

SALOHIDDIN YO'LDOSHEV

Full-Stack Developer (Python / FastAPI)

Tashkent, Uzbekistan | ysalohiddin02@gmail.com | github.com/NeoCode-02 | tongyulduzi.uz

PROFILE

Full-stack developer from Tashkent, mostly backend. I built **tongyulduzi.uz**, the website and publishing system of a national newspaper. FastAPI and PostgreSQL on the back, Next.js on the front. I took it from an empty database to a live product that editors and readers use every day, and handed it over to the team, who run it on their own now. I like being responsible for the whole thing: the API and the schema, the interface on top of it, and keeping it online after launch. Available to start immediately, comfortable working remotely, English B2.

EXPERIENCE

Full-Stack Developer, Tong Yulduzi

Freelance | 2026

Python, FastAPI, PostgreSQL, Redis, Next.js / React, TypeScript, Docker, Nginx

tongyulduzi.uz

- Backend and data.** Built the FastAPI backend for a national children's and teenagers' newspaper: REST APIs for articles, media and the issue archive, and the PostgreSQL schema behind them with SQLAlchemy models and Alembic migrations.
- Search and relevance.** Built article search across the whole issue archive on PostgreSQL full-text search, using tsvector columns with GIN indexes and ts_rank to order results by relevance. I tuned the ranking against what editors and readers were actually searching for, so the article people wanted came up first instead of the newest one.
- Performance.** Made pages load faster by caching the most-read ones in Redis and adding the indexes the slow queries were missing. I measured the timings before and after each change instead of guessing.
- Frontend.** Built the reader-facing site and the editors' publishing tool in Next.js from the agreed designs, matching the spacing, type and colours closely, and made both work properly on the phones most of our readers come from.
- Auth and roles.** Added role-based accounts (writer, editor, admin) with JWT auth, so the editorial team runs the paper on its own and does not need a developer for daily work.
- CI/CD.** Set up deployment through GitHub Actions. Every push runs the pytest suite against a throwaway database, builds the Docker image, and rolls back to the previous one automatically if the health check fails, so readers never see the site go down.
- Reliability and infra.** Scheduled nightly encrypted pg_dump backups, stored away from the server, and ran a full restore rehearsal to prove the newspaper's archive cannot be lost. Kept costs low by running the whole thing as one Docker Compose stack behind Nginx on a single small VPS.

PROJECTS

Pulse, Real-Time Chat & Social Platform

github.com/NeoCode-02/social-media

FastAPI, PostgreSQL, Redis, Docker

- Real-time messaging over WebSockets, with Redis Pub/Sub passing messages between worker processes so everyone in a conversation sees them instantly, whichever worker they are connected to.
- Users, conversations and message history in PostgreSQL, with SQLAlchemy and Alembic migrations. One docker compose up brings up the API, database and Redis on any machine.

AuraFlix, Video Streaming Platform

github.com/NeoCode-02/AuraFlix

FastAPI, PostgreSQL, Redis

- Served video in chunks using HTTP range requests, so playback starts within a second or two instead of making the viewer wait for the whole file to download.
- Moved view counts into Redis and flush them into PostgreSQL in batches from a background worker, instead of writing to the database on every play.

E-commerce Backend

github.com/NeoCode-02/Ecommerce

Django, PostgreSQL

- Product catalogue, shopping basket and checkout on the Django ORM over PostgreSQL, with third-party payment and email integrations, and the Django admin as the shop owner's control panel.
- Customer reviews with ratings, and an audit log that records who changed what and when.

SKILLS

Languages Python, SQL, JavaScript / TypeScript

Backend FastAPI, Django, REST API design, PostgreSQL (schema design, indexing, query tuning, full-text search), Redis, Celery, SQLAlchemy, Alembic, JWT auth, third-party API integrations

Frontend Next.js, React, TypeScript, HTML & CSS, responsive UI built from design files

Infra & Tools Docker & Docker Compose, GitHub Actions (CI/CD), Linux servers, Nginx, pytest, Git & GitHub, AI tools (Claude Code, Codex, OpenCode)

CERTIFICATIONS & LANGUAGES

UIC Academy, Backend Developer Course (2025)

Uzbek: Native

AWS Certified Cloud Practitioner Essentials

English: B2 (Upper-Intermediate)