Le but de se Sprint étant le démarrage du project ionic cordova la creation du project sur le cloud firebase l’impémentation de authetification sur l’applicatione en utilistan le service de l’uthetification de firebase .

# Firebase Authentification simplifie l'authentification, un système d’authentification peut être difficile à comprendre et peut causer de lourdes conséquences s’il est mal géré. Firebase Authentification proposer un service prêt à l’emploi qui ne va pas stocker les identifiants, mais assurer le mécanisme de l’authentification elle-même et il est indépendant du stockage des données.

# Le schéma ci-dessous illustre les diférent service Authentification compatible avec firebase dans le cadre de se project on dispose de 3 system :

# Authetification email password

# Authetification en utilisan le compte facebook

# Authetification en utilsan le compte google

# C:\Users\yassine\AppData\Local\Microsoft\Windows\INetCache\Content.Word\facebook-provider.png

# Pour le mondant l’authentification social présente un intérêt majeur car ça facile l’accès à l’application et ça encourage les personne ayant télécharger l’application à utiliser en évitant l’étape de création de compte mais les utilisateurs n’ayant pas de compte google ou facebook peuvent toujours crée un compte en fournissant un email et un mot de passe

# Le services d’authetification de firebase n’a aucun lien avec la base de donnée ce qui veut dire qu’on a pas accès au info de l’utilisateur en ayan accès a l’application ce qui présente un avantage majeur au niveau de la sécurité j’ai même constaté que même en tant qu’administrateur depuis la console d’administration on peut envoyer un mail de reset crée un utilisateur supprimer voir son id mais les mot de passe ne sont pas disponible en clair voilà à quoi ressemble la console d’admin user sur firebase avec les différent action possible

# C:\Users\yassine\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Capture.jpg

Pour l’implémentation de services sur notre application j’ai due passer par plusieures étape la premier étant d’activer les services requis sur firebase comme mentionner dans le shema pluhaut les mode de connexion activé sont :

Adresse e-mail/Mot de passe se mode Permet aux utilisateurs de s'inscrire avec leur adresse e-mail et leur mot de passe. Le SDK de firebase proposent également la validation de l'adresse e-mail, la récupération du mot de passe et les primitives de modification de l'adresse e-mail.

Google Sign-In est automatiquement configuré dans notre applications mais pour Facebook la configuration nécessité la création d’un compte facebook dévelopeur et l’autorisation de nôtre application a utilisé se services .

Après avoir terminé toute les configuration et les autorisation requise au niveau des de firebase et facebook reste la partie client de l’application qui consite en :

Création d’une page de login qui propose

Les 3 mode de connexion

La possibilité de crée un compte si utilisateur n’a pas de compte social ou souhaite utilisé un email et mot de pass .

La possibilité de récupéré le mot de passe.

Les bonnes pratiques au niveau du frimewark ionic consiste en la création d’un service qui gère l’authentification et après l’injecter dans chacune des pages qui ont en besoin dans notre

Application on auras 3 pages :

Page de login

Page de récupération de mot de passe

Page de création de compte

Les fonctions de connexion avec login,Facebook et Google sont implémenter dans le service d’authentification en utilisant le sdk de firebase pour l’authentification mais en passant par AngularFire et son module d’authentification qui est un module qui gère la liaison avec Firebase afin qu'elle soie synchronisés de manière transparente et instantanée en plus de fonctionalité de creation de compte et de connexions se service implémente aussi les fonction de déconnexion, reset du mot de passe une des méthode les plus important de se services est le monitoring de état de authetification, En utilisant un observateur, on s’assure que l'objet Auth n'est pas dans un état intermédiaire - comme l'initialisation – lorsqu’on obtenez l'utilisateur actuel. Lorsque vous utilisez l'observateur OnAuthStateChanged attend jusqu'à ce que getRedirectResult décide avant le déclenchement ce qui permet de restriendre l’accées au fonctionalité de l’application pour les utilisateurs non authetifier cet état est géré par firebase donc même si l’implémentation est au niveau client.