

Персональная
Информация

Гришин Антон
Москва, Россия
[alchemmist](#)
in [anton-grishin](#)
✉ [@alchemmist](#)
☎ +7(915)067-2638
✉ anton.ingrish@gmail.com



Обо мне

Я студент, начинающий разработчик. Программирую 4 года, за это время участвовал в разработке 44 репозиторий, отправил 921 коммит и написал 176729 строчек кода. В прошлом профессиональный [волейболист](#). Также я [преподаватель](#) Python, за все время обучил > 30 учеников. Сооснователь и разработчик в [Баллките](#).

Навыки

Python: `FastAPI`, `pydantic`, `Flask`, `SQLAlchemy`, `faststream`, `alembic`, `pytest`.
Go: `http`, `grpc`, `protobuf`, `tgbotapi`, `reflect`, `gofsm`.
Databases: `postgres`, `sqlite`, `redis`, `Yandex Object Storage`.
Message brokers: `RabbitMQ`, `Mosquitto`.
Other technologies: `SQL`, `Java`, `JavaScript`, `bash`.
Dev tools: `Docker`, `Podman`, `Make`, `CI/CD`, `Linux`, `Git`

Образование

[Центральный Университет](#) — Математика и компьютерные науки, 2028

Сертификаты

Яндекс Лицей. Экстерном поступил на 2-й курс [Лицея Академии Яндекса](#) по «Промышленной Разработке». Решил 236 задач, реализовал и защитил 3 проекта, финальный на 100/100 баллов. Окончил с [аттестатом с отличием](#).

Сентябрь 2022 — Апрель 2023

Курс Python. Прошёл [курс](#) по `Python` на Stepik. Выполнил 71 задание и завершил его с [отличием](#).

Сентябрь 2021 — Апрель 2022

Достижения

Прошёл в **финал (1) Чемпионата России** по спортивному программированию, среди 5000 участников. Разработал [микросервисную архитектуру](#) из 9 сервисов для [веб-приложения](#), агрегирующего спортивные события по всей России. И разработал [алгоритм](#) обработки и проверки ежегодных государственных отчётов о спортивных мероприятиях. Технический стек: `Kafka`, `React`, `RabbitMQ (FastStream)`, `FastAPI`, `OAuth`.

Декабрь 2024

Победитель научно-практической конференции «Наука для жизни» среди 117 проектов с [проектом умного дома](#) для частных и государственных образовательных учреждений. Технический стек: `Redis`, `Zigbee2MQTT`, `websockets`, `Go`, `Python`, `Flask`, `React`.

Апрель 2024

Участник хакатона Nuclear IT Hack, где моя команда 48 часов работала над кейсом Росатома: AI сервис для определения эмоционального тона онлайн-встреч.

Апрель 2024

Призёр 3-й степени, олимпиады МПГУ по информатике: «Прикладная информатика»

Июнь 2023

Интересы

Формальная верификация: Прохожу курс [softwarefoundations](#). Прочитал первый том, [доказал](#) 250 теорем на `Coq`.

Linux: Использую Arch с композитором Hyprland. Опубликовал свои [.dotfiles](#). С нуля написал свой Neovim [конфиг](#) и цветовую [тему](#). Написал более 20 кастомных [скриптов](#).

Разработал бота в Telegram по модели SMB (Single Message Bot) на Go для коробочной системы лояльности ballkit.ru.

Технологии: Go, tgbotapi, grpc, gofsm.

Разработал прототипы дизайна админ панели и интерфейса планшета управления (т.н. "хаба"). Сверстал их (1, 2) на React. Написал бота для администрирования системы. Реализовал взаимодействие с 5-ю умными устройствами по протоколу Zigbee, через брокер сообщений Mosquitto, и 3 сценария в кабинете. Технологии: Python, Go, JavaScript, bash, Flask, sqlite, redis, postgres, JsSip.

Разработал механику онлайн дуэлей с реальными игроками в Telegram боте. А так же удобный движок рендеринга сообщений в чате через jinja2 шаблоны, так как для текстовой игры требовалось гибко работать с большим количеством сценарных и пейзажных текстов. Технологии: Python, Flask, alembic, python-telegram-bot, SQLAlchemy, jinja2.

В команде разработали полнофункциональный модульный игровой движок с архитектурой ECS. А также survival top-down шутер игру о зомби-апокалипсисе. Технологии: Python, Pygame, esper, pytmx, aseprite, Tiled.

Разработал десктоп-приложение на PyQt6. Настроил pre-commit для автоматической конвертации .ui файлов в .py, а также actions для автоматической сборки нового релиза под все операционные системы при каждом коммите в main. Недавно перевёл проект на uv для удобного управления зависимостями. Технологии: Python, PyQt6, threading, schedule.

- Разработал бота для администрирования и продажи WireGuard ключей по подписке. Написал наглядную аналитику по клиентам прямо в боте. Технологии: Python, sqlite, python-telegram-bot.
- Бот для агрегирования релевантных сообщений из чатов, каналов и групп по ключевым словам. Технологии: Python, theleton, redis

- Парсер португальского посольства с механизмами авторизации (1), разрешения капчи (2), и ветвящимся обходом форм на сайте (3). Написал Docker образ для контейнеризованного парсинга selenium из под движка Chromium. Настроил pre-commit хуки для линтера ruff и статического анализатора pyright. Технологии: Python, selenium, pycorg, SQLAlchemy, FastAPI.
- Cli-инструмент для безопасного проведения операций на Outline серверах на тысячах клиентов. Технологии: Python.

Спроектировал базу данных (1). Написал бэкенд с ключевой бизнес-логикой на Spring и сервис авторизации на Go, а так же фронтенд на React.

Технологии: Java, SpringBoot, Lombok, JavaScript, React, Go, crypto, pgx.

→ больше проектов на [GitHub](https://github.com)