

# Kevin Cao

☎ +1-(925)-457-8339 ✉ [kevincao558@gmail.com](mailto:kevincao558@gmail.com) [in kevincao](#) [Q kevincao](#)

## EDUCATION

**University of California, Riverside**

*Bachelors of Science in Computer Science + Business*

Riverside, CA

*September 2021 – June 2025*

## EXPERIENCE

**ACM @ UCR**

December 2023 – June 2025

*Frontend Engineer*

*Riverside, CA*

- Worked with **10+** engineers and designers to launch a student resource portal, streamlining event visibility and increasing engagement among **100+** monthly users.
- Engineered functional and reusable React components from Figma design specifications, ensuring high-fidelity implementation and adherence to UI/UX standards.
- Integrated Google Calendar API for real-time syncing, reducing scheduling errors by **80%** and improving attendance coordination.

**Headstarter**

June 2024 – September 2024

*Software Engineering Fellow*

*Remote*

- Engineered **CI/CD** pipelines using **GitHub Actions** to automate testing and deployment processes, accelerating release cycles and improving code quality.
- Conducted **20+** code reviews, reducing production bugs by **30%** and improving maintainability.
- Implemented secure payment processing with **Stripe API**, enabling subscription-based transactions with tokenized card storage and webhook-based invoice reconciliation.
- Refactored backend endpoints for JSON compression and response size reduction, improving average load times by **25%**.

## PROJECTS

**DevLink** | *Go, Python, SQLite, Docker, AWS(SES)*

**GitHub**

- Delivered job alerts to **1,000+** simulated users in real time by designing a distributed microservice system using **Go** and **Kafka**, enabling scalable, asynchronous event processing.
- Ingested and normalized **3K+** job postings from external APIs and RSS feeds via scheduled Go workers, publishing events to Kafka for real-time processing.
- Ensured **99.99%** reliable alert delivery by designing a fault-tolerant Kafka consumer pipeline that queues messages and sends emails through **AWS SES**.

**MatchaNotes** | *NextJS, TypeScript, PostgreSQL, Redis*

**GitHub**

- Implemented **Redis** caching for high-traffic product queries with automatic TTL-based invalidation, decreasing API latency by **120ms** and offloading **40%** of database requests.
- Centralized product query logic into reusable Next.js server actions, reducing front-end code duplication and simplifying data flow across pages
- Reduced filtered query response time by **45%** by adding B-tree indexes to frequently queried PostgreSQL fields.

**StyleSwiper** | *NextJS, TypeScript, Firebase, Tailwind-CSS*

**GitHub**

- Minimized critical bugs and regressions by **15%** through comprehensive unit testing with **Jest**, significantly improving software stability and reducing post-deployment issues.
- Built secure multi-method user authentication using **Firebase** (OAuth, JWT, email/password), protecting user data and ensuring seamless access control.
- Integrated dynamic filtering and sorting features for personalized outfit selections, enhancing UI responsiveness by **30%**.

## TECHNICAL SKILLS

**Languages:** Go, TypeScript, JavaScript, Python, C++, Java, HTML/CSS

**Frameworks:** React, Node.js, Next.js, Express

**Tools:** Docker, Git, GitHub, GitHub Actions, Postman, Jira

**Testing/CI:** Jenkins, Jest, Selenium

**Databases/Cloud:** Amazon Web Services(AWS), SQL, PostgreSQL, MongoDB, Firestore, Redis