COS 787: Assignment 4

Myron Ouyang u16008368

May 2019

1 Problem/ Solution

Road accidents occur on a daily bases around the country, the causes range from environmental circumstances to simple human error. It is possible to gather data on accidents as they occur and build a database of road accidents, storing as much detail regarding each accident as possible. This data can then be processed to produce useful information such as; general patterns relating/ resulting in road accidents, common time that accidents occur, high accident zones etc.

This alone will not be able to solve the problem of road accidents, but it will be able to provide a starting point to generate some preventative measures against it. The Open Street Map (OSM) database will be used to determine the location and surrounding environment of the accident.

2 Description of Application

In a perfect scenario, input data for this system will come from a variety of other applications; such as Google Maps, Waze, Discovery Insure App etc. However for the purposes of this assignment a simple mobile application is created and used as the primary source of input data. Users can simply notify the App an accident has occurred near the user via a single button press, the application will then send the users current location, time and vehicle speed to the database to be stored. The App will also ping other nearby users to confirm the accident.

This system will be implemented only in South Africa. This means that the data can be localized and that there will be comparatively less data to be stored. Since the main goal of this system is to produce useful processed data, the data needs to be properly indexed and categorized so allow for efficient execution of queries.

To be able to produce comprehensive and usable results, a lot of meta data surrounding an accident needs to be stored, such as; the time and location of the accident, parties involved (pedestrians, cars, objects), road and weather conditions, surrounding points of interests etc.

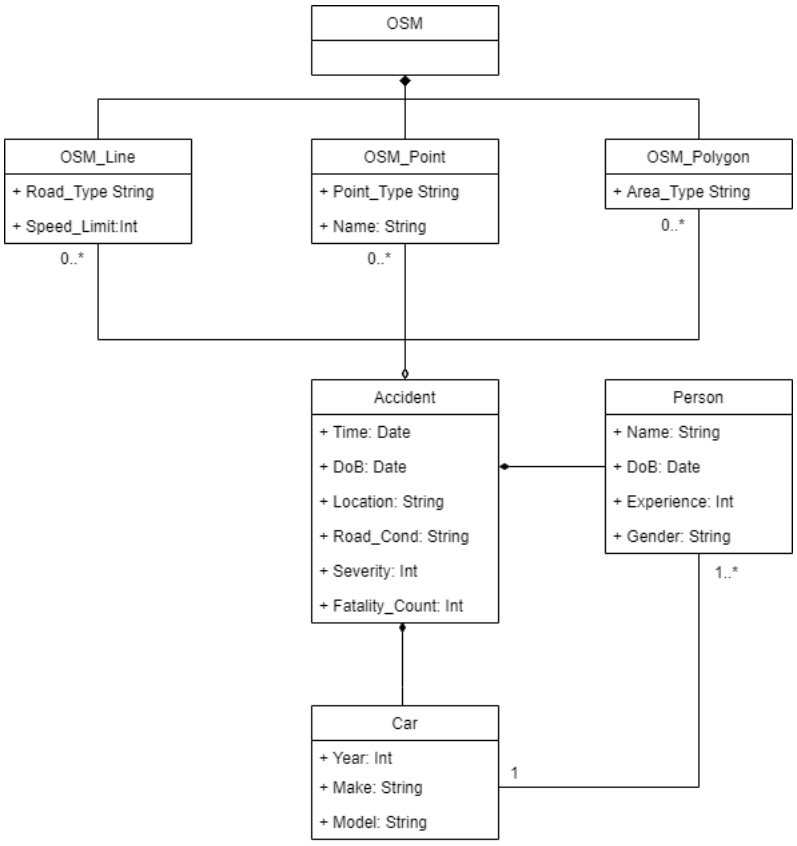
3 Database

A NoSQL database will be used to implement this system. Due to the nature of the system, there will be a large amount of concurrent users using the system, the database needs to able to handle this level of traffic whilst also being scalable. The attributes related to each accident will not be consistent, hence a schema-less approach is most appropriate. Research done by Baralis et al. (2017) shows that NoSQL databases can efficiently handle requests from a higher concurrent user base compared to SQL and it can has better performance when it comes to large data sets [Sharma et al. (2018)].

It is known that NoSQL databases fall short when it comes to built in geospatial functions [Agarwal & Rajan (2017)], however that is not the a problem for this use case, as the general goal is to just be able to gather simple information regarding road accidents. Hence, simple geospatial functions will suffice for this use case.

MongoDB will be the chosen database, for its built in geospatial operations and easy compatibility with most mobile application frameworks. As the data grows, MongoDB has the ability to easily implement sharding on the data set, resulting in relatively low effort scalability of the system. This document database is perfect for storing data with dynamic attributes.

4 High Level UML



Assumptions: All tables include an appropriate identifying attribute (Primary key) and their respective foreign keys. Being a schema-less setup, there will be attributes that are not accounted for in this UML diagram and inversely, the attributes noted might not be present in the actual data. The attributes listed are the ones that are most likely to appear.

References

Agarwal, S. & Rajan, K. (2017), Analyzing the performance of nosql vs. sql databases for spatial and aggregate queries, Vol. 17.

Baralis, E., Dalla Valle, A., Garza, P., Rossi, C. & Scullino, F. (2017), Sql versus nosql databases for geospatial applications, pp. 3388–3397.

Sharma, M., Sharma, V. D. & Bundele, M. M. (2018), Performance analysis of rdbms and no sql databases: Postgresql, mongodb and neo4j, *in* '2018 3rd International Conference and Workshops on Recent Advances and Innovations in Engineering (ICRAIE)', pp. 1–5.