**Documentation center**

Разработать хранилище документов с возможностью загрузки файлов (upload), хранения, последующей выгрузки (download) и удаления.

Файлы и каталоги должны иметь дополнительное текстовое описание (кроме наименования), которое задается при загрузке файла в хранилище.

**Хранилище**

Хранилище должно быть представлено в виде иерархической структуры – каталоги и файлы внутри каталогов. Для простоты и скорости реализации допустимо реализовать лишь один уровень вложенности, т.е без вложенных каталогов.

Требуется ограничить загрузку файлов по расширениям. Можно взять файлы MS Office – достаточно минимум двух различных типов файлов, например \*.doc и \*.xls.

Возможные опции хранения файлов и их описаний: либо файловая система веб-сервера, либо СУБД сервер. Нужно осознанно выбрать подходящий тип хранилища исходя из постановки задачи.

Для простоты решения не требуется реализация редактирования или перемещения документов в каталогах. Достаточно лишь добавления и удаления.

**UI**

Структура файлов и каталогов должна быть представлена для пользователя в удобном виде. Нужно выбрать конкретное подходящее UI представление исходя из постановки задачи.

Хранилище должно позволять:

1. Создавать каталоги, указав наименование и описание.
2. 3агружать файлы с помощью веб браузера, указав наименование и описание.
3. Просматривать файлы и каталоги включая описание.
4. Выгружать файлы на клиент с помощью веб браузера.
5. Удалять файлы и каталоги.

Элементы хранилища должны иметь графические иконки. Иконки для различных типов файлов и каталогов должны отличаться.

**Технические требования**

Тип приложения: SPA (Single page application) – приложение не должно обновлять страницу полностью при работе. Авторизация не требуется.

Тип СУБД (если требуется): SQL Server, версия произвольная до 2016 включительно.

Серверные технологии: .NET или .NET Core, ASP.NET 5 или ASP.NET Core. Предпочтительно (но не обязательно) использование .NET Core. Сервер IIS не является обязательным к использованию, но допустим. Достаточным будет использование любого development сервера.

Клиентские технологии: любые, позволяющие реализовать требуемый функционал. Можно использовать серверный рендеринг на Razor.

Среда разработки: Visual Studio либо Visual Studio Code.

**Установка и запуск**

Предполагается, что СУБД сервер уже установлен или входит в состав приложения. Можно использовать любой движок.

Решение должно либо производить установку и настройку среды работы полностью самостоятельно, либо содержать краткую инструкцию по установке, понятную разработчику.