

Subhajit Guchhait

Ames, IA | +1 (515) 766-1635 | subhajitguchhait04@gmail.com | [Github](#)

Software Developer with 2+ years building REST APIs and event-driven microservices in Go, Node.js, and TypeScript on AWS. Skilled in MongoDB and PostgreSQL modeling, Docker deployments, and AI features with OpenAI APIs.

EDUCATION

Iowa State University

Bachelor of Science in Computer Science

Ames, IA

May 2026

TECHNICAL SKILLS

Languages: TypeScript, JavaScript, Python, Java, Go **Databases:** PostgreSQL, MySQL, MongoDB, Prisma, Redis, DynamoDB

Frontend: React, Next.js, React Native, Redux, Recoil, Tailwind CSS, HTML5, CSS

Backend: Node.js, Express.js, Go, REST APIs, Microservices, JWT Authentication, RabbitMQ

AI/ML: OpenAI APIs, Anthropic Claude API, LLM Integration, Prompt Engineering, Embeddings, Vector Search

Cloud & DevOps: AWS, Docker, Kubernetes, Ingress, ArgoCD, Helm, Nginx, GitHub Actions, GitOps, CI/CD, Rate Limiting, Captcha, DDoS Protection

Tools & Practices: CloudWatch, Prometheus, Grafana, Jest, Postman, Turborepo, Git, Agile (Scrum)

EXPERIENCE

Software Developer, Workiva

Ames, IA

Enterprise Workflow Automation and Compliance Platform

May 2025 – Present

- Developed 25-plus REST API endpoints on Node.js and Express.js for user management, workflow execution, and document processing, secured with JWT authentication and role-based access control (RBAC) and documented in OpenAPI/Swagger.
- Built event-driven microservices in Go backed by RabbitMQ that process workflow and notification events, decoupling long-running jobs so user-facing requests never block on downstream processing.
- Designed data models and tuned queries in MongoDB and PostgreSQL behind millions of workflow records through targeted indexing and query restructuring.
- Containerized services with Docker and automated builds and deployments to AWS EC2 through GitHub Actions, with Prometheus and Grafana dashboards to catch production failures early.
- Wrote unit and integration tests with Jest and validated 25-plus endpoints with Postman, and improved backend efficiency by reducing redundant queries and unnecessary processing.

Research Assistant, Business and Marketing Department

Iowa State University, Ames, IA

Data Analyst

January 2025 – May 2026

- Built a Python pipeline that scored roughly 5,000 SEC 10-K filings from every Fortune 500 company over 10 years against a lexicon of 50-plus AI and ML terms, replacing hundreds of hours of manual reading with one automated run.
- Applied OpenAI embeddings and semantic similarity to separate genuine AI strategy in CEO and management-discussion sections from incidental mentions, turning unstructured filings into a year-over-year signal of AI investment.

OPEN SOURCE CONTRIBUTIONS

Jitsi Video Conference Software | *TypeScript, React, Redux, Node.js*

Open Source Contributor

- Shipped a merged feature to Jitsi's public API that surfaces message and reply identifiers on chat events, letting apps that embed Jitsi correlate messages and render reply threads. **(merged)**
- Fixed a data-loss bug that dropped chat reply metadata before it reached the app state, wiring it through end to end and closing a tracked project issue. **(merged)**

PROJECTS

.spark, AI-Powered Full-Stack App Generator

- Built a Next.js/React platform that turns a prompt into a deployable full-stack app, each in an isolated sandbox with a live preview and browser code-server editor so one user's code can never touch another's.
- Wrote a TypeScript streaming parser over the Claude API that writes files mid-generation so LLM Integration output appears live, with a fallback that keeps large Prompt Engineering scaffolds from breaking.
- Built a self-healing orchestrator on AWS Auto Scaling with an Express.js loop that reconciles the fleet every 10 seconds so dead instances are evicted before a user is handed one, keeps a warm pool of Docker sandboxes to cut cold-start latency, and runs a 30-minute reaper that terminates stale machines so idle sessions stop leaking cost.

Second Brain, AI Content Management

- Built a full-stack TypeScript app with a React client that recalls saved videos, tweets, and articles by meaning, turning each into a 1536-dimension Embeddings vector through OpenAI APIs and using Vector Search to score 100 nearest-neighbor candidates down to the top 10, so results match on intent rather than the exact keywords a user typed.
- Built the Node.js/Express.js API that stores content in MongoDB, secured with JWT auth and Zod validation that rejects bad input before it reaches the database, with fallbacks so a failed fetch never blocks a save.
- Containerized the app with Docker behind an Nginx single-origin proxy on AWS EC2, fronted by Cloudflare for HTTPS and DDoS Protection, removing CORS and keeping the server off the public internet.

Crucible, Sandboxed Code Execution Judge

- Built a code-execution judge from scratch that runs Python, JavaScript, and Java against hidden tests inside hardened Docker containers, capping each run at 256 MB, half a CPU, a 3-second timeout, and 64 processes with no network so untrusted submissions cannot exhaust the host, fork-bomb it, or reach the internet.
- Structured it as a Turborepo monorepo with an Express.js API, worker, and Next.js/React web app, exposing a submit-and-poll REST API tested with Postman so long container runs never tie up server connections.