**Authorization Basic**

Il n’est pas possible avec le swagger de surcharger le header autorisation. J’ai donc décidé de passer par une authentification key pour pouvoir passer la chaîne de caractères contenant Basic username:password dans le header authorization. Une fois le token transmit dans le header je le récupéré et je le passe pour récupérer les identifiants.

On vérifie si l’utilisateur existe dans la table user ou dans la table admin si oui il est autorisé à exécuter l’api

**Génération de QCM**

Une fois connecté l’utilisateur pourra afficher la liste des « uses » disponibles en base et les sujets associés pour chaque use. Ceci pour mieux soumettre l’enveloppe de génération de QCM.

Pour la génération de QCM par lot de 5, 10 et 20 aléatoirement, si le nombre de questions pour un « use » donnée et un ou plusieurs sujets données est inférieur à 5 lorsqu’on veut créer un questionnaire de 5 questions (10 lorsqu’on souhaite 10 questions, 20 lorsqu’on souhaite 20 questions) on affiche le nombre de questions trouvés en base. Si aucune question n’est trouvée on affiche une liste vide.

**Statut de l’API**

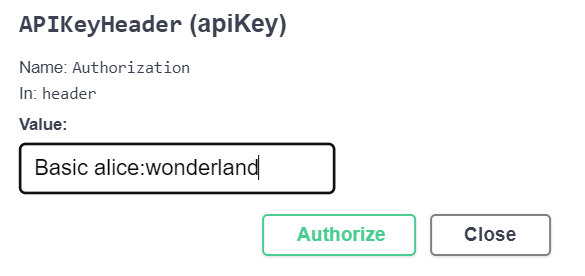
Avant de pouvoir faire des requêtes nous pouvons tester si l’api est fonctionnelle. Pour cela pour ce point de terminaison nous avons choisi de le laisser en public à tous ; pas besoin d’être autoriser pour cette requête.

**Test de l’API**

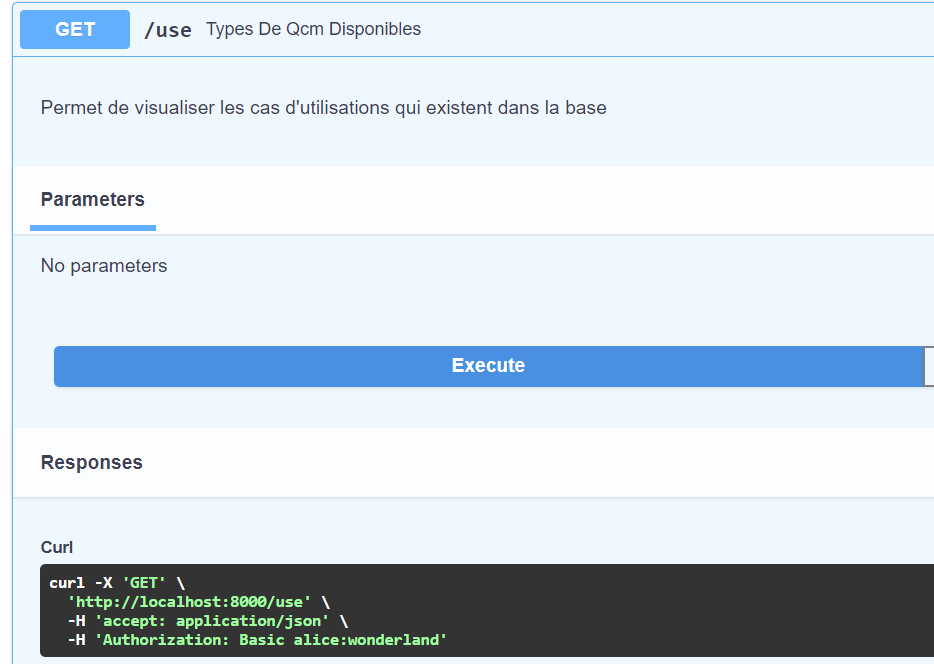
Pour tester l’API il faut :

1/

* Installer les librairies (fichier requirements.txt ou requirements\_all\_venv\_libs.txt)
* Exécuter l’api avec la commande « **uvicorn main:app --reload** »
* Se connecter à l’api [**http://localhost:8000/docs**](http://localhost:8000/docs)
* Une fois connecter vous pouvez tester les différents points de terminaisons (à l’exception **/status** et **/users** tous les points de terminaisons nécessitent une autorisation)
* Pour les points de terminaisons nécessitant une authentification :
  + Renseigner les informations de connexions sous la forme « **Basic username :password** » dans la partie « Authorize » qui sera transmise dans le header



* + Sélectionner ensuite le test à réaliser. Une fois exécuté on peut voir que les informations ont été transmis dans le header



2/ on peut choisir de directement faire le test en ligne de commande

* Ajout d’un utilisateur

**curl -X 'PUT' \**

**'http://localhost:8000/users' \**

**-H 'accept: application/json' \**

**-H 'Content-Type: application/json' \**

**-d '{**

**"username": "Bob",**

**"password": "Builder"**

**}'**

* Affichage des uses disponibles

**curl -X 'GET' \**

**'http://localhost:8000/use' \**

**-H 'accept: application/json' \**

**-H 'Authorization: Basic alice:wonderland'**

* Affichage des subjects disponibles

**curl -X 'POST' \**

**'http://localhost:8000/qcm' \**

**-H 'accept: application/json' \**

**-H 'Authorization: Basic alice:wonderland' \**

**-H 'Content-Type: application/json' \**

**-d '{**

**"use\_type": "Test de validation",**

**"questions\_number": 5,**

**"subject": [**

**"Classification"**

**]**

**}'**

* Génération d’un QCM de 5 questions (on peut le relancer plusieurs fois pour voir que les questions affichent aléatoirement)

**curl -X 'POST' \**

**'http://localhost:8000/qcm' \**

**-H 'accept: application/json' \**

**-H 'Authorization: Basic Fidele:Zongo' \**

**-H 'Content-Type: application/json' \**

**-d '{**

**"use\_type": "Test de positionnement",**

**"questions\_number": 5,**

**"subject": [**

**"BDD", "Docker"**

**]**

**}'**

* Ajout d’une question

**curl -X 'PUT' \**

**'http://localhost:8000/questions' \**

**-H 'accept: application/json' \**

**-H 'Authorization: Basic admin:4dm1N' \**

**-H 'Content-Type: application/json' \**

**-d '{**

**"question": "FastAPI Permet il de",**

**"subject": "FastAPI",**

**"use": "FastAPI",**

**"correct": "Mettre en place des API rapide",**

**"responseA": "De gagner de l'\''argent",**

**"responseB": "Marcher vite",**

**"responseC": "Voyager",**

**"responseD": "Permet une documentation automatique",**

**"remark": "Ajout question"**

**}'**

Pour vérifier l’ajout de la question précédente on peut exécuter ces requetes :

**curl -X 'GET' \**

**'http://localhost:8000/use' \**

**-H 'accept: application/json' \**

**-H 'Authorization: Basic admin:4dm1N'**

Où

**curl -X 'GET' \**

**'http://localhost:8000/subject?use=FastAPI' \**

**-H 'accept: application/json' \**

**-H 'Authorization: Basic admin:4dm1N'**