

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»



Направление подготовки/специальность
09.03.01 Информатика и вычислительная техника

направленность (профиль)
«Технологии разработки программного обеспечения»

Раздаточный материал к выпускной квалификационной работе

«Разработка бэкенда веб-системы для театральной индустрии»

Обучающегося 4 курса
очной формы обучения
Букина Даниила Юрьевича

Руководитель выпускной квалификационной
работы:
Доктор педагогических наук, профессор
кафедры информационных технологий и
электронного обучения
Власова Елена Зотиковна

Содержание работы

ВВЕДЕНИЕ	3
1. АНАЛИЗ ТЕХНОЛОГИЙ ДЛЯ РАЗРАБОТКИ БЭКЕНДА	5
1.1 О бэкенд разработке	5
1.2 Об экосистеме Spring Boot.....	6
1.3 СУБД и технологии ORM.....	8
1.4 Язык программирования Kotlin	13
1.5 Об архитектуре приложений	15
2 ПРОЕКТИРОВАНИЕ ВЕБ-СИСТЕМЫ.....	21
2.1 Проектирование базы данных	21
2.2 Проектирование архитектуры приложения	29
3 РАЗРАБОТКА БЭКЕНДА	33
3.1 Инициализация проекта и управление зависимостями	33
3.2 Разработка слоёв приложения.....	37
3.3 Тестирование и проверка работоспособности приложения	44
3.4 Документация API.....	47
3.5 Размещение приложения с помощью Docker Compose.....	50
ЗАКЛЮЧЕНИЕ.....	53
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	54

ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Целью работы является разработка серверной части (бэкенда) веб-базируемого сервиса для театральной индустрии с возможностью просмотра информации о пьесах и добавления пользователями отзывов. Современные приложения позволяют разграничить труд разработчиков на две составляющие – бэкенд и фронтенд. В данной работе рассмотрен процесс разработки именно первого сегмента.

- анализ технологий, доступных для бэкенд разработки;
- выбор технологического стека;
- разработка архитектуры базы данных;
- проектирование системы;
- разработка программных модулей;
- оформление документации.

ВЫБРАННЫЕ ТЕХНОЛОГИИ

- Kotlin – язык программирования. Современный, надёжный, JVM-based.
- Spring Boot – web framework. Отлично для создания RESTful API
- PostgreSQL – СУБД
- Spring Data – доступ к данным. Удобные шаблонные репозитории и внедрение собственных запросов
- Spring Security – разграничение ресурсов
- Swagger UI – документация
- Docker Compose – развёртывание и контейнеризация

АРХИТЕКТУРНОЕ РЕШЕНИЕ

В качестве архитектурного шаблона был выбран паттерн Clean Architecture. Его схема показана на рисунке 1.

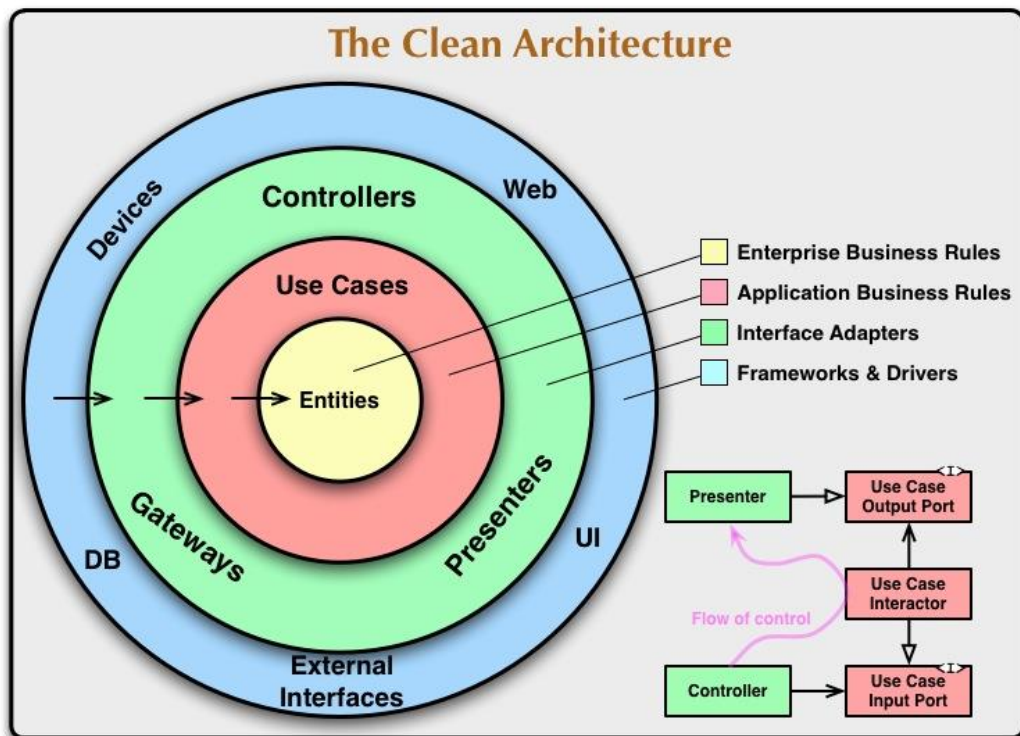


Рисунок 1. Паттерн Clean Architecture

В качестве имплементации (реализации) этого паттерна был выбран шаблон Repository-Service. Его схема показана на рисунке 2.

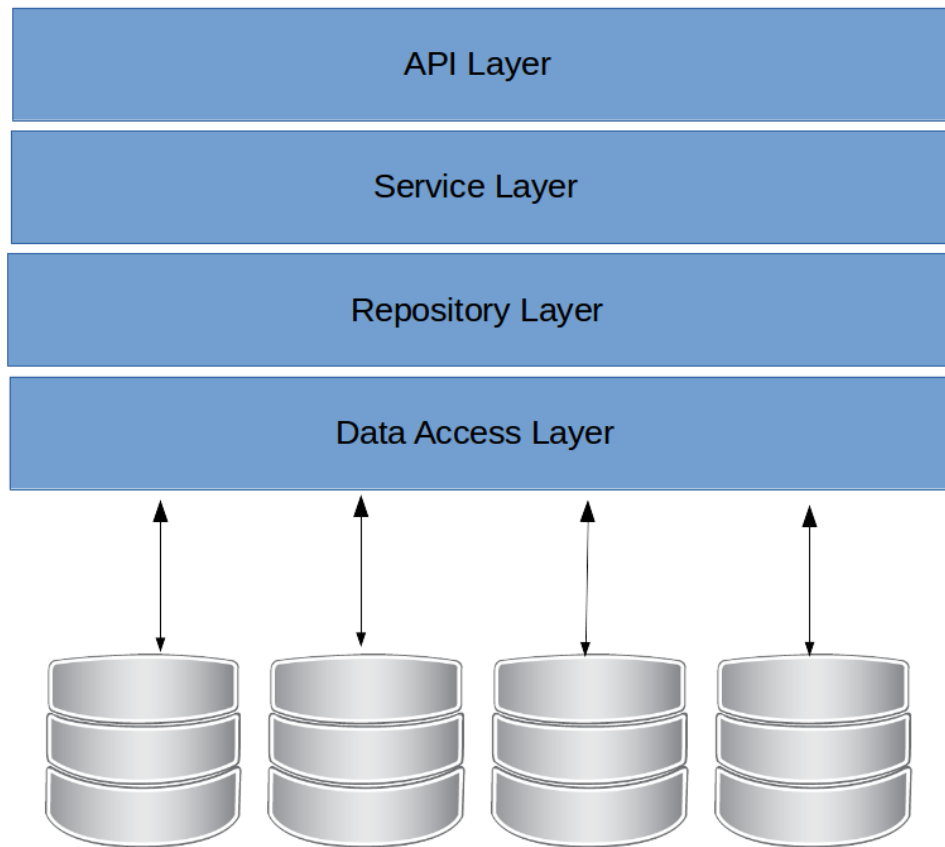


Рисунок 2. Паттерн Repository-Service

ДОКУМЕНТАЦИЯ ПРОЕКТА

Скриншот интерфейса документации проекта показан на рисунке 3.

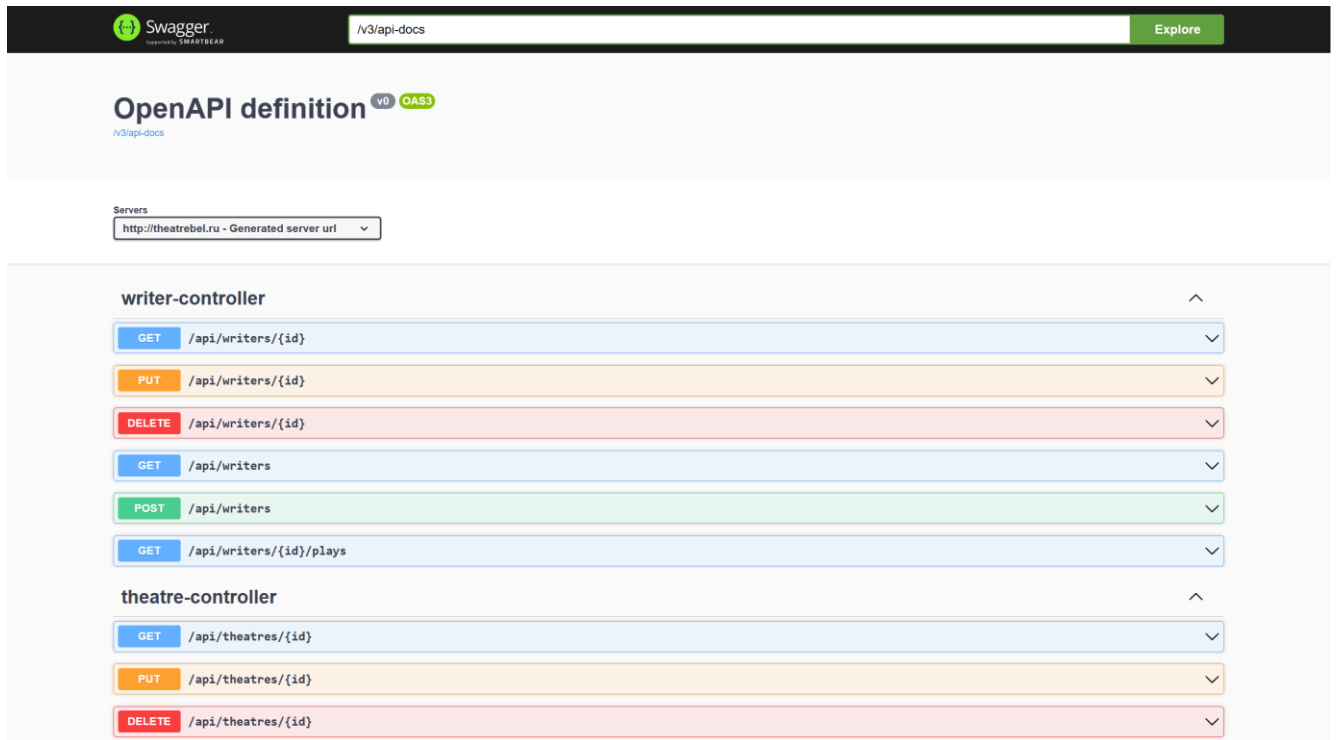


Рисунок 3. Интерфейс документации

QR-код с ссылкой на документацию проекта приведён на рисунке 4.



Рисунок 4. Ссылка на документацию

ОПУБЛИКОВАННЫЙ ПРОЕКТ

QR-код с ссылкой на опубликованный проект приведён на рисунке 5.



Рисунок 5. Ссылка на проект

К API можно обращаться с помощью клиентов, например, curl или Postman

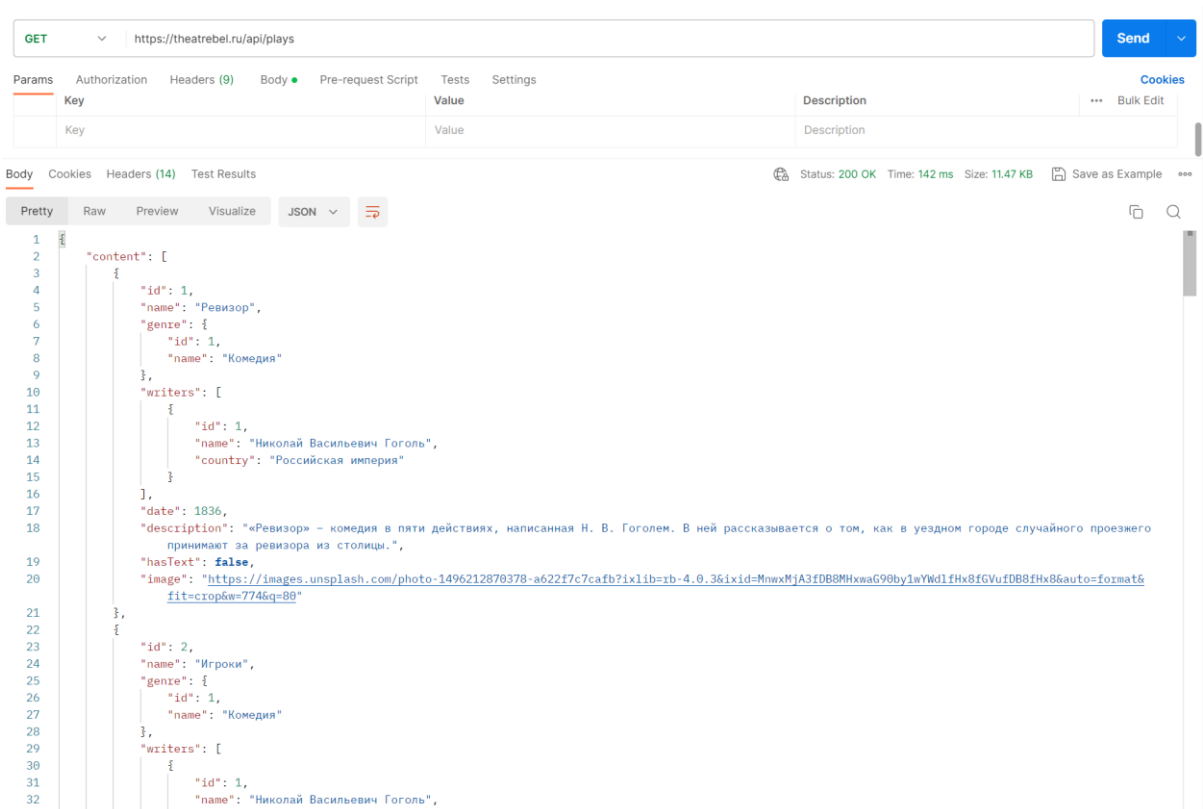


Рисунок 6. Обращение к API через Postman