Ali Cruz

Restaurant Portal Project Report

The Restaurant Portal project is designed to help restaurant managers handle reservations and customer information through a web interface. This report explains how the system works, focusing on the "RestaurantServer.py" and "restaurantDatabase.py" files, as well as the SQL code that sets up the database.

The "RestaurantServer.py" file sets up a web server that processes requests related to reservations and customer management. The main part of this file is the "RestaurantPortalHandler" class. This class handles custom HTTP GET and POST requests.

When the "RestaurantPortalHandler" is initialized, it connects to the database using the "RestaurantDatabase" class. The "do_POST" method processes requests to add, modify, and delete reservations and customers. When a POST request is received, it reads the form data submitted by the user, interacts with the database, and then responds with an HTML page. Similarly, the "do_GET" method processes requests to display different web pages, such as the home page, the form for adding a reservation, the list of reservations, and the list of customers. These HTML responses include embedded CSS to make the pages visually appealing and easy to navigate.

The "restaurantDatabase.py" file contains the "RestaurantDatabase" class, which handles all interactions with the MySQL database. This class is responsible for establishing and maintaining the database connection using the provided credentials. It includes methods to add, delete, and modify both reservations and customers. For instance, the "addReservation" method inserts a new reservation into the database, ensuring that the associated customer exists. If the customer does not exist, it first adds the customer to the "Customers" table. The "getAllReservations" method retrieves all reservation records from the database, while "getAllCustomers" retrieves all customer records.

In addition to basic operations like adding and deleting records, the "RestaurantDatabase" class also manages customer preferences. The "modifyCustomerPreferences" method updates a customer's dining preferences, such as their favorite table and dietary restrictions, while "getCustomerPreferences" retrieves these preferences from the database. The class also uses stored procedures to perform complex operations efficiently. For example, the "findReservations" stored procedure retrieves all reservations for a specific customer, and the "addSpecialRequest" stored procedure updates the special requests for a particular reservation.

The SQL code sets up the database schema and stored procedures needed for the application. It creates three main tables: "Customers", "Reservations", and "DiningPreferences". The "Customers" table stores customer details such as their name and contact information. The "Reservations" table records reservation details, including the associated customer, reservation time, number of guests, and special requests. The "DiningPreferences" table keeps track of customer preferences like favorite table and dietary restrictions.

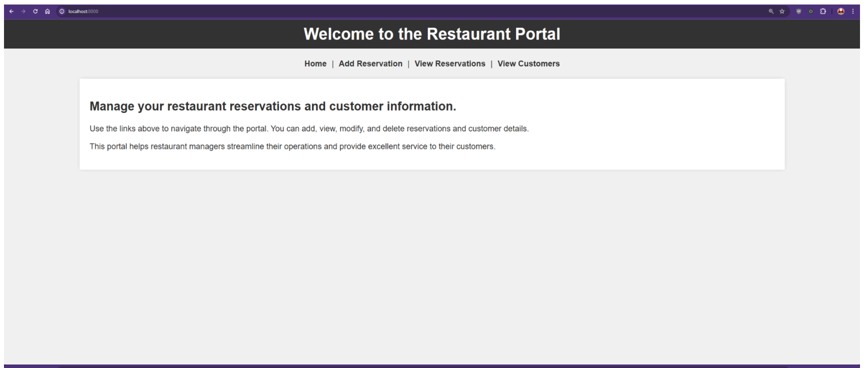
Stored procedures defined in the SQL code help simplify complex database operations. For instance, the "findReservations" procedure retrieves all reservations for a specific customer, while the "addSpecialRequest" procedure updates special requests for a reservation. The "addReservation" procedure ensures that a customer exists before adding a reservation, and if the customer does not exist, it first adds the customer to the database. The initial data inserted into the "Customers", "Reservations", and "DiningPreferences" tables serves as sample data for testing and demonstration purposes.

The Restaurant Portal project helps restaurant managers easily handle reservations and customer information. This system allows managers to add, modify, view, and delete reservations and customer details.

Appendix

Restaurant Portal Demonstration: https://youtu.be/5XEu7Y_vXNc

Screenshots:



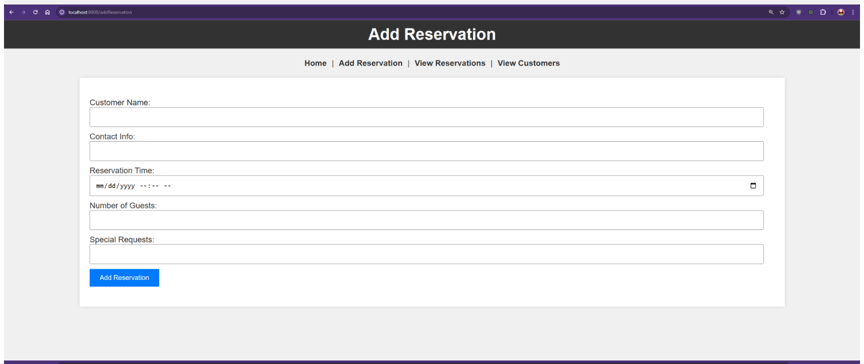
Welcome to the Restaurant Portal

Home | Add Reservation | View Reservations | View Customers

Manage your restaurant reservations and customer information.

Use the links above to navigate through the portal. You can add, view, modify, and delete reservations and customer details.

This portal helps restaurant managers streamline their operations and provide excellent service to their customers.



Add Reservation

Home | Add Reservation | View Reservations | View Customers

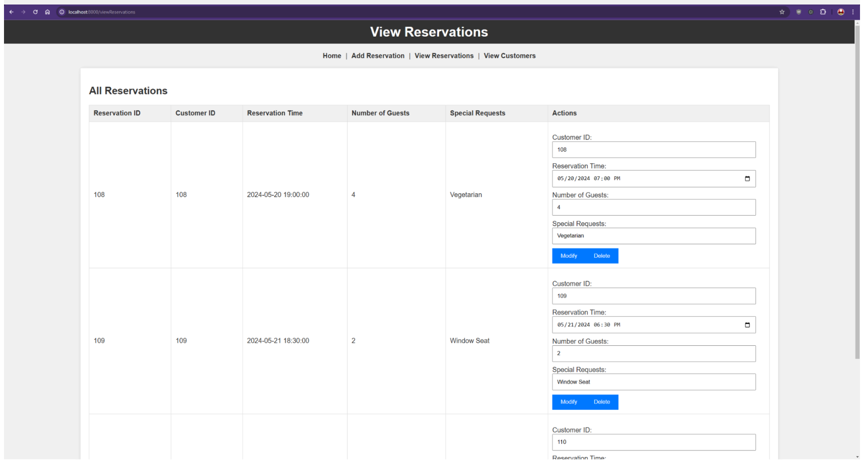
Customer Name:

Contact Info:

Reservation Time: mm/dd/yyyy --:-- --

Number of Guests:

Special Requests:



View Reservations

Home | Add Reservation | View Reservations | View Customers

All Reservations

Reservation ID	Customer ID	Reservation Time	Number of Guests	Special Requests	Actions
108	108	2024-05-20 19:00:00	4	Vegetarian	Customer ID: 108 Reservation Time: 05/20/2024 07:00 PM Number of Guests: 4 Special Requests: Vegetarian Modify Delete
109	109	2024-05-21 18:30:00	2	Window Seat	Customer ID: 109 Reservation Time: 05/21/2024 06:30 PM Number of Guests: 2 Special Requests: Window Seat Modify Delete
					Customer ID: 110 Reservation Time:

Link to GitHub repository: https://github.com/AliCruz1/bookish-octo-barnacle