



IPN
Instituto Politécnico Nacional



UPIIZ
*Unidad Profesional Interdisciplinaria de Ingeniería
Campus Zacatecas*

Ingeniería en:
Sistemas Computacionales

Materia:
Desarrollo de Aplicaciones Móviles Nativas.

Docente:
Ing. Efraín Arredondo Morales

Actividad:
"Practica III.- Aplicaciones nativas"

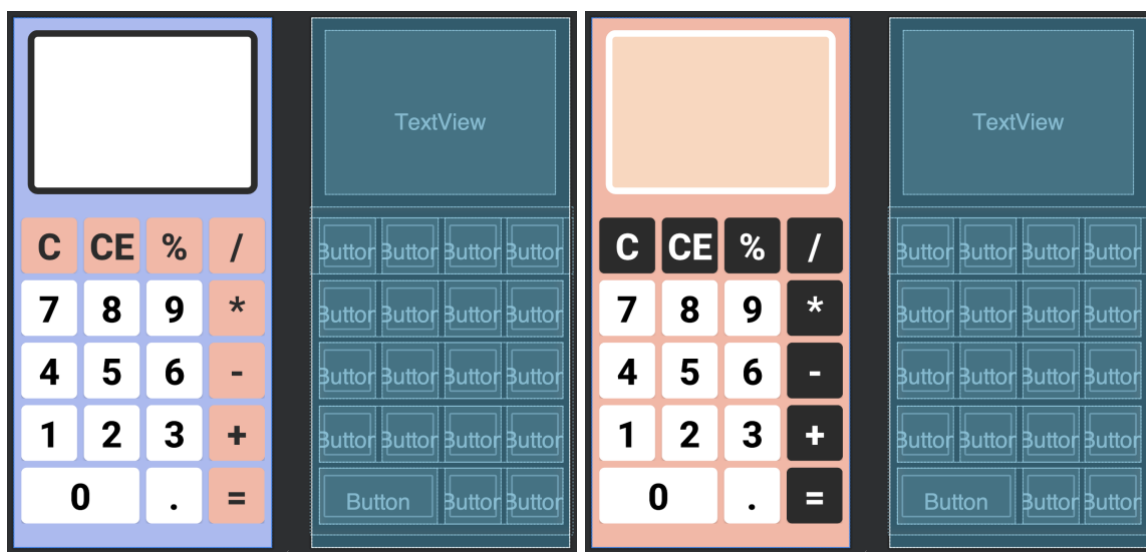
Alumno:
Alejandro Tamayo Castro

Grupo:
7CM1

Zacatecas, Zac. octubre 01 de 2023

Diseño

En el diseño se optó por modelos de botones pertenecientes a Material Design 3, con fondos en blanco para los números y en otros colores para las acciones u operaciones, se optó por mostrar los números al centro superior, para los colores de la aplicación se usaron los documentos de temas, para el claro y oscuro con los siguientes colores: #FCB5A4, #2B2B2B, #FFD5BC para el tema oscuro, y #A9BBF2, #FCB5A4 para el tema claro.



Código

Se hace uso de botones y listeners de los mismos, agrando en forma de cadenas los números y signos, al momento de clicar en los simbolos de operación se guarda el número que se tiene hasta al momento por medio de una subcadena, iniciando desde el 0 hasta el tamaño de la cadena, antes de insertar el símbolo, se guarda el signo y su ubicación. Para hacer los cálculos se usa un when en el listener del igual, aquí inicialmente se guarda la subcadena del segundo número, y por medio del operador se hace la operación matemática correspondiente. Se implementó además una variable booleana para denotar si la cadena que está en pantalla es un resultado,

si es así entonces al momento de clicar en un número la pantalla se limpia para permitir ingresar números nuevos.

```
zero.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "0"
}

n1.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "1"
}

n2.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "2"
}

n3.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "3"
}

n4.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "4"
}

n5.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "5"
}

n6.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "6"
}

n7.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "7"
}

n8.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "8"
}

n9.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "9"
}
```

```
divide.setOnClickListener{ it: View!
    if(res)
        res=false
    num1 = screen.text.toString()
    screen.text = screen.text.toString() + "/"
    location = screen.length()-1
    operator = screen.text.substring(location)
}

percent.setOnClickListener{ it: View!
    if(res)
        res=false
    num1 = screen.text.toString()
    screen.text = screen.text.toString() + "%"
    location = screen.length()-1
    operator = screen.text.substring(location)
}
```

```

dot.setOnClickListener{ it: View!
    if(res){
        res = false
        screen.text = ""
    }
    screen.text = screen.text.toString() + "."
}

//Operaciones
add.setOnClickListener{ it: View!
    if(res)
        res=false
    num1 = screen.text.toString()
    screen.text = screen.text.toString() + "+"
    location = screen.length()-1
    operator = screen.text.substring(location)
}

minus.setOnClickListener{ it: View!
    if(res)
        res=false
    num1 = screen.text.toString()
    screen.text = screen.text.toString() + "-"
    location = screen.length()-1
    operator = screen.text.substring(location)
}

mult.setOnClickListener{ it: View!
    if(res)
        res=false
    num1 = screen.text.toString()
    screen.text = screen.text.toString() + "*"
    location = screen.length()-1
    operator = screen.text.substring(location)
}

```

```

equal.setOnClickListener{ it: View!
    num2 = screen.text.substring(location+1, screen.length())
    when (operator) {
        "+" -> {
            val result = (num1.toDoubleOrNull() ?: 0.0) + (num2.toDoubleOrNull() ?: 0.0)
            screen.text = result.toString()
            res = true
        }

        "-" -> {
            val result = (num1.toDoubleOrNull() ?: 0.0) - (num2.toDoubleOrNull() ?: 0.0)
            screen.text = result.toString()
            res = true
        }

        "*" -> {
            val result = (num1.toDoubleOrNull() ?: 0.0) * (num2.toDoubleOrNull() ?: 0.0)
            screen.text = result.toString()
            res = true
        }

        "/" -> {
            val result = (num1.toDoubleOrNull() ?: 0.0) / (num2.toDoubleOrNull() ?: 0.0)
            screen.text = result.toString()
            res = true
        }

        "%" -> {
            val result = (num1.toDoubleOrNull() ?: 0.0) * ((num2.toDoubleOrNull() ?: 0.0)/100)
            screen.text = result.toString()
            res = true
        }
    }
}

```

Funcionamiento

A continuación, se muestran algunos ejemplos del funcionamiento de la aplicación.

