



Alumno: Amezcua Arévalo Santiago

Profesor: Cardoso Arellano Cristian

Materia:

Desarrollo de Aplicaciones Móviles

Semestre: 6to

Periodo: 2025-2

Nombre de la actividad:

Actividad M3-01

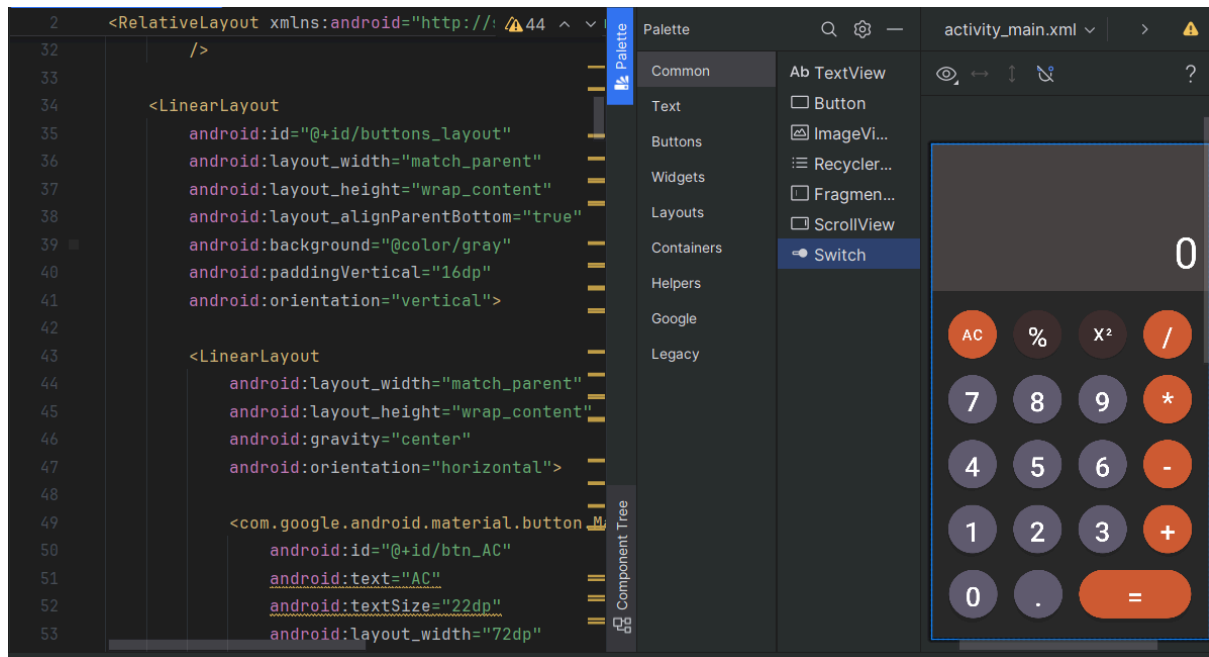
Fecha de entrega: 14 de Abril del
2025

Universidad Nacional Autónoma de
México

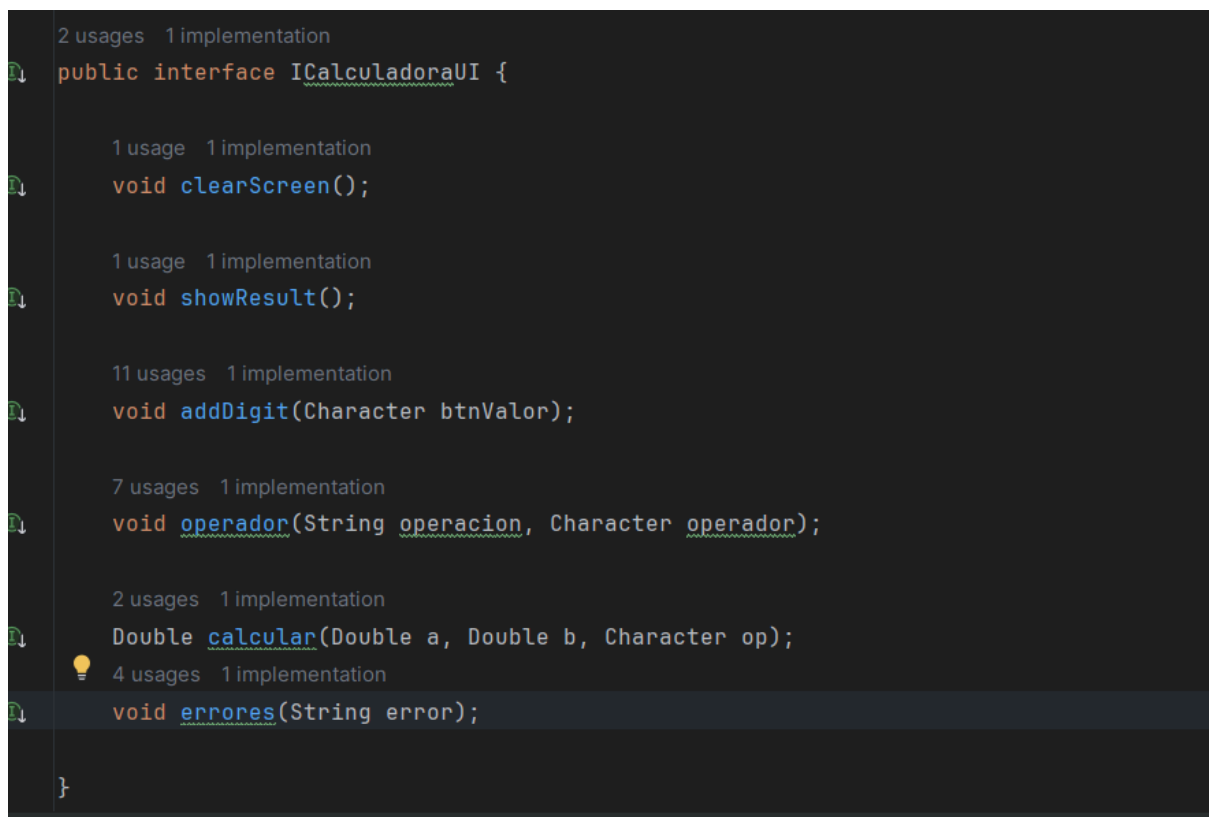
Facultad de Contaduría y
Administración

Licenciatura en Informática

Para la creación de la calculadora primero cree una interfaz utilizando LinearLayouts para que se acomodasen los botones y el texto, como yo quería;



Después cree una Interfaz que me serviría para crear un objeto en main que haría todo, y ese objeto sería una instancia de la clase que implementaba la interfaz. En este hice varios métodos.



Para empezar estaba el constructor que inicializaba los botones para buscar su ID correspondiente en la vista que ya había hecho, después utilice lambdas para hacer llamadas a la función onClickListener, los números y el punto (.), se concatenaban entre sí, pero si en medio de esas llamadas había un operador, se concatenaban como cadenas diferentes, esto lo logre con una bandera que evaluaba si el usuario había presionado alguno en medio de esta nueva solución, también un método de limpiar pantalla y de calcular el resultado per sé.

Después está el método que muestra en pantalla el resultado, el cual evalúa si existen suficientes elementos para poder mostrar algo en pantalla.

```
269
    1 usage
270    @Override
271    public void showResult() {
272        if (this.opC != null && this.resul != null) {
273            Double num2 = 0.0;
274            if (!this.operaC.isEmpty()) {
275                num2 = Double.parseDouble(this.operaC);
276                this.resF = calcular(this.resul, num2, this.opC);
277                txv_resultado.setText(String.valueOf(this.resF));
278            } else {
279                txv_resultado.setText(String.valueOf(this.resul));
280            }
281        } else if (this.resul != null) {
282            txv_resultado.setText(String.valueOf(this.resul));
283        } else {
284            txv_resultado.setText("0");
285        }
286    }
287
```

Conclusiones:

Después de desarrollar esta aplicación me queda más claro el funcionamiento de las interfaces y como la optimización de código puede mejorar el funcionamiento de la aplicación, además del manejo de errores, también reforcé mis conocimientos sobre la UI y como se utilizan los LinearLayout y el RelativeLayout en Android