

Verda AI

➤ URL du site (frontend) : <https://verda-ai.netlify.app>

➤ URL du backend (API FastAPI) : <https://verda-backend.onrender.com>

Endpoint principal : /identify

Le backend prend en charge :

- la réception de l'image
- le prétraitement
- l'inférence du modèle MobileNetV2 finement entraîné
- la génération de la structure JSON retournée au frontend

➤ **Description des moyens mis en œuvre :**

Architecture générale

Le projet est composé de deux parties :

❖ Frontend

Développé en HTML / CSS / JavaScript, hébergé sur **Netlify**.
Il permet :

- téléversement et prévisualisation d'image
- compression locale pour réduire l'empreinte carbone
- envoi de la requête POST vers l'API backend
- affichage des résultats :
 - ✓ nom de la plante
 - ✓ maladie détectée
 - ✓ taux de confiance
 - ✓ conseils écologiques (si disponibles)

❖ Backend

Développé en **Python (FastAPI)** et déployé sur **Render.com**.
Il inclut :

- ✓ un modèle MobileNetV2 finement entraîné
- ✓ un agent génératif optionnel (GenerativeVerdaAgent)
- ✓ un système CORS permettant au frontend d'appeler l'API
- ✓ un endpoint sécurisé pour l'envoi des images

❖ Modèle IA

Le modèle utilisé est :

- ✓ **MobileNetV2**
- ✓ Finement entraîné pour la classification de maladies végétales
- ✓ Format : fine_tuned_mobilenet.pth

Les performances sont optimisées pour fonctionner en CPU sur Render.

➤ Explication : Pourquoi les recommandations écologiques d'Ollama ne s'affichent-elles pas ?

- Ollama fonctionne localement, pas sur Render

Ollama est un moteur d'IA qui tourne **sur votre ordinateur**, pas dans le cloud.

Mais lorsque votre backend est déployé sur Render, il tourne dans un serveur cloud séparé ∴ il ne peut **pas accéder à Ollama installé sur votre PC**.

- Render ne permet pas l'exécution d'Ollama

Pour exécuter Ollama, il faut :

- Docker
- beaucoup de RAM
- accès GPU ou CPU local
- accès root

Render Free ne permet **aucune de ces opérations**.

Même sur les plans payants, Render empêche l'installation d'Ollama.

➤ Conclusion :

- ✓ Le diagnostic par MobileNetV2 fonctionne parfaitement en ligne
- ✓ L'affichage des maladies et taux de confiance fonctionne
- ✓ La génération de recommandations via Ollama **ne peut fonctionner que localement**, pas sur Render

Pour activer les recommandations en ligne, il faudrait:

- Utiliser un service LLM externe (Groq, OpenAI, Gemini).
- Créer des recommandations statiques intégrées dans votre code

