Marathon Prep – Final Project

For my final project, I decided to revert and update a past assignment that had a lot of potential to be better. The Marathon Prep application I built had a good foundation of what I wanted and was really a topic I care a lot about. In my previous submission, I had used PDFs as the content populated when clicking on a card but decided to change that due to insanely long response times. I hope it is up to the standard of what we learned overall in this semester. Thank you again for an awesome learning experience!

Page Routing

Page routing in this project was set up using React Router. The Router component wraps the entire application, serving as the foundation for routing. Inside the Router, a Routes component contains individual Route elements that map specific paths (like /about or /contact) to corresponding components (About, Contact, etc.). A Menu component is used to display navigational links, making it easy for users to switch between pages. By using this setup, the app allows seamless navigation between pages like Home, Events, About, and Contact without reloading the entire page, delivering a smooth user experience.

Established React Hooks/Hamburger Menu

For this project, I used multiple hooks with React. One of the more common one was the useState hook. It was heavily used for toggling the hamburger menu, tracking form input and even possibly selecting events (if the data actually worked). In my Menu.js, the useState hook is used to manage whether the nav menu is open or closed.

Data Passed

For data being passed, I had the normal card data for Marathon times/training but I also did want to implement some sort of large API. I wanted to use a Race API that would grab the events of marathons that are available to see. Unfortunately, there isn't any real good API's for this functionality and the one I did find I could not get it to populate.

Components Setup

For components, since I decided to build from my previous project, I made sure to include the Accordion, Carousel, and Footer as standard. For all my pages, the header is strongly present as well as the footer. I tried to ensure they are in a similar color palette so that way it comes off as user friendly & consistent. On top of these features, I also made sure to implement the menu (navigation bar) across all pages and making sure that it routes to the correct links.

Styling

As for styling, I strongly wanted to make it consistent across all pages. Especially with my Contact and About page. As it is a page that contains information for the user and interactive button, it needed to be. I used primarily the 'Bebas Neue' font just for its futuristic look and blocky feel.

Known Bugs/Issues

The main big issue I encountered that I could not fix was getting the API to actually populate events for my Events page. The idea was to populate a list of events that would be happening nearby and let the user look at the details. I could not find a good API that had the data I needed.

The other bugs I managed to resolve were making changes to the cards. Originally, I had PDFs populated but it took way too long to load in and was not very user friendly. So I got rid of it and made it refer to a video instead.