Upload an extension

Summary: This section provides information about **sharing** a custom extension with the wider CAST user community.

Introduction

If you have created a custom extension that you are actively using and you think other users could benefit from this, you can share your extension with the wider CAST community using **CAST Extend**.

Prerequisites to sharing your extension

plugin.nuspec file

The custom extension you have created must contain a **plugin.nuspec** file, as detailed in Extension description file.

Use the correct contributor type for your extension

You must ensure that you define the correct contributor type in the following places:

Ensure the the correct contributor type is defined in the plugin.nuspec file

Ensure the **<ID>** and **<authors>** elements are correctly defined in the Extension description file and use the correct contributor type

Have your extension in an appropriately named folder

Your extension should already be stored in an appropriately named folder, according to the contributor type. Either:

- for CAST AIP extensions provided by the **CAST User Community**: **com.castsoftware.uc.nameoftheextension**
- for CAST AIP extensions provided by the **CAST Labs**: **com.castsoftware.labs.nameoftheextension**

Please see Extension description file for more information.

Ensure your CAST Extend login has been assigned the correct contributor type

By default, ALL CAST Extend logins have the **CAST User Community** contributor type applied to them - this means that all logins have permission to upload a **CAST User Community** extension. However, if you want to upload any other type of extension (CAST Labs for example), you need to request that your CAST Extend login is granted the appropriate additional contributor type (in this example, "CAST Labs").

Packaging in .nupkg format

You can only share an extension in **.nupkg** format (this is essentially an archive file) - any other formats will be rejected by **CAST Extend**. To create the .nupkg file, you should proceed as follows:

The packaging is executed with the command line utility called **NuGet.exe** - you can download this file free of charge from https://dist.nuget.org/index.html. CAST recommends using a batch script to run the packaging - this script should be run from outside the folder containing your extension. An example is shown below - please adapt this to include the name of the folder containing your extension:

```
set EnableNugetPackageRestore=true
NuGet.exe pack com.castsoftware.uc.nameoftheextension\plugin.nuspec
```

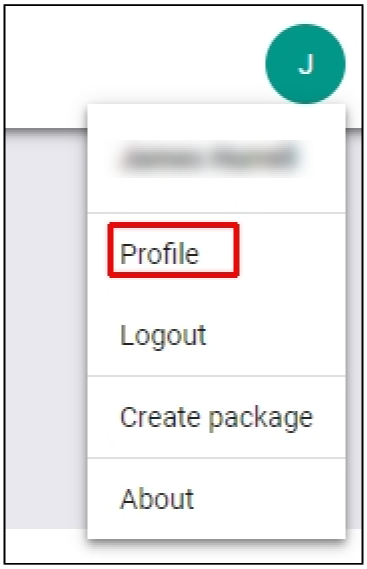
If the package action is successful, a file entitled **com.castsoftware.uc.nameoftheextension.<version_number>.nupkg** will be created. The **<version_number>** will be determined automatically by the contents of the **<version> tag** in the **plugin.nuspec** file.

The "pack" command will ignore any empty folders located in the root folder, and therefore these empty folders will not be included in the resulting .nupkg file. If you need to rely on the existence of empty folders inside your extension you need to include the "on demand" creation of that folder within the extension code. Known examples are:

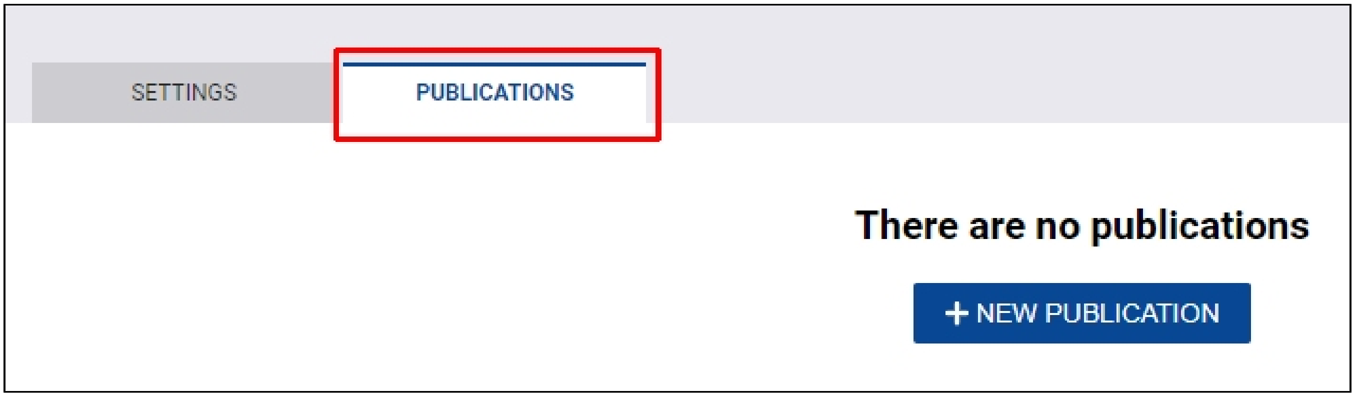
- Temporary folders for UA preprocessing, i.e.:
  - Configuration\Languages\<language>\plugin\gen
  - Configuration\Languages\<language>\plugin\results

Upload the extension via the CAST Extend website

You can share your extension by uploading it to the **CAST Extend website** - you will need to register for an account if you do not have one already. Click **Profile** under your username:

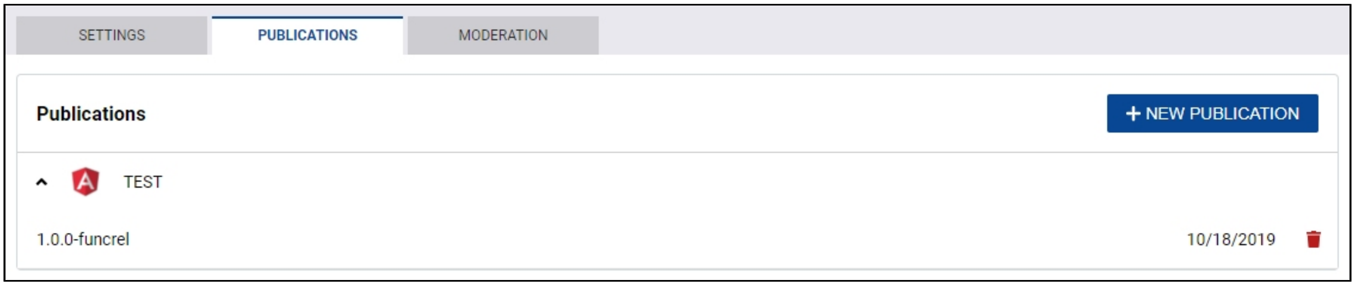


Choose the **Publications** tab:



There are no publications

Click **New Publication** to upload your .nupkg file. If your extension conforms to the required norms it will be **accepted**:



Publications

TEST

1.0.0-funcrel 10/18/2019

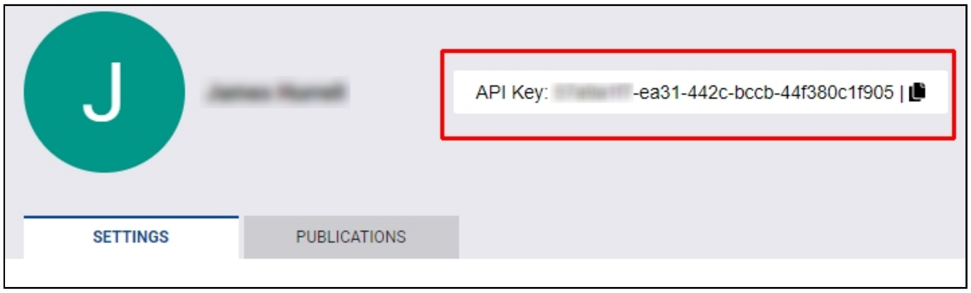
Once the extension has been successfully uploaded, a user with **Moderation rights** will need to validate the extension before it can be downloaded for use by others.

> Note that you can see all extensions that you have uploaded but that have not yet been validated: please see Manage uploaded extensions.

Upload the extension via a CLI using Nuget.exe

You can share your extension by uploading it to the **CAST Extend website** using a **command line** utility called **NuGet.exe** - you can download this utility free of charge from https://dist.nuget.org/index.html. CAST recommends using a batch script to run the upload - this script should be run from outside the folder containing your extension. An example is shown below - please adapt this to your environment:

```
nuget.exe push com.castsoftware.uc.nameoftheextension.nupkg <apikey> -Source https://extendng.castsoftware.com
/api/package
```

Option	Description
nuget.exe	Full path to the nuget.exe utility
push	Upload the extension
com.castsoftware.uc.nameoftheextension.nupkg	Full path to the extension in .nupkg format
<apikey>	Your unique API Key. This key uniquely identifies you as a registered user on the CAST Extend website. You can find your API key in you user **T Extend** website:  J [redacted] API Key: [redacted]-ea31-442c-bccb-44f380c1f905
-Source	The URL of the CAST Extend website: **https://extendng.castsoftware.com/api/package**

> Note that you can view the nuget.exe push command line parameter documentation here: https://docs.nuget.org/ndocs/tools/nuget.exe-cli-reference#push

Running the batch should result in a successful upload as follows:

```
Pushing com.castsoftware.uc.angularjs.1.1.0.nupkg to 'https://extendng.castsoftware.com/api'...
  PUT https://extendng.castsoftware.com/api
  OK https://extendng.castsoftware.com/api 4371ms
Your package was pushed.
```

Once the extension has been successfully uploaded, a user with Moderation rights will need to validate the extension before it can be downloaded for use by others.

> Note that you can see all extensions that you have uploaded but that have not yet been validated: please see **View unlisted extensions** in Manage uploaded extensions.