

LABORATORIO	02 – About Me
OBJETIVO GENERAL	Utilizar Widgets Básicos
INSTRUCCIONES	Elabora una aplicación que replique el ejemplo del video colocando tu información básica, así como una foto de perfil tuya. Agregando una pequeña descripción de lo que te gusta hacer.
NOMBRE ESTUDIANTE	Itsai Zempoaltecatl Mejia
FECHA	25 febrero 2022
GRUPO	8SA

## RECURSOS

### Revisa la sección de videos, la lectura 02 y lectura 03

Puntos a entregar:

- 1 video de tu aplicación de mínimo 15 segundos donde se aprecie el funcionamiento de la misma (debe ser obligatoriamente emulado en un celular físico o virtual, NO WEB)
- Un reporte donde colocarás partes de tu código explicando que fue lo que desarrollaste, con capturas de pantalla solo de las partes importantes a explicar.

Utiliza estos links que te servirán de apoyo:

- [https://github.com/XeniaEns/cascaron\\_app\\_about\\_me](https://github.com/XeniaEns/cascaron_app_about_me)
  - <https://docs.flutter.dev/cookbook/images/network-image>
  - <https://docs.flutter.dev/release/breaking-changes/theme-data-accent-properties>
  - <https://api.flutter.dev/flutter/widgets/ClipOval-class.html>
-

Cree un Widget llamado Information que es el que contiene todos los Widgets que cree para este ejemplo. Primero con ayuda de Column agregue los componentes en orden vertical, cada componente lo separe en Widgets para modificar uno solo a la vez y no toda la interfaz en conjunto.

```
class Information extends StatelessWidget {
  const Information({
    Key? key,
  }) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Column(
      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
      children: const [
        PhotoProfile(),
        Name(),
        UserName(),
        Description(),
      ],
    ); // Column
  }
}
```



**PhotoProfile** define a la imagen de perfil, consta de un Image para mostrar la imagen contenido en un ClipOval le da el efecto ovalado que vemos en la foto, contenido en un Padding, esto es para reducir un poco el tamaño de la imagen ya que se mostraba muy grande.

```
class PhotoProfile extends StatelessWidget {
  const PhotoProfile({
    Key? key,
  }) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.all(20),
      child: ClipOval(
        child: Image.network(
          'https://lh3.googleusercontent.com/...',
        ), // ClipOval // Padding
      ),
    );
  }
}
```

**Name** es un Container con un Center que a su vez tiene un Text, el Container es usado para darle ese efecto que el texto esta dentro de un rectángulo y Center centra este Text con mi nombre y una fuente extraiga de GoogleFonts.

```
class Name extends StatelessWidget {
  const Name({
    Key? key,
  }) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Container(
      color: Colors.cyan,
      width: double.infinity,
      height: 50,
      child: Center(
        child: Text('Itsai Z. Mejia',
          style: GoogleFonts.robotoMono(fontSize: 30)),
      ), // Center // Container
    );
  }
}
```

**UserName** como en el componente anterior este es similar, ya que es un Container para el efecto del rectángulo, pero cuenta con Row que permite agregar elementos horizontalmente, lo use para agregar un Icon y Text con mi usuario y la fuente de GoogleForms.

```
class UserName extends StatelessWidget {
  const UserName({
    Key? key,
  }) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Container(
      color: Colors.cyan,
      width: double.infinity,
      height: 50,
      child: Row(
        mainAxisAlignment: MainAxisAlignment.spaceEvenly,
        children: [
          const Icon(
            Icons.people,
            color: Colors.black,
            size: 24.0,
          ), // Icon
          Text('@itsaimejia', style: GoogleFonts.robotoMono(fontSize: 30))
        ],
      )); // Row // Container
  }
}
```

Description por ultimo este widget es el que contiene el Text con mi descripción y la misma fuente de GoogleFonts, usar SizedBox no era necesario, pero investigando un poco en este caso era mas recomendable que Container, por regla del lenguaje únicamente. Text esta contenido en un Center para centrar el texto (creo no se nota), Center en SizedBox y este ultimo en un Padding, este lo use para agregar un espacio arriba (por eso .only).

```
class Description extends StatelessWidget {
  const Description({
    Key? key,
  }) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: EdgeInsets.only(top: 10),
      child: SizedBox(
        width: double.infinity,
        child: Center(
          child: Text(
            'Me gusta todo tipo de musica, la comida salada,
            style: GoogleFonts.robotoMono(fontSize: 20))),
        )); // Padding
  }
}
```

Nota: Todos los componentes cuentan con un width: double.infinity para que dimensionen al tamaño del Padding principal que esta en Body

```
body: const SizedBox(  
  height: double.infinity,  
  width: double.infinity,  
  child: Padding(  
    padding:  
      EdgeInsets.only(left: 70, right: 70, top: 20, bottom: 50),  
    child: Information(),  
  )); // Padding // SizedBox // Scaffold
```