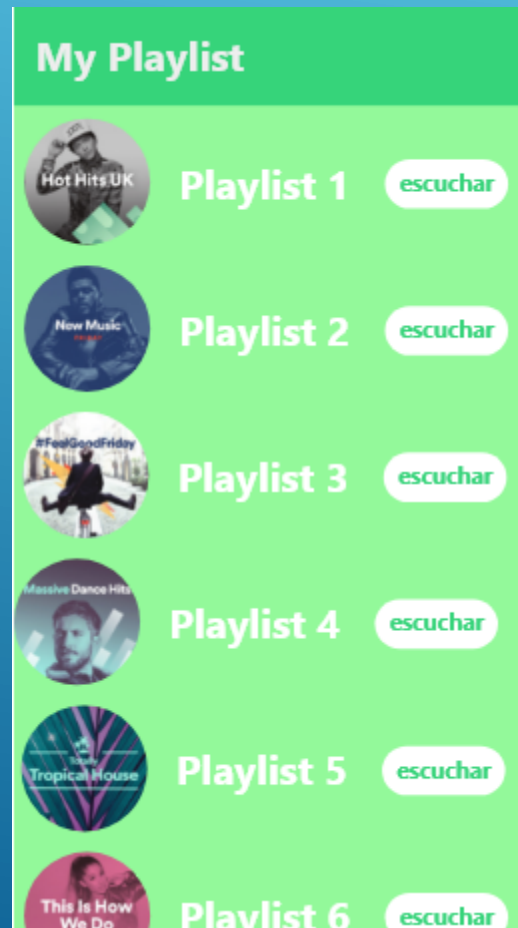


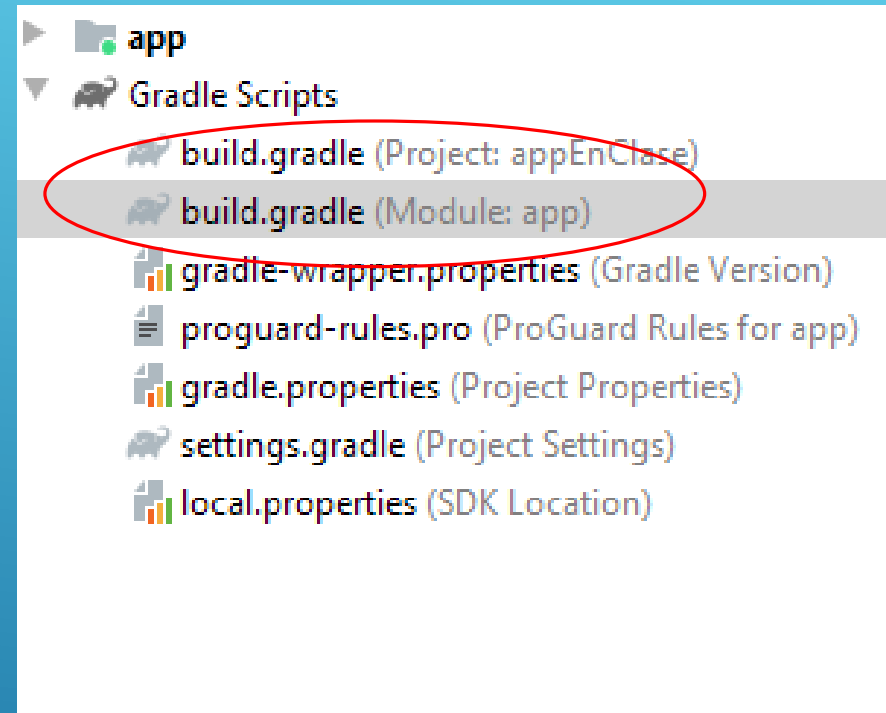
# RESOLUCIÓN DEL HITO 4

Johan Alejandro Marin Montoya

# 1.- Crear un Screen el cual Te lleve a otro activity



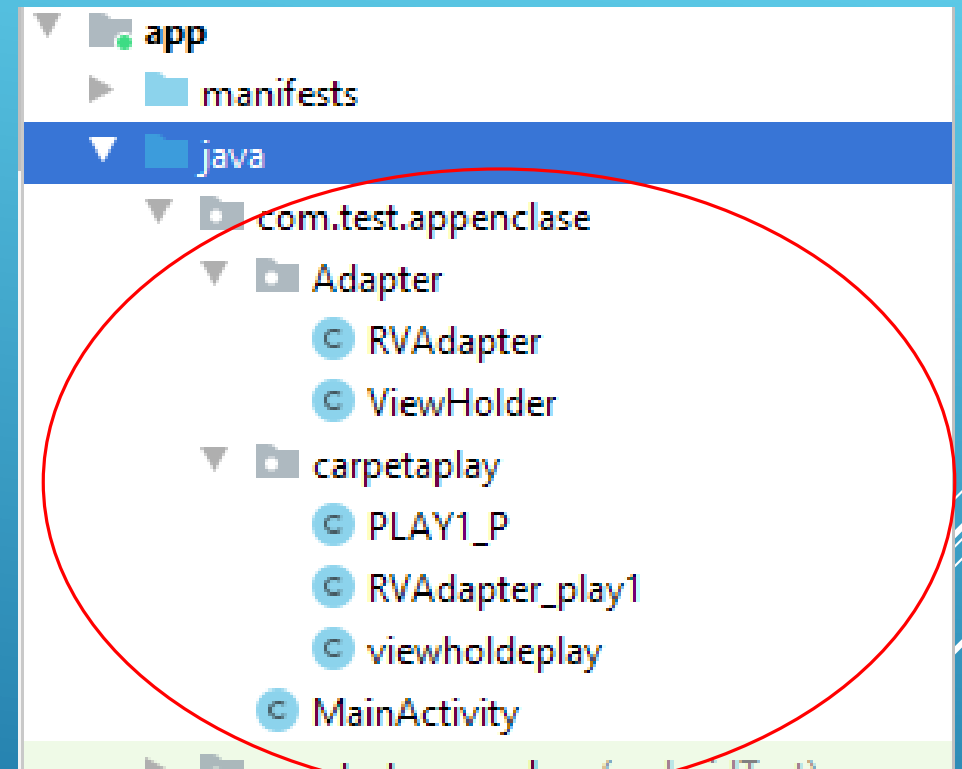
Primeramente hay que descargar las librerías que vamos a usar en, por lo tanto tenemos que acceder a la siguiente dirección de nuestro proyecto **build.grandle**



```
dependencies {  
    implementation fileTree(dir: 'libs', include: ['*.jar'])  
  
    implementation 'androidx.appcompat:appcompat:1.1.0'  
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'  
    testImplementation 'junit:junit:4.12'  
    androidTestImplementation 'androidx.test.ext:junit:1.1.1'  
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.2.0'  
  
    //recyclerview  
    implementation 'com.android.support:preference-v7:29.0.0'  
    //glyde for images  
    implementation 'com.github.bumptech.glide:glide:4.11.0'  
    annotationProcessor 'com.github.bumptech.glide:compiler:4.11.0'  
    //Circle ImageView  
    implementation 'de.hdodenhof:circleimageview:3.1.0'  
}
```

Copiar la siguientes líneas de código y descargar la librería

Copiar la siguientes líneas de código y descargar la librería

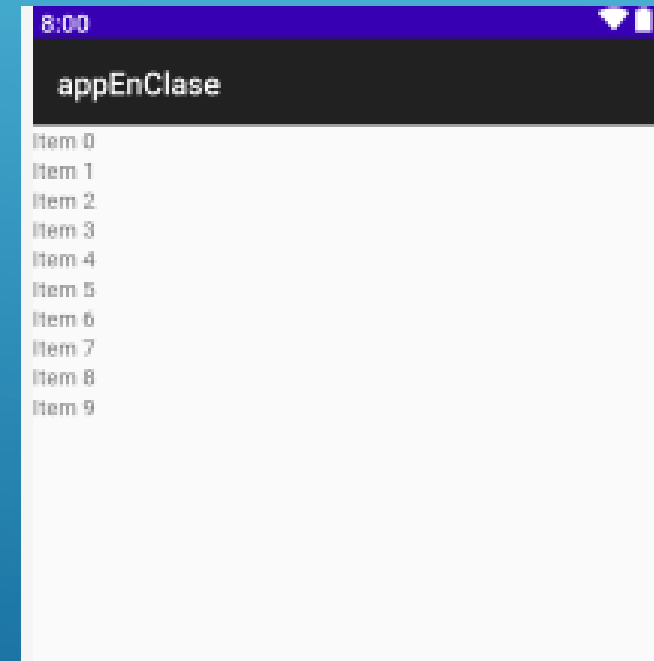


```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/rvContainer"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
    </androidx.recyclerview.widget.RecyclerView>

</RelativeLayout>
```

Usar recycle en el main Layout  
y obtendremos esto por el  
momento



```
<de.hdodenhof.circleimageview.CircleImageView
    android:id="@+id/civItem"
    android:layout_width="60dp"
    android:layout_height="60dp"
    android:src="@mipmap/ic_launcher"/>

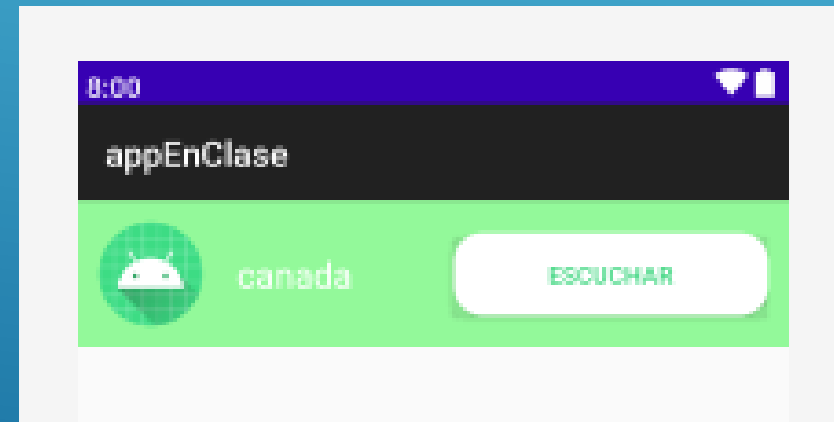
<TextView
    android:id="@+id/tvImage"
    android:layout_width="140dp"
    android:layout_height="wrap_content"
    android:text="@string/canada"
    android:layout_marginStart="5dp"
    android:layout_toEndOf="@+id/civItem"
    android:layout_centerVertical="true"
    android:padding="15dp"
    android:textColor="@color/blanco"
    android:textSize="20sp"/>

<Button
    android:id="@+id/btplay"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/escuchar"

    android:background="@drawable/cajas"
    android:layout_toEndOf="@id/tvImage"
    android:layout_centerVertical="true"
    android:textColor="@color/verdefuerte"

/>
```

Una vez hecho el código del layout hay que ir al java para modificar las imágenes que saldrán pero el diseño ya está hecho.



Hay que agarrar y colocar los links para sacar las imágenes



```
private RecyclerView recyclerView;  
private RVAdapter rvAdapter;  
private ArrayList<String> imagesName = new ArrayList<>();  
private ArrayList<String> imagesURL = new ArrayList<>();  
//almacenamos todos los nombres de las imagenes en una sola variable imgURL y imgnam  
private final String imgURL =  
"V40gA40,yB3d2qH,9jMbaX2,rYndmdq,sypYnSP,HBeK1ot,YCqbt8r,eLk31XX,XzffKgy,4ZHp7FO,D1ab  
k5C,XzffKgy";  
private final String imgNam = "Playist 1,Playist 2,Playist 3,Playist 4,Playist  
5,Playist 6,Playist 7,Playist 8, Playist 9,Playist 10,Playist 11,Playist 12";
```



```
protected void onCreate(Bundle savedInstanceState)
{
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    initImageBitMaps();
    initRecyclerView();
}
```



Colocamos todas la funciones  
que usaremos.

```
public void initImageBitmaps()  
{  
    imagesURL.addAll(Arrays.asList(imgURL.split(",")));  
    imagesName.addAll(Arrays.asList(imgNam.split(",")));  
}
```



Sitar o mostrar con los recycles  
view todo esto en el  
MainActivity

Una vez que creamos nuestras variables con arrays, iniciamos como objetos según el nombre de nuestras variables

Sacamos la URL

Mostramos como se va a ver el diseño

Y mandamos a otra Activity

```
public RVAdapter(Context context, ArrayList<String> imageNames, ArrayList<String> imagesURI)
{
    this.context = context;
    this.imageNames = imageNames;
    this.imagesURI = imagesURI;
}

@Override
public ViewHolder onCreateViewHolder(ViewGroup viewGroup, int viewType)
{
    View view = LayoutInflater.from(viewGroup.getContext()).inflate(R.layout.rl_list_item, viewGroup, false);
    ViewHolder vHolder = new ViewHolder(view);
    return vHolder;
}

@Override
public void onBindViewHolder(@NonNull ViewHolder holder, final int position)
{
    String URI = "https://i.imgur.com/";
    Glide.with(context)
        .asBitmap()
        .load(URI + imagesURI.get(position) + ".png") //aqui le paso carga desde una url internet
        .into(holder.getCircleImage());
    holder.getTvImage().setText(imageNames.get(position));

    holder.getTvImage().setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Toast.makeText(context, "my " + imageNames.get(position), Toast.LENGTH_SHORT).show();
        }
    });

    holder.getBoton().setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent inn = new Intent(context, PLAY1_P.class);
            context.startActivity(inn);
        }
    });
}
```

## 2.- Crear un Screen donde estén todas las canciones y aparezca un mensaje



▼ carpetaplay

- PLAY1\_P
- RVAdapter\_play1
- viewholdeplay



Creamos un Activity en blanco y seguimos los mismos pasos que la parte anterior

```
<androidx.recyclerview.widget.RecyclerView  
    android:id="@+id/rvContainerREPRODUCIR"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent">
```

```
</androidx.recyclerview.widget.RecyclerView>
```

```
<androidx.recyclerview.widget.RecyclerView  
    android:id="@+id/rvContainerPAUSA"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
>
```

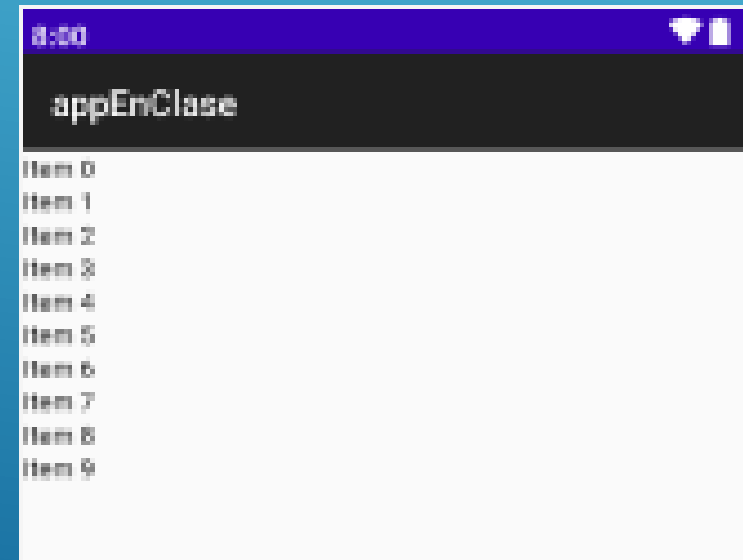
```
</androidx.recyclerview.widget.RecyclerView>
```

```
<androidx.recyclerview.widget.RecyclerView  
    android:id="@+id/rvContainerAPAGAR"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
>
```

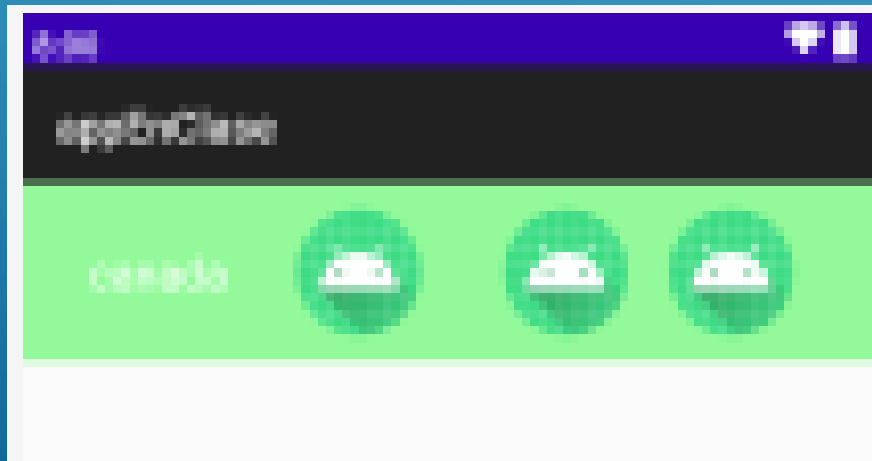
```
</androidx.recyclerview.widget.RecyclerView>
```



Y de esta manera igual  
que la anterior sección  
podemos realizar



Un TextView y 3 cricleview  
el cual de forma al activity



```
<TextView
    android:id="@+id/tvImage1"
    android:layout_width="240dp"
    android:layout_height="wrap_content"
    android:text="@string/canada"
    android:layout_marginStart="5dp"

    android:layout_centerVertical="true"
    android:padding="15dp"
    android:textColor="@color/blanco"
    android:textSize="20sp"/>

<de.hdodenhof.circleimageview.CircleImageView
    android:id="@+id/reproducir"
    android:layout_width="60dp"
    android:layout_height="60dp"
    android:layout_marginLeft="120dp"
    android:src="@mipmap/ic_launcher"
/>

<de.hdodenhof.circleimageview.CircleImageView
    android:id="@+id/pausa"
    android:layout_width="60dp"
    android:layout_height="60dp"
    android:src="@mipmap/ic_launcher"
    android:layout_marginLeft="220dp"
/>

<de.hdodenhof.circleimageview.CircleImageView
    android:id="@+id/apagar"
    android:layout_width="60dp"
    android:layout_height="60dp"
    android:src="@mipmap/ic_launcher"
    android:layout_marginLeft="300dp"
/>
```

```

public class PLAY1_P extends AppCompatActivity {
    private RecyclerView recyclerView1,recyclerView2,recyclerView3;
    private RVAdapter_play1 rvAdapterPlay1;
    private ArrayList<String> imagesName = new ArrayList<>();
    private ArrayList<String> imagesURLPLAY = new ArrayList<>();
    private ArrayList<String> imagesURLPAUSA = new ArrayList<>();
    private ArrayList<String> imagesURLAPAGAR = new ArrayList<>();

    //almacenamos todos los nombres de las imagenes en una sola variable imgURL y imgnam
    private final String imagesurlPLAY = "59AoTWq,59AoTWq,59AoTWq,59AoTWq,59AoTWq,59AoTWq,59AoTWq,59AoTWq,59AoTWq,59AoTWq,59AoTWq";
    private final String imagesurlPAUSA = "2Nat0RQ,2Nat0RQ,2Nat0RQ,2Nat0RQ,2Nat0RQ,2Nat0RQ,2Nat0RQ,2Nat0RQ,2Nat0RQ,2Nat0RQ,2Nat0RQ";
    private final String imagesurlAPAGAR = "LJjd1Qe,LJjd1Qe,LJjd1Qe,LJjd1Qe,LJjd1Qe,LJjd1Qe,LJjd1Qe,LJjd1Qe,LJjd1Qe,LJjd1Qe,LJjd1Qe";
    private final String imagesNameeee = "Music 1,Music 2,Music 3,Music 4,Music 5,Music 6,Music 7,Music 8, Music 9,Music 10,Music 11,Music 12";

```



Sacamos los links necesarios e  
inicializamos la variables  
respectivamente



Concatenamos y agregamos las imágenes que queremos que se ven en el circle



```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_p_l_a_y1_p);  
    initImageBitMaps();  
    initRecyclerView();  
}  
public void initImageBitMaps()  
{//agregamos dentro del array las imagenes y sus nombres  
    //le setiamos un valor numerico  
    //funcion addAll hace lo mismo sin crear una variable extra  
    imagesURLPLAY.addAll(Arrays.asList(imagesurlPLAY.split(",")));  
    imagesURLPAUSA.addAll(Arrays.asList(imagesurlPAUSA.split(",")));  
    imagesURLAPAGAR.addAll(Arrays.asList(imagesurlAPAGAR.split(",")));  
    imagesName.addAll(Arrays.asList(imagesNameeee.split(",")));  
}
```

```

public void onBindViewHolder(@NonNull final viewholdeplay holder, int position) {
    //glide libreria importante
    String URI = "https://i.imgur.com/";
    Glide.with(context)
        .asBitmap()
        .load(URI + imagesURIplay.get(position) + ".png")
        .into(holder.getCircleImage1());
    holder.getTvImage1().setText(imageNames.get(position));

    Glide.with(context)
        .asBitmap()
        .load(URI + imagesURIpausa.get(position) + ".png")
        .into(holder.getCircleImage2());

    Glide.with(context)
        .asBitmap()
        .load(URI + imagesURIapagar.get(position) + ".png")
        .into(holder.getCircleImage3());

    //cuando aga klik mostrara el mensaje en el circleimage play
    holder.getCircleImage1().setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            //mandamos el mensaje al presionar en el texto segun su posicion
            Toast.makeText(context, "play music", Toast.LENGTH_SHORT).show();
        }
    });

    //cuando aga klik mostrara el mensaje en el circleimage pause
    holder.getCircleImage2().setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            //mandamos el mensaje al presionar en el texto segun su posicion
            Toast.makeText(context, "pause music", Toast.LENGTH_SHORT).show();
        }
    });
}

```



Mostramos el mensaje que vamos a mostrar cuando el usuario le de click a play stop

### 3.- El botón Play debe cambiarse "Pause

Playlist 1		
music 1	⏸	⏮
music 2	▶	⏮
music 3	▶	⏮
music 4	▶	⏮
music 5	▶	⏮
music 6	▶	⏮
music 7	▶	⏮
music 8	▶	⏮
music 9	▶	⏮
music 10	▶	⏮
music 11	▶	⏮
music 12	▶	⏮

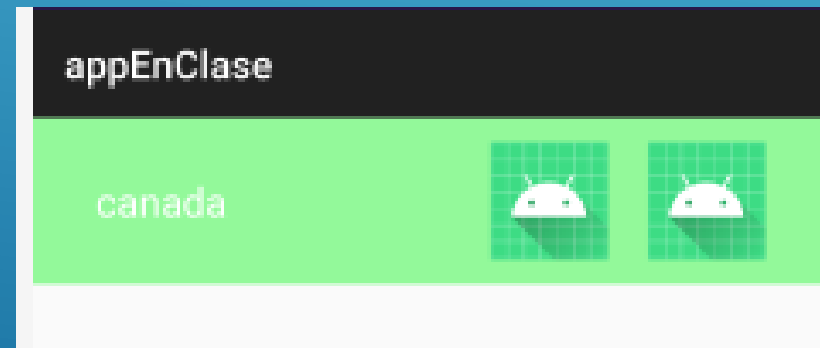
```
<TextView
    android:id="@+id/tvImage1"
    android:layout_width="240dp"
    android:layout_height="wrap_content"
    android:text="@string/canada"
    android:layout_marginStart="5dp"

    android:layout_centerVertical="true"
    android:padding="15dp"
    android:textColor="@color/blanco"
    android:textSize="20sp"/>

<de.hdodenhof.circleimageview.CircleImageView
    android:id="@+id/reproducir"
    android:layout_width="60dp"
    android:layout_height="60dp"
    android:layout_marginLeft="220dp"
    android:src="@mipmap/ic_launcher"
/>

<de.hdodenhof.circleimageview.CircleImageView
    android:id="@+id/pausa"
    android:layout_width="60dp"
    android:layout_height="60dp"
    android:src="@mipmap/ic_launcher"
    android:layout_marginLeft="220dp"
/>
```

Se cambia ligeramente el código del layout de tal manera que nos saldrá así



El código no cambiara más que en esta parte así dando click en el circle que cambie de icono de play a pause

```
holder.getCircleImage1().setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        //mandamos el mensaje al presionar en el texto segun su posicion  
        Toast.makeText(context, "play music", Toast.LENGTH_SHORT).show();  
    }  
});  
  
//cuando aga clik mostrara el mensaje en el circleimage pause  
holder.getCircleImage2().setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        //mandamos el mensaje al presionar en el texto segun su posicion  
        Toast.makeText(context, "pause music", Toast.LENGTH_SHORT).show();  
    }  
});
```