Assignment 3 Report

Darius Seifert Booth

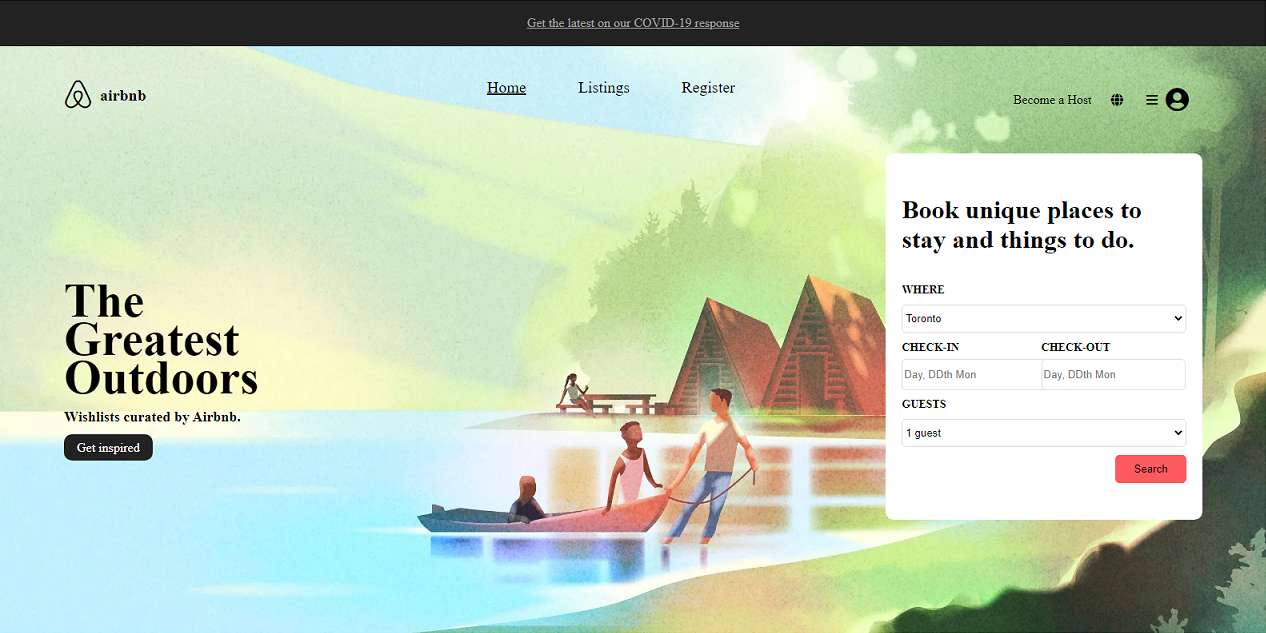
109908202

[dseifert-booth@myseneca.ca](mailto:dseifert-booth@myseneca.ca)

Table of Contents

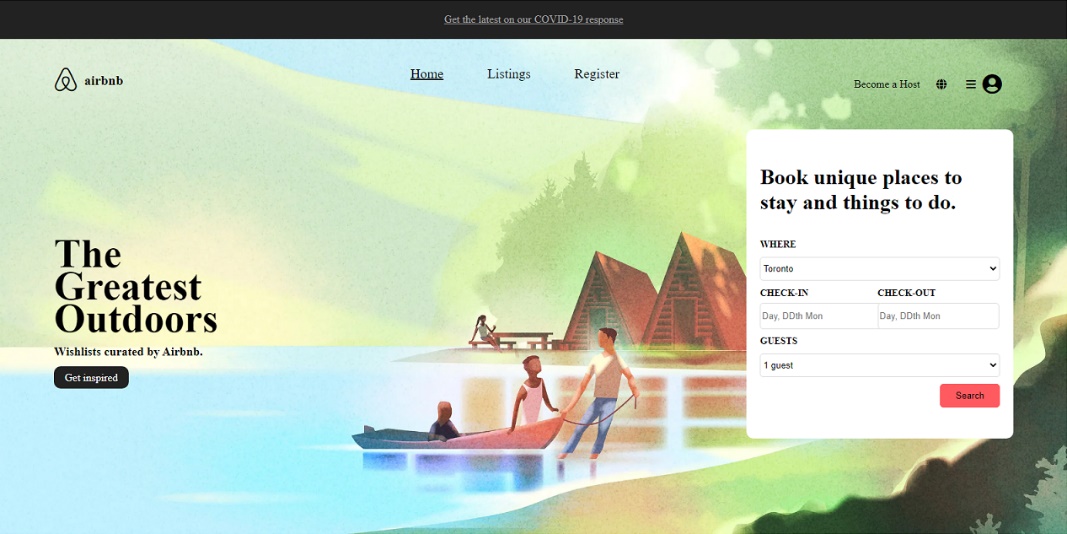
1. Purpose of Application
2. Home Page
3. Listing Page
4. Registration Page
5. Login Page
6. Dashboard Page
7. Database Functionality
8. Libraries used
9. Github/Heroku Links

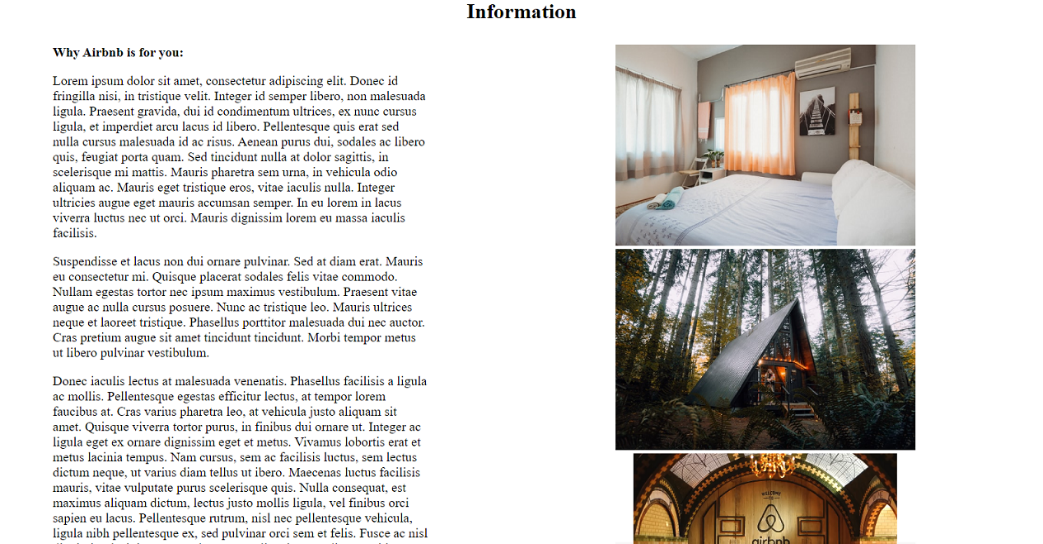
1. Purpose of Application

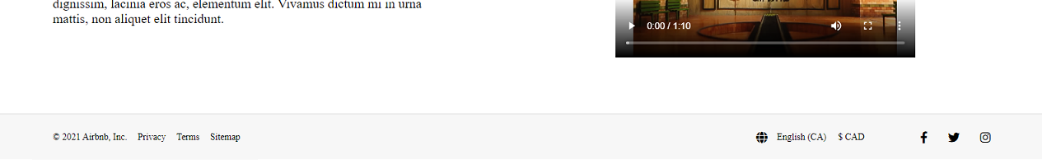


The purpose of my application is to replicate the appearance of Airbnb’s website, with functionality that matches what Airbnb’s website would offer its clients.

2. Home Page



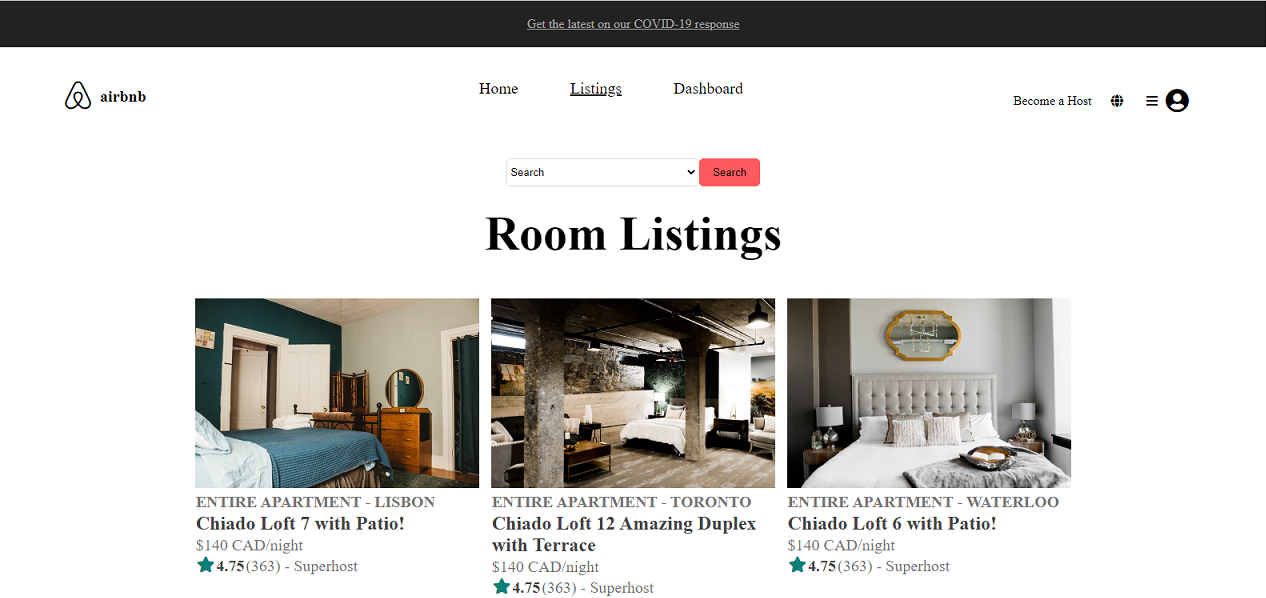


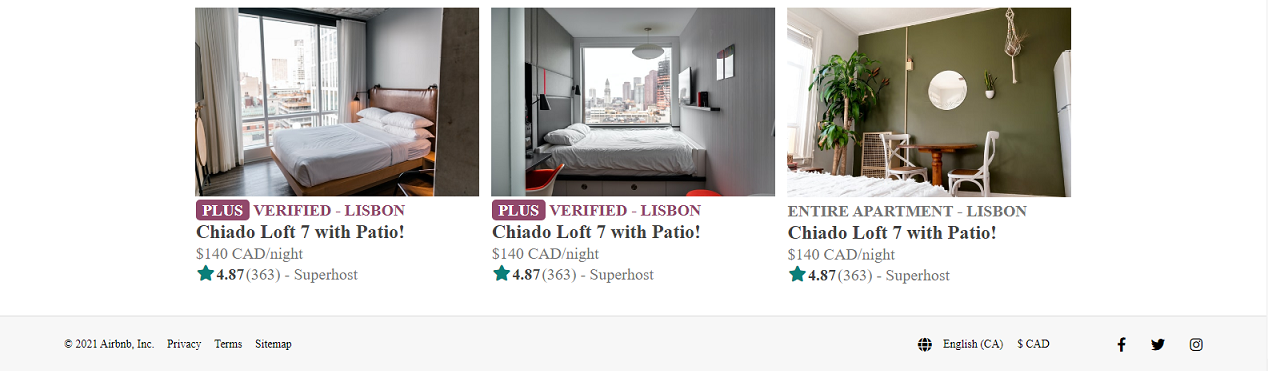


The home page is simple, providing a background, some pictures, a video, and text that can be replaced with information regarding Airbnb and what it has to offer. There is a form at the top which, at the moment, does nothing. Across the top of the page, there are links which can take you to the Room Listing page and the Registration page, and at the top-right, there is a profile icon which can take you to the Login page.

Should a user be logged in, the links to the Registration and Login pages will be replaced with links to the user’s Dashboard page.

3. Listing Page





The Listing page provides a look at rooms that are currently logged into the system. Navigation functionality remains the same as the Home page’s.

The search bar at the top allows any user (logged in or not) to pick a location, and the page will display all matching rooms.

If an admin is logged into the site, they have access to a form at the bottom of the page, which will allow them to enter a new room into the database.

4. Registration Page

Graphical user interface, application

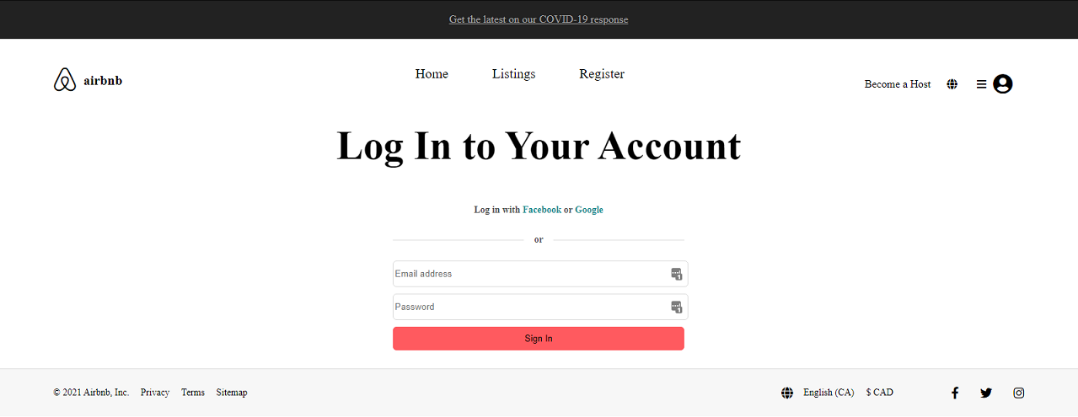
Description automatically generated

The Registration page supplies clients with a form with which they can fill out basic information so that the application can create an entry in the database for their profile. The form will display an error message for each text box, should any fields be left empty. The email field will display a separate error message should there already be an existing user with the email provided, as well as some extra error messages for the password field.

Once all fields are filled out and valid, the application creates a new user profile and saves it to the database, sends an email thanking the user for creating an account with Airbnb, then creates a new session for the user, and redirects the user to their new Dashboard page.

As of right now, the Registration page is currently bare-bones, as I am unsure of how I want to style it.

5. Login Page

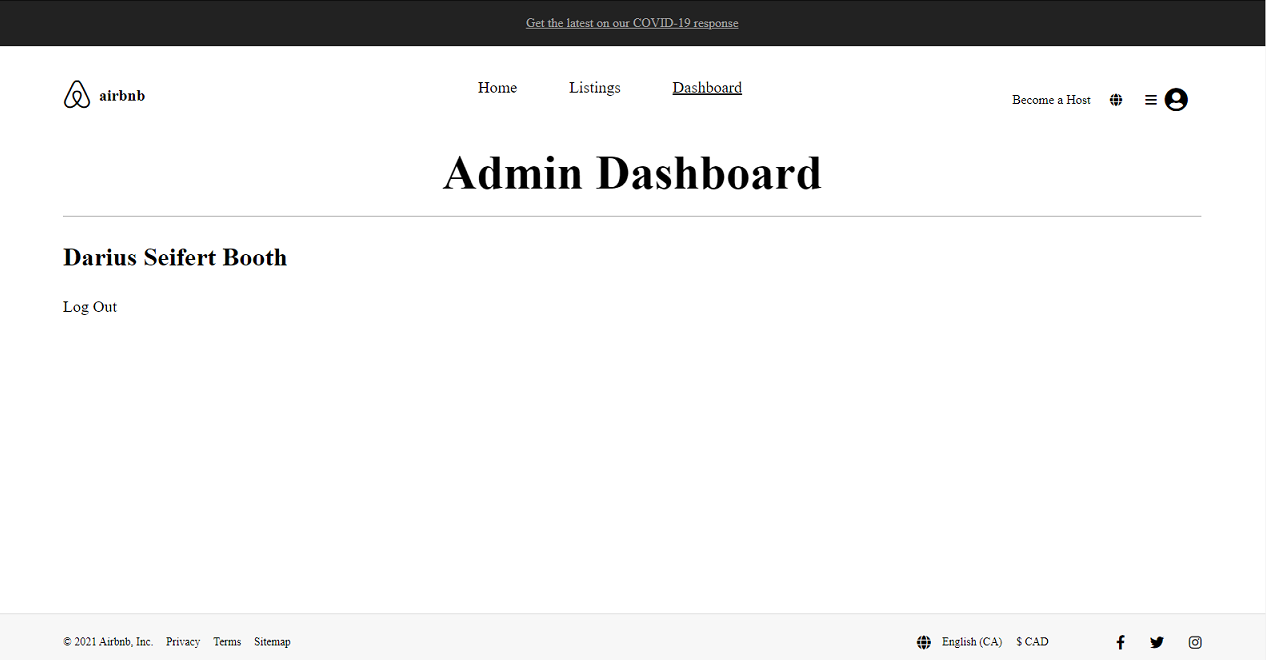


The Login page provides a form for returning users to log into their accounts. The login form will display error messages should the fields be empty. The email field will provide an additional error message should the email provided not be found within the database, and the password field will provide an additional error message should the password not match the profile of the email provided.

Once the fields are filled out and valid, the application creates a new session for the user, and redirects them to their Dashboard Page.

As of right now, the Login page is currently bare-bones, as I am unsure as to how I would want to style it.

6. Dashboard Page



The Dashboard page does not do much as of right now. It replaces the Registration and Login pages when a session is active, and the Log Out link will destroy the current session and redirect the client to the Home page. The name of the user is displayed on the dashboard.

The Dashboard page checks to see if the current user is an admin or a normal user, and displays a different header accordingly.

As of right now, the Dashboard page is currently bare-bones, as I am unsure as to how I would want to style it.

7. Database Functionality

The application connects to a MongoDB database, which currently stores all user profiles that have been registered to the system. All code regarding the database and users is handled through the user.js file, which is called by the server.js file.

As of right now, there is no way to set a user profile to the admin state outside of directly altering the profile in the Mongo database. As we wouldn’t want random users to be able to set their accounts to admin states, this seemed like the best solution.

The database now retains all information about each room, and the Listing page will dynamically retrieve and display information about each room. New rooms can be added by admins, however there is currently no way to add reviews to each room.

8. Libraries Used

The only library used for this application outside of what was mandatory for completion of the assignment is ‘nodemailer’, which is used to send the email to the account of the user that registered with Airbnb.

9. Github/Heroku Links

To the Github repository:

<https://github.com/dseifert-booth/WEB345-Assignment>

To the Heroku repository:

<https://dashboard.heroku.com/apps/dseifert-booth-assignment>