



РЕЗЮМЕ

Юрий Громыко Георгиевич

Full-stack разработчик · Автоматизация финансов и промышленности

Пишу программы. Люблю сложные задачи. Выпускаю то, что работает.

Открыт к новым проектам

9+ лет в своём деле

Алматы · удалённо / гибрид

Full-stack разработчик, ориентированный на результат, более 9 лет предпринимательского опыта в разработке. Специализация — автоматизация бухгалтерии и промышленности, банковские интеграции, ERP- и CRM-системы, продакшн-пайплайны на LLM. Одинаково уверенно работаю с архитектурой интерфейсов, бэкендом на Python и PHP, проектированием баз данных и администрированием серверов. Веду проект от спецификации до продакшена и поддерживаю его годами.

✉ jyri.gromyko2002@gmail.com

📍 Алматы, Казахстан

in linkedin.com/in/jyri-gromyko

🌐 cv.gromi.fi

9+

лет в разработке

80+

сданных проектов

6

лет на одной ERP-системе

25+

активных репозиториях

Профиль

Кто я и чем занимаюсь

Начал заниматься разработкой как предприниматель в 2017 году и с тех пор сдал более 80 проектов: интернет-магазины, ERP- и CRM-системы, банковские интеграции, пайплайны компьютерного зрения, маркетинговую автоматизацию. Большинство из них до сих пор работает.

Мой фокус — там, где бизнес-правила запутаны, а данные грязные: расчёт НДС для двух форм юрлица, автоматическая сверка банковских выписок с чеками, геометрия фасадных панелей на canvas, обогащение лидов из десятка разных API. Такие системы не выглядят эффектно — но именно на них держится компания.

Работаю с документированием: архитектурные решения фиксируются в ADR, у каждого модуля есть своя спецификация, тесты прогоняются до того, как что-то объявлено готовым. Я написал собственный контракт для AI-агентов (AGENTS.md), которому следуют все инструменты — чтобы работа оставалась воспроизводимой и проверяемой от сессии к сессии.

Опыт работы

Предприниматель и full-stack разработчик

2017 — настоящее время

Gromyko Jyri Tmi. / GG Solutions Oy · Хельсинки, Финляндия

Собственное ИП и общество с ограниченной ответственностью. Веду клиентские проекты целиком: постановка, архитектура, реализация, вывод в продакшн, поддержка и выставление счетов.

- Проектирование и разработка систем автоматизации бухгалтерии: учёт, расчёт НДС и пенсионных расходов, годовая отчётность, дебиторка, банковские интеграции (Nordea, Holvi).
- Full-stack приложения и SaaS-продукты: интерфейсы на React / Vue.js, бэкенды на Python, Django, Laravel и Flask, базы PostgreSQL и MySQL.
- Архитектура API и баз данных: REST API, многошаговые миграции, массовая обработка SQL и синхронизация с кассовыми и ERP-системами.
- Администрирование и деплой: Docker / Podman Compose, systemd + Nginx, Redis, Celery, общая инфраструктура разработки для нескольких проектов.
- Продакшн-системы на LLM: интеграции Claude и Gemini, анализ изображений, версионизируемый реестр промптов с прогонами eval, учёт стоимости каждого вызова.
- Координация команд: разработчики, дизайнеры, маркетологи; разбиение работы на параллельные пакеты и ревью результатов.
- Мультиканальные маркетинговые кампании (Google Ads, Meta, LinkedIn, SEO) и оптимизация конверсии с измеримым результатом.

Python Django Flask Laravel React PostgreSQL MySQL Docker Redis Celery REST API LLM

Военная служба

6 месяцев

Силы обороны Финляндии — Егерская бригада · Соданкюля, Северная Лапландия

Срочная служба в арктических условиях. Выносливость, работа в команде и дисциплина под нагрузкой.

Навыки и технологии

Что использую ежедневно, чем владею уверенно и что просто знакомо

БЭКЕНД

Python Django Flask FastAPI PHP Laravel
Node.js REST API SQLAlchemy Celery
WordPress / WooCommerce

ФРОНТЕНД

React Redux Toolkit TypeScript Vite
Konva / Canvas MUI Tailwind Vue.js i18next
jQuery HTML / CSS

БАЗЫ ДАННЫХ

PostgreSQL MySQL MariaDB SQLite Redis
ChromaDB SQL-massakäsittely

БАНКИ И ФИНАНСЫ

Nordea Holvi Firefly III ALV / VAT YEL
Tilinpäätös (KPL) Reskontra Sähköinen laskutus
Stripe Visma Nova

ИНФРАСТРУКТУРА И ДЕПЛОЙ

Docker / Compose Podman systemd + Nginx Linux
Git Google Cloud AWS CI/CD Playwright
Selenium

AI И ДАННЫЕ

Claude API Gemini API OpenAI API
openai-agents SDK LiteLLM Vision-analysys
Prompt-versiointi & evalit NLP RAG

ДАННЫЕ И ИНТЕГРАЦИИ

Gmail API Google Sheets API Google Calendar API
Telegram Bot API / Telethon XML / CSV / XLSX ETL
WooCommerce REST Web scraping Proxy-rotatio

МАРКЕТИНГ И РОСТ

SEO Google Ads Meta Ads LinkedIn Ads
Konversio-optimointi A/B-testaus GDPR-compliance

Ядро Уверенно Знаком

Ключевые проекты

Пять систем, где работа была самой глубокой и самой длинной



Editori — ERP для строительства

2021 — 2026

Полный жизненный цикл фасадного ремонта и реконструкции квартир в одной системе — изначально «редактор стройплощадки Westface»

Архитектор и ведущий разработчик

В продакшене

Шесть лет разработки ERP для подрядчика, занимающегося ремонтом фасадов и реконструкцией квартир. Система покрывает всю цепочку: заведение проекта, визуальное редактирование фасада и стен на интерактивном canvas, учёт элементов и их истории, настраиваемые статусные воркфлоу, тикетинг, снабжение материалами, персонал и техника, финансы. Три отдельных приложения работают через один REST API.

ЧТО Я СДЕЛАЛ

- Canvas-редактор фасадов и стен (Konva / react-konva): панели, швы, отверстия, изоляционные ленты, молдинги, опорные рамы, вертикальное профилирование — каждый параметр как пресет уровня проекта, который можно менять, не ломая старые проекты.
- Выгрузка прямо в производство: схема обработки на 99 колонок для линии ЧПУ (Cut Rite / Schmid B300) — координаты сверления X1–X10 и Y1–Y10, карманы, рамы, диагональныерезы, вентиляционные отверстия, обработка кромок, коды материалов (HPL, фиброцемент, алюминиевый композит) и порядок разбора по паллетам. Нарисованный в редакторе фасад превращается в программу для станка без промежуточных рук.
- Настраиваемая машина состояний: группы статусов, логика и типы статусов плюс история изменений для каждой сущности (элемент, квартира, комната, тикет, компания). Клиент описывает свои воркфлоу без изменений в коде.
- Учёт материалов и склада: номенклатура с категориями, остатки, движения, заявки на материалы с позициями, поставщики с историей цен, заказы.
- Тикетинг для бригад и техники: типы и группы тикетов, команды и оборудование, статусы по фазам проекта, зависимости между тикетами и полная история статусов.
- Права доступа на двух уровнях: spatie/laravel-permission на бэкенде и CASL на фронтенде, роли на уровне проекта — один и тот же пользователь может быть прорабом в одном проекте и наблюдателем в другом.
- Импорт и экспорт Excel (maatwebsite/excel, xlsx), генерация PDF в браузере (jspdf + html2canvas), мультиязычность (i18next) и уведомления в реальном времени.

- Воспроизводимая среда разработки: Docker Compose (MySQL 8, Redis 7, Nginx), запуск одной командой и health-check `php artisan app:selftest`, архитектурные решения зафиксированы в ADR.

МАСШТАБ

111 моделей Eloquent	171 миграции	137 контроллеров	248 маршрута API	900+ модуля фронтенда	3 приложения, 1 API				
Laravel 10	PHP 8.2	MySQL 8	Redis 7	React 18	TypeScript	Redux Toolkit	Konva	Refine.dev	MUI
Sanctum	Docker								



B2B CRM и автоматизация лидов

2026 — настоящее время

Поиск лидов, обогащение, кампании-последовательности и воронка продаж на одной платформе

Архитектор и единственный разработчик

В продакшене · v8.4

GDPR-совместимая платформа для исходящих продаж: лог писем, автоматизация кампаний по последовательностям, обогащение лидов и CRM на React. Два бэкенда живут параллельно — продакшн на Flask и переписывание на Django, они делят одну схему PostgreSQL и совпадают эндпоинт в эндпоинт.

ЧТО Я СДЕЛАЛ

- Движок кампаний: многошаговые последовательности, варианты сообщений с A/B-тестами (multi-armed bandit), ветвления по условиям, автоматические действия и зачисление получателей.
- Оркестрация 17 источников лидов под одним планировщиком: Finder.fi, PRH/YTJ, Jooble, LinkedIn Jobs и Messages, People Data Labs, Hunter, ZeroBounce, QuickEmail, Apify, Apilayer, Abstract, определение текста и скоринг PageSpeed.
- Скоринг PageSpeed и рекламных пробелов: Google PageSpeed Insights, определение SSL, Google Ads Transparency и Meta Ads Library сводятся в оценку 0–100 и приоритет hot / warm / cold, который обновляется автоматически при смене сайта компании.
- Цепочка обогащения, где каждый контакт проходит через платные и бесплатные источники (PDL → Hunter → ZeroBounce → Gemini), пока не получит проверенный доставляемый адрес и обоснованный контекст по компании.
- Воронка продаж: сделки, стадии и история стадий, задачи и шаблоны задач, лента активности, сохранённые фильтры и настройки колонок для каждого пользователя.
- GDPR-слой: чёрные списки на уровне проекта, модуль комплаенса, обработка отписок, события bounce и ротация OAuth для почтовых аккаунтов.
- Прокси-слой со sticky-сессиями: один выходной IP закрепляется за блоком запросов через ContextVar, а оборванная сессия повторяется с новым идентификатором. Замер: из 12 последовательных запросов 11 успешных на 11 разных финских IP, медиана 1,6 с.
- Продакшн без контейнеров: systemd + Nginx + нативный PostgreSQL. Разработка — на Podman Compose с локальными оверрайдами.

МАСШТАБ

39+ доменных модели	17 источников лидов	17 Django-приложений	v8.4 выпущенная версия	78 коммитов за 6 мес.					
Python	Flask 3.1	Django 6.0	SQLAlchemy 2.0	PostgreSQL 16	React 19	Vite 8	Celery	Redis	Telegram API
Gemini	systemd + Nginx								



Автоматизированная бухгалтерия

2026 — настоящее время

От банковской выписки до годовой отчётности для двух форм юрлица — без участия человека

Архитектор и разработчик

В ежедневном использовании

Сквозная автоматизация бухучёта для ИП, ООО и личного счёта. Система забирает выписки, достаёт чеки из почты, сверяет их с транзакциями, считает НДС и пенсионные расходы, публикует проводки в Firefly III и формирует готовую годовую отчётность.

ЧТО Я СДЕЛАЛ

- Загрузка и разбор выписок: Nordea (CSV / JSON) и Holvi (XLSX / JSON). У ООО два счёта Holvi — основной и налоговый резерв — в учёт публикуется только основной, а резерв отслеживается как бюджет.
- Автоматический сбор чеков через Gmail API (OAuth2), конвертация писем в PDF и сопоставление чека с нужной транзакцией по сумме, дате и контрагенту.
- Движок НДС: ставки и налоговые классы, обратный НДС, расходы по пенсионной схеме, карта контрагентов и правила, которые живут в конфигурации, а не в коде.
- Идемпотентная публикация в Firefly III: транзакции идентифицируются по external_id, поэтому повторный запуск никогда не создаёт дублей. Финский план счетов маппится в категории Firefly, бюджеты читаются из плана в Google Sheets, PDF-чеки прикрепляются к транзакциям.
- Генератор годовой отчётности: баланс и отчёт о прибылях и убытках по финскому закону о бухучёте для обеих форм юрлица, на выходе XLSX и PDF. Плюс готовая к подаче на Vero.fi HTML-декларация по НДС.
- Модуль планирования: план против факта по трём счетам, исходный план берётся из Google Sheets, отчёт публикуется обратно во вторую таблицу.
- Более 20 скриптов запуска (.sh и .bat), прогоняющих весь пайплайн одной командой; Chrome и шрифты определяются автоматически в Linux, macOS и Windows.

МАСШТАБ

3	2	20+	0
учётных контура (ОУ / ТМЛ / личный)	банка интегрировано	скрипта пайплайна	дублей (идемпотентно)
Python	Gmail API	Google Sheets API	Nordea
Holvi	Firefly III	SQLite	Gemini
openpyxl	Chrome headless		
Docker			



SKT: софт для расчёта смет с компьютерным зрением

2026 — настоящее время

Оценка зданий и расчёт стоимости мойки окон в столичном регионе Хельсинки

Архитектор и разработчик · клиент Suomen Kiipeilytekniikka Oy

Активный клиентский проект

Заменяет расчёт смет клиента на базе Visma Nova веб-приложением. Пайплайн собирает объявления Oikotie, геокодирует здания, получает панорамные снимки, прогоняет по ним Claude Vision, считает человеко-часы и формирует коммерческое предложение. Расширяется до поиска жилищных товариществ по почтовым индексам и выгрузки обратно в Nova.

ЧТО Я СДЕЛАЛ

- Пайплайн из 10 фаз: скрейпер Oikotie (карточки, детали и фото) → геокодирование Digitransit → Google Street View (с подписью URL) и Mapillary → Claude Sonnet Vision → формула человеко-часов v2.0 с уточнением v3.0 → предложение одной командой.

- Извлечение данных управляющих компаний из свободного текста через LLM, импорт списка из 584 управляющих контор и сопоставление с существующей базой предложений.
- Учёт стоимости каждого вызова LLM с собственным self-test — цена анализа изображений видна до того, как прогон масштабируется на сотни объектов.
- Streamlit-дашборд на 5 страниц по результатам, база из 11 таблиц и импортёры для Excel-выгрузок клиента.
- Работа разбита на шесть параллельных этапов (S0–S5), которые можно вести в девяти одновременных сессиях — у каждого свой файл, владелец и критерии приёмки.

МАСШТАБ

10	264	1,7 GB	584	84
фаз пайплайна	объекта в пилотной зоне	изображений для анализа	управляющие конторы	коммита
Python 3.11	Claude Sonnet Vision	Google Street View	Mapillary	Digitransit
SQLite	Streamlit	Docker		
Visma Nova				



Personal Advisory Council

2026

Self-hosted совет из нескольких ИИ, оценивающий решения параллельно

- Архитектор и разработчик
- Рабочий прототип

Платформа загружает банковские выписки (XLSX), заметки Obsidian и README из GitHub-репо, параллельно запускает восемь специализированных агентов-персон по вопросу пользователя, проводит перекрёстное рецензирование каждого ответа и синтезирует вердикт председателя с оценкой антихрупкости (0–100) и списком трёх действий.

ЧТО Я СДЕЛАЛ

- Пятиэтапный воркфлоу: загрузка → агент-архитектор предлагает состав совета → интерактивное редактирование → параллельный прогон → рецензирование и синтез.
- Мультипользовательский режим с изоляцией сессий, роутинг по URL, ссылки на сессии для шаринга и кэш, который не даёт случайно перезапустить дорогие операции.
- Учёт токенов и стоимости в реальном времени, настраиваемая глубина рассуждения и отслеживание прогресса во время прогона.
- Стек на Docker Compose: бэкэнд FastAPI, фронтенд React, прокси LiteLLM, ChromaDB для векторного поиска и опциональный Telegram-бот.

МАСШТАБ

8	5	0–100
агентов-персон параллельно	этапов воркфлоу	оценка антихрупкости
Python 3.12	FastAPI	SQLAlchemy 2.0
openai-agents SDK	LiteLLM	ChromaDB
React 19	Tailwind 4	
Docker Compose	Telegram	



Торговая автоматизация и рыночные данные

2022 — 2023

Копитрейдинг-боты, статистика стратегий и сентимент-анализ новостей на трёх рынках

- Архитектор и разработчик
- Сдано

Двухлетняя работа с целью измерить, какая часть результатов успешных инвесторов вообще воспроизводима. Публичные профили топ-трейдеров читаются программно, их открытые позиции зеркалируются на демо-счёт, каждая сделка логируется для анализа. Вокруг этого выросли сбор новостей, сентимент-анализ заголовков и сравнение стратегий.

ЧТО Я СДЕЛАЛ

- Копитрейдинг-бот для трёх рынков: форекс (cTrader / IC Markets), акции и криптовалюта (Binance). Профили опрашиваются каждые десять секунд; новая открытая позиция зеркалируется и закрывается вместе с оригинальной.
- Парсер трейдеров и стратегий из публичных профилей результативности (FXBlue): выборка пользователей, сбор истории каждой стратегии и статистическое сравнение — какие результаты выдерживают проверку, а какие шум.
- Новостной конвейер с сентимент-анализом: сбор зарубежных и крипто-новостей (в том числе Cointelegraph), классификация заголовков и картирование источников медиа-индексов — сигнал отделялся от шума до того, как на нём что-то строилось.
- Моделирование и тестирование стратегий: выгрузка истории сделок и PnL-анализ, поиск закономерностей, демо-сделки в лог и сравнение результатов по периодам.
- Развёртывание на выделенном сервере в бесконечном цикле, чтобы наблюдение шло круглые сутки без присмотра; каждая сделка сохраняется с датой, направлением, инструментом и объёмом.

МАСШТАБ

3 рынка охвачено	10 s интервал опроса	24/7 работа без присмотра	2 года разработки
Python	cTrader	IC Markets	Binance API
FXBlue	Sentiment analysis	NLP	SQLite / CSV
VPS	cron		

Клиентские проекты и e-commerce

Коммерческие заказы, большинство до сих пор в продакшене



camu.fi

2020 — 2023

Интернет-магазин на 8 000+ товаров с постоянной синхронизацией с кассой

Разработчик и сопровождение

- Синхронизация товаров и остатков с кассовой системой по XML каждые пять минут — миллионы SQL-команд в сутки, при этом витрина не тормозит.
- Конверсия поднята с 0,2 % до 1 %: SEO, скорость страниц, объектный кэш Redis, оптимизация изображений и переработанная воронка покупки.
- 14 отдельных фоновых сервисов: синхронизация товаров, выгрузка заказов, обновление остатков, починка и оптимизация изображений, SEO-скрипты, фид Shopping, чистка базы и прогрев кэша.

WordPress

WooCommerce

PHP

MySQL

Redis

Python

XML

SEO



westface.fi

2022 — 2026

Магазин фасадных панелей с интеграцией в ERP Visma Nova / Loocos

Разработчик и владелец интеграции

- Два Python-микросервиса связывают магазин с ERP: импортёр распаковывает архив Nova и обновляет каталог через WooCommerce REST API (остатки, цена, статус, категории, бренд, фасадные атрибуты); экспорт заказов работает в обратную сторону.
- Мультиязычность на WPML, SEO на Rank Math и хранилище заказов WooCommerce HPOS. JSON-LD товаров приведён к принципу единственного источника, чтобы поисковик не видел две конкурирующие сущности Product.
- Полная локальная Docker-копия боевого магазина, чтобы тестировать изменения не трогая продажи — включая сброс кэша и перезапись URL.

WordPress

WooCommerce

PHP 8.2

MySQL

Python

WPML

Rank Math

Visma Nova

Docker



whitebalance.fi

2022 — 2026

Магазин снаряжения для скалолазания и активного отдыха (Kiipeilytekniikka)

Разработчик

- Та же пара интеграций Nova / Loocos, что и на westface, адаптированная под другую модель атрибутов (бренд, цвет, размер) и другой режим хранения заказов.
- Слой перевода в рантайме на TranslatePress — товары хранятся только на финском, поэтому дубликаты записей на каждый язык не создаются.
- Локальный стек разработки со своими заглушками (REST-аутентификация без HTTPS), явно помеченными как локальные, чтобы они никогда не попали в продакшн.

WordPress

WooCommerce 8.4

PHP 8.2

MySQL

Python

ACF

TranslatePress

Rank Math

Docker



tilasa.fi

2018

Сайт и магазин итальянской мебели-трансформера

Дизайн и разработка

- Откидные и настенные кровати как товарные категории, скачивание каталогов и заявка на рассрочку прямо с сайта.
- Сделано на годы: сайт работает без перерывов с 2018 года и в 2026-м всё ещё жив без переделки.

WordPress

WooCommerce

PHP

SEO



Финская национальная опера — кампания

2019

Таргетированная digital-кампания и воронка продажи билетов

Планирование кампании и оптимизация

- Сегментация аудитории и отдельные сообщения по каналам в LinkedIn Ads и Google Ads, оптимизация конверсии на каждом шаге воронки продажи билетов.

LinkedIn Ads

Google Ads

Analytics

CRO



Palsbo — управление помещениями и защита форм

2021 — 2026

Импорт помещений из Excel, группы помещений и защита от спама

Разработчик

- Регулярные импорты списков помещений из Excel чистятся и нормализуются; модель данных групп помещений была спроектирована отдельной спецификацией до реализации.
- Защита форм обратной связи через Cloudflare Turnstile, план и спецификация написаны до кода.

PHP

WordPress

Python

Cloudflare Turnstile

XLSX



accounta.fi

2021 — 2023

SEO в бухгалтерской отрасли — одной из самых конкурентных ниш Финляндии

SEO-стратегия и реализация

- Кластерное дерево примерно из 400 запросов по всей отрасли: бухучёт, бухгалтерские фирмы, расчёт зарплат, годовая отчётность, НДС и регистрация компании — каждая ветка со своим назначением страницы, от услуг до блога.
- Анализ примерно 50 доменов отрасли (в том числе Accountor, Azets, Rantalainen, Talenom, Netvisor, Visma, Zervant): органический трафик, ссылающиеся домены и глубина ссылочного профиля — на что имеет смысл нападать, а на что нет.
- Результаты отслеживались по данным Search Console, а не по оценкам — динамика позиций и кликов фиксировалась, и список запросов пересматривался по ней.

SEO

Search Console

Keyword research

Backlink analysis

Content strategy



Betonikoneet.fi и Liikuntasaumat.fi

2019 — 2021

Видимость в поиске и Google Ads для строительной техники и швов

SEO и Google Ads

- Анализ запросов и структура кампаний для обеих ниш, исправление технических SEO-ошибок и расширение каталога до сотен позиций.
- Разбор ссылочных профилей конкурентов в нише деформационных швов: откуда в отрасли реально берутся ссылки и какие из них воспроизводимы.

SEO

Google Ads

WordPress

Analytics



Лид-машина — из ручной работы в платформу за шесть лет

2019 — 2026

Одна и та же задача, решённая заново столько раз, что стала продуктом

Разработчик

- 2019–2020: первые парсеры и черновик лид-CRM, анализ целевых групп, скрипты звонков и таблицы учёта лидов; парсер событий Хельсинки и разбор ниш по скандинавским реестрам компаний (Proff.no/dk/se).
- 2021–2022: парсеры Fonecta и Finder и полная база компаний строительной отрасли — более 400 компаний с названиями, доменами, руководителями, диапазонами оборота и бизнес-ID. Плюс автоматическая рассылка по вакансиям.
- 2026: та же потребность, но продуктом — 17 источников лидов, цепочка обогащения, кампании-последовательности и воронка продаж в одной платформе. Шесть лет ручной работы сложились в систему.

Python

Finder.fi

Fonecta

Proff

PRH / YTJ

Web scraping

CRM



kiipeilytekniikka.fi

2025 — 2026

Сайт и портфолио компании промышленного альпинизма

Поддержка и локальная среда разработки

- Локальный Docker-клон боевого сайта с импортом базы, чтобы правки делались безопасно до релиза.

WordPress

PHP

MySQL

Docker

+ более 74 других сданных проектов за эти годы

Инструменты и инфраструктура

То, что я построил, чтобы всё остальное стало возможным



AutoCrawler — самоуправляемый QA-агент

Определяет фреймворк веб-приложения, статически извлекает маршруты, формы и селекторы, поднимает приложение через Playwright, обходит его с заполнением форм через LLM и детекцией аномалий по скриншотам, отдаёт отчёты в SQLite и HTML. Адаптеры под восемь фреймворков (Django, Flask, FastAPI, Laravel, React, Vue, Next.js, Nuxt), восемь тестовых приложений и интеграционные тесты, требующие не менее 80 % покрытия маршрутов.

Python 3.11

Playwright

SQLite

LLM vision

pytest



gromi-core — общая Python-библиотека

Мультипровайдерный LLM-клиент (Claude, Gemini) с учётом стоимости, принудительным JSON и логикой ретраев/фолбэков; структурированное JSON-логирование; слоистая конфигурация (env → .env → ruproject); декораторы tenacity, настроенные под LLM API. Публикуется как приватный пакет, который другие проекты ставят зависимостью.

Python

Claude API

Gemini API

tenacity

GitHub Packages



gromi-prompts — версионизируемый реестр промптов

Промпты — версионизируемые артефакты, а не строки, закопанные в код: у каждого YAML-мета, история версий и минимум десять тест-кейсов в JSONL. CLI прогоняет evals, сравнивает версии между собой и делает сухой рендер без вызова LLM. В реестре 19 семейств промптов — классификация расходов, распознавание чеков, анализ фасадов, персонализация писем, SEO-контент, артикулы товаров.

[Python](#)[JSONL evals](#)[YAML](#)[CLI](#)[GitHub Packages](#)

myaitools — самораспухающая библиотека инструментов

Реализует цикл LATM («LLMs as Tool Makers»): распознать → специфицировать → собрать → протестировать → задокументировать → зарегистрировать → использовать. В начале сессии каталог инструментов подставляется в контекст, в конце задачи дешёвая локальная эвристика бесплатно решает, стоило ли захватить подзадачу как инструмент — и если да, отдельный агент-строитель пишет его с тестами и регистрирует. За рассуждение платишь один раз, инструмент остаётся.

[Python](#)[Bash](#)[Claude Code hooks](#)[JSON manifest](#)[pytest](#)

proxymaker — координатор ротации

Единственный процесс, который обращается к endpoint'ам ротации провайдеров.

Провайдеры лимитируют ротацию (один раз в две минуты), поэтому параллельные вызовы получают 429 и выходной IP замерзает — а ротация одного сервиса рвёт запрос другого. Брокер сериализует, выдерживает кулдаун и координирует ротацию по кворуму: IP меняется только когда все подключённые сервисы проголосовали, или когда срабатывает предохранитель. Зависший сервис никогда не блокирует остальных.

[Python](#)[HTTP daemon](#)[flock](#)[cron](#)[curl_cffi](#)

Общая инфраструктура разработки

Один Docker-стек (MySQL 8, Redis 7, Mailpit) обслуживает все проекты в общей сети: базы создаются скриптом, каждый проект подключается к внешней сети, а его лаунчер сам поднимает инфраструктуру, если она не запущена. Семь проектов делят один стек без конфликтов портов.

[Docker Compose](#)[MySQL 8](#)[Redis 7](#)[Mailpit](#)[Bash](#)



Финансовый стек Firefly III

Локальная установка Firefly III с Data Importer, AI-категоризатором и почтовым репортером. Запуск одной командой делает импорт базы, поднятие контейнеров и health-check — именно сюда бухгалтерская автоматизация публикует транзакции и чеки.

Docker Compose

Firefly III

MySQL

Mailpit



Контракт для AI-агентов и обучающий пайплайн

Единый обязывающий контракт работы (AGENTS.md), зеркалируемый во все AI-инструменты разработки — Claude Code, Copilot, OpenCode, Cursor, Windsurf, Codex, Aider — чтобы работа оставалась воспроизводимой независимо от инструмента. Поверх — трёхслойное хранилище знаний в Obsidian (модули, подписки, спецификации) и архитектурные решения в формате ADR. В том же репозитории обучающий пайплайн: саммари, флешкарты и NLP-карты.

Markdown

Obsidian

Node.js

Python

ADR



autoreger — оркестрация квот API

Оркестрирует регистрации и квоты по пяти сервисам обогащения (People Data Labs, Apilayer, Apify, Hunter, ZeroBounce) в двух режимах: burst выбирает один сервис до конца, pulse крутит по одному ключу на сервис ради более ровного профиля нагрузки. Два корневых оркестратора и пять проектных бутстрапов, всё с логами.

Python

Bash

Selenium

Proxy rotation

cron



Автоматизация подключения к MailFlow

Полуавтоматический инструмент для подключения почтовых ящиков к сервису прогрева. Один видимый Chrome с постоянным профилем через авторизованный прокси: логинишься один раз, дальше инструмент проходит ящики по одному, проводит мастер app-пароля Gmail и предзаполняет форму. IP прокси ротируется раз за сессию с учётом кулдауна.

Python

Selenium

Chrome profile

Proxy

Личные проекты и эксперименты

Личные упражнения, на которых я осваивал новые техники



krisha_hunter

Пайплайн поиска квартиры в Алматы: обход выдачи → текстовый фильтр с алиасами и распознаванием отрицаний → Gemini vision по фото → база знаний по ЖК и районам → скоринг 0–100 → Telegram. Геоданные из 2GIS и Google Routes. Около \$0,001–0,003 на объявление.

Python

Gemini 2.5 Flash

2GIS API

Google Routes

SQLite

Telegram



pupu_bot

Браузерный бот, который действительно чинит детект headless, а не обходит его: постоянный профиль, разовый ручной логин, экспорт и восстановление сессии каждые 30 секунд и команда диагностики с отчётом по версии Chrome, вьюпорту, утечке User-Agent и геолокации. Набор тестов на jsdom без браузера покрывает 34 проверки.

Python

Selenium

Chrome

jsdom

Bash

bibi_bot

Автоматизация Telegram на Telethon для бота с reply-клавиатурой. Исчерпание квоты распознаётся, и работа сама продолжается после паузы; команда preflight проверяет конфиг, сессию и разбор карточки до запуска. Юнит-тесты покрывают всю логику решений.

Python

Telethon

unittest

iOS-приложение на WebView и релиз в App Store

Упаковка веб-сервиса в нативное iOS-приложение вокруг WebView и весь путь релиза в App Store: сертификаты, provisioning, требования ревью и та часть работы, о которой никто не догадывается до первого отказа.

iOS

WebView

App Store Connect



Автоматизация социальных платформ

Серия небольших браузерных и API-автоматизаций для разных платформ (в том числе Instagram и Tinder) — упражнения в том, как держать сессию живой, как распознавать исчерпание квоты и как заставить автоматизацию ломаться громко, а не тихо, когда меняется интерфейс.

Python

Selenium

REST APIs



crimsonalter

Парсер форумов на XenForo и менеджер загрузок из облаков: поиск тем по ключевым словам, оценка объёма до скачивания, загрузки из Mail.ru Cloud и Яндекс.Диска, всё раскладывается по темам. Два разных движка форума, один за DDoS-Guard и капчей — под него понадобился отдельный браузерный bootstrap.

Python

curl_cffi

Playwright

XenForo

Как я работаю

Принципы, которые видны в каждом проекте выше



Архитектурные решения фиксируются

Каждое неочевидное решение попадает в ADR: что решили, какие варианты взвешивали и почему. Через полгода никому не придётся гадать.



Доказательства раньше утверждений

Ничто не объявляется готовым, пока тесты не прогнаны и вывод не прочитан. Health-check команды — часть каждого стека, а не довесок.



Идемпотентность по умолчанию

Повторный прогон бухгалтерской задачи, синхронизации товаров или импорта лидов никогда не должен плодить дубли. Внешние идентификаторы и dry-run — правило, а не исключение.



ИИ как инструмент, а не срезанный угол

Вызовы LLM версионизируются, тестируются и учитываются по стоимости. Промпты живут в реестре с eval-кейсами — как любой другой продакшн-код.



Воспроизводимая среда

Каждый проект стартует одной командой на чистой машине. Общая инфраструктура убирает конфликты портов и разговоры «у меня работает».



Работа режется на параллельные куски

Крупные работы делятся на независимые пакеты со своим файлом, владельцем и критериями приёмки — чтобы несколько разработчиков (или агентов) двигались одновременно и не сталкивались.

Образование

Бакалавр наук — AI in Business

[IU International University of Applied Sciences](#)

2024 — настоящее время · Дистанционно
Искусственный интеллект в бизнесе: аналитика данных, машинное обучение, цифровые бизнес-модели и управление проектами.

9+ лет продакшн-кода

Самообразование и практика

2017 — настоящее время
Большая часть навыков получена на клиентских системах, которые должны были заработать и потом работать годами.

Языки

Финский		Родной
Английский		Свободный
Русский		Продвинутый
Шведский		Начальный

Рекомендации

Юрий сделал для Tilasa сайт [tilasa.fi](#) ещё в 2018 году. В 2026-м он всё ещё работает. Рекомендую Юрия за экспертизу и качество работы.

Jussi Jokinen
Tilasa Oy

Делает то, что обещал — в срок и аккуратно. Спасибо!

Miika Junttila
Peer Car Share Oy

Увлечения

 Боевое самбо (самозащита без оружия)

 Бокс

 Кикбоксинг

 Разработка собственных инструментов

О себе

Я в своей стихии, когда задача сложная, и люблю распутывать запутанное. Автоматизация банковской интеграции, архитектура масштабируемого SaaS, оптимизация массовой обработки — я вкладываю в это одну и ту же энергию. Мне нравится делать скучный софт, потому что именно на скучном софте общество работает без сбоев.

Больше всего мотивирует момент, когда запутанное целое превращается в понятную и надёжную систему. Хороший софт не просто технически работает — он понятен, поддерживаем и живуч в долгую. Поэтому я вкладываюсь в чистую архитектуру, тестируемость и автоматизацию.

Давайте поговорим

Напишите, что нужно построить — отвечаю в тот же день.

✉ jyri.gromyko2002@gmail.com

in linkedin.com/in/jyri-gromyko

🌐 cv.gromi.fi

📍 Алматы, Казахстан

Это резюме генерируется из одного источника на трёх языках. · cv.gromi.fi