**Задание:** Написать backend на языке Python c использованием SQL базы данных для таблицы архива документов и документооборота. Создать базу данных и таблицы на сервере SQL.

**Описание:**

**Создание backend на языке Python с использованием SQL базы данных для таблицы архива документов**

1. **Введение**: Настоящее техническое задание описывает разработку backend-части системы для управления таблицей архива документов. Backend будет создан на языке Python с использованием SQL базы данных. Система должна предоставлять функциональность добавления, редактирования, поиска документов, а также возможность создания связей между документами. Доступ к системе будет осуществляться через авторизированный вход с разделением прав доступа на уровни "Админ" и "Гость".
2. **Требования:**

2.1. База данных:

* + Использование SQL базы данных для хранения информации о документах.
  + Создание таблицы "Список документов" с следующими столбцами:
    - Название: текстовое поле для указания названия документа.
    - Автор: текстовое поле для указания автора документа.
    - Дата создания: поле для хранения даты создания документа.
    - Цвет: текстовое поле для указания цвета документа.
    - Теги: текстовое поле для указания тегов, связанных с документом.
    - Формат файла: текстовое поле для указания формата файла документа.

2.2. Функциональность:

* Добавление документов:
  + Реализация метода для добавления новых документов в таблицу.
  + Валидация входных данных перед добавлением в базу данных.
* Редактирование документов:
  + Реализация метода для редактирования существующих документов в таблице.
  + Поиск документов по уникальному идентификатору и обновление соответствующих полей.
* Поиск документов:
  + Реализация метода для поиска документов по различным критериям, таким как название, автор, теги и т. д.
  + Возврат результатов поиска в удобном формате (например, в виде списка документов).
* Создание связей между документами:
  + Реализация метода для установки связей между документами, позволяющих указать, что один документ связан с другим.
  + Хранение информации о связях в базе данных.
* Журнал истории добавленных или редактированных файлов:
  + Реализация функциональности для сохранения истории добавления и редактирования документов.
  + Хранение информации о каждом действии (дата, время, пользователь, документ) в базе данных.
* Авторизированный вход:
  + Разработка механизма авторизации с использованием логина и пароля.
  + Разделение прав доступа на уровни "Админ" и "Гость".

2.3. Интерфейс:

* Разработка API для взаимодействия с backend-частью системы.
* API должен предоставлять методы для выполнения всех описанных функций.
* Возвращаемые данные должны быть в удобном формате (например, JSON).

1. Технические требования:
   * Язык программирования: Python.
   * Использование фреймворка(библиотек) для создания backend, такого как Flask или Django.
   * Использование SQL базы данных, такой как PostgreSQL или MySQL.
   * Реализация CRUD (Create, Read, Update, Delete) операций для работы с таблицей документов.
   * Разделение логики работы с базой данных и бизнес-логики.
2. **План работ:**
   * Создание SQL базы данных и таблицы "Список документов".
   * Разработка методов для добавления и редактирования документов.
   * Разработка методов для поиска документов по различным критериям.
   * Реализация функциональности создания связей между документами.
   * Разработка функциональности журнала истории добавленных или редактированных файлов.
   * Разработка механизма авторизации с разделением прав доступа.
   * Разработка API для взаимодействия с backend-частью системы.
   * Тестирование и отладка функциональности.
3. **Ожидаемые результаты:**
   * Backend-часть системы для управления таблицей архива документов, реализованная на языке Python.
   * SQL база данных с таблицей "Список документов".
   * Функциональность добавления, редактирования и поиска документов.
   * Возможность создания связей между документами.
   * Функциональность журнала истории добавленных или редактированных файлов.
   * Механизм авторизации с разделением прав доступа.
   * API для взаимодействия с backend-частью системы.