

Designing a REST API

1. Описание сущностей

В библиотечном каталоге у нас будут следующие сущности:

- **Book (Книга)**
 - **id** (уникальный идентификатор книги)
 - **title** (название книги)
 - **authorId** (идентификатор автора книги)
 - **categoryId** (идентификатор категории книги)
 - **publishedDate** (дата публикации книги)
 - **isbn** (международный стандартный книжный номер)
 - **availability** (доступность книги - в наличии/выписана)
- **Author (Автор)**
 - **id** (уникальный идентификатор автора)
 - **name** (имя автора)
 - **biography** (краткая биография автора)
 - **birthDate** (дата рождения автора)
 - **books** (список книг, написанных автором)
- **Category (Категория)**
 - **id** (уникальный идентификатор категории)
 - **name** (название категории)
 - **description** (описание категории)
 - **books** (список книг в категории)

2. Описание операций с функциями API

API должен поддерживать следующие операции:

Книги (Books):

- Получить список всех книг (с поддержкой фильтрации и пагинации)
- Получить информацию о конкретной книге по **id**
- Добавить новую книгу

- Обновить информацию о книге
- Удалить книгу по id

Авторы (Authors):

- Получить список всех авторов (с поддержкой фильтрации и пагинации)
- Получить информацию о конкретном авторе по id
- Добавить нового автора
- Обновить информацию об авторе
- Удалить автора по id

Категории (Categories):

- Получить список всех категорий (с поддержкой фильтрации и пагинации)
- Получить информацию о конкретной категории по id
- Добавить новую категорию
- Обновить информацию о категории
- Удалить категорию по id

3. Проектирование REST API

Основные конечные точки (endpoints):

1. Books

- GET /api/books - получить список всех книг
- GET /api/books/{id} - получить информацию о книге по id
- POST /api/books - добавить новую книгу
- PUT /api/books/{id} - обновить информацию о книге
- DELETE /api/books/{id} - удалить книгу по id

2. Authors

- GET /api/authors - получить список всех авторов
- GET /api/authors/{id} - получить информацию об авторе по id
- POST /api/authors - добавить нового автора
- PUT /api/authors/{id} - обновить информацию об авторе

- **DELETE /api/authors/{id}** - удалить автора по id

3. Categories

- **GET /api/categories** - получить список всех категорий
- **GET /api/categories/{id}** - получить информацию о категории по id
- **POST /api/categories** - добавить новую категорию
- **PUT /api/categories/{id}** - обновить информацию о категории
- **DELETE /api/categories/{id}** - удалить категорию по id

Фильтрация и пагинация:

- **Фильтрация будет поддерживаться через параметры запроса, например:**
 - **GET /api/books?authorId=1&categoryId=2&availability=true** - фильтрация книг по автору, категории и доступности.
- **Пагинация будет поддерживаться через параметры запроса page и size:**
 - **GET /api/books?page=1&size=10** - получить первую страницу, содержащую 10 книг.

Кеширование:

- **Кеширование будет применяться для запросов GET к конечным точкам, чтобы ускорить доступ к часто запрашиваемым данным.**
- **В заголовках ответа будет добавляться информация о времени жизни кеша (Cache-Control: max-age=3600), чтобы данные обновлялись каждый час.**

Аутентификация:

- **Для доступа к API будет использоваться аутентификация через токены (например, JWT - JSON Web Tokens).**
- **Токен будет передаваться в заголовке авторизации (Authorization: Bearer {token}).**

Обработка ошибок:

- **Все ошибки будут возвращать соответствующие коды состояния HTTP:**
 - **400 Bad Request** - неверный запрос (например, отсутствует обязательное поле).
 - **401 Unauthorized** - неавторизованный доступ.
 - **404 Not Found** - ресурс не найден.
 - **500 Internal Server Error** - ошибка сервера.

Применение модели зрелости Ричардсона с HATEOAS (7 баллов)

- **Уровень 0: Использование HTTP.**
- **Уровень 1: Конечные точки представляют ресурсы.**
- **Уровень 2: Использование стандартных HTTP-методов.**
- **Уровень 3: Реализация HATEOAS: ответы содержат ссылки (links) для перехода к другим связанным ресурсам и действиям.**
Например, в ответе на GET /api/books/{id} будет ссылка на автора книги, ссылку на категорию и ссылки для обновления или удаления книги.