

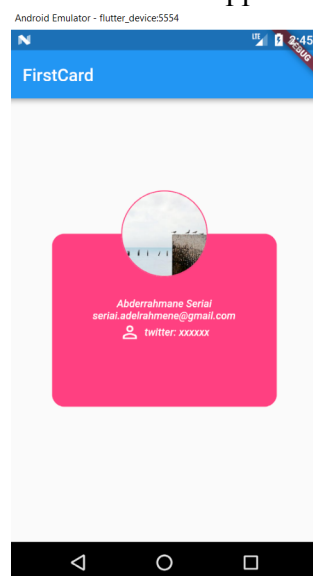
HAI912I : Programmation Mobile Avancée et IoT Flutter

Le but de ces exercices est de vous familiariser avec la création d'interfaces graphiques en utilisant Flutter : les widget stateless, et stateful, ainsi que la gestion des états en utilisant les Bloc (Business Logic Component)

Application 1

Avec l'application « FirstCard », vous allez être amenés à créer une carte de profil d'une personne. Vous allez ainsi devoir créer un widget qui regroupe 2 parties : une partie qui contient une image de profil, et une autre partie qui contient les informations concernant le profil notamment : le nom, le prénom, l'adresse email, et le compte d'un réseau social.

La figure suivante montre à quoi doit ressembler l'application que vous allez créer :



Pour créer cette application vous allez avoir besoin d'un widget Stateless : une classe qui hérite de « StatelessWidget ».

```
class MyApp extends StatelessWidget {  
  const MyApp({Key? key}) : super(key: key);  
  
  // This widget is the root of your application  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Flutter Demo',  
      theme: ThemeData(...), // ThemeData  
      home: const MyHomePage(),  
    ); // MaterialApp  
  }  
}
```

Afin d'avoir la barre d'application vous pouvez utiliser le widget « Scaffold » avec son attribut AppBar.

Voici quelques widgets dont vous aurez besoin afin de développer cette application :

Container : Un widget pratique qui combine des widgets courants de peinture, de positionnement et de dimensionnement.

```
return Container(  
  width: 300,  
  height: 200,
```

Afin de décorer un widget, il existe le widget `BoxDecoration()` qui peut être utilisé.

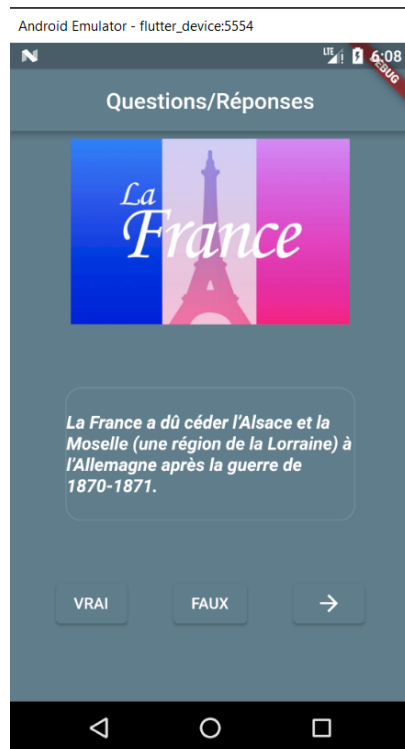
Stack : Un widget utile si vous souhaitez superposer plusieurs enfants de manière simple, par exemple en ayant du texte et une image, superposés avec un dégradé et un bouton attaché au bas.

```
@override  
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(  
      title: Text("FirstCard"),  
      centerTitle: false,  
    ), // AppBar  
    body: Container(  
      alignment: Alignment.center,  
      child: Stack(...), // Stack  
    ), // Container  
  ); // Scaffold  
}  
  
Container _getCard() {...}  
  
Container _getAvatar() {...}
```

Application 2

Avec l'application « Questions/Réponses », vous allez être amenés à créer un Quiz sur une thématique qui vous intéresse. Vous allez ainsi devoir créer un widget de type « StatefulWidget » qui regroupe une image, une question ainsi qu'un ensemble de boutons.

La figure suivante montre à quoi doit ressembler l'application que vous allez créer :



La particularité de cette application par rapport à la première application « FirstCard » est l'utilisation du « State ».

```
class MyQuizPage extends StatefulWidget {
  const MyQuizPage({Key? key, required this.title}) : super(key: key);

  final String title;

  @override
  State<MyQuizPage> createState() => _MyQuizPageState();
}
```

Une classe « Question » qui le modèle d'une question doit être créée comme suit :

```
class Question {
  String questionText;
  bool isCorrect;

  Question({required this.questionText, required this.isCorrect});
}
```

Voici quelques widgets dont vous aurez besoin afin de développer cette application :

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(...), // AppBar
    backgroundColor: Colors.blueGrey,
    body: Container(...), // Container
  ); // Scaffold
}

_checkAnswer(bool userChoice, BuildContext context) {...}

_nextQuestion() {...}
```
