**Etude de cas Arcane : consignes.**

**Problématique :**Dans le cadre d’un projet de création d’une application web de gestion immobilière, on nous demande de créer un ensemble de microservices. Ces microservices doivent permettre à un utilisateur de renseigner un bien immobilier avec les caractéristiques suivantes : nom, description, type de bien, ville, pièces, caractéristiques des pièces, propriétaire) et de consulter les autres biens disponibles sur la plateforme.   
  
Nous souhaitons proposer aux utilisateurs les fonctionnalités suivantes :

* Un utilisateur peut modifier les caractéristiques d’un bien (changer le nom, ajouter une pièce, etc… )
* Les utilisateurs peuvent renseigner/ modifier leurs informations personnelles sur la plateforme (nom, prénom, date de naissance)
* Les utilisateurs peuvent consulter uniquement les biens d’une ville particulière
* **Fonctionnalité bonus** : Un propriétaire ne peut modifier que les caractéristiques de son bien sans avoir accès à l’édition des autres biens.

**Objectif :**Créer un microservice avec une API REST qui permet aux utilisateurs de réaliser toutes les fonctionnalités citées ci-dessus. Il n’est pas demandé de développer l’interface.  
  
**Choix technologiques :**

* Langage : Python
* Framework : Flask
* Base de données : Choix libre
* **Pas de Front**

**Livrable :**Transmettre le code sous la forme d’un repository git (sur github par exemple), le Readme contiendra toutes les instructions pour pouvoir le faire tourner en local.  
  
**Critères d’évaluation :**  
Méthodologie

* de gestion de projet, cela devra être reflété dans l’historique Git
* Modèle de données utilisé
* Structuration de l’API REST
* Structuration/ Architecture du code python