

# Сайт “Что посмотреть?”

---

Спецификация требований  
к программному обеспечению  
Версия 1.0

# Содержание

<b>1. Введение</b>	<b>3</b>
1.1. Назначение	3
1.2. Область применения	3
1.3. Определения, акронимы и сокращения	3
1.4. Ссылки	4
<b>2. Общее описание</b>	<b>4</b>
2.1. Видение продукта	4
2.2. Перспективы продукта	4
2.4. Классы и характеристики пользователей	5
<b>3. Требования к программе</b>	<b>5</b>
3.1. Требования функционального характера	5
3.2 Требования нефункционального характера	7
3.2.1 Требования к производительности	7
3.2.2 Требования к безопасности	7
3.2.3 Требования к внешним системам	7
3.2.4 Атрибуты качества	7
3.2.5 Интерфейсы передачи данных	8
<b>4. UML и ER диаграммы</b>	<b>9</b>
4.1 Диаграмма вариантов использования	9
4.2 Диаграмма классов	10
4.3 ER-диаграмма	11
<b>Приложения</b>	<b>14</b>
Приложение 1. Результаты интервью	14
Приложение 2. Описание вариантов использования	14
Приложение 3. Описание программного интерфейса	15

# 1. Введение

## 1.1. Назначение

Спецификация требований к программному обеспечению описывает функциональные и нефункциональные требования к выпуску 1.0 сайта «Что посмотреть?».

Этот документ предназначен для команды, которая будут реализовывать и проверять корректность работы сайта.

Кроме специально обозначенных случаев, все указанные здесь требования имеют высокий приоритет и приписаны к выпуску 1.0.

## 1.2. Область применения

Сайт «Что посмотреть?» разработан для онлайн-кинотеатра и позволяет пользователям формировать списки просмотренных фильмов, сериалов и мультфильмов, а также списки тех кинокартин, которые пользователь хотел бы посмотреть в будущем.

Полная функциональность сайта перечислена в разделах «Функциональные требования» и «Нефункциональные требования».

## 1.3. Определения, акронимы и сокращения

**Пользователь** - человек, использующий сайт для решения стоящих перед ним задач.

**Подборка** – термин, объединяющий в себе:

- наименование,
- описание,
- свойство (приватная или публичная),
- фильмы, сериалы, мультфильмы (могут отсутствовать).

**Приватная подборка** – подборка доступная для просмотра только пользователю, создавшему ее.

**Публичная подборка** – подборка доступная для просмотра всем пользователям.

**Сайт** – сайт «Что посмотреть?», разработанный в соответствии с данной спецификацией.

**API** – функция (или набор функций) приложения, которую можно вызвать из другого приложения.

**Apple** – учетная запись пользователя Apple.

**Google** – учетная запись пользователя Google.

**VK** – учетная запись пользователя ВКонтакте.

**UML** - унифицированный язык моделирования

**Авторизация** – процесс проверки (подтверждения) данных прав при попытке выполнения этих действий.

**Фильмы** – фильмы, сериалы, мультфильмы.

## 1.4. Ссылки

- [Авторизация через Apple \(API Sign in with Apple\);](#)
- [Использование OAuth 2.0 для доступа к API Google;](#)
- [API ВКонтакте;](#)
- [API Яндекс.Метрики](#)
- [Руководство по обеспечению доступности веб-контента \(WCAG\) 2.1](#)

## 2. Общее описание

### 2.1. Видение продукта

Сайт предоставляет следующую функциональность:

- авторизацию через аккаунты в социальных сетях;
- создание публичных и частных подборок фильмов;
- добавление заметок о фильмах в частных подборках;
- возможность получить случайный фильм для просмотра из выбранной подборки;
- просматривать публичные подборки фильмов;
- удалять свои подборки.

### 2.2. Перспективы продукта

В будущем разрабатываемый сайт станет основным сайтом онлайн-кинотеатра, с возможностью смотреть фильмы, не переходя на другой сервис.

## 2.4. Классы и характеристики пользователей

Класс	Характеристика
Неавторизованный	<p>Посетитель сайта, который не прошел авторизацию.</p> <p>Обладает на сайте следующими правам:</p> <ul style="list-style-type: none"><li>● поиск фильмов;</li><li>● переход на сайт онлайн-кинотеатра для просмотра фильма;</li><li>● просмотр публичных подборок с фильмами;</li><li>● получить случайный фильм из публичной подборки.</li></ul>
Авторизованный	<p>Посетитель сайта, который прошел авторизацию.</p> <p>Обладает на сайте следующими правами:</p> <ul style="list-style-type: none"><li>● поиск фильмов;</li><li>● создание/удаление приватных и публичных подборок с фильмами;</li><li>● редактирование своих подборок с фильмами;</li><li>● переход на сайт онлайн-кинотеатра для просмотра фильма;</li><li>● просмотр чужих публичных подборок с фильмами;</li><li>● добавление/изменение/удаление заметок к фильмам в приватных подборках.</li></ul>

## 3. Требования к программе

### 3.1. Требования функционального характера

ФТ001. Сайт должен предоставлять возможность авторизоваться через социальные сети:

- VK;
- Google;
- Apple.

ФТ002. При авторизации сайт должен запрашивать данные (ФИО, e-mail, возраст, аватар) пользователя из социальной сети.

ФТ003. Сайт должен предоставлять возможность создавать публичные подборки фильмов, доступные для просмотра другим пользователям.

ФТ004. Сайт должен предоставлять возможность создавать приватные подборки фильмов, не доступные для просмотра другим пользователям.

ФТ005. Подборка фильмов на сайте включает в себя следующее:

- наименование подборки (заполнение обязательно);
- описание подборки (заполнение необязательно);
- свойство подборки: приватная или публичная (заполнение обязательно, по умолчанию - приватная);
- фильмы (заполнение необязательно).

ФТ006. Сайт должен предоставлять возможность редактировать (добавлять/удалять фильмы, изменять название и описание подборки, менять свойство подборки с публичной на приватную и наоборот) публичные и приватные подборки, созданные пользователем.

ФТ007. Сайт должен предоставлять возможность удалить созданную подборку.

ФТ008. Поиск фильмов для добавления в подборку должен осуществляться следующими способами:

- поисковая строка;
- поиск в каталоге с фильмами, используя фильтры:
  - название фильма (полное или частичное);
  - жанр;
  - страна;
  - год выхода или интервал (на выбор пользователя).

ФТ009. Сайт должен автоматически формировать подборку просмотренных пользователем фильмов. Подборка должна быть приватной.

ФТ010. Сайт должен предоставлять возможность удалять фильмы из подборки с просмотренными фильмами.

ФТ011. Сайт должен предоставлять возможность отметить любой фильм как просмотренный, таким образом добавив фильм в подборку с просмотренными фильмами.

ФТ012. Сайт должен предоставлять возможность просматривать публичные подборки других пользователей.

ФТ013. У каждой публичной подборки должно отображаться имя автора подборки.

ФТ014. Сайт должен предоставлять возможность поиск публичных подборок по наименованию подборки.

ФТ015. Сайт должен предоставлять возможность поиск публичных подборок по содержащимся в них фильмам.

ФТ016. Сайт должен предоставлять возможность добавлять заметку к фильму в приватной подборке.

ФТ017. Сайт должен предоставлять возможность редактировать заметку к фильму в приватной подборке.

ФТ018. Сайт должен предоставлять возможность удалять заметку к фильму в приватной подборке.

ФТ019. Если фильм содержится более чем в одной приватной подборке, то заметка к этому фильму должна отображаться во всех приватных подборках пользователя, с указанием названия подборки, в которой была сделана заметка.

ФТ020. Сайт должен предоставлять возможность получить случайный (выбранный рандомно) фильм из любой доступной для просмотра подборки.

ФТ021. Сайт должен предоставлять возможность перейти на сайт онлайн-кинотеатра для просмотра выбранного фильма.

## **3.2 Требования нефункционального характера**

### **3.2.1 Требования к производительности**

1. Максимальное количество пользователей, которые могут одновременно находиться на сайте, равно 10000 человек.
2. При скорости соединения не менее 30 Мбит/с и размере файла не более 200 Кбайт время ответа сервера не должно превышать 200 мс.

### **3.2.2 Требования к безопасности**

Для авторизации пользователей должен использоваться протокол HTTPS.

### **3.2.3 Требования к внешним системам**

Для отслеживания посещаемости сайта должна использоваться Яндекс.Метрика.

### **3.2.4 Атрибуты качества**

1. Сайт должен иметь адаптивный дизайн для мобильных телефонов.
2. Заметка к фильму должна содержать от 1 до 4000 символов.

3. Название подборки должно содержать от 1 до 100 символов.
4. Описание подборки должно содержать от 0 до 2000 символов.
5. Изменение названия или описания подборки с просмотренными фильмами для пользователя недоступно.
6. Публичные подборки должны отображаться в хронологическом порядке по дате их создания.
7. Для поиска публичных подборок по фильмам, содержащимся в них, необходимо указать от 1 до 5 фильмов.
8. Доступность сайта должна соответствовать уровню АА руководства по обеспечению доступности веб-контента (Web Content Accessibility Guidelines (WCAG)).

### **3.2.5 Интерфейсы передачи данных**

API работы с подборками фильмов описан в swagger-документе (Приложение 3).



## 4. UML и ER диаграммы

### 4.1 Диаграмма вариантов использования

Основные функциональные возможности сайта отображены на диаграмме вариантов использования (Рисунок 1). Описание вариантов использования приведено в Приложении 2.

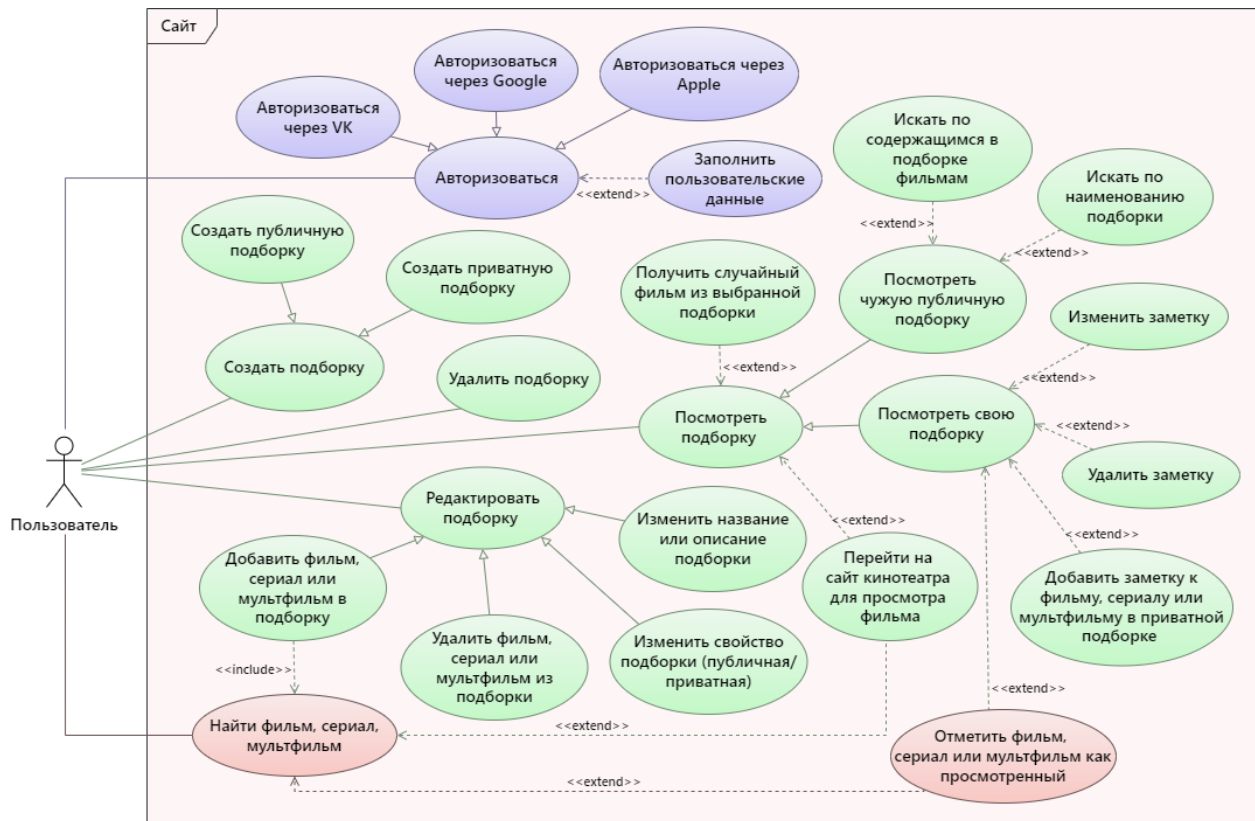


Рисунок 1 - Диаграмма вариантов использования

## 4.2 Диаграмма классов

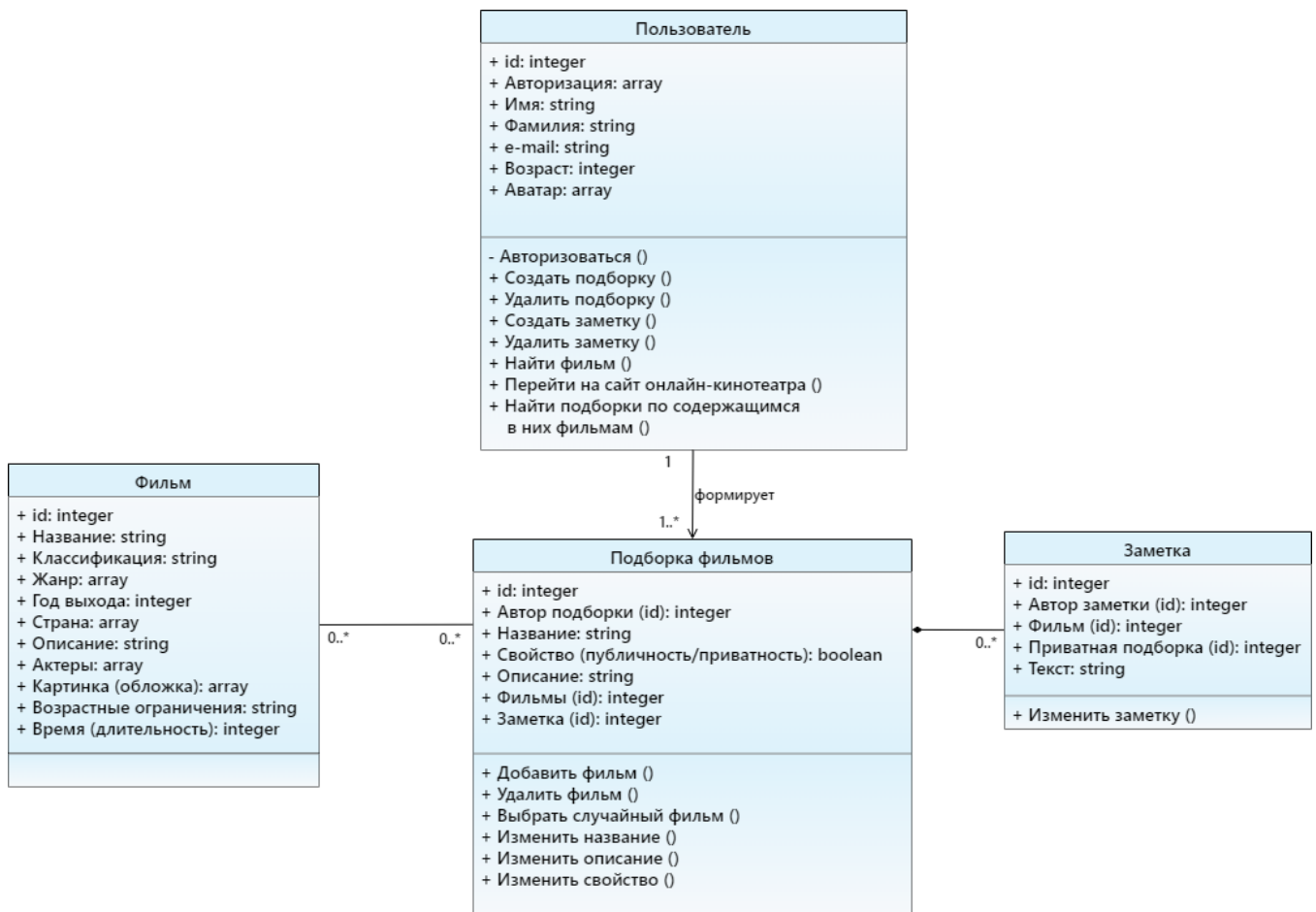


Рисунок 2 - Диаграмма классов

### 4.3 ER-диаграмма

Структура и связи сущностей, разрабатываемой системы, отображены на ег-диаграмме (рисунок 3).

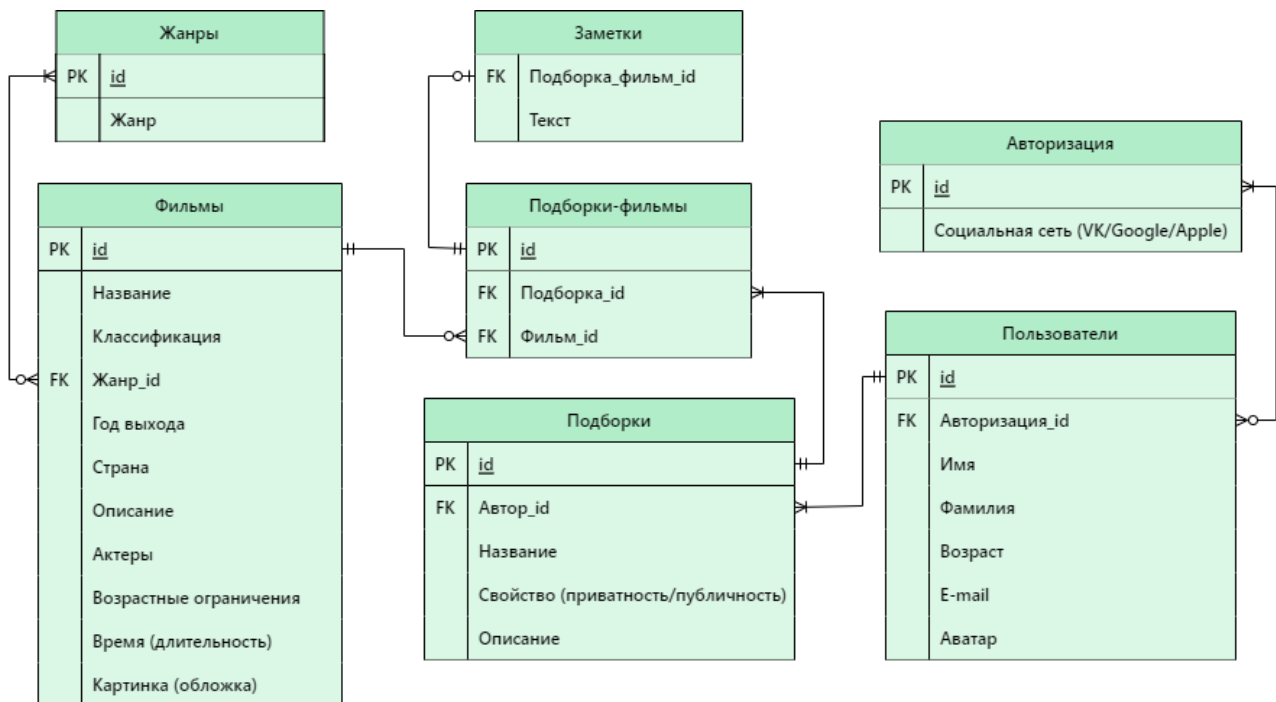


Рисунок 3 - ER-диаграмма

Хранение данных организовано следующим образом:

#### Users (Пользователи)

Поле (атрибут)	Тип	Описание поля
id	integer	id пользователя (заполняется автоматически)
authorization_id	array	id авторизации
name	string	имя пользователя
surname	string	фамилия пользователя
age	integer	возраст
email	string	адрес электронной почты
photo	array	аватар пользователя

#### authorization (Авторизация)

Поле (атрибут)	Тип	Описание поля
id	integer	id авторизации (заполняется автоматически)
social_network	string	вид авторизации: VK, Google, Apple

#### movie\_list (Подборки)

Поле (атрибут)	Тип	Описание поля
id	integer	id подборки (заполняется автоматически)
author_id	integer	id пользователя, создавшего подборку
list_name	string	наименование подборки
property	boolean	свойство подборки: - приватная (true) - публичная (false) по умолчанию приватная
description	string	описание подборки

#### movies (Фильмы)

Поле (атрибут)	Тип	Описание поля
id	integer	id фильма, сериала или мультфильма (заполняется автоматически)
movie_title	string	название фильма
classification	string	классификация: фильм, сериал, мультфильм
genre	string	жанр
release_year	integer	год производства

country	array	страна
description	string	описание фильма
actors	array	актеры
age_restrictions	string	возрастная классификация
movie_duration	integer	продолжительность фильма в минутах
picture	array	обложка

#### genres (Жанры)

Поле (атрибут)	Тип	Описание поля
id	integer	id жанра (заполняется автоматически)
genre	string	жанр

#### notes (Заметки)

Поле (атрибут)	Тип	Описание поля
list_movie_id	integer	id сочетания подборок и фильмов, сериалов, мультфильмов
text	string	текст заметки

#### list\_movies (Подборки-фильмы)

Поле (атрибут)	Тип	Описание поля
id	integer	id сочетания подборок и фильмов, сериалов, мультфильмов
list_id	integer	id подборки
movie_id	integer	id фильма, сериала или мультфильма

## Приложения

### Приложение 1. Результаты интервью

Ссылка на интервью:

<https://docs.google.com/document/d/1j6JvirJDwFxxi2vPp4Yr4xYbJcrh92cHThM1YLLMiE/edit>

### Приложение 2. Описание вариантов использования

**UC-01** Посмотреть подборку

**Краткое описание ВИ:** Пользователь выбирает и открывает свою (приватную/публичную) или чужую публичную подборку. Смотрит содержащиеся в ней фильмы.

**Действующие лица:** Пользователь

**Предусловие:** Пользователь авторизовался. Подборка создана.

Основной поток:

1. Пользователь выбирает для просмотра подборку фильмов.
2. Пользователь открывает подборку.
3. Если подборка не пустая, то переходим на следующий шаг.
4. Пользователь читает описание подборки и просматривает содержащиеся в подборке фильмы.
5. Вариант использования завершает свою работу.

**Альтернативный поток:**

**3a.** Подборка не содержит ни одного фильма.

1. Пользователь видит пустую подборку.
2. Управление переходит на шаг 5.

**Постусловие:** В случае успешного выполнения основного потока, пользователь посмотрел название, описание подборки и какие фильмы содержатся в подборке.

**Результат:** Если ВИ выполнен успешно, содержимое подборки просмотрено пользователем.

**УС-02** Добавить заметку к фильму, сериалу или мультфильму в приватной подборке

**Краткое описание ВИ:** Пользователь добавляет заметку к фильму в приватной подборке.

**Действующие лица:** Пользователь

**Предусловие:** Пользователь открыл для просмотра приватную подборку. Подборка содержит фильм.

**Основной поток:**

1. Пользователь выбирает фильм в приватной подборке.
2. Пользователь выбирает опцию “Добавить заметку”.
3. Система отображает текстовое поле для заметки.
4. Пользователь пишет заметку в текстовое поле.
5. Пользователь добавляет написанную заметку к фильму.
6. Система отображает, что у фильма есть заметка.
7. Вариант использования завершает свою работу.

**Постусловие:** У фильма в приватной подборке отображается заметка.

**Результат:** Если ВИ выполнен успешно, к фильму добавлена заметка.

### Приложение 3. Описание программного интерфейса

Ссылка на swagger-документацию:

<https://app.swaggerhub.com/apis-docs/Rikki-tikki-tavi/OnlineCinema/1.0.0>

#### 1. Кто будет взаимодействовать с API

- Система онлайн-кинотеатра
- Пользователи

#### 2. Методы и действия API

- 1) POST/movie\_list  
Добавить подборку.
- 2) DELETE/movie\_list/{listId}  
Удалить подборку.
- 3) PUT/movie\_list/{listId}

Изменить название, описание или свойство подборки.

4) GET/list\_movies/{listId}

Посмотреть подборку.

5) POST/list\_movies/{listId}/movies/{movieId}

Добавить фильм в подборку.

6) DELETE/list\_movies/{listId}/movies/{movieId}

Удалить фильм из подборки.

7) GET/list\_movies/{listId}/movies

Получить случайный фильм из подборки.

8) POST/notes

Добавить заметку к фильму в подборке.

9) GET/notes/{noteId}

Посмотреть заметку к фильму в подборке.

10) DELETE/notes/{noteId}

Удалить заметку к фильму в подборке.

11) PUT/notes/{noteId}

Изменить заметку к фильму в подборке.

12) GET/list\_movies/{movieId1}/movie\_list

Получить список подборок, в которых есть выбранные фильмы (от 1 до 5 фильмов включительно).

### 3. К каким ресурсам будем обращаться

- таблица Подборки
- таблица Подборки-фильмы
- таблица Заметки

### 4. Какие параметры запроса будут передаваться в каждый из методов?

- POST/movie\_list

Добавить подборку.

1) Запрос

- a. author\_id
- b. list\_name
- c. property
- d. description

2) Ответ

- a. id



- DELETE/movie\_list/{listId}  
Удалить подборку.
  - 1) Запрос
    - a. id
  - 2) Ответ
    - a. 200 OK
  
- PUT/movie\_list/{listId}  
Изменить название, описание или свойство подборки.
  - 1) Запрос
    - a. id
    - b. author\_id
    - c. list\_name
    - d. property
    - e. description
  - 2) Ответ
    - a. 200 OK
  
- GET/list\_movies/{listId}  
Посмотреть подборку.
  - 1) Запрос
    - a. list\_id
  - 2) Ответ
    - a. [{id, list\_id, movie\_id}]
  
- POST/list\_movies/{listId}/movies/{movieId}  
Добавить фильм в подборку.
  - 1) Запрос
    - a. list\_id
    - b. movie\_id
  - 2) Ответ
    - a. 200 OK
  
- DELETE/list\_movies{listId}/movies/{movieId}  
Удалить фильм из подборки.
  - 1) Запрос
    - a. list\_id

- b. movie\_id
- 2) Ответ
  - a. 200 OK
- GET/list\_movies/{list\_id}/movies  
Получить случайный фильм из подборки.
  - 1) Запрос
    - a. list\_id
    - b. random\_choice
  - 2) Ответ
    - a. list\_id
    - b. movie\_id
- POST/notes  
Добавить заметку к фильму в подборке.
  - 1) Запрос
    - a. text
    - b. list\_id
    - c. movie\_id
  - 2) Ответ
    - a. 200 OK
- GET/notes/{noteId}  
Посмотреть заметку к фильму в подборке.
  - 1) Запрос
    - a. list\_movie\_id
  - 2) Ответ
    - a. list\_movie\_id
    - b. text
- DELETE/notes/{noteId}  
Удалить заметку к фильму в подборке.
  - 1) Запрос
    - a. list\_movie\_id
  - 2) Ответ
    - a. 200 OK

- PUT/notes/{noteId}

Изменить заметку к фильму в подборке.

1) Запрос

- a. list\_movie\_id
- b. text

2) Ответ

- a. 200 OK

- GET/list\_movies/{movie\_id\_1}/movie\_list

Получить список подборок, в которых есть выбранные фильмы (от 1 до 5 фильмов включительно).

1) Запрос

- a. movieId1
- b. movieId2
- c. movieId3
- d. movieId4
- e. movieId5

2) Ответ

- a. [{list\_id}]