



INGENIERÍA DE SOFTWARE

PROGRAMACIÓN EN DISPOSITIVOS MÓVILES

PROFESOR: JORGE ALBERTO HERNÁNDEZ
TAPIA

ALUMNO: LUIS ARTURO ORTIZ SUAREZ

MATRICULA: 18100172

AI3. CONTADOR SIMPLE

29/05/2022

Contenido

INTRODUCCIÓN.....	2
DESARROLLO	2
VISTA	3
CÓDIGO	4
CONCLUSIÓN	5
REPOSITORIO	5

INTRODUCCIÓN

Para esta actividad, se realizará una aplicación utilizando Dart y flutter, en inicializar un contador numérico, con 2 botones, uno para la función de sumar al contador y otro con la función de restar al contador.

Una de las consignas es que el contador no puede disminuir de 0 ni tampoco ser mayor a 50, por lo que se debe evitar que se realicen las operaciones de los botones sobre el contador en estas situaciones.

DESARROLLO

Primeramente, se crea la appbar y colocamos un título en ella, también elegimos los temas que utilizaremos dentro de la aplicación, posterior creamos nuestro cuerpo de la aplicación, la cual consta de un texto “el contador vale”, y debajo de este un texto que mostrara el valor del contador entre 2 botones alusivos, uno para sumar y otro para restar.

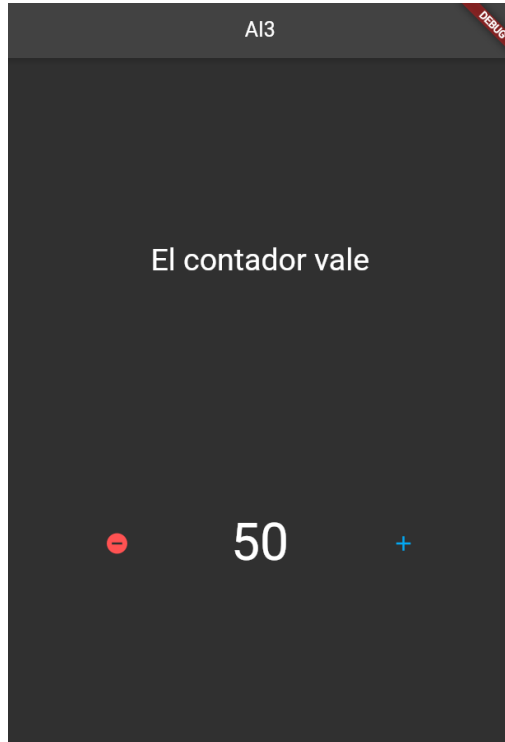
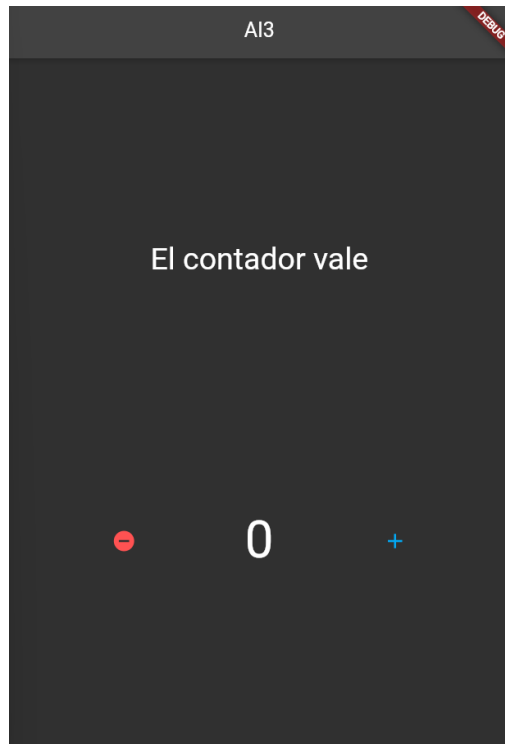
Dentro de nuestro statefulWidget debemos crear nuestra variable “contador” de tipo entero, y la inicializamos en 0, esta variable será mostrada en pantalla en el texto que se encuentra entre los botones.

Para resolver el problema de delimitación de nuestra variable, utilizamos sentencias if, dentro de la guardia de este, se colocan las siguientes condiciones “contador>49” y “contador<1” a evaluar, para que en caso de ser ciertas, no se realice ninguna instrucción, y en caso contrario, dentro del “else”, ejecutar las funciones de suma o resta de nuestros botones. A continuación, se muestran las condicionantes:

```
if (contador > 49) {  
} else {  
    setState(() {  
        contador++;  
    });  
  
if (contador < 1) {  
} else {  
    setState(() {  
        contador--;  
    });  
}
```

VISTA

Las siguientes imágenes muestran el resultado final del desarrollo.



CÓDIGO

```
import 'package:flutter/material.dart';
import 'package:flutter/widgets.dart';

void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: HomePage(),
      darkTheme: ThemeData.dark(),
      theme: ThemeData.light(),
    );
  }
}

class HomePage extends StatefulWidget {
  const HomePage({Key? key}) : super(key: key);

  @override
  State<HomePage> createState() => _HomePageState();
}

class _HomePageState extends State<HomePage> {
  int contador = 0;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("AI3"),
        centerTitle: true,
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.spaceEvenly,
          children: [
            Container(
              child: Column(
                children: [
                  Text(
                    "El contador vale",
                    style: TextStyle(fontSize: 30),
                  ),
                  SizedBox(height: 50),
                ],
              ),
            ),
            Row(
              mainAxisAlignment: MainAxisAlignment.spaceEvenly,
              children: [
```

```

IconButton(
  icon: Icon(Icons.do_not_disturb_on),
  color: Colors.redAccent,
  onPressed: () {
    if (contador < 1) {
    } else {
      setState(() {
        contador--;
      });
    }
  },
),
Text(
  "$contador",
  style: TextStyle(fontSize: 50),
),
IconButton(
  icon: Icon(Icons.add),
  color: Colors.lightBlue,
  onPressed: () {
    if (contador > 49) {
    } else {
      setState(() {
        contador++;
      });
    }
  },
),
],
),
],
),
),
);
}
}
}

```

CONCLUSIÓN

Para la resolución de este problema, con unas sentencias de evaluación “if” podemos indicarle a la aplicación en qué momento ejecutar la funcionalidad de los botones.

REPOSITORIO

https://github.com/laos89/pdm_ai3