

Examen 1 (30%)

Examen de Programmation Flutter : Création de Formulaire de base

Durée : 3h40

Notes : à remettre votre fichier main.dart et tous les autres fichier dart que vous avez créé dans un fichier zip.

Partie 1 : Création de l'interface utilisateur (60 points)

1. Créez une application Flutter avec une interface utilisateur qui contient les éléments suivants
 - Trois champs de texte pour saisir le nom, l'adresse e-mail et le numéro de téléphone.
 - Un bouton "Enregistrer" pour soumettre les données du formulaire.
 - Deux boutons radio pour sélectionner le genre : "Masculin" et "Féminin".
2. Assurez-vous que les éléments sont correctement disposés et esthétiquement agréables.
 - Utiliser les containers de type Column et autre pour disposer les éléments.
3. Changer les couleurs de l'application pour que la couleur principale soit rouge (#F40000) et le texte soit blanc.
4. Donner le titre « Création d'utilisateur » à l'application dans le appBar.
5. Ajouter une action dans le appBar pour avoir un icône « Utilisateur », il n'est pas nécessaire d'implémenter le code onPressed du bouton.
6. Créer un « Drawer » qui contient deux éléments en plus du nom de l'application : « Accueil » et « Quitter ».
7. Assurez-vous que votre application MaterialApp est à l'intérieur un Scaffold (afin de pouvoir avoir le appBar, le drawer et la possibilité de faire des SnackBar).

Partie 2 : Logique de formulaire (40 points)

1. Lorsque l'utilisateur appuie sur le bouton "Enregistrer", assurez-vous que les données sont valides :
 - Aucun champ de texte ne doit être vide.
 - Bonus 1: Ajouter la validation pour s'assurer que les noms sont au moins de 3 caractères, le courriel est correctement formaté et le numéro de téléphone contient exactement 10 chiffres.
2. Si les données sont valides, affichez un message de confirmation avec les informations saisies par l'utilisateur.
 - Cette information peut être affiché dans un TextField.
 - Bonus 2: Faire afficher le message dans une SnackBar au lieu d'un TextField.

3. Si les données ne sont pas valides, affichez un message d'erreur approprié près des champs concernés.
 - Afficher l'information dans un TextField.
 - Bonus 3: Faire afficher le message dans une SnackBar au lieu d'un TextField.
4. Assurez-vous que les boutons radio fonctionnent correctement et enregistrent le choix de l'utilisateur.
 - Bonus 4: Lorsque vous avez terminé la validation de votre utilisateur, sauvegarder les informations dans une classe conteneur « Utilisateur » qui contient vos données (contient les membres publics et un constructeur). Changez la fonction « toString » pour qu'elle affiche les informations de l'utilisateur de cette façon :

```
Utilisateur $user  
  
Courriel : $courriel  
  
Numéro de téléphone : $telephone  
  
Genre : $genre
```

- Après avoir créé votre objet utilisateur faire un « print » de cette variable.

Conseils :

- Utilisez les widgets Flutter appropriés pour chaque élément de l'interface utilisateur (TextField, ElevatedButton, Radio, etc.).
- Utilisez la gestion des états pour gérer les changements dans les champs de texte et les boutons radio. Votre formulaire devrait être un StatefulWidget et un TextEditingController devrait gérer tous les TextField.

Évaluation :

- Partie 1 : Création de l'interface utilisateur (60 points)
 - Champs de texte et boutons correctement disposés : 10 points
 - Boutons radio correctement disposés et fonctionnels : 10 points
 - Esthétique et mise en page globale (incluant le code): 20 points
- Utilisation appropriée des widgets Flutter : 20 points

- Partie 2 : Logique de formulaire (40 points)
 - Validation des données et affichage de messages appropriés : 15 points
 - Gestion des états et interactions des widgets : 15 points
 - Fonctionnalité : 10 points

Bonus 1 : 5 points

Bonus 2 : 2.5 points

Bonus 3 : 2.5 points

Bonus 4 : 5 points.

Notes : à remettre votre fichier main.dart et tous les autres fichier dart que vous avez créé dans un fichier zip.