My Portfolio Website

TESSA FABRY

Site Features

PAGES

- Portfolio
- About Me
- Contact
- Blog

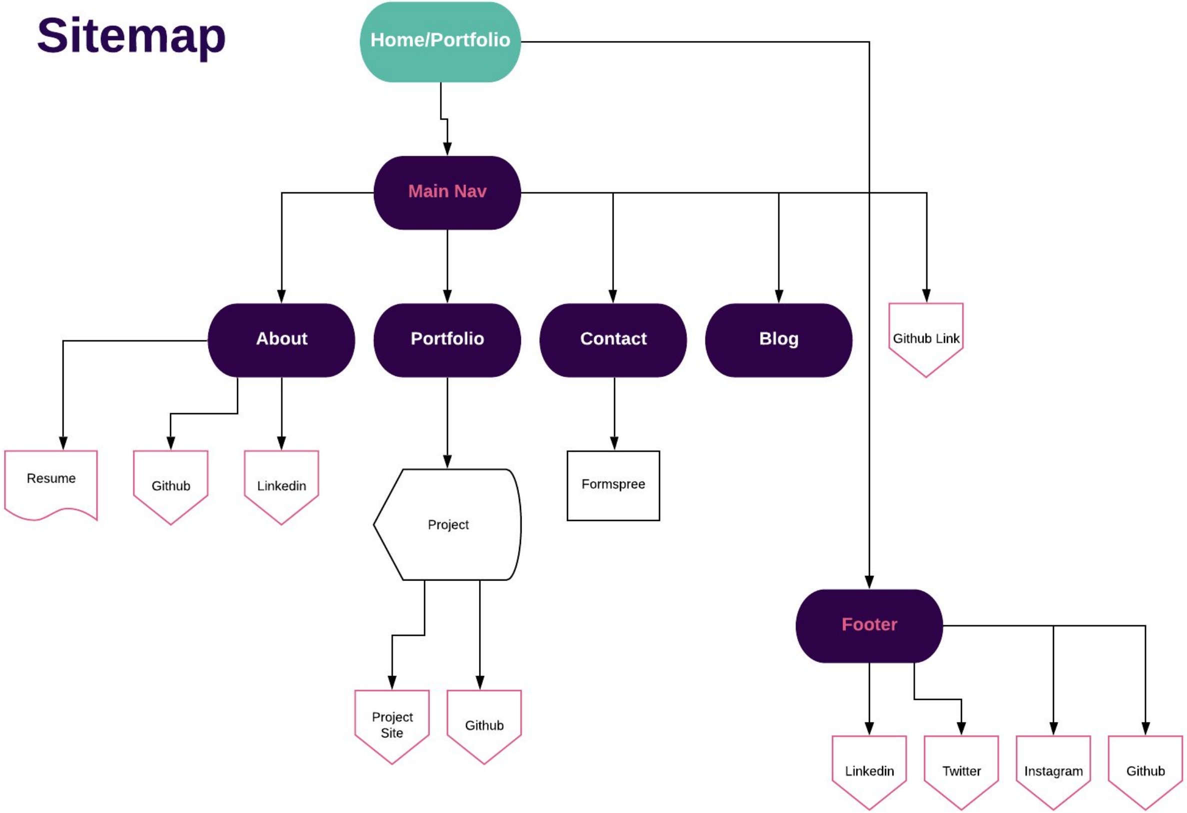
NAVIGATION

- Responsive navbar accessible on all pages

LINKS

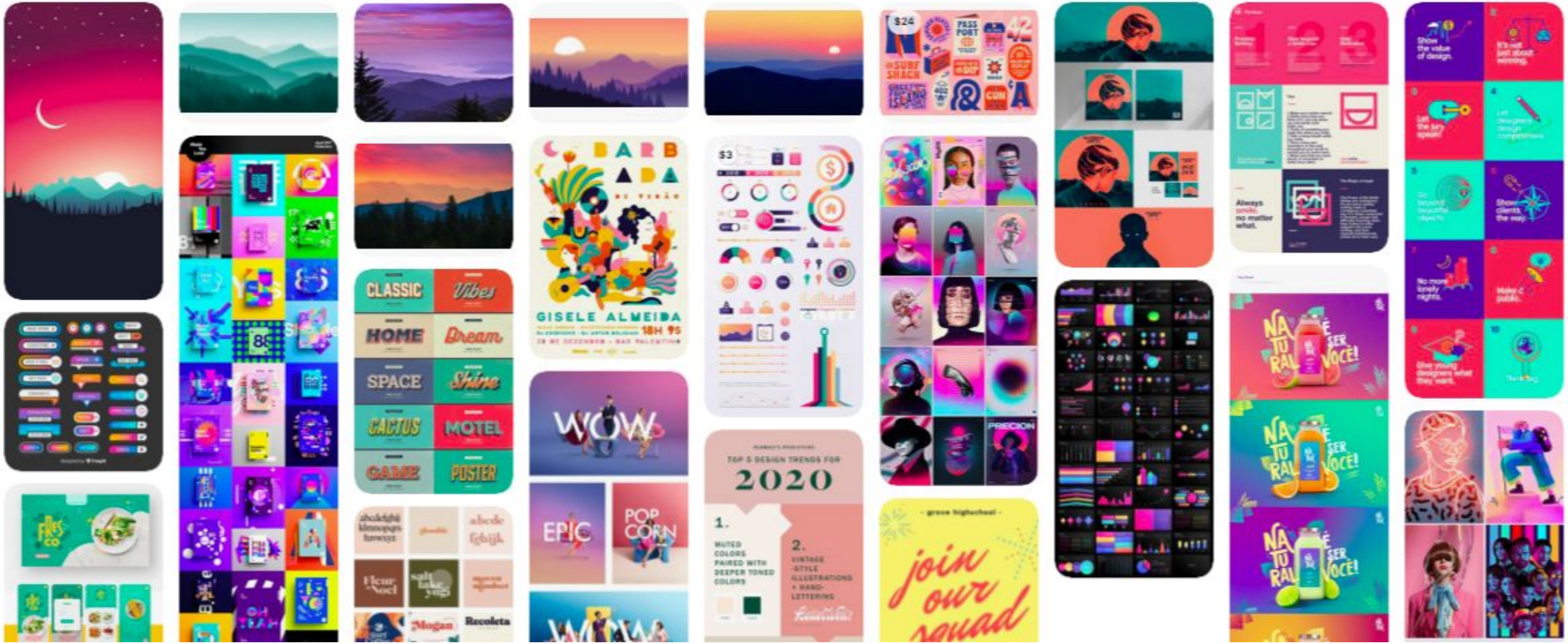
- Github
- Linkedin
- Projects
- CV Download

Sitemap


Sitemap
Home/Portfolio
Main Nav
About
Portfolio
Contact
Blog
Github Link
Resume
Github
Linkedin
Project
Formspree
Project Site
Github
Footer
Linkedin
Twitter
Instagram
Github

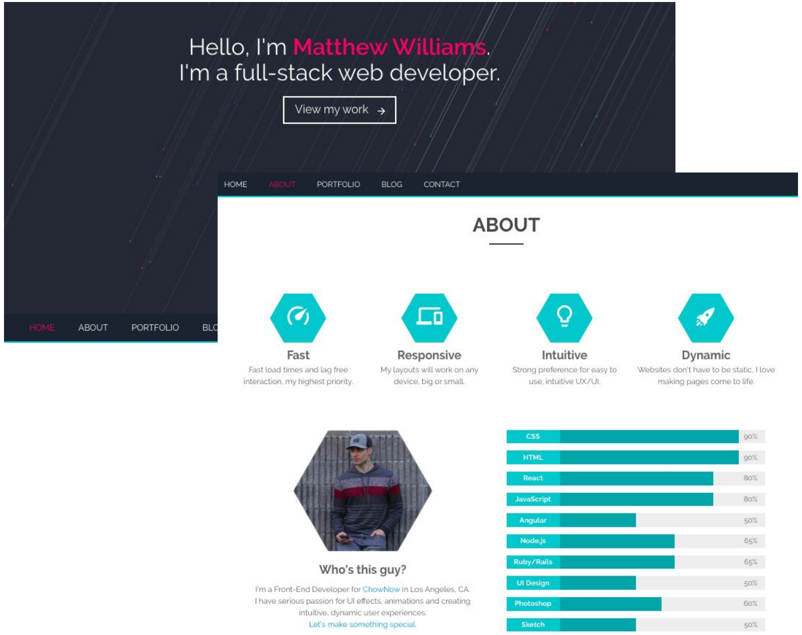
Design Inspiration

MOODBOARD



Design Inspiration

LAYOUT IDEAS



Hello, I'm Matthew Williams.
I'm a full-stack web developer.

View my work →

ABOUT

Fast
Fast load times and lag free interaction, my highest priority.

Responsive
My layouts will work on any device, big or small.

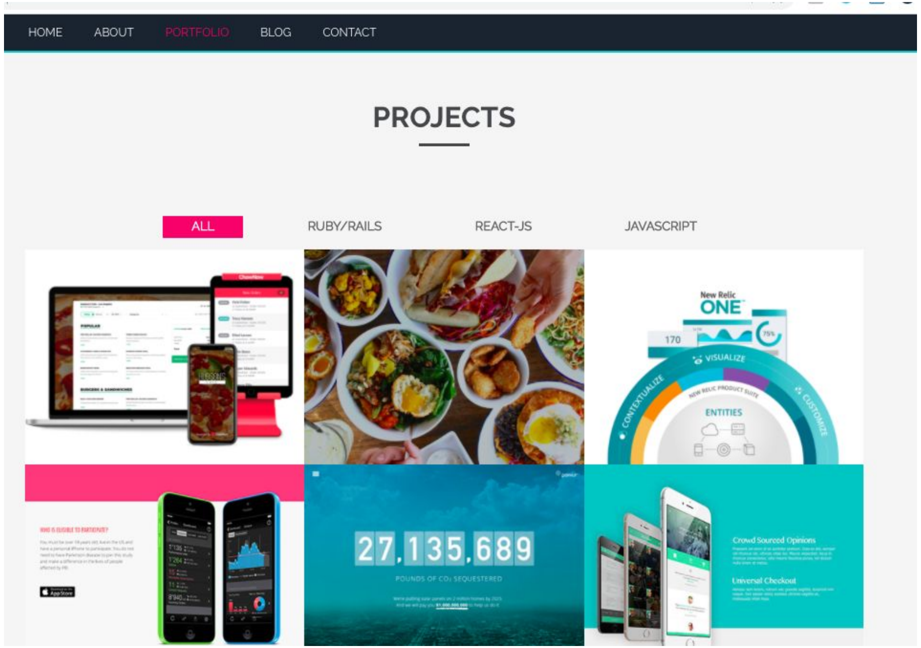
Intuitive
Strong preference for easy to use, intuitive UX/UI.

Dynamic
Websites don't have to be static, I love making pages come to life.

Who's this guy?

I'm a Front-End Developer for ChowNow in Los Angeles, CA. I have serious passion for UI effects, animations and creating intuitive, dynamic user experiences.
Let's make something special.

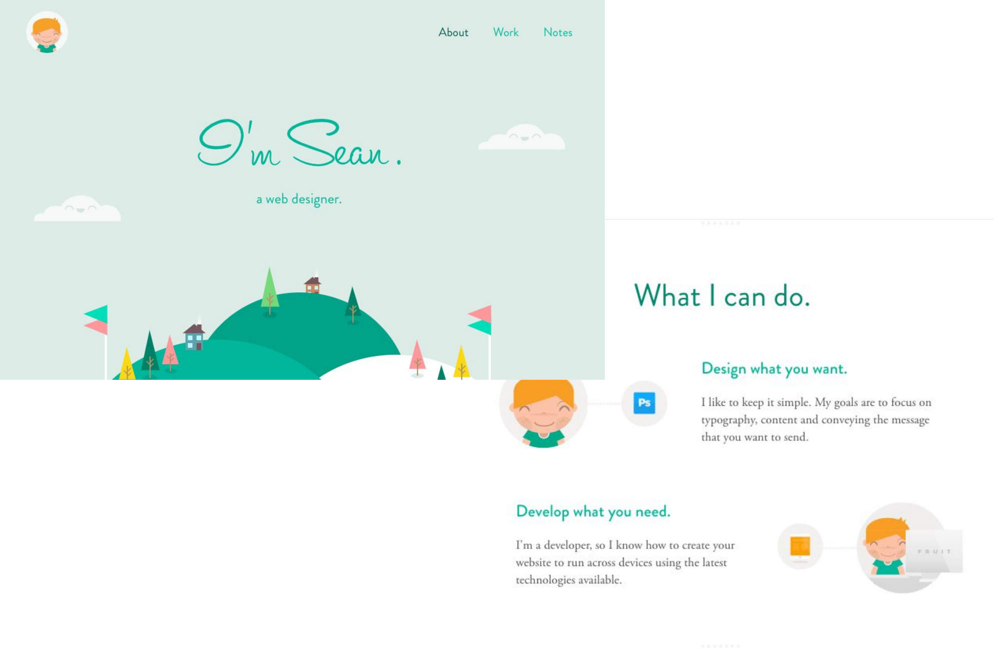
CSS	90%
HTML	90%
React	80%
JavaScript	80%
Angular	50%
Node.js	65%
Ruby/Rails	65%
UI Design	50%
Photoshop	60%
Sketch	50%



PROJECTS

Design Inspiration

LAYOUT IDEAS



About Work Notes

I'm Sean.

a web designer.

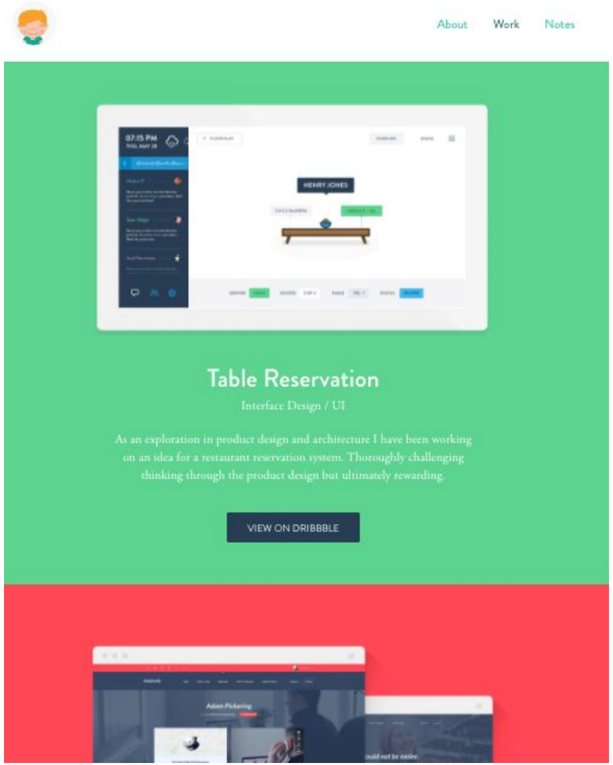
What I can do.

Design what you want.

I like to keep it simple. My goals are to focus on typography, content and conveying the message that you want to send.

Develop what you need.

I'm a developer, so I know how to create your website to run across devices using the latest technologies available.



About Work Notes

Table Reservation

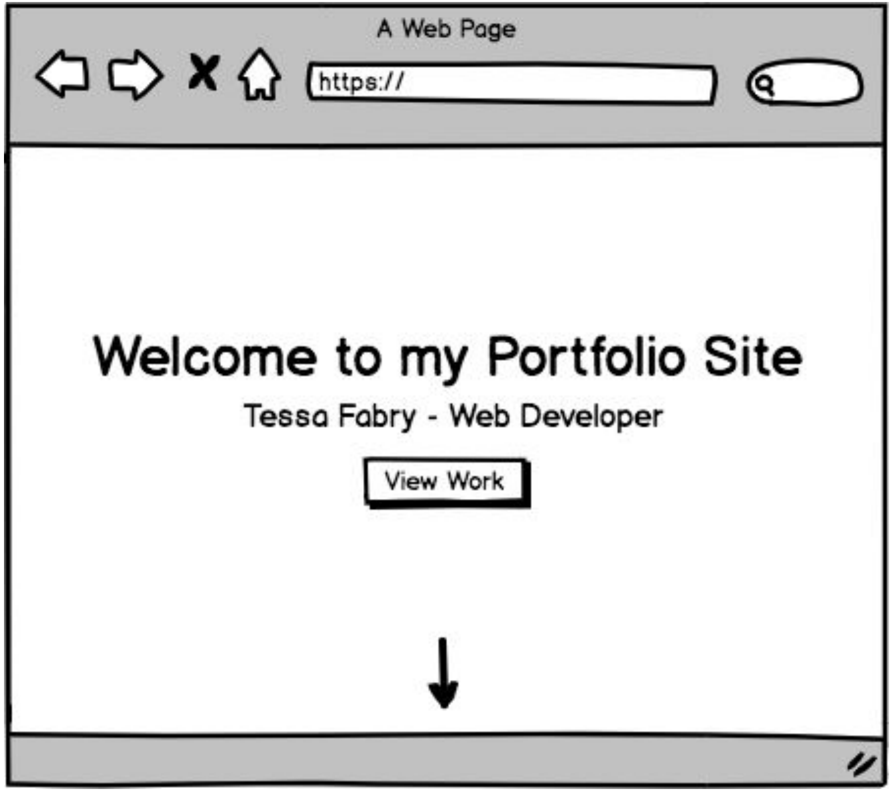
Interface Design / UI

As an exploration in product design and architecture I have been working on an idea for a restaurant reservation system. Thoroughly challenging thinking through the product design but ultimately rewarding.

VIEW ON DRIBBBLE

Wireframes

LANDING

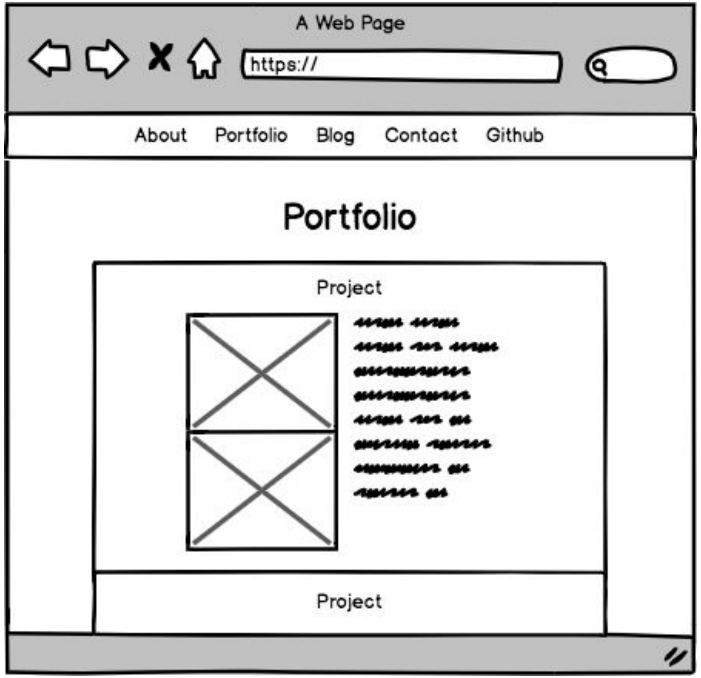


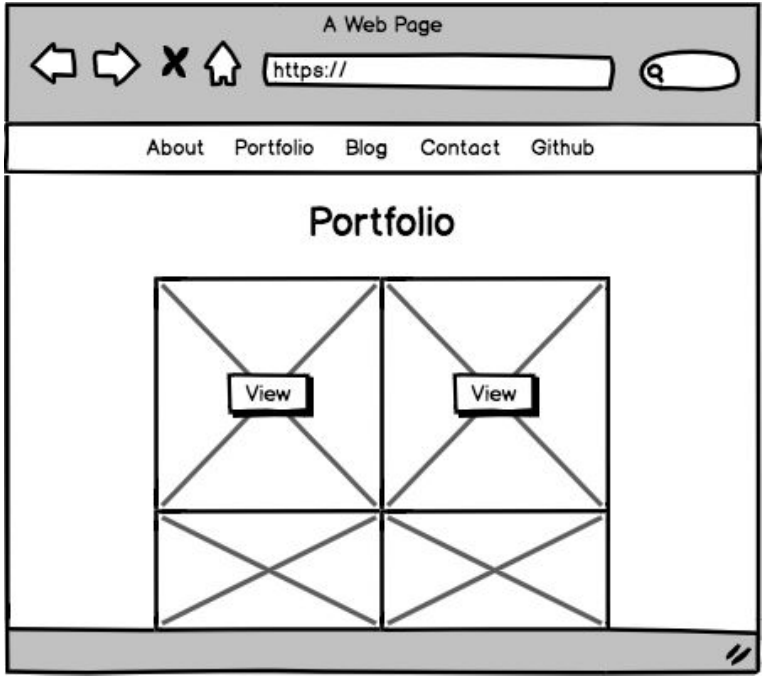
Welcome to my Portfolio Site

Tessa Fabry - Web Developer

Wireframes

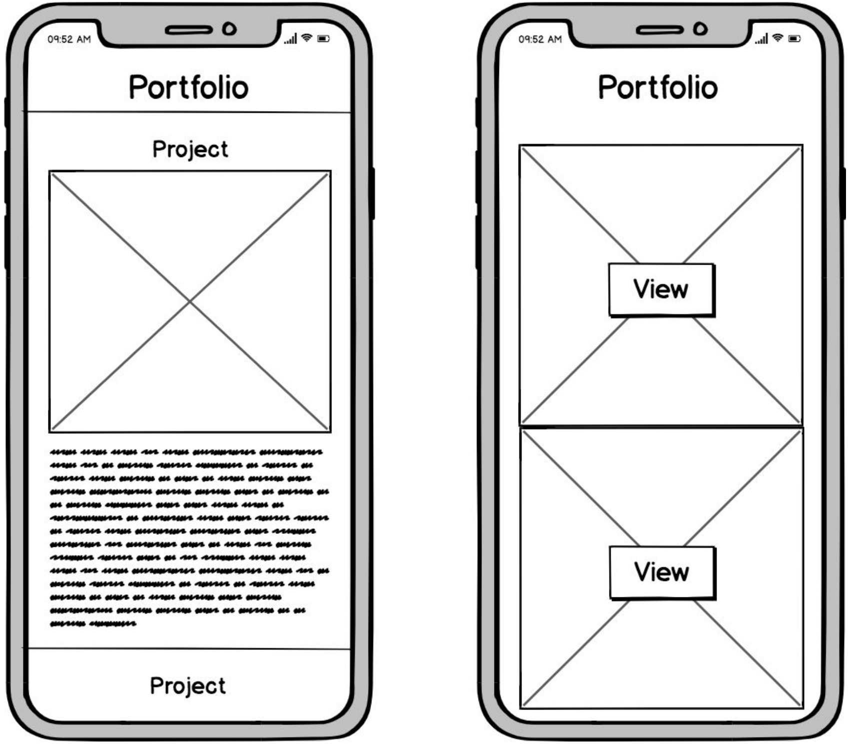
PORTFOLIO





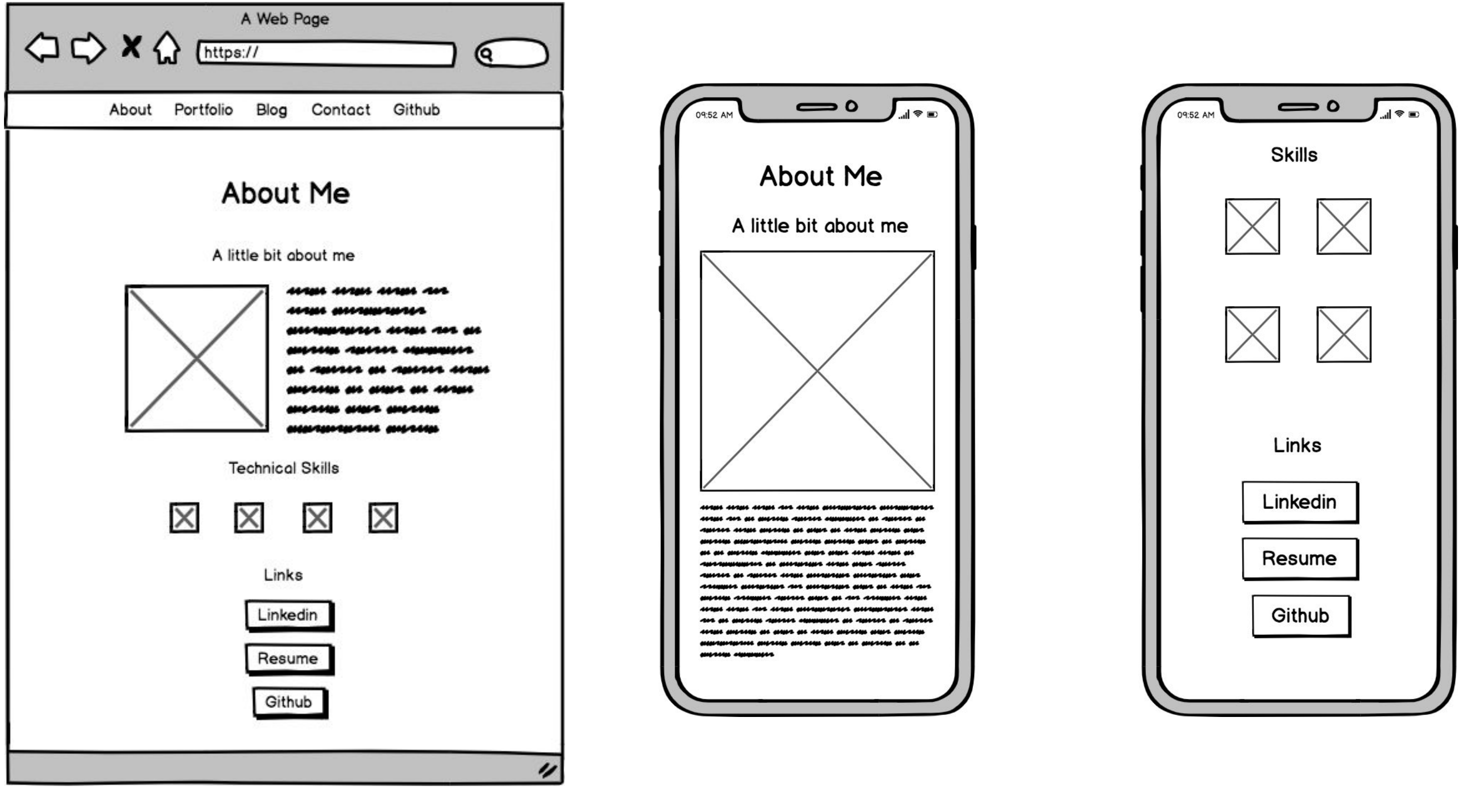
Wireframes

PORTFOLIO - MOBILE



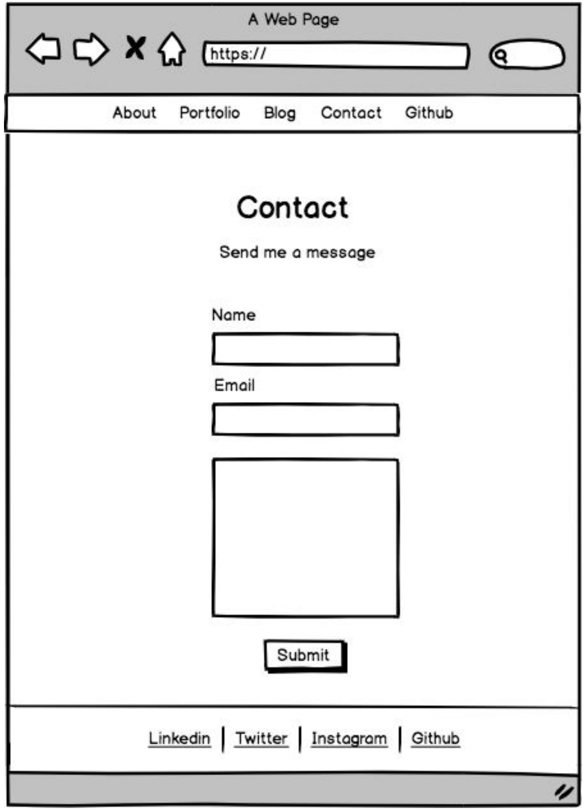
Wireframes

ABOUT

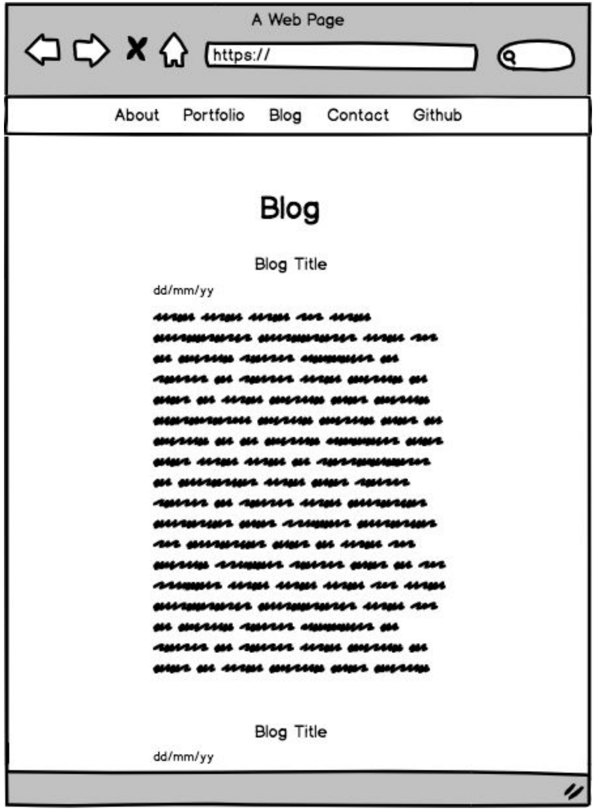


Wireframes

CONTACT



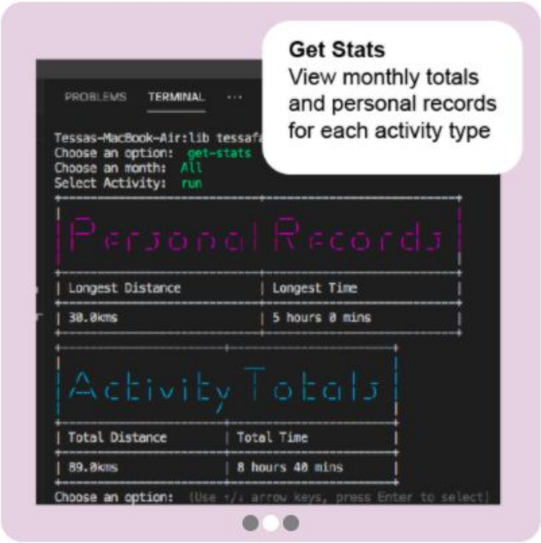
BLOG



Components

PROJECT SLIDESHOW

Each project will display a slideshow of images related to the project.



```
Tessas-MacBook-Air:lib tessaf
Choose an option:  get-stats
Choose an month:  All
Select Activity:  run
+-------------------------------+------------------------------+
|Personal Records                                              |
+-------------------------------+------------------------------+
| Longest Distance              | Longest Time                 |
+-------------------------------+------------------------------+
| 30.0kms                       | 5 hours 0 mins               |
+-------------------------------+------------------------------+
+-------------------------+-------------------------+
|Activity Totals                                    |
+-------------------------+-------------------------+
| Total Distance          | Total Time              |
+-------------------------+-------------------------+
| 89.0kms                 | 8 hours 40 mins         |
+-------------------------+-------------------------+
Choose an option:  (Use ↑/↓ arrow keys, press Enter to select)
```

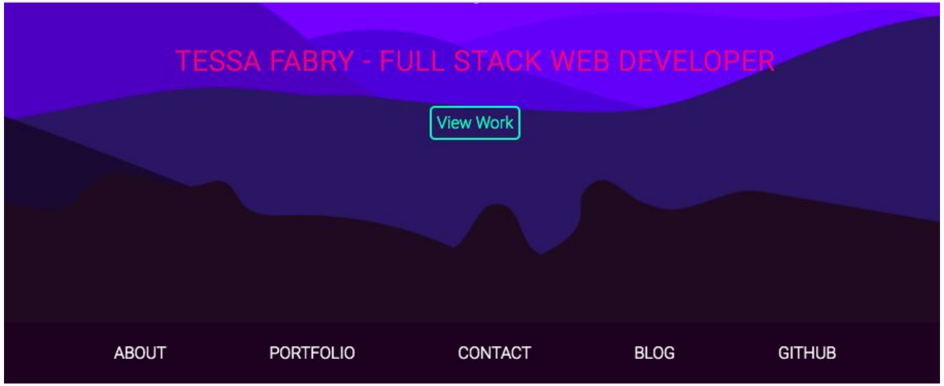
Get Stats
View monthly totals and personal records for each activity type

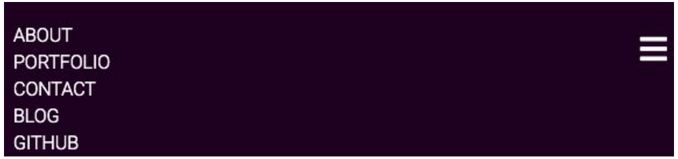
Github

A command line based fitness app written in Ruby. The activity tracker allows users to track exercise progress by recording activities including running, cycling, swimming, walking and hiking. Users can record the time and distance of each workout and use the app to calculate their monthly and overall totals for each exercise type.

Components

RESPONSIVE NAVBAR



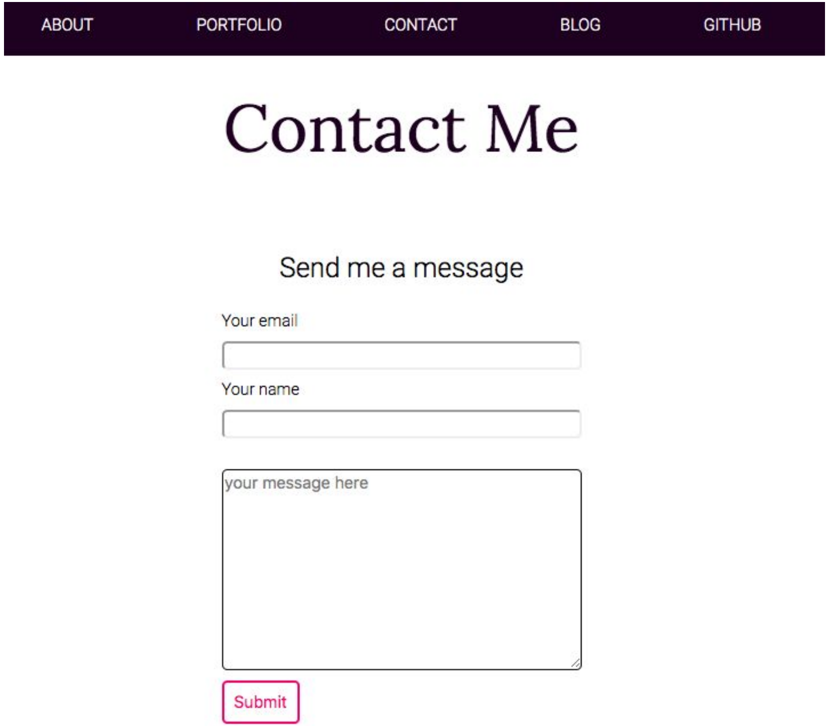


Portfolio

Portfolio

Components

CONTACT FORM



Contact Me

Send me a message

Your email

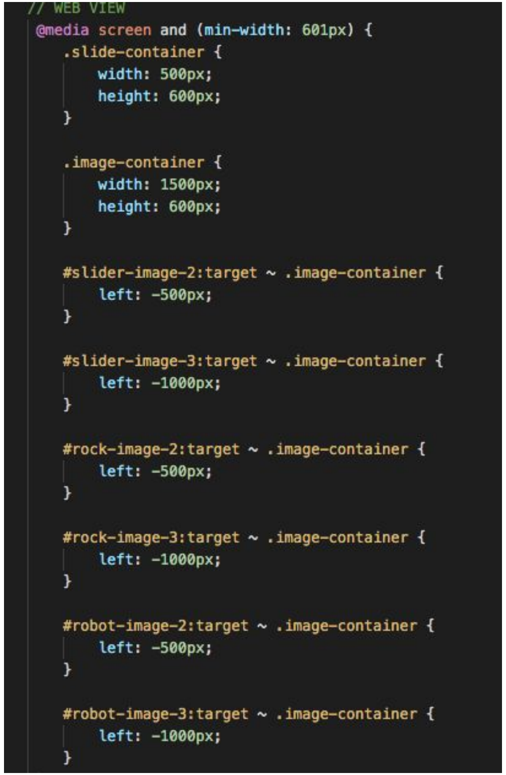
Your name

your message here

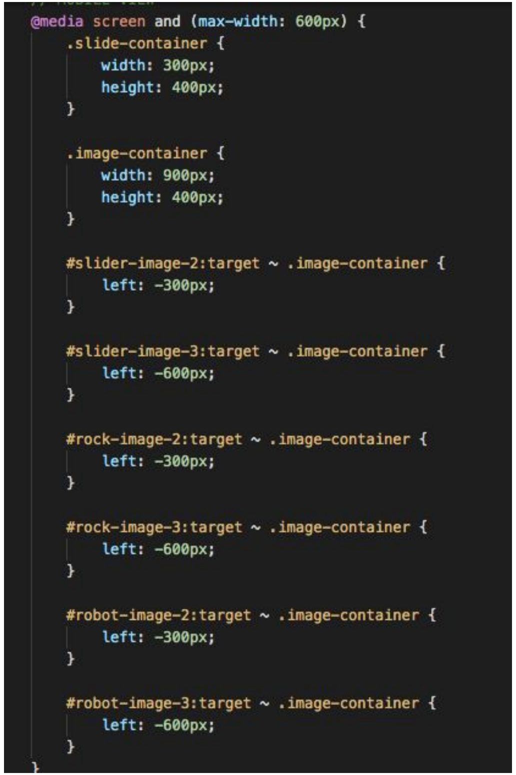
Challenges

SLIDESHOW

- Difficult to implement using only CSS
- Couldn’t implement all of the features I would have liked to include, such as arrows
- Had to change a lot of properties when adjusting size to make responsive



```css
// WEB VIEW
@media screen and (min-width: 601px) {
    .slide-container {
        width: 500px;
        height: 600px;
    }

    .image-container {
        width: 1500px;
        height: 600px;
    }

    #slider-image-2:target ~ .image-container {
        left: -500px;
    }

    #slider-image-3:target ~ .image-container {
        left: -1000px;
    }

    #rock-image-2:target ~ .image-container {
        left: -500px;
    }

    #rock-image-3:target ~ .image-container {
        left: -1000px;
    }

    #robot-image-2:target ~ .image-container {
        left: -500px;
    }

    #robot-image-3:target ~ .image-container {
        left: -1000px;
    }
```

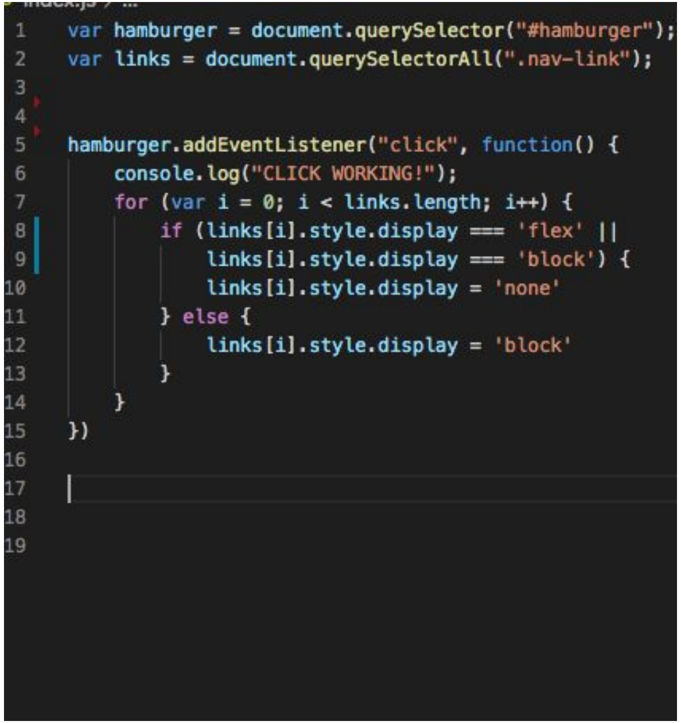


```css
@media screen and (max-width: 600px) {
    .slide-container {
        width: 300px;
        height: 400px;
    }

    .image-container {
        width: 900px;
        height: 400px;
    }

    #slider-image-2:target ~ .image-container {
        left: -300px;
    }

    #slider-image-3:target ~ .image-container {
        left: -600px;
    }

    #rock-image-2:target ~ .image-container {
        left: -300px;
    }

    #rock-image-3:target ~ .image-container {
        left: -600px;
    }

    #robot-image-2:target ~ .image-container {
        left: -300px;
    }

    #robot-image-3:target ~ .image-container {
        left: -600px;
    }
}
```

Challenges

RESPONSIVE NAVBAR

- Ended up having to use Javascript to implement click to show navbar in mobile
- Aligning all the elements within the navbar - with some of them hidden
- Javascript overriding css, had to use !important. Not ideal



```
var hamburger = document.querySelector("#hamburger");
var links = document.querySelectorAll(".nav-link");


hamburger.addEventListener("click", function() {
    console.log("CLICK WORKING!");
    for (var i = 0; i < links.length; i++) {
        if (links[i].style.display === 'flex' ||
            links[i].style.display === 'block') {
            links[i].style.display = 'none'
        } else {
            links[i].style.display = 'block'
        }
    }
})
```