

Topics (Slide 03: Typescript & Angular, Slide 04: Navigation)

Duration: 1 Hour 30 mins

UI: Input, label, button, items, list, icons, alerts, select

In this tutorial, you will create a multipage app to display laptop information stored in the array as a list. Create the following beside the home page:

```
ionic generate page Brand
```

```
ionic generate page Detail
```

```
ionic generate page Settings
```

```
ionic generate service Data
```

In the generated Service file, define a new data type (laptop) used in tutorial 2 as follows. Add new field 'Image' to the type:

```
export interface laptop {  
    Brand:    String,  
    CPU:      String,  
    GPU:      String,  
    RAM:      Number,  
    Weight:   Number,  
    Screen:   Number,  
    Storage:  boolean,  
    OS:       boolean,  
    Image:     String,  
    ManuDate: Date  
}
```

Create an array called 'List' of the type laptop. Insert few records in the array List (at least three).

In the Home page, add five buttons as shown below.

Clicking on the buttons navigate to the specified page

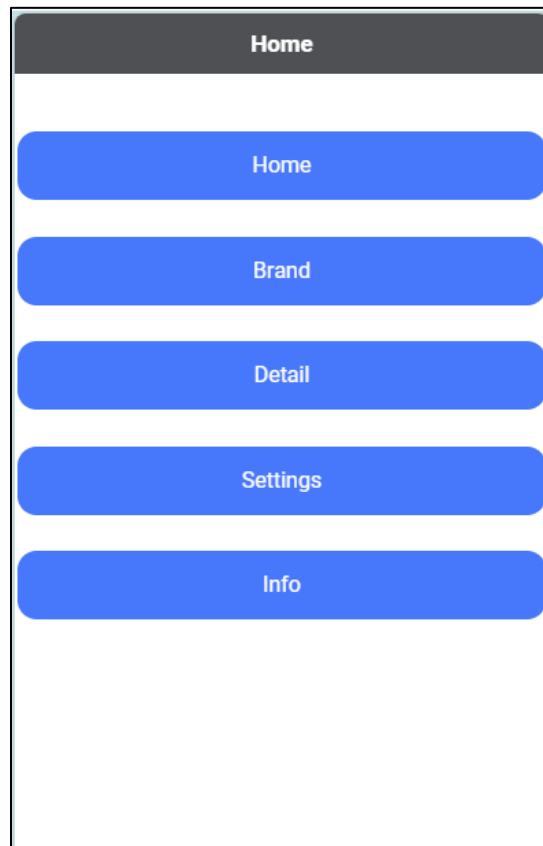
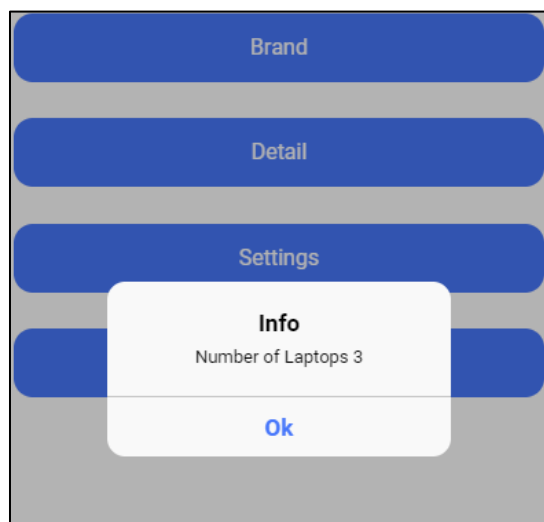


Figure 1: Home Page View

Clicking on the button info, displays an alert message showing the number of records in the array List.



In the Brand page:

1. Display all records in the array List as a list of **Cards**.
2. In each card display: **Brand** (as header), **Image**, **CPU**, **RAM** values as shown.
3. In addition, in each card display a button titled '**View Details**'. Clicking the button will navigate to the page '**Detail**' and will display the same data but with more fields: Brand, Image, CPU, **GPU**, RAM and **Screen** size as shown below.
4. Display a button '**Close**' to navigate back to the 'Brand' page.

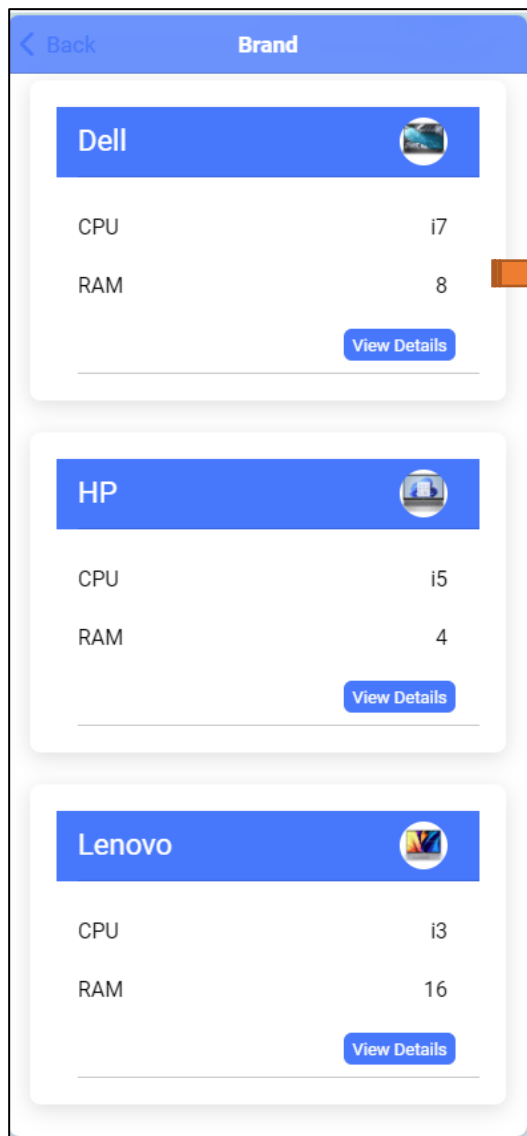


Figure 2: Brand Page View



Figure 3: Detail Page View

In the Settings page:

Display a Select component, which allows the user to select from different colors.

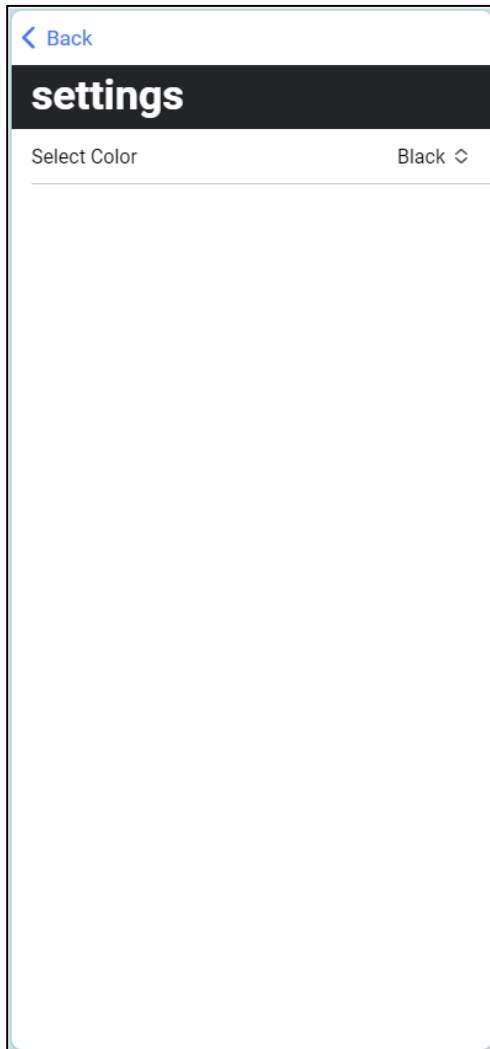


Figure 4: Brand Page View

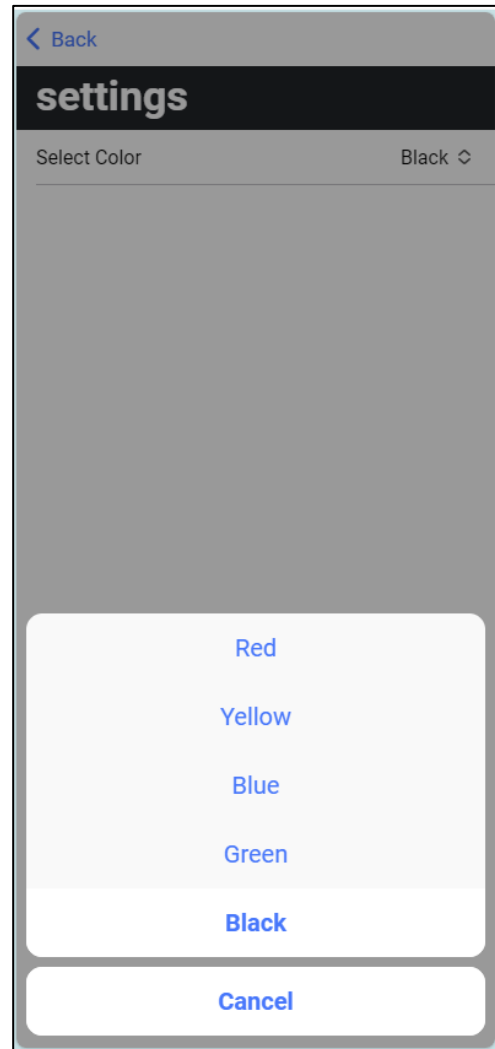


Figure 5: Brand Page View

The selected color will change automatically the toolbar color. In the figure, you will notice the selected color is black and the toolbar color is set to **black**. The toolbar color will change for all pages and not only Settings page.

Hint: Define a string variable in the Data service (for example, `barColor`). Tie the **Select component** using `ngModel` with the string variable `barColor`. Use the string variable `barColor` to change the toolbar color as follows:

```
<ion-toolbar color="{{DataSrv.barColor}}">  
OR <ion-toolbar [color]="DataSrv.barColor">
```

In the **Brand**, **Detail** and **Settings** pages, display navigation buttons at the top of the page to allow the user for quick navigation to other pages. Show also the back button in the navigation bar (you may use `<ion-back-button slot="start"></ion-back-button>`). See figures below:

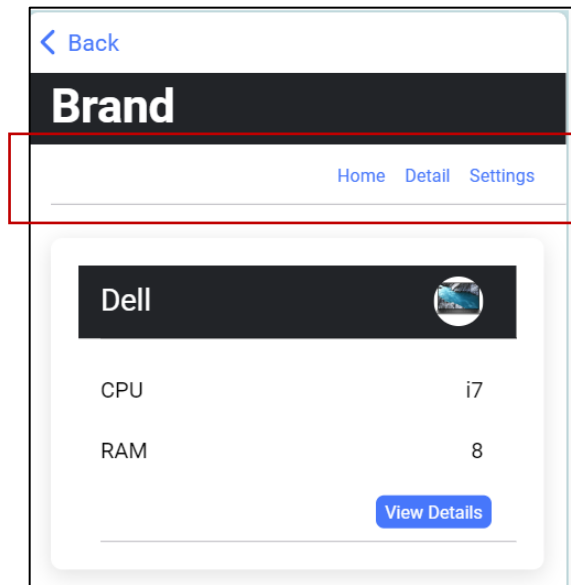


Figure 6: Brand Page View

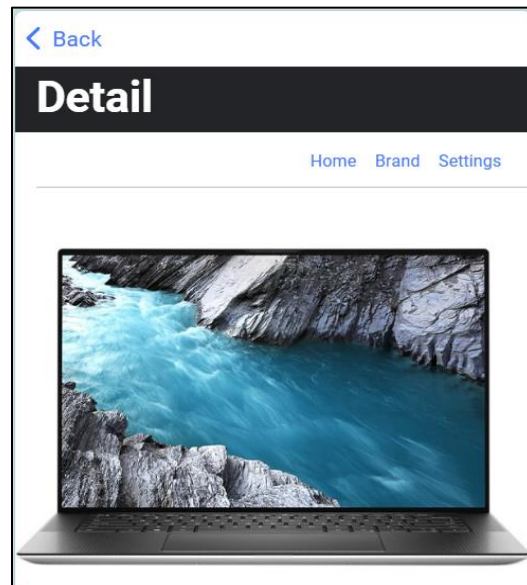


Figure 7: Brand Page View

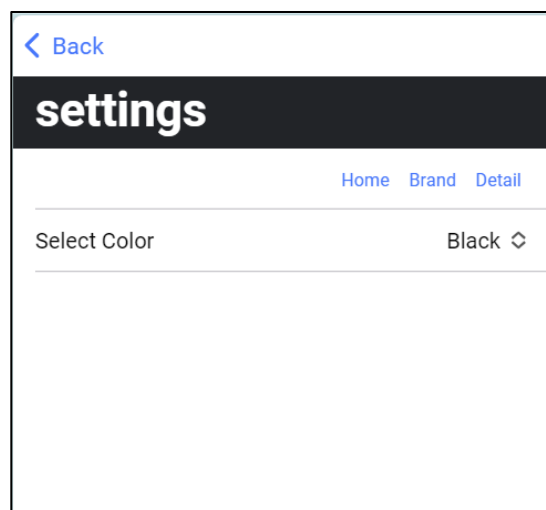


Figure 8: Settings Page View