighted that digital ordering systems reduce manual errors and improve order accuracy by maintaining centralized customer and order information [2]. Research on restaurant management systems focused on database-driven applications for managing customers, menu items, billing information, and order records. The proposed system demonstrated that relational database management systems provide efficient storage, retrieval, and processing of restaurant data [3]. A web-based food delivery management system was introduced to enhance communication between customers and restaurant staff. The study emphasized real-time order updates, automated order processing, and improved customer satisfaction through faster service [4]. A study on smart restaurant applications explored the integration of information technology into food service operations. The researchers identified that automated menu management, inventory monitoring, and sales analysis help restaurants optimize resources and improve decision-making [5]. Research on cloud-based food ordering platforms analyzed the advantages of centralized data management and remote accessibility. The study concluded that cloud-enabled systems provide scalability, reliability, and better management of large volumes of customer and transaction data [6]. A paper on mobile-based food ordering applications investigated user-friendly interfaces for improving customer interaction. The research showed that easy navigation, digital menus, and online order tracking increase customer engagement and convenience [7]. A restaurant automation system study focused on integrating ordering, payment processing, and reporting functionalities into a single platform [8]. The research demonstrated that automation reduces workload for restaurant employees and improves overall operational performance [9]. Research on intelligent food recommendation systems explored the use of customer preferences and previous order history to provide personalized food suggestions. The study indicated that recommendation features can enhance customer experience and increase sales [9]. A database management approach for restaurant systems was proposed to organize customer details, food information, order transactions, and payment records efficiently. The research emphasized database normalization and secure data handling for maintaining system reliability [10]. A comprehensive review of digital restaurant management solutions highlighted the importance of integrating ordering, inventory control, analytics, and customer feedback modules. The study concluded that web-based food management systems provide an effective solution for improving restaurant productivity and service quality.

From the above literature studies, it is observed that existing food ordering systems mainly focus on automation, database management, order tracking, and customer convenience.

However, there is still a need for lightweight, cost-effective, and easily deployable systems for small and medium-scale restaurants. The proposed Order Stack: Web-Based Food Order Management System Using PHP and MySQL addresses these requirements by providing an integrated platform for customer ordering, menu management, order processing, and centralized database operations.

3. Design and Analysis

A. How Food Order Management System Works

A Food Order Management System is a database-based application designed to manage the complete food ordering process efficiently. It helps customers place orders, allows administrators to manage menu items, and maintains records of customers, orders, payments, and delivery details. The system uses a Database Management System (DBMS) to store, organize, and retrieve information. The working of the system involves the following components:

- 1) **Database Schema:** The database schema defines the overall structure and organization of the system database. It includes different tables that store and manage important information such as customer details, food items, orders, payments, and order transactions. The schema ensures proper data organization and reduces duplication.
- 2) **Database Tables:** Tables are used to store and manage data in a structured format. The major tables used in a Food Order Management System include:
- 3) **Customer Table:** The Customer table stores information about users who place food orders. It contains details such as customer ID, customer name, contact number, email address, delivery address, and login information.
- 4) **Order Table:** The Order table maintains details of customer orders. It includes order ID, customer ID, order date, order status, payment status, and total order amount. This table helps track and manage all placed orders.
- 5) **Order Details Table:** The Order Details table stores information about individual food items included in each order. It contains details such as order ID, food item ID, quantity, price, and subtotal. This helps maintain complete records of each transaction.
- 6) **Menu Items Table:** The Menu Items table contains details of all available food products. It stores information such as item ID, food name, category, description, price, and availability status. Administrators can add, update, or remove menu items through this table.
- 7) **Payment Table:** The Payment table manages transaction-related information, including payment ID, order ID, payment method, payment date, and payment status. It helps maintain secure and accurate payment records.
- 8) **Relationships Between Tables:** Relationships define the connection between different database tables. For

example, the Customer table is connected with the Order table through customer ID, allowing multiple orders to be linked to a single customer. Similarly, the Order Details table connects orders with menu items using order ID and item ID. These relationships maintain data consistency and integrity.

B. Benefits of Food Order Management System

- 1) **Improved Efficiency:** The system automates the food ordering process, reducing manual work and minimizing errors in order entry, billing, and record maintenance.
- 2) **Faster Order Processing:** It allows customers to place orders quickly and enables restaurants to process, update, and complete orders in less time.
- 3) **Enhanced Customer Experience:** The system provides a convenient platform for customers to view menus, place orders, check order status, and receive better service.
- 4) **Better Inventory Management:** The system helps monitor food item availability and stock levels, preventing shortages and reducing food wastage.
- 5) **Accurate Data Management:** All customer, order, payment, and menu information is stored systematically in the database, making data retrieval and management easier.
- 6) **Reporting and Analysis:** The system can generate reports related to sales, customer orders, popular food items, and business performance, helping administrators make better decisions.
- 7) **Cost Reduction:** By automating daily operations, the system reduces paperwork, saves time, and lowers operational costs.
- 8) **Secure Transaction Management:** The system maintains payment records and protects important information by providing controlled access to authorized users.
- 9) **System Integration:** The Food Order Management System can be integrated with other applications such as billing systems, delivery management systems, and customer relationship management (CRM) tools to improve overall business operations.

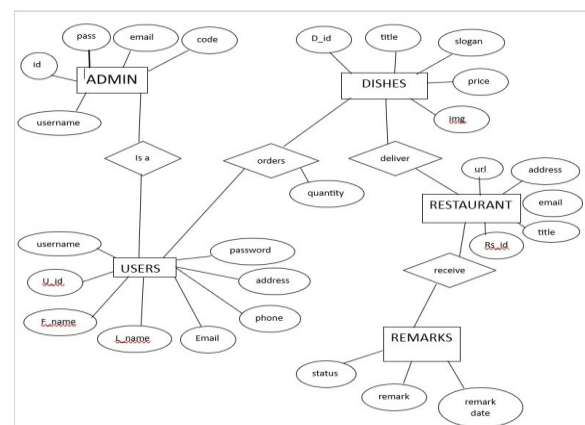


Fig. 1. E-R diagram

C. Schema Diagram

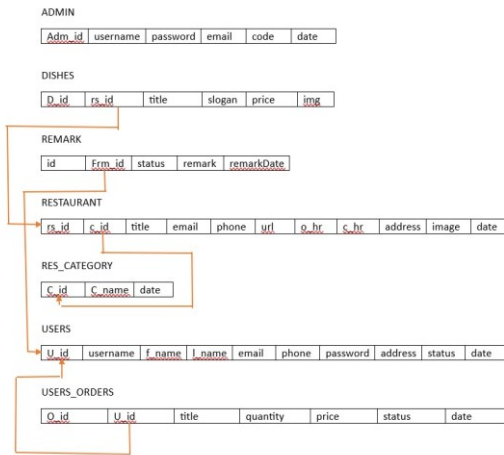


Fig. 2. Schema diagram

4. Requirements Specification

A. Functional Requirements

Functional requirements describe the features and operations that the Food Order Management System should perform. The major functional requirements of the system are:

1. **Customer Registration:** The system should allow new customers to create an account by providing necessary details. Customer information should be stored securely in the database. The system should validate user inputs during registration to avoid duplicate or incorrect data.
2. **User Login:** Registered customers should be able to log in using their username/email and password. The system should authenticate user credentials before providing access. Invalid login attempts should display appropriate error messages.
3. **Menu Management:** The system should display available food items with details such as name, category, description, and price. Administrators should be able to add, update, or remove menu items.
4. **Food Ordering:** Customers should be able to select food items and add them to their order. The system should calculate the total amount based on selected items and quantities. Customers should be able to confirm and place orders.
5. **Order Management:** The system should store order details and maintain order history. Administrators should be able to view and update order status such as *pending*, *confirmed*, or *completed*.
6. **Payment Management:** The system should maintain payment details associated with each order. Payment status should be recorded for future reference.

B. Non-Functional Requirements

Non-functional requirements define the technical environment and quality attributes required for the successful operation of the system.

1) Hardware Requirements

Operating System: Windows 10 or above. **Processor:** Intel

Core i5 8th Generation or higher **RAM:** Minimum 2 GB **Hard Disk:** Minimum 3 GB free storage space

Performance Requirements: The system should provide fast response while processing customer requests. Database operations should be performed efficiently without data loss.

Security Requirements: User login credentials should be protected. Only authorized users should access administrative functions. Customer and transaction information should be stored securely.

C. Data Requirements

Data requirements define the information needed by the system for performing different operations.

1) Customer Registration Data

For registering a new customer, the system requires the following details: Customer ID, Username, Email Address, Password, Confirm Password, Contact Number, and Delivery Address.

2) Login Data

For user authentication, the system requires: Username or Email Address Password

3) Food Item Data

For managing menu items, the system stores: For registering a new customer, the system requires the following details: Food Item ID, Food Name, Category, Description, Price, Availability Status.

4) Order Data

For processing customer orders, the system maintains: Order ID, Customer ID, Food Item Details, Quantity, Total Amount, Order Date, and Order Status.

5) Payment Data

For transaction management, the system stores: Payment ID, Order ID, Payment Method, Payment Date, and Payment Status.

5. Implementation

Pseudo Code:

1. User Authentication:

Function authenticateUser(username, password): Query = "SELECT * FROM users WHERE username = ? AND password = ?" Result = executeQuery(Query, [username, password]) If Result is not empty: Return Result Else: Return null.

2. Fetching Restaurants by Category:

Function getRestaurantsByCategory(category): Query = "SELECT * FROM restaurant WHERE c_id IN (SELECT c_id FROM res_category WHERE c_name = ?)" Restaurants = executeQuery(Query, [category]) Return Restaurants.

3. Placing Food Order:

Function placeOrder(userId, dishId, quantity): Dish = getDishById(dishId) Total_Price = Dish.price * quantity Query = "INSERT INTO users_orders (u_id, title, quantity, price) VALUES (?, ?, ?, ?)" executeQuery(Query, [userId, Dish.title, quantity, Total_Price]) Return "Order placed successfully!" Else: Return "Dish not found!"

4. Retrieving User Orders:

Function getUserOrders(userId):Query = "SELECT * FROM users_orders WHERE u_id = ?" Orders = executeQuery(Query, [userId]) Return Orders.

5. Updating Order Status:

Function updateOrderStatus(orderId, status): Query = "UPDATE users_orders SET status = ? WHERE o_id = ?" executeQuery(Query, [status, orderId]) Return "Order status updated successfully!"

6. Adding Remark to Order:

Function addRemarkToOrder(orderId, remark, status):Query = "INSERT INTO remark (frm_id, status, remark) VALUES (?, ?, ?)"executeQuery(Query, [orderId, status, remark]) Return "Remark added successfully!"

7. Retrieving Remarks for an Order:

Function getRemarksForOrder(orderId): Query = "SELECT*FROM remark WHERE frm_id = ?"Remarks = executeQuery(Query, [orderId]) Return Remarks

8. Execute Database Query:

Function executeQuery(query, params): Connect to Database Execute SQL Query using database connection Return Query Result Close Database Connection.

A. Implementation Aspects

A Food Order Management System implementation involves developing and integrating various features that support efficient food ordering, processing, and management operations. The major implementation features include:

1) Order Placement

The system allows customers to view available menu items, select required food items, specify quantities, and place orders easily.

2) Order Processing

The backend system receives customer orders, verifies details, calculates the total amount, and updates the order status throughout the processing cycle.

3) Inventory Management

The system monitors available food items and ingredient quantities to maintain stock levels and prevent unavailable items from being ordered.

4) Payment Management

The system manages payment details and supports different payment methods while maintaining secure transaction records.

5) Order Tracking

Customers can check the current status of their orders, such as order placed, confirmed, preparing, and delivered.

6) User Authentication:

The system provides secure registration and login functionality, allowing users to create accounts, access profiles, and manage personal information.

7) Feedback and Review Management

Customers can provide feedback and reviews about food items and services, helping improve customer satisfaction and

service quality.

8) Reporting and Analytics

The system generates reports related to sales, customer orders, popular food items, and overall business performance to support decision-making.

9) Admin Dashboard:

An administrative interface allows administrators to manage customers, food items, orders, payments, and system information efficiently.

10) Integration with POS Systems

The system can be integrated with Point of Sale (POS) systems to improve billing operations and synchronize online and offline transactions.

Review Phase:

The Review Phase is an important stage after system development and implementation. It evaluates the performance, effectiveness, and success of the Food Order Management System by comparing the actual results with the expected objectives.

Development Review:

A development review is conducted immediately after project completion to analyze the overall development process. It identifies successful implementations, technical challenges, and possible improvements that can be considered in future projects.

Post-Implementation Review:

A post-implementation review is performed after the system has been in operation for a specific period. It evaluates the actual performance of the system by comparing it with the original requirements, cost expectations, and expected benefits.

Schedule:

```
-- Schedule Table DROP TABLE IF EXISTS schedule;
CREATE TABLE schedule ( schedule_id INT
AUTO_INCREMENT PRIMARY KEY, date DATE NOT
NULL, open_time TIME DEFAULT '12:00:00', close_time
TIME DEFAULT '00:00:00' ); -- Tables Reservation Table
DROP TABLE IF EXISTS tables; CREATE TABLE tables (
tables_id INT AUTO_INCREMENT PRIMARY KEY, t_date
DATE NOT NULL, t_tables INT NOT NULL ); -- Users Table
DROP TABLE IF EXISTS users; CREATE TABLE users (
user_id INT AUTO_INCREMENT PRIMARY KEY,
username VARCHAR(50) NOT NULL, email
VARCHAR(100) NOT NULL, password VARCHAR(255)
NOT NULL, reg_date TIMESTAMP DEFAULT
CURRENT_TIMESTAMP, role_id INT DEFAULT 1 ); --
Foreign Key Relations ALTER TABLE users ADD
CONSTRAINT fk_role FOREIGN KEY (role_id)
REFERENCES role(role_id); ALTER TABLE reservation
ADD CONSTRAINT fk_user FOREIGN KEY (user_fk)
REFERENCES users(user_id); COMMIT;
```

B. PHP Code of Index File

```
<?php require "header.php"; ?> <header class="text-center">
 <a
href="reservation.php" class="btn btn-dark">Make
Reservation</a> </header> <section class="container text-
center"> <h3>Welcome to Shiro Restaurant</h3>  <p>Multi-cuisine
restaurant providing quality food and online reservation
facility.</p> </section> <section class="container text-
center"> <h3>Gallery</h3>   </section>
<section class="container text-center"> <h3>Reservation</h3>
<a href="reservation.php" class="btn btn-outline-
dark">Reserve Now</a> </section> <section
class="container"> <h3>Opening Hours</h3> <form
method="post"> <input type="date" name="date" required>
<button name="check_schedule">Check Time</button>
</form> <?php if(isset($_POST['check_schedule'])) { require
"includes/dbh.inc.php"; $date=$_POST['date']; $result=$conn-
>query("SELECT * FROM schedule WHERE date='$date'");
$row=$result->fetch_assoc(); echo "Opening:
".$row['open_time']." Closing: ".$row['close_time']; } ?>
</section> <footer class="text-center"> <p>Near Cubbon Park,
Bengaluru | Email: restaurant@gmail.com</p> </footer>
<?php require "footer.php"; ?>

```

6. Result

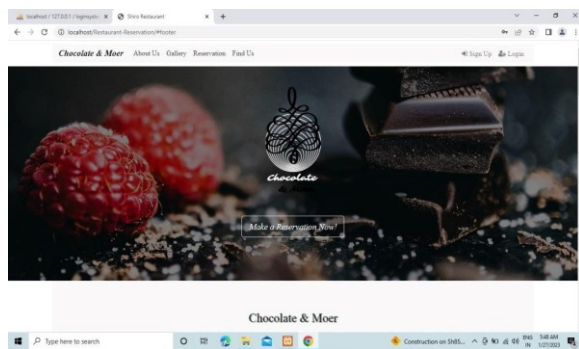


Fig. 3. Home page

Figure 3 illustrates the home page of the restaurant reservation system. The page provides options for users to make reservations, register a new account, and log in to the system. By scrolling down, users can view additional sections such as restaurant details, gallery, reservation information, and other features displayed in the following figures.

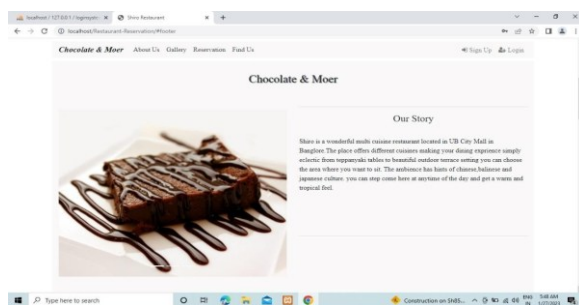


Fig. 4. Restaurant story

Figure 4 displays the restaurant information section, including the restaurant's history, location, and details about its multi-cuisine offerings. The page also allows users to select their preferred dining area for table booking and choose from the available cuisines offered by the restaurant.

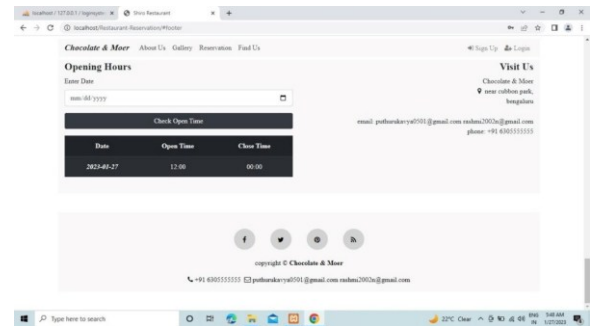


Fig. 5. Visiting page

Figure 5 shows the restaurant's opening hours section, where customers can check the available timings for reservations. If customers have any queries or issues, they can contact the restaurant using the provided email address and phone number or visit the restaurant directly as shown in Figure 5.

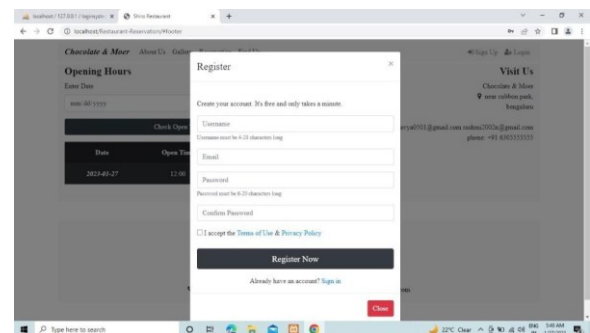


Fig. 6. Registration page

Figure 6 illustrates the customer registration page. New customers can create an account by entering their username, email address, and password. To complete the registration process, users must agree to the Terms and Conditions and Privacy Policy before submitting the registration form.

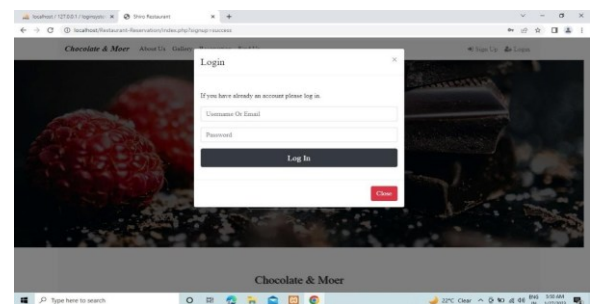


Fig. 7. Login page

After successful registration, the customer can access the login page from the home page, as shown in Figure 7. To log in, the customer must enter their registered username or email address along with the password. Upon successful authentication, the user is granted access to the system and its features.

Fig. 8. New reservation page

Fig. 9. Submit reservation

After successfully completing the registration and login process, as shown in Figures 6 and 7, the reservation page is displayed. Figures 8 and 9 illustrate the reservation form, where customers can enter their first name, last name, reservation date, preferred time, number of guests, and telephone number. Customers can also provide any special requests or additional comments in the remarks section. Finally, after accepting the Terms and Conditions and Privacy Policy, they can submit the reservation request successfully.

Fig. 10. View reservation page

Figure 10 shows the View Reservation page, where customers can view the details of their restaurant table bookings after successfully completing the registration, login, and reservation processes. The page displays the reservation information and also provides an option for customers to make a new reservation whenever required.

7. Conclusion

The Restaurant Reservation System is a web-based application developed to simplify the process of reserving tables in a restaurant. The system allows customers to register, log in, and book tables according to their preferred date, time, and seating requirements. It provides a simple, user-friendly, and efficient platform for managing restaurant reservations.

The application stores customer and reservation details in a secure database, making it easy to maintain, retrieve, and manage information. By automating the reservation process, the system reduces manual effort, improves accuracy, and enhances the overall customer experience.

Although this project is a prototype and includes only the essential features of a restaurant reservation system, it demonstrates the practical implementation of database management and web technologies. Future enhancements may include online payment integration, real-time table availability, reservation cancellation and modification, email or SMS notifications, customer feedback, and administrative reporting.

With these additional features, the Restaurant Reservation System can become a reliable, efficient, and cost-effective solution for restaurants to manage reservations, improve operational efficiency, and provide better customer service.

8. Future Enhancements

The Restaurant Reservation System can be enhanced by incorporating advanced features such as online table booking, secure payment integration, real-time table availability, reservation modification, cancellation facilities, and customer feedback management.

The application can be deployed on a cloud-based or online server, allowing customers to access the system anytime and from anywhere in the world through the internet.

References

- [1] A. Silberschatz, H. F. Korth, and S. Sudarshan, *Database System Concepts*, 7th ed. New York, NY, USA: McGraw-Hill Education, 2020.
- [2] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 7th ed. Hoboken, NJ, USA: Pearson Education, 2016.
- [3] I. Sommerville, *Software Engineering*, 10th ed. Boston, MA, USA: Pearson Education, 2016.
- [4] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 9th ed. New York, NY, USA: McGraw-Hill Education, 2020.
- [5] L. Welling and L. Thomson, *PHP and MySQL Web Development*, 5th ed. Boston, MA, USA: Addison-Wesley Professional, 2017.
- [6] R. Nixon, *Learning PHP, MySQL & JavaScript*, 6th ed. Sebastopol, CA, USA: O'Reilly Media, 2021.
- [7] P. Deitel and H. Deitel, *Java How to Program*, 11th ed. New York, NY, USA: Pearson Education, 2018.
- [8] R. Lerdorf, K. Tatroe, and P. MacIntyre, *Programming PHP*, 4th ed. Sebastopol, CA, USA: O'Reilly Media, 2020.
- [9] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 4th ed. Cambridge, MA, USA: MIT Press, 2022.
- [10] W3Schools, "PHP Tutorial, MySQL Tutorial, and HTML/CSS Documentation," Aug. 2026. [Online]. Available: <https://www.w3schools.com>