Пётр Корсунский

Middle Python Backend Engineer

Python · Payments · Telegram Mini Apps · Voice AI · Web3
Украина (Remote) | moddiedev@gmail.com | t.me/moddie_dev | github.com/Modd1e
https://portfolio-moddie-dev.vercel.app/ru

Python Backend Engineer с 2+ годами опыта проектирования асинхронных бэкенд-систем, платежных решений, распределенных очередей задач и интеграции AI/GPU-инфраструктуры. Специализируюсь на FastAPI, PostgreSQL, Redis и оптимизации фоновых процессов под высокие требования к отказоустойчивости и транзакционной целостности. Имею практический опыт создания сквозных продуктов от архитектуры БД до клиентских интерфейсов на React / Next.js.

Основной стек: Python 3.11+, FastAPI, AsyncIO, Dishka, PostgreSQL, Redis, SQLAlchemy 2.0, Alembic, Celery / ARQ, Docker

ОПЫТ РАБОТЫ

Коммерческий проект (NDA) — AI Video & GPU Platform — Middle Python Backend Engineer

04.2026 — настоящее время | Remote / Постоянная занятость

• Спроектировал и внедрил автономный GPU-агент на RunPod: параллельная обработка задач из бэкенд-очереди с динамическим масштабированием воркеров.

• Реализовал алгоритм автомасштабирования (auto-scaling) и безопасного завершения процессов (graceful shutdown) для контейнеров, снизив издержки на простой GPU.

• Развернул пайплайны обработки медиа с аппаратным ускорением NVIDIA и локальным запуском LLM; оптимизировал пакетную обработку изображений.

• Провёл рефакторинг сервисов интеграции и устранил узкие места в обработке очередей, повысив общую стабильность бэкенда.

Stack: Python, FastAPI, AsyncIO, RunPod API, Docker, NVIDIA CUDA, Pillow, PostgreSQL, Redis

Voice AI Assistant — Python Backend Engineer

01.2026 — 04.2026 | Коммерческий проект (NDA)

• Спроектировал агентский сервис голосовой телефонии реального времени на базе MTProto (Pyrogram).

• Построил потоковый конвейер обработки аудио с минимальной задержкой через неблокирующие Linux pipes: VAD → Whisper STT → LLM (потоковая генерация токенов) → ElevenLabs TTS.

• Реализовал логику ведения диалога с прерыванием речи (interruption handling) и контекстным управлением состоянием звонка.

Stack: Python, AsyncIO, Pyrogram, ElevenLabs API, Whisper, VAD, Linux IPC / Pipes

VolGen — Backend / Web3 Engineer

09.2025 — 01.2026 | Коммерческий проект

• Разработал финансовое ядро торговой системы в сети Base: операции со смарт-контрактами через Web3.py, свопы на Aerodrome DEX и интеграция с API Binance.

• Спроектировал модуль управления транзакциями с контролем nonce, динамическим пересчетом газа и гарантией идемпотентности.

• Реализовал сохранение промежуточных состояний пайплайна в PostgreSQL, обеспечив восстановление незавершенных операций при рестартах воркеров.

Stack: Python, FastAPI, Web3.py, EVM, Base, PostgreSQL, Redis, ARQ, Docker

Опыт работы (продолжение)

RYFT Core — Backend Engineer

05.2025 — 09.2025 | Коммерческий проект

• Спроектировал и реализовал серверную часть VPN-сервиса: биллинг, жизненный цикл подписок, учет клиентских устройств и ограничения сессий.

• Реализовал интеграцию с узлами Marzban и Xray-core для автоматической генерации профилей доступа, динамического отзыва ключей и сбора сетевой статистики.

• Настроил очереди ARQ для синхронизации состояний конфигураций с автоматическим разрешением сетевых таймаутов.

Stack: Python, FastAPI, PostgreSQL, Redis, ARQ, Marzban API, Xray-core, Docker

Danskyi TMA — Fullstack / Backend Engineer

11.2024 — 05.2025 (поддержка по наст. вр.) | Коммерческий проект

• С нуля разработал архитектуру сервиса онлайн-бронирования в формате Telegram Mini App: REST API, схему базы данных, админ-панель и клиентский интерфейс.

• Реализовал алгоритм валидации расписания и бронирования слотов с использованием блокировок в PostgreSQL, исключив овербукинг и race conditions.

• Интегрировал платежные шлюзы с обработкой вебхуков и асинхронную доставку уведомлений через бота на Aiogram и очередь ARQ.

Stack: Python, FastAPI, Dishka, PostgreSQL, Redis, ARQ, Aiogram 3, Next.js, TypeScript, Docker

ВЫБОРОЧНЫЕ ПРОЕКТЫ

Crypto Game TMA — Backend Engineer

2026 | Проектная разработка

• Спроектировал финансовый модуль с учетом транзакций по методу двойной записи (double-entry bookkeeping) для исключения балансовых расхождений.

• Вынес генерацию и подписание EVM/TON транзакций в изолированный микросервис с обменом сообщениями через gRPC и RabbitMQ.

Stack: Python, FastAPI, PostgreSQL, RabbitMQ, Celery, gRPC, Web3.py, Next.js

PaperPump — Blockchain / Indexer Engineer

2025 | Проектная разработка

• Разработал смарт-контракты на Tact и высокопроизводительный парсер блокчейна TON на TypeScript: обход дерева транзакций, парсинг по opcode и декодирование метаданных Jetton.

Stack: TypeScript, Tact, TON, TonAPI, @ton/core

CS2 Server Manager — Backend Developer

2025 | Проектная разработка

• Реализовал сервис обработки платежных вебхуков с автоматической доставкой привилегий на игровой сервер по протоколу RCON и панель администрирования.

Stack: Python, aiohttp, RCON, Webhooks

ТЕХНИЧЕСКИЕ НАВЫКИ

Backend: Python 3.11+, FastAPI, AsyncIO, Dishka (DI), Pydantic, SQLAlchemy 2.0, Alembic, REST API, WebSocket, gRPC, TypeScript, React / Next.js

Данные и очереди: PostgreSQL (индексация, блокировки, транзакции), Redis (кэширование, pub/sub), RabbitMQ, Celery, ARQ

Архитектура: Microservices, Event-Driven, Idempotency, Double-entry bookkeeping, Clean Architecture

Инфраструктура и AI: Docker, Docker Compose, Linux (bash, IPC pipes, systemd), RunPod, NVIDIA CUDA, Git, CI/CD, Prometheus, Grafana

Тестирование и линтинг: pytest, pytest-asyncio, Ruff, mypy

Web3 и Telegram: Web3.py, EVM (Base), TON, Tact, Aiogram 3, Telegram Mini Apps

ЯЗЫКИ И ОБРАЗОВАНИЕ

Языки: Английский — B1 / Intermediate (чтение техдокументации, переписка) | Украинский — Native | Русский — Fluent

Образование: Профильное самообразование и инженерная практика: Computer Science, распределенные системы, базы данных, асинхронное программирование (2023 — настоящее время)