

Intro to Frontend Development with Flutter



Ben Cheng

RISD ID

May 1, 2024

Outline I

- 1 Layout and Design**
- 2 Interaction**
- 3 Navigation**
- 4 Animation**

1 Layout and Design

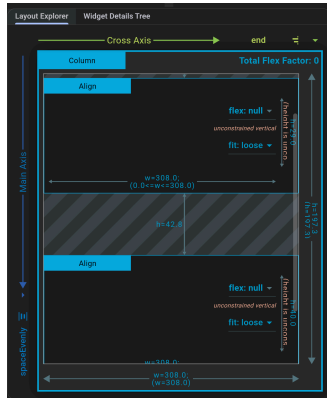
2 Interaction

3 Navigation

4 Animation

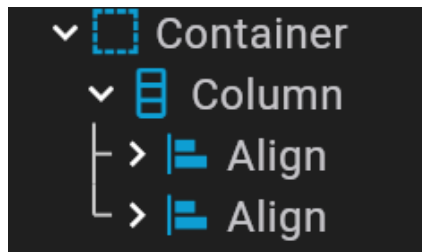
Everything is a widget

and widgets are wrapped by widgets



Layout

- Column
- Row
- Flex
- Stack
- Spacer
- Align



Container()

```
140 Container(  
141     padding: const EdgeInsets.all(12),  
142     decoration: BoxDecoration(  
143         borderRadius: const BorderRadius.all(Radius.circular(10)),  
144         color: Colors.deepPurple.shade100,  
145     ),
```

- BoxDecoration: color, border, border radius, shadow, image, gradient, shape
- An alternative when you want to restrict size: SizedBox()

Google Fonts

```
flutter pub add google_fonts
```

```
3 import 'package:google_fonts/google_fonts.dart';
```

```
100 Text(  
101     "Food Picker",  
102     style: GoogleFonts.robotoSlab(  
103         textStyle: const TextStyle(  
104             fontWeight: FontWeight.w900, fontSize: 80, height: 1.1))  
105         ,  
106     )
```

1 Layout and Design

2 **Interaction**

3 Navigation

4 Animation

Callback

```
114 ElevatedButton(  
115     child: const Text("Add"),  
116     onPressed: () => {  
117         _addOption( // a custom callback function declared earlier  
118             value: _optionTextEditingController.value.text)  
119     },  
120 ))
```

setState

Declaration of the custom call back function:

```
114 void _addOption({required String value}) {  
115     setState(() {  
116         _options.add(value);  
117         _optionTextEditingController.clear();  
118     });  
119 }
```

Update Where?

```
151 Text(  
152     _options[index], // actual text determined by the data in the  
        list_options  
153     style: GoogleFonts.robotoSlab(  
154         textStyle: const TextStyle(fontSize: 20)),  
155     )
```

1 Layout and Design

2 Interaction

3 Navigation

4 Animation

Navigator

- Routes (screens) are arranged as a stack.
- Push or pop to move between screens.

```
80 FloatingActionButton.extended(  
81     onPressed: () => {  
82         if (_options.isNotEmpty)  
83             {  
84                 Navigator.push(context, MaterialPageRoute(builder: (  
85                     context) {  
86                         var choice = Random().nextInt(_options.length);  
87                         return Result(result: _options[choice]);  
88                     })))  
89             },
```

Other Navigation Tools

- Page
- Routes
- Tabs
- (Deeplinks)

1 Layout and Design

2 Interaction

3 Navigation

4 **Animation**

GestureDetector()

Callbacks: onTapDown, onTapUp

```
201 GestureDetector(  
202     behavior: HitTestBehavior.opaque,  
203     onTapDown: (details) {  
204         setState(() {  
205             scale = 1.075;  
206         });  
207     },  
208     onTapUp: (details) {  
209         setState(() {  
210             scale = 1;  
211         });  
212     },
```

Implicit Animation

- 1 AnimatedContainer()
- 2 Some properties of the widget are not constant
- 3 Animation automatically triggered when properties change

```
219 AnimatedContainer(  
220     duration: Durations.short2,  
221     height: scale * 400, // non-constant height  
222     width: MediaQuery.of(context).size.width * 0.75 * scale,  
        // non-constant width  
223     child: Center(  
224         child: Text(  
225             widget.result,  
226             style: GoogleFonts.robotoSlab(  
227                 textStyle: const TextStyle(fontSize: 40)),  
228             )),
```

Explicit Animation

When

- You want to “manually” trigger animation.
- You want the animation to run forever.
- You want different animation (duration, curve) for different property.
- Your animation has “discontinuities”.