

# Compte rendu : weather and battery

## I- Définition de la mission :

Cette section contient deux blocs de code distincts. Le premier code a pour objectif d'ajouter les caractéristiques techniques des drones, telles que la température maximale supportée, le taux d'humidité maximal, ainsi que d'autres paramètres environnementaux et techniques importants pour la gestion de la mission. Ces informations sont stockées et associées à chaque drone, permettant de personnaliser leur utilisation en fonction de leurs spécifications et des conditions environnementales.

Le deuxième code est un script qui interagit directement avec les drones enregistrés grâce aux caractéristiques fournies par le premier code. Il vérifie les informations sur chaque drone pour déterminer s'il peut commencer une mission. Le script accède notamment aux données météorologiques en temps réel, telles que la température ambiante, l'humidité, ainsi que d'autres facteurs climatiques potentiellement critiques pour la sécurité du vol. En parallèle, il vérifie également le niveau de batterie de chaque drone pour s'assurer que celui-ci est suffisant pour démarrer la mission.

## II- Les entrées et la sortie de la mission :

### Script 1 :

**1- Les entrées :** Les entrées du premier script définissent les caractéristiques techniques et environnementales d'un drone, telles que la vitesse maximale du vent, la couverture nuageuse et les températures acceptables pour un vol sûr. Elles incluent aussi des paramètres comme l'humidité, le temps de vol maximal et la sensibilité à la pluie, afin d'assurer une performance optimale. Ces paramètres permettent de garantir que le drone fonctionne dans des conditions appropriées. Ils sont essentiels pour adapter son comportement aux conditions extérieures durant une mission.

**2- Les sorties :** La sortie du code fournit les caractéristiques essentielles de chaque modèle de drone, telles que la vitesse maximale du vent, la couverture nuageuse, et les plages de température pour garantir des vols sûrs. Elle inclut également l'humidité maximale et le temps de vol avant que la batterie ne soit épuisée. La sensibilité à la pluie est précisée, indiquant si le drone peut voler sous la pluie. Ces paramètres permettent d'évaluer si un drone peut débiter une mission en fonction des conditions météorologiques présentes.

### Script 2 :

**1- Les entrées :** Le type d'entrée représente les paramètres essentiels pour une mission de drone, permettant de définir son parcours, ses caractéristiques et sa communication. Il inclut généralement une série de coordonnées géographiques. L'entrée spécifie également le modèle du drone, ce qui permet d'adapter les paramètres de vol en fonction des capacités et des spécifications techniques du drone, comme sa vitesse maximale, sa durée de vol, et d'autres caractéristiques. La vitesse de vol est également précisée, indiquant à quelle vitesse le drone doit se déplacer pendant la mission. Enfin, l'adresse IP du drone est incluse pour permettre la connexion.

**2- Les sorties :** Cette sortie donne un aperçu de l'état actuel de la mission d'un drone. Elle précise si la mission présente des risques, le temps estimé pour la compléter et le temps restant avant que la batterie ne soit vide. La distance totale à parcourir et le niveau de batterie restant sont également indiqués, fournissant des informations essentielles sur la faisabilité du vol. Enfin, le statut de la mission résume si la batterie est suffisante pour terminer le vol ou si des ajustements sont nécessaires.

### III- Explication du code de la mission :

#### Script 1 :

Ce code permet d'ajouter de nouveaux drones à un fichier JSON. Il charge les données depuis `input_drone.json` et vérifie si un drone existe déjà dans `all_drones.json`. Si ce n'est pas le cas, un nouvel objet **Drone** est créé avec les informations fournies et ajouté à la base de données. Enfin, le fichier `all_drones.json` est mis à jour pour refléter les nouveaux ajouts. Ce processus évite les doublons tout en maintenant la liste des drones à jour.

#### Script 2 :

Ce code exécute une mission de drone en vérifiant les conditions météorologiques, la batterie et la faisabilité de la mission. Il charge les configurations depuis des fichiers JSON, choisit un drone, puis récupère les données météo en temps réel à partir de l'API OpenWeather. Le code évalue si les conditions météorologiques (vent, nuages, température, humidité, pluie) sont compatibles avec les capacités du drone. Ensuite, il calcule la distance totale et la durée estimée de la mission, tout en prenant en compte l'autonomie restante de la batterie. Si la mission dépasse l'autonomie, un avertissement est généré. Les résultats sont ensuite enregistrés dans un fichier JSON.

### IV- Explication des fonctions du script :

#### Script 1 :

Ce script permet d'ajouter de nouveaux drones dans un fichier JSON. La classe **Drone** définit les caractéristiques d'un drone et la méthode `to_dict` convertit ces données en dictionnaire. Le code lit d'abord les nouveaux drones depuis `input_drone.json`, puis vérifie si un fichier `all_drones.json` existe pour charger les anciens drones. Si un drone n'existe pas encore, il est ajouté. Enfin, tous les drones sont enregistrés dans `all_drones.json` avec les nouvelles données.

#### Script 2 :

Le script contient plusieurs fonctions et classes essentielles pour évaluer la faisabilité d'une mission drone. La fonction `get_battery_level` se connecte au drone via son IP pour récupérer le niveau de batterie et son état de santé grâce à la bibliothèque Olympe. La classe **Mission** permet de calculer la distance totale à parcourir à partir d'une liste de coordonnées GPS et d'estimer la durée de la mission en fonction de la vitesse du drone. La classe **Drone** définit les limites environnementales que le drone peut supporter, comme le vent, l'humidité ou la température. La classe **Meteo**, quant à elle, interagit avec l'API OpenWeather pour obtenir les conditions météo actuelles, vérifier la validité de la clé API et

déterminer si ces conditions permettent de lancer la mission en toute sécurité. Ensemble, ces éléments permettent d'estimer les risques de la mission selon la météo et l'autonomie du drone.