

Farm Management System



Department of Computer Science

**Sadaqat Rasool
Alina Khan**

**NIM-BSCS-2021-20
NIM-BSCS-2021-42**

Year 2025

Final year project report submitted in partial fulfillment of requirement for degree of Bachelor of
Science in Computer Science

**Namal University,
30-KM, Talagang Road, Mianwali, Pakistan.
www.namal.edu.pk**

DECLARATION

The project report titled **Farm Management System** is submitted in partial fulfillment of the degree of Bachelor of Science in Computer Science, to the Department of Computer Science at Namal University, Mianwali, Pakistan.

It is declared that this is original work done by the team members listed below, under the guidance of our supervisor **Ramzan Shahid Khan Niazi**. No part of this project or its report is plagiarized from anywhere, and any help taken from previous work is cited properly.

No part of the work reported here is submitted in fulfillment of requirement for any other degree/qualification in any institute of learning.

Team Members	University ID	Signatures
Sadaqat Rasool	NIM-BSCS-2021-20	_____
Alina Khan	NIM-BSCS-2021-42	_____

Supervisor

Ramzan Shahid Khan Niazi

Signatures with date

Date: November 19, 2025

Table of Contents

Introduction.....	6
1.1 Purpose.....	6
1.2 Scope	6
1.3 Definitions, Acronyms, and Abbreviations.....	7
1.4 Overview	7
System Description.....	7
Overall Description.....	8
System Environment.....	8
Product Perspective	9
Product Features.....	9
Analysis and Design	10
3.1 Functional Requirements	10
3.2 Non-Functional Requirements.....	11
3.3 System Models.....	11
3.3.1 Data Flow Diagrams (DFD)	11
3.3.2 Use Case Diagrams.....	12
3.3.3 ER Diagram	13
3.3.4 Sequence Diagrams	13
3.3.5 Activity Diagram	17
3.4 External Interface Requirements	17
3.4.1 User Interfaces.....	17
3.4.2 Hardware Interfaces	17
3.4.3 Software Interfaces	17
3.4.4 Communication Interfaces	17
3.5 Constraints and Limitations	18
3.6 Assumptions and Dependencies.....	18
Remaining Work and Timeline	19
4.1 Outstanding Tasks	19
High Priority Tasks.....	19

Medium Priority Tasks	19
Additional Tasks	20
4.2 Status of Each Task	20
4.3 Timeline and Milestones	20
4.4 Summary of Milestones	21
4.5 Conclusion	21

List of Figures

1. *DFD-level-0*
2. *DFD-level-1*
3. *DFD-level-2*
4. *UseCase Diagram*
5. *ER Diagram*
6. *SSD Authentication*
7. *SSD Crop Management*
8. *SSD Livestock Management*
9. *SSD Resource Management*
10. *SSD Chatbot*
11. *SSD Reminders*
12. *SSD Report Generation*
13. *Activity Diagram*

Abstract

The Farm Management System is developed for farmers in order to make their agricultural activities efficient. Conventional farming systems lead to inefficiency, delayed information, and difficulties in managing resources. The project suggests an integrated web-based solution that incorporates six major services: Crop Management, Livestock Management, Resource Management, Alerts and Reminders, Report Generation, and ChatBot for real-time assistance. The system ensures an easy user interface that has no hassle when a farmer uses the system, thereby ensuring ease in using different functionalities. Developing this system also required a front-end user interface robust enough for implementation and, hence, introduced user authentication so that only approved access to the system is granted. Also, ChatBot was introduced on the back end to facilitate quick support to the farmers as well as give instant answers. Even though the ChatBot feature is completely ready, the other five services are only initiated with development work at the backend. In the near future, the backend services should be completed and the ChatBot feature must be enhanced with integration of high-end analytics for the betterment of the farming community in making proper decisions and executing actions. The system is meant to provide the farmer with the tools required for better operations and higher productivity in the production of sustainable agriculture.

Introduction

Agriculture is one of the sectors that offer food and resources to various communities around the world. Nevertheless, most farmers face difficulties when managing their farms. Some of the challenges include tracking crops, animals, and the use of other resources, communicating and making the right decisions within the right timeframe. The Farm Management System, FMS has been developed in a way to eliminate such difficulties since it offers an all-inclusive web-based interface that makes managing a farm simpler. The project has been chosen in order to meet the ever-increasing demand of efficient farming techniques in this competitive market, and by introducing technology, FMS will enable the farmers to access tools that can increase their productivity, resource utilization, and better decision-making capabilities. It includes six service areas in total: Crop Management, Livestock Management, Resource Management, Alerts and Reminders, Report Generation, and a Chatbot for instant support. The main objective is to create a user-friendly platform for the farmer to manage his operations better. This will include a workable front-end interface, implementation of user authentication to make the application secure, and including a Chatbot to provide instant support. Overall, the success of this project can be evaluated in terms of the functionality of services provided, the level of user satisfaction, and the overall efficiency of farm management.

1.1 Purpose

This SRS document will outline the requirements and specifications of the Farm Management System. It is to guide developers, stakeholders, and users to have a common understanding of what the project entails and what functionalities it will carry out. The SRS is to find poor farming management by setting up an ordered approach toward the design of a system with meeting all the requirements a farmer needs to have. Thus, a distinct definition of the system requirements assures that the resulting product is indeed viable, usable, and properly directed toward achieving the main purposes of the project.

1.2 Scope

In terms of scope, the Farm Management System would create all key functionalities to help the farmer manage the different aspects of farming. The critical goals of this project are: Crop Management Livestock Management Resource Management Alerts and Reminders Report Generation A real-time help facility in the form of a chatbot. It's built especially for farmers and eliminates any administrative features. The industrial applications of this project are wide-ranging, as it can be used by the individual farmer, agricultural cooperatives, and farming organizations to boost productivity and make operations more efficient.

1.3 Definitions, Acronyms, and Abbreviations

1. **API:** Application Programming Interface - A set of rules that allows different software applications to communicate with each other.
2. **Chatbot:** A computer program designed to simulate conversation with human users, especially over the Internet.
3. **FMS:** Farm Management System - A web-based platform designed to assist farmers in managing their agricultural activities.
4. **UI:** User Interface - The means by which a user interacts with a computer system or application.
5. **DFD:** Data Flow Diagram - A visual representation of the flow of data within a system, illustrating how data moves between processes, data stores, and external entities.
6. **SSD:** System Sequence Diagram - A type of interaction diagram that shows how objects interact in a particular scenario of a use case, detailing the sequence of messages exchanged between the user and the system.

1.4 Overview

This document has been designed with a comprehensive aim towards the understanding of the Farm Management System. From here, the entire overview will then lead the discussion in Chapter 2. Chapter 3 then presents detailed requirements: a mixture of both functional and non-functional requirements regarding the system in question. Chapter 4 will detail the system features. Chapter 5 will include appendices with more supporting information. The chapters are arranged to build one upon another to provide a coherent and logical flow of information that ends in a comprehensive understanding of the project and its goals.

System Description

This agricultural sector has continuously relied on technology to maximize its productivity and efficiency. The number of studies has emphasized the significance of the farm management systems in agriculture as challenges farmers face today are on the aspect of resource allocation, crop monitoring, and livestock management. For example, Nizi Solutions (n.d.) wrote on the dairy farm management software that helps in optimizing the health and productivity of animals on the farm, reflecting how technology may empower the smooth running of operations and facilitate better decision-making processes. Also, AgriWebb (n.d.) offers an all-rounded farm management app for every critical function: from livestock and crop management to record-keeping, displaying the growing trend of digital solutions in agriculture. Very recent literature also hints at the need for user-friendly interfaces for farm management software. GR Farms in a recent paper argues that intuitive design helps in higher user interaction and adoption for better management practices among farmers. Of course, the goals of this present study also include developing an easily used platform with the aim of simplifying tasks in farm

management.

Access to real-time data is what is emphasized on the need by Bushel Farm (n.d.), as this ensures that the management can make decisions timely. Alerts and reminders are among features which the current project should have for the ability of farmers to react in a timely manner in respect of the changes happening on the farm. This project, therefore, adds on to this insight by proposing a comprehensive Farm Management System. It encompasses six main services, which comprise Crop Management, Livestock Management, Resource Management, Alerts and Reminders, Report Generation, and a ChatBot for real-time assistance. Using the available literature, the gaps that the current solutions face will be bridged with an integrated approach toward farm management.

Overall Description

- **User Characteristics**

The primary users of the Farm Management System are farmers who may have varying levels of technical expertise. The system is designed to be intuitive and accessible, ensuring that even those with limited experience in technology can effectively utilize its features. Additionally, the system may be used by agricultural consultants and extension workers who assist farmers in managing their operations.

- **Constraints**

The development of the system is subject to several constraints, including budget limitations, time constraints for project completion, and the need for compatibility with existing agricultural practices. Furthermore, the system must comply with data protection regulations to ensure the privacy and security of user information.

- **Assumptions**

It is assumed that users will have access to the internet and basic computing devices, such as smartphones or computers, to utilize the system effectively. Additionally, it is assumed that users will be willing to adopt new technologies to improve their farming practices.

- **Dependencies**

The system's functionality may depend on third-party services, such as weather data providers and agricultural databases, to deliver accurate and timely information to users. The successful integration of these services is crucial for the system's overall effectiveness.

System Environment

- **Hardware Requirements**

Server: A reliable server with sufficient processing power and storage capacity to host the application and manage user data.

User Devices: Smartphones, tablets, or computers with internet access for users to interact with the system.

- **Software Requirements**

Operating System: The server should run on a stable operating system, such as Linux or Windows Server.

Web Technologies: The application will be developed using the MERN stack, which includes:

- **MongoDB:** A NoSQL database for storing user data and application information.
- **Express.js:** A web application framework for Node.js to build the backend services.
- **React:** A JavaScript library for building the user interface.
- **Node.js:** A JavaScript runtime for executing server-side code.

- **Operating Environment**

The Farm Management System will operate as a web-based application, accessible through standard web browsers. This ensures that users can access the system from various devices without the need for extensive installations.

Product Perspective

The Farm Management System is designed to complement existing agricultural practices and technologies. Unlike traditional farm management methods that rely on manual record-keeping and fragmented information sources, this system offers an integrated solution that centralizes data and provides real-time insights.

Relations to Existing Systems

- **Comparison with Other Solutions:** While other farm management systems exist, many lack the comprehensive features and user-friendly design that this project aims to provide. The integration of a Chatbot for real-time assistance sets this system apart, offering users immediate support and guidance.
- **Interoperability:** The system is designed to be interoperable with other agricultural tools and platforms, allowing for seamless data exchange and enhancing the overall user experience.

Product Features

1. **Crop Management:**

Users can register and monitor their crops, track growth stages, and schedule activities such as planting and harvesting.

2. **Livestock Management:**

This feature allows users to manage livestock health, feeding schedules, and breeding records, ensuring optimal care and productivity.

3. Resource Management:

Users can track and manage resources such as seeds, fertilizers, and equipment, helping to optimize resource allocation.

4. Alerts and Reminders:

The system provides timely alerts and reminders for critical tasks, ensuring that users do not miss important activities such as irrigation, fertilization, or livestock vaccinations.

5. Report Generation:

Users can generate detailed reports on crop yields, livestock performance, and resource usage, facilitating informed decision-making and strategic planning.

6. Chatbot for Real-Time Assistance:

The integrated Chatbot offers users immediate support, answering queries and providing guidance on best practices, thereby enhancing user engagement and satisfaction.

By incorporating these features, the Farm Management System aims to provide a holistic approach to agricultural management, ultimately improving productivity and sustainability in farming practices.

Analysis and Design

This chapter outlines the analysis and design of the Farm Management System, detailing the software process model, functional and non-functional requirements, system models, and external interface requirements. The methodologies and diagrams presented here provide a comprehensive understanding of the system's architecture and functionality, ensuring that the project can be re-implemented based on the information provided.

3.1 Functional Requirements

The functional requirements define the specific behaviors and functionalities that the Farm Management System must support. The following list outlines these requirements:

- The system allows users to log in securely using a username and password.
- The system shall enable users to register and manage their crops, including tracking growth stages and scheduling activities.
- The system shall allow users to manage livestock, including health records, feeding schedules, and breeding information.
- The system shall provide resource management features to track and manage inputs such as seeds, fertilizers, and equipment.
- The system shall generate alerts and reminders for critical tasks, such as irrigation and fertilization.

- The system shall allow users to generate reports on crop yields, livestock performance, and resource usage.
- The system shall allow users to use the ChatBot feature to provide real-time assistance and answer user queries.

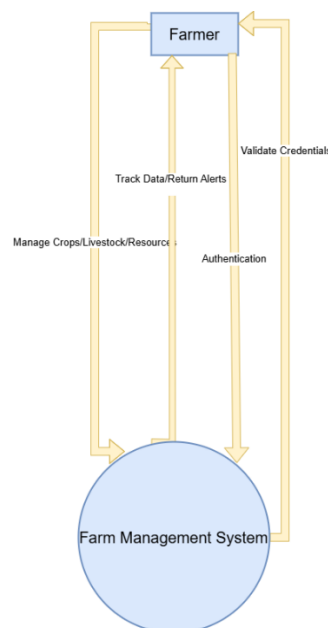
3.2 Non-Functional Requirements

Non-functional requirements specify the quality attributes of the system. The following requirements are essential for the Farm Management System:

- **Performance:** The system shall respond to user requests within 2 seconds under normal load conditions.
- **Security:** The system shall implement secure authentication and authorization mechanisms to protect user data.
- **Scalability:** The system should be designed to accommodate an increasing number of users and data without performance degradation.
- **Usability:** The user interface shall be intuitive and easy to navigate, allowing users to complete tasks with minimal training.

3.3 System Models

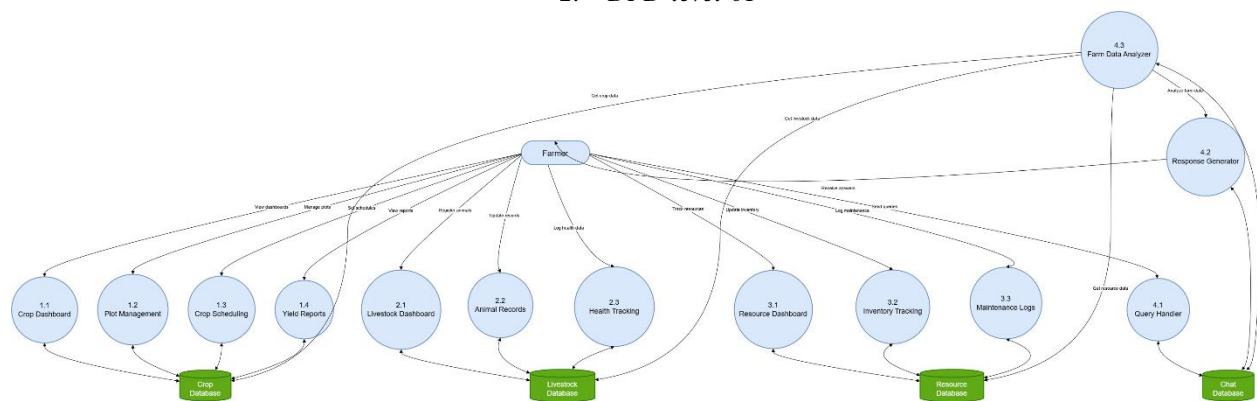
3.3.1 Data Flow Diagrams (DFD)



1. DFD-level-0

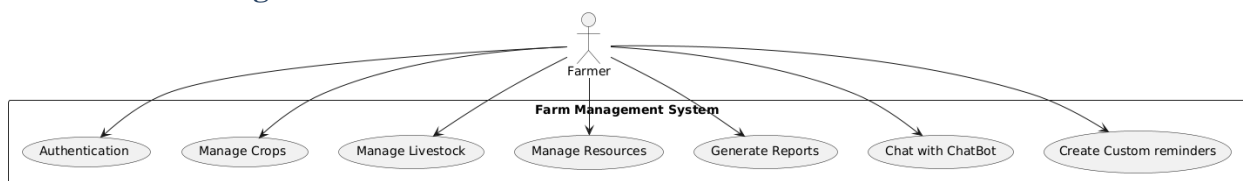


2. DFD-level-01



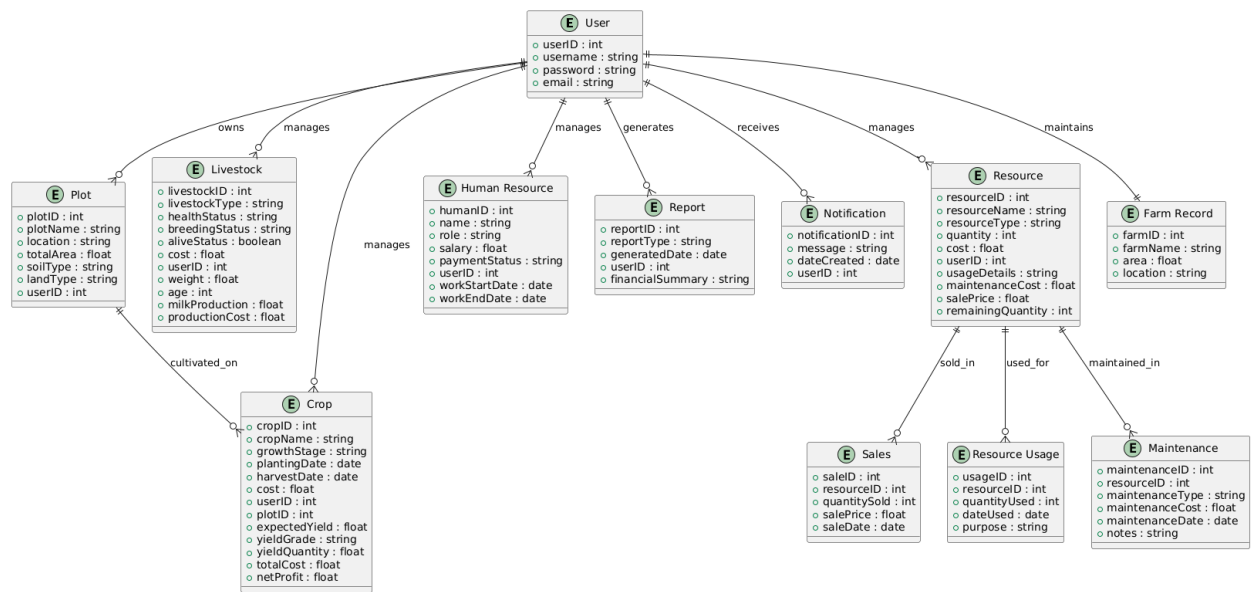
3. DFD-level-02

3.3.2 Use Case Diagrams



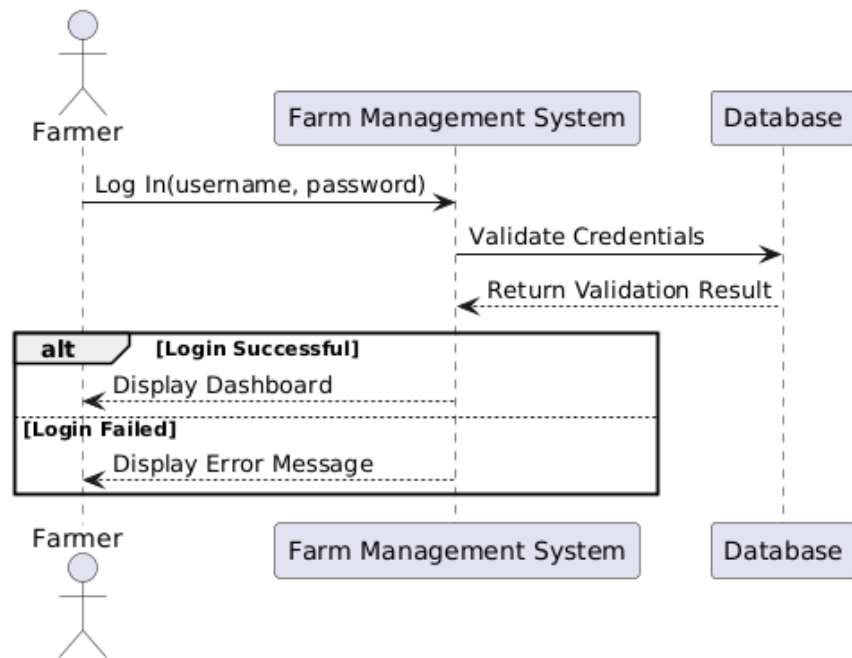
4. UseCase Diagram

3.3.3 ER Diagram

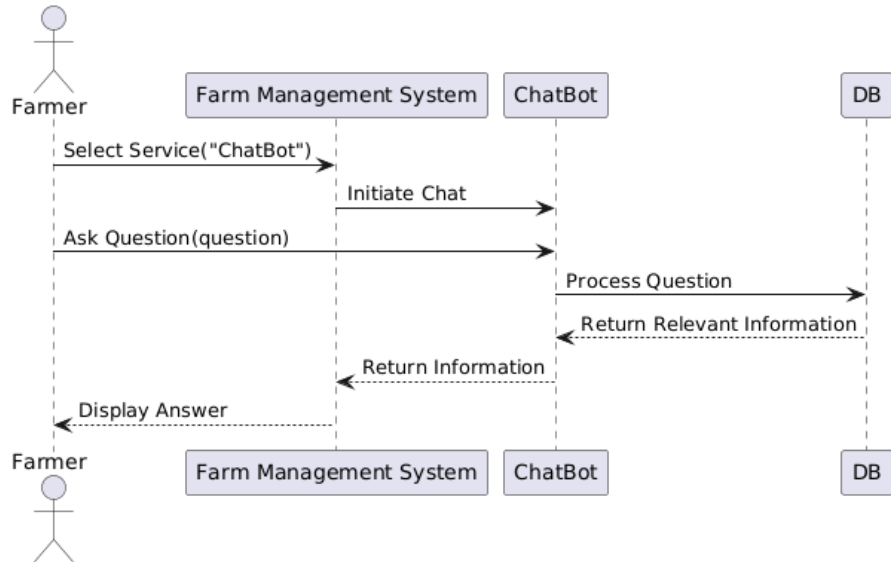


1. ER Diagram

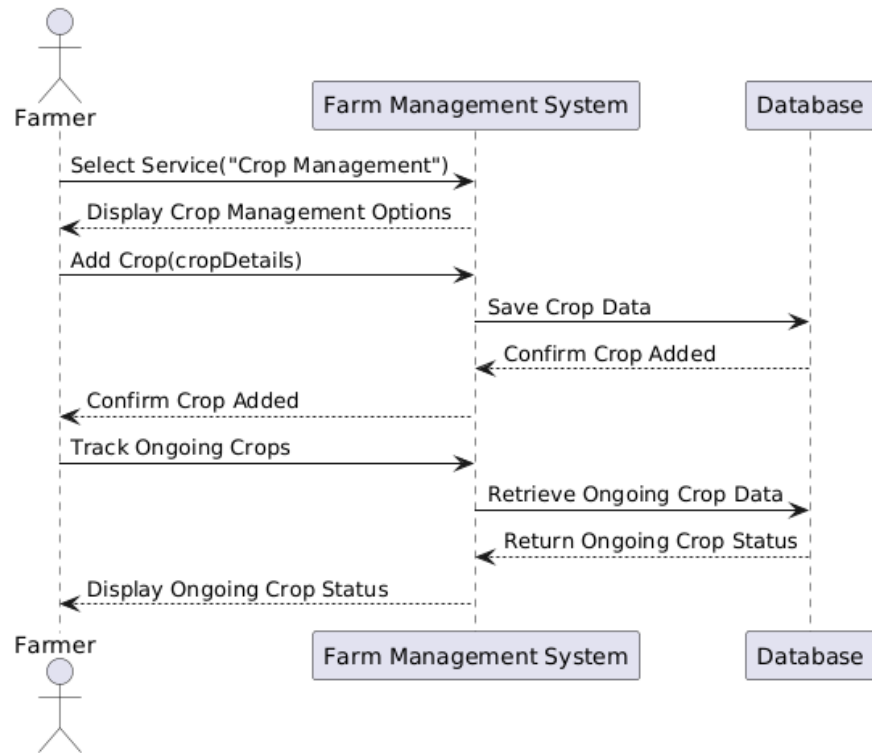
3.3.4 Sequence Diagrams



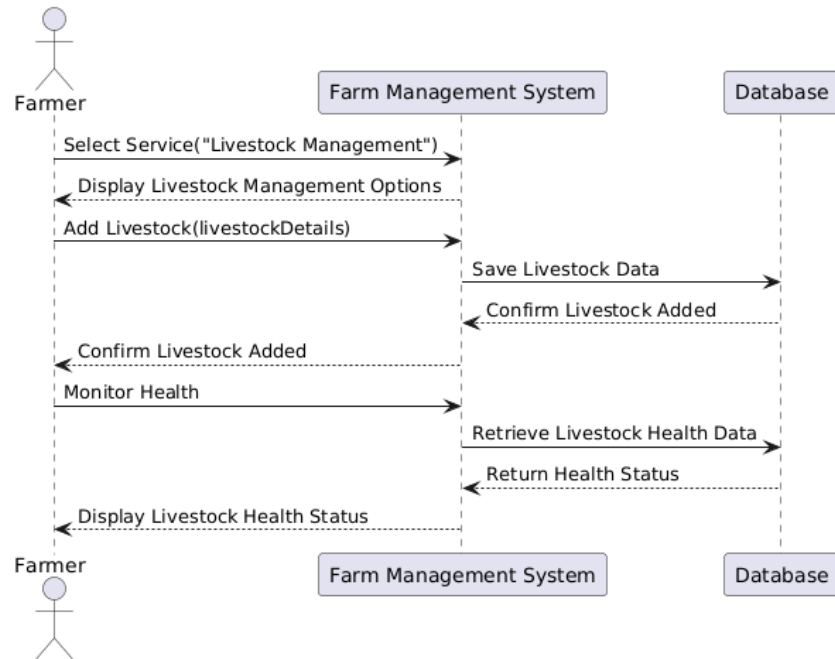
2. SSD-Authentication



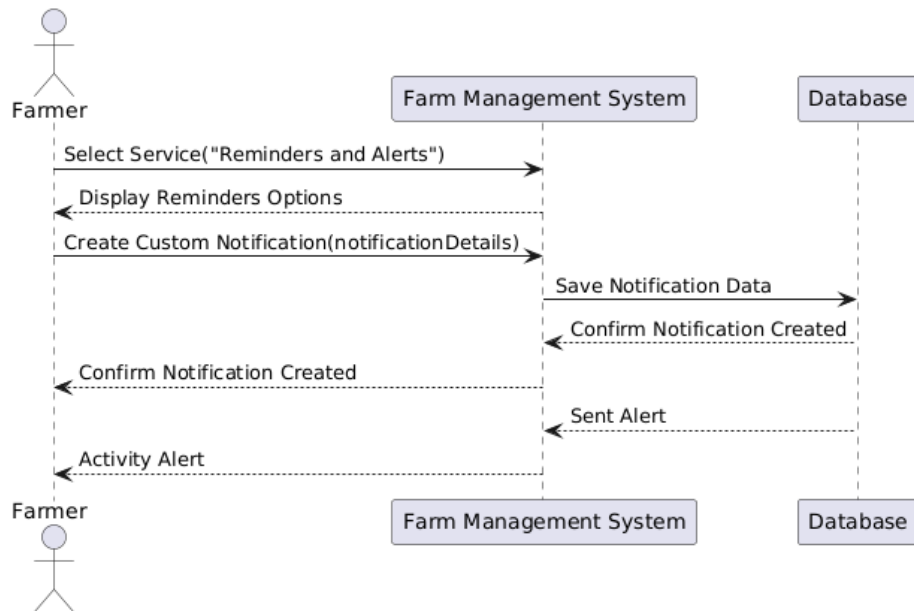
3. SSD-Chatbot



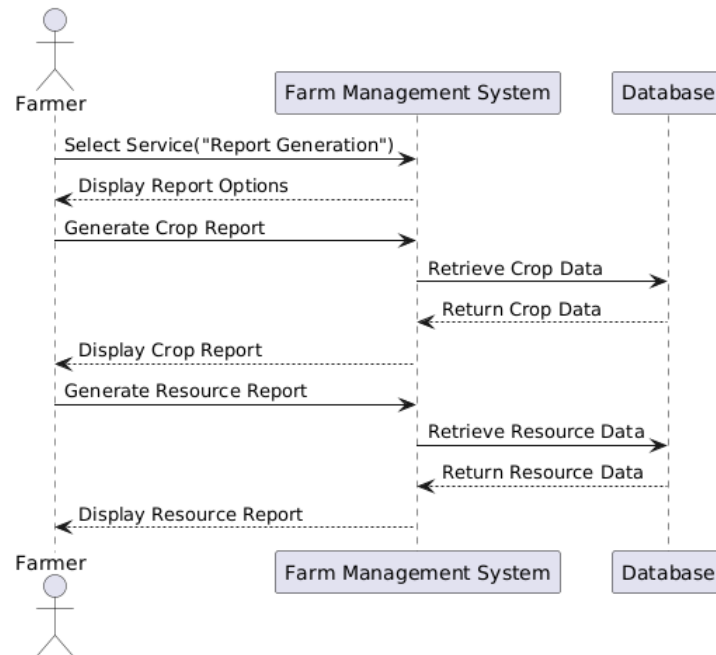
4. SSD-Crop Management



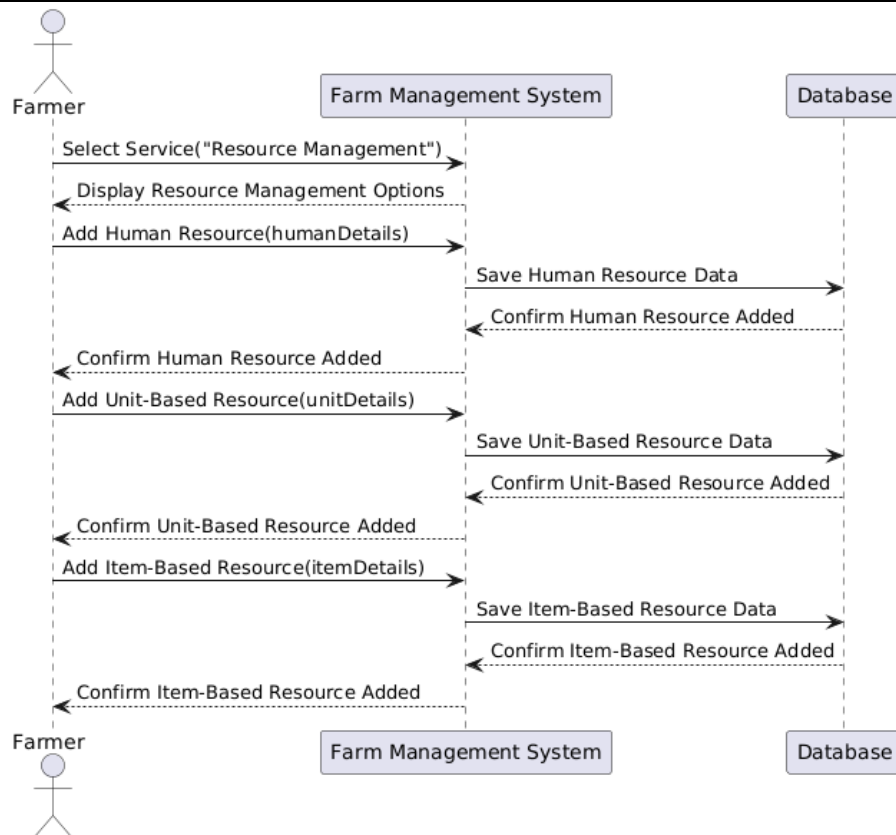
5. SSD-Livestock Management



6. SSD-Reminders

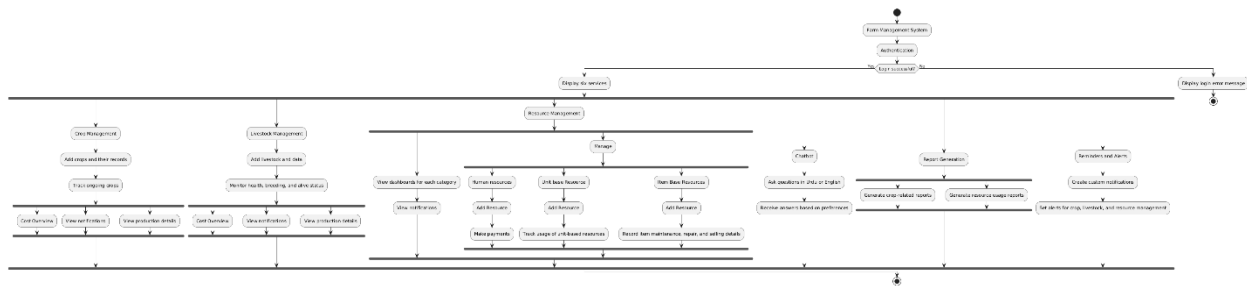


7. SSD-Report Generation



8. SSD-Resource Management

3.3.5 Activity Diagram



13. Activity Diagram

3.4 External Interface Requirements

3.4.1 User Interfaces

The user interface of the Farm Management System will be designed with usability in mind. Key features include:

- **Dashboard:** A central hub for users to access all functionalities.
- **Forms:** Simple forms for entering and updating crop and livestock information.
- **Reports:** A dedicated section for generating and viewing reports.

3.4.2 Hardware Interfaces

The system will require the following hardware components:

- **Server:** A dedicated server with at least 16 GB of RAM and 500 GB of SSD storage to host the application.
- **User Devices:** Smartphones, tablets, or computers with internet access.

3.4.3 Software Interfaces

The following software components will be utilized:

- **APIs:** Integration with third-party APIs for weather data and agricultural information.
- **Libraries:** Use of libraries such as Axios for API calls and Redux for state management in React.

3.4.4 Communication Interfaces

The system will employ the following communication protocols:

- **HTTP/HTTPS:** For secure communication between the client and server.

3.5 Constraints and Limitations

The development and implementation of the Farm Management System are subject to several constraints:

- **Cost:** Budget limitations may restrict the scope of features and technologies used.
- **Time:** The project must be completed within a specified timeframe, which may impact on the depth of testing and refinement.
- **Technological Barriers:** Variability in users' access to technology and internet connectivity may limit the system's effectiveness in certain regions.

3.6 Assumptions and Dependencies

The following assumptions and dependencies are critical to the successful implementation of the system:

- **Assumption:** Users will have stable internet connectivity to access the web-based application.
- **Dependency:** The system relies on third-party services for weather data and agricultural databases, which are essential for providing accurate information to users.
- **Assumption:** Users will possess basic computing skills to navigate the system effectively.
- **Dependency:** The performance of the system is dependent on the server's capacity and the efficiency of the underlying database management system.

Remaining Work and Timeline

This chapter outlines the remaining tasks necessary for the completion of the Farm Management System project. It provides a clear roadmap for the outstanding work, including the status of each task and a timeline for their completion. The focus is on ensuring that all functionalities are implemented effectively before the project deadline.

4.1 Outstanding Tasks

High Priority Tasks

1. **Implement Backend for Crop Management**
 - **Description:** Develop backend services to manage crop data, including CRUD (Create, Read, Update, Delete) operations.
 - **Status:** In progress
 - **Expected Completion:** February 15, 2025
2. **Implement Backend for Livestock Management**
 - **Description:** Create backend functionalities for managing livestock data, including health records and feeding schedules.
 - **Status:** Pending
 - **Expected Completion:** February 28, 2025
3. **Implement Backend for Resource Management**
 - **Description:** Develop backend services to track and manage agricultural resources such as seeds, fertilizers, and equipment.
 - **Status:** Pending
 - **Expected Completion:** March 15, 2025

Medium Priority Tasks

1. **Implement Backend for Report Generation**
 - **Description:** Create functionalities to generate reports based on user data, including crop yields and livestock performance.
 - **Status:** Pending
 - **Expected Completion:** March 31, 2025
2. **Implement Backend for Alerts and Reminders**
 - **Description:** Develop backend logic to send alerts and reminders for critical farming tasks.
 - **Status:** Pending
 - **Expected Completion:** April 15, 2025

Additional Tasks

1. **Integrate Urdu Language Support**
 - **Description:** Implement localization features to support the Urdu language in the user interface and documentation.
 - **Status:** Pending
 - **Expected Completion:** April 30, 2025

4.2 Status of Each Task

Task	Status	Expected Completion
Implement Backend for Crop Management	In progress	February 15, 2025
Implement Backend for Livestock Management	Pending	February 28, 2025
Implement Backend for Resource Management	Pending	March 15, 2025
Implement Backend for Report Generation	Pending	March 31, 2025
Implement Backend for Alerts and Reminders	Pending	April 15, 2025
Integrate Urdu Language Support	Pending	April 30, 2025

4.3 Timeline and Milestones

Milestone	Expected Completion
Completion of Crop Management Backend	February 15, 2025
Completion of Livestock Management Backend	February 28, 2025
Completion of Resource Management Backend	March 15, 2025
Completion of Report Generation Backend	March 31, 2025
Completion of Alerts and Reminders Backend	April 15, 2025
Completion of Urdu Language Integration	April 30, 2025

4.4 Summary of Milestones

- **February 15, 2025:** Completion of the backend for Crop Management.
- **February 28, 2025:** Completion of the backend for Livestock Management.
- **March 15, 2025:** Completion of the backend for Resource Management.
- **March 31, 2025:** Completion of the backend for Report Generation.
- **April 15, 2025:** Completion of the backend for Alerts and Reminders.
- **April 30, 2025:** Completion of Urdu Language Support.

4.5 Conclusion

This chapter has provided a detailed overview of the remaining work required to complete the Farm Management System project. By clearly outlining outstanding tasks, their current status, and a structured timeline, this roadmap ensures that the project remains on track for completion by the end of April 2025. The focus on implementing backend functionalities and integrating Urdu language support will enhance the system's usability and accessibility for a broader user base.

References

1. Nizi Solutions, “Dairy Farm & Animal Management Software,” [Online]. Available: <https://www.nizisolutions.com/dairy-farm-animal-management-software/>. [Accessed: Jan. 18, 2025].
2. GR Farms, “Agricultural Products and Services,” [Online]. Available: <https://grfarms.com.pk/our-products/>. [Accessed: Jan. 18, 2025].
3. AgriWebb, “Farm Management App for Livestock and Crop Management,” [Online]. Available: <https://www.agriwebb.com> [Accessed: Jan. 18, 2025].
4. Bushel Farm, “Farm Management Software Solutions,” [Online]. Available: <https://bushelfarm.com/> [Accessed: Jan. 18, 2025].