

PROJECT SPECIFICATION

Build a Portfolio Site

Design

CRITERIA	MEETS SPECIFICATIONS
Required Elements	<p>The page at minimum includes all of the following:</p> <ul style="list-style-type: none">• at least 4 images• title text (h1, h2, etc.)• regular (paragraph) text• a logo.
Semantic HTML	<p>HTML5 semantic tags such as <code><header></code>, <code><footer></code>, <code><article></code>, <code><section></code> etc. are used to add meaning to the code.</p> <p>No <code><div></code> or <code><section></code> tags are without a <code>CSS</code> class or id.</p> <p>Check out the W3C documentation on HTML Structural Elements to learn more!</p>
Custom Design	<p>Provide at least one of the following customizations:</p> <ul style="list-style-type: none">• Customize images and text.• Customize placement of the elements on the page (grid layout) with <code>HTML</code>, <code>CSS</code> or both.• Customize <code>CSS</code> styles applied at minimum to paragraph and heading elements.

Grid-Based Layout	<p>Page utilizes a grid-based layout with styles making use of the <code>flexbox</code> layout or a framework like <code>Bootstrap</code>, <code>Foundation</code>, etc.</p> <p>If you're using <code>Bootstrap</code> or standard <code>HMTL/CSS</code>: the rows and columns of the grid must be wrapped in an element with a <code>container</code> class.</p>

Responsiveness

CRITERIA	MEETS SPECIFICATIONS
Cross-Device Compatibility	<p>All content is responsive and displays on all display sizes. This includes:</p> <ul style="list-style-type: none"> • Desktop • Mobile: Google Nexus 5 • Tablet: Apple iPad <p><i>TIP:</i> Test responsiveness with Chrome Developer Tools device emulation by right-clicking anywhere on page, selecting 'Inspect Element', clicking the rectangle to the left of the Elements tab, select Apple iPad or Google Nexus 5 from Device drop-down list, and click reload.</p>
Provide All Content	All content is rendered on all three devices. No content should be hidden on mobile devices.
Viewport meta Tag	Viewport <code>meta</code> tag is included in <code>HTML</code> . (i.e. <code><meta name="viewport" ...</code>)
Responsive	If a CSS framework is used, classes provided by the CSS

Images	framework are used to make images responsive, otherwise media-queries are used to ensure responsiveness of images.
--------	--

Separation of Concerns

CRITERIA	MEETS SPECIFICATIONS
Styles Separated From HTML	<p>Portfolio completely separates structure (<code>HTML</code>) from design/style (<code>CSS</code>). There are no <code>style</code> attributes present in the body of the <code>HTML</code> document. There are no <code><style></code> elements in the document.</p> <p><i>Note:</i> It is acceptable to include <code>height</code> and <code>width</code> attributes in <code></code> elements.</p>
File structure	<p>Files are organized with a directory structure that separates files based on functionality. For example:</p> <ul style="list-style-type: none"> <code>css/</code> for stylesheets <code>img/</code> for images <code>js/</code> for JavaScript files

Code Quality

CRITERIA	MEETS SPECIFICATIONS
HTML Formatting rules	<ul style="list-style-type: none"> All code (<code>HTML</code> element names, attributes, attribute values) is lowercase (except <code>text/CDATA</code>). Code does not have trailing white spaces. Indentation is consistent (either all tabs or all 2 spaces or all 4 spaces etc). Code uses a new line for every block, list or table element and indent every such child element (it's acceptable to put all <code></code> elements in one line). [Optional] When quoting attribute values, code uses double quotation marks.

<p>HTML Style Rules</p>	<ul style="list-style-type: none"> • <code>HTML</code> documents use <code>HTML5</code> <code><!doctype html></code>. • Code passes <code>HTML</code> and <code>CSS</code> validators. • <i>[Optional]</i> Code does not use entity references unless necessary e.g. characters with special meaning in <code>HTML</code> (like <code><</code> and <code>&</code>) as well as control or “invisible” characters (like no-break spaces). • <i>[Optional]</i> Code omits type attributes for style sheets and scripts.
<p>CSS Formatting Rules</p>	<ul style="list-style-type: none"> • Code does not have trailing white spaces. • Indentation is consistent (either all tabs or all 2 spaces or all 4 spaces etc). • Code indents all block content, that is rules within rules as well as declarations to reflect hierarchy and improve understanding. • Code uses a semicolon after every declaration for consistency and extensibility reasons. • Code always uses a space after a property name's colon, but no space between property and colon, for consistency reasons. • Code always use a single space between the last selector and the opening brace that begins the declaration block. • Code always start a new line for each selector and declaration. • Code always put a blank line (two line breaks) between rules. • <i>[Optional]</i> Code uses double quotation marks for attribute selectors or property values. Do not use quotation marks in <code>URI</code> values (<code>url()</code>).
<p>CSS Style Rules</p>	<ul style="list-style-type: none"> • Code uses meaningful or generic ID and class names that are as short as possible but as long as necessary. • Code does not use element names in conjunction with IDs or classes. • Code uses shorthand properties where possible. • <i>[Optional]</i> Code omits unit specification after 0 values. • <i>[Optional]</i> Code includes leading 0s in decimal values for readability. • <i>[Optional]</i> Code uses 3-character hexadecimal notation where possible. • <i>[Optional]</i> Code separate words in ID and class names by a hyphen. • <i>[Optional]</i> Code avoids user agent detection as well as <code>CSS</code> “hacks”—try a different approach first.

General Meta Rules	<ul style="list-style-type: none">• <code>HTML</code> templates and documents use <code>UTF-8</code> encoding. (no <code>BOM</code>) i.e. <code><meta charset="utf-8"></code>. <p>*[Optional] Mark todos and action items with <code>TODO</code></p>

Suggestions to Make Your Project Stand Out!

Use `srcset` in the `img` elements to provide optimized images to users on all device sizes.

Include additional JavaScript functionality, while maintaining required components. For example: Bootstrap Navbar, Polymer Components.



English ▼