Project Deliverable 1

Project Pitch:

A web application that allows the user to analyze trends in traffic accidents in the United States across various routes, locations, and times.

Detailed description:

Our application will have a layout similar to that of Google Maps or other directions applications, which will let users specify the starting location and the destination of the path they want to travel, after which the application will draw the path on the map. Then, the application will break the path into several line segments and perform multiple complex queries on the database that stores 4 million tuples of recently recorded traffic accidents in the United States to display what segments of the path the user plans to travel are more dangerous based on how many traffic accidents have been recorded near those segments over the years and how the number of accidents in those path segments has changed over time.

To make the application more useful and specific, our group plans on adding functionality to the application where users can specify various parameters of traffic accidents that would be included in the path's level of danger analysis, such as time of the day, road conditions, type of traffic accident, the time period over which traffic accidents will be considered, and some others. For instance, if a user plans to drive from Gainesville to Tallahassee in the morning and the user knows it will be raining, he/she will be able to select "Gainesville" as the starting location, "Tallahassee" as the destination, and "morning" and "rainy" as query parameters, and "last year" for the application to consider only traffic accidents that happened in the past year in the analysis. After that, the application draws the path on the map and displays it on the website layout for the user to see the path and directions, after which the application will select traffic accidents based on specified user's query parameters, break the entire path into several line segments, and for each path segment the application will assign a score based on how dangerous that path segment is relative to other path segments in terms on the number and severity of traffic accidents in those path segments, and display those scores along with the number of accidents along the path for the user to be able to see which parts of the path are the most dangerous ones in terms of likelihood of traffic accidents.

In essence, the application we plan on creating will be similar to Google Maps with the addition of displaying "danger level" scores along the path based on user-specified query parameters, and how those danger levels have changed over time.

Motivation of the database needs of the application and the potential user interest:

Since our application will perform multiple complex queries to calculate and assign a "danger" score to each segment of the path, it is imperative that it be integrated with a database system. First, to calculate the number of accidents for each path segment based on user-specified, query parameters and how the number of accidents has changed over time, the application needs to perform a trend analysis of recorded traffic accidents. To do so, the application needs to have access to the data of all traffic accidents that have been recorded in the past and only consider those traffic accidents that match the query parameters. As a result, because we need to store a large amount of data of all recorded accidents (4 million tuples) and be able to run queries specified by user-chosen parameters, such as time of day or severity of traffic accident, an obvious solution is to use a database to store all recorded traffic accidents for the application to perform trend analysis on. Conversely, without using a database, we would not be able to efficiently store and access such large amounts of data, and we also wouldn't be able to perform complex queries specified by user's parameters without having database support in our application. Therefore, not only is it beneficial but also necessary for our application to be integrated with a database system for storing data on traffic accidents in the United States and performing complex and specific queries on that data.

Second, the users of our application will be everyday people who plan on traveling long-distance and are interested in knowing which segments of the path they should pay more attention to based on how relatively dangerous that path segment is based on how many accidents happened along that path over time. As of right now, those users can use a service like Google Maps, but such services only provide directional information on how to get from one place to another, without explicitly specifying which parts of the road are more accident-prone. Considering that the users of our application are interested in seeing which segments of the path are more dangerous, the users will benefit from using our application since it will explicitly display a "traffic accident danger" score for each path segment. As a result, our application will benefit its users' driving safety since the users will know which parts of the road they should pay attention to based on their assigned "Danger" scores. If users pay closer attention to, or avoid altogether, more dangerous parts of the path, it will result in safer driving conditions for the application's users and fewer traffic accidents overall.

Web-based user interface functionality:

Our project will require the user to be able to select an origin and destination for their path and, optionally, the time and weather condition when they embark as well as the time range of the traffic data they wish to use on their analysis.

In turn, our application will display a few possible routes on a map and, on a sliding modal, rank them based on their "danger score" providing the individual scores of each segment of each path. Additionally, if the user has not selected a specific time to embark, we will display trends about which

hours of a day are most dangerous, which days of the week are most dangerous, and how each of those has been changing over the years.

Project's Goals regarding trend analysis:

The application will be able to determine not only how many accidents have happened near a certain segment of a path but also how that number has evolved over the years. Additionally, we want to make it possible for the user to see which hours of the day and which days of the week are most safe (and how that has evolved over time).

Real-world data used in the project:

At its minimum, our application requires a data set with the geolocations, time and weather condition of traffic accidents in the United States so that we can compute the distances to path segments and give the user the ability to filter by time and weather condition. As of the time of this writing, we have identified at least one such data set with around 4 million rows. Additionally, it contains data about temperature, humidity, wind speed and direction, etc. at the time of the accident but we chose to limit our use of this data set to the geolocations, time and weather conditions in order to limit the scope of the project. Still, we acknowledge that, were this application to be expanded in the future, these would be good factors to add to our analysis.

Potential queries:

Notice that the distance from an accident to any segment/point is computed from the geolocation of the accident on every query.

Question: How has the number of accidents near this segment changed over the years?

Colloquial Query: Filter accidents by distance to the segment, order by year and count for each year.

Question: What days of the week tend to be the most dangerous to go through this segment?

Colloquial Query: Count the number of accidents who are within a given distance to the segment and are at each day of the week (computed from date).

Question: How much more dangerous is a specific segment path during rainy days?

Colloquial Query: Count the number of accidents that happened when it was raining and are near the segment. Then, count the number of accidents that happened near the segment. Compare the two numbers.

Question: What hours of the day tend to be the most dangerous to go through this segment?

Colloquial Query: For each hour of the day, filter by the accidents that are near the segment and count the results.

Question: What areas have the highest increase of traffic accidents over time?

Colloquial Query: Divide the map into separate blocks for each area. For each block, count the data that is within it for each year.

Software Used in the Project:

Frontend:

Core: HTML, CSS, Typescript, React.

Additionally using: Google Maps API, Google Directions API.

Backend:

Core: NodeJS + Typescript, Oracle, TypeORM.

Deployment: Google Cloud.