



AZEEOO

Test technique – Flutter

Objectif

L'objectif de ce test est d'évaluer la capacité du candidat à :

- Développer un SDK en Flutter pouvant être intégré dans différentes applications (React Native, Android natif avec Java et iOS natif avec Swift).
- Gérer l'affichage d'un profil utilisateur en récupérant ses données via une API.
- Intégrer le SDK dans une application React Native comprenant une tab bar avec deux onglets.

Détails du Test

1. Création du SDK Flutter

- Développer un SDK en Flutter qui permet d'afficher le profil d'un utilisateur.
- Le profil doit contenir au minimum :
 - Nom
 - Prénom
 - Avatar (image de profil)
- Les données du profil doivent être récupérées via une API REST que nous mettons à disposition :

Unset

```
curl --location 'https://api.azeoo.dev/v1/users/me' \  
  
--header 'Accept-Language: fr-FR' \  
  
--header 'X-User-Id: 1' \  
  
--header 'Authorization: Bearer  
api_474758da8532e795f63bc4e5e6beca7298379993f65bb861f2e8e13c352cc  
4dcebcc3b10961a5c369edb05fbc0b0053cf63df1c53d9ddd7e4e5d680beb514d  
20'
```

- Le SDK doit exposer une méthode permettant d'afficher le profil d'un utilisateur à partir d'un `userId` fourni en paramètre.

2. Intégration dans une application React Native

- Créer une application React Native avec une tab bar contenant **deux onglets** :
 1. **Onglet 1 - Entrée de l'UserId**
 - Une page React Native avec un champ de saisie permettant d'entrer un `userId` (vous pouvez tester avec `userId` : 1 et 3).
 - Un bouton pour sauvegarder l'`userId` en mémoire.
 - Lorsqu'on appuie sur le bouton de sauvegarde, l'onglet 2 doit être rafraîchi pour afficher les nouvelles informations du profil correspondant à l'`userId`.
 2. **Onglet 2 - Affichage du profil**
 - Cet onglet doit appeler le SDK Flutter et afficher l'écran du profil de l'utilisateur en fonction de l'`userId` saisi dans l'onglet 1.
 - Le SDK doit être invoqué correctement dans l'application React Native.

Contraintes Techniques

- **Langages et Frameworks :**
 - Flutter (Dart) pour le SDK
 - React Native (JavaScript/TypeScript) pour l'application de test
 - API REST
- **Interfaçage SDK - React Native :**
 - L'appel du SDK doit être fait de manière propre, soit via un package Flutter Module intégré, soit via Native Modules dans React Native.
- **Tests et Qualité de Code :**
 - Un effort doit être fait sur l'architecture du code.
 - Utilisation de bonnes pratiques de développement et de documentation.

Critères d'Évaluation

Critère	Description
Fonctionnalité	Le SDK fonctionne correctement et affiche bien le profil utilisateur à partir de l' <code>userId</code> .
Intégration	Le SDK est bien intégré dans l'application React Native et est appelé correctement.
Qualité du Code	Code propre, structuré et commenté. Utilisation des bonnes pratiques de développement.
Architecture	Séparation des responsabilités, modularité et facilité d'intégration.
Explication et Documentation	Capacité du candidat à expliquer son choix d'implémentation et à fournir une documentation claire.

Durée du Test

Temps estimé pour la réalisation : **16 heures**.

Ce test permettra de valider la capacité du candidat à travailler sur un SDK destiné à être utilisé dans plusieurs technologies et d'intégrer proprement des modules dans une application hybride.

Livrable

- **Code Source** : Héberger le projet sur GitHub et nous fournir l'accès en nous partageant le lien (samuel.verdier@azeoo.com et yoann.correia@azeoo.com).
- **Documentation** : Inclure un fichier README avec des instructions sur la manière de lancer le projet, ainsi que toute information utile concernant l'architecture ou les choix techniques.
- **Présentation vidéo** : Optionnellement, fournir une vidéo de présentation de l'application (explication des choix techniques, points bloquants rencontrés, etc.).