

Project 1: Dynamic Web Site

Port a small *static* website (4-6 pages) to a *dynamic* (server-side) website implemented in PHP.

1. Learning Objectives

- Convert a static website to a dynamic one.
- Refine an existing design in support of additional target audiences.
- Implement a dynamic web site with PHP.
- Use partials to reuse web page components across a website.
- Process user input server-side.
- Improve form usability with *sticky forms*.

2. Deadlines & Receiving Credit

Milestone	Points	Grading Method	Slip Days	Deadline
Milestone 1 (<i>p1m1</i>)	15	Feedback (<i>completion</i>)	Not Permitted	2/6, 4:00pm
Milestone 2 (<i>p1m2</i>)	15	Feedback (<i>completion</i>)	Not Permitted	2/13, 4:00pm
Final (<i>p1fin</i>)	100	Rubric	Maximum: 2 days	2/20, 4:00pm

No slip days permitted for milestones. Late submissions will receive a 0.

Milestones are graded *twice*. First for feedback (completion grade only). Lastly, for points at the final submission (via rubric). **All work is graded via rubric for points at the final submission.** Use your milestone feedback to improve your final grade; **revise milestone work prior to final submission.**

Completion credit will be awarded so long as you made a good faith effort to complete the milestone's requirements. Very obviously **incomplete milestone submissions will receive a 0** for the completion grade.

Milestone feedback is not a *pre-grade*. This feedback is designed to catch large problems (which we sometimes miss). Regardless of the feedback (or lack of feedback) that you get, **you are responsible for meeting all of the project's requirements for the final submission.**

Failure to *push* your submission to GitHub is equivalent to not submitting the assignment. You will receive a 0. It is your responsibility to verify that you submitted your assignment.

3. Git Repository & Submission

Clone `git@github.coecis.cornell.edu:info2300-2020sp/YOUR_GITHUB_USERNAME-project-1.git`. Replace **YOUR_GITHUB_USERNAME** in the URL with **your GitHub username**. This is usually your NetID.

Submit **all** materials to your GitHub repository for this assignment. **See *README.md* in your Git repository for submission instructions for each milestone.** Never email your submission to the instructor.

You are **required** to sign the `submit-*.md` files. **Only sign a submission file when you are ready to have your submission graded.** If a `submit-*.md` is signed, we will begin grading the submission at the deadline. If we begin grading your submission, we will not accept any late submissions via your slip days. After the late deadline (deadline + permitted slip days), we will grade all submissions for partial credit regardless of a signed submission file.

Your submission time is based on the time of your last *push* (not *commit*) to GitHub. If you push *anything* after the deadline, then your assignment is late. Your late submission will be graded if the assignment permits slip days and you have available slip days. Otherwise you will receive a 0. **A screenshot of the *commit* times from GitHub is not proof of on time submission;** GitHub does not show push times, it shows commit times.

Tip: Commit and push your changes every time you work on your project. Every time you commit and push you store your changes on the GitHub server. This acts as a back-up for your work. It also means that if you forget to submit before the deadline, there's something already on the server that the TAs can grade for partial credit.

4. Documenting Design (Design Journey)

The design process is the most important part of designing and coding a web site. You will document your design process in the design journey: **documents/design-journey.md**.

The design journey must be written in Markdown. The following links are Markdown references:

- <https://guides.github.com/features/mastering-markdown/>
- <https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet>

You will need to include photos of your hand-drawn sketches in the design-journey. Take a picture of your sketches (I recommend the Microsoft Office Lens app) and place the image files in the `documents` directory and link them in your Design Journey. **DO NOT PLACE DESIGN JOURNEY IMAGES IN THE WEB SITE'S IMAGES DIRECTORY and vice versa!**

We will grade your **design-journey.md** in Markdown Preview. **Everything, including images, must be visible in Markdown Preview.** If it's not visible in Markdown Preview, then we won't grade it. We won't give you partial credit either. This is your warning.

If you included images in your design journey, they must be visible in Markdown Preview. No credit will be provided for images in your repository that are not properly linked in Markdown. No credit will be given for design work not included in the `documents/design-journey.md` file; **do not rename this file or put your answers in another file!** Remember to check and test all assignment submissions!

5. Requirements

Port an existing *static* website (4-6 pages) that you've created for a specific target audience to a *dynamic* website implemented with PHP that now supports two different target audiences.

- Use an existing **static** website that **you created** (100% your work) for a specific target audience.
 - If you do not have such a website, you will need to create one. However, if you're in this class, you should already have one.
 - You may use Projects 1, 2, or 3 (not 4) from INFO 1300.
- **All code must be your own work.** You may not reuse an existing site that you did not 100% create.
- Thoroughly plan your site. **Design and plan first, then code.**
- You will refine the existing design to also support an additional target audience.
- Final website must be designed for **two specific target audiences.**
- **Navigation bar must be implemented with partials.**
 - Navigation bar must programmatically **indicate the current page.**
- Site must include **1 additional partial** besides the navigation bar.
- Site must include a **sticky form with a minimum of 4 inputs** (does not include submit button).
 - Form must provide error message handling via PHP (no JavaScript).
 - Form confirmation page must show the form input from the user.
- **All resources are cited** according to the course citation policy.

5.1. Existing Site

- The existing site must be 100% your own work.
- The existing site must have at least 4 pages.
- The existing site should be specifically designed for a specific target audience.
- The existing site must be **static**: only .html, .css, and image files.
- You must refine the design of the existing site to support an additional target audience (2 total).

5.2. Design Requirements

- Refine the existing site's design to support 2 target audiences.
 - Target audiences must be specific.
 - The two target audiences must be distinct.
 - *Future employers* is specific while general target audiences like *everyone* are not.
- The site must be well-designed, have a consistent "look and feel," and be aesthetically pleasing.
- Design should emphasize content.
 - Avoid hero images; a gigantic pretty image isn't exactly focused on content.

- Site must effectively utilize visual design principles: repetition & consistency, typography, color, contrast, hierarchy, alignment, and proximity.
- Content should be logically organized for both target audiences.
- Site must have clear navigation appropriate for both target audiences.
- Site must include images.
 - **All images must be cited according to the course citation policy.**
- You should design your website for desktop computers.
 - **You do not need to make your website responsive.** You can if you want, but it's not required.
- Site must include a form.
 - Form should be **relevant to users** to serve the actual needs of at least 1 of the target audiences.
 - Contact forms are discouraged.
 - Form must be specifically designed for the target audience(s).
 - Form must be properly integrated into your existing web page. It should not be "tacked" on.
 - Form supports scanning pattern for usability.
 - Form feedback should be specifically designed for your target audience(s).

5.3. Design Process Requirements

- You must demonstrate that you followed the design process.
- You must design and plan first, then code.
- Design must be **thoroughly** planned in the **design-journey.md**.
 - Plan the design through liberal use of sketching, card sorting, etc.
 - **You should show us the evolution of your design.**
 - Minor changes to the design are acceptable during implementation and do not require planning.
 - **Major changes in implementation require revising the design in the design journey.**
- You must document the design of the existing website.
 - Provide sketches of the current site.
- You must refine the design of the existing website by iterating the sketches and iterating the content organization/navigation.
 - You are required to refine/change your design.
 - In each sketch where you refine your design, explain what you've refined.
 - You should re-organize the content to support **both** target audiences.
 - You must thoroughly document the content reorganization process.
 - You must show carding sorting for target audience 1, target audience 2, and both audiences combined.

5.4. Partial Requirements

- All reusable content should be implemented as a partials using PHP's `include()`.
 - You are required to have a partial for the site's navigation.
 - You are required to have 1 additional partial *besides* the navigation.
 - The 2 required partials must be used on all pages of the site.
- The website's navigation should be implemented as a partial.
 - The navigation partial should programmatically indicate the current page.
- All *includes* must reside in the **includes** directory with a **.php** file extension.

5.5. Sticky Form Requirements

- **Form must meet the needs of a least one target audience.**
- Plan your form before coding. Document your pseudocode in the design journey.
- Your site must include at least 1 HTML form with a minimum of 4 controls.
 - You may not count the submit control as part of the 4 controls.
 - 2 of the input controls must be required. (via PHP, not HTML5)
 - All input components must have an associated `<label>` element for accessibility.
- You must implement server-side form validation.
 - Client-side form validation is prohibited. (i.e. No `required` HTML attributes, no JavaScript, etc.)
 - Check whether the required inputs exist in PHP (`isset()` , `empty()`). If not, show an error message to the user.
- Validated and submitted forms should show a confirmation page with the form input displayed on screen.
 - All input must be properly sanitized when placed on the site: `htmlspecialchars()` (do not use `htmlentities()`).
 - **All** inputs must be echo'd to the confirmation page.
- Non-validated forms should return to the user to the form to fix their mistakes.
 - The form must be sticky: the user should not have to re-input what they've already submitted.
 - Error messages should shown for all mistakes.
 - Error messages should be appropriate for both target audiences.

NOT PERMITTED:

- **Search forms are not permitted.** (you need a database for this)
- Forms that do not serve real users' needs.
- No file uploads.

5.6. Coding Requirements

- Site must be coded in valid HTML, CSS, and PHP.
 - **JavaScript is prohibited for this assignment.**
 - All user accessible pages (viewable in the browser) must contain valid HTML.
 - Validate HTML by viewing the PHP page in the browser, then view the HTML source in the browser. Copy the source and paste it into the HTML validator: <https://validator.w3.org/nu/#textarea>
- You may assume that your web page will be viewed in a desktop browser.
 - You do not need to design a web page for mobile browsers.
 - You do not need to use media queries.
 - Your site does not need to be responsive.

5.7. Best Practices

As a professional web developer, your boss will not give you a rubric telling you that your design needs to be aesthetically pleasing or that you need to name the main HTML file **index.php**. It is expected that you know the standards, conventions, and expectations of your field. I expect the same in this class. **This write-up may not explicitly state these expectations and we will deduct points if you fail to follow them.**

As a reference, here are *some* of these expectations that you should already know:

- Your website is designed to **meet the needs of a specific target audience(s)**.
 - Design employs **visual design principles**.
 - Your design is **aesthetically pleasing**.
 - A multi-page site should be **well organized and include proper navigation**.
- Your code is documented with comments where appropriate.
- Main page is named **index.html** or **index.php**.
- The **HTML is well structured** for your site's content (i.e. use of `<header>`, `<main>`, `<section>`, `<aside>`, `<footer>`, ``, ``).
 - **No** `` or `<i>`, etc.
- **External styling via CSS**. No inline or internal styling (i.e. `<style>` elements or `style=""` attributes).
- Your code is **well written, readable, formatted, and properly indented**.
- **No broken or dead links**. Remember that some computers use **case sensitive** file and folder names!
 - *Your instructor's computer is case sensitive.*
- **No hotlinking** of external resources. This almost always includes *embedding*.
 - Google fonts is hotlinking unless you host the fonts locally.
- **Validated HTML5 and CSS3**. You must have 0 errors; warnings are permitted.
- Error free code (PHP). Warnings are okay.
- All files are reasonably sized (~< 1MB) for the web (this includes: images, videos, PDFs, etc.)
 - GitHub has a maximum file size of 100MB.

- Well organized site files.
 - Images are located in an **images** directory.
 - Your CSS files are located in the **styles** directory.
 - Your PHP includes are located in the **includes** directory.
- Your site should display reasonably well (not necessarily identically) across Firefox and Chrome.

Milestone 1: Plan your Web Site

For Milestone 1, you will identify your existing site that you will port. You will also identify your two target audiences and refine the existing design in support of both audiences.

1. Copy Your Existing Site into Your Repository

Copy **the contents** of your existing **web site** into the **root** of your repository.

- Only copy the web site; do not copy anything else. (Only copy .html, .css, and images files.)
- Do not copy the **documents** folder; do not copy the design journey.
- Do not copy the README or the existing site's submit.md files!
- Do not copy the **.git** folder.
- **Just copy the contents of the site.** (.html files, images folder, styles folder, etc.)

Your Milestone 1 submission must have the *original* web site's files to receive credit. Do not yet convert your site to use PHP. **Just commit and push the existing website's HTML, CSS, and images.** That's it!

2. Existing Design

Document the existing design in the **Milestone 1** section the design journey: **documents/design-journey.md**

DO NOT COPY ANY EXISTING DESIGN RESOURCES/TEXT INTO THE DESIGN JOURNEY! Your design journey must be 100% original to this class.

You should document the existing site's target audience, their needs, and sketch the existing design.

A word about sketches: Sketches are quick and help you generate ideas that will improve your overall design. You should sketch on paper. Sketching using a computer program takes too long and misses the point of sketching.

Your sketches don't need to be polished. The lines don't need to be straight. You don't need to write out all text; squiggly lines are okay to *represent* text so long as the text isn't necessary to understand the design. You should only use one color. The idea here is to generate ideas.

Take a picture of your sketches (I recommend the Microsoft Office Lens app) and place the image files in the *documents* directory and link them in your Design Journey. **DO NOT PLACE DESIGN JOURNEY IMAGES IN THE WEB SITE'S IMAGES DIRECTORY!**

Everything, including images, must be visible in Markdown Preview. If it's not visible in Markdown Preview, then we won't grade it. We won't give you partial credit either. This is your warning.

3. Refined Design

Plan the refined design that will support both target audiences. Complete the **Milestone 1** section of the design journey: **documents/design-journey.md**.

You should document the second target audience, their needs, and then refine your design.

Remember here, the design process is what's important here. While the final result is important, what really matters is that you show us how your final result came to exist. Thoroughly document the evolution of your refined design.

4. Partial

Plan out the partials you will implement in the next milestone in the design journey.

Milestone 2: Draft Website

For Milestone 2, you will finish your PHP templates and plan your sticky form **in the design journey**.

1. Port your Site to PHP

Port your existing site to PHP. Rename HTML files to have a **.php** extension and implement your partials. Make sure you include the following line of PHP code on the top of every page (not partial): `include("includes/init.php");`

2. Implement your Refined Design

Implement the your refined design. Move content around, update the navigation, change the styling, etc.

At this point your website should be mostly done except for the sticky form. This means your content should be in place. Your navigation should work. Your CSS is mostly there. The website should look like it's coming together.

3. Sticky Form

In the design journey, plan out your sticky form. After planning it, start coding it. It doesn't need to be sticky yet. The error messages don't need to work yet. You don't need to validate the inputs yet. At a minimum you should `echo()` the inputted values on the confirmation page.

Final Submission: Complete & Polished Website

For the final submission, finish implementing your refined website and complete the design journey. Your final site should be complete and polished. Your form should be sticky and provide server-side only validation to missing inputs (No JavaScript, No HTML `required`, etc.)

Tell us how your complete and polished website meets the needs of your target audiences in the design journey. Also take this time to reflect on your experience of building this website.

