MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

Object Oriented Programming

Mini Project Report on

Implementation of Cattle Management System using Java & JavaFX

SUBMITTED BY

Surya Lakavalli	23	230905193
Shashank Shettigar	7	230905045
Abhuday Verma	52	230905418
Shrey Jain	53	230905422
Arnav Kataria	5	230905023

Section B

Under the Guidance of:

Dr. T Sujithra
Department of Computer Science and Engineering
Manipal Institute of Technology, Manipal, Karnataka – 576104

October 2024

Table of Contents

1. Introduction
2. Problem statement
3. Implementation details
4. Results (Output screenshots)
5. Conclusion
6. Individual contributions (with photo)
7. References

Introduction

The Cattle Management System we have designed is a comprehensive solution to the many problems that todays farmers face like maintaining accurate records, processing insurance claims efficiently and ensuring data integrity. It is is a comprehensive software solution designed to simplify and streamline the management of cattle-related operations for farmers, insurance companies, and administrators.

Built using Java and JavaFX, this system offers a modern and intuitive user interface, providing users with essential features such as cattle registration, insurance policy management, premium calculation, claim filing, and ownership tracking. Furthermore, using JAVAFX we provide a modern and intuitive user interface

We use techniques like SHA-256 hashing to detect unauthorized modifications and encryption methods to secure data transmissions. By implementing these features, the system ensures that sensitive information, such as ownership and financial details, is protected from tampering and unauthorized access.

With a focus on both user convenience and data reliability, this system stands as a significant innovation in the field of agricultural technology, demonstrating how modern software solutions can transform traditional livestock management practices into a more secure, efficient, and scalable framework.

Problem Statement

1. The objective of this project is to design and implement a Cattle Management System (CMS) using basic Java and JavaFX to manage cattle registration, insurance claims, and ensure data integrity during the process. This system will enable farmers to register cattle, apply for insurance claims in case of cattle loss or illness, and track the status of their claims. The system will also focus on ensuring data integrity, preventing unauthorized modifications, and guaranteeing secure transactions.

System Features:

1. Farmer Registration and Login:

o Farmers can register by providing personal details (e.g., name, farm location) and will be authenticated upon login.

o Post-login, farmers can manage their cattle, view insurance policies, and file insurance claims.

2. Cattle Registration:

o Farmers can register cattle with details such as breed, age, weight, vaccination records, and insurance policy information.

o Each cattle is assigned a unique identification number (UID).

o Data integrity will be ensured by generating a checksum for each cattle record to prevent unauthorized alterations.

o A JavaFX form will be used to collect and display cattle information.

3. Insurance Policies and Premium Management:

o Farmers can view available insurance policies for their cattle.

o The system will calculate premiums based on the cattle's details (e.g., breed, age, health conditions).

o Farmers can pay premiums through the system and keep track of their policy statuses using JavaFX forms.

4. Insurance Claim Process:

o In the event of cattle loss, illness, or death, farmers can file an insurance claim through the system by submitting required documentation.

o The system will validate the claim, ensuring all details are accurate and no unauthorized changes have been made to the cattle records.

o The system will use JavaFX to provide an easy-to-use interface for submitting claims, tracking claim statuses, and viewing policy details.

5. Claim Review and Approval Process:

o Administrators (admins) review insurance claims, verify documentation, and approve or reject claims (automated).

o Admins can view all submitted claims and validate details against system records using JavaFX for user interaction.

o Once a claim is approved, the system will process the insurance payout and ensure data integrity through checksums.

6. Cattle Ownership and Transfer Tracking:

o Farmers can track cattle ownership, including transfers or changes to cattle details.

o Any update to cattle records will generate a new checksum to maintain data integrity.

o Display all cattle owned by a farmer using a list or table view in JavaFX.

7. Audit Logging for Insurance Claims:

o The system will maintain an audit log of all transactions related to cattle registration, insurance claims, and premium payments.
o The audit log will store the checksum of each transaction to detect any data tampering, ensuring the integrity of the insurance claims process.
o Admins can view the logs to monitor activity and ensure compliance.

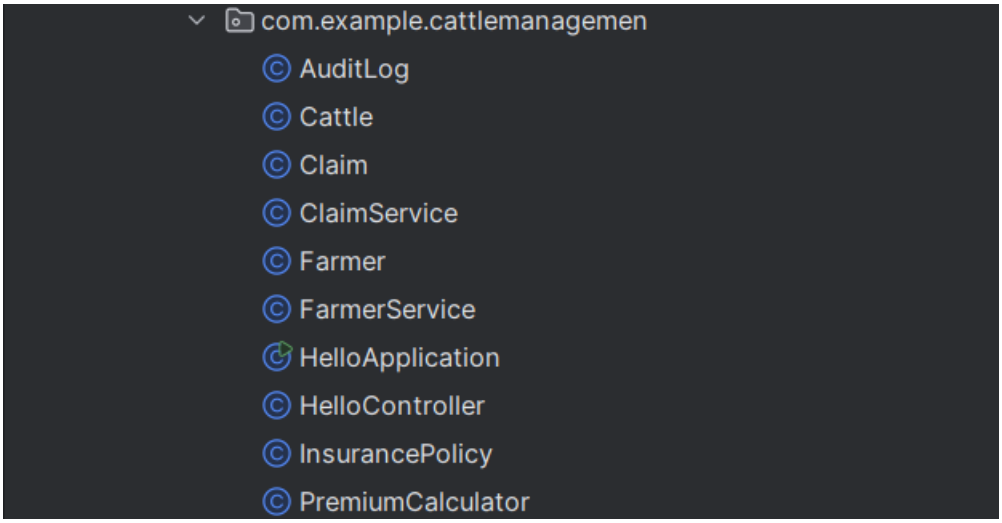
Research Focus:

1. Data Integrity and Security:
o Investigate methods for ensuring data integrity using checksums or hashing algorithms (e.g., SHA-256) to detect unauthorized modifications to cattle records and insurance claims (any encryption decryption algorithm).
o Explore encryption techniques to secure data transmissions between the farmer, insurance providers, and admins.
2. Insurance Risk and Premium Calculation:
o Develop a risk-based approach for calculating insurance premiums based on factors like cattle breed, age, vaccination records, and health conditions.
3. Audit Log:
o Design an immutable audit log where every action related to cattle management and insurance processing is logged with a checksum to guarantee the validity of records.

Implementation Details

The "Cattle Management System" is designed to manage cattle ownership, insurance, and claim processing while ensuring data integrity. The system uses Java and JavaFX for its graphical user interface and provides functionality for farmer registration, cattle registration, insurance management, and claim processing, we have used IntelliJ community edition to run the program. Data integrity is achieved using checksum verification, and data security is enhanced using hashing algorithms like SHA-256 and the creation of an audit log.

The 'cattlemanagemen' project consists of 10 classes along with the main class 'HelloApplication' through which the code is run.



Here is a brief overview of what each class contains:

 Audit log Class: The AuditLog class in the CattleManagement project is responsible for maintaining a record of actions and events within the system. It uses a static list, logEntries, to store log messages. This class is essential for tracking and auditing system activities, ensuring transparency and traceability of operations.

 Cattle Class: The Cattle class in the CattleManagement project models individual cattle and manages their insurance and claims. It includes attributes like id, breed, age, weight, vaccinationRecords, owner, and a list of claims. The class maintains data integrity through a checksum generated using the SHA-256 hashing algorithm.

 Farmer Class: Represents a farmer and contains fields like name, contact details, and a list of owned cattle.

 Claim Class: The Claim class in the CattleManagement project represents an insurance claim filed for a specific piece of cattle. It contains attributes like claimId (a unique identifier for the claim), cattle (a reference to the associated Cattle object), and status (the current status of the claim, initially set to "Pending").

 ClaimService Class: is responsible for managing insurance claims. It maintains a HashMap called claims, where each claim is stored with a unique claimId as the key. This class handles the creation and retrieval of insurance claims, providing a centralized service for managing claims in the system.

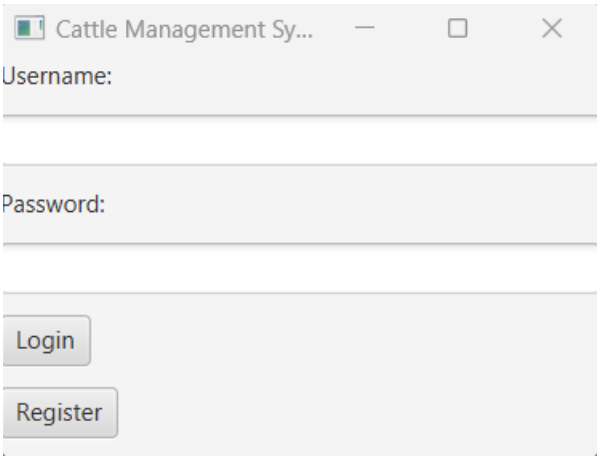
 FarmerService: The FarmerService class in the CattleManagement project manages farmer registration, authentication, and retrieval. It uses a static HashMap named farmers to store Farmer objects, with the farmer's username as the key. This class provides essential services for managing farmer accounts, including registration, login, and farmer lookup.

 InsurancePolicy Class: Holds the values of the insurance policy that is available to the farmers.

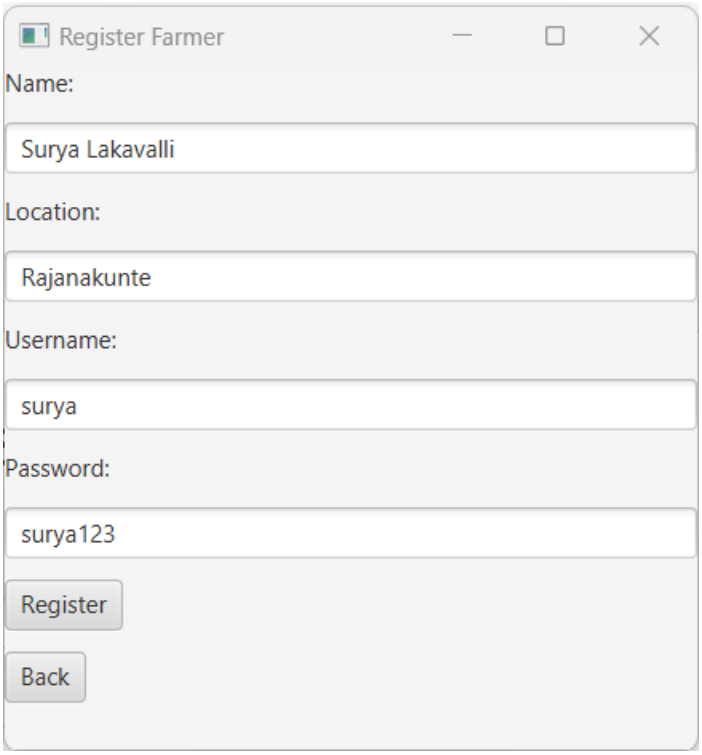
 Premium Calculator: It calculates the amount that the farmer has to pay for insuring their cattle. It uses the formula weight*age*100 dollars.

 Premium Calculator: The HelloApplication class is the main entry point for the JavaFX application in the Cattle Management System project. It extends the Application class and overrides the start(Stage primaryStage) method, where it initializes the primary user interface (UI). Within this method, various UI components are created, including labels for prompts, text fields for user input, and buttons for actions such as logging in and registering. The createLoginScreen() method is likely responsible for constructing the login interface, while methods like handleLogin() and handleRegister() define the event handling logic for button clicks, managing user interactions by validating input and responding to actions. The UI is organized using layout containers, and a Scene is created and set on the primary stage with a specific title and dimensions. Additionally, CSS styles are applied for visual enhancement, ensuring a cohesive design. The class may also include navigation methods to switch between different screens or views, such as transitioning from the login screen to the registration screen. Finally, a static main method is provided to launch the JavaFX application by calling launch(args), initiating the application's lifecycle and displaying the UI to the user. Overall, the HelloApplication class plays a crucial role in presenting the application's functionality, managing user interactions, and ensuring a smooth user experience.

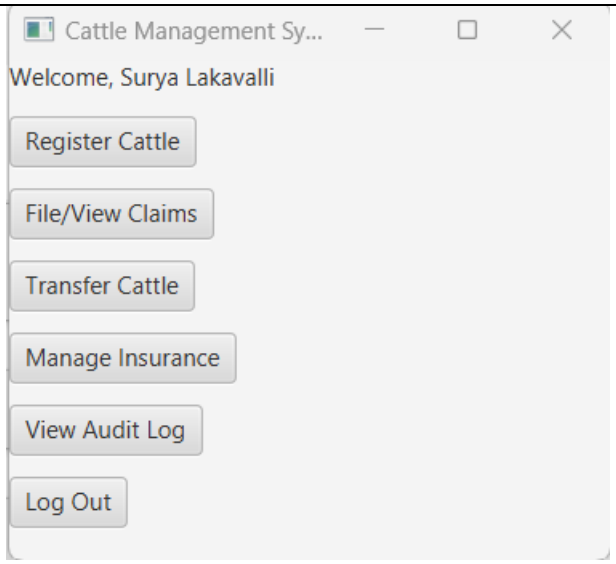
Screenshots of result



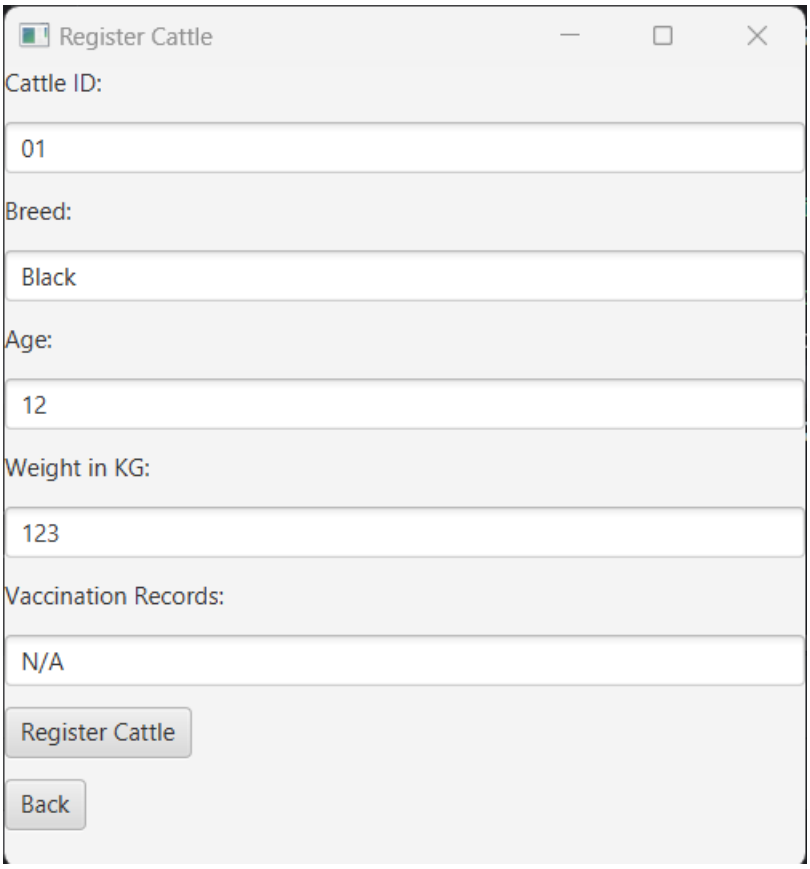
The login Page



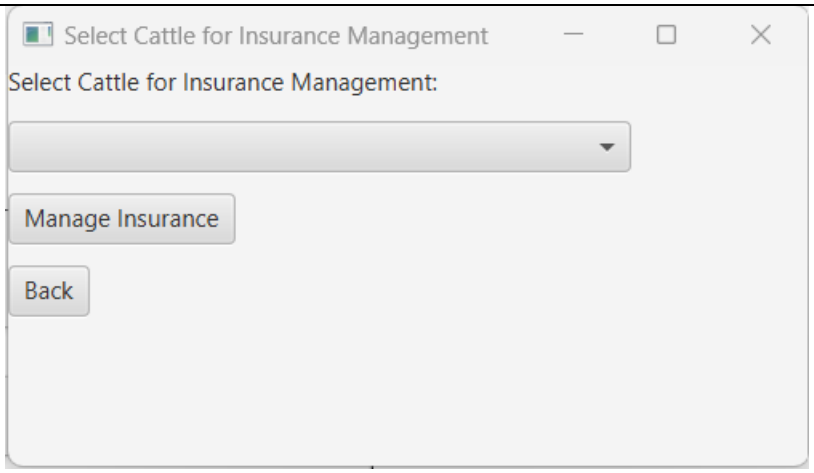
Farmer Registration Page



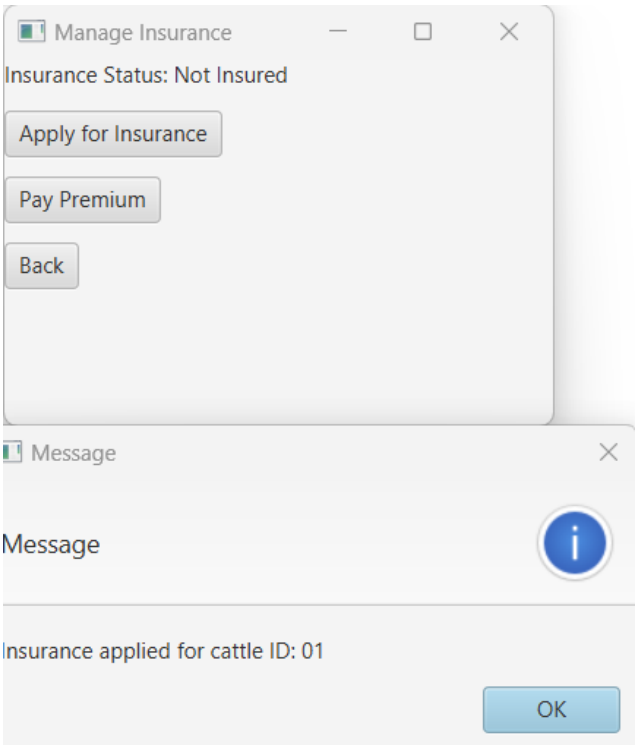
Main Menu



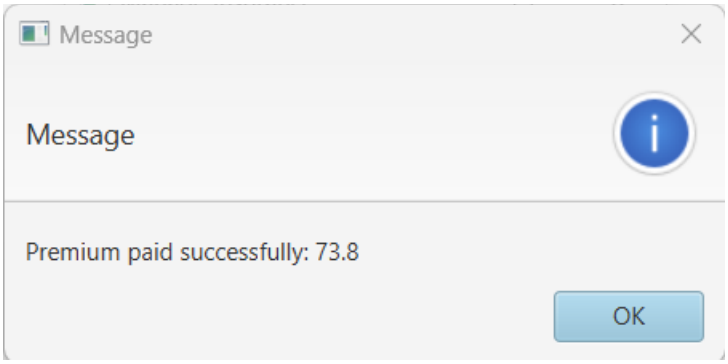
Cattle registration Screen



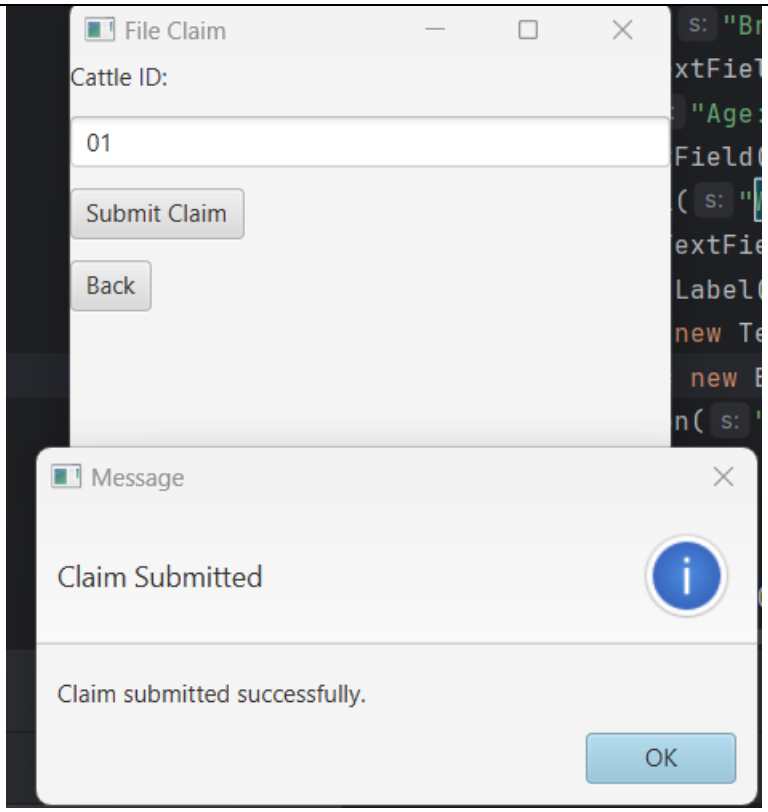
Insurance Management menu



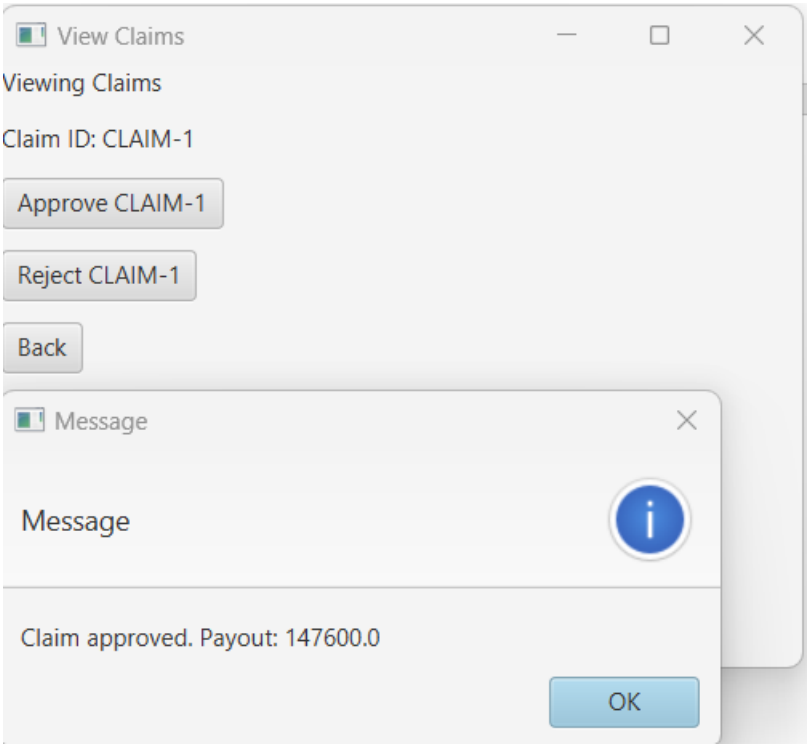
Applying for insurance



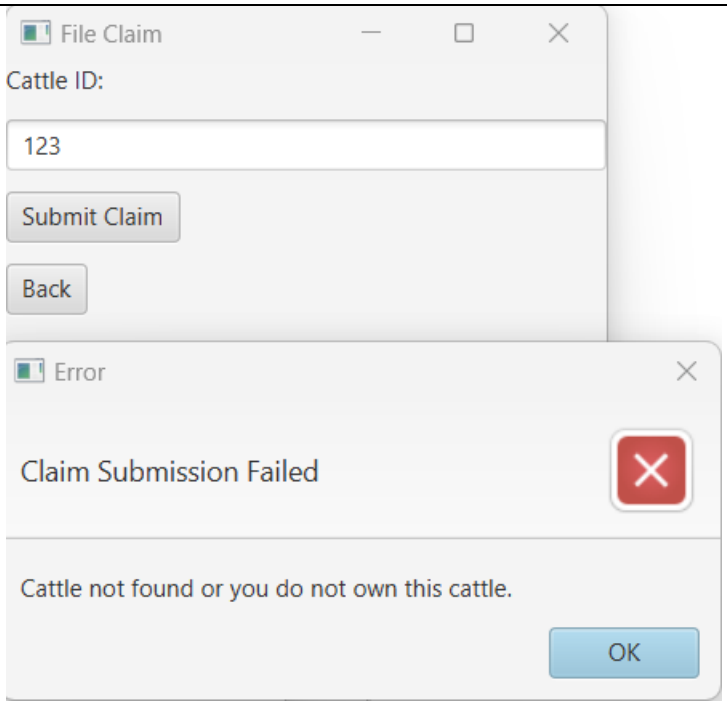
Payment of premium



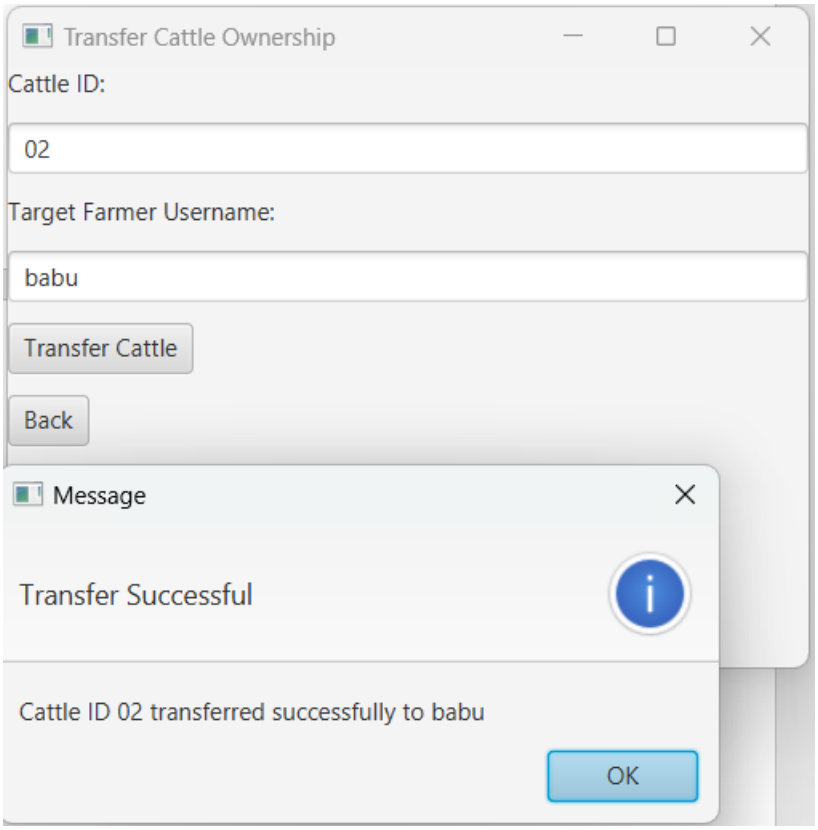
Filing of claim for cattle



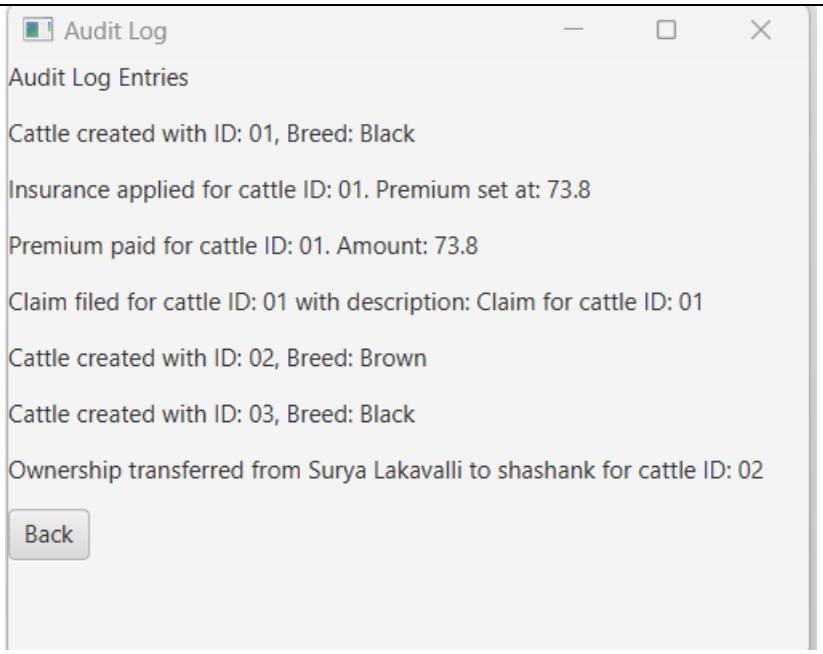
Approval of claim and insurance payout



Trying to file claim when insurance hasn't been paid cattle id not found



Cattle Transfer to farmer 2



Audit Log entries

Conclusion

To sum up, the Cattle Management System, which was created with Java and JavaFX, is a reliable and easy-to-use platform for effectively managing operations involving cattle. The solution greatly lowers the complexity involved with conventional livestock management techniques by integrating cutting-edge features including secure data handling, insurance claim administration, and ownership monitoring.

The system's scalability, security, and dependability are guaranteed by the application of contemporary software development techniques. The use of data integrity techniques, such as data encryption and SHA-256 checksums, highlights the project's dedication to safeguarding private data and upholding stakeholder trust. Furthermore, JavaFX's user-friendly graphical interface improves the user experience overall by making it simple and accessible for administrators, insurance brokers, and farmers.

This project not only shows how powerful and adaptable Java and JavaFX are for creating complex applications, but it also shows how technology can have a good effect on the agriculture industry. Future features like automated data analysis, real-time health monitoring, and interaction with IoT devices can be added to the Cattle Management System, which paves the way for even more effective and data-driven livestock management techniques.

Individual Contributions

Surya- Incharge of cattle transferring and filing the claims and the payout and compiling the main helloapplication class + doing the report

Shashank- Incharge of the cattle registration part and implementation of it in the main using javaFX

Abhyuday- Incharge of login Page

Arnav-Incharge of data integrity audit logging and security focusing on checksum and hashing algorithm

Shrey- Incharge of the insurance policies and implementing it in the main

References

https://youtu.be/ZrygmO6L__U?si=DxCKwp4CicnrwZch

https://github.com/inforkgodara/javafx-login-registration-admin-panel

https://youtu.be/FJeuiB5PqXc?si=N6haVSdyag3PsGaw

https://youtu.be/FJeuiB5PqXc?si=ptvtbphkh9MuXgFw

https://youtu.be/ZrygmO6L__U?si=DxCKwp4CicnrwZch