

Evidencia dia 1 semana 11

Componente Spinner:

El widget Spinner muestra una lista desplegable para seleccionar un único elemento.

Siguiendo con el ejercicio de la clase anterior, vamos a diseñar una nueva actividad, la cual incluirá un spinner que desplegará una lista “producto”. En el main activity crearemos un button con el método intent, que nos enviará a la actividad del spinner.

Para la lista productos crearemos una clase (Producto), definiremos 3 variables código, nombre y precio, los métodos constructor con los 3 valores inicializados y sus métodos constructor por defecto y los Get Setter para las 3 variables que nos permitirá establecer y recuperar esos valores. (POO/Encapsulamiento)

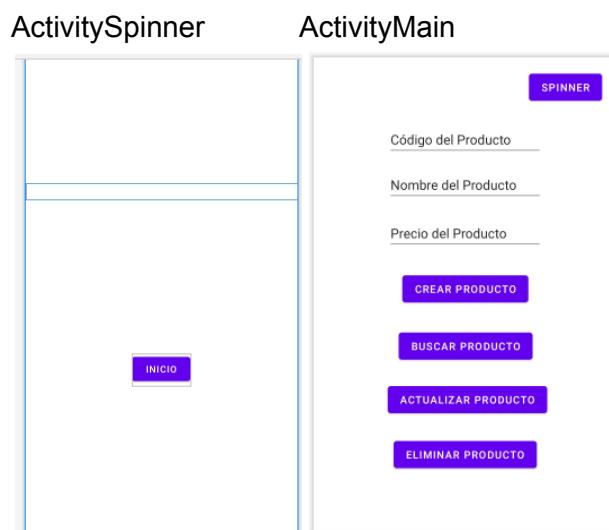
Diseño de Interfaz activity Spinner:

1 Spinner

1 Button

Se arrastran los componentes al layout, se personaliza cada componente en strings.xml y se realizan los constraints (restricciones).

Interfaz



Código Java Actividad Spinner

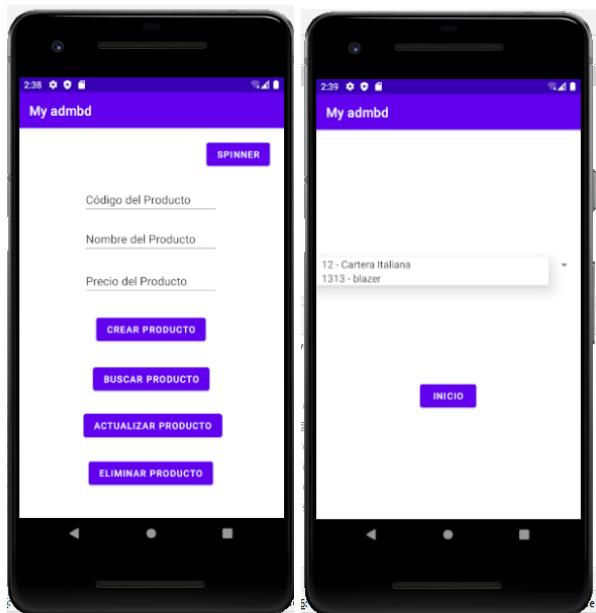
```
public class MainActivitySpinner extends AppCompatActivity {  
    private Spinner spl;  
  
    ArrayList<String> listaProductos;  
    ArrayList <Producto> productosList;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main_spinner);
        sp1 = findViewById(R.id.sp1);
        consultarListaProductos();
        ArrayAdapter<String> adaptador = new ArrayAdapter <> (this,
android.R.layout.simple_spinner_item, listaProductos);
        sp1.setAdapter(adaptador);
    }
    public void consultarListaProductos(){
        Adminindb admins = new Adminindb(this, "Productos", null,1);
        SQLiteDatabase base = admins.getReadableDatabase();
        Producto p1 = null;
        productosList = new ArrayList <Producto>();
        Cursor fila = base.rawQuery("select * from Producto", null);
        while (fila.moveToNext()){
            p1 = new Producto();
            p1.setCodigo(fila.getInt(0));
            p1.setNombre(fila.getString(1));
            p1.setPrecio(fila.getInt(2));
            productosList.add(p1);
        }
        ObtenerProductos();           base.close();
    }
    public void ObtenerProductos(){
        listaProductos = new ArrayList<String>();
        for(int i=0;i<productosList.size();i++){
            listaProductos.add(productosList.get(i).getCodigo() + " -
"+productosList.get(i).getNombre());
        }
    }
    public void Inicio (View v){
        Intent i = new Intent(this, MainActivity.class);
        startActivity(i);

    }
}
```

Emulador

Main



Reflexión: Se aplicaron bastantes conceptos de POO en esta clase, ya podemos ir diseñando algunas funciones más complejas y útiles.