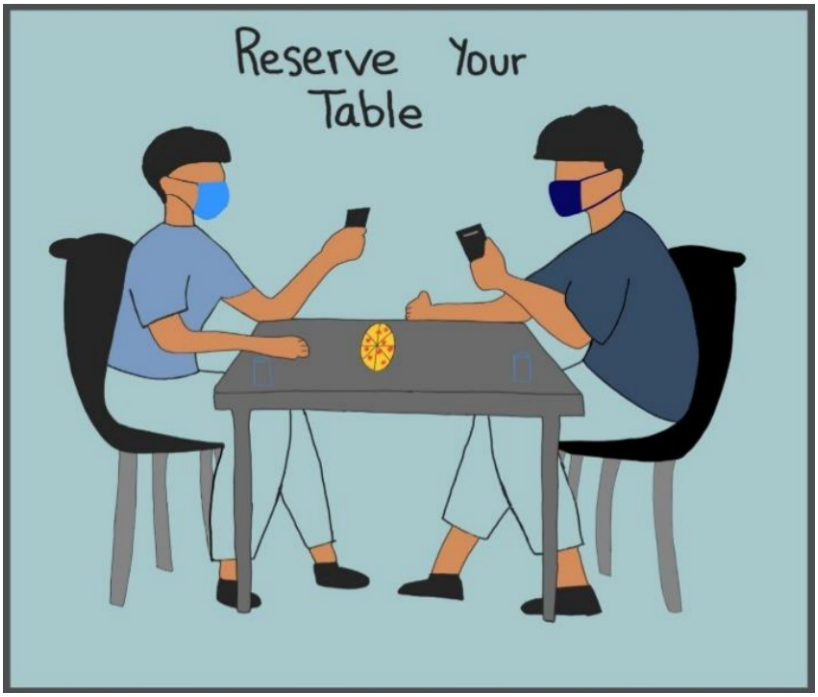
Coding Project Final Report



Reserve Your Table

Prepared by

Matthew Alvero
Ramzi Rimawi
Ryan Thommes
Thomas Sobczak
for use in CS 440
at the
University of Illinois Chicago

November 2021

Table of Contents

List of Figures

I Project Description

1 Project Overview

Reserve Your Table is an application meant to help bring people together during the current COVID-19 pandemic. The project strives to allow users to reserve a table at some of their favorite restaurants, browse the restaurant's menu, order food, and pay all from the app itself. It is designed to help keep unknown interactions to a minimum while still being able to experience dining in with friends and family in a safe manner.

2 Project Domain

The goal of this project is to make the customer feel like they are safe and comfortable when dining in during these hard times of the pandemic. To achieve this the project will be designed in a way for the users to be completely contact-free from the world except for who they choose to be in contact with. The mobile application will be downloaded from the Android and Apple market. The users will then be able to then search for a restaurant, make a reservation, order, and pay all from their fingertips without any interactions with the restaurant.

3 Relationship to Other Documents

Our coding project was built off of the development project requirement document of group 20 from the Spring 2021 semester, which was prepared by Juan Vazquez, Nguyen Hoa Pham, Gurleen Kaur, and Jujhar Singh. This document uses the same UML diagram as the one in the project description document described in [1] as the main context of work for the project.

4 Naming Conventions and Definitions

4a Definitions of Key Terms

User: A person using the application to dine into a restaurant.

Customer: A person that signed up for an account and is ready to use it.

Restaurant portal: The restaurant facing side of the application.

4b UML and Other Notation Used in This Document

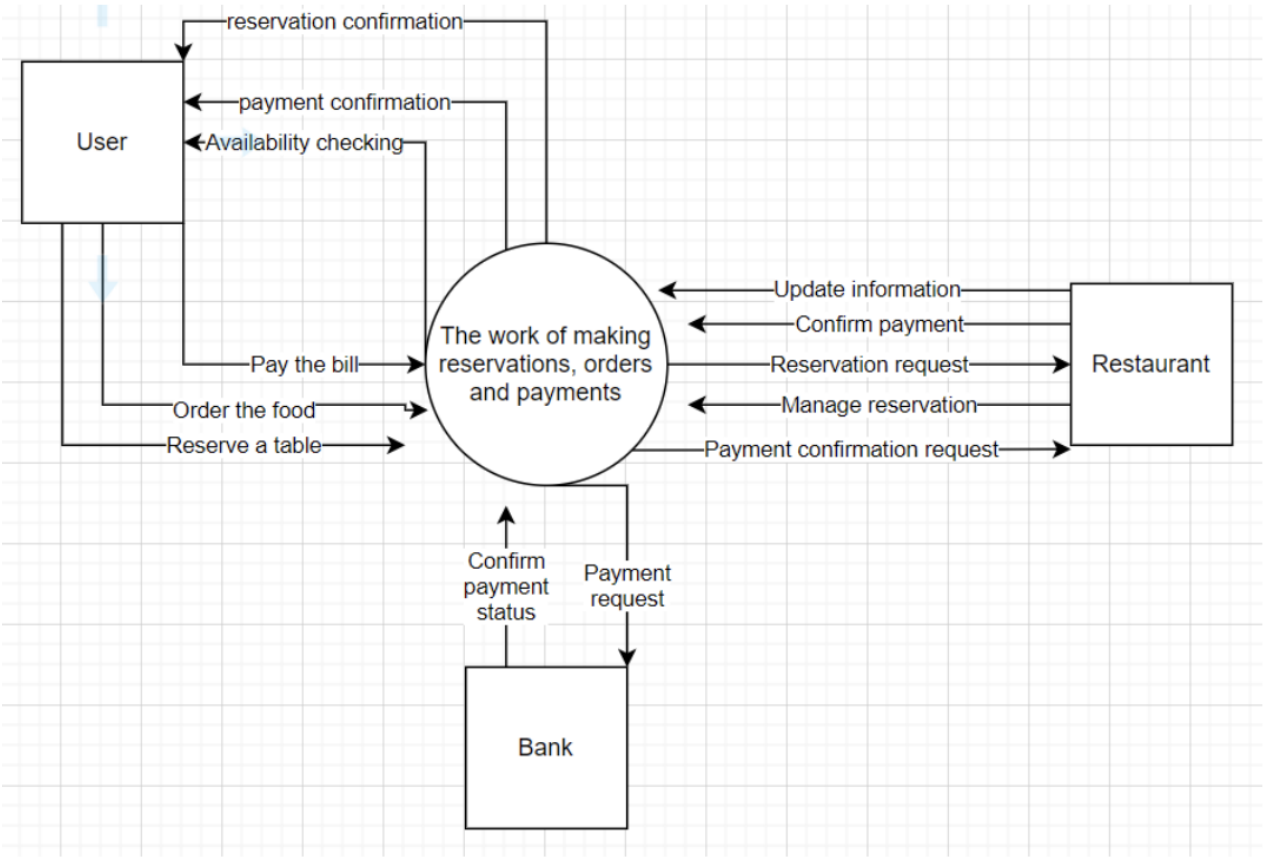


Figure 1 - Context of the work of *ReserveYourTable* as described by Vazquez, Hoa Pham, Kaur, and Singh in [1].

4c Data Dictionary for Any Included Models

Making a reservation = user + {restaurant + time/date/#guests of reservation + order from menu + payment}

Accepting reservations = employee {reservation + confirmation}

II Project Deliverables

Our group has completed a functioning prototype of the described application. We created a customer-facing application to reserve a table, order food, and pay, as well as a restaurant-facing application where restaurant employees can manage and review orders, restaurant information, reservations, and seating. Both sides of the application, customer and restaurant, are supported by a database in the backend running on AWS and MySQL.

5 First Release

The first release date was October 8, 2021, where we demonstrated our early application to Teaching Assistant Mojtaba Malepourshahraki. For the first release of our application, we had completed the front-end development for the customer side of the application using ASAP.NET framework. We made the home page of the application with an embedded Google Map and a banner across the top of the page with “Help”, “Signup”, and “Login” buttons.

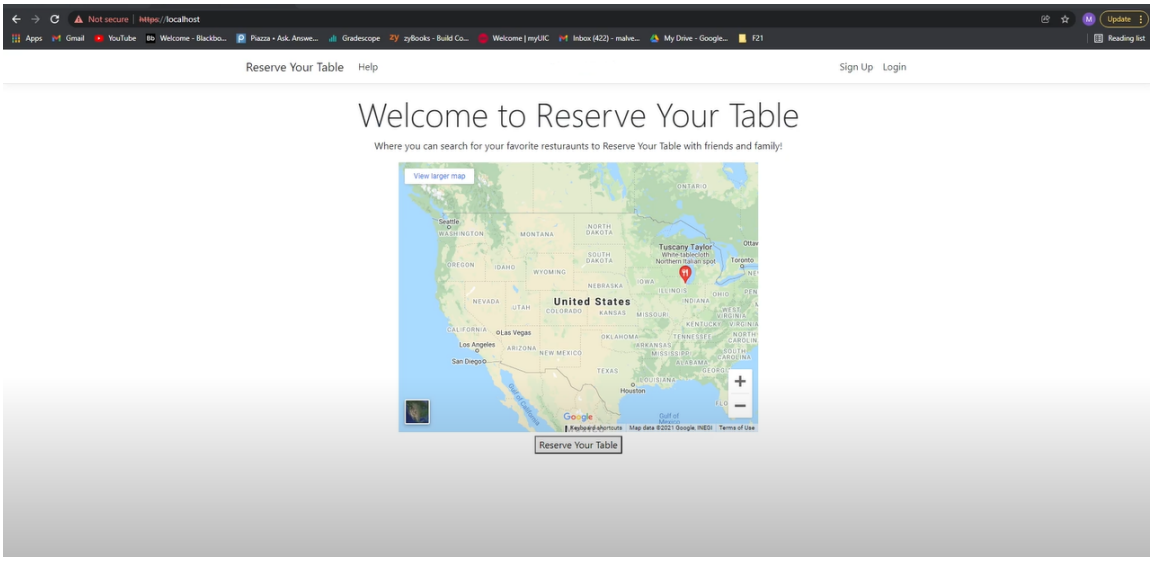


Figure 2 - User home page

After clicking the “Reserve Your Table” button on the home page under the embedded map, the next page was a reserve page where the customer can enter information about the restaurant, table size, date, and time of the reservation.

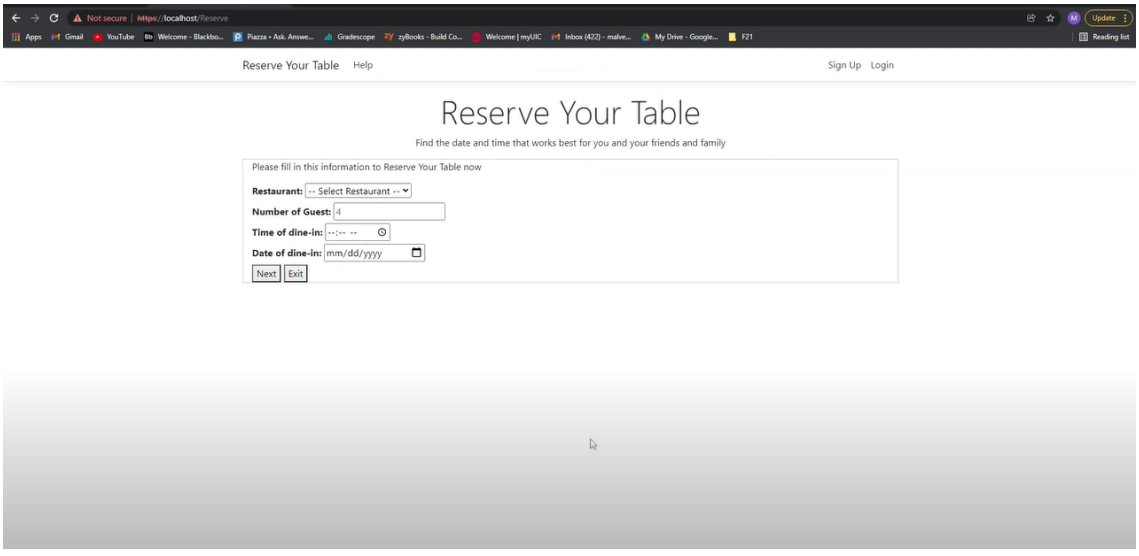


Figure 3 - Table reservation page

The next page was an order page where the customer could choose which menu items they want to order.

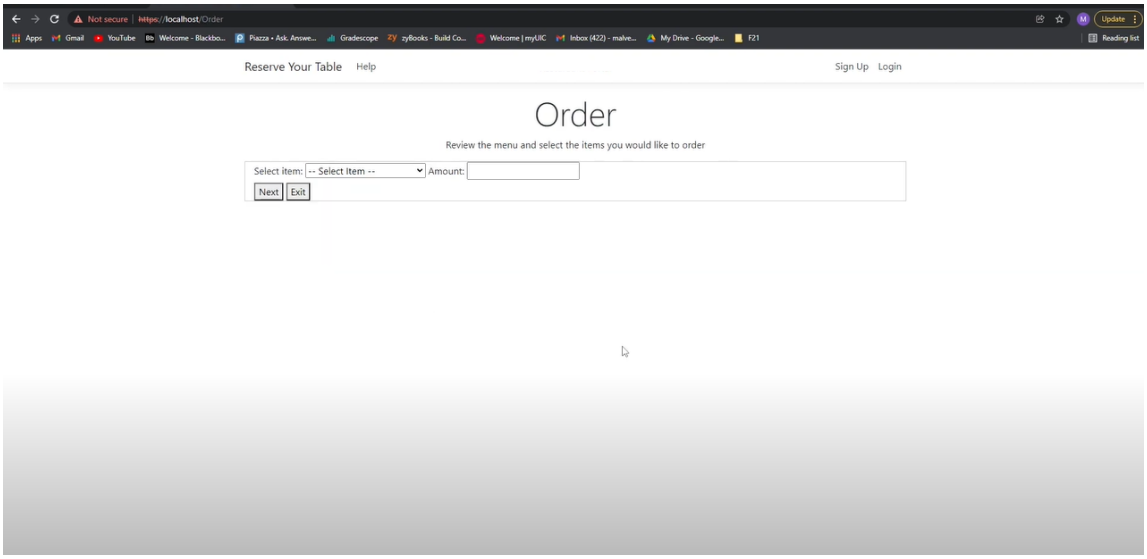


Figure 4 - Order page

The next page was a payment page where the customer could input credit card information to complete the payment of their order.

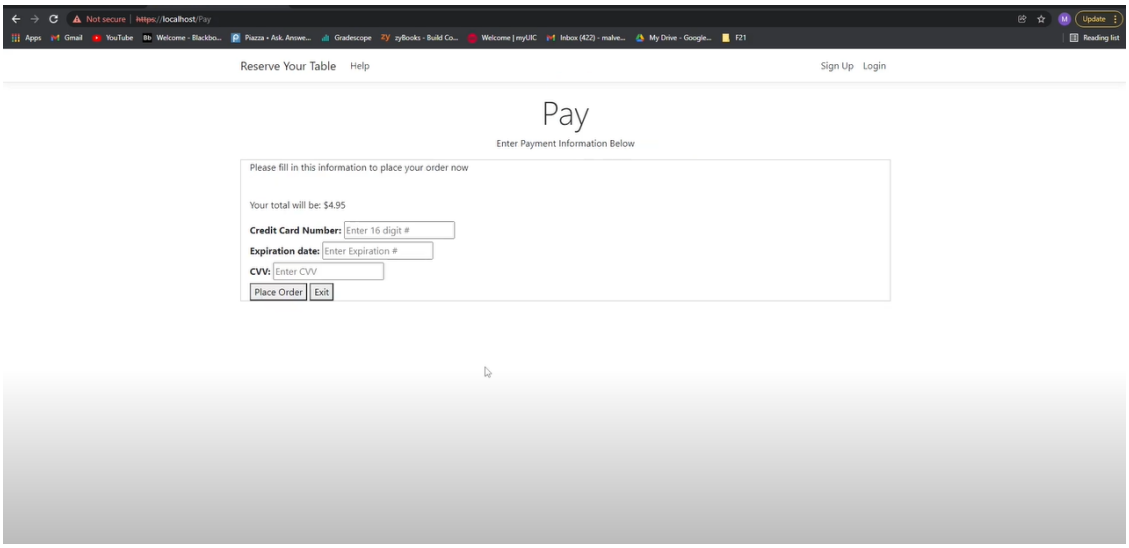


Figure 5 - Payment page

Mojtaba gave us some comments following the demonstration including having a real sample restaurant to choose from with real menu items, and also creating a backend to save the selected information on a database.

6 Second Release

The second release date was November 5, 2021, where we demonstrated the second release of our application to TA Mojtaba. For the second release of the application, we completed the front-end development for the restaurant side of the application which we called the “Restaurant Portal”. Another feature we added in this release was that we successfully connected our application to a database using Entity Framework Core with MySQL. The first page of our Restaurant Portal was a login page for restaurants.

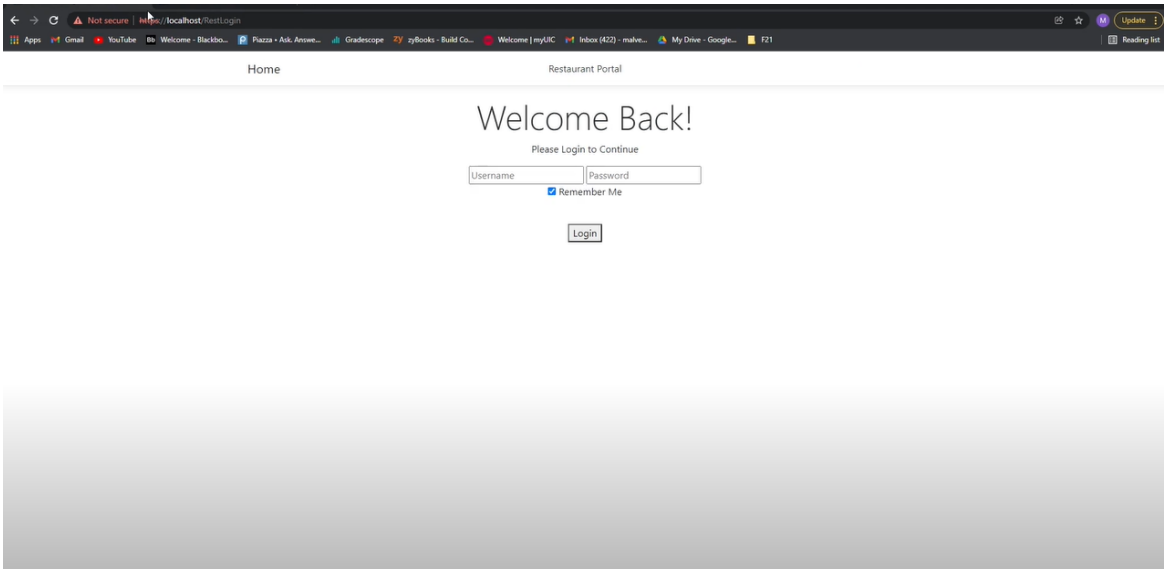


Figure 6 - Restaurant login page

After successful login with valid credentials, the next page holds four buttons: UPDATE, MANAGE, CONFIRM, and HELP.

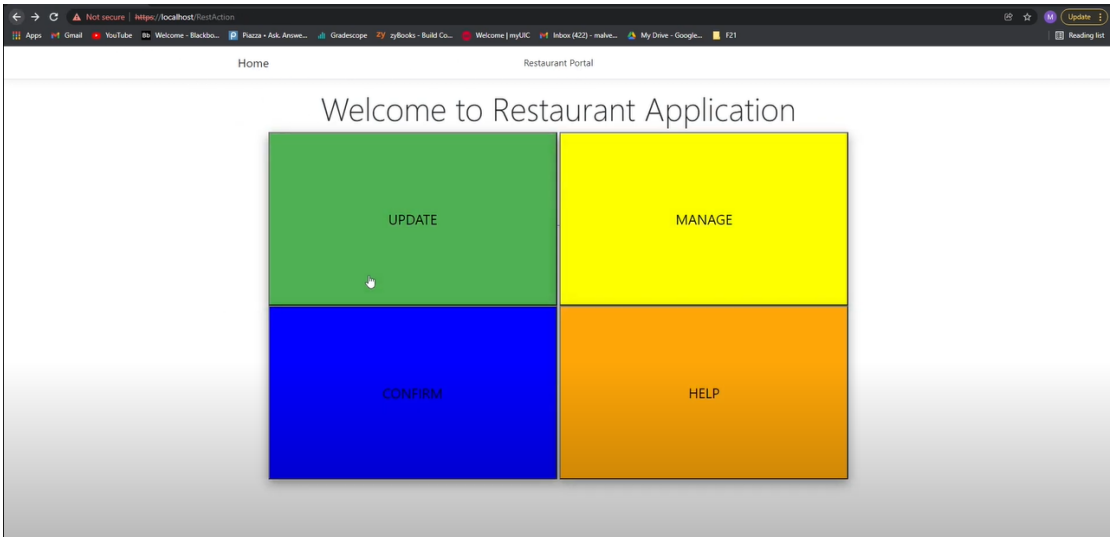


Figure 7 - Restaurant portal home page

On the UPDATE page, the restaurant employee can update the restaurant's contact information, menu, seating, and account settings.

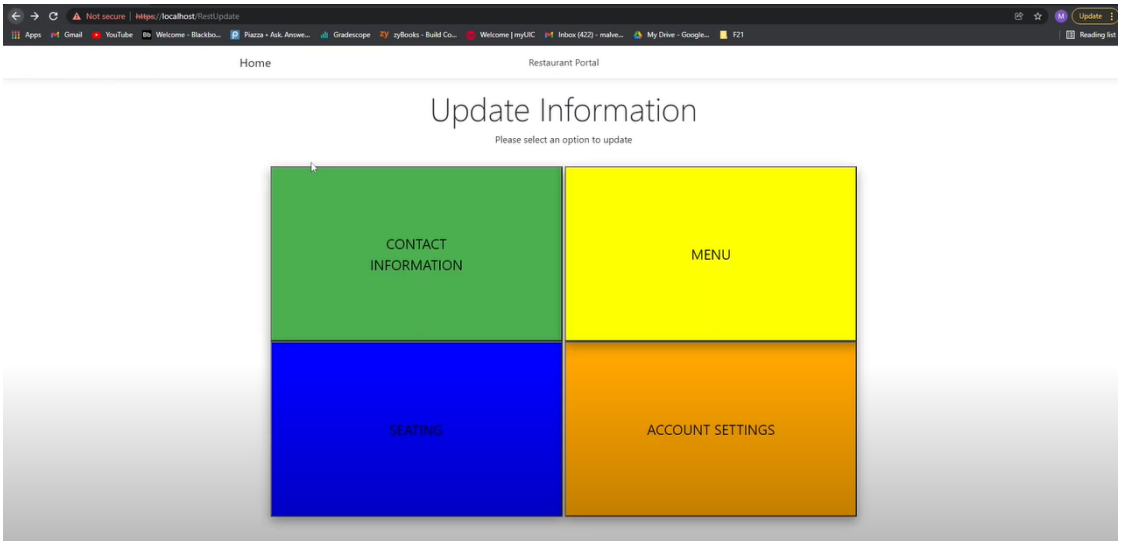


Figure 8 - Update page

On the MANAGE page, the employee can update a reservation or add a new reservation for walk-in customers.

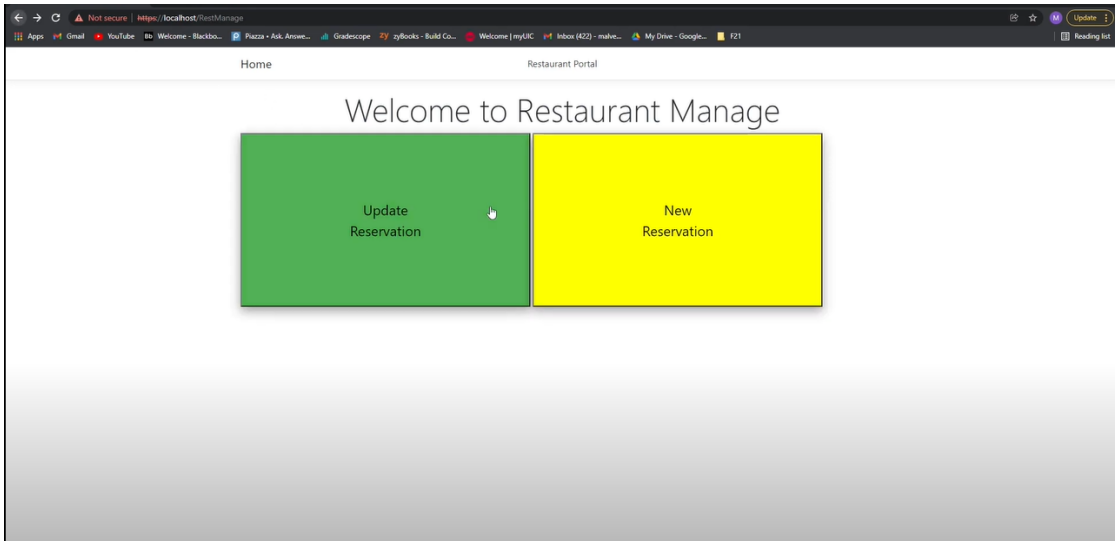


Figure 9 - Manage page

On the CONFIRM page, the employee can confirm the payments of the customers.

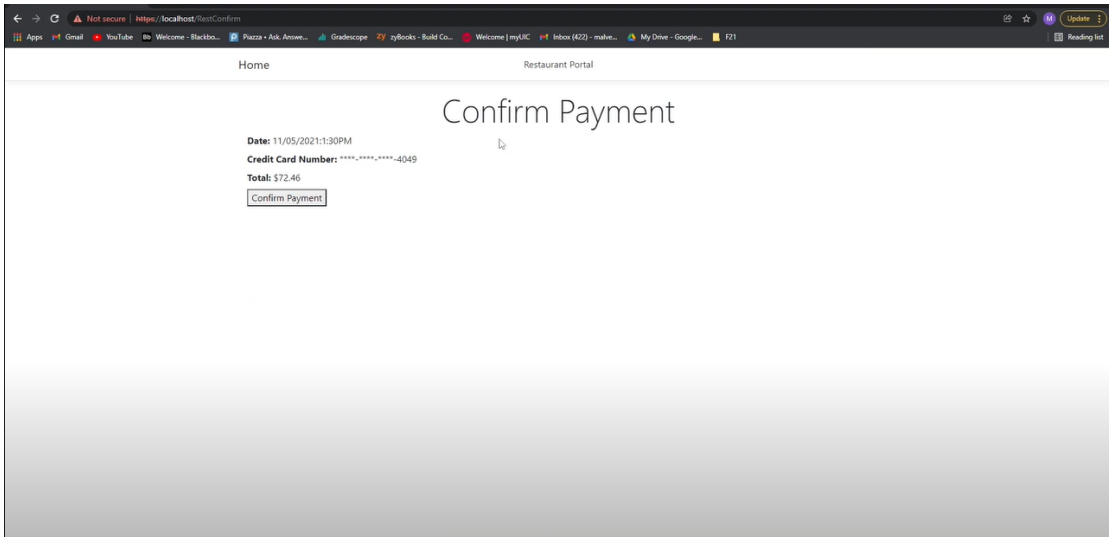


Figure 10 - Confirm payment page

The HELP page shows instructions for using the restaurant portal for anyone who needs help using it.

The major accomplishment of this release along with completing the front-end for the restaurant side of the application, was that we linked a database to our application so we didn't just have the GUI of the application, which was the only thing we had for the first release, we were now able to save the users' selections on the website and see the information updated on MySQL.

7 Comparison with Original Project Design Document

The original project design called for ReserveYourTable to be two separate entities, the first being a phone application for the user side and the restaurant side being a tablet application. However, after we discussed the direction we should head in, we ultimately decided that it would be best to prototype the application as a web app. No one had any experience developing mobile applications, let alone two separate apps. We had some experience developing web apps which would allow us to prototype faster and more efficiently. Additionally, web apps can be used on any device with a browser so we believed that it would be suitable for the project.

III Testing

8 Items to be Tested

1. Thomas Sobczak testing Ryan Thommes's sign up form and page (ID01 - Signing up for an account).
2. Matthew Alvero testing Ramzi Rimawi's add item form and page (ID02 - Adding item to menu).
3. Ryan Thommes testing Matthew Alvero's reservation form and page (ID03 - Making a Reservation).
4. Ramzi Rimawi testing Thomas Sobczak's Order from menu page (ID04 - User orders items from menu)

9 Test Specifications

ID01 - Signing up for an account

Description: The user inputs their personal information to create an account with the *Reserve Your Table* application.

Items covered by this test: The signup page as well as the database table for the user account information.

Requirements addressed by this test: The account must be able to store the information submitted in the form into the database.

Environmental needs: Access to the database system for the program (MySQL).

Intercase Dependencies: N/A

Test Procedures: Go to the sign up page and input the necessary information to sign up for an account and press submit.

Input Specification: A first name, a last name, an email address, and a password.

Output Specifications: The user will now be able to login into the account they created using the email address and password combination they entered into the sign up form.

Pass/Fail Criteria: To pass the test the information inputted into the form must be stored into the database and the user must be able to then login using the account they signed up for.

ID02 - Adding item to menu

Description: Restaurant client users input a new item to be added to a menu to the database.

Items covered by this test: Add item page and form

Requirements addressed by this test: The item name being added must be unique.

Environmental needs: Access to the database system for the program (MySQL).

Intercase Dependencies: N/A

Test Procedures: Navigate to the add item page and input item information to be added.

Input Specification: Item name, Item price.

Output Specifications: When customer users are attempting to order items, the new item shall be listed in the list of items to be ordered..

Pass/Fail Criteria: To pass the test, the new item added should be stored in the database and it should show up in the list of items for the restaurant's menu when ordering.

ID03 - Making a Reservation

Description: The user wants to make a reservation; inputs the restaurant name, number of guests, and reservation time and date.

Items covered by this test: Reservation form

Requirements addressed by this test: The restaurant name should be pulled from the database and the number of guests and date and time needs to be valid.

Environmental needs: Access to the database system for the program (MySQL).

Intercase Dependencies: N/A

Test Procedures: Click the "Reserve Your Table" button on the home page and fill out the reservation form.

Input Specification: Restaurant name, number of guests, time and date

Output Specifications: When the user clicks next it should continue to the next page.

Pass/Fail Criteria: To pass the test, the reservation should be added to the database.

ID04 - User orders items from menu

Description: The user gets to the Order page (**Figure 4**) and selects items from the restaurant menu they want to order.

Items covered by this test: The order page as well as the database table for users' orders.

Requirements addressed by this test: Users must be able to successfully order from the provided menu.

Environmental needs: Access to the database system for the program (MySQL).

Intercase Dependencies: N/A

Test Procedures: On the Order page, select items from the menu and amount of each item to order. Then press the Next button.

Input Specification: Menu item and amount of item to be ordered.

Output Specifications: The user's selections should show up in a query in the connected MySQL server with the items selected and number of each item.

Pass/Fail Criteria: To pass the test the selections that the user makes on the Order page must be stored and show up in the database.

10 Test Results

ID01 - Signing up for an account

Date(s) of Execution: The test was run on November 27, 2021.

Staff conducting tests: Thomas Sobczak

Expected Results: Data inputted into the form be stored in the database.

Actual Results: The data inputted into the form was successfully stored into the database.

Test Status: Pass

ID02 - Adding item to menu

Date(s) of Execution: The test was run on November 27, 2021.

Staff conducting tests: Matthew Alvero

Expected Results: Data inputted into the form be stored in the database.

Actual Results: The data inputted into the form was successfully stored into the database.

Test Status: Pass

ID03 - Making a Reservation

Date(s) of Execution: The test was run on November 27, 2021.

Staff conducting tests: Ryan Thommes

Expected Results: Data inputted into the form be stored in the database.

Actual Results: The data inputted into the form was successfully stored into the database.

Test Status: Pass

ID04 - User orders items from menu

Date(s) of Execution: The test was run on November 27, 2021.

Staff conducting tests: Ramzi Rimawi

Expected Results: Data inputted on the order page will be stored in the database.

Actual Results: The data inputted on the order page was successfully stored into the database.

Test Status: Pass

IV Inspection

12 Items to be Inspected

1. The signup page for users to sign up for an account with their personal information.
2. Adding item page for restaurant side users.
3. Making a reservation
4. Ordering items from menu

13 Inspection Results

Sign up page was inspected by Thomas Sobczak on November 27, 2021. Thomas discovered that the sign up page works and is effective. Some flaws include the code does not check for duplicate accounts therefore accounts can have the same account information.

For the item adding page, it was inspected by Matthew Alvero on November 27, 2021. It was discovered that form was able to communicate with the database to add a new row into the menu database. It was also discovered that duplicate items were allowed to be made, therefore multiple items with the same name could be displayed on the order page for customer client side users.

V Recommendations and Conclusions

Sign up page has been tested and inspected. Next actions include having a more advanced security system in place to stop unauthorized people from accessing other people's accounts.

The add item page has been inspected and tested out. Next steps would be to fix the duplication issue.

VI Project Issues

14 Open Issues

Our investigation into whether the new version of the prototype will be best suitable for a web application or a mobile application is not yet decided. A mobile application and web application would be most beneficial for the user side, but is unclear for the restaurant side. Small restaurants may be using tablets for their infrastructure, but larger restaurants may be using some sort of embedded system. Most restaurants will not want to replace their entire infrastructure so it is important to allow for ReserveYourTable to be integrated into already existing systems.

15 Waiting Room

The menu implementation right now could be improved to show more of the menu at once, alongside pictures of the menu items.

A feature that may not be easy to implement but have a large knock-on effect would be a map of the restaurant's layout. As of right now, everything is form and text based, but a visual map would help streamline both the user and restaurant side of the application. It would allow users to see how the restaurant is oriented and let them see and choose a physical table. Additionally, it allows the restaurant to get an overview of what is happening at each table. On the restaurant side it could show when the next party is scheduled to arrive, which tables need to place an order, be served, or any other pertinent information regarding a specific table.

Another feature to implement would be a single form for users to fill out with dynamic page controlling. The form in question would be the reservation form, as going through multiple pages is complex in delivering multiple amounts of information from one page to another.

16 Ideas for Solutions

After reservations are made by a user, they are stored in the database. However, there are no checks for scheduling or table conflicts.

The framework the application is currently using might not be suitable for data delivery between pages. Completely changing the framework to a more suitable one would make the application much easier to manage for developers to create new content and deliver business side logic.

17 Project Retrospective

Some things that went well was the full front end development for the project. The team was able to complete the basic front end development within a few weeks with minor tweaking after completion. However, the same cannot be said for the back end implementation. With the development team on different operating systems, finding a framework that would work well for all team members was difficult and as a result, we chose a framework that in the end was not suitable for the application in terms of developing purposes. The only tools that the team investigated on was Entity Framework Core as it was the most used in projects using .NET Core 3.1. If the team was able to use an easier framework than EF Core, the back end implementation would have been done more quickly and efficiently.

VII Glossary

Database: a data structure that stores information in a computer and is accessible in many different ways.

AWS: Amazon Web Services

ASAP.NET: An open-source, server-side web-application framework designed for web development to produce dynamic web pages. It was developed by Microsoft to allow programmers to build dynamic web sites, applications and services. [2]

VIII References / Bibliography

[1] J. Vazquez, N. Hoa Pham, G. Kaur, J. Singh, Requirement Document for Reserve Your Table, Spring 2021.

[2] “ASP.NET.” Wikipedia, Wikimedia Foundation, 17 Nov. 2021, https://en.wikipedia.org/wiki/ASP.NET.