

L'architecture typique aura 5 composants:

1) avoir un petit programme simulant le drone et envoyant des données de type drone à votre solution (voir le sujet pour plus de détails sur un message). Votre système stockera le message dans un flux distribué le rendant disponible pour les composants 2 et 3.

2) gérer le message d'alerte du flux

3) stocker le message dans un stockage distribué (ex: HDFS / S3)

4) analyser les données stockées avec un traitement distribué composant (comme une étincelle). Pour prouver la capacité de votre système à analyser les données du magasin, répondez à 4 questions de votre choix. (ex: y a-t-il plus de violations pendant la semaine ou le week-end?)

5) chargez le fichier CSV dans votre composant de stockage distribué. Cela doit être fait (couple) de ligne par (couple) de ligne.

Tous les composants doivent être évolutifs et utilisés de manière évolutive. Pour le composant 3), vous pouvez utiliser kafka connect ou son équivalent (kinesis firehose). Tout code doit être écrit en scala fonctionnel (la compilation en jvm sur javascript n'a pas d'importance). Sauf si je l'accepte à titre d'exception, les mots clés «for, while, return, var» sont interdits ainsi que l'importation de tout élément mutable.

Une exception pour l'instant: si vous souhaitez afficher un certain nombre de messages ou d'alerte reçus / stockés... vous pouvez utiliser le mot-clé.

Certains étudiants peuvent choisir en option de coder les cinq composants dans un autre langage fonctionnel (F #, Haskell...).

Si vous êtes intéressé, vous pouvez écrire le simulateur de drone dans Go.

Pour le projet, vous devez utiliser un dépôt git, le travail des différents membres du groupe doit être visible dans différents commits. Pour la soumission, vous devez m'envoyer un e-mail avec votre dépôt git et le dernier hachage de validation La soumission tardive est acceptée, moins 2 points par jour (s) tardif (s).

Une fois ces 5 parties, vous pouvez travailler sur la partie facultative:

la partie facultative est assez ouverte, l'objectif est que chaque groupe travaille sur quelque chose qui est curieux ou qu'il trouve intéressant pour son CV. Ils peuvent être effectués dans la langue de votre choix, sauf si vous utilisez une étincelle.

Voici quelques suggestions: 1) projet déployé sur le cloud (azur, aws, gcp,...) en utilisant IaC comme terraform.

2) site Web utilisant sa base de données / file d'attente dédiée pour afficher chaque alerte reçue (au lieu d'une impression de base de courrier électronique / journal / console)

3) utilisant docker et docker composer pour le composant 5 du projet

4) une fois le composant 4 terminé, en utilisant spark -notebook / zeppelin pour générer des graphiques

5) en utilisant un dataviz ou un site Web personnalisé pour présenter le résultat de l'analyse d'étincelles

6) en utilisant un modèle ml et en ajoutant des informations dans le message pour réaliser la maintenance prédictive du drone.