

“Launcher”

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 “Launcher”
2. Launcher Files & Folders Structure
3. Installation
4. HTML Structure
5. Inner Linking (Popup Window)
6. CSS Files and Structure
7. JavaScripts
8. Send Email Script
9. PSD Files and Structure
10. Licensing Terms

1. About “Launcher”

Launcher is the premium Ajax based template. It is designed for temporary usage until the main site is under construction. It contains a countdown to the grand opening date, brief information about the main site (or company) and social links.

Also, there is an opportunity for visitors to leave their email address to be informed when the site will be opened.

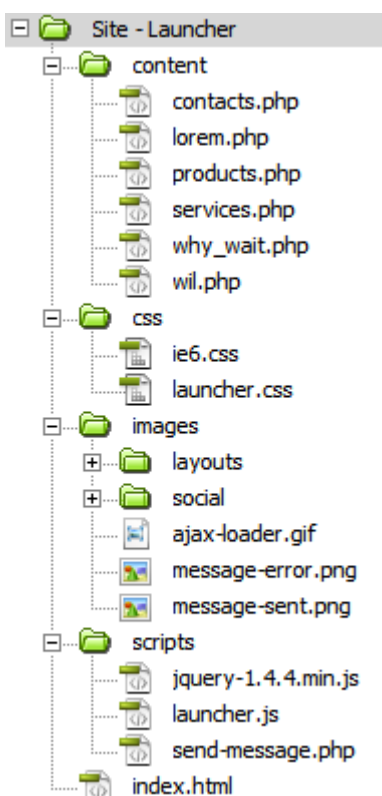
All content is loaded "on the fly" into popup window.

Smooth and good looking animation and Ajax features of this template are realized with the JavaScript jQuery library.

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

2. Launcher Files & Folders Structure

Launcher files are located in the **launcher/** folder from the downloaded archive:



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 and icons used in this Template

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

3. Installation

To install **Launcher** template you have to copy all files & folders from the **launcher/** 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 structure of the Launcher template. Open index.html file with a text editor (Highly recommended to use Adobe® Dreamweaver® or Notepad ++ editors).

Index.html structure:

```
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <link rel="stylesheet" type="text/css" href="css/launcher.css" />
6 <script type="text/javascript" src="scripts/jquery-1.4.4.min.js"></script>
7 <script type="text/javascript" src="scripts/launcher.js"></script>
8 <script type="text/javascript">
9   var isIE6=false;
10 </script>
11 <!--[if IE 6]>
12 <link rel="stylesheet" type="text/css" href="css/ie6.css" />
13 <script type="text/javascript">
14   var isIE6=true;
15 </script>
16 <![endif]>
17 <title>Launcher</title>
18 </head>
19 <body>
20 <div id="background"></div>
21 <div id="wrapper">
22   <div class="clock">
23     <h1 class="will-open">We will open after</h1>
24     <div class="clock-inner">
25       <div class="divider">:</div>
26       <div class="days">
27         <p>Days</p>
28         <div class="d-bg">
29           <div class="d-wrap">
30             <div class="day" style="top:0px;">
31             </div>
32             <div class="mask3"></div>
33           </div>
34         </div>

```

CSS and JavaScript files attachment

We have to define **isIE6** JavaScript variable equal to **false** by default.

CSS style sheet file used for IE6 browser only

And if the visitor' browser is IE6 then the **isIE6** variable will be equal to

Site Title

Section with an additional big clock background

Wrapper div contains whole site content

clock section contains the countdown clock

Blinking colon

Div block with a "**days**" class contains a number of days to the target date

Number of days will be placed here by JavaScript.

Mask layer is used to simulate obfuscation

```

35 <div class="hours">
36 <p>Hours</p>
37 <div class="h-bg">
38 <div class="h-wrap">
39 <div class="hour" style="top:0px;"></div>
40 </div>
41 <div class="mask2"></div>
42 </div>
43 </div>
44 <div class="minutes">
45 <p>Minutes</p>
46 <div class="m-bg">
47 <div class="m-wrap">
48 <div class="minute" style="top:0px;"></div>
49 </div>
50 <div class="mask2"></div>
51 </div>
52 </div>
53 <div class="seconds">
54 <p>Seconds</p>
55 <div class="s-bg">
56 <div class="s-wrap">
57 <div class="second1" style="top:0px;"></div>
58 <div class="second2" style="top:0px;"></div>
59 </div>
60 <div class="mask2"></div>
61 </div>
62 </div>
63 </div>
64 </div>
65 <div class="button-bg"> <span id="openContent" class="button-l"><span class="button-m">
<span style="padding:0px 60px" class="button">Won't wait?</span></span></span> </div>
66 <div class="content-wrap">
67 <div class="content-top">
68 <div class="content-bot">
69 <div class="content-mid">
70 <div class="left-col">
71 <h2 class="heading" style="color:#555">About Launcher</h2>
72 <ul class="ul-links">
73 <li><a class="popup" href="#wil">What is Launcher</a></li>
74 <li><a class="popup" href="#why_wait">Why you should wait</a></li>
75 <li><a class="popup" href="#products">Launcher featured products</a></li>
76 <li><a class="popup" href="#services">Launcher Services</a></li>
77 <li><a class="popup" href="#contacts">How to find us</a></li>
78 <li><a class="popup" href="#lorem">Lorem ipsum dolor</a></li>
79 </ul>
80 </div>
81 <div class="right-col">
82 <h2 class="heading">Be the first to know</h2>
83 <p>Give us your email address and we will inform you when our site will open:</p>
84 <div class="sendEmail">
85 <input value="Your email here" type="text" id="email" onkeyup=
"validateEmail()" size="20" />
86 <span class="button-l" id="sendEmail"><span class="button-m"><span style="
padding:0px 40px" class="button">Send</span></span></span>
87 <div class="message-status"></div>
88 </div>
89 <div class="clear"></div>

```

Div block with a "hours" class contains a number of hours to the target date

Number of hours will be placed here by JavaScript.

Mask layer is used to simulate obfuscation

Div block with a "minutes" class contains a number of minutes to the target date

Number of minutes will be placed here by JavaScript.

Mask layer is used to simulate obfuscation

Div block with a "seconds" class contains a number of seconds to the target date

First character of seconds' number will be placed into the div with a "second1" class and second character into the div with a "second2" class

Mask layer is used to simulate obfuscation

"Won't wait" button

"content-wrap" section contains main content which is hidden by default

Left content column with information links inside.

Right content column with "send email address" form and social links inside.

Send Email form with one input field and span button

JavaScript email validation will be executed on "onkeyup" event.

After the message with visitor' email address will be sent, the status message will be added to the div block with a "message-status" class.

Section with social icons. Just specify your social URL in the appropriate link in the HREF attribute. Link description located in the hidden by default **span** element.

```

90     <h2 class="heading">Launcher.Social</h2>
91     <div class="bar-icons">
92         <a href="#">  <span>Twitter</span> </a>
93         <a href="#">  <span>Facebook</span> </a>
94         <a href="#">  <span>Flickr</span> </a>
95         <a href="#">  <span>Delicious</span> </a>
96         <a href="#">  <span>Blogger</span> </a>
97         <a href="#">  <span>My &nbsp;Space</span> </a>
98         <a href="#">  <span>DIGG</span> </a>
99         <a href="#">  <span>You&nbsp;Tube</span> </a> </div>
100     </div>
101     <div class="clear"></div>
102 </div>
103 </div>
104 </div>
105 </div>
106 </div>
107 <div class="dark-layer"></div>
108 <div id="popup">
109     
110     <div class="popup-content"></div>
111 </div>
112 <div style="display:none;" class="dummy-block"></div>
113 </body>
114 </html>

```

Dark layer will fade in after the popup window is opened

Popup window container

Close image

Content of the popup window will be placed here

Here will be placed the same content as inside the popup window to calculate popup window size before it is opened. Dummy block is always hidden.

5. Inner Linking (Popup Window)

As the Launcher is one-page template, all additional content should be loaded into Popup Window. So let's see how the Inner Linking works. This you should to know first:

*All files with HTML content you want to put into Popup Window with Ajax request must be located in the **content/** folder.*

*All files must have a **.php** extension.*

Example. We want to put HTML content from the **content/about.php** file into the Popup Window. We should use the following formatting:

```
<a class="popup" href="#about">About Launcher</a>
```

When we'll specify a "popup" class, the script will load file content with Ajax into Popup Window

Filename of the **php** file from the **content/** folder specified in the **href** attribute **after the hash character**.

As you can see it is very easy to use Inner Linking. All you need is to specify filename in the HREF attribute after the hash character and to specify the "**popup**" class to load content into Popup Window with Ajax.

6. CSS Files and Structure

There are two CSS files used in this Template. You can find css attachment in the HEAD section of index.html file.

```
<head>
-----
<link rel="stylesheet" type="text/css" href="css/launcher.css" />
-----
<link rel="stylesheet" type="text/css" href="css/ie6.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.

launcher.css file 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.

```
/* --- Clock--- */
```

All styling rules for clock timer

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

All styling rules for content pages

```
/* --- Headings --- */
```

Rules for headings.

```
/* --- Forms--- */
```

This section contains style rules for buttons and send email form

```
/*---- Popup Window-----*/
```

Styling rules for the popup window

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

Social icons 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.

And the second CSS file **ie6.css** contains few rules for Internet Explorer 6 browser.

7. JavaScripts

Launcher template uses two JavaScript files

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

```
<head>
-----
<script type="text/javascript" src="scripts/jquery-1.4.4.min.js"></script>
<script type="text/javascript" src="scripts/launcher.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).

The second script file (launcher.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.

Let's look at the custom script more closely.

The first part of the script is used for Ajax loader image initialization, to set target date, to get current date and to calculate difference between them.

```
1 // JavaScript Document
2 /*-----Ajax Loader initialization----- */
3 var ajaxLoader = new Image();
4   ajaxLoader.src = 'images/ajax-loader.gif';
5 /*-----Current and target dates----- */
6 var currentDate = new Date();
7 var targetDate = new Date();
8   targetDate.setUTCFullYear(2012); // Target year in UTC time zone
9   targetDate.setUTCMonth(0); // Target month in UTC time zone (from 0(Jan) to 11(Dec))
10  targetDate.setUTCDate(1); // Target day in UTC time zone (from 1 to 31)
11  targetDate.setUTCHours(0); // Target hours in UTC time zone (from 0 to 23)
12  targetDate.setUTCMinutes(0); // Target minute in UTC time zone (from 0 to 59)
13  targetDate.setUTCSeconds(0); // Target second in UTC time zone (from 0 to 59)
14 /*-----Difference between target and current date----- */
15 var diff = {};
16   diff.ms = targetDate.getTime() - currentDate.getTime();
17   diff.days = Math.floor((diff.ms)/(1000 * 60 * 60 * 24));
18   diff.hours = (diff.ms - diff.days*(1000 * 60 * 60 * 24))/(1000 * 60 * 60);
19   diff.fullHours = Math.floor(diff.hours);
20   diff.mins = (diff.hours - diff.fullHours)*60;
21   diff.fullMins = Math.floor(diff.mins);
22   diff.seconds = (diff.mins - diff.fullMins)*60;
23   diff.fullSec = Math.floor(diff.seconds);
```

Current date

In these variables you need to set target time according to UTC time zone. On this screenshot target date equal to 1st January, 2011 00:00 (New Year)

Some math to calculate difference between target and current dates

```

24 if (diff.fullHours<10) diff.fullHours = "0"+diff.fullHours;
25 if (diff.fullMins<10) diff.fullMins = "0"+diff.fullMins;
26 if (diff.fullSec<10) diff.fullSec = "0"+diff.fullSec;
27 if (diff.days<100 && diff.days>=10) diff.days = "0"+diff.days;
28 if (diff.days<10) diff.days = "00"+diff.days;

```

If some of calculated difference values is one-character we have to add "0" before it or "00" for number of days.

Next part of the script is executed when the document is loaded. Here are animation functions for the clock timer, some effects for links, rules for handling popup links, send email form handling and animation of social bar icons:

Here we have to insert calculated difference values inside div blocks with "day", "hour" and "minute" classes.

```

30 $(function() {
31     $(".day").html(diff.days);
32     $(".hour").html(diff.fullHours);
33     $(".minute").html(diff.fullMins);
34     if ((diff.fullSec+"").length==2) {
35         var second1 = (diff.fullSec+"").substr(0,1);
36         var second2 = (diff.fullSec+"").substr(1,1);
37     }
38     else {
39         var second1 = 0;
40         var second2 = diff.fullSec;
41     }
42     $(".second1").html(second1);
43     $(".second2").html(second2);
44     function changeDay() {
45         var currentDay = $(".day:first").html()*1;
46         var newDay = currentDay-1;
47         if (newDay < 100 && newDay >= 10) newDay = "0"+newDay;
48         if (newDay < 10) newDay = "00"+newDay;
49         $(".d-wrap").prepend('<div class="day">'+newDay+'</div>');
50         $(".day:eq(0), .day:eq(1)").animate({top:"+=80px"}, 2500, function() { $(".day:eq(1) '
51     ).remove() });
52     }
53     function changeHour() {
54         var currentHour = $(".hour:first").html()*1;
55         var newHour = currentHour-1;
56         if (newHour < 0) {
57             newHour = 23;
58             changeDay();
59         }
60         if (newHour < 10 && newHour >= 0) newHour = "0"+newHour;
61         $(".h-wrap").prepend('<div class="hour">'+newHour+'</div>');
62         $(".hour:eq(0), .hour:eq(1)").animate({top:"+=80px"}, 2000, function() { $(".
63     '.hour:eq(1) ' ).remove() });
64     }
65     function changeMinute() {
66         var currentMin = $(".minute:first").html()*1;
67         var newMin = currentMin-1;
68         if (newMin < 0) {
69             newMin = 59;
70             changeHour();
71         }
72         if (newMin < 10 && newMin >= 0) newMin = "0"+newMin;
73         $(".m-wrap").prepend('<div class="minute">'+newMin+'</div>');
74         $(".minute:eq(0), .minute:eq(1)").animate({top:"+=80px"}, 1500, function() { $(".
75     '.minute:eq(1) ' ).remove() });
76     }
77 }

```

Here we have to insert calculated seconds difference. But first of all we have to split seconds number by characters, because first and second characters are in different div blocks.

Here we have to insert first seconds character inside div block with "second1" class, and second character inside the "second2" div.

changeDay() function is used to change day number when the hour' number is changing from 0 to 23.

changeHour() function is used to change hours number when the minutes' number is changing from 0 to 59.

changeMinute() function is used to change minutes number when the first seconds' character is changing from 0 to 5.

```

74 function changeSecond1() {
75     var currentSecond = $(".second1:first").html() * 1;
76     var newSecond = currentSecond - 1;
77     if (newSecond < 0) {
78         newSecond = 5;
79         changeMinute();
80     }
81     $(".s-wrap").prepend('<div class="second1">' + newSecond + '</div>');
82     $(".second1:eq(0), .second1:eq(1)").animate({top: "+=80px"}, 500, function() { $(".second1:eq(1)").remove(); });
83 }
84 setInterval(function() {
85     var currentSecond = $(".second2:first").html() * 1;
86     var newSecond = currentSecond - 1;
87     if (newSecond < 0) {
88         newSecond = 9;
89         changeSecond1();
90     }
91     $(".s-wrap").prepend('<div class="second2">' + newSecond + '</div>');
92     $(".second2:eq(0), .second2:eq(1)").animate({top: "+=80px"}, 500, function() { $(".second2:eq(1)").remove(); });
93 }, 1000);
94 setTimeout(function() { $(".divider").hide(); }, 500)
95 setInterval(function() {
96     $(".divider").show()
97     setTimeout(function() { $(".divider").hide(); }, 500)
98 }, 1000)
99
100 /*-----"Won't wait" animation----- */
101 $("#openContent").click(function() {
102     $(".clock").animate({top: "-70px"}, 600);
103     $(".button-bg").fadeOut(600)
104     $(".content-wrap").delay(600).fadeIn(600)
105 })
106 /*-----"About" links animation----- */
107 $(".ul-links li a").hover(
108     function() {
109         $(".this").animate({marginLeft: "5px"}, 150)
110     },
111     function() {
112         $(".this").animate({marginLeft: "0px"}, 150)
113     }
114 )
115 /*-----Popup Window----- */
116 $(".a.popup").click(function(e) {
117     e.preventDefault();
118     var filename = $(this).attr('href').substr(1);
119     popUp("content/" + filename + ".php");
120 })
121 $(".dark-layer, .close-popup").click(function() { closePopup(); })
122 /*-----Send Email form----- */
123 $("#input#email").click(function() {
124     if ($("#this").val() == "Your email here") { $("#this").val("") }
125 })
126 $("#sendEmail").click(function() {
127     validateEmail();
128     if (!$("#input#email").hasClass("invalid")) {
129         $(".input#email, #sendEmail").hide()
130         $(".sendEmail").append('')
131         jQuery.get("scripts/send-message.php", {email: $("#input#email").val()}, function(status) {
132             $(".sendEmail .ajax-loader").fadeOut(300, function() { $(".sendEmail .ajax-loader").remove(); })
133             $(".message-status").html(status)
134             $(".message-status").delay(400).fadeIn(600)
135         })
136     }
137 })

```

changeSecond1() function is used to change first seconds' character when the 2nd seconds' character is changing from 0 to 9.

This is the main timer function, which is used to animate (scroll) 2nd character of seconds' number. It is executing with 1 second interval. With every second this function adds div block with a new second inside before existing character, then it animates them (scroll), and after, old second' character will be removed. If new second is < 0, then it will be equal to 9 and changeSecond1() function will be executed.

These function are used to animate "colon", to make it blink with every second.

When we'll click on the "won't wait" button, then the "clock" will be animated to top position equal to "-70px" for 600ms, "won't wait" button will be hidden and content section will fade in for 600ms.

Animation for list links. First function will be executed when we'll take the mouse cursor on the list element and second one is used when we'll take the mouse cursor away from it.

Handling rule for links with a "popup" class. When we'll click on it, then the filename value will be equal to HREF attribute of this link. Then the popUp() function will be executed to load .php file from content/ folder.

When we'll click on the "dark-layer" or on the image with a "close-popup" class, then the closePopup() function will be executed.

When we'll click on the email field its value will be removed

When we'll click on the sendEmail button validateEmail() function will be executed. And if the entered email is correct then the input field and "submit" button will be hidden. Ajax-loader image will be added to "sendEmail" block. Then - Ajax request to "send email" php script, and after request is completed, the message with a sending status will be added to "message-status" block.

```

138  /*-----Social bar animation----- */
139  $(".bar-icons a").hover(
140      function() {
141          $(this).children("span").fadeIn(150)
142      },
143      function() {
144          $(this).children("span").fadeOut(150)
145      }
146  )
147  })

```

Animation for social icons. When we'll take the mouse cursor on it, then the children "span" element will be faded in for 150ms.

The next part of code contains functions which could be called from any place of the document. These functions are used to validate email address, to call and to close a popup window:

validateEmail() function is used to validate input field with an "email" id attribute. Validation is based on the regular expressions. If the email address is wrong then the "invalid" class will be added to the email field. Otherwise it will be removed.

```

148  function validateEmail() {
149      var email = $("#input#email").attr("value");
150      var re = /^[^w+([\.-]?\w+)*@((((([a-z0-9]{2,})|([a-z0-9](-[a-z0-9]+))\.)[a-z0-9])|([a-z0-9]+(-?)+[a-z0-9]+\.[a-z]{2,}((com|net|org|edu|int|mil|gov|arpa|biz|aero|name|coop|info|pro|museum|mobi|travel))$)/i;
151      if(re.test(email)) $("#input#email").removeClass("invalid");
152      else $("#input#email").addClass("invalid");
153  }
154  function popUp(location) {
155      var popupWidth = 700;
156      var popupContent = $('.popup-content');
157      var popupLeftPadding = popupContent.css("padding-left").split('px')[0]
158      popupContent.html('').show()
160      $('#popup').css({marginLeft:-ajaxLoader.width/2-popupLeftPadding-10,top:$ (window) .
161      scrollTop()+100}).show()
162      jQuery.get(location,function(data) {
163          $(".dummy-block").css({width:popupWidth}).html(data);
164          $(".ajax-loader").fadeOut(0,0)
165          popupContent.delay(400).animate({width:$ ("dummy-block").width()-popupLeftPadding*2,
166          height:$ ("dummy-block").height(),600,function() {
167              popupContent.html(data);
168              if(isIE6) {
169                  $('#dark-layer').css({position:"absolute",height:$ (window) .height(),top:$ (
170                  window).scrollTop()})
171                  $(window).scroll(function() {
172                      $('#dark-layer').css({top:$ (window) .scrollTop()})
173                  })
174                  $('#dark-layer').delay(300).fadeOut(300,0.7);
175              })
176              $('#popup').delay(400).animate({marginLeft:-$ ("dummy-block").width()/2-10,600)
177          })
178      })
179  }
180  function closePopup() {
181      $('#dark-layer').fadeOut(200)
182      $('#popup').delay(200).fadeOut(300,function() {
183          $('#popup-content').html("").css({width:"auto",height:"auto"});
184      });
185  }

```

popUp() function provides an opportunity to call a Popup Window. First of all function calculates window size to set it in the center of screen with Ajax loader inside of it.

Ajax GET request to specified file.

Then we have to add loaded HTML content to the dummy block to calculate window size for animation. Then Ajax loader will be hidden

"width" and "height" properties will be animated for 600ms, and after - loaded content will be inserted into the "popup-content" container

This case is for IE6 browser, we have to add "absolute" position to "dark-layer" because the IE6 browser doesn't support "fixed" value.

Also we have to animate "margin-left" property of the "popup" container

closePopup() function provides an opportunity to close a Popup Window. Window will be closed using fade animation for 0.3 seconds after the 200ms delay, and after it is closed, HTML content from the "popup-content" container will be removed.

8. "Send Email" Script

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

```
1  <?php
2  if (isset($_POST['email'])) $email = $_POST['email'];
3
4  $mail_to = "demo@idangero.us"; //change this email to yours
5  $subject = "Launcher subscription";
6  $message = "
7  New Launcher' visitor want to know whent the site will be opened.
8  His email: $email
9  ";
10 $headers = "From: ".$email.">\r\n";
11 $headers .= "Reply-To: ".$email.">\r\n";
12 $headers .= "X-Mailer: PHPMailer.">\r\n";
13 $headers .= "Content-Type: text/plain; charset=\>utf-8\>";
14 if (mail($mail_to, $subject, $message, $headers)) {
15     ?>
16
17     <div class="mail-sent">
18         <p>Thanks! Your email has been successfully sent!</p>
20     </div>
21     <?php
22     }
23     else {
24     ?>
25     <div class="mail-error">
26         <p>Error Occured. Try again later.</p>
28     </div>
29     <?php
30     }
31     ?>
```

Checking for existence of **\$email** variable

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 any HTML formatting.

9. PSD Files & Structure

There are **three** layered **.psd** files comes with the **Launcher** template. You can locate them into the **psd/** folder from 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 higher version)

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



Second **psd** file is the **social-bar.psd**, which contains social icons styling:

And the final **psd** file is the **ajax-loader.psd**, which will help you to redesign Ajax loader.

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

10. 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 **Launcher** (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**.