Add this in the App.Config/ any config file

<add key="webUrl" value="https://\*\*\*\*.sharepoint.com/sites/\*\*\*\*" />

<add key="UserName" value="\*\*\*\*@\*\*\*\*.onmicrosoft.com" />

<add key="Password" value="\*\*\*\*\*" />

Initialize the stub after adding the reference:

SharePointDLL.DocLibOperations operations = new DocLibOperations(userName:"xxx",webUrl:"xxx",password:"xxx");

## Methods

The custom SharePoint dll used mainly for Document management with respect to CRUD operation to perform daily activities. Mentioned below Methods used to perform the task,

1. Add Document
2. Search Document
3. Delete Document
4. Update Document

## Add Document

The Add Document method which allows to add any document type to predefined Document Library in a SharePoint 2013 Site. While file upload, the associated Metadata values also be assigned with respect to file. When file uploaded using either Windows or Web Application, the CSOM code snippet performs upload operation using Binary Stream objects. The add document method will accept input parameter as below,

var addDocJson = operation.AddDocument(ddpLibraries.SelectedValue.ToString(), txtFilePath.Text, MetadataProp, rootFolder);

* ddpLibraries.SelectedValue.ToString() – Library Name
* txtFilePath.Text – Local file path
* MetadataProp – Dictionary object Key (Internal Metadata Name) & Value (Metadata Value)
* rootFolder – Root folder name (default blank value)

And output parameter will return JSON object which contains mentioned below details,

[{"DocumentName":"EX0515.xlsx","ID":"5","Version":"1.0","CreatedBy":"xxx","CreatedDate":"8/13/2019 7:12:04 AM","ModifiedBy":"xxx","ModifiedDate":"8/13/2019 7:12:04 AM","Pathxxx","Code":"CD00023232","BillingName":"Cosmo Products","Billing\_x0020\_Address":"Pune","Tax\_x0020\_Number":"DKJJD3092384"}]

## Search Document

Search the uploaded documents in SharePoint Document Library using Search Method.

Two types of Search Documents

### Site level Search

To Search the document, the Search Method accepts following input parameters with Search by,

var SearchedItems = operation.SearchSiteDocument(txtSearchKeyword.Text.ToString().Trim());

* txtSearchKeyword.Text.ToString().Trim()) – Keyword Begin with Search

And output parameter will return the JSON object which contains mentioned below details,

{[

{

"Cells": [

{

"Key": "Rank",

"Value": "16.8894462585449",

"ValueType": "Edm.Double"

},

{

"Key": "DocId",

"Value": "17597594213045",

"ValueType": "Edm.Int64"

},

{

"Key": "Filename",

"Value": "CC Supplies.pdf",

"ValueType": "Edm.String"

},

{

"Key": "Path",

"Value": "https://tsinfo.sharepoint.com/sites/Guru/Customer/CC Supplies.pdf",

"ValueType": "Edm.String"

},

{

"Key": "ModifiedOWSDate",

"Value": "2019-08-11T13:31:57Z",

"ValueType": "Edm.String"

},

{

"Key": "ModifiedById",

"Value": "xxx",

"ValueType": "Edm.String"

},

{

"Key": "ListItemID",

"Value": "1",

"ValueType": "Edm.String"

},

{

"Key": "CreatedOWSDate",

"Value": "2019-08-11T13:31:57Z",

"ValueType": "Edm.String"

},

{

"Key": "AuthorOWSUser",

"Value": "xxx",

"ValueType": "Edm.String"

},

{

"Key": "OriginalPath",

"Value": "xxx",

"ValueType": "Edm.String"

},

{

"Key": "\_ranking\_features\_",

"Value": null,

"ValueType": "Null"

},

{

"Key": "PartitionId",

"Value": "0fa8312a-b53b-4040-8203-f241cdd7faaa",

"ValueType": "Edm.Guid"

},

{

"Key": "UrlZone",

"Value": "0",

"ValueType": "Edm.Int32"

},

{

"Key": "Culture",

"Value": "en-US",

"ValueType": "Edm.String"

},

{

"Key": "ResultTypeId",

"Value": "0",

"ValueType": "Edm.Int32"

},

{

"Key": "RenderTemplateId",

"Value": "~sitecollection/\_catalogs/masterpage/Display Templates/Search/Item\_Default.js",

"ValueType": "Edm.String"

}

]

}

]}

### Library level Search

To Search the document, the Search Method accepts following input parameters with Search by,

var SearchedItems = operation.SearchLibDocument(txtLibKeywordSearch.Text.ToString().Trim(), ddpLibraries.SelectedValue.ToString(), MetadataProp);

* txtLibKeywordSearch.Text.ToString().Trim() - Keyword Begin with Search
* ddpLibraries.SelectedValue.ToString() – Library Name
* MetadataProp - Dictionary object Key (Internal Metadata Name) & Value (Metadata Value)

And output parameter will return the JSON object which contains mentioned below details,

[{"DocumentName":"EC Estate Agents.pdf","ID":"4","Version":"1.0","CreatedBy":"xxx","CreatedDate":"8/13/2019 6:08:51 AM","ModifiedBy":"xxx","ModifiedDate":"8/13/2019 6:08:51 AM","Path":"/sites/Guru/Customer/New Customer/EC Estate Agents.pdf","Code":"1111111","BillingName":"ParaCompany","Billing\_x0020\_Address":"xx","Tax\_x0020\_Number":"12233"}]

## Delete Document

Delete the document from Document Library using Delete Method by input param Document ID or Document Name

var deleteDocBool = operation.DeleteDocument(LibName, fileName);

* LibName- Library Name
* filename – Document ID

And Output parameter deleteDocBool is Boolean value.

## Update Document

To update existing SharePoint document in a Document Library, the input parameter will accept following parameter values

var updateDocJson = operation.UpdateDocument(LibName, fileName, MetadataProp);

* LibName – Library Name
* filename – Document ID
* MetadataProp - Dictionary object Key (Internal Metadata Name) & Value (Metadata Value)

And Output parameter will return JSON object with Document ID and updated values.

[{"DocumentName":"EC Estate Agents.pdf","ID":"4","Version":"1.0","CreatedBy":"xxx","CreatedDate":"8/13/2019 6:08:51 AM","ModifiedBy":"xxx","ModifiedDate":"8/13/2019 6:08:51 AM","Path":"/sites/Guru/Customer/New Customer/EC Estate Agents.pdf","Code":"1111111","BillingName":"ParaCompany","Billing\_x0020\_Address":"Bangalore","Tax\_x0020\_Number":"12233"}]