Preface

Responsiveness

My website has two main categories for responsiveness, screens wider than 600px and less wide than 600px. In ranges within those categories, it is also responsive.

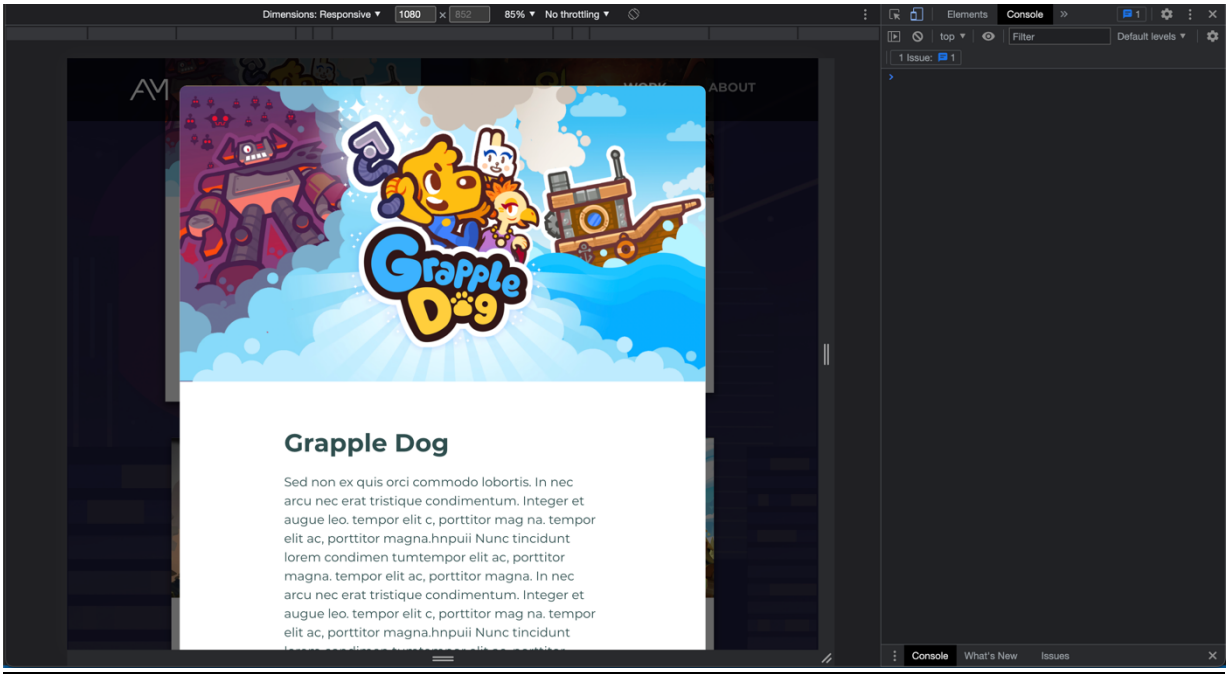
Notes

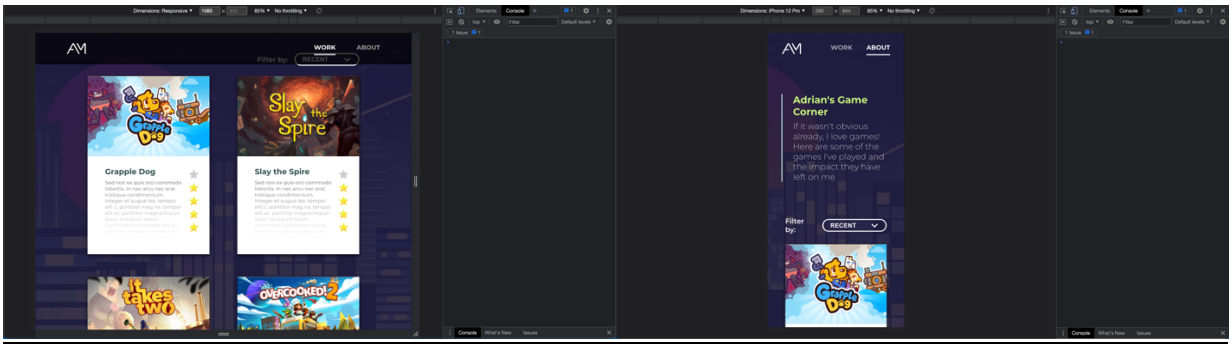
When deploying on Github, I encountered two problems.

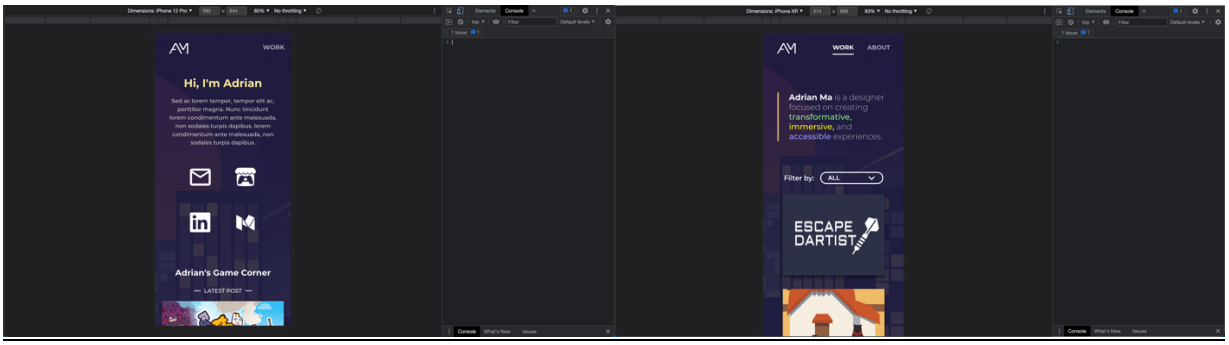
1. One is that the root of the project goes to an empty page rather than the home page which I set as my home page. I couldn't figure out a solution to this problem in time, but pressing 'Work' or the logo on the nav bar navigates to the home page.
2. The second problem is that two specific images on my site would not render on the deployed site even though they rendered fine locally. This broke the < 600px wide responsive rendering for the gallery section of the about page on the deployed site. I struggled with this problem for a long time as my format for those two images was identical to other images on the site, but could not figure out a solution in time.

Also, I overestimated what I could accomplish in the time allotted for this project in my FP2 proposal plan because I didn't account for roadblocks that I will mention later; elements on my roadmap that I didn't get to are the gallery page and the project breakdown pages.

Screenshots of No Errors or Warnings on Various Pages







Part 1: Overview

This website is a portfolio site for my work. My goal for this project was to provide a quick and engaging format for viewing my work, while making the site easy to maintain. The information I wanted to convey were my projects, some info about me, as well as blog posts

about games I've played. I kept my website interesting and engaging by adding interactive animations and illustrated visuals such as the background, while also making the UX of the site as seamless as I could, never overloading the user with too much information at once, while creating responsive animations for contrast and filter options for customizability. My target audience is recruiters or industry professionals looking to assess my skill, as well as myself because my main goal of this project was to make the portfolio easy for me to add to and edit.

Part 2: How the User Would Interact with The Site

— user group 1 (viewers)

1. Scroll down to see projects
    a. After scrolling down a certain number of pixels on any page, a dark background will show up behind the nav bar for better contrast against scrolling elements
2. Hover over a project cover
    a. Hovering over a project on the work page zooms in on the background image while bringing up the project title and skills involved
3. Filter project by type
    a. By choosing a category from the 'filter by' dropdown on the work page the grid can show only projects that fit a certain category
4. Hover over the 'see archive' button
    a. Hovering over the 'see archive' button at the bottom of the work page triggers a shutter out animation
5. Filter game posts by rating or recency
    a. Go to the game corner page from the about page and filter posts by recency, ascending rating, or descending rating
6. Open a blog post modal by pressing one of the cards
    a. Clicking one of the game post cards opens up a model over the screen that can scroll without the background part of the page scrolling
7. Scroll to read the post and exit by pressing outside the modal or the close button at the bottom
    a. Clicking outside the modal or the button at the bottom of each post will close the modal and return scrolling to the full page

— user group 2 (portfolio owner)

1. add a new project
    a. In projects.csv, add a new row with the title, skills involved, categories, and cover image filename for a project and save and it will show up on the site
2. Add a new game post
    a. In gameposts.csv, add a new row with the title, cover image filename, rating, summary, section names, and section contents (that can include more image filenames) for a post and save and it will show up on the site

Part 3: External Tools

1. The first library I used was React Router, which essentially a necessity as it allowed me to link different pages of my site together with clean and concise code.
2. The second library I used was Papa Parse.
    a. I used papa parse because I wanted to read in csv data into react, and Papa Parse has a system to do this efficiently.
    b. I used it to read the csv files containing the data for the work page and the game posts page and convert the data into arrays of structs.
    c. Papa Parse makes it so that I can add new posts or projects to my type without typing any code or even html tags or file hierarchies, making the site very easy to maintain.

Part 4: Iterations

I iterated on my prototypes many times over the course of this project. My initial design for my proposal was much more complicated and filled with cool animations, but after implementing some of it and testing it with some users, I realized a lot of my design was taking away from the main goal of my site: communicate my work experience quickly and in an engaging way.

As such, I greatly simplified the design of my home page, immediately diving into projects, and also split up my projects into ones that go onto the “work” page and ones that fit into the “archive” page so that I could showcase only my best work at the highest level of the IA. This focus on clear communication rather than graphics is also apparent in the simplified nav bar and a much more muted version of the background than before.

Part 5: Challenges

My greatest obstacle was in parsing a single string in the csv file into a full blog post complete with different sections headers, paragraphs, and images. In order to make adding new content as easy as possible and simplify referencing images in the public folder to something as easy as typing __.jpg somewhere in the paragraph, I had to experiment a lot with regex and test many different formats. What made this even more difficult was that these posts appeared on a modal which I initially tried to create using component libraries but eventually ended up building myself because they lacked the exact functionality I was looking for. Interactions such as

scrolling only on the modal but not the background and clicking only outside the modal but not inside to close it were challenging obstacles to overcome.