**AnaSayfa.java**

package com.example.hayatsinyalim;  
import android.Manifest;  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.content.pm.PackageManager;  
import android.location.Location;  
import android.location.LocationListener;  
import android.location.LocationManager;  
import android.net.Uri;  
import android.os.Bundle;  
import android.preference.PreferenceManager;  
import android.provider.Settings;  
import android.util.Log;  
import android.view.View;  
import android.widget.ImageView;  
import android.widget.TextView;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
import androidx.core.content.ContextCompat;  
  
import com.google.android.gms.tasks.OnCompleteListener;  
import com.google.android.gms.tasks.Task;  
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.ValueEventListener;  
import com.google.firebase.messaging.FirebaseMessaging;  
  
import org.osmdroid.config.Configuration;  
import org.osmdroid.tileprovider.tilesource.TileSourceFactory;  
import org.osmdroid.util.GeoPoint;  
import org.osmdroid.views.MapView;  
import org.osmdroid.views.overlay.Marker;  
  
import android.content.Context;  
import android.widget.Toast;  
  
import java.util.Date;  
import java.util.Timer;  
import java.util.TimerTask;  
public class AnaSayfa extends AppCompatActivity {  
 private MapView mapView;  
 TextView ad, someTextView;  
 private static final int *PERMISSION\_REQUEST\_CODE* = 500;  
 Marker marker;  
 private Timer timer;  
 String token;  
 ImageView imgKonum;  
 private LocationManager locationManager;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_ana\_sayfa*);  
  
 mapView = findViewById(R.id.*mapView*);  
 ImageView BtnMainMenu, BtnChat, BtnUser;  
 BtnMainMenu = findViewById(R.id.*mainmenubtn*);  
 BtnChat = findViewById(R.id.*chatBtn*);  
 BtnUser = findViewById(R.id.*userBtn*);  
 ad = findViewById(R.id.*ad*);  
  
 timer = new Timer();  
 marker = new Marker(mapView);  
  
 imgKonum = findViewById(R.id.*circleImageView*);  
 locationManager = (LocationManager) getSystemService(Context.*LOCATION\_SERVICE*);  
 mapView.setTileSource(TileSourceFactory.*MAPNIK*);  
 mapView.setBuiltInZoomControls(true);  
 mapView.setMultiTouchControls(true);  
 mapView.setBuiltInZoomControls(false);  
 boolean isLocationEnabled = locationManager.isProviderEnabled(LocationManager.*GPS\_PROVIDER*) ||  
 locationManager.isProviderEnabled(LocationManager.*NETWORK\_PROVIDER*);  
 if (isLocationEnabled) {  
 imgKonum.setImageResource(R.drawable.*acikkonum*);  
 } else {  
 imgKonum.setImageResource(R.drawable.*kapalikonum*);  
 }  
 FirebaseAuth mAuth = FirebaseAuth.*getInstance*();  
 String userID = mAuth.getUid();  
 FirebaseMessaging.*getInstance*().getToken()  
 .addOnCompleteListener(new OnCompleteListener<String>() {  
 @Override  
 public void onComplete(@NonNull Task<String> task) {  
 if (task.isSuccessful()) {  
 // Token başarıyla alındı  
 token = task.getResult();  
 DatabaseReference userRef = FirebaseDatabase.*getInstance*().getReference().child("users").child(userID);  
 userRef.child("FcmToken").setValue(token);  
 } else {  
 }  
 }  
 });  
 FirebaseDatabase database = FirebaseDatabase.*getInstance*();  
 DatabaseReference ref = database.getReference("users/" + mAuth.getUid());  
 ref.addListenerForSingleValueEvent(new ValueEventListener() {  
 @Override  
 public void onDataChange(DataSnapshot dataSnapshot) {  
 if (dataSnapshot.exists()) {  
 String adSoyad = dataSnapshot.child("AdSoyad").getValue(String.class);  
 String telefonNo = dataSnapshot.child("TelefonNo").getValue(String.class);  
 ad.setText(adSoyad);  
 } else {  
  
 }  
 }  
  
 @Override  
 public void onCancelled(DatabaseError databaseError) {  
  
 }  
 });  
 GeoPoint focusPoint = new GeoPoint(39.03682309172971, 35.555307561350894);  
 mapView.getController().animateTo(focusPoint);  
 mapView.getController().zoomTo(6);  
 Configuration.*getInstance*().load(this, PreferenceManager.*getDefaultSharedPreferences*(this));  
 BtnMainMenu.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(AnaSayfa.this, MainMenu.class);  
 startActivity(intent);  
 overridePendingTransition(0, 0);  
 finish();  
 }  
 });  
 BtnChat.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(AnaSayfa.this, Chat.class);  
 intent.putExtra("Durumu", "Kull");  
 startActivity(intent);  
 overridePendingTransition(0, 0);  
 finish();  
 }  
 });  
 BtnUser.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(AnaSayfa.this, User.class);  
 startActivity(intent);  
 overridePendingTransition(0, 0);  
 finish();  
 }  
 });  
 imgKonum.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 LocationManager locationManager = (LocationManager) getSystemService(Context.*LOCATION\_SERVICE*);  
 if (!locationManager.isProviderEnabled(LocationManager.*GPS\_PROVIDER*)) {  
 // Eğer GPS kapalıysa, kullanıcıyı konum ayarlarına yönlendir  
 Intent intent = new Intent(Settings.*ACTION\_LOCATION\_SOURCE\_SETTINGS*);  
 startActivity(intent);  
 } else {  
 Toast.*makeText*(AnaSayfa.this, "Zaten Konumunuz Açık!", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
 timer.schedule(new TimerTask() {  
 @Override  
 public void run() {  
 // Burada yapılacak işlemler  
 // Örneğin:  
 runOnUiThread(new Runnable() {  
 @Override  
 public void run() {  
 getLocation();  
 }  
 });  
 }  
 }, 0, 2000); // İlk çalışma zamanı (0ms) ve tekrarlama süresi (1000ms) belirtildi  
 }  
 private void getLocation() {  
 mapView.getOverlays().remove(marker);  
 if (ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED* &&  
 ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_COARSE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED*) {  
 ActivityCompat.*requestPermissions*(this, new String[]{Manifest.permission.*ACCESS\_FINE\_LOCATION*}, 1);  
 return;  
 }  
 locationManager.requestSingleUpdate(LocationManager.*GPS\_PROVIDER*, new LocationListener() {  
 @Override  
 public void onLocationChanged(Location location) {  
 double latitude = location.getLatitude();  
 double longitude = location.getLongitude();  
 FirebaseAuth mAuth = FirebaseAuth.*getInstance*();  
 FirebaseDatabase database = FirebaseDatabase.*getInstance*();  
 DatabaseReference usersRef = database.getReference("users");  
 String userId = mAuth.getUid(); // Kullanıcının UID'si  
 Date simdikiZaman = new Date();  
 String zaman = simdikiZaman.toString();  
 usersRef.child(userId).child("SonKonum").setValue(zaman + " : " + latitude +","+longitude);  
 DatabaseReference konums = database.getReference("Konumlar");  
 konums.child(userId).setValue(zaman + " : " + latitude + "," + longitude);  
 GeoPoint markerLocation = new GeoPoint(latitude , longitude);  
 marker.setPosition(markerLocation);  
 marker.setAnchor(Marker.*ANCHOR\_CENTER*, Marker.*ANCHOR\_BOTTOM*);  
 mapView.getOverlays().add(marker);  
 }  
 @Override  
 public void onStatusChanged(String provider, int status, Bundle extras) {}  
 @Override  
 public void onProviderEnabled(String provider) {}  
 @Override  
 public void onProviderDisabled(String provider) {}  
 }, null);  
 }  
 private boolean checkLocationPermission() {  
 if (ContextCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED*) {  
 // İzin verilmedi, kullanıcıdan izin iste  
 ActivityCompat.*requestPermissions*(this, new String[]{Manifest.permission.*ACCESS\_FINE\_LOCATION*}, *PERMISSION\_REQUEST\_CODE*);  
 return false;  
 }  
 return true;  
 }  
 public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {  
 super.onRequestPermissionsResult(requestCode, permissions, grantResults);  
 if (requestCode == *PERMISSION\_REQUEST\_CODE*) {  
 if (grantResults.length > 0 && grantResults[0] == PackageManager.*PERMISSION\_GRANTED*) {  
  
 } else {  
 showPermissionRationale();  
 }  
 }  
 }  
 private void showPermissionRationale() {  
 AlertDialog.Builder builder = new AlertDialog.Builder(this);  
 builder.setTitle("Konum İzni");  
 builder.setMessage("Uygulama konum bilgisi almak için izin istiyor. Devam etmek istiyor musunuz? Daha fazla bilgi için [buraya tıklayın].");  
 builder.setPositiveButton("Evet", (dialog, which) -> {  
 ActivityCompat.*requestPermissions*(this,  
 new String[]{Manifest.permission.*ACCESS\_FINE\_LOCATION*},  
 *PERMISSION\_REQUEST\_CODE*);  
 });  
 builder.setNegativeButton("Hayır", (dialog, which) -> {  
 showPermissionRationale();  
 });  
 AlertDialog alertDialog = builder.create();  
 alertDialog.setOnShowListener(dialog -> {  
 alertDialog.getButton(DialogInterface.*BUTTON\_NEUTRAL*).setOnClickListener(view -> {  
 Uri uri = Uri.*parse*("https://hayatsinyalim.tomtal.com.tr/");  
 Intent intent = new Intent(Intent.*ACTION\_VIEW*, uri);  
 startActivity(intent);  
 });  
 });  
 alertDialog.show();  
 }  
 @Override  
 protected void onResume() {  
 super.onResume();  
 boolean isLocationEnabled = locationManager.isProviderEnabled(LocationManager.*GPS\_PROVIDER*) ||  
 locationManager.isProviderEnabled(LocationManager.*NETWORK\_PROVIDER*);  
 if (isLocationEnabled) {  
 imgKonum.setImageResource(R.drawable.*acikkonum*);  
 } else {  
 imgKonum.setImageResource(R.drawable.*kapalikonum*);  
 }  
 }  
}

**Açıklama ;**

1. **OnCreate Metodu:**
   * **onCreate** metodu, bir aktivitenin oluşturulduğu yerdir. Bu durumda, **AnaSayfa** sınıfı bir **AppCompatActivity** ile genişletilmiş ve bu metot içinde çalışıyor.
   * **setContentView** metoduyla **activity\_ana\_sayfa** adlı bir layout tanımlanıyor.
   * Harita (**MapView**) ve diğer görünümler (butonlar, metin alanları, resimler) **findViewById** metoduyla belirli ID'lere sahip layout öğelerine bağlanıyor.
   * **Timer** sınıfı oluşturuluyor, bununla zamanlayıcı işlemleri yapılacak.
2. **Kullanıcı ve Firebase İşlemleri:**
   * **FirebaseAuth** ve **FirebaseDatabase** sınıfları kullanılarak Firebase'e erişim sağlanıyor.
   * Kullanıcının kimlik bilgileri alınıyor ve kullanıcının token'ı alınıp veritabanına kaydediliyor.
   * Kullanıcı bilgileri (**AdSoyad**, **TelefonNo**) veritabanından çekilip ekranda gösteriliyor.
3. **Harita ve Konum İşlemleri:**
   * Harita için yapılandırma yapılıyor (**setTileSource**, **setBuiltInZoomControls** gibi).
   * Konum izni kontrol ediliyor, eğer izin yoksa kullanıcıdan izin isteniyor.
   * **LocationManager** ile konum izleme işlemi gerçekleştiriliyor.
   * **getLocation** metodu, konum güncellemelerini alır, bu konumu Firebase veritabanına kaydeder ve haritada işaretler.
4. **İzin Yönetimi:**
   * Kullanıcıya konum izni için istekte bulunuluyor. İzin verilmediği durumlarda kullanıcıya açıklama gösteriliyor.
5. **Arayüz Kontrolleri:**
   * Butonlara tıklama durumunda ilgili aktivitelere yönlendirme işlemleri gerçekleştiriliyor.
6. **Toast ve Diğer Geri Bildirimler:**
   * Kullanıcıya konum izniyle ilgili geri bildirimler (**Toast** mesajları) gösteriliyor.