HELBet – Challenges

# Challenge n°1 - Authentification via Firebase

## Contexte

Voulant intégrer l'authentification via le service **Firebase**, je suis le tuto de mon professeur, monsieur Riggio. Cependant, ça ne fonctionne pas comme prévu pour moi et malgré le fait qu'aucune erreur ne s'affiche explicitement dans la console, mon application ne rajoute pas les nouveaux utilisateurs à la Realtime Database et une log m'informe :

W/System (10365): Ignoring header X-Firebase-Locale because its value was null.

Après plusieures heures de navigation, je ne trouve pas de solution suffisante et quelque chose me tracasse. J'arrive sur une discussion Github assez récente car commencée en janvier 2023 qui est marquée "fermée – sans solution"

Lien GitHub : https://github.com/firebase/flutterfire/issues/10254

Graphical user interface, text, application

Description automatically generated

## Tentative n°1 – Recommencer

Avant de me lancer dans plus de travail fastidieux, je vais simplement réessayer en suivant une vidéo YouTube.

Vidéo : <https://www.youtube.com/watch?v=QAKq8UBv4GI&t=2s>

### Etape 1 – Préparation et liaison du projet Firebase

1. Aller sur le panneau de contrôle Firebase : <https://console.firebase.google.com/u/0/>
2. Créer un nouveau projet
3. Sur la page du projet créé, sur le menu à gauche, *Créer > Authentification*
4. *Sign-in method > Adresse e-mail / Mot de passe*, activer l'authentification par email et mot de passe
5. Sur Android Studio, *Tools > Firebase* et dans le menu contextuel, *Authentication > Authenticate using a custom authentication system*
6. Suivre les étapes :
   1. Lier le projet Firebase au projet Android Studio (cliquer simplement sur le bouton)
   2. Ajouter le SDK d'authentification à l'application en cliquant simplement sur "accepter le changement"

### Etape 2 – Compléter le code pour intéragir avec la db

1. Créer une activité "Login" et une activité "Register"
2. Modeler ces activités comme on le souhaite MAIS respecter ces contraintes :
   1. Un champ pour l'email et un pour le mot de passe chacun ayant son propre ID (registerEmail, loginEmail, registerPswd, loginPswd)
   2. Un bouton login avec son ID pour l'activité Login et un bouton register avec un son ID pour l'activité Register
3. Ajouter au manifest, la permission d'utilisation d'internet
4. Compléter les classes Java pour implémenter les fonctionnalités présentées sur la doc : <https://firebase.google.com/docs/auth/android/password-auth> (j'ai suivi cette vidéo : <https://www.youtube.com/watch?v=QAKq8UBv4GI&t=2s>)

# Résultat 1 – En partie concluant

En partie concluant car je me rend compte que ce n'était pas l'authentification en elle-même qui me posait problème mais la Realtime database.

|  |  |
| --- | --- |
| Graphical user interface, application, Teams  Description automatically generatedPage d'enregisytrement | Graphical user interface, application, Teams  Description automatically generatedPage de connexion |
| Graphical user interface, application, Teams  Description automatically generated  Détails du compte et déconnexion | Graphical user interface, application, Teams  Description automatically generatedMessages d'erreurs type Toast (ici 1 exemple) |

# Challenge n°2 - Liaison avec une Realtime Database de Firebase

## Contexte

Je dois créer une Realtime Database pour stocker les inofrmations de mes utilisateurs. Ceci est mon deuxième essais sur un second projet, ça n'avait pas fonctionné la première fois (voir Challenge 1).

## Tentative n°1 – Suivre une vidéo youtube

Recherche : "setup realtime database firebase android studio"

Vidéo : <https://youtube.com/watch?v=FAa8Wk_uDIA>

### Etape n°1 – Liaison avec Firebase

1. Dans l'assistant firebase d'Android studio, connecter l'app à Firebase
2. Ajouter le SDK (juste accepter les changements)