

GalleryNET – An open source online gallery project



Created by: imageComponent.NET

Last updated: June 28, 2008

Overview

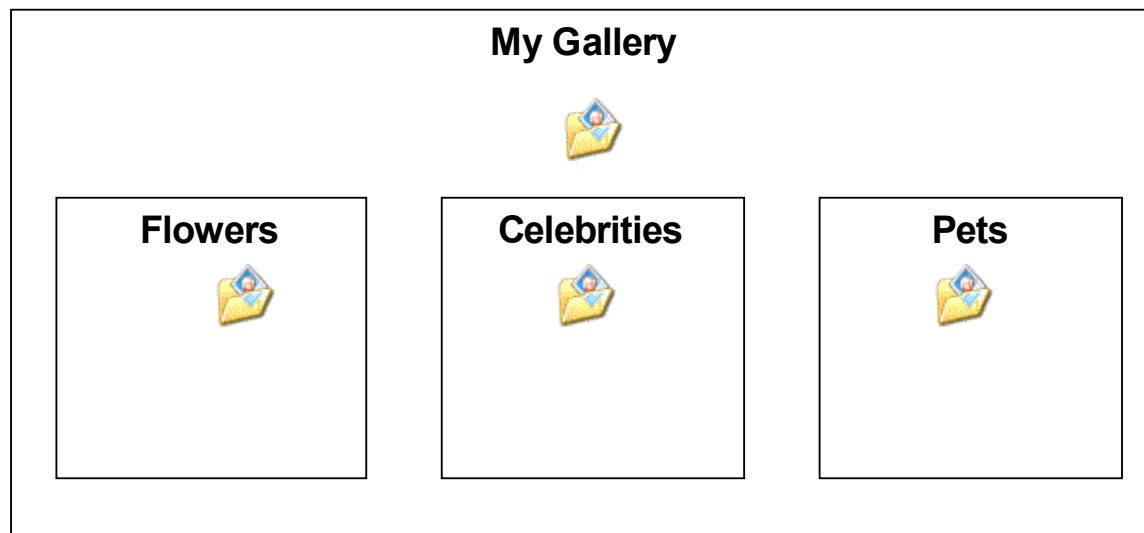
GalleryNET is a free and open source project for creating online Gallery. It's written in Microsoft © C# and ASP.NET. It provides a certain level of flexibility for the end-users to customize the display style. The most exciting feature is, once the template and style sheet files are done, the end-users can simply upload additional galleries and image files to the web-server via FTP. Without writing any code, the new files will be automatically populated by the control.

GalleryNET borrows the LGW (Lightbox Gone Wild) idea from Kevin Hale. (<http://particletree.com/features/lightbox-gone-wild/>) Thanks a lot to his work. I made some modification based on the comments from that page; also, I added a new feature to position the popup.

Introduction

Before describing its feature, please allow me to clarify the term “Gallery” in this system in order to gain better understanding.

A Gallery is referred to a file system. It contains folder, subfolders, and files. Use the next figure as an example:



This gallery [**My Gallery**] has three sub-categories: [Flowers], [Celebrities], and [Pets]. **GalleryNET** is able to retrieve the file system structure and automatically draw the whole gallery.

In this document and the code comments, when you see sub-categories, sub-folders, or sub-galleries, they are all the same thing.

By using the template, the users can easily configure information to display. Please see [**Gallery Data**] section for more details.

Release Notes

2008/06/29 – Version 3.00a is released. Built with VS.NET 2008.

2005/01/07 – Version 2.2.

2004/11/11 – Version 2.1.

2004/06/09 – Version 2.0.

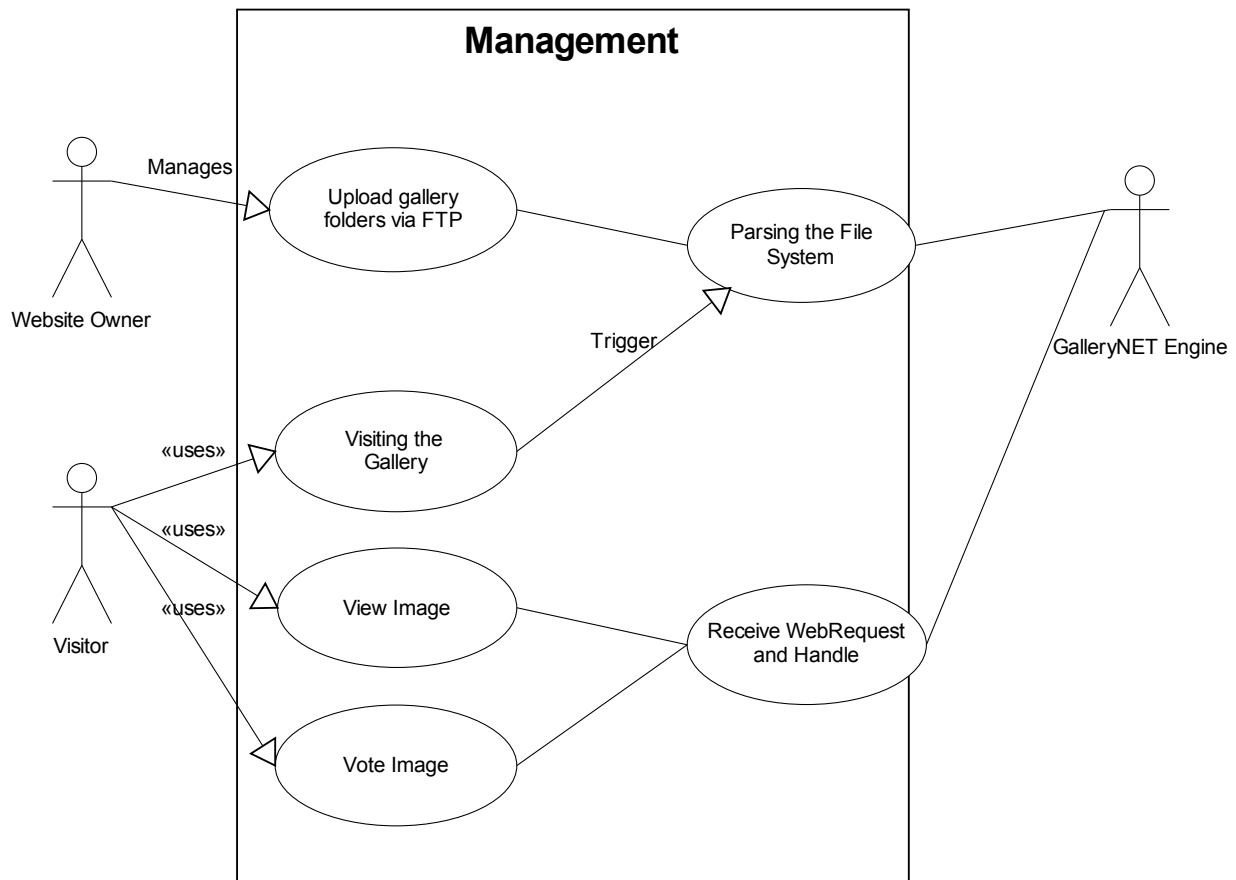
2002 (or 2003) – Version 1.0

A little history

GalleryNET originally was an ISAPI application written in Delphi and it worked with ASP. I did publish that to a website, but it ended up not too many people was aware of that. The version 1.0 was similar to version 2, which basically require only a few lines of code in the front-end to render the whole gallery based on some html template files. I remembered I didn't see any other similar products at that time. I might be wrong. ☺ Version 2 was written completely in C# and worked for ASP.NET 1.x. . I did find some people used version 2 online; I appreciate them. When I looked the source today, the control was poorly written and very inflexible. But some ideas are not bad, so I inherit them to version 3 .

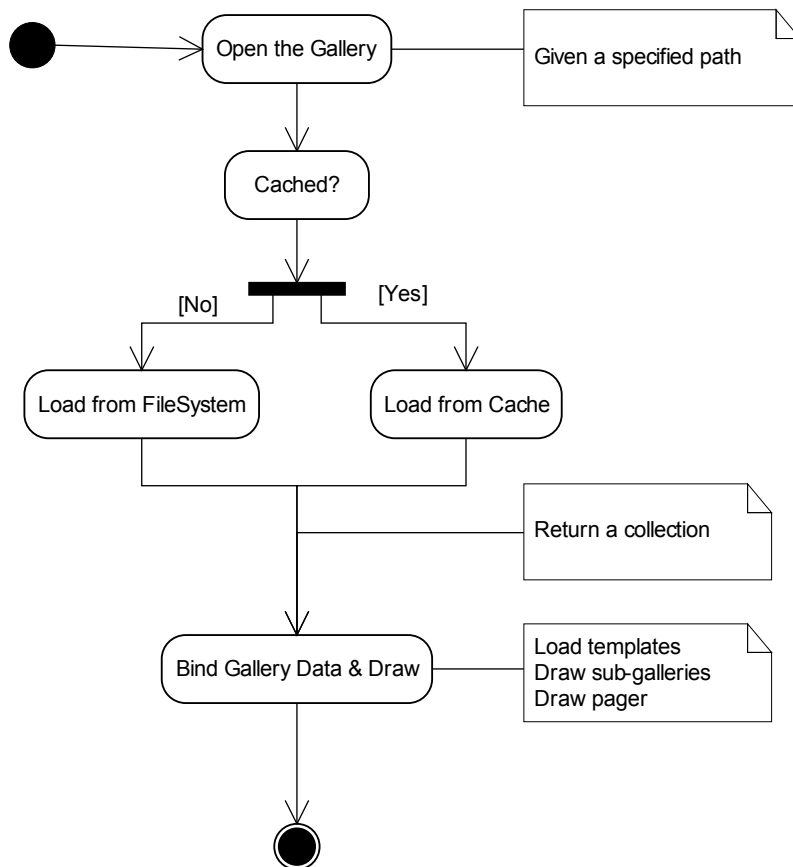
Use Case

Here is a very simple use case diagram to explain what will happen after the **GalleryNET** site is setup.



Work Flow

The following diagram describes briefly how GalleryNET draws the control.



A Brief Introduction

In your aspx file, first thing is to declare the control:

```
<%@ Register Assembly="IC.Web.GalleryNET"  
Namespace="IC.Web.GalleryNET.Controls" TagPrefix="cc1" %>
```

Then you need to place the control:

```
<cc1:GalleryGrid BorderStyle="Dashed" ID="galleryGrid" runat="server"  
RepeatColumns="3" RepeatDirection="Horizontal" SubFolderDisplayMode="Complex"  
AllowPaging="true" PageSize="9" CellSpacing="0" CellPadding="3"  
HorizontalAlign="Center" Width="640" PagerMode="NumericPages"  
ErrorMessageCssClass="errorMessage">  
    <pagerstyle cssclass="pager" />  
    <subfoldertablestyle cssclass="subfolder" />  
</cc1:GalleryGrid>
```

Please refer to the development guide for more details on each property. In this example, it specifies:

- use “Complex” mode to display the subfolders;
- allow paging
- each page has maximum 9 images
- display numeric page indices
- and some css classes for different blocks

According to the work flow diagram, the grid control will see if there is a specified path, if no, it will look up the web.config and use that one as the root path (see later section for details). Then it will cache the file system data. The DataSource will be a collection of GalleryImageInfo. The grid will also load the user-defined templates to display customized layout.

Behind the scene, GalleryGrid is a System.Web.UI.WebControls.DataList, and it will always turn on Footer and Header, unless you disable Paging and subfolder display. GalleryImage is a CompositeControl. It's a template-based control, but instead of writing a data bound control, I used a different technique to render data. It's easier to code and buildup the template, the drawback is the developer cannot use code-insight of VS.NET. Please refer to ShowImage.aspx for more details.

Web.Config

GalleryNET requires some entries in the web.config file. First, you need to add a config section as follows:

```
<configSections>  
    <section name="GalleryNET"  
type="IC.Web.GalleryNET.Configuration.GalleryConfigHandler, IC.Web.GalleryNET"  
>  
</configSections>
```

This declares to use the GalleryConfigHandler class in the IC.Web.GalleryNET assembly to load all configuration settings.

Next you need to add the details of the settings:

```
<GalleryNET>
  <settings>
    <add name="templatePath" value="templates" />
    <add name="thumbnailWidth" value="120" />
    <add name="thumbnailHeight" value="120" />
    <add name="galleryPath" value="gallery" />
    <add name="enablePicLens" value="false" />
  </settings>
</GalleryNET>
```

templatePath

This entry specifies the template folder. GalleryNET supports to load template dynamically. Separating template files from the control page will be easy to manage and code.

thumbnailWidth

This entry defines the thumbnail image width.

thumbnailHeight

This entry defines the thumbnail image height.

galleryPath

This entry specifies the gallery root folder. It should be a folder under your website virtual directory.

enablePicLens

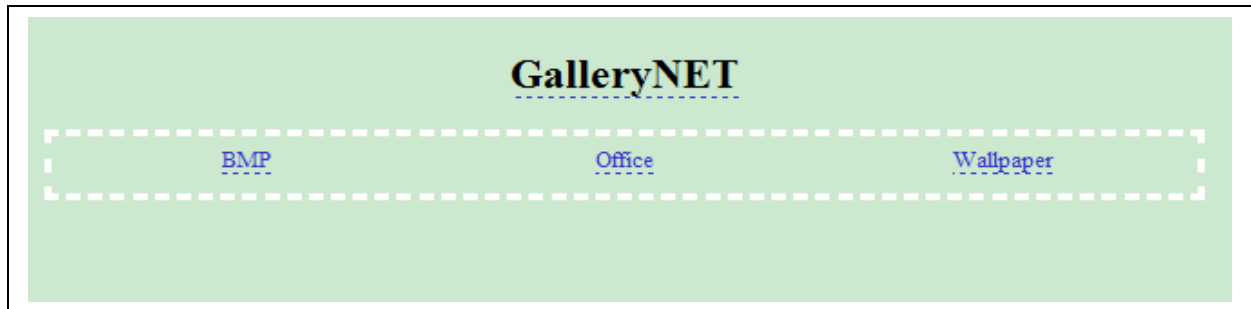
This entry specifies if turning PicLens on in GalleryNET. It's currently having a bug (see details later). see <http://www.piclens.com/> for more details of PicLens.

Gallery Data

Subfolder Information

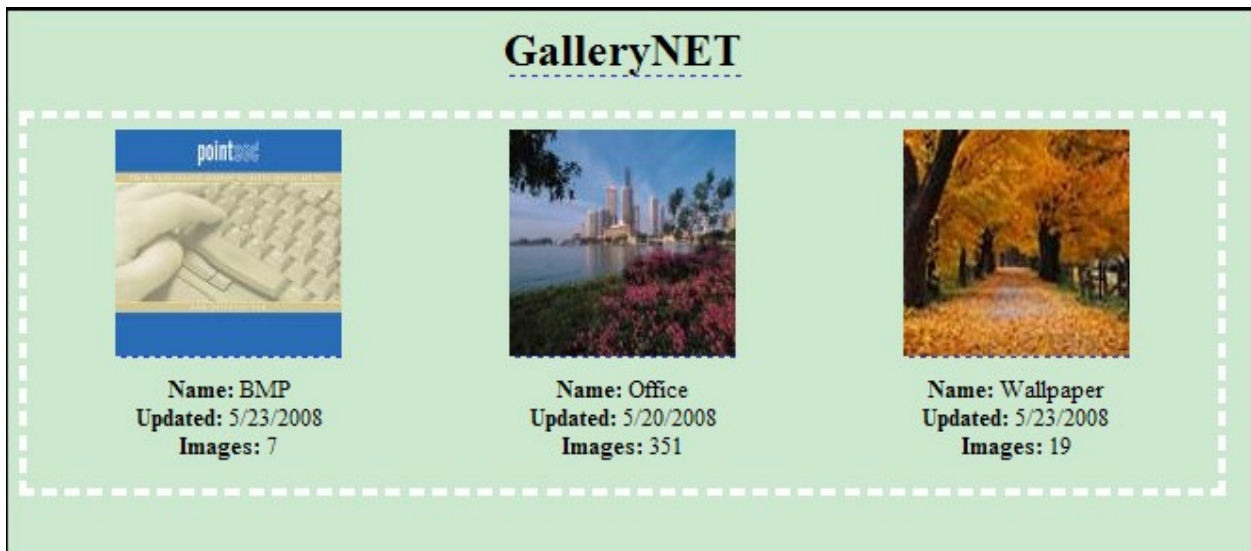
Subfolder is referred to a category in your main gallery. In GalleryGrid control, you can define two modes to display subfolders. [Simple] and [Complex] mode. Unfortunately, in the current version (3.0 a), the user cannot customize what information to display.

In simple mode, only the folder name will be shown as follows:



Clicking the name will lead you to the details of the corresponding sub-gallery.

In complex mode, a random thumbnail will be picked and displayed as follows:



The system will automatically print name, last updated date, and the number of images underneath.

The customization of subfolder display is pending.

GalleryImageInfo

Each image in the gallery is a GalleryImageInfo object and each object has the following properties:

- CreateTime
- Description
- FileSize
- FullFileName
- ImageHeight
- ImageType
- ImageUrl
- ImageWidth
- PixelFormat
- Rank
- SimpleFileName
- ThumbnailFileName
- ThumbnailUrl
- TotalRatings
- ViewedTimes

Please refer to the development document for details of each of those properties.

Paging

GalleryGrid supports 4 modes of paging by assigning different value to the property **PagerMode** property. You can also specify a css class to use a different display style via defining the **PageStyle** property.

NextPrev

Display only next and previous buttons. Good for a gallery with only a few pages.



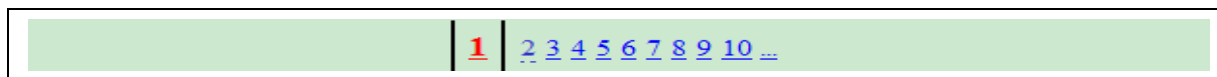
FullNextPrev

Display first, last, next, and previous buttons.



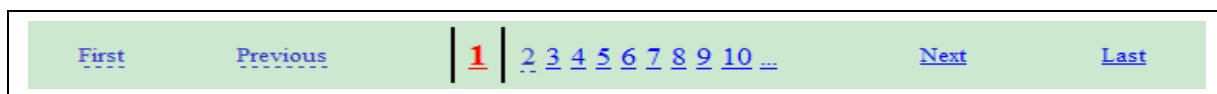
NumericPages

Display numeric page index.



Combination

Display both buttons and numeric page indices



Known Bugs

GalleryNET currently has the following known bugs:

1. **Image popup dragging javascript doesn't work in FireFox .**

The AJAX popup image panel cannot be dragged in FireFox. The script should be correct. It works when I used it in a non-GalleryNET scenario. But with all other codes, it seems in FireFox, the style.top is readonly at that point. It's very weird. I hope someone else is able to help me out.

2. **PicLens cannot be activated.**

I tried to enable PicLens for GalleryNET, because it's really cool. I followed the instruction and created the rss file. The file should be correct, because I tested it in an HTML file and it did work properly. But with GalleryNET scenario, PicLens just cannot be activated.

3. **Loading a gallery with many files is time-consuming.**

This is a performance issue. I don't know if I should call it as a bug, because loading images from the disk is an IO operation, which is always time-consuming. I did a test with 360 images under a subfolder. To open this gallery took around 7 seconds. The second time opening will be much faster because after the first time loading, I store the whole collection to the cache. I am not sure if it's a good idea or not. The most significant drawback is the users cannot see the data change, including viewed times and voting until the cache is time-out.

License

GalleryNET follows Open Group Test Suite License. Please refer to the website (<http://www.opensource.org/licenses/opengroup.php>) for detailed description.

Anyone is welcome to modify the source code or this document.

Furthermore, if you feel happy of this project and decide to run it, please add the following link to your page so that more people can be aware of this project. Hence, hope that we can have more people involved to improve it.

[Website Link Html]

```
<div style="text-align:center;margin:5pt; font-family:Calibri; font-size:medium;">This free open source project is provided by <b><i><a href="http://gallerynet.imagecomponent.net">GalleryNET</a></i><b>.
```