

ERICK OCHIENG' ONYANGO

Software Engineer

+254 792 417 156

Nairobi, Kenya

onyangoricks@gmail.com

github.com/kglllev

Technical Skills

Languages

Elixir, TypeScript, Go, Dart, Python

Backend

Phoenix, NestJS, FastAPI, Chi (Go), Absinthe (GraphQL), Ecto, GenStage, Oban, Broadway, BullMQ

Frontend & Mobile

React, Next.js, Vue.js, Nuxt, Flutter (BLoC), Tailwind CSS

Data

PostgreSQL, PostGIS, Redis, ETS/Nebulex, Elasticsearch, ClickHouse, Vespa

Messaging & Real-time

Kafka, RabbitMQ, Redis Streams/Pub-Sub, Phoenix Channels, SSE, WebSockets

Infrastructure

Docker, Kubernetes, AWS (S3, SQS, SSM), Caddy, Nginx, blue-green deploys, pgBackRest, Grafana/Loki

Payments

Paystack, M-Pesa, Stripe

Education

BSc Information Technology

Meru University of Science and Technology

2013 – 2017

Volunteering

Aflatoun Trainer

Aflatoun International

2016 – Present

Links

github.com/kglllev

instaescrow.ke

Experience

• InstaEscrow.ke

Founder & Principal Engineer

2024 – Present • Nairobi, Kenya

Solo-designed and built InstaEscrow (instaescrow.ke) from scratch—a payment escrow platform that protects buyers and sellers in Kenya's social commerce market by holding M-Pesa and card payments until delivery is confirmed.

- Architected the payment and escrow engine (Elixir/Phoenix): wallet management, hold-and-release flows, dispute resolution, seller verification, and Paystack/M-Pesa integration with idempotent webhook reconciliation and Oban-backed retry guarantees
- Built the IAM service (Elixir/Phoenix): phone/email OTP auth, Google OAuth, JWT tokens, multi-account-type support (vendors and buyers), verification document uploads, and fan-out workers syncing user data to downstream services via Oban
- Designed a real-time event delivery service using Redis Streams with per-user XREAD BLOCK, missed-event replay via Last-Event-ID, presence tracking with TTL-based heartbeats, and per-user connection caps—serving SSE to web and mobile clients
- Developed a multi-channel notification platform (NestJS/BullMQ) delivering push, SMS, WhatsApp, and email with campaign scheduling and delivery tracking
- Built an AI-powered analytics engine (Python/FastAPI) that converts natural language questions into SQL queries across multiple service databases using Ollama/Qwen LLM, with role-based schema access, query history, and results streamed to clients via SSE
- Developed an AI chat service (Python/FastAPI) with RAG-based knowledge retrieval, multi-tenant conversation management, and escalation workflows—embedded as a support widget in the platform
- Built the production infrastructure: Docker Compose orchestration, Caddy reverse proxy, zero-downtime blue-green deploys via Caddy Admin API, pgBackRest database backups, and Grafana/Loki/Promtail observability stack
- Shipped the vendor dashboard and buyer flows in Next.js with real-time order tracking via SSE, and a Flutter mobile app with BLoC state management, invoice generation, and QR-code payment links

Stack: Elixir, Phoenix, NestJS, Python, FastAPI, Ollama, PostgreSQL, Redis, Paystack, M-Pesa, Docker, Caddy, Next.js, Flutter

• SafeBoda

Senior Software Engineer

2021 – Present • Uganda & Kenya

One of East Africa's largest ride-hailing platforms. Core contributor across identity, dispatch, payments, notifications, and internal tooling.

- Led the rewrite of the identity and access management service from Node.js to Elixir/Phoenix, adding a GraphQL API (Absinthe), JWT/TOTP auth, and token blacklisting via Nebulex/Redis—reducing auth latency and eliminating session-fixation vulnerabilities
- Extended the real-time dispatch engine—a stateless Elixir service using finite state machines for trip lifecycle, Phoenix Channels for live updates, and a matching funnel processing thousands of ride requests daily
- Built a Kafka event processor (NestJS) powering driver earnings, loyalty points, automated suspension, and SOS coordination, with BullMQ for reliable async delivery
- Developed the notification platform handling push (FCM), SMS (Africa's Talking), WhatsApp, and email across millions of monthly deliveries with queue-based fan-out and campaign scheduling
- Built the internal ops portal (Vue.js) for driver/rider management, trip monitoring, financial reconciliation, KYC, surge pricing, and dispatch configuration

Stack: Elixir, Phoenix, NestJS, Kafka, RabbitMQ, PostgreSQL, Redis, GraphQL, Vue.js, Firebase, AWS, Docker, Kubernetes

- **Mindful Care**

- **Full Stack Engineer**

- 2018 – 2021 • New York, USA (Remote)

- Mental healthcare company operating across 8+ US states (NY, NJ, CT, FL, MI, IL, IN, PA), offering psychiatric urgent care, therapy, TMS, and their proprietary MicroTherapy™ sessions. 450+ five-star reviews.

- Architected and maintained the patient-facing web platform and internal clinical tools in Elixir/Phoenix with LiveView and Tailwind—powering online booking, same-day/next-day appointment scheduling, and the hybrid in-person/telehealth experience across all locations
 - Built integration adapters synchronising patient records, appointment scheduling, and billing across DrChrono (EHR), Active Campaign (CRM), and Daily.co (telehealth video)—replacing manual workflows and enabling the multi-state expansion
 - Implemented insurance verification and payment processing with Stripe, supporting multiple payer networks (Aetna, Cigna, Oxford, Medicare/Medicaid managed plans)
 - Built automated patient communication pipelines via Twilio SMS/voice for appointment reminders, follow-ups, and care coordination across the growing provider network

- **Stack:** Elixir, Phoenix, LiveView, Tailwind, Stripe, Twilio, Daily.co

- **ImagineAPI**

- **Software Engineer, Contract**

- Oct 2024 – Feb 2025

- Real-time AI image generation platform routing prompts to MidJourney and streaming results back to users.

- Designed back-pressured pipelines using Redis Streams and GenStage, enabling the system to absorb traffic spikes without dropping events or degrading latency
 - Built image-splitting algorithms to decompose MidJourney grid outputs into individual images with concurrent processing across pipeline stages

- **Stack:** Elixir, Redis Streams, GenStage, Docker

References

Available on request.