

TECHNICAL SKILLS

- **Front-End:** JavaScript(ES5 & ES6), TypeScript, React, Redux, Vue.js, jQuery, Bootstrap, Bulma, D3, Chart.js, HTML5, CSS3
- **Back-End:** Node.js, Express, Python, Java, GraphQL, Apollo, MongoDB, MySQL, Postgres, Redis, RESTful API
- **Testing/Development:** Mocha, Chai, Jest, AWS EC2, Docker, Heroku
- **Developer Tools:** Git, npm, Webpack, Babel, pgAdmin, Agile Methods, TDD

ENGINEERING & APPLICATIONS

Anchor Technologies

April 2020 - Sept. 2020

Full-Stack Engineer | **React, Apollo, GraphQL, D3, Postgres**

Interactive dashboard of a user's cybersecurity posture, offering ongoing risk management and remediation planning

- Consistently met product design requirements within scheduled deadlines while adhering to Agile development methods
- Leveraged GraphQL and Apollo to fetch/update customer data, that was displayed across interactive D3 models
- Provided input to CEO on functional requirements, product designs, schedules, and technical assistance

forever404 | Front-End Engineer | **React, Redux, Bootstrap, Node, Express**

Github: github.com/Project-Greenfield

Multi-feature e-commerce product page

- Developed purpose-built image gallery component, using React-Bootstrap and custom JS to support hover-to-zoom functionality, scrollable thumbnail photostrip, and toggleable viewing modes
- Designed modular and reusable React components, utilizing React Hooks, for re-use across application
- Operated under Agile Methods and Test Driven Development in a team of four engineers, each owning a unique component, to develop a coordinated front end

System Design | Back-End Engineer | **Postgres, Redis, Docker, Kubernetes, AWS EC2**

Github: github.com/Team-Quekcaya/qa-service

Replaced an existing API with a back end system that can scale to meet the demands of production traffic.

- Stress tested the service with Loader.io throughout development, to detect possible bottlenecks and measure optimizations
- Cached recent database queries in Redis, resulting in decreased latency, increased throughput and less of a load on the database
- Scaled the service by configuring a Kubernetes cluster and load balancer to distribute incoming traffic across multiple virtual machine instances, leading to a 200% increase in successful client requests per second

Variant | Front-End Engineer | **React, Node, Spotify Developer API, Bootstrap**

Site: <https://variant-mus.herokuapp.com> **Github:** github.com/mbrown98/variant

Playlist Generator connected to User's Spotify Account that curates playlist based on user inputs

- Implemented OAuth to allow the application to interact directly with/edit a user's Spotify account based on requested scopes
- Used Spotify API to generate an empty user playlist, populate it with recommended songs, and then retrieve that playlist to display in an iframe that played generated songs

EDUCATION

Hack Reactor | Advanced Software Engineering Immersive Program

2019 - 2020

University of Maryland | Economics (B.A.) and Political Science (B.A.)

2016 - 2019