

Edwin Schleif, Software Engineer

Pleasant Hill, CA · edwin@edwincloud.com · (707) 317-1773 · edwincloud.com

Work Experience

Goplim · *goplim.com* · **Software Engineer** · San Francisco Bay Area · Mar 2026 to Present

Built Goplim end to end as an independent full-stack project: a voice-first learn-by-repetition platform with multi-device live session sync, AI-assisted content, and a creator marketplace with Stripe subscriptions.

- Architected as microservices: a Go API and a TypeScript MCP server, containerized via Docker Compose with Nginx reverse proxy on Linux production.
- Voice-first learning loop where users speak answers out loud and get AI-graded feedback.
- Multi-device live session sync across web, Android, and iOS via a custom WebSocket hub.
- Creator marketplace with Stripe Checkout, webhook reconciliation, and 80/20 revenue split for creators.
- Public share-link playback so any recorded session can be replayed without login.
- Multi-mode personality feedback (Grill mode, Show Love mode) with adjustable intensity.
- AI subsystem: Gemini for question generation and AI breakdown, Whisper for voice transcription, TTS for spoken playback.
- TypeScript MCP server exposing DB-backed tools to AI clients.

Stack: Microservices, Go, Gin, PostgreSQL, WebSockets, Next.js, React, TypeScript, Tailwind, Kotlin, Jetpack Compose, Swift, SwiftUI, Docker, Docker Compose, Nginx, Linux, MCP, Stripe, Gemini, Whisper, TTS

Curtis Instruments · *EV company* · **Staff Software Engineer** · Livermore, CA · Jun 2021 to Mar 2026

- Designed, built, and maintained Windows desktop applications to interface motor controllers through CANopen.
 - Written in C#, WPF, and Windows Forms on .NET.
 - Built the CANopen communication layer used to read and write device data in real time.
 - Built the MQTT communication layer for real-time messaging between the apps and the server.
 - Supported embedded software engineers as the end users, debugging issues in the apps and on the CAN bus.
- Designed, built, and maintained the Curtis external and internal web portals end to end.
 - Frontend originally written in React as a single-page application, later migrated to Next.js with TypeScript.
 - Backend REST APIs originally written in .NET Web API on Linux, later migrated to Go with Gin, persisting to PostgreSQL.
 - Features include software licensing, Stripe integration, document management, and role-based access.
- Architected the portal backend as a set of independently deployable Go microservices.

- Separate services for authentication, licensing, Stripe billing, document management, and notifications.
- Each service owns its own data and exposes a versioned REST API.
- Deployed independently to Kubernetes through the Jenkins CI/CD pipelines.
- Wrote CLI tools in C# and Go for engineers to communicate with CANopen devices and manage licenses, services, and resources.
- Owned the Jenkins CI/CD pipelines that built the desktop apps and deployed containerized backend services to Kubernetes.

Stack: C#, .NET, ASP.NET Web API, WPF, Windows Forms, Linux, Go, Gin, Microservices, REST APIs, Next.js, React, TypeScript, PostgreSQL, SQL Server, Docker, Kubernetes, Jenkins, Stripe, CANopen, MQTT

First Republic Bank · *Private bank* · Senior Software Engineer · San Francisco, CA · Oct 2020 to Jun 2021

- Maintained the enterprise CRM platform used across all departments of the bank.
 - Migrated the frontend to React with TypeScript.
 - Built and modified production React components and pages end to end.
- Built and maintained .NET backend services for the CRM and related internal tools.
 - Services written in C# on .NET / ASP.NET, persisting to SQL Server.
 - Migrated classic ASMX web services to RESTful ASP.NET Web API.
 - Cloud workflows on Azure with Azure Functions and Azure SQL.
- Production support across departments.

Stack: C#, .NET, ASP.NET, ASP.NET Web API, ASMX, React, TypeScript, JavaScript, SQL Server, Azure, Azure Functions, Azure SQL

Jefferies Financial Group · *Investment bank* · Senior Software Engineer · New York, NY · Feb 2019 to Sep 2020

- Led feature development, maintenance, and bug support of Navigator, the primary desktop platform for prime brokerage clients to monitor portfolios, positions, and P&L in real time.
- Integrated real-time pricing feeds from Bloomberg, Reuters, eSignal, and ICE Data Services.
- Contributed to J-Prime, the web-based portal for portfolio access and reporting, leading the frontend migration to a modern framework.
- Built backend services for position tracking and risk analytics.

Stack: C#, .NET, WPF, DevExpress, Angular, ASP.NET, SQL Server, Excel RTD, Bloomberg API, VSTO

Mizuho Bank · *Corporate bank* · Senior Software Engineer · Jersey City, NJ · Jan 2016 to Jan 2019

- Developed and maintained the statement generation and distribution platform for institutional clients.
- Built Windows services and console tools for automated processing.
- Built desktop utilities for internal operations and reporting.

Stack: C#, VB.NET, .NET, Windows Forms, SQL Server, SSIS, Python, TypeScript

Pronology · *Broadcast tech* · **Senior Software Engineer** · New York, NY · Apr 2014 to Jan 2016

- Developed time-code-driven media asset management interfaces for broadcast recording workflows.
- Built core services and console applications to integrate broadcast recording hardware.

Stack: C#, WPF, .NET, WCF, SQL Server, Windows Services

Safra National Bank of New York · *Private bank* · **Senior Software Engineer** · New York, NY · Apr 2007 to Mar 2014

- Developed, maintained, and supported the broker system for trading stocks, futures, and options.
- Integrated the trading platform with banking departments including client services, credit, operations, and middle office.
- Enhanced Internet Banking and middle/back-office applications.
- Built desktop and web interfaces for internal operations and reporting.

Stack: C#, VB.NET, VB6, .NET, ASP.NET Web Forms, ASP.NET MVC, Windows Forms, SQL Server, WCF

Skills & Technologies

Languages: C#, Go, TypeScript, JavaScript, Python, Java, Kotlin, VB.NET

Frontend: React, Next.js, WPF, Angular, ASP.NET Web Forms, Tailwind CSS

Backend: .NET / ASP.NET Core, Gin, Kotlin, Express.js, REST APIs, WCF

Mobile: Android, Kotlin, Xamarin

Cloud: AWS (EC2, RDS, S3, CloudFront, VPC), Azure (DevOps, App Services, Azure SQL, Blob Storage, Service Bus)

Infrastructure: Kubernetes, Docker, Nginx, Linux (Ubuntu), Jenkins

Data: PostgreSQL, SQL Server, Redis, Kafka, MQTT

DevOps: Jenkins, Azure DevOps Pipelines, CI/CD, automated builds and deployment

Scrum: Jira, Azure DevOps Boards, Polarion

Education

Computer Science, UGF (Universidade Gama Filho), Rio de Janeiro, Brazil (2006)