

---

Question

[Yone Moreno](#) · Jan 25, 2021

## How could we retrieve an image from an external system using Ensemble?

Hello,

We would be grateful if you could help us,

Our aim is to develop a method which retrieves an image stored in an external system, using a REST Operation

To upload the image we have the following code:

```
/// Operacion para gestionar el Registro imagen identidad digital en Tarjeta Sanitari
a
Class Operaciones.REST.miSCS.miSCS Extends EnsLib.REST.Operation
{
Parameter INVOCATION = "Queue";

/// Subir la imagen del usuario
/// La documentación para crear este metodo es:
/// https://docs.intersystems.com/irisforhealthlatest/csp/docbook/Doc.View.cls?KEY=GN
ET_http#GNET_http_post_multipart
/// https://docs.intersystems.com/irisforhealthlatest/csp/docbook/Doc.View.cls?KEY=GN
ET_mime
Method SubirImagen(pRequest As Mensajes.Request.miSCS.SubirImagen, pResponse As Mensa
jes.Response.miSCS.SubirImagen) As %Library.Status
{
$$$LOGINFO("pRequest.imagen: "_pRequest.imagen) //Creamos Request y Response HTTP
Set httpRequest=##class(%Net.HttpRequest).%New()
set tResponse = ##class(%Net.HttpResponse).%New() // Generamos la parte que contiene
a las demás partes, la "raiz", RootMIMEPart
    Set RootMIMEPart=##class(%Net.MIMEPart).%New()

    //Creamos la primera "subparte" con el expediente
    Set ExpedienteMIMEPart=##class(%Net.MIMEPart).%New()
    Set contentdisp="form-data; name="expediente""
    Do ExpedienteMIMEPart.SetHeader("Content-Disposition",contentdisp)
    Set ExpedienteMIMEPart.Body=pRequest.expediente
    //Do ExpedienteMIMEPart.SetHeader("Content-
Type","text/plain") ; esto no se envia en POSTMAN

    //Asignamos la imagen
    //Esto si funciona (la guarda como jpg y al descargarla se ve)
    Set docStream=##class(%Stream.FileBinary).%New()
    //Set sc=docStream.LinkToFile("C:\InterSystems\avatar prueba.jpg") // Esto no
funciona (guarda el base64 directo, no el binario;
    // es decir, guarda lo mismo que enviamos desde POSTMAN)
```

```
//set docStream = pRequest.imagen

//Si ponemos pRequest.imagen como string, enviamos el base64 por postman y decodificamos aqui a binario si
//guarda la imagen como jpg
set binary = $system.Encryption.Base64Decode(pRequest.imagen)
do docStream.Write(binary)

// Instanciamos la segunda "subparte" con la imagen
Set ImagenMIMEPart=##class(%Net.MIMEPart).%New()
// Incluimos la imagen, el docStream, en la subparte MIME
Do ImagenMIMEPart.BodySet(docStream)
// Cabeceras
Set ImagenMIMEPart.ContentType = "image/jpeg"
Do ImagenMIMEPart.SetHeader("Content-Disposition","form-data; name=""imagen""; filename=""imagen.jpg"")

// Insertamos las dos subpartes en la raiz
Do RootMIMEPart.Parts.Insert(ExpedienteMIMEPart)
Do RootMIMEPart.Parts.Insert(ImagenMIMEPart) // Creamos el MIME writer
Set writer=##class(%Net.MIMEWriter).%New() // Preparamos para enviar el mensaje MIME a la peticion HTTP
Set httpRequest=##class(%Net.HttpRequest).%New()
Set status=writer.OutputToStream(httpRequest.EntityBody)
if $$$ISERR(status) {do $SYSTEM.Status.DisplayError(status) Quit} // Escribimos el contenido del mensaje MIME en la peticion HTTP
Set status=writer.WriteMIMEBody(RootMIMEPart)
if $$$ISERR(status) {do $SYSTEM.Status.DisplayError(status) Quit} // Creamos la URL
set url = ##class(Util.TablasMaestras).getValorMaestra("MISCS.PARAMETRIZACIONES","url")
set path = ##class(Util.TablasMaestras).getValorMaestra("MISCS.PARAMETRIZACIONES","path")
set servicio = ##class(Util.TablasMaestras).getValorMaestra("MISCS.PARAMETRIZACIONES","servicio")
set recurso = ##class(Util.TablasMaestras).getValorMaestra("MISCS.PARAMETRIZACIONES","subirImagen") set URL = "http://"_url_path_servicio_recurso

$$$LOGINFO("URL: "_URL)
//Escribimos el tipo de contenido que enviamos en la raiz
Set httpRequest.ContentType="multipart/form-data; boundary="_RootMIMEPart.Boundary
$$$LOGINFO("> httpRequest.ContentType: "_httpRequest.ContentType) //Enviamos al sistema externo
set tSC=httpRequest.Post(URL,0)
$$$LOGINFO("tSC: "_tSC) //Lanzamos excepcion si hubo error
if $$$ISERR(tSC){
$$$ThrowOnError(tSC)
}
//Obtenemos respuesta
set tResponse = httpRequest.HttpResponse //Leemos respuesta
if (tResponse.Data.AtEnd = 0) {
set linea = tResponse.Data.Read()
$$$LOGINFO("Linea: "_linea)
} //Rellenamos response
set pResponse = ##class(Mensajes.Response.miSCS.SubirImagen).%New()
set pResponse.resultado = 1
set pResponse.informacion = linea
//Traducimos codigo del resultado, a descripcion
```

```
//set pResponse.informacion = ##class(Util.TablasMaestras).getValorMaestra("MISCS.COD
IGOSTSI",linea) Quit pResponse
}

XData MessageMap
{
<MapItems>
  <MapItem MessageType="Mensajes.Request.miSCS.SubirImagen">
    <Method>SubirImagen</Method>
  </MapItem>
</MapItems>
}
```

We have written the following code to retrieve the image:

```
/// Operacion para gestionar el la imagen
Class Operaciones.REST.miSCS.miSCS Extends EnsLib.REST.Operation
{

/// Obtener la imagen del usuario
Method ConsultarImagen(pRequest As Mensajes.Request.miSCS.ConsultarImagen, pResponse
As Mensajes.Response.miSCS.ConsultarImagen) As %Library.Status
{
set httpRequest = ##class(%Net.HttpRequest).%New()
set tResponse = ##class(%Net.HttpResponse).%New()
set httpRequest.ContentType = "application/json"
set url          = ##class(Util.TablasMaestras).getValorMaestra("MISCS.PARAMETRIZACIONES"
,"url")
set path         = ##class(Util.TablasMaestras).getValorMaestra("MISCS.PARAMETRIZACIONES"
,"path")
set servicio     = ##class(Util.TablasMaestras).getValorMaestra("MISCS.PARAMETRIZACIONES"
,"servicio")
set recurso      = ##class(Util.TablasMaestras).getValorMaestra("MISCS.PARAMETRIZACIONES"
,"consultarImagen") set URL = "http://"_url_path_servicio_recurso

$$$LOGINFO("URL: "_URL) set tFormVarNames = "expediente"
set tData("expediente") = pRequest.expediente set tSC = ..Adapter.SendFormDataArray(.
tResponse,"GET",httpRequest,.tFormVarNames,.tData,URL)
$$$LOGINFO("tSC: "_tSC) if $$$ISERR(tSC){
$$$ThrowOnError(tSC)
}
if (tResponse.Data.AtEnd = 0) {
set linea = tResponse.Data.Read()
$$$LOGINFO("Linea: "_linea)
} set pResponse = ##class(Mensajes.Response.miSCS.ConsultarImagen).%New()
set pResponse.resultado = 1
set pResponse.informacion = linea Quit pResponse
}

XData MessageMap
```

---

```
{  
<MapItems>  
  <MapItem MessageType="Mensajes.Request.miSCS.ConsultarImagen">  
    <Method>ConsultarImagen</Method>  
  </MapItem>  
</MapItems>  
}
```

1 To upload an image we do the following

- We choose a JPG and open it with Notepad, we observe: `ÿÿÿà JFIF...`

- We encode it to base64 using the following conversor: <https://www.motobit.com/util/base64-decoder-encoder.asp>, the result is: `/9j/4CAQSk...`

-2 If we send the request to our ESB REST Service using POSTMAN to the patient 9999

The screenshot shows a Postman interface for a REST client. The request is a POST to `http://localhost:19621/sanidad/scs/test/miSCS/v01r00/subirlimagen`. The Headers tab is active, showing a 'request' header with a value containing a base64 encoded image. The Body tab is also active, showing a JSON response in 'Pretty' format:

```
1 {  
2   "resultado": true,  
3   "informacion": "Acción Correcta",  
4   "error": null  
5 }
```

The external system replies that it has been successfully uploaded

If we get this image directly from the external system endpoint, we get it as a correct JPG:

The screenshot shows a REST client interface for a GET request to the URL `http://tsws-pre.scs-salud.org/servicios/imagenidscs/consultarImagen`. The request has a parameter `expediente` with the value `9999`. The response status is `200 OK` with a time of `119 ms` and a size of `9.36 KB`. The response body is displayed in 'Raw' mode, showing a corrupted image file with the filename `25 enero paciente 9999.jpg`. The image content is represented as a series of garbled characters and symbols.

It is correct because if we save it and open, it loads the original image:



3 However here comes the challenge:

When we execute the method "consultarImagen" to get the uploaded image from the ESB, to the patient 9999:

1 The external system replies something which looks like a JPG:

# How could we retrieve an image from an external system using Ensemble?

Published on InterSystems Developer Community (<https://community.intersystems.com>)

Visual Trace interface showing a workflow diagram and a detailed log entry. The log entry includes fields like ID, Tipo, Texto, Registrado, Origen, Sesión, Job, Clase, Método, Seguimiento, and Pila. The 'Texto' field contains a URL and a base64-encoded image data.

And we send to the service something which could look like the original JPG:

Visual Trace interface showing a workflow diagram and a detailed log entry. The log entry shows an XML response with a 'resultado' field set to 'true' and an 'informacion' field containing a URL and base64-encoded image data.

But when we see the output being sent to the POSTMAN, we see some strange symbols like "/x00" or "/x01"

Postman showing the body of a POST request. The body is a JSON object with 'resultado' set to true and 'informacion' containing a URL and base64-encoded image data. The base64 data is heavily escaped with hex codes like \x00 and \x01.

So if we save this response:

```
1  [
2  "resultado": true,
3  "informacion": "ÿøÿà\x00DIEJFIF\x00\x01"
4  "error": null
5  ]
```

And we leave just the image part:

```
1  ÿøÿà\x00DIEJFIF\x00\x01\x01\x00\x00\x01\x00\x01\x00\x01\x00\x00ÿ0\x00...
  ...ÿ00\t\x06\x07...
  ...DIEJFIF\x00\x01\x01\x00\x00\x01\x00\x01\x00\x00ÿ0\x00...
```

It does not represent an image



4 In addition, we have tried in our REST Service, to convert the image to base64 as follows:

```
/// Obtener la imagen guardada
Method consultarImagen(pInput As %Stream.Object, Output pOutput As %Stream.Object) As
%Status
{
Set pOutput=##class(%GlobalBinaryStream).%New()
set claseAux = ##class(%ZEN.Auxiliary.jsonProvider).%New() //Necesitamos sacar la cab
ecera "request" de la petición REST
set req = $tr(pInput.GetAttribute("request"),"")
set tSC= claseAux.%ConvertJSONToObject(.req,"Mensajes.Request.miSCS.ConsultarImagen",
.objetoEntrada,1) //Enviamos al Proceso
set tSC = ..SendRequestSync("miSCS",objetoEntrada,.objetoSalida)

//Implementar el codificar la imagen que viene en objetoSalida.informacion de texto c
on letras extrañas a base64 $$$LOGINFO("objetoSalida.informacion: "_objetoSalida.info
rmacion)
set imagenBase64 = $System.Encryption.Base64Encode(objetoSalida.informacion)
$$$LOGINFO("imagenBase64: "_imagenBase64)
set objetoSalida.informacion = imagenBase64
$$$LOGINFO("objetoSalida.informacion: "_objetoSalida.informacion)
```

```
//Convertimos el OBJETO devuelto por el Proceso en JSON
set tSC = claseAux.%WriteJSONStreamFromObject(.pOutput,.objetoSalida,,,"aeloqtuw") /
/Enviamos el JSON con cabeceras
Do:$$$ISOK(tSC) pOutput.SetAttribute("Content-Type","application/json")
do pOutput.SetAttribute("Access-Control-Allow-Origin","*")
do pOutput.SetAttribute("Access-Control-Allow-Credentials","true")
do pOutput.SetAttribute("Access-Control-Allow-Methods","GET")
do pOutput.SetAttribute("Access-Control-Allow-Headers","request,Access-Control-Allow-Headers, Origin,Accept, X-Requested-With, Content-Type, Access-Control-Request-Method, Access-Control-Request-Headers")
Quit tSC
}
```

If we get the image now, we observe it as base64:



When we download it:

```
1 {
2   "resultado": true,
3   "informacion": "\/9j\/4AAQSkZJRgABAQAAQABAAD
4   "error": null
5 }
```

Get just the image part:

```
1 \/9j\/4AAQSkZJRgABAQAAQABAAD\/2wCEAAkGBxMSEhUTEhUWFhUWGBgbGRcYF
```

Then decode it from base64:



The Form.SizeLimit is 10000000bytes. Please, do not post more data using this form.

Source data from the Base64 string:



Type (or copy-paste) some text to a textbox below. The text can be a Base64 string to decode or any string to encode to a Base64.

or select a file to convert to a Base64 string.

Examinar... No se ha seleccionado ningún archivo. Convert the source data

What to do with the source data:

**encode** the source data to a **Base64** string (base64 encoding)  
 Maximum characters per line:

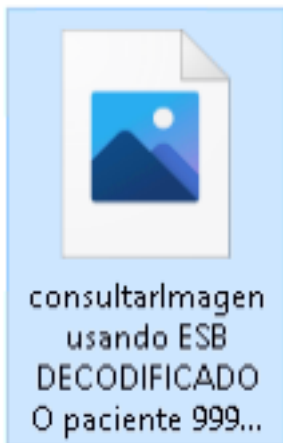
**decode** the data from a **Base64** string (base64 decoding)

Output data:

output to a textbox (as a string)

export to a binary file, filename:

If we save the string generated as a jpg, it does not represent an image:



Could you help us??

We have also read:

<https://community.intersystems.com/post/decode-base64-file>

<https://community.intersystems.com/post/how-save-png-image-cache-base64-...>

<https://community.intersystems.com/post/send-image-multi-part-file>

<https://docs.intersystems.com/irisforhealthlatest/csp/docbook/Doc.View.c...>

<https://docs.intersystems.com/irisforhealthlatest/csp/docbook/Doc.View.c...>

Thanks for your help,

[#Code Snippet](#) [#REST API](#) [#Ensemble](#)

---

Source

URL:<https://community.intersystems.com/post/how-could-we-retrieve-image-external-system-using-ensemble>