**Design**

Technologies Used:

**ReactJS** – for dynamic frontend components

**Openstreetmap/OSM API** – provides interactive map

**LeafletJS** – fetches geographic data for map

Design Choices:

**Fetch vs Axios for HTTP requests to OSM** – I opted for Fetch API as it is built in, lightweight and simple

**React** – I used ReactJS as I want to get more practice using it and it is suitable for this project - which contains a lot of dynamic parts/components and allows me to manage states and re-render components, such as the button to toggle favourites, and the map when an address is entered

**Map/Geocoding** – I looked at Google Maps API, and Mapbox as well as OpenStreetMap to implement an interactive map. I went with OpenStreetMap, as it is open source which meant there was a lot of documentation and support to use which helped me. It also integrates well with Leaflet, which provides features for the map like markers, icons, other overlays etc.

**Storage** – As the app should work offline, I needed some form of offline/peristent storage. I looked into ways to do this, using a local file, using browser localStorage, and IndexedDB. I chose localStorage, as it did what I needed, IndexedDB is typically used for more complex projects. I also decided a backend database would not be needed, as I am storing the data locally anyway. Offline, you can view the chargers, and add, and remove them from your favourites.

Additional Features:

* Filter map by favourite chargers or all chargers
* View distance of each charger from entered address

A screenshot of a map

Description automatically generated

To run, ensure React is installed and run `npm start` within the “my-react-app” folder

References:

<https://leafletjs.com/examples/quick-start/>

<https://www.flaticon.com/free-icon/electric-car_2175411?related_id=2175423&origin=search>

<https://leafletjs.com/examples/custom-icons/>

<https://legacy.reactjs.org/docs/getting-started.html>

<https://stackoverflow.com/questions/14560999/using-the-haversine-formula-in-javascript>