**État d'avancement du projet**

Date : 23 juin 2025

Projet : Analyse de logs en temps réel avec Spark & Kafka

Groupe: Yannick BOYER, Robin PEIGNET, Yohann DUBOEUF

# Tâches réalisées

1. Générateur de logs (log\_generation\_complete.py) — Complet

* Génère des logs au format Apache
* Configurable via la CLI :
  + Vitesse de génération
  + Pourcentage d’URL générant des erreurs
  + Pourcentage des utilisateurs générant des erreurs
  + Probabilité que la requête soit une erreur
* Envoi les logs à un conteneur Kafka sur un topic (http-logs)

1. Analyse Spark Streaming (log\_analysis\_complete.py) — Complet

* Lit les logs depuis le topic Kafka (http-logs)
* Détection des batchs
* Calcule les métriques suivantes :
  + Taux d’erreur
  + Taux d’erreur par IP
  + Taux d’erreur par URL
* Compare ces métriques par rapport à des seuils
* Si ces seuils sont dépassés, le programme envoi des logs dans le topic alerts du conteneur Kafka

1. Automatisation du déploiement — Complet

* Le fichier docker-compose.yml permet de déployer tous les conteneurs nécessaires pour faire fonctionner le projet :
  + Kafka et Zookeeper
  + Générateur de logs
  + Job Spark Streaming
* Afin de démarrer le projet, il suffit de lancer la commande :

docker compose up -d

* Pour stopper le projet, il faut lancer la commande :

docker compose down -v

1. Calcul des seuils — Manquant

* A ce jour, le calcul des seuils ne sont pas fait automatiquement

1. Sécurité (chiffrement et authentification Spark/Kafka) — À configurer

* Le chiffrement et l’authentification de Spark et Kafka ne sont pas encore réalisés.

1. Visualisation / monitoring — À mettre en place (Grafana/Prometheus)

# Livrables fournis

* log\_generation\_complete.py
* log\_analysis\_complete.py
* docker-compose.yml
* Documentation des deux programmes