



Instituto Tecnológico de San Juan del Río



HolaMundo (Java/Kotlin) **–Desarrollo de aplicaciones para** **dispositivos móviles–**

Valeria Hernández Yáñez

19590674

P R E S E N T A:

Ingeniería en Tecnologías de la Información y Comunicación

San Juan del Río, Qro., 21 Abril de 2023.



**Instituto Mexicano
de Normalización y
Certificación. A.C.**



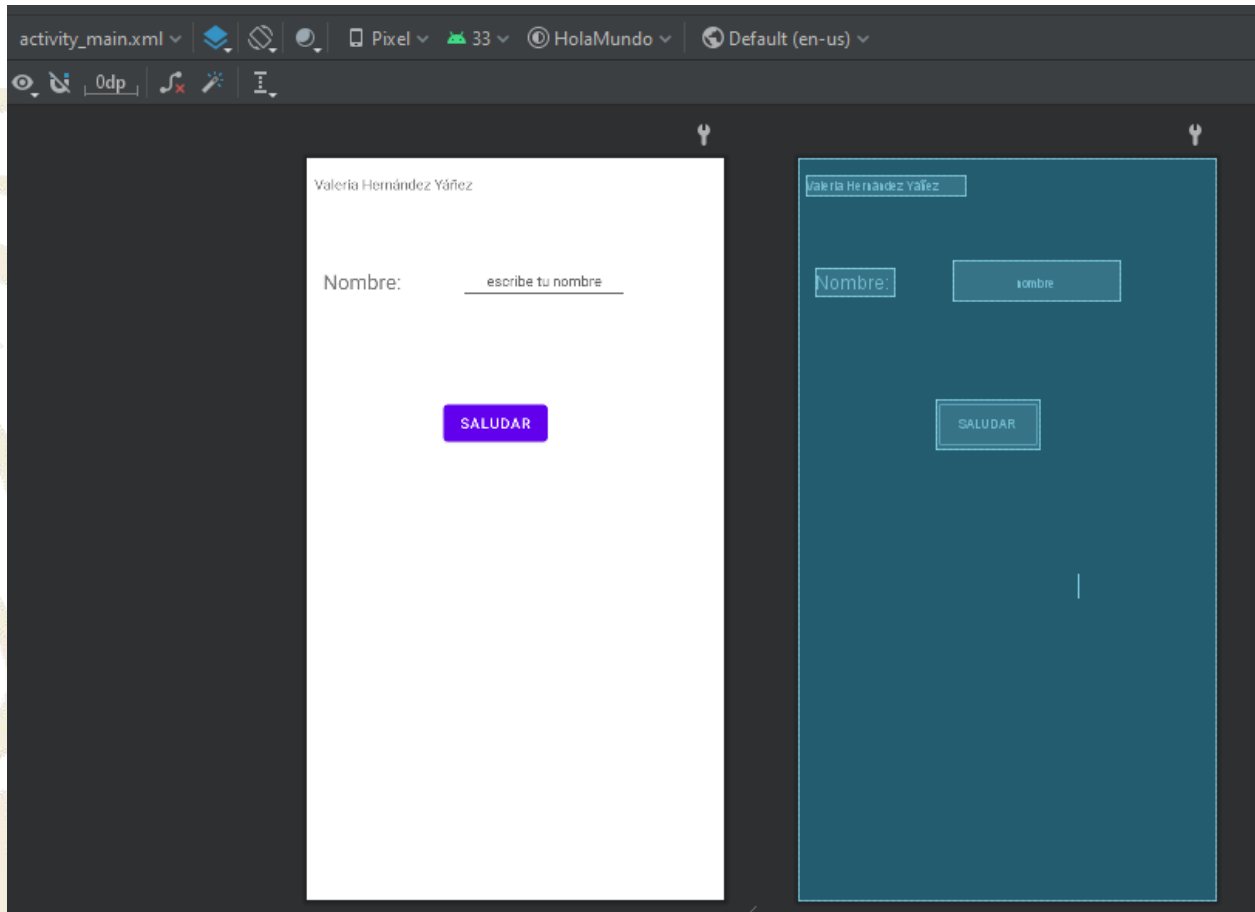
Av. Tecnológico #2 Col. Centro C.P.76800 San Juan del Río, Querétaro. Tel. 4272724118
o 4272724178 ext: **ext** e-mail: **inserte correo**@tecnm.mx tecnm.mx | sjuanrio.tecnm.mx



2023
AÑO DE
Francisco
VILLA
EL REVOLUCIONARIO DEL PUEBLO



Diseño Java



- Se utiliza un Textview
- Se utiliza un Edittext para darle un valor a la variable
- Un Button para generar la acción indicada



Codigo Java

```
activity_main.xml x MainActivity.java x
package com.example.holamundo;

import ...

public class MainActivity extends AppCompatActivity {
    private EditText nomb;
    private TextView recivee;

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

        nomb = (EditText) findViewById(R.id.nombre);
        recivee = (TextView) findViewById(R.id.recive);
    }

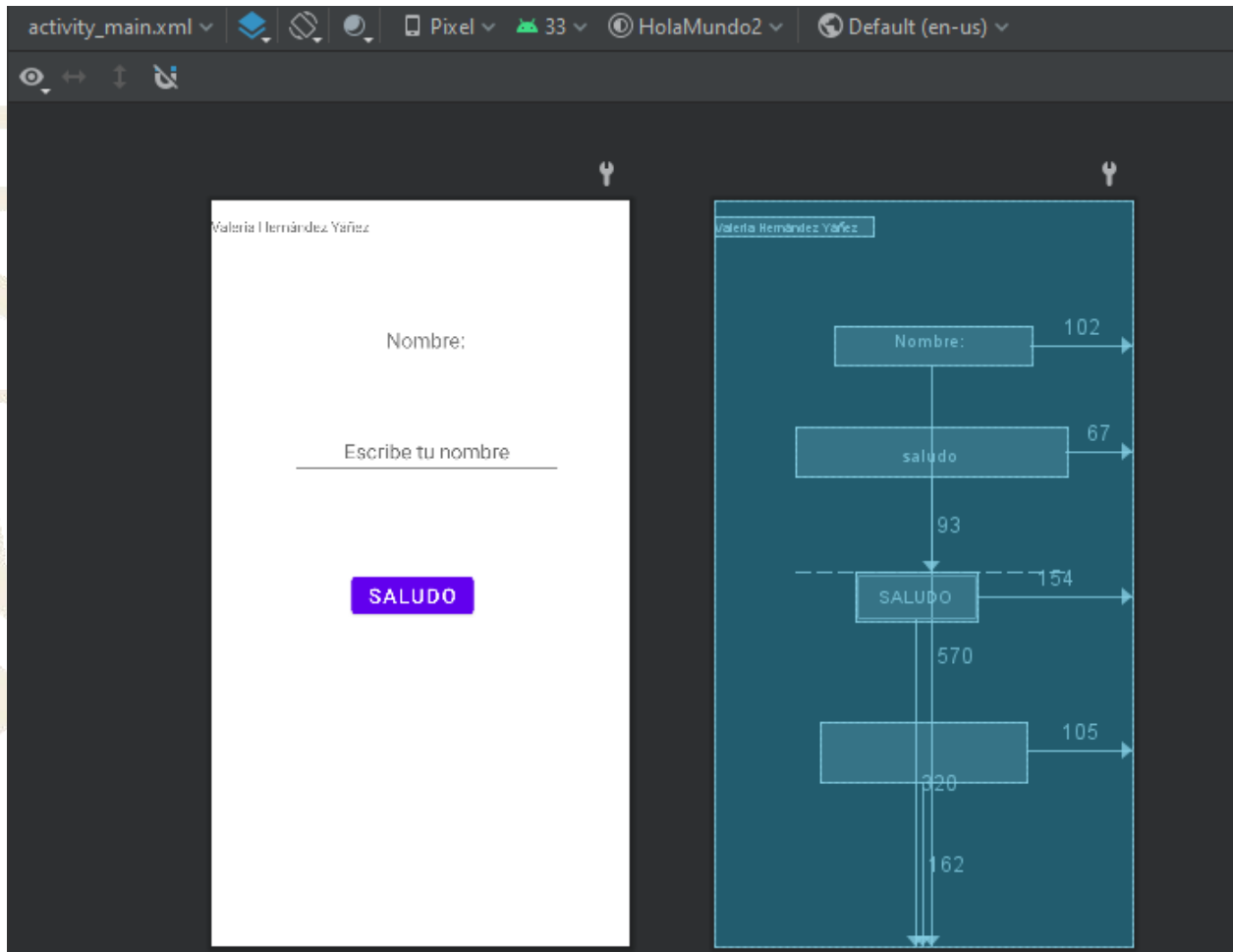
    public void captura(View view) {
        String n;
        n = nomb.getText().toString();

        if (n.trim().length() > 0) {
            recivee.setText("Hola " + n + " como estas?!");
        } else {
            Toast.makeText(context: this, text: "campo vacio", Toast.LENGTH_LONG).show();
        }
    }
}
```

- Declaramos la variable del EditText, donde guardaremos el valor de nuestra variable, y la mandamos a llamar con *findViewById*.
- Declararemos la variable del TextView quien será la encargada de recibir la variable del EditText y la mandamos a llamar con *findViewById*.
- En dado caso de no poner nada en el campo del EditText mandara un mensaje de *campo vacío* y no dejara hacer la acción del *saludo*.
- El Button estará programado con la función de *View* para poder llamar la variable y hacer la acción indicada



Diseño Kotlin



- Se utiliza un Textview
- Se utiliza un Edittext para darle un valor a la variable
- Un Button para generar la acción indicada



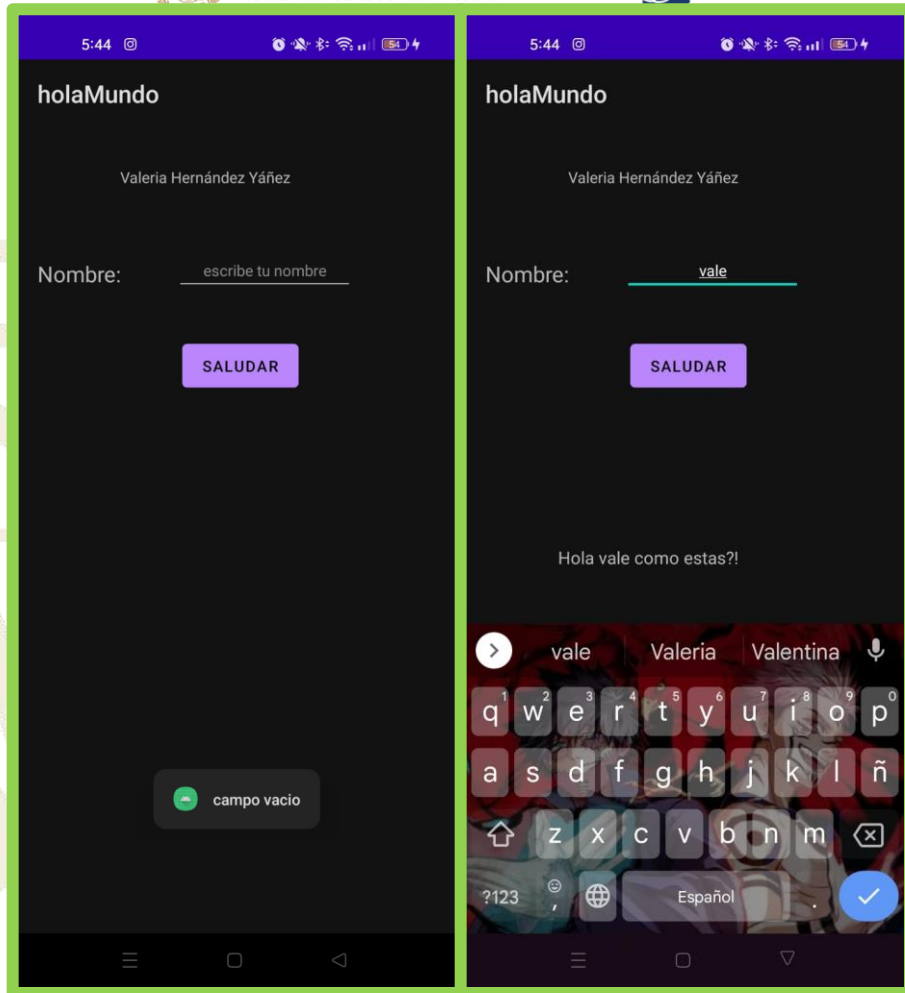


Codigo segunda pagina

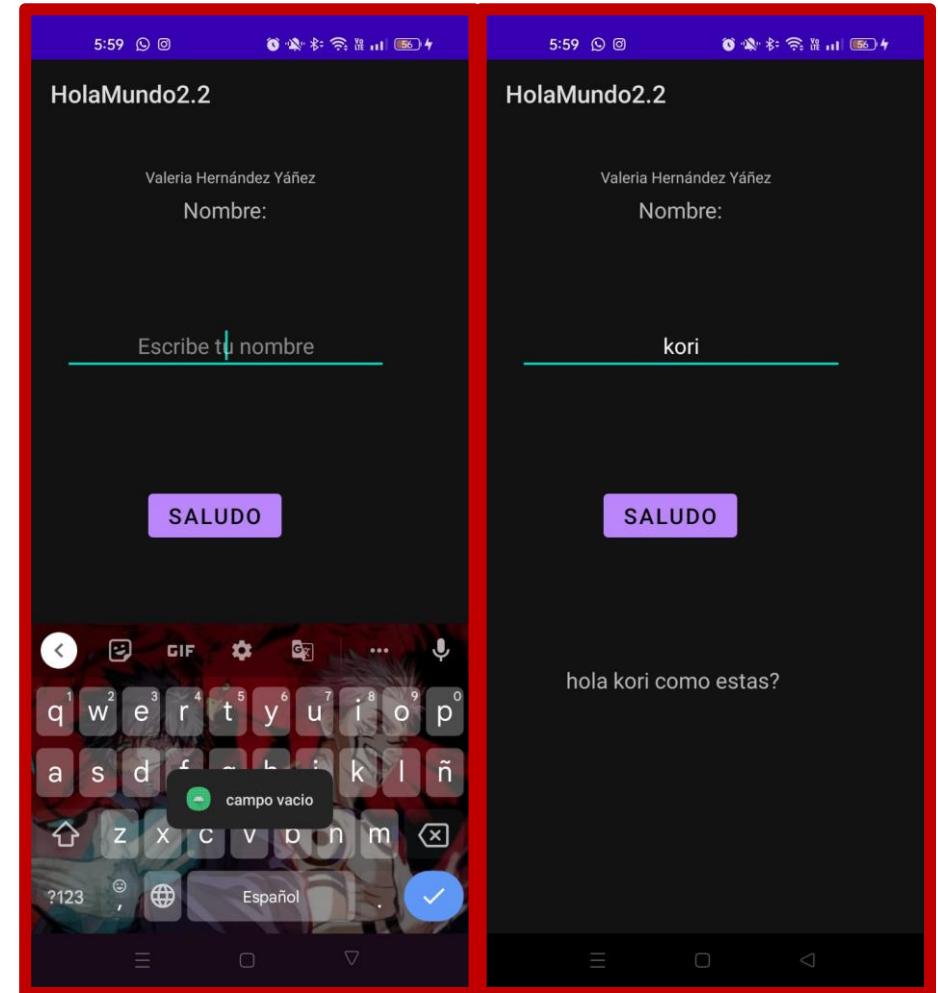
```
activity_main.xml x MainActivity.kt x
1 package com.example.holamundo2
2
3 import ...
4
5
6
7
8
9
10 class MainActivity : AppCompatActivity() {
11     override fun onCreate(savedInstanceState: Bundle?) {
12         super.onCreate(savedInstanceState)
13         setContentView(R.layout.activity_main)
14
15         var saludo : EditText = findViewById(R.id.saludo)
16         var btn1 : Button = findViewById(R.id.btn1)
17         var nom1 : TextView = findViewById(R.id.nomb1)
18         btn1.setOnClickListener { it: View!
19             if (saludo.text.trim().length > 0) {
20                 nom1.text = "hola " + (saludo.text) + " como estas?"
21             } else {
22                 Toast.makeText( context: this, text: "campo vacio", Toast.LENGTH_LONG).show()
23             }
24         }
25     }
26 }
27 }
```

- En kotlin declaramos y llamamos las variables al mismo tiempo no necesario declararlas y luego llamarlas
- Agregamos la función para el botón donde llamara la variable para generar la función del saludo
- Se mandara un mensaje de campo vacío en dado caso que el campo de Edittext este vacío y no dejara avanzar.





Resultado final java y kotlin



Video:

<https://youtube.com/shorts/aWln9rGgcx8>

