916-276-9852
anthonyjfojas@gmail.com

ANTHONY FOJAS

Software Engineer | San Francisco Bay Area

github.com/ajfojas
linkedin.com/in/ajfojas

TECHNICAL SKILLS

Front End: JavaScript | React | React Hooks | React Context API | Styled-Components | HTML | CSS
Back End: Node.js | Express | MySQL | PostgreSQL | MongoDB | Mongoose | Redis
Testing/Analytic Technologies: Mocha | Chai | Jest | Enzyme | New Relic | Loader.io | Artillery.io | Postman
Deployment Technologies: Amazon Web Services (AWS) EC2 | Docker
Other Technologies: NGINX | Webpack | Babel | CircleCI | Git | Github | Vim | Nano | Trello
Methodologies: Agile Development | Scrum & Kanban Practices

FULL STACK PROJECTS

MINC TECH - *Reviews List Microservice for Lodging and Hotel Web Application* (**Live Site**)
React, Styled-Components, Node.js, Express, MySQL, Redis, Jest, Enzyme, AWS EC2, Docker, Webpack, Babel, CircleCI

- Optimized page load speeds by identifying bottlenecks with Google PageSpeed Insights analytics and enabling Express compression middleware, leading to a PageSpeed score of 96
- Enabled data caching utilizing Redis with LRU eviction policy to improve page performance and load speeds
- Containerized microservice with Docker for runtime consistency and ease of deployment on AWS EC2 t2.micro instance, adhering to microservices architecture design
- Implemented pagination in React to promote a streamlined UI and UX across 100+ different listings with varying amounts of reviews
- Applied TDD practices, achieving >90% code coverage with Jest and Enzyme unit and integration tests

DROP GRAV - *System Design of Photo Gallery Microservice*
Node.js, Express, PostgreSQL, New Relic, Loader.io, Artillery.io, Postman, AWS EC2, NGINX, Trello

- Horizontally scaled back end infrastructure by directing web traffic through an NGINX load balancer and across 4 EC2 server instances, increasing the maximum number of requests per second (RPS) handled from 250 to 1000
- Optimized read query times for PostgreSQL database holding 100+ million records via B-tree indexing, reducing overall latency from over 120 sec to under 2 ms
- Stress tested with Loader.io and analyzed server performance metrics with New Relic, pinpointing bottlenecks affecting RPS, latency, and error rates

PACK RAT - *Magic: The Gathering Card Collection Tracker* (**Live Site**)
React, React Hooks, React Context API, Styled-Components, Node.js, Express, PostgreSQL, Webpack, Babel

- Designed a modular and user-friendly front end with React for component separation and conditional rendering
- Handled global application state with React Hooks and Context API, leading to a simpler state management system
- Implemented RESTful API routes in Express to bridge data communication between client and Postgres database

EDUCATION

Hack Reactor - *Advanced Software Engineering Immersive* | *San Francisco, CA* 2019
University of California, Davis - *Bachelor of Science, Biochemistry & Molecular Biology* | *Davis, CA* 2017

WORK EXPERIENCE

UC Davis Department of Plant Sciences - *Junior Plant Scientist* | *Davis, CA* 2017 - 2019

- Conducted plant reproductive biology research in collaboration with Mars, Inc. to gain a better understanding of seed formation in Arabidopsis thaliana
- Supervised student workers and provided training in good laboratory practices, techniques, and safety procedures
- Iteratively coordinated future goals with my team on a bi-weekly schedule