Kevin Chen

(585) 797-5153 | kc681269@gmail.com | Rochester, NY | linkedin.com/in/k3vnc | github.com/kvn8888

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

Ivalua Level 1 (Platform) and Level 2 (Technical and Configuration) Certifications – Ivalua, 2025

Technical Skills

Programming Languages: Java, Python, JavaScript, TypeScript, C, C++, C Sharp, SQL, Bash.
Frontend: HTML, CSS, React.js, Next.js, Tailwind CSS.
Backend: Node.js, Express.js, Flask, .NET, Prisma ORM, REST APIs, OAuth 2.0, JWT.
Databases: MongoDB, PostgreSQL, Microsoft SQL Server, MariaDB, DynamoDB, MySQL.
Dev Tools: Git, GitHub, GitHub Actions, GitLab (CI/CD, Runner), SVN, Docker, Docker Compose, Nx Monorepo, VS Code, Visual Studio, Postman, cURL, Vim, Jest, JUnit, Pytest, Unix/Linux.
Cloud & Infra: AWS (Lambda, EventBridge, SNS, Comprehend, EC2, S3, CloudWatch, IAM), Azure (Function Apps), Terraform, Terraform Cloud, Vercel, nginx, UFW, SSL/TLS.
Embedded & Hardware: STM32Cube, STM32 HAL, I2C, UART, SPI, DMA, DAC, GPIO, NVIC.
Other: Apache HTTP Server, Selenium, JSON, XML, Agile Methodologies, Scrum, SwiftUI, AVFoundation.

Experience

Customer Application Engineer Intern Oct 2025 – Apr 2026
Ivalua *Fremont, CA (Hybrid)*

- Provide post–go-live support for Ivalua's enterprise procurement platform, validating configuration and data change requests using C#, .NET, and Microsoft SQL Server.
- Diagnosed and resolved a role-based authorization misconfiguration for a major semiconductor client that prevented admin users from assigning required access profiles, restoring correct permissions without disrupting ongoing operations.
- Write and execute SQL scripts to clean up customer environments (e.g., removing blank workflows and invalid records) to maintain data integrity and workflow consistency.
- Use tools such as Visual Studio, Tortoise SVN, and Microsoft SQL Server Management Studio to debug issues, review changes, and deliver safe, incremental updates.
- Participate in weekly team meetings to report progress, discuss blockers, and coordinate next steps with senior engineers.

Projects

TA Portal – DevOps & Infrastructure Lead | *GitHub Actions, Docker, nginx, Prisma, Next.js* | *Course Project*2025

- Designed and implemented a GitHub Actions CI/CD pipeline using Nx monorepo tooling, enabling parallel lint/test execution and automated deployments triggered by branch naming conventions.
- Containerized a 4-service architecture (MariaDB, Workflow API, Express.js backend, Next.js frontend) using Docker Compose with health checks, dependency ordering, and persistent volumes.
- Configured nginx as a reverse proxy with SSL termination, routing traffic to Docker containers and eliminating the need for application-level HTTPS handling.
- Established SSH key-based authentication between GitHub Actions and the production VM, enabling secure passwordless automated deployments.

- Set up UFW firewall rules to allow HTTP/HTTPS traffic and GitHub Actions runner access while maintaining server security.
- Fixed Prisma ORM container issues by adding client generation steps to Dockerfiles, resolving runtime initialization errors.
- Configured Next.js basePath routing to work behind nginx reverse proxy, eliminating 404 errors in production.
- Authored comprehensive deployment documentation covering Docker, nginx, firewall configuration, and troubleshooting procedures.

Stock Sentiment Tracker | *AWS, Terraform, Boto3, Python, GitHub Actions, Git* | *Cloud Engineering Course Project*2025

- Architected and co-developed a serverless sentiment analysis platform for stock discussions with a team of four, reducing manual research time for users.
- Programmed AWS Lambda functions using Python (Boto3) to perform sentiment analysis with Amazon Comprehend, optimizing logic to reduce redundant API calls and improve data processing speed.
- Implemented an automated notification system using EventBridge and SNS, delivering daily email alerts to users for their subscribed tickers.
- Authored Infrastructure as Code (IaC) Terraform scripts to provision and manage all cloud infrastructure (EventBridge, Lambda, SNS, IAM), enabling 100% reproducible environments and cutting down deployment time by 90%.

Azure Function & Google Cloud TTS | *Terraform, Azure, GitHub Actions, Node.js* | *Personal Project* 2025

- Engineered and deployed Azure Function App infrastructure using Terraform for a Node.js serverless application, fully managed via Terraform Cloud for state consistency.
- Built a complete CI/CD pipeline in GitHub Actions that automated infrastructure provisioning, cutting manual deployment steps by 100%.
- Diagnosed and resolved complex deployment blockers, including Azure regional quotas, Terraform provider bugs, and state file conflicts, leading to a 100% success rate for subsequent deployments.
- Authored detailed documentation on the troubleshooting process, establishing best practices for managing cloud resource limitations and provider-specific versioning issues.

IoT Environmental Monitor | *C, STM32 HAL, I2C, UART, AWS IoT* | *Personal Project* 2025

- Developed and programmed an embedded environmental monitor on an STM32 Nucleo board to collect sensor data at a 1Hz frequency and display it on a 1602 LCD.
- Used low-level drivers in C to interface with a DHT22 temperature/humidity sensor via I2C, implementing error handling for bus communication failures.
- Retargeted the `_write` syscall to stream debug logs over UART, reducing debugging time compared to manual methods.

MIDI Player | *C, STM32Cube, Embedded Systems, Git* | *Course Project* 2024

- Engineered and debugged a MIDI player on an STM32 Nucleo board, using C to parse MIDI file data structures and generate corresponding audio signals.
- Implemented performance optimization by managing hardware resources, including DMA for efficient data transfer and DAC for precise analog waveform generation, producing clear audio on a piezo buzzer.
- Designed the system to utilize microcontroller peripherals, including USART for file reception, GPIO for button-based UI, and NVIC for interrupt-driven controls (track selection, play/pause).
- Conducted code reviews and debugging sessions to ensure firmware stability and adherence to embedded C best practices.

Enterprise Application Project | *MERN Stack, Recharts, Jest* | *Enterprise Engineering Course Project* 2024

- Collaborated with a team of four using Agile Methodologies to develop a full-stack MERN application for a simulated enterprise, meeting specified business requirements.
- Built and tested a secure RESTful API with Node.js, Express.js, and MongoDB, featuring over 12 CRUD endpoints and achieving 95% unit test coverage with Jest.
- Developed a responsive React client with a Recharts analytics dashboard to visualize key metrics, which improved user task completion time and informed strategic decisions.
- Managed the deployment of the full MERN stack to an Ubuntu server, configuring Apache HTTP Server as a reverse proxy to the backend.
- Integrated and debugged an analytics dashboard with Recharts to provide data-driven insights, visualizing key metrics like sales trends and employee performance which informed strategic decisions in project reviews.

WatchOS Voice Chatbot | *SwiftUI, XCode, Groq API, REST* | *Personal Project* 2024

- Programmed a proof-of-concept voice-to-voice chatbot on watchOS using SwiftUI for the user interface and AVFoundation for audio capture and playback.
- Integrated with the Groq API for near-real-time AI inference, leveraging Whisper for speech-to-text and Llama 3.1 for text generation, achieving an average response time of under 2 seconds.
- Engineered the network layer to handle asynchronous API calls and manage data flow between the app and the backend services efficiently.

Rush Hour Puzzle Game | *Java, JavaFX, JUnit, Git* | *Course Project* 2023

- Collaborated in a team of three to build a fully functional Rush Hour puzzle game with a JavaFX GUI, following Agile/Scrum practices.
- Designed core game logic with a backtracking algorithm and wrote JUnit tests covering 90% of code to ensure stability.
- Managed Git workflow (branching, merging, and reviews) to maintain efficient version control and collaboration.