Ali Ezzeddine

Full Stack Engineer

P. 860-818-6558|a.k.ezzeddine@gmail.com Portfolio | LinkedIn | Github

SKILLS

Languages: Javascript, Python, HTML5, CSS, SQL, SQLite3,
Frameworks/Technologies: React, Redux, Flask, Express, SQLAlchemy, Sequelize, PostgreSQL, AWS S3, Git, Node
Key Skills: Pair Programming, SCRUM development framework, Object-Oriented Programming (OOP)

PROJECTS

Bopify - Spotify Clone Live Site | Github

*Bopify is a music streaming service taking inspiration from Spotify using a Flask/SQLAlchemy Backend and a React/*JSX Front-end

- As a huge Spotify fan and a large consumer of music, I'm proud of the fact that I was able to recreate something I adore while also being challenged in my full-stack skill set. I am especially proud of the functionality of my project. It has *most* of the features Spotify offers such as playlist creation, music playback, adding music to playlists, and following users/playlists with more to come in the coming days.
- The biggest challenge I encountered while working on this project was creating the music player component that receives data from a redux store to keep track of what's currently playing and what's queued up. Styling the music player came easy but making it function as intended for it to work was tough. Redux made the process painless since I could always key into that ever-updating state to change what music is playing, especially when users can manually add songs to the ever-growing queue.
- In this project, I implemented an AWS S3 framework where users can upload images stored on their computer to my site when editing a playlist so they can give it their unique flair. Without this framework, users would need a URL on deck to upload a picture which makes the user experience a lot less intuitive. AWS S3 also cut down on storage on my local app since I can upload the seed data, such as songs, to my bucket and extract it from there, eliminating the need to store a large number of songs in my project's local files.

Enhance - Behance Clone Live Site | Github

Enhance is an art portfolio-sharing website built using a Python/SQLAlchemy Backend and a React/JSX Front-end

- In my first group project working with 3 other developers on creating this website, a huge challenge we had to overcome was communicating tasks that each individual should focus on. I took a leadership role to dole out roles and tasks to each one of us to keep everything moving as smoothly as possible. When a task or component was completed we would come together, and talk about what was done and how we feel about the work, and once everyone was satisfied the task would be marked as complete and the developer who completed it would get a new task to complete. It kept us going to complete each goal so we could finish our project in time while also keeping quality high
- In this project, we also used an AWS S3 framework to allow users to upload images. Since we were recreating a website where users share artwork they created it would make the most sense to have that functionality of letting users pick and choose what images they would like to display without having to upload them onto a third-party image hosting website. I believe having an AWS framework on a lot of projects would give users peace of mind knowing their files will be kept securely in a bucket that no one else can access other than the site admins.
- Going back to my key skills, this project showcased my ability to pair program exceptionally well with other developers. There were times when all of us were watching someone else's screen to figure out why a certain bug was occurring or what should go next when coding. Having another pair of eyes coding along with you allows you to make some mistakes while having another developer walk you through why that might be a mistake and what can be done to fix it or improve upon it. Programming solo, while it might be more "relaxing" doesn't give you the benefit of knowing any issue that might arise can be easily fixed by having another developer code along with you. I believe if we didn't emphasize pair programming during this project we would not have created a clean, UX-friendly website.

GroundBnB - AirBnB Clone Live Site | Github

GroundBnB is a vacation rental website where users can "rent" out their homes built using an Express/Sequelize Backend and a

React/JSX Front-end

- As my first ever full stack project I created, there came a whole host of challenges I had to overcome to get my project working as intended. Since I was working with React for the first time in a production setting, there were times where I was stuck as to why a certain component might not be working such as querying for a logged-in user that might not exist which would break my application. As I continued working and learning what I could online as well as the resources I was provided, it came to me like it was second nature. I eventually understood the workflow that React expected while also having conditionals to ensure a webpage will always load with no application-breaking bugs. As I finished the project I learned that working with React is extremely fun and makes creating HTML components painless compared to working strictly with HTML5 and the Document Object Model (DOM). This was also my first time working with Redux to centralize my application's state. Creating the reducers was a huge hurdle for me but when I got the idea of it it became straightforward, where each function in the store has a dedicated job to move data around and deliver it to the reducer.
- Since I was recreating a website like AirBnB I thought it would be a good idea to make it a bit more unique compared to just making it focus on houses. I instead chose to focus on making a cave rental website where you can book a time to spend a night (or as long as you please!) in a cave that users can host.
- After finishing this project, and even while working on it, I was immensely proud of the backend I created for it. Since this was my first full stack project, I wasn't confident with how routes should be laid out for an API or how to best create my tables, but once I started doing the actual work it began to come to me a lot easier than I expected. Sequelize makes creating tables and relationships painless while also making it easy to seed data for your project. When it came to querying data to send from my API to my Frontend, there were a lot of aggregate functions I needed to implement to send data I'd need in the future, such as getting an average number of all numbers in my database or finding the min or max. Seeing my database and Backend server come alive was a fantastic feeling, it felt like I was developing an integral part of my application while also challenging myself to put what I've learned into code.

EXPERIENCE

Sales Associate
AutoDirect LLC, Aug 2021 - May 2022

- Delivered up to five cars a day to customers while keeping track of certain info to make each experience more personable for the customer
- Acquired 60 reviews in the period which resulted in the company maintaining a 5-star status on Google
- Resolved customer complaints by communicating what it is that was wrong and trying my best to solve any issue that arises which led to an increase in customer retention and satisfaction

Grocery Clerk
Stop & Shop, Sep 2016 - April 2020

- Was the first choice when the company needed to train new employees resulting in an increase in productivity for myself and the new employee while keeping employee retention
- Came up with a way to better stock materials during the day while customers were shopping which resulted in an increase in efficiency for my department while keeping customers satisfied
- Trained in using an Electric Pallet Jack and was the first person called when they needed it used which triggered better task completion and speed for both the company and myself due to the work ethic I bring to the table

EDUCATION

App Academy - Immersive software development course with a focus on full-stack web development (Winter 2022)