

**DESARROLLO DE APP MOVILES I**  
**GONZALEZ DIAZ ANSELMO ALEXIS**  
**Aarón Hernández García**

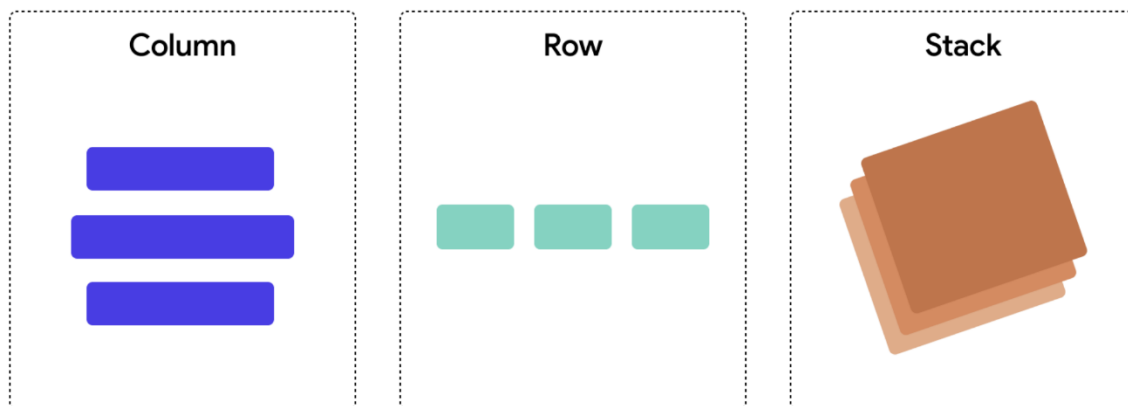
**Layouting**

**23/01/2024**

## Stack, Column and Row

En Flutter Rows, Columns, y Stacks son widgets de diseño fundamentales que te ayudan a estructurar la interfaz de usuario organizando otros widgets en diferentes disposiciones visuales. Así es como funciona cada uno:

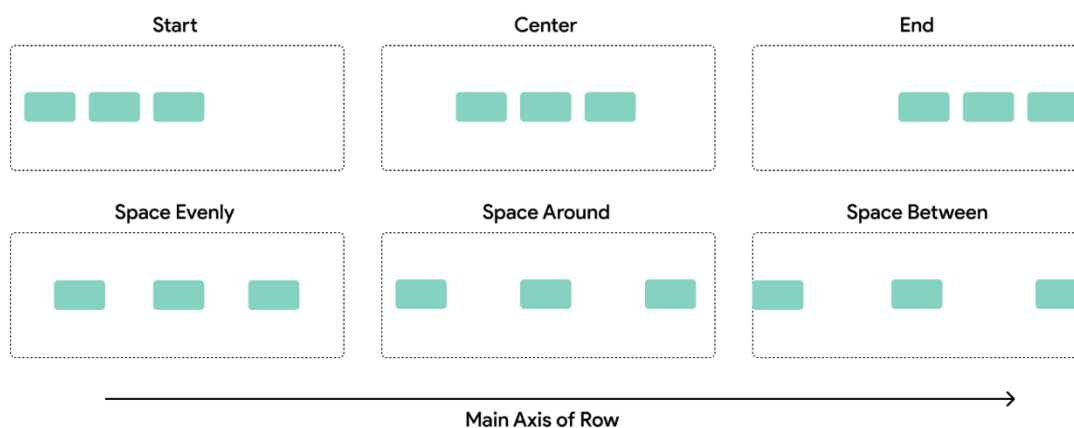
- **Stack:** apila sus widgets secundarios uno sobre otro, lo que permite superponer elementos. En una pila Stack, los widgets se pueden posicionar de forma absolutamente relativa a los bordes de la pila Stack, lo que le permite controlar la ubicación exacta de cada elemento.
- **Column:** organiza sus widgets secundarios verticalmente, apilándolos de arriba a abajo. Esto es ideal para colocar elementos que deben aparecer en una secuencia vertical, como una lista de mensajes en una aplicación de chat o entradas en un formulario.
- **Row:** organiza sus widgets secundarios en una línea horizontal. Esto resulta útil cuando desea colocar elementos uno al lado del otro en la pantalla.



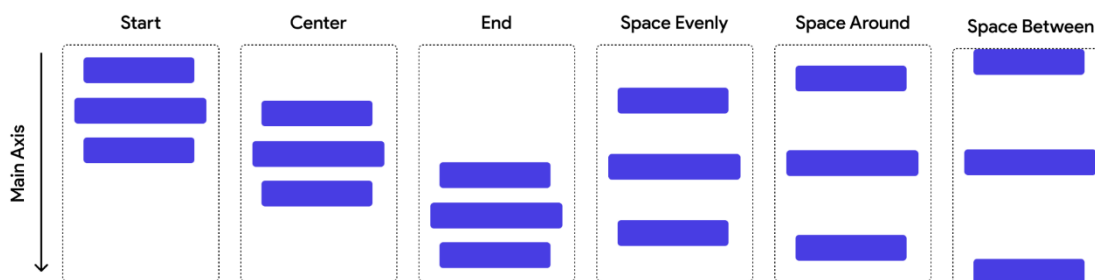
## Configuraciones Generales

**Main Axis:** El eje principal es la dirección principal en la que se disponen los widgets secundarios en Row o Column.

**Row:** El eje principal se extiende **horizontalmente**. Los widgets secundarios se organizan de izquierda a derecha.

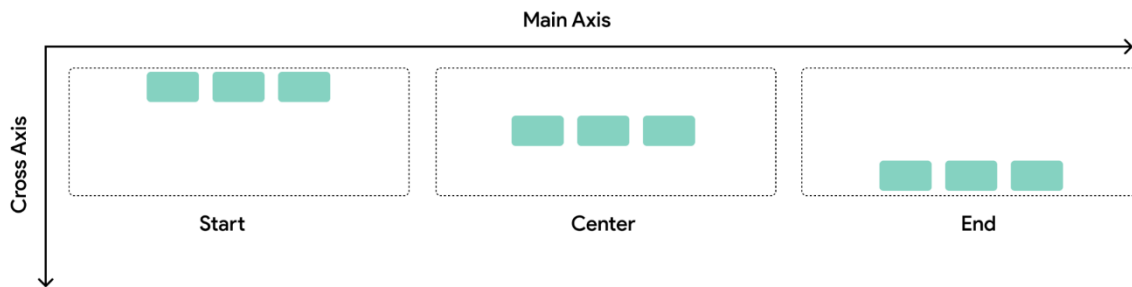


**Column:** El eje principal se extiende **verticalmente**. Los widgets secundarios se disponen de arriba a abajo.

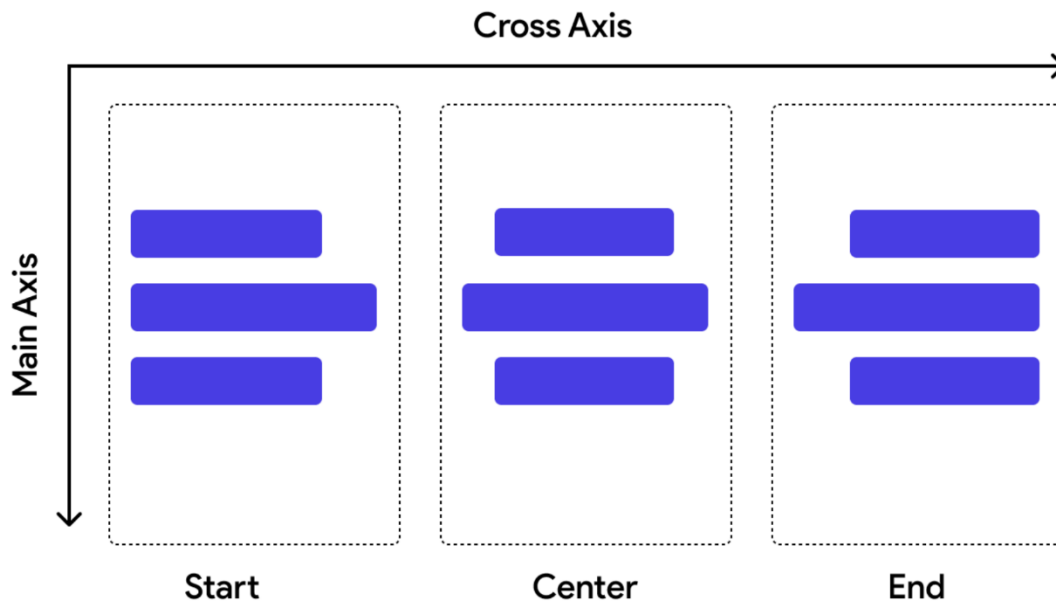


**Cross Axis:** El eje transversal es **perpendicular al eje principal** y controla la alineación y el espaciado de los widgets en esta dirección secundaria. Tiene los siguientes tipos: Inicio, Centro, Fin.

**Row:** el eje transversal se extiende **verticalmente** y determina cómo se alinean los widgets secundarios de arriba a abajo dentro de la fila.



**Column:** el eje transversal se extiende **horizontalmente** . Controla cómo se alinean los widgets secundarios de izquierda a derecha dentro de la columna.



Para **Stacks**, el concepto de ejes principal y transversal es menos aplicable porque los widgets están alineados en relación con toda el área de Stack. En FlutterFlow, puedes controlar la Stackalineación de los elementos secundarios mediante la Stackpropiedad denominada *Alineación de elementos secundarios predeterminada* , que posiciona a los elementos secundarios mediante las coordenadas Xy Y.

## Ejemplos aplicables

### Stack

- **Situación:** Diseñar una tarjeta de presentación que muestre el nombre y el puesto de una persona superpuestos sobre su foto de perfil.
- **Situación:** Crear un diseño de fondo donde un ícono central está decorado con múltiples formas geométricas detrás de él.
- **Situación:** Mostrar una notificación flotante (como un contador de mensajes no leídos) sobre el ícono de una aplicación.

### Column

- **Situación:** Construir una pantalla de registro que tenga un campo para ingresar el nombre, correo y contraseña, apilados verticalmente.
- **Situación:** Diseñar una página de perfil que tenga la foto del usuario en la parte superior, seguida de su biografía y botones para seguir o enviar mensaje.
- **Situación:** Crear un diseño de factura con un encabezado, una lista de productos y un total, colocados de arriba hacia abajo.

### Row

- **Situación:** Crear un menú de navegación horizontal con íconos (Inicio, Buscar, Perfil) alineados uno al lado del otro.

- **Situación:** Diseñar una barra de acciones con botones para "Guardar", "Cancelar" y "Eliminar" en una fila horizontal.
- **Situación:** Mostrar una fila de imágenes de perfil en miniatura para indicar los participantes en un grupo.

## Código

### Stack:

```
// Cambiar el contenido según el ejemplo seleccionado
switch (_selectedExample) {
  case 1: // Ejemplo de Stack
    content = Stack(
      alignment: Alignment.center,
      children: [
        Container(
          width: 150,
          height: 150,
          color: Colors.blue,
        ), // Container
        const Text(
          'Ej.1: Stack',
          style: TextStyle(color: Colors.white, fontSize: 24),
        ), // Text
      ],
    ); // Stack
    break;
```

## Column:

```
case 2: // Ejemplo de Column
  content = Column(
    mainAxisAlignment: MainAxisAlignment.center,
    children: const [
      Text('Ejemplo 2', style: TextStyle(fontSize: 20)),
      Text('Column', style: TextStyle(fontSize: 20)),
      Text('Aaron Hdez', style: TextStyle(fontSize: 20)),
    ],
  ); // Column
  break;
```

## Row:

```
case 3: // Ejemplo de Row
  content = Row(
    mainAxisAlignment: MainAxisAlignment.spaceAround,
    children: const [
      Icon(Icons.star, size: 40, color: Colors.yellow),
      Icon(Icons.favorite, size: 40, color: Colors.red),
      Icon(Icons.thumb_up, size: 40, color: Colors.blue),
    ],
  ); // Row
  break;
```

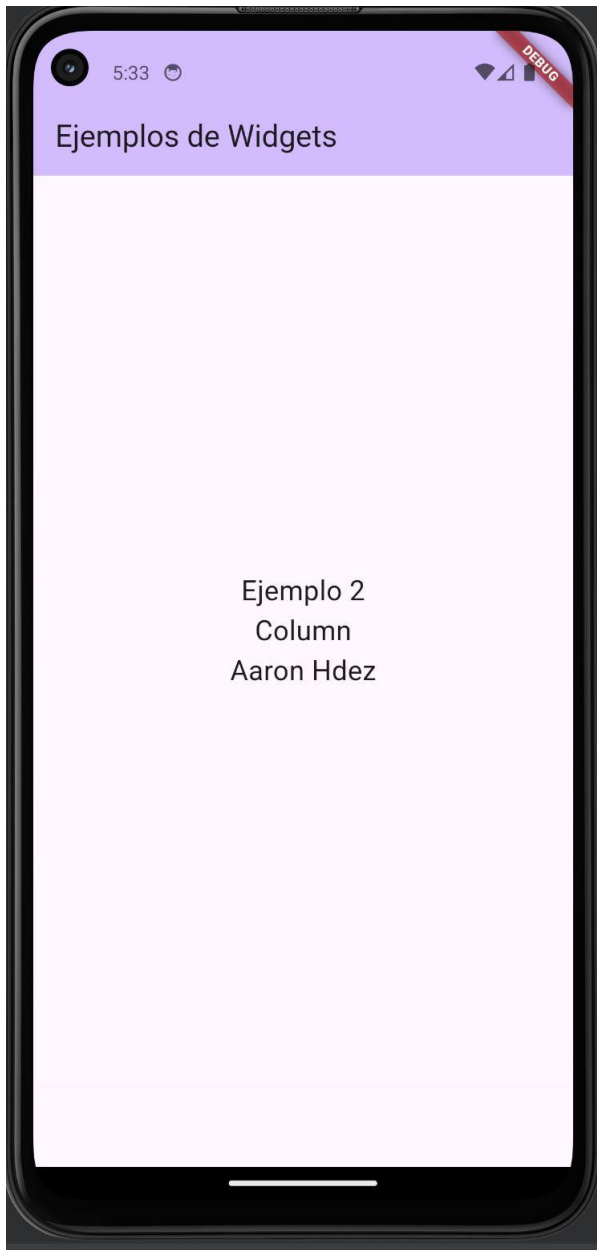
## Ejemplos Emulados:

### Stack:





## Column:



Row:

