

“FARMER MARKET-PLACE WEBSITE”

Minor Project-II

Submitted by:-

NAME:

SHIVANGI TIWARI
SWATI UPADHYAY
VEDANSHI CHOUKSEY

ENROLLMENT NO.

0206CS221183
0206CS221210
0206CS221222

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GYAN GANGA INSTITUTE OF TECHNOLOGY & SCIENCES

JABALPUR (M.P.)

RAJIV GANDHI PRODYOGIKI VISHWAVIDYALAY, BHOPAL (M.P.)

Batch- 2022-2026

CERTIFICATE

This is to certify that the Minor Project-II entitled “FARMER MARKET-PLACE WEBSITE” submitted by Shivangi Tiwari, Swati Upadhyay and Vedanshi Chouksey has been carried out under my guidance & supervision. The project report is approved for submission towards partial fulfillment of the requirement for the award of degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING from RAJIV GANDHI PROUDYOGIKI VISHWA-VIDYALAYA, BHOPAL (M.P).

Prof. Shivendu Dubey
Guide

Dept. of Computer Science and Engineering

Dr. Ashok Verma
HOD

Dept. of Computer Science and Engineering

CERTIFICATE

This is to certify that the Minor Project-II entitled “FARMER MARKET-PLACE WEBSITE” is submitted by Shivangi Tiwari, Swati Upadhyay and Vedanshi Chouksey for the partial fulfillment of the requirement for the award of degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING from RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL (M.P).

Internal Examiner

Date:

External Examiner

Date:

DECLARATION

We hereby declare that the project entitled “FARMER MARKET-PLACE WEBSITE” which is being submitted in partial fulfillment of the requirement for award of the Degree of Bachelor of Technology in Computer Science to “RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL (M.P.)” is an authentic record of our own work done under the guidance of Prof. Shivendu Dubey, Department of Computer Science and Engineering, GYAN GANGA INSTITUTE OF TECHNOLOGY & SCIENCES, JABALPUR.

The matter reported in this Project has not been submitted earlier for the award of any other degree.

Date:

Place: JABALPUR

ACKNOWLEDGEMENT

We sincerely express indebtedness to esteemed and revered guide Prof. Shivendu Dubey of Department of Computer Science for his invaluable guidance, supervision and encouragement throughout the work. Without his kind patronage and guidance the project would not have taken shape.

We take this opportunity to express deep sense of gratitude to Dr. Ashok Verma, Head of Department of Computer Science for his encouragement and kind approval. Also we thank him in providing the computer lab facility. We would like to express our sincere regards to him for advice and counseling from time to time.

We owe sincere thanks to all the faculties in Department of Computer Science and Engineering for their advice and counseling time to time.

Date:

Place: JABALPUR

Name of team members:

Shivangi Tiwari

Swati Upadhyay

Vedanshi Chouksey

TABLE OF CONTENTS

Serial No.	Title	Page No.
	ABSTRACT	
1.	INTRODUCTION	1
	1.1 Purpose of Project	2
	1.2 Scope of Project	3
	1.3 Intended Audience	3
2.	TEAM ARCHITECTURE	4
3.	OVERALL DESCRIPTION	6
4.	PRODUCT PERSPECTIVE	7
5.	PROBLEM STATEMENT	8
6.	SPECIFIC REQUIREMENTS	9
	6.1 External Interface Requirements	9
	6.1.1 User Interface	9
	6.1.2 Hardware Interface	10
	6.1.3 Software Interface	11
	6.1.4 Communication Interface	12
	6.2 Performance Requirements	13
	6.3 Non-Functional Requirements	14
7.	SOFTWARE SYSTEM ATTRIBUTES	15
8.	SOFTWARE PROCESS MODEL	16
	8.1 Why not Evolutionary models?	16
	8.2 Why not Waterfall model?	16

	8.3 Why Incremental RAD model?	16
	8.4 Observation	17
9.	DETERMINING PROJECT FEASIBILITY	18
10.	DESIGN TECHNIQUES	19
11.	DESIGN MODEL	
	11.1 Data Flow Diagram	24
	11.2 Class Diagram	28
	11.3 Sequence Diagram	32
	11.4 ER Diagram	36
12.	SCREENSHOTS	40
13.	CONCLUSION	44
14.	BIBLIOGRAPHY	45

ABSTRACT

The Farmer Market-place Website is a MERN stack-based platform designed to bridge the gap between farmers and buyers. It allows farmers to list their products, and buyers to browse, search, and purchase agricultural goods. The platform supports secure payments, messaging, product trend analysis, and efficient communication between users

This project deals with developing an e-commerce website for online fresh foods product sale. It provides list of farmers that offer fresh fruits and vegetables, and products page for each farmer's offerings. It also provides a cart for ease of remembering the choices selected by user. The user can also view their order history to go back to the farmer from whom they purchased the last batch of products.

This system helps farmer to reduce cost of transportation and different extra expenses that they need to spend on selling of the grains in the wholesale market. Through this system they get fare price for their grains. Buyers are also supported through this portal So we have developed this kind of user interface here that is able to allow users to do all the stuff as mentioned above accurately and precisely and also this will help in reducing middle man in between and help farmers to get the fare price of the grains.

1. INTRODUCTION

Electronic Commerce is process of doing business through computer networks. E-Commerce is now seen as a major tool for many businesses and has become a normal part of almost every other business plan. The immediate benefits, in terms of cost savings, efficiencies and enhanced profitability, it offers, are beneficial at every stage in the supply chain. This paper outlines different aspects of developing an e-commerce website for Indian farmers to get involve in business directly without any need of a mediator in between and the optimum solution to the challenges involved in developing one. It consists of the planning process, which starts with determining the use case, domain modelling and architectural pattern of the web application.

The sole purpose of this project is to let farmers gain more income by becoming independent. As our respected Prime Minister often says that farmers are the backbone of a country so it will be our honour to help them with this proposed plan. This e-commerce website will let farmers sell their products directly to end-users at an appropriate cost without any middlemen. Today both the producer and consumer get looted because of the mediators that work in between which eventually led to an increase in the market price that in return affect the overall demand and supply chain. The platform not only helps the producers getting the deserving price for their goods but the customer getting fresh vegetables and fruits at a decent amount.

Agriculture is the backbone of many economies, especially in developing countries. Despite their vital role in food production, farmers often struggle to get fair prices for their produce due to the presence of multiple middlemen and lack of direct access to markets. Traditional marketing channels are inefficient, time-consuming, and often biased against small-scale farmers.

With the rapid growth of internet connectivity and digital technology, there is a significant opportunity to bridge this gap. The proposed Farmer Market-place Website aims to provide a digital platform where farmers can list their agricultural products and directly interact with buyers. This system reduces dependency on intermediaries, ensures transparent pricing, and offers features like secure payment, messaging, product trend analysis, and easy search functionalities.

1.1 Purpose of Project

The primary purpose of the Farmer Market-place Website is to empower farmers by providing them with a digital platform to directly connect with buyers, eliminating the need for intermediaries. This system enables farmers to showcase and sell their agricultural products online, access real-time price trends, securely receive payments, and communicate with buyers through an integrated messaging system.

By leveraging modern web technologies (MERN stack), the project aims to:

- Promote transparency in agricultural product pricing.
- Expand the market reach for small and large-scale farmers.
- Ensure secure and traceable transactions.
- Simplify the product listing and discovery process.
- Strengthen the agricultural supply chain by fostering direct farmer-consumer interaction.

Ultimately, the system seeks to improve the livelihood of farmers and enhance the efficiency of the agricultural market through digitization.

1.2 Scope of the Project

The Farmer Market-place Website is designed to revolutionize the way agricultural products are marketed and sold by leveraging digital technology. The platform provides a virtual space where farmers can list and manage their produce while buyers can search for, communicate with, and purchase directly from the farmers.

- User authentication with role-based access (Farmer or Buyer).
- Product listing by farmers with image upload, pricing, and unit details.
- Search functionality for buyers to filter and find products easily.
- Real-time messaging system for direct farmer-buyer communication.
- Secure payment integration (e.g., PayPal) for seamless transactions.
- Display of market price trends to help buyers and farmers make informed decisions.

1.3 Intended Audience

This project is intended for:

- Farmers – who want to sell their produce directly to consumers.
- Buyers/Consumers – who are looking to purchase fresh farm products online.
- Market Administrators – who may oversee platform activity and user interactions.
- Developers & Researchers – interested in agricultural e-commerce solutions.

The system is designed to be user-friendly for both tech-savvy users and those with minimal digital experience.

2. Team Architecture

The development of the Farmer Market-place Website was carried out by a team of three members, each contributing their expertise in specific domains to ensure the successful completion of the project. The team followed a collaborative approach using version control tools (like Git and GitHub) and task allocation to maintain clarity, coordination, and consistency throughout the project lifecycle.

Member 1 – Frontend Developer (UI/UX Specialist)

This member was primarily responsible for designing and developing the user interface of the application using React.js and Tailwind CSS. Their role included:

- Creating responsive and user-friendly components for the buyer and farmer dashboards.
- Designing intuitive navigation and layout for all key pages (Login, Signup, Product Listings, Profile, etc.).
- Implementing dynamic rendering of data using state management (React Hooks).
- Ensuring cross-browser compatibility and mobile responsiveness.
- Collaborating closely with the backend developer for seamless API integration.

Member 2 – Backend Developer (API & Database Manager)

The second team member focused on building and managing the backend of the application using Node.js and Express.js, with MongoDB as the primary database. Their tasks included:

- Designing and implementing RESTful APIs for user authentication, product management, and messaging.
- Securing endpoints using JWT tokens and role-based access control (farmer vs buyer).
- Integrating Multer for file uploads and bcrypt for password hashing.
- Structuring the MongoDB schema for users, products, messages, and orders.
- Managing error handling, validation, and performance optimization at the server level.

Member 3 – Full Stack Developer & Integrations Lead

The third member worked across both frontend and backend to handle the overall integration of system components. Their responsibilities included:

- Implementing the real-time messaging system using web sockets or REST APIs.
- Ensuring data consistency and seamless communication between frontend and backend.
- Conducting unit and integration testing, bug fixing, and performance tuning.
- Managing the final deployment of the project on platforms like Heroku, Netlify, or Render.

3. Overall Description

- A web-based platform built using MERN stack (MongoDB, Express.js, React.js, Node.js).
- Connects farmers directly with buyers, removing the need for intermediaries.
- Key features include:
 - Role-based login (Farmer / Buyer)
 - Product listing and management
 - Search and filter for buyers
 - Real-time messaging system
 - Price trend charts
 - Secure online payments (e.g., PayPal)
- Responsive UI – accessible via desktop and mobile.
- Benefits:
 - Fair pricing for farmers
 - Direct access to fresh produce for buyers
 - Transparent and efficient trade
- Future scope: Mobile app, logistics, multilingual support, government tie-ups.

4. Product Perspective

The Farmer Market-place Website is an independent web-based application that serves as a digital platform to connect farmers directly with buyers. It is designed using a modular architecture, where each component—such as user authentication, product management, secure payments, real-time messaging, and price trend tracking—operates cohesively within the system. The application follows a client-server model, with the frontend developed using React.js to provide an interactive and responsive user experience, and the backend built on Node.js and Express.js to handle server-side logic and API communication. MongoDB is used as the database to store information related to users, products, messages, and transactions. Additionally, the platform integrates external services like PayPal for secure payment processing and includes support for image uploads to enhance product visibility. The overall system is designed to be scalable and flexible, making it easy to incorporate additional features such as logistics tracking and mobile app support in future versions.

5. Problem Statement

Traditional agricultural marketing in many regions involves multiple intermediaries between farmers and end consumers, leading to reduced profits for farmers and inflated prices for buyers. Farmers often lack access to broader markets, transparent pricing, and direct communication with buyers. Additionally, the absence of a digital platform restricts their ability to showcase their products effectively, access real-time market trends, or receive fair compensation for their efforts.

This project aims to solve these issues by developing a web-based marketplace that enables farmers to list and sell their produce directly to buyers. The platform ensures secure payments, transparent pricing, real-time communication, and up-to-date price trends—ultimately empowering farmers and improving buyer access to fresh, fairly priced goods.

6. Specific Requirements

6.1 External Interface Requirements

6.1.1 User Interface

The user interface (UI) of the Farmer Market-place Website is designed to be intuitive, responsive, and role-based, catering to both farmers and buyers. It is developed using React.js with Tailwind CSS for a modern and mobile-friendly experience.

- Login/Registration Pages: Separate options for Farmers and Buyers with secure authentication and role-based redirection.
- Farmer Dashboard: Allows farmers to upload product details (name, image, price, unit), view listings, manage orders, and communicate with buyers.
- Buyer Dashboard: Enables buyers to browse/search products, add items to cart, make payments, view order history, and chat with farmers.
- Search & Filter Interface: Buyers can search products by category, price range, or product name.
- Messaging System: A clean chat UI for one-on-one farmer-buyer conversations.
- Payment Gateway Integration: A simple and secure checkout process using PayPal.
- Price Trends Page: Graphs and charts display recent price trends for key commodities.

The overall UI is built to ensure minimal learning curve, ease of navigation, and accessibility on both desktop and mobile devices.

6.1.2 Hardware Interface

The Farmer Market-place Website is a web-based application and does not require any specialized hardware to operate. However, to access and use the platform efficiently, the following minimum hardware requirements are recommended:

- For End Users (Farmers & Buyers):
 - Desktop or Laptop with at least 2 GHz processor
 - Minimum 4 GB RAM
 - Internet connection (minimum 1 Mbps recommended)
 - Smartphone (Android/iOS) with a modern web browser for mobile access
 -
- For Server Hosting:
 - Cloud-based or dedicated server with:
 - At least 2-core CPU
 - 4–8 GB RAM
 - 50 GB storage space (SSD preferred)
 - Secure internet connection
 - Node.js and MongoDB runtime environments installed
 -

Since the application is built on a client-server model, the hardware mainly affects hosting and user device performance. No additional hardware components (e.g., sensors, embedded systems) are required for this system.

6.1.3 Software Interface

The Farmer Market-place Website uses a modern full-stack JavaScript environment for seamless communication between the frontend, backend, and database. The following software interfaces are involved:

- Frontend Interface (Client Side):
 - Built using React.js
 - Styled with Tailwind CSS
 - Interacts with backend APIs via Axios (HTTP client)
- Backend Interface (Server Side):
 - Developed using Node.js with Express.js
 - Exposes RESTful APIs for authentication, product handling, messaging, and payments
 - Handles middleware, routing, and request validation
- Database Interface:
 - Uses MongoDB as the NoSQL database
 - Data is accessed and manipulated through Mongoose ORM for schema modelling.

All components work together to create a seamless, interactive, and secure user experience across the system.

6.1.3 Communication Interface

The communication interface of the Farmer Market-place Website ensures smooth interaction between the client-side, server-side, and external services. The following communication protocols and technologies are used to ensure seamless operation and real-time updates:

- **HTTP/HTTPS Protocol:**

- The primary method for communication between the frontend and backend.
- Secure HTTPS is used to protect data during transmission, especially sensitive user information and payment details.

- **RESTful APIs:**

- The backend exposes RESTful APIs for various functionalities such as user authentication, product management, messaging, and payment processing.
- These APIs are consumed by the frontend using **Axios** to fetch data asynchronously and update the UI without page reloads.

- **PayPal API:**

- Securely integrates with PayPal to handle payment transactions.
- Communicates via the PayPal API to verify payments, process transactions, and send back payment confirmation.

- **JWT Authentication:**

- JSON Web Tokens (JWT) are used for secure communication between the client and the server to authenticate and authorize users for protected routes.
- Tokens are passed in HTTP headers to validate requests and ensure user identity during API calls.

This communication infrastructure ensures that the platform remains highly responsive, secure, and capable of handling real-time interactions between farmers, buyers, and external services.

6.2 Performance Requirements

- **Response Time:** Pages should load in under 3 seconds, and API responses in under 2 seconds.
- **Scalability:** Must support 1,000 concurrent users and scale horizontally or vertically as needed.
- **Database Performance:** Queries should execute in under 200 milliseconds; use indexes for frequently accessed data.
- **Security & Reliability:** Ensure 99.9% uptime, secure HTTPS connections, and 99.99% transaction accuracy with PayPal.
- **Concurrent Users:** Handle 500 simultaneous logins and 200 active messages without lag.
- **Error Handling:** Graceful error messages and automatic alerts for critical system failures.

6.3 Non-Functional Requirements

- **Usability:** The system should have a clean, intuitive interface that's easy to navigate for both farmers and buyers.
- **Reliability:** The platform should maintain high uptime ($\geq 99.9\%$) and recover quickly from failures.
- **Security:** All data must be encrypted during transmission (HTTPS). User authentication uses JWT for secure session handling.
- **Maintainability:** The codebase should be modular and well-documented to allow easy updates and debugging.
- **Portability:** The web app should be accessible across all modern browsers and mobile devices.
- **Performance:** The system should handle concurrent users efficiently without slowing down key functionalities.

7. Software System Attributes

The Farmer Market-place Website is built to ensure stability, efficiency, and adaptability through the following key software attributes:

- **Modularity:** The system is divided into independent modules like authentication, product management, messaging, and payments—making development and maintenance easier.
- **Security:** Implements secure authentication using JWT, data encryption via HTTPS, and secure payment integration using PayPal API.
- **Scalability:** The backend is designed to scale horizontally (adding more servers) to support a growing user base and increasing data load.
- **Availability:** Designed for high availability with proper error handling, recovery mechanisms, and uptime of 99.9%.
- **Maintainability:** Clean, well-commented code with separate frontend and backend codebases makes updates and bug fixes more efficient.
- **Reusability:** Components and APIs are written in a reusable manner to support future enhancements or mobile app integration.

8. Software Process Model

8.1 Why not Evolutionary Models?

The Evolutionary Model involves continuous changes and feedback, which can cause delays and increase cost. Since our project has well-defined requirements and a clear goal, this model is unnecessary. It also risks inconsistent design, which is not ideal for integrating features like payments and authentication.

8.2 Why not Waterfall Model?

The Waterfall Model is rigid and follows a strict sequence of phases. It doesn't allow going back to make changes once a stage is completed. Since our project might require adjustments during development, this model lacks flexibility and is not suitable for handling evolving user needs or feedback.

8.3 Why Incremental RAD Model?

The Incremental Rapid Application Development (RAD) model allows building the project in smaller, manageable parts (modules), which are developed and tested independently. This approach:

- Enables faster delivery of functional components.
- Supports user feedback and iterative improvement.
- Makes it easier to modify or add features during development.
- Ensures parallel development by multiple team members, improving efficiency.

It suits our project well due to its modular structure—registration, product listing, payments, etc.—each of which can be developed incrementally.

8.4 Observation

During the initial stages of development, we observed that using a flexible and modular approach like the Incremental RAD model helped us manage tasks efficiently within our small team. Breaking the system into components like user login, product listing, and messaging allowed us to work in parallel and integrate features gradually. It also made testing and debugging easier, and provided room for implementing changes based on user feedback without disrupting the whole system.

9. Determining Project Feasibility

The feasibility study is not a full-blown systems study. Rather, the feasibility study is used to gather broad data to make a decision on whether to proceed with system study. To ensure the success of the Farmer Market-place Website, we evaluated its feasibility across multiple dimensions:

- Technical**

Feasibility:

The MERN stack (MongoDB, Express.js, React.js, Node.js) is well-supported, scalable, and suitable for building a full-featured web application with real-time capabilities.
- Economic**

Feasibility:

The project uses open-source technologies, minimizing costs. It can be hosted on low-cost cloud services, making it affordable for small teams or startups.
- Operational**

Feasibility:

The system is user-friendly and can be easily adopted by farmers and buyers with minimal training. Features like product search, messaging, and payment are intuitive.
- Legal**

Feasibility:

By integrating secure payment gateways and ensuring data privacy with HTTPS and JWT, the system adheres to basic online commerce and security standards.

10. Design Techniques

The Farmer Market-place Website leverages Node.js, Express, React, and MongoDB for efficient, scalable, and maintainable web application development. Here's how these technologies work together to create a robust solution:

1. Node.js for Backend Development:

- **Description:** Node.js is a runtime environment built on Chrome's V8 JavaScript engine, which allows for server-side JavaScript development. This enables a unified language across both frontend and backend development.
- **Design Benefit:**
 - Asynchronous, Non-blocking I/O: Node.js handles multiple simultaneous requests efficiently, making it ideal for real-time applications (e.g., messaging, live updates).
 - Event-driven Architecture: Node.js efficiently manages I/O operations with minimal resource consumption, enhancing performance, especially with high concurrent requests.

2. Express.js for API Development:

- **Description:** Express.js is a lightweight web application framework for Node.js that simplifies the creation of RESTful APIs. It allows developers to handle routing, middleware, and HTTP requests effectively.
- **Design Benefit:**
 - Scalable API Design: Express helps create scalable routes for core functionalities such as user registration, product management, messaging, and payment processing.

- Middleware Support: Express uses middleware to handle requests such as user authentication (JWT), logging, error handling, and input validation.

3. React.js for Frontend Development

- **Description:** React.js is a JavaScript library for building dynamic and responsive user interfaces. It enables the creation of component-based, reusable UI elements.
- **Design Benefit:**
 - Component-based Architecture: React's component structure helps manage complex UI elements by breaking them into smaller, reusable components (e.g., Product Card, Navbar, Cart).
 - Virtual DOM: React uses a virtual DOM to optimize UI updates, ensuring smooth interactions and enhancing performance, particularly for frequently updated content like product listings.
 - State Management: Using React Context API or Redux, state management for data such as user login status, product information, and shopping cart is centralized, providing a consistent and smooth user experience.

4. MongoDB for Database Management:

- **Description:** MongoDB is a NoSQL database that stores data in JSON-like documents. It is well-suited for applications that require flexibility, scalability, and fast access to data.
- **Design Benefit:**

- Flexible Schema: MongoDB's schema-less structure allows for easy modification of the data model. This is beneficial for handling dynamic and unstructured data such as product listings, user profiles, and messages.
- Horizontal Scalability: MongoDB scales out by sharding data across multiple servers, making it capable of handling large volumes of data and user traffic as the platform grows.
- Efficient Querying: MongoDB supports complex queries and indexing, making it easy to fetch product listings, user details, orders, and messages efficiently.
- **Database Collections:**
 - Users: Stores both farmer and buyer details (name, email, role, password, etc.).
 - Products: Contains information about products listed by farmers (title, price, description, image URL, etc.).
 - Orders: Tracks buyer orders, including the products purchased and payment status.
 - Messages: Handles real-time messaging between buyers and farmers.
 -

These design techniques, utilizing Node.js, Express, React, and MongoDB, provide a robust and scalable architecture for the Farmer Market-place Website, supporting features like user authentication, real-time messaging, product listings, and secure payments.

MODULES :

1. User Management Module

Purpose: Handles registration, login, authentication, and role-based access.

Features:

- User registration (Farmer / Buyer)
 - Secure login using JWT
 - Role-based access (Farmer Dashboard vs Buyer Dashboard)
 - Forgot Password / Reset password
-

2. Product Management Module

Purpose: Enables farmers to add and manage product listings.

Features:

- Add product with image, description, unit, and price
 - View/manage farmer's products
 - Product availability status
 - Unit-based pricing (e.g., ₹/kg, ₹/litre)
-

3. Search & Marketplace Module

Purpose: Allows buyers to explore, search, and filter products.

Features:

- Quick search bar
 - Category-based filtering
 - Product details view
 - Add to cart
-

4. Messaging & Notification Module

Purpose: Supports real-time communication between farmers and buyers.

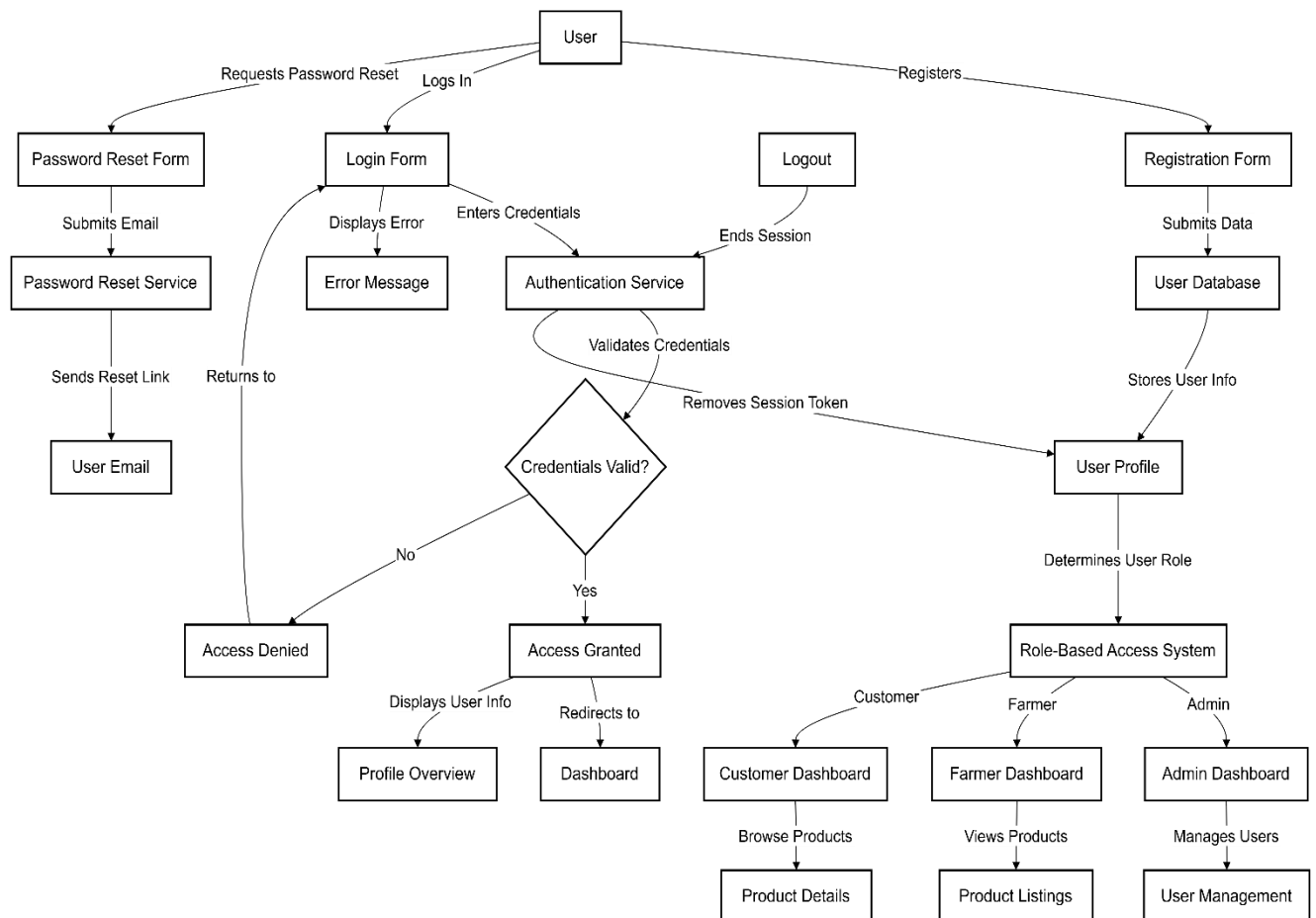
Features:

- Chat system with multiple users
 - Media attachments (images, product references)
 - Search messages
 - New message notifications
-

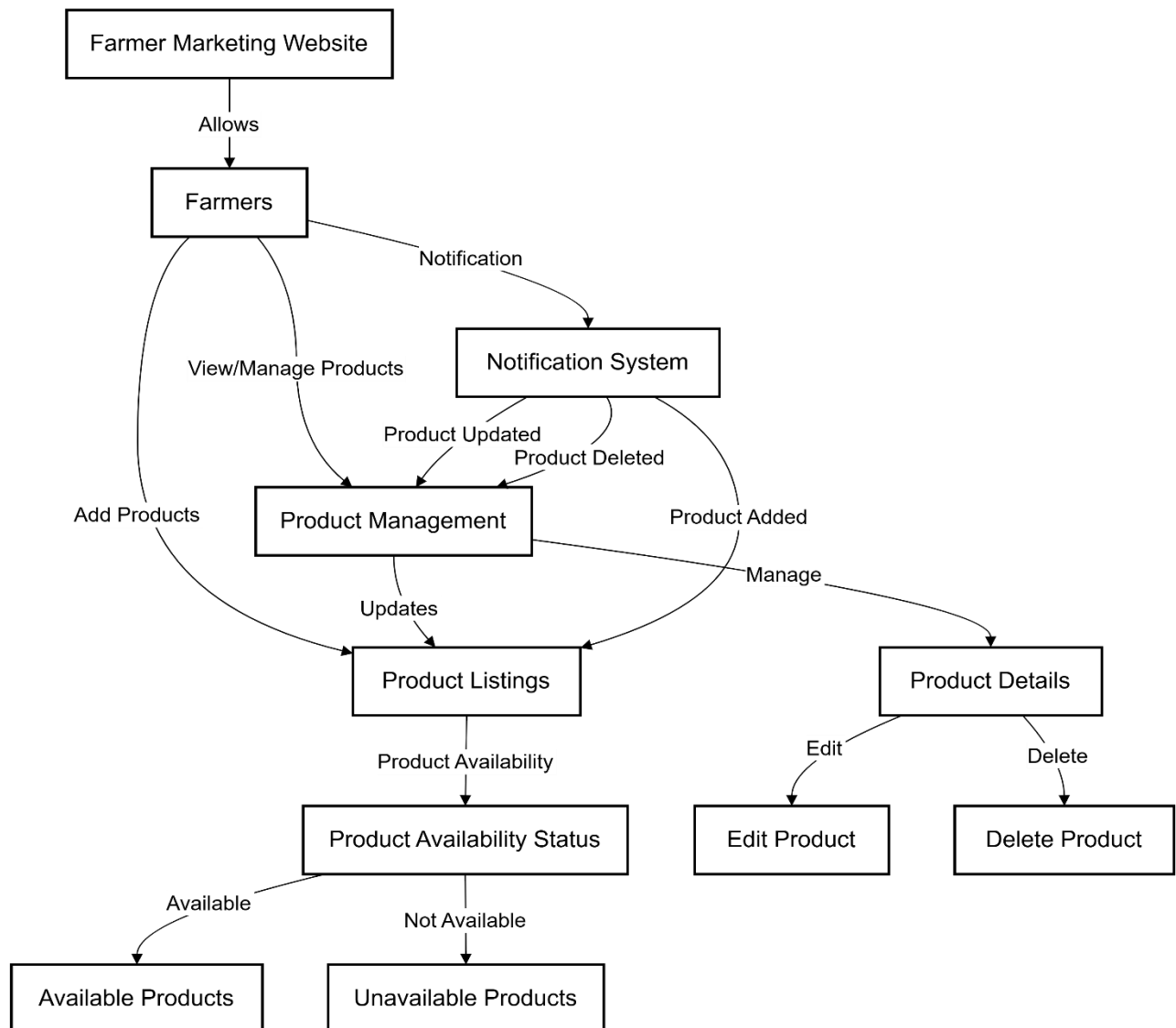
11. Design Models

11.1 Data Flow Diagram

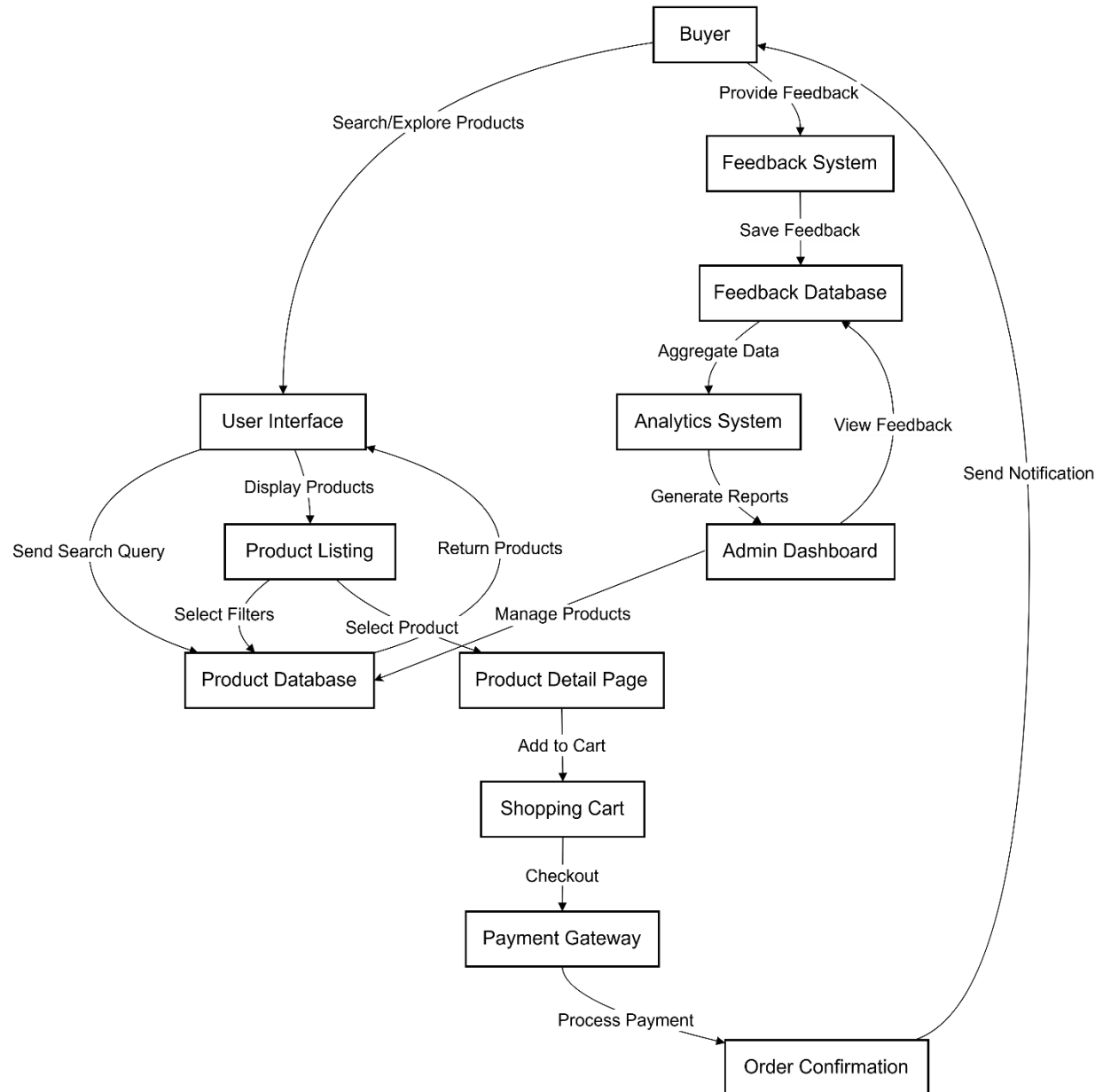
MODULE 1:



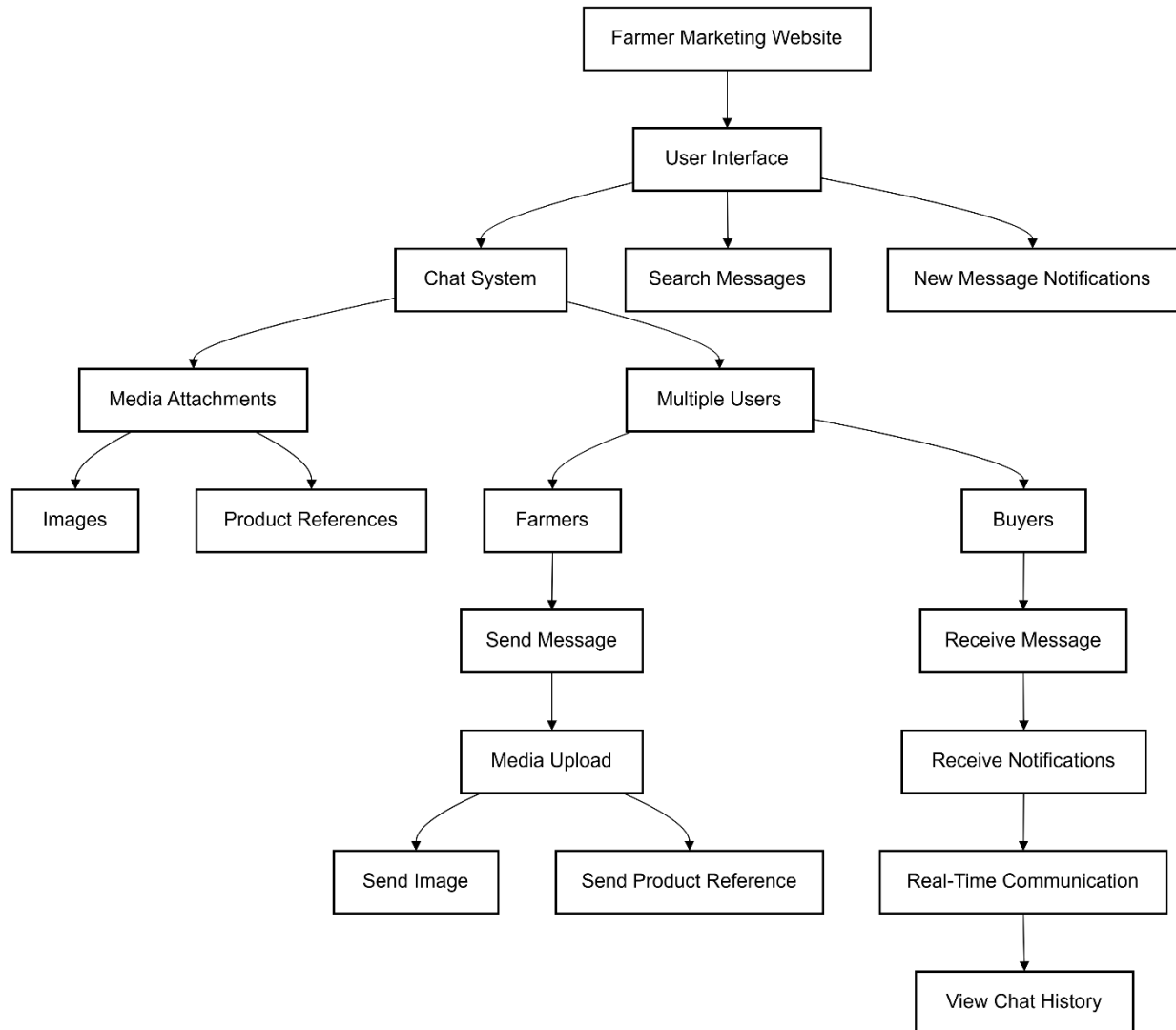
MODULE 2:



MODULE 3:

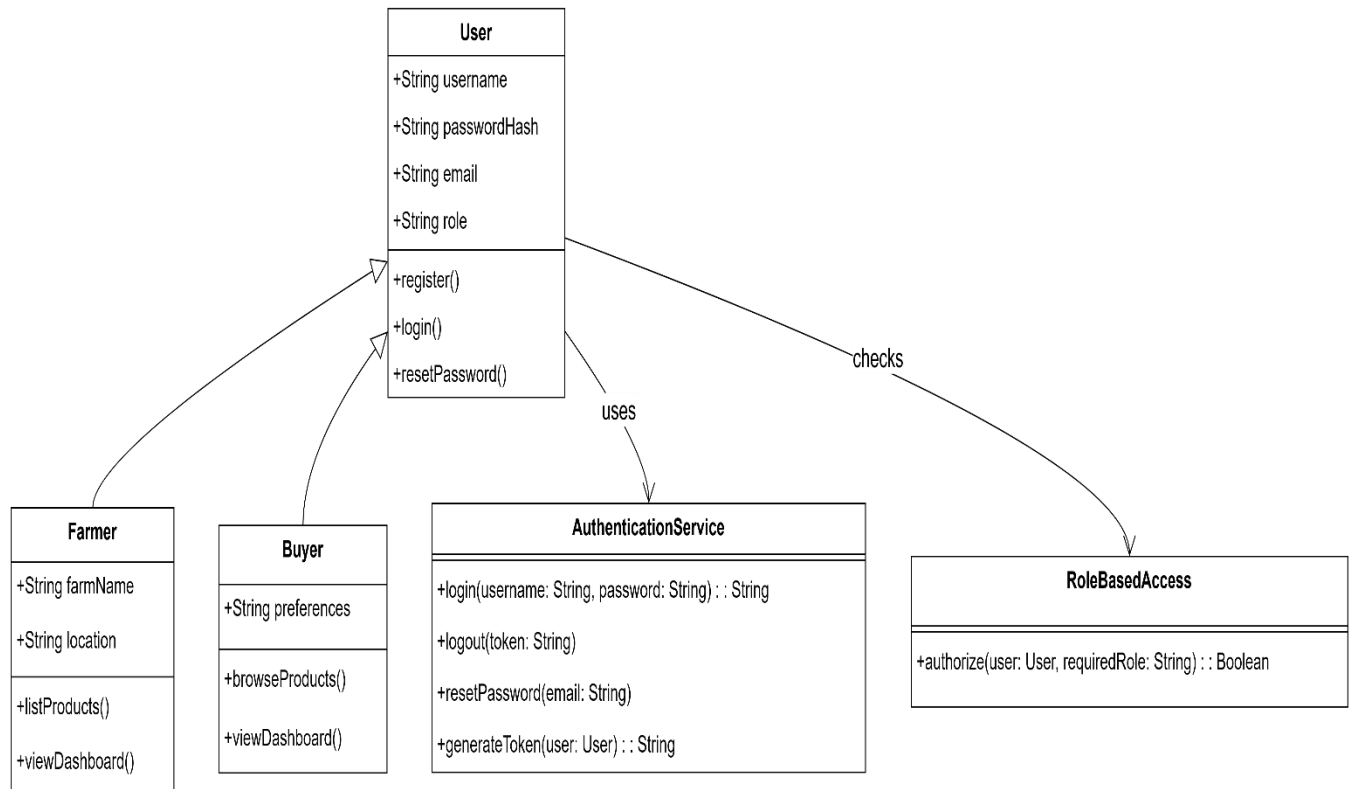


MODULE 4:

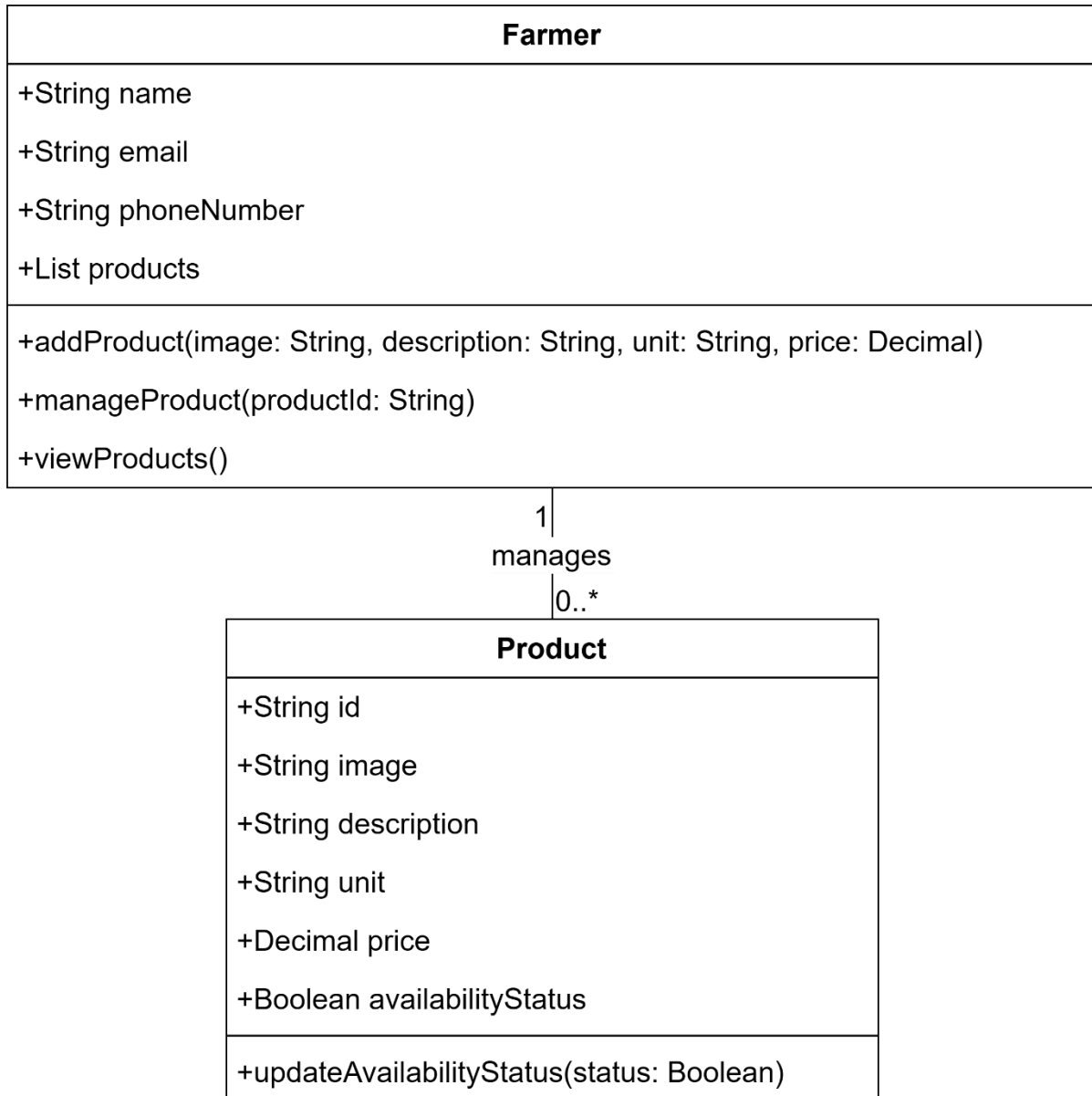


11.2 Class Diagram

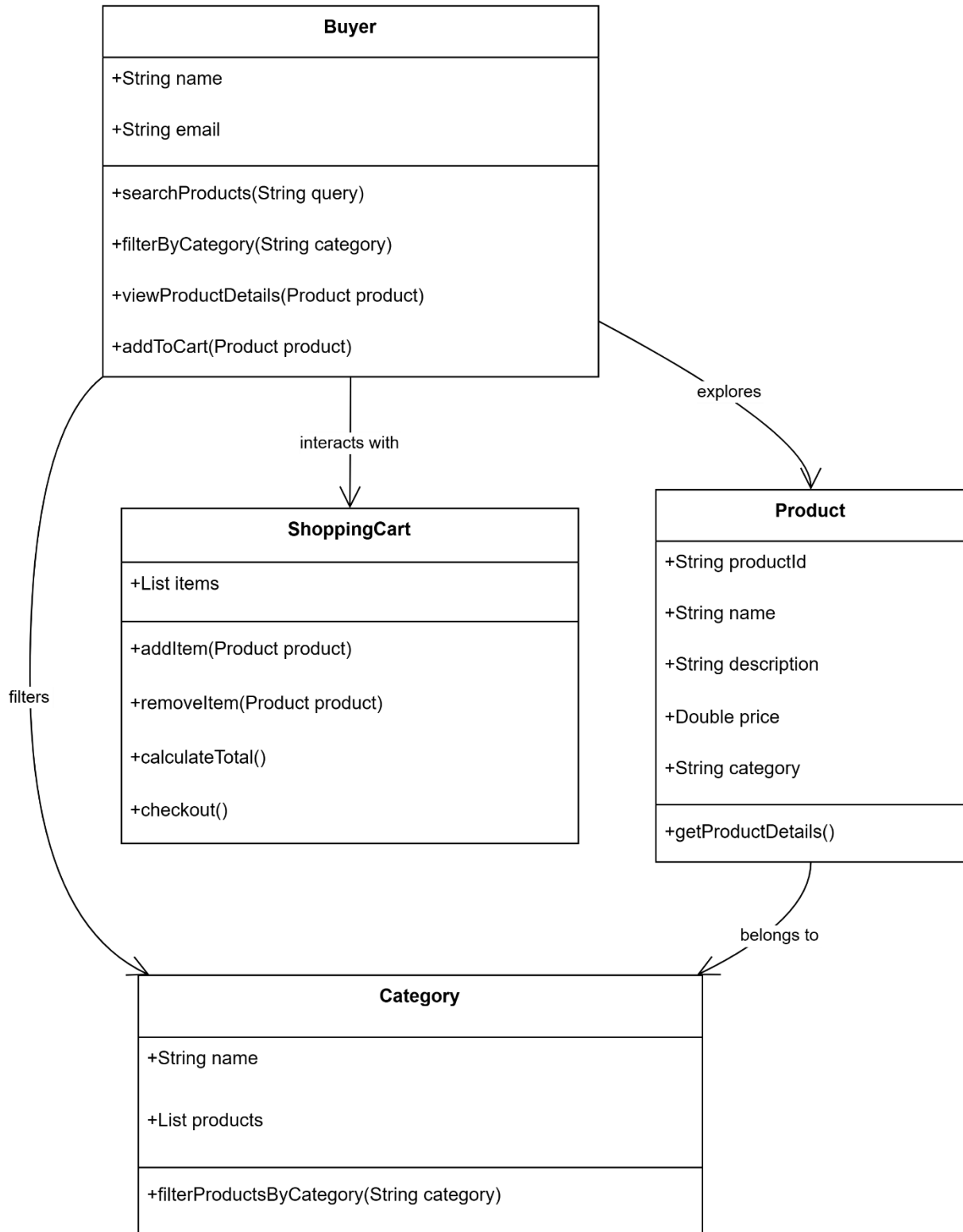
MODULE 1:



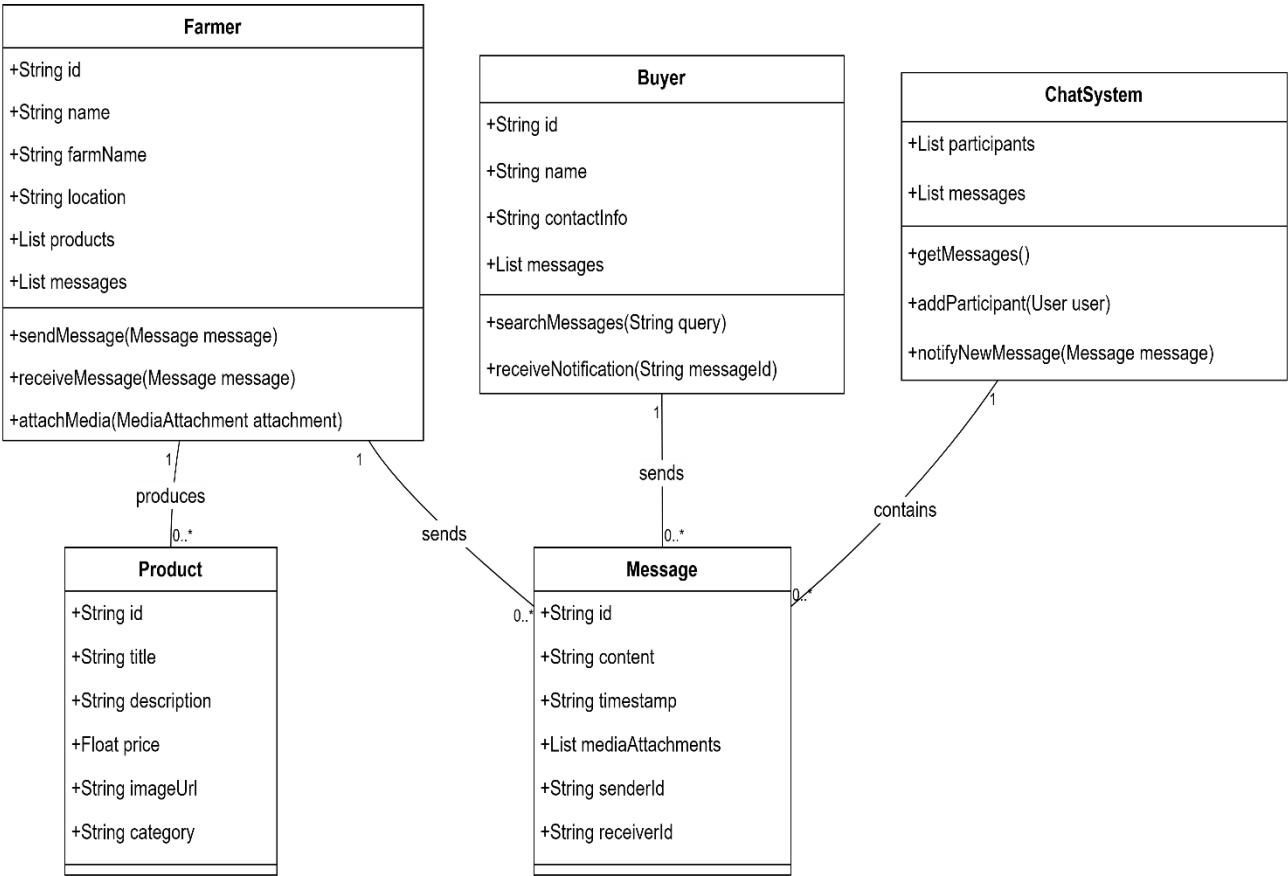
MODULE 2:



MODULE 3:

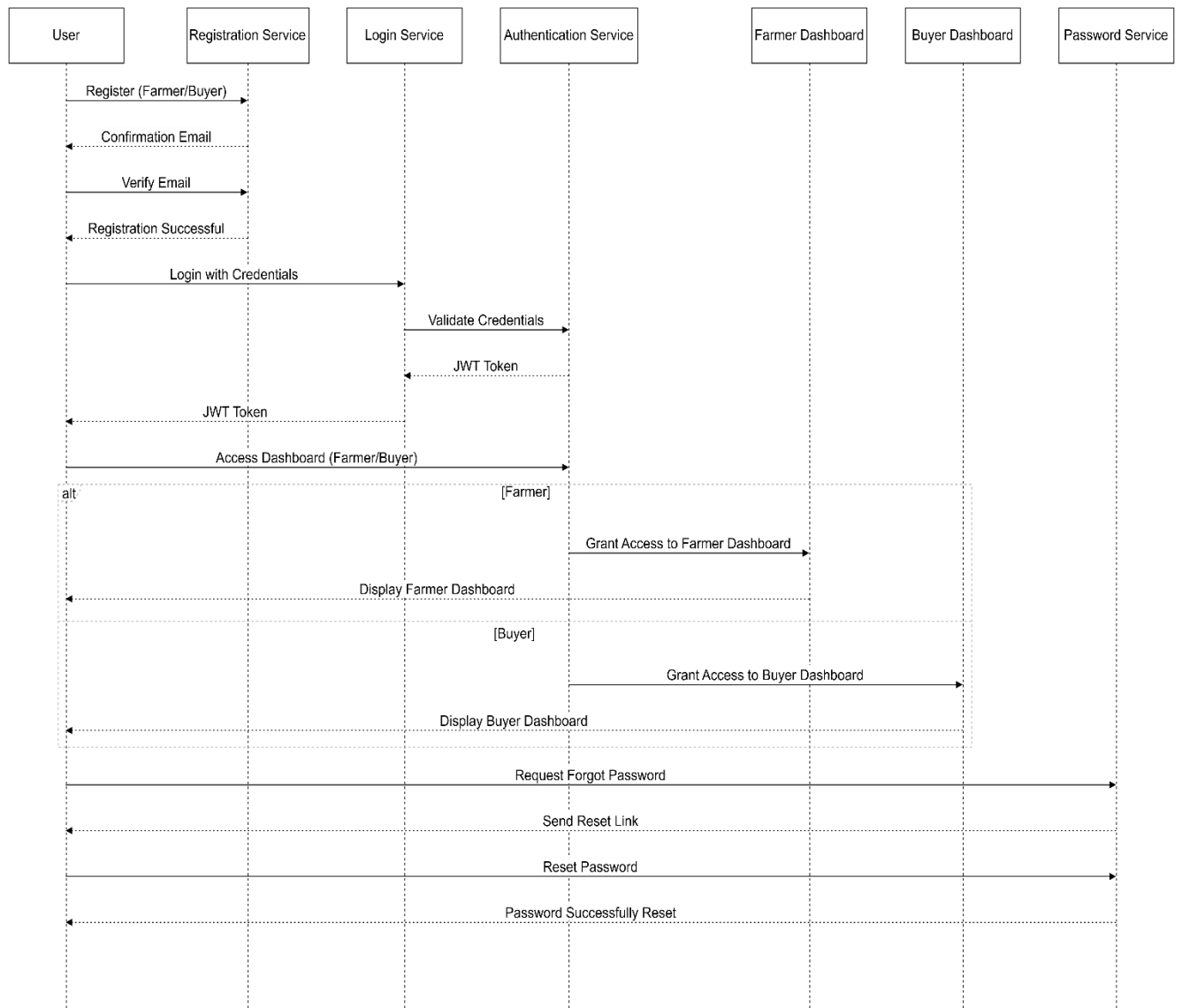


MODULE 4:

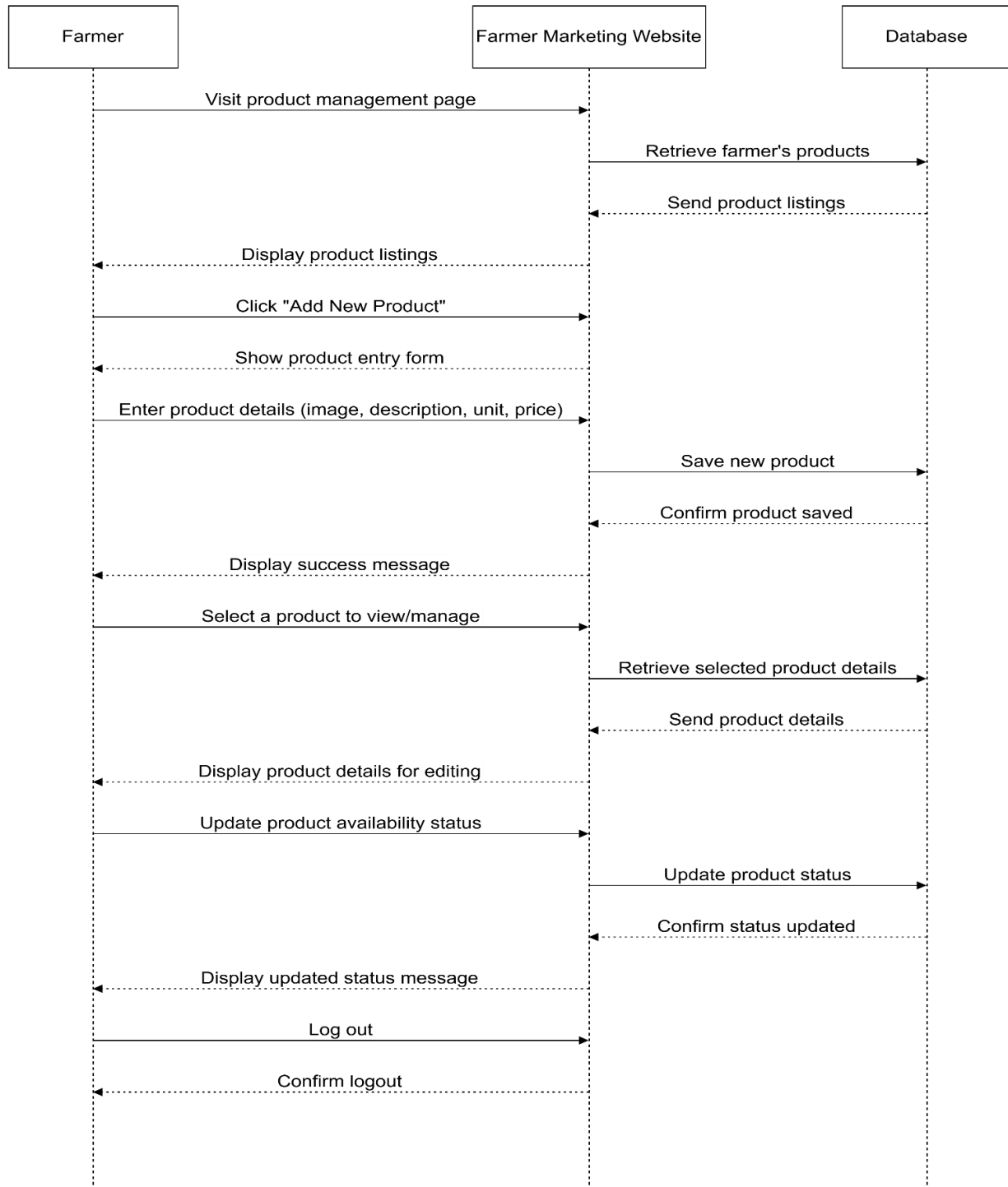


11.3 Sequence Diagram

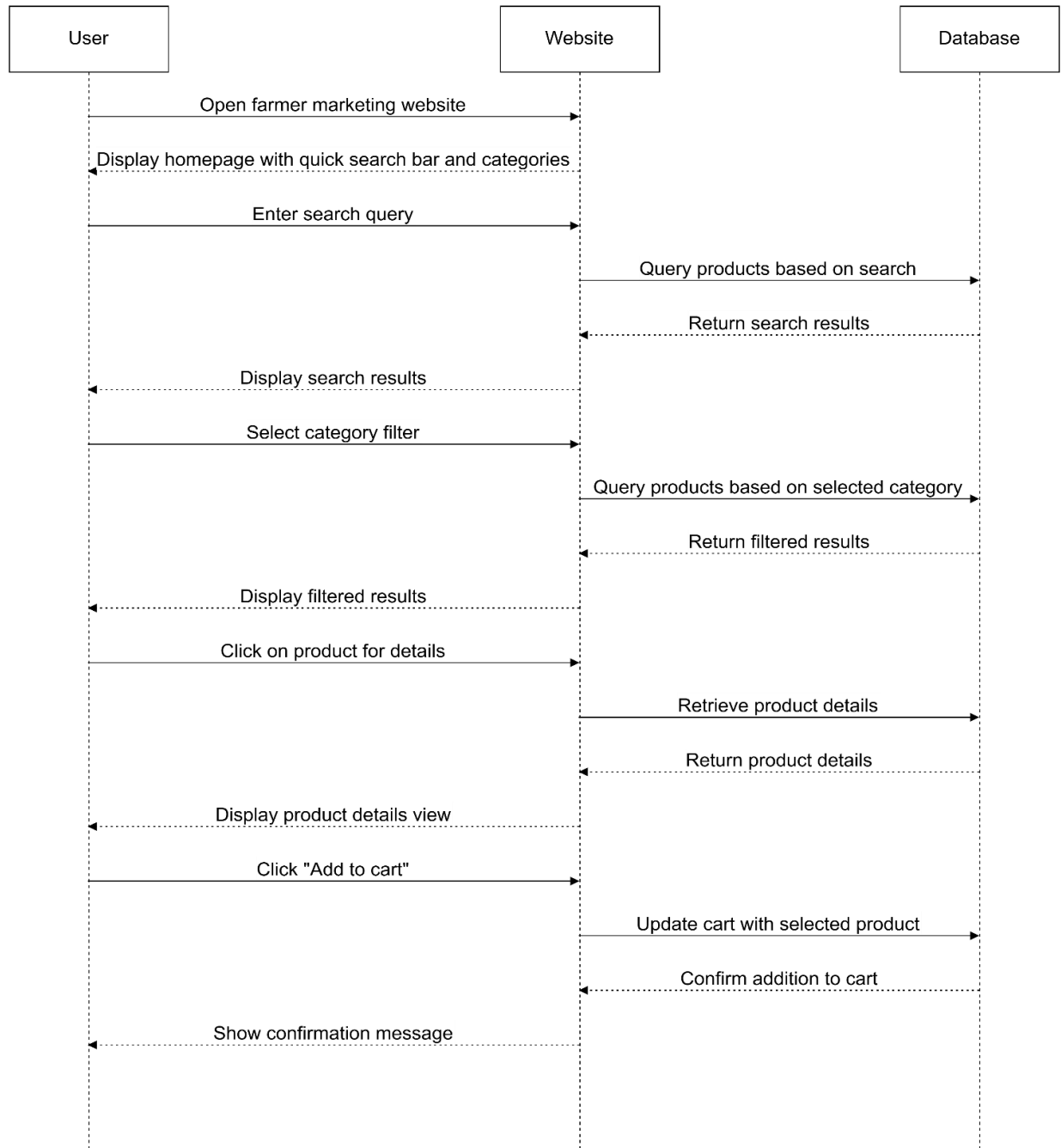
MODULE 1:



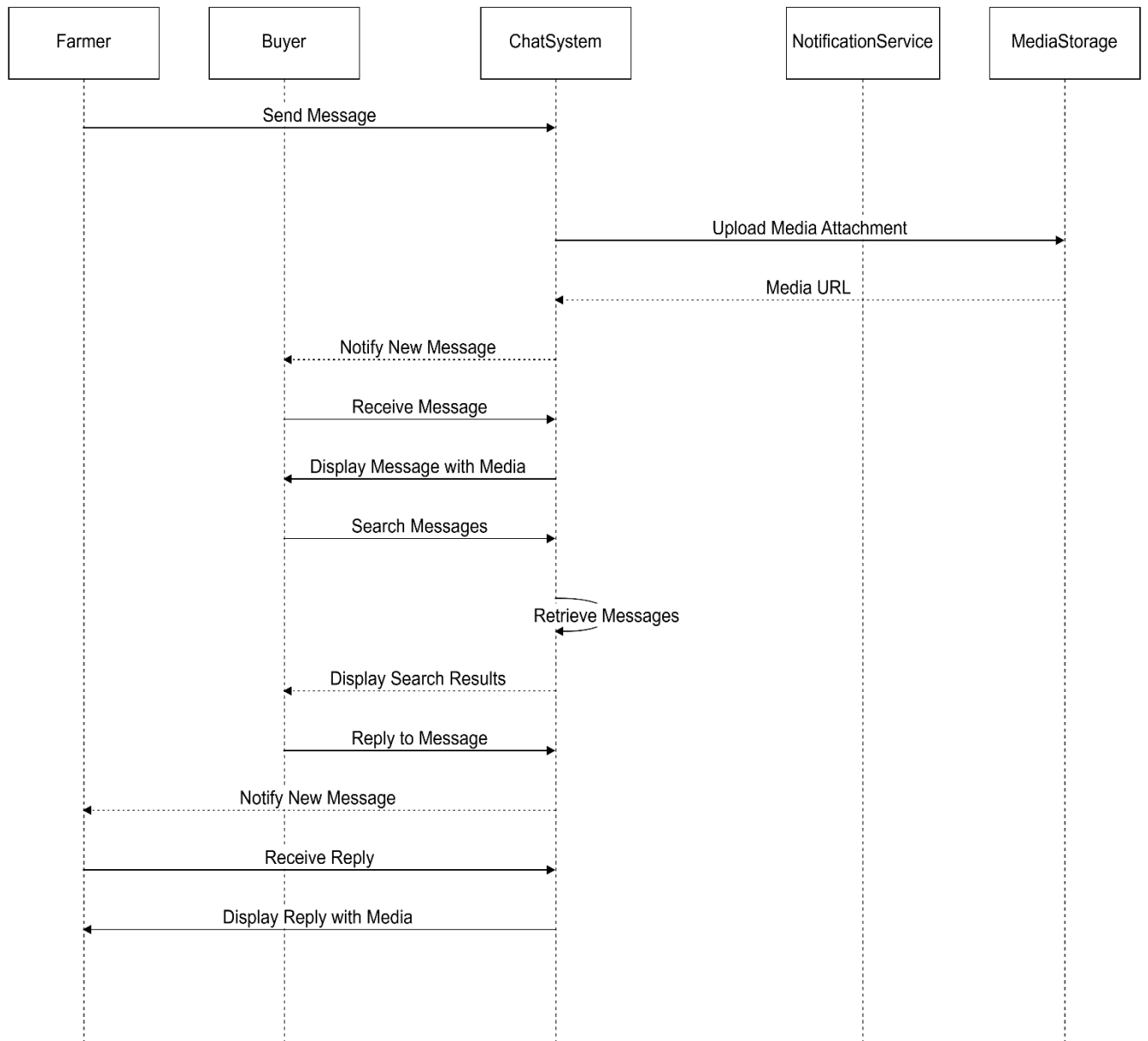
MODULE 2:



MODULE 3:

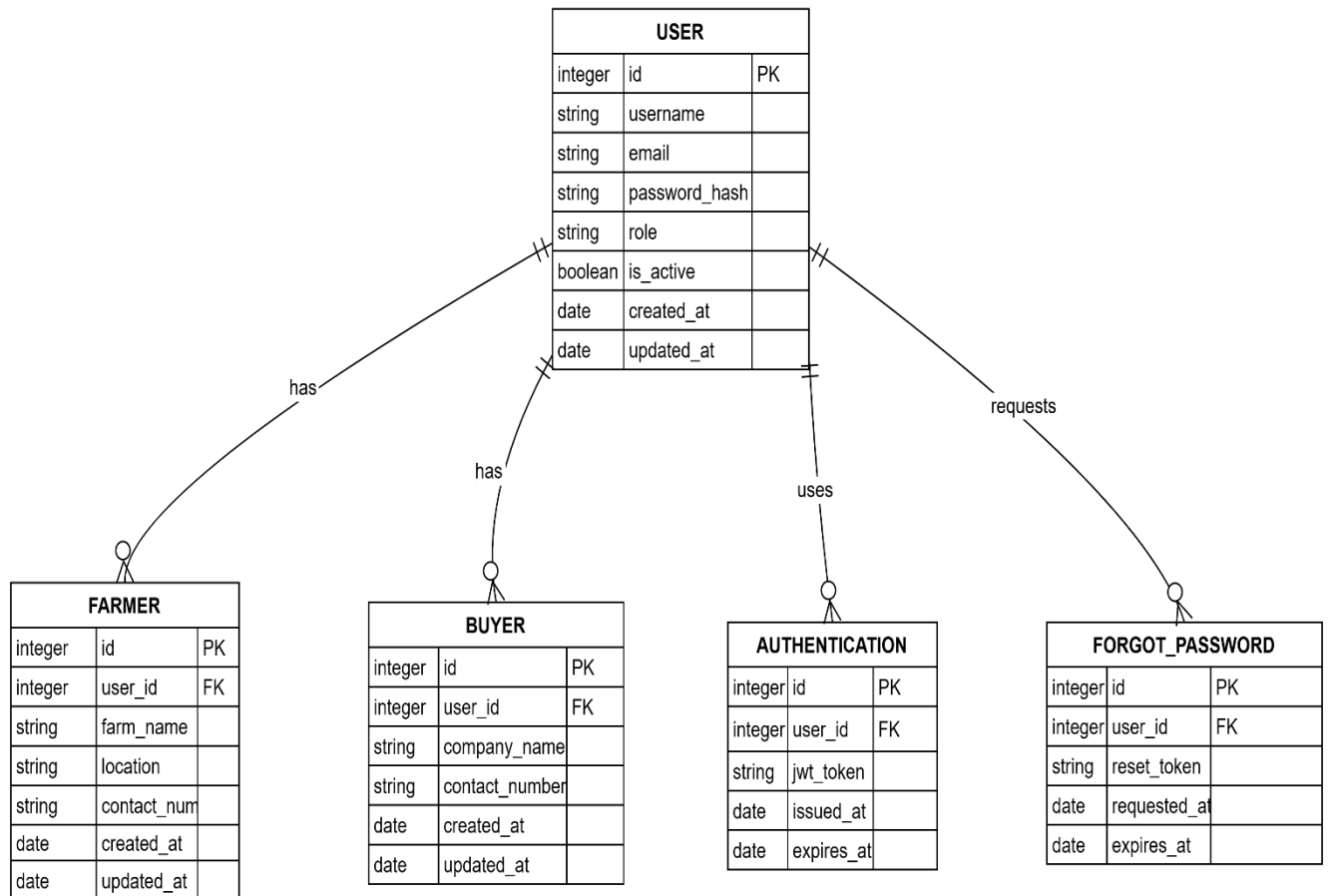


MODULE 4:

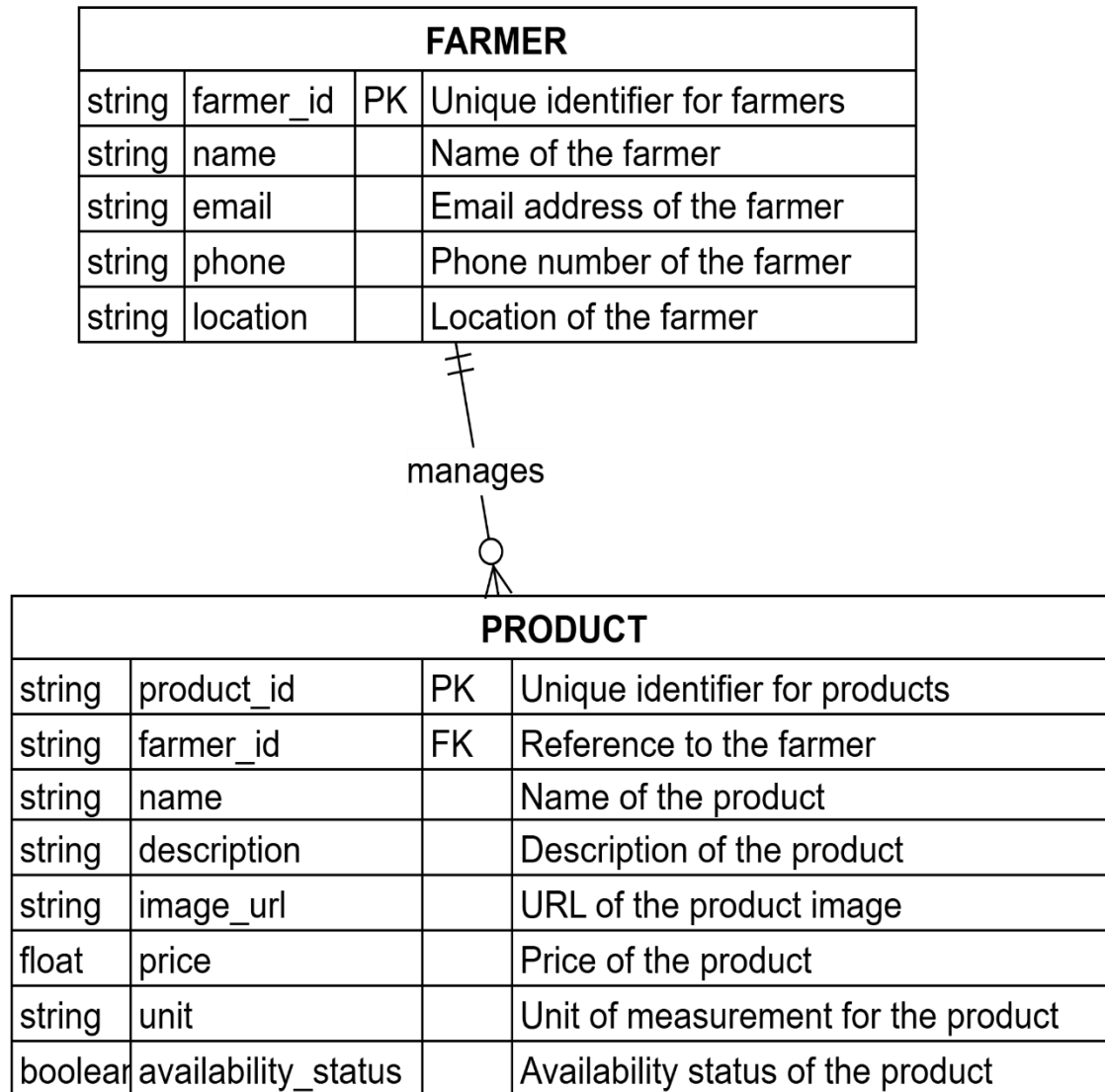


11.4 E-R Diagram

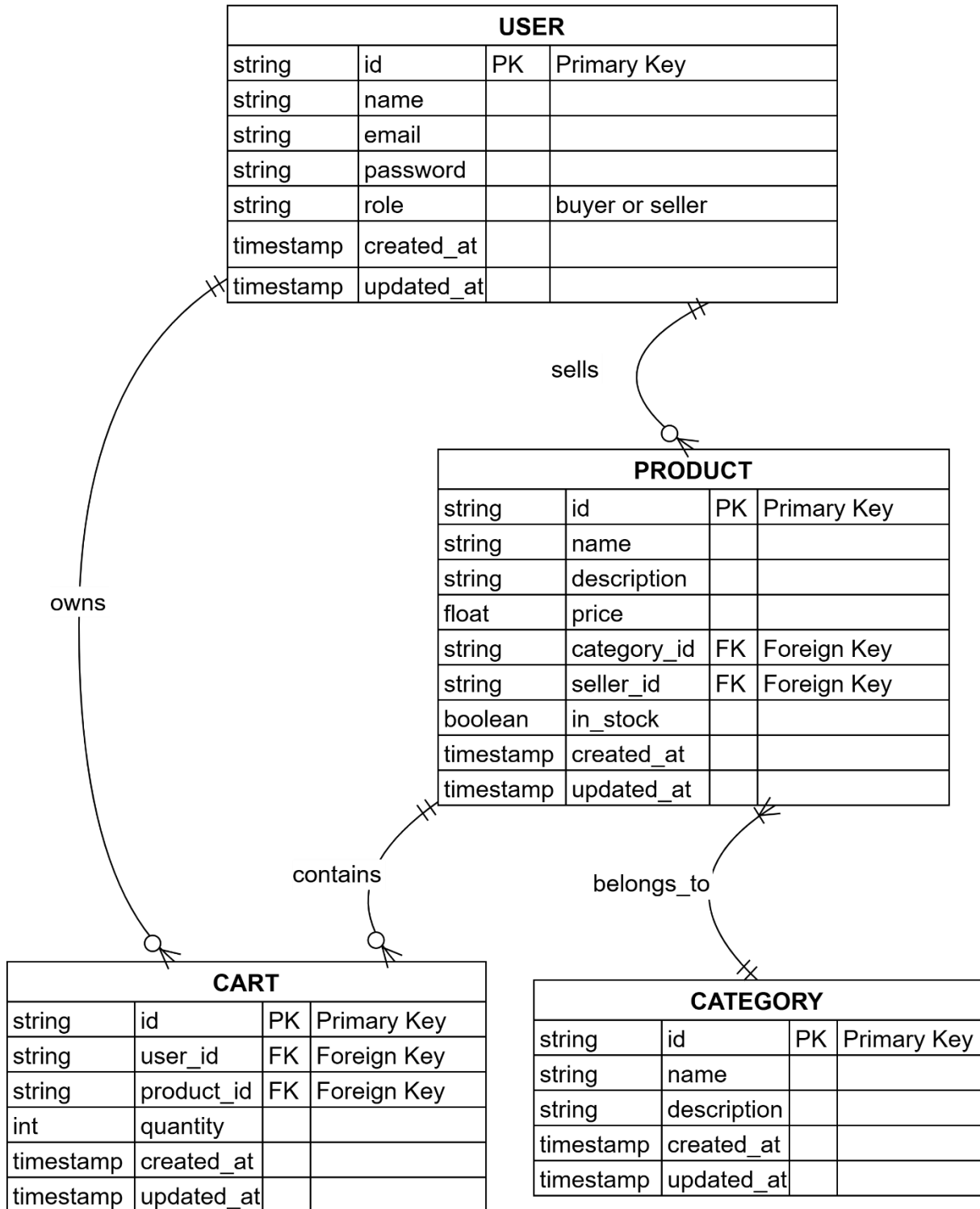
MODULE 1:



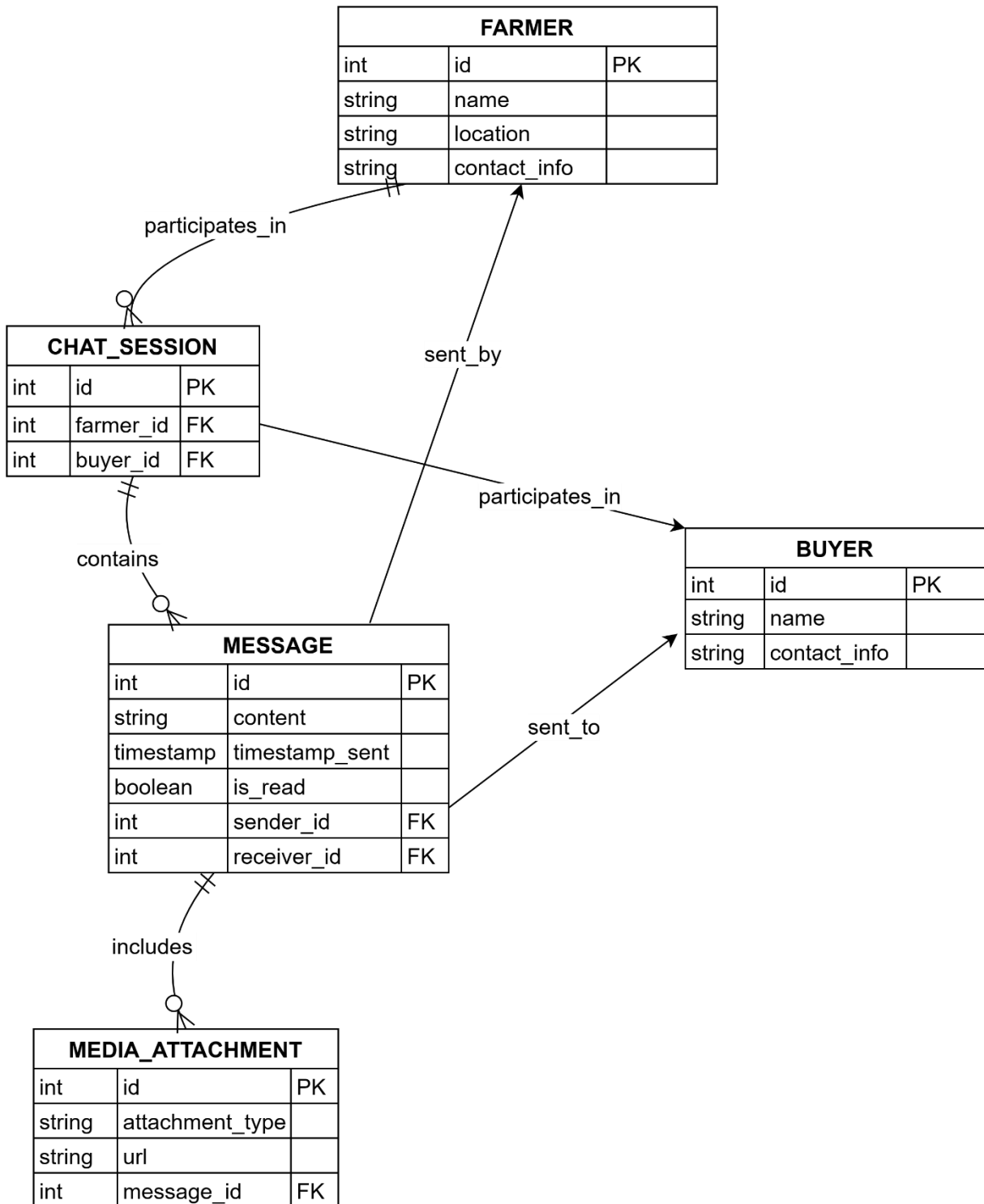
MODULE 2:



MODULE 3:

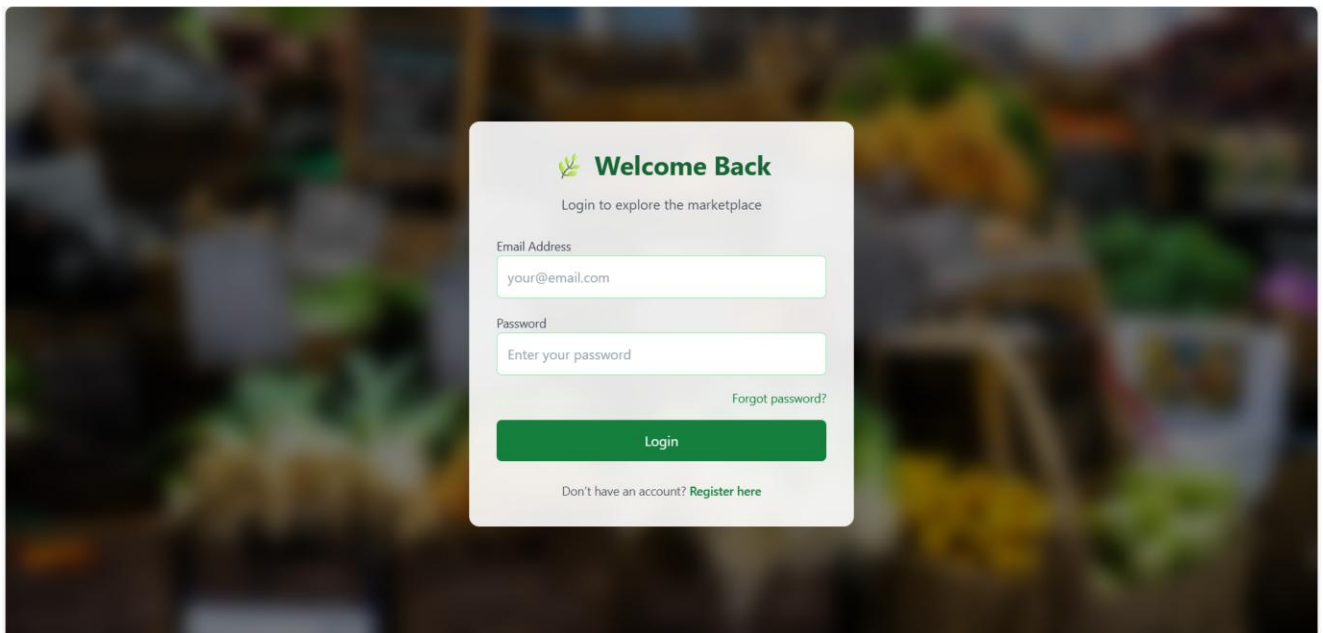
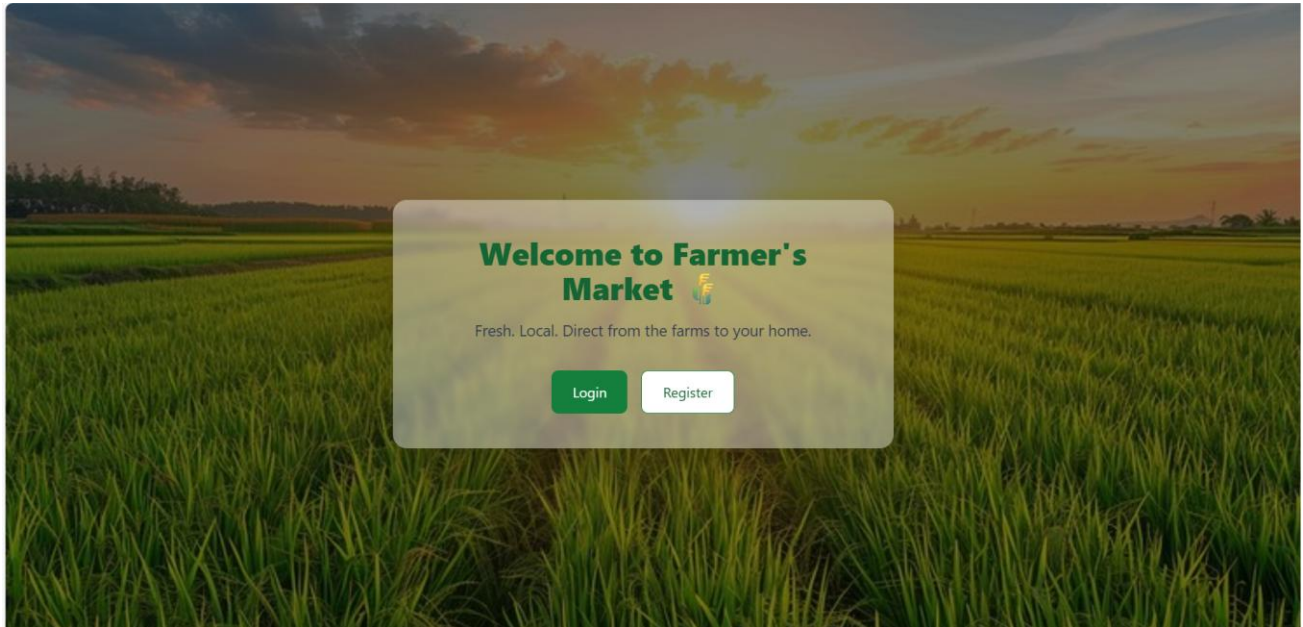


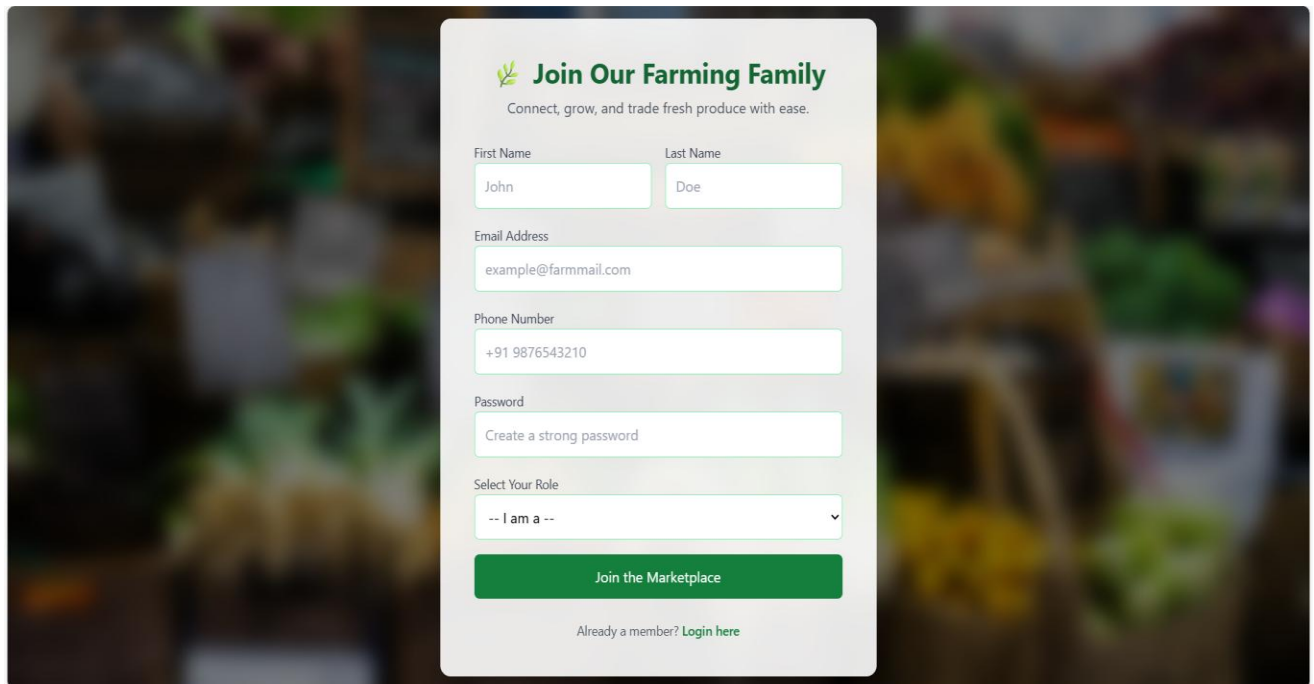
MODULE 4:



12. Screenshots

MODULE 1:





Join Our Farming Family
Connect, grow, and trade fresh produce with ease.

First Name: John Last Name: Doe

Email Address: example@farmmail.com

Phone Number: +91 9876543210

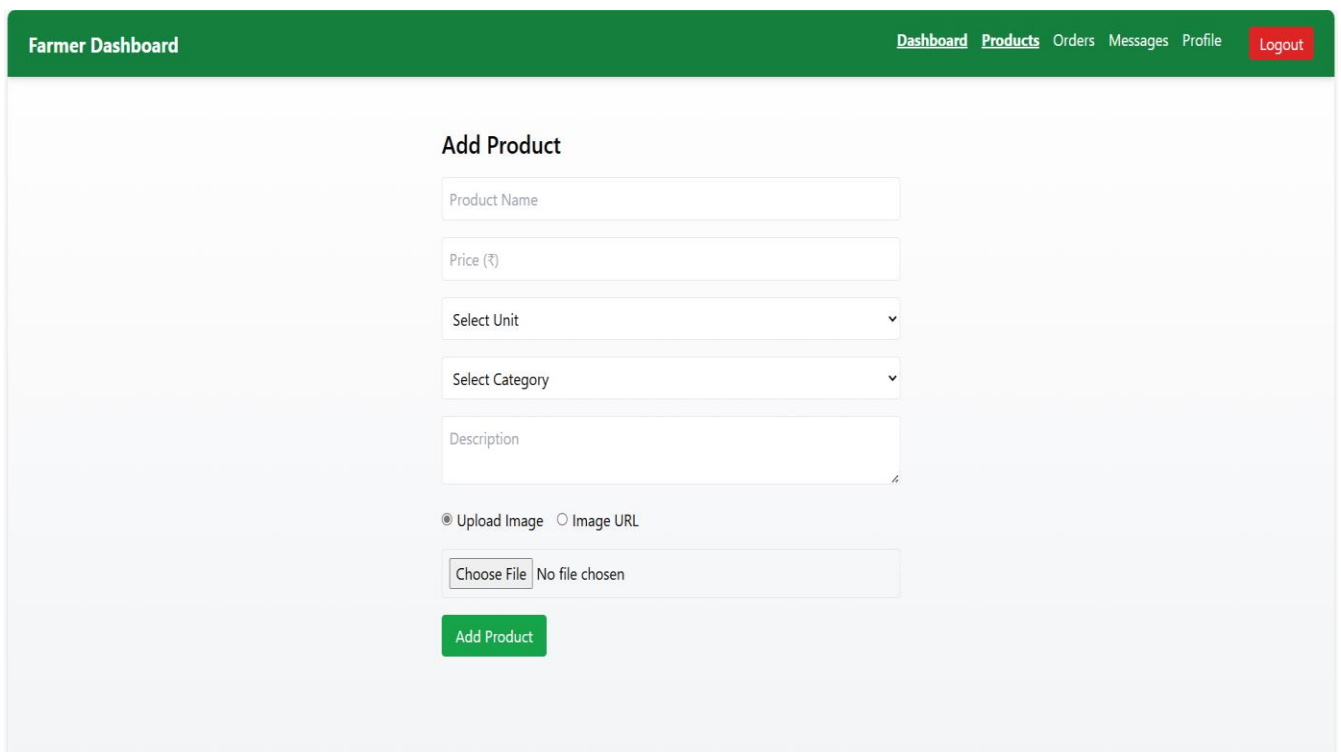
Password: Create a strong password

Select Your Role: -- I am a --

[Join the Marketplace](#)

Already a member? [Login here](#)

MODULE 2:



Farmer Dashboard [Dashboard](#) [Products](#) [Orders](#) [Messages](#) [Profile](#) [Logout](#)

Add Product

Product Name

Price (₹)

Select Unit

Select Category

Description

☒ Upload Image ☐ Image URL

Choose File No file chosen

[Add Product](#)


Farmer Dashboard

[Dashboard](#) [Products](#) [Orders](#) [Messages](#) [Profile](#) [Logout](#)

My Products

Filter by Category:


All



wheat
₹25
Crops
fresh crop

Edit

Delete



corn
₹20
Crops
fresh

Edit

Delete


MODULE 3:

FarmConnect

Search Results

Search for products...


Search



wheat
fresh crop
\$25

Add to Cart

View Details



corn
fresh
\$20

Add to Cart

View Details

Your Cart



wheat
Category: Crops
Unit: kg
Quantity: 1
₹25.00 / kg × 1

[Remove](#)

corn
Category: Crops
Unit: piece
Quantity: 2
₹20.00 / piece × 2

[Remove](#)**Total: ₹65.00**

MODULE 4:



Farmer Messages

☒ Use Dummy Chats

Conversations

Buyer A

Buyer B

Chat with Buyer A

Hi, is your spinach still available?
10:04 pm

Yes, I just harvested fresh ones this morning!
10:04 pm

Type your message...

[Send](#)

13. Conclusion

The Farmer Market-place Website aims to bridge the gap between farmers and consumers by providing a digital platform for seamless interaction, product listing, secure payments, and real-time price trend analysis. By leveraging modern web technologies such as the MERN stack (MongoDB, Express.js, React.js, and Node.js), this system empowers farmers to market their products directly, eliminating intermediaries and maximizing their profits. The integration of features like messaging, user authentication, and a dynamic search system enhances user experience and trust. Overall, this project not only supports digital transformation in agriculture but also promotes transparency, efficiency, and sustainability in the agricultural supply chain.

Through intuitive design and effective user interfaces, the website ensures ease of access for users with varying levels of digital literacy. It also supports a scalable backend architecture that can accommodate future growth and the addition of features such as logistics tracking, product rating systems, or AI-powered demand forecasting.

In summary, this project demonstrates how modern technology can contribute to the socio-economic upliftment of farmers by promoting digital inclusion. It not only serves as a tool for economic empowerment but also supports sustainable agricultural practices by facilitating direct market access, reducing waste, and improving price realization. The platform can be further enhanced and deployed at scale to support rural communities and boost agricultural commerce across the country.

14. Bibliography

- Suryawanshi, S. B., Patil, S. D., & Wankhade, K. D. (2020). *Smart Agriculture Using IoT, a Solution for Optimally Monitoring Farming Conditions. In 2020 International Conference on Inventive Computation Technologies (ICICT) (pp. 291–294). IEEE.*
<https://doi.org/10.1109/ICICT48043.2020.9112426>
- Chauhan, M. P., & Sharma, V. (2019). *Design and Development of an Online Platform for Farmers and Buyers. International Journal of Computer Applications, 178(11), 25–29.*
- Prakash, N., & Kumar, P. (2021). *Role of E-Marketing in Agricultural Business in India. International Journal of Research and Analytical Reviews (IJRAR), 8(1), 102–106.*
- Kumar, S., & Singh, R. (2020). *Digital Agriculture: E-Mandi and the Farmer's Marketplace. Journal of Rural Development and Agriculture, 5(2), 45–52.*
- FAO & ITU. (2016). *E-Agriculture Strategy Guide: Piloted in Asia-Pacific countries. Food and Agriculture.*