LocalZonely

Final Project Report

CSCI 3308: Fall 2022

Team 016-04

Pierce Avner, Keith Bliss, Jithin Renji, Nick O’Connor

December 2, 2022

Project Description:

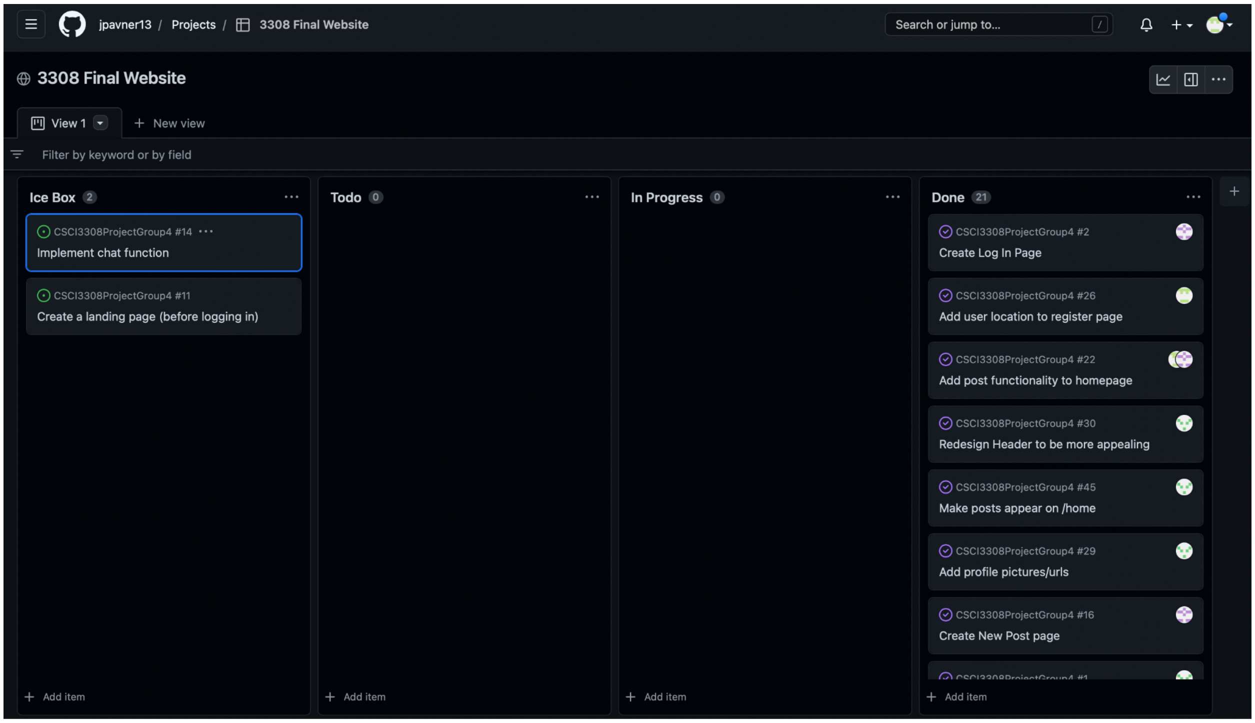
LocalZonley is a website designed to allow users to make the most of their travel experiences by connecting them with locals from their travel destinations. LocalZonely leverages the knowledge of locals from around the country to enhance the travel experience of those visiting other peoples' hometowns. Users of our site can sign up for an account where they, themselves, act as both a local expert for their own hometowns, and can connect with other local experts to find out more about the places they want to travel.

Our goal is to make travel experiences better. Travel is expensive, and we want people to make the most of their costly travels. The best way to experience a place is by talking to a local! By using our website, you can find the best attractions such as museums, bars, and restaurants and get local advice about your travel plans.

Our application is hosted locally currently, but future versions of LocalZonely will be hosted publicly and will be available to users across the internet. We have created this application through the use of a common stack based on NodeJS and a PostgreSQL server hosted in Docker Containers for maximum compatibility across systems.

Project Tracker:

Our project tracker can be accessed through the GitHub project board attached to our repository. The current status of the board is shown in *Figure 1* below.
https://github.com/users/jpavner13/projects/2/views/1


jpavner13 / Projects / 3308 Final Website
Search or jump to...
3308 Final Website
View 1
New view
Filter by keyword or by field
Add item

Ice Box 2

CSCI3308ProjectGroup4 #14
Implement chat function

CSCI3308ProjectGroup4 #11
Create a landing page (before logging in)

Todo 0

In Progress 0

Done 21

CSCI3308ProjectGroup4 #2
Create Log In Page

CSCI3308ProjectGroup4 #26
Add user location to register page

CSCI3308ProjectGroup4 #22
Add post functionality to homepage

CSCI3308ProjectGroup4 #30
Redesign Header to be more appealing

CSCI3308ProjectGroup4 #45
Make posts appear on /home

CSCI3308ProjectGroup4 #29
Add profile pictures/urls

CSCI3308ProjectGroup4 #16
Create New Post page

CSCI3308ProjectGroup4 #1

Figure 1. GitHub Project Board.

Demonstration Video:

https://drive.google.com/file/d/1TdWsJx3_r2yu7T337fZuc6OEFOKClFRJ/view?usp=sharing

Version Control System:

The project is hosted through the GitHub version control system and can be accessed at: https://github.com/jpavner13/CSCI3308ProjectGroup4

Team Contributions:

Pierce:
I initially started by creating the repository with the docker containers and basic needs for a functional website. This way we could all start working independently without conflicts from setting up the site from the very beginning. After that, I created the menu and footer. From there, I started tackling the home page with the functionality of the search bar, queries for the database based on desired filters, as well as cards to display all of the desired information about users and posts from a given location. I also worked on styling to make the home page more dynamic.

Keith:
I created the front end for the register page, and worked on integrating both the register and login pages with their respective backend components. Additionally, I worked on managing the project tracker in the GitHub project board to ensure that we were up-to-date on the current status of the project. I led the development of our final presentation and created both the Use-Case Diagram and the Architecture Diagram for the project.

Nick:
I was the one who set up the early features of the index.js and the sql database. All of the database's tables were first implemented by me but later updated by other users. In addition to setting up the index.js file, I also implemented the login, register, and user pages' GET and POST functions. The users.ejs page was also done by me.

Jithin:
I created the initial wireframes for the website and the front end pages for the login and new post pages. I also implemented the GET and POST API routes for the new post page. I made sure that these new API routes would integrate well with the front end for the home page. I was also responsible for the live demo and the demo video for this project.

Use-Case Diagram:

The Use-Case Diagram for the LocalZonely application is presented below in *Figure 2.* As shown, there is only 1 possible user type for our application. This user has permissions to access all of the functions shown. Administratively, a user can create their account, login and logout, and edit the information attached to their account. Functionally, a user can initially search for a desired location, where they will then be able to view all users attached to that location, as well as all posts that have been generated for the location. The user can also create their own posts for the location that they have attached to their profile.

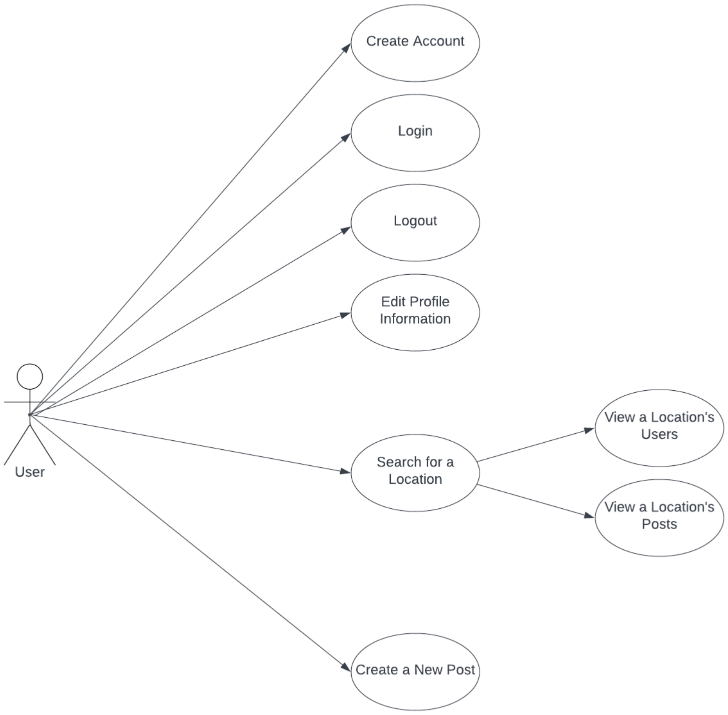


Figure 2. Use-Case Diagram.

Test Results:

Test Case I: User creates a new account

- **Scenario:** The user will be presented with the initial login screen of the site and will be asked to register a new account for themselves on the site.
- **Results:** The user quickly noticed the redirect link to the registration page. They were able to quickly fill in all required fields relating to their personal information (name, email, username, password, etc.). We noticed that the user was not prompted on the requirements for some fields, such as the password complexity requirements, which would be a future improvement. The user also presented some confusion on whether some fields were optional or required, such as phone number or twitter handle. We adjusted the formatting of the page to clearly denote the optional fields. Lastly, the user had some difficulty understanding the use of a URL for profile picture upload, but this is a limitation of our technology stack and technical knowledge.

Test Case II: User logs in to an existing account

- **Scenario:** The user will be presented with the initial login screen of the site and will be asked to login to an existing account with provided information.
- **Results:** The user was quickly able to identify the key fields for login on the main login page. They successfully input a given username and password, and were able to login to the site via the submit button, where they were then redirected to the homepage.

Test Case III: User views local experts and posts from a location

- **Scenario:** The user will be given a location to search for and will explore posts and users attached to that location.
- **Results:** The user was initially confused about what to do to accomplish the task when presented with the home page. In future design iterations, we plan to adjust the homepage layout to make it more clear that the user should input a location in the search bar field. Additionally, we plan to present post information directly on the homepage, to make it more clear to the user that they have successfully logged in to the site. The user then figured out how to input the desired location into the search bar, where they were then successfully redirected to the users and posts for that location. They then scrolled through the page, and were able to recognize the difference between users and posts for their given location.

Test Case IV: User updates their account with new information

- **Scenario:** The user will be presented with the homepage for the site and will be tasked with adding phone number information to their profile.
- **Results:** The user quickly recognized the "edit profile" option on the navigation bar and was successfully redirected to the profile editing page. They then quickly recognized the empty field where they input their phone number and successfully saved their changes. The green message indicating that they had successfully made the update was presented to the user, but there was still some confusion on where they could see that they had completed the update.

Deployment:

We had initially wished to deploy our website on the CSEL server, but encountered technical issues which prevented us from doing so. To maintain reliability, we decided to use local hosting, which we had been using throughout the development process. To use the application, first, clone the GitHub repository to your local machine via https://github.com/jpavner13/CSCI3308ProjectGroup4. Once the repository is cloned, within the "Components" directory, run the command "`docker-compose up`" from your terminal. Note that you must have previously installed the Docker Desktop application. After running this command, open the web browser of your choice, and navigate to "`localhost:3000`". From here, you will be presented with the Login page of the LocalZonely application. You can login with an existing account, or you can choose to register a new account where you will be redirected to the Registration page. Once you have concluded using the application, return to your terminal and run the command "`docker-compose down`" to terminate the site.