Rapport de Design du Système-Mini Capstone

1. Introduction

Ce projet vise à développer un système intelligent permettant à la fois la prédiction des maladies du sol et la recommandation des cultures les plus adaptées aux caractéristiques de ce dernier. Dans le contexte du Sahel, où les conditions climatiques sont souvent extrêmes et les ressources limitées, un tel outil pourrait offrir une assistance précieuse aux agriculteurs dans la gestion de leurs terres et l'amélioration de leur rendement.

2. Architecture du Système

2.1 Vue d'ensemble

Le système se compose de deux modules principaux :

- Prédiction des maladies du sol : Ce module prend en entrée des données sur les propriétés du sol (pH, humidité, nutriments, etc.) et prédit les maladies susceptibles d'affecter ce sol.

- Recommandation de cultures : En se basant sur les caractéristiques du sol et les conditions climatiques, ce module propose les cultures les mieux adaptées.

Ces modules interagissent avec une base de données relationnelle contenant des informations sur les types de sols, les maladies, les cultures, et les analyses chimiques. Les utilisateurs saisissent les données via une interface, et reçoivent les résultats générés par les modèles d'intelligence artificielle.

2.2 Technologies utilisées

- Base de données : MySQL

- Backend : Python (Flask)

- Machine Learning : Scikit-learn, Pandas, NumPy

- Interface utilisateur : React\_native

- Stockage des fichiers : CSV pour les jeux de données internes

3. Conception de la Base de Données

3.1 Modèle de données

Les principales tables sont :

- soil\_type : Contient les différents types de sols (sableux, limoneux, etc.).

- soil\_analysis : Contient les résultats des analyses de sol (pH, humidité, nutriments).

- soil\_diseases : Contient les maladies des sols et les symptômes associés.

- crop\_recommendation : Contient les données utilisées pour recommander les cultures adaptées à un sol donné.

3.2 Choix des types de données

- Les colonnes numériques utilisent le type FLOAT (pH, humidité, température, etc.).

- Les textes sont stockés en TEXT (noms de maladies, types de sol).

- Les identifiants sont en INTEGER PRIMARY KEY avec AUTO\_INCREMENT.

4. Modèle de Prédiction et Recommandation

4.1 Prédiction des Maladies du Sol

Le modèle de prédiction est basé sur un algorithme supervisé (Random Forest ou Logistic Regression). Les variables indépendantes sont les propriétés physico-chimiques du sol, et la variable cible est la maladie encodée. Les données sont nettoyées, normalisées et le modèle est entraîné sur un ensemble de données annoté.

4.2 Recommandation de Cultures

Ce modèle prend comme entrée les propriétés du sol (pH, azote, potassium, humidité, etc.) et recommande une culture adaptée en sortie. Le modèle est entraîné sur un jeu de données synthétique basé sur des recherches agronomiques spécifiques au Sahel. Le type de modèle utilisé est le k-Nearest Neighbors (k-NN) ou Random Forest.

5. Flux de Données

- L'utilisateur saisit les données via une interface.

- Les données sont stockées dans la base.

- Le modèle les récupère pour prédire une maladie ou recommander une culture.

- Le résultat est retourné à l'utilisateur.

6. Sécurité et Intégrité des Données

6.1 Sécurisation des données

- Utilisation de connexions sécurisées (HTTPS)

- Authentification des utilisateurs

- Séparation des rôles (admin, utilisateur)

6.2 Intégrité des données

- Clés étrangères entre les tables pour assurer la cohérence

- Validation côté serveur des entrées utilisateurs

- Contraintes de type sur les colonnes (ex. : pH entre 0 et 14)

7. Tests de Conception

- Tests unitaires : vérification des fonctions de transformation des données.

- Tests d'intégration : simulation de l'entrée utilisateur jusqu'à la sortie du modèle.

- Tests de performance: rapidité des requêtes SQL et réponse du modèle.

8. Conclusion

La conception du système repose sur une architecture modulaire et des bases solides en intelligence artificielle et en gestion de données. Ce design permettra une évolution fluide vers l’implémentation et le déploiement. Les prochaines étapes consisteront à développer les modèles de machine learning, les intégrer à l’interface utilisateur et tester le système sur des cas réels.

Annexes

- Diagramme ER des tables de la base de données

- Exemple de jeu de données (CSV)

- Captures d'écran de l'interface utilisateur (à venir)