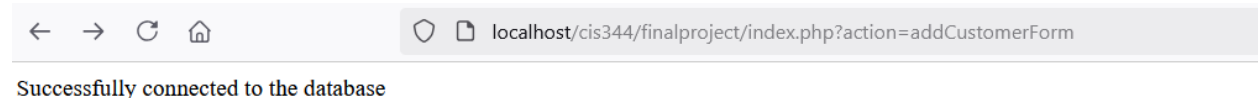


Final Project Report

General Outline

The Restaurant Server Portal is designed so that customers can make a reservation at the restaurant and set their preferred time, number of guests and notes to the restaurant about special requests. Another primary feature is the ability of the restaurant to view all the reservations recorded. Extra features also include the ability to add a customer and get customer preferences. Optional features were modifying the reservations table by allowing deleting rows and changing field entries, but I did not execute those bonus features.

User Guide



Restaurant Portal

[Add Reservation](#) | [View Reservations](#) [Add a Customer](#) [View a Preference for a Customer](#) Successfully connected to the database

To add a reservation. Go to the home page seen above and select Add Reservation. The form below should be displayed. Enter the customer ID if there is one, the reservation time, the number of guests and any notes you have for the restaurant. Click submit.

A screenshot of a web browser window showing the 'Add Reservation' form. The address bar shows the URL 'localhost/cis344/finalproject/index.php?action=addReservation'. Below the address bar, a message reads 'Successfully connected to the database.' The form has the title 'Add Reservation' and contains the following fields: 'Customer ID:' with a text input, 'Reservation Time:' with a date and time picker, 'Number of Guests:' with a numeric input, and 'Special Requests:' with a text area. A 'Submit' button is at the bottom left of the form.

To view all the reservations on file click View Reservations on the tabs above. This should appear.

← → ↺ 🏠 localhost/cis344/finalproject/index.php?action=viewReservations
Successfully connected to the database.

All Reservations

Reservation ID	Customer ID	Reservation Time	Number of Guests	Special Requests
1	1	2024-12-15 19:00:00	2	Window seat
2	2	2024-12-16 18:30:00	4	no almonds
3	3	2024-12-17 20:00:00	3	Vegetarian menu
4	4	2024-12-18 21:00:00	5	None
5	5	2024-12-19 19:30:00	2	Celebrating anniversary
6	1	2024-12-19 19:50:00	7	Vegan
7	1	2024-12-13 08:00:00	1	Vegan
8	1	2024-12-13 08:00:00	1	Vegan
9	1	2024-12-13 08:00:00	1	Vegan
10	1	2024-12-13 08:00:00	1	Vegan
11	1	2024-12-13 08:00:00	1	Vegan
12	1	2024-12-13 08:00:00	1	Vegan
13	3	2024-12-13 08:00:00	1	Vegan
14	3	2024-12-13 08:00:00	1	Vegan
15	2	2024-12-16 13:40:00	5	Vegan
17	1	2024-05-14 08:00:00	1	lactose intolerance
18	1	2024-05-14 08:00:00	1	lactose intolerance
19	2	2024-12-17 09:00:00	5	No almonds
20	2	2024-12-16 09:00:00	3	No peanuts.

[Back to Home](#) Successfully connected to the database.

To add a customer into the system. Click the tab Add a Customer. Enter Customer Name and Contact Info in the file below and the new customer should be updated in a customers table row.

← → ↺ 🏠 localhost/cis344/finalproject/index.php?action=addCustomer
Successfully connected to the database.

Add New Customer

Customer Name:

Contact Info:

[Back to Home](#) Successfully connected to the database.

To view preferences for a specific customer, Enter the Customer ID as seen below and click View Preferences as seen below. A list should appear. An error message appeared for mine

← → ↺ 🏠 localhost/cis344/finalproject/templates/viewPreferences.php

View Customer Preferences

Enter Customer ID:

[Back to home](#)

Errors

The first error I encountered was because I had not saved home.php in a templates sub folder.

← → ↺ 🏠 localhost/cis344/finalproject/RestaurantServer.php
Successfully connected to the database
Warning: include(templates/home.php): Failed to open stream: No such file or directory in C:\xampp\htdocs\cis344\finalproject\RestaurantServer.php on line 27
Warning: include(): Failed opening 'templates/home.php' for inclusion (include_path='C:\xampp\php\PEAR') in C:\xampp\htdocs\cis344\finalproject\RestaurantServer.php on line 27

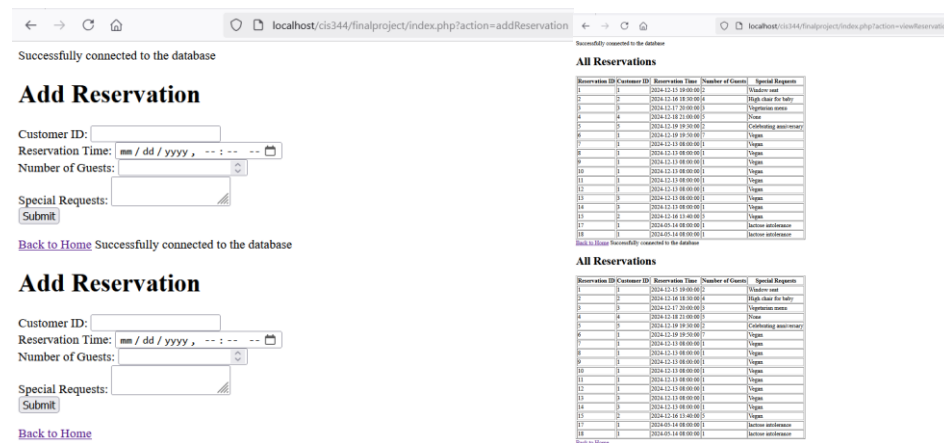
Once I did this home.php was able to run successfully. I also had to create an index.php file in order to access home.php from RestaurantServer.php.

Another error occurred when running RestaurantServer.php. The home page was showing but once I clicked on either the add reservations tab or the view reservations tab, this error would display.



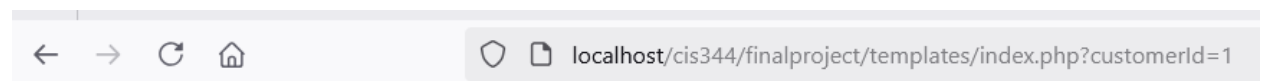
The solution to this was moving index.php from the templates folder and moving the server and database files as well to the primary final project folder.

Once I did this the links to the tabs were displaying but they were showing double results.



I'm still trying to fix this error. I had insert exit() on the last line of the if else statement but it did not fix this issue.

I also encountered this error while running viewPreferences.php



Not Found

The requested URL was not found on this server.

Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12 Server at localhost Port 80

I was unable to fix it too. Other issues that I was unable to resolve were making sure the customer information added entered the Customers table.

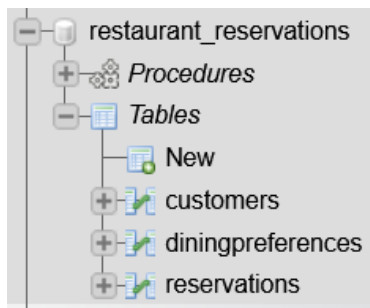
Technical Details

This project came with 2 source codes and 3 template codes. The two major codes were RestaurantServer.php and restaurantDatabase.php. These 2 codes required minor additions of writing 2 required methods: the addCustomer method and the viewPreferences method. I did not do bonus features, so these are the only methods I added. The addCustomer method receives input from the user about their name and contact information and updates it into the Customers table whereas the viewPreferences method enables the user to input the customerId and displays the rows in the reservations table with special requests from the customer. The source code came with 2 functioning methods: the addReservation method with adds rows to the reservations table and the viewReservations table which displays all the rows in the reservations table. These methods are connected to the templates addReservation.php, viewReservations.php, addCustomer.php and viewPreferences.php which not only provide a front-end user interface for the input of data but also feed those inputs to the tables.

The template home.php provides and organizes the tabs for the portal, and we had to create index.php which is the backbone of the portal, and all the templates and source codes connect to this file.
























Database Design


















For this project, we were required to create 3 tables: customers, reservations and the dining-preferences table. I used phpMyAdmin to create these tables.



We were also required to populate each table with at least five rows.



	customerId	customerName	contactInfo
<input type="checkbox"/> Edit <input type="image"/> Copy <input type="image"/> Delete	1	John Denga	johndenga@gmail.com
<input type="checkbox"/> Edit <input type="image"/> Copy <input type="image"/> Delete	2	Jane Smith	janesmith@outlook.com
<input type="checkbox"/> Edit <input type="image"/> Copy <input type="image"/> Delete	3	Emily Hassan	emilyhassan@yahoo.com
<input type="checkbox"/> Edit <input type="image"/> Copy <input type="image"/> Delete	4	Adam Perez	adamp@gmail.com
<input type="checkbox"/> Edit <input type="image"/> Copy <input type="image"/> Delete	5	Sarah Lee	sarahlee@live.com

Extra options					
 	▼ reservationId	customerId	reservationTime	numberOfGuests	specialRequests
<input type="checkbox"/>   	1	1	2024-12-15 19:00:00	2	Window Seat
<input type="checkbox"/>   	2	2	2024-12-16 18:30:00	4	High Chair for baby
<input type="checkbox"/>   	3	3	2024-12-17 20:00:00	3	Vegetarian Menu
<input type="checkbox"/>   	4	4	2024-12-18 21:00:00	5	None
<input type="checkbox"/>   	5	5	2024-12-19 19:30:00	2	Celebrating anniversary
<input type="checkbox"/>   	6	1	2024-12-19 19:50:00	7	Vegan
<input type="checkbox"/>   	7	1	2024-12-13 08:00:00	1	Vegan

 	▼ preferenceId	customerId	favoriteTable	dietaryRestrictions
<input type="checkbox"/>   	1	1	Table 5	None
<input type="checkbox"/>   	2	2	Table 3	Gluten-free
<input type="checkbox"/>   	3	3	Table 8	Vegan
<input type="checkbox"/>   	4	4	Table 10	None
<input type="checkbox"/>   	5	5	Table 2	Dairy-free

We were also required to do 3 stored procedures: the addReservation procedure to add a reservation, the addSpecialRequest procedure to add a special request to a particular reservationId and the findReservations procedure to pull out reservations for a particular customerId.

Routines

☐ Check all  Export  Drop

	Name	Type	Returns	
<input type="checkbox"/>	addReservation	PROCEDURE		   
<input type="checkbox"/>	addSpecialRequest	PROCEDURE		   
<input type="checkbox"/>	findReservations	PROCEDURE		   

Execute routine `addReservation`

Routine parameters

Name	Type	Function	Value
p_customerId	INT	<input type="text"/>	<input type="text"/>
p_customerName	VARCHAR	<input type="text"/>	<input type="text"/>
p_contactInfo	VARCHAR	<input type="text"/>	<input type="text"/>
p_reservationTime	DATETIME	<input type="text"/>	<input type="text"/>
p_numberOfGuests	INT	<input type="text"/>	<input type="text"/>
p_specialRequests	VARCHAR	<input type="text"/>	<input type="text"/>

Execute routine `addSpecialRequest`

Routine parameters

Name	Type	Function	Value
inputReservationId	INT	<input type="text"/>	<input type="text"/>
requests	VARCHAR	<input type="text"/>	<input type="text"/>

Execute routine 'findReservations'

Routine parameters

Name	Type	Function	Value
inputCustomerId	INT		2

Go

Close

SET @p0='2'; CALL `findReservations`(@p0);

Execution results of routine 'findReservations'

reservationId	customerId	reservationTime	numberOfGuests	specialRequests
2	2	2024-12-16 18:30:00	4	High Chair for baby
15	2	2024-12-16 13:40:00	5	Vegan
19	2	2024-12-17 09:00:00	5	No almonds
20	2	2024-12-16 09:00:00	3	No peanuts.

Security Measures

Users of the 4 methods used are only allowed to input data and output data and not to modify it. The restaurantDatabase.php code requires not only knowledge of the name of the database but also the password to the database. However, front end users do not require any passwords to carry out their business. Moreover, in regards to the SQL code, stored procedures are useful as a security procedure as they permit specific users or roles to execute certain procedures.

Lesson and Challenges

This was a deep dive into the world of methods. I have not previously used them and only used separate php files in previous projects. I had challenges saving data in the correct folders. In the future, I will format the front end of my work better. Overall, this project and class has helped me to understand databases better and I will carry this knowledge with me even though I will not pursue a career as a developer or programmer.