

Ressources Matériel

Matériel	Modèle choisi	Justification de nos choix
Micro-ordinateur (commun)	Raspberry Pi 3 Modèle B	<ul style="list-style-type: none"> Suffisant pour gérer Python et la 4G. Consomme moins d'énergie que le Pi 4 ou que le B+. Facilité de codage qu'on ne retrouve pas forcément dans le Raspberry Pi 4/5 +. Possède les ports GPIO nécessaires (UART, I2C).
Interface (commun)	GrovePi+	<ul style="list-style-type: none"> Rapidité de mise en œuvre. Compatible avec la plupart des Raspberry et de nos capteurs.
Stockage (commun)	Carte Micro SD 32Go	<ul style="list-style-type: none"> Capacité de stockage supérieure à 16 Go, indispensable pour héberger l'OS et conserver les codes Python localement.
Communication (commun)	Récepteur Radio (433MHz ou LoRa)	<ul style="list-style-type: none"> La fréquence 433MHz traverse mieux la végétation (forêt) que le Wi-Fi (2.4GHz) ou le Bluetooth.
Geo-localisation	module GPS Grove	<ul style="list-style-type: none"> indépendance du système : Permet de séparer la fonction de géolocalisation

(personnelle)	109020022	de la communication 4G (gérée par le module SIM7600 de l'étudiant Sohan). Cela évite les conflits sur les ports série du Raspberry Pi et assure le suivi GPS même en cas de panne du réseau mobile.
Météo (personnelle)	Grove - Temp&Humi Sensor (BME280)	<ul style="list-style-type: none"> Il est plus fiable que le DHT22 et me donne la pression atmosphérique en plus, ce qui est un bon indicateur pour le comportement des abeilles qui sont sensibles à la météo.

Ressources Logiciel

Logiciel / Bibliothèque	Version	Rôle dans le projet
Raspberry Pi OS Lite <i>(Système d'Exploitation)</i>	64-bit	Système d'exploitation du Raspberry Pi
Python (<i>Langage de programmation</i>)	3.9+	Langage pour le développement des scripts et la gestion des capteurs
grovepi (<i>Bibliothèque Python</i>)	Récente	Communication avec les capteurs Grove via l'interface GrovePi+
smbus2 (<i>Bibliothèque Python</i>)	Récente	Gère le protocole I2C. Indispensable pour extraire les données environnementales du capteur BME280.
pyserial (<i>Bibliothèque Python</i>)	Récente	Gestion de la communication série (UART). Utilisé pour envoyer les commandes AT au module SIM7600 et lire le récepteur radio.
Thonny / VS Code (<i>Outil de Développement</i>)	Récente	Environnements de développement utilisés pour l'écriture, et les tests des scripts Python.
Git (<i>Outil de Développement</i>)	Récente	Permet de conserver tout l'historique des modifications des fichiers et du code source.
PlantUML / Draw.io (<i>Outil de Modélisation</i>)	Récente	Outils de modélisation graphique utilisés pour créer nos différents diagrammes
MySQL / MariaDB (<i>Base de données</i>)	Récente	Modélisation et administration de la base de données
PHP (<i>Langage Serveur - Commun</i>)	Récente	Traitement des données côté serveur
LWS (serveur web)	Récente	Fournisseur d'hébergement

permettant de déployer notre infrastructure web (Base de données et API PHP) et d'assurer la disponibilité continue des données en ligne.