A PROJECT REPORT

ON

APPLICATION FOR REAL TIME CRIME REPORTING

Mini Project report submitted in the partial fulfillment
of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

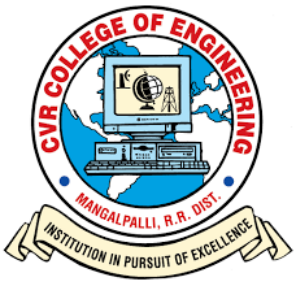
Submitted By

P. SANDEEP	**17B81A05M5**
CH. SHIVA	**17B81A05P1**
K. SAITEJA REDDY	**17B81A05L3**

Under the guidance of

Ms. BOGA HARIPRIYA

Assistant Professor



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CVR COLLEGE OF ENGINEERING

(***An Autonomous institution, NBA, NAAC Accredited and Affiliated to JNTUH, Hyderabad***)

Vastunagar, Mangalpalli (V), Ibrahimpatnam (M),
Rangareddy (D), Telangana- 501 510

June 2020

Cherabuddi Education Society's
CVR COLLEGE OF ENGINEERING
(An Autonomous Institution)
ACCREDITED BY NBA, NAAC 'A' Grade
(Approved by AICTE & Government of Telangana and Affiliated to JNTU Hyderabad)
Vastunagar, Mangalpalli (V), Ibrahimpatnam (M), R.R.District.
Web: http://www.cvr.ac.in, email: info@cvr.ac.in
Ph : 08414 – 252222, 252369, Office Telefax : 252396, Principal : 252396 (O)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project entitled "**Application For Real Time Crime Reporting**" being submitted by **P. SANDEEP (17B81A05M5), CH. SHIVA (17B81A05P1) and K. SAITEJA REDDY(17B81A05L3)** in partial fulfillment of the requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering to Jawaharlal Nehru Technological University (JNTUH), Hyderabad, is a bonafide work carried out by them under my guidance and supervision. The results have provided in this report have not been submitted to any other university or institution for the award of any degree.

Ms. Boga Haripriya
Assistant Professor
Project Supervisor

Dr. A. Vani Vathsala
Professor
Head of the Department

External Examiner

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragement and guidance has been a source of inspiration throughout the course of the project.

It is a great pleasure to convey our profound sense of gratitude to our **Principal Dr. Nayanathara K. S** and **Dr. A. Vani Vathsala, Head of the CSE Department**, CVR College of Engineering for being kind enough for arranging the necessary facilities for executing the project in the college.

We would like to express our sincere gratitude to our guide, **Ms. Boga Haripriya, Assistant Professor, CSE Dept.**, CVR College of Engineering**,** whose guidance and valuable suggestions have been indispensable to bring about the successful completion of our project.

We would also like to express our gratitude to all the staff members and lab faculty, department of **Computer Science and Engineering**, CVR College of Engineering for the constant help and support.

We wish a deep sense of gratitude and heartfelt thanks to management for providing excellent lab facilities and tools. Finally, we thank all those whose guidance helped us in this regard.

P.Sandeep **17B81A05M5**

Ch.Shiva **17B81A05P1**

K.Saiteja Reddy **17B81A05L3**

ABSTRACT

Each of the modules used within the app shall be able to satisfactorily do their job. The model has the sided architecture. On one side is the Client side that facilitates the use of application for capturing or recording the crime and generate the report of the crime taken along with the location of the crime. We have developed an Android Application that gives add-on advantages over Reporting the live crime along with the e-logging of complaints from within the application. In the wide spreading market of existing apps available on the Google Play Store there are apps that may facilitate the e-filling of the reports but there are no apps that volunteer the aid to quickly capture, record and report along with providing the assistance in case of locating the nearest hospitals to broad the victim to. Having a centralized control over reporting the crime live shall serve as a boon to the society making the system fully transparent and an organizational-building better managerial accounts to help the police and judiciary councils in better control and monitoring.This shall also facilitate the police to keep an eagle's eye watch over the crime. Shall save the life of the patient by reducing the time need to file in the report manually. One may not require police verification report before admitting the patient to the hospital in case of Emergency since the e-report shall do what is needed to file in as police consent.

Table of Contents

LIST OF FIGURES

1.INTRODUCTION

1.1. MOTIVATION

Crime is one of the major challenges that governments around the world are facing. Many of the crimes committed were unreported to the authorities. Based on this situation we developed an app as APPLICATION FOR REAL TIME CRIME REPORTING by using this any citizen can report a crime and it reaches to the respective officials without visiting to the police stations.Making crime reporting through mobile makes the public much easier to report so that implies in increasing the reporting crime percentage in records which might helps in reducing crime

1.2. PROBLEM STATEMENT

The main theme is to develop a user-friendly application for the public to inform police about the crimes and emergencies in real time. Reporting crime through the internet or mobile is very low in statistics. Using an Application for reporting easy for police as well as citizens.

1.3. PROJECT REPORT ORGANIZATIONS

This book contains six chapters. The first chapter contains motivation and problem statement of the project. The second chapter includes the characteristics of the problem, design challenges and proposed solution. The third chapter includes requirements and specifications.

The fourth chapter includes which contains UML diagrams and Technology Description . The fifth chapter includes Implementation which contains the technologies used for developing the application and code snippets. The fifth chapter also contains testcases and screenshots of the applications. The sixth chapter looks into the future enhancements and conclusion of the project.

2.PROPOSED MODEL

2.1. INTRODUCTION TO THE CHARACTERISTICS OF THE PROBLEM

Crime is one of the major challenges that most governments around the world are strangling with. Complaining about a crime that's happening in life is very low in statistics that very few people going to complain to the emergency number. Nowadays manual complaint that people has witnessed is not happening for sure. The main thing is that people are reluctant to complain through emergency numbers and register the location. So this sort of problem may be reduced through making this system automatic by mobile handheld devices as possible tools for effective crime detection and reporting

Through the existing system most of the reporting devices are through the website or in the mobile reported devices are reporting to few static sources like email etc. and very applications take location and report it to backend. By this project we maintain a web application for police in taking complains and for citizen making user friendly application for authorized people and forwarding their current location to report the live crime that is happening in live. This is increases in reporting crime but the limitation is that handling of few fake complaint will be a limitation as it is not an anonymous for citizen that problem might be very less.

2.2. DESIGN CHALLENGES

Cross -platform application:

- Choosing a cross platform works for both Android & iOS platforms with common codebase.
- Chosen platform don't have academic knowledge

Database:

- Selecting a database such that it can synchronize both mobile and web application.
- Uploading a multimedia data to database for visual proof for chosen cross platform

2.3.PROPOSED SOLUTION

In the proposed model the citizen reports a crime with proof of image, location, and description of the scene this aspects will be sent to the police

As per the project architecture we use required API for the location, speech to text transformation for description, and uploading data to firebase as both the sides will access and upload data we use flask as a server which is maintained by admin/police. In return, police will be notified through API calls to the client which in this case is a citizen.

In the proposed model the platform that we are using is flutter v1.12.13 . Flutter is Google's UI toolkit for building beautiful, natively compiled applications for mobile, web, and desktop from a single codebase. At the backend we are using the framework named Flask 1.1.2. Flask is an API of Python that allows us to build up web-applications. Flask is a relatively young framework. Flask is considered more "Pythonic" than Django is simply because Flask web application code is, in most cases, more explicit and it is extremely flexible Flask is based on WSGI(Web Server Gateway Interface) toolkit and Jinja2 template engine . The database here in the project we are using to synchronize both flutter and flask if firebase. Firebase provides a real-time database and back-end 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.

3.REQUIREMENTS AND SPECIFICATIONS

3.1 SOFTWARE REQUIREMENTS

3.1.1 Functional Requirements

These requirements define the capabilities and functions that the implemented system must have in order to achieve its intended purpose. It includes a set of inputs, behaviour and outputs in line with the objectives of the study. They include:

1. Login and Logout:To gain access to the web application users must first signup/create account with their respective details and later login using username and password. Users need to logout in order to exit the system.
2. Contact information: A user can see the contact details of the respective police stations and emergency numbers.
3. Reported crimes: A user can check the crimes reported by him.
4. Image Capturing: A user can take and upload an image into the app.
5. Voice Message: A user can Write a description with his voice speech.

3.1.2 Non-Functional Requirements

These requirements that specify the criteria used to judge the operation of the system. They were constructed in agreement with functional requirements that define specific behaviour and functions. They include:

1. Usability: the system interface should be easy to use.
2. Reliability and availability: the system should be reliable and always available to perform tasks requested by the user.
3. Scalability: the system should be able to adopt additional functionalities. Additional data should be easy to incorporate.
4. Integrity: the system being data oriented, it should ensure that the data analysed and stored is not altered or corrupted.
5. Performance: the system should have an acceptable response time while performing its functions.

6. Security: The system should allow only authorized users to use its functionalities.

3.2 SYSTEM SPECIFICATIONS

3.2.1 Software specifications

The software used for the development of this project are:

1. Flutter SDK 1.12.13: For developing Android/ iOS app from a single code base.
2. Firebase: Firebase provides a real-time database and back-end 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.
3. Windows Operating System(8.1 or above version)
4. Android Operating System (marshmallow version or above version)

Programming Languages used

1. Python3.0
2. Dart 2.60
3. HTML & CSS

IDE'S used

1. Visual Studio Code
2. Sublime Text
3. Notepad

3.2.2 Hardware Specifications

The hardware used for the development of this project are:

1. Any vendor software phone (Motorola, Samsung etc)
2. A PC with
    a. Processor : intel i3 or higher
    b. RAM : 2 GB
    c. Hard disk :1TB

4. ANALYSIS AND DESIGN

4.1.USECASE DIAGRAM

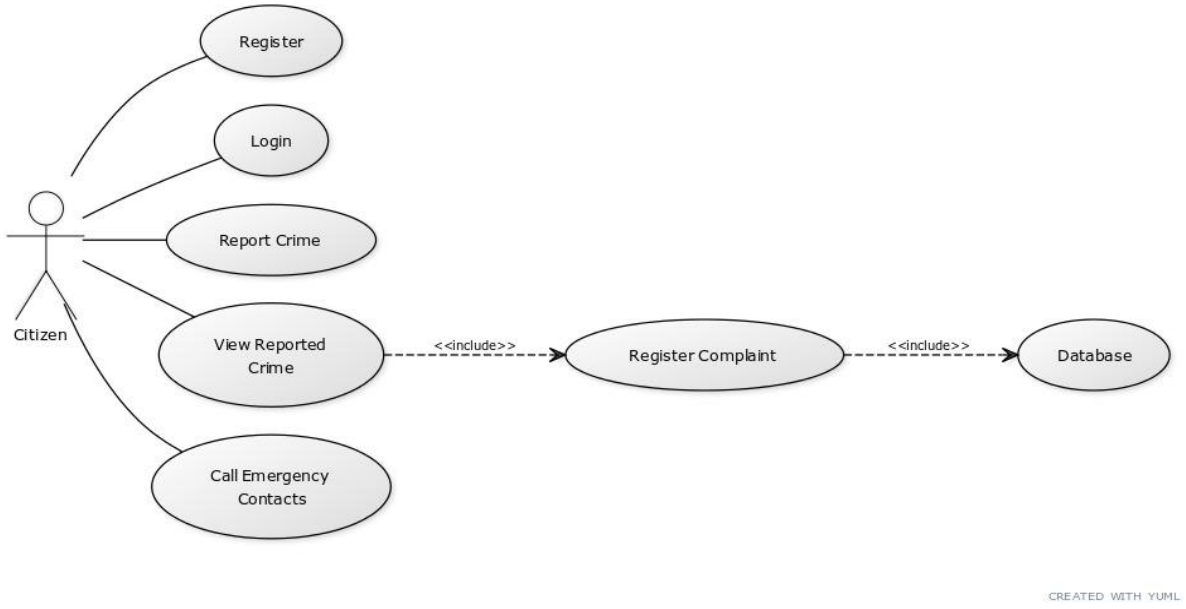


Figure 4.1(a) : Citizen Usecase Diagram

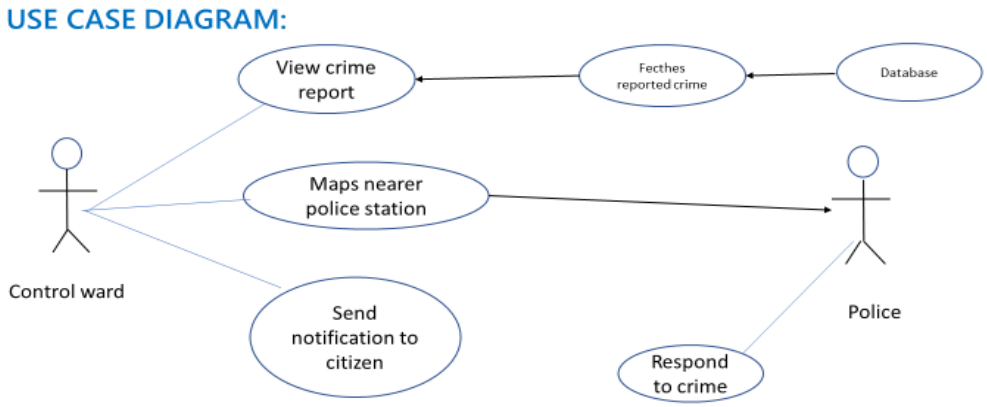


Figure 4.1(b) : Police Usecase Diagram

4.2.CLASS DIAGRAM

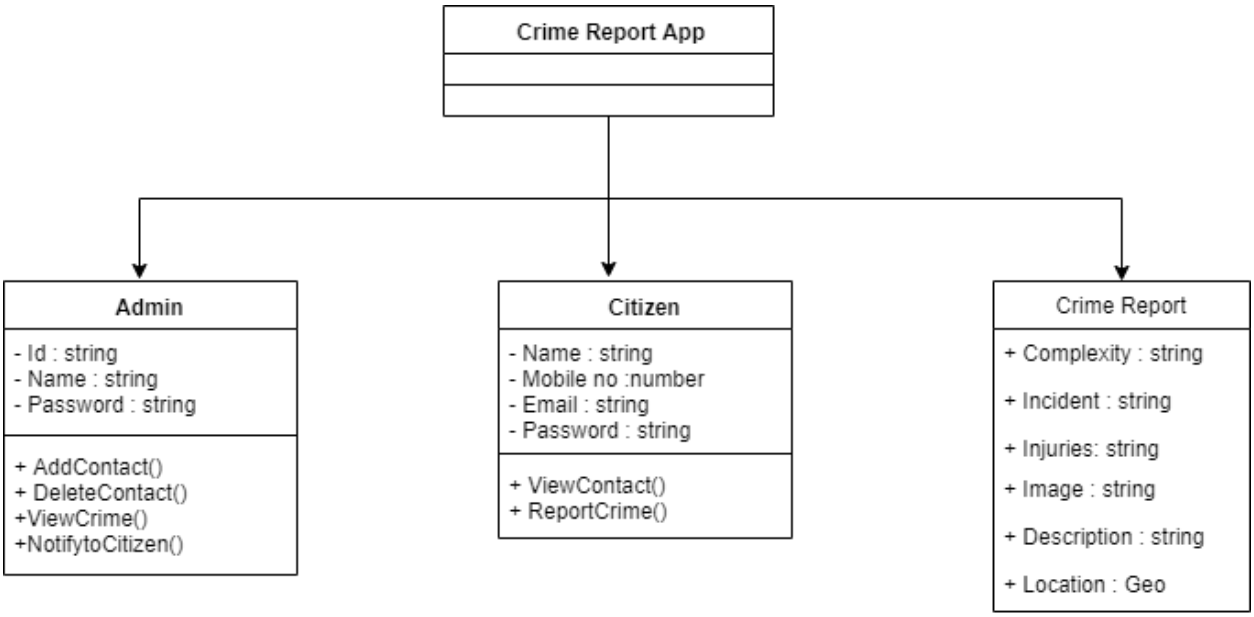


Figure 4.2 : Class Diagram

4.3. ACTIVITY DIAGRAM

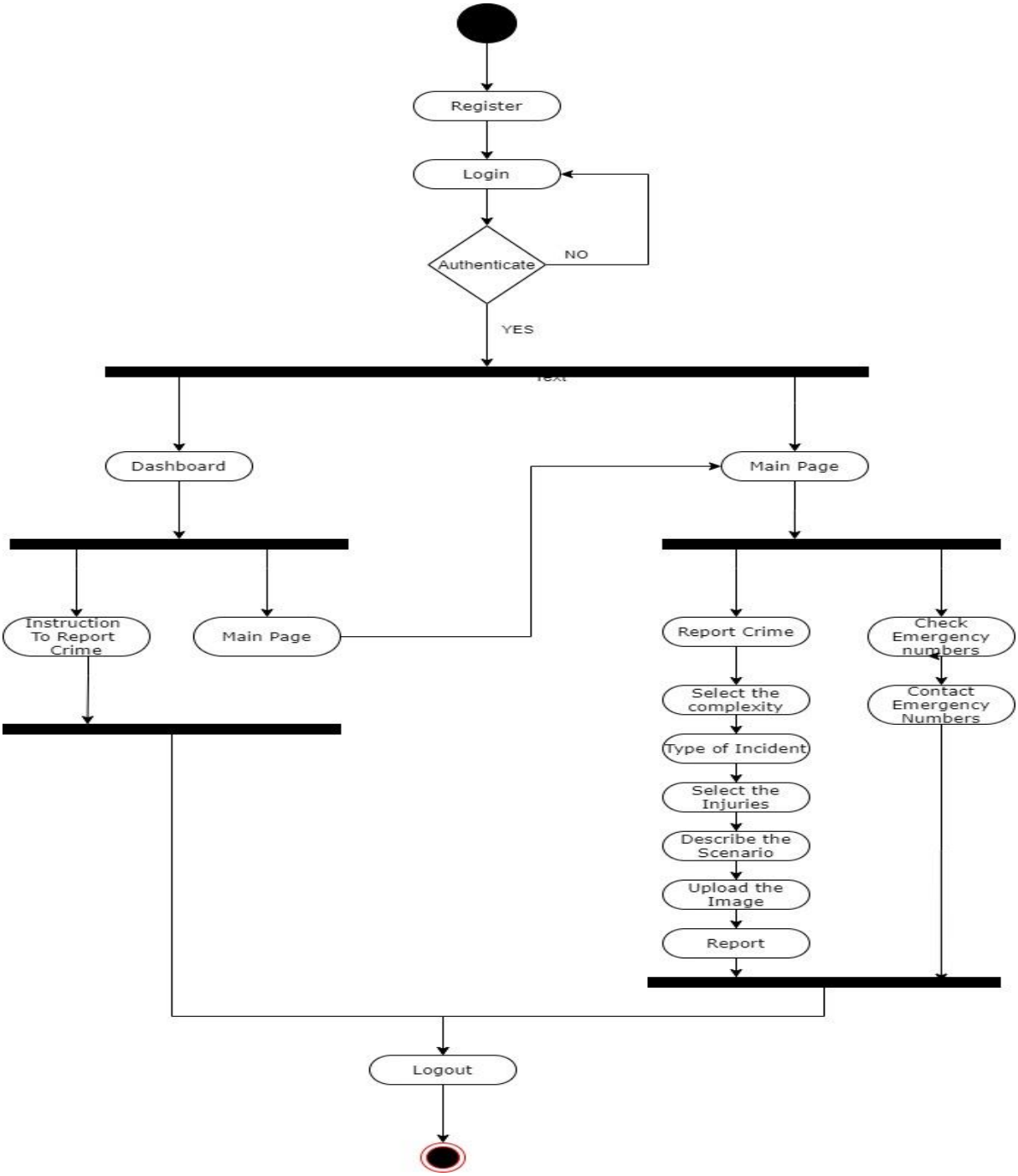


Figure 4.3(a) : Citizenship Activity Diagram

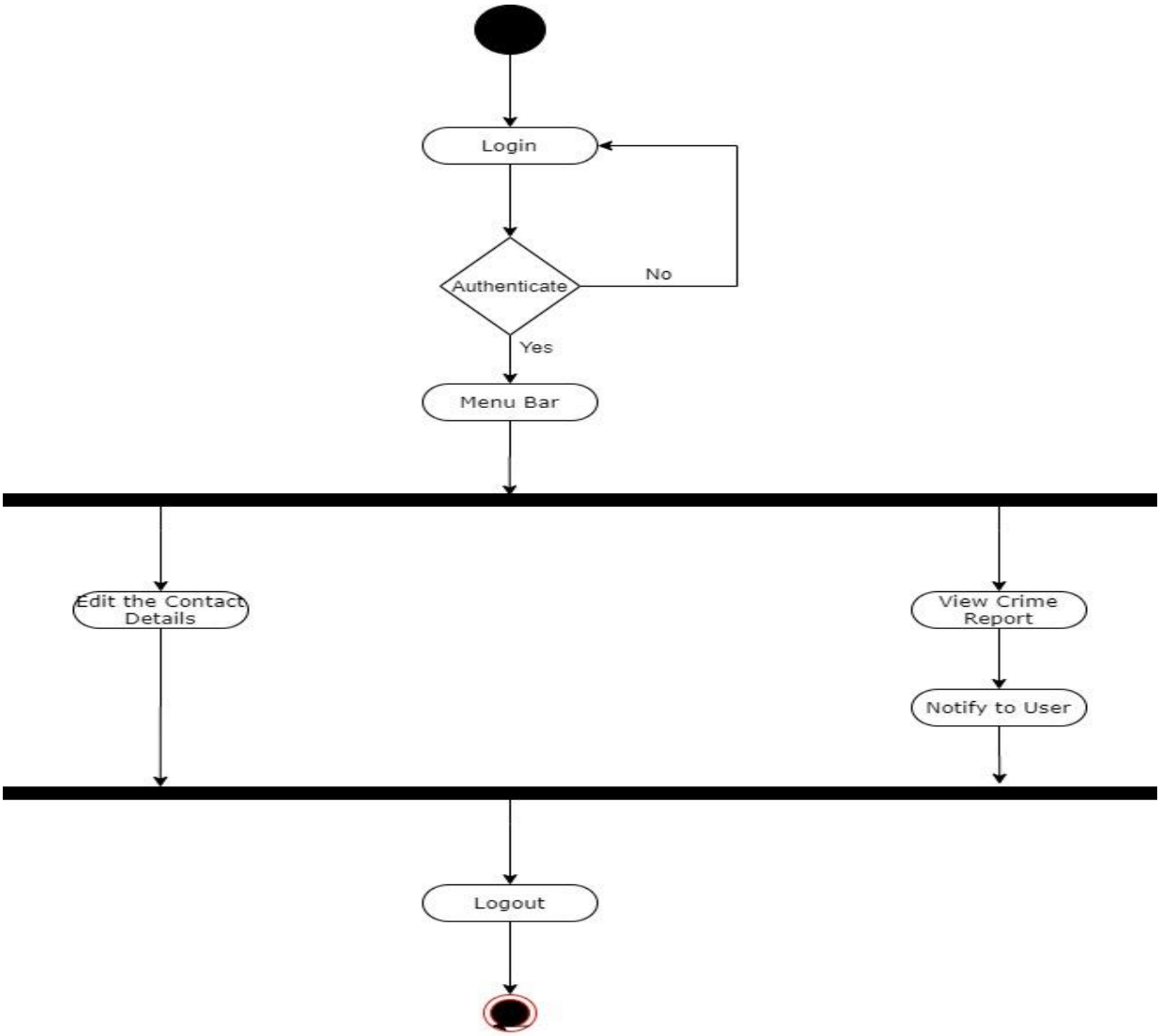


Figure 4.3(b) : Control ward activity diagram

4.4. SEQUENCE DIAGRAM

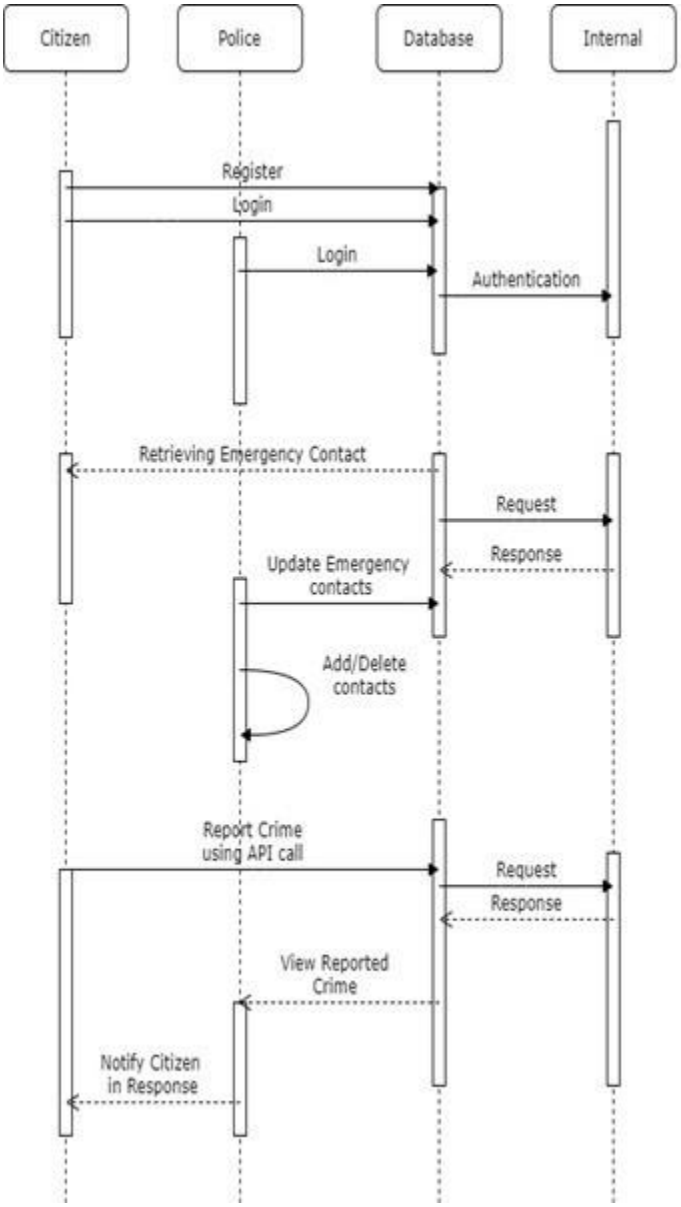


Figure 4.4 : Sequence Diagram

4.5.SYSTEM ARCHITECTURE

Using the mobile application the data which is selected by the user is sent to the firebase with the help of API calls such as speech to text and location coordinates. From the firebase the data is taken by the server which is maintained by the admin such as police and the needed action is taken against the crime report.

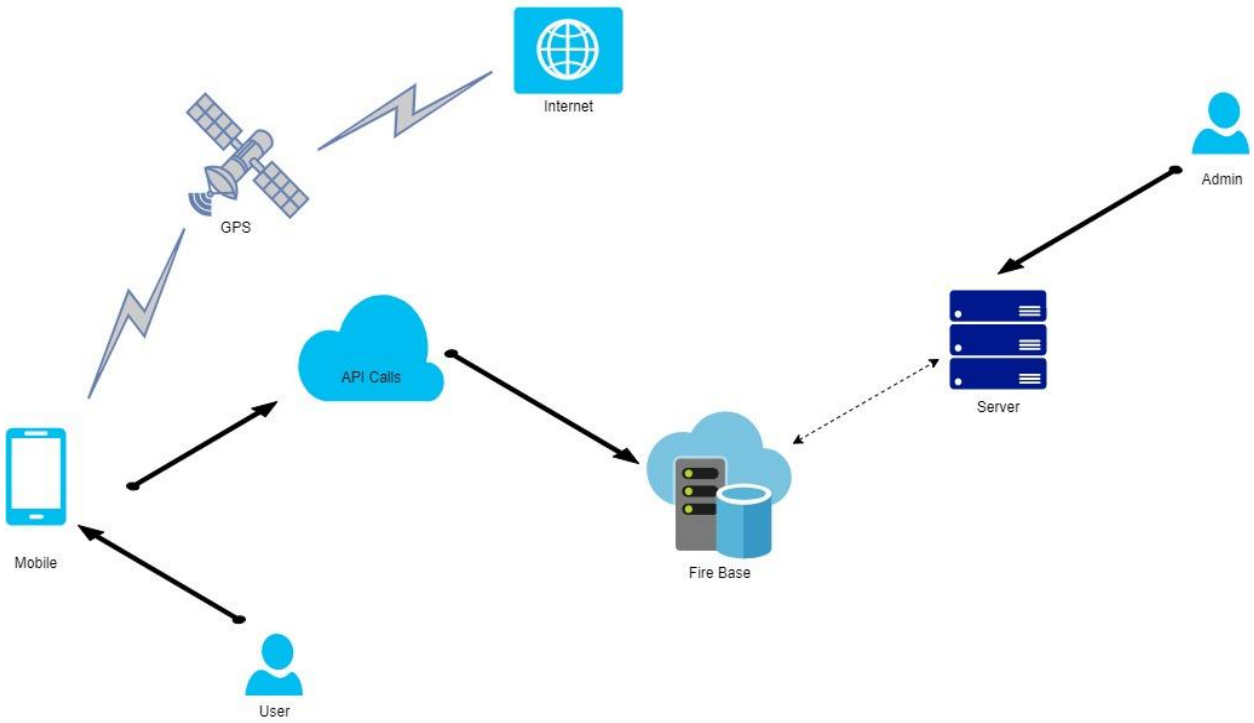


Figure 4.5 : System Architecture

4.6. TECHNOLOGY DESCRIPTION

The project is divided into two parts server side and script side.

4.6.1 Server Side

Server side is written using Flask and Firebase.

Flask

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Extensions are updated far more frequently than the core Flask program.

HTML

HTML is a hypertext markup language which is in reality a backbone of any website. Every website can't be structured without the knowledge of html. If we make our web page only with the help of html, than we can't add many of the effective features in a web page, for making a web page more effective we use various platforms such as CSS. So here we are using this language to make our web pages more effective as well as efficient. And to make our web pages dynamic we are using Java script.

Firebase

Firebase is a Backend-as-a-Service — BaaS — that started as a YC11startup and grew up into a next-generation app-development platform on Google Cloud Platform.

Firebase frees developers to focus crafting fantastic user experiences. You don't need to manage servers. You don't need to write APIs. Firebase is your server, your API and your datastore, all written so generically that you can modify it to suit most needs. Yeah, you'll occasionally need to use other bits of the Google Cloud for your advanced applications.

4.6.2. Client Side

Client side is written using Flutter.

Flutter SDK

Flutter is Google’s UI toolkit for building beautiful, natively compiled applications for mobile, web, and desktop from a single codebase. Flutter apps are written in the Dart language and make use of many of the language's more advanced features. Flutter runs in the Dart virtual machine which features a justin-time execution engine.

Flutter's engine, written primarily in C++, provides low-level rendering support using Google's Skia graphics library. Additionally, it interfaces with platformspecific SDKs such as those provided by Android and iOS. The Flutter Engine is a portable runtime for hosting Flutter applications. It implements Flutter's core libraries, including animation and graphics, file and network I/O, accessibility support, plugin architecture, and a Dart runtime and compile toolchain. Most developers will interact with Flutter via the Flutter Framework, which a modern, reactive framework, and a rich set of platform, layout and foundation widgets.

5. IMPLEMENTATION AND TESTING

5.1 IMPLEMENTATION

5.1.1 Mobile Application

1. **Login Authentication Module**First the citizen has to enter his username and password .These details are compared with the details in the database .If correct, it then redirects to dashboard page.else he need to sign in for the app

```
import 'package:firebase_auth/firebase_auth.dart';
Future<void> validateandSubmit() async {
    try {
      String temp = _email.text.toString();

      FirebaseUser user = (await FirebaseAuth.instance
              .signInWithEmailAndPassword(
                  email: temp, password: _password.text.toString()))
          .user;
      final snackBar = SnackBar(
        content: Text(
          'welcome ${user.email}',
          style: TextStyle(
            color: Colors.black,
            fontSize: 20,
          ),
          textAlign: TextAlign.center,
        ),
        backgroundColor: Colors.green,
      );
      _scaffoldKey.currentState.showSnackBar(snackBar);
      Navigator.pushNamed(context, '/first', arguments: mapData);

      print('Signed in: ${user.uid}');
    } catch (e) {
      print("exception:" + e.message);
      final snackBar = SnackBar(
        content: Text(
          e.message,
          style: TextStyle(
            color: Colors.black,
            fontSize: 20,
          ),
          textAlign: TextAlign.center,
```

```
        ),
        backgroundColor: Colors.red[600],
      );
      _scaffoldKey.currentState.showSnackBar(snackBar);
    }
  }
```

2. **Contact Module**

   In the current module the contacts are retrieving from database which are being managed by police and he can directly call through app where here used api called url_launcher

```
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:url_launcher/url_launcher.dart';
  void _launchCaller(String contact) async {
    var url = "tel:${contact}";
    if (await canLaunch(url)) {
      await launch(url);
    } else {
      throw 'Could not place call';
    }
  }
  Widget _buildListItem(BuildContext context, DocumentSnapshot document) {
    String branch = document['Branch'].toString();
    String contact = document['contact'].toString();
    return ListTile(
      title: Container(
        child: Column(
          children: <Widget>[
            Row(
              children: [
                Expanded(
                  child: Text(
                    "Area",
                    style: Theme.of(context).textTheme.headline,
                  ),
                ),
                Container(
                  decoration: const BoxDecoration(
                    color: Color(0xffdd),
                  ),
                  padding: const EdgeInsets.all(10.0),
                  child: Text(
                    branch,
                    style: Theme.of(context).textTheme.display1,
```

```
                  ),
                ),
              ],
            ),
            InkWell(
                child: Icon(
                  Icons.call,
                  color: Colors.green[900],
                ),
                onTap: () {
                  _launchCaller(contact);
                },
              )
            ],
          ),
          SizedBox(height: 30.0),
        ],
    );
  }
}
```

3. **Injuries Module:**

   In this module the inbuilt api for body parts where the reporter can select body parts of injuries where he can this can be helpful for police so that they will be knowledged about the complexity

```
import'package:human_anatomy/human_anatomy.dart';


classHumanAnatomyPageextendsStatelessWidget {
var _finalBodyPartList = [];

@override
  Widget build(BuildContext context) {
returnHumanAnatomy(
onChanged: bodyPartList,
    );
  }
voidbodyPartList(List value) {
      _finalBodyPartList = value;
    }
```

```
}
```

4. **Speech to Text Module**

   To make user friendly we included speech to text so that he can describe the whole scenario in detail through his voice which saves him a lot of time

```dart
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'dart:async';
import 'dart:math';
import 'package:speech_to_text/speech_recognition_error.dart';
import 'package:speech_to_text/speech_recognition_result.dart';
import 'package:speech_to_text/speech_to_text.dart';

  Future<void> initSpeechState() async {
    bool hasSpeech = await speech.initialize(
        onError: errorListener, onStatus: statusListener);
    if (hasSpeech) {
      _localeNames = await speech.locales();

      var systemLocale = await speech.systemLocale();
      _currentLocaleId = systemLocale.localeId;
    }

    if (!mounted) return;

    setState(() {
      _hasSpeech = hasSpeech;
    });
  }


  void startListening() {
    lastWords = "";
    lastError = "";
    speech.listen(
        onResult: resultListener,
        listenFor: Duration(seconds: 90),
        localeId: "en_US",
        onSoundLevelChange: soundLevelListener,
        cancelOnError: true,
        partialResults: true);
```

```
  setState(() {});
}

void stopListening() {
  speech.stop();
  setState(() {
    level = 0.0;
  });
}

void cancelListening() {
  speech.cancel();
  setState(() {
    level = 0.0;
  });
}
```

5. **Location and Image Module**

   As reporter is taking the live proof of image his live location also forwarded to police which makes lot more easier to police

```
import 'package:geolocator/geolocator.dart';
import 'package:image_picker/image_picker.dart';
 Widget _previewImage() {
    return FutureBuilder<File>(
        future: _imageFile,
        builder: (BuildContext context, AsyncSnapshot<File> snapshot) {
          if (snapshot.connectionState == ConnectionState.done &&
              snapshot.data != null) {
           mapData['image'] = snapshot.data.path;
            return Image.file(
              snapshot.data,
              fit: BoxFit.cover,);
          } else if (snapshot.error != null) {
            return Padding(
              padding: const EdgeInsets.only(top: 60.0),
              child: const Text(
                'Error picking image.',
```

```
                textAlign: TextAlign.center,
              ), );
          } else {
            return Padding(
              padding: const EdgeInsets.only(top: 60.0),
              child: const Text(
                'You have not yet picked an image.',
                textAlign: TextAlign.center,
              ),);} });
  }
  void _onImageButtonPressed(ImageSource source) {
    setState(() {
      _imageFile = ImagePicker.pickImage(source: source);
    }); }
  sampleFunction() async {
    _p = await Geolocator().getCurrentPosition(desiredAccuracy: LocationAccu
racy.high);
    mapData["Latitude"] = _p.latitude.toString();
    mapData["Logitude"] = _p.longitude.toString();
    print(mapData);
    setState(() {
    }); }
                Container(
                      padding: EdgeInsets.all(10.0),
                      alignment: Alignment.bottomRight,
                      child: FloatingActionButton(
                        backgroundColor: Colors.black87,
                        child: Icon(Icons.arrow_forward),
                        onPressed: () {
                          sampleFunction();
                            Navigator.pushNamed(context, route,arguments: ma
pData);
                        },), ),
```

6. Upload to Database Module

The whole data is uploaded to database in the given code the datatype mapdata contains all the information as shown in the code.

```
import 'package:firebase_storage/firebase_storage.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
    Future uploadImage(Map mapData,var now,BuildContext context) async{
      String fileName = basename(mapData['image']);
      final StorageReference firebaseStorageRef = FirebaseStorage.instance.r
ef().child(fileName);
      final StorageUploadTask uploadTask = firebaseStorageRef.child(now.toSt
ring()+".jpg").putFile(File(mapData['image']));
      var img_url = await (await uploadTask.onComplete).ref.getDownloadURL()
;
      url = img_url.toString();
      upload_Database(mapData, context);
    }
  void upload_Database(Map mapData, BuildContext context) async{
    var dbTime = new DateTime.now();
    GeoPoint _geoPoint=new GeoPoint(double.parse(mapData['Latitude']), doubl
e.parse(mapData['Logitude']));
    String email = await inputData();
    String docum = email.toString();
    docum = docum+" "+dbTime.toString();
    var doc= Firestore.instance.collection('crime').document(docum);
    doc.setData({
      'Complexity':mapData['priority'],
      'Crime-type':mapData['incidentType'],
      'Description':mapData['Description'],
      'Injuries':mapData['bodyPart'],
      'location':_geoPoint,
      'date':dbTime,
      'Image_url':url,
      'email':email.toString(),
    });  }
```

5.1.2 Web Application

1. **Login Module**

   Since there will be only one credential the police need to login to the credential given if matches it will be redirecting to dashboard else denied

```
@app.route('/login',methods=['POST','GET'])
def details():
    if request.form['user'] == username and request.form['password'] == pwd:
        return render_template('button3.html')
    else:
        return "Invalid Credentials. Please try again."
```

2. **Contact Module**

   As mentioned above in mobile app that it contacts are retrieved from database that will that will be maintained in this module

```
@app.route('/contact_information',methods = ['GET','POST'])
def contact_information():
    contact = db.collection(u'Contacts').stream()
    return render_template('Phone_Dir.html',contact = contact)
@app.route('/delete_details/<id>')
def delete_details(id):
    x=db.collection(u'Contacts').document(id).delete()
    return '<h1>deleted</h1>' if  x else "<h1> not deleted</h1>"

@app.route('/add_details',methods=['POST'])
def add_details():
    # SNo=request.form['sno']
    branch=request.form['branch']
    contact=request.form['contact']
    x= {
        # u'Sno':SNo,
        u'contact' :contact,
        u'Branch':branch
    }
    g = db.collection(u'Contacts').document('branch').set(x)
    if g:
        return 'Done'
    else:
        return 'fill complete form'
```

3. Crime Report Module

The reports that reported by citizen are maintained here

```
@app.route('/crime_record',methods = ['GET','POST'])
def crime_record():
    crime = db.collection(u'crime').stream()
    return render_template('index.html',dicti = crime)
```

4. Notification Module

As soon police has been updated then the citizen will notified fron this module

```
import smtplib
@app. route ('/notify', methods = ['POST'])
def notify():
    sender = "abcd@gmail.com"
    rec = request.form['email']
    server = smtplib.SMTP('smtp.gmail.com',587)
    server.ehlo()
    server.starttls()
    server.login(sender,'123456')
    server.sendmail(sender,sender,"Your crime report has been received")
    return 'Notification  has been sent'
```

5.2 Testing

5.2.1 Mobile Application

1.Login and Signup

The citizen need to login if there are credentials else citizen needs to sign up as shown below

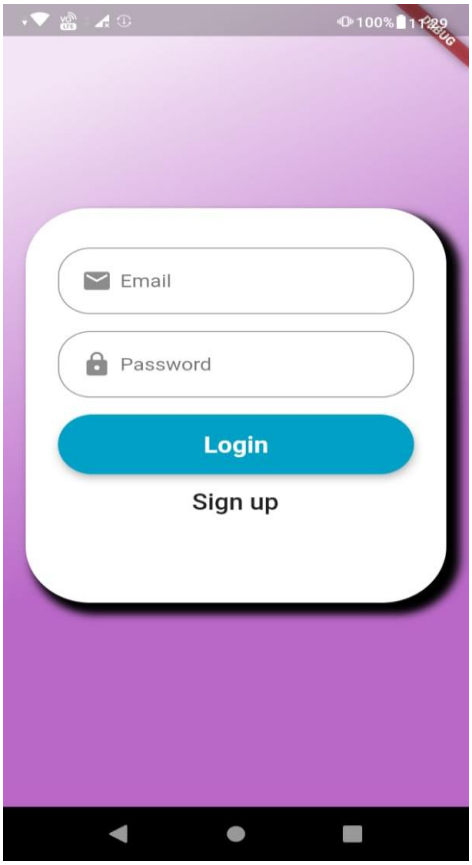


Figure 5.2.1(a)Login

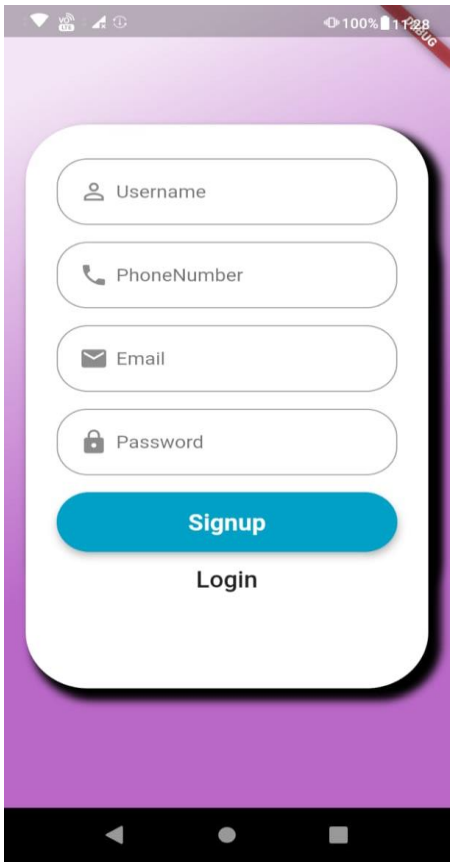


Figure 5.2.1(b) Signup

2.Dashboard and Menubar

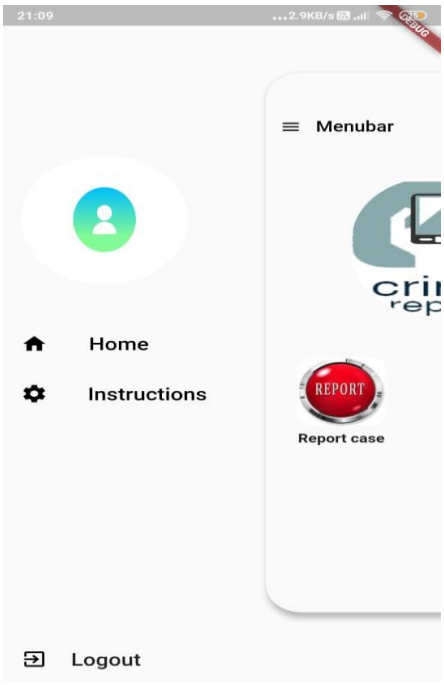


Figure 5.2.1(c) Dashboard &Menubar

3.Contacts

The user can contact from icon given a side directly as show in the below screenshot.

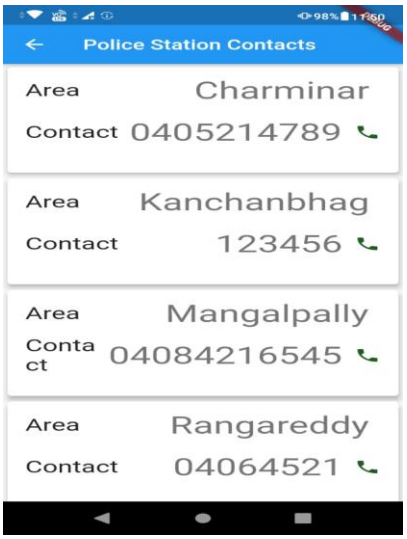


Figure 5.2.1(d) View Contacts

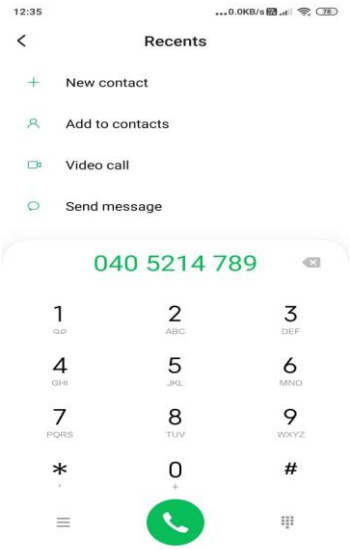


Figure 5.2.1(e) Call

4.Injuries

If a reporter see any injuries citizen can select the body parts as shown in below figure and that will be mapped to the parts of dictionary

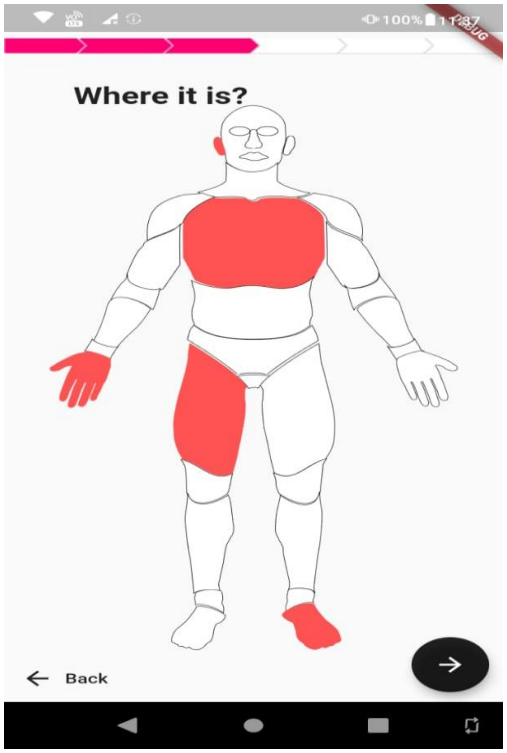


Figure 5.2.1(f) Injured Body Parts

5.Uploading an Image

Here citizen will be giving the live image proof which makes police much easier to know the case.

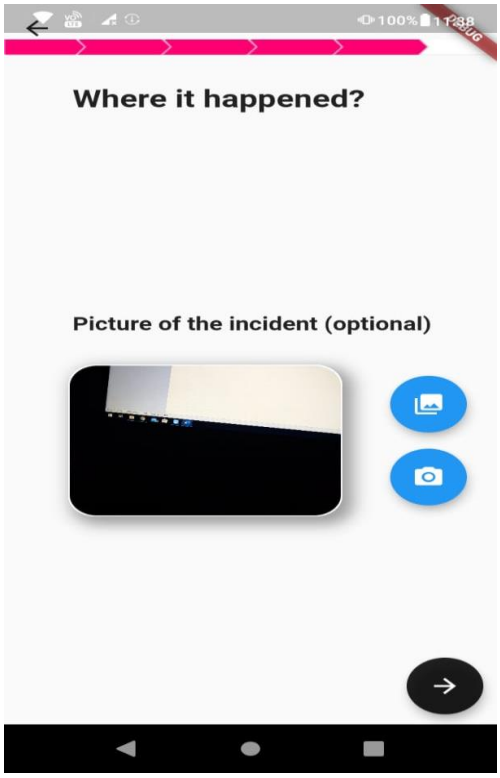


Figure5.2.1(g) Image

6.Incident Summary Module

The whole Summary will be seen at the last as shown in given figure what ever the reporter has selected and elaborated about the situation will be displayed so that he can recheck what ever he has mentioned.

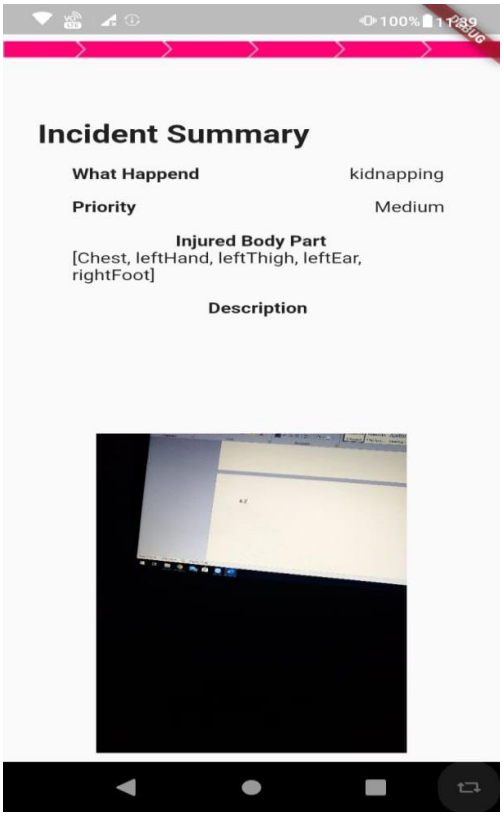


Figure 5.2.1(h) Incident Summary

5.1.3 Web Application

1. **Login**

Figure 5.2.2(a) Login

2. **Contact Details**

The contact details are updated as shown in below figure

Figure 5.2.2(b) Contacts Details

3. Crime Report Details

The crime report that are reported will be seen here

Figure 5.2.2(c) Crimes

4. Redirecting to Location

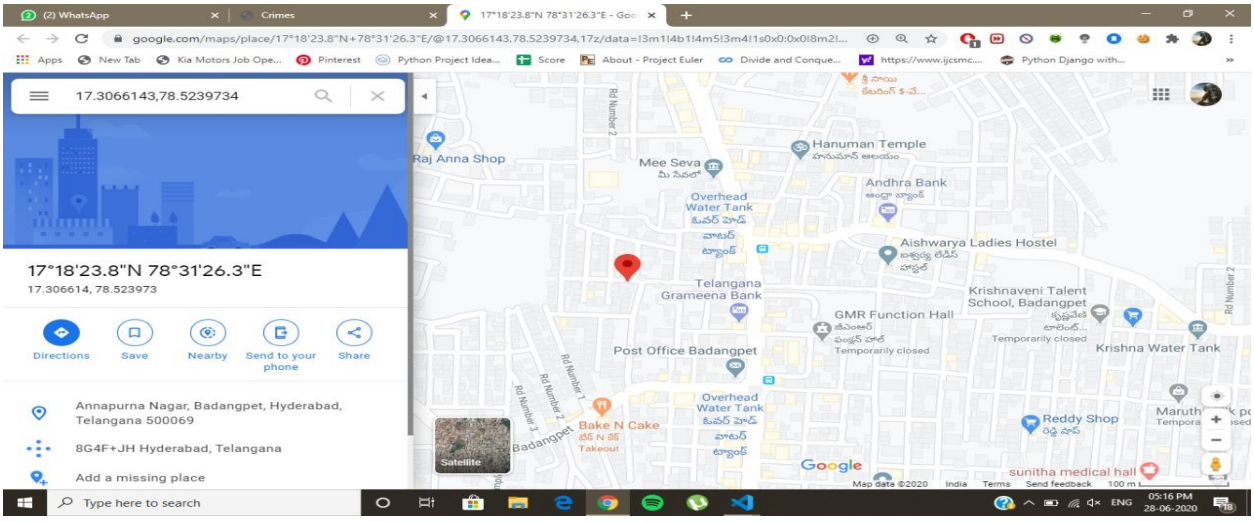


Figure 5.2.2(d) View Location

6. CONCLUSION & FUTURE SCOPE

6.1 Conclusion

- In the modern world, the use of computers and Mobile phones is becoming rampant, utilizing this factor will be increased.

- This report presents simple ,cost-effective and most importantly effective online crime reporting

- This application is useful to both citizen and police.

- The fact that very few citizens even know the phone numbers of police stations and can therefore not contact them in case of an emergency through this app maintaing data of emergency contacts through area wise might help citizen to be updated

- To safeguard people‘s lives and property, application for crime reporting and monitoring process is key as this will effectively bridge the communication gap between the police and the general public in fighting crime.

- The ultimate result of this study is to therefore develop a mobile crime fighting application that will be used for crime reporting and monitoring.

6.2 Future Scope

- The Mobile crime reporting can further be enhanced to enable other multimedia attachments and sharing so that the police can have graphic, audio or video evidence of the crime being reported.

- The other feature that can be incorporated into the application is a live chat option with the police. This will lead to reducing the mistrust that is currently there between the police and the general public . Most of the fake reporting also be reduced.

REFERENCES

1.Flutter:https://flutter.dev/

2. Firebase API: https://firebase.google.com/

3. Flask: https://flask.palletsprojects.com/en/1.1.x/