Brandon Minjares

209-535-1750 | bmminjar@ucsc.edu | linkedin.com/in/brandonminjares | github.com/brandonminjares

Education

University of California, Santa Cruz Dec 2022
Bachelor's of Computer Science

Relevant Coursework: Data Structures and Algorithms, Full-Stack Web Development, Database Systems, Algorithm Analysis, Programming Abstractions in Python, Applied Machine Learning

Experience

Webbege | *WordPress Developer* July 2018 – Feb 2020

- Developed and maintained 4 web applications using WordPress, JavaScript, HTML, CSS.
- Collaborated with customers to achieve key marketing objectives.
- Incorporated digital marketing tools Google Analytics and HubSpot.

Projects

Filmbot | *ReactJS, JavaScript, Python, MongoDB*

- Developed a full-stack web application with **Express** serving a **REST API** with **React JS** as the frontend.
- Implemented **User-based Collaborative Filtering** in **Python** to provide film recommendations.
- Utilized **The Movie Database API** to **fetch** then display information about films.
- Started YouTube channel **Filmbot** to market the application.

Bam! | *ReactJS, JavaScript, MongoDB*

- Built peer-to-peer mobile web application to **transfer files by bumping mobile devices**.
- Leveraged **WebTorrent API** w/ GPS and accelerometer data to establish secure connection between clients.
- Application creates **private and secure file transfer** by not requiring the user's phone number or email.
- Lead a team of 4 using **Agile methodology** principles.

Mailhub | *ReactJS, JavaScript, Postgres, Material UI*

- Created a **web-mail client** that is a clone of Gmail where users can send, reply to, and delete mail.
- **Search function** allows for quick access to emails using **Regex** to find and highlight keywords.
- Clean user-interface using **Material UI** allows for seamless and fun navigation.
- Implemented a full end-to-end testing suite with **Jest** testing framework.

Sorting Algorithm Visualizer | *ReactJS, JavaScript*

- Designed **visual representation of 4 Sorting algorithms** to demonstrate sorting times.
- User can sort 1 algorithm or have **4 algorithms sort simultaneously** on the same randomly generated array.
- **Clock speed** is added to each algorithm to **compare runtime complexity** for different array sizes.
- Implemented a full testing suite with **Jest** testing framework to ensure the array was sorted correctly.

Animal Classifier | *Python*

- Developed classifier using **Convolutional Neural Network** from 10,000 animal images.
- Incorporated **VGG model** using **transfer learning** to improve model accuracy.
- Model achieved **97% accuracy** making it the highest out of 100 participants.

Programming and Tools

Languages: JavaScript, Python, SQL(Postgres), NoSQL(MongoDB), HTML, CSS
Libraries/Frameworks: ReactJS, Express, Material UI
Developer Tools: Git, Jest, Docker, VS Code