**Tutoriel d’utilisation de la plateforme en local**

* Duplication en local des fichiers back
  + Créer un dossier dédié au back
  + Dans ce dossier, dézipper l’archive **back.zip** Ce dossier contient alors, entre autres, les scripts python assurant différents traitements sur les données, les fichiers d’initialisation de la base de données ainsi que divers fichiers de configuration docker, en particulier le fichier « docker-compose.yml » qui configure les différents conteneurs et permet de les lancer.
  + Note : il peut s’avérer nécessaire de modifier les ports définis dans le fichier « docker-compose.yml » si le lancement des conteneurs pose problème
* Lancement des conteneurs back
  + Se placer dans le dossier du serveur dédié au backend (contenant le fichier « **docker-compose.yml** »)
  + Si le dossier « data » existe, le supprimer pour que les containers soient entièrement reconstruits (il peut être nécessaire de stopper les containers en cours d’exécution au préalable pour cela)
  + Exécuter la commande **docker compose up -d**
  + Les trois conteneurs sont lancés : scripts backend, base de données et phpMyAdmin
* Duplication en local des fichiers front
  + Créer un dossier dédié au front, se placer dedans et dézipper l’archive **front.zip**
* Lancer le front en local
  + Dans un terminal, celui de Visual Studio Code par exemple, se placer dans le dossier principal du front (dossier créé précédemment) et lancer la commande **npm install** pour installer toutes les dépendances
  + Dans les fichiers **backend\_services.js** et **db\_interactions.js** (dossier « services ») s’assurer que les variables baseURL pointent vers l’adresse désirée selon qu’il s’agisse d’une utilisation en local ou d’un déploiement sur serveur
  + Dans ce même terminal exécuter **npm run serve** pour lancer le front en local
* Note : pour accéder à phpMyAdmin, écrire : localhost/(port du container phpMyAdmin) dans la barre de recherche du navigateur