

Spark Documentation

Content

- Image: [The Logo](#)
- Image: [The Logo for Mobile Devices](#)
- HTML: [The Navigation](#)
- HTML: [Links](#)
- CSS: [Some useful CSS classes](#)
- HTML: [The Grid System](#)
- Images: [The Homepage Slider](#)
- Video: [Embed videos](#)
- Javascript: [Forms and Ajax](#)
- PHP: [Set Your Email Address](#)
- HTML: [Include CSS and JS into HTML](#)
- ERROR: [I get a "404 Page Not Found" error!](#)
- [Need help?](#)

Image: The Logo

Path: /images/logo.png

Size: 300×80px

PSD: Logo.psd

The logo is a PNG image with transparent background. **You can re-use this logo by simply modifying the text.** In order to achieve this, I have provided the PSD file with an editable text layer. It makes use of the font "Open Sans" (Bold) which is also provided and can be found in the /Extras/Fonts_Open-Sans/ directory.



[Back to top](#)

Image: **The Logo for Mobile Devices**

Path: `/images/logo-mobile.png`

Size: 50×50px

PSD: `Logo-mobile.psd`

This small version of the logo replaces the regular logo on devices with small screen definitions. You can resize your browser's window to simulate theses.

I recommend using only the symbol of your logo (as in the favicon) without the text next to it, as the size doesn't allow for it.



[Back to top](#)

HTML: The Navigation

The navigation system is implemented in a very simple and natural way. From one side there is **the menu**, and on the other side there is the different “**parts**” of the page.

The **menu** is nothing else than links pointing to IDs located on the page. And each “**part**” of the page is delimited using the `<article>` HTML5 tag that should have the appropriate ID previously pointed-to in the menu.

So for example in the menu we would have:

```
<a href="#contact">Contact Us</a>
```

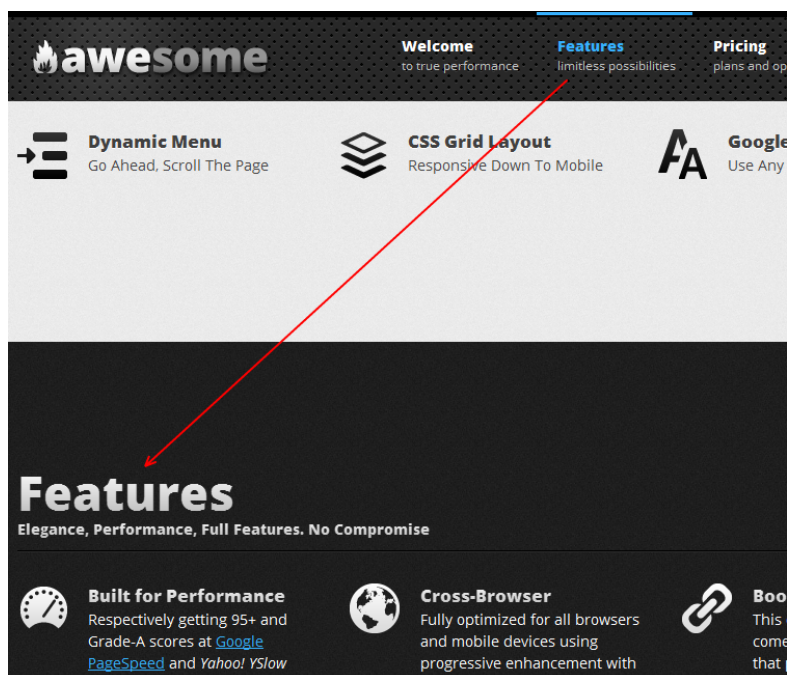
and lower in the page we must have:

```
<article id="contact">...</article>
```

As simple as that!

Note: If your menu contains a link pointing to an ID that doesn't exist, an alert will be shown to inform you that the navigation is broken.

CSS Note: **Do not** use ID to style `<article>` elements, as their IDs get modified by javascript in order to avoid interference with browser's native scroll functions. Instead, add a new CSS class to the desired `<article>` element if required.



[Back to top](#)

HTML: Links

External links: You can have them open in a new window by simply adding the class `.targetblank`, for example:

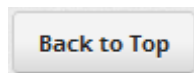
```
<a href="http://www.google.com/" class="targetblank">Link</a>
```

Internal links: Pointing to an existing ID in the current page and would like to have the page scrolls nicely to it? Easy, just add the class `.animate`, for example:

```
<a href="#contact" class="animate">Contact Us</a>
```

Buttons: You can transform any link into a beautiful button using the class `.button`. For example you could create a back-to-top button like this:

```
<a href="#" class="button animate">Top</a>
```

[Back to top](#)

css: The `.dark` class

Applied to any `<article>` element, it allows you to change from the bright global background to a dark (almost black) background for the entire `<article>`.

It's a simple way to add contrast between the different parts of the page, and the great part is that all elements inside the `<article>` section will automatically adapt to a dark background (ie. The text color will switch from black to white, etc.)!

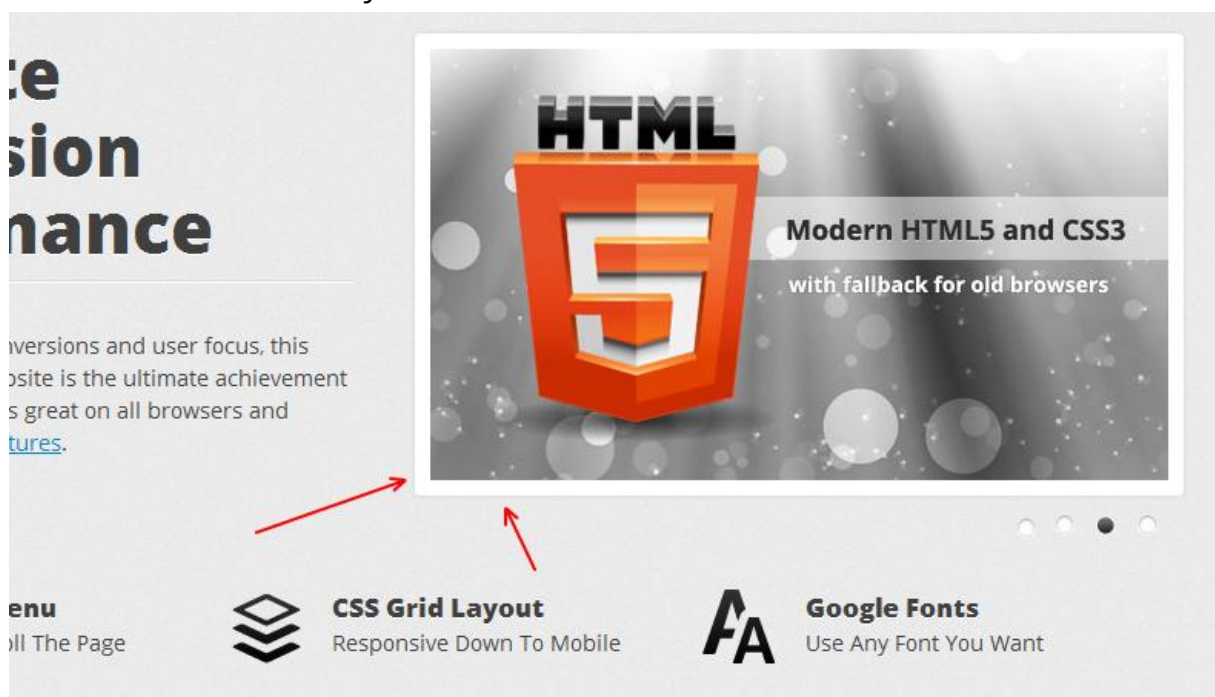
[Back to top](#)

css: The `.box` class

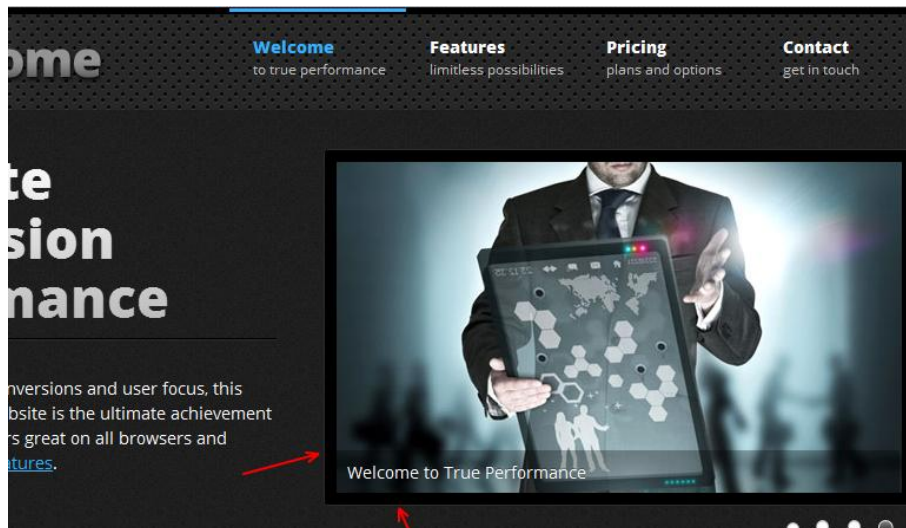
Applied to any HTML block elements (as `<div>`), it adds a nice rounded border and a light background highlight.

The homepage slider doesn't have any particular styling; it's simple images inside a `<div class="box">` with the `.box` class! Beautiful isn't it? Feel free to add this CSS style to any element that you would like to highlight.

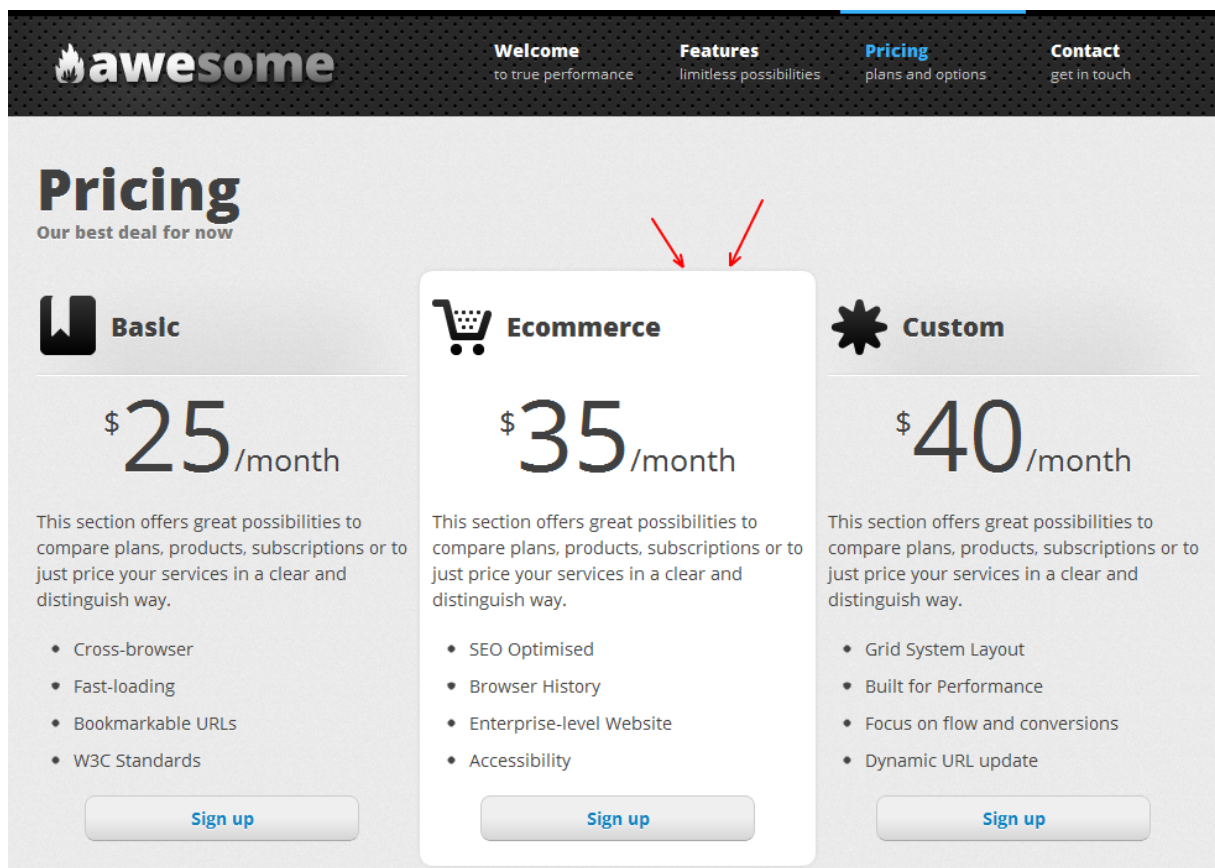
Here is how the `.box` style looks like:



and here's how it looks in a dark context (in this case `<article id="home" class="dark">`):



Another example that makes use of the `.box` class is in the Pricing comparison chart:



Note: In this last example the `.box` class is combined with a new class, `.featured`, which adds bigger rounded corners and adds some negative top margin.

[Back to top](#)

HTML The Grid System

Be assured, it's very simple! It works with rows. In one row there are **16 columns**.

In order to fill an entire row, be sure to accumulate 16 columns.

You can divide a row by 2, 3, 4... and up to 16 columns, or you could also have one row that takes the entire width of 16 virtual columns.

The syntax goes like this:

```
<div class="container"><!--Required container of 960px wide-->
  <div class="x columns">
    Replace X with a number of columns in letters.
  </div>
</div>
```

Here are some examples:

```
<div class="container">
  <div class="sixteen columns">
    This block is 16 columns wide,
    it takes the entire width of the page.
  </div>
</div>
```

```
<div class="container">
  <div class="ten columns">
    Main content
  </div>
```

```
<div class="six columns">
    Right sidebar
</div><!--Note: 10 + 6 = 16 columns -->
</div>
```

```
<div class="container">
    <div class="sixteen columns">
        Title that takes the entire width.
    </div>

    <!-- One row of 16 columns has been completed,
        starting a new row: -->
    <div class="three columns">
        Left sidebar
    </div>
    <div class="six columns">
        Contact form
    </div>
    <div class="four columns">
        Social links
    </div>
    <div class="three columns">
        Right sidebar
    </div><!--Note: 3 + 6 + 4 + 3 = 16 columns -->
</div>
```

More examples are available at getskeleton.com and deeper documentation can be found directly in the `skeleton.css` file located in the `/css/` directory.

[Back to top](#)

Images **The Homepage Slider**

Path: /images/slides/

Size: 520×280px

PSD: Homepage_Slides.psd

Feel free to use the PSD provided to create new slides and to re-use the background images and text styles!

Don't forget to add your new slides in the HTML, the syntax is very simple:

```
<div class="flexslider">
  <ul class="slides">
    <li></li>
    <li></li>
    <li></li>
  </ul>
</div>
```

For information, the slider is then initialized in Javascript in the `main.js` file:

```
$('.flexslider').flexslider();
```

The great part is that you can add as many sliders as you want in the page, with different options/effects.

Full documentation for FlexSlider is available at flex.madebymuffin.com

[Back to top](#)

Video: **Embed a video**

Videos from Youtube (or any other) can be easily embedded. In order to make them responsive (so it scales with the grid) you need to put them inside a `<div>` with the “video” CSS class. Here is an example of an embedded music video, this code snippet can be put anywhere in the page:

```
<div class="video">
  <iframe width="560" height="315"
src="http://www.youtube.com/embed/k9cAyb_Igow" frameborder="0"
allowfullscreen></iframe>
</div>
```

Javascript: **Forms and Ajax**

Built with the same idea of simplicity as the other elements, forms can be simply sent through Ajax by adding the CSS class `.send-with-ajax` to the `<form>` tag.

```
<form action="ajax.php" method="post" class="send-with-ajax">
...
<div class="ajax-response"></div>
</form>
```

Don't forget to add an empty div for the response as shown in the example above!

That's all you need to get a working ajax form, but you could do more: add the required attribute to required fields:

```
<form action="ajax.php" method="post" class="send-with-ajax">
<input type="text" id="name" placeholder="Your name"
required="required" />
```

```
<input type="email" id="email" placeholder="Your email"
required="required" />

<div class="ajax-response"></div>
</form>
```

*Note that the type of second input is **email** so the user won't be able to submit the form until he enters an email. New browsers have native support for these tags, and older browsers get javascript fallback.*

[Back to top](#)

PHP: Set Your Email Address

In order to receive in your inbox the messages sent via the Contact form, you must define your email address in the `ajax.php` file in line 9:

```
1 <?php
2 /*
3     Spark - Simple and Effective
4     Rev. 5
5 */
6
7
8 // Replace the email address with the one that should receive the contact form inquiries.
9 define('TO_EMAIL', 'put_your_own_email_address_here@your_domain.com');
10
11 // === You don't need to change anything else. ===
12
13
14
15 $aErrors = array();
16 $aResults = array();
17
18 /* Functions */
```

[Back to top](#)

HTML: Include CSS and JS files

This is an exclusive feature available with Spark for the very first time on the entire Envato Marketplaces! Using this very simple way of including CSS and JS into your HTML document, you allow all your CSS and JS files to be minified and merged into one file!

```
/min-merge-cache=css/skeleton-v1.1.css,flexslider-v1.8.css,main-v2.css,media-queries-v0.4.css,sprites-v1.7.css,theme-default-v1.css
```

The above line is the URL to the final, minified, merged and cached version of all the listed files!

Important: If you need to include a file that is already minified, please make sure that its name ends with “.min.css” or “.min.js” in order to avoid double-magnification. Also, if you have a particular script that you don’t want to minify but still want it to be merged with the other files, then just ends its name with “.nomin.js”, it will be included as-is without any modification.



Important: If you edit one of the file, you will not see the changes when refreshing the page. This is normal behavior as the resulting merged file is cached on both the server and the browser – This is an official recommendation for web performance (see Yahoo! Performance Rules and Google Page Speed). In order to clear both caches, all you need to do is to upgrade the version by renaming the file + updating the html code with the latest version number:

For example if I edit the file “main-v1.3.js”, I will need to rename it to “main-v1.4.js” and to update this name in the HTML document where it is included (to call the v1.4 and not the v1.3).

[Back to top](#)

ERROR: I get a “404 Page Not Found” error!

Don't panic. This error occurs because you might have placed the site in a subfolder and not on your server's root (for example in www.example.com/folder/ instead of just www.example.com).

To solve this issue, open the file “.htaccess” (I recommend [Notepad++](#)) and replace all occurrences of “/static-server/...” with the full path prepended.

For example if you placed Spark in a /folder/, simply add “/folder” before each occurrence of “/static-server/...”:

Before:

```
/static-server/...
```

After:

```
/folder/static-server/...
```



You will need to add `"/folder"` before `"/static-server/..."` **4 times**: on line 6, line 10, 11 and 12.

[Back to top](#)

Need more help?

Please feel free to contact me for anything! I'll provide full support for using, customizing and extending all aspects of this website.

You can contact me through the ThemeForest's comments or directly to my email: jonathan+dev@maddim.com – I'll try my best to answer promptly.

[Back to top](#)