SMART VEHICLE TRACKING APPLICATION

Report submitted for the partial fulfillment of the requirements for the degree of Bachelor of Technology in
Information Technology

Submitted by

Ritaman Baral
(IT2014/008)
(Univ. Roll: 11700214056)

Rajdip Das
(IT2014/016)
(Univ. Roll: 11700214054)

Somyadeep Chowdhury
(IT2014/051)
(Univ. Roll: 11700214067)

Under the Guidance of Dr. Abhijit Das

RCC Institute of Information Technology
Canal South Road, Beliaghata, Kolkata – 700 015
[Affiliated to Maulana Abul Kalam Azad University of Technology]

Acknowledgement

We would like to express our sincere gratitude to Dr. Abhijit Das of the department of Information Technology, whose role as project guide was invaluable for the project. We are extremely thankful for the keen interest he took in advising us, for the books and reference materials provided for the moral support extended to us.

Last but not the least we convey our gratitude to all the teachers for providing us the technical skill that will always remain as our asset and to all non-teaching staff for the gracious hospitality they offered us.

Place: RCCIIT, Kolkata

Ritaman Baral

Date:14.05.2018

Rajdip Das

Somyadeep Chowdhury

Department of Information Technology
RCCIIT, Beliaghata,
Kolkata – 700 015,
West Bengal, India

Approval

This is to certify that the project report entitled “Smart Vehicle Tracking Application” prepared under my supervision by Ritaman Baral (IT2014/008), Rajdip Das (IT2014/016), Somyadeep Chowdhury (IT2014/051), be accepted in partial fulfillment for the degree of Bachelor of Technology in Information Technology.

It is to be understood that by this approval, the undersigned does not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn thereof, but approves the report only for the purpose for which it has been submitted.

……………………………………………..………..
Dr. Abhijit Das (HOD of Information Technology),
Project Guide

Certificate Of Acceptance

The report of the Project titled “Smart Vehicle Tracking Application” by Ritaman Baral (IT2014/008), Rajdip Das, (IT2014/016), Somyadeep Chowdhury (IT2014/051) 8th Semester of 2018 has been prepared under our supervision for the partial fulfillment of the requirements for B Tech (IT) degree in Maulana Abul Kalam Azad University of Technology.

Name of the Examiner   Signature with Date

1. ……………………………………….
   …………………………….……….

2. ………………………………………..
   …………………………………….

3. ……………………………………
   ………………………………….

INDEX

Content Index:

1. Table Index

2. Picture Index

3. Introduction

Nowadays, they are a lot of varieties application in our life. One of the applications is vehicle tracking system. Vehicle tracking system using GPS and Android based smart phone is a device that can track vehicle at anywhere. Basically, this project aims to develop vehicle tracking system using GPS and Android based smart phone that will be more useful for users and more reliable. To make life more easily, vehicle tracking system is invented. This project uses smart phone and personal computer which is one to monitor the vehicle location and another one act as a tracker. This system uses android mobile phone as mobile terminal because it is more convenient and flexible. Android mobile phone will link to the web server and write the information. Thus, user can retrieve the information for further action. GPS is used to provide a very accurate location, time, and date. The satellite will transmit the information to the GPS so that GPS can receive the information. By measuring the distance from satellite, it allows the data to control the location.

Our vehicle tracking system deals with the use of android application that automatically collects vehicle location information from individual vehicles. Basically, it collects fleet data from various vehicle locations. This tracking system generally use Global Positioning System or GPS Technology to locate each vehicle.

With changing times, the mobile technology has changed a lot in the last few years. Google’s Android is one of the latest and unique innovations, which instantly has taken over the mobile market.

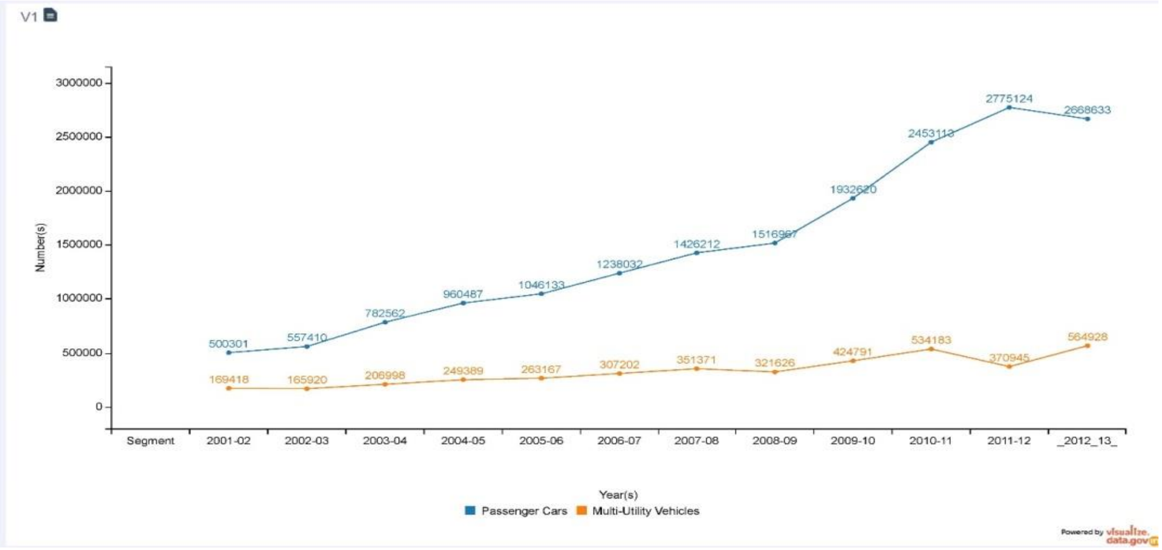
Vehicle Tracking application can be immensely useful for large and populated cities.

4. Problem Defination

Problem statement of this project is once the vehicle is being stolen, owner cannot track their vehicle. They do not know where to find the vehicle and the chances to get their vehicle back is very low. It makes so hard to track the vehicle when there is no evidence. So this system will track and monitor the vehicle’s location. But how can the location of a vehicle be tracked using Android application?

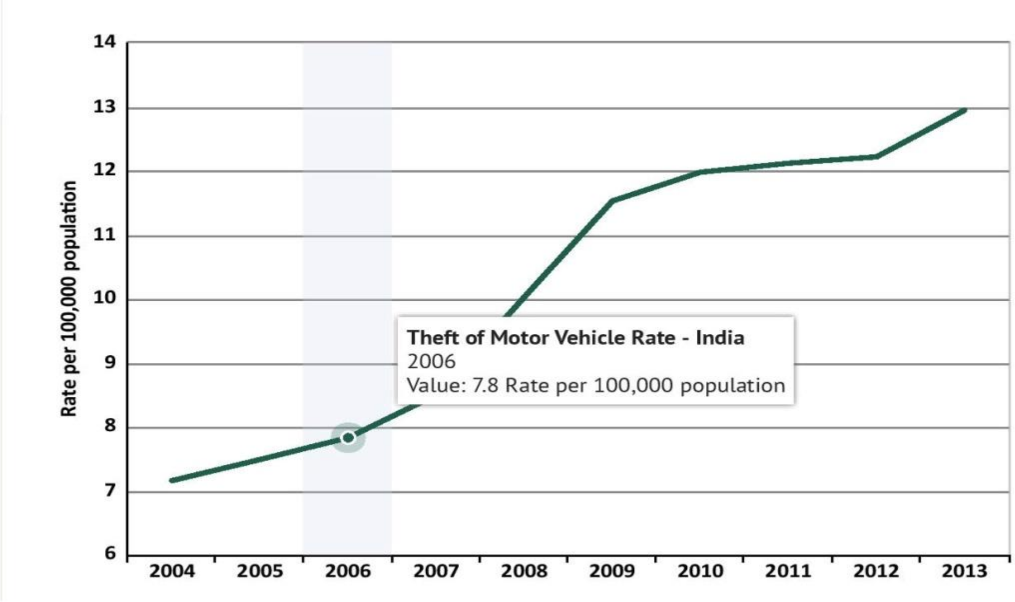
For this time being, vehicle tracking system has been nominated as one of the best tracking system for the losses of vehicle. Generally, GPS tracking uses a system of satellites orbiting earth to find the actual location. So by improving this system, it uses GPS and Android based smart phones.

If we consider India itself, about 8 million vehicles are produced annually in the country today. In 2009, the country reported 121.63 million registered motor vehicles, a motorization rate of 22 vehicles per 1000 population (Road Transport Yearbook, 2008).



(Figure 1: The growth of number of passenger cars in India over the years from 2001-2013.)

According to data.gov.in, the production of passenger cars in India was 500301 in 2001-02. It has increased to 2668633 in 2012-13, registering an absolute growth of 2168332 passenger cars. In percentage terms, the growth was 433.4%. Thus the compound annual growth rate is 16.4%. The maximum growth rate of 40% was recorded in year 2003-04. With such tremendous increase in the numbers of vehicles, the crimes involves with them have also increased over the years. Issues such as car theft have also increased.



(Figure 2: The trend of passenger vehicle theft in India per 100,000 population.)

5. Literature Survey

For vehicle tracking many designs that have been proposed and implemented in the case of implementation or in the case of the system design all proposed methods and implementation are unique. The real time vehicle monitoring system GPS module is installed on the vehicles for the transmission of real time location of individual vehicle to receiver boards which is installed on the vehicle.

This android application includes two modules:

- Admin Module
- Client Module

Admin Module

- Admin Login: Admin will Login into his firebase project account.
- View Vehicle and Driver Location:- Admin will get updated about the current location of the vehicle.

- Registration: Admin will get to know about the trackers details from firebase account will register .
- Only admin can get to know about the current location of the application user.

Client Module

- Client Login: User can login with userID and password.
- Vehicle Tracking: System will track location of both vehicle and driver using Google Maps .
- Client Registration: Client has to register themselves with Name , email id and password.
- Send Location Details: System will automatically update the individual vehicle location using Google maps into his firebase account.
- Client Details: Admin will be able get about the current location of the user in his firebase account.

Features of this Project:

- Real-time vehicle tracking through Google maps.
- Vehicle location and status tracking on web-based GUI application using API key.
- Graphical display of information including vehicle status, location.
- Implementation of Global Positioning System(GPS).
- User friendly interface for easy access and personal use.

Firebase:

Firebase is a mobile and web app development platform that provides developers with a plethora of tools and services to help them develop various applications.

Brief history: Back in 2011, before Firebase was Firebase, it was a startup called Envolve. As Envolve, it provided developers with an API that enabled the integration of online chat functionality into their website.

What's interesting is that people used Envolve to pass application data that was more than just chat messages. Developers were using Envolve to sync application data such as a game state in real time across their users.

This led the founders of Envolve, James Tamplin and Andrew Lee, to separate the chat system and the real-time architecture. In April 2012, Firebase was created as a separate company that provided Backend-as-a-Service with real-time functionality. After it was acquired by Google in 2014, Firebase rapidly evolved into the multifunctional behemoth of a mobile and web platform that it is today.

Real-time Database

The Firebase Real-time Database is a cloud-hosted NoSQL database that lets you store and sync between your users in real-time.

The Real-time Database is really just one big JSON object that the developers can manage in real-time.

We are using Firebase because:

Firebase provides a real-time 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's cloud. The company provides client libraries that enable integration with Android, iOS, JavaScript, Java, Objective-C, swift and Node.js applications. The database is also accessible through a REST API and bindings for several JavaScript frameworks such as AngularJS, React, Ember.js and Backbone.js.

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 real-time database can secure their data by using the company's server-side-enforced security

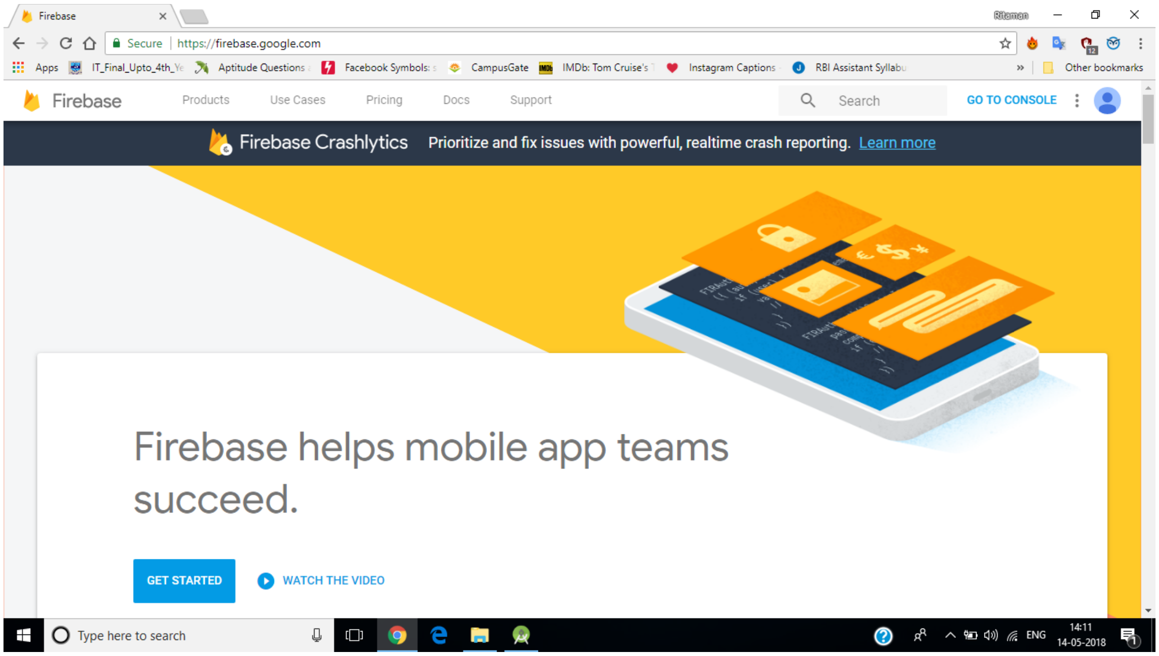
Rules. Cloud Firestore which is Firebase's next generation of the Real-time Database was released for beta use.

When building a project, if we desire one of the three following things, we'll typically consider it a good time to use Firebase.

We want short development time. It doesn't matter if it's a prototype or something we just want done tonight. If we want it fast, we know we can use Firebase to cut down development time and avoid messing with servers and data storage.

We want my data in real-time. Any time we deal with real-time data synchronization in our applications, it slows development down and forces me to draw my focus away from the application. Firebase fixes this and allows me to make anything we want work in real-time without a second thought. This is huge to us.

We want my application to scale. This is pretty straight forward. If we want my application to scale well, we can trust that Firebase will handle all my data without missing a step.



Prioritize and fix issues with powerful, realtime crash reporting.

Firebase helps mobile app teams succeed.

(Figure 3: Outlook of Firebase)

ANDROID STUDIO:

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. It is available for download on Windows, macOS and Linux based operating systems. It is a replacement for the Eclipse Android Development Tools (ADT) as primary IDE for native Android application development.

Android Studio was announced on May 16, 2013 at the Google I/O conference. It was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0. The current stable version is 3.1 released in March 2018.

Gradle:

In Android Studio, Gradle is a custom build tool used to build android packages (apk files) by managing dependencies and providing custom build logic.

Advantages:

- This system helps admin to keep track of the driver so that driver cannot do any type of cheating.
- This system helps admin to keep record of user details and able to get every details related to drivers movement using Firebase.
- This application enables admin for an easy and hastle free tracking of vehicle from anywhere.
  .

Disadvantages:

- If there is network failure due to environment hazardous, system will fail to track location of the vehicle.
- Once the user or driver log out of their account, admin won't be able to track their details.
- .If the location of the mobile is turned off intentionally admin will not be able to track the location.

6. SRS(Software Requirement Specification)

Software Requirements:

Software Components	**Version Number**
Microsoft Windows	XP and Above
Firebase	-
Vysor	-
Microsoft Andriod Studio	3.0

(**Table 3:** Softwware requirements)

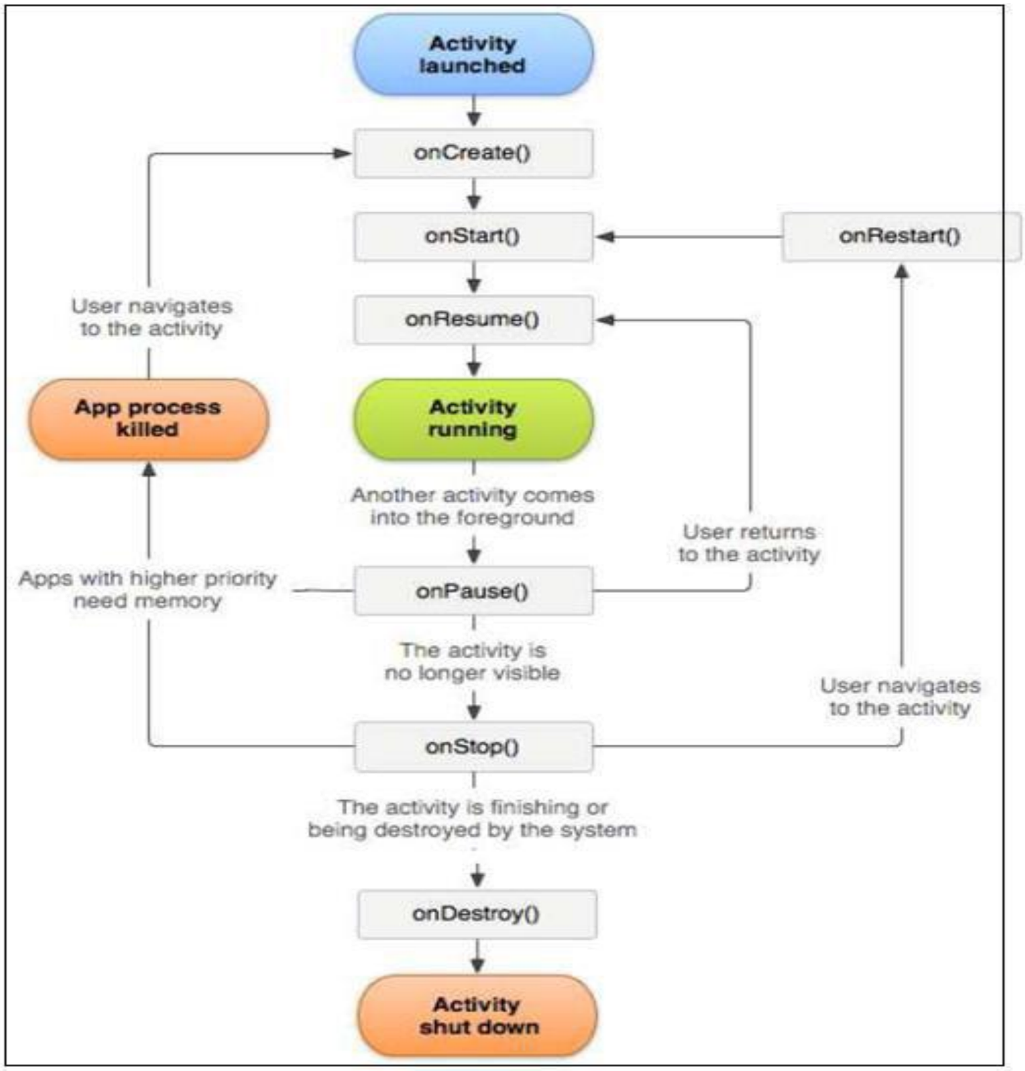
Hardware Requirement:

Hardware components	Minimum criteria
Processor	i3
HardDisk	500 GB
Memory	4 GB RAM

(**Table 3**: Hardware Requirements)

7. Design

7.1. Andriod Application Life Cycle


Activity launched
onCreate()
onStart()
onRestart()
User navigates to the activity
onResume()
App process killed
Activity running
Another activity comes into the foreground
User returns to the activity
Apps with higher priority need memory
onPause()
The activity is no longer visible
User navigates to the activity
onStop()
The activity is finishing or being destroyed by the system
onDestroy()
Activity shut down

(**Figure 4**: Andriod Application Life Cycle)

The above figure defines:

- Starting State: When an activity does not yet exist in memory, it is in the starting state.

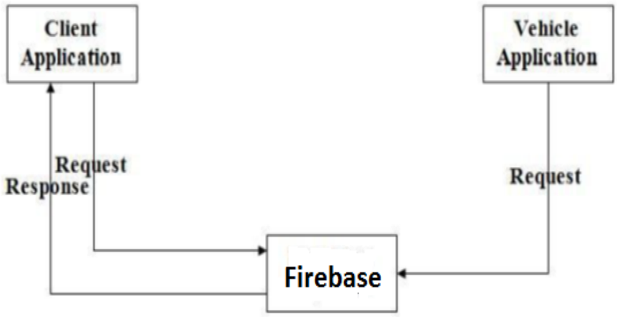
- Resumed/Running State: An activity that is in the foreground is in the running state. Any activity that is currently on the screen and interacting with the user is the running activity at that particular point in time. It exists at the top of the Activity stack.

- Paused State: When an activity is not in focus (i.e. not interacting with the user), but is still visible on the screen, it is in the Paused state.

- Stopped State: An activity that is not visible on the screen, but exists in the memory is in the Stopped State.

- Destroyed State: A Destroyed activity results from the removal of an activity (that is no longer required) from the memory. Such removals generally occur, when the activity manager decides that there is no use for such activities anymore.

7.2. System Design



(Figure 5: Typical Architecture)

Above figure shows the typical architecture of the system, it shows how system works and what phases of system are.

The major content of this architecture are following:

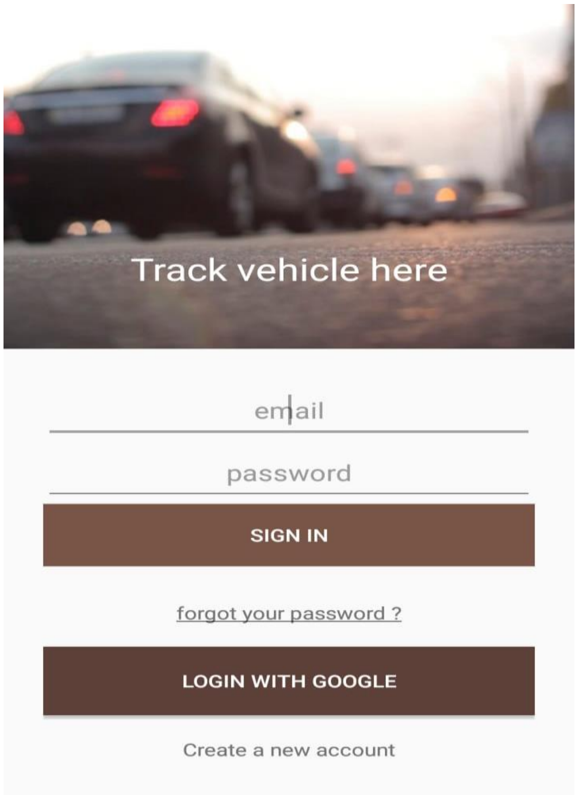
- User Interface (Android Application)
- Admin Firebase

8. Result and Discussions

8.1. Results

After successful implementation of Vehicle Tracking System we obtained following results:

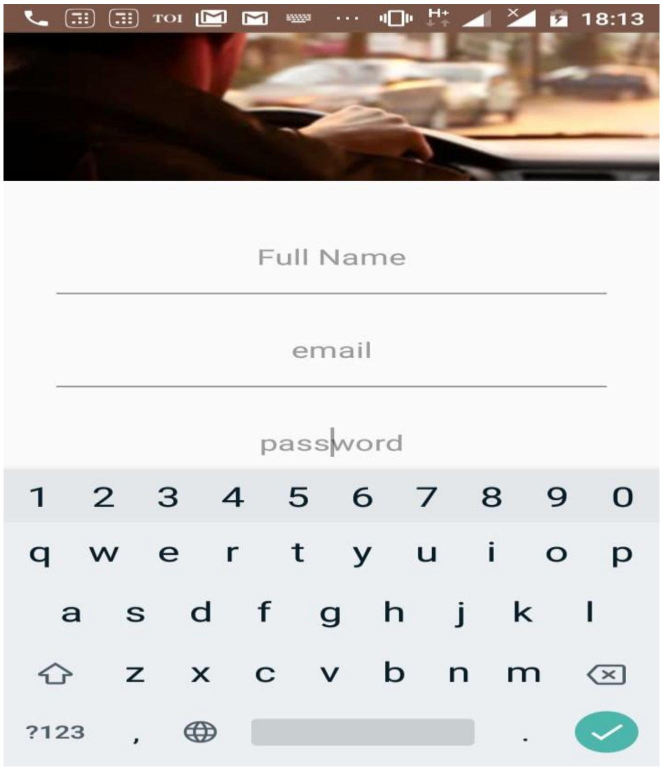
Step 1: At monitoring side, initially user needs to perform Login activity. Login page is shown in Figure 5. It provides Login interface to the user.



(Figure 6: Login Page)

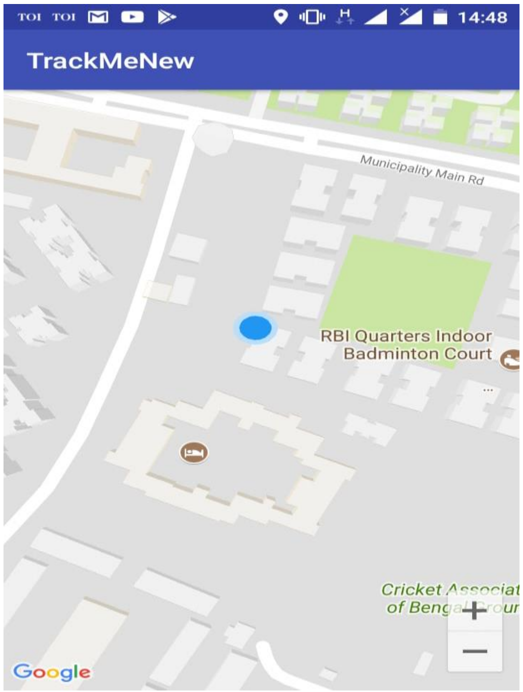
Step 2: When user will enter user name and password then system will do validation to check whether the entered username and password is correct or not. If the entered username or password is wrong then system gives an error message. And if it is correct then user gets directed to next page with successful login.

Step 3: New users can register themselves using their email account. New registration layout is shown in the figure below.



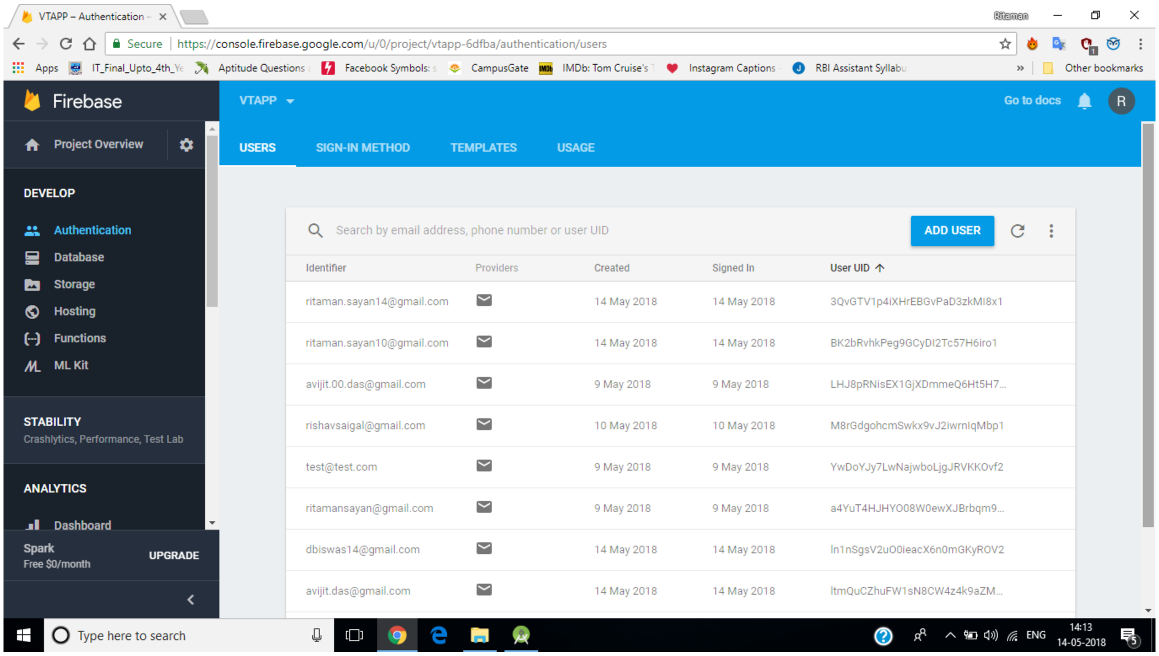
(**Figure 7**: Sign Up page)

Step 4: After successful login,the current location of that body will be shared respectively by using GPS system. Live location sharing page is shown in the the figure.



(Figure 8: Live location is being tracked.)

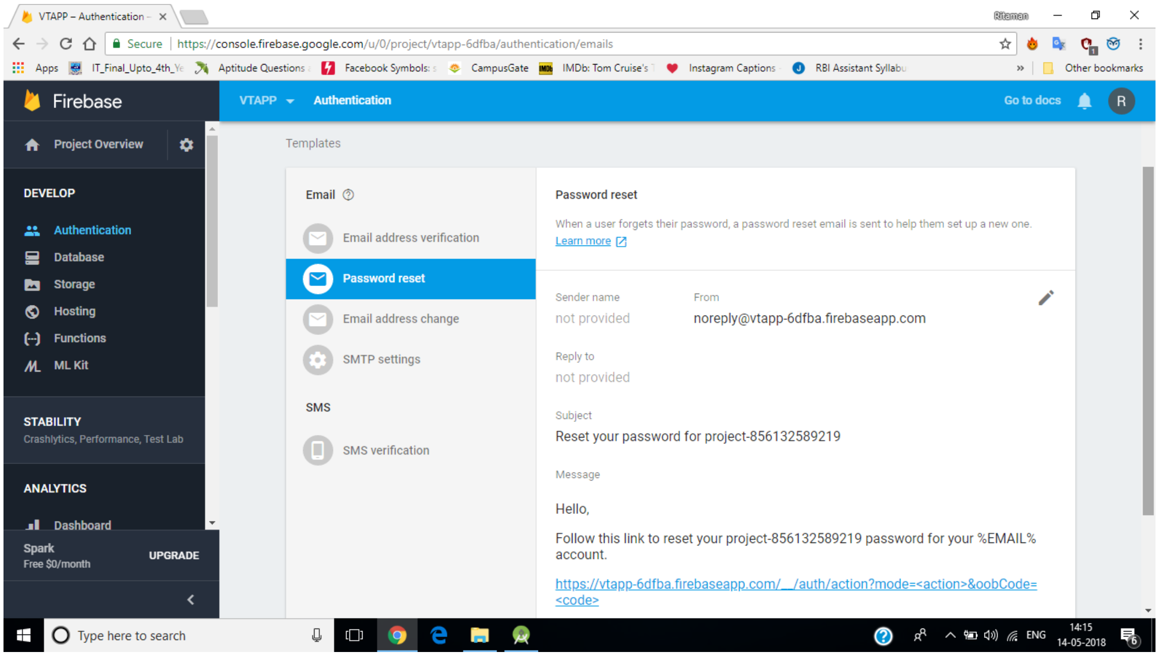
- This project has Firebase which store the location details from the application.



Identifier	Providers	Created	Signed In	User UID ↑
ritaman.sayan14@gmail.com	✉	14 May 2018	14 May 2018	3QvGTV1p4iXHrEBGvPaD3zkMI8x1
ritaman.sayan10@gmail.com	✉	14 May 2018	14 May 2018	BK2bRvhkPeg9GCyDI2Tc57H6iro1
avijit.00.das@gmail.com	✉	9 May 2018	9 May 2018	LHJ8pRNisEX1GjXDmmeQ6Ht5H7...
rishavsaigal@gmail.com	✉	10 May 2018	10 May 2018	M8rGdgohcmSwkx9vJ2iwrnIqMbp1
test@test.com	✉	9 May 2018	9 May 2018	YwDoYJy7LwNajwboLjgJRVKKOvf2
ritamansayan@gmail.com	✉	9 May 2018	9 May 2018	a4YuT4HJHYO08W0ewXJBrbqm9...
dbiswas14@gmail.com	✉	14 May 2018	14 May 2018	In1nSgsV2uO0ieacX6n0mGKyROV2
avijit.das@gmail.com	✉	14 May 2018	14 May 2018	ltmQuCZhuFW1sN8CW4z4k9aZM...

(Figure 9: Client details stored in Firebase)

- If client forgets their password, they can recover their password using Password Recovery Option.



Password reset

When a user forgets their password, a password reset email is sent to help them set up a new one. Learn more

Sender name: not provided

From: noreply@vtapp-6dfba.firebaseapp.com

Reply to: not provided

Subject: Reset your password for project-856132589219

Message

Hello,

Follow this link to reset your project-856132589219 password for your %EMAIL% account.

https://vtapp-6dfba.firebaseapp.com/__/auth/action?mode=<action>&oobCode=<code>

(Figure 10: Password recovery page)

8.2. Discussions

Firstly, we need the location information of a certain vehicle and the technologies used are:

- Global Positioning System-GPS: It is the technology where satellites send down radio signals in which GPS units and receivers use to work out their current location (which is shown by latitude, longitude and elevation).There are 24 working satellites circling the globe at any given moment. A GPS navigator or GPS tracker searches for the transmission signal from at least three satellites.
- Client-Server technology: Client will ask for the location of the vehicle through their android devices. Request to the server is made automatically.
- Vehicle will be equipped with GPS device. Through GPS it will find its location and deliver it to Server.
- Server will handle the locations and deliver it to the intended client.

Secondly, we will map the location using Google-maps to show the map

Finally, we store the information of the vehicle in our database. For this purpose, we use Firebase. Vehicle in-charge will update all information regarding their routes. And, when the client request for vehicle information, it will update the location data into the admins firebase project account.

9. Conclusion and Future Scope

9.1. Conclusion

Vehicle tracking system resulted in improving overall productivity with better fleet management that in turn offers better return on your investments. Better scheduling or route planning can enable you handle larger jobs loads within a particular time. Vehicle tracking both in case of personal as well as any other purpose improves safety and security, communication medium, performance monitoring and increases productivity. This system has many advantages such as large capability, wide areas range, low operation costs, effective, strong expandability and easy to use in vehicle traffic administration.

Overall, the development of the system was a success. Some extra features were included into it to convince its flexibility. Advanced improvements could be made by registering the application in the Google play store so that it doesn't ask the user for permission to use.

- The current vehicle tracking software and graphical user interface configuration allows a user to remotely login to the system and monitor or track vehicles. The security feature of the system is enhanced with the user login requirements and other credentials. Thus the main objective of this project to design and construct a cost effective system have been achieved.

- The main limitation of the software is the real time implementation. This can't be implemented with this much time efficiency in any of the real time applications. This limitation is mainly due to the processing time. We are still in our testing phase and analyzing its strength against different modes.

- The designed system could be applied anywhere due to its ease of usage and effectiveness, which could be a future scope. We believe that this simple concept, when released into the world, will be one of those revolutionary changes in everyday activities and will become even more popular.

- ➢ This application is under beta mode as the application has not yet got the google play store authentication certification.

9.2. Future Scope

- ➢ In near future, this application can be modified by adding the ability to share the track and other related information obtained via SMS, Bluetooth or mail or via Chat Messenger(“WhatsApp”) and other social networking site(“Facebook, Twitter”).
- ➢ User friendly and simple interface provides an ideal compatibility with any vehicle tracking approach in real life.

10.Bibliography

- Data.gov.in for all the graphs and statistics
- Google.co.in
- Wikipedia.org
- abhiandroid.com
- www.tutorialpoints.com
- https://www.youtube.com/watch?v=qS1E-Vrk60E&t=363s
- https://www.youtube.com/watch?v=7SJsnPYkYqo&t=217s

11.Appendix

ACTIVITY MAIN:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
android:orientation="vertical"
    android:weightSum="5"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:layout_weight="0.90"
        android:textColor="@color/white"
        android:paddingTop="150dp"
        android:textSize="25sp"
        android:background="@drawable/a1"
        android:text="Track vehicle here"
        android:layout_gravity="center"
        android:gravity="center"
        />

    <!--<ImageView
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:src="@drawable/a1"
        android:layout_weight="0.90"
        android:scaleType="centerCrop"/> -->
```

```
  <LinearLayout
      android:paddingTop="25dp"
      android:paddingLeft="25dp"
      android:paddingRight="25dp"
      android:layout_weight="2"
      android:layout_width="match_parent"
      android:orientation="vertical"
      android:weightSum="3"
      android:layout_height="0dp">
      <EditText
          android:inputType="text"
          android:id="@+id/email"
          android:layout_weight="1"
          android:layout_width="match_parent"
          android:layout_height="0dp"
          android:hint="email"
          android:nextFocusDown="@id/pass"
          android:layout_gravity="center"
          android:gravity="center"/>
      <EditText

          android:id="@+id/pass"
          android:layout_width="match_parent"
          android:layout_height="0dp"
          android:hint="password"
          android:layout_weight="1"
          android:inputType="textPassword"

          android:layout_gravity="center"
          android:gravity="center"/>
      <Button

android:onClick="SignIn"
          android:layout_weight="1"
          android:background="#795548"
          android:textColor="@android:color/white"
          android:layout_width="match_parent"
          android:layout_height="0dp"
          android:text="Sign In"/>
  </LinearLayout>

    <LinearLayout
        android:paddingTop="10dp"
        android:layout_weight="2"
        android:weightSum="3"
        android:paddingRight="25dp"
        android:paddingLeft="25dp"
        android:orientation="vertical"
        android:layout_width="match_parent"
        android:layout_height="0dp">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            android:layout_gravity="center"
            android:gravity="center"
            android:text="@string/fg"/>
        <Button
            android:layout_weight="1"
            android:background="#5D4037"
            android:layout_width="match_parent"
            android:textColor="@android:color/white"
            android:layout_height="0dp"
            android:text="Login with Google"/>
        <TextView
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            android:layout_gravity="center"
            android:gravity="center"
android:onClick="Register"
            android:text="Create a new account"/>

    </LinearLayout>

</LinearLayout>
```

ACTIVITY SIGNUP:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:weightSum="5"
  >

    <TextView
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:layout_weight="0.10"
        android:textColor="@color/white"
        android:paddingTop="150dp"
        android:textSize="25sp"
        android:background="@drawable/ddd"

        android:layout_gravity="center"
        android:gravity="center"
        />

    <!--<ImageView
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:src="@drawable/a1"
        android:layout_weight="0.90"
        android:scaleType="centerCrop"/> -->


    <LinearLayout
        android:paddingTop="25dp"
        android:paddingLeft="25dp"
        android:paddingRight="25dp"
        android:layout_weight="5"
        android:layout_width="match_parent"
        android:orientation="vertical"
        android:weightSum="0"
        android:layout_height="0dp">
    <EditText
      android:inputType="text"
      android:nextFocusDown="@id/reg_email"
      android:id="@+id/reg_name"
      android:layout_weight="1"
      android:layout_width="match_parent"
      android:layout_height="0dp"
      android:hint="Full Name"
      android:layout_gravity="center"
      android:gravity="center"/>
    <EditText
      android:inputType="text"
      android:nextFocusDown="@id/reg_pass"
      android:id="@+id/reg_email"
      android:layout_weight="1"
      android:layout_width="match_parent"
      android:layout_height="0dp"
      android:hint="email"
      android:layout_gravity="center"
      android:gravity="center"/>
    <EditText
      android:inputType="textPassword"
      android:id="@+id/reg_pass"
      android:layout_width="match_parent"
      android:layout_height="0dp"
      android:hint="password"
      android:layout_weight="1"
      android:layout_gravity="center"
      android:gravity="center"/>

    <Button
android:onClick="RegisterUser"
      android:layout_weight="1"
      android:background="#5D4037"
      android:layout_width="match_parent"
      android:textColor="@android:color/white"
      android:layout_height="0dp"
      android:text="Sign Up"/>
    <TextView
      android:layout_width="match_parent"
      android:layout_height="0dp"
      android:layout_weight="1"
      android:layout_gravity="center"
```

```
            android:gravity="center"
            android:id="@+id/goback"
            android:text="Already have an account ? "/>
    </LinearLayout>


</LinearLayout>
```

ACTIVITY SPLASH SCREEN:

```
<?xml version="1.0" encoding="utf-8"?> <RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">

    <VideoView android:id="@+id/videoView"
        android:layout_alignParentTop="true"
        android:layout_alignParentBottom="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentRight="true"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent">
    </VideoView>
</RelativeLayout>
```

MAIN ACTIVITY JAVA:

```
package com.example.avijit.vt;

import android.app.ProgressDialog;
import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.v7.app.ActionBar;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.example.avijit.vt.R;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;

public class MainActivity extends AppCompatActivity {

    private FirebaseAuth mAuth;
    private FirebaseAuth.AuthStateListener mAuthListner;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ActionBar actionBar = getSupportActionBar();
        actionBar.hide();
        mAuth=FirebaseAuth.getInstance();
        mAuthListner=new FirebaseAuth.AuthStateListener() {
            @Override
            public void onAuthStateChanged(@NonNull FirebaseAuth firebaseAuth) {
                FirebaseUser user=firebaseAuth.getCurrentUser();
                if(user!=null){
                    Toast.makeText(getApplicationContext(),"Welcome
"+user.getEmail(),Toast.LENGTH_SHORT).show();
                    Intent intent=new Intent(getApplicationContext(),UserDetails.class);
                   // intent.putExtra("email",user.getEmail());

                    startActivity(intent);
                    finish();
                }
                else{
                    Toast.makeText(getApplicationContext(),"Please
Login!!!",Toast.LENGTH_SHORT).show();
                }
            }
        };
```

```
-
    }

    @Override
    protected void onStart() {
        super.onStart();
        mAuth.addAuthStateListener(mAuthListner);
    }

    @Override
    protected void onStop() {
        super.onStop();
        mAuth.removeAuthStateListener(mAuthListner);
    }


    public void SignIn(View view) {




        String email=((EditText)findViewById(R.id.email)).getText().toString();
        String pass=((EditText)findViewById(R.id.pass)).getText().toString();
        if (email.matches("")|| pass.matches(""))
        {
            Toast.makeText(getApplicationContext(),"enter
correctly",Toast.LENGTH_SHORT).show();
        }
        else {


            mAuth.signInWithEmailAndPassword(email, pass).addOnCompleteListener(new
OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if (task.isSuccessful()) {
                        Toast.makeText(getApplicationContext(), "Login Successfull",
Toast.LENGTH_SHORT).show();
                    } else {
                        Toast.makeText(getApplicationContext(), "Login Unsuccessfull",
Toast.LENGTH_SHORT).show();
                    }
                }
            });

        }

    }

    public void Register(View view) {
        startActivity(new Intent(this,signup.class));
        finish();
    }
}
```

ANDROID MANIFEST:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.bright.trackmenew">

    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission    android:name="android.permission.ACCESS_FINE_LOCATION"
/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity" />

        <meta-data
            android:name="com.google.android.geo.API_KEY"
            android:value="@string/google_maps_key" />

        <activity android:name=".LoginActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```