TECHNICAL

DOCUMENT

Pandora's Website

TABLE OF CONTENTS

CREATING THE DATABASE

☐ Download MongoDB at https://www.mongodb.com/download-center/community if it is not already installed on your local machine.

☐ Open command prompt by typing *cmd* in the search bar located at the bottom of your screen (for Windows OS).

☐ Navigate to the directory where you will be using MongoDB; *server* folder in this case.

☐ Type *use Website* in the command prompt.

☐ Type *db.users.insert({name: "Cameron Green", email:* ccgreen2@uh.edu*, password: "test1234"})* in the command prompt. (*Note:* user authentication is required, so logging in as a premade user may not be possible. It is recommended to register as a user prior to logging in)

☐ See *Figure 1* in the Appendix for a successfully created database and its collections.

SETTING UP SERVER SIDE

This section assumes that you have successfully completed all the previous section(s). If you have not, please go there and complete the missing section(s).

☐ Open Visual Studio Code if it is not already open.

☐ Open the root directory for the project – *Cameron_Green_Final_Project_4339.*

☐ Open a new terminal and type *cd server.*

☐ Once the terminal has moved to that directory, type *npm run dev.*

☐ Ensure that the message logged to the console reads as follows:

```
Space Travel Agency is boarding on port 4000
http://localhost:4000
```

☐ If the message is logged as such, proceed to the next section. Otherwise, repeat the processes in this section.

☐ See *Figure* 2 in the Appendix for a successfully initialized *server* folder.

SETTING UP CLIENT SIDE

This section assumes that you have successfully completed all the previous section(s). If you have not, please go there and complete the missing section(s).

☐ Open a new terminal and type *cd client*.

☐ Once the terminal has moved to that directory, type *npm run serve*.

☐ Ensure that the message logged to the console reads as follows:

`DONE`   `Compiled successfully in 1386ms`

(note: the number of milliseconds needed to compile can
be any number. It does not have to be 1386ms).

☐ If the message is logged as such, proceed to the next section. Otherwise, repeat the processes in this section.

☐ See *Figure 3* in the Appendix for a successfully initialized *client* folder

GETTING TO THE WEBSITE

This section assumes that you have successfully completed all the previous section(s). If you have not, please go there and complete the missing section(s).

☐ Open any browser.

☐ Type http://localhost:8080/ (or whichever URL is provided by the terminal) into the URL.

☐ See *Figure 4* in the Appendix for a successfully initialized website.

☐ Register an account or log in if an account already exists for you (users **need** to register if they have not already done so).

Once you are logged in, you are free to explore the website. Have fun and remember; here at Pandora, we make the impossible, possible!

APPENDIX

Figure 1 shows the collections created in the MongoDB database once scripts are successfully run. (***Note:*** user authentication is required, so logging in as a premade user may not be possible. It is recommended to register as a user prior to logging in)

```
_id: ObjectId("5ea8cbaf8b20fd15d42cff3f")
name: "Cameron Green"
email: "ccgreen2@uh.edu"
password: "test1234"
```

Figure 2 shows the terminal message on a successful running of the *npm run dev* script for the *server* folder.

```
[nodemon] 2.0.3
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index.js`
App is running on 4000
Database is connected
```

Figure 3 shows the terminal message on a successful running of the *npm run serve* script for the *client* folder.
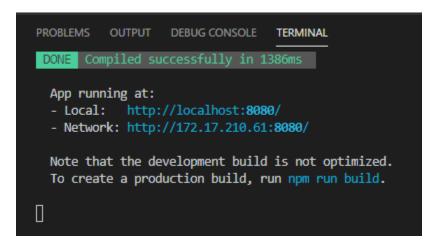
```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

 DONE  Compiled successfully in 1386ms

  App running at:
  - Local:    http://localhost:8080/
  - Network:  http://172.17.210.61:8080/

  Note that the development build is not optimized.
  To create a production build, run npm run build.
```

Figure 4 shows the page users should see upon entering the website.