Macy Aviles

UX/UI Designer

Front-end Developer

CSS Style Guide

This living css style guide was created as a general template to show clients and businesses the benefits of using a style guide. The use of style guides allows for semantic and cohesive deisgns and code.

My Role

Position: Project Leader / Front-End Developer

Goal: build a responsive, living css style guide for a client in 4 days

Technology:

Jade	Sass	Express
jQuery	Bourbon	Node.js

Style Guide

v0.0.1

Created by Tyler Anyan & Macy Aviles

MENU

Colors

#ee9966

#88aa66

CSS Style Guide Design Process

In the Code Fellows Bootcamp, my class was divided into teams of two for close collaboration and teamwork. We were to make a basic style guide that would reflect our creativity while keeping the main purpose of a style guide in mind. The time constraint for this project was 4 days.

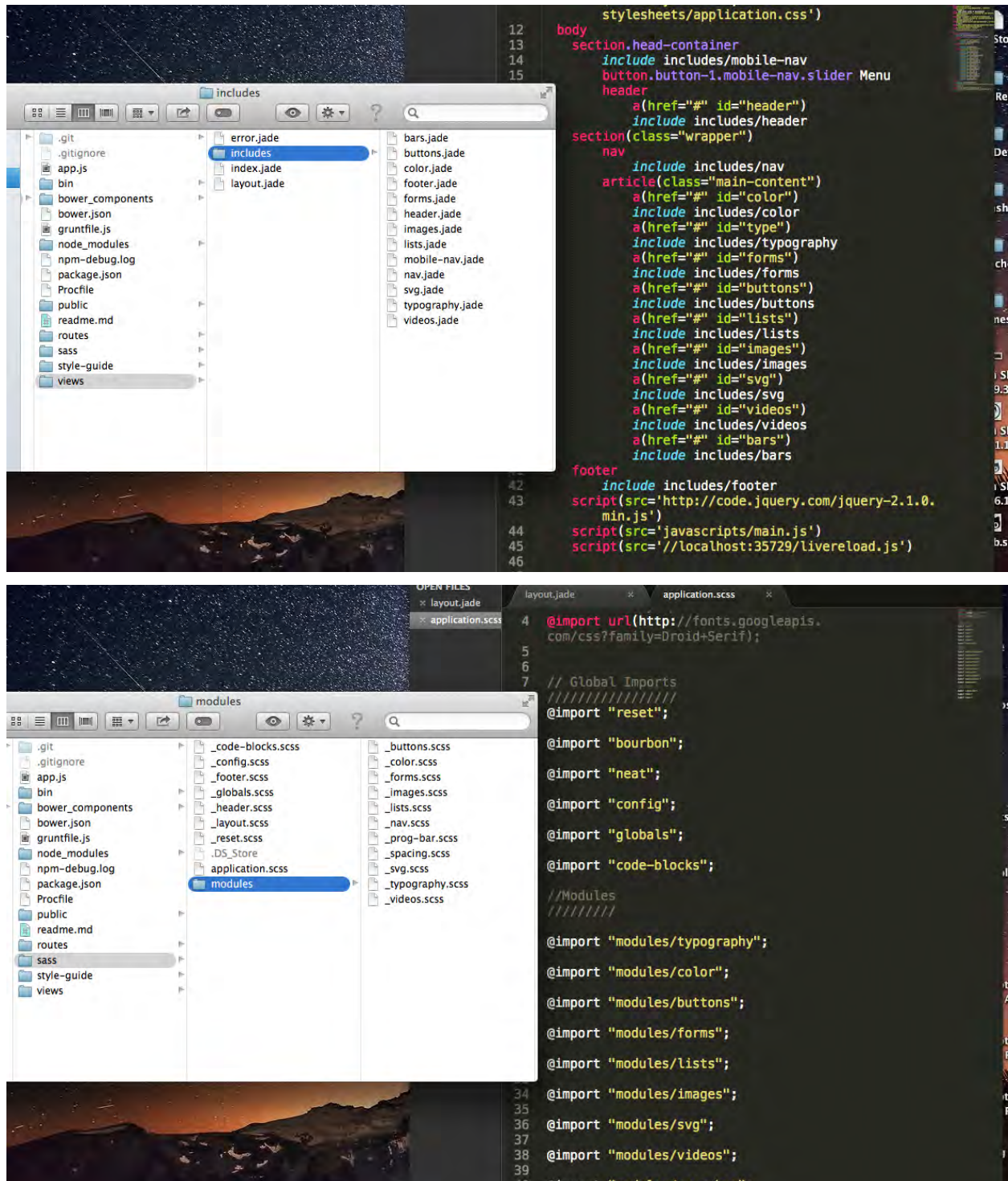We started by listing out the goals we wanted the style guide to achieve:

1. Clean, simple, and to the point.
2. Easy to read and understand the information.
3. Easy to navigate.
4. Responsive across all media so a dev could pull it up on a phone for quick reference or have it displayed on a tablet.

My colleague and I set up a Trello Board to add manageable tasks to keep us on track and to keep communication open. We began populating the board and also using it to brainstorm what kind of content we wanted in our guide.
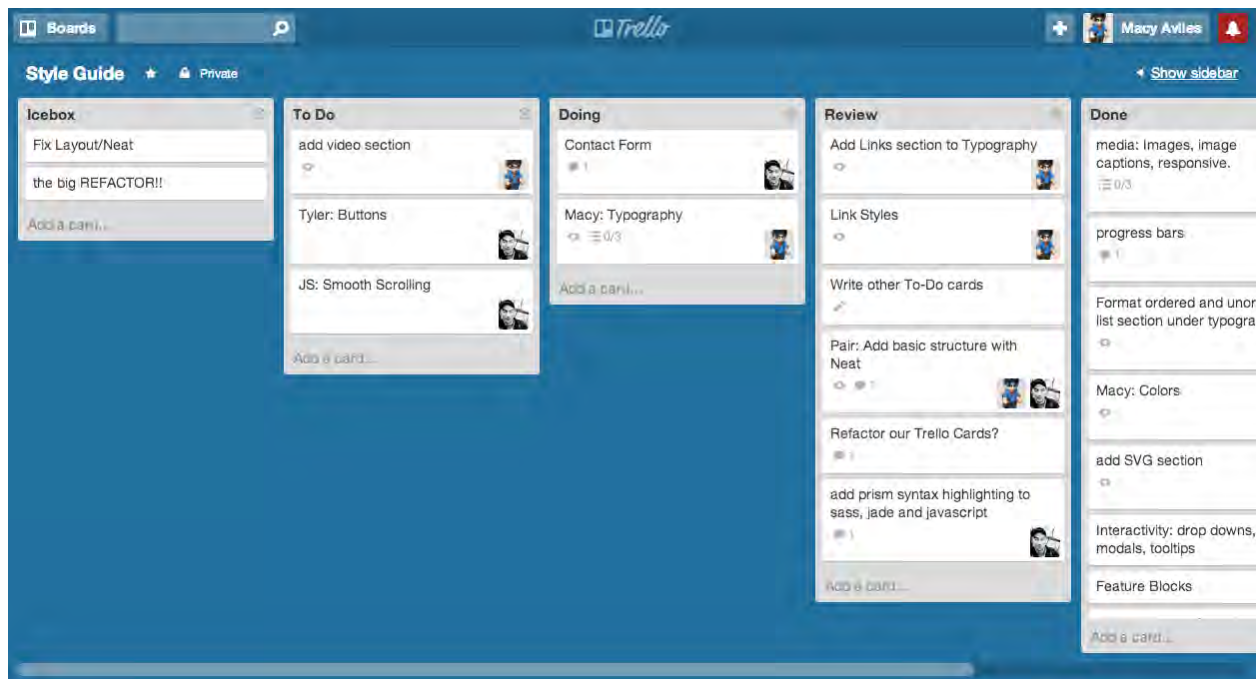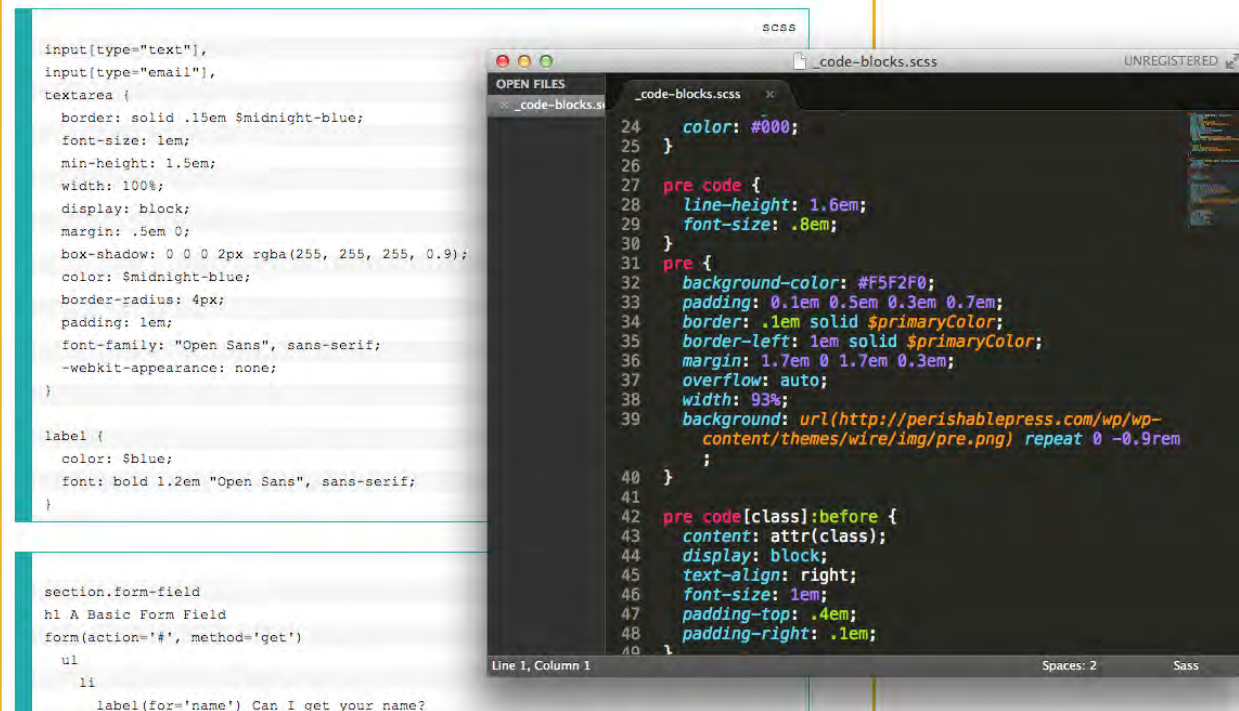

We also set up a file structure that allowed us to template using Jade and Sass. This helped us work in modular environments. We also set up Node.js and Express as our development environment.

```
                                    stylesheets/application.css')
12      body
13        section.head-container
14          include includes/mobile-nav
15          button.button-1.mobile-nav.slider Menu
          header
            a(href="#" id="header")
              include includes/header
        section(class="wrapper")
          nav
              include includes/nav
          article(class="main-content")
            a(href="#" id="color")
              include includes/color
            a(href="#" id="type")
              include includes/typography
            a(href="#" id="forms")
              include includes/forms
            a(href="#" id="buttons")
              include includes/buttons
            a(href="#" id="lists")
              include includes/lists
            a(href="#" id="images")
              include includes/images
            a(href="#" id="svg")
              include includes/svg
            a(href="#" id="videos")
              include includes/videos
            a(href="#" id="bars")
              include includes/bars
          footer
42            include includes/footer
43        script(src='http://code.jquery.com/jquery-2.1.0.
              min.js')
44        script(src='javascripts/main.js')
45        script(src='//localhost:35729/livereload.js')
46
```

```
OPEN FILES
  × layout.jade
  × application.scss
4   @import url(http://fonts.googleapis.
    com/css?family=Droid+Serif);
5
6
7   // Global Imports
    ////////////////////
    @import "reset";

    @import "bourbon";

    @import "neat";

    @import "config";

    @import "globals";

    @import "code-blocks";

    //Modules
    //////////
    @import "modules/typography";

    @import "modules/color";

    @import "modules/buttons";

    @import "modules/forms";

    @import "modules/lists";
34  @import "modules/images";
35
36  @import "modules/svg";
37
38  @import "modules/videos";
39
```
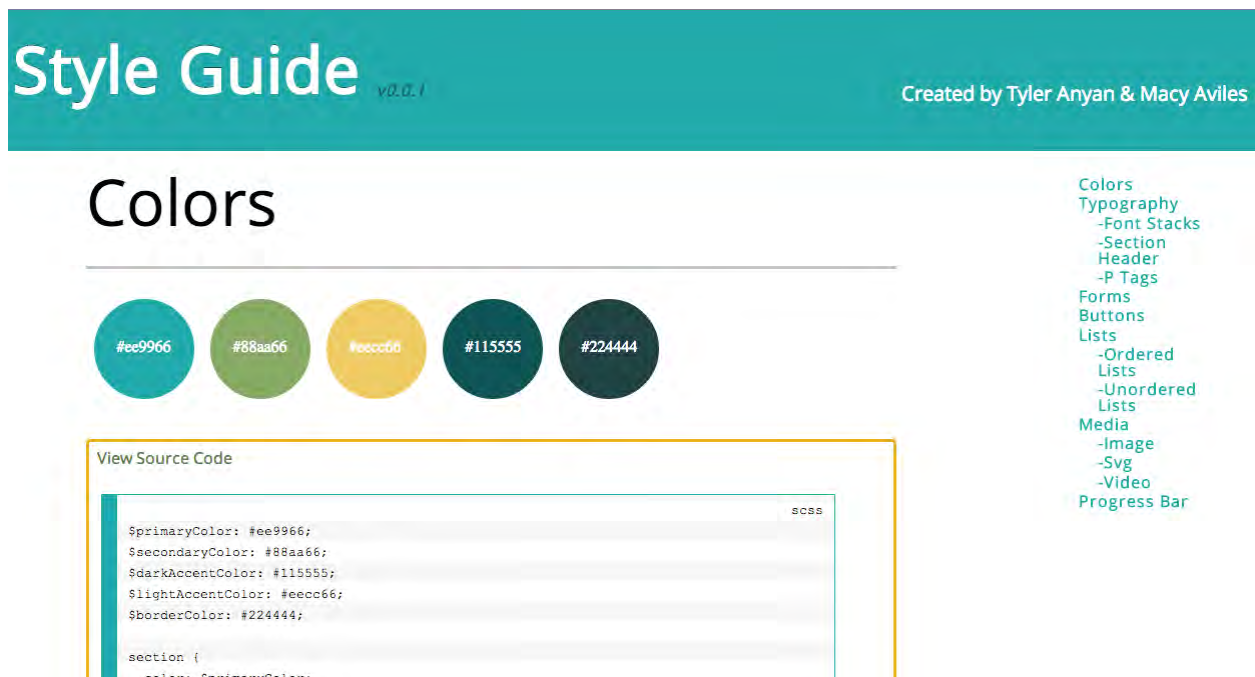
As we were populating our style guide, I searched for different jQuery syntax highlighter plugins to help with our code blocks. We found that many of the plugins did not support highlighting for Jade. Rather than keep looking, I made the custom code blocks to allow for easy to read code snippets instead.

```scss
input[type="text"],
input[type="email"],
textarea {
  border: solid .15em $midnight-blue;
  font-size: 1em;
  min-height: 1.5em;
  width: 100%;
  display: block;
  margin: .5em 0;
  box-shadow: 0 0 0 2px rgba(255, 255, 255, 0.9);
  color: $midnight-blue;
  border-radius: 4px;
  padding: 1em;
  font-family: "Open Sans", sans-serif;
  -webkit-appearance: none;
}

label {
  color: $blue;
  font: bold 1.2em "Open Sans", sans-serif;
}
```

```
section.form-field
  h1 A Basic Form Field
  form(action='#', method='get')
    ul
      li
        label(for='name') Can I get your name?
```

```scss
                                            scss
                         _code-blocks.scss        UNREGISTERED
OPEN FILES     _code-blocks.scss  ×
  _code-blocks.s
24       color: #000;
25     }
26
27   pre code {
28       line-height: 1.6em;
29       font-size: .8em;
30   }
31   pre {
32       background-color: #F5F2F0;
33       padding: 0.1em 0.5em 0.3em 0.7em;
34       border: .1em solid $primaryColor;
35       border-left: 1em solid $primaryColor;
36       margin: 1.7em 0 1.7em 0.3em;
37       overflow: auto;
38       width: 93%;
39       background: url(http://perishablepress.com/wp/wp-
            content/themes/wire/img/pre.png) repeat 0 -0.9rem
            ;
40   }
41
42   pre code[class]:before {
43       content: attr(class);
44       display: block;
45       text-align: right;
46       font-size: 1em;
47       padding-top: .4em;
48       padding-right: .1em;
49
Line 1, Column 1                          Spaces: 2        Sass
```

Using Jade and Sass, we worked modularly to complete and style each section of the style guide. We then added Javascript to add interactivity. The final product can be seen below.

Style Guide v0.0.1

Created by Tyler Anyan & Macy Aviles

Colors

#ee9966 #88aa66 #eecc66 #115555 #224444

View Source Code

```scss
                                            scss
$primaryColor: #ee9966;
$secondaryColor: #88aa66;
$darkAccentColor: #115555;
$lightAccentColor: #eecc66;
$borderColor: #224444;

section {
  color: $primaryColor;
```

Colors
Typography
  -Font Stacks
  -Section
  Header
  -P Tags
Forms
Buttons
Lists
  -Ordered
  Lists
  -Unordered
  Lists
Media
  -Image
  -Svg
  -Video
Progress Bar

Progress Bar

Here is a simple progress bar, hover over the meter to see the effect.

View Source Code

MENU

Buttons

PRIMARY   SECONDARY   WARNING

SUBMIT

TOGGLE PANEL

View Source Code

```scss
%button {
    padding: 1em;
    border-radius: 4px;
    text-transform: uppercase;
    transition: all .2s ease;
    text-align: center;
    letter-spacing: 1px;
    text-decoration: none;
    font: bold 1em "Open Sans", sans-serif;
    text-decoration: none;
    border: none;
}

%button-slide {
```

MENU

Links

Link 1
Link 2
Link 3
Link 4

View Source Code

```scss
a {
    display: block;
    font-family: 'Open Sans', 'sans-serif';
    font-size: 1em;
    font-weight: bold;
    text-decoration: none;
    color: $primaryColor;
    &:hover {
        color: $primaryHover;
    }
    &:active {
        color: $primaryHover;
    }
}
```

```jade
a(href="#") Link 1
a(href="#") Link 2
```

CSS Style Guide Features

Custom designed Code Blocks: these code blocks were designed with efficiency and simplicity in mind.

alternating shades keeps lines easy to read

just give your code block a class name of the language and the name displays at the top right corner, allowing for ease in adding to future segments
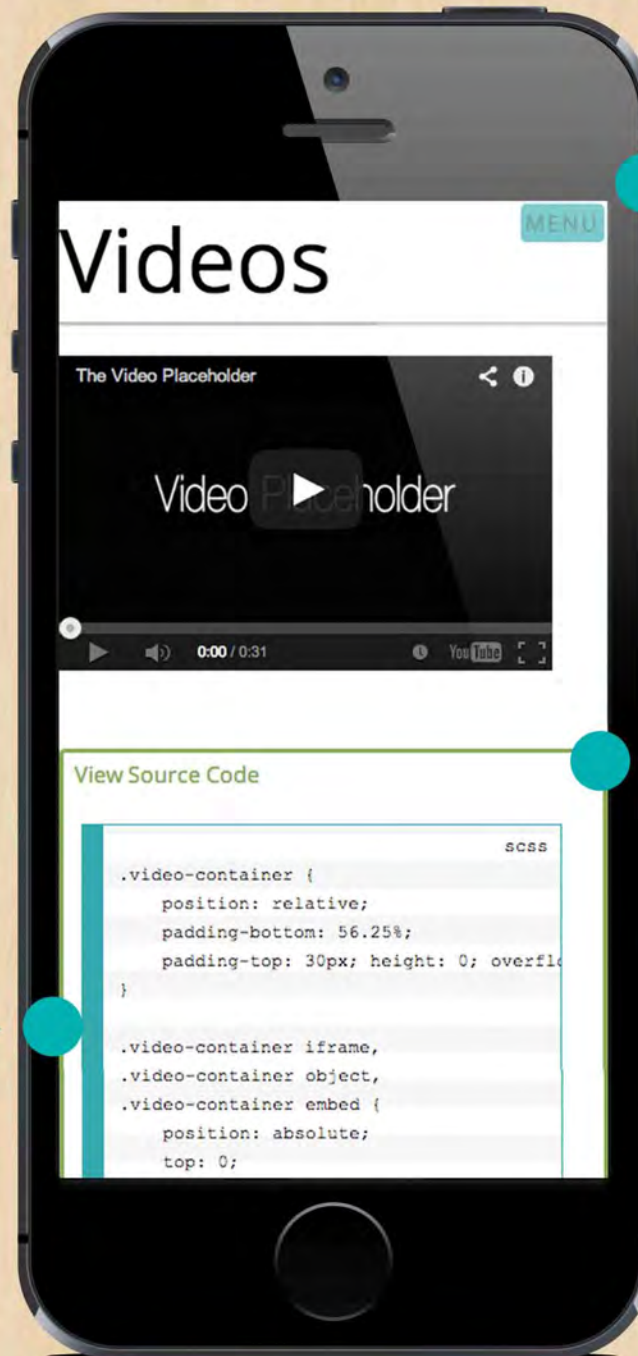
Smooth Scroll Sticky Nav: provides easy navigation and referencing so you can focus on your work

Collapsable Code Blocks: allows for organization and aesthetic functionality

Videos

MENU

The Video Placeholder

Video Placeholder

0:00 / 0:31 You Tube

View Source Code

```scss
.video-container {
    position: relative;
    padding-bottom: 56.25%;
    padding-top: 30px; height: 0; overfl
}

.video-container iframe,
.video-container object,
.video-container embed {
    position: absolute;
    top: 0;
```
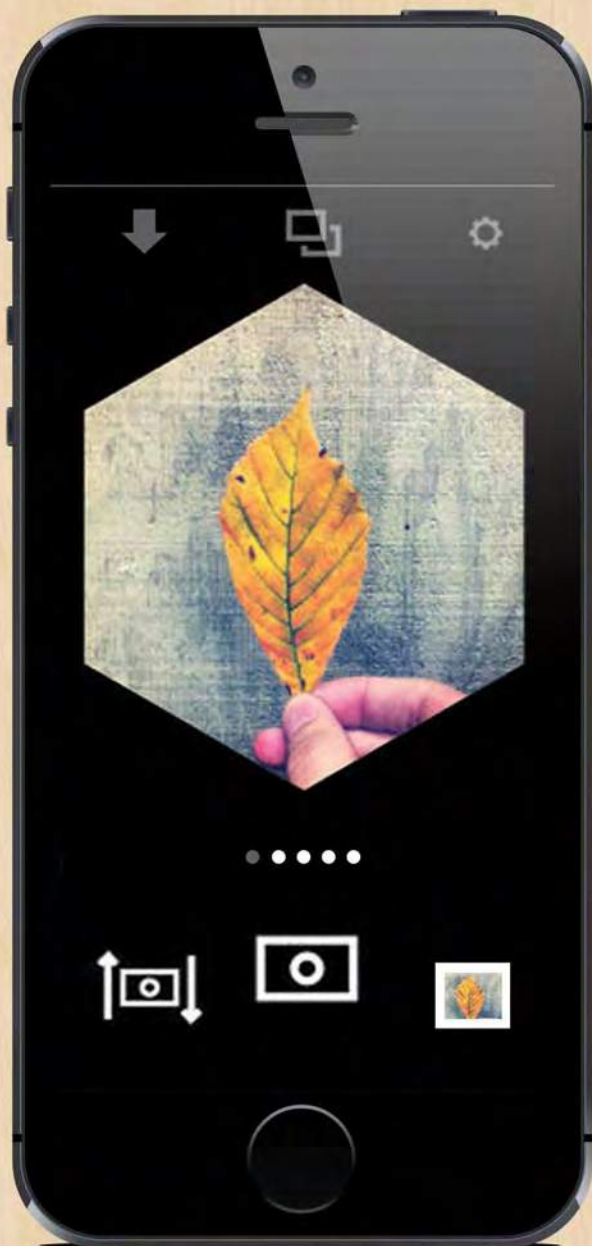
Bezel Photo

With Bezel Photo it's never been easier to make your photography get noticed. You can now get professional looking results with a beautiful, simple interface.

My Role

Position: UX/UI Designer

Goal: redesign Bezel's UI to create a better, fluid user experience

Technology:

Photoshop

POP app

Xcode

Bezel Design Process

Bezel is an iPhone application that allows users to give phone pictures a professional touch. Upon my first meeting with Bezel's developer, I learned that Bezel had not been updated since iOS 5 and needed an overhaul of their UI. Bezel's existing interface was clean and minimalistic, but the experience was clunky and confusing.

My client went over user concerns and new features to be added to the application . The main concern was the application's lack of easy navigation in addition to the difficulty of saving pictures. My client also wanted to emphasize the accessibility of in-app purchases. After going over Bezel's vision, my client requested my work in creating a more fluid user experience while keeping the overall design.

The first phase of this project was to focus on brainstorming, studying existing products and prototyping. I applied my previous knowledge and experience to user fluidity to the current Bezel product and began trying out new angles. Keeping in mind that my client's request was easier user experience, I sectioned off the project into different areas of work.
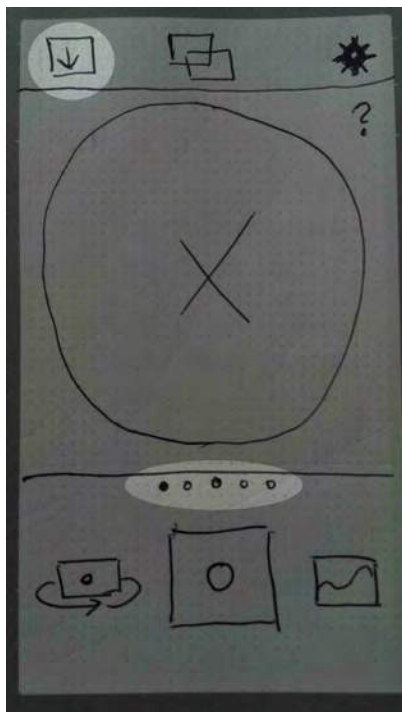
First, I addressed the difficulty of saving pictures. Prior to my work in this project, the top left icon was used for in-app purchases. Based on my studies on existing products, I repurposed the top left icon to saving pictures to the camera roll. This allows the user to save the photo anytime: after the capture, while applying the professional touches, before and after sharing. Next, I addressed the users' concerns of using options not seen on the main screen. In the original application, other options were not easily accessible and mostly hidden from use. Rather than adding more icons, I augmented the problem by adding pagination to the bottom menu bar. This allows the user to see other options in addition to seeing their position in the flow.

Previously, Bezel's top left icon was for in-app purchases. Since I repurposed the icon to focus only on saving pictures, I moved the in-app purchases option to the user flow bar. This option was added to the end of the pagination so that users know there were more features and options after they've gone through the steps.

In addition, this option was added to the bottom of the settings menu. Having the purchase option in two different locations is key for Bezel's future profitability while giving users the opportunity to expand the app.
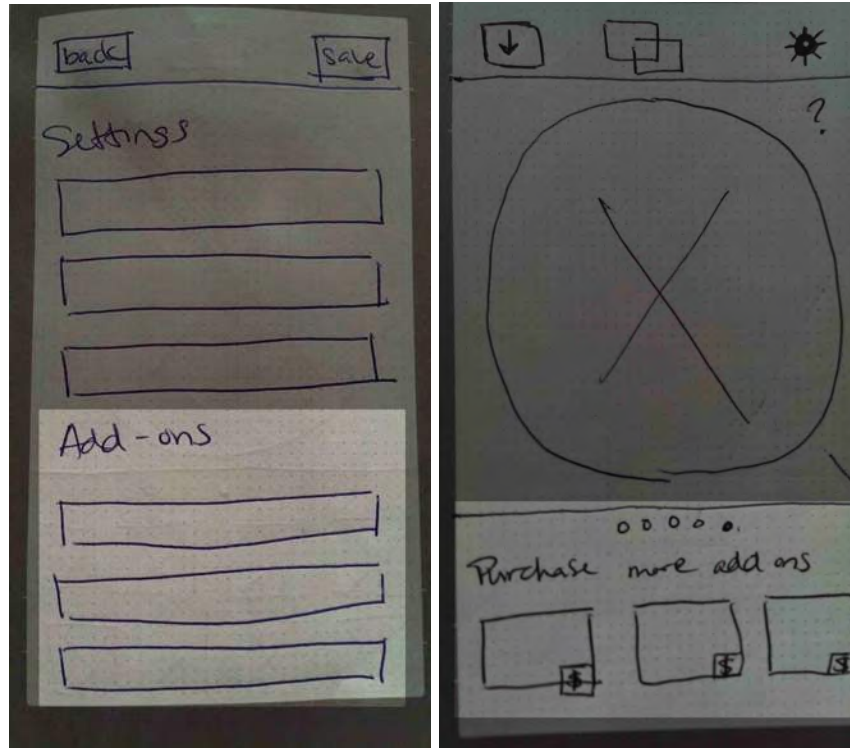
With the main user concerns addressed, I rearranged and adjusted the app to solidify easy user experience. I brought my wireframes together and constructed a clickable prototype using POP app, a mobile prototyping application for smartphones. I had several users test my prototype and give me feedback. Once I felt like my client's requests were fully addressed, I arranged another meeting with Bezel's developer. My client was very satisfied with the changes I made and now plans to implement them with the next iOS 7 update.

Bezel Wireframes
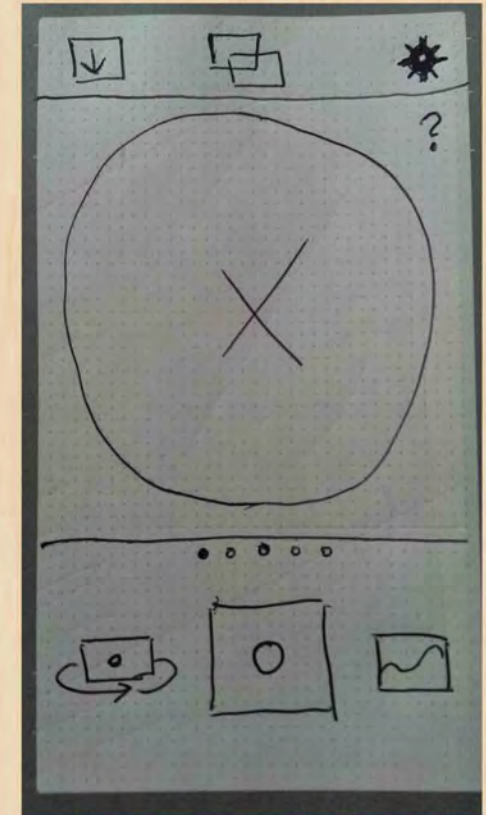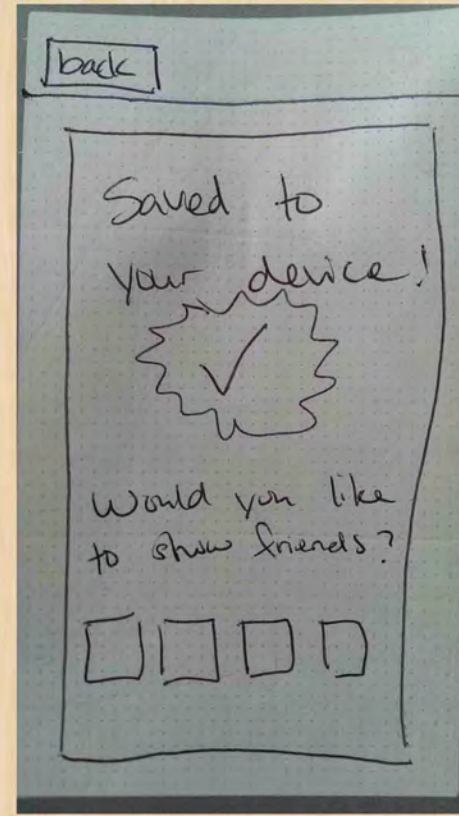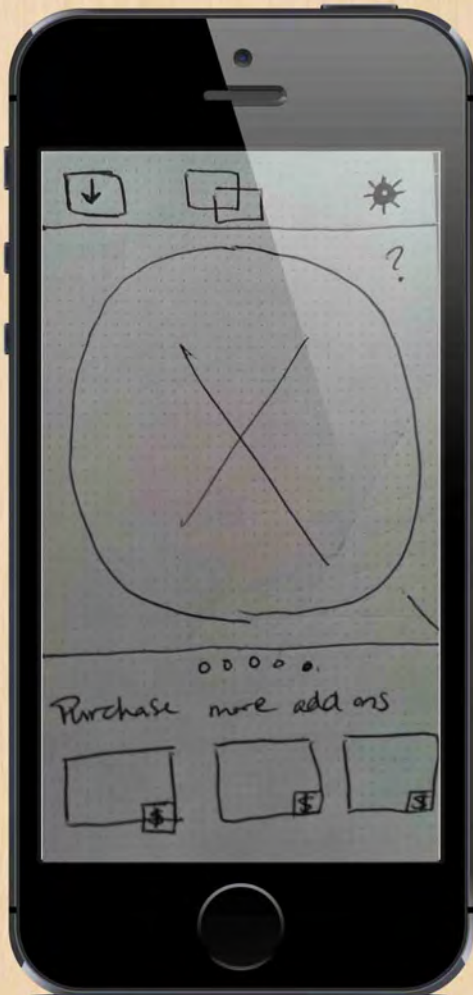
Here are some of the wireframes used to develop the prototype

Voyager

Voyager is a user generated exploration tool that allows users to find and explore new and fun points of interests in a city. Whether someone is visiting from out of town or a local who just wants to find something to do, Voyager gives every users a chance to explore.

My Role

Position: Project Leader / Front-End Developer

Goal: build a responsive, living web application prototype in 4 days

Technology:

HTML5   CSS3

jQuery   Skeleton

Voyager

Go on a hidden landmark tour, *crafted by a local*

Search by city

How It Works
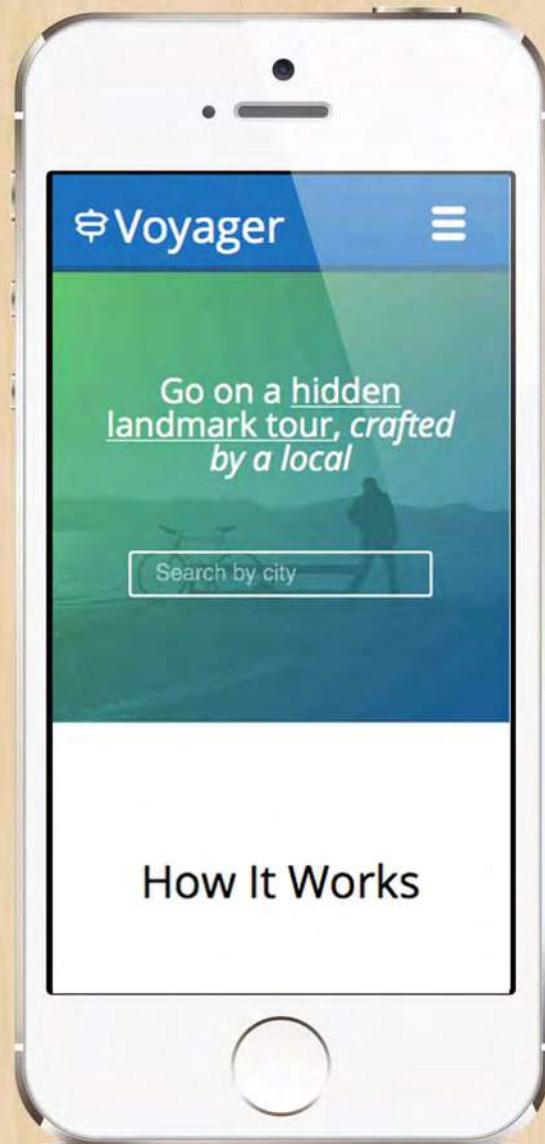
Voyager Design Process

During the Code Fellows UX Engineering bootcamp I participated in, I was assigned to a team of three to build out a prototype web app in four days. After a deep brainstorming session, we settled on Voyager, a web app that allows for user-generated expeditions of cities.

Voyager was inspired by a weekend trip in Wenatchee, WA. During my stay, I found many off-beaten, off-the-main-road drives that were surrounded by the most beautiful trees, orchards, and mountains. I thought about how my trip could be shared with others who also sought out less traveled roads. That's where Voyager comes in. By allowing locals to curate expeditions, a visitor could have the chance to see unique and interesting spots around town quickly and easily.

As a result of the time constraint, my team decided to build out the three core pages to a site: a homepage, a city listings page, and an expedition details page.
We began by sketching out wireframes, utilizing a mobile-first strategy and adding content as the window expanded.
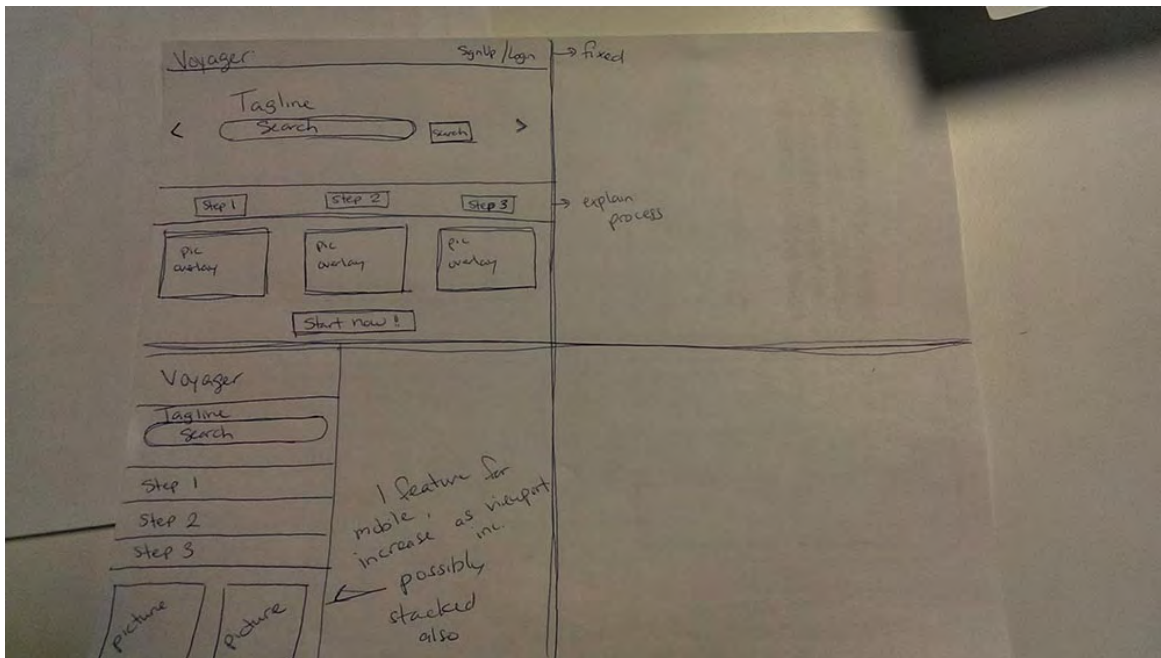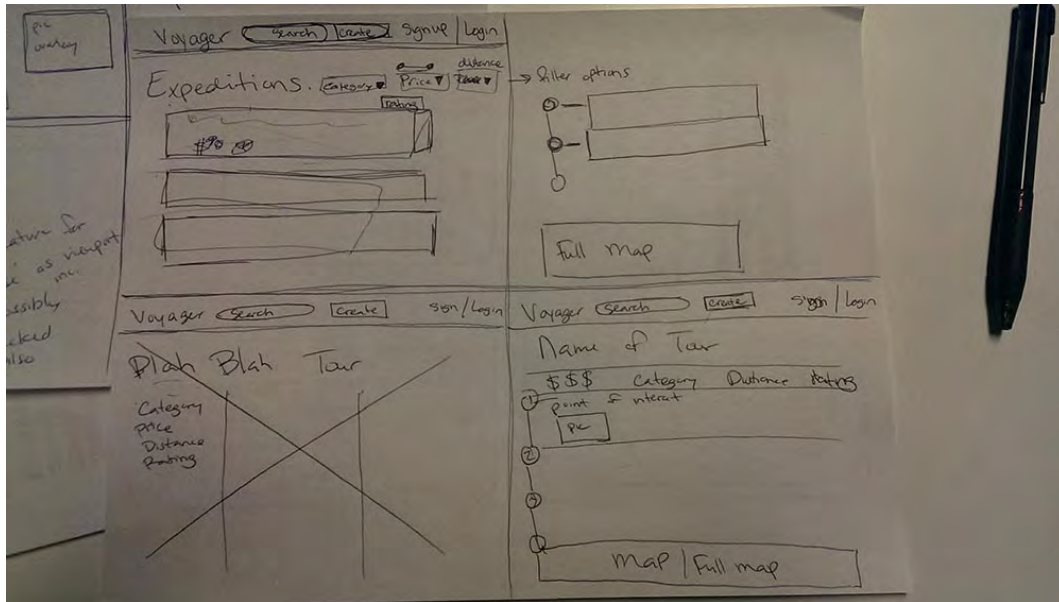
Next, a style guide was developed to allow for consistency throughout the project. We decided to go with a modern font, Open Sans, as well as several different shades of blue and an accent color of green, to play into a nature, outdoors-y, explorer mood. Inspiring, hip but not alienating, engaging, motivating, and discovery are the words we used to keep us on track with the feel of the app we were to build out.
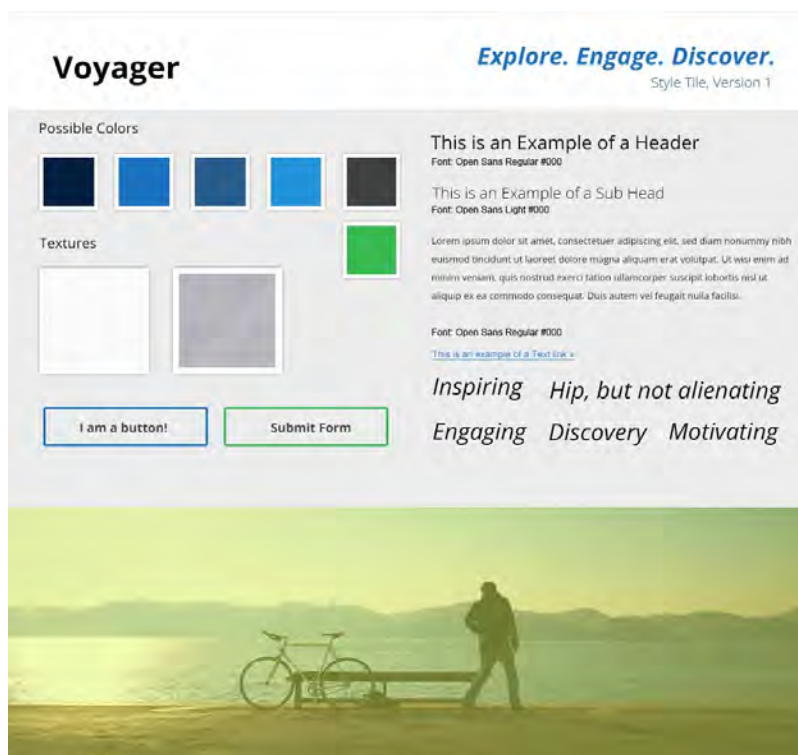
At this point, we were ready to begin coding out the structure of the pages using a mobile-first strategy. We utilized pair programming as we built out a semantic HTML structure. Skeleton was our grid system of choice for this project.

```html
index.html
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <meta charset="utf-8">
5          <meta http-equiv="X-UA-Compatible" content="IE=edge">
6          <title>Voyager | Explore new horizons</title>
7          <meta name="description" content="This shit is on fire">
8          <meta name="viewport" content="width=device-width, initial-scale=1">
9
10         <link rel="stylesheet" href="css/normalize.css">
11         <link rel="stylesheet" href="css/skeleton.css">
12         <link rel="stylesheet" href="css/layout.css">
13
14         <script src="js/vendor/modernizr-2.6.2.min.js"></script>
15     </head>
16     <body>
17
18         <!-- Primary Page Layout
19         ================================================== -->
20
21     <header>
22       <div class="container">
23         <div class="sixteen columns clearfix navbar">
24           <a href="index.html"><h1><span class="icon-directions"></span>Voyager</h1></a>
25           <button id="mobile-menu-trigger" class="link mobile-menu-trigger icon-th-menu"></button>
26           <nav>
27             <button class="link-inverse">Design an Expedition</button>
28             <button class="link-inverse">Log In</button>
29             <button class="inverse">Sign Up</button>
30           </nav>
31         </div>
```

After the core HTML structure was in place, we began to incorporate design elements in CSS, incorporating elements from the style guide throughout. JQuery was then used to add interactivity to our prototype.

Voyager Wireframes
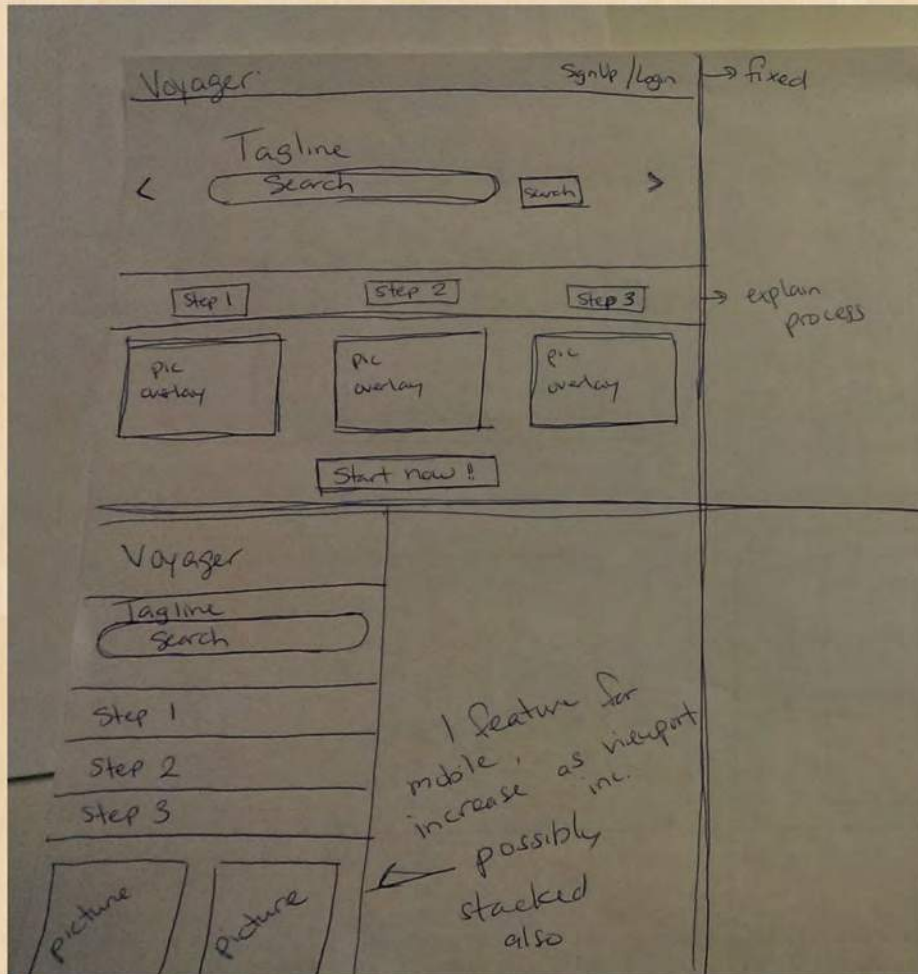
Front Page - Mobile & Desktop

Voyager Wireframes

Voyager Screenshots

Misc Items

Pretty Boi Brand

Original---->

ROCK n RELIEF 2

Gamers Vibes