Google **Feed** Module

1. How to install the W**4**Plego_Google Feed extension?

Step 1. To install the module through composer, you need to add a link from the repository to Composer.json, which is located in the locally deployed project.

An example of a link from a repository

Clone this repository

SSH ∨

~~git clo~~➤ git@bitbucket.org:GitWEB4PRO/google-feed.git

VS Code is a source-code editor developed by Microsoft.

Clone in VS Code

Close

An example of adding to a project in Composer.Json

```
 87          }
 88        },
 89        "version": ████████
 90        "minimum-stability": "stable",
 91 ⚠      "repositories": [
 92           {
 93               "type": "composer",
 94               "url": "https://repo.magento.com/"
 95           },
 96           {
 97               ████████████████████████
 98               ███████████████████████████████
 99           },
 00           {
 01 ➤             "type": "vcs",
 02               "url": "git@bitbucket.org:GitWEB4PRO/google-feed.git"
 03           }
 04        ],
 05        "extra": {
 06           "magento-force": "override"
 07        }
 08     }
```

Step 2. On a locally deployed project, you need to change to the project root directory

Step 3. Create or checkout a development branch

Step 4. Using the Composer name and version, enter the command to add the W4Plego/GoogleFeed module to composer.json
Command example:
composer require <extension-name>:<version>
(composer require "w4plego/google_feed": "dev-master")

Step 5. After adding the module to composer.json, the following command is executed in turn:
bin/magento setup: upgrade
bin/magento setup: di:compile

Step 6. After we check the presence of the module on the local project

bin/magento module:status <extension-name>(bin/magento module:status "w4plego/google_feed")

Step 7. Adding, Committing, and Pushing Code Changes to Your Repository via Git

git add -A
git commit -m "Install <extension-name>"
git push origin <branch-name>

Step 8. Once the build and deployment is complete, use SSH to log in to the remote environment and verify the installed extension

bin/magento module:status <extension-name>

2. How to set up cron for a module and for a project ?

Step 1. In the remote environment, check for the presence of cron in the root of the project. Enter the command to view the availability of cron
crontab -l

Step 2. If there is no cron, execute the command to install
bin/magento cron:install
An example of a cron installed on Magento 2.4.3 version can be seen in the screenshot
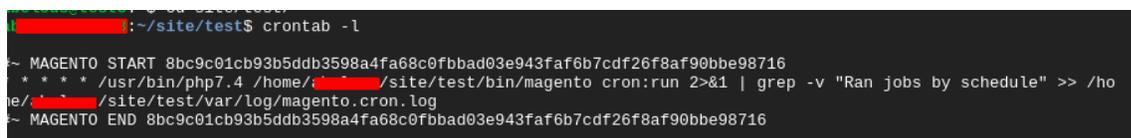

```
                :~/site/test$ crontab -l

~ MAGENTO START 8bc9c01cb93b5ddb3598a4fa68c0fbbad03e943faf6b7cdf26f8af90bbe98716
 * * * * /usr/bin/php7.4 /home/       /site/test/bin/magento cron:run 2>&1 | grep -v "Ran jobs by schedule" >> /ho
e/      /site/test/var/log/magento.cron.log
~ MAGENTO END 8bc9c01cb93b5ddb3598a4fa68c0fbbad03e943faf6b7cdf26f8af90bbe98716
```

Step 3. Check the time zone of our store or set the right time for a specific country
Finding path **Stores/Configuration/General**
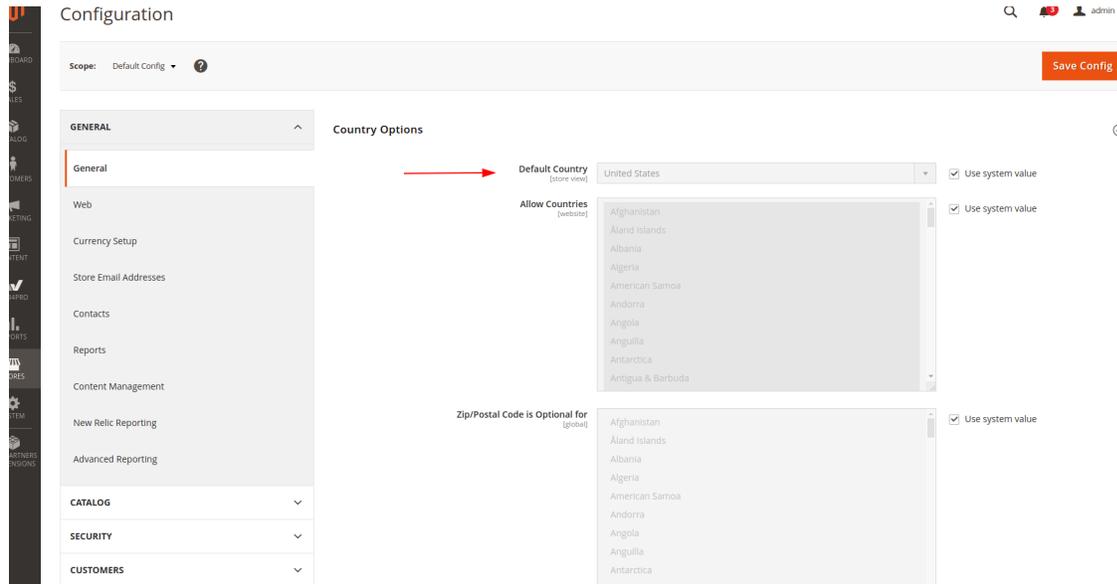
Step 4. Next, go to setting up the cron in the W4PlegoGoogle Feed module - the path to find
Stores/Configuration/WEB4PRO/W4plego Google Feed

The W4Plego Google Feed cron will update the xml file according to the time that was set in General (timezone)
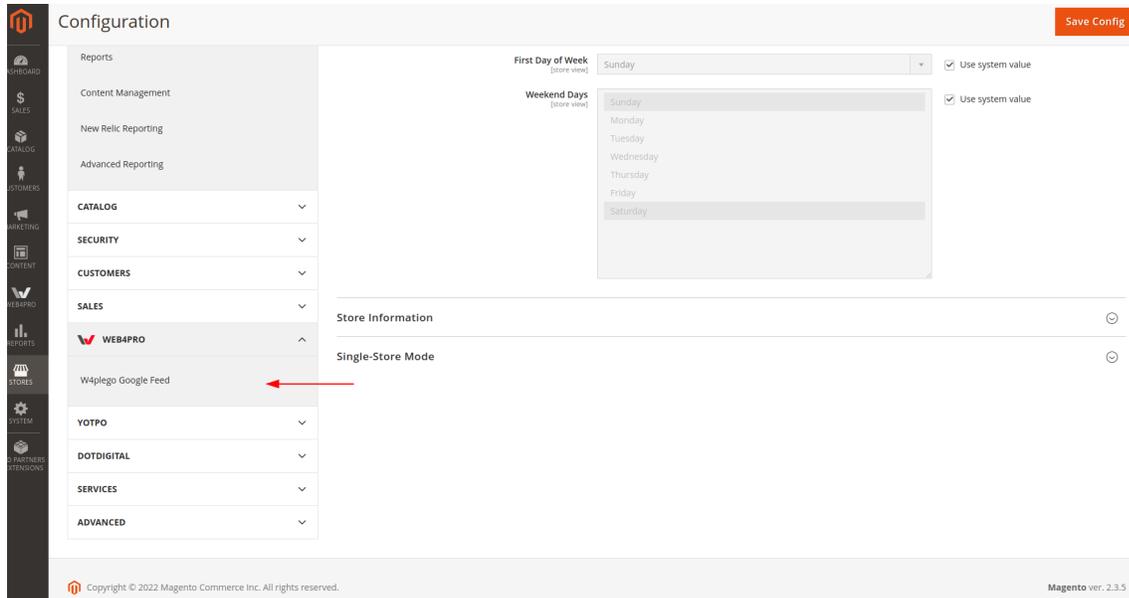
Step 5. Setting the time to update the file by cron in the W4plego Google Feed module

Turn on the cron and set the time (according to the time zone of the country that we set in the store settings) and select the update period. Next, click on the save button.

There are three fields for setting cron:
1. Turn on/off
2. Time setting (hours/minutes/seconds format)
3. Update period (with a frequency of once a day, once a week, once a month)

Step 6. After saving the cron, you need to execute several commands in the root of the project to run the cron:

bin/magento cron:run

bin/magento cache: clean

bin/magento setup: upgrade

bin/magento setup:di: compile

3. How to use the Google Feed module on the Magento side?

Click the WEB4PRO tab in the Magento side-bar -> select the "Web4Pro Google Feed" module
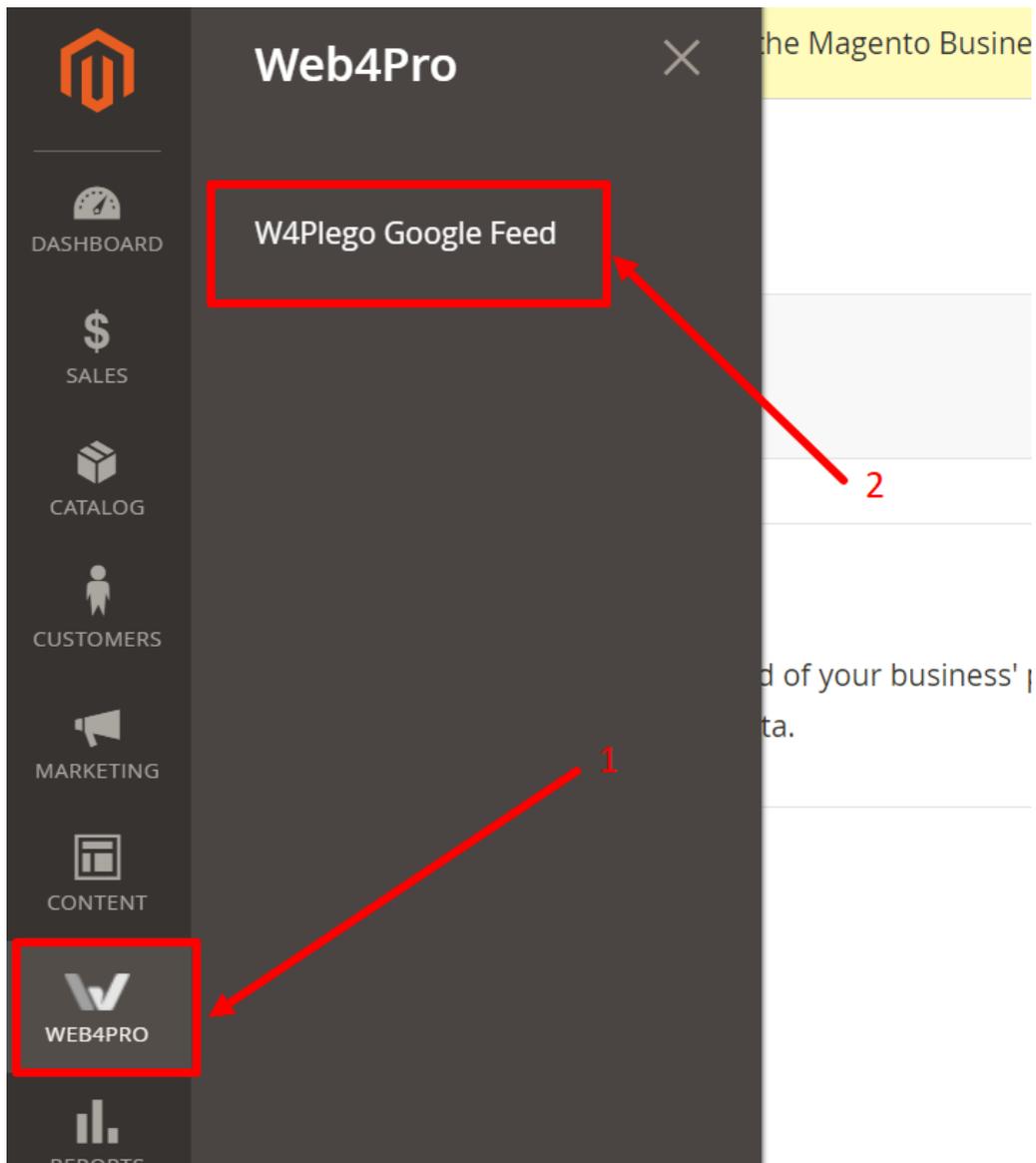
After that, you can see an empty grid (if the Google Feed module you use for the first time). For adding an entry click the "Add Feed" button.

Then you can see the page for adding a new feed. You need to enter the following data:
1) The filename in XML format (mandatory field)
2) The Path to this file (mandatory field). Example: "/media/" for base path (path must be writable)
3) Select the "Store View" (mandatory field). Each store that is available to customers is presented according to a specific view. You need to choose the Store View, for which the Google Feed file will be generated

New Feed

Q   🔔8   👤 admin ▾

← Back   Reset   **Save**

Feed

Filename *   test27.xml

example: feed.xml

Path *   /media/

example: "/media/" for base path (path must be writeable)

Store View *   EN   ▾   ❓

Do not forget to click the "Save" button. Also, you can click the "Reset" button if you do not want to apply changes. In this case, a new feed will not be created. The "Back" button is for returning to the previous page with the grid.

If the new feed was created successfully, you can see the record about it in the grid. You can see the columns with the following data about created feed:

- ID - the record number will be displayed in the grid
- Filename - it is the name of the file that was entered at the moment of feed creation
- Path - it is the path to the file that was entered at the moment of feed creation
- Link for the Google Merchant Center - this link is inactive while you are not clicking on the "Generate" button. If you open the link at this stage, you will be redirected to the 404 page
- Last generated - this column will be empty while are you not clicking the "Generate" button

- Store View - it is the Store View that was selected at the moment of feed creation
- Action - in this column you can see the "Generate" button. This button is for generating an XML file for Google Merchant Center

After clicking the "Generate" button the date of file generation will appear in the "Last Generated" column. Also the link in the "Link for Google Merchant Center" becomes active.

W4plego Google Feed

It means that the file was created in the media folder and the user can open it through the link. After opening the file you can see the following information:

General data
- RSS XMLNS - it is the default link that indicates the version of Google Merchant Center
- Title - it is the file name, that you entered on the stage of creating a new feed
- Link - it is a unique domain

Products data
- Title - it is the item name
- Link - it is the product URL. You can see this product on the site by going through the link
- Description - it is the product description

- Image - it is a link to the product image. You can see how this product looks by going through the link
- ID - it is the product SKU
- Price - it is product price in a currency that setting for selected Store View
- Availability - it is an indicator for showing product is in stock or out of stock

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<rss xmlns:g="http://base.google.com/ns/1.0" version="2.0">
  <head/>
  <title>test27.xml</title>
  <link>https://bioactiv.dockers.w4p.tech</link>
  ▼<channel>
    ▼<item>
      <g:title>Adenovirus IgG </g:title>
      <g:link>https://bioactiv.dockers.w4p.tech/en/adenovirus-igg.html</g:link>
      <g:description/>
      <g:image_link>https://bioactiv.dockers.w4p.tech/media/catalog/product/a/d/adenovirus_igg_1.jpg</g:image_link>
      <g:price>75.00 EUR</g:price>
      <g:id>ADVG0010BA</g:id>
      <g:availability>In Stock</g:availability>
    </item>
    ▼<item>
      <g:title>Adenovirus IgM</g:title>
      <g:link>https://bioactiv.dockers.w4p.tech/en/adenovirus-igm.html</g:link>
      <g:description/>
      <g:image_link>https://bioactiv.dockers.w4p.tech/media/catalog/product/a/d/adenovirus_igm_1.jpg</g:image_link>
      <g:price>75.00 EUR</g:price>
      <g:id>ADVM0010BA</g:id>
      <g:availability>In Stock</g:availability>
    </item>
    ▼<item>
      <g:title>Chikungunya Virus IgG capture</g:title>
      <g:link>https://bioactiv.dockers.w4p.tech/en/chikungunya-virus-igg-c.html</g:link>
      <g:description/>
      <g:image_link>https://bioactiv.dockers.w4p.tech/media/catalog/product/c/h/chikungunya_virus_igg_capture_1.jpg</
      <g:price>300.00 EUR</g:price>
      <g:id>CHIG0590</g:id>
      <g:availability>In Stock</g:availability>
    </item>
```

NOTE: All products existing in Magento are exported in the XML files. For example, if in Magento 1678 products, you will see 1678 products in the XML file - ? need to check

NOTE: All types of products are exported in the XML files - Simple Products, Grouped Products, Configurable Products, Virtual Products, Bundle Products, and Downloadable products

For deleting some feed you need to click on it in the grid. After that, the Feed page will be opened, and you can click the "Delete" button. Then, the "You deleted the feed" message should be displayed in the grid, and entry with feed should disappear from the grid.

For convenience, users can filter entries in the grid. It helps faster and simpler find necessary entry. Filters can be applied by the Last Generated date and/or by the Store View. For applying filter user need to select necessary parameters and click the Search button. For removing applied filters you need to click the "Reset Filter" button.

W4plego Google Feed

Search   Reset Filter   1 records found   20 ▾ per page   <   1   of 1   >

ID	Filename	Path	Link for Google Merchant Center	Last Generated	Store View	Action
				From 📅	DE ▲	
				To 📅		
6	test28.xml	/media/	https://bioactiv.dockers.w4p.tech/media/test28.xml			

Main Website
   Main Website Store
      EN
      DE
usa
   USA
      English
      Deutsch
Germany
   Germany
      Deutsch

Add Feed

DASHBOARD
MANAGE PDF TEMPLATES
LANDOFCODER
SOLWIN
FREEGO THEME CPANEL
SALES
CATALOG
CUSTOMERS
MARKETING

admin