ASP(Classic) Example

1. Uploading Images, Videos and Files

> ⊘ **Caution**
>
> Among the following codes provided as the guide, file upload part is a **sample code** and has insufficient security.
>
> As for the file upload, please use the code used within your project and refer the following code to handle the system link.

> ⓘ TABS Upload5 was used for the file upload. You may use the upload component you have been using for the actual linking with ASP.

import.asp

```
<%@ CodePage=65001 Language=VBScript %>
<% Option Explicit %>
<%
    On Error Resume Next
    Dim UPLOAD_PATH
    UPLOAD_PATH = "C:\inetpub\wwwroot\upload"

    Dim Upload, fileName
    'Create an object to process the upload.
    Set Upload = Server.CreateObject("TABSUpload4.Upload")
    Upload.CodePage = 65001


    'Start the upload.
    Upload.Start "C:\TEMP"
    Upload.Save UPLOAD_PATH, False

    'Path of the stored file (except the path)
    fileName = Upload.Form("file").ShortSaveName

    'Return the result in json format
    Response.ContentType = "application/json"
    Response.write("{""uploadPath"":""/upload/" & fileName & """}")
%>
```

2. Importing HWP, MS Word and Excel Documents

> ⊘ **Caution**
>
> Among the following codes provided as the guide, file upload part is a **sample code** and has insufficient security.
>
> As for the file upload, please use the code used within your project and refer the following code to handle the system link.

> ⓘ TABS Upload5 was used for the file upload. You may use the upload component you have been using for the actual linking with ASP.

import.asp

```
<%@ CodePage=65001 Language=VBScript %>
<% Option Explicit %>
```

```
<%
    On Error Resume Next
    Dim CONVERTER, FONTS, WORK, UPLOAD_PATH
    CONVERTER = "C:\workspace\seimporter\sedocConverter\sedocConverter.exe"
    FONTS = "C:\workspace\seimporter\fonts"
    WORK = "C:\workspace\seimporter\tmp"
    UPLOAD_PATH = "C:\inetpub\wwwroot\upload"


    Dim filePath, outputPath, uuid, relativeOutputPath


    Dim Upload
    'Create an object to process the upload.
    Set Upload = Server.CreateObject("TABSUpload4.Upload")
    Upload.CodePage = 65001

    'Start the upload.
    Upload.Start "C:\TEMP"
    Upload.Save UPLOAD_PATH, False

    'Path of the stored file (except the path)
    filePath = Upload.Form("file").SaveName

    'Generate UUID (unique path)
    uuid = CreateGUID()

    outputPath = "C:\inetpub\wwwroot\output\" & uuid
    relativeOutputPath = "/output/" & uuid

    'Document conversion
    Dim wshShell, strCmd, result
    'Set wshShell = CreateObject( "WScript.Shell" )
    strCmd = CONVERTER & " -pz -f " & FONTS & " """ & filePath & """ " & outputPath & " " & WORK
    'result = wshShell.Run(strCmd, 0, true)
    result = Exec(strCmd, 1)

    If Not result = 0 Then
        Response.write "Error : " & result
    Else
        'Delete the original document once the conversion is completed.
        DeleteExistFile(filePath)
    End If

    Set wshShell = nothing
    Set Upload = Nothing


    'Serialize document.pb file and transfer it
    Dim binText
    binText = ReadBinaryFile(outputPath & "\" & "document.pb")

    'Load and delete the pb file
    DeleteExistFile(outputPath & "\" & "document.pb")

    'Return the result in json format
    Response.ContentType = "application/json"
    Response.write("{""importPath"":""" & relativeOutputPath & """, ""serializedData"":" & binText & "}")

    Function Exec(c, t)
        Dim s, e : Set s = CreateObject("WScript.Shell") : Set e = s.Exec(c)
        Do While e.Status = 0
            Call s.Run("waitfor /t 1 OneSecond", 0, True)
            t = t - 1
            If 0 >= t Then
            Call s.Run("taskkill /t /f /pid " & e.ProcessId, 0, True)
            Exit Do
            End If
        Loop
        Set Exec = e
    End Function
```

```vbscript
'Generate UUID (unique path)
Function CreateGUID()
    Dim tmpTemp
    tmpTemp = Right(String(4,48) & Year(Now()),4)
    tmpTemp = tmpTemp & Right(String(4,48) & Month(Now()),2)
    tmpTemp = tmpTemp & Right(String(4,48) & Day(Now()),2)
    tmpTemp = tmpTemp & Right(String(4,48) & Hour(Now()),2)
    tmpTemp = tmpTemp & Right(String(4,48) & Minute(Now()),2)
    tmpTemp = tmpTemp & Right(String(4,48) & Second(Now()),2)
    CreateGUID = tmpTemp
End Function

Function DeleteExistFile(filePath)
    Dim fso, result
    Set fso = CreateObject("Scripting.FileSystemObject")
    If fso.FileExists(filePath) Then
        fso.DeleteFile(filePath) 'Delete the file if there is any.
        result = 1
    Else
        result = 0
    End If
    DeleteExistFile = result
End Function

Function ReadBinaryFile(FileName)
    Const adTypeBinary = 1
    Const adTypeText = 2

    'Create Stream object
    Dim BinaryStream
    Set BinaryStream = CreateObject("ADODB.Stream")
    Dim bin, str, cnt

    'Specify stream type - we want To get binary data.
    BinaryStream.Type = adTypeBinary

    'Open the stream
    BinaryStream.Open

    'Load the file data from disk To stream object
    BinaryStream.Position = 0 'Set the stream position to the start
    BinaryStream.LoadFromFile FileName

    cnt = 1
    Do
        bin = BinaryStream.Read(1024)

        if Not isNull(bin) then
            if cnt = 1 then
                str = str & BinaryToString(bin, 17)
            Else
                str = str & "," & BinaryToString(bin, 1)
            end if
            cnt = cnt + 1
        end if
    Loop While Not IsNull(bin)

    ReadBinaryFile = "[" & str & "]"
    BinaryStream.Close
    Set BinaryStream = Nothing
End Function

Function BinaryToString(Binary, startPosition)
        'Antonin Foller, http://www.motobit.com
        'Optimized version of a simple BinaryToString algorithm.

        Dim cl1, cl2, cl3, pl1, pl2, pl3
        Dim L
        cl1 = startPosition  '1 or 17
        cl2 = 1
```

```
            cl3 = 1
            L = LenB(Binary)

            Do While cl1<=L
              pl3 = pl3 & CStr(AscB(MidB(Binary,cl1,1)))
              If cl1 < L Then
                  pl3 = pl3 & ","
              End if
              cl1 = cl1 + 1
              cl3 = cl3 + 1
              If cl3>300 Then
                pl2 = pl2 & pl3
                pl3 = ""
                cl3 = 1
                cl2 = cl2 + 1
                If cl2>200 Then
                  pl1 = pl1 & pl2
                  pl2 = ""
                  cl2 = 1
                End If
              End If
            Loop
            BinaryToString = pl1 & pl2 & pl3
    End Function
%>
```

See also

- Java Spring Framework Example
- Java Servlet Example
- ASP.NET (C#) Example
- ASP(Classic) Example
- PHP Example
- PHP4 Example
- Django Example
- Ruby On Rails Example
- Wordpress plugin Example