

**Proposé par.** Aimene BELFODIL, Adnene BELFODIL et Enzo MEROTTO

Ce projet a été inspiré d'une vraie collaboration entre l'entreprise Atos et le laboratoire LIRIS à l'INSA de Lyon.<sup>1</sup>



Figure 1: Ferme Urbaine Verticale (Atos x LIRIS INSA Lyon 2018)

## 1 Context et Introduction

Une ferme urbaine verticale permet de cultiver des quantités significatives de produits alimentaires dans des tours, parois ou structures verticales, afin de développer l'agriculture de proximité (villes, quartiers) ou pallier le manque de terres cultivables en milieu hostile.

Outre l'avantage cité ci-dessus, une ferme urbaine verticale peut être équipée de divers capteurs la rendant facilement supervisable et donc pilotable. Cela permet entre autres:

---

<sup>1</sup>Source: [https://atos.net/fr/2018/news-fr\\_2018\\_01\\_16/atos-lance-le-premier-projet-de-ferme-urbaine-verticale-et-digitale-au-monde](https://atos.net/fr/2018/news-fr_2018_01_16/atos-lance-le-premier-projet-de-ferme-urbaine-verticale-et-digitale-au-monde)

- **La maintenance assistée.** L'opérateur de la ferme utilise une tablette tactile équipée de réalité virtuelle pour faciliter son diagnostic et la résolution des pannes ;
- **La maintenance prédictive,** qui permet l'émission de signaux précurseurs de pannes. Ces signaux peuvent déclencher des alarmes de panne imminente, ou encore aider les opérateurs à estimer la durée de vie restante des composants matériels, pour anticiper les pannes et optimiser la production ;
- **L'optimisation puis l'implémentation de recettes de culture.** Une recette de culture correspond à l'ensemble des conditions scientifiques permettant le développement d'une plante. L'analyse de données permettra notamment d'identifier ces recettes pour le meilleur développement possible de la plante.

La suite de l'énoncé de ce projet est une mise-en-context et ne représente pas un cas d'usage réel. Il est en effet inspirée du projet issue de la collaboration entre le laboratoire de l'INSA de Lyon et Atos en 2018.

## 2 Objectif du projet

L'objectif de ce présent projet est la réalisation, le développement et la simulation de déploiement d'une architecture cloud permettant de:

1. Recevoir en temps réel les données des différents capteurs de la ferme urbaine simulée.
2. Stocker l'historique de ces données afin de les rendre exploitable.
3. Permettre a un opérateur de superviser les données d'un capteur via une interface simplifiée.
4. Permettre de déceler les anomalies des données remontées (anomalie sur capteur ou anomalie sur la plantation) afin de notifier au plus vite l'opérateur de la ferme des problèmes potentiels.

## 3 Données du projet

Nous nous mettons en situation ou nous disposons de capteurs (sensors) mesurant deux grandeurs: la **temperature** de la plante ainsi que son **taux d'humidité**.

Ces mêmes capteurs sont achetés chez deux fournisseurs différents: Un fournisseur anglais de la marque **EN-v2** et un fournisseur français identifié **FR-v8**.

Les deux capteurs sont censés mesurer à la fois la température et le taux d'humidité. Ils peuvent être configuré pour envoyer leurs données en REST (utilisant la méthode **POST**) sur une **URL**. Le Payload du **POST** est en format **msgpack** <sup>2</sup> encodé en base64<sup>3</sup>. Un exemple de données envoyée est donnée ci-dessous:

```
halzZW5zb3JfaWSmNzQ2MzEyrnNlbnNvc192ZXJzaW9upUZSLXY4qHBsYW50X2lkzG
AAAAGkdGltZbQyMDIOLTAzLTIOVDE4OjIxOjI4WqhtZWZzdXJlc4KrdGVtcGVyYXR1
cmWlMTLCsEOoaHVtaWRpdGWjMTI1
```

Il arrive qu'on utilise deux capteurs des deux fournisseur sur la même plante dans les but de les comparer.

Nous vous fournissons un simulateur d'une mini ferme urbaine de ces capteurs dans le lien

*[https://moodle.ia-institut.fr/pluginfile.php/2598/mod\\_folder/content/0/docker-compose.yaml?forcedownload=1](https://moodle.ia-institut.fr/pluginfile.php/2598/mod_folder/content/0/docker-compose.yaml?forcedownload=1)*

Chaque instance docker simule un scenario de récolte de données avec l'un ou l'autre type de capteur. Il y'a 7 scenarios prédéfinis plus un scenario paramétrable afin de vous permettre de tester des cas d'usage différents de la ferme urbaine simulée.

## 4 Rendu détaillée du projet

1. Proposez et dessinez une architecture qui répond à l'objectif susmentionné. L'architecture proposée doit mettre en oeuvre plusieurs composants indépendants dont une base de données de type relationnel. Il est demandé que:
  - (a) Le schema de l'architecture ainsi que celui de la base de données doivent être présent dans le rendu.
  - (b) L'architecture doit contenir au moins trois composants indépendants.
  - (c) La solution proposée doit permettre de requêter en REST (GET) les données historiques de chaque plante (et capteur).
  - (d) La solution proposée doit permettre de visualiser les données historiques de chaque plante (et capteur).

---

<sup>2</sup><https://msgpack.org/index.html>

<sup>3</sup><https://en.wikipedia.org/wiki/Base64>

- (e) La solution doit permettre au moins de détecter "automatiquement" un type d'anomalie. Vous avez la charge d'imaginer le type d'anomalie que votre solution supporte.
- 2. Implémentez l'architecture que vous proposez à l'aide des langages de programmation de votre choix. Chaque composant doit être conteneurisé en utilisant docker.
- 3. Déployez votre architecture en utilisant kubernetes sur minikube ou kind. Vous devez utiliser au moins une fois l'outil helm pour un de vos composants<sup>4</sup>.
- 4. Proposez une solution de détection d'anomalies en utilisant les données simulées. Vous devez mettre en oeuvre au moins une technique avancée de détection d'anomalie:
  - (a) Utilisez plusieurs capteurs paramétrés afin de prouver que votre algorithme de détection d'anomalie atteint votre objectif.
  - (b) Quelques anomalies ont été introduites dans les 7 scénarios prédéfinis.
- 5. **(Bonus)** Proposez et mettez en oeuvre un moyen de superviser l'état de santé de votre architecture de récolte de données (Vous pouvez utiliser prometheus et grafana<sup>5</sup>).

## 5 Organisation

Le projet se fera en groupe de 3 personnes. Les groupes sont prédéfinis à l'avance. Nous restons (Aimene, Adnene et Enzo) à votre disposition tout au long de la réalisation du projet pour tout type de question.

## 6 Présentation

Le rendu final du projet est prévu pour le 12 Avril 2024 entre 13h30 et 16h30. Chaque groupe doit présenter (oral + démo live) de ce qu'il a réalisé. Veuillez noter que:

- Les trois intervenants de chaque groupe doivent participer à la présentation orale.

---

<sup>4</sup>Par exemple postgres en utilisant <https://artifacthub.io/packages/helm/bitnami/postgresql>

<sup>5</sup>Vous pouvez utiliser <https://bitnami.com/stack/grafana/helm> and <https://artifacthub.io/packages/helm/bitnami/kube-prometheus>

- La durée de la présentation avec démo est de **20mn** suivi de **20mn** de question. Les questions peuvent se porter sur le code, le fonctionnel ou les perspectives.
- **30%** de la note concerne la forme de la présentation orale.
- La note finale du projet est individuelle.