COMP 2511 Assignment #2: Responsive Design/Accessibility (Work in Pairs)

This assignment intends to modify and improve your work from assignment 1 to create a website that is accessible by different people and is responsive across different devices. **Web Accessibility** and **Responsive Web Design** principles are important aspects of Web development that affect the experience of users with different abilities and devices. You will also, connect your website to a Web hosting tool and make it a public website.

Upon successful completion of this assignment the students will:

- Improve the accessibility of Websites based on an action plan.
- Know how to apply Flex/Grid layouts and to create responsive designs (eg. Using Column Drop, Mostly Fluid, or Layout Shifter)
- Style Websites to be responsive on devices with different sizes. (e.g., small screen, medium screen, and large screen devices)
- Use media queries to define multiple CSS style breakpoints for your websites.

Teamwork

Before starting the project, read the requirements. Then meet with your project partners to divide tasks equally. Team members must contribute equally to the project. The contribution of each member must be visible in the history of your git repository. Make sure to review the pages that are developed by your teammates.

Submission

All assignment files must be included in the git project shared with your instructor, by the due date. No changes must be made to the project after the deadline. The contribution of each team member must be visible in Git history, otherwise, no grade will be awarded. If you have given your instructor access to your repository in your first assignment and you are working in the same repository (which you should be!), there is no need to make any change to the repository permissions.

Make sure to include the **link of your repository**, and the **link of your website** in your D2L submission page.

Part 1 - Re-designing the website

In this part of the assignment, you will be **re-designing** your website! You are free to relax the CSS requirements from the previous assignment and come up with a design of your choice. For the layout of the pages, you should follow the given layouts. Sample sketches are given in this document for small, medium, and large screens. You are expected to adjust the layout of your website for different screen sizes.

Sketches: Your website should look like the following sketches in **small, medium and large** devices. Your break points for small, medium, and large devices should be at **590 px, and 790 px**.

In **small** size device, your page should be shown in 1 column format. The menu items can be shown vertically. Alternatively, you may use a hamburger menu component. In **medium** size devices, you will be using a 2X2 grid in the middle of your page. In **large** devices, the content should be centered with a fixed size of 790 px, and the remaining spaces on the side (Mostly fluid design pattern) should have a plain color of your choice. Note that these are simple sketches, feel free to adjust the sizes and spaces in your design.
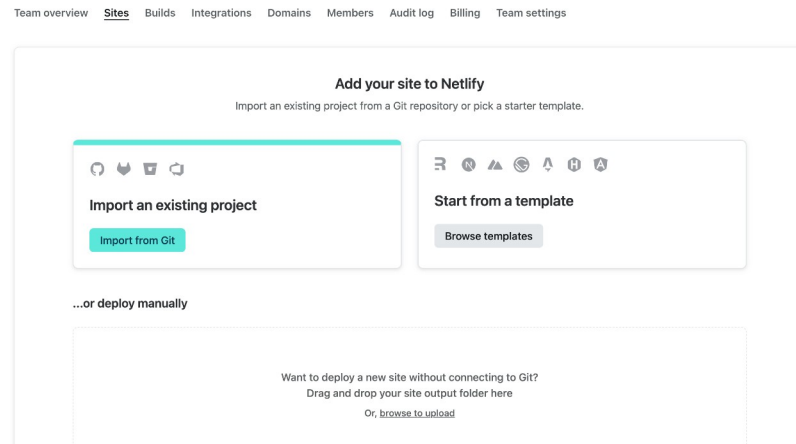

Small device

Medium device

add your logo here!

Your menu items should be here

Write something about your Website
e.g., the name, the slogan, the tagline

Add a relevant image here. Use a nice transparent image

List some of your content (e.g., products, services, etc) in a 2 X 2 grid format.

Include a simple contact form here ...

Add a submit button for your form

Create a footer with a content of your choice!

Large device

Part 2 - Connecting your project to Netlify

1. Create a free account in Netlify (https://www.netlify.com/), or login with your Github login.
2. Add your site to Netlify by importing your git repository.

3. Connect it to your Github repository and pick a repository.
4. The first time you use your Netlify account, you need to connect it to your Github repository and authorize Netlify to access All or Selected repositories.

5. In the following image, <<COMP2511>> is the name of the Github account and <<demo>> is the name of the repository we want to connect.

Import an existing project from a Git repository

From zero to hero, three easy steps to get your site on Netlify.

1. Connect to Git provider   2. Pick a repository   3. Site settings, and deploy!

Pick a repository from GitHub

Choose the repository you want to link to your site on Netlify. When you push to Git, we run your build tool of choice on our servers and deploy the result.

COMP2511 ˅   Q Search repos

⬤ COMP2511/**demo**   🔒 Private   >

Can't see your repo here? **Configure the Netlify app on GitHub.**

6. Click on the name of your repository to move to the next step.
7. Make sure to use the right branch. Most of you might be working on the <<main>> branch, but if it is not the case, select the correct repository to deploy.

Site settings for COMP2511/demo

Get more control over how Netlify builds and deploys your site with these settings.

Owner

MRU ˅

Branch to deploy

main ˄

main

8. Deploy the website and wait until the deployment is complete.
9. Now, your website is ready, and you can access in the URL provided in your Netlify page:

rococo-flan-99be2f

• https://rococo-flan-99be2f.netlify.app

Deploys from GitHub.

Last published at 9:55 AM.

⚙ Site settings   ⚙ Domain settings   ☆ Favorite site

10. Netlify uses the <<index.html>> file in the root directory of your repository as the starting point of your website.
11. Test the URL for your website and make sure everything works as expected. **Include this URL in your assignment submission.**

Part 3 - Website Accessibility

Now, that your website is live, we will do some manual and automated analysis:

Manual analysis - Open your website and evaluate the accessibility of your website based on the accessibility topics that you have learned in class. You may find more criteria in the accessibility checklist provided by WebAIM https://webaim.org/standards/wcag/checklist. If you notice any issues, please fix them and improve the accessibility of your website to the best you can.

Automated analysis - Then, use the Wave accessibility tool at https://wave.webaim.org/ to check the issues in your website. Read the issues identified by Wave and fix them as much as you can!

Grading

	Points
General	
The contribution of each team member must be visible in Git history, otherwise, no grade will be awarded.	
Website is published online.	6
The repository and website URLs are included in the D2L submission.	4
Design	
The header is designed based on the given sketch.	5
The section below header has a background, text, and picture and is designed based on the given sketch.	10
The footer is designed with a custom content.	5
All pages of the website follow the given layouts and designs have professional look.	10
Responsive Design	

The layout of the middle part of the pages (the middle 4 gray boxes shown in the sketches):
 	
   - follows a 1 column format, in small devices.
   - follows 2X2 grid layout, in medium devices.
   - follows mostly fluid layout, in large devices.
	6
	12
	12

Media queries are defined correctly to change to medium and large layouts at 590 px, and 790 px.	15
Web Accessibility	
The website is accessible: the accessibility criteria discussed in class are considered in the design of the page.	15