

GEO TAG INSTALLATION

-MANUAL-



GEOSYs



Index

Introduction	2
About the technologies	3
Experience Builder Developer Edition	3
Microsoft Graph.....	4
SharePoint Taxonomy	4
Installation	6
ArcGIS Experience Builder Developer Edition	6
GeoTag Widget	12
Microsoft Graph and Sharepoint configuration	18
Bibliography	26
Github Repositories	26
Experience Builder Developer Edition	26
Microsoft Graph.....	26
Others	26

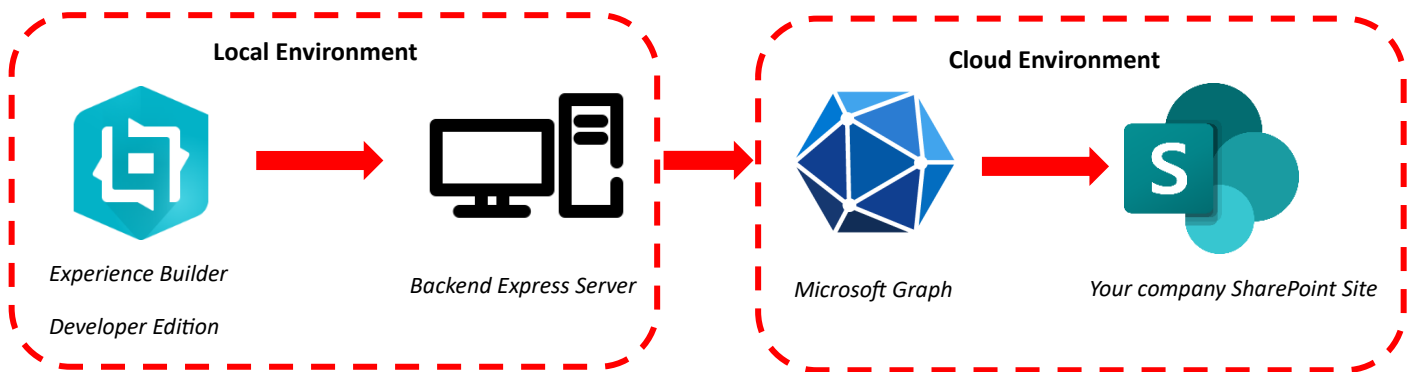
Introduction

We Geosys have developed a powerful custom widget for Experience Builder, capable of seamlessly browse files and folders stored on your company's SharePoint Sites, pinpoint them on a map and later retrieve them using a user-friendly interface integrated within Experience Builder, merging these two powerful worlds of storing data and GIS.

In this manual you will find an explanation of the technologies that we used for a better understanding of the widget, and how to install it and configure it.

If your company already installed the widget, please refer to the other manual, "GeoTag Usage" to know how to use it and take full advantage of.

Clarified that, this is a schema of all the subprocess that the Widget goes through:



As we indicate on the schema, there are two services that need to be installed locally. While we recommend using a server machine, It's also functional on a standard computer. For this manual, we demonstrate using a Windows Server machine, but the steps are equally applicable to a regular Windows machine.

For a more detailed understanding of how the widget operates, please refer to the "GeoTagFlowchart" image in the Documentation folder.

About the technologies

Experience Builder Developer Edition



For having custom widgets on Experience Builder, Esri gives plenty of tools to work with, but also there are some limitations that we are going to point out. The most important one is that currently, on version 1.12, there is no way to use custom Widgets on Experience Builder Online. Esri has a specific edition for testing and using those custom Widgets called “Experience Builder Developer Edition” that visually and functionality is identical to EB Online.

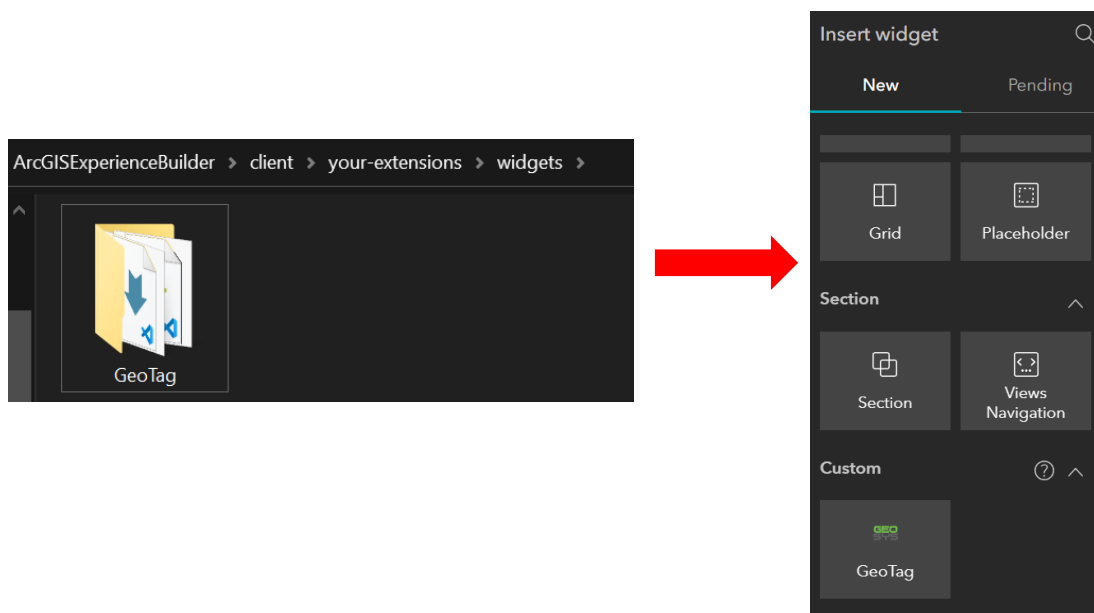
EBDE (for now on, we will use this short name for referring to Experience Builder Developer Edition) works locally on a server, but you can easily deploy the Web Apps that you will create to ArcGIS Online or Enterprise. Also, if you have existing “experiences” on your EB Online or in your computer you can import them onto the Dev Edition easily.

EBDE SDK has two folders, the “server” and “client”. Both are different services that are executed separately, but they need each other to work properly.

By default, it runs on port 3001 with the HTTPS protocol using a self-signed certificate.

Another thing to consider about EBDE is that when Esri updates the Experience Builder to a new version, you need to manually download the SDK with the latest version and move the custom widgets to the new version. If you don't have the latest version of EBDE, it can cause some problems like, maybe if you want to import an experience from EB Online maybe it will be incompatible, so make sure always to have the latest version. You can check the latest versions of Experience Builder on here: <https://developers.arcgis.com/experience-builder/guide/release-versions/>

Here we can see how it works adding a custom widget and how it appears in EBDE:



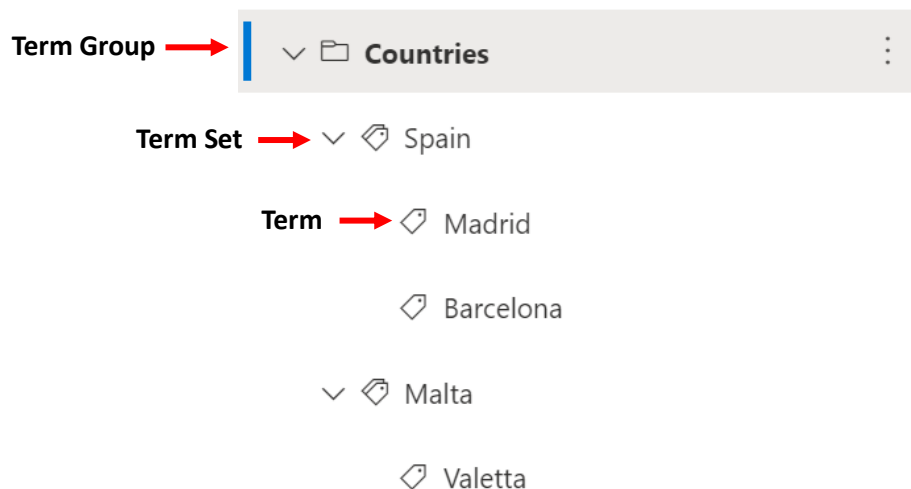
Microsoft Graph



Microsoft has a complex environment, with multiple applications such as SharePoint, Outlook, Office, Teams... Each one has their respective APIs to work with, but Microsoft Graph what it does is grouping all these APIs onto a same Endpoint, where you can access data from SharePoint and Teams and office with ease. For the case of our widget, we only use the features that interact with SharePoint, but is a powerful API that each day is growing.




SharePoint Taxonomy

Taxonomy is the science of classification, and SharePoint gives us a powerful and easy way to classify our files called "Term Store". SharePoint has 3 levels of classification: Term Groups, Term Sets and Terms. The Terms are the identifiers that you apply to files, so you can have a structure like this:

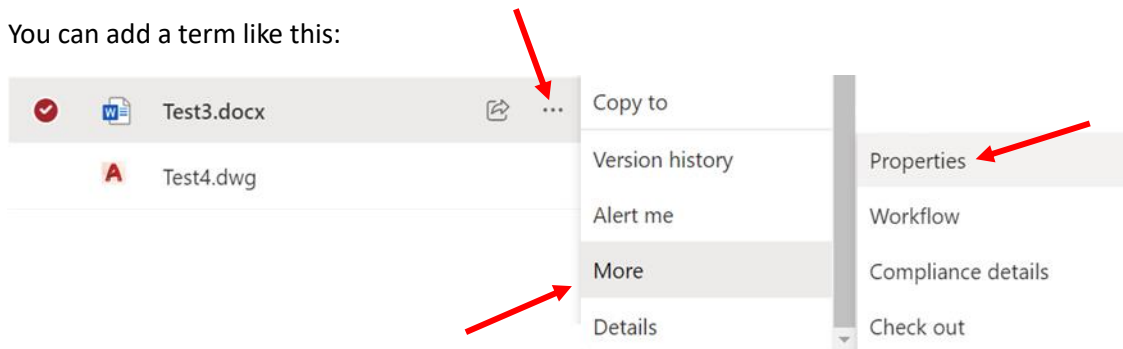


If you want to apply those terms onto files, you will need to create a column that will store those values for all the files of your SharePoint Site (folders cannot contain terms).

The column can hold a Term Set, and you can apply all the terms of the selected Term Set to the files. Here the column is called "Countries":

 Name ▾	Countries ▾	Modified ▾
 Test1.docx	Barcelona	A few seconds ago
 Test2.xlsx	Madrid	A few seconds ago

You can add a term like this:



Here it will appear an input where you can write the desired term (it needs to exist the term).

Test3.docx

Name *

Test3.docx

Title

Enter value here

Image Tags ⓘ

Enter value here

Countries

Type term to tag

Clicking here will display all the terms that you can apply to a file:

Countries ×

Type term to tag

Select a tag

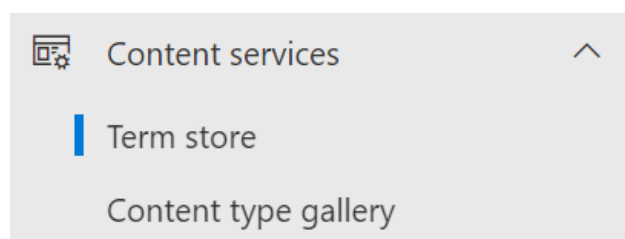
Spain

☐ Barcelona

☐ Madrid

For our Widget, we take profit on this, having a Term Set that stores all the terms that will have the files.

You can find the Term Store on the SharePoint Admin center (only for users with specific roles), inside Content Services

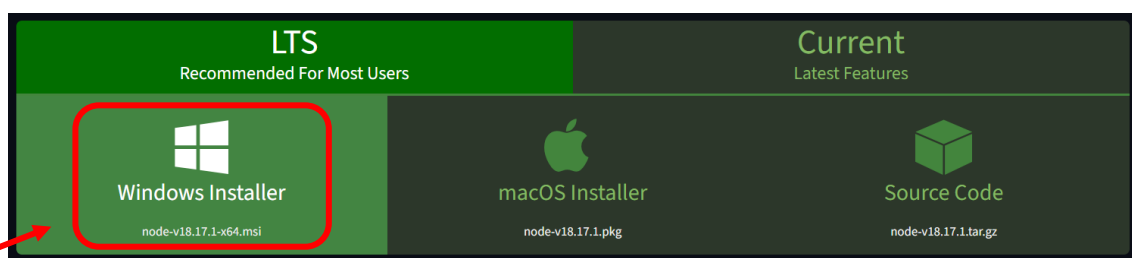


Installation

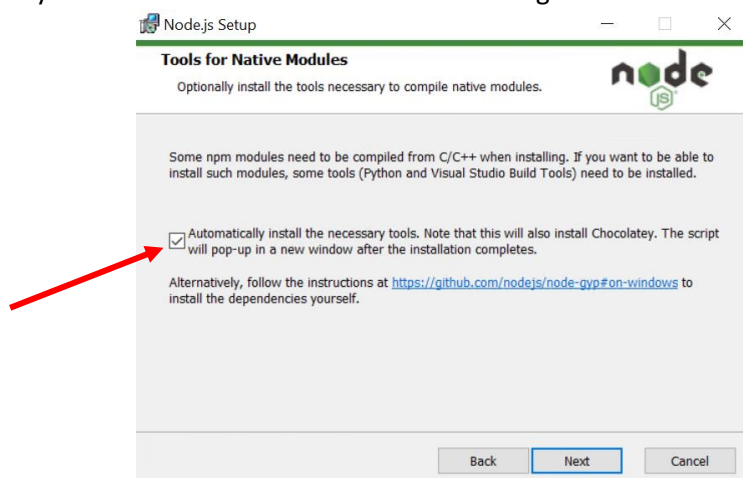
ArcGIS Experience Builder Developer Edition

We are going to explain all the steps for having a functional EBDE server. When making this manual, the latest version was 1.12, so please, also refer to the official documentation for avoiding version issues: <https://developers.arcgis.com/experience-builder/guide/install-guide/>

EBDE uses Node.js, when making this manual we were using version 18.17.1, but please, use the latest LTS version for Windows. You can find it here: <https://nodejs.org/en/download>



The installation is very straightforward, you will only need to select “next” until the next screen, here you will need to check the box for installing all the necessary tools:



After ending the installation, it will appear a terminal, we just simply need to press any key to continue. This will take a little bit of time.

```

Administrator: Install Additional Tools for Node.js
=====
Tools for Node.js Native Modules Installation Script
=====

This script will install Python and the Visual Studio Build Tools, necessary
to compile Node.js native modules. Note that Chocolatey and required Windows
updates will also be installed.

This will require about 3 GiB of free disk space, plus any space necessary to
install Windows updates. This will take a while to run.

Please close all open programs for the duration of the installation. If the
installation fails, please ensure Windows is fully updated, reboot your
computer and try to run this again. This script can be found in the
Start menu under Node.js.

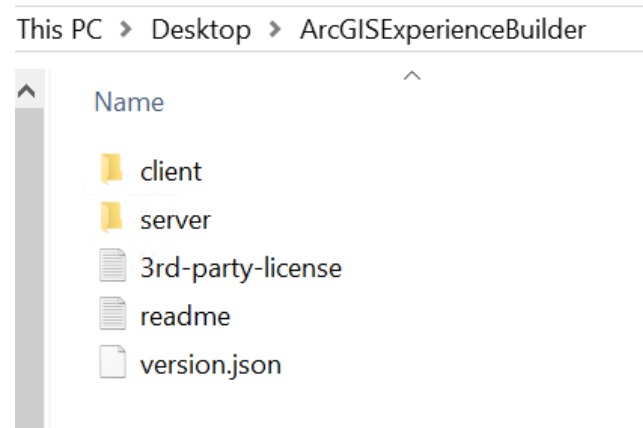
You can close this window to stop now. Detailed instructions to install these
tools manually are available at https://github.com/nodejs/node-gyp#on-windows

Press any key to continue . . .
  
```

After the installation ends successfully, we can start downloading the EBDE. Here we can find the latest versions, please download the latest release:

<https://developers.arcgis.com/downloads#arcgis-experience-builder>

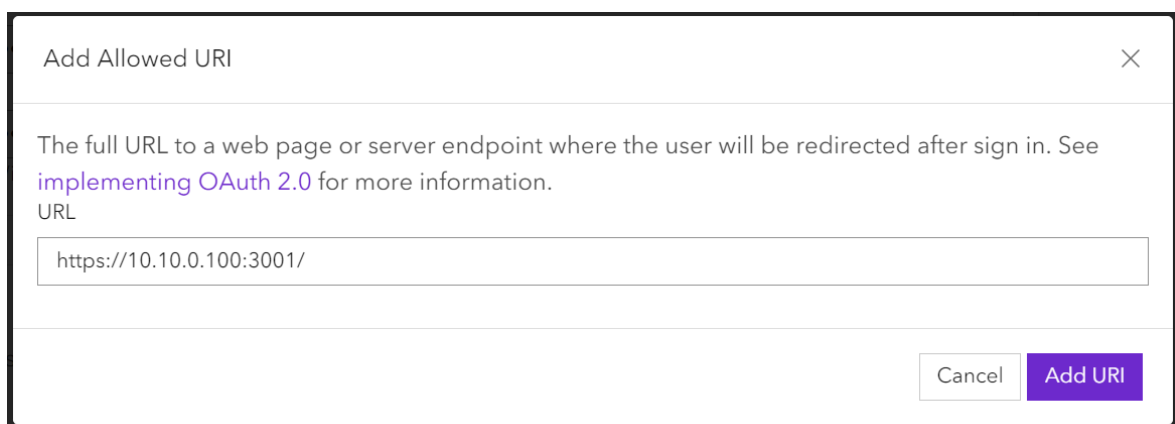
Move it to an accessible folder and unzip it. You will have a file structure like this. The main logic of EBDE is split into two folders: “client” and “server”.



Now you will need to create an OAuth2 App for synchronizing the EBDE with your Esri infrastructure. Go to <https://developers.arcgis.com/dashboard/> and click the OAuth 2.0 tab.

Go to new application, give it a name (description not necessary) and after that you will have created the OAuth app. Now you will need to scroll down to the section “Redirect URLs” and click on “Add URL”.

Here you will need to write the IP or name of the machine of your own server, it is important to use https and port 3001. In our case we used the IP of our server that is “10.10.0.100”.

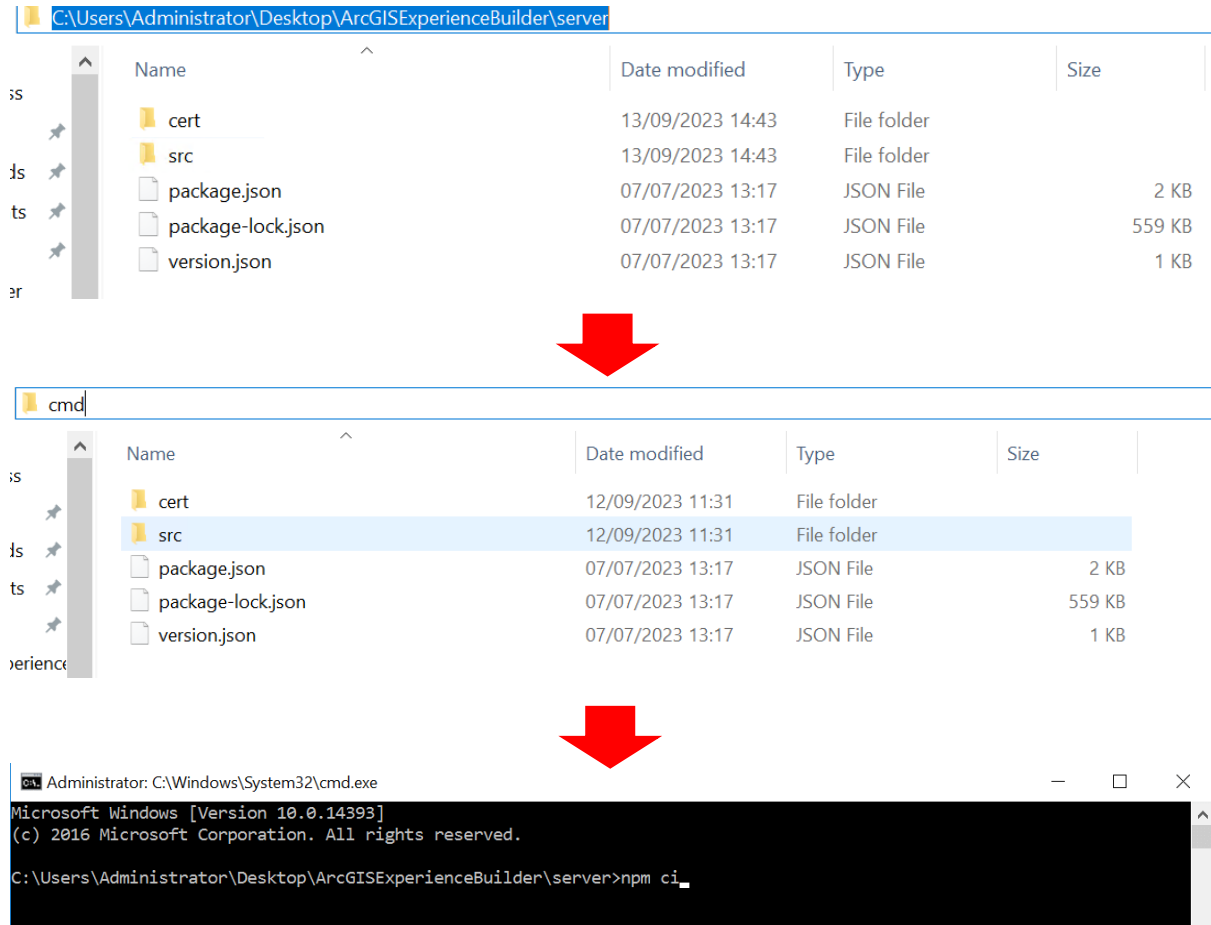


Save the Client ID. All the users will need it for accessing the server for the first time.

Now you will need to install some node packages on the server. With a terminal, you will need to execute the command “npm ci” inside the “server” folder.

Tip: To open a command prompt directly on the folder that you want you just simply need to write “cmd” on the directory bar:

Delete the current route and just write “cmd”, then press enter:



Write the command “npm ci” and press enter.

Maybe there are some warnings and vulnerabilities, this is common for a node installation, but don’t panic, this will not affect the security or affect the server in some way.

```
C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\server>npm ci
npm WARN deprecated @types/mongodb@4.0.7: mongodb provides its own types. @types/mongodb is no longer needed.
npm WARN deprecated multer@1.4.4: Multer 1.x is affected by CVE-2022-24434. This is fixed in v1.4.4-lts.1 which drops su
pport for versions of Node.js before 6. Please upgrade to at least Node.js 6 and version 1.4.4-lts.1 of Multer. If you n
eed support for older versions of Node.js, we are open to accepting patches that would fix the CVE on the main 1.x relea
se line, whilst maintaining compatibility with Node.js 0.10.

added 712 packages, and audited 713 packages in 46s

83 packages are looking for funding
  run `npm fund` for details

10 vulnerabilities (6 moderate, 4 high)

To address issues that do not require attention, run:
  npm audit fix

To address all issues possible (including breaking changes), run:
  npm audit fix --force

Some issues need review, and may require choosing
a different dependency.

Run `npm audit` for details.
```

Now you will have finished the installation of the server. Check if everything is working fine by executing on the same terminal “npm start”. This command line is for starting the EBDE manually. If everything is going perfectly, it should appear something like this:

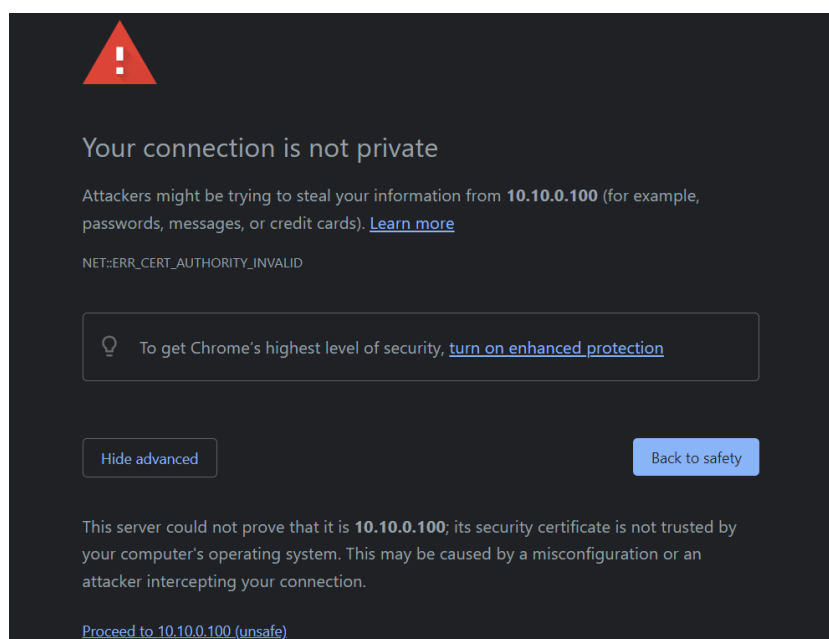
```
C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\server>npm start

> exb-server@0.1.0 start
> cross-env NODE_ENV=production node src/server

Apps folder: C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\server\public\apps
Http server running on port 3000
Https server running on port 3001
```

Try to connect with another computer, using the URL that you specified. If it doesn't work, check the firewall of your server and if necessary, add an Inbound and Outbound Rule for the port 3001 and 3002 (for the backend server that you will install).

If everything is well configured, it should appear a screen like this indicating that the connection is not private. This is because EBDE is using a self-signed certificate. You can feel free to use your own certificates. But for the moment you will just need to select the option “Proceed to 10.10.0.100”



Now you will need to specify your ArcGIS URL organization and the Client ID that we generated before.

Specify the URL to your ArcGIS Online organization or ArcGIS Enterprise

https://geosys-mt.maps.arcgis.com

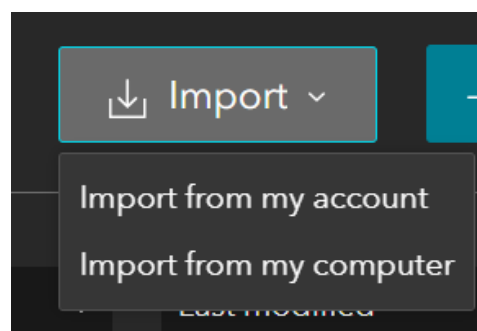
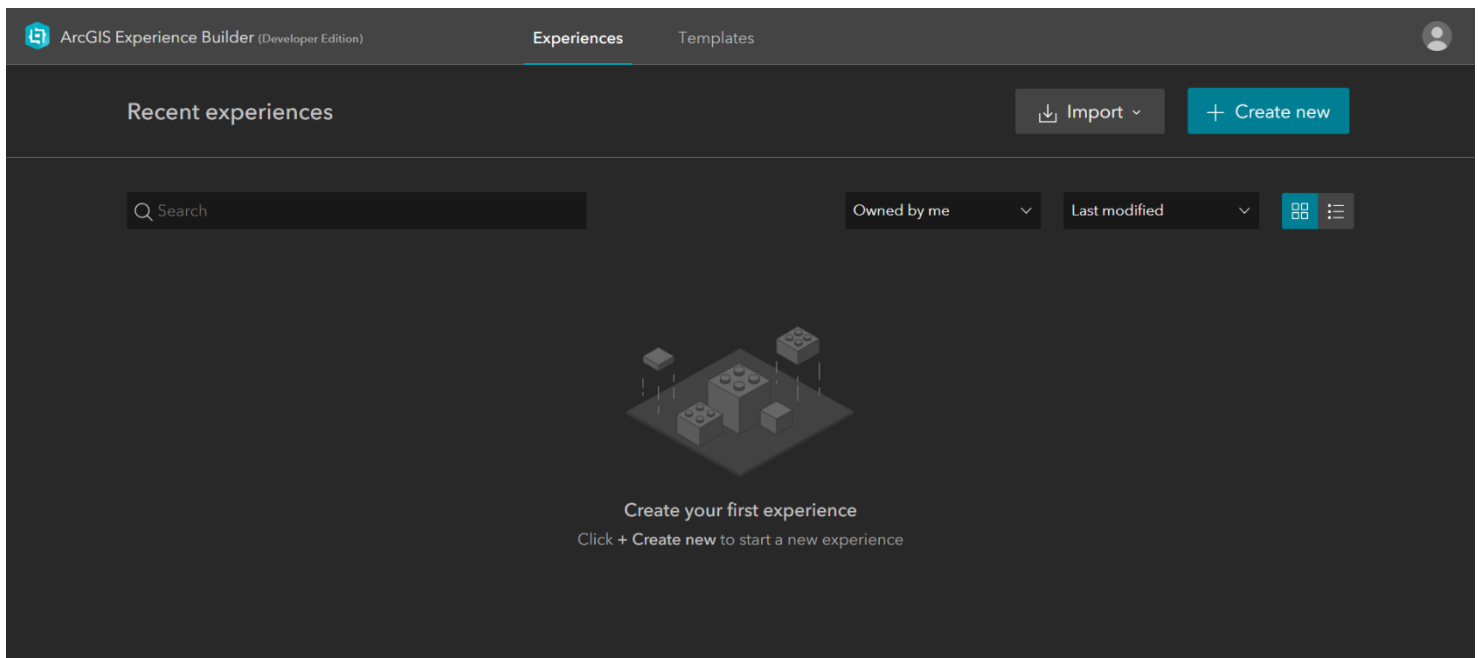
Provide the Client ID for Experience Builder registered in the ArcGIS Enterprise you specify above

.....

Sign in

Then it will redirect to a normal esri Sign in page, write your credentials.

And now it should open the main page of EBDE. As you can see, is almost identical to EB Online, with the difference that we have the option of importing our experiences from our account or computer.



Now you will need to go back to the server machine to finish the configuration, this time inside the “client” folder. The client folder is the one that will store the custom widget, for the moment we are going to install it by default, and later we are going to add the widget.

We just need to open a terminal and execute the command “npm ci” again (remember the tip for opening the terminal on the desired folder). It will also show some warnings and vulnerabilities, but nothing to worry about.

```
C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\client>npm ci
npm WARN ERESOLVE overriding peer dependency
npm WARN While resolving: react-popper@1.3.11
npm WARN Found: react@18.2.0
npm WARN   node_modules/react
npm WARN   react@"^18.2.0" from the root project
npm WARN   36 more (@emotion/react, @emotion/styled, ...)
npm WARN
npm WARN Could not resolve dependency:
npm WARN peer react@"0.14.x || ^15.0.0 || ^16.0.0 || ^17.0.0" from react-popper@1.3.11
npm WARN   node_modules/reactstrap/node_modules/react-popper
npm WARN     react-popper@"^1.3.6" from reactstrap@8.10.1
npm WARN       node_modules/reactstrap
npm WARN
npm WARN Conflicting peer dependency: react@17.0.2
npm WARN   node_modules/react
npm WARN   peer react@"0.14.x || ^15.0.0 || ^16.0.0 || ^17.0.0" from react-popper@1.3.11
npm WARN     node_modules/reactstrap/node_modules/react-popper
npm WARN       react-popper@"^1.3.6" from reactstrap@8.10.1
npm WARN         node_modules/reactstrap
npm WARN deprecated @npmcli/move-file@1.1.2: This functionality has been moved to @npmcli/fs
npm WARN deprecated @npmcli/move-file@2.0.1: This functionality has been moved to @npmcli/fs
npm WARN deprecated source-map-url@0.4.1: See https://github.com/lydell/source-map-url#deprecated
npm WARN deprecated urix@0.1.0: Please see https://github.com/lydell/urix#deprecated
npm WARN deprecated rollup-plugin-terser@7.0.2: This package has been deprecated and is no longer maintained. Please use
@rollup/plugin-terser
npm WARN deprecated source-map-resolve@0.5.3: See https://github.com/lydell/source-map-resolve#deprecated
npm WARN deprecated chokidar@2.1.8: Chokidar 2 does not receive security updates since 2019. Upgrade to chokidar 3 with
15x fewer dependencies
```

If we want to start it manually, we can also use “npm start”, but for the moment is not necessary because we haven’t added the widget yet.

Now the last part is making the server folder as a Windows service. We must go again to the command prompt inside the “server” folder. Before that if you still have the “server” or “client” started (with the command npm start that you used before) you need to stop it by pressing “Ctrl+C”. Caution, closing the command prompt doesn’t stop the service, so be careful with that.

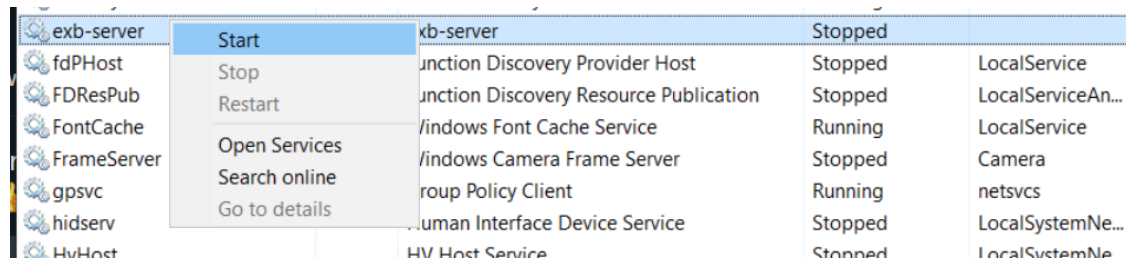
After being sure that nothing is running, we need to open the terminal inside “server” and write “npm run install-windows-service”. If everything went fine, it should appear the following:

```
C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\server>npm run install-windows-service
> exb-server@0.1.0 install-windows-service
> winser -i --startcmd="node src/server"

Use start command "node src/server".
The program "exb-server" was installed as a service.

C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\server>
```

Now we need to open the task manager and then go to “Services”. There we will need to find the service “exb-server” and we need to Start it.

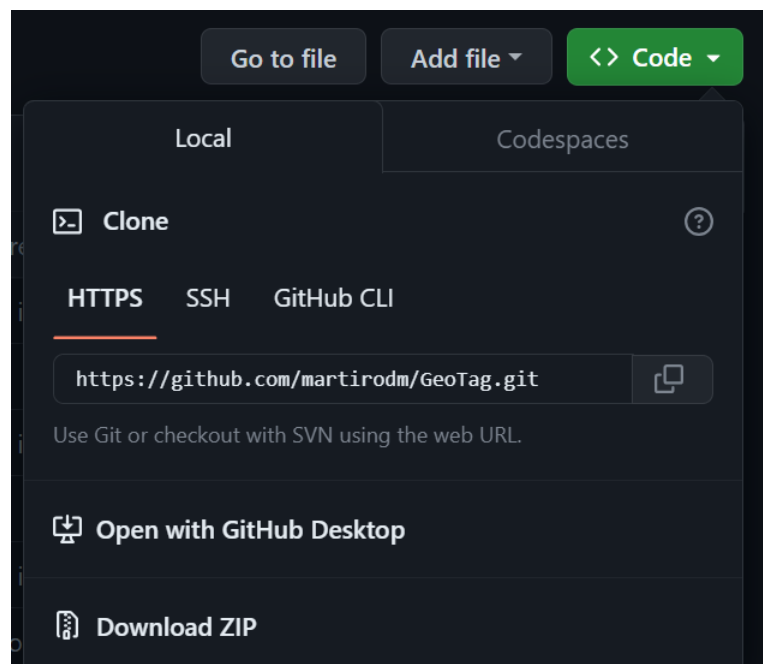


Now the EBDE is a service, so this means you will not need no write each time you want to start the server “npm start” because it will execute with the server automatically.

GeoTag Widget

Now you will need to add the custom widget that we GEOSYS, developed. Firstly, go to the next Github Repository: <https://github.com/martirodm/GeoTag>

Then if you go to <> Code you will se some options to download or clone the code. If you have .git on your computer you can simply clone it, and if it is not the case just download the ZIP



You will need to clone it or decompress the zip on the folder “widgets” that is inside “your-extensions” on the folder “client”. You can delete the file “.gitkeep” and the folder “simple”.

This PC > Desktop > ArcGISExperienceBuilder > client > your-extensions > widgets >

Name	Date modified	Type	Size
simple	13/09/2023 14:43	File folder	
.gitkeep	07/07/2023 13:17	GITKEEP File	0 KB
GeoTag-main	13/09/2023 15:23	Compressed (zipped)...	487 KB

After extracting the zip, delete the zip and only leave the folder. Now change the name of the folder to only “GeoTag”. This step is very important, and it needs to be exactly identical the name.

This PC > Desktop > ArcGISExperienceBuilder > client > your-extensions > widgets

Name	Date modified	Type	Size
GeoTag	13/09/2023 15:25	File folder	

Also, if inside GeoTag there is again the same folder that contains all the files, just move all the files inside the first GeoTag folder and delete the other folder.

ArcGISExperienceBuilder > client > your-extensions > widgets > GeoTag > ~~GeoTag main~~

It should be like this:

This PC > Desktop > ArcGISExperienceBuilder > client > your-extensions > widgets > GeoTag

Name	Date modified	Type	Size
src	13/09/2023 15:25	File folder	
.gitignore	13/09/2023 04:59	GITIGNORE File	2 KB
config.json	13/09/2023 04:59	JSON File	1 KB
icon	13/09/2023 04:59	SVG Document	135 KB
manifest.json	13/09/2023 04:59	JSON File	1 KB
package.json	13/09/2023 04:59	JSON File	1 KB
package-lock.json	13/09/2023 04:59	JSON File	108 KB
README.md	13/09/2023 04:59	MD File	1 KB

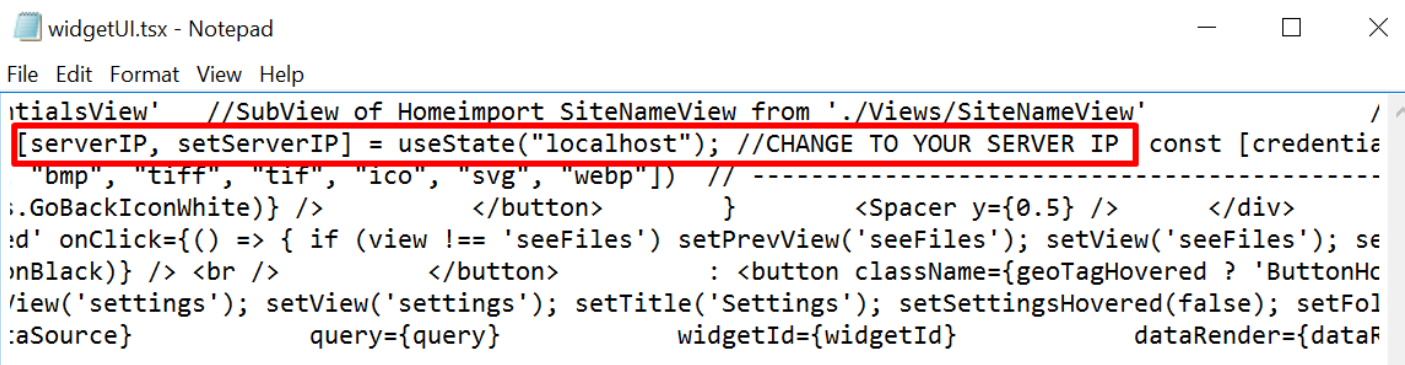
You will need to change some parameters. Go to “src”, then to “runtime” and open the file “widgetUI.tsx”, you can use notepad if you don’t have any code editor.

It will look something like this:

widgetUI.tsx - Notepad

```
File Edit Format View Help
import React, { useState, createContext, useContext } from 'react'//import { DataSourceCompon
port SeeTaggedFilesView from './Views/SeeTaggedFilesView' //SubView of SeeFilesimport AddTa
als] = useState({ client_id: '', client_secret: '', tenant_id: '', }) const [siteN
- if (view === prevView) { setPrevView(null) } // If the date source is not configured,
className="left-bar"> {view === 'home' || view === 'appCredentials' || view === 'siteN
les'); setFolderFinalId(folderId) }}<img src={String(BlackIcons.EyeFileIconBlack
ed'} onMouseOver={() => setGeoTagHovered(true)} onMouseOut={() => setGeoTagHovered(false)} on
<img src={String(settingsHovered ? HoverIcons.SettingIconHover : WhiteIcons.Setting
eMapWidgetIds={useMapWidgetIds} /> {view === 'geoTag' && <GeoTagView setVi
```

You need to find the following piece of code:

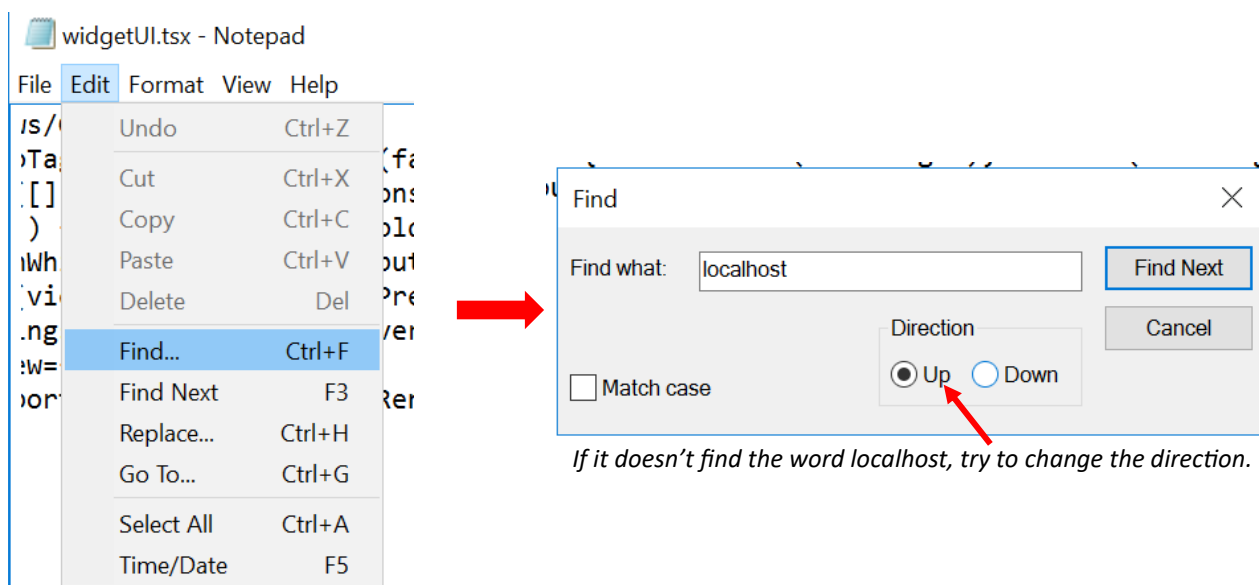


```

//SubView of Homeimport SiteNameView from './Views/SiteNameView'
[serverIP, setServerIP] = useState('localhost'); //CHANGE TO YOUR SERVER IP const [credential
"bmp", "tiff", "tif", "ico", "svg", "webp"]) // -----
GoBackIconWhite)} /> </button> } <Spacer y={0.5} /> </div>
d' onClick={() => { if (view !== 'seeFiles') setPrevView('seeFiles'); setView('seeFiles'); se
nBlack)} /> <br /> </button> : <button className={geoTagHovered ? 'ButtonHc
/view('settings'); setView('settings'); setTitle('Settings'); setSettingsHovered(false); setFol
aSource} query={query} widgetId={widgetId} dataRender={dataR

```

If you don't find it, you can use the search tool of notepad:



Now you will need to replace the word "localhost" with the IP of you server machine.

```

const [serverIP, setServerIP] = useState("10.10.0.100"); //CHANGE TO YOUR SERVER IP

```

Save and close the notepad.

After that, inside the folder GeoTag (be careful that has all the files, not the duplicated folder), open a command prompt and execute "npm install".

```

C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\client\your-extensions\widgets\GeoTag>npm install
added 221 packages, and audited 222 packages in 45s

16 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\client\your-extensions\widgets\GeoTag>

```

Close the terminal after that.

Now you will need to download a Backend server that will make the connection between EBDE and Microsoft Graph. You will find the backend server on the next github:

<https://github.com/martirodm/GeoTagExpress>

Clone the project or download the Zip onto the folder “arcgis-experience-builder-version”. Extract it and change the name to “GeoTagExpress”:

This PC > Desktop > ArcGISExperienceBuilder >

Name	Date modified	Type	Size
client	13/09/2023 15:04	File folder	
GeoTagExpress	13/09/2023 15:33	File folder	
server	13/09/2023 14:59	File folder	
version.json	07/07/2023 12:59	JSON File	1 KB
3rd-party-license	07/07/2023 12:59	Text Document	68 KB
readme	07/07/2023 12:59	Text Document	1 KB

If it happens again the problem of having the same folder inside, just do the same, move all the files to the first folder and delete the duplicated one.

This PC > Desktop > ArcGISExperienceBuilder > GeoTagExpress

Name	Date modified	Type	Size
.gitignore	13/09/2023 06:53	GITIGNORE File	1 KB
package.json	13/09/2023 06:53	JSON File	1 KB
package-lock.json	13/09/2023 06:53	JSON File	37 KB
README.md	13/09/2023 06:53	MD File	1 KB
server	13/09/2023 06:53	JavaScript File	15 KB
service-setup	13/09/2023 06:53	JavaScript File	1 KB

After that you will need to copy the self-signed certificate of the server and paste it on the GeoTagExpress folder. You will find the certificates here:

This PC > Desktop > ArcGISExperienceBuilder > server > cert

Name	Date modified	Type	Size
server.cert	07/07/2023 13:17	CERT File	2 KB
server.key	07/07/2023 13:17	KEY File	2 KB

Just copy-paste it to the GeoTagExpress folder

This PC > Desktop > ArcGISExperienceBuilder > GeoTagExpress >

Name	Date modified	Type	Size
daemon	13/09/2023 15:41	File folder	
node_modules	13/09/2023 15:35	File folder	
.gitignore	13/09/2023 06:53	GITIGNORE File	1 KB
package.json	13/09/2023 06:53	JSON File	1 KB
package-lock.json	13/09/2023 15:35	JSON File	37 KB
README.md	13/09/2023 06:53	MD File	1 KB
server.cert	07/07/2023 13:17	CERT File	2 KB
server	13/09/2023 16:12	JavaScript File	15 KB
server.key	07/07/2023 13:17	KEY File	2 KB
service-setup	13/09/2023 15:37	JavaScript File	1 KB

Now open inside here again a command prompt and write the command “npm install”:

```
C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\GeoTagExpress>npm install

added 97 packages, and audited 98 packages in 4s

13 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\GeoTagExpress>_
```

After that, you will convert it to a Windows Service, like we did before with the “server” folder. Now write the command “node service-setup.js”

If everything went well it should say “Service installed”.

```
description: 'ExpressGeoTag',
logpath: null,
env: undefined,
execPath: null,
logOnAs: {
  account: null,
  password: null,
  domain: 'WIN-D4JBFT9UD43',
  mungeCredentialsAfterInstall: true
},
workingdirectory: 'C:\\Users\\Administrator\\Desktop\\ArcGISExperienceBuilder\\GeoTagExpress',
stopparentfirst: undefined,
stoptimeout: 30,
logmode: 'rotate',
logging: undefined,
allowServiceLogon: undefined
}
}
Service installed.
```

Check on task manager to see the service. Is called “expressgeotag.exe”. If it’s stopped, start it.

expressgeotag.exe		Start	Stopped	
exb-server	138	Stop	Running	
EventSystem	380	Restart	Running	LocalService
EventLog	348		Running	LocalServiceNe...
EntAppSvc		Open Services	Stopped	appmodel
embeddedmode		Search online	Stopped	LocalSystemNe...
EFS		Go to details	Stopped	

Now that we have the two services started, we just need to activate the widget. On the “client” folder execute “npm run build:prod”. It would appear some warnings, but it doesn’t affect the widget.

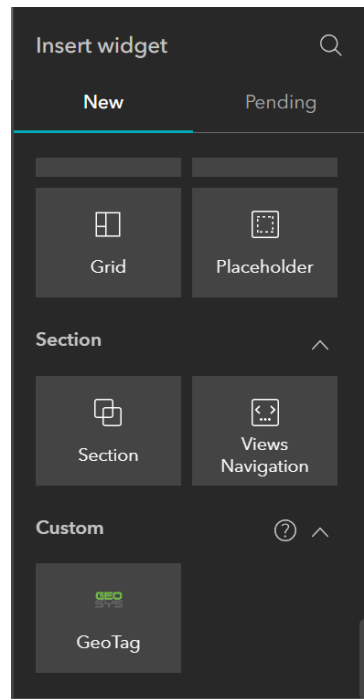
```
C:\Users\Administrator\Desktop\ArcGISExperienceBuilder\client>npm run build:prod

> exb-client@1.12.0 build:prod
> cross-env NODE_ENV=production OUTPUT_FOLDER=./dist-prod webpack --mode production

Entrypoint widgets/GeoTag/dist/runtime/widget [big] 500 KiB = widgets/GeoTag/dist/runtime/widget.js
Entrypoint widgets/GeoTag/dist/setting/setting 2.03 KiB = widgets/GeoTag/dist/setting/setting.js
chunk (runtime: widgets/GeoTag/dist/setting/setting) widgets/GeoTag/dist/setting/setting.js (widgets/GeoTag/dist/setting/setting) 2.67 KiB (javascript) 697 b
tes (runtime) [entry] [rendered]
runtime modules 697 bytes 4 modules
dependent modules 126 bytes [dependent] 3 modules
cacheable modules 2.55 KiB
./jimu-core/lib/set-public-path.ts 331 bytes [built] [code generated]
./your-extensions/widgets/GeoTag/src/setting/setting.tsx 2.23 KiB [built] [code generated]
chunk (runtime: widgets/GeoTag/dist/runtime/widget) widgets/GeoTag/dist/runtime/widget.js (widgets/GeoTag/dist/runtime/widget) 1.1 MiB (javascript) 999 bytes
(runtime) [entry] [rendered]
runtime modules 999 bytes 6 modules
dependent modules 428 KiB [dependent] 112 modules
cacheable modules 698 KiB
./jimu-core/lib/set-public-path.ts 331 bytes [built] [code generated]
./your-extensions/widgets/GeoTag/src/runtime/widget.tsx + 209 modules 698 KiB [built] [code generated]

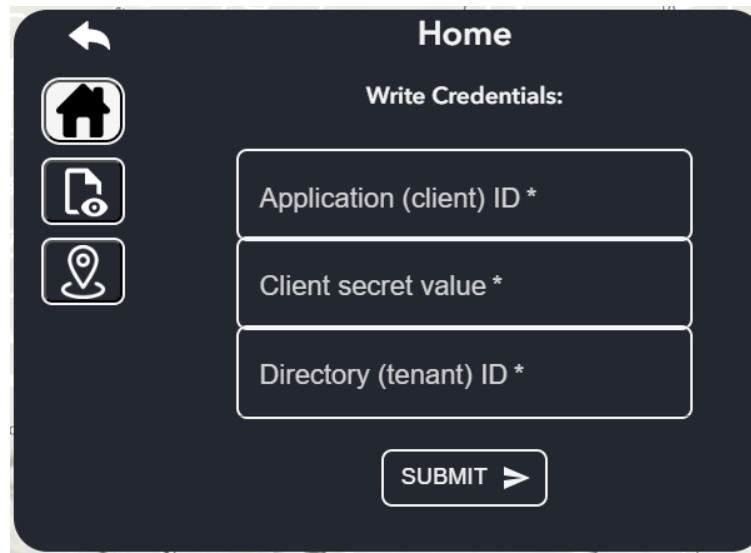
WARNING in asset size limit: The following asset(s) exceed the recommended size limit (244 KiB).
This can impact web performance.
Assets:
  widgets/GeoTag/dist/runtime/widget.js (500 KiB)
```

So now if we go to <https://your-serverip:3001/> and work on an experience, it should appear the widget:

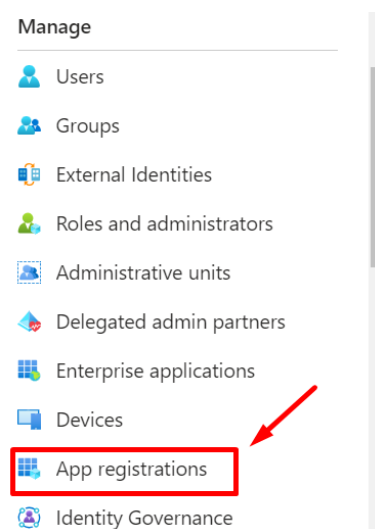


Microsoft Graph and Sharepoint configuration


The widget asks you for some credentials, this are for syncing the SharePoint of your company with the Widget. Let's see how we can get these values. You need to be a Microsoft Global Admin for doing these steps.



Firstly, you will need to open the portal of Azure: <https://portal.azure.com/> and go to the Azure Active Directory. Once there you will need to go to "App registrations"



And create a new registration

 **New registration**

You can name it as you want, but you need to select the option “Single tenant”. Ignore the “Redirect URI” option

* Name

The user-facing display name for this application (this can be changed later).

GeoTAG ✓

Supported account types

Who can use this application or access this API?

- ☒ Accounts in this organizational directory only (geosys.com.mt only - Single tenant)
- ☐ Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)
- ☐ Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
- ☐ Personal Microsoft accounts only

Now, we have created the app. Here we can find two of the 3 required values for login on the widget.

GeoTAG

Search

Delete Endpoints Preview features

Overview

Quickstart

Integration assistant

Manage

Branding & properties

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

Got a second? We would love your feedback on Microsoft identity platform (previously Azure AD for developer). →

Essentials

Display name
GeoTAG

Application (client) ID
558bc81c-8c9e-42e9-9629-46cf0aac21d5

Object ID
f349c4fd-bc56-40ee-b0d2-1a8ed8653d45

Directory (tenant) ID
baf298ec-7aae-477f-9fb9-ae5f31dd821

Supported account types
My organization only

Client credentials
Add a certificate or secret

Redirect URIs
Add a Redirect URI

Application ID URI
Add an Application ID URI

Managed application in local directory
GeoTAG

We can get the last value on “Certificates & secrets”. Click on new client secret

Certificates (0) Client secrets (0) Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value ⓘ	Secret ID
-------------	---------	---------	-----------

Then write the description that you want a choose an expiration date. We recommend 24 months.

Add a client secret

Description GeoTAG

Expires 730 days (24 months)

Copy the value. Is important to save this key on a safe place, as the next time you reload the page you will never see it again.

+ New client secret

Description	Expires	Value ⓘ	Secret ID
GeoTAG	9/12/2025	cjj8Q~jJP6J-YLQBcDemSBgPnlTB0Qlk...	001d1cdd-d1e8-4fac-b2d7-6b9eee4b...

Now we need to give the application some permissions. Go to API permissions and click on Add a permission.

Manage

- Branding & properties
- Authentication
- Certificates & secrets
- Token configuration
- API permissions**
- Expose an API

Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. [Learn more about permissions and consent](#)

+ Add a permission ✓ Grant admin consent for geosys.com.mt

API / Permissions name	Type	Description	Admin consent req...	Status
▼ Microsoft Graph (1)				
User.Read	Delegated	Sign in and read user profile	No	...

Select Microsoft Graph

Request API permissions

Select an API

Microsoft APIs APIs my organization uses My APIs

Commonly used Microsoft APIs

Microsoft Graph
Take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, and Windows 10. Access Microsoft Entra ID, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner, and more through a single endpoint.

Azure Communication Services
Rich communication experiences with the same secure CPaaS platform used by Microsoft Teams

Azure Rights Management Services
Allow validated users to read and write protected content

Azure Service Management
Programmatic access to much of the functionality available through the Azure portal

Application permissions

< All APIs

Microsoft Graph
<https://graph.microsoft.com/> Docs

What type of permissions does your application require?

Delegated permissions

Your application needs to access the API as the signed-in user.

Application permissions

Your application runs as a background service or daemon without a signed-in user.

And you will need to give all of the following permissions:

API / Permissions name	Type	Description	Admin consent req...	Status
Microsoft Graph (7) ...				
Files.ReadWrite.All	Application	Read and write files in all site collections	Yes	✔ Granted for geosys.com,... ...
Sites.FullControl.All	Application	Have full control of all site collections	Yes	✔ Granted for geosys.com,... ...
Sites.Manage.All	Application	Create, edit, and delete items and lists in all site collec...	Yes	✔ Granted for geosys.com,... ...
Sites.Read.All	Application	Read items in all site collections	Yes	✔ Granted for geosys.com,... ...
Sites.ReadWrite.All	Application	Read and write items in all site collections	Yes	✔ Granted for geosys.com,... ...
TermStore.Read.All	Application	Read all term store data	Yes	✔ Granted for geosys.com,... ...
TermStore.ReadWrite.All	Application	Read and write all term store data	Yes	✔ Granted for geosys.com,... ...

After that, you need to grant admin consent. If you are not a global admin, please ask him to consent it, if not is not going to work.

+ Add a permission ✔ Grant admin consent for geosys.com.mt

After that, the status will appear as Granted.

Status

...		
✔	Granted for geosys.com,...	...
✔	Granted for geosys.com,...	...
✔	Granted for geosys.com,...	...
✔	Granted for geosys.com,...	...
✔	Granted for geosys.com,...	...
✔	Granted for geosys.com,...	...
✔	Granted for geosys.com,...	...

Now with the 3 values that we got, we can now access to the widget

←

Home

🏠

📄

📍

Write Credentials:

Application (client) ID *

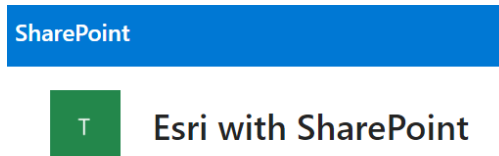
Client secret value *

Directory (tenant) ID *

SUBMIT ➡

The last step will be to create a SharePoint Column that will store all the tags. After writing the 3 values, you will need to also write the name of the SharePoint site that you want to work with.

As an example, we are going to use this SharePoint site



So here we can write the name

The screenshot shows a dark-themed mobile application interface. At the top, there is a back arrow icon and the word "Home". Below the back arrow are three icons: a house, a document with a magnifying glass, and a location pin. To the right of these icons, the text "Write the name of your SharePoint Site:" is displayed. Below this text is a text input field containing the text "Esri with SharePoint". At the bottom right, there is a "SUBMIT" button with a right-pointing arrow.

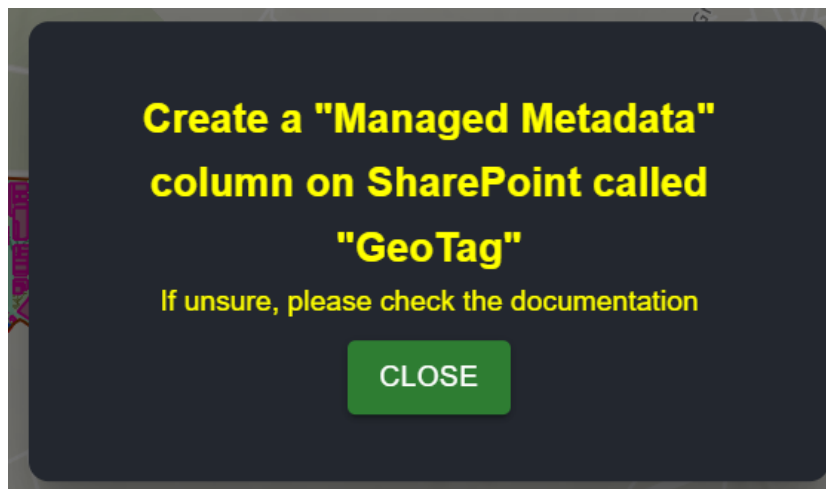
Now please go to "GeoTag" file and choose a random file to work with.



Select the file and on personalized, write a tag for testing. It can be whatever name. Click ADD

The screenshot shows a dark-themed mobile application interface titled "GeoTag File". Below the title, the text "Add Tag to Test1.dwg" is displayed. There are three tabs: "BY COORDINATE", "BY FIELD", and "PERSONALIZED". The "PERSONALIZED" tab is selected and underlined. Below the tabs, there is a text input field with the placeholder text "Enter tag... *" and the text "Test" entered. At the bottom, there is an "ADD" button.

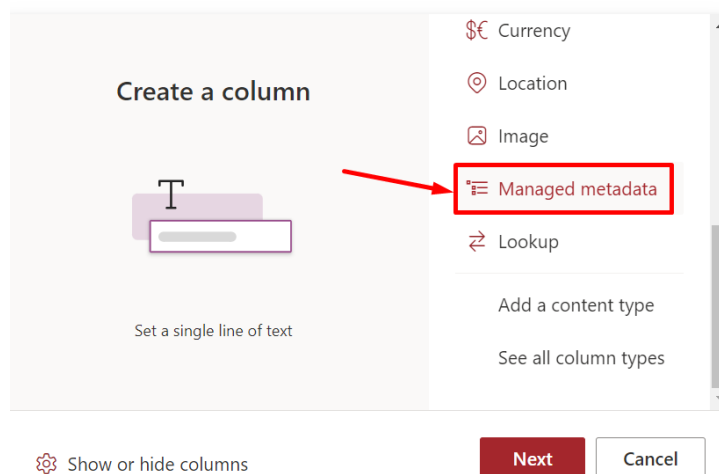
It should appear this warning.



Go to the SharePoint site that you've chosen, then go to Documents and click on Add column.



Select "Managed metadata"



THE COLUMN MUST BE NAMED “GeoTag”. With Upper case G and T. If the name is not identical it will not work.

Create a column

[Learn more about column creation.](#)

Name *

GeoTag

Description

Type

Managed metadata

Term set or term *

Selected: None

Select

More options

Save

Cancel

Then go to Select Term set or term and choose “GeoTag” and then “GeoTag” again:

Select term set or term

GeoTag

> Esri

✓ GeoTag

→ GeoTag

> People

> Search Dictionaries

After that click on More options and then allow multiple values. You don't need to modify the other parameters.

More options ▾

Display format

Choose if the column should display just the term (for example, Sydney) or the term plus its parents (for example, Geolocations: Australia: Sydney)

☒ Term only

☐ Term plus parents

Allow users to type new values

Choose if users can add values to the term set from this column. (Only open term sets allow this.)

☐ No

Allow multiple values

Specify whether the column will allow more than one value

☒ Yes 

Require that this column contains

And now it will appear the column GeoTag. Here it will show all the tags that have all the files. If you want you can hide the column, it will still work perfectly.

 Name ▾	Modified ▾	Modified By ▾	Version ▾	GeoTag ▾
---	------------	---------------	-----------	----------

And that will be it. Please refer to the “GeoTagUsage” Manual for knowing how to work with it.

Bibliography

Github Repositories

GeoTag Widget: <https://github.com/martirodm/GeoTag>

GeoTag Express: <https://github.com/martirodm/GeoTagExpress>

Experience Builder Developer Edition

Install guide: <https://developers.arcgis.com/experience-builder/guide/install-guide/>

ArcGIS Developers: <https://developers.arcgis.com/dashboard/>

Experience Builder SDK: <https://developers.arcgis.com/downloads#arcgis-experience-builder>

Microsoft Graph

Official Microsoft documentation: <https://learn.microsoft.com/en-us/graph/api/overview?view=graph-rest-1.0>

Azure: <https://portal.azure.com/>

Others

Node: <https://nodejs.org/en/download>