

AKASH SINGH

Full-Stack Engineer (Backend-Leaning) | AI Voice Agents | Event-Driven Systems

Bengaluru, Karnataka, India | akashsingh095@gmail.com

akash-dev.com | linkedin.com/in/akash-singh-495333157 | github.com/akssingh0102 | leetcode.com/akashsingh095

PROFESSIONAL SUMMARY

Full-stack software engineer with around 5 years building production LLM and voice AI systems, event-driven backends, and data-intensive frontends. Backend-leaning across Python, TypeScript, Go, and Node.js. Shipped real-time voice agents against sub-800ms turn-latency budgets, Kafka pipelines with exactly-once semantics, and RAG analytics agents used daily by non-technical operators. Early engineer at a pre-Series-A startup through acquisition.

TECHNICAL SKILLS

Languages: Python, TypeScript, JavaScript, Go, Node.js, C++, SQL

AI and Machine Learning: LLM orchestration, RAG pipelines, tool calling, LangChain, LangSmith, OpenAI API, embeddings, pgvector, NLP-to-SQL, structured outputs, LLM evaluation and observability, prompt engineering

Voice and Real-Time Systems: Streaming ASR, streaming TTS, OpenAI Whisper, barge-in handling, endpointing and turn detection, WebSockets, WebRTC, gRPC

Backend and Event-Driven Architecture: FastAPI, Django, AsyncIO, Apache Kafka, Celery, Redis pub/sub, RabbitMQ, distributed task queues, idempotent consumers, webhook ingestion and routing, retry and dead-letter handling, REST APIs, microservices

Databases and Data: PostgreSQL, MySQL, MongoDB, Redis, Elasticsearch, Apache Airflow, database design, query optimization

Cloud and DevOps: AWS, GCP, Docker, Kubernetes, Nginx, Linux, Prometheus, CI/CD, Jenkins, Git

Frontend: React, Next.js, TypeScript, Tailwind CSS, Turborepo, Three.js, WebGL, D3.js

PROFESSIONAL EXPERIENCE

Software Development Engineer II, Full-Stack AI Engineer

Petasense | Bengaluru, India | April 2025 - Present

- Built a RAG-powered conversational analytics agent using LangChain, pgvector, and OpenAI embeddings, letting non-technical operators query account-specific industrial datasets in plain English. Multi-stage async pipeline covering semantic retrieval, NLP-driven dynamic SQL generation, multi-turn context memory, and structured output validation; achieved 95% query accuracy and cut analyst workflow time by 60%.
- Designed a tool-use agent framework with LangChain tool-calling and configurable retrieval chains, plus an LLM evaluation harness benchmarking retrieval precision and SQL execution correctness against a 200-query golden dataset, giving the team a reproducible signal for prompt iteration and model upgrades.
- Replaced a flat 2D machine schematic with an interactive Three.js digital twin spanning 75 modelled component types, rendering live vibration, temperature, and open alarms directly onto the geometry so a failing bearing surfaces on that bearing rather than in a separate table. Sensors are fitted by dragging onto mount points, layout edits apply optimistically and roll back on refusal, and a 2D fallback covers non-WebGL browsers.
- Owned a high-availability Annotations System across microservices with Redis caching and role-based access control, delivering sub-second response times, over 99% uptime, and full audit logging on every write path.
- Partnered directly with reliability engineers and plant operators to translate ambiguous maintenance workflows into product requirements, then shipped against them end to end.

Software Development Engineer

Exotel | Bengaluru, India | June 2023 - April 2025

- Owned the full pipeline for production LLM voice agents in contact center automation, from telephony media ingest through streaming ASR, intent resolution, and TTS playback back onto the live call, designed against a sub-800ms turn-latency budget, the threshold past which callers talk over the bot.
- Solved the hard parts of live conversation rather than the happy path: barge-in with in-flight TTS cancellation, endpointing and turn detection on partial ASR results, and graceful degradation when any single stage slowed, so a slow LLM call never blocked the audio loop.
- Engineered a real-time call intelligence pipeline in Node.js, Django, Redis, and OpenAI Whisper, processing thousands of call recordings into automated transcripts, multi-label sentiment scoring, and agent performance summaries; improved analysis accuracy by 20% and eliminated 6 hours of manual QA per analyst per week.
- Designed an event-driven ingestion system on Apache Kafka and Node.js consuming call lifecycle events at scale, anchoring exactly-once semantics downstream through idempotent writes and unique constraints rather than trusting

the transport layer, with routing into analytics and CRM integrations; cut event processing latency by 40% and eliminated data loss during peak traffic spikes.

- Shipped a context-aware LLM co-pilot for live agents using GPT-4 with retrieval-augmented generation over internal knowledge bases, surfacing in-call suggestions, next-best-action prompts, and dynamic FAQ resolution; drove a 25% CSAT improvement across 3 enterprise customers.
- Architected the Automated Contact Center platform frontend in TypeScript with Next.js and a Turborepo monorepo, introducing code-splitting and edge caching that cut page load time by 30% and lifted engagement by 25%, directly impacting trial-to-paid conversion.

Software Development Engineer

Cogno AI (merged with Exotel) | Mumbai, India | April 2022 - June 2023

Early engineer at a pre-Series-A startup, acquired by Exotel in 2023. Intern through July 2022.

- Built the video layer for LiveChat on Django, WebSockets, and the Dyte SDK, with participant tokens minted server-side so no call credential reached the browser and signalling riding the WebSocket channel chat already held open. Focused on failure paths: denied camera and microphone permissions, rooms orphaned by a closed tab, and mid-call network drops that must return the customer to the same session; cut video support tickets by 30% and lifted session engagement by 18%.
- Rebuilt LiveChat agent assignment from a 30-second polling sweep into an event-driven service on Django, Celery, and Redis, routing on query domain and preferred language against live agent availability, with an atomic claim so two concurrent events cannot take the same agent and bounded retry with backoff; removed the 30-second worst-case wait, improved scheduling efficiency by 66%, and eliminated 110 hours of manual operational work per week.
- Brought Instagram and Facebook into LiveChat Fusion as first-class channels on the Meta Graph API, normalising each platform payload into one conversation model so agents worked a single queue instead of a tab per network. Inbound webhooks acknowledge immediately and fan out to workers, messages are deduplicated on the platform message ID so a retry cannot post twice, and outbound sends respect per-page rate limits; expanded across 3 channels and cut average response time by 15%.

PROJECTS

DistributedKV | Fault-Tolerant Key-Value Store in Go

- Implemented Raft consensus from scratch with consistent hashing over virtual nodes, automatic leader election, read replicas, and a custom binary gRPC wire protocol for inter-node RPC.
- Sustained under 2ms p99 read latency across a 4-node Docker cluster, with automatic failover verified at the leader-crash boundary and zero data loss.

AgentBench | LLM Evaluation and Testing Framework

- Configurable evaluation framework running automated suites against RAG pipelines and tool-use agents, scoring retrieval precision, answer faithfulness, and SQL execution equivalence against golden datasets.
- Per-run tracing, latency profiling, and token cost analysis producing structured reports, so model swaps and prompt changes are decided on evidence rather than manual spot checks.

Driftbell | Multi-Tenant SaaS Changelog Platform

- Built zero to one to paying customers solo using Next.js 14, TypeScript, Prisma, and PostgreSQL, with a web dashboard, public changelog, email digests, and an embeddable JavaScript widget distributed via a single script tag.
- Shipped the widget as a zero-dependency IIFE bundle under 15KB, plus end-to-end Stripe billing with webhook-driven plan enforcement across three tiers via typed server actions on every write path.

Topograph | Service Topology and Observability Platform

- Unified metrics, logs, and traces from a microservice fleet into one pipeline: Kafka for ingest and backpressure, Elasticsearch on the query side, sub-second metric visibility, and a rolling-baseline detector alerting on deviation rather than a static threshold.
- Derived service topology from traffic rather than configuration, using span parent-child relationships and caller metadata, collapsed across instances and decayed over time, rendered as live dependency maps in React.

ACHIEVEMENTS

- Ranked in the top 212 nationally in a 2019 competitive programming contest.
- Solved 600+ algorithmic problems across LeetCode and GeeksforGeeks; 4-star rating on HackerRank.

EDUCATION

Bachelor of Technology, Computer Science

Lovely Professional University | Phagwara, Punjab, India | Graduated 2022