

Лабораторная работа: разработка RestAPI для электронной библиотеки.

Описание работы:

Вы должны получить возможность получать любую информацию о книге, как и саму книгу. Либо же информация об авторе.

Книги и авторы не должны повторяться, для этого нужно создать условие, если это художественное произведение переведенное с другого языка, то можно загружать книги разных издательств, если это учебник то в зависимости от переиздания. Базу данных может заполнять только администратор.

Так же понадобится сделать поиск по названию книги, жанру или автору.

Подход для разработки представлений выбираете самостоятельно.

Требования к выполняемой работе

1. Общие положения

- Разработать систему управления электронной библиотекой, которая будет позволять пользователям получать актуальную информацию о книгах и авторах, а администратору — управлять базой данных.

2. Функциональные требования

2.1 Получение информации о книге:

- Пользователь должен иметь возможность найти информацию по названию книги, жанру или автору.

- Каждый экземпляр книги должен содержать:

1. Название книги (строка, максимум 100 символов).
2. Автор книги (внешний ключ на таблицу авторов).
3. Год выпуска (целое число от 1000 до 9999).
4. Жанр книги (строка, максимум 100 символов).
5. Категория книги (строка, максимум 100 символов).
6. Издательство (строка, максимум 100 символов).

7. Изображение обложки книги (изображение).
8. Файл с текстом книги (файл).

при этом записи должны быть уникальными по сочетанию полей "Название книги", "Автор", "Год выпуска", "Издательство".

2.2 Добавление книг:

- Администратор должен иметь возможность добавлять новые книги в базу. При добавлении следует учитывать:

- Если книга — художественное произведение, допускается добавление различных изданий от разных издательств.

- Если книга — учебник, добавление новых экземпляров возможно только в зависимости от переиздания (издатель, год издания).

2.3 Уникальность записей:

- Каждая книга и автор должны быть уникальными в базе данных. Для этого необходимо реализовать валидацию при добавлении книги.

2.4 Информация об авторах:

- Должна быть возможность получить информацию об авторах, включая биографию, список произведений и другие значимые данные.

3. Нефункциональные требования

3.1 Структура и оформление кода:

- Код должен быть чистым, структурированным и соблюдать общепринятые стандарты программирования.

- Использовать принципы ООП для организации кода.

- Каждый класс и метод должен содержать комментарии, объясняющие их функциональность.

3.2 Версионный контроль:

- Все работы должны быть загружены в репозиторий на GitHub.

- Репозиторий должен содержать файл README с описанием проекта, инструкциями по запуску и использованию системы, описанием структуры кода.

3.3 Тестирование:

- Необходимо провести тестирование разработанной системы и подготовить отчет о проведенных тестах.