

Arthur Buikis

+371 27848052 | abuikis@inbox.lv | linkedin.com/in/artdeveloper | github.com/Artufe | Upwork | artufe.github.io |

Riga, LV

PROFESSIONAL SUMMARY

Senior backend / platform engineer with 8+ years building data-heavy systems in Python. At Giraffe 360 I own performance and reliability of the NAS CMS - a Django + Rust (PyO3) media-processing orchestration platform behind large-scale real-estate virtual tours. Shipped ~90 merged MRs across backend, ML pipeline, Kubernetes infra, and a Tauri desktop tool; introduced Rust into a mature Python codebase; cut hot-path endpoint latency by up to 800x; and act as the bridge between the ML/R&D team and the production pipeline for 12+ processing clients.

CORE COMPETENCIES

Python / Django

Rust / PyO3

PostgreSQL performance tuning

Kubernetes orchestration

Celery & async pipelines

ML pipeline engineering

API latency optimization

Feature-flag / release governance

WORK EXPERIENCE

Giraffe 360

Nov 2024 - Present

Software Developer - NAS CMS team - Riga, LV

- **Introduced Rust into a mature Python codebase.** Designed and shipped the first Rust extension (MR !1306) - PyO3 bindings with a maturin-built wheel providing O(1) graph reads on large projects. Expanded to a Rust critical-path ETA engine (!1329): DAG walk over unfinished jobs using p50 duration percentiles, PostgreSQL client pool (r2d2 + postgres crate) that releases the GIL. Stack crosses Rust, Python (DRF + Celery), and TypeScript (Tauri + React) in a single feature.
- **Cut hot-path endpoint latency by 30-800x.** Rearchitected work_started to bypass full graph reads (1,000 assets: 15.5s -> 19ms, constant-time, ~815x); eliminated N+1s in mark_artifacts_reviewed (500 stills: 215s -> 6.9s, ~31x); rewrote mark_unreviewed for panos (13.9s -> 0.08s, ~165x) and the upload endpoint (205s -> 128s via bulk_create_nodes/edges, with remaining work pushed to async Celery). Led the Q1 NAS API KPI.
- **Indexed a 10M-row webhook table with zero downtime.** Partial B-tree on STATE_INITIAL, GIN jsonb_path_ops index on payload, composite (state, created) - all shipped via non-atomic CREATE INDEX CONCURRENTLY Django migrations. Eliminated sequential scans on deployment-resume and daily project-filter queries.
- **Bridge to the ML team: integrated 12+ processing clients** into the production pipeline - pano image enhancement, sky replacement v2, monodepth, POI, pillar detection, house-number blur, image embeddings, TV replacement, and more. Each integration spans the processing-job schema, the graph engine's dependency model, and k8s deployment.
- **Built the shadow-deployment / safe-rollout system.** Across 5+ MRs (!1129, !1163, !1166, !1150, !1152), designed the quota-based experimental-features mechanism used to ramp new ML models into production without risking the full pipeline. Feature flags / dynamic settings govern 70+ namespaces and ~200 k8s artifact actions.
- **Kubernetes & reliability.** Owned TTL overrides with math expressions (!1250, !1233), job-creation retry logic (!1229), multi-environment env propagation (!1190), upload race-condition fixes (!1258), and pipeline-ordering corrections that prevented data corruption in the legacy pipeline (!1086, !1073, !1072).
- **Coplaner (workforce assignment).** Shipped real-time online tracking (!1215), role-based assignment permissions (!1214), and a dual-queue design (standard + student) backed by Redis Z-Sets with periodic rebuild and atomic pop.
- **Platform-wide performance wins.** Migrated serialization from stdlib json to orjson (!1278); introduced toggleable global GZip compression via dynamic settings (!1321); unified asset-type priority calculations (!1307); rewrote a RAM-leaking metrics job from mocked zeros to efficient DB aggregation.
- **Cross-functional delivery.** Wrote implementation plans before coding, ran two-stage reviews (spec compliance - code quality) per task, and coordinated wire contracts across the NAS CMS backend and the platform-lab Tauri/React desktop tool. Pair with DevOps on pre-prod sync and hand off optimized endpoints for regression testing.

Stack: Python - Django 3.2 + Channels - DRF - Celery - RabbitMQ - PostgreSQL - MongoDB - Redis - Rust - PyO3 - maturin - Kubernetes - Kueue - Triton - Tauri v2 - React/TypeScript - orjson - Metabase - GitLab CI - Sentry

Upwork

Nov 2017 - Present

Freelance Full-Stack Developer - Remote

- 12+ contracts in backend and automation with a **100% Job Success Score**.
- Delivered both embedded in client teams and solo; long-standing client relationships driven by clear communication and consistent delivery.
- Signature freelance projects: MarkFlow (social-analytics SaaS with investment secured), MyProxy (mobile-phone-as-4G/5G-proxy platform, still live), and the Expired Domain Search crawler (700M+ domains, 43 TB).

Strange Logic

Feb 2018 - Jan 2021 / Nov 2021 - Dec 2023

Senior Developer / Technical Support Specialist - Brighton, UK (Remote)

- Core engineering on TDN, Strange Logic's SaaS product. Streamlined L2/L3 support for the PHP application (**500+ tickets resolved**) and led the migration to Stripe for payments.
- On the second engagement, drove adoption of ClickHouse as a new back-end database and migrated core APIs from PHP to Python.
- Kicked off the Expired Domain Search and Mass VAT Checker initiatives.

Lethub

Jan 2021 - Nov 2021

Web Scraping Automation Contractor - Vancouver, BC (Remote)

- Built a Python-based pipeline to scrape listings from the three largest UK real-estate directories.
- Designed a database schema handling up to **4.3 TB** of listings, images, and historical sales records.

SELECTED PROJECTS

MarkFlow - markflow.eu

Solo-built social-analytics SaaS integrating Meta and Google APIs. Investment secured; private project.

MyProxy - client.myproxy.io

Designed a mobile-phone-as-4G/5G-proxy system with multi-device support, rotation, and a monitoring dashboard. **Reduced cost per GB of mobile proxy data by up to 20x**. Still live.

Expired Domain Search (Strange Logic)

Celery-based distributed system that crawled **700M+ domains** over a year - running DNS, WHOIS, page screenshots, tech-stack fingerprinting, and email extraction to surface buy-worthy expired domains for the TDN product. Generated **43 TB** of data across 12 servers.

TECHNICAL SKILLS

Languages: Python (primary), Rust, TypeScript/JavaScript, SQL, Bash, PHP

Backend & APIs: Django (ORM, migrations, signals, Channels), Django REST Framework, FastAPI, Flask, Celery, Tauri v2

Rust interop: PyO3, maturin, serde, r2d2, postgres crate

Databases: PostgreSQL (JSONB, GIN / partial / composite indexes, CREATE INDEX CONCURRENTLY, read-replica), MongoDB, ClickHouse, Redis (cache, broker, Z-Set priority queues), MySQL

Infra & DevOps: Kubernetes (GPU/CPU/Win), Kueue, RabbitMQ, Docker (Compose & Swarm), Nginx, Jenkins, GitLab CI, Linux

ML serving / data: Triton Inference Server, shadow deployments, quota-based rollouts, Metabase, S3, SMB/NAS, Jupyter

Frontend: React, TypeScript, Tauri v2 desktop apps, Vue.js, Jinja2

Observability: Sentry, Prometheus, Grafana, orjson, Discord alerting, feature-flag governance

Misc: Selenium, web scraping, Stripe, OAuth, pytest, asyncio

LANGUAGES

Latvian - Native **English** - Fluent **Russian** - Conversational **German** - Conversational