

## Énoncé de TP Flutter – Formulaire d’inscription

### Objectif du TP

Créer une application Flutter **avec deux écrans**, qui ressemble exactement aux captures d’écran fournies dans l’image jointe :

1. **Écran d’inscription**
2. **Écran affichant les informations saisies**

L’application doit être fonctionnelle et **respecter le style et la disposition de l’image** :

- Dégradé bleu → violet sur l’AppBar
- TextFields avec bordures arrondies
- Messages d’erreur en rouge sous chaque champ
- Bouton **pleine largeur**
- Radio buttons Homme / Femme

### Architecture demandée

L’application sera organisée en **4 fichiers** :

1. **main.dart** : point d’entrée de l’application.
2. **inscription.dart** : écran principal de saisie.
3. **custom\_textfield.dart** : widget réutilisable pour les champs de formulaire.
4. **result\_page.dart** : écran affichant les informations saisies.

Les étudiants devront respecter cette architecture.

### Description des écrans

#### Écran d’inscription

- Titre : **Inscription**
- Champs :
  - Nom complet (texte, obligatoire)
  - Email (texte, obligatoire, format email)
  - Mot de passe (texte masqué, minimum 6 caractères)
  - Retaper mot de passe (texte masqué, doit correspondre au mot de passe)
  - Genre (radio buttons : Homme / Femme, valeur par défaut : Homme)

- Bouton **S'inscrire** prenant toute la largeur.
- Validation :
  - Chaque champ doit afficher un message d'erreur sous le champ si la saisie est invalide.
  - Le bouton ne permet la navigation vers l'écran suivant que si tous les champs sont valides.
- Apparence :
  - Dégradé sur l'AppBar (bleu → violet)
  - Bordures arrondies sur tous les TextFields

### Écran des informations

- Titre : **Informations**
- Affiche les données saisies :
  - Nom complet
  - Email
  - Genre
- Les informations doivent être affichées dans une **Card** ou un **Container** avec un peu d'ombre et des coins arrondis.

### Fonctionnalités demandées

1. **Navigation** : utiliser Navigator.push pour passer de l'écran d'inscription à l'écran des informations.
2. **TextField réutilisable** : créer un widget CustomTextField dans custom\_textfield.dart permettant de gérer :
  - Label et icône
  - Mot de passe masqué avec icône pour afficher/masquer
  - Validation
3. **Validation du formulaire** :
  - Nom et email obligatoires
  - Email au format correct
  - Mot de passe minimum 6 caractères

- Confirmation du mot de passe identique

#### 4. **Gestion du clavier :**

- Page scrollable si clavier ouvert (SingleChildScrollView)
- Fermeture du clavier si clic en dehors des TextFields (GestureDetector optionnel)

#### 5. **Radio Button pour le genre :**

- Deux options : Homme et Femme
- Valeur par défaut : Homme

#### 6. **Bouton pleine largeur** pour valider le formulaire.

#### 7. **Affichage des erreurs** : texte rouge sous chaque champ invalide.

### **Livrables**

- **main.dart**
- **inscription.dart**
- **custom\_textfield.dart**
- **result\_page.dart**
- Capture d'écran de l'application avec les deux écrans fonctionnels.

### **Bonus (optionnel)**

1. Ajouter un **gradient** également dans la ResultPage AppBar.
2. Ajouter une **animation** lors de la navigation.
3. Styliser le bouton avec des **coins arrondis et une ombre**.

Annexe :

2:57

## Inscription

 **Nom complet**  
Veuillez entrer votre nom

 **Email**  
Veuillez entrer un email

 **Mot de passe**   
Minimum 6 caractères

 **Retaper mot de passe** 

☒ Homme ☐ Femme

[S'inscrire](#)

2:59

## Informations

 Mounir salah

 ounir@gmail.com

 Homme