

# SHAWN BOYLE

(801) 541-6884

[shawn.boyle01@gmail.com](mailto:shawn.boyle01@gmail.com)

[PORTFOLIO](#)

[GITHUB](#)

[LINKEDIN](#)

## SKILLS

JavaScript / Python / React / Redux / Node.js / Flask / Express / SQLAlchemy / HTML5 / CSS3 / PostgreSQL / Docker

## RELEVANT WORK EXPERIENCE

### Full Stack Software Engineer | Flapmax (Contract)

Nov 2021 - Dec 2021

- Engineered optimization for trained AI Image Classification models, using **FastAPI** for **RESTful** routing and model file validation, then improving throughput, latency and memory footprint without affecting accuracy.
- Constructed an intuitive, comprehensive front-end UI through **React**, allowing clients to upload and optimize models, then displaying data through rendered charts and statistics, providing clients with clarifying data and an enhanced product.
- Integrated **Azure cloud storage**, using **Postgres** database and **Blob** services for **SQL** and **NoSQL** storage on the cloud, providing data management with either relational or non-relational storage.

### L1 Technical Support Engineer | Instructure

May 2020 - Dec 2020

- Assisted students, teachers, and admins over the phone or live chat to resolve **80+ cases per 20 hour weeks** on the **Canvas** platform, by navigating a thorough troubleshooting process centralized around known bugs and common issues.
- Utilized several **Canvas** technologies and extensions to parse student activity data in **JSON**, replicate existing bugged environments for **testing and debugging**, and assist in implementing external API's into our system for teachers' utilization.
- Demonstrated strong communication with the team weekly by discussing goals and milestones and utilizing feedback to maximize performance.

## TECHNICAL PROJECT EXPERIENCE

### PROGRAMMATIC | Group Project built with React, Redux, Flask-SQLAlchemy, AWS S3, PostgreSQL, Docker

[Live Site](#) | [GitHub](#)

- Engineered a visualization of **Dijkstra's Algorithm** by calculating the shortest path for a weighted graph of nodes, rendering the graph and edges between the nodes with a **CSS** grid and **React**, allowing for an animation of the traversal and shortest path for an enhanced **UI/UX**.
- Programmed a fluent search algorithm with **React** to match user queries with comprehensive filtering, returning results utilizing the **useLocation** hook for accurate DOM rendering.
- Developed a seamless, responsive discussion section for users to read feedback as well as engage in discussion by creating, editing, or deleting their comments utilizing **React** / **Redux** to manipulate the front-end dynamically and efficiently.

### ZOOLODEX | Solo Project using React, Redux, Flask-SQLAlchemy, AWS S3, PostgreSQL, Docker

[Live Site](#) | [GitHub](#)

- Integrated **Google Maps API** for users to explore geographical locations, having the coordinates be associated with seeded regions, soon allowing users to upload and view events with markers on the map.
- Designed a responsive navigation bar utilizing **React** and **CSS** to adjust styling conditionally when the Y coordinate scroll amount of the window object surpasses page image headers, providing a visually appealing **UI/UX**.
- Constructed a fluent backend **Flask-SQLAlchemy** server, handling comprehensive validations for user authentication and **AWS S3** uploads, then providing that data to the **PostgreSQL** database, wrapping all data for deployment with **Docker**.

### CRESCENDO | Solo Project utilizing React, Redux, Node.js, Express, Sequelize, AWS S3, PostgreSQL

[Live Site](#) | [GitHub](#)

- Developed a relational database with 12 tables held by **PostgreSQL**, which I queried on the **Node.js** backend's **Express** server with the **Sequelize ORM**, allowing server data to have clarity and accessibility.
- Utilized the audio element to construct a feature-rich audio player, maintaining audio controls throughout the website with dynamic progress tracking.
- Formed a cohesive **Redux** store through efficient **API** queries, cutting down on server load and resulting in a quick, optimal load time for data rendered on the frontend with **React**.

## EDUCATION

### App Academy | Full- Stack Web Development

Oct 2021

- Rigorous 1000-hour software development course with <3% acceptance rate
- Topics include: **TDD (Test Driven Development)**, scalability, algorithms, **OOP (Object Oriented Programming)**, coding style, **REST**, security, single-page apps, pair programming, design patterns, and web development/coding best practices.