

# My Portfolio Website



By John Thompson

<https://jtdotcode-portfolio.netlify.app>

# Purpose

## Purpose

The purpose of this project is to demonstrate my skills as a web developer and to showcase my previous projects on a website.

This website could be used to show future employees my skills and abilities which might lead to employment.

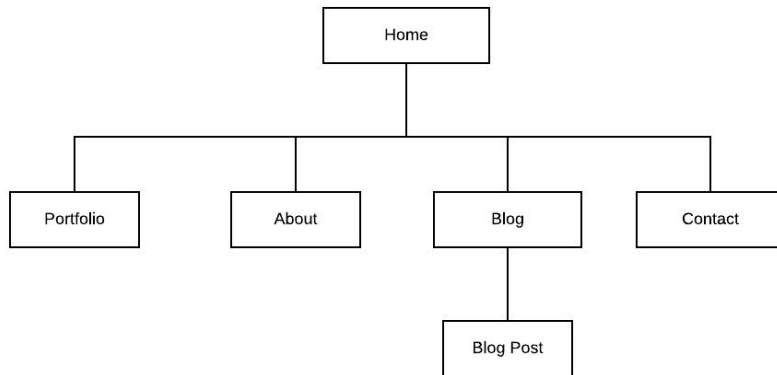
# Structure and Navigation

This website has 5 pages home, about, contact, portfolio, blog which are all linked via two responsive menus, the desktop and iPad navigation menu will expand and shrink from 769px and below that the desktop menu you will hide and the mobile burger menu will appear. The mobile burger menu is based on javascript and CSS is designed is to fill the whole screen when the user clicks on it.

Overall I have tried to design the site so it will be responsive at all resolutions and high contrast so it's easy for people with bad vision to see and engaging at any resolution.

# Overview/Sitemap

## Site Map John Thompson - Portfolio Website



# Styling Ideas

Not being great at graphic design I thought it would be best to keep it simple. While I was putting together the wireframes I was thinking I liked the look of the black and white high contrast wireframe. So maybe I will just make it look like the wireframe.

I also felt that when I added my portrait picture it was to clean for the look, so I thought maybe make it look like a b&w photocopy which would give the site a wanted poster feel.

I also choose to have the portrait picture on each page which made the site also have a dear leader feel as well, which I think makes people think about me.

Overall the theme is designed to make the audience think about a paper resume.

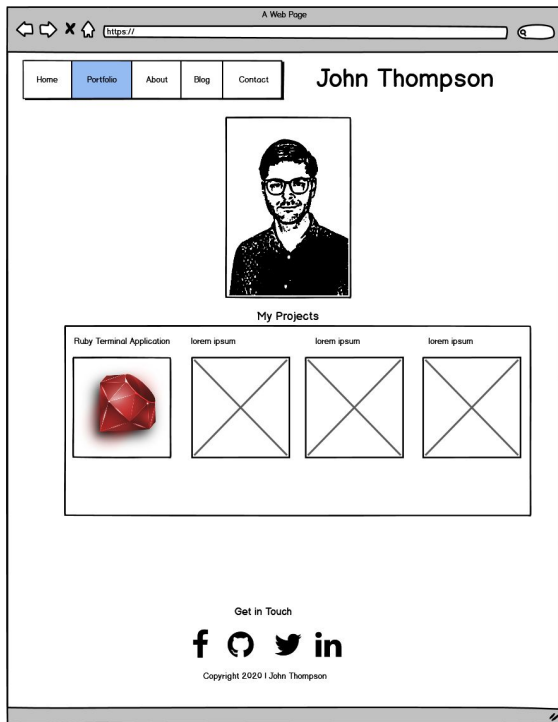
# Wireframe vs Realty (I suck at CSS)

I had some issues with implementing my original wireframe due to my lack of understanding of flexbox and grid; the most issues I had was with getting content to flow or wrap.

In the end I used a combination of grid and flexbox and I just implemented a single column design. When I started using the scaling properties like vw I found that I didn't need to have the elements wrap to the nextline for the site to look ok.

Over all the site look fairly close to the wireframe.

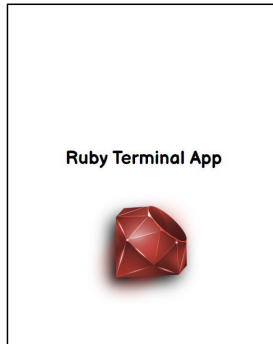
# Wireframe vs Realty (I suck at CSS)



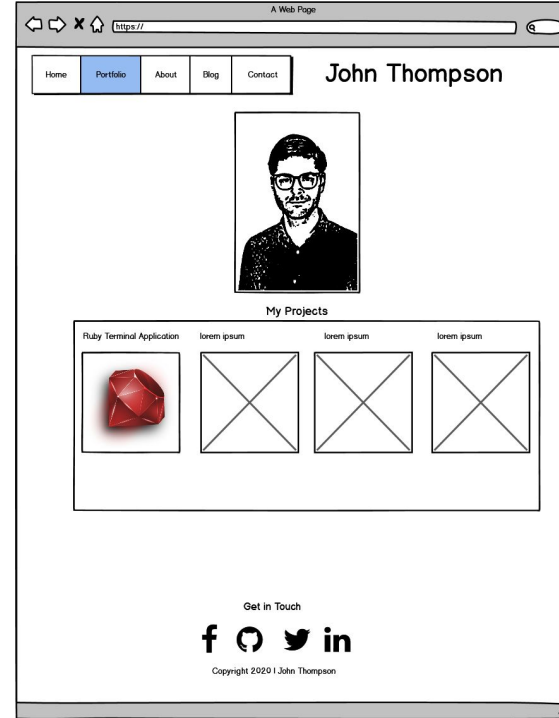
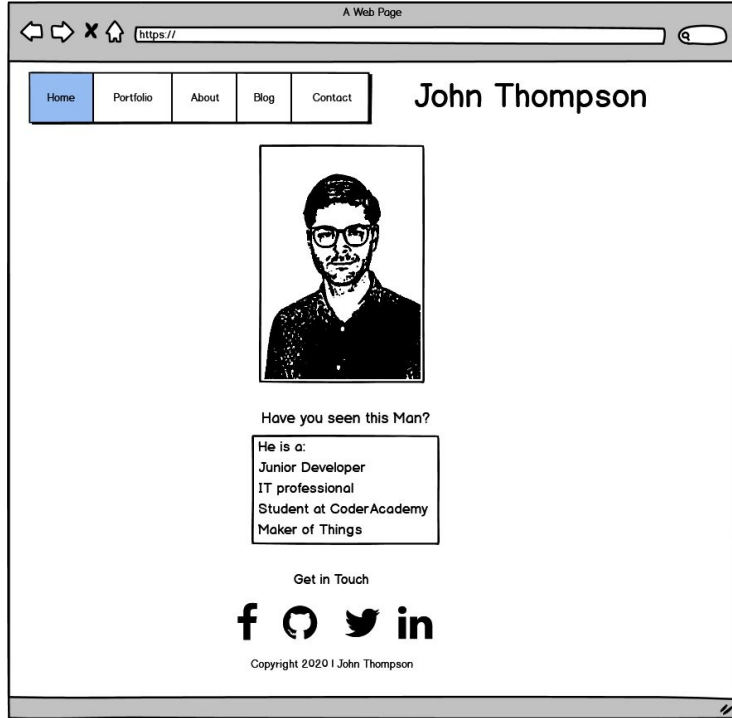
John Thompson



My Projects

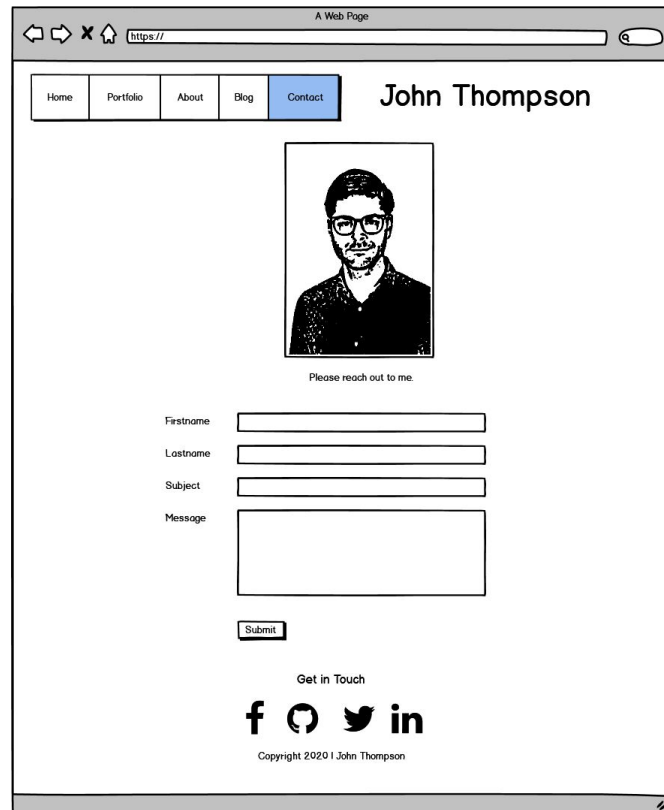
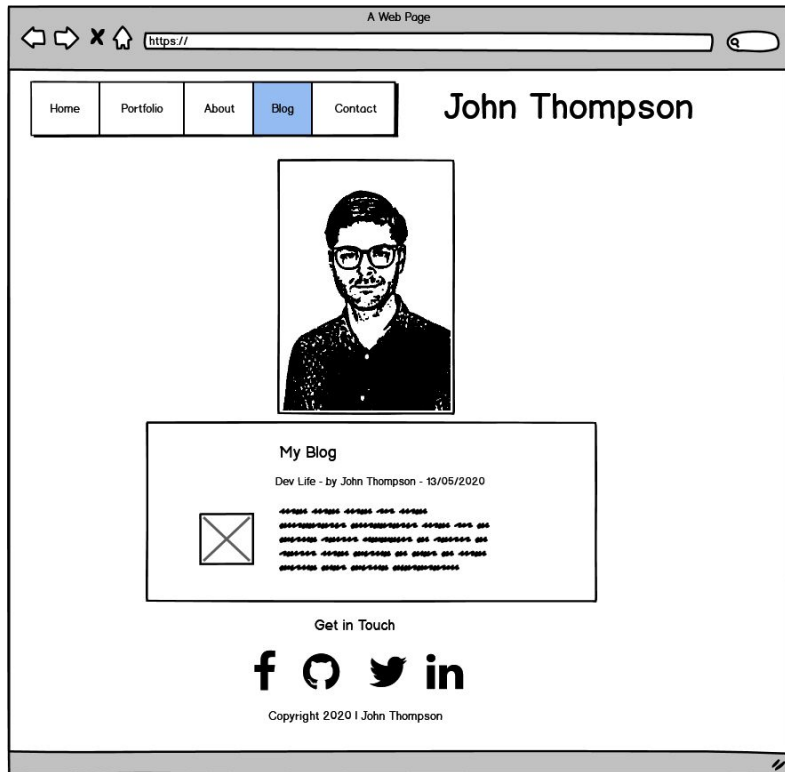


# Wireframes - Desktop

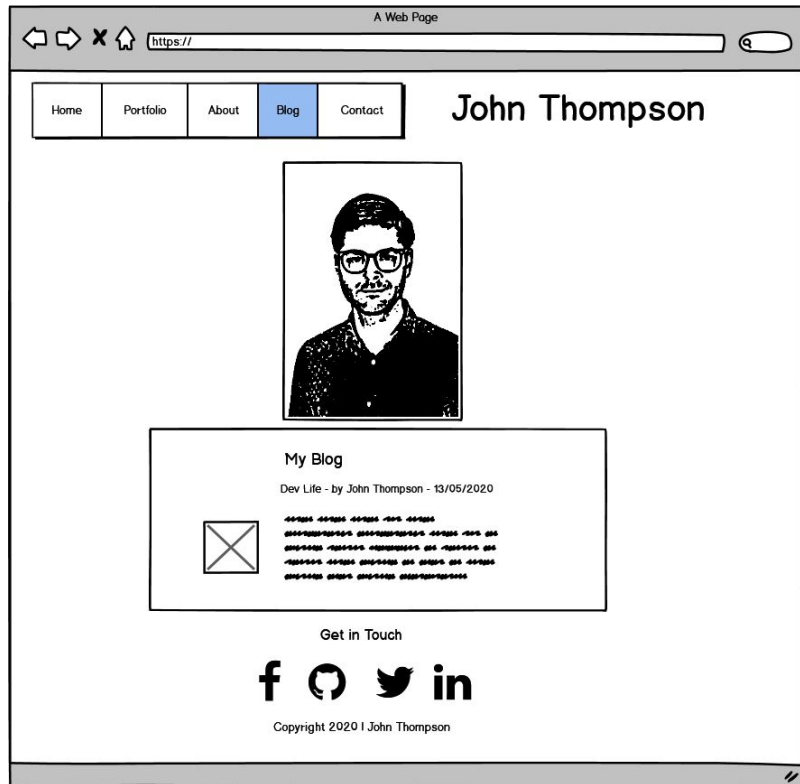




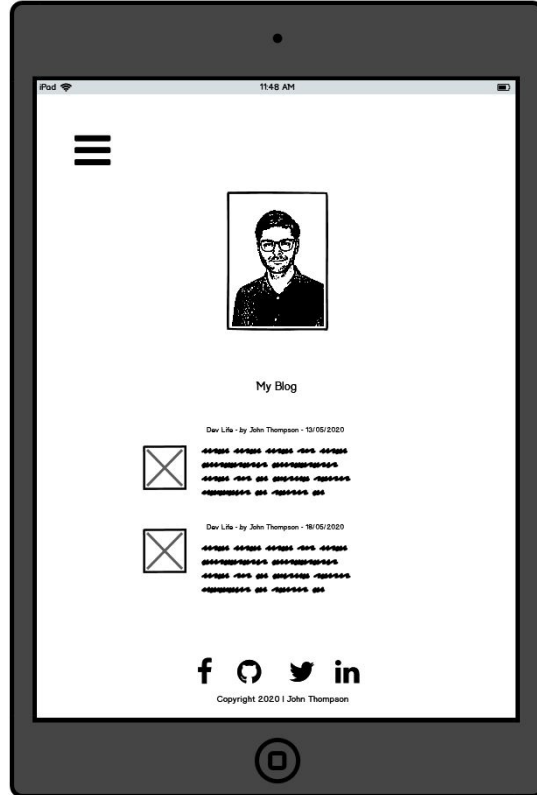
# Wireframes - Desktop



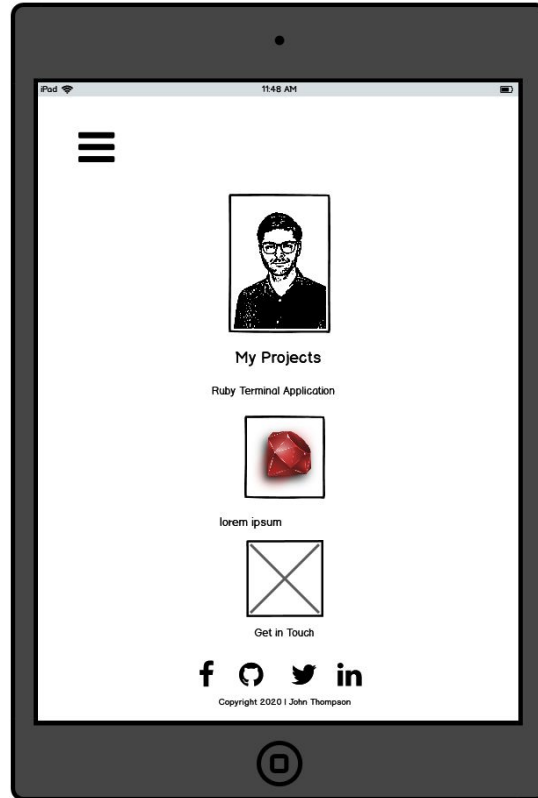
# Wireframes - Desktop



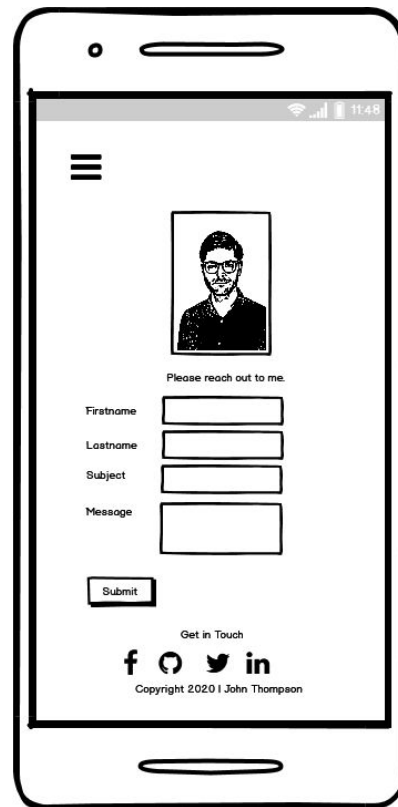
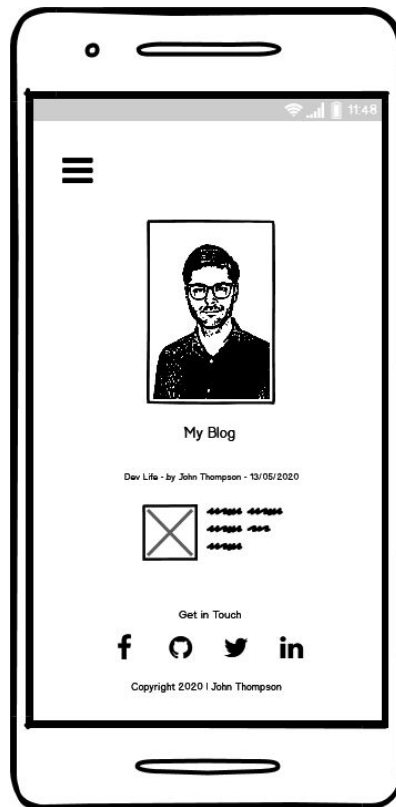
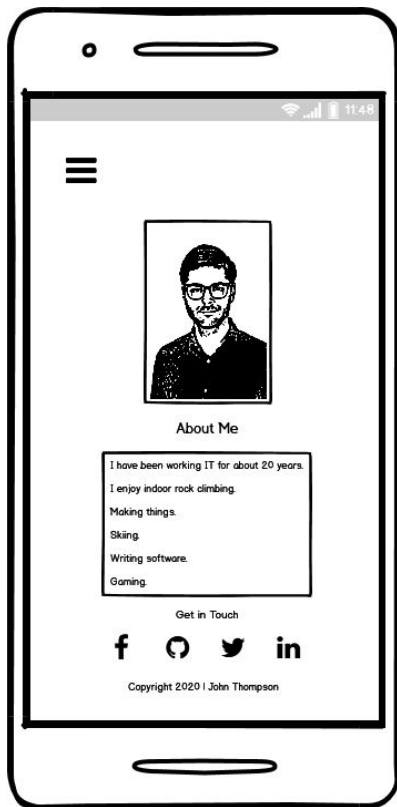
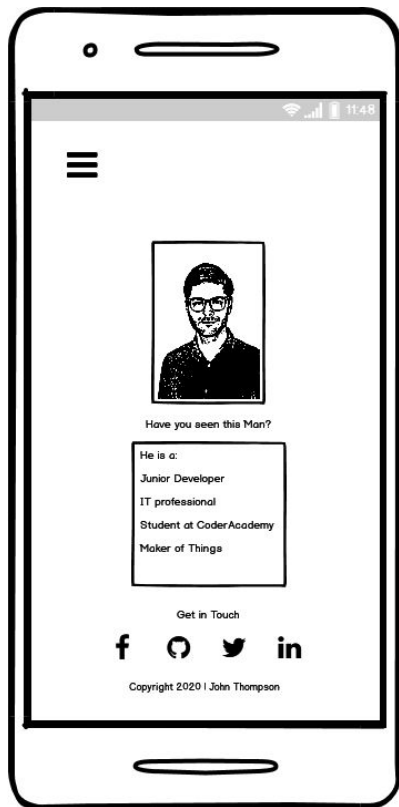
# Wireframes - Ipad view



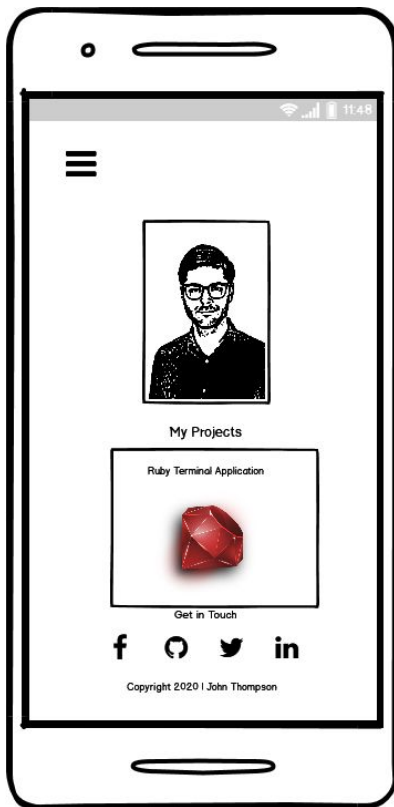
# Wireframes - Ipad view



# Wireframes - Mobile



# Wireframes - Mobile



# Components Used

- HTML
- CSS
- FORMS
- FONT-AWESOME
- FLEX
- GRID

# Components

I didn't want to have too many images so I tried to use as much css as possible, I think that css gives the site a better and sharper look at all resolutions. which is more engaging to the audience.

- I used CSS for all buttons including the desktop menubar.
- And for the mobile navbar used javascript and css.
- I made all images scale by utilizing the viewwidth property
- Most text uses vw so it scales with the browser resolution.
- I only used two media queries
  - for desktop anything over 769px resolution hides mobile nav bar with: `.mobile-menu {display: none;}`
  - for mobile and ipad anything under 768px hides the desktop nav bar with: `.desktop-menu {display: none;}`

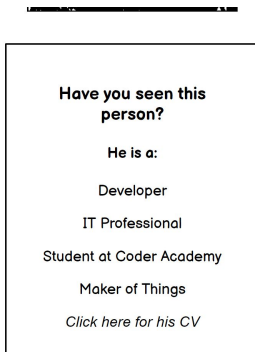


# Components

## #Home-text id

- for most of the text on the site this made it easier to control the text sizes.
- The use of a single box to contain everything I think makes it feel like a piece of paper (cv) which I think is engaging to the audience.

```
#home-text {  
  display: flex;  
  text-align: center;  
  margin-top: 10px;  
  padding: 3vw;  
  border: solid 3px black;  
  border-radius: 2px;  
  width: 30vw;  
  max-width: 500px;  
  min-width: 180px;  
  font-size: 2.5vw;  
}
```



# Components

## #Contact-text

- I had to create a id for the contact form as I had trouble formatting the container to get everything to fit correctly.
- I tried to make the contact box with cleanlines.

```
✓ #contact-text {  
  display: flex;  
  flex-direction: column;  
  flex-flow: row wrap;  
  text-align: center;  
  align-items: center;  
  margin-top: 10px;  
  padding: 3vw;  
  border: solid 3px black;  
  border-radius: 2px;  
  width: 200px;  
  min-width: 40vw;  
  font-size: 2vw;  
}
```

Please reach out to me.

First Name

Last Name

Subject

Message

# Components

## CSS Desktop and Mobile Menus

The desktop menu is a css button group.

The Mobile Menu uses javascript and css.

### Desktop menu css

```
/* menu start - desktop */  
  
.btn-group input[type=button] {  
  background-color: #255; /* Green background */  
  border: 2px solid #000; /* Green border */  
  color: #fff; /* White text */  
  /* padding: 20px 24px; Some padding */  
  padding: 2vw 3vw; /* Some padding */  
  cursor: pointer; /* Pointer/hand icon */  
  float: left; /* Float the buttons side by side */  
  box-shadow: 0.3vw 0.4vw;  
}  
  
/* Clear floats (clearfix hack) */  
.btn-group:after {  
  content: "";  
  clear: both;  
  display: table;  
}  
  
.btn-group input[type=button]:not(:last-child) {  
  border-right: none; /* Prevent double borders */  
}  
  
/* Add a background color on hover */  
.btn-group input[type=button]:hover {  
  background-color: #add8e6;  
}  
  
.btn-group input[type=button]:focus {  
  background-color: #add8e6;  
}  
  
.desktop-menu {  
  margin-left: 20px;  
  margin-top: 30px;  
}  
  
/* menu end - desktop */
```

### Mobile menu css

```
.overlay {  
  height: 100%;  
  width: 0;  
  position: fixed;  
  z-index: 1;  
  top: 0;  
  left: 0;  
  background-color: #000; /* Black background */  
  background-color: rgba(0,0,0, 0.9);  
  overflow-x: hidden;  
  transition: 0.5s;  
}  
  
.overlay-content {  
  position: relative;  
  top: 25%;  
  width: 100%;  
  text-align: center;  
  margin-top: 30px;  
}  
  
.overlay a {  
  padding: 8px;  
  text-decoration: none;  
  font-size: 36px;  
  color: #818181;  
  display: block;  
  transition: 0.3s;  
}  
  
.overlay a:hover, .overlay a:focus {  
  color: #fff;  
}  
  
.overlay .closebtn {  
  position: absolute;  
  top: 20px;  
  right: 45px;  
  font-size: 60px;  
}  
  
.mobile-menu {  
  margin-top: 30px;  
  margin-left: 20px;  
}
```

# Components

## Desktop and Mobile Menu Html

```
<!-- mobile device menu's -->
<nav>
  <div class="mobile-menu">
    <div id="mobile-Nav" class="overlay">
      <a href="javascript:void(0)" class="closebtn" onclick="closeNav()">&times;</a>
      <div class="overlay-content">
        <a href="index.html">Home</a>
        <a href="portfolio.html">Portfolio</a>
        <a href="about.html">About</a>
        <a href="blog.html">Blog</a>
        <a href="contact.html">Contact</a>
      </div>
    </div>
  </div>

  <span style="font-size:30px;cursor:pointer;margin-left:20px;margin-top:20px;" onclick="openNav()">&#9776;</span>
</div>

<!-- mobile device menu's -->

<!-- desktop menu -->
<div class="desktop-menu">
  <div class="btn-group">
    <input type="button" value="Home" onClick="javascript:location.href = 'index.html';" />
    <input type="button" value="Portfolio" onClick="javascript:location.href = 'portfolio.html';" />
    <input type="button" value="About" onClick="javascript:location.href = 'about.html';" />
    <input type="button" value="Blog" onClick="javascript:location.href = 'blog.html';" />
    <input type="button" value="Contact" onClick="javascript:location.href = 'contact.html';" />
  </div>
</div>

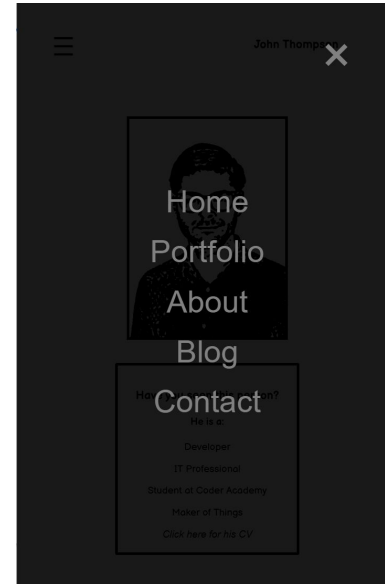
<!-- desktop menu -->
<div id="header-name"><h1>John Thompson</h1></div>
</nav>
```

I used these menus so they would make engage the audience at any resolution.

## Desktop Menu

Home	Portfolio	About	Blog	Contact
------	-----------	-------	------	---------

## Mobile and Ipad Menu



# Components

Javascript for the menu.

It uses the DOM to change the id mobile-Nav style selector for width.

```
<script>
  //    script for mobile device menu
  function openNav() {
    document.getElementById("mobile-Nav").style.width = "100%";
  }

  function closeNav() {
    document.getElementById("mobile-Nav").style.width = "0%";
  }
  //    script for mobile device menu

</script>
```

# Components

Flexbox and Grid together why not?

```
nav {  
  grid-area: nav;  
  display: flex;  
  flex-direction: row;  
}
```

# Components

## Font Awesome?

I found font awesome to be really useful when making the footer, because it made the icon for the footer links super sharp and easy to place.

```
<footer>
  <p>Get in Touch</p>
  <span>
    <a href="https://au.linkedin.com/" title="linkedin" aria-label="linkedin"> <i class="fab fa-linkedin-in"></i></a>
    <a href="https://www.facebook.com/" title="facebook" aria-label="facebook"><i class="fab fa-facebook-f"></i></a>
    <a href="https://github.com/" title="github" aria-label="github"><i class="fab fa-github"></i></a>
    <a href="https://twitter.com/" title="twitter" aria-label="twitter"><i class="fab fa-twitter"></i></a>
  </span>
  <p>Copyright 2020 | John Thompson</p>
</footer>
```

**Thank You**