This API has been designed in Visual Studio 2017, net Core 2.1 and has been tested in Postman.

**CRUD Operations**

* **To upload the Document POST**

~URL/Document/Upload/

To choose the file - Go to Form-data in Body tab. Choose Key as file and then click on select file to upload the fie

File will be uploaded successfully in the Blob.

Example: https://navdalstorage.blob.core.windows.net/navdal- container/CleverBooks%20Geometry%20Markers%20Test.pdf

And entry will be created in SQL Table with the Id, DocName, Location, FileSize, Position.

Position will be zero and

* **To get list of the Documents: GET**

[~URL/Document/GetAll](https://localhost:44383/Document/GetAll)

This method will give the results in Alphabetical order when position =0.

When it is reordered (assigned specific positions )

Expected Results:

 "name": "CleverBooks Geometry Markers Test.pdf",

 "location": "https://navdalstorage.blob.core.windows.net/navdal- container/CleverBooks%20Geometry%20Markers%20Test.pdf",

 "fileSize": 568791,

 "position": 1

Document has been stored in Blob and Meta data is stored in SQL Table DocumentList.

* **To download the Document: GET**

[~URL /Document/Download/](https://localhost:44383/Document/Downlaod/) followed by Doc name (fully or partial)

This method will search filename and display the document if found else display the “Not found ”message

User can provide few characters of the filename or the complete name.

Example :

[~URL /Document/Download/](https://localhost:44383/Document/Downlaod/)CleverBooks Geometry Markers Test.pdf

Or

[~URL /Document/Download/](https://localhost:44383/Document/Downlaod/)Markers

* **To reorder the documents: PATCH**

[~URL /Document/Reload](https://localhost:44383/Document/Reload)

This method expects two fields in the body Name and position

API will find the doc with its name and will update the position of that doc.

If it does not find the document, it will show an error message.

It will bulk update the position of all the docs mentioned in the body of the Postman.

Example of the bulk update

{

        "name": "CleverBooks Geometry Markers Test.pdf",

        "position": 4

    },

    {

        "name": "Engineering.pdf",

        "position": 5

    },

    {

        "name": "Travel Africa with Augmented Reality\_MARKER\_FREE.pdf",

        "position": 6

    }

]

**To DELETE the document: DELETE**

[~URL /Document/](https://localhost:44383/Document/Reload) Followed by Doc name.

Document name should exactly match stored in the DB.

It it’s found then the entry is deleted from the DB and from the blob.

Example

[~URL /Document/](https://localhost:44383/Document/Reload) CleverBooks Geometry Markers Test.pdf

* **Packages Installed**

Below Entity Framework packages which are compatible with.net core 2.1

Install-Package Microsoft.EntityFrameworkCore.SqlServer.Design -Version 1.1.6

Install-Package Microsoft.EntityFrameworkCore.SqlServer -Version 2.1.1

Install-Package Microsoft.EntityFrameworkCore.Tools -Version 2.1.1

* **SQL Database**

Query to create Table (DocumentLog) in the Database( DocumentMgt)

in (localdb)[\\mssqllocaldb](file:///\\mssqllocaldb)

USE [DocumentMgt]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[DocumentList] (

[DocumentID] INT IDENTITY (1, 1) NOT NULL,

[Name] NVARCHAR (MAX) NULL,

[FilePath] NVARCHAR (MAX) NULL,

[FileSize] BIGINT NULL,

[UploadedDateTime] DATETIME NULL,

[Position ] INT NULL

);