

Quick Admin

HTML template from WebTunes

QuickAdmin is a template for a CMS or site back-end. Implemented using XHTML Strict and CSS3 enhancement brings a clean and easy to use interface. It comes with a variety of components, sample code and great icons to help you define your back-end.

Table of contents

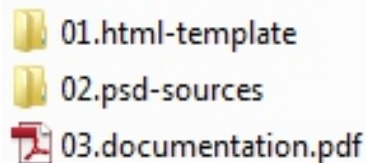
1. Package contents
2. Template structure
3. Template layout
 - 3.1 Template layout
 - 3.2 Grid system
4. Javascript
 - 4.1 Calendar
 - 4.2 Sortable Tables
 - 4.3 Resizable Tables
 - 4.4 Charts
 - 4.5 Panels with tabs
 - 4.6 Gallery - Lightbox
5. CSS Files
6. Design sources
7. Credits & Support

1. Package contents

1.1 Download & Unpack

After you downloaded the package just extract the contents using a known application like winrar, winzip or just windows normal utility if you're a Windows user.

In some cases after unpacking the PSD sources can be also packed due to their large file size in most cases. You can extract the design sources only if you need it, there's no use to have PSD files taking a few hundreds mega bytes taking your space.



1.2 Package contents

The package contains the following files & directories:

- 01.html-template** - the html theme with necessary files
- 02.psd-sources** - the design in psd format with layout & slices
- 03.documentation.pdf** - the product documentaion

2. Template structure

The theme is organized in 3 directories:

- /_layout/** - the layout files (images, javascript, css)
- /_themes/** - the additional css for extra color themes
- /_content/** - the images used as content in template

Theme folder **/_layout/**

All the styles and behavior are loaded using **custom.js** and **styles.js**, all the images used in layout are loaded from **/_layout/images/**.

The folder **/_layout/scripts/** contains all the javascript plugins, each plugin has it's own folder and along with this any additional css files & images :

/_layout/scripts/jquery.colorbox
/_layout/scripts/jquery.flot
/_layout/scripts/jquery.fullcalendar
/_layout/scripts/jquery.morris
/_layout/scripts/jquery.raphael
/_layout/scripts/jquery.tablesorter
/_layout/scripts/uniform
/_layout/scripts/list
/_layout/scripts/nicEdit
/_layout/scripts/table.resizable

All the plugins are loaded and executed in custom.js so any other further enhancements or debugging can be made in this file.

Using this structure brings easy understanding of the theme files and requirements.

/_content

/_layout
/_layout/images
/_layout/scripts
/_layout/scripts/jquery.colorbox
/_layout/scripts/jquery.flot
/_layout/scripts/jquery.fullcalendar
/_layout/scripts/jquery.morris
/_layout/scripts/jquery.raphael
/_layout/scripts/jquery.tablesorter
/_layout/scripts/uniform
/_layout/scripts/list
/_layout/scripts/nicEdit
/_layout/scripts/table.resizable
/_layout/custom.js
/_layout/style.css
/_layout/1140.css

/_themes/

/calendar.html
/charts.html
/dashboard.html
/error-403.html
/error-404.html
/error-500.html
/error-503.html
/faq.html
/form-elements.html
/gallery.html
/index.html
/interface-elements.html
/tables.html

3. Template Layout

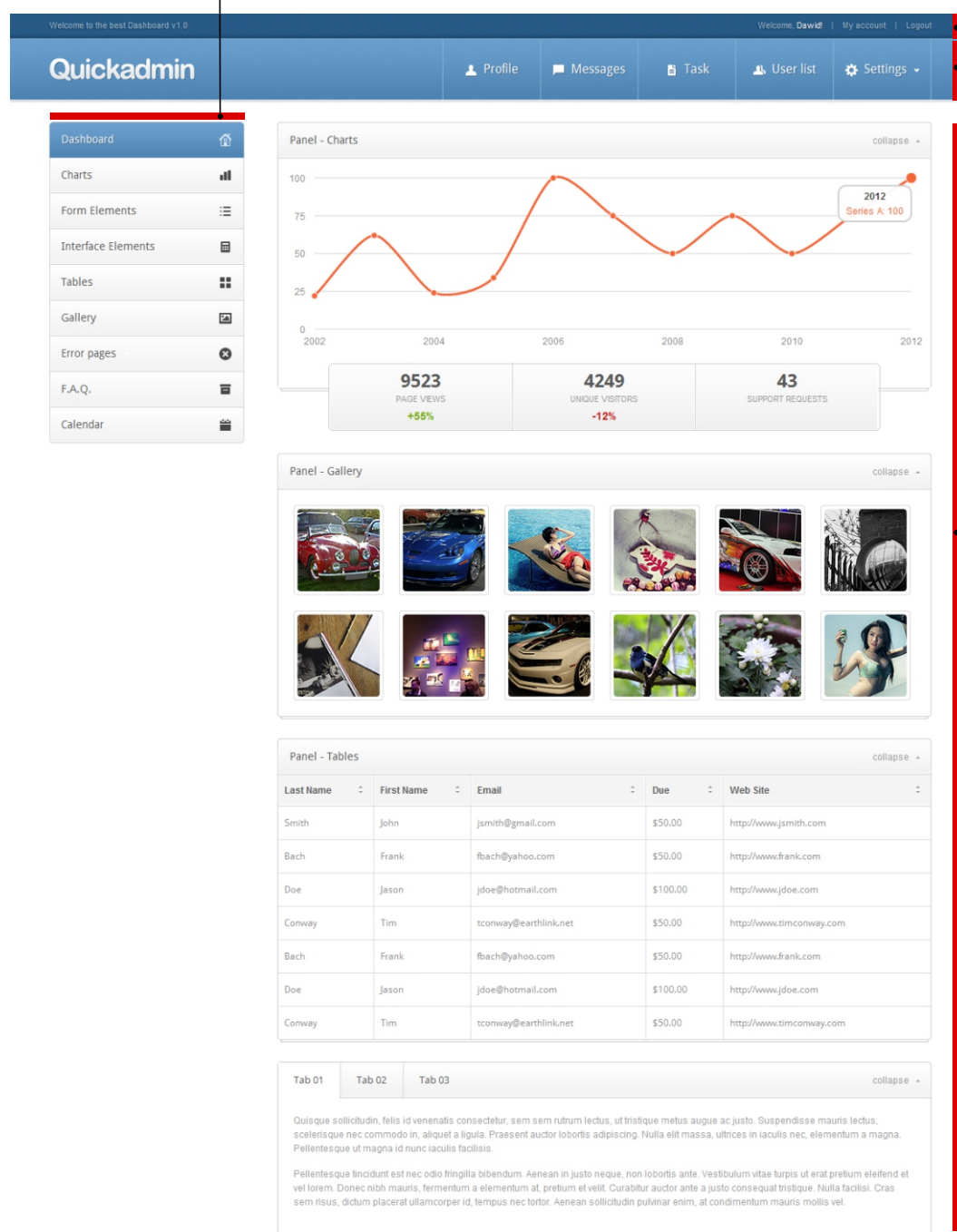
3.1 Template layout

The layout is organized in 5 sections named by id: **#content**, **#sidebar**, **#user-account**, **#user-options**

Here is a sample of general layout structure:

#sidebar

The main page navigation is tab based, to add a new list elements you need to create another li and a link and that's all.



#user-account

#user-options

It's designed to show the user options, account related

#content

The content is structured by grid, this is the actual page content while navigating in the admin.

3. Template Layout

3.2 Grid system

For a simple customization the layout columns are created using a grid, there are a few css clases.

Here are the default columns : **onecol**, **twocol**, **threecol**, **fourcol**, **fivecol**, **sixcol**, **sevendcol**, **eightcol**, **ninecol**, **tencol**, **elevencol** we use a responsive 1140 grid (<http://cssgrid.net/>) :

```
<div>
...
</div>
```

```
<div class="sixcol">
...
</div>
```

```
<div class="sixcol">
...
</div>
```

```
<div class="fourcol">
...
</div>
```

```
<div class="fourcol">
...
</div>
```

```
<div class="fourcol">
...
</div>
```

```
<div class="threecol">
...
</div>
```

```
<div class="threecol">
...
</div>
```

```
<div class="threecol">
...
</div>
```

```
<div class="threecol">
...
</div>
```

```
<div class="threecol">
...
</div>
```

```
<div class="ninecol">
...
</div>
```

Here are some sample columns, when you're using columns for layout you should add the columns in a div container with the class **.fixed**, so the floats are handled and there's no rendering weirdness.

```
<div class="fixed">
  <div class="threecol"> Column one </div>
  <div class="threecol"> Column one </div>
  <div class="threecol"> Column one </div>
</div>
```

4. Javascript

4.0 Javascript

The theme has a custom javascript file: `_layout/custom-revivas.js` - it contains almost all the javascript that loads the plugins and adds extra functionality. Every feature or panel has all the related code in one place, the code is grouped by features.

Loading components

Not all the components are needed in the same time so it will be better if you include them only when you need them.

*** All the plugins are using CSS and Javascript files so we recommend to add the CSS files in the `<head>` and the Javascript files before `</body>` closing tag.**

We will detail every Javascript plugin or functions in the next sections:

```
12
13  $(function() {
14
15      init_tables();
16
17      init_charts();
18
19      init_panels();
20
21      init_wysiwyg();
22
23      init_forms();
24
25      init_calendar();
26
27      init_gallery();
28
29      init_sideNavigation();
30
31      init_logoHover();
32
33      init_faq();
34
35      init_notices();
36
37  });
38
```

4.1 Calendar

The scripts required by the calendar are located in `_layout/scripts/jquery.fullcalendar/`, there are required CSS files and Javascript, we tried to include a minimized Javascript version (to ensure better loading time) and the original version.

Include the CSS file :

```
<link href='_layout/scripts/fullcalendar/fullcalendar.css' rel='stylesheet' type='text/css' />
```

Include the JS file :

```
<script type="text/javascript" src="_layout/scripts/fullcalendar/fullcalendar.min.js"></script>
```

To learn more about the plugin we strongly recommend you to visit the project documentation available here :

<http://arshaw.com/fullcalendar/docs/>

4.2 Sortable Tables

The table behavior is given by **JQuery TableSorter** plugin, it only Javascript files, we tried to include a minimized Javascript version (to ensure better loading time) and the original version.

For sortable tables include the JS file :

```
<script type="text/javascript" src="_layout/scripts/jquery.tablesorter/jquery.tablesorter.min.js"></script>
```

To learn more about the plugin we strongly recommend you to visit the project documentation available here :

<http://tablesorter.com/docs/>

To create a sortable table you must add the class "**sortable**" to your table and it's ready.

HTML Structure :

```
<table class="sortable">  
....  
</table>
```

Javascript

We defined a standard use so it can be very easy to integrate, by default just using the details above you don't need to add extra Javascript code, in case you want to add more classes you can find the code in `_layout/custom.js/`

4.3 Resizable Tables

The table behavior is given by **Table Resizable** plugin, it requires only Javascript files, we tried to include a minimized Javascript version (to ensure better loading time) and the original version.

For sortable tables include the JS file :

```
<script type="text/javascript" src="_layout/scripts/resizable.tables/resizable.tables.js"></script>
```

To learn more about the plugin we strongly recommend you to visit the project documentation available here :

<http://bz.var.ru/comp/web/resizable.html>

To create a sortable table you must add the class "**resizable**" to your table and it's ready.

HTML Structure :

```
<table class="resizable">  
....  
</table>
```

Javascript

We defined a standard use so it can be very easy to integrate, by default just using the details above you don't need to add extra Javascript code, in case you want to add more classes you can find the code in `_layout/custom.js/`

4.4 Charts

The charts behavior is provided by a few plugins: Flot, Raphael, Morris it requires you to include some Javascript, we tried to include a minimized Javascript version (to ensure better loading time) and the original version.

For sortable tables include the JS file :

```
<!--[if lte IE 8]><script language="javascript" type="text/javascript" src="_layout/scripts/jquery.flot/excanvas.min.js"></script><![endif]-->
<script type='text/javascript' src='_layout/scripts/jquery.raphael/raphael-min.js'></script>
<script type='text/javascript' src='_layout/scripts/jquery.morris/morris.min.js'></script>
<script type='text/javascript' src='_layout/scripts/jquery.flot/jquery.flot.min.js'></script>
<script type='text/javascript' src='_layout/scripts/jquery.flot/jquery.flot.pie.js'></script>
```

To learn more about the plugins we strongly recommend you to visit the projects documentation available here :

<http://code.google.com/p/flot/>

<http://dmitrybaranovskiy.github.io/raphael/>

<https://github.com/oesmith/morris.js/>

Due to a hard way of explanations we encourage you to read the scripts documentation.

4.5 Panels with tabs

The Tabs interaction is a custom script, you don't need to include anything else besides **custom.js**.

HTML Structure :

```
<div class="panel-wrapper">
<div class="panel">
  <div class="tabs">
    <ul>
      <li class="active"><a href="#" rel="tab-01-content">Tab 01</a></li>
      <li><a href="#" rel="tab-02-content">Tab 02</a></li>
      <li class="last"><a href="#" rel="tab-03-content">Tab 03</a></li>
    </ul>
    <div class="collapse">collapse</div>
  </div>
  <div class="tabs-content">
    <div id="tab-01-content" class="active"> ... </div>
    <div id="tab-02-content"> ... </div>
    <div id="tab-03-content"> ... </div>
  </div>
</div>
<div class="shadow"></div>
</div>
```

There are two sections, the **tabs** that contain the tabs. The **REL** attribute from the link indicate the ID of the div that's holding the tab content. Follow the code to understand it better.

The section **tabs-content** holds all the <div> with tab contents, create a new <div> with the ID specified in the **REL** attribute.

The class **active** used in tabs and tabs content is used to identify the current tab selected and the current tab selected content.

Javascript

The code that handles the tabs behavior can be found in **custom.js**, look for function : **init_panels()** .

4. Javascript

4.6 Gallery - Lightbox

The Lightbox and Modal Form behavior is given by **ColorBox** jQuery plugin, it requires CSS files and Javascript, we tried to include a minimized Javascript version (to ensure better loading time) and the original version.

Include the CSS file :

```
<link href='_layout/scripts/jquery.colorbox/colorbox.css' rel='stylesheet' type='text/css' />
```

Include the JS file :

```
<script type="text/javascript" src="_layout/scripts/jquery.colorbox/jquery.colorbox-min.js"></script>
```

To learn more about the plugin we strongly recommend you to visit the project documentation available here :

<http://colorpowered.com/colorbox/>

Lightbox

To attach the Lightbox you must use the example below:

HTML Structure :

```
<ul class="content-gallery fixed">
  <li>
    <div><a href="_content/panel-gallery/gallery-item-large-1.jpg" rel="gallery"></a></div>
    
  </li>
  ....
</ul>
```

The link's URL contains the path to the larger image

The image SRC contains the path to your thumbnail image

Javascript

The code that handles the Lightbox behavior can be found in **custom.js**

5.1 Theme CSS

The theme css is located in `/style.css`, the rules are grouped together on sections and scope so you can easy understand what the group refers to, also the naming it's pretty obvious (`#toolbar`, `.panel`, `.notice`).

```
56
57  /*----- Layout for desktop version -----*/
58
59  body.texture { background:url("images/back-
60
61  #header-wrapper { height:120px;margin-bottom:2
62
63  #content { background:transparent; }
64  #sidebar { background:transparent;margi
65
66  #navigation { overflow:hidden; }
67  #navigation li { border-right:1px solid #ccc;
68  #navigation li.first { border-radius:4px 4x 0 0;-mo
69  #navigation li.first:hover { background:url("images/back-
70  #navigation li.first:hover a { background:url("images/back-
71
72  #navigation li.active.first { border-top:none; }
73
74  #navigation li.last { border-radius:0 0 4px 4px;-m
75
76  #navigation .active.sub { border-bottom:1px solid #ccc
77
78  #navigation a { text-decoration:none;font-fa
79  #navigation a:link { }
80  #navigation a:visited { }
81  #navigation a:hover { text-decoration:none;color:#
82  #navigation a:active { }
83
84
```

5.2 General classes

There are a few generic classes added by us to make some fast changes without adding any new code :

.hr - adds a horizontal line

.img - it can be used on place holders, is adding background color, box shadow, you must add **height="size"** and **width="size"** to img

.strong - available in headings if you want to be strong by default

.text-left - aligns text to the left in the selected element

.text-right - aligns text to the right in the selected element

.text-center - aligns text to the center in the selected element

.inpad - is setting the padding size to 15px on every side

.topad - is setting the padding size to 15px on top side

.bopad - is setting the padding size to 15px on bottom side

.hpad - is setting the padding size to 15px on the top and bottom side

.vpad - is setting the padding size to 15px on the left and right side

5. CSS Files

5.3 Customizing the template

Using the current structure any color variation it's called a "theme", there is a **/_theme/** folder in the package with eight available colors.

If you want to create a variation of the current theme, create a new css file in **/_themes/** named **theme-color_name.css**, it's important to create a folder for the images that are used for the theme.

Follow all the steps:

1. create a new css file in **/theme/theme-theme_name.css**
2. create a folder for the theme in **/_theme/** like this **/_theme/theme-theme_name/**
3. don't forget to include the new css file in your html header bellow the **style.css** declaration:

```
<head>
...
<link rel='stylesheet' href='_layout/style.css' type='text/css' media='all' />
<link rel='stylesheet' href='_themes/theme-theme_name.css' type='text/css' media='all' />
...
</head>
```

Keep in mind that CSS files located in **/_themes/** are overwriting the original **/_layout/style.css** rules.

To change a rule just copy the selector from style.css in your new file and add the new changes:

```
body { background:red; text-color:white; }
```

/_layout/style.css

```
body { background:green; text-color:black; }
```

/_themes/theme-theme_name.css

6. Design sources

The PSD files can be found in the package in **/02.psd-design/**, the main file is **Personal Portfolio & Gallery HTML.psd**, it contains all the layouts.

Design Layout

The layout sections and pages are grouped for an easy understanding of design structure.

If you want to show a different page content on the design just hide one of the group located in **Pages** and show the one you want.

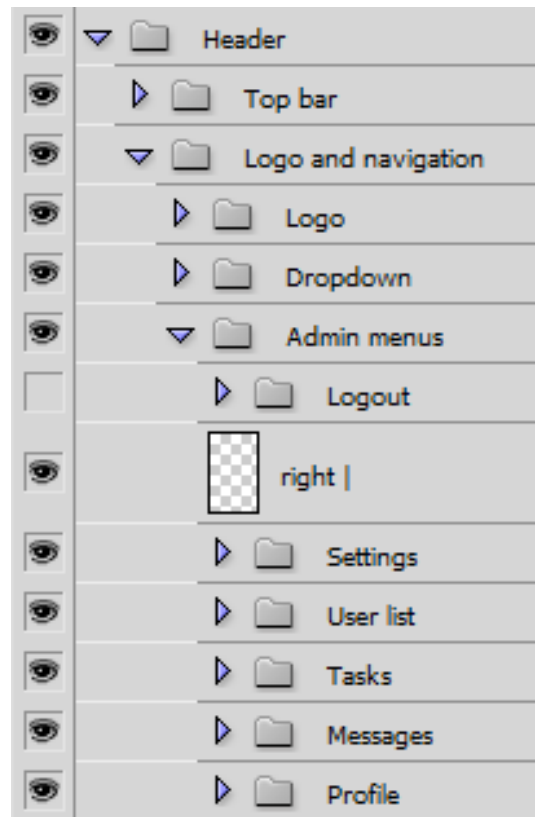
If it's important for you there are also a few guide lines in the layout, just enable the guides by pressing CTRL+R in Photoshop.

Design slices

In the development process during the conversion from PSD to HTML file, we saved some slices that can be useful to easy customize some of the image files.

The slices can be found in **/02.psd-design/_slices/**

*** some files have *.psb extension, there are just small parts of the layout with transparent background ready to be saved.**



7. Credits & Support

7.1 Credits

Creative Commons Images

The images used in the live version are licenced under Creative Commons on www.flickr.com, they are presentation purpose only, not included in the original package and any further use in commercial projects must come along with the author permission.

JavaScript Files

jQuery Framework

<http://code.google.com/p/flot/>

ColorBox - used at image lightbox and modal form

<http://jacklmoore.com/colorbox/>

jQuery Flot - used for charts

<http://code.google.com/p/flot/>

Morris - used for charts

<https://github.com/oesmith/morris.js>

Raphael - used for charts

<http://dmitrybaranovskiy.github.io/raphael/>

TableSorter - used for sorting tables

<http://tablesorter.com>

TableResizable - used for resizing columns on tables

<http://bz.val.ru/comp/web/resizable.html>

jQuery Uniform - used for styling forms

<http://uniformjs.com/>

List - used for dynamic search in Frequently Asked Questions

<http://listjs.com/>

NicEdit - used for WYSIWYG editor

<http://nicedit.com/>

Icons Packages

mimiGlyphs - used on side navigation and buttons

<http://salleedesign.com/resources/mimi-glyphs/>

CSS Grid

1140 Responsive Grid - used on CSS columns

<http://cssgrid.net/>

7.2 Support

If you encounter any problem with our product or you just want to ask something don't hesitate to contact us via ThemeForest using the contact form located on our profile page: <http://themeforest.net/user/webtunes> , or you can send us an e-mail at webtunes.design@gmail.com.

Once again, thank you for purchasing this package,

Webtunes Team