Saidou Jalloh

Restaurant Care

http://localhost/restaurant_care/

1. Introduction

The Restaurant Care System is designed to streamline the process of managing customer reservations. It allows customers to make table bookings with special requests, modify existing reservations, and specify dietary preferences. The project adheres to modern web development practices, ensuring a user-friendly and responsive interface.

2. System Features

- Make a Reservation: Customers can reserve a table by providing their name, contact information, and other preferences.

- Manage Reservations: View all reservations in a tabular format with options to modify or delete existing reservations.

- Modify Reservations: Edit reservation details, including date, time, table preferences, and special requests.

- Update Preferences: Specify and update dietary preferences to accommodate customer needs.

- Error Handling: The system gracefully handles errors, ensuring a seamless user experience.

3. Code File Overview

Frontend Files

1. `styless.css`: Defines the styling for the entire system. Implements a responsive and modern design for forms, tables, and buttons.

2. `scripts.js`: Adds interactive functionality to enhance the user experience. Currently includes a console log for system initialization.

3. `home.php`: Serves as the landing page of the system. Provides navigation to 'Reserve Your Table' and 'Manage Your Reservations.'

4. `addReservation.php`: Displays a form for creating a new reservation. Includes fields for customer details, reservation date and time, table number, and special requests.

5. `modifyReservation.php`: Enables users to edit existing reservations. Pre-fills the form with current reservation details for easy modifications.

6. `updatePreferences.php`: Allows customers to update their dietary preferences. Saves preferences to the database for future reference.

7. `viewReservations.php`: Displays all reservations in a tabular format. Includes options to modify or cancel each reservation.

Backend Files

8. `index.php`: Routes requests to the appropriate page or action. Utilizes the `RestaurantServer.php` file for request handling.

9. `RestaurantDatabase.php`: Handles database interactions, such as adding, updating, deleting, and fetching reservations. Ensures data integrity with prepared statements to prevent SQL injection.

10. `RestaurantServer.php`: Manages the application's logic and routes user actions. Processes form submissions, retrieves data from the database, and renders the appropriate templates.

Database Script

11. `create_tables.sql`: Defines the database schema for the system. Includes three tables:

- `Customers`: Stores customer details.

- `Reservations`: Tracks reservation details, linked to the `Customers` table via a foreign key.

- `Preferences`: Records dietary preferences for customers.

4. Flow of Execution

1. Home Page: Displays links for making a reservation and managing reservations. Routes the user to the respective actions in `RestaurantServer.php`.

2. Add Reservation: User fills out the form in `addReservation.php`. Data is sent to the server and processed by `addReservationWithPreferences` in `RestaurantDatabase.php`.

3. View Reservations: Fetches all reservations using `getAllReservations` in `RestaurantDatabase.php`. Displays the data in a tabular format in `viewReservations.php`.

4. Modify Reservation: User edits an existing reservation using the form in `modifyReservation.php`. Updates the data in the database via `updateReservation` in `RestaurantDatabase.php`.

5. Delete Reservation: Deletes a reservation by calling `deleteReservation` in `RestaurantDatabase.php`.

5. Conclusion

The Restaurant Care System is a fully functional web application that simplifies the reservation process. It is built with PHP and MySQL, ensuring a robust backend, while the frontend uses responsive design principles for a seamless user experience. The modular structure allows for easy scalability and maintenance.

Key Strengths

- User-friendly interface with a clean layout.

- Secure backend operations with prepared statements.

- Flexibility to add future features, such as analytics or online payments.

Potential Improvements

- Add a search feature to filter reservations by customer name or date.

- Include email notifications for reservation confirmations or updates.

- Enhance the design with advanced JavaScript interactivity.