

Web Level 1 - Final Project

Overview

For the final project, you will create a website on a topic of your choice. This project allows you to showcase your web development skills acquired in this program.

Example

Here are some examples of a final project. Note that some of these examples went far above and beyond the requirements.

- Masoma Gym - <https://masomaahmadi.github.io/masoma-gym/>
- Music Together - <https://123kimia.github.io/app-web-final-project/>
- Handmade Crochet Crafts - <https://zhaabdullahi.github.io/app-web-final-project/>
- Everykids <https://shugofa2.github.io/APP-WEB-Final-Project/>
- Personal website - <https://nice-roz.github.io/app-web-final-project/>
- Coffee Shop - <https://zainab1sadat.github.io/app-web-final-project/index.html>
- University of Oxford - <https://zabiada.github.io/zabiada-app-web-final-project/index.html>
- Afghan Tour - <https://arezu-rezaiy.github.io/app-web-final-project/>
- The Nobel Prize - <https://yasminzafary.github.io/app-web-final-project/index.html>
- BTS - <https://elnazmohmand.github.io/app-web-final-project/>

Other Examples

- <https://nayeonk.github.io/app-final-project/project-example.html>
- <https://nayeonk.github.io/app-final-project/plantpal-example.html>

Requirements

General Setup

1. Pick a topic you want to create a website about. It can be any topic -- a personal website, a student organization website, a business website, a website about your favorite thing, or anything you like. It just cannot look exactly like an assignment or lab we did in class.
2. You must use **VS Code** to code your final project and will push this to a public GitHub repository.
3. Create a new folder on your computer's Desktop folder. Name it `APP-Web-Final-Project`.
4. Open this folder in **VS Code**.
5. Create a file named `index.html` in this folder. The `index.html` file will be the home page of your website.
6. Create at least one external stylesheet named `styles.css` inside the `APP-Web-Final-Project` folder. You can write CSS that is shared among other files in this folder. You can create additional `.css` files as you'd like.
7. Create a `.html` file for every page that you will have in your final project.

Requirements

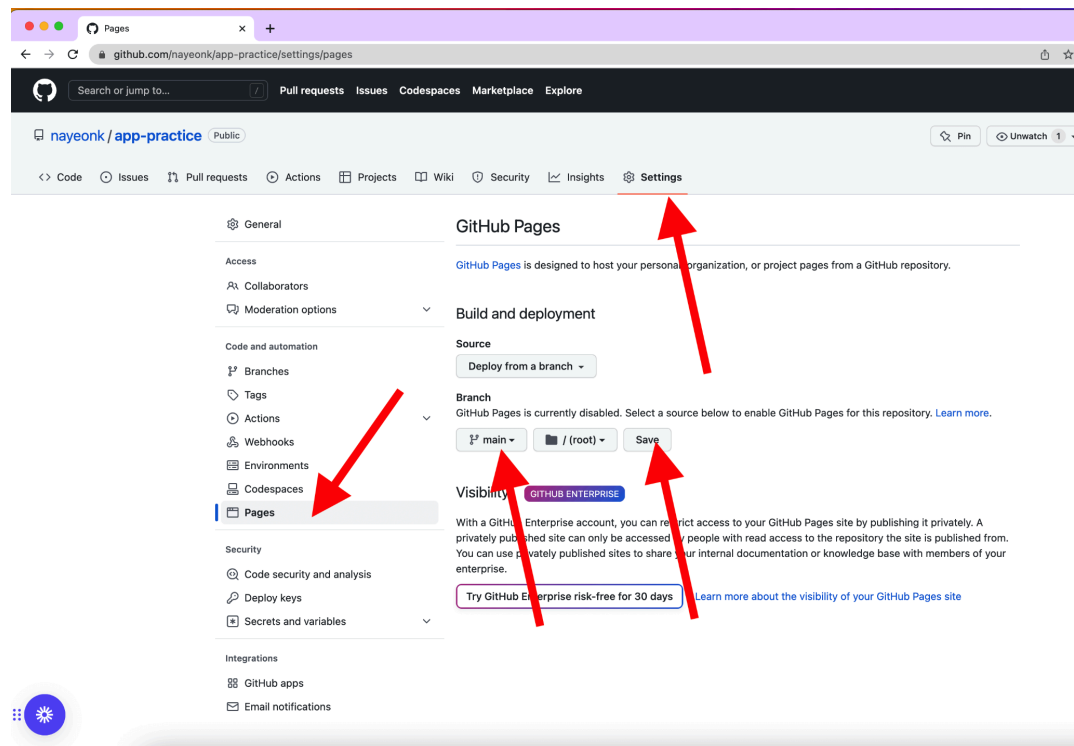
1. The website must have at least **three** distinct pages.
 - a. Form confirmation pages don't count.
 - b. You cannot have just one page with multiple sections.
2. Each page must have a similar design and structure.
 - a. All three pages must have the same **navigation menu**.
 - b. All three pages must have a similar **header** structure.
 - c. All three pages must have a similar **footer**.
3. **Navigation menu** must have at least three working links. Clicking on a navigation menu link sends the user to another page of your website.
 - a. The navigation menu must be built using **CSS Flexbox**. Refer to Lesson #7 if you need to review how to do this.

4. **Form:** There must be at least one form on one of the pages on your website. The form must submit to another page and show a confirmation page.
 - a. You can add whatever input tags you like on the form, but it must have all the correct attributes and label tag.
5. **CSS Grid:** There must be at least one place you use **CSS Grid**. Refer to Lesson #7 if you need to review how to do this. Some examples:
 - a. Photo grid
 - b. Some text on the left side, image on the right side
 - c. A row of testimonials
 - d. See more examples at:
 - i. <https://nayeonk.github.io/app-final-project/project-example.html>
 - ii. <https://nayeonk.github.io/app-final-project/plantpal-example.html>
6. The website must have a clear purpose and topic.
7. The entire website must contain information about the chosen topic.
8. The website must have real content. No placeholder text or images.
9. The website must have a good balance between text and graphics. Do **not** create a text-only or image-only website.
10. The website must be your original work. Do **not** use class examples, labs, or assignments, and just modify some content. Do **not** use HTML templates found online.
11. The website must look consistent across all pages. Pick a color scheme and use the same colors in all pages. Use the same fonts for all pages.
12. The website must use at least **one** font that is not a standard web font. Use Google fonts or a custom font file.
13. Do not use CSS frameworks like Bootstrap or Tailwind for this project. This project is meant to demonstrate your raw HTML/CSS skills.
14. All HTML and CSS code is valid. Before submitting the final project, run your code through a validator to make sure your code is validated.
 - a. HTML: https://validator.w3.org/#validate_by_input ,
 - b. CSS: https://jigsaw.w3.org/css-validator/#validate_by_input

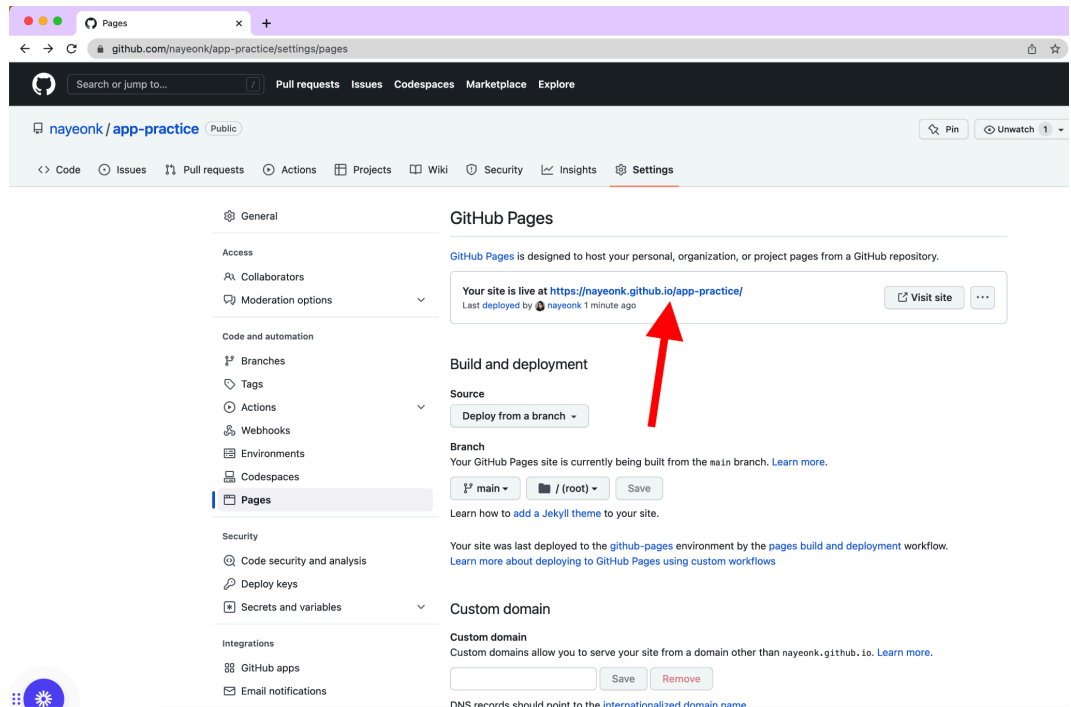
Submitting your Final Project

When you are finished with the project, you will upload your final project files into your GitHub repository and publish the website using GitHub pages.

1. Log in to your account on <https://github.com/>. If you don't have an account, create one.
2. Create a new GitHub repository named **app-web-final-project**. Review Lesson #7 video if you don't know how to do this.
3. Upload your final project files into this GitHub repository.
4. Publish the files using **GitHub Pages**. Instructions:
 - a. Click on **Settings** tab. Then click **Pages** on the left sidebar. Next, select the **main** branch under **Branch**. Click **Save** to save the settings.



- b. Wait a few minutes and reload the page. GitHub will then show you the URL the files have been published to. Click the URL to see your website!



- c. If the message doesn't show up, you can try going directly to the site URL, which would be <https://{username}.github.io/app-final-project>. Replace {username} with your username.
- d. Your website is now live! You can share this link with anyone to show what you've accomplished in this class! 🎉 Direct Message your **GitHub repo URL** and this

URL to your LA to complete your final project submission.

