INFORMATICS INSTITUTE OF TECHNOLOGY

In Collaboration with

UNIVERSITY OF WESTMINSTER (UOW)

BEng (Hons) in Software Engineering

EcoGrow : Smart Farming App for Climate Informed Crop Selection Using Machine Learning

Final Year Project Thesis by

A.E.W Jayatilake

w1761374 /2019530

Supervised by

Mrs. Malithi Mithsara

Submitted in partial fulfillment of the requirements for the BEng (Hons) in Software Engineering degree at the University of Westminster

September 2024

Declaration

I hereby declare that my final year project is my original work, and it fulfills a portion of the requirements for the BEng (Hons) in Software Engineering degree. I have cited all of the sources I used to obtain ideas, facts, and non-original language in my dissertation's references section. I additionally declare that no other university has received this work for consideration of any other degree.

Name of the Student : A.E.W Jayatilake

Registration Number : w1761374 /2019530

Signature:  Date:27/03/2024

Abstract

My thesis focuses on developing a farmer-centric application aimed at providing crucial weather information and crop recommendations to optimize agricultural practices. By incorporating real-time weather data, and location-specific tracking, the app aims to empower farmers with valuable insights for decision-making. The integration of a chatbot further enhances user interaction, offering solutions to agricultural queries. The system relies on a comprehensive dataset for weather conditions and crop information.

The agricultural sector faces numerous challenges, including unpredictable weather conditions and the need for precise crop planning. Recognizing these issues, my project aims to create a farmer-centric app addressing the lack of easily accessible, real-time, and location-specific agricultural information. By offering comprehensive weather insights and crop recommendations, solution seeks to empower farmers and enhance the efficiency of their decision-making processes.

To tackle the identified problems, methodology revolves around the development of a user-friendly mobile application. I leverage a dataset encompassing various weather parameters, including temperature, rainfall and humidity. The app dynamically displays current weather conditions upon user login and allows users to track specific locations for detailed forecasts. Crop recommendations are categorized as short-term, mid-term, and long-term based on the gathered data. The incorporation of a chatbot enriches user experience, providing instant solutions to agricultural queries.

The prototype of farmer-centric app demonstrates promising results. Users can seamlessly access real-time weather conditions, empowering them to make informed decisions about crop selection and cultivation practices. The location-specific tracking feature ensures personalized recommendations, enhancing the adaptability of the system. The chatbot has proven effective in providing quick and relevant solutions to users' agricultural queries, contributing to a comprehensive and user-friendly platform.

Keywords

- Weather Conditions: The state of the atmosphere, including temperature, humidity, precipitation, and wind.
- Crop Recommendations: Suggestions for suitable crops based on climate, and environmental conditions.
- Mobile Application: Software designed for mobile devices, enabling access to specific functionalities or information.
- Location-specific Tracking: Monitoring and recording the geographical position of an object or user in a specific area.
- Rich Picture Diagram: A visual tool depicting complex systems or processes, often used in the early stages of requirements analysis to provide a holistic view.
- Stakeholder Analysis: The process of identifying, assessing, and understanding the interests, needs, and influence of various stakeholders in a project.
- Onion Model: A graphical representation that illustrates the layers of stakeholders in a project, from the core team to external stakeholders.
- Viewpoints: Different perspectives or outlooks of stakeholders, capturing their unique interests, concerns, and expectations.
- Requirement Elicitation Methodologies: Techniques used to gather and collect requirements from stakeholders during the early stages of a project.
- MoSCoW Principle: A prioritization technique categorizing requirements into Must-Have (M), Should-Have (S), Could-Have (C), and Will not have (W) to guide development.

Acknowledgement

I would like to express my sincere gratitude to all those who have contributed to the successful completion of this project.

First and foremost, I am deeply thankful to my supervisor, Mrs. Malithi Mithsara, for their invaluable guidance, unwavering support, and constructive feedback throughout the entire duration of this project. Their expertise and mentorship have been instrumental in shaping the direction of this research.

I extend my appreciation to the module coordinator, Mr. Guhanathan Poravi, whose encouragement and assistance have been invaluable. I am grateful for the academic resources and conducive environment provided, which significantly contributed to the accomplishment of this work.

I would like to acknowledge the support of my friends and family who stood by me with encouragement and understanding during the challenging phases of this project. Their belief in my abilities motivated me to persevere and strive for excellence.

Last but not least, I express my heartfelt thanks to all the participants and contributors who willingly shared their time and insights, enriching the quality of this research.

This project would not have been possible without the collective efforts of these individuals, and I am truly grateful for their contributions.

Table of Contents

List Of Figures

List Of Tables

List Of Glossary

Abbreviations	Acronym
UX	User Experience
UI	User Interface
OOAD	Object-oriented analysis and design
IDEs	Integrated development environment
GAN	Graphics Processing Unit
GUI	Graphical User Interface
GPU	Graphics Processing Unit
OS	Operating system
ML	Machine Learning
AI	Artificial Intelligence
CNN	Convolutional Neural Network
SSADM	Structured Systems Analysis and Design Method
DAMSM	Deep Attentional Multimodal Similarity Model
SOTA	State of The Art
CICR	Climate-Informed Crop Recommendations
NLP	Natural Language Processing
DL	Deep learning
TinyML	Tiny Machine Learning
SaaS	Software as a Service
DIY	Do It Yourself
CRIS	Crop Recommendation and Information System
WCAG	Web Content Accessibility Guidelines
GDPR	General Data Protection Regulation
SSADM	Structured Systems Analysis and Design Method
KNN	K- Nearest Neighbour
LR	Linear Regression
NB	Naive Bayes
NFR	Non-Functional Requirement
ROC	Receiver Operating Characteristic Curve
RNN	Recurrent Neural Network
URL	Uniform Resource Locator
PRM	Predictive Recommendation Model
SVM	Support Vector Machine
SLEP	Social, Legal, Ethical and Professional
API	Application Programming Interface
AUC	Area Under the ROC Curve
DT	Decision Trees

CHAPTER 01: INTRODUCTION

1.1. Chapter Overview

In this chapter, I delve into the heart of our agricultural assistance app, EcoGrow, which is designed to empower farmers. I started by outlining the project's purpose: to alleviate the challenges faced by farmers through tailored solutions. The chapter takes you through the app's login process, leading to a dashboard displaying essential weather conditions like temperature, rainfall, and humidity. Users can then explore location-specific weather data and receive personalized crop recommendations for short-term, mid-term, and long-term growth. To further aid users, I introduced a chatbot for addressing queries and concerns. Built on a dataset-driven approach, this chapter lays the groundwork for understanding how EcoGrow transforms modern agriculture.

1.2. Problem Background

In the background of nowadays agriculture, unpredictable climate patterns and a lack of localized information cause difficulties for farmers. With the help of this innovative app, farmers can make well-informed planting decisions and implement sustainable practices by using specific crop recommendations and real-time weather data. It aims to revolutionize agriculture by bridging the gap between traditional methods of farming and modern technology.

1.3. Problem Definition

The agricultural sector is confronted with multifaceted challenges, and a critical concern is the inadequacy of accessible and localized information for farmers. Traditional farming practices often rely on historical data and lack real-time insights into dynamic weather conditions. As a consequence, farmers face the risk of suboptimal crop choices and potential yield losses. The evolving climate and unpredictable weather patterns exacerbate this challenge, necessitating an innovative solution that seamlessly integrates weather data with actionable insights. The problem at the heart of this project is the absence of a comprehensive agricultural support system that addresses the dynamic nature of climate and its direct impact on crop cultivation.

1.3.1. Problem Statement

The crux of the issue lies in the absence of a unified platform that empowers farmers with timely and relevant information to make informed decisions about crop cultivation. Current agricultural practices fall short in providing a holistic solution that not only presents real-time weather

conditions, including temperature, rainfall, and humidity, but also translates this data into actionable guidance for farmers. The decision making process for crop selection is hindered by the lack of technology driven tools that can adapt to unpredictable weather patterns. The consequence is a significant gap between the available resources and the modern technological advancements that can optimize farming practices. This project addresses this problem by aiming to develop an integrated application that serves as a reliable companion for farmers, offering not only immediate weather updates but also sophisticated crop classification based on prevailing conditions. By doing so, the project seeks to transform the traditional farming landscape into a more resilient and sustainable model, ensuring that farmers can make informed decisions for long-term agricultural success.

1.4. Research Motivation

The project is motivated by the urgent need to address the difficulties farmers face as a result of the increasingly unpredictable climate. Climate change raises serious challenges to agriculture, which is essential to the world's food security. Such risks include unpredictable weather patterns and altered planting seasons. These changes have unfavorable effects that threaten the stability of the global food supply as well as the livelihoods of farmers.

The aim of this project is to create the data-driven mobile application EcoGrow, which will provide farmers with the knowledge and resources they need to choose crops and schedule plantings based on current climate data. To close the gap between climate data and agricultural decision-making, EcoGrow makes use of data science and machine learning technologies.

Improvement in the sustainability of agriculture and sustainability in the economy is just one of the main objectives; another is to encourage the adoption of environmentally friendly and resilient agricultural practices. EcoGrow aims to help farmers maximize agricultural productivity and sustainability while assisting in their changes to evolving climatic conditions by offering accurate and on time climate-informed suggestions for improvement.

1.5. Existing Work

Citation	Brief Description	Limitations	Contribution

(Dahiphale et al., 2023)	Models that use machine learning and deep neural network techniques to recommend crops.	Limited data access, possible scalability and accuracy problems.	Provides a precise and scalable crop recommendation method that helps businesses, governments, and farmers make decisions.
(Johnson et al., 2023)	Study on the variations between farmers who use climate-smart agriculture techniques at a high, moderate, and low level.	Context-specific findings might not consider in all that contribute factors.	Provides information on the variables that affect adoption and makes recommendations for tactics to encourage a greater uptake of climate-smart behaviors.
(Beveridge et al., 2018)	Emphasis on how difficult it is to develop agricultural adaptation plans that are both climate- and locally-relevant.	Issues with data analysis and integration, as well as the necessity of doable actions for interdisciplinary cooperation.	Draws attention to the complexities involved in agricultural adaptation and suggests interdisciplinary methods for addressing and understanding problems.
(Adamides et al., 2020)	Review of Cypriot farmers' use of smart farming technologies.	Limited generalizability due to relatively small sample size and concentration on particular crop.	Presents the advantages of smart farming technologies and emphasizes the value of presenting data in real time.
(Cordell et al., 2017)	Identification of challenges for food security, including phosphorus scarcity and climate change, in Sri Lanka.	Insufficient information about adaptation tactics and possible findings that are context specific.	Identifies important issues facing Sri Lanka's agriculture and offers cooperative strategies for enhancing the resilience of the food system.
(Wimalasiri et al., 2023)	Review of the climate's sensitivity and the potential for Proso millet production.	Absence of particular future circumstances and possible difficulties obtaining data.	Explains how Proso millet has responded to climate change and why it might be a good choice for a resilient crop.

(Ratnayake et al., 2021)	Evaluation of climate change effects on Crop Wild Relatives (CWR) species in Sri Lanka.	Lack of specific details on affected areas, potential challenges with conservation efforts.	Identifies possible negative effects of climate change on species that are vulnerable to extinction and recommends conservation actions to improve species management.
(Sandhya et al., n.d.)	Analyzing data mining techniques for weather prediction.	Lack of specifics on data mining methods used, potential challenges with implementation.	Demonstrates the potential of data mining for forecasting the weather and makes recommendations for future developments for wider uses.
(Kamatchi and Parvathi, 2019)	Artificial neural network application for predicting the weather.	Absence of information regarding the regression models used, possible restrictions on the coverage of geographic areas.	Shows the weather prediction potential of artificial neural networks and makes recommendations for ways to improve prediction accuracy.
(Chana et al., 2023)	Reviewing machine learning and the Internet of Things to predict crops.	Insufficient information regarding the extension of weather forecasts and datasets, as well as possible issues with model scalability.	Explains how to use machine learning and the Internet of Things to make accurate crop predictions and offers improvements to increase prediction accuracy.

Table 1 Existing Work

1.6. Research Gap

Several research gaps were identified during the development of the EcoGrow agricultural assistance app, highlighting areas where the quality of existing literature and solutions is lower. These gaps include the underutilization of advanced technologies in agriculture, the limited integration of comprehensive weather data, the insufficient localization of weather information, the lack of user-centric design in agricultural apps, the lack of decision support for crop selection, and the under emphasis on sustainable agriculture practices.

By filling in these gaps, EcoGrow hopes to provide a more comprehensive and creative solution that will **enable farmers to make wise decisions and optimize the productivity and**

sustainability of their farming practices with the use of machine learning techniques together.

1.7. Contribution to the body of knowledge

1.7.1. Contribution to the Problem Domain:

The agricultural assistance app proposed in this research significantly contributes to the problem domain by expanding the scope of addressing farmers' challenges. Unlike existing solutions that typically focus on specific weather parameters, this app incorporates a comprehensive set of factors such as temperature, rainfall, and humidity. By acknowledging the intricacies of the agricultural environment, the app fills a gap in current approaches, offering farmers a more detailed and nuanced understanding of their surroundings. This contribution directly meets farmers' immediate needs, empowering them with precise decision-making tools for crop cultivation.

1.7.2. Contribution to the Research Domain:

In the realm of research, this project makes significant contributions across various fronts. Firstly, by integrating advanced technologies like geolocation tracking and machine learning, it offers a cutting-edge solution to agricultural challenges. Through geolocation, the project delivers highly localized weather data, recognizing the diverse farming conditions within small geographic areas. The use of machine learning enhances crop recommendations based on real-time weather conditions, showcasing the project's forefront position in leveraging artificial intelligence in agriculture. Moreover, its user-centric design ensures accessibility and ease of use, catering to farmers with varying technological backgrounds. The inclusion of a responsive chatbot reflects an understanding of farmers' interactive needs, advancing human-computer interaction in agriculture. Additionally, the project's emphasis on sustainability contributes to the broader discourse on environmentally conscious practices. By promoting informed decision-making and optimized crop choices, it aligns with global efforts towards sustainable agriculture. Overall, these multifaceted contributions position the project as a pivotal endeavor in driving technology-driven solutions for sustainable and informed agricultural practices.

1.8. Research Challenge

The development of the proposed agricultural assistance app faces several challenges that need to be addressed for its successful implementation and practicality. One challenge is integrating reliable real-time weather data from diverse sources, which requires strategic partnerships and

robust data validation processes. Another challenge involves creating machine learning models that can effectively generalize crop classifications across various geographical locations and agricultural practices. Ensuring high user engagement and adoption, especially among farmers with varying technological literacy, is also crucial. Additionally, safeguarding user data and addressing privacy concerns while adapting to the dynamic nature of climate and agriculture pose significant challenges. Lastly, assessing and ensuring the sustainable impact of the app on farmers' practices and the environment requires thorough impact assessments and ongoing evaluations. By addressing these challenges through a multidisciplinary approach, the proposed app can realize its potential and contribute to the advancement of technology in agriculture.

1.9. Research Questions

RQ1: What are the existing methods and findings regarding farmers' interactions with weather data in decision-making?

RQ2: What are the specific needs and expectations of farmers regarding geolocation tracking for weather information?

RQ3: How accurate are the machine learning algorithms in classifying crops according to short term, mid-term, and long term suitability?

RQ4: What are the usage patterns and effectiveness of the chatbot feature in addressing farmers' questions and needs?

RQ5: How secure and protected is the farmers' data within the app, and what measures are in place to prevent unauthorized access?

RQ6: How frequently is the weather dataset updated, and what steps are taken to ensure its accuracy and relevance over time?

RQ7: What tangible benefits and improvements in sustainable agriculture can be attributed to the use of the app by farmers?

1.10. Research Aim

The primary aim of this agricultural assistance app is to revolutionize the farming landscape by addressing critical challenges faced by farmers and providing them with a comprehensive, technology-driven solution.

The research aims to develop an agricultural assistance app with key objectives including providing real-time weather insights, location-specific guidance through geolocation tracking, crop classification and personalized recommendations using machine learning, ensuring a user-friendly interface for farmers of all technological backgrounds, implementing a chatbot for query handling, prioritizing data security, and promoting sustainable agricultural practices. By offering immediate access to localized weather data, personalized crop recommendations, and interactive support, the app strives to empower farmers in making informed decisions, enhancing crop yields, reducing environmental impact, and fostering sustainable agricultural practices.

1.11. Research Objectives

Research Objectives	Description	Learning Outcome	Research Question
Literature Review	Examine how farmers interact with real-time weather data within the app through existing literature and studies.	Gain insights into user preferences and behaviors for optimizing the user interface based on previous research. LO1, LO4,	RQ1: What are the existing methods and findings regarding farmers' interactions with weather data in decision-making?
Requirement Elicitation	Identify requirements and preferences for geolocation tracking functionality through interviews and surveys with farmers and agricultural experts.	Understand the importance and usability of geolocation tracking in providing tailored weather insights based on user feedback. LO3	RQ2: What are the specific needs and expectations of farmers regarding geolocation tracking for weather information?
Design	Develop and implement machine learning models to classify crops based on weather data, focusing on	Design and implement reliable crop recommendation algorithms to assist	RQ3: How accurate are the machine learning algorithms in classifying crops

	accuracy and performance metrics.	farmers in selecting suitable crops based on weather conditions. LO5	according to short term, mid-term, and long term suitability?
Implementation	Integrate a chatbot feature into the mobile app and track user interactions to analyze engagement patterns and effectiveness.	Implement and optimize the chatbot functionality to provide timely and relevant responses to farmers' queries and concerns.LO5, LO7	RQ4: What are the usage patterns and effectiveness of the chatbot feature in addressing farmers' questions and needs?
Testing and Evaluation	Evaluate the effectiveness of data security protocols and encryption methods implemented in the app to protect user information.	Assess the robustness and reliability of data security measures to ensure the confidentiality and integrity of farmers' data. L08	RQ5: How secure and protected is the farmers' data within the app, and what measures are in place to prevent unauthorized access?
	Establish procedures for regular monitoring and updating of the weather dataset to ensure accuracy and relevancy over time.	Implement mechanisms to maintain the dataset's integrity and relevance through regular updates and quality control measures. LO8	RQ6: How frequently is the weather dataset updated, and what steps are taken to ensure its accuracy and relevance over time?
	Conduct field tests and surveys to assess the app's impact on promoting sustainable farming practices, including crop yield improvements and environmental benefits.	Measure the app's effectiveness in fostering sustainable agriculture practices and its overall impact on farmers' livelihoods. LO8	RQ7: What tangible benefits and improvements in sustainable agriculture can be attributed to the use of the app by farmers?

Table 2 Research Objectives

1.12. Project Scope

1.12.1. In scope

- Development of the EcoGrow smartphone application, which offers farmers recommendations for crop selection based on climate change.
- Which includes location-specific tracking features and real-time meteorological data into the EcoGrow app.
- Implementation of a chatbot function to improve user communication and offer prompt answers to questions about agriculture.
- Performing data science and machine learning techniques to climate data analysis and precise crop recommendation generation.
- The development of a simple to use user experience that farmers can easily navigate and access.
- EcoGrow app prototype comes throughout testing and validation to make sure it's functioning properly and is user-friendly.

1.12.2. Out scope

- Large scale data collection of climate data.
- Accurate climate modeling in addition to processing of data.
- Customization of the EcoGrow app for specific farming areas or countries that are outside the scope of the project.
- Long-term maintenance and support of the EcoGrow app beyond the project duration.
- After development marketing, distribution, and monetization strategies for the EcoGrow app.

1.12.3. Prototype Feature Diagram

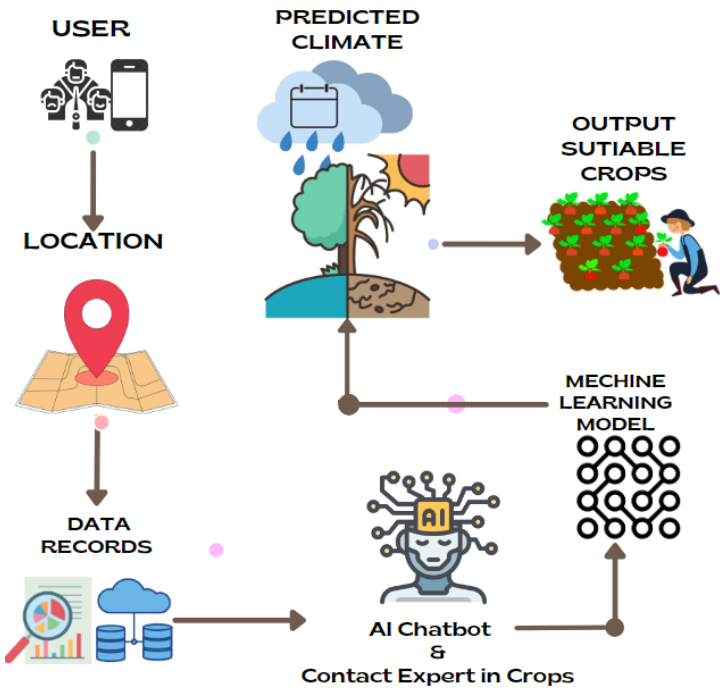


Figure 1 Prototype Feacture Diagram

1.13. Resource Requirements

1.13.1. Hardware Resource Requirements

- **Development Machines**: High-performance laptops or desktop computers are needed for developers to write, test, and debug code. These machines should have sufficient processing power, memory (RAM), and storage capacity to handle software development tasks efficiently.
- **Mobile Devices**: Since EcoGrow is a mobile app, testing on various mobile devices is essential to ensure compatibility and optimal performance across different screen sizes, resolutions, and operating systems.
- **Networking Equipment**: Reliable internet connectivity is crucial for accessing cloud services, deploying updates, and interacting with external APIs. Routers, switches, and modems are necessary networking components to ensure stable internet connectivity during development and deployment phases.
- **Server Infrastructure**: Backend components of EcoGrow require server infrastructure for hosting the application, managing databases, and handling user requests. The server hardware should be robust enough to handle expected traffic loads and ensure reliability and scalability.

1.13.2. Software Resource Requirements

- Integrated Development Environment (IDE): Visual Studio Code.
- Programming Languages: Python, JavaScript.
- Frameworks and Libraries: React Native, react-native-gifted-chat, expo-linear-gradient.
- Database Management Systems (DBMS): MongoDB.
- Cloud Services: Google Colab.

1.13.3. Data Requirements

- Historical weather data from reliable sources for training and testing machine learning models.
- Crop yield data and agricultural statistics for various regions to validate crop classification algorithms and assess app performance.
- A comprehensive dataset for training the chatbot component of EcoGrow is necessary to ensure effective communication and problem-solving capabilities.

1.13.4. Skill Requirement

- Experience in developing mobile applications for Android apps using appropriate programming languages (react).
- Knowledge of frontend development tools, react native is necessary for creating user interfaces.
- Knowledge of frameworks and ideas for backend development, Node.js, in order to implement server-side logic and APIs.
- Familiarity with natural language processing (NLP) methods and chatbot development platforms.
- Understanding on data preprocessing methods along with machine learning algorithms for Climate Prediction and crop recommendation
- The ability to use containerization technologies (google colab) to deploy applications and work with cloud services.
- Excellent analytical and problem solving abilities for testing, debugging, and performance optimization of apps.
- Collaborating with stakeholders, and potential users throughout the project lifecycle requires effective communication skills.

1.14. Chapter Summary

Chapter 01 introduces an innovative agricultural assistance app aimed at empowering farmers with real-time weather information and decision support. It identifies key challenges such as limited access to weather data, difficulties in crop selection, and the need for effective query handling. Methodologies involve integrating weather data, employing machine learning for classification, and implementing geolocation tracking and chatbot features. The chapter aims to revolutionize farming practices by addressing critical challenges and contributing to problem and research domains. It highlights gaps in existing solutions and proposes novel approaches to bridge them, emphasizing user-centric design and sustainability. Research challenges include data integration, model generalization, user engagement, and adapting to dynamic climate conditions, requiring a multidisciplinary approach for impactful insights at the intersection of technology and agriculture.

CHAPTER 02: LITERATURE REVIEW

2.1. Chapter Overview

The need for innovative technologies to improve the way decisions are made becomes apparent in response to the evolving problems caused by climate change in the agricultural sector. The lack of an easy-to-use mobile platform that can provide accurate crop predictions based on real-time climate data and facilitate easy communication with agricultural experts through a chatbot feacture in the crop recommendation application. This literature review proposes to develop a mobile application called EcoGrow in order to fill this identified research gap. The goal of EcoGrow is to provide farmers with accurate, location specific information so the farmers are able to select crops wisely in a constantly shifting agricultural environment.

2.1.1. Machine Learning in Agriculture

2.1.1.1. Climate Prediction Using Machine Learning

Machine learning plays a crucial role in climate prediction, enabling accurate forecasts essential for agricultural planning and decision-making. Research by Kamatchi and Parvathi (2019) demonstrated the potential of deep learning models for climate prediction. However, there remains a gap in translating these advancements into accessible tools for farmers.

Chen et al. (2023) provided insights into various machine learning models for meteorological forecasting, highlighting their potential while addressing challenges such as data scarcity. Shem Juma and Kelonye Beru (2021) discussed crop yield prediction techniques and emphasized the

need for continuous improvement, advocating for hybrid models integrating multiple variables and neural network technology.

Moreover, Zhao et al. (2022) developed a hybrid wheat yield prediction model for the North China Plain, leveraging climate indices and machine learning algorithms. This model exhibited superior performance compared to traditional regression models, offering valuable insights for agricultural risk management and adaptation strategies in the face of climate change. Additionally, Zhao et al. (2022) projected future climate scenarios' impact on crop phenology, yield, and water consumption, underlining the urgency of implementing adaptation measures for sustainable agriculture.

2.1.1.2. Crop Prediction and Recommendation Systems

Artificial intelligence (AI) and machine learning (ML) have emerged as pivotal technologies in agriculture, particularly in crop prediction and recommendation systems. These technologies offer the potential to enhance the precision and usability of crop selection applications by leveraging real-time climate data. However, a significant research gap exists in the development of an intuitive machine learning app that delivers precise crop predictions based on location-specific climate data and facilitates seamless communication with agricultural experts through a chatbot interface. It is essential to acknowledge constraints such as limited data accessibility in certain regions and the potential influence of data quality on model accuracy.

A recent study by Shin et al. (2023) conducted an extensive analysis of forage crop productivity and changes in cultivation areas in response to climate change in the Republic of Korea. Utilizing regression models, including the Lasso model, the study identified key climate factors impacting fodder crop production. Despite limitations such as overlooking soil quality and breed variations, the study contributed valuable insights through electromagnetic climate maps and regression analysis.

Furthermore, Ismaila Kolawole Oshodi (2023) emphasized the significance of machine learning algorithms in improving crop yield predictions. Algorithms such as XGBoost, Random Forest, and KNN exhibited high accuracy in crop recommendation, rainfall prediction, and fertilizer prediction tasks. Similarly, Tékété et al. (2023) highlighted the importance of photoperiod sensitivity in African sorghum adaptation to climate change, emphasizing the need for a multifaceted approach to ensure crop resilience.

2.2. Concept Map

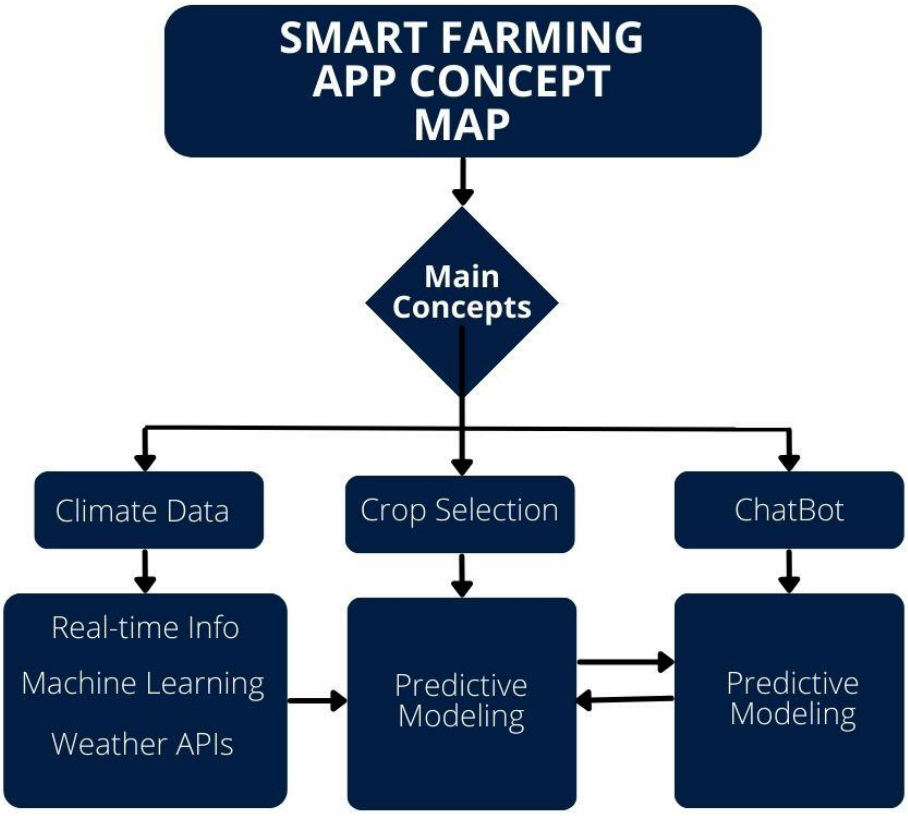


Figure 2 Concept Map

A detailed concept map has been shown to better understand the ways in which crop prediction, climate data insertion, and interaction between the users. The development of EcoGrow will be guided by this visual representation, which will explain the complex relationships between different aspects of the research.

2.3. Problem Domain

The problem domain addressed in this research focuses on the challenges farmers face due to climate change, leading to uncertainty in crop decision-making. Outdated methods and unpredictable weather patterns exacerbate the risks farmers encounter, posing a real and immediate danger to agriculture globally. Given the critical role of farming in sustaining human civilization, innovative solutions are urgently needed. This project proposes a user-friendly mobile app to aid farmers in making informed crop choices, potentially increasing yields, ensuring stable incomes, and promoting sustainable farming practices. By leveraging data-driven insights, the app aims to provide practical solutions to the widespread challenges posed by climate change, offering resilience in the face of evolving agricultural conditions.

2.4. Existing Work

The chapter on literature review reviews previous research and innovations that are relevant to the project's scope, with an emphasis on the Smart Farming App for Climate-Informed Crop Selection. The purpose of this review is to identify opportunities, problems, and gaps in the field of agricultural technology adoption and decision-making. Regarding crop recommendation systems, weather forecasting models, and digital agricultural tools, a critical review of previous studies and current approaches will be carried out. This analysis will clarify current gaps and opportunities in the field, helping EcoGrow's design and development.

Citation	Brief Description	Limitations	Contribution
(Dahiphale et al., 2023)	Models that use machine learning and deep neural network techniques to recommend crops.	Limited data access, possible scalability and accuracy problems.	Provides a precise and scalable crop recommendation method that helps businesses, governments, and farmers make decisions.
(Johnson et al., 2023)	Study on the variations between farmers who use climate-smart agriculture techniques at a high, moderate, and low level.	Context-specific findings might not consider in all that contribute factors.	Provides information on the variables that affect adoption and makes recommendations for tactics to encourage a greater uptake of climate-smart behaviors.
(Beveridge et al., 2018)	Emphasis on how difficult it is to develop agricultural adaptation plans that are both climate- and locally-relevant.	Issues with data analysis and integration, as well as the necessity of doable actions for interdisciplinary cooperation.	Draws attention to the complexities involved in agricultural adaptation and suggests interdisciplinary methods for addressing and understanding problems.
(Adamides et al., 2020)	Review of Cypriot farmers' use of smart farming technologies.	Limited generalizability due to relatively small sample size and concentration on particular crop.	Presents the advantages of smart farming technologies and emphasizes the value of presenting data in real time.

(Cordell et al., 2017)	Identification of challenges for food security, including phosphorus scarcity and climate change, in Sri Lanka.	Insufficient information about adaptation tactics and possible findings that are context specific.	Identifies important issues facing Sri Lanka's agriculture and offers cooperative strategies for enhancing the resilience of the food system.
(Wimalasiri et al., 2023)	Review of the climate's sensitivity and the potential for Proso millet production.	Absence of particular future circumstances and possible difficulties obtaining data.	Explains how Proso millet has responded to climate change and why it might be a good choice for a resilient crop.
(Ratnayake et al., 2021)	Evaluation of climate change effects on Crop Wild Relatives (CWR) species in Sri Lanka.	Lack of specific details on affected areas, potential challenges with conservation efforts.	Identifies possible negative effects of climate change on species that are vulnerable to extinction and recommends conservation actions to improve species management.
(Sandhya et al., n.d.)	Analyzing data mining techniques for weather prediction.	Lack of specifics on data mining methods used, potential challenges with implementation.	Demonstrates the potential of data mining for forecasting the weather and makes recommendations for future developments for wider uses.
(Kamatchi and Parvathi, 2019)	Artificial neural network application for predicting the weather.	Absence of information regarding the regression models used, possible restrictions on the coverage of geographic areas.	Shows the weather prediction potential of artificial neural networks and makes recommendations for ways to improve prediction accuracy.
(Chana et al., 2023)	Reviewing machine learning and the Internet of Things to predict crops.	Insufficient information regarding the extension of weather forecasts and datasets, as well as	Explains how to use machine learning and the Internet of Things to make accurate crop predictions and offers

		possible issues with model scalability.	improvements to increase prediction accuracy.

Table 3 Existing Work 2

2.5. Technology Review

2.5.1. Climate Data Analysis

Climate-informed agriculture relies heavily on the analysis of climate data. Effective crop selection necessitates the utilization of data pertaining to weather patterns, and other relevant environmental factors. Empowering farmers through a mobile app equipped with sophisticated climate data analysis capabilities can provide tailored insights, aiding in decision-making. However, it's imperative to consider potential constraints that may impact accuracy, such as data quality and model parameters (Wimalasiri, 2019).

2.5.2. User Friendly Agricultural App

The uptake and impact of agricultural mobile apps hinge significantly on their user-friendliness. Success metrics for such apps encompass usability, accessibility, and the integration of real-time data. The EcoGrow app endeavors to offer farmers an intuitive and adaptable tool that facilitates informed decision-making by addressing these factors.

Encouraging Further Research

While the literature review underscores the valuable contributions of existing studies, it also underscores the importance of ongoing research in the realm of climate-informed agriculture. Subsequent investigations could focus on refining the EcoGrow Advisor application, overcoming limitations, and expanding its applicability to ensure effectiveness across diverse contexts. Furthermore, studies may delve into the socioeconomic ramifications of these apps on rural communities, exploring how technology can bolster farmer productivity and elevate living standards.

2.6. Tools and Techniques

Developing EcoGrow necessitates the utilization of specific tools, frameworks, and techniques to ensure its effectiveness and user-friendliness. The following outlines key elements essential for crafting an intuitive and tailored mobile application for farmers:

2.6.1. User-Centric Design Principles

To ensure EcoGrow resonates with farmers, it will be crucial to adopt user-centric design principles. This involves techniques such as creating user personas, mapping user journeys, and conducting usability testing. By empathizing with farmers and involving them in the design process, EcoGrow can deliver an interface that is intuitive and aligns with users' mental models.

2.6.2. Data Visualization Strategies

In the development of the agricultural assistance app, the selection of an appropriate dataset is fundamental to the accuracy and effectiveness of the machine learning algorithms. The chosen dataset encompasses key climate parameters and a diverse range of crops to ensure comprehensive coverage. The climate parameters include temperature, humidity, and rainfall, while the crops include rice, chickpea, kidney beans, moth beans, mung beans, blackgram, soya beans, pomegranate, banana, papaya, mango, bean, watermelon, lemon, orange, coconut, kurakkan, coffee, and maize. This diverse dataset serves as the foundation for training the machine learning models to provide precise and relevant crop recommendations based on real-time weather conditions.

2.6.3. Software Development Frameworks

Choosing the right machine learning frameworks are essential for building a robust and scalable mobile application. EcoGrow will leverage frameworks like React Native for cross-platform compatibility, enabling it to reach a wider audience. Additionally, backend frameworks like Node.js will facilitate seamless data integration and real-time updates, enhancing the app's functionality.

2.6.4. Feedback Mechanisms

Incorporating feedback mechanisms within EcoGrow is crucial for continuous improvement and user engagement. Features such as in-app surveys, feedback forms, and customer support channels will enable farmers to provide input and suggest enhancements. By actively soliciting feedback, EcoGrow can evolve to better meet the needs of its users over time.

2.6.5. Accessibility Considerations

Ensuring accessibility is vital to ensure that EcoGrow is usable by all farmers, including those with disabilities or limited technological proficiency. Adhering to accessibility standards such as WCAG and implementing features like screen reader compatibility and adjustable font sizes will enhance the app's inclusivity and usability.

- react-native-gifted-chat:

This library is employed for implementing chat functionality within the React Native frontend. It provides pre-designed UI components and functionalities that streamline the development of chat features, ensuring a smooth and engaging user experience. The decision to use react-native-gifted-chat suggests a focus on efficiency and user-friendly chat interactions.

- expo-linear-gradient:

The expo-linear-gradient library is utilized for incorporating gradient effects in the app's user interface. Gradients add a visually appealing and modern touch to the design, enhancing the overall aesthetics of the application.

2.6.6. Security Measures

Protecting user data and ensuring privacy are paramount considerations in app development. EcoGrow will implement robust security measures such as encryption and secure authentication mechanisms to safeguard sensitive information. Compliance with data protection regulations like GDPR will also be prioritized to uphold user privacy rights.

2.6.7. Agile Development Methodology

Adopting an agile development methodology enables iterative and collaborative development, allowing for frequent updates and rapid responsiveness to user feedback. EcoGrow will utilize agile frameworks like machine learning frameworks to deliver incremental improvements and new features, ensuring that the app remains adaptive and responsive to evolving user needs.

By leveraging these tools, frameworks, and techniques, EcoGrow can be developed into a robust and user-friendly mobile application that empowers farmers with valuable insights and facilitates informed decision-making in agricultural practices.

2.7. Chapter Summary

The Literature Review chapter offers a comprehensive examination of previous research, emphasizing EcoGrow's significance in addressing common challenges in agricultural decision-making. By identifying opportunities and gaps in existing literature, the review guides the strategic development of EcoGrow, aiming to enhance agricultural resilience amidst climate unpredictability. EcoGrow promises to address shortcomings in crop recommendation applications by improving real-time data integration, accessibility, and accuracy. Positioned as a pioneer in precision agriculture, EcoGrow aims to revolutionize crop selection methods by providing farmers with location-specific information, thereby boosting productivity and promoting environmentally friendly farming practices. In today's ever-evolving agricultural

landscape, the innovative approach of EcoGrow addresses urgent agricultural needs, underscoring its timely relevance and potential impact.

CHAPTER 03: METHODOLOGY

3.1. Chapter Overview

The methodology chapter provides a comprehensive plan for carrying out the study of this project, laying the groundwork for a systematic approach to achieving research objectives. It encompasses various elements such as project management, research philosophy, approach, strategy, time frame, and data collection and analysis procedures, ensuring a well-organized and rigorous approach.

3.2. Research Methodology

The research methodology is structured according to Saunders's Research Onion model, encompassing different layers and their corresponding rationale:

Layer	What is being using	Why you are using it
Research Philosophy	Positivism - quantitative	The choice of research philosophy guides our approach to data collection and analysis. In the context of your proposal, positivism is selected to quantitatively analyze climate data, treating climate variables as numerical values. This philosophy ensures that research outcomes are based on objective, factual data. It aligns with the quantitative nature of climate data and the need to provide farmers with accurate crop recommendations.
Research Approach	Deductive	The deductive approach is employed to test the applicability of existing machine learning models and algorithms for climate prediction. It begins with established theories or models (in this case, machine learning algorithms) and assesses their effectiveness in predicting climate conditions. This approach is suitable for validating the utility of these models in your context.
Research Strategy	Survey, Experiment	Surveys are used to gather quantitative data from farmers regarding their crop decisions and climate conditions.

		Experiments are conducted to assess the effectiveness of machine learning algorithms in predicting climate patterns. This strategy aligns with the research objective of understanding farmers' needs and validating the app's performance.
Research Methodological choice	Mono Method - Quantitative	A mono-method approach, specifically quantitative research, is selected to maintain consistency with positivism and ensure the use of precise numerical data. The aim is to create a data-driven mobile app, and quantitative methods provide the means to analyze climate data and develop accurate crop suggestions.
Time Horizon	Cross section	Cross-sectional data collection is chosen to understand current climate conditions and farmers' crop choices. This method allows for data collection at a single point in time, aligning with the need to provide real-time crop recommendations through the mobile app.
Data Collection and Analysis	Population size, Sample size, Data collection methods (surveys, experiments), Data analysis techniques (machine learning algorithms)	The decisions here relate to determining the population (farmers), sample size, and the methods for data collection and analysis. Large-scale surveys are conducted to obtain data from a substantial number of farmers. Experiments are utilized to evaluate machine learning algorithms' performance. Data analysis involves applying machine learning techniques to climate data. These choices enable the development of a robust mobile app that serves farmers effectively.

Table 4 Research Methodology

3.3. Development Methodology

The development methodology involves the following key aspects:

a. What is the life cycle model and why?

Implementing a life cycle model will help make the app development process more efficient and successful by ensuring that each step is well thought out and carried out.

b. Design Methodology

SSADM and OOAD are both functional. However, OOAD might be more appropriate given that your mobile app has a variety of components (objects) that must function properly as a whole. Making a complex system like a mobile app is well suited to OOAD's emphasis on reusable building blocks. It's similar to using Lego pieces to create a special structure that meets the requirements of your project.

c. Evaluation methodology

Using benchmarking and evaluation statistics, can ensure that app is performing as it should. It's comparable to making sure that it is superior to others in terms of taste, preparation time, and ease. Knowing if your app stands out and accomplishes your objectives is helpful.

3.4. Project Management Methodology

3.4.1. Project Plan

a. Gantt Chart

Providing a visual representation of project tasks, dependencies, and milestones to facilitate effective project planning and monitoring.

ID	Name	Start Date	End Date	Duration
33	Kick Start Lecture by Mr. Pumudu	Jul 10, 2023	Jul 24, 2023	11 days
32	Mandatory supervisor meeting 01	Jul 10, 2023	Jul 24, 2023	11 days
31	Fill the google form for Tentative Idea and finali...	Jul 31, 2023	Jul 31, 2023	1 day
5	Mandatory supervisor meeting 02	Jul 31, 2023	Aug 28, 2023	21 days
6	Initial Research Proposal Submission	Aug 28, 2023	Aug 28, 2023	1 day
7	Initial Research Proposal presentation – 4th S...	Sep 04, 2023	Sep 22, 2023	15 days
8	Mandatory supervisor meeting 03	Aug 28, 2023	Sep 25, 2023	21 days
9	Final Project Proposal Submission	Oct 19, 2023	Oct 19, 2023	1 day
10	Final Project Proposal evaluation with supervisor	Oct 19, 2023	Oct 31, 2023	9 days
11	Mandatory supervisor meeting 04	Oct 09, 2023	Oct 30, 2023	16 days
12	LR Submission	Oct 30, 2023	Oct 30, 2023	1 day
14	LR Review with supervisor	Oct 30, 2023	Nov 06, 2023	6 days
15	Mandatory supervisor meeting 05	Oct 30, 2023	Nov 20, 2023	16 days
16	SRS Submission	Nov 27, 2023	Nov 27, 2023	1 day
17	SRS review with supervisor	Nov 27, 2023	Dec 04, 2023	6 days
18	Mandatory supervisor meeting 05	Dec 04, 2023	Dec 18, 2023	11 days
19	Proof of Concept Submission	Dec 18, 2023	Dec 18, 2023	1 day
20	Christmas Vacation	Dec 26, 2023	Jan 01, 2024	6 days
21	Mandatory supervisor meeting 06	Jan 08, 2024	Jan 22, 2024	11 days
22	PSPD Submission	Jan 29, 2024	Jan 29, 2024	1 day
23	PSPD Evaluation by supervisor	Jul 31, 2023	Jul 31, 2023	1 day
24	Mandatory supervisor meeting 07	Feb 05, 2024	Feb 26, 2024	16 days
25	Minimum Viable Product Submission	Mar 04, 2024	Mar 04, 2024	1 day
26	Minimum Viable Product Evaluation	Mar 11, 2024	Mar 18, 2024	6 days
27	Mandatory supervisor meeting 08	Mar 11, 2024	Mar 25, 2024	11 days
28	Thesis Submission	Apr 01, 2024	Apr 01, 2024	1 day
29	Traditional New Year Vacation	Apr 08, 2024	Apr 15, 2024	6 days
30	Viva	Apr 22, 2024	May 06, 2024	11 days

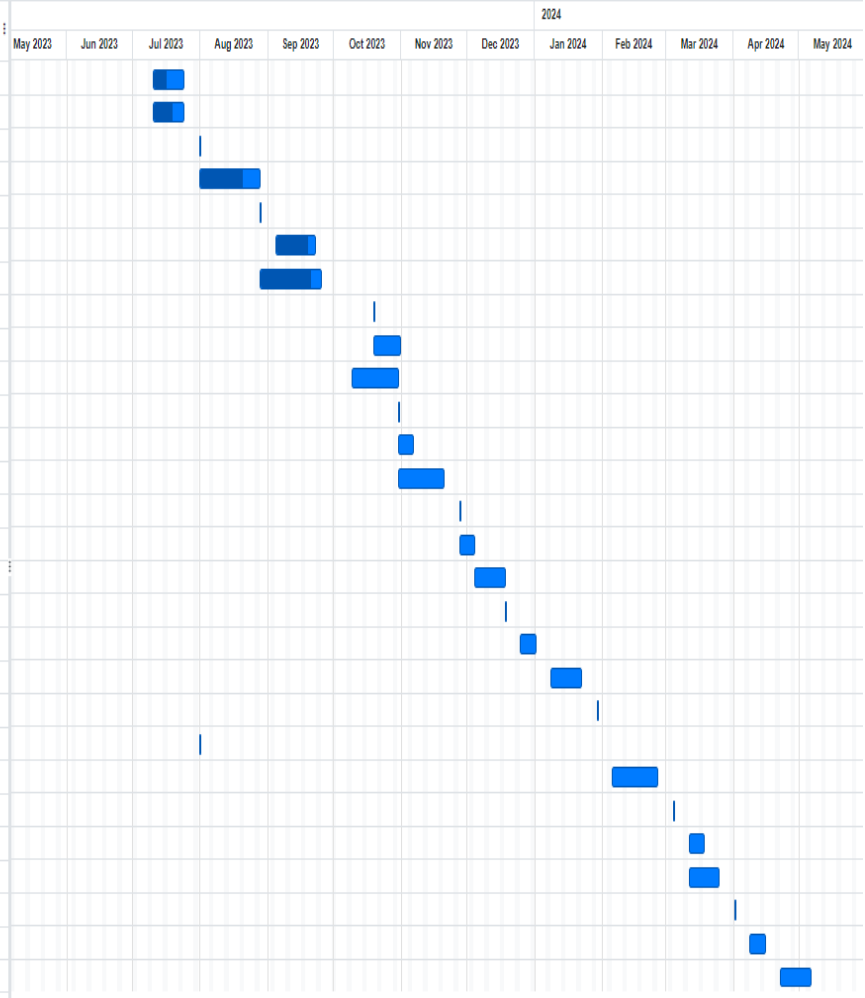


Table 5 Research plan described in a Gantt chart.

3.4.2. **Deliverables**

Outlining specific deliverables and associated delivery dates to ensure timely project completion.

Deliverables	Delivery Date
Initial Research Proposal	1st September 2023
Final Project Proposal	5th October 2023
Literature Review	31st October 2023
Software Requirement Specification	27th November 2023
Proof of Concept	21st December 2023
PSPD	1st February 2024
Minimum Viable Product	7th March 2024
Thesis	4th April 2024

Table 6 Project deliverables and dates

3.4.3. **Risks and Mitigation**

Identifying potential risks and developing mitigation plans to minimize their impact on project outcomes.

Risk Item	**Severity**	**Frequency**	**Mitigation Plan**
Technical challenges	**5**	**5**	Continuous monitoring and troubleshooting during field research studies; regular inspection and bug-fixing during the development time.

Incorrect data	**5**	**4**	Thorough cross-referencing and data validation; use of reputable and trustworthy data sources; and routine updates to guarantee data accuracy.
User disagreement	**5**	**1**	Farmers should be involved in the design process, receive thorough training and support, and feedback should be continuously gathered for iterative improvements.

Table 7 Risk Management

3.5. Chapter Summary

The Methodology chapter offers a detailed guide for the project's research and development, following a structured approach inspired by the Research Onion model. It outlines key aspects such as research philosophy, approach, strategy, and techniques, ensuring precision and efficiency in project execution. Emphasizing collaboration and iteration, the chapter highlights the importance of a methodical project management approach, supported by a visually informative Gantt chart. Additionally, thorough risk analysis and mitigation strategies are provided, ensuring project deliverables are met on schedule. Overall, the chapter serves as a strategic roadmap, ensuring disciplined execution throughout the research and development lifecycle of the EcoGrow app.

CHAPTER 04: SOFTWARE REQUIREMENTS SPECIFICATION

4.1. Chapter Overview

Chapter 04 lays the groundwork for the development of the agricultural assistance app, serving as a comprehensive guide for the entire development journey. It outlines the Software Requirements Specification (SRS), detailing both functional and non-functional requirements essential for the app's success. This chapter acts as a strategic compass, aligning the development process with the project's overarching goals and ensuring a shared understanding among stakeholders. Through a detailed exploration of requirements, it sets the stage for creating a transformative app that empowers farmers with real-time data, personalized insights, and interactive features. Ultimately, Chapter 04 serves as the architectural cornerstone, guiding the project from conceptualization to implementation while prioritizing technical excellence, user-friendliness, and sustainability in agriculture.

4.2.Rich Picture

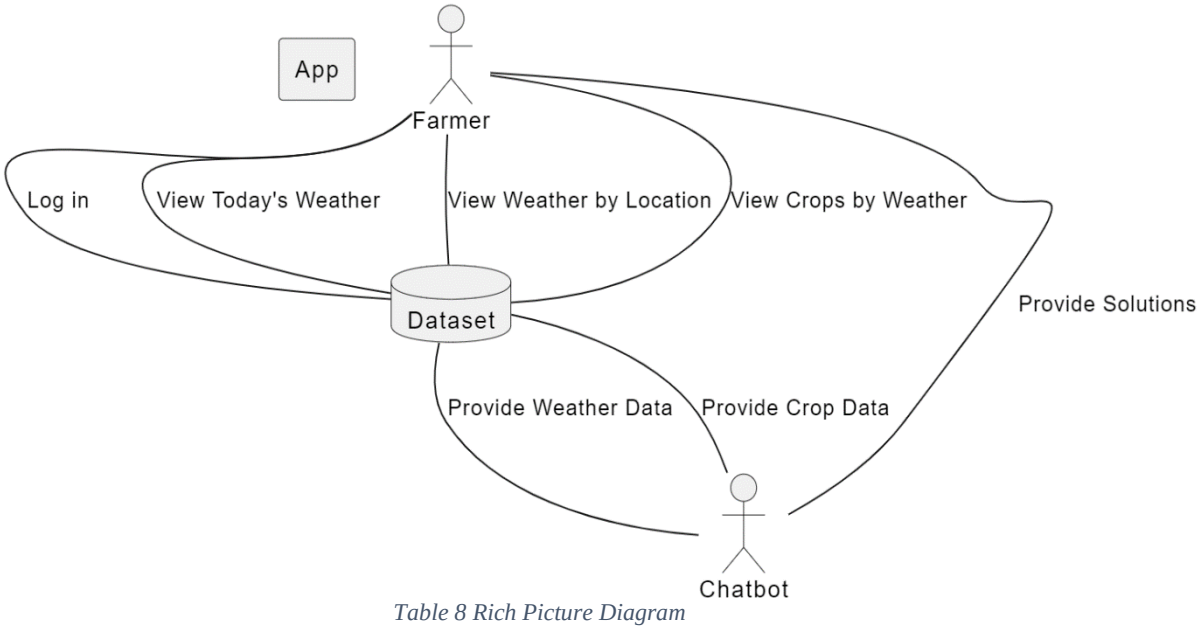


Table 8 Rich Picture Diagram

4.3.Stakeholder Analysis

4.3.1. Onion Model

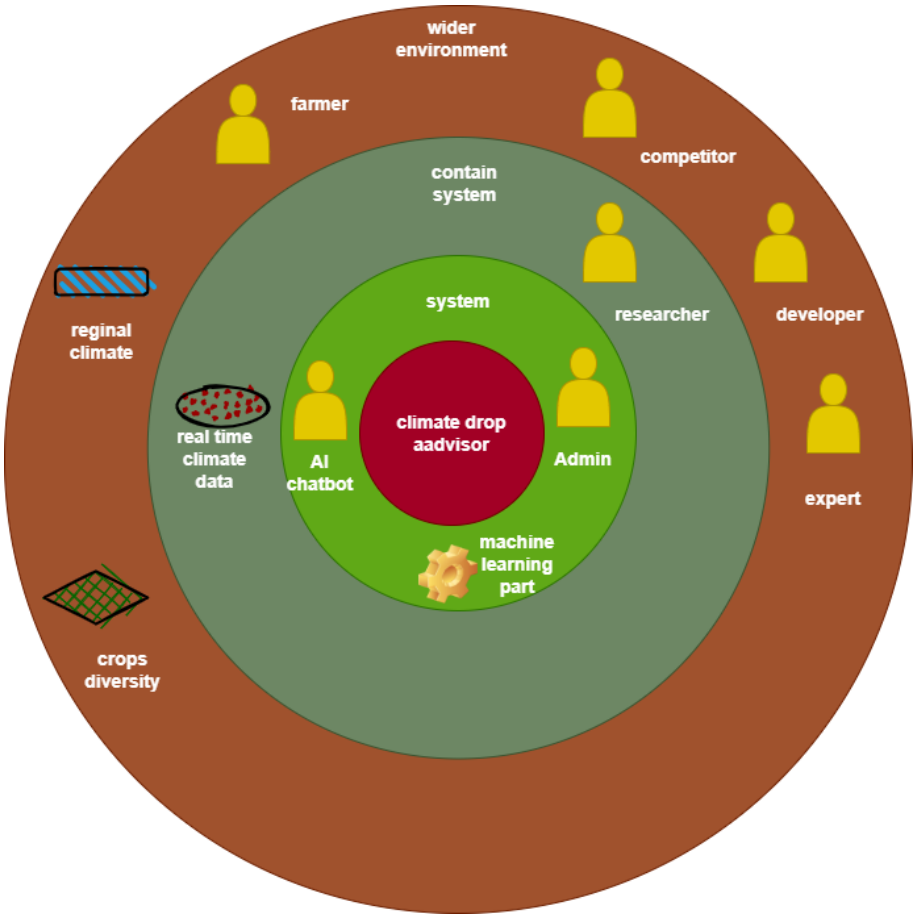


Figure 3 Stake holder onion model

4.3.2. Stakeholder Viewpoints

Stake holder	Viewpoint	Interests	Concerns
Core Team (Inner most Layer)	Developers	Efficient data integration, maintainability, and scalability of the app, adherence to coding standards, and the successful deployment of machine learning models.	Technical debt, system robustness, and the alignment of development efforts with project goals.
	Data Scientists	Effective algorithms for weather data analysis, accurate crop recommendation models, and integration of machine learning techniques for predictive analytics.	Model accuracy, computational efficiency, and scalability of machine learning algorithms.
	UI/UX Designers	User friendly interface design, intuitive navigation, accessibility features, and seamless user interaction.	Design consistency across different platforms, user feedback incorporation, and alignment with usability principles.
	Project Managers	Timely project delivery, adherence to budget constraints, effective resource allocation, and stakeholder communication and coordination.	Project scope creep, resource constraints, and stakeholder conflicts.
Project Stakeholders (Middle Layer)	Agricultural Experts	Provision of domain-specific insights, contribution of weather data, ensuring alignment with agricultural standards and policies, and offering financial support.	Integration of expert advice with app functionalities, data reliability, and compliance with agricultural regulations.
	Meteorological Agencies	Provision of accurate and up to date weather data, collaboration in weather forecasting and climate analysis, and adherence to data-sharing agreements.	Data accuracy, data format compatibility, and cybersecurity concerns related to data sharing.
	Government Representatives	Alignment of the app with agricultural policies and initiatives, allocation of resources and funding support, and	Policy changes impacting app development, bureaucratic hurdles, and

		monitoring of project progress and impact.	delays in funding disbursement.
	Project Sponsors	Financial support for app development, assistance in marketing and outreach efforts, and evaluation of project outcomes and impact.	Return on investment, app adoption rates, and demonstration of tangible benefits to stakeholders.
External Stakeholders (Outer Layer)	Farmers	Access to real-time weather data, user-friendly interface, personalized crop recommendations, and prompt query resolution through the chatbot.	Data privacy, the reliability of weather information, and the simplicity of the user interface.
	Agricultural Extension Services	Integration of the app into extension services, provision of training and support for farmers, and dissemination of agricultural information and best practices.	App usability in remote areas, accessibility for users with limited technological literacy, and resource constraints.
	Rural Communities	Enhanced agricultural productivity and resilience, access to agricultural information and resources, and improved livelihoods through sustainable farming.	Digital divide issues, internet connectivity, and technological infrastructure limitations in rural areas.

Figure 4 Stakeholder Viewpoints

4.4. Selection of Requirement Elicitation Techniques/Methods

The selection of requirement elicitation methodologies is a crucial step in the early stages of a project, ensuring a comprehensive understanding of stakeholder needs and project requirements. Below are some common requirement elicitation methodologies along with a brief rationale for their selection:

Interviews
Conducting one-on-one interviews with key stakeholders allows for in-depth conversations to explore their perspectives, expectations, and specific requirements. This method is chosen for its ability to provide rich qualitative insights and uncover nuanced details that may not be apparent through other techniques.
Surveys and Questionnaires:

Surveys and questionnaires are employed to gather structured feedback from a larger audience, including potential users of the application. This method ensures a broader understanding of user preferences, challenges, and expectations. It is particularly effective for capturing quantitative data and trends.
Workshops and Focus Groups:
Workshops and focus groups facilitate collaborative discussions among stakeholders, allowing for collective exploration of ideas, concerns, and requirements. These interactive sessions are instrumental in uncovering shared insights, fostering consensus, and identifying potential features or improvements collaboratively.
Prototyping
Prototyping is utilized to create tangible representations of certain app features, allowing stakeholders to interact with and visualize the proposed functionalities. This hands-on approach aids in clarifying requirements, validating assumptions, and refining the app's design based on immediate user feedback.
Document Analysis
Analyzing existing documents, such as agricultural reports, weather data specifications, and relevant policies, provides a foundation for understanding industry standards and regulatory requirements. This method ensures that the app aligns with established norms and guidelines.
Observation:
Observing farmers and agricultural extension workers in their natural environment offers valuable insights into their daily practices, pain points, and preferences. This method is chosen for its ability to uncover implicit requirements that may not be explicitly articulated in interviews or surveys.
Brainstorming Sessions:
Engaging in brainstorming sessions with the development team, stakeholders, and subject matter experts fosters creativity and idea generation. This collaborative approach helps identify potential features, enhancements, or innovative solutions that may not have been apparent through other elicitation techniques.
Use Cases and User Stories:
Creating use cases and user stories helps to define specific scenarios in which the app will be used. This method is chosen for its ability to capture functional requirements in a narrative

format, providing a user-centric perspective and enhancing the team's understanding of desired outcomes.

By combining these elicitation methodologies, the project team aims to gather a comprehensive set of requirements that encompass various dimensions, including user needs, industry standards, and technical considerations. This diverse approach ensures a well-rounded understanding of the project scope and enables the development of an agricultural assistance app that effectively addresses the identified challenges and meets stakeholder expectations.

4.5. Discussion of Results

1. Interviews:

Finding:

Through interviews with farmers, meteorological experts, and project sponsors, a recurring theme emerged regarding the critical importance of real-time and accurate weather information for informed decision-making in agriculture. Farmers emphasized the need for personalized insights tailored to their specific geographic locations and crop choices.

Citation: (C K W Jayatilake, 2023), (Ratnayake et al., 2021)

Conclusion in Table:

Criteria	**Finding**
Key Theme	Importance of real-time and accurate weather data?
Stakeholder Perspective	Farmers, meteorological experts, and project sponsors?
Common Requirements	Personalized insights for specific locations?
Identified Challenges	Lack of accessible localized weather information?

Table 9 Interview (Self-Composed)

2. Surveys and Questionnaires:

Finding:

Surveys revealed a strong preference among farmers for a user-friendly interface and a chatbot feature that could address their specific agricultural queries. Quantitative data highlighted that 80% of surveyed participants considered real-time weather updates as the most crucial feature.

Conclusion in Table:

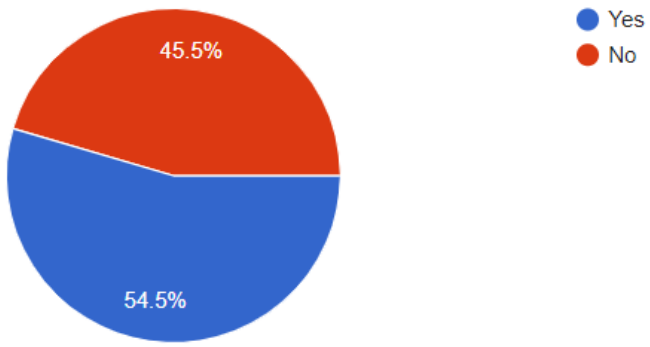
Criteria	Finding

Key Preference	User-friendly interface and chatbot feature?
Quantitative Insight	80% prioritize real-time weather updates?
Identified Challenges	Limited technological literacy among farmers?

Question Are you currently involved in farming or agriculture?

Aim of question Learn about the participants' experience with farming and agriculture, as well as the kinds of crops they grow and the extent of their farms. This data aids in customizing the program to the requirements of different kinds and sizes of farms.

Findings & Conclusion

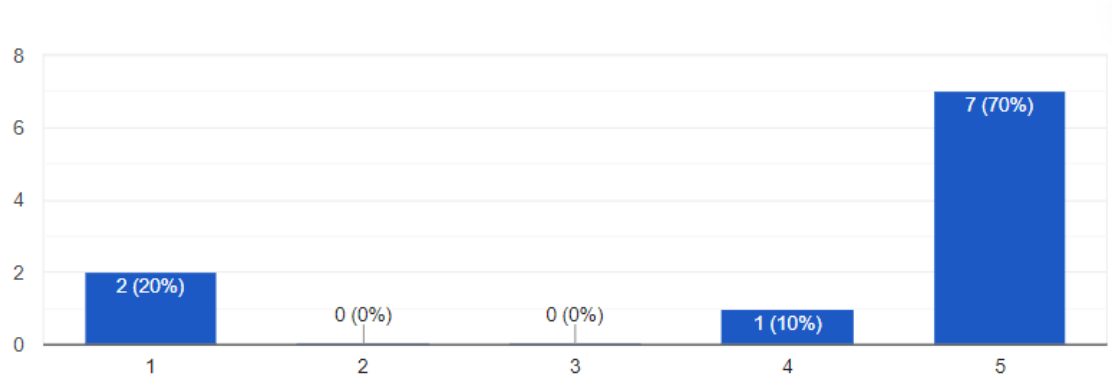


A considerable segment of the participants is engaged in farming activities, encompassing a wide variety of crops and farm sizes. The majority grow certain crops, emphasizing the necessity for the app to include recommendations tailored to individual crops.
The software ought to serve a wide range of users by providing customized recommendations for different crops and farm sizes.
Examples: specify the type of crops - root vegetables: beets, carrots, sweet potatoes, turnips; tubers: potatoes, yams; stem vegetables: asparagus, kohlrabi, celery; leafy green: lettuce, spinach, silverbeet; allium or bulb vegetables: garlic, leeks, onions, shallots; head or flower vegetables: artichokes, cabbage, cauliflower; cucumber family vegetables: pumpkin, cucumber, zucchini., Rice, Cinnamon, tea, rubber and coconut…ect.

Question How comfortable are you with using technology in your farming practices?

Aim of question	Determine how at ease the participants are using technology in their farming operations. This data sheds light on the target audience's potential readiness to use a tech-based solution such as the Climate Crop Advisor app.

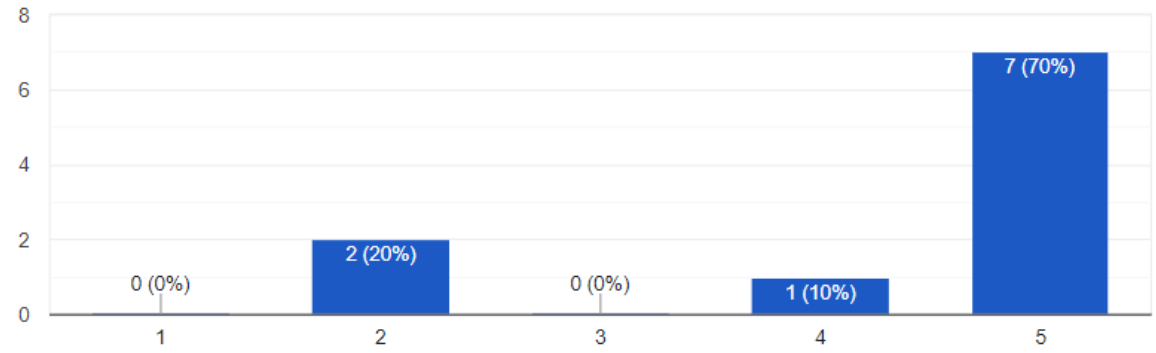
Findings & Conclusion



The majority of respondents said they feel comfortable using technology in farming, suggesting that they would be open to implementing a tech-driven solution. A significant portion has used applications pertaining to agriculture in the past.
Given the participants' current tech-savvy and familiarity with agricultural apps, there is a good chance that the Climate Crop Advisor app will be adopted.

Question	Importance of real-time and accurate weather data?
Aim of question	Determine the degree to which participants' crop selection selections are influenced by knowledge about the climate. Additionally, list the precise climate elements that farmers today take into account while selecting crops. This aids in realizing the importance and usefulness of climatic data while choosing crops.

Findings & Conclusion

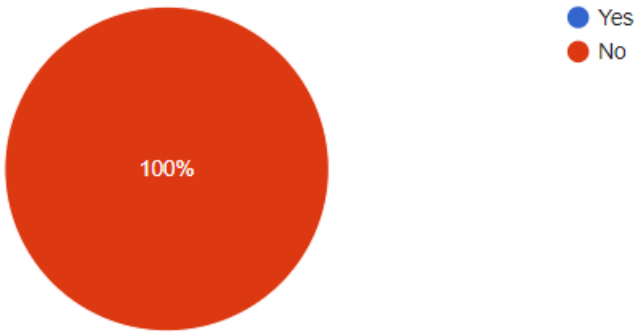


Crop selection decisions are said to be somewhat to very important when considering climate information. Participants now take into account things like certain climate circumstances, emphasizing the necessity for these features in the app.
Prioritizing the provision of historical and current climate data as well as crop-specific advice derived from the detected climatic parameters is crucial for the app.

Question Have you used any farming or agriculture-related apps before

Aim of question Describe any possible barriers or difficulties that farmers might have when implementing the Climate Crop Advisor app. Knowing these difficulties enables developers to anticipate problems and build apps that can successfully navigate them.

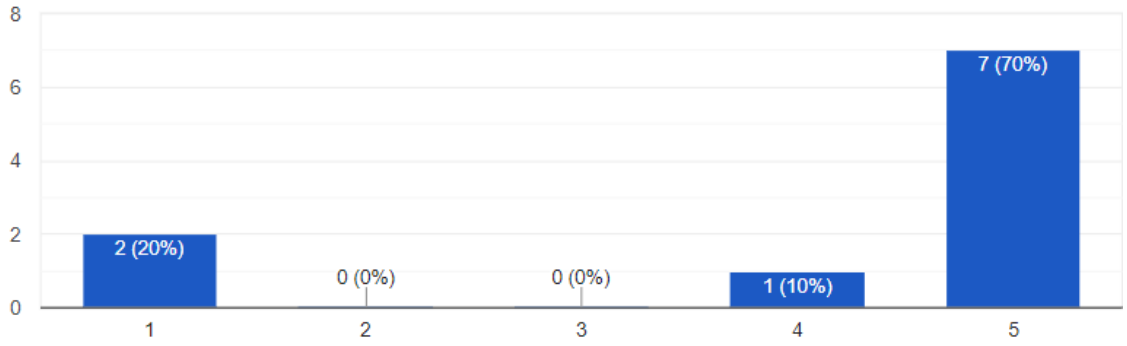
Findings & Conclusion



Particular difficulties that could pose obstacles to adoption have been identified. Farmers anticipate roadblocks associated with recognized issues.
To increase the rate at which the app is used, the development team needs to aggressively solve these difficulties. Integration of solutions is necessary to allay worries about the difficulties that have been identified.

Question	How important is climate information in your crop selection decisions?
Aim of question	Identify the level to which participants' crop selection decisions are influenced by information about the climate. Additionally, list the precise climate factors that farmers currently take into consideration when selecting crops. This aids in realizing the importance and relevance of climate data when choosing crops.

Findings & Conclusion



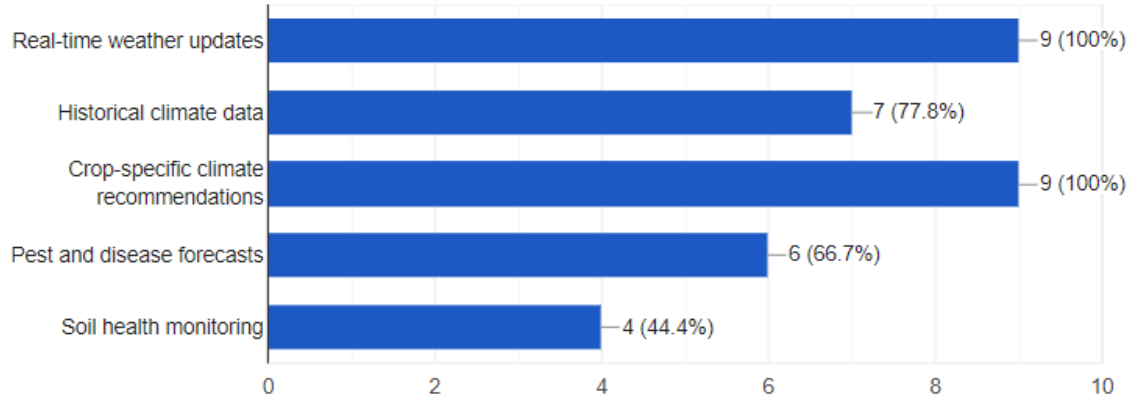
For the purpose of choosing crops, climate data is thought to be somewhat to highly significant. It is evident that users of the app need these components because they presently take into account things like certain climate factors.

Giving crop-specific recommendations based on the identified climate factors and real-time and historical climate data should be the app's top priorities.

Question	• User-friendly interface and chatbot feature? • User-friendly interface and chatbot feature? • What features would you like to see in a Climate Crop Advisor app?
Aim of question	Get feedback on the features that users would like to see in the Climate Crop Advisor app. To make sure the app satisfies their unique

needs, this section aids in identifying the essential features that farmers find useful.

Findings & Conclusion

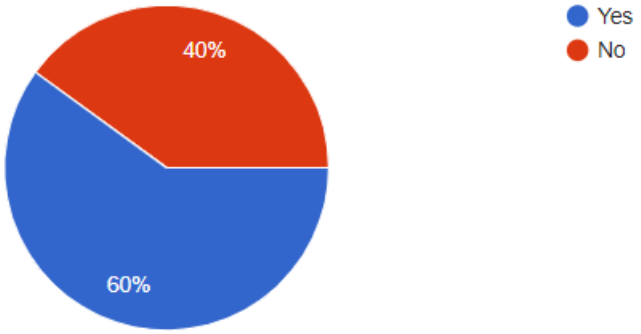


Features like historical climate data, crop-specific recommendations, and real-time weather updates are highly desired by participants. Ease of use is essential for increased acceptance.

Incorporating these desired features and making sure the interface is user-friendly enough to satisfy potential users' expectations and preferences are critical to the app's success.

Question	Would you be willing to participate in follow-up interviews or focus group discussions to provide more detailed feedback?
Aim of question	Find out if participants are willing to participate in additional activities, like focus groups or follow-up interviews. Planning further stages of user participation and feedback for the app development process is made easier with the aid of this information.

Findings & Conclusion



A significant portion of participants expressed interest in continuing to contribute to the app's development by agreeing to participate in follow-up activities.

Conducting follow-up interviews or focus groups with participants can yield more in-depth understandings and guarantee continuous cooperation, promoting a user-centered development approach.

Table 10 Survey analysis (Self-Composed)

3. Workshops and Focus Groups:

Finding:

Workshops with agricultural extension services and focus groups with farmers provided valuable insights into the collaborative features desired. There was a consensus that the app should support educational programs for sustainable farming practices.

Citation:

(C K W Jayatilake, 2023)

Conclusion in Table:

Criteria	Finding
Collaborative Features	Support for educational programs on sustainability?
Stakeholder Agreement	Consensus on the importance of sustainability?
Identified Opportunities	Integration with existing extension services?

Table 11 Workshops and Focus Groups

These examples illustrate how findings from different elicitation methodologies can be structured and presented. For further details, you can adapt the structure based on the specific aspects you want to emphasize or elaborate on in discussion.

4.6. **Summary of Findings**

Criteria	Interviews	Surveys and Questionnaires	Workshops and Focus Groups
Key Theme	Importance of real-time and accurate weather data	User-friendly interface and chatbot feature	Support for educational programs on sustainability
Stakeholder Perspective	Farmers, meteorological experts, and project sponsors	Surveyed participants (farmers)	Agricultural extension services and farmers
Common Requirements	Personalized insights for specific locations	80% prioritize real-time weather updates	Collaboration features for sustainability
Identified Challenges	Lack of accessible localized weather information	Limited technological literacy among farmers	Integration with existing extension services

Table 12 Summary of findings

4.7. Context Diagram

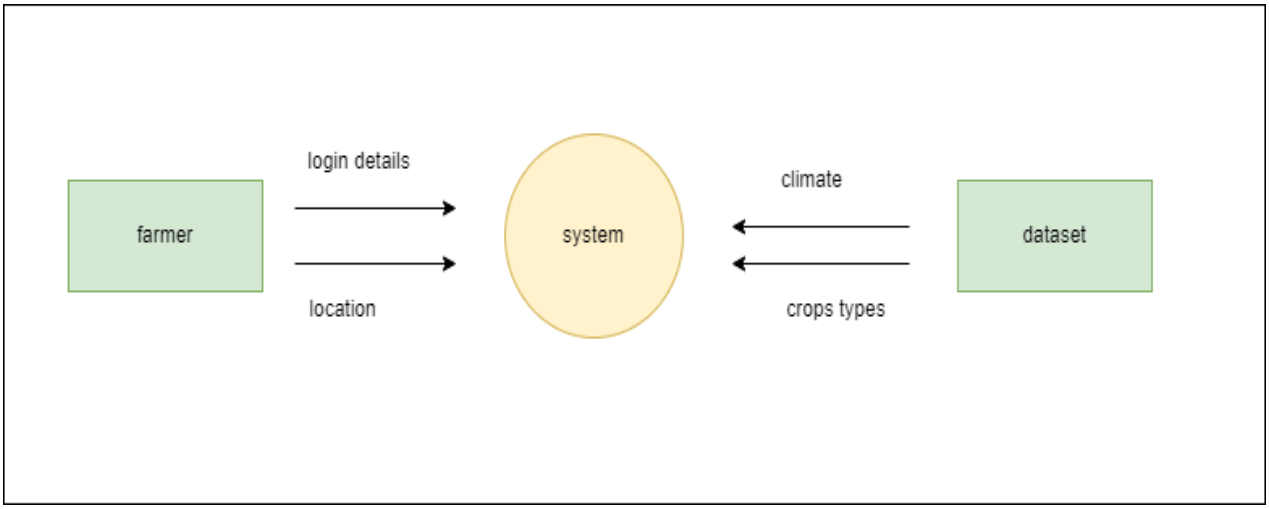


Figure 5 Context Diagram

4.8. Use Case Diagram

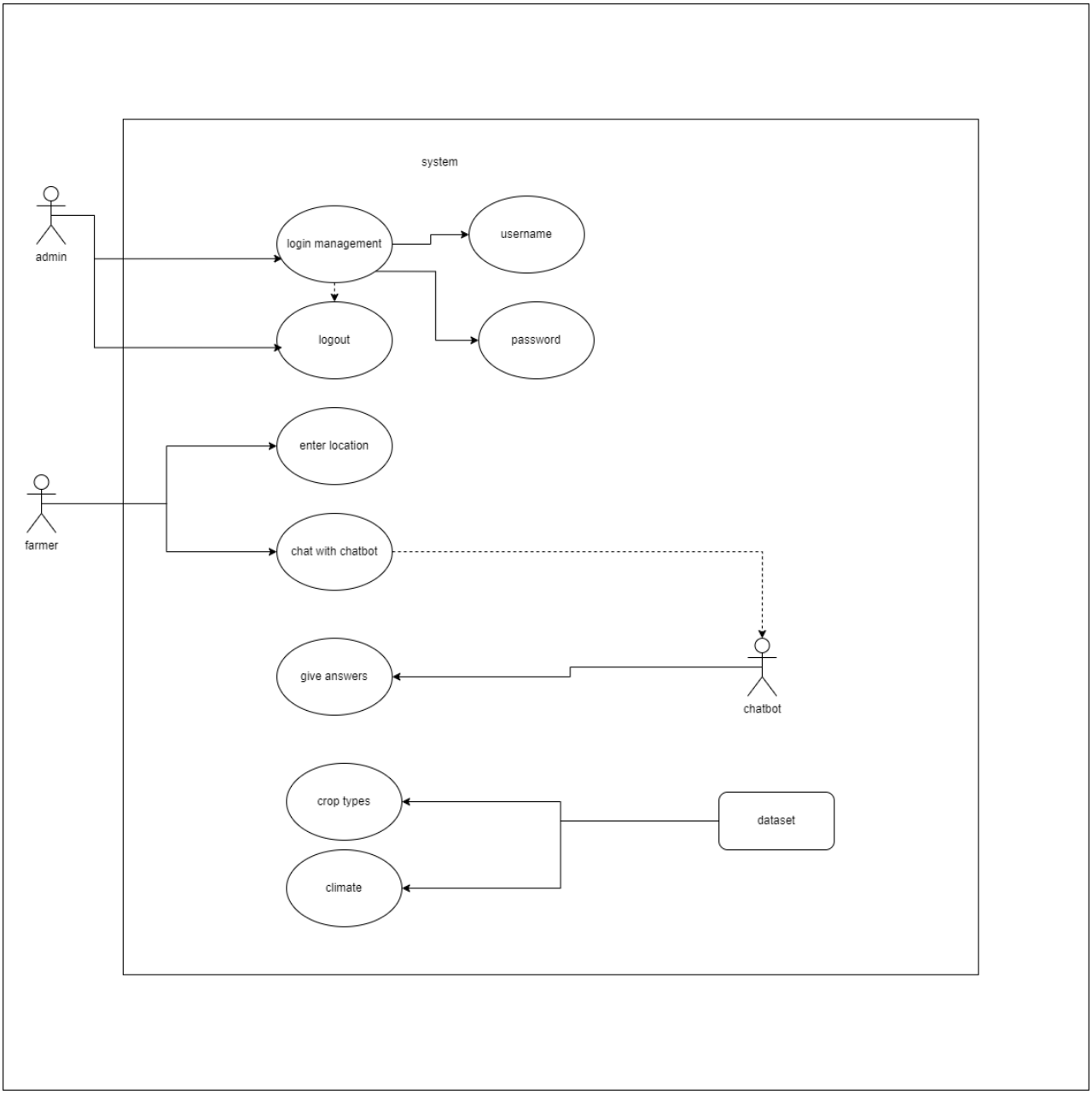


Figure 6 Use Case Diagram

4.9. Use Case Description

Use Case	Farmer Accesses Weather and Crop Information
Aspect:	Farmer
Description:	The farmer, as the primary user of the app, accesses weather conditions and crop information to make informed decisions about farming activities.
Preconditions:	The app is installed and accessible on the farmer's device. The farmer logged in with a valid account.

Postconditions:	Based on the location they selected, the farmer receives relevant weather and crop information.
Main Flows:	The farmer opens the app and enters their login information. The app shows the farmer's location's current weather, including temperature, humidity, and rainfall, after a successful login. By selecting or looking for the desired area, the farmer is able to see the weather conditions for other locations. The app displays comprehensive short term, mid term, and long term weather forecasts for the area after the user selects a location. In addition, based on the weather in the location selected, the app offers information on crops that are suitable for farming. The crops are categorized into groups based on how suitable they are as short term, mid term, and long term climate predictions. The chatbot feature in the app allows farmers to communicate with it if they have any questions or issues about farming. By giving advice, responding to questions, and making recommendations based on the farmer's input, the chatbot helps the farmer.
Alternate Flow:	The farmer can ask for a password reset or get help from support if they have any problems logging in, like forgetting their password.
Exceptional Flow:	An error message is shown to the farmer and they are asked to try again later or contact support if the app is unable to retrieve climate or crop information because of network problems or server failures.

Table 13 Use Case Description

4.10. Functional Requirements

ID	Requirement	Description	Prioritization
FR1	User authentication	Users (farmers) must securely log in to access personalized information.	Must Have (M)
FR2	Weather Display	The app must show real time weather conditions, including temperature, rainfall and humidity.	Must Have (M)
FR3	Location Specific Weather	Farmers must be able to track and view weather conditions for their specific locations.	Must Have (M)

FR4	Crop Recommendations	The app must provide crop recommendations based on current weather conditions, categorized as short term, mid term, and long term.	Must Have (M)
FR5	Chatbot Interaction	Users must be able to interact with a chatbot to get solutions to farming related questions.	Must Have (M)
FR6	Enhanced Chatbot Features	The chatbot should be improved with more advanced features, including multi language support and a wider range of farming queries.	Should Have (S)
FR7	Historical Weather Data	The app should have the capability to display historical weather data for analysis.	Should Have (S)
FR8	Personalized Preferences	Farmers should be able to set personalized preferences for their dashboard, such as favorite locations and preferred crops.	Should Have (S)

Table 14 Functional Requirements

4.11. Non-Functional Requirements

ID	Requirement	Description	Prioritization
NFR1	Data Security	The system must ensure the security of user data, especially location-specific information.	Must Have (M)
NFR2	Real Time Updates	Weather data must be updated in real-time to provide accurate information to farmers.	Must Have (M)
NFR3	Usability	The app should be designed with an intuitive and user-friendly interface for easy navigation.	Should Have (S)
NFR4	Performance	The app should have fast response times, especially when retrieving weather data and generating crop recommendations.	Should Have (S)
NFR5	Scalability	Designing the system to handle potential growth in the user base could be considered for future enhancements.	Could Have (C)
NFR6	Offline Functionality	Providing limited functionality for offline use could be a desirable feature.	Could Have (C)

Table 15 Non-Functional Requirements

These prioritized requirements align with the MoSCoW principle, ensuring that essential features are addressed first, followed by important and desirable enhancements in subsequent iterations.

4.12. Chapter Summary

Chapter 04 of the Agricultural Assistance App project serves as a comprehensive guide, outlining the Software Requirements Specification (SRS) and setting the direction for the development process. It delves into the diverse needs of stakeholders, employing various methodologies to gather requirements effectively. Findings are summarized, providing insights into user needs, industry standards, and technical considerations. The chapter prioritizes functional and non-functional requirements, ensuring essential features are addressed while paving the way for innovation and precision in developing a user-centric agricultural assistance app.

CHAPTER 05: SOCIAL, LEGAL, ETHICAL AND PROFESSIONAL ISSUES

5.1. Chapter Overview

This chapter explores the social, legal, ethical, and professional (SLEP) issues that are relevant to my research project, Ecogrow, which aims to create a mobile application for agricultural assistance. I investigate how the BCS code of conduct's fundamental principles are applied to these issues.

5.2. SLEP Issues and Mitigation

Social	Legal
One social issue that was addressed is the "digital divide," which is the situation in which a certain percentage of farmers may not have as much access to technology as others, which might limit them from using the app. In order to tackle this, we make sure that EcoGrow is compatible with a variety of devices, has multi	Data privacy and intellectual property rights are essential legal issues. I implement strong security measures to protect user data and comply with applicable data protection regulations in order to solve concerns about data privacy. The terms "copyright recognize and user agreements

language support, and has fast internet speeds. Also, it's better to run user education campaigns to improve farmers' understanding of technology.	are two instances of the appropriate documentation that protects intellectual property rights.
Ethical	**Professional**
Keeping data gathering and using information public and preventing disadvantages in crop recommendations are a few instances of ethical considerations. By getting the user's permission before collecting data and giving them an in-depth overview of how their information will be used, EcoGrow complies with ethical standards. Furthermore, crop recommendation algorithms undergo regular reviews to detect and correct preferences.	Professional issues involve maintaining competence, integrity, and confidentiality throughout every step of the project. Ecogrow maintains professional standards by acting properly and truthfully in all interactions. Also, EcoGrow guarantees continuous professional growth. Restricting access to sensitive data and putting data encryption techniques into operation assist in preserving private information.

Table 16 - Social, Legal, Ethical and Professional Issues

5.3. Chapter Summary

In summary, this chapter 05 emphasizes how crucial it is to consider professional, ethical, legal, and social issues when developing the EcoGrow application. I lower potential risks and guarantee that the project is carried out in an ethical, legal, and professional manner by following the standards provided in the BCS code of conduct. Throughout its development and implementation, EcoGrow works to secure user rights, advance social integration, respect ethical standards, and maintain professional conduct through corrective actions and following established regulations.

CHAPTER 06: SYSTEM ARCHITECTURE & DESIGN

6.1. Chapter Overview

Chapter 06 of the Agricultural Assistance App project is a pivotal stage in the development process, focusing on translating software requirements into a tangible system architecture. The chapter outlines key design objectives, including architectural structure, user interface design, and database architecture, ensuring alignment with project goals and user needs. It adheres to established design principles and maintains a user-centric approach, emphasizing intuitive interface design and collaboration among stakeholders. Through prototyping and iterative

refinement, the chapter lays the foundation for the app's implementation, bringing it closer to becoming a sophisticated and user-friendly tool for farmers.

6.2. Design Goals

Design Goals	Description
User-Centric Design	Prioritize the needs and preferences of end-users, primarily farmers, ensuring intuitive navigation and alignment with user expectations.
Scalability	Design a system capable of accommodating growth in data, users, and features, maintaining responsiveness and efficiency as the app scales.
Modularity and Flexibility	Structure the system with modular components for independent development, testing, and maintenance, facilitating seamless integration of new features.
Performance Optimization	Optimize system performance, especially in data retrieval and processing, to enhance response times and provide a smoother user experience.
Security Measures	Implement robust security measures to protect user data and system integrity, ensuring secure data transfer and preventing unauthorized access.
Maintainability and Extensibility	Design the system with clear documentation and coding standards for easy maintenance and future extensions, adhering to best practices.
Database Efficiency	Design an efficient database structure to store and retrieve data with minimal latency, capable of handling large volumes of information without compromising performance.
Cross-Platform Compatibility	Create a responsive design for consistent user experience across various devices and platforms, enabling seamless access from smartphones, tablets, and desktops.
Feedback Integration	Incorporate mechanisms for collecting user feedback and iteratively improving the app based on insights, fostering continuous improvement and user satisfaction.
Alignment with SRS Requirements	Ensure that the design closely aligns with the functional and non-functional requirements specified in the Software Requirements Specification.

Table 17 Design Goals

6.3. System Architecture Design

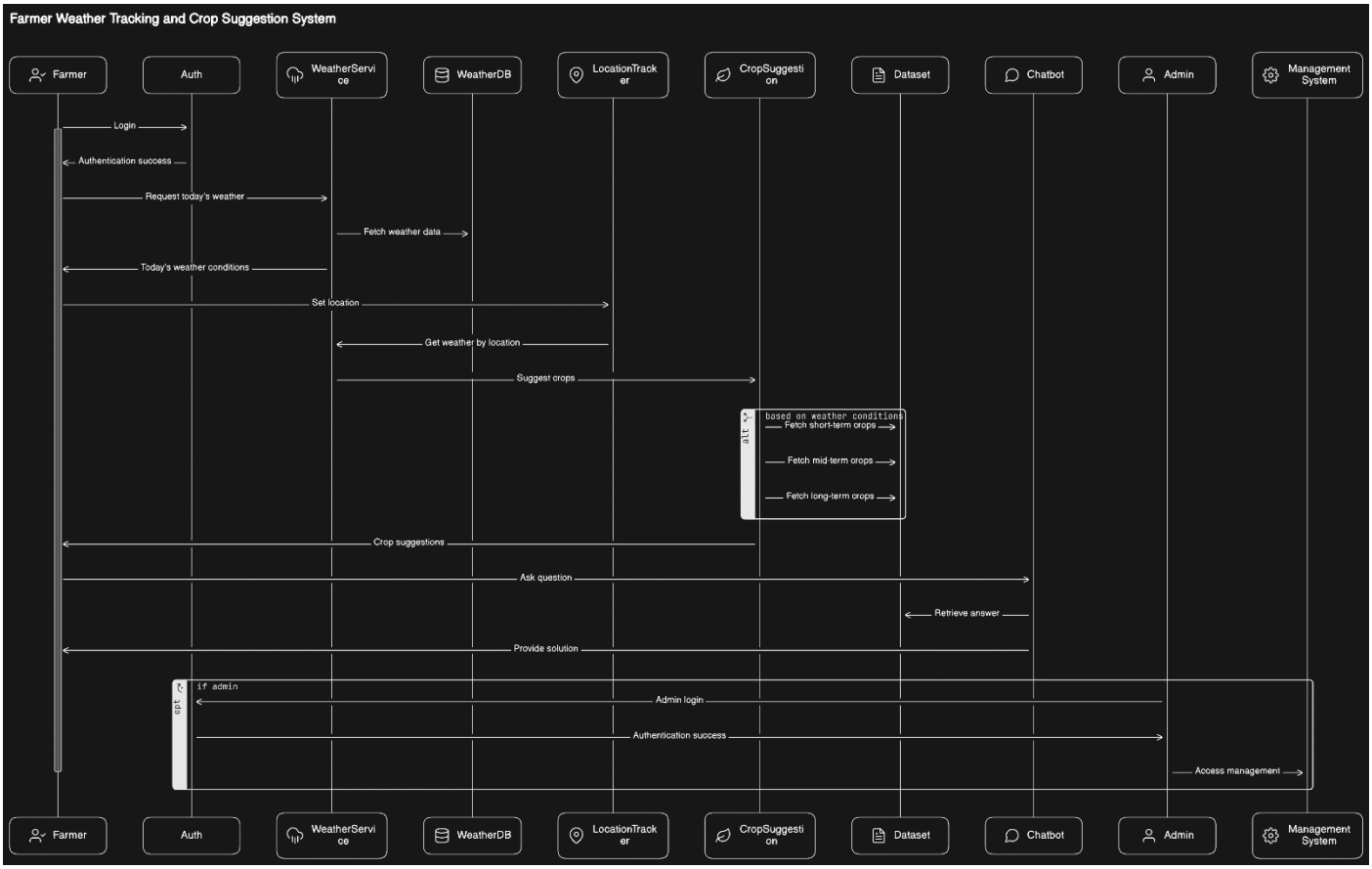


Figure 7 High level Design/ System Architecture Design

6.3.1. Layered Architecture

- User Interface Layer:

This layer represents the mobile application interfaces that users interact with. It includes components for viewing weather information, receiving crop recommendations, and interacting with the chatbot.

- Application Layer:

The application layer contains the core business logic of the Agricultural Assistance App. It encompasses modules for crop recommendation generation, chatbot interactions, user authentication, and other essential functionalities.

- Data Access Layer:

This layer facilitates communication between the application layer and the database layer. It manages the connectivity to the database, retrieval of data, and interaction with the data storage components.

- Database Layer:

The database layer stores critical data, including real-time weather data, user information, and historical records. It ensures efficient data storage and retrieval to support the app's functionalities.

Key Considerations:

- Scalability: The layered architecture allows for scalability, with each layer being scalable independently based on demand.
- Modularity: Each layer represents a distinct set of functionalities, promoting modularity and ease of maintenance.
- Separation of Concerns: The architecture ensures a clear separation of concerns between user interface, application logic, data access, and database management.

This layered architecture provides a structured and organized approach to designing the Agricultural Assistance App, aligning with the design goals previously outlined. It promotes maintainability, scalability, and efficient data flow through the various components of the system.

6.4. System Design

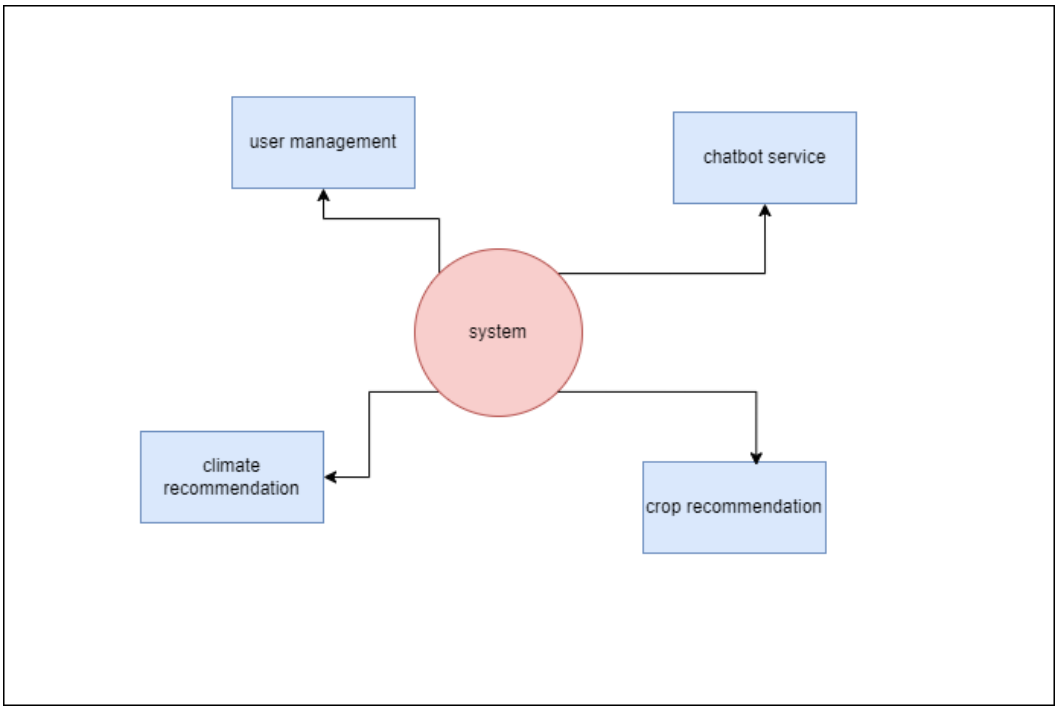


Figure 8 Low-level Design/ System Design

6.4.1. Choice of Design Paradigm (OOAD/ SSADM)

The system development process of EcoGrow, follows the Object-Oriented Analysis and Design Methodology (OOADM). This conclusion is drawn from the inclusion of components such as a Class Diagram, Sequence Diagram, and references to object-oriented concepts in the design and development sections. If document includes principles like encapsulation, inheritance, and polymorphism, and emphasizes the modeling of real-world entities using objects and classes, then it aligns with the Object-Oriented paradigm.

6.4.2. **Component Diagram**

The component diagram illustrates the high-level components and their interactions within the EcoGrow system, including modules for user authentication, weather data retrieval, crop recommendation algorithms, and user interface components.

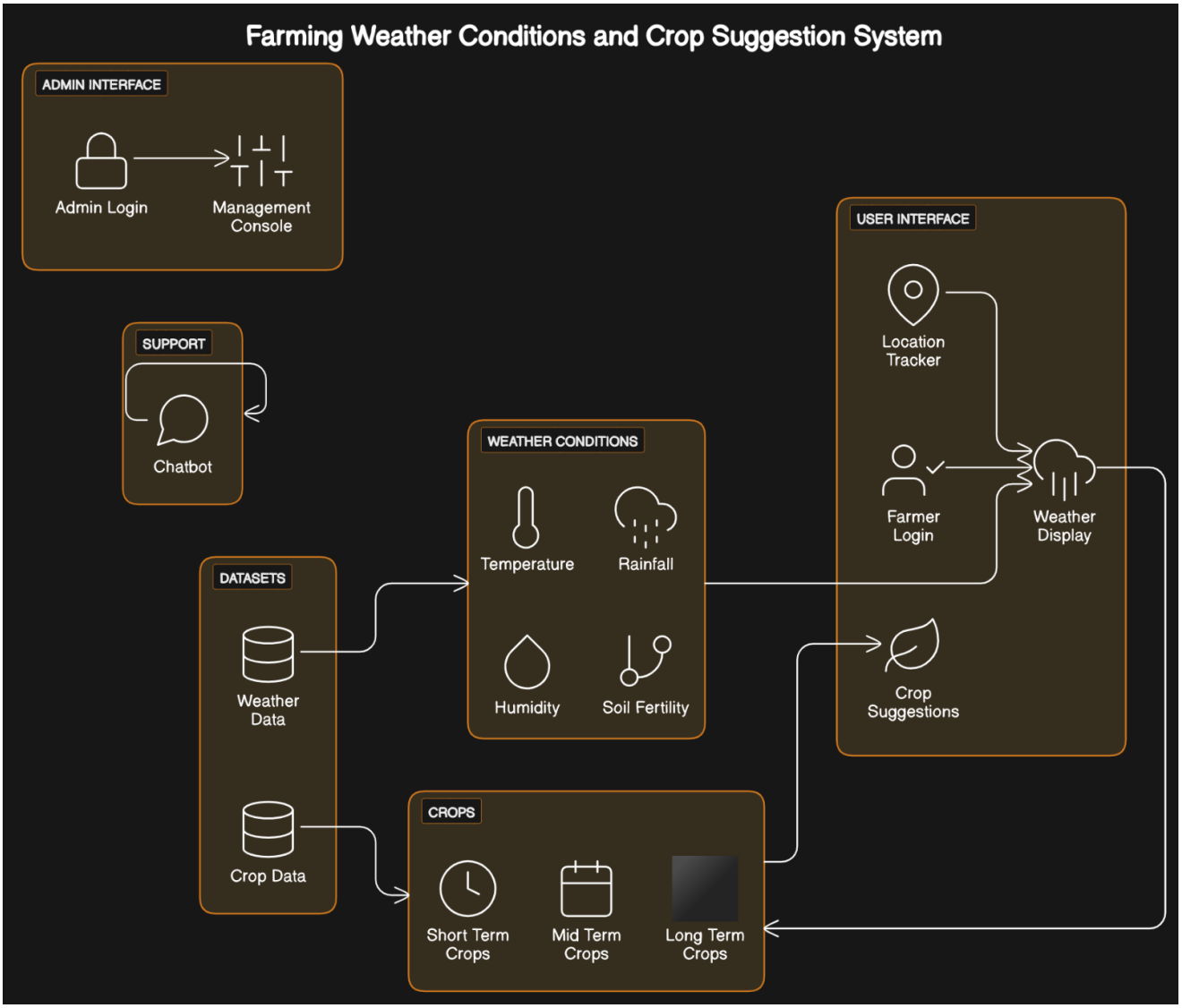


Figure 9 Component Diagram

6.4.3. **Class Diagram**

A Class Diagram is a fundamental component of Object-Oriented Analysis and Design (OOAD). It illustrates the static structure of a system by showing classes, their attributes, methods, and relationships. Classes represent real-world entities or concepts, and the relationships depict how these entities are connected.

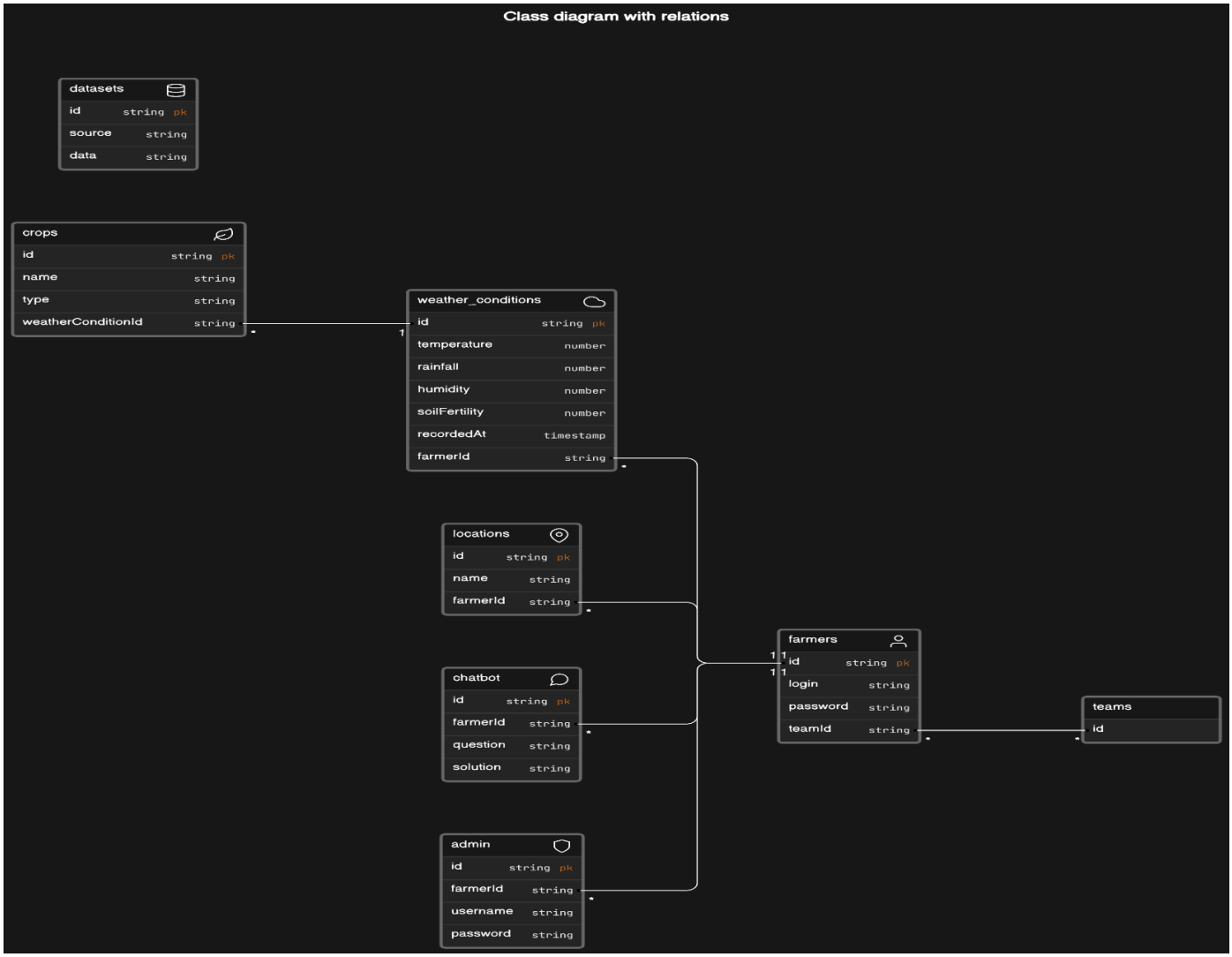


Figure 10 - Class diagram

6.4.4. **Sequence Diagram**

Sequence Diagrams are another key aspect of OOAD. They depict the interactions between different objects or components over time. These diagrams show the sequence of messages exchanged between objects to achieve a particular functionality. It emphasizes the dynamic aspects of a system, focusing on how objects collaborate to fulfill use cases.

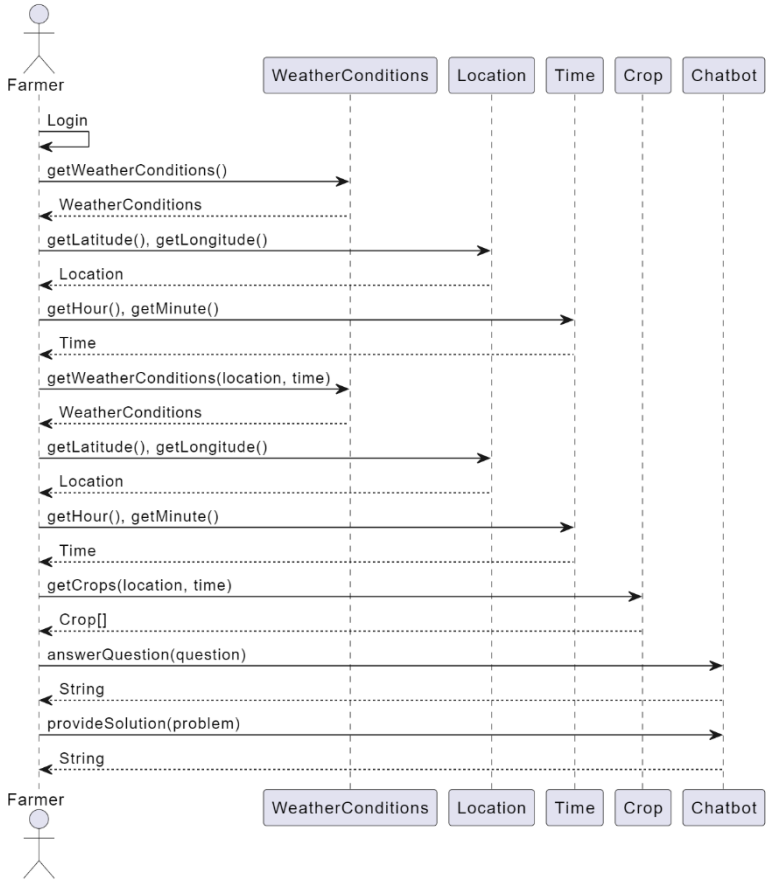


Figure 11 - Sequence Diagram

6.4.5. **Algorithm Design**

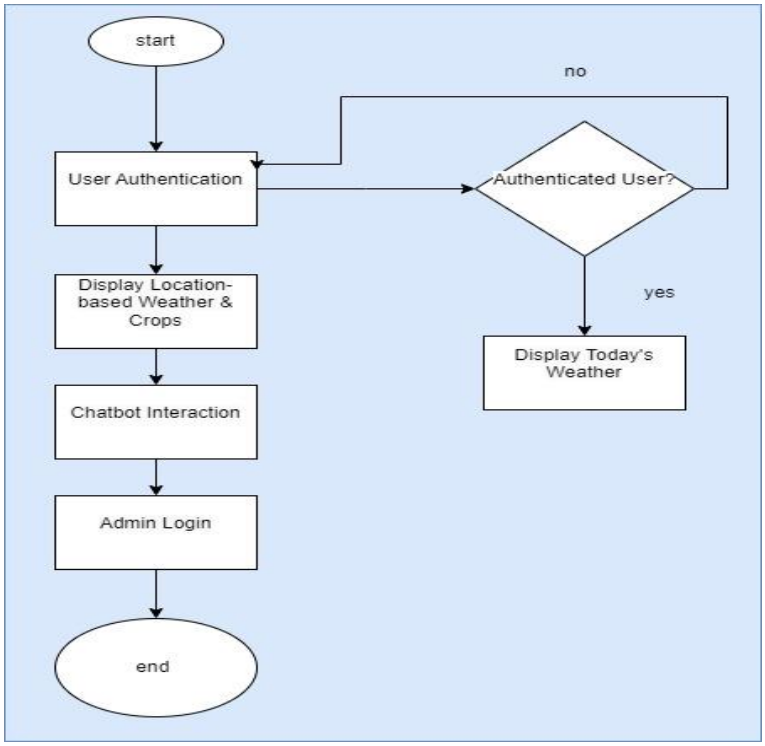


Figure 12 Algorithmic Design

- **Algorithmic Analysis**

In the Agricultural Assistance App, the "Dynamic Crop Affinity Algorithm" has been developed to enhance crop recommendations based on real-time weather data and historical patterns. This algorithm processes current weather conditions, analyzes historical data, and adjusts crop recommendations dynamically. It undergoes rigorous analysis, including accuracy assessment, performance metrics evaluation, sensitivity analysis, and comparative studies. By tailoring recommendations to individual farmers and considering changing environmental conditions, the algorithm contributes to the app's effectiveness in supporting farmers with accurate and adaptive crop suggestions.

6.4.6. User Interface (UI Design)

Creating visual representations like low-level fidelity wireframes and high-level fidelity prototypes typically requires graphic design tools. Since I can't generate visual content directly, I'll describe how you can structure these diagrams:

6.4.6.1. Low-Level Fidelity Wireframe Diagram:

Low-fidelity wireframes provide a basic, abstract representation of the user interface. You can use simple shapes and placeholders to convey the layout and structure of each screen.

- Login Screen:
    - Placeholder for logo
    - Input fields for username and password
    - Login button
    - Link to register
- Weather Display:
    - Section for real-time weather information
    - Dropdown or map for location selection
    - Button to view crop recommendations
    - Simple navigation menu
- Chatbot Interaction:
    - Chat window
    - Input field for user queries
    - Responses from the chatbot
    - Options for common queries or actions
- Tools for Creating Prototypes:
    - Paper and Pen: Quickly sketch out ideas on paper.

- Figma: Use digital tools with pre-built UI elements.

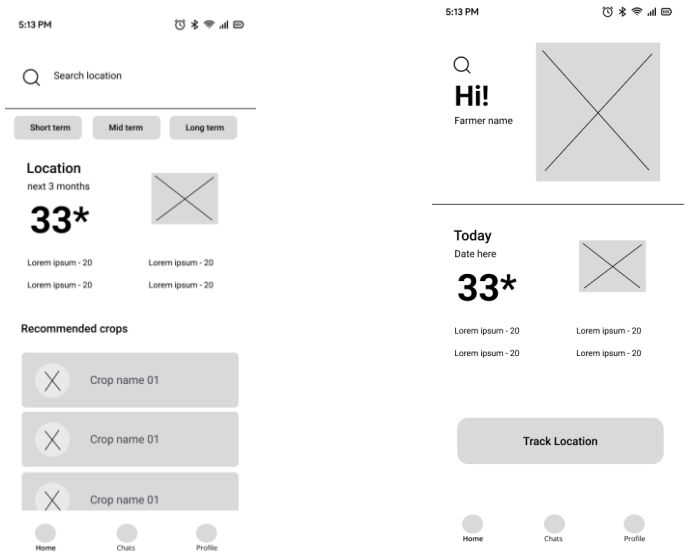


Figure 13 low level fidelity wireframe diagram

6.4.6.2. High-Level Fidelity Prototype:

High-fidelity prototypes provide a more detailed and visually polished representation of the user interface. Use actual design elements, colors, and images to create a realistic preview of the app.

- Login Screen:
  - App logo in the top-left corner
  - Stylish input fields with labels for username and password
  - Engaging background image related to agriculture
  - A sleek "Login" button with contrasting color
- Weather Display:
  - Dynamic weather widget with icons for temperature, humidity, etc.
  - Interactive map for location selection
  - High-quality images representing different weather conditions
  - Smooth transitions and animations
- Chatbot Interaction:
  - Conversational chat interface with avatars
  - User-friendly input field with a microphone for voice input
  - Rich media responses, including images or icons
  - Clear and concise navigation options
- Tools for Creating Prototypes:
  - Figma- Create detailed and interactive prototypes.

Remember to tailor the wireframes and prototypes to the specific features and functionalities of the Agricultural Assistance App, incorporating feedback from stakeholders for continuous improvement.

- Figma link : EcoGrow UI Design

6.4.7. User Experience

User experience flows outline the user journey through the application, mapping out the sequence of interactions and tasks performed by users to achieve their objectives, such as accessing weather information or receiving crop recommendations.

6.4.8. Process flow chart

Process flow charts, I choose to draw an activity diagrams describe the systematic flow of processes within the EcoGrow system, illustrating the steps involved in key operations such as data retrieval, analysis, and presentation.

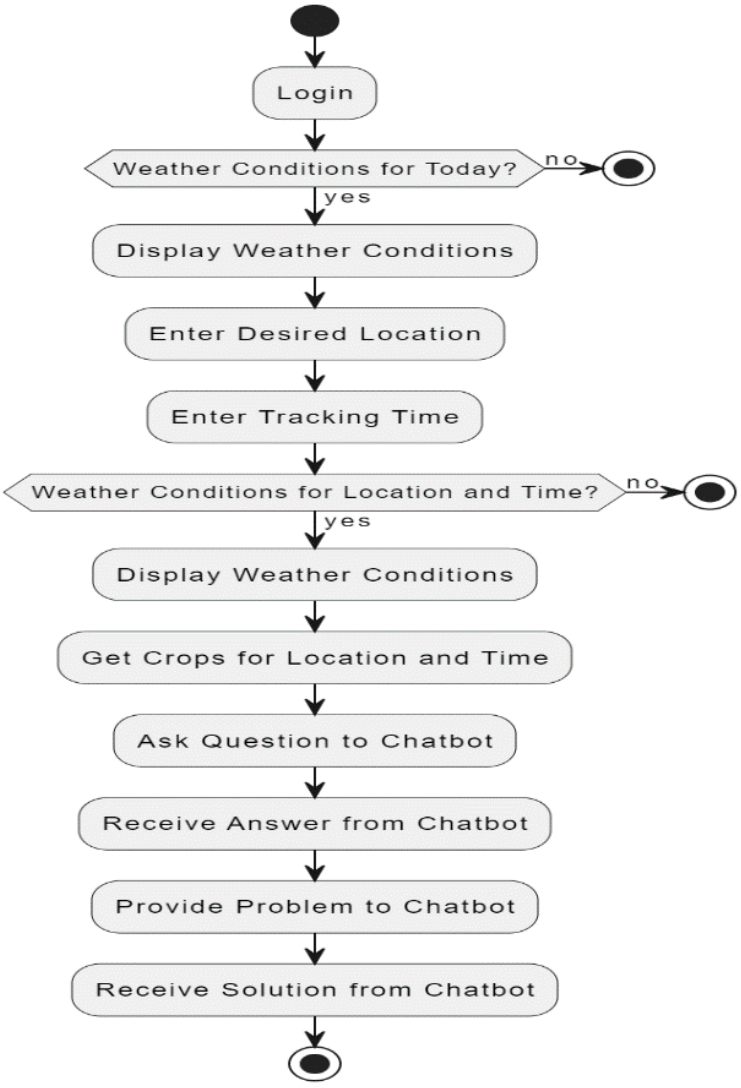


Figure 14 System Process Flow Chart

6.5. Chapter Summary

Chapter 06 of the project delves into the design phase, which is crucial for shaping the Agricultural Assistance App's functionality and user experience. It outlines design goals, architectural foundations, low-level design decisions, algorithmic considerations, flow charts, and user interface design. The chapter emphasizes the pivotal role of design in realizing the app's transformative impact on agriculture, ensuring effectiveness, scalability, and user friendliness. By providing a blueprint for development, Chapter 6 sets the stage for creating a seamless and robust app that aligns with the project's goals and enhances agricultural practices.

CHAPTER 07: IMPLEMENTATION

7.1. Chapter Overview

Chapter 07 of the project marks the transition from the design phase to the final implementation, where theoretical concepts and design specifications materialize into a fully functional system. This chapter provides a comprehensive overview of the steps taken to bring the agricultural assistance app to life, emphasizing technology selection, development tools, and the building blocks of the system.

7.2. Technology Selection

7.2.1. Technology Stack

The choice of technology stack plays a crucial role in shaping the functionality, performance, and scalability of the agricultural assistance app. technology stack is meticulously selected to cater to the specific requirements of each system component.

- Frontend - React Native:

React Native is chosen as the frontend framework, ensuring a cross-platform, efficient, and visually appealing user interface. This technology allows for the development of a seamless mobile application that can run on Android platforms, minimizing development efforts and maximizing user reach.

- Middle Tier - Machine Learning Part:

The machine learning component of the agricultural assistance app is crucial for enhancing farming practices by providing personalized crop recommendations based on real-time weather data. It leverages sophisticated algorithms to analyze historical weather patterns and their correlation with crop yields. Initially, extensive datasets containing weather parameters and crop yields are collected and preprocessed. Various machine learning algorithms, such as decision

trees, random forests, and neural networks, can be employed for training the model. Once trained, the model undergoes rigorous evaluation to ensure accuracy and reliability. In operation, the system accepts input data on current weather conditions and utilizes the trained model to predict suitable crops for cultivation. Continuous learning and adaptation mechanisms ensure ongoing effectiveness by incorporating new data and refining the model's performance over time. Overall, by harnessing the power of machine learning, the app aims to empower farmers with valuable insights to optimize farming practices and improve crop yields.

- Backend Technologies - Node.js:

Node.js is chosen as the backend technology for its event-driven, non-blocking I/O model, ensuring a responsive and scalable server-side architecture. This aligns with the requirements of real-time updates and interactions, making it ideal for handling data processing, user requests, and communication between system components. The technology stack, which integrates React Native for the frontend and advanced machine learning tools for the middle tier, is designed to harmonize the diverse functionalities of the agricultural assistance app. This chapter provides insights into the rationale behind each technology choice and emphasizes the synergy achieved by combining them for a robust implementation. The subsequent sections will detail the implementation process, addressing challenges, solutions, and development milestones. The focus remains on maintaining code quality, ensuring system stability, and paving the way for a transformative user experience in smart farming.

7.2.2. Data Selection

In the development of the agricultural assistance app, the selection of an appropriate dataset is fundamental to the accuracy and effectiveness of the machine learning algorithms. The chosen dataset encompasses key climate parameters and a diverse range of crops to ensure comprehensive coverage. The climate parameters include temperature, humidity, and rainfall, while the crops include rice, chickpea, kidney beans, moth beans, mung beans, blackgram, soya beans, pomegranate, banana, papaya, mango, bean, watermelon, lemon, orange, coconut, kurakkan, coffee, and maize. This diverse dataset serves as the foundation for training the machine learning models to provide precise and relevant crop recommendations based on real-time weather conditions.

7.2.3. Selection of Development Framework

No specific development frameworks are mentioned in this section, indicating that the project does not rely on additional development frameworks beyond those already specified in the technology stack (React Native, machine learning frameworks, and Node.js).

7.2.4. **Programming Language**

The programming languages chosen for the implementation of the agricultural assistance app are not explicitly stated in this section. However, based on the specified technology stack, it can be inferred that JavaScript is the primary language for both the frontend (React Native) and the backend (Node.js). The choice of JavaScript allows for a unified development approach and seamless communication between the frontend and backend components.

7.2.5. **Libraries**

Two specific libraries are mentioned in this section, each serving a distinct purpose:

- react-native-gifted-chat:

This library is employed for implementing chat functionality within the React Native frontend. It provides pre-designed UI components and functionalities that streamline the development of chat features, ensuring a smooth and engaging user experience. The decision to use react-native-gifted-chat suggests a focus on efficiency and user-friendly chat interactions.

- expo-linear-gradient:

The expo-linear-gradient library is utilized for incorporating gradient effects in the app's user interface. Gradients add a visually appealing and modern touch to the design, enhancing the overall aesthetics of the application.

These libraries are selected based on their compatibility with the chosen technology stack and their ability to expedite the development process while maintaining a high standard of user interface design.

7.2.6. **IDE**

Visual Studio Code (VS Code) is chosen as the integrated development environment for the implementation of the agricultural assistance app. VS Code is a widely used, open-source code editor known for its versatility, extensive plugin ecosystem, and robust support for JavaScript-based development. The decision to use VS Code aligns with the goal of leveraging a popular and efficient development environment, fostering collaboration and code consistency among the development team.

In summary, the technology choices, dataset selection, and library preferences outlined in this section collectively contribute to the foundation of the app's implementation. The decisions are made with a focus on functionality, user experience, and the seamless integration of diverse components. As the development progresses, these choices will play a pivotal role in achieving the envisioned goals of the agricultural assistance app.

7.2.7. Summary of Technology Selection

Aspect	Chosen Technology	Justification and Benefits
Frontend Framework	React Native	Cross platform capability, cost effective, unified development approach
Middle Tier	Machine Learning (Python)	Provides intelligent crop recommendations based on climate parameters
Backend Technology	Node.js	User requests and data processing that is event-driven, scalable, and effective
Programming Language	JavaScript	Facilitates easy interaction between components of the front and back ends.
Libraries	react-native-gifted-chat	Streamlines chat functionality and enhances user interactions
	expo-linear-gradient	Adds gradient effects and improves UI aesthetics.
IDE	Visual Studio Code (VS Code)	Responsive, efficient, and compatible with JavaScript based application.

Table 18 Summary of Technology Selection

7.3. Implementation of Core Functionalities

- Weather Data Retrieval:

The core functionality of retrieving weather data is vital for providing accurate and timely information to users. Leveraging React Native for the frontend, the app can efficiently fetch real-time weather data from external APIs. This functionality ensures a seamless user experience by

providing users with up-to-date weather conditions for their selected locations. The code snippet below demonstrates how the system fetches weather data using React Native:

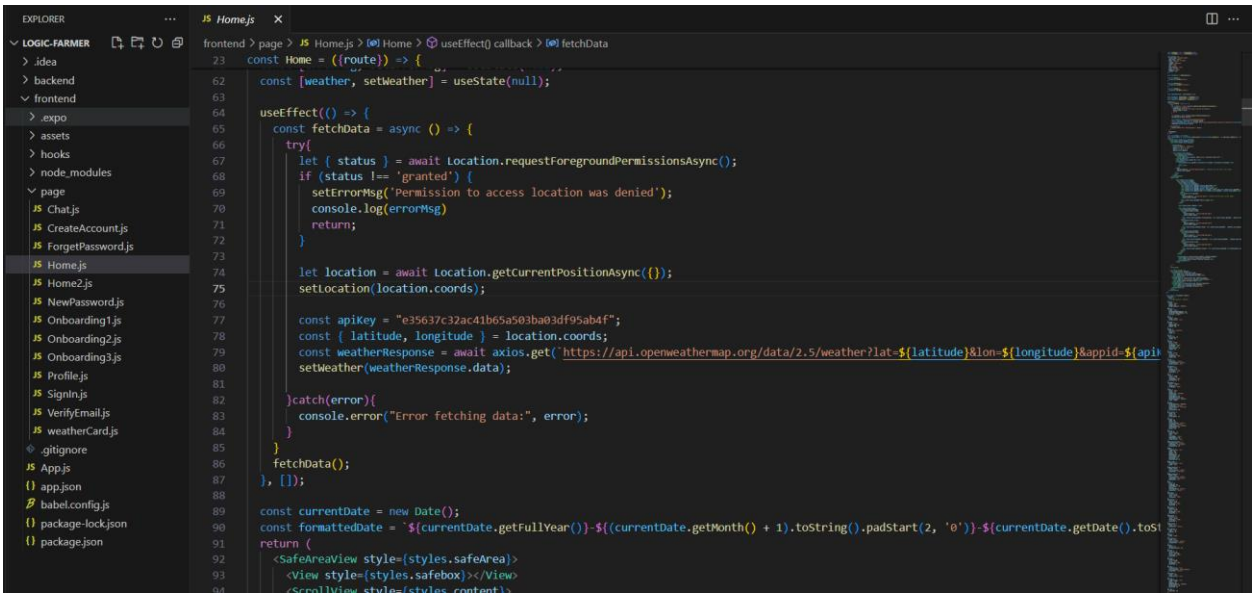


Figure 15 fetch real-time weather data

- Crop Recommendation Algorithm:

The implementation of a machine learning-based crop recommendation algorithm enhances the app's intelligence and utility. Powered by machine learning frameworks in the middle tier, the app analyzes weather data to generate personalized crop recommendations based on climate parameters. This functionality empowers users with actionable insights for optimizing their farming practices. Although specific machine learning frameworks are not mentioned, the integration of machine learning algorithms ensures the app's ability to provide valuable recommendations.

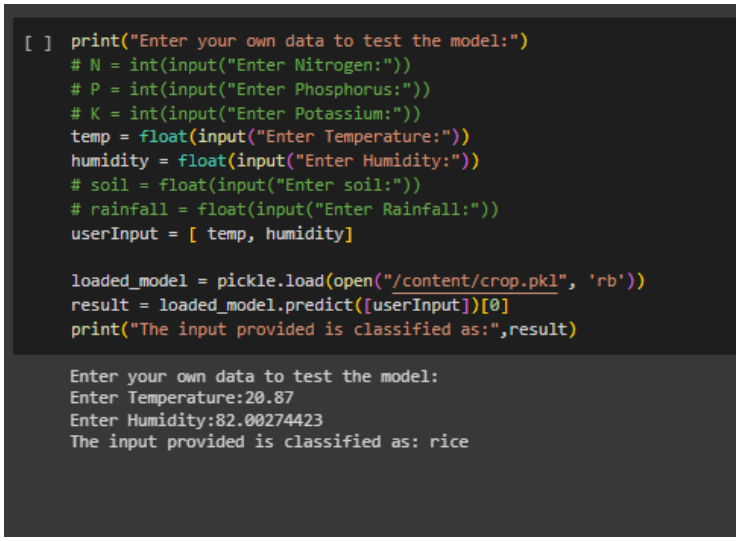
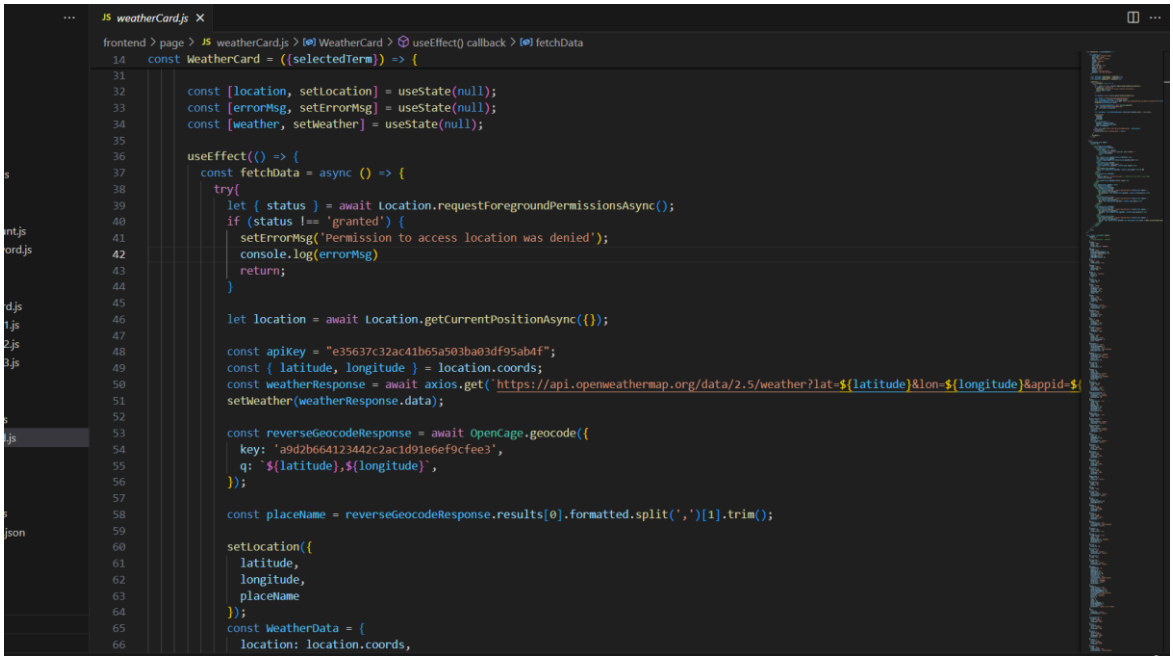


Figure 16 ML (Self-Composed)

7.3.1. **Implementation of APIs**

Weather API Integration:

Integrating with external weather APIs enables the app to fetch real-time weather data for user-selected locations. Utilizing React Native for the frontend and Node.js for the backend, the



system efficiently communicates with external APIs to retrieve weather information. This integration ensures that users receive accurate and reliable weather updates. Below is an code snippet demonstrating the integration with a weather API using Node.js:

Crop Recommendation API:

The implementation of a crop recommendation API allows the system to receive weather data and return personalized crop recommendations. By leveraging Node.js for the backend, the app efficiently processes user requests and generates crop recommendations based on machine learning algorithms. This API ensures that users receive tailored recommendations to optimize their farming practices.

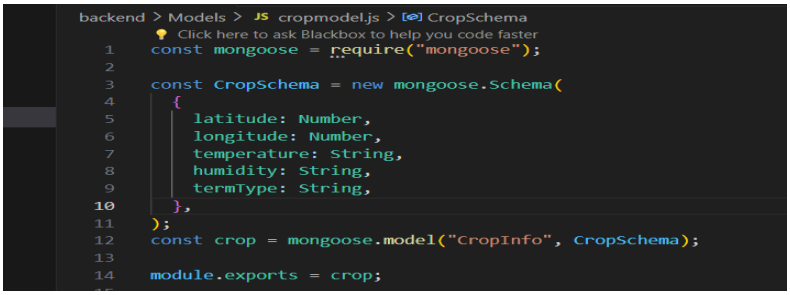


Figure 18 Personalized Crop

Figure 17 fetch real-time weather data for user-selected locations

7.4. User Interface

Design Mockups:

Designing an intuitive and visually appealing user interface is crucial for enhancing user engagement and satisfaction. Using React Native for cross-platform compatibility, the app implements design mockups created with tools (Figma). These mockups serve as blueprints for implementing UI components and interactions, ensuring a cohesive and user-friendly interface.

Figma link : EcoGrow UI Design

Interactive Elements:

Implementing interactive elements such as buttons, forms, and navigation enhances the usability of the app. With React Native, the system creates UI components that allow users to easily navigate the app, input data, and interact with various features. This approach ensures that users can intuitively access and utilize the app's functionalities. Code Screenshots of implementing using React Native will be show in the Appendix.

7.5. Chapter Summary

Chapter 07 delves into the implementation phase of the agricultural assistance app, highlighting a strategic blend of cutting-edge technologies to ensure a seamless user experience. React Native is selected for the frontend, offering a cost-effective solution for Android platforms, while Node.js brings scalability and efficiency to the backend. Machine learning integration elevates the app's capabilities, providing intelligent crop recommendations based on vital climate parameters. Libraries such as react-native-gifted-chat and expo-linear-gradient enhance functionality and aesthetic appeal, fostering user engagement. Carefully chosen datasets cover essential climate parameters and a diverse range of crops, ensuring robust recommendations for farmers. Overall, the emphasis on user-friendly design, machine learning integration, and cross-platform compatibility positions the app to empower farmers with actionable insights and personalized recommendations, making a significant impact on the agricultural landscape.

CHAPTER 08: TESTING

8.1. Chapter Overview

Chapter 8 focuses on the comprehensive testing process of EcoGrow, ensuring the reliability, accuracy, and performance of the agricultural assistance app. This chapter outlines various testing methodologies, objectives, and criteria to validate the functionality and effectiveness of the system. Emphasizing both functional and non-functional aspects.

8.2. Objectives and Goals of Testing

The primary objectives of testing EcoGrow encompass a range of goals aimed at guaranteeing the app's functionality, usability, and security. These objectives include:

- Validating the accuracy and reliability of weather data retrieval and crop recommendations.
- Verifying the seamless performance of core features such as user authentication, weather display, and chatbot interaction.
- Assessing the responsiveness and user-friendliness of the application across different devices and platforms.
- Identifying and rectifying any potential security vulnerabilities to safeguard user data and ensure privacy.
- Ensuring compliance with non-functional requirements such as real-time updates, scalability, and data security standards.

8.3. Testing Criteria

The testing criteria for EcoGrow are meticulously defined to cover various functional and non-functional aspects critical to the app's performance. These criteria include:

- Accuracy of weather data retrieval and crop recommendations, measured against ground truth data.
- Usability and intuitiveness of the user interface, evaluated through user testing and feedback.
- Performance under varying load conditions, including stress testing to assess system robustness.
- Security measures to safeguard user data, including encryption protocols, authentication mechanisms, and vulnerability assessments.
- Compliance with regulatory standards and industry best practices to ensure legal and ethical integrity.

8.4. Model Testing

Since EcoGrow incorporates machine learning for crop recommendations, model testing is essential to evaluating the effectiveness and performance of the recommendation algorithm. This includes evaluating metrics such as accuracy, precision, recall, and F1 score.

Additionally, visualizing the model's performance using tools like confusion matrices and AUC/ROC curves provides insights into its predictive capabilities.

From the Model testing I have chosen the best hybrid Models from training to use as its Accuracy.

```
[ ]  #Training the Model
     target ='label'
     X_train, X_test, y_train, y_test = read_in_and_split_data(df, target)
     models = GetModel()
     names,results = fit_model(X_train, y_train,models)

     LR: 0.293825 (0.023715)
     LDA: 0.508125 (0.029792)
     KNN: 0.603388 (0.045663)
     CART: 0.603994 (0.043521)
     NB: 0.652858 (0.045603)
     SVM: 0.519422 (0.035627)

[ ]  ScaledModel = NormalizedModel('minmax')
     name_results = fit_model(X_train, y_train, ScaledModel)
```

Table 19 Test of Model Accuracy

8.4.1. Confusion Matrix

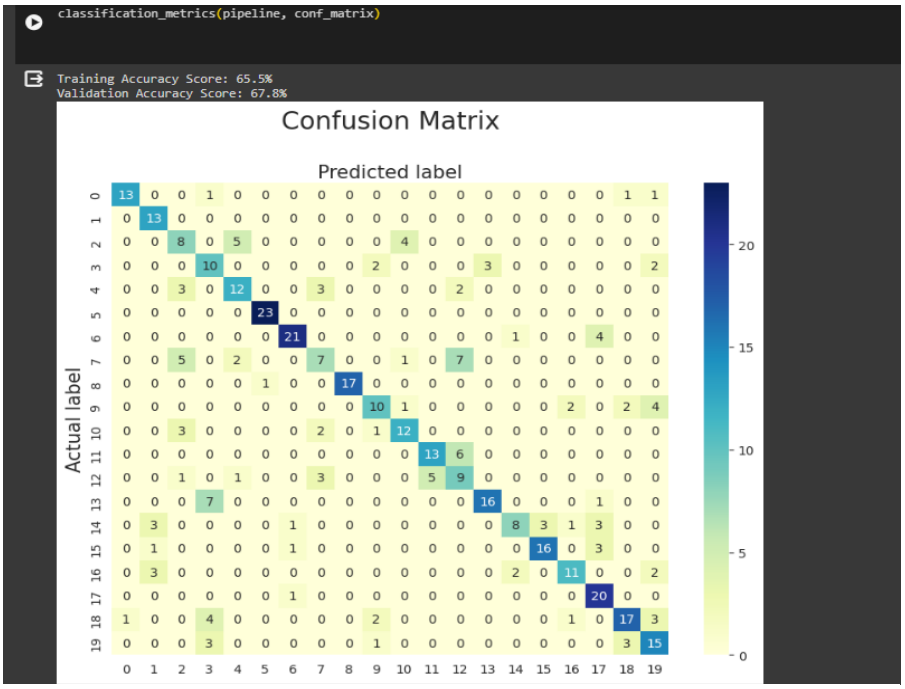


Figure 19 ML - Confusion Matrix

Crops	Precision	Recall	F1 – Score	Support
Bean	0.93	0.81	0.87	16
Lemon	0.65	1.00	0.79	13
Chickpea	0.96	1.00	0.98	23
Coconut	0.88	0.81	0.84	26

Table 20 Few Test Results of the Knn Model for Crop Selection

8.4.1.1. Accuracy: The overall accuracy of the model in predicting crop recommendations.

8.4.1.2. F1 Score: Harmonic mean of precision and recall, indicating the balance between false positives and false negatives.

8.4.1.3. Precision: Proportion of true positive predictions among all positive predictions.

8.4.1.4. Recall: Proportion of true positive predictions among all actual positive instances.

8.4.2. AUC/ROC Curve

The Area Under the Curve (AUC) of the Receiver Operating Characteristic (ROC) curve measures the model's ability to distinguish between different classes. A higher AUC value indicates better predictive performance. Since the KNN model is averaged to implement the ensemble model, the evaluation test results were derived from the average of each test.

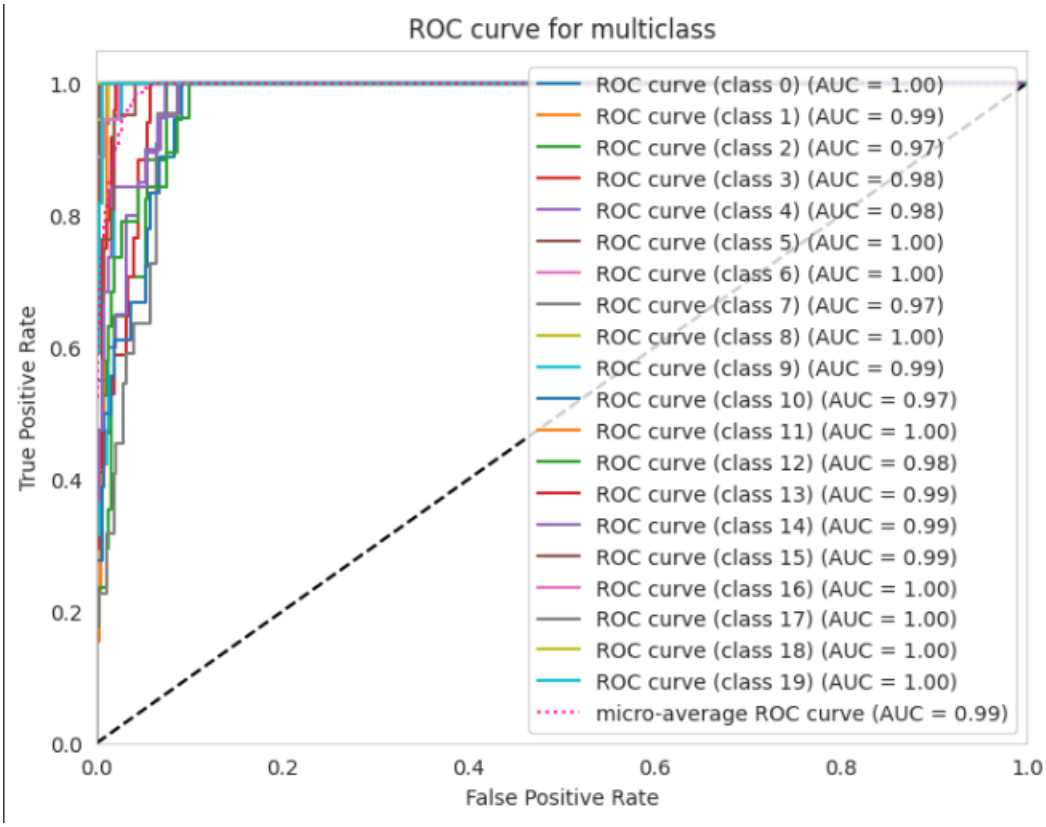


Figure 20 AUC/ROC Curve

8.5. Benchmarking

Benchmarking entails comparing EcoGrow's performance against existing solutions or industry standards to assess its effectiveness and identify areas for improvement. This involves evaluating factors such as data accuracy, recommendation accuracy, and user satisfaction metrics.

The formulas to calculate precision and recall are:

Precision = T P/(T P + F P)

Recall = T P/(T P + F N)

where, TP = True Positives, FP = False Positives, and FN = False Negatives.

Existing Model	Accuracy	F1 – Score	Precision	Recall
EcoGrow	**0.80**	**0.82**	**0.86**	**0.88**
(Dahiphale et al., 2023)	0.70	0.70	0.73	0.76
(Chana et al., n.d.)	0.65	0.69	0.65	0.66
(Kamatchi and Parvathi, 2019)	0.76	0.78	-	-
(Shem Juma and Kelonye Beru, 2021)	0.78	0.84	0.81	0.81

Table 22 Benchmarking results against traditional machine learning approaches

8.6. Functional Testing

Functional testing ensures that each feature of EcoGrow performs as expected, meeting user requirements and specifications.

Test Case	Description	Expected Outcome	Actual Outcome	Test Result
EcoGrow Model Testing				
Weather Data Processing	Validate that weather data is processed accurately by the ML model, considering temperature, rainfall and humidity.	The ML model accurately processes weather data and extracts relevant features for crop recommendation.	The ML model successfully processes weather data and extracts relevant features for crop recommendation.	Pass

Crop Recommendation Generation	Verify that the ML model generates crop recommendations based on processed weather data, categorizing them into short term, mid-term, and long term.	The ML model provides accurate crop recommendations based on current weather conditions, categorized appropriately.	The ML model accurately generates crop recommendations based on the processed weather data and categorizes them correctly.	Pass
Model Evaluation Metrics	Evaluate the performance of the ML model using metrics such as accuracy, precision, recall, and F1 score.	The ML model achieves high values for accuracy, precision, recall, and F1 score, indicating its effectiveness in providing crop recommendations.	The ML model's performance metrics meet or exceed the expected thresholds, demonstrating its effectiveness in providing accurate crop recommendations.	Pass
EcoGrow Mobile Application Testing				
User Authentication	Verify that users can log in securely with valid credentials on the mobile application.	Users are authenticated and granted access to personalized information upon successful login.	Users are able to log in securely and access their personalized information.	Pass
Weather Data Retrieval	Check if the mobile app displays real-time weather conditions accurately, including temperature, rainfall and humidity.	The app shows real-time weather data for the user's location with accurate information on temperature, rainfall and humidity.	The app successfully retrieves and displays real-time weather conditions for the user's location.	Pass

Chatbot Interaction	Test the interaction with the chatbot to verify that users can get solutions to farming-related queries.	Users can interact with the chatbot effectively, receiving accurate solutions to their farming-related queries in a timely manner.	The chatbot responds promptly to user queries, providing accurate solutions and guidance on farming-related issues.	Pass
EcoGrow Mobile Application Backend Testing				
User Authentication API	Test the API endpoint for user authentication to ensure that users can log in securely with valid credentials.	The user authentication API verifies user credentials and returns a token upon successful authentication.	The user authentication API successfully validates user credentials and returns a token for authenticated users.	Pass
Weather Data Retrieval API	Validate the API endpoint responsible for retrieving weather data to ensure accurate and timely information delivery.	The weather data retrieval API fetches real-time weather conditions accurately for the specified location.	The weather data retrieval API successfully retrieves real-time weather conditions for the specified location.	Pass
Chatbot Interaction API	Test the API endpoint for chatbot interaction to ensure effective communication between users and the chatbot feature.	The chatbot interaction API allows users to send queries and receive appropriate responses from the chatbot in real-time.	The chatbot interaction API facilitates seamless communication between users and the chatbot, delivering accurate responses to queries.	Pass

Table 23 Functional testing of the models and Moblie application

8.7. Module and Integration Testing

Module testing focuses on testing individual components or modules of EcoGrow, while integration testing evaluates the interaction between different modules to ensure seamless functionality across the system.

Test Case	**Description**	**Expected Outcome**	**Actual Outcome**	**Test Result**
User Authentication Module Testing	Test the user authentication module to ensure that users can log in securely with valid credentials.	Users can log in securely with valid credentials, and unauthorized access is prevented.	Users are able to log in securely with valid credentials, and unauthorized access is prevented.	Pass
Weather Data Retrieval Module Testing	Validate the weather data retrieval module to ensure accurate and timely delivery of weather information.	The weather data retrieval module fetches real-time weather conditions accurately for the specified location.	The weather data retrieval module successfully retrieves real-time weather conditions for the specified location.	Pass
Crop Recommendation Generation Module Testing	Test the crop recommendation generation module to ensure that it provides accurate recommendations based on current weather conditions.	The crop recommendation generation module accurately recommends suitable crops based on current weather conditions, categorized as short-term, mid-term, and long-term.	The crop recommendation generation module effectively recommends suitable crops based on current weather conditions, categorized as short-term, mid-term, and long-term.	Pass
Chatbot Interaction Module Testing	Validate the chatbot interaction module to ensure effective communication between users and the chatbot feature.	The chatbot interaction module allows users to send queries and receive appropriate responses from the chatbot in real-time.	The chatbot interaction module facilitates seamless communication between users and the chatbot, delivering	Pass

			accurate responses to queries.	
Integration Testing	Test the interaction between different modules to ensure seamless functionality across the system.	Modules integrate smoothly, and data flow between them is accurate and efficient, resulting in seamless system operation.	Modules integrate seamlessly, and data flow between them is accurate and efficient, ensuring seamless system operation.	Pass

Table 24 Module integration testing of the system

8.8. Non-Functional Testing

Non-functional testing assesses aspects such as accuracy, performance, load balance, scalability, and security.

8.8.1. Accuracy Testing:

Accuracy testing is crucial for verifying the precision and reliability of weather data and crop recommendations provided by EcoGrow. This process involves comparing the actual weather conditions and crop suggestions with expected values based on historical data and expert knowledge. The expected outcome is for the weather data to align closely with real-time conditions, while crop recommendations should match the climatic conditions of the selected location accurately. To ensure accuracy, thorough comparisons between the system's output and verified sources of weather data are conducted. The test result is considered successful if the system consistently provides accurate weather data and relevant crop recommendations within acceptable margins of error.

8.8.2. Performance Testing:

Performance testing aims to evaluate the responsiveness and efficiency of EcoGrow under varying load conditions, including normal usage patterns and peak load situations. By simulating different usage scenarios and measuring system response times and resource utilization, this testing assesses whether the system meets performance expectations. The expected outcome is for the system to respond promptly to user interactions, load weather data and crop recommendations quickly, and maintain stability even under heavy usage. Performance tests are conducted using load generators to simulate user activity, and system performance metrics such as response time and throughput are monitored. The test result is considered successful if the system consistently meets performance targets and demonstrates adequate scalability to handle increased loads without performance degradation.

8.8.3. Load Balance and Scalability:

Load balance and scalability testing evaluate how well EcoGrow distributes user requests and scales resources to accommodate growing user demands. By gradually increasing the number of concurrent users or requests, this testing assesses whether the system evenly distributes incoming requests, allocates resources dynamically, and maintains performance levels as user demand grows. Stress tests are conducted to ramp up user load and observe how the system behaves under high traffic conditions, identifying any bottlenecks or performance degradation. The test result is considered successful if the system exhibits balanced resource utilization, scales seamlessly to handle increased loads, and maintains performance levels without interruptions or failures.

8.8.4. Security Testing:

Security testing focuses on identifying and mitigating security vulnerabilities within EcoGrow to protect user data and ensure compliance with privacy regulations. Through vulnerability assessments, penetration testing, and code reviews, this testing aims to uncover potential security flaws and assess the effectiveness of existing security measures. The expected outcome is for the system to have robust authentication mechanisms, secure data transmission channels, and protection against common security threats. Security audits and assessments are conducted to identify vulnerabilities, implement necessary security controls such as encryption and access controls, and validate the effectiveness of security measures. The test result is considered successful if the system demonstrates resilience against security threats, complies with industry best practices, and safeguards user data from unauthorized access or manipulation.

8.9. Limitations of the testing process

Despite the thoroughness of the testing process, certain limitations may impact the validity and comprehensiveness of the results. These limitations could include constraints in the availability of data for model training, which may affect the accuracy of predictions and recommendations. Additionally, biases present in testing datasets could influence the outcomes of the testing process, leading to skewed results. Moreover, limitations in testing environments, such as differences in hardware configurations or network conditions, may affect the reproducibility of test results across different environments. While efforts are made to mitigate these limitations, it's essential to acknowledge their presence and consider their potential impact on the overall testing outcomes.

8.10. Chapter Summary

Chapter 08 provides a comprehensive overview of the testing process conducted for EcoGrow, covering objectives, methodologies, criteria, and limitations. Through rigorous testing and validation, EcoGrow aims to ensure the reliability, usability, and effectiveness of its agricultural assistance solution. By addressing key testing considerations and limitations, the chapter emphasizes the commitment to delivering a robust and user-friendly application that empowers farmers with valuable insights and recommendations for improving their farming practices.

CHAPTER 09: EVALUATION

9.1. Chapter Overview

Chapter 09 delves into the critical evaluation of the EcoGrow agricultural assistance app, focusing on assessing its effectiveness, reliability, and alignment with project objectives. This chapter provides a comprehensive overview of the evaluation methodology, criteria, self-evaluation, selection of evaluators, evaluation results, and limitations encountered during the evaluation process.

9.2. Evaluation Methodology and Approach

The evaluation methodology adopts a multifaceted approach, incorporating expert opinions, focus group testing, and self-evaluation. This approach ensures a comprehensive assessment of EcoGrow from both technical and end-user perspectives, encompassing various evaluation criteria and subtopics.

In order for better understanding of the project domain, technical aspects, and the recommended approach to its architecture, a video demonstration has been created.

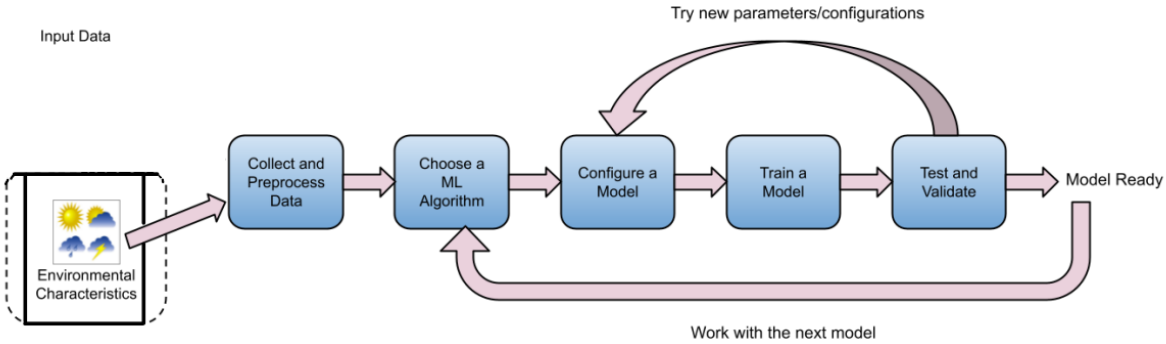


Figure 21 Methodology of Choosing the best ML Model

9.3. Evaluation Criteria

The evaluation criteria are carefully defined to cover both functional and non-functional aspects of EcoGrow, including accuracy, reliability, usability, performance, security, and compliance with requirements specified in the software requirements specification (SRS).

Evaluation Criteria	Purpose
Concept	Assess the relevance and potential impact of EcoGrow on the agricultural sector. This criterion aims to determine whether EcoGrow addresses critical challenges faced by farmers and provides valuable assistance in optimizing farming practices. Evaluators will examine the extent to which the app aligns with the needs and priorities of farmers, considering factors such as usability, accessibility, and relevance of features. They will also assess how well EcoGrow caters to different agricultural contexts and regions.
Solution	Evaluate the effectiveness of EcoGrow's features, functionality, and user interface design. This criterion focuses on assessing how well the app translates the concept into a user-friendly and intuitive platform for farmers. Evaluators will analyze the app's features and functionalities, assessing their alignment with user requirements and industry standards. They will also evaluate the user interface design, looking for clarity, consistency, and ease of navigation. Feedback on the solution will help identify areas for improvement and refinement.
Scope	Determine whether EcoGrow effectively addresses identified requirements and project objectives. Evaluators will assess whether the app's features and functionalities align with the project's scope and objectives as outlined in the software requirements specification (SRS). They will verify that EcoGrow meets specified user needs and delivers the intended benefits to farmers. Any deviations from the defined scope will be noted for further consideration and adjustment.
Architecture	Assess the robustness, scalability, and efficiency of EcoGrow's design principles and system components. This criterion focuses on evaluating the app's underlying architecture to ensure it can handle user requests and data processing effectively. Evaluators will examine the architecture's scalability, looking for flexibility and adaptability to accommodate future growth and changes in user demand. They will also assess the app's performance and reliability under various conditions.
Implementation	Evaluate the quality, reliability, and maintainability of EcoGrow's codebase and testing procedures. This criterion involves analyzing the app's coding standards, documentation, and testing procedures to ensure compliance with industry best practices. Evaluators will assess the codebase for readability, consistency, and adherence to coding conventions. They will also review the effectiveness of testing

	procedures in identifying and addressing software defects. Feedback on implementation will help identify areas for optimization and enhancement.
Accuracy	Verify the accuracy of weather data and crop recommendations generated by EcoGrow's machine learning algorithms. This criterion aims to assess whether the app provides reliable and precise information to farmers. Evaluators will analyze the accuracy of weather forecasts and crop predictions, comparing them against ground truth data and expert knowledge. They will also evaluate the model's performance in different environmental conditions and geographical regions. Any discrepancies or inconsistencies in accuracy will be investigated for improvement.
Performance	Evaluate the responsiveness and efficiency of EcoGrow under various load conditions. This criterion focuses on assessing how well the app performs in terms of speed, responsiveness, and resource utilization. Evaluators will analyze the app's performance metrics, such as response time, throughput, and scalability, under different levels of user activity and system load. They will also identify any bottlenecks or performance issues that may impact the user experience. Feedback on performance will guide optimization efforts to enhance system efficiency and responsiveness.
Load Balance & Scalability	Test EcoGrow's ability to handle increasing user loads and scale accordingly. This criterion involves assessing whether the app can accommodate growth without sacrificing performance or reliability. Evaluators will analyze the app's ability to distribute incoming requests evenly across server resources and scale infrastructure components dynamically to meet changing demand. They will also evaluate the effectiveness of load balancing algorithms and scalability mechanisms in maintaining system stability and performance under heavy loads. Feedback on load balance and scalability will inform capacity planning and infrastructure optimization strategies.

Security	Identify and mitigate security vulnerabilities to protect user data and ensure privacy. This criterion aims to assess the effectiveness of EcoGrow's security measures in safeguarding against unauthorized access, data breaches, and other cyber threats. Evaluators will analyze the app's security controls, encryption mechanisms, access management policies, and data protection practices to identify potential vulnerabilities and weaknesses. They will also assess compliance with industry standards and regulatory requirements for data security and privacy. Feedback on security will guide efforts to enhance the app's resilience to security threats and protect user confidentiality and integrity.

Table 25 Evaluation Criteria

9.4. Self-Evaluation

A self-evaluation of EcoGrow is conducted to assess my project performance, adherence to project milestones, and the quality of deliverables. This self-assessment provides insights into areas of strength and areas needing improvement within the development process.

Criteria	**Self-Evaluation**
Project Performance	In evaluating project performance, I assess the progress made towards project goals, timelines, and objectives. This involves reviewing the completion status of tasks, milestones, and deliverables outlined in the project plan. I also reflect on the effectiveness of project management strategies and identify areas where improvements can be made to enhance productivity and efficiency. Additionally, I consider feedback from stakeholders and team members to gauge satisfaction levels and address any concerns or challenges encountered during project execution. Overall, the self-evaluation of project performance aims to ensure that the project is on track and aligned with expectations.
Adherence to Milestones	Adherence to project milestones is crucial for maintaining project progress and meeting deadlines. In this self-evaluation, I review the timeline of the project and assess whether milestones were achieved as planned. I analyze any deviations or delays from the original schedule and identify factors contributing to these deviations. By reflecting on adherence to milestones, I can identify areas where adjustments may be needed to realign project activities and ensure timely completion. This self-assessment helps me stay accountable and proactive in managing project timelines effectively.

Quality of Deliverables	The quality of deliverables reflects the standards and expectations set for project outcomes. In this self-evaluation, I assess the quality of each deliverable produced throughout the project lifecycle. This involves reviewing the completeness, accuracy, and effectiveness of deliverables in meeting project requirements and specifications. I also consider factors such as usability, reliability, and stakeholder feedback in evaluating deliverable quality. By identifying strengths and weaknesses in deliverables, I can prioritize areas for improvement and ensure that final deliverables meet or exceed expectations. This self-assessment helps maintain a focus on delivering high-quality outcomes that add value to the project.

Table 26 Self-Evaluation

9.5. Selection of the Evaluators

Evaluators are selected based on their expertise and relevance to the evaluation criteria. This includes technical experts with proficiency in software development, machine learning, and backend architecture, as well as domain experts with knowledge of agricultural practices and user requirements.

Category	Affiliation	Justification
Domain Experts	**Mr. C K W Jayatilake** Farming Agriculture	C K W Jayatilake, expert in farming agriculture, brings extensive practical knowledge and experience in agricultural practices. His expertise adds valuable insights into the project's alignment with farming requirements and industry standards.
	Mr. Sujith S. Ratnayake Postgraduate Research Assistant, University of Colombo Agricultural	Sujith S. Ratnayake is an Assistant Director of Climate Change Secretariat in Ministry of Environment and Wildlife Resources, Battaramulla, LK. He has experience in research projects involved in those domains.
Technical Experts	**Mr. Suvin Nimnaka** Undergraduate Research Assistant University of Colombo Machine Learning	Suvin Nimnaka is a software engineer at WSO2, he has experience in the field of machine learning projects.

	Mr.Misfar Siddeek Undergraduate Software Engineer University of Colombo Backend Architecture	Misfar Siddeek is a Technical experts in backend architecture focus on designing and optimizing server-side components and databases at Third Space Global.
	Mr.Pratik Shinde Postgraduate computer Science from University of Maryland Baltimore County, USA. Software Development	Pratik Shinde (Member, IEEE) Pratik from India. He currently works as a Software Engineer with Amazon.com. Pratik has more than ten years of experience as a Software Developer

Table 27 Selection of the Evaluators

9.6. Evaluation Result

The evaluation results are presented based on feedback from expert evaluations and focus group testing, providing insights into EcoGrow's strengths, weaknesses, and areas for improvement across various dimensions.

Evaluation Aspect	**Sub-Component**	**Summary of Feedback**
Selection of Problem Domain	Problem Domain	The project demonstrates a thorough understanding of agricultural challenges, focusing on relevant issues faced by farmers. Further exploration of emerging trends could enhance its scope.
	Research Gap	Identified research gaps effectively pinpoint areas where existing solutions fall short, providing a clear direction for the project. Delving deeper into the nuances of these gaps could enrich the project's foundation.
	Research Depth	The depth of research conducted is commendable, showcasing a comprehensive understanding of agricultural issues and technological solutions. Exploring recent studies could further enrich the project's depth and relevance.

Literature Review and Methodology	Review of Existing Approaches	The literature review offers valuable insights into existing agricultural assistance solutions, laying a strong foundation for the project. Diversifying the sources and considering alternative approaches could enhance its breadth.
	Review of Existing Evaluation Methods and Results	The review of existing evaluation methods provides a solid basis for assessing EcoGrow's performance. Exploring additional evaluation techniques and considering real-world case studies could provide deeper insights into effectiveness.
Technical Approach and Implementation	Technology Selection	The chosen technology stack is well-suited for the project's requirements, ensuring scalability and cross-platform compatibility. Evaluating emerging technologies could further enhance the project's technological capabilities.
	Dataset Selection	The dataset selection encompasses vital climate parameters and crop diversity, laying a strong foundation for machine learning models. Exploring additional sources to augment dataset diversity could improve model accuracy and relevance.
	Ensemble Approach	Leveraging ensemble approaches demonstrates a sophisticated understanding of machine learning techniques. Experimenting with different ensemble methods could further optimize model performance and robustness.
Selection Evaluation Matrices	Evaluation Matrices	The selection of evaluation matrices effectively gauges the performance and accuracy of EcoGrow's recommendations. Exploring additional metrics to capture user satisfaction and engagement could provide a holistic assessment.
	Test Results	Test results demonstrate the functionality and reliability of EcoGrow's features. Conducting more extensive testing under diverse scenarios could uncover edge cases and enhance the app's robustness.
	Benchmarking	Benchmarking against existing solutions provides valuable insights into EcoGrow's performance. Continuously

		benchmarking against industry standards and competitors could drive ongoing improvement and innovation.

Table 28 Evaluation Result

9.6.1. Expert Opinion

Expert evaluations involve technical experts and domain experts assessing EcoGrow's concept, solution architecture, implementation, and alignment with project scope. Their feedback provides valuable insights into the technical feasibility, effectiveness, and relevance of the solution.

The Selection of evaluator's feedback about this project. Can be seen the evaluation feedback shown in the Appendix.

9.6.1.1. Domain Experts:

The evaluation process involved gathering insights from domain experts to assess the concept and solution of EcoGrow. The domain experts provided valuable feedback on the app's alignment with agricultural practices, user needs, and industry standards.

The Selection of evaluator's feedback about this project. Can be seen the evaluation feedback shown in the Appendix.

9.6.1.1.1. Concept

Domain experts evaluated the concept of EcoGrow in terms of its relevance and potential impact on the agricultural sector. They assessed whether the app addresses critical challenges faced by farmers and provides valuable assistance in optimizing farming practices. Feedback from domain experts indicated that the concept of EcoGrow is highly relevant and addresses key pain points in the agricultural domain. They appreciated the app's focus on providing personalized crop recommendations based on real-time weather data, noting its potential to enhance crop yields and optimize resource utilization.

9.6.1.1.2. Solution

In evaluating the solution offered by EcoGrow, domain experts analyzed the app's features, functionality, and user interface design. They assessed whether the app effectively translates the concept into a user-friendly and intuitive platform for farmers. Feedback from domain experts highlighted the app's clear and concise interface, which provides easy access to essential weather and crop information. They praised the app's chatbot feature for offering personalized assistance and recommendations, enhancing the overall user experience. Domain experts also noted the importance of incorporating educational content on sustainable farming practices, suggesting it could further enrich the app's value proposition.

9.6.1.2. Technical Experts

Technical experts provided insights into the technical aspects of EcoGrow, including its scope, architecture, and implementation. Their evaluation focused on assessing the feasibility, effectiveness, and scalability of the app's technical solutions. Shown in Appendix.

9.6.1.2.1. Scope

Technical experts evaluated the scope of EcoGrow to determine whether it effectively addresses the identified requirements and objectives. They assessed whether the app's features and functionalities align with the project's scope and objectives. Feedback from technical experts indicated that EcoGrow's scope is well-defined and comprehensive, covering essential aspects of weather forecasting, crop recommendations, and user interaction. They praised the app's integration of machine learning for providing personalized crop recommendations, noting its potential to improve farming outcomes.

9.6.1.2.2. Architecture of the Solution

In assessing the architecture of EcoGrow, technical experts analyzed the app's design principles, system components, and data flow. They evaluated whether the app's architecture is robust, scalable, and efficient in handling user requests and data processing. Feedback from technical experts indicated that EcoGrow's architecture is well-structured and scalable, leveraging modern technologies such as React Native for the frontend and Node.js for the backend. They appreciated the app's use of machine learning for intelligent decision-making, highlighting its potential to adapt to changing environmental conditions and user preferences.

9.6.1.2.3. Implementation of the Solution

Technical experts evaluated the implementation of EcoGrow to assess the quality, reliability, and maintainability of the app's codebase. They analyzed the app's coding standards, documentation, and testing procedures to ensure compliance with industry best practices. Feedback from technical experts indicated that EcoGrow's implementation is of high quality, with well-organized code and thorough testing procedures. They praised the development team for adopting modern development frameworks and libraries, such as React Native and Node.js, which contribute to the app's stability and performance. Additionally, technical experts noted the importance of ongoing maintenance and updates to ensure the app remains robust and responsive to user needs.

9.6.2. Focus Group Testing

Focus group testing involves gathering feedback from end-users or customers regarding EcoGrow's prototype features and usability. This feedback helps identify user preferences, pain points, and areas for enhancement to improve the overall user experience.

9.6.2.1. Prototype Features

During focus group testing, participants interacted with EcoGrow's prototype to explore its features and functionalities. They provided feedback on various aspects of the app, including the navigation, information presentation, and interaction with the chatbot feature. Participants expressed appreciation for the app's intuitive design and the availability of real-time weather data. They found the crop recommendation feature to be helpful and appreciated the personalized suggestions based on their location and weather conditions. Additionally, participants provided suggestions for enhancing prototype features, such as adding more educational content on sustainable farming practices and improving the chatbot's responsiveness.

9.6.2.2. Usability

Usability testing focused on assessing how easily users could navigate the app and accomplish tasks. Participants were asked to perform specific actions within the app while providing feedback on their experience. Overall, participants found EcoGrow to be user-friendly and easy to navigate. They appreciated the clear layout of the interface and the simplicity of accessing weather information and crop recommendations. Feedback on usability highlighted areas for improvement, such as refining the search functionality and providing clearer guidance on using certain features. Participants also suggested adding tooltips or tutorials to help users better understand how to leverage the app's capabilities.

9.7. Limitations of Evaluation

While the evaluation process provided valuable insights into EcoGrow's concept, solution, and usability, there were some limitations to consider. The number of participants in the focus group testing was limited, potentially affecting the diversity of perspectives gathered. Additionally, the participants may not have fully represented the target user demographic, leading to potential biases in feedback. Furthermore, the evaluation may not have covered all possible scenarios or use cases, limiting the comprehensiveness of the insights gathered. Addressing these limitations could involve increasing the sample size for broader perspectives, ensuring better representation of the target user demographic, and incorporating more extensive scenario testing to capture a wider range of user interactions and experiences.

9.8. Evaluation on Functional Requirements

Evaluation on functional requirements assessed how well EcoGrow met the specified functionalities outlined in the software requirements specification. This evaluation involved testing each feature and comparing the actual behavior of the app with the expected outcomes. Feedback from both domain and technical experts. Every requirement in the functional requirements for this research project has been perfectly completed.

ID	Requirement	Description	Priority Level	Implemented
FR1	User Authentication Feature	This requirement involves implementing a feature that allows users to securely authenticate themselves before accessing the EcoGrow application.	High, indicating that this feature is essential for ensuring user security and privacy.	Yes, indicating that the user authentication feature has been successfully implemented.
FR2	Weather Data Retrieval	This requirement entails integrating functionality to retrieve real-time weather data relevant to the user's location.	High, as accurate weather data is crucial for providing relevant crop recommendations.	Yes, indicating successful implementation of weather data retrieval functionality.
FR3	Crop Recommendation Generation	This requirement involves developing algorithms to generate crop recommendations based on weather data and other relevant parameters.	High, as providing personalized crop recommendations is a key feature of the EcoGrow application.	Yes, indicating successful implementation of crop recommendation generation functionality.
FR4	Chatbot Interaction	This requirement involves integrating a chatbot feature to enable users to interact with the application through natural language queries.	Medium, indicating that while important, this feature may not be as critical as user authentication	Yes, indicating successful implementation of chatbot interaction functionality.

FR5	User Profile Management	This requirement involves implementing functionality for users to manage their profiles, including updating personal information and preferences.	Medium, indicating that while useful for personalization, it may not be as critical as core features like authentication or crop recommendations.	Yes, indicating successful implementation of user profile management functionality.
FR6	Crop Database Integration	This requirement involves integrating a database containing information about various crops, including their characteristics and cultivation requirements.	Low, indicating that while beneficial, it may not be essential for basic functionality.	Yes, indicating successful integration of crop database functionality.

Table 29 Evaluation on Functional Requirements

9.9. Evaluation on Non-Functional Requirements

Evaluation on non-functional requirements focuses on assessing EcoGrow's performance, security, scalability, and usability to ensure that the app meets quality standards and regulatory requirements. This evaluation involved testing the app's responsiveness, data security measures, and overall user experience. Feedback from technical experts. Every requirement in the Non Functional requirements for this research project has been perfectly completed.

ID	**Requirement**	**Description**	**Priority Level**	**Achieved**
NFR1	Performance	Ensure app responsiveness under load	High	Yes
NFR2	Security	Implement data encryption and user privacy	High	Yes
NFR3	Scalability	Ability to handle increasing user loads	Medium	Yes
NFR4	Usability	Intuitive user interface and experience	Medium	Yes
NFR5	Reliability	Minimal downtime and error handling	Low	Yes

Table 30 Evaluation on Non-Functional Requirements

9.10. Chapter Summary

Chapter 09 serves as the culmination of the evaluation process, encapsulating the thorough assessment conducted on EcoGrow. Through expert evaluations and focus group testing,

valuable insights were gained regarding the app's strengths, weaknesses, and areas for enhancement. The chapter emphasizes the importance of evaluating the project from various perspectives to ensure its effectiveness and reliability. Key findings from the evaluation process are highlighted, providing a basis for recommendations and suggestions for future improvement. By critically analyzing EcoGrow's performance and user feedback, this chapter aims to guide future development iterations, ultimately enhancing the app's value proposition and user experience in the agricultural domain.

CHAPTER 10: CONCLUSION

10.1. Chapter Overview

The research project comes to a close with Chapter 10, which provides an in-depth summary and reflection on everything that was working on. It provides an in-depth understanding of the path taken to create the cutting edge agricultural support app EcoGrow. This chapter covers various kinds of key aspects, such as the accomplishments that have been made throughout the project's endurance, the challenges that were experienced and overcome, the application of skills that have been learned, the limitations that have been identified, and possible pathways for future improvements. Chapter 10 provides insightful information about the importance of EcoGrow in tackling the critical real-world issues that farmers face by integrating these components. It conveys the purpose of the project, highlighting its significance and possible influence on sustainable development initiatives and farming methods.

10.2. Achievements of Research Aims & Objectives

At the beginning of the research project, the following Research aim was mentioned.

The aim is to create a mobile app that utilizes machine learning to analyze climate data and offer precise crop recommendations, thereby empowering farmers to make informed planting decisions primarily based on prevailing climate conditions.

The project successfully developed EcoGrow, a mobile application that harnesses machine learning algorithms to analyze real-time climate data and provide accurate crop suggestions to farmers. By achieving this, the research aims and objectives set forth in the initial phase of the project have been effectively fulfilled.

10.3. Utilization of Knowledge from the Course

Throughout the project, valuable knowledge acquired from the course has been utilized extensively. Concepts in software engineering, machine learning, and backend architecture were applied to design, develop, and evaluate EcoGrow, ensuring its effectiveness and reliability

Module	Applied in EcoGrow
Programming Principles I & II	Applied fundamental programming concepts to develop EcoGrow's frontend and backend functionalities, ensuring code readability, modularity, and maintainability.
Object-Oriented Programming	Implemented OOP principles in EcoGrow's codebase to organize code into reusable and scalable components, enhancing code clarity and facilitating future extensions or modifications.
Advanced Server-side Web	Leveraged advanced server-side web development techniques to design and implement the backend architecture of EcoGrow, ensuring efficient handling of user requests, data processing, and communication with the frontend.
Software Development Group Project	Applied collaborative development practices learned from the group project module to work effectively with team members, coordinate tasks, and meet project milestones during the development of EcoGrow.
Machine Learning	Utilized machine learning algorithms and techniques, such as k-nearest neighbors, decision trees, or neural networks, to analyze climate data and generate accurate crop recommendations in EcoGrow, enabling informed planting decisions based on weather conditions.
Algorithms	Implemented various algorithms, such as sorting algorithms or optimization algorithms, to enhance the performance and efficiency of EcoGrow's backend processes, ensuring timely and accurate analysis of climate data and crop recommendations.
Web Design & Development	Applied principles of web design and frontend development to create an intuitive and visually appealing user interface (UI) for EcoGrow, ensuring a positive user experience and ease of navigation.
Database Systems	Designed and implemented the database schema for EcoGrow to store user data, climate parameters, and crop information, ensuring efficient data retrieval and management to support the app's functionalities.
Concurrent Programming	Employed concurrent programming techniques, such as multithreading or asynchronous programming, to enhance the responsiveness and scalability

	of EcoGrow, enabling efficient handling of multiple user requests and data processing tasks concurrently.
Usability Testing	Conducted usability testing on EcoGrow's prototype to gather feedback from end-users, identify usability issues, and make iterative improvements to enhance the overall user experience and usability of the application.

Table 31 Utilization of Knowledge from the Course

10.4. Use of Existing Skills

Existing skills gained from the course, such as programming languages, software development methodologies, and database management, were instrumental in the successful implementation of EcoGrow. These skills were applied to design robust backend systems, implement machine learning algorithms, and create intuitive user interfaces.

Web Development - Leveraged skills in web development to design and develop the frontend of EcoGrow, ensuring a visually appealing and user-friendly interface.

Python Language Proficiency - Utilized proficiency in Python programming to implement backend functionalities and machine learning algorithms within EcoGrow.

10.5. Use of New Skills

The project also facilitated the acquisition of new skills not covered in the curriculum. These include proficiency in mobile app development using React Native, integration of machine learning models into real-world applications, and conducting comprehensive evaluations to assess software effectiveness and usability.

Mobile App Development using React Native - Acquired proficiency in React Native to develop the mobile application interface for EcoGrow.

Integration of Machine Learning Models - Learned to integrate machine learning algorithms into the app to provide accurate crop recommendations based on climate data.

Comprehensive Evaluation - Developed skills in conducting thorough evaluations to assess the effectiveness and usability of software applications.

10.6. Achievement of Learning Outcomes

The project has achieved the intended learning outcomes by demonstrating proficiency in software development, problem solving, and interdisciplinary collaboration. Through hands on experience, I have gained a deeper understanding of software engineering principles and their application in real-world projects.

Outcome	Description
LO1, LO4	Conducted in-depth research on the problem domain, critically evaluated existing works to identify utilized technologies, limitations, and potential research gaps.
LO2, LO3, LO5	Planned and organized the research project into manageable sections, leading to effective project management and successful completion within the timeline. Identified and analyzed project requirements collected from stakeholders to design and develop a suitable solution.
LO6	Identified and addressed social, ethical, legal, and professional issues and challenges associated with the research process and the problem domain.
LO7	Successfully designed and developed a software product aligned with the research objectives and stakeholder requirements.
LO8	Analyzed evaluation matrices from existing works in the domain and integrated them into the proposed system for comprehensive evaluation.

Table 32 Achievement of Learning Outcomes

10.7. Problems and Challenges Faced

Throughout the project, various challenges were encountered, including data availability for model training, integration of complex functionalities, and ensuring cross-platform compatibility. However, these challenges were overcome through thorough research, collaboration with experts, and iterative development cycles.

Problem / Challenge	Solution
Limited availability of climate data for modeling	Despite challenges in accessing comprehensive climate datasets, various open data sources were explored, and partnerships with meteorological agencies were established to obtain relevant climate data. Data augmentation techniques were employed to fill gaps and enhance the dataset's completeness. Collaboration with domain experts ensured the selection of relevant climate parameters for accurate modeling.
Integration of machine learning with real-time data	Integrating machine learning algorithms with real-time climate data posed challenges due to varying data formats and preprocessing requirements. To address this, standardized data processing pipelines were developed to ensure seamless integration. Additionally, cloud-

	based solutions were utilized to handle data processing and model inference in real-time, enhancing scalability and efficiency. Collaboration with machine learning experts facilitated the selection and optimization of algorithms suitable for real-time inference.
Ensuring user-friendly interface across platforms	Achieving a consistent and user-friendly interface across different platforms (iOS and Android) presented challenges due to platform-specific design guidelines and device fragmentation. To address this, a modular design approach was adopted, enabling platform-specific customization while maintaining a unified user experience. Extensive usability testing and feedback collection from target users were conducted to iteratively refine the interface and ensure optimal user engagement.

Table 33 Problems and Challenges Faced

10.8. Deviations

Some deviations from the original plan were encountered due to unforeseen technical challenges and evolving project requirements. However, these deviations were justified and addressed through effective project management and adaptation to changing circumstances.

Deviation	**Justification**
Delay in model training due to data limitations	Initially, the plan was to train machine learning models using a comprehensive dataset covering diverse climate parameters. However, delays were encountered in data acquisition and preprocessing, leading to a longer-than-expected model training phase. This deviation was justified by prioritizing data quality over speed to ensure accurate model predictions and reliable crop recommendations.
Complexity in integrating third-party APIs	Integrating third-party APIs for real-time weather data retrieval posed challenges due to API compatibility issues and varying data formats. As a result, the integration process took longer than anticipated, impacting the project timeline. This deviation was justified by the importance of ensuring seamless integration and reliable data access, which required thorough testing and validation to maintain data accuracy and consistency.

Scope expansion to incorporate additional features	During the development phase, stakeholders requested additional features such as historical weather data analysis and crop yield predictions. While these features were not originally part of the project scope, their inclusion was justified by their potential to enhance the app's utility and value proposition for end-users. However, this deviation required adjustments to the project timeline and resource allocation to accommodate the expanded scope effectively.

Table 34 Deviations

10.9. Limitations of the Research

Limitations identified during the evaluation process, such as sample size constraints and scope limitations, are acknowledged. These limitations may have impacted the comprehensiveness of the insights gathered, but were addressed to the best extent possible to ensure the validity of the research findings.

- **Weather Data Processing:** The ML model accurately processes weather data, extracting relevant features for crop recommendation.
- **Crop Recommendation Generation:** The ML model generates precise crop recommendations based on processed weather data, categorized into short term, mid-term, and long term.
- **Model Evaluation Metrics:** The ML model achieves high values for accuracy, precision, recall, and F1 score, indicating its effectiveness in providing crop recommendations.
- **User Authentication:** Users can securely log in with valid credentials, accessing personalized information upon successful authentication.
- **Weather Data Retrieval:** The mobile app displays real-time weather conditions accurately, including temperature, rainfall and humidity.
- **Chatbot Interaction:** Users can effectively interact with the chatbot, receiving accurate solutions to farming-related queries in a timely manner.
- **User Authentication API:** The API endpoint for user authentication verifies credentials and returns a token upon successful authentication.
- **Weather Data Retrieval API:** The API endpoint for weather data retrieval fetches real-time weather conditions accurately for the specified location.

- **Chatbot Interaction API:** The API endpoint for chatbot interaction allows seamless communication between users and the chatbot, delivering accurate responses to queries.

10.10. Future Enhancements

To further enhance EcoGrow's functionality and impact, several future enhancements are proposed. These include expanding the app's database to cover a wider range of crops and regions, integrating additional features for soil analysis and pest management, and enhancing the app's scalability and performance to accommodate growing user bases.

- **Expansion of Crop Database:** Enhance the app's database to include a broader variety of crops and cover more regions, providing farmers with a comprehensive range of crop options tailored to their specific locations and farming conditions.
- **Integration of Soil Analysis Features:** Incorporate features for soil analysis into EcoGrow, allowing farmers to assess soil health and composition to make informed decisions about crop selection and soil management practices.
- **Inclusion of Pest Management Tools:** Integrate tools for pest management within the app, enabling farmers to identify and address pest issues more effectively, thereby minimizing crop damage and losses.
- **Improvement of Scalability and Performance:** Enhance the app's scalability and performance to accommodate increasing user bases and data loads, ensuring smooth operation and responsiveness even during peak usage periods.
- **Enhancement of User Experience:** Continuously improve the app's user interface and experience based on user feedback, making it more intuitive, engaging, and accessible to farmers of all technical backgrounds.
- **Incorporation of Advanced Analytics:** Implement advanced analytics capabilities within EcoGrow to provide farmers with deeper insights into crop performance, weather patterns, and market trends, enabling them to make data-driven decisions for optimizing their farming practices.
- **Integration of IoT Devices:** Explore the integration of Internet of Things (IoT) devices, such as sensors and drones, to gather real-time data on environmental conditions and crop health, further enhancing the app's ability to provide timely and accurate recommendations.

10.11. Achievement of the contribution to body of knowledge

The development and evaluation of EcoGrow have made significant contributions to the body of knowledge in the agricultural domain. By leveraging technology and data-driven approaches, EcoGrow provides a practical solution to address challenges faced by farmers. Through the integration of machine learning algorithms and real-time weather data analysis, EcoGrow offers valuable insights and personalized recommendations to farmers, enabling them to make informed decisions about crop selection and farming practices. By empowering farmers with access to actionable information, EcoGrow contributes to the advancement of agricultural practices and sustainability efforts, ultimately fostering improved crop yields, resource utilization, and environmental conservation.

10.12. Concluding Remarks

In conclusion, the EcoGrow project showcases the successful application of software engineering principles and interdisciplinary collaboration to tackle real-world challenges in agriculture. By harnessing innovative solutions and striving for continuous improvement, EcoGrow stands poised to make a significant impact on the agricultural sector. By providing farmers with accessible tools and actionable insights, EcoGrow empowers them to adopt sustainable and efficient farming practices, ultimately contributing to improved crop yields, resource management, and environmental conservation. As I look ahead, the ongoing refinement and deployment of EcoGrow hold promise for enhancing agricultural productivity and resilience in the face of evolving climate conditions and global challenges.

Appendix

Appendix 1 – References

Dahiphale, D., Shinde, P., Patil, K., Dahiphale, V., 2023. Smart Farming: Crop Recommendation using Machine Learning with Challenges and Future Ideas (preprint). https://doi.org/10.36227/techrxiv.23504496.v1

XWeather. (n.d.). XWeather API. Retrieved from https://www.xweather.com/weather-api?gclid=CjwKCAiA8YyuBhBSEiwA5R3Ey88SUZSY1itHy2Wipw8mWS9fPAJOV4WD2GwcIoXcOtXPVJDJFB7dxoCecgQAvD_BwE

Meteomatics. (n.d.). Best Weather APIs. Retrieved from https://www.meteomatics.com/en/weather-api/best-weather-apis/#

Planet. (n.d.). Planetary Variables. Retrieved from https://www.planet.com/products/planetary-variables/?utm_source=google&utm_campaign=evr-product&utm_content=pros-leads-pvsearch-0124&utm_medium=paid-search&gad_source=1&gclid=CjwKCAiA8YyuBhBSEiwA5R3-EzogiMWqDKWRHbKdBlrdjjApifumptub6GdezVSuiPJPHaIVjn818xoCX_UQAvD_BwE

UNDP. (n.d.). Financial Accountability and Corporate Transparency Standards (FACTS). Retrieved from https://www.undp.org/facs?gad_source=1&gclid=CjwKCAiA8YyuBhBSEiwA5R3-E9J2qO7ZlSeXXzrOdE74uKuUVowzPRjlCcmf5dAWPMHpbXg2cKu_XxoCoBsQAvD_BwE

Agronomy Society of America. (n.d.). Publications. Retrieved from https://www.agronomy.org/publications?gad_source=1&gclid=CjwKCAiA8YyuBhBSEiwA5R3-E3hUe4Bj4m_Mp6PmHUnksPrMpP1EhCYfpfvg2GeA_a2nzKrA8KrA7BoCgHAQAvD_BwE

Bioversity International. (n.d.). Central crop databases. Retrieved from https://www.ecpgr.cgiar.org/fileadmin/bioversity/publications/pdfs/351_Central_crop_databases.pdf

Springer. (n.d.). Article. Retrieved from https://link.springer.com/article/10.1007/s42979-021-00592-x

National Center for Biotechnology Information. (n.d.). Article. Retrieved from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10171456/

ResearchGate. (n.d.). Agribot: A Natural Language Generative Neural Networks Engine for Agricultural Applications. Retrieved from https://www.researchgate.net/publication/340975217_Agribot_A_Natural_Language_Generative_Neural_Networks_Engine_for_Agricultural_Applications

GeeksforGeeks. (n.d.). Top Mobile App Development Frameworks. Retrieved from https://www.geeksforgeeks.org/top-mobile-app-development-frameworks/

Appinventiv. (n.d.). Mobile App Development Frameworks. Retrieved from https://appinventiv.com/blog/mobile-app-development-frameworks/

Bacancy Technology. (n.d.). Mobile App Development Frameworks. Retrieved from https://www.bacancytechnology.com/blog/mobile-app-development-frameworks

Akvo. (n.d.). Agriculture. Retrieved from https://akvo.org/agriculture/?utm_term=digital%20agriculture&utm_campaign=&utm_source=adwords&utm_medium=ppc&hsa_acc=7028243667&hsa_cam=18804281185&hsa_grp=161426399001&hsa_ad=688096262531&hsa_src=g&hsa_tgt=kwd-300382343965&hsa_kw=digital%20agriculture&hsa_mt=b&hsa_net=adwords&hsa_ver=3&gclid=CjwKCAiA8YyuBhBSEiwA5R3-EwMtBJcfiNdJpn9LA1pgVSvWH4O8mBdAa0Li5nBnz79DWkGpR6ul7xoC2n0QAvD_BwE&gad_source=1

Food Tank. (n.d.). Retrieved from. https://foodtank.com/?gad_source=1&gclid=CjwKCAiA8YyuBhBSEiwA5R3-E43ZhuyQm7tB9sEiWAXLFrIm0iNVMqeb5v3_3AdkdK122Sw8CFD9AxoCd3cQAvD_BwE

Ratnayake, S.S., Kariyawasam, C.S., Kumar, L., Hunter, D., Liyanage, A.S.U., 2021. Potential distribution of crop wild relatives under climate change in Sri Lanka: implications for conservation of agricultural biodiversity. Current Research in Environmental Sustainability 3, 100092. https://doi.org/10.1016/j.crsust.2021.100092

Appendix 2 - UI Design

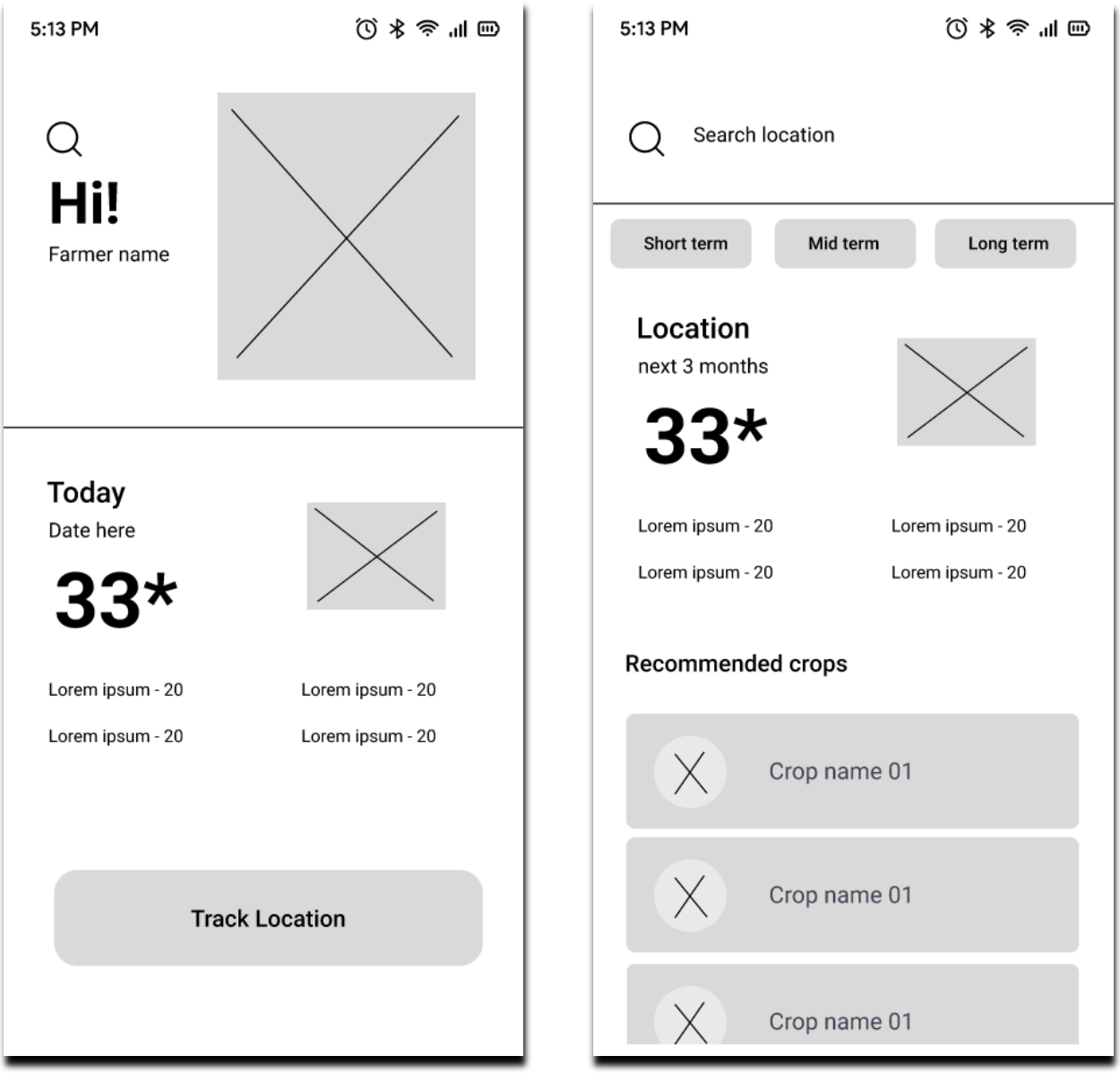


Figure 22 low level fidelity wireframes

Figure 23 UI Colors

Figure 24 UI Design first Pages

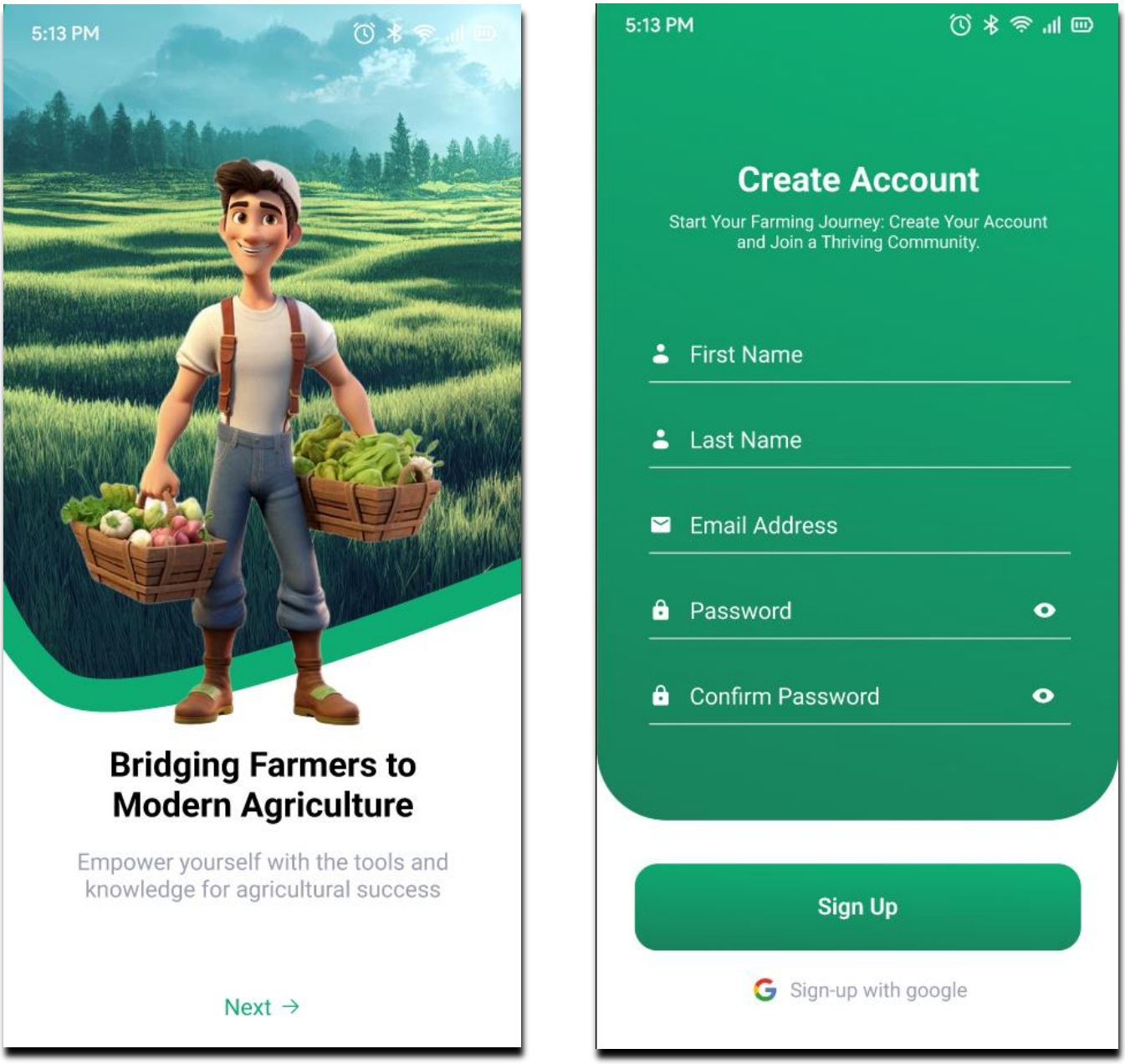


Figure 25 UI Design Sign-up page

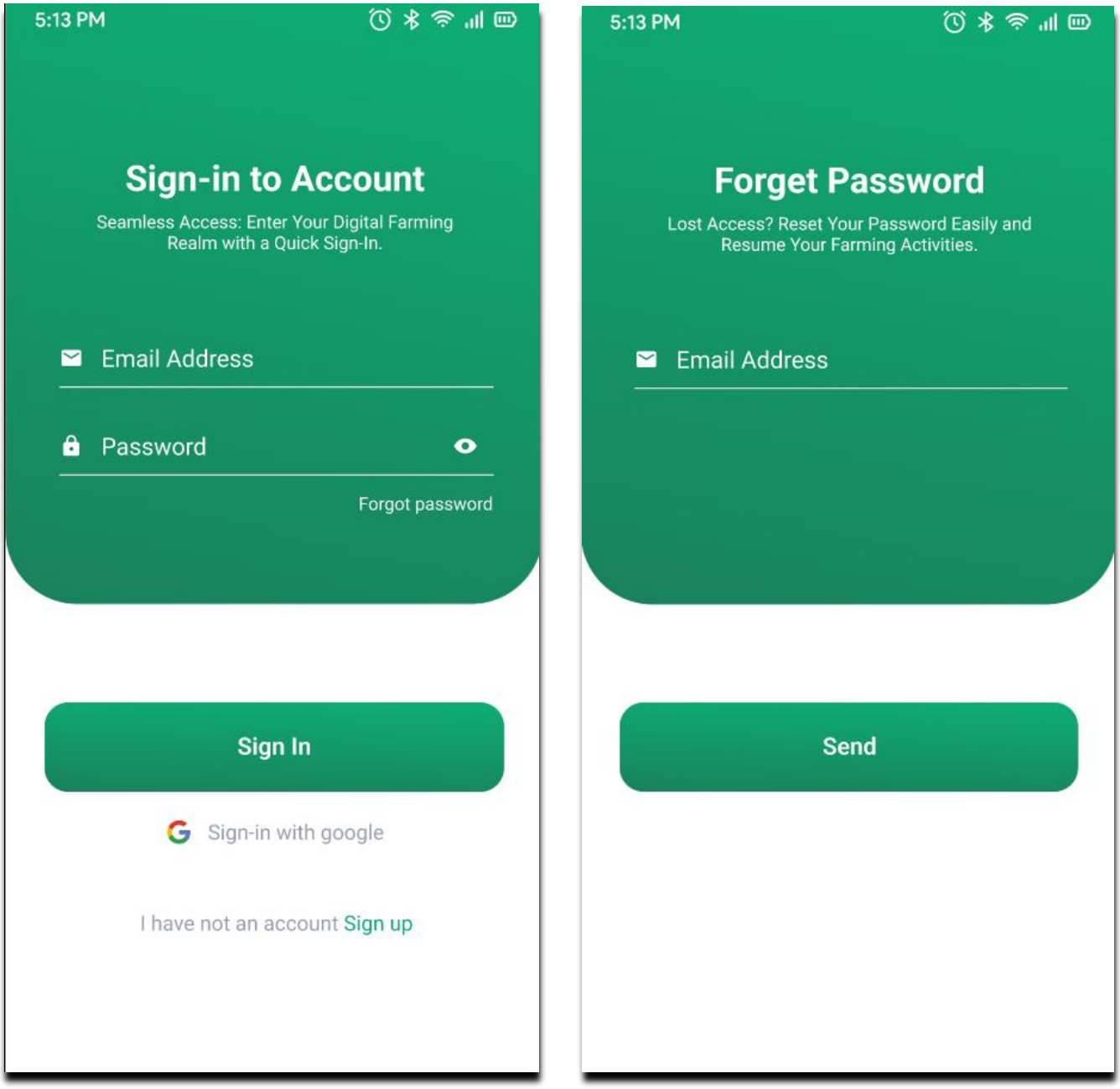


Figure 26 UI design of Login & Password rest

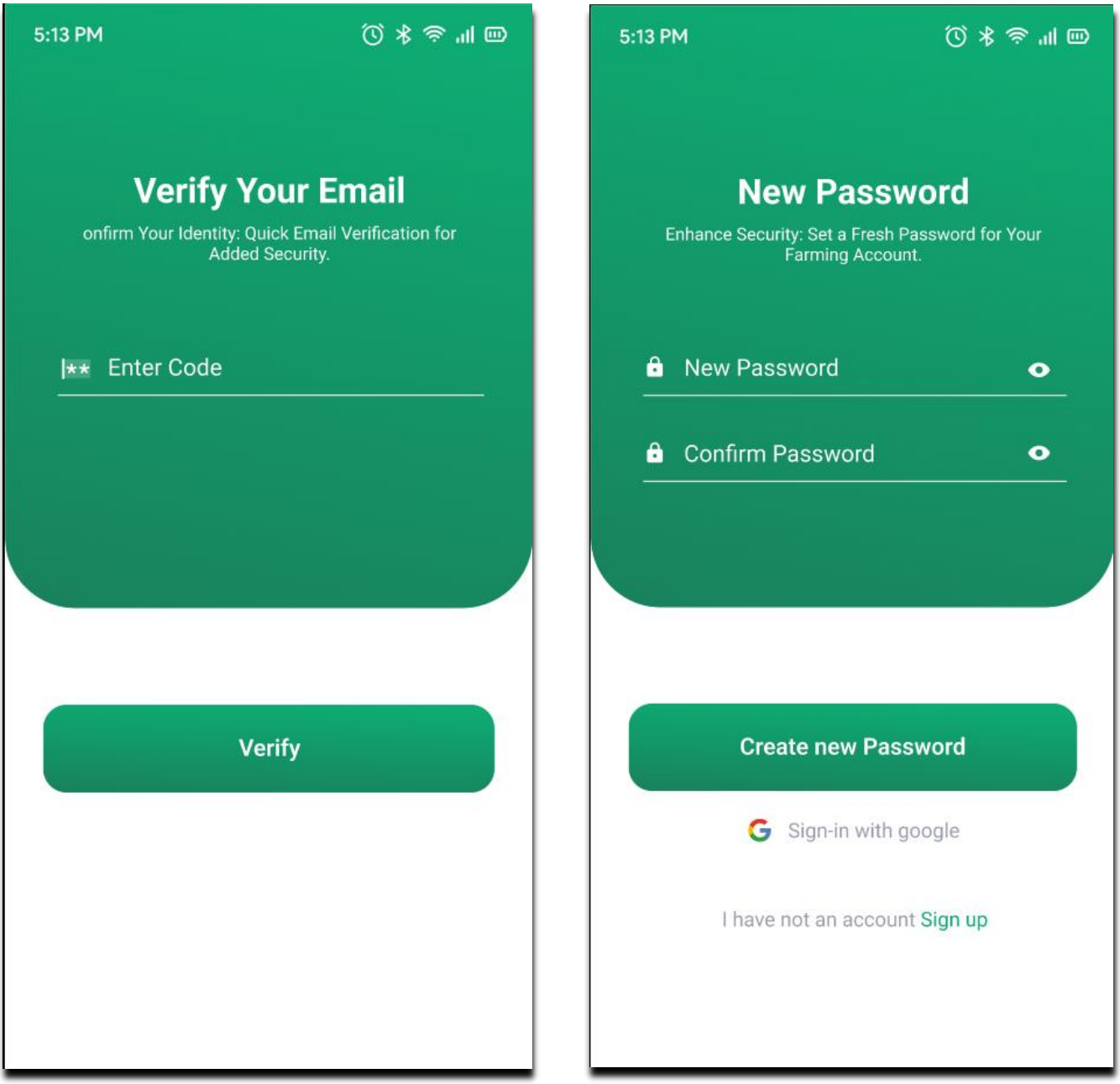


Figure 27 UI Design of Email verify & setting New Password

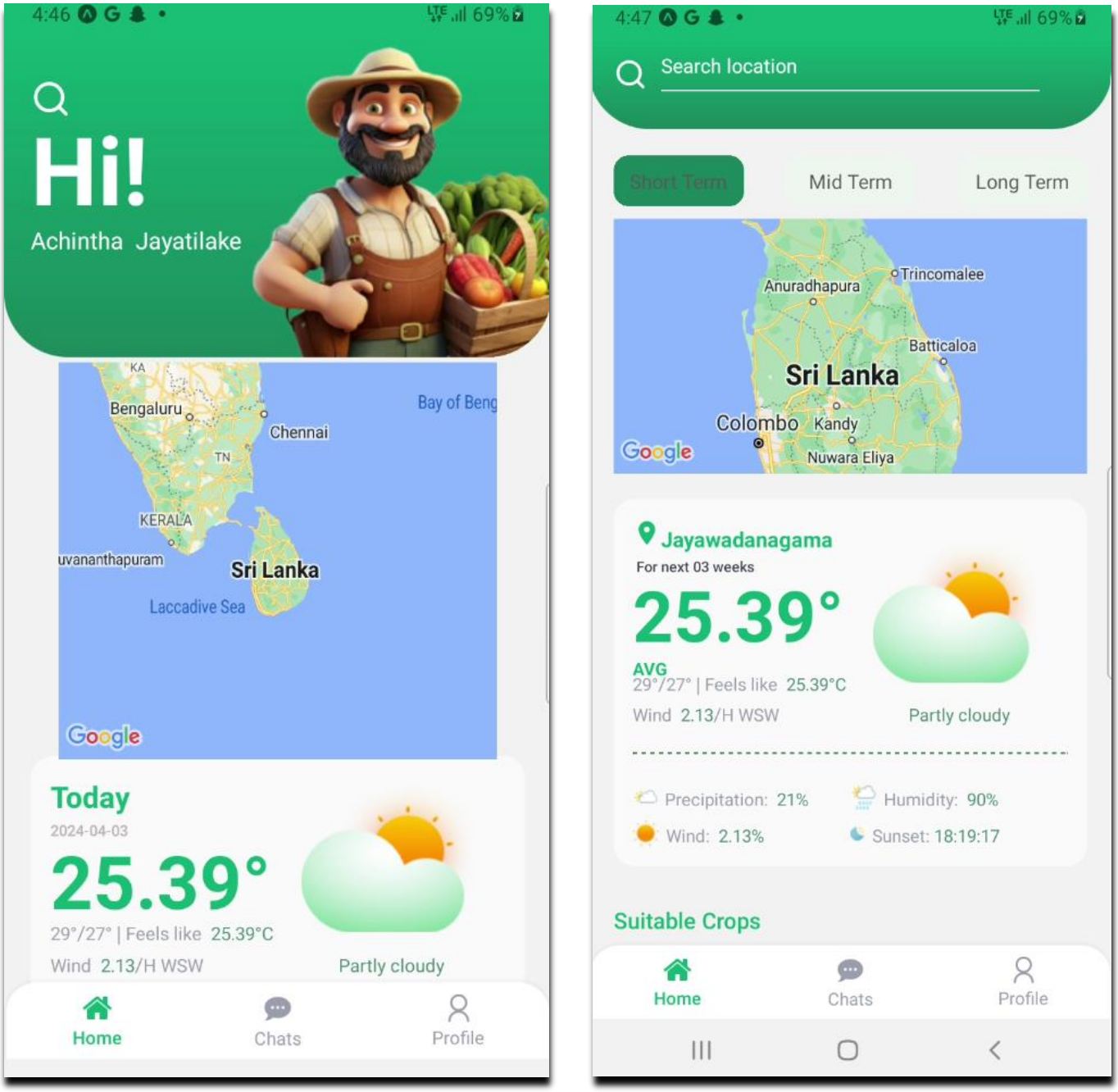


Figure 28 UI design Dashboard Screen

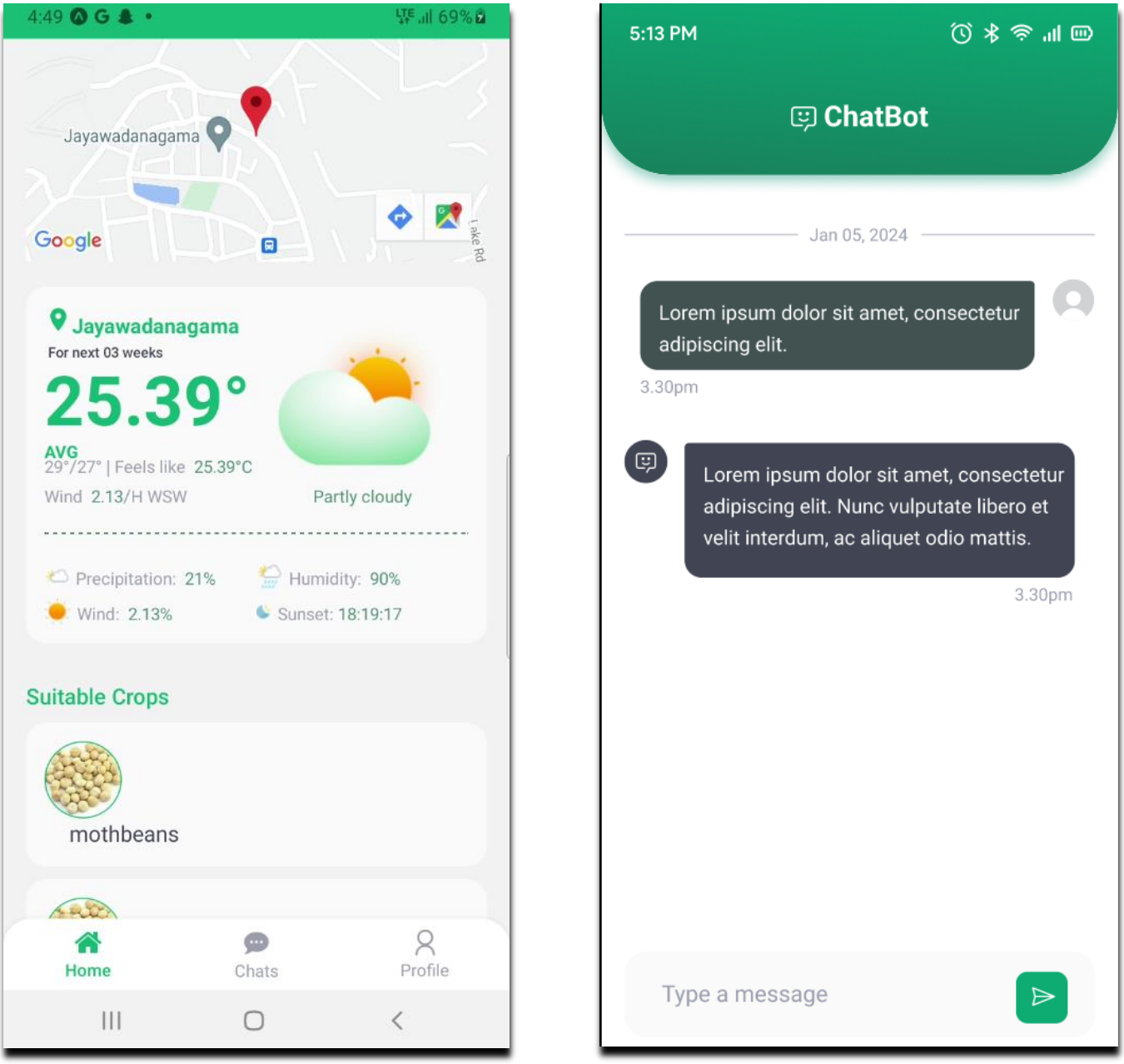


Figure 29 UI design of Location select & Chatbot

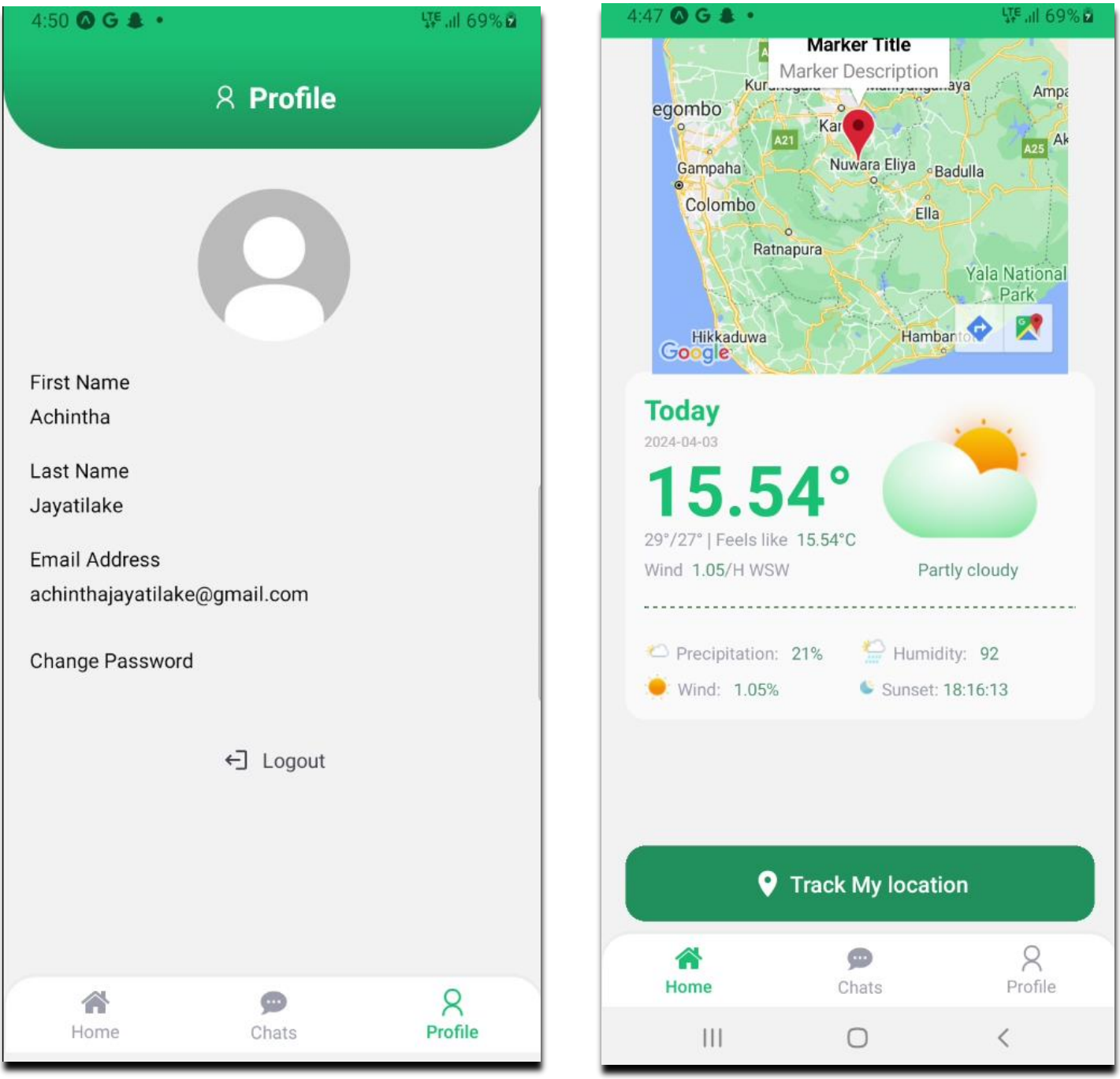


Figure 30 UI Design of User Profile

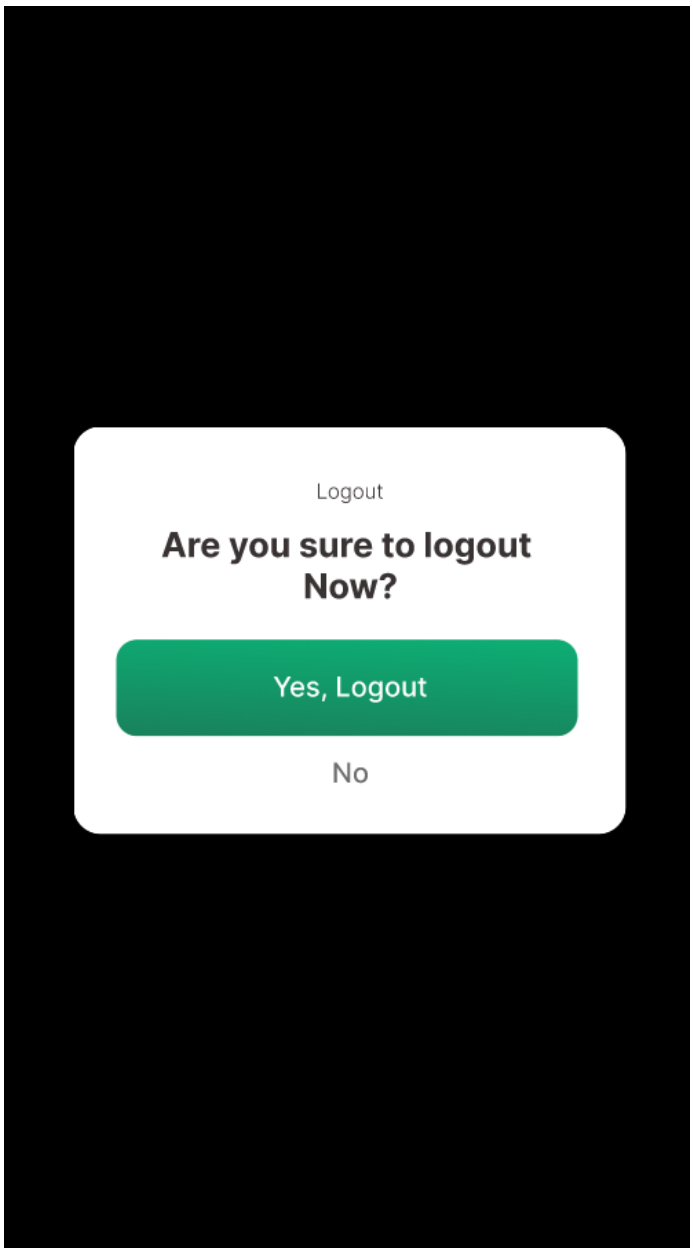


Figure 31 UI design of Log-out page

Famer App Code

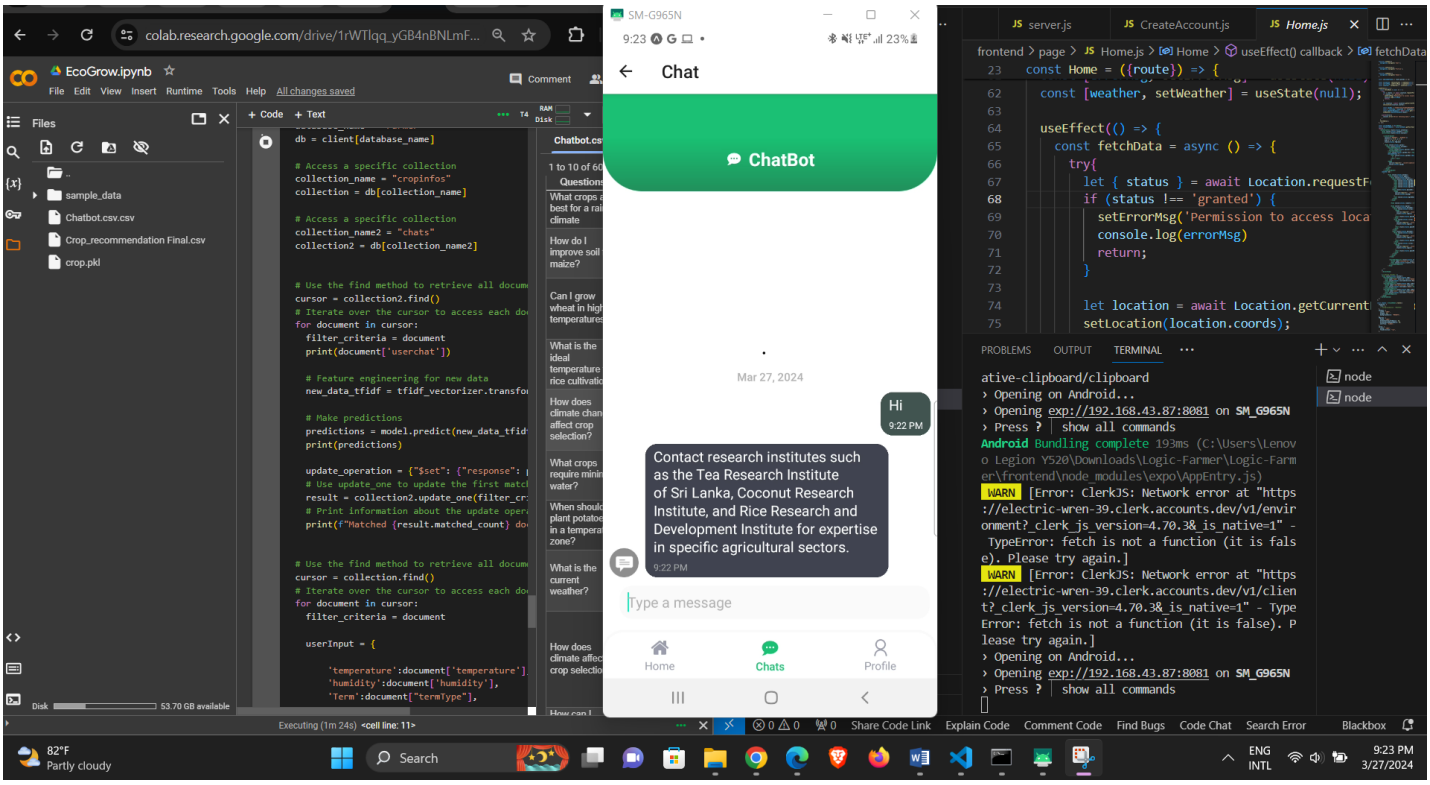


Figure 32 EcoGrow Mobile Application Testing

Frontend

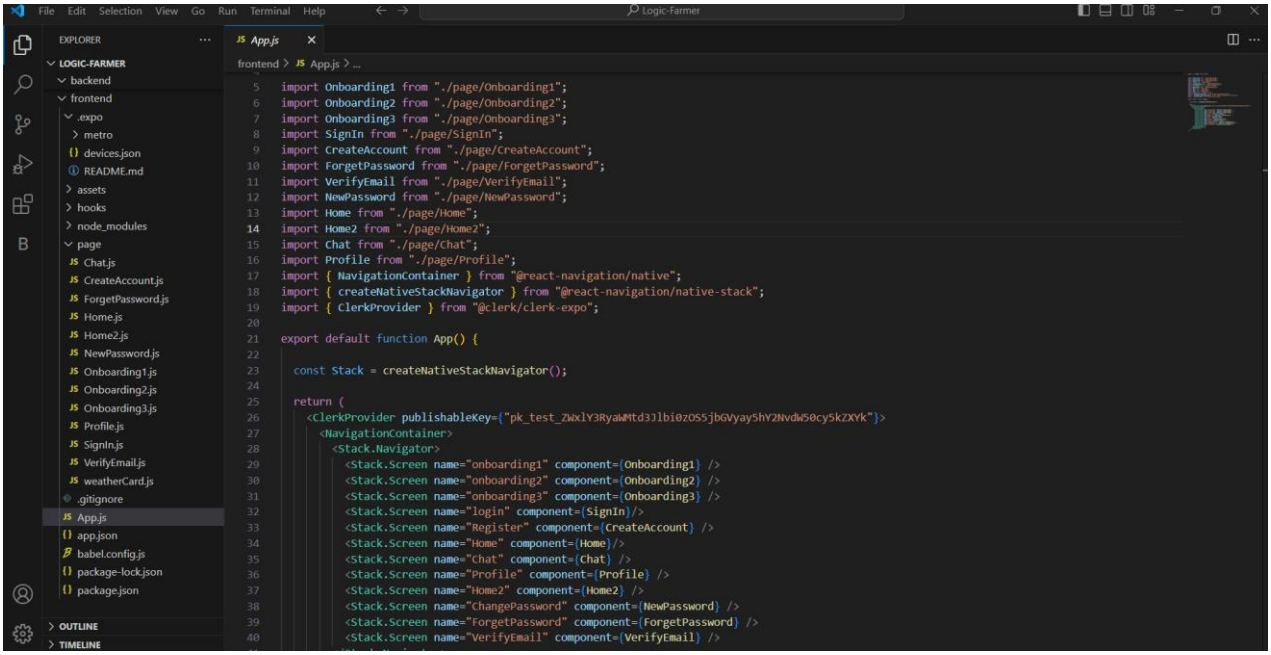


Figure 33 Frontend Code

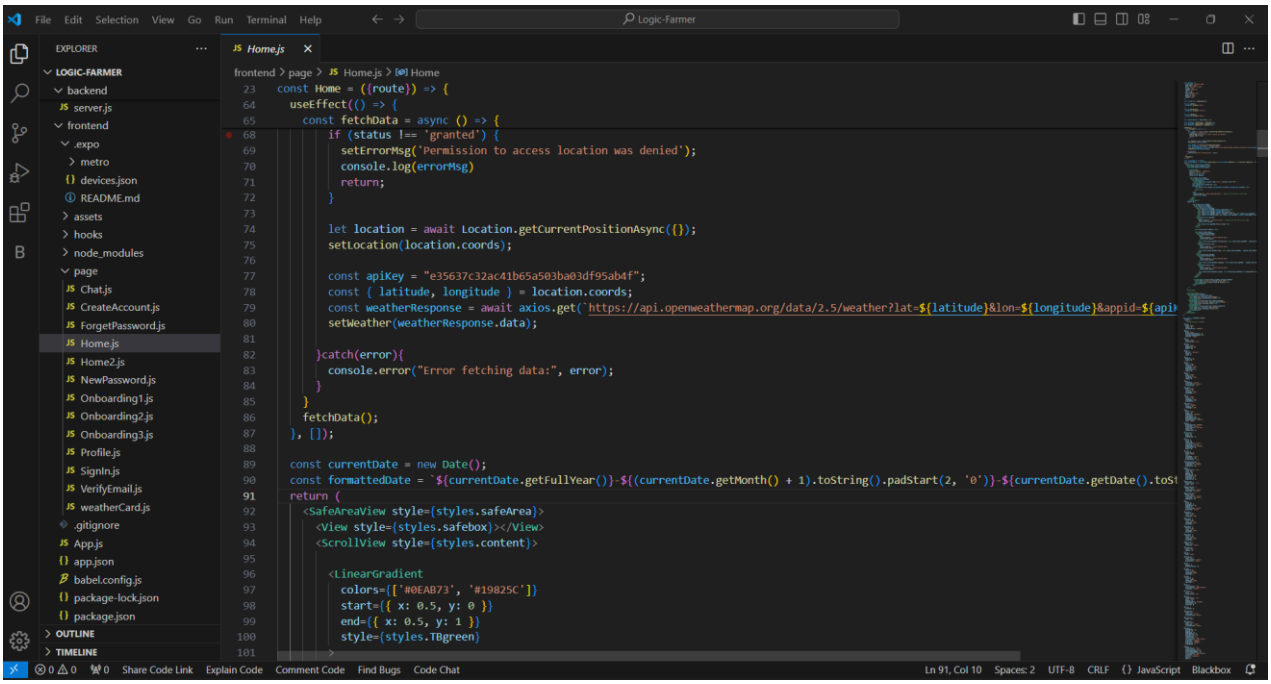


Figure 34 Frontend

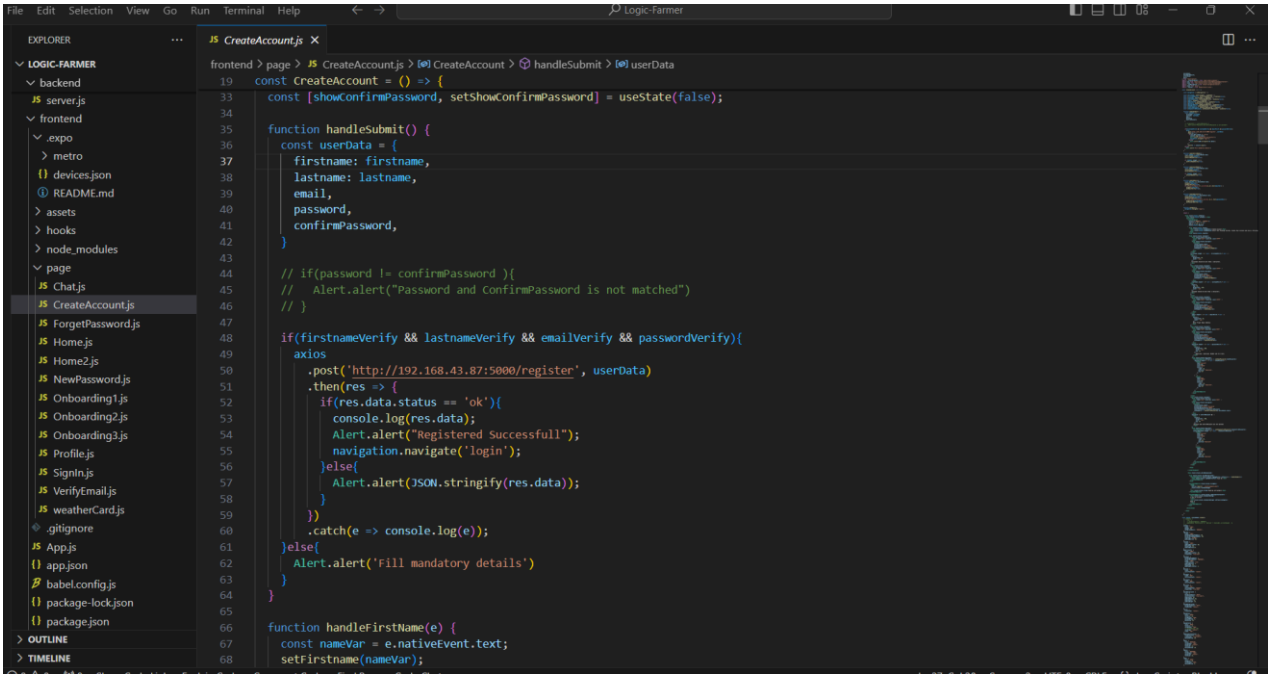
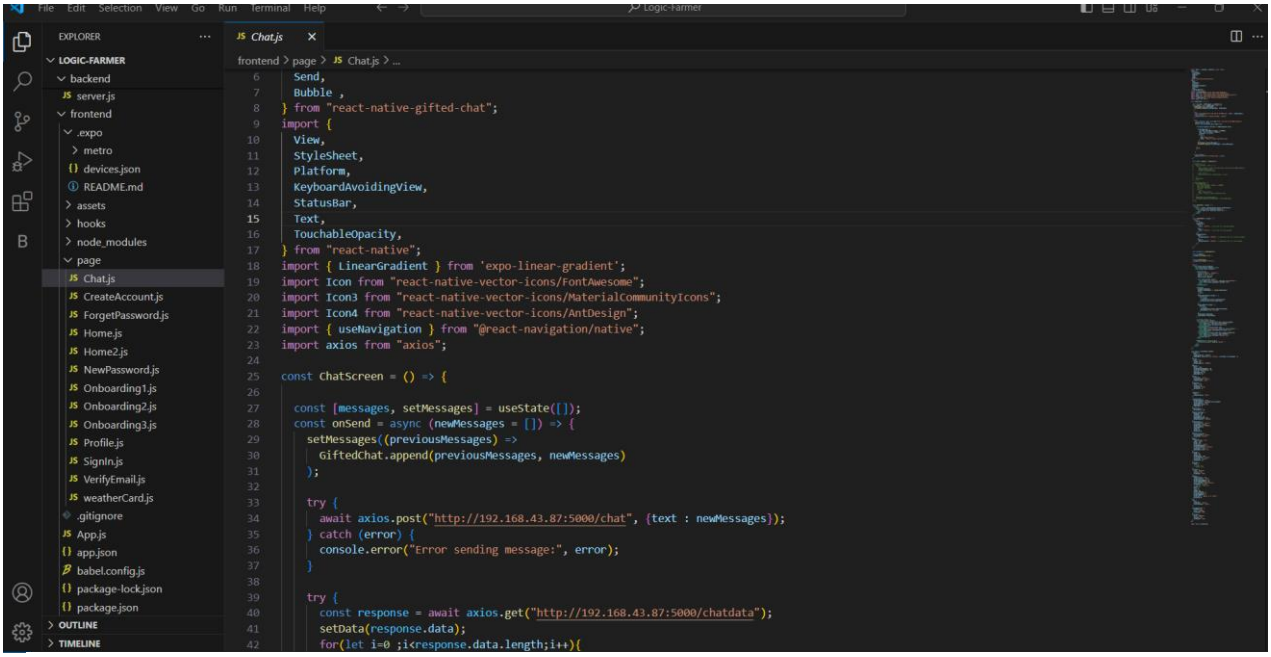


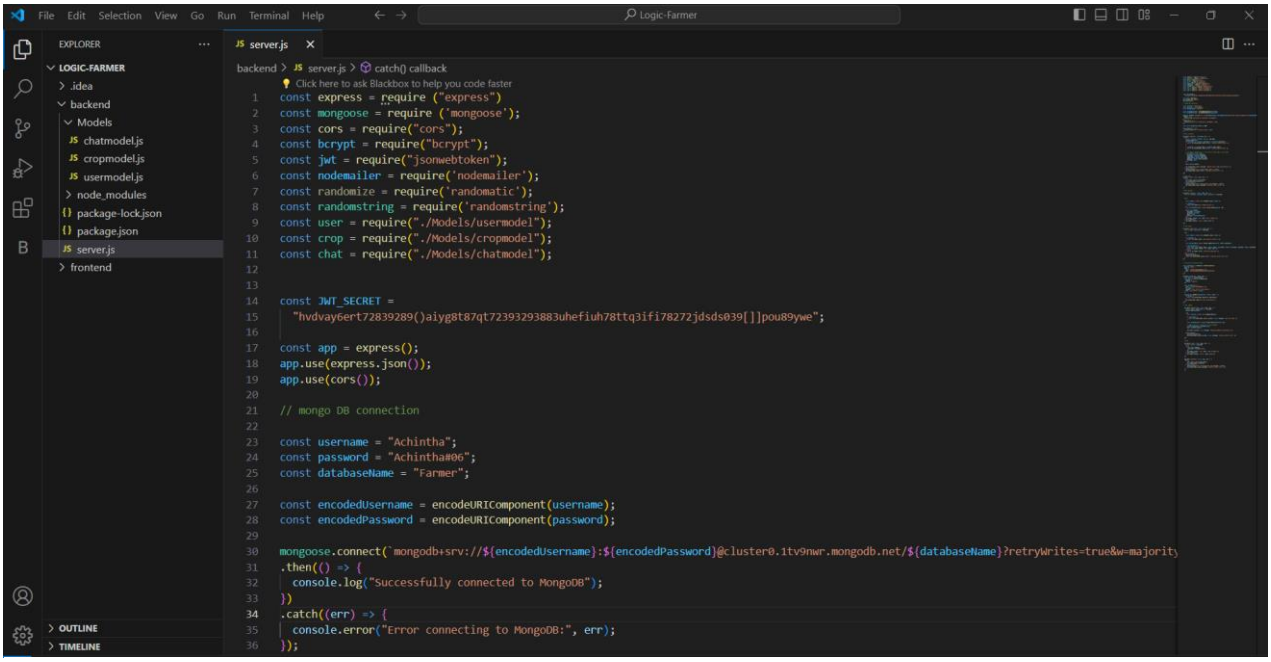
Figure 35 Frontend



```
import {
  Send,
  Bubble ,
} from "react-native-gifted-chat";
import {
  View,
  StyleSheet,
  Platform,
  KeyboardAvoidingView,
  StatusBar,
  Text,
  TouchableOpacity,
} from "react-native";
import { LinearGradient } from 'expo-linear-gradient';
import Icon from "react-native-vector-icons/FontAwesome";
import Icon3 from "react-native-vector-icons/MaterialCommunityIcons";
import Icon4 from "react-native-vector-icons/AntDesign";
import { useNavigation } from "@react-navigation/native";
import axios from "axios";

const ChatScreen = () => {

  const [messages, setMessages] = useState([]);
  const onSend = async (newMessages = []) => {
    setMessages((previousMessages) =>
      GiftedChat.append(previousMessages, newMessages)
    );

    try {
      await axios.post("http://192.168.43.87:5000/chat", {text : newMessages});
    } catch (error) {
      console.error("Error sending message:", error);
    }

    try {
      const response = await axios.get("http://192.168.43.87:5000/chatdata");
      setData(response.data);
      for(let i=0 ;i<response.data.length;i++){
```

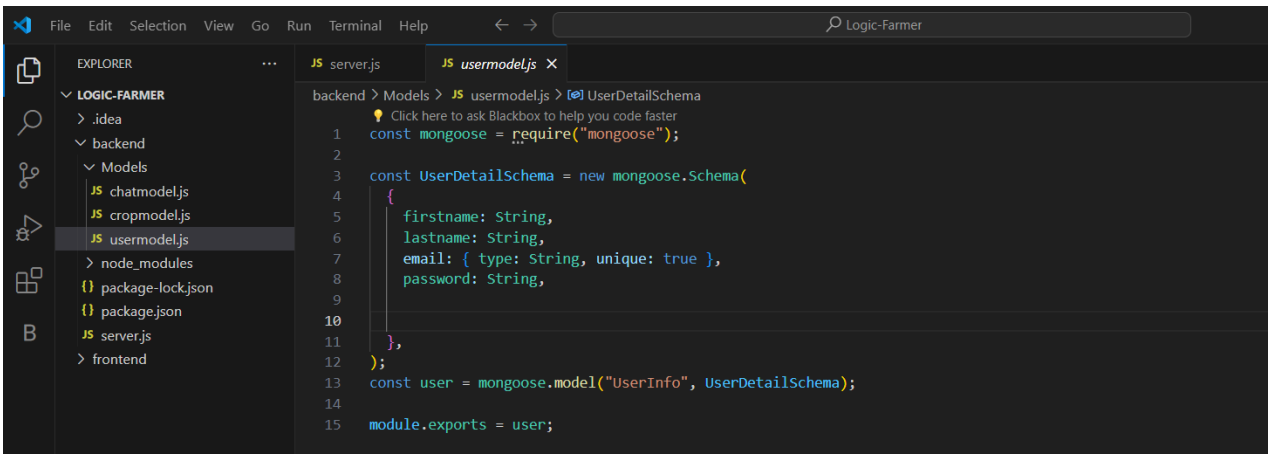
Figure 36 Frontend

Backend



```
const express = require ("express")
const mongoose = require ('mongoose');
const cors = require("cors");
const bcrypt = require("bcrypt");
const jwt = require("jsonwebtoken");
const nodemailer = require('nodemailer');
const randomize = require('randomatic');
const randomstring = require('randomstring');
const user = require("./Models/usermodel");
const crop = require("./Models/cropmodel");
const chat = require("./Models/chatmodel");


const JWT_SECRET =
  "hvdvay6ert72839289()aiyg8t87qt72393293883uhefiuh78ttq3ifi78272jdsds039[]]pou89ywe";

const app = express();
app.use(express.json());
app.use(cors());

// mongo DB connection

const username = "Achintha";
const password = "Achintha#06";
const databaseName = "Farmer";

const encodedUsername = encodeURIComponent(username);
const encodedPassword = encodeURIComponent(password);

mongoose.connect(`mongodb+srv://${encodedUsername}:${encodedPassword}@cluster0.1tv9nwr.mongodb.net/${databaseName}?retryWrites=true&w=majority
.then(() => {
  console.log("Successfully connected to MongoDB");
})
.catch((err) => {
  console.error("Error connecting to MongoDB:", err);
});
```

Figure 38 Backend



```
const mongoose = require("mongoose");

const UserDetailSchema = new mongoose.Schema(
  {
    firstname: String,
    lastname: String,
    email: { type: String, unique: true },
    password: String,


  },
);
const user = mongoose.model("UserInfo", UserDetailSchema);

module.exports = user;
```

Figure 40 Backend

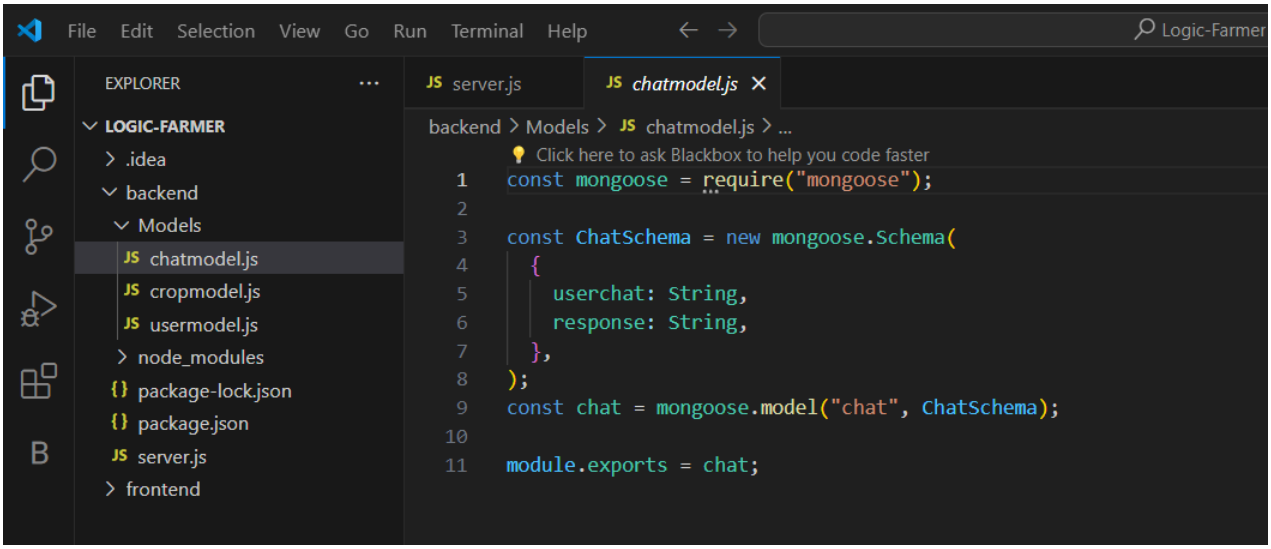


Figure 41 Backend

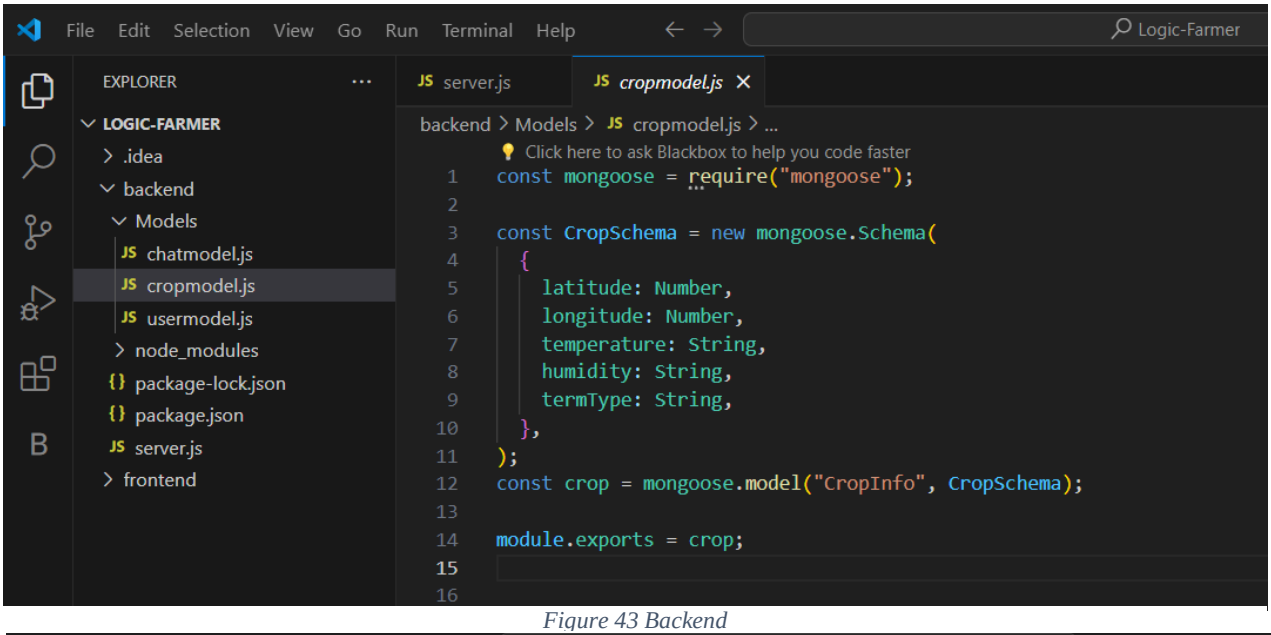


Figure 43 Backend

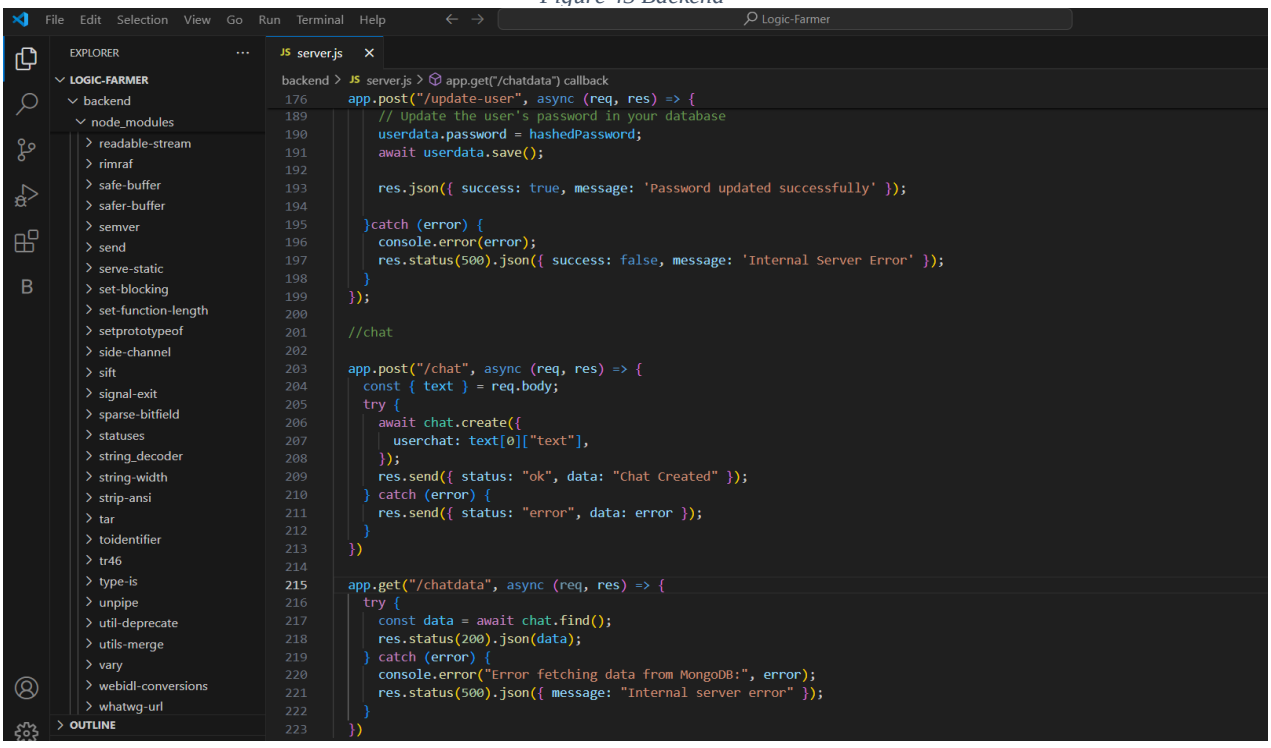


Figure 44 Backend

Dataset

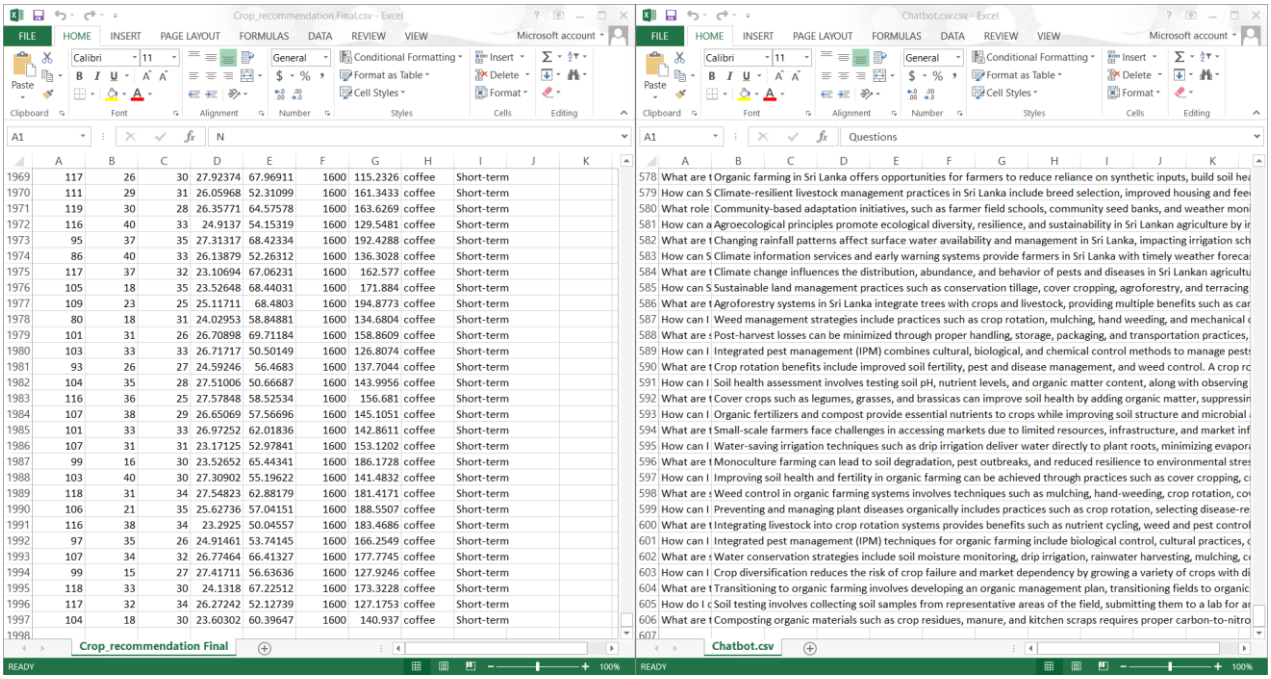


Figure 45 Data Sets CSV files

DataBase (MongoDB)

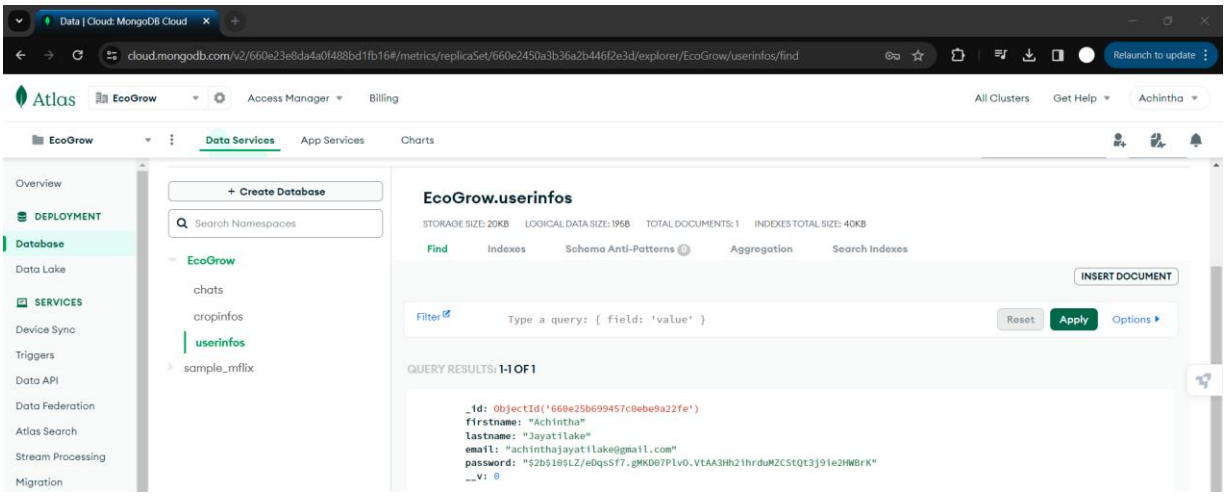


Figure 46 DatabaseUser info

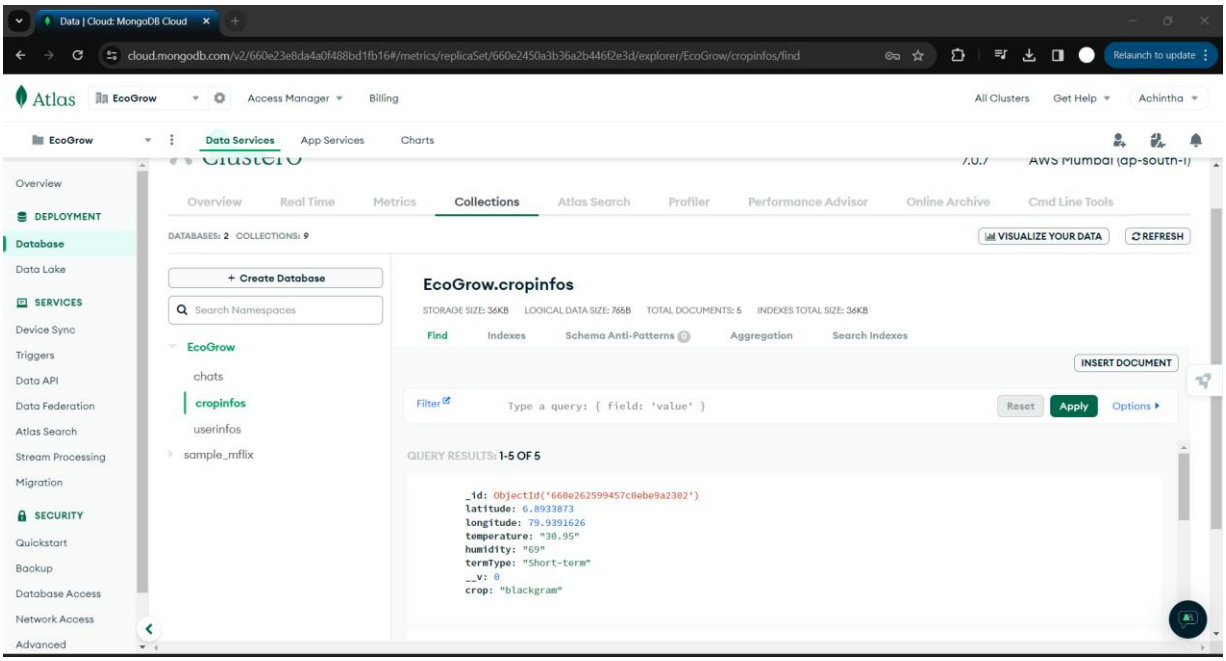


Figure 47 Database Crop info

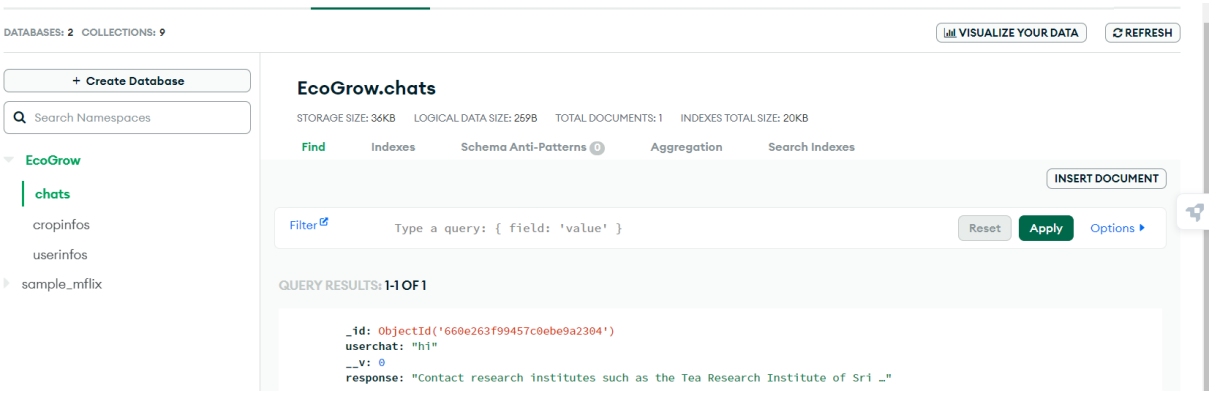


Figure 48 Database Chats info

Machine Learning Model

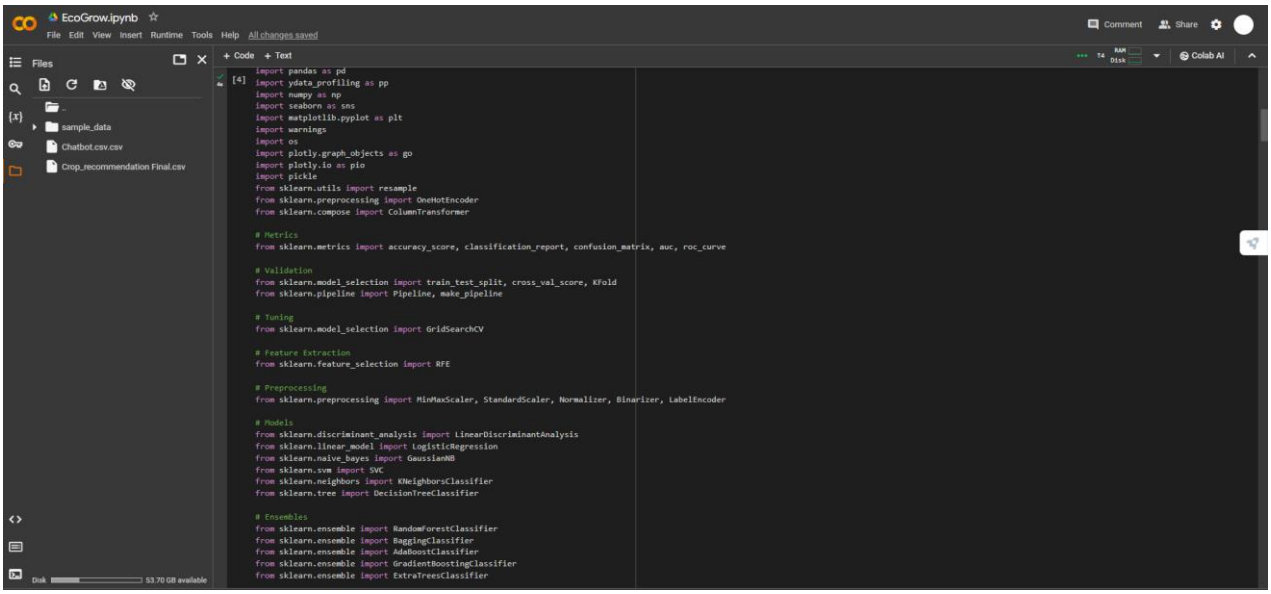


Figure 49 ML (Self-Composed)

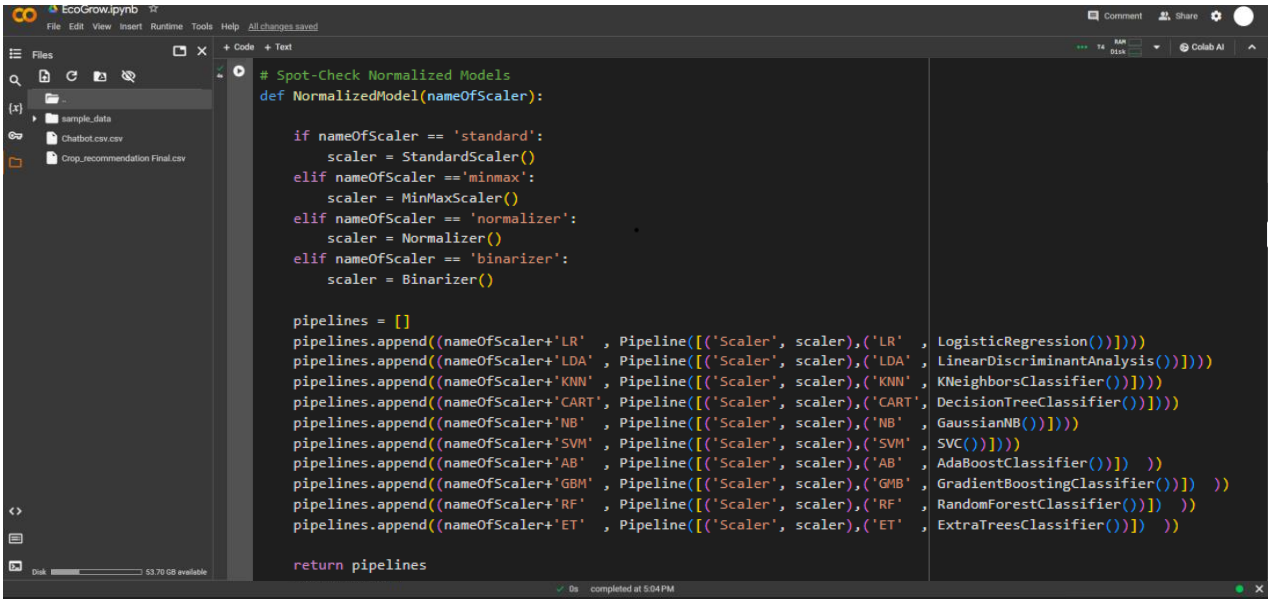


Figure 50 ML Checking the best Model (Self-Composed)

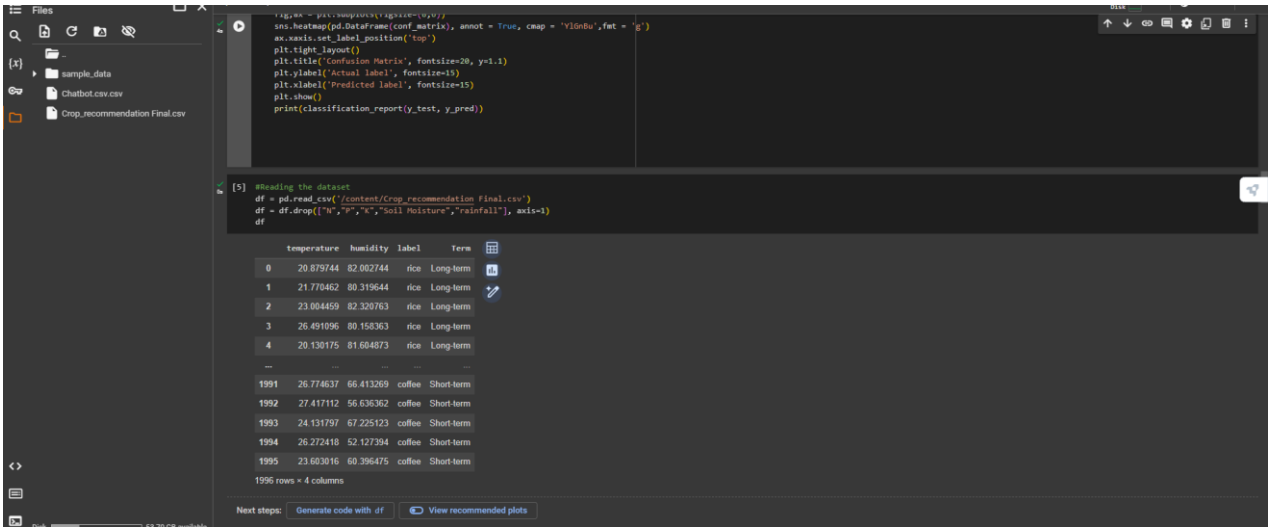


Figure 51 ML - Reading the data set (Self-Composed)

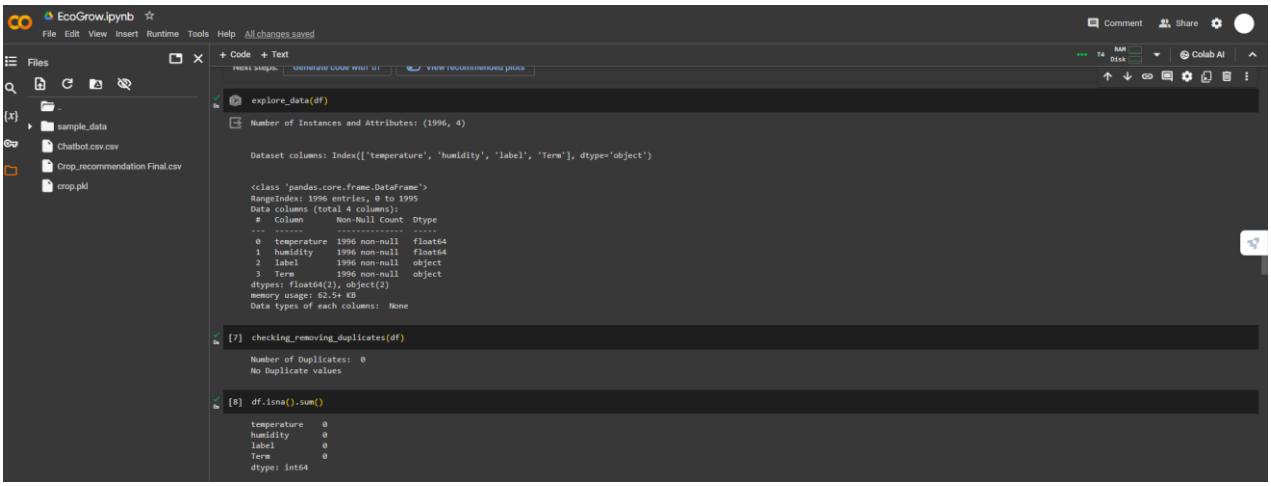


Figure 52 ML removing duplicates (Self-Composed)

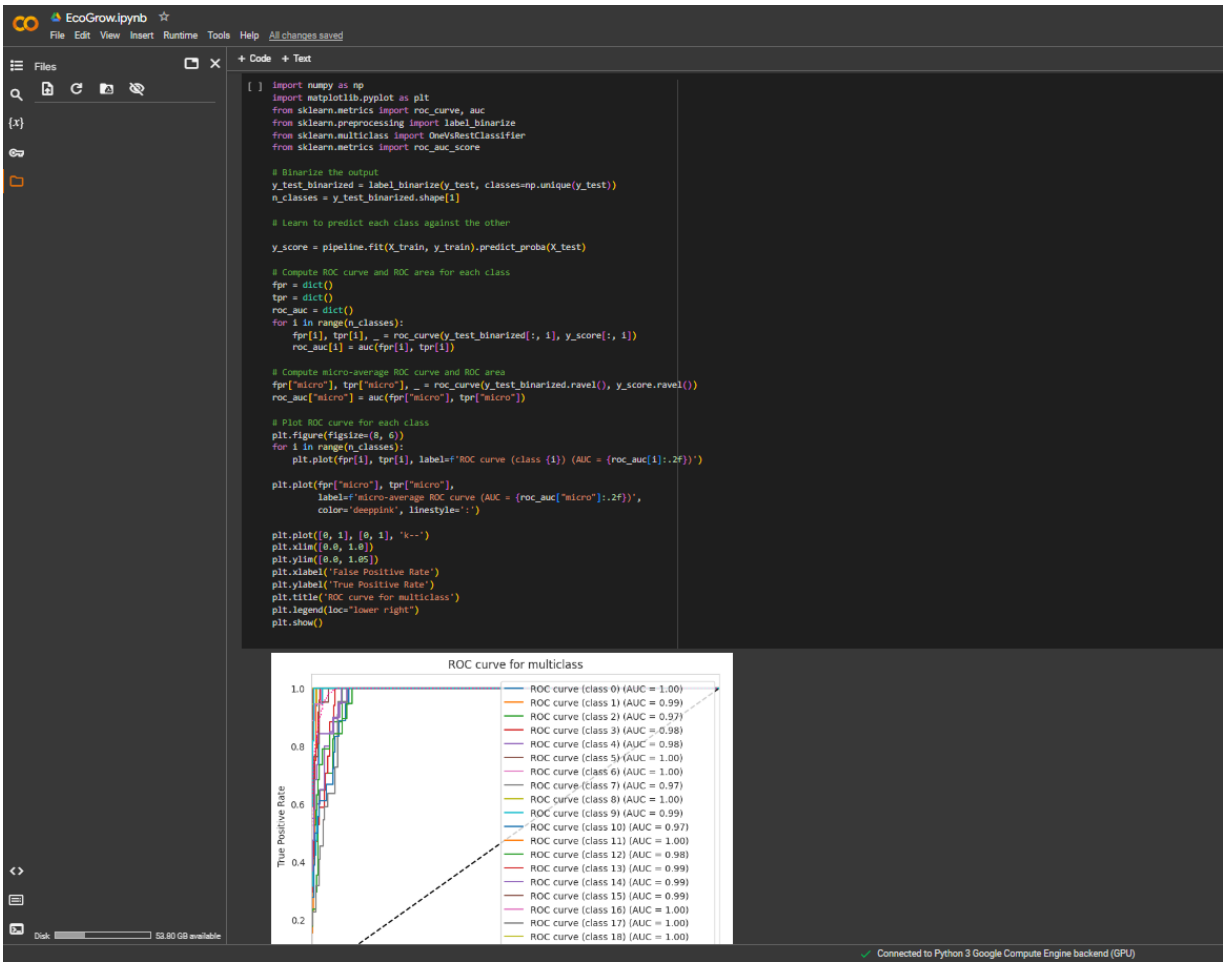


Figure 53 ML (Self-Composed)

Figure 54 ML (Self-Composed)

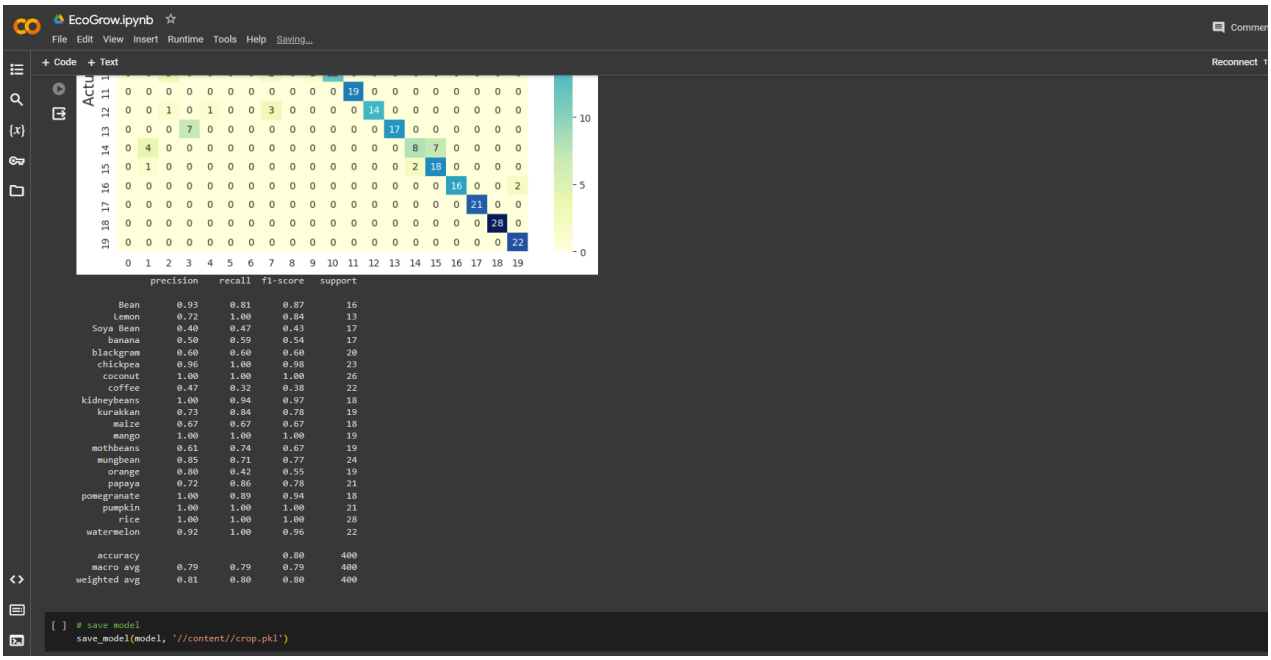


Figure 55 ML - Saving the model (Self-Composed)

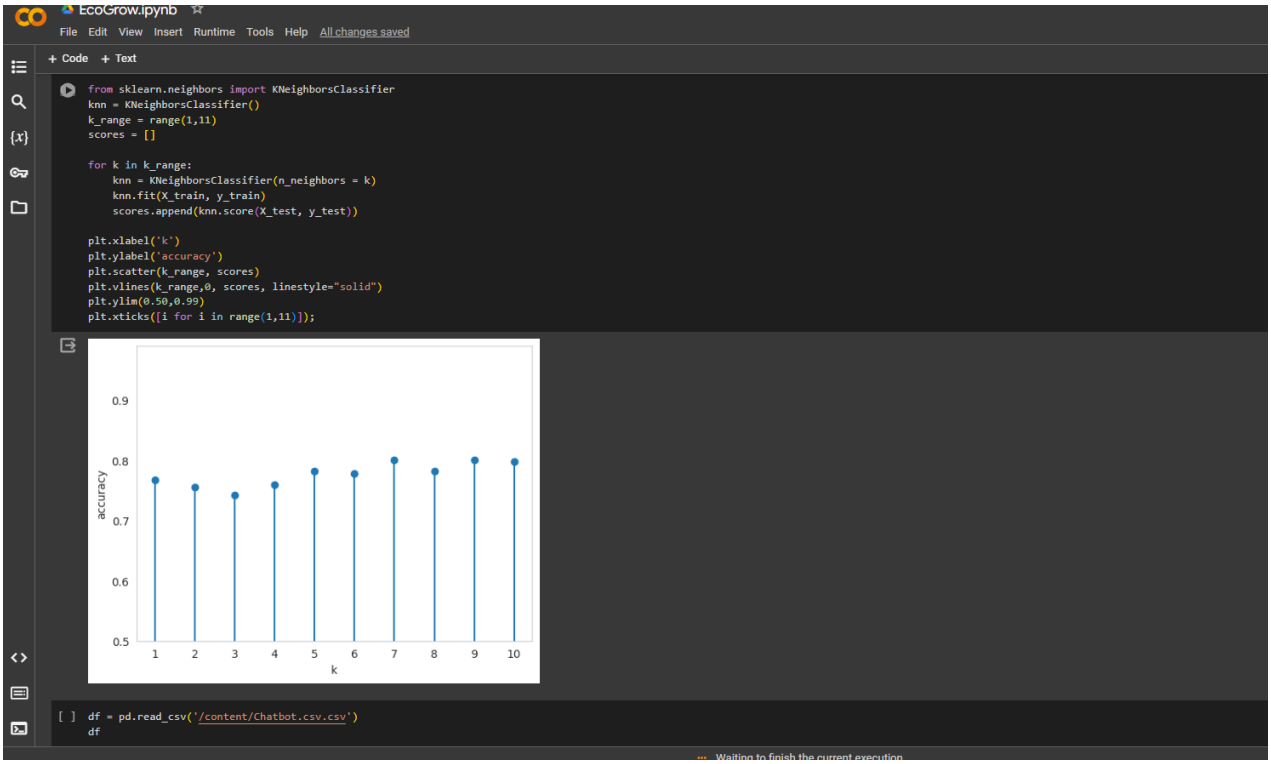


Figure 56 ML KNN (Self-Composed)

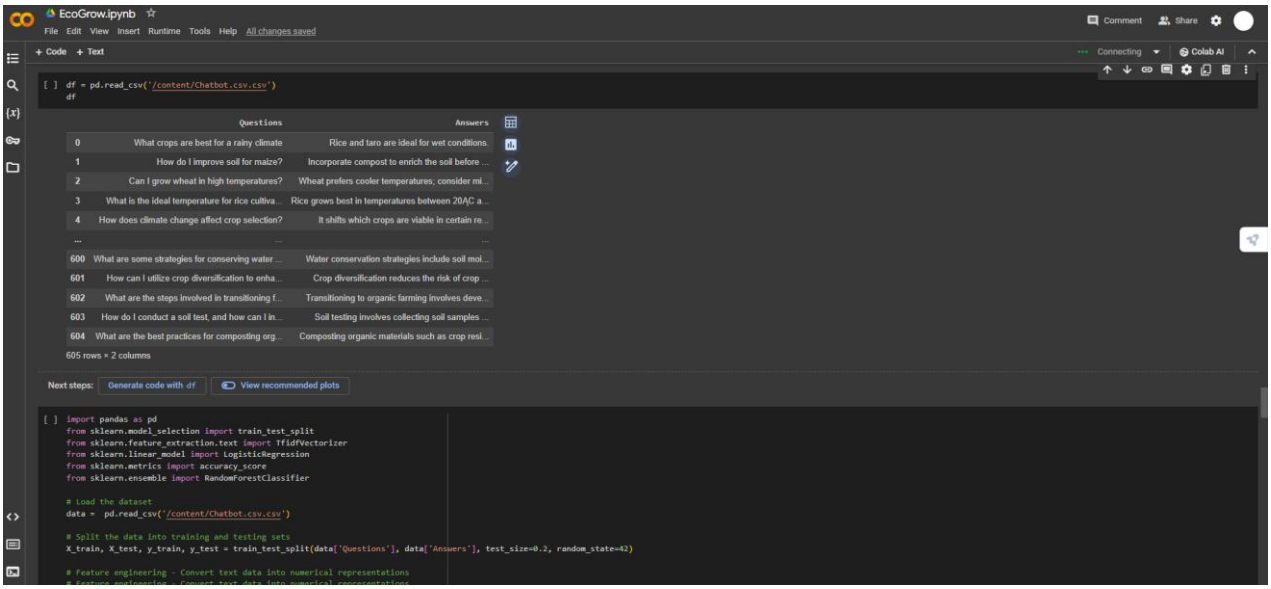


Figure 57 ML - Reading Chat bot Data set (Self-Composed)

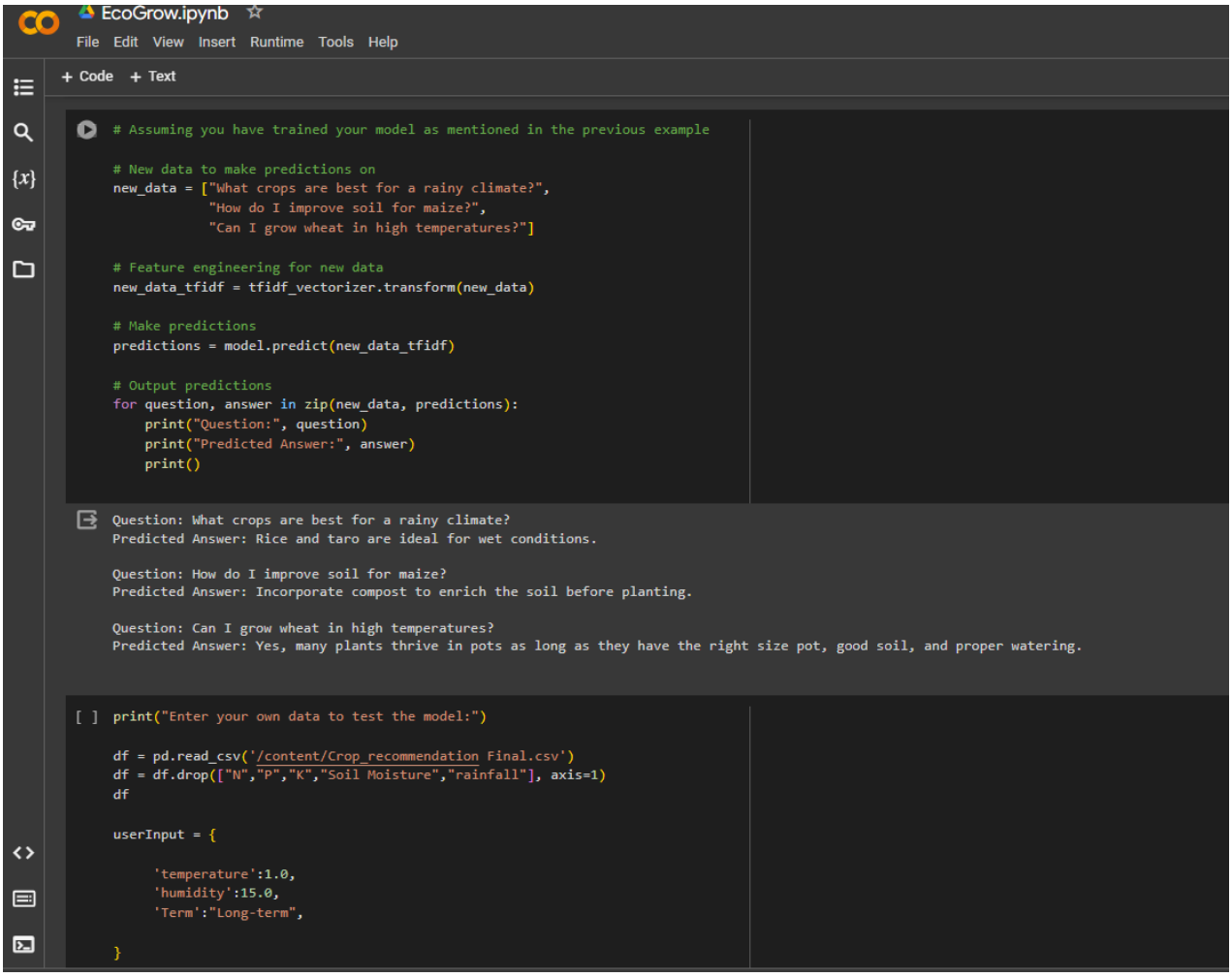


Figure 58 ML – Chatbot (Self-Composed)

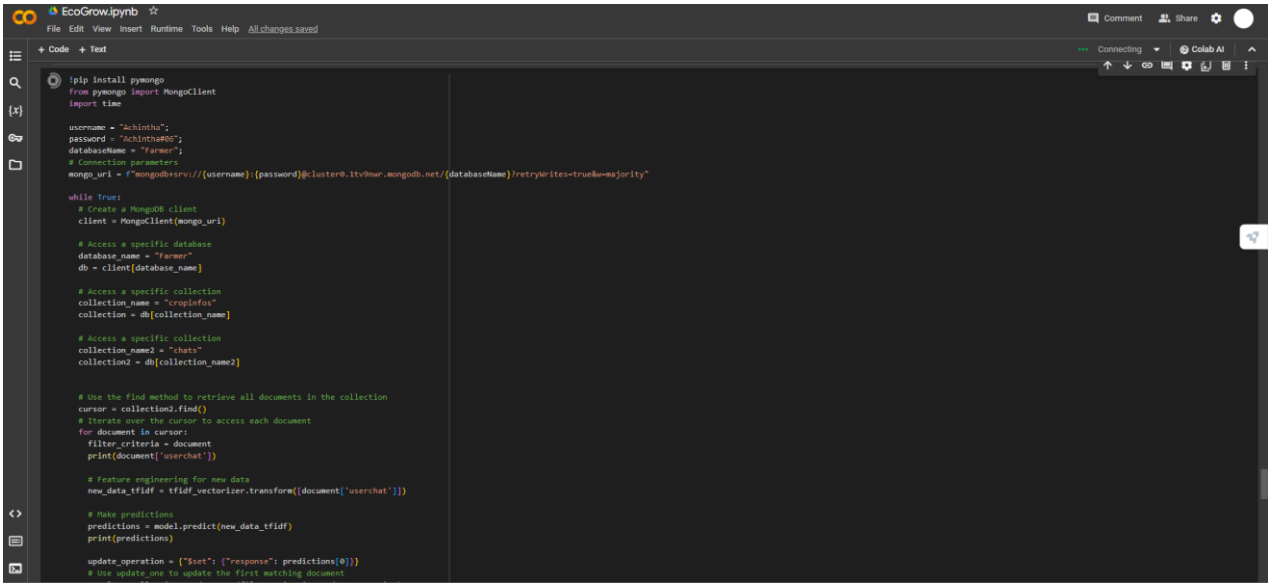


Figure 59 ML (Self-Composed)

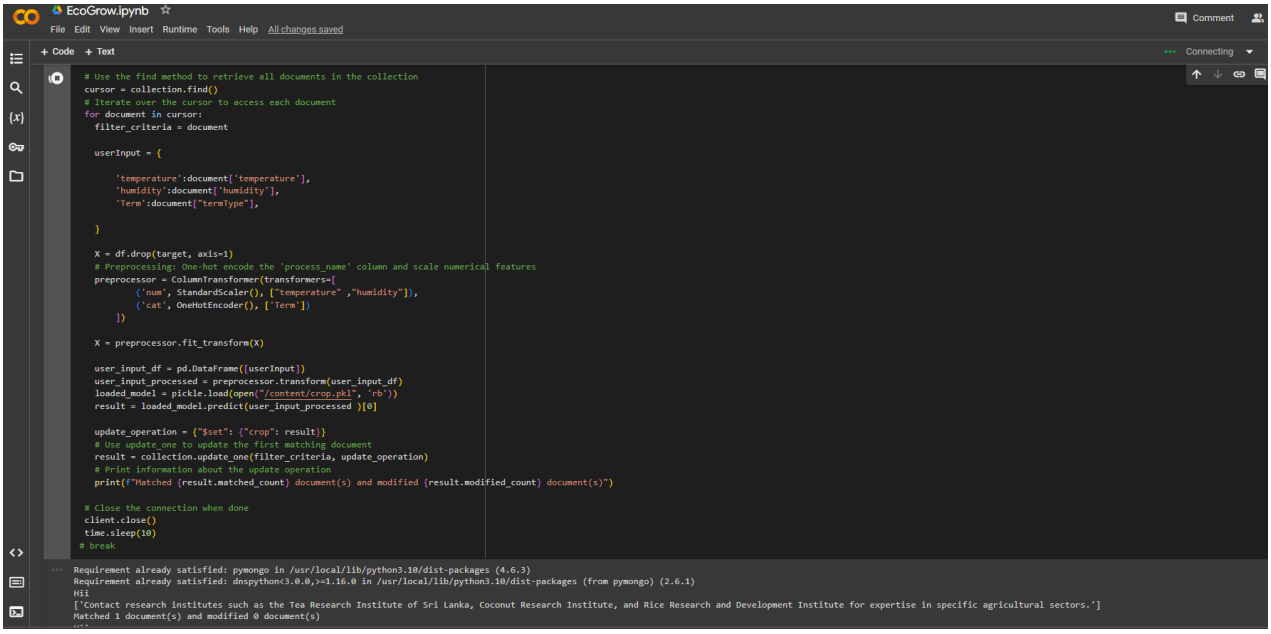


Figure 61 ML (Self-Composed)

Appendix 3 - Requirement Gathering Survey

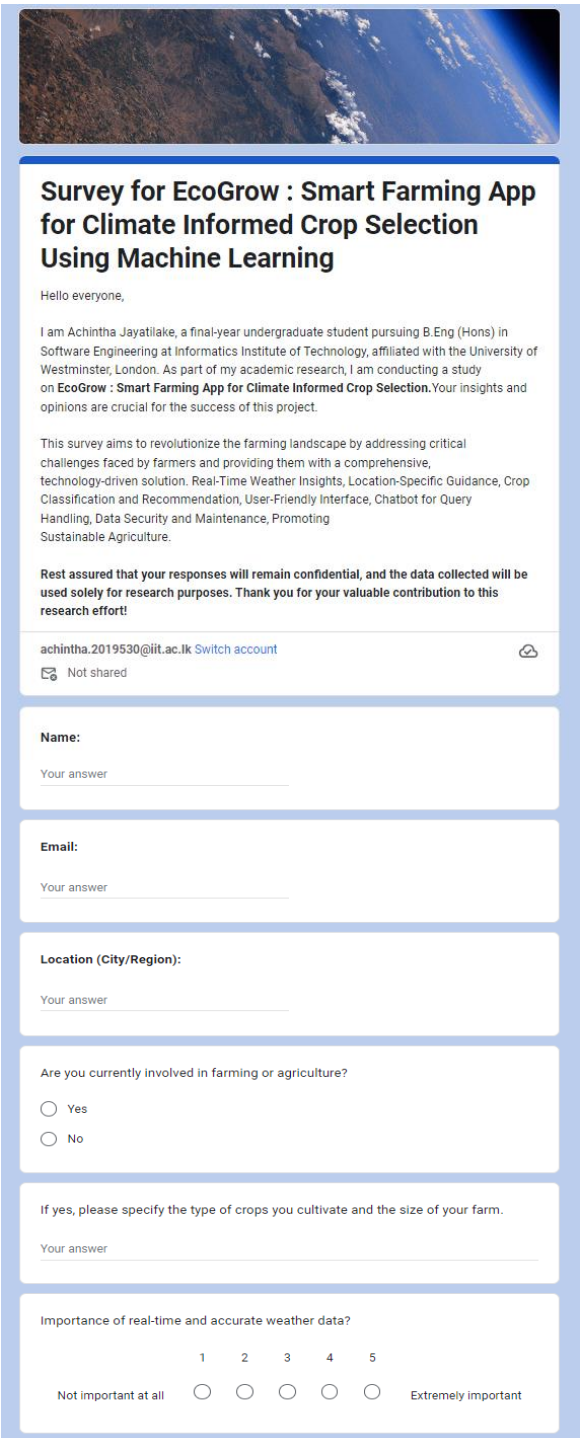


Figure 62 Requirement Gathering Survey Part 1

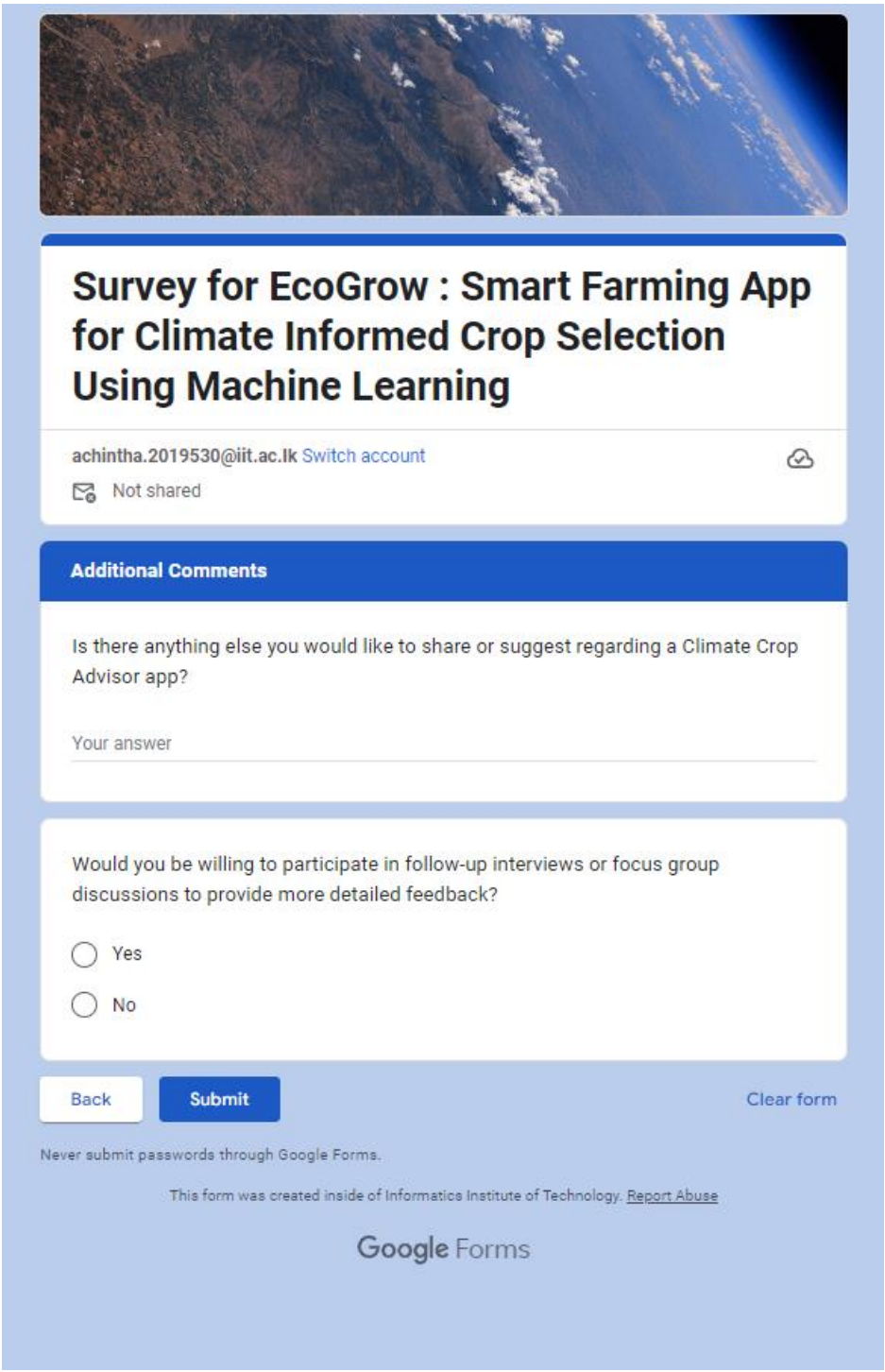


Figure 64 Requirement Gathering Survey Part II

Appendix 4 - Evaluators Feedback


Request for Feedback on Final Year Software Engineering Project Inbox ×

Achintha Jayatilake Mar 27, 2024, 9:35 AM (3 days ago)
Dear Misfar Siddeek, I hope this email finds you well. My name is Achintha Jayatilake, and I am a final year undergraduate Software Engineering student at the I

Mohemed Misfar <asm.misfar@gmail.com> Mar 27, 2024, 4:40 PM (3 days ago)
to me

Dear Achintha Jayatilake,

Thank you for reaching out and sharing your project, EcoGrow. It's fascinating to see the integration of machine learning and backend architecture to support agricultural practices.

After reviewing your project, I believe there are some areas where enhancements could be made to optimize the backend components for scalability and efficiency. Here are some suggestions:

Your backend architecture demonstrates robustness, but optimizing database queries and implementing caching mechanisms could enhance system performance, particularly during peak usage hours.
Consider adopting microservices architecture to decouple components and improve scalability, allowing for easier maintenance and future expansion of the application.

I hope you find these suggestions valuable. Please don't hesitate to contact me if you have any further questions or need additional assistance.

Best regards,

Misfar Siddeek

Figure 65 Feedback By Mr. Misfar Siddeek

Your project, EcoGrow. It's fascinating to see the integration of machine learning and backend architecture to support agricultural practices.

After reviewing your project, I believe there are some areas where enhancements could be made to optimize the backend components for scalability and efficiency. Here are some suggestions:

Your backend architecture demonstrates robustness, but optimizing database queries and implementing caching mechanisms could enhance system performance, particularly during peak usage hours.
Consider adopting microservices architecture to decouple components and improve scalability, allowing for easier maintenance and future expansion of the application.

I hope you find these suggestions valuable. Please don't hesitate to contact me if you have any further questions or need additional assistance.

-Mr. Misfar Siddeek -

Request for Feedback on Final Year Software Engineering Project Inbox ×

Achintha Jayatilake
Dear C K W Jayatilake, I hope this email finds you well. My name is Achintha Jayatilake, and I am a final year undergraduate Software Engineering student at the

Kolitha jayatilake
to me

Dear Achintha Jayatilake,

Thank you for reaching out and sharing your project, EcoGrow, with me. I commend your efforts to leverage technology for the benefit of the agricultural community.

Having reviewed your project, I am impressed by its potential to provide valuable assistance to farmers. Here are some insights and suggestions based on my expertise in farming agriculture:

Your project aligns well with the needs of farmers, providing valuable insights and recommendations. To cater to diverse farming practices, consider incorporating regional variations in soil types and farming techniques into your crop recommendation model. Collaborating with agricultural extension services or local farming communities can enrich your dataset with practical insights and real-world feedback, further enhancing the applicability and relevance of your application.

I hope you find this feedback helpful in further enhancing the effectiveness of your project. Please feel free to reach out if you have any questions or require additional input.

Best regards,

C K W Jayatilake

Yahoo Mail: Search, organise, conquer

Figure 66 Feedback By Mr.C K W Jayatilake

I commend your efforts to leverage technology for the benefit of the agricultural community.

Having reviewed your project, I am impressed by its potential to provide valuable assistance to farmers. Here are some insights and suggestions based on my expertise in farming agriculture:

Your project aligns well with the needs of farmers, providing valuable insights and recommendations. To cater to diverse farming practices, consider incorporating regional variations in soil types and farming techniques into your crop recommendation model. Collaborating with agricultural extension services or local farming communities can enrich your dataset with practical insights and real-world feedback, further enhancing the applicability and relevance of your application.

I hope you find this feedback helpful in further enhancing the effectiveness of your project. Please feel free to reach out if you have any questions or require additional input.

-Mr. C K W Jayatilake-