



Походяев Константин

Мужчина, 21 год, родился 3 февраля 2004

+7 (913) 6002673 — предпочитаемый способ связи • Telegram: kspohodyaev

konstantinpohodyaev@yandex.ru

Другой сайт: <https://github.com/KonstantinPohodyaev>

Проживает: Москва, м. Окская

Гражданство: Россия, есть разрешение на работу: Россия

Не готов к переезду, не готов к командировкам

Желаемая должность и зарплата

Python-разработчик

Специализации:

— Программист, разработчик

Занятость: полная занятость, частичная занятость, проектная работа, стажировка

График работы: сменный график, гибкий график, удаленная работа

Желательное время в пути до работы: не имеет значения

Опыт работы — 3 года 1 месяц

Февраль 2024 —
настоящее время
1 год 7 месяцев

Проектная деятельность

Москва

Python-разработчик

Опыт работы с Python и его библиотеками, включая фреймворки Django, DRF, Flask и FastAPI. Работал с реляционными СУБД (SQLite3, PostgreSQL) и различными API (Telegram Bot API, Яндекс.Практикум, OpenWeatherAPI, GoogleAPI, WildberriesAPI и другие). Проекты разворачиваю в Docker с использованием Nginx и CI/CD.

Мои проекты доступны на GitHub:

<https://github.com/KonstantinPohodyaev>

Ключевые проекты:

1. PriceWatcher (API + Telegram-бот) - Telegram-бот для отслеживания цен на маркетплейсах (Wildberries, OZON и др.). Пользователи могут подписываться на товары, и при снижении цены бот пришлёт уведомление! Так же есть возможность отслеживания истории товаров, удобная кастомная авторизация авторизация.

Реализовано:

- Реализация кастомной авторизации и аутентификации с использованием JWT-токенов (шифрование Fernet, хранение в БД).
- Интеграция fastapi-users для пользовательской логики авторизации.
- Реализация CRUD-операций через SQLAlchemy 2.0 и Pydantic.
- Создание удобного и интуитивного интерфейса бота без накопления сообщений.
- Реализация авторизации и регистрации через Telegram с использованием JWT.
- Настройка уведомлений о снижении цен (интеграция с API Wildberries/Ozon).
- Регулярная проверка цен и фоновое обновление данных о товаре через APSchedule.
- Развёртывание всех компонентов с использованием Docker и Docker Compose.
- Конфигурация Nginx в качестве реверс-прокси.
- Организация работы со статическими файлами (аватары).

Стек: FastAPI, SQLAlchemy 2.0, PostgreSQL, pydantic, alembic, FastAPI-Users, APScheduler,

python-telegram-bot, docker, nginx, uvicorn, python-dotenv, aiofiles, aiohttp

2. Telegram-бот для управления продажами фейерверков для компании Joker Fireworks. Разрабатывался в формате хакатона в команде из 7-ми человек.

Реализовал следующие функции:

- CRUD операции с моделями товаров, категорий, тегов, медиа, написание соответствующих pydantic-схем.
- Реализация эндпоинтов для работы с товарами
- Фильтрация товаров по параметрам
- Настройка скачивания медиа-файлов с Яндекс облака
- Сборка и настройка всей логики кнопок проекта

Стек: python, FastAPI, python-telegram-bot, postgresql, uvicorn, sqlalchemy, pydantic, aiohttp

3. Проект, на котором можно подобрать рецепт блюда на любой вкус. Реализовал REST API на DRF, расширил функционал административной части Django, разместил проект на удаленном сервере в Docker-контейнерах.

Стек: python, django, DRF, djoser, Pillow, docker, nginx, gunicorn

4. Проект, который собирает отзывы пользователей на произведения. Сами произведения не хранятся, здесь нельзя посмотреть фильм или послушать музыку. Разрабатывался в команде из 3-х человек.

Реализовал следующие функции:

- Описал модели Title, Genre и Category
- Написал загрузчик данных в БД
- Описал вьюсеты и сериализаторы для соответствующих моделей

Стек: python, django, DRF, djoser, django-filters, requests, django-rest-framework-simplejwt

5. Проект, который собирает пожертвования на различные целевые проекты: на медицинское обслуживание нуждающихся хвостатых, на обустройство кошачьей колонии в подвале, на корм оставшимся без попечения кошкам — на любые цели, связанные с поддержкой кошачьей популяции. С возможностью выгрузки данных в Google-таблицы

Стек: python, fastapi, fastapi-users, fastapi-users-db-sqlalchemy, SQLAlchemy, requests, pydantic, alembic, google-api-python-client

6. Парсер, который может собрать нужную информацию с сайта документации Python или документов PEP. Парсер реализован на базе фреймворка Scrapy.

Паук рер парсит все документы PEP: сохраняет номер, название и статус каждого в csv-файл с префиксом per_. Так же подсчитывает количество каждого статуса и общее количество всех статусов и сохраняет это в csv-файл с префиксом status_summary.

Стек: python, requests, scrapy, lxml

7. Проект - укоротитель ссылок, который может заменять длинные и некрасивые оригинальные ссылки на красивые и лаконичные! Это одностраничное приложение с API. На главной странице выведена форма для добавления пары (original; short).

Стек: python, Flask, Flask-SQLAlchemy, Flask-WTF, Jinja2, SQLAlchemy, WTForms

8. weather_app - простое и элегантное приложение для получения информации о погоде, созданное с использованием Flask. Weather App получает актуальные данные о погоде через API OpenWeatherMap. Приложение также включает кастомные страницы ошибок для улучшения пользовательского интерфейса. Есть возможность хранить в БД устаревшие данные о погоде в городах, максимум 5 записей, при добавлении шестой, самая старая автоматически удаляется.

Стек: python, Flask, Flask-SQLAlchemy, Flask-WTF, Flask-Migrate, requests, OpenWeatherMap

9. virtual_teacher (Образовательный проект KARTStudy) - это сайт на Django, на котором вы можете подобрать себе образовательный курс для подготовки к экзаменам, вступительным испытаниям или просто для пополнения своих знаний. К сайту прилагается бот в Telegram и

API, с помощью которых так же можно совершать все необходимые операции. Проект разрабатывал, опираясь на свой опыт с сфере преподавания.
Стек: python, django, DRF, djoser, simple-jwt, python-dotenv, pythonTelegramBotAPI, requests

Август 2022 —
настоящее время
3 года 1 месяц

Репетитор по математике и физике
Москва

Преподаватель

Имею опыт преподавания технических предметов. Могу доступно преподнести и объяснить материал.

Образование

Неоконченное высшее

2026

Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет), Москва

Факультет робототехники и комплексной автоматизации, Автоматизация технологических процессов и производств (по отраслям)

Повышение квалификации, курсы

2025

Python-разработчик расширенный
Яндекс, backend-разработка

Навыки

Знание языков

Русский — Родной
Английский — B1 — Средний

Навыки

Python PostgreSQL Django Framework Работа с базами данных Docker
Flask API GitHub REST API aiohttp Backend FastAPI C++
pandas Numpy Linux SQL СУБД Git Pytest Kanban Django
Scrum JSON API Базы данных SQLAlchemy

Дополнительная информация

Обо мне

Мотивация: участвовать в разработке продуктов и сервисов, которыми будут пользоваться другие компании для решения своих задач. Для меня важна атмосфера в команде, для того чтобы вместе оценивать результат общих усилий, перенимать друг у друга опыт, учиться новому. Python выбрал за возможность быстрой реализации функциональных и разнообразных проектов.

Для повышения знаний этой сфере изучал следующие источники:

- 1. Марк Лутц "Изучаем Python";
- 2. Мэттью Фаумер "Asyncio и конкурентное программирование на python";
- 3. Различные курсы Python-разработки.

Благодаря опыту в сфере преподавания достаточно легко нахожу общий язык с человеком. Обладаю высоким уровнем стрессоустойчивости и самодисциплины, выносив к большому

объему работы. С интересом подхожу к новым задачам и возможности узнать что-либо новое.

Дополнительно:

- Учусь в МГТУ им. Н. Э. Баумана, Автоматизация технологических процессов и производств
- Окончил курсы от Яндекса: python-разработчик расширенный (14 месяцев)

Увлечен спортом и познаванием программирования.

На связи:

- в Telegram @kspohodyaev
- почта: konstantinpohodyaev@yandex.ru