

24 de agosto de 2021

Preparado por: AMST\_Grupo1

**Manual de Técnico - Calidad de Aire en Restaurantes**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | |  |
| Proyecto | Analizador de Calidad de Aire en restaurantes |
| Título | Manual Técnico |
| Versión | 0.4 |
| Estado | Revisión |
| Autores | Ronaldo Rodríguez |
| Fecha de creación | 24/08/2021 |
| Confidencialidad | Solo para uso estudiantil |
|  |  |
|  |  |

**Historial del documento**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Autores** | **Comentarios** |
| 0.1 | 24/08/2021 | Ronaldo Rodríguez | Borrador |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contenido

[Introducción 4](#_Toc80890454)

[Diseño 4](#_Toc80890455)

[Diseño del hardware 5](#_Toc80890456)

[Diseño del software 5](#_Toc80890457)

[Metodología 5](#_Toc80890458)

[Resultados 5](#_Toc80890459)

[Conclusiones 5](#_Toc80890460)

[Presupuesto 5](#_Toc80890461)

[Apéndice 5](#_Toc80890462)

[Código Android Studio 5](#_Toc80890463)

[Clases Java 5](#_Toc80890464)

[Login Activity 5](#_Toc80890465)

[Main Activity 5](#_Toc80890466)

[RegistrarResUsuario 5](#_Toc80890467)

[Restaurantes 5](#_Toc80890468)

[Restaurantes Adapter 5](#_Toc80890469)

[Restaurantes para Jefes 5](#_Toc80890470)

[Documentos XML 5](#_Toc80890471)

[Activity login 5](#_Toc80890472)

[Restaurant user card 5](#_Toc80890473)

[Activity main 5](#_Toc80890474)

[Activity registrar res usuario 5](#_Toc80890475)

[Activity restaurantes 5](#_Toc80890476)

[Activity restaurantes para jefes 5](#_Toc80890477)

[Código Arduino 5](#_Toc80890478)

[Código ESPE8266 5](#_Toc80890479)

[Código CalidadDeAire 5](#_Toc80890480)

# Introducción

Debido a la pandemia se requiere tener bio-seguridad en locales para poder efectuar su trabajo o para tener una mejor estadía en los hogares. Se debe analizar la calidad de aire, para así ayudar en la prevención del virus COVID-19. Por ello se diseñó una aplicación avanzada que será capaz de monitorear la calidad de aire que se encuentra en un restaurante o local en específico, esto se lograra mediante un dispositivo con sensores que evaluaran la calidad de aire presente y esa información la podemos ver en nuestra aplicación avanzada para que los usuarios estén alertas antes de acudir al restaurante.

# Diseño

## Diseño del hardware

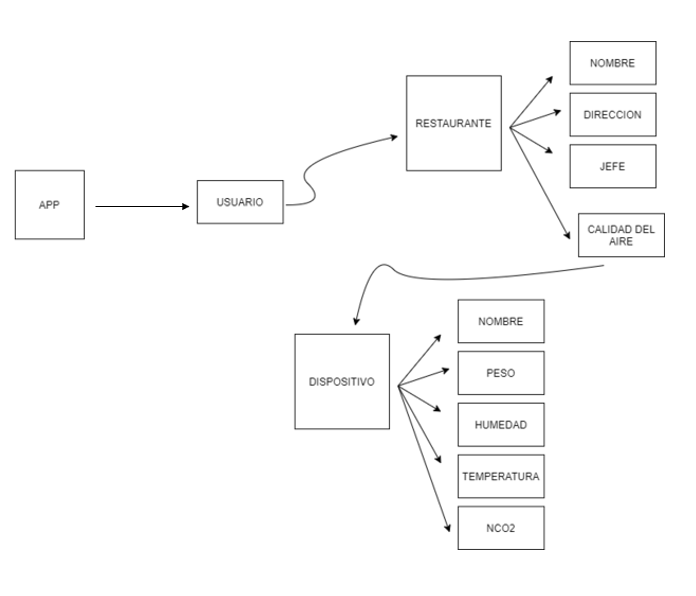


**Figura 1 - Diagrama de diseño**

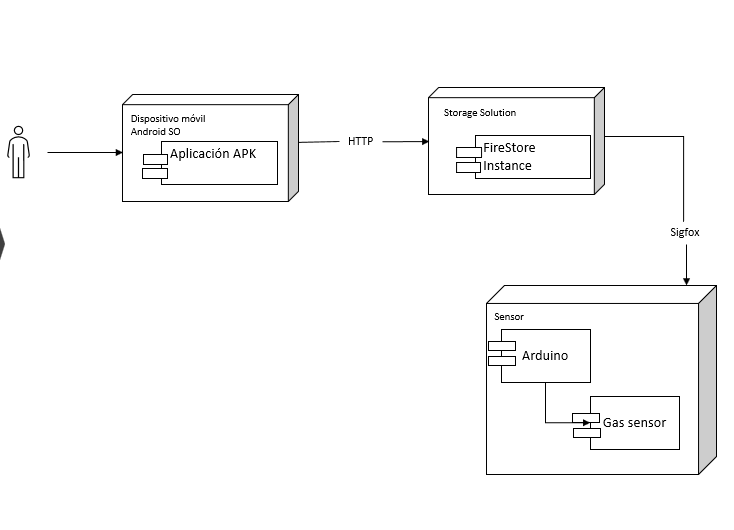
Diagrama

Descripción generada automáticamente

**Figura 2 – Diagrama de casos de usos**



**Figura 3 - Diagrama de árbol**

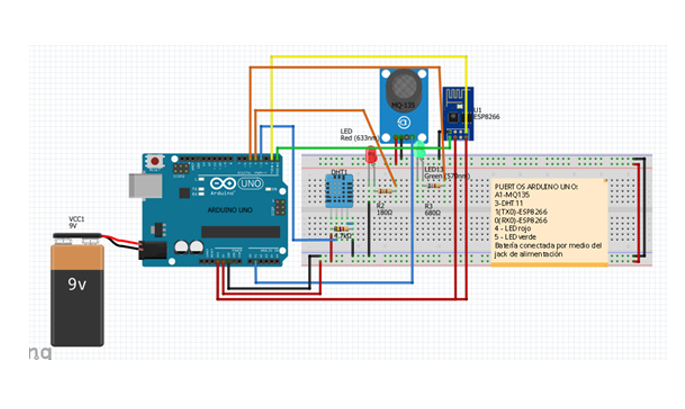


**Figura 4 - Diagrama de Despliegue**

Diagrama

Descripción generada automáticamente

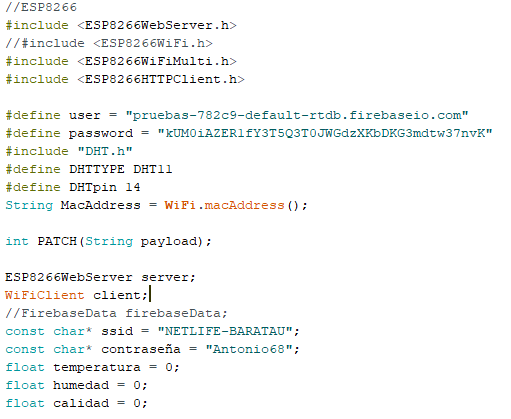
**Figura 5 - Diagrama de clases**

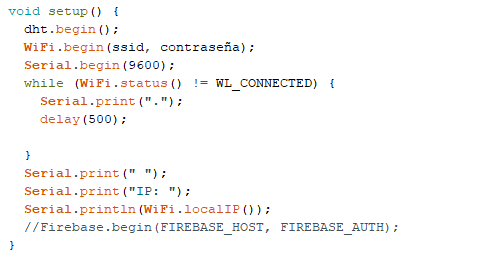


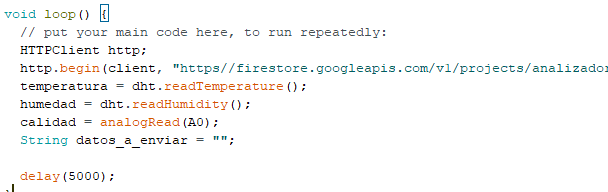
**Figura 6 - Diagrama del Dispositivo**

## Diseño del software

* Código ARDUINO ESP8266

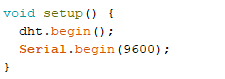
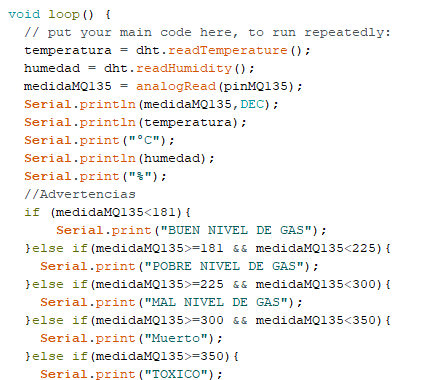
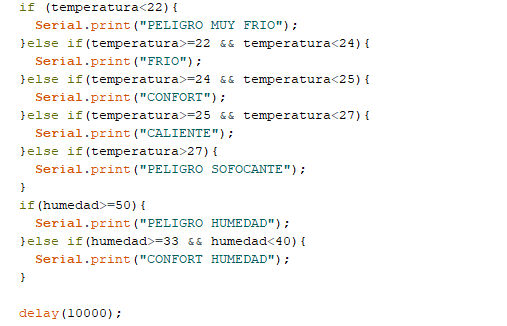






* Código ARDUINO CalidadDeAire



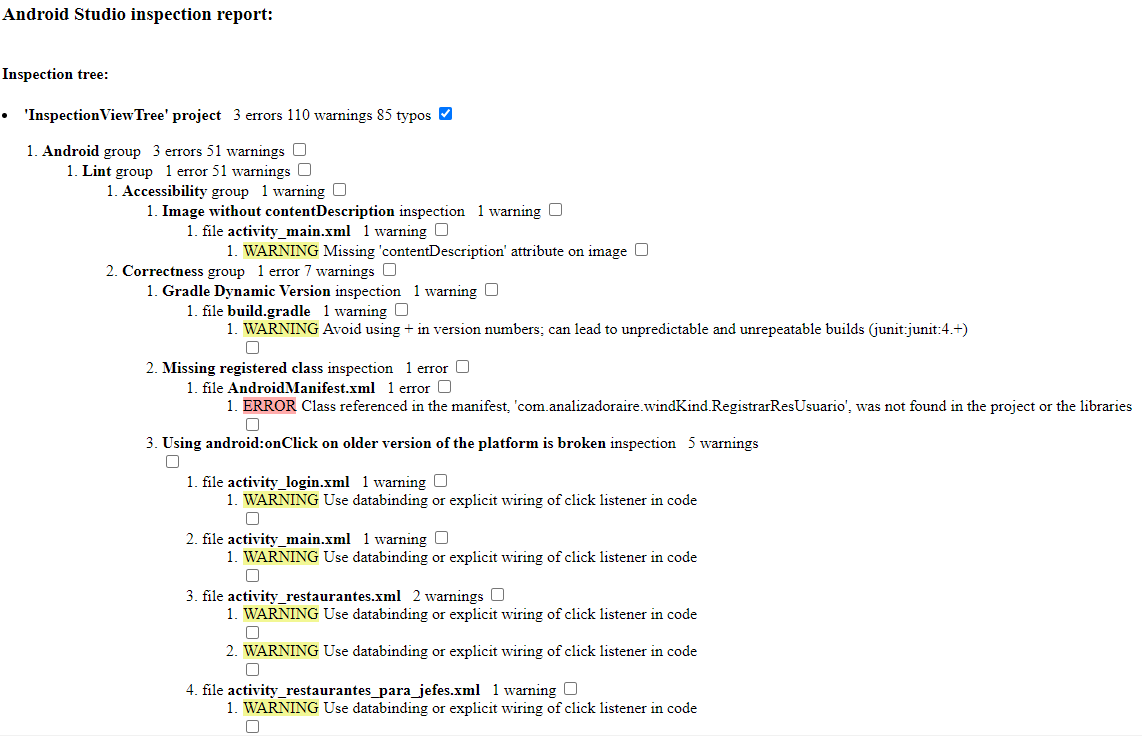
* Código Aplicación Móvil en Android Studio

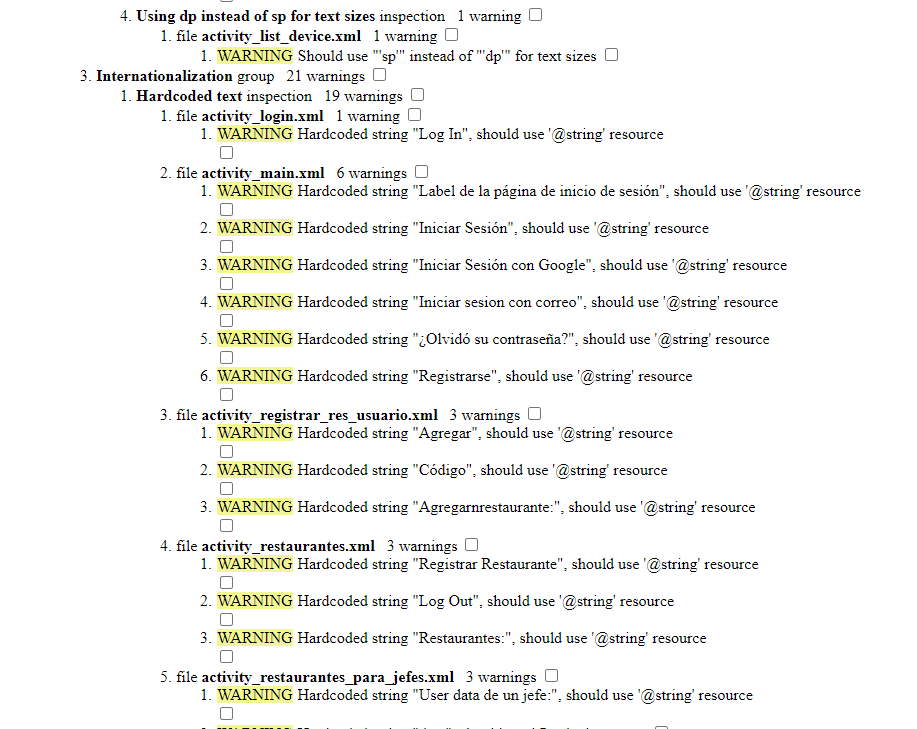
# Metodología

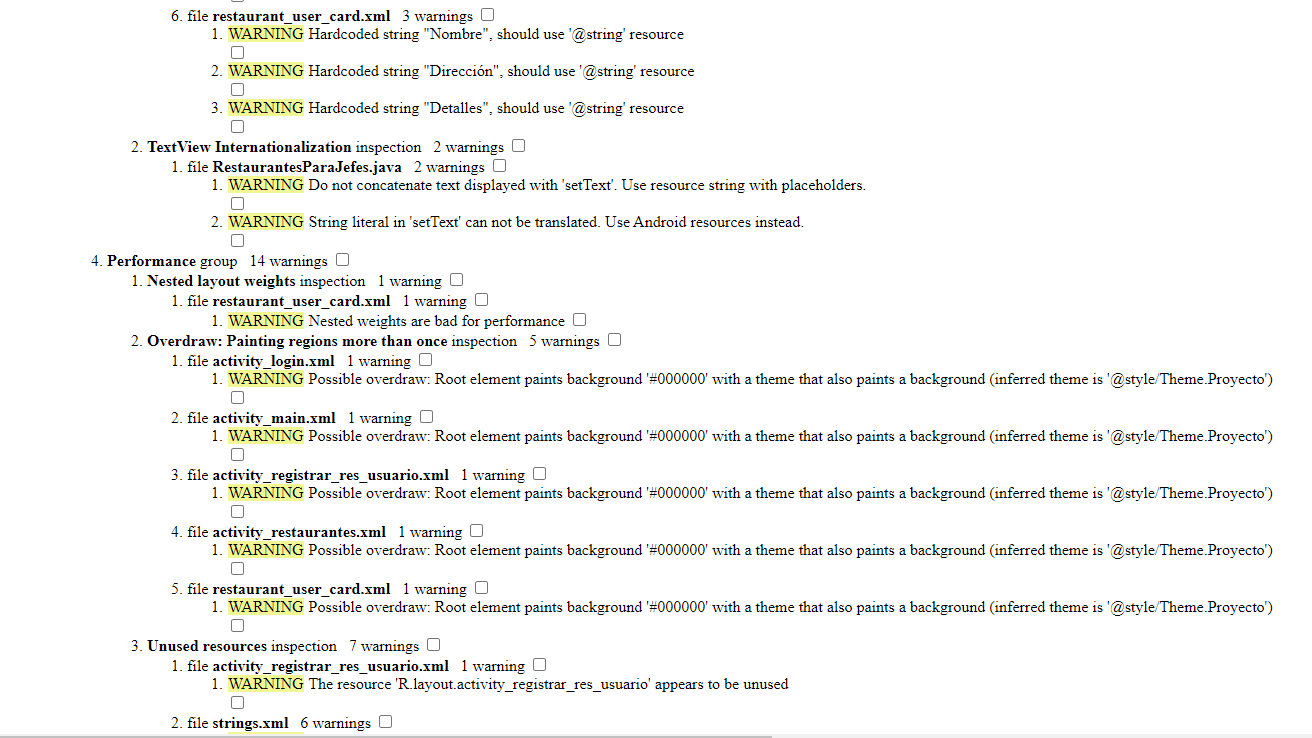
# Resultados

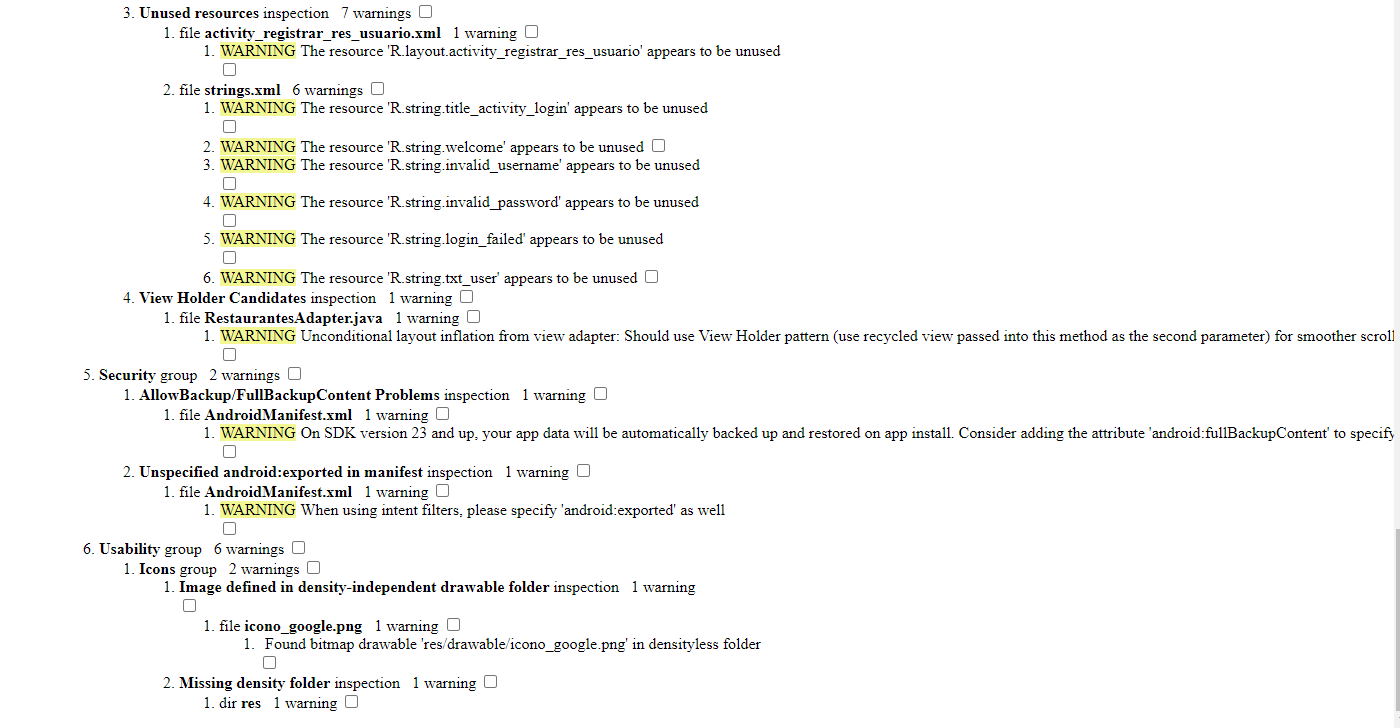
ANALIZADOR DE CODIGO FUENTE ANDROID STUDIO

**Capturas:**

****







# Conclusiones

El proyecto se desarrollo de manera paulatina, gracias a las diferentes plataforma que existen hoy en día como Zigfox se puede mejorar la calidad de vida de las personas creando estas aplicaciones útiles que tienen como principal objetivo perseverar la salud de las personas.

# Presupuesto

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Cantidad | Nombre |  | Descripción | Software en la nube | Vendedor | Precio |
| 1 | Modulo Wifi ESP8266 |  | Modulo Wifi | No | electrónica | 2.99 |
| 1 | Thinxtra |  | mensajería | Si | Thinxtra | Anual |
| 1 | Arduino UNO |  | Microcontrolador | No | electrónica | 2.99 |
| 1 | Modulo DHT111 |  | Sensor de Humedad | No | electrónica | 1.59 |
| 1 | Modulo MQ-135 |  | Sensor de Temperatura | No | electrónica | 1.50 |
| 1 | Android Studio |  | Sistema open source | No | Free | free |
| 1 | Servicio de notificación, autenticación |  | Firebase | Si | Free | fee |

# Apéndice

## Código Android Studio

### Clases Java

#### Login Activity

**private** FirebaseAuth **mAuth**;  
 **private** EditText **ETemail**, **ETpassword**;  
  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_login***);  
  
 **mAuth** = FirebaseAuth.*getInstance*();  
  
 **ETemail** = (EditText) findViewById(R.id.***username***);  
 **ETpassword** = (EditText) findViewById(R.id.***password***);  
}  
  
  
  
 **public void** signIn(View v){  
  
 String email = **ETemail**.getText().toString();  
 String password = **ETpassword**.getText().toString();  
  
 **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"**);  
 Intent intent = **new** Intent(LoginActivity.**this**, MainActivity.**class**);  
 startActivity(intent);  
 } **else** {  
 *// If sign in fails, display a message to the user.* Log.*w*(***TAG***, **"signInWithEmail:failure"**, task.getException());  
 Toast.*makeText*(getApplicationContext(), **"Authentication failed."**,  
 Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 });  
 }

#### Main Activity

**private** FirebaseAuth **mAuth**;  
**private final** Context **context** = **this**;  
  
*// Cuando se crea la instancia*@Override  
**protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
  
 *// Initialize Firebase Auth* **mAuth** = FirebaseAuth.*getInstance*();  
}  
  
*// Cuando se vuelve visible*@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**){  
 changeBasedOnRole(currentUser);  
 }  
}  
  
**public void** goToMailLogin(View view){  
 Intent intent = **new** Intent(**this**, LoginActivity.**class**);  
 startActivity(intent);  
}  
  
*/\*\*  
 \* Devuelve true si es que el usuario es jefe false si es que el usuario es usuario  
 \** ***@return*** *\*/***public void** changeBasedOnRole(FirebaseUser currentUser){  
 **if**(currentUser != **null**){  
 DocumentReference df = FirebaseFirestore.*getInstance*().collection(**"usuarios"**).document(currentUser.getUid());  
 df.get().addOnCompleteListener(**new** OnCompleteListener<DocumentSnapshot>() {  
 @Override  
 **public void** onComplete(@NonNull Task<DocumentSnapshot> task) {  
 **if** (task.isSuccessful()){  
 DocumentSnapshot doc = task.getResult();  
 **if** (doc.exists()){  
 **if** ((Boolean) ((Map) doc.getData().get(**"role"**)).get(**"jefe"**)){  
 Intent intent = **new** Intent(**context**, RestaurantesParaJefes.**class**);  
 startActivity(intent);  
 } **else** {  
 Intent intent = **new** Intent(**context**, Restaurantes.**class**);  
 startActivity(intent);  
 }  
 }  
 }  
 }  
 });  
 }  
}

#### RegistrarResUsuario

**private** EditText **codigo**;  
  
@Override  
**protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_registrar\_res\_usuario***);  
  
 **codigo** = (EditText) findViewById(R.id.***et\_codResUsu***);  
}  
  
  
**public void** regResUser(View view) {  
 String regCode = **codigo**.getText().toString();  
  
 FirebaseFirestore.*getInstance*()  
 .collection(**"restaurante"**)  
 .whereEqualTo(**"regCode"**, regCode)  
 .limit(1)  
 .get()  
 .addOnSuccessListener(**new** OnSuccessListener<QuerySnapshot>() {  
 @Override  
 **public void** onSuccess(QuerySnapshot documentSnapshots) {  
 List<DocumentSnapshot> snapshotsList = documentSnapshots.getDocuments();  
 **if** (snapshotsList.size() == 0) {  
 Toast.*makeText*(getApplicationContext(), **"Restaurante no existe."**,  
 Toast.***LENGTH\_SHORT***).show();  
 }  
 **for** (DocumentSnapshot res : snapshotsList) {  
 **if** (res.exists()) {  
 FirebaseUser currentUser = FirebaseAuth.*getInstance*().getCurrentUser();  
 DocumentReference rr = res.getReference();  
 DocumentReference df = FirebaseFirestore.*getInstance*().collection(**"usuarios"**).document(currentUser.getUid());  
 df.get().addOnCompleteListener(**new** OnCompleteListener<DocumentSnapshot>() {  
 @RequiresApi(api = Build.VERSION\_CODES.***N***)  
 @Override  
 **public void** onComplete(@NonNull Task<DocumentSnapshot> task) {  
 **if** (task.isSuccessful()){  
 DocumentSnapshot doc = task.getResult();  
 **if** (doc.exists()){  
 **if** ((Boolean) ((Map) Objects.*requireNonNull*(doc.getData()).get(**"role"**)).get(**"jefe"**)){  
 Intent intent = **new** Intent(RegistrarResUsuario.**this**, MainActivity.**class**);  
 startActivity(intent);  
 } **else** {  
 ArrayList<DocumentReference> referencias = ((ArrayList<DocumentReference>) Objects.*requireNonNull*(doc.getData().get(**"restaurantes"**)));  
 **if** (referencias.contains(rr)){  
 Toast.*makeText*(getApplicationContext(), **"Restaurante ya está registrado."**,  
 Toast.***LENGTH\_SHORT***).show();  
 } **else** {  
 referencias.add(rr);  
  
 Map<String, Object> update = **new** HashMap<>();  
 update.put(**"restaurantes"**, referencias);  
  
 df.set(update, SetOptions.*merge*());  
 }  
 }  
 }  
 }  
 }  
 });  
 } **else** {  
 Toast.*makeText*(getApplicationContext(), **"Restaurante no existe."**,  
 Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 }  
 });  
}

#### Restaurantes

ListView **listaRestaurantes**;  
**private boolean backCounter** = **false**;  
  
@Override  
**protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_restaurantes***);  
  
 Context context = getApplicationContext();  
  
 **listaRestaurantes** = (ListView) findViewById(android.R.id.***list***);  
 ArrayList<String[]> datos = **new** ArrayList<>();  
  
 FirebaseUser currentUser = FirebaseAuth.*getInstance*().getCurrentUser();  
 *// Only accesable when loggeed in so there's no need to check for a null user* DocumentReference df = FirebaseFirestore.*getInstance*().collection(**"usuarios"**).document(currentUser.getUid());  
 df.get().addOnCompleteListener(**new** OnCompleteListener<DocumentSnapshot>() {  
 @RequiresApi(api = Build.VERSION\_CODES.***N***)  
 @Override  
 **public void** onComplete(@NonNull Task<DocumentSnapshot> task) {  
 **if** (task.isSuccessful()){  
 DocumentSnapshot doc = task.getResult();  
 **if** (doc.exists()){  
 **if** ((Boolean) ((Map) Objects.*requireNonNull*(doc.getData()).get(**"role"**)).get(**"jefe"**)){  
 Intent intent = **new** Intent(context, MainActivity.**class**);  
 startActivity(intent);  
 } **else** {  
 **final int**[] counter = {0};  
 ArrayList<String[]> datos = **new** ArrayList<>();  
  
 Log.*println*(Log.***DEBUG***, **"restaurantes: "**, String.*valueOf*(Objects.*requireNonNull*(doc.getData().get(**"restaurantes"**))));  
 ArrayList<DocumentReference> referencias = ((ArrayList<DocumentReference>) Objects.*requireNonNull*(doc.getData().get(**"restaurantes"**)));  
  
 referencias.forEach((n) -> {  
 ((DocumentReference) n).get().addOnCompleteListener(**new** OnCompleteListener<DocumentSnapshot>() {  
 @Override  
 **public void** onComplete(@NonNull Task<DocumentSnapshot> task) {  
 DocumentSnapshot res = task.getResult();  
 **if** (res.exists()) {  
 String[] resData = **new** String[3];  
 resData[0] = String.*valueOf*(res.getData().get(**"nombre"**));  
 resData[1] = String.*valueOf*(res.getData().get(**"direccion"**));  
 resData[2] = String.*valueOf*(n);  
 Log.*println*(Log.***DEBUG***, **"restaurantes: "**, **"Completed "** + String.*valueOf*(resData) + **" and exists"**);  
 datos.add(resData);  
 counter[0] += 1;  
 **if** (counter[0] == referencias.size()){  
 setListAdapter(**new** RestaurantesAdapter(context, R.layout.***restaurant\_user\_card***, datos));  
 }  
 }  
 }  
 });  
 });  
 }  
 }  
 }  
 }  
 });  
  
 setListAdapter(**new** RestaurantesAdapter(context, R.layout.***restaurant\_user\_card***, datos));  
}  
  
@Override  
**public void** onBackPressed() {  
 *// super.onBackPressed();* **if** (**this**.**backCounter**) {  
 **this**.finishAffinity();  
 }  
  
 **this**.**backCounter** = **true**;  
 Toast.*makeText*(**this**, **"Presione otra vez para salir"**, Toast.***LENGTH\_SHORT***).show();  
  
 **new** Handler(Looper.*getMainLooper*()).postDelayed(**new** Runnable() {  
  
 @Override  
 **public void** run() {  
 **backCounter**=**false**;  
 }  
 }, 2000);  
  
}  
  
**public void** logOut(View view){  
 DBHelper.*logOut*();  
 Intent intent = **new** Intent(**this**, MainActivity.**class**);  
 startActivity(intent);  
}  
  
**public void** goToRegister(View view) {  
 Intent intent = **new** Intent(**this**, RegistrarResUsuario.**class**);  
 startActivity(intent);  
}

#### Restaurantes Adapter

**private** LayoutInflater **mInflater**;  
  
**private** ArrayList<String[]> **mString**;  
  
**private int mViewResourceId**;  
  
**private** Context **context**;  
  
**public** RestaurantesAdapter(@NonNull Context context, **int** viewResourceId, ArrayList<String[]> strings) {  
 **super**(context, viewResourceId);  
  
 **mInflater** = (LayoutInflater) context.getSystemService(Context.***LAYOUT\_INFLATER\_SERVICE***);  
  
 **mString** = strings;  
  
 **mViewResourceId** = viewResourceId;  
  
 **this**.**context** = context;  
}  
  
@Override  
**public int** getCount() {  
 **return mString**.size();  
}  
  
@Override  
**public** String getItem(**int** position){  
 **return mString**.get(position).toString();  
}  
  
@Override  
**public** View getView(**int** position, View convertView, ViewGroup parent){  
 convertView = **mInflater**.inflate(**mViewResourceId**, **null**);  
  
 TextView nom = (TextView) convertView.findViewById(R.id.***nomResUser***);  
 nom.setText(**mString**.get(position)[0]);  
  
 TextView dir = (TextView) convertView.findViewById(R.id.***dirResUser***);  
 dir.setText(**mString**.get(position)[1]);  
  
 Button btn = (Button) convertView.findViewById(R.id.***btn\_verResUser***);  
 btn.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 Toast.*makeText*(**context**, **"Dentro del click "** + **mString**.get(position)[2], Toast.***LENGTH\_SHORT***).show();  
 }  
 });  
  
 **return** convertView;  
}

#### Restaurantes para Jefes

@Override  
**protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_restaurantes\_para\_jefes***);  
  
 String name = **""**;  
 String email = **""**;  
 String uid = **""**;  
 **final** String[] message = {**"Valor inicial"**};  
 Uri photoUrl;  
  
 FirebaseUser user = FirebaseAuth.*getInstance*().getCurrentUser();  
 **if** (user != **null**) {  
 *// Name, email address, and profile photo Url* name = user.getDisplayName();  
 email = user.getEmail();  
 photoUrl = user.getPhotoUrl();  
  
 *// The user's ID, unique to the Firebase project. Do NOT use this value to  
 // authenticate with your backend server, if you have one. Use  
 // FirebaseUser.getIdToken() instead.* uid = user.getUid();  
 }  
  
 Log.*println*(Log.***DEBUG***, **"UUID"**, user.getUid());  
  
 updateData(name, email, uid, **"okay jefes"**);  
}  
  
**public void** updateData(String name, String email, String uid, String message){  
 TextView data = (TextView) findViewById(R.id.***data***);  
  
 data.setText(**"Nombre: "** + name + **"\nEmail: "** + email + **"\nUUID: "** + uid + **"\nMessage: "** + message);  
}  
  
**public void** logOut(View view){  
 DBHelper.*logOut*();  
 Intent intent = **new** Intent(**this**, MainActivity.**class**);  
 startActivity(intent);  
}

### Documentos XML

#### Activity login

<**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:id="@+id/container"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#000000"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 tools:context="LoginActivity"**>  
  
 <**Button  
 android:id="@+id/login"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="start"  
 android:layout\_marginStart="48dp"  
 android:layout\_marginTop="16dp"  
 android:layout\_marginEnd="48dp"  
 android:layout\_marginBottom="64dp"  
 android:onClick="signIn"  
 android:text="@string/action\_sign\_in"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/password"  
 app:layout\_constraintVertical\_bias="0.2"** />  
  
 <**EditText  
 android:id="@+id/password"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="24dp"  
 android:layout\_marginTop="8dp"  
 android:layout\_marginEnd="24dp"  
 android:hint="@string/prompt\_password"  
 android:imeActionLabel="@string/action\_sign\_in\_short"  
 android:imeOptions="actionDone"  
 android:inputType="textPassword"  
 android:selectAllOnFocus="true"  
 android:textColor="#7D9AD5"  
 android:textColorHint="#7D9AD5"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/username"** />  
  
 <**EditText  
 android:id="@+id/username"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="24dp"  
 android:layout\_marginTop="250dp"  
 android:layout\_marginEnd="24dp"  
 android:hint="@string/prompt\_email"  
 android:inputType="textEmailAddress"  
 android:selectAllOnFocus="true"  
 android:textColor="#7D9AD5"  
 android:textColorHint="#7D9AD5"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
 <**ProgressBar  
 android:id="@+id/loading"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:layout\_marginStart="32dp"  
 android:layout\_marginTop="64dp"  
 android:layout\_marginEnd="32dp"  
 android:layout\_marginBottom="64dp"  
 android:visibility="gone"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="@+id/password"  
 app:layout\_constraintStart\_toStartOf="@+id/password"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.3"** />  
  
 <**TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="100dp"  
 android:text="Log In"  
 android:textColor="#5EE7FD"  
 android:textSize="48sp"  
 app:layout\_constraintBottom\_toTopOf="@+id/username"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"** />  
</**androidx.constraintlayout.widget.ConstraintLayout**>

#### Restaurant user card

<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="300dp"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:background="@drawable/layout\_bg"**>  
  
 <**LinearLayout  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="9"  
 android:orientation="vertical"**>  
  
 <**TextView  
 android:id="@+id/nomResUser"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:gravity="center"  
 android:text="Nombre"  
 android:textColor="#000000"  
 android:textSize="14sp"** />  
  
 <**TextView  
 android:id="@+id/dirResUser"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:gravity="center"  
 android:text="Dirección"  
 android:textColor="#000000"  
 android:textSize="14sp"** />  
  
 </**LinearLayout**>  
  
 <**Button  
 android:id="@+id/btn\_verResUser"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="7.5dp"  
 android:layout\_weight="1"  
 android:text="Detalles"  
 android:background="#FF6200EE"  
 android:textColor="#FFFFFFFF"  
 android:textSize="14sp"** />  
  
</**LinearLayout**>

#### Activity main

<**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="#000000"  
 tools:context=".MainActivity"**>  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:gravity="center\_horizontal"  
 android:orientation="vertical"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 tools:context=".MainActivity"**>  
  
 <**ImageView  
 android:id="@+id/logoIS"  
 android:layout\_width="390dp"  
 android:layout\_height="wrap\_content"  
 app:srcCompat="@drawable/logo\_completo\_negro"** />  
  
 <**TextView  
 android:id="@+id/LabelIS"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="15dp"  
 android:contentDescription="Label de la página de inicio de sesión"  
 android:gravity="center"  
 android:text="Iniciar Sesión"  
 android:textColor="#2656C5"  
 android:textIsSelectable="false"  
 android:textSize="34sp"  
 android:textStyle="bold"  
 tools:text="Iniciar Sesión"** />  
  
 <**Button  
 android:id="@+id/btnISGoogle"  
 android:layout\_width="310dp"  
 android:layout\_height="wrap\_content"  
 android:text="Iniciar Sesión con Google"  
 android:textIsSelectable="false"  
 app:cornerRadius="20dp"  
 app:icon="@drawable/google\_logo"  
 app:iconSize="20dp"  
 tools:text="Iniciar Sesión con Google"** />  
  
 <**Space  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"** />  
  
 <**Button  
 android:id="@+id/btnISCorreo"  
 android:layout\_width="310dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="15dp"  
 android:onClick="goToMailLogin"  
 android:text="Iniciar sesion con correo"  
 app:cornerRadius="20dp"** />  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="horizontal"**>  
  
 <**TextView  
 android:id="@+id/LnkForgotPass"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:clickable="true"  
 android:focusable="true"  
 android:gravity="center"  
 android:text="¿Olvidó su contraseña?"  
 android:textColor="#55E5FD"** />  
  
 <**TextView  
 android:id="@+id/LnkRegister"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:clickable="true"  
 android:focusable="true"  
 android:gravity="center"  
 android:text="Registrarse"  
 android:textColor="#55E5FD"** />  
 </**LinearLayout**>  
  
 </**LinearLayout**>  
</**androidx.constraintlayout.widget.ConstraintLayout**>

#### Activity registrar res usuario

<**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="#000000"  
 tools:context=".RegistrarResUsuario"**>  
  
 <**Button  
 android:id="@+id/btn\_AggResUs"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="60dp"  
 android:text="Agregar"  
 android:onClick="regResUser"  
 app:cornerRadius="20dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/et\_codResUsu"** />  
  
 <**EditText  
 android:id="@+id/et\_codResUsu"  
 style="@style/Widget.AppCompat.AutoCompleteTextView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="48dp"  
 android:background="#3C26C6DA"  
 android:ems="15"  
 android:gravity="center\_horizontal|center\_vertical"  
 android:hint="Código"  
 android:inputType="number"  
 android:textColor="#42A5F5"  
 android:textColorHint="#26C6DA"  
 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"** />  
  
 <**TextView  
 android:id="@+id/textView3"  
 android:layout\_width="250dp"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="Agregar\nrestaurante:"  
 android:textColor="#AB47BC"  
 android:textSize="42sp"  
 android:textStyle="bold"  
 app:layout\_constraintBottom\_toTopOf="@+id/et\_codResUsu"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
</**androidx.constraintlayout.widget.ConstraintLayout**>

#### Activity restaurantes

<**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="#000000"  
 tools:context=".Restaurantes"**>  
  
  
 <**Button  
 android:id="@+id/buttonReg"  
 android:layout\_width="225dp"  
 android:layout\_height="56dp"  
 android:layout\_marginStart="24dp"  
 android:layout\_marginLeft="24dp"  
 android:layout\_marginEnd="33dp"  
 android:layout\_marginRight="33dp"  
 android:layout\_marginBottom="24dp"  
 android:onClick="goToRegister"  
 android:text="Registrar Restaurante"  
 android:background="#FF6200EE"  
 android:textColor="#FFFFFF"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toStartOf="@+id/button"  
 app:layout\_constraintStart\_toStartOf="parent"** />  
  
 <**Button  
 android:id="@+id/button"  
 android:layout\_width="85dp"  
 android:layout\_height="56dp"  
 android:layout\_marginEnd="24dp"  
 android:layout\_marginRight="24dp"  
 android:layout\_marginBottom="24dp"  
 android:onClick="logOut"  
 android:text="Log Out"  
 android:background="#FF6200EE"  
 android:textColor="#FFFFFF"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"** />  
  
 <**ListView  
 android:id="@android:id/list"  
 android:layout\_width="0dp"  
 android:layout\_height="0dp"  
 android:layout\_marginStart="24dp"  
 android:layout\_marginLeft="24dp"  
 android:layout\_marginTop="16dp"  
 android:layout\_marginEnd="24dp"  
 android:layout\_marginRight="24dp"  
 android:layout\_marginBottom="16dp"  
 android:divider="@android:color/transparent"  
 android:dividerHeight="7.5dp"  
 app:layout\_constraintBottom\_toTopOf="@+id/button"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/lbl\_restaurantes"** />  
  
 <**TextView  
 android:id="@+id/lbl\_restaurantes"  
 android:layout\_width="123dp"  
 android:layout\_height="27dp"  
 android:layout\_marginStart="24dp"  
 android:layout\_marginLeft="24dp"  
 android:layout\_marginTop="20dp"  
 android:layout\_marginEnd="309dp"  
 android:layout\_marginRight="309dp"  
 android:text="Restaurantes:"  
 android:textColor="#03A9F4"  
 android:textColorHighlight="#009688"  
 android:textColorHint="#FFEB3B"  
 android:textSize="20sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
</**androidx.constraintlayout.widget.ConstraintLayout**>

#### Activity restaurantes para jefes

<**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"  
 tools:context=".RestaurantesParaJefes"**>  
  
 <**TextView  
 android:id="@+id/textView2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="50dp"  
 android:layout\_marginLeft="50dp"  
 android:layout\_marginTop="50dp"  
 android:text="User data de un jefe:"  
 android:textSize="24sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"** />  
  
 <**TextView  
 android:id="@+id/data"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="50dp"  
 android:layout\_marginLeft="50dp"  
 android:layout\_marginTop="20dp"  
 android:text="data"  
 android:textSize="16sp"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView2"** />  
  
 <**Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="300dp"  
 android:onClick="logOut"  
 android:text="Log Out"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"** />  
  
</**androidx.constraintlayout.widget.ConstraintLayout**>

## Código Arduino

### Código ESPE8266

|  |
| --- |
| #include <ESP8266WiFi.h> |
|  | #include <FirebaseESP8266.h> |
|  | #define FIREBASE\_HOST "analizadoraire-default-rtdb.firebaseio.com" |
|  | #define FIREBASE\_AUTH "fg5qwePSPATyMEn5OPm6y1nkTyXB6lyCwSrM9UuJ" |
|  | #include "DHT.h" |
|  | #define DHTTYPE DHT11 |
|  | #define DHTpin 14 |
|  | FirebaseData firebaseData; |
|  | const char\* ssid = "NETLIFE-BARATAU"; |
|  | const char\* contraseña = "Antonio68"; |
|  | float temperatura = 0; |
|  | float humedad = 0; |
|  | float calidad = 0; |
|  | String mac = "50:02:91:EC:B8:6B"; |
|  | bool isRegistered = false; |
|  | bool alarma = false; |
|  | float bateria = 100; |
|  | String regCode = "98765432"; |
|  | DHT dht(DHTpin, DHTTYPE); //iniciar dht11 |
|  |  |
|  | void setup() { |
|  | // put your setup code here, to run once: |
|  | Serial.begin(115200); |
|  | dht.begin(); |
|  | /\*while (!Serial) { |
|  | ; // wait for serial port to connect. Needed for native USB port only |
|  | }\*/ |
|  | WiFi.begin(ssid, contraseña); |
|  | WiFi.begin(ssid, contraseña); |
|  | while (WiFi.status() != WL\_CONNECTED) { |
|  | while (WiFi.status() != WL\_CONNECTED) { |
|  | Serial.print("."); |
|  | delay(500); |
|  | } |
|  | Serial.print(" "); |
|  | Serial.print("IP: "); |
|  | Serial.println(WiFi.localIP()); |
|  | Firebase.begin(FIREBASE\_HOST, FIREBASE\_AUTH); |
|  | Firebase.setString(firebaseData, regCode+ "/regCode", regCode); |
|  | } |
|  | } |
|  |  |
|  | void loop() { |
|  | // put your main code here, to run repeatedly: |
|  | temperatura = dht.readTemperature(); |
|  | humedad = dht.readHumidity(); |
|  | calidad = analogRead(A0); |
|  | //Advertencias |
|  | if (calidad < 181) { |
|  | alarma = false; |
|  | } else if (calidad >= 181 && calidad < 225) { |
|  | alarma = false; |
|  | } else if (calidad >= 225 && calidad < 300) { |
|  | alarma = true; |
|  | } else if (calidad >= 300 && calidad < 350) { |
|  | alarma = true; |
|  | } else if (calidad |
|  | >= 350) { |
|  | alarma = true; |
|  | } |
|  |  |
|  | if (temperatura < 22) { |
|  | alarma = false; |
|  | } else if (temperatura >= 22 && temperatura < 24) { |
|  | alarma = false; |
|  | } else if (temperatura >= 24 && temperatura < 25) { |
|  | alarma = false; |
|  | } else if (temperatura >= 25 && temperatura < 27) { |
|  | alarma = false; |
|  | } else if (temperatura > 27) { |
|  | alarma = true; |
|  | } |
|  |  |
|  | if (humedad >= 50) { |
|  | alarma = true; |
|  | } else if (humedad >= 33 && humedad < 40) { |
|  | alarma = false; |
|  | } |
|  | Firebase.setFloat(firebaseData, regCode + "/temperatura", temperatura); |
|  | Firebase.setFloat(firebaseData, regCode +"/humedad", humedad); |
|  | Firebase.setFloat(firebaseData, regCode +"/calidad", calidad); |
|  | Firebase.setFloat(firebaseData, regCode +"/bateria", bateria); |
|  | Firebase.setBool(firebaseData, regCode +"/isRegistered", isRegistered); |
|  | Firebase.setBool(firebaseData, regCode +"/alarma", alarma); |
|  | delay(5000); |
|  | } |

### Código CalidadDeAire

|  |
| --- |
| #include "DHT.h" |
|  | #include <SoftwareSerial.h> |
|  | #define DHTPIN 2 |
|  | #define DHTTYPE DHT11 // DHT 11 |
|  | #define LEDVERDE 5 |
|  | #define LEDROJO 6 |
|  | #define PILA 1 |
|  | int analogValor; |
|  | float voltaje = 0; |
|  | float minimo = 2.7; |
|  | SoftwareSerial espSerial(5, 6);//Definición pines de serial |
|  | DHT dht(DHTPIN, DHTTYPE); |
|  | String str; |
|  | void setup() { |
|  | Serial.begin(115200); |
|  | espSerial.begin(115200); |
|  | dht.begin(); |
|  | pinMode(LEDVERDE, OUTPUT); |
|  | pinMode(LEDROJO, OUTPUT); |
|  | delay(2000); |
|  | } |
|  | void loop(){ |
|  | //Lecturas de sensores |
|  | float h = dht.readHumidity(); |
|  | float t = dht.readTemperature(); |
|  | float calidad = analogRead(A0); |
|  | str = String(h) + String(";") + String(t)+ String(";") + String(calidad);//Envío de info a ESP |
|  |  |
|  | analogValor = analogRead(PILA); |
|  | voltaje = 0.0048 \* analogValor; |
|  |  |
|  | /\*Codigo para leer batería\*/ |
|  | if (voltaje >= minimo) { |
|  | digitalWrite (LEDVERDE, HIGH); |
|  | digitalWrite (LEDROJO, LOW); |
|  | } else { |
|  | digitalWrite (LEDROJO, HIGH); |
|  | digitalWrite (LEDVERDE, LOW); |
|  | }delay(4000); |
|  |  |
|  | } |