

## FINAL LAB PROJECT

SUBMITTED BY:- MANSOOR NAWAZ AND ADNAN FAZAL

PROJECT NAME : BLOOD BANK DONATION

### DASHBOARD

```
package com.android.iunoob.bloodbank.activities;

import android.app.ProgressDialog;
import android.content.Intent;
import android.os.Bundle;
import android.support.annotation.NonNull;
import android.support.design.widget.FloatingActionButton;
import android.support.v4.app.Fragment;
import android.util.Log;
import android.view.View;
import android.support.design.widget.NavigationView;
import android.support.v4.view.GravityCompat;
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;
import android.widget.TextView;

import com.android.iunoob.bloodbank.R;
import com.android.iunoob.bloodbank.fragments.AboutUs;
import com.android.iunoob.bloodbank.fragments.AchievementsView;
import com.android.iunoob.bloodbank.fragments.BloodInfo;
import com.android.iunoob.bloodbank.fragments.HomeView;
import com.android.iunoob.bloodbank.fragments.NearByHospitalActivity;
import com.android.iunoob.bloodbank.fragments.SearchDonorFragment;
import com.android.iunoob.bloodbank.viewmodels.UserData;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.Query;
import com.google.firebase.database.ValueEventListener;

import static com.android.iunoob.bloodbank.R.id.home;

public class Dashboard extends AppCompatActivity
    implements NavigationView.OnNavigationItemSelectedListener {

    private FirebaseAuth mAuth;
    private TextView getUsername;
    private TextView getEmail;
    private FirebaseDatabase user_db;
    private FirebaseUser cur_user;
```

```

private DatabaseReference userdb_ref;

private ProgressDialog pd;

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

    pd = new ProgressDialog(this);
    pd.setMessage("Loading...");
    pd.setCancelable(true);
    pd.setCanceledOnTouchOutside(false);

    mAuth = FirebaseAuth.getInstance();
    user_db = FirebaseDatabase.getInstance();
    cur_user = mAuth.getCurrentUser();
    userdb_ref = user_db.getReference("users");

    getUserEmail = findViewById(R.id.UserEmailView);
   .getUserName = findViewById(R.id.UserNameView);
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);

    FloatingActionButton fab = (FloatingActionButton)
findViewById(R.id.fab);
    fab.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            startActivity(new Intent(Dashboard.this,
PostActivity.class));
        }
    });

    DrawerLayout drawer = (DrawerLayout)
findViewById(R.id.drawer_layout);
    ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(
        this, drawer, toolbar, R.string.navigation_drawer_open,
R.string.navigation_drawer_close);
    drawer.addDrawerListener(toggle);
    toggle.syncState();

    NavigationView navigationView = (NavigationView)
findViewById(R.id.nav_view);
    navigationView.setNavigationItemSelectedListener(this);
    View header = navigationView.getHeaderView(0);

    getUserEmail = (TextView) header.findViewById(R.id.UserEmailView);
    getUserName = (TextView) header.findViewById(R.id.UserNameView);

    Query singleuser = userdb_ref.child(cur_user.getId());
    pd.show();
    singleuser.addListenerForSingleValueEvent(new ValueEventListener()
{
    @Override
    public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

```

```

        //pd.show();
        String name =
dataSnapshot.getValue(UserData.class).getName();

        getUsername.setText(name);
        getUserEmail.setText(cur_user.getEmail());

        pd.dismiss();
    }

    @Override
    public void onCancelled(@NonNull DatabaseError databaseError) {
        Log.d("User", databaseError.getMessage());
    }
});

    if(savedInstanceState == null)
    {
getSupportFragmentManager().beginTransaction().replace(R.id.fragmentcontain
er, new HomeView()).commit();
        navigationView.getMenu().getItem(0).setChecked(true);
    }

}

    @Override
    public void onBackPressed() {
        DrawerLayout drawer = (DrawerLayout)
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.dashboard, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        // Handle action bar item clicks here. The action bar will
        // automatically handle clicks on the Home/Up button, so long
        // as you specify a parent activity in AndroidManifest.xml.
        int id = item.getItemId();

        if (id == R.id.donateinfo) {
getSupportFragmentManager().beginTransaction().replace(R.id.fragmentcontain

```

```

er, new BloodInfo()).commit();
    }
    if (id == R.id.devinfo) {
getSupportFragmentManager().beginTransaction().replace(R.id.fragmentcontain
er, new AboutUs()).commit();
    }

    return super.onOptionsItemSelected(item);
}

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

    if (id == home) {
getSupportFragmentManager().beginTransaction().replace(R.id.fragmentcontain
er, new HomeView()).commit();

        } else if (id == R.id.userprofile) {
            startActivity(new Intent(getApplicationContext(),
ProfileActivity.class));

        }
        else if (id == R.id.user_achiev) {
getSupportFragmentManager().beginTransaction().replace(R.id.fragmentcontain
er, new AchievementsView()).commit();

        }
        else if (id == R.id.logout) {
            mAuth.signOut();
            Intent intent = new Intent(getApplicationContext(),
LoginActivity.class);
            startActivity(intent);
        }
        else if (id == R.id.blood_storage){

getSupportFragmentManager().beginTransaction().replace(R.id.fragmentcontain
er, new SearchDonorFragment()).commit();

            } else if (id == R.id.nearby_hospital) {
getSupportFragmentManager().beginTransaction().replace(R.id.fragmentcontain
er, new NearByHospitalActivity()).commit();

            }

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

    @Override
    protected void onStart() {
        super.onStart();
        FirebaseUser currentUser = mAuth.getCurrentUser();

```

```

        if(currentUser == null)
        {
            Intent intent = new Intent(getApplicationContext(),
LoginActivity.class);
            startActivity(intent);
            finish();
        }
    }

    @Override
    protected void onResume() {
        super.onResume();
        FirebaseUser currentUser = mAuth.getCurrentUser();
        if(currentUser == null)
        {
            Intent intent = new Intent(getApplicationContext(),
LoginActivity.class);
            startActivity(intent);
            finish();
        }
    }
}

```

## LOGIN ACTIVITY

```

package com.android.iunoob.bloodbank.activities;

import android.app.ProgressDialog;
import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.android.iunoob.bloodbank.R;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;

public class LoginActivity extends AppCompatActivity {

    private Button signin, signup, resetpass;
    private EditText inputemail, inputpassword;
    private FirebaseAuth mAuth;
    private ProgressDialog pd;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

```

```

super.onCreate(savedInstanceState);
setContentView(R.layout.activity_login);

pd = new ProgressDialog(this);
pd.setMessage("Loading...");
pd.setCancelable(true);
pd.setCanceledOnTouchOutside(false);

mAuth = FirebaseAuth.getInstance();

if(mAuth.getCurrentUser() != null)
{
    Intent intent = new Intent(getApplicationContext(),
Dashboard.class);
    startActivity(intent);
    finish();
}

inputEmail = findViewById(R.id.input_username);
inputpassword = findViewById(R.id.input_password);

signin = findViewById(R.id.button_login);
signup = findViewById(R.id.button_register);
resetpass = findViewById(R.id.button_forgot_password);

signin.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        final String email = inputemail.getText().toString()+"";
        final String password =
inputpassword.getText().toString()+"";

        try {
            if(password.length()>0 && email.length()>0) {
                pd.show();
                mAuth.signInWithEmailAndPassword(email, password)
                    .addOnCompleteListener(LoginActivity.this,
new OnCompleteListener<AuthResult>() {
                        @Override
                        public void onComplete(@NonNull
Task<AuthResult> task) {
                            if (!task.isSuccessful()) {
                                Toast.makeText(getApplicationContext(),
                                    "Authentication
Failed",
                                Toast.LENGTH_LONG).show();
                                Log.v("error",
task.getException().getMessage());
                            } else {
                                Intent intent = new
Intent(getApplicationContext(), Dashboard.class);
                                startActivity(intent);
                                finish();
                            }
                        }
                    });
                pd.dismiss();
            }
        }
    }
});

```

```

        });
    }
    else
    {
        Toast.makeText(getApplicationContext(), "Please
fill all the field.", Toast.LENGTH_LONG).show();
    }

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

signup.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(getApplicationContext(),
ProfileActivity.class);
        startActivity(intent);
    }
});

resetpass.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(getApplicationContext(),
RestorePassword.class);
        startActivity(intent);
    }
});

}

}

```

## POST ACTIVITY

```

package com.android.iunoob.bloodbank.activities;

import android.app.ProgressDialog;
import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;

```

```

import android.widget.Toast;

import com.android.iunoob.bloodbank.R;
import com.android.iunoob.bloodbank.viewmodels.UserData;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.Query;
import com.google.firebase.database.ValueEventListener;

import java.util.Calendar;

public class PostActivity extends AppCompatActivity {

    ProgressDialog pd;

    EditText text1, text2;
    Spinner spinner1, spinner2;
    Button btnpost;

    FirebaseDatabase fdb;
    DatabaseReference db_ref;
    FirebaseAuth mAuth;

    Calendar cal;
    String uid;
    String Time, Date;

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

        pd = new ProgressDialog(this);
        pd.setMessage("Loading...");
        pd.setCancelable(true);
        pd.setCanceledOnTouchOutside(false);

        getSupportActionBar().setTitle("Post Blood Request");
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        text1 = findViewById(R.id.getMobile);
        text2 = findViewById(R.id.getLocation);

        spinner1 = findViewById(R.id.SpinnerBlood);
        spinner2 = findViewById(R.id.SpinnerDivision);

        btnpost = findViewById(R.id.postbtn);

        cal = Calendar.getInstance();

        int day = cal.get(Calendar.DAY_OF_MONTH);
        int month = cal.get(Calendar.MONTH);
        int year = cal.get(Calendar.YEAR);
    }

```



```

int hour = cal.get(Calendar.HOUR);
int min = cal.get(Calendar.MINUTE);
month+=1;
Time = "";
Date = "";
String ampm="AM";

if(cal.get(Calendar.AM_PM) ==1)
{
    ampm = "PM";
}

if(hour<10)
{
    Time += "0";
}
Time += hour;
Time += ":";

if(min<10) {
    Time += "0";
}

Time +=min;
Time +=(" "+ampm);

Date = day+"/"+month+"/"+year;

FirebaseUser cur_user = mAuth.getInstance().getCurrentUser();

if(cur_user == null)
{
    startActivity(new Intent(PostActivity.this,
LoginActivity.class));
} else {
    uid = cur_user.getId();
}

mAuth = FirebaseAuth.getInstance();
fdb = FirebaseDatabase.getInstance();
db_ref = fdb.getReference("posts");

try {
    btnpost.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            pd.show();
            final Query findname =
fdb.getReference("users").child(uid);

            if(text1.getText().length() == 0)
            {
                Toast.makeText(getApplicationContext(), "Enter your
contact number!",
                                Toast.LENGTH_LONG).show();
            }
            else if(text2.getText().length() == 0)

```

```

        {
            Toast.makeText(getApplicationContext(), "Enter your
location!",
                                Toast.LENGTH_LONG).show();
        }
        else {
            findname.addListenerForSingleValueEvent(new
ValueEventListener() {
                @Override
                public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
                    if (dataSnapshot.exists()) {
db_ref.child(uid).child("Name").setValue(dataSnapshot.getValue(UserData.cla
ss).getName());
db_ref.child(uid).child("Contact").setValue(text1.getText().toString());
db_ref.child(uid).child("Address").setValue(text2.getText().toString());
db_ref.child(uid).child("Division").setValue(spinner2.getSelectedItem().toS
tring());
db_ref.child(uid).child("BloodGroup").setValue(spinner1.getSelectedItem().t
oString());
db_ref.child(uid).child("Time").setValue(Time);
db_ref.child(uid).child("Date").setValue(Date);
                        Toast.makeText(PostActivity.this, "Your
post has been created successfully",
                                    Toast.LENGTH_LONG).show();
                        startActivity(new
Intent(PostActivity.this, Dashboard.class));

                    } else {
                        Toast.makeText(getApplicationContext(),
"Database error occured.",
                                    Toast.LENGTH_LONG).show();
                    }
                }

                @Override
                public void onCancelled(@NonNull DatabaseError
databaseError) {
                    Log.d("User", databaseError.getMessage());
                }
            });
        }
    });
}
catch (Exception e)
{
    e.printStackTrace();
}
pd.dismiss();
}

@Override

```

```

    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case android.R.id.home:
                onBackPressed();
                return true;
        }

        return super.onOptionsItemSelected(item);
    }
}

```

## PROFILE ACTIVITY

```

package com.android.iunoob.bloodbank.activities;

import android.app.ProgressDialog;
import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

import com.android.iunoob.bloodbank.R;
import com.android.iunoob.bloodbank.viewmodels.UserData;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.Query;
import com.google.firebase.database.ValueEventListener;

public class ProfileActivity extends AppCompatActivity {

    private EditText inputemail, inputpassword, retypePassword, fullName,
address, contact;
    private FirebaseAuth mAuth;
    private Button btnSignup;
    private ProgressDialog pd;
    private Spinner gender, bloodgroup, division;

    private boolean isUpdate = false;

    private DatabaseReference db_ref, donor_ref;

```

```

private FirebaseDatabase db_User;
private CheckBox isDonor;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    pd = new ProgressDialog(this);
    pd.setMessage("Loading...");
    pd.setCancelable(true);
    pd.setCanceledOnTouchOutside(false);
    pd.show();
    setContentView(R.layout.activity_profile);

    db_User = FirebaseDatabase.getInstance();
    db_ref = db_User.getReference("users");
    donor_ref = db_User.getReference("donors");
    mAuth = FirebaseAuth.getInstance();

    inputemail = findViewById(R.id.input_userEmail);
    inputpassword = findViewById(R.id.input_password);
    retypePassword = findViewById(R.id.input_password_confirm);
    fullName = findViewById(R.id.input_fullName);
    gender = findViewById(R.id.gender);
    address = findViewById(R.id.inputAddress);
    division = findViewById(R.id.inputDivision);
    bloodgroup = findViewById(R.id.inputBloodGroup);
    contact = findViewById(R.id.inputMobile);
    isDonor = findViewById(R.id.checkbox);

    btnSignup = findViewById(R.id.button_register);
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);

    if (mAuth.getCurrentUser() != null) {

        inputemail.setVisibility(View.GONE);
        inputpassword.setVisibility(View.GONE);
        retypePassword.setVisibility(View.GONE);
        btnSignup.setText("Update Profile");
        pd.dismiss();
        /// getActionBar().setTitle("Profile");
        getSupportActionBar().setTitle("Profile");
        findViewById(R.id.image_logo).setVisibility(View.GONE);
        isUpdate = true;

        Query Profile = db_ref.child(mAuth.getCurrentUser().getUid());
        Profile.addListenerForSingleValueEvent(new ValueEventListener()
        {
            @Override
            public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {

                UserData userData =
dataSnapshot.getValue(UserData.class);

                if (userData != null) {
                    pd.show();

```

```

        fullName.setText(userData.getName());
        gender.setSelection(userData.getGender());
        address.setText(userData.getAddress());
        contact.setText(userData.getContact());
        bloodgroup.setSelection(userData.getBloodGroup());
        division.setSelection(userData.getDivision());
        Query donor =
donor_ref.child(division.getSelectedItem().toString())
        .child(bloodgroup.getSelectedItem().toString())
            .child(mAuth.getCurrentUser().getUid());

        donor.addListenerForSingleValueEvent(new
ValueEventListener() {

            @Override
            public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {

                if(dataSnapshot.exists())
                {
                    isDonor.setChecked(true);
                    isDonor.setText("Unmark this to leave
from donors");
                }
                else
                {
                    Toast.makeText(ProfileActivity.this,
"Your are not a donor! Be a donor and save life by donating blood.",
                        Toast.LENGTH_LONG).show();
                }
                pd.dismiss();
            }

            @Override
            public void onCancelled(@NonNull DatabaseError
databaseError) {

                Log.d("User", databaseError.getMessage());
            }

        });

    }

    @Override
    public void onCancelled(@NonNull DatabaseError
databaseError) {

        Log.d("User", databaseError.getMessage());
    }

    });

    } else pd.dismiss();
    btnSignup.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

```

```

        final String email = inputemail.getText().toString();
        final String password = inputpassword.getText().toString();
        final String ConfirmPassword =
retypePassword.getText().toString();
        final String Name = fullName.getText().toString();
        final int Gender = gender.getSelectedItemId();
        final String Contact = contact.getText().toString();
        final int BloodGroup =
bloodgroup.getSelectedItemId();
        final String Address = address.getText().toString();
        final int Division = division.getSelectedItemId();
        final String blood =
bloodgroup.getSelectedItem().toString();
        final String div = division.getSelectedItem().toString();

        try {

            if (Name.length() <= 2) {
                ShowError("Name");
                fullName.requestFocusFromTouch();
            } else if (Contact.length() < 11) {
                ShowError("Contact Number");
                contact.requestFocusFromTouch();
            } else if (Address.length() <= 2) {
                ShowError("Address");
                address.requestFocusFromTouch();
            } else {
                if (!isUpdate) {
                    if (email.length() == 0) {
                        ShowError("Email ID");
                        inputemail.requestFocusFromTouch();
                    } else if (password.length() <= 5) {
                        ShowError("Password");
                        inputpassword.requestFocusFromTouch();
                    } else if (password.compareTo(ConfirmPassword)
!= 0) {
                        Toast.makeText(ProfileActivity.this,
"Password did not match!", Toast.LENGTH_LONG)
                            .show();
                        retypePassword.requestFocusFromTouch();
                    } else {
                        pd.show();
                        mAuth.createUserWithEmailAndPassword(email,
password)
                            .addOnCompleteListener(ProfileActivity.this, new
OnCompleteListener<AuthResult>() {

                                @Override
                                public void onComplete(@NonNull
Task<AuthResult> task) {

                                    if (!task.isSuccessful()) {
                                        Toast.makeText(ProfileActivity.this, "Registration failed! try agian.",
                                        Toast.LENGTH_LONG)
                                            .show();
                                        Log.v("error",
task.getException().getMessage());

                                    } else {

```

```

String id = mAuth.getCurrentUser().getUid();
db_ref.child(id).child("Name").setValue(Name);
db_ref.child(id).child("Gender").setValue(Gender);
db_ref.child(id).child("Contact").setValue(Contact);
db_ref.child(id).child("BloodGroup").setValue(BloodGroup);
db_ref.child(id).child("Address").setValue(Address);
db_ref.child(id).child("Division").setValue(Division);

if(isDonor.isChecked())
{
donor_ref.child(div).child(blood).child(id).child("UID").setValue(id).toString();
donor_ref.child(div).child(blood).child(id).child("LastDonate").setValue("Don't donate yet!");
donor_ref.child(div).child(blood).child(id).child("TotalDonate").setValue(0);
donor_ref.child(div).child(blood).child(id).child("Name").setValue(Name);
donor_ref.child(div).child(blood).child(id).child("Contact").setValue(Contact);
donor_ref.child(div).child(blood).child(id).child("Address").setValue(Address);

}

Toast.makeText(getApplicationContext(), "Welcome, your account has been created!", Toast.LENGTH_LONG)

.show();
Intent intent = new Intent(ProfileActivity.this, Dashboard.class);
startActivity(intent);

finish();
}
pd.dismiss();

});
}

} else {

String id = mAuth.getCurrentUser().getUid();
db_ref.child(id).child("Name").setValue(Name);
db_ref.child(id).child("Gender").setValue(Gender);
db_ref.child(id).child("Contact").setValue(Contact);
db_ref.child(id).child("BloodGroup").setValue(BloodGroup);
db_ref.child(id).child("Address").setValue(Address);
db_ref.child(id).child("Division").setValue(Division);

if(isDonor.isChecked())
{
donor_ref.child(div).child(blood).child(id).child("UID").setValue(id).toString();
donor_ref.child(div).child(blood).child(id).child("LastDonate").setValue("Don't donate yet!");
}
}
}

```

```

donor_ref.child(div).child(blood).child(id).child("TotalDonate").setValue(0);
donor_ref.child(div).child(blood).child(id).child("Name").setValue(Name);
donor_ref.child(div).child(blood).child(id).child("Contact").setValue(Contact);
donor_ref.child(div).child(blood).child(id).child("Address").setValue(Address);

        }
        else
        {

donor_ref.child(div).child(blood).child(id).removeValue();

        }
        Toast.makeText(getApplicationContext(), "Your
account has been updated!", Toast.LENGTH_LONG)
            .show();
        Intent intent = new
Intent(ProfileActivity.this, Dashboard.class);
        startActivity(intent);
        finish();
    }
    pd.dismiss();

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

private void ShowError(String error) {

    Toast.makeText(ProfileActivity.this, "Please, Enter a valid
" + error,
        Toast.LENGTH_LONG).show();
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case android.R.id.home:
            onBackPressed();
            return true;
    }

    return super.onOptionsItemSelected(item);
}
}

```

XML



## ACTIVITY DASHBOARD

```
<?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"
    class="com.google.android.gms.maps.SupportMapFragment"
    android:id="@+id/drawer_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true"
    tools:context=".activities.Dashboard"
    tools:openDrawer="start">

    <include
        layout="@layout/app_bar_dashboard"
        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_dashboard"
        app:menu="@menu/activity_dashboard_drawer" />

</android.support.v4.widget.DrawerLayout>
```

## ACTIVITY LOGIN

```
<?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"
    android:orientation="vertical"
    android:background="@drawable/red_button_bg"
    tools:context=".activities.LoginActivity">

    <LinearLayout
        android:padding="10dp"
        android:gravity="center"
        android:orientation="vertical"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <LinearLayout
            android:layout_marginTop="20dp"
            android:layout_gravity="center"
```

```

        android:orientation="vertical"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">

        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:background="@drawable/circle"
            android:src="@mipmap/blood_bank_icon_round" />

        <LinearLayout
            android:layout_marginTop="10dp"
            android:orientation="horizontal"
            android:layout_width="match_parent"
            android:layout_height="wrap_content">
            <ImageView
                android:layout_weight="1"
                android:src="@drawable/facebook_circle"
                android:layout_width="30dp"
                android:layout_height="30dp" />

            <ImageView
                android:layout_width="30dp"
                android:layout_height="30dp"
                android:layout_weight="1"
                android:src="@drawable/twitter_512" />
        </LinearLayout>

    </LinearLayout>

    <EditText
        android:textColor="@android:color/white"
        android:id="@+id/input_username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textEmailAddress"
        android:hint="@string/prompt_email" />

    <EditText
        android:textColor="@android:color/white"
        android:id="@+id/input_password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:hint="@string/prompt_password" />

    <Button
        android:textAllCaps="false"
        android:id="@+id/button_login"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:textStyle="bold"
        android:background="@drawable/signin_btn"
        android:textColor="@color/primary"
        android:text="@string/action_sign_in_short" />

    <Button

```

```

        android:textAllCaps="false"
        android:id="@+id/button_register"
        android:layout_width="wrap_content"
        android:layout_marginTop="20dp"
        android:layout_height="wrap_content"
        android:padding="2dp"
        android:textColor="@android:color/white"
        android:background="@drawable/transparent_black_ten_percent"
        android:text="@string/register" />

<Button
    android:textAllCaps="false"
    android:id="@+id/button_forgot_password"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:padding="2dp"
    android:textColor="@android:color/white"
    android:background="@drawable/transparent_black_ten_percent"
    android:text="@string/forgotpass" />

</LinearLayout>

</ScrollView>

```

## ACTIVITY PROFILE

```

<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"
    android:background="@drawable/red_button_bg"
    android:clipToPadding="false"
    android:fillViewport="true"
    android:scrollbarStyle="outsideOverlay"
    tools:context=".activities.ProfileActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:background="#FFFFFF"
        android:orientation="vertical"
        android:padding="10dp">

        <ImageView
            android:id="@+id/image_logo"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:layout_marginTop="?attr/actionBarSize"
            android:layout_marginBottom="40dp"
            android:background="@drawable/circle"
            android:src="@mipmap/blood_bank_icon_round" />

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="3dp">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Name:" />

    <EditText
        android:id="@+id/input_fullName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="52dp"
        android:hint="@string/displayName"
        android:inputType="text"
        android:textColor="#090909" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="3dp">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Sex:" />

    <Spinner
        android:id="@+id/gender"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="67dp"
        android:layout_weight="1"
        android:entries="@array/gender">

    </Spinner>
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="3dp">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Blood Group:" />

    <Spinner
        android:id="@+id/inputBloodGroup"
        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:layout_marginLeft="13dp"
        android:layout_weight="1"
        android:entries="@array/Blood_Group">

    </Spinner>
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="3dp">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Contact No:" />

    <EditText
        android:id="@+id/inputMobile"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:hint="@string/mobile"
        android:inputType="text"
        android:textColor="#0B0B0B" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="3dp">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Address:" />

    <EditText
        android:id="@+id/inputAddress"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="40dp"
        android:hint="@string/userAddress"
        android:inputType="text"
        android:textColor="#0B0B0B" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:padding="3dp">

    <TextView

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Division:" />

        <Spinner
            android:id="@+id/inputDivision"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="42dp"
            android:layout_weight="1"
            android:entries="@array/division_list"></Spinner>
    </LinearLayout>

    <EditText
        android:id="@+id/input_userEmail"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/prompt_email"
        android:inputType="textEmailAddress"
        android:textColor="#0B0B0B" />

    <EditText
        android:id="@+id/input_password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/prompt_password"
        android:inputType="textPassword"
        android:textColor="#0B0B0B" />

    <EditText
        android:id="@+id/input_password_confirm"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/prompt_confirm_password"
        android:inputType="textPassword"
        android:textColor="#0B0B0B" />

    <CheckBox
        android:id="@+id/checkbox"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Mark this to be a donor" />

    <Button
        android:id="@+id/button_register"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="40dp"
        android:layout_marginBottom="40dp"
        android:background="@drawable/signin_btn"
        android:text="@string/signup"
        android:textAllCaps="false"
        android:textColor="@color/colorPrimary" />

</LinearLayout>

```

```
</ScrollView>
```

## ACTIVITY SPLASH

```
<?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"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context=".activities.SplashActivity"
    android:gravity="center"
    android:orientation="vertical"
    android:background="@drawable/red_button_bg">

    <LinearLayout
        android:background="@drawable/circle"
        android:layout_gravity="center"
        android:orientation="vertical"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center">
        <ImageView
            android:id="@+id/load_image"
            android:layout_weight="1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:src="@drawable/spalsh_animation"/>
    </LinearLayout>
    <TextView
        android:layout_marginTop="15dp"
        android:textColor="@android:color/white"
        android:textSize="48sp"
        android:textStyle="bold"
        android:text="@string/app_name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Donate Blood, Save Life"
        android:textColor="#FFFFFF"
        android:textSize="24sp" />

</LinearLayout>
```

## BLOOD INFO

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:background="@color/primary"
    android:layout_height="match_parent">

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

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:background="@drawable/boarder"
            android:backgroundTint="@color/primary_dark"
            android:gravity="center"
            android:paddingTop="8dp"
            android:text="Why Should You Donate Blood?"
            android:textSize="20dp" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:background="@color/common_google_signin_btn_text_dark_default"
            android:gravity="start"
            android:padding="10dp"
            android:text="@string/info_"
            android:textSize="20sp" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:background="@drawable/boarder"
            android:backgroundTint="@color/primary_dark"
            android:gravity="center"
            android:paddingTop="8dp"
            android:text="@string/eligibility_"
            android:textSize="20sp" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:background="@color/common_google_signin_btn_text_dark_default"
            android:gravity="start"
            android:padding="10dp"
            android:text="@string/male_"
            android:textSize="20sp" />

```



```
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_marginLeft="10dp"
            android:layout_marginRight="10dp"
            android:background="@color/common_google_signin_btn_text_dark_default"
            android:gravity="start"
            android:padding="10dp"
            android:text="@string/female_"
            android:textSize="20sp" />

    </LinearLayout>

</ScrollView>
```