



Build Your IT Skill

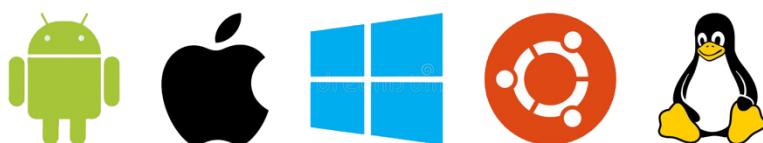
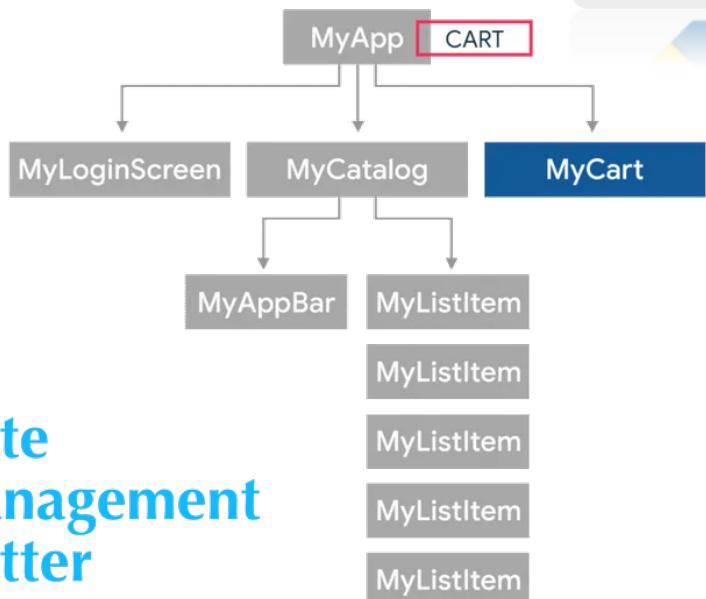
ETEC IT PROFESSIONAL TRAINING CENTER

Flutter Statemanagement

- BLOC
- GETX



State
Management
Flutter



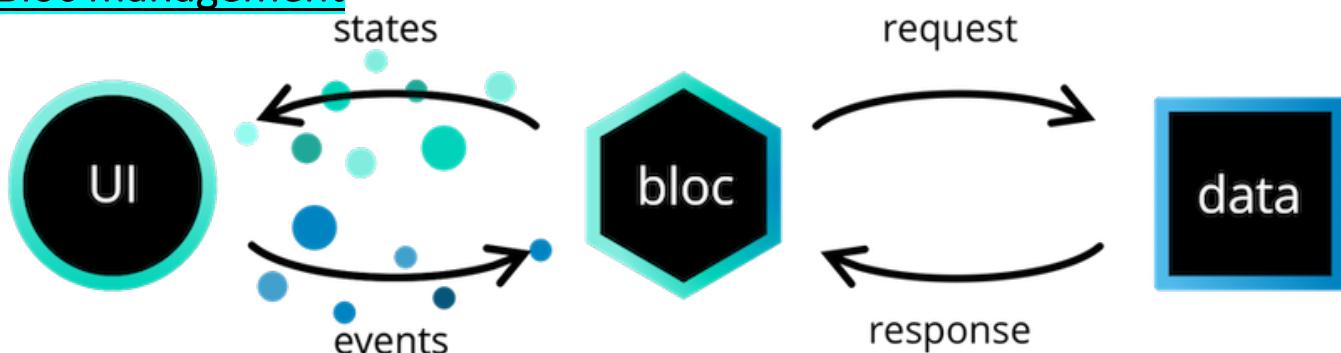
ស្ថិកស្ថិតិថាយៈ និង អ៊ូលុន
Software Developer

About StateManagement

StateManagement គឺជាការ កាត់ផ្ទាយ ហើយ Manage រវាងData នឹង UI ដែលអាចបង្កើតឡើង

- អាចធ្វើឡើង StatelessWidget អាច Change State របស់ UI បាន
- អាច initialize data
- កាត់បន្ថយការប្រើប្រាស់ setState()
- អាចប្រើប្រាស់ Provider & Multi-Provideretc.

I. Bloc management



- Package

```
dependencies:  
  flutter_bloc: ^8.1.1
```

- State

```
part of 'counter_bloc.dart';

class CounterState {
  CounterState({required this.counter});
  int counter;
}

class CounterInitial extends CounterState {
  CounterInitial() : super(counter: 0);
}
```

- Event

```

part of 'counter_bloc.dart';

abstract class CounterEvent extends Equatable {
  const CounterEvent();

  @override
  List<Object> get props => [];
}

class IncrementEvent extends CounterEvent {}
class DecrementEvent extends CounterEvent {}

```

- Bloc

```

import 'dart:io';

import 'package:bloc/bloc.dart';
import 'package:equatable/equatable.dart';
part 'counter_event.dart';
part 'counter_state.dart';

class CounterBloc extends Bloc<CounterEvent, CounterState> {
  CounterBloc() : super(CounterInitial()) {
    on<IncrementEvent>((event, emit) {
      emit(CounterState(counter: state.counter + 1));
    });
    on<DecrementEvent>((event, emit) {
      emit(CounterState(counter: state.counter - 1));
    });
  }
}

```

Part Screen1

```
main.dart — blocError
main.dart X counter_bloc.dart counter_event.dart
```

lib > main.dart > main

```
1 import 'package:bloc_counter2_4/bloc/counter_bloc.dart';
2 import 'package:flutter/material.dart';
3 import 'package:flutter_bloc/flutter_bloc.dart';
4
5 Run | Debug | Profile
6 void main() {
7   Bloc.observer = AppBlocObserver();
8   runApp(const MyApp());
9 }
```

```
10 class AppBlocObserver extends BlocObserver {
11   @override
12   void onChange(BlocBase bloc, Change change) {
13     super.onChange(bloc, change);
14     if (bloc is Cubit) print(change);    Avoid `print` calls in production code.
15   }
16
17   @override
18   void onTransition(Bloc bloc, Transition transition) {
19     super.onTransition(bloc, transition);
20     print(transition);    Avoid `print` calls in production code.
21   }
22 }
```

```
23
24 class MyApp extends StatelessWidget {
25   /// {@macro app}
26   const MyApp({Key? key}) : super(key: key);
27
28   @override
29   Widget build(BuildContext context) {
30     return BlocProvider(
31       create: (_) => CounterBloc(),
32       child: const AppView(),
33     ); // BlocProvider
34   }
35 }
36 
```



```

37 class AppView extends StatelessWidget {
38   const AppView({Key? key}) : super(key: key);
39
40   @override
41   Widget build(BuildContext context) {
42     return BlocBuilder<CounterBloc, CounterState>(
43       builder: (context, state) {
44         return const MaterialApp(
45           color: Colors.blueAccent,
46           home: MyHomePage(title: 'Counter Blog'),
47         ); // MaterialApp
48       },
49     ); // BlocBuilder
50   }
51 }
52
53
54 class MyHomePage extends StatefulWidget {
55   const MyHomePage({Key? key, required this.title}) : super(key: key);
56
57   final String title;
58
59   @override
60   State<MyHomePage> createState() => _MyHomePageState();
61 }
62
63 class _MyHomePageState extends State<MyHomePage> {
64   @override
65   Widget build(BuildContext context) {
66     return Scaffold(
67       appBar: AppBar(
68         title: Text(widget.title),
69         actions: [
70           MaterialButton(
71             onPressed: () {
72               Navigator.push(
73                 context,
74                 MaterialPageRoute(
75                   builder: (context) => const SecondScreen(),
76                 )); // MaterialPageRoute
77           },
78         ],
79       ),
80     );
81   }
82 }

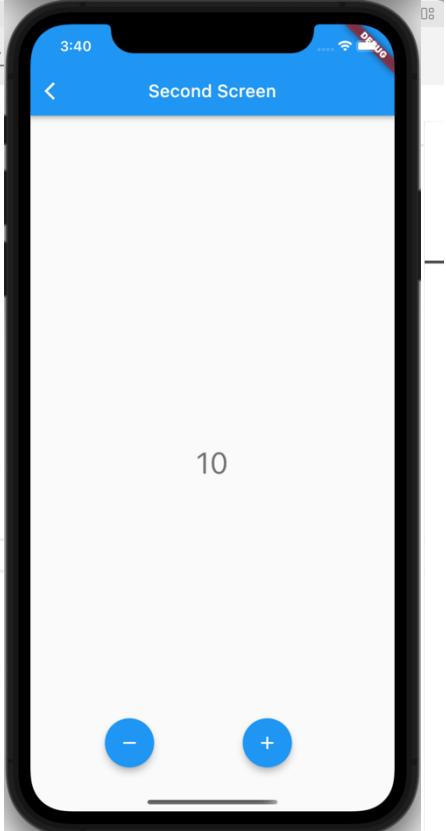
```

```

78         child: const Text(
79             'Next',
80             style: TextStyle(fontSize: 28),
81         ), // Text
82     ), // MaterialButton
83     const SizedBox(
84         width: 40,
85     ) // SizedBox
86 ],
87 ), // AppBar
88 body: BlocBuilder<CounterBloc, CounterState>(
89     builder: (context, state) {
90         return Center(
91             child: Column(
92                 mainAxisAlignment: MainAxisAlignment.center,
93                 children: <Widget>[
94                     Text(
95                         '${state.counter}',
96                         style: Theme.of(context).textTheme.headline4,
97                     ), // Text
98                 ], // <Widget>[]
99             ), // Column
100         ); // Center
101     },
102 ), // BlocBuilder
103 floatingActionButton: Row(
104     mainAxisAlignment: MainAxisAlignment.spaceEvenly,
105     children: [
106         FloatingActionButton(
107             onPressed: () {
108                 context.read<CounterBloc>().add(DecrementEvent());
109             },
110             tooltip: 'Decrement',
111             child: const Icon(Icons.remove),
112         ), // FloatingActionButton
113         FloatingActionButton(
114             onPressed: () {
115                 context.read<CounterBloc>().add(IncrementEvent());
116             },
117             tooltip: 'Increment',
118             child: const Icon(Icons.add),
119         ), // FloatingActionButton
120     ],
121 ), // Row
122 ); // Scaffold
123 }
124
125

```

Part Screen 2



```
lib > second_screen.dart > SecondScreen > build
  1 import 'package:bloc_counter2_4/bloc/counter_bloc.dart';
  2 import 'package:flutter/material.dart';
  3 import 'package:flutter_bloc/flutter_bloc.dart';
  4
  5 class SecondScreen extends StatelessWidget {
  6   const SecondScreen({Key? key}) : super(key: key);
  7
  8   @override
  9   Widget build(BuildContext context) {
 10     return Scaffold(
 11       appBar: AppBar(
 12         title: const Text('Second Screen'),
 13       ), // AppBar
 14       body: BlocBuilder<CounterBloc, CounterState>(
 15         builder: (context, state) {
 16           return Center(
 17             child: Column(
 18               mainAxisAlignment: MainAxisAlignment.center,
 19               children: <Widget>[
 20                 Text(
 21                   '${state.counter}',
 22                   style: Theme.of(context).textTheme.headline4,
 23                 ), // Text
 24               ], // <Widget>[]
 25             ), // Column
 26           ); // Center
 27         },
 28       ), // BlocBuilder
 29       floatingActionButton: Row(
 30         mainAxisAlignment: MainAxisAlignment.spaceEvenly,
 31         children: [
 32           FloatingActionButton(
 33             onPressed: () {
 34               context.read<CounterBloc>().add(DecrementEvent());
 35             },
 36             tooltip: 'Decrement',
 37             child: const Icon(Icons.remove),
 38           ), // FloatingActionButton
 39           FloatingActionButton(
 40             onPressed: () {
 41               context.read<CounterBloc>().add(IncrementEvent());
 42             },
 43             tooltip: 'Increment',
 44             child: const Icon(Icons.add),
 45           ), // FloatingActionButton
 46         ], // Row
 47       ), // Scaffold
 48     );
 49   }
 50 }
```

in* ⌂ ① ② ③ ④ ⌂ Spaces: 2 UTF-8 LF Dart Dart DevTools Flutter: 3.0.4 iPhone 11 (ios simulator) ⌂ ⌂

II. GetX management

```
dependencies:  
  get: ^4.6.5
```

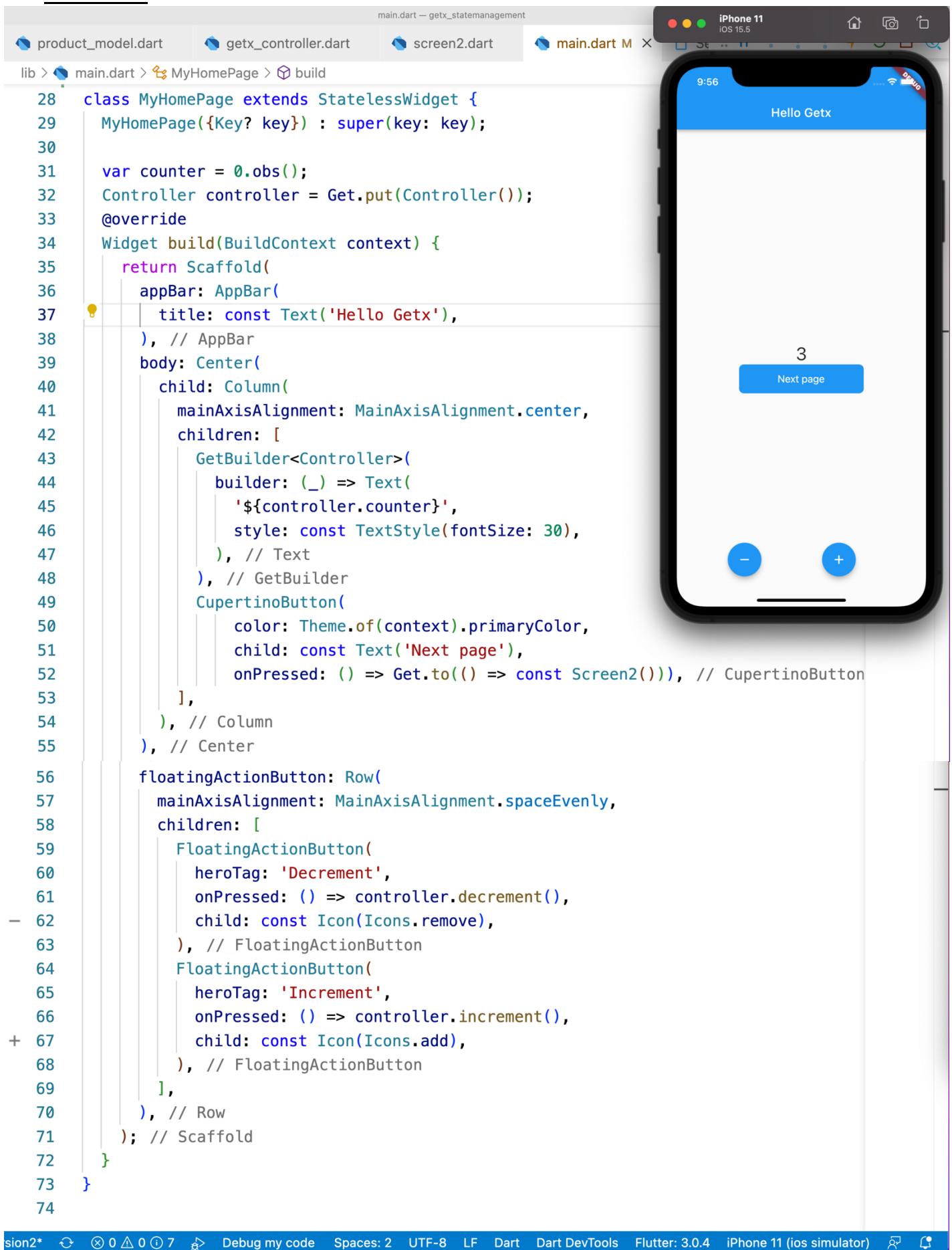
- Controller

```
class Controller extends GetxController {  
  var counter = 0.obs();  
  increment() {  
    counter = counter + 1;  
    update();  
  }  
  
  decrement() {  
    counter = counter - 1;  
    update();  
  }  
}
```

- MaterialApp

```
void main() {  
  runApp(const MyApp());  
}  
  
class MyApp extends StatelessWidget {  
  const MyApp({Key? key}) : super(key: key);  
  
  @override  
  Widget build(BuildContext context) {  
    return GetMaterialApp(  
      title: 'Flutter Demo',  
      theme: ThemeData(  
        primarySwatch: Colors.blue,  
      ),  
      home: MyHomePage(),  
    );  
  }  
}
```

● Scaffold



The screenshot shows a Flutter development environment. On the left, the code editor displays the `main.dart` file, which contains the following code:

```
main.dart — getx_statemanagement
lib > main.dart > MyHomePage > build
28 class MyHomePage extends StatelessWidget {
29   MyHomePage({Key? key}) : super(key: key);
30
31   var counter = 0.obs();
32   Controller controller = Get.put(Controller());
33   @override
34   Widget build(BuildContext context) {
35     return Scaffold(
36       appBar: AppBar(
37         title: const Text('Hello Getx'),
38       ), // AppBar
39       body: Center(
40         child: Column(
41           mainAxisAlignment: MainAxisAlignment.center,
42           children: [
43             GetBuilder<Controller>(
44               builder: (_) => Text(
45                 '${controller.counter}',
46                 style: const TextStyle(fontSize: 30),
47               ), // Text
48             ), // GetBuilder
49             CupertinoButton(
50               color: Theme.of(context).primaryColor,
51               child: const Text('Next page'),
52               onPressed: () => Get.to(() => const Screen2()), // CupertinoButton
53             ],
54           ), // Column
55         ), // Center
56         floatingActionButton: Row(
57           mainAxisAlignment: MainAxisAlignment.spaceEvenly,
58           children: [
59             FloatingActionButton(
60               heroTag: 'Decrement',
61               onPressed: () => controller.decrement(),
62               child: const Icon(Icons.remove),
63             ), // FloatingActionButton
64             FloatingActionButton(
65               heroTag: 'Increment',
66               onPressed: () => controller.increment(),
67               child: const Icon(Icons.add),
68             ), // FloatingActionButton
69           ],
70         ), // Row
71       ); // Scaffold
72     }
73   }
74 }
```

The right side of the interface shows an iPhone 11 simulator running the application. The screen displays "Hello Getx" at the top, a blue button labeled "Next page", and two floating action buttons at the bottom labeled "-" and "+".

Example2 with List

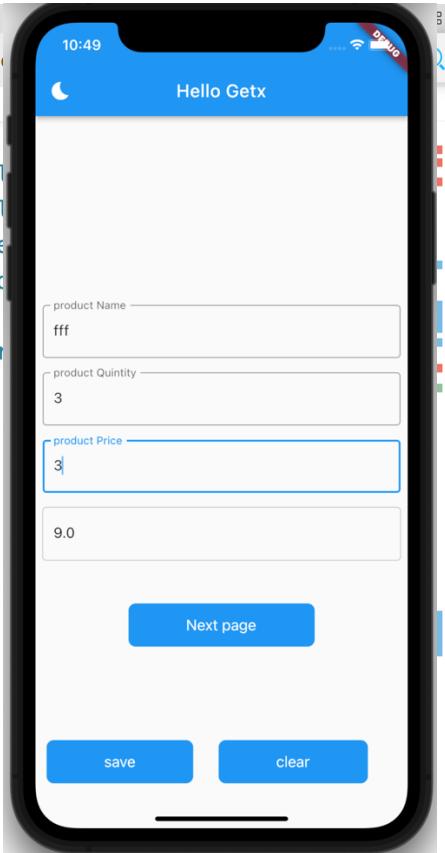
- ProductController

```
class ProductController extends GetxController {  
    var products = <Product>[].obs();  
    addProductCard(Product product) {  
        products.add(product);  
        print('object added');  
        update();  
    }  
  
    updateProductCard(Product pro) {}  
    removeProductCard(Product pro) {}  
}
```

- ProductModel

```
class Product {  
    late int code;  
    late String name;  
    late double price;  
    late int qty;  
    late double total;  
    Product(  
        {required this.code,  
         required this.name,  
         required this.price,  
         required this.qty,  
         required this.total,  
     });  
}
```

- Scaffold



```
main.dart — getx_statemangement
lib > main.dart > _MyHomePageState > build
34 class _MyHomePageState extends State<MyHomePage> {
35   TextEditingController nameController = TextEditingController();
36   TextEditingController priceController = TextEditingController();
37   TextEditingController qtyController = TextEditingController();
38   TextEditingController totalController = TextEditingController();
39   //-----
40   ProductController productController = Get.put(ProductController());
41
42   @override
43   Widget build(BuildContext context) {
44     return Scaffold(
45       appBar: AppBar(
46         leading: InkWell(
47           child: const Icon(Icons.bedtime),
48           onTap: () {},
49         ), // InkWell
50         title: const Text('Hello Getx'),
51       ), // AppBar
52       body: Center(
53         child: Column(
54           mainAxisAlignment: MainAxisAlignment.center,
55           children: [
56             Padding(
57               padding: const EdgeInsets.all(8.0),
58               child: TextFormField(
59                 controller: nameController,
60                 decoration: const InputDecoration(
61                   label: Text('product Name'), border: OutlineInputBorder(),
62                 ). // TextFormField

```

```

63   ), // Padding
64   Padding(
65     padding: const EdgeInsets.all(8.0),
66     child: TextFormField(
67       controller: qtyController,
68       decoration: const InputDecoration(
69         label: Text('product Quintity'),
70         border: OutlineInputBorder(), // InputDecoration
71     ), // TextFormField
72   ), // Padding
73   Padding(
74     padding: const EdgeInsets.all(8.0),
75     child: TextFormField(
76       onChanged: ((newValue) {
77         setState(() {
78           int qty = int.parse(qtyController.text);
79           double price = double.parse(newValue.toString());
80           totoalController.text = (price * qty).toString();
81         });
82       }),
83       controller: priceController,
84       decoration: const InputDecoration(
85         label: Text('product Price'), border: OutlineInputBorder(), //
86     ), // TextFormField
87   ), // Padding
88   Padding(
89     padding: const EdgeInsets.all(8.0),
90     child: TextFormField(
91       enabled: false,
92       controller: totoalController,
93       decoration: const InputDecoration(
94         hintText: 'total', border: OutlineInputBorder()), // InputDecoration
95     ), // TextFormField
96   ), // Padding
97   const SizedBox(
98     height: 40,
99   ), // SizedBox
100  CupertinoButton(
101    color: Theme.of(context).primaryColor,
102    child: const Text('Next page'),
103    onPressed: () => Get.to(() => ListProductScreen()), // CupertinoBu
104  ],
105  ), // Column
106 ), // Center

```

```

107     floatingActionButton: Row(
108       mainAxisAlignment: MainAxisAlignment.spaceEvenly,
109       children: [
110         CupertinoButton(
111           color: Theme.of(context).primaryColor,
112           child: const Text('save'),
113           onPressed: () => productController.addProductCard(Product(
114             code: Random().nextInt(200),
115             name: nameController.text,
116             price: double.parse(priceController.text),
117             qty: int.parse(qtyController.text),
118             total: double.parse(totoalController.text),
119           )), // Product // CupertinoButton
120         CupertinoButton(
121           color: Theme.of(context).primaryColor,
122           child: const Text('clear'),
123           onPressed: () {}), // CupertinoButton
124       ],
125     ), // Row
126   ); // Scaffold
127 }
128 }

```



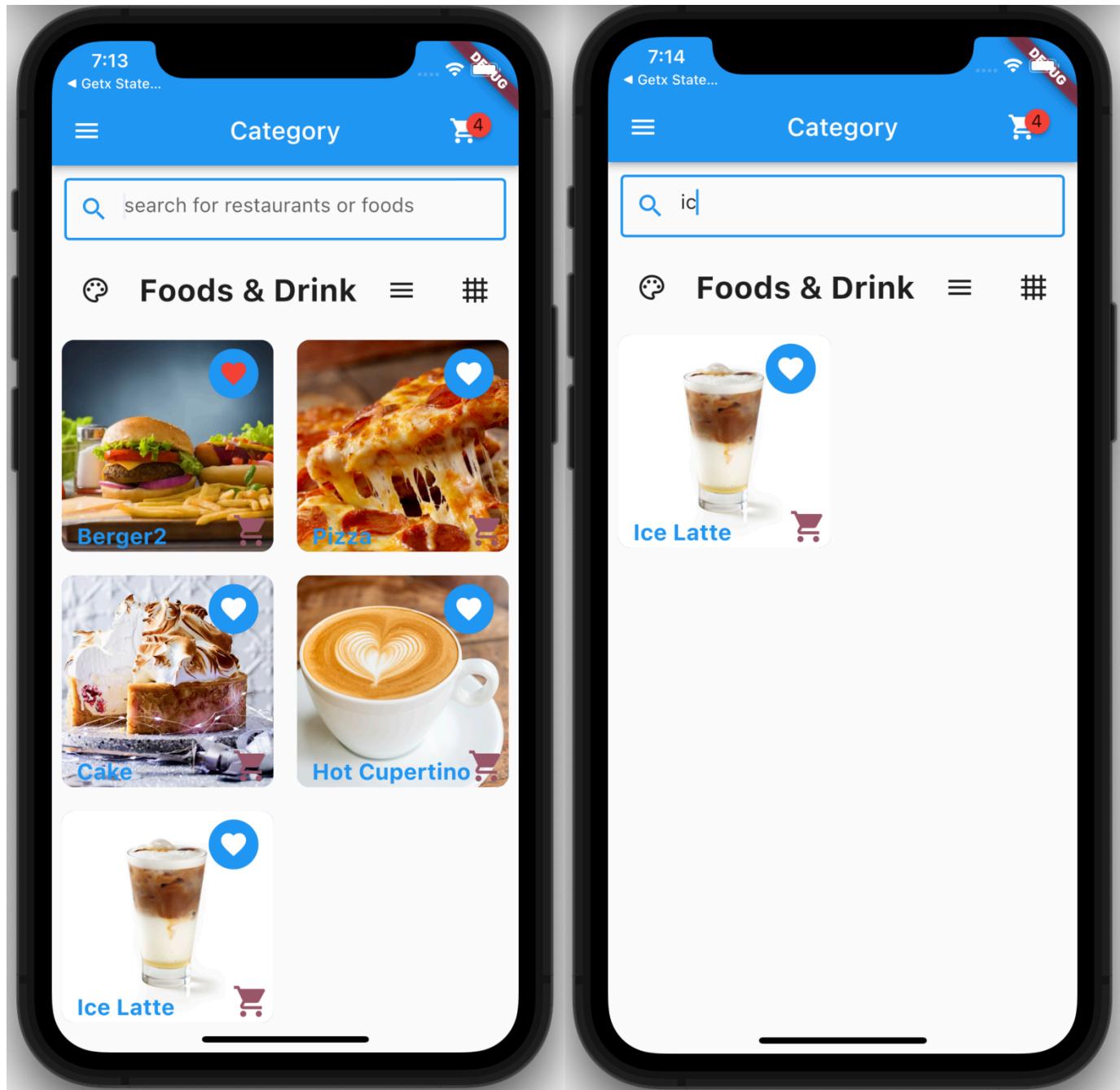
```

5  class ListProductScreen extends StatelessWidget {
6   ListProductScreen({Key? key}) : super(key: key);
7   final ProductController productController = Get.find();
8   @override
9   Widget build(BuildContext context) {
10    return Scaffold(
11      appBar: AppBar(
12        title: const Text('List Product'),
13      ), // AppBar
14      body: ListView.builder(
15        itemCount: productController.products.length,
16        itemBuilder: (context, index) {
17          var pro = productController.products[index];
18          return ListTile(
19            leading: CircleAvatar(
20              child: Text(pro.code.toString()),
21            ), // CircleAvatar
22            title: Text(
23              pro.name,
24              style: const TextStyle(fontSize: 25),
25            ), // Text
26            subtitle: Text(pro.total.toString()),
27          ); // ListTile
28        },
29      ), // ListView.builder
30    ); // Scaffold
31  }
32 }
33

```



Apply with GetX



Ice Latte

price : 3.0\$

<https://www.lacantina.ae/wp-content/upl>

- 1 +

Buy now

Add to Card

7:14 Getx State... DEBUG

7:15 Getx State... DEBUG

All Shopping Card

Item	Quantity	Price
food	4	3.5\$
Berger1	2	7.5\$
Cake	1	10.0\$
Hot Cupertino	1	3.0\$



Thank you..!!