

UNIVERSITÉ DE MONTPELLIER

M2 GÉNIE LOGICIEL

TD1 : Programmation par contraintes

HAI912I : DÉVELOPPEMENT MOBILE AVANCÉ, IoT ET EMBARQUÉ

Étudiants :

Denis BEAUGET

Année : 2021 – 2022

Encadrant :

Serial ABDELHAK-DJAMEL



UNIVERSITÉ
DE MONTPELLIER



Table des matières

1	Lien vers le GitHub :	2
2	Application 1 :	2
2.1	Objectifs :	2
2.2	Méthodes utilisés :	2
3	Application 2 :	3
3.1	Objectifs :	3
3.2	Méthodes utilisés :	3

1 Lien vers le GitHub :

https://github.com/Beauget/HAI913I_TP1

2 Application 1 :

Précision : N'ayant pas eu le temps de configurer mon téléphone pour l'utilisation de flutter (en raison d'un ordinateur personnel trop peu puissant) j'ai réalisé l'intégralité de mon travail sur le moteur de rendu de chrome intégré à flutter, les images sont donc issu de celui-ci et les tailles correspondaient donc chez moi mais peut-être pas forcément sur un téléphone.

Ce rendu est très court, il a pour but de faciliter la correction et d'expliqué brièvement les méthodes utilisées pour mes applications.

2.1 Objectifs :

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.

2.2 Méthodes utilisés :

Pour cette application, j'ai suivi les conseils indiqués en utilisant le widget `Stack()`¹ et j'ai simplement 'empilé' 2 containers que j'ai positionné avec le widget `Positioned()`² et finalement l'attribut `decoration` : pour l'ajout de l'image.

1. Stack : <https://api.flutter.dev/flutter/widgets/Stack-class.html>

2. Positioned : <https://api.flutter.dev/flutter/widgets/Positioned-class.html>

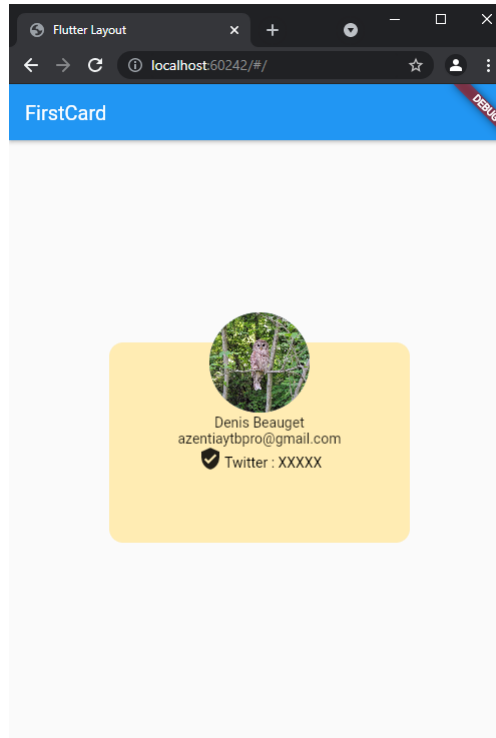


FIGURE 1 – Rendu de l'application

3 Application 2 :

3.1 Objectifs :

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.

3.2 Méthodes utilisés :

Pour cette application, j'ai eu plus de difficultés que la précédente, j'ai donc créé une classe question qui m'a permis de mettre au point un 'modèle' pour mes questions de ce type :

Question(questionText, isCorrect, imageURL)

Ensuite, j'ai utilisé le widget Expanded()³ (je pense que ce n'était pas le plus adapté mais sur mon rendu chrome c'était plus clair) qui m'a permis de créer mes boutons et d'ajouter ma question avec les méthodes issues de ma classe.

3. Expanded : <https://api.flutter.dev/flutter/widgets/Expanded-class.html>

Pour l'ajout des images, j'ai utilisé un package flutter⁴ pour n'avoir qu'à mettre le lien web dans la création de mes questions.

Pour la vérification des questions et la gestion du bouton 'Suivant' j'ai créé de 2 méthodes :

- CheckAnswer(bool userPickedAnswer)
- CheckAnswerNewt()

La première vérifie si on n'est pas à la fin du Quizz et renvoie une Alert() issu du package rflutter_alert⁵ avec le score et relance le Quizz. Sinon elle vérifie si la réponse à la question est bonne et rajoute 1 au score final.

La seconde vérifie également si le quizz est terminé et sinon passe à la question suivante.

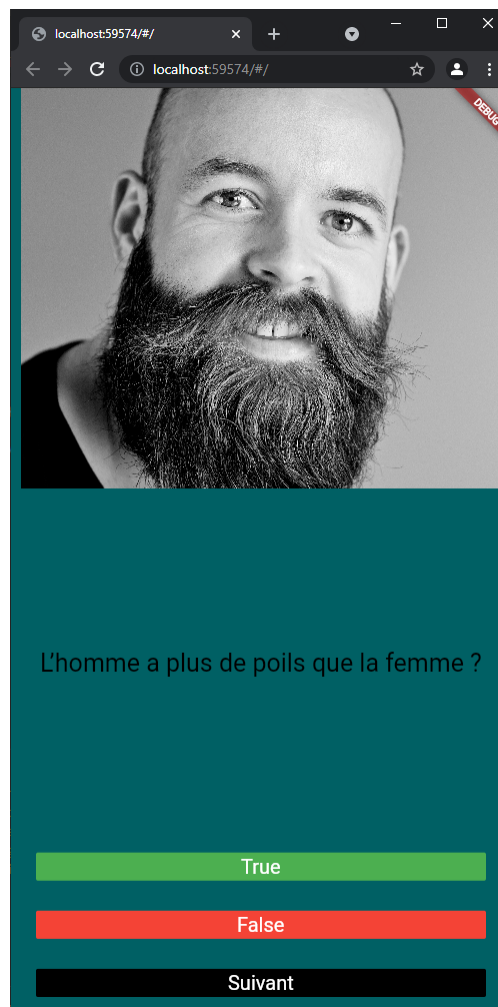


FIGURE 2 – Rendu de l'application

4. Image : https://pub.dev/packages/cached_network_image

5. Alert : https://pub.dev/packages/rflutter_alert

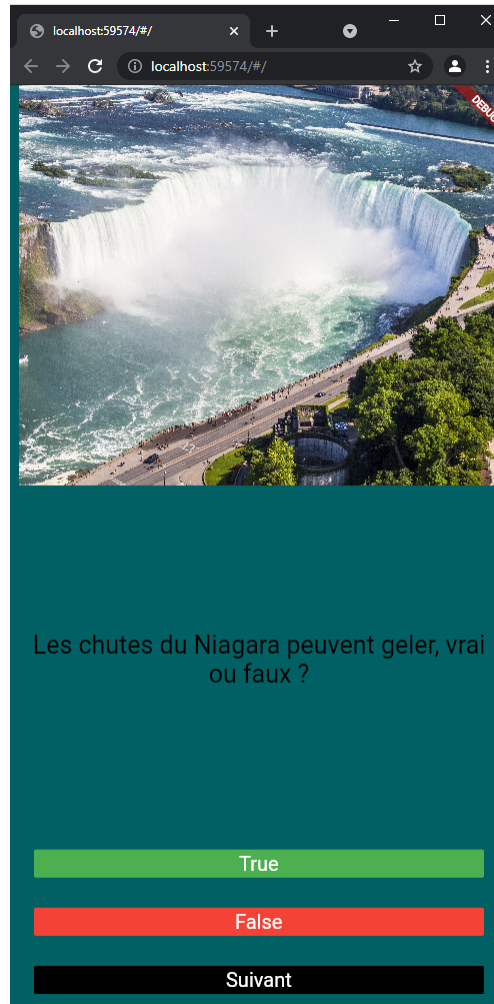


FIGURE 3 – Rendu du changement d'image

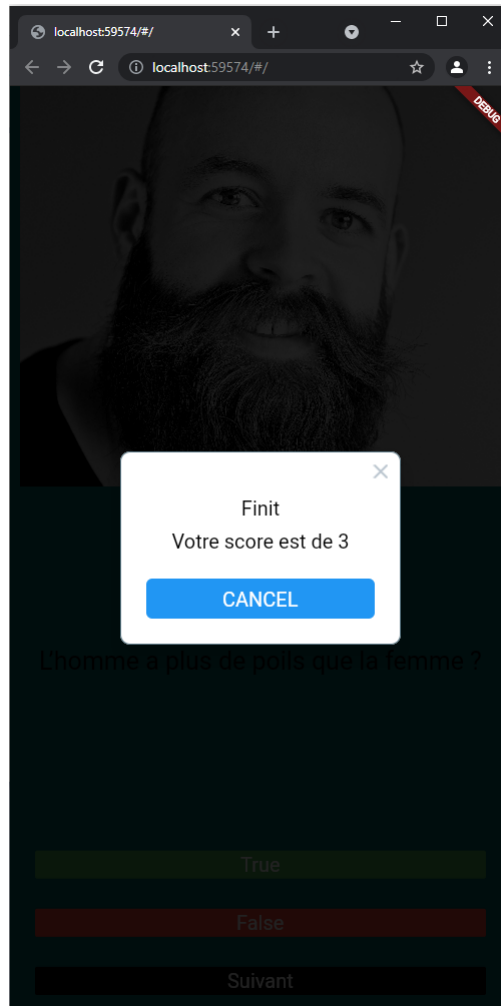


FIGURE 4 – Apperçu du score