Blenderbox Guidelines for HTML Templates

last modified: 08.28.09
version: 1.0
created by: Caleb Brown

contact: Caleb Brown

Technical Lead

cbrown@blenderbox.com

tel 718 963 4594 x106
fax 718 218 7804

www.blenderbox.com

0.0 INTRODUCTION

This document is a list of guidelines, tips, and examples for creating HTML, Javascript and CSS for Blenderbox projects. Where applicable, please follow these guidelines to create a clean, SEO friendly website that adheres to Blenderbox's best practices.

1.0 THE BASICS

- Use the default package for directory structure and the base styles.

- Under most circumstances, **do not use in-line Javascript** and **never use in-line CSS**.

- Do not make the CSS overly complicated, but rather let the styles cascade where reasonable and applicable while using as little mark up as necessary.

- CSS files should always be named after the media type they are associated with. For example, a CSS file associated with the 'all' media type should be named 'all.css.'
A list of all W3C types can be found here http://www.w3.org/TR/CSS2/media.html.

- Do not edit the RESET block in the package's '/stylesheets/all.css' file. This style comes from Yahoo's YUI project and helps to make pages cross-browser compliant.

- Do not use the @import tag in your CSS but instead use the link attribute.

- Keep scripts at the bottom of the page and CSS links in the head of the page.

- When beginning work on the templates, always add default styles for HTML content in the HTML block of the '/stylesheets/all.css' file.

- Organize the CSS into logical sections such as HTML, classes (site wide only), layout, promos, etc. as seen in the '/stylesheets/all.css' file.

- Do not use tables when you can avoid it, and most likely, you can avoid it.

- Always use a list when listing something, i.e. a list of news items in a promo chip, a list of subsections in the left navigation, a list of search results, or a list of links to pages on the site in the footer. Following this guideline helps Search engines understand the content better and focus on what content is important.

- Make sure **all** links and buttons have a rollover state. If you cannot find a rollover or on state in a PSD, please contact project management and let them know and they will provide you with one.

- Only use ids when there is only one element on the page that needs that style, otherwise add a class.

- Do not put all images in the /images folder, but rather, logically separate them into sub-directories. The most commonly separated image folders are already a part of the default package.

2.0 TIPS TO GO BY

- Use the line-height attribute when vertically aligning a single line of text or input field. This is often helpful when styling footers etc.

- Add padding to input fields so the text does not run flush against the field on the top, bottom, or left side. When working with this, make sure the descenders of the text are not cut off by the bottom of the input.

- Write the CSS with a hierarchical approach. For instance, if the header element appears on the top of the page, put the #header element at the top of the CSS file.

- All label tags attributes should be set to the value of the input field id so a user can click the label to select the field. All fields should have a label if there is design for it.

- Do not add padding to block elements with set height or width without adjusting the height and width to account for the padding. Under most circumstances, please only add padding to elements with no set height and width.

  For instance, if you have an input field with the style

  input.inputField { height:30px; width:150px; }

  and you would like to add a padding of 2px to the field, the style must become

  input.inputField { height:26px; width:146px; padding:2px; }

- Use div tags with a non-breaking space to clear. The basic clear classes are clear, clearLeft, and clearRight. Using paragraph tags or break tags can add unwanted spacing.
  `<div class="clear"> </div>`
  `<div class="clearLeft"> </div>`
  `<div class="clearRight"> </div>`

3.0 DELIVERING THE TEMPLATES
- Make sure the CSS and HTML on all pages validates. You can use the Firefox web developer plug-in to test validation locally (Tools > Validate Local HTML and Tools > Validate Local CSS).
- Test the files in IE6, IE7, IE8, Firefox, and Safari
- Make sure there are no 404s before sending the final package. This can most easily be accomplished by viewing the Net tab in the Firebug extension.

4.0 BUTTONS / IMAGE ROLLOVERS / IMAGES WITH TEXT

Do not use Javascript to handle rollovers of buttons, navigation elements, etc. because it means browsers with Javascript disabled will not see the correct styles for the site. Instead, use CSS sprites, the technique outlined here - http://blog.blenderbox.com/2008/03/11/css-and-rollovers/. The package already has an example of how to handle navigation buttons that use images. Two things to note that are not included in the post are one, you should always put the text from the image in the link, and two, use 'text-indent:-2000px;' to make the text is invisible to the user but visible to search engines and screen readers. This is a best for SEO and 508 compliance and is included in the default package.

For images with text like logos, follow the same principle to make the page more SEO Friendly.

5.0 IMAGES

- Cut out as much white-space from an image as you can.
- Save .png files instead of .gif files when you can. If you do use .png files, make sure to account for the fact that IE 6 cannot natively display .png files. Look into pngfix.js and make sure to apply it to the CSS as well as inline images.
- Save .gif files when there are few colors or the image is a vector to optimize for performance and quality.
- Save .jpg files at 'High' in the Save for Web settings when using Photoshop.
- All images should have alt tags describing the image and their height and width attributes should be set to aid with browser performance.

6.0 THE DEFAULT PACKAGE

/images

> **/backgrounds** - All backgrounds go here.
>
> **/buttons** - All button images go here.
>
> **/emails** - All images for emails go here.
>
> **/headers** - All headers go here.
>
> **/icons** - All icons go here.
>
> **/nav** - Images for the navigation go here.
>
> **/promos** - All images for promos are here.
>
> **bullet.gif** - This is the image for use with unordered lists.

/javascripts

> **/behaviors** - All page specific Javascripts go here.
>
> **/jquery** - All jQuery plug-ins and miscellaneous jQuery related files files go here.
>
> **application.js** - All Javascript related to the entire application goes here.

/stylesheets

> **all.css** - Styles for all media types.
>
> **all-ie.css** - Styles for IE users. This style is included using a conditional in the head tag that validates.
>
> **all-ie6.css** - Styles for IE6 and below users. This style is included using a conditional in the head tag that validates.
>
> **print.css** - Styles for printing a page. This should mostly just give elements on the page the display:none attribute.

index.html - A basic file for starting the site with most relevant HTML elements for style.

favicon.ico - Ask design for a favicon or get creative and create one yourself (don't spend too much time on it as you may have to re-do the work regardless).