

@\$trolabe



Online travel tool to display currency exchange rate and top attractions for a destination. Allowing fellow travelers to make the best decisions financially without hurting their pockets and to planning the perfect day. Uses JS api to offer real-time information.

Avleen Dhillon, Karl Linfeldt, Ryan Merced,
Andre' Grady, Benjamin Stanfield

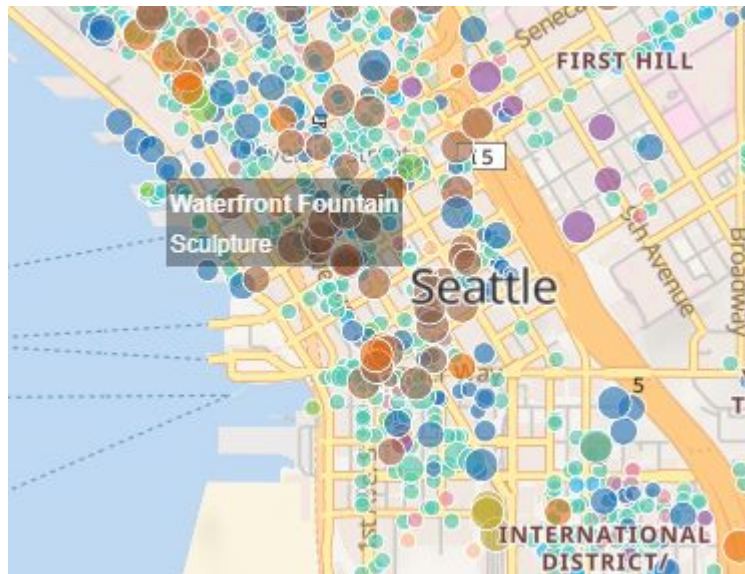
**What is
@\$trolabe?
Seriously Karl this
is your slide.**

API we used!

OpenTripMap API/

- API that allows you to retrieve data from OpenTripMap through an HTTP request
- Displays list of nearby attractions within surrounding area
- Provides informative details of attractions via open sources such as Wikipedia, Wikidata, etc.

opentripmap.io



Exchange API/



exchangerate-api.com

- Real-time API data request for the latest currency exchange rates
- Currently 161 countries and their currencies available
- Make the best financial decision possible

**How we
Implemented!**

Wireframe /



User Narrative/

As a tourist -

I WANT to be able to search a city or region for attractions

So THAT I can view a list of places to visit

As a consumer -

I WANT to be able to view the latest currency exchange rates

So THAT I can make better decisions on cost of activities

As a user -

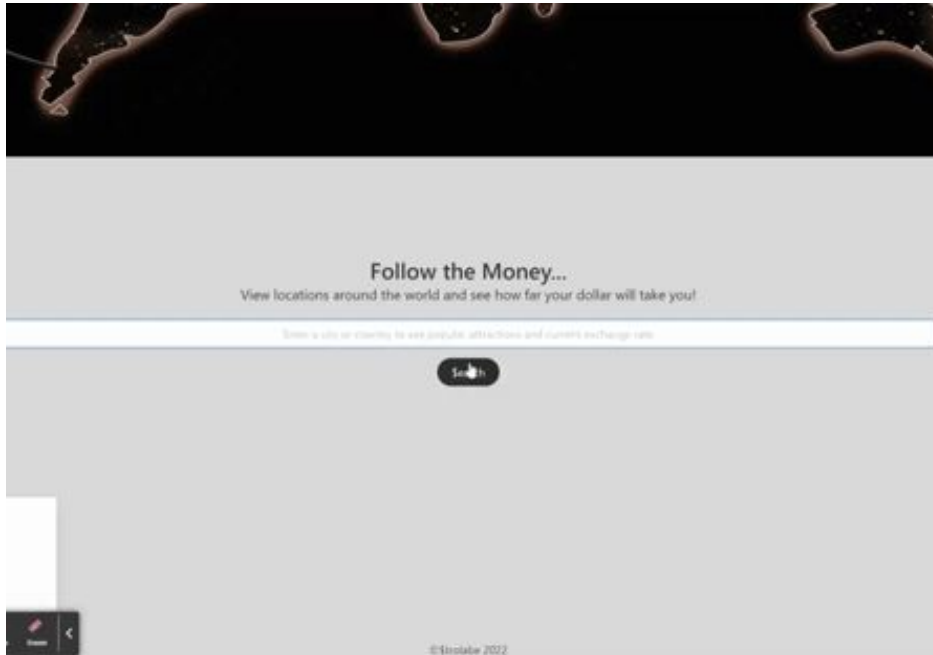
I WANT to be able to save a list of attractions

So THAT I can review my chosen attractions later

The Guided Tour!

Talking and Walking

OpenTripMap API/



We built a basic search input element and linked it into a javascript file specific to the API functionality. Then we made multiple fetch functions to both GeoLocate and then Search for high level data results of Location name and a unique id. This runs upon clicking the search button or pressing the enter key. The id is stored as a data-value within the html element for easy access with a click event and is available for the next fetch that is made on the Details click.

This fetches large object arrays for each query in the search result list. This info ranges from image links to lon/lat, detailed addresses, and robust html template literals of info blurbs. The data can then be passed into local storage by the Add to Favorites button.

Clicking the Favorites button at the top of the page sends us to our favorites list that auto-populates from local storage with an onload attribute calling the local storage fetch function.

Exchange API/

The exchange api gives us live updated exchange rates for 161 currencies around the world. Once the data is fetched on page load, it is built into a template literal and appended into a dropdown menu. The currency exchange value is stored as a data-value that is called out onclick and added to the text box via the `textContent` method. This lets you know the value = to 1 US Dollar. Quick, simple, and effective.



**What went
wrong?**

Oops.



**What went
right?**

Yay!

Milestones reached:

Just getting the fetch to work for each api.

Actually getting the apis to run instead of walk.

No. Seriously.

Search bar and clicks and event targets in the dropdown.

Data attributes in the html!

Actually using that data constructively.

Working through our problems together.



Whats next?

Our dreams for the future!

Love to see our favorites in a modal and no more separate pages.

Modal for our detailed search information display.Model everything!

Clean up the actual search functionality for better results and more accurate locations.

Autofill searches to pinpoint common search areas better.

Cleaner UI to better represent the exchange rate.

Make sure the detailed search result div follows the scroll of the screen.

Create better functionality for displaying the tags.

Insert dramatic hopeful image of a better world?!

