Бэкенд

sber\_cert.pem – сертификат для работы с gigachat

requirements.txt – набор версий библиотек для стартовой загрузки и корректной работы приложения

Dockerfile.txt – Dockerfile создает образ для запуска Python-приложения

docker-compose.yml – файл конфигурации Docker Compose определяет два сервиса (web и db) и настраивает их взаимодействие.

.env – содержит переменные окружения для приложения. Они используются для настройки работы сервисов и обеспечения безопасности.

* APP

database.py – программа на Python настраивает асинхронное подключение к базе данных PostgreSQL с использованием SQLAlchemy

main.py – основа для сервиса рекомендаций, которая, работает с книгами, фильмами и, использует ИИ для генерации описания

wait\_for\_db.py – программа предназначена для проверки доступности PostgreSQL базы данных перед запуском основного приложения. Она пытается подключиться к БД несколько раз с интервалами, прежде чем сдаться.

* \_\_pycache\_\_ - КЭШ ПРОГРАММЫ
* Core
* \_\_pycache\_\_ - КЭШ ПРОГРАММЫ

config.py - программа представляет собой модуль настроек для приложения

constants.py – списки авторов и жанров

settings.py – базовые настройки

* Models
* \_\_pycache\_\_ - КЭШ ПРОГРАММЫ

book.py - код создает модель SQLAlchemy для таблицы книг в базе данных

content.py - код создает модель SQLAlchemy для таблицы фильмов и сериалов в базе данных

user.py - код определяет модель User для работы с базой данных

* Routers
* \_\_pycache\_\_ - КЭШ ПРОГРАММЫ

ai.py - программа нужна для генерации краткого описания контента с использованием GigaChat API.

auth.py - программа реализует систему аутентификации

books.py - программа предоставляет API для поиска книг и получения информации о них

movieas.py – программа предоставляет API для взаимодействия с данными Кинопоиска

preferences.py - программа реализует работу с пользовательскими предпочтениями и рейтингами с использованием FastAPI и асинхронной работы с базой данных через SQLAlchemy

user.py - программа представляет собой API для управления пользовательскими данными

* schemas
* \_\_pycache\_\_ - КЭШ ПРОГРАММЫ

ai.py – программа определяет две Pydantic-модели для запроса и ответа API, связанного с суммаризацией контента

book.py – определяет Pydantic-модели для запроса и ответа API, связанного с книгами

token.py - определяет Pydantic-модели для запроса и ответа API, связанного с токеном

user.py - определяет Pydantic-модели для запроса и ответа API, связанного с пользователем

* services
* \_\_pycache\_\_ - КЭШ ПРОГРАММЫ

auth.py - программа реализует систему аутентификации с использованием JWT в FastAPI

gigachat\_client.py - программа представляет собой клиент для работы с API GigaChat

kinopoisk\_client.py - программа представляет собой обертку для работы с неофициальным API Кинопоиска

open\_library.py - программа представляет собой асинхронный клиент для работы с API OpenLibrary

user\_service.py - программа предоставляет асинхронные функции для работы с пользователями в базе данных

* utils
* \_\_pycache\_\_ - КЭШ ПРОГРАММЫ

file\_upload.py - программа представляет собой функцию для сохранения профильных изображений пользователей в FastAPI приложении

security.py - программа предоставляет функционал для работы с аутентификацией пользователей в FastAPI-приложении.