



Application mobile avec Ionic

Exercice 2 : dans le cloud

Information et évaluation

- La première séance était une initiation
- Mettez vous en binôme pour la suite du cours/exercice
- L'évaluation portera sur :
 - L'avancement dans les exercices
 - La qualité du code et respect des principes du framework
 - Les fonctionnalités

Objectifs

- Avoir une authentification de type SSO
 - Que l'ensemble des données soient stockées sur un serveur distant
 - Ainsi, chaque utilisateur de votre application pourra sauvegarder et récupérer ses propres listes
-
- Le serveur sera ici dans un cloud, accessible via des API REST
 - Plusieurs solutions existent, mais Firebase est la solution attendue

Firebase

- Ensemble de services cloud fournis par Google
 - Gestion de l'authentification
 - Stockage d'informations en ligne
 - Sécurisation
 - Analytique (profil des utilisateurs, etc)
 - ...

⇒ **Solution spécialisée pour le mobile !**

- Pour en savoir plus :
 - <https://firebase.google.com/docs>
 - <https://lesveilleursdenuit.fr/introduction-a-firebase/>

Pré-requis

- Avoir fini les exercices précédents
- Créer un projet Firebase (via votre compte Google)
- S'assurer d'avoir la formule Spark (gratuite, mais limitée par des quotas)
- Configurer ce projet (BD, application, modes d'authentification)
- Choisir le stockage à utiliser :
« Cloud Firestore » (plus récent) ou « Realtime Database »
- Installer la librairie cliente officielle dans votre projet :
<https://firebaseopensource.com/projects/angular/angularfire2/>

Étape 1 : Authentification

- Ajouter une page d'authentification dans l'application :
 - Elle doit gérer au moins un facteur SSO (Google ou Facebook via Firebase)
 - Si possible, l'authentification Facebook doit se faire via le natif du téléphone (<https://ionicframework.com/docs/native/facebook>)
- Si je ne suis pas authentifié, je suis bloqué sur la page d'authentification
 - Conseil : intervenir au niveau du routeur de votre application (authentification réalisée avant d'accéder aux pages ?)
- Ajouter une page « profile » où l'on peut voir son profil et se déconnecter
- Liens utiles
 - <https://github.com/angular/angularfire/blob/master/docs/install-and-setup.md>
 - <https://github.com/angular/angularfire/tree/master/docs/auth>



Étape 2 : Stockage des listes et items dans le cloud

- L'ensemble des informations ne sont plus stockées de manière volatile dans le service, mais entièrement dans le cloud Firebase.
- Dans un premier temps, se contenter de sauver ses listes dans le cloud et de les y récupérer lorsque l'on recharge l'application.
- Dans un deuxième temps, écouter les changements sur la BD, de sorte que 2 appareils partageant la même liste se synchronisent automatiquement via Firebase.
- **Attention de ne pas stocker les secrets Firebase dans Git !**
ApiKey, etc. Documenter comment définir un environnement dans le fichier lisez-moi.

