

Ilya Volkov

Backend / DevOps Engineer · Python, Kubernetes

Belgrade, Serbia · remote or hybrid in Belgrade

iillyyaa159@gmail.com

Telegram [@devtemp](https://t.me/devtemp)

github.com/iillyyaa1997

cv.unicorns.place

SUMMARY

In software development since 2017. For the past 6 years, I have worked on the backend and infrastructure of an online proctoring platform that runs in 8 regions and in dedicated client installations. Main stack: Python/Django, Celery, PostgreSQL, Kubernetes, Helm, GitLab CI. From 2021 to 2026, I was the release master and supported the platform in production. I work with Claude Code AI agents and integrate them into the team's workflows

EXPERIENCE

Constructor Tech (formerly Examus, Alemira) — Backend / DevOps Engineer

Nov 2020 – Oct 2026

Online proctoring platform · team of 10–20

After Alemira acquired Examus, I moved to Alemira, which was later renamed Constructor Tech

- Release master for the proctoring product: shipped releases every week, later every two weeks
- Supported Kubernetes and the backend in production: investigated and fixed the causes of incidents, improved monitoring in Prometheus and Grafana
- Helped 3+ engineers get up to speed on the project and develop their skills; advised the team on infrastructure and releases

Constructor Tech, Belgrade Aug 2023 – Oct 2026

- 2026 • Prepared the proctoring services for installation with the platform's on-prem installer: packaged 6 services into the installation bundle, moved secrets out of the code in 6 repositories, took the installation process through to an E2E check in a test environment
- 2026 • Built AI tools that the team used: Claude Code agent management, development and release skills for working with merge requests, Zulip and YouTrack, and a set of developer scripts
- 2026 • As maintainer of the open-source Moodle plugin, shipped all 2026 releases and made it compatible with Moodle 5 — github.com/constructor-tech/alemira-proctor-moodle-plugin
- 2026 • Tested AI agents and spec-driven development with Constructor Studio (formerly Cypilot) on an experimental port of the backend from Django to .NET. Prototype: 119 of 119 routes, about 1,400 tests, about 77% coverage
- 2025–2026 • Set up production and staging environments for the dedicated installations of two large clients — a university and a certification platform — and was the release manager for these installations: handled deployments, maintained separate release branches, troubleshoot rollout issues
- 2025–2026 • Wrote new APIs in .NET (Minimal API, EF Core) and moved parts of the Django backend to .NET
- 2023–2024 • Moved 8 services to shared Helm charts and a single configuration for Redis, RabbitMQ and PostgreSQL. Tuned PostgreSQL for growing load and moved the databases to Postgres Operator

Alemira, contract Mar 2022 – Jul 2023

- 2023 • Moved the platform from infrastructure in Russia to infrastructure abroad, using custom Helm charts and CI for separate GitLab and Kubernetes environments, and a Slack bot that synchronized branches between the GitLab instances
- 2022–2023 • Automated builds and releases of the Electron desktop client for Windows and macOS, including notarization: a macOS runner on EC2 that starts on demand, white-label builds for several brands, and alpha → canary → current channels. Builds are distributed through a CDN (AWS CloudFront)
- 2022–2023 • Wrote a shared Helm chart library and CI templates. Developed a Celery/RabbitMQ queue exporter whose metrics HPA used to scale the workers. Introduced werf and review environments, and deployed the platform to Azure AKS
- 2022–2023 • Added a separate retention period for each client and reworked the background cleanup

Examus Nov 2020 – Feb 2022

- 2021–2022 • Supported the regional installations together with an external DevOps team, as the person responsible for them on the company side
- 2021–2022 • Prepared the face identification service to run in several regions: photos in S3 with per-region buckets, admin sign-in through GitLab, Sentry for each environment

- 2021 • Implemented automatic deletion of exam video recordings after their retention period expires, as required by personal data protection law. Recordings are stored in S3; deletion runs in the background
- 2021 • Launched the production environment in the European region: CI/CD deployment and Helm charts with autoscaling (HPA, VPA)
- 2020–2021 • Set up autoscaling for ML services based on uWSGI and RabbitMQ metrics, as well as VPA

College Expert, Kazan — Backend / DevOps Engineer

Nov 2019 – Nov 2020

- Solely responsible for the backend and DevOps for two products: a mobile marketplace for services (iOS/Android) and the company's internal platform
- Developed the backend using Django/DRF with Channels (WebSocket) and Celery. Implemented two-stage payments with YooKassa, scheduling, push and SMS notifications, and map search with server-side clustering
- Independently set up the server and a self-hosted GitLab instance with runners for builds and deployments using docker-compose

Earlier, Kazan

2017 – 2019

Soft-Artel Feb 2019 – Sep 2019 • Node.js Developer: backend of the company's product – Node.js microservices with RabbitMQ and PostgreSQL; Docker builds and deployment to Kubernetes through GitLab CI

Softcraft Jun 2018 – Sep 2018 • Software Engineer: API for a customer's web service using C#/ASP.NET with Entity Framework and MSSQL

PROFIT Apr 2017 – May 2018 • Software Engineer: APIs and website scrapers in C# and Node.js, WinForms desktop applications

PERSONAL PROJECTS

- **Warren** — my own distributed system of nodes in different countries. I develop the whole product alone: architecture, backend, mobile, desktop and web clients, infrastructure and tests
 - management backend in Python (Litestar, PostgreSQL), node services in Go, client in Flutter: Android now, then iOS, macOS, Windows and Linux
 - nodes update themselves, and each new version goes through a canary check based on metrics; the infrastructure is defined in Ansible
 - E2E tests cover node failure, rollout and rollback
- **Kubernetes platform** — my own k3s cluster on two servers that runs my applications and self-hosted services: task tracker, wiki, photo gallery. I built it and run it alone
 - GitOps with Argo CD, self-hosted GitLab with runners, secrets in SealedSecrets, Longhorn storage, databases on CloudNativePG
 - monitoring with Prometheus, Loki and Grafana; alerts are sent to Telegram through a bot
 - database backups to B2 with restore checks; all applications deploy through shared CI components and a Helm library
- **uDeck** and **uZora** — open-source macOS apps in Swift — github.com/iillyyaa1997
 - uDeck — a plugin panel at the top edge of the screen, with a plugin catalog and E2E tests in disposable virtual machines
 - uZora — a menu bar agent that tracks the Mac's health (disk, temperature, battery) and passes the data to AI agents over MCP

SKILLS

Languages and platforms	Python (primary); also C#/.NET, Go, TypeScript, Swift, PHP, Bash
Backend	Django, DRF, Celery, FastAPI, .NET Minimal API, EF Core
Databases and queues	PostgreSQL, Redis, RabbitMQ, S3/MinIO
Kubernetes and CI/CD	Kubernetes (EKS, k3s), Helm, werf, Argo CD, GitLab CI, Docker
Cloud and IaC	AWS, Azure AKS, Terraform, Ansible
Monitoring	Prometheus, Grafana, Loki, Sentry, Datadog, OpenTelemetry
Integrations	Moodle, Canvas, D2L, Keycloak, Entra ID, LDAP
AI in development	Claude Code (agents, MCP, skills), Constructor Studio (formerly Cypilot)

EDUCATION AND LANGUAGES

- Interregional Competence Center (Kazan), 2013–2017. Secondary vocational diploma with honors, Software Technician
- Russian — native. English — B1: I read technical documentation and handle work correspondence