Project Deliverables

This is a collection of my experiences building the apps on my page, as well as the struggles that I encountered while building these apps.

PokeDex:

The Pokedex project was a good project which introduced quite a bit of new Javascript concepts then I had experience using. The app consisted of using an external API to pull pokemon from a database and display them to the user with some of their statistics.

Tech Stack:

- HTML
- Bootstrap
- VanillaJS

Some of the problems that I overcame with this project, was that it was the first time that I had interacted with the DOM in more complex ways using JS. This was a struggle for me at first, when learning to create modals, as well as making sure that I was displaying the data in the way that I wanted to display it.

Overall, this was a good learning experience of how to build something from scratch with no framework or library, only using bootstrap to help with normalized styling.

Movie Findr API:

The movie Findr API was my first attempt at programming something that would interact with the back-end of an application. The whole process of building this project involved building a small database with MongoDB, and then using NodeJS as a base for building an API that would then interact with that database.

Tech Stack:

- NodeJS
- Express
- Passport
- MongoDB

I found that I encountered many issues with this project that I had to overcome. This was my first experience entering the back-end realm of programming, so it was challenging to understand some of the interactions, and how to perform operations on my database using Javascript in Node.

One problem that I found that I overcame with this project was returning a JWT to the user of the application to authenticate their actions using the API. Passport was challenging for me to use at first, but after reading the documentation and understanding a little further I felt as if I had a more solid grasp and understood when, and how to send these JWT tokens with requests.

Movie Findr Client (React):

The movie Findr Client that I built in React was to provide a front-end to interact with the API that I had previously just built. This project was built in React and was my first time experimenting with building a project in React. This was a good experience when building with React as I feel I got a good grasp on components, state management, props, and more.

Tech Stack:

- React
- React Bootstrap

I had many challenges with this project, but I specifically struggled understanding different React hooks at first, such as useEffect. This was the most challenging part of the project for me, making sure that I was able to understand when to use this hook, and why we would want to use it in some scenarios. After gaining a better grasp on hooks and reading more thoroughly into scenarios where they may be used, I feel as if I overcame this confusion.

Movie Findr Client(Angular):

The movie Findr Client was a rebuild of the React Movie Findr Client within Angular instead of React. This served as an introduction to Angular, since it is another popular framework. This project was a great learning experience, as I was left up to a lot of interpretation, and had to read the Angular docs to fully wrap my head around most concepts that I used in the project.

Tech Stack:

- Angular
- Angular Material
- TypeScript

The hardest challenge that I faced within this project was getting used to using TypeScript instead of just plain Javascript. Mainly, the challenge behind this was figuring out when it was optimal to give variables a type or use any type. This was a good challenge, and after experimenting with my code, I figured out the trick to controlling the outcome of my functions, etc by controlling the type of variables. I also thought that this was super clear, and concise in how the code actually read, being as you could interpret the type of input the function needed to take as well as the possible return from the function due to the type of it's arguments.

Meet App:

The meet app was a first introduction into serverless functions and programming. I believe that this project helped me wrap my head around how using serverless for my applications could actually help improve performance and make a difference in the end product of development.

Tech Stack:

- React
- Axios
- AWS Lambda
- 0Auth

The biggest challenge for me in this application was not necessarily any of the code for the front end of our application, but instead was actually just setting up the serverless functions instead. Setting up the serverless functions to work for my authentication was the hardest part for me as I was confused at first by the process in which I could actually use AWS Lambda to communicate with google for my 0Auth process. To learn this and help alleviate some of this challenge, I watched quite a few Youtube videos which broke down how AWS Lambda functions actually interact with code that we write and read up over the actual "lecture" content again.

Chat App:

The Chat App served as an introduction to programming in React Native, and building Native applications in general. This was something I was greatly looking forward to as I wanted to be able to build mobile apps, and see how my applications could interact with the OS of a mobile device.

Tech Stack:

- React Native
- Firebase
- Gifted Chat
- Expo

The bigger challenges I faced within this project was actually getting my application to update in real time with messages that were sent. After debugging for a long while along with reading up on material on the topic again I was able to figure it out.

Another challenge I faced was sending pictures to other users. I was consistently getting errors when sending pictures at first, and had to re-read example code many times as well as debug my own to be able to get an idea of how I could actually make this work correctly.