University of Arkansas – CSCE Department
Capstone II – Final Report – Spring 2021

Rezerve

Brayden Alvarez, Dalynn Hatch, Landry Ishimwe Karangwa,

Rylan McAlister, Tyler Gerth, Colton Tucker

Abstract

Due to the COVID-19 pandemic, many smaller barbershops and salons were hurt tremendously due to restrictions preventing customers from doing in-person business with them. This caused some of these shops to go out of business entirely and caused others great financial loss. Our application, Rezerve, aims to solve the problem of these barbershops reopening and trying to regain customers in a post-pandemic society. We are doing this by partnering with these businesses and creating a hub in which customers can search their local area for barbershops and salons.

Since barbershops and salons are some of the most personal of customer service businesses, we want to capitalize on that by creating a web application that allows customers to find and make relationships with the specific people working at these businesses rather than just seeing a business name. Rather than simply being an appointment-making service for the business, Rezerve aims to allow customers to browse various barbershops and salons and see and learn about the people working there. That way, when a customer shows up for an appointment, it isn’t a brand new experience, but instead already somewhat familiar. Creating these meaningful relationships will allow customers to put a name to the face of the person cutting their hair and make them more likely to want to return in the future or recommend the environment of the shop to a friend, thus creating more business.

1.0 Problem

The first facet of the problem is that, due to the outbreak of coronavirus, many businesses have had to shut their doors for a while. Now that restrictions are slowly being lifted, many smaller businesses are having trouble keeping their doors open due to lack of customers. This is especially relevant with businesses that handle hair care, since hair care has both the problem of being an industry in which you cannot social distance effectively and one that is often not a large company conglomerate, instead being smaller self-owned shops. When the pandemic winds to a close and society starts to regain some sense of normalcy, barbershops and salons will likely have a hard time recovering after exhausting their financial stores during the time of the virus. If there is no solution found for this problem, then, as these businesses reopen, they will likely have difficulty regaining their clientele, as well as finding new clients.

The second facet comes from the fact that barbershops and salons are very personal businesses and need a way to leverage that. When most people enter a salon for the first time, they may not know what to expect or have any idea how they want their hair cut or styled. This often leads to dissatisfaction with the level of service provided to them, because it ended up being different from what they expected. Because of this, barbershops and salons, and more importantly the employees of these businesses, need a reliable way to directly interact with and talk to potential new customers beforehand in addition to just scheduling appointments.

As stated before, with coronavirus, many small local businesses have been having hard times trying to get people in their doors and spending money. Many have gone under because they did not get enough exposure or traffic from customers. Without this application, many of these types of businesses will continue to go under or be forced to shut down locations in more towns and cities.

2.0 Objective

The objective of this project was to design and implement an application that would bolster a multitude of features that seamlessly connect people to barbershops and salons in their area. The application features a two-pronged approach with functionality for both businesses and customers to display important information to each other. The primary focus was to create an application that serves as a fast and fluid method for potential customers to get in touch with barbershops and salons.

3.0 Background

3.1 Key Concepts

In order to solve this problem, we leveraged several different technologies. The first was React, a frontend framework used to design the view of our application. React utilizes a combination of JavaScript, HTML, and CSS in order to provide a user interface (UI) that both users and businesses using our product can interact with. It allowed us as developers to include outside libraries, such as Firebase, Stripe, Google Maps, and styling components that were used throughout the application as a whole.

The second main technology used in development was Firebase. Firebase is a service provided by Google that allows developers to create databases to store application information, authenticate users who are using the application, and provide notifications and messaging services. It also provides a dashboard where developers are able to monitor the application as a whole and see things like how many users are currently using it and the space available in the database.

The third main technology being used was Stripe. Stripe is a secure and reputable payments service that provides an API for developers to use. This way, developers do not have to worry about dealing with the storage of personally identifiable information such as credit card numbers, instead just passing it on to Stripe. We implemented this API for the Payments section of our project, where users can schedule and pay for appointments with various barbershops and salons.

The fourth main technology we utilized was the Google Maps API. This allowed us to access real-time geolocation data, so that customers could search for businesses within a selected address or city, and businesses could encode their business location into the database when signing up.

The last main technology we utilized was Heroku. Heroku is a service used for deploying applications. It is a way that developers can put their application onto a server so that it can be accessed at any time from any user. Instead of having to deal with hosting the server ourselves and all the configuration needed for that, Heroku provides a pre-existing computing network to which our application can be added.

3.2 Related Work

Currently, there are already several applications out there allowing customers to manage their hair care appointments and search for new businesses. theCut[1] and The Barber Post[2] are apps which already have much of the same functionality that we plan to develop, such as searching for locations via Google Maps and setting up appointments with barbers and salonists. Another application, Taper[3] allows businesses to build out their own pages for clients to discover when searching. Our goal was to build something similar to these applications, and then add additional functionality that sets Rezerve apart. One of the most important things we added was the ability for customers to view employees and those employees' services directly from the app, so they could see exactly what they would be getting out of their appointment. Also, we added the ability for customers to directly interact over messaging with employees at barbershops and salons, to create more personal relationships before they even walk in the door. On the business side, the important features were the ability to view holistic business performance data to determine what needed to be improved about the business as a whole.

This was done by adding in features like direct messaging between clients and businesses, a business info page for selecting employees and their services, and a data analytics section for business owners to determine how well their shop is doing and what needs to be improved. Besides these added features, we still included things like "Check Out as Guest" and setting reviews for businesses, to make sure that customers had the same functionality that they would expect from other similar applications.

As we developed, we worked closely with our product champion to, who communicated with the businesses we planned on releasing this application to, to get further feedback and ideas about the kinds of features they like and what can be improved. This way, the functionality was able to be adapted to cover usability concerns and requests that arose. This is something else that sets our application apart from the other applications currently out there. Since these customers and businesses have used other similar services in the past, we were able to adapt Rezerve to address any concerns they had with other applications.

4.0 Design

4.1 Requirements and/or Use Cases and/or Design Goals

- The app allows customers and employees to create an account and have their past appointment and user data be stored and accesible
- The app allows both guests and customers to search for barbershops and salons on a map, either in their area or areas they will be visiting
- Customers are able to research particular barbershops.
- The app allows customers to schedule an appointment at either a given date and time or the first available date and time for an employee.
- The app enables customers to pay the barber within the app.
- The app allows barbers to interact with customers through messages.
- The app allows employees of businesses to manage their own pages, setting items like availability and services offered, and interact with potential and existing clients through messages.
- The app allows businesses to view information about their number of customers and profits based on a time period (Day, Week, Month, Year)
- This app allows new users to create a new business at a valid location, setting up their account to be visible to customers

4.2 Detailed Architecture

The main flow of this application has two parts - one for businesses using the applications and one for customers using the application. Both of these flows were built using the React framework. Users will be navigated to one or the other upon logging in to the application.

Customer Flow:

Starting with the customer flow, when a user directs to the website they will be greeted by the home page (Figure 1). This page contains several tabs at the top that allow customers to filter by the type of business they are looking for. They can then search by location and be presented with a list of businesses within a general area of that location. When clicking on a business they will see more detailed information such as the address, a general description, and reviews left by other customers (Figure 2). Then they will be able to click on an employee and see a list of the services they offer. If one of those services is what they are looking for, they can select an appointment time, pay, and request the appointment to be booked (Figure 3). All of these features are also offered if you are only accessing the site as a guest, but if you actually create a user account you will be able to access many more features. These include the ability to view and manage your upcoming, requested, and past appointments (Figure 4). The ability to send and receive messages from your stylist, and the ability to leave reviews for a business upon completion of an appointment (Figure 5).

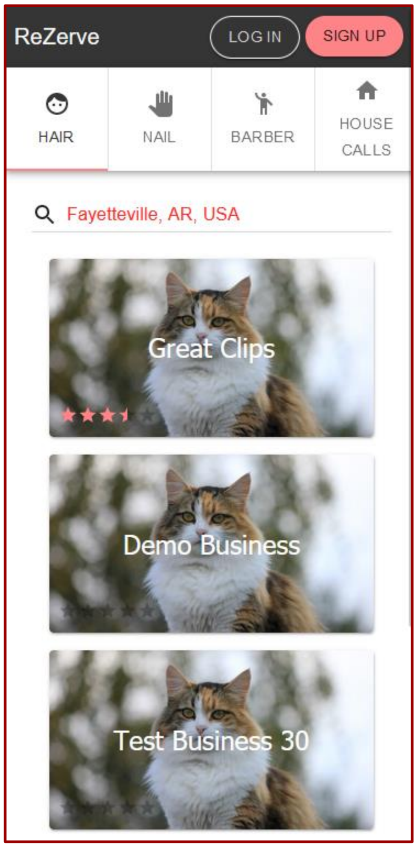
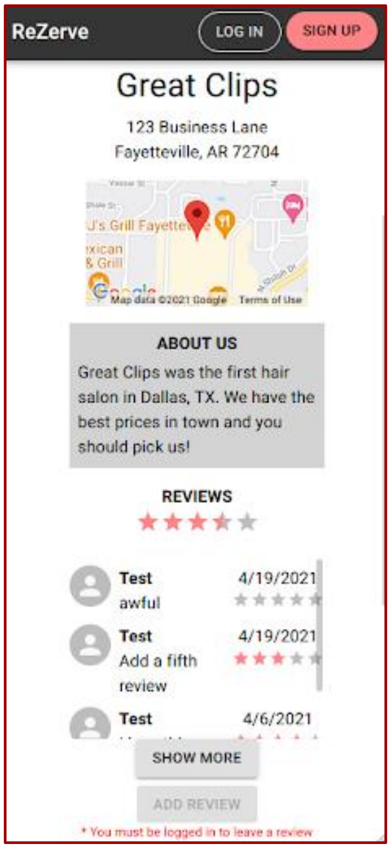


Figure 1



123 Business Lane
Fayetteville, AR 72704

ABOUT US

Great Clips was the first hair salon in Dallas, TX. We have the best prices in town and you should pick us!

REVIEWS

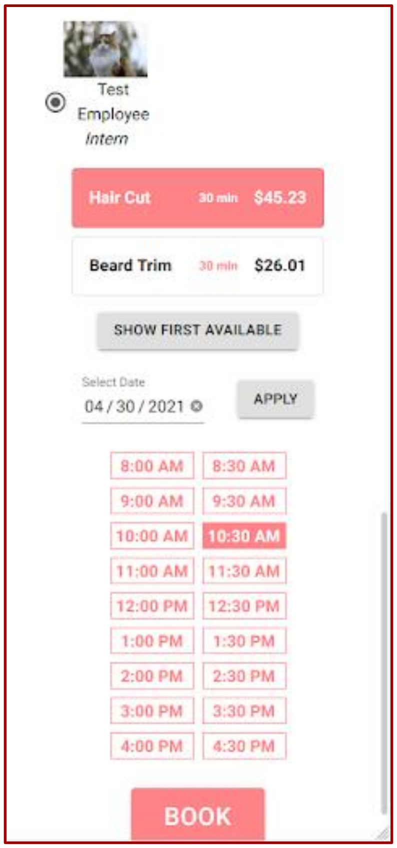
Test 4/19/2021
awful

Test 4/19/2021
Add a fifth review

Test 4/6/2021

* You must be logged in to leave a review

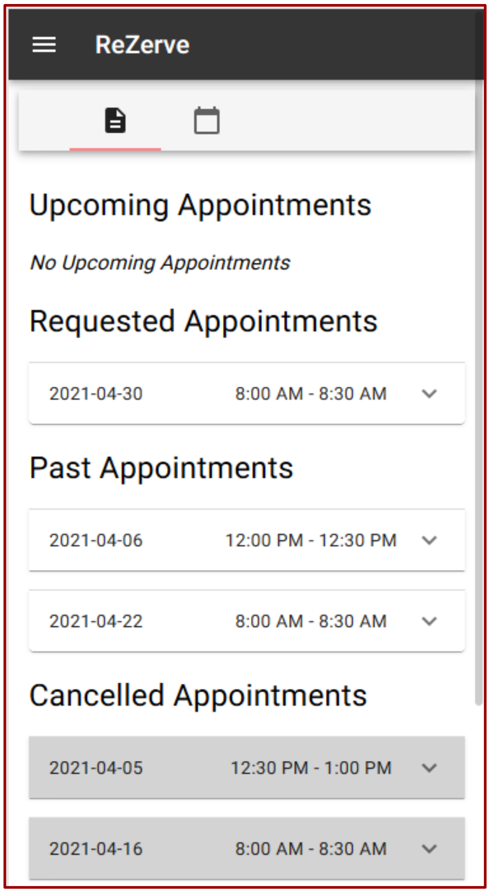
Figure 2



Hair Cut	30 min	$45.23
Beard Trim	30 min	$26.01

Select Date
04 / 30 / 2021

Figure 3



Upcoming Appointments

No Upcoming Appointments

Requested Appointments

2021-04-30 8:00 AM - 8:30 AM

Past Appointments

2021-04-06 12:00 PM - 12:30 PM

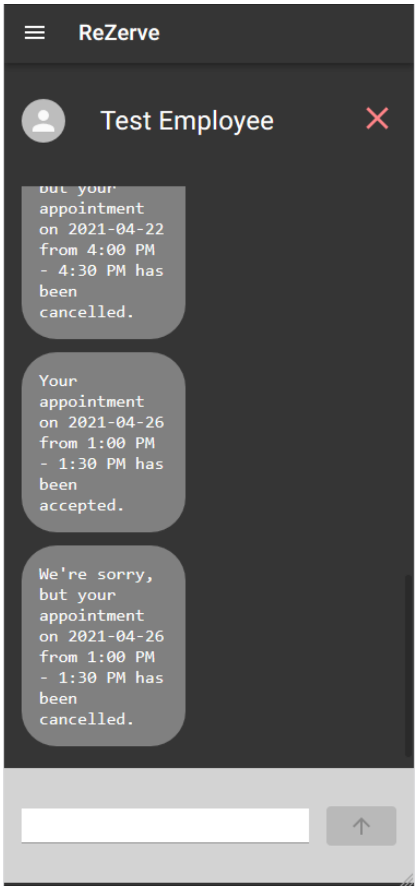
2021-04-22 8:00 AM - 8:30 AM

Cancelled Appointments

2021-04-05 12:30 PM - 1:00 PM

2021-04-16 8:00 AM - 8:30 AM

Figure 4



but your appointment on 2021-04-22 from 4:00 PM - 4:30 PM has been cancelled.

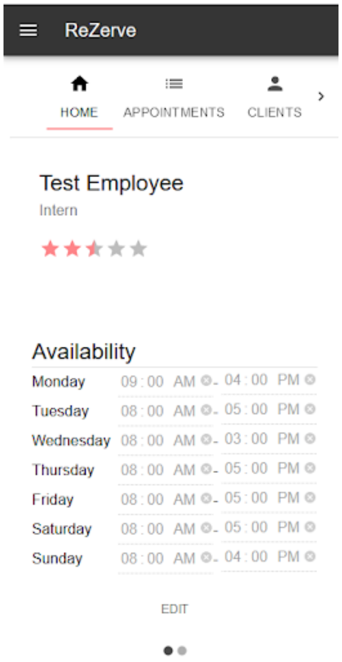
Your appointment on 2021-04-26 from 1:00 PM - 1:30 PM has been accepted.

We're sorry, but your appointment on 2021-04-26 from 1:00 PM - 1:30 PM has been cancelled.

Figure 5

<u>Business Flow:</u>

On the business side of things, the first step will be for the owner to create a user account, and then create a business profile that's attached to their account. They will then have two different pages they can access, that normal customers can't: Employee Home and Business Home. The employee home page is what all employee type users will be directed to after logging in. This page will provide employees with a place to view and edit their information like name, position, availability, services offered, etc (Figure 6). Here they can also view and manage their appointments and view a calendar view of what their upcoming day or week looks like (Figure 7). Also they have the ability to see a list of their current clients and how many times they have visited for an appointment. On the Business Home side of things, if the currently logged in employee is the owner, they will be able to see and edit all of the business information displayed to customers. Then they will be able to see an overview of the business' employees and a performance page. This performance page will track things such as revenue, number of appointments, and the rating and reviews of the business (Figure 8).



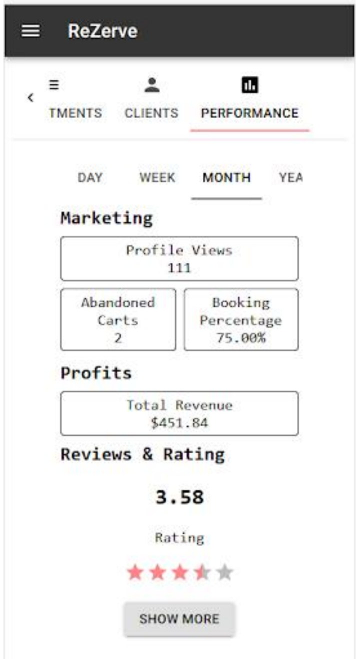
Test Employee

Intern

Availability

Monday	09 : 00 AM - 04 : 00 PM
Tuesday	08 : 00 AM - 05 : 00 PM
Wednesday	08 : 00 AM - 03 : 00 PM
Thursday	08 : 00 AM - 05 : 00 PM
Friday	08 : 00 AM - 05 : 00 PM
Saturday	08 : 00 AM - 05 : 00 PM
Sunday	08 : 00 AM - 04 : 00 PM

Figure 6



Marketing

Profile Views 111

Abandoned Carts 2

Booking Percentage 75.00%

Profits

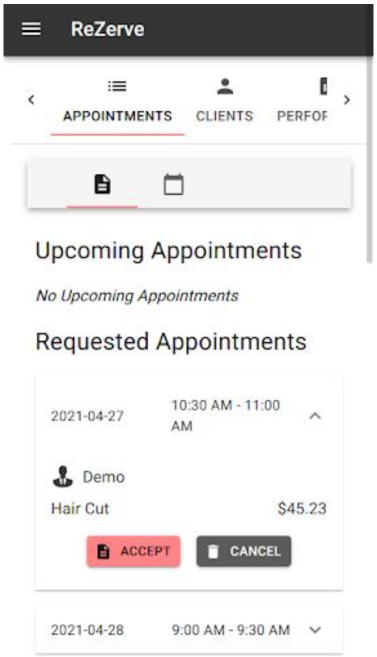
Total Revenue $451.84

Reviews & Rating

3.58

Rating

Figure 7



Upcoming Appointments

No Upcoming Appointments

Requested Appointments

2021-04-27 10:30 AM - 11:00 AM

Demo

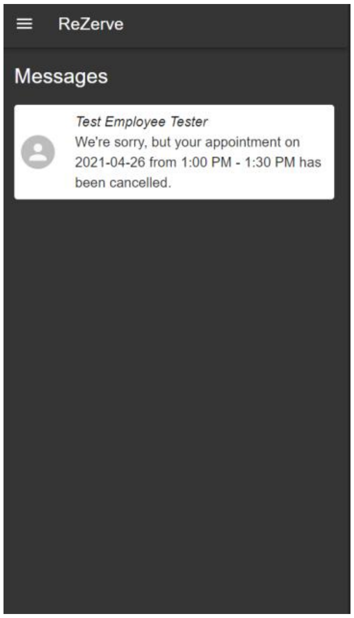
Hair Cut $45.23

2021-04-28 9:00 AM - 9:30 AM

Figure 8

Other Components:

Outside of the two main flows, there are several other usable components throughout the application. This includes things like a Messaging screen (Figure 9), which can be used by both employees and customers to message each other, a Profile page (Figure 10), where users can set their basic profile information, and a sidebar (Figure 11), a component available to logged in users where both of the previously mentioned components are contained. Finally, there is the server. The server code was contained in the same repository as the user interface, but deployed separately, and handles passing Stripe payments from the user interface to the Stripe API, making sure to take out a cut of any payments as the business fee for using the Rezerve application.

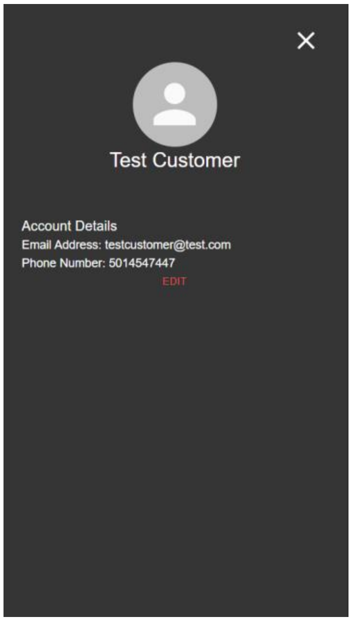


Messages

Test Employee Tester

We're sorry, but your appointment on 2021-04-26 from 1:00 PM - 1:30 PM has been cancelled.

Figure 9



Account Details

Email Address: testcustomer@test.com

Phone Number: 5014547447

Figure 10

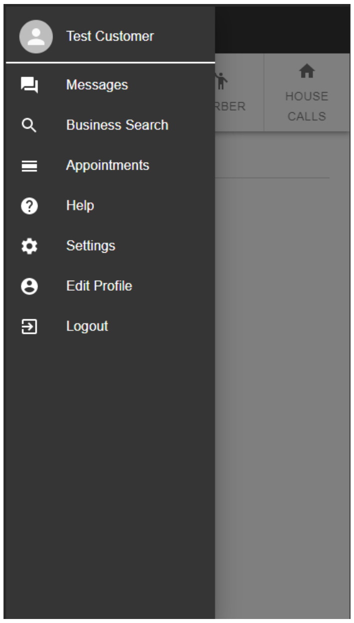


Figure 11

<u>Technologies Used:</u>

We primarily used React as the framework for our development, using it to design and develop our main user interface (the flows for which are shown above). Within this code, we included the Stripe [4], Firebase [5], and Google Maps APIs for many of the core functionality within our application. These included the various tasks of user payments, database storage and user authentication, and accessing geo-location data. For the server side of processing Stripe payments and handling fees, we created a NodeJS server within the application. Once the code was developed, we used Heroku as a deployment platform, which allowed us to deploy the user interface and server separately, as well as contain secret keys that we didn't want stored within the application code.

Within the React application, the main libraries we used were React Redux, as a way to essentially create an in-memory database that managed state throughout the application, and Material UI, a styling library that provided us with common components to use across throughout the code and color theming to help the application have a consistent look and feel. For a deeper dive into these technologies and APIs, feel free to read our Software Justification Document and API Document, included alongside this report.

<u>Lessons Learned:</u>

Throughout the development of this application, our development team learned a lot of lessons related to creating an application for use in a business. One of those lessons was learning how to estimate task length and what can reasonably be accomplished in a given time period. Since this application was a lot larger in scope, and needed to be more complex due to the inherent risk of dealing with personal user data, than anything we had worked on before, we initially thought some tasks would take less time than they actually ended up taking. This pushed our development timeline back and forced many of us to work long hours to make sure that everything we were trying to accomplish got completed. Everything from Firebase storage, to user authentication, to payment access had to be carefully managed and accessed, and many of these functionalities were intertwined, which leads to the next point - working alongside other developers.

All of us had worked in group projects before, but generally, the code for those projects could be developed separately and brought together without too much hassle. However, for a project of this magnitude, where one person might be working on accessing customer data, another working with employee appointments, and a third developing the sign-in process, they would all require some form of user authentication, which might be being developed by a fourth person entirely. And, since none of us were working on this project full time, instead having either work or school as another priority, that meant that peoples' schedules were out of sync and one might find themselves sitting around waiting on another's code to be done. That forced us to communicate a lot more and, throughout the semester, caused us to start doing a form of standups halfway through each week, where we would update each other on our progress in order to have some visibility on what was getting done.

A third lesson learned was to make sure and have standards set up to do things correctly from the start. Since none of us had done much work with React before when we initially started development, we each had different styles of coding, creating and naming components, and accessing styling. This caused some conflict when trying to merge code together, as it was difficult to understand fully what other peoples' code was doing, when it was written so differently from our own. This brought upon a large refactor during our second sprint, whereupon we set out styling standards, chose to use TypeScript across the application as a whole for consistency and the typing it provides (to better describe what the code is doing), and also set up React Redux as a way to manage application-wide state.

Future Work:

While we were able to accomplish a large majority of the goals we initially committed to, there was still some work left over that we will continue to work on in the coming weeks in order to have the application fully ready for our product champion and the businesses that will be using it. First and foremost is the cleaning up of existing code. Throughout the development process, we moved quickly when creating and implementing new features, but as always, there were solutions put in place that were not the most efficient or could be improved upon. Before this application is fully ready to go into production, several pieces need to be fixed. This includes making the Sign Up process a bit cleaner, as well as including more error handling to make sure that the user does not input faulty location data. Additionally, the Payments process needs to be fleshed out a little bit more to verify that money is being handled correctly and safely (throughout development, we have been using a test API key to prevent any issues with real money).

More future work would include the implementation of additional features. Although we got the core features added, there are still additional ones that it would be nice to add. The first feature would be filtering between different types of businesses when the customer is searching (something that is already in place in the user interface, but not quite implemented in our database and code yet). This would allow the user to search specifically for nail salons, barbershops, or other types of stylists, without having to scroll through a list and view each one to determine what kind they were. Another needed feature would be a way to tip employees after completing an appointment with them. Currently, payment only occurs on the initial booking of appointments, but never after that. At most hair care businesses, customers usually have the option to pay an additional tip to their stylist based on how well of a job they did. Ideally, this would be able to occur in our application, but it was outside of our original scope, so we weren't able to get to it. Finally, we were not able to get to the point of allowing new businesses and users to save a profile picture upon creating an account. It was something we looked into, but realized we would not have the time as it would probably require setting up a content distribution network or another database to store image files. None of the group members had experience with this, so we agreed to look into it more in the future before the project gets released.

Potential Impact:

______While the pandemic is winding down as more and more people get access to the vaccination, Rezerve could still see a great impact on the salon and barber industry, especially for smaller businesses. The suite of features that the Rezerve application offers will allow customers to more easily find and get in touch with hair stylists in ways they previously couldn't, as well as provide them the ability to communicate with their stylist before or after appointments and improve that relationship overall. A lot of the impact will be seen when looking at the business side, as managing a business can be a lot of work. By providing a fluid interface that lets business owners manage their employees, business information, view performance statistics, and more, Rezerve can directly impact how that business is doing. Employees will be able to swiftly see their upcoming appointments and perform actions like cancel them or message the customer beforehand to provide information about their appointment.

The impact of the application can also grow as Rezerve gets more traction and customers or businesses are able to start providing feedback on features they want to see added or changed within the application. This includes some of the features in the aforementioned "Future Work" section. The goal of Rezerve is to positively impact the sometimes daunting task of finding a barbershop or salon and booking an appointment. On the flip side, it also provides a reliable and easy to use interface that employees and business owners can use to improve the quality of life of their work.

4.3 Risks

Risk	Risk Reduction
User payment data storage	Used a reputable outside payment company (Stripe)
User personal information storage	Using Firebase, a product made by Google, that provides authentication

4.4 Tasks

Note: Apart from our Capstone team working on this application, there was one other developer working on it who was a colleague of our Champion. They had some tasks, mainly around routing between components and initial styling of different pages, but their overall role was greatly diminished. Their work will not be included in the Task List, as this list is focused on the work the group members completed.

Overall, our application development was split up into two sections, the fall and the spring. In the fall, we worked with our product owner to start gaining an understanding of the project and the technologies with which we would be using. Then, we started building out basic pages for the application. After seeing the structure of the basic application we decided to discuss making some user experience changes with our product owner. Following this discussion, we began working on fully fleshing out mocks (to be used more in the spring). This way, we as developers had a better idea of what we needed to build and the data we needed to be taking into account.

In the spring, we started focusing on creating the pages to look like the new fully fleshed out mocks. We had a big project rework meeting where we changed the project to use TypeScript along with React Redux. Following this, we also decided on a ruleset for future work so that we would have one style for the whole project. Then we spent the most time focusing on getting all the different functionalities required for the business and customer flow.

Fall Task List:

- Read documentation on the React framework and how it works (all members)
- Set up example React project to understand the flow of the React framework (all members)
- Design of the Business Info page
    - Create the HTML layout of the page
        - Needs to be able to see information about the business, such as name, contact info, address, and workers
    - Load mock data into the page
    - Connect the Business Info page to the rest of the application
- Design of the Customer Appointments page within the dashboard
    - Create the HTML layout of the page
        - Need to be able to see past and future appointments a customer has scheduled with a business, including name of business, date time, and cost
    - Load mock data into the page
    - Connect Customer Appointments page to the rest of the application
- Creation of a Firebase project and connection of it to our React project
    - Create a project in Firebase for the application and add the necessary library configurations to connect our application to its API
- Creation of a NoSql Firestore database within Firebase
    - Create a database within Firebase that will contain all the customer and business data for our application
    - Load mock data into the database
    - Connect the database to our application so the mock data can be pulled in

- Design of the Payments checkout page
    - Create the HTML layout of the page
        - Use Stripe API to take care of payments
        - Need a customer to be able to enter their card information and other needed Stripe components so they can schedule appointments
    - Verify that the checkout page successfully links to Stripe
    - Connect the Payments checkout page to the rest of the application
- Design nearby businesses page
    - Create HTML layout of page
        - Using Google Maps API to handle location tracking
        - Ability to search for business based off user address and built in location services
    - Set up appointments by clicking on a marker placed on business's location

Spring Task List:

- Creation of the Employee Home Page
    - This page serves as the homepage for Employees when they login to the application
    - This page includes a toolbar with the following tabs:
        - **Home -** This tab contains all of the basic information for the employee to see/edit, their availability, which can be set, and the ability to add/remove offered services
        - **Appointments** - This tab contains two-part approach to displaying the schedule and appointment information for the employee. The first part will be a list view of their appointments, and the second will be a more detailed calendar view of their entire day.
            - Appointments show as upcoming, requested, cancelled, and past, with the employee being able to accept or deny requested appointments, or cancel upcoming. Employees can also use this screen to message customers
        - **Clients -** This tab contains a compiled list of the employees clients, with a number indicating how many times they have visited. The list also includes the ability to message one or multiple of the clients
        - **Performance -** This tab contains an assortment of performance statistics such as their sales, profile popularity, and ratings/reviews.
- Creation of Home Page
    - This page serves as the default page when the user access the web app and also gets displayed when the customer login to the web app
- Creation of Login and Sign In pages
    - These pages allow the user to log in with their accounts and provide them with the option to reset their passwords.

- Creation of Sign Up pages
  - These pages allow the customers, employees and business owners to create accounts on the website to have more access.
    - The employee signup allows the employees to create an account and link it to the salons.
    - The customer signup provides the customer with more access to the website; access like making reservations, rating barbers, and barbershops.
- Redesign of the Business Info Page
  - This page serves as the screen that clients interact with when trying to select the barbershop/salon that they want to make an appointment with
  - It shows the location of the establishment, a description of the business, and a list of the employees
  - We remade this screen from the Fall in order to have a more consistent style and feel across the application
- Creation of the Book Appointment Page
  - This page allows customers to select an employee at a business and book an appointment with them (embedded within the Business Info page)
  - It displays information about the employee’s services and availability
  - User can select employee, service, and date and see available appointment times on that day (taking into account employee’s schedule and working hours)
  - Can also choose to find first available appointment for a specific employee and service
- Creation of the Business Performance Page
  - This page allows shop owners to view metrics about their business, such as profits, number of abandoned carts, and overall business rating
  - It is only accessible by the business owner
  - This page will filter the data based on a time period that is selected by the owner and contains the options (Day, Week, Month, Year)
- Redesign Backend
  - This task was over reworking our Firestore database to be more efficient and without duplicated data
    - This was done by building out a DB schema, replacing the existing schema with it, and then adding data into the new schema
  - This task also involved enabling Authentication for the application so users could start making accounts and moving through the application flow
    - Authentication should be persistent until the user chooses to log out, so a user could close the browser and open it back up without being logged out
- Creation of the Employee Accounts Page
  - This page will be accessible by employees of businesses
  - This will allow employees to set their appointments, take payments for the business, as well as have a to-do list of what they need to get done (such as things like getting new equipment, learning new styles, etc.)
- Creation of the Employee Services Page
  - This page is part of the Employee Home Page

  - Shows information about the specific services an employee offers (since different employees specialize in different things), a description of those services, and the price of the service
- Creation of the Business Search Page
  - Allows a user to search for a location via Google Maps API and find all businesses in our database within a 50 mile radius of that spot
  - Uses geolocations to create a bounding box within which to search
  - Displays all the businesses, with links to take the customer to their Business Info page
- Creation of Customer Rating and Review Page
  - This page is accessible by customers and is contained within the customer's Business Info page
  - This will allow customers to rate and review the shops they have been to
  - These ratings/reviews will be posted on the businesses page so others can see them
  - Ratings/Reviews will be stored in a DB to be pulled in when people view the list of businesses
  - Guests without an account cannot create a review
- Account Registration Flow
  - This includes the implementation of the flow for Customers and Businesses to sign up and create accounts on the website
  - The first half is the ability to create user accounts which we are using Firebase authentication to manage, and the second part is creating business accounts which employees are tied to
- Messaging Between Employees and Customers
  - Accessible by both
  - Allows the two parties to send direct messages to each other regarding appointments or general questions
  - Can be accessed via a Messaging tab in the Sidebar or from the Appointments page
- Stripe Integration
  - Accessible by customers to allow for payment of bookings/appointments
  - Server created to handle talking with the application and Stripe API
  - Allow splitting up payments to direct money to the businesses, stripe, and us.
- Creation of the Forum Page
  - Used to allow customers to directly interact and message with each other
  - Allows customers to talk about different businesses they've visited
  - *Deprecated from Project Proposal due to losing members and being unable to complete - Customer/Employee Messaging still in place*
- Deploy the Application via Heroku
  - We set up a Heroku account for the application and did the configuration in order to hook our Github repository to Heroku
  - Deployed the application and started load testing it to make sure it can handle multiple users accessing it at once

- Testing
    - We continuously tested our application throughout the development process to make sure that later changes aren't breaking earlier ones

4.5 Schedule

<u>Fall Tasks:</u>

- Week 1 (10/19 - 10/25)
    - Read documentation on React and how it works
        - Everyone
    - Set up example React projects to understand the flow
        - Everyone
    - Design of the Business Info page
        - Tyler Gerth and Landry Ishimwe Karangwa
- Week 2 (10/26 - 11/1)
    - Creation of the Firebase project and connection of it to the React project
        - Everyone
    - Creation of the Firestore Database
        - Tyler Gerth
- Week 3 (11/2 - 11/15)
    - Design of the Payments Checkout page
    - Design of the Nearby Business page

<u>Spring Tasks:</u>

- Sprint 1 (01/11 - 01/24)
    - Redesign Business Info Page
        - Tyler Gerth
    - Create Book Appointments Page
        - Dalynn Hatch, Brayden Alvarez
    - Create Employee Home Page - Home Tab
        - Colton Tucker, Rylan McAlister
    - Create Business SignUp and Business Account info
        - Landry Emery Ishimwe Karangwa
- Sprint 2 (01/25 - 02/07)
    - Updating the Project to use Typescript, Redux
        - Tyler Gerth, Dalynn Hatch, Colton Tucker, Rylan McAlister
    - Building out Dev Rules to Follow
        - Everyone
    - Create Employee Performance Page
        - Display reviews, abandoned carts, and star rating
            - Tyler Gerth
        - Dalynn Hatch
        - Brayden Alvarez

  - Employee Home Page
    - Colton Tucker - Clients Tab
    - Rylan McAlister - Appointments/Calendar Tab
    - Tyler Gerth - Performance Tab

  - Updated Business Sign-Up and Business Account info to Material UI
    - Landry Emery Ishimwe Karangwa
- Sprint 3 (02/08 - 02/21)
  - Redesign Backend - Tyler Gerth and Rylan McAlister
  - Business Home Page
    - Menu bar, AppBar and linking necessary components to Business Home Page: Landry Emery Ishimwe Karangwa
  - Project-Wide Styling Updates - Colton Tucker
  - Initial Deployment to Heroku - Colton Tucker
- Sprint 4 (2/22 - 3/7)
  - Google Maps Business Search - Tyler Gerth
  - User login Page - Landry Emery Ishimwe Karangwa
- Sprint 5 (3/8 - 3/21)
  - Appointment Availability and Booking - Tyler Gerth
  - Reset account page - Landry Emery Ishimwe Karangwa
- Spring 6 (3/22 - 4/4)
  - Appointment Statuses (Requesting, Accepting, Cancelling) - Tyler Gerth
  - Viewing Existing and Upcoming Appointments - Tyler Gerth
  - Editing of Employee Data - Tyler Gerth
  - User Logout - Tyler Gerth
  - Stripe Integration - Rylan McAlister
  - Add Review Functionality - Dalynn Hatch, Brayden Alvarez
  - Styling update on login Page, account reset Page, and sign up page
    - Landry Emery Ishimwe Karangwa
- Sprint 7 (4/5 - 4/18)
  - User Messaging (Between Customer and Employee) - Tyler Gerth
  - Persistent User Sessions - Tyler Gerth
  - Creation of ‘First Available Appointment’ functionality within the Book Appointment page - Tyler Gerth
  - Addition of Employee Services to Employee Home - Tyler Gerth
  - Request the user’s current position on Home Page - Landry Emery Ishimwe Karangwa
  - Display businesses using Google Maps - Landry Emery Ishimwe Karangwa
  - Stripe Integration - Rylan McAlister
  - Business Performance Functionality - Dalynn Hatch
  - Business Rating Bugs - Dalynn Hatch, Brayden Alvarez

4.6 Deliverables

For the final project, we plan to submit the following items:

- Software Justification Document: This does contain the major software frameworks we used throughout the project and why we chose to use them.
- Software Design Document: This document does include an overview of each of the pages within our application and how they contribute to the overall goal of the application.
- API Document: This does contain the APIs and outside services we used throughout our project. This document does contain the API, a description of how and why it was used, and what we had to pay in order to access it.
- Website Code: This is the code for the Rezerve website that our team created throughout the project. Since this is a project that our product owner wants to move into a real startup, the API keys and any sensitive information will be removed from the codebase. Additionally, anyone wanting to access it must reach out to one of the group members, as our product champion wants to keep the code in a private repository
    - https://github.com/rezerve-startup/rezerve
- Link to the Deployed Website: https://rezerve-startup.herokuapp.com
- Final Report: This does contain the full description of our project, containing the work each of the team members accomplished from start to finish. It also contains a fleshed-out description of why the project was a necessary solution to the problem of barbershops and salons reopening their doors to customers post-pandemic, and how the solution we came up with solves the various facets of that problem. It describes the design decisions we made, as well as the timeline of the project. Finally, it describes any references we used when writing the report/coding the project.

5.0 Key Personnel

Tyler Gerth – Gerth is a senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. He has completed relevant courses to this project such as Software Engineering and Programming Paradigms which allows him to have an understanding of the software design process and how to work with several different programming languages in order to accomplish his tasks. Gerth has been working as an Applications Development Intern at J.B. Hunt since the summer of 2018, which has allowed him to gain an understanding of how software is actually used in the workplace and excel in creating full stack applications from frontend websites to microservices. Gerth was responsible for a wide array of items across the application, some of them including helping to set up the Firebase backend, the connection of React Redux throughout the application, the creation of Booking Appointments, Messaging, and the Customer Business Search.

Blake Hatch – Hatch is a senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. He has completed all required courses for his degree up to the last year, which allows him to have an understanding of the software design process required for this project. Hatch has been working as an Applications Development Intern at J.B. Hunt since November 6, 2018, which has allowed him to gain industry knowledge and excel in all aspects of a project. Hatch is responsible for the development on the business performance page as well as some functionality throughout the project with the reviews.

Landry Emery Ishimwe Karangwa – Ishimwe is a senior Computer Engineering major in the Computer Science and Computer Engineering Department at the University of Arkansas. Ishimwe has studied Software Engineering, which allows him to work in a team and design a register-like website where an employee can check out a customer. He had worked with the Razorbots team, using Ros C++ to implement the lunar robot's functionalities. Ishimwe worked on a web app during the Spring 2020 24 hours hackathon that records high notes and lower notes from voice input and then suggests what songs one can perform. Ishimwe has been responsible for creating the login page, the account reset page, and the linking page between User Sign Up and Business Sign Up pages that Tucker worked on. He also worked on the App bar and the functionality that displays businesses available on the home page using the User's current position and Google Maps.

Brayden Alvarez - Alvarez is a senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. He has completed Software Engineering, Database, and Programming Paradigms which gave him the knowledge for how to complete his tasks. Alvarez has been working as Application Development intern at J.B. Hunt since June 2020. This has given him knowledge of how tech works in the real world and how a professional development environment works. This far into the project, Alvarez has worked on the development of the business performance page alongside Dalynn Hatch. Alvarez has added some functionality through the review and ratings process and worked on bugs with the ratings process.

Rylan McAlister – McAlister is a senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. They have completed Programming Foundations and Programming Paradigms along with Software Engineering to give them the ability to take on this project. Other classes they have taken are Computer Networks and Database Management which also helps them be able to complete their work. Though still looking for an internship, they have been able to successfully create their website for software engineering and manage it on their own.

Colton Tucker - Tucker is a senior Computer Science student at the University of Arkansas, and is currently pursuing a Bachelor of Science degree in Computer Science. They have completed numerous CSCE courses such as Programming Paradigms, Algorithms, and many others. The experience gained from these courses as well as working as a part-time DevOps Engineering intern for First Orion will help him work with the team to develop this application. Over the course of the past two years at First Orion, Colton has worked in depth developing web applications, experiencing both the frontend and backend side of things. Tucker worked on creating the Employee Home Page as well as redesigning some of the style across the whole website. He also worked on the Sign Up functionality for users, customers, and businesses.

Nathaniel Bekele (Product Champion) - Bekele is a senior Architecture & Graphic Design student at the University of Arkansas with a focus on Environmental Design. He is a driven and curious individual with both personal and professional experience in many different fields. Bekele's desire is to create solutions for problems he encounters in his community, which has resulted in him diving into CS and tech due to its accessibility.Bekele took a managerial role throughout the project, where he met with the team members on a biweekly basis to touch base about the state of the application and provide mocks about how he wanted the application to look. He also was the contact for the businesses who will be using this application, and gave their feedback to the development team, so they could adjust their work accordingly.

6.0 Facilities and Equipment

Currently, this project has not required any external facilities or equipment. However, the project does use some outside API's (Google Maps and Stripe), as well as Heroku for deployment and Firebase for the backend.

7.0 References

[1] https://www.thecut.co/barbers

[2] https://gopanache.io/

[3] http://www.taper.app

[4] https://firebase.google.com/docs

[5] https://stripe.com/docs