



## Administrator & Developer's Guide



# Contents

## **2** Installation

2 Installing SiteManager 2014

## **4** Understanding Channels & Users

7 Creating a Channel

8 Privacy

10 Example 1: Creating a Member Area

12 Example 2: Creating a Client's Area

## **14** Creating a User

## **15** Setting-up Approval Workflow

16 To Approve or Decline a Page

## **18** Site's Language & Settings

19 Language Files

## **20** Mini Sites

20 Creating a Mini Site

## **25** Template Development

29 Built-in Templates

31 Template Files

31 Adding a New Template Manually

32 Applying Your Template

35 More on Template File  
(default.master)

## **38** Module Development

38 Creating a Module

38 Embedding a Module

40 Creating an Advanced Module

## **42** Listing Template Development

44 Creating a Listing Template

57 Listing Layout

57 Listing Types:

Calendar-based & General Listing

63 Listing Entries

64 Listing Script - Using JQuery

68 Script Templates

71 Listing Cataloging

## **72** Support is Available

# Installation

## Installing SiteManager 2014

SiteManager 2014 can be installed on most shared hosts or dedicated servers that support ASP.NET 3.5. Below are the general installation instructions for a shared host.

### A. Database Setup

1. Create a blank database in MSSQL 2005/later.
2. Run **storeproc.sql** in MS SQL Management Studio against your newly created database.
3. Run **ASPNET\_REGSQL.EXE** in the .NET framework 2.0 folder:

```
C:\WINDOWS\Microsoft.NET\Framework\v2.0.x\ASPNET_REGSQL.EXE
```

Note that you don't need to run ASPNET\_REGSQL.EXE directly from your web server. You can use any computer running .NET framework v.2.0 which has a connection (via internet or local network) to your database server. The wizard that comes up will guide you through the setup.

### B. Website Setup

1. Unzip the SiteManager package.
2. Copy/ftp all files under the root of your website so that they can be accessed from:  
`http://YOURSERVER/`
3. Grant ASP.NET "write permission" to the **resources** folder.
4. Open file **web.config** using your preferred text editor.
  - a. Update the database connection setting in the web.config file.  
For example:

```
<connectionStrings>
  <add name="SiteConnectionString"
        connectionString="Data Source=DATABASESERVER;
        Initial Catalog=DATABASENAME;
        uid=john;pwd=123;trusted_connection=false;"
        providerName="System.Data.SqlClient"/>
</connectionStrings>
```

- b. Specify your mail settings. For example:

```
<system.net>
  <mailSettings>
    <smtp from=info@sitemanager.com>
    <network host="mail.sitemanager.com"
      password="password"
      userName=info@sitemanager.com />
  </smtp>
</mailSettings>
</system.net>
```

- c. Choose a folder in your web server and grant "write permission" to it. Then set the FileStorage setting in **web.config** with the folder location. For example:

```
<add key="FileStorage" value="D:\secure_storage"/>
```

Note: For secure storage, we recommend that you choose a folder on your hosting server that is not accessible through HTTP.

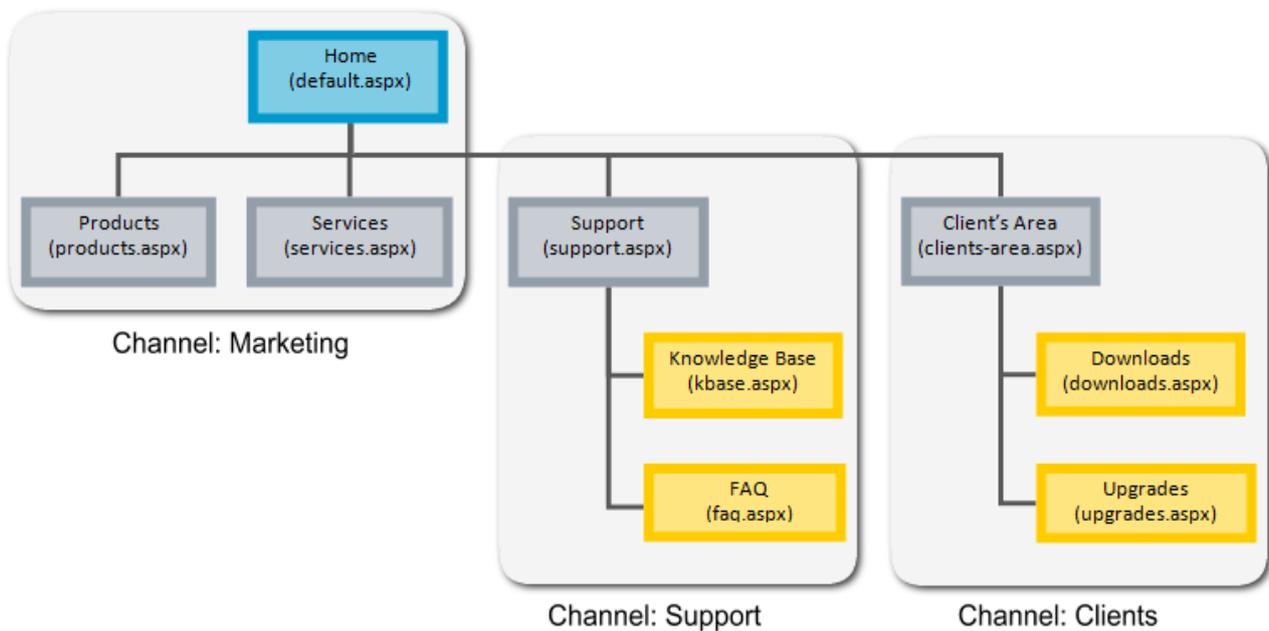
### C. Finalizing

1. Open <http://YOURSERVER/setup/default.aspx> where you can specify admin password.
2. Remove the file <http://setup/default.aspx> from your server.
3. Installation is complete. Open <http://YOURSERVER>.

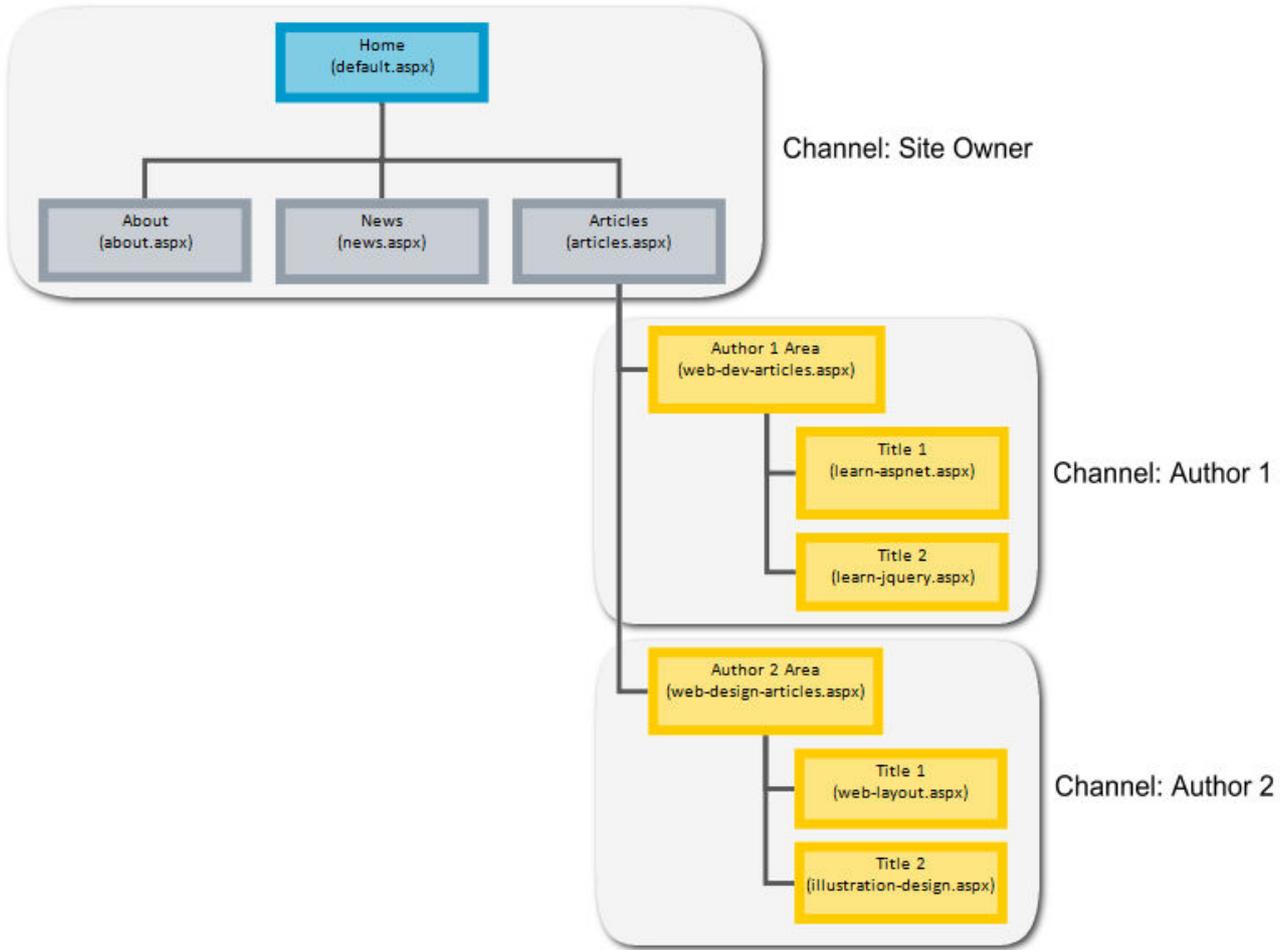
# Understanding Channels & Users

In SiteManager, content/pages are organized into channels. Channels can be created based on departments in your company or organization. For example, you (as an Administrator) can create channels such as “Accounting”, “IT Department”, “Support”, etc. You can even create channels for your customers or clients.

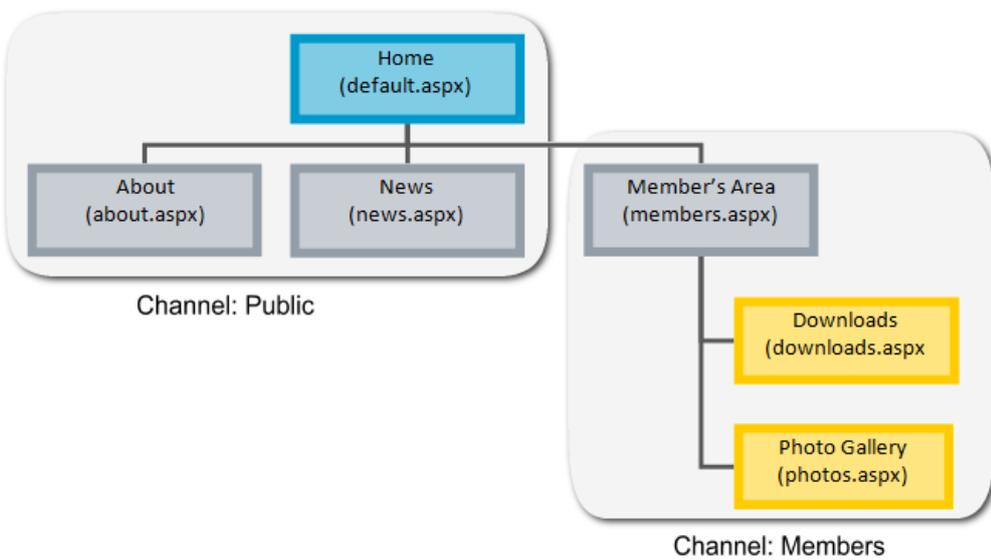
Below is an illustration of a site structure with its channels:



In an Article Portal, for example, you can create a channel for each author so that each author has its own area in which to post articles.



In a Club Website, for example, you can create a channel for Public or Members only.



When creating a channel, the system will automatically create 6 roles for the channel to which users can be assigned. For example: if we create a channel named “**Members**”, the following 6 Roles are created for it:

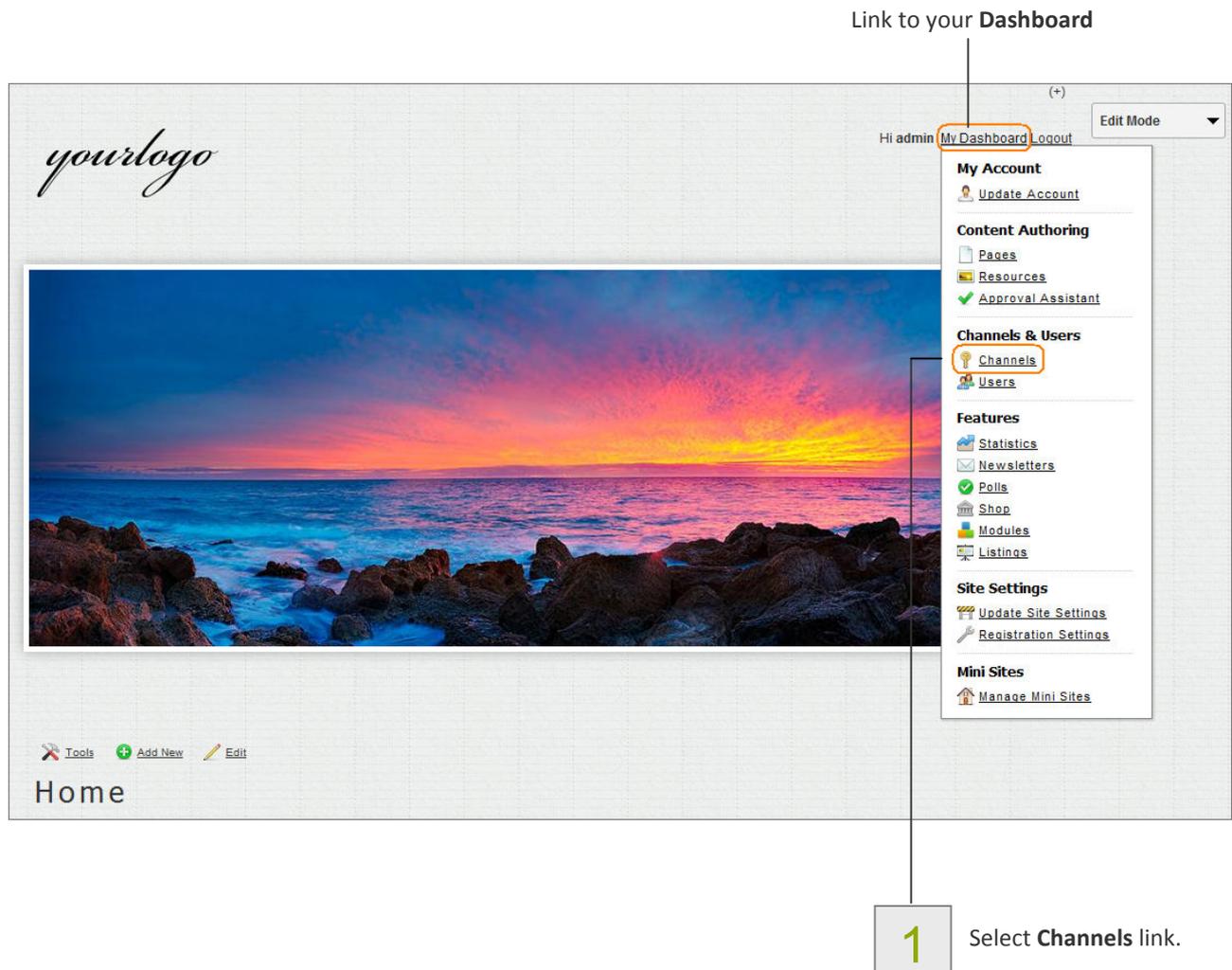
<b>Members Subscribers</b>	Subscribers have read-only access to the content of the channel.
<b>Members Authors</b>	Authors can contribute content to the channel (Create a Page, Edit a Page, and Delete a Page).
<b>Members Editors</b>	Editors are the first tier of the approval workflow. Authors must submit their work for approval. An editor may approve or reject content submitted by Authors.
<b>Members Publishers</b>	Publishers are the top tier of the approval workflow. A page approved by an Editor must go to the Publisher(s) for final approval and publication; if no one is assigned to this channel, work approved by an Editor is published directly to the site.  <b>Note:</b> If no Editor or Publisher is assigned, Authors may publish their work directly to the site.
<b>Members Resource Managers</b>	Resource managers are able to upload resources (images, media files, documents, etc.) to the channel.
<b>Members Module Managers</b>	Module managers are able to embed modules.

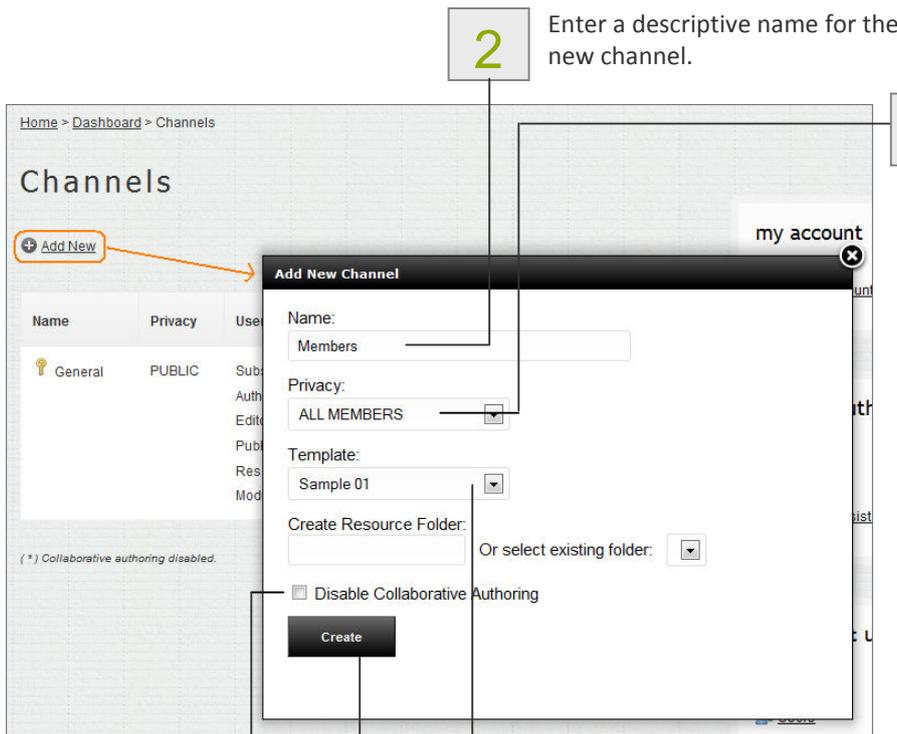
Because users are assigned to specific channels, it’s easy to give site visitors access to protected areas they need to see, while keeping them locked out of areas where they don’t belong.

Note that a user may have more than one role.

# Creating a Channel

To create a channel, hover your **Dashboard**, then select the **Channels** link.





2 Enter a descriptive name for the new channel.

3 Specify who will be able to access/view the channel content by choosing from the Privacy drop-down list. There are 3 options under privacy for *each* channel:

- Public; will make the channel content accessible to the public.
- All Members; will make the channel content accessible to all registered users.
- Channel Members Only; will make the channel content accessible only to users who are assigned rights to this specific channel (users that have one of the 6 channel's Roles listed above).

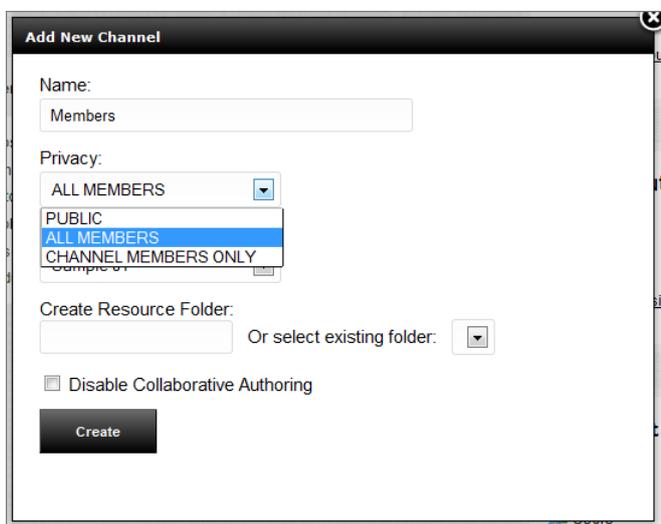
5 Tick the **Disable Collaborative Authoring** checkbox if you want only the user who creates a page to be able to edit that page.

4 Choose a template for the channel from the drop-down list. Templates are pre-existing documents that define the layout and overall appearance of your web pages.

6 Click the **Create** button to add your channel to the system.

## Privacy

When creating a channel, you have three options to allow public (1) or restricted access (2) to the content within the channel.



The 3 types of channel access privacy are:

- Public; will make the channel content accessible to the public.
- All Members; will make the channel content accessible to all registered users.
- Channel Members Only; will make the channel content accessible only to users who are assigned rights to this specific channel (users that have one of the 6 channels' Roles listed above).

Below is a summary of user rights within channels.

		Public	Logged-in Users							
			Users without Role(s)	Users with Role(s)						
				Subscribers	Authors	Editors	Publishers	Resource Managers	Module Managers	Administrators
Privacy	Public	View	View	View	View, Contribute	View, Approve/Decline	View, Approve/Decline	View, Contribute resources	Embed Modules	All
	All Members	-	View	View	View, Contribute	View, Approve/Decline	View, Approve/Decline	View, Contribute resources	Embed Modules	All
	Channel Members Only	-	-	View	View, Contribute	View, Approve/Decline	View, Approve/Decline	View, Contribute resources	Embed Modules	All

As seen in the table:

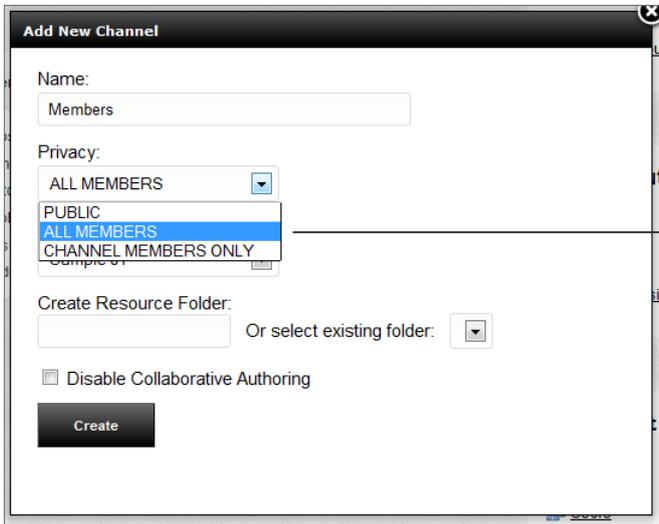
- Public cannot view content within the channel when privacy is set to **Public**. Only registered users may view the content. This type of channel is useful if you want to create a member area on your website where only registered users may access the content.
- Not all members can view content within the channel when privacy is set to **Channel Members Only**. Only users that have one of the 6 Roles in that channel may access the content. This type of channel is useful if you want to create a more private area on your website, where only certain users (besides registered users) may access to the content.

Note that if you are the only person who manages your website, after installation, you can just login as an Administrator and start managing your website's content. In this case, you don't need to setup channels and users.

By default, all pages that have **Channel Members Only** privacy are not accessible from the site navigation if a user is not authorized. Site users that do not have one of the six roles listed above will not be able to see the links on the site navigation. Even if they know the URL, they won't be able to access the page.

# Example 1: Creating a Member Area

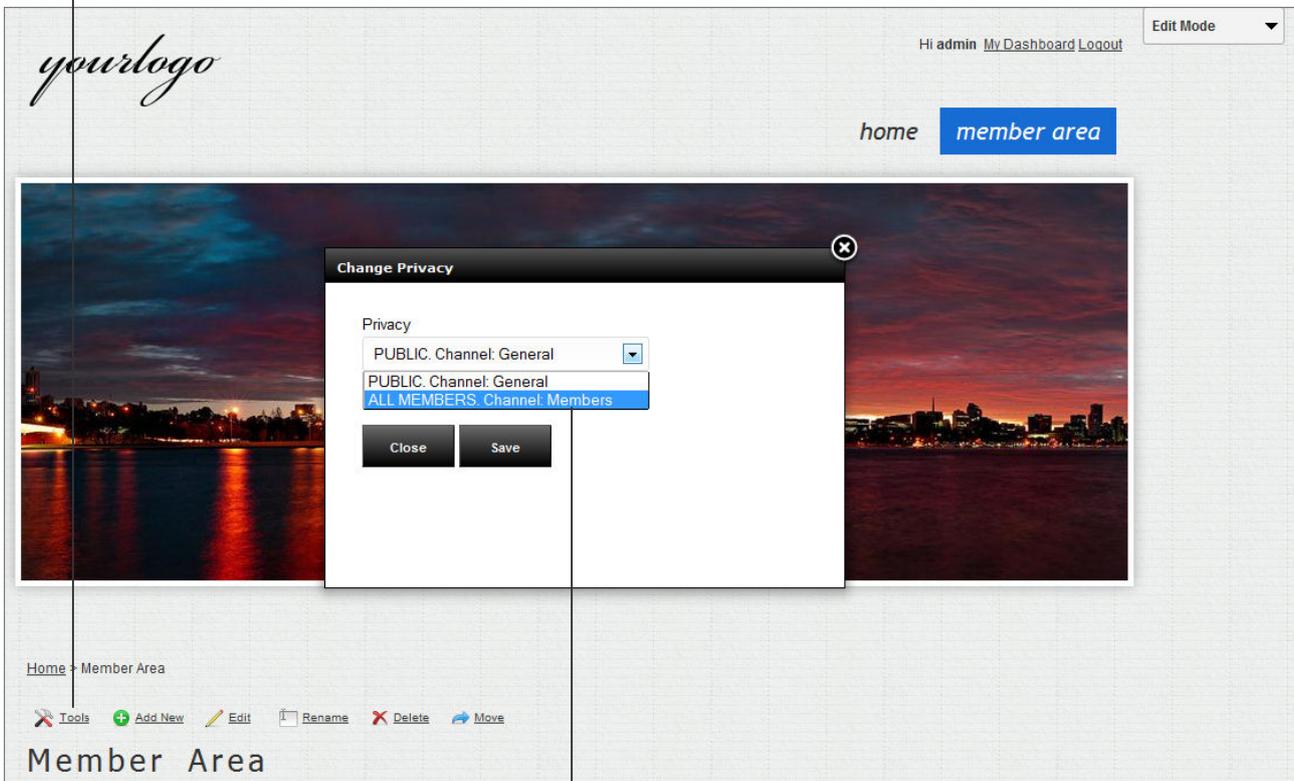
When planning your website, you may want to make some content available to the general public, while making other content available only to members, such as club members or website subscribers. The following steps explain how to create a member area:



1 Create a channel and set Privacy to **All Members**. In this example, we named the channel: **Members**.

2

Go to a page that you'd like to make viewable by members only. In this example, the page chosen is **member-area.aspx** (Creating a page will be explained in the next section). On the page information (**Tools**), click the **Change** link to open the **Change Privacy** dialog.

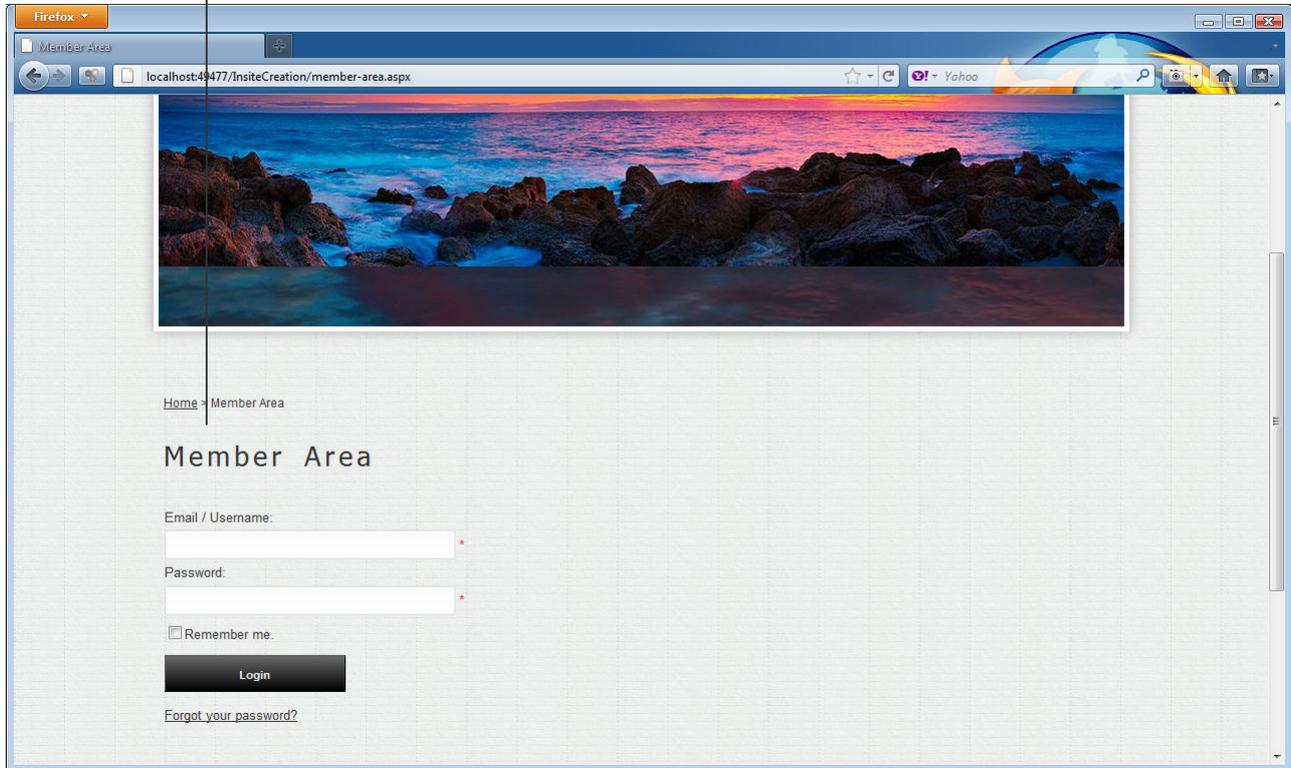


3

Change the page channel to the **Channel Members** with privacy **ALL MEMBERS** click **Save**.

4

When site visitors try to open the **member-area.aspx**, a login form will be displayed, requiring them to enter a valid username and password to access the page. Since the channel privacy is set to **All Members**, all registered users will be able to login and view the page.



**Note:**

When creating the channel **Members**, the system will automatically create 6 Roles to which users can be assigned:

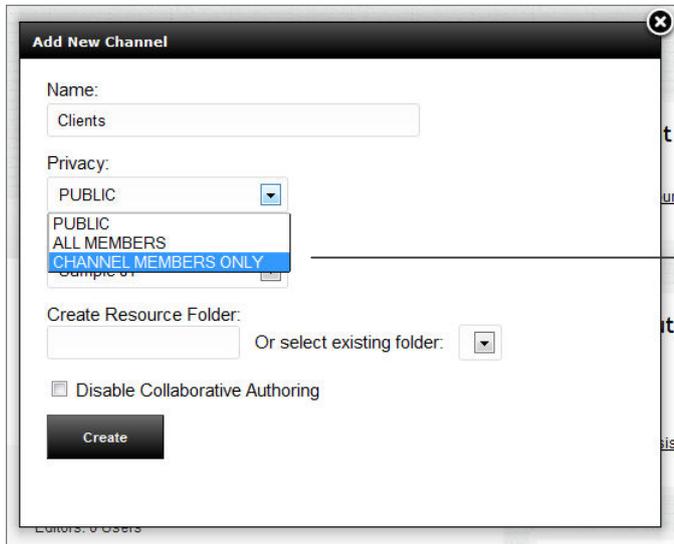
- Members Subscribers;
- Members Authors
- Members Editors
- Members Publishers
- Members Resource Managers
- Members Modules Managers

Since the **Members** channel privacy is set to **Public**, all registered users (regardless of their own roles) may also view the content.

If you want to allow only users from a certain role, see example 2.

## Example 2: Creating a Client's Area

If you want to provide your clients with a unique content area where they can securely view (or even modify) work in progress, download project contracts and access other sensitive documents. In this case, not all registered users are able to open that area. Only users that are members of the Channel are allowed to view the area.



1

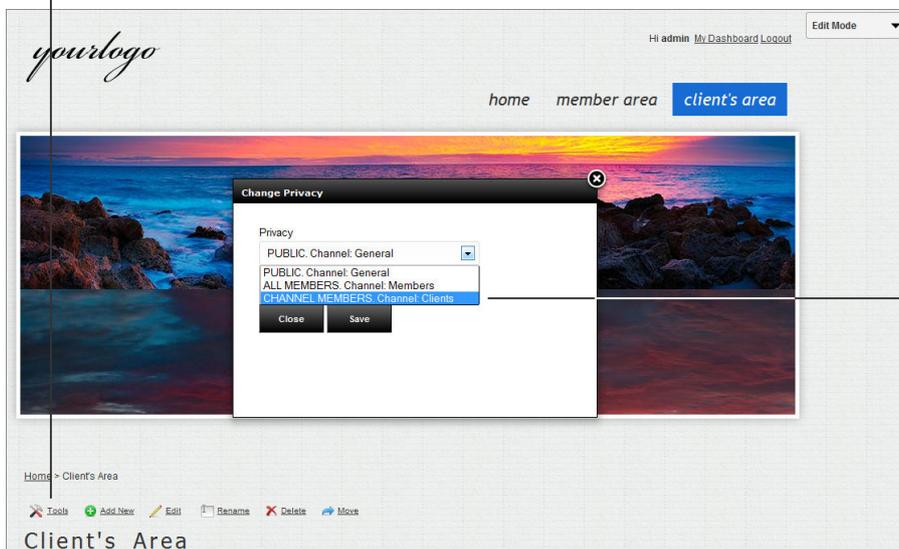
Create a Channel and set Privacy to **Channel Members Only**. This means that pages assigned to this channel can only be viewed by users from this channel and not by all registered users. In this example, we named the channel: **Clients**.

When this channel is created, the system will automatically create six roles to which users can be assigned:

1. Clients Subscribers
2. Clients Authors
3. Clients Editors
4. Clients Publishers
5. Clients Resource Managers
6. Clients Module Managers

2

Go to a page that you'd like to make accessible only to your clients. In this example, the page chosen is **clients-area.aspx**. On the page channel information (**Tools**), click **Change** link to open **Change Privacy** dialog.

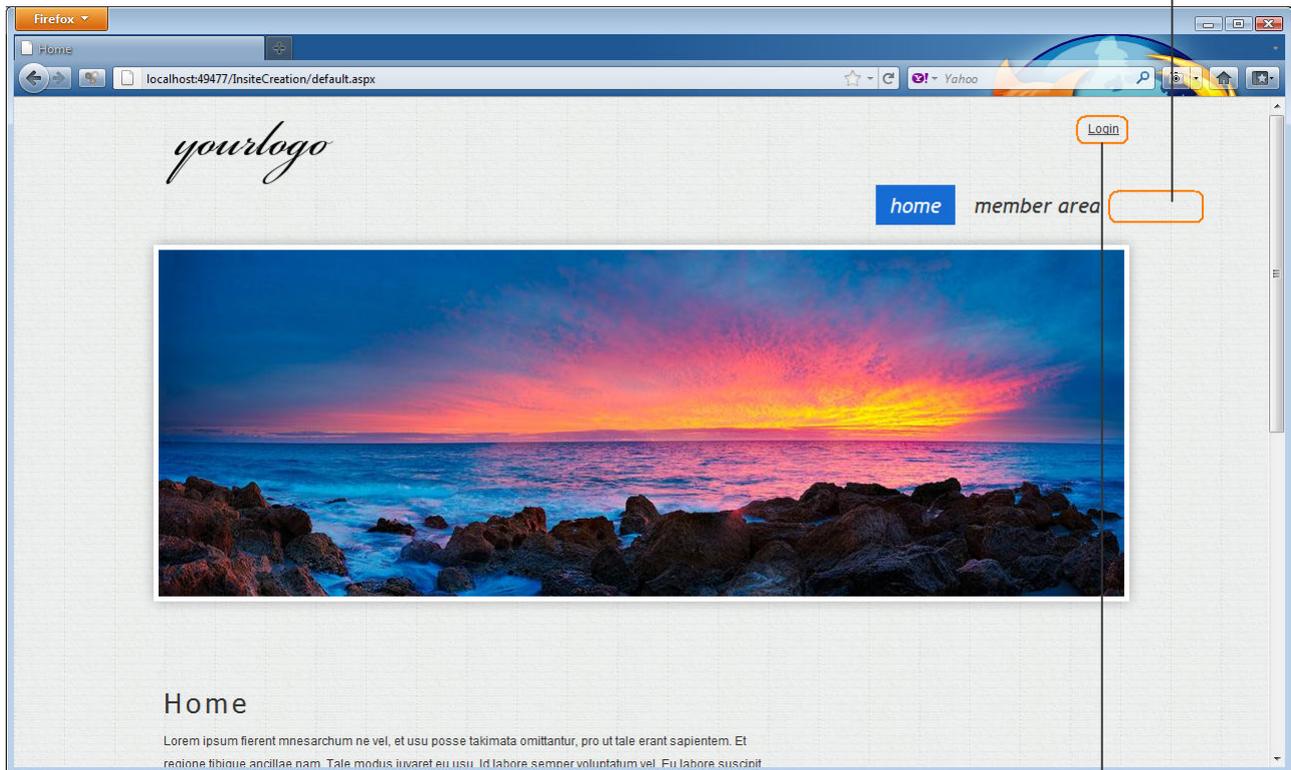


3

Change the page channel to the new **Clients** channel with privacy CHANNEL MEMBERS ONLY and click **Save**.

As a result, the page **will not be visible on site navigation.**

4

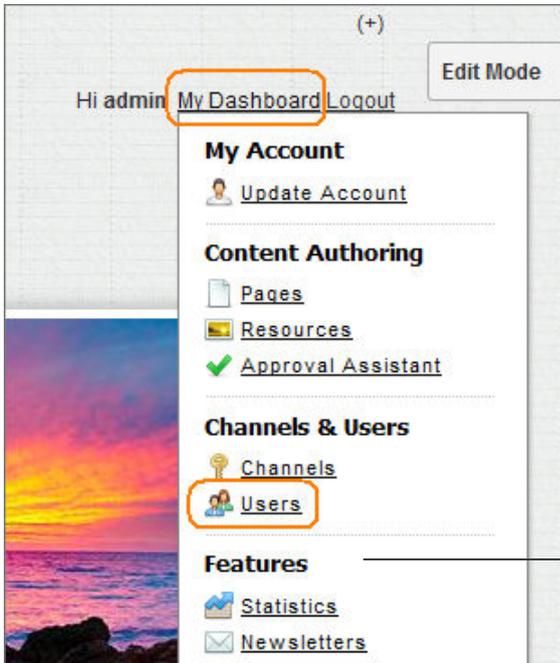


Your clients will need to be logged in first (via the login link or login form provided on the site) to be able to see links to any pages assigned to the **Clients** channel. Site users that do not have one of the six roles listed above will not be able to see links to the **clients-area.aspx**. Even if they know the URL, they won't be able to access the page.

5

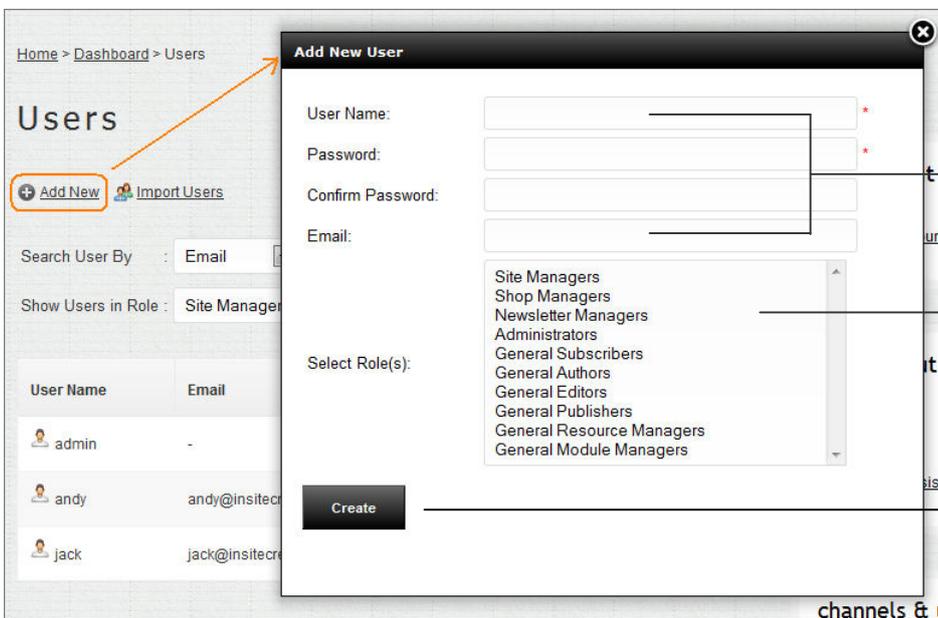
# Creating a User

To create a user, hover your **Dashboard**.



1 Select the **Users** link.

A list of all existing users will be displayed on the Users screen:



2 Fill in the new user information.

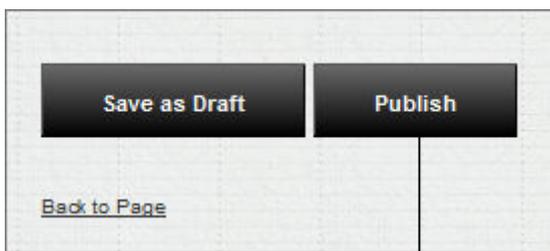
3 Select one or more roles you want the user to have.

4 Click **Create** to add the user to the system. The message "Your account has been successfully created." will be displayed.

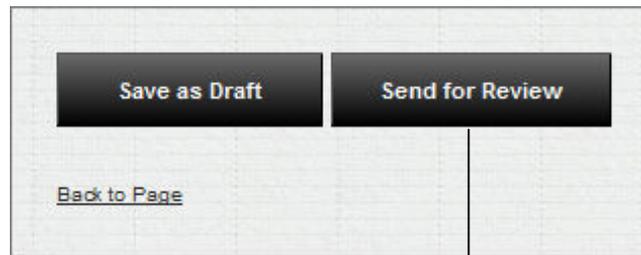
# Setting-up Approval Workflow

SiteManager supports approval workflow that consists of Editor and/or Publisher approval. You can assign users either as an Editor or Publisher in a channel so that the channel has an approval workflow, otherwise pages in that channel will be published directly without approval.

When editing page content, you have either a **Publish** button or a **Send for Review** button. If approval is needed, the **Send for Review** button is displayed; otherwise the **Publish** button will be displayed.



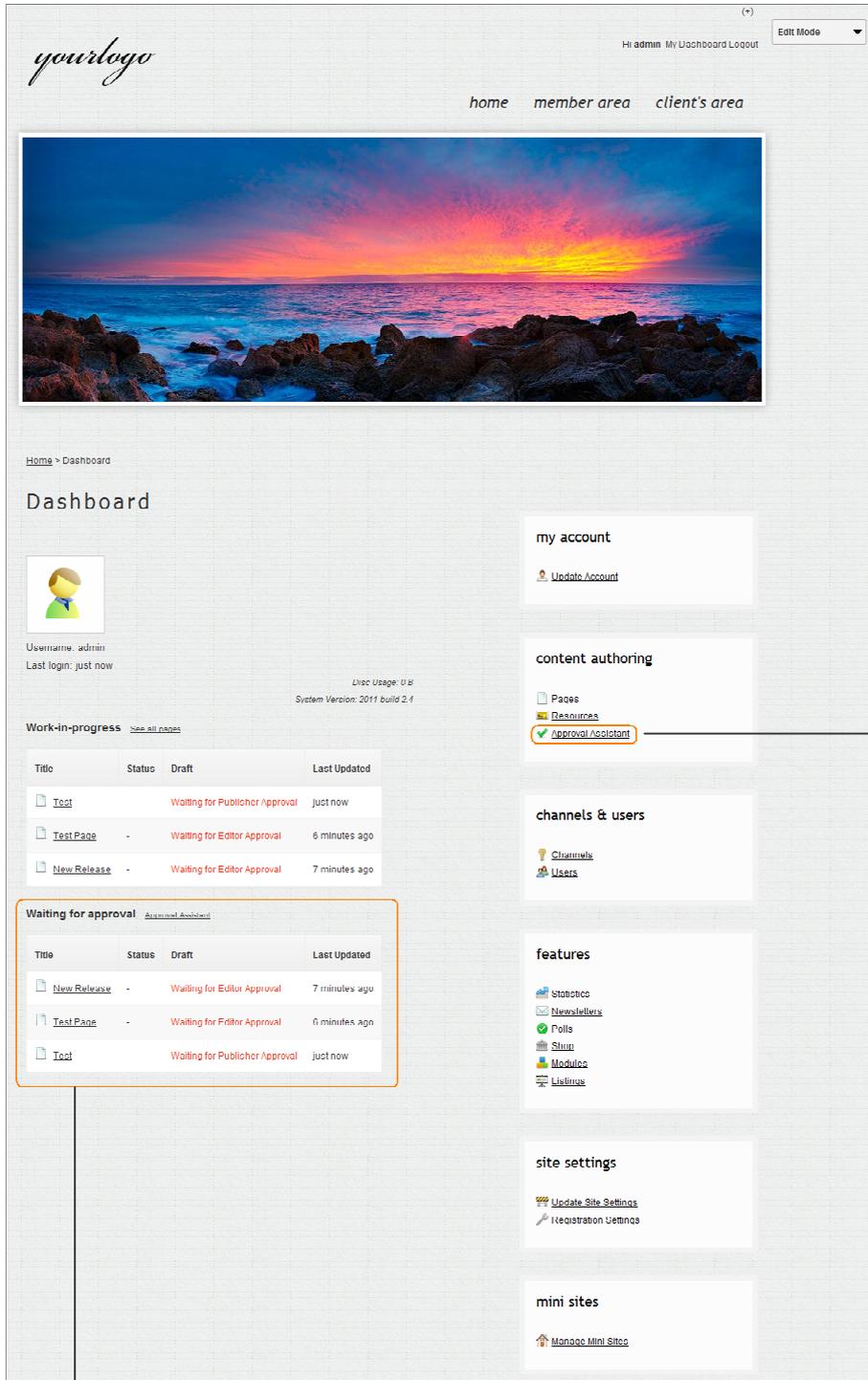
Publish Button



Send for Review Button

# To Approve or Decline a Page

To see a list of pages which have been submitted for approval, open your **Dashboard**.

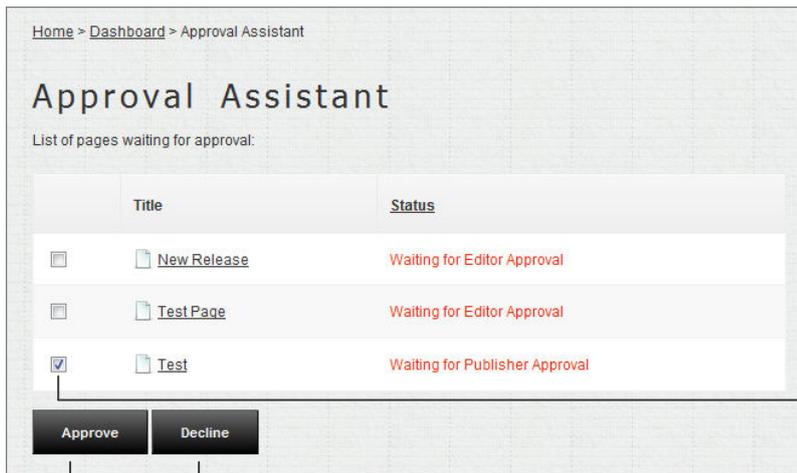


1

Users who have an **Editor** or **Publisher** role will be able to see the list of pages which have been submitted for approval. Note that if a page is submitted, it is passed to the **Editors** first and then to the **Publishers**. Publishers will not see the page here until it is approved by an Editor.

2

An **Approval Assistant** is also available to users with an Editor or Publisher role. Try selecting the **Approval Assistant** link.



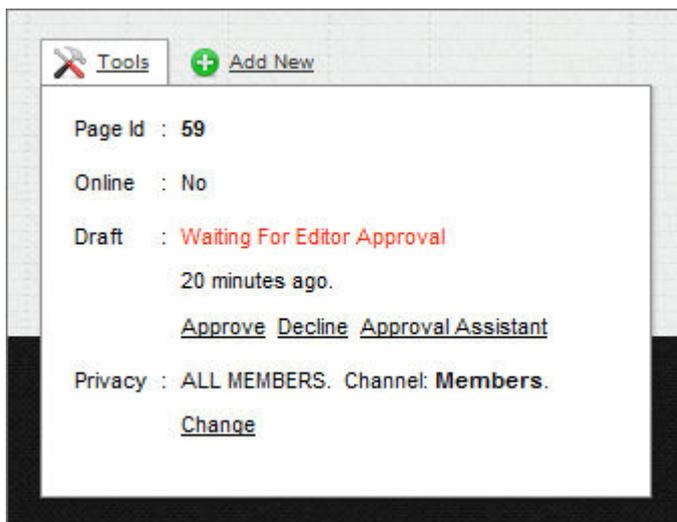
3

Select the page you wish to approve or decline from the list by clicking the checkbox next to that page.

4

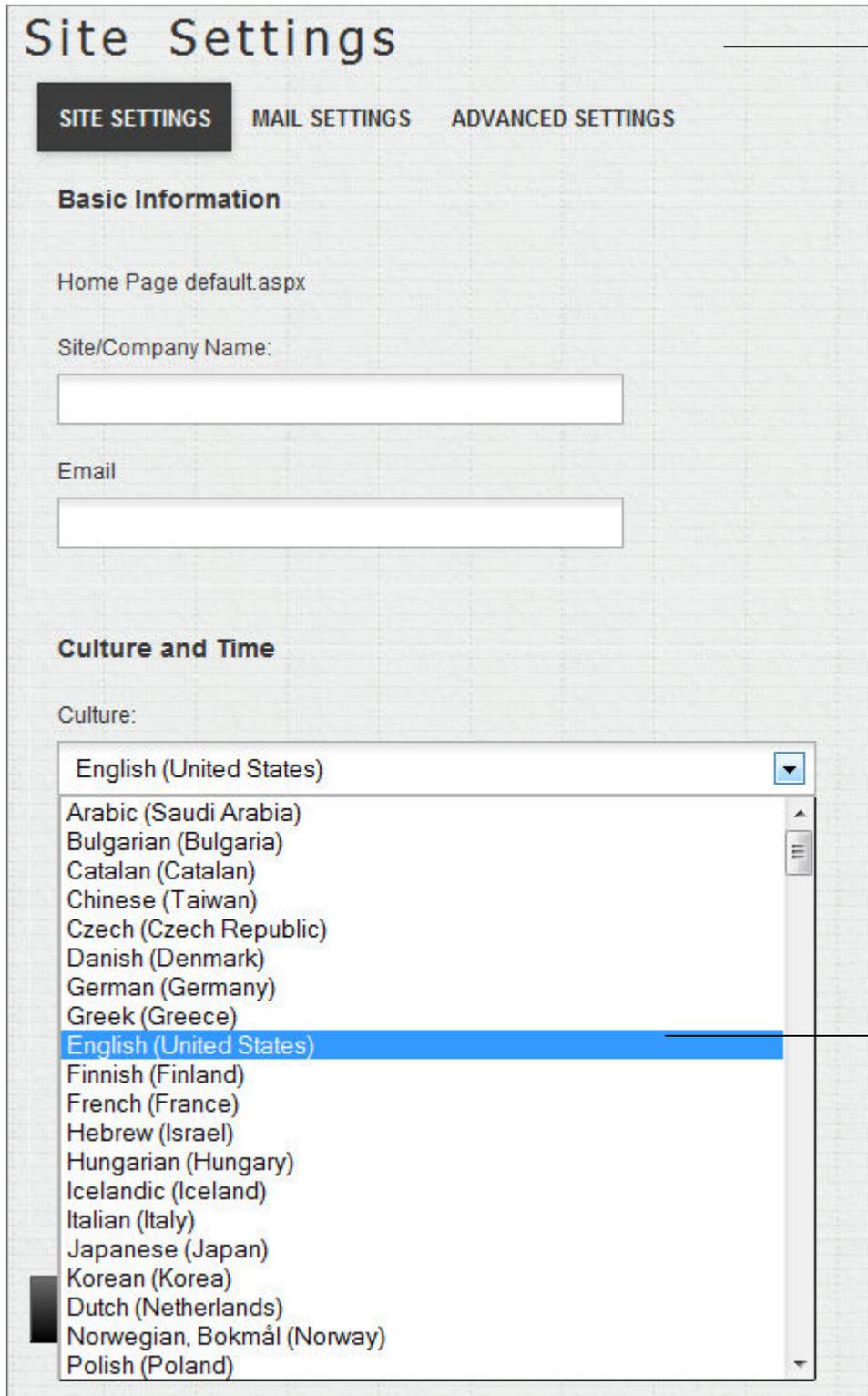
Click the **Approve** button or the **Decline** button. When a page is declined, the page is made available for the author to make revisions.

You can also approve or decline a page by opening the page and using the **Approve** or **Decline** links in the Page Info box.



# Site's Language & Settings

SiteManager allows you to change the language (culture) of your site. Your site will render the text, captions, date, and number format according to the selected language (culture). Go to **Dashboard** and select **Update Site Settings**.



The screenshot shows the 'Site Settings' panel with three tabs: 'SITE SETTINGS', 'MAIL SETTINGS', and 'ADVANCED SETTINGS'. The 'SITE SETTINGS' tab is active. Under the 'Basic Information' section, there are fields for 'Home Page default.aspx', 'Site/Company Name', and 'Email'. The 'Culture and Time' section features a 'Culture:' dropdown menu. The dropdown is open, showing a list of cultures with 'English (United States)' selected and highlighted in blue. A red callout box with the number '1' points to the top of the panel, and another red callout box with the number '2' points to the dropdown menu.

**1** The **Site Settings** panel is displayed.

**2** Change culture setting here.

# Language Files

A culture in SiteManager requires language files. If you select a culture for your site that doesn't have language files, then the default English language files will be used automatically.

Language files are found in the following folders:

- <yourwebsite>/App\_LocalResources
- <yourwebsite>/dialogs/ App\_LocalResources
- <yourwebsite>/systems/ App\_LocalResources
- <yourwebsite>/modules/ App\_LocalResources

Language files (also called resource files) are files with the extension “**resx**” that contain language translation of the application text or captions. You can translate the application panels of SiteManager by adding your language resource files. To translate the panels to German for example, duplicate all resource files in the above folders and add “**.de-DE**” to the end of the file name. For example:

**account.ascx.resx**

would be:

**account.ascx.de-DE.resx**

You can then translate all the text or captions defined in the new resource files. The translation will be applied automatically if the site has the culture setting set to German (Germany).

Below are some country codes you can use:

- da-DK - Danish (Denmark)
- de-DE - German (Germany)
- el-GR - Greek (Greece)
- en-US - English (United States)
- fi-FI - Finnish (Finland)
- fr-FR - French (France)
- it-IT - Italian (Italy)
- ja-JP - Japanese (Japan)
- nl-NL - Dutch (Netherlands)
- pl-PL - Polish (Poland)
- sv-SE - Swedish (Sweden)
- zh-CN - Chinese (People's Republic of China)
- en-GB - English (United Kingdom)
- nl-BE - Dutch (Belgium)
- nn-NO - Norwegian, Nynorsk (Norway)
- es-ES - Spanish (Spain)

# Mini Sites

SiteManager also allows you to create multiple mini sites with different languages (cultures). With this feature, you can:

- Create multiple-language sites,
- Create mini sites for each of your company's departments,
- Create mini sites for each of your club members,
- Etc.

The term “mini” does not refer to the size of the site. You are able to create as many pages as you like on a “mini” site, just as you would on the main site.

## Creating a Mini Site

To create a new site, go to the **Dashboard**, then select the **Manage Mini Sites** menu. Follow the steps below:

1 Enter the **Home Page** for the new site (ex. **default-de** for German).

2 Select Design template.

3 Click the **Create Site** button to create your new site.

# Mini Sites

+ Add New

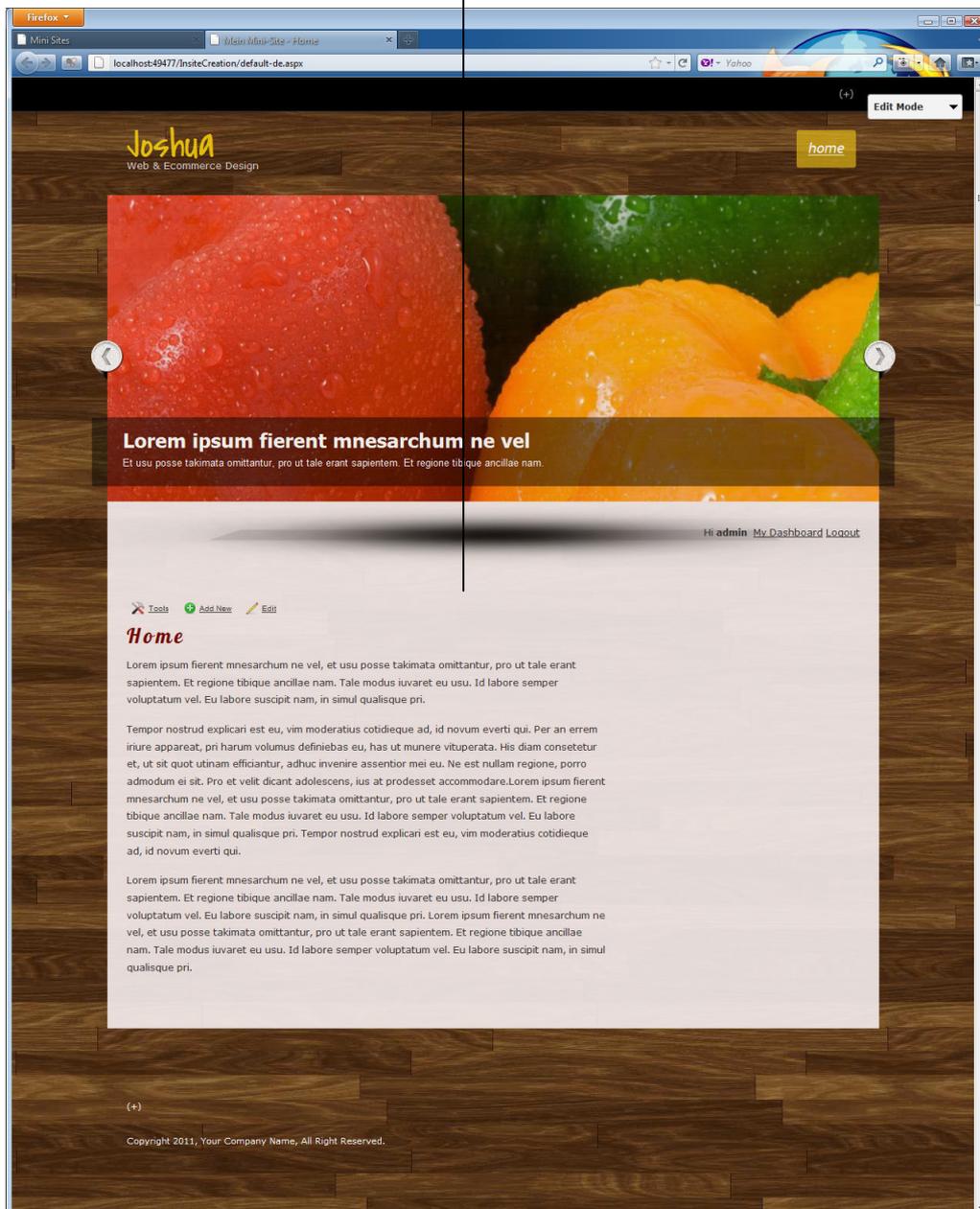
Home Page	Site Name	Culture	
 default.aspx		English (United States)	<a href="#">Go To</a>
 default-de.aspx	Mein Mini-Site	English (United States)	<a href="#">Go To</a> <a href="#">Delete</a>

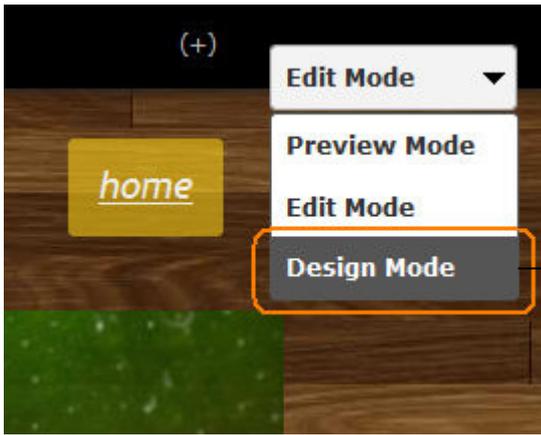
4

A new site is now created. Try clicking the **Go To** link.

5

Now you are on the main page of your new site. You can change the site name or logo to be displayed on the header, copyright info on the footer and any other design aspect.





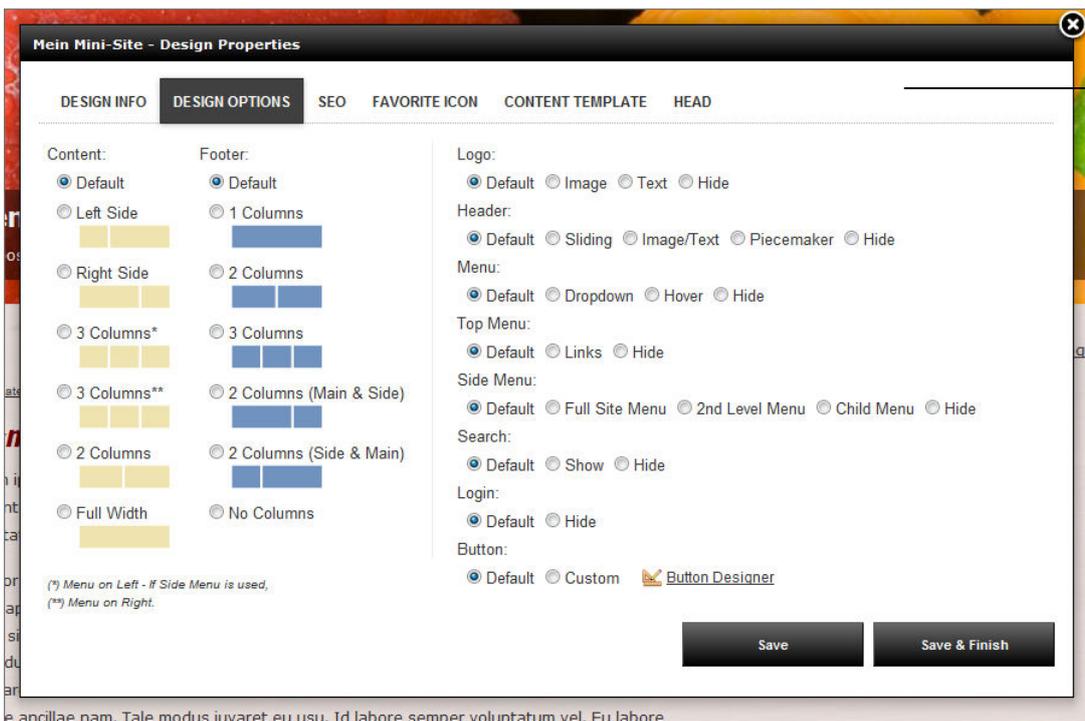
6

Set your page to **Design Mode**.



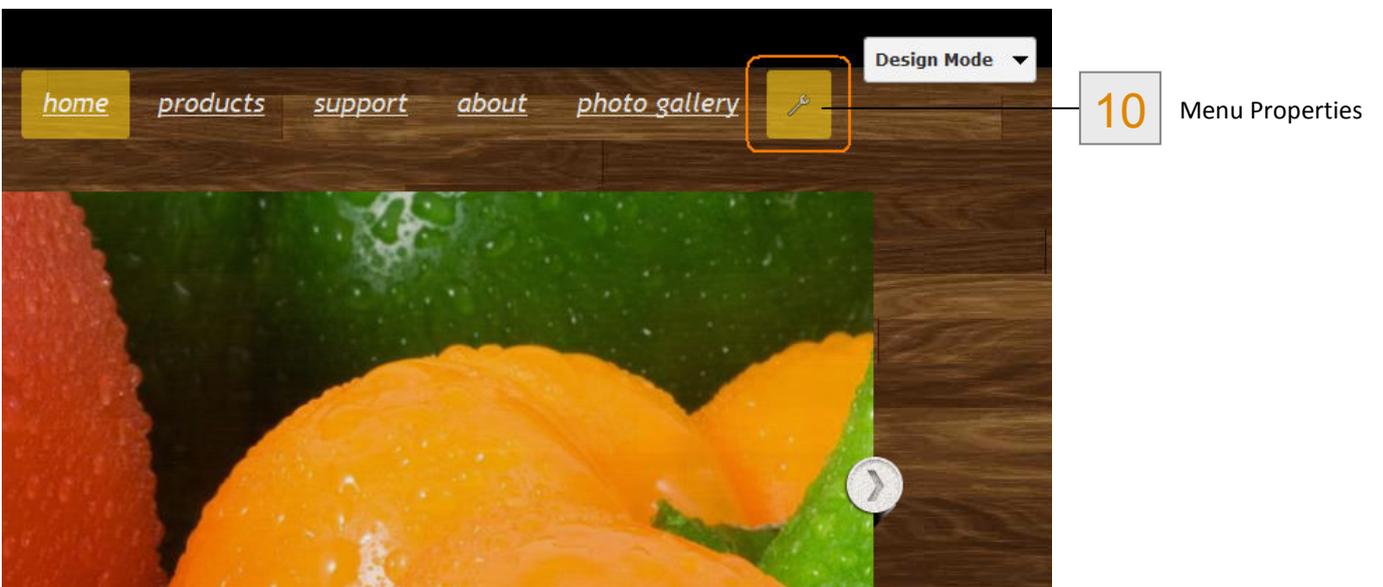
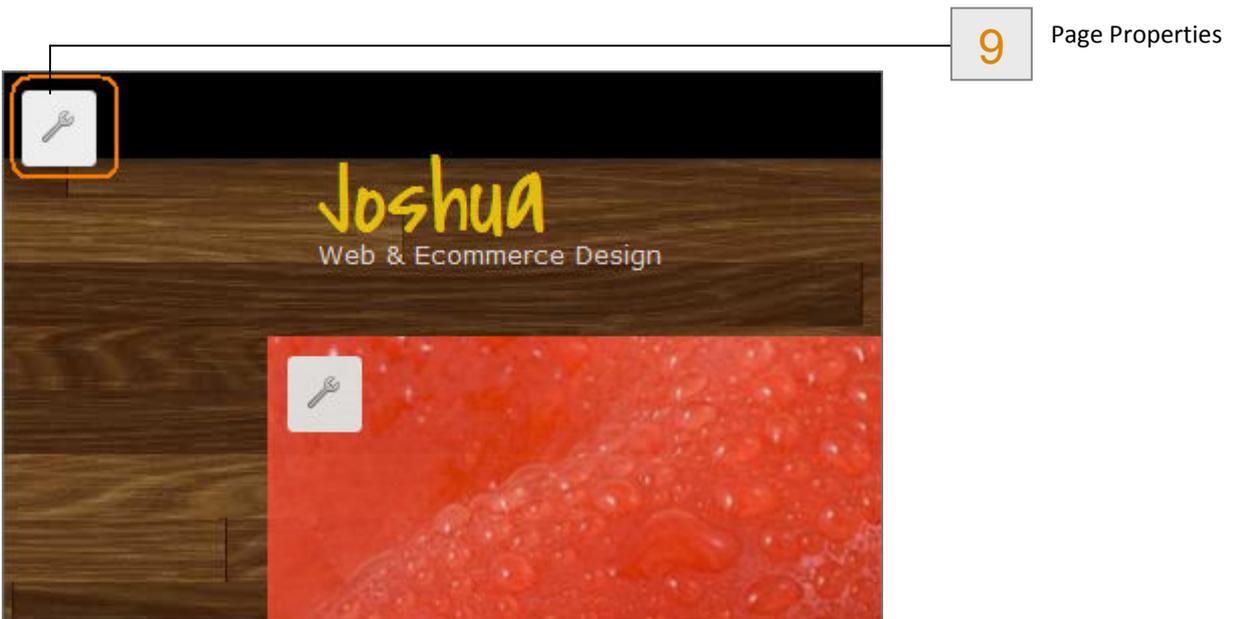
7

Here you can specify the design aspects of your site.



8

Design Properties



11 Banner Properties





Here are some facts about a mini site:

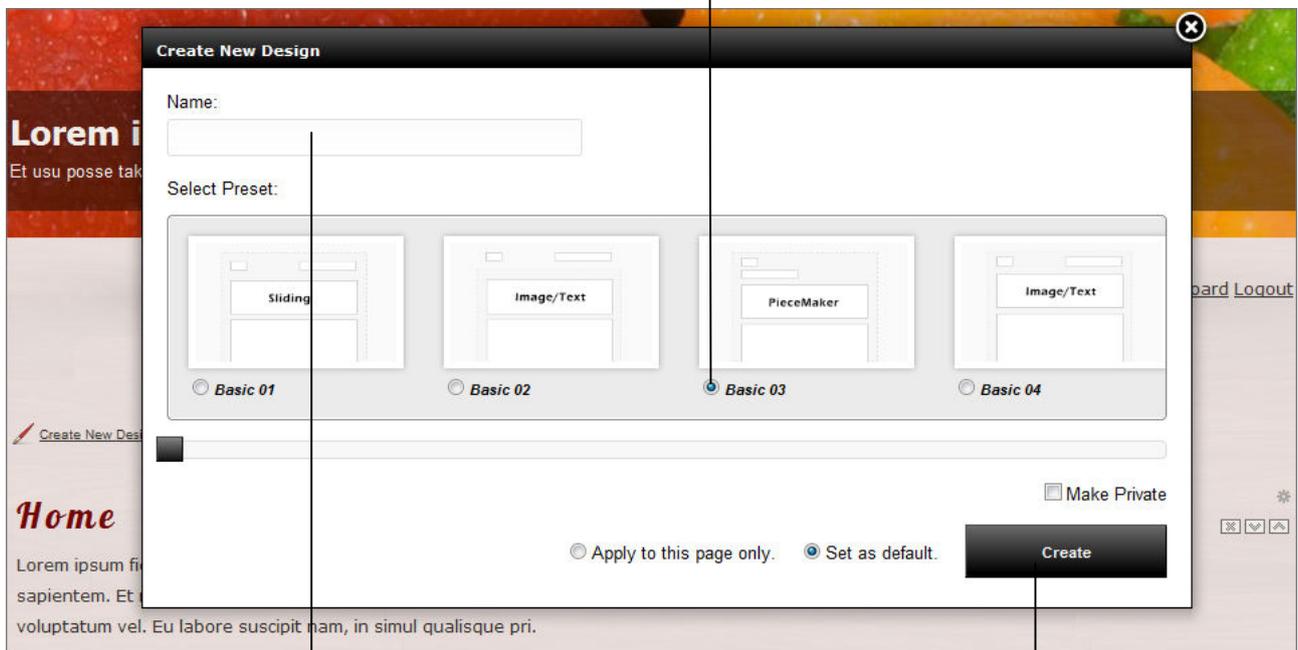
- A mini site is created in the same location (domain) of your existing site.
- A mini site has a separate menu/navigation system.
- Each mini site has a role called "Site Managers". You can assign users to this role to be an administrator of the site.

# Template Development

SiteManager uses design templates to separate content from design. You can change the design of your web page by applying a different template to the page. One of SiteManagers' strength with regard to template development is its simple and easy way to create a new template/design. SiteManager provides several basic layouts or designs you can choose as the base of your template.



1 Click "Create New Design" to create new template.

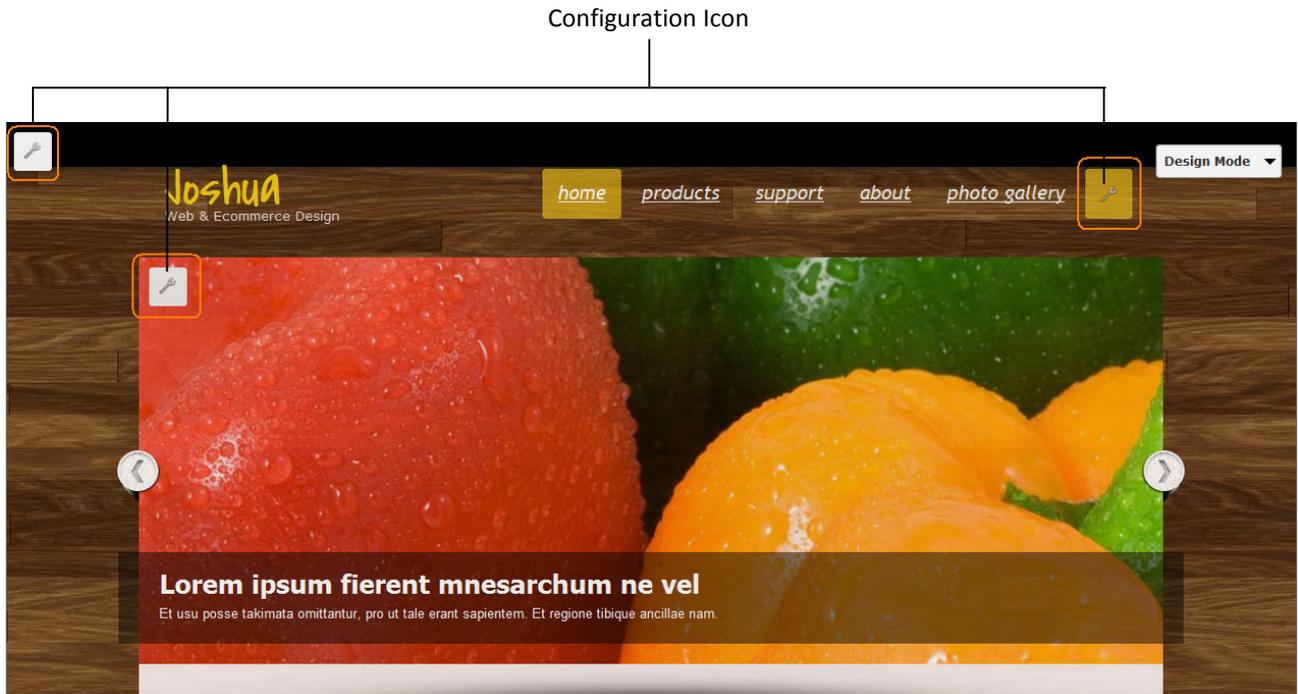


2 Select the basic layout for your template.

3 Enter the name of your template.

4 Create the template.

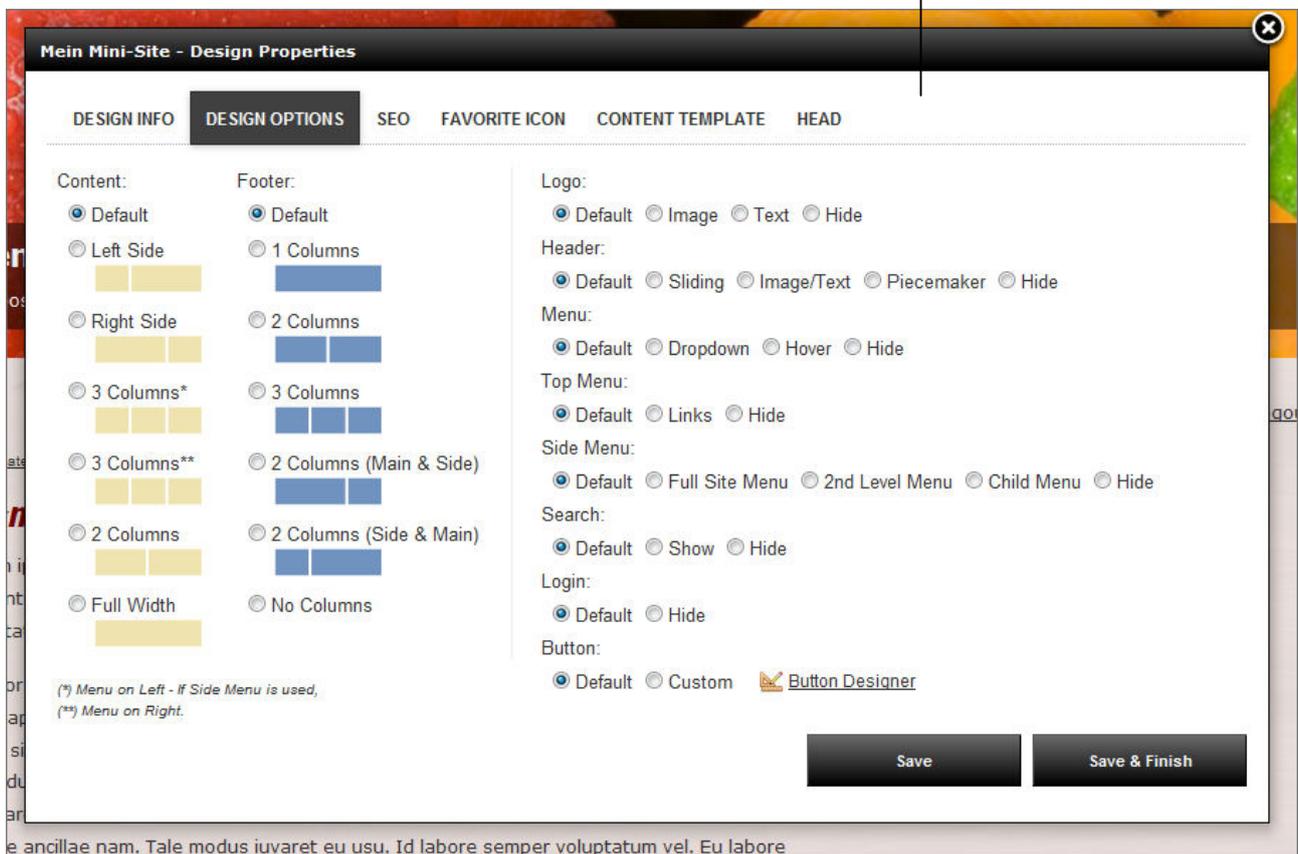
Here is the created template:

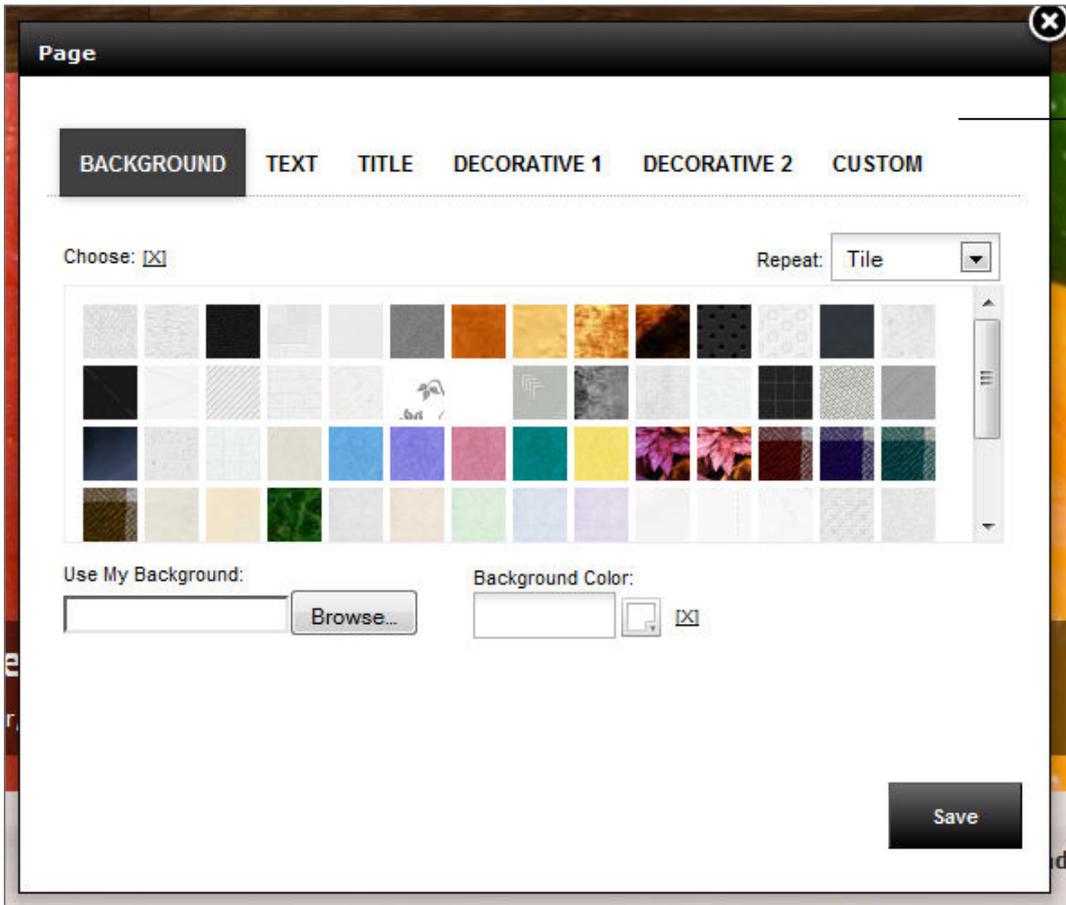


Configuration Icon

You can then use the “**Design Properties**” link to change the template properties. You can also configure the individual part of page such as top menu, main menu, banner, page properties, footer by hovering the page element and click on configure icon.

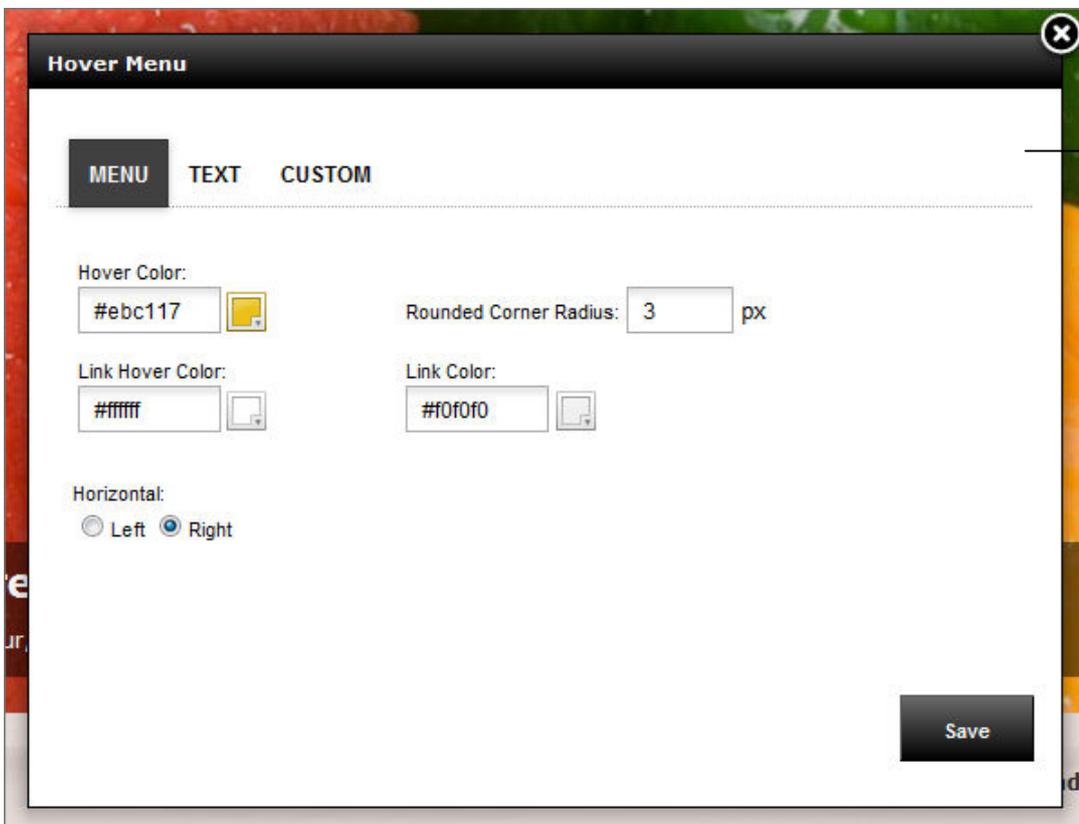
1 Design Properties





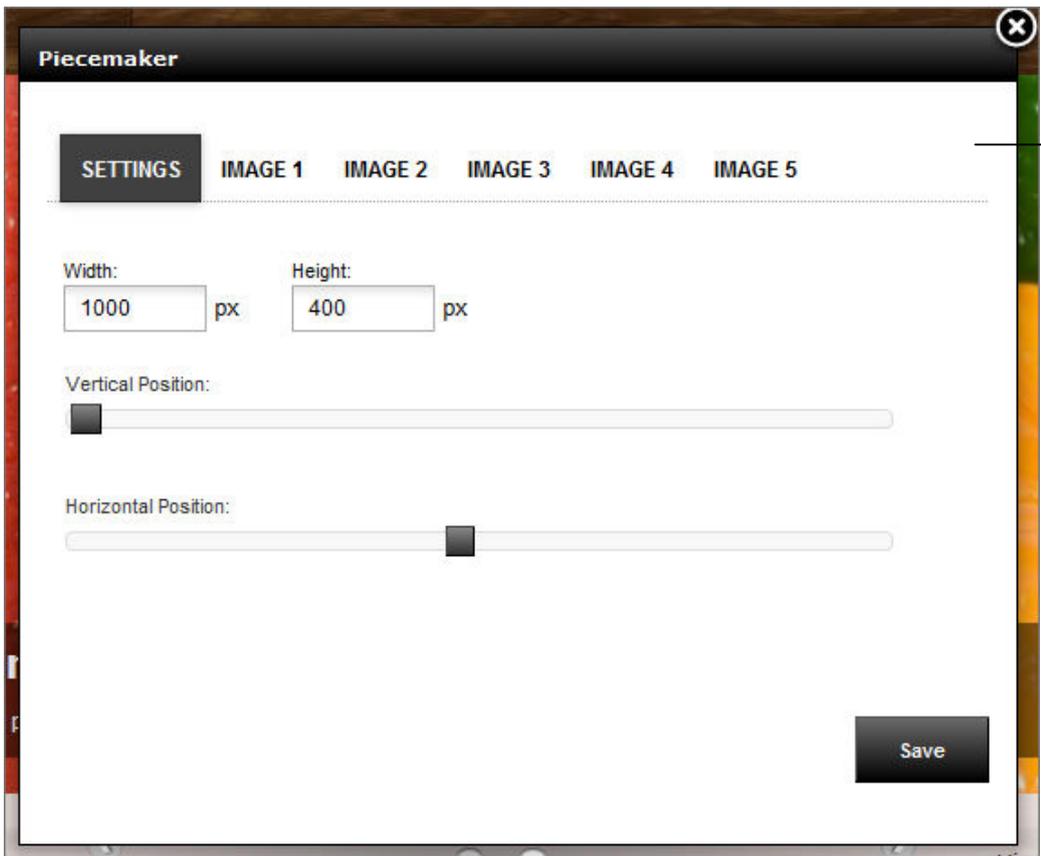
2

Page Properties



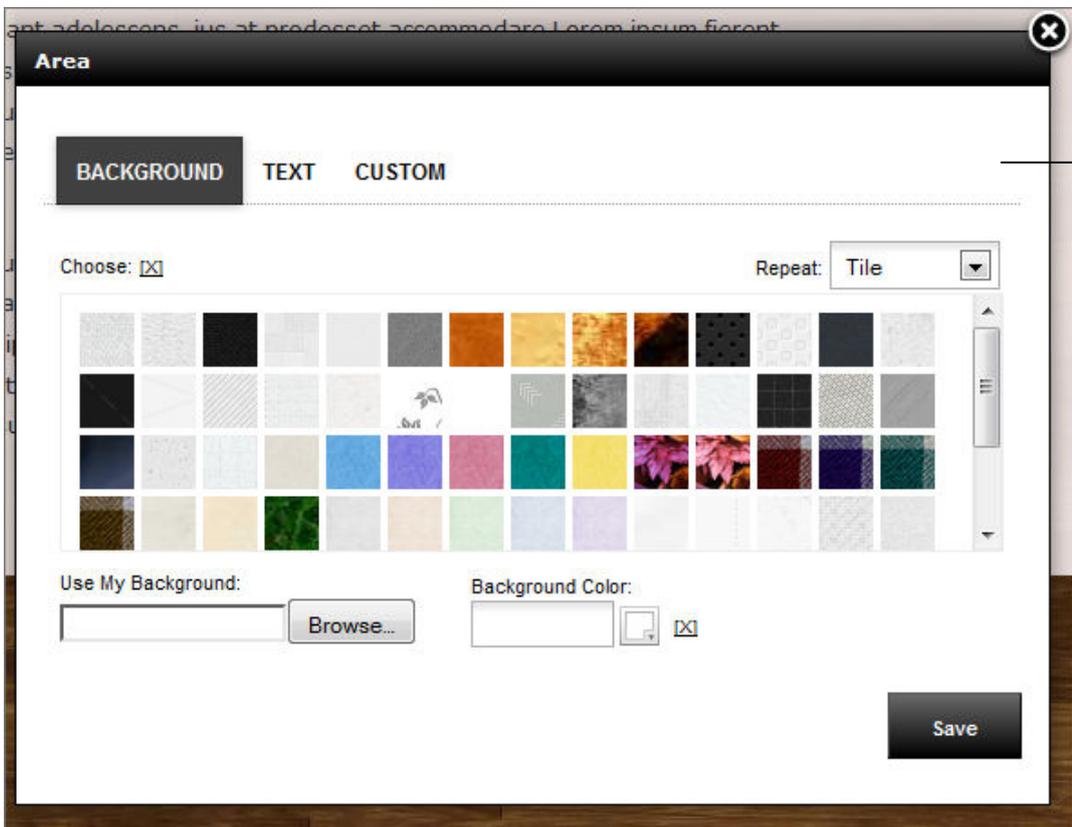
3

Menu Properties



4

Banner Properties

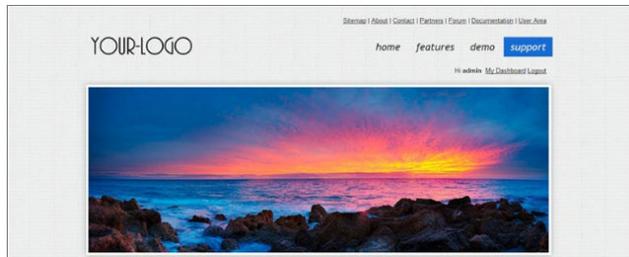


5

Footer Properties

# Built-in Templates

Below are the built-in templates in SiteManager:



Sample 01



Sample 02



Sample 03



Sample 04



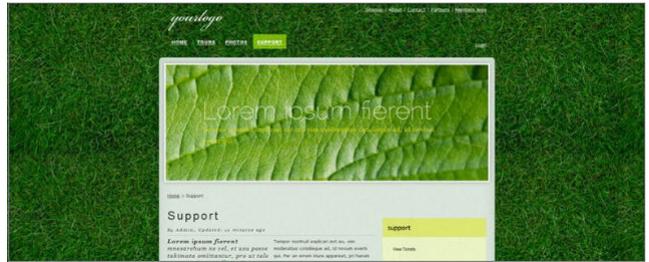
Sample 05



Sample 06



Sample 07



Sample 08



Sample 09

# Template Files

You can create your own template(s) or use the built-in templates provided by SiteManager. You can also modify the built-in templates to meet your needs.

A basic template contains the following files:

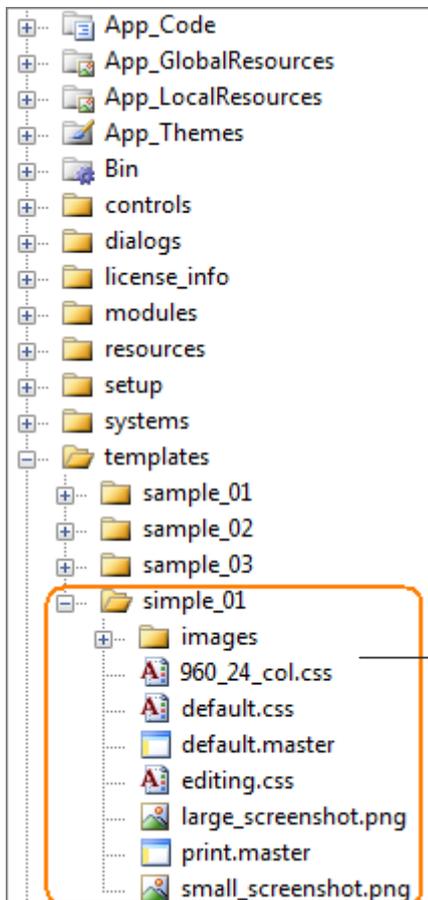
- default.master, main template file.
- default.css, a stylesheet for the template file.
- print.master, template for printer friendly page
- editing.css, stylesheet for WYSIWYG Editor

**TIP:** You can copy these files from any built-in template and modify them according to your site design.

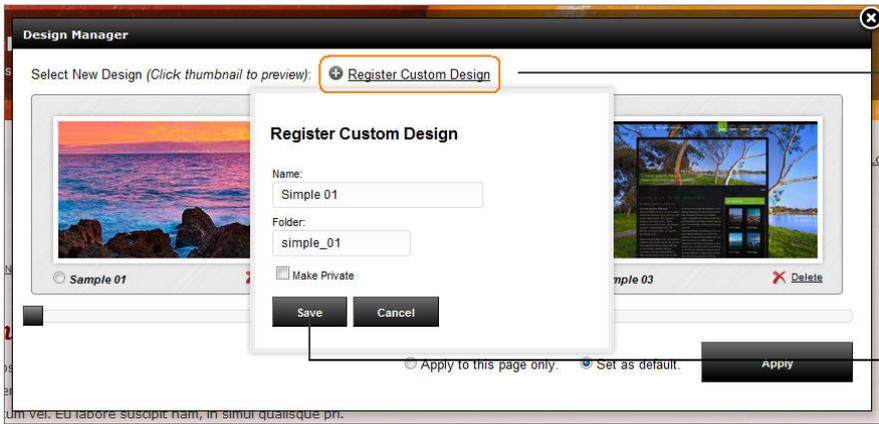
## Adding a New Template Manually

Before we look further at how to create a template, the steps below show how to add a new template to use in SiteManager.

### SiteManager files



1 Templates are located in the **templates** folder. Copy your template files here. In this example, we named our new template folder: **simple\_01**.

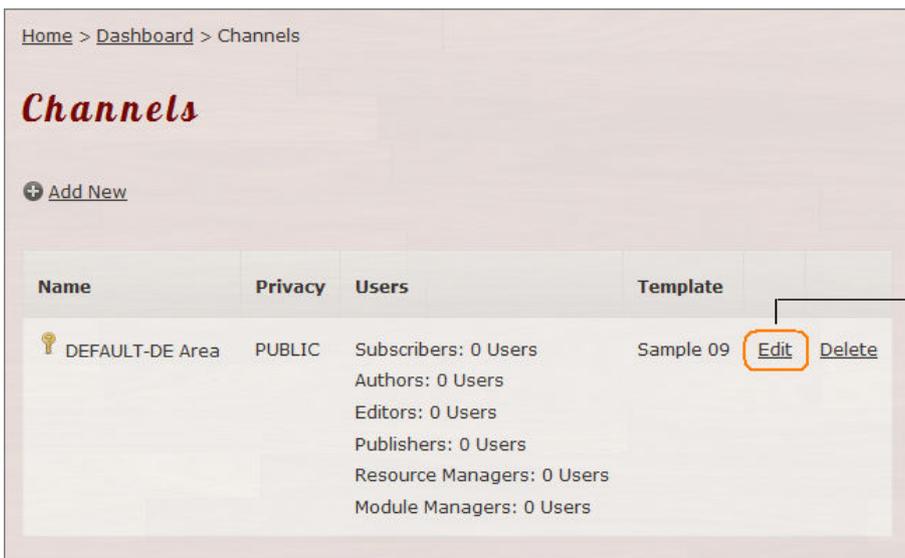


**2** On **Design Mode**, select **Design Manager** then click **Register Custom Design**. Here you can register the template. Fill in the name and select the folder **simple\_01**.

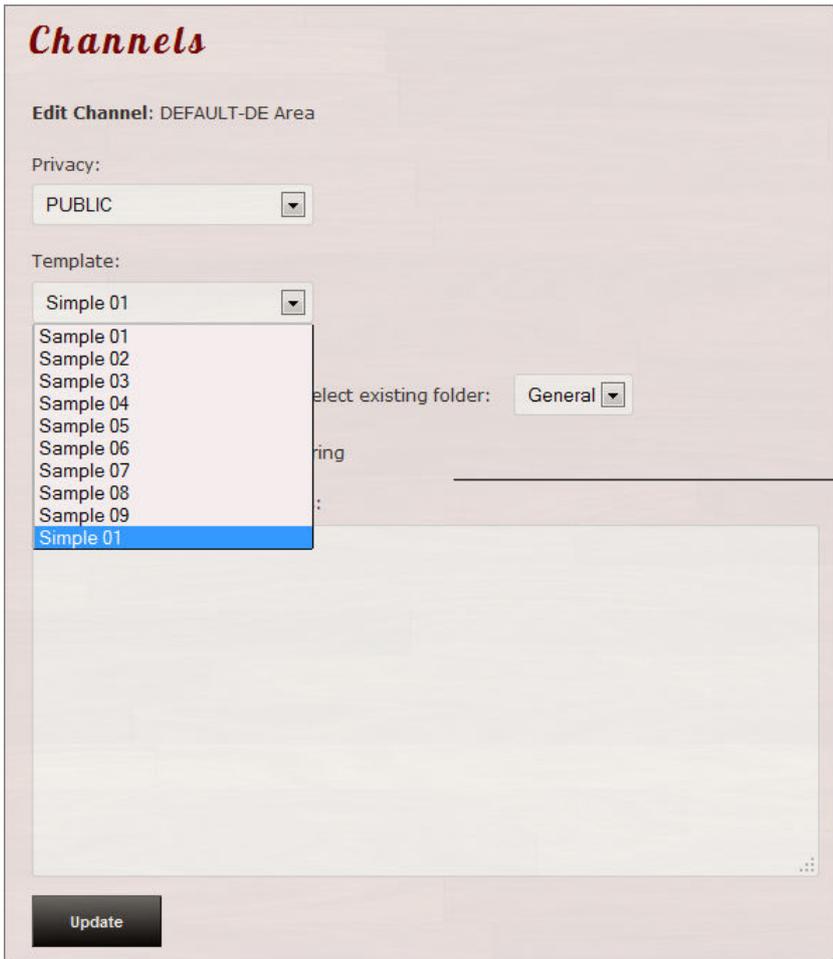
**3** Click **Save** to finish. Now the template is available for use.

## Applying Your Template

If you want to apply the template to all pages within a certain channel, go to the **Dashboard** and select the **Channels** menu.



**1** Click **Edit** on the channel in which you want to apply the template to all pages in that channel.

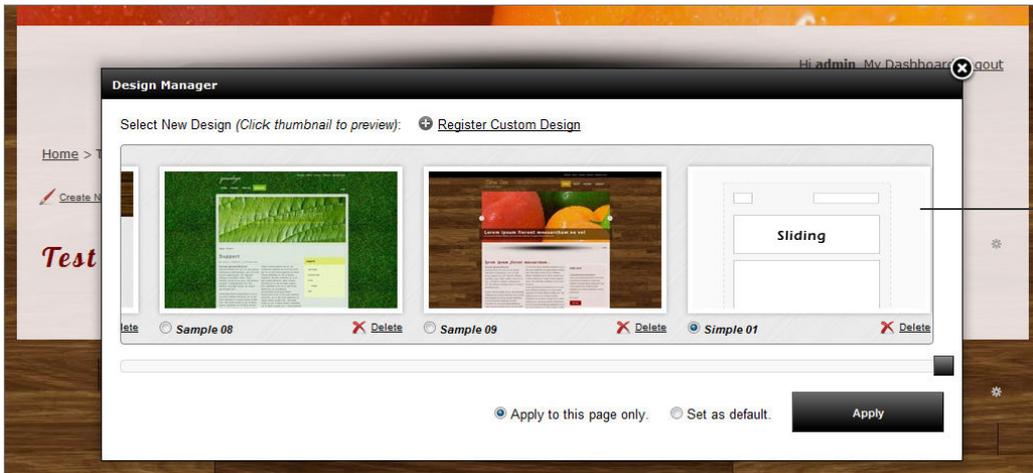


2 Here you can change the default template.

If you want to apply your template to a specific page, go to that page and edit it.



1 Set page on **Design Mode**. Click **Design Manager** to edit the page.



2

Here you can change the page template.

In the next section, we will look further into the main template file: **default.master**.

# More on Template File (default.master)

Now we will look further into a template file. The main template file is the **default.master** file, which contains mainly regular HTML code with placeholders for various elements (menu, title, body content, etc). You can create and edit this file using any text editor. We would recommend that you use Visual Web Developer which you can download from the Microsoft website.

**Below are the basic requirements for the default.master template file:**

```
1 <@ Master Language="VB" Inherits="BaseMaster" %>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-trans
3 <html lang="en" xmlns="http://www.w3.org/1999/xhtml" >
4 <head runat="server">
5 <title>Your Company Name - </title>
6 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
7 <link href="default.css" rel="stylesheet" />
8 </head>
9 <body>
10 <form id="Form1" runat="server">
11
32 <asp:contentplaceholder ID="placeholderTitle" runat="server"></asp:contentplaceholder>
33 <asp:ContentPlaceHolder ID="placeholderBody" runat="server"></asp:ContentPlaceHolder>
49 </form:Form>
50 </body>
51 </html>
```

**1** It must inherit from SiteManager's BaseMaster class

**2** A header element with the runat property set to "server"

**3** A link to default.css. You can copy & modify the default.css from the SiteManager's prebuilt template.

**4** A Form element in the form of <form:Form ...>

**5** Placeholders to display title, body content, etc.

Below is a list of available placeholders and its preview. Some of the placeholders require additional control script to be defined in master page for example:

```

300 | <div class="clear"></div>
301 | <div class="grid_24">
302 | <asp:ContentPlaceholder ID="placeholderMenu" runat="server"></asp:ContentPlaceholder>
303 | </div>

```

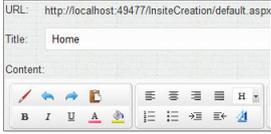
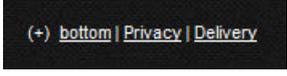
```

85 | placeholderMenu.Controls.Add(Utils.LoadMyControl(Me, "~/controls/menu_dropdown.ascx"))

```

In the example, the *menu\_dropdown.ascx* will be loaded into **placeholderMenu** placeholder. With this mechanism, you can create your own placeholder and custom control to render the placeholder.

Predefined Placeholder	Available Controls	Preview
placeholderTopMenu	default	
placeholderRegister	default	
placeholderWelcome	default	
placeholderDashboardLink	default	
placeholderLoginLogout	default	
placeholderLogo	logo_text.ascx logo.ascx	
placeholderSearch	default	
placeholderCountrySelect	default	
placeholderLoginForm	default	
placeholderMenu	menu_dropdown.ascx menu_fadein.ascx	
placeholderHeader	sliding_image.ascx image.ascx slicing_image.ascx	
placeholderBreadcrumb	default	
placeholderPrint	default	

placeholderAuthoring	default	
placeholderOrderNow	default	
placeholderCartInfo	default	
placeholderLeft_SideMenu	menu_side.ascx menu_vertical.ascx menu_children.ascx	
placeholderLeft	default	Left content.
placeholderTitle	default	
placeholderBody	default	Page content (body).
placeholderContentRating	default	
placeholderComments	default	
placeholderRight_SideMenu	menu_side.ascx menu_vertical.ascx menu_children.ascx	
placeholderRight	default	Right content.
placeholderBottomMenu	default	

# Module Development

## Creating a Module

A module in SiteManager is simply an ASP.NET (ascx) User Control. You can find many references on the internet about how to create ASP.NET User Controls. With ASP.NET User Controls you can create any application and embed it onto any SiteManager page.

There is only one basic requirement for an SiteManager module:

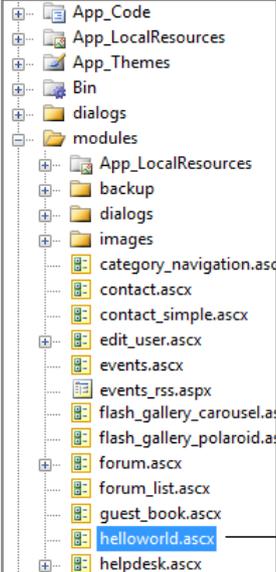
1 A module should inherit from SiteManager's BaseUserControl class. This line is a must.



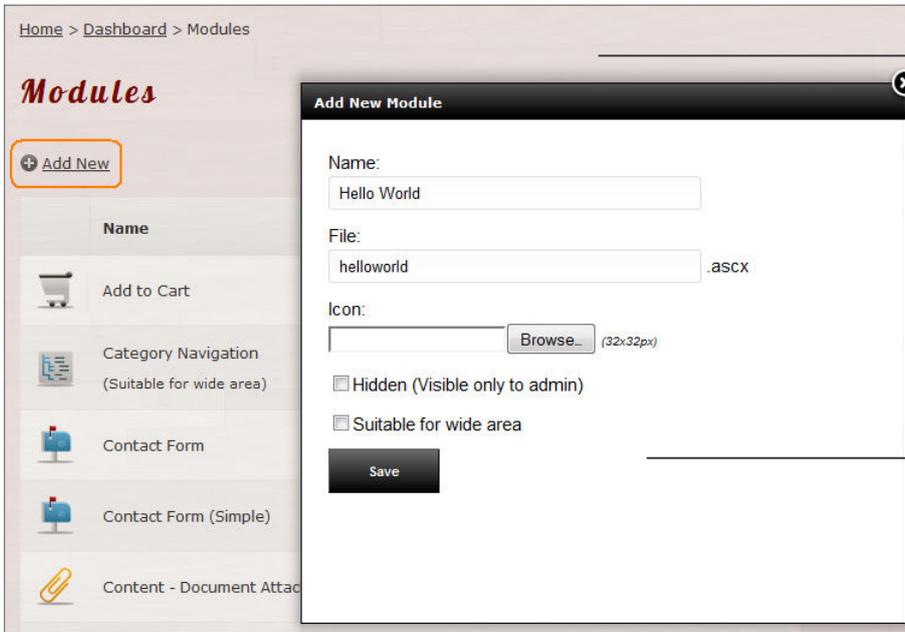
```
1 <@ Control Language="VB" Inherits="BaseUserControl" %>
2
3 <script runat="server">
4     Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
5         Literal1.Text = "Hello World!"
6     End Sub
7 </script>
8
9 <asp:Literal ID="Literal1" runat="server"></asp:Literal>
```

## Embedding a Module

First, you need to register the module.



1 Copy/FTP your module file into the **modules** folder on your website. In this example, the module file is **helloworld.ascx**.



**2** To register the module, open your **Dashboard**, then select the **Modules** link. A list of built-in modules will be displayed.

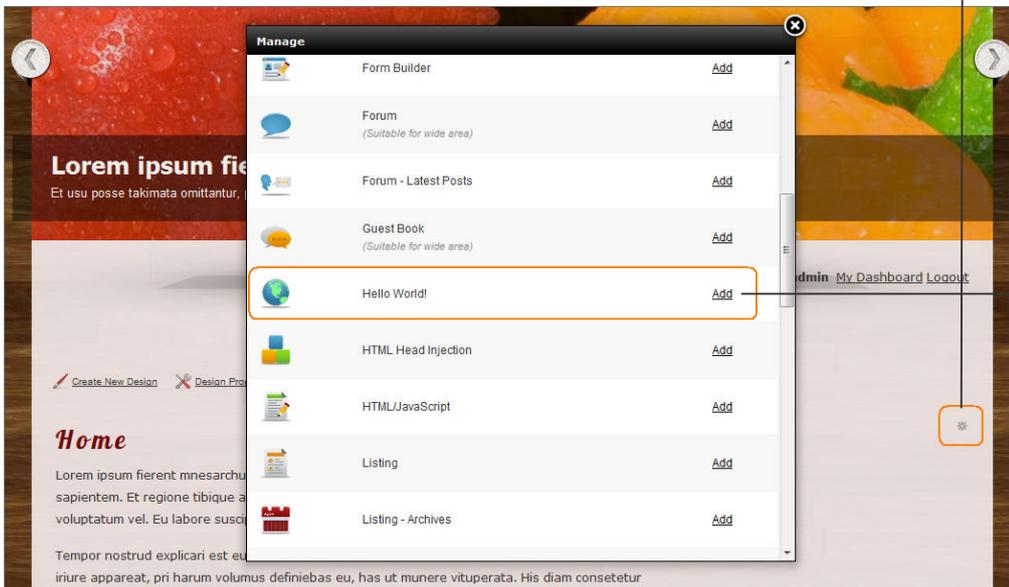
**3** Enter the module name and the ascx file name. Click **Save** to finish.

Now, your module is ready to use.

To be able to embed a module on a page, first go to the page where you want to embed your module. If you have an Administrator or Module Manager role, you will see the **Add module** link.

Select the **Add Module** icon to open the **Add Module** dialog.

**4**



**5**

You will see your module is listed here. Click the **Add** link next to the module.



**6**

Your module is now embedded.

# Creating an Advanced Module

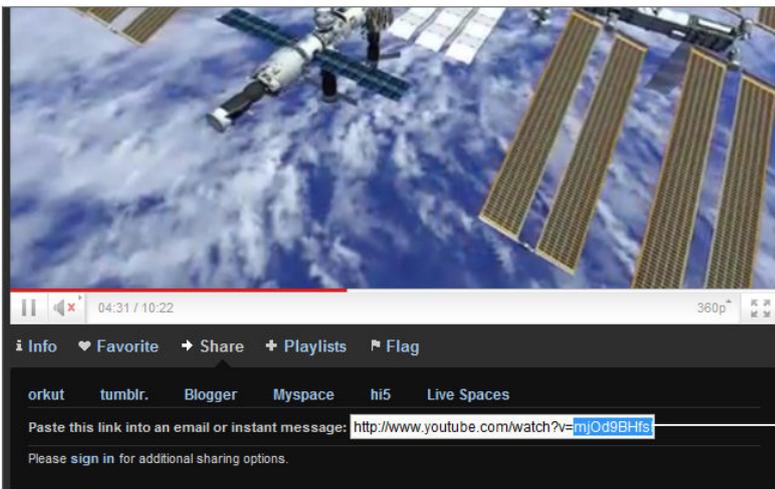
Using modules, you can enhance your SiteManager website with added applications.

A module/application sometimes requires some settings. For example, if you want to create a Youtube module, you would need to specify which video you want to display.

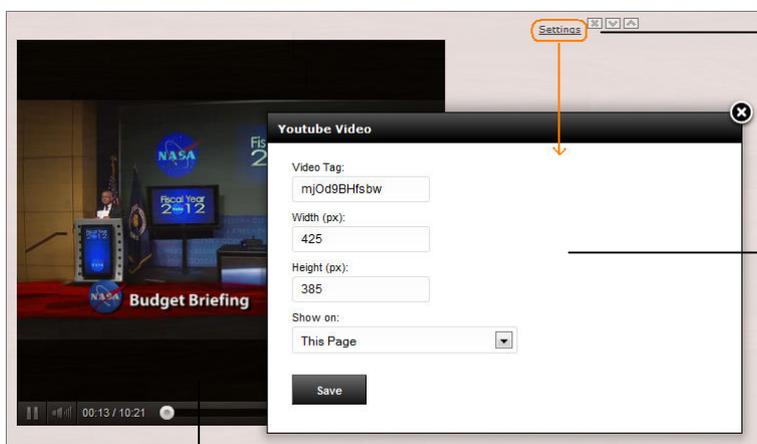
You can hard-code this information, but this can reduce the flexibility of your module.

SiteManager allows you to define your own variables in your module (for example: youtube video tag, width & height of the video, etc).

Here is an example of a module that allows you to specify a Youtube Video tag.



1 Video Tag from Youtube.



2 The **Settings** link is displayed on the module to open the **Settings** dialog.

3 The Video Tag is entered in the module's settings.

4 Youtube video will be displayed after saving the settings.

By using custom settings (in this example: Video Tag, width & height of the video), you can use the same module to display various Youtube videos with flexible dimensions for the video display.

For storing the settings data as well as retrieving it, SiteManager provides several methods you can use:

- SetModuleData()
- SaveModuleData()
- GetModuleData()

Below, you will see how to create the Youtube Video module. We have named the file **youtube.ascx**.

Here is the full listing of **youtube.ascx**:

```

<%@ Control Language="VB" Inherits="BaseUserControl"%>

<script runat="server">
    Protected Sub Page_Init(ByVal sender As Object, ByVal e As System.EventArgs)

        'STEP 1: Create your custom variables
        Dim sVideoTag As String = GetModuleData("VideoTag")
        Dim sWidth As String = GetModuleData("Width", 425)
        Dim sHeight As String = GetModuleData("Height", 385)
        Dim nShowOn As String = GetModuleData("ShowOn") 'ShowOn is Insite built-in (optional) variable.

        'STEP 2: Internal Module Setup (This is a Must)
        ModuleSetup(placeHolderModule, panelSettings, 470, 290)

        'STEP 3: Your module content here..
        If sVideoTag = "" Then
            litModule.Text = ""
        Else
            litModule.Text = "<object width="" & sWidth & "" height="" & sHeight & ""> & _
                "<param name=""movie"" value=""http://www.youtube.com/v/" & sVideoTag & "&hl=en&fs=1""></param>" & _
                "<param name=""wmode"" value=""opaque""></param>" & _
                "<param name=""allowFullScreen"" value=""true""></param>" & _
                "<param name=""allowscriptaccess"" value=""always""></param>" & _
                "<embed src=""http://www.youtube.com/v/" & sVideoTag & _
                "&hl=en&fs=1"" type=""application/x-shockwave-flash"" allowscriptaccess=""always"" & _
                " allowfullscreen=""true"" width="" & sWidth & "" height="" & sHeight & _
                "" wmode=""opaque""></embed></object><br /><br />"
        End If

        'STEP 4: Fill-in Module Setting Form with the custom variables
        If Not Page.IsPostBack Then
            txtVideoTag.Text = sVideoTag
            txtWidth.Text = sWidth
            txtHeight.Text = sHeight
            ddlShowOn.SelectedValue = nShowOn
        End If
    End Sub

    'STEP 5: Save variables
    Protected Sub btnSave_Click(ByVal sender As Object, ByVal e As System.EventArgs)
        SetModuleData("VideoTag", txtVideoTag.Text)
        SetModuleData("Width", txtWidth.Text)
        SetModuleData("Height", txtHeight.Text)
        SetModuleData("ShowOn", ddlShowOn.SelectedValue)
        SaveModuleData()
        Response.Redirect(HttpContext.Current.Items("page"))
    End Sub
</script>

<asp:Placeholder ID="placeholderModule" runat="server">
    <asp:Literal ID="litModule" runat="server"></asp:Literal>
</asp:Placeholder>

<asp:Panel ID="panelSettings" runat="server">
    <div style="padding:10px;padding-left:15px;">
    <fieldset class="web2small">
        <legend>
            <asp:Literal ID="litLegend" meta:resourcekey="litLegend" Text="Youtube Video" runat="server"></asp:Literal>
        </legend>

        <asp:Label ID="lblVideoTag" meta:resourcekey="lblVideoTag" runat="server"
            AssociatedControlID="txtVideoTag" Text="Video Tag:"></asp:Label>
        <asp:TextBox runat="server" id="txtVideoTag"></asp:TextBox>

        <asp:Label ID="lblWidth" meta:resourcekey="lblWidth" runat="server"
            AssociatedControlID="txtWidth" Text="Width (px):"></asp:Label>
        <asp:TextBox runat="server" id="txtWidth"></asp:TextBox>
        <asp:RegularExpressionValidator ID="rfv1" runat="server" ControlToValidate="txtWidth"
            ErrorMessage="*" ValidationExpression="[0-9]*" ValidationGroup="Settings">
        </asp:RegularExpressionValidator>

        <asp:Label ID="lblHeight" meta:resourcekey="lblHeight" runat="server"
            AssociatedControlID="txtHeight" Text="Height (px):"></asp:Label>
        <asp:TextBox runat="server" id="txtHeight"></asp:TextBox>
        <asp:RegularExpressionValidator ID="rfv2" runat="server" ControlToValidate="txtHeight"
            ErrorMessage="*" ValidationExpression="[0-9]*" ValidationGroup="Settings">
        </asp:RegularExpressionValidator>

        <asp:Label ID="lblShowOn" meta:resourcekey="lblShowOn" runat="server"
            AssociatedControlID="ddlShowOn" Text="Show on:"></asp:Label>
        <asp:DropDownList ID="ddlShowOn" runat="server"><!--The dropdown ID must be ddlShowOn-->
            <asp:ListItem Value="1" meta:resourcekey="optShowOn1" Text="This Page"></asp:ListItem>
            <asp:ListItem Value="2" meta:resourcekey="optShowOn2" Text="Child Pages"></asp:ListItem>
            <asp:ListItem Value="3" meta:resourcekey="optShowOn3" Text="This Page & Child Pages"></asp:ListItem>
            <asp:ListItem Value="4" meta:resourcekey="optShowOn4" Text="All Pages using the same template"></asp:ListItem>
        </asp:DropDownList>
        <br /><br />
        <asp:Button runat="server" id="btnSave" meta:resourcekey="btnSave" Text="Save" OnClick="btnSave_Click"></asp:Button>
    </fieldset>
    </div>
</asp:Panel>

```

1

Create your custom variables here. Use: **GetModuleData([Variable Name], [Optional:Default Value])**

2

Setup your module by calling: **ModuleSetup([Module Placeholder ID], [Setting Panel ID], [Setting Panel Width], [Setting Panel Height])**

3

Embed your youtube video in a Literal Control.

4

Fill-in your Module Settings Form with the custom variables.

5

Save custom variables here. Use: **SetModuleData([Variable Name], [Value])** And then call: **SaveModuleData()**

Note

A Placeholder to show the module. In this example we name the placeholder ID: **placeholderModule**.

Note

A Panel is used for the module's settings. In this example, we name the Panel ID: **panelSettings**.

Note

Use a dropdown to specify the visibility settings of your module. The selected value should be stored in variable **ShowOn** (See Step 5): **SetModuleData("ShowOn", ddlShowOn.SelectedValue)**

If your module requires user authorization, you can check the user's roles using the following properties:

- `IsUserLoggedIn`; checks if user has logged-in. Value: true/false.  
Example: **`Me.SiteManagerUser.IsUserLoggedIn`**
- `IsPublisher`; shows if user has a Publisher role. Value: true/false.  
Example: **`Me.SiteManagerUser.IsPublisher`**
- `IsSubscriber`; shows if user has a Subscriber role. Value: true/false.  
Example: **`Me.SiteManagerUser.IsSubscriber`**
- `IsAuthor`; shows if user has an Author role. Value: true/false.  
Example: **`Me.SiteManagerUser.IsAuthor`**
- `IsEditor`; shows if user has an Editor role. Value: true/false.  
Example: **`Me.SiteManagerUser.IsEditor`**
- `IsResourceManager`; shows if user has a Resource Manager role. Value: true/false.  
Example: **`Me.SiteManagerUser.IsResourceManager`**
- `IsAdministrator`; shows if user has an Administrator role. Value: true/false.  
Example: **`Me.SiteManagerUser.IsAdministrator`**
- `IsReader`; shows if user doesn't have any role assigned or has a Subscriber role. Value: true/false.  
Example: **`Me.SiteManagerUser.IsReader`**

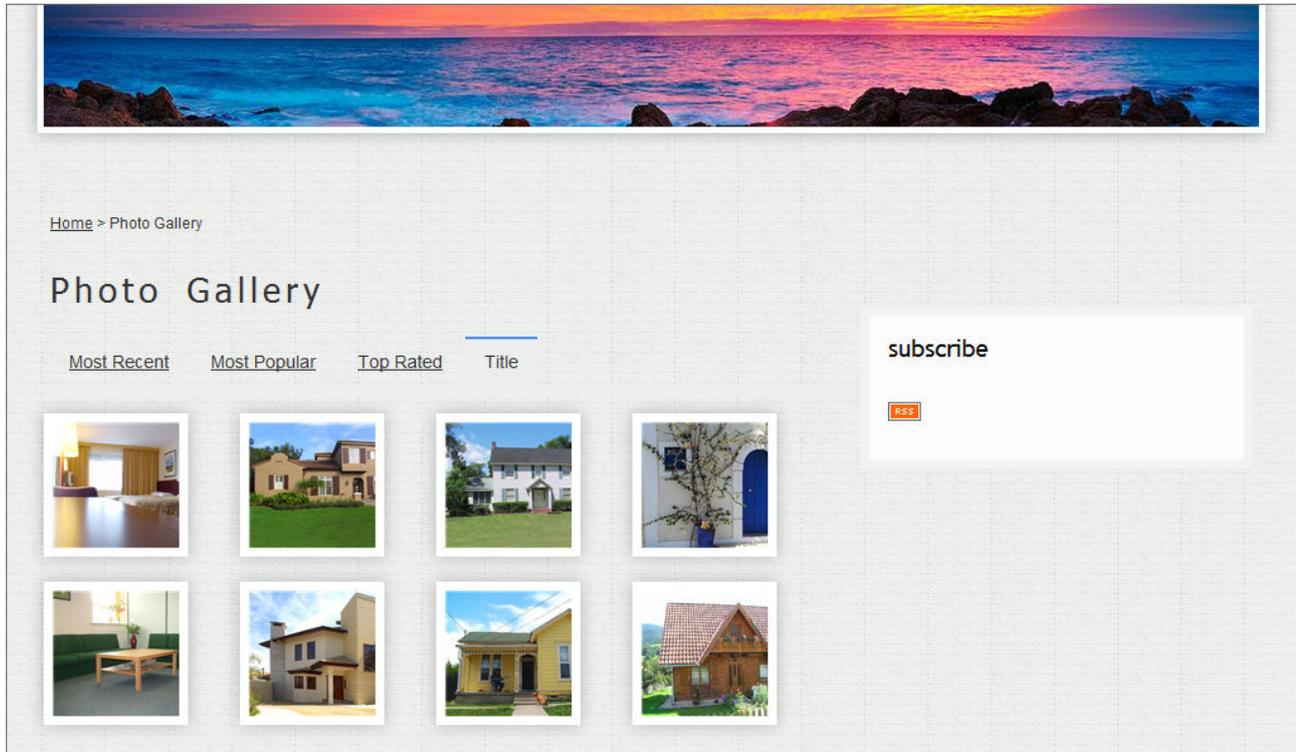
A custom module can also obtain several items of page/site information:

- `PageID`; ID of the current page where the module is embedded.  
Example: **`Me.SiteManagerPage.page_id`**
- `Culture`; Culture name of the current page where the module is embedded.  
Example: **`Me.SiteSetting.Culture`**  
Sample value: 'en-US' or 'de-DE' (depends on the culture setting of the current site - See the **Site Settings** panel)
- `RootID`; ID of the Home page. Example:  
**`Me.SiteManagerPage.root_id`**

# Listing Template Development

Listings in SiteManager are template-driven. With a listing template you can format the look and feel of your listing and specify its behavior.

Below is an example of the “Photo Gallery” listing template applied to a page:



# Creating a Listing Template

Listing templates are created in the same way as site templates – by inserting pre-defined placeholders in html code, and defining styles in css as you would with any other website.

As a simple example of a listing, let's look at a bulleted list:

```
<ul>
<li>Title 1 here..</li>
<li>Title 2 here..</li>
<li>Title 3 here..</li>
</li>
```

This bulleted list can be divided into 3 parts:

- 1) Header section: <ul>
- 2) Repeating Item section: <li>Title...</li>
- 3) Footer section: </ul>

To create the listing template, go to the **Dashboard** and select the **Listing Templates** menu.

The screenshot shows the 'Listing Templates' dashboard. A modal window titled 'Add New Listing Template' is open. The 'Name' field contains 'Bulleted List'. The 'Copy from template' dropdown is set to 'Blank Template'. A 'Create' button is visible at the bottom of the modal. Two red callouts are present: '1' points to the 'Name' field with the text 'In this example, we named the listing template: **Bulleted List**.' and '2' points to the 'Create' button with the text 'Click **Create**.'

Home > Dashboard > Listing Templates

## Listing Templates

Name:

Visible to:

Make available on all mini sites.

**BASIC LAYOUT**   ADVANCED SETTINGS   SCRIPT   ENTRIES   CATALOGING

Layout:  
 Flow (clean, naturally inline)  
 Tabular  

Records per page:

Paging Style:

<< < 1 2 3 4 5 > >>

Header:

Repeating Item:

Footer:

[Customize your Elements](#)

[View SQL Script](#)

[Back to List](#)

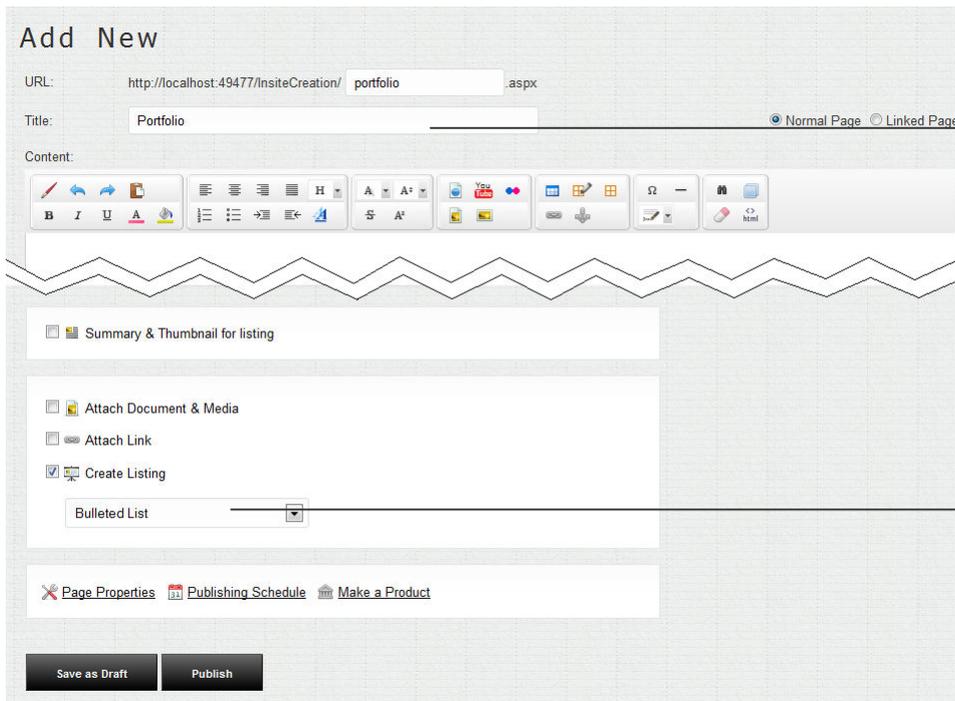
**3** The Listing Templates Panel is displayed.

**4** Fill in the Header section here.

**5** Fill in the Repeating Item section here. In this example, we use the [%TITLE%] tag to show the page title.

**6** Fill in the Footer section here.

**7** Click **Save & finish**. Now, we are ready to use the listing template.



8

Try creating a new page. In this example we create a Portfolio page.

9

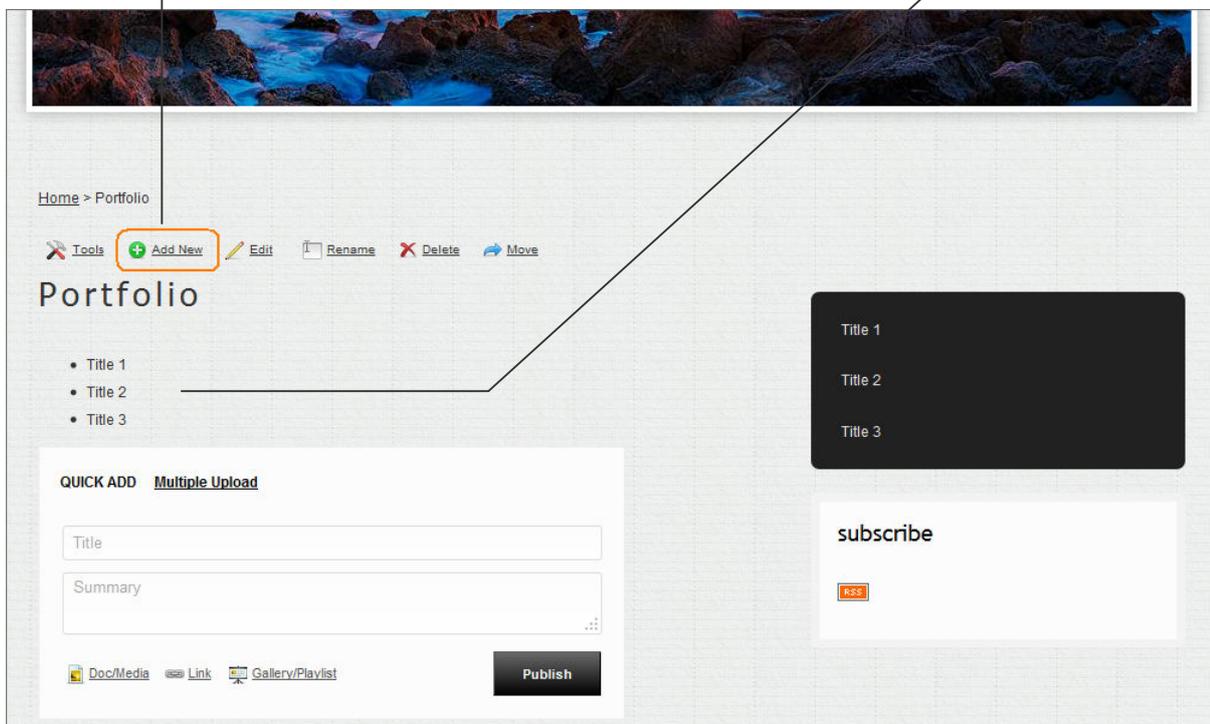
Check the **Make this page a listing/gallery** option. Choose **Bulleted List** and publish the page.

10

Under the new Portfolio page, create another new page by selecting the **Add New** link. In this example, we create 3 child pages: Title 1, Title 2, Title 3.

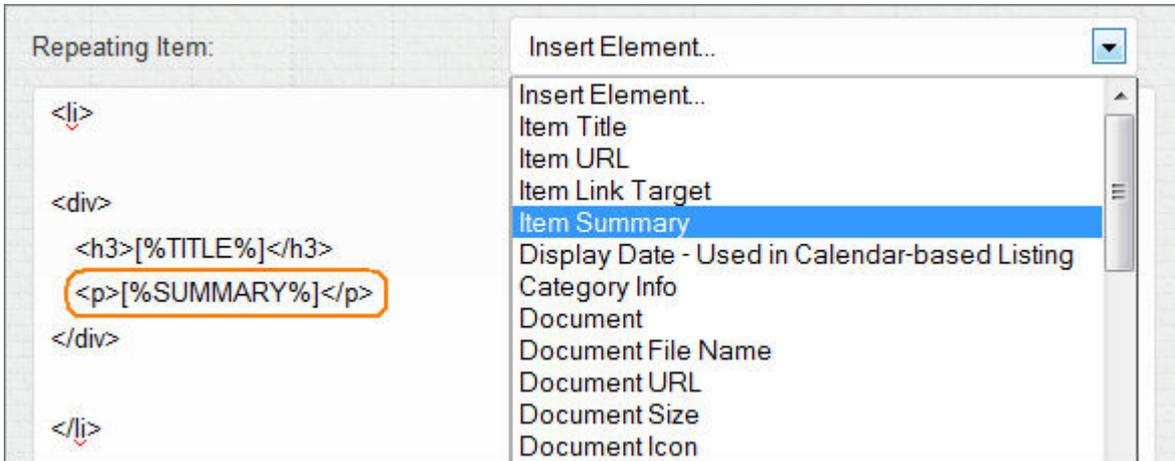
11

Here is the listing with the **Bulleted List** template applied.



# Displaying Summaries

To display summary text for each listing item, insert the [%SUMMARY%] tag on the Repeating Item section.

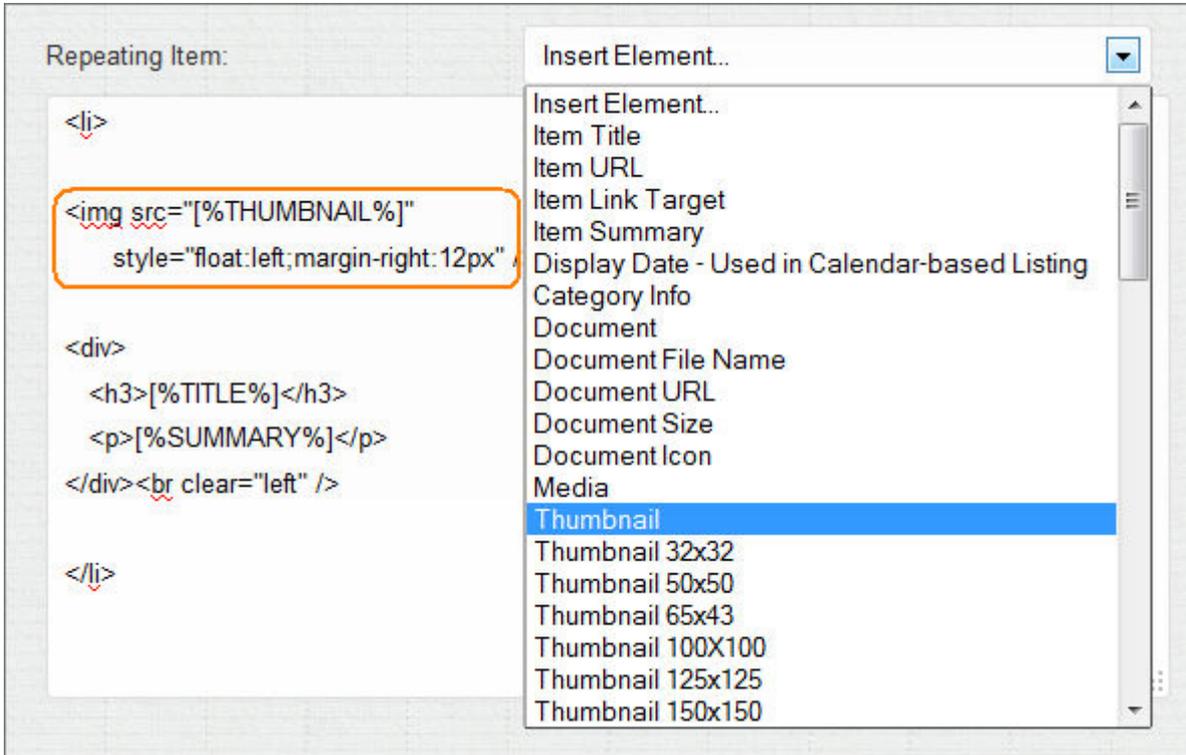


In the example, we also added few formatting, by adding `<h3>` for the title and `<p>` for the summary. The result is shown below:

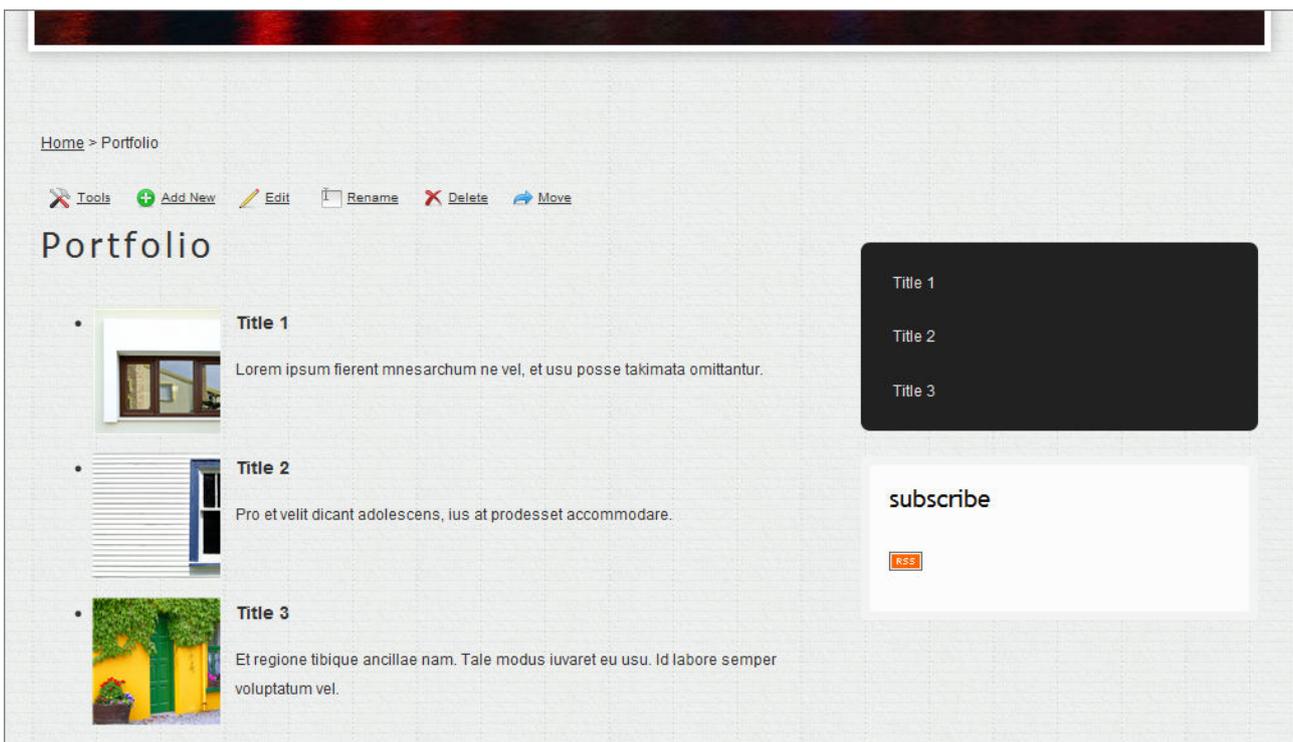


# Displaying Thumbnails

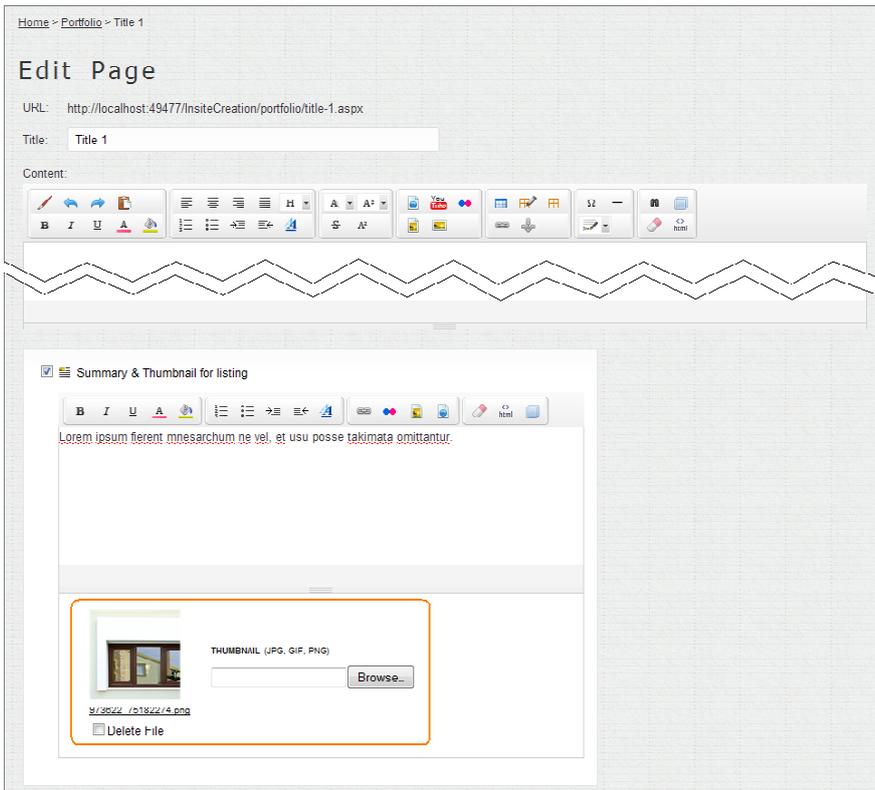
To display Thumbnails on the listing, use the [%THUMBNAIL%] tag on the **Repeating Item** section. [%THUMBNAIL%] will be replaced with the Image URL of the thumbnail.



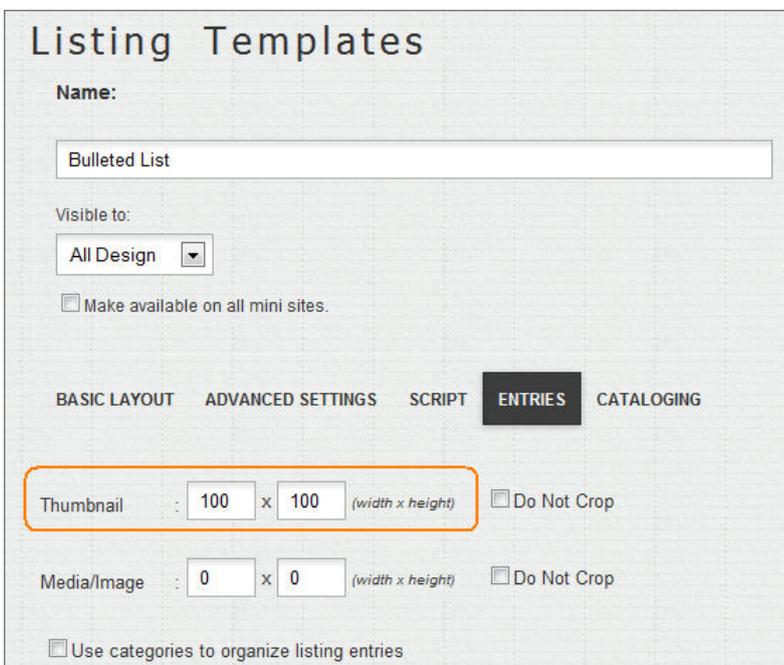
If your pages have a thumbnail attachment, it will be displayed in the listing as shown below:



To attach or update a thumbnail, edit the page and upload your image file in the **Thumbnail** section.

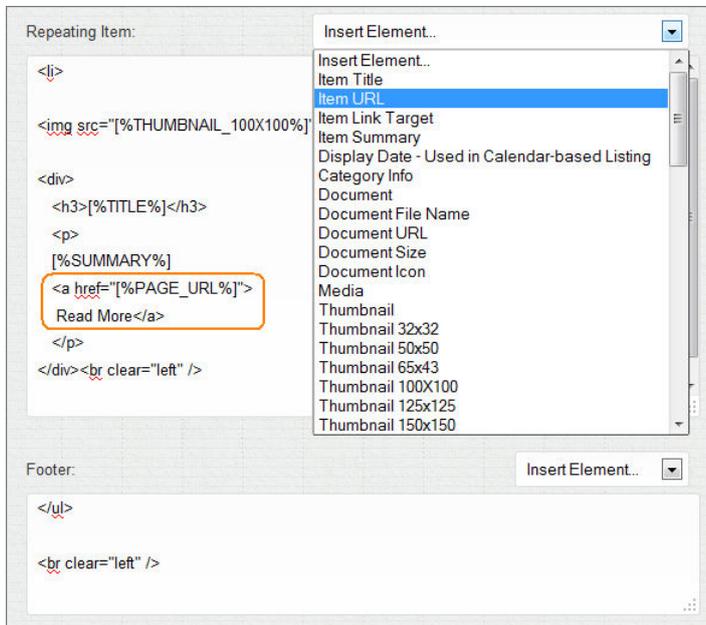


You can specify the thumbnail dimensions (width x height) by selecting the **Entries** Tab on the Listing Template.



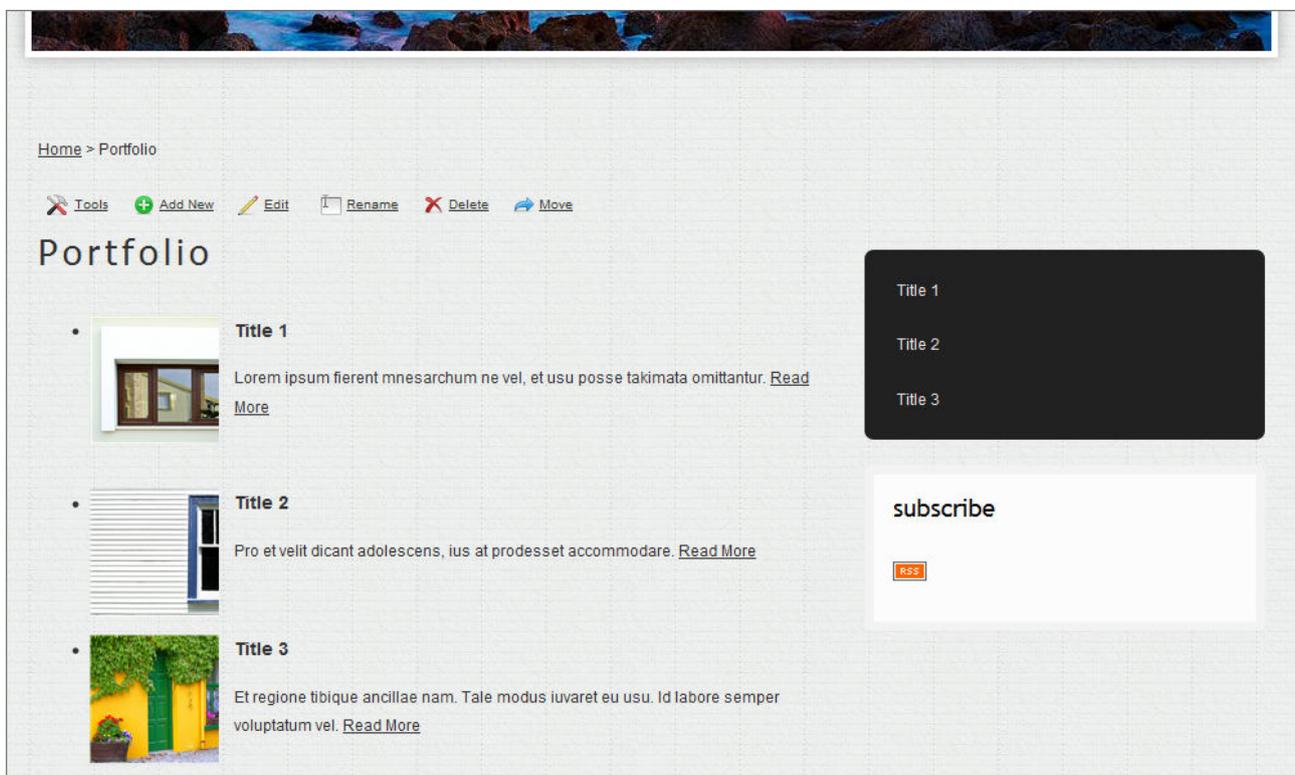
## Adding “Read More” Links

A “Read More” link connects each listing item to the relevant page. To add a “Read More” link on the listing, use the [%PAGE\_URL%] tag on the **Repeating Item** section. [%PAGE\_URL %] will be replaced with URL of the listing item.



The screenshot shows the 'Repeating Item' editor interface. On the left, the HTML code for the repeating item is displayed. A red box highlights the code: `<a href="[%PAGE_URL%]">Read More</a>`. On the right, the 'Insert Element..' dropdown menu is open, showing a list of available tags. 'Item URL' is selected and highlighted in blue. Other visible tags include 'Item Title', 'Item Link Target', 'Item Summary', 'Display Date - Used in Calendar-based Listing', 'Category Info', 'Document', 'Document File Name', 'Document URL', 'Document Size', 'Document Icon', 'Media', 'Thumbnail', and various thumbnail sizes (32x32, 50x50, 65x43, 100x100, 125x125, 150x150). Below the main editor, a 'Footer' section is visible with its own 'Insert Element..' dropdown.

The result is shown below:



The screenshot displays a website portfolio page. At the top, there is a navigation breadcrumb 'Home > Portfolio' and a toolbar with icons for 'Tools', 'Add New', 'Edit', 'Rename', 'Delete', and 'Move'. The main heading is 'Portfolio'. Below it, there are three listing items, each with a thumbnail image, a title, and a short description followed by a 'Read More' link. The items are:

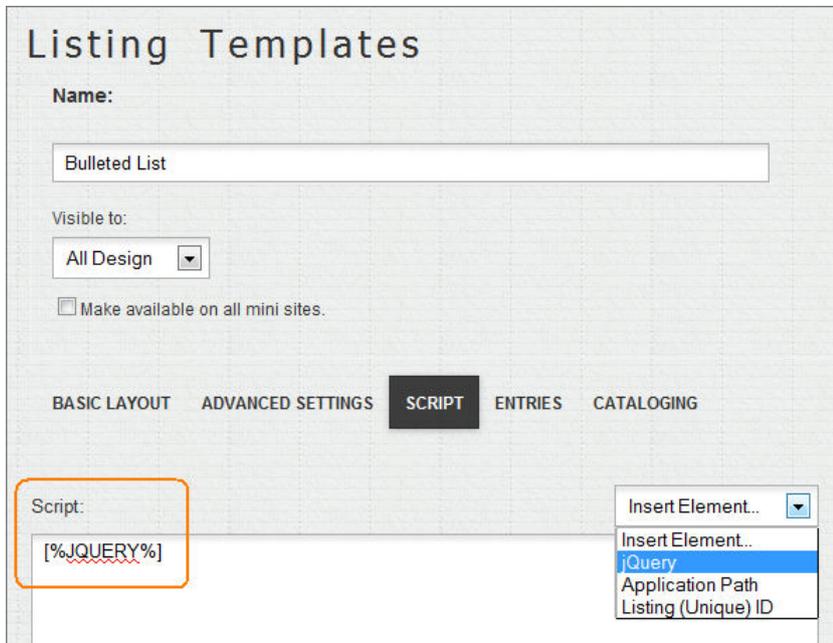
- Title 1**: Lorem ipsum fierent mnesarchum ne vel, et usu posse takimata omittantur. [Read More](#)
- Title 2**: Pro et velit dicant adolescens, ius at prodesset accommodare. [Read More](#)
- Title 3**: Et regione tibique ancillae nam. Tale modus iuvaret eu usu. Id labore semper voluptatum vel. [Read More](#)

On the right side of the page, there is a dark sidebar with three items labeled 'Title 1', 'Title 2', and 'Title 3'. Below the sidebar, there is a white box with the text 'subscribe' and an 'RSS' button.

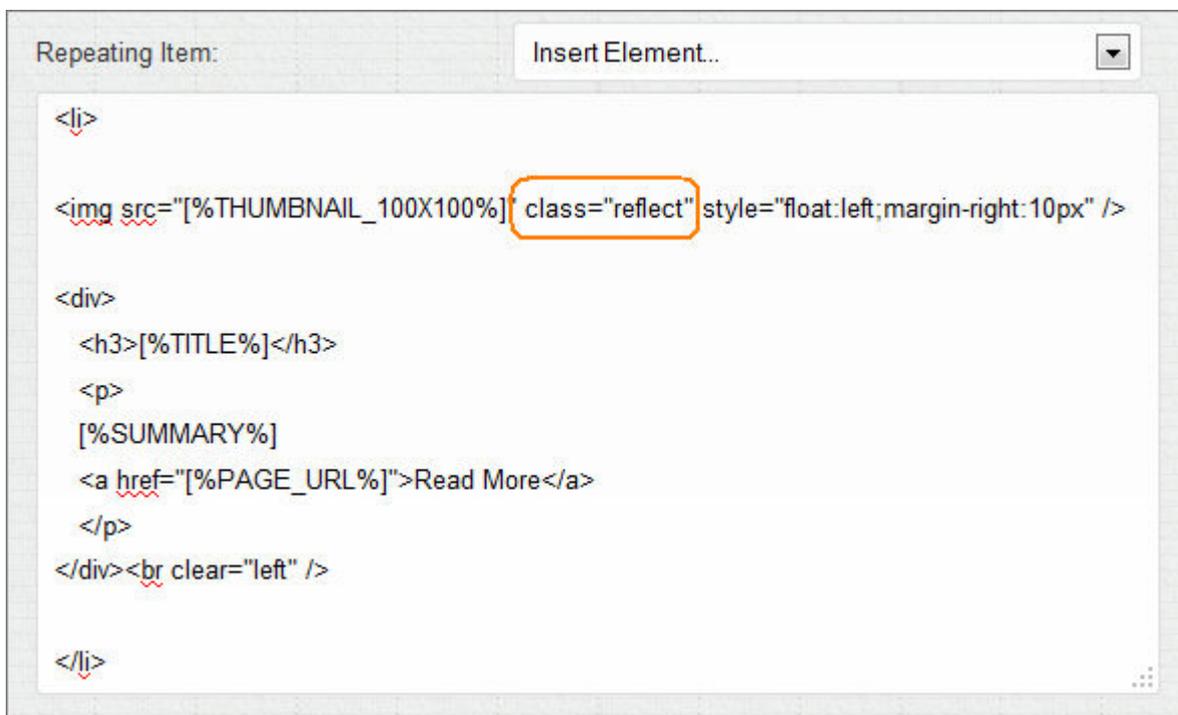
## Adding Reflection & Lightbox

SiteManager has a built-in feature to display an image with a reflection effect and to open the hyperlink using a Lightbox. This feature requires JQuery.

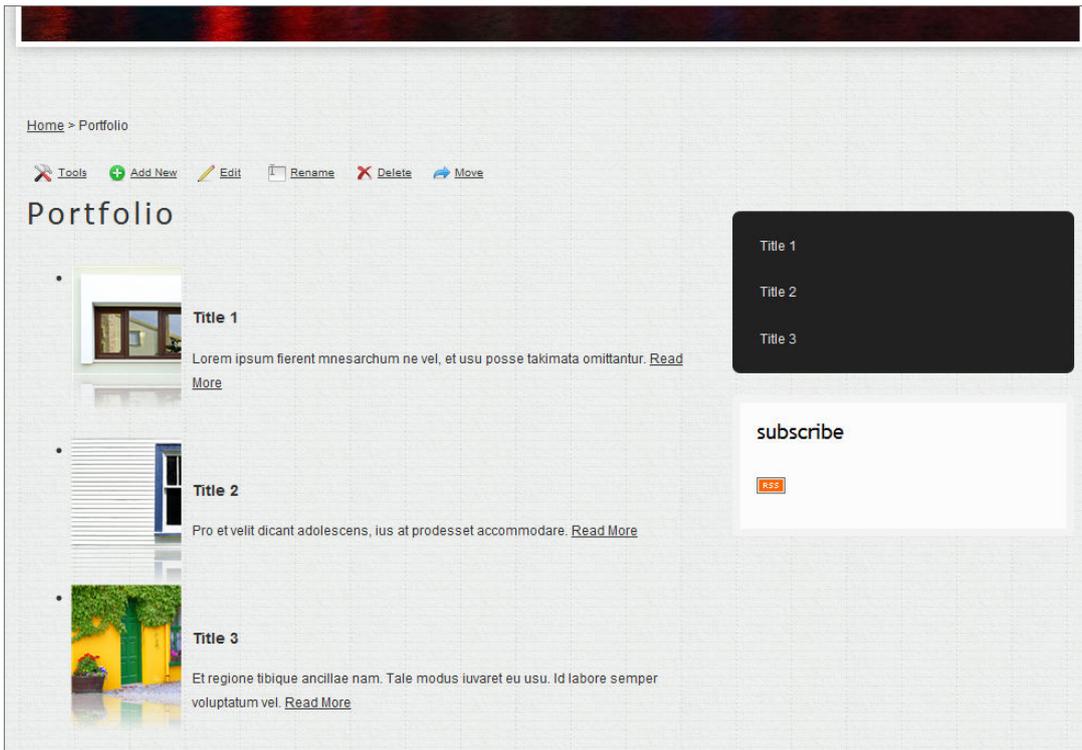
To include JQuery in your listing, use the [%JQUERY%] tag on the Script section.



Add **class="reflect"** on the **IMG** element to enable the reflection effect.



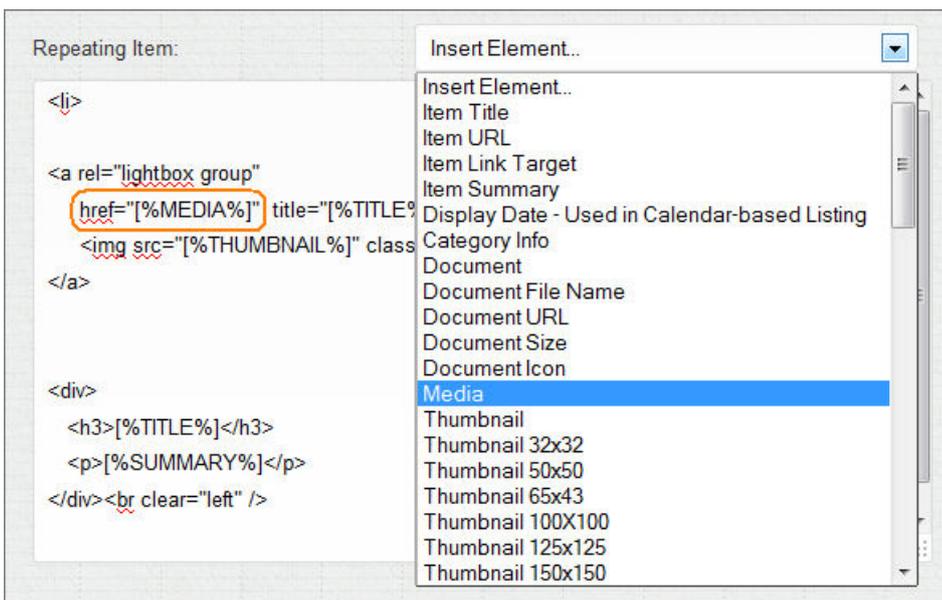
The reflection effect is now displayed on each thumbnail within your listing.



Here we add a hyperlink that opens the actual (larger) Image. The actual image we use is the Media attachment of the page.

```
<a rel="lightbox group" href="[%MEDIA%]" title="[%TITLE%]">
  
</a>
```

[%MEDIA%] will be replaced with the Media attachment URL



Note that you can attach or update the Media attachment by editing a page and uploading your image file in the **Media** section.

The screenshot shows the 'Edit Page' interface for a page titled 'Title 1'. The URL is `http://localhost:49477/InsiteCreation/portfolio/title-1.aspx`. The page content area contains a rich text editor with the text 'Lorem ipsum fierent mnesarchum ne vel. et usu posse takimata omittantur.' Below the text is a section for adding a thumbnail, labeled 'THUMBNAIL (JPG, GIF, PNG)', with a 'Browse...' button and a 'Delete File' checkbox. The thumbnail image is `973622_75182274.png`. Below this is a section for 'Attach Document & Media', which has two 'Browse...' buttons for 'DOCUMENT' and 'MEDIA (MP4, FLV, MP3, JPG, GIF, PNG)'. The media attachment section is highlighted with an orange border and shows the same thumbnail image with the filename `973622_75182274.jpg` and a 'Delete File' checkbox. At the bottom left, there are checkboxes for 'Attach Link' and 'Create Listing'.

Then, to use the Lightbox, add attribute: **rel="lightbox"** or **rel="lightbox group"** on the Media attachment hyperlink (**group** means the lightbox will display a series of images).

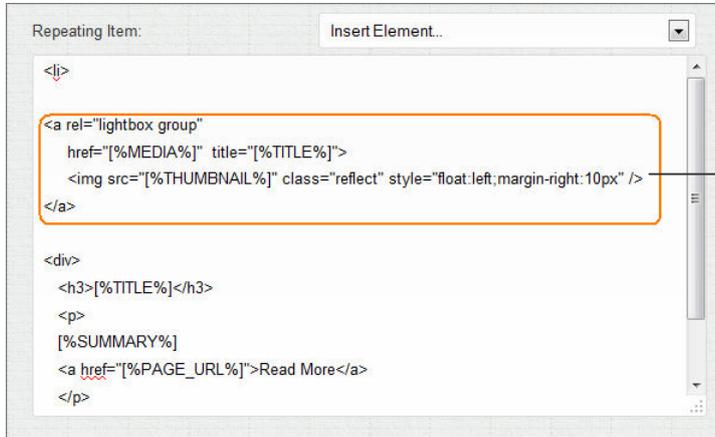
```
Repeating Item: Insert Element... ▼  
  
</li>  
<a rel="lightbox group"  
  href "[%MEDIA%]" title "[%TITLE%]">  
  <img src "[%THUMBNAIL%]" class="reflect" style="float:left;margin-right:10px" />  
</a>  
  
<div>  
  <h3>[%TITLE%]</h3>  
  <p>[%SUMMARY%]</p>  
  <a href "[%PAGE_URL%]">Read More</a>  
</div><br clear="left" />
```

Now each thumbnail on your listing will open within a lightbox showing the larger image from the Media attachment.



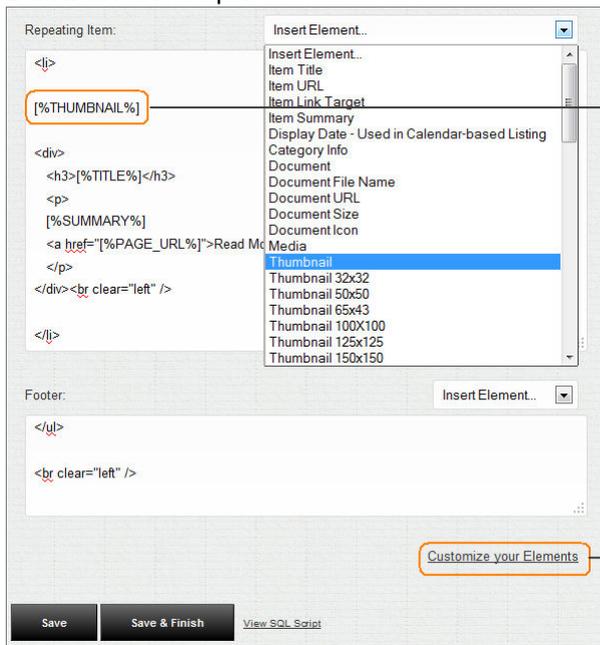
## If Thumbnail is not attached (use Element Templates)

If a thumbnail is not attached on a listing entry page, the [%THUMBNAIL\_URL%] tag will be replaced with an empty string. This results in the IMG element not having a source value. To prevent this, you can use an Element Template feature.



With an Element Template, you can completely hide this block if the thumbnail does not exist.

Below are the steps:



**1** Replace the block with the [%THUMBNAIL%] tag.

**2** Select the **Customize your Elements** link.

Then we'll create an Element Template for the [%THUMBNAIL%] tag.



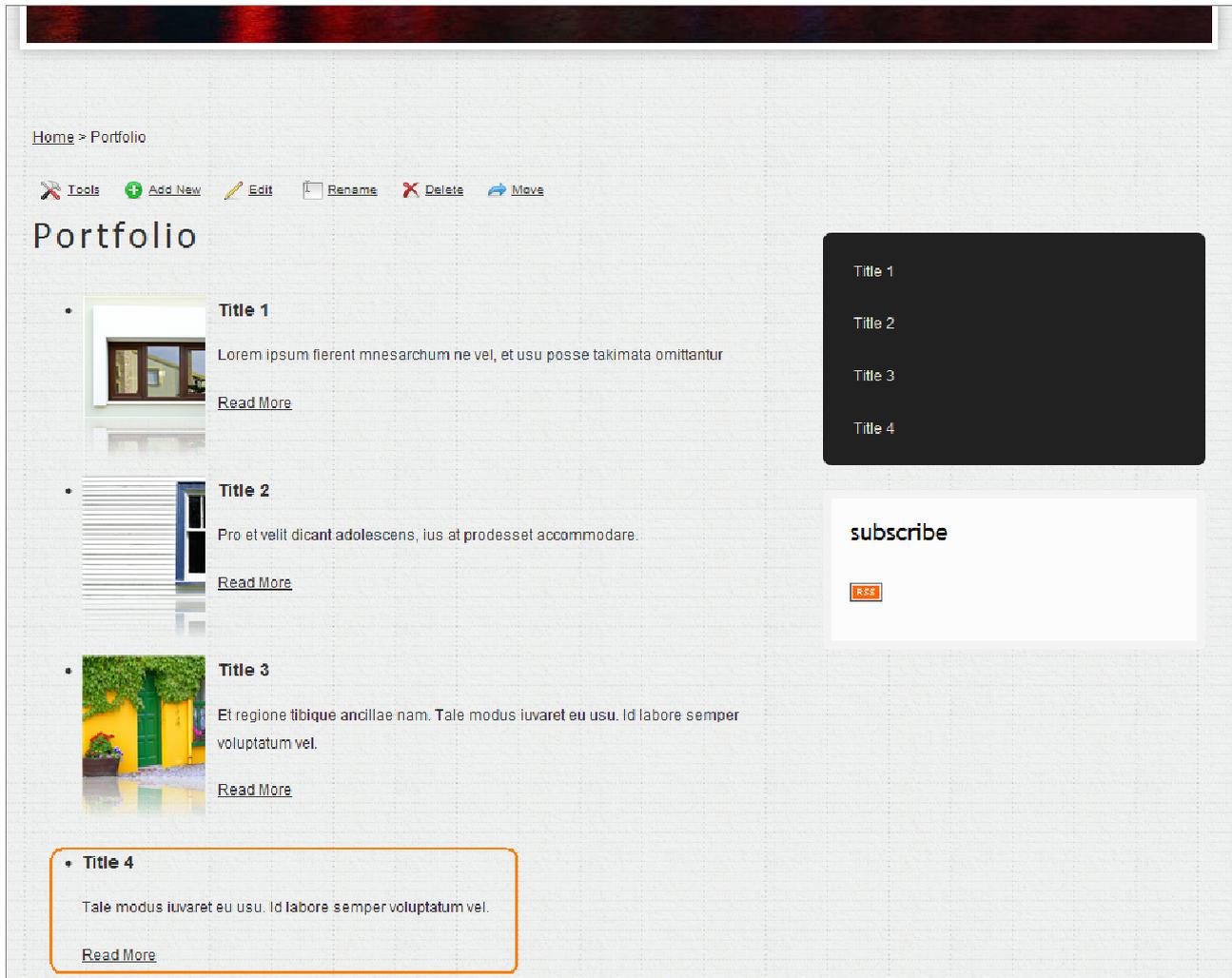
**3** Select **Thumbnail** from the dropdown.

**4** Enter the rule:  
.jpg|.JPG|.gif|.GIF|.png|.PNG  
With this rule applied, the template will only be used if there is an image attachment.

**5** Fill in the previous thumbnail block here.

**6** Click **Save**.

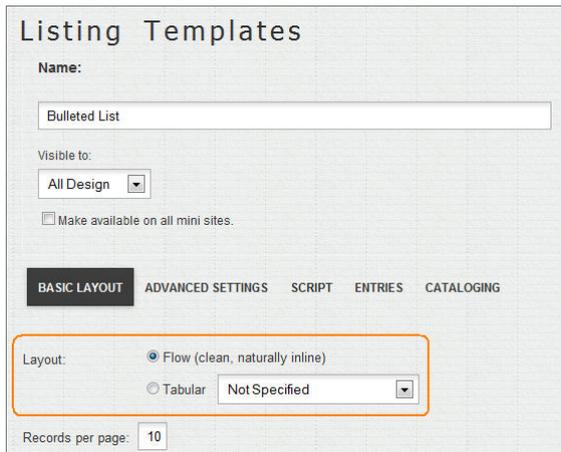
Now, if a user doesn't attach a thumbnail image, the thumbnail block will not be displayed.



# Listing Layout

Listings in SiteManager have two different layout types:

1. Tabular layout; the Listing will be formatted as a Table. This layout allows you to specify the number of columns.
2. Flow layout; the Listing will not be formatted. It is intended for a simple, non tabular listing. This type of listing is sometimes used with a Javascript framework (ex. JQuery) to format the listing.



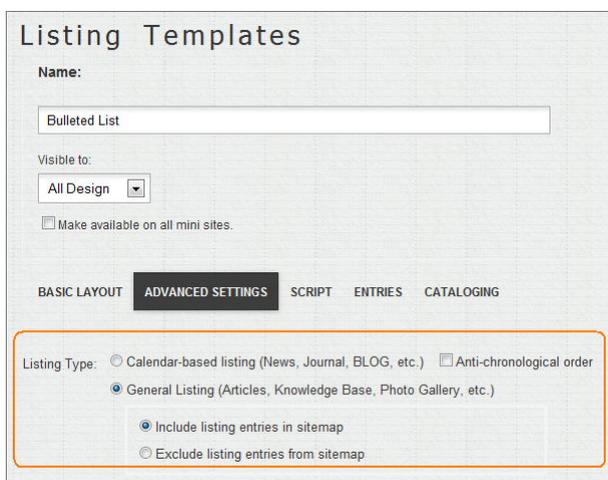
The screenshot shows the 'Listing Templates' configuration page. The 'Name' field is 'Bulleted List'. The 'Visible to' dropdown is 'All Design'. There is a checkbox for 'Make available on all mini sites'. Below this are tabs for 'BASIC LAYOUT', 'ADVANCED SETTINGS', 'SCRIPT', 'ENTRIES', and 'CATALOGING'. The 'BASIC LAYOUT' tab is active. Under 'Layout:', there are two radio buttons: 'Flow (clean, naturally inline)' (selected) and 'Tabular'. Next to 'Tabular' is a dropdown menu showing 'Not Specified'. Below this is a 'Records per page' field set to '10'.

# Listing Types: Calendar-based & General Listing

SiteManager provides two basic types of listing depending on the nature of your content:

1. Calendar-based Listing: for managing periodically updated content such as newsletters, journals, blogs, etc.
2. General listing: for general purpose listings such as directories, FAQ pages, knowledge base pages, photo galleries, or even e-commerce enabled product lists.

The listing type setting is provided on the **Advanced Settings** Tab.



The screenshot shows the 'Listing Templates' configuration page with the 'ADVANCED SETTINGS' tab selected. The 'Name' field is 'Bulleted List'. The 'Visible to' dropdown is 'All Design'. There is a checkbox for 'Make available on all mini sites'. Below this are tabs for 'BASIC LAYOUT', 'ADVANCED SETTINGS', 'SCRIPT', 'ENTRIES', and 'CATALOGING'. The 'ADVANCED SETTINGS' tab is active. Under 'Listing Type:', there are two radio buttons: 'Calendar-based listing (News, Journal, BLOG, etc.)' and 'General Listing (Articles, Knowledge Base, Photo Gallery, etc.)' (selected). There is also a checkbox for 'Anti-chronological order'. Below these are two radio buttons: 'Include listing entries in sitemap' (selected) and 'Exclude listing entries from sitemap'.

# Calendar-based Listing

An example of a Calendar-based listing is a blog page:

[Home](#) > [Blog](#)

## Blog

Lorem ipsum fierent mnesarchum ne vel, et usu posse takimata omittantur, pro ut tale erant sapientem. Et regione tibique ancillae nam. Tale modus iuaret eu usu. Id labore semper voluptatum vel. Eu labore suscipit nam, in simul qualisque pri.

Search within the list:

All Dates  From Last 24 Hours  From Last Week  From Last Month  From Last 60 days

[Most Recent](#) [Most Popular](#) [Last Commented](#)



Et regione tibique ancillae  
4 minutes ago  
[MORE >>](#)

### categories

- BUSINESS (1)
- DEVELOPMENT (0)

[All Categories](#)

[Manage Categories](#)

### calendar

October 2011							
	Sun	Mon	Tue	Wed	Thu	Fri	Sat
	25	26	27	28	29	30	1
	2	3	4	5	6	7	8
	9	10	11	12	13	14	15
	16	17	18	19	20	21	22
	23	24	25	26	27	28	29
	30	31	1	2	3	4	5

[Show All](#)

### subscribe

In a Calendar-based listing, when you add new pages as listing entries, you'll be presented with a Display Date input field.

Home > Blog

## Add New

URL:

Title:

Normal Page  Linked Page

Content:

Category:   
Business  
Development

Display Date:  Automatic  
 Oct 5 2011

Summary & Thumbnail for listing

THUMBNAIL (JPG, GIF, PNG)

Attach Document & Media  
 Attach Link  
 Create Listing

[Page Properties](#) [Publishing Schedule](#) [Make a Product](#)

[Back to Page](#)

By default, the listing will be ordered by the specified Display Date. You can display more ordering options in the Listing Template:

## Listing Templates

**Name:**

BLOG (65x43 Thumbnail)

Visible to:

All Design

Make available on all mini sites.

BASIC LAYOUT   **ADVANCED SETTINGS**   SCRIPT   ENTRIES   CATALOGING

Listing Type:  Calendar-based listing (News, Journal, BLOG, etc.)  Anti-chronological order  
 General Listing (Articles, Knowledge Base, Photo Gallery, etc.)

Include listing entries in sitemap  
 Exclude listing entries from sitemap

Manual order  
 Default order by:

Most Recent (based on Publish Date)

**Listing Functions:**

Show Ordering:

Most Recent (based on Display Date)    Top Downloads  
 Most Recent (based on Last Updated Date)    Today's Top Downloads  
 Most Recent (based on Publish Date)    Download Size  
 Most Popular (based on Total Hits)    Item Title  
 Today's Most Popular (based on Hits Today)    Author (based on Owner)  
 Top Rated    Author (based on Person Last Updating)  
 Last Commented    Price  
 Top Commented

Show Search  
 Show Date Range Filter  
 Use Expandable View (requires AJAX-enabled)

If you choose to display the Ordering feature, select **Most Recent (based on Display Date)** since a Calendar-based listing is ordered by the Display Date.

Here the Ordering feature is displayed with the **Most Recent** tab, which is selected by default.

Home > Blog

## Blog

Lorem ipsum fierent mnesarchum ne vel, et usu posse takimata ornitantur, pro ut tale erant sapientem. Et regione tibique ancillae nam. Tale modus iuvaret eu usu. Id labore semper voluptatum vel. Eu labore suscipit nam, in simul qualisque pri.

Search within the list:  **Search**

All Dates  From Last 24 Hours  From Last Week  From Last Month  From Last 60 days

**Most Recent**   Most Popular   Last Commented

 Et regione tibique ancillae  
4 minutes ago  
[MORE >>](#)

**categories**

- BUSINESS (1)
- DEVELOPMENT (0)

[All Categories](#)  
[Manage Categories](#)

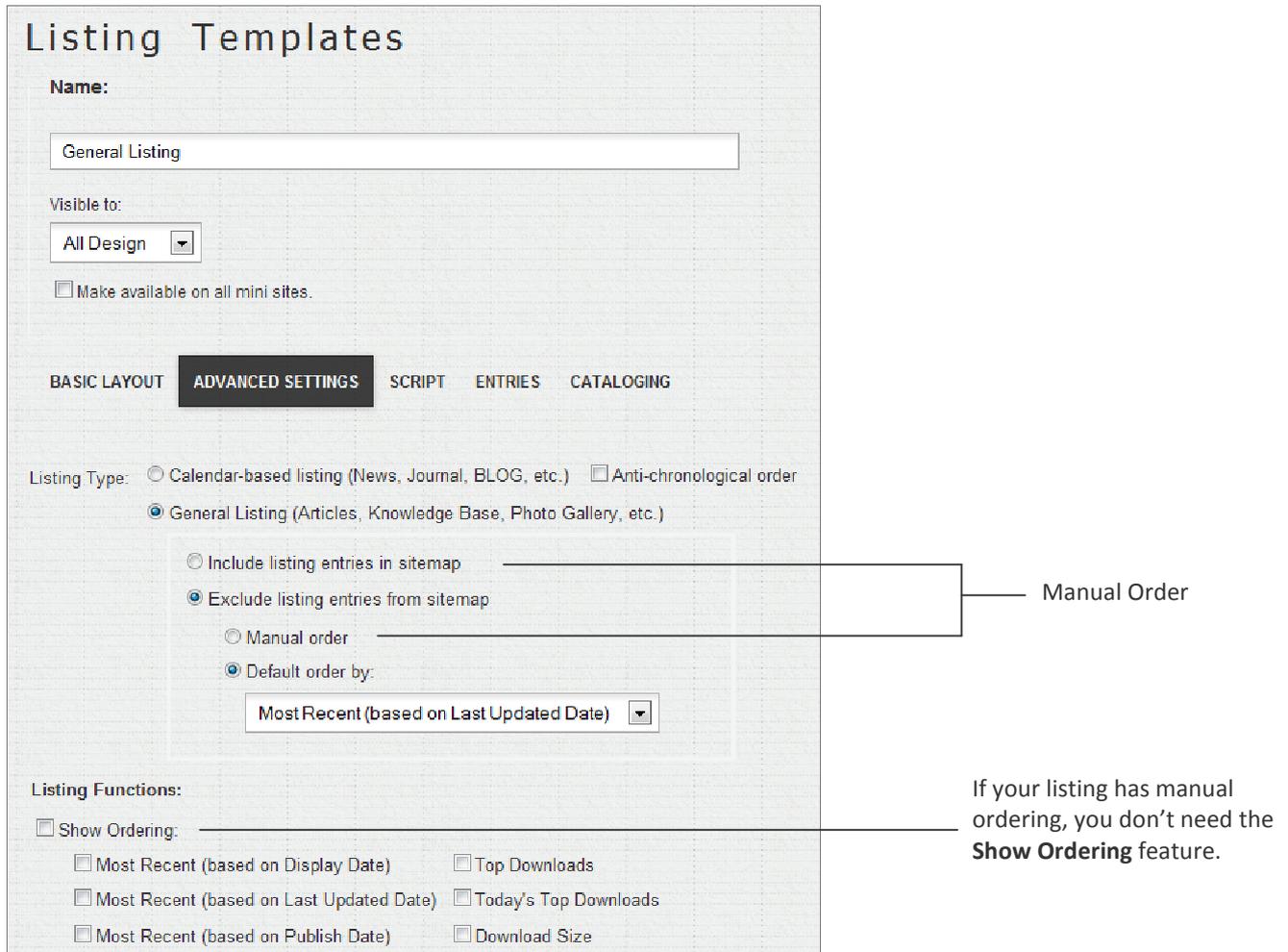
**calendar**

October 2011							
>>	Sun	Mon	Tue	Wed	Thu	Fri	Sat
<	25	26	27	28	29	30	1

# General Listings

Depending on your content, you may choose whether General Listing entries are included in the sitemap or excluded from it.

If you choose Exclude listing entries from sitemap, your listing entries will not be displayed on the site menus/navigation. This option is recommended for managing a large number of entries.



The screenshot shows the 'Listing Templates' configuration page, specifically the 'ADVANCED SETTINGS' tab. The 'Name' field is 'General Listing'. The 'Visible to' dropdown is set to 'All Design'. There is a checkbox for 'Make available on all mini sites.' The 'Listing Type' section has three options: 'Calendar-based listing (News, Journal, BLOG, etc.)', 'Anti-chronological order', and 'General Listing (Articles, Knowledge Base, Photo Gallery, etc.)', with the latter selected. Under 'General Listing', there are three radio button options: 'Include listing entries in sitemap', 'Exclude listing entries from sitemap', and 'Manual order'. The 'Exclude listing entries from sitemap' option is selected. Below this is a dropdown menu for 'Default order by:' set to 'Most Recent (based on Last Updated Date)'. The 'Listing Functions' section has a 'Show Ordering' checkbox, which is unchecked. Below it are six checkboxes for different ordering methods: 'Most Recent (based on Display Date)', 'Top Downloads', 'Most Recent (based on Last Updated Date)', 'Today's Top Downloads', 'Most Recent (based on Publish Date)', and 'Download Size'. Two annotations with lines pointing to the 'Manual order' radio button and the 'Show Ordering' checkbox are present. The first annotation is labeled 'Manual Order' and the second is labeled 'If your listing has manual ordering, you don't need the Show Ordering feature.'

If you choose to include the listing on the sitemap, you'll need to specify the page ordering manually when adding or editing a listing entry page. This manual order feature is also presented if your listing entries are excluded from the sitemap and you have selected **Manual Order** (as shown in the above picture).

Home > Portfolio

## Add New

URL:

Title:

Normal Page  Linked Page

Content:

Display Order:

Title 1	▲
Title 2	▲
Title 3	▲
Title 4	▲
>><<	▲

Up

Down

Category:

Summary & Thumbnail for listing

<BODY> <BR>

THUMBNAIL (JPG, GIF, PNG)

Attach Document & Media

Attach Link

Create Listing

Page Properties  Make a Product

[Back to Page](#)

Manual Page Order setting is displayed when you add or edit a page.

# Listing Entries

In the Listing Template, you can also configure settings for your listing entries. Select the **Entries** tab to configure the settings.

Home > Dashboard > Listing Templates

## Listing Templates

Name:

Visible to:

Make available on all mini sites.

**BASIC LAYOUT**   **ADVANCED SETTINGS**   **SCRIPT**   **ENTRIES**   **CATALOGING**

Thumbnail :  x  (width x height)    Do Not Crop

Media/Image :  x  (width x height)    Do Not Crop

Use categories to organize listing entries

One-Level entries

**Authoring Functions:**

Hide "Quick Add"    Hide "Make a Product" setting

Hide "Multiple Upload"    Hide "Side Content" setting

Hide "Summary" editor    Hide "Attach Document & Media"

Hide "Content" editor    Hide "Attach Link"

Hide "Page Properties" setting    Hide "Create Listing"

Hide "Schedule" setting

Limit allowed sub-listings:

- BLOG (200x150 Thumbnail)
- BLOG (440x200 Thumbnail)
- BLOG (590x220 Thumbnail)
- BLOG (65x43 Thumbnail)
- Bulleted List

     [View SQL Script](#)

[Back to List](#)

Specify thumbnail dimensions to display on your listing.

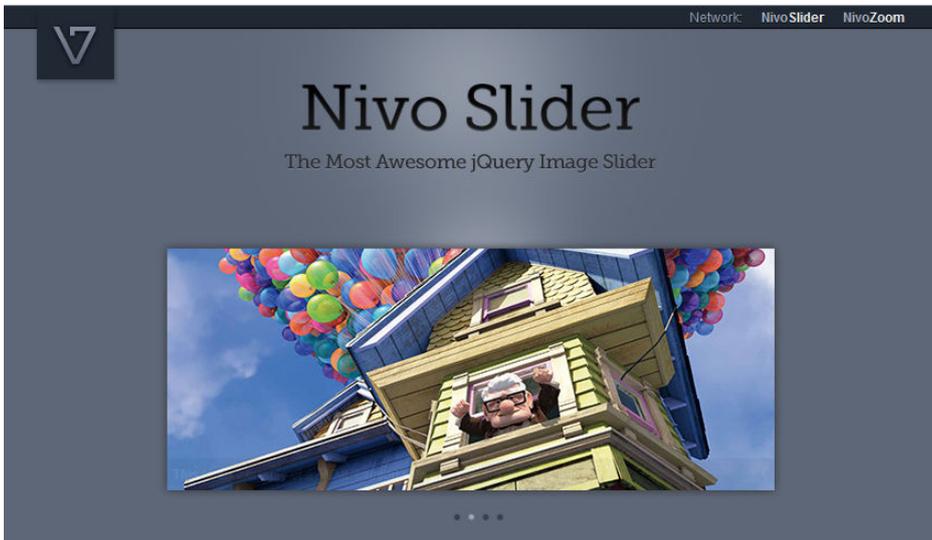
The **Use categories to organize listing entries** option allows you to organize listing entries using custom categories.

It's possible to create sub-listings within a listing. This setting is used if you want to limit which listings can be created under the current listing.

# Listing Script – using JQuery

You can use a Listing in SiteManager to create a creative presentation using a JavaScript framework and its plugins (e.g. JQuery).

Here, we'll look at how to create a listing presentation using the Nivo Slider script, which requires JQuery (<http://nivo.dev7studios.com/>).



First, copy the Nivo Slider files into folder **systems/scripts/nivo-slider/**. Then include the scripts and css in your listing by specifying the includes in the **Script** section of the Listing Template.

Listing Templates

Name:  
Photo Gallery (Sliding)

Visible to:  
All Design   
 Make available on all mini sites.

BASIC LAYOUT   ADVANCED SETTINGS   **SCRIPT**   ENTRIES   CATALOGING

Script:

```
[%JQUERY%]  
<link rel="stylesheet" href "[%APP_PATH%]systems/scripts/nivo_slider/css" type="text/css" n  
<script src "[%APP_PATH%]systems/scripts/nivo_slider/jquery.nivo_slider.pack.js" type="text/j  
  
<script type="text/javascript">  
  $(document).ready(function(){  
    $('#slider').nivoSlider();  
  });  
</script>
```

Disable Ajax

Script Templates

Save   Save & Finish   [View SQL Script](#)

1

Include Nivo's script & css.

2

Apply the slider using:  
\$ (#slider).nivoSlider()  
to element with ID="slider".

**Note:** Please refer to the JQuery documentation for further info regarding this implementation.

Now here is the header, footer, and repeating item sections:

The screenshot shows the 'Listing Templates' configuration interface. At the top, the 'Name' field contains 'Photo Gallery (Sliding)'. Below it, 'Visible to' is set to 'All Design'. There are tabs for 'BASIC LAYOUT', 'ADVANCED SETTINGS', 'SCRIPT', 'ENTRIES', and 'CATALOGING'. The 'BASIC LAYOUT' tab is active, showing options for 'Layout' (Flow selected), 'Records per page' (10), and 'Paging Style' (Style A-01). The 'Header' section contains the code: `<div id="slider" style="width:550px">`. The 'Repeating Item' section contains the code: ``. The 'Footer' section contains the code: `</div>`. At the bottom, there are buttons for 'Save', 'Save & Finish', and 'View SQL Script'.

2 Give ID="slider" to the div element.

3 Here we use Media (Image) attachment.

And that's all! You can see the result by applying the listing template on any page you want.

Note that this listing template requires Media (Image) attachment in specific dimension (800 x 500 px). To specify the Media attachment dimension, go to **Entries** Tab.

The screenshot shows the 'Listing Templates' configuration interface. The 'Name' field contains 'Photo Gallery (Sliding)'. The 'Visible to' dropdown is set to 'All Design'. There is a checkbox for 'Make available on all mini sites.' Below this are tabs for 'BASIC LAYOUT', 'ADVANCED SETTINGS', 'SCRIPT', 'ENTRIES', and 'CATALOGING', with 'ENTRIES' being the active tab. Under the 'ENTRIES' tab, there are two rows for dimensions: 'Thumbnail' with width 270 and height 250, and 'Media/Image' with width 800 and height 500. Both rows have a 'Do Not Crop' checkbox. At the bottom, there is a checkbox for 'Use categories to organize listing entries'.

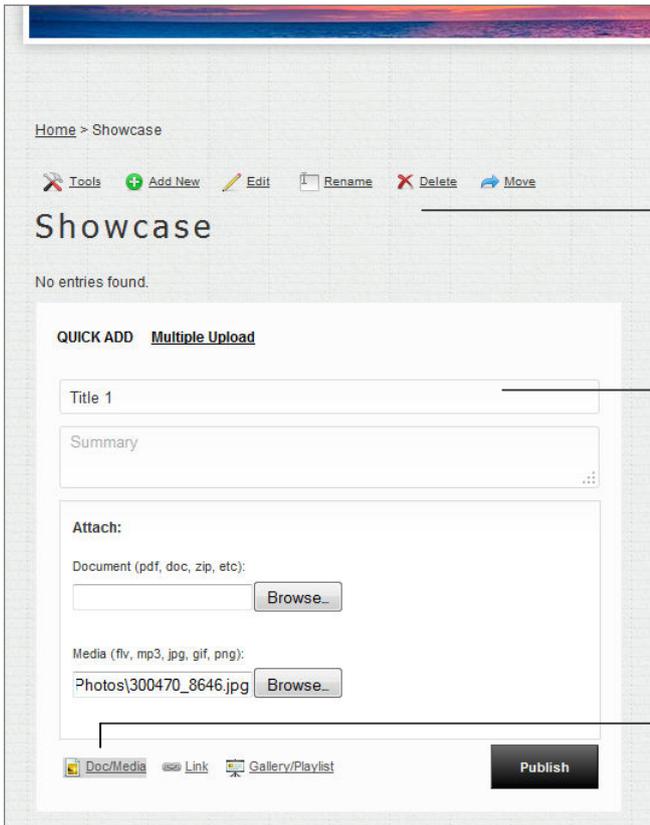
You can also set content template for listing entries. This allows you to specify a template to format your content title, body, attachment preview, etc. In **Design Mode**, open a listing entry page, click on **Design Properties** and then select **Content Template** tab.

The screenshot shows the 'Design Properties' dialog box with the 'CONTENT TEMPLATE' tab selected. The 'Content Template' field contains the following HTML code:

```
<h1 class="title">[%TITLE%] [%CATEGORY_BREADCRUMB%]</h1>
<!-- <div class="meta">By <span>[%OWNER%]</span>, Updated: [%DISPLAY_DATE%]</div> -->
<div class="body">
<div>[%CONTENT_BODY%]</div>
<div>[%ATTACH_LINK%]</div>
</div>
<div>[%MEDIA%]</div>
<div>[%DOCUMENT%]</div>
<div>[%LISTING%]</div>
```

Below the code field, there is a 'Media/Image' dimension field with width 100 and height 100, and a 'Do Not Crop' checkbox. There is also a 'Custom Tags' link. At the bottom right, there are 'Save' and 'Save & Finish' buttons.

Try create a page and apply the listing. Then add child pages with Media (image) attachment. You can use the **Quick Add** panel.

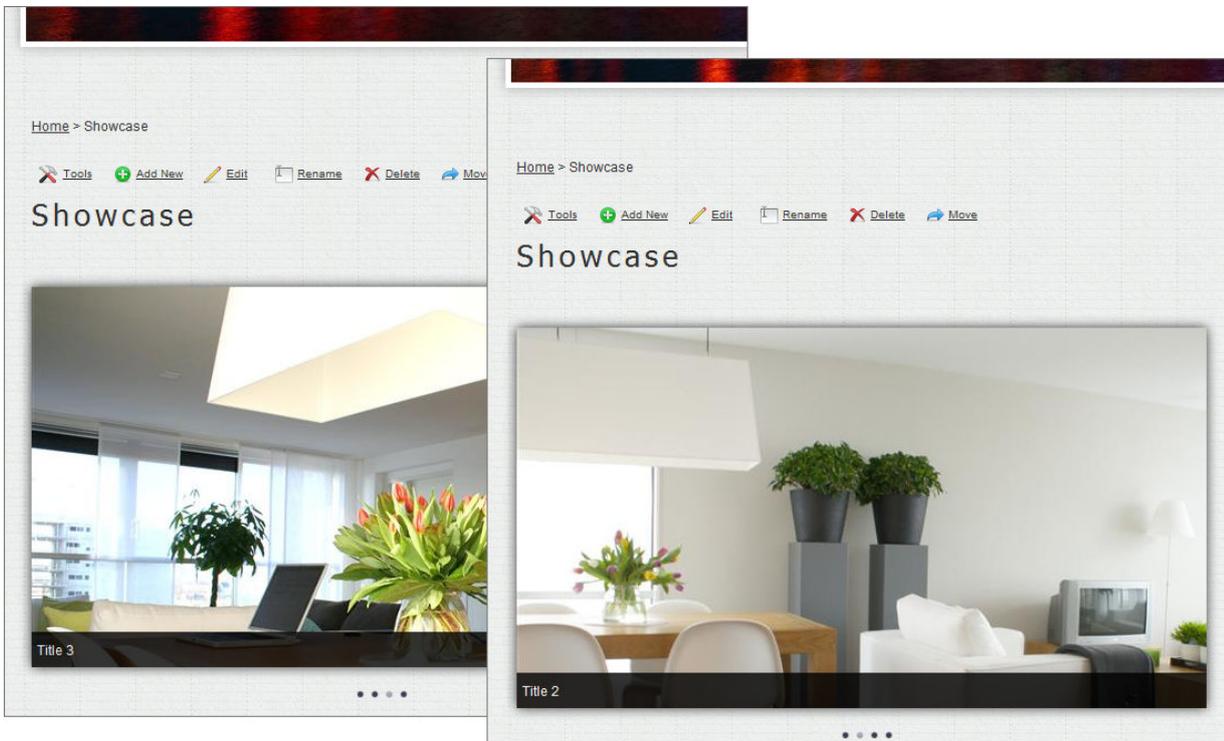


Here we create a page with title **Showcase**. Apply the listing template to this page.

Add child pages here.

Attach the image file as Media attachment. It will be resized to 800x500px according to the setting we've specified before.

Here is the result:



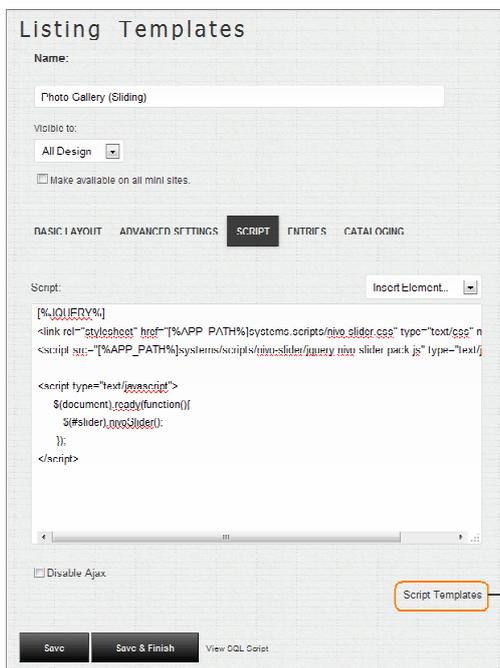
With the listing script, you can integrate many custom scripts or javascript framework plugins to format your listing and create creative presentations.

# Script Templates

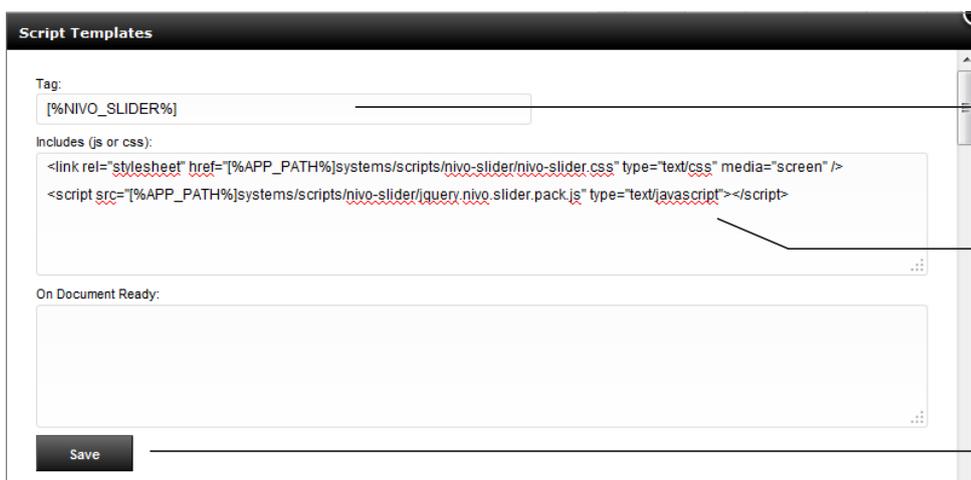
Sometimes we use several Listing Templates on a single page, for example if we embed **Listing** module multiple times on a page. This results in the listing script being embedded multiple times. Consider the previous example, in which we have the Nivo script and css:

```
<link rel="stylesheet" href="[%APP_PATH%]systems/scripts/nivo-slider/nivo-slider.css" type="text/css" media="screen" />
<script src="[%APP_PATH%]systems/scripts/nivo-slider/jquery.nivo.slider.pack.js" type="text/javascript"></script>
```

To prevent the above js and css being embedded multiple times, we use **Script Templates**.



1 Select **Script Templates**.

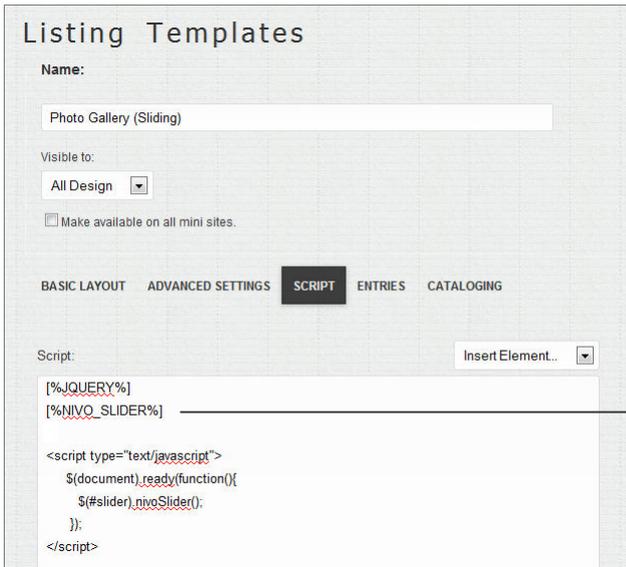


2 Create a new Tag: [%NIVO\_SLIDER%]

3 Move the include js and css here.

4 Click **Save**.

Now you can use the new [%NIVO\_SLIDER%] tag on the Script Template. This will ensure that the js and css includes will be embedded once, even if the listing template is used multiple times on a page.

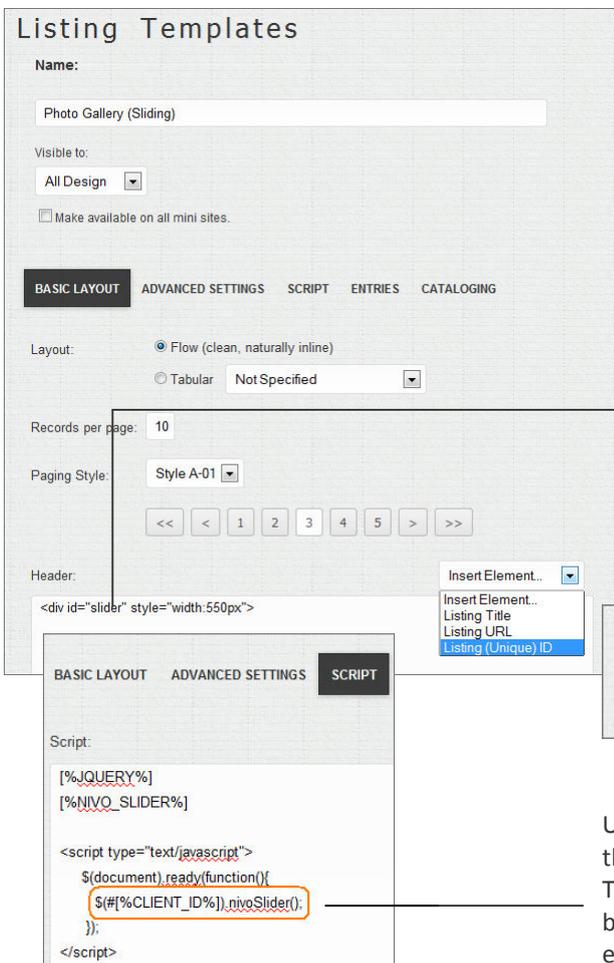


5 Use the new [%NIVO\_SLIDER%] tag.

The last step is to remove any hardcoded element ID which can generate multiple elements having the same ID. In this example, we have div element with ID="slider". We don't want multiple elements with the same ID="slider" embedded on a page (in case we use this listing template multiple times). There are 2 solutions:

1. Use the [%CLIENT\_ID%] tag
2. Use inline Class selectors instead of ID selectors

Option 2 is easier, but we will see first the option 1.



We can replace this ID with a unique ID by selecting **Listing (Unique) ID** from the dropdown. If this listing template is used multiple times on a page, the [%CLIENT\_ID%] will give a unique ID to each div element.

Use the [%CLIENT\_ID%] tag in the Script Template too. This will ensure the slider will be applied to the unique div element.

However, this option will not work with the Nivo Slider script since the Nivo's css has a hardcoded ID selector.

```
systems/scripts/niv...ider/nivo-slider.css X
/* The Nivo Slider styles */
#slider {
    position:relative;
    min-height:220px;
    margin:30px 30px 60px 30px;
    background:#202834 url(loading.gif) no-repeat 50% 50%;
    -moz-box-shadow:0px 0px 10px #333;
    -webkit-box-shadow:0px 0px 10px #333;
    box-shadow:0px 0px 10px #333;
}
#slider img {
    position:absolute;
    top:0px;
    left:0px;
    display:none;
}
```

We will use option 2 by modifying the above css from ID selector to class selector:

```
systems/scripts/niv...ider/nivo-slider.css X
/* The Nivo Slider styles */
.slider {
    position:relative;
    min-height:220px;
    margin:30px 30px 60px 30px;
    background:#202834 url(loading.gif) no-repeat 50% 50%;
    -moz-box-shadow:0px 0px 10px #333;
    -webkit-box-shadow:0px 0px 10px #333;
    box-shadow:0px 0px 10px #333;
}
.slider img {
    position:absolute;
    top:0px;
    left:0px;
    display:none;
}
```

Here we use:  
**.slider**  
instead of  
**#slider**.

Now we can use class, instead of ID for the div element:

Listing Templates

Name: Photo Gallery (Sliding)

Visible to: All Design

Make available on all mini sites.

BASIC LAYOUT | ADVANCED SETTINGS | SCRIPT | ENTRIES

Layout:  Flow (clean, naturally inline)  Tabular Not Specified

Records per page: 10

Paging Style: Style A-01

Header: `<div class="slider" style="width:550px">`

Script:

```
[%JQUERY%]
[%NIVO_SLIDER%]
<script type="text/javascript">
$(document).ready(function(){
  $('.slider').nivoSlider();
});
</script>
```

Now we use:  
**class="slider"**  
instead of:  
**id="slider"**

Remember to change your JQuery to use the class selector to apply the Nivo Slider. Here we use:  
**\$( '.slider' ).nivoSlider ();**  
Instead of:  
**\$( '#slider' ).nivoSlider ();**

# Listing Cataloging

Cataloging allows you to specify custom fields for your listing to create a custom catalog/database. Listing Cataloging enables you to create a variety of useful lists, for example a Book Library, Product List, Hotel & Property Database, or Inventory.

### Listing Templates

**Name:**

**Visible to:**

Make available on all mini sites.

**BASIC LAYOUT   ADVANCED SETTINGS   SCRIPT   ENTRIES   CATALOGING**

**Custom Fields**

Field 1	Property Type	Choice	House, Apartment, Villa, Land	<input checked="" type="checkbox"/> Searchable
Field 2	Bedrooms	Choice	Studio, 1, 2, 3, 4, 5+	<input checked="" type="checkbox"/> Searchable
Field 3	Price	Number		<input checked="" type="checkbox"/> Searchable
Field 4		Text Memo		<input type="checkbox"/> Searchable
Field 5		Number		<input type="checkbox"/> Searchable
Field 6		Checkbox		<input type="checkbox"/> Searchable
Field 7		Choice		<input type="checkbox"/> Searchable
Field 8		Checklist		<input type="checkbox"/> Searchable

### Add New

URL:

Title:

Property Type:

Bedrooms:

Price:

Content:

**B I U A**   **H**   **A A²**   **YouTube**   **Ω**   **html**

### Find Properties

Search within the list:

All Dates    From Last 24 Hours    From Last Week    From Last Month    From Last 60 days

[Most Recent](#)   [Most Popular](#)   [Last Commented](#)

**Modern House**

 Tempor nostrud explicari est eu, vim moderatius cotidieque ad, id novum everti qui. Per an errem iniure appareat, pri harum volumus definiebas eu, has ut munere vituperata. His diam consetetur et, ut sit quot utnam efficiantur, adhuc invenire assentior mei eu.

Bedrooms: 5+  
Price: \$215,000

**Villa - A beautiful house and the yard**

 Lorem ipsum fierent mnesarchum ne vel, et usu posse takimata omittantur, pro ut tale erant sapientem. Et regione tibi que ancillae nam. Tale modus iuaret eu usu. Id labor semper voluptatum vel. Eu labore suscipit nam, in simul qualisque pri.

Bedrooms: 4  
Price: \$115,000

### find

Property Type:

Bedrooms:

Price:

# Support is Available

SiteManager's interface is designed to be as straightforward and self-explanatory as possible, enabling your users to work with confidence. However, if you have any questions or problems, support is available via our easy & fast helpdesk system at: [www.jmasitemanager.com/support.aspx](http://www.jmasitemanager.com/support.aspx) .