Party Finder

Zoie Nuño
Aidan St. Cyr
Nolan Lee
Hayden Schlichting
Dan Medvedev
Ali Almutawa Jr.

Project Description

PARTY  FINDER

DEMO

http://r-16-t-5.eastus.cloudapp.azure.com:3000/login

Goals

PARTY FINDER

- **Application goal: bring random individuals together for shared parties.**
  - Login and register pages will be available.
  - Pages are connected to a database.
  - Database stores user's first and last name, email, username, and hashed password.
  - Profile page will display user information.
  - Information displayed on the profile page includes the user's name, email, the number of parties they have attended, and the number of parties they have hosted

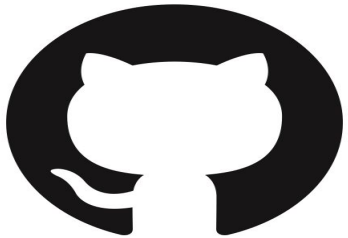
Party Page Overview

PARTY FINDER

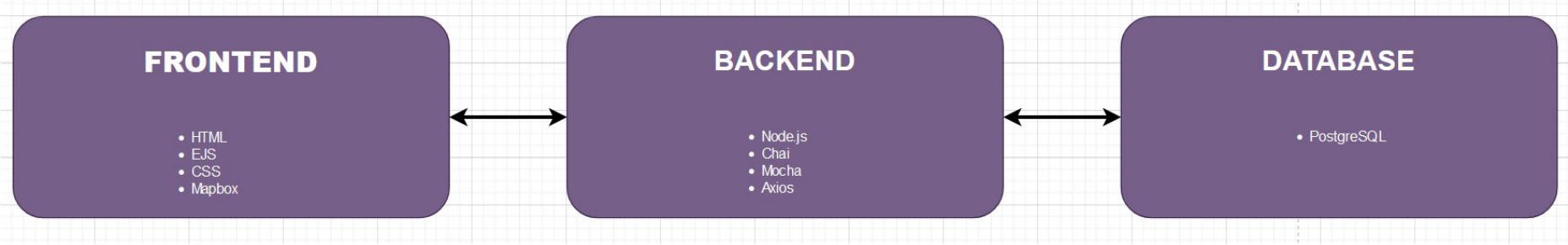
- Party page will display a map of parties based on geographic location.
- Parties will be represented by pinpoints on the map.
- Clicking on a pinpoint will display party information.
- Information displayed includes the date, start time, location, name, image, and description of the party.
- Pop-up window will display the party host and their information.

In addition, users will be able to create, delete, and edit their own parties as they chose.

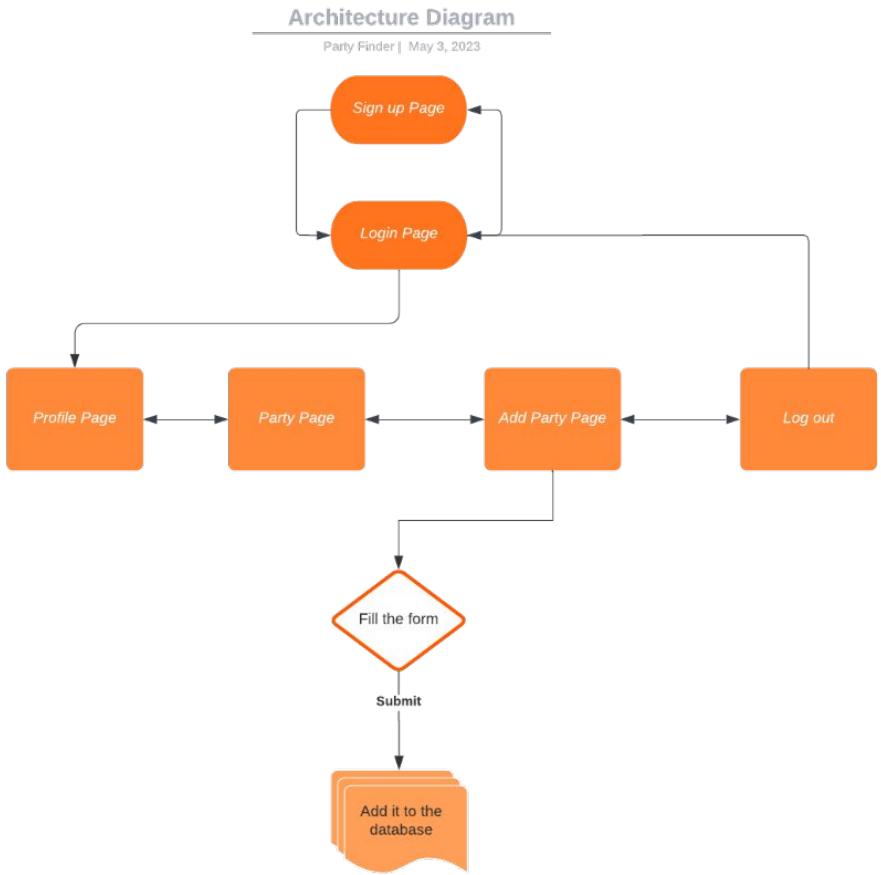
Tools

Name	Logo	Purpose	Rating
Github Project Tracker		Used to keep track of the progress made on the project by each member	★★★☆☆
Github		VCS Repository	★★★★★
MySQL/PostgreSQL		Creation of the database for storing user and party information	★★★★★
VS Code		Code development environment	★★★★★

HTML		Frontend web page design	★★★★★
EJS		Used to embed Javascript within HTML	★★★★★
NodeJS		Application server	★★★★★
Microsoft Azure		Deployment server (Platform as a service) (Paid) (No infrastructure needed)	★★★★☆
Mapbox		Outside API in order to display a map	★★★★☆



Architecture Diagram



Architecture Diagram

Challenges

- Converting user inputted addresses into latitude and longitude for map API.
  - Had a lot of challenges with standardizing the latitude and longitude throughout the API
  - Challenges getting the map size to fit correctly within the other elements
- Allowing users to upload a personal profile picture, as well as saving the image to be rendered on the profile page.
  - We able to have profile picture update within same session
  - Could not find a way to store picture and redisplay
  - Did not figure out how to make back end and front end work together.
- Displaying the number of parties attended and hosted on the profile page
  - As this information is held in a different table than the primary user information, we had to do a separate query using the userID to get the values from their respective table. We also had to check whether or not this data was null, and if it was, we had to default the values to zero in order to get it to display.

Additional Challenges

- Back end to front end linkage
  - Creating API calls within a function of a script inside a .ejs file

- Making sure the user cannot log in without the correct credentials.
  - Checking the database, and making sure the credentials they entered are correct, and in the right format.

- Displaying user details on their profile page.
  - We had to pull the information being input from the register page, and then use that same information to display it on the part page.

- Displaying party host information on party page.