

## COMP 1537 – Assignment 3

**Due:** The week of October 10<sup>th</sup>, before the beginning of your scheduled lab class, for marks! This assignment is out of **30** – heavier because you are doing more work!

**This assignment must be done individually.**

### Objective

You can use other layout techniques such as float and clear and change the value of the display property. However, **you must use media queries and you must use CSS grid** as how we went over in lecture. For some more information and reading on CSS grid, read through this article:

<https://css-tricks.com/snippets/css/complete-guide-grid/>

As well, consult the examples and the links at the end of the slides for more information. The links are provided for your reference as well as to serve as an online book and resource for you.

When using the container elements (e.g., section, div, nav, aside, etc.), use the id attribute to uniquely select them with CSS. Make your layouts look like figure 1. Those are three different layout variations that are shown within three media query dimensions (specified below). Note that you may not be able to read the text in this document. That's okay! You are copying the layout, not the content or assets.

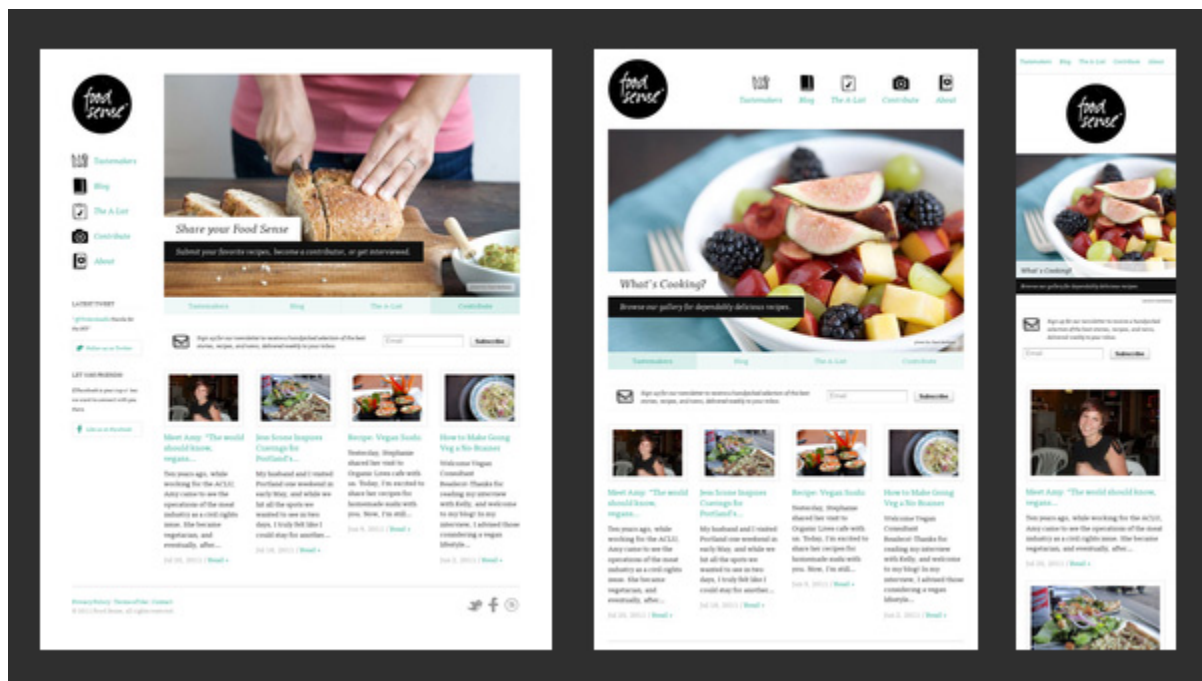


Figure 1

## Requirements

The requirements for this assignment:

- Follow the layout, but use your own content – you will lose marks if you use the same assets from the screenshot (that includes logos, images, etc.)
- Do **not** use image placeholders or “lorem ipsum” text for this assignment, use your own content (can be anything, sports scores, movie review, favorite recipes, etc.)
- Change captions, titles, logos, color scheme, and categories to match your content – not this screen capture
- You must use the following semantic elements:
  - Footer, header section, aside, figure, figcaption
- You must use at least one web font – either with `@font-face` or with `@import`
- Your media queries will support the resolutions of:
  - 700 pixels wide and under
  - 701 to 1280 pixels wide range
  - 1281 to 1600 pixels wide range
- The layout changes for each of the resolutions shown. Make yours do the same
- Use the CSS grid for this and change the grid attributes within your media queries
- Use percentages for widths so that the layout is also liquid – you shouldn’t need any fixed width values although you may need fixed height values for some items (e.g., banner)
- Any text that you see in the screen shots has to be text – not a screen shot of text and images
- You are creating one document that has three media queries in it – not three separate documents
- Your CSS is in an external file (e.g., `style.css`)
- If content doesn’t fit vertically in the browser content pane, the user can scroll down to see more content (we did talk about this but here’s a nice tutorial on how to do it:  
<https://medium.com/@beyondborders/beginner-css-grid-sticky-navigation-scrolling-content-7c4de0a8d1dc> )
- The circle must be re-scalable and therefore cannot be images such as PNG, JPEG, GIF – but they can be CSS shapes, CSS paths, or embedded SVG documents (look in the CSS-basics example folder and view the file, ‘`shapes.html`’ for some examples of using shapes)

One of the main purposes of this assignment, aside from learning how to create layouts in CSS with grid, is to learn how to reverse engineer a layout. Clients will often see a layout on a site/app and ask for you to copy the layout.

Once you have completed, ensure that your document is valid. It is suggested that you use the HTML 5 validator online at:

<https://en.rakko.tools/tools/58/>

To ensure that you don’t lose marks. Also use the CSS validator in Visual Studio Code (CSS Formatter & HTML CSS Support)

As well, include a readme.txt file that follows this format:

```
[Student Name], [Student ID], [Set], [Date]

This assignment is [enter percent]% complete.

[explanation if not complete, what is working/not
working]
```

*Figure 2: readme.txt*

## Submission

Create a zip archive of any and all text files that are part of your assignment submission. Your HTML files will have the extension “.html”, your CSS files will have the extension “.css”, and your JavaScript files will have the extension “.js”. If you have images, be sure to include them in this archive that you create.

Once you create your zip archive file, rename it to follow the format SurnameFirstnameCOMP1537Assignment3. My zip file would be FergusonArronCOMP1537Assignment3, for instance.

Finally, submit it to the learning hub in the folder labeled “assignment 3”.