

Илья Моис

Software Engineer · Удалённо или в офисе в Тбилиси, Грузия (GMT+4)

job.offers@mois.pro · github.com/mois-ilya · linkedin.com/in/moisilya · русский (родной), английский (профессиональный)

О СЕБЕ

Software engineer с восьмилетним опытом в продуктовых интерфейсах, API-контрактах, бэкенд-сервисах и инструментах для разработчиков. В Tonkeeper 21 месяц был единственным разработчиком консоли и самостоятельно поддерживал TypeScript SDK; также работал с публичной схемой TonAPI, интеграционной документацией и кошельком. Сейчас в одиночку строю два продукта.

ОПЫТ

Independent — Software Engineer

март 2026 — настоящее время · Тбилиси · два закрытых продукта, разработка в одиночку

- **Разделение расходов** — строю сервис на событийной модели: правки дописываются, а не заменяют счёт, поэтому параллельные изменения сохраняются, а баланс можно восстановить из истории. Property-based тесты сверяют взаиморасчёты с более простой эталонной реализацией.
- **Распознавание чеков** — строю сервис, который превращает фотографию чека в позиции для одновременного разбора несколькими людьми. Он целиком работает на Cloudflare Workers; этапы распознавания могут переключаться между поставщиками моделей, а результат отбрасывается, если арифметика не сходится с итогом чека.

Tonkeeper — Software Engineer, Developer Platform & DX

июнь 2024 — март 2026 · удалённо · TON Apps, команда из 23 человек, делающая Tonkeeper — кошелёк с 12M+ MAU.

Большая часть работы публична и доступна по ссылкам ниже.

- **Консоль разработчика** — единственный разработчик 21 месяц. Перевёл биллинг с TON на USDT без принудительной миграции существующих пользователей и **заменял MobX** на React Query, устранив дублирующиеся события и запросы к серверу.
- **Продуктовая работа в консоли** — выпустил управление и тарификацию вебхуков, тарифы API, историю платежей, конструктор аналитических запросов и инструменты для airdrop.
- **TypeScript SDK, ~16 тыс. загрузок в неделю** — самостоятельно поддерживал с середины 2024-го. Перестроил пакеты и генератор: значения int64 сохраняются как bigint, корректно обрабатываются особенности сериализации OpenAPI, а один клиент работает в браузере и Node с единственной рантайм-зависимостью.
- **Схема API и интеграционная документация** — исправлял места, неоднозначность которых проявлялась в сгенерированных клиентах: форматы целых чисел и дат, типы ответов и группировку методов. Написал руководства по отслеживанию транзакций, подписи данных и нормализации хеша.
- **Кошелёк** — выпустил сценарии стейкинга в неликвидных пулах, работу над свопами через SSE и исправления TON Connect для веба, десктопа и расширения; сделал отказы сторонних API восстанавливаемыми вместо падения приложения.

Quintegro — Frontend Developer

июль 2023 — май 2024 · удалённо

- Вёл разработку модуля изменения брони для платформы, связывающей хозяев и гостей.
- Определил API-контракты между фронтендом и бэкендом, генерировал из них типы TypeScript и настроил CI.

Minacu — Full Stack Engineer

апрель 2023 — июль 2023 · частичная занятость, удалённо

- Построил потоковое распознавание речи поверх gRPC, анализ текста через OpenAI API и интеграцию с Telegram через tdlb — Python и NestJS на бэкенде, Next.js на фронтенде.

Upper Echelon Products — JavaScript Developer, затем Technical Team Lead

апрель 2021 — март 2023 · удалённо

- После повышения до лида провёл 3–5 инженеров и 1–2 QA примерно через 11 релизов и был наставником пяти младших разработчиков.
- Построил пайплайн сбора данных из Amazon Seller Central и перевёз бэкенд с VPS на AWS с CI/CD. Привёл фронтенд-архитектуру к единому виду и перевёл документацию API на Swagger.

Apptech — Software Engineer

июль 2020 — ноябрь 2020 · частичная занятость, Санкт-Петербург

- Сделал пуш-уведомления и обмен сообщениями в реальном времени на Firebase для второй версии мессенджера.

GET Information Technology — Software Engineer

март 2018 — февраль 2020 · Санкт-Петербург

- Делал интерфейсы на Backbone/Marionette и ExtJS и сервисы на ColdFusion, отдававшие им данные.
- Внедрил ESLint и Babel вместе с инструментами, отслеживавшими их распространение по кодовой базе.

ИЗБРАННЫЙ OPEN SOURCE

- **Спецификация TON Connect** — уточнил два места, где две реализации, соответствующие спецификации, могли подписать разные сообщения для одного запроса: формат адреса в ответе на подпись данных и кодирование appDomain. Обе правки находятся в [спецификации протокола](#).
- **Подпись данных** — реализовал SignData в @tonconnect/sdk для текста, бинарных данных и TON-ячеек и опубликовал читаемую [эталонную реализацию на TypeScript](#).
- **Возможности кошельков** — добавил в SDK requiredFeatures и preferredFeatures: приложение объявляет свои требования, а несовместимые кошельки отсекаются до подключения.

НАВЫКИ

- **Языки и фронтенд** — TypeScript, JavaScript, Python; Go (читаю и делаю небольшие правки); React, Next.js, React Native
- **Бэкенд и API** — Node.js, Fastify, NestJS, PostgreSQL, OpenAPI, REST, gRPC, вебхуки, event sourcing
- **Облако и тестирование** — Cloudflare Workers, Durable Objects, AWS, Docker, CI/CD; Vitest, fast-check, Testcontainers
- **Предметные области** — платформы и SDK для разработчиков · платежи и биллинг · Telegram Mini Apps и боты · блокчейн

ОБРАЗОВАНИЕ

MSc, High Technology and Economics of Innovation

2019 — 2021 · Университет ИТМО

BSc, Information Systems and Technologies

2015 — 2019 · РГПУ им. Герцена