CaBelle: Hotel Booking System Project

Final Report

Spring 2024

Alyssa Bustos - Quality Assurance and Front-End Developer
Carlos Hernandez - Quality Assurance, Back-End Developer, and Front-End Developer
Chelsea Ogbedeagu - Back-End Developer and Front-End Developer
Klarissa Navarro - UI/UX Designer and Front-End Developer
Michael Baldo - Quality Assurance, Back-End Developer and Front-End Developer
Tina Pham - Project Manager, Technical Writer, Back-End Developer, and Front-End Developer

Approach

Programming Languages

In our application, we used three coding languages for both our frontend and backend development: JavaScript, HTML (HyperText Markup Language), and CSS (Cascading Style Sheets). JavaScript is utilized for interactions between the client and the website such as: website sliders, pop-up messages, and other interactive responses from the user. We used this language to connect our code from Visual Studio Code to the FireBase database platform. HTML is used for the structure of our application and presents different utilities a client can use when visiting our website in their desired web browser. This language also allows us to determine the content that will be displayed in the web browser. Lastly, CSS is utilized for visual presentation of our HTML code in our website such as: styling of text, headings, font size, spacing, links, and paragraphs.

Algorithms

No algorithm was used for our application.

Datasets

Our dataset is our FireBase database. We had datasets for hotels, users, and booking information.

Libraries

FireBase has a wide variety of libraries available for backend development. We specifically loaded Firebase JavaScript SDK libraries from the CDN (content delivery network) into our backend code. We imported the initializeApp function from the FireBase App library. From the FireBase Authentication library, we imported the following functions: getAuth, createUserWithEmailAndPassword, signInWithEmailAndPassword, onAuthStateChanged, signOut, GoogleAuthProvider, and signInWithPopup. We also retrieved getDatabase, set, get, update, remove, ref, and child methods from the FireBase Database library.

Software Description

This software application allows a guest and/or registered user to book hotel room reservations by searching for rooms based on their preferences and selecting their arrival and departure dates. They may create, update, or delete their accounts, and cancel hotel room reservations. By canceling their room, there will be a prompt at the bottom of the page that notifies the user of a cancellation fee. As the user browses the web application with the navigation bar, they can learn more about affiliated hotels. The following information about the

hotels will be listed on several pages of our web application: hotel contact information, hotel address, hotel service hours, list of available facilities, types of rooms available, hotel policies, parking information, smoking policy, refund policy, discounts, special events, local tourist attractions, and more.

Hotel employees have access to booking and user information. They can record, manage, and modify reservations, check-ins, and check-outs. Managers will also be able to generate booking/billing receipts. The application will send a confirmation email to the customer once they have finished reserving a hotel room.

Front-End

In our frontend, we first designed our UI layout for each web page with the collaborative web design application, Figma. This gave us a general idea of how we want our web pages to look so we can have an organized plan of what we need to code and develop. The coding languages we used for the front-end were JavaScript, HTML, and CSS. Developing our CSS files was the most important step of this process since the front-end is the graphical user interface of our application. In other words, it is what the user sees and interacts with. We created code for navigation menus, buttons, images, text boxes, and many more.

Back-End

Similar to front-end, we used JavaScript, HTML, and CSS for the backend development. Developing our HTML file was the most important step of this process since the back-end focuses on the functions of our application. Our HTML files also included our JavaScript code that connects to our FireBase database.

Description of Software Algorithms

We did not use an algorithm for our application.

Database

We heavily relied on FireBase to store and retrieve user and hotel information. We also used it for user authentication via email sign-up and allow users to directly log-in to our application with their Google email address.

FireBase is an application development platform provided by Google which provides a set of backend cloud computing services. It hosts databases, services, authentication, and

integration for a variety of applications. Firebase uses multiple languages, primarily JavaScript, to cater to different platforms and developer preferences.

Examples of user data we stored were their full name, custom ID number, phone numbers, home addresses, and credit card information. This information is stored in FireBase's Real-time Database, and immediately updates the information when the user clicks on the insert, update, or delete interactive buttons on the Account Information web page. The hotel employees and managers can retrieve this data to check reservations and generate booking/billing reports. We also manually inserted the names of several hotels, and the prices for different hotel rooms. When the user selects their desired hotel destination, the database will send information about the hotel room prices to our application so it can be displayed on our room search and payment web pages.

Code Layout

In this section of the report, we will provide a description of the functionality of the files we have created.

- Login.html
    - This is the HTML code for the homepage and our login sidebar. Our code will insert our logo onto the homepage, and show a navigation bar that allows the user to easily browse to other web pages like our hotel location page, newsletter page, and discount page. The login sidebar will prompt the user to enter their email address and password into the provided text boxes. The user may also sign in with their Google email by pressing on the "Sign in with Google" button. If the user does not have an account, they may choose to either sign up, which will redirect the user to a registration pop-up page, or they may choose a Guest Login, which will redirect them back to the homepage and allow the user to continue browsing our application. On the registration page, it will prompt the user to enter their full name, email address, and password of choice. At the bottom of the web page, our code will display ways to contact the hotel.
    - Our HTML code also provides functions for all our interactive features like the web page buttons.
    - This file also includes our JavaScript code which connects to the authentication database of FireBase.
- Login.css
    - This is the visual representation of the Login.html file.
- Login.js
    - This provides additional JavaScript code and adds validation checks to our login page. Such as the user using an '@' symbol and ensuring that the user's password has a minimum length of numeric characters.

    - This will be available for every HTML file besides the user-account.html, database.html, confirmation.html, and payment.html (i.e. this file is located on web pages that we believe where a typical hotel would allow a user to sign in or register for an account).
- attractions-hotel-1.html
    - This file displays the local tourist attractions of the Hilton Hotel.
- attractions-hotel-2.html
    - This file displays the local tourist attractions of the Tipton Hotel.
- attractions-hotel-3.html
    - This file displays the local tourist attractions of the Double Tree Hotel.
- attractions.css
    - This is the visual representation of the attractions-hotel-1.html, attractions-hotel-2.html, and attractions-hotel-3.html
- confirmation.html
    - This file contains code for a confirmation message that will be displayed after the user has finished booking their hotel room. It will also display information to confirm the cardholder's name, hotel ID, room type, arrival date, departure date, and total costs.
- confirmation.css
    - This is the visual representation of the confirmation.html file.
- database.html
    - This code will allow employees and managers to login with their email or Google email to retrieve information about users. It will display the user's full name, ID number, phone number, home address, and credit card information.
- discount-brand.html
    - This file will display the discounts of the general hotel brand. Each discount will provide a name, description, and when the discount can be used.
- discount-brand.css
    - This is the visual representation of the discount-brand.html file.
- discount-hotel-1.html
    - This file contains code for one of the three hotels in our database. The information displayed will focus on the discounts for the Hilton Hotel.
    - This file also includes our JavaScript code which connects to the hotel database of FireBase.
- discount-hotel-2.html
    - This file contains code for one of the three hotels in our database. The information displayed will focus on the discounts of the Tipton Hotel.
    - This file also includes our JavaScript code which connects to the hotel database of FireBase.
- discount-hotel-3.html

    - This file contains code for one of the three hotels in our database. The information displayed will focus on the discounts of Doubletree Hotel
    - This file also includes our JavaScript code which connects to the hotel database of FireBase.
- discount-hotel.css
    - This is the general visual representation for all three discount-hotel.html files.
- hotel-1.html
    - This file contains code to display basic information about the hotels. The information displayed will focus on the Hilton Hotel. It will provide hotel contact information, hotel address, hotel service hours, list of facilities, types of rooms available, hotel policies, parking information, smoking policy, refund policy, and more.
    - The navigation bar provides discounts, newsletters, local tourist attractions and booking. If you click any of the following from the navigation, the user will be redirected to a different page from the one the user clicked on.
    - It's also the home page of the Hilton Hotel.
    - By the hotel name, there's a rating of the hotel.
    - This file also includes our JavaScript code which connects to the hotel database of FireBase.
- hotel-2.html
    - This file contains code to display basic information about the hotels. The information displayed will focus on the Tipton Hotel. It will provide hotel contact information, hotel address, hotel service hours, list of facilities, types of rooms available, hotel policies, parking information, smoking policy, refund policy, and more.
    - The navigation bar provides discounts, newsletters, local tourist attractions and booking. If you click any of the following from the navigation, the user will be redirected to a different page from the one the user clicked on.
    - It's also the home page of the Tipton Hotel.
    - By the hotel name, there's a rating of the hotel.
    - This file also includes our JavaScript code which connects to the hotel database of FireBase.
- hotel-3.html
    - This file contains code to display basic information about the hotels. The information displayed will focus on the Double Tree Hotel. It will provide hotel contact information, hotel address, hotel service hours, list of facilities, types of rooms available, hotel policies, parking information, smoking policy, refund policy, and more.

    - The navigation bar provides discounts, newsletters, local tourist attractions and booking. If you click any of the following from the navigation, the user will be redirected to a different page from the one the user clicked on.
    - It's also the home page of the Double Tree Hotel.
    - By the hotel name, there's a rating of the hotel.
    - This file also includes our JavaScript code which connects to the hotel database of FireBase.
- hotel.css
    - This is the general visual representation for all three hotel.html files.
- new-location.html
    - This file shows the different hotel locations that are in our database. By using the drop-down selection feature, the user may choose a hotel in their destination, and be redirected to the hotel's web page for additional information.
    - This file also includes our JavaScript code which connects to the hotel database of FireBase.
- new-location.css
    - This is the visual representation of the new-location.html file.
- newsletter-brand.html
    - This file contains code for displaying our newsletter, discounts, and exhibits.
- newsletter-redo.css
    - This is the visual representation of the newsletter-brand.html file.
- newsletter-hotel-1.html
    - This file contains code for displaying special promotions, discounts, and local tourist attractions for the Hilton Hotel.
    - This file also includes our JavaScript code, which is connected to the FireBase database.
- newsletter-hotel-2.html
    - This file contains code for displaying special promotions, discounts, and local tourist attractions for the Tipton Hotel.
    - This file also includes our JavaScript code, which is connected to the FireBase database.
- newsletter-hotel-3.html
    - This file contains code for displaying special promotions, discounts, and local tourist attractions for the DoubleTree Hotel.
    - This file also includes our JavaScript code, which is connected to the FireBase database.
- newsletter-hotel.css
    - This is the general visual representation for all three newsletter-hotel.html files.
- payment.html

    - This file will prompt for the user's credit card information. It will also ask the user to select the hotel they wish to stay at, select a room type, select their arrival and departure date via the calendar drop-down selection feature, and present the user with the estimated cost for their hotel room during their stay.
    - This file also includes our JavaScript code which connects to the hotel database of FireBase.
- payment.css
    - This is the visual representation of the payment.html file.
- policy-hotel-1.html
    - This file will display the policies for the Hilton hotel. It will include additional parking information, a detailed smoking policy and fine, a detailed refund policy, and an additional payment fee notice to notify users that they may have to pay an extra fee for using hotel facilities like the spa, bar, etc.
    - This file also includes our JavaScript code, which is connected to the FireBase database.
- policy-1.css
    - This is the visual representation of the policy-hotel-1.html file.
- policy-hotel-2.html
    - This file will display the policies for the Tipton Hotel. It will include additional parking information, a detailed smoking policy and fine, a detailed refund policy, and an additional payment fee notice to notify users that they may have to pay an extra fee for using hotel facilities like the spa, bar, etc.
    - This file also includes our JavaScript code, which is connected to the FireBase database.
- policy-2.css
    - This is the visual representation of the policy-hotel-2.html file.
- policy-hotel-3.html
    - This file will display the policies for the Double Tree Hotel. It will include additional parking information, a detailed smoking policy and fine, a detailed refund policy, and an additional payment fee notice to notify users that they may have to pay an extra fee for using hotel facilities like the spa, bar, etc.
    - This file also includes our JavaScript code, which is connected to the FireBase database.
- policy-3.css
    - This is the visual representation of the policy-hotel-3.html file.
- search.html
    - This file will allow users to search for hotels and hotel room types.
    - This file also includes our JavaScript code which connects to the hotel database of FireBase.
- search.css

    - This is the visual representation of the search.html file.
- user-account.html
    - This file will prompt the user to enter their information into text boxes on the Account Information page. It will ask the user for their full name, custom ID number, phone number, and home address. This file also contains code for interactive buttons, like the Insert, Update, and Delete buttons, and their functions. By using these buttons, the user will be able to insert, update, and delete their information from the FireBase database. FireBase will update this action and the user's information in real-time.
    - This file also includes our JavaScript code which connects to the user database of FireBase.
- user-account.css
    - This is the visual representation of the user-account.html file.

Conclusion

Initially when we were deciding on roles, most of the group had stated that they did not have experience in frontend and backend development. For some of us, this was our first time coding with JavaScript, HTML, and CSS for front-end and back-end development. We had a difficult time figuring out how to code in these languages in a short amount of time, so we were unable to accomplish everything we had originally planned. Similarly, for most of the group, it was also our first time working with the FireBase database so we spent a lot of time learning and developing the backend portion of our application. We had to watch and read many tutorials to properly connect and implement our code from Visual Studio Code to FireBase. In addition to our time constraints, we had limited time to code as many of us had outside commitments like work, assignments, projects, and exams in other classes.

Due to time constraints for this semester-long project, there were several features that we had to omit from our original proposal. For example, we did not have enough time to figure out how to send the user an email that confirms their reservation. Also, although we have a section for resetting the password displayed, the functionality wasn't implemented, so we could not allow a user to reset their password if they decided they simply wanted to change their password or if they needed to change it if they forgot. We also did not have enough time to figure out how to send a password reset link and newsletters to the user's email. Another feature that we had to exclude was a feature to use an online payment form like PayPal and Apple Pay. While researching, we discovered the only way to connect these online payments was to have a legitimate business. Therefore, we could only allow credit card payments on our web application. Other features that we had to leave out were the option to choose the number of guests per room, showroom availability for both users and employees, allow users to retrieve information about their previous reservations, allow employees to upgrade hotel rooms for the user, add additional fees to billing receipts, and send the user a cancellation fee email.

There was one feature that we had to make a modification that deviated from our original proposal. Our original plan was to allow a registered user to find the closest hotels with their zip code. We modified this idea later in the planning process and decided that we would allow users to search hotels based on different locations worldwide. Our development team designed a map where users can click on a destination in the world and find the closest hotel to where they would like to visit. However, we did not have enough time to implement this so we simplified this idea by adding three general hotels into our database without specifying which parts of the world would have the affiliated hotel.

Overall, this was a good learning experience as we explored and learned how to develop software in multiple coding languages. The main lesson that we learned from this project was how to develop code in a team setting since we all had many ideas and opinions on how to execute our web application. It was really helpful to learn about creating a SRS, Class Diagram, and Data Flow Diagram. These three documents greatly assisted us in our structural plan for our application. Although we did not have time to implement and fix any errors we encountered from our test cases, it was also our first time creating test cases so we can use this knowledge for future code development projects. This was a great introduction to learning how software engineering works. We will learn from our errors and strengthen our development skills in the near future.

Future Works

If we were to continue this project in the future, then we would improve on our CSS files. We spent most of our time developing our backend code so we did not have as much time to develop the frontend portion of our web application. We would also add a feature to ensure the security of user's private information, add more hotel locations in different parts of the world, add more hotels into the database, add a feature that will prompt the user to use a strong password with specifications, add a feature that verifies the validity of the user's credit card, and add additional costs (i.e. room service) to billing. We would also figure out a better way to allow online payment services, like Paypal, as an option to pay, allow users to receive newsletters, discount info, and promotional events via email, and send confirmation emails to users after they have finished booking their reservations. Lastly, we would also work on how to fix a scaling issue we had encountered. Since we all used different devices to code, some of the scaling for our files would look different on other devices. For some group members, the words on the files would overlap while others would have gaps. If we were given more time, then we would work on a method to ensure our web application would look the same on any device and web browser.

References

- Paterson, Keith. “Connect an HTML Webpage to a Firebase Database for CRUD Using Vanilla JavaScript.” *YouTube*, 15 July 2022, https://www.youtube.com/watch?v=_p4Hgzm_oNQ.
- Paterson, Keith. “Simple Authentication with Firebase 9 and Vanilla JavaScript.” *YouTube*, 19 Feb. 2023, https://www.youtube.com/watch?v=ERoJpIT1BHY.
- Simon, Gary. “From Prototype UI Design to HTML, CSS & JS Tutorial.” *YouTube*, 5 Sept. 2019, https://www.youtube.com/watch?v=GTyMUjhA-o4.
- Tutorials, Creative. “Hotel Booking App Using React and Firebase[Full Video].” *YouTube*, 8 Apr. 2022, https://www.youtube.com/watch?v=cJ-PC2lb4b0.
- Anas, Shaikh. “How to Make a Responsive Hotel Booking Website Using HTML / CSS / JS.” *YouTube*, 22 Oct. 2022, https://www.youtube.com/watch?v=RO4ukusoTh8.
- Halid. “How to Create Navbar in HTML and CSS.” *YouTube*, 12 May 2023, https://www.youtube.com/watch?v=2DVkNvVcQAU.
- Halid. “How to Make a Website with Login and Register Page: HTML CSS & Javascript.” *YouTube*, 14 Mar. 2023, https://www.youtube.com/watch?v=uTCmNDnP2VY.
- Delaney, Jeff. “Node.Js Ultimate Beginner’s Guide in 7 Easy Steps.” *YouTube*, 21 May 2020, https://www.youtube.com/watch?v=ENrzD9HAZK4.
- Media, Traversy. “Intro to JavaScript Unit Testing with Mocha JS & Chai.” *YouTube*, 1 May 2017, https://www.youtube.com/watch?v=MLTRHc5dk6s.