**Utilisation du Projet**

**Côté serveur :**

Voici les différentes URL :

**API Google**

* Création d’un utilisateur :

http://localhost:8080/user/add/{userKey}

* Création de compte : http://localhost:8080/account/add/ {accountName} ?userKey=UserExistantDansBDD
* Consultation du nombre de mail non lu : http://localhost:8080/emailNonLuORM?userKey= UserExistantDansBDD
* Consultation du prochain évènement du calendrier : http://localhost:8080/calendarNextEventORM?userKey= UserExistantDansBDD
* Consultation du nombre de contact :

http://localhost:8080/contact?userKey= UserExistantDansBDD

* Affichage des utilisateurs sauvegardé dans le credential :

http://localhost:8080/admin

Pour pouvoir ajouter différent utilisateur il faut toujours déconnecter son compte google actif ou passer sur un autre navigateur.

**API Microsoft**

* Page d’acceuil :

http://localhost:8080

* Création d’un utilisateur comme pour google
* Création de compte :

http://localhost:8080/add/microsoftAccount/ {accountName} ?userKey=UserExistantDansBDD

* Consultation du nombre de mail non lu : http://localhost:8080/mail?userKey= UserExistantDansBDD
* Consultation du prochain évènement du calendrier : http://localhost:8080/events?userKey= UserExistantDansBDD
* Consultation du nombre de contact :

http://localhost:8080/contacts?userKey= UserExistantDansBDD

**Côté client :**

Pour lancer les .jar il faut lancer deux invites de commande puis aller dans le répertoire du jar côté serveur puis de lancer la commande : java -jar « NomDuJar.jar ».

Ensuite il faut aller sur la deuxième invite de commande et se rendre dans le répertoire du jar client. Une fois dans le bon répertoire, lancer les commandes que vous souhaitez parmi les suivantes :

* Ajout d’un compte : java -jar « NomDuJar.jar » add NomduCompte
* Voir les informations du compte :
  + java -jar « NomDuJar.jar » view NomduCompte
  + java -jar « NomDuJar.jar » NomduCompte

**Prérequis :**

Pour pouvoir lancer le serveur il faut commencer par lancer la base de données pour stockées les utilisateurs, et les différents compte Microsoft et Google.

**Configuration :**

* Paramétrage de la **base de données** :

Il faut aller dans la partie serveur, puis dans le dossier ressources et ouvrir « application.properties ».

Une fois la page ouverte il faut renseigner les informations pour que votre base de données fonctionne correctement, comme vous pourrez le voir ci-dessous :

spring.datasource.url=jdbc:mysql://localhost:3306/javaproject

spring.datasource.username=root

spring.datasource.password=

spring.datasource.driverClassName=com.mysql.jdbc.Driver

spring.jpa.database=MYSQL

spring.jpa.show-sql=true

spring.jpa.generate-ddl=true

spring.datasource.continue-on-error=false

* Paramétrage de l’emplacement du fichier **credentials.json**

Par défaut le fichier doit être le dossier « Java\_Serveur » qui doit se trouver dans le dossier du user courant.

Si vous voulez le déplacer il faut aller dans la partie Serveur, puis aller dans la package fr.ynov.dap.dap.config et ouvrir le fichier Config.java.

Ensuite il faut chercher la constante CREDENTIALS\_FILE\_PATH et modifier sa valeur qui est actuellement :

System.*getProperty*("user.home") + System.*getProperty*("file.separator") + "Java\_Serveur" + System.*getProperty*("file.separator") + "credentials.json";

Pour ajouter un séparateur il faut passer par : System.*getProperty*("file.separator")

* Paramétrage de l’emplacement et du contenu de **auth.properties**

Pour le fichier auth.properties, par défaut il se trouve dans le même dossier que le credentials.json. Pour le modifier il faut aller dans le fichier Config.java et cette fois-ci modifier la valeur de la constante : AUTH\_PROPERTIES.

En ce qui concerne le contenu, il faut ajouter votre id et votre mot de passe que l’api de Microsoft vous a fourni comme vous pouvez le voir ci-dessous :

appId=833849d8-b392-4d97-999e-1b067683dd5f

appPassword=ysfdYFAW82^)ykqDTK457#$

redirectUrl=http://localhost:8080/authorize

**Bonne utilisation**