Real-Time Smart City Map On Temporal Data

Mini Project Report

FOR THE DEGREE OF

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Submitted By

Ankit Kumar (IRM2016002), Avinash Yadav (ITM2016004), Pratyay Kumar (ISM2016002), Neeraj Mishra (IIM2016003), Nilotpal Pramanik (IRM2016501)

UNDER THE SUPERVISION OF

Dr. Sonali Agarwal

Assistant Professor

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, ALLAHABAD

(A UNIVERSITY ESTABLISHED UNDER SEC.3 OF UGC ACT, 1956 VIDE NOTIFICATION NO. F.9-4/99-U.3 DATED 04.08.2000 OF THE GOVT. OF INDIA)

A CENTRE OF EXCELLENCE IN INFORMATION TECHNOLOGY ESTABLISHED BY GOVT.OF INDIA

April, 2019

CANDIDATE'S DECLARATION

We hereby declare that the work presented in this project report entitled **Real-Time Smart City Map For Temporal Data**, submitted towards fulfillment of mini project sixth semester (2019) report of B.Tech (IT) at Indian Institute of Information Technology, Allahabad, is an authenticated record of our original work carried out from January 2019 to April 2019 under the guidance of **Dr. Sonali Agrawal**. Due acknowledgements have been made in the text to all other material used. The project was done in full compliance with the requirements and constraints of the prescribed curriculum.

Place: Allahabad
Date: 29/04/2019

Ankit Kumar (IRM2016002)
Avinash Yadav (ITM2016004)
Pratyay Kumar (ISM2016002)
Neeraj Mishra (IIM2016003)
Nilotpal Pramanik (IRM2016501)

CERTIFICATE

This is to certify that the above statement made by the candidates is correct to the best of our knowledge.

Date: 21/02/2019

Place: Allahabad

Dr. Sonali Agrawal

Assistant Professor

IIIT-Allahabad

Acknowledgement

We have tried our best to present this project in a complete manner without failing the deadlines, in this project. However, it would not have been possible without the kind support and help of many individuals. We would like to extend our sincere thanks to all of them.

We are highly indebted to **Dr. Sonali Agrawal** for her guidance and constant supervision as well as for providing necessary information regarding the project whenever we were stuck. We are largely indebted to them for their support in the project. We would also like to thank **Mr. Ashutosh Kumar** for guiding us and providing many useful information for our project.

Table of Contents

ABSTRACT

The globally used google-map is a very interactive application which provides an efficient map along with the real-time information about the traffic on the roads or streets and offers brief information about the neighbourhood areas of that particular place in various aspects like medical shops, restaurants, tourist spots, hotels and etc. But it does not provide the exact real-time condition of the road whether it is closed for construction for a particular duration or any instantaneous incidents like road accidents, bridge or flyover collapsed and other issues which are essential information for passengers and pedestrians. So here, for a specific place or area we are implementing such a map providing all such information in real-time through a mobile application. These all kind of information are acquired from the local traffic controllers and supplied to the admin of this application. On the basis of those information the map is updated manually in real-time with user understandable specific interacted manners.

***Keywords**: Real-time map generation, GPS data, smart-city, smartphone sensors, Voronoi Graph, Bipartite Graph, Firebase, Cloud Messaging.*

1. Introduction :

This report presents the idea of a mapping system for a defined campus/city using a various graph related techniques on GPS data sets containing the data points of people who use the locations in the map on a regular basis which is different from regular conventional GOOGLE MAPS as it basically intended to implement the temporal data and restrictions that are NOT PERMANENT or even long lasting and are imposed for smaller amount of time with attempts of several high performance mechanisms implementations for real time data processing. Here we try to achieve various high performing and highly accurate navigation mechanisms to be used on GPS data sets including day to day data points from user's mobile or equivalent sensors. In this process we discovered various methods to generate missing data points in the map like image processing methods (average filter, histogram equalization, etc) to build the basic structure of map and methods to make the performance faster like histogram storing with their pros and cons. We also explored various advancements on real time Temporal GPS data. We will try to implement all this using appropriate Machine Learning and/any other techniques as per necessary. Functionalities of this Real time MAPPING system includes notification for the user for any restrictions imposed between the intended path to destination. Providing a powerful dashboard for the administrator to manage the restrictions that may get uplifted after a predefined time interval.

2. Motivation :

When we talk about maps for navigation always GOOGLE pops up in our minds but google maps as we all know takes at least around a month to get updated but also the information it conveys is not too detailed. Consider a situation or rather a premise where a lot of people work/live just like our college campus, if we need to go to a particular location and the usual road is under construction or the particular building is closed today then either we have to go there in person or rely on the information of anyone who went there recently since we cannot find such information on google maps. So this project popped in our minds to connect a populous and large premise with this kind of real time dynamic navigation system and save precious time of everyone.

3. Problem Definition :

The problem definition can be precisely framed as Real time Map for a smart connected campus / premise where situation and condition (temporal) about any place can be obtained on user's mobile phones and also an administrator side which allow to impose and uplift certain temporary restrictions within.

4. <u>Literature Survey</u> :

Over the last several years researchers are quite focused in the smart city project. Real-time map generation and providing instantaneous information for a particular area in specific manner is also one of their research attraction in this domain. In a very recent work citywide travel times, traversed paths, and missing values over a large-scale road network are estimating using spatially and temporally sparse GPS traces *(Weizi Li et al)* [1]. And such map generating and map matching *(Moustafa et al)* [2] procedure contributed a quite significant responses in prospect of the interactive map construction for a specific area or place. Utilizing Hidden Markov Model(HMM) algorithms, online Viterbi algorithm on the Collected accurate location data at high sampling rate (1 sample/sec) using a combined GPS/GLONASS receiver a real-time map for noisy cellular-based trajectory traces is achieved. As well as they also providing cell-phone associated information for localization systems; opening up new possibilities for location-based services from both network and client side. Prior work primarily related in this area is about detection of the real-time traffic congestion and suggesting the optimal path through smartphone *(Garg et al)* [15]. There only longitude and latitude were retrieved as the location data and proposed a client-server architecture for optimal alternate road for driving using smartphone as well as provides information about dynamic change of traffic lights after determining congestion in the different directions.

Traditionally, map matching algorithms use road segment connectivity with data about position, speed, distance etc. for map matching which is suitable for high frequency data but a considerable drop in correct link identification is seen for low frequency data such a fleet of private cars. A weight based shortest path and vehicle trajectory aided map-matching algorithm is therefore required (*Mohammad Quddus et al*) [10]. For map matching purposes sensors are used for GPS trajectory data collection. However, the data presented often is unable to capture real positions due to certain constraints. Thus meta information such as vehicular braking data, vehicle status, traffic congestion etc. is also taken into consideration and incorporated to provide accurate GPS trajectories. (*Gang Hu et al*) [5]. Environmental factors and other meta information factors can also impact the GPS trajectory data, thus consideration to these must also be given (*Kaveh Sarrafan et al*) [6].

To generate map we just need the data points of the GPS sensor which can be used to create a consistent and unambiguous map [12]. The data points can be converted to a neat map using only a 4 step image processing inspired algorithm using averaging, contours filter followed by creating a Voronoi Graph (A set of points that are situated at equal distance from the nearest two points (the boundary points) of a set of polygons which are closed. Then the updation can be carried out using BIPARTITE graph methods generation the basic map structure for further processing. [13] We can also generate the topological graphs automatically using the CAD structured files. Here the .dmx files which are generally used by CAD applications are given as input and the map of those are created automatically. Generally it is difficult to create map of a building but we can find their blueprints in the form of CAD files which this paper presents a method to convert into maps of our interest. [14] Here we are presented with a fairly unique way to decide whether to include the waypoint into the road or the path is valid or not using probabilistic modelling approach. Here instead of using only GPS data points we also consider the data from other sensors like bluetooth, position(height) sensors. This idea is inspired by the studies of González, Hidalgo & Barabási (2008) - study of mobile position on human body, Tam & Lam (2008) - study of vehicles GPS data, Bierlaire, Chen & Newman (2010) - study of Smartphones GPS data. The data is splitted into multiple unimodal segments - the continuous journey without vehicle change or mode change and then each segment is associated with some probability and their cumulative effect is considered for better detection of roads and removal of wrong data.

To predict the traffic in real time, and planning the new fastest routes according to the predicted traffic [4] can be done by Gaussian Process Regression, KNN graph techniques with the help of SCATS sensor using STRF model over the service layer API of the OpenTripPlanner. The comparison of the map generation algorithm [11] like KDE algorithm, minimum link algorithm, incremental track intersection, k- means algorithm is done in this paper using the OpenStreetMap dataset. So we can use the efficient and compatible algorithm for our Temporal map. Using the multiple sensing sources we have to create the real time and unambiguous map [3] using the OpenStreetMap API. To implement this, they are using simple graph theory and map matching algorithms by creating and updating routes and sub routes in real time.

Concluding Remarks based on Literature Review : After going through all these selected papers and we can feel that the need of this project in real life is quite relevant in the domain of smart-city related research. Our general understanding towards this problem is utilizing GPS location data and incorporating probabilistic multimodal map matching algorithm in very specific HMM is very much helpful in constructing map in desired format. Initially we used cellular based trajectories in accurate location data at high sampling rate (1 sample/sec) using a combined GPS/GLONASS receiver. According to the target of this project, a client-server architecture for this particular application can be proposed.

5. Map Formation :

Real-time location data is acquired in terms of Longitude and Latitude from the mobile application. Utilizing the Google API for location the data is fetched in a specific time span like 1 minute along with exact date and time format. This location data is stored as a single .csv or .kml file over Firebase for realtime, secure and reliable usage. For each execution of the app the data is separated through mentioning the name of the particular starting location point and the finishing location point.

For the app we need to build a map from the Dataset collected it is done according to the following sequence for Plottable data. The Data after preprocessing looks like:

1. **Waypoint** : having several attributes like colour, width, track name it belongs to, its description. This type of entity describe data point in between a waypoint/road/track.

2. **Legend Point** : This entity is usually some building / important place or a legend in the map having mainly 4 attributes:
    a. Name
    b. Description (coordinates and height)
    c. Timestamp
    d. Visibility (binary, 0-not visible 1 - visible)

5.1. Data Preprocessing

The Data collected is raw gps data with inconsistencies and missing values and thus is unfit to be plotted on a map. So the data is prepared as follows:

5.1.1. Cleaning Un Important Data

Since the data is collected over mobile and using google API the data contains several attributes many of which are irrelevant to the project hence needed to be removed.

5.1.2. Removing Inconsistency in data

Since the modern day GPS data is not 100% correct or reliable due to GPS inaccuracies. Some data points are way out of the usual positions and hence acts like Outliers and need to be removed.

5.1.3. Overcome Missing Values

Due to poor network connections and/or black spots (no network coverage) the data can be spare at some points or no data points could be there in case of roads. So, the roads may become incomplete and could make no sense. So, we added any possible missing values in the dataset.

5.2. Divide Data into Points and Roads

The data at first contained 3 types of geometries :

5.2.1. Data Points : This type of geometry contains all the points having information about a specific point along with its other attributes as described before.

5.2.2. Way Points : This type of geometry contains set of Data points to form a road with source and destination and turns.

5.2.3. Multi Geometry Points : This geometry contains set of data points and waypoints to form complex paths.

But due to inconsistency we had to remove the multi Geometry Points.
We followed this paper[12] to process the data into plottable form and map creation as a 5 step process :

- Generate a 2D histogram indicating the number of GPS fixes found in each cell : According to Nyquist - Shannon Sampling Theorem, the cell width should be at most half the minimum road width to prevent aliasing. The GPS readings' inherent errors might mean that the trace misrepresents the roads' true positions. Also, the roads' true positions becomes clearer as noise due to errors becomes less significant.

Fig.1 Image of U.K. map histogram with one pixel per cell

● So the solution is to apply a classic image processing technique "Averaging". In this we consider the effects of neighbour cells to estimate value of a cell/pixel, this helps in reducing the noise and enhancing the dataset. And also includes in between missing data points for which we also use continuation increment in between continuous data point as there may be some data points left due to discretization

Fig.2. Same map with average filter applied.

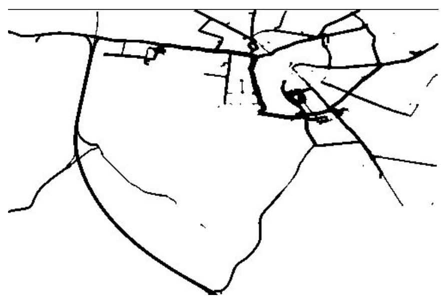


Fig. 3. Image after binarization.

● As we can see the map doesn't make a lot of sense so we apply binarization with thresholding to determine which data points are most likely correct and discard the remaining. At this point we reduce the inconsistent data point in the dataset available (inconsistency due to network coverage, blind spots, etc).

● After binarization we get a well defined and almost consistent map, but to get a correct map we need to apply a "Contour filter" to trace the boundary of the
roads.

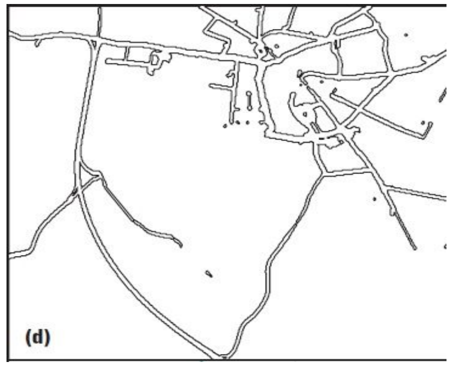


Fig. 4. Image after contourization.

● Lastly we create a centerline which will ensure the most likely road segments by averaging the outlines, but this will create some spikes due to inconsistencies which can be easily removed using thresholding so final map will look like in fig.5.

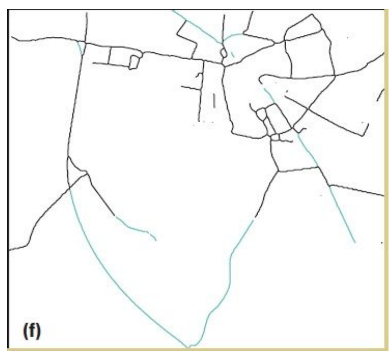


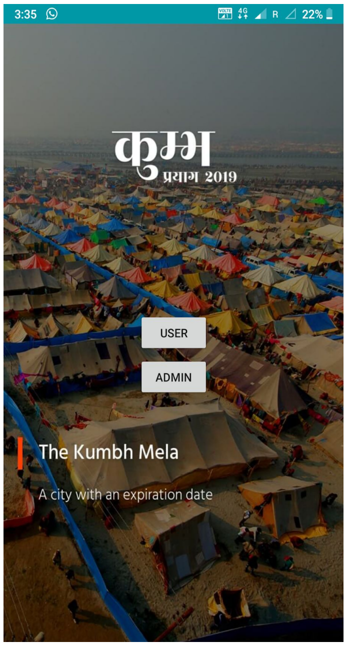
Fig. 5. Final inconsistent map after centerlining and thresholding.

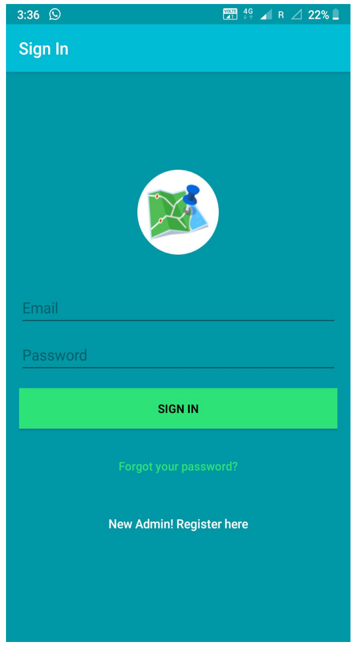
6. App Formation

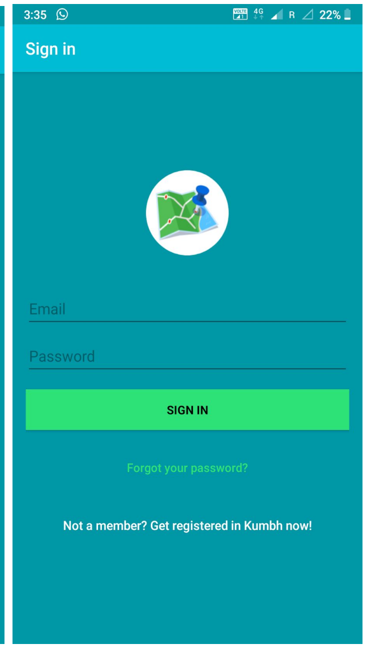
6.1. Front End

6.1.1. Register, Login, Signup (User and Admin)

The users are provided with registering option which allows new users to avail the facilities provided by the app. Previous existing users are authenticated by their email id's and passwords. On successful authentication they are forwarded to dashboards.



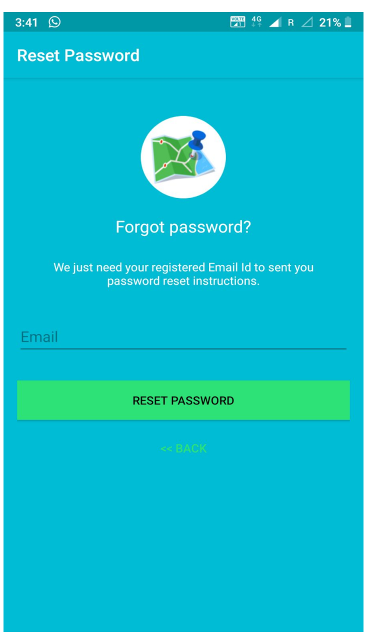


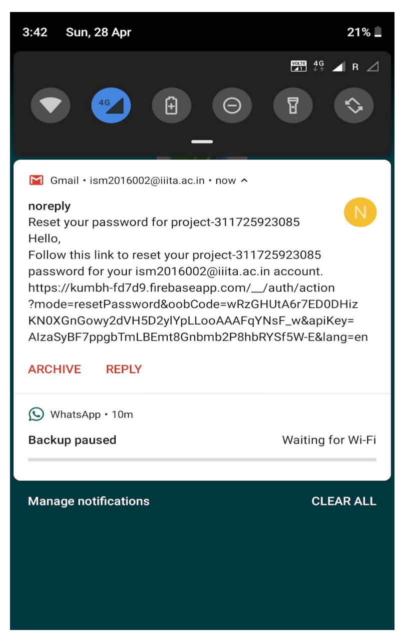


(Main page) (Admin login) (User login)

6.1.2. Forgotten Password Management and Security

The facility for resetting password is also provided by the app. On clicking the option for forgot password the user is asked for his/her email id to which the password reset link is sent. The user can then change his password. The firebase engine provides a secure method of authentication and server side encryption thus maintaining an effective and secure app experience to the user

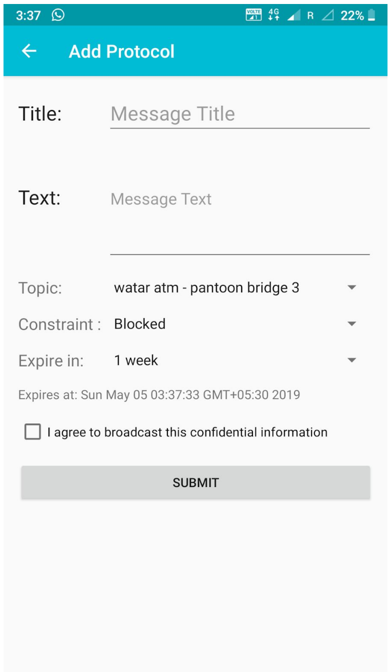


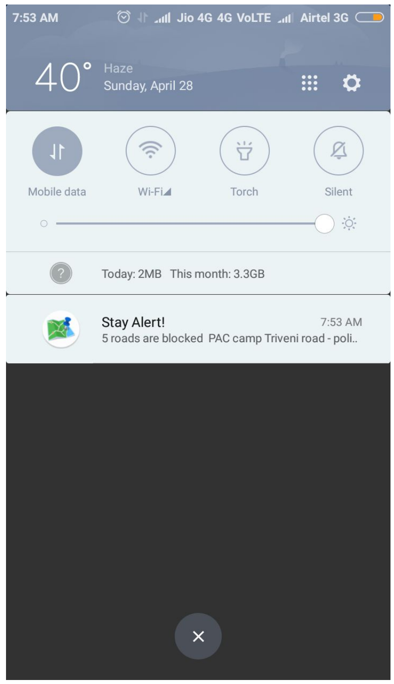


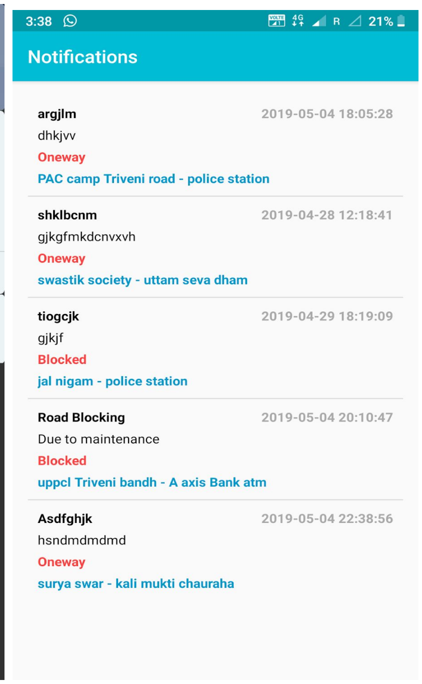
(Reset Password) (Mail for password reset)

6.1.3. Adding Constraints and Notification

The admin dashboard allows him/her to add constraints on the roads and as soon as this is done a notification is broadcasted to all the users regarding this change when they are logged in.



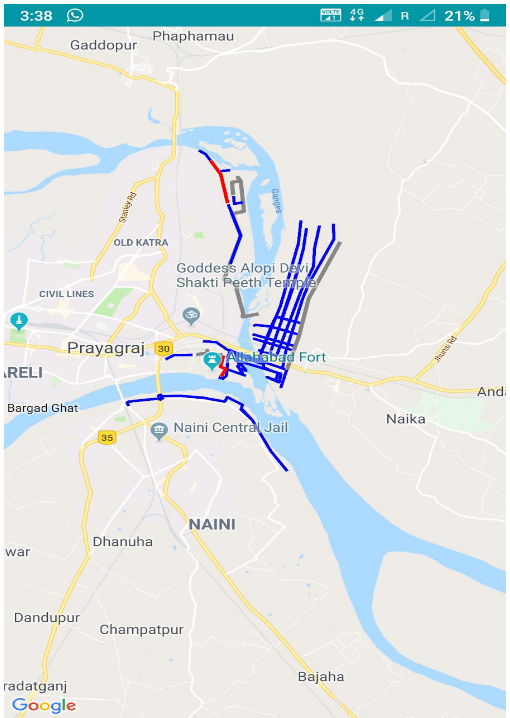


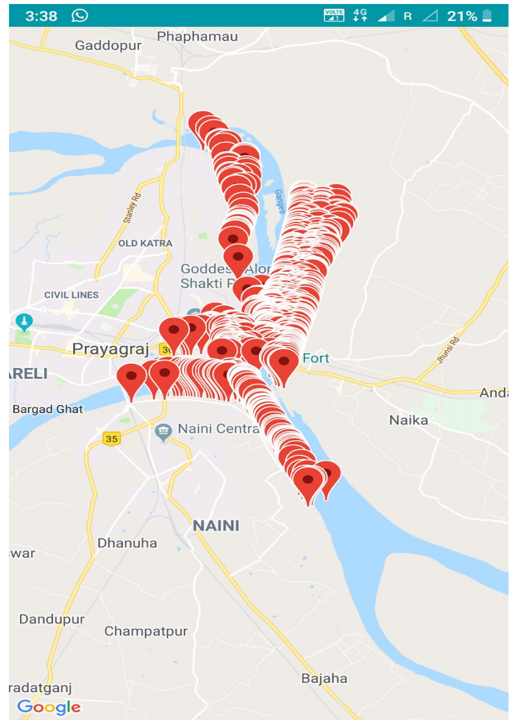


(Protocol addition window) (Instant Notification) (All notifications)

6.1.4. Maps

The user and admin dashboards allow them to view two maps one showing all the blocked roads (in red color), one way roads (in gray color). and the other showing the markers to coordinates on the local locations in the map.





(Map with blocked roads marked in red,
Map with one way roads marked in Grey,
Map with open roads marked in blue)

(Map with marked locations)

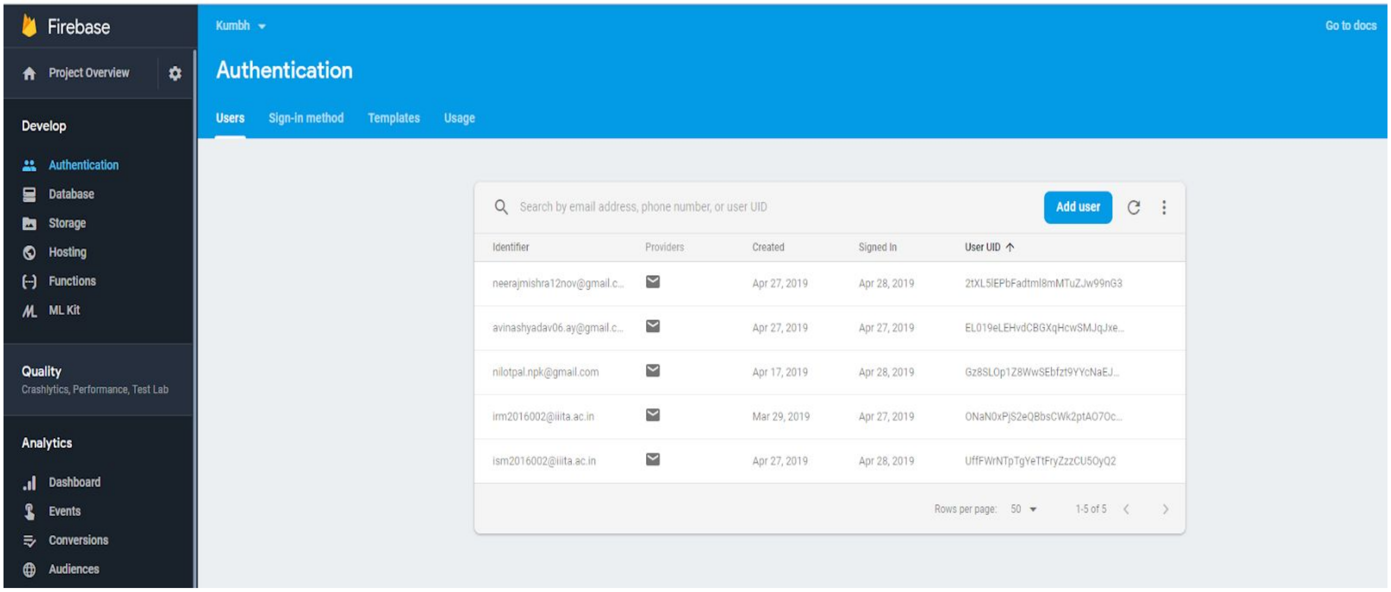
6.2. Back end

6.2.1. App Architecture

The architecture of the android application mainly based on

- **Possible Constraints :**
  KML dataset has been processed specifically in line (road) segment data and point (location) based data. For displaying blocked roads and one way roads on the already generated map on the basis of admin information entry, the road name has been matched (string matching process) with the database and perform the required action over the location/line(road) segment.

- **User data authentication and Security :**
  During user signup user id and password is needed to enter. That information is stored in the firebase in the form of encoded string which is matched and in case of login that candidature is matched and provide authentication and security in the application.



User Details in Secure Authentication with key

- **User Push Notification:**
  For each updation on the map is notified to user by utilizing Firebase through using Cloud messaging protocol. After clicking on the notification bar the user is directly moved in notification window in the user domain.

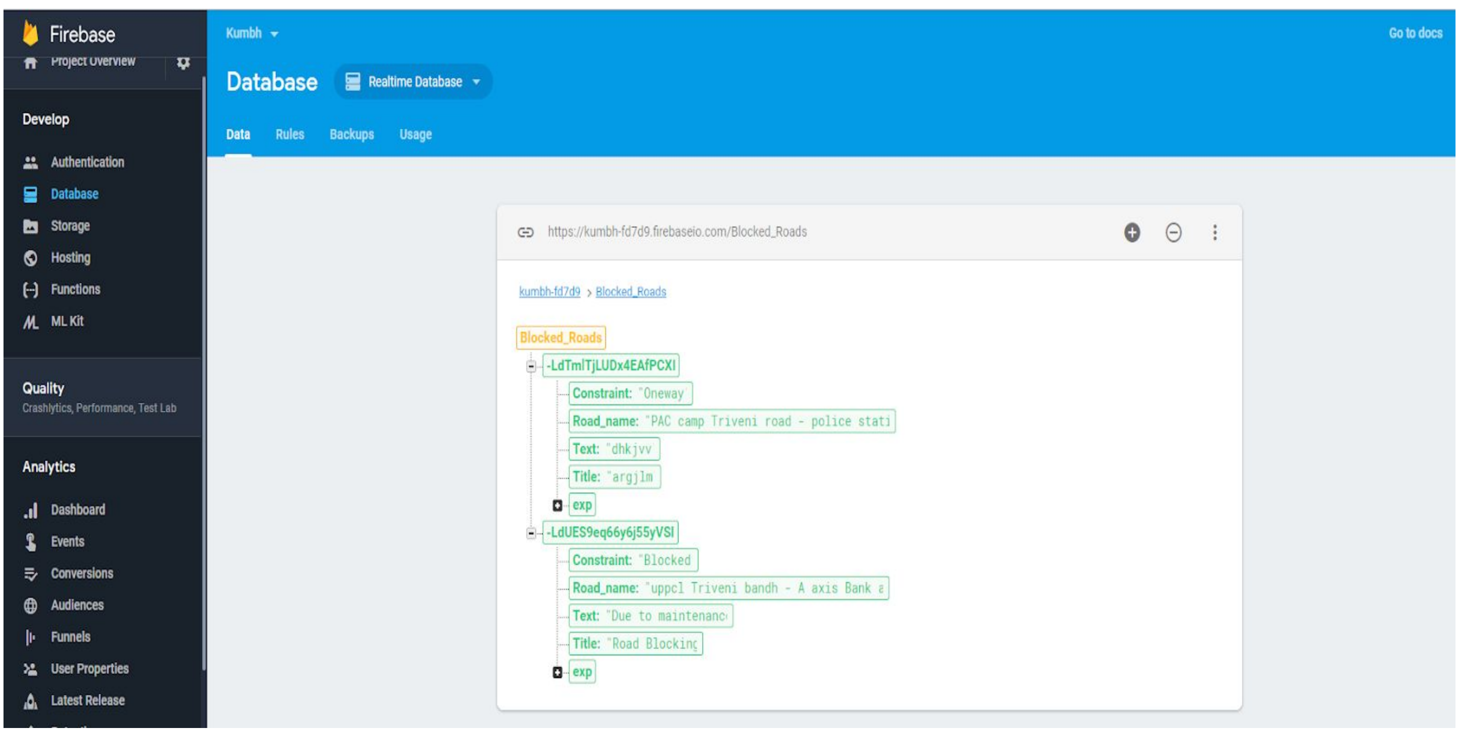
6.2.2. Firebase

This the online application providing server access and management for any mobile application usage and user engagement. We have used Firebase for login/signup data management and notification sending protocol. There we are storing user login user-id and password during sign up and during login the authentication is checked to enter the user to the android application. There are several services offered by firebase, some of which we have used, which are :

- **Cloud Messaging:** This is a part of Firebase protocol which has been used for providing instant notification to the application. After each specific entry of information to the firebase database from the admin side an instant notification is generated over the users mobile as “Stay Alert!!”.

- **Firebase Auth :** Firebase Auth is a service that can authenticate users using only client-side code. It supports social login providers Facebook, GitHub, Twitter and Google (and Google Play Games). Additionally, it includes a user management system whereby developers can enable user authentication with email and password login stored with Firebase.

- **Realtime database :** Firebase provides a realtime database and backend as a service. The service provides application developers an API that allows application data to be synchronized across clients and stored on Firebase cloud. The REST API uses the Server-Sent Events protocol, which is an API for creating HTTP connections for receiving push notifications from a server. Developers using the realtime database can secure their data by using the company's server-side-enforced security rules.



Real Time Database

7. Hardware and Software Requirements

7.1. Software Requirements

7.1.1. Android Studio
compileSdkVersion 28
buildToolsVersion '27.0.3'
minSdkVersion 18
targetSdkVersion 22
versionCode 24
Google Firebase Account

7.2. Additional Libraries Requirements

7.2.1. For implementing map functionality the required plugin of 'com.google.android.gms:play-services-maps:15.0.1', 'com.google.firebase:firebase-database:16.1.0', 'com.google.firebase:firebase-messaging:17.6.0', and 'com.google.android.gms:play-services-location:15.0.1' to access the Google API services for both Location and Map respectively.

7.2.2. In designing the GUI of the App, Relative, Linear, Constraint and various other types of layouts and Views are used .

7.2.3. For implementing Material Designing Features 'com.github.navasmdc:MaterialDesign:1.+@aar' is imported.

7.3. Hardware (For Android Studio and App)

7.3.1.1. Android Studio (RAM 4GB) SmartPhone (RAM 2GB)

7.3.1.2. Android Studio (processor i3 6 gen or higher) SmartPhone (modern day processor)

7.4. Operating System

7.4.1.1. Android Studio - Ubuntu 16.04 (64 bit) or Windows 10 64 bit SmartPhones - Android version 6 or higher(upto 8 for full functionality)

8. Activity Breakdown Chart

S.I.	TIMELINE / MODULES	Jan	Feb	March	April
1	Resaerch Papers Review				
2	Experimental Setup and dataset collection				
3	Design and Coding				
4	Final Testing and Report Preparation				

9. Conclusions, Outcome and Possible Future work :

- In this app, Implemented constraints are related to roads. Though, this app is designed by keeping in the mind that it'll be owned by government which have all information about the roads but they have no online system for broadcasting this news. Using this system, They can easily notify which road is open blocked or one way for desired time.
- The future scope for this application can be some cases like, at some location, if government gets an information about catching fire or planted bomb so they can easily notify all users through this app.
- On the hand of user side, User can rate a location(pandals, shops, ghats) and give some feedback.
- If somewhere riot got happened, so any user can notify admin and admin can confirm that news and further notify to all users.
- This application can be easily integrated to many industrial areas like IT industry locations, film industry locations, temporal fair locations where bunch of offices/shops are located in any specific area where google maps have not much information about that area.

10. References

[1] Citywide estimation of traffic dynamics via sparse GPS traces [IEEE Intelligent Transportation Systems July 2017] DOI: 10.1109/MITS.2017.2709804

[2] Accurate Real-time Map Matching for Challenging Environments [IEEE Transactions on Intelligent Transportation Systems (vol. 18, no. 4, pp. 847-857, April 2017)] DOI: 10.1109/TITS.2016.2591958

[3] Unambiguous, Real-Time and Accurate Map Matching for Multiple Sensing Sources [2018 14th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)] DOI: 10.1109/WiMOB.2018.8589103

[4] Dynamic route planning with real-time traffic predictions [Information Systems Elsevier (vol. 64, pp. 258-265, 2017)] DOI: 10.1016/j.is.2016.01.007

[5] IF-Matching: Towards Accurate Map-Matching with Information Fusion [IEEE Transactions on Knowledge and Data Engineering (vol. 29, no. 1, pp. 114-127, 2017)] DOI: 10.1109/TKDE.2016.2617326

[6] A Real-Time Range Indicator for EVs Using Web-Based Environmental Data and Sensorless Estimation of Regenerative Braking Power [IEEE Transactions on Vehicular Technology (Volume: 67, Issue: 6, June 2018)] DOI: 10.1109/TVT.2018.2829728

[7] Fast Online Map Matching for Recovering Travelling Routes from Low-Sampling GPS Data [2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)] DOI:10.1109/SmartWorld.2018.00165

[8] Comfort route navigation providing high communication Quality and Energy Saving for mobile devices [2014 IEEE 3rd Global Conference on Consumer Electronics (GCCE)] DOI: 10.1109/GCCE.2014.7031167 https://sci-hub.tw/10.1109/GCCE.2014.7031167

[9] A Real-Time Embedded System for Monitoring of Cargo Vehicles, Using Controller Area Network (International Journal of Research)

[10] Shortest path and vehicle trajectory aided map-matching for low-frequency GPS data [Transportation Research Part C: Emerging Technologies Elsevier (vol. 55, pp. 328-339, 2015)] DOI: 10.1016/j.trc.2015.02.017

[11] A comparison and evaluation of map construction algorithms using vehicle tracking data [GeoInformatica (vol. 19, no. 3, pp. 601-632, July 2015)] DOI: 10.1007/s10707-014-0222-6

[12] Scalable, Distributed, Real-Time Map Generation [IEEE Pervasive Computing (Volume: 5 , Issue: 4 , Oct.-Dec. 2006)] DOI: 10.1109/MPRV.2006.83

[13] Automatic generation of topological indoor maps for real-time map-based localization and tracking 2011 International Conference on Indoor Positioning and Indoor Navigation] DOI: 10.1109/IPIN.2011.6071951

[14] Probabilistic Multimodal Map Matching With Rich Smartphone Data [Journal of Intelligent Transportation Systems (vol. 19, no. 2, pp. 134-148, 2015)] DOI: 10.1080/15472450.2013.764796

[15] Real-time traffic congestion detection and optimal path selection using smartphone [2014 6th IEEE Power India International Conference (PIICON)] DOI: 10.1109/POWERI.2014.7117645