Vehicle Enhance Tracking System

A PROJECT REPORT

Submitted By

Vinayak Inamdar (160800107023)

Kartik Jain (160800107024)

Akhilesh Kushwaha (160800107030)

In partial fulfillment for the award of the degree

Of

BACHELOR OF ENGINEERING

In

COMPUTER ENGINEERING

Vadodara Institute of Engineering, Kotambi

Gujarat Technological University, Ahmedabad

DEC,2019

Vadodara Institute of Engineering, Kotambi

Computer Engineering

2019

CERTIFICATE

Date:

This is to certify that the project entitled "Vehicle Enhance Tracking System" has been carried out by Vinayak Inamdar (160800107023), Kartik Jain (160800107024), Akhilesh Kushwaha (160800107030) under my guidance in fulfillment of the degree of Bachelor Of Engineering in Computer Engineering 7th Semester of Gujarat Technological University, Ahmedabad during the academic year 2019- 2020.

Guide:

Prof. Bhavika Tailor

Professor in Computer Engineering

Prof. Ajaysinh Rathod

Head of the Department

ABSTRACT

Nowadays, if we consider a current scenario about security of a vehicle in India, then we know that if we park the two-wheeler vehicles or cars in public areas, then we have to take responsibility of its security at our own risk.

So, the goal of our project is to make the people feel risk-free about security of their vehicles especially in public or remote areas located in cities.

Hence, we have designed a solution to overcome this problem of vehicle security to provide benefit to the people.

We will be providing live tracking of vehicles to their respective owners if it is found that the authenticated vehicle pickers have took off their vehicle such as car, bike or scooty to some of the police stations. So, that the people can easily reach to that location to get their vehicles back.

We will also implement application which will provide all kind of information regarding their vehicle.

Our implementation of solution to this problem will also give notifications to the respective vehicle owners if their vehicle is found to be harmed or moved beyond certain distance limit.

ACKNOWLEDGEMENT

With a sense of gratitude and respect, we would like to extend our heartiest thanks to all those who provided help & guidance to during our college period. It was a pleasant and highly educative experience to work for project on VEHICLE ENHANCE LOCATION TRACKING SYSTEM. We are grateful to Prof. Bhavika Tailor who was kind enough to consider our choice and trust on us for providing a good project and was always ready to provide the best instruction and guidance to do work better. Many things and person help us for working on this project each of their contribution is very valuable for us.

LIST OF TABLES

LIST OF FIGURES

INDEX

1. INTRODUCTION

1.1 PROJECT SUMMARY

"Vehicle Enhancement Location Tracking System"

This is a Mobile-app in which we are attempting to achieve a totally new genre wherein we will bring Certainty at one stop for all kinds of Vehicles worldwide where Owners, Traffic Policemen, Admin members can log into it and add essential Documents, Generate Challan, Verify Documents and Payment Status, etc. Create User's profile, Verify ID Proof, save all crucial documents such as License, RC Book and also Track Live Location of the registered Vehicle.

Today many vehicle systems that are in use now days are some form of Automatic Vehicle Location. It is a concept for determining the geographic location of a vehicle and transmitting this information to a remotely located server. The location is determined using GPS and transmission mechanism could be a satellite, terrestrial radio cellular connection from the vehicle to a satellite or nearby cell tower. GSM is the most common used service for this purpose. There is no such application in existing system that can be connected with our registered vehicle through GSM Module providing facilities like Storing documents of all Vehicles, Digitalized Payments or Showing Live Location of Towed/Damaged Vehicles.

This application keeps track of records of all users about their payments for generated challan and it can be used by Admin, Traffic Policemen as well as registered users. There will be an facility for the payment of challan and monitor the vehicle location if the vehicle is moved from one place to another place or if the vehicle is being damaged by anyone in your absence .Also after creating the account, the user will be getting the password for log into the account to upload the documents like License, Pay Penalty if Challan is Generated, Give reviews and can take advantage of Vehicle Monitoring Services.

1.2 PURPOSE

The purpose of developing the application is to bring Vehicle Monitoring Service at a single platform so that everyone can use it easily without worrying about Vehicle's documents.

Another major purpose of proposed system is to monitor the vehicle, detect the damage if the vehicle is fallen and making the challan digital so that the user can easily pay the payment without any argument.

1.3 SCOPE

Scope describes the functions and features that are to be delivered to end-users, also data-input and output, contents, performance, constraints, interface and reliability.

This System will be able to Manage Vehicle Enhancement Services remotely and also maintain co-ordination among Customers and Admin.

1.4 OBJECTIVE

The basic objective of this project is to develop a system which can cater all the needs of Recent Era regarding Vehicle Enhancement at one platform.

Main Objective of Vehicle Enhancement System is to develop a system which can bring Security/Storing of all kinds of useful documents related to all types of vehicles of whole Society to make it a useful application for Customers and to provide all the Vehicle monitoring Services to the Users.

Vehicle tracking improves safety and security, communication medium, performance monitoring and it will increase productivity. So in future it will play a major role in our day-to-day living.

1.5 TECHNOLOGY AND LITERATURE REVIEW

The Key Technologies used in developing the Vehicle Enhancement and Location Tracking System are:

1. **Servers:**

   Xamp Server (Localhost)

2. **Tools:**

   Arduino IDE

   Android Studio IDE

   Microsoft Visio 2007

We have used MySQL as the underlying database. The usage of MySQL Server was done by analyzing the pros and cons of various databases existing in the market.

MySQL is a relational database management system (RDBMS), and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or use MySQL "front-ends", desktop software and web applications that create and manage MySQL databases, build database structures, back up data, inspect status, and work with data records. The official set of MySQL front-end tools, MySQL Workbench is actively developed by Oracle, and is freely available for use.

phpMyAdmin is a free and open source tool written in PHP intended to handle the administration of MySQL with the use of a web browser. It can perform various tasks such as creating, modifying or deleting databases, tables, fields or rows; executing SQL statements; or managing users and permissions.

2. System Requirement Study

2.1 User Characteristics

Customer/User:

Customer can create their account free and use our facilities provided for user. This application also has different services as follow:

- ¤ Login/Signup
- ¤ Store Government Documents
- ¤ Track the Vehicle Location
- ¤ Get notification in case of damage
- ¤ Get notification if vehicle is towed by traffic police
- ¤ Pay Traffic Challan Online

Admin:

Admin manages consumer & customer profile. Admin is responsible for managing this site or supporting the system to be worked. Admin also manages fallowing:

- ¤ Manage Websites and Application
- ¤ Manage User Blogs
- ¤ Manage User Account
- ¤ Manage vehicle information and locations
- ¤ Feedback Maintenance

2.2 Hardware and Software Requirement

NOTE: These requirements of system are minimum are minimum. So to run this application on system this is minimum required to execute application efficiently.

Software Requirements:

Application Software	:	Browser (Mozilla Firefox, Internet Explorer, Google Chrome etc.), Android Mobile of API min 17
Operating System	:	Window 2007,Window XP, Linux, Mobile

Table 2.2.1: Software Requirement

Hardware Requirements:

Graphic Card		:	Not Required
Processer		:	Intel Core i3
Minimum Memory Required	[Ram]	:	512 MB
	[Hard disk]	:	40 GB

Table 2.2.2: Hardware Requirement

2.3 Constraints

2.3.1 Regulatory Polices

It is a document that describes philosophy, principles and fundamental factors used by the system in its regulatory program. Regulatory documents are policies, standards, guides, notices, procedures and information documents support and provide information on legally enforceable instruments. These documents do not create those instruments.

2.3.2 Hardware Limitation

System integration, the process of putting hardware, software and people related to the system together for making a system. It is done incrementally so that sub systems (modules) area integrated one at a time.

Interface problem between sub system and actual system usually found at the stage of integration which causes the hardware limitation.

Problems of uncoordinated system components also lead to a hardware limitation.

2.3.3 Interface to other Applications

This constraint describes the application's relationship to other systems. For example,

- Does the application is independent?

  This application does integrate different application to provide some of the user-friendly functions.

  Like, we use Android here as designing tool of Application and php for the website etc.

- Is it a component of larger application?

Not really but it is a large application which include other application into single system which designs the best App for Enhancement of location tracking system.

- Is it replacement of another system?

  No, it does not replace any system. It just merges the different existing system into single frame.

2.3.4 Parallel Operations

This system is parallel processing so, system is parallel to update existing account and all effects will be shown as User account. E.g. If user delete an account from this system then the Tracking, Notification, addresses etc. will also deleted from that account database where it is actually stored. So, this example will clear you about parallel operation of this system.

2.3.5 Higher Order Language Requirement

In my service I have used Java as higher order language. It uses Android app for the updating in services and uses php and iot for the vehicle detection etc.

Reliability Requirements

These requirements mean to specify the numerical reliability targets for the application based on system level reliability targets and known reliability of other components.

Our application is reliable in each and every case i.e. either in erroneous code or in error free code.

2.3.6 Criticality of the Application

The main criticality of this system is if the server goes down then this system won't work.

2.3.7 Safety Security Consideration

The system is providing basic security to user's account. All users' accounts are secure by password mechanism. If any security violation will be found then at time service will be blocked. Unauthorized user cannot login to this system.

2.4 TIMELINE CHART

2.4.1 Timeline chart of the system

Table 2.4.1: TimeLine Chart

3. SYSTEM ANALYSIS

3.1 STUDY OF CURRENT SYSTEM

As we know now a day's carrying a vehicle for going to nearby areas/places like as malls, and food market has become an essential choice for all of the people, but we need to carry our vehicle at our own risk.

Sometimes we even don't know or we are not aware of about our vehicle if it gets towed/fallen or if it is damaged by some stranger.

In existing system owner use to get SMS from tracking device, which did not provide the exact location of tracked vehicle. But it didn't provide the exact location of the tracked vehicle. So as to overcome this system we have made a solution here.

As per the observation existing system provides only some features like SMS services whereas the contribution which we made is like locating our vehicle's location on Google Map and/or user will also be able to watch live streaming of their own vehicle being carried/towed away. This will enhance the user with complete information about his organization vehicle location.

3.2 PROBLEM AND WEEKNESS OF CURRENT SYSTEM

Problem facing current system:

- ¤ We are not aware of about our own vehicle location if it is towed.
- ¤ We need to carry the documents like RC Book, license, insurance, PUC of vehicle to the nearby police station.
- ¤ We even need to pay the penalty charges/fine by going physically to the police station.
- ¤ Vehicle owner/user are also tensed about their vehicle's situation and parking location.

3.3 REQUIREMENT OF NEW SYSTEM

In today's era everyone is using mobile phones for communication. At the same time Mobile Providers are also providing the variety of services to users.

In attempt to expand on this, we propose a GPS based vehicle tracking system for an organization to help to find addresses of their vehicles and locate their positions on mobile devices. The organizations are investing money in monitoring and tracking vehicles aiming at improving services and ensuring the safety in cargos transports. The proposed technology allows organizations to track real-time information about their organizational vehicle during travel. Today for local transport most people use Bus as a medium. But due to their irregularity public faces various problems like not reaching on time, bus failure, no proper schedule etc.

Looking at this scenario we have created an Android application which provides the exact location of all organizational vehicles. The system contains single android mobile that is equipped with GPS and GSM modems along with processor that is installed in vehicle.

During vehicle motion because of certain reasons like damage/towed/fallen its location update can be continuously reported to a user/vehicle driver/owner using GPRS service. This location information will be plotted using Google maps on monitoring device.

¤ **User Requirements**

User requirements include not many things, but the most important thing is user must be aware that system works properly with full availability, reliability, security and safety. The user responsibilities are as follow:

- Should know the data needed to address the problem.
- Should know how to use it
- Should adhere to guidelines and prescribed standards.
- Actors: (User)

1. Admin
2. User
3. Traffic Policemen

¤ **System Requirements**

Vehicle Enhance and Location Tracking System is mainly to be built to provide or maintain the database about customer's payment records due to the generation of challan in his/her account by the traffic police.

It also stores the documents of customer's vehicles such as Vehicle's insurance, RC Book, License and PUC.

So this, system is very useful for all the people as well as for Traffic Police for verification of documents and generating a challan for payment as it also provides facility of online payment.

3.4 FEASIBILITY STUDY

Technical issues were also raised during the feasibility study. It has been observed that technical know hoe is available in penalty and it is easy accessible to the organization.

The requirement required for the proposed system is also easy available and it has the technical capacity to hold the volume of the data required for the new system.

The system can also be expanded if the need arises. There are technical Guarantee of accuracy, reliability and ease of access of data, and data security.

Preliminary investigation examine the project Feasibility, the likelihood the system will be useful to the organization. There are three basic tests of feasibility study for computerization of a new system, and each one is equally important. These tests are as follow:

¤ **Technical Feasibility**

Technical Feasibility determines whether the work for the project be done with the present equipment, current procedure, existing software's technology and available personal?

It happens that after a system is prepared a new technology arises and the user wants the system based on that technology. Thus it is important to check the system to be technically feasible.

Here the system Global Info Source at Cell it needs big storage for dictionary of address, blogs, comments, websites, user profile etc. Also require high speed operation for user friendly application output.

To achieve these goals and to design this application we required following hardware and software:

System Platform:

Android Studio 3.0

Arduino IDE

Development Tools:

Android Studio IDE

Arduino IDE

Hardware Platform:

Power supply

GSM SIM 800

GPS

Driver IC

Adxl335

- **Economical Feasibility**

Economic feasibility looks at the financial aspects of the project. Economic feasibility concerns with the returns from the investments in project.

In this system we are using plans to acquire the necessary hardware and software require for the system and there is no economical hindrance towards its purchase. A brief description of the hardware and software required in the system is given above in the report. As this system is also providing SMS service; therefore we need to consider the aspect of recharging SMS pack service regularly.

As per this application is very useful for general people as well as for small business owners it is easy to recover the returns of investments.

- **Behavioral Feasibility**

Behavioral feasibility defines the user response to the system. It is important to concern with user who is dealing with the system.

In this system our system is usable by any of the people who are Vehicle owners or even Traffic policemen can take benefits of our system.

Our system is user friendly and also we provide easy navigation, attractive look, with useful guidance with the processes.

.

- **Implementation Feasibility**

The Implementation feasibility deals with the study whether the service, which is being developed will run in the environment available with us, will the management of the organization approve the system?

This service can run in any environment with one constraint that there is internet connection for the site to be viewed. It does not cause any harm to the running system.

After all the tests we can able to give conclusion that in terms of service like available to all, speed in processing, data management, security and secure payment transaction.

3.5 REQUIREMENTS VALIDATIONS

a. The Id no field should contain only 10 digits.
b. The Date format should be MM-DD-YYYY.
c. From date and to date must be past date.
d. To date must be greater than from date.
e. All forms id number must be 10 digits.
f. Email Id must have "." And "@" symbol.
g. Website Name must be Start with "www.".
h. Phone number must contain numeric value.
i. Fax number must contain numeric value.
j. Mobile number must be verified
k. Mobile number must contain (+) before phone number.

3.6 FUNCTIONS OF NEW SYSTEM

3.6.1 Use Case Model

Figure 3.6.1: Use Case

3.7 DATA MODELING

3.7.1 E-R Diagram

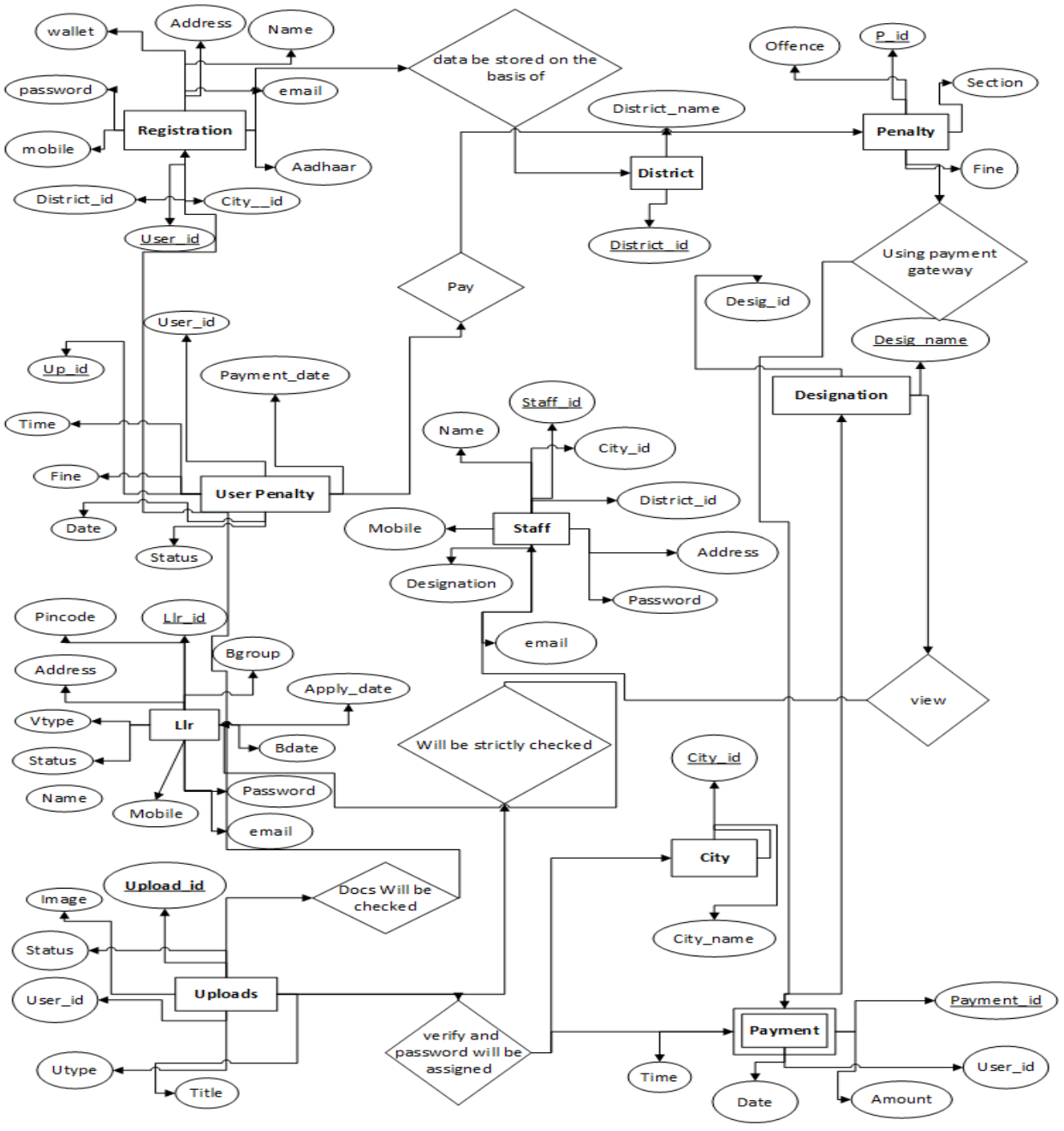


Figure 3.7.1: Class Diagram

3.7.2 Class Diagram

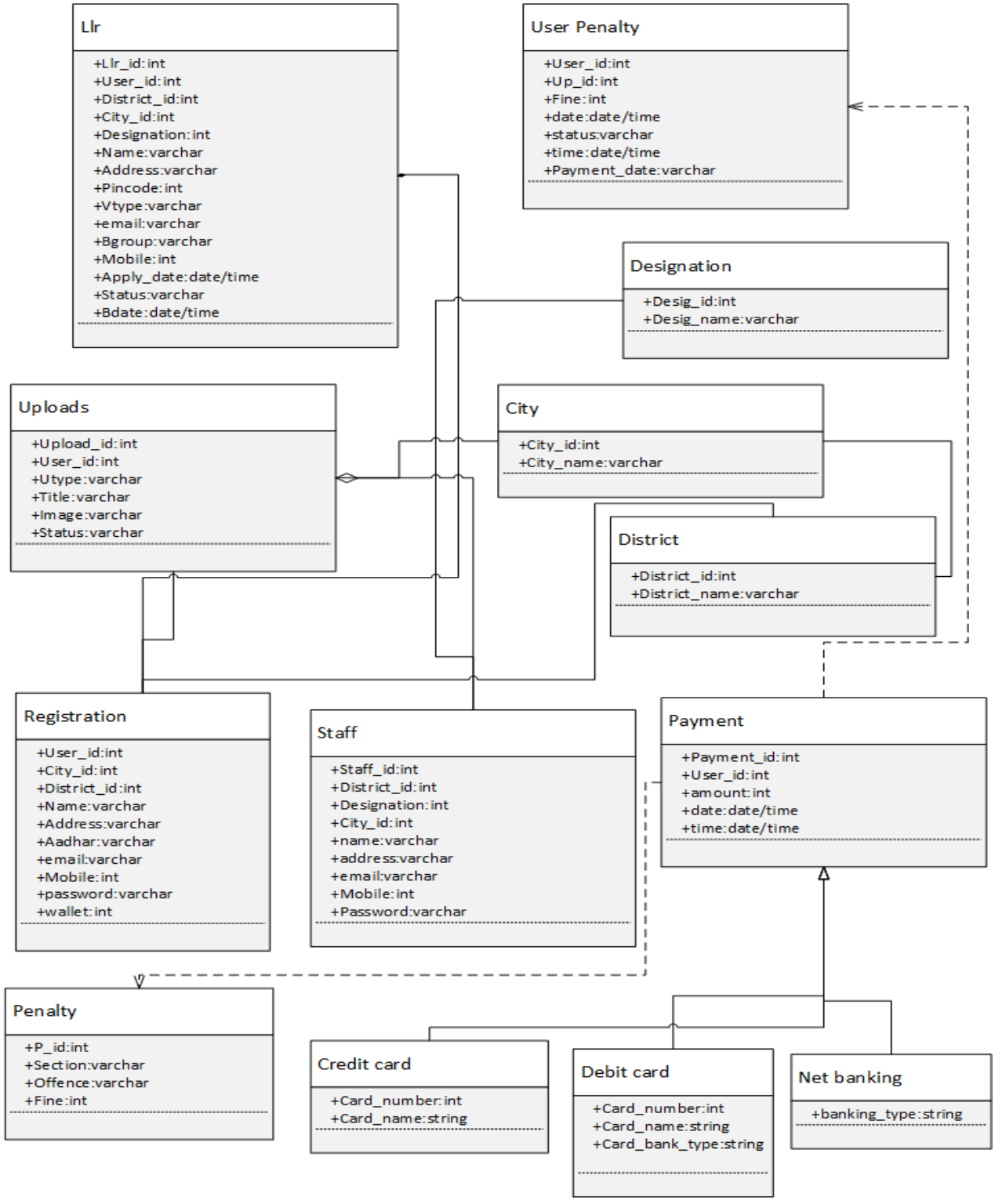


Figure 3.7.2: Class Diagram

3.7.3 System Activity

3.7.2.1 Log In

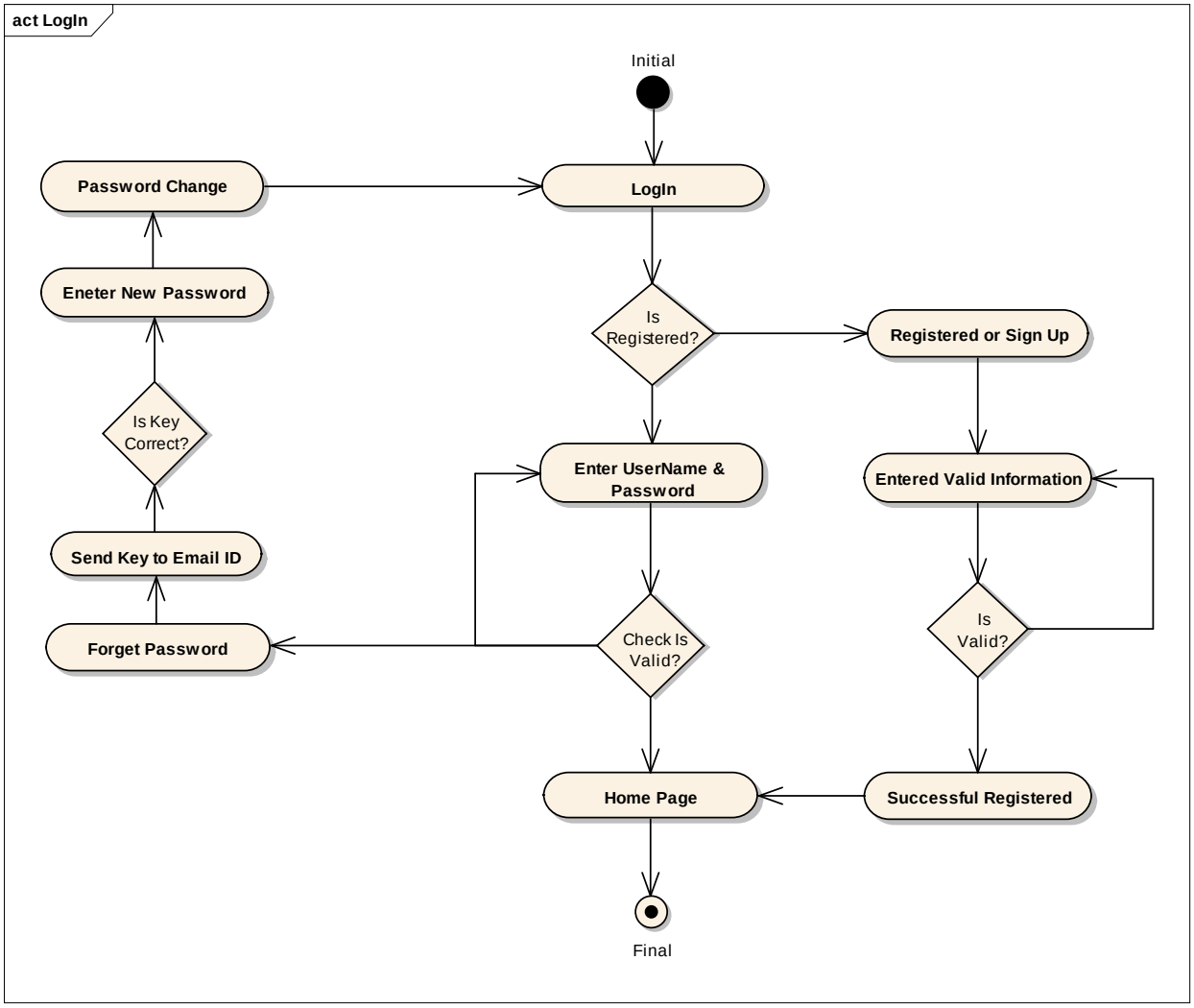


Figure 3.7.2.1 Log In Activity

3.7.2.2 Payment

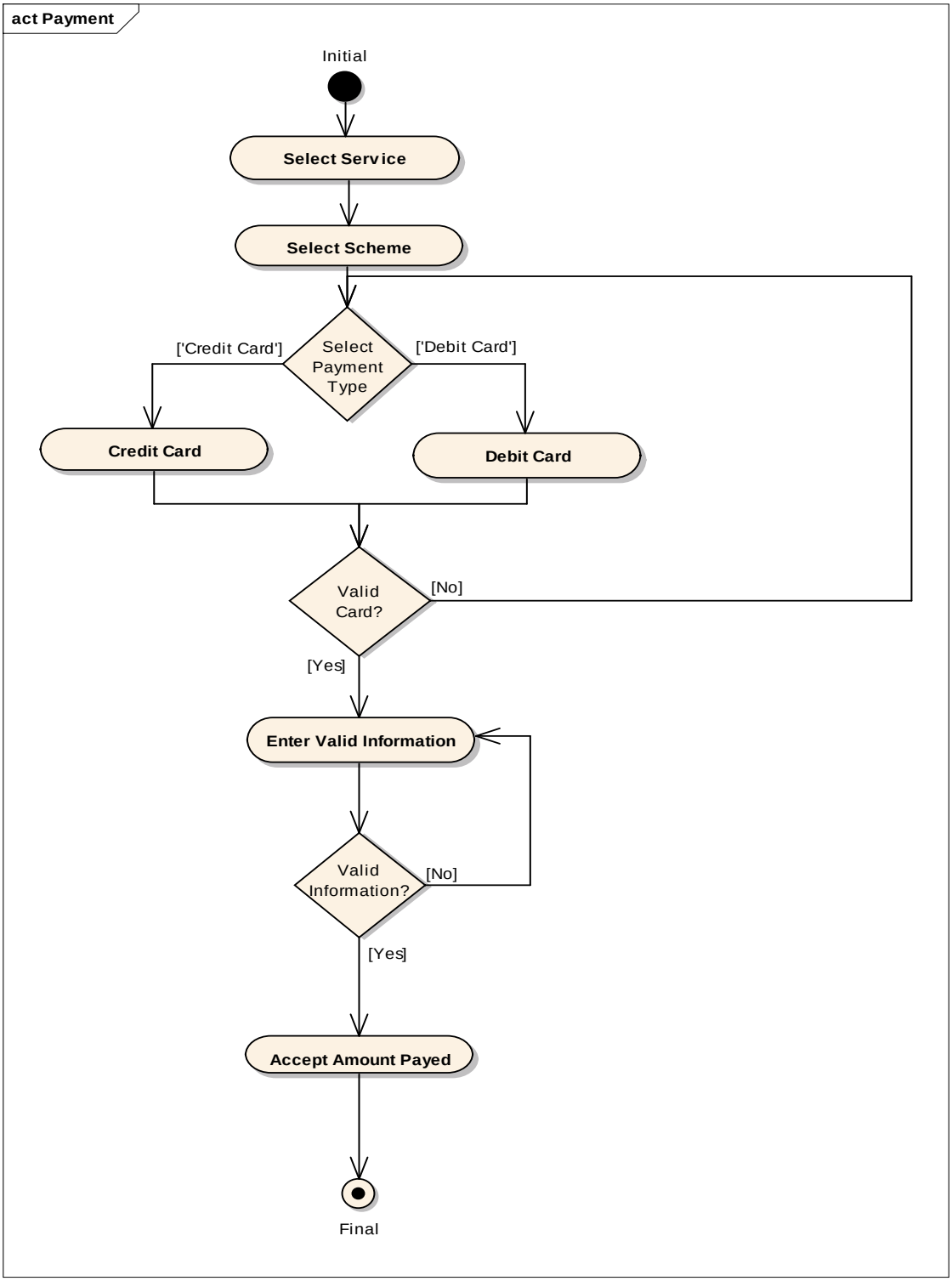


Figure 3.7.2.2 Payment Activity

Data Dictionary

1. Table Name: Designation
   Primary Key: Desig_id
   Table Key: designation

Field Name	Data Type	Size	Constraints	Description
Desig_id	Int	11	Primary Key	For designation unique identification
Desig_name	varchar	20	-	To store the designation name

Table 3.7.3.1 Designation

2. Table Name: Uploads
   Primary Key: Upload_id
   Table Key: uploads

Field Name	Data Type	Size	Constraints	Description
Upload_id	Int	11	Primary Key	For upload document unique identification
User_id	Int	5	Foreign Key	For knowing to which user it belong
UType	Varchar	10	-	To store the type of user
Title	Varchar	20	-	To store the name of document

Image	Varchar	20	-	To store the images of document
Status	Varchar	10	-	To store the status of the user

Table 3.7.3.2: Uploads

3. Table Name: User Penalty
   Primary Key: UP_id
   Table Key: user_penalty

Field Name	**Data Type**	**Size**	**Constraints**	**Description**
UP_id	int	11	Primary Key	For user penalty unique identification
User_id	int	11	Foreign Key	For knowing to which user it belong
Fine	int	10	-	To store the fine for user
Date	Date/Time	10	-	To store the date of the penalty
Status	Varchar	12	-	To store the status of the penalty
Payment_Date	Varchar	10	-	To store the date of the payment
Time	Date/Time	10	-	To store the time of the penalty

Table 3.7.3.3: User Penalty

4. Table Name: Registration

   Primary Key: User_id

   Table Key: registration

Field Name	Data Type	Size	Constraints	Description
User_id	Int	11	Primary Key	For user unique identification
District_id	int	11	Foreign Key	For knowing to which district it belong
City_id	int	11	Foreign Key	For knowing to which city it belong
Name	varchar	50	-	To store the customer name
Address	Varchar	50	-	To store the customer address
Aadhaar	varchar	16	-	To store the customer aadhaar number
Email	Varchar	50	-	To store the customer email address
Mobile	int	10	-	To store the customer mobile number
Password	Varchar	8	-	To store the customer password
Wallet	int	10	-	To store the customer

Table 3.7.3.4 Registration

5. Table Name: Penalty
   Primary Key: P_id
   Table Key: penalty

Field Name	Data Type	Size	Constraints	Description
P_id	Int	11	Primary Key	For penalty unique identification
Section	Varchar	20	-	To store the section name
Offence	Varchar	20	-	To store the offence for penalty
Fine	int	10	-	To store the fine amount

Table 3.7.3.5: Penalty

6. Table Name: District
   Primary Key: District_id
   Table Key: district

Field Name	Data Type	Size	Constraints	Description
District_id	Int	11	Primary Key	For district unique identification
District_Name	varchar	20	-	To store the district name

Table 3.7.3.6: District

7. Table Name: Payment
   Primary Key: Payment_id
   Table Key: payment

Field Name	Data Type	Size	Constraints	Description
Payment_id	Int	11	Primary Key	For payment unique identification
User_id	Int	11	Foreign Key	For knowing to which user it belong
Amount	Int	10	-	To store the amount of payment
Date	Date/Time	10	-	To store the date of payment
Time	Date/Time	10	-	To store the time of payment

Table 3.7.3.7: Payment

8. Table Name: City
   Primary Key: C_ID
   Table Key:city

Field Name	Data Type	Size	Constraints	Description
City_id	int	11	Primary Key	For city unique identification
City_Name	Varchar	20	-	To store the city name

Table 3.7.3.8: City

9. Table Name: Staff

   Primary Key: Staff_id

   Table Key: Staff

Field Name	Data Type	Size	Constraints	Description
Staff_id	Int	11	Primary Key	For staff unique identification
Designation	int	10	Foreign Key	For knowing to which designation id it belong
District_id	int	11	Foreign Key	For knowing to which district it belong
City_id	int	11	Foreign Key	For knowing to which city it belong
Name	varchar	50	-	To store the customer name
Address	Varchar	50	-	To store the staff member address
Email	Varchar	50	-	To store the staff member email address
Mobile	int	10	-	To store the staff member mobile number
Password	Varchar	8	-	To store the staff member password

Table 3.7.3.9 Staff

10. Table Name: Llr

Primary Key: Llr_id

Table Key: Llr

Field Name	Data Type	Size	Constraints	Description
Llr_id	Int	11	Primary Key	For llr unique identification
User_id	Int	11	Foreign Key	For knowing to which user it belong
Designation	int	10	Foreign Key	For knowing to which designation id it belong
District_id	int	11	Foreign Key	For knowing to which district it belong
City_id	int	11	Foreign Key	For knowing to which city it belong
Name	varchar	50	-	To store the user name
Address	Varchar	50	-	To store the user address
Pincode	Int	6	-	To store the user pincode no
Vtype	Varchar	10	-	To store the user apply type of vehicle
Email	Varchar	50	-	To store the user email address
BGroup	Varchar	3	-	To store the user blood group
Mobile	int	10	-	To store the user mobile number
Password	Varchar	8	-	To store the user

				password
Apply_date	Date/Time	12	-	To store the user apply date of licence
Status	Varchar	10	-	To store the user status of licence
Bdate	Date/Time	12	-	To store the user birthday date

Table 3.7.3.10 Staff

3.8 Control Flow Diagram

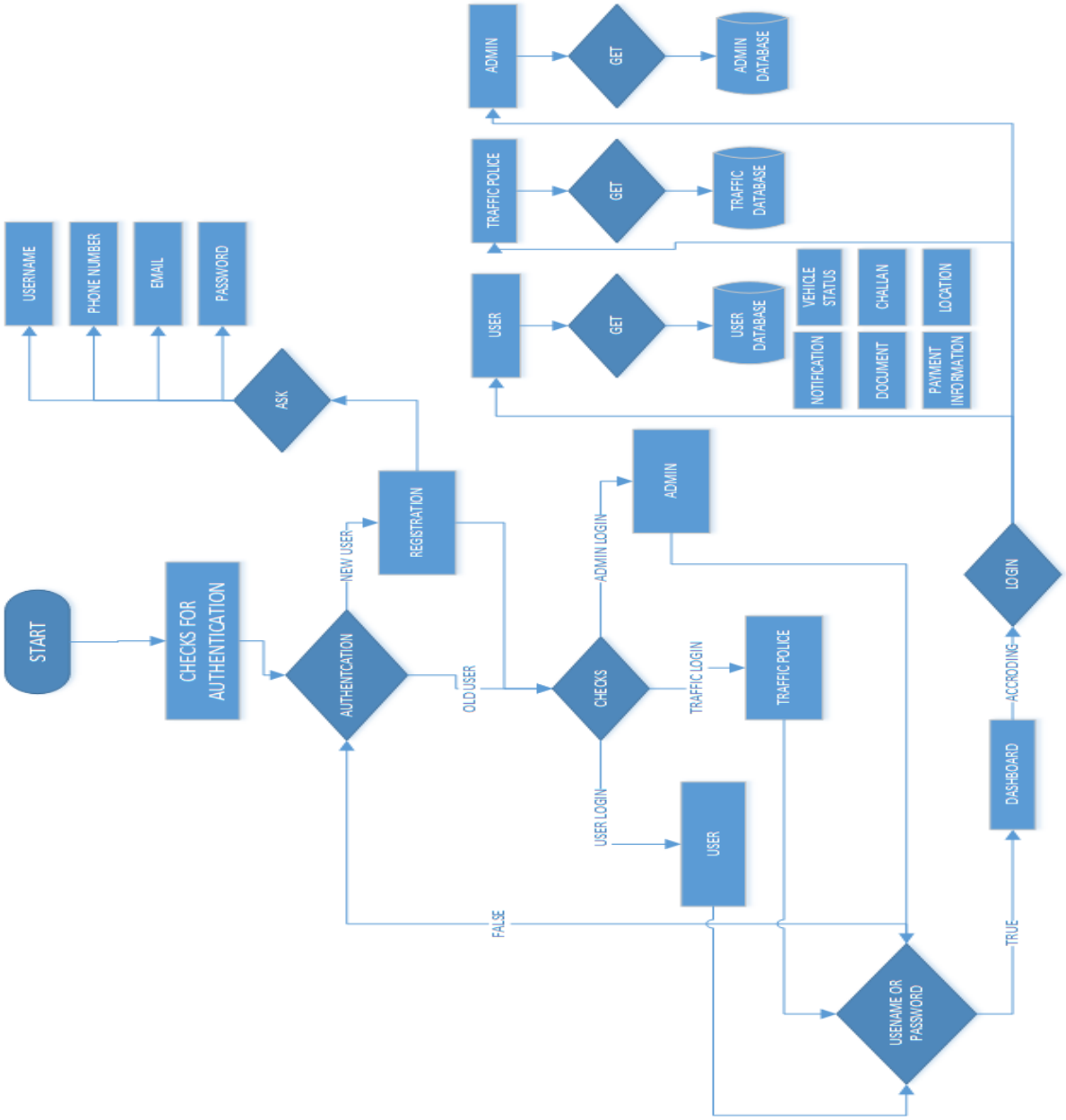


Figure 3.8.1: Control Flow Diagram

3.9 Sequence Diagram

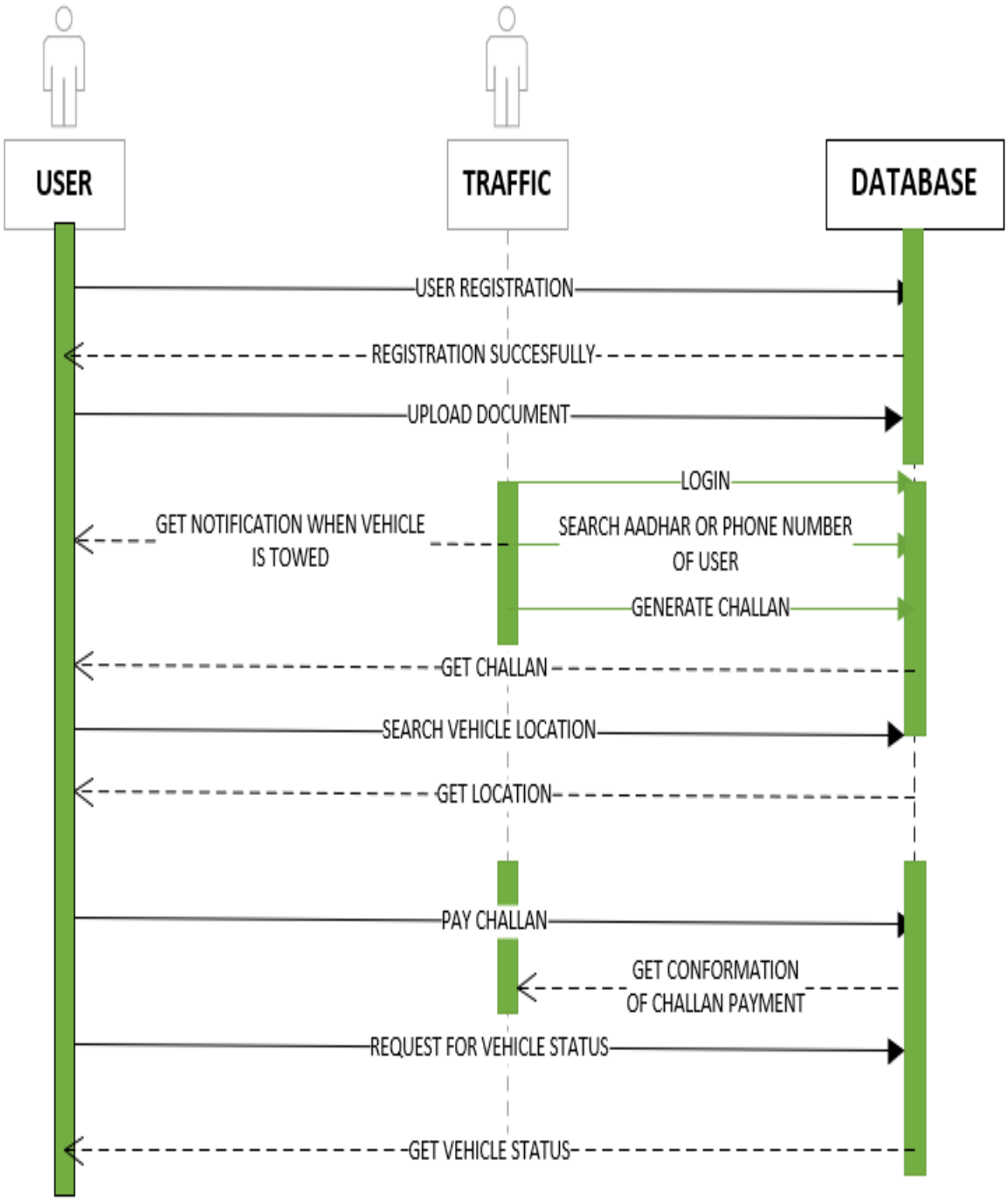
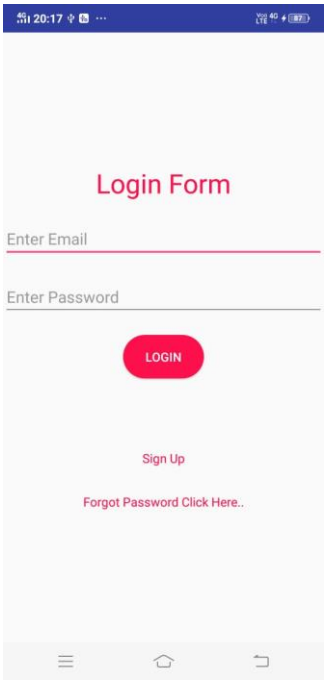
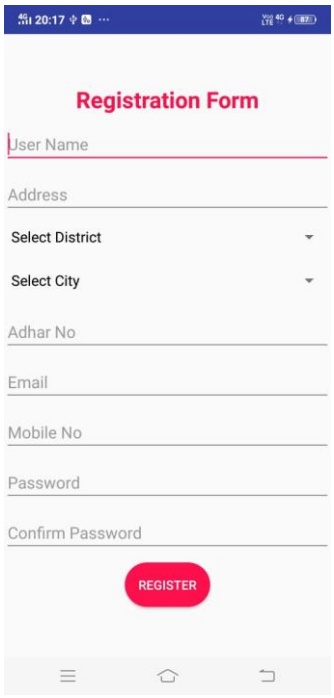


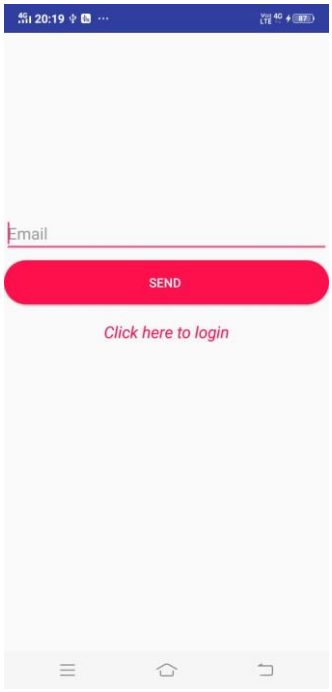
Figure 3.9.1: Sequence Diagram

4. SYSTEM DESIGN

4.1 Design of Application







4.2 System Procedural Design

4.2.1 Design Pseudo Code or Algorithm for method or operation Source code for login and registration

Login.java

```
package r.vehciletowing.com.vehicletowrto;

import android.app.ProgressDialog;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import java.util.ArrayList;
import r.vehciletowing.com.vehicletowrto.API.APICall;
import r.vehciletowing.com.vehicletowrto.POJO.AllUser;
import r.vehciletowing.com.vehicletowrto.POJO.User;

public class Login extends AppCompatActivity implements
View.OnClickListener {

    private EditText editEmail;
    private EditText editPassword;
    private Button btnLogin;
    private TextView txtSignUp;
    private TextView txtforgot;

    private String semail,spassword;

    public static final String MyPREFERENCES = "MyPrefs" ;
    public static final String Email = "emailKey";
    public static final String Login = "loginKey";
    public static final String User_ID = "uidKey";
    public static final String User_Type = "utypeKey";
    private SharedPreferences sharedpreferences;

    private ArrayList<User> ResultList;
    public String uid;
    public String srole;



    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
```

```
        editEmail = (EditText)findViewById(R.id.EtLEmail);
        editPassword = (EditText)findViewById(R.id.EtLPass);

        btnLogin = (Button) findViewById(R.id.btnLLogin);
        btnLogin.setOnClickListener(this);

        txtSignUp = (TextView)findViewById(R.id.tvLSignup);
        txtSignUp.setOnClickListener(this);

        txtforgot =(TextView)findViewById(R.id.tvLforgot);
        txtforgot.setOnClickListener(this);

        ResultList = new ArrayList<User>();

        SharedPreferences splogin = getSharedPreferences(MyPREFERENCES,
MODE_PRIVATE);

        String slogin = splogin.getString(Login, "");
        String stype = splogin.getString(User_Type, "");

        //   Toast.makeText(LoginActivity.this, slogin,
Toast.LENGTH_LONG).show();


        if(slogin.contains("Yes"))
        {
            if(stype.contains("User"))
            {
                Intent go = new Intent(Login.this, HomeActivity.class);
                startActivity(go);
                finish();
            }
            else if(stype.contains("Employee"))
            {
               /* Intent go = new Intent(Login.this,
RTOhomeActivity.class);
                startActivity(go);*/
                finish();
            }
        }

    }

    @Override
    public void onClick(View v) {

        if(v==txtSignUp)
        {
            Intent i = new Intent(Login.this,Register.class);
            startActivity(i);
            finish();
        }

        if(v==txtforgot)
        {
            Intent i = new Intent(Login.this,ForgotPass.class);
            startActivity(i);
```

```
            finish();
        }

        if(v==btnLogin)
        {
            semail = editEmail.getText().toString().trim();
            spassword = editPassword.getText().toString().trim();

            if(semail.equals(""))
            {
                editEmail.setError("Enter Email");
                return;
            }
            else if(spassword.equals(""))
            {
                editPassword.setError("Enter Password");
                return;
            }
            else
            {
                new PostLogin().execute();
            }
        }

    }



    public class PostLogin extends AsyncTask<Void, Void, String>
    {

        private ProgressDialog dialog;

        @Override
        protected void onPreExecute() {
            // TODO Auto-generated method stub
            dialog = new ProgressDialog(Login.this);
            dialog.setMessage("Please Wait..!");
            dialog.show();
            super.onPreExecute();
        }

        @Override
        protected String doInBackground(Void... params) {
            // TODO Auto-generated method stub
            String result;
            APICall api = new APICall();
            result = api.PostLogin(semail,spassword);
            return result;
        }

        @Override
        protected void onPostExecute(String result) {
            // TODO Auto-generated method stub

            if(result!="")
            {
```

```
                if(result.contains("Error"))
                {
                    Toast.makeText(Login.this,"Invalid
Email/Password",Toast.LENGTH_SHORT).show();
                }
                else
                {
                    sharedpreferences =
getSharedPreferences(MyPREFERENCES,MODE_PRIVATE);
                    SharedPreferences.Editor editor =
sharedpreferences.edit();
                    editor.putString(Email,semail.trim());
                    editor.putString(Login,"Yes");
                    editor.putString(User_Type,result.trim());
                    editor.commit();

                    new  GetUserDetail().execute();
                    if(result.contains("User"))
                    {
                        srole= "User";
                        Intent i = new
Intent(Login.this,HomeActivity.class);
                        startActivity(i);
                        finish();
                    }
                    else if(result.contains("Employee"))
                    {
                        srole= "Employee";
                       /* Intent i = new
Intent(Login.this,RTOhomeActivity.class);
                        startActivity(i);*/
                        finish();
                    }
                }
            }
            else
            {
                Toast.makeText(Login.this,"Check your Internet
Connection",Toast.LENGTH_SHORT).show();
            }
            dialog.dismiss();
            super.onPostExecute(result);
        }

    }

    public class GetUserDetail extends AsyncTask<Void,Void,AllUser>
    {
//        private ProgressDialog dialog;

        @Override
        protected void onPreExecute() {
            // TODO Auto-generated method stub

   /*            dialog = new ProgressDialog(LoginActivity.this);
               dialog.setMessage("Loading...");
               dialog.show();
```

```
   */            super.onPreExecute();
        }


        @Override
        protected AllUser doInBackground(Void... params) {
            // TODO Auto-generated method stub

            APICall api = new APICall();
            AllUser result = api.getUserDetail(semail);
            return result;
        }

        @Override
        protected void onPostExecute(AllUser result) {
            // TODO Auto-generated method stub
            if(result !=null)
            {
                ResultList.addAll(result.getData());

                if(srole.contains("User"))
                {
                    uid =
ResultList.get(0).getUser_ID().toString().trim();
                }
                else
                {
                    uid =
ResultList.get(0).getEmployee_ID().toString().trim();
                }
                if(uid!="")
                {
                    //     Toast.makeText(LoginActivity.this,"Uid=" +uid,
Toast.LENGTH_LONG).show();

                    sharedpreferences =
getSharedPreferences(MyPREFERENCES,MODE_PRIVATE);
                    SharedPreferences.Editor editor =
sharedpreferences.edit();
                    editor.putString(User_ID,uid.trim());
                    editor.commit();

                }
                else
                {
                    return;
                }
            }
            else
            {

            }
            //            dialog.dismiss();
            super.onPostExecute(result);
        }
```

```
    }

}
```

Registration.java

```
package r.vehciletowing.com.vehicletowrto;

import android.app.ProgressDialog;
import android.content.Intent;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

import java.util.ArrayList;

import r.vehciletowing.com.vehicletowrto.API.APICall;
import r.vehciletowing.com.vehicletowrto.POJO.AllUser;
import r.vehciletowing.com.vehicletowrto.POJO.User;

public class Register extends AppCompatActivity implements
View.OnClickListener {

    private ArrayList<User> AllDistrict;
    private ArrayList<String> districts;
    private Spinner spindistrict;
    private ArrayAdapter<String> DistrictAdapter;
    protected int selecteddistrict;

    private ArrayList<User> AllCity;
    private Spinner spincity;
    private ArrayList<String> cities;
    private ArrayAdapter<String> CityAdapter;
    protected String did;
    private String cid1;
    private int selectedcid;
    private EditText editname;
    private EditText editaddress;
    private EditText editaadhar;
    private EditText editemail;
    private EditText editmobile;
    private EditText editpassword;
    private EditText editcpassword;
    private Button btnsignup;
    private String sname;
    private String saddress;
    private String semail;
    private String smobile;
    private String spassword;
```

```
    private String scpassword;
    private String saadhar;



    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);




        spindistrict = (Spinner)findViewById(R.id.spinDistrict);
        spincity = (Spinner)findViewById(R.id.spinCity);
        AllDistrict = new ArrayList<User>();
        AllCity = new ArrayList<User>();

        districts = new ArrayList<String>();
        cities = new ArrayList<String>();

        districts.add("Select District");
        cities.add("Select City");

        DistrictAdapter = new
ArrayAdapter<String>(Register.this,android.R.layout.simple_spinner_dropdow
n_item,districts);
        CityAdapter = new
ArrayAdapter<String>(Register.this,android.R.layout.simple_spinner_dropdow
n_item,cities);

        spindistrict.setAdapter(DistrictAdapter);
        spincity.setAdapter(CityAdapter);

        new getDistrict().execute();
        spindistrict.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {

            @Override
            public void onNothingSelected(AdapterView<?> parent) {
                // TODO Auto-generated method stub

            }

            @Override
            public void onItemSelected(AdapterView<?> parent, View view,
int position, long id) {
                // TODO Auto-generated method stub
                selecteddistrict = spindistrict.getSelectedItemPosition();

                if(selecteddistrict==0)
                {
                    cities.clear();
                    cities.add("Select City");
                }
                else
```

```
                {
                    did = AllDistrict.get(selecteddistrict-
1).getDistrict_ID().toString().trim();
                    new getCity().execute();
                }
                CityAdapter.notifyDataSetChanged();
            }
        });
        editname = (EditText) findViewById(R.id.EtName);
        editaddress = (EditText) findViewById(R.id.EtAddress);
        editaadhar = (EditText) findViewById(R.id.EtAdharNo);
        editemail = (EditText) findViewById(R.id.EtEmail);
        editmobile = (EditText) findViewById(R.id.EtMobileNo);
        editpassword = (EditText) findViewById(R.id.EtPassword);
        editcpassword = (EditText) findViewById(R.id.EtConPassword);
        btnsignup = (Button)findViewById(R.id.btnRegister);
        btnsignup.setOnClickListener(this);
    }

    @Override
    public void onClick(View v) {

        if(v==btnsignup)
        {
            sname= editname.getText().toString().trim();
            saddress= editaddress.getText().toString().trim();
            saadhar= editaadhar.getText().toString().trim();
            semail= editemail.getText().toString().trim();
            smobile= editmobile.getText().toString().trim();
            spassword= editpassword.getText().toString().trim();
            scpassword= editcpassword.getText().toString().trim();
            selectedcid = spincity.getSelectedItemPosition();




            if(selectedcid!=0)
            {
                cid1 = AllCity.get(selectedcid-
1).getCity_ID().toString().trim();
            }
            if(sname.equals(""))
            {
                editname.setError("Enter Name");
                return;
            }
            else if(selecteddistrict==0)
            {
                Toast.makeText(Register.this,"Select
District",Toast.LENGTH_SHORT).show();
                return;
            }
           /* else if(selectedcid==0)
            {
                Toast.makeText(Register.this,"Select
```

```
City",Toast.LENGTH_SHORT).show();
                return;
            }*/

            else if(saddress.equals(""))
            {
                editaddress.setError("Enter Address");
                return;
            }
            else if(saadhar.equals(""))
            {
                editaadhar.setError("Enter Aadhar no");
                return;
            }
            else if(semail.equals(""))
            {
                editemail.setError("Enter Email");
                return;
            }
            else if(smobile.equals(""))
            {
                editmobile.setError("Enter Mobile");
                return;
            }
            else if(spassword.equals(""))
            {
                editpassword.setError("Enter Password");
                return;
            }
            else if(scpassword.equals(""))
            {
                editcpassword.setError("Enter Password Again");
                return;
            }
            else
            {
                if(spassword.matches(scpassword))
                {

//Toast.makeText(Register.this,"APICall",Toast.LENGTH_SHORT).show();
                    new Postregister().execute();
                   // Toast.makeText(this, "i m here 2",
Toast.LENGTH_SHORT).show();
                }
                else
                {
                    editcpassword.setText("");
                    Toast.makeText(Register.this,"Enter same password",
Toast.LENGTH_SHORT).show();
                    return;
                }

            }
        }

    }
```

```
  private class Postregister extends  AsyncTask<Void,Void,String>
  {
      private ProgressDialog dialog;

      @Override
      protected void onPreExecute() {
          dialog = new ProgressDialog(Register.this);
          dialog.setMessage("Please Wait..!");
          dialog.show();
          super.onPreExecute();
      }

      @Override
      protected String doInBackground(Void... voids) {
          String result;
          APICall api = new APICall();

result=api.saveuser(sname,did,cid1,saddress,semail,smobile,scpassword,saad
har);
          return result;
      }

      @Override
      protected void onPostExecute(String result) {
          if(result!="")
          {
              dialog.dismiss();

Toast.makeText(Register.this,result,Toast.LENGTH_SHORT).show();
              if(result.contains("Registration Successful..!"))
              {
                  Intent i = new Intent(Register.this,Login.class);
                  startActivity(i);
                  finish();
              }
              else
              {
                  return;
              }
          }
          else
          {
              Toast.makeText(Register.this,"Check your Internet
Connection",Toast.LENGTH_SHORT).show();
          }
          super.onPostExecute(result);

      }
  }

    public class getDistrict extends AsyncTask<Void, Void,AllUser>
    {

        private ProgressDialog dialog;

        @Override
```

```
        protected void onPreExecute() {
            // TODO Auto-generated method stub

            dialog = new ProgressDialog(Register.this);
            dialog.setMessage("Please wait");
            dialog.show();
            super.onPreExecute();
        }

        @Override
        protected AllUser doInBackground(Void... params) {
            // TODO Auto-generated method stub

            APICall api = new APICall();
            AllUser result = api.getDistrictList();
            return result;
        }

        @Override
        protected void onPostExecute(AllUser result) {
            // TODO Auto-generated method stub

            if(result != null)
            {
                AllDistrict.addAll(result.getData());
                for(User s : result.getData())
                {
                    districts.add(s.getDistrict_Name());
                }

            }
            else
            {

            }
            DistrictAdapter.notifyDataSetChanged();
            dialog.dismiss();
            super.onPostExecute(result);
        }
    }

    public class getCity extends AsyncTask<Void, Void,AllUser>
    {

        private ProgressDialog dialog;

        @Override
        protected void onPreExecute() {
            // TODO Auto-generated method stub

            dialog = new ProgressDialog(Register.this);
            dialog.setMessage("Please wait");
            dialog.show();
            super.onPreExecute();
        }

        @Override
```

```
        protected AllUser doInBackground(Void... params) {
            // TODO Auto-generated method stub

            APICall api = new APICall();
            AllUser result = api.getCityList(did);
            return result;
        }

        @Override
        protected void onPostExecute(AllUser result) {
            // TODO Auto-generated method stub

            if(result != null)
            {
                AllCity.addAll(result.getData());
                for(User s : result.getData())
                {
                    cities.add(s.getCity_Name());
                }

            }
            else
            {

            }
            CityAdapter.notifyDataSetChanged();
            dialog.dismiss();
            super.onPostExecute(result);
        }
    }

}
```

Splash.java

```
package r.vehciletowing.com.vehicletowrto;

import android.content.Intent;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class Splash extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_splash);
        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                startActivity(new Intent(Splash.this,Login.class));
                finish();
            }
        },3000);
    }
}
```

Forgotpass.java

```
package r.vehciletowing.com.vehicletowrto;

import android.app.ProgressDialog;
import android.app.SearchManager;
import android.content.Intent;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import r.vehciletowing.com.vehicletowrto.API.APICall;

public class ForgotPass extends AppCompatActivity implements
View.OnClickListener {

    private EditText editEmail;
    private Button btnsend;
    private TextView txtlogin;
    private String semail;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_forgot_pass);

        editEmail = (EditText)findViewById(R.id.editText1);

        btnsend = (Button) findViewById(R.id.button1);
        btnsend.setOnClickListener(this);

        txtlogin = (TextView)findViewById(R.id.textView1);
        txtlogin.setOnClickListener(this);
    }



    @Override
    public void onClick(View v) {
        if(v==btnsend)
        {
            semail = editEmail.getText().toString().trim();

            if(semail.equals(""))
            {
                editEmail.setError("Enter Email");
                return;
            }
            else
            {
```

```
                new PostForgetPass().execute();
            }
        }

        if(v==txtlogin)
        {
            Intent i = new Intent(ForgotPass.this,Login.class);
            startActivity(i);
            finish();
        }

    }

    public class PostForgetPass extends AsyncTask<Void, Void, String>
    {
        private ProgressDialog dialog;

        @Override
        protected void onPreExecute()
        {
            //
            dialog = new ProgressDialog(ForgotPass.this);
            dialog.setMessage("Please Wait..!");
            dialog.show();
            super.onPreExecute();
        }

        @Override
        protected String doInBackground(Void... params)
        {
            // TODO Auto-generated method stub
            String result;

            APICall api =new APICall();
            result = api.PostForgotPassword(semail);
            return result;
        }

        @Override
        protected void onPostExecute(String result) {
            // TODO Auto-generated method stub

            if(result!="")
            {
                //
Toast.makeText(ForgotPasswordActivity.this,result,Toast.LENGTH_SHORT).show
();

                if(result.contains("Sent"))
                //if(result.contains("$pass"))
                {
                    Toast.makeText(ForgotPass.this,"Email sent
Successfully..!",Toast.LENGTH_SHORT).show();
                    Intent i = new Intent(ForgotPass.this,Login.class);
                    startActivity(i);
                    finish();
                }
```

```
                else
                {
                    Toast.makeText(ForgotPass.this,"Please try
again..!",Toast.LENGTH_SHORT).show();
                }
            }
            else
            {
                Toast.makeText(ForgotPass.this,"Check your Internet
Connection",Toast.LENGTH_SHORT).show();
            }
            dialog.dismiss();
            super.onPostExecute(result);
        }

    }
}
```

5. Canvas

5.1 EMPATHY MAPPING

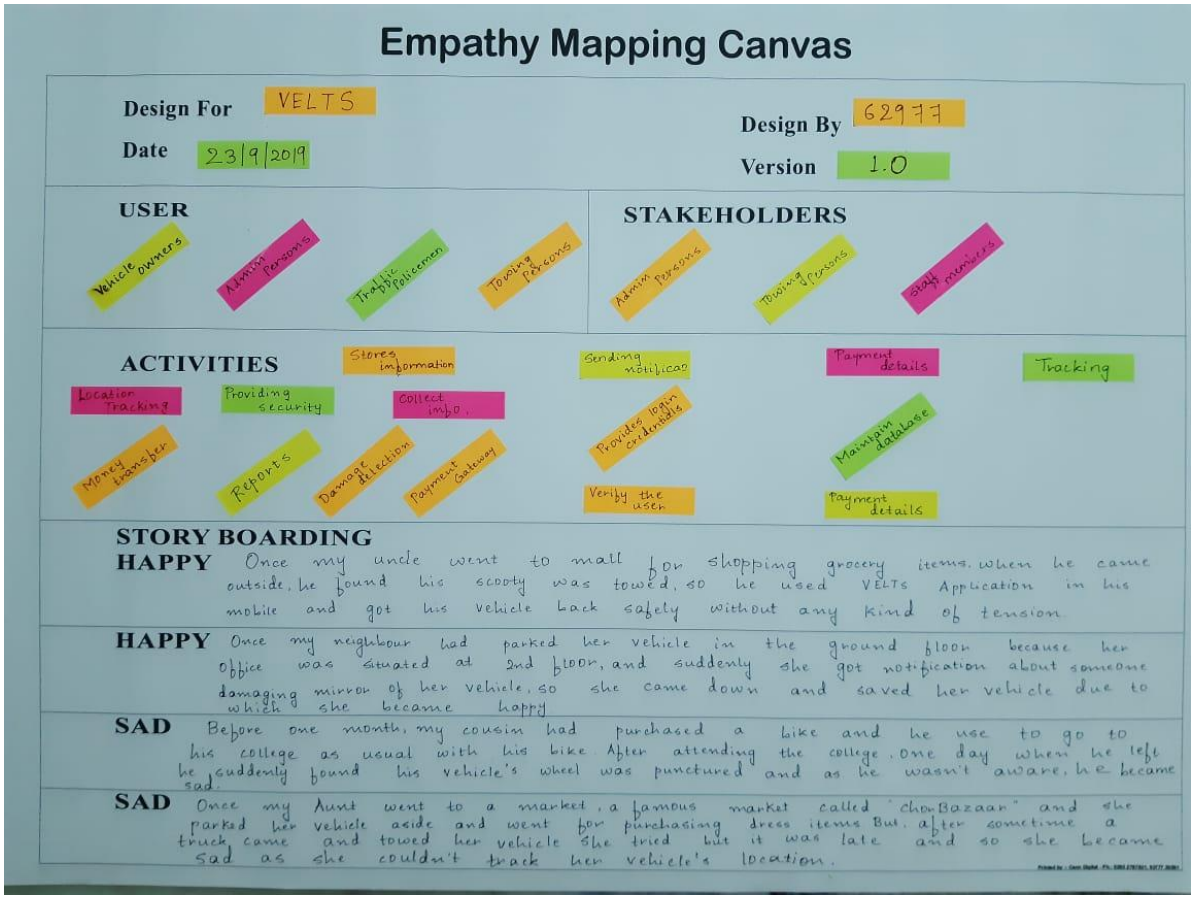


Fig 5.1.1 Empathy Mapping Canvas

5.2 AEIOU CANVAS

Fig 5.2.1 AEIOU Summary

5.3 IDEATION CANVAS

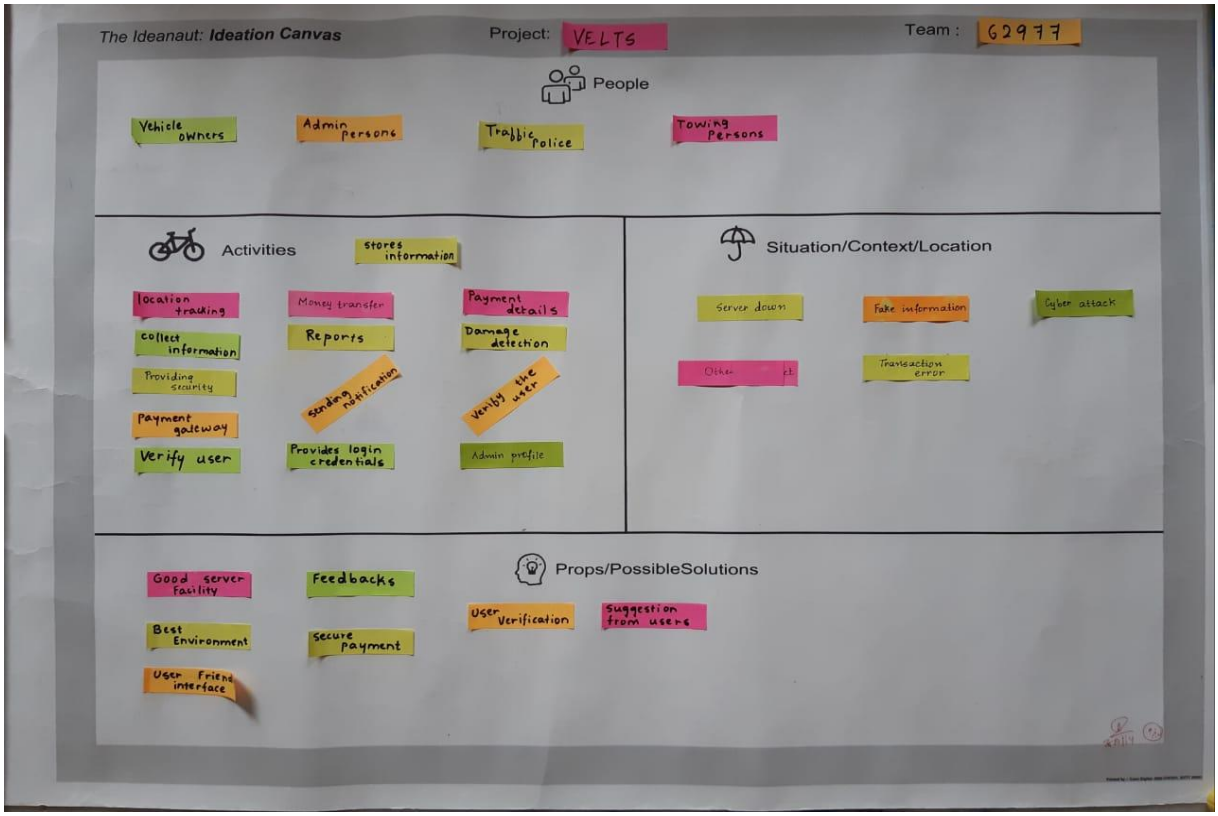


Fig 5.3.1 Ideation canvas

5.4 PRODUCT DEVELOPMENT CANVAS

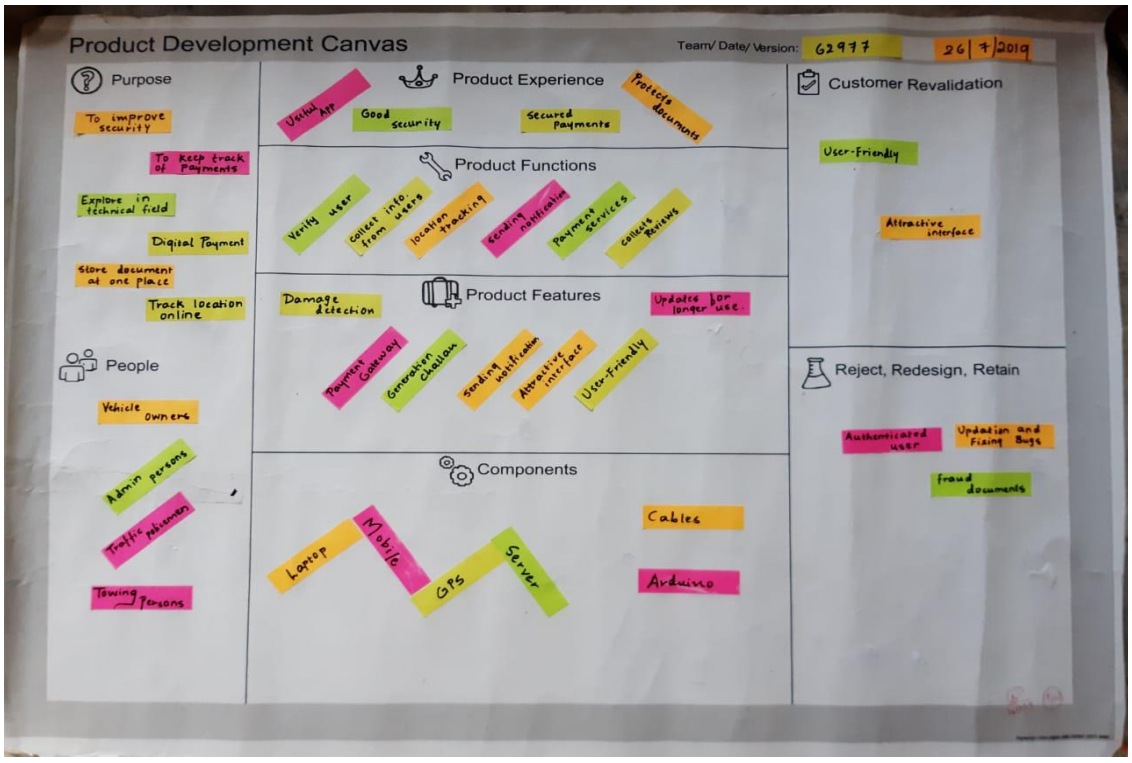


Fig 5.4.1 Product Development Canvas

6. Conclusion

This is to conclude from the Vehicle Enhancement Location Tracking System (VELTS) that we may develop a vehicle tracking, monitoring and alerting system using combination of reliable Sensor, GPS, GSM/GPRS with high speed processor. The system will have latest technology and optimized algorithm with moderate cost. It will bring transparency and a step towards digitalization as it will also store various kinds of information of user as well as traffic policemen. The system may focus on real time position of vehicle and it can be installed in cargo, trucks, buses, cars, two-wheelers and boats.

References

- https://www.etrailer.com/faq-towing-a-vehicle-overview.aspx
- https://en.wikipedia.org/wiki/Vehicle_tracking_system
- https://avlview.com/vehicle-tracking/live-gps-vehicle-tracking/
- https://trackimo.com/live-map-tracking-system/

Dupli Checker

PLAGIARISM SCAN REPORT

20%
Plagiarised

80%
Unique

Date	2019-10-13
Words	681
Characters	4266
Exculde Url	None

Content Checked For Plagiarism

This is a Mobile-app in which we are attempting to achieve a totally new genre wherein we will bring Certainty at one stop for all kinds of Vehicles worldwide where Owners, Traffic Policemen, Admin members can log into it and add essential Documents, Generate Challan, Verify Documents and Payment Status, etc. Create User's profile, Verify ID Proof, save all crucial documents such as License, RC Book and also Track Live Location of the registered Vehicle. Today many vehicle systems that are in use now days are some form of Automatic Vehicle Location. It is based on determining the location of a vehicle and transmitting this information to a remotely located server. The location is determined using GPS usin a gsm module. GSM is the most common used service for this purpose. There is no such application in existing system that can be connected with our registered vehicle through GSM Module providing facilities like Storing documents of all Vehicles, Digitalized Payments or Showing Live Location of Towed/Damaged Vehicles. This application keeps track of records of all users about their payments for generated challan and it can be used by Admin, Traffic Policemen as well as registered users. There will be an facility for the payment of challan and monitor the vehicle location if the vehicle is moved from one place to another place or if the vehicle is being damaged by anyone in your absence .Also after creating the account, the user will be getting the password for log into the account to upload the documents like License, Pay Penalty if Challan is Generated, Give reviews and can take advantage of Vehicle Monitoring Services. VIER(CE) Page 2 1.2 PURPOSE The purpose of developing the application is to bring Vehicle Monitoring Service at a single platform so that everyone can use it easily without worrying about Vehicle's documents. Another major purpose of proposed system is to monitor the vehicle, detect the damage if the vehicle is fallen and making the challan digital so that the user can easily pay the payment without any argument. 1.3 SCOPE Scope describes the functions and features that are to be delivered to end-users, also data-input and output, contents, performance, constraints, interface and reliability. This System will be able to Manage Vehicle Enhancement Services remotely and also maintain co-ordination among Customers and Admin. 1.4 OBJECTIVE The basic objective of this project is to develop a system which can cater all the needs of Recent Era regarding Vehicle Enhancement at one platform. Main Objective of Vehicle Enhancement System is to develop a system which can bring Security/Storing of all kinds of useful documents related to all types of vehicles of whole Society to make it a useful application for Customers and to provide all the Vehicle monitoring Services to the Users. VIER(CE) Page 3 Vehicle tracking improves safety and security, communication medium, performance monitoring and it will increase productivity. So in future it will play a major role in our dayto-day living. 1.5 TECHNOLOGY AND LITERATURE REVIEW The Key Technologies used in developing the Vehicle Enhancement and Location Tracking System are: 1. Servers: Xamp Server (Localhost) 2. Tools: Arduino Android Studio IDE Microsoft Visio 2007 We used MySQL server as the database. The usage of MySQL Server was done by analyzing the advantage and disadvantage of various databases existing in the market. MySQL is a relational database management system (RDBMS), and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or use MySQL "front-ends", desktop software and web applications that create and manage MySQL databases, build database structures, back up data, inspect status, and work with data records. The official set of MySQL front-end tools, MySQL Workbench is actively developed by Oracle, and is freely available for use. phpMyAdmin is a free and open source tool written in PHP intended to handle the administration of MySQL with the use of a web browser. It can perform various tasks such as creating, modifying or deleting databases, tables, fields or rows; executing SQL statements; or managing users

and permissions.

Matched Source

Similarity 4%

Title: (PDF) Smart Vehicle Tracking System

tracking improves safety and security, communication medium, performance. international journal of distributed and parallel systems (ijdps)the emerging synchronous duty-cycled wakeup schedules together with the unreliable communication of wsns post new challenges for efficient...

https://www.researchgate.net/publication/274174371_Smart_Vehicle_Tracking_System

---

Similarity 10%

Title: 2.MYSQL Setting Path in the Environment Variable - YouTube

mysql is a relational database management system (rdbms), and ships with no gui tools to administer mysql databases or manage dataxampp also provides support for creating and manipulating databases in mysql and sqlite among others.once xampp is installed, it is possible...

https://www.youtube.com/watch?v=JNMWpEPp-aU

Similarity 3%

Title: Car pooling final synopsis | Databases | Graphical User Interfaces

sign up: users need to first register to this application by filling up the required information like first name, last name, email id, contactno, gender, age, date of birth, username and password.mysql databases, build database structures, back up data, inspect status, and work with data records.

https://www.scribd.com/document/351619921/Car-pooling-final-synopsis

---

Similarity 3%

Title: PhpMyAdmin Flashcards | Quizlet

phpmyadmin is a free and open source tool written in php intended to handle the administration of mysql or mariadb with the use of a web browser. it can perform various tasks such as creating, modifying or deleting databases, tables, fields or row.

https://quizlet.com/124313910/phpmyadmin-flash-cards/

Similarity 3%

Title: MySQL - Wikipedia

least 2015 to use the dual-licensing strategy long used by mysql ab, with proprietary and gpl versions available.phpmyadmin is a free and open source tool written in php intended to handle the administration of mysql with the use of a web browser.

https://en.wikipedia.org/wiki/MySQL
