

Stockify HFT Engine - App Summary

What it is

A high-frequency options trading platform with real-time analytics, Greeks calculation, and an event-driven architecture. Built as microservices using FastAPI, Kafka, TimescaleDB, and Redis.

Who it's for

Not found in repo. (Implied from README: teams building or operating options trading analytics/HFT pipelines.)

What it does

- Real-time market data ingestion from the Dhan API with circuit breakers.
- Publishes and consumes streaming data via Kafka using Avro serialization.
- Computes Greeks (BSM), IV skew, and gamma exposure with vectorized math.
- Persists enriched time-series data to TimescaleDB.
- Serves REST APIs for frontend clients (snapshots, historical data, profiles).
- Streams live updates to clients over WebSocket.
- Caches hot data and supports deduplication with Redis.

How it works

Dhan API -> Ingestion Service -> Kafka (Avro) for raw market data events.

Processor Service consumes Kafka, computes analytics, and forwards enriched data.

Storage Service writes to TimescaleDB for time-series persistence.

Realtime Service streams updates to the frontend via WebSocket.

OCD Backend provides REST APIs; Core library shares models/analytics; Redis caches hot data.

How to run (minimal)

- Install Docker Engine and Docker Compose v2+.
- Create .env from .env.example and set required vars: POSTGRES_PASSWORD, SECRET_KEY, DHAN_CLIENT_ID, DHAN_ACCESS_TOKEN.
- Run: docker compose up -d --build.