

Kevin Chen

SOFTWARE ENGINEER · FULL-STACK

US Citizen

✉ kevinmichaelchen@gmail.com | 🏠 kchen.io | 📧 kevinmichaelchen | 🌐 kevinchen93 | 🐦 @kevinmchen

Summary

I'm a full-stack generalist (since 2015). I want to use technology to help people.

Work Experience

The Drivers Cooperative

Remote

SOFTWARE ENGINEER

Mar. 2022 - Current

- Implemented a dispatch ranking service in Go using H3 (hexagonal hierarchical geospatial indexing).
- Boosted performance by 93% using event sourcing, Postgres triggers, and denormalization (wide tables).
- Built an admin UI for dispatchers using React, Typescript, Material UI, Google Maps, Hasura, and GraphQL.

Care.com

Remote

SOFTWARE ENGINEER II

Mar. 2021 - Mar. 2022

- Productionized components in the supply-side of a two-sided care-giving/seeking marketplace, built on Golang, AWS, DynamoDB, Lambdas, Kafka, EventBridge, gRPC, protobuf, Splunk, OpenTelemetry, and more.
- Enhanced data models in provider ingestion pipeline, allowing for onboarding of senior-care centers.
- Designed a pipeline (using S3, Lambda, and Kafka) to enrich Elasticsearch records with AI-generated provider rankings, improving search experience.
- Implemented a component to capture auto-accept leads, yielding greater revenue and lead conversion. (Used Lambda + DynamoDB TTLs).
- Designed and implemented a system to periodically recalibrate Google Ads geo-targeting, reducing costs and improving end-user ad experience. (Used EventBridge, Lambda, S3, gRPC, and Elasticsearch).

IrisVR, Inc.

New York, NY

SOFTWARE ENGINEER II

Oct. 2018 - Mar. 2021

- Designed and implemented the "Content Library" back-end that allowed customers' wireless headsets to sync large 3D CAD files up to the Cloud. (Used Golang, gRPC, Protocol Buffers, Cloud Firestore, and Cloud Storage.)
- Designed and implemented the back-end for Issue Tracking. (Used Go, sqlboiler ORM, Postgres, GraphQL, and Protocol Buffers.)
- Created a monorepo for org-wide Protocol Buffers (API + data contracts).
- Implemented observability (o11y) best practices for debugging and hotspot detection.
- Improved performance (up to 75% in some cases) using Redis caching, batched gRPC calls, and parallelized database (Firestore) queries.
- Boosted Lighthouse scores from 50% to 90% by porting irisvr.com to GatsbyJS.

Tesla Government, Inc.

Tysons Corner, VA

SOFTWARE ENGINEER

Jun. 2015 - Sept. 2018

- Transitioned a monolithic Spring app to a microservices architecture.
- Built a CI/CD pipeline (using yum repos, RPMs, shared Jenkinsfile logic, scripts for SemVer auto-tagging, Bintray artifact publishing, and externalized configurations (window objects) for our frontends).
- Shipped a Bookmarks service in Golang that notified subscribed users of CMS page updates.
- Wrote Go and Java SDKs for initializing AWS SQS queues and SNS topics.
- Worked on an identity service built on Spring Boot, React, Redux, and redux-saga.

Williams College

Williamstown, MA

SOFTWARE ENGINEER

Sept. 2011 - Jun. 2015

- Developed a JSON API for the student website, Williams Students Online, which provides widely consumed services, such as review systems for professors, courses, and dorms. Built on Ruby on Rails, MySQL, Javascript.
- Developed a Ruby on Rails JSON API for a student-run hackathon group, which ended up securing a \$15,000 prize. The accompanying Android app would let potential users purchase tickets to entertainment events.
- Interned for Julintani, LLC and worked on Android apps that displayed dining hall menus and bus routes around your college. Developed a shared Java library. Updated networking operations to latest Android standards. Wrote a Ruby script to crawl Peoplesoft and extract course data, and a Ruby on Rails JSON API to serve that data.
- "Computational Identification of Individual Spotted Salamanders": Built a Java Swing-based GUI that allowed biologists to determine if they had seen a particular salamander individual before. The UI consisted of three phases: the user uploads photographs of wild salamanders; they straighten their spines; a list of likely "matches" is displayed for a given salamander.