

Project 3: Multipage Website

Due: 05/07/2025

- **General and Global Requirements**

In this assignment, you will create a small website using your Project 2, and **three** other students' Project 1s.

- Each original HTML document will be its own webpage in your website. (4 pages of content)
- You must try to maintain the integrity of each HTML document as-is. i.e., do *not* change the HTML documents by adding content, or moving content around much.
- There are places where you have to add content – described later in this document. Also, it's okay to move content around to maintain consistency between pages. You may also remove internal navigation (if any), and resize or crop images as necessary.
- Try to use as much of the CSS you created in Project 2 as a starting point for Project 3. Your goal is to use all the original CSS you started with, and only add or tweak CSS as necessary.
- It *will be* necessary to tweak the CSS.
- You will also create a new homepage that will serve as an entry point to all the other pages. (total of 5 pages)
- There must be a common (consistent) multicolumn layout on all pages except the homepage.
- You may use a multicolumn layout on the homepage, but it's not required.

It is your responsibility to obtain original Project 1 documents (HTML and image files) from other students.

- **Validation**

All HTML and CSS files must be validated. Also: warnings do not have to be fixed.

- **Responsive/Adaptive Design**

Every page on the website must display properly on all devices.

- The website design must work at browser-widths down to mobile-size (480px wide), and large desktops (1200px wide), and everything in between.
- On small devices: there must be no horizontal scrolling and no elements that overlap their borders.
- On desktops, there must be no paragraphs of text that stretch more than about 700px wide.
- On all devices and desktops, all text and images must be legible and usable.

- All files in the file system must be less than 300 KB.
- Remember to check your images file sizes including background images!

- **Search Engine Optimized**

Every page must be optimized for search engines.

- Proper TITLE tags in the HEAD
- Proper semantic use of HTML elements
- Proper outline structure using H1, H2s, etcetera
- Proper use descriptive text in all alt="" attributes in images
- Descriptive text in all hyperlinks
- A **sitemap.xml** file must exist in the website's root (in the **/project3** folder)

- **New Content That Goes on Each Webpage**

Each page in your website must have the following items **in common**

("In common" means that these items must be identical on each page in the website)

- A banner (in a HEADER element)
- A main menu (in a NAV element)
- A footer (in a FOOTER element)
- The overall layout using CSS Grid Layout
- A "container" to keep at least some of the content constrained in some appropriate way
- You just have to demonstrate that you know how to do it somewhere in the website
- **Title tags**
- Each TITLE tag in the five HTML documents must conform to a certain style.
- The home page must follow this pattern:

```
<title>yourname, CSC 170 Project 3</title>
```

- **The Banner**

The banner (header) typically has three parts: (#1) a logo, (#2) some big text, and (#3) some subtext, like a catch phrase.

- The banner must contain some large text that acts as the name for the website.
- You can have the name for the website be simply: your name, or something else if you're feeling creative
- You can choose whether or not to design a logo. (It would be awesome if you did.)
- Here's an example of a combo-logo which also contains big text that acts as the name of the website:

- You must include a catch phrase/subtitle. If you're not feeling creative, at the very least, you can use this:
CSC 170 - Web Design and Development

Banner Requirements

- The banner must be wrapped in a HEADER tag.
- The name of the website (plus the logo if you use one) must be wrapped in an H1
- The H1 (but not the catch phrase) must be wrapped in an anchor tag and linked to your homepage.
- **Main Menu**
- You must have a horizontal-style main menu (navigation), and links to each page in your website.
- The "current" page must be somehow styled differently to indicate that it's the current page
- No matter what page the user is looking at, the main menu must answer the following two questions: (#1) where am I now? (#2) Where can I go?

Main Menu HTML Requirements

- The main menu must be wrapped in a NAV tag.
- The main menu must be coded as an unordered list.
- Each list item must contain an anchor tag that links to one of the other webpages in the website.

You can use [the professor's menu-highlighter.js solution \(ZIP\)](#).

- **Footer**
- The footer on each page (a "page footer" ...not necessarily the only footer) must contain your name and "CSC 170 Webpage Design and Development"
- On each of the four sub-pages (all the pages except for the homepage: index), the page footer must contain a link to the location of the Project 1 from where the content came from.
- You can add other content to the page footer if you're feeling creative, but whatever it is, keep in mind that everything in the footer must appear the same on each page.

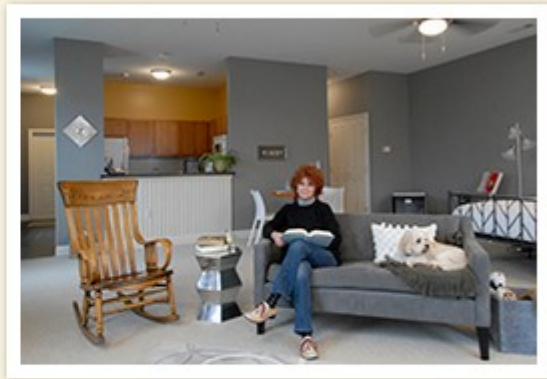
Footer HTML Requirements

- The footer must be wrapped in a FOOTER tag.
- **New Homepage**
- **Homepage layout:** You will create a new homepage (index) that will serve as an entry point to the other four pages. It will have all the same common elements (header, nav, footer and layout) as the other pages, except that it will have new content.
- **Homepage content:** The new content of the homepage must be a summary of each of the subpages
 - Your original Project 1 document, plus the other Project 1 documents you are using.
 - You just need to write an excerpt OR maybe just use the first paragraph of each of the subpages and make it part of your homepage.
- **Design:** Each excerpt must clearly indicate that it's referring to a subpage, for example, each should have its own heading and some other visual clues that tell the user that they're looking at a group of excerpts that each refer to some other section in the website.
- In addition to headings for each excerpt, include embedded links that are redundant with the main nav, that also link to the subpages in your website.
- **jQuery-Powered Slideshow:** The homepage must have a jQuery slideshow that automatically rotates through at least four images. The slideshow must be positioned below the HEADER and NAVIGATION and above all other content. The slideshow must be appropriately sized for the space it fills and work correctly (normally).
 - DO NOT use giant images in your slideshow
 - Note: it's okay to have the slide show occupy one side of the homepage and have the excerpts (below) along the other side.

Here's an example of excerpts that lead to subpages. (Notice the text that says "Read more...")

Jean Allcorn and Hudson M. Valentine

When I decided to move back to my hometown of Rochester, I searched online for an apartment that would fit my criteria. I had a list of five, but Erie Station Village kept coming out on top! Described as quaint and quiet (but with a modern edge), Erie Station evoked memories of old Shortsville, Canandaigua, and the 19th Ward.



[Read more about Jean Allcorn and Hudson M. Valentine >](#)

Jean and James Doyle

To the management and staff of Erie Station Village: This is a very overdue thank-you to all of you! You have made our summer busy and fun! Christine was so gracious on the very hot, humid day of the picnic. Planning an event like that, for all kinds of weather and for all ages, took a lot of time and energy on her part. We do appreciate it! I enjoy meeting with other residents, but especially being with staff members in a relaxed atmosphere.



[Read more about Jean and James Doyle >](#)

• **PHP Includes**

- Your website must use PHP includes on every page to factor-out redundant code. At a bare minimum, you must factor-out the NAV element.
- The include file(s) must be in a sub-folder named **inc**
- When you move your NAV element into an include you must implement a solution (either PHP or JS) to fix the "current menu highlighter" that shows the user what page they're on.

- You don't necessarily have to use the **menu-highlighter.js** file provided by the professor - you can find and implement some other solution

Section: Hidden Verification Fields

Right before the closing `</body>` tag, add the following two hidden fields:

```
<input type="hidden" name="myEmail" value="YOUR_EMAIL_HERE">
```

```
<input type="hidden" name="myAnimal" value="YOUR_FAVORITE_ANIMAL_HERE">
```

- Replace `YOUR_EMAIL_HERE` with your actual email address.
- Replace `YOUR_FAVORITE_ANIMAL_HERE` with any fruit you like.

The section “*Hidden Verification Fields*” will not be graded and it's only to help verify the originality of your work.

Suggested Workflow

- **Prepare your Project 2 HTML file**

- Start by creating a new/empty folder named **project3**
- Make *a copy* of your entire Project 2.
- Rename the Project 2 HTML file from `index.html` to something that describes its content (something like it's H1), e.g. **mark-zuckerberg.php**
- In your copy of your Project 2 page, add the "common elements" described earlier in this document, restructuring the HTML as you go, as necessary.

- **Add the Other Project 1 files**

- When your copy of your Project 2 page is perfect make a copy of the HTML file. Name the file `"start.html"`
- Hollow out the content the `start.html` file (not the common elements – just the content).
- Make three copies of your `start.html` file. Name them names that describes the content that will go in them (the content from the other students), e.g. **alan-turning.html**, **tim-wu.html**, and **biz-stone.html**
- Fix the common navigation links so they all work correctly.
- Carefully copy HTML content from the other students' HTML pages. Be sure to *copy the code* from the other HTML files – not the text from a web browser.
 - You *will* need to re-structure the other pages' outlines (the H1, H2s, et cetera) when you bring them over.
 - If you have a Project 1 that used its own internal navigation, you may remove it.
 - Other than those things, you need to use as much of the HTML from the other students without modification, as much as possible.

- **Build your Homepage**

- Make another copy of your start.html file. Name it **index.html** and fill it with the excerpts and links as described earlier in this document.

- **Generate a sitemap.xml File**

- Create a **sitemap.xml** file and insert it into the root of your website - in the **project3/** folder on the server. In order for you to create the **sitemap.xml** file, you **can** use online resources to read more about it and/or online sitemap generators.

- **Implement PHP Includes**

- Convert your .html files to .php files and factor-out the NAV element and anything else you want to move to separate include files.
- Be sure to install the **menu-highlighter.js** file to insert the "is-current" class on the appropriate menu item for each page.

Remember: after you do this step, you must use a localhost to see your files on your laptop, or upload your files to the webserver and run them there.

- **Finish the rest of the Project**

Now your pages should be set up, just make sure that each page has all required elements. These include:

- Mobile optimization
- The common header
- The footer
- Navigation
- Perhaps more! Make sure you fulfill all requirements and check against the rubric!

Location

Your multipage web site with image(s) and CSS must be uploaded into the **project3** folder on betaweb and also submitted on BB as zip file.

The name of the homepage must be exactly this: **index.php**

- Subpage filenames can be whatever makes sense for the current page, so long as they follow standard naming conventions for webpages: all lowercase. No spaces. Use hyphens to separate words. E.g. for a webpage titled: "Mark Zuckerberg, CEO of Facebook", you would use "**mark-zuckerberg.php**"
- The image(s) you use must be in a sub-directory of the **project3** folder, and the sub-directory must be named **images**
- The external CSS file must be in a sub directory of the **project3** folder, and the subdirectory must be named **css**

Check your work using a browser, as usually.

Report your work

To receive credit for this project: in Blackboard, in **Project 3**, post a link to your webpage and submit the *zipped* folder project3.

Credit

This assignment has been used in previous terms by other instructors, Prof. Read-McFarland, Prof. Kostin, etc.; I've edited it with my own modifications.