

Sardorbek Axadjonov

Backend Developer

+998 20 007 04 24 · axadjonovsardorbeck@gmail.com
Tashkent, Uzbekistan

[GitHub](#) · [LinkedIn](#) · [Telegram](#)

Backend developer with 2+ years of Go in production, building microservices and event-driven serverless systems for travel, e-commerce, and fleet-management platforms. At UDEVS I lead a 5-person cross-functional team delivering two Furasentr products, with 90%+ test coverage on booking and payment paths and k6 load tests at 500–700 RPS on the Knative deployment.

SKILLS

LANGUAGE

Go

DATA

PostgreSQL, pgvector, Redis, MongoDB, SQL

SERVICES

REST, gRPC, Gin, chi, Knative Eventing / CloudEvents

AUTH

JWT, OTP, Casbin (RBAC)

STORAGE

MinIO, S3, CDN

INFRA

Knative / FaaS, Docker, GitLab CI, goose, k6, Grafana

INTEGRATIONS

Click, Payme, Firebase FCM, Telegram, Google Gemini, Yandex Maps

ARCHITECTURE

Microservices, event-driven, serverless, monolithic

EDUCATION

Najot Ta'lim

Go Foundation + Bootcamp · Major: Golang

SPOKEN LANGUAGES

Uzbek (native) · Russian (B2) · English (B1)

EXPERIENCE

UDEVS — Backend Developer · Team Lead

Feb 2026 – Present

- Lead the project team for two Furasentr products — 5 people across frontend, backend, mobile, QA, and PM; task breakdown, code review, and release coordination on a Knative serverless, event-driven stack.
- Furasentr Autopark** (multi-tenant fleet SaaS): built the backend for four role-based app surfaces — per-company scoping enforced on every fleet call, trip / vehicle assignment, multi-device sessions, analytics, Yandex geocoder. Telemetry keeps per-trip state in cache and rejects out-of-order points; a fix within ~50 m folds into a stop cluster and never reaches the database, and the first point past it flushes one row with its dwell time — an hour parked writes **1** row, not ~180. Access checks cache the allow for 30 s and never the block, and blocking an owner cascades invalidation through every company and member.
- Furasentr** (truck-parts e-commerce, **11** Go services): built the ordering service — native Go PDF invoicing, Click and Payme over JSON-RPC, WebP / CDN images, Firebase fan-out — plus BuckyDrop purchase orders, Measoft delivery, and an OX ERP sync. Two-step AI image search over pgvector: Gemini crops every part, only the tapped crop is embedded and searched, and a miss widens the filter brand + model → brand → catalogue-wide. Catalog reads sit behind a 30 s cache fronted by *singleflight*, so a burst of identical requests becomes one query.
- CentralTour** (flight & hotel platform): owned the markup / commission engine — rules matched across **7** dimensions (platform, geography, aggregator, service, hotel, agency, agent profile), one winner per level by specificity; platform layers apply in parallel on a single net base while partner layers cascade (agency, then self), so no other agent's markup can leak in. Non-UZS fares priced in provider currency with FX frozen at quote, and every quote snapshotted so payment charges what was shown. Integrated GoGlobal, TBO, and Dida; built the admin-panel backend solo, with agency / platform scope filters on every query.
- Optimised Redis caching and parallelised independent queries in CRUD handlers that previously ran sequentially, across Furasentr and Autopark; load-tested with k6 to **700** and **500 RPS** on the Knative deployment, **90%+** test coverage on booking / payment paths, run in GitLab CI.

Cody Soft — Golang Developer (remote)

Sep 2024 – Feb 2026

- Built start-up products on microservice and monolithic architectures — gRPC between services, REST + Swagger at the edge, JWT + Casbin RBAC, PostgreSQL with Redis, MinIO media, goose migrations, Docker Compose.
- iMed** (online learning platform, imedteam.uz) — sole backend for web-admin and mobile REST APIs: phone-OTP + JWT with Casbin RBAC, Firebase push, a Telegram ops bot, and a YouTube→S3 pipeline (yt-dlp with a PO-token sidecar clearing bot checks without cookies) via CDN. In production: **8,000+** users, **2,000+** paid orders, **420M+** UZS.
- Click Merchant API (prepare / complete): MD5-signed, row-locked idempotent handlers (**SELECT ... FOR UPDATE**) so retried callbacks never double-charge or double-enrol, with a RESERVED → PAID / CANCELLED / EXPIRED order state machine.
- Promo-code engine — percent / fixed, per-user and total caps, validity windows, per-course scope — enforced by a redemptions ledger with partial unique indexes and cancel / expire release triggers.
- Versioned Redis cache — an *entity:version* counter bumped on every write is folded into the cache key, so stale lists expire by TTL with no SCAN / KEYS.

Vector-PVT — Golang Developer

May 2025 – Jul 2025

- Backend for the Vector-Taxi platform across user, admin, and business panels — RESTful APIs with Swagger, Casbin RBAC, and large PostgreSQL and MongoDB datasets.