SMART CROP PREDICTOR
A PROJECT REPORT

Submitted by

SIVARAJ T (113224071091)

In the partial fulfillment of the award of the degree

Of

BACHELOR OF TECHNOLOGY IN
INFORMATION TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

VELAMMAL ENGINEERING COLLEGE,CHENNAI-66.
(AnAutonomousInstitution,Affiliated toAnnaUniversity,chennai)

OCTOBER-2025

VELAMMAL ENGINEERING COLLEGE DEPARTMENT,
OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

certified that this project report titled **" SMART CROP PREDICTOR"** is the bonafide work of **SIVARAJ T (113224071091)**, Who carried out the Mini project work under my supervision

SIGNATURE

INTERNAL GUIDE,
Mrs.Rangini.M,
Assistant Professor,
Department Of InformationTechnology,
Velammal Engineering College,
Ambattur-Redhills road, Chennai – 600066.

SIGNATURE

HEAD OF THE DEPARTMENT,
Dr.Jeevaa Katiravan
Professor and Head,
Department Of Information Technology,
Velammal Engineering College,
Ambattur-Redhills road, Chennai - 600066.

CERTIFICATE OF EVALUATION

COLLEGE NAME :VELAMMAL ENGINEERING
COLLEGE BRANCH :INFORMATION TECHNOLOGY
SEMESTER :III

SI.NO	Project Members	Title of the Project	Project Supervisor
1	SIVARAJ T (113224071091)	SMART CROP PREDICTOR	Mrs RANGINI.M

above student This report of project work submitted by the above student in partial fulfillment for the award of Bachelor of INFORMATION TECHNOLOGY degree in Velammal Engineering College was evaluated and confirmed to be-reports of the work done by the and then assessed.

Submitted for Internal Evaluation held
on....................... .

INTERNAL
EXAMINER

ABSTRACT

The aim of this project is to analyze agricultural data and develop a system that recommends the most suitable crop based on soil and environmental factors using modern data analysis techniques. The data is processed using Python libraries such as Pandas for data handling, Matplotlib and Seaborn for data visualization, and Scikit-learn for machine learning model development. Various types of visualizations such as count plots, box plots, and heatmaps are used to represent the data in a meaningful and easy-to-understand format. The count plot shows the frequency of different crops in the dataset, while the box plot compares the distribution of essential soil nutrients like Nitrogen (N), Phosphorus (P), and Potassium (K). The correlation heatmap helps in understanding the relationship between different soil and climatic parameters.

Using these insights, a Random Forest Classifier model is trained to predict the most suitable crop based on parameters such as temperature, humidity, rainfall, pH, and nutrient content. Through this system, farmers and agricultural planners can identify the best crop to grow under given conditions, improving productivity and sustainability. The analysis reveals that the model achieves high accuracy in prediction, demonstrating the effectiveness of machine learning in supporting modern agricultural decision-making.

KEYWORDS

Crop Recommendation, Agriculture, Machine Learning, Random Forest, Python, Pandas, Seaborn, Matplotlib, Data Visualization, Soil Analysis, Sustainable Farming

ACKNOWLEDGEMENT

Behind every achievement lies an unfathomable sea of gratitude to those who achieved it, without whom it would ever have come into existence. To them we express our words of gratitude.We give all the glory and thanks to our almighty GOD for showering upon, the necessary wisdom and grace for accomplishing this project . We express our gratitudeand thanks to Our Parents first for giving health and sound mind for completing this project . First of all, we would like to express our deep gratitude to our beloved and respectable Chairman Thiru M.V. Muthuramalingam and our Chief Executive Officer Thiru M.V.M. Velmurugan for thier kind encouragement . We express our deep gratitude to Dr.Sathish Kumar, Principal , Velammal Engineering College for his helpful attitude towards this project . We wish to express our sincere thanks and gratitude to Dr.Jeevaa Katiravan , Professor and Head of the Department, Department of Information technology for motivating anencouraging in every part of our project.We express our sincere gratitude to the Project Coordinator Mrs.Rangini.M Assistant Professor, Department of Information technology for their invaluable guidance in shaping of this project without them this project would not have been possible.Our thanks to all other Faculty and Non- Teaching Staff members of our department for their support and peers for having stood by me and helped me to complete this project.

TABLE OF CONTENTS

1.INTRODUCTION

The “Crop Recommendation System using Machine Learning” project aims to analyze agricultural and environmental data to suggest the most suitable crop for cultivation based on soil and weather conditions. The main objective of this project is to transform raw agricultural data into meaningful insights that can help farmers make data-driven decisions and improve crop productivity.

This project utilizes Python and its powerful libraries such as Pandas, Seaborn, Matplotlib, and Scikit-learn for data processing, visualization, and model building. Various visualizations, including count plots, box plots, and heatmaps, are used to represent the relationships between soil nutrients, temperature, humidity, pH levels, and rainfall. These visualizations make it easier to understand patterns in soil fertility and environmental influence on crop growth.

A Random Forest Classifier machine learning model is implemented to predict the most suitable crop based on key features like Nitrogen (N), Phosphorus (P), Potassium (K), temperature, humidity, pH, and rainfall. The model analyzes historical crop data to learn patterns and relationships between environmental factors and crop types, providing accurate recommendations for future cultivation.

Through this project, farmers and agricultural planners can gain valuable insights into soil health and crop suitability, leading to better resource management and increased yield. Furthermore, this project demonstrates the potential of Machine Learning and Data Visualization in addressing real-world agricultural challenges, bridging the gap between traditional farming practices and modern data-driven decision-making in agriculture.

1.1 MOTIVATION

The motivation behind this project, “Crop Recommendation System using Machine Learning,” arises from the growing need for data-driven decision-making in modern agriculture. Farmers often face challenges in selecting the most suitable crop for cultivation due to varying soil conditions, climate, and nutrient availability. Although agricultural data is available, it is often underutilized or difficult to interpret.

By applying Machine Learning and data analysis techniques, valuable insights can be extracted from soil and environmental data to recommend the optimal crop for a given condition. This approach helps in improving farm productivity, resource management, and sustainable agricultural practices. The project demonstrates how advanced data analytics and predictive modeling can bridge the gap between traditional farming methods and modern, technology-driven agriculture, enabling farmers to make informed, efficient, and profitable decisions.

1.2 PROBLEMS

1. **Dependence on Traditional Knowledge:**

Many farmers rely on traditional knowledge or personal experience to decide which crops to cultivate. This can lead to suboptimal crop choices, reduced yield, and inefficient use of resources.

2. **Underutilization of Agricultural Data:**

Although soil and environmental data are often collected, they are not effectively analyzed or used to guide crop selection decisions.

3. **Difficulty in Identifying Patterns:**

Without proper data analysis and visualization tools, it is challenging to identify patterns and relationships between soil nutrients, climate conditions, and crop productivity.

4. **Inaccurate Crop Selection**:

Choosing crops without a systematic approach can result in poor harvests, financial losses, and wasted resources such as fertilizers and water.

5. **Operational Inefficiency in Farming Practices:**

Farmers and agricultural planners lack data-driven guidance for optimal crop planning, leading to inefficient use of land, labor, and inputs.

1.3 PROPOSED SYSTEM

- The proposed system, “Crop Recommendation System using Machine Learning,” utilizes Python as the core programming language along with data science libraries such as Pandas, Seaborn, Matplotlib, and Scikit-learn. This system aims to overcome the limitations of traditional, experience-based crop selection methods by providing a data-driven approach for recommending the most suitable crops based on soil and environmental conditions.

- The system focuses on collecting, processing, and analyzing agricultural data, including soil nutrients (N, P, K), temperature, humidity, pH, and rainfall, to provide accurate crop recommendations. Pandas is used for efficient data handling and preprocessing, while Seaborn and Matplotlib are used to create meaningful visualizations such as count plots, box plots, and heatmaps. These visualizations help in identifying patterns, trends, and relationships between environmental factors and crop types.

- The Random Forest Classifier model is trained using historical crop data to predict the most suitable crop for given conditions. This predictive approach allows farmers and agricultural planners to make informed decisions regarding crop selection, soil management, and resource allocation. By implementing this system, traditional trial-and-error methods can be replaced with data-driven crop planning, improving productivity, sustainability, and efficiency in farming practices.

- Furthermore, the system serves as a practical example of applying Machine Learning and data visualization techniques in agriculture, demonstrating the value of technology in modern farming and decision-making.

2. SOFTWARE USED

1. **Python** – Main language for data analysis, machine learning, and visualization.
2. **Pandas** – Library for data handling, filtering, cleaning, and summarizing datasets.
3. **Matplotlib** – Library for creating line charts, bar charts, box plots, and other visualizations.
4. **Seaborn** – Library for advanced statistical data visualization and graphical representations.
5. **Scikit-learn** – Library for building, training, and evaluating machine learning models.
6. **Jupyter Notebook / Google Colab** – Environment for writing and running Python code interactively.
7. **Microsoft Word** – For preparing project documentation and report.
8. **Operating System (Windows / Linux)** – Platform to run Python and other software efficiently.

2.1 PLATFORM USED

1. **Python (Anaconda / Jupyter Notebook / Google Colab):**

Primary platform for writing, executing, and visualizing Python code, performing data analysis, and building machine learning models.

2. **Operating System (Windows / Linux):**

Provides the environment to run Python, libraries, and tools efficiently.

3. **Microsoft Word:**

For preparing, formatting, and documenting the project report professionally.

2.2 TOOLS AND LIBRARIES

1. **Python Libraries:**

- **Pandas** – for data manipulation, filtering, and summarizing datasets.
- **Matplotlib** – for creating line charts, bar charts, box plots, and other visualizations.
- **Seaborn** – for advanced statistical data visualization and improved graphical representation.
- **Scikit-learn** – for building, training, and evaluating machine learning models (Random Forest Classifier).

2.**Development Tools:**

- **VS Code / Jupyter Notebook / Google Colab** – environments for writing, executing, and visualizing Python code interactively.

3.**Data Tools / Other Platforms:**

- **Python (Jupyter Notebook / Google Colab)** – for coding, data analysis, model building, and visualization.
- **Microsoft Word** – for report preparation and documentation.
- **Operating System (Windows / Linux)** – to run Python, libraries, and tools efficiently.

3.TOOLS IMPLEMENTATION

3.1 PYTHON:

Python is themainprogramming language used in this project for data analysis and visualization. It offers simple syntaxandreadability, making coding easier and faster. With support for numerous libraries, Python allowsefficient handling, processing, and visualization of large datasets. Its flexibility and powerful featuresmake it ideal for creating line charts, bar charts, and scatter plots to understand patterns in the data. Pythonis widely used in real-world data science and business analytics applications, making it a perfect choiceforthis project.

3.2 PANDAS AND MATPLOTLIB:

Pandas isapowerful Python library used in this project for handling, cleaning, and summarizing data efficiently.Itprovides easy-to-use data structures, such as DataFrames and Series, which allow organized storage andmanipulation of data. With Pandas, operations like filtering, grouping, sorting, and aggregating data becomestraightforward and time-efficient. This library plays a crucial role in preparing the dataset for visualization and analysis, ensuring that the data is accurate and wellstructured. Pandas is widely used in data science and analytics projects for its robustness, flexibility, and ability to handle large volumes of data effectively.

3.3 SEABORN:

Seaborn enhances visualizations with heatmaps and distribution plots, helping to analyze correlations between features like soil nutrients and crop types.

3.4 STREAMLIT:

Streamlit is an open-source Python framework used for building interactive web applications for data analysis and visualization. It allows developers to quickly create dashboards and display charts, tables, and other visual elements without extensive web development knowledge. Streamlit integrates

4. ALGORITHM:

- Step 1: Collect agricultural data including soil nutrients (N, P, K), temperature, humidity, pH, rainfall, and crop types.
- Step 2: Clean the data using Pandas — handle missing values, remove errors, and ensure consistency.
- Step 3: Organize the data into DataFrames for analysis and machine learning.
- Step 4: Analyze patterns and relationships between soil, climate, and crop types.
- Step 5: Visualize the data using Matplotlib and Seaborn (count plots, box plots, heatmaps).
- Step 6: Train a Random Forest Classifier on the dataset to predict suitable crops.
- Step 7: Make predictions based on input parameters (N, P, K, temperature, humidity, pH, rainfall).
- Step 8: Interpret results and summarize findings for reporting or interactive application.

5.SOURCE CODE:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
import joblib
import warnings

warnings.filterwarnings("ignore", category=FutureWarning)

# Load dataset
data = pd.read_csv("C:\\Users\\thula\\Downloads\\crop_data.csv")
print(data.head())

# Visualizations
plt.ion()  # allows code to continue after plotting

plt.figure(figsize=(8, 5))
sns.countplot(x="crop", data=data, hue="crop", palette="Set2", legend=False)
plt.title("Crop Frequency")
plt.show()

# Split features and target
X = data.drop("crop", axis=1)
y = data["crop"]

# Train-Test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train model
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Save model
joblib.dump(model, "crop_model.pkl")
```

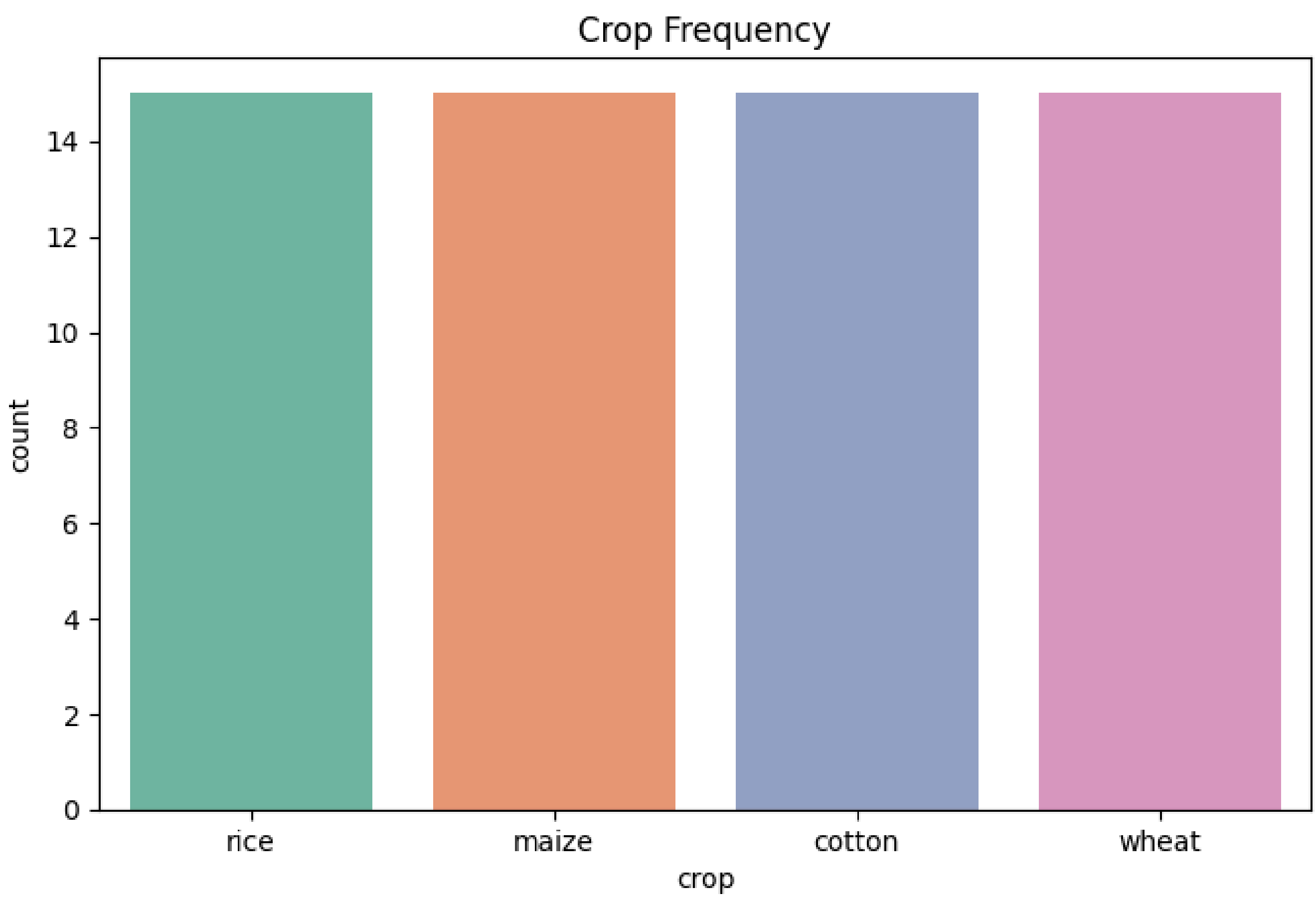
```
# Prediction function
def recommend_crop(N, P, K, temperature, humidity, ph, rainfall):
    model = joblib.load("crop_model.pkl")
    sample = pd.DataFrame([[N, P, K, temperature, humidity, ph, rainfall]],
                          columns=["N", "P", "K", "temperature", "humidity", "ph", "rainfall"])
    prediction = model.predict(sample)
    return prediction[0]

# Example prediction
result = recommend_crop(80, 50, 45, 25, 65, 6.8, 150)
print("\n🌱 Recommended Crop:", result)
```

6.OUTPUT

	N	P	K	temperature	humidity	ph	rainfall	crop
0	90	42	43	25	80	6.5	200	rice
1	80	40	40	26	82	6.8	180	rice
2	78	38	38	27	84	6.2	150	rice
3	60	35	36	28	70	6.0	100	maize
4	40	30	30	29	72	5.8	80	maize

🌱 Recommended Crop: cotton



7. DRAWBACKS

- **Limited Dataset Size:**

The accuracy of the model depends on the dataset. A small dataset may not cover all soil types, climates, or crops, which can reduce prediction reliability for new or unseen conditions.

- **Regional Constraints:**

The model is trained on specific soil and climate data. Its predictions may not be accurate for regions with different environmental conditions unless retrained with local data.

- **Static Model:**

The Random Forest model is static. Any changes in soil fertility, weather patterns, or new crops require retraining the model with updated data.

- **No Real-Time Data Integration:**

The system does not connect to real-time soil sensors, weather forecasts, or external agricultural databases, which limits its dynamic adaptability.

- **Simplified Assumptions:**

The model assumes crop growth depends only on the provided parameters (N, P, K, temperature, humidity, pH, rainfall). Other factors like pest attacks, irrigation methods, and market demand are not considered.

- **User Dependency on Input Accuracy:**

Incorrect or approximate values of soil nutrients, rainfall, or other parameters can lead to wrong crop recommendations.

8.FUTURE ENHANCEMENT

- **Integration with Real-Time Data:**
  Connect the system to soil sensors, weather APIs, or satellite data to provide real-time crop recommendations based on current conditions.
- **Expanded Dataset:**
  Include more crops, soil types, and climatic regions to improve model accuracy and make it applicable to a wider range of areas.
- **Advanced Machine Learning Models:**
  Explore models like XGBoost, Gradient Boosting, or Neural Networks to potentially improve prediction accuracy and handle more complex patterns.
- **Mobile / Web Application:**
  Develop a user-friendly mobile or web app for farmers to input soil and weather data easily and receive instant crop recommendations.
- **Incorporate Additional Parameters:**
  Include factors such as soil moisture, pest risk, irrigation methods, market demand, and fertilizer usage for more comprehensive recommendations.
- **Explainable AI:**
  Add a feature to explain why a crop is recommended, helping farmers understand the reasoning behind predictions and build trust in the system.
- **Decision Support System:**
  Extend the system to provide planting schedules, yield estimates, and crop rotation plans for better farm management and sustainability.

9.CONCLUSION

The Crop Recommendation System using Machine Learning successfully demonstrates how data analysis and predictive modeling can assist in modern agriculture. By analyzing soil nutrients, environmental factors, and historical crop data, the system provides accurate recommendations for the most suitable crops, helping farmers make informed decisions.

Through data visualization with Matplotlib and Seaborn, patterns and relationships between soil, climate, and crop types are clearly understood, while the Random Forest Classifier ensures reliable predictions. This project highlights the importance of data-driven agriculture, improving productivity, resource management, and sustainability.
Although the system has limitations, such as dependency on dataset size and environmental constraints, it lays a strong foundation for future enhancements, including real-time data integration, expanded crop coverage, and mobile/web application support.

Overall, the project demonstrates the practical application of Python and machine learning in agriculture, bridging the gap between traditional farming practices and modern technology-based solutions.

10. REFERENCE

1. Scikit-learn Developers, "Scikit-learn: Machine Learning in Python," [Online]. Available: https://scikit-learn.org/
2. Wes McKinney, "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython," O'Reilly Media, 2nd Edition, 2017.
3. Hunter, J. D., "Matplotlib: A 2D Graphics Environment," Computing in Science & Engineering, 2007.
4. Seaborn Documentation, "Statistical Data Visualization," [Online]. Available: https://seaborn.pydata.org/
5. Jason Brownlee, "Machine Learning Algorithms from Scratch," Machine Learning Mastery, 2020.
6. Pandas Documentation, "Python Data Analysis Library," [Online]. Available: https://pandas.pydata.org/
7. UCI Machine Learning Repository, "Crop Recommendation Dataset," [Online]. Available: https://archive.ics.uci.edu/
8. S. Sharma, "A Review on Crop Recommendation System using Machine Learning Techniques," International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE), Vol. 10, Issue 5, May 2022.
9. Streamlit Documentation, "Streamlit for Data Science Apps," [Online]. Available: https://streamlit.io/
10. T. Hastie, R. Tibshirani, J. Friedman, "The Elements of Statistical Learning," Springer, 2009.