FACULTY OF APPLIED FUNDAMENTAL SCIENCES

***Scientia et Lux***

B.P. 155

Ruhengeri

Rwanda

**T**

: +250

30 30

788 90

: +250

30

32

788 90

**W**

:

[w](http://www.ines.rw/)

[www.ines.ac.r](http://www.ines.rw/)

**E**

:

inesruhengeri@yahoo.fr

**INSTITUT D’ENSEIGNEMENT SUPÉRIEUR**

**DE RUHENGERI**

***Accrédité par***

***Arrêté Ministériel N° 005/2010/Mineduc***

***du***

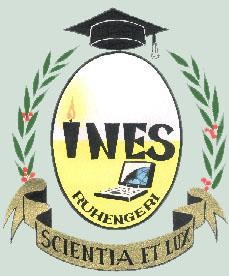
***Ju***

***16***

***in***

***2010***

***Scientia et Lux***



**GROUP 4 ASSIGNMENT**

**Musanze,**

**Novemberer**

**2024**

DEPARTMENT OF COMPUTER SCIENCE

OPTION OF SOFTWARE ENGINEERING

Artificial Inteligence

­­­

Year III SWE Group B

Members

IRUTABYOSE Yoramu 23/22764

MASONGA SHEMA Prince 23/20861

ISHIMWE Kevin 23/22823

IZABAYO HARANIRA Jean Luc Severin 23/22493

ISHIMWE HABYARIMANA Regis 23/19890

ISHIMWE HITAYEZU Herve 23/22202

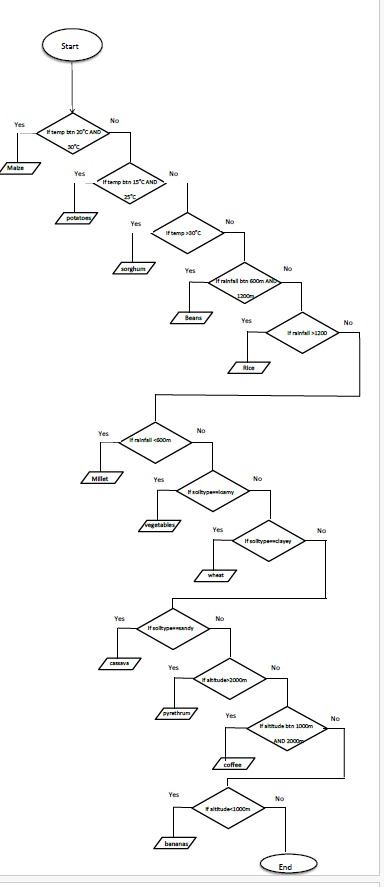
**Day 2 Report**

**Group #4**

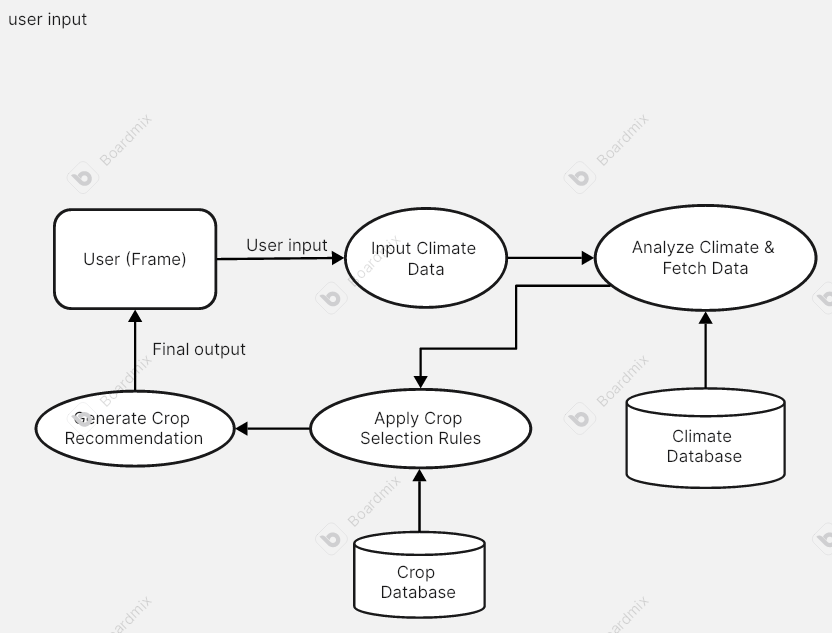
**System Design & Initial Python Development**

**1. System Design**

* **Flowchart:** A detailed flowchart was created to represent the logical flow of the crop recommendation system the figure below.



* **Data Flow Diagram (DFD):** A structured DFD was designed to illustrate the data movement between different system components as showed in the figure below

.

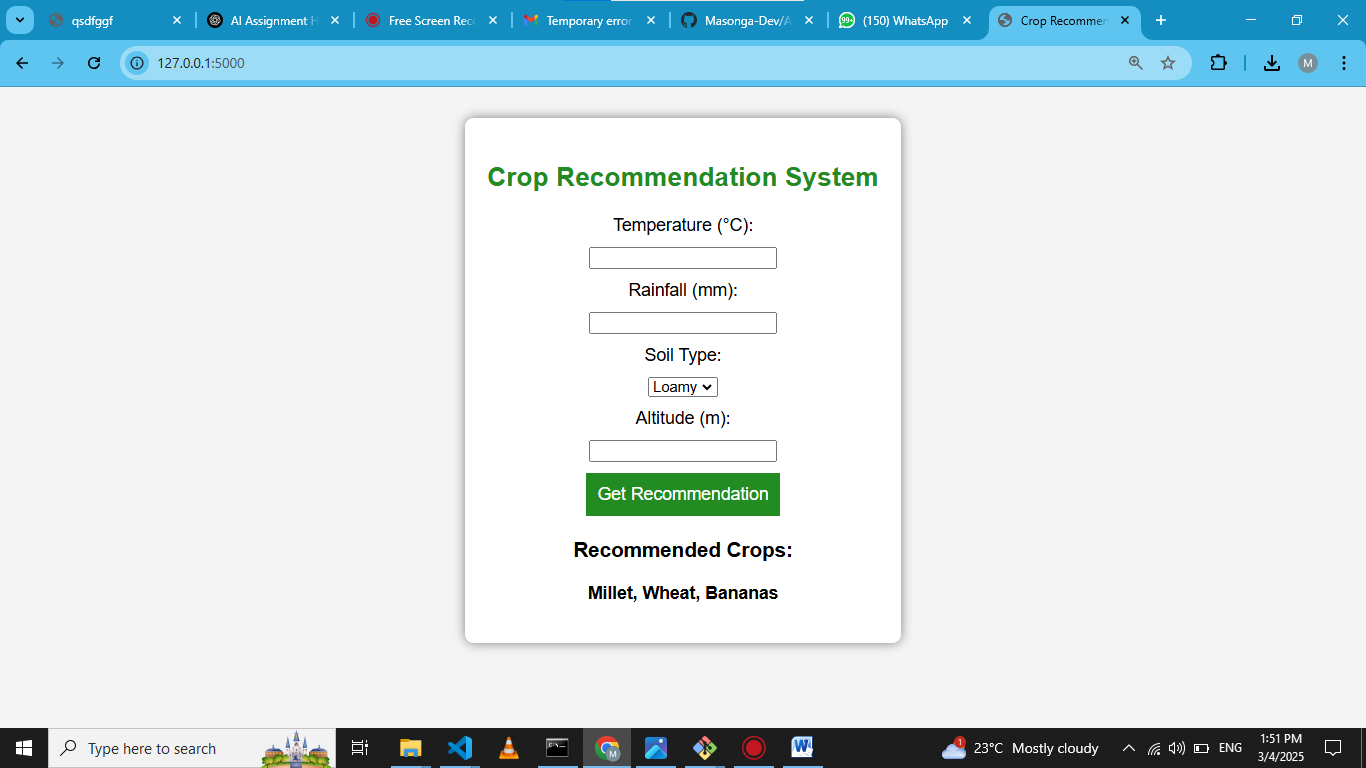
* **Architecture:** The system follows a web-based architecture, with a Flask backend handling requests and rendering recommendations based on user inputs.

**2. Initial Python Development**

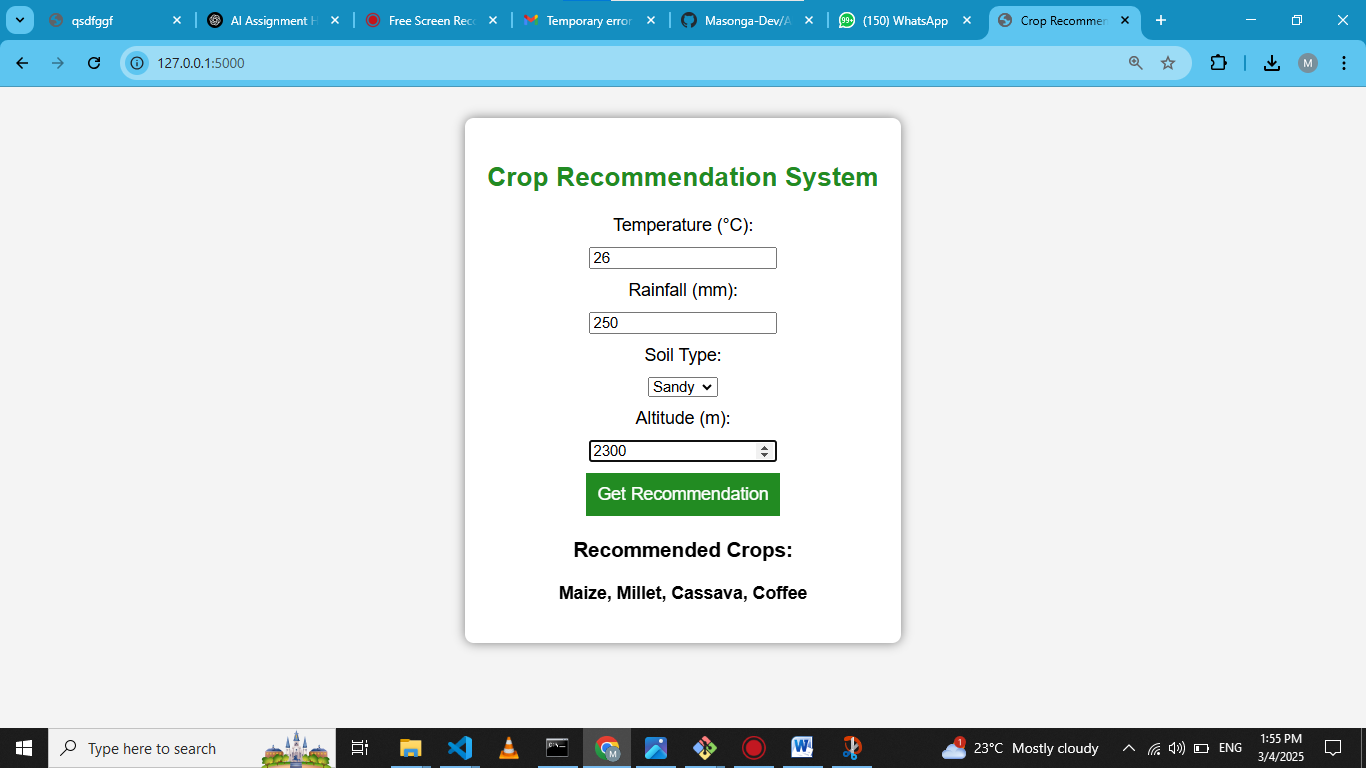
* **Technology Stack:**
  + Backend: Flask (Python)
  + Frontend: HTML, CSS
  + Data Processing: Rule-based logic for crop recommendations
* **Rule-Based Logic Implemented:**
  + **Temperature:** Determines suitable crops based on temperature range.
  + **Rainfall:** Matches crops with required rainfall levels.
  + **Soil Type:** Suggests crops based on soil properties.
  + **Altitude:** Identifies crops that grow well at given altitudes.
* **Flask Web App Development:**
  + Implemented a Flask-based web interface for users to input climate data.
  + Developed a recommendation engine that suggests crops based on user inputs.
  + Integrated a user-friendly HTML form for data submission.

**3. Testing & Documentation**

* **Testing:**
  + The Python logic was tested with various inputs to verify crop recommendations.



* + The Flask web app was deployed on Binder and tested for proper functionality.



* **Issues Encountered & Fixes:**
  + **Binder 404 Error:** Fixed by ensuring Flask runs on 0.0.0.0 and binds to the correct port.
  + **Port Conflicts:** Resolved by changing Flask to run on port 5000 and updating the Binder proxy link.
  + **Missing index.html:** Ensured index.html was present in the templates/ folder to avoid rendering errors.

**4. Next Steps**

* Enhance the user interface for better usability.
* Integrate additional rule-based logic for more precise recommendations.
* Prepare for the next phase of development, including advanced AI integration.

**GitHub & Deployment Links:**

* **GitHub Repository:** [Masonga-Dev/AI\_Group4\_ExpertSystem\_Assignment2](https://github.com/Masonga-Dev/AI_Group4_ExpertSystem_Assignment2)
* **Binder Web App:** [![Binder]([https://mybinder.org/badge\_logo.svg)](https://mybinder.org/v2/gh/Masonga-Dev/AI\_Group4\_ExpertSystem\_Assignment2.git/main)](https://mybinder.org/badge_logo.svg)%5d(https://mybinder.org/v2/gh/Masonga-Dev/AI_Group4_ExpertSystem_Assignment2.git/main))