## Project 1 - Multi Page Responsive Website

***Projects are worth a total of 60% of your final grade.***

| | | | |
|---|---|---|---|
| Project was handed in on time (Y / N) | Y | | |
| Days Late (if applicable) | | Marks deducted: | 0% |

| ***Structure*** | | ***Mark out of 12:*** | 9.75 |
|---|---|---|---|
| Use E (excellent), G (good), S (sufficient), I (insufficient), or P (poor) to rate the student's code on the following: | | | |
| Clean and Easy to Read | G | | |
| Dry | G | | |
| Best practices, as discussed in class have been followed | G | | |
| Project represents the provided design well | E | | |

| ***Evaluation*** | | ***Mark out of 11:*** | 10 |
|---|---|---|---|
| Site is responsive and uses media queries | ☑ | | |
| Flexbox is used | ☑ | | |
| Sass variables are used | ☑ | | |
| All images considered content have descriptive alt text | ☑ | | |
| All forms are implemented using the proper accessibility techniques | ☐ | | |
| Mobile view provides good user experience | ☑ | | |
| Files, folders and project repo follow a consistent naming convention (ie. kabob-case) | ☑ | | |
| Naming convention throughout HTML & CSS is consistent (ie. kabob-case) or follows a known pattern (ie. BEM) | ☑ | | |
| Project is organized using Sass partials (minimum 2) | ☑ | | |
| Navigation should link to other pages made for the project | ☑ | | |
| Site is live on GH Pages, Netlify, or student's own URL | ☑ | | |

| *Presentation* | *Mark out of 1:* | 1 |
|---|---|---|
| Student presented their project | ☑ | |

| | *Mark out of 24:* | 20.75 |
|---|---|---|
| | *FInal Mark:* | 86% |

## *Notes*

**Successes:**

This is looking so great, you've made an excellent representation of the design files and the media queries make this multi-page site responsive on all screen sizes. Mobile has a good user experience too!

All your inputs are linked to labels making them accessible! Also, great to see that you hid your labels with the visuallyHidden snippet.

You wrote descriptive alt tags for all img elements so they are accessible and non-sighted users can have a good experience interacting with them. Also, this text is used for SEO so people can find your page and it will be in place of your image if it can't load for some reason.

**Areas for Improvement:**

The navigation menu you added is really cool. I noticed you aren't using a nav element for it so the browser doesn't know as much about its use. Also, the desktop nav is inside the header but the mobile nav isn't. For a future version you could try and make the HTML more DRY by only writing the nav once and using styling to achieve a mobile nav instead of entirely new elements.

Some of the indentation is inconsistent so it makes reading some of the element hierarchy more difficult. Make sure to indent any nested elements in the HTML consistently. The ul in gallery can be adjusted.

Some of the icons <i> are purely decorative so they can be given the attribute aria-hidden="true" to remove them from the accessibility tree so they are not read out by screen readers. line 95 etc.

The class names about1 - about4 can be more descriptive for the type of content each section would hold. These class names should remain relevant if the copy writing changes. Good class names for this page could be about, main-feature, locations, programs etc.

The social links inside the contact don't have any text content so the browser or assistive tech can't identify them. To make them accessible you need to nest a sr-only/visually-hidden hidden span inside the <a> tag with text like "facebook" in between the span tags.

You can wrap the phone number and email address in <a> tags to turn them into links and use mailto:info@email.com and tel:123-232-2323 inside the href. Then you can click or tap on your phone or browser to use them.

Another semantic element you can use is the <address> for denoting contact information on the page.

The textareas in your forms are not properly linked to the label. textarea needs to have an id and match it to the label 'for' attribute to link them.

It is best practice to not use ids for styling because they are too specific. You can use a class name and then have the option to reuse it making the code more DRY.

I recommend putting mixins and variables inside a separate Sass partial to keep them organized.