**Project Overview**

For this project, you'll be building a portfolio website. You will be provided a design mockup as a PDF-file, and you must replicate that design in HTML and CSS. You will develop a responsive website that will display images, descriptions and links to each of the portfolio projects you will complete through the course of your Nanodegree program. Please note that while you should aim to recreate the mockup, you may also use your own custom images to personalize this project.

Once you've successfully replicated the design mockup, you are encouraged to continue tweaking and making customizations to the design to personalize it and make it your own! This is your living portfolio site, so make sure you're happy with it.

**Why this project?**

The first thing potential employers look for isn’t which school you attended or what your GPA was; they want to see what you’ve built! You need a professional portfolio to show off your skills.

In this project you’ll build a great looking portfolio where you can show off images, descriptions and links to all of your projects. You'll do so by writing code that conforms to the [**Udacity Style Guide**](http://udacity.github.io/frontend-nanodegree-styleguide/).

**What will I learn?**

You will learn how HTML is used to define the content of a web page and start to think of every web page as a number of boxes on the screen. You’ll then learn how to position those boxes and make them beautiful using CSS.

You'll accomplish this while conforming to a [**style guide**](http://udacity.github.io/frontend-nanodegree-styleguide/) which is an important constraint once you step into any existing development position.

**Why is this important to my career?**

* Without a portfolio, recruiters won’t even look at your resume much less interview you.
* At their very foundation, web applications are just content delivery mechanisms; HTML defines how that language is structured.
* Reading poorly designed content is never an enjoyable experience; CSS empowers you to create beautiful content regardless of your user’s device.

**Note**

If you were previously enrolled in our Intro to HTML and CSS single course and successfully completed this project prior to enrolling in the Nanodegree program (which entails having graduated from the course and having access to your course certificate), simply email us at frontend-project@udacity.com with your passing evaluation and we'll give you credit for this project.

The design mockup for this project is located [**here**](https://storage.googleapis.com/supplemental_media/udacityu/2655898586/design-mockup-portfolio.pdf).

* [**Udacity HTML/CSS Style Guide**](http://udacity.github.io/frontend-nanodegree-styleguide/)
* [**HTML Validator**](http://validator.w3.org/#validate_by_input)
* [**CSS Validator**](https://jigsaw.w3.org/css-validator/#validate_by_input)

## How do I Complete this Project?

All of the lessons you'll need to help you build this can be found in our [**Intro to HTML and CSS course**](https://classroom.udacity.com/courses/ud304), [**Responsive Web Design Fundamentals**](https://classroom.udacity.com/courses/ud893), and [**Responsive Images**](https://classroom.udacity.com/courses/ud882). Depending on your background knowledge of HTML and CSS, you may not need to take the whole thing to complete this project. To evaluate what you'll need to know to do it, start by downloading the mockup we've provided. Next, go to the course to gain the knowledge you need to complete the project.

Be sure to follow the [**Project Rubric**](https://review.udacity.com/#!/rubrics/45/view) closely when designing your page. For example, note that [**HTML5 semantic (structural) elements**](https://www.w3.org/wiki/HTML_structural_elements#Enter_HTML5_structural_elements) must be used to organize content.

Instructions:

1. Download the design mockup file from the Downloadables section (in the lower right hand corner of your screen, just below this text) and review it.
2. Identify the various boxes you will need to build in order to recreate this design.
3. Write your HTML and CSS files, continue to iterate until your work is representative of the design mockup.
4. Take the time to personalize your portfolio with custom colors, additional content and your own images.
5. Validate your HTML and CSS against the [**W3C's Validators**](http://validator.w3.org/). Note: the validators consider the following errors, whereas Udacity accepts these errors as acceptable:
6. HTML5: **Bad value X-UA-Compatible for attribute http-equiv on element meta** when using the X-UA-Compatible meta tag.
7. CSS3: **Property [some property here] is an unknown vendor extension** when using vendor prefixed properties (like -moz-box-sizing).

## What now?

If you need tutorials on HTML and CSS to create this project, go straight over to your first course in this Nanodegree: [**Intro to HTML and CSS**](https://classroom.udacity.com/courses/ud304). Don't wait, get started learning now!

* [**Udacity HTML/CSS Style Guide**](http://udacity.github.io/frontend-nanodegree-styleguide/)
* [**HTML Validator**](http://validator.w3.org/#validate_by_input)
* [**CSS Validator**](https://jigsaw.w3.org/css-validator/#validate_by_input)

# Build a Portfolio Site

**DUE**

May 19

## Project Submission

You will be provided with a design mockup as a PDF-file and must replicate that design in HTML and CSS. You will develop a responsive website that will display images, descriptions and links to each of the portfolio projects you will complete throughout the course of your Nanodegree program.

The design mockup is located [**here**](https://storage.googleapis.com/supplemental_media/udacityu/2655898586/design-mockup-portfolio.pdf).

## Evaluation

Your project will be evaluated by a Udacity Code Reviewer according to the [**Build a Portfolio Project Rubric**](https://review.udacity.com/#!/rubrics/45/view). Be sure to review it thoroughly before you submit. All criteria must "meet specifications" in order to pass.

Submit [**here**](https://review.udacity.com/#!/projects/45/start).

Please note the HTML and CSS style guidelines below - following these is necessary to make sure your code passes the Code Quality rubric point.

## Style Guidelines

### CSS Guidelines

* CSS - Use consistent indentation (tabs or spaces). (See: CSS General Formatting Rules-Indentation)
* CSS - Selectors, properties and property values (with the exception of strings) should be lowercase, including letters in hexadecimal color values. For example #f06c13 instead of #F06C13 and #ccc rather than #CCC. (See: CSS-General Formatting Rules-Capitalization.)
* CSS - Use IDs and class names that are meaningful or generic. (See: CSS-ID and Class Naming.)
* CSS - Use ID and class names that are as short as possible but as long as necessary. (See: CSS-ID and Class Naming.)
* CSS - Use Shorthand properties in all possible places for margin, border, padding, background. Examples: [**https://developer.mozilla.org/en-US/docs/Web/CSS/Shorthand\_properties**](https://developer.mozilla.org/en-US/docs/Web/CSS/Shorthand_properties). (See: CSS Style Rules-Shorthand Properties.)
* CSS - Indent block content between curly braces. (See: CSS Formatting Rules-Block Content Indentation.)
* CSS - End all declarations with a semicolon. (See: CSS Formatting Rules-Declaration Stops.)
* CSS - Add single spaces after each property name’s colon (See: CSS Formatting Rules-Property Name Stops.)
* CSS - Remove trailing white spaces from code. (See: CSS General Formatting Rules-Trailing Whitespace)
* CSS - Include a single space before the opening curly brace. (See: CSS-Declaration Block Separation.)
* CSS - ID and class names should not be used with type selectors. (See Style Guide: CSS Style Rules-Type Selectors.)
* CSS - Separate style rules with a blank line. (See: CSS-Rule Separations.)
* CSS - Remove units of measure after 0 values. (See: CSS Style Rules-0 and Units.)
* CSS - Use leading zeros for values in the code. (See: CSS Style Rules-Leading 0s.)
* CSS - Use 3 character hexadecimal notation for color in all possible places. For example #ccc rather than #cccccc. (See: CSS Style Rules-Hexadecimal Notation.)
* CSS - Separate words in ID and class names with a hyphen. (See: CSS Style Rules-ID and Class Name Delimiters.)
* CSS - Remove CSS hacks or user detection from code. (See: CSS Style Rules-Hacks.)
* CSS - Separate selectors and declarations with new lines. (See: CSS-Selector and Declaration Separation.)
* CSS - Use double quotes for all attribute selectors and property values. (See: CSS Formatting Rules-CSS Quotation Marks.)
* CSS - Identify groups of related style rules with section comments. (See: CSS Meta Rules-Section Comments.)

### HTML Guidelines

* HTML - Element and attribute names should all be lowercase. (See: Capitalization.)
* HTML - Use HTML5. (See: HTML Style Rules-Document Type.)
* HTML - Use UTF-8 as character encoding. (See: General Meta Rules-Encoding.)
* HTML - Place all block/list/table elements on a separate line and indent child elements. (See: HTML-General Formatting.)
* HTML - Use semantic tags. (See: HTML Style Rules-Semantics.)
* HTML - Provide alternate content for multimedia. (See: HTML Style Rules-Multimedia Fallback.)
* HTML - Remove trailing white spaces from code. (See: HTML - General Formatting Rules - Trailing Whitespace.)
* HTML - Use consistent indentation (tabs or spaces) (See HTML - General Formatting Rules - Indentation)
* HTML - Use comments to explain the code. (See: General Meta Rules-Comments.)
* HTML - Mark action items with TODO. (See: General Meta Rules-Action Items.)
* HTML - Remove entity references. (See: HTML Style Rules-Entity References.)
* HTML - Remove type attribute from link and script elements. (See: HTML Style Rules-Type Attributes.)
* HTML - Use double rather than single quotation marks for attribute values. (See: HTML-HTML Quotation Marks.)

## Submission

1. The master branch is the default Github repository branch. If you wish to submit another branch, you'll need to set it as the [**new default branch**](https://help.github.com/articles/setting-the-default-branch/) inside your Github repository.
2. When you're ready to submit your project go back to your [**Udacity Home**](https://www.udacity.com/me), click on Project 1, and we'll walk you through the rest of the submission process. Due to the high volume of submissions we receive, please allow up up to **7 business days** for your evaluation to be returned.
3. If you are having any problems submitting your project or wish to check on the status of your submission, please email us at **fullstack-project@udacity.com** or visit us in the [**discussion forums**](http://discussions.udacity.com/).

## What's Next?

You will get an email as soon as your reviewer has feedback for you. In the meantime, review your next project and feel free to get started on it or the courses supporting it!

#### **Supporting Materials**

**[Videos Zip File](http://d2uz2655q5g6b2.cloudfront.net/2736698543/P1%3A%20Build%20a%20Portfolio%20Site%20Videos.zip" \t "_blank)**

**[design-mockup-portfolio.pdf](https://d17h27t6h515a5.cloudfront.net/topher/2016/August/57b22517_design-mockup-portfolio/design-mockup-portfolio.pdf" \t "_blank)**

You have not submitted the project yetSUBMIT PROJECT