

SIVA SANKAR REDDY ASAM

Bengaluru, India | +91 8688660794 | aasamsivasankarreddynani143@gmail.com | [Linkedin](#) | [GitHub](#) | [Portfolio](#)

Summary

Software Engineer with **1+** years of experience building scalable, real-time full-stack applications. Proven in enhancing performance (+60%), owning features end-to-end, and **developing GenAI-powered solutions like DocuMentor (RAG-based PDF Q&A)**. Expertise in **TypeScript, React/Next.js, Node.js/NestJS, Python, FastAPI, LangChain, PostgreSQL/pgvector, Docker, and AWS/Cloud Deployment**. Actively applying LLMs and RAG techniques to create intelligent, data-driven applications. Strong foundation in DS&A (500+ LeetCode). Seeking an **SDE 1/2** role to tackle complex technical challenges and drive innovation.

Skills

- **Languages:** JavaScript/TypeScript, **Python (FastAPI, LangChain, ML/AI)**, Go, SQL
- **AI & Generative AI:** LLMs (**TinyLlama/Ollama**), **RAG Architecture, LangChain, Vector Databases (PostgresSQL/pgvector)**, Prompt Engineering, Hugging Face Transformers, Document Loaders, Text Splitters, Embeddings
- **Frontend:** React/Next.js, State Management (Redux/Zustand), Data fetching (Tanstack Query), Performance Optimization (React DevTools, Lighthouse), UI Libraries (Tailwind CSS, Shadcn UI), HTML/CSS
- **Backend:** Node.js (NestJS, Express.js), Frappe Framework, Django, Flask, API Design (REST, GraphQL), WebSockets (Socket.IO), Microservices (Conceptual), Message Queues (BullMQ, Redis Streams), JWT Authentication, ORMs (Prisma)
- **Databases:** PostgreSQL (pgvector), MongoDB, MySQL, Redis (Caching, Pub/Sub)
- **Cloud & DevOps:** AWS (ECS, EC2, S3, Lambda, RDS, ECR, CloudWatch), Docker, Terraform (IaC), **CI/CD (GitHub Actions)**, **Cloud Deployment (Vercel, Render/Fly.io)**, Kubernetes (Conceptual)
- **Testing:** Jest, Supertest, React Testing Library (RTL), Unit Testing
- **Architecture & Design:** System Design (HLD, LLD), API Design, Database Modeling, OOP/OOD, SOLID Principles, Design Patterns, **Data Structures & Algorithms**
- **Tools:** Git, GitHub, Jira, Confluence, Postman, Docker Desktop, VS Code, Insomnia REST, Figma (collaboration)

Work Experience

Software Engineer | **Nirmaan (Stratos Infra Technologies Pvt. Ltd.)** | Bengaluru, India | **July 2024 - Present**

- **Accelerated application performance by 60%** and slashed API latency by **55%** through architecting and deploying multi-layer caching strategies (SWR client-side, Redis server-side).
- **Engineered a high-performance data table component** handling **>10k rows**, featuring server-side pagination and dynamic filtering, improving data rendering speed by up to **60%** via optimized SQL queries (within Frappe).
- **Owned and delivered a real-time internal communication system** utilizing WebSockets (Socket.IO), enhancing team collaboration.
- **Architected and built a context-aware chatbot** integrating with the database via Python backend logic to provide instant, relevant answers to domain-specific user queries.
- **Modernized legacy codebase modules (~80% refactored)** using TypeScript, SOLID principles, and modular design, reducing estimated technical debt by **45%** and improving code maintainability.
- **Integrated Python-based AI analytics**, providing real-time operational insights that informed key data-driven decisions.
- **Optimized RESTful APIs** and refined frontend state management, achieving a 35% reduction in load times for specific complex UI workflows.

Full Stack Engineer | **Attica Gold Company** | Bengaluru, India | **March 2024 - July 2024**

- Constructed a **PoC CCTV management system** (MERN stack, Flask, and OpenCV) for multi-stream video processing.
- Leveraged **AWS services (S3, Lambda)** for scalable and secure storage of video footage (**>10TB/month**).
- **Secured application access** using JWT-based authentication and Role-Based Access Control (RBAC).

Projects

DocuMentor – AI-Powered Context-Aware Document Q&A | [GitHub Link](#) | [Live Demo](#)

- Engineered and deployed **DocuMentor**, a full-stack GenAI application allowing users to upload PDFs and receive context-aware answers using a RAG (Retrieval-Augmented Generation) pipeline
- Built a **FastAPI (Python) backend** with LangChain for PDF parsing, chunking, embedding generation (Sentence Transformers), and orchestration of the RAG flow with a locally hosted **LLM (TinyLlama via Ollama)**.
- Implemented vector storage and similarity search using **PostgreSQL with the pgvector extension**
- Developed a responsive **Next.js (TypeScript) frontend** with Tailwind CSS and Shadcn/ui for PDF upload and real-time Q&A display
- Containerized services using **Docker** and configured for cloud deployment (e.g., Hugging Face Spaces for backend, Vercel for frontend), showcasing end-to-end application development and deployment
- **Tech Stack Highlights:** Python, FastAPI, LangChain, Ollama (TinyLlama), Next.js, TypeScript, PostgreSQL (pgvector), Docker, Sentence Transformers, Tailwind CSS, Shadcn/ui, [HF spaces, Vercel].

Aether – Collaborative Real-Time Incident Response Platform | [GitHub Link](#) | [Live Demo](#)

- **Designed, engineered, and deployed** a full-stack, real-time incident management platform from concept to production on AWS Fargate within an 8-week sprint, simulating enterprise SDLC.
- **Built a scalable NestJS (TypeScript/Node.js) backend** featuring modular architecture, PostgreSQL/Prisma data layer, Redis caching (if used), JWT auth, and robust error handling.
- **Created a dynamic, responsive frontend using Next.js (React/TypeScript)**, Tanstack Query, Zustand, and Tailwind CSS, enabling efficient state management and interaction.
- **Mastered real-time communication using WebSockets (Socket.IO)** to build a collaborative "war room" with live chat, status updates, and task synchronization, solving authentication race conditions and ensuring low-latency (<100ms target) updates. (Estimate latency if possible)
- **Orchestrated deployment using Docker and GitHub Actions CI/CD**, automating builds, tests, and zero-downtime deployments (if applicable) to AWS ECS Fargate. Managed infrastructure using Terraform (IaC).
- **Tech Stack:** TypeScript, React, Next.js, Node.js, NestJS, PostgreSQL, Prisma, WebSockets, Docker, GitHub Actions, AWS ECS Fargate, Terraform, Redis, Zustand, Tanstack Query, Tailwind CSS.