**Черабаев Андрей Александрович**

15.05.1993 г.р.

**О специалисте**

| ***Направление*** | *Python* |
| --- | --- |
| ***Стаж разработки по направлению (Python)*** | *более 6 лет* |
| ***Общий стаж практической разработки*** | *более 10 лет* |
| ***Образование*** | *Высшее: УлГУ, Инженерно-физический факультет высоких технологий, Стандартизация и сертификация (2010-2015 г.)* |
| ***Гражданство*** | *РФ* |
| ***Локация*** | *Ульяновск (МСК+1)* |
| ***Иностранные языки*** | *Английский (B2)* |

**Навыки и знания**

| ***Профессиональные навыки*** | * *Опыт работы с асинхронным Python для высоконагруженных сервисов.* * *Глубокое понимание структур данных и алгоритмов для оптимизации производительности кода.* * *Практика использования Event-driven архитектуры (Kafka, RabbitMQ, Redis Streams).* * *Работа с ORM (SQLAlchemy, Django ORM, Tortoise ORM) и понимание их плюсов/минусов в разных случаях.* * *Опыт работы с PostgreSQL настройка индексов, шардирование, репликация.* * *Использование NoSQL-хранилищ (MongoDB, Redis, ClickHouse) в продакшене.* * *Опыт написания юнит- и интеграционных тестов (pytest, unittest, mock, coverage).* * *Контейнеризация и оркестрация с Docker, Kubernetes.* * *Опыт работы с CI/CD (GitHub Actions, GitLab CI, Jenkins, ArgoCD).* * *Разработка микросервисной архитектуры и работа с API Gateway.* * *Опыт работы с OAuth 2.0, JWT, OpenID Connect для аутентификации.* * *Оптимизация и профилирование кода.* * *Инструменты логирования и мониторинга (ELK Stack, Grafana, Prometheus, Sentry).* * *Опыт работы с платежными API и банковскими протоколами.* * *Настройка RBAC, ABAC для управления доступами внутри системы.* * *Понимание принципов работы high-load систем и стратегий отказоустойчивости.* | |
| --- | --- | --- |
| ***Основные технологии*** | ***Платформы*** | *MacOS, Linux, Windows* |
| ***Языки программирования*** | *Python, PHP, JS* |
| ***Инструменты*** | *Django, DRF, FastAPI, SQLModel, SQLAlchemy, Alembic, Typer, PyJWT, asyncio, ELK Stack, Redis, RabbitMQ, Amazon S3, Celery, Docker, docker-compose, K8S, Nginx, PyTest, REST, GraphQL, git, Swagger, HTML/CSS, GitLab CI/CD, Kafka, pdm, ruff, black, flake8, mypy, coverage* |
| ***Базы данных*** | *PostgreSQL, MySQL, SQLite* |
| ***Сферы проектной деятельности*** | *FinTech, E-commerce, Transport, LifeStyle* | |

**Участие в проектах**

| **Внутрибанковская система управления и координирования выдачей оборудования** | |
| --- | --- |
| ***Период работы*** | *12.2022 - 02.2025* |
| ***Длительность*** | *2 года 3 месяца* |
| ***Роль*** | *Backend разработчик (FastAPI, Django)* |
| ***Описание проекта*** | *Внутренняя система учеты и выдачи оборудоавния сотрулникам. Есть возможность заводить оборудование, назначать правила выдачи, прослеживать подписание ЭЦП. Много интеграций с другими сервисами внутри банка: авторизация, S3, система ЭЦП, реестр увольнений, служба пропусков.* |
| ***Состав команды*** | *1 фронтенд-разработчик, 4 бэкенд-разработчиков, 1 QA, 1 DevOps, 3 аналитика, 1 ПМ* |
| ***Технологии на проекте*** | *FastAPI, Django, DRF, PostgreSQL, OpenSearch, SQLModel, Typer, SQLAlchemy, Alembic, Kafka, grafana, docker, docker-compose, CI/CD, Kubernetes, Amazon S3, Celery, redis, GitLab, REST, openapi* |
| ***Задачи/реализованный функционал*** | *Инициировал и возглавил миграцию проекта с Django+DRF на FastAP. Оптимизировал взаимодействие с базой данных, внедрив SQLModel и SQLAlchemy для работы с PostgreSQL.*  *Участвовал в разработке архитектуры новой версии системы, внедрив разделение данных по подразделениям сотрудников с использованием PostgreSQL schemas.*  *Оптимизировал процесс авторизации и управления доступами, внедрив OAuth 2.0 и OpenID Connect, а также реализовав интеграцию с внутренними сервисами банка, включая систему единого входа (SSO).*  *Разработал и внедрил централизованный логинг и мониторинг с использованием OpenSearch, Grafana, Prometheus и Sage*  *Настроил обработку событий через Kafka, для взаимодействия системы с другими банковскими сервисами (например, автоматическое обновление реестра уволенных сотрудников и блокировка доступа к оборудованию).*  *Реализовал интеграцию с системой цифровых подписей (ЭЦП) для подтверждения получения и возврата оборудования сотрудниками.*  *Принимал участие в разработке микросервиса для управления хранилищем документов на Amazon S3.*  *Оптимизировал обработку фоновых задач (например, регулярные проверки доступности оборудования и автоматическое уведомление сотрудников) с использованием Celery и Redis.*  *Совместно с DevOps специалистами разработывал CI/CD пайплайны на GitLab*  *Активно участвовал в код-ревью и менторстве команды, внедрял новые возможности языка (например, асинхронные обработчики в FastAPI)* |

| **Маркетплейс банковских услуг** | |
| --- | --- |
| ***Период работы*** | *05.2022 - 12.2022* |
| ***Длительность*** | *7 месяцев* |
| ***Роль*** | *Backend разработчик (FastAPI)* |
| ***Описание проекта*** | *Сервис подбора банковских услуг – вклады, страхование, кредиты. Платформа предоставляет пользователям персонализированные рекомендации на основе анализа их финансового профиля, включая кредитную историю, доходы и расходы. Пользователи могут сравнивать предложения от различных банков и страховых компаний, фильтруя их по ключевым параметрам: процентные ставки, сроки, условия страхования и дополнительные опции. Для удобства пользователей доступен сквозной поиск по всем категориям услуг с фильтрацией по тегам, рейтингам банков и отзывам других клиентов. Платформа интегрирована с внешними API банков и страховых компаний, что позволяет получать актуальные данные в режиме реального времени.* |
| ***Состав команды*** | *1 фронтенд-разработчик, 4 бэкенд-разработчиков, 1 QA, 1 DevOps, 3 аналитика, 1 ПМ* |
| ***Технологии на проекте*** | *FastAPI, PostgreSQL, SQLAlchemy, Alembic, docker, docker-compose, CI/CD, Celery, redis, GitLab, REST, openapi* |
| ***Задачи/реализованный функционал*** | *Разрабатывал и внедрял новый функционал, включая систему персонализированных рекомендаций банковских продуктов на основе анализа финансового профиля пользователей. Использовал FastAPI + SQLAlchemy для построения высокопроизводительных API.*  *Интегрировал внешние API банков и страховых компаний, обеспечив автоматическое обновление данных по кредитам, вкладам и страховым программам в режиме реального времени. Для этого использовал asyncio, Celery, Kafka и балансировку нагрузки на API.*  *Принимал участие в разработке системы скоринга пользователей, анализируя их транзакционную активность, кредитную историю и платежеспособность.*  *Оптимизировал хранение данных, реализовав денормализацию ключевых таблиц и использование индексов в PostgreSQL*  *Реализовал систему кэширования с использованием Redis*  *Настроил сквозной поиск по финансовым продуктам, используя OpenSearch и внедрив полнотекстовый поиск с фильтрацией по ключевым параметрам (ставки, сроки, рейтинг, отзывы).*  *Реализовал автоматическую обработку и обогащение данных, используя Kafka для потоковой обработки информации о пользователях, их финансовом поведении и актуальных предложениях банков.*  *Настроил систему логирования и мониторинга с Grafana, Prometheus, Sentry*  *Проводил ревью кода* |

| **Корпоративная соцсеть** | |
| --- | --- |
| ***Период работы*** | *03.2021 - 05.2022* |
| ***Длительность*** | *1 год 2 месяца* |
| ***Роль*** | *Backend разработчик (FastAPI)* |
| ***Описание проекта*** | *Внутрикорпоративная социальная сеть, позволяющая пользователям создавать сообщества, делиться новостям, получать награды, благодарности и внутреннюю валюту за проявленную активность, управлять мероприятиями в календаре, загружать и просматривать файлы. Так же есть возможность создания и управления организационной структурой, что позволяет выстраивать иерархию сотрудников. В рамках платформы у пользователей есть возможность создавать и администрировать свои собственные “внутренние сайты”/блоги.* *Присутствует поиск как по отдельным сервисам, так и сквозной, включая поиск по тегам.* |
| ***Состав команды*** | *9 фронтенд-разработчиков, 12 бэкенд-разработчиков, 6 QA, 2 DevOps, 4 аналитика, 1 ПМ* |
| ***Технологии на проекте*** | *FastAPI, PostgreSQL, ClickHouse, SQLAlchemy, Alembic, asyncio, Kafka, grafana, black, ruff, flake8, mypy, coverage, pdm, pytest, docker, docker-compose, CI/CD, Kubernetes, Amazon S3, Celery, redis, GitLab, Rest, openapi* |
| ***Задачи/реализованный функционал*** | *Реализовал ролевую модель*  *Принимал участие в реализации сервиса конструктора сайтов*  *Создал скрипты для миграции данных из старых сервисов в новые*  *Провел исследование инструментов для форматирования кода, выбрал наиболее подходящий, настроил выбранный инструмент в проекте и добавил его в CI/CD*  *Регулярно проводил ревью кода* |

| **CRM для банковской сферы** | |
| --- | --- |
| ***Период работы*** | *10.2021 - 03.2022* |
| ***Длительность*** | *6 месяцев* |
| ***Роль*** | *Backend разработчик (FastAPI, Django)* |
| ***Описание проекта*** | *CRM для приложения по инвестированию, предоставляющая инструменты для управления клиентами, отслеживания их инвестиционной активности и анализа финансовых портфелей. Платформа позволяет инвестиционным менеджерам создавать персональные профили клиентов, включая их финансовые цели. Система поддерживает автоматическое отслеживание всех транзакций клиентов, включая покупку и продажу активов, а также предоставляет аналитику по доходности портфелей. Имеет сквозной поиск по всем данным системы, включая клиентов, сделки, активы и теги.* |
| ***Состав команды*** | *3 бэкенд-разработчика, 1 ПМ* |
| ***Технологии на проекте*** | *FastAPI, Django, DRF, PostgreSQL, OpenSearch, SQLAlchemy, Alembic, Kafka, grafana, docker, docker-compose, CI/CD, Kubernetes, Amazon S3, Celery, redis, GitLab, REST, openapi* |
| ***Задачи/реализованный функционал*** | *Оптимизация структуры базы данных*  *Участие в переносе кодовой базы с Django на FastAPI* |

| **Интернет-магазин электроники** | |
| --- | --- |
| ***Период работы*** | *04.2021 - 10.2021* |
| ***Длительность*** | *6 месяцев* |
| ***Роль*** | *Backend разработчик (FastAPI)* |
| ***Описание проекта*** | *Онлайн-сервис поиска и покупки электроники и гаджетов, предоставляющий пользователям возможность сравнивать товары от различных продавцов, фильтровать их по характеристикам, ценам и отзывам. Платформа поддерживает персонализированные рекомендации на основе истории просмотров и покупок, а также предпочтений пользователя. Реализован сквозной поиск по всему каталогу товаров, включая фильтрацию по тегам, брендам и категориям.* |
| ***Состав команды*** | *1 фронтенд-разработчик, 2 бэкенд-разработчиков* |
| ***Технологии на проекте*** | *FastAPI, MySQL, React, ElasticSearch, SQLAlchemy, Alembic, docker, docker-compose, Kubernetes, GitLab, REST* |
| ***Задачи/реализованный функционал*** | *Разработал архитектуру поисковой системы на основе ElasticSearch*  *Создал обертку над библиотекой elasticsearch-py для упрощения работы с поисковыми запросами*  *Реализовал сложные поисковые запросы, маппинг документов и настройку производительности ElasticSearch*  *Разработал административную часть для управления поисковыми индексами и настройками*  *Регулярно проводил ревью кода, обеспечивая его качество и соответствие стандартам разработки* |

| **Интернет-магазин смесей для электронных сигарет (Китай)** | |
| --- | --- |
| ***Период работы*** | *04.2019 - 04.2021* |
| ***Длительность*** | *2 года* |
| ***Роль*** | *Backend разработчик (FastAPI)* |
| ***Описание проекта*** | *Сервис предоставляет возможность поиска и подбора составов для электронных сигарет. Главная особенность проекта в минимизации создания новых рецептов, взамен предлагается составить рецепт, на основе которого система подберет максимально близкий из уже существующих. Это реализовано через алгоритм нечеткого поиска по составу, при котором мы ищем рецепты, отличающиеся по составу на определенное количество ингредиентов или на определенную погрешность в составе.* |
| ***Состав команды*** | *1 фронтенд-разработчик, 3 бэкенд-разработчиков* |
| ***Технологии на проекте*** | *FastAPI, PostgreSQL, ElasticSearch, SQLAlchemy, Alembic, docker, docker-compose, CI/CD, AlibabaCloud, GitLab, REST, openapi* |
| ***Задачи/реализованный функционал*** | *Управление командой разработки из 4 человек*  *Написание бэкенда приложения*  *Реализация алгоритма нечеткого поиска похожих рецептов с погрешностями в составе*  *Реализация полнотекстового поиска через ElasticSearch*  *Архитектура базы данных* |