|  |  |
| --- | --- |
| **Data Mining – Big DATA**  **Projet de fin de semestre** | |
| NINH Thi Hoa Lan |  |
| ANGOSTO Estelle | Mr SOUIDI Mohamed |
| MASLIAH Anita | Année scolaire 2021-2022 |

ITS2

**Table des matières**

[Table des figures 2](#_Toc95858826)

[Introduction 3](#_Toc95858827)

[Data source 3](#_Toc95858828)

[Structure du Pipeline de Traitement 4](#_Toc95858829)

[Conclusion 7](#_Toc95858830)

# Table des figures

[Figure 1 : Structure du projet général 3](#_Toc95858815)

[Figure 2 : Structure de notre projet 4](#_Toc95858816)

[Figure 3 : Producer 5](file:///C:\Users\Utilisateur\Documents\Semestre3_ITS\Big_Data\Projet\BigData\Rapport_Projet_2.docx#_Toc95858817)

[Figure 4 : Consumer Spark 5](file:///C:\Users\Utilisateur\Documents\Semestre3_ITS\Big_Data\Projet\BigData\Rapport_Projet_2.docx#_Toc95858818)

[Figure 5 : Script My SQL 6](file:///C:\Users\Utilisateur\Documents\Semestre3_ITS\Big_Data\Projet\BigData\Rapport_Projet_2.docx#_Toc95858819)

# Introduction

Le traitement de donnée se fait de manière générale sur des fichiers de données persistantes. Mais que ce passe-t-il lorsqu’on souhaite traiter un flux de donnée en temps réel ? Les volumes de données ne cessent d’augmenter. D’ici à 2025, de 88 % à 97 % des données mondiales ne seront pas stockées, les données seront collectées et traité en mémoire en temps réel. Il faut alors utiliser le streaming et utiliser un bus de données. Ainsi dans le cadre de ce module nous allons réaliser un pipeline de traitement de données à l’aide de Python, Kafka, Spark, MySQL et Superset. Ce pipeline permettra la collecte des données, leur organisation, leur transformation et enfin leur affichage. Le but étant d’appliquer toutes ces étapes pour permettre une transmission sécurisée des données transformées.

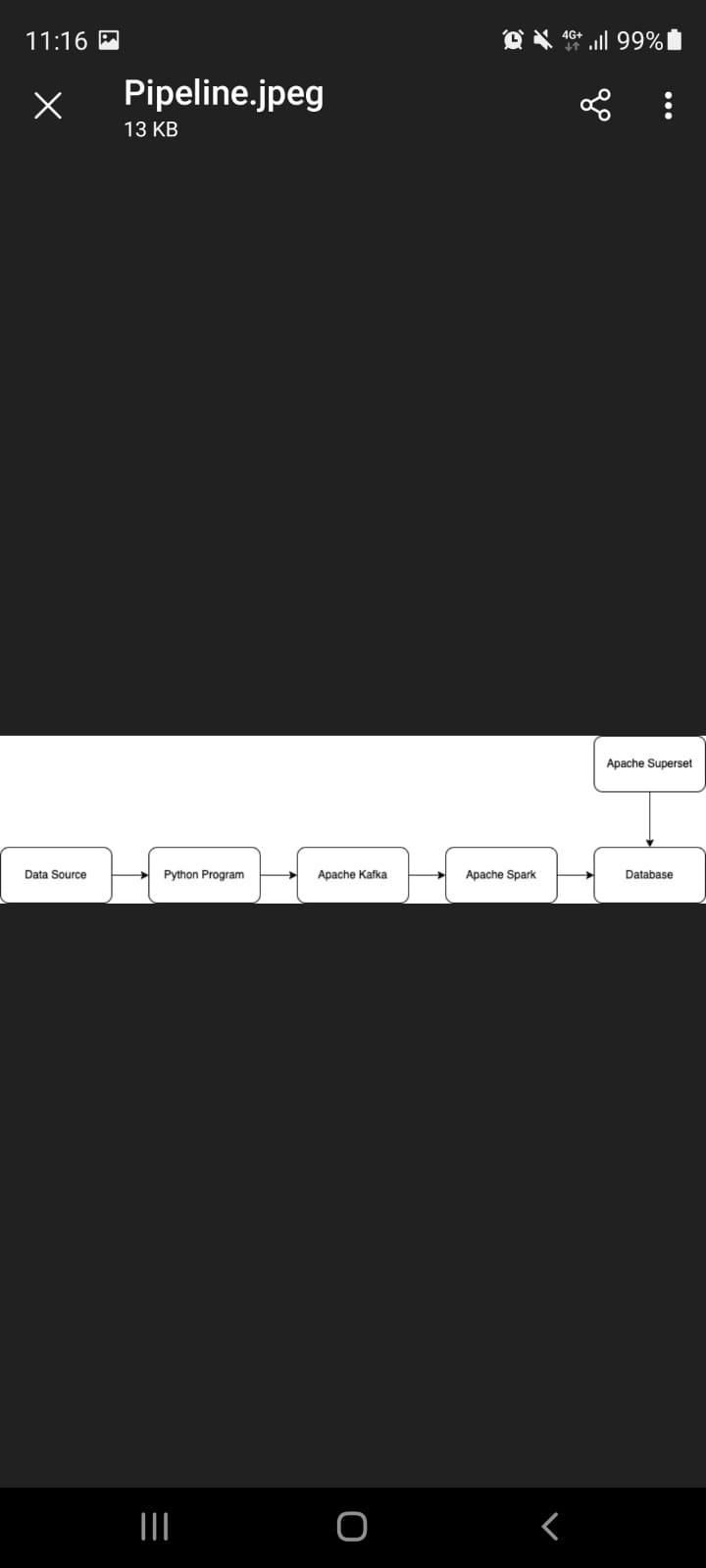


Figure 1 : Structure du projet général

# Data source

L’idée première était de récupérer des tweets concernant le Covid 19 en créant une application Tweeter. Nous avions d’ailleurs détaillé notre démarche dans le précédent rapport. Cependant, n’ayant jamais reçu de réponse favorable de la part de Tweeter, nous avons décidé de nous rabattre sur l’utilisation d’un fichier CSV récupéré sur Kaggle traitant du même sujet.

Ce fichier contient : Le nom des pays, leur code ISO 3166 Alpha 3, la date, le taux d’incidence (cas, mort, guérit), l’augmentation journalière (cas, mort, guérit), le taux de contagion empirique...

Nous n’afficherons pas toutes les données et pour ce faire nous sélectionnerons seulement certaines données soit grâce au langage SQL en modifiant la table MySQL, soit en modifiant les RDD crées avec Spark.

# Structure du Pipeline de Traitement

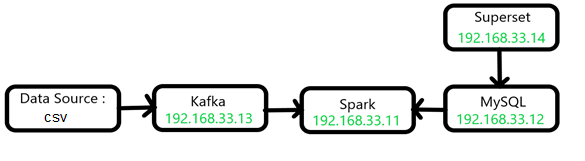


Figure 2 : Structure de notre projet

Les machines virtuelles contenant les différents logiciels utilisés ont été détaillées dans le précédent rapport. Les liens entre les différentes VM et l’envoie/la récupération des données se fait grâce à Python.

**Kafka**

Apache Kafka est un logiciel Open Source qui permet l’enregistrement, la transmission et le traitement des données. Sa principale fonction est la centralisation et l’optimisation des flux de données. Le logiciel se comporte comme un messager entre l’émetteur (la source/ producer) et le récepteur (Spark/consumer). Nous passons par une méthode avec un fichier semi- structuré json qui est rempli grâce à notre fichier csv. On créer ainsi le producer Kafka.

Une image contenant texte

Description générée automatiquement

Figure 3 : Producer

**Spark**

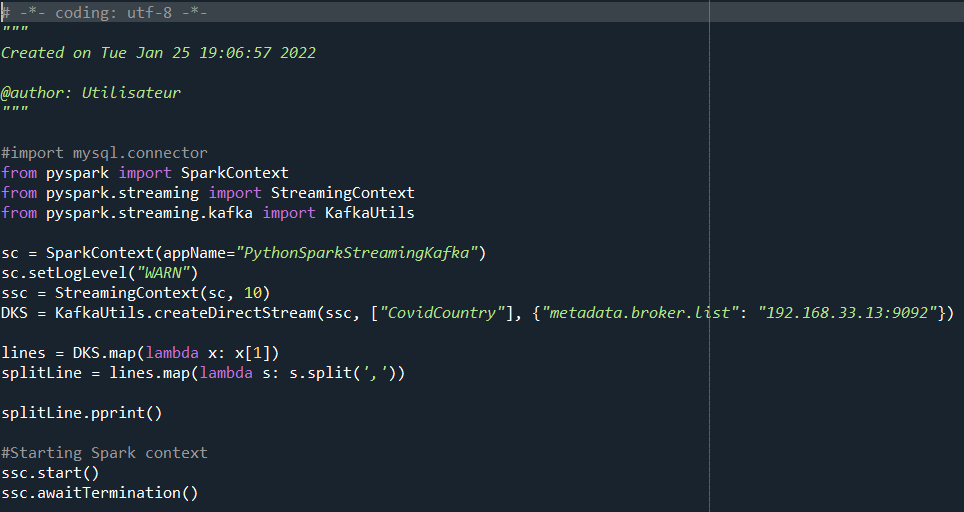
Spark est un framework open source. Plus précisément c’est une infrastructure de traitement de données distribuées. Spark va se comporter comme un consumer pour Kafka et récupérer les données qui seront ensuite transformées en RDD. Les RDD (Resilient Distributed Datasets) sont une collection immuable d’objet permettant de persister explicitement les données intermédiaires en mémoire, de contrôler leur partitionnement afin d’optimiser l’emplacement des données, de manipuler les données grâce à des opérateur.

Figure 4 : Consumer Spark

**MySQL**

C’est un système de gestion des bases de données relationnelles basée sur le language SQL (Structured Query Language). Il va nous permettre d’effectuer des requêtes afin de trier les données et d’en récupérer seulement une partie. Par exemple les dernières colonnes du fichier CSV ne nous intéressent pas. Une base de données ainsi qu’une table a été créée pour ce projet.

Une image contenant texte

Description générée automatiquement

Figure : Script My SQL

**Superset**

C’est un logiciel open source qui permet de visualiser des données et notamment un grand nombre de données. Il permet également l’exploration des données de manière dynamique.

# Conclusion

Ce projet nous a fait découvrir un nouveau thème, par conséquent, nous avons rencontré de nombreuses difficultés lors de la mise en œuvre du projet. Tout d'abord, comme mentionné ci-dessus, notre intention initiale était d'utiliser les données sur Twitter, mais cela n'a pas été possible car notre demande n'a pas été approuvée. Cela nous a pris pas mal de temps à cette étape. Après cela, nous avons eu pas mal de problèmes pour récupérer les données, des problèmes de machine virtuelle, etc.

Malgré ces nombreuses difficultés, nous avons toujours tenter de faire de notre mieux pour avancer dans ce projet. Bien que le projet n'ait pas atteint son stade final, nous avons également appris beaucoup de connaissances différentes liées au projet dans le processus de résolution des problèmes.

Cela nous donne une meilleure base pour mener à bien nos futurs projets.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |