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 *Expected May 2027*

- **Cumulative GPA: 3.18**
- **Dean's List: Spring 2023, Spring 2025**
- **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)

Certifications

AWS Certified Cloud Practitioner – Amazon Web Services, 2025

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, GitHub Actions, GitLab (CI/CD, Runner), Docker, Docker Hub, VS Code, Postman, cURL, Vim, Jest, Unix/Linux.
Cloud & Infra: AWS (Lambda, EventBridge, SNS, Comprehend, EC2, S3, CloudWatch, IAM), Terraform, Vercel.
Other: Apache HTTP Server, 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.

Azure Function & CI/CD | *Terraform, Azure (Functions, Storage, App Service Plan), GitHub Actions, Node.js* 2025

- Engineered and deployed Azure Function App infrastructure using Terraform for a Node.js serverless application, managed via Terraform Cloud.
- Implemented a CI/CD pipeline with GitHub Actions for automated infrastructure provisioning and (planned) application deployment.
- Systematically diagnosed and resolved multi-layered deployment blockers related to Azure regional quotas, Terraform provider version incompatibilities, and state management conflicts.
- Successfully provisioned all required Azure resources (Resource Group, Service Plan, Storage Account, Function App) after extensive troubleshooting, achieving a stable deployment.
- Documented complex issue resolution, including analysis of Azure Student account limitations and Terraform best practices for state and provider management.

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

- Built full-stack MERN application with MongoDB REST API, Express.js backend, and React frontend with CRUD operations and Jest testing.
- Deployed on Ubuntu server with Node.js and MongoDB, implementing corporate branding across departmental applications.
- Integrated Recharts dashboard for data analytics including employee breakdowns, sales insights, and cost analysis visualizations.