

# **MOBILENEX: A SIM CARD MANAGEMENT MOBILE APPLICATION**

**Project Documentation Submitted to the Faculty of the School of  
Engineering and Technology  
National University, Fairview Quezon City**

**In Partial Fulfillment of the Requirements for Mobile Programming  
MOBPROG**

## **Submitted By:**

Bumatay, Jules

Lazaro, Llorenz

Madrid, Maoi

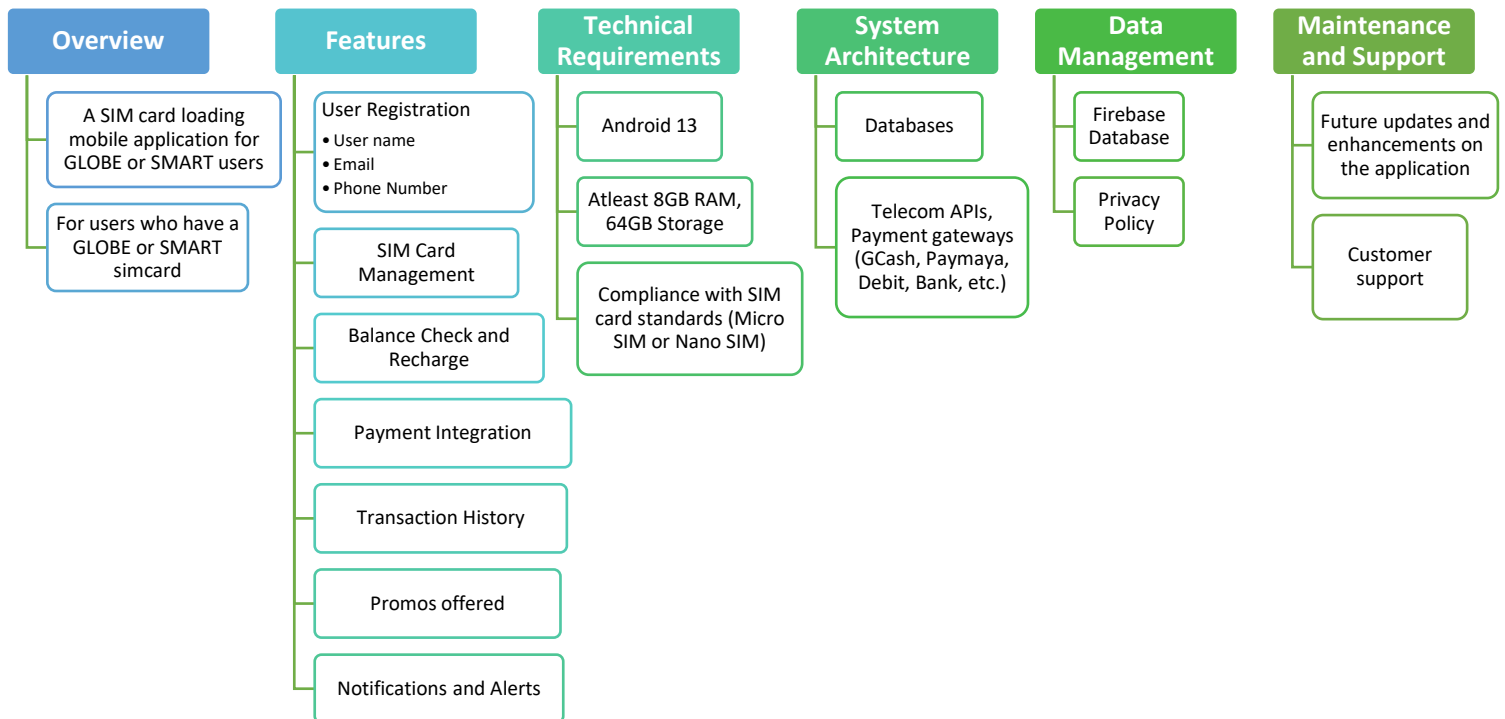
## I. Project Description

MobileNex is a SIM Card loading mobile application for users who have GLOBE or SMART SIM cards. The mobile application derives from SIM Card mobile applications such as SMART GigaLife, GOMO PH, and GCash. MobileNex, is an application that can allow users to view their current balance and the SIM Card offers. MobileNex also allows users to manage their balance and through the application itself. The overall concept of MobileNex is an all-in-one SIM Card management application that can make the users view, and manage their balance load in one without the need of switching applications to complete an action.

## II. System Documentation

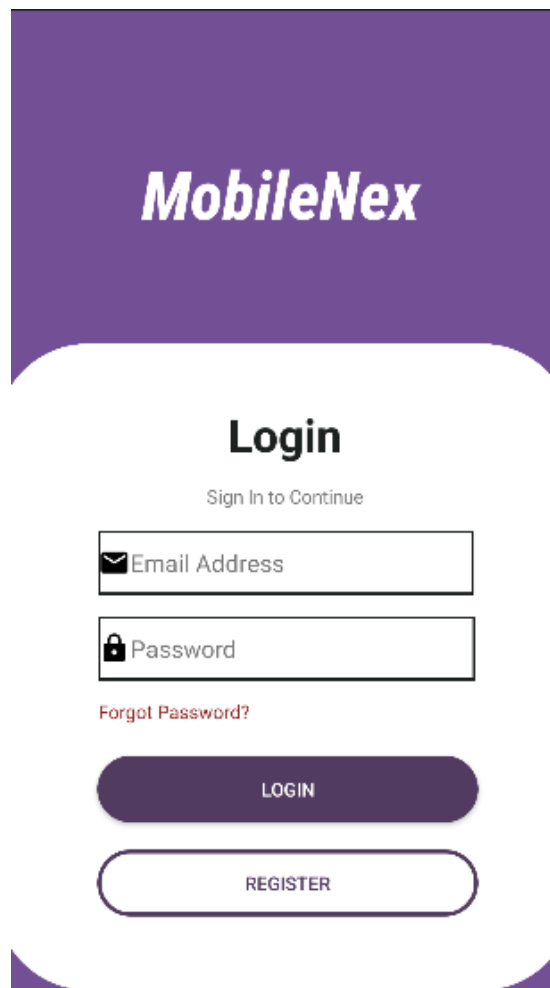
### A. Product Requirement Documentation

With MobileNex as a mobile application, it needs to be on a smartphone, specifically one that runs Android. Below is a diagram that shows the summary of the product requirement documentation (PRD) for MobileNex.



## B. UX Design Documentation

Here are the UX Designs of MobileNex.



The image shows a mobile application login screen. At the top, there is a purple header with the 'MobileNex' logo in white. Below the header, the word 'Login' is centered in a bold, black font. Underneath 'Login' is the text 'Sign In to Continue' in a smaller, gray font. There are two text input fields: the first is labeled 'Email Address' with an envelope icon, and the second is labeled 'Password' with a lock icon. Below the password field is a link that says 'Forgot Password?' in red text. At the bottom, there are two buttons: a purple 'LOGIN' button and a white 'REGISTER' button with a purple outline.

*Figure 1. Login Page*

The first one is the Login Page which appears upon opening MobileNex. The name of the application can be seen above whereas the login area is located just below it. There are text fields for the user's respective email address and password. There is also an option for the user in case they forgot their password. There are also two buttons which are login confirmation and user registration.

The image shows a mobile application screen for the 'Forgot Password' feature. The background is a solid purple color. In the center, there is a white rounded rectangle containing the following elements: a bold title 'FORGOT PASSWORD', a subtitle 'Please Provide your account email to retrieve password.', an email input field with a black envelope icon and the placeholder text 'Email Address', a dark purple rounded button labeled 'RESET PASSWORD', and a white rounded button with a purple border labeled 'BACK'.

**FORGOT PASSWORD**

Please Provide your account email to retrieve password.

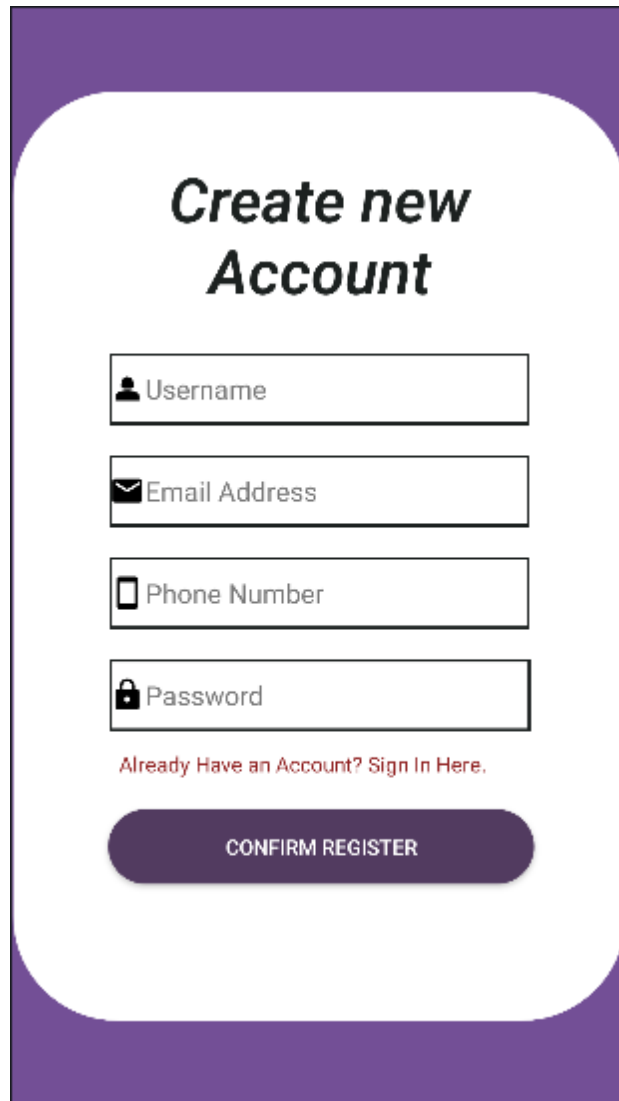
 Email Address

RESET PASSWORD

BACK

*Figure 2. Forgot Password Page*

The second page is the Forgot Password. Here, the screen pops up if the user clicks the "Forgot Password?" option. The application requests the user to input their email address that they initially inputted upon creating their account. There is a button to confirm the user to reset their password and a button to go back to the Login Page in case they changed their mind.

A mobile app registration screen with a purple header and footer. The main content area is white with rounded corners. At the top, the text "Create new Account" is displayed in a bold, black, sans-serif font. Below this, there are four vertically stacked input fields, each with a small icon on the left: a person icon for "Username", an envelope icon for "Email Address", a phone icon for "Phone Number", and a padlock icon for "Password". Below the input fields, there is a line of red text that reads "Already Have an Account? Sign In Here.". At the bottom of the form, there is a large, rounded, purple button with the text "CONFIRM REGISTER" in white, uppercase letters.

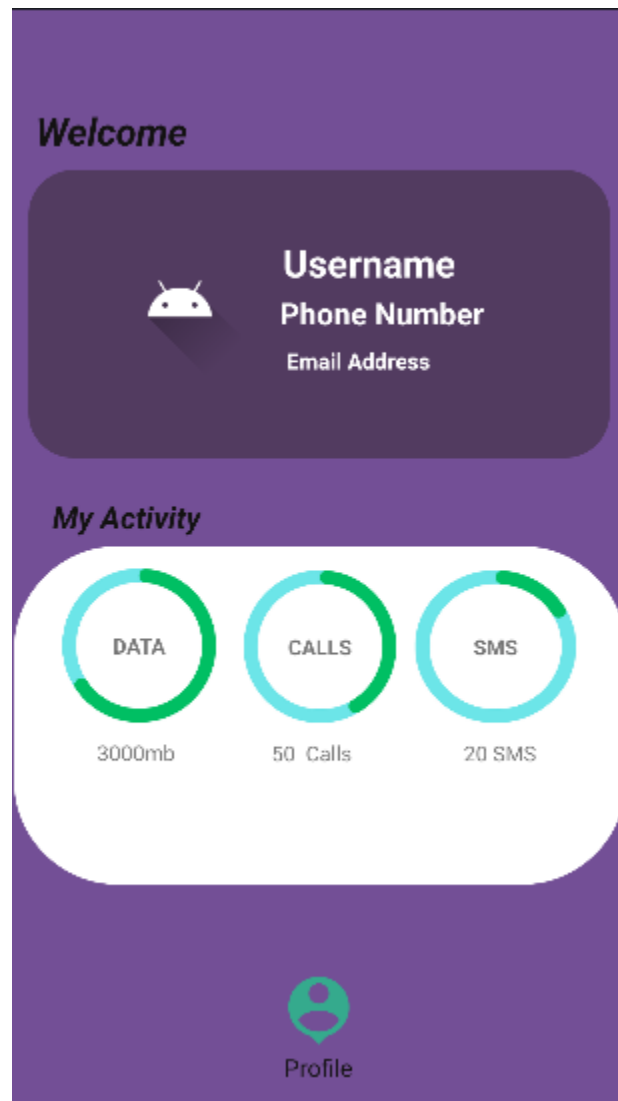
## Create new Account

Already Have an Account? Sign In Here.

CONFIRM REGISTER

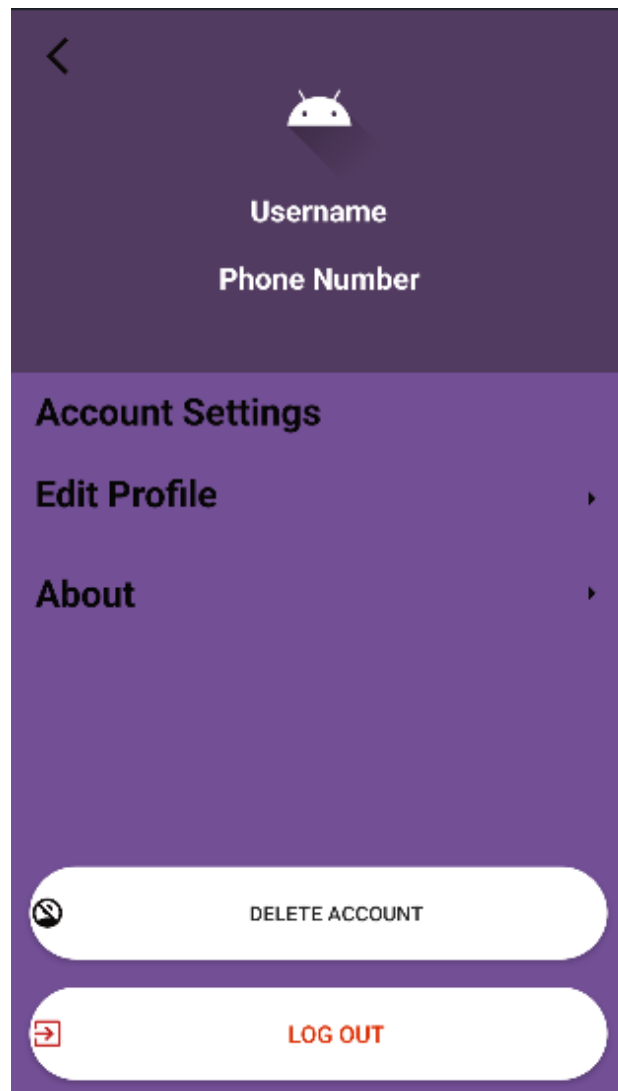
*Figure 3. Register Page*

The third one is the Register Page. Here there are four text fields which requires the user to input their respective username, email address, phone number, and password. There is also an option if the user already has an account which will direct them to the Login Page. Once they finish filling up the information needed, a "CONFIRM REGISTER" button is found at the bottom to make an account for MobileNex.



*Figure 4. Main Page*

The fourth one is the Main Page. The user is greeted with a display of their username, phone number and email address. It also displays the user's remaining data, the number of calls made, and the number of SMS text messages sent.



*Figure 5. Profile Page*

The fifth one is the Profile Page. The page displays the user's username and phone number. Here, the user may opt to edit their profile, or view what the application is about. There are two buttons below that allows the user to delete their account or log out of the application.

< Android

Username  
Phone Number

### Edit Profile

Username

Email

Phone Number

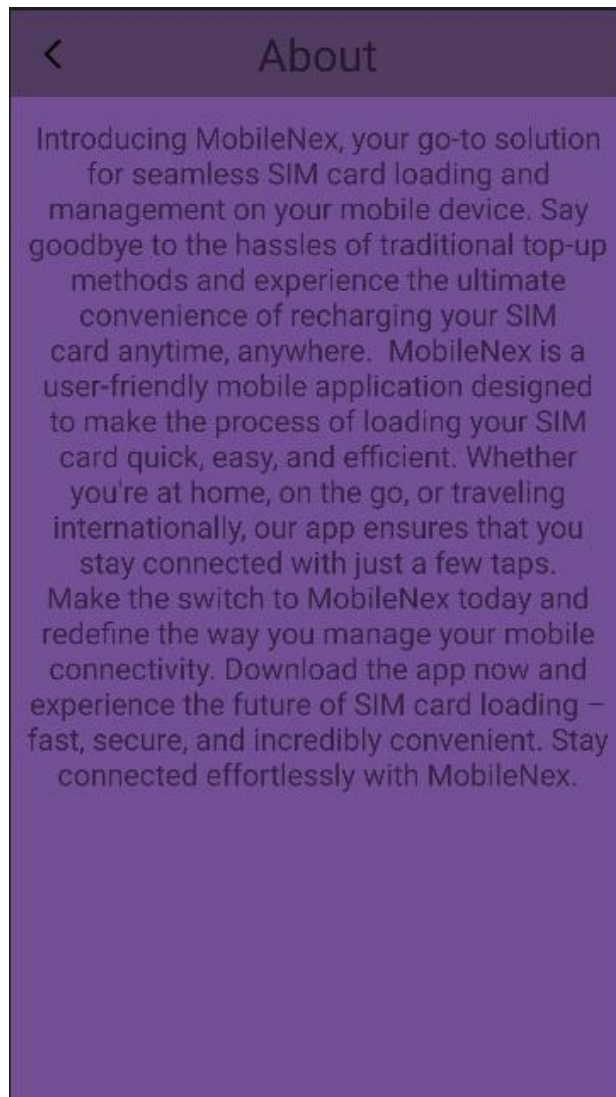
Password

EDIT PROFILE

*Figure 6. Edit Profile Page*

The sixth one is the Edit Profile Page. The page displays the user's username and phone number. Here, the user may edit either their username, email address, phone number, password, or everything else. To update the edits made, there is an edit profile button that is seen below to complete the action.

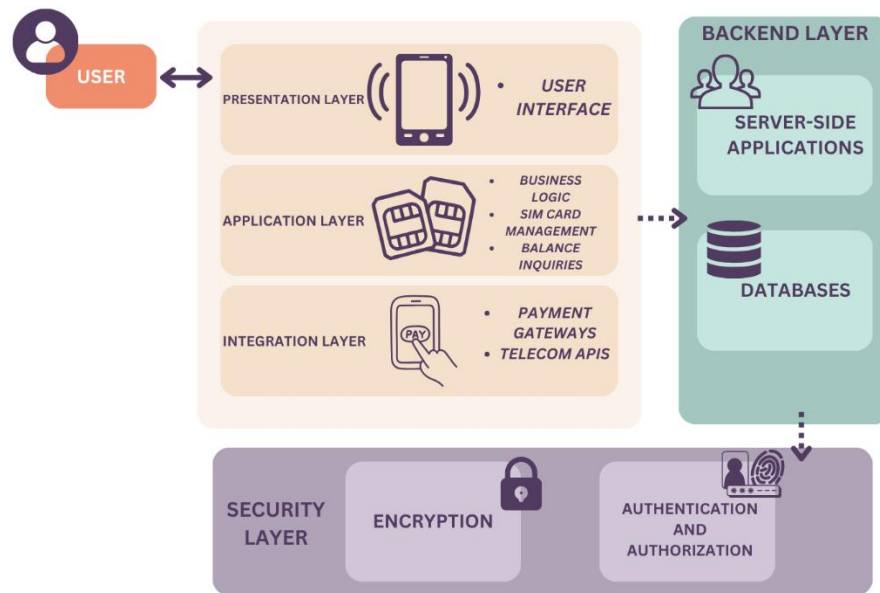




*Figure 7. About Page*

The seventh one is the About Page. The page displays what MobileNex is about. It also shows a back button so the user may exit the about page if they're done reading it.

## C. Software Architecture



This is a simple diagram showcasing the software architecture of MobileNex. It is simplified into 5 layers that make up MobileNex.

In the first layer is the Presentation Layer. It is the first layer that the user sees or interacts with upon accessing the application. It includes the user interface (UI) of the application. This includes the elements such as the Login and Registration screens, SIM Card management interface, balance check, and many more.

The next layer is the Application Layer. Here lies the business logic of the application. It is the one that manages user authentication, SIM Card management, balance inquiries, and payment integrations.

The next one is the Integration Layer. It consists of the payment gateways to be used for secure and convenient transactions. Another part of the layer are the Telecom APIs which interfaces with the user's used telecom services in their respective systems to facilitate SIM card promos, activation, and balance inquiries.

Next is the Backend Layer. Here are the server-side applications and the database. The data processing, interactions with external services. The database includes the user data which they have inputted creating an account, and the SIM Card information.

Last is the Security Layer which deals with encryption and authentication and authorization. Utilization of encryption protocols to secure the user data, payment transactions

happens here. Implementation of mechanisms for user authentication also happens here to ensure that only authorized users can access and make changes to the application.

#### D. Source Code Documentation

##### Activity main.java

```
package com.example.llorenz;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

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.auth.FirebaseUser;

public class MainActivity extends AppCompatActivity {

    EditText editTextEmail, editTextPassword;
    Button Login, buttonreg;
    FirebaseAuth mAuth;
    ProgressBar progressBar;
    TextView forgot;

    @Override
    public void onStart() {
        super.onStart();
        // Check if user is signed in (non-null) and update UI accordingly.
        FirebaseUser currentUser = mAuth.getCurrentUser();
        if(currentUser != null){
            Intent intent = new Intent(getApplicationContext(), MainActivitypage.class);
            startActivity(intent);
            finish();
        }
    }
}
```

```

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

    editTextEmail = findViewById(R.id.ResetEmail);
    editTextPassword = findViewById(R.id.Password);
    Login = findViewById(R.id.Resetbtttn);
    buttonreg = findViewById(R.id.bckbutton);
    forgat = findViewById(R.id.forgat);
    forgat.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View view) {
            Intent intent = new Intent(MainActivity.this,
MainActivityForgotpassword.class);
            startActivity(intent);
            finish();
        }
    });

    buttonreg.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(MainActivity.this,
MainActivityRegister.class);
            startActivity(intent);
            finish();
        }
    });

    Login.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            progressBar.setVisibility(View.VISIBLE);
            String email, password;
            email = String.valueOf(editTextEmail.getText());
            password = String.valueOf(editTextPassword.getText());

            if (TextUtils.isEmpty(email)){
                Toast.makeText(MainActivity.this, "Enter email",
Toast.LENGTH_SHORT).show();
                return;
            }
            if (TextUtils.isEmpty(password)){
                Toast.makeText(MainActivity.this, "Enter password",
Toast.LENGTH_SHORT).show();
                return;
            }
        }
    });
}

```

```

        mAuth.signInWithEmailAndPassword(email, password)
            .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    progressBar.setVisibility(View.GONE);
                    if (task.isSuccessful()) {
                        Toast.makeText(getApplicationContext(), "Login
Successful", Toast.LENGTH_SHORT).show();
                        Intent intent = new Intent(getApplicationContext(),
MainActivity.class);
                        startActivity(intent);
                        finish();

                    } else {

                        Toast.makeText(MainActivity.this, "Authentication
failed.",
                                Toast.LENGTH_SHORT).show();

                    }
                }
            });
    }
}

```

## Main Acitivity.xml

```
<androidx.constraintlayout.widget.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"
android:background="#734f96"
tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView3"
        android:layout_width="214dp"
        android:layout_height="64dp"
        android:fontFamily="sans-serif-condensed-medium"
        android:lineSpacingExtra="8sp"
        android:text="MobileNex"
        android:textAllCaps="false"
        android:textColor="#FFFFFF"
        android:textSize="50sp"
        android:textStyle="bold|italic"
        android:typeface="normal"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.502"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.154" />

    <androidx.constraintlayout.widget.ConstraintLayout
        android:id="@+id/constraintLayout2"
        android:layout_width="433dp"
        android:layout_height="483dp"
        android:background="@drawable/bg4"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="1.0">

        <EditText
            android:id="@+id/ResetEmail"
            android:layout_width="281dp"
            android:layout_height="48dp"
            android:background="@drawable/bg3"
            android:drawableLeft="@drawable/baseline_mail_24"
            android:ems="10"
            android:hint="Email Address"
            android:inputType="textEmailAddress"
            android:textColor="#787878"
            android:textColorHint="#787878"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
```

```
app:layout_constraintHorizontal_bias="0.507"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.321" />
```

#### <EditText

```
    android:id="@+id/Password"
    android:layout_width="282dp"
    android:layout_height="48dp"
    android:layout_marginTop="16dp"
    android:background="@drawable/bg3"
    android:drawableLeft="@drawable/baseline_lock_24"
    android:ems="10"
    android:hint="Password"
    android:inputType="textPassword"
    android:textColorHint="#787878"
    app:layout_constraintEnd_toEndOf="@+id/ResetEmail"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="@+id/ResetEmail"
    app:layout_constraintTop_toBottomOf="@+id/ResetEmail" />
```

#### <Button

```
    android:id="@+id/Resetbtttn"
    android:layout_width="285dp"
    android:layout_height="50dp"
    android:layout_marginTop="56dp"
    android:background="@drawable/bg"
    android:text="Login"
    android:textColor="#FFFFFF"
    app:iconTint="#B16262"
    app:layout_constraintEnd_toEndOf="@+id/Password"
    app:layout_constraintHorizontal_bias="0.333"
    app:layout_constraintStart_toStartOf="@+id/Password"
    app:layout_constraintTop_toBottomOf="@+id/Password" />
```

#### <TextView

```
    android:id="@+id/forgat"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="12dp"
    android:text="Forgot Password?"
    android:textColor="#912323"
    app:layout_constraintBottom_toTopOf="@+id/Resetbtttn"
    app:layout_constraintEnd_toEndOf="@+id/Password"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="@+id/Password"
    app:layout_constraintTop_toBottomOf="@+id/Password"
    app:layout_constraintVertical_bias="0.047" />
```

#### <Button

```
    android:id="@+id/bckbutton"
    android:layout_width="285dp"
    android:layout_height="50dp"
```

```
android:layout_marginTop="20dp"
android:background="@drawable/bg2"
android:text="Register"
android:textColor="#523B60"
app:layout_constraintEnd_toEndOf="@+id/Resetbtn"
app:layout_constraintHorizontal_bias="0.0"
app:layout_constraintStart_toStartOf="@+id/Resetbtn"
app:layout_constraintTop_toBottomOf="@+id/Resetbtn" />
```

<TextView

```
android:id="@+id/textView2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Login"
android:textColor="#1C2120"
android:textSize="34sp"
android:textStyle="bold"
app:layout_constraintBottom_toTopOf="@+id/ResetEmail"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.478" />
```

<TextView

```
android:id="@+id/textView4"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginBottom="16dp"
android:text="Sign In to Continue"
android:textColor="#787878"
app:layout_constraintBottom_toTopOf="@+id/ResetEmail"
app:layout_constraintEnd_toEndOf="@+id/textView2"
app:layout_constraintStart_toStartOf="@+id/textView2" />
```

</androidx.constraintlayout.widget.ConstraintLayout>

</androidx.constraintlayout.widget.ConstraintLayout>



## MainActivityregister.java

```
package com.example.llorenz;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FirebaseFirestore;

import java.util.HashMap;
import java.util.Map;

public class MainActivityRegister extends AppCompatActivity {

    public static final String TAG = "TAG";
    EditText editTextEmail, editTextPassword, editTextPhoneNumber, editTextUsername;
    Button buttonReg;
    TextView signIn;
    FirebaseAuth mAuth;
    ProgressBar progressBar;
    FirebaseFirestore fStore;
    String userId;

    @Override
    public void onStart() {
        super.onStart();
        // Check if user is signed in (non-null) and update UI accordingly.
        FirebaseUser currentUser = mAuth.getCurrentUser();
        if(currentUser != null){
            Intent intent = new Intent(getApplicationContext(), MainActivitypage.class);
            startActivity(intent);
            finish();
        }
    }
}
```

```

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

    mAuth= FirebaseAuth.getInstance();
    fStore= FirebaseFirestore.getInstance();

    editTextEmail = findViewById(R.id.Regemail);
    editTextPassword = findViewById(R.id.RegPassword);
    editTextPhoneNumber = findViewById(R.id.PhoneNumber);
    editTextUsername = findViewById(R.id.Username);
    signIn = findViewById(R.id.signIn);
    buttonReg = findViewById(R.id.Buttonreg);
    progressBar = findViewById(R.id.progressBar1);

    // Navigate to Sign In page
    signIn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(MainActivityRegister.this,
MainActivity.class);
            startActivity(intent);
            finish();
        }
    });
    //Register button connections

    buttonReg.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            progressBar.setVisibility(View.VISIBLE);
            //Obtain the entered data
            String email, password;
            email = String.valueOf(editTextEmail.getText());
            password = String.valueOf(editTextPassword.getText());
            String Phone = editTextPhoneNumber.getText().toString();
            String Username = editTextUsername.getText().toString();

            if (TextUtils.isEmpty(email)){
                Toast.makeText(MainActivityRegister.this, "Enter email",
Toast.LENGTH_SHORT).show();
                return;
            }
            if (TextUtils.isEmpty(password)){
                Toast.makeText(MainActivityRegister.this, "Enter password",

```

```

Toast.LENGTH_SHORT).show();
    return;
}
if (password.length() < 8) {
    editTextPassword.setError("Password Must be >= 8 Characters");
    return;
}
// Register the user in firebase
mAuth.createUserWithEmailAndPassword(email, password)
    .addOnCompleteListener( new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {
            progressBar.setVisibility(View.GONE);
            if (task.isSuccessful()) {
                //add user information in database
                Toast.makeText(MainActivityRegister.this, "Account
created.", Toast.LENGTH_SHORT).show();
                userId = mAuth.getCurrentUser().getUid();
                DocumentReference documentReference =
fStore.collection("users").document(userId);
                Map<String, Object> user = new HashMap<>();
                user.put("uname", Username);
                user.put("email", email);
                user.put("phone", Phone);
                user.put("pass", password);
                documentReference.set (user).addOnSuccessListener(new
OnSuccessListener<Void>() {
                    @Override
                    public void onSuccess(Void unused) {
                        Log.d(TAG, "onSuccess: user Profile is
created for"+ userId);
                    }
                });

                Intent intent = new Intent(getApplicationContext(),
MainActivity.class);

                startActivity(intent);
                finish();

            } else {
                // If sign in fails, display a message to the user.
                Toast.makeText(MainActivityRegister.this,
"Authentication failed.", Toast.LENGTH_SHORT).show();
            }
        }
    });
}
}
}
}

```

## Main Activity\_register.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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"
    android:background="#734f96"
    tools:context=".MainActivity">

    <ProgressBar
        android:id="@+id/progressBar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:visibility="gone"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="410dp"
        android:layout_height="621dp"
        android:background="@drawable/bg4"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.502">

        <TextView
            android:id="@+id/textView3"
            android:layout_width="262dp"
            android:layout_height="95dp"
            android:fontFamily="sans-serif-light"
            android:text="Create new Account"
            android:textAlignment="center"
            android:textColor="#1C2120"
            android:textSize="41sp"
            android:textStyle="bold|italic"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.083" />

        <EditText
            android:id="@+id/Username"
            android:layout_width="281dp"
            android:layout_height="48dp"
            android:layout_marginTop="36dp"
```

```
android:background="@drawable/bg3"
android:drawableLeft="@drawable/baseline_person_2_24"
android:ems="10"
android:hint="Username"
android:inputType="textEmailAddress"
android:textColor="#1C2120"
android:textColorHint="#787878"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView3" />
```

<EditText

```
android:id="@+id/Regemail"
android:layout_width="281dp"
android:layout_height="48dp"
android:layout_marginTop="20dp"
android:background="@drawable/bg3"
android:drawableLeft="@drawable/baseline_mail_24"
android:ems="10"
android:hint="Email Address"
android:inputType="textEmailAddress"
android:textColor="#1C2120"
android:textColorHint="#787878"
app:layout_constraintEnd_toEndOf="@+id/Username"
app:layout_constraintHorizontal_bias="0.0"
app:layout_constraintStart_toStartOf="@+id/Username"
app:layout_constraintTop_toBottomOf="@+id/Username" />
```

<EditText

```
android:id="@+id/RegPassword"
android:layout_width="282dp"
android:layout_height="48dp"
android:layout_marginTop="20dp"
android:background="@drawable/bg3"
android:drawableLeft="@drawable/baseline_lock_24"
android:ems="10"
android:hint="Password"
android:inputType="textPassword"
android:textColor="#1C2120"
android:textColorHint="#787878"
app:layout_constraintEnd_toEndOf="@+id/PhoneNumber"
app:layout_constraintHorizontal_bias="0.0"
app:layout_constraintStart_toStartOf="@+id/PhoneNumber"
app:layout_constraintTop_toBottomOf="@+id/PhoneNumber" />
```

<EditText

```
android:id="@+id/PhoneNumber"
android:layout_width="281dp"
android:layout_height="48dp"
android:layout_marginTop="20dp"
android:background="@drawable/bg3"
android:drawableLeft="@drawable/baseline_smartphone_24"
android:ems="10"
```

```

        android:hint="Phone Number"
        android:inputType="textEmailAddress"
        android:textColorHint="#787878"
        app:layout_constraintEnd_toEndOf="@+id/Regemail"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="@+id/Regemail"
        app:layout_constraintTop_toBottomOf="@+id/Regemail" />

```

#### <TextView

```

        android:id="@+id/FrgtPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp"
        android:layout_marginEnd="108dp"
        android:text="Already Have an Account? "
        android:textColor="#912323"
        app:layout_constraintEnd_toEndOf="@+id/PhoneNumber"
        app:layout_constraintTop_toBottomOf="@+id/RegPassword" />

```

#### <TextView

```

        android:id="@+id/signIn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp"
        android:text="Sign In Here."
        android:textColor="#912323"
        app:layout_constraintBottom_toTopOf="@+id/Buttonreg"
        app:layout_constraintEnd_toEndOf="@+id/RegPassword"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toEndOf="@+id/FrgtPassword"
        app:layout_constraintTop_toBottomOf="@+id/RegPassword"
        app:layout_constraintVertical_bias="0.0" />

```

#### <Button

```

        android:id="@+id/Buttonreg"
        android:layout_width="285dp"
        android:layout_height="50dp"
        android:layout_marginTop="52dp"
        android:background="@drawable/bg"
        android:text="Confirm Register"
        android:textColor="#FFFFFF"
        app:layout_constraintEnd_toEndOf="@+id/RegPassword"
        app:layout_constraintHorizontal_bias="0.333"
        app:layout_constraintStart_toStartOf="@+id/RegPassword"
        app:layout_constraintTop_toBottomOf="@+id/RegPassword" />

```

</androidx.constraintlayout.widget.ConstraintLayout>

</androidx.constraintlayout.widget.ConstraintLayout>

## MainActivityforgotpassword.java

```
package com.example.llorenz;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;

public class MainActivityForgotpassword extends AppCompatActivity {

    Button Resetbtn, bckbutton;
    EditText ResetEmail;
    ProgressBar progressBar;
    FirebaseAuth mAuth;
    String Email;

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

        Resetbtn = findViewById(R.id.Resetbtn);
        bckbutton = findViewById(R.id.bckbutton);
        ResetEmail = findViewById(R.id.ResetEmail);
        progressBar = findViewById(R.id.forgetprogressBar);

        mAuth = FirebaseAuth.getInstance();

        Resetbtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Email = ResetEmail.getText().toString().trim();

                if (Email.isEmpty()) {
                    Toast.makeText(MainActivityForgotpassword.this, "Please provide your email", Toast.LENGTH_SHORT).show();
                }else{
                    forgotpassword();
                }
            }
        });
    }
}
```

```

    }
}

});

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

}

private void forgotpassword() {

    FirebaseAuth auth = FirebaseAuth.getInstance();

    auth.sendPasswordResetEmail(Email)
        .addOnCompleteListener(new OnCompleteListener<Void>() {
            @Override
            public void onComplete(@NonNull Task<Void> task) {
                if (task.isSuccessful()){
                    Toast.makeText(MainActivityForgotpassword.this, "Check your
email", Toast.LENGTH_SHORT).show();
                    startActivity(new Intent(MainActivityForgotpassword.this,
MainActivity.class));
                    finish();

                }else{
                    Toast.makeText(MainActivityForgotpassword.this, "Error
:"+task.getException().getMessage(), Toast.LENGTH_SHORT).show();
                }
            }
        });
}

}

}

```



## Main Activity forgotpassword.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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"
    android:background="#734f96"
    tools:context=".MainActivity">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:id="@+id/constraintLayout"
        android:layout_width="410dp"
        android:layout_height="419dp"
        android:background="@drawable/bg4"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.493"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.486">

        <EditText
            android:id="@+id/ResetEmail"
            android:layout_width="281dp"
            android:layout_height="48dp"
            android:background="@drawable/bg3"
            android:drawableLeft="@drawable/baseline_mail_24"
            android:ems="10"
            android:hint="Email Address"
            android:inputType="textEmailAddress"
            android:textColorHint="#787878"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.508"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.468" />

        <Button
            android:id="@+id/Resetbtttn"
            android:layout_width="285dp"
            android:layout_height="50dp"
            android:layout_marginTop="32dp"
            android:background="@drawable/bg"
            android:text="RESET PASSWORD"
            android:textColor="#FFFFFF"
            app:iconTint="#B16262"
            app:layout_constraintEnd_toEndOf="@+id/ResetEmail"
            app:layout_constraintHorizontal_bias="1.0"
```

```
app:layout_constraintStart_toStartOf="@+id/ResetEmail"
app:layout_constraintTop_toBottomOf="@+id/ResetEmail" />
```

#### <Button

```
android:id="@+id/bckbutton"
android:layout_width="285dp"
android:layout_height="50dp"
android:layout_marginTop="24dp"
android:background="@drawable/bg2"
android:text="Back"
android:textColor="#523B60"
app:layout_constraintEnd_toEndOf="@+id/Resetbtttn"
app:layout_constraintHorizontal_bias="0.0"
app:layout_constraintStart_toStartOf="@+id/Resetbtttn"
app:layout_constraintTop_toBottomOf="@+id/Resetbtttn" />
```

#### <TextView

```
android:id="@+id/textView"
android:layout_width="331dp"
android:layout_height="35dp"
android:layout_marginBottom="20dp"
android:text="Please Provide your account email to retrieve password."
android:textAlignment="center"
android:textColor="#000000"
android:textSize="14sp"
app:layout_constraintBottom_toTopOf="@+id/ResetEmail"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent" />
```

#### <TextView

```
android:id="@+id/textView3"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="FORGOT PASSWORD"
android:textColor="#000000"
android:textSize="30sp"
android:textStyle="bold|italic"
app:layout_constraintBottom_toTopOf="@+id/textView"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.526"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.844" />
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

#### <ProgressBar

```
android:id="@+id/forgetprogressBar"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:visibility="gone"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
```

```
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toTopOf="parent" />
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

### MainActivityPage.java

```
package com.example.llorenz;  
  
import androidx.annotation.Nullable;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ImageView;  
import android.widget.TextView;  
  
import com.google.firebase.auth.FirebaseAuth;  
import com.google.firebase.firestore.DocumentReference;  
import com.google.firebase.firestore.DocumentSnapshot;  
import com.google.firebase.firestore.EventListener;  
import com.google.firebase.firestore.FirebaseFirestore;  
import com.google.firebase.firestore.FirebaseFirestoreException;  
  
import java.util.Objects;  
  
public class MainActivityPage extends AppCompatActivity {  
  
    FirebaseAuth fAuth;  
    FirebaseFirestore fstore;  
    TextView textViewemail, textViewphone, textViewusername;  
    String userId;  
  
    ImageView imageViewsetting;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main_activitypage);  
  
  
        textViewemail = findViewById(R.id.ProfilepageEmail);  
        textViewphone = findViewById(R.id.Profileppagehone);  
        textViewusername = findViewById(R.id.profilepageUsername);  
        imageViewsetting = findViewById(R.id.settings);  
    }  
}
```

```

fAuth = FirebaseAuth.getInstance();
fstore = FirebaseFirestore.getInstance();

imageViewsetting.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent(MainActivitypage.this,
MainActivityProfilepage.class);
        startActivity(intent);
        finish();
    }
});

userId = Objects.requireNonNull(fAuth.getCurrentUser()).getUid();

DocumentReference documentReference =
fstore.collection("users").document(userId);
documentReference.addSnapshotListener(this, new EventListener<DocumentSnapshot>()
{
    @Override
    public void onEvent(@Nullable DocumentSnapshot documentSnapshot, @Nullable
FirebaseFirestoreException error) {
        textViewphone.setText(documentSnapshot.getString("phone"));
        textViewusername.setText(documentSnapshot.getString("uname"));
        textViewemail.setText(documentSnapshot.getString("email"));
    }
});

}
}

```

## Main\_Activity\_Page.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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"
    android:background="#734f96"
    android:gravity="center"
    android:orientation="vertical"
    android:visibility="visible"
    tools:context=".MainActivity" >

    <androidx.constraintlayout.widget.ConstraintLayout
        android:id="@+id/layout1"
        android:layout_width="388dp"
        android:layout_height="192dp"
        android:background="@drawable/bg5"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.478"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.198" >

        <ProgressBar
            android:id="@+id/progressBar1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:visibility="gone"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent" />

        <ImageView
            android:id="@+id/Profilepicpage"
            android:layout_width="101dp"
            android:layout_height="106dp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.177"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.418"
            app:srcCompat="@drawable/ic_launcher_foreground" />

        <TextView
            android:id="@+id/Profileppagehone"
            android:layout_width="165dp"
            android:layout_height="28dp"
            android:text="Phone Number"
            android:textColor="#FFFFFF"
```

```
    android:textSize="20sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.753"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

<TextView

```
    android:id="@+id/profilepageUsername"
    android:layout_width="163dp"
    android:layout_height="28dp"
    android:text="Username"
    android:textColor="#FFFFFF"
    android:textSize="25sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.755"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.274" />
```

<TextView

```
    android:id="@+id/ProfilepageEmail"
    android:layout_width="213dp"
    android:layout_height="25dp"
    android:text=" Email Address"
    android:textColor="#FFFFFF"
    android:textSize="15sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.96"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.7" />
```

</androidx.constraintlayout.widget.ConstraintLayout>

<TextView

```
    android:id="@+id/textView9"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="My Activity"
    android:textColor="#131212"
    android:textSize="20sp"
    android:textStyle="bold|italic"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.086"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
```

```

        app:layout_constraintVertical_bias="0.461" />

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Welcome"
    android:textColor="#131212"
    android:textSize="25sp"
    android:textStyle="bold|italic"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.051"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.091" />

<ImageView
    android:id="@+id/settings"
    android:layout_width="58dp"
    android:layout_height="54dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.945"
    app:srcCompat="@drawable/baseline_person_24" />

<TextView
    android:id="@+id/textView6"
    android:layout_width="56dp"
    android:layout_height="20dp"
    android:text=" Profile"
    android:textColor="#131212"
    android:textSize="16sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="@+id/settings"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toStartOf="@+id/settings"
    app:layout_constraintTop_toBottomOf="@+id/settings"
    app:layout_constraintVertical_bias="0.0" />

<androidx.constraintlayout.widget.ConstraintLayout
    android:layout_width="409dp"
    android:layout_height="226dp"
    android:background="@drawable/bg4"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.478"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/layout1"
    app:layout_constraintVertical_bias="0.284">

```

```
<ImageView
    android:id="@+id/imageView"
    android:layout_width="112dp"
    android:layout_height="103dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.094"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.121"
    app:srcCompat="@drawable/progressbar1" />
```

```
<ImageView
    android:id="@+id/imageView2"
    android:layout_width="109dp"
    android:layout_height="102dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.129"
    app:srcCompat="@drawable/progressbar2" />
```

```
<ImageView
    android:id="@+id/imageView5"
    android:layout_width="108dp"
    android:layout_height="102dp"
    android:background="@drawable/progressbar4"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.89"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.129" />
```

```
<TextView
    android:id="@+id/textView11"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="DATA"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="@+id/imageView"
    app:layout_constraintEnd_toEndOf="@+id/imageView"
    app:layout_constraintStart_toStartOf="@+id/imageView"
    app:layout_constraintTop_toTopOf="@+id/imageView" />
```

```
<TextView
    android:id="@+id/textView12"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="CALLS"
    android:textStyle="bold"
```



```
app:layout_constraintBottom_toBottomOf="@+id/imageView2"
app:layout_constraintEnd_toEndOf="@+id/imageView2"
app:layout_constraintStart_toStartOf="@+id/imageView2"
app:layout_constraintTop_toTopOf="@+id/imageView2" />
```

<TextView

```
    android:id="@+id/textView13"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="SMS"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="@+id/imageView5"
    app:layout_constraintEnd_toEndOf="@+id/imageView5"
    app:layout_constraintStart_toStartOf="@+id/imageView5"
    app:layout_constraintTop_toTopOf="@+id/imageView5" />
```

<TextView

```
    android:id="@+id/textView14"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="68dp"
    android:text="3000mb"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.156"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView" />
```

<TextView

```
    android:id="@+id/textView15"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="76dp"
    android:text="50 Calls"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.26"
    app:layout_constraintStart_toEndOf="@+id/textView14"
    app:layout_constraintTop_toBottomOf="@+id/imageView2"
    app:layout_constraintVertical_bias="0.846" />
```

<TextView

```
    android:id="@+id/textView16"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginEnd="60dp"
    android:layout_marginBottom="76dp"
    android:text="20 SMS"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toEndOf="@+id/textView15"
    app:layout_constraintTop_toBottomOf="@+id/imageView5"
```

```
        app:layout_constraintVertical_bias="0.846" />
    </androidx.constraintlayout.widget.ConstraintLayout>
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

## MainActivityProfilePage.java

```
package com.example.llorenz;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.EventListener;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.FirebaseFirestoreException;

import java.util.Objects;

public class MainActivityProfilepage extends AppCompatActivity {

    FirebaseAuth fAuth;

    FirebaseUser user;
    FirebaseFirestore fstore;
    Button deletebutton;

    Button button;
    TextView textViewphone, textViewusername, texteditprofile, EditAbout;
    String userId;
```

```

    ImageView back;

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

        button = findViewById(R.id.button2);

        EditAbout = findViewById(R.id.About);
        texteditprofile = findViewById(R.id.EditProfile);
        textviewphone = findViewById(R.id.Profilepagephone);
        textviewusername = findViewById(R.id.profilepageUsername);
        deletebutton = findViewById(R.id.deletebu);
        back = findViewById(R.id.backbuttontopage);

        FirebaseAuth.getInstance();
        FirebaseFirestore.getInstance();

        userId = Objects.requireNonNull(FirebaseAuth.getInstance().getCurrentUser()).getUid();

        EditAbout.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(MainActivityProfilepage.this,
MainActivityAbout.class);
                startActivity(intent);
                finish();
            }
        });

        texteditprofile.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(MainActivityProfilepage.this,
MainActivityEditProfile.class);
                startActivity(intent);
                finish();
            }
        });

        button.setOnClickListener(new View.OnClickListener() {
            @Override

```

```

        public void onClick(View v) {

            FirebaseAuth.signOut();

            Intent intent = new Intent(getApplicationContext(), MainActivity.class);
            startActivity(intent);
            finish();
        }
    });

    DatabaseReference documentReference =
    firestore.collection("users").document(userId);
    documentReference.addSnapshotListener(this, new EventListener<DocumentSnapshot>()
    {
        @Override
        public void onEvent(@Nullable DocumentSnapshot documentSnapshot, @Nullable
        FirebaseFirestoreException error) {
            textViewphone.setText(documentSnapshot.getString("phone"));
            textViewusername.setText(documentSnapshot.getString("uname"));
        }
    });

    back.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(MainActivityProfilepage.this,
            MainActivitypage.class);
            startActivity(intent);
            finish();
        }
    });

    user = FirebaseAuth.getCurrentUser();
    if (user != null) {
        userId = ((FirebaseUser) user).getUid();
    } else {
        // Handle the case when the user is not logged in
        // You might want to redirect the user to the login page
        Intent intent = new Intent(this, MainActivity.class);
        startActivity(intent);
        finish();
    }

    // Button to delete the user account and data

```

```

// Button to delete the user account and data
deletebutton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        // Method to delete the user account and data
        deleteUserAccountAndData();
    }
});

}

private void deleteUserAccountAndData() {
    if (user != null) {
        // Delete user data from Firestore
        firestore.collection("users").document(userId)
            .delete()
            .addOnSuccessListener(new OnSuccessListener<Void>() {
                @Override
                public void onSuccess(Void aVoid) {
                    // User data deleted successfully
                    Toast.makeText(MainActivityProfilepage.this, "User data
deleted.", Toast.LENGTH_SHORT).show();

                    // Delete the user account
                    deleteUserAccount();
                }
            })
            .addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {
                    // Failed to delete user data
                    Toast.makeText(MainActivityProfilepage.this, "Failed to
delete user data. Please try again.", Toast.LENGTH_SHORT).show();
                }
            });
    }
}

private void deleteUserAccount() {
    if (user != null) {
        // Delete the user account from Firebase Authentication
        user.delete().addOnCompleteListener(new OnCompleteListener<Void>() {
            @Override
            public void onComplete(@NonNull Task<Void> task) {
                if (task.isSuccessful()) {
                    // Account deleted successfully
                    Toast.makeText(MainActivityProfilepage.this, "Account deleted.",
Toast.LENGTH_SHORT).show();

                    // Sign out the user
                    FirebaseAuth.signOut();

                    // Redirect to the Login or registration page
                    Intent intent = new Intent(getApplicationContext(),

```

```
MainActivity.class);
        startActivity(intent);
        finish();
    } else {
        // Failed to delete the account
        Toast.makeText(MainActivityProfilepage.this, "Failed to delete
account. Please try again.", Toast.LENGTH_SHORT).show();
    }
}

});
}

}
}
```

Main Activity Profile Page.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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"
    android:background="#734f96"
    android:gravity="center"
    android:orientation="vertical"
    android:visibility="visible"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/EditProfile"
        android:layout_width="388dp"
        android:layout_height="42dp"
        android:drawableRight="@drawable/baseline_arrow_right_24"
        android:text="Edit Profile"
        android:textColor="#070606"
        android:textSize="25sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toTopOf="@+id/deletebu"
        app:layout_constraintEnd_toEndOf="@+id/Accountsttns"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="@+id/Accountsttns"
        app:layout_constraintTop_toBottomOf="@+id/Accountsttns"
        app:layout_constraintVertical_bias="0.088" />

    <TextView
        android:id="@+id/About"
        android:layout_width="388dp"
        android:layout_height="35dp"
        android:drawableRight="@drawable/baseline_arrow_right_24"
        android:text="About"
        android:textColor="#070606"
```

```
android:textSize="25sp"
android:textStyle="bold"
app:layout_constraintBottom_toTopOf="@+id/deletebu"
app:layout_constraintEnd_toEndOf="@+id/EditProfile"
app:layout_constraintHorizontal_bias="0.0"
app:layout_constraintStart_toStartOf="@+id/EditProfile"
app:layout_constraintTop_toBottomOf="@+id/EditProfile"
app:layout_constraintVertical_bias="0.132" />
```

#### <ProgressBar

```
android:id="@+id/progressBar1"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:visibility="gone"
app:layout_constraintBottom_toTopOf="@+id/deletebu"
app:layout_constraintEnd_toEndOf="parent" />
```

#### <androidx.constraintlayout.widget.ConstraintLayout

```
android:id="@+id/layout1"
android:layout_width="417dp"
android:layout_height="245dp"
android:background="#523B60"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.0"
tools:ignore="VisualLintBounds">
```

#### <ImageView

```
android:id="@+id/Profilepicpage"
android:layout_width="101dp"
android:layout_height="106dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.115"
app:srcCompat="@drawable/ic_launcher_foreground"
android:contentDescription="TODO" />
```

#### <TextView

```
android:id="@+id/Profileppagehone"
android:layout_width="165dp"
android:layout_height="28dp"
android:text="Phone Number"
android:textAlignment="center"
android:textColor="#FFFFFF"
android:textSize="20sp"
android:textStyle="bold"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
```

```
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.774" />
```

<TextView

```
    android:id="@+id/profilepageUsername"
    android:layout_width="162dp"
    android:layout_height="32dp"
    android:text="Username"
    android:textAlignment="center"
    android:textColor="#FFFFFF"
    android:textSize="20sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.572" />
```

<ImageView

```
    android:id="@+id/backbuttontopage"
    android:layout_width="31dp"
    android:layout_height="30dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.041"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.074"
    app:srcCompat="@drawable/baseline_arrow_back_ios_new_24" />
```

</androidx.constraintlayout.widget.ConstraintLayout>

<TextView

```
    android:id="@+id/Accountsttns"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Account Settings"
    android:textColor="#070606"
    android:textSize="25sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toTopOf="@+id/deletebu"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.073"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/layout1"
    app:layout_constraintVertical_bias="0.03" />
```

<Button

```
    android:id="@+id/button2"
    android:layout_width="391dp"
    android:layout_height="63dp"
    android:background="@drawable/bg4"
```



```

        android:drawableLeft="@drawable/baseline_exit_to_app_24"
        android:text="Log out"
        android:textColor="#DD2C00"
        android:textSize="16sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="@+id/deletebu"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="@+id/deletebu"
        app:layout_constraintTop_toBottomOf="@+id/deletebu"
        app:layout_constraintVertical_bias="0.609"
        tools:ignore="VisualLintButtonSize" />

```

<Button

```

        android:id="@+id/deletebu"
        android:layout_width="391dp"
        android:layout_height="63dp"
        android:background="@drawable/bg4"
        android:drawableLeft="@drawable/baseline_no_accounts_24"
        android:text="Delete Account"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/layout1"
        app:layout_constraintVertical_bias="0.781"
        tools:ignore="VisualLintButtonSize" />

```

</androidx.constraintlayout.widget.ConstraintLayout>

## MainActivityProfileEditPage.java

```
package com.example.llorenz;
```

```

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.DocumentSnapshot;
import com.google.firebase.firestore.EventListener;

```

```

import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.FirebaseFirestoreException;

import java.util.Objects;

public class MainActivityEditProfile extends AppCompatActivity {

    FirebaseAuth fAuth;

    FirebaseUser user;
    FirebaseFirestore fstore;

    Button button;
    TextView textViewphone, textViewusername;

    TextView getViewphone, getViewEmail, getViewusername, getViewpassword;
    String userId;

    ImageView back;

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

        //nothing to do with the edit profile
        textViewphone = findViewById(R.id.Profileppagephone);
        textViewusername = findViewById(R.id.profilepageUsername);
        back = findViewById(R.id.backbuttontopage);
        //
        getViewEmail = findViewById(R.id.editTextEmailAddress);
        getViewphone = findViewById(R.id.Phonenumberedit);
        getViewusername = findViewById(R.id.editTextname);
        getViewpassword = findViewById(R.id.editTextPassword);

        fAuth = FirebaseAuth.getInstance();
        fstore = FirebaseFirestore.getInstance();

        userId = Objects.requireNonNull(fAuth.getCurrentUser()).getUid();

        back.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(MainActivityEditProfile.this,
MainActivityProfilepage.class);
                startActivity(intent);
                finish();
            }
        });
    }
}

```

```

    }
});

//show the name and phone number on profile
DocumentReference documentReference =
fstore.collection("users").document(userId);
documentReference.addSnapshotListener(this, new EventListener<DocumentSnapshot>()
{
    @Override
    public void onEvent(@Nullable DocumentSnapshot documentSnapshot, @Nullable
FirebaseFirestoreException error) {
        textViewphone.setText(documentSnapshot.getString("phone"));
        textViewusername.setText(documentSnapshot.getString("uname"));
    }
});

// Edit Profile button functionality
button = findViewById(R.id.editbutton);
button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        // Get the updated information from the EditText fields
        String updatedEmail = getTextEmail.getText().toString().trim();
        String updatedPhone = getViewtextViewphone.getText().toString().trim();
        String updatedUsername = getViewtextViewusername.getText().toString().trim();
        String updatedPassword = getViewtextpassword.getText().toString().trim();

        // Update Firebase Authentication email if it has changed
        if (!updatedEmail.isEmpty()) {
            user.updateEmail(updatedEmail).addOnCompleteListener(new
OnCompleteListener<Void>() {
                @Override
                public void onComplete(@NonNull Task<Void> task) {
                    if (task.isSuccessful()) {
                        Toast.makeText(MainActivityEditProfile.this, "Email
updated successfully.", Toast.LENGTH_SHORT).show();
                    } else {
                        Toast.makeText(MainActivityEditProfile.this, "Failed to
update email.", Toast.LENGTH_SHORT).show();
                    }
                }
            });
        }

        // Update Firestore with the new information
        DocumentReference userRef = fstore.collection("users").document(userId);
        if (!updatedEmail.isEmpty()) {
            userRef.update("email", updatedEmail);
        }
        if (!updatedPhone.isEmpty()) {
            userRef.update("phone", updatedPhone);
        }
    }
});

```



```
android:layout_width="97dp"
android:layout_height="58dp"
android:contentDescription="TODO"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.115"
app:srcCompat="@drawable/ic_launcher_foreground" />
```

#### <TextView

```
android:id="@+id/Profileppagehone"
android:layout_width="165dp"
android:layout_height="28dp"
android:text="Phone Number"
android:textAlignment="center"
android:textColor="#FFFFFF"
android:textSize="20sp"
android:textStyle="bold"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.502"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.955" />
```

#### <TextView

```
android:id="@+id/profilepageUsername"
android:layout_width="162dp"
android:layout_height="32dp"
android:text="Username"
android:textAlignment="center"
android:textColor="#FFFFFF"
android:textSize="20sp"
android:textStyle="bold"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.495"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.646" />
```

#### <ImageView

```
android:id="@+id/backbuttontopage"
android:layout_width="31dp"
android:layout_height="30dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.042"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.121"
app:srcCompat="@drawable/baseline_arrow_back_ios_new_24" />
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

```
<EditText
```

```
    android:id="@+id/editTextname"  
    android:layout_width="281dp"  
    android:layout_height="48dp"  
    android:layout_marginTop="108dp"  
    android:background="@drawable/bg3"  
    android:drawableLeft="@drawable/baseline_person_2_24"  
    android:ems="10"  
    android:hint="Username"  
    android:inputType="text"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.507"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id/layout1" />
```

```
<EditText
```

```
    android:id="@+id/editTextEmailAddress"  
    android:layout_width="281dp"  
    android:layout_height="48dp"  
    android:layout_marginTop="20dp"  
    android:background="@drawable/bg3"  
    android:drawableLeft="@drawable/baseline_mail_24"  
    android:ems="10"  
    android:hint="Email"  
    android:inputType="textEmailAddress"  
    app:layout_constraintEnd_toEndOf="@+id/editTextname"  
    app:layout_constraintHorizontal_bias="0.0"  
    app:layout_constraintStart_toStartOf="@+id/editTextname"  
    app:layout_constraintTop_toBottomOf="@+id/editTextname" />
```

```
<EditText
```

```
    android:id="@+id/Phonenumberedit"  
    android:layout_width="281dp"  
    android:layout_height="48dp"  
    android:layout_marginTop="16dp"  
    android:background="@drawable/bg3"  
    android:drawableLeft="@drawable/baseline_smartphone_24"  
    android:ems="10"  
    android:hint="Phone Number"  
    android:inputType="textEmailAddress"  
    app:layout_constraintEnd_toEndOf="@+id/editTextEmailAddress"  
    app:layout_constraintHorizontal_bias="0.0"  
    app:layout_constraintStart_toStartOf="@+id/editTextEmailAddress"  
    app:layout_constraintTop_toBottomOf="@+id/editTextEmailAddress" />
```

```
<EditText
```

```
    android:id="@+id/editTextPassword"  
    android:layout_width="281dp"  
    android:layout_height="48dp"  
    android:layout_marginTop="20dp"
```

```

        android:background="@drawable/bg3"
        android:drawableLeft="@drawable/baseline_lock_24"
        android:ems="10"
        android:hint="Password"
        android:inputType="textPassword"
        app:layout_constraintEnd_toEndOf="@+id/Phonenumberedit"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="@+id/Phonenumberedit"
        app:layout_constraintTop_toBottomOf="@+id/Phonenumberedit" />

<TextView
    android:id="@+id/text1"
    android:layout_width="269dp"
    android:layout_height="37dp"
    android:text="Edit Profile"
    android:textAlignment="center"
    android:textColor="#000000"
    android:textSize="30sp"
    android:textStyle="bold|italic"
    app:layout_constraintBottom_toTopOf="@+id/editTextname"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/layout1" />

<Button
    android:id="@+id/editbutton"
    android:layout_width="230dp"
    android:layout_height="50dp"
    android:layout_marginTop="44dp"
    android:background="#CAC235"
    android:text="EDIT PROFILE"
    android:textColor="#0D47A1"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="@+id/editTextPassword"
    app:layout_constraintHorizontal_bias="0.47"
    app:layout_constraintStart_toStartOf="@+id/editTextPassword"
    app:layout_constraintTop_toBottomOf="@+id/editTextPassword"
    app:layout_constraintVertical_bias="0.0" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

## MainActivityAbout.java

```

package com.example.llorenz;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageView;

```

```

public class MainActivityAbout extends AppCompatActivity {

    ImageView back1;

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

        back1 = findViewById(R.id.back12);

        back1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(MainActivityAbout.this,
MainActivityProfilepage.class);
                startActivity(intent);
                finish();
            }
        });
    }
}

```

### Main Activity About.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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"
    android:background="#734f96"
    tools:context=".MainActivityAbout">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:id="@+id/constraintLayout3"
        android:layout_width="410dp"
        android:layout_height="58dp"
        android:background="#523B60"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"

```



```

app:layout_constraintVertical_bias="0.0">

<TextView
    android:id="@+id/About1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="About"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<ImageView
    android:id="@+id/back12"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.041"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:srcCompat="@drawable/baseline_arrow_back_ios_new_24" />
</androidx.constraintlayout.widget.ConstraintLayout>

<TextView
    android:id="@+id/textView8"
    android:layout_width="390dp"
    android:layout_height="654dp"
    android:text="Introducing MobileNex, your go-to solution for seamless SIM card
loading and management on your mobile device. Say goodbye to the hassles of traditional
top-up methods and experience the ultimate convenience of recharging your SIM card
anytime, anywhere. MobileNex is a user-friendly mobile application designed to make the
process of loading your SIM card quick, easy, and efficient. Whether you're at home, on
the go, or traveling internationally, our app ensures that you stay connected with just a
few taps. Make the switch to MobileNex today and redefine the way you manage your mobile
connectivity. Download the app now and experience the future of SIM card loading - fast,
secure, and incredibly convenient. Stay connected effortlessly with MobileNex."
    android:textAlignment="center"
    android:textSize="20sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.476"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/constraintLayout3"
    app:layout_constraintVertical_bias="0.684" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

### III. User Documentation

#### A. End-User Documentation

MobileNex provides solutions to users' problems when it comes to managing and loading their respective SIM cards. Here are some of them:

1. **Convenience:**
  - a. Users can easily manage their SIM cards directly from their own phones. They can check, recharge, manage their balance, and other services that their SIM cards offer. They don't need to visit a physical store to do these anymore.
2. **Balance and Data Monitoring:**
  - a. MobileNex allows their users to monitor their data usage, load balance, and validity period. This enables them to stay informed about their usage and to manage their usage to prevent sudden disconnection and possible deactivation of their SIM cards.
3. **Recharge Flexibility:**
  - a. This offers them to recharge or top-up their SIM cards anytime and anywhere especially when they're in need of connectivity.
4. **Multiple Services in One Place:**
  - a. Users can access a variety of services that their SIM cards offer. This includes activation of new plans, offers, data monitoring, and availing of promos within the application.
5. **Notifications and Alerts:**
  - a. The application can provide timely alerts and notifications for low balance, available promos, or updates related to their SIM card services. This makes the users informed immediately of the status of their SIM card balance and load.

#### B. System Admin Documentation

Here are the standard system administrator's documents:

- **Functional description**
  - As a system admin, their responsibilities include managing backend infrastructure, overseeing server-side operations, and ensuring the security and performance of MobileNex. Collaborate with development teams, monitor system analytics, and implement improvements for a reliable and efficient application. Document configurations, procedures, and updates for future reference and troubleshooting. The role is pivotal in maintaining a seamless and secure user experience.
- **System admin guide –**
  - MobileNex's System Administrator's role involves monitoring various system behaviors in different environments and ensuring seamless interactions with external systems. In cases of malfunctions, a quick troubleshooting guide can be followed to identify, communicate, and collaborate for swift issue resolution.
  - **System Behaviors in Different Environments and Interactions with Other Systems:**
    - **Production Environment:**

- Behavior: High traffic, live user data.
- Action: Regularly monitor server loads and optimize for peak times.
- **Testing Environment:**
- Behavior: Simulated user interactions.
- Action: Supervise testing processes, ensuring accurate simulations.
- **Integration with Payment Gateways:**
- Behavior: Financial transactions.
- Action: Regularly check payment gateway integrations, monitor transaction logs.
- **Telecom APIs Interaction:**
- Behavior: SIM card activations, balance inquiries.
- Action: Monitor and troubleshoot any discrepancies in API interactions.
- **Dealing with Malfunction Situations:**
  - **Server Downtime:**
  - Symptoms: Unresponsive application.
  - Action: Execute backup servers, investigate root cause promptly.
  - **Database Issues:**
  - Symptoms: Data corruption, slow queries.
  - Action: Regularly optimize databases, perform routine health checks.
  - **Security Breach:**
  - Symptoms: Unauthorized access.
  - Action: Implement security patches, conduct a thorough investigation.
  - **Integration Failures:**
  - Symptoms: Failed transactions, communication errors.
  - Action: Collaborate with relevant service providers, troubleshoot integration points.
  - **User Data Privacy Concerns:**
  - Symptoms: Privacy policy violations.
  - Action: Ensure GDPR compliance, address user concerns promptly.
- **Quick Troubleshooting Guide:**
  - **Identify the Issue:**
  - Use monitoring tools to pinpoint the source of the problem.
  - **Consult Documentation:**
  - Refer to system configurations and procedures documented for quick reference.
  - **Communication:**
  - Promptly inform relevant stakeholders of the issue and the steps being taken.

- **Collaborate with Development Teams:**
  - Work closely with developers to resolve technical issues efficiently.