VICTOR KIPRUTO ROP

Data Engineer

Nairobi, Kenya · +254 723 484 552 · kiprutovictor39@gmail.com

github.com/kipruto45 · victor-kipruto-rop-vzgdni.aptifolio.app

SUMMARY

Data Engineering with production experience in ETL pipeline design, real-time streaming, and dimensional warehousing. I build data systems the way they should work in production-idempotent, tested, containerised, and monitored. Comfortable across the full data stack from raw ingestion to Grafana dashboards, with every project backed by CI/CD automation on GitHub Actions.

TECHNICAL SKILLS

Languages	Python, SQL (PostgreSQL, MySQL), Shell / Bash
ETL & Pipelines	Apache Airflow (DAG design), Apache Kafka, Pandas, NumPy, SQLAlchemy, Faker
Cloud & DevOps	AWS (S3, EC2), Docker, Docker Compose, GitHub Actions
Warehousing	Star Schema, Dimensional Modelling, SCD Type 2, Fact & Dimension Tables
Data Quality	Idempotent Upserts, Deduplication, Validation Engines, Pytest (Unit & Integration)
Visualisation	Grafana, Plotly, Jupyter Notebooks

PROJECTS

Kenyan Market ETL Pipeline *github.com/kipruto45*

Python · Pandas · PostgreSQL · SQLAlchemy · Grafana · GitHub Actions

- Designed an idempotent upsert pipeline so the job can re-run at any point without producing duplicate rows — a deliberate production constraint, not an afterthought.
- Wrote custom deduplication logic that aggregates quantities on conflicting records rather than dropping them, preserving data fidelity.
- Achieved 38/38 passing unit tests covering null injection, type coercion failures, and empty dataset scenarios. CI runs on Python 3.8 – 3.11 via GitHub Actions with Black, isort, and mypy gating every push.
- Connected the PostgreSQL output layer to Grafana for live operational dashboards on market data trends.

Stack: Python, Pandas, PostgreSQL, SQLAlchemy, Grafana, Pytest, GitHub Actions

M-Pesa Airflow Transaction Pipeline *github.com/kipruto45*

Apache Airflow · Python · Faker · PostgreSQL · Docker Compose · GitHub Actions

- Built a DAG-orchestrated ETL pipeline for M-Pesa transaction processing with a configurable synthetic data generator producing realistic Kenyan phone number distributions using Faker.
- Implemented a multi-stage cleaning pipeline (duplicate removal, null handling, type coercion) driven by a rule engine — rules are config-driven, not hardcoded.
- Containerised the complete Airflow + PostgreSQL stack with Docker Compose; environment is fully reproducible with a single command.
- CI/CD pipeline runs on Python 3.9 – 3.12 via GitHub Actions with 100% test pass rate gating all merges.

Stack: Apache Airflow, Python, Faker, PostgreSQL, Docker Compose, Pytest, GitHub Actions

Real-Time Transaction Streaming System *github.com/kipruto45*

Apache Kafka · Python · Jupyter · Plotly · Docker · GitHub Actions

- Built a Kafka producer/consumer model for live transaction ingestion with real-time message aggregation and low-latency throughput.
- Implemented consumer group design for horizontal scalability and configured exactly-once semantics to prevent double-counting in downstream aggregations.

- Produced live visualisation outputs directly from the consumer layer using Plotly inside Jupyter, giving immediate feedback on streaming data.
- Containerised the full Kafka + Zookeeper stack with Docker Compose; GitHub Actions validates Kafka setup on every commit.

Stack: Apache Kafka, Python, Jupyter, Plotly, Docker Compose, GitHub Actions

ADDITIONAL EXPERIENCE

Cloud ETL Ingestion Pipeline*Large-Scale Batch Processing*

Python · Docker · AWS S3 · PostgreSQL · SQLAlchemy

- Processed 1.6 M+ rows at 70,000 rows/sec with peak memory under 300 MB, achieved via chunked reading and automatic encoding detection.
- Cut full load time from 4 minutes to 45 seconds by tuning connection pooling and switching to 5,000-row batch execution.
- Separated Extract, Transform, and Load into independent Docker microservices with exponential-backoff retry on transient cloud failures.

End-to-End SQL Pipeline Portfolio*OLTP to Dimensional Warehouse*

Python · PostgreSQL · AWS S3 · SQLAlchemy · SCD Type 2

- Designed a multi-tier data lake pattern from CSV staging through OLTP models into a star schema warehouse.
- Deployed SCD Type 2 on customer dimensions to preserve full historical change timelines.
- Wrote recursive CTE queries to surface cross-platform revenue shifts, fraud signals, and customer retention metrics.

EDUCATION

BSc Data Science*Expected August 2028*

The Cooperative University of Kenya, Nairobi · Member, Data Science Club: regional challenges, workshops, industry lectures

Kenya Certificate of Secondary Education (KCSE)*2023*

St. Patrick's High School, Iten · Volunteer Maths & Science Tutor, Community Tutoring Program

---

References and project walkthroughs available on request