



ASSIGNMENT 3-5

WEB322

Stephen Ditta

scditta@myseneca.ca

033787144

WEB322NBB

Sharmin Ahmed

Table of Contents:

1) Intro

2) Assignment 3

a) Application Architecture

b) User Registration Module

c) Authentication Module

3) Assignment 4

a) Administrator Module

b) Search Module

4) Assignment 5

a) Room Description Page and Book Room Module

Intro

During this assignment 3-5 I accomplished many tasks through the front-end as well as the back-end while working on my Airbnb clone (A place to stay). As always it was a challenge but was very rewarding in the end putting the whole project together. Throughout this report I will be going over a few details of what I did and how I accomplished it, starting at the user registration module.

Application Architecture

Application architecture was interesting as well as helpful when it came to pathing and directory organization in general. Being able to follow the MVC design pattern helped me understand where my files were, what they were in charge of and made the pathing so much cleaner and easier to read / understand.

User Registration Module

While working through the user registration module I was able to gain a better understanding of how the connection to the NoSQL mongo dB database through the back-end. A little bit of research as well as reviewing the online notes for WEB322 helped me to understand how the connection was accomplished. Installing the dependency mongoose and requiring it gave me access to mongoose's massive library of code. From this point I my research and understanding helped me to build that connection using a "mongoose.connect" which connects me directly to my mongo dB database "assignment". From there I created a short debugger to check if the database was truly connected when the server was launched through node.js (localhost).

Authentication Module

The authentication module was probably the trickiest part for me during the time I spent on this assignment. It took me time to understand how I was going to accomplish this module but I broke it up into parts which helped a lot. I started with adding my required fields and dependencies such as express-handlebars, body-parser and client-sessions. These dependencies were the main structure throughout the assignment.

- Express handlebars made it efficient and easy to render information from the database to the page. It allowed me to show and hide information based on specific criteria I provided it.
- Body parser extracted the entire body of a request and allowed me to utilize the extracted information within my handlebars and mongoose. Body parser is also used strictly for text only.
- Client session allowed me to create a unique session middleware. This session was useful because once a user is logged into their account their information is saved with the session. Unless the server was restarted the user can go to any of the pages, logged into their account. At a specific time of inactivity however, the session will be automatically reset logging out the user.

I also used encryption when storing the password to the mongo dB database collection “users” using the third-party package bcryptjs. Using this created a layer of encryption over the password masking it with a long randomized UUID. The purpose of using encryption is so that if the event of a hack or virus was to get to the database the passwords would be safe not directly displaying what each account wrote.

A Place To Stay Home Room Listing Stephen Ditta ▼

Welcome to the dashboard
Stephen Ditta
You are an admin

[Create](#) [Update](#) [Delete](#)

Create a Room

Room Name

Cost

Description

Location

Photo(s)

```

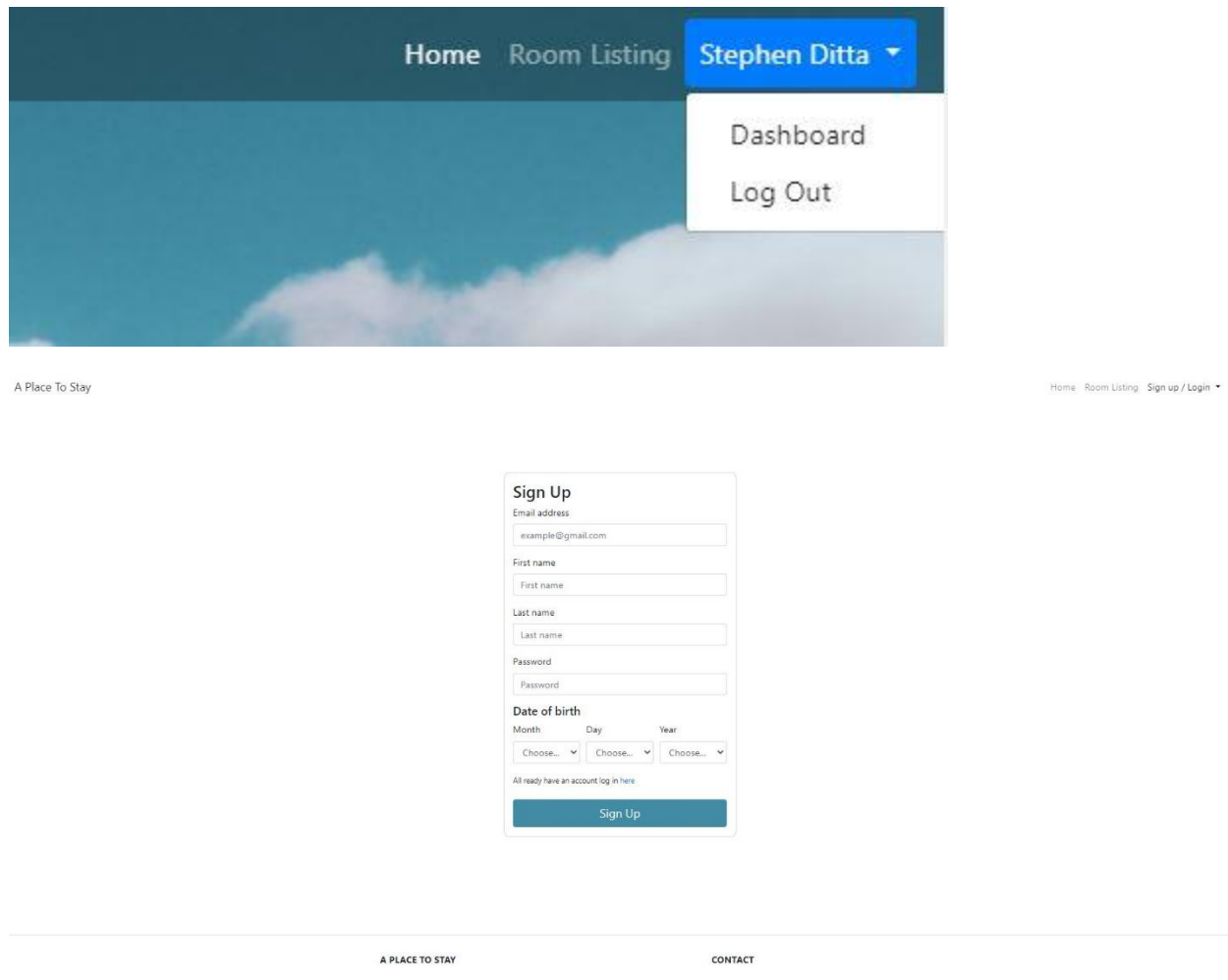
_id: ObjectId("5fc9565cb6c1c56f48fb5bbd")
admin: false
firstname: "Brad"
lastname: "Chad"
email: "brad@gmail.com"
password: "$2a$10$z/LUXDp8tLucsuUt62/IKu4C0Z07wBucsybGXgddnDL/ryFR1tom"
dateOfBirth: 2002-09-21T04:00:00.000+00:00
__v: 0

```

```

_id: ObjectId("5fd431eadee3745e90f87bde")
admin: false
firstname: "Stephen"
lastname: "Ditta"
email: "scditta@myseneca.ca"
password: "$2a$10$HEvq1bbGPJm2MaQNS/05o0mks4ruBXAXUKHJ8Nududu08jrL93rFq"
dateOfBirth: 1996-08-01T04:00:00.000+00:00
__v: 0

```



Administrator Module

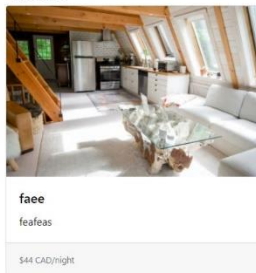
For the administrator module I designed my database to work where once a user creates a new account the collection called "rooms" containing the documents room title, price, description, location, and photo are store. The rooms can be populated through the dashboard if the user is set to an admin. Otherwise if the user is not an admin, they can just look around the website and navigate through the room listings.

```
_id: ObjectId("5fd42e74dee3745e90f87bdd")
roomtitle: "faee"
price: 44
description: "feafeas"
location: "Vancouver"
roomphoto: "1607741044281room1.jpg"
ownername: "Stephen Ditta"
owneremail: "sdittawork@gmail.com"
__v: 0
```

A Place To Stay

[Home](#) [Room Listing](#) [Stephen Ditta](#) ▼

Rooms



A PLACE TO STAY

Maecenas rhoncus nibh ut arcu
vestibulum, pellentesque rhoncus
tellus consequat. Proin aliquet a

CONTACT

🏠 Aurora, ON 2321, CA
✉️ scditta@myseneca.ca

Search Module

A search module was also created on the home page allowing the user to fill in the form. Once successfully completed a find is used to gather any room that has information according to the location specified. The user is then redirected to the room listing page.

Room Description Page and Book Room Module

Finally, the last part was to allow the user to be able to click on the populated rooms and receive the information accordingly based on the room of choice. I achieved this through handlebars being able to utilize the id of the document from the database and when a room is clicked it finds and only displays the documents that match with the document id. The room also allows logged in users to be able to click on the booking button once both check in and check out dates are satisfied. Once done a ticket is

created and info such as the dates, how long the booking was for, and the total price are displayed for the user.

A Place To Stay

[Home](#) [Room Listing](#) [Stephen Ditta](#)



faee

faafeas

\$44 CAD/night

Check-in Date dd/mm/yyyy

Check-out Date dd/mm/yyyy

[Book Room](#)

Overall, this assignment was challenging but I was really satisfied with the outcome learning and understanding how the back-end of web application work as well as develop throughout production.