



Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное
учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.03 Прикладная информатика

О Т Ч Е Т

по индивидуальному заданию

Название: Разработка микросервиса каталога фильмов

Дисциплина: Проектирование информационных систем

Студент

ИУ6-75Б
(Группа)

(Подпись, дата)

Р.Х. Касим

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

А.Ф. Прутик

(И.О. Фамилия)

Москва, 2025

Цель работы: целью данной работы является разработка микросервиса каталога фильмов для распределённой системы онлайн-кинотеатра. Сервис предназначен для хранения и предоставления информации о фильмах, жанрах, странах и участниках фильмов, а также для административного управления каталогом.

Требования к системе

В рамках работы были реализованы следующие требования:

- предоставление REST API для получения списка фильмов и детальной информации;
- поддержка поиска, фильтрации и пагинации;
- управление справочниками (жанры, страны, персоны);
- административное управление фильмами (создание, обновление, удаление);
- защита административных эндпоинтов с помощью токена администратора;
- хранение данных в реляционной базе данных PostgreSQL.

1 Описание решения и используемые технологии

Микросервис реализован с использованием микросервисной архитектуры. Сервис является независимым backend-компонентом и взаимодействует с клиентами через REST API.

Используемый стек технологий:

Backend:

- Python 3.11;
- FastAPI — фреймворк для создания REST API;
- Uvicorn — ASGI-сервер;
- SQLAlchemy — ORM для работы с базой данных.
- База данных:
- PostgreSQL.

Инфраструктура:

- Docker;
- Docker Compose.

2 Архитектурные схемы

В рамках работы были разработаны архитектурные диаграммы:

- диаграмма контекста (C4 System Context);
- диаграмма контейнеров (C4 Container);
- диаграмма компонентов (C4 Component);
- диаграмма кода (UML Class Diagram).

Диаграммы представлены на рисунках 1-4. Диаграмма кода отражает внутреннюю структуру микросервиса и показывает взаимодействие между слоями API, CRUD, ORM-моделями и базой данных.

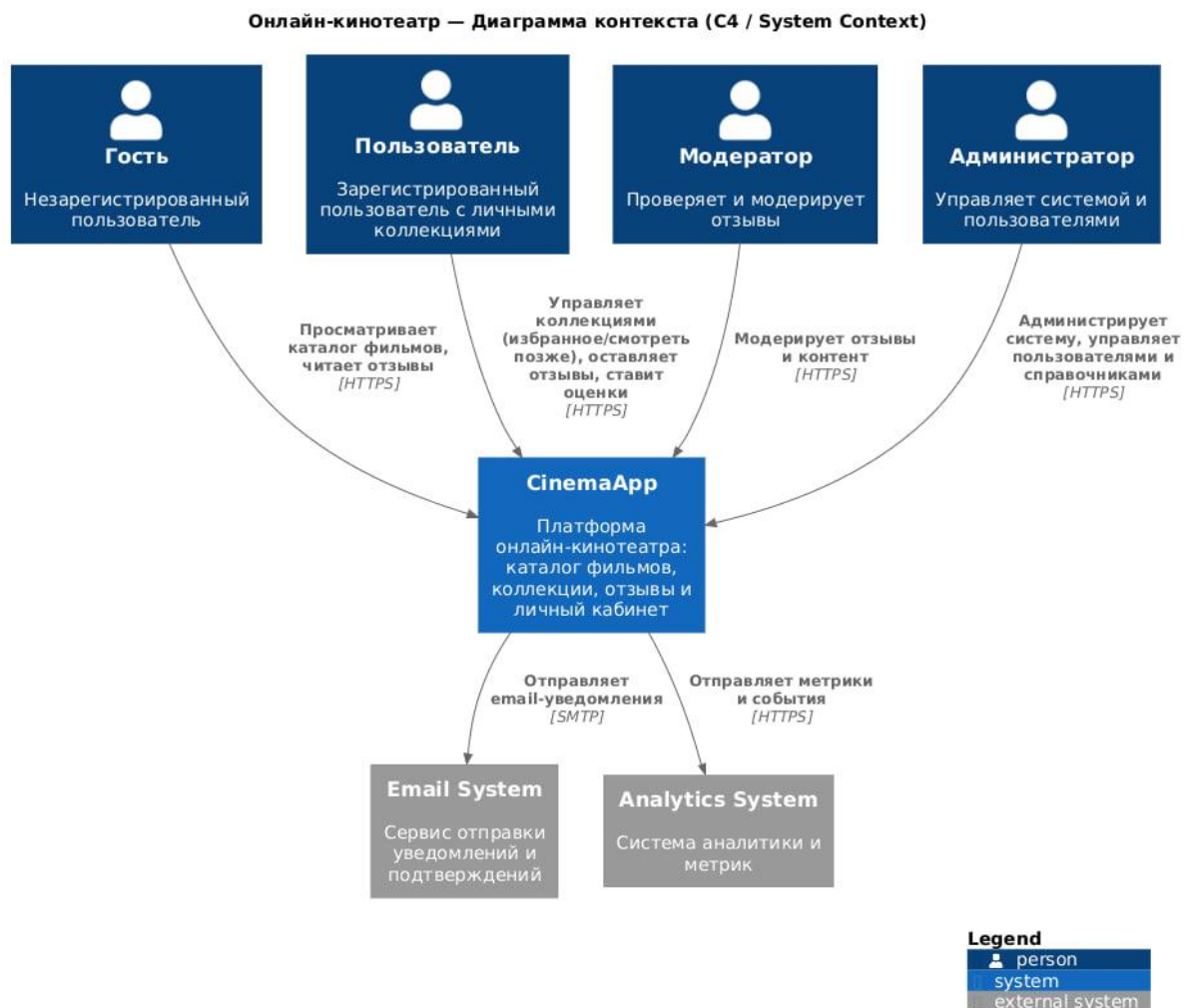


Рисунок 1 – диаграмма контекста

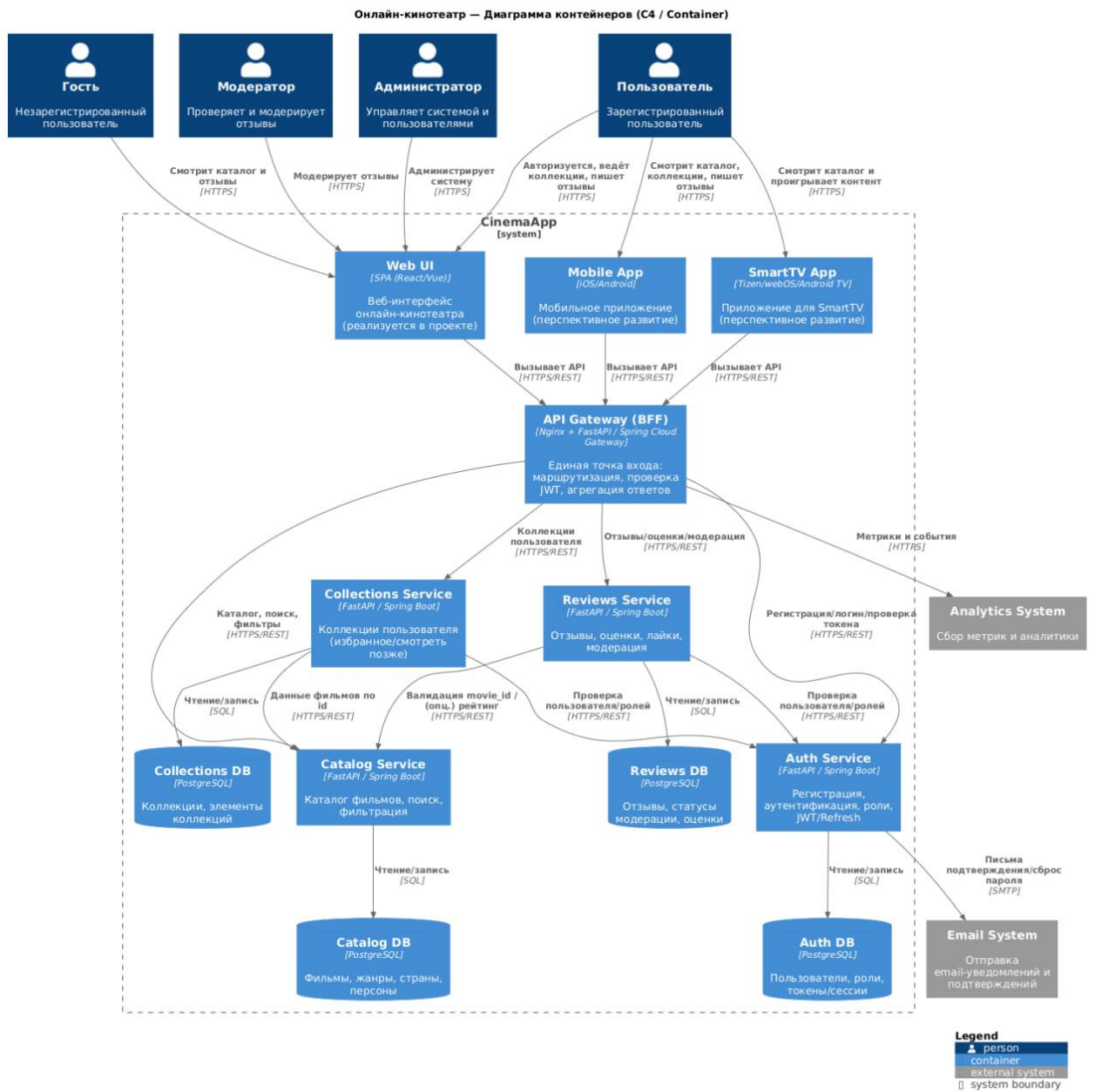


Рисунок 2 – диаграмма контейнеров

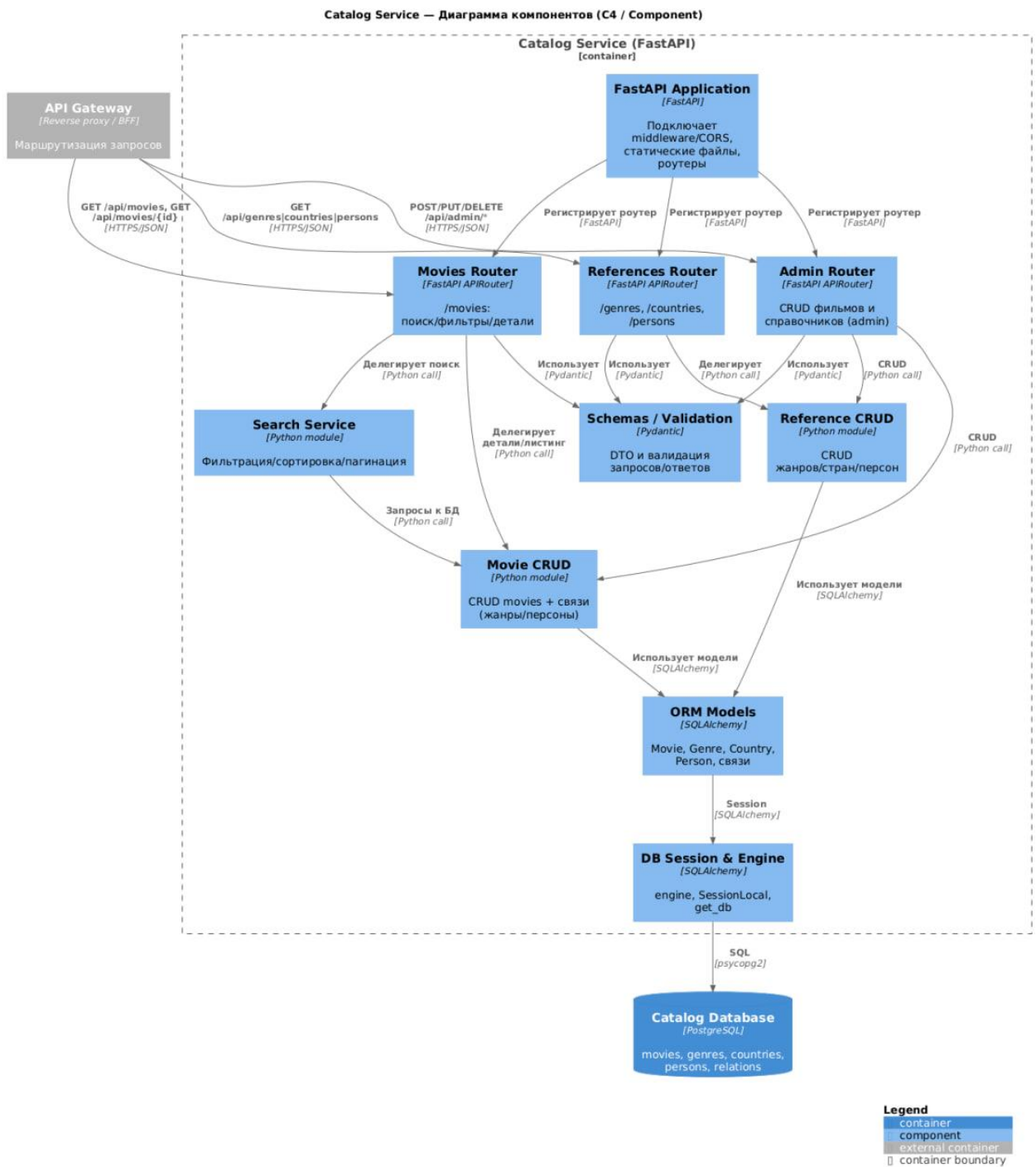


Рисунок 3 – Диаграмма компонентов

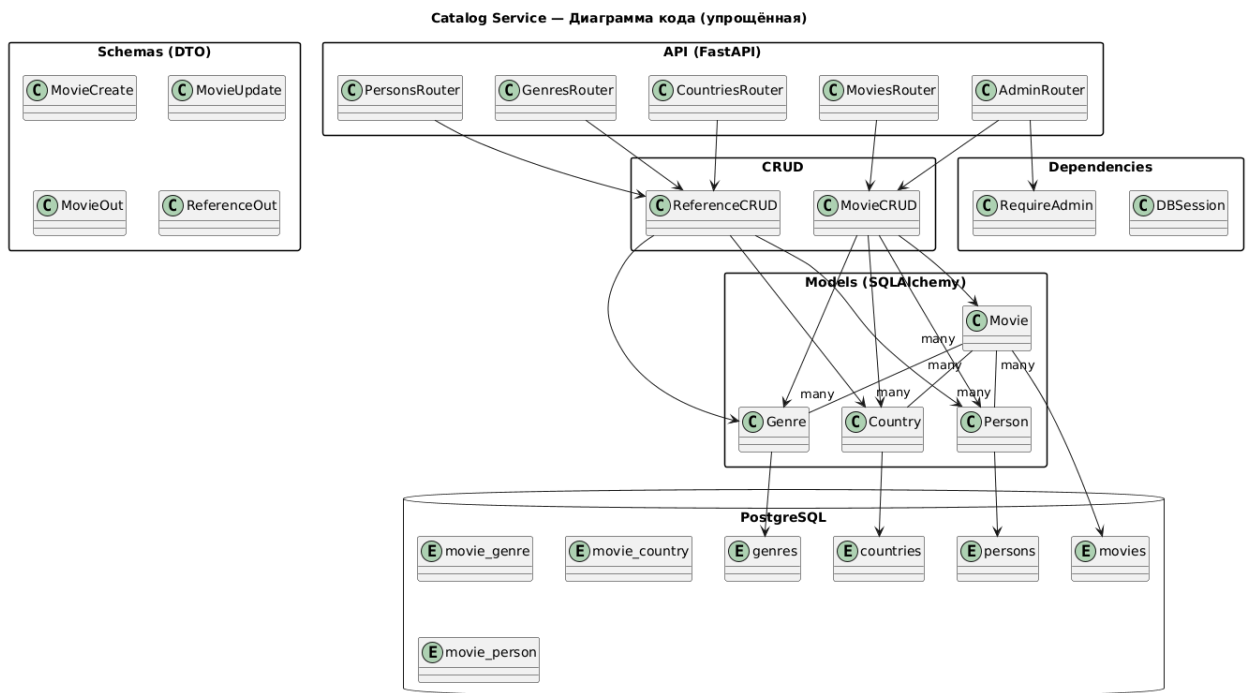


Рисунок 4 – Диаграмма кода

3 Детали реализации

Микросервис каталога предоставляет следующие основные эндпоинты:

Публичные эндпоинты:

- GET `/api/movies` — получение списка фильмов;
- GET `/api/movies/{id}` — получение информации о фильме;
- GET `/api/genres` — получение списка жанров;
- GET `/api/countries` — получение списка стран;
- GET `/api/persons` — получение списка персон.

Административные эндпоинты:

- POST `/api/admin/movies` — добавление фильма;
- PUT `/api/admin/movies/{id}` — обновление фильма;
- DELETE `/api/admin/movies/{id}` — удаление фильма.

Доступ к административным эндпоинтам защищён с помощью заголовка `X-Admin-Token`.

Вывод: в результате работы был успешно спроектирован и реализован микросервис каталога фильмов. Сервис обеспечивает централизованное хранение и предоставление данных о фильмах и справочниках, поддерживает административное управление каталогом и легко интегрируется в общую архитектуру онлайн-кинотеатра.