iDANGERO.US PREMIUM SITE TEMPLATES SCRIPTS GRAPHICS

Photo Exhibition
Home Gallery About Contacts PhEx.Social
Previous Photo
Next Photo
Rainy
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ut diam lectus, non adipiscing orci. Maecenas dapibus tellus vitae turpis vulputate ut aliquet nisl sodales. Fusce nec mauris sit amet odio placerat condimentum id at nunc
Template

“Photo Exhibition”

Premium Template By iDangero.us

Documentation

Thank you for purchasing this item. If you have any questions that are beyond the scope of this help file, please feel free to contact us via support ticket form here.

Table of Contents

1. About “Photo Exhibition”

Photo Exhibition is an Ajax based site template, designed by the iDangero.us team.

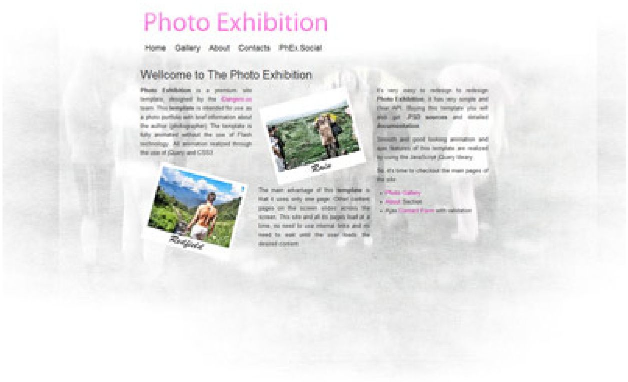
This template is intended for use as a photo portfolio with brief information about the author (photographer). The template is fully animated without the use of Flash technology. Smooth and good looking animation and Ajax features of this template are realized by using the JavaScript jQuery library.

The main advantage of this template is that it uses only one page. Other content pages on the screen slides across the screen. This site and all pages load at a time, no need to use internal links and no need to wait until the user loads the desired content.

This template is easy to redesign, it has very simple and clear API.

Also **Photo Exhibition** is powered with an Ajax Contact Form with validation.

Here is a couple of screenshots:



Wellcome to The Photo Exhibition

Photo Exhibition is a premium site template, designed by the iDangero.us team. This **template** is intended for use as a photo portfolio with brief information about the author (photographer). The template is fully animated without the use of Flash technology. All animation realized through the use of jQuery and CSS3

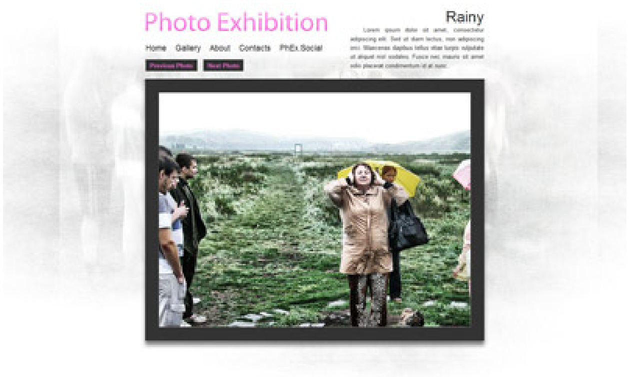
The main advantage of this **template** is that it uses only one page. Other content pages on the screen slides across the screen. This site and all its pages load at a time, no need to use internal links and no need to wait until the user loads the desired content.

It's very easy to redesign to redesign **Photo Exhibition**, it has very simple and clear API. Buying this template you will also get .PSD sources and detailed **documentation**.

Smooth and good looking animation and ajax features of this template are realized by using the JavaScript jQuery library.

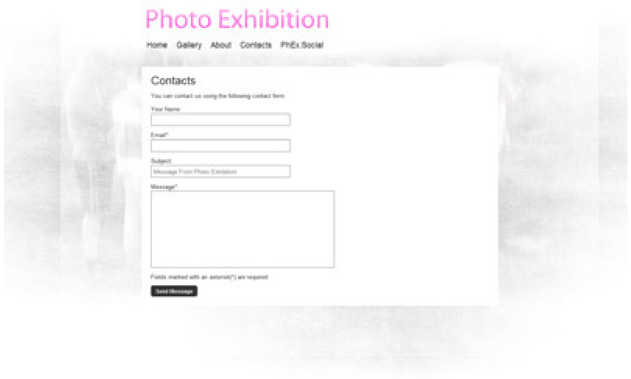
So, it's time to checkout the main pages of the site:

- Photo Gallery
- About Section
- Ajax Contact Form with validation



Rainy

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ut diam lectus, non adipiscing orci. Maecenas dapibus tellus vitae turpis vulputate ut aliquet nisl sodales. Fusce nec mauris sit amet odio placerat condimentum id at nunc.



Contacts

You can contact us using the following contact form:

Your Name:

Email*

Subject:

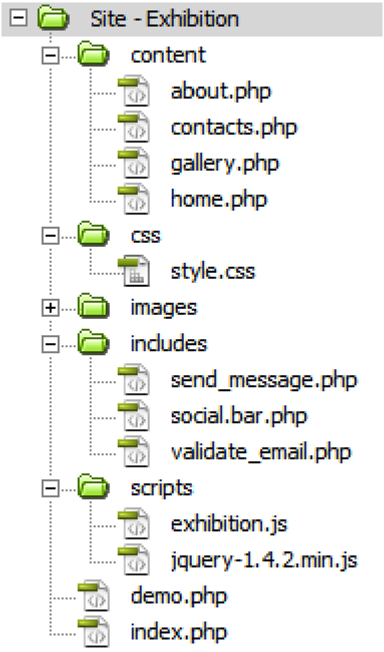
Message From Photo Exhibition

Message*

Fields marked with an asterisk(*) are required

2. Photo Exhibition Files Structure

Photo Exhibition files are located in the **exhibition/** folder from the downloaded archive:



```
Site - Exhibition
  content
    about.php
    contacts.php
    gallery.php
    home.php
  css
    style.css
  images
  includes
    send_message.php
    social.bar.php
    validate_email.php
  scripts
    exhibition.js
    jquery-1.4.2.min.js
  demo.php
  index.php
```

content: in this folder there are .php files with a html content pages.

css: folder with a .css files

images: here you can find layouts images used in this Template

includes: 2 php scripts used for Ajax contact form, and the social bar layout.

scripts: JavaScripts used in this template

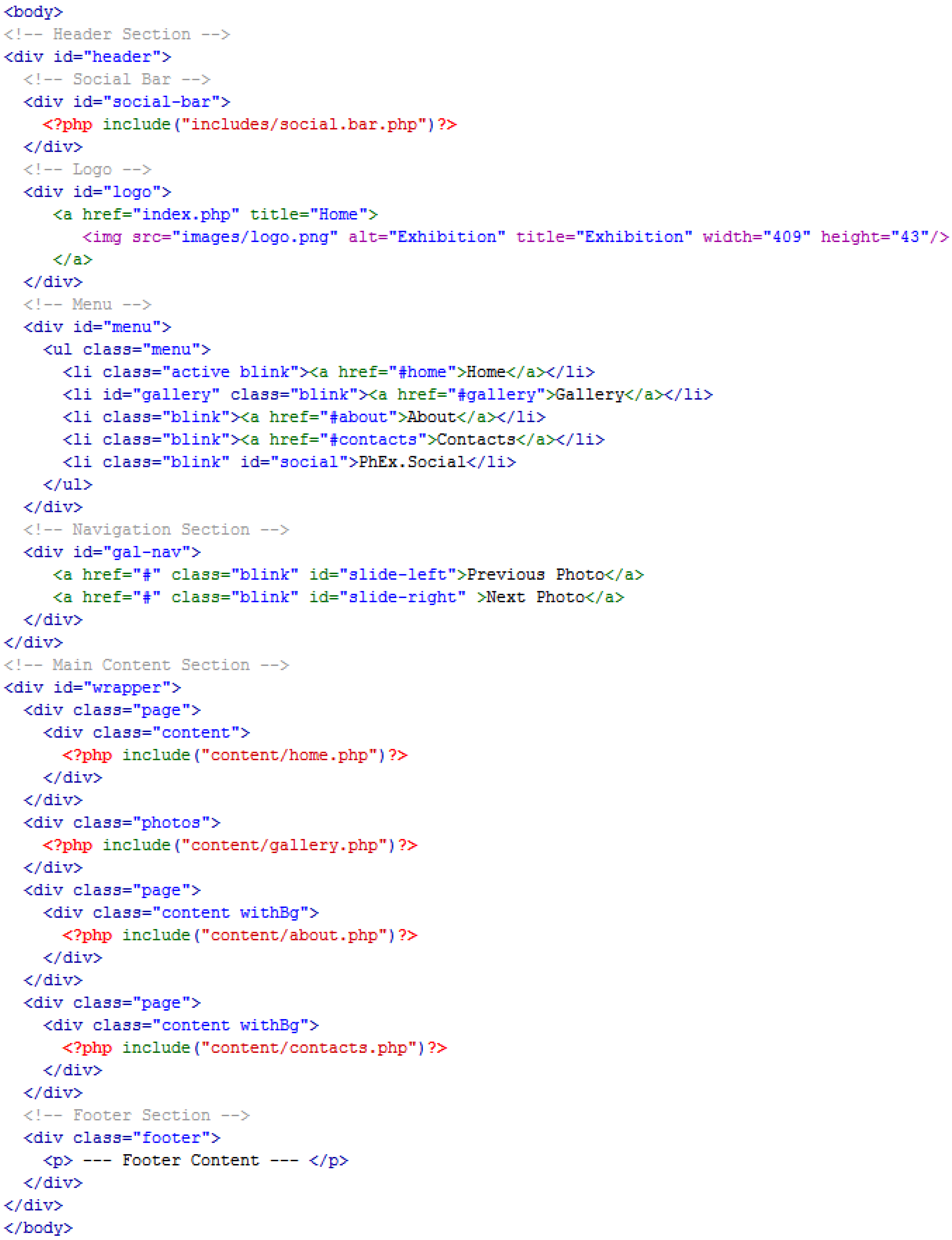
And the main **index.php** in the root folder

3. Installation

To install **Photo Exhibition** template you have to copy all files & folders from the **exhibition/** folder to your web-site root folder on the server (It could be a real server or local testing server).

4. HTML Structure

Let’s look at the main HTML/PHP structure of Photo Exhibition Template. Open index.php file with a text editor (Highly recommended to use Adobe® Dreamweaver® or Notepad ++ editors) **Index.php** <body> structure:



```
<body>
<!-- Header Section -->
<div id="header">
  <!-- Social Bar -->
  <div id="social-bar">
    <?php include("includes/social.bar.php")?>
  </div>
  <!-- Logo -->
  <div id="logo">
     <a href="index.php" title="Home">
        <img src="images/logo.png" alt="Exhibition" title="Exhibition" width="409" height="43"/>
     </a>
  </div>
  <!-- Menu -->
  <div id="menu">
    <ul class="menu">
      <li class="active blink"><a href="#home">Home</a></li>
      <li id="gallery" class="blink"><a href="#gallery">Gallery</a></li>
      <li class="blink"><a href="#about">About</a></li>
      <li class="blink"><a href="#contacts">Contacts</a></li>
      <li class="blink" id="social">PhEx.Social</li>
    </ul>
  </div>
  <!-- Navigation Section -->
  <div id="gal-nav">
     <a href="#" class="blink" id="slide-left">Previous Photo</a>
     <a href="#" class="blink" id="slide-right" >Next Photo</a>
  </div>
</div>
<!-- Main Content Section -->
<div id="wrapper">
  <div class="page">
    <div class="content">
      <?php include("content/home.php")?>
    </div>
  </div>
  <div class="photos">
    <?php include("content/gallery.php")?>
  </div>
  <div class="page">
    <div class="content withBg">
      <?php include("content/about.php")?>
    </div>
  </div>
  <div class="page">
    <div class="content withBg">
      <?php include("content/contacts.php")?>
    </div>
  </div>
  <!-- Footer Section -->
  <div class="footer">
    <p> --- Footer Content --- </p>
  </div>
</div>
</body>
```

Header Section

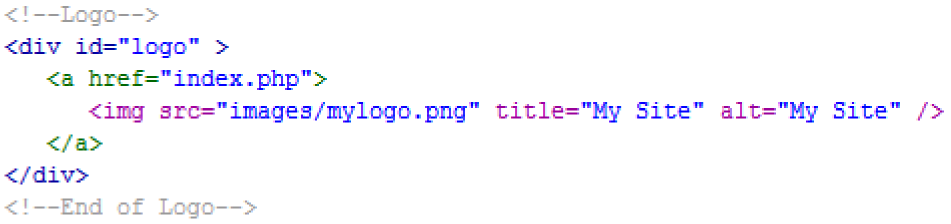
Header Section contains site Logo, Social.Bar, Menu and Navigation section with "Next Page" and "Previous Page" buttons for the Gallery page.

Social.Bar

Read more about Social.Bar in the chapter 9.

Logo Section

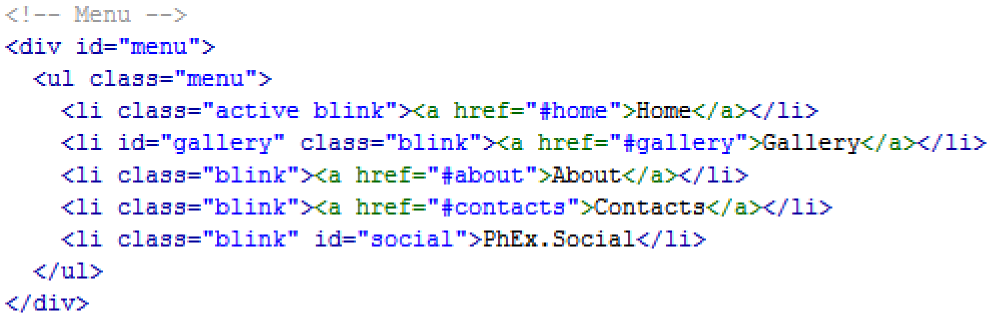
To change site **logo** you have to replace the logo.png (located in **images** folder) image to yours own or you could change the source of logo image to your image like this:



```
<!--Logo-->
<div id="logo" >
   <a href="index.php">
      <img src="images/mylogo.png" title="My Site" alt="My Site" />
   </a>
</div>
<!--End of Logo-->
```

Menu Section

Here is the HTML formatting for menu section:



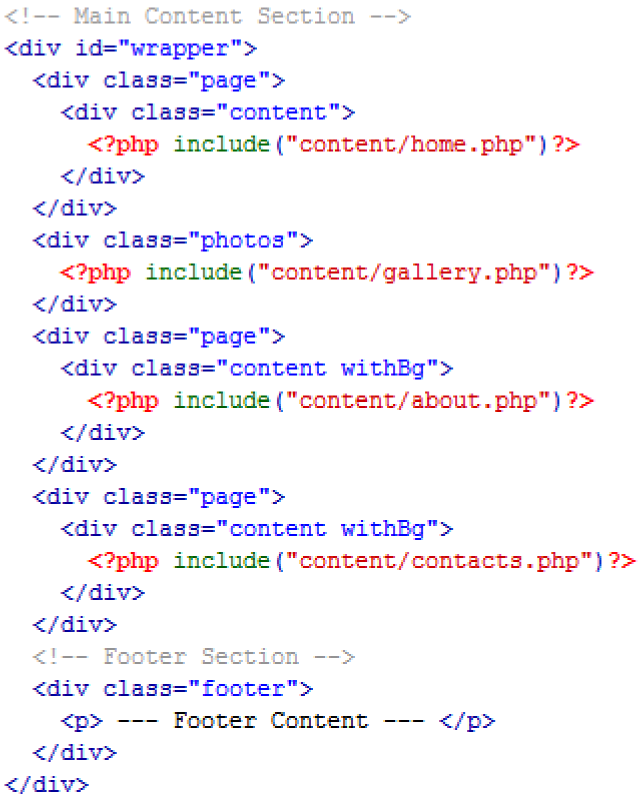
```
<!-- Menu -->
<div id="menu">
  <ul class="menu">
    <li class="active blink"><a href="#home">Home</a></li>
    <li id="gallery" class="blink"><a href="#gallery">Gallery</a></li>
    <li class="blink"><a href="#about">About</a></li>
    <li class="blink"><a href="#contacts">Contacts</a></li>
    <li class="blink" id="social">PhEx.Social</li>
  </ul>
</div>
```

To create your own menu link, first of all you have to create list element in the list (between the ***<ul class="menu"> and </ul>*** tags) using the ***<li>*** tag. Then put the link inside it with a simple ***<a>*** tag with any value of the ***href*** attribute. How it works? As you'll see below the main content section contains div blocks with "page" or "photos" class. Each this block (page) has its own index number. Each menu link (list item ***<li>)*** has its index number too. These index numbers start from 0. So when you'll click, the menu link with, for example, index number 1 (second link), then the screen will slide to page with the same index number (1).

Also don't forget to add "blink" class for link to add nice fade effect on hover action.

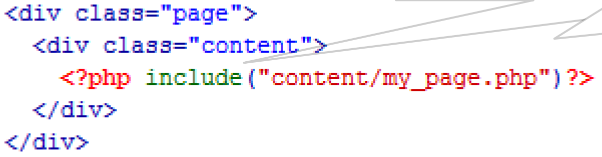
Main Content Section

This section contains all content pages of the site, footer and photo gallery page. Here is HTML formatting for this section:



```
<!-- Main Content Section -->
<div id="wrapper">
  <div class="page">
    <div class="content">
      <?php include("content/home.php")?>
    </div>
  </div>
  <div class="photos">
    <?php include("content/gallery.php")?>
  </div>
  <div class="page">
    <div class="content withBg">
      <?php include("content/about.php")?>
    </div>
  </div>
  <div class="page">
    <div class="content withBg">
      <?php include("content/contacts.php")?>
    </div>
  </div>
  <!-- Footer Section -->
  <div class="footer">
    <p> --- Footer Content --- </p>
  </div>
</div>
```

To create your own page, you have to use the following formatting inside the div block with a “wrapper” id attribute:



```
<div class="page">
  <div class="content">
    <?php include("content/my_page.php")?>
  </div>
</div>
```

This is the PHP function used to include file with a HTML or PHP content

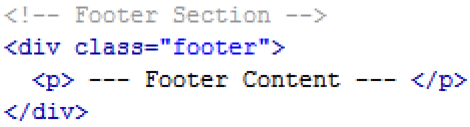
Here you have to specify a relative path to the file with a HTML content of the page. It could be HTML or PHP file

Index number of the created page has great importance. It must be the same as the index number of the menu link you’ll wish to assign with this page.

You can also add class ***whiteBg*** to the div block with a ***“content”*** class to add white background color for this page.

Footer Section

Footer section is used for copyrights information of you company or project



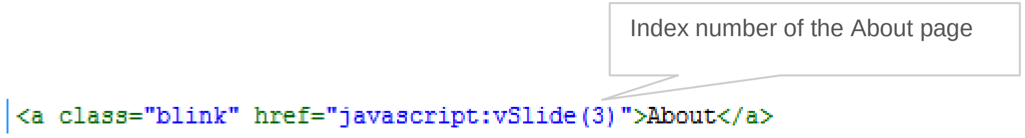
```
<!-- Footer Section -->
<div class="footer">
  <p> --- Footer Content --- </p>
</div>
```

Change **`--- Footer Content ---`** to something like this: **© 2010 My Company. All Rights Reserved**

5. Inner Linking (How it works?)

To slide pages from the simple link from the text page there is a JavaScript function vSlide($index), which is accepts as the argument the index number of the page you wish to slide to.

Example. We need to slide to the About page from somewhere. Of course we know the index number of this page. For example it is equal to 3 (forth page). We should use the following formatting for the link:



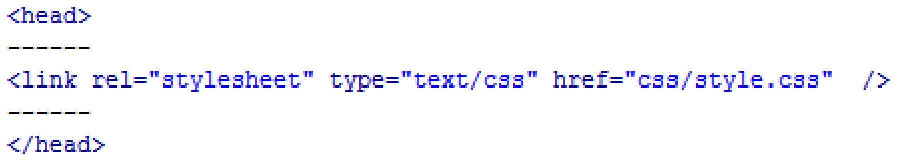
Index number of the About page

```
<a class="blink" href="javascript:vSlide(3)">About</a>
```

As you can see it's very easy to use inner linking in this template.

6. CSS Files and Structure

There is one CSS file used in this Template. You can find css attachment in the HEAD section of index.php file.



```
<head>
------
<link rel="stylesheet" type="text/css" href="css/style.css"  />
------
</head>
```

By default it is located in the ***css/*** folder. You can easily change the path or filename in the ***href*** attribute of ***link*** tag.

`style.css` file is contains all general layouts for the template. It is separated into sections:

`/* --- Common Rules --- */`

There are common rules in this section, such as font size, fonts, link colors, layout images etc.

`/* --- Header & Logo --- */`

This section contains style rules for header and logo sections

`/* --- Menu & Navigation --- */`

Rules for menu and navigation buttons

`/* --- Content Pages --- */`

All styling rules for content pages

`/* --- Photo Gallery --- */`

Rules for Photo Gallery and all its elements

`/* --- Social.Bar --- */`

Styling for the Social.Bar

`/* --- Footer --- */`

Form styling

`/* --- Footer --- */`

All rules for the Products Section

`/*---- Ajax Loader-----*/`

Ajax Loader styling

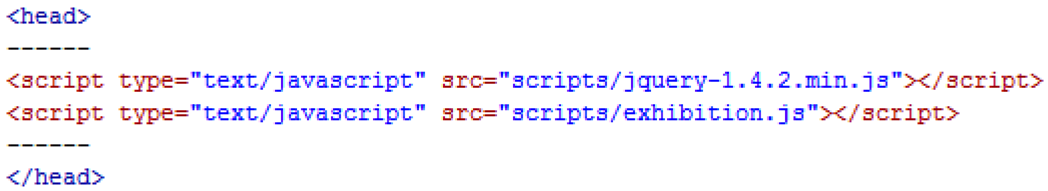
`/*----End of CSS File----*/`

If you would like to edit a specific section of the site, simply find the appropriate section in the CSS file, and then scroll down until you find the appropriate style that needs to be edited.

7. JavaScripts

Photo Exhibition Template uses two JavaScript files

Attachment of JavaScript files can be found in the HEAD section of index.php file:



```
<head>
------
<script type="text/javascript" src="scripts/jquery-1.4.2.min.js"></script>
<script type="text/javascript" src="scripts/exhibition.js"></script>
------
</head>
```

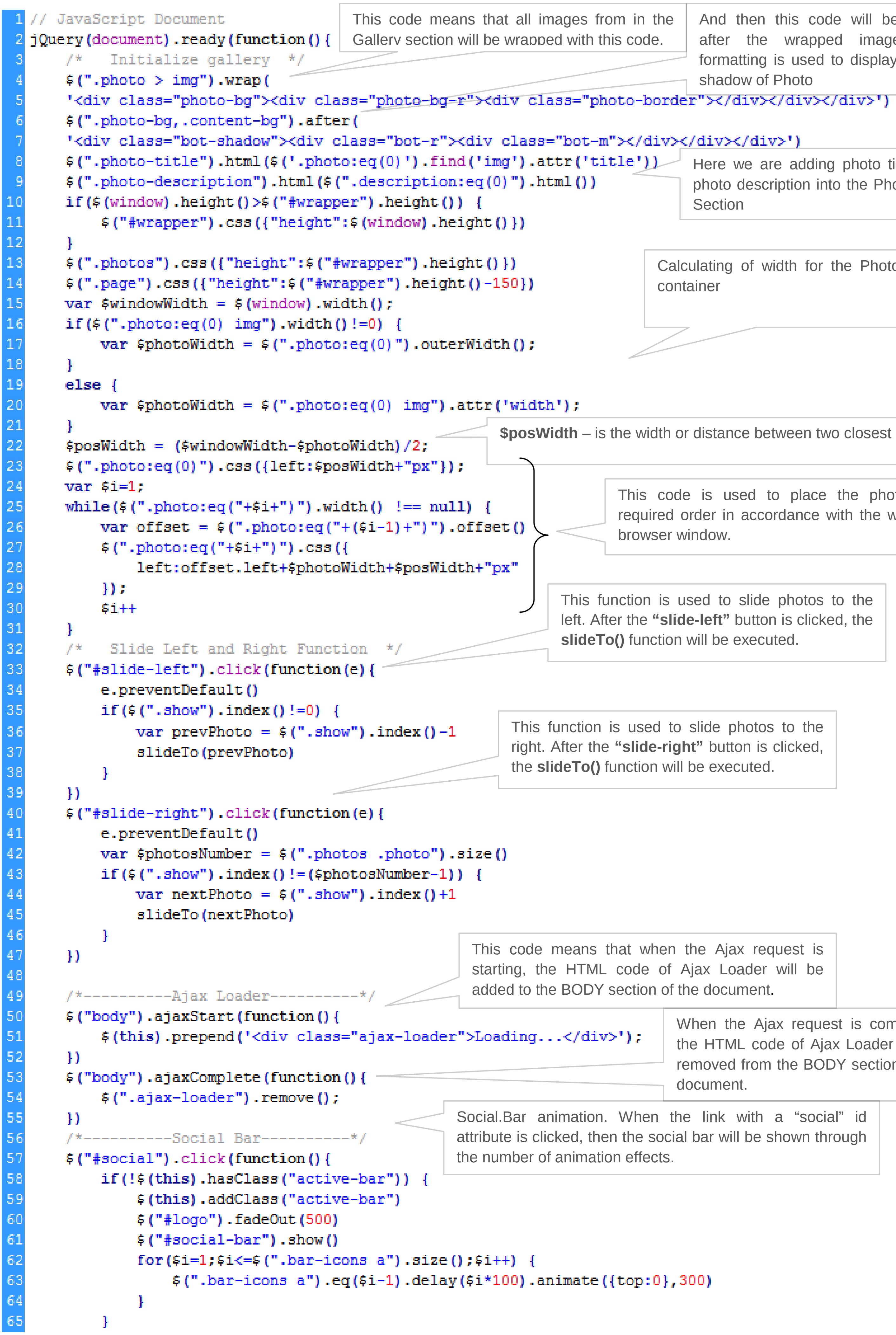
You can easily change the path or filename in the **`src`** attribute of **`script`** tag.

The first script file is the jQuery (JavaScript library that greatly reduces the amount of code that you must write).

And the second script file (exhibition.js) is the custom script used for animation, effects and Ajax requests. You can learn more about jQuery methods used here (**`fadeOut(), fadeIn(), fadeTo(), animate(), jQuery.get(), html(), delay(), hover() etc.`**) by visiting this [jQuery Documentation]() Site.

First of all let’s look at the custom script more closely.

The first part of the script is executed when the document is loaded. Here are the effects and rules for handling menu links, its animation, Ajax loader, Gallery initialization and the social bar animation.



```
// JavaScript Document
jQuery(document).ready(function(){
    /*  Initialize gallery  */
    $(".photo > img").wrap(
    '<div class="photo-bg"><div class="photo-bg-r"><div class="photo-border"></div></div></div>')
    $(".photo-bg,.content-bg").after(
    '<div class="bot-shadow"><div class="bot-r"><div class="bot-m"></div></div></div>')
    $(".photo-title").html($('.photo:eq(0)').find('img').attr('title'))
    $(".photo-description").html($(".description:eq(0)").html())
    if($(window).height()>$("#wrapper").height()) {
        $("#wrapper").css({"height":$(window).height()})
    }
    $(".photos").css({"height":$("#wrapper").height()})
    $(".page").css({"height":$("#wrapper").height()-150})
    var $windowWidth = $(window).width();
    if($(".photo:eq(0) img").width()!=0) {
        var $photoWidth = $(".photo:eq(0)").outerWidth();
    }
    else {
        var $photoWidth = $(".photo:eq(0) img").attr('width');
    }
    $posWidth = ($windowWidth-$photoWidth)/2;
    $(".photo:eq(0)").css({left:$posWidth+"px"});
    var $i=1;
    while($(".photo:eq("+$i+")").width() !== null) {
        var offset = $(".photo:eq("+($i-1)+")").offset()
        $(".photo:eq("+$i+")").css({
            left:offset.left+$photoWidth+$posWidth+"px"
        });
        $i++
    }
    /*  Slide Left and Right Function  */
    $("#slide-left").click(function(e){
        e.preventDefault()
        if($(".show").index()!=0) {
            var prevPhoto = $(".show").index()-1
            slideTo(prevPhoto)
        }
    })
    $("#slide-right").click(function(e){
        e.preventDefault()
        var $photosNumber = $(".photos .photo").size()
        if($(".show").index()!=($photosNumber-1)) {
            var nextPhoto = $(".show").index()+1
            slideTo(nextPhoto)
        }
    })

    /*----------Ajax Loader----------*/
    $("body").ajaxStart(function(){
        $(this).prepend('<div class="ajax-loader">Loading...</div>');
    })
    $("body").ajaxComplete(function(){
        $(".ajax-loader").remove();
    })
    /*----------Social Bar----------*/
    $("#social").click(function(){
        if(!$(this).hasClass("active-bar")) {
            $(this).addClass("active-bar")
            $("#logo").fadeOut(500)
            $("#social-bar").show()
            for($i=1;$i<=$(".bar-icons a").size();$i++) {
                $(".bar-icons a").eq($i-1).delay($i*100).animate({top:0},300)
            }
        }
```

This code means that all images from in the Gallery section will be wrapped with this code.

And then this code will be added after the wrapped image. This formatting is used to display bottom shadow of Photo

Here we are adding photo title and photo description into the Photo Info Section

Calculating of width for the Photo container

$posWidth – is the width or distance between two closest photos

This code is used to place the photos in the required order in accordance with the width of the browser window.

This function is used to slide photos to the left. After the **“slide-left”** button is clicked, the **slideTo()** function will be executed.

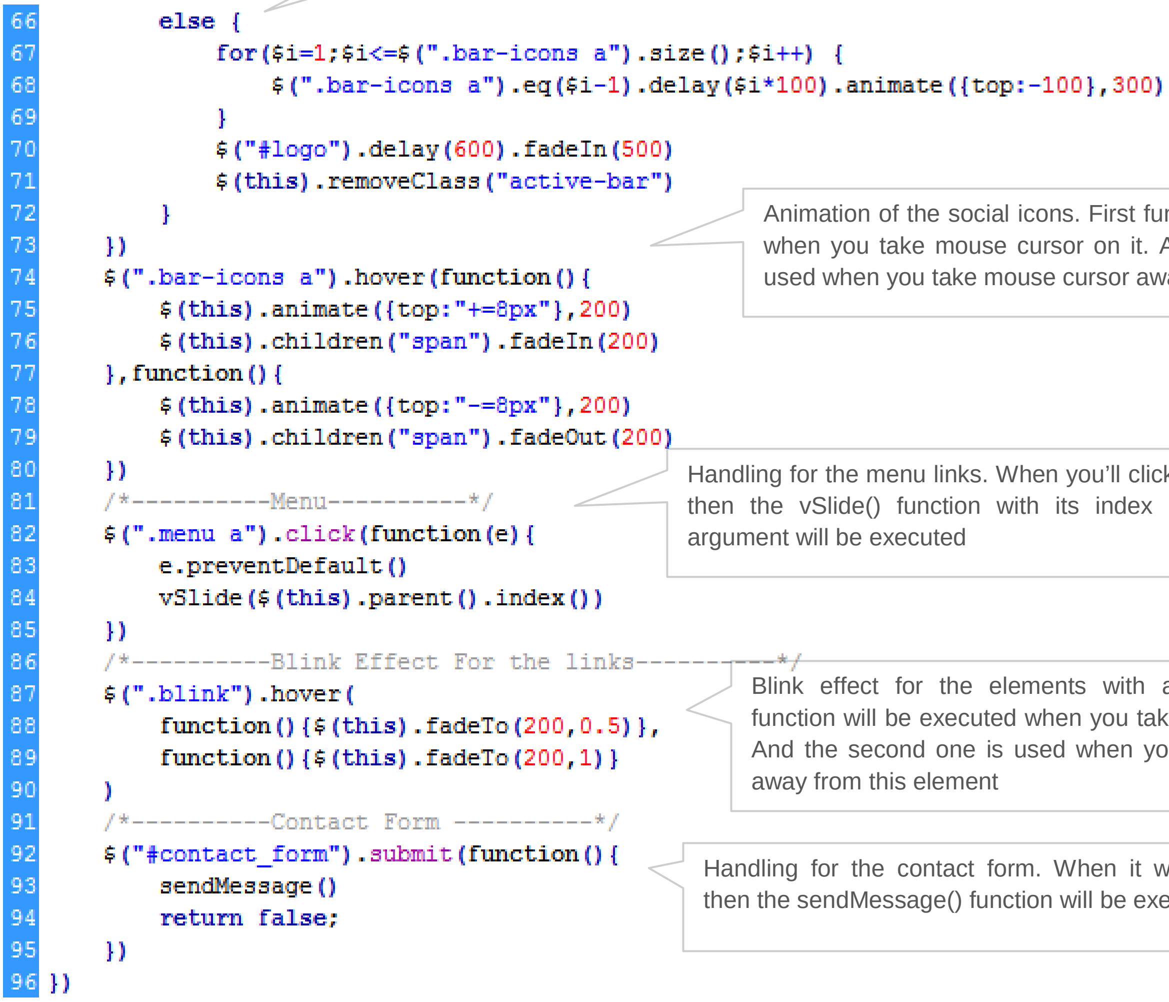
This function is used to slide photos to the right. After the **“slide-right”** button is clicked, the **slideTo()** function will be executed.

This code means that when the Ajax request is starting, the HTML code of Ajax Loader will be added to the BODY section of the document.

When the Ajax request is completed, the HTML code of Ajax Loader will be removed from the BODY section of the document.

Social.Bar animation. When the link with a “social” id attribute is clicked, then the social bar will be shown through the number of animation effects.

This part is used to hide social bar through the number of animation effects



```
        else {
            for($i=1;$i<=$(".bar-icons a").size();$i++) {
                $(".bar-icons a").eq($i-1).delay($i*100).animate({top:-100},300)
            }
            $("#logo").delay(600).fadeIn(500)
            $(this).removeClass("active-bar")
        }
    })
    $(".bar-icons a").hover(function(){
        $(this).animate({top:"+=8px"},200)
        $(this).children("span").fadeIn(200)
    },function(){
        $(this).animate({top:"-=8px"},200)
        $(this).children("span").fadeOut(200)
    })
    /*----------Menu----------*/
    $(".menu a").click(function(e){
        e.preventDefault()
        vSlide($(this).parent().index())
    })
    /*----------Blink Effect For the links----------*/
    $(".blink").hover(
        function(){$(this).fadeTo(200,0.5)},
        function(){$(this).fadeTo(200,1)}
    )
    /*----------Contact Form ----------*/
    $("#contact_form").submit(function(){
        sendMessage()
        return false;
    })
})
```

Animation of the social icons. First function will be executed when you take mouse cursor on it. And the second one is used when you take mouse cursor away from icon .

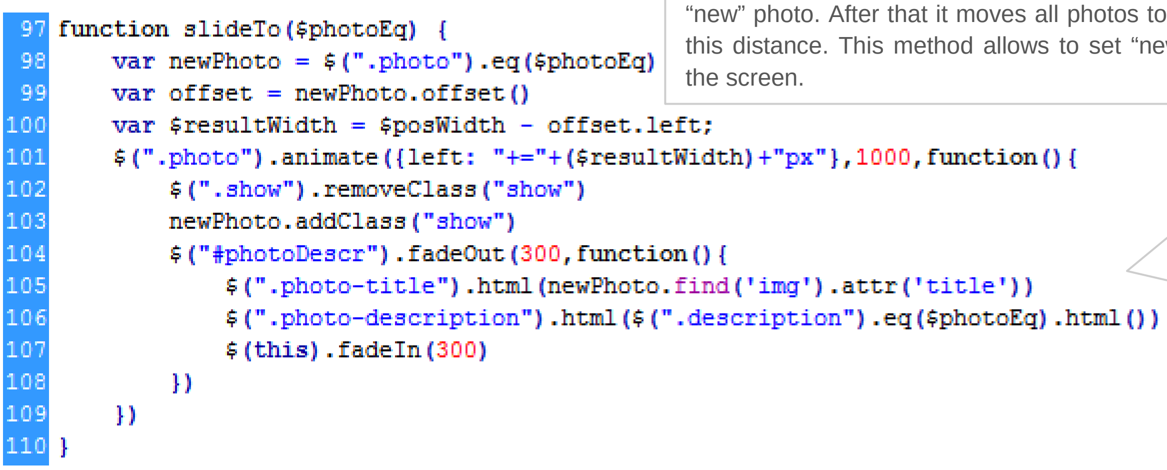
Handling for the menu links. When you'll click the menu link, then the vSlide() function with its index number as an argument will be executed

Blink effect for the elements with a "blink" class. First function will be executed when you take mouse cursor on it. And the second one is used when you take mouse cursor away from this element

Handling for the contact form. When it will be submitted, then the sendMessage() function will be executed

The next part of code contains functions which could be called from any place of the document. These functions are used to slide pages and photos across the screen, for submitting and validation of the contact form.

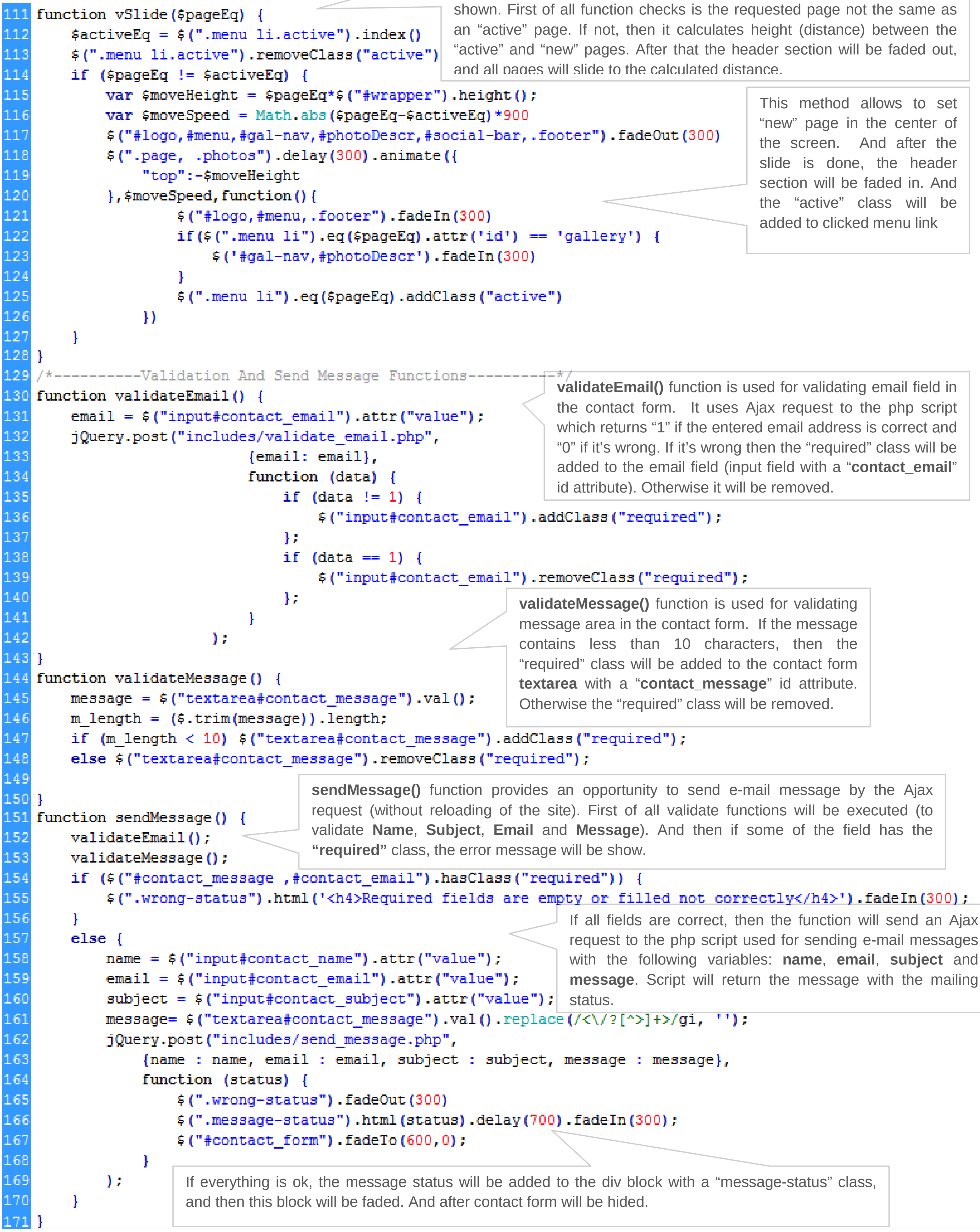
slieTo() function is used to slide photos. As the argument this function accepts an index number of photo, which is must be shown. First of all this function calculates the distance between "active" photo and "new" photo. After that it moves all photos to the left or to the right on this distance. This method allows to set "new" photo in the center of the screen.



```
function slideTo($photoEq) {
    var newPhoto = $(".photo").eq($photoEq)
    var offset = newPhoto.offset()
    var $resultWidth = $posWidth - offset.left;
    $(".photo").animate({left: "+="+($resultWidth)+"px"},1000,function(){
        $(".show").removeClass("show")
        newPhoto.addClass("show")
        $("#photoDescr").fadeOut(300,function(){
            $(".photo-title").html(newPhoto.find('img').attr('title'))
            $(".photo-description").html($(".description").eq($photoEq).html())
            $(this).fadeIn(300)
        })
    })
}
```

And after the photos are moved, the photo title and photo description will be added into the Photo Info Section.

Code continuous on the next page



```
function vSlide($pageEq) {
    $activeEq = $(".menu li.active").index()
    $(".menu li.active").removeClass("active")
    if ($pageEq != $activeEq) {
        var $moveHeight = $pageEq*$("#wrapper").height();
        var $moveSpeed = Math.abs($pageEq-$activeEq)*900
        $("#logo,#menu,#gal-nav,#photoDescr,#social-bar,.footer").fadeOut(300)
        $(".page, .photos").delay(300).animate({
            "top":-$moveHeight
        },$moveSpeed,function(){
                $("#logo,#menu,.footer").fadeIn(300)
                if($(".menu li").eq($pageEq).attr('id') == 'gallery') {
                    $('#gal-nav,#photoDescr').fadeIn(300)
                }
                $(".menu li").eq($pageEq).addClass("active")
            })
    }
}
/*----------Validation And Send Message Functions---------*/
function validateEmail() {
    email = $("input#contact_email").attr("value");
    jQuery.post("includes/validate_email.php",
                        {email: email},
                        function (data) {
                            if (data != 1) {
                                $("input#contact_email").addClass("required");
                            };
                            if (data == 1) {
                                $("input#contact_email").removeClass("required");
                            };
                        }
                    );
}
function validateMessage() {
    message = $("textarea#contact_message").val();
    m_length = ($.trim(message)).length;
    if (m_length < 10) $("textarea#contact_message").addClass("required");
    else $("textarea#contact_message").removeClass("required");

}
function sendMessage() {
    validateEmail();
    validateMessage();
    if ($("#contact_message ,#contact_email").hasClass("required")) {
        $(".wrong-status").html('<h4>Required fields are empty or filled not correctly</h4>').fadeIn(300);
    }
    else {
        name = $("input#contact_name").attr("value");
        email = $("input#contact_email").attr("value");
        subject = $("input#contact_subject").attr("value");
        message= $("textarea#contact_message").val().replace(/<\/?[^>]+>/gi, '');
        jQuery.post("includes/send_message.php",
            {name : name, email : email, subject : subject, message : message},
            function (status) {
                $(".wrong-status").fadeOut(300)
                $(".message-status").html(status).delay(700).fadeIn(300);
                $("#contact_form").fadeTo(600,0);
            }
        );
    }
}
```

vSLide() function provides an opportunity to slide pages vertically. As the argument function accepts the index number of page, which is has to be shown. First of all function checks is the requested page not the same as an "active" page. If not, then it calculates height (distance) between the "active" and "new" pages. After that the header section will be faded out, and all pages will slide to the calculated distance.

This method allows to set "new" page in the center of the screen. And after the slide is done, the header section will be faded in. And the "active" class will be added to clicked menu link

validateEmail() function is used for validating email field in the contact form. It uses Ajax request to the php script which returns "1" if the entered email address is correct and "0" if it's wrong. If it's wrong then the "required" class will be added to the email field (input field with a "**contact_email**" id attribute). Otherwise it will be removed.

validateMessage() function is used for validating message area in the contact form. If the message contains less than 10 characters, then the "required" class will be added to the contact form **textarea** with a "**contact_message**" id attribute. Otherwise the "required" class will be removed.

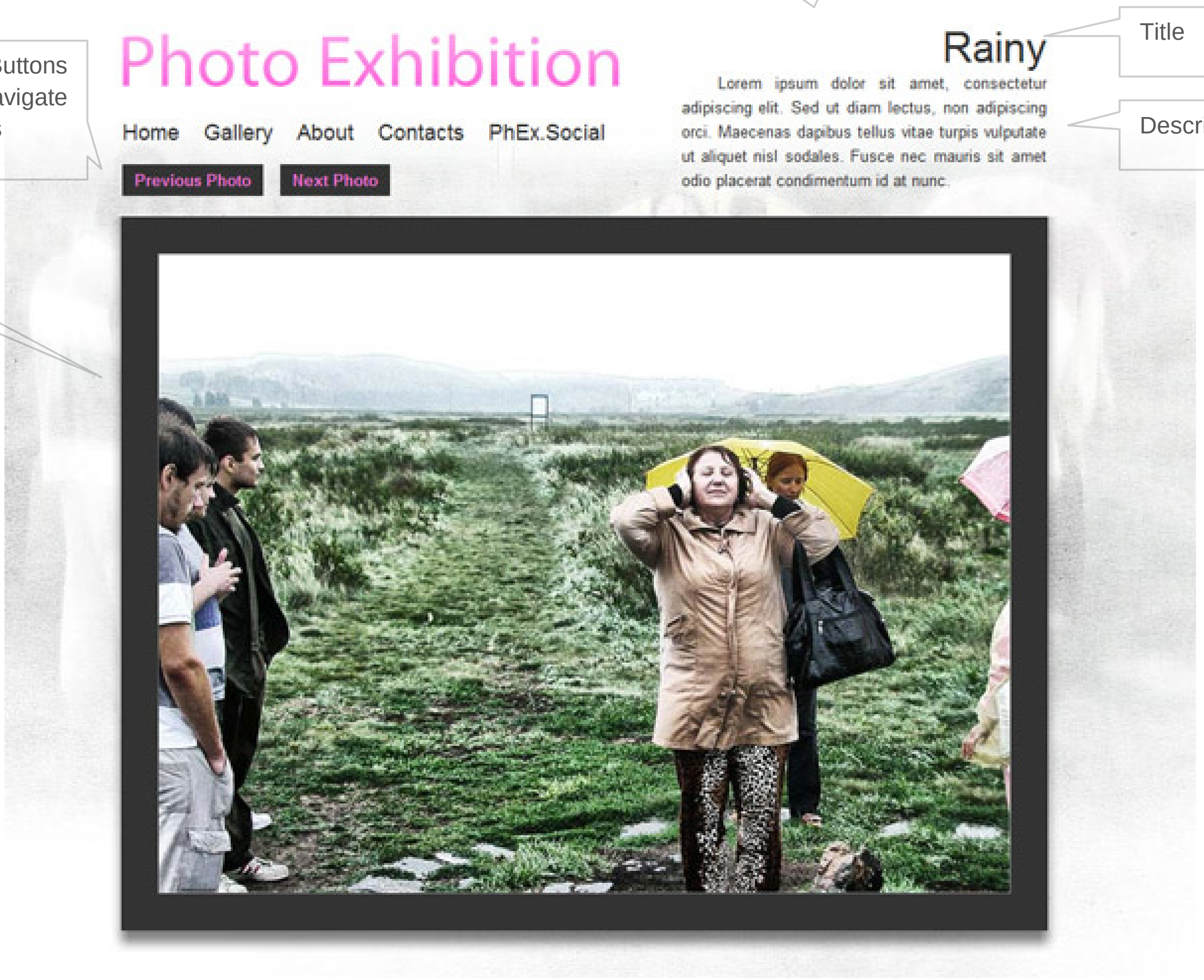
sendMessage() function provides an opportunity to send e-mail message by the Ajax request (without reloading of the site). First of all validate functions will be executed (to validate **Name**, **Subject**, **Email** and **Message**). And then if some of the field has the **"required"** class, the error message will be show.

If all fields are correct, then the function will send an Ajax request to the php script used for sending e-mail messages with the following variables: **name**, **email**, **subject** and **message**. Script will return the message with the mailing status.

If everything is ok, the message status will be added to the div block with a "message-status" class, and then this block will be faded. And after contact form will be hided.

8. Image (Photo) Gallery

Here is how the Gallery page looks like:



Rainy

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed ut diam lectus, non adipiscing orci. Maecenas dapibus tellus vitae turpis vulputate ut aliquet nisl sodales. Fusce nec mauris sit amet odio placerat condimentum id at nunc.

Photo Info Section contains photo title and photo description

Navigation Buttons are used to navigate between photos

Title

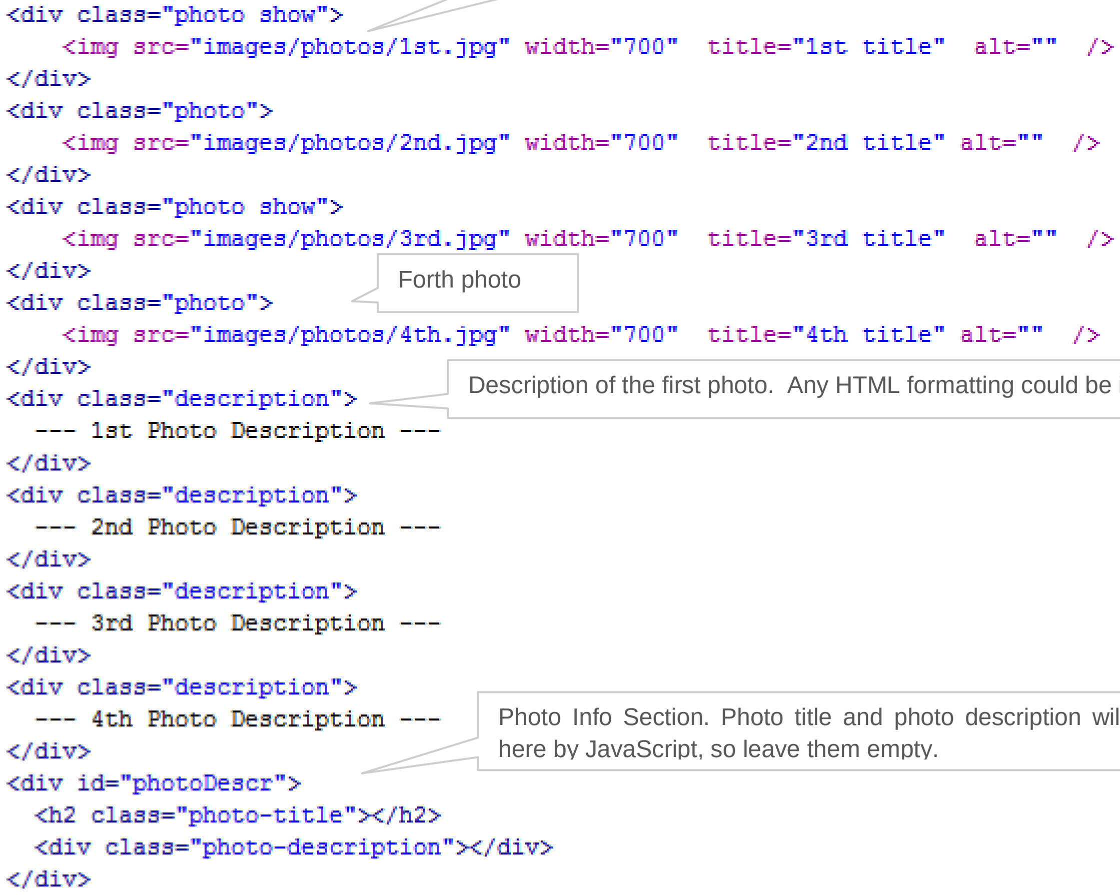
Description

Photo

Now let’s look at the HTML code of Photo Gallery and learn how to put your own photos into it. HTML content of the photo gallery located in the **content/gallery.php** file.

Div block for the first photo. Each div block used as a photo container must have a **“photo”** class. And the first div must have an additional **“show”** class.

First photo



```
<div class="photo show">
    <img src="images/photos/1st.jpg" width="700"  title="1st title"  alt=""  />
</div>
<div class="photo">
    <img src="images/photos/2nd.jpg" width="700"  title="2nd title" alt=""  />
</div>
<div class="photo show">
    <img src="images/photos/3rd.jpg" width="700"  title="3rd title"  alt=""  />
</div>
<div class="photo">
    <img src="images/photos/4th.jpg" width="700"  title="4th title" alt=""  />
</div>
<div class="description">
  --- 1st Photo Description ---
</div>
<div class="description">
  --- 2nd Photo Description ---
</div>
<div class="description">
  --- 3rd Photo Description ---
</div>
<div class="description">
  --- 4th Photo Description ---
</div>
<div id="photoDescr">
  <h2 class="photo-title"></h2>
  <div class="photo-description"></div>
</div>
```

Forth photo

Description of the first photo. Any HTML formatting could be inside

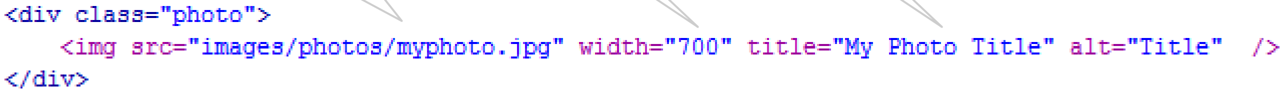
Photo Info Section. Photo title and photo description will be added here by JavaScript, so leave them empty.

As you can see a HTML formatting of the Photo Gallery is very simple. To add your own photo to the gallery, first of all, you have to create photo container (<div class=”photo”> … </div>). Then you have to put the image inside this container using the <img /> tag. Don’t forget to specify photo title and width with a “title” and “width” attributes:

Location of the image file

Photo width

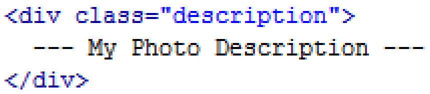
Photo title



```
<div class="photo">
    <img src="images/photos/myphoto.jpg" width="700" title="My Photo Title" alt="Title"  />
</div>
```

After the photo container is added, you’ll have to create description block for the photo using the following formatting:

You can use here any HTML formatting



```
<div class="description">
  --- My Photo Description ---
</div>
```

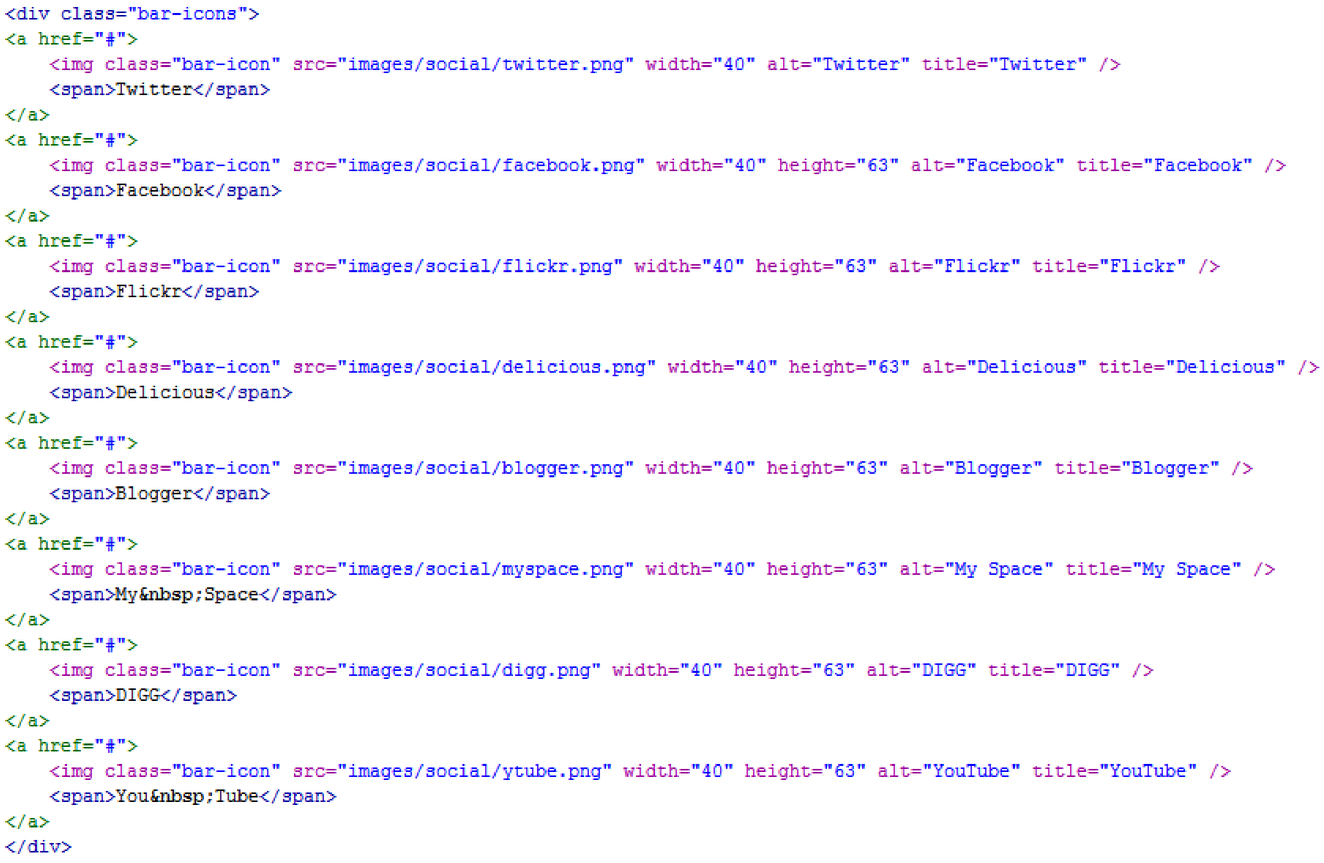
Couple of things you have to know. All photo containers must be before the description containers. Each description block will be applied to the photo container with the same index number. It’s mean – first description block for the first photo container, second description block for the second photo container etc.

9. Social.Bar

SOCIAL.BAR is a panel at the left top of the home page, where you can see the animated social icons which you can link to your relevant social pages (profiles).

digg
You
Tube

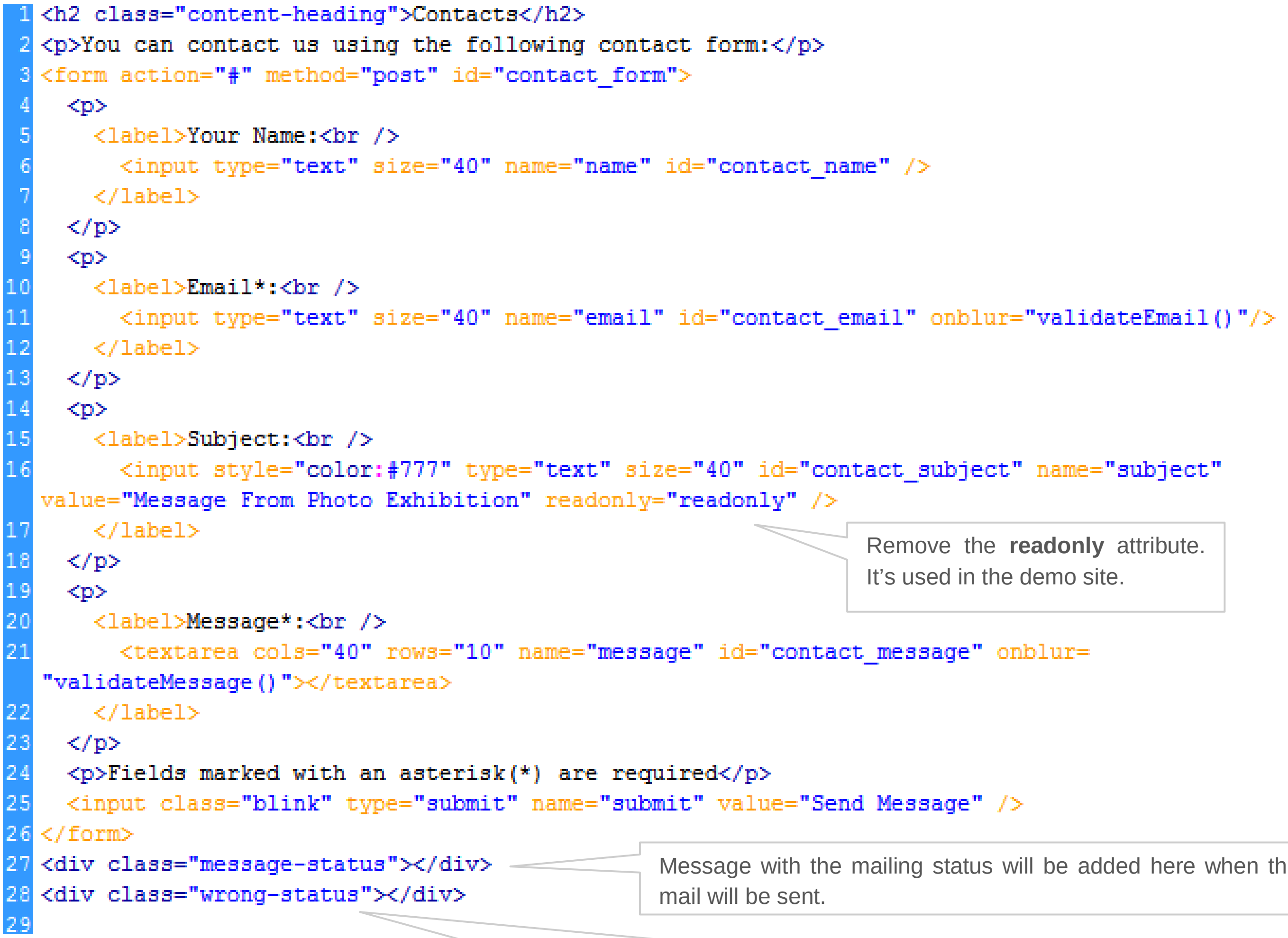
To edit Social.Bar just open the **includes/social.bar.php** file:



```
<div class="bar-icons">
<a href="#">
    <img class="bar-icon" src="images/social/twitter.png" width="40" alt="Twitter" title="Twitter" />
    <span>Twitter</span>
</a>
<a href="#">
    <img class="bar-icon" src="images/social/facebook.png" width="40" height="63" alt="Facebook" title="Facebook" />
    <span>Facebook</span>
</a>
<a href="#">
    <img class="bar-icon" src="images/social/flickr.png" width="40" height="63" alt="Flickr" title="Flickr" />
    <span>Flickr</span>
</a>
<a href="#">
    <img class="bar-icon" src="images/social/delicious.png" width="40" height="63" alt="Delicious" title="Delicious" />
    <span>Delicious</span>
</a>
<a href="#">
    <img class="bar-icon" src="images/social/blogger.png" width="40" height="63" alt="Blogger" title="Blogger" />
    <span>Blogger</span>
</a>
<a href="#">
    <img class="bar-icon" src="images/social/myspace.png" width="40" height="63" alt="My Space" title="My Space" />
    <span>My&nbsp;Space</span>
</a>
<a href="#">
    <img class="bar-icon" src="images/social/digg.png" width="40" height="63" alt="DIGG" title="DIGG" />
    <span>DIGG</span>
</a>
<a href="#">
    <img class="bar-icon" src="images/social/ytube.png" width="40" height="63" alt="YouTube" title="YouTube" />
    <span>You&nbsp;Tube</span>
</a>
</div>
```

10. Ajax Contact Form

HTML code of the contact page is in the **content/contacts.php** file:



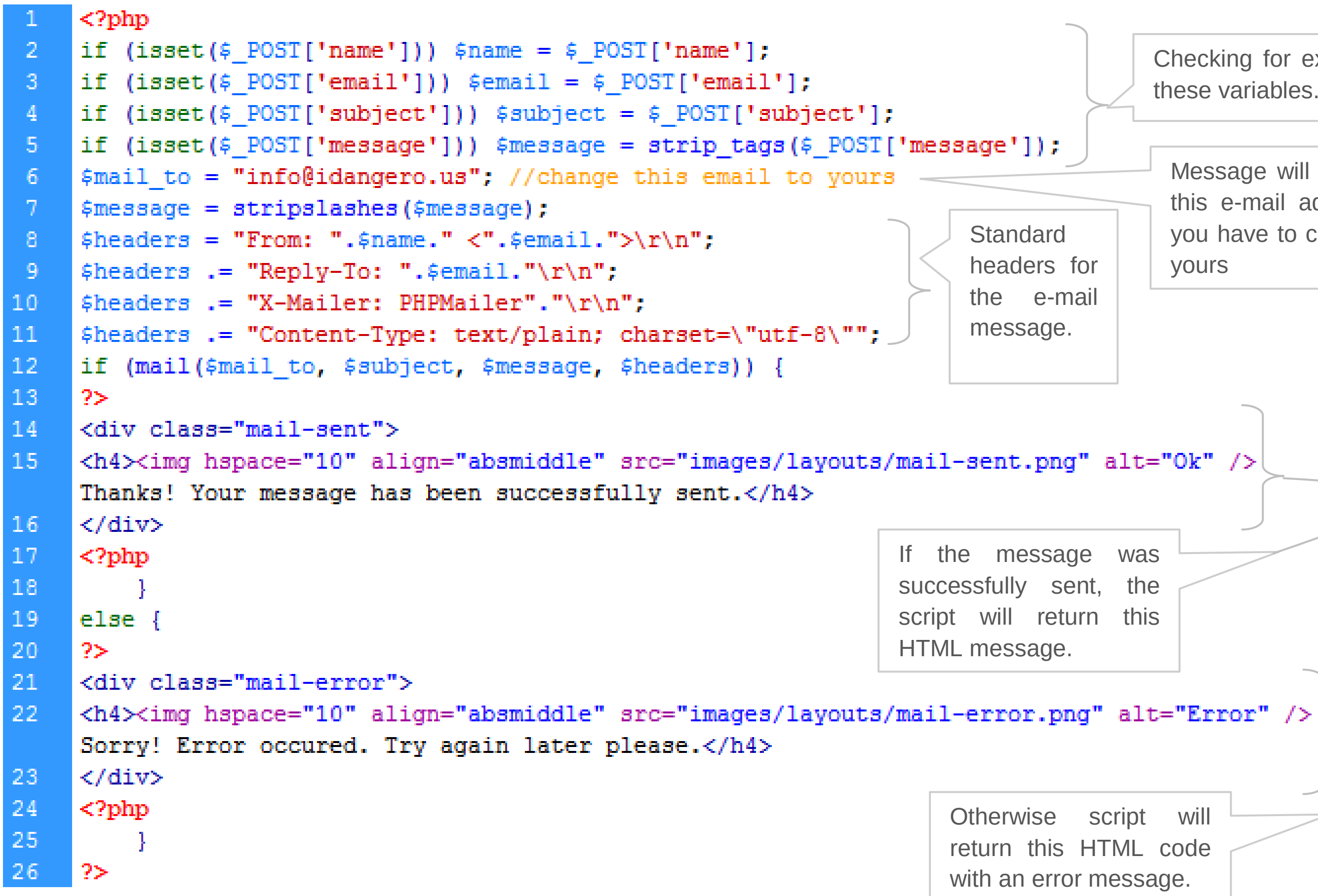
```
<h2 class="content-heading">Contacts</h2>
<p>You can contact us using the following contact form:</p>
<form action="#" method="post" id="contact_form">
  <p>
    <label>Your Name:<br />
      <input type="text" size="40" name="name" id="contact_name" />
    </label>
  </p>
  <p>
    <label>Email*:<br />
      <input type="text" size="40" name="email" id="contact_email" onblur="validateEmail()"/>
    </label>
  </p>
  <p>
    <label>Subject:<br />
      <input style="color:#777" type="text" size="40" id="contact_subject" name="subject"
value="Message From Photo Exhibition" readonly="readonly" />
    </label>
  </p>
  <p>
    <label>Message*:<br />
      <textarea cols="40" rows="10" name="message" id="contact_message" onblur=
"validateMessage()"></textarea>
    </label>
  </p>
  <p>Fields marked with an asterisk(*) are required</p>
  <input class="blink" type="submit" name="submit" value="Send Message" />
</form>
<div class="message-status"></div>
<div class="wrong-status"></div>
```

Remove the **readonly** attribute. It's used in the demo site.

Message with the mailing status will be added here when the mail will be sent.

If some of the fields will be not correct, then the error message will be added here.

If the all fields are correct sendMessage() script will send and Ajax request to the php mailing script located in the **includes/send_message.php** file. This script will send a message and will return mailing status message. Let's look at this script more closely:



```php
<?php
if (isset($_POST['name'])) $name = $_POST['name'];
if (isset($_POST['email'])) $email = $_POST['email'];
if (isset($_POST['subject'])) $subject = $_POST['subject'];
if (isset($_POST['message'])) $message = strip_tags($_POST['message']);
$mail_to = "info@idangero.us"; //change this email to yours
$message = stripslashes($message);
$headers = "From: ".$name." <".$email.">\r\n";
$headers .= "Reply-To: ".$email."\r\n";
$headers .= "X-Mailer: PHPMailer"."\r\n";
$headers .= "Content-Type: text/plain; charset=\"utf-8\"";
if (mail($mail_to, $subject, $message, $headers)) {
?>
<div class="mail-sent">
<h4><img hspace="10" align="absmiddle" src="images/layouts/mail-sent.png" alt="Ok" />
Thanks! Your message has been successfully sent.</h4>
</div>
<?php
    }
else {
?>
<div class="mail-error">
<h4><img hspace="10" align="absmiddle" src="images/layouts/mail-error.png" alt="Error" />
Sorry! Error occured. Try again later please.</h4>
</div>
<?php
    }
?>
```

Checking for existence of these variables.

Message will be sent to this e-mail address. So you have to change it to yours

Standard headers for the e-mail message.

If the message was successfully sent, the script will return this HTML message.

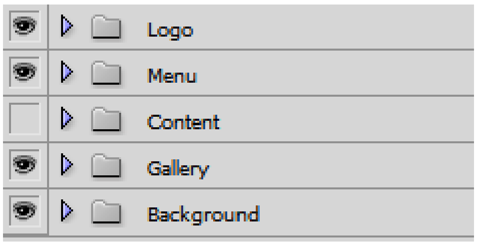
Otherwise script will return this HTML code with an error message.

You can easily edit this status messages using the any HTML formatting.

11. PSD Files & Structure

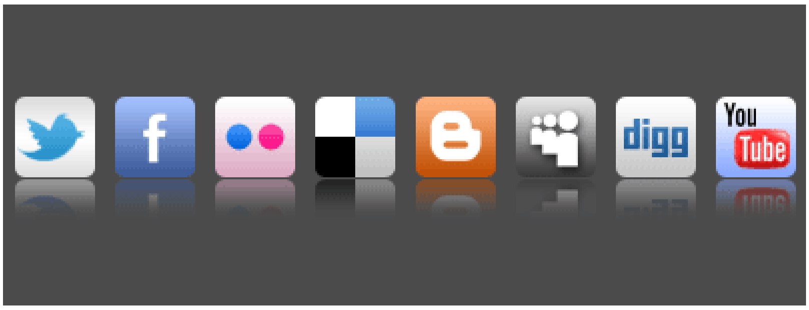
There are the two layered **.psd** files comes with the Photo Exhibition template. You can locate them into the **psd/** folder form the downloaded archive. Using these files you can easily redesign template. To edit these files you'll need an Adobe™ Photoshop™ (Recommended is CS4 or CS5 version)

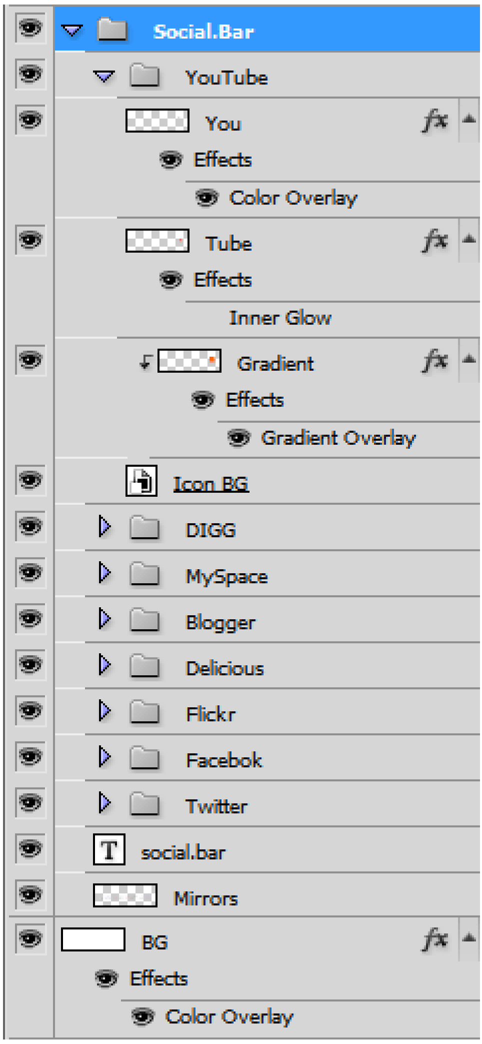
First .psd file is the **exhibition.psd**. It contains all general layouts for template styling:



Logo
Menu
Content
Gallery
Background

Second .psd file is the **social-bar.psd**. This file contains the Social.Bar layered icons





Social.Bar
YouTube
You
Effects
Color Overlay
Tube
Effects
Inner Glow
Gradient
Effects
Gradient Overlay
Icon BG
DIGG
MySpace
Blogger
Delicious
Flickr
Facebok
Twitter
social.bar
Mirrors
BG
Effects
Color Overlay

:

All layers in these file are well organized and easy to understand and redesign.

12. Licensing Terms

Regular License (RL)

RL gives you as a customer non-exclusive & non-transferable right to use the product you've bought, in this case is the **Photo Exhibition** (further "**Item**")

1. We do not limit the number of **Item**'s copies you are going to use. Using one **Item** you can create, for example, even 100 web-sites.

2. You can use the **Item** by itself or it's also possible to apply it in other project you work at.

3. You can use the **Item** for your own purpose as well as for your clients.

4. You can use the **Item** in commercial projects.

5. You can reproduce the **Item**:

- on a web-site or as a web-site

- as a part of software you create not for sale

- as a printed variant

- in digital(electronic) format (as a presentation or an e-book)

- in video products

6. All photos used on iDangero.us for demonstration photo gallery functioning are our private property (if only there is nothing said about contrary). It's prohibited to use these photos wherever.

7. You are not allowed to sell, resell, license or give the **Item** free (any way) without our written consent. Please, do not offer to do it to any person.

8. You also do not have the right to use the **Item** in a project for selling (for example, for other templates, scripts, graphics and so on).

9. It's prohibited to rework / redesign / reproduce the **Item** (i.e.to rename it or change graphics & so on) and after this to sell it as your own.

10. In spite of reselling limitation you could claim money for the **Item** from your client.

11. If the **Item** (the whole **Item** or its parts) is created with materials used by GNU General Public License (GPL) (or some other license) it means you should follow all the terms of the license using the **Item**.