K.J. Somaiya College of Science & Commerce
Vidyavihar (E), Mumbai – 400077.
Autonomous
Affiliated to University of Mumbai

PROJECT REPORT
ON
Feeder

SUBMITTED BY
Priti Mainkar

TYBSC COMPUTER SCIENCE
UNIVERSITY OF MUMBAI
2019-2020

PROJECT GUIDE
Mrs. Monali Deshpande

K.J. Somaiya College of Science & Commerce

Vidyanagar, Vidyavihar, Mumbai – 400077.

Autonomous – Affiliated to University of Mumbai

DEPARTMENT OF COMPUTER SCIENCE

(2018-2019)

CERTIFICATE

This is to verify that Ms. **Priti Mainkar** of T.Y. B.Sc. [Computer Science Semester-VI] Seat no. **1998- 62**, Has successfully completed the project entitled As **Feeder** During the academic year 2019 – 2020.

Internal Guide Examiner
Date:

Course Coordinator College Seal
External
B.Sc. (CS) Date:
Date:

Acknowledgment

Achievement is finding out what you would be doing rather than what you have to do. It is not until you undertake such a project that you realize how much effort and hard work it really is, what your capabilities are and how well you present yourself or other things. It tells you how much we rely on the efforts and goodwill of others. It gives me immense pleasure to present this report towards the fulfillment of my project.

It has been rightly said that we are built on the shoulders of others. For everything I have achieved, the credit goes to all those who had helped me to complete this project successfully.
I take this opportunity to express my profound gratitude to the management of K. J. Somaiya College of Science and Commerce for giving me this opportunity to accomplish this project work.
A special vote of thanks to my teachers, Mrs. Monali Deshpande and Miss. Bhavna Pandya for their most sincere, useful and encouraging contribution throughout the project span.

Finally, I would like to thank all my friends and the entire Computer Science department who directly or indirectly helped me in the completion of this project and to my family without whose support, motivation and encouragement this would not have been possible.

Miss Priti Mainkar

Index

1. Organizational Overview

Feeder is an NGO management system. It is a platform that provides hands to the needy and homeless. We seek a world of hope, love, tolerance, social justice and peace where poverty has been overcome and everyone lives with dignity and security.

It provides a platform for people who want to volunteer for social causes, as well as the people who are trying to help the society by their donations.

NGOs and Orphanages are always in need of donations in the form of money, clothes, books, food and utilities. What we provide is room for these organizations and donors as well as volunteers to shake hands under a roof.

The end users organizations, volunteers as well as donors.
Here, organizations can get in touch with registered volunteers near their area. Volunteers can register themselves or directly get in touch with the nearest organization by finding it via the map. People can make direct donations by getting in contact with the organizations or make donations through us.
The app also has the feature for organizations to find nearby Restaurants (as restaurants can have leftover food at the end of the day), nearby Schools (to spread awareness campaigns for children so that they could be taught the importance of helping others from a young age) and to find nearby Hospitals (in case of emergencies).

This application is useful as it brings all the like-minded people who work for the betterment of society directly under one roof and makes the donation process easy.

The organizations on the map as well as the volunteers are manually verified and then added, hence saving people from getting fooled by fake organizations.

The project does not require registration or authentication, hence can be used by everyone in a fast and efficient way.

2. Description of Existing System

The present system is not very effective. It is completely manual and time consuming.

The volunteers have to personally find organizations and go to the location where they have to contact the representatives.

To make donations, they cannot be done online quick and fast. Donors have to get in contact with organizations and then can make donations in the form of a check or cash which requires the donors to spend a lot of time for finding the right organization and then going to its location to make the donation.

If organizations require volunteers, they have to create campaigns to spread awareness or search for people manually who might be interested in helping around.

Many a times volunteers cannot find organizations even though they might be living quite close to one, or organizations maybe unable to find nearby volunteers and people wishing to donate do not know the process or do not have the time to go through the process of doing it.

Lack of a platform where everyone can find one another is the major problem in the present system.

3. Limitations of Present System

1. Time consumption

Donors, Volunteers and Organizations have to manually get in contact with each other for donations which is time consuming.

2. Lack of a platform

Many a times volunteers cannot find organizations even though they might be living quite close to one, or organizations are unable to find nearby volunteers and people wishing to donate do not know the process or do not have the time to go through the process of doing it.

3. Hectic

It is not possible for full time working people to go to organizations personally to get to know the process and make donations as they don't get time out of their busy schedules.

4. Lack of Verification

Sometimes people are fooled into making donations for fake organizations.

To avoid all these limitations and to make the system more accurate and efficient, the system needs to be computerized.

4. Proposed System and Its Advantages

To reduce the inconveniences that are found in the present system, the proposed system has been automated so as to produce a user-friendly interface.

The proposed system will allow users to locate the exact location of the organisations. The organisations will be able to view the nearby volunteers and be able to get in touch with them for help as required. They will not have to manually find helpers through campaigns. The users will be able to donate money from the comfort of their homes and will know where their money was donated.

The proposed system will have the following features-

1. Helps donors to locate the nearby orphanages and ngos.
2. Helps organizations to locate nearby restaurants, schools and hospitals.
3. Helps organizations find volunteers.
4. Helps donors to donate without having to search for an organization.
5. Allows people to register for volunteering.

Advantages of proposed system-

1. Application is easily accessible by users and saves time.
2. It eliminates the time required for the donation process.
3. Users can view exact location of organizations.
4. Eliminates the time required to find verified organisations and volunteers.
5. Ensures data accuracy.
6. User friendly and robust system.
7. Saves time and efforts of processing.
8. Verification of organisations is already done manually, hence, no chance of getting fooled by fake organisations.

5. Stakeholders

Stakeholders refer to “ an individual, group or organization, who may affect, be affected by, or perceive itself affected by a decision, activity or outcome of a project. ”

Following are the stakeholders of this system -

1. Organisations (NGOs and Orphanages)
   Organisations are the ones whose location will be displayed to the users along with the contact information, timings and address details. They can find volunteers through the app.

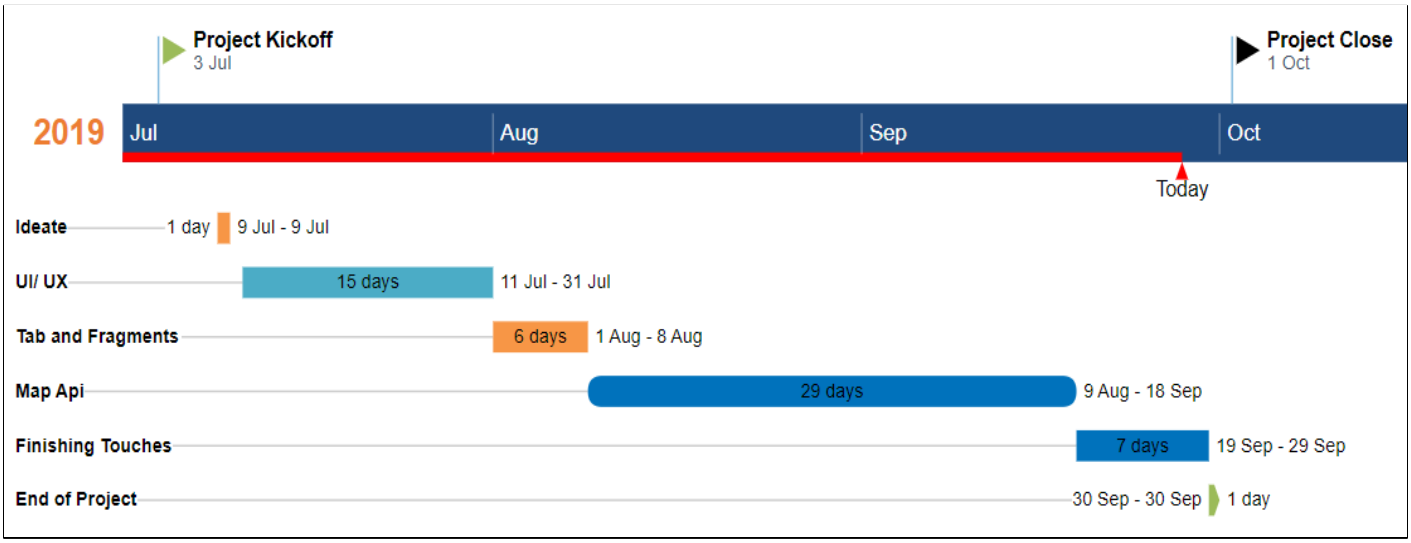
2. Donors
   These are the ones who will make donations to organisations through the app or by directly contacting them.

3. Volunteers
   These are the ones who can register to be helpers for organisations or find verified organisations through the app.

6. Gantt Chart

A Gantt chart is a horizontal bar chart developed as a production control tool in 1917 by Henry L. Gantt, an American engineer and social scientist. Frequently used in project management, a Gantt chart provides a graphical illustration of a schedule that helps to plan, coordinate, and track specific tasks in a project.

This chart lists the timeline of development of project from July to October.
The horizontal axis has the timeline whereas the vertical axis has the tasks.



7. Technologies Used and their Description

Technologies include Hardware and Software requirements.

1. **Software Requirements-**

Windows 10
Android Studio (Latest Version)

Android Studio
It is Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. Android Studio was announced on May 16, 2013 at the Google I/O conference. It was in early access preview stage starting from version 0.1 in May 2013.The current stable version is 3.1 released in March 2018.

The following features are provided in the current stable version:

- Gradle-based build support
- Android-specific refactoring and quick fixes
- Lint tools to catch performance, usability, version compatibility and other problems
- ProGuard integration and app-signing capabilities
- Template-based wizards to create common Android designs and components
- A rich layout editor that allows users to drag-and-drop UI components, option to preview layouts on multiple screen configurations
- Support for building Android Wear apps
- Built-in support for Google Cloud Platform, enabling integration with Firebase Cloud Messaging (Earlier 'Google Cloud Messaging') and Google App Engine
- Android Virtual Device (Emulator) to run and debug apps in the Android studio.

2. **Hardware Requirements**

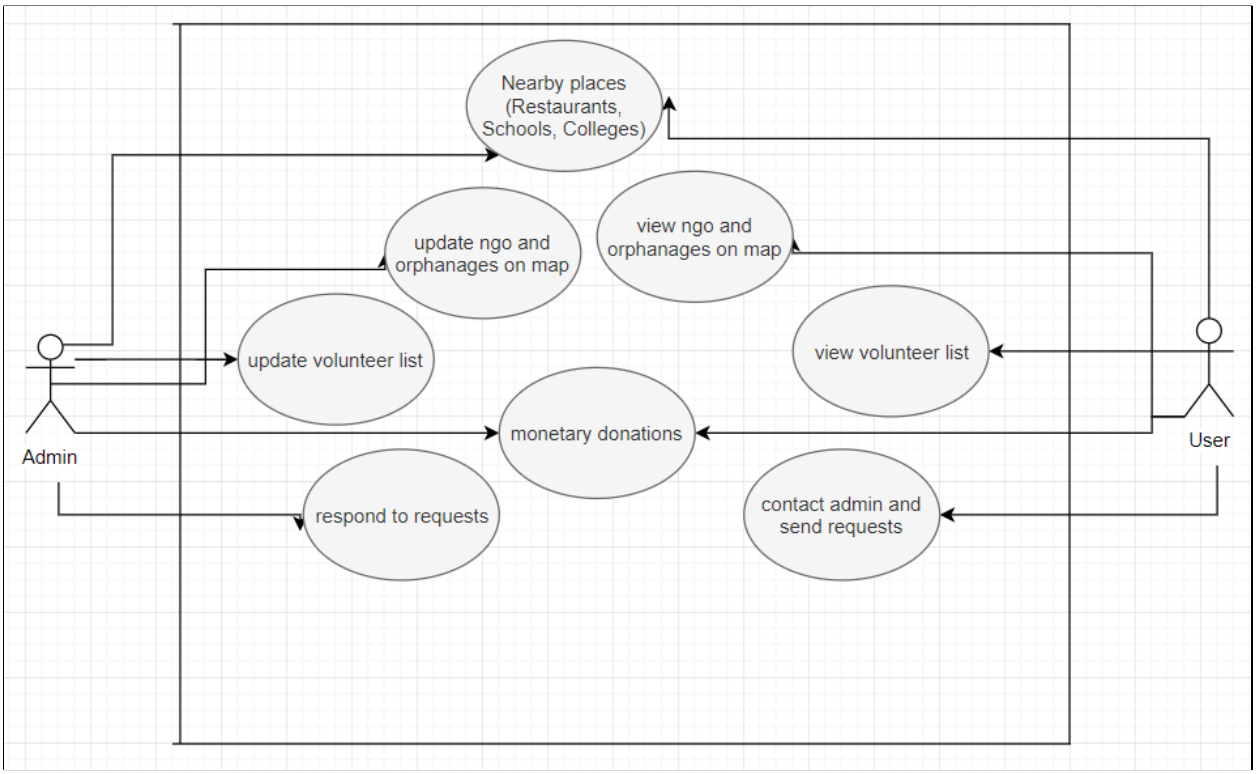
Processor - i3 core
Memory - 4GB RAM
Android Phone with Lollipop and above

8. Event Table

It is a table that lists events in row and key pieces of information about each event in columns.

Sr.no	Event	Trigger	Source	Activity	Response	Destination
1.	Donor donates money	Donate fragment	Donor	Sends Money	Money sent succesfully email.	Bank Account
2.	Search nearby organisations	NGO fragment	Donor, Volunteer	Click on "find here"	gets location of nearby organisations	Google Map
3.	Select organisation and find way	Current location	Donor, Volunteer	Click on organisation marker	Get way to reach address	Google Map
4.	find volunteers	volunteer fragment	Organisations	Swipe right to find nearby volunteers	get names and contact details of nearby volunteers	volunteer list
5.	Contact Admin	register as volunteer or organisation	volunteer, organisations	Click on "Contact us"	after verification, get represented on the app	feeder database
6.	find nearby places	through google nearbyplaces api	all users	click on "nearby places"	get nearby schools, hospitals and restaurants	Google Map

9. Use-Case Diagram and Basic Scenarios and Use-Case Description

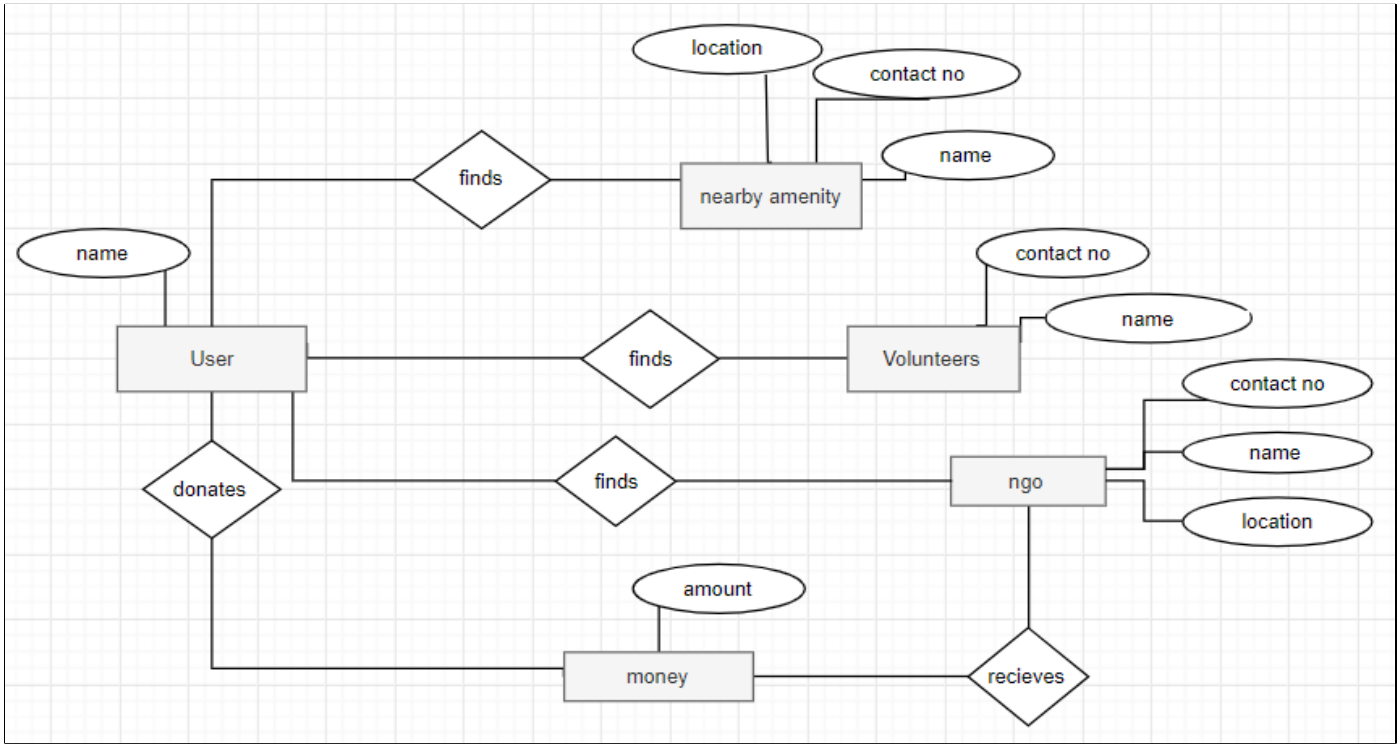


Here, the system can be accessed by the Admin and the user.

Admin Scenarios -
The admin can access the system through the source code. Admin can update the nearby places using api, can add new markers manually for NGOs and orphanages. Admin can receive requests
from users and then add their details onto the application interface (in list of volunteers or in the nearby NGO and orphanages map) manually. The admin can also accept monetary donations from the user and send the donations to nearby NGOs.

User Scenarios -
The users are of three types (organisations, donors and volunteers). All users can access the list of volunteers as well as the details of nearby NGOs and orphanages. All users can also access information about nearby places (restaurants, schools and hospitals) if required. All users can contact the admin with requests (for registration, or any other). Users can also make monetary donations to the organisations via the admin.

10. Entity-Relationship Diagram



An ERD shows the relationships of entity sets stored in a database.
It consists of entities, attributes and relationships.

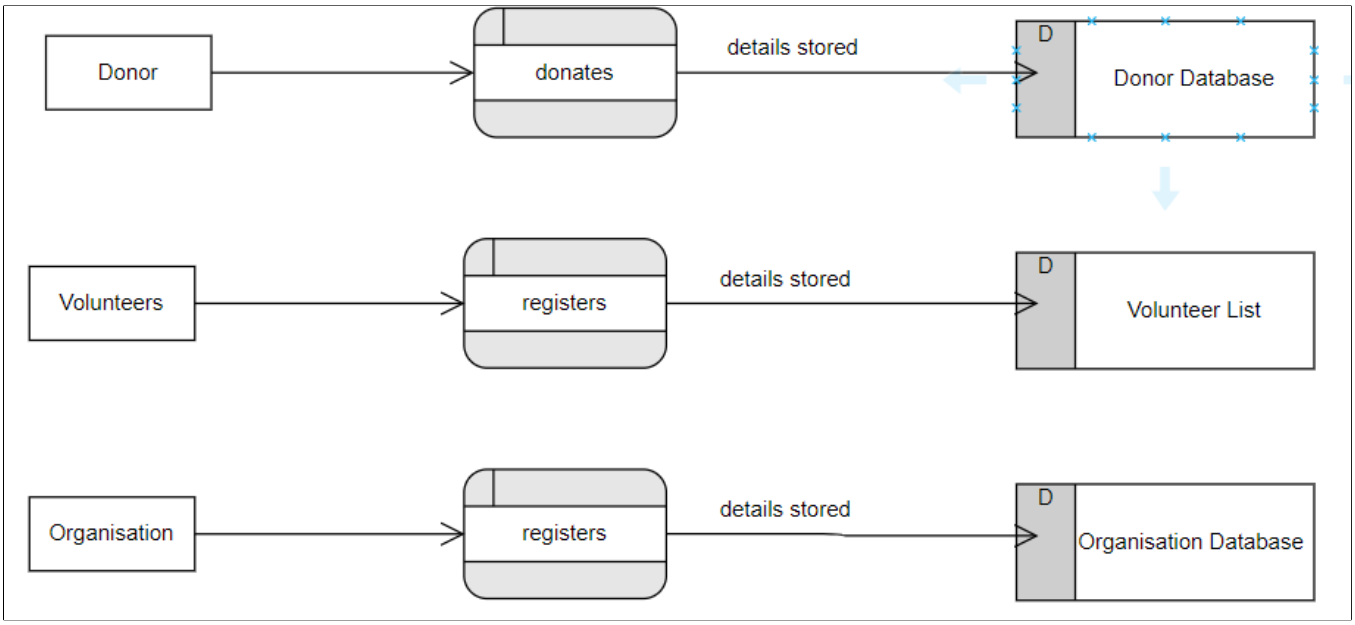
Entities are represented by rectangles. It is the object or concept about which you want to store information.

Attributes are represented by ovals. It is used to store the characteristics of the entity.

Relationships are represented by a diamond. It shows how entities share information in the database.

11. Data Flow Diagram

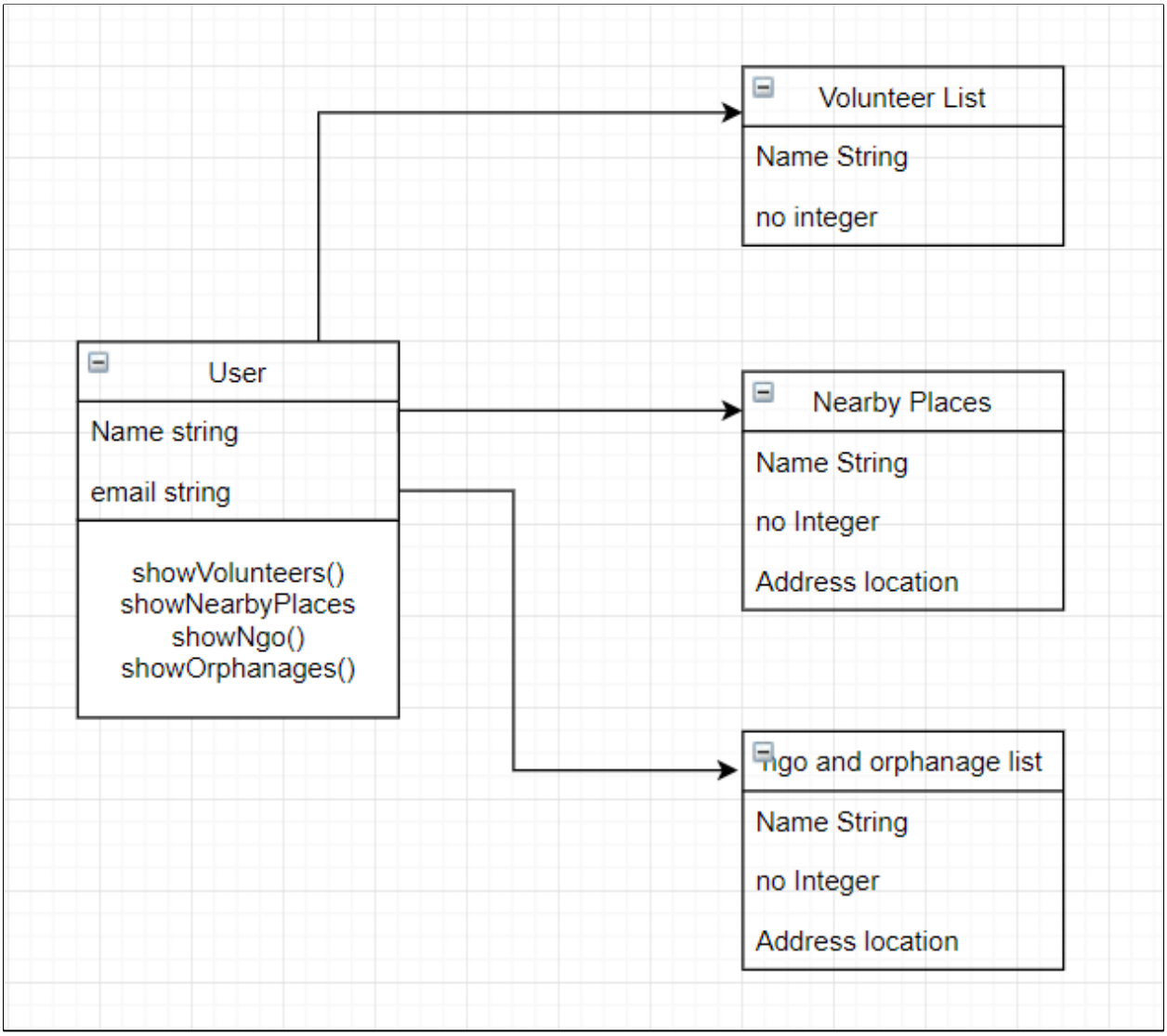
A DFD illustrates how data is processed by a system in terms of inputs and outputs. As its name indicates, its focus is on the flow of information, where data comes from, where it goes and how it is stored.



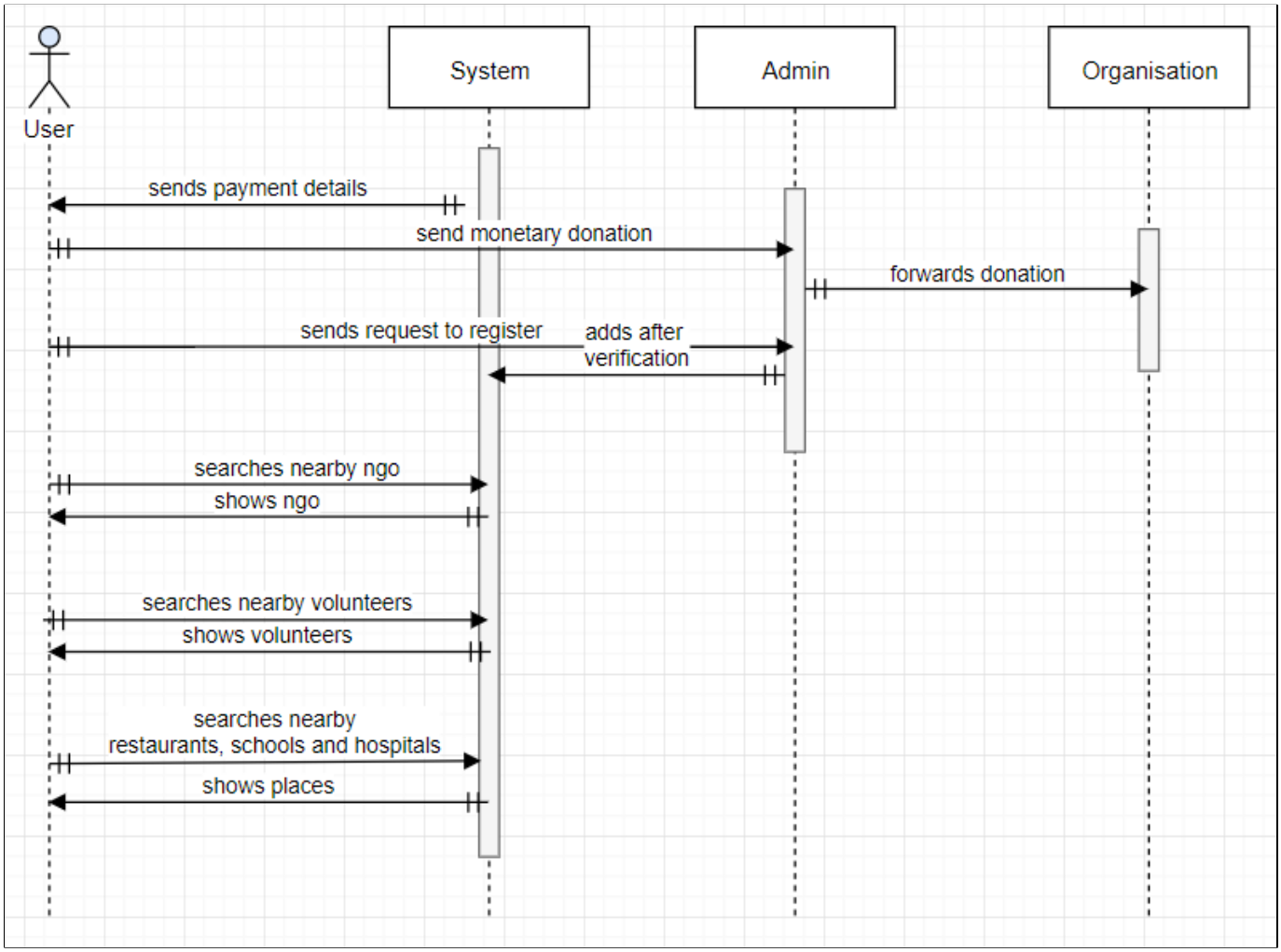
Here, when the donor donates money, his email address, name and other details are stored in database.
When volunteer registers, his name, contact details and city are stored in volunteer list.
When organisation registers, their name, contact details and exact location is stored in database.

12. Class Diagram

Class diagram is the illustration of relationships and source code dependencies among classes in UML.
Here, the user class can access the classes for list of volunteers, nearby places(like restaurants, schools and hospitals) and nearby NGOs and orphanages by passing functions.
Here the users can be organisations, donors as well as volunteers.



13. Sequence Diagram

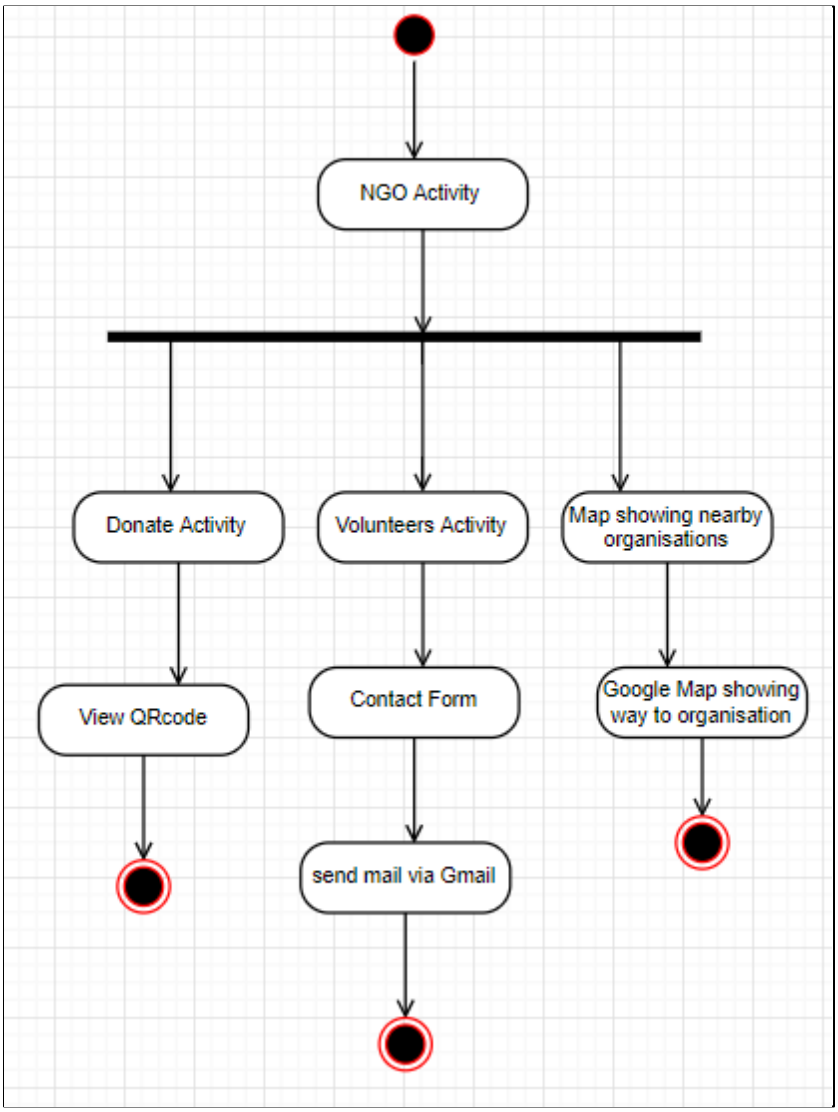


The above diagram shows the sequence of processes (internal as well as external) that take place when -

1. User makes a monetary donation
2. User searches for nearby ngo and orphanages.
3. User searches for nearby amenities.
4. User searches for nearby volunteers.
5. New registration takes place (Volunteers as well as Organisations).

Here, the user can be a volunteer, donor or organisation.

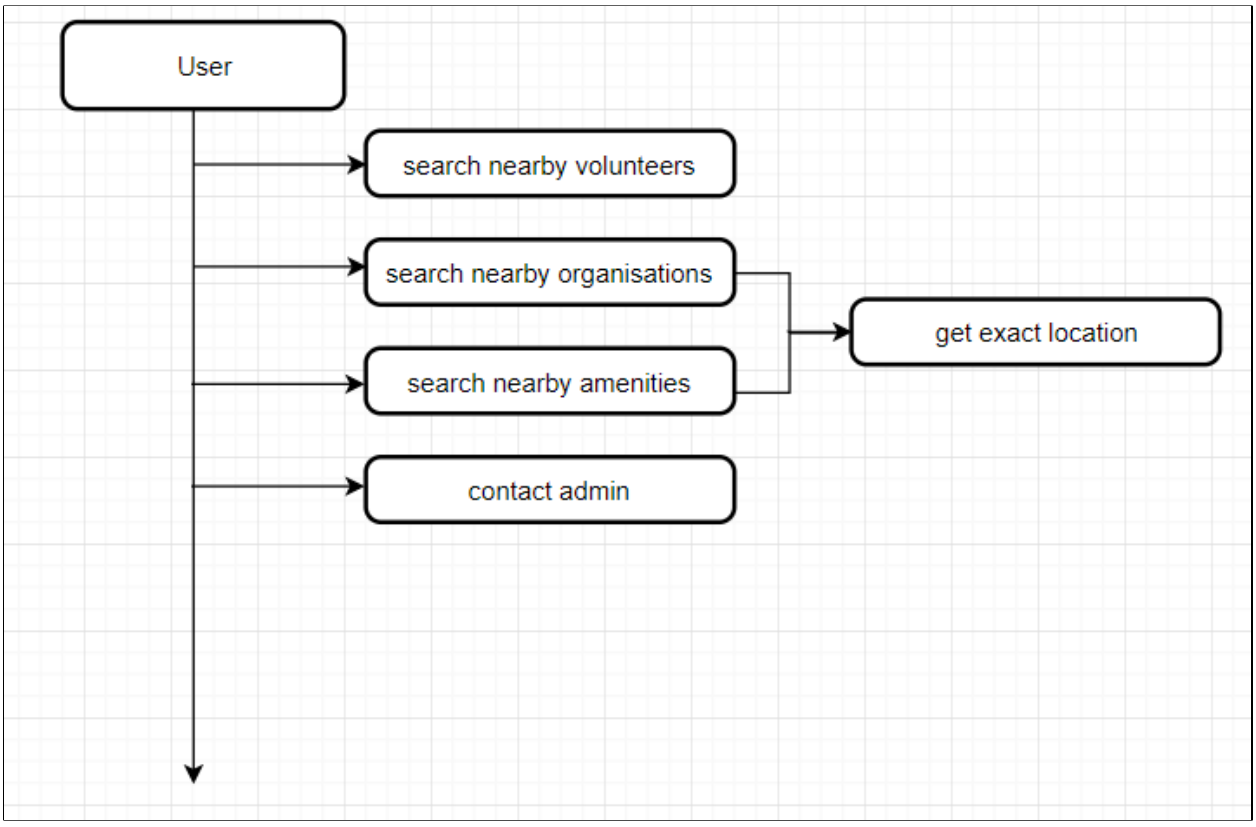
14. State Diagram



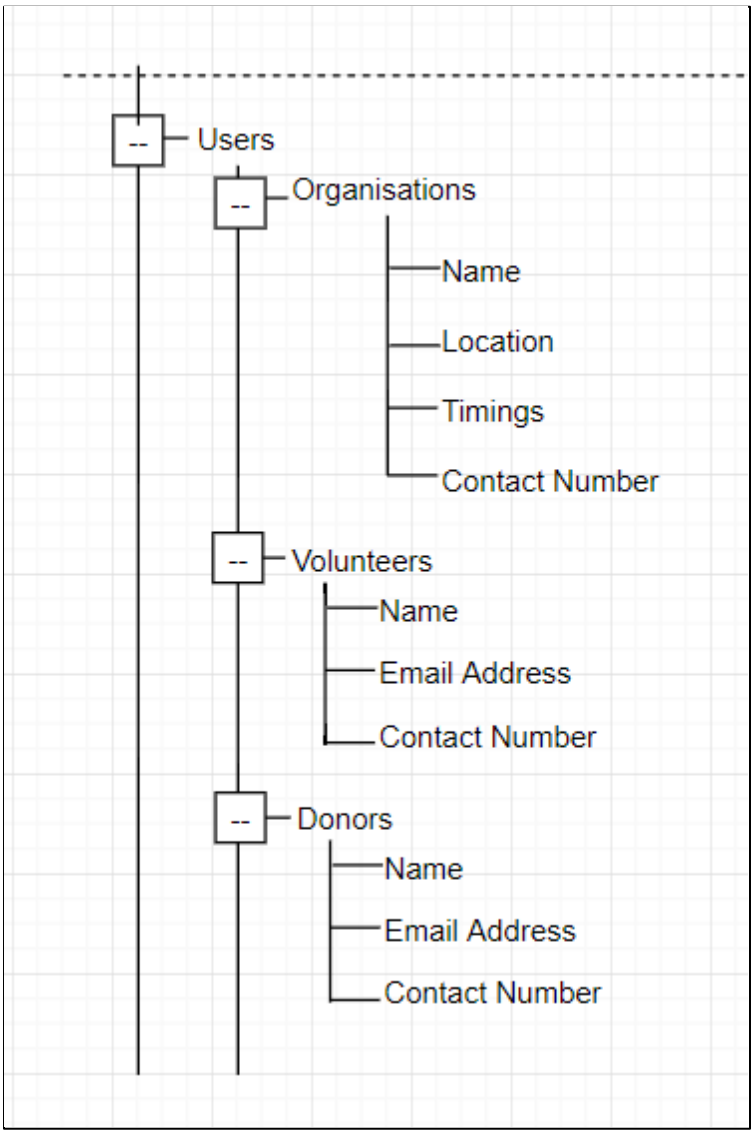
A state diagram is used to represent the condition of the system or part of the system at finite instances of time. It's a behavioral diagram and it represents the behavior using finite state transitions.

15. Menu Tree

A tree diagram is a chart that begins with one central item and then branches into more and then keeps branching until line of enquiry beginning with central item is exhausted.



16. List of Tables with Attributes and Constraints



17. System Coding

content_main.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"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:background="#FFFFFF"
  app:layout_behavior="@string/appbar_scrolling_view_behavior"
  tools:context=".MainActivity"
  tools:showIn="@layout/app_bar_main">

  <android.support.design.widget.TabLayout
    android:id="@+id/tab"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#189789"
    app:tabIndicatorColor="#FF5722"
    app:tabTextColor="#FFFFFF"></android.support.design.widget.TabLayout>

  <android.support.v4.view.ViewPager
    android:id="@+id/pager"
    android:layout_width="match_parent"
    android:layout_height="622dp"
    android:layout_below="@+id/tab"
    android:layout_marginTop="1dp" />
</RelativeLayout>
```

nav_header_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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="@dimen/nav_header_height"
  android:background="@drawable/side_nav_bar"
  android:gravity="bottom"
  android:orientation="vertical"
  android:paddingLeft="@dimen/activity_horizontal_margin"
  android:paddingTop="@dimen/activity_vertical_margin"
  android:paddingRight="@dimen/activity_horizontal_margin"
  android:paddingBottom="@dimen/activity_vertical_margin"
  android:theme="@style/ThemeOverlay.AppCompat.Dark">

  <ImageView
    android:id="@+id/imageView"
```

```
    android:layout_width="91dp"
    android:layout_height="111dp"
    android:contentDescription="@string/nav_header_desc"
    android:paddingTop="@dimen/nav_header_vertical_spacing"
    app:srcCompat="@mipmap/ic_launcher" />

  <TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

</LinearLayout>
```

app_bar_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
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=".MainActivity">

  <android.support.design.widget.AppBarLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:theme="@style/AppTheme.AppBarOverlay">

    <android.support.v7.widget.Toolbar
      android:id="@+id/toolbar"
      android:layout_width="match_parent"
      android:layout_height="?attr/actionBarSize"
      android:background="?attr/colorPrimary"
      app:popupTheme="@style/AppTheme.PopupOverlay" />

  </android.support.design.widget.AppBarLayout>

  <include layout="@layout/content_main" />
</android.support.design.widget.CoordinatorLayout>
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v4.widget.DrawerLayout
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:id="@+id/drawer_layout"
  android:layout_width="match_parent"
```

```
  android:layout_height="match_parent"
  android:fitsSystemWindows="true"
  tools:openDrawer="start">

  <include
    layout="@layout/app_bar_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />


  <android.support.design.widget.NavigationView
    android:id="@+id/nav_view"
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_gravity="start"
    android:fitsSystemWindows="true"
    app:headerLayout="@layout/nav_header_main"
    app:menu="@menu/activity_main_drawer" />
</android.support.v4.widget.DrawerLayout>
```

activity_main_drawer.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  tools:showIn="navigation_view">

  <group android:checkableBehavior="single">

    <item
      android:id="@+id/nav_hungerspot"
      android:icon="@drawable/ic_menu_gallery"
      android:title="Nearby Places" />
  </group>

  <item android:title="Communicate">
    <menu>
      <item
        android:id="@+id/nav_contactus"
        android:icon="@drawable/ic_menu_share"
        android:title="Contact Us" />
    </menu>
  </item>

</menu>
```

MainActivity.java

```
package ml.pritimainkar.afinal;
```

```
import android.content.Intent;
import android.os.Bundle;
import android.support.design.widget.NavigationView;
import android.support.design.widget.TabLayout;
import android.support.v4.view.GravityCompat;
import android.support.v4.view.ViewPager;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuItem;

public class MainActivity extends AppCompatActivity
    implements NavigationView.OnNavigationItemSelectedListener {
  TabLayout tabLayout;
  ViewPager viewPager;
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Toolbar toolbar = findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);


    //TABLAYOUT
    tabLayout=(TabLayout)findViewById(R.id.tab);
    viewPager=(ViewPager)findViewById(R.id.pager);

    tabLayout.addTab(tabLayout.newTab().setText("Donate"));
    tabLayout.addTab(tabLayout.newTab().setText("Ngo"));
    tabLayout.addTab(tabLayout.newTab().setText("Volunteers"));
    tabLayout.setTabGravity(TabLayout.GRAVITY_FILL);

    MyAdapter adapter = new MyAdapter(this,getSupportFragmentManager(), tabLayout.getTabCount());
    viewPager.setAdapter(adapter);

    viewPager.addOnPageChangeListener(new
TabLayout.TabLayoutOnPageChangeListener(tabLayout));

    tabLayout.addOnTabSelectedListener(new TabLayout.OnTabSelectedListener() {
      @Override
      public void onTabSelected(TabLayout.Tab tab) {
        viewPager.setCurrentItem(tab.getPosition());
      }

      @Override
      public void onTabUnselected(TabLayout.Tab tab) {
```

```
        }

        @Override
        public void onTabReselected(TabLayout.Tab tab) {

        }
    });

    //end of tablayout
    DrawerLayout drawer = findViewById(R.id.drawer_layout);
    NavigationView navigationView = findViewById(R.id.nav_view);
    ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(
            this, drawer, toolbar, R.string.navigation_drawer_open, R.string.navigation_drawer_close);
    drawer.addDrawerListener(toggle);
    toggle.syncState();
    navigationView.setNavigationItemSelectedListener(this);
}

@Override
public void onBackPressed() {
    DrawerLayout drawer = findViewById(R.id.drawer_layout);
    if (drawer.isDrawerOpen(GravityCompat.START)) {
        drawer.closeDrawer(GravityCompat.START);
    } else {
        super.onBackPressed();
    }
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {

    int id = item.getItemId();
    return super.onOptionsItemSelected(item);
}

@SuppressWarnings("StatementWithEmptyBody")
@Override
public boolean onNavigationItemSelected(MenuItem item) {
    // Handle navigation view item clicks here.
    int id = item.getItemId();
```

```
    if (id == R.id.nav_hungerspot) {
        Intent intent= new Intent(this,MapActivity.class);
        startActivity(intent);

    } else if (id == R.id.nav_contactus) {
        Intent intent= new Intent(this, contactus.class);
        startActivity(intent);

    }

    DrawerLayout drawer = findViewById(R.id.drawer_layout);
    drawer.closeDrawer(GravityCompat.START);
    return true;
  }
}
```

MyAdapter.java

```
package ml.pritimainkar.afinal;

import android.content.Context;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentPagerAdapter;
import android.support.v4.app.FragmentManager;

public class MyAdapter extends FragmentPagerAdapter {

  private Context myContext;
  int totalTabs;

  public MyAdapter(Context context, FragmentManager fm, int totalTabs) {
    super(fm);
    myContext = context;
    this.totalTabs = totalTabs;
  }

  // this is for fragment tabs
  @Override
  public Fragment getItem(int position) {
    switch (position) {
      case 0:
        Donate donate = new Donate();
        return donate;
      case 1:
        Ngo ngo = new Ngo();
        return ngo;
      case 2:
        Volunteers volunteers = new Volunteers();
```

```
            return volunteers;
        default:
            return null;
    }
  }
  // this counts total number of tabs
  @Override
  public int getCount() {
    return totalTabs;
  }
}
```

---

fragment_donate.xml

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

  <!-- TODO: Update blank fragment layout -->
    <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:orientation="vertical">

      <Space
        android:layout_width="match_parent"
        android:layout_height="35dp"
         />

      <ImageView
        android:id="@+id/imageView2"
        android:layout_width="311dp"
        android:layout_marginLeft="33dp"
        android:layout_height="172dp"
        android:src="@drawable/donate" />

      <Space
        android:layout_width="match_parent"
        android:layout_height="11dp" />


      <TextView
        android:id="@+id/textView17"
        android:layout_width="373dp"
        android:layout_height="47dp"
```

```
            android:layout_marginLeft="33dp"
            android:text="Email- feeder@gmail.com"
            android:textColor="#070707"
            android:textSize="24sp"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/textView16"
            android:layout_width="310dp"
            android:layout_height="67dp"
            android:layout_alignParentStart="true"
            android:layout_alignParentLeft="true"
            android:layout_alignParentBottom="true"
            android:textAlignment="center"
            android:layout_marginStart="31dp"
            android:layout_marginLeft="31dp"
            android:layout_marginTop="10dp"
            android:text="Send money to the details provided after you agree to the TnC. We will do the donations on your behalf and email you the receipt."
            android:textColor="#FD050505"
            android:textSize="14sp" />

        <Space
            android:layout_width="match_parent"
            android:layout_height="19dp" />

        <CheckBox
            android:id="@+id/agreed"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="I have read and agreed to the given terms and conditions."
            android:textColor="#141313" />

        <TextView
            android:id="@+id/textView15"
            android:layout_width="321dp"
            android:layout_height="102dp"
            android:layout_alignParentStart="true"
            android:layout_alignParentLeft="true"
            android:layout_alignParentBottom="true"
            android:layout_marginStart="31dp"
            android:layout_marginLeft="31dp"
            android:layout_marginTop="10dp"
            android:text="* TERMS AND CONDITIONS - \n Send a confirmation email of the money sent to the above email and we will be in touch with you. The donations has to be in Indian currency only and Rs. 1000 and above. All donations below Rs, 1000 will be considered invalid and no refund will be generated for them. Please know that this is not an NGO but simply  an initiative taken by like-minded individuals. *
"
```

```
            android:textColor="#2A3F3C"
            android:textSize="10sp"
            android:textStyle="italic" />

        <Space
            android:layout_width="match_parent"
            android:layout_height="111dp" />

    </LinearLayout>


</RelativeLayout>
```

Donate.java

```
package ml.pritimainkar.afinal;

import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.CheckBox;
public class Donate extends Fragment implements View.OnClickListener{

  View view;
  Context context;
  CheckBox agreed;

  public Donate() {
    // Required empty public constructor
  }

  @Override
  public View onCreateView(LayoutInflater inflater, ViewGroup container,
                           Bundle savedInstanceState) {
    view= inflater.inflate(R.layout.fragment_donate, container, false);
    context= view.getContext();
    agreed = (CheckBox)view.findViewById(R.id.agreed);
    agreed.setOnClickListener(this);
    return  view;

  }

  private void showQR() {
    Intent intent= new Intent(context, qrCode.class);
    startActivity(intent);
```

```
    }

    @Override
    public void onClick(View v) {
        if (agreed.isChecked() == true) {
            showQR();
        }


    }
}
```

---

fragment_ngo.xml

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

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


        <Space
            android:layout_width="match_parent"
            android:layout_height="35dp" />

        <ImageView
            android:id="@+id/imageView3"
            android:layout_width="311dp"
            android:layout_marginLeft="33dp"
            android:layout_height="172dp"
            android:src="@drawable/ngo" />

        <Space
            android:layout_width="match_parent"
            android:layout_height="11dp" />


        <TextView
            android:id="@+id/textView3"
            android:layout_width="311dp"
```

```
        android:layout_height="99dp"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="20dp"
        android:layout_marginLeft="10dp"
        android:layout_marginTop="10dp"
        android:text="Giving is not just about making a donation, it is about making a difference. If you have surplus clothes, books or food, give them to those who don't have enough. Find the NGO's and orphanages near you here!"
        android:textAlignment="center"
        android:textColor="#080909"
        android:textSize="14sp" />

    <Button
        android:id="@+id/nearby_ngo"
        android:layout_width="151dp"
        android:layout_height="32dp"
        android:layout_marginLeft="100dp"
        android:background="#009688"
        android:text="Find Here!"
        android:textColor="#FCF9F9" />


    <Space
        android:layout_width="match_parent"
        android:layout_height="66dp" />


  </LinearLayout>
</ScrollView>
```

Ngo.java

```
package ml.pritimainkar.afinal;


import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;

public class Ngo extends Fragment implements View.OnClickListener {
  Button ngo;
  View view;
```

```
    Context context;
    public Ngo() {
        // Required empty public constructor

    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {

        view= inflater.inflate(R.layout.fragment_ngo, container, false);
        context= view.getContext();
        ngo = (Button)view.findViewById(R.id.nearby_ngo);
        ngo.setOnClickListener(this);
        return  view;

    }
    @Override
    public void onClick(View v) {
        Intent intent= new Intent(context, map_activity.class);
        startActivity(intent);

    }
}
```

---

fragment_volunteers.xml

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

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

        <Space
            android:layout_width="match_parent"
            android:layout_height="35dp" />


        <ImageView
            android:id="@+id/imageView4"
            android:layout_width="311dp"
            android:layout_marginLeft="33dp"
```

```
    android:layout_height="172dp"
    android:src="@drawable/volunteer" />

<Space
    android:layout_width="match_parent"
    android:layout_height="11dp" />


<TextView
    android:id="@+id/textView3"
    android:layout_width="310dp"
    android:layout_height="41dp"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentBottom="true"
    android:layout_marginStart="35dp"
    android:layout_marginLeft="35dp"
    android:layout_marginTop="10dp"
    android:text="Fill out the below form to become a volunteer and we will get back to you! "
    android:textColor="#080909"
    android:textSize="12sp" />

<Button
    android:id="@+id/be_one"
    android:layout_width="157dp"
    android:layout_height="34dp"
    android:layout_marginLeft="100dp"
    android:background="#009688"
    android:text="Form"
    android:textColor="#FFFFFF" />

<Space
    android:layout_width="match_parent"
    android:layout_height="18dp" />


<TextView
    android:id="@+id/textView2"
    android:layout_width="308dp"
    android:layout_height="35dp"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentBottom="true"
    android:layout_marginStart="35dp"
    android:layout_marginLeft="35dp"
    android:layout_marginTop="10dp"
    android:text="NEARBY VOLUNTEERS-"
    android:textColor="#080909"
```

```
        android:textSize="14sp" />

    <ListView
        android:id="@+id/volunteerList"
        android:layout_width="match_parent"
        android:layout_height="375dp"
        android:layout_alignParentTop="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:layout_marginTop="0dp"

        android:textColor="#F8504747"
        app:layout_constraintEnd_toEndOf="parent"
        tools:layout_editor_absoluteY="234dp">

    </ListView>


    <Space
        android:layout_width="match_parent"
        android:layout_height="24dp" />



  </LinearLayout>

</ScrollView>
```

Volunteers.java

```
package ml.pritimainkar.afinal;

import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ListView;

import java.util.ArrayList;

public class Volunteers extends Fragment implements View.OnClickListener {
  View view;
  Context context;
  Button beone;
```

```
    private ListView mListView;

    public Volunteers() {
        // Required empty public constructor
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        view= inflater.inflate(R.layout.fragment_volunteers, container, false);
        context= view.getContext();
        beone= (Button)view.findViewById(R.id.be_one);
        beone.setOnClickListener(this);
        mListView = (ListView)view.findViewById(R.id.volunteerList);
        ArrayList<String> listData = new ArrayList<>();
        listData.clear();
        listData.add(0, "There are no volunteers currently in your area. Be One!");
        ArrayAdapter arrayAdapter = new ArrayAdapter(context,android.R.layout.simple_list_item_1,listData);
        mListView.setAdapter(arrayAdapter);
        return  view;
    }

    @Override
    public void onClick(View v) {
        Intent intent= new Intent(context, contactus.class);
        startActivity(intent);

    }
}
```

---

activity_map.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"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MapActivity">

    <fragment
        android:id="@+id/map"
        android:name="com.google.android.gms.maps.SupportMapFragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

    <Button
        android:id="@+id/btnRestaurant"
        android:layout_width="130dp"
```

```
    android:layout_height="30dp"
    android:layout_alignParentEnd="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="222dp"
    android:layout_marginRight="222dp"
    android:layout_marginBottom="567dp"
    android:background="#FF5722"
    android:text="Restaurant"
    android:textColor="#FCF6F6" />

  <Button
    android:id="@+id/btnHospital"
    android:layout_width="104dp"
    android:layout_height="30dp"
    android:layout_alignParentEnd="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="193dp"
    android:layout_marginRight="193dp"
    android:layout_marginBottom="45dp"
    android:background="#FF5722"
    android:text="Hospital"
    android:textColor="#F7FFFFFF" />

  <Button
    android:id="@+id/btnSchool"
    android:layout_width="104dp"
    android:layout_height="30dp"
    android:layout_alignParentEnd="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="66dp"
    android:layout_marginRight="66dp"
    android:layout_marginBottom="44dp"
    android:background="#FF5722"
    android:text="School"
    android:textColor="#FCFFFFFF" />


</RelativeLayout>
```

MapActivity.java

```
package ml.pritimainkar.afinal;


import android.Manifest;
import android.content.pm.PackageManager;
```

```
import android.location.Location;
import android.os.Build;
import android.os.Bundle;
import android.support.v4.app.ActivityCompat;
import android.support.v4.app.FragmentActivity;
import android.support.v4.content.ContextCompat;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.GoogleApiAvailability;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;

public class MapActivity extends FragmentActivity implements OnMapReadyCallback,
    GoogleApiClient.ConnectionCallbacks,
    GoogleApiClient.OnConnectionFailedListener,
    LocationListener {

  private GoogleMap mMap;
  double latitude;
  double longitude;
  private int PROXIMITY_RADIUS = 1000;
  GoogleApiClient mGoogleApiClient;
  Location mLastLocation;
  Marker mCurrLocationMarker;
  LocationRequest mLocationRequest;

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

    if (android.os.Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
      checkLocationPermission();
    }
```

```
    //Check if Google Play Services Available or not
    if (!CheckGooglePlayServices()) {
        Log.d("onCreate", "Finishing test case since Google Play Services are not available");
        finish();
    }
    else {
        Log.d("onCreate","Google Play Services available.");
    }

    // Obtain the SupportMapFragment and get notified when the map is ready to be used.
    SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
            .findFragmentById(R.id.map);
    mapFragment.getMapAsync(this);
}

private boolean CheckGooglePlayServices() {
    GoogleApiAvailability googleAPI = GoogleApiAvailability.getInstance();
    int result = googleAPI.isGooglePlayServicesAvailable(this);
    if(result != ConnectionResult.SUCCESS) {
        if(googleAPI.isUserResolvableError(result)) {
            googleAPI.getErrorDialog(this, result,
                    0).show();
        }
        return false;
    }
    return true;
}

@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    mMap.setMapType(GoogleMap.MAP_TYPE_HYBRID);

    //Initialize Google Play Services
    if (android.os.Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
        if (ContextCompat.checkSelfPermission(this,
                Manifest.permission.ACCESS_FINE_LOCATION)
                == PackageManager.PERMISSION_GRANTED) {
            buildGoogleApiClient();
            mMap.setMyLocationEnabled(true);
        }
    }
    else {
        buildGoogleApiClient();
        mMap.setMyLocationEnabled(true);
    }
```

```
Button btnRestaurant = (Button) findViewById(R.id.btnRestaurant);
btnRestaurant.setOnClickListener(new View.OnClickListener() {
    String Restaurant = "restaurant";
    @Override
    public void onClick(View v) {
        Log.d("onClick", "Button is Clicked");
        mMap.clear();
        String url = getUrl(latitude, longitude, Restaurant);
        Object[] DataTransfer = new Object[2];
        DataTransfer[0] = mMap;
        DataTransfer[1] = url;
        Log.d("onClick", url);
        GetNearbyPlacesData getNearbyPlacesData = new GetNearbyPlacesData();
        getNearbyPlacesData.execute(DataTransfer);
        Toast.makeText(MapActivity.this,"Nearby Restaurants", Toast.LENGTH_LONG).show();
    }
});

Button btnHospital = (Button) findViewById(R.id.btnHospital);
btnHospital.setOnClickListener(new View.OnClickListener() {
    String Hospital = "hospital";
    @Override
    public void onClick(View v) {
        Log.d("onClick", "Button is Clicked");
        mMap.clear();
        String url = getUrl(latitude, longitude, Hospital);
        Object[] DataTransfer = new Object[2];
        DataTransfer[0] = mMap;
        DataTransfer[1] = url;
        Log.d("onClick", url);
        GetNearbyPlacesData getNearbyPlacesData = new GetNearbyPlacesData();
        getNearbyPlacesData.execute(DataTransfer);
        Toast.makeText(MapActivity.this,"Nearby Hospitals", Toast.LENGTH_LONG).show();
    }
});

Button btnSchool = (Button) findViewById(R.id.btnSchool);
btnSchool.setOnClickListener(new View.OnClickListener() {
    String School = "school";
    @Override
    public void onClick(View v) {
        Log.d("onClick", "Button is Clicked");
        mMap.clear();
        if (mCurrLocationMarker != null) {
            mCurrLocationMarker.remove();
        }
        String url = getUrl(latitude, longitude, School);
        Object[] DataTransfer = new Object[2];
```

```
            DataTransfer[0] = mMap;
            DataTransfer[1] = url;
            Log.d("onClick", url);
            GetNearbyPlacesData getNearbyPlacesData = new GetNearbyPlacesData();
            getNearbyPlacesData.execute(DataTransfer);
            Toast.makeText(MapActivity.this,"Nearby Schools", Toast.LENGTH_LONG).show();
        }
    });


}

protected synchronized void buildGoogleApiClient() {
    mGoogleApiClient = new GoogleApiClient.Builder(this)
            .addConnectionCallbacks(this)
            .addOnConnectionFailedListener(this)
            .addApi(LocationServices.API)
            .build();
    mGoogleApiClient.connect();
}

@Override
public void onConnected(Bundle bundle) {
    mLocationRequest = new LocationRequest();
    mLocationRequest.setInterval(1000);
    mLocationRequest.setFastestInterval(1000);
    mLocationRequest.setPriority(LocationRequest.PRIORITY_BALANCED_POWER_ACCURACY);
    if (ContextCompat.checkSelfPermission(this,
            Manifest.permission.ACCESS_FINE_LOCATION)
            == PackageManager.PERMISSION_GRANTED) {
        LocationServices.FusedLocationApi.requestLocationUpdates(mGoogleApiClient,
mLocationRequest, this);
    }
}

private String getUrl(double latitude, double longitude, String nearbyPlace) {

    StringBuilder googlePlacesUrl = new
StringBuilder("https://maps.googleapis.com/maps/api/place/nearbysearch/json?");
    googlePlacesUrl.append("location=" + latitude + "," + longitude);
    googlePlacesUrl.append("&radius=" + PROXIMITY_RADIUS);
    googlePlacesUrl.append("&type=" + nearbyPlace);
    googlePlacesUrl.append("&sensor=true");
    googlePlacesUrl.append("&key=" + "AIzaSyATuUiZUkEc_UgHuqsBJa1oqaODI-3mLs0");
    Log.d("getUrl", googlePlacesUrl.toString());
    return (googlePlacesUrl.toString());
}
```

```
    @Override
    public void onConnectionSuspended(int i) {

    }

    @Override
    public void onLocationChanged(Location location) {
        Log.d("onLocationChanged", "entered");

        mLastLocation = location;
        if (mCurrLocationMarker != null) {
            mCurrLocationMarker.remove();
        }
        //Place current location marker
        latitude = location.getLatitude();
        longitude = location.getLongitude();
        LatLng latLng = new LatLng(location.getLatitude(), location.getLongitude());
        MarkerOptions markerOptions = new MarkerOptions();
        markerOptions.position(latLng);
        markerOptions.title("Current Position");

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_YELLOW));
        mCurrLocationMarker = mMap.addMarker(markerOptions);

        //move map camera
        mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
        mMap.animateCamera(CameraUpdateFactory.zoomTo(15));
        Toast.makeText(MapActivity.this,"Your Current Location", Toast.LENGTH_LONG).show();

        Log.d("onLocationChanged", String.format("latitude:%.3f longitude:%.3f",latitude,longitude));

        //stop location updates
        if (mGoogleApiClient != null) {
            LocationServices.FusedLocationApi.removeLocationUpdates(mGoogleApiClient, this);
            Log.d("onLocationChanged", "Removing Location Updates");
        }
        Log.d("onLocationChanged", "Exit");

    }

    @Override
    public void onConnectionFailed(ConnectionResult connectionResult) {

    }
    public static final int MY_PERMISSIONS_REQUEST_LOCATION = 99;
    public boolean checkLocationPermission(){
        if (ContextCompat.checkSelfPermission(this,
                Manifest.permission.ACCESS_FINE_LOCATION)
```

```
        != PackageManager.PERMISSION_GRANTED) {

      if (ActivityCompat.shouldShowRequestPermissionRationale(this,
          Manifest.permission.ACCESS_FINE_LOCATION)) {
        ActivityCompat.requestPermissions(this,
            new String[]{Manifest.permission.ACCESS_FINE_LOCATION},
            MY_PERMISSIONS_REQUEST_LOCATION);


      } else {

        ActivityCompat.requestPermissions(this,
            new String[]{Manifest.permission.ACCESS_FINE_LOCATION},
            MY_PERMISSIONS_REQUEST_LOCATION);
      }
      return false;
    } else {
      return true;
    }
}

@Override
public void onRequestPermissionsResult(int requestCode,
                                       String permissions[], int[] grantResults) {
    switch (requestCode) {
      case MY_PERMISSIONS_REQUEST_LOCATION: {
        if (grantResults.length > 0
            && grantResults[0] == PackageManager.PERMISSION_GRANTED) {

          if (ContextCompat.checkSelfPermission(this,
              Manifest.permission.ACCESS_FINE_LOCATION)
              == PackageManager.PERMISSION_GRANTED) {

            if (mGoogleApiClient == null) {
              buildGoogleApiClient();
            }
            mMap.setMyLocationEnabled(true);
          }

        } else {
          Toast.makeText(this, "permission denied", Toast.LENGTH_LONG).show();
        }
        return;
      }


    }
}
```

```
}
```

GetNearbyPlacesData.java

```
package ml.pritimainkar.afinal;

import android.os.AsyncTask;
import android.util.Log;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;

import java.util.HashMap;
import java.util.List;


public class GetNearbyPlacesData extends AsyncTask<Object, String, String> {

  String googlePlacesData;
  GoogleMap mMap;
  String url;

  @Override
  protected String doInBackground(Object... params) {
    try {
      Log.d("GetNearbyPlacesData", "doInBackground entered");
      mMap = (GoogleMap) params[0];
      url = (String) params[1];
      DownloadUrl downloadUrl = new DownloadUrl();
      googlePlacesData = downloadUrl.readUrl(url);
      Log.d("GooglePlacesReadTask", "doInBackground Exit");
    } catch (Exception e) {
      Log.d("GooglePlacesReadTask", e.toString());
    }
    return googlePlacesData;
  }

  @Override
  protected void onPostExecute(String result) {
    Log.d("GooglePlacesReadTask", "onPostExecute Entered");
    List<HashMap<String, String>> nearbyPlacesList = null;
    DataParser dataParser = new DataParser();
    nearbyPlacesList =  dataParser.parse(result);
    ShowNearbyPlaces(nearbyPlacesList);
    Log.d("GooglePlacesReadTask", "onPostExecute Exit");
  }
```

```
  private void ShowNearbyPlaces(List<HashMap<String, String>> nearbyPlacesList) {
    for (int i = 0; i < nearbyPlacesList.size(); i++) {
      Log.d("onPostExecute","Entered into showing locations");
      MarkerOptions markerOptions = new MarkerOptions();
      HashMap<String, String> googlePlace = nearbyPlacesList.get(i);
      double lat = Double.parseDouble(googlePlace.get("lat"));
      double lng = Double.parseDouble(googlePlace.get("lng"));
      String placeName = googlePlace.get("place_name");
      String vicinity = googlePlace.get("vicinity");
      LatLng latLng = new LatLng(lat, lng);
      markerOptions.position(latLng);
      markerOptions.title(placeName + " : " + vicinity);
      mMap.addMarker(markerOptions);
      markerOptions.icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_RED));
      //move map camera
      mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
      mMap.animateCamera(CameraUpdateFactory.zoomTo(11));
    }
  }
}
```

DownloadUrl.java

```
package ml.pritimainkar.afinal;
import android.util.Log;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;


public class DownloadUrl {

  public String readUrl(String strUrl) throws IOException {
    String data = "";
    InputStream iStream = null;
    HttpURLConnection urlConnection = null;
    try {
      URL url = new URL(strUrl);
      urlConnection = (HttpURLConnection) url.openConnection();
      urlConnection.connect();
      iStream = urlConnection.getInputStream();

      BufferedReader br = new BufferedReader(new InputStreamReader(iStream));

      StringBuffer sb = new StringBuffer();
```

```
            String line = "";
            while ((line = br.readLine()) != null) {
                sb.append(line);
            }

            data = sb.toString();
            Log.d("downloadUrl", data.toString());
            br.close();

        } catch (Exception e) {
            Log.d("Exception", e.toString());
        } finally {
            iStream.close();
            urlConnection.disconnect();
        }
        return data;
    }
}
```

DataParser.java

```
package ml.pritimainkar.afinal;
import android.util.Log;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;


public class DataParser {
    public List<HashMap<String, String>> parse(String jsonData) {
        JSONArray jsonArray = null;
        JSONObject jsonObject;

        try {
            Log.d("Places", "parse");
            jsonObject = new JSONObject((String) jsonData);
            jsonArray = jsonObject.getJSONArray("results");
        } catch (JSONException e) {
            Log.d("Places", "parse error");
            e.printStackTrace();
        }
        return getPlaces(jsonArray);
    }
```

```
  private List<HashMap<String, String>> getPlaces(JSONArray jsonArray) {
    int placesCount = jsonArray.length();
    List<HashMap<String, String>> placesList = new ArrayList<>();
    HashMap<String, String> placeMap = null;
    Log.d("Places", "getPlaces");

    for (int i = 0; i < placesCount; i++) {
      try {
        placeMap = getPlace((JSONObject) jsonArray.get(i));
        placesList.add(placeMap);
        Log.d("Places", "Adding places");

      } catch (JSONException e) {
        Log.d("Places", "Error in Adding places");
        e.printStackTrace();
      }
    }
    return placesList;
  }

  private HashMap<String, String> getPlace(JSONObject googlePlaceJson) {
    HashMap<String, String> googlePlaceMap = new HashMap<String, String>();
    String placeName = "-NA-";
    String vicinity = "-NA-";
    String latitude = "";
    String longitude = "";
    String reference = "";

    Log.d("getPlace", "Entered");

    try {
      if (!googlePlaceJson.isNull("name")) {
        placeName = googlePlaceJson.getString("name");
      }
      if (!googlePlaceJson.isNull("vicinity")) {
        vicinity = googlePlaceJson.getString("vicinity");
      }
      latitude = googlePlaceJson.getJSONObject("geometry").getJSONObject("location").getString("lat");
      longitude                                                                                  =
googlePlaceJson.getJSONObject("geometry").getJSONObject("location").getString("lng");
      reference = googlePlaceJson.getString("reference");
      googlePlaceMap.put("place_name", placeName);
      googlePlaceMap.put("vicinity", vicinity);
      googlePlaceMap.put("lat", latitude);
      googlePlaceMap.put("lng", longitude);
      googlePlaceMap.put("reference", reference);
      Log.d("getPlace", "Putting Places");
    } catch (JSONException e) {
```

```
            Log.d("getPlace", "Error");
            e.printStackTrace();
        }
        return googlePlaceMap;
    }
}
```

---

activity_map_activity.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"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".map_activity">

    <fragment
        android:id="@+id/google_map"
        android:name="com.google.android.gms.maps.SupportMapFragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

    <ImageView
        android:id="@+id/imageView5"
        android:layout_width="138dp"
        android:layout_height="75dp"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="179dp"
        android:layout_marginRight="179dp"
        android:layout_marginBottom="18dp"
        android:src="@drawable/topography" />

    <ImageView
        android:id="@+id/imageView7"
        android:layout_width="72dp"
        android:layout_height="55dp"
        android:layout_alignTop="@+id/imageView7"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentBottom="true"
        android:layout_marginTop="0dp"
        android:layout_marginEnd="14dp"
        android:layout_marginRight="14dp"
        android:layout_marginBottom="28dp"
        android:background="#009688"
        android:src="@drawable/ic_menu_send" />
```

```
</RelativeLayout>
```

map_activity.java

```
package ml.pritimainkar.afinal;

import android.Manifest;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.Bundle;
import android.support.v4.app.ActivityCompat;
import android.support.v4.app.FragmentActivity;
import android.widget.Toast;

import com.google.android.gms.location.FusedLocationProviderClient;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
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.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;

public class map_activity extends FragmentActivity implements GoogleMap.OnMapLoadedCallback,
OnMapReadyCallback {

  Location current;
  FusedLocationProviderClient fusedLocationProviderClient;
  private static final int REQUEST_CODE = 101;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_map_activity);
    fusedLocationProviderClient = LocationServices.getFusedLocationProviderClient(this);
    fetchLastLocation();
  }

  private void fetchLastLocation() {
    if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
      ActivityCompat.requestPermissions(this, new String[]
```

```
            {Manifest.permission.ACCESS_FINE_LOCATION}, REQUEST_CODE);
        return;
    }
    Task<Location> task = fusedLocationProviderClient.getLastLocation();
    task.addOnSuccessListener(new OnSuccessListener<Location>() {
        @Override
        public void onSuccess(Location location) {
            if (location != null) {
                current = location;
                Toast.makeText(getApplicationContext(), current.getLatitude() + "" + current.getLongitude(),
Toast.LENGTH_SHORT).show();
                SupportMapFragment supportMapFragment = (SupportMapFragment)
                        getSupportFragmentManager().findFragmentById(R.id.google_map);
                supportMapFragment.getMapAsync(map_activity.this);


            }
        }

    });
  }

    @Override
    public void onMapLoaded () {

    }

    @Override
    public void onMapReady (GoogleMap googleMap){
        LatLng latLng = new LatLng(current.getLatitude(), current.getLongitude());
        MarkerOptions markerOptions = new MarkerOptions().position(latLng)
                .title("you are here");

markerOptions.icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_GREEN));
        googleMap.animateCamera(CameraUpdateFactory.newLatLng(latLng));
        googleMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLng, 12));
        googleMap.addMarker(markerOptions);


        // Markers NGO//
        Marker m1= googleMap.addMarker( new MarkerOptions()
                .position(new LatLng(18.975997, 72.813136))
                .anchor(0.5f, 0.5f)
                .title("Annamrita")
                .snippet("no-022 23531530, Time- 9.30-18.30(sundayclosed) ")
                .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_ORANGE)));

        Marker m2= googleMap.addMarker( new MarkerOptions()
```

```
        .position(new LatLng(19.105048, 72.841602))
        .anchor(0.5f, 0.5f)
        .title("Akshaya Patra Foundation")
        .snippet("no-09594951155, Time- 9.30-18.30(saturday, sunday closed) ")
        .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_ORANGE)));

    Marker m3= googleMap.addMarker( new MarkerOptions()
        .position(new LatLng(19.122328, 72.917804))
        .anchor(0.5f, 0.5f)
        .title("Akshaya Patra Foundation")
        .snippet("no-09967800390, Time- 10.00-19.00 ")
        .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_ORANGE)));

    Marker m4= googleMap.addMarker( new MarkerOptions()
        .position(new LatLng(19.184799, 72.958548))
        .anchor(0.5f, 0.5f)
        .title(" Shree Bhairavnath Charitable Trust")
        .snippet("no-08652489871, Time- 24 hours) ")
        .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_ORANGE)));

    Marker m5= googleMap.addMarker( new MarkerOptions()
        .position(new LatLng(19.228528, 72.966948))
        .anchor(0.5f, 0.5f)
        .title(" Akshaya Patra Foundation")
        .snippet("no-09967800390, Time-3.30-18.30(sunday closed) ")
        .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_ORANGE)));

    Marker m6= googleMap.addMarker( new MarkerOptions()
        .position(new LatLng(19.186952, 72.980014))
        .anchor(0.5f, 0.5f)
        .title("Asara Trust")
        .snippet("no-09820176644, Time- unknown ")
        .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_ORANGE)));

    Marker m7= googleMap.addMarker( new MarkerOptions()
        .position(new LatLng(19.008861, 73.117064))
        .anchor(0.5f, 0.5f)
        .title("Kamal Arnav Charitable Trust")
        .snippet("no-09870567450, Time- unknown ")
        .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_ORANGE)));

    Marker m8= googleMap.addMarker( new MarkerOptions()
        .position(new LatLng(19.244885, 73.123563))
        .anchor(0.5f, 0.5f)
        .title("Suadha NGO")
        .snippet("no- 02516517678, Time-unknown")
        .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_ORANGE)));
```

```
// end of markers NGO//

//markers orphanages
Marker o1= googleMap.addMarker( new MarkerOptions()
      .position(new LatLng(19.063412, 73.007488))
      .anchor(0.5f, 0.5f)
      .title("Vatsalya Trust")
      .snippet(" no-  022 25782958, Time- 11.00-17.00")
      .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_RED)));

Marker o2= googleMap.addMarker( new MarkerOptions()
      .position(new LatLng(19.091933, 73.009918))
      .anchor(0.5f, 0.5f)
      .title("Desire Society ")
      .snippet("no- 09505117777, Time- 24 hours")
      .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_RED)));

Marker o3= googleMap.addMarker( new MarkerOptions()
      .position(new LatLng(19.094962, 73.010889))
      .anchor(0.5f, 0.5f)
      .title(" Jan Vikas Society  ")
      .snippet("no- 08291505789, TIme- 24 hours")
      .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_RED)));

Marker o4= googleMap.addMarker( new MarkerOptions()
      .position(new LatLng(19.220642, 72.961758))
      .anchor(0.5f, 0.5f)
      .title("Maa Niketan(girls)")
      .snippet("no- 022 21730876, Time- 6.00-22.00")
      .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_RED)));

Marker o5= googleMap.addMarker( new MarkerOptions()
      .position(new LatLng(19.213060, 73.107155))
      .anchor(0.5f, 0.5f)
      .title("Astitva (deaf) ")
      .snippet("no- 0251 2471358")
      .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_RED)));

Marker o6= googleMap.addMarker( new MarkerOptions()
      .position(new LatLng(19.215348, 73.101190))
      .anchor(0.5f, 0.5f)
      .title(" Janani Ashish Charitable Trust ")
      .snippet("no- 02512455879, Time- 7.00-21.00")
      .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_RED)));

Marker o7= googleMap.addMarker( new MarkerOptions()
      .position(new LatLng(19.212606, 73.174827))
      .anchor(0.5f, 0.5f)
```

```
            .title("Nila bal sadan ")
            .snippet("no- 025126046764")
            .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_RED)));
        //end of markers orphanages


    }


}
```

---

activity_qr_code.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=".qrCode">

  <ImageView
    android:id="@+id/imageView6"
    android:layout_width="364dp"
    android:layout_height="426dp"
    app:layout_constraintTop_toTopOf="parent"
    app:srcCompat="@drawable/qrcode"
    tools:layout_editor_absoluteX="0dp" />

  <TextView
    android:id="@+id/textView5"
    android:layout_width="333dp"
    android:layout_height="73dp"
    android:layout_marginStart="16dp"
    android:layout_marginLeft="16dp"
    android:layout_marginTop="408dp"
    android:text="Make transactions by scanning the above QR code \n OR \n At - feeder@upi"
    android:textAlignment="center"
    android:textColor="#3F2222"
    android:textSize="14sp"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
</android.support.constraint.ConstraintLayout>
```

---

activity_contactus.xml

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

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


   <Space
      android:layout_width="match_parent"
      android:layout_height="43dp" />

   <TextView
      android:id="@+id/textView20"
      android:layout_width="wrap_content"
      android:layout_height="30dp"
      android:layout_alignParentStart="true"
      android:layout_alignParentLeft="true"
      android:layout_alignParentTop="true"
      android:layout_marginStart="21dp"
      android:layout_marginLeft="21dp"
      android:text="After clicking submit, use gmail or similar app to finish action."
      android:textColor="#742810"
      android:textSize="12sp"
      android:textStyle="italic" />

   <EditText
      android:id="@+id/full_name"
      android:layout_width="match_parent"
      android:layout_height="63dp"
      android:hint="Your Full Name"
      android:textSize="12sp" />

   <EditText
      android:id="@+id/number"
      android:layout_width="match_parent"
      android:layout_height="59dp"
      android:hint="Your Contact Number"
      android:inputType="number"
      android:textSize="12sp" />

   <EditText
      android:id="@+id/ngo_name"
      android:layout_width="match_parent"
```

```
        android:layout_height="92dp"
        android:hint="NGO name or Full name of the person you are helping"
        android:textSize="12sp" />

    <EditText
        android:id="@+id/ngo_number"
        android:layout_width="match_parent"
        android:layout_height="78dp"
        android:hint="Contact number of NGO or the person who you are helping"
        android:inputType="number"
        android:textSize="12sp" />

    <EditText
        android:id="@+id/ngo_email"
        android:layout_width="match_parent"
        android:layout_height="63dp"
        android:hint="NGO email address"
        android:inputType="textEmailAddress"
        android:textSize="12sp" />

    <Space
        android:layout_width="match_parent"
        android:layout_height="44dp" />

    <Button
        android:id="@+id/btnOK"
        android:layout_width="138dp"
        android:layout_height="36dp"
        android:layout_gravity="end"
        android:layout_marginRight="100dp"
        android:background="#009688"
        android:text="SUBMIT"
        android:textColor="#F8F3EFEF"
        android:textSize="18sp" />

    <Space
        android:layout_width="match_parent"
        android:layout_height="202dp" />
  </LinearLayout>

</ScrollView>
```

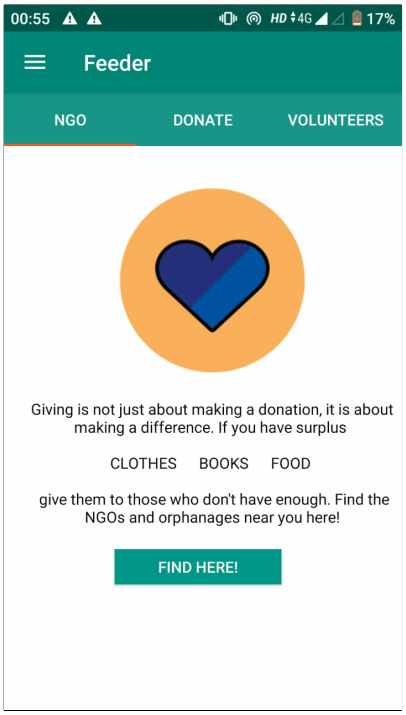
contactus.java

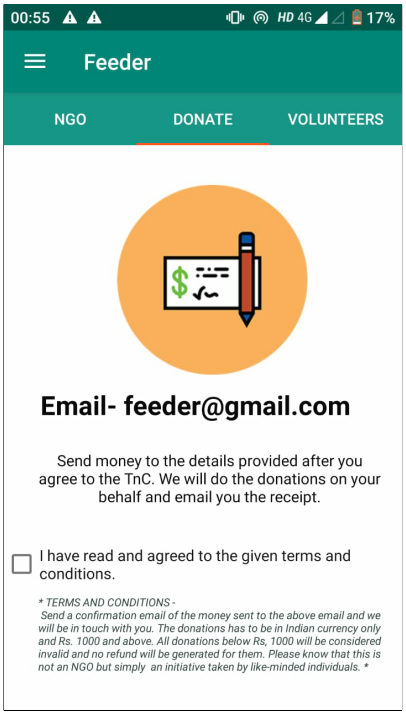
```
package ml.pritimainkar.afinal;


import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
```

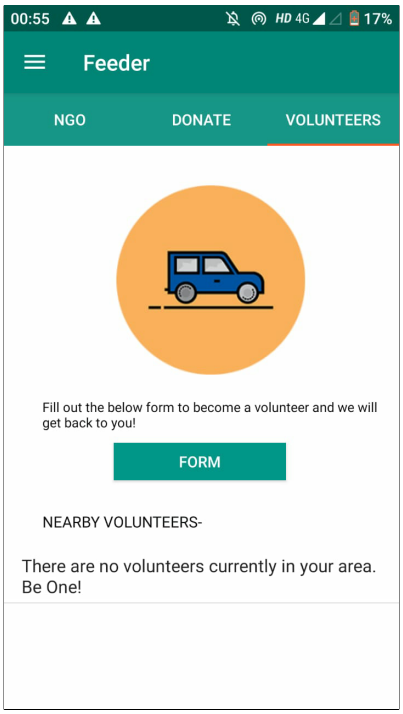
```
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class contactus extends AppCompatActivity{
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_contactus);

    ((Button) findViewById(R.id.btnOK)).setOnClickListener(new View.OnClickListener() {
      public void onClick(View v) {
        String to = "pritimainkar7@gmail.com";
        String sub = "feeder_contact_us";
        String full_name = ((EditText)findViewById(R.id.full_name)).getText().toString();
        String number = ((EditText)findViewById(R.id.number)).getText().toString();
        String ngo_name = ((EditText)findViewById(R.id.ngo_name)).getText().toString();
        String ngo_number = ((EditText)findViewById(R.id.ngo_number)).getText().toString();
        String ngo_email = ((EditText)findViewById(R.id.ngo_email)).getText().toString();

        String mess = ("FullName = "+ full_name+
            "\n Contact Number = "+ number+
            "\n NGO name = "+ ngo_name+
            "\n NGO number = "+ ngo_number+
            "\n NGO email = "+ ngo_email+
            "\n \n Please give Identity Proof before Sending Email - " +
            "\n Attach photos of adharcard, pancard,etc or give link of any active social media handle");

        Intent mail = new Intent(Intent.ACTION_SEND);
        mail.putExtra(Intent.EXTRA_EMAIL,new String[]{to});
        mail.putExtra(Intent.EXTRA_SUBJECT, sub);
        mail.putExtra(Intent.EXTRA_TEXT, mess);
        mail.setType("message/rfc822");
        startActivity(mail);
      }
    });
  }
}
```

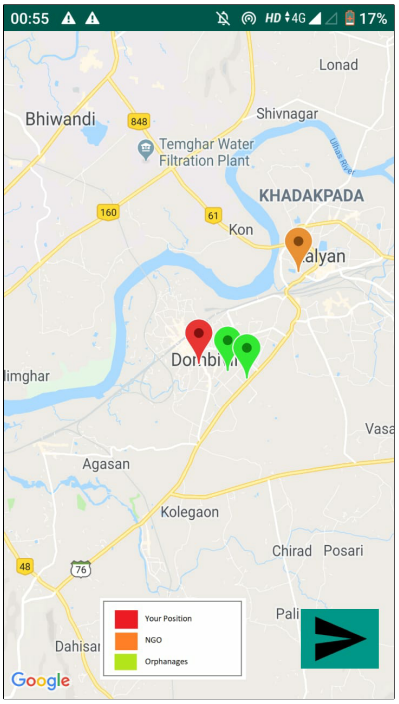
---

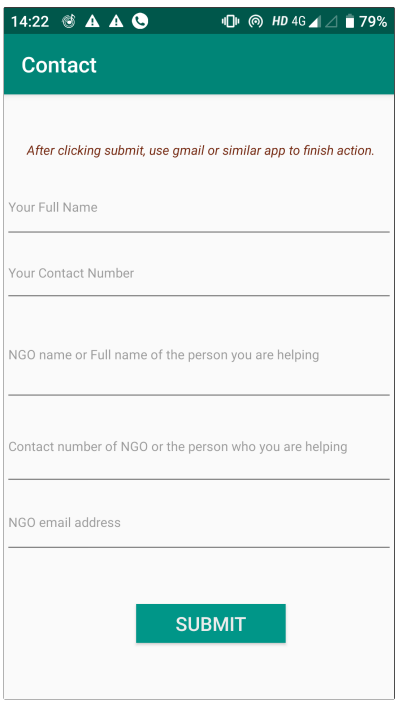
18. Screen Layouts and Report Layouts

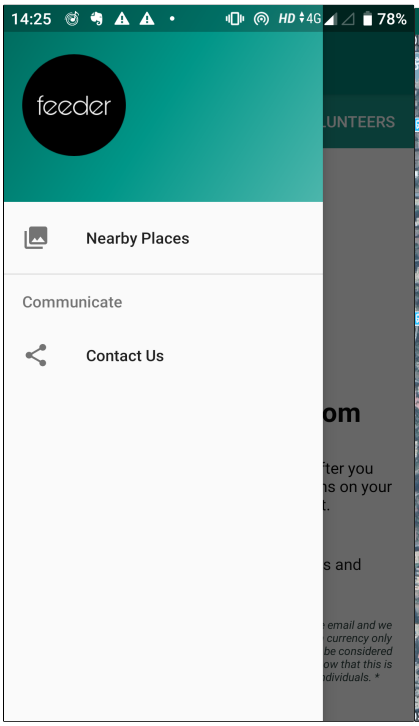


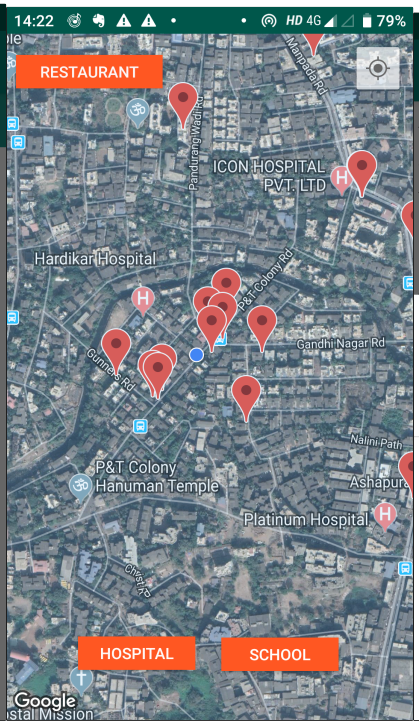


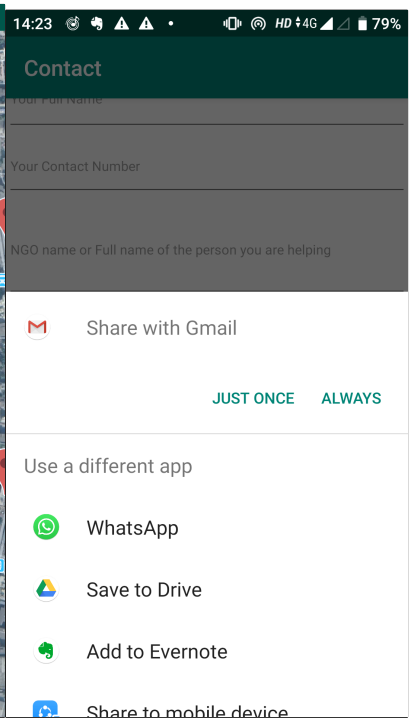


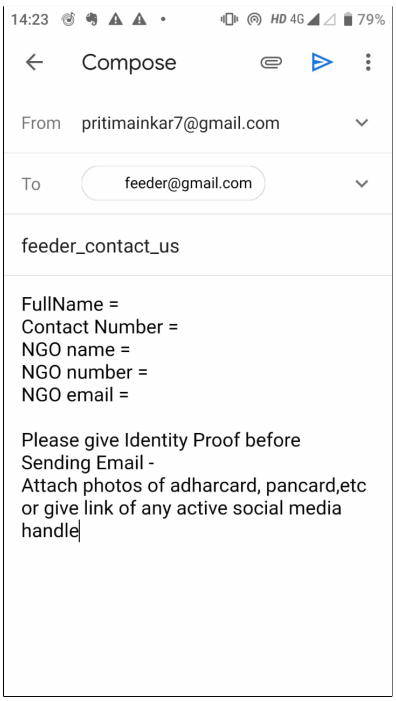





14:25
78%
feeder
Nearby Places
Communicate
Contact Us


14:22
79%
RESTAURANT
ICON HOSPITAL PVT. LTD
Hardikar Hospital
P&T Colony Hanuman Temple
Platinum Hospital
HOSPITAL
SCHOOL
Google


14:23
79%
Contact
Your Contact Number
NGO name or Full name of the person you are helping
Share with Gmail
JUST ONCE
ALWAYS
Use a different app
WhatsApp
Save to Drive
Add to Evernote


14:23
79%
Compose
From
pritimainkar7@gmail.com
To
feeder@gmail.com
feeder_contact_us
FullName =
Contact Number =
NGO name =
NGO number =
NGO email =
Please give Identity Proof before Sending Email -
Attach photos of adharcard, pancard,etc or give link of any active social media handle


00:55
17%
Feeder

19. Future Enhancement

As future magnification, some additional stuff could be implemented and integrated into the system making it more reliable and flexible.

1. The project can be extended to add chat functionality.
2. Can add option to start campaigns for people in need of immediate large amounts of money.
3. Can add functionality to send donations directly to ngo and orphanages.
4. Can add secure payment interface.
5. Can add more ngo and orphanages in map.
6. Can improve the security level, making the application more secure and reliable.

20. Bibliography

www.Microsoftdocs.com
www.stackoverflow.com
www.youtube.com
www.w3schools.com