# falconG Responsive Gallery generator

Version 1.1

## description

The falconG program generates a multi-language, responsive (mobile first) hierarchical WEB gallery from

1. from a special gallery structure file and a series of images.
2. any directory (folder) and its sub-directories which contain JPG, PNG, GIF, BMP image files and helper files describing the order, the title and the description of these images. It can process a JAlbum ([www.jalbum.net](http://www.jalbum.net)) generated gallery source[[1]](#footnote-1). OR

The user can specify all aspects of the galleries.

The resulting gallery may contain a virtual hierarchy of image galleries for all of the specified languages, Google Analytics code, Facebook like/share links can be added to the galleries and watermarks to the images.

## features

1. Generates a multilingual, hierarchical and embeddable responsive photo gallery with freely modifiable colors and fonts.
2. Any number of languages may be used. Generates one HTML file for each languages, so the download only contains texts from one language.
3. Each album and image may have a title, which is always displayed and a description which is only shown when requested for each languages. (Image and gallery names are not the original file or directory names.)
4. The hierarchy is virtual as each HTML files are in the same physical directory on the server and the same is true for all image and thumbnail files as well.
5. A simple index file (named *\_index.html*) is generated for each languages, but you can use your own index page and can put the gallery in an IFRAME.
6. The number of images and sub-galleries are only restricted by free disk space, although with tens of thousands of images or sub-galleries the access speed may be low (this depends on the operation system.)
7. A thumbnail image is generated for each source image. These thumbnails are smaller to make download faster and to use fewer resources. The user can specify image and thumbnail dimensions. Thumbnails are always resized, but images may be marked as not to be resized.
8. Only one copy of each image and thumbnail is stored on the server.
9. Any image may appear in any number of galleries. In all galleries it will have the same title and description text.
10. A watermark may be applied to all images in a user specified font, color, transparency and position.
11. The user may set a Google analytics code for each pages
12. The galleries can link to externally created ‘About’ pages.

## How to Use

You can create a gallery

1. from a hierarchical directory structure with images and special description files (e.g. from the file the program *Jalbum* creates), or
2. using a single gallery description file (*gallery.struct)* which contains all language album and image data (In this case the image file can be anywhere on the source machine) or
3. using only the GUI (not yet works)

You may set the path to the **source gallery** and the **destination gallery** which must be different from the source gallery. The content of the destination gallery is the one to upload to the server machine.

To generate a gallery for any number of languages you need either language files or an existing UTF-8 encoded *gallery.struct* file with all these languages. Other text files should be either native or UTF-8 encoded. The resulting files are always UTF-8 encoded.

An American English language file called *en.lang* is provided in the program directory and you can translate its content to any language you need[[2]](#footnote-2). The names of the language files are not important, but its eextension must be ‘*.lang*’. Each language file must start with the line: *falconG Language file* followed by lines in format *name=text.* Only the *text* part should be translated. Please look up your country code on the net and please translate the copyright text too[[3]](#footnote-3). The order in which language files are searched for is:

1. source directory
2. destination directory or
3. program directory

Do not use more *.lang* files then the number of languages you want to use! For instance if you only want to use German, then only put the single *de.lang* file into the source directory.

The next step depends on whether you want to use

1. A single UTF-8 coded *gallery.struct* file, or
2. a hierarchical directory structure with images and optionally text files for file ordering, image/album title and description.

### Case #1

The *gallery.struct* file must be encoded in UTF-8 and it can be created by any UTF-8 aware text editors. Do not use document editors (Word, OpenOffice, etc) to edit your files!

The *gallery.struct* file contains the common language texts and the names and paths of all albums and images. Paths can be relative to the source directory set in the GUI or can be absolute paths[[4]](#footnote-4). Any part of any line starting with the character ‘#’ is a comment and will not be used (except for the first line).

The *gallery.struct* file contains the following sections:

Header section  
Language section  
Album section

*Header section*

starts with the file type line

*# falconG Gallery Structure file 1.2*

followed by a copyright line, e.g.

*# © - András Sólyom (2019)*

followed by the source and destination directory as a comment

*Language section*

contains blocks for the compulsory texts. starts with

*[Language count:<number of languages>*

(no closing brace!) followed by the language blocks.

Each block starts with a line with the number of the block (1,…) at the first character position, followed by the translation lines and ends with a ‘]’ at the first character position

Example:

[Language count:2

0

abbrev=en

name=English

icon=

images=Images

albums=Albums

toAlbums=To Albums

homePage=Home

about=About

contact=Contact

captions=Captions

share=Share

latestTitle=Sampling of Most Recent Uploads

latestDesc=This gallery contains some of the latest uploads of this site. Of course all photographs presented here can be found in their corresponding galleries too. You just have to find them...

countryCode=en\_US

countOfImages=%1 image(s) and %2 sub-album(s) in this album

falconG=Site generated by <strong>falconG</strong> - © A.Sólyom 2018

1

abbrev=hu

name=Magyarul

icon=

images=Képek

albums=Albumok

toAlbums=Albumokhoz

homePage=Főoldal

about=Rólam

contact=Kapcsolat

captions=Képaláírás

share=Megosztás

latestTitle="Kedvcsináló"

latestDesc="Ebben a galériában található néhány a legutoljára feltöltött képeim közül, amelyek természetesen a saját galériájukban is megtekinthetőek. Persze azt meg kell keresni..."

countryCode=hu\_HU

countOfImages=%1 kép és %2 album van ebben az albumban

falconG=Site generated by <strong>falconG</strong> - © A.Sólyom 2018

]

*Album section*

The structure of the album section is different for new (hand made) and processed files. What is common though:

* hierarchy is denoted by the indentation level of lines, i.e. the number of SPACE characters before the text starts. For instance image lines or sub album lines in an album start with one space more at the front then the album line has
* album lines are separated from each other by an empty line
* title, description and icon lines are at the same level as the album or image they belong to.
* title, description and icon lines are optional. (icon lines are only used with albums!)

Example #1   
hand created agallery.struct file (descriptiona in *blue italics*) with two languages: English and Hungarian.:

# Album structure:

(A:1) *album ID line for root album*

[Title-en*:<title in English>*]

[Title-hu*:<title in Hungarian>*]

[Descr-en*:<descriptin in English>*] *arbitrarily long line. Put*

[Descr-hu*:<description in Hungarian>*] *\n in the text for line feed*

[Icon:<path of icon image>] relative or absolute path>]

*relative/path/of/image.jpg notice the space at the start!*

relative/path/of/second\_image.jpg

[Title-en*:<title in English>*] *previous image has no title*

[Title-hu*:<title in Hungarian>*] *and this one has no description*

empty line before first sub-album

path/of/first/sub-album *at the same level where images*

[Title-en*:<title in English>*] *were in the parent album*

[Title-hu*:<title in Hungarian>*]

*relative/path/of/another/image.jpg notice the 2 spaces before it!*

relative/path/of/another/!!notResized\_image.jpg *!! : do not resize*

In this example the name of the first sub album is *sub-album*. This has nothing to do with a real name of a directory. It may be one and it may not, and in fact will not be used as the title is used instead. The two exclamation marks at the beginning of the image name signals that the image will not be resized.

Example #2

The same album as before but now with concrete values. The top level album has no name , the one below it have no images, just sub albums and the first sub album has two images in it. (Because of WinWord the long lines are wrapped here, but they are one line in the file):

# Album structure:

(A:1)

[Title-en:Andreas Falco Photography]

[Title-hu:Andreas Falco Photography]

Magyarorszag

[Title-en:Hungary]\*3634818861

[Title-hu:Magyarország]

[Descr-en:Hungary is a relatively small landlocked country in the Pannonian Basin, (or Carpathian Basin) in Central Europe]]

[Descr-hu:Hazánk Európa 18. legnagyobb országa]

[Icon:Magyarorszag/Gyor/7\_05720\_1\_2-HDR.jpg]

Magyarorszag/Budapest/Matyas templom/A magyar korona

[Title-en:The Hungarian crown]\*299965539

[Title-hu:A magyar korona]

[Icon: Magyarorszag/Budapest/Matyas templom/!!Cr\_4402full.jpg]

Magyarorszag/Budapest/Matyas templom/!!CrownCopy\_4402full.jpg   
 Magyarorszag/Budapest/Matyas templom/!!CrownSceptreApple\_4406.jpg

Example #3

Processed album from Exaple #2

# Album structure:

(A:1)

[Title-en:Andreas Falco Photography]\*2014183073

[Title-hu:Andreas Falco Photography]

[Icon:3114355225]

Magyarorszag(A:4131623263)

[Title-en:Hungary]\*3634818861

[Title-hu:Magyarország]

[Descr-en:Hungary is a relatively small landlocked country in the Pannonian Basin, (or Carpathian Basin) in Central Europe]]\* 1267573459

[Descr-hu:Hazánk Európa 18. legnagyobb országa]

[Icon:3127873579]

A magyar korona(A:3175693179)Magyarorszag/Budapest/Matyas templom/

[Title-en:The Hungarian crown]\*299965539

[Title-hu:A magyar korona]

[Icon:1220531141]

!!Cr\_4402full.jpg(1220531141,3504x2332,3504x2332,2019-03-10,754239)Magyarorszag/Budapest/Matyas templom/

!!CrownSceptreApple\_4406.jpg(3575104234,3128x2346,3128x2346,2019-03-10,569127)Magyarorszag/Budapest/Matyas templom/

### Case #2

Source images (possibly together with text files for file ordering, image/album title and description) are already in a hierarchical directory system. If you have downloaded the program JAlbum when it was still free for personal use you can use it to create all sub-directories and all text files. If you don’t have it just put the images into a hierarchy and do not care about the texts. When you run falconG to process a directory without the *gallery.struct* file it will create one and you can add the texts later to it[[5]](#footnote-5).

## File Structures

1. AFAIK there are no multi-language JAlbum gallery, but if you delimit your text for the different languages with two consecutive ‘@’ characters (example: @@English Text@@Dautcher texte@@Texto en alemán@@), then falconG will generate separate HTML files for all languages. (You must also have the corresponding .lang, files.) [↑](#footnote-ref-1)
2. If you plan to use different language sets to different galleries then it makes sense to put the corresponding *.lang* files into the source gallery directory. [↑](#footnote-ref-2)
3. I respectfully ask you to *not replace* the copyright text with one of your own. [↑](#footnote-ref-3)
4. On windows these starts with the drive (e.g. C:\), on linux these start with ‘/’ [↑](#footnote-ref-4)
5. Checking to checkboxes you make falcon to generate empty text holders for all languages. You only need to fill them in. [↑](#footnote-ref-5)