Joey Faris

Los Angeles, CA | 949-945-8025 | joeyfaris12@gmail.com | LinkedIn | Website | Github

SOFTWARE ENGINEER

Highly skilled software developer with a strong foundation in full-stack development, experienced in building and deploying scalable applications using JavaScript, React, Node.js, and AWS. Adept at collaborating in Agile environments, with expertise in front-end development, RESTful API design, cloud infrastructure, and containerization. Always expanding technical expertise and striving to grow as an engineer to stay at the forefront of the quickly evolving industry.

TECHNICAL SKILLS

Javascript, React, Redux, HTML, CSS, Tailwind, Material UI, Node, Python, Java, SQL, PostgreSQL, MongoDB, REST APIs, Unit Testing, Docker, Git, Github, BitBucket, AWS (EC2, S3, Lambda), pgAdmin, JIRA, VSCode, Postman, Insomnia

EDUCATION

California Polytechnic State University, San Luis Obispo — San Luis Obispo, CA
Bachelor's Degree in Business Administration — September 2015 - June 2019

Flatiron School — Online
Full Stack Web Development, Javascript and Ruby on Rails program — June 2022 - September 2022

EXPERIENCE

Pycube, Inc. — Remote
Software Engineer — November 2022 - Present

- Developed front-end applications with JavaScript, React, Redux, HTML, and CSS for engaging user interfaces.
- Implemented server-side logic and RESTful APIs using Node.js, Express.js, PostgreSQL, and MongoDB for efficient data management.
- Containerized applications with Docker for consistent development and deployment environments.
- Managed version control with Git and collaborated on codebases using BitBucket for efficient code review.
- Deployed applications on AWS infrastructure (EC2, S3, Lambda) for scalable and reliable cloud hosting.
- Leveraged Redux Toolkit (RTK) and RTK Query for centralized state management, optimized data fetching, and automatic cache invalidation, reducing boilerplate code and improving application performance.

MonteFiore Medical Center (Pycube, Inc. Contract) — June 2023 - Present

- Implemented an invoice validation service using a five-step processing workflow using React hooks and Redux.
- Collaborated closely with graphic designers to translate their Figma designs into interactive and visually appealing front-end implementations.
- Designed and implemented API endpoints using Express.js, applying the MVC design pattern for structured layer-wise functionality.
- Integrated SharePoint APIs with OAuth for online excel correction.
- Deployed and managed applications on AWS, utilizing services such as EC2, S3, and Lambda for scalable and reliable cloud infrastructure.
- Collaborated with the DevOps team to design and configure the CI/CD pipeline, streamlining the development process.
- Participated in a Scrum team and Agile Practices including: Test-Driven Development (TDD), Behavior-Driven Development (BDD) and pair programming.

Baylor Scott & White Health (Pycube, Inc. Contract) — November 2022 - June 2023

- Implemented metrics (Asset Usage, Asset Cycles, Asset Path, Asset Inventory) using MongoDB aggregation pipeline and Express.js for API development.
- Successfully implemented dashboard components using React hooks for effective data visualization and user interaction.
- Optimized JavaScript code for faster load times and smoother user experiences, employing techniques such as lazy loading, code splitting, and minimizing network requests.
- Utilized JIRA for project management, task tracking, and issue resolution, resulting in streamlined workflows and enhanced productivity within our team.
- Used Docker to define and create applications by encapsulating them in containers.
- Presented weekly progress reports and demos to managers and C-level personnel, providing updates on development progress and collaborating with them to craft future roadmap and address any feedback they had about the product.

Projects

Docker Optimizer Github

- Developed a CLI tool in Python that analyzes Docker image resource usage, helping developers identify and optimize container size by detecting unused files and dependencies.
- Implemented real-time container analysis using Docker SDK and system call tracing (dtrace/strace), providing insights into file system usage patterns across different operating systems.
- Implemented multi-stage build optimization and layer analysis to minimize container bloat, focusing on reducing cold start times in cloud environments and CI/CD pipeline efficiency.
- Designed an automated optimization pipeline to reduce Docker image sizes by intelligently removing unused files and dependencies, targeting up to 50% reduction in image size and improved deployment efficiency.

Scothing (In Progress) Github

- Developed a clothing classification system using deep learning and PyTorch, classifying images into 10 categories through a REST API.
- Created a FastAPI service for real-time clothing analysis with image processing and quick model predictions.
- Designed a flexible training setup with metrics like accuracy and precision, optimized for both CPU and GPU use.

Facetracker Github

- Developed a real-time face and hand landmark detection application using Python, OpenCV, and MediaPipe, enhancing user experience with accurate facial and hand tracking features.
- Implemented a model download and validation system that verifies the presence of face and hand landmark models, ensuring smooth setup and efficient resource management.
- Integrated FPS (frames per second) display functionality to monitor real-time processing performance, optimizing user experience with seamless visual feedback.
- Built cross-platform compatibility checks for camera permissions, improving accessibility across macOS, Windows, and Linux operating systems.

Twitter / X Bot Link | Github

- Engineered sophisticated prompt templates to generate technically accurate content across multiple domains (AI, Quantum Mechanics, Software Engineering).
- Implemented chain-of-thought prompting techniques to ensure logical consistency and depth in generated technical explanations.
- Developed content filtering systems to validate LLM outputs against domain-specific technical criteria and terminology.
- Created prompt frameworks that maintain consistent voice and style while discussing complex technical concepts.

Color Palette Generator Link | Github

- Built a full-stack application combining Python backend for color analysis with a responsive React frontend.
- Implemented advanced color theory algorithms using colormath library to generate harmonious color combinations.
- Designed RESTful API endpoints handling image upload, processing, and palette generation with error handling.
- Created an intuitive UI that allows users to manipulate and export generated color palettes in multiple formats.