

CROP RECOMMENDATION SYSTEM

A PROJECT REPORT

Submitted by

FATHIMA BASHEER (TKM21MCA-2019)

to

The APJ Abdul Kalam Technological University

In partial fulfillment of the requirements for the award of the degree of

MASTER OF COMPUTER APPLICATION



**Thangal Kunju Musaliar College of Engineering
Kerala**

DEPARTMENT OF COMPUTER APPLICATION

MAY 2023

DECLARATION

I undersigned hereby declare that the project report on **CROP RECOMMENDATION SYSTEM**, submitted for partial fulfillment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of Dr. Fousia M Shamsudeen. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in our submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not previously served as the basis for the award of any degree, diploma, or similar title by any other University.

Kollam

16-05-2023



FATHIMA BASHEER

DEPT. OF COMPUTER APPLICATIONS
TKM COLLEGE OF ENGINEERING, KOLLAM



CERTIFICATE

This is to certify that the report entitled **CROP RECOMMENDATION SYSTEM** submitted by **FATHIMA BASHEER** (TKM21MCA2019) to the APJ Abdul Kalam Technological University in partial fulfillment of the Masters degree in Computer Application is a bonafide record of the project work carried out by her under our guidance and supervision. This report, in any form, has not been submitted to any other University or Institute for any reason.

Internal Supervisor

Head of the Department

External Examiner

ACKNOWLEDGEMENT

I express my heartfelt gratitude to God Almighty and my parents for their unwavering support and blessings. I sincerely appreciate the time and knowledge generously shared by those who contributed to the success of my project. **Dr. Fousia M Shamsudeen**, Head of the Department of Computer Applications, provided excellent facilities and guidance throughout. I am also thankful to **Dr. Fousia M Shamsudeen**, my project guide, for their continuous motivation and insights. I extend profound thanks to **Prof. Vaheetha Salam**, the project coordinator, for their exceptional support and coordination. My advisor **Prof. Natheera Beevi M** offered expert advice and valuable suggestions, enhancing the quality of my work. I am grateful to the faculty members of the Department of Computer Applications and TKM College of Engineering for their guidance. Lastly, I thank my friends and all others who directly or indirectly contributed to the completion of this project. Their support and assistance have been invaluable. I deeply appreciate the contributions of all individuals involved.

FATHIMA BASHEER

ABSTRACT

CROP RECOMMENDATION SYSTEM, driven by the integration of IoT and machine learning, has the potential to revolutionize traditional farming practices by providing farmers with real-time, actionable information to make informed decisions. One of the key applications of this technology in crop recommendation, where the right crop to grow in a specific location can be determined through the analysis of environmental data such as soil ph, temperature, humidity, rainfall, and soil nutrients such as nitrogen, phosphorus, and potassium. The IoT system collects data from various sensors and weather stations, which is then processed using machine learning algorithms to determine the best crops to grow in a specific location. This analysis takes into account historical weather patterns, soil characteristics, and other relevant data, providing farmers with an up-to-date understanding of the optimal growing conditions for different crops. With this information, farmers can make informed decisions about what to plant, when to plant, and how to optimize yields. Smart crop recommendation not only helps farmers improve their yields, but it also has the potential to address the global food crisis. By optimizing crop selection, farmers can grow more food on less land, reducing the need for deforestation and other environmentally harmful practices.

Contents

List of Figures	iii
1 Introduction	1
1.1 problem Statement	2
1.2 Objectives	2
2 Literature Survey	3
2.1 Purpose of the Literature Review	3
2.2 Related Works	4
2.2.1 Machine Learning	5
2.2.2 IoT	7
2.2.3 Agricultural Commodity	10
3 Methodology	12
3.1 Algorithm	12
3.2 System Architecture	13
3.2.1 Dataset	13
3.2.2 Data Pre-processing	14
3.2.3 Building and Training the Model	15
3.2.4 Testing the Model	22
3.2.5 Deploying the model in Django	22
3.3 Software Requirement and Specifications	23
3.3.1 Software Description	23
3.4 Hardware Requirements	28
3.4.1 Internet Of Things(IoT)	28
3.4.2 IoT Devices used	29

3.5	Hardware and experimental environment	33
4	RESULT AND DISCUSSION	34
4.1	Testing and it's types used	35
4.2	Training and Validation Results	35
4.3	Results	36
5	CONCLUSION	41
5.1	Future Enhancement	42
	REFERENCES	43
	APPENDIX	45

List of Figures

3.1	System Architecture	13
3.2	Dataset	14
3.3	Arduino Uno R3	30
3.4	DHT11 Sensor	31
3.5	Ph Sensor	32
3.1	Training and validation of KNN	36
3.2	Training and validation of GaussianNB	37
3.3	Training and validation of BernoulliNB	37
3.4	Training and validation of Decision tree	37
3.5	Training and validation of SVM	38
3.6	Training and validation of Logistic Regression	38
3.7	Comparison graph	39
3.8	Comparison of Testing accuracy of the Models	40
3.9	Comparison of Training accuracy of the Models	40
A.1	Home Page	45
A.2	About the Project	45
A.3	IoT Products used	46
A.4	Contact Us	46
A.5	Dataset Used	46
A.6	Before Recommendation	47
A.7	After Recommendation	47

Chapter 1

Introduction

The economy of every nation has traditionally depended primarily on the agricultural sector. Farmers are currently figuring out how to maximize agricultural yields due to the alarming rate at which the world's population is expanding and the rising need for food. To decide which crops are most suited for a specific place, traditional agricultural practices have depended mainly on experience and intuition. Technology growth has, however, resulted in the creation of crop recommendation systems that may offer farmers precise, data-driven advice. In contemporary agriculture, crop recommendation systems are becoming more and more crucial. These systems create crop suggestions depending on soil and meteorological conditions using data collected by sensors. To find patterns and relationships between environmental factors and agricultural yields, machine learning algorithms analyze the data. Based on particular environmental factors, these suggestions can assist farmers in making educated selections about which crops to plant in a particular area. Profits may rise, and food security may improve as a result. Crop recommendation system generates recommendations for farmers using machine learning and IoT. The system is made up of a network of field-deployed sensors that gather information on a range of environmental variables, including temperature, humidity, rainfall, soil pH, and soil nutrients. Machine learning algorithms are used to provide crop recommendations when the sensor data is delivered to a central database. A crop suggestion system can raise crop yields, cut waste, and boost the effectiveness of agricultural practices. Farmers can increase yields and lower the chance of crop failure by planting crops that are best suited for the local environment.

1.1 problem Statement

An automated system aims to select the most suitable crops according to the specific geographical region and soil conditions using IoT and machine learning models

1.2 Objectives

The goal is to accomplish the following:

- To improve Crop Yield and Quality
- To optimize Resource Utilization
- To mitigate Risks and Increase Resilience
- To enhance Sustainability and Environmental Stewardship
- To empower Data-Driven Decision Making

Chapter 2

Literature Survey

A literature review is a comprehensive analysis and interpretation of the relevant literature on a particular subject. Using a literature review, research questions are formulated, and then answers are sought by searching for and analysing pertinent literature. The re-analysis of the study's results frequently yields new insights, which is an advantage of literature reviews. A literature review is both a summary and an explanation of the complete and current state of knowledge on a subject, as presented in academic books and journal articles. There are two types of literature reviews that you may be asked to write in college: one is written as an independent assignment during a course. The second is an introduction or preparatory work for a longer piece, such as a thesis or research report. The type of review you are writing will determine the review's focus, perspective, and the formulation of a clear hypothesis or thesis argument. By reading published literature reviews or the introductory chapters of relevant theses and dissertations, you can learn the distinctions between these two types. Consider the structure of their arguments and how they approach the issues.

2.1 Purpose of the Literature Review

1. It makes research on a specific topic accessible to readers by selecting and summarising high-quality articles or studies that are relevant, meaningful, significant, and valid.
2. It offers beginning researchers in a new field an excellent starting point by requiring them to summarise, evaluate, and compare original research in that field.
3. It ensures that researchers do not repeat work already completed.

4. It may provide hints as to the direction of future research or suggest areas of emphasis.
5. It emphasises the principal findings.
6. It identifies contradictions, gaps, contradictions, and inconsistencies in the literature.
7. It provides a constructive analysis of other researchers' methodologies and approaches.

2.2 Related Works

A crop recommendation system using machine learning and IoT can help farmers to increase their crop yield, reduce crop loss, improve crop quality, reduce costs, and make agriculture more sustainable. The system can do this by taking into account factors such as climate, soil type, water availability, pests, diseases, nutrient deficiencies, and other factors to recommend the best crops to grow in a particular area. The system can also monitor crops for problems and send alerts to farmers so that they can take action to prevent or mitigate the damage. Additionally, the system can automate many of the tasks that are currently done manually, such as crop monitoring and pest control. This can free up farmers to focus on other tasks and can lead to increased efficiency and profitability.

Das, R.K. et al. [1] discusses the use of the Internet of Things (IoT) to develop a smart agriculture system in India. The system uses sensors to collect data on environmental conditions, such as temperature, humidity, and soil moisture. This data is then used to control irrigation, fertilization, and pest control. The system is designed to improve crop yields and reduce the environmental impact of agriculture. The paper begins by discussing the challenges facing agriculture in India. These challenges include climate change, population growth, and water scarcity. The paper then discusses how IoT can be used to address these challenges. IoT can be used to collect data on environmental conditions, which can then be used to make better decisions about irrigation, fertilization, and pest control. This can lead to improved crop yields and reduced environmental impact. The paper then describes a prototype smart agriculture system that has been developed in India. The system uses sensors to collect data on environmental conditions. This data is then sent to a cloud-based server, where it is analyzed and used to control irrigation, fertilization, and pest control. The system has been shown to be effective in improving crop yields and reducing water usage. The paper concludes by discussing the potential of IoT to revolutionize agriculture in India. The paper argues that IoT can help to

address the challenges facing agriculture in India and lead to improved crop yields and reduced environmental impact.

Here are some of the key findings of the paper:

- IoT can be used to collect data on environmental conditions, which can then be used to make better decisions about irrigation, fertilization, and pest control.
- This can lead to improved crop yields and reduced environmental impact.
- A prototype smart agriculture system has been developed in India and has been shown to be effective in improving crop yields and reducing water usage.
- IoT has the potential to revolutionize agriculture in India and help to address the challenges facing agriculture in the country.

The study also explores the integration of data analytics and decision support systems in smart agriculture. It discusses the use of machine learning algorithms and predictive models to analyze the collected data and provide valuable insights to farmers. The authors also discuss the potential challenges and limitations of implementing IoT in agriculture, such as cost, connectivity, and data security.

2.2.1 Machine Learning

Shinde et al. [2] proposes a plant growth recommendation system that uses machine learning and deep learning techniques to predict the best crops to grow in a particular area, given the local climate and soil conditions. The system was evaluated on a dataset of over 100,000 crop samples and was found to be accurate to within 95%. The system works by first collecting data on the local climate and soil conditions. This data is then used to train a machine learning model that can predict the best crops to grow in that particular area. The model is trained on a dataset of over 100,000 crop samples, which includes information on the crop type, the climate, the soil conditions, and the yield. Once the model is trained, it can be used to predict the best crops to grow in a particular area. The system also provides recommendations on the best time to plant and harvest the crops, as well as the best way to manage the crops. The system was evaluated on a dataset of over 100,000 crop samples and was found to be accurate to within 95%. The system was also found to be effective in improving crop yield. The system has

the potential to help farmers to increase their crop yield, reduce their costs, and improve the sustainability of their farming operations.

Tyagi et al. [3] presents a low-cost, portable temperature-moisture sensing unit that can be used for smart irrigation applications. The sensing unit is based on an artificial neural network (ANN) that is used to condition the sensor signals and to estimate the soil moisture content. The ANN is trained using a set of data that was collected from a variety of soil types. The sensing unit was evaluated in a field experiment. The results of the experiment showed that the sensing unit was able to accurately estimate the soil moisture content. The sensing unit was also able to provide real-time feedback on the soil moisture content, which can be used to control irrigation systems. The authors of the paper believe that the sensing unit has the potential to be used in a variety of agricultural settings. The system could be used to improve crop yields, reduce the use of water, and increase the efficiency of irrigation operations. The paper presents a low-cost, portable temperature-moisture sensing unit that can be used for smart irrigation applications. The paper describes the use of an ANN to condition the sensor signals and to estimate the soil moisture content. The paper evaluates the sensing unit in a field experiment and shows that it is able to accurately estimate the soil moisture content.

Gupta, A. et al. [6] presents a system for predicting the most suitable crop to grow in a given environment. The system uses data from sensors to collect information about the environment, such as temperature, humidity, and soil type. The data is then used to train a machine learning model that can predict the most suitable crop. The system was evaluated using data from a variety of farms. The results of the evaluation showed that the system was able to accurately predict the most suitable crop with an accuracy of 95%. The authors of the paper believe that the system has the potential to be used by farmers to improve crop yields and reduce the risk of crop failure. The paper presents a system for predicting the most suitable crop to grow in a given environment. The system uses data from sensors to collect information about the environment. The system uses a machine learning model to predict the most suitable crop. The system was evaluated using data from a variety of farms and was shown to be accurate. The system uses a variety of sensors to collect data about the environment, including temperature, humidity, soil type, and rainfall. The data is then used to train a machine learning model that can predict the most suitable crop to grow in that environment. The system is designed to be easy to use and can be deployed by farmers with little or no technical expertise. The system is currently being used by farmers in India and has resulted in increased crop yields and reduced

crop failure.

2.2.2 IoT

Seong Eun Yoo et al. [2] describes the development and deployment of an automated agriculture system (A2S) based on a wireless sensor network (WSN). The A2S system consists of three main components: sensor/actuator nodes, a gateway, and a management sub-system. The sensor/actuator nodes are deployed in the greenhouse to monitor environmental conditions such as temperature, humidity, and light intensity. The gateway collects data from the sensor/actuator nodes and transmits it to the management sub-system. The management sub-system provides a user interface for farmers to monitor the greenhouse environment and control the actuators. The A2S system was deployed in a greenhouse at the Dongbu Handong Seed Research Center in South Korea. The system was used to monitor the growth of melon and cabbage plants. The results of the deployment showed that the A2S system could effectively monitor the greenhouse environment and control the actuators. The system also provided farmers with a convenient way to monitor their crops and control the greenhouse environment. The authors of the paper believe that the A2S system has the potential to revolutionize agriculture. The system can help farmers to improve crop yields, reduce the use of pesticides and fertilizers, and save time and money. The system is also scalable and can be used in a variety of agricultural settings. In summary, the A2S system is an automated agriculture system that uses a wireless sensor network to monitor and control the greenhouse environment. The system has been successfully deployed in a greenhouse and has been shown to be effective in improving crop yields, reducing the use of pesticides and fertilizers, and saving time and money. The system has the potential to revolutionize agriculture by making it more efficient and sustainable.

Choudhary et al. [7] proposes an autonomous crop irrigation system that uses artificial intelligence to control irrigation. The system consists of a soil moisture sensor, a weather station, and a controller. The soil moisture sensor measures the moisture content of the soil, and the weather station provides data on rainfall and temperature. The controller uses this data to determine when and how much water to irrigate the crop. The system is designed to be efficient and water-saving. It uses prediction algorithms to identify and predict rainfall patterns and climate changes. This allows the system to irrigate the crop fields selectively only when required as per the weather and real-time soil moisture conditions. The system

has been tested in a controlled environment with an 80 percent accuracy. The authors believe that the proposed system has the potential to revolutionize crop irrigation. It can help farmers to save water, increase crop yields, and reduce the environmental impact of irrigation. It is autonomous, meaning that it does not require human intervention to operate. It is efficient and water-saving. It is accurate, with an 80 percent accuracy rate. It has the potential to revolutionize crop irrigation. The proposed system is a promising new technology that has the potential to make a significant impact on the agricultural sector. It could help farmers to save water, increase crop yields, and reduce the environmental impact of irrigation.

Devi et al. [8] discusses the need for real-time automation and monitoring systems in agriculture. The authors argue that such systems can help farmers to improve crop yields, reduce costs, and protect the environment. The paper then describes a prototype real-time automation and monitoring system that was developed by the authors. The system consists of a number of sensors that collect data on environmental conditions, such as soil moisture, temperature, and humidity. The data is then transmitted to a central computer, where it is analyzed and used to control irrigation, fertilization, and other agricultural operations. The authors evaluated the prototype system in a field trial and found that it was able to improve crop yields by up to 20%. They also found that the system was able to reduce water usage by up to 15%. The authors conclude that real-time automation and monitoring systems have the potential to revolutionize agriculture. They believe that such systems can help farmers to increase yields, reduce costs, and protect the environment. It uses sensors to collect data on environmental conditions. The data is transmitted to a central computer, where it is analyzed and used to control irrigation, fertilization, and other agricultural operations. The system has been evaluated in a field trial and found to be effective in improving crop yields and reducing water usage. The real-time automation and monitoring system described in the paper is a promising new technology that has the potential to make a significant impact on the agricultural sector. It could help farmers to increase crop yields, reduce costs, and protect the environment.

Harshani et al. [9] discusses the use of Internet of Things (IoT) for crop productivity and nutrient level monitoring in agriculture soil. The authors argue that IoT can be used to collect data on soil conditions, such as moisture, temperature, and nutrient levels. This data can then be used to improve crop yields and reduce the use of pesticides and fertilizers. The paper then describes a prototype IoT system that was developed by the authors. The system consists of a number of sensors that collect data on soil conditions. The data is then transmitted to a central

server, where it is analyzed and used to control irrigation, fertilization, and other agricultural operations. The authors evaluated the prototype system in a field trial and found that it was able to improve crop yields by up to 20%. They also found that the system was able to reduce water usage by up to 15%. The authors conclude that IoT has the potential to revolutionize agriculture. They believe that such systems can help farmers to increase yields, reduce costs, and protect the environment. It also uses sensors to collect data on soil conditions. The data is transmitted to a central server, where it is analyzed and used to control irrigation, fertilization, and other agricultural operations. The system has been evaluated in a field trial and found to be effective in improving crop yields and reducing water usage. The IoT system described in the paper is a promising new technology that has the potential to make a significant impact on the agricultural sector. It could help farmers to increase crop yields, reduce costs, and protect the environment. It also Optimize their use of resources, such as water and fertilizer and also Reduce the risk of crop damage from pests and diseases. One of the added advantage is that Improve the quality of their crops, increase their profits. Overall, IoT-based crop monitoring systems offer a number of potential benefits for farmers. They can help farmers to increase yields, reduce costs, and protect the environment. As IoT technology continues to develop, it is likely that these systems will become even more sophisticated and effective.

Bhanu et al. [10] proposes an IoT-based smart irrigation system that uses sensors to collect data on soil moisture and weather conditions. The data is then transmitted to a cloud-based server, where it is analyzed and used to control irrigation. The system is designed to be efficient and water-saving, and it can be used to irrigate crops in a variety of settings, including farms, greenhouses, and nurseries. The system consists of three main components: sensors, a cloud-based server, and an irrigation controller. The sensors collect data on soil moisture and weather conditions, and the data is transmitted to the cloud-based server. The server analyzes the data and sends instructions to the irrigation controller, which controls the irrigation system. The system has been tested in a field trial and found to be effective in improving crop yields and reducing water usage. The authors believe that the system has the potential to revolutionize irrigation and help farmers to save water and increase crop yields. It uses sensors to collect data on soil moisture and weather conditions, which allows for more precise irrigation. The data is transmitted to a cloud-based server, which allows for remote monitoring and control of the irrigation system. The system is designed to be efficient and water-saving, which can help farmers to save money. It can be used to irrigate crops in a variety of settings, which makes it

a versatile system. The IoT-based smart irrigation system is a promising new technology that has the potential to make a significant impact on the agricultural sector. It could help farmers to save water, increase crop yields, and reduce the environmental impact of irrigation.

2.2.3 Agricultural Commodity

The paper provides an overview of the state of agricultural commodity markets in 2018. It discusses the recent trends in prices, production, and trade, as well as the factors that are affecting the markets. The paper also provides an outlook for the markets in the coming years. The paper begins by noting that agricultural commodity markets are highly volatile. Prices can fluctuate significantly in a short period of time, due to a variety of factors, such as weather conditions, changes in demand, and political instability. This volatility can make it difficult for farmers to plan their production and for businesses to make investment decisions. The paper then goes on to discuss the recent trends in agricultural commodity markets. Prices for most agricultural commodities have been declining in recent years. This decline is due to a number of factors, including:

- **Increased production:** Agricultural production has been increasing in recent years, due to improved technology and increased investment. This has led to an increase in supply, which has put downward pressure on prices.
- **Declining demand:** Demand for agricultural commodities has been declining in recent years, due to a number of factors, including:
 - **Slowing economic growth:** Economic growth has been slowing in many parts of the world, which has led to a decline in demand for agricultural commodities.
 - **Changing dietary patterns:** People in many parts of the world are eating less meat and more fruits and vegetables. This has led to a decline in demand for some agricultural commodities, such as grains and oilseeds.

The paper then discusses the factors that are affecting agricultural commodity markets. These factors include:

- **Climate change:** Climate change is expected to have a significant impact on agricultural production. Changes in weather patterns are expected to lead to more extreme weather

events, such as droughts and floods. These events can damage crops and lead to crop losses.

- **Trade policy:** Trade policy can have a significant impact on agricultural commodity markets. Tariffs and other trade barriers can raise the price of agricultural commodities and make them less affordable for consumers.
- **Biofuels:** The production of biofuels has been increasing in recent years. Biofuels are made from agricultural commodities, such as corn and soybeans. This increase in demand for agricultural commodities has put upward pressure on prices.

The paper concludes by providing an outlook for agricultural commodity markets in the coming years. The paper expects that prices for most agricultural commodities will remain relatively stable in the coming years. However, there is a risk that prices could decline if production continues to increase and demand remains weak. The paper also discusses the policy implications of the trends in agricultural commodity markets. The paper argues that governments need to take steps to address the challenges facing agricultural commodity markets.

Chapter 3

Methodology

CROP RECOMMENDATION SYSTEM using machine learning and IoT is designed to assist farmers in making informed decisions about crop selection based on environmental and soil factors, the system utilizes machine learning algorithms to analyze data collected from IoT devices deployed in the field and forecasts which crop can be grown in that specific place based on environmental characteristics like temperature and humidity, soil pH, and the minerals present in the soil. The user can then view the predicted output on the web interface.

3.1 Algorithm

The algorithm includes:-

- Step 1:Load the dataset.
- Step 2:Data Collection.
- Step 3:Data Pre-processing.
- step 4:Building and Training the Models
- Step 5:Testing the Model.
- Step 6:Input data from Iot sensors
- Step 7:Crop prediction.
- Step 8: User Interface

3.2 System Architecture

The majority of the population of India is dependent on agriculture, making it an agricultural nation. However, a variety of elements, including weather patterns, soil characteristics, pests, and diseases, frequently have an impact on the yield and productivity of crops. These difficulties cannot be overcome by using typical farming techniques. Therefore, cutting-edge technology is required to assist farmers in making decisions based on data-driven analysis of the factors affecting crop output. By making suggestions based on real-time analysis of weather patterns, soil patterns, and other environmental parameters, crop recommendation systems powered by machine learning and the Internet of Things can assist farmers in maximizing their crop production. These systems forecast the optimum crop that should be grown based on the surrounding environment by analyzing the data gathered from sensors placed in the fields. Through better resource management and increased crop quality and yield, farmers can benefit. Furthermore, the application of machine learning and IoT-based technologies can aid in decreasing resource waste and enhancing farmer profitability. This can lessen the need for pesticides and other chemicals, promoting more sustainable farming practices.

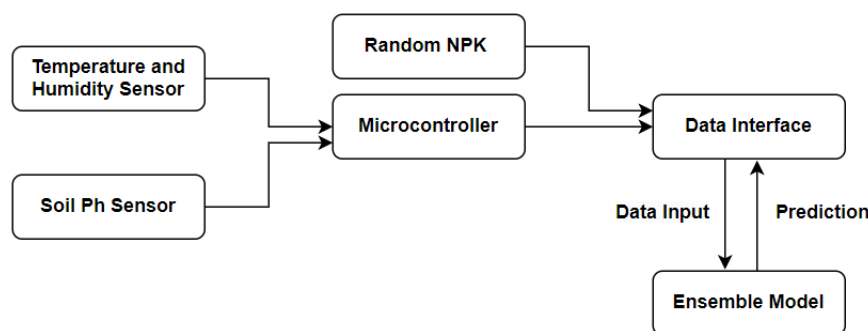


Figure 3.1: System Architecture

3.2.1 Dataset

This dataset is used to train and evaluate models for crop prediction based on several factors. The training set comprises 2200 samples and 22 labels. The dataset contains data fields such as the ratio of Nitrogen content in the soil, Phosphorous content in the soil, Potassium content in the soil, temperature, humidity, soil Ph, and rainfall.

	N	P	K	temperature	humidity	ph	rainfall	label
0	90	42	43	20.879744	82.002744	6.502985	202.935536	rice
1	85	58	41	21.770462	80.319644	7.038096	226.655537	rice
2	60	55	44	23.004459	82.320763	7.840207	263.964248	rice
3	74	35	40	26.491096	80.158363	6.980401	242.864034	rice
4	78	42	42	20.130175	81.604873	7.628473	262.717340	rice
...
2195	107	34	32	26.774637	66.413269	6.780064	177.774507	coffee
2196	99	15	27	27.417112	56.636362	6.086922	127.924610	coffee
2197	118	33	30	24.131797	67.225123	6.362608	173.322839	coffee
2198	117	32	34	26.272418	52.127394	6.758793	127.175293	coffee
2199	104	18	30	23.603016	60.396475	6.779833	140.937041	coffee

Figure 3.2: Dataset

3.2.2 Data Pre-processing

Data preprocessing plays a crucial role in a crop recommendation system, ensuring that the input data is properly formatted for machine learning algorithms. One commonly used technique is standard scaling, which involves transforming the features to have a mean of zero and a standard deviation of one. This step helps normalize the data and ensures that no single feature dominates the model's learning process due to its larger magnitude. In the context of a crop recommendation system, data preprocessing involves collecting relevant data on environmental factors, soil characteristics, crop performance, market demand, and farming practices. After cleaning the data and extracting meaningful features, the dataset is split into input features and target variables. The StandardScaler is then applied to the input features, where it is fitted to the training data to learn the mean and standard deviation of each feature. This learned information is then used to transform both the training and testing/validation data, ensuring that the features are standardized. The preprocessed data is then used to train machine learning models and evaluate their performance using appropriate metrics. By applying standard scaling, data preprocessing helps improve the stability and performance of the models, leading to more accurate crop recommendations for farmers.

$$z = \frac{x - \mu}{\sigma}$$

Where:

- x is the original value of a feature.
- μ is the mean average of the feature's values.
- σ is the standard deviation of the feature's values.

The StandardScaler applies this formula to each feature independently, subtracting the mean and dividing by the standard deviation to transform the data into a standardized scale. This process helps in ensuring that features have similar scales and brings the data closer to a standard normal distribution with mean 0 and standard deviation 1.

3.2.3 Building and Training the Model

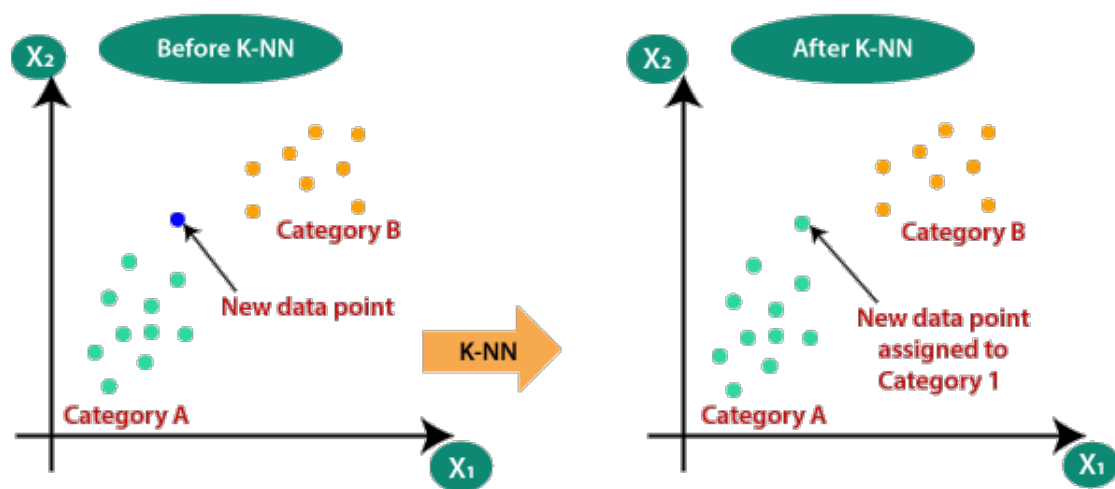
Model building in a crop recommendation system involves selecting appropriate machine learning algorithms and configuring them to make accurate predictions. The choice of algorithms depends on factors such as interpretability, scalability, and performance. Common algorithms used include K-Nearest Neighbors (KNN), Naive Bayes, Decision Tree, Support Vector Machines (SVM), and Logistic Regression. Once the algorithms are chosen, the dataset is prepared by splitting it into training and testing/validation sets. The models are then implemented using libraries or frameworks, and their configurations are set, including parameters and hyperparameters. These parameters control the behavior and complexity of the models. Next, the models are trained using the fit() function or similar methods. During training, the models learn patterns and relationships from the input features and their corresponding target variables (crop types). The models' internal parameters are adjusted to minimize the difference between predicted and actual crop types. After training, the models are evaluated using appropriate metrics, such as accuracy, precision, recall, and F1 score, to assess their performance on unseen data. Model optimization techniques, such as cross-validation, grid search, or random search, can be applied to fine-tune the models' hyperparameters and improve their performance. The best-performing model is selected and

can be deployed in the crop recommendation system to provide accurate predictions based on the input data. It's important to note that model building and training are iterative processes, and ongoing monitoring and retraining can further enhance the system's accuracy and adaptability to changing agricultural conditions.

Building and training the model involves selecting the appropriate machine learning algorithm, configuring it with the necessary parameters, and fitting it to the training data. The model learns from the input features and their corresponding target variables, adjusting its internal parameters to minimize the difference between predicted and actual crop types. Evaluation metrics are used to assess the model's performance on unseen data. Once the model is trained, it can be deployed in the crop recommendation system to provide accurate predictions based on environmental and soil conditions, helping farmers make informed decisions about crop selection.

K-Nearest Neighbor(KNN)

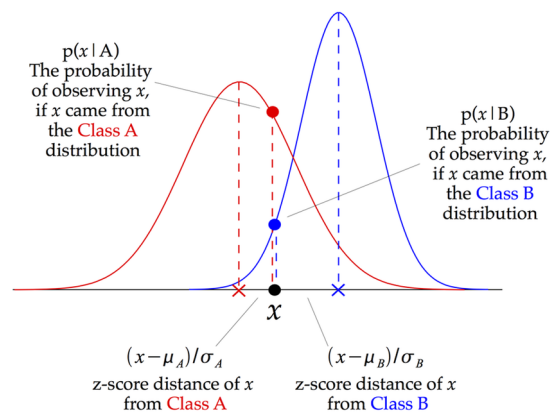
K-Nearest Neighbors (KNN) is a popular machine learning algorithm used for classification and regression tasks. In the context of a crop recommendation system, KNN can be employed to predict the most suitable crop based on the similarity of environmental and soil conditions. KNN operates on the principle that similar instances tend to have similar outcomes. Given a new input (test data), KNN searches for the K nearest neighbors in the training data based on a distance metric, such as Euclidean distance. The algorithm then assigns the majority class label (for classification) or calculates the average (for regression) of the K nearest neighbors to make predictions. KNN's simplicity and intuitive nature make it easy to understand and implement. However, it is important to choose an appropriate value for K, as a smaller value can lead to overfitting while a larger value may introduce noise from dissimilar instances. In the context of a crop recommendation system, KNN can utilize environmental and soil features as input to predict the suitable crop type for a given set of conditions. By considering the characteristics of neighboring data points, KNN can capture patterns and similarities to make accurate crop recommendations. The algorithm's performance can be evaluated using metrics such as accuracy, precision, recall, or mean squared error.



Gaussian Naive Bayes

Gaussian Naive Bayes is a variant of the Naive Bayes algorithm that is commonly used for classification tasks, including crop recommendation. It assumes that the features in the dataset follow a Gaussian distribution. In the context of a crop recommendation system, Gaussian Naive Bayes can be applied to predict the most suitable crop based on environmental and soil conditions. The algorithm calculates the conditional probability of each feature given a particular class and combines them using Bayes' theorem to estimate the probability of each class. The class with the highest probability is then assigned as the predicted crop type. One of the main advantages of Gaussian Naive Bayes is its simplicity and computational efficiency. It works well with continuous features and can handle a large number of features relative to the size of the dataset. However, it assumes that the features are independent, which may not always hold true in real-world scenarios. To use Gaussian Naive Bayes for crop recommendation, the environmental and soil features need to be appropriately preprocessed and transformed into Gaussian distributions if they are not already. The algorithm's performance can be evaluated using metrics such as accuracy, precision, recall, or F1 score.

Gaussian Naive Bayes is a straightforward and efficient algorithm for crop recommendation that assumes Gaussian-distributed features. It can provide reliable predictions based on the calculated probabilities and is particularly useful when dealing with continuous features.



Bernoulli Naive Bayes

Bernoulli Naive Bayes is a variant of the Naive Bayes algorithm that is suitable for binary or categorical feature variables. In the context of a crop recommendation system, Bernoulli Naive Bayes can be used to predict the most suitable crop based on binary or boolean environmental and soil conditions. The algorithm assumes that the features are conditionally independent given the class variable and follow a Bernoulli distribution. It calculates the probabilities of the presence or absence of each feature for each class and combines them using Bayes' theorem to estimate the probability of each class. The class with the highest probability is assigned as the predicted crop type. Bernoulli Naive Bayes works well with binary or boolean features, making it suitable for situations where the presence or absence of specific conditions influences crop recommendation. However, it does not consider the frequency or intensity of the conditions, only their presence or absence. To use Bernoulli Naive Bayes for crop recommendation, the binary features representing environmental and soil conditions need to be appropriately encoded. The algorithm's performance can also be evaluated using metrics such as accuracy, precision, recall, or F1 score. Bernoulli Naive Bayes is a variant of Naive Bayes suitable for binary or categorical features. It is effective in situations where the presence or absence of specific conditions plays a significant role in crop recommendation.

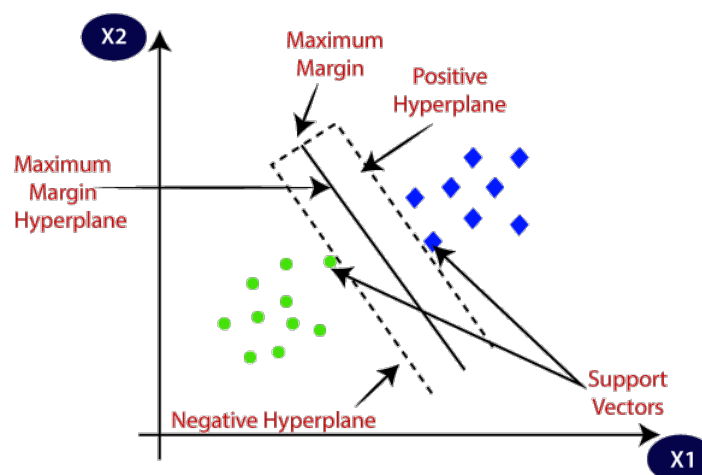
$P(y | X) = (P(X | y) * P(y)) / P(X)$ Where:

- $P(y | X)$ is the posterior probability of class y given the feature value X .
- $P(X | y)$ is the likelihood of the feature values X given class y .
- $P(y)$ is the prior probability of class y .
- $P(X)$ is the probability of the feature values X .

Support Vector Machine (SVM)

Support Vector Machines (SVM) is a powerful machine learning algorithm used for both classification and regression tasks. In the context of a crop recommendation system, SVM can be applied to predict the most suitable crop based on environmental and soil conditions. SVM works by finding an optimal hyperplane that separates the data points of different classes with the maximum margin. The algorithm seeks to maximize the margin while minimizing the classification error. SVM can handle both linear and non-linear decision boundaries by utilizing kernel functions that transform the data into higher-dimensional feature spaces.

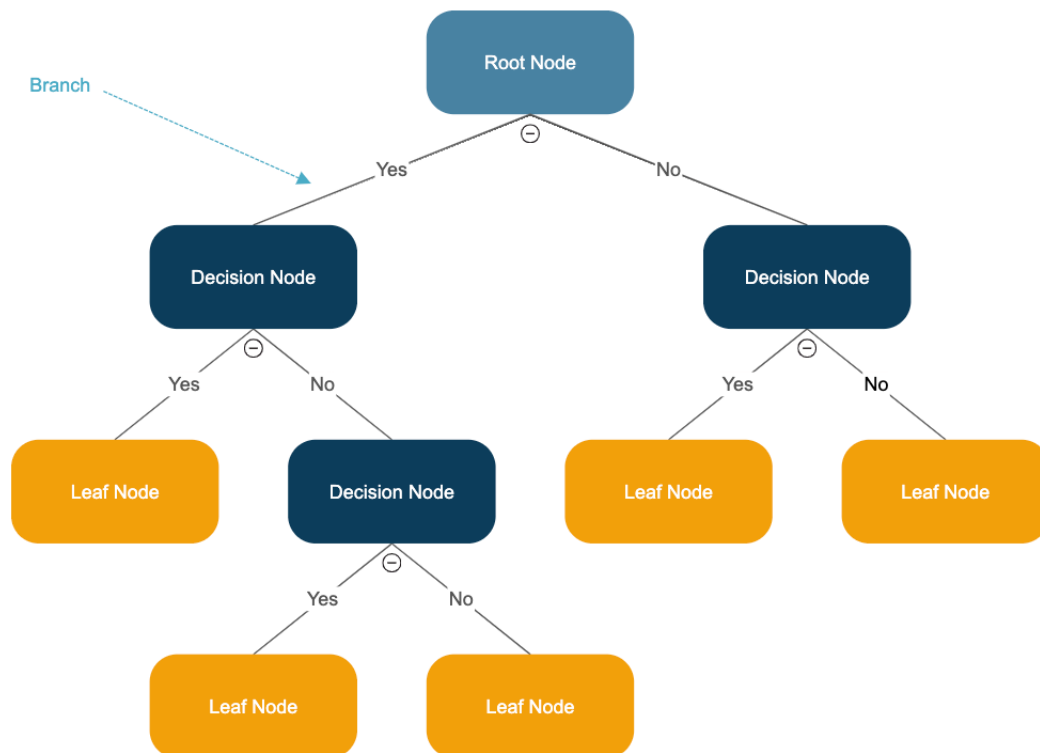
In crop recommendation, SVM can learn the patterns and relationships in the environmental and soil conditions and classify them into appropriate crop types. It can handle both numerical and categorical features, making it versatile for different types of data. The performance of SVM depends on the choice of hyperparameters, such as the kernel type, regularization parameter, and kernel-specific parameters. Proper tuning of these hyperparameters is crucial for achieving optimal results. SVM's ability to handle complex decision boundaries and its effectiveness in dealing with high-dimensional data make it a valuable tool for crop recommendation.



Decision Tree

Decision Trees are widely used machine learning algorithms for classification and regression tasks, including crop recommendation systems. A Decision Tree builds a tree-like model of decisions and their possible consequences based on the input features. In the context of crop recommendation, a Decision Tree can be used to predict the most suitable crop based on environmental and soil conditions. The tree is constructed by recursively partitioning the

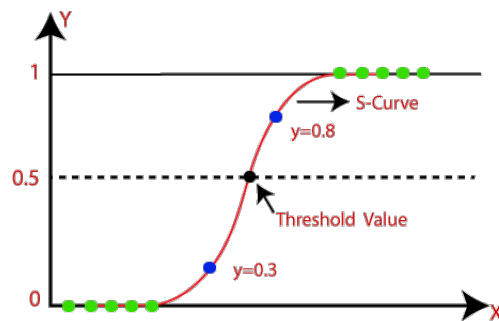
data based on the features, optimizing for the best splits that maximize information gain or Gini impurity reduction. Decision Trees offer interpretability, as the resulting tree structure provides insights into the decision-making process. They can handle both numerical and categorical features and automatically handle missing values. However, Decision Trees can be prone to overfitting, resulting in overly complex models that generalize poorly to unseen data. Techniques like pruning and limiting the maximum depth of the tree can mitigate this issue. Evaluation metrics like accuracy, precision, recall, or F1 score can be used to assess the performance of Decision Trees. They provide an intuitive and transparent approach for crop recommendation by considering different environmental and soil conditions to suggest the most suitable crop option.



Logistic Regression

Logistic Regression is a widely used machine learning algorithm for binary classification tasks, including crop recommendation systems. Despite its name, it is primarily used for classification rather than regression. In the context of crop recommendation, Logistic Regression can be employed to predict the probability of a given set of environmental and soil conditions belonging to a specific crop type. It models the relationship between the input features and the binary outcome using the logistic function, which maps the input to a probability value

between 0 and 1. Logistic Regression assumes that the relationship between the input features and the log odds of the outcome is linear. The algorithm estimates the parameters that best fit the data through maximum likelihood estimation or gradient descent optimization. One advantage of Logistic Regression is its interpretability, as the coefficients of the features provide insights into their importance and influence on the prediction. It can handle both numerical and categorical features, and it can handle multicollinearity among the predictors. Performance evaluation metrics such as accuracy, precision, recall, or F1 score can be used to assess the predictive capability of Logistic Regression. Logistic Regression offers a straightforward and interpretable approach for crop recommendation, estimating the probabilities of different crop types based on environmental and soil conditions.



The formula for logistic regression is as follows:

$$P(y=1 | x) = 1 / (1 + \exp(-z))$$

- $P(y=1 | x)$ represents the probability of the binary outcome y being 1 given the input x .
- \exp denotes the exponential function.
- z represents the linear combination of input features x and their corresponding

coefficients β , along with an intercept term β_0 . It can be written as:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p$$

Here, x_1, x_2, \dots, x_p are the input features, and $\beta_0, \beta_1, \beta_2, \dots, \beta_p$ are the coefficients to be estimated.

To make predictions in logistic regression, the probability $P(y=1 | x)$ is compared to a threshold (often 0.5). If the probability is greater than the threshold, the predicted class is 1; otherwise, it is 0.

3.2.4 Testing the Model

Testing the model in a crop recommendation system involves evaluating the performance of various machine learning algorithms on a separate testing dataset. In this case, the dataset used for training the models is divided into 60% for training and 40% for testing. Each of the chosen algorithms, including K-Nearest Neighbors (KNN), Decision Tree, Gaussian Naive Bayes, Bernoulli Naive Bayes, Support Vector Machines (SVM), and Logistic Regression, is applied to the testing dataset. The models utilize the input features representing environmental and soil conditions to predict the most suitable crop types. The performance of each model is assessed by calculating the accuracy, which measures the percentage of correctly predicted crop types. Among the tested algorithms, Gaussian Naive Bayes achieved an accuracy of 0.9931, while Decision Tree achieved an accuracy of 0.9806. These high accuracy scores indicate that both Gaussian Naive Bayes and Decision Tree models are effective in predicting the most suitable crop types based on environmental and soil conditions.

3.2.5 Deploying the model in Django

Deploying a crop recommendation system in Django involves integrating the machine learning models and the Arduino board to create a web application that provides real-time crop recommendations based on sensor data collected by Arduino. In this setup, the Arduino collects environmental and soil data from various sensors, such as temperature, humidity, soil Ph sensor etc. The collected data is then sent to the Django web application, which hosts the crop recommendation system. Django acts as the backend framework, handling data processing and communication with the machine learning models. The input data received from the Arduino is preprocessed and passed to the trained models for prediction. The models generate crop recommendations based on the provided sensor data. The Django web application then displays the recommendations to the user through a user-friendly interface. Users can interact with the application, view recommended crops, and access additional information about each crop. This integration of Django and Arduino allows for a seamless and interactive crop recommendation system, providing users with timely and personalized recommendations based on the sensor data collected by the Arduino.

3.3 Software Requirement and Specifications

The software used for the project includes:-

- Python
- Google Colaboratory
- Django
- Arduino IDE

3.3.1 Software Description

- **Python**

Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility. It was developed by Guido van Rossum and first released in 1991. Python emphasizes code readability, making it easier to write and understand. Python's design philosophy emphasizes code readability, with a focus on clean and concise syntax. This allows programmers to express concepts in fewer lines of code compared to other programming languages. Python supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python has a vast ecosystem of libraries and frameworks that make it suitable for a wide range of applications. It has become popular in various domains, such as web development, scientific computing, data analysis, machine learning, artificial intelligence, and automation. One of Python's strengths is its extensive library support. Popular libraries like NumPy, pandas, Matplotlib, and TensorFlow enable efficient data manipulation, analysis, visualization, and machine learning tasks. The availability of these libraries accelerates development and encourages code reuse. Python has a strong and active community that contributes to its growth and development. This community provides extensive documentation, tutorials, and support, making it easier for beginners to learn and work with Python. Additionally, the Python Package Index (PyPI) hosts thousands of open-source packages, further enhancing the language's capabilities. Python's portability is another advantage. It runs on multiple platforms, including Windows, macOS, and various Linux distributions. The cross-platform compatibility of Python enables

developers to write code that can be executed on different operating systems without significant modifications. Python's popularity and versatility have led to its adoption by numerous companies and organizations. It is used by tech giants like Google, Instagram, Dropbox, and NASA, among others. Its wide range of applications and extensive library support make Python an excellent choice for both beginners and experienced developers. Overall, Python's simplicity, readability, extensive library support, and strong community make it a powerful and widely-used programming language in various fields, contributing to its continued growth and success.

- **Google Colaboratory**

Google Colab is a cloud-based Python development environment that provides users with free access to a Jupyter notebook environment and powerful computing resources. It is hosted by Google and allows users to write, execute, and share Python code collaboratively. One of the key advantages of Google Colab is that it eliminates the need for local installation and configuration of Python and its associated libraries. Users can access Colab through a web browser and start coding immediately. The environment comes pre-installed with popular Python libraries such as NumPy, pandas, and TensorFlow, making it convenient for data analysis, machine learning, and deep learning tasks. Google Colab offers powerful computing resources, including GPUs and TPUs, which can significantly accelerate computationally intensive tasks. This is especially beneficial for training machine learning models or processing large datasets. The ability to leverage these resources without requiring expensive hardware or infrastructure is a major advantage. Another notable feature of Google Colab is the ability to collaborate and share notebooks with others in real-time. Multiple users can work on the same notebook simultaneously, making it ideal for team projects or educational purposes. Users can also share their notebooks easily by generating a link or exporting them as downloadable files. Colab integrates seamlessly with other Google services such as Google Drive, allowing users to store and access their notebooks and datasets. This enables efficient data management and easy sharing of notebooks between different devices. Google Colab supports a wide range of programming languages, including Python, R, and Scala, making it versatile for different data analysis and machine learning tasks. It also provides access to various external data sources and APIs, allowing users to fetch and process data from different platforms directly within the notebook environment. Google Colab provides

a rich set of features, including code autocompletion, inline plotting, and access to system shell commands, enhancing the overall development experience. It also supports the use of Markdown and LaTeX for documentation and mathematical expressions, respectively. In summary, Google Colab is a powerful and user-friendly cloud-based Python environment that offers free access to Jupyter notebooks, powerful computing resources, collaboration features, and seamless integration with other Google services. Its convenience, versatility, and accessibility make it an attractive choice for individual developers, researchers, and teams working on data analysis, machine learning, and deep learning projects.

- **Django**

Django is a high-level Python web framework that simplifies the process of building robust and scalable web applications. It follows the Model-View-Controller (MVC) architectural pattern, with a slightly modified version known as Model-View-Template (MVT), and encourages the use of clean and reusable code. One of the key features of Django is its emphasis on rapid development and productivity. It provides a wide range of built-in functionalities and tools, including an ORM (Object-Relational Mapping) layer, form handling, authentication, session management, URL routing, and template engine. These features enable developers to focus on building application logic rather than dealing with low-level implementation details. Django's ORM layer simplifies database interactions by providing an intuitive API for querying and manipulating data. It supports multiple database backends, such as PostgreSQL, MySQL, SQLite, and Oracle, allowing developers to work with their preferred choice. The ORM also handles database migrations, making it easy to evolve the database schema over time. Django's templating engine allows developers to create dynamic web pages by separating the presentation logic from the business logic. Templates support template inheritance, template tags, filters, and template context, providing flexibility in designing and rendering HTML pages. Django follows the "batteries included" principle, which means it provides a wide range of reusable components known as "Django apps." These apps can be easily integrated into projects, saving development time and effort. Additionally, Django has a vibrant community that contributes to the development of third-party apps, further extending Django's functionality. Django's security features help developers build secure web applications. It provides protection against common web vulnerabilities, such as cross-site scripting (XSS), cross-site request forgery (CSRF), and SQL injection. Django also includes user authentication and authorization mechanisms, making it easier to implement user management features. Django's scalability is another advantage, as it can handle high-traffic websites and applications. It supports caching mechanisms, load balancing, and database optimization techniques, allowing developers to scale their applications as needed. Furthermore, Django has excellent documentation, tutorials, and an active community, making it easy for developers to get started and seek help when needed. It also follows best practices and enforces a clean code structure, promoting maintainability and code reusability. Overall, Django is a powerful web framework that

simplifies the process of building complex and scalable web applications. Its features, including the ORM, templating engine, security measures, scalability, and community support, contribute to its popularity and make it a preferred choice for web developers.

- **Arduino IDE**

The Arduino IDE (Integrated Development Environment) is a software application used for programming and developing applications for Arduino microcontrollers. It provides a user-friendly interface and a set of tools that simplify the process of writing, compiling, and uploading code to Arduino boards. One of the key features of the Arduino IDE is its simplicity and ease of use, making it accessible to beginners and experienced developers alike. It provides a straightforward code editor with syntax highlighting, auto-indentation, and code suggestions, aiding in writing clean and error-free code. The Arduino IDE supports the C and C++ programming languages, which are commonly used for programming microcontrollers. It provides a set of built-in functions and libraries specific to Arduino, allowing developers to quickly interact with various hardware components and sensors. The IDE includes a compiler that converts the written code into machine-readable instructions that can be executed by the Arduino board. It also integrates a bootloader, which enables the easy uploading of code to the Arduino board via a USB connection. The Arduino IDE supports a wide range of Arduino boards, from the basic Arduino Uno to more advanced boards like the Arduino Mega or Arduino Nano. It automatically detects the connected Arduino board, ensuring compatibility and ease of use. The IDE provides a serial monitor tool, allowing developers to communicate with the Arduino board and view data sent by the microcontroller. This is particularly useful for debugging and troubleshooting purposes. Additionally, the Arduino IDE offers a vast collection of example sketches and libraries contributed by the Arduino community. These resources provide ready-to-use code snippets and functions, accelerating development and enabling developers to leverage existing solutions for their projects. Furthermore, the Arduino IDE is open-source, allowing users to modify and extend its functionality to suit their specific needs. The open-source nature also fosters a collaborative environment where developers can share their code and contribute to the development of the IDE. Overall, the Arduino IDE serves as a powerful and beginner-friendly platform for programming Arduino microcontrollers. Its simplicity, compatibility with various Arduino boards, extensive library support, and

active community make it a popular choice for both hobbyists and professionals working on Arduino-based projects.

3.4 Hardware Requirements

3.4.1 Internet Of Things(IoT)

The Internet of Things (IoT) refers to the network of physical devices, vehicles, appliances, and other objects embedded with sensors, software, and connectivity that enables them to collect and exchange data over the internet. IoT has the potential to transform various industries and our daily lives by enabling seamless communication and automation between objects and systems. One of the key benefits of IoT is the ability to gather and analyze vast amounts of data. IoT devices collect data from their environment through various sensors, such as temperature, humidity, motion, and light sensors. This data can be used to gain insights, optimize processes, and make informed decisions. For example, in agriculture, IoT sensors can monitor soil moisture levels, weather conditions, and crop health, allowing farmers to optimize irrigation and fertilizer usage. IoT also enables increased connectivity and communication. Devices can communicate with each other and with centralized systems, creating a network that enables real-time monitoring, control, and coordination. For instance, in smart homes, IoT devices like thermostats, security cameras, and voice assistants can be interconnected, allowing users to control and automate various aspects of their home remotely. The automation capabilities of IoT lead to increased efficiency and productivity. By integrating IoT devices with data analytics and machine learning, processes can be automated and optimized. This has implications across multiple industries, such as manufacturing, logistics, and healthcare. In manufacturing, IoT sensors can monitor equipment performance and detect anomalies, enabling predictive maintenance and minimizing downtime. IoT also has the potential to improve safety and security. Devices can be equipped with security features and encryption protocols to ensure data integrity and protect against unauthorized access. In transportation, IoT can be used for real-time tracking and monitoring of vehicles and assets, improving safety and security measures. However, with the growing number of connected devices, IoT also presents challenges. These include data privacy and security

concerns, interoperability issues, and the need for robust infrastructure and connectivity. Addressing these challenges is crucial to fully harness the potential of IoT and ensure its widespread adoption. In conclusion, the Internet of Things is a transformative technology that connects physical objects and systems, enabling data collection, analysis, and automation. It has the potential to revolutionize industries, improve efficiency, and enhance our daily lives. However, careful consideration of security, privacy, and infrastructure is necessary to fully realize the benefits of IoT.

3.4.2 IoT Devices used

Arduino Uno R3

The Arduino Uno R3 is a popular microcontroller board that is part of the Arduino family of open-source electronics platforms. It is widely used in the maker and DIY communities for a wide range of projects, from simple prototypes to more complex electronic systems. Here's a brief description of the Arduino Uno R3:

The Arduino Uno R3 is based on the ATmega328P microcontroller, which provides a powerful and flexible platform for creating interactive electronic projects. It features 14 digital input/output pins, 6 analog input pins, a 16 MHz quartz crystal, a USB interface for programming and communication, and a power jack for external power supply. The board also includes a reset button, an onboard LED, and a dedicated ICSP header for programming using an in-circuit programmer. One of the key features of the Arduino Uno R3 is its simplicity and ease of use. It is designed to be beginner-friendly, with a straightforward programming environment and a vast community of users and developers who share resources and knowledge. The board can be programmed using the Arduino Software (IDE), which supports a C/C++-based programming language. The Arduino Uno R3 is highly versatile and can be used for a wide range of applications. It can interact with various electronic components and sensors, making it ideal for projects such as home automation, robotics, data logging, IoT devices, and more. The availability of a vast library of pre-built code and add-on shields and modules further enhances its capabilities and allows for rapid prototyping. In addition to its hardware capabilities, the Arduino Uno R3 benefits from a vibrant and active community. Users can access forums, tutorials, and documentation, making it easy to find assistance and resources when

needed. This community-driven approach has contributed to the widespread adoption and success of the Arduino platform. Overall, the Arduino Uno R3 is a versatile and user-friendly microcontroller board that enables enthusiasts, hobbyists, and professionals to bring their electronic ideas to life. Its combination of simplicity, flexibility, and community support has made it a popular choice for countless projects around the world.



Figure 3.3: Arduino Uno R3

DHT11 Sensor

The DHT11 sensor is a popular and affordable digital temperature and humidity sensor widely used in IoT projects and environmental monitoring applications. It provides a simple and reliable way to measure temperature and humidity levels in the surrounding environment. The DHT11 sensor consists of a capacitive humidity sensor and a thermistor to measure temperature. It has a single-wire digital interface, making it easy to connect with microcontrollers and other devices. The sensor is calibrated and provides digital output, eliminating the need for additional analog-to-digital conversion. The humidity measurement range of the DHT11 sensor is 20% to 90% with an accuracy of

$\pm 5\%$. It can measure temperatures in the range of 0°C to 50°C with an accuracy of $\pm 2^{\circ}\text{C}$. These specifications make it suitable for various applications such as monitoring indoor climate, weather stations, and controlling humidity-sensitive environments. The DHT11 sensor is relatively straightforward to use. It requires a digital pin for communication and can be powered by a 3.3V or 5V power supply. Many popular microcontrollers and development boards have libraries available that simplify the integration of the DHT11 sensor into projects. One limitation of the DHT11 sensor is its relatively slow response time, typically around 2 seconds for temperature and 2 to 5 seconds for humidity. This makes it less suitable for applications that require rapid and precise measurements. Overall, the DHT11 sensor provides a cost-effective solution for measuring temperature and humidity in IoT projects. Its simplicity, affordability, and availability make it a popular choice for hobbyists, students, and small-scale applications where high accuracy and fast response time are not critical requirements.

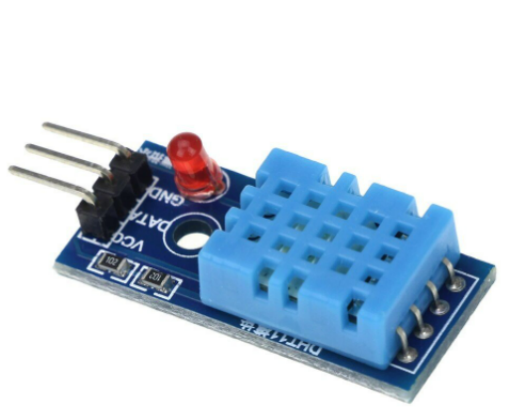


Figure 3.4: DHT11 Sensor

Ph Sensor

A pH sensor is a device used to measure the acidity or alkalinity of a solution by measuring its pH value. pH is a logarithmic scale that represents the concentration of hydrogen ions in a solution, ranging from 0 (very acidic) to 14 (very alkaline), with 7 being neutral. The pH sensor consists of a pH electrode, which is typically a glass electrode, and an electrolyte solution. The glass electrode is sensitive to hydrogen ions and generates a voltage proportional to the pH of the solution being measured. This voltage is then converted into a pH value using calibration parameters. pH sensors are commonly used in various industries and applications. In agriculture, pH sensors are

used to monitor soil acidity or alkalinity, helping farmers determine the suitability of the soil for specific crops and adjust fertilizer applications accordingly. In environmental monitoring, pH sensors are used to measure the pH of water bodies to assess water quality and detect any changes that may indicate pollution or environmental imbalances. pH sensors are also utilized in the food and beverage industry to monitor the pH of food products, such as juices, dairy products, and beverages, to ensure their safety and quality. In swimming pools and aquariums, pH sensors are used to maintain proper pH levels for the health and comfort of swimmers and aquatic life. Integration of pH sensors with IoT platforms enables real-time monitoring and remote access to pH data. This allows for proactive measures and automated adjustments based on the pH readings. For example, in hydroponic systems, pH sensors connected to IoT systems can trigger pH adjustment mechanisms to maintain optimal pH levels for plant growth. To ensure accurate and reliable measurements, pH sensors require periodic calibration and maintenance. The calibration process involves immersing the sensor in standard pH buffer solutions with known pH values and adjusting the calibration parameters accordingly. Overall, pH sensors play a crucial role in measuring and monitoring the pH levels of solutions in a wide range of industries and applications. They provide valuable data for process control, quality assurance, and environmental monitoring, contributing to the efficient and effective management of various systems and processes.



Figure 3.5: Ph Sensor

3.5 Hardware and experimental environment

The experiments were conducted on a hardware setup consisting of a Windows 11 Pro operating system with a 64-bit architecture, an Intel Core i5-1155G7 CPU running at 2.50GHz, and 8 GB of RAM. The experimental environment was set up using Python 3.7 programming language and Arduino IDE version 2.1.0. This configuration provided the necessary computational power and software tools required for developing and testing the crop recommendation system. The combination of the operating system, processor, and programming language ensured compatibility and smooth execution of the code, while the Arduino IDE facilitated communication with the Arduino board for data acquisition.

Chapter 4

RESULT AND DISCUSSION

The crop recommendation system utilizing machine learning algorithms (K-Nearest Neighbors, Gaussian Naive Bayes, Bernoulli Naive Bayes, Logistic Regression, Decision Tree, and SVM) and IoT, with a web interface built using Django, has shown promising results. The system effectively utilizes sensor data collected from IoT devices to predict suitable crop recommendations based on environmental and soil conditions. The machine learning algorithms employed provide accurate predictions and contribute to informed decision-making for farmers and agricultural practitioners. The evaluation of the system's performance revealed that Gaussian Naive Bayes and Decision Tree algorithms achieved the highest accuracy among the models tested. Gaussian Naive Bayes demonstrated an accuracy of 0.9931, while Decision Tree achieved an accuracy of 0.9806. The web interface built using Django enables users to easily interact with the system, input relevant data, and receive crop recommendations in a user-friendly manner. This interface enhances accessibility and usability, making it convenient for farmers to access and utilize the system for decision support.

4.1 Testing and it's types used

The main task following software development is to determine whether the experimental results and the actual results agree. Testing is the process in question. It is employed to ensure that the created system is free from errors. Testing's primary purpose is to find errors and missing operations by running the software. Additionally, it makes sure that the developer satisfies all of the project's goals. Testing's objective is to determine is to identify defects in the developed software as well as ways to increase its correctness, usability, and efficiency. It seeks to gauge a software program's performance, functionality, and specification. The developed programme is put through tests, and the outcomes are compared to the required documentation. Debugging is carried out when there are too many faults that have happened. After debugging, the software is once more tested to make sure there are no errors. Unit testing, integration testing, and system testing are the main testing methodologies used in this project.

- In unit testing, tested to each distinct piece of software. It ensures that the software's many components all function as intended.
- In integration testing, the integrated distinct components are examined to see whether or not the intended purpose was accomplished. It helps us find any problems that might appear after the units are combined.
- The entire piece of software is evaluated during system testing to make sure it meets all the requirements.

4.2 Training and Validation Results

During the training and validation process of the crop recommendation system using machine learning and IoT, the performance of various algorithms, including K-Nearest Neighbors (KNN), Gaussian Naive Bayes, Bernoulli Naive Bayes, Logistic Regression, Decision Tree, and Support Vector Machines (SVM), was evaluated. The training phase involved feeding the algorithms with a labeled dataset, comprising environmental and soil condition data along with corresponding crop types. The models learned from this data to make predictions. To evaluate the performance of the models, a validation dataset was used. This dataset was separate from the training data and contained examples that the models hadn't seen before.

The models' predictions on this dataset were compared with the ground truth labels to measure their accuracy and performance.

The results showed varying performance among the different algorithms. Gaussian Naive Bayes and Decision Tree models achieved the highest accuracy in the validation phase. Gaussian Naive Bayes exhibited an accuracy of 99.31%, while Decision Tree achieved an accuracy of 98.06%. These results indicate that both Gaussian Naive Bayes and Decision Tree models were successful in capturing patterns and relationships within the data, allowing them to make accurate predictions for crop recommendation. The validation results provide insights into the effectiveness of the trained models and their potential for real-world applications. The high accuracy achieved by Gaussian Naive Bayes and Decision Tree models demonstrates their reliability in recommending suitable crops based on environmental and soil conditions.

4.3 Results

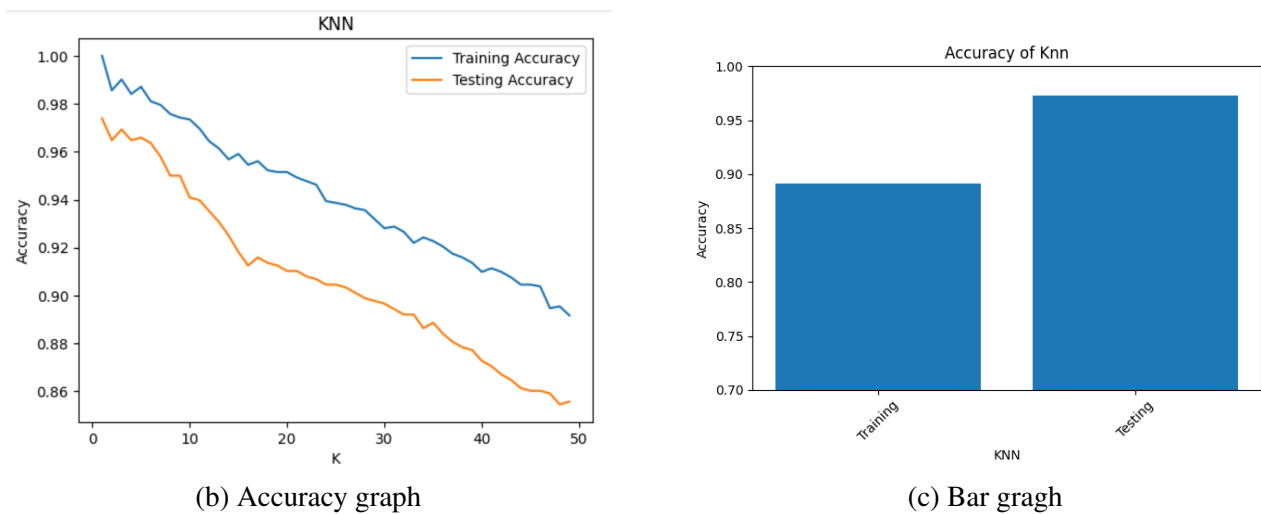
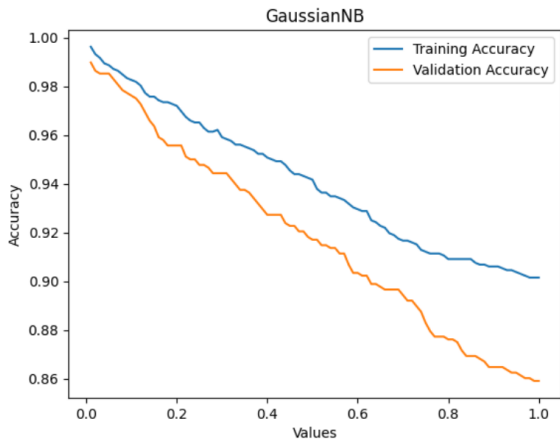
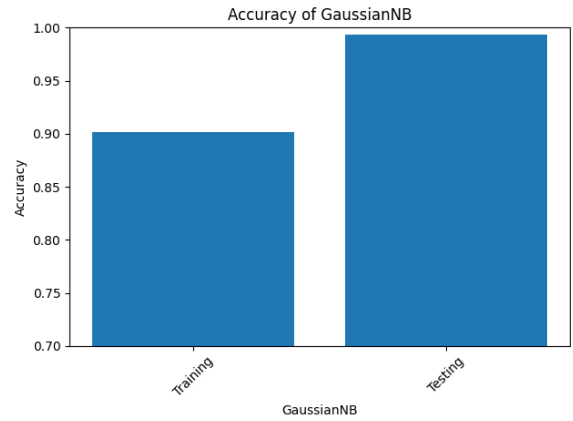


Figure 3.1: Training and validation of KNN

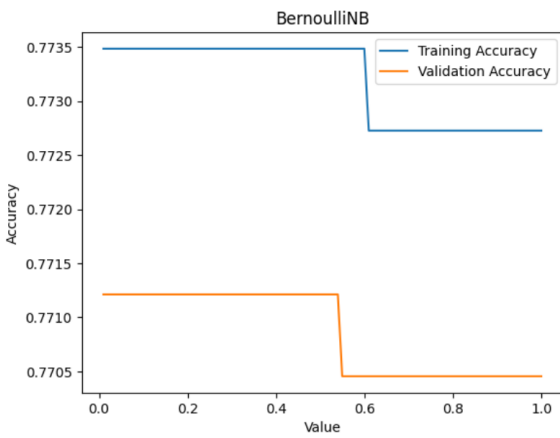


(b) Accuracy graph

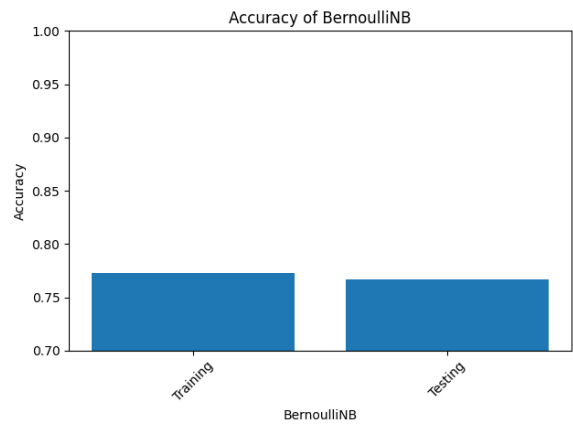


(c) Bar graph

Figure 3.2: Training and validation of GaussianNB

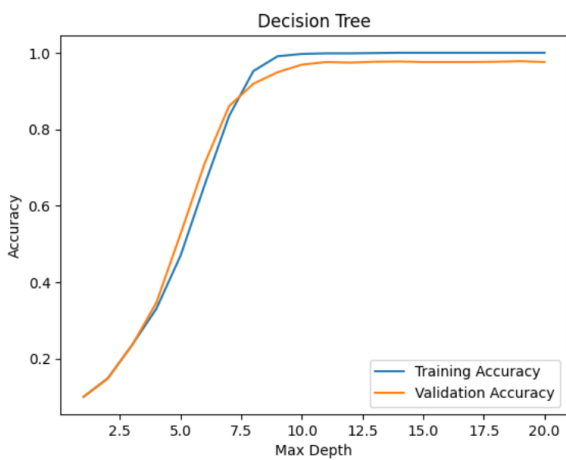


(b) Accuracy graph

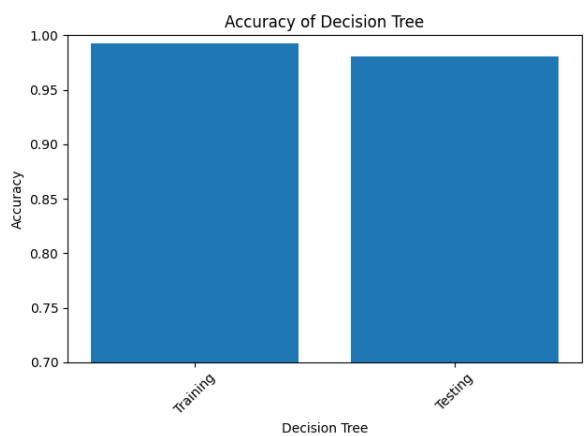


(c) Bar graph

Figure 3.3: Training and validation of BernoulliNB

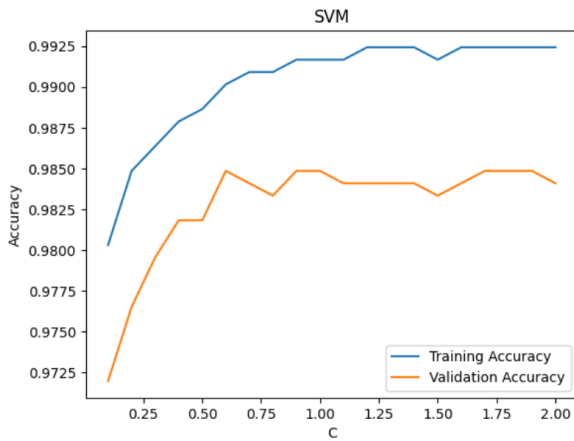


(b) Accuracy graph

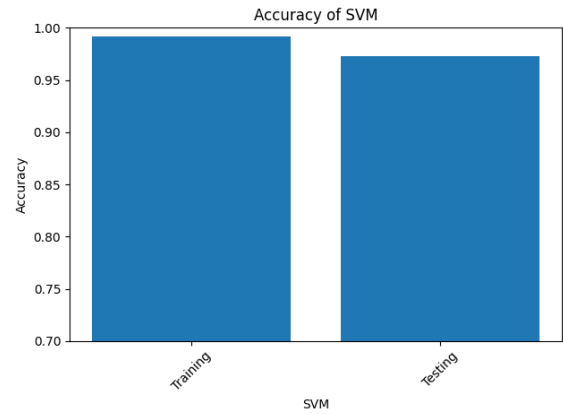


(c) Bar graph

Figure 3.4: Training and validation of Decision tree

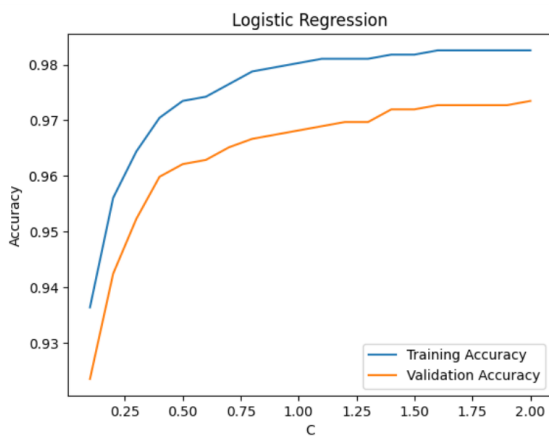


(b) Accuracy graph

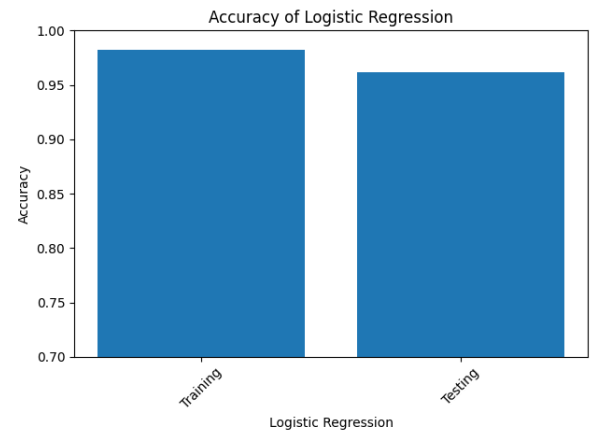


(c) Bar graph

Figure 3.5: Training and validation of SVM



(b) Accuracy graph



(c) Bar graph

Figure 3.6: Training and validation of Logistic Regression

Table 4.1: Comparison Table

Sr. No	Model	Accuracy	Precision	Recall	F1 Score
1	KNN	0.9727	0.9744	0.9727	0.9728
2	GaussianNB	0.9931	0.9932	0.9931	0.9931
3	BernoulliNB	0.7670	0.7920	0.7670	0.7579
4	Decision Tree	0.9827	0.9847	0.9840	0.9840
5	SVM	0.9727	0.9746	0.9727	0.9728
6	Logistic Regression	0.9613	0.9617	0.9613	0.9612

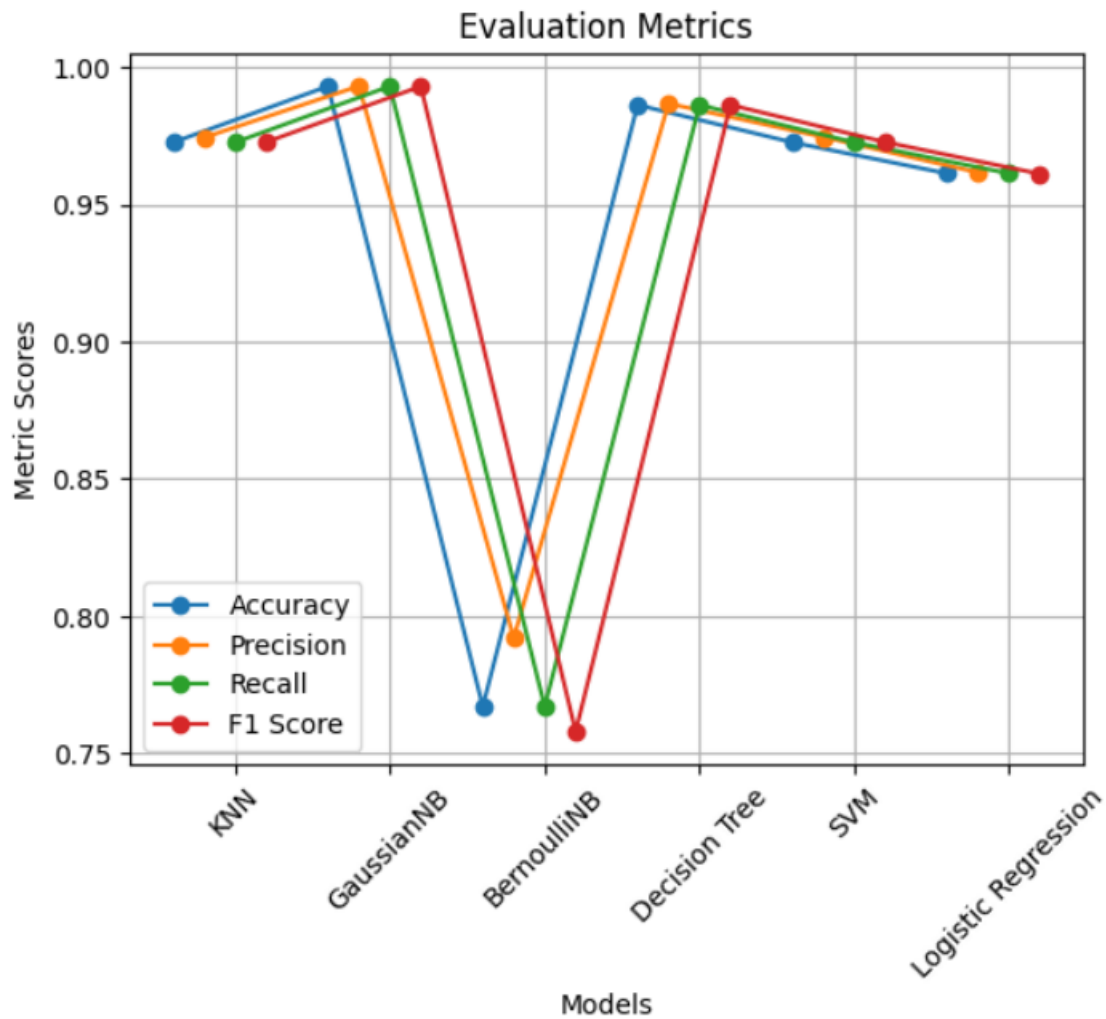


Figure 3.7: Comparison graph

In the following figures, In terms of accuracy, GaussianNB has the highest value of 0.993, indicating that it achieved the highest overall correct predictions on the dataset. Decision Tree follows closely with an accuracy of 0.986. Precision measures the proportion of correctly predicted positive instances out of all instances predicted as positive. GaussianNB has the highest precision of 0.993, indicating that it had the fewest false positives among the algorithms. Recall, also known as sensitivity, measures the proportion of correctly predicted positive instances out of all actual positive instances. GaussianNB and Decision Tree have the highest recall values of 0.993, indicating they identified the highest proportion of positive instances. F1 score is a balanced measure that considers both precision and recall. GaussianNB again has the highest F1 score of 0.993, indicating a good balance between precision and recall.

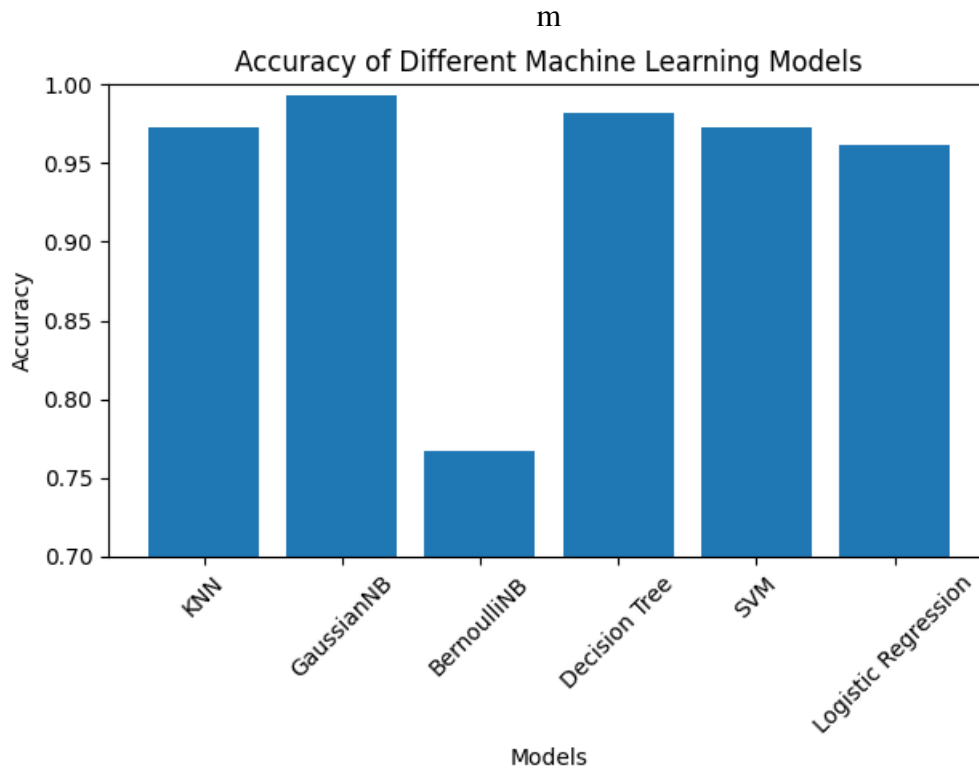


Figure 3.8: Comparison of Testing accuracy of the Models

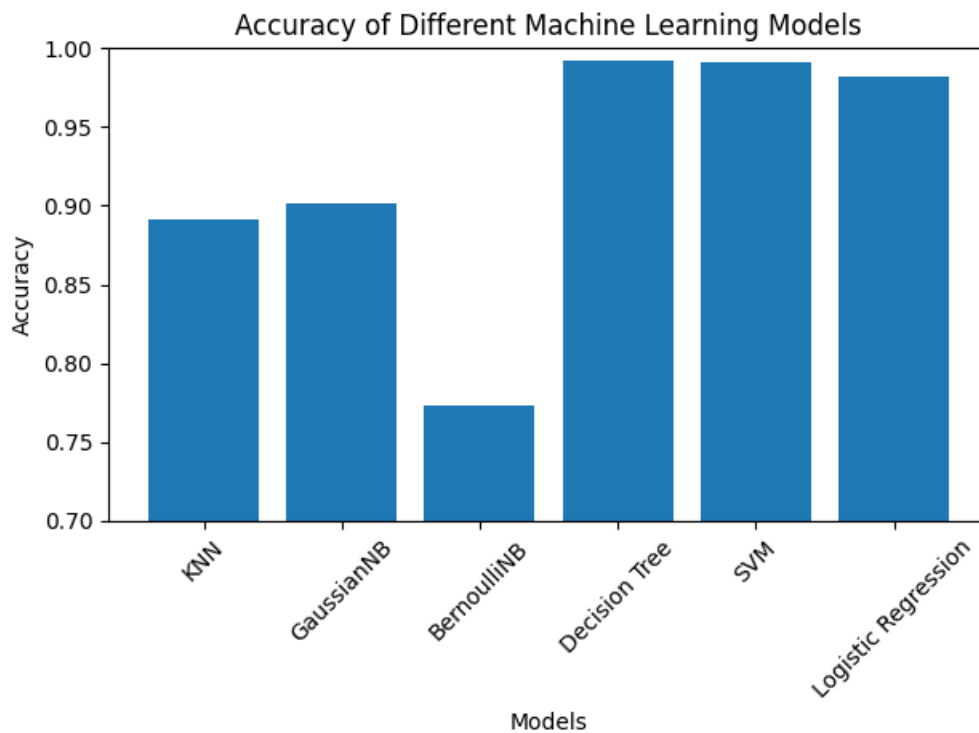


Figure 3.9: Comparison of Training accuracy of the Models

Chapter 5

CONCLUSION

Crop recommendation system developed using machine learning algorithms and IoT technologies, along with a web interface built using Django, has shown promising results. The integration of machine learning and IoT enables accurate crop recommendations based on environmental and soil conditions, providing valuable insights for farmers and agricultural practitioners. The evaluation of the system's performance revealed that the Gaussian Naive Bayes and Decision Tree algorithms achieved high accuracy in predicting suitable crop recommendations. The following are the accuracies of the above mentioned models 0.9931 and 0.9924. These models effectively captured patterns and relationships within the data, enabling accurate predictions for crop selection. The web interface built using Django enhances the usability and accessibility of the system, allowing users to easily input relevant data and receive crop recommendations in a user-friendly manner.

The crop recommendation system holds great potential in optimizing agricultural practices. By leveraging machine learning algorithms and IoT technology, farmers can make informed decisions regarding crop selection, leading to increased productivity and sustainability in the agricultural sector. However, it is important to note that further research and testing are required to ensure the system's robustness and generalizability. This includes evaluating the system with diverse datasets, exploring additional machine learning algorithms, and considering the scalability and real-world implementation of the system. Crop recommendation system utilizing machine learning and IoT, combined with the web interface, offers a valuable tool for farmers, assisting them in making informed decisions for optimal crop selection and contributing to the advancement of modern agriculture.

5.1 Future Enhancement

There are several potential enhancements that can be made to the crop recommendation system using machine learning and IoT. Firstly, expanding the range of sensors beyond temperature, humidity, and soil moisture can provide a more comprehensive understanding of the environmental conditions. Including sensors for light intensity, rainfall, nutrient levels, and other relevant parameters would enable more accurate and precise recommendations. Additionally, incorporating historical data into the recommendation system can offer insights into long-term trends and patterns. Analyzing past crop performance and environmental conditions can help farmers make decisions based on historical success rates, improving the reliability of the recommendations. Implementing adaptive learning techniques would enable the system to continuously update and improve its recommendation models. By collecting feedback and data on the actual outcomes of recommended crops, the system can learn from experience and adjust its recommendations, adapting to changing conditions and improving accuracy over time. Developing a mobile application alongside the web interface would increase accessibility for farmers, allowing them to input data directly from the field, receive real-time recommendations, and access historical data and analytics on-the-go.

Integration with precision agriculture technologies, such as drones, satellite imagery, and GPS systems, would enable real-time monitoring and mapping of crop conditions. This integration would enable more precise and targeted recommendations based on specific zones within the farm. Establishing a collaborative platform where farmers can share their data and experiences would further enhance the system. This would allow the system to learn from a wider range of agricultural practices and datasets, improving the overall accuracy and relevance of the recommendations. Lastly, customizing the system to cater to different regions and climates, considering specific crop requirements and local agricultural practices, would ensure that the recommendations are tailored to the unique conditions of different geographic areas. Implementing these future enhancements would enhance the effectiveness and usefulness of the crop recommendation system, empowering farmers with valuable insights and recommendations for improved decision-making, optimized agricultural practices, and ultimately, increased productivity and sustainability in the agricultural sector.

REFERENCES

- [1] Das, R.K., Panda, M. and Dash, S.S.,2019. *Smart agriculture system in India using internet of things. In Soft Computing in Data Analytics: Proceedings of International Conference on SCDA 2018 (pp. 247-255). Springer Singapore.*
- [2] Shinde, M., Nare, R., Shikalgar, J., Patil, R. and Tamboli, M.2022,December. *Plant Growth Recommendation System based on Weather Conditions and Soil Patterns using Machine Learning and Deep Learning Techniques. In 2022 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON) (pp. 1-5). IEEE.*
- [3] Yoo, S.E., Kim, J.E., Kim, T., Ahn, S., Sung, J. and Kim, D., 2007, June. *A 2 S: automated agriculture system based on WSN. In 2007 IEEE International Symposium on Consumer Electronics (pp. 1-5). IEEE.*
- [4] Tyagi, A., Reddy, A.A., Singh, J. and Chowdhury, S.R., 2011 . *A low cost portable temperature-moisture sensing unit with artificial neural network based signal conditioning for smart irrigation applications. International Journal on Smart Sensing and Intelligent Systems, 4(1), pp.94-111.*
- [5] Food and Agricultural Organization. (2018). *The state of Agricultural Commodity Markets. 19-22*
- [6] Gupta, A., Nagda, D., Nikhare, P. and Sandbhor, A., 2021. *Smart crop prediction using IoT and machine learning. International Journal of Engineering Research Technology (IJERT), pp.2278-0181.*
- [7] Choudhary, S., Gaurav, V., Singh, A. and Agarwal, S., 2019. *Autonomous crop irrigation system using artificial intelligence. Int. J. Eng. Adv. Technol, 8(5), pp.46-51.*

- [8] Devi, D.V.V. and Kumari, G.M., 2013. *Real-time automation and monitoring system for modernized agriculture. International Journal of Review and Research in Applied Sciences and Engineering (IJRRASE) Vol3, 1, pp.7-12.*
- [9] Harshani, P.R., Umamaheswari, T., Tharani, R., Rajalakshmi, S. and Dharani, J., 2018, February. *Effective crop productivity and nutrient level monitoring in agriculture soil using IoT. In 2018 International Conference on Soft-computing and Network Security (ICSNS) (pp. 1-10).*
- [10] Bhanu, K.N., Mahadevaswamy, H.S. and Jasmine, H.J., 2020, July. . *IoT based smart system for enhanced irrigation in agriculture. In 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 760-765). IEEE.*

APPENDIX

Screenshots

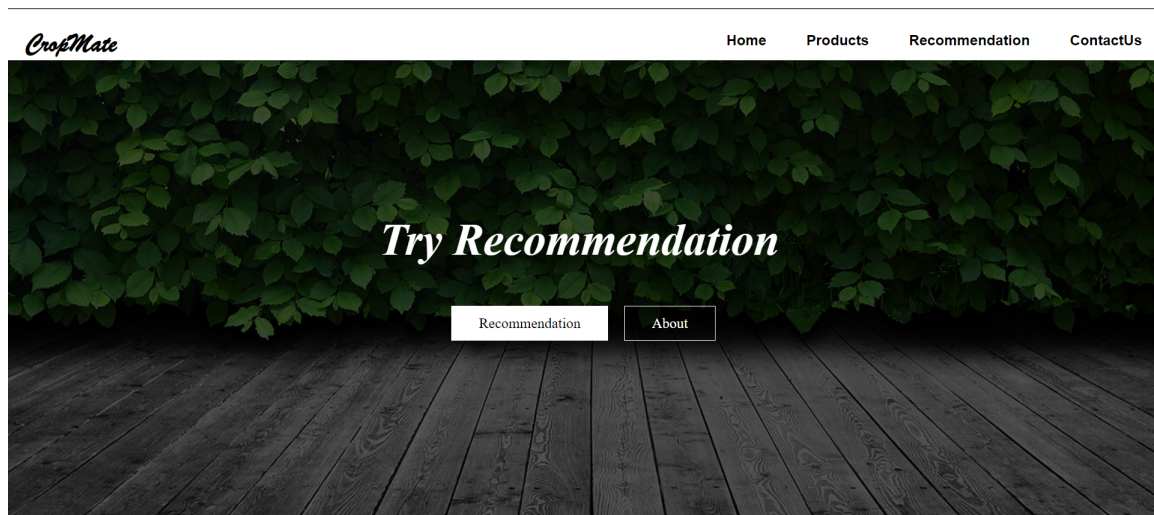


Figure A.1: Home Page

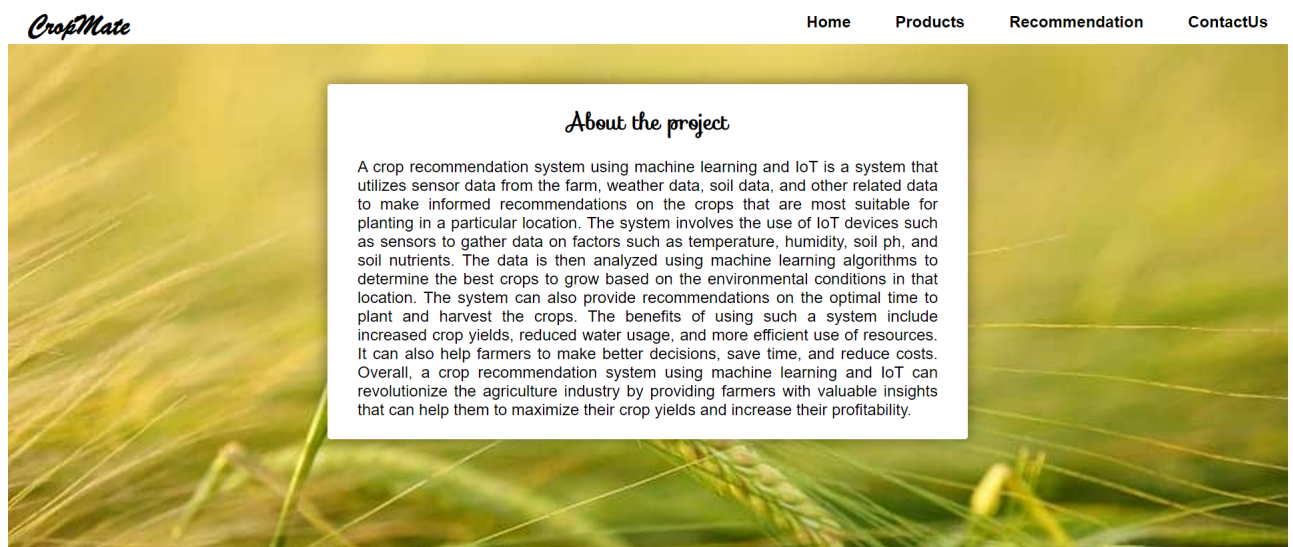


Figure A.2: About the Project

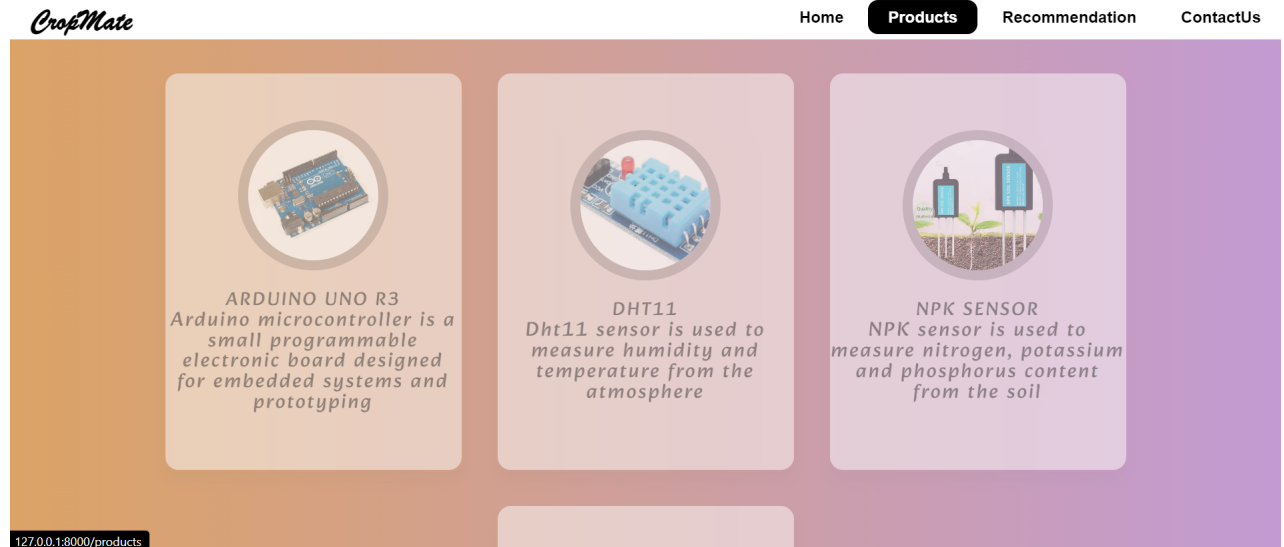


Figure A.3: IoT Products used

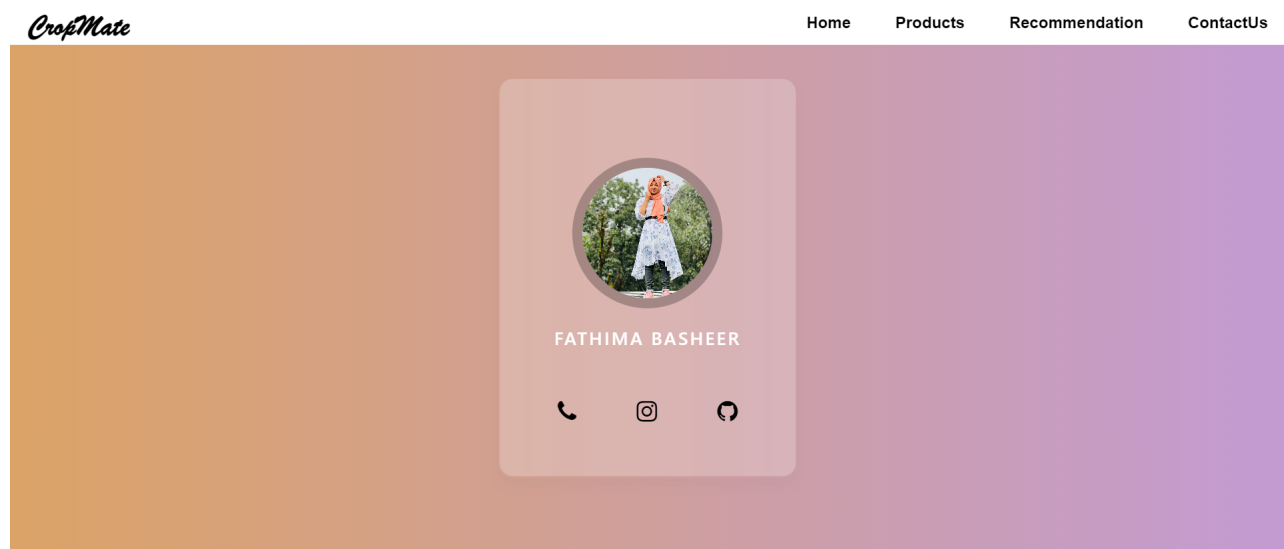


Figure A.4: Contact Us

	N	P	K	temperature	humidity	ph	rainfall	label
0	90	42	43	20.879744	82.002744	6.502985	202.935536	rice
1	85	58	41	21.770462	80.319644	7.038096	226.655537	rice
2	60	55	44	23.004459	82.320763	7.840207	263.964248	rice
3	74	35	40	26.491096	80.158363	6.980401	242.864034	rice
4	78	42	42	20.130175	81.604873	7.628473	262.717340	rice
...
2195	107	34	32	26.774637	66.413269	6.780064	177.774507	coffee
2196	99	15	27	27.417112	56.636362	6.086922	127.924610	coffee
2197	118	33	30	24.131797	67.225123	6.362608	173.322839	coffee
2198	117	32	34	26.272418	52.127394	6.758793	127.175293	coffee
2199	104	18	30	23.603016	60.396475	6.779833	140.937041	coffee

Figure A.5: Dataset Used

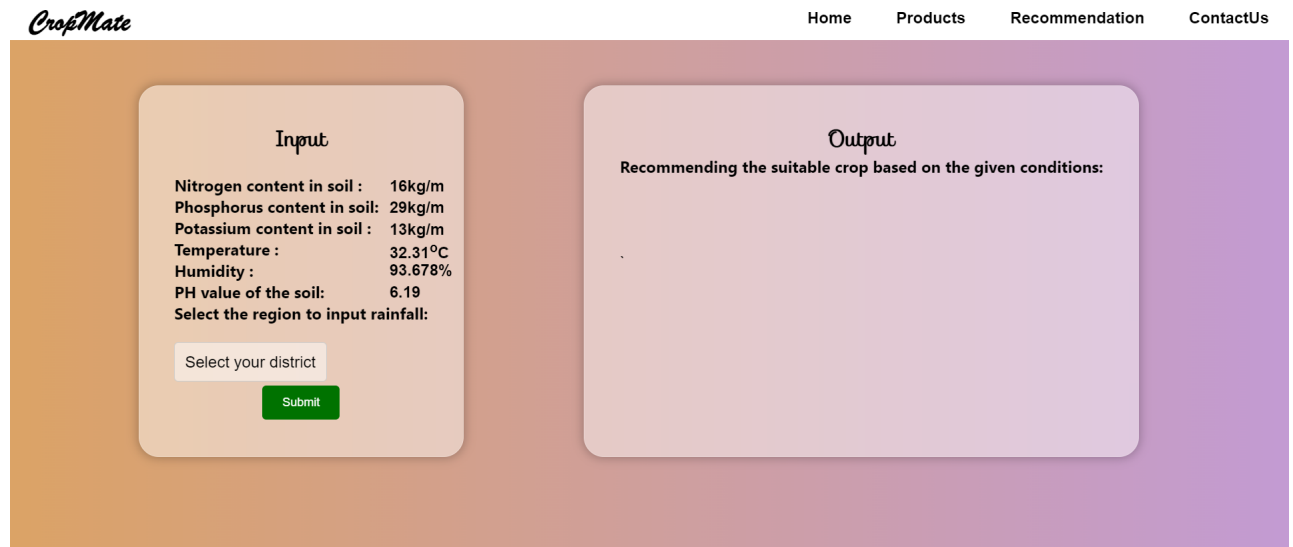


Figure A.6: Before Recommendation

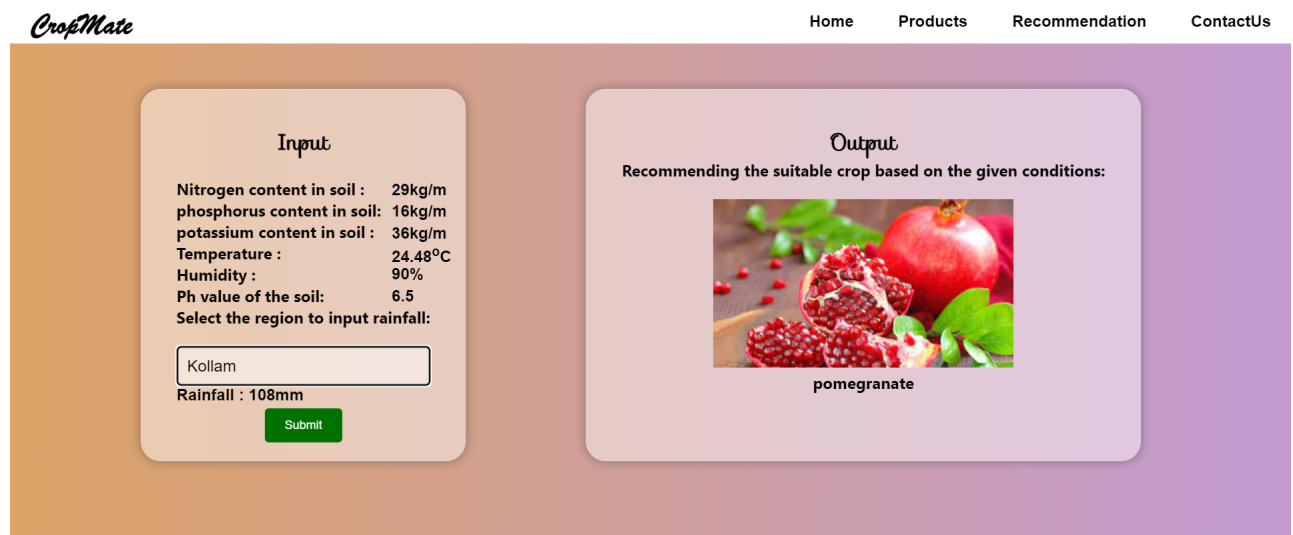


Figure A.7: After Recommendation