

Reserve Your Table Coding Project Summary

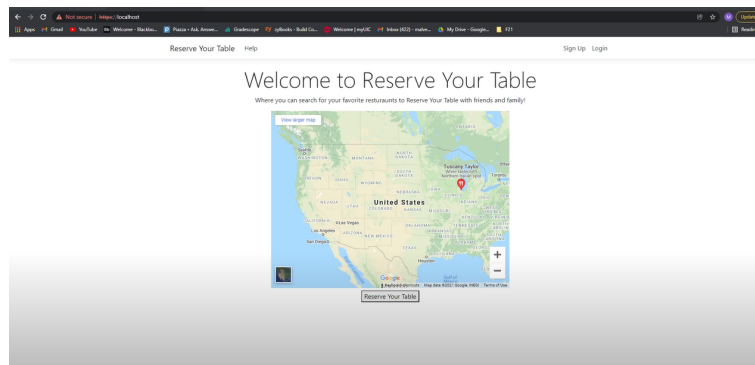
Group 6: Thomas Sobczak, Ryan Thommes, Ramzi Rimawi, Matthew Alvero

The *Reserve Your Table* project was originally thought out and developed during the COVID-19 pandemic era. The application is designed to help people dine into their favorite restaurants without having to worry about unknown interactions. During these especially hard times, it is important to keep interactions with other people to the minimum by having all those interactions done virtually through the application itself. The application was first proposed to limit virus transmission. However, the ability to reserve, order, and pay all on one system over the internet is a great and convenient way for people to ease their way back into dining in-person.

The *Reserve Your Table* application prototype was built using ASP.NET Core 3.1 with a MYSQL backend being hosted using AWS. The original application was designed by J. Vazquez, N. Hoa Pham, G. Kaur, J. Singh of Group 20 from the Spring 2021 semester. They stated that the application be built on the mobile development platform however we took the approach of developing it as a web application for the prototype build. We as a group found that using a web development approach rather than a mobile development one would be more beneficial in developmental efficiency and testing purposes as we were all familiar with some basic front end skills.

Based on the previous group's development requirements and diagrams, our group completed a working prototype very similar to what the previous group envisioned. We developed a customer-sided interface as well as a restaurant-sided interface which we called the "Restaurant Portal". On the customer side of the application we included an embedded map on the homepage to search for restaurants, then a page for users to reserve a table at the restaurant, then a page to order items from the provided menu, and lastly a page for the user to enter payment information to complete their order. On the restaurant side of the application we created a restaurant login page, then after logging in we created 4 buttons that take you to UPDATE, MANAGE, CONFIRM, and HELP pages. On the UPDATE page, the restaurant employees can update the restaurant's contact information, menu, seating, and account settings. On the MANAGE page, the employees can update a reservation or add a new reservation for walk-in customers. On the CONFIRM page, the employees can confirm the customers' payments. On the

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



For the back-end of our prototype we decided to use a MySQL database hosted through AWS. From there we connected the database to the application using Microsoft Entity Framework Core as well as the Pomelo MySQL package for Entity Framework Core. We used Entity Framework Core because it was the only package that supported .NET Core 3.1. Within the application code itself, we made classes for each of the tables that we wanted to make. We went the extra step and configured each class and property of the class to use Entity Framework Fluent API.

In our current implementation of the project, we tested out the full stack functionality of the application. Specifically, we tested out the applications ability to fetch and receive data from the database. The pages we tested out were the sign up page, add item page, reservation page and the order page. In our testing, we have concluded that all of our test requirements were met. However, after testing we discovered that there were some flaws that each page had. We concluded that the flaws needed to be fixed in the final release of the application.

During the development of the project, we ran into some issues on the project as a whole. The approach that we had in implementing the application was for the application to be a web application instead of a mobile application which is what the previous group originally planned for. We did this in order to best utilize each member's skills in basic front end development and back end development. The web application approach was also used because we believed that users on the customer side of the application would benefit with an application that would be dynamic with the interface of a user's device. However, we determined that the question still remained of whether the restaurant users would benefit from this approach.