LOVELY PROFESSIONAL UNIVERSITY

PROJECT NAME: ANDROID APPLICATION FOE CRIMINAL MANAGEMENT AND WOMAN SAFETY

Submitted by -   Indranil Maji, Saptarshi Chatterjee, Subham Mahendru
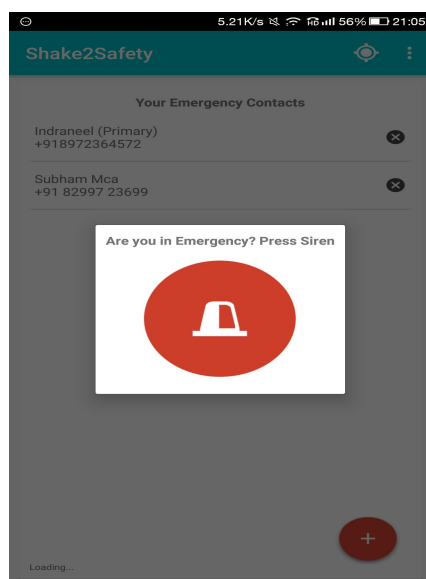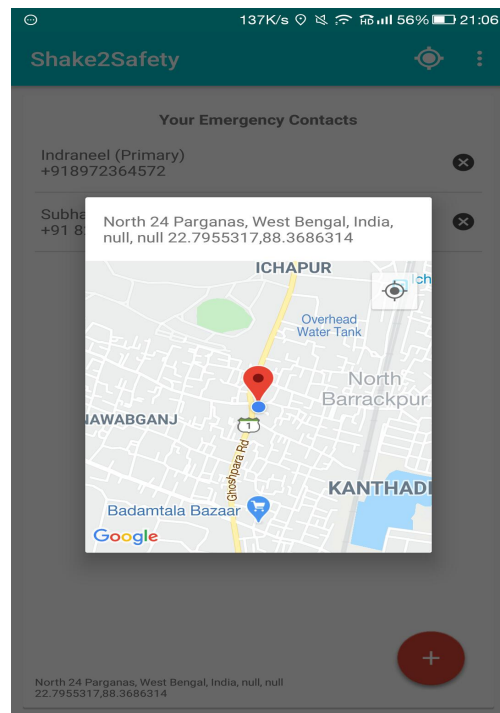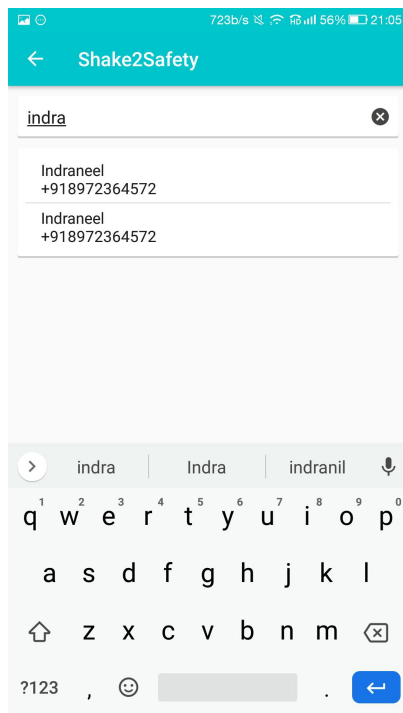
Subject - ADVANCE JAVA
Section - D1918
Reg.no – 11905492, 11906426, 11918825

-OUR MAIN MOTO IS TO MAKE EVERYONE AWARE WHAT KING OF CRIME AND CIMINALS ARE PRESNT IN THE SOCIATY AND ALSO MAKE AN APP FOR WOMAN WHO CAN USE IT IN TIME OF RISK JUST FROM A SHAKE.

WE HAVE DEVIDED THIS APP IN TWO PARTS SO WILL BE UPLOAD ALL THE XML AND JAVA FILE AS PER THE MODULES AND SCREEN SHOTS TOO.
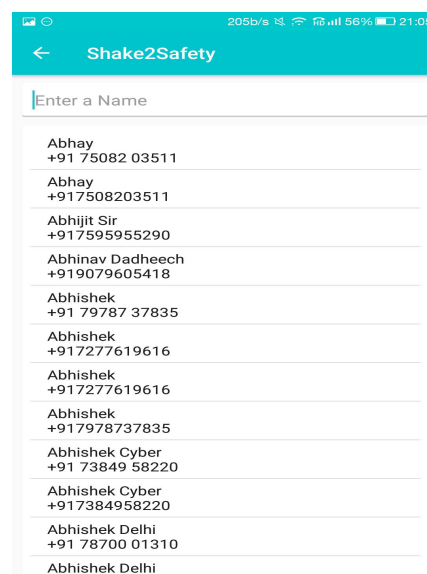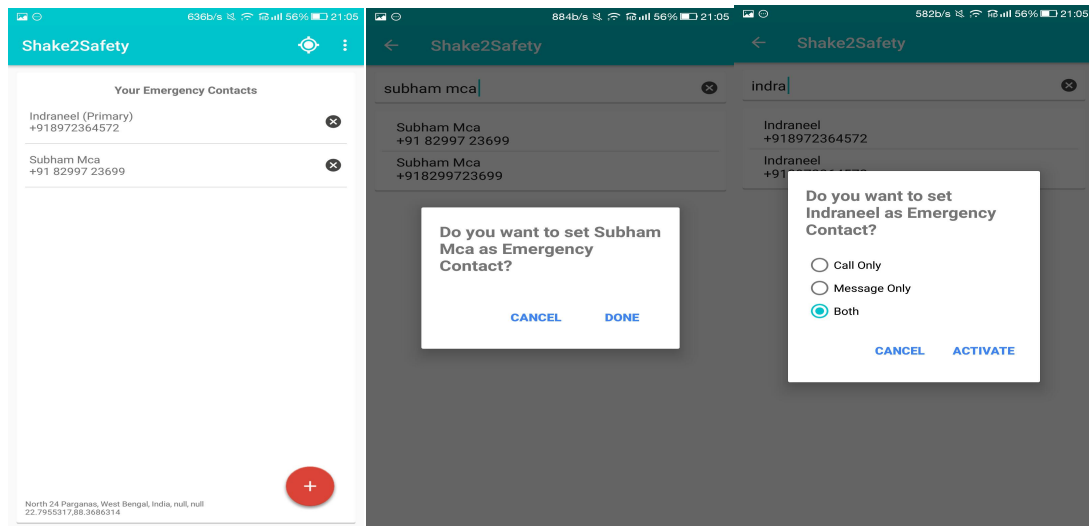
ONCE THE NUMBERS ARE ADDED WHEN U SHAKE YOUR PHONE THE PERSON ADDED IN THE LIST WILL GET AUTOMATIC YOUR CORDINATES AND

CALL BY YOU AND YOU CAN EVEN UP ON A SIREN WHICH CAN TURN HELPFULL.

XML CODE:

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:layout_width="match_parent"
android:layout_height="match_parent">
    <Button android:id="@+id/dialog_button"
android:layout_width="100px"
android:layout_height="wrap_content"
android:layout_marginTop="5dp" android:layout_marginRight="5dp"
android:text=" Ok "/>
</RelativeLayout>

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:paddingBottom="20sp"
android:layout_width="wrap_content"
android:layout_height="wrap_content">
    <TextView android:textStyle="bold" android:id="@+id/tvTitle"
android:padding="10sp" android:layout_width="wrap_content"
android:layout_height="wrap_content" android:text="Are you in
Emergency? Press Siren"/>
    <Button android:id="@+id/bSiren"
android:background="@drawable/siren"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="5sp"
android:layout_below="@+id/tvTitle"
android:layout_centerHorizontal="true"/>
</RelativeLayout>




<?xml version="1.0" encoding="utf-8"?>
```

```xml
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:padding="20sp" android:layout_width="match_parent"
android:layout_height="match_parent">
    <TextView android:textSize="22sp" android:textStyle="bold"
android:id="@+id/tvTitleDialog"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/set_sensitivity"/>
    <TextView android:id="@+id/tvSensitivity"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="20sp"
android:layout_below="@+id/tvTitleDialog"/>
    <SeekBar android:id="@+id/sbSensitivity"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_marginTop="10sp" android:maxHeight="3sp"
android:progressDrawable="@drawable/seekbar_progress"
android:thumb="@drawable/thumb"
android:layout_below="@+id/tvSensitivity"/>
    <Button android:textStyle="bold" android:textColor="#4184f3"
android:id="@+id/bDone"
android:background="@drawable/simple_button_selector_transpare
nt" android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="20sp" android:text="@string/done"
android:layout_below="@+id/sbSensitivity"
android:layout_alignParentRight="true"/>
</RelativeLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<DateTimeView
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:textAppearance="@style/TextAppearance.Compat.Notificat
ion.Time" android:id="@+id/time"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:singleLine="true"/>
```

JAVA PART (BACKEND)

```java
package android.support.v4.app;

import android.app.Notification;
import android.os.Binder;
import android.os.IBinder;
import android.os.IInterface;
import android.os.Parcel;
import android.os.RemoteException;

public interface INotificationSideChannel extends IInterface {

    public static abstract class Stub extends Binder implements
INotificationSideChannel {
        private static final String DESCRIPTOR =
        static final int TRANSACTION_cancel = 2;
        static final int TRANSACTION_cancelAll = 3;
        static final int TRANSACTION_notify = 1;

        private static class Proxy implements
INotificationSideChannel {
            private IBinder mRemote;

            public String getInterfaceDescriptor() {
                return Stub.DESCRIPTOR;
            }

            Proxy(IBinder iBinder) {
                this.mRemote = iBinder;
            }

            public IBinder asBinder() {
                return this.mRemote;
            }

            public void notify(String str, int i, String str2,
Notification notification) throws RemoteException {
                Parcel obtain = Parcel.obtain();
                try {
                    obtain.writeInterfaceToken(Stub.DESCRIPTOR);
                    obtain.writeString(str);
                    obtain.writeInt(i);
                    obtain.writeString(str2);
                    if (notification != null) {
                        obtain.writeInt(1);
                        notification.writeToParcel(obtain, 0);
                    } else {
                        obtain.writeInt(0);
                    }
                    this.mRemote.transact(1, obtain, null, 1);
```

```java
                } finally {
                    obtain.recycle();
                }
            }

            public void cancel(String str, int i, String str2)
    throws RemoteException {
                Parcel obtain = Parcel.obtain();
                try {
                    obtain.writeInterfaceToken(Stub.DESCRIPTOR);
                    obtain.writeString(str);
                    obtain.writeInt(i);
                    obtain.writeString(str2);
                    this.mRemote.transact(2, obtain, null, 1);
                } finally {
                    obtain.recycle();
                }
            }

            public void cancelAll(String str) throws
    RemoteException {
                Parcel obtain = Parcel.obtain();
                try {
                    obtain.writeInterfaceToken(Stub.DESCRIPTOR);
                    obtain.writeString(str);
                    this.mRemote.transact(3, obtain, null, 1);
                } finally {
                    obtain.recycle();
                }
            }
        }

        public IBinder asBinder() {
            return this;
        }

        public Stub() {
            attachInterface(this, DESCRIPTOR);
        }

        public static INotificationSideChannel
    asInterface(IBinder iBinder) {
            if (iBinder == null) {
                return null;
            }
            IInterface queryLocalInterface =
    iBinder.queryLocalInterface(DESCRIPTOR);
            if (queryLocalInterface == null
    || !(queryLocalInterface instanceof INotificationSideChannel)) {
```

```java
            return new Proxy(iBinder);
        }
        return (INotificationSideChannel) queryLocalInterface;
    }

    public boolean onTransact(int i, Parcel parcel, Parcel
parcel2, int i2) throws RemoteException {
        String str = DESCRIPTOR;
        if (i == 1) {
            parcel.enforceInterface(str);
            notify(parcel.readString(), parcel.readInt(),
parcel.readString(), parcel.readInt() != 0 ? (Notification)
Notification.CREATOR.createFromParcel(parcel) : null);
            return true;
        } else if (i == 2) {
            parcel.enforceInterface(str);
            cancel(parcel.readString(), parcel.readInt(),
parcel.readString());
            return true;
        } else if (i == 3) {
            parcel.enforceInterface(str);
            cancelAll(parcel.readString());
            return true;
        } else if (i != 1598968902) {
            return super.onTransact(i, parcel, parcel2, i2);
        } else {
            parcel2.writeString(str);
            return true;
        }
    }
}

    void cancel(String str, int i, String str2) throws
RemoteException;

    void cancelAll(String str) throws RemoteException;

    void notify(String str, int i, String str2, Notification
notification) throws RemoteException;
}




package android.support.v4.app;

import androidx.core.app.RemoteActionCompat;
import androidx.versionedparcelable.VersionedParcel;
```

```java
public final class RemoteActionCompatParcelizer extends
androidx.core.app.RemoteActionCompatParcelizer {
    public static RemoteActionCompat read(VersionedParcel
versionedParcel) {
        return
androidx.core.app.RemoteActionCompatParcelizer.read(versionedP
arcel);
    }

    public static void write(RemoteActionCompat
remoteActionCompat, VersionedParcel versionedParcel) {

androidx.core.app.RemoteActionCompatParcelizer.write(remoteAct
ionCompat, versionedParcel);
    }
}




package android.support.v4.media;

import android.content.ComponentName;
import android.content.Context;
import android.media.browse.MediaBrowser;
import android.os.Bundle;
import java.util.List;

class MediaBrowserCompatApi21 {
    static final String NULL_MEDIA_ITEM_ID =
"android.support.v4.media.MediaBrowserCompat.NULL_MEDIA_ITEM";

    interface ConnectionCallback {
        void onConnected();

        void onConnectionFailed();

        void onConnectionSuspended();
    }

    static class ConnectionCallbackProxy<T extends
ConnectionCallback> extends
android.media.browse.MediaBrowser.ConnectionCallback {
        protected final T mConnectionCallback;

        public ConnectionCallbackProxy(T t) {
            this.mConnectionCallback = t;
        }
```

```java
        public void onConnected() {
            this.mConnectionCallback.onConnected();
        }

        public void onConnectionSuspended() {
            this.mConnectionCallback.onConnectionSuspended();
        }

        public void onConnectionFailed() {
            this.mConnectionCallback.onConnectionFailed();
        }
    }

    static class MediaItem {
        public static int getFlags(Object obj) {
            return ((android.media.browse.MediaBrowser.MediaItem)
obj).getFlags();
        }

        public static Object getDescription(Object obj) {
            return ((android.media.browse.MediaBrowser.MediaItem)
obj).getDescription();
        }

        private MediaItem() {
        }
    }

    interface SubscriptionCallback {
        void onChildrenLoaded(String str, List<?> list);

        void onError(String str);
    }

    static class SubscriptionCallbackProxy<T extends
SubscriptionCallback> extends
android.media.browse.MediaBrowser.SubscriptionCallback {
        protected final T mSubscriptionCallback;

        public SubscriptionCallbackProxy(T t) {
            this.mSubscriptionCallback = t;
        }

        public void onChildrenLoaded(String str,
List<android.media.browse.MediaBrowser.MediaItem> list) {
            this.mSubscriptionCallback.onChildrenLoaded(str,
list);
        }
```

```java
        public void onError(String str) {
            this.mSubscriptionCallback.onError(str);
        }
    }

    public static Object
createConnectionCallback(ConnectionCallback connectionCallback)
{
        return new ConnectionCallbackProxy(connectionCallback);
    }

    public static Object createBrowser(Context context,
ComponentName componentName, Object obj, Bundle bundle) {
        return new MediaBrowser(context, componentName,
(android.media.browse.MediaBrowser.ConnectionCallback) obj,
bundle);
    }

    public static void connect(Object obj) {
        ((MediaBrowser) obj).connect();
    }

    public static void disconnect(Object obj) {
        ((MediaBrowser) obj).disconnect();
    }

    public static boolean isConnected(Object obj) {
        return ((MediaBrowser) obj).isConnected();
    }

    public static ComponentName getServiceComponent(Object obj) {
        return ((MediaBrowser) obj).getServiceComponent();
    }

    public static String getRoot(Object obj) {
        return ((MediaBrowser) obj).getRoot();
    }

    public static Bundle getExtras(Object obj) {
        return ((MediaBrowser) obj).getExtras();
    }

    public static Object getSessionToken(Object obj) {
        return ((MediaBrowser) obj).getSessionToken();
    }

    public static Object
createSubscriptionCallback(SubscriptionCallback
subscriptionCallback) {
```

```java
        return new
SubscriptionCallbackProxy(subscriptionCallback);
    }

    public static void subscribe(Object obj, String str, Object
obj2) {
        ((MediaBrowser) obj).subscribe(str,
(android.media.browse.MediaBrowser.SubscriptionCallback) obj2);
    }

    public static void unsubscribe(Object obj, String str) {
        ((MediaBrowser) obj).unsubscribe(str);
    }

    private MediaBrowserCompatApi21() {
    }
}
package android.support.v4.media;

import android.graphics.Bitmap;
import android.media.MediaMetadata;
import android.media.Rating;
import android.os.Parcel;
import java.util.Set;

class MediaMetadataCompatApi21 {

    public static class Builder {
        public static Object newInstance() {
            return new android.media.MediaMetadata.Builder();
        }

        public static void putBitmap(Object obj, String str, Bitmap
bitmap) {
            ((android.media.MediaMetadata.Builder)
obj).putBitmap(str, bitmap);
        }

        public static void putLong(Object obj, String str, long j)
{
            ((android.media.MediaMetadata.Builder)
obj).putLong(str, j);
        }

        public static void putRating(Object obj, String str, Object
obj2) {
            ((android.media.MediaMetadata.Builder)
obj).putRating(str, (Rating) obj2);
        }
```

```java
        public static void putText(Object obj, String str,
CharSequence charSequence) {
            ((android.media.MediaMetadata.Builder)
obj).putText(str, charSequence);
        }

        public static void putString(Object obj, String str, String
str2) {
            ((android.media.MediaMetadata.Builder)
obj).putString(str, str2);
        }

        public static Object build(Object obj) {
            return ((android.media.MediaMetadata.Builder)
obj).build();
        }

        private Builder() {
        }
    }

    public static Set<String> keySet(Object obj) {
        return ((MediaMetadata) obj).keySet();
    }

    public static Bitmap getBitmap(Object obj, String str) {
        return ((MediaMetadata) obj).getBitmap(str);
    }

    public static long getLong(Object obj, String str) {
        return ((MediaMetadata) obj).getLong(str);
    }

    public static Object getRating(Object obj, String str) {
        return ((MediaMetadata) obj).getRating(str);
    }

    public static CharSequence getText(Object obj, String str) {
        return ((MediaMetadata) obj).getText(str);
    }

    public static void writeToParcel(Object obj, Parcel parcel, int
i) {
        ((MediaMetadata) obj).writeToParcel(parcel, i);
    }

    public static Object createFromParcel(Parcel parcel) {
        return MediaMetadata.CREATOR.createFromParcel(parcel);
```

```
        }

    private MediaMetadataCompatApi21() {
    }
}
```

2.CRIMINAL DATABASE

JAVA-

I.package com.example.codehead.criminalintent;

import java.util.Date;
import java.util.UUID;

//this is the model class for crimes
public class Crime {

    private UUID mId;
    private String mTitle;
  //private String mDetails;
    private Date mDate;
    private boolean mSolved;
    private String mSuspect;

    public   Crime(){
        this(UUID.randomUUID());
    }

    public Crime(UUID id) {
        mId = id;
        mDate = new Date();
    }

```java
public UUID getId() {
    return mId;
}

public String getTitle() {
    return mTitle;
}

public void setTitle(String title) {
    mTitle = title;
}

public Date getDate() {
    return mDate;
}

public void setDate(Date date) {
    mDate = date;
}

public boolean isSolved() {
    return mSolved;
}

public void setSolved(boolean solved) {
    mSolved = solved;
}
public String getSuspect(){
    return mSuspect;
}
public void setSuspect(String suspect){
    mSuspect=suspect;
}
public String getPhotoFileame(){
    return "IMG_" + getId().toString() +".jpg";
```

```java
    }

    /*public String getDetails() {
        return mDetails;
    }

    public void setDetails(String details) {
        mDetails = details;
    }*/
}
```

ii.package com.example.codehead.criminalintent;

```java
import android.content.Context;
import android.content.Intent;
import android.support.v4.app.Fragment;

import java.util.UUID;

public class CrimeActivity extends SingleFragmentActivity {

    private static final String EXTRA_CRIME_ID =

"com.example.codehead.criminalintent.crime_id";

    public static Intent newIntent(Context packageContext,
UUID crimeId) {
        Intent intent = new Intent(packageContext,
CrimeActivity.class);
        intent.putExtra(EXTRA_CRIME_ID, crimeId);
        return intent;
    }

    @Override
    protected Fragment createFragment() {
```

```java
            UUID crimeId = (UUID)
getIntent().getSerializableExtra(EXTRA_CRIME_ID);
            return CrimeFragment.newInstance(crimeId);
        }
}

iii.package com.example.codehead.criminalintent;

import android.app.Activity;
import android.app.Dialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.app.DialogFragment;
import android.support.v7.app.AlertDialog;
import android.view.LayoutInflater;
import android.view.View;
import android.widget.DatePicker;


import java.util.Calendar;
import java.util.Date;
import java.util.GregorianCalendar;

public class DatePickerFragment extends DialogFragment {
    private static final String ARG_DATE="date";
    public static final String
EXTRA_DATE="com.example.codehead.criminalintent.date";
    private DatePicker mDatePicker;
    public static DatePickerFragment newInstance(Date
date){
        Bundle args=new Bundle();
        args.putSerializable(ARG_DATE,date);
        DatePickerFragment fragment=new
DatePickerFragment();
        fragment.setArguments(args);
```

```java
            return fragment;
    }


    @Override
    public Dialog onCreateDialog(Bundle
savedInstanceState){
            View v=
LayoutInflater.from(getActivity()).inflate(R.layout.dialog_date,
null);

            Date date= (Date)
getArguments().getSerializable(ARG_DATE);
            Calendar calendar=Calendar.getInstance();
            calendar.setTime(date);
            int year=calendar.get(Calendar.YEAR);
            int month=calendar.get(Calendar.MONTH);
            int day=calendar.get(Calendar.DAY_OF_MONTH);

mDatePicker=v.findViewById(R.id.dialog_date_picker);
            mDatePicker.init(year,month,day,null);

            return new AlertDialog.Builder(getActivity())
                    .setView(v)
                    .setTitle(R.string.date_picker_title)
                    .setPositiveButton(android.R.string.ok,
new DialogInterface.OnClickListener() {
                            @Override
                            public void onClick(DialogInterface
dialog, int which) {
                                    int year=mDatePicker.getYear();
                                    int
month=mDatePicker.getMonth();
```

```java
                              int
day=mDatePicker.getDayOfMonth();
                                        Date date=new
GregorianCalendar(year,month,day).getTime();

sendResult(Activity.RESULT_OK,date);
                        }
                   }).create();
     }
     private void sendResult(int resultCode,Date date){
          if (getTargetFragment()==null)
               return;

          Intent intent=new Intent();
          intent.putExtra(EXTRA_DATE,date);

getTargetFragment().onActivityResult(getTargetRequestCode()
,resultCode,intent);
     }
}




DATA BASE-
package com.example.codehead.criminalintent.database;

import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import static
com.example.codehead.criminalintent.database.CrimeDbSche
ma.CrimeTable.Cols.DATE;
```

```java
import static
com.example.codehead.criminalintent.database.CrimeDbSche
ma.CrimeTable.Cols.SOLVED;
import static
com.example.codehead.criminalintent.database.CrimeDbSche
ma.CrimeTable.Cols.SUSPECT;
import static
com.example.codehead.criminalintent.database.CrimeDbSche
ma.CrimeTable.Cols.TITLE;
import static
com.example.codehead.criminalintent.database.CrimeDbSche
ma.CrimeTable.Cols.UUID;
import static
com.example.codehead.criminalintent.database.CrimeDbSche
ma.CrimeTable.NAME;


public class CrimeBaseHelper extends SQLiteOpenHelper {
    private static final int VERSION=1;
    private static final String
DATABASE_NAME="crimeBase.db";

    public CrimeBaseHelper(Context context) {
        super(context, DATABASE_NAME, null, VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        //create table
        db.execSQL("create table "+ NAME+"(" +"_id integer
primary key autoincrement,"+ UUID+","+ TITLE +","+ DATE
+","+ SOLVED +","+ SUSPECT +")");
    }

    @Override
```

```java
        public void onUpgrade(SQLiteDatabase db, int oldVersion,
int newVersion) {


        }
}
```

XML-

```xml
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".welcome">


    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginBottom="8dp"
        android:layout_marginEnd="8dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        android:orientation="vertical"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent">

        <Button
            android:id="@+id/addcontact"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="5dp"
            android:background="@color/colorPrimary"
            android:text="ADD CONTACT +"
            android:textColor="#ffffff"
            android:textStyle="bold" />

        <Button
            android:id="@+id/next"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Done"
            android:layout_margin="5dp"
            android:background="@color/colorPrimary"
            android:textStyle="bold"
            android:textColor="#ffffff"/>
```

```xml
        <android.support.v7.widget.RecyclerView
            android:id="@+id/recycle"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="5dp">

        </android.support.v7.widget.RecyclerView>

    </LinearLayout>
</android.support.constraint.ConstraintLayout>
```

DISTRESS AREA ALARM
```xml
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".DistressMode">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <TextView
            android:id="@+id/heading"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:gravity="center"
            android:text="DISTRESS MODE"
            android:textSize="37dp"
            android:textStyle="bold"
            android:layout_marginTop="90dp"/>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_marginTop="20dp">

            <TextView
                android:id="@+id/msgsent"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:text="MESSAGE SENT"
                android:textStyle="bold"
                android:textSize="25sp"
                android:layout_marginTop="30dp"
                android:layout_marginLeft="50dp"/>

            <ImageView
```

```xml
                android:id="@+id/msgsentimg"
                android:layout_width="50dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginTop="20dp"
                app:srcCompat="@drawable/success" />
        </LinearLayout>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_marginTop="20dp">

            <TextView
                android:id="@+id/tracking"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:text="LOCATION SENT"
                android:textStyle="bold"
                android:textSize="25sp"
                android:layout_marginTop="30dp"
                android:layout_marginLeft="50dp"/>

            <ImageView
                android:id="@+id/tarckingimg"
                android:layout_width="50dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginTop="20dp"
                app:srcCompat="@drawable/success" />

        </LinearLayout>

        <Button
            android:id="@+id/gethelp"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="GEt HELP"
            android:background="@color/colorPrimary"
            android:textColor="#ffffff"
            android:textStyle="bold"
            android:layout_marginTop="40dp"
            android:layout_marginLeft="30dp"
            android:layout_marginRight="30dp"/>
    </LinearLayout>
</android.support.constraint.ConstraintLayout>
```

LAYOUT

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

```xml
<android.support.v7.widget.CardView
android:id="@+id/card"
android:layout_width="400dp"
android:layout_height="100dp"
app:cardElevation="2dp"
android:layout_margin="2dp">

<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ImageView
        android:id="@+id/contactimg"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:layout_margin="10dp"
        app:srcCompat="@drawable/unknown_user" />

    <TextView
        android:id="@+id/contactname"
        android:layout_width="223dp"
        android:layout_height="30dp"
        android:layout_alignParentEnd="true"
        android:layout_alignParentTop="true"
        android:layout_marginEnd="0dp"
        android:layout_marginTop="13dp"
        android:text="TextView"
        android:textSize="17sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/contactno"
        android:layout_width="221dp"
        android:layout_height="30dp"
        android:layout_alignParentEnd="true"
        android:layout_below="@+id/contactname"
        android:text="TextView"
        android:textStyle="bold" />
</RelativeLayout>

</android.support.v7.widget.CardView>


</RelativeLayout>
```

SCREEN SHOTS: