

Visual Positioning System for Augmented Reality

Ewelina RUPNIK
Mathieu BREDIF



Projet Commun
TSI

Vincent MIRAS
Ibrahim SALL
Loïs BILHERAN
Hugo GESLIN



Sommaire



Introduction



Organisation



VPS et LamAR



Format de données



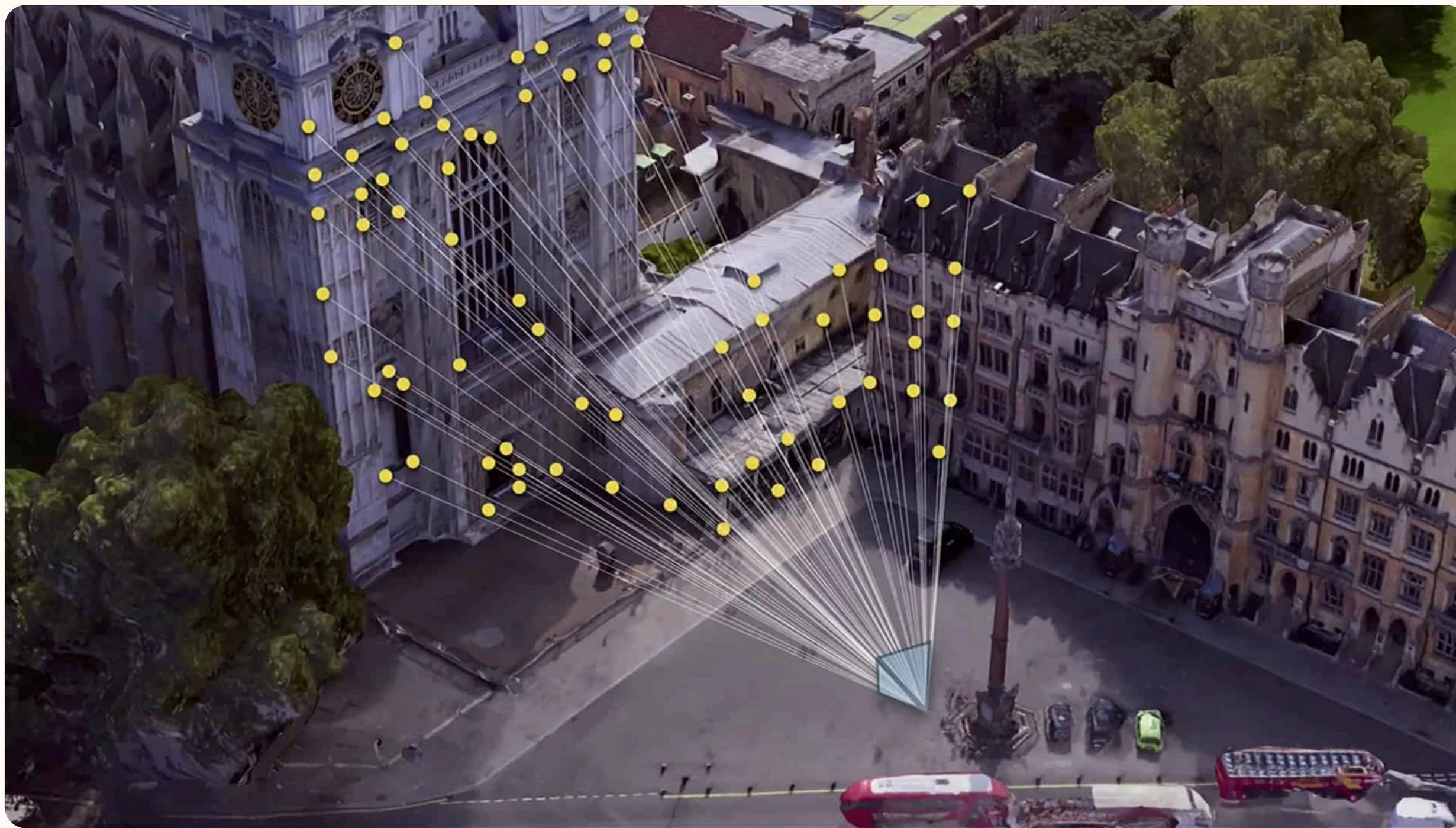
Implémentation



Résultats



Introduction

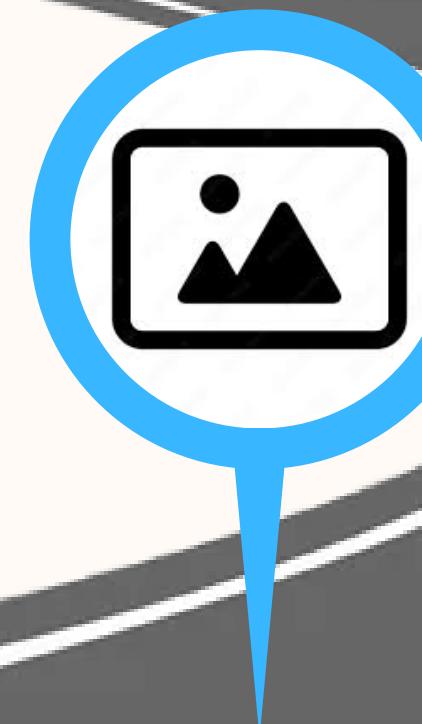


Objectifs de l'API

Renvoyer la position
géo référencée



Accepter une
image



Positionner l'image



Organisation

L'équipe



Loïs Bilheran
Développeur



Vincent Miras
Chef de projet



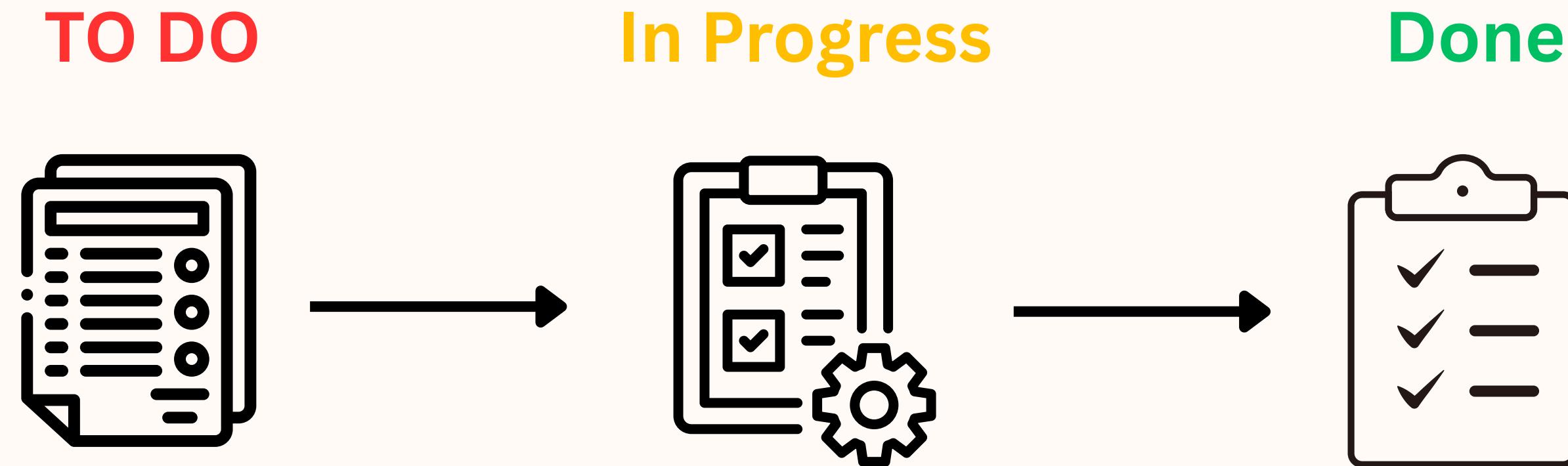
Ibrahim Sall
Scrum Master



Hugo Geslin
Développeur

Gestion de projet

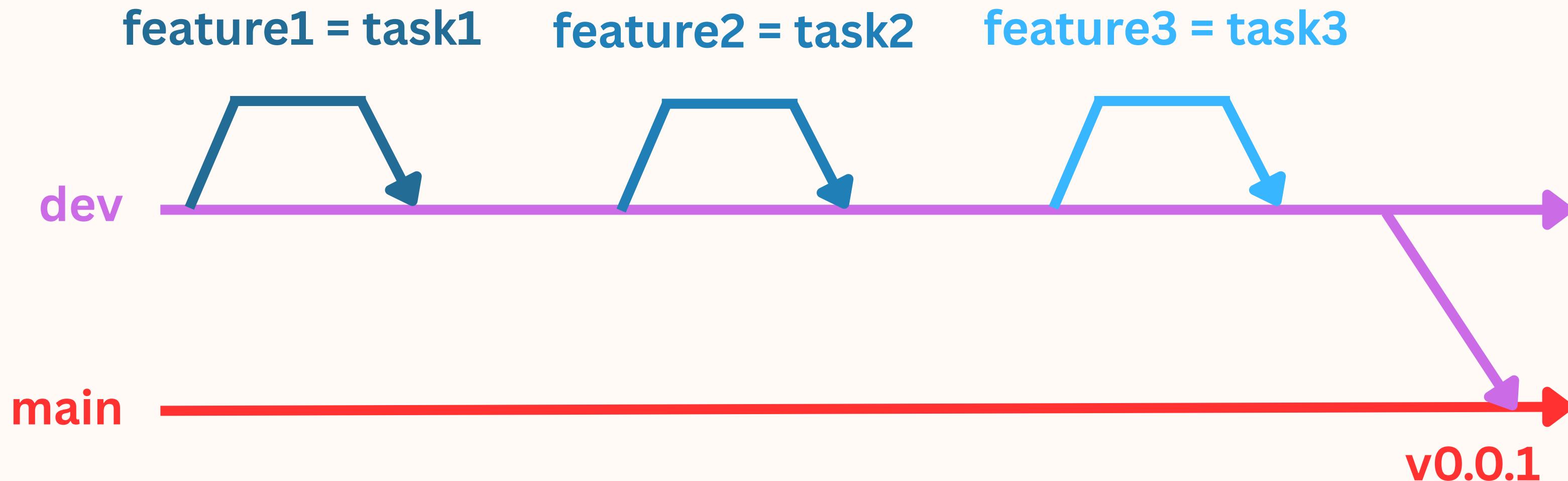
GitHub Backlog:



Sprint hebdomadaire

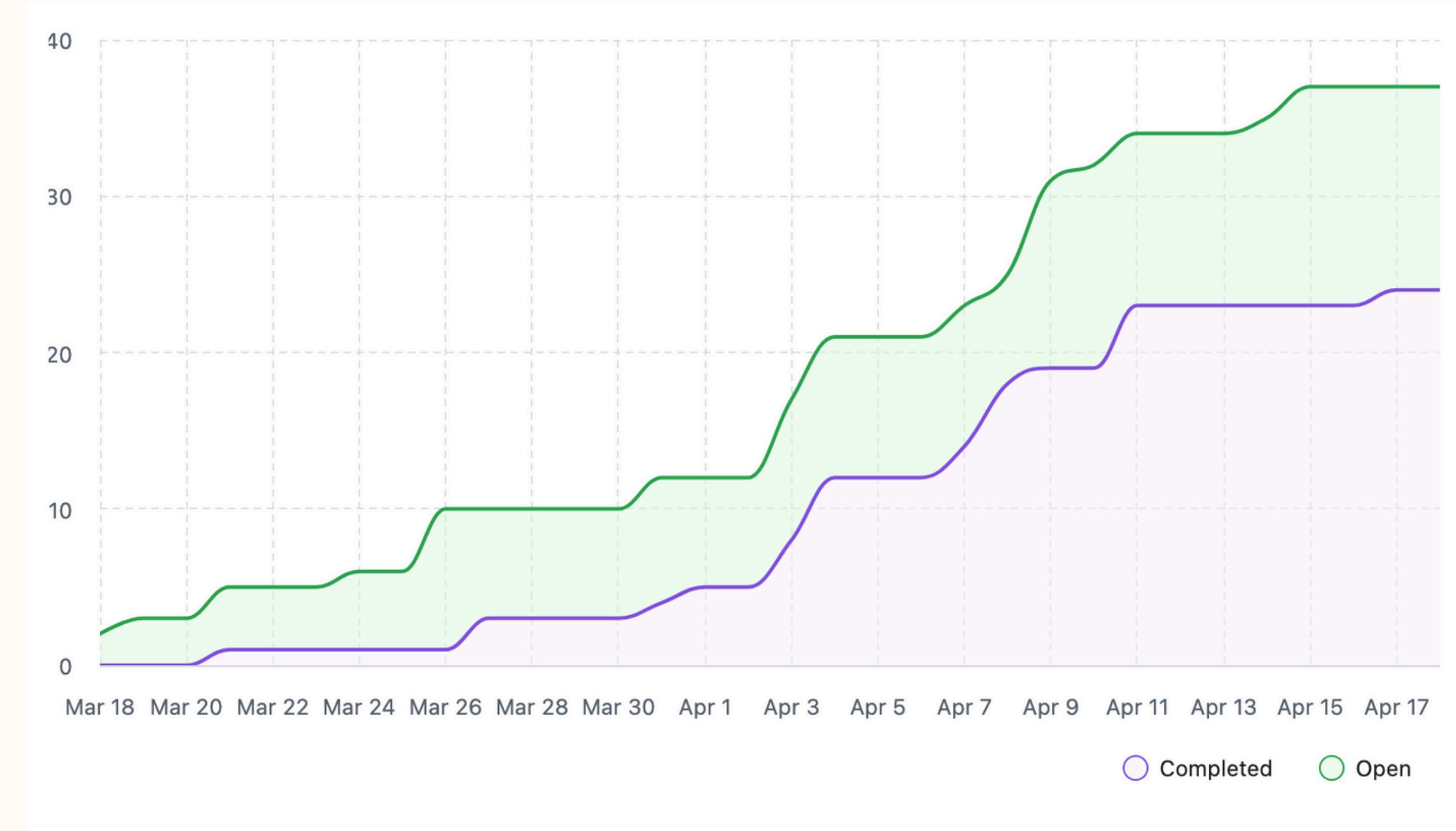


Gestion de projet



Arborescence Github

Gestion de projet



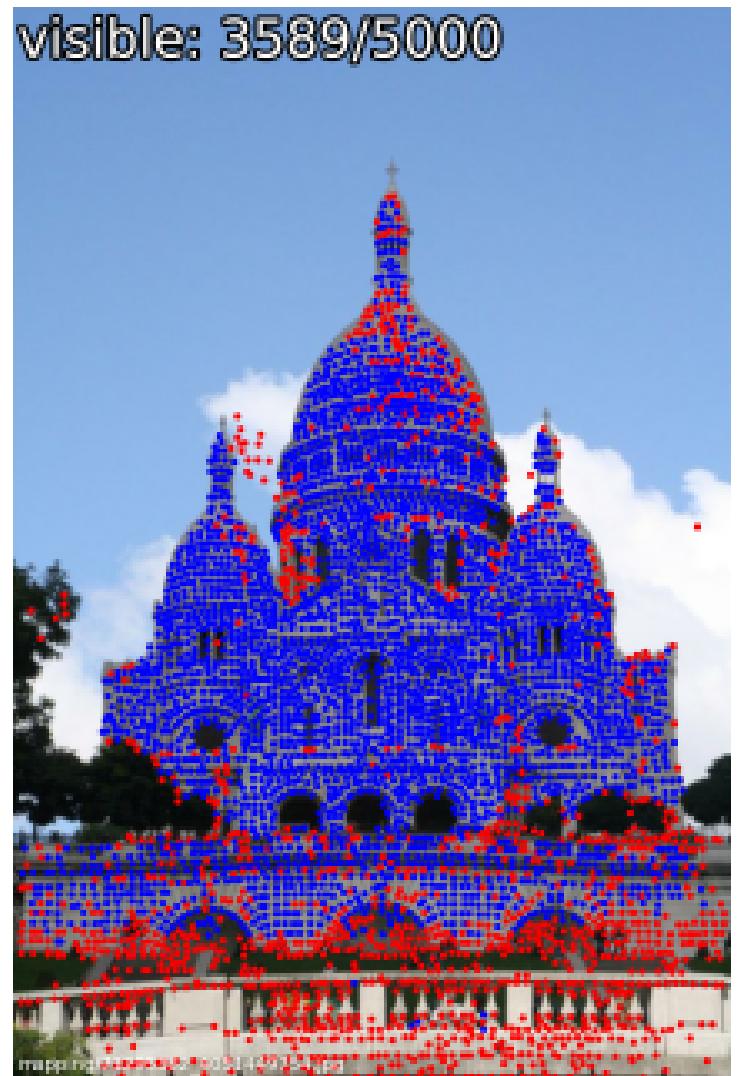
Avancement des tâches Github



LamAPI

VPS
&
LaMAR

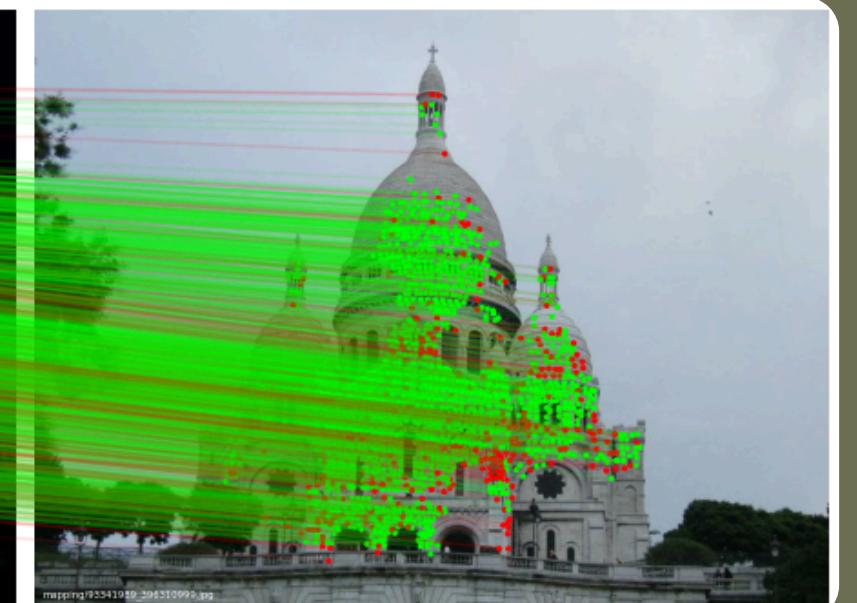
Qu'est ce qu'un VPS ?



Analyse de l'image

Visual Positioning System :

- Analyse d'une image
- Comparaison des points saillant avec un jeu de données
- Calcul de la position relative de la prise de l'image



Comparaison de points saillants entre images

LaMAR : un algorithme



Différentes utilités :

- Référencement des points saillants
- Unification des photos en un seul modèle
- VPS intégré au sein des calculs

Facile à utiliser :

- Travail existant et performant
- Arborescence et format de fichier déjà implémenté
- Possède un dataset associé
- Simple image docker à déployer

LaMAR : un dataset

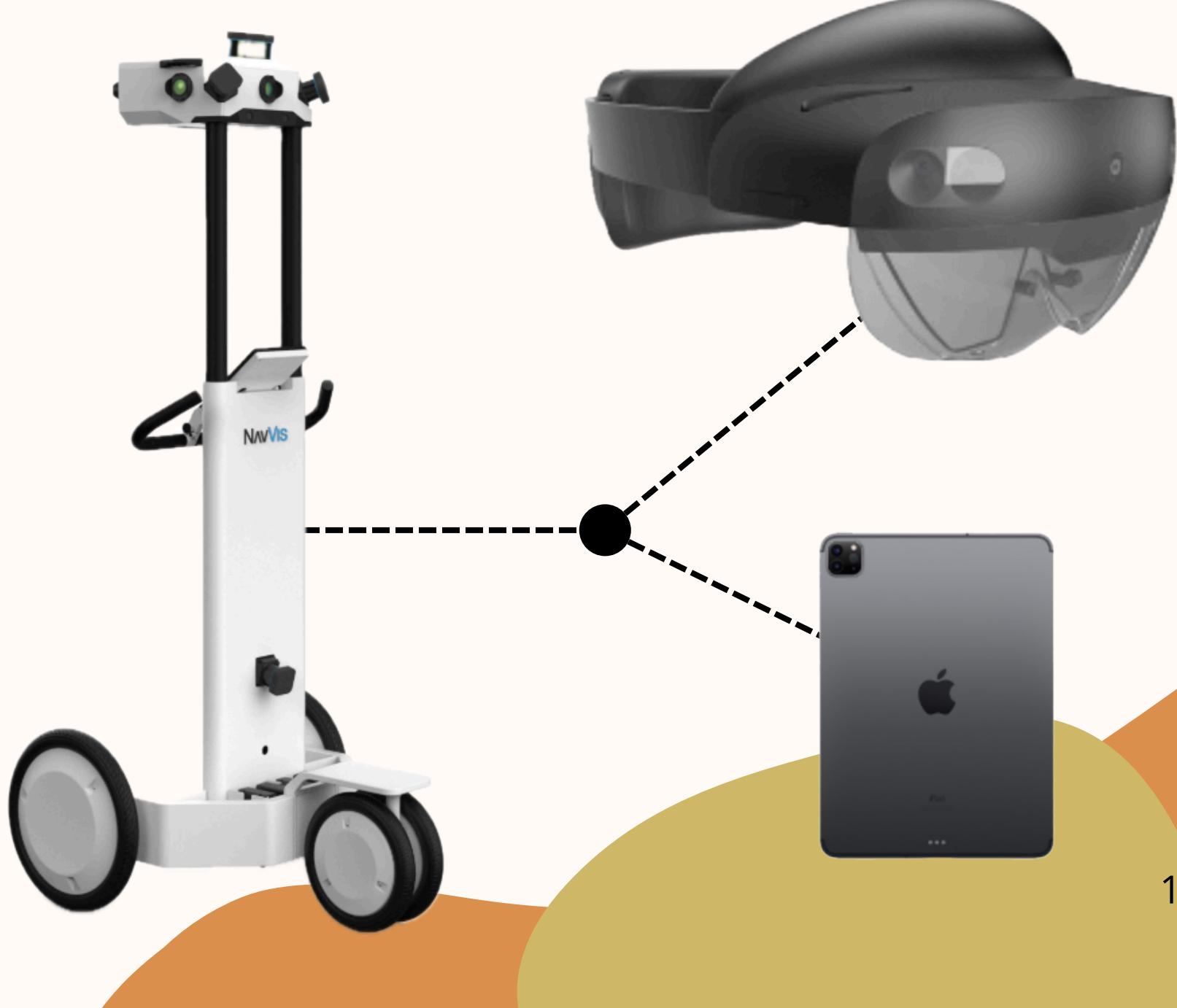


Avantages du dataset LaMAR :

- Données dans des conditions RA
- Plusieurs capteurs pour la prise de données
- Valable pour des lieux intérieurs et extérieurs
- Modifications temporelles effacées



Position renvoyée dans un repère local



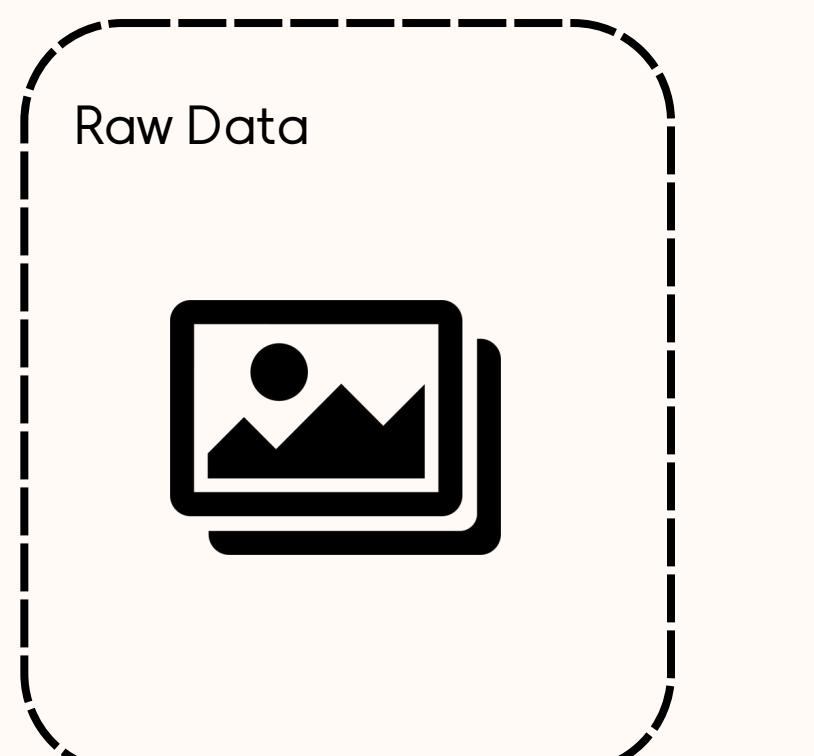


Modèle de données

Format de données: Capture

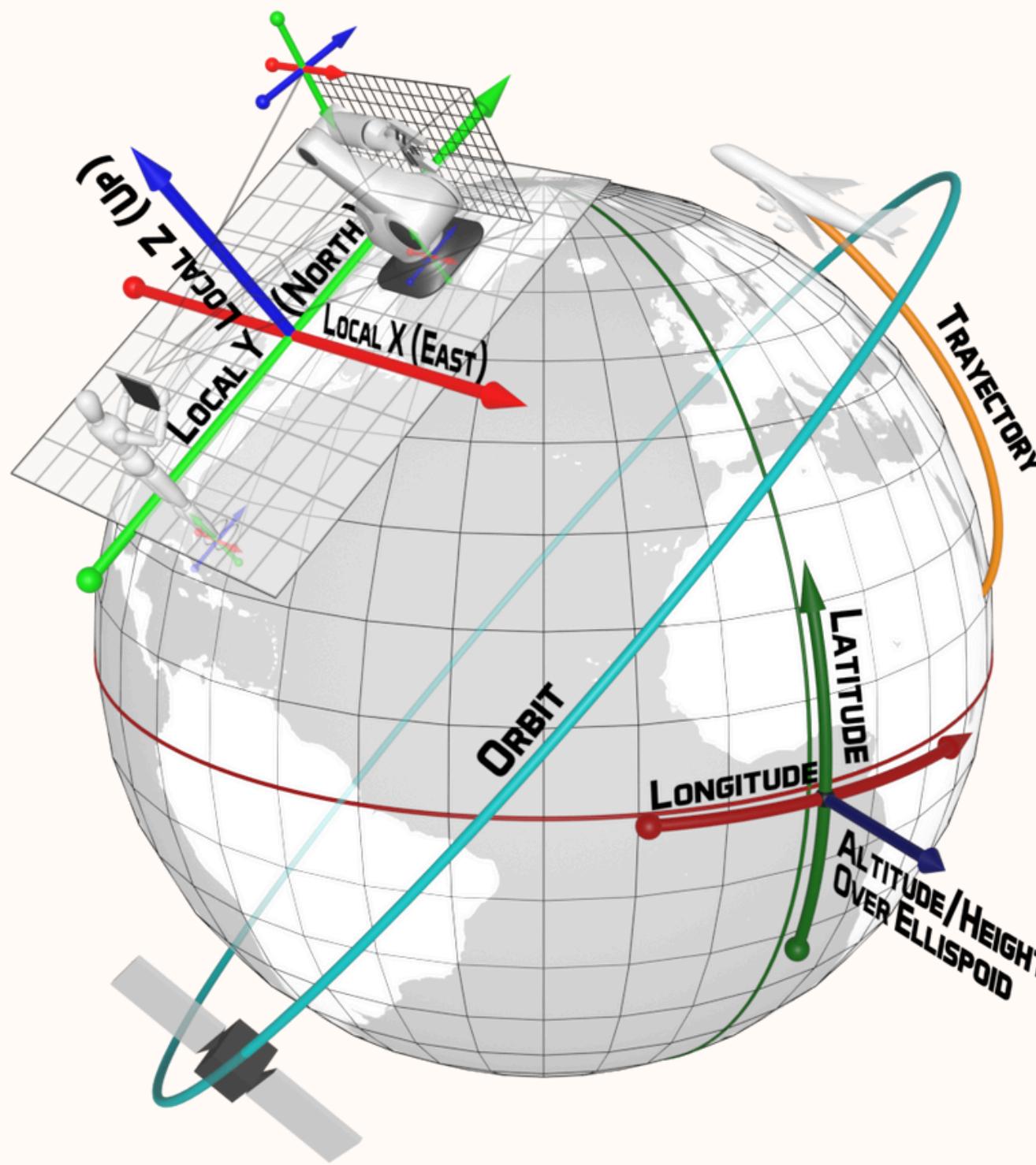
Phone Capture

- Sensors
- Images
- Trajectories



- Séparation en sessions
- Possibilité d'ajout de données complémentaires (bluetooth, wifi, profondeur...)
- Format simple et compréhensible

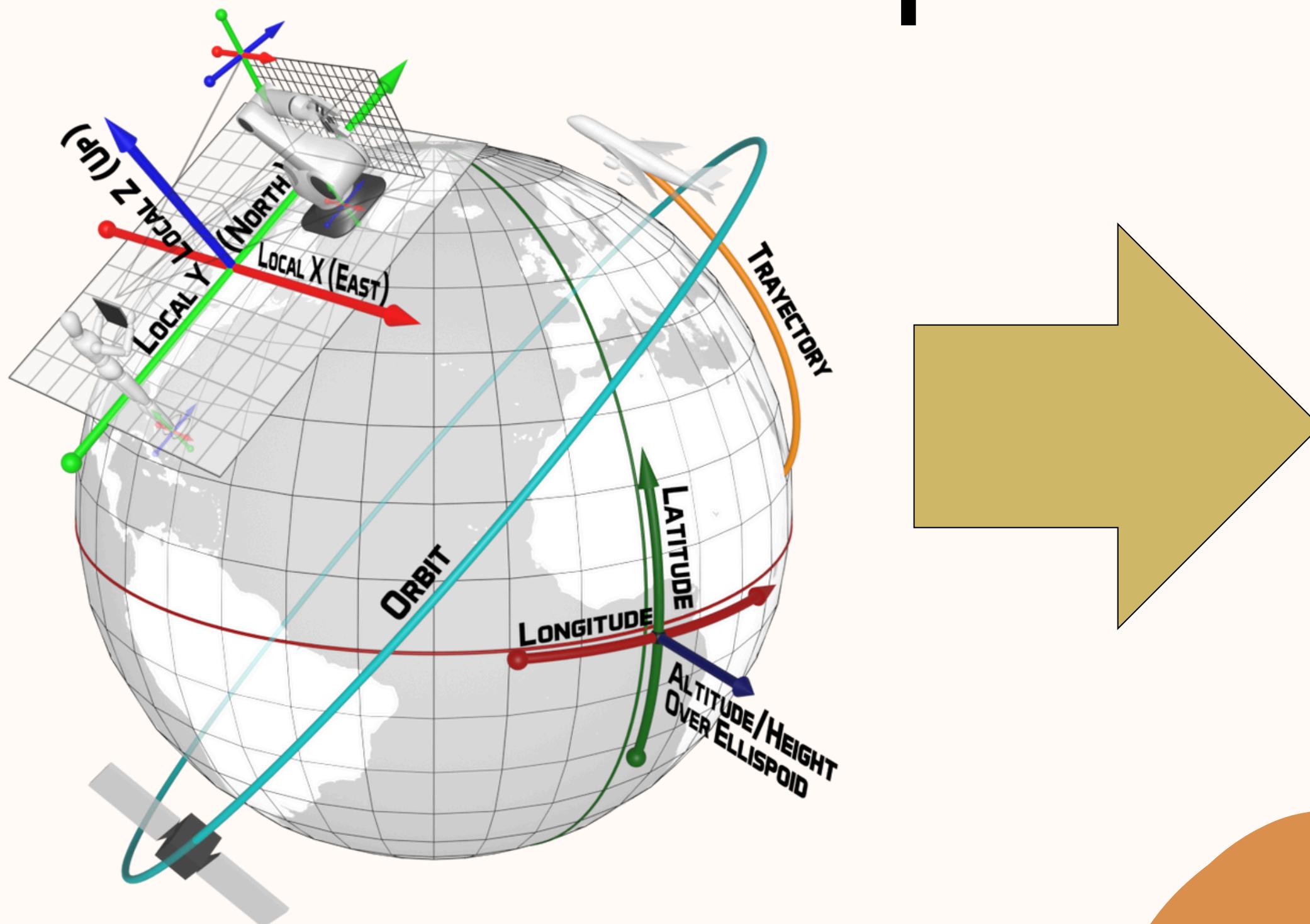
Échange de données : standard Geopose



Geopose :

- Vecteur d'échange pour la localisation et l'orientation d'objets géométriques à la surface de la Terre
- Standard OGC ce qui facilite l'interopérabilité.
- Plusieurs niveaux d'implémentation sont disponibles

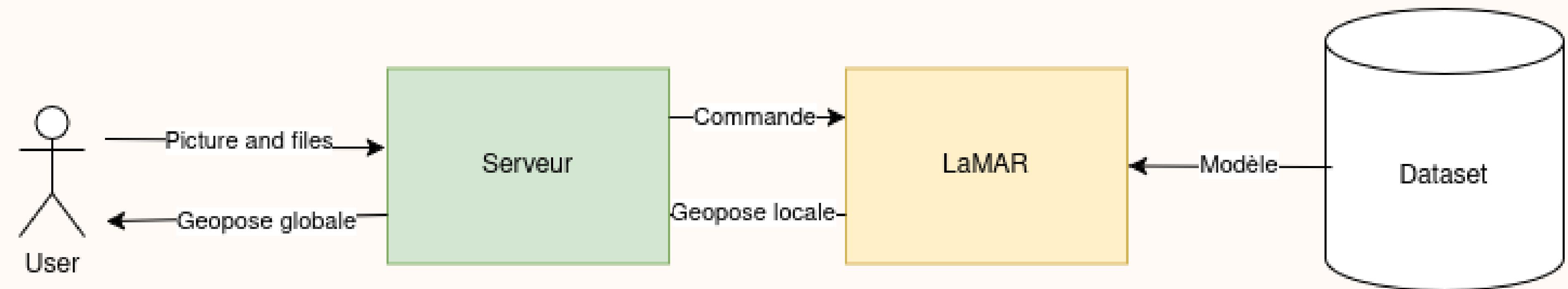
Échange de données : standard Geopose



```
"frameSpecification": {  
    "authority": "EPSG",  
    "id": "4979",  
    "parameters":  
        "longitude=-122.30&latitude=47.70  
        &height=11.0"  
},  
"quaternion": {  
    "x": 0.20056154657066608,  
    "y": -0.08111602541464237,  
    "z": 0.36606032744426537,  
    "w": -0.9050939692261301  
},  
"validTime": 1630560671227
```

Réalisation Technique

Architecture



Serveur Flask



Avantages :

- Serveur très simple à monter
- Avantages du traitement des données
- Flexibilité et possibilité de déploiement

Inconvénients :

- Moins performant pour du temps réel que d'autres frameworks
- Peu de composants intégrées par défaut
- Difficile à maintenir avec l'augmentation de fonctionnalités

Geopose protocol



OSCP-Geopose-protocol:

- Architecture client-serveur
- Convertit les données client en une Geopose
- Conforme au standard OGC

Architecture:

- Un client à usage unique envoie une requête contenant de quoi faire tourner un VPS
- Un serveur permet de traiter la demande et renvoyer la Geopose associée aux informations transmises

Geopose protocol.2

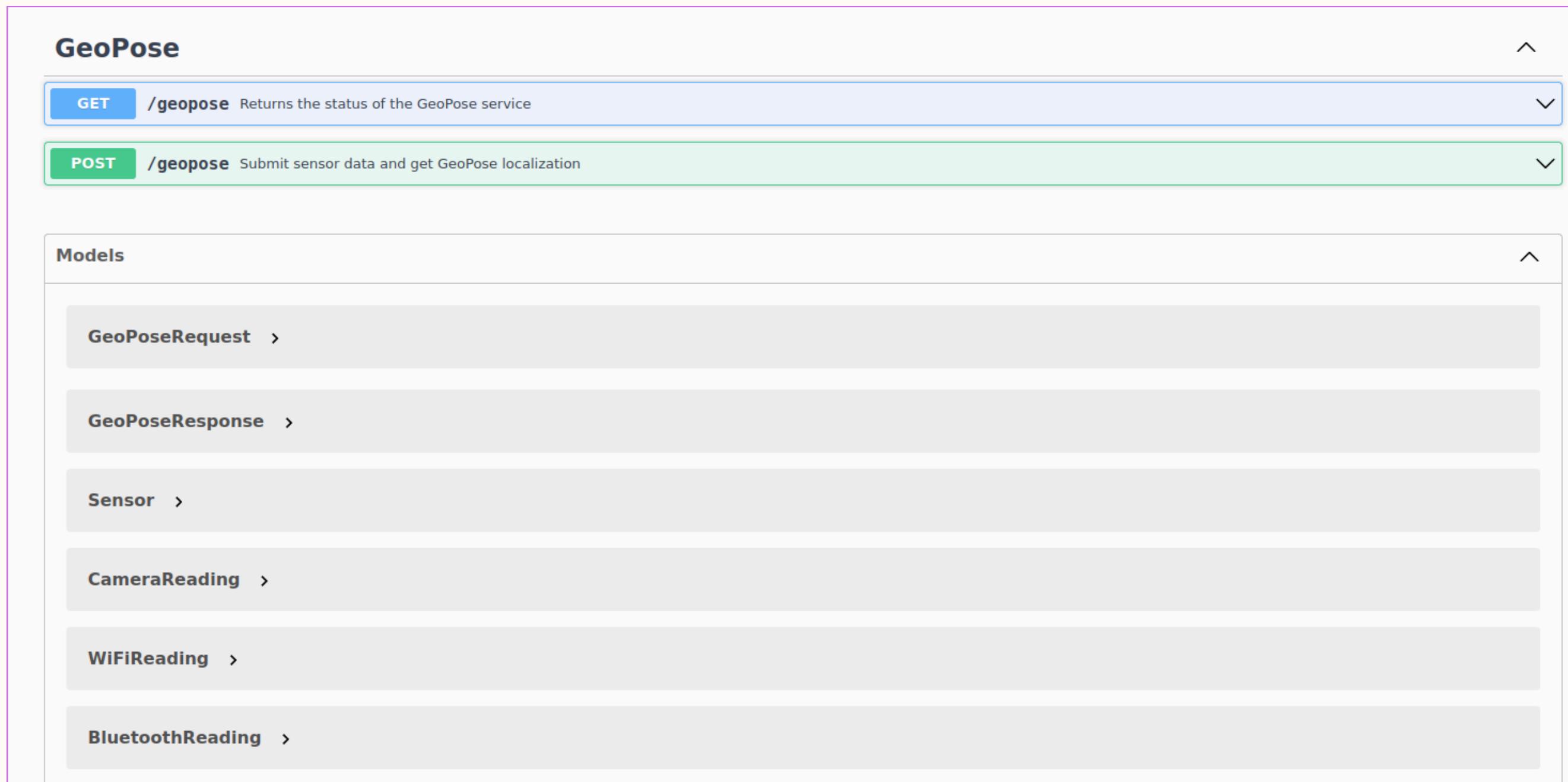
Compléments



Ce qui diffère au sein de notre API

- Usage d'un serveur flask unique
- Transit de la donnée via le format Capture
- Implémentation du VPS

Description des API avec Swagger (Open API)



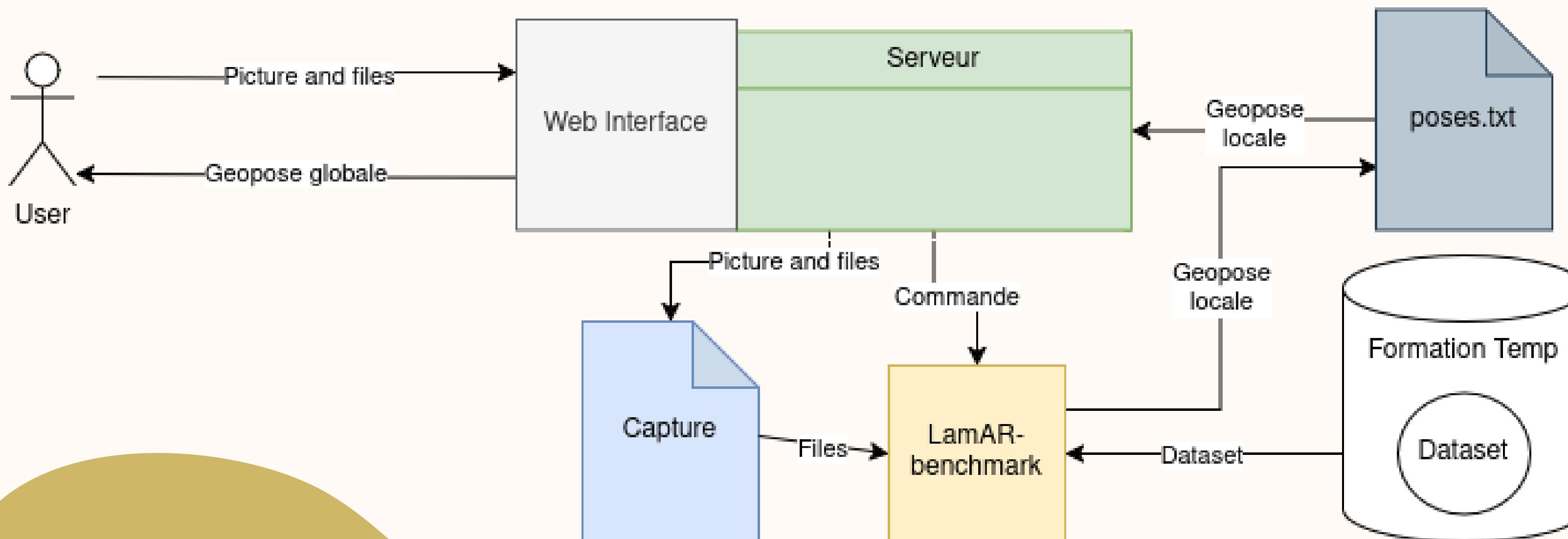
The screenshot shows the Swagger UI interface for the GeoPose API. The top section, titled "GeoPose", contains two operations: a blue "GET /geopose" operation which returns the status of the GeoPose service, and a green "POST /geopose" operation which submits sensor data and gets GeoPose localization. Below this, under the "Models" section, there is a list of data structures: "GeoPoseRequest", "GeoPoseResponse", "Sensor", "CameraReading", "WiFiReading", and "BluetoothReading". Each model name is followed by a small arrow icon indicating it can be expanded.

Test des API par utilisateur

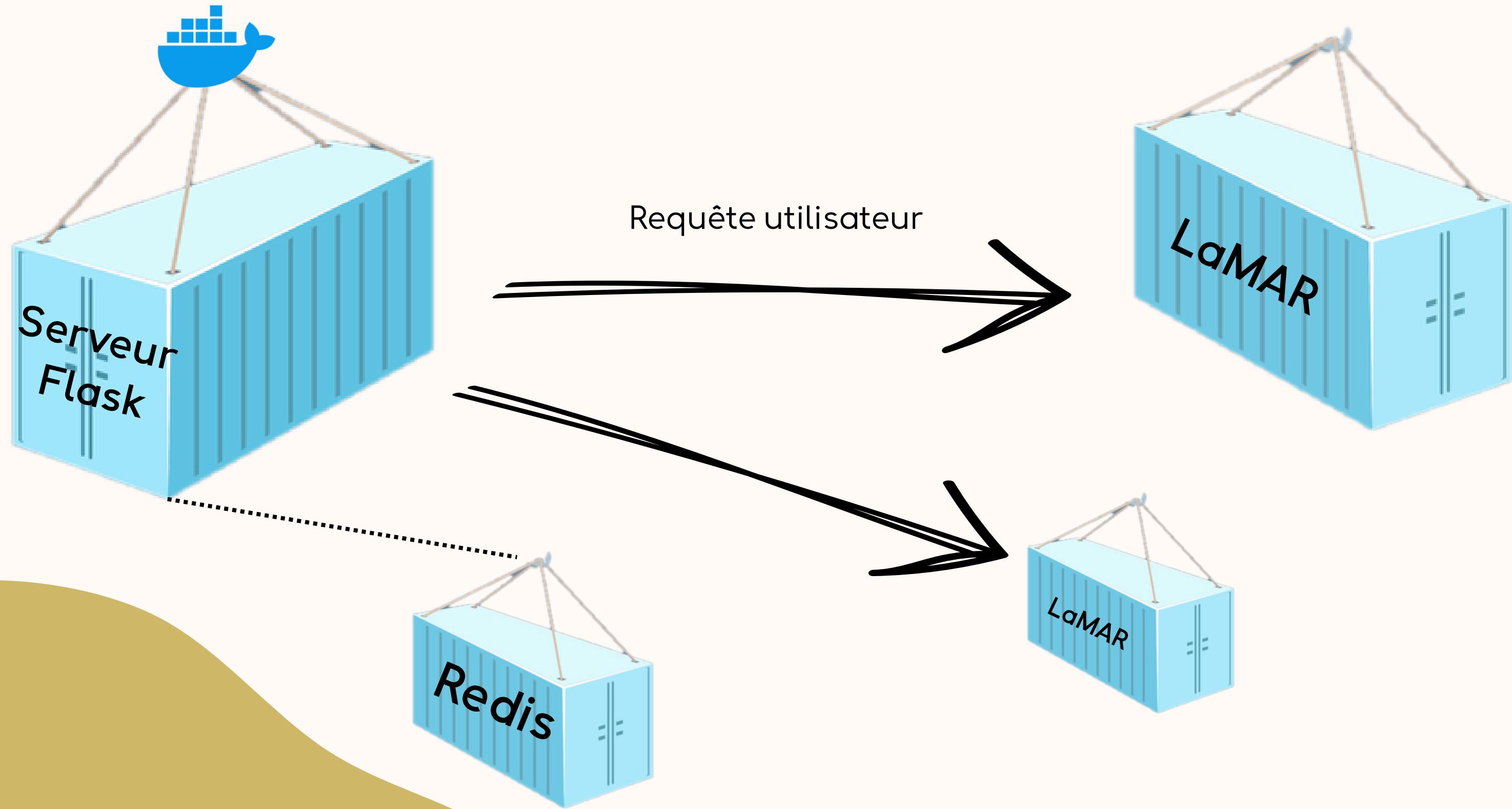
&

Modèle de données

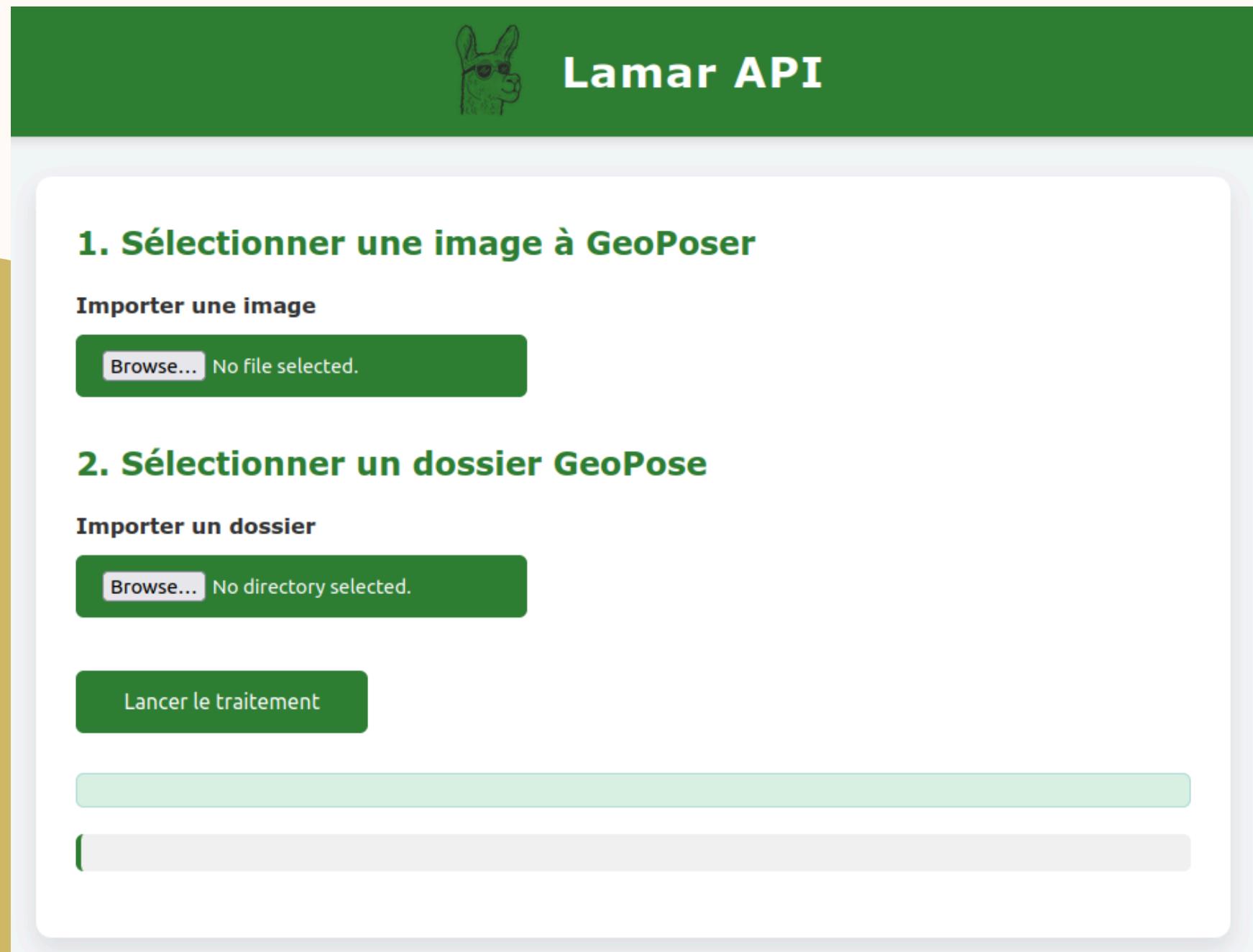
Architecture détaillée



Conteneurisation



Interface Web



The screenshot shows the Lamar API web interface. At the top, there is a green header bar with the Lamar API logo and the text "Lamar API". Below the header, there are two main sections:

- 1. Sélectionner une image à GeoPoser**
Importer une image
 No file selected.
- 2. Sélectionner un dossier GeoPose**
Importer un dossier
 No directory selected.

At the bottom of the interface, there is a button labeled "Lancer le traitement" and a progress bar consisting of two horizontal bars: a green one at the top and a grey one below it.

- Faciliter le requêtage de l'API
- Normalisation de la donnée en entrée
- Validation de la réponse

Geoposering API

Interface Web

1. Image choisie



Interface Web

1. Image choisie

2. Dossier choisi

Lamar API

1. Sélectionner une image à GeoPoser

Importer une image

Parcourir... 99889539500.jpg



2. Sélectionner un dossier GeoPose

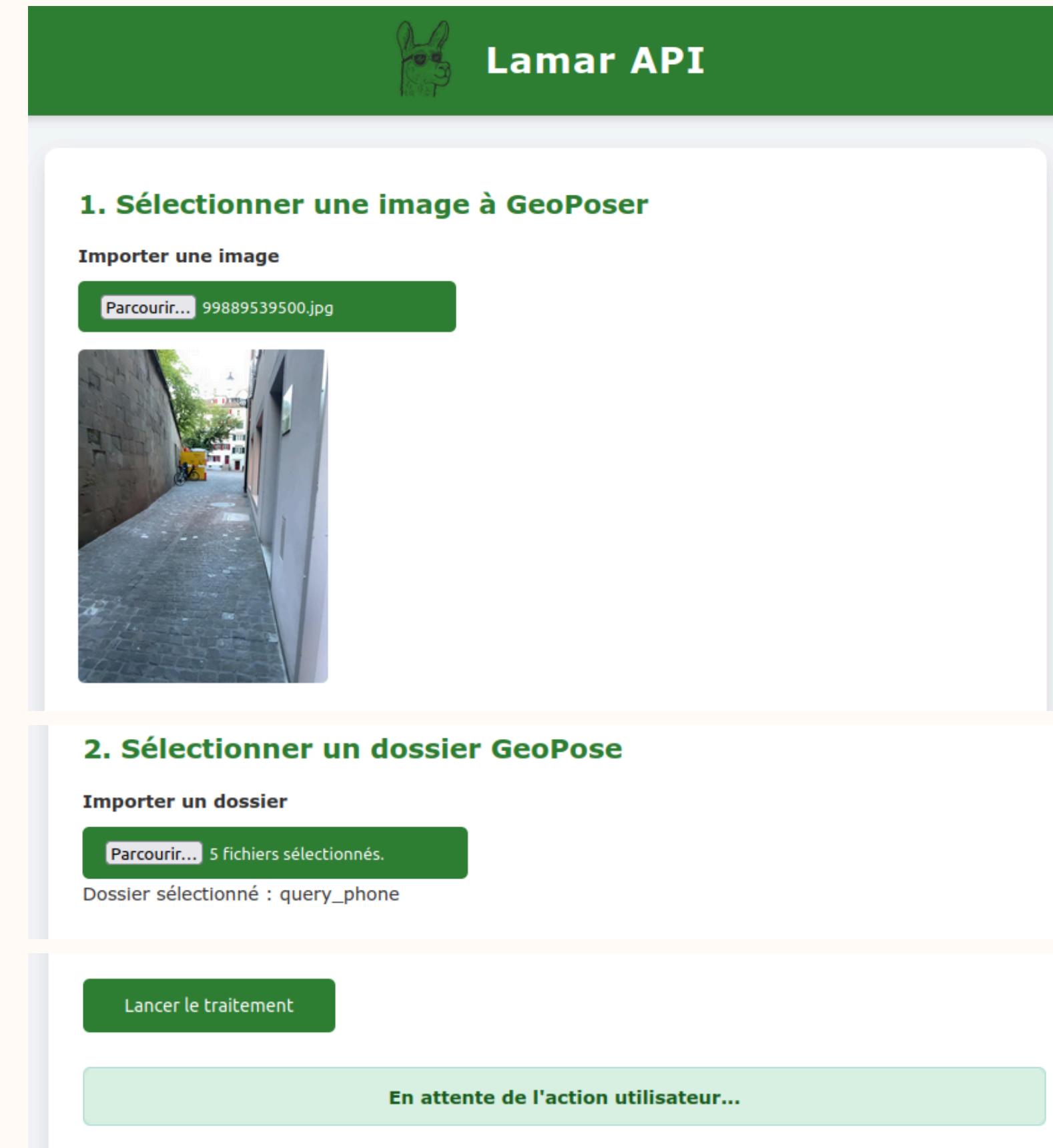
Importer un dossier

Parcourir... 5 fichiers sélectionnés.

Dossier sélectionné : query_phone

Interface Web

1. Image choisie
2. Dossier choisi
3. État du calcul



The screenshot shows the Lamar API web interface with three main sections:

- 1. Sélectionner une image à GeoPoser**
Importer une image
Parcourir... 99889539500.jpg

- 2. Sélectionner un dossier GeoPose**
Importer un dossier
Parcourir... 5 fichiers sélectionnés.
Dossier sélectionné : query_phone
- Lancer le traitement**
En attente de l'action utilisateur...

Geopose finale

Geopose de sortie :

- Transformation de coordonnées locales en coordonnées globales
- Respect de la norme Geopose grâce à l'utilisation de Geopose protocol
- Ajout d'un marqueur sur une carte Leaflet

Fonctions utiles à rajouter :

- Téléchargement du json avec un bouton
(Actuellement une fonction existe pour l'écriture auto mais n'est pas utilisée)
- Affichage du point en 3D dans le jeu de données LaMAR servant de comparaison
- Affichage de l'orientation sur la carte Leaflet
- Ajout d'une notion de précision pour la position donnée

Conclusion



- + Pipeline serveur opérationnel
- + GeoPose valide
- + Protocole OSCP GeoPose respecté

- LaMar prend trop de temps à répondre pour que la requête aboutisse
- Pas de partie bonus (jeu de données ENSG)

=> Grande expérience acquise et nombreuses débouchés pour LamAPI

Merci pour votre écoute !



Annexes

