



Université de Montpellier

TP1 : Création d'interfaces graphiques en Flutter

Réalisé par :

EZZEROUG EZZRAIMI Elhadi & BERBER Aissa

Année Universitaire : 2024-2025

Exercice 1 - Profile Card

Objectif :

Créer une application Flutter qui représente le profil d'une personne. Cette application contient un widget regroupant deux parties : 1. Un avatar (photo de profil). 2. Les informations relatives au profil (e.g. nom, prénom, email, comptes sur les réseaux sociaux, etc.).

Structure de l'application

L'application est basée sur un widget stateless (sans état) et suit le design Material Design. Voici les éléments principaux :

1. Point d'entrée de l'application

Le point d'entrée est défini avec la méthode `main` qui lance l'application via le widget racine `MyApp`.

2. Widget de la page d'accueil : `ProfileHomePage`

Le widget `ProfileHomePage` est un `StatelessWidget`. Sa structure comprend :

- **AppBar** : Une barre d'application en haut avec un titre.
- **Body** : Un conteneur centralisé qui contient une `Stack` permettant de superposer plusieurs widgets, comme l'avatar, les informations et des décorations.

Description des widgets utilisés

- `MaterialApp` : Application utilisant Material Design.
- `Scaffold` : Conteneur principal avec une barre d'application et un corps (body).
- `AppBar` : Barre d'application affichant le titre "Profile Card".
- `Container` : Utilisé pour regrouper et styler des éléments. Il est centralisé via `alignment: Alignment.center`.
- `Stack` : Permet de superposer plusieurs widgets comme l'avatar et la carte d'informations.
- `BoxDecoration` : Utilisé pour styler les conteneurs, comme ajouter des ombres, des bordures ou des dégradés.
- `Text` : Affiche les informations textuelles du profil.

Conclusion

Ce TP permet de se familiariser avec les concepts de base de Flutter, notamment :

- La création de widgets stateless.
- L'utilisation des outils de positionnement et décoration (`Container`, `BoxDecoration`, `Stack`).
- La personnalisation d'une application Material Design.

Ce projet pose les bases pour développer des applications plus complexes en Flutter.

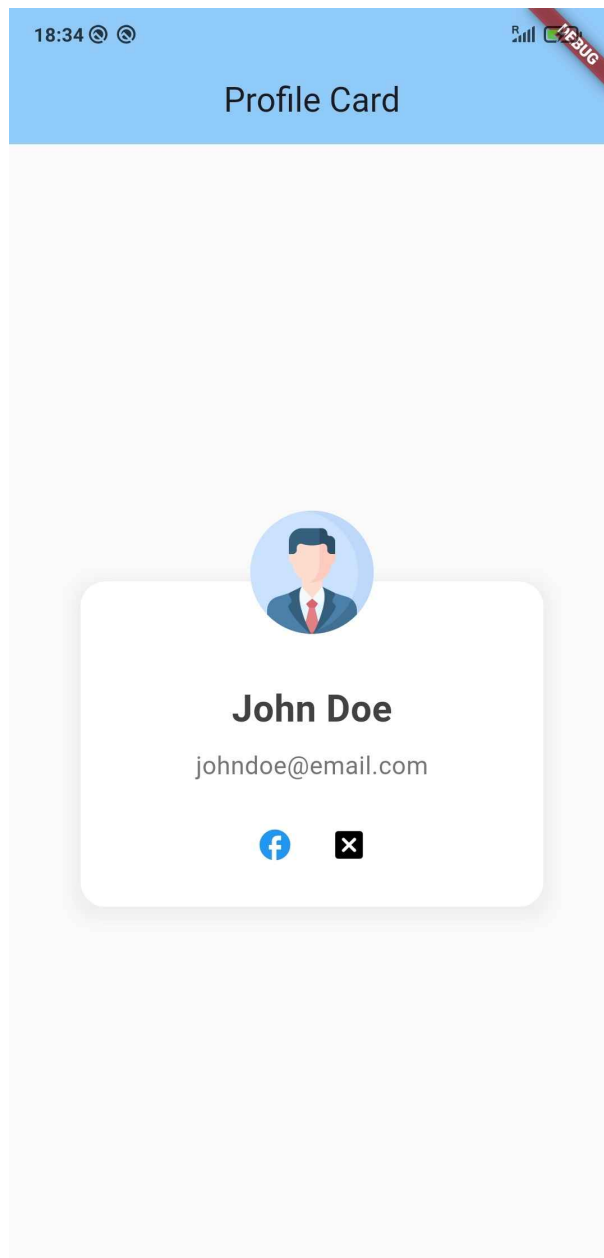


FIGURE 1 – Profile Card

Exercice 2 : Quiz

Objectif

Le but de cet exercice est de créer une application Flutter permettant de répondre à des questions d'un quiz sur une thématique donnée, en l'occurrence, des questions liées au cinéma.

Structure de l'application

L'application est un widget `Stateful` permettant de gérer l'état de l'application, contrairement au premier exercice où le widget était `Stateless`. La page principale du quiz est représentée par le widget `MyQuizPage`, qui hérite de la classe `StatefulWidget`.

1. Modélisation des questions

Les questions sont modélisées à l'aide d'une classe `Question`, qui contient trois attributs :

- `questionText` : Le texte de la question.
- `isCorrect` : Un booléen indiquant si la réponse est correcte.
- `image` : L'image associée à la question.

Exemple d'une question :

```
Question(questionText: 'Leonardo DiCaprio a remporté son premier Oscar  
pour "Inception".', isCorrect: false, image: '1.jpg')
```

2. Gestion de l'état

La classe `State<MyQuizPage>` gère l'état de l'application et déclare les variables nécessaires, telles que `questionId` (pour suivre la question courante) et `questions` (qui contient la liste des questions).

La méthode `updateQuestions()` permet de passer à la question suivante, et si nous arrivons à la dernière question, le quiz recommence.

3. Affichage de la question

Pour chaque question, l'application affiche :

- Une image associée à la question (avec `Image.asset`).
- Le texte de la question.
- Deux boutons (`Vrai` et `Faux`) permettant de répondre à la question.

Lorsque l'utilisateur clique sur l'un des boutons, la méthode `setState()` est appelée pour mettre à jour l'état de l'application et afficher un `SnackBar` avec un message indiquant si la réponse est correcte ou incorrecte.

4. Comportement des boutons

Les boutons `"Vrai"` et `"Faux"` permettent de vérifier la réponse de l'utilisateur et d'afficher un message de feedback :

- Si la réponse est correcte, un message `"C'est vrai!"` est affiché.
- Si la réponse est incorrecte, un message `"Non, c'est faux"` apparaît.

5. Navigation vers la question suivante

Un bouton supplémentaire est présent pour passer à la question suivante, utilisant un `IconButton` avec l'icône d'une flèche vers la droite. Ce bouton permet de faire défiler les questions.

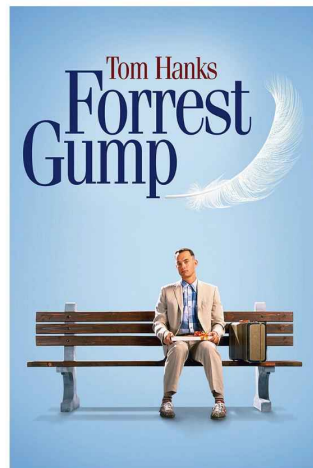
Conclusion

Ce TP permet d'acquérir une bonne compréhension de l'utilisation des widgets `Stateful` et `Stateless` en Flutter, ainsi que des interactions avec les utilisateurs, comme la gestion des réponses à un quiz et l'affichage de feedback avec des `SnackBar`. Le projet présente aussi un exemple de gestion de la navigation entre différentes questions et de l'affichage d'images et de texte.

0:12 @ @



Quiz ciné



Le personnage de "Forrest Gump" a été interprété par Tom Hanks.

Vrai

Faux



FIGURE 2 – Quiz