Roman Shalashov

Full-Stack Developer / Python Backend

roman-webdev.vercel.app | github.com/roman-webdev

kzena2000@gmail.com | Telegram @RN_SV7

Profile

Full-Stack Developer focused on Python APIs, database-backed web applications and Telegram automation. Portfolio projects cover booking and CRM, e-commerce, CSV processing and customer operations. Technical work emphasizes validation, transaction integrity, clear user workflows and explicit demo and release boundaries.

Technical Skills

- **Backend:** Python, FastAPI, Flask, REST APIs, aiogram 3
- **Data:** PostgreSQL, SQLite, SQLAlchemy, Alembic, Psycopg
- **Frontend:** React, TypeScript, JavaScript, HTML, CSS, Vite
- **Quality and integration:** pytest, unittest, Vitest, CSRF, session auth, RBAC, webhook HMAC, Stripe test API
- **Delivery tools:** Git/GitHub, Render, Neon, Vercel; Cloudflare Workers, D1, Queues and Cron in an experimental lab

Personal Portfolio Projects

Independent developer (AI-assisted). Developed these projects independently with AI tools; practical work covers application code, tests and integrations. No commercial IT work experience.

AstraSynq | CSV automation and integrations

Release Candidate 0.5.0-rc.1; synthetic demo

GitHub

- CSV mapping, validation and duplicate review; transactional import/outbox workflow with session authentication, CSRF protection and role-based access.
- Worker leases, fencing and backoff retries support at-least-once delivery. The public demo uses simulated delivery and has no persistent worker.

Stack: FastAPI, React, TypeScript, PostgreSQL, SQLAlchemy, Alembic

MANOR HOUSE | Booking and CRM

Live public booking showcase; restricted admin CRM

GitHub

- Service and barber selection, schedule-based availability, booking and rescheduling. Availability is rechecked within write transactions protected by PostgreSQL advisory locks.
- Protected admin CRM includes client notes and tags, schedule exceptions, catalog management, manual reminder tracking and analytics. Telegram notification adapter is present.

Stack: Python, Flask, PostgreSQL, Psycopg, JavaScript

ShopFlow | E-commerce

Portfolio demo; Stripe test or mock checkout

GitHub

- Product variants, persistent cart, validated checkout, database-backed orders and a protected admin dashboard with a multilingual interface.
- CSRF protection, signed Stripe webhook validation and explicit rejection of live payment keys. Test cases cover order persistence, retries and integration failures.

Stack: Python, Flask, PostgreSQL, SQLAlchemy, JavaScript, Stripe test API

HELIXPRIMUS | Customer operations

Synthetic demo; deterministic mock analysis and delivery

GitHub

- Inbox triage, exact policy matching and an editable reply workflow with persisted approval, guarded mock sending and audit history; SLA filters and operational analytics.
- Rule-based provider contract with fixed confidence estimates. Mock Send records a reply without contacting a customer; no external LLM or RAG. Shared temporary SQLite; no authentication or concurrency guarantees.

Stack: FastAPI, React, TypeScript, SQLAlchemy, Alembic, SQLite

SmartSave | Telegram finance assistant

Local Beta 1.3.15-beta1 (RC); public documentation-only showcase 1.3.14

Public documentation - 1.3.14

- Income and expenses, accounts and transfers, budgets and savings goals, CSV preview/confirmation and duplicate protection, monthly reports and local financial insights.
- Current-screen transaction UX and stale/replayed callback safeguards. Offline regression checks are documented; native Oracle Linux and live Telegram acceptance remain pending.

Stack: Python, aiogram 3, SQLite, unittest, CSV

DRIVEFIX | Service website and reliability lab

Browser demo, legacy API and separate experimental lab

Legacy source and revision notes

- Current Cloudflare Pages showcase provides service selection and browser-only synthetic confirmation. Legacy Flask/Render source contains validation and a Telegram adapter.
- Separate mock-only Cloudflare lab explores durable booking/outbox recovery, idempotency and crash/restart scenarios. Local verification is documented; cloud deployment, cloud failure test and production remain on hold.

Stack: HTML, CSS, JavaScript; legacy Flask; Workers, D1, Queues, Cron

Education

Bachelor's degree in Cybersecurity

V. N. Karazin Kharkiv National University

English

Working English: technical documentation, written and spoken communication