

Olufemi Adesina

SOFTWARE ENGINEER | BACKEND DEVELOPER | DEVOPS & SOLUTIONS

ARCHITECT

+447494867597 | femadesina1125@gmail.com | Nottingham, UK | [LinkedIn](#) | [Portfolio](#) | [Github](#)

Backend Leaning Full-Stack Engineer (TypeScript | Python | Terraform | Azure DevOps)

Backend-Leaning Full Stack Python and TypeScript Developer & Cloud Solutions Architect with 8+ years' experience building scalable, high-performance systems for data-intensive applications. Specialised in designing and deploying production-grade APIs using Express and FastAPI/Django stacks with expertise in PostgreSQL for schema design, indexing, and query optimisation on small and large complex workloads. Experienced in architecting microservices and data pipeline infrastructure within GCP, Azure and AWS, leveraging containerised deployments through Docker and Terraform and serverless based functions using Google Cloud Functions.

Proficient in building robust CI/CD pipelines using GitHub Actions and Azure DevOps, implementing automated testing strategies, and establishing quality gates for predictable and reliable releases.

Implemented authentication/authorisation patterns; particularly involving JWT-based systems, and worked in equivalent BigQuery-like services on other platforms (Redshift) for analytics workloads.

I enjoy working in fast-paced, ambiguous environments typical of scaling startups. I usually take ownership of technical decisions and determine tasks alongside Project Managers and Business Analysts while collaborating closely with cross-functional teams including other developers, data scientists and product leadership. Continuous improvement is important to me.

CORE:

Cloud & IaC:

Microsoft Azure (App Services, AKS, Functions, VNets, App Gateway), Terraform, ARM Templates, Azure Monitor, Azure Log Analytics, Kubernetes, Docker, Azure CLI, Azure API Management, Azure Bastion, Azure DB - specific service for specific engine)

AWS (Lambda, S3, CloudFront, CloudFormation, VPC, Load Balancers, App Scaling Groups, Aurora, RDS, DynamoDB, Glue, Kinesis, Kinesis Firehose, General Sharding, Elasticache, Beanstalk, API Gateway, Fargate, Route53, Parameter Store)

DevOps & CI/CD:

Azure DevOps Pipelines, GitHub Actions, BitBucket Pipelines, YAML automation, CI/CD orchestration, Infrastructure as Code deployment automation, AWS CodeDeploy

Monitoring & Observability:

AWS CloudWatch, Azure Application Insights, Datadog, Prometheus, Grafana, Log Analytics, alerting pipelines & incident response bespoke system

Languages & Scripting:

PowerShell, Linux, Python, JavaScript, TypeScript (Node.js, Express, FastAPI), C#, GoLang

Networking & Security:

Azure Active Directory Management, VNet and VNet peering, DNS (Route 53) management, Load Balancers, Azure Key Vault, AWS Secrets Manager, CI/CD security,

Leadership Quality:

Agile Delivery Planning & Retrospection (Often led), Sprint Backlog Ownership, Infrastructure Planning, Mentorship and leading education sessions, Worked closely with CTO and effective stakeholders in collaboration

Certifications:

- AWS Solutions Architect Associate (Received Jan 19)
- Microsoft Certified: Azure Administrator Associate (in progress)

PROFESSIONAL EXPERIENCE

May 2022 — Present | Senior Full Stack Engineer | Senior Principal Cloud Engineer

Service Robotics Ltd

- Architected and deployed scalable cloud infrastructure for enterprise health and social-care platform using AWS and Terraform, supporting data-intensive analytics workloads serving thousands of daily users across multiple client organizations
- Led infrastructure migration from monolithic EC2 architecture to microservices-based approach combining serverless (Lambda) and containerized services (ECS/Fargate), improving API response times by 60% and enabling independent scaling of critical components
- Designed and implemented automated data pipeline infrastructure using Lambda scheduling with ACID-compliant ingestion processes, audit logging in DynamoDB, and full Infrastructure-as-Code deployment via Terraform for multiple client-facing analytics dashboards
- Managed PostgreSQL infrastructure at scale, renovating older architecture to modern practices of Database compliance, implemented targeted indexing, partitioning strategies, and query optimization that reduced complex reporting endpoint load times from 8s to under 500ms
- Established comprehensive CI/CD pipelines and automated testing culture using GitHub Actions, implementing unit tests (Jest), integration tests, and quality measures (ESLint, Prettier) that reduced production incidents by 40% and accelerated release cycles
- Built FastAPI backend services with proper error handling, request validation (Pydantic), and performance optimization for high-throughput analytics workloads
- Implemented JWT-based authentication and role-based authorization (RBAC) across GraphQL and REST APIs, ensuring secure multi-tenant access patterns with audit trails for healthcare compliance requirements
- Architected async event-driven workflows using SNS→SQS→Lambda patterns with rate limiting, guards, and CloudWatch observability for AI-powered conversational features (OpenAI/Claude APIs)
- Provided technical mentorship to junior developers and co-led AWS certification preparation sessions alongside CTO, fostering DevOps best practices across the engineering team
- Operated pragmatically in startup environment with evolving requirements, balancing technical debt against feature velocity

Technologies: Python (FastAPI), PostgreSQL, AWS (Lambda, RDS, SNS/SQS, ECS/Fargate, DynamoDB, CodeDeploy, CloudFormation, Route 53), Terraform, GraphQL, TypeScript/Node, GitHub Actions, Jest, Cypress, OpenAI APIs, BitBucket Pipelines, C# (External Project - iOS App/CRM)

August 2018 — May 2022 | Full-Stack Developer

Firemind Ltd

- Architected and deployed AWS and Azure infrastructure for client engagements using Terraform (Infrastructure-as-Code), provisioning EC2, Lambda, Fargate, RDS, and S3 with security-first configurations ensuring consistency and scalability across environments
- Designed secure Python APIs using FastAPI and Node.js (Express) with RESTful endpoints including proper contract specifications, idempotency, pagination, and rate-limiting to ensure reliability under load
- Consulted with clients on AWS and Azure infrastructure optimization, conducting security assessments and right-sizing initiatives that frequently led to expanded engagements due to client satisfaction
- Implemented Redis caching layer (AWS ElastiCache) for high-traffic endpoints, reducing repeated database reads and stabilizing session handling during peak loads with budget-conscious scheduled scaling
- Built GraphQL APIs consolidating data from multiple backend services, reducing network overhead and enabling efficient client-side data fetching with field-level granularity
- Developed data pipeline and ETL infrastructure using AWS Redshift for analytics workloads on specific client projects
- Managed CI/CD workflows and deployment automation across multiple client projects, establishing quality gates and testing strategies
- Collaborated within Agile cross-functional teams, translating business requirements from stakeholders into pragmatic technical solutions and coordinating legacy system migrations to cloud-native architectures with minimal downtime
- Mentored junior developers and co-led AWS certification training sessions with CTO and Senior Cloud Solutions Architect, establishing knowledge-sharing practices and raising team technical capabilities

Technologies: AngularJS, jQuery, Angular, React, Typescript (Node.js), Python (FastAPI), ExpressJS, REST APIs, GraphQL, Jest, Sinon, Cypress, AWS, Azure, MySQL, PostgreSQL, Redis (Elasticache), MongoDB, Electron, ETL, Redshift

EDUCATION

Oct 2015 — May 2018 | BSc Computer Science | 1st Honours

University of Essex

Completed courses in:

- Advanced Web Applications,
- Database Security, Artificial Intelligence (AI)
- Machine Learning targeting Neural Networks and Natural Language Programming (NLP),
- Operating Systems
- Robot Engineering (C++)

SKILLS & ABILITIES

General Skills

• Evaluation & Assistance

- Regularly assist during project planning phases, specialising in API design and network and infrastructure architecture. Resolve potential flaws.

• Adaptability

- Rapidly onboard onto new codebases and frameworks, even in complex or legacy systems. Experienced working with varied technology stacks and shifting project requirements, maintaining delivery pace without sacrificing quality.

• Problem Solving

- Lead discovery and research phases to define optimal solutions. Analyze trade-offs (performance, cost, scalability) and make informed decisions.

• Documentation

- Typically responsible for producing technical and security documentation, maintain knowledge bases, and work closely with stakeholders and designers. Clear, proactive communication ensures teams stay aligned, deadlines are met, and risk is mitigated.

• Meeting Tight Deadlines & Organised Communication

- Operate in fast-paced environments. Manage multiple priorities and meet tight deadlines ensuring effective communication with teams and keeping deliverables high quality. Open channels of feedback, anticipate issues early and adapt plans accordingly.

Technical Skills

Programming & Backend Development

- Python (8/10) | TypeScript (9/10) | Go (6/10) | Java (5/10) | C# (6/10) | Rust (4/10)

- Python:

- 6+ years developing production-grade backend systems using FastAPI and Django (ORM, REST APIs, Django-Ninja) for data-intensive applications and microservices architectures.
- Built scalable APIs handling high-throughput workloads with proper error handling, request validation (Pydantic), and performance optimization for data pipeline and analytics use cases.
- Extensive experience deploying Python backends to GCP, AWS, and Azure using containerized (Docker) and serverless approaches with comprehensive automated testing (pytest, unittest) integrated into CI/CD pipelines.
- Proficient in data processing libraries (Pandas, NumPy) for ETL workflows, data transformation, and analytics backend features serving business intelligence dashboards.

- TypeScript:

- 7 years building backend microservices and APIs using Node.js/Express and NestJS, with strong focus on type safety, shared contracts between services, and maintainable architectures.
- Comfortable with modern tooling (ESLint flat config, Prettier, pnpm/Yarn, tsx/ts-node) and optimization practices following ESM standards with Deno, Vite, and Bun.
- Advanced TypeScript features including generics, utility types, and type-safe API contracts for reduced runtime errors and improved developer experience.

API Development & Architecture

- REST API Design (9/10) | GraphQL (7/10) | gRPC (5/10)

- Designed and deployed production APIs using FastAPI (Python) and Express/NestJS (Node.js) with industry-standard patterns: idempotency, pagination, rate-limiting, proper HTTP semantics, and comprehensive error handling.
- Experience with Falcon framework patterns and lightweight, performance-focused API design principles.
- Built GraphQL APIs consolidating multiple data sources, reducing over-fetching and improving client performance through field-level query optimization.

- AI API Integration: Integrated OpenAI and Anthropic (Claude) APIs into production systems with prompt engineering, cost management, rate-limiting, and guardrails for safe conversational features.

Database & Data Infrastructure

- PostgreSQL (9/10) | MySQL (8/10) | BigQuery-equivalent experience (7/10) | Redis/ElastiCache (8/10)
 - Deep expertise in PostgreSQL including schema design, complex query optimization, indexing strategies, partitioning for scale, and performance tuning on data-heavy workloads.
 - Hands-on experience with data modeling for relational databases, designing normalized schemas and denormalized reporting tables for analytics performance.
 - Worked with AWS Redshift (BigQuery-equivalent) for analytics workloads, building aggregation pipelines and optimizing columnar storage queries.
 - Implemented Redis/ElastiCache for caching layers, session management, rate-limiting, and queue-backed async processing patterns.
 - Familiar with NoSQL (DynamoDB single-table design, MongoDB) for event-driven data and flexible document storage.

Cloud Infrastructure & DevOps

- Google Cloud Platform (8/10) | AWS (9/10) | Azure (7/10)
 - GCP Experience:
 - Deployed and managed Python microservices on Google Cloud Platform using Cloud Functions, Cloud Run, and GKE (Kubernetes Engine).
 - Experience with GCP data services and cloud-native architecture patterns, comfortable navigating GCP console and CLI tooling.
 - Familiar with BigQuery and GCP's analytics/data pipeline ecosystem for processing and serving large datasets.
- Infrastructure as Code (9/10):
 - Extensive Terraform and Pulumi experience provisioning and managing cloud infrastructure across GCP, AWS, and Azure with modular, reusable configurations.
 - Automated environment provisioning, networking, IAM policies, database instances, serverless functions, and monitoring infrastructure through version-controlled IaC.
- CI/CD & Automation (9/10):
 - Built robust CI/CD pipelines using GitHub Actions and Azure DevOps implementing automated testing strategies, quality gates (linting, security scanning), and reliable deployment workflows.
 - Established build pipelines with containers, automated quality checks (unit/integration/E2E tests), code coverage reporting, and multi-environment promotion (dev → staging → production).
- Containerization & Orchestration (8/10):
 - Provisioned, scaled, and maintained services using Docker and Kubernetes clusters with health checks, auto-scaling, rolling deployments, and resource management.
 - Experience with serverless architectures (AWS Lambda, GCP Cloud Functions) and hybrid approaches balancing cost, performance, and operational complexity.
- Authentication & Security
 - Strong understanding of authentication and authorization patterns including JWT-based systems, OAuth2 flows, and session management.
 - Experience implementing NextAuth-style authentication in Node.js backends and equivalent JWT patterns in Python (FastAPI/Django).
 - Security-conscious development including input validation, SQL injection prevention, secure credential management (AWS Secrets Manager, environment variables), and API rate-limiting.

Testing & Quality Assurance

- Unit Testing (9/10) | Integration Testing (8/10) | E2E Testing (7/10)
 - Coordinated automated testing culture with comprehensive test coverage using pytest (Python), Jest (TypeScript/Node), and Cypress (E2E).
 - Built test suites covering unit tests, integration tests with database fixtures, mocked external dependencies, and end-to-end user flows.
 - Integrated tests into CI/CD pipelines as quality gates, preventing regressions and ensuring reliable releases.

Monitoring & Observability

- Datadog (6/10) | Prometheus/Grafana (6/10) | CloudWatch (8/10)
 - Implemented application health monitoring, log aggregation, alerting, and distributed tracing to ensure uptime, track performance metrics, and enable rapid incident response.
 - Experience with cloud-native monitoring (CloudWatch, GCP Operations) and third-party observability platforms.

Frontend (Supporting Skills)

- React (7/10) | Vue (6/10) | HTML5/CSS3 | TailwindCSS | ShadCN

- Proficient in React for building responsive interfaces when full-stack context required, though primary focus is backend development.
- Comfortable collaborating with frontend teams and understanding client-side performance considerations for API design.

PROJECTS & FREELANCE

Omnistats | Sports Analytics Platform

- Architected data pipeline infrastructure to centralize multiple third-party sports data provider APIs, implementing intelligent caching layers and queue-backed asynchronous job processing for time-sensitive data updates.
- Designed PostgreSQL schema with optimized indexing for high-throughput read patterns serving real-time odds and statistics, complemented by DynamoDB for flexible event data storage where appropriate for cost and performance.
- Built scalable backend services handling thousands of concurrent API requests with rate-limiting, retry logic, and graceful degradation patterns typical of pragmatic startup environments balancing feature velocity with system reliability.
- Deployed using Infrastructure as Code (Terraform) with automated CI/CD pipelines ensuring consistent, repeatable deployments across environments.

- **Tools:** Python/Node.js, PostgreSQL, DynamoDB, Redis caching, AWS (Lambda, SQS), Terraform, CI/CD

Demo: omnistats.bet

Documint | Document Generation Service

- Designed and implemented serverless data pipeline using AWS Lambda to transform and generate large-scale PDFs (100+ pages), streaming output directly to S3 to prevent client-side memory issues and timeout failures.
- Architected event-driven workflow using SQS queues for reliable async processing, handling backpressure during peak loads and enabling horizontal scaling without infrastructure management overhead.
- Reduced client memory consumption by 90% and stabilized long-running document operations by offloading compute-intensive work to Lambda functions with proper error handling and retry mechanisms.
- Deployed using Serverless Framework and Terraform with comprehensive IAM policies following principle of least privilege for security best practices.

- **Tools:** Node.js, Puppeteer, AWS Lambda, SQS, S3, DynamoDB, IAM, Serverless Framework, Terraform

Scalexa | Cloud Migration & DevOps Consultancy

- Led cloud migration initiatives for clients transitioning workloads to AWS and Azure, establishing maintainable infrastructure foundations using Terraform (Infrastructure as Code) for repeatable, version-controlled provisioning.
- Built robust CI/CD pipelines using Azure DevOps and AWS CodePipeline with automated testing gates, quality checks, and multi-environment deployment strategies enabling faster, safer releases.
- Implemented baseline observability using Datadog and Prometheus for application monitoring, log aggregation, and alerting—critical for understanding system health in production and responding to incidents proactively.
- Developed NestJS backend microservices (modules, controllers, providers) with standardized testing patterns, implementing guards and interceptors for authentication, validation, and caching middleware to establish engineering best practices.
- Worked pragmatically with ambiguous requirements typical of early-stage companies, balancing technical debt against business velocity while maintaining system stability and security.

- **Tools:** NestJS, Python, Node.js, AWS, Azure, Terraform, Azure DevOps, Datadog, Prometheus, Docker, PostgreSQL

Velents | Event Management Platform

- Hardened production AWS deployment infrastructure and standardized CI/CD workflows using Make and pipeline automation, significantly improving release reliability and rollback procedures.
- Resolved critical production issues including Java runtime configuration, SSL certificate chain validation, and DNS routing problems—demonstrating ability to debug complex systems under pressure.
- Implemented transactional email infrastructure using Mailgun and AWS SES with proper deliverability configuration (SPF, DKIM, DMARC), stabilizing key user notification flows and reducing bounce rates.
- Deployed infrastructure improvements using Terraform ensuring reproducible environments and reducing manual configuration drift.

- **Tools:** AWS (EC2, S3, SES, Route53), Terraform, Linux, Java, Node.js, SSL/TLS, DNS, Make