WEB X CA

PREREQUISITES - TRACKIFY

Name of student	Laksh Sodhai
Class_Roll no	D15A_57
D.O.P	20/03/25
D.O.S	27/03/25
Sign and Grade	

Title : Flower Prediction

Introduction:

Flower Prediction is a machine learning-based web application that classifies flowers into species based on petal and sepal features using popular datasets like the Iris dataset. Built using Flask for the backend and optionally a simple HTML/CSS/JavaScript frontend, the system utilizes a trained ML model to provide accurate real-time predictions. Users can input features, and the system will display the predicted flower species with high accuracy.

System Requirements

1. Hardware Requirements:

- **Processor:** Intel Core i5 / AMD Ryzen 5 or higher (dual-core, 2.0 GHz or faster)
- **RAM:** Minimum 8GB (16GB recommended)
- **Storage:** At least 1GB free space (256GB SSD recommended)
- **Network:** Stable internet connection (especially for MongoDB Atlas users)

2. Software Requirements:

- **Operating System:** Windows 10/11, macOS 10.15+, or Ubuntu 20.04+
- **Code Editor:** Visual Studio Code or compatible IDE
- **Version Control:** Git 2.25+
- Python: Version 3.8 or higher

Technology Stack

Layer	**Technology**
Frontend	HTML/CSS/Typescript(or Streamlit/Flask Templates)
Backend	Flask (Python 3.8+)
ML Model	Scikit-learn
Styling	SCSS / Bootstrap (optional)
APIs	RESTful Flask APIs

Setup Instructions :

Backend Steup:

First Navigate to backend folder:

- Cd project

  Python & Flask:

  1. Install Python 3.8+ from https://www.python.org/downloads/. Ensure 'Add to PATH' is checked during installation.

  2. Create Virtual Environment:

python -m venv venv

venv\Scripts\activate # For Windows

source venv/bin/activate # For macOS/Linux

3. Install Dependencies:

pip install -r requirements.txt

4. Run Flask App:

python app.py

The app will run at: http://localhost:5000

Frontend Setup

1. Navigate to frontend folder:

cd project

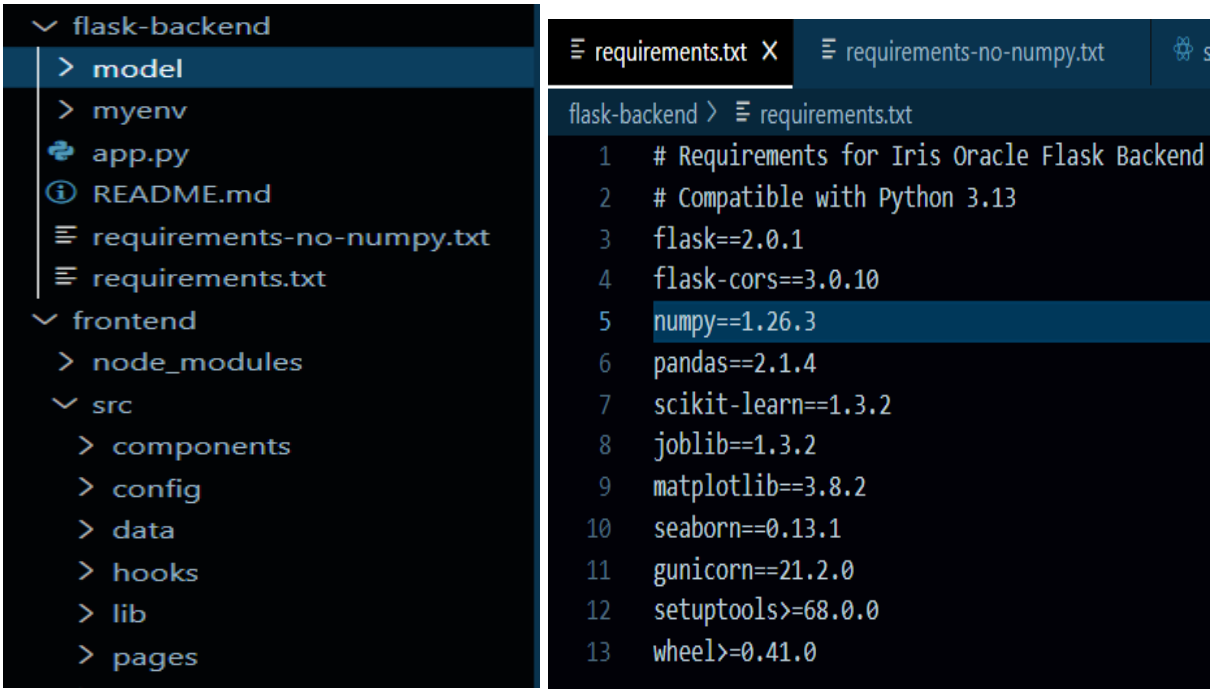
2. Install dependencies:

npm install

3. npm run dev

Frontend will run at: http://localhost:8081

Project Structure



Features Overview

- Input Form: Users enter flower features (petal/sepal length and width)
- Prediction: Model predicts flower species (e.g., Setosa, Versicolor, Virginica)
- Responsive UI: Clean and user-friendly interface
- Model Integration: Seamless ML model usage with Flask

Conclusion:

By completing the above installations and setup steps, your system will be fully ready to run the Flower Prediction application. Accurate installation of Python, Flask, and the ML dependencies ensures a smooth development and testing experience.