

Akash Chavan

Santa Clara, CA • iakashchavan@gmail.com • (747) 244-1272

Experienced **Software Engineer** with over **6 years** of industry experience specializing in **Python-based web applications, AI integration** and **cloud infrastructure**. Proven track record of developing scalable platforms, real-time monitoring systems, and AI-powered applications using **Django, React** and **AWS**. Proficient in database optimization, **RESTful API** development, and containerized deployments. Skilled in implementing Generative AI models for practical applications including **LLMs** and **RAG systems**. **Actively seeking opportunities** to lead complex development projects and build advanced AI-powered solutions.

AWARDS, RECOGNIZATION, CERTIFICATIONS

- Special Recognition in Graduate Studies
Associated with Department of Computer Science, California State University, Los Angeles
- Deep Learning Certification
Associated with Udacity
- National Cyber League (NCL) Spring 2024 Individual Game
Associated with National Cyber League (Ranked 743 out of 7,412 participants)

PROFESSIONAL EXPERIENCE

ABBVIE

AI Engineer

Remote, US

May 2025 - Present

Architected an end-to-end security compliance solution integrating a scalable Python FastAPI backend and an AI-powered agent interface, accelerating security workflow automation and reducing manual research time.

- Architected scalable Python FastAPI with Docker containerization (main backend, embedding service, Python ARQ workers), reducing docker build times by 95% through strategic ML model isolation and enabling seamless development iteration.
- Resolved critical deployment blocker by identifying and implementing pgvector extension configuration for PostgreSQL, enabling 768-dimensional vector embeddings with HNSW indexing for sub-50ms semantic search latency and unblocking production-ready RAG pipeline.
- Deployed production ready API stack with PostgreSQL + pgvector, Redis caching layer, AWS S3 integration, and ARQ background workers, while proactively implementing automatic database migrations and health check monitoring across all microservices.
- Developed real-time streaming UI using [Node.js](#) Server-Sent Events (SSE) and React 19 features, implementing custom hooks for connection management, automatic reconnection with exponential backoff and UI updates for responsive chat experience with <100ms perceived latency.
- Built enterprise-grade multi-step reasoning AI agent using LangGraph state machine architecture with adaptive RAG pipeline, leveraging hybrid search (70% semantic vector + 30% keyword), cross-encoder re-ranking and query expansion to retrieve contextually relevant document chunks with 40-60% cost optimization through intelligent model selection.
- Developed interactive React-based web interface with real-time WebSockets communication, markdown rendering (React-Markdown), and syntax highlighting ([Prism.js](#)), enabling teams to query AI tools via natural language with responsive UX.
- Optimized query processing workflows by implementing 4-dimensional quality scoring (accuracy, completeness, clarity, citations), multi-tier Redis caching (24h embedding, 2h response, 72h chunk cache with 80%+ latency reduction), and semantic chunking with content-type adaptation for 35% improved retrieval accuracy.

POLARIS WIRELESS

Python Developer

Santa Clara, CA

Sept 2024 - May 2025

Developed comprehensive synthetic data generation tools and analytics solutions for QA and product demonstration teams in a fast-paced environment.

- Designed and implemented Python-based synthetic data generation tools supporting QA testing and product demonstration workflows, enabling teams to create realistic datasets for validation and presentation purposes.
- Created automated data ingestion scripts for loading synthetic data into analytics platforms with comprehensive KPI monitoring to ensure complete and accurate data processing.
- Utilized scientific python libraries like pandas/pyspark, numpy, dask for advanced data processing, statistical analysis and mathematical operations on large-scale synthetic datasets.
- Implemented memory-efficient XML parsing tools processing 70k+ files per day using streaming techniques and batch processing, handling geographic coordinate data.
- Implemented comprehensive data validation and reporting systems generating CSV outputs with statistical summaries and day-wise breakdown for network event analysis.
- Designed scalable architecture supporting concurrent file processing with configurable batch sizes and memory optimization for handling enterprise-scale telecommunications datasets.
- Architected geolocation intelligence REST API using FastAPI with async/await patterns, implementing LOCATE, TRACK, MONITOR endpoints supporting concurrent location requests with Pydantic validation.
- Built geofencing engine with PostGIS spatial queries for polygon and circular region definitions, implementing point-in-polygon detection and proximity calculations between targets with configurable distance thresholds for entry/exit alerting.
- Implemented Haversine distance calculations with uncertainty radius propagation, enabling accurate meeting detection between subscribers with temporal LEADING/WITHIN/TRAILING classification.
- Built real-time alert streaming using Server-Sent Events pushing entry/exit notification and proximity alerts to connected clients with automatic reconnection.
- Implemented Celery-based task queue and Redis broker for asynchronous processing of analysis jobs.

FOSSEE, IIT BOMBAY

Software Engineer

Mumbai, India

Feb 2019 - May 2022

Built and maintained scalable e-learning platform serving educational content to thousands of users with advanced code evaluation capabilities

- Implemented a scalable e-learning platform handling 10,000+ concurrent users using Django, combining video lessons, automated code evaluation, and interactive quizzes with Python, C, C++, Java, R, Scilab and Bash Support.
- Designed a secure, distributed code execution system using Django Channels and Tornado web Server to safely evaluate student-submitted code in real-time with process isolation and resource constraints.
- Implemented a leader-follower architecture for code evaluation using process pools to handle concurrent requests, implementing job queues for load balancing and efficient resource utilization, reducing response time by 50%.
- Optimized Django ORM queries using select_related and prefetch_related, resolving N+1 query issues and reducing page load times by 60%.
- Implemented asynchronous background task processing using Django Celery and Redis for notifications, scheduled reminders and resource-intensive operations, improving system reliability.
- Migrated frontend from Django templates to [Vue.js](#), creating a responsive single-page application with improved user experience and real-time updates, increasing user engagement by 40%.
- Created REST APIs using Django REST Framework to support mobile clients and third-party integrations, with proper authentication and rate limiting.
- Implemented analytics dashboard for instructors to monitor student progress, identify learning gaps and optimize course content based on performance data.
- Implemented timetabling solutions using Pandas/PySpark, NumPy and Bash, reducing scheduling errors and enhanced operational efficiency
- Implemented automated configuration management using Docker containers and orchestration tools for distributed system deployment and maintenance.

VIRTUAL LABS, IIT BOMBAY

Software Engineer

Mumbai, India

Oct 2017 - Feb 2019

Developed remote access solution for Single Board Heater System enabling distributed laboratory experiments with real-time monitoring and control capabilities

- Created a web application enabling remote access to Single Board Heater System (SBHS), enhancing accessibility and user experience for laboratory experiments in distributed learning environments.
- Implemented a load-sharing leader-follower architecture using Raspberry Pi's, optimizing communication between SBHS devices and the central server, improving system efficiency by 45%.
- Developed a lightweight Flask server deployment on Raspberry Pi's, establishing an efficient communication medium between SBHS hardware and the main server with 99.5% uptime.
- Designed and implemented a Moderator interface providing administrators remote control capabilities over SBHS devices, streamlining laboratory management operations.
- Created a health monitoring script to track and report SBHS hardware status in real-time, improving system reliability and reducing maintenance downtime by 25%.
- Integrated real-time data visualization using Scilab, allowing users to view and analyze SBHS readings through a desktop application with interactive charts and graphs.
- Developed a slot booking interface, streamlining the process for users to reserve and utilize SBHS devices efficiently, reducing scheduling conflicts by 40%.
- Created an automated power management feature, optimizing energy usage by automatically turning devices on/off, reducing power consumption by 30%.

PROJECTS

AI-powered document learning platform

- Built an AI-powered document analysis platform using [Next.js](#), React 19, TypeScript and PostgreSQL.
- Implemented a microservices architecture with separate FastAPI service for document processing and embedding generation.
- Designed RESTful APIs handling user authentication, document management, realtime chat and vector similarity search.
- Integrated OpenAI GPT-4 for intelligent document Q&A with context-aware responses and proper citation formatting.
- Implemented Retrieval Augmented Generation (RAG) pipeline using sentence-transformers for semantic document search.
- Built custom embeddings service using FastAPI and sentence-transformers for PDF document vectorization.
- Designed and implemented pgvector extension for PostgreSQL to enable efficient semantic similarity search.
- Deployed scalable application infrastructure using Docker containers and AWS services (S3, EC2)
- Implemented secure file storage and retrieval system using AWS S3 with presigned URLs.
- Configured CI/CD pipeline and container orchestration for development and production environments.
- Designed system to handle concurrent user uploads and document processing through asynchronous job queues.
- Implemented efficient document chunking and processing pipeline handling PDFs up to 10MB.

Email Digital Twin - AI-powered Email Response System

- Developed a digital twin application using FastAPI backend and [Next.js](#) frontend that analyzes incoming emails with PDF attachments and generates responses mimicking user writing style.
- Implemented OAuth 2.0 Gmail integration with secure session management, enabling seamless email retrieval, thread analysis, thread analysis and PDF attachment processing using PyPDF2 for text extraction.
- Created AI-powered analysis pipeline using OpenAI GPT-4o-mini to process email context, extract key insights and generate contextually relevant responses based on conversation history and attachment content.
- Build scalable microservices architecture with modular service layers (Gmail API, PDF processing, AI analysis) and RESTful API endpoints supporting both individual email and thread-based operations.

SKILLS

- **Databases:** PostgreSQL, MySQL, MongoDB, Redis, DynamoDB

- **Programming Languages:** Python, JavaScript/TypeScript, Java, SQL
- **Backend Frameworks:** Django, FastAPI, [Node.js](#), [Express.js](#), Flask, GraphQL
- **Frontend Technologies:** React, Vue, [Next.js](#), TailwindCSS, HTML, CSS
- **Cloud & DevOps:** AWS (EC2, S3, Lambda, CloudFront), Docker, Kubernetes, CI/CD, GitHub Actions
- **AI/ML:** LangChain, LangGraph, Vector Databases, RAG Systems
- **Testing:** Test-driven development (TDD), PyTest, UnitTest, Jest, Mocha, Chai
- **AI Tools:** Claude Code, Cursor, Codex, Cline, Copilot, Kilo Code

EDUCATION

CALIFORNIA STATE UNIVERSITY, LOS ANGELES

MS in Computer Science (GPA: 4/4)

Los Angeles, US

May 2024

DR. BABASAHEB AMBEDKAR MARATHWADA UNIVERSITY

B.E Computer Science and Engineering

Aurangabad, India

May 2016