Visualizer for Homes and Cars For Sale/Rent

A novel visualizer that visualizes homes and cars for sale/rent by using maps and charts. Accessible visualizations give users better insights into data and help them make better decisions.

Kshitiz Sareen
Aarshil Patel

Problem Description

We created a visualiser that users can use to explore homes and cars for sale/rent.
Our problem is summarised as follows:

- A map visualizer that users can navigate through to see the locations of the homes and cars for sale/rent.
- A filtering component that users can use to filter data according to their requirements.
- A bar chart that groups homes/cars according to their location.
- A horizontal stacked bar chart that groups homes/cars according to their type.
- A multi-series line chart to show properties of individual listings. For homes, we display the price and square feet; for cars, we show mileage and price.
- A pie chart represents the number of homes/cars in a particular location displayed on the map.

End Goals

We built this project to solve the problems listed below

- Provide different visualization interactions and filtering choices to help users search for their desired product.
- Provide interactive capabilities for the user to manipulate data visually or look more precisely into the data for examination and analysis.
- Compare other listings to help users make decisions powered by visual data representation.

Architecture

We built our application using a technology stack of React.js, D3.js Go, MySQL and AWS.

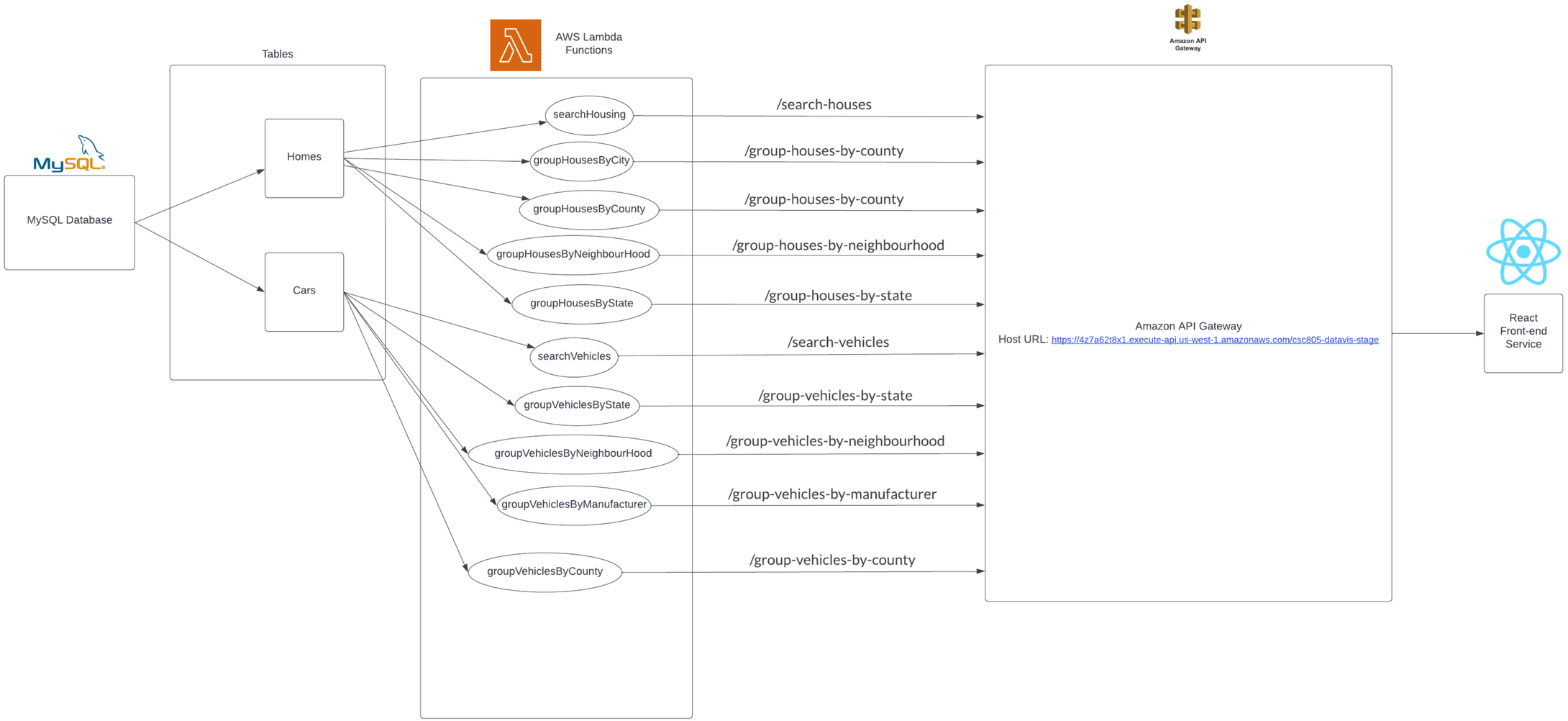
- React.js acts as our front-end service, where we display the relevant data. We created our map using React-Map-GL and made our charts using D3.js. We also created our filtering components using React-bootstrap.

- We used Go to create our backend service by creating multiple functions in a microservices architecture and hosted them using AWS Lambda. Our backend service fetches relevant data from the database and sends it to our React application. Our front end interacts with our back end using APIs created using Amazon API Gateway.

- We created a MySQL database and hosted it using AWS RDS. This database contains two tables: Homes and Vehicles. The Homes table contains all the homes, and the vehicles table contains all the vehicles.

Architecture

Architecture Diagram.



Technologies

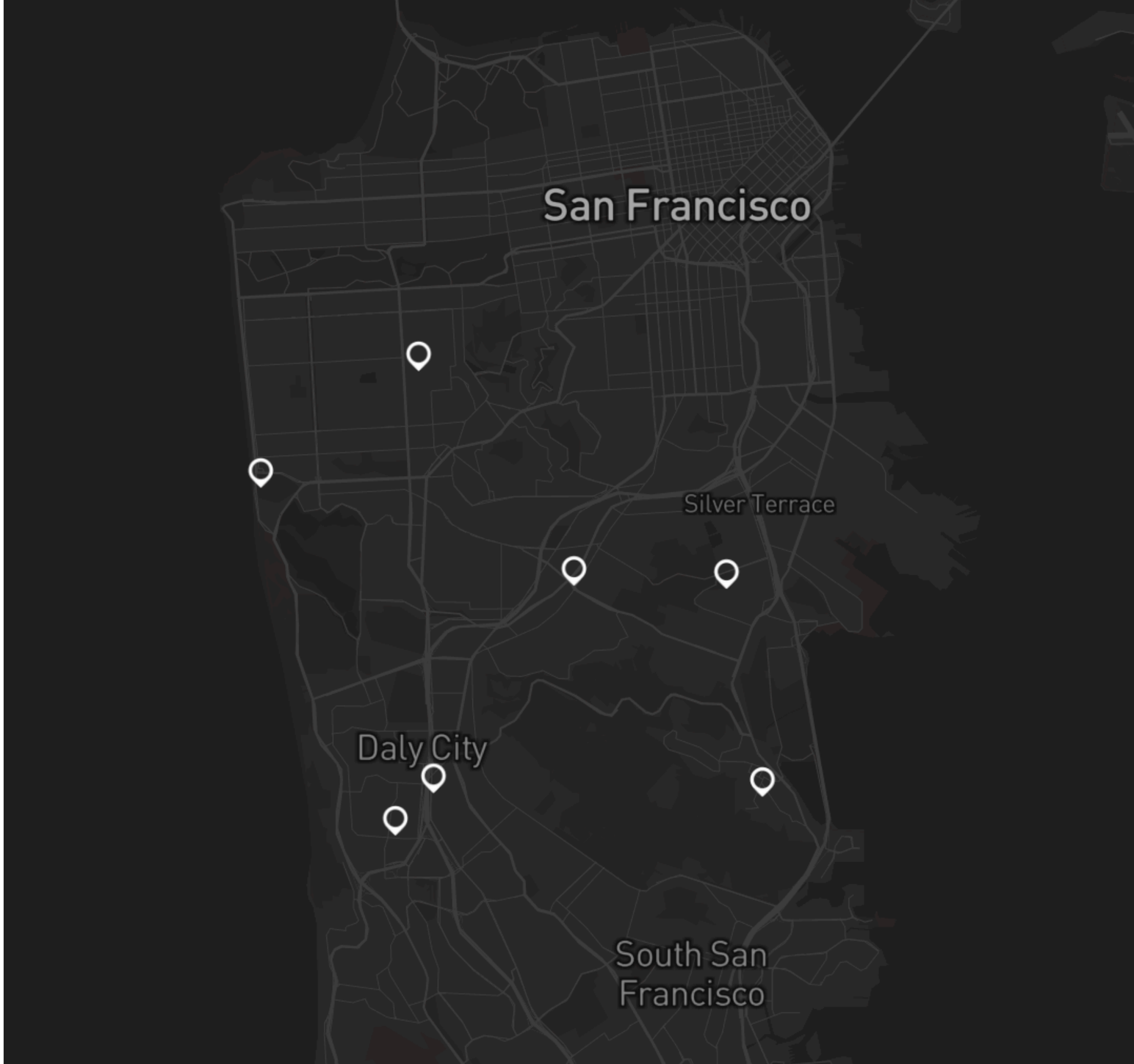
Here are the technologies we used in our project

- D3.js (We used D3.js to build our charts)
- React.js (We used React.js to make our website)
- Go (We used Go to create our backend functions)
- AWS Lambda (We used AWS Lambda to host our backend functions)
- AWS API Gateway (We used AWS API Gateway to create an API that triggers our backend functions)
- MySQL (We used MySQL to store our data)
- AWS RDS (We used Amazon RDS to host our data)
- Python (We used Python to clean our data)
- Pandas (We used Pandas to clean our data)

Images

Map Visualizer

- Map of San Francisco that displays 7 vehicles for sale.



Images

Charts Visualizer

- The charts page shows a bar chart, Pie Char, Horizontally stacked bar charts, and a multi-series line chart that visualizes 7 vehicles for sale in San Francisco.