



ISLAMIC UNIVERSITY OF TECHNOLOGY(IUT)
The Organization of the Islamic Cooperation(OIC)

Gazipur-1704, Dhaka, Bangladesh

Project Report Title:

Daily Life Assistant

Submitted By:

MD. Hasib Hasan

(114411)

Ibrahim Ndiaye

(114441)

Supervised By:

Ashrafal Alam Khan

Lecturer

Department of computer science and engineering (CSE)
Islamic University of Technology (IUT)
The Organization of the Islamic Cooperation (OIC)
Gazipur-1704, Dhaka, Bangladesh

Certification

This is to certify that this project entitled “**Daily Life Assistant**” is a true work of MD. Hasib Hasan (114411) & Ibrahim Ndiaye (114441) who was successfully carried out the work project under the supervision of **Ashraful Alam Khan** . This project count as our final year project which put an end to our four years program as B.Sc in Computer Science & Engineering at Islamic University of Technology (IUT) Dhaka, Bangladesh.

Authors:

Signature:

Name: MD. Hasib Hasan

Roll No: 114411

Date:

Signature:

Name: Ibrahim Ndiaye

Roll No: 114441

Date:

Supervisor:

Signature:

Name: Ashraful Alam Khan

Date:

Head of Department

Signature:

Name: Prof. Dr. M.A. Mottalib

Date:

Department of Computer Science & Engineering (CSE)

Islamic University of Technology (IUT)

Acknowledgement

Thanks to Allah (SWT), the most gracious and the most merciful. The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of people whose ceaseless cooperation made it possible, whose constant guidance and encouragement crown all efforts with success.

We are grateful to our project supervisor **Ashraful Alam Khan** for the guidance, inspiration and constructive suggestions that help us in the participation of this project.

Finally we also wish to take this opportunity to express our sincerest gratitude and heartiest thanks to the head of CSE department, **Prof. Dr. M.A. Mottalib**, as well as the Organization of the Islamic Cooperation for endless support it provides in the field of education.

Islamic University of Technology (IUT)

Gazipur, Dhaka, Bangladesh

MD. Hasib Hasan(114411)

Ibrahim Ndiaye (114441)

Abstract

We are in the age of big changes, particularly the technology revolution and this has affected all aspect of our life. In the modern day one of the greatest invention is Mobile Phone. Mobile phone mainly helps us to communicate with each other. But now the mobile phone improves so much that they can help us in our daily life. We cannot think a day without the mobile phone. It makes human life easier.

For this purpose we try build an application for mobile phone so that it could be more useful to human and may solve some of their daily problem.

Daily Life Assistant is an android mobile app which may solve your daily life problems which you face every day. This project has two major parts. One is Destination Alert and other is Mobile transection. First part will help you to reach your destination safely and other part is for automation of mobile transection system.

Table Contents

1. Introduction.....	6
2. Background Analysis.....	8
3. Motivation.....	10
4. Problem Identification.....	12
5. Goals.....	13
6. Application Overview.....	14
6.1 Destination alert.....	14
6.2 Mobile Transaction.....	15
7. Source Code of the Project.....	16
7.1 AlertDialogManager.java.....	16
7.2 ConnectionDetector.java.....	18
7.3 GPSTracker.java.....	19
7.4 MainActivity.java.....	26
7.5 MyService.java.....	35
7.6 OpeningActivity.java.....	40
8. Screenshots of the project.....	41
8.1 Project Menu.....	41
8.2 Daily life assistant.....	42
9. Limitations.....	43
10. Conclusion.....	44

Introduction

PDA (personal digital assistant) is a term for any small mobile handheld device that provides computing and information storage and retrieval capabilities for personal or business use, often for keeping schedule calendars and address book information handy. The term handheld is a synonym.

A “smartphone” or smart phone is a mobile phone with an advanced mobile operating system which combines features of a personal computer operating system with other features useful for mobile or handheld use. They typically combine the features of a cell phone with those of other popular mobile devices, such as personal digital assistant (PDA), media player and GPS navigation unit. Most smartphones can access the Internet, have a touch screen user interface, can run third-party apps, music players and are camera phones.

The smartphone is now one of the most popular devices in the world and has reached an average of around 1.5 billion users till this date. In fact it is so popular it is almost creating an era of itself; welcome to the “smartphone era”.

Android is an open source mobile operating system (OS) currently developed by Google, based on the Linux kernel and designed primarily for touch screen mobile devices such as smartphones and tablets. Android's user interface is based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input.

With over 1 million apps in the play store, android has become a very popular choice of OS for users all over the world. Despite having an abundance of apps, it still lacks some very basic apps that should be preinstalled in the system, apps which would be very useful for today's modern human's lifestyle.

Background Analysis

Questionnaire:

We have asked a couple of students in IUT about their thoughts regarding the project.

Q: What do you think about a location based app which gives you an alert whenever you are close to reaching your destination?

Student A: I think that sounds like a great idea. And it would be very useful to me as I like to travel a lot. If you could create this app it could let me relax and enjoy the ride as opposed to worry if I have passed my destination or not. I think a lot of people would find it useful for their own personal reasons.

Student B: Personally, I think that would help a lot of people, especially me! When I take the bus from my home to IUT in the mornings before class starts, I tend to fall asleep on the bus without knowing. I've had some embarrassing situations where I could not wake up before my stop. If you guys could implement this app it could wake me up when I'm asleep to let me know that I am close to my destination.

Student C: That sounds interesting! I'm actually surprised Google Maps doesn't have that feature included in their app. I find this really useful! I mean, I like to check out new places and constantly checking my phone to check if I am close or not when I use Google maps can get sometimes frustrating. But If I can put my phone away and focus on other things while the app just lets me know if I'm close to my destination, that would be really cool.

We have also asked questions to the vendors sitting at the different balance transfer booths.

(These questions were translated from Bangla to English)

Q: Brother, what is the greatest difficulty you face daily when you work?

A: Well brother, it is annoying to use all these different devices; I have to transfer balance from this phone if a customer is using Grameen phone sim, then I have to transfer balance from another phone if a customer is using Banglalink, it is a hassle but we are now getting used to it.

B: Brother, writing everything down in this notebook during and after every transaction is a nuisance. Frankly I don't like seeing this book.

Motivation

- The main motivation for our project is to increase interaction between humans and Smart phones, and to assist people with their daily life and make it easier for them to go through the day. In life there are already many important things to worry about. We want to try to help people worry less about certain aspects in life, to the best of our ability.
- With around 1.5 billion users in the world, Smart phones are used by a large population around the world and have become essential to our daily lives. Our thought was to take advantage of the worldwide trend and to maximize the potential of those devices by solving our daily problems and thus making them our friend.
- Every day, we are facing small but irritating problems in life and it seems that some solutions have still not emerged to solve these problems, so we want to integrate the solutions to these types of problems by creating a bundle of applications to make our life a little bit simpler.
- One of the many common functions in all smart phones is the GPS. Our wish is to use GPS to aid in the process of solving problems.

- Assume you are a first year student at IUT and want to make your way to the campus from your house. Most people are unfamiliar with the location of IUT so when they are traveling from their home to IUT they need some sort of friend that will alert them that are getting closer to their destination. This is where our application comes into use.
- Sometimes we get sleepy during the journey, and end up going past our destination. This type of simple problem can irritate us a lot and we may face problems because of it.
- We also want to use SMS processing to reduce the use of manual systems of mobile balance transfer (Flexi load, I-top up, etc...).
- Service providers need to keep track of the different transactions made by using the old pen and paper method.
- We wish to process the SMS which will be received by the service provider and take all the information like the transaction amount, destination number, and the remaining balance for that particular distributor and put it into a database so that he doesn't need to calculate or remember his balance.

Problem Identification

From the current scenario we see that in our daily life we are facing some problems. The problems are given below:

- Reaching your destination without any hassles
- Riding without continuously looking at your phone
- Manually noting the transactions details.
- The use of multiple phones for different transactions
- Making a mistake in the balance calculations.
- Slow process of final outcome of transactions.
- Hybrid use of pen and paper with phone as opposed to just phone.
- Passing away the destination.
- Reach the destination in time.

Goals

- Our primary goal is to increase the interaction between humans and computers.
- Making human daily life easy
- Digitalizing of manual information
- Making a journey to an unknown place less stressful.
- Reaching the exact location in a self-sufficient manner
- Making the process of balance transfer time efficient
- Removing the pen and paper system.
- Making the smart phone device our friend.

Application Overview

Daily Life Assistant is an android mobile app which may solve your daily life problems which you face every day. This project has two major parts. One is Destination Alert and other is Mobile transection. First part will help you to reach your destination safely and other part is for automation of mobile transection system.

6.1 Destination alert

Once you open the app, you will see a menu which will ask you whether you want to enter the Location alert app or the Mobile Transaction app.

When you click on the Location alert, you will enter the Location alert application. The first thing you will see is a map with your current location marked on it using a Google Maps style format.

The little blue dot will indicate your current location in the map.

To set a destination all the user has to do is touch any location on the map.

As you move closer to your destination, the blue dot will move accordingly thanks to the help of GPS.

As soon as you are within a 500 meter radius from your destination with respect to the longitude and latitude coordinates, an alarm will ring notifying you that you are close to your destination.

Every time you select a destination, there will be a small notification message informing you of the distance between your current location and the destination.

6.2 Mobile Transaction

For every Service Provider there are different types of formats. All the formats should be included in a single app. User should give only two inputs, one is the mobile number and another is the amount of money. Show all the transaction in a single table.

Source Code of the Project

The project contains 6 classes:

- AlertDialogManager.java
- ConnectionDetector.java
- GPSTracker.java
- MainActivity.java
- MyService.java
- OpeningActivity.java

7.1 AlertDialogManager.java

```
package com.example.locationalert;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;

public class AlertDialogManager {
    /**
     * Function to display simple Alert Dialog
     * @param context - application context
     * @param title - alert dialog title
     * @param message - alert message
     * @param status - success/failure (used to set icon)
     *               - pass null if you don't want icon
     */
    public void showAlertDialog(Context context, String title, String
message,
                               Boolean status) {
```



```
AlertDialog alertDialog = new AlertDialog.Builder(context).create();
```

```
    // Setting Dialog Title  
    alertDialog.setTitle(title);
```

```
    // Setting Dialog Message  
    alertDialog.setMessage(message);
```

```
    if(status != null)  
        // Setting alert dialog icon  
        alertDialog.setIcon((status) ? R.drawable.success :  
R.drawable.fail);
```

```
    // Setting OK Button  
    alertDialog.setButton("OK", new  
DialogInterface.OnClickListener() {  
        public void onClick(DialogInterface dialog, int which)  
    {  
        }  
    });
```

```
    // Showing Alert Message  
    alertDialog.show();  
    }  
}
```

7.2 ConnectionDetector.java

```
package com.example.locationalert;
import android.content.Context;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;

public class ConnectionDetector {
    private Context _context;
    public ConnectionDetector(Context context){
        this._context = context;
    }
    /**
     * Checking for all possible internet providers
     * **/
    public boolean isConnectingToInternet(){
        ConnectivityManager connectivity = (ConnectivityManager)
        _context.getSystemService(Context.CONNECTIVITY_SERVICE);
        if (connectivity != null)
        {
            NetworkInfo[] info = connectivity.getAllNetworkInfo();
            if (info != null)
                for (int i = 0; i < info.length; i++)
                    if (info[i].getState() == NetworkInfo.State.CONNECTED)
                    {
                        return true;
                    }
        }
        return false;
    }
}
```

7.3 GPSTracker.java

```
package com.example.locationalert;
import android.app.AlertDialog;
import android.app.Service;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.IBinder;
import android.provider.Settings;
import android.util.Log;

public class GPSTracker extends Service implements LocationListener {

    private final Context mContext;

    // flag for GPS status
    boolean isGPSEnabled = false;

    // flag for network status
    boolean isNetworkEnabled = false;

    // flag for GPS status
    boolean canGetLocation = false;

    Location location = null; // location
    double latitude; // latitude
    double longitude; // longitude
```

```

// The minimum distance to change Updates in meters
private static final long MIN_DISTANCE_CHANGE_FOR_UPDATES = 10;
// 10 meters

// The minimum time between updates in milliseconds
private static final long MIN_TIME_BW_UPDATES = 1000 * 60 * 1;
// 1 minute

// Declaring a Location Manager
protected LocationManager locationManager;

public GPSTracker(Context context) {
    this.mContext = context;
    getLocation();
}

public Location getLocation() {
    try {
        locationManager = (LocationManager) mContext
            .getSystemService(LOCATION_SERVICE);

        // getting GPS status
        isGPSEnabled = locationManager

.isProviderEnabled(LocationManager.GPS_PROVIDER);

        // getting network status
        isNetworkEnabled = locationManager

.isProviderEnabled(LocationManager.NETWORK_PROVIDER);

        if (!isGPSEnabled && !isNetworkEnabled) {
            // no network provider is enabled
        } else {
            this.canGetLocation = true;

```

```

        if (isNetworkEnabled) {

locationManager.requestLocationUpdates(

LocationManager.NETWORK_PROVIDER,
                                MIN_TIME_BW_UPDATES,
MIN_DISTANCE_CHANGE_FOR_UPDATES, this);
        Log.d("Network", "Network Enabled");
        if (locationManager != null) {
            location = locationManager

.getLastKnownLocation(LocationManager.NETWORK_PROVIDER)
;
                                if (location != null) {
                                    latitude =
location.getLatitude();
                                    longitude =
location.getLongitude();
                                }
        }
        // if GPS Enabled get lat/long using GPS

```

Services

```

        if (isGPSEnabled) {
            if (location == null) {

locationManager.requestLocationUpdates(

LocationManager.GPS_PROVIDER,

MIN_TIME_BW_UPDATES,

MIN_DISTANCE_CHANGE_FOR_UPDATES, this);

```

```

Log.d("GPS", "GPS Enabled");
if (locationManager != null) {
    locationManager = locationManager;
    locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
    if (location != null) {
        latitude = locationManager.getLatitude();
        longitude = locationManager.getLongitude();
    }
}
}
}
}

} catch (Exception e) {
    e.printStackTrace();
}

return locationManager;
}

/** Stop using GPS listener Calling this function will stop using
GPS in your app ** */
public void stopUsingGPS() {
    if (locationManager != null) {
        locationManager.removeUpdates(GPSTracker.this);
    }
}

/**
 * Function to get latitude
 **/
public double getLatitude() {
    if (location != null) {
        latitude = locationManager.getLatitude();
    }
}

```

```

}

        // return latitude
        return latitude;
    }

/**
 * Function to get longitude
 */
public double getLongitude() {
    if (location != null) {
        longitude = location.getLongitude();
    }

    // return longitude
    return longitude;
}

/**
 * Function to check GPS/wifi enabled
 *
 * @return boolean
 */
public boolean canGetLocation() {
    return this.canGetLocation;
}

/**
 * Function to show settings alert dialog On pressing Settings
button will
 * lauch Settings Options
 */
public void showSettingsAlert() {

```

```

        AlertDialog.Builder alertDialog = new
AlertDialog.Builder(mContext);

        // Setting Dialog Title
        alertDialog.setTitle("GPS is settings");

        // Setting Dialog Message
        alertDialog
                .setMessage("GPS is not enabled. Do you want
to go to settings menu?");

        // On pressing Settings button
        alertDialog.setPositiveButton("Settings",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface
dialog, int which) {

                        Intent intent = new Intent(

Settings.ACTION_LOCATION_SOURCE_SETTINGS);
                        mContext.startActivity(intent);
                    }
                });

        // on pressing cancel button
        alertDialog.setNegativeButton("Cancel",
                new DialogInterface.OnClickListener() {
                    public void onClick(DialogInterface
dialog, int which) {

                        dialog.cancel();
                    }
                });

        // Showing Alert Message
        alertDialog.show();

```



```
    }

    @Override
    public void onLocationChanged(Location location) {
    }

    @Override
    public void onProviderDisabled(String provider) {
    }

    @Override
    public void onProviderEnabled(String provider) {
    }

    @Override
    public void onStatusChanged(String provider, int status, Bundle
extras) {
    }

    @Override
    public IBinder onBind(Intent arg0) {
        return null;
    }
}
```

7.4 MainActivity.java

```
package com.example.locationalert;

import java.io.IOException;
import java.util.List;

import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.GooglePlayServicesUtil;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.gms.maps.model.Polyline;

import android.annotation.SuppressLint;
import android.app.Activity;
import android.app.Dialog;
import android.app.ProgressDialog;
import android.content.Intent;
import android.graphics.Point;
import android.location.Address;
import android.location.Criteria;
import android.location.Geocoder;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.AsyncTask;
import android.os.Bundle;
import android.os.Handler;
import android.support.v4.app.FragmentActivity;
import android.text.Html;
```

```

import android.util.Log;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import android.view.View.OnClickListener;

@SuppressLint("NewApi") public class MainActivity extends
FragmentActivity implements LocationListener {

    GoogleMap mGoogleMap;
    GPSTracker gps;
    ProgressDialog pDialog;
    LatLng origin, locDes, latLng;
    MarkerOptions markerOptions;
    Boolean isInternetPresent = false;
    ConnectionDetector cd;

    Handler eHandler;
    public String errorMessage = "Connection Time out!!! Try
Again.";

    // Alert Dialog Manager
    AlertDialogManager alert = new AlertDialogManager();
    MainActivity mapRouteActivity;
    MarkerOptions options;
    LocationManager locationManager;

    Location location;
    public static Location loc;

```

```

    LatLng destPoint;
    String provider = "";

    Thread distanceChecker;
    Intent intent;

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

        mapRouteActivity = this;
        progressDialog = new ProgressDialog(this);

        progressDialog.setMessage(Html.fromHtml("<b>Load</b><br/>Loading
Map..."));
        progressDialog.setIndeterminate(false);
        progressDialog.setCancelable(false);

        cd = new ConnectionDetector(getApplicationContext());

        // Check if Internet present
        isInternetPresent = cd.isConnectingToInternet();
        if (!isInternetPresent) {
            // Internet Connection is not present
            alert.showAlertDialog(MainActivity.this,
                "Internet Connection Error",
                "Please connect to working Internet
connection", false);
            // stop executing code by return

            return;
        }
    }

```

```

    }

    gps = new GPSTracker(this);

    int status = GooglePlayServicesUtil

.isGooglePlayServicesAvailable(getBaseContext());

    if (status != ConnectionResult.SUCCESS) {

        int requestCode = 10;
        Dialog dialog =
GooglePlayServicesUtil.getErrorDialog(status, this,
            requestCode);
        dialog.show();

    } else {
        // Google Play Services are available
        // Initializing
        // Getting reference to SupportMapFragment of the
activity_main
        SupportMapFragment fm = (SupportMapFragment)
getSupportFragmentManager()
            .findFragmentById(R.id.gmap);

        // Getting Map for the SupportMapFragment
mGoogleMap = fm.getMap();

        // Enable MyLocation Button in the Map
mGoogleMap.setMyLocationEnabled(true);

```

```

        locationManager = (LocationManager)
getSystemService(this.LOCATION_SERVICE);

        Criteria criteria = new Criteria();
        String bbb =
locationManager.getBestProvider(criteria, true);
        Location myLocation =
locationManager.getLastKnownLocation(bbb);
        double lat= myLocation.getLatitude();
        double lng = myLocation.getLongitude();
        LatLng currentLatLng = new LatLng(lat, lng);

        mGoogleMap.moveCamera(CameraUpdateFactory.newLatLngZoom(
currentLatLng, 16));

        // Getting reference to btn_find of the layout
activity_main
        Button btn_find = (Button) findViewById(R.id.btn_find);

        // Defining button click event listener for the find button
        OnClickListener findClickListener = new OnClickListener() {

                @Override
                public void onClick(View v) {
                        // Getting reference to EditText to get
the user input location
                        EditText etLocation = (EditText)
findViewById(R.id.et_location);

                        // Getting user input location
                        String location =
etLocation.getText().toString();

```

```

if(location!=null && !location.equals("")){
    new
    GeocoderTask().execute(location);
    }
};

// Setting button click event listener for the find
button
    btn_find.setOnClickListener(findClickListener);

    mGoogleMap.setOnMapClickListener(new
    GoogleMap.OnMapClickListener() {

        @Override
        public void onMapClick(LatLng point) {
            // TODO Auto-generated method stub
            //lstLatLngs.add(point);
            destPoint = point;
            MarkerOptions marker = new
            MarkerOptions().position(
                new LatLng(point.latitude,
                point.longitude)).title(String.valueOf(point.latitude) + " " +
                String.valueOf(point.longitude));
            mGoogleMap.clear();

            if(intent != null)
                stopService(intent);
            //mGoogleMap.addMarker(new
            MarkerOptions().position(point));
            mGoogleMap.addMarker(marker);
            //showToast(destPoint);

```

```

        intent = new Intent(getBaseContext(),
MyService.class);
        intent.putExtra("LAT", point.latitude);
        intent.putExtra("LONG", point.longitude);
        startService(intent);

    }
});

}

}

public float showToast(LatLng destPoint){

    float[] results = new float[5];
    loc.distanceBetween(gps.getLatitude(), gps.getLongitude(),
destPoint.latitude, destPoint.longitude, results);
    //Toast.makeText(this, String.valueOf(results[0]), 1500);
    Log.e("Distance", String.valueOf(results[0]));
    return results[0];

}

@Override
public void onLocationChanged(Location arg0) {
    // TODO Auto-generated method stub

}

```



```

@Override
public void onProviderDisabled(String arg0) {
    // TODO Auto-generated method stub
}

@Override
public void onProviderEnabled(String arg0) {
    // TODO Auto-generated method stub
}
@Override
public void onStatusChanged(String arg0, int arg1, Bundle arg2) {
    // TODO Auto-generated method stub
}
private class GeocoderTask extends AsyncTask<String, Void,
List<Address>>{
    @Override
    protected List<Address> doInBackground(String...
locationName) {
        // Creating an instance of Geocoder class
        Geocoder geocoder = new
Geocoder(getApplicationContext());
        List<Address> addresses = null;
        try {
            // Getting a maximum of 3 Address that
matches the input text
addresses = geocoder.getFromLocationName(locationName[0], 3);
        } catch (IOException e) {
            e.printStackTrace();
        }
        return addresses;
    }
    @Override
    protected void onPostExecute(List<Address> addresses) {

```

```

        if(addresses==null || addresses.size()==0){
            Toast.makeText(getBaseContext(), "No Location found",
Toast.LENGTH_SHORT).show();
        }
        // Clears all the existing markers on the map
        mGoogleMap.clear();
        // Adding Markers on Google Map for each matching
address
            for(int i=0;i<addresses.size();i++){
                Address address = (Address) addresses.get(i);
                // Creating an instance of GeoPoint, to display in Google Map
                latLng = new LatLng(address.getLatitude(),
address.getLongitude());
                String addressText = String.format("%s, %s",
                address.getMaxAddressLineIndex() > 0 ?
address.getAddressLine(0) : "",
                address.getCountryName());
                markerOptions = new MarkerOptions();
                markerOptions.position(latLng);
                markerOptions.title(addressText);
                mGoogleMap.addMarker(markerOptions);
                // Locate the first location
                if(i==0)
                    mGoogleMap.animateCamera(CameraUpdateFactory.newLatLng
(latLng));
            }
        }
    }
}

```

7.5 MyService.java

```
package com.example.locationalert;

import java.io.IOException;
import java.util.Timer;
import java.util.TimerTask;

import android.app.Service;
import android.content.Intent;
import android.content.res.AssetFileDescriptor;
import android.location.Location;
import android.media.AudioManager;
import android.media.MediaPlayer;
import android.media.MediaPlayer.OnCompletionListener;
import android.os.Bundle;
import android.os.Handler;
import android.os.IBinder;
import android.os.Looper;
import android.os.Vibrator;
import android.widget.Toast;

public class MyService extends Service {

    TimerTask checker;
    final Handler handler = new Handler();
    Timer t = new Timer();
    Double d_lat;
    Double d_long;

    @Override
    public IBinder onBind(Intent arg0) {
        return null;
    }
}
```

```

@Override
public int onStartCommand(Intent intent, int flags, int startId) {
    // Let it continue running until it is stopped.
    d_lat = intent.getExtras().getDouble("LAT");
    d_long = intent.getExtras().getDouble("LONG");

    Toast.makeText(this, "Service Started",
Toast.LENGTH_SHORT).show();

    /*
    * Thread bg = new Thread(new Runnable() {
    *
    * GPSTracker gps = new GPSTracker(getBaseContext());
Location loc;
    * float[] results = new float[3];
    *
    * @Override public void run() {
    *
    * while(true){ Looper.prepare();
    *
    * try {
    *
    * Thread.sleep(5*1000);
loc.distanceBetween(gps.getLatitude(),
    * gps.getLongitude(), d_lat, d_long, results);
    * Toast.makeText(getBaseContext(),
String.valueOf(results[0]) ,
    * Toast.LENGTH_SHORT).show();
    *
    * } catch (InterruptedException e) {
    *
    * e.printStackTrace(); }
    *

```

```

        * Looper.loop(); } } });
        *
        * bg.start();
        */
        checkDistance();

        return START_STICKY;
    }

    public void checkDistance() {

        checker = new TimerTask() {

            GPSTracker gps = new
GPSTracker(getApplicationContext());
            Location loc;
            float[] results = new float[3];

            @Override
            public void run() {
                handler.post(new Runnable() {

                    @Override
                    public void run() {
                        // TODO Auto-generated method stub
                        loc.distanceBetween(gps.getLatitude(),
                        gps.getLongitude(), d_lat, d_long, results);
                        Toast.makeText(getApplicationContext(),
                        String.valueOf(results[0]), Toast.LENGTH_SHORT)
                            .show();
                        if(results[0] <= 500.00){

                            try {
                                playSound();

```

```

onDestroy();
    } catch
(IllegalArgumentException e) {}
        catch (SecurityException e) {}
        catch (IllegalStateException e) {}
        catch (IOException e) {}
    }
}
});

}
};

//t.schedule(checker, 500, 5000);
t.schedule(checker, 1000);

}
@Override
public void onDestroy() {
    super.onDestroy();
    Toast.makeText(this, "Service Destroyed",
Toast.LENGTH_SHORT).show();
}
private void playSound() throws IllegalArgumentException,
SecurityException, IllegalStateException, IOException{

    final Vibrator v = (Vibrator)
getService(getApplicationContext().VIBRATOR_SERVICE);

long[] pattern = {0, 100, 1000, 200, 1000, 300, 1000, 400, 1000, 500,
1000};
final AudioManager mAudioManager = (AudioManager)
getService(AUDIO_SERVICE);

```

```

final int originalVolume =
mAudioManager.getStreamVolume(AudioManager.STREAM_MUSIC);
mAudioManager.setStreamVolume(AudioManager.STREAM_MUSIC,
mAudioManager.getStreamMaxVolume(AudioManager.STREAM_MUSIC), 0);

        MediaPlayer mp = new MediaPlayer();
        mp.setAudioStreamType(AudioManager.STREAM_MUSIC);
        AssetFileDescriptor afd =
getBaseContext().getResources().openRawResourceFd(R.raw.alarm);
        if (afd == null) return;
mp.setDataSource(afd.getFileDescriptor(), afd.getStartOffset(),
afd.getLength());
        afd.close();
        mp.prepare();
        mp.start();
        //v.vibrate(pattern, -1);

        mp.setOnCompletionListener(new OnCompletionListener()
        {
            @Override
            public void onCompletion(MediaPlayer mp)
            {
mAudioManager.setStreamVolume(AudioManager.STREAM_MUSIC,
originalVolume, 0);
                v.cancel();
            }
        });
    }
}

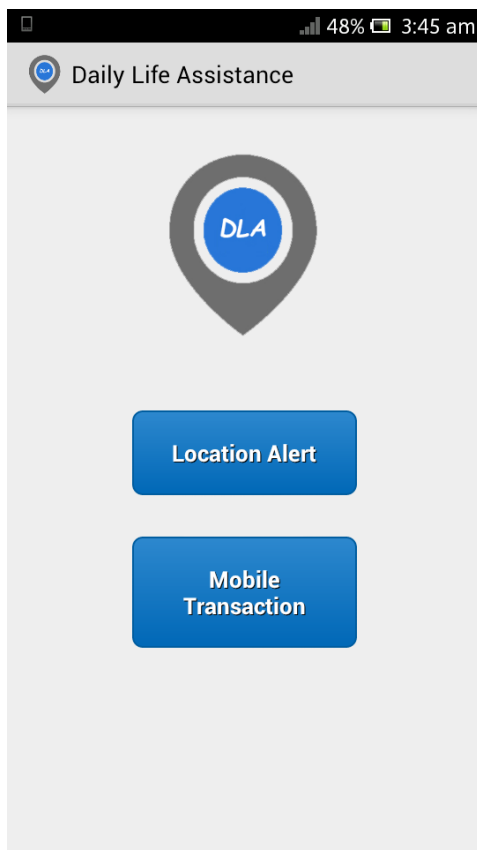
```

7.6 OpeningActivity.java

```
package com.example.locationalert;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.Toast;
public class OpeningActivity extends Activity{@Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_opening);
        Button btnLocationAlert = (Button)findViewById(R.id.btnSetAlarm);
        Button btnLoad = (Button)findViewById(R.id.btnMessage);
        btnLocationAlert.setOnClickListener(new OnClickListener(){
            @Override
            public void onClick(View arg0) {
                Intent i = new Intent(OpeningActivity.this, MainActivity.class);
                startActivity(i);
            }
        });
        btnLoad.setOnClickListener(new OnClickListener() {@Override
            public void onClick(View arg0) {
                Toast.makeText(getApplicationContext(), "Not Available Yet", 2000).show();
            }
        });
    }
}
```


Screenshots of the project

8.1 Project Menu:

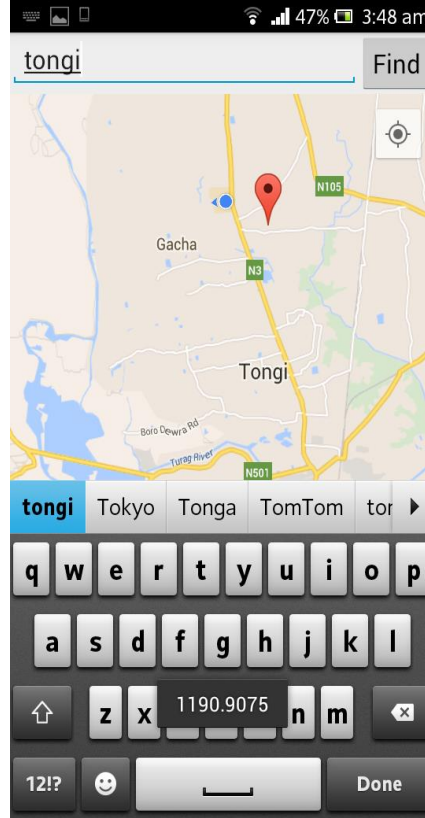
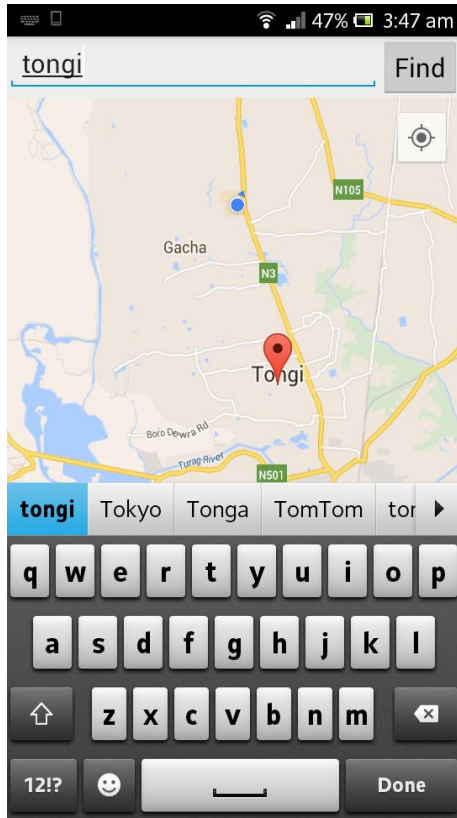


This is the first screen that appears as soon as you enter the Daily Life Assistance app.

You can choose between the Location Alert part and the Mobile transaction part.

Unfortunately we have not yet implemented the Mobile transaction part.

8.2 Daily life assistant:



Once you have entered the Location Alert part, you will see a Google Maps type of interface. From there you can select your destination. Once you have selected your destination, it will be marked with a marker. You can also search for a particular address, which will be automatically selected. As soon as you have selected your destination, a small notification message will appear letting you know of the distance between your current location and your selected location. After an interval of 5 seconds, it will again look for the distance between your current location and the selected location until it has seen that the distance is equal to 500 m.

At this point, an alarm will ring which lets us know that we are close to our destination.

Limitations

Unfortunately, we were not able to implement the Mobile transaction part as we faced some difficulties. The vendors at the credit stalls wanted a unified system which addressed all of their problems under one roof.

The problem with that is that every Service Provider has its own unique way of identifying their transaction codes with their own systems. This complicates things as there is no more sequence to the procedure and makes it difficult to implement in a programming aspect.

Another problem is that in order to transfer balance to a certain number, the Service Provider has its own sim card associated software with its own security policies which we cannot access.

This makes it impossible for us to operate on it at all.

The credit vendors also wanted to send transfer balance from different Service Providers all in one phone. This makes it impossible as the maximum amount of sim cards you can use in the same phone in the Android OS is 2.

Furthermore, there is still no SDK which supports the dual sim function.

Conclusion

The Daily Life Assistant is a bundle of apps which contains solutions to daily dilemmas. One of the main part of Daily Life Assistant is to incorporate apps which should be pre-installed on your current smartphone. The whole concept of the Daily Life Assistant is to keep adding new apps which will make the life of the normal average human being much simpler, regardless of the type of lifestyle that individual has. We wish to improve our work by continuously adding new applications which ultimately benefits the user in a simple and important way.