



Mobile App Development Class 02, Series 04

Mobile App

HahuJobs

CLASS 04

HahuJobs

01 Assets

02 Routing

03 Slivers

04 Custom ScrollView



Assets in Flutter

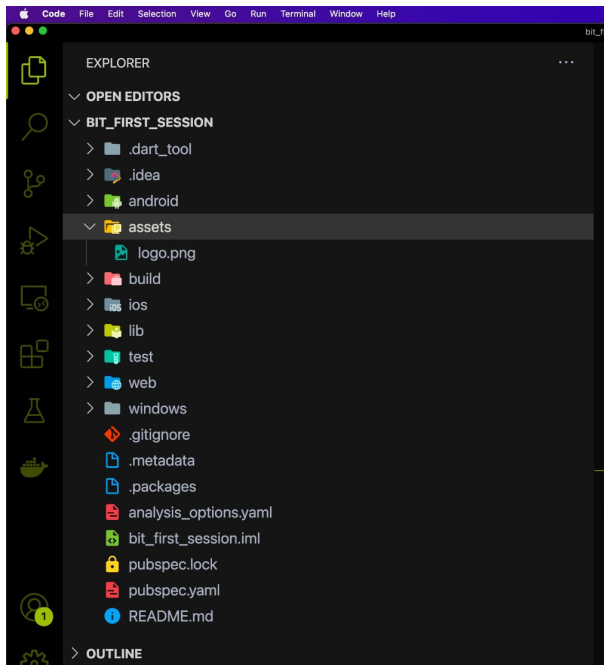
What is it?

An asset is a file that is bundled and deployed with your app, and is accessible at runtime. Common types of assets include static data (for example, JSON files), configuration files, icons, and images (JPEG, WebP, GIF, animated WebP/GIF, PNG, BMP, and WBMP).

Assets in Flutter

How to define assets.

Assets are defined in `pubspec.yaml` file that is found in the root directory of a flutter project.



```
bit_first_session - pubspec.yaml

1  uses-material-design: true
2
3  assets:
4    - assets/logo.png
5    # your rest of assets go here.
6    # the assets folder can be named in your choice.
7
8
```

Assets in Flutter

Reading assets.

```
1 Scaffold(  
2   appBar: AppBar(  
3     automaticallyImplyLeading: false,  
4     title: Text("Logo Page"),  
5   ),  
6   body: Center(  
7     // reading Image assets  
8     child: Image.asset("assets/logo.png")),  
9 )
```

```
1 Future _readJson() async {  
2   var _data = await rootBundle.load("assets/data.json");  
3   // process your data  
4 }  
5
```

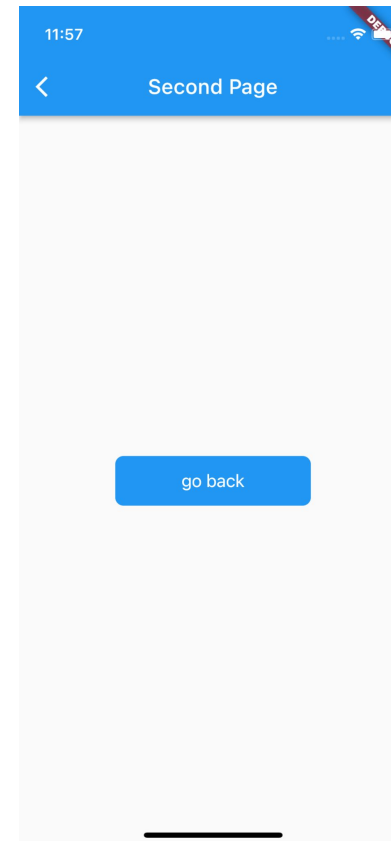
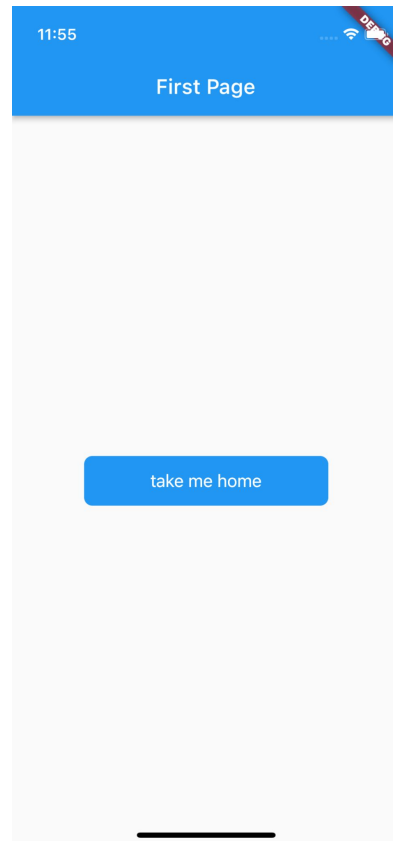


Routing & Navigation in Flutter

is the concept of switching between pages or screens without necessarily changing the initial page. Flutter has an imperative routing mechanism, the Navigator widget, and a more idiomatic declarative routing mechanism (which is similar to build methods as used with widgets), the Router widget.

Routing & Navigation in Flutter

Example - Suppose you have to pages named `FirstPage()` & `SecondPage()`



Routing & Navigation in Flutter

Navigation - Example

```
1 class FirstPage extends StatelessWidget {
2   const FirstPage({Key? key}) : super(key: key);
3
4   @override
5   Widget build(BuildContext context) {
6     return Scaffold(
7       appBar: AppBar(
8         automaticallyImplyLeading: false,
9         title: Text("First Page"),
10      ),
11      body: Center(
12        child: CupertinoButton(
13          color: Colors.blue,
14          child: Text("take me home"),
15          onPressed: () {
16            // navigator in action
17            Navigator.of(context).push(
18              new MaterialPageRoute(builder: (context) => SecondPage()));
19          },
20        ),
21      );
22    }
23  }
```

```
1
2 class SecondPage extends StatelessWidget {
3   const SecondPage({Key? key}) : super(key: key);
4
5   @override
6   Widget build(BuildContext context) {
7     return Scaffold(
8       appBar: AppBar(
9         automaticallyImplyLeading: false,
10        title: Text("Second Page"),
11      ),
12      body: Center(
13        child: CupertinoButton(
14          color: Colors.blue,
15          child: Text("go back"),
16          onPressed: () {
17            // pop action is used to remove the current page from the page stack
18            Navigator.of(context).pop();
19          },
20        ),
21      );
22    }
23  }
24 }
```




Sliver

what is it ?

A sliver is a portion of a scrollable area that you can define to behave in a special way. You can use slivers to achieve custom scrolling effects, such as elastic scrolling.



Custom ScrollView

what is it ?

is a scroll view widget that allows us to create various scrolling effects such as grids, lists, and expanding headers. It has a **sliver property** where we can pass a list of widgets that include SliverAppBar, SliverToBoxAdapter, SliverList, and SliverGrid, etc.

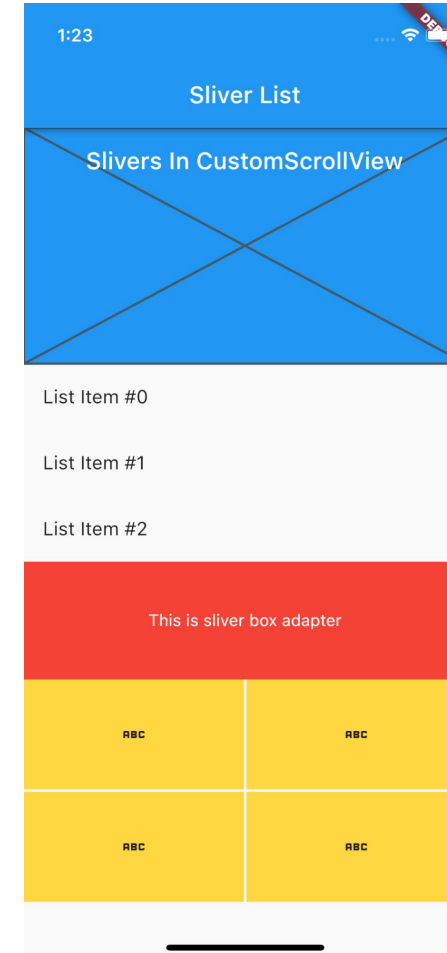
Custom ScrollView

Custom Scrollview - example

```

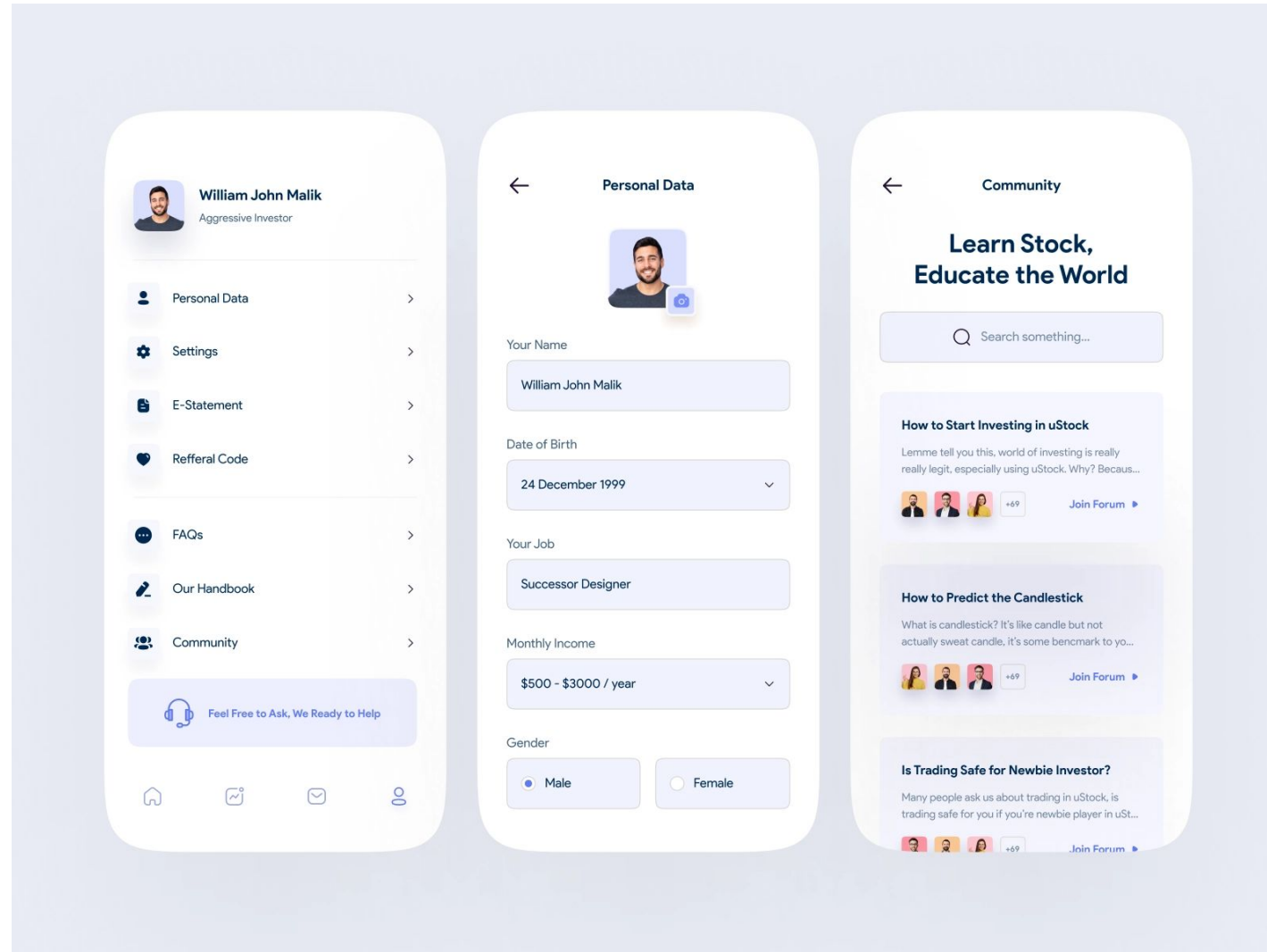
1 CustomScrollView(
2   slivers: [
3     const SliverAppBar(
4       title: Text("Slivers In CustomScrollView"),
5       floating: true,
6       flexibleSpace: Placeholder(),
7       expandedHeight: 200,
8     ),
9     SliverList(
10      delegate: SliverChildBuilderDelegate(
11        (context, index) => ListTile(title: Text('List Item #$index')),
12        childCount: 3,
13      ),
14    ),
15    SliverToBoxAdapter(
16      child: Container(
17        height: 100,
18        color: Colors.red,
19        child: Center(
20          child: Text(
21            "This is sliver box adapter",
22            style: TextStyle(color: Colors.white),
23          ),
24        ),
25      ),
26    ),
27    SliverGrid.count(
28      crossAxisCount: 2,
29      childAspectRatio: 2,
30      crossAxisSpacing: 2,
31      mainAxisSpacing: 2,
32      children: List.generate(
33        4,
34        (index) => Container(
35          color: Colors.amberAccent, child: Icon(Icons.abc)),
36      ),
37    ),
38  ],

```



PRACTICE

Implement the design attached below with your instructor .





Reading Assignment

- Forms
- Bottom Navs
- Dependency Injection
- Bloc, riverpod and get_it
- replicate this ui [design](#) - only the light mode.

Thank you!

HahuJobs

ለህገር ልጅ ለህገር ልጅ !

Tesfahun Kebede

tesfahunkebede336@gmail.com

