1. Introduction

This report provides a detailed analysis of the Restaurant Management System project. The system is designed to streamline restaurant operations, including user management, menu item management, and order processing. It features user authentication, role-based access control, and a database-driven architecture.

2. Project Team

- Urooba Gohar (22p-9216)

- Hina Rashid (22p-9198)

- Laraib Shahid (22p-0503)

3. System Overview

The Restaurant Management System is a web application developed using PHP, HTML, CSS, and MySQL. It allows customers to view the menu, place orders, and track their order status. Administrators can manage the menu items, view order details, and update order statuses. The system ensures data persistence through a MySQL database and uses session management for user authentication.

4. Key Features

- **User Authentication and Management:**

  - o **Registration:** New users can register by providing their name, email, password, address, and phone number (register.php). The system checks for duplicate emails during registration.

  - o **Login:** Registered users can log in using their email and password (login.php). The system authenticates users and sets session variables to manage user sessions.

  - o **Logout:** Users can log out of the system, which destroys their session (logout.php).

  - o **User Roles:** The system supports two user roles:

    - ▪ **Admin:** Administrators have full access to manage menu items and orders.

    - ▪ **User:** Regular users can view the menu and place orders.

- **Menu Management (Admin):**

  - **Add Items:** Administrators can add new menu items, including images, names, prices, and categories (add_items.php).

  - **View Items:** Administrators can view a list of all menu items in a tabular format (view_items.php).

- **Order Management:**

  - **Place Orders:** Users can place orders for menu items displayed on the main page (index.php, order_item.php). Orders are initially marked as "pending."

  - **View Orders (Admin):** Administrators can view a comprehensive list of all orders, including customer details and ordered items (view_order_items.php).

  - **View Orders (User):** Users can view their own order history (view_orders.php).

  - **Update Order Status (Admin):** Administrators can update the status of orders (e.g., from "pending" to "delivered") (update_item_status.php).

- **User Interface:**

  - The system provides separate dashboards for administrators (admin_dashboard.php) and users (user_dashboard.php).

  - The main menu is displayed on the index.php page, allowing users to browse and place orders.

  - Basic styling is implemented using CSS to provide a functional and readable interface.

- **Database Management:**

  - The system uses MySQL to store data, including user information, menu items, and order details.

  - The database connection is established in db.php, which is included in other PHP files to interact with the database.

- **Error Handling:**

  - The system includes basic error handling to display database errors and inform users of registration or login issues.

5. File Descriptions

- add_items.php: Allows administrators to add new menu items to the database, including image uploads and item details.

- admin_dashboard.php: The main dashboard for administrators, providing navigation links to manage menu items and orders.

- db.php: Contains database connection details and establishes a connection to the MySQL database.

- index.php: Displays the restaurant menu to users and enables them to place orders.

- login.php: Handles user login, authenticates users against the database, and sets session variables.

- logout.php: Handles user logout by destroying the current session.

- order_item.php: Processes user orders, inserting them into the orders table with a "pending" status.

- register.php: Handles user registration, storing user details in the users table.

- update_item_status.php: Allows administrators to update the status of orders.

- user_dashboard.php: Provides a basic dashboard for regular users.

- view_items.php: Displays a list of all menu items to administrators.

- view_order_items.php: Displays a detailed view of all orders to administrators, including customer and item information, and allows for updating order status.

- view_orders.php: Displays the order history for individual users.

6. Detailed Analysis

- **Authentication and Authorization:**

  - The system effectively uses sessions to manage user login states.

  - Role-based access control is implemented to restrict access to admin functionalities.

- **Data Handling:**

- o The system uses prepared statements to prevent SQL injection vulnerabilities.

- **User Interface:**

    - o The UI is functional.

- **Functionality:**

    - o The system covers the core functionalities of a restaurant management system.

- **Database Design:**

    - o The database schema is well-structured to support the system's functionalities.

7. Conclusion

The Restaurant Management System provides a solid foundation for managing restaurant operations. It includes essential features such as user authentication, menu management, and order processing. However, there are areas for improvement, particularly in security, user experience, and functionality. Addressing these areas will significantly enhance the system's robustness, usability, and overall quality.