

# Módulo 4: Elementos Visuales y Flujo de Interacción entre Pantallas

## Lección: Selección de Valores o Acciones de una Lista en Flutter: Uso del ListTile

### Objetivos de la Lección

- Comprender el uso del widget `ListTile` para crear elementos de lista con título, subtítulo e íconos.
- Aprender a personalizar un `ListTile` para representar opciones o acciones en una lista.
- Conocer las propiedades y funcionalidades principales del `ListTile`.
- Aplicar ejemplos básicos de `ListTile` en Flutter.

### Introducción de la Lección

El widget `ListTile` en Flutter es una forma rápida y sencilla de crear elementos de lista que incluyen un título, un subtítulo y opciones para íconos o widgets adicionales a la izquierda y derecha del elemento. Este widget es útil para representar una lista de opciones o acciones donde cada `ListTile` puede responder a eventos como toques. En esta lección, exploraremos las propiedades y la personalización del `ListTile`, y aprenderemos cómo crear listas interactivas y bien organizadas.

### Funcionalidades y Propiedades Principales del ListTile

- **title:** Define el texto principal o título del `ListTile`.
- **subtitle:** Texto adicional que aparece debajo del título.

- **leading:** Widget que se muestra a la izquierda del ListTile, comúnmente un ícono o imagen.
- **trailing:** Widget que se muestra a la derecha del ListTile, ideal para íconos de acción o indicadores.
- **onTap:** Callback que se ejecuta cuando el usuario toca el ListTile, permitiendo definir acciones.
- **selected:** Define si el ListTile está seleccionado, lo que permite cambiar su estilo visual.

## Ejemplo Simple de ListTile

En este ejemplo, creamos una lista de ListTile que representa configuraciones comunes en una aplicación, como "Perfil", "Notificaciones" y "Tema". Cada elemento de lista incluye un título, un ícono y una acción cuando el usuario lo selecciona.

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

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

```
class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: ListTileExample(),
    );
  }
}
```

```
class ListTileExample extends StatelessWidget {
```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text('Ejemplo de ListTile'),
    ),
    body: ListView(
      children: [
        ListTile(
          leading: Icon(Icons.person),
          title: Text('Perfil'),
          subtitle: Text('Ver y editar tu perfil'),
          trailing: Icon(Icons.arrow_forward_ios),
          onTap: () {
            print("Perfil seleccionado");
          },
        ),
        Divider(),
        ListTile(
          leading: Icon(Icons.notifications),
          title: Text('Notificaciones'),
          subtitle: Text('Configurar notificaciones'),
          trailing: Icon(Icons.arrow_forward_ios),
          onTap: () {
            print("Notificaciones seleccionado");
          },
        ),
        Divider(),
        ListTile(
          leading: Icon(Icons.brightness_6),

```

```

        title: Text('Tema'),
        subtitle: Text('Seleccionar tema de la aplicación'),
        trailing: Icon(Icons.arrow_forward_ios),
        onTap: () {
            print("Tema seleccionado");
        },
    ),
],
),
);
}
}

```

En este ejemplo:

- **title** y **subtitle** definen el texto principal y el texto descriptivo de cada `ListTile`.
- **leading** muestra un ícono a la izquierda para representar la opción (perfil, notificaciones, tema).
- **trailing** utiliza un ícono de flecha a la derecha, indicando que el elemento puede llevar a otra pantalla.
- **onTap** permite que cada `ListTile` reaccione al toque del usuario, imprimiendo la opción seleccionada en la consola.

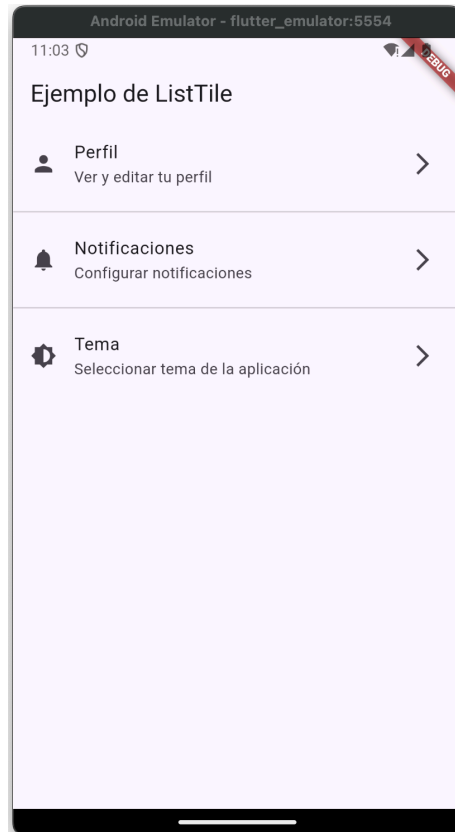


Imagen de la pantalla que incluye el Widget ListTile.

Creado por Javier A. Dastas (2024)

## Personalización de ListTile para Selección

Para indicar que un ListTile está seleccionado, se puede usar la propiedad `selected`. En el siguiente ejemplo, un ListTile cambia su apariencia cuando está seleccionado:

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

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

```

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: ListTileSelectedExample(),
    );
  }
}

class ListTileSelectedExample extends StatefulWidget {
  @override
  _ListTileSelectedExampleState createState() =>
  _ListTileSelectedExampleState();
}

class _ListTileSelectedExampleState extends
State<ListTileSelectedExample> {
  int _selectedIndex = -1;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Ejemplo de ListTile Seleccionado'),
      ),
      body: ListView.builder(
        itemCount: 5,
        itemBuilder: (context, index) {
          return ListTile(
            title: Text('Opción ${index + 1}'),

```

```

        selected: _selectedIndex == index,
        selectedTileColor:
Colors.lightBlueAccent.withOpacity(0.3),
        onTap: () {
            setState(() {
                _selectedIndex = index;
            });
        },
    );
},
),
);
}
}

```

En este ejemplo:

- **selected** define si un `ListTile` está seleccionado, y **selectedTileColor** permite cambiar el color del `ListTile` cuando se selecciona.
- Al tocar un elemento, el índice de selección (`_selectedIndex`) cambia, y se actualiza el color del `ListTile` seleccionado.

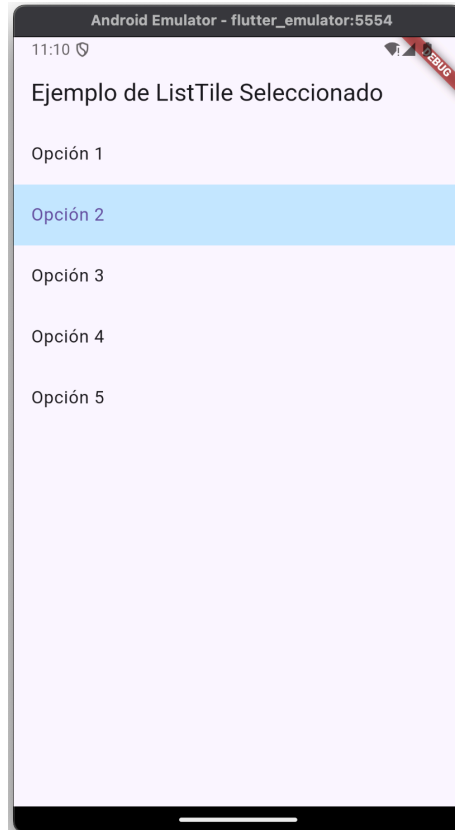


Imagen de la pantalla que incluye el Widget ListTile personalizado.

Creado por Javier A. Dastas (2024)

## Relación con Otros Conceptos

El widget `ListTile` es una herramienta clave para estructurar listas de opciones en Flutter, ofreciendo una experiencia de usuario organizada y fácil de navegar. Este widget se utiliza comúnmente junto con `ListView` para crear listas extensibles de opciones, ideal para configuraciones, menús y selecciones de acción. Su implementación es importante para la creación de aplicaciones centradas en la UX/UI, donde la organización y claridad de las opciones es esencial.



## Resumen de la Lección

En esta lección, aprendimos a usar el widget `ListTile` en Flutter para crear listas de opciones o acciones. Vimos cómo personalizar el `ListTile` con título, subtítulo, íconos, y reacciones al toque del usuario. También exploramos cómo resaltar un `ListTile` cuando está seleccionado. Este widget es útil para aplicaciones que requieren listas de opciones, configuraciones o acciones.

## Actividad de la Lección

En esta actividad, aplicaremos lo aprendido sobre el widget `ListTile` para crear una lista de opciones que permita al usuario seleccionar configuraciones o acciones en una aplicación. Esta actividad ayuda a consolidar el uso de listas interactivas y opciones seleccionables en Flutter.

La actividad tiene dos partes:

### 1. Crear Lista de Configuraciones con `ListTile`:

- Implementa una lista de `ListTile` que represente opciones de configuración (por ejemplo, perfil, notificaciones, y preferencias).
- Incluye un título, subtítulo y un ícono a la izquierda para cada opción.
- Al tocar cada opción, el nombre de la configuración debe imprimirse en la consola.

### 2. Resaltar `ListTile` Seleccionado:

- Permite que uno de los `ListTile` esté seleccionado y resáltalo con un color diferente.
- Al tocar una opción, el `ListTile` seleccionado debe cambiar visualmente, permitiendo al usuario ver qué opción está activa.
- Esta parte permite trabajar con la selección y personalización de `ListTile` en listas interactivas.

### Entrega de la Actividad:

- Desarrolla un documento en formato PDF que incluya el código desarrollado para esta actividad, capturas de pantalla de la aplicación en ejecución y una breve explicación de cómo `ListTile` mejora la experiencia de usuario en la selección de opciones y acciones.

Con esta actividad, se busca consolidar el conocimiento en la creación de interfaces interactivas que faciliten la organización y selección de opciones, mejorando la experiencia de usuario en aplicaciones Flutter.