Website Performance Checklist 1.0

Requirements: Firefox, addons: YSlow, lori, Live HTTP Headers, Web Developer Toolbar

URL _____ Date ____ / ____ / _____

Basic Performance Optimizations
[ ] Static Page: If it is slow ask your *webhost*
[ ] Dynamic Page: If it is slow implement caching, identify sources of low performance
[ ] WordPress: Install the *WP Super Cache* plugin
[ ] Any File: Check webserver performance with *DownTester* or *SpeedTest.net*
[ ] Check embedded JavaScripts and images from external domains and decide which are really required
[ ] Avoid HTTP header redirects, use the correct URL directly (use *YSlow* to find out if the page is affected)
[ ] Reduce DNS Lookups: try to keep the number of different hostnames in your page as low as possible
[ ] If it makes sense to host files on a Content Delivery Network (CDN) do so
[ ] Identify files where to add expiration headers and use *mod_expires* in Apache to add expiration headers to files
[ ] Add output compression to your site using either *mod_deflate* or `ob_start("ob_gzhandler");` in your PHP files
Images
[ ] Use the *WebDeveloper Toolbar* to identify the number of images and whether they can be optimized (type, compression)
[ ] Use *Firebug* and open the network tab to find out which images could be merged (CSS sprites, see spritegen.website-performance.org)
[ ] Use Images->Find Broken Images in the *Web Developer Toolbar* to check if the page contains references to non-existent images (i.e. 404)
HTML
[ ] Cleanup HTML code, remove comments
[ ] Streamline your HTML page and avoid unnecessary tags and inline CSS and JavaScript - prefer CSS over tables
Cascading Style Sheets (CSS)
[ ] Store CSS code in an external file and put the `<link rel...>` into the `<head>` of the document
[ ] Compress CSS, e.g. using *CSSDrive.com*
[ ] Avoid CSS Expressions for Internet Explorer
[ ] Don't use the AlphaImageLoader filter in CSS for PNGs and if you need in only use it in IE6 for PNGS with real alpha transparency
JavaScript & AJAX
[ ] Store JavaScripts in an external file and add the `<script>` tag to the bottom of the page
[ ] Compress JavaScript e.g. using *JavaScriptCompressor.com*
[ ] Load JavaScript framework code from Google via the *Google AJAX Libraries API*
[ ] If your AJAX responses are not user- or state-dependent make them cacheable by using GET instead of POST

Comments? Suggestions? Just go to http://www.saschakimmel.com/checklist and let me know.