

# TUTORIAL ANDROID

---

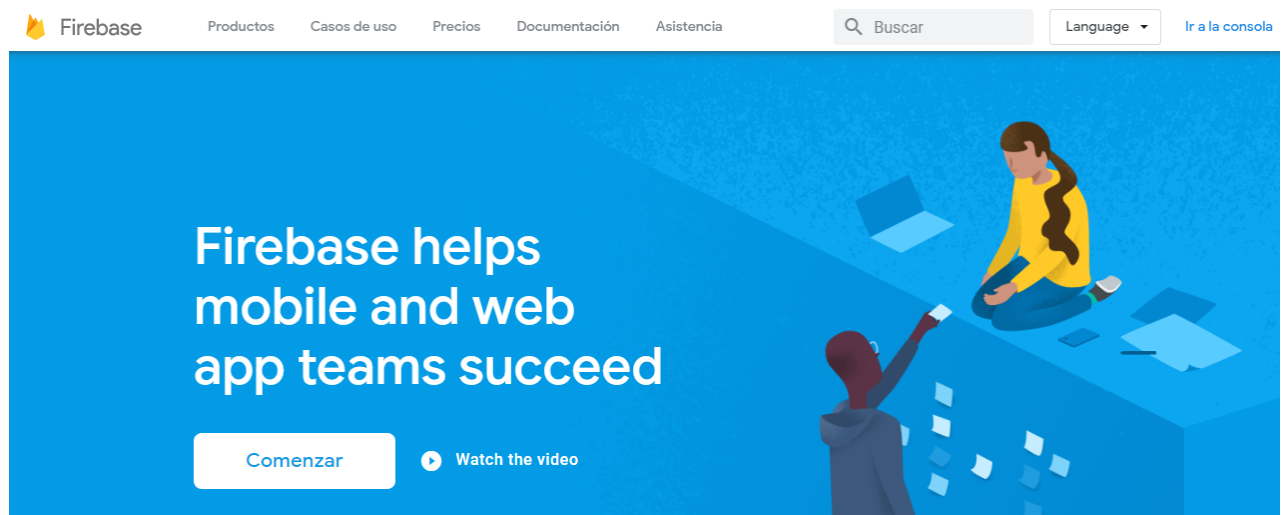
Daniela Hernández Valenzuela

Rubén Raúl Hernández Mac-Kinley

## CONEXIÓN FIREBASE ANDROID

---

1. Registrarse en [firebase](#)
2. Presionar "Ir a la consola"



3. Presionar "Añadir proyecto"

# ¡Te damos la bienvenida Firebase!

Herramientas de Google para desarrollar una  
espectaculares, interactuar con los usuarios  
través de los anuncios para móviles.

 [Más información](#)  [Documentación](#)  [As](#)

Proyectos recientes



Añadir proyecto



[Ver un proyecto de demostración](#)

4. Ingresa el nombre del proyecto, acepta los terminos y condiciones y presiona "Crear proyecto"

## Añadir un proyecto

Nombre del proyecto

Tutorial

📱 + iOS + </>

**Nota:** Los proyectos abarcan las aplicaciones de distintas plataformas

ID del proyecto

tutorial-ebffd

Ubicaciones

Estados Unidos (Analytics)  
nam5 (us-central) (Cloud Firestore)

☒ Usar la configuración predeterminada para compartir datos de Google Analytics for Firebase

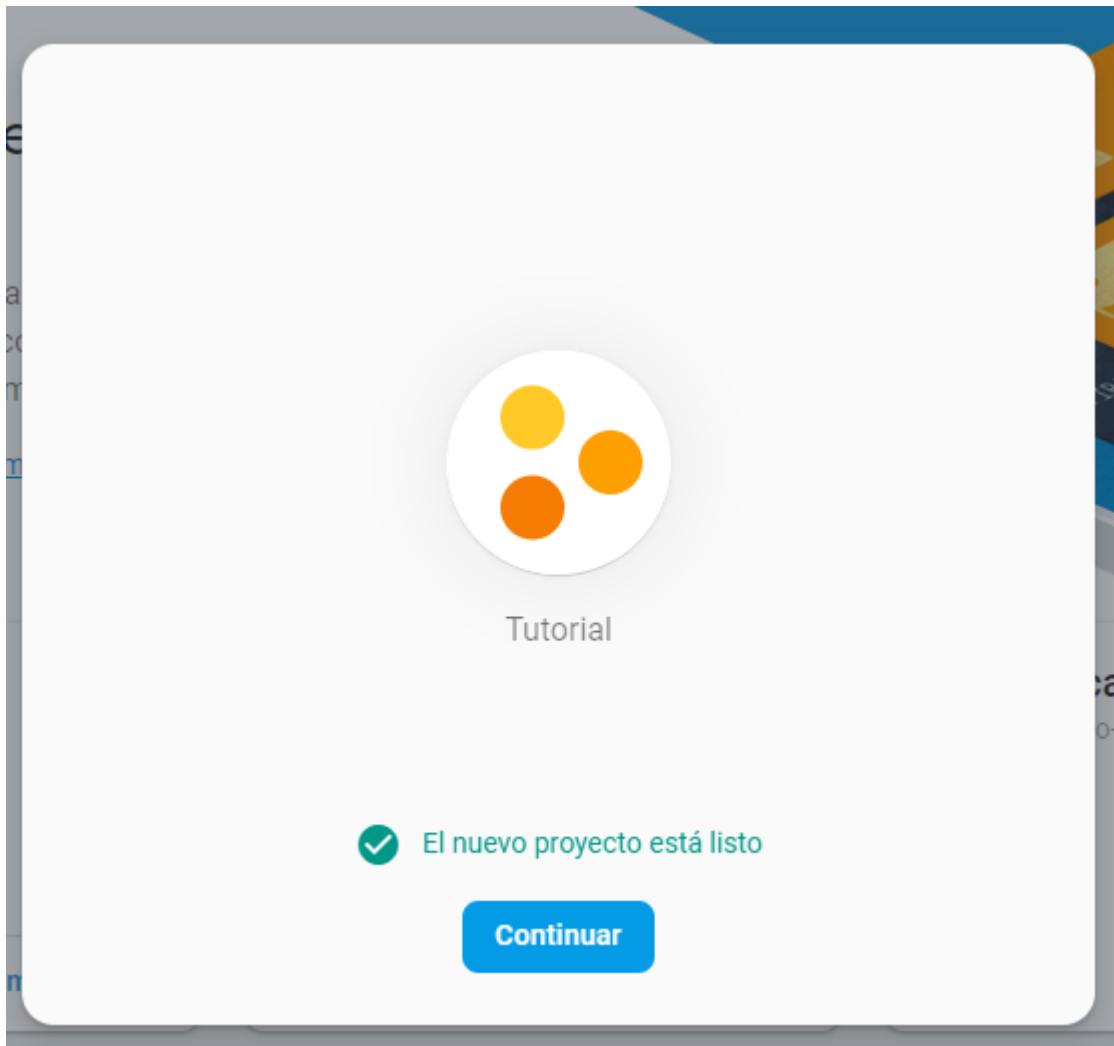
✓ Compartir tus datos de Analytics con todas las funciones de Firebase

✓ Comparte tus datos de Analytics con nosotros para ayudarnos a mejorar nuestros productos y servicios.

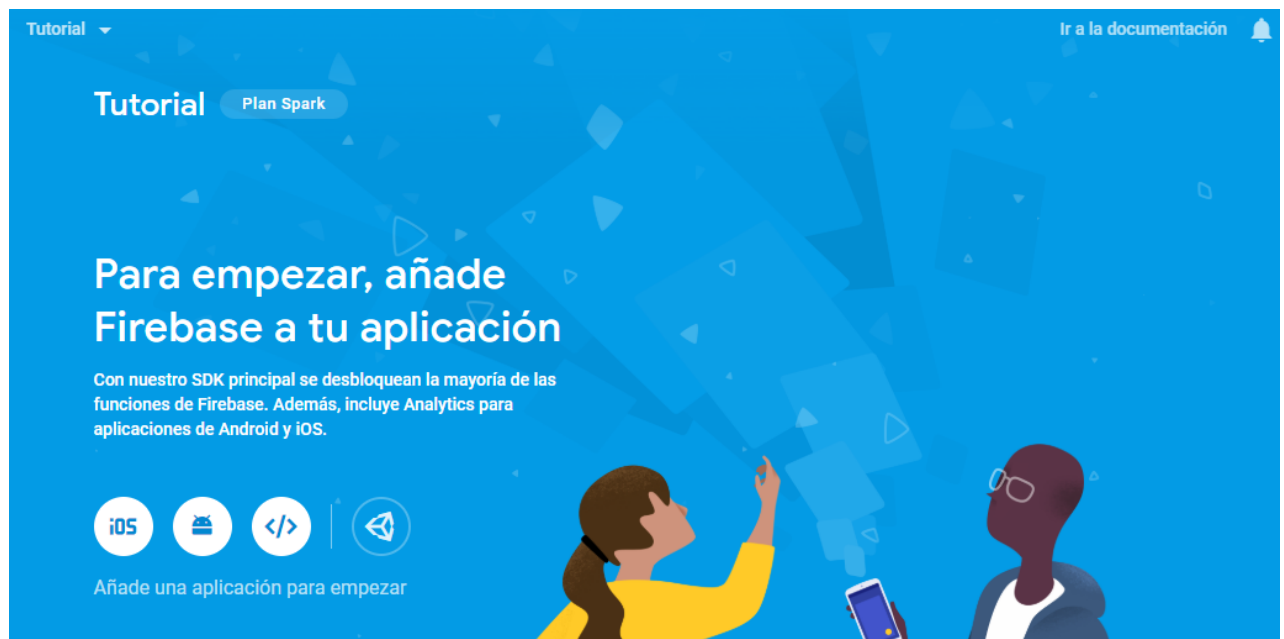
✓ Comparte tus datos de Analytics con nosotros para habilitar el servicio de asistencia técnica.

✓ Comparte tus datos de Analytics con nosotros para habilitar las

5. Presiona "Continuar" y se abrirá la consola del proyecto Firebase



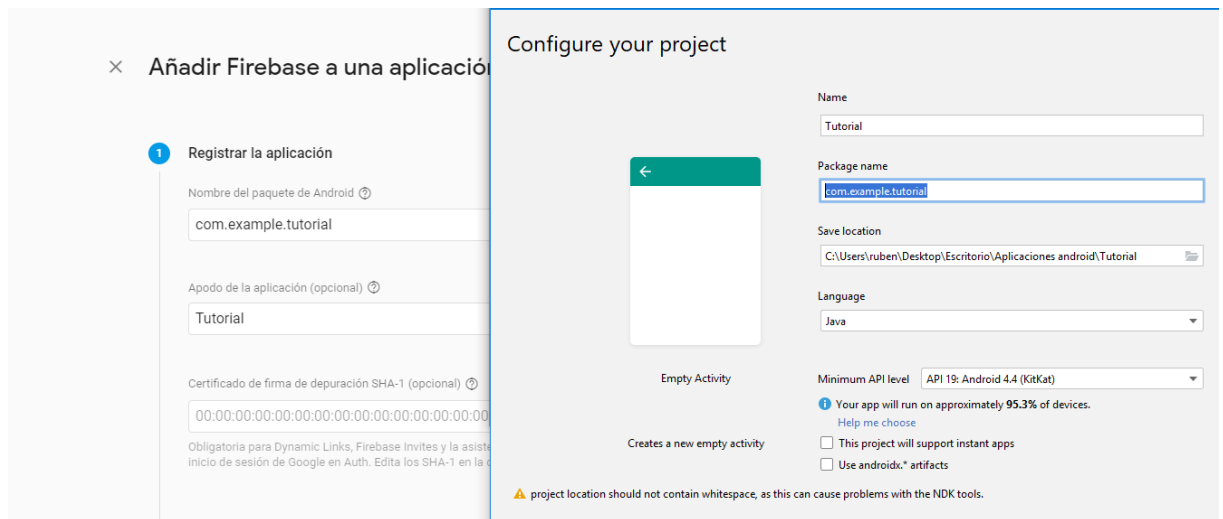
6. Presiona el logotipo de Android



7. Vincular firebase con el proyecto

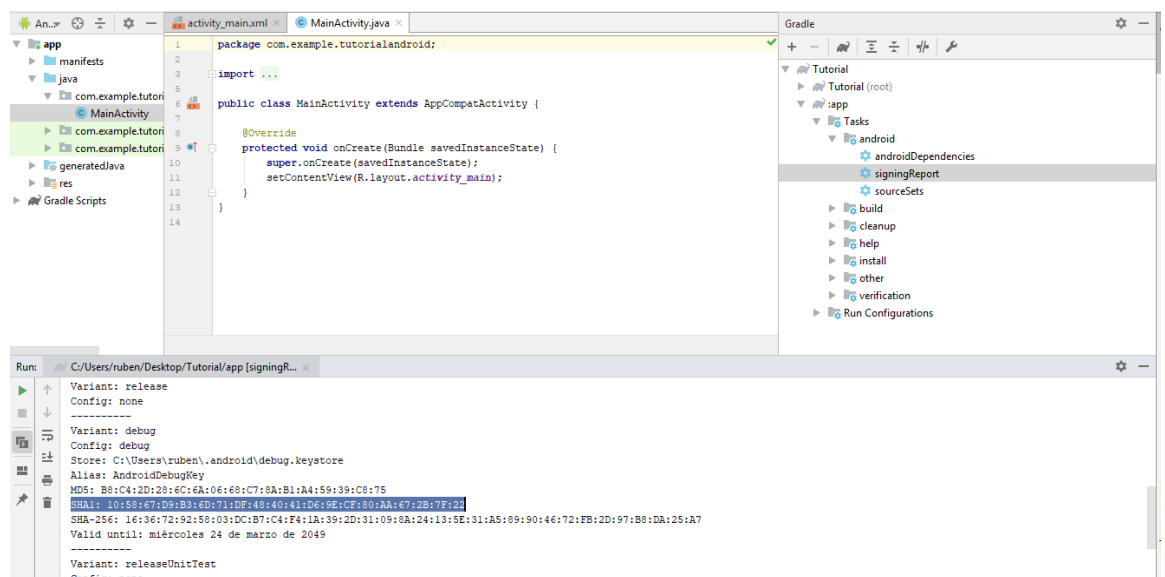
1. Crear un proyecto en Android Studio

2. Copiar el nombre del package name del proyecto de Android Studio y pegarlo en el nombre del paquete de android que solicita Firebase



3. A partir del Proyecto creado en Android Studio obtener el SHA

1. Ir a gradle en android studio
2. Ir a la carpeta :app
3. Ir a la carpeta "Task"
4. Ir a la carpeta "android"
5. Ir al archivo "signingReport"
6. Copiar SHA1 y pegarlo en el Firebase en donde dice "Certificado de firma de depuración SHA1"



4. En firebase, click registrar aplicación
8. Descargar el archivo "google-services.json" y sigue las instrucciones brindadas por google (Se muestran en la imagen)

[Descargar google-services.json](#)

Cambia a la vista **Proyecto** de Android Studio para ver el directorio "root" de tu proyecto.

Mueve el archivo google-services.json que acabas de descargar al directorio "root" del módulo de la aplicación de Android.



  
google-services.json

[Anterior](#) [Siguiente](#)

9. Presiona siguiente

10. Añadir el SDK de Firebase

1. Modificar los archivos build.gradle (Project) y build.gradle (Module)

1. build.gradle (Project)

```
buildscript {  
    dependencies {  
        // Add this line  
        classpath 'com.google.gms:google-services:4.2.0'  
    }  
}
```

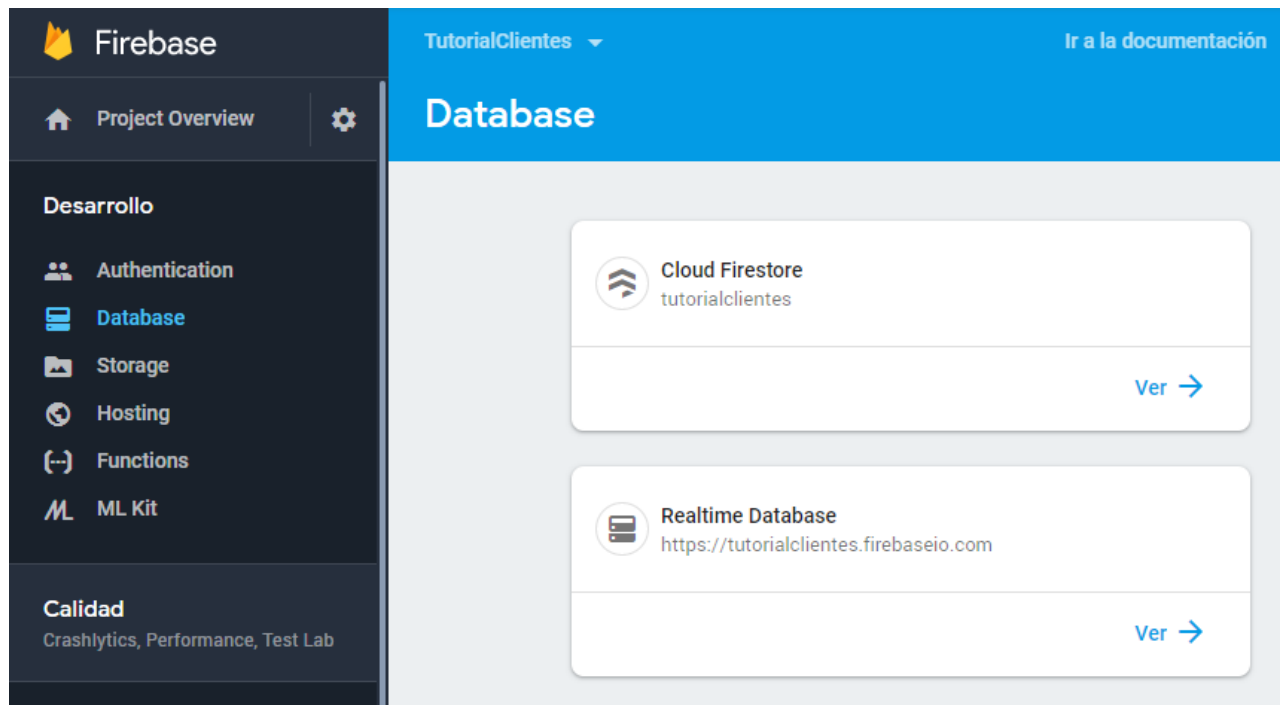
2. build.gradle (Module)

```
dependencies {  
    // Add this line  
    implementation 'com.google.firebase:firebase-core:16.0.9'  
}  
...  
// Add to the bottom of the file  
apply plugin: 'com.google.gms.google-services'
```

11. En firebase, presionar siguiente

12. En firebase, presiona "Saltar este paso"

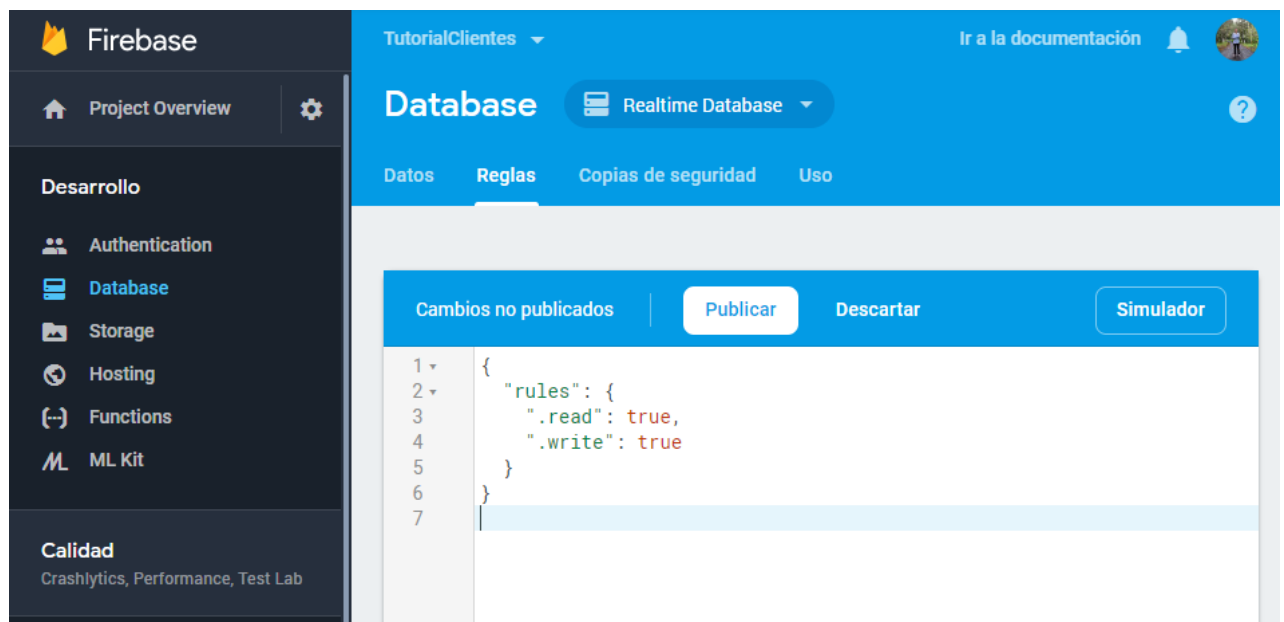
13. Dentro de firebase seleccionar la opción "Database" y después seleccionar "Realtime Database"



14. Dentro ir a la pestaña de "Reglas" y pegar el siguiente código

```
{
  "rules": {
    ".read": true,
    ".write": true
  }
}
```

15. Debería quedar de esta manera



16. Después de clic en "Publicar" esto establece que cualquiera pueda acceder a nuestra base de datos

## CRUD DE CLIENTES

Antes de comenzar nuestro proyecto se llamara TutorialClientes. Por lo cual donde encuentres una referencia a nuestro proyecto deberias cambiarlo por el nombre del tuyo.

## PERMISOS NECESARIOS PARA INTERNET ANDROID

1. Dentro del proyecto en android ir al archivo "AndroidManifest.xml" en la carpeta manifests (ruta de la carpeta app > manifests).
2. Agregar la siguiente linea de codigo

```
<uses-permission android:name="android.permission.INTERNET" />
```

3. Deberia quedar algo como el siguiente codigo

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.tutorial.tutorialclientes">

    <uses-permission android:name="android.permission.INTERNET" />
    .....
</manifest>
```

## TOOLBAR

1. Crear un layout llamado "toolbar" en la carpeta layout (ruta de la carpeta: res > layout)

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.Toolbar
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:theme="@style/AppTheme"
    android:background="@color/colorPrimary"
    app:titleTextColor="@android:color/white"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

</android.support.v7.widget.Toolbar>
```

## CREAR UN CLIENTE

1. Crear una clase llamada conexión y pega el siguiente código (importa la librerías necesarias)

```
package com.tutorial.tutorialclientes;

import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
```



```

public class Conexion {

    DatabaseReference databaseReference;

    public DatabaseReference conexion(){
        databaseReference = FirebaseDatabase.getInstance().getReference();
        return databaseReference;
    }
}

```

2. Crear un layout llamado "activity\_registrar\_cliente" en la carpeta layout (ruta de la carpeta: res > layout)

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView 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/scrollView2"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fillViewport="true"
    tools:context=".RegistrarCliente">

    <android.support.constraint.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <include
            android:id="@+id/toolbar"
            layout="@layout/toolbar"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />

        <android.support.design.widget.TextInputLayout
            android:id="@+id/textInputLayout5"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="50dp"
            android:layout_marginTop="50dp"
            android:layout_marginEnd="50dp"
            android:gravity="center"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.0"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/toolbar">

            <EditText
                android:id="@+id/edt_nombre_registrar"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"

```

```
        android:drawablePadding="5dp"
        android:hint="Nombre del cliente"
        android:inputType="text" />
```

```
</android.support.design.widget.TextInputLayout>
```

```
<android.support.design.widget.TextInputLayout
    android:id="@+id/textInputLayout3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="50dp"
    android:layout_marginTop="50dp"
    android:layout_marginEnd="50dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textInputLayout5">
```

```
    <EditText
```

```
        android:id="@+id/edt_correo_registrar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:drawablePadding="5dp"
        android:hint="Correo electronico"
        android:inputType="text" />
```

```
</android.support.design.widget.TextInputLayout>
```

```
<android.support.design.widget.TextInputLayout
    android:id="@+id/textInputLayout2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="50dp"
    android:layout_marginTop="50dp"
    android:layout_marginEnd="50dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textInputLayout3">
```

```
    <EditText
```

```
        android:id="@+id/edt_telefono_registrar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:drawablePadding="5dp"
        android:hint="@string/edt_telefono"
        android:inputType="number" />
```

```
</android.support.design.widget.TextInputLayout>
```

```
<android.support.design.widget.TextInputLayout
    android:id="@+id/textInputLayout10"
    android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:layout_marginStart="50dp"
        android:layout_marginTop="50dp"
        android:layout_marginEnd="50dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textInputLayout2">

        <EditText
            android:id="@+id/edt_direccion_registrar"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:drawablePadding="5dp"
            android:hint="@string/edt_direccion"
            android:inputType="number" />

    </android.support.design.widget.TextInputLayout>

    <Button
        android:id="@+id/btn_registrar_automovil"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="50dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="50dp"
        android:layout_marginBottom="8dp"
        android:onClick="btn_agregar_cliente"
        android:text="Agregar cliente"
        android:textSize="18sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textInputLayout10" />

    </android.support.constraint.ConstraintLayout>
</ScrollView>

```

### 3. Crear una clase llamada "Cliente"

```

package com.tutorial.tutorialclientes;

public class Cliente {
    public String nombre;
    public String correo;
    public String direccion;
    public String telefono;

    public Cliente() {
        super();
    }
}

```

```

    public Cliente(String nombre, String correo, String direccion, String telefono) {
        this.nombre = nombre;
        this.correo = correo;
        this.direccion = direccion;
        this.telefono = telefono;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public String getCorreo() {
        return correo;
    }

    public void setCorreo(String correo) {
        this.correo = correo;
    }

    public String getDireccion() {
        return direccion;
    }

    public void setDireccion(String direccion) {
        this.direccion = direccion;
    }

    public String getTelefono() {
        return telefono;
    }

    public void setTelefono(String telefono) {
        this.telefono = telefono;
    }
}

```

4. Crear una nueva clase llamada "RegistrarCliente" (Esta clase estara asociada a tu layout "activity\_registrar\_cliente") y pega el siguiente código

```

package com.tutorial.tutorialclientes;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.text.TextUtils;
import android.view.MenuItem;
import android.view.View;

```

```

import android.widget.EditText;
import android.widget.Toast;

import com.google.firebase.database.DatabaseReference;

public class RegistrarCliente extends AppCompatActivity {

    //Creando variable de referencia para la base de datos (Firebase)
    DatabaseReference firebase;
    //Llamando a la clase conexion para la creacion de la referencia
    Conexion conexion = new Conexion();

    EditText edt_nombre_registrar, edt_correo_registrar, edt_telefono_registrar,
    edt_direccion_registrar;

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

        //Una vez se inicia la actividad se manda a llamar a firebase
        firebase = conexion.conexion();

        //Toolbar
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        getSupportActionBar().setTitle("Registrar cliente");
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        getSupportActionBar().setDisplayShowHomeEnabled(true);

        //Inicializacion de los componentes que ocuparemos
        edt_nombre_registrar = findViewById(R.id.edt_nombre_registrar);
        edt_correo_registrar = findViewById(R.id.edt_correo_registrar);
        edt_telefono_registrar = findViewById(R.id.edt_telefono_registrar);
        edt_direccion_registrar = findViewById(R.id.edt_direccion_registrar);
    }

    //Metodo para agregar un nuevo cliente a firebase (btn_agregar_cliente)
    public void btn_agregar_cliente (View view){
        //Obtencion de lo que escribe el usuario en la activity
        String nombre = edt_nombre_registrar.getText().toString();
        String correo = edt_correo_registrar.getText().toString();
        String telefono = edt_telefono_registrar.getText().toString();
        String direccion = edt_direccion_registrar.getText().toString();

        //Verificacion de cada uno de los campos
        if (TextUtils.isEmpty(nombre)) {
            Toast.makeText(getApplicationContext(), "Ingresa el nombre",
            Toast.LENGTH_SHORT).show();
        }else if (TextUtils.isEmpty(correo)) {
            Toast.makeText(getApplicationContext(), "Ingresa el correo electrónico",
            Toast.LENGTH_SHORT).show();
        }else if (TextUtils.isEmpty(telefono)) {

```

```

        Toast.makeText(getApplicationContext(), "Ingresa el teléfono",
Toast.LENGTH_SHORT).show();
    }else if (TextUtils.isEmpty(direccion)) {
        Toast.makeText(getApplicationContext(), "Ingresa la dirección",
Toast.LENGTH_SHORT).show();
    }else if (!TextUtils.isEmpty(nombre) && !TextUtils.isEmpty(correo) &&
!TextUtils.isEmpty(telefono)) {
        //Creacion de un nuevo cliente mediante los parametros obtenidos
        Cliente cliente = new Cliente(nombre,correo,telefono,direccion);
        //nombreReferenciaFirebase.nodoHijo.nodoHijo.setValue(Valor)
        //Esto se guarda en la base de datos es decir decimos que en la referencia
en firebase
        //Guarde en cliente un hijo llamado nombre con el valor de el cliente que
estamos creando
        firebase.child("Cliente").child(nombre).setValue(cliente);
        Toast.makeText(getApplicationContext(),"Cliente agregado
correctamente",Toast.LENGTH_SHORT).show();
        finish();
    }
}

//Boton de regreso en el toolbar
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()){
        case android.R.id.home:
            Intent intent = new Intent(RegistrarCliente.this,
visualizarClientes.class);
            startActivity(intent);
            finish();
            return true;

        default:
            return super.onOptionsItemSelected(item);
    }
}
}
}

```

5. Con esto podriamos dar de alta clientes dentro de firebase.

## VISUALIZAR LOS CLIENTES

1. Crear un layout llamado "activity\_visualizar\_clientes" en la carpeta layout (ruta de la carpeta: res > layout)

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.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=".VisualizarClientes">

```

```
<include
    android:id="@+id/toolbar"
    layout="@layout/toolbar"
    android:layout_width="0dp"
    android:layout_height="0dp"
    app:layout_constraintBottom_toTopOf="@+id/relativeLayout3"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

```
<android.support.constraint.Guideline
    android:id="@+id/linea1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintGuide_percent="0.08" />
```

```
<RelativeLayout
    android:id="@+id/relativeLayout3"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:orientation="vertical"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/linea1">
```

```
<android.support.v7.widget.RecyclerView
    android:id="@+id/rcv_visualizar_clientes"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentStart="true"
    android:layout_alignParentEnd="true"
    android:layout_marginStart="20dp"
    android:layout_marginTop="10dp"
    android:layout_marginEnd="20dp">
```

```
</android.support.v7.widget.RecyclerView>
```

```
<android.support.design.widget.FloatingActionButton
    android:id="@+id/btn_float_registrar_clientes"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"
    android:layout_alignParentBottom="true"
    android:layout_marginRight="15dp"
    android:layout_marginBottom="15dp"
    android:onClick="btn_float_registrar_clientes"
    android:src="@drawable/ic_add_"
    app:backgroundTint="@color/colorPrimary" />
```

```
</RelativeLayout>
```

```
</android.support.constraint.ConstraintLayout>
```

2. Crear una nueva clase llamada "VisualizarClientes " (Esta clase estara asociada a tu layout "activity\_visualizar\_clientes") y pega el siguiente código

```
package com.tutorial.tutorialclientes;

import android.content.Intent;
import android.support.annotation.NonNull;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.support.v7.widget.Toolbar;
import android.view.View;

import com.firebase.ui.database.FirebaseRecyclerAdapter;
import com.google.firebase.database.DatabaseReference;

public class VisualizarClientes extends AppCompatActivity {

    //Creando variable de referencia para la base de datos (Firebase)
    Conexion conexion = new Conexion();
    //Llamando a la clase conexion para la creacion de la referencia
    DatabaseReference firebase;

    RecyclerView rcv_visualizar_clientes;
    FloatingActionButton btn_float_registrar_clientes;
    FirebaseRecyclerAdapter<Cliente, AdaptadorCliente.ViewHolder> adapter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_visualizar_clientes);
        //Conexion a firebase
        firebase = conexion.conexion();

        //Toolbar
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        getSupportActionBar().setTitle("Lista de clientes");

        rcv_visualizar_clientes = findViewById(R.id.rcv_visualizar_clientes);
        btn_float_registrar_clientes = findViewById(R.id.btn_float_registrar_clientes);

        //Setear al linerlayaour manager nuestro reciclerView
        LinearLayoutManager layoutManager = new LinearLayoutManager(this);
        layoutManager.setOrientation(LinearLayoutManager.VERTICAL);
        rcv_visualizar_clientes.setLayoutManager(layoutManager);

        //FirebaseUI definido para llamar de nuestro firebase la lista
```



```

        adapter = new FirebaseRecyclerAdapter<Cliente, AdaptadorCliente.ViewHolder>(
            /*Clase que utilizaremos*/Cliente.class,
            /*La interfaz grafica*/R.layout.cliente,
            /*ViewHolder archivo ClienteAdapter*/AdaptadorCliente.ViewHolder.class,
            /*La referencia de firebase donde buscara*/firebase.child("Cliente")
        ) {
            @Override
            protected void populateViewHolder(final AdaptadorCliente.ViewHolder
viewHolder, Cliente model, final int position) {
                viewHolder.txv_cliente_nombre.setText(model.getNombre());
                viewHolder.txv_cliente_correo.setText(model.getCorreo());
                viewHolder.txv_cliente_telefono.setText(model.getTelefono());
                viewHolder.txv_cliente_direccion.setText(model.getDireccion());
                viewHolder.cdv_cliente.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View v) {
                        Intent intent = new Intent(VisualizarClientes.this,
ConsultarCliente.class);

                        intent.putExtra("nombre",viewHolder.txv_cliente_nombre.getText().toString());

                        intent.putExtra("correo",viewHolder.txv_cliente_correo.getText().toString());

                        intent.putExtra("telefono",viewHolder.txv_cliente_telefono.getText().toString());

                        intent.putExtra("direccion",viewHolder.txv_cliente_direccion.getText().toString());
                        startActivity(intent);
                    }
                });
            }
        };

        rcv_visualizar_clientes.setAdapter(adapter);

        //Scroll para la funcion del boton flotante
        rcv_visualizar_clientes.addOnScrollListener(new RecyclerView.OnScrollListener()
{
            @Override
            public void onScrollStateChanged(@NonNull RecyclerView recyclerView, int
newState) {
                super.onScrollStateChanged(recyclerView, newState);
                if(newState == RecyclerView.SCROLL_STATE_IDLE){
                    btn_float_registrar_clientes.show();
                }
            }

            @Override
            public void onScrolled(@NonNull RecyclerView recyclerView, int dx, int dy) {
                super.onScrolled(recyclerView, dx, dy);
                if(dy>0 || dy<0 && btn_float_registrar_clientes.isShown()){
                    btn_float_registrar_clientes.hide();
                }
            }
        }
    }
}

```

```

    });
}

//Funcion del boton flotante
public void btn_float_registrar_clientes (View view){
    Intent intent = new Intent(VisualizarClientes.this, RegistrarCliente.class);
    startActivity(intent);
}
}

```

3. Para la visualizar de los datos necesitaremos un adaptador que nos ayudara con el trabajo de la informacion de los datos de firebase en este caso. Por lo cual debemos crear una clase llamada "AdaptadorCliente" que tendra el siguiente codigo

```

package com.tutorial.tutorialclientes;

import android.content.Context;
import android.support.annotation.NonNull;
import android.support.v7.widget.CardView;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import java.util.List;

public class AdaptadorCliente extends RecyclerView.Adapter<AdaptadorCliente.ViewHolder>
{

    List<Cliente> cliente;
    static Context context;

    public static class ViewHolder extends RecyclerView.ViewHolder{

        public CardView cdv_cliente;
        public TextView txv_cliente_nombre, txv_cliente_correo, txv_cliente_direccion,
txv_cliente_telefono;

        public ViewHolder(View view){
            super(view);
            cdv_cliente = view.findViewById(R.id.cdv_cliente);
            txv_cliente_nombre = view.findViewById(R.id.txv_cliente_nombre);
            txv_cliente_correo = view.findViewById(R.id.txv_cliente_correo);
            txv_cliente_direccion = view.findViewById(R.id.txv_cliente_direccion);
            txv_cliente_telefono = view.findViewById(R.id.txv_cliente_telefono);
        }
    }

    public AdaptadorCliente(Context context, List<Cliente> cliente){
        this.context = context;
        this.cliente = cliente;
    }
}

```

```

@NonNull
@Override
public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int i) {
    View view =
LayoutInflater.from(parent.getContext()).inflate(R.layout.cliente,parent,false);
    ViewHolder viewHolder = new ViewHolder(view);
    return viewHolder;
}

@Override
public void onBindViewHolder(@NonNull ViewHolder viewHolder, int i) {
}

@Override
public int getItemCount() {
    return cliente.size();
}
}

```

4. Una vez creada la clase anterior deberemos crear un layout donde mostraremos la informacion que estara recibiendo nuestro adaptador para esto creamos un layout llamado "cliente" que nos ayudara a mostrarlo mediante un cardview, el codigo de nuestro layout debera quedar de la siguiente manera

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.CardView
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/cdv_cliente"
android:layout_margin="10dp"
app:cardCornerRadius="20dp"
android:layout_width="match_parent"
android:layout_height="wrap_content">

<android.support.constraint.ConstraintLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:paddingTop="20dp"
    android:paddingBottom="20dp">

    <TextView
        android:id="@+id/txv_cliente_nombre"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:fontFamily="sans-serif"
        android:gravity="left"
        android:text="nombre"
        android:textColor="@android:color/black"
        android:textSize="18sp"

```

```
android:textStyle="bold"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent" />
```

```
<TextView
    android:id="@+id/txv_cliente_correo"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="32dp"
    android:layout_marginTop="16dp"
    android:layout_marginEnd="16dp"
    android:drawablePadding="10dp"
    android:gravity="center_vertical"
    android:text="correo"
    android:textSize="14sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/txv_cliente_nombre" />
```

```
<TextView
    android:id="@+id/txv_cliente_direccion"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="32dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="16dp"
    android:drawablePadding="10dp"
    android:gravity="center_vertical"
    android:text="direccion"
    android:textSize="14sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/txv_cliente_correo" />
```

```
<TextView
    android:id="@+id/txv_cliente_telefono"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="32dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="16dp"
    android:layout_marginBottom="8dp"
    android:drawablePadding="10dp"
    android:gravity="center_vertical"
    android:text="telefono"
    android:textSize="14sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/txv_cliente_direccion" />
```

```
</android.support.constraint.ConstraintLayout>
```

```
</android.support.v7.widget.CardView>
```

5. El funcionamiento de esto es que firebaseUI que nos ayuda a obtener informacion de la base de datos. Posteriormente se ingresa la informacion a el adaptador creado por esto la muestra de informacion es mucho mas rapida y sencilla. Pues firebaseUI se conecta directamente a el nodo de la base de datos si existe algun cambio dentro de este lo muestra pues esta a la escucha.
6. Con esto tendríamos funcionando la visualizacion de los clientes que se encuentren dados de alta en nuestro database de firebase.

## CONSULTAR CLIENTE

1. Esto nos ayudara a visualizar a un unico cliente para poder realizar operaciones de actualizacion y eliminacion en la base de datos.
2. Creamos una clase llamada "ConsultarCliente" que tendra el siguiente codigo:

```
package com.tutorial.tutorialclientes;

import android.content.DialogInterface;
import android.content.Intent;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.TextView;

import com.google.firebase.database.DatabaseReference;

public class ConsultarCliente extends AppCompatActivity {

    //Creando variable de referencia para la base de datos (Firebase)
    DatabaseReference firebase;
    //Llamando a la clase conexion para la creacion de la referencia
    Conexion conexion = new Conexion();

    TextView edt_nombre_consultar, edt_correo_consultar, edt_telefono_consultar,
    edt_direccion_consultar;
    String nombre, correo, telefono, direccion;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_consultar_cliente);
        firebase = conexion.conexion();

        //Toolbar
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
```

```

getSupportActionBar().setTitle("Cliente");
getSupportActionBar().setDisplayHomeAsUpEnabled(true);
getSupportActionBar().setDisplayShowHomeEnabled(true);

edt_nombre_consultar = findViewById(R.id.edt_nombre_consultar);
edt_correo_consultar = findViewById(R.id.edt_correo_consultar);
edt_telefono_consultar = findViewById(R.id.edt_telefono_consultar);
edt_direccion_consultar = findViewById(R.id.edt_direccion_consultar);

//Obtener variables del cardview

correo = getIntent().getStringExtra("correo");
telefono = getIntent().getStringExtra("telefono");
direccion = getIntent().getStringExtra("direccion");
nombre = getIntent().getStringExtra("nombre");

edt_nombre_consultar.setText(getIntent().getStringExtra("nombre"));
edt_correo_consultar.setText(getIntent().getStringExtra("correo"));
edt_telefono_consultar.setText(getIntent().getStringExtra("telefono"));
edt_direccion_consultar.setText(getIntent().getStringExtra("direccion"));
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()){
        case android.R.id.home:
            Intent intent = new Intent(ConsultarCliente.this,
visualizarClientes.class);
            startActivity(intent);
            finish();
            break;
        case R.id.action_delete:
            AlertDialog.Builder eliminar = new AlertDialog.Builder(this);
            eliminar.setMessage("¿Desea eliminar este cliente?");
            eliminar.setTitle("Eliminar cliente");
            eliminar.setPositiveButton("Si", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which) {
                    firebase.child("Cliente").child(nombre).removeValue();
                    finish();
                }
            });

            eliminar.setNegativeButton("No", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which) {
                    dialog.cancel();
                }
            });

            AlertDialog dialog = eliminar.create();
            dialog.show();
            break;
    }
}

```

```

        case R.id.action_edit:
            Intent intent2 = new Intent(ConsultarCliente.this,
ActualizarCliente.class);
            intent2.putExtra("nombre", nombre);
            intent2.putExtra("correo", correo);
            intent2.putExtra("telefono", telefono);
            intent2.putExtra("direccion", direccion);
            startActivity(intent2);
            break;
    }
    return super.onOptionsItemSelected(item);
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_consultar_cliente, menu);
    return true;
}
}

```

3. Creamos un layout llamado "activity\_consultar\_cliente" que tendra un codigo de la siguiente manera:

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.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:orientation="vertical"
    tools:context=".Consultar.ConsultarClienteActivity">

    <include
        android:id="@+id/toolbar"
        layout="@layout/toolbar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <android.support.constraint.Guideline
        android:id="@+id/guideline4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        app:layout_constraintGuide_percent="0.2995896" />

    <TextView
        android:id="@+id/edt_nombre_consultar"
        android:layout_width="match_parent"

```

```
android:layout_height="wrap_content"
android:layout_marginStart="50dp"
android:layout_marginTop="50dp"
android:layout_marginEnd="50dp"
android:drawablePadding="5dp"
android:gravity="center"
android:textSize="18sp"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.0"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="@+id/guideline4" />
```

```
<TextView
    android:id="@+id/edt_correo_consultar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="50dp"
    android:layout_marginTop="50dp"
    android:layout_marginEnd="50dp"
    android:drawablePadding="15dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/edt_nombre_consultar" />
```

```
<TextView
    android:id="@+id/edt_telefono_consultar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="50dp"
    android:layout_marginTop="50dp"
    android:layout_marginEnd="50dp"
    android:drawablePadding="15dp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/edt_correo_consultar" />
```

```
<TextView
    android:id="@+id/edt_direccion_consultar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="50dp"
    android:layout_marginTop="50dp"
    android:layout_marginEnd="50dp"
    android:drawablePadding="15dp"
    app:layout_constraintBottom_toTopOf="@+id/guideline2"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/edt_telefono_consultar" />
```

```
<android.support.constraint.Guideline
```



```
        android:id="@+id/guideline2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        app:layout_constraintGuide_percent="0.93" />
```

```
</android.support.constraint.ConstraintLayout>
```

4. Creamos dentro de la carpeta "res" un paquete llamado "menu" y dentro de menu crearemos un layout que contendra nuestras dos acciones de eliminar y actualizar el codigo del layout llamado "menu\_consultar\_cliente" debe quedar de la siguiente manera:

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto">

    <item
        android:id="@+id/action_delete"
        android:icon="@drawable/ic_delete"
        android:orderInCategory="100"
        app:showAsAction="always" />

    <item
        android:id="@+id/action_edit"
        android:icon="@drawable/ic_edit"
        android:orderInCategory="100"
        app:showAsAction="always" />

</menu>
```

Debes cambiar los icon por los que tu consideres pertinentes en este caso agregar a drawable los iconos de preferencia para eliminar y editar.

5. Con esto obtendremos una vista previa sobre el cliente en el que nos encontramos y las diferentes acciones que deseamos realizar.

## ACTUALIZAR CLIENTE

1. Creamos una clase llamada "ActualizarCliente" con el codigo de la siguiente manera:

```
package com.tutorial.tutorialclientes;

import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.text.TextUtils;
import android.view.MenuItem;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
```

```

import android.widget.Toast;
import com.google.firebase.database.DatabaseReference;

public class ActualizarCliente extends AppCompatActivity {

    Conexion conexion = new Conexion();
    DatabaseReference firebase;

    TextView edt_nombre_actualizar;
    EditText edt_correo_actualizar, edt_telefono_actualizar, edt_direccion_actualizar;
    String recibonombre;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_actualizar_cliente);
        firebase = conexion.conexion();

        //Toolbar
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        getSupportActionBar().setTitle("Actualizar cliente");
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        getSupportActionBar().setDisplayShowHomeEnabled(true);

        edt_nombre_actualizar = findViewById(R.id.edt_nombre_actualizar);
        edt_correo_actualizar = findViewById(R.id.edt_correo_actualizar);
        edt_direccion_actualizar = findViewById(R.id.edt_direccion_actualizar);
        edt_telefono_actualizar = findViewById(R.id.edt_telefono_actualizar);

        recibonombre = getIntent().getStringExtra("nombre");
        Toast.makeText(getApplicationContext(), recibonombre,
Toast.LENGTH_SHORT).show();

        edt_nombre_actualizar.setText(getIntent().getStringExtra("nombre"));
        edt_correo_actualizar.setText(getIntent().getStringExtra("correo"));
        edt_telefono_actualizar.setText(getIntent().getStringExtra("telefono"));
        edt_direccion_actualizar.setText(getIntent().getStringExtra("direccion"));
    }

    public void btn_actualizar_cliente(View view) {
        String correo = edt_correo_actualizar.getText().toString();
        String telefono = edt_telefono_actualizar.getText().toString();
        String direccion = edt_direccion_actualizar.getText().toString();

        if (TextUtils.isEmpty(correo)) {
            Toast.makeText(getApplicationContext(), "Ingresar correo",
Toast.LENGTH_SHORT).show();
        } else if (TextUtils.isEmpty(telefono)) {
            Toast.makeText(getApplicationContext(), "Ingresar telefono",
Toast.LENGTH_SHORT).show();
        } else if (TextUtils.isEmpty(direccion)) {
            Toast.makeText(getApplicationContext(), "Ingresar direccion",

```

```

Toast.LENGTH_SHORT).show();
        }else if (!TextUtils.isEmpty(correo) && !TextUtils.isEmpty(telefono) &&
!TextUtils.isEmpty(direccion)) {

firebase.child("Cliente").child(recibonombre).child("correo").setValue(correo);

firebase.child("Cliente").child(recibonombre).child("telefono").setValue(telefono);

firebase.child("Cliente").child(recibonombre).child("direccion").setValue(direccion);
        Toast.makeText(getApplicationContext(), "Cliente acutalizado
correctamente", Toast.LENGTH_SHORT).show();
        Intent intent = new Intent(ActualizarCliente.this,
VisualizarClientes.class);
        startActivity(intent);
        finish();
    }
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case android.R.id.home:
            String nombre = edt_nombre_actualizar.getText().toString();
            String correo = edt_correo_actualizar.getText().toString();
            String telefono = edt_telefono_actualizar.getText().toString();
            String direccion = edt_direccion_actualizar.getText().toString();

            if (TextUtils.isEmpty(correo)) {
                Toast.makeText(getApplicationContext(), "Ingresar correo",
Toast.LENGTH_SHORT).show();
            } else if (TextUtils.isEmpty(telefono)) {
                Toast.makeText(getApplicationContext(), "Ingresar telefono",
Toast.LENGTH_SHORT).show();
            } else if (TextUtils.isEmpty(direccion)) {
                Toast.makeText(getApplicationContext(), "Ingresar direccion",
Toast.LENGTH_SHORT).show();
            } else if (!TextUtils.isEmpty(correo) && !TextUtils.isEmpty(telefono) &&
!TextUtils.isEmpty(direccion)) {

firebase.child("Cliente").child(recibonombre).child("correo").setValue(correo);

firebase.child("Cliente").child(recibonombre).child("telefono").setValue(telefono);

firebase.child("Cliente").child(recibonombre).child("direccion").setValue(telefono);
                Toast.makeText(getApplicationContext(), "Cliente acutalizado
correctamente", Toast.LENGTH_SHORT).show();
                Intent intent = new Intent(ActualizarCliente.this,
VisualizarClientes.class);
                startActivity(intent);
                finish();
            }

            Intent intent = new Intent(ActualizarCliente.this,

```

```

visualizarClientes.class);
        startActivity(intent);
        finish();
        return true;

        default:
            return super.onOptionsItemSelected(item);
    }
}
}

```

2. Despues crear un layout llamado "activity\_actualizar\_cliente" con el codigo de la siguiente manera:

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView 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=".ActualizarCliente"
    android:fillViewport="true">

    <android.support.constraint.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <include
            android:id="@+id/toolbar"
            layout="@layout/toolbar"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent" />

        <TextView
            android:id="@+id/edt_nombre_actualizar"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginStart="8dp"
            android:layout_marginTop="50dp"
            android:layout_marginEnd="8dp"
            android:gravity="center_horizontal"
            android:inputType="text"
            android:textSize="18sp"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toBottomOf="@+id/toolbar" />

        <android.support.design.widget.TextInputLayout
            android:id="@+id/textInputLayout3"
            android:layout_width="match_parent"

```

```

        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="50dp"
        android:layout_marginEnd="32dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/edt_nombre_actualizar">

        <EditText
            android:id="@+id/edt_correo_actualizar"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:drawablePadding="5dp"
            android:inputType="text" />

    </android.support.design.widget.TextInputLayout>

    <android.support.design.widget.TextInputLayout
        android:id="@+id/textInputLayout2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="50dp"
        android:layout_marginEnd="32dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textInputLayout3">

        <EditText
            android:id="@+id/edt_direccion_actualizar"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:drawablePadding="5dp"
            android:inputType="number" />

    </android.support.design.widget.TextInputLayout>

    <android.support.design.widget.TextInputLayout
        android:id="@+id/textInputLayout12"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="50dp"
        android:layout_marginEnd="32dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textInputLayout2">

        <EditText
            android:id="@+id/edt_telefono_actualizar"

```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:drawablePadding="5dp"
        android:inputType="number" />

</android.support.design.widget.TextInputLayout>

<Button
    android:id="@+id/btn_actualizar_automovil"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="32dp"
    android:layout_marginTop="50dp"
    android:layout_marginEnd="32dp"
    android:onClick="btn_actualizar_cliente"
    android:text="Actualizar cliente"
    android:textSize="18sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textInputLayout12" />

</android.support.constraint.ConstraintLayout>
</ScrollView>

```

3. Con esto se podrá actualizar los campos del cliente menos el que se utiliza como llave primaria en este caso el nombre.

## ELIMINAR CLIENTE

1. Se ocupará la clase "ConsultarCliente" pues la acción se lleva en las opciones dentro de los items del menú. Para esto deberás revisar el código e ir a el apartado siguiente:

```

case R.id.action_delete:
    AlertDialog.Builder eliminar = new AlertDialog.Builder(this);
    eliminar.setMessage("¿Desea eliminar este cliente?");
    eliminar.setTitle("Eliminar cliente");
    eliminar.setPositiveButton("Si", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            firebase.child("Cliente").child(nombre).removeValue();
            finish();
        }
    });

    eliminar.setNegativeButton("No", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            dialog.cancel();
        }
    });

```

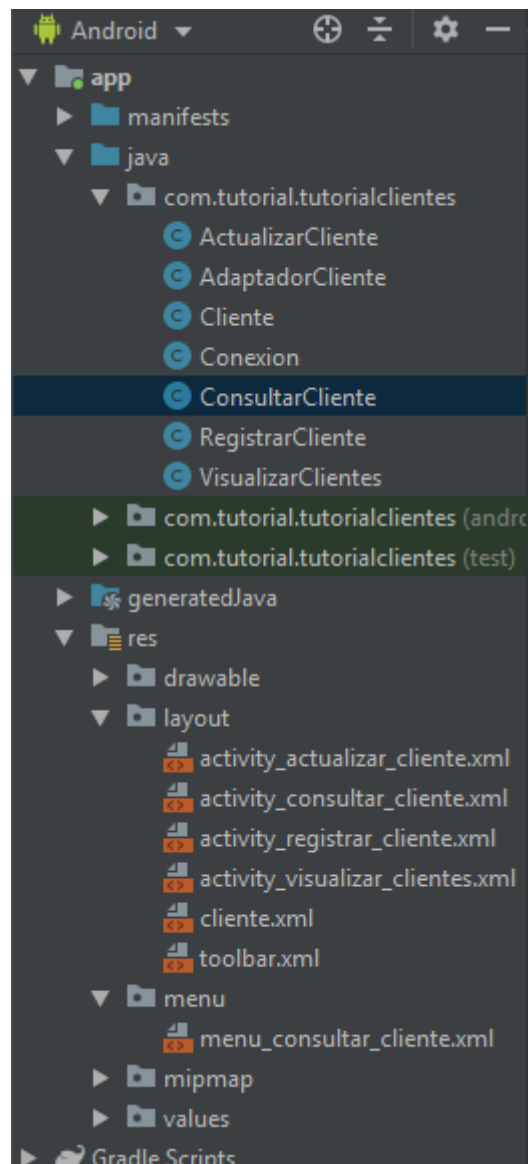
```
});  
  
AlertDialog dialog = eliminar.create();  
dialog.show();  
break;
```

2. Esto crea una ventana de mensaje que nos pide confirmacion de eliminacion en dado caso de ser positiva nuestra respuesta como se puede ver se hace referencia a la base de datos y con `removeValue()` se elimina de la base de datos nuestro cliente.

## DISTRIBUCCION DE LAS CARPETAS

---

1. En dado caso de que hasta aqui vayas bien tus carpetas deberan tener la siguiente estructura:



## PROYECTO EN GITHUB DE PRUEBA

---

1. Este proyecto de prueba lo podras encontrar en la siguiente ruta, en dado caso de no funcionar la base de datos recuerda que deberas cambiar el archivo JSON que se encuentra adentro por el tuyo asi como el SHA-1.

```
https://github.com/MacNebur/TutorialClientes.git
```