**GirisYap.java**

package com.example.hayatsinyalim;  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.webkit.WebView;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.LinearLayout;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import com.google.firebase.FirebaseException;  
import com.google.firebase.auth.PhoneAuthCredential;  
import com.google.firebase.auth.PhoneAuthProvider;  
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.ValueEventListener;  
  
import java.util.ArrayList;  
import java.util.List;  
import java.util.concurrent.TimeUnit;  
public class GirisYap extends AppCompatActivity {  
 List<String> telefonNumaralariListesi = new ArrayList<>();  
 List<String> telefonNumaralariListesi2 = new ArrayList<>();  
 List<String> uidlist2 = new ArrayList<>();  
 Button loginbutton;  
 EditText edttel,edtsif;  
 TextView hata;  
 String adSoyad;  
 WebView WebView;  
 LinearLayout kayit,gif;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_giris\_yap*);  
 WebView = findViewById(R.id.*webView*);  
 loginbutton = findViewById(R.id.*loginButton*);  
 edttel = findViewById(R.id.*phones*);  
 edtsif = findViewById(R.id.*Sifre*);  
 hata = findViewById(R.id.*HataTxt*);  
 kayit = findViewById(R.id.*kayıts*);  
 gif = findViewById(R.id.*LoadGif*);  
 WebView = findViewById(R.id.*webView*);  
 WebView.getSettings().setLoadsImagesAutomatically(true);  
 WebView.getSettings().setJavaScriptEnabled(true);  
 WebView.setBackgroundColor(0x00000000); // Şeffaf arka plan  
 DatabaseReference telefonNumaralariRef = FirebaseDatabase.*getInstance*().getReference().child("TelefonNumaralari");  
 telefonNumaralariRef.addListenerForSingleValueEvent(new ValueEventListener() {  
 @Override  
 public void onDataChange(DataSnapshot dataSnapshot) {  
 List<String> uidListesi = new ArrayList<>();  
 for (DataSnapshot snapshot : dataSnapshot.getChildren()) {  
 String uid = snapshot.getKey();  
 String telefonNo = snapshot.getValue(String.class);  
 if (uid != null && telefonNo != null) {  
 uidListesi.add(uid);  
 telefonNumaralariListesi.add(telefonNo);  
 }  
 }  
 for (int i = 0; i < uidListesi.size(); i++) {  
 String uid = uidListesi.get(i);  
 String telefonNo = telefonNumaralariListesi.get(i);  
 telefonNumaralariListesi2.add(telefonNo);  
 uidlist2.add(uid);  
 }  
 }  
 @Override  
 public void onCancelled(DatabaseError databaseError) {  
  
 }  
 });  
 loginbutton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 String phoneNumber = "+90"+edttel.getText().toString();  
 String aranan = phoneNumber;  
 boolean bulundu = false;  
 for (String eleman : telefonNumaralariListesi2) {  
 if (eleman.equals(aranan)) {  
 bulundu = true;  
 break;  
 }  
 }  
 if (bulundu)  
 {  
 FirebaseDatabase database = FirebaseDatabase.*getInstance*();  
 DatabaseReference reference = database.getReference("users/" + uidlist2.get(telefonNumaralariListesi2.indexOf(phoneNumber)));  
 reference.addListenerForSingleValueEvent(new ValueEventListener() {  
 @Override  
 public void onDataChange(DataSnapshot dataSnapshot) {  
 if (dataSnapshot.exists()) {  
 adSoyad = dataSnapshot.child("AdSoyad").getValue(String.class);  
 String sifre = dataSnapshot.child("Sifre").getValue(String.class);  
 String telefonNo = dataSnapshot.child("TelefonNo").getValue(String.class);  
 String TelNo2 = telefonNo;  
 if (TelNo2.equals("+90"+edttel.getText().toString()) && sifre.equals(edtsif.getText().toString())) {  
 long timeout = 120;  
 TimeUnit timeUnit = TimeUnit.*SECONDS*;  
 PhoneAuthProvider.*getInstance*().verifyPhoneNumber(  
 phoneNumber,  
 timeout,  
 timeUnit,  
 GirisYap.this,  
 callbacks  
 );  
 kayit.setVisibility(View.*GONE*);  
 gif.setVisibility(View.*VISIBLE*);  
 String html = "<html><body style='margin:0;padding:0;'><img src='file:///android\_res/drawable/load.gif' width='100%' height='100%'></body></html>";  
 WebView.loadDataWithBaseURL(null, html, "text/html", "utf-8", null);  
 } else {  
 hata.setText("Hata:\nTelefon Numarası Veya Şifre Hatalı!");  
 }  
 }  
 else  
 {  
 hata.setText("Hata:\nKullanıcı Bulunamadı Lütfen Kayıt Olunuz!");  
 }  
 }  
 @Override  
 public void onCancelled(DatabaseError databaseError) {  
  
 }  
 });  
 }  
 else  
 {  
 hata.setText("Hata:\nKullanıcı Bulunamadı Lütfen Kayıt Olunuz!");  
 }  
 }  
 });  
 }  
 PhoneAuthProvider.OnVerificationStateChangedCallbacks callbacks =  
 new PhoneAuthProvider.OnVerificationStateChangedCallbacks() {  
 @Override  
 public void onVerificationCompleted(@NonNull PhoneAuthCredential phoneAuthCredential) {  
 Toast.*makeText*(GirisYap.this, "Doğrulama Kodu Başarıyla Gönderildi", Toast.*LENGTH\_SHORT*).show();  
 }  
 @Override  
 public void onVerificationFailed(@NonNull FirebaseException e) {  
 Toast.*makeText*(GirisYap.this, "Doğrulama Kodu Gönderilirken Bir Hata Oluştu Lütfen Daha Sonra Tekrar Deneyiniz", Toast.*LENGTH\_SHORT*).show();  
 }  
 @Override  
 public void onCodeSent(@NonNull String verificationId, @NonNull PhoneAuthProvider.ForceResendingToken forceResendingToken) {  
 Intent intent = new Intent(GirisYap.this, TelOnay.class);  
 intent.putExtra("verificationId", verificationId);  
 intent.putExtra("Ad", adSoyad.toString());  
 intent.putExtra("Secenek", "Giris");  
 startActivity(intent);  
 }  
 };  
}

**Açıklama ;**

İlk olarak, kullanıcının girmiş olduğu telefon numarası ve şifreyi kontrol eder. Eğer girilen telefon numarası Firebase'deki kayıtlı telefon numaralarından biriyle eşleşiyorsa ve aynı zamanda şifre de doğruysa, doğrulama işlemi için Firebase'den telefon numarasına doğrulama kodu gönderir. Bunun ardından kullanıcıya bir yükleme animasyonu gösterir.

Eğer girilen telefon numarası kayıtlı değilse veya şifre yanlışsa, kullanıcıya bir hata mesajı gösterir.

**PhoneAuthProvider.OnVerificationStateChangedCallbacks** aracılığıyla doğrulama işleminin sonuçlarını dinler. Doğrulama işlemi başarılı olduğunda, kullanıcıya bir bildirim gösterir ve doğrulama kodunu gönderir. Kod gönderme işlemi başarısız olursa da kullanıcıya bir hata mesajı gösterir.

Son olarak, doğrulama kodu gönderildiğinde, **onCodeSent** işlevi, bir sonraki aşama olan telefon numarası doğrulama aktivitesine geçişi sağlar ve bu aktiviteye gerekli verileri (doğrulama ID'si, kullanıcının adı vb.) gönderir.