

Actividad 2 Introducción

Índice

Desarrollo de la actividad.....	2
Primera vista.....	2
Título.....	2
Primer elemento.....	2
Segundo elemento.....	6
Segunda Vista.....	7
Título.....	7
Primer elemento.....	7
Segundo elemento.....	7
Tercer elemento.....	8
Cuarto elemento.....	8
Quinto elemento.....	9

Desarrollo de la actividad

Primera vista

Para realizar el cuerpo de la primera vista se crea una clase llamada MyHomePage que retorna un Scaffold:

```
class MyHomePage extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    //var appState = context.watch<MyAppSta  
  
    return Scaffold(  
      // ...  
    );  
  }  
}
```

Título

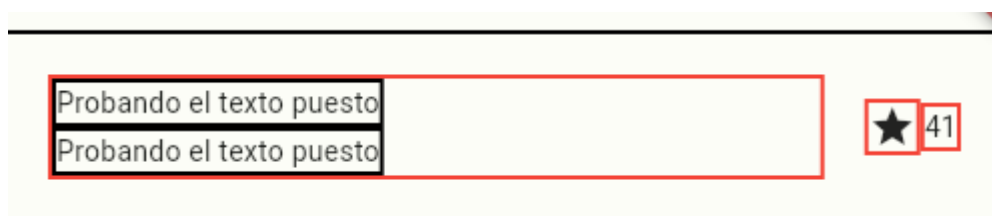
Dentro de nuestro Scaffold empezamos a construir el AppBar. Pondremos el título de la primera vista de manera que quede centrada con un Center.

```
appBar: AppBar(  
  title: Center(child: Text("Vista 1")),  
), // AppBar
```



Primer elemento

Para esta parte he encapsulado los elementos en un Row para que queden de la siguiente manera (marcado en rojo):



```

), // BoxDecoration
child: Row(
  children: [
    Expanded(
      child: Padding(
        padding: const EdgeInsets.all(20.0),
        child: Container(
          decoration: BoxDecoration(
            border: Border.all(
              color: Colors.black,
              width: 2.0
            ) // Border.all
          ), // BoxDecoration
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.start,
            children: [
              Container(
                decoration: BoxDecoration(
                  border: Border.all(
                    color: Colors.black,
                    width: 2.0
                  ) // Border.all
                ), // BoxDecoration
                child: Text("Probando el texto")
              ), // Container
              Container(
                decoration: BoxDecoration(
                  border: Border.all(
                    color: Colors.black,
                    width: 2.0

```

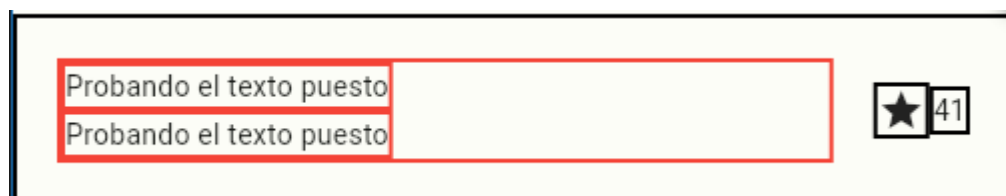
```

    ), // Container
  ], // Column
), // Container
), // Padding
), // Expanded
Container(
  decoration: BoxDecoration(
    border: Border.all(
      color: Colors.black,
      width: 2.0
    ) // Border.all
  ), // BoxDecoration
  child: Icon(
    Icons.star,
  ), // Icon
), // Container
Padding(
  padding: const EdgeInsets.only(right:
  child: Container(
    decoration: BoxDecoration(
      border: Border.all(
        color: Colors.black,
        width: 2.0
      ) // Border.all
    ), // BoxDecoration
    child: Text("41")
  ), // Container
), // Padding
]
), // Row

```

En la segunda imagen aparece un Expanded. Esto lo que hace es expandir el elemento todo el espacio disponible, indistintamente del contenido.

Dentro del row tenemos tres elementos, empezamos por el primero:



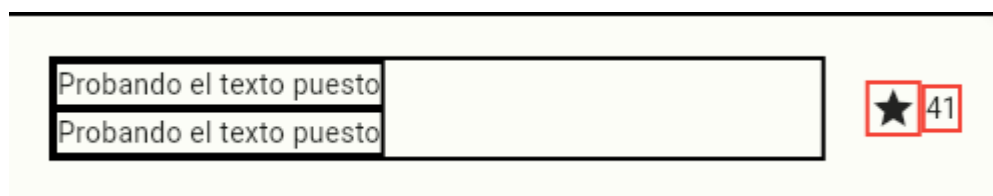
```

child: Column(
  crossAxisAlignment: CrossAxisAlignment.start,
  children: [
    Container(
      decoration: BoxDecoration(
        border: Border.all(
          color: Colors.red,
          width: 2.0
        ) // Border.all
      ), // BoxDecoration
      child: Text("Probando el texto puesto")
    ), // Container
    Container(
      decoration: BoxDecoration(
        border: Border.all(
          color: Colors.red,
          width: 2.0
        ) // Border.all
      ), // BoxDecoration
      child: Text("Probando el texto puesto")
    ), // Container
  ]
)

```

El Widget contiene un Column para indicar que los hijos se pongan en forma de columna. La línea que pone "crossAxisAlignment" es para indicar que queremos poner los elementos que se encuentran dentro de Column partiendo del inicio, es decir, ubicado a la izquierda. Los Containers no sirven para darle una decoración a los widgets, fondo, borde, etc. Los Containers son una especie de capa que se superpone a la que viene a continuación, no es el Row ni el Column el que se decora sino el Container, y el row y el column es contenido por este. Para decorar el contenedor uso BoxDecoration, que tiene una propiedad llamada border, que me permite pintar el borde y el grosor del contenedor.

Por último tenemos el icono y el número:



```

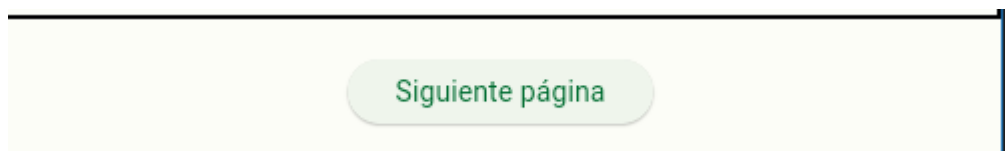
Container(
  decoration: BoxDecoration(
    border: Border.all(
      color: Colors.red,
      width: 2.0
    ) // Border.all
  ), // BoxDecoration
  child: Icon(
    Icons.star,
  ), // Icon
), // Container
Padding(
  padding: const EdgeInsets.only(right: 20.0),
  child: Container(
    decoration: BoxDecoration(
      border: Border.all(
        color: Colors.red,
        width: 2.0
      ) // Border.all
    ), // BoxDecoration
    child: Text("41")
  ), // Container
), // Padding

```

En primer lugar tenemos el icono, que simplemente se dibuja llamando al objeto Icon pasándole por argumento un icono. A continuación tenemos el texto 41. Este texto está encapsulado en un Padding para que haya un espacio entre su extremo derecho y este. De igual manera colocamos Text.

Segundo elemento

El segundo elemento es el botón donde que nos redirigirá a la siguiente vista:



```

), // Container
    SizedBox(height: 20),
    Column(
      children: [
        ElevatedButton(
          onPressed: () {
            Navigator.push(
              context,
              MaterialPageRoute(builder: (context) => PageSecond()),
            );
          },
          child: Text("Siguiente página")
        ) // ElevatedButton
      ],
    ) // Column
  ),
);

```

SizedBox es para que haya un espaciado entre bloque y bloque. Dentro de column insertamos ElevatedButton, widget que nos da un botón ya hecho y que solamente tenemos que indicarle las acciones a realizar. El widget recibe dos argumentos, qué realizar cuando se presiona y qué contenido tiene el botón. En onPressed vamos a indicarle que tiene irse a otra página. Usamos Navigator.push para hacerlo pasándole por parámetro el context (que es el estado en el que se encuentra la página) un MaterialPageRoute, indicando que queremos construir con el context que le pasamos al Widget que contiene el cuerpo de la siguiente página.

Segunda Vista

Título

Para el título usamos el mismo código que en la anterior página:

```

appBar: AppBar(
  title: Center(child: Text("Vista 2")),
), // AppBar

```

Primer elemento

En esta parte hay una imagen sacada de internet.

```

children: [
  Image.network("http://2.bp.blogspot.com/-sm0o6dv4nmQ/TlPxt_mZq6I/A4

```

Segundo elemento

El segundo elemento es idéntico al elemento de la página anterior, pero sin bordes


```

Row(
  children: [
    Expanded(
      child: Padding(
        padding: const EdgeInsets.all(20.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Text("Probando el texto puesto"),
            Text("Probando el texto puesto"),
          ]
        ), // Column
      ), // Padding
    ), // Expanded
    Icon(
      Icons.star,
    ), // Icon
    Padding(
      padding: const EdgeInsets.only(right: 20.0),
      child: Text("41"),
    ), // Padding
  ],
), // Row

```

Tercer elemento

El tercer elemento consta de tres imagenes repartidas por todo el bloque Column:



```

Row(
  mainAxisAlignment: MainAxisAlignment.spaceEvenly,
  children: [
    Image.asset("assets/telefono.png"),
    Image.asset("assets/cursor.png"),
    Image.asset("assets/compartir.png")
  ]
)

```

Con Row estamos indicándole la dirección de posición. En mainAxisAlignment indicamos con spaceEvenly que los elementos hijos se separen dejando el mismo espacio.

Cuarto elemento

El cuarto elemento contiene un texto justificado:


En el vasto y misterioso cosmos que se extiende más allá de nuestra atmósfera, la humanidad ha encontrado tanto un desafío como una fascinación inagotables. Desde los albores de la civilización, hemos levantado la mirada hacia el cielo nocturno, maravillados por las estrellas que parpadean en la oscuridad y los planetas que vagan por el éter. Sin embargo, fue solo en el siglo XX cuando la tecnología y la ambición humana se unieron para llevar a cabo una hazaña que antes solo se consideraba un sueño: la exploración espacial.

```
Padding(  
  padding: const EdgeInsets.symmetric(horizontal: 20),  
  child: Text("En el vasto y misterioso cosmos que se ex  
    textAlign: TextAlign.justify,), // Text  
) // Padding
```

El texto se encuentra dentro de un Padding porque le hemos dado un padding de 20 pixeles a los laterales con `EdgeInsets.symmetric()`. Después colocamos el texto y una propiedad `textAlign` indicando `justify`.

Quinto elemento

Por último insertamos el botón que nos lleva a la página inicial:



Página inicial

```
ElevatedButton(  
  onPressed: () {  
    Navigator.pop(context);  
  },  
  child: Text("Página inicial")  
) // ElevatedButton
```

En este caso, en `onPressed`, vamos a usar la función `.pop` que lo hace es volver al estado anterior, matando la página actual.