

# Rapport 1 (04/10/2024)

## I. Fonctionnalités de la Plateforme

### a. Affichage du plan satellite et outils de sélection :

- Développer une interface pour visualiser des images satellites en haute résolution avec une vue de type "plan satellite".
- Ajouter des outils de sélection de régions spécifiques permettant l'interaction directe sur l'image, en utilisant des coordonnées GPS ou des polygones dessinables par l'utilisateur pour isoler des zones de l'oasis.

#### Proposition:

Nous étudions trois exemples d'outils et bibliothèques qui peuvent répondre à ces besoins :

Pour une version simplifiée utilisant Flask et Python, voici trois outils et bibliothèques légères qui permettront de visualiser et interagir avec des images satellites sans surcharger l'application :

#### 1. Flask avec un simple HTML Canvas pour la sélection

- **Description :** Utiliser un élément `<canvas>` HTML dans l'interface web pour afficher l'image satellite et permettre une sélection simple par dessin. Flask peut fournir l'image satellite à l'élément canvas.
- **Fonctionnalités de base :** Affichage de l'image sur le canvas. L'utilisateur peut cliquer et dessiner un rectangle ou une autre forme de base pour indiquer une région.
- **Outils de sélection :** Utilisation de JavaScript pour capturer les coordonnées des sélections faites sur le canvas et les envoyer au serveur Flask pour traitement.
- **Exemple d'utilisation :** Afficher une image satellite sur un canvas et permettre à l'utilisateur de sélectionner une zone en dessinant un rectangle. Flask pourrait alors utiliser ces coordonnées pour récupérer ou traiter la zone sélectionnée.

#### 2. Folium

- **Description :** Folium est une bibliothèque Python qui facilite la création de cartes interactives en intégrant des images satellites, tout en restant simple d'utilisation. Elle s'appuie sur Leaflet.js mais permet de manipuler les cartes directement avec Python, ce qui s'intègre parfaitement dans un projet Flask.
- **Fonctionnalités de base :** Affichage des cartes satellites et ajout de marqueurs simples.
- **Outils de sélection :** Même si Folium ne dispose pas de fonctions de dessin de polygones interactifs, vous pouvez ajouter des marqueurs pour indiquer des points d'intérêt. Des polygones peuvent être ajoutés en prédéfinissant des coordonnées, mais pour des sélections personnalisées, une extension plus tard serait nécessaire.
- **Exemple d'utilisation :** Intégrer Folium avec Flask pour afficher une carte satellite basique avec des points d'intérêt, accessible depuis une interface web minimale.

### 3. OpenCV pour Python

- **Description** : OpenCV est principalement une bibliothèque de traitement d'images, mais elle peut être utilisée pour charger, afficher et manipuler les images satellites au niveau pixel.

- **Fonctionnalités de base** : Chargement et affichage d'images satellites, manipulation basique comme le redimensionnement et l'ajustement des couleurs.

- **Outils de sélection** : OpenCV permet d'extraire des sous-régions d'images (via des coordonnées). Bien que l'outil soit plus orienté traitement d'image que géospatiale, il peut être utilisé pour des manipulations de base des images.

- **Exemple d'utilisation** : Charger une image satellite et permettre une sélection rectangulaire de zone via des coordonnées X/Y, le tout avec Flask pour servir l'image sur le web.

**Nous choisirons Flask car il offre une approche simplifiée, intégrée avec Python, idéale pour des projets de machine learning et de traitement d'images où l'on peut commencer avec des fonctionnalités basiques et étendre progressivement les capacités sans se heurter aux contraintes d'un outil plus spécialisé ou complexe.**

#### b. Intégration des modèles d'IA dans la plateforme

- **Framework et technologies**: Utiliser Flask pour intégrer le modèle IA sur la plateforme. Flask peut gérer les requêtes et fournir les prédictions sur les images importées.

- **Formats de modèles**: Préparer le modèle au format ONNX ou TensorFlow SavedModel pour une compatibilité optimale avec Flask.

- **Exécution locale**: Mettre en place une architecture de communication locale avec Flask en backend pour les calculs IA et Flutter en frontend pour l'interface utilisateur.

#### c. Affichage des résultats et KPI :

- **Définition des KPI** : Inclure des indicateurs tels que la surface (en km<sup>2</sup>) de l'oasis, le taux de changement annuel et la détection de zones critiques (par exemple, réduction de végétation).

- **Visualisation des statistiques par année**: Développer une interface qui affiche l'évolution de ces KPIs au fil des années, permettant de comparer les valeurs entre plusieurs périodes.

## II. Développement par phases

### Phase 1 : Importer et traiter les images

- **Importation des images**: Créer un bouton d'importation qui permet de charger 7 images satellites pour une même période. Les images doivent être transformées en matrices de couleur.

- **Fusion et affichage des images** : Fusionner les matrices des 7 images pour obtenir une vue composite de l'oasis, qui sera ensuite affichée dans l'interface.

### Phase 2 : Analyse statistique

- Statistiques et visualisation des évolutions : Analyser et afficher l'évolution de la surface de l'oasis par année. Intégrer des graphiques de tendance qui montrent l'impact du climat et d'autres facteurs au fil du temps.

### **III. Technologies et Outils**

a. Traitement d'images et matrices de couleur en Python :

- Utiliser des bibliothèques comme OpenCV et NumPy pour transformer les images en matrices de couleur, fusionner ces matrices et gérer les modifications d'image nécessaires pour l'affichage.

b. Développement et infrastructure de la plateforme :

- Backend : Flask pour la gestion des modèles et des requêtes utilisateur.
- Frontend : Flutter pour une interface graphique interactive permettant une communication avec Flask.
- Visualisation géospatiale : Intégrer des cartes géographiques en utilisant GeoServer, Mapbox ou Folium pour une meilleure gestion de la visualisation des images satellites et des statistiques géospatiales.

c. Bases de données et stockage

- Stocker les images et résultats dans une base de données géospatiale comme PostGIS, qui pourra gérer les informations des années successives et faciliter l'analyse des tendances.