

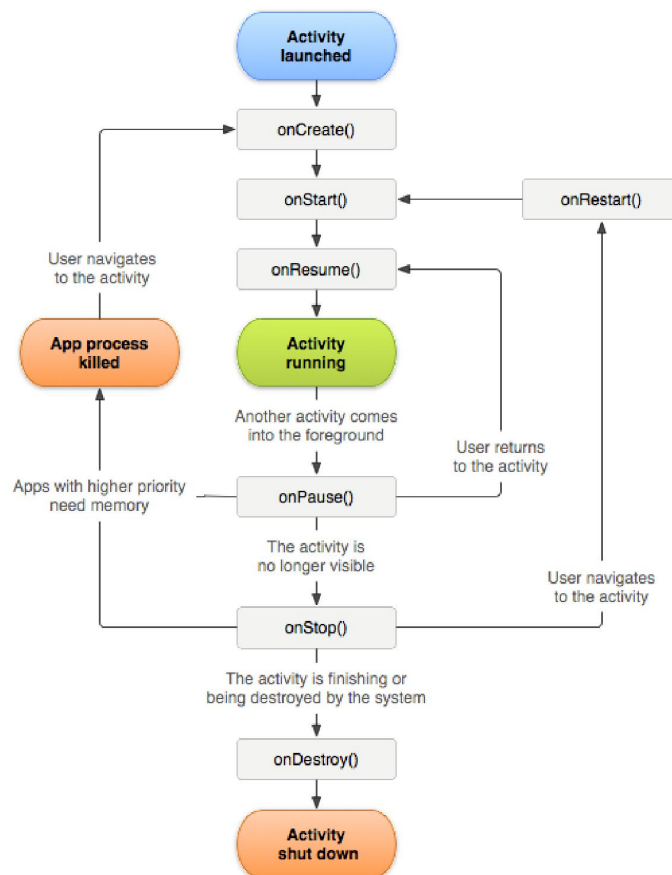
## PROGRAMACION EN ANDROID CON JAVA

### PRACTICA No. 4

### EVENTOS

#### QUE ES UNA ACTIVITY (ACTIVIDAD)

- Las actividades representan el componente principal de la interfaz gráfica de una aplicación Android. Se puede pensar en una actividad como el elemento análogo a una ventana en cualquier otro lenguaje visual.
- Las activities están conformadas por dos partes: la parte lógica y la parte gráfica

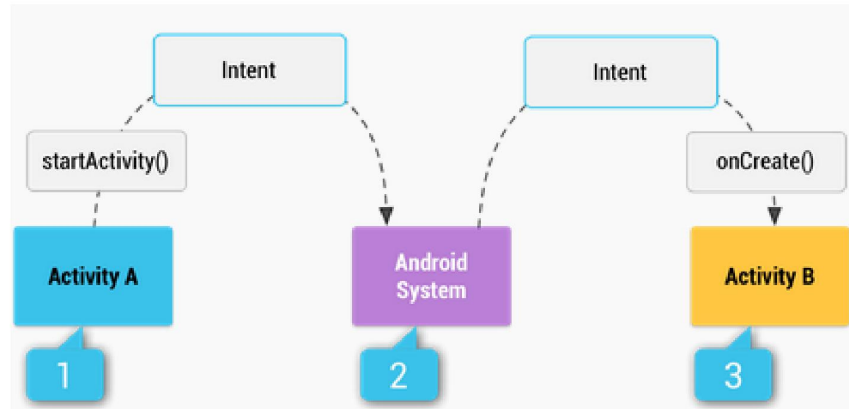


#### QUE ES UN INTENT (INTENSION)

- Un *intent* es el elemento básico de comunicación entre los distintos componentes Android que hemos descrito anteriormente. Se pueden entender como los

mensajes o peticiones que son enviados entre los distintos componentes de una aplicación o entre distintas aplicaciones.

- Mediante un intent se puede mostrar una actividad desde cualquier otra, iniciar un servicio, enviar un mensaje broadcast, iniciar otra aplicación, etc.



## PARTE 1. CREACION DEL PROYECTO

Ejecute el programa Android Studio. En la ventana que se despliega haga clic sobre el menú File y la Opción New Project.

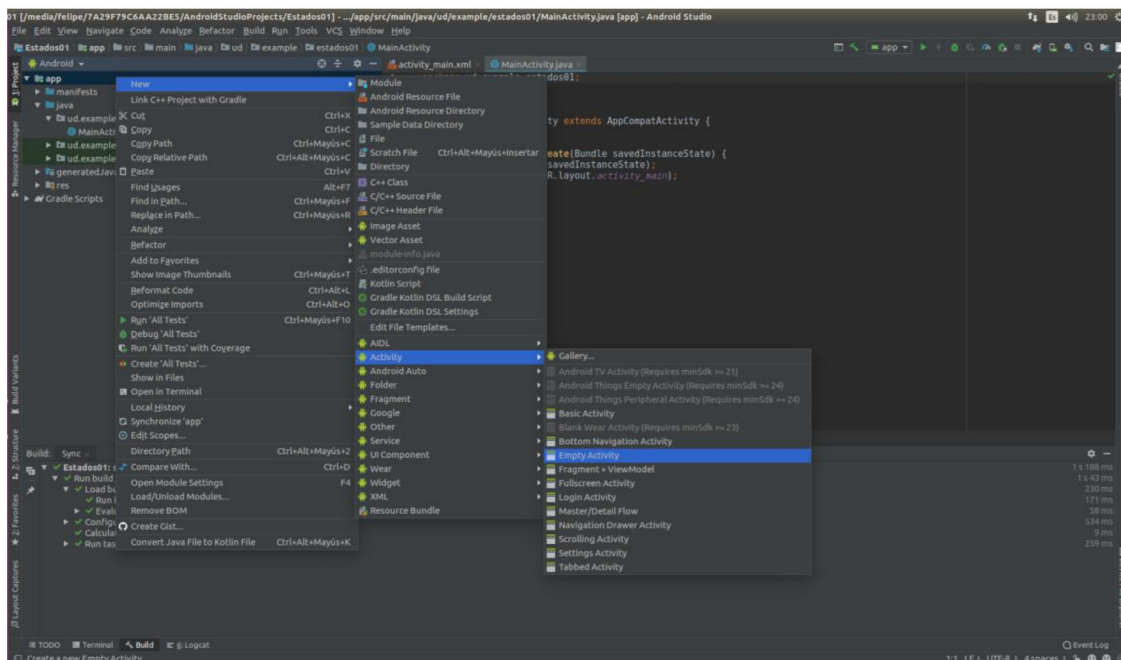
En la nueva ventana seleccione la pestaña Phone and Table, elija la opción Empty Activity y haga clic en el botón Next.

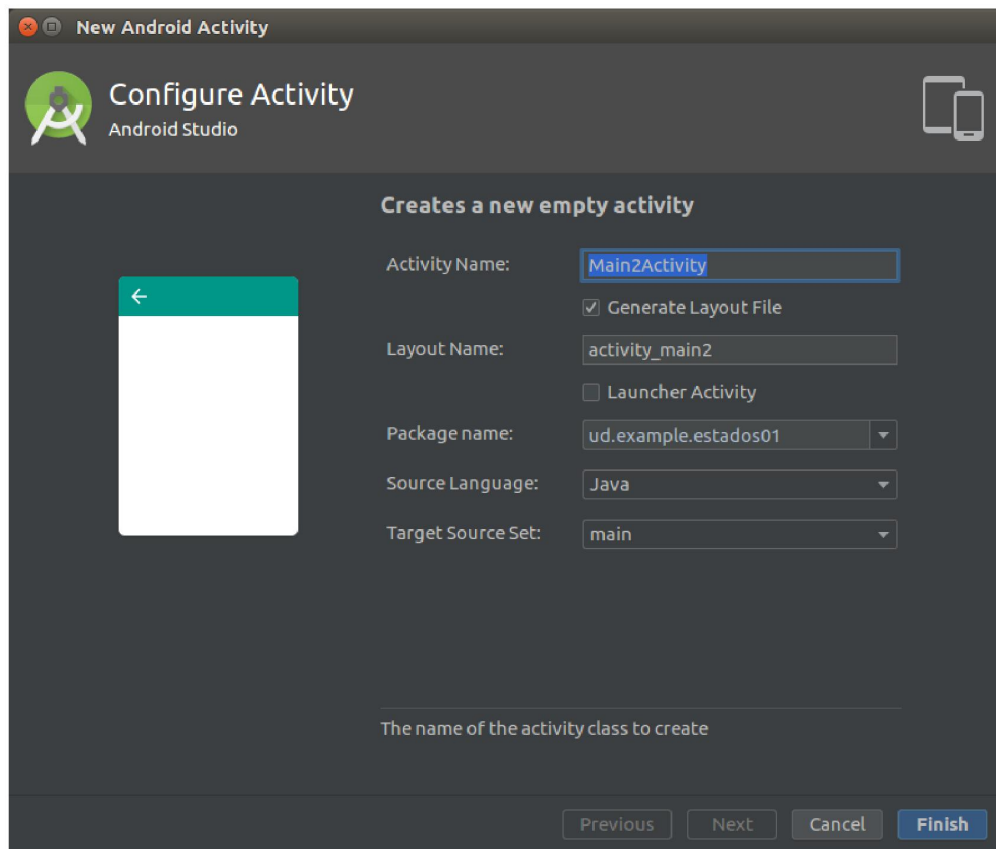
En la siguiente ventana escriba como nombre del proyecto **Estados01** y como dominio escriba **ud.example.estados01**, seleccione la mínima versión de android de ejecución para su nueva aplicación (se recomienda la API 15 por defecto), seleccione como lenguaje Java y haga clic sobre el botón Finish.

## PARTE 2. ADICION DE CODIGO

### Adición de una nueva Actividad

Haga clic derecho sobre el proyecto y en el menú flotante seleccione la opción Empty Activity del submenú Activity del menú New, y acepte los parámetros tal como están en la ventana que se despliega y finalice.





### Modificación del archivo *activity\_main.xml*

Nuestra actividad al final debería lucir de la siguiente manera:



Regrese a la actividad principal, y haga doble clic sobre el archivo *activity\_main.xml* del paquete *layout*, y cambie el modo de visualización a texto, por medio de la pestaña *Text*. Modifique el código de la siguiente manera:

```
<?xml version="1.0" encoding="utf-8"?>
<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=".MainActivity">

    <LinearLayout
        android:layout_width="395dp"
        android:layout_height="715dp"
        android:orientation="vertical"
        tools:layout_editor_absoluteX="8dp"
        tools:layout_editor_absoluteY="8dp">

        <Button
            android:id="@+id/button"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Button" />

    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

### Modificación del archivo *MainActivity.java*

Haga doble clic sobre el archivo *MainActivity.java* del paquete *java*, y modifique el código de la siguiente manera:

Agregue el siguiente atributo privados a la clase

```
private Button myBoton;
```

Agregue los siguientes métodos a la clase:

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

    Toast.makeText(this, "Se ejecuta el metodo
onCreate", Toast.LENGTH_SHORT).show();
}
```

```
myBoton = (Button) findViewById(R.id.button);
myBoton.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        // Perform action on click
        Intent i = new Intent(getApplicationContext(),
Main2Activity.class);
        startActivity(i);
    }
});
}

@Override
protected void onPause() {
    super.onPause();
    Toast.makeText(this, "Se ejecuta el metodo
onPause", Toast.LENGTH_SHORT).show();
}

@Override
protected void onRestart() {
    super.onRestart();
    Toast.makeText(this, "Se ejecuta el metodo
onRestart", Toast.LENGTH_SHORT).show();
}

@Override
protected void onResume() {
    super.onResume();
    Toast.makeText(this, "Se ejecuta el metodo
onResume", Toast.LENGTH_SHORT).show();
}

@Override
protected void onStart() {
    super.onStart();
    Toast.makeText(this, "Se ejecuta el metodo
onStart", Toast.LENGTH_SHORT).show();
}

@Override
protected void onDestroy() {
    super.onDestroy();
    Toast.makeText(this, "Se ejecuta el metodo
onDestroy", Toast.LENGTH_SHORT).show();
}

@Override
protected void onStop() {
    super.onStop();
    Toast.makeText(this, "Se ejecuta el metodo
onStop", Toast.LENGTH_SHORT).show();
}
```

Ejecuté y Evalué la aplicación

### **TRABAJO ADICIONAL**

1. Evalúe en cómo se puede aprovechar cada uno de los eventos trabajados en esta práctica.