Kevin Chen

(585) 797-5153 | kc681269@gmail.com | https://www.linkedin.com/in/k3vnc/ |

Education

Rochester Institute of Technology Rochester, NY

Bachelor of Science in Software Engineering

- **Cumulative GPA: 3.13**
- **Dean's List: Spring 2023**
- **Relevant Courses:** Eng Cloud Software Systems, Software Testing, Engineering of Enterprise Software Systems, Engineering of Software Subsystems (Embedded), Software Process & Project Management, Web Engineering, Software Development and Problem Solving 1 & 2 (Python, Java, Git)

Technical Skills

Languages: JavaScript, TypeScript, Python, Java, C, C++, SQL.
Frontend: HTML, CSS, React.js, Next.js, Tailwind CSS.
Backend: Node.js, Express.js, REST APIs, OAuth 2.0, JWT.
Databases: MongoDB, PostgreSQL.
Dev Tools: Git, GitHub, Docker, VS Code, Postman, cURL, Vim, Jest.
Cloud & Infra: AWS (Lambda, EventBridge, SNS, Comprehend), Terraform.
Other: STM32Cube, Selenium, JSON, XML.

Projects

Stock Sentiment Tracker | *AWS (Lambda, EventBridge, SNS, Comprehend), Terraform, Boto3, Python* 2025

- Collaborated in a team of four to build a sentiment analysis platform for stock discussions using AWS cloud services.
- Developed AWS Lambda functions using Python and Boto3 to query DynamoDB and analyze stock sentiment with Amazon Comprehend, returning structured scores for frontend display.
- Configured EventBridge rules and Lambda triggers to send daily email alerts via SNS for subscribed watchlist tickers.
- Wrote Terraform code to provision infrastructure (EventBridge, Lambda, SNS, IAM roles), excluding DynamoDB.
- Ensured efficient sentiment analysis by skipping entries with existing scores and conditionally invoking Comprehend.

Enterprise Application Project | *MERN Stack (MongoDB, Express.js, React,, Node.js), Recharts* 2024

- Designed and implemented a full-stack application using the MERN stack, adhering to departmental and corporate requirements.
- Developed and tested a MongoDB-backed RESTful API with Express.js, including endpoints for CRUD operations, database initialization, and unit tests using Jest.
- Created a React-based client application, integrating features such as filtering, adding, editing, and deleting data.
- Deployed the full application on an Ubuntu server with Node.js and MongoDB, ensuring independent functionality for each department's APIs and client apps.
- Collaborated on implementing a consistent corporate theme across all departmental applications with shared landing pages and branding.
- Conducted operations reviews with the professor and had frequent team meetings.
- Added an integrated dashboard with Recharts for data analytics, providing metrics such as employee breakdowns, sales insights, and cost analysis with interactive visualizations.

Blackjack Strategy Helper | *Chrome Extension, Node.js, Express.js, Next.js, JSON* 2025

- Developed a Chrome extension to monitor a casino site's live game state by detecting updated JSON files.
- Implemented a Node.js + Express.js API backend that parses card states and returns actions from a strategy chart
- Built a Next.js frontend that polls an API endpoint every 20ms to display real-time play recommendations.
- Designed the system for minimal user input and fast feedback loops; built and deployed as a solo project.