Imagen que contiene exterior, cielo, nieve, agua

Descripción generada con confianza muy alta

**ESCUELA SUPERIOR POLITÉNICA DEL LITORAL**

**PROYECTO DE APLICACIONES MÓVILES Y SERVICIOS TELEMÁTICOS**

**MANUAL TÉCNICO**

TRAKING DE MOCHILA

**MG. ADRIANA COLLAGUAZO**

**GRUPO #3**

BODERO CASTRO JULIO

DODDY CASTILLO

**PARALELO 1**

2T – 2019

**Manual Implementación Técnica del Proyecto**

El siguiente manual técnico expone los detalles presentes en el proyecto realizado. Este proyecto tiene La finalidad de obtener la ubicación de una mochila.

**Introducción del sistema implementado**

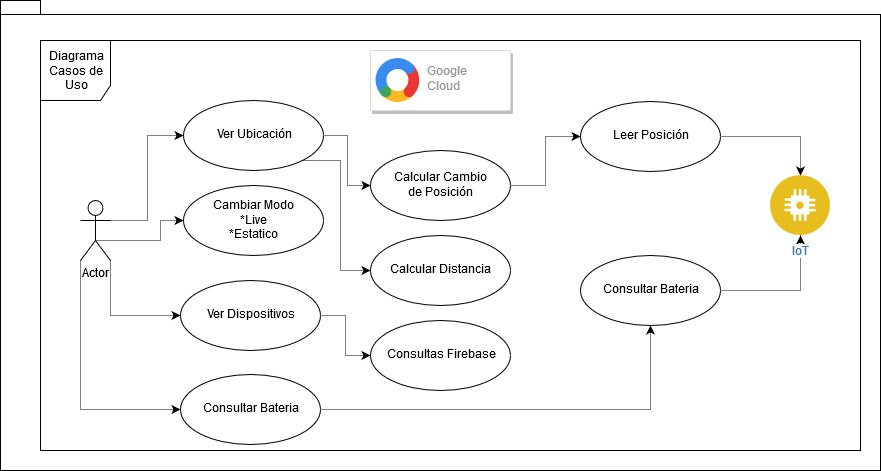
Este sistema nos ayuda a comunicar la posición de una maleta en comparación con el usuario, la aplicación le indica al usuario cual fue la última posición de su maleta y donde se encuentra.

1. **Recursos**

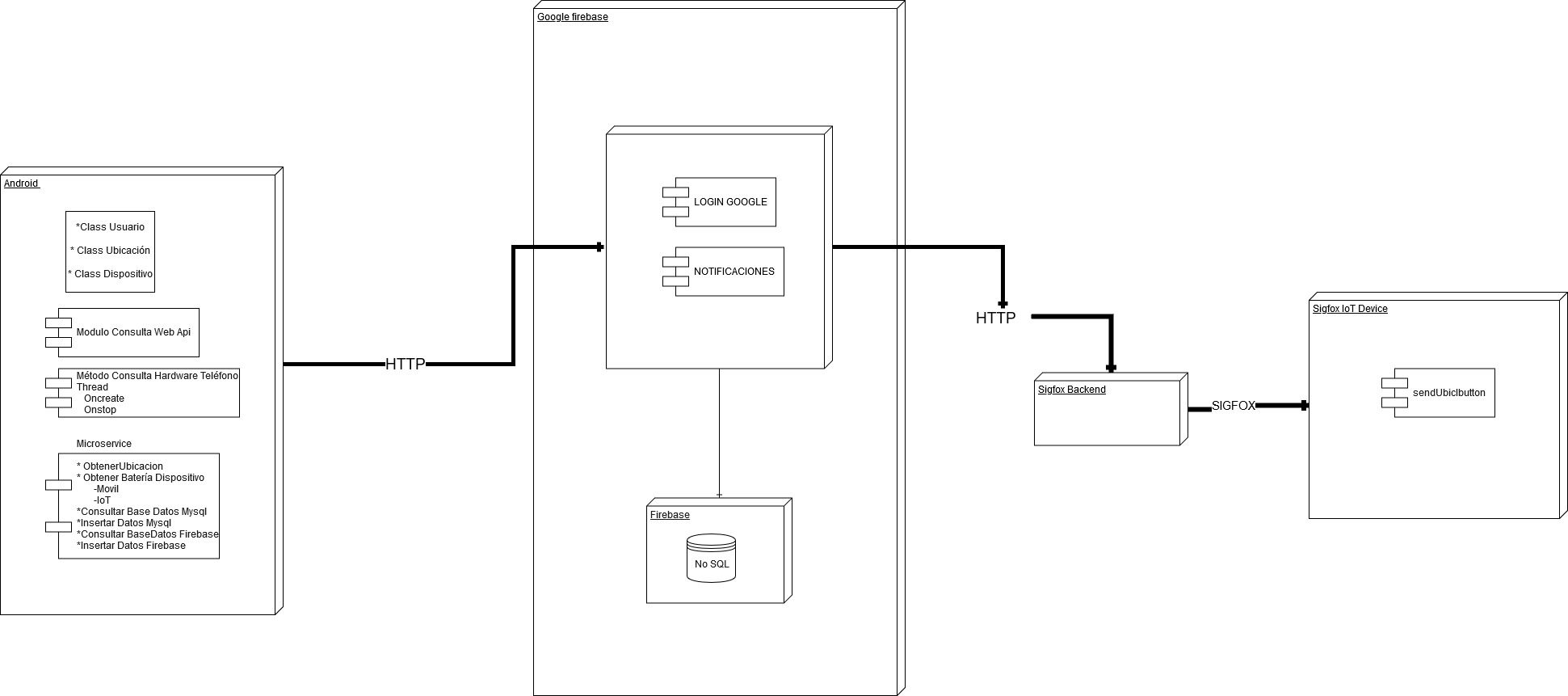
**Para realizar la implementación del proyecto fueron necesarias las siguientes herramientas:**

|  |  |
| --- | --- |
| Hardware | Software |
| Módulo Thinxtra ($109.25) | IDE Arduino |
| 2 PC | Android Studio |
| 30 jumpers ($ 2) | Base de datos Firebase |
| 1 placa Arduino ($20) | Herramienta colaborativa Asana |
| Ibuton ($50 | Plaforma SigFox |
|  |  |
|  |  |
|  |  |

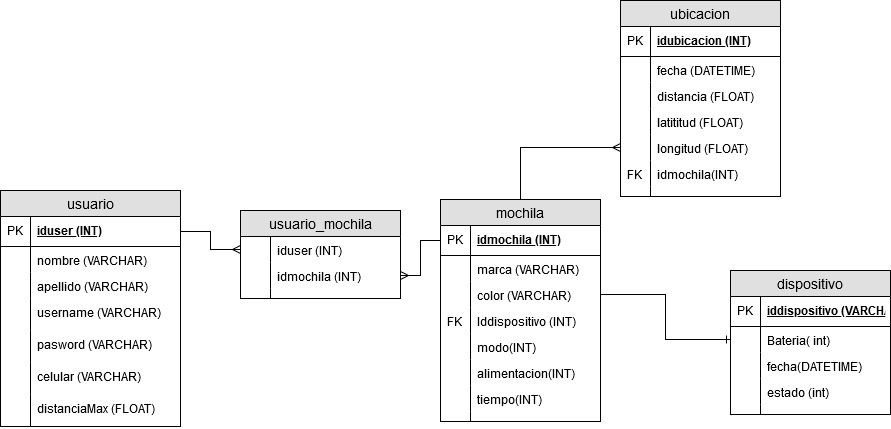
1. **Diagramas Caso Uso**

En este sistema se describe cual es la interacción del usuario y los diferentes sistemas a los que se realizan requerimientos

1. **Diagrama de despliegue**

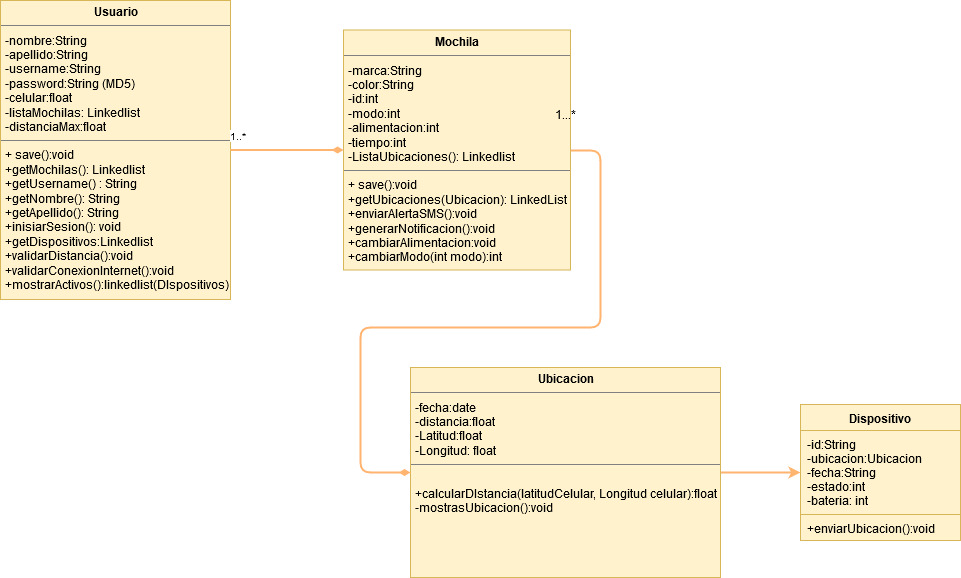
El diagrama de despliegue mostrado a continuación explica acerca de la interacción entre los diferentes módulos que forman parte del proyectos se observa la interaccion que tiene diferentes funciones de la aplicación con la plataforma de Google Cloud para la interacción del Login y notificaciones además se observa como nuestro dispositivo de Sigfox interactúa con el backend y este a su vez con Firebase de Google.

1. **Modelo de entidad-relación de la base de datos.**

El diagrama a continuación muestra las relaciones entre las diferentes entidades de las tablas de la base de datos ,Además, en el diagrama se puede observar los campos que poseen cada tabla, así como su tamaño y el tipo de variable con el que almacenaran los datos.

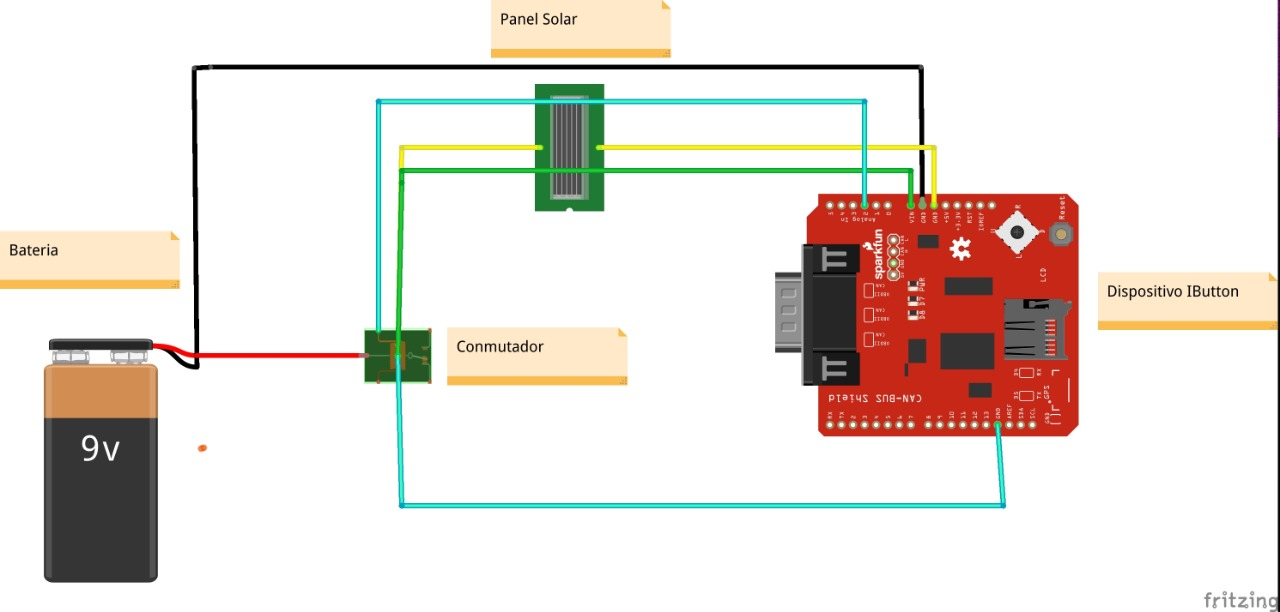
1. **Diagrama de Clase**

Se puede apreciar todas las clases necesarias con sus funciones para la creación de la aplicación



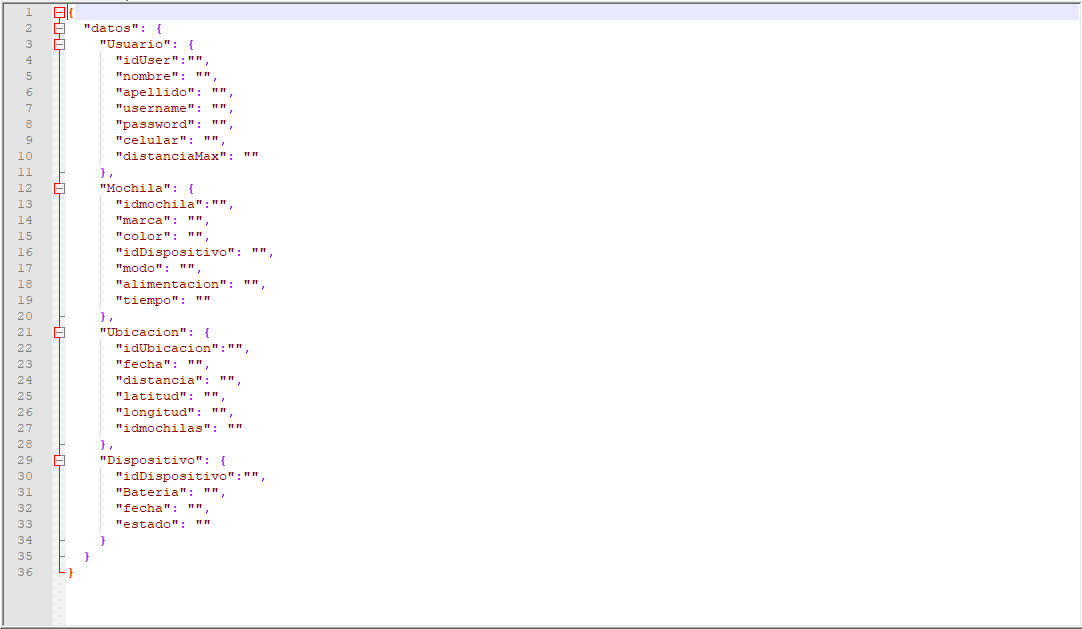
1. **Diagrama de circuito**

Se utilizo un dispositivo Ibutton para el envió de posición, un relé o conmutador para el cambio de energía entre solar y batería.



1. **Bases de datos No Relacional**

En este grafico se indica todas las variables que se utilizara y almacenara en firebase para la interacción con la aplicación.



1. **Explicación del código fuente Arduino desarrollado con los comentarios correspondientes**

El código en Arduino implementado recibe señales de un modulo bluetooh para la conmutación entre energía solar o de baterías.

// Incluimos librería

#include <Thread.h>

#include <SoftwareSerial.h>

SoftwareSerial Blue(2, 3);// 3-RXD ; 2-TXD

#define ANALOGCARGA 0

// Variables

int analogValor = 0;

float voltaje = 0;

//Controlador de carga

long int data;

int carga1 = 13;

int carga2= 12;

long int password1 = 79;// solar

long int password2 = 92; // usb

long int password3 =91; // apagas

char state = 0;

Thread thrBlink2 = Thread();

Thread thrBlink3 = Thread();

void setup() {

Serial.begin(9600);

//Controlador de Carga

pinMode(carga1, OUTPUT);

pinMode(carga2, OUTPUT);

digitalWrite(carga1, LOW);

digitalWrite(carga2, LOW);

Blue.begin(9600);

//Bluetooth

thrBlink2.enabled = true;

thrBlink2.setInterval(500);

thrBlink2.onRun(codigo\_Bluetooth);

//Carga

thrBlink3.enabled = true;

thrBlink3.setInterval(500);

thrBlink3.onRun(medir\_carga);

}

void blink() {

//definir tarea

}

void loop() {

// Esperamos 5 segundos entre medidas

thrBlink2.run();

thrBlink3.run();

}

void codigo\_Bluetooth(){

if(Blue.available()==0){

return;

}

if(Blue.available()>0)

{

data = Blue.parseInt();

}

delay(1);

//Serial.print(data);

if (data == password1)

{

digitalWrite(carga1,HIGH);

digitalWrite(carga2, LOW);

Serial.println("Solar ON ");

}

if( data == password2)

{

digitalWrite(carga1,LOW);

digitalWrite(carga2, HIGH);

Serial.println("USB ON");

}

if( data == password3)

{

digitalWrite(carga1,LOW);

digitalWrite(carga2, LOW);

Serial.println("OFF All");

}

}

void medir\_carga(){

// Leemos valor de la entrada analógica

analogValor = analogRead(ANALOGCARGA);

// Obtenemos el voltaje

voltaje = 0.0029 \* analogValor;

sigfoxSend(&voltaje,sizeof(voltaje));

Serial.print("Voltaje: ");

Serial.println(voltaje);

delay(1000);

}

bool sigfoxSend(const void\* data, uint8\_t len) {

uint8\_t\* bytes = (uint8\_t\*)data;

Serial.println("AT$RC");

Serial.print("AT$SF=");

for(uint8\_t i = len-1; i < len; --i) {

if (bytes[i] < 16) {Serial.print("0");}

Serial.print(bytes[i], HEX);

}

Serial.print('\r');

}

Este código realiza el control de conmutadores con un modulo bluetooh

#include <SoftwareSerial.h>

SoftwareSerial Blue(2, 3);// 3-RXD ; 2-TXD

int relay=7;

long int data;

long int arreglo[3]={92,79,91};

char state=0;

void setup() {

// put your setup code here, to run once:

pinMode(relay,OUTPUT);

digitalWrite(relay,LOW);

Serial.begin(9600);

Blue.begin(9600);

}

void loop() {

while(Blue.available()==0) ;

if(Blue.available()>0)

{

data = Blue.parseInt();

}

delay(400);

int delimitador=0;

for(int j=0;j<3;j++)

{

if(data==arreglo[j])

{

delimitador++;

}

}

if(delimitador==0)

{

long int i=0;

while(i<data)

{

digitalWrite(relay,HIGH);

Serial.println("Relay Accionado");

delay(1000);

digitalWrite(relay,LOW);

Serial.println("Relay No Accionado");

delay(1000);

i+=2;

}

}

}

1. **Explicación del código Java desarrollado en Android Studio con los comentarios correspondientes**

Esta clase nos realiza las consultas a base de datos y realiza el login a firebase

**LoginActivity.java**

|  |
| --- |
| public class LoginActivity extends AppCompatActivity { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private EditText email; |
|  |

|  |
| --- |
| private EditText pass; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| static final int GOOGLE\_SIGN\_IN = 123; |
|  |

|  |
| --- |
| private GoogleSignInClient mGoogleSignInClient; |
|  |

|  |
| --- |
| private FirebaseAuth mAuth; |
|  |

|  |
| --- |
| private DatabaseReference db\_reference; |
|  |

|  |
| --- |
| private FirebaseUser currentUser; |
|  |

|  |
| --- |
| private static final String TAG = "EmailPassword"; |
|  |

|  |
| --- |
| private HashMap<String, String> info\_user; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| protected void onCreate(Bundle savedInstanceState) { |
|  |

|  |
| --- |
| super.onCreate(savedInstanceState); |
|  |

|  |
| --- |
| setContentView(R.layout.activity\_login); |
|  |

|  |
| --- |
| email = findViewById(R.id.text\_email); |
|  |

|  |
| --- |
| pass = findViewById(R.id.text\_pass); |
|  |

|  |
| --- |
| info\_user = new HashMap<>(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| if (Variables.HAY\_INTERNET) { |
|  |

|  |
| --- |
| this.mAuth =FirebaseAuth.getInstance(); |
|  |

|  |
| --- |
| this.db\_reference = FirebaseDatabase.getInstance().getReference().child("Aplicacion"); |
|  |

|  |
| --- |
| this.currentUser = mAuth.getCurrentUser(); |
|  |

|  |
| --- |
| GoogleSignInOptions gso = new GoogleSignInOptions.Builder(GoogleSignInOptions.DEFAULT\_SIGN\_IN) |
|  |

|  |
| --- |
| .requestIdToken(getString(R.string.default\_web\_client\_id)) |
|  |

|  |
| --- |
| .requestEmail() |
|  |

|  |
| --- |
| .build(); |
|  |

|  |
| --- |
| mGoogleSignInClient = GoogleSignIn.getClient(this, gso); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| Intent intent = getIntent(); |
|  |

|  |
| --- |
| String msg = intent.getStringExtra("msg"); |
|  |

|  |
| --- |
| if (msg != null) { |
|  |

|  |
| --- |
| if (msg.equals("cerrarSesion")) { |
|  |

|  |
| --- |
| cerrarSesion(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private void cerrarSesion() { |
|  |

|  |
| --- |
| mGoogleSignInClient.signOut().addOnCompleteListener(this, |
|  |

|  |
| --- |
| task -> updateUI(null)); |
|  |

|  |
| --- |
| signOut(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onStart() { |
|  |

|  |
| --- |
| super.onStart();/\* Check if user is signed in (non-null) and update UI accordingly.\*/ |
|  |

|  |
| --- |
| if (Variables.HAY\_INTERNET) { |
|  |

|  |
| --- |
| FirebaseUser currentUser = mAuth.getCurrentUser(); |
|  |

|  |
| --- |
| if (currentUser != null) { |
|  |

|  |
| --- |
| getUser(info\_user); |
|  |

|  |
| --- |
| StartThread(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| protected void onActivityResult(int requestCode, int resultCode, Intent data) { |
|  |

|  |
| --- |
| super.onActivityResult(requestCode, resultCode, data); |
|  |

|  |
| --- |
| if (requestCode == GOOGLE\_SIGN\_IN) { |
|  |

|  |
| --- |
| Task<GoogleSignInAccount> task = GoogleSignIn.getSignedInAccountFromIntent(data); |
|  |

|  |
| --- |
| try { |
|  |

|  |
| --- |
| GoogleSignInAccount account = task.getResult(ApiException.class); |
|  |

|  |
| --- |
| if (account != null) firebaseAuthWithGoogle(account); |
|  |

|  |
| --- |
| } catch (ApiException e) { |
|  |

|  |
| --- |
| Log.w("TAG", "Fallo el inicio de sesión con google.", e); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private void firebaseAuthWithGoogle(GoogleSignInAccount acct) { |
|  |

|  |
| --- |
| Log.d("TAG", "firebaseAuthWithGoogle:" + acct.getId()); |
|  |

|  |
| --- |
| AuthCredential credential = GoogleAuthProvider.getCredential(acct.getIdToken(), null); |
|  |

|  |
| --- |
| mAuth.signInWithCredential(credential) |
|  |

|  |
| --- |
| .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() { |
|  |

|  |
| --- |
| @Override |
|  |

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

|  |
| --- |
| if (task.isSuccessful()) { |
|  |

|  |
| --- |
| FirebaseUser user =mAuth.getCurrentUser(); |
|  |

|  |
| --- |
| GuardadUsuario(info\_user); |
|  |

|  |
| --- |
| updateUI(user); |
|  |

|  |
| --- |
| Toast.makeText(getApplicationContext(), "Bienvenido.. " + user.getDisplayName(), Toast.LENGTH\_LONG).show(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| Toast.makeText(getApplicationContext(), "Usuario o Contraseña Incorrectos!", Toast.LENGTH\_LONG).show(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private void updateUI(FirebaseUser user) { |
|  |

|  |
| --- |
| if (user != null) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| HashMap<String, String> info\_user = new HashMap<String, String>(); |
|  |

|  |
| --- |
| info\_user.put("user\_name", user.getDisplayName()); |
|  |

|  |
| --- |
| info\_user.put("user\_email", user.getEmail()); |
|  |

|  |
| --- |
| info\_user.put("user\_photo", String.valueOf(user.getPhotoUrl())); |
|  |

|  |
| --- |
| info\_user.put("user\_id", user.getUid()); |
|  |

|  |
| --- |
| info\_user.put("user\_phone", user.getPhoneNumber()); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| Intent intent = new Intent(this, perfil\_usuario\_1.class); |
|  |

|  |
| --- |
| intent.putExtra("info\_user", info\_user); |
|  |

|  |
| --- |
| startActivity(intent); |
|  |

|  |
| --- |
| finish(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public void login\_google(View view) { |
|  |

|  |
| --- |
| if (Variables.HAY\_INTERNET) { |
|  |

|  |
| --- |
| Intent signInIntent = mGoogleSignInClient.getSignInIntent(); |
|  |

|  |
| --- |
| startActivityForResult(signInIntent, GOOGLE\_SIGN\_IN); |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| Toast.makeText(LoginActivity.this, "Conexion Perdida", Toast.LENGTH\_SHORT).show(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public void register(View view) { |
|  |

|  |
| --- |
| if (Variables.HAY\_INTERNET) { |
|  |

|  |
| --- |
| Intent signInIntent = new Intent(getApplicationContext(), RegisterActivity.class); |
|  |

|  |
| --- |
| startActivity(signInIntent); |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| Toast.makeText(LoginActivity.this, "Conexion Perdida", Toast.LENGTH\_SHORT).show(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public void login(View view) { |
|  |

|  |
| --- |
| if (Variables.HAY\_INTERNET) { |
|  |

|  |
| --- |
| FirebaseUser user =mAuth.getCurrentUser(); |
|  |

|  |
| --- |
| if (user == null) { |
|  |

|  |
| --- |
| signIn(email.getText().toString(), pass.getText().toString()); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| Toast.makeText(LoginActivity.this, "Conexion Perdida", Toast.LENGTH\_SHORT).show(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private void signIn(String email, String password) { |
|  |

|  |
| --- |
| Log.d(TAG, "signIn:" + email); |
|  |

|  |
| --- |
| if (!validateForm()) { |
|  |

|  |
| --- |
| return; |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| // [START sign\_in\_with\_email] |
|  |

|  |
| --- |
| mAuth.signInWithEmailAndPassword(email, password) |
|  |

|  |
| --- |
| .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() { |
|  |

|  |
| --- |
| @Override |
|  |

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

|  |
| --- |
| if (task.isSuccessful()) { |
|  |

|  |
| --- |
| // Sign in success, update UI with the signed-in user's information |
|  |

|  |
| --- |
| Log.d(TAG, "signInWithEmail:success"); |
|  |

|  |
| --- |
| FirebaseUser user = mAuth.getCurrentUser(); |
|  |

|  |
| --- |
| Toast.makeText(getApplicationContext(), "Bienvenido... ", Toast.LENGTH\_LONG).show(); |
|  |

|  |
| --- |
| getUser(info\_user); |
|  |

|  |
| --- |
| StartThread(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| // If sign in fails, display a message to the user. |
|  |

|  |
| --- |
| Log.w(TAG, "signInWithEmail:failure", task.getException()); |
|  |

|  |
| --- |
| Toast.makeText(LoginActivity.this, "Usuario o Contraseña Incorrectos!!", |
|  |

|  |
| --- |
| Toast.LENGTH\_SHORT).show(); |
|  |

|  |
| --- |
| updateUI(null); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| // [START\_EXCLUDE] |
|  |

|  |
| --- |
| if (!task.isSuccessful()) { |
|  |

|  |
| --- |
| Toast.makeText(LoginActivity.this, "Authentication failed.", |
|  |

|  |
| --- |
| Toast.LENGTH\_SHORT).show(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| // [END\_EXCLUDE] |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }); |
|  |

|  |
| --- |
| // [END sign\_in\_with\_email] |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private void signOut() { |
|  |

|  |
| --- |
| mAuth.signOut(); |
|  |

|  |
| --- |
| updateUI(null); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private boolean validateForm() { |
|  |

|  |
| --- |
| boolean valid = true; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| String email = this.email.getText().toString(); |
|  |

|  |
| --- |
| if (TextUtils.isEmpty(email)) { |
|  |

|  |
| --- |
| this.email.setError("Required."); |
|  |

|  |
| --- |
| valid = false; |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| this.email.setError(null); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| String password = this.pass.getText().toString(); |
|  |

|  |
| --- |
| if (TextUtils.isEmpty(password)) { |
|  |

|  |
| --- |
| pass.setError("Required."); |
|  |

|  |
| --- |
| valid = false; |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| pass.setError(null); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| return valid; |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public void StartThread() { |
|  |

|  |
| --- |
| Thread thread = new Thread() { |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void run() { |
|  |

|  |
| --- |
| while (info\_user.isEmpty()) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| try { |
|  |

|  |
| --- |
| Intent intent = new Intent(getApplicationContext(), perfil\_usuario\_1.class); |
|  |

|  |
| --- |
| intent.putExtra("info\_user", info\_user); |
|  |

|  |
| --- |
| startActivity(intent); |
|  |

|  |
| --- |
| finish(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| }catch (ConcurrentModificationException e){ |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| thread.start(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public void getUser(HashMap<String, String> info\_user){ |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| FirebaseUser user =mAuth.getCurrentUser(); |
|  |

|  |
| --- |
| if(user== null){ |
|  |

|  |
| --- |
| return; |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| FirebaseDatabase.getInstance().getReference().child("Aplicacion").child("Usuarios") |
|  |

|  |
| --- |
| .child(user.getUid()) |
|  |

|  |
| --- |
| .addListenerForSingleValueEvent(new ValueEventListener() { |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onDataChange(DataSnapshot dataSnapshot) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| info\_user.put("user\_iduser",String.valueOf(dataSnapshot.child("user\_email").getValue())); |
|  |

|  |
| --- |
| info\_user.put("user\_name",String.valueOf(dataSnapshot.child("user\_name").getValue())); |
|  |

|  |
| --- |
| info\_user.put("user\_email", String.valueOf(dataSnapshot.child("user\_email").getValue())); |
|  |

|  |
| --- |
| info\_user.put("user\_photo", String.valueOf(dataSnapshot.child("user\_photo").getValue())); |
|  |

|  |
| --- |
| info\_user.put("user\_id", user.getUid()); |
|  |

|  |
| --- |
| info\_user.put("user\_phone",String.valueOf(dataSnapshot.child("user\_phone").getValue())); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onCancelled(DatabaseError databaseError) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| }public void GuardadUsuario(HashMap<String, String> info\_user) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| FirebaseUser user = mAuth.getCurrentUser(); |
|  |

|  |
| --- |
| if (user != null) { |
|  |

|  |
| --- |
| if (info\_user == null) { |
|  |

|  |
| --- |
| info\_user = new HashMap<>(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| info\_user.put("user\_iduser", user.getEmail()); |
|  |

|  |
| --- |
| info\_user.put("user\_name", user.getDisplayName()); |
|  |

|  |
| --- |
| info\_user.put("user\_email", user.getEmail()); |
|  |

|  |
| --- |
| info\_user.put("user\_photo", String.valueOf(user.getPhotoUrl())); |
|  |

|  |
| --- |
| info\_user.put("user\_id", user.getUid()); |
|  |

|  |
| --- |
| info\_user.put("user\_phone", user.getPhoneNumber()); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| DatabaseReference data = db\_reference.child("Usuarios").child(info\_user.get("user\_id")); |
|  |

|  |
| --- |
| data.child("user\_name").setValue(info\_user.get("user\_name")); |
|  |

|  |
| --- |
| data.child("user\_email").setValue(info\_user.get("user\_email")); |
|  |

|  |
| --- |
| data.child("user\_phone").setValue(info\_user.get("user\_phone")); |
|  |

|  |
| --- |
| data.child("user\_iduser").setValue(info\_user.get("user\_iduser")); |
|  |

|  |
| --- |
| data.child("user\_photo").setValue(info\_user.get("user\_photo")); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

}

Esta clase nos ayuda a realizar el respectivo registro en Firebase para el login y la base de datos

**RegisterActivity.java**

|  |
| --- |
| public class RegisterActivity extends AppCompatActivity { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private Button back; |
|  |

|  |
| --- |
| private Button register; |
|  |

|  |
| --- |
| private HashMap<String, String> info\_user; |
|  |

|  |
| --- |
| private EditText nombre; |
|  |

|  |
| --- |
| private EditText apellido; |
|  |

|  |
| --- |
| private EditText usuario; |
|  |

|  |
| --- |
| private EditText pass; |
|  |

|  |
| --- |
| private EditText confirmPass; |
|  |

|  |
| --- |
| private EditText correo; |
|  |

|  |
| --- |
| private EditText telefono; |
|  |

|  |
| --- |
| private DatabaseReference db\_reference; |
|  |

|  |
| --- |
| private static final String TAG = "EmailPassword"; |
|  |

|  |
| --- |
| private FirebaseAuth mAuth; |
|  |

|  |
| --- |
| private static final String ALPHA\_NUMERIC\_STRING = "ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789"; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| protected void onCreate(Bundle savedInstanceState) { |
|  |

|  |
| --- |
| super.onCreate(savedInstanceState); |
|  |

|  |
| --- |
| setContentView(R.layout.activity\_register); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| db\_reference = FirebaseDatabase.getInstance().getReference().child("Aplicacion"); |
|  |

|  |
| --- |
| mAuth = FirebaseAuth.getInstance(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| back = findViewById(R.id.anterior); |
|  |

|  |
| --- |
| register = findViewById(R.id.btn\_register); |
|  |

|  |
| --- |
| nombre = findViewById(R.id.txt\_nombre); |
|  |

|  |
| --- |
| apellido = findViewById(R.id.txt\_apellido); |
|  |

|  |
| --- |
| usuario = findViewById(R.id.txt\_usuario); |
|  |

|  |
| --- |
| pass = findViewById(R.id.txt\_pass); |
|  |

|  |
| --- |
| confirmPass = findViewById(R.id.txt\_confirmpass); |
|  |

|  |
| --- |
| correo = findViewById(R.id.txt\_correo); |
|  |

|  |
| --- |
| telefono = findViewById(R.id.txt\_telefono); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| back.setOnClickListener(new View.OnClickListener() { |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onClick(View v) { |
|  |

|  |
| --- |
| finish(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }); |
|  |

|  |
| --- |
| Intent intent = getIntent(); |
|  |

|  |
| --- |
| info\_user = (HashMap<String, String>) intent.getSerializableExtra("info\_user"); |
|  |

|  |
| --- |
| cargarDatos(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| register.setOnClickListener(new View.OnClickListener() { |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onClick(View v) { |
|  |

|  |
| --- |
| createAccount(correo.getText().toString(), pass.getText().toString()); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private void cargarDatos() { |
|  |

|  |
| --- |
| if (info\_user != null) { |
|  |

|  |
| --- |
| nombre.setText(info\_user.get("user\_name")); |
|  |

|  |
| --- |
| apellido.setText(info\_user.get("user\_name")); |
|  |

|  |
| --- |
| correo.setText(info\_user.get("user\_email")); |
|  |

|  |
| --- |
| telefono.setText(info\_user.get("user\_phone")); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private void createAccount(String email, String password) { |
|  |

|  |
| --- |
| Log.d(TAG, "createAccount:" + email); |
|  |

|  |
| --- |
| if (!validateForm()) { |
|  |

|  |
| --- |
| return; |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| // [START create\_user\_with\_email] |
|  |

|  |
| --- |
| mAuth.createUserWithEmailAndPassword(email, password) |
|  |

|  |
| --- |
| .addOnCompleteListener(this, new OnCompleteListener<AuthResult>() { |
|  |

|  |
| --- |
| @Override |
|  |

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

|  |
| --- |
| if (task.isSuccessful()) { |
|  |

|  |
| --- |
| // Sign in success, update UI with the signed-in user's information |
|  |

|  |
| --- |
| Log.d(TAG, "createUserWithEmail:success"); |
|  |

|  |
| --- |
| FirebaseUser user = mAuth.getCurrentUser(); |
|  |

|  |
| --- |
| GuardadUsuario(info\_user, user); |
|  |

|  |
| --- |
| Intent intent = new Intent(getApplicationContext(), perfil\_usuario\_1.class); |
|  |

|  |
| --- |
| intent.putExtra("info\_user", info\_user); |
|  |

|  |
| --- |
| startActivity(intent); |
|  |

|  |
| --- |
| finish(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| // If sign in fails, display a message to the user. |
|  |

|  |
| --- |
| Log.w(TAG, "createUserWithEmail:failure", task.getException()); |
|  |

|  |
| --- |
| Toast.makeText(RegisterActivity.this, "Authentication failed." + task.getException().getMessage(), |
|  |

|  |
| --- |
| Toast.LENGTH\_SHORT).show(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| // [START\_EXCLUDE] |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| // [END\_EXCLUDE] |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }); |
|  |

|  |
| --- |
| // [END create\_user\_with\_email] |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private boolean validateForm() { |
|  |

|  |
| --- |
| boolean valid = true; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| String email = this.correo.getText().toString(); |
|  |

|  |
| --- |
| if (TextUtils.isEmpty(email)) { |
|  |

|  |
| --- |
| this.correo.setError("Required."); |
|  |

|  |
| --- |
| valid = false; |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| this.correo.setError(null); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| String password = this.pass.getText().toString(); |
|  |

|  |
| --- |
| if (TextUtils.isEmpty(password)) { |
|  |

|  |
| --- |
| pass.setError("Required."); |
|  |

|  |
| --- |
| valid = false; |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| pass.setError(null); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| String confir = this.confirmPass.getText().toString(); |
|  |

|  |
| --- |
| if (TextUtils.isEmpty(password)) { |
|  |

|  |
| --- |
| confirmPass.setError("Required."); |
|  |

|  |
| --- |
| valid = false; |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| pass.setError(null); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| if (password != confir) { |
|  |

|  |
| --- |
| pass.setError("Contraseñas No iguales"); |
|  |

|  |
| --- |
| confirmPass.setError("Contraseñas No iguales"); |
|  |

|  |
| --- |
| valid = false; |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| pass.setError(null); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| return valid; |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private void GuardadUsuario(HashMap<String, String> info\_user, FirebaseUser user) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| if (info\_user == null) { |
|  |

|  |
| --- |
| info\_user = new HashMap<>(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| info\_user.put("user\_phone", telefono.getText().toString()); |
|  |

|  |
| --- |
| info\_user.put("user\_iduser", usuario.getText().toString()); |
|  |

|  |
| --- |
| info\_user.put("user\_name", nombre.getText().toString()); |
|  |

|  |
| --- |
| info\_user.put("user\_email", correo.getText().toString()); |
|  |

|  |
| --- |
| info\_user.put("user\_photo", "https://upload.wikimedia.org/wikipedia/commons/thumb/1/12/User\_icon\_2.svg/1200px-User\_icon\_2.svg.png"); |
|  |

|  |
| --- |
| info\_user.put("user\_id", user.getUid()); |
|  |

|  |
| --- |
| info\_user.put("user\_phone", telefono.getText().toString()); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| DatabaseReference data = db\_reference.child("Usuarios").child(info\_user.get("user\_id")); |
|  |

|  |
| --- |
| data.child("user\_name").setValue(info\_user.get("user\_name")); |
|  |

|  |
| --- |
| data.child("user\_email").setValue(info\_user.get("user\_email")); |
|  |

|  |
| --- |
| data.child("user\_phone").setValue(info\_user.get("user\_phone")); |
|  |

|  |
| --- |
| data.child("user\_iduser").setValue(info\_user.get("user\_iduser")); |
|  |

|  |
| --- |
| data.child("user\_photo").setValue(info\_user.get("user\_photo")); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public static String randomAlphaNumeric(int count) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| StringBuilder builder = new StringBuilder(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| while (count-- != 0) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| int character = (int) (Math.random() \* ALPHA\_NUMERIC\_STRING.length()); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| builder.append(ALPHA\_NUMERIC\_STRING.charAt(character)); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| return builder.toString(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

}

Esta clase realiza todas las consultas a la base de datos de firebase de las posiciones de la mochila y la ubica en el mapa, así mismo se mantiene escuchando nuevos registros de posición.

**MapFragment.java**

|  |
| --- |
| public class MapFragment extends Fragment implements OnMapReadyCallback, GoogleMap.OnMyLocationButtonClickListener, GoogleMap.OnMyLocationClickListener { |
|  |

|  |
| --- |
| private static final int TAG\_CODE\_PERMISSION\_LOCATION = 200; |
|  |

|  |
| --- |
| private FirebaseAuth mAuth; |
|  |

|  |
| --- |
| private FirebaseUser currentUser; |
|  |

|  |
| --- |
| private DatabaseReference ref; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private MapViewModel mapViewModel; |
|  |

|  |
| --- |
| private static final String TAG = MapFragment.class.getSimpleName(); |
|  |

|  |
| --- |
| private GoogleMap mMap; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private LinkedList<LatLng> posiciones; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private HashMap<String, Marker> mMarkers; |
|  |

|  |
| --- |
| public static double DISTANCIA = 0.0d; |
|  |

|  |
| --- |
| public static boolean Ready = false; |
|  |

|  |
| --- |
| private Location location; |
|  |

|  |
| --- |
| private ChildEventListener child; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public View onCreateView(@NonNull LayoutInflater inflater, |
|  |

|  |
| --- |
| ViewGroup container, Bundle savedInstanceState) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| mapViewModel = |
|  |

|  |
| --- |
| ViewModelProviders.of(this).get(MapViewModel.class); |
|  |

|  |
| --- |
| View root = inflater.inflate(R.layout.fragment\_map, container, false); |
|  |

|  |
| --- |
| mMap = null; |
|  |

|  |
| --- |
| ((perfil\_usuario\_1) getActivity()).getSupportActionBar().setTitle("Home Mapa"); |
|  |

|  |
| --- |
| SupportMapFragment mapFragment = (SupportMapFragment) getChildFragmentManager() |
|  |

|  |
| --- |
| .findFragmentById(R.id.map1); |
|  |

|  |
| --- |
| if (mapFragment == null) { |
|  |

|  |
| --- |
| FragmentManager fm = getFragmentManager(); |
|  |

|  |
| --- |
| FragmentTransaction ft = fm.beginTransaction(); |
|  |

|  |
| --- |
| mapFragment = SupportMapFragment.newInstance(); |
|  |

|  |
| --- |
| ft.replace(R.id.map1, mapFragment).commit(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| mapFragment.getMapAsync(this); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| DISTANCIA = 0.0d; |
|  |

|  |
| --- |
| mMarkers = new HashMap<>(); |
|  |

|  |
| --- |
| posiciones = new LinkedList<>(); |
|  |

|  |
| --- |
| this.mAuth = FirebaseAuth.getInstance(); |
|  |

|  |
| --- |
| this.currentUser = mAuth.getCurrentUser(); |
|  |

|  |
| --- |
| ActivityCompat.requestPermissions(getActivity(), new String[] { |
|  |

|  |
| --- |
| Manifest.permission.ACCESS\_FINE\_LOCATION, |
|  |

|  |
| --- |
| Manifest.permission.ACCESS\_COARSE\_LOCATION }, |
|  |

|  |
| --- |
| TAG\_CODE\_PERMISSION\_LOCATION); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| this.ref = FirebaseDatabase.getInstance().getReference().child("datos\_gps").child(Variables.ID\_MALETA); |
|  |

|  |
| --- |
| this.ref.removeEventListener(new ValueEventListener() { |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onDataChange(@NonNull DataSnapshot dataSnapshot) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onCancelled(@NonNull DatabaseError databaseError) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| StartThread(); |
|  |

|  |
| --- |
| HiloNotificacions(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| return root; |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onMapReady(GoogleMap googleMap) { |
|  |

|  |
| --- |
| // Authenticate with Firebase when the Google map is loaded |
|  |

|  |
| --- |
| mMap = googleMap; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| mMap.setMaxZoomPreference(16); |
|  |

|  |
| --- |
| if (ContextCompat.checkSelfPermission(getActivity().getApplicationContext(), android.Manifest.permission.ACCESS\_FINE\_LOCATION) == |
|  |

|  |
| --- |
| PackageManager.PERMISSION\_GRANTED && |
|  |

|  |
| --- |
| ContextCompat.checkSelfPermission(getActivity().getApplicationContext(), android.Manifest.permission.ACCESS\_COARSE\_LOCATION) == |
|  |

|  |
| --- |
| PackageManager.PERMISSION\_GRANTED) { |
|  |

|  |
| --- |
| mMap.setMyLocationEnabled(true); |
|  |

|  |
| --- |
| mMap.getUiSettings().setMyLocationButtonEnabled(true); |
|  |

|  |
| --- |
| }else { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| Toast.makeText(getActivity().getApplicationContext(), "No inicia Mi posicion", Toast.LENGTH\_LONG).show(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| mMap.setOnMyLocationButtonClickListener(this); |
|  |

|  |
| --- |
| mMap.setOnMyLocationClickListener(this); |
|  |

|  |
| --- |
| LocationManager locationManager = (LocationManager) getContext().getSystemService(LOCATION\_SERVICE); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| try { |
|  |

|  |
| --- |
| Location myLocation = locationManager.getLastKnownLocation(LocationManager.GPS\_PROVIDER); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| if (myLocation != null) { |
|  |

|  |
| --- |
| Log.d("TAG", "Not null"); |
|  |

|  |
| --- |
| this.location = myLocation; |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| else { |
|  |

|  |
| --- |
| Log.d("TAG", "NULL"); |
|  |

|  |
| --- |
| locationManager.requestLocationUpdates(LocationManager.GPS\_PROVIDER, 0, 0, (LocationListener) this); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| catch (SecurityException se) { |
|  |

|  |
| --- |
| Log.d("TAG", "SE CAUGHT"); |
|  |

|  |
| --- |
| se.printStackTrace(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| subscribeToUpdates(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| //Toast.makeText(getActivity(),"La distancia de tu maleta ahora es de .."+String.valueOf(DISTANCIA), Toast.LENGTH\_SHORT).show(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onStart() { |
|  |

|  |
| --- |
| Ready = true; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| super.onStart(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onResume() { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| if(child!=null){ |
|  |

|  |
| --- |
| ref.removeEventListener(child); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| super.onResume(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onDestroy() { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| super.onDestroy(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private void subscribeToUpdates() { |
|  |

|  |
| --- |
| child = new ChildEventListener() { |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onChildAdded(DataSnapshot dataSnapshot, String previousChildName) { |
|  |

|  |
| --- |
| setMarker(dataSnapshot); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onChildChanged(DataSnapshot dataSnapshot, String previousChildName) { |
|  |

|  |
| --- |
| setMarker(dataSnapshot); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onChildMoved(DataSnapshot dataSnapshot, String previousChildName) { |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onChildRemoved(DataSnapshot dataSnapshot) { |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onCancelled(DatabaseError error) { |
|  |

|  |
| --- |
| Log.d(TAG, "Failed to read value.", error.toException()); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }; |
|  |

|  |
| --- |
| this.ref.addChildEventListener(child); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private void setMarker(DataSnapshot dataSnapshot) { |
|  |

|  |
| --- |
| // When a location update is received, put or update |
|  |

|  |
| --- |
| // its value in mMarkers, which contains all the markers |
|  |

|  |
| --- |
| // for locations received, so that we can build the |
|  |

|  |
| --- |
| // boundaries required to show them all on the map at once |
|  |

|  |
| --- |
| String key = dataSnapshot.getKey(); |
|  |

|  |
| --- |
| HashMap<String, Object> value = (HashMap<String, Object>) dataSnapshot.getValue(); |
|  |

|  |
| --- |
| double lat = Double.parseDouble(value.get("Latitud").toString()); |
|  |

|  |
| --- |
| double lng = Double.parseDouble(value.get("Longitud").toString()); |
|  |

|  |
| --- |
| Date date = new Date(); |
|  |

|  |
| --- |
| DateFormat hourdateFormat = new SimpleDateFormat("HH:mm dd/MM/yyyy"); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| if (lat != 0 && lng != 0) { |
|  |

|  |
| --- |
| LatLng location = new LatLng(lat, lng); |
|  |

|  |
| --- |
| posiciones.add(location); |
|  |

|  |
| --- |
| if (!mMarkers.containsKey(key)) { |
|  |

|  |
| --- |
| mMarkers.put(key, mMap.addMarker(new MarkerOptions().title(hourdateFormat.format(date)).position(location))); |
|  |

|  |
| --- |
| } else { |
|  |

|  |
| --- |
| mMarkers.get(key).setPosition(location); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| LatLngBounds.Builder builder = new LatLngBounds.Builder(); |
|  |

|  |
| --- |
| for (Marker marker : mMarkers.values()) { |
|  |

|  |
| --- |
| builder.include(marker.getPosition()); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| mMap.animateCamera(CameraUpdateFactory.newLatLngBounds(builder.build(), 350)); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public void StartThread() { |
|  |

|  |
| --- |
| Thread thread = new Thread() { |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void run() { |
|  |

|  |
| --- |
| while (true) { |
|  |

|  |
| --- |
| try { |
|  |

|  |
| --- |
| Thread.sleep(5000); |
|  |

|  |
| --- |
| while (posiciones.isEmpty()) { |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| LocationManager locationManager = (LocationManager) getContext().getSystemService(LOCATION\_SERVICE); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| try { |
|  |

|  |
| --- |
| Location myLocation = locationManager.getLastKnownLocation(LocationManager.GPS\_PROVIDER); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| if (myLocation != null) { |
|  |

|  |
| --- |
| Log.d("TAG", "Not null"); |
|  |

|  |
| --- |
| location = myLocation; |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| else { |
|  |

|  |
| --- |
| Log.d("TAG", "NULL"); |
|  |

|  |
| --- |
| locationManager.requestLocationUpdates(LocationManager.GPS\_PROVIDER, 0, 0, (LocationListener) getContext()); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| catch (SecurityException se) { |
|  |

|  |
| --- |
| Log.d("TAG", "SE CAUGHT"); |
|  |

|  |
| --- |
| se.printStackTrace(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| DISTANCIA = CalcularDistancia(posiciones.getLast().latitude, posiciones.getLast().longitude |
|  |

|  |
| --- |
| ,location.getLatitude(), location.getLongitude()); |
|  |

|  |
| --- |
| //toast.show(); |
|  |

|  |
| --- |
| //Toast.makeText(contex,"La distancia de tu maleta ahora es de .."+String.valueOf(DISTANCIA), Toast.LENGTH\_SHORT).show(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } catch (InterruptedException e) { |
|  |

|  |
| --- |
| e.printStackTrace(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| thread.start(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private double CalcularDistancia(double lat1, double long1, double lat2, double long2) { |
|  |

|  |
| --- |
| double constante = Math.PI / 180; |
|  |

|  |
| --- |
| lat2 = lat2 \* constante; |
|  |

|  |
| --- |
| long2 = long2 \* constante; |
|  |

|  |
| --- |
| lat1 = lat1 \* constante; |
|  |

|  |
| --- |
| long1 = long1 \* constante; |
|  |

|  |
| --- |
| double R = 6371; //km |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| double deltalat = lat2 - lat1; |
|  |

|  |
| --- |
| double deltalong = long2 - long1; |
|  |

|  |
| --- |
| double a = Math.pow(Math.sin(deltalat / 2), 2) + Math.cos(lat1) \* Math.cos(lat2) \* Math.pow(Math.sin(deltalong / 2), 2); |
|  |

|  |
| --- |
| double c = 2 \* R \* Math.atan(Math.sqrt(a)) \* 1000; |
|  |

|  |
| --- |
| return c; |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public void HiloNotificacions() { |
|  |

|  |
| --- |
| Thread thread = new Thread() { |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void run() { |
|  |

|  |
| --- |
| while (true) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| try { |
|  |

|  |
| --- |
| Thread.sleep(10000); |
|  |

|  |
| --- |
| while (posiciones.isEmpty() && DISTANCIA != 0) { |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| if (DISTANCIA >= Variables.MAXIMO\_DISTANCIA && Variables.RECIBIR\_NOTIFICACION) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| enviarDatos(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } catch (InterruptedException e) { |
|  |

|  |
| --- |
| e.printStackTrace(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| thread.start(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public void enviarDatos() { |
|  |

|  |
| --- |
| String url = "https://things.ubidots.com/api/v1.6/devices/MyCellTraking/?token=BBFF-ibSk9rqslckzhIWiPyu2YYN18FG5Iw"; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| try { |
|  |

|  |
| --- |
| RequestQueue requestQueue = Volley.newRequestQueue(getActivity().getApplicationContext()); |
|  |

|  |
| --- |
| final org.json.JSONObject jsonBody = new org.json.JSONObject(); |
|  |

|  |
| --- |
| final org.json.JSONObject jsonBody1 = new org.json.JSONObject(); |
|  |

|  |
| --- |
| jsonBody1.put("value", DISTANCIA); |
|  |

|  |
| --- |
| jsonBody.put("Distancia", jsonBody1); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| JsonObjectRequest request = new JsonObjectRequest |
|  |

|  |
| --- |
| (Request.Method.POST, url, jsonBody, new Response.Listener<JSONObject>() { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onResponse(JSONObject response) { |
|  |

|  |
| --- |
| // TODO Auto-generated method stub |
|  |

|  |
| --- |
| // hazle un print al object o lo que gustes |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }, new Response.ErrorListener() { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onErrorResponse(VolleyError error) { |
|  |

|  |
| --- |
| // TODO Auto-generated method stub |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| }); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| requestQueue.add(request); |
|  |

|  |
| --- |
| } catch (JSONException e) { |
|  |

|  |
| --- |
| e.printStackTrace(); |
|  |

|  |
| --- |
| } catch (NullPointerException e) { |
|  |

|  |
| --- |
| e.printStackTrace(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public boolean onMyLocationButtonClick() { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| return false; |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Override |
|  |

|  |
| --- |
| public void onMyLocationClick(@NonNull Location location) { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| } |
|  |

}