

NIKHIL BAJAJ

Pune, India
9359368491
nbajaj201@gmail.com
[nikhilbajaj.xyz](mailto:nbajaj201@gmail.com)
LinkedIn

AT A GLANCE

4 European markets served by backend systems you built

Kernel-level network observability built with custom BPF filters, no app instrumentation required

40% version-control reliability gain (doc management system)

25% ETL processing time cut via pipeline redesign

SKILLS

Languages

Go, Ruby, TypeScript, Python, C/C++

Backend

Rails, FastAPI, Django, Node.js, Sidekiq

Data

PostgreSQL, MongoDB, Redis, ClickHouse

Cloud/Infra

AWS (ECS, ECR, EFS, S3, Lambda), Docker, Terraform, Kubernetes

Streaming

Kafka, WebSockets, event-driven design

Observability

Prometheus, Grafana

EDUCATION

B.E., Electronics & Telecommunication

Pune Institute of Computer Technology
08/2015 – 05/2019

LANGUAGES

English (Advanced)

PRINCIPAL BACKEND ENGINEER

Distributed systems · real-time data pipelines · high-throughput infrastructure

Backend engineer with 6+ years shipping production systems across four international markets, alongside independent projects in kernel-level network observability and resilient infrastructure design. Comfortable owning a system end to end: architecture, data pipelines, and operational reliability.

EXPERIENCE

Principal Software Engineer · GoGroup, Remote (India) 05/2022 – Present · **ACTIVE**

B2B platform connecting automotive parts suppliers, workshops, and insurers across Europe

- ▶ Architected backend + AWS infrastructure (**ECS/ECR**, Docker) running production workloads across **four markets**: Germany, Netherlands, Spain, New Zealand
- ▶ Built a **real-time ranking engine** for multi-vendor offer evaluation, replacing manual procurement comparison with instant, ranked pricing
- ▶ Designed an **event-driven ingestion pipeline** (SFTP monitoring + edge-level validation) that closed the data-reliability gaps in downstream services
- ▶ Built an API telemetry and persistence layer generating **audit-ready timelines** for complex, multi-step repair-job lifecycles
- ▶ Owned CI/CD end-to-end (**GitHub Actions** to Docker to ECR to ECS), cutting release friction across the team
- ▶ Shaped quarterly backend/infra roadmap as the most senior IC on the team

Software Engineer · Altizon Inc, Pune (India) 10/2019 – 01/2022

Industrial IoT analytics platform

- ▶ Shipped **Ruby on Rails APIs** and React front-end features for a cloud-based IIoT analytics platform, including complex MongoDB queries against a large-scale data lake
- ▶ Designed and built a **document management system from scratch** (Rails, React, AWS S3/EFs), lifting version-control reliability by **~40%**
- ▶ Built ETL/background-job pipelines that cut data processing time by **~25%** and powered real-time alerting

INDEPENDENT PROJECTS

TotallyHuman · High-performance resilient scraping infrastructure

Core stack: Python · FastAPI · curl_cffi · TLS/JA3 fingerprint resilience · Linux

- ▶ Authored a resilient web-automation runtime engineered to operate reliably against enterprise-grade bot-mitigation systems, edge WAF rules, and JA3/TLS fingerprint checks
- ▶ Developed a containerized **FastAPI** mask-service sidecar that performs low-level network fingerprint matching to align outbound traffic with standard consumer-browser profiles
- ▶ Engineered high-frequency, resilient proxy-rotation layers and memory-isolated request dispatchers to prevent packet-level anomalies from triggering false-positive security blocks

SignalSnitch · Non-invasive network-level observability engine

Core stack: Go · Python · FastAPI · ClickHouse · Docker · React · Tailwind CSS

- ▶ Engineered a protocol-agnostic, low-level observability tool that intercepts raw kernel-level packet streams (TCP ports 9092, 5672) using custom **Berkeley Packet Filters (BPF)** to map real-time distributed-system topologies, with no code instrumentation or app SDKs required
- ▶ Designed a high-throughput multi-threaded parsing engine in **Go**, decoupling stream capture via goroutines and memory-efficient channels, using explicit type-assertion to extract unfragmented serialization frames
- ▶ Architected an OLAP ingestion pipeline built around a centralized **FastAPI** aggregator backed by a columnar **ClickHouse** (MergeTree) time-series warehouse, cutting analytical graph-processing latency across millions of captured packet-metadata rows
- ▶ Built a dynamic **React/Next.js** interactive map using network-graph theory to visualize directional microservice traffic, scaling stroke width and color live against packet volume