

# RYANDOCKSTADER.COM

RYAN DOCKSTADER

URL: <http://www.ryandockstader.com>

## THE PROBLEM

The problem my application solves is, I really needed something that showcased what I could do, and previously I'd written it with frameworks and external libraries. While I did write the code to use the libraries, I don't necessarily think it really showed completely what I *personally* can do. So, I decided to completely re-write my portfolio site in pure HTML/CSS/JS.

## DESIGN CONSIDERATIONS

I wanted to make sure that the design flowed well and wasn't jumpy. I wanted the UI to explain itself to the user, and subtly show at least most of the topics that we went over this semester. I wanted the code to be easy to understand, and review by people interested in the site.

## ROAD BUMPS

### SINGLE PAGE APPLICATION

One thing that I really wanted to do was make a Single Page Application without using a framework like Angular, React or View. But I also wanted the site to be simple, and readable by anyone who wanted to view the source code. So, I went about writing the JavaScript in piecemeal dividing each logical chunk I could think of as I coded it into its own function. In most cases, each function was only called once so it's much more of a readability thing than it is DRY coding type of thing. In the end, it's still a little difficult to sort through, but I think I made it as simple as possible.

### FORMING MY OWN GRID

To get each element exactly where I wanted it to be, I had to form my own grid functionality. To do this, I researched flex box and I was quite surprised with how easy it was to set up. CSS3 really has come quite a long ways, and does negate the need for heavy grid libraries.

### THE SPEED OF JAVASCRIPT

Since I did the entire project by DOM manipulation in JavaScript, the page switches felt VERY sudden. So sudden that it didn't feel natural, so I went about adding a delay. It was a delicate balance and it's one that I'm still making minor changes to, to get that perfect balance between it feeling fast without it feeling jumpy.

### THE CUSTOM MODAL

One of the hardest parts of creating the application was to get the modal to display properly. I wanted it to display over what is currently there, but still have modifiable objects within the modal. One of the trickiest parts for me was making the rest of the page unresponsive while the modal is open (so you can't just open a ton of modals on

top of each other). I was able to accomplish it eventually, with lots of internet sleuthing. Thank goodness for stack overflow, and W3 schools!

## NOT OVERLOADING THE WEATHER API

Another challenge was making sure I didn't overload the weather API that I am using. To accomplish this, I have a sync date stored in local storage that I check on page load. If that sync date is more than 10 minutes ago, then the data will reload. Otherwise it pulls the existing data from local storage.

## CODING TOPICS

### JAVASCRIPT

Having a solid JS base was very important here, since the primary section of the page is built, and manipulated using JS. The base HTML body consists of one section tag, and a script import tag.

### JSON

To make the page expandable, I built all the divs that have a repeating pattern as JSON objects. Two places that I feel this really worked well is the education and projects pages. This way, I can easily add new projects and education as I create and acquire them.

I also had to use JSON in my AJAX request to get the local weather, but I'll go over that more in detail in the AJAX section

### AJAX

I wanted to subtly use AJAX in my page, and I accomplished this through displaying the weather in the top right corner. It's defaulted to the zip code of 84321 but can be changed by clicking the weather icon and updating the ZIP code. To accomplish this, I sent the ZIP code in the URL, and store the zip code and the returned data in a weatherData variable that is stored in Local Storage

### LOCAL STORAGE

I used Local Storage to store the weather information, the zip code input by the user, and the last sync date with the API. This used in connection with AJAX allows me to not overload the weather API and persist the users zip code each time they access the page.

### DOCUMENT OBJECT MODEL

I used the DOM to manipulate the page through JavaScript, creating a Single Page Application.

### CSS3 USING JAVASCRIPT

I primarily used this to add and remove classes, but there are some root styles that get added and removed from the main section

### JAVASCRIPT EVENTS

---

## ON CLICK

On click is all over my JavaScript to create the various buttons found around the website

## CSS3 TRANSITIONS

---

### PAGE CHANGE

When the Page changes, I have an opacity transition that makes the page feel more like a refresh, but a little smoother (at least that was the intention)

---

### BUTTONS

I added a transition on all buttons to subtly change the color of the different button types when you hover over them.

---

### NAV ITEMS

There is a border-bottom transition on the nav items to get the underlining effect

## CSS3 ANIMATIONS

---

### JUMBOTRON

To get the jumbotron to change pictures, I have a background image animation that changes the background image in a loop

---

### SNACK BAR FADE IN

The snack bar fades in when the zip code is saved

---

### SNACK BAR BOUNCE OUT

The snack bar does a little dance and then travels to the bottom of the page, while fading out.

---

### MODAL FADE IN

The modal fades in when the weather Icon is Clicked

---

### MODAL FADE OUT

The modal fades out when the modal is closed, or the zip code is submitted.

## CSS3 TRANSFORMS

---

### MAIN PAGE TITLE

For the title on the main page, I had to transform the location to get it exactly where I needed it to be.

---

## BOUNCE ANIMATION

Transform is used to bounce and then slide the snack bar out of the users page.

## ALTERATIONS

### MEDIA QUERIES

Due to time constraints, I ended up having to drop media queries so my site isn't as responsive as I would have liked it to be. This is something that I plan on doing at some point, but unfortunately isn't in the final assignment.

## ACCOMPLISHMENTS

Really, I'm quite proud of this site. I set out with a goal of building each portion of it in JavaScript, and I accomplished that. I think each element feels like it belongs there, and it really feels like a complete site (as long as you're viewing it on a desktop).