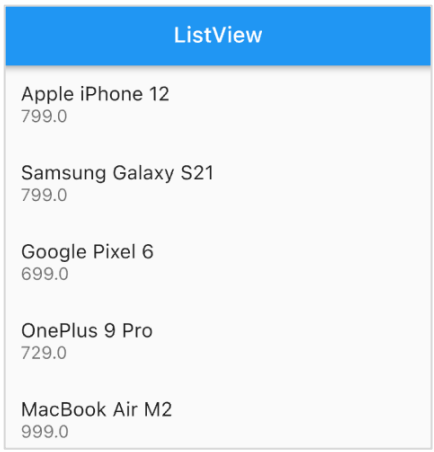
CROSS-PLATFORM MOBILE APP DEVELOPMENT

(503107)

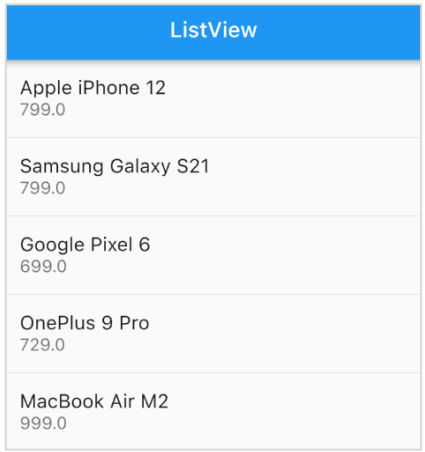
LAB 3

EXERCISE 1

Given a list of products with each product including its name and price in the **products.dart** file. Let's display this list on a ListView widget in **3 different ways** corresponding to 3 different constructors of ListView widget: ListView(), ListView.buider() và ListView.separated().



ListView() and **ListView.builder()** give identical UI except that ListView.builder() is more performant

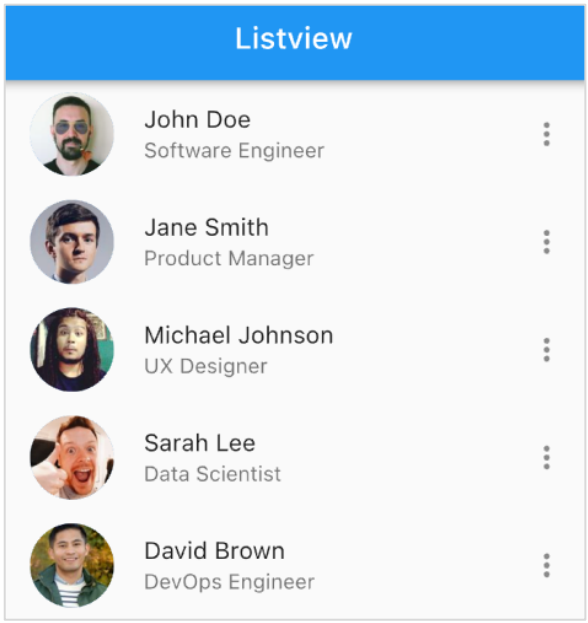


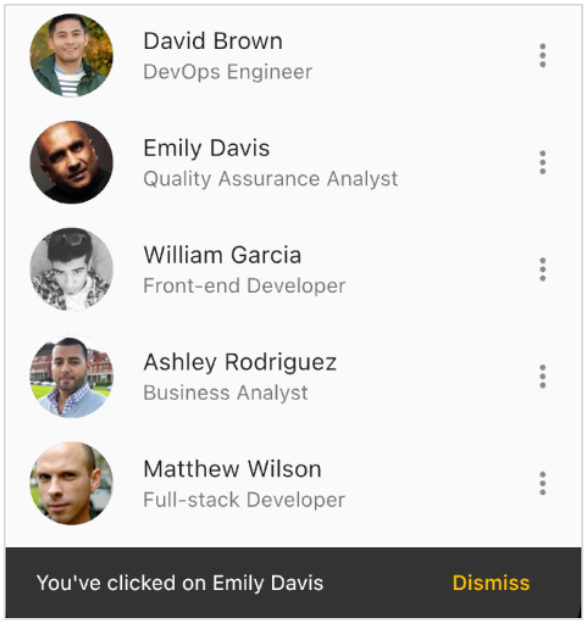
ListView.separated() allow you to put a **Divider** widget between the **ListTile** widgets

EXERCISE 2

Use the data in the given **users.dart** file to build a Listview widget that displays a list of users as shown below. The listview items must be generated using the map() function from the list of users to the list of **ListTile** widget. User avatars are downloaded from an arbitrary public api service, you can use the following: https://randomuser.me/api/portraits/men/1.jpg

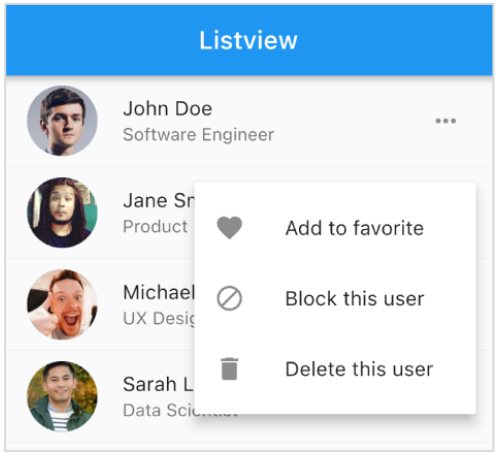
When clicking on any user, display a message with SnackBar to display that person's name.





EXERCISE 3

Continuing with the previous exercise, let's modify the Listview widget, using the Listview.separated() constructor to create a Listview with separators between each item. Assign a **PopupMenuButton** widget to the trailing property of ListTile to display a popup menu every time the icon ... (3 dots) of each item is clicked.



EXERCISE 4

Use the product data in the products.dart file as before to create an interface using the GridView as shown below. Each GridView item uses the Card widget in conjunction with InkWell to display a material ripple effect every time they are tapped. Similar to ListView exercise, You must create GridView in two ways corresponding to two constructor methods: **GridView()** and **GridView.builder()**.

Fig 1. GridView in normal state

Fig 2. GridView with some selected item

Next, let's adjust the source code to allow gridview items to be selectable. When pressing and holding (long press) on items, they will be switched to the selected state as in figure 2. If do so again, they will be returned to their original state.

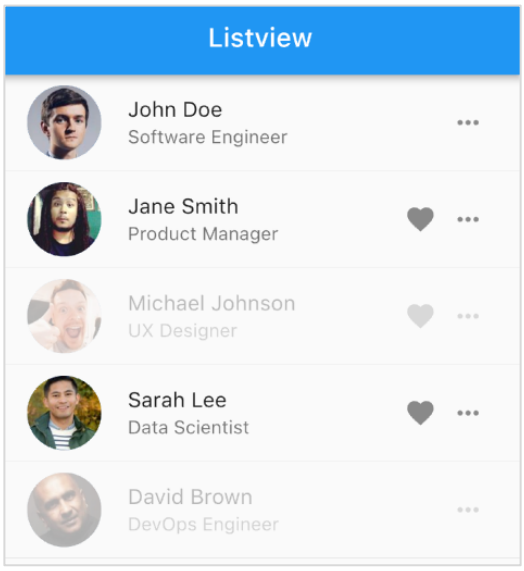
EXERCISE 4

Keep working on the previous exercise, this time you need to focus on handling the onTap() event of the items in the popup menu. The specific requirements are as follows:

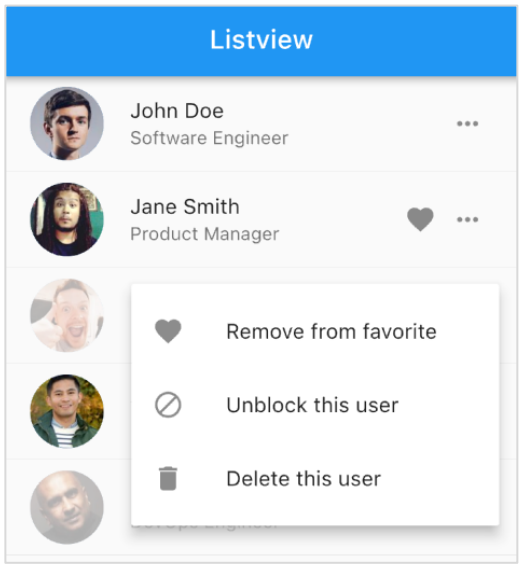
- When "Add to favorite" is selected, the respective user will display a heart-shaped icon on the right. Later, if the popup menu is displayed again, there will be a "Remove from

favorite" option instead of the "Add to favorite" option. If "Remove from favorite" is selected, the heart icon will be removed from the user.

- When "Block this user" is selected, the corresponding line will be greyed out (using the Opacity widget). Similarly, the popup menu will later show the "Unblock this user" option for blocked users. If "Unblock this user" is selected, the corresponding line will light up again (opacity = 1).
- When "Delete this user" is selected, the corresponding user will be removed from the list.



Jane Smith is in the favorite list and **David Brown** is blocked



Popup menu showing options to unblock a user from blocked status and remove from favorite list