SafeWayz Project Proposal

Project Title:	*Safe.wayz*
Lab Section Number:	**L02**
Student Names:	*Jash Mehta, Aditya Sharma, Anando Zaman, Zackary Ren, Daniel Di Cesare*
Group Number:	**02**
Student McMaster Emails:	*mehtaj8@mcmaster.ca, shara24@mcmaster.ca, zamana11@mcmaster.ca, renx11@mcmaster.ca, dicesard@mcmaster.ca.*

By virtue of submitting this document I electronically sign and date that the work being submitted is my own individual work.

Abstract

As cities get larger with increasing population, so does the crime rate. The safety concerns of travelling from one destination to another can be worrisome depending on the routes taken to get there. Our proposed solution is an Android application which finds the optimal path for the user while avoiding high crime-rate areas. The target city of choice will be San Francisco as there is a lot of publicly available data involving crimes at various street intersections. A large dataset for this location will be analyzed to produce the optimal path for the user.

1. Objective

The objective of the project is to develop an Android Application that informs users of the safest and fastest route from their intended start to their destination given their particular specifications.

2. Motivation

Summary:

Many individuals within the city make late-night commutes either from school or work. Knowing that 700 violent crimes (per 100k people) [1] occur annually, with the majority occurring between 6-11 pm [2], this commute is often a dangerous undertaking.

The main concern is that there are casualties and injuries occurring across major cities that can be prevented. People are unaware of the particular crime rates in certain areas/neighbourhoods and unknowingly venture through them only to be injured, assaulted or worse. We are trying to address this problem via our product.

Why is this problem important to be addressed?
Addressing this problem is important because it is a matter of public safety. Everybody should feel safe in the community they are in, and we are enabling the public to have more control over this issue.

Users of the product:
The users of the product will be the general public. There are no restrictions whatsoever. Anybody that has access to a mobile device will be a potential user of the product because as a person, they will most likely be concerned about their safety.

Expected outcome and utility of the product:
The expected outcome from the deployment and utilization of our product is a decreased casualty/ mortality rate in locations where people leverage our product. People will be able to leverage the power of big data to make better-informed decisions to prioritize their safety.

3. Prior Work

There are a few applications that currently already have the functionality of directing users to a destination based on crime data. An application known as SketchFactor was able to implement the mentioned functionality based on feedback from users on their perspectives of "sketchy" areas.

However, their implementation gave rise to a problem - the data provided by users was often racially skewed. In other words, it didn't reflect the actual crime or danger levels of a certain path.

We are proposing an alternative implementation and execution. Rather than rely on data provided by users, we believe it would be best to use historic data regarding crimes (robbery, traffic violation, vandalism etc...) in our target location and generate optimal paths backed by data. This reduces the bias of the general community from skewing the results through racial targeting of certain areas with specific ethnic groups.

4. Input/output and proposed solutions

The dataset that will be used is extracted from kaggle open datasets.
(URL:https://www.kaggle.com/c/sf-crime/data). This dataset targets the city of San Francisco for crime related activities at various intersections followed by its positional coordinates (long,lat). In addition, it also includes the time at which the crime occured, which can be used to compute alternative routes for a given time of day. This dataset is translating into a weighted graph based on crime frequencies which will be used to find the least crime path

Outputs produced:

The outputs produced from the data will be the shortest path with the lowest weighted crime activity. This output is what will guide the user to their desired destination. The most efficient path from point A to point B will be considered but the numbers of crimes, and possibly the severity of the crimes(ie; homicides), will be weighed with higher priority. All these factors combined together will produce the optimal path.

Proposed solution in terms of input used:

Using the kaggle dataset on crime rates in San Francisco, a weighted graph for crime rates at intersections will be generated. To create this weighted graph, three features from our dataset will be utilized: time, location (street intersection & geographic coordinates), and number of crimes at that location. Intersections will be represented as nodes and streets will be represented as edges between the nodes. The graph weight will be the ratio between the number of crimes occurred at a location and the total number of crimes in the city. This weighted graph of San Francisco will only need to be generated once; subsequent uses will only traverse the graph.

When a user wants to travel from point A to point B, the application will locate the points on the graph previously made. To find an optimal path, in terms of distance and crime rate, we will utilize a weighted graph traversal algorithm such as Djikstra or A*. For example, the user may want to navigate from OAK ST / LAGUNA ST to KIRKWOOD AV / DONAHUE ST. Our application would calculate possible paths from the given start to end coordinates and check if any intermediate street intersections are dangerous. If the intersection at EDDY ST / JONES ST was determined to have a high crime weighting, an alternative path would be suggested.

5. Algorithmic challenges:

To begin, we will take our crime dataset and filter (sort) the .csv by the name of the intersection. This information will be output into a JSON; where the current intersection has the number of crimes and what other streets it is connected to (search). After this data is collected, a ‘weighting’ will be calculated per intersection, based on the average number of crimes. This weighting will be used when we apply Djikstra’s algorithm (graph traversal) to calculate a path between two points with the least risk to a potential crime. Using the resulting data, we can plot the locations of the intersections using the google maps API; this provides a visual representation of the path.

6. Project plan

Milestones and Deadlines

Milestone	Hard Deadline	Soft Deadline	Date
0. Team assignment	❖ Pick team members		Week of Jan 27
1. Project proposal	❖ Finish project proposal		Week of Feb 2
2. Presentation 1	❖ Class presentation	❖ Prepare Presentation ❖ Start Project Development ➢ Create node tree of paths ➢ Summation of crimes at each node	Week of Feb 9
3. Progress check 1	❖ Prototype for demonstration	❖ Data sorting completion ➢ Crimes sorted based on location ❖ Basic implementation of the maps API	Week of March 8
4. Presentation 2	❖ Second class presentation	❖ Finish plotting crime data on maps API ❖ Plot best route on maps API	Week of April 6
5. Final project coded	❖ Java implementation in Eclipse ❖ Design document ❖ Team evaluation	❖ Finalize mobile app	Day of April 12

SafeWayz Project Proposal

Roles

Jash Mehta
→ Researcher - Finding Datasets
→ Programmer - hashing the JSON to sort streets intersections hashmap by crime

Aditya Sharma
→ Project Leader - keep the group on track by assigning roles and planning
→ Client/Tester - Testing of the completed algorithms so far(DFS) to match google maps
→ Programmer - Created Adjacency List of all data. Performed data cleaning and formatting from CSV to JSON via Python. Developed DFS Java algorithm code

Anando Zaman
→ Researcher - Found valid crime dataset along with all streets of San Francisco via Kaggle
→ Programmer - Developed rough android app for testing Java code. Developed BFS algorithm in Java along with code for hashing JSON datasets.

Daniel Di Cesare
→ Designer - Rough app design
→ Programmer - Android Google maps plotting and Geo-encoding tests
- Planning of Dykstra's algorithm for mapping optimal paths

Zackary Ren
→ Designer
→ Programmer - Planning of Dykstra's algorithm for mapping optimal paths

Iterations

1. Contemplating use of DFS, but usage of BSF was implemented to lower time taken for completion
2. Non-crime data was merged with crime data
3. Adjacency List of all data was created in a JSON format
4. Added extra features to accompany sorting and searching algorithms

Log - Currently complete (Future updates can be found in the log.docx file)

- DFS done for creation of all possible paths.
- BFS done to find closest paths in terms of intersections
- App framework for scrap demos
- Created a weighting formula and assigned weightings for recency and crimes to streets
- Developed Adjacency List to map streets
- Merged data from non-crimes and crimes to a single JSON file.

References
APA FORMAT:

Donald, H. M. (2019, October 3). Opinion | The Streets of San Francisco. Retrieved January 26, 2020, from https://www.wsj.com/articles/the-streets-of-san-francisco-11570143083

Laskow, S. (2014, September 2). When Big Data Maps Your Safest, Shortest Walk Home. Retrieved January 26, 2020, from https://nextcity.org/daily/entry/pedestrian-open-data-safest-shortest-walk-map

Tarlekar, S. (2016). Algorithm to Determine the Safest Route. Retrieved January 26, 2020, from http://ijcsit.com/docs/Volume 7/vol7issue3/ijcsit20160703106.pdf

Location Inc. (n.d.). San Francisco, CA Crime Rates. Retrieved from https://www.neighborhoodscout.com/ca/san-francisco/crime

National Center for Juvenile Justice. (n.d.). Retrieved January 26, 2020, from https://www.ojjdp.gov/ojstatbb/offenders/qa03401.asp?qaDate=2016