**Техническое задание на разработку бизнес-модели**

**Описание проекта**

Проект представляет собой систему управления библиотекой. В основе системы лежат две главные сущности: "Книги" и "Авторы". Каждая книга может быть написана только одним автором, но один автор может написать множество книг.

**Основные сущности и их атрибуты**

**1. Сущность "Автор" (Author)**

**Атрибуты:**

* id (уникальный идентификатор автора)
* first\_name (имя автора)
* last\_name (фамилия автора)
* birth\_date (дата рождения)

**2. Сущность "Книга" (Book)**

**Атрибуты:**

* id (уникальный идентификатор книги)
* name (название книги)
* release\_date (год издания)
* author\_id (идентификатор автора, связывающий книгу с автором)

**Функциональные требования**

1. **Управление авторами**
   * Добавление нового автора с указанием всех необходимых атрибутов.
   * Редактирование информации об авторе.
   * Удаление автора, если у него нет связанных книг.
2. **Управление книгами**
   * Добавление новой книги с указанием всех необходимых атрибутов.
   * Редактирование информации о книге.
   * Удаление книги.
   * Связывание книги с автором через author\_id.
3. **Отображение информации**
   * Просмотр списка всех авторов с основными атрибутами.
   * Просмотр списка всех книг с основными атрибутами.
   * Просмотр детальной информации об авторе, включая список всех его книг.

**Требования к API**

1. **API для авторов:**
   * **POST /authors** - добавление нового автора.
   * **GET /authors** - получение списка всех авторов.
   * **GET /authors/{id}** - получение детальной информации об авторе.
   * **PUT /authors/{id}** - обновление информации об авторе.
   * **DELETE /authors/{id}** - удаление автора.
2. **API для книг:**
   * **POST /books** - добавление новой книги.
   * **GET /books** - получение списка всех книг.
   * **GET /books/{id}** - получение детальной информации о книге.
   * **PUT /books/{id}** - обновление информации о книге.
   * **DELETE /books/{id}** - удаление книги.

**Технические детали**

1. **Язык программирования**: Java
2. **Фреймворк**: Spring Boot для разработки REST API
3. **База данных**: PostgreSQL