Programming Usable Interfaces

Fall 2022 Final Project

Pat Vaidos

Prof. Alexandra Ion

Purpose

The purpose of this website is to provide users with helpful concert information displayed on a map. The target audience is users who are avid concertgoers and music lovers who want to visually see their favorite artists' tour dates on a map. Users first open the page to an overlay displaying instructions on how to use the site and its purpose. Users can return to these details at any time while interacting with the site. Once outside of the information page, users can enter the name of the artist they wish to see in the search bar and interact with the map on the page to explore which cities and countries their artist will be performing in. Users can also scroll through the event information in the left-hand window to get more detailed information, such as the name of the event, the location, venue, and date, as well as a link to the Ticketmaster listing for the concert if they wish to learn more and buy tickets. Users can make multiple searches during one session, to explore other artists as well.

The page mimics the Spotify design language using green and black as the colors of the site. As the name implies, this site is useful for exploring artists' tours in different cities. In further evolutions of this project, users would also be able to look up events based on location to see what events are coming to their town. Google Maps API also has the capability of asking users for permission to view their location, which can be helpful in implementing this feature. In the future, this site would also be able to allow users to save their favorite events for later reference. This site is made for a laptop or desktop device (1280 x 1041px) but is responsive to a tablet view (768 x 677px).

Interactions

When the site opens, an overlay pops up that explains the purpose of the site.

Click anywhere on the page to exit the overlay page. When the page opens, the site automatically populated the page with events near the location of the user. With their permission, Google Maps uses the coordinates to find relevant events according to the location

Submitting a response in the search bar

Type in the name of an artist in the search bar and either click the search icon or press enter to submit.

Scroll through the left-hand window to view all search results

The window on the left-hand side of the screen will populate with search results for this artist. This window is scrollable for viewing all results. Users are able to click “Click here for more details” which will bring them directly to the Ticketmaster page for the specific event they are interested in for ticket information.

Tools

Ticketmaster’s Discovery API

I chose to use Ticketmaster’s Discovery API because I needed a source that I can retrieve and filter through concert data to display on the page and to find locations to mark on the map. Although there is a handful of useful APIs that accomplish this task, I went with Ticketmaster because it has the most extensive data that I wanted to display (location coordinates for venues, links to booking tickets, etc) as well as thorough and comprehensive documentation and tutorials which helped immensely with this project. This API helped the project achieve more depth than simply downloading a JSON file and importing it – as it has up-to-date events added to it and provides a lot of helpful details for the user to peruse.

Google Maps API

Google Maps API is the most comprehensive map API available and has a lot of helpful resources for implementing it in a project such as this one. I used Google Maps as a functional tool within the site, as well as an aesthetic choice. Making the map large enough to fit the screen allows for easy manipulation and visibility of the map, which was important to participants during my initial research stage of the project. It was also easy to change the style settings of the map to fit the visual aesthetics of the whole page.

Bootstrap

I chose the Bootstrap library to set up the visual layout of the site and easily implement responsiveness into my website. Bootstrap also had in-depth documentation on how to create the visual elements I wanted to include in my project, which helped make the site functional, responsive, and as close to the prototype as possible.

Iterations

After beginning the project, I realized that it was going to be really difficult to implement the saved event pages with the data, as that would require me to make event objects independent of the data from Ticketmaster. I also decided that I would change the window that showed the events to a side window, rather than a drop-down window from the navigation as I had previously considered. I found that a single-column format was more successful to view the results, rather than a multi-column side-scrolling window. I also decided to not include images in the results, as this would mean having to create a separate filter for the JSON data to find images based on the keyword search, which would result in more latency for the results to show up. I also defaulted the marker icon to the Google Maps icon, rather than using a green icon as initially planned, since the visibility of the marker would be significantly more difficult to interpret had I stuck to the original prototype. Lastly, I decided to not include information markers for each event, since this would also require adding more objects to the code and more filtering, which would make latency in the page loading more difficult, therefore more of a nuisance for users to interact with.

Challenges

I faced several challenges while I was implementing my site. The most challenging part was the technical aspect of implementing the markers on the map from the event, which was mostly an issue with how I chose to implement the markers. Since I was using a different approach to initiating a map from what the documentation recommended, I had a very difficult time actually getting the

markers to appear, which unfortunately shaved a lot of time off of being able to implement the other features I wanted to include.