THE TECHNICAL UNIVERSITY OF KENYA

SCHOOL : SCHOOL OF COMPUTING AND INFORMATION TECHNOOGY

DEPARTMENT : COMPUTER SCIENCE AND INFORMATICS

COURSE : BACHELOR OF TECHNLOGY IN INFORMATION TECHOLOGY

TITLE: **FARMERS HUB**

MEMBERS

SCCJ/01060/2021 - MATHEW MAGENDA
SCCJ/00653/2021 - PETER MWANGI
SCCJ/02813/2021 - ADNAAN OSMAN

DECLARATION

We declare that this business plan is our original work and has not been copied from any other source. This document has been compiled and researched by us to the best of our ability.

Signature: _________

Date: _________

Supervisor's Name: ________

Supervisor's Signature: ________

Date: _________

ACKNOWLEDGEMENT

We would like to express my sincere gratitude to our families and friends for their constant encouragement and belief in our vision. Above all, we thank God for providing us with the strength and inspiration to pursue this endeavour.

DEDICATION

This business plan is dedicated to our friends for their unwavering support and inspiration. Their encouragement and belief in our abilities have been instrumental in the realization of this project.

Table of Contents

CHAPTER ONE

Introduction

In the rapidly evolving agricultural sector, technology plays a crucial role in enhancing productivity and efficiency. The project area for this report is a farming app designed to support farmers by providing up-to-date weather information and facilitating engagement with agronomists. The app operates by aggregating weather data from various sources and delivering tailored updates to farmers, enabling them to make informed decisions regarding their farming practices. Additionally, it offers a platform for farmers to connect with agronomists, who can provide expert advice and support.

The problem that this app addresses is rooted in the challenges faced by farmers in accessing timely and accurate weather information and expert guidance. Traditionally, farmers have relied on sporadic weather forecasts and limited access to agronomic expertise, which can lead to suboptimal decision-making and decreased productivity. By integrating weather updates and agronomist engagement into a single platform, the app aims to provide a comprehensive solution that meets the needs of modern farmers.

Background of the Study

The farming app is developed for a client engaged in agricultural technology solutions. The client operates in a competitive market, providing tools and services that enhance agricultural efficiency. Their business focuses on leveraging technology to address common challenges faced by farmers, including weather forecasting and access to expert advice.

Currently, the client's operations involve developing and maintaining various agricultural tools and platforms. However, they identified a gap in the market for a unified solution that integrates weather updates with agronomist consultation. This gap led to the conception of the farming app, which is designed to streamline these services and offer a more cohesive user experience.

Problem Statement(s)

The primary problem the farming app aims to address is the lack of integrated access to real-time weather information and agronomic expertise for farmers. Traditionally, farmers have faced several issues:

1. **Fragmented Weather Information:** Farmers often rely on multiple sources for weather updates, which can be inconsistent and difficult to interpret. This fragmentation hampers their ability to make timely and informed decisions regarding planting, harvesting, and other critical activities.
2. **Limited Access to Agronomic Expertise:** Access to qualified agronomists is often limited, especially in remote or underserved areas. Farmers may struggle to obtain expert advice on crop management, pest control, and other aspects of farming, leading to reduced productivity and potential losses.

Objectives

The objectives of the farming app project are designed to address specific needs within the agricultural sector, focusing on integrating weather updates and agronomist engagement into a unified platform. The objectives are formulated to be SMART:

1. **Investigate the current challenges faced by farmers regarding weather forecasting and agronomist consultation**:
   - **Specific:** Identify gaps in existing weather and agronomy services.
   - **Measurable:** Conduct surveys and interviews with at least 50 farmers and agronomists.
   - **Achievable:** Utilize existing research methods and tools.
   - **Relevant:** Address key issues affecting farming productivity.
   - **Time-bound:** Complete investigation within 1 month.
2. **Develop a user-friendly farming app that integrates real-time weather updates and agronomist engagement**:
   - **Specific:** Design and implement features for weather forecasts and expert consultations.
   - **Measurable:** Ensure the app includes at least three core functionalities: weather updates, agronomist chat, and notifications.
   - **Achievable:** Leverage available technology and development resources.
   - **Relevant:** Provide a comprehensive solution for modern farmers.
   - **Time-bound:** Complete development within 4 months.

Scope and Limitation of the Study

Scope: The project focuses on developing a farming app that provides real-time weather updates and facilitates engagement with agronomists. The scope includes:

- Designing and implementing features for weather forecasts and agronomic consultations.
- Ensuring the app is accessible on major mobile platforms (iOS and Android).
- Conducting user testing to validate functionality and user experience.

Limitations:

- The app will not include advanced agricultural tools like soil testing or pest management, which are beyond the current project scope.
- The project is limited to developing the app's core functionalities and does not cover extensive marketing or large-scale deployment strategies.
- Data accuracy depends on the quality of external weather services and agronomic input.

Justification

The farming app presents several compelling advantages and challenges:

- **Interestingness:** The integration of real-time weather updates with direct agronomist engagement is a novel approach that addresses significant gaps in current agricultural tools.
- **Challenge:** Developing an app that meets the needs of diverse users and operates reliably under varying conditions is a complex and challenging task.
- **Timeliness:** With increasing reliance on technology in agriculture, the app provides a timely solution that aligns with current trends in digital farming.
- **Advantages:** The app's realization offers improved decision-making for farmers, better access to expert advice, and potentially increased productivity and efficiency in farming operations.

Project Risk and Mitigation

Technical Issues

- **Mitigation:** Employ rigorous testing and quality assurance processes throughout development.

Data Accuracy

- **Mitigation:** Partner with reputable weather data providers and regularly review data accuracy.

User Adoption

- **Mitigation:** Conduct user training sessions and provide clear documentation to facilitate smooth adoption.

Budget Overruns

- **Mitigation:** Develop a detailed budget and regularly monitor expenses to stay within financial constraints.

viii. Budget and Resources

Hardware:

- Development and testing devices (e.g., smartphones, tablets).

Software:

- Development tools and platforms (e.g., IDEs, cloud services).

Human Resources:

- Software developers, UI/UX designers, project managers, and quality assurance testers.

Other Costs:

- Licensing fees for third-party services (e.g., weather data providers).
- Marketing and user training expenses.

ix. Project Schedule

The project schedule includes a work breakdown structure outlining key phases:

1. **Research and Planning:** 1 month

2. **Design and Development:** 1month
3. **Testing and Refinement:** 1 month

CHAPTER TWO: LITERATURE REVIEW

The literature review provides a comprehensive analysis of existing research and developments related to agricultural technology, with a focus on weather forecasting tools and agronomist engagement platforms. This chapter critically examines the contributions of previous studies and technologies to identify gaps and opportunities for innovation within the context of the farming app project.

Weather Forecasting in Agriculture

1 Historical Overview of Weather Forecasting Tools

Early agricultural weather forecasting relied on traditional methods and observational data. Farmers used local weather patterns and seasonal signs to predict weather conditions. As technology advanced, tools such as meteorological stations and satellite data became available, providing more accurate and timely weather forecasts.

2 Modern Weather Forecasting Technologies

Recent advancements include the integration of weather data with digital platforms, offering real-time updates and predictive analytics. Studies lhave demonstrated the effectiveness of these technologies in improving crop management and reducing risks. The use of machine learning algorithms for weather prediction, as discussed by [Author, Year], has further enhanced forecast accuracy and utility for farmers.

Agronomist Engagement and Digital Platforms

1 The Role of Agronomists in Modern Agriculture

Agronomists play a crucial role in advising farmers on crop management, soil health, and pest control. Their expertise helps optimize agricultural practices and improve yields. The importance of accessible agronomic advice is well-documented in studies such as [Author, Year], which emphasize the benefits of expert consultation for enhancing agricultural productivity.

2 Digital Platforms for Agronomist Engagement

The rise of digital platforms has facilitated greater access to agronomic expertise. They offer online consultation services, allowing farmers to connect with experts remotely.

Gaps in Existing Research and Opportunities for Innovation

1 Identified Gaps

While existing literature covers various aspects of weather forecasting and agronomist engagement, there are gaps in integrating these elements into a single, cohesive platform. The need for an integrated approach that combines real-time weather updates with expert consultation is evident.

2 Opportunities for Innovation

The development of a farming app that integrates weather data and agronomist engagement presents an opportunity to address these gaps. By creating a unified platform, the app can offer farmers a streamlined experience, improving access to critical information and expert advice.

CHAPTER THREE: METHODOLOGY

This chapter outlines the methodology employed in the development of the farming app. It describes the model/framework used, the techniques for data collection, tools for data analysis, and processes for implementation and testing. Additionally, it includes the time schedule and project cost estimates.

Model/Framework

The development of the farming app follows a structured framework to ensure systematic progress and effective results. The chosen framework is the **Agile Development Methodology** combined with elements of the **Model-View-Controller (MVC) design pattern**. This approach allows for iterative development, continuous feedback, and flexible adaptation to changing requirements.

Data Collection Techniques

1. User Requirements Gathering

- **Surveys and Questionnaires:** To gather initial requirements and understand user needs, surveys and questionnaires are distributed to farmers and agronomists. This helps identify key features and functionalities required in the app.
- **Interviews:** Detailed interviews with selected stakeholders provide deeper insights into specific needs, challenges, and expectations.

2. Market Research

- **Literature Review:** Analyzing existing literature on agricultural technology and weather forecasting tools to identify current solutions and gaps.
- **Competitive Analysis:** Reviewing similar apps and platforms to benchmark features and performance.

Data Analysis and Tools
2. Process Analysis

- **Flowcharts and Diagrams:** Creating flowcharts and process diagrams to map out workflows and data flows within the app.
- **Use Case Analysis:** Identifying and documenting use cases to ensure the app meets user requirements and functions as intended.

System Implementation and Testing
5.1. Development Tools

- **Programming Languages:** The app is developed using Java.
- **Development Platforms:** Integrated Development Environments (IDEs) like Android Studio **APIs:**
- Weather data is integrated through third-party APIs (e.g., OpenWeatherMap, Weatherbit).

Time Schedule
The project schedule includes a work breakdown structure outlining key phases:

1. **Research and Planning:** 1 month
2. **Design and Development:** 1month
3. **Testing and Refinement:** 1 month

Project Cost
7.1. Estimated Costs

- **Software and Tools:** Costs for development tools, licenses, and APIs.
- **Testing Costs:** Expenses for testing tools and user testing incentives.
- **Miscellaneous Costs:** Additional costs for marketing, user training, and unforeseen expenses.

7.2. Budget Allocation

A detailed budget breakdown will be provided in the appendix, showing the allocation of resources and estimated costs for each phase of the project.

CHAPTER FOUR: SYSTEM ANALYSIS AND DESIGN

This chapter provides an in-depth analysis and design of the farming app project. It includes a detailed description of how the current system operates, methods for data collection, and requirement definitions and specifications for both the current system and the proposed project. The analysis uses various system modeling tools to illustrate the design and functionality of the app.

Current System Analysis

1. System Description

The current system for managing weather information and agronomic advice often involves disparate tools and platforms. Farmers typically use separate sources for weather forecasts and agronomic consultations, leading to fragmented and less efficient decision-making processes. Key components of the current system include:

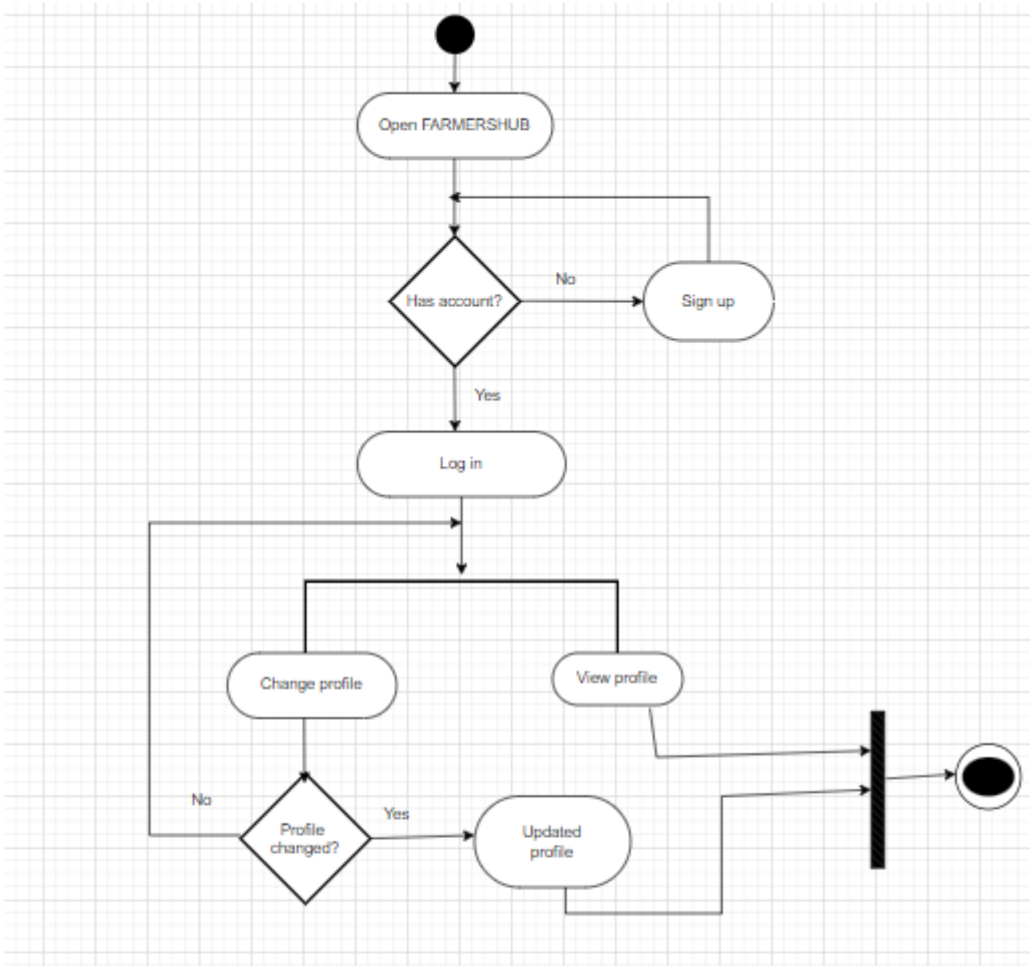
- **Weather Forecasting Tools:** These tools provide weather data but may lack integration with agricultural needs.
- **Agronomist Consultation Services:** Usually available through separate platforms, requiring farmers to seek advice independently.

2. System Analysis Modeling

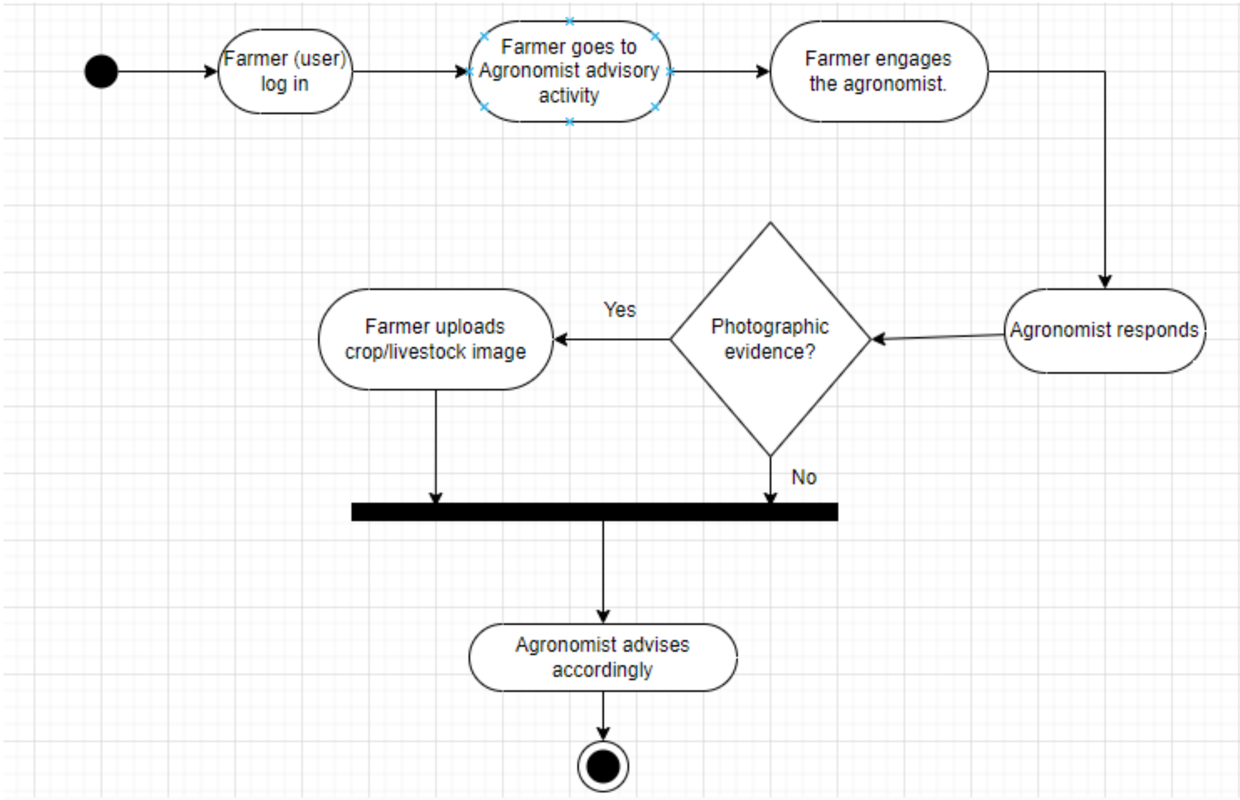
To analyze the current system, several modeling tools are used:

Activity diagram

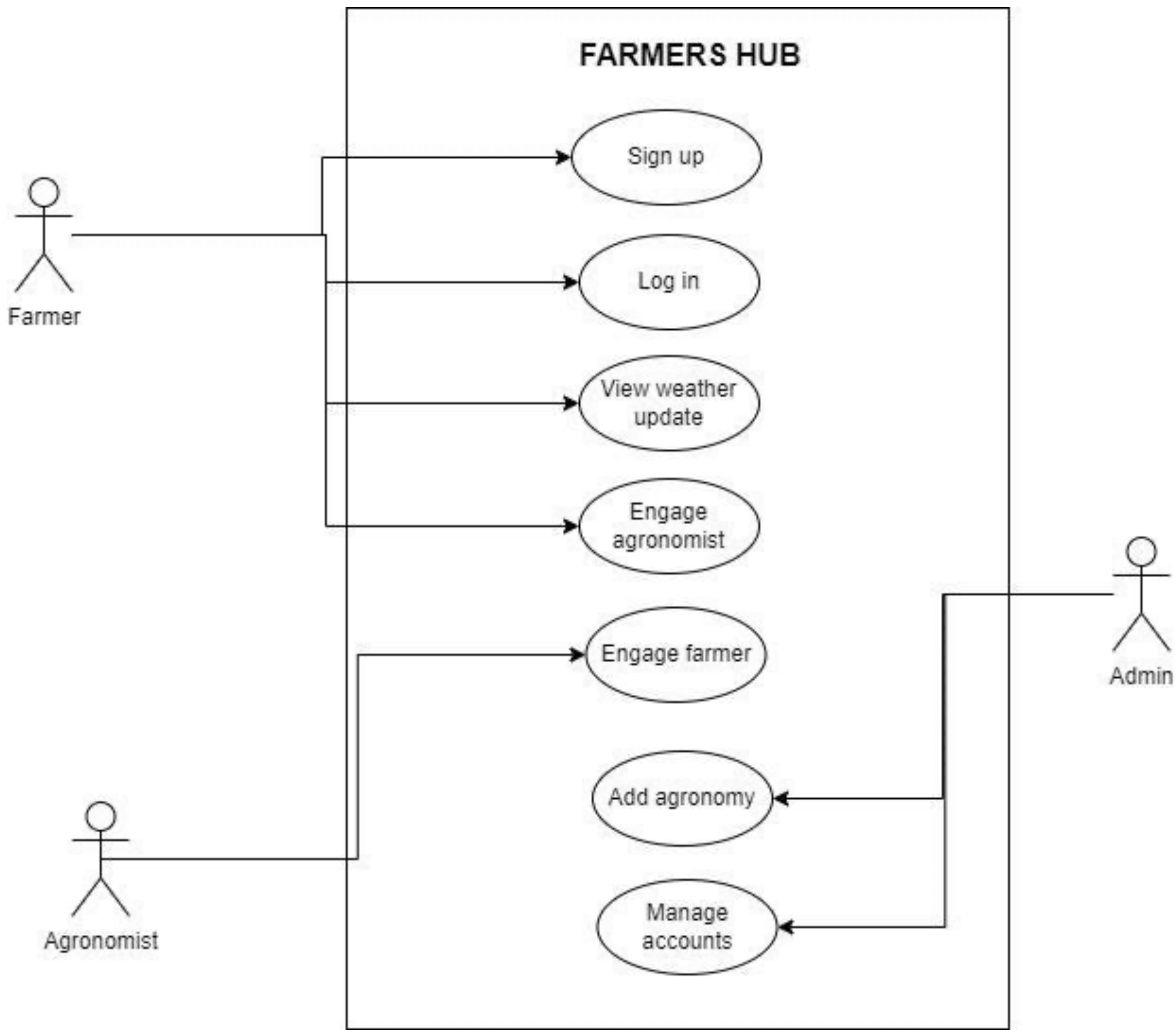
The activity diagram below portrays the control flow of a user updating a profile, from the log in point to the log out point showing the various decision paths that exists while the activity is being executed.



An activity diagram portraying the consultation activity between the farmer and the agronomist in FARMERSHUB.


Farmer (user) log in
Farmer goes to Agronomist advisory activity
Farmer engages the agronomist.
Agronomist responds
Photographic evidence?
Yes
Farmer uploads crop/livestock image
No
Agronomist advises accordingly

Use Cases



The actors in the ‘Farmers hub’ are:

Farmer

Agronomist

System administrator

Use-cases:

Use cases associated with the farmer:

Create account (sign up) – The farmer input his details amongst them; username, password, which he/she will use them as his/her credentials when logging in.

Log in – The farmer uses his/her credentials to log into his/her account in the Farmers hub and securely access his/her personalized dashboard and services.

View weather update – Allows the farmer to check the latest weather updates which entails crucial weather information that is important for planning agricultural activities.

Engage agronomist – The farmer communicates and consults with an agronomist to get expert advice and support for farming activities.

Use cases associated with the agronomist:

Engage farmer – The agronomist offers advices, support and services to the farmers logged into the Farmers hub.

Use cases associated with the System admin:

Manage accounts – Allows the user to manage user accounts to maintain the user database and ensure proper access control and user management.

Data Gathering Methods

1. Surveys and Questionnaires

Surveys and questionnaires were employed to gather user requirements and understand current challenges. This method provided quantitative data on user needs and preferences.

2. Interviews

Interviews with farmers and agronomists offered qualitative insights into specific issues and expectations. Detailed interviews helped identify gaps in existing solutions.

3. Observation

Observations of current system usage provided practical insights into user behavior and interactions with existing tools.

Data Analysis
1. Statistical Analysis

Data from surveys were analyzed using statistical tools to identify common patterns and requirements. Metrics such as user satisfaction and feature importance were assessed.

2. Qualitative Analysis

Interview and observation data were analyzed to extract key themes and insights. Techniques such as coding and thematic analysis were used to interpret qualitative data.

Requirement Definitions and Modeling of the Current System
1. Requirement Definitions

1.1. Functional Requirements

Functional requirements of the current system include:

- **Weather Data Retrieval:** Ability to collect and display weather information from external sources.
- **Consultation Scheduling:** Functionality for arranging consultations with agronomists.

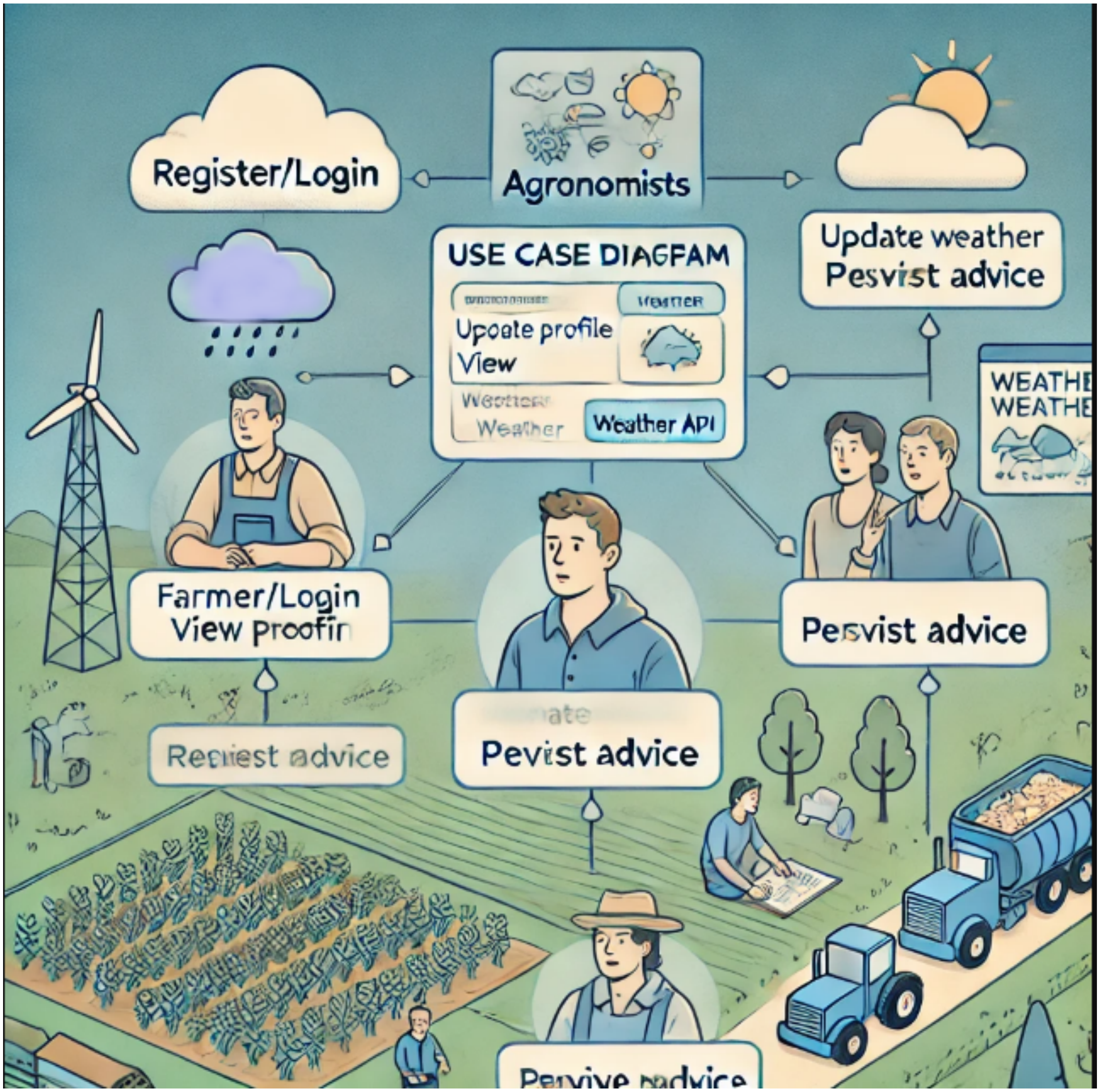
1.2. Non-Functional Requirements

Non-functional requirements include:

- **Performance:** Timely delivery of weather updates and consultation responses.
- **Usability:** User-friendly interfaces for both farmers and agronomists.

4.2. Requirement Modeling

4.2.1. Context Diagrams



Requirement Definitions and Specifications for the Project

1. Functional Requirements

1.1. Weather Update Integration

The app will integrate with weather data APIs to provide real-time weather updates. Key features include:

- **Location-Based Forecasts:** Tailored weather information based on the user's location.

- **Alert Notifications:** Real-time alerts for weather changes that may impact farming activities.

1.2. Agronomist Engagement

Features for engaging with agronomists will include:

- **Consultation Scheduling:** An integrated calendar for booking and managing consultations.
- **Real-Time Messaging:** A chat feature for direct communication with agronomists.

2. Non-Functional Requirements

2.1. Performance and Scalability

- **Response Time:** Ensuring quick load times for weather updates and consultation responses.
- **Scalability:** Ability to accommodate increasing numbers of users and data volume.

2.2. Security and Privacy

- **Data Encryption:** Protecting user data during transmission and storage.
- **Access Controls:** Implementing role-based access for different user types (farmers, agronomists).

2.3. Usability and Accessibility

- **Intuitive Interface:** Easy navigation and interaction for users with varying technical skills.
- **Accessibility Features:** Support for users with disabilities, including screen readers and adjustable text sizes.

CHAPTER FIVE: SYSTEM DESIGN

This chapter provides a comprehensive description of the system design for the farming app project. It covers the overall design of the system, including the architecture, user interface design, and database design at the conceptual, logical, and physical levels. The chapter uses various modeling tools to illustrate the design elements and ensure a robust and effective system.

System Design Overview

1. System Architecture

The farming app is designed using a three-tier architecture that separates concerns into different layers:

- **Presentation Layer (Front-End):** This layer consists of the user interface, where users interact with the app. It includes screens for weather updates, agronomist consultations, and notifications.
- **Application Layer (Back-End):** This layer handles the business logic and processing. It integrates with external APIs for weather data, manages user interactions, and handles data processing.
- **Data Layer (Database):** This layer manages data storage and retrieval. It includes the database where user information, weather data, and consultation records are stored.

2.2. User Interface Design

The user interface (UI) design focuses on creating an intuitive and user-friendly experience. Key UI components include:

- **Dashboard:** Provides an overview of weather updates and upcoming consultations.
- **Weather Forecast Screen:** Displays real-time weather information and forecasts.
- **Notifications Screen:** Shows alerts and updates related to weather and consultations.

Wireframes and mockups of these screens are included in the appendix.

Database Design

Database design is critical to ensuring efficient data management and retrieval. The design process includes conceptual, logical, and physical design phases.

1. Conceptual Design

The conceptual design focuses on identifying the main entities and their relationships without considering how they will be implemented physically.

1.1. Entities and Relationships

- **User:** Represents both farmers and agronomists.
  - Attributes: UserID, Name, Email, Role (Farmer/Agronomist), etc.
- **WeatherData:** Contains weather information relevant to users.
  - Attributes: WeatherID, Location, Temperature, Humidity, ForecastDate, etc.
- **Consultation:** Represents scheduled consultations between users and agronomists.
  - Attributes: ConsultationID, UserID, AgronomistID, ScheduledDate, Status, etc.

Physical Design

The physical design focuses on how the database will be implemented and optimized in a specific database management system (DBMS).

1. Indexing and Optimization

- **Indexes:** Creating indexes on frequently queried columns (e.g., UserID, WeatherID) to improve query performance.
- **Partitioning:** Dividing large tables into smaller, manageable pieces based on criteria such as date ranges or geographical locations.

3.3.2. Storage Considerations

- **Data Storage:** Allocating sufficient storage for user data, weather information, and consultation records.

- **Backup and Recovery:** Implementing backup strategies to ensure data integrity and recovery in case of failures.

3.3.3. Database Management System (DBMS)

The app uses a relational DBMS such as PostgreSQL or MySQL for storing and managing data. The choice of DBMS is based on its scalability, reliability, and support for complex queries.

4. Design Validation and Review

4.1. Design Review

The system design is reviewed through various means:

- **Peer Review:** Engaging team members and stakeholders to review the design and provide feedback.
- **Prototyping:** Developing prototypes or mockups to validate design choices with end-users.

4.2. Validation

Validation involves ensuring that the design meets the specified requirements and performs as expected:

- **Requirement Verification:** Ensuring that the design addresses all functional and non-functional requirements.
- **Performance Testing:** Evaluating the system's performance under different loads to ensure it meets performance criteria.

5. Conclusion

This chapter has detailed the system design for the farming app, including the overall architecture, user interface design, and database design at conceptual, logical, and physical levels. The design ensures that the app is robust, efficient, and user-friendly, addressing the needs of farmers and agronomists effectively.

CHAPTER SIX: SYSTEM IMPLEMENTATION

1. Tools Used for Coding and Testing

1.1. Coding Tools

- **Integrated Development Environments (IDEs):**
  - **Android Studio:** Used for developing the Android version of the app, providing tools for coding, debugging, and emulating the app on various devices.
- **Programming Languages:**
  - **Java:** Utilized for Android app development, known for its robustness and performance.
- **Version Control Systems:**
  - **Git:** Used for source code management, allowing multiple developers to collaborate and maintain version history.

2. System Test Plan

2.1. Test Objectives

The primary objectives of the system test plan are to:

- Verify that the app meets all specified requirements and functions correctly.
- Identify and fix defects before the app is released to users.
- Ensure the app performs well under expected load conditions.

2.2. Test Scope

The test scope includes:

- **Functional Testing:** Ensuring all features work as intended, including weather updates, consultation scheduling, and user notifications.
- **Performance Testing:** Evaluating the app's responsiveness and stability under various load conditions.

- **Usability Testing:** Assessing the user interface and experience to ensure ease of use and accessibility.
- **Security Testing:** Verifying that user data is protected and that the app is resilient to potential security threats.

2.3. Test Phases

- **Unit Testing:** Testing individual components or functions of the app to ensure they operate correctly in isolation.
- **Integration Testing:** Verifying that different component of the app work together as expected.
- **System Testing:** Conducting end-to-end testing of the entire app to ensure all features and functionalities work together.
- **Acceptance Testing:** Performing tests based on user requirements to validate that the app meets user needs and expectations.

Proposed Change-Over Techniques

Parallel Change-Over

- **Description:** Both the old and new systems run concurrently for a period, allowing users to transition gradually to the new system while the old system remains operational.
- **Advantages:** Provides a safety net, as the old system can be used if issues occur with the new system.
- **Disadvantages:** Involves additional costs and complexity due to maintaining two systems simultaneously.

CHAPTER SEVEN: LIMITATIONS, CONCLUSIONS AND RECOMMENDATIONS

1. Limitations

Throughout the development and research process of the farming app, several limitations were encountered:

1.1. Time Constraints

The project faced tight deadlines that restricted the time available for in-depth testing and refinement. Limited time impacted the thoroughness of user acceptance testing and the iterative design process, potentially affecting the final quality and usability of the app.

1.2. Financial Limitations

Budget constraints constrained the ability to access certain advanced tools and resources. This limitation affected the scope of performance and security testing, as well as the potential for incorporating additional features or enhancements that could have improved the app's functionality.

1.3. Data Access and Availability

Obtaining comprehensive and accurate weather data for all intended regions proved challenging. Limited access to certain data sources and variability in data quality impacted the app's ability to deliver consistent and precise weather updates across different geographical locations.

1.4. User Feedback

Some users provided limited feedback due to unavailability or lack of engagement. This lack of comprehensive user input affected the ability to fully address all potential usability issues and incorporate diverse perspectives into the app's design.

2. Conclusion

The development of the farming app has demonstrated significant potential for enhancing agricultural practices by providing timely weather updates and facilitating agronomist consultations. The study has integrated theoretical knowledge from system design and user experience principles with practical implementation, resulting in a functional and user-oriented application.

3. Recommendations

Based on the findings and experiences from the project, the following recommendations are proposed for future improvements:

3.1. Enhancing Data Accuracy and Coverage

- **Expand Data Sources:** Collaborate with additional weather data providers to improve the accuracy and coverage of weather information. Consider integrating satellite data and local meteorological sources for more comprehensive forecasting.
- **Regular Updates:** Implement mechanisms for regularly updating weather data and ensuring its accuracy to maintain the app's reliability.

3.2. Improving User Feedback Mechanisms

- **Enhanced Feedback Channels:** Develop more robust feedback mechanisms within the app, such as surveys and in-app feedback forms, to gather diverse user input and address usability issues more effectively.
- **User Engagement:** Increase user engagement through targeted outreach and educational resources to encourage active participation in providing feedback.

3.3. Expanding Functionalities

- **Additional Features:** Explore the addition of new features based on user feedback, such as crop management tools, pest detection, and localized farming tips.
- **Customization Options:** Introduce customizable settings for users to tailor the app's functionality to their specific needs and preferences.

3.4. Financial and Resource Management

- **Seek Funding Opportunities:** Explore additional funding sources or partnerships to support further development, testing, and enhancement of the app.
- **Leverage Open Source Tools:** Utilize open-source tools and resources to reduce development costs and expand the app's capabilities.

3.5. Continuous Improvement

- **Iterative Development:** Adopt an iterative development approach to continuously refine the app based on user feedback and emerging needs. Regularly review and update the app's features and performance to stay aligned with user expectations and technological advancements.