

MAR ATHANASIOUS COLLEGE OF ENGINEERING
(Affiliated to APJ Abdul Kalam Technological University, TVM)
KOTHAMANGALAM



Department of Computer Applications

Mini Project Report

MILK STATION

Done by

Rejitha Ramesh

Reg No: MAC20MCA-2017

Under the guidance of
Prof Elizabeth Poullose

2020-2022

MAR ATHANASIUS COLLEGE OF ENGINEERING
(Affiliated to APJ Abdul Kalam Technological University, TVM)
KOTHAMANGALAM

CERTIFICATE



MILK STATION

Certified that this is the bonafide record of project work done by

Rejitha Ramesh
Reg No: MAC20MCA-2017

During the academic year 2020-2022, in partial fulfilment of requirements for
award of the degree,

Master of Computer Applications
Of
APJ Abdul Kalam Technological University
Thiruvananthapuram

Faculty Guide
Prof. Elizabeth Poullose

Head of the Department
Prof. Biju Skaria

Project Coordinator
Prof. Nisha Markose

Internal Examiner

1. Prof. Nisha Markose
2. Prof. Liji Jose P

ACKNOWLEDGEMENT

First and foremost, I thank God Almighty for his divine grace and blessings in making all this possible. May he continue to lead me in the years to come.

I would like to express my special gratitude and thanks to Mini project guide Prof. Elizabeth poulose, Assistant Professor, Department of Computer Applications for her guidance and constant supervision as well as for providing necessary information regarding the Mini project & also for her support.

I am highly indebted to our project coordinators Prof. Beena Jacob, Associate Professor and Prof. Nisha Markose, Associate Professor, Department of Computer Applications for their guidance and support.

I am also grateful to Prof. Biju Skaria, Head of the Department, Department of the Computer Applications, for his valuable guidance as well as timely advice which helped me a lot during preparation of the project.

I profusely thank other Professors in the department and all other staffs of MACE, for their guidance and inspirations throughout my course of study. No words can express my humble gratitude to my beloved parents who have been guiding me in all walks of my journey. My thanks and appreciations also go to my friends and people who have willingly helped me out with their abilities.

CONTENTS

1	Introduction	7
2	System Analysis	10
2.1	Module Description	10
2.2	Feasibility Analysis	11
2.2.1	Technical Feasibility	11
2.2.2	Economical Feasibility	11
2.2.3	Operational Feasibility.....	12
2.3	System Environment	12
2.3.1	Software Environment	12
2.3.2	Hardware Environment	15
2.4	Actors and Roles	15
2.5	Use Case Model	16
2.5.1	Business Rules.....	16
2.5.2	Use Cases	17
2.5.3	Use Case Diagram	18
2.6	Activity Diagram	19
2.7	Sequence Diagram	20
2.8	Class Diagram.....	23
3	System Design	24
3.1	Database Design	24
3.2	UI Design	28
4	Testing and Implementation	34
4.1	Testing.....	34
4.2	Implementation	37
4.3	Git History	38

5 Conclusion	39
6 Scope for Future Enhancements	41
7 Appendix	41
7.1 Minimum Software Requirements.....	42
7.2 Minimum Hardware Requirements	42
8 References	43

LIST OF FIGURES

2.5.3	Use case diagram	18
2.6	Activity diagram	19
2.7.1	Sequence diagram for registration	20
2.7.2	Sequence diagram for dairy collection	21
2.7.3	Sequence diagram for purchase product	21
2.7.3	Sequence diagram for payment	22
2.8	Class diagram	23
3.2.1	Home page	28
3.2.2	Farmer registration	29
3.2.3	Login page	29
3.2.4	Supply milk	30
3.2.5	Sales details	30
3.2.6	Payment for farmer	31
3.2.7	Purchase details	31
3.2.8	Add product	32
3.2.9	Request milk for customer	32
3.2.10	Milk products	33
3.2.11	Report	33
4.3	Git history	38

1. INTRODUCTION

The goal of **MILK STATION** is to encourage dairy industries. The **MILK STATION** is a software application to maintain day-to-day transactions in a milk distribution. This software help to register all the suppliers, buyer details, Sales details etc. **MILK STATION** is an application designed to manage activities related to people, their daily work such as collecting milk from members, sales to the customer and all the related processes. It became tough for dairy owners to manage all dairy work manually. To reduce manual work, a **MILK STATION** can help to make day-to-day dairy related activities easier. The whole process admin can manage with one app. Milk software also assists in reporting, and accounting.

MILK STATION is been developed with the aim of providing a tool, which will prove efficient enough to manage the customer and daily entries of milk, save the organization's time as well as resources .This project provides new facility to the user, and helps them to save their time and money.

Dairy farm needs a **MILK STATION**. It can ease a variety of a dairy process like member management, report analysis, accounting and finance, milk supply. This website will be showing the details of employees in the milk station information system, it will show quality of milk sold in the dairy, it will help them to know their frequenters customers and their information details in their dairy database. It will help to know daily sale each day enable to access the business progress, It will enable them to their know suppliers who brought for them in daily time.

This Project is based on the concept of managing dairy products and their records. Design of this is so simple that the user won't find difficulties while working on it. This project helps the user for easy management of transaction activities. This web application helps to register all the suppliers, purchase, Sales details etc. This project deals with the purchase and sale of milk by making records in the database. The milk station is a comprehensive approach to manage the Dairy products collection, making payments. The admin of the website can

manage employees and payroll. This is software developed with the purpose of helping the dairy farm mitigate their losses and increase their productivity.

- Facilitate easily maintenance
- Maintain Daily inventory reports and records of Members
- Quick access to all records
- Reduce manual work
- Prevent and reduce human error
- Help with the automatic registration of dairy people
- Allow multiple users to use the software at the same time
- Helping the dairy owner to mitigate the losses and increase productivity

The existing system requires more time for processing. It requires more critical work. This system is more error prone and difficult to maintain. It is costly. Immediate response to the queries is difficult and time consuming. More men power needed.

Productivity of Indian dairy animals is lower compared to other country. It is due to the lack of scientific livestock feeding practice, inadequacy and unavailability of livestock healthcare. Some other problems associated with the system are improper milk marketing facilities and uncertain price of milk for producers, lack of infrastructure for milk collection, transportation, processing, lack of veterinary and extension services, milk losses due to lack of cold chain facilities and lack of clean milk production practices.

The tools used for the development of the project include Html, CSS, Bootstrap and Javascript at the Front End. PHP and MySQL is used as backend. Microsoft Visual Studio Code was used to write the code. The Editor provides all the settings to write your Html, CSS, Javascript. Xampp distribution which offers the MySQL database server is used. XAMPP only offers MySQL (Database Server) & Apache (Webserver) in one setup and you can manage them with the xampp starter. The MySQL database server allows connecting to the

phpMyAdmin. phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. PhpMyAdmin supports a wide range of operations on MySQL.

The proposed system will be showing the details of employees in milk station information system, it will show quality of milk selling in the dairy, it will help them to know their frequent customers and their information details in their dairy database. It will be help to know how much they sell each day, enable access to the business progress, it will enable them to know their suppliers who brought from them in regularly. In addition, this system will provide central database for everything being sold and simple management of information being processed.

Advantages of Proposed System

- The initial investment in dairy farming business is low in comparison to other Industry.
- It is environment-friendly.
- The demand for milk product is increasing rapidly.
- This system required less time for completion of any work.
- This system is reduces the chances of error.
- This system should work smoothly and very fast.
- It saves time and manpower.
- The system is user friendly and anyone having computer knowledge can handle it easily.

2. SYSTEM ANALYSIS

2.1. Module Description

1. Registration:

User can register into the system providing their basic details. After registration the admin needs to verify the users.

2. Milk Collection:

The system also monitors the milk collection from farmers. The farmer gives the milk to the vendor, the vendor will have to make the payments to the farmers according to the milk purchases.

3. Milk Sale & Milk Products:

The collected milk will be sold to the customers. The vendor will decide the price based on the thickness of the milk.

4. Purchase Products:

The customers can purchase milk and milk products. After making the payments, the receipt is given to the customers.

5. Payments:

This module asks for payment after purchasing milk products to be done online from the customer. It also provides payment to the farmer after milk is supplied. The payments are made as per the purchases, sales and once the vendor updates the purchase details from the farmers, he makes the payments, and the payment details are being updated for future use.

6. Report:

The admin gets all the reports from the sales report, payment report and the receipt. The print-out can be generated based on the sales and payment.

2.2. Feasibility Analysis

Feasibility study is an assessment of the practicality of a proposed project or system. A feasibility study aims to objectively and rationally uncover the strengths and weaknesses of an existing business or proposed venture, opportunities and threats present in the natural environment, the resources required to carry through, and ultimately the prospects for success. In its simplest terms, the two criteria to judge feasibility are cost required and value to be attained. The various types of feasibilities that are to be determined are:

- Technical Feasibility
- Economical Feasibility
- Operational Feasibility

2.2.1. Technical Feasibility

Technical feasibility is used to determine the requirement of technologies for the current system. This assessment focuses on the technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. Technical feasibility also involves the evaluation of the hardware, software, and other technical requirements of the proposed system.

The web application uses PHP language which is user friendly and efficient along with MySQL. Also the tools, operating system and language used in this system is compatible with the existing one. So the project is technically feasible.

2.2.2. Economic Feasibility

Economic feasibility is a kind of cost-benefit analysis of the examined project, which assesses whether it is possible to implement it. This term means the assessment and analysis of a project's potential to support the decision-making process by objectively and rationally identifying its strengths, weaknesses, opportunities and risks associated with it, the resources that will be needed to implement the project, and an assessment of its chances of success. It is

used to determine the benefits and savings that are expected from the proposed system and compare them with the costs. If benefits outweigh costs, then the decision is made to design and implement the system.

2.2.3. Operational Feasibility

Operational feasibility is the measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. This feasibility test asks if the system will work when it is developed and installed. This assessment involves undertaking a study to analyze and determine whether and how well the organization's needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

2.3. System Environment

System environment specifies the hardware and software configuration of the new system. Regardless of how the requirement phase proceeds, it ultimately ends with the software requirement specification. A good SRS contains all the system requirements to a level of detail sufficient to enable designers to design a system that satisfies those requirements. The system specified in the SRS will assist the potential users to determine if the system meets their needs or how the system must be modified to meet their needs.

2.3.1. Software Environment

Tool: As a Text editor we use Visual Studio Code. It is an integrated development environment made by Microsoft for Windows, Linux and macOS. As we are developing the project using Html, CSS, and Javascript , the Visual Studio provide rich library support for these.

Operating system: Windows 7/8/10 or latest

Front end: HTML, CSS, JavaScript

Back end: PHP, MySQL

Environment Tools: XAMPP Server

- **PHP**

The term PHP is an acronym for Hypertext Preprocessor. It is a server-side scripting language that is used for web development. It can be easily embedded with HTML files. HTML codes can also be written in a PHP file. The PHP codes are executed on the server-side whereas HTML codes are directly executed on the browser.

- **HTML**

The HyperText Markup Language or HTML is the standard markup language for creating Web pages. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets. HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages.

- **CSS**

Cascading Style Sheets (CSS) is used to style and layout web pages. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility; provide more Department of Computer Application, MACE 13 flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate.

- **JavaScript**

JavaScript is a programming language that adds interactivity to your websites. JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard. It has dynamic typing, prototype-based object-orientation, and first-class functions. It is multi-paradigm, supporting event-driven, functional, and imperative programming styles.

- **MySQL**

MySQL is an open-source relational database management system. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database.

- **Visual Studio**

Microsoft Visual Studio is an integrated development environment from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. A fully-featured, extensible, free IDE for creating modern applications for Android, iOS, Windows, as well as web applications and cloud services.

- **XAMPP Server**

XAMPP Server is a Web development platform on Windows that allows you to create dynamic Web applications with Apache2, PHP, MySQL and MariaDB.

- **Github**

Git is an open-source version control system that was started by Linus Torvalds the same person who created Linux. Git is similar to other version control systems Subversion, CVS, and Mercurial to name a few. Version control systems keep these revisions straight, storing the modifications in a central repository. This allows developers to easily collaborate, as they can download a new version of the software, make changes, and upload the newest revision. Every developer can see these new changes, download them, and contribute. Git is the preferred version control system of most developers, since it has multiple advantages over the other systems available. It stores file changes more efficiently and ensures file integrity better.

2.2.4. Hardware Environment

Processor: Ryzen Core i3 processor or more

More Main memory: 8 GB RAM

Hard disk: 256 GB SSD

2.4. Actors and Roles

The main actors involved are:

- Admin
- Farmer
- Vendor
- Customer

- **Admin**

The admin has the unique login that completely controls the working. Admin will verify the reports entered by the vendor. Admin provides username and password to each of the vendor after their verification. Admin can see the users and verify them. He can also view the reports. The admin can add or remove users whenever required. The admin gets all the reports from the sales report, payment report and the receipt.

- **Farmer**

Farmers can manage the profile. Farmers can login to the website and sell his products to the vendor. The farmer gives the milk to the vendor, the vendor will have to make the payments to the farmers according to the milk purchases. Farmers will receive the payment depending on the quantity of milk provided. After making the payments, the receipt is given to the farmers.

- **Vendor**

The vendor manages the profile. The vendor will Collect milk and update customer account. Vendor manages the payment. The farmer gives the milk to the vendor, the vendor will have to make the payments to the farmers according to the milk purchases. The payments are made as per the purchases, sales and once the vendor updates the purchase details from the farmers, he makes the payments, and the payment details are being updated for future use. After making the payments, the receipt is given to the customers and farmers.

- **Customer**

A customer can manage the profile. Customer can view milk product details like milk quantity, milk price etc. The customer can see the total monthly or yearly milk details. The customer can view all payment details. After making the payments, the receipt is given to the customers.

2.5. Use Case Model

A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artefacts or classes, in order to better understand, alter, maintain, or document information about the system.

2.5.1. Business Rules

Once the users register into the system their accounts need to be verified by the admin. The farmer needs to be verified by the vendor. Then only the farmer can supply milk. The payment can be initiated only when the purchase is done.

2.5.2. Use Cases

1. Registration

User can register into the system providing their basic details. After registration the admin needs to verify the users.

2. Dairy Collection

The system also monitors the milk collection from farmers. The farmer gives the milk to the vendor, the vendor will have to make the payments to the farmers according to the milk purchases.

3. Milk Products

The collected milk will be sold to the customers. The vendor will decide the price based on the thickness of the milk.

4. Purchase Products

The customers can purchase milk and milk products. After making the payments, the receipt is given to the customers.

5. Payments

This module asks for payment after purchasing milk products to be done online from the customer. It also provides payment to the farmer after milk is supplied. The payments are made as per the purchases, sales and once the vendor updates the purchase details from the farmers, he makes the payments, and the payment details are being updated for future use.

6. Report

The admin gets all the reports from the sales report, payment report and the receipt. The print-out can be generated based on the sales and payment.

2.5.3. Use Case Diagram



Figure 2.5.3: Use Case Diagram

2.6. Activity Diagram

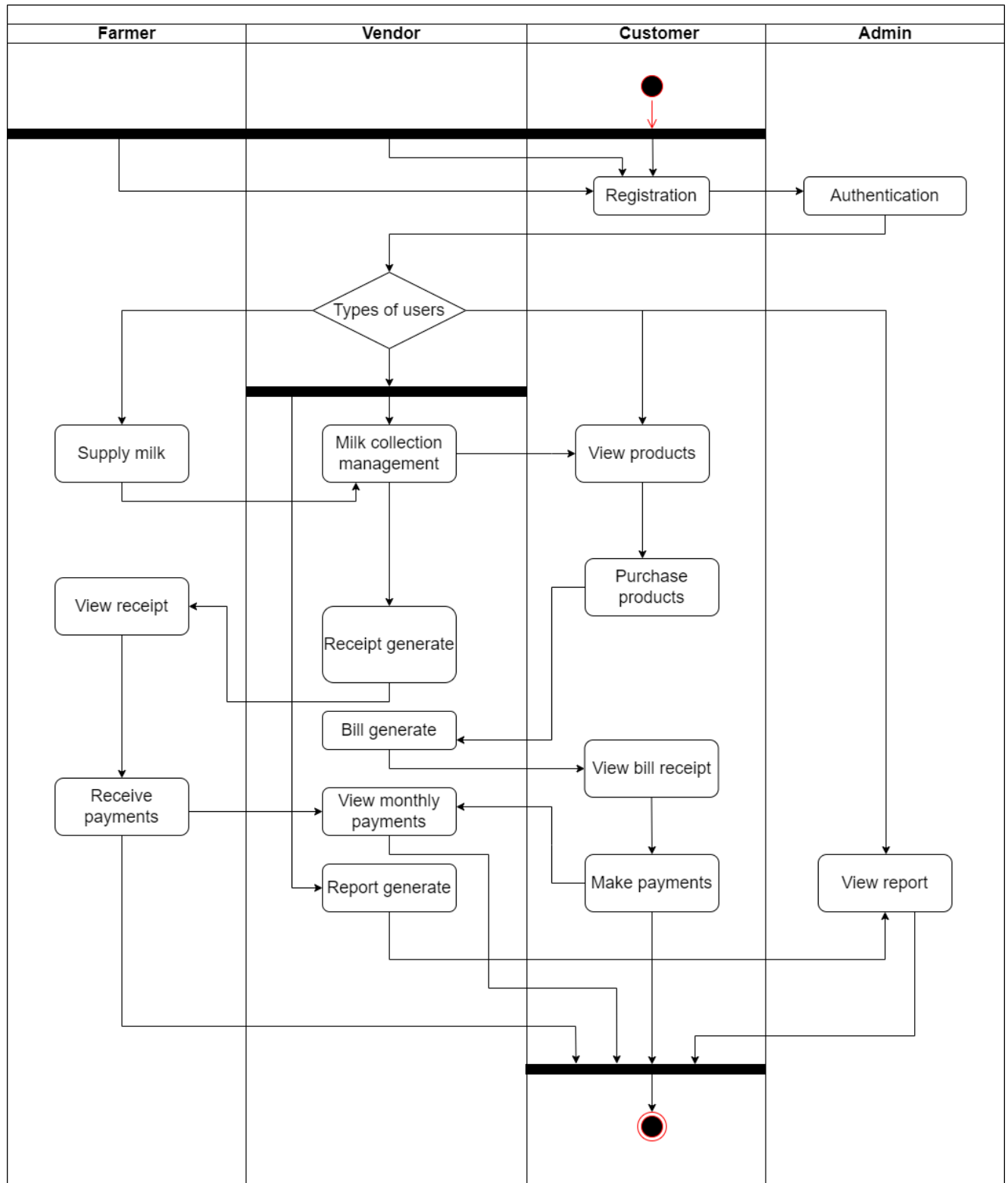


Figure 2.6: Activity Diagram

2.7. Sequence Diagram

1. Registration

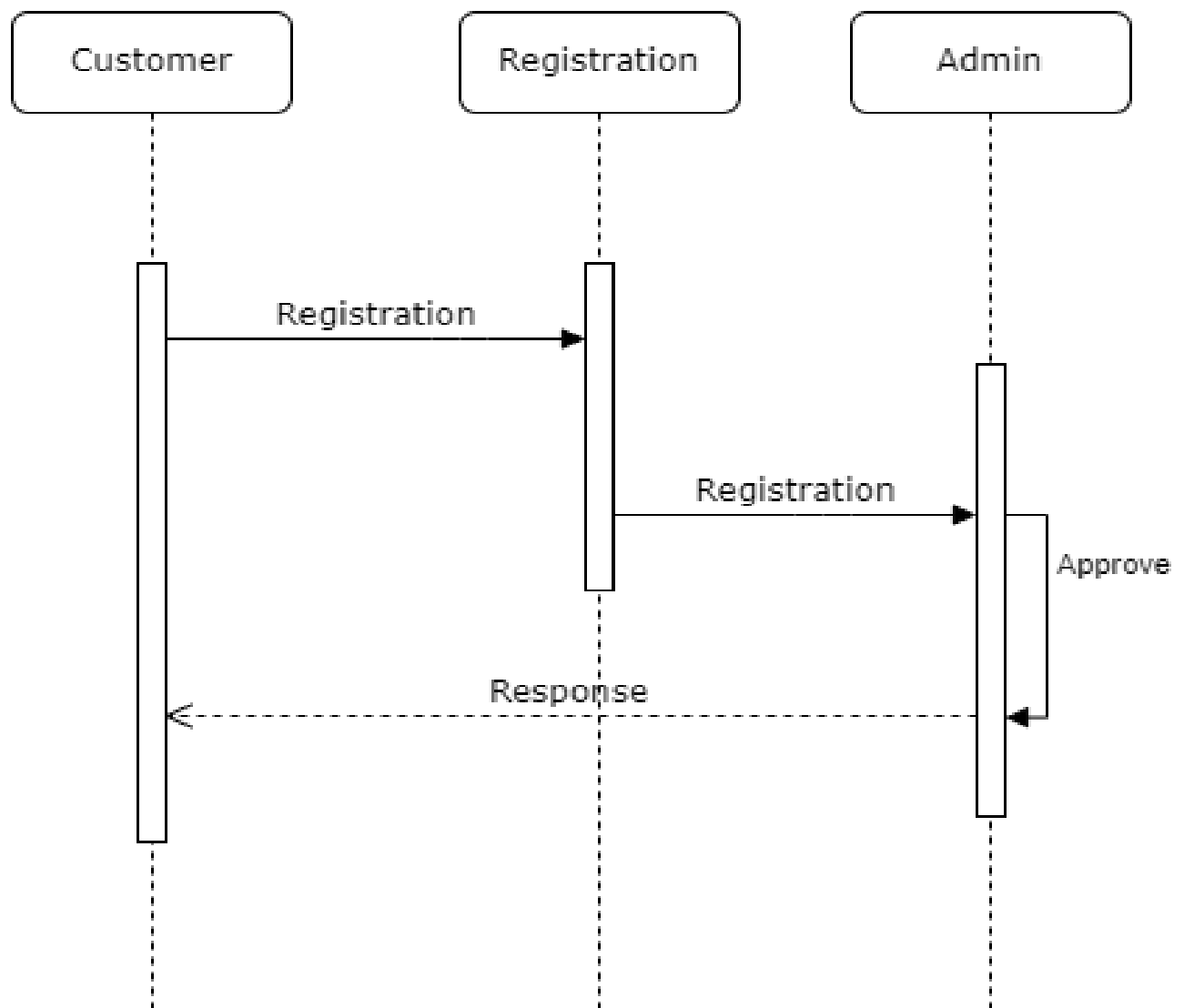


Figure 2.7.1: Sequence Diagram for Registration

2. Dairy Collection

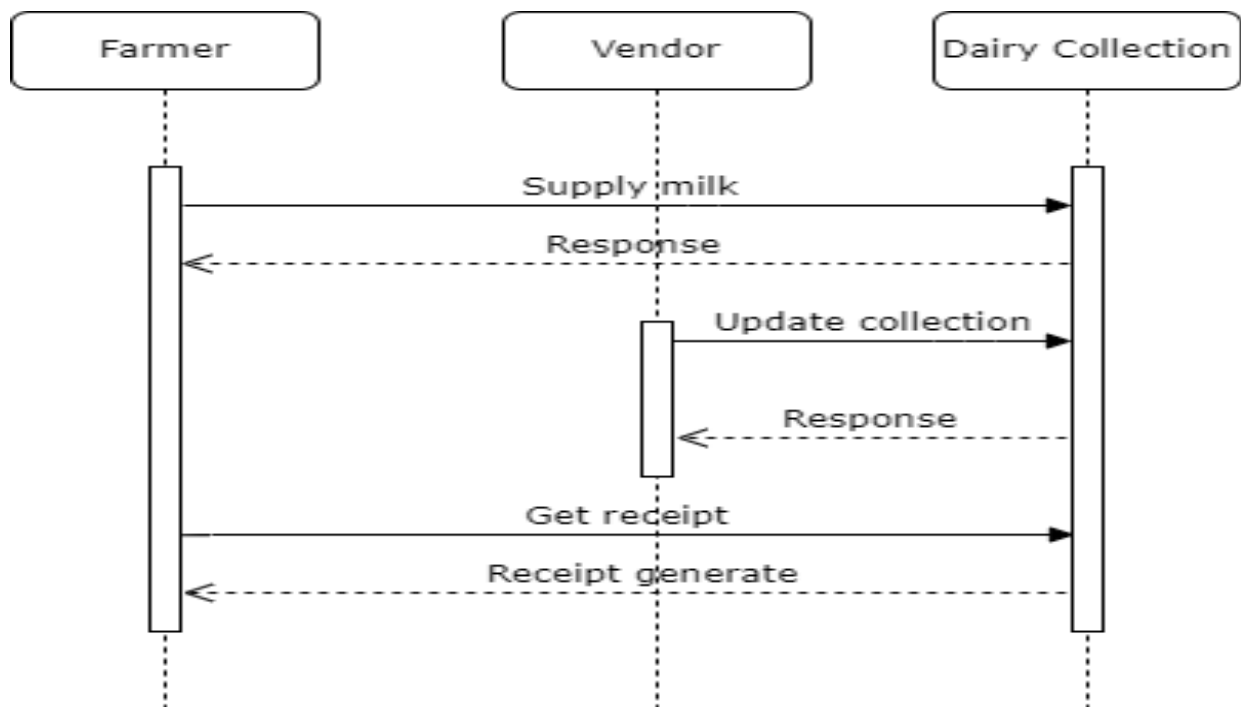


Figure 2.7.2: Sequence Diagram for Dairy Collection

3. Purchase Product

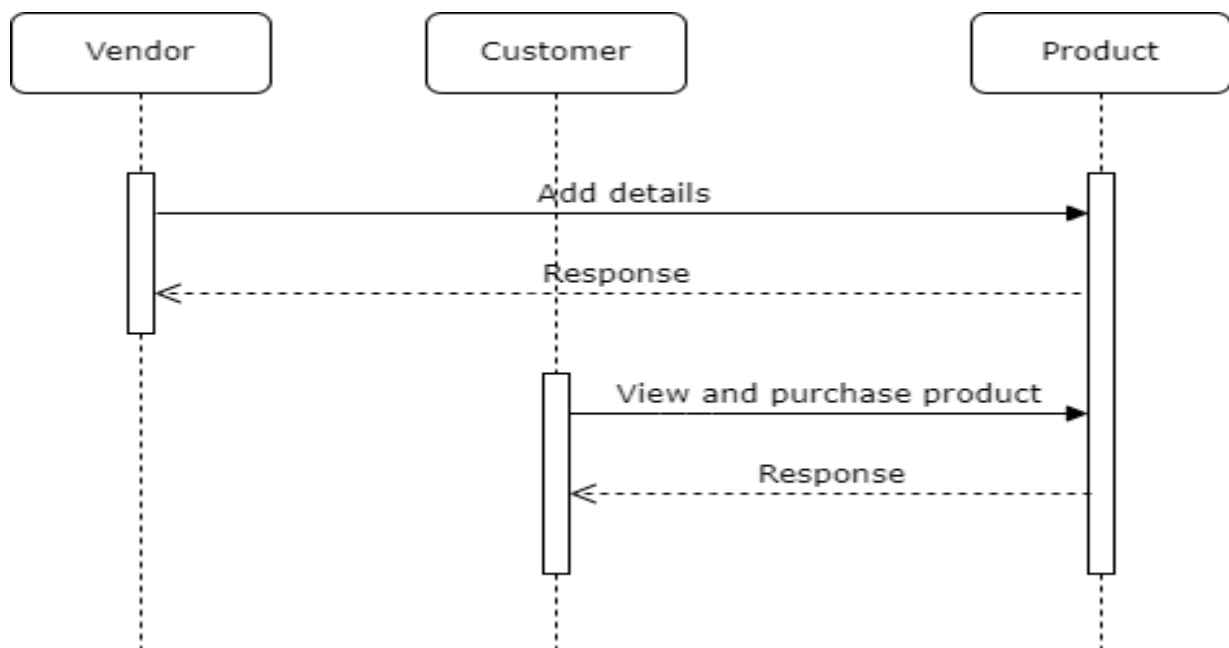


Figure 2.7.3: Sequence Diagram for Product Purchase

4. Payment

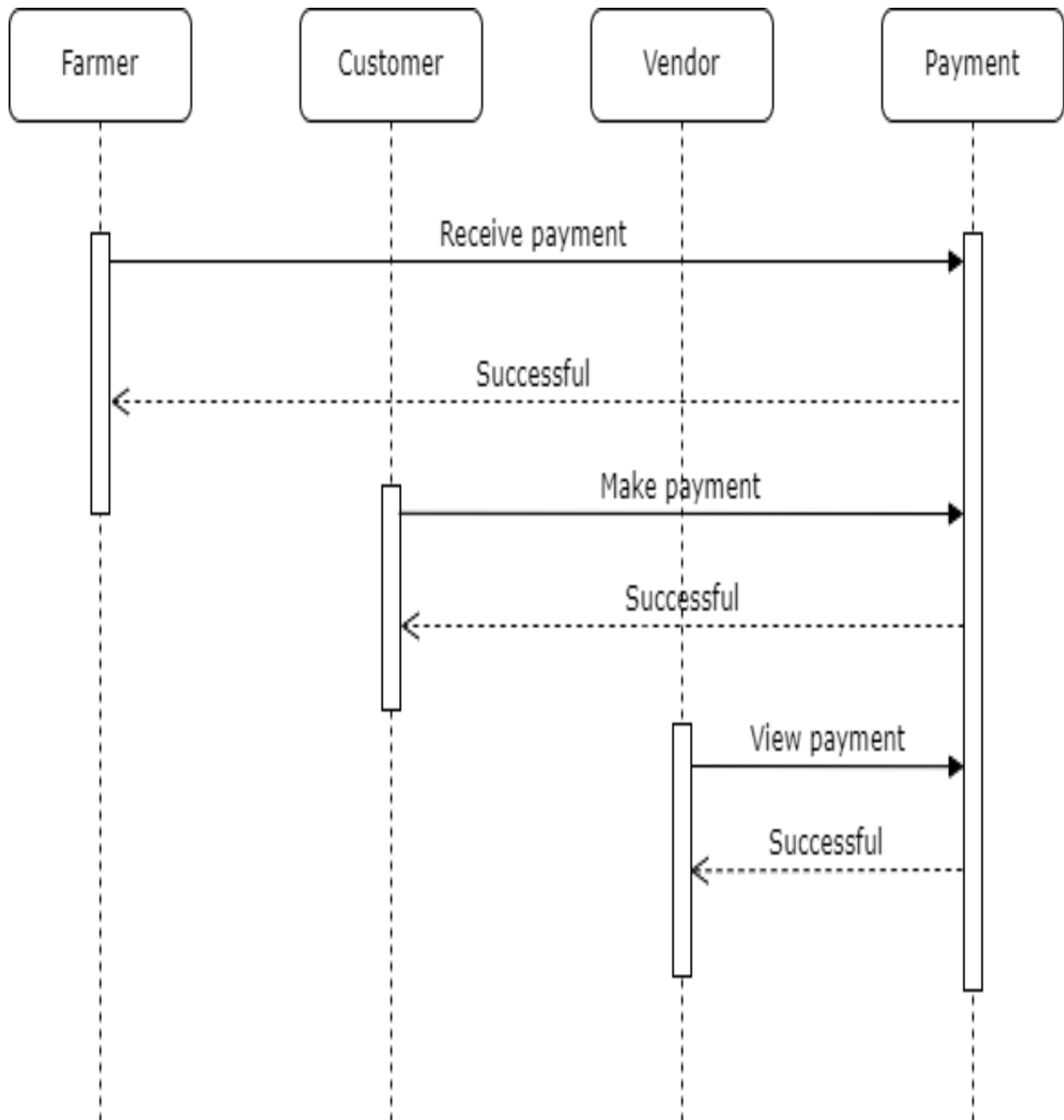


Figure 2.7.4: Sequence Diagram for Payment

2.8. Class Diagram

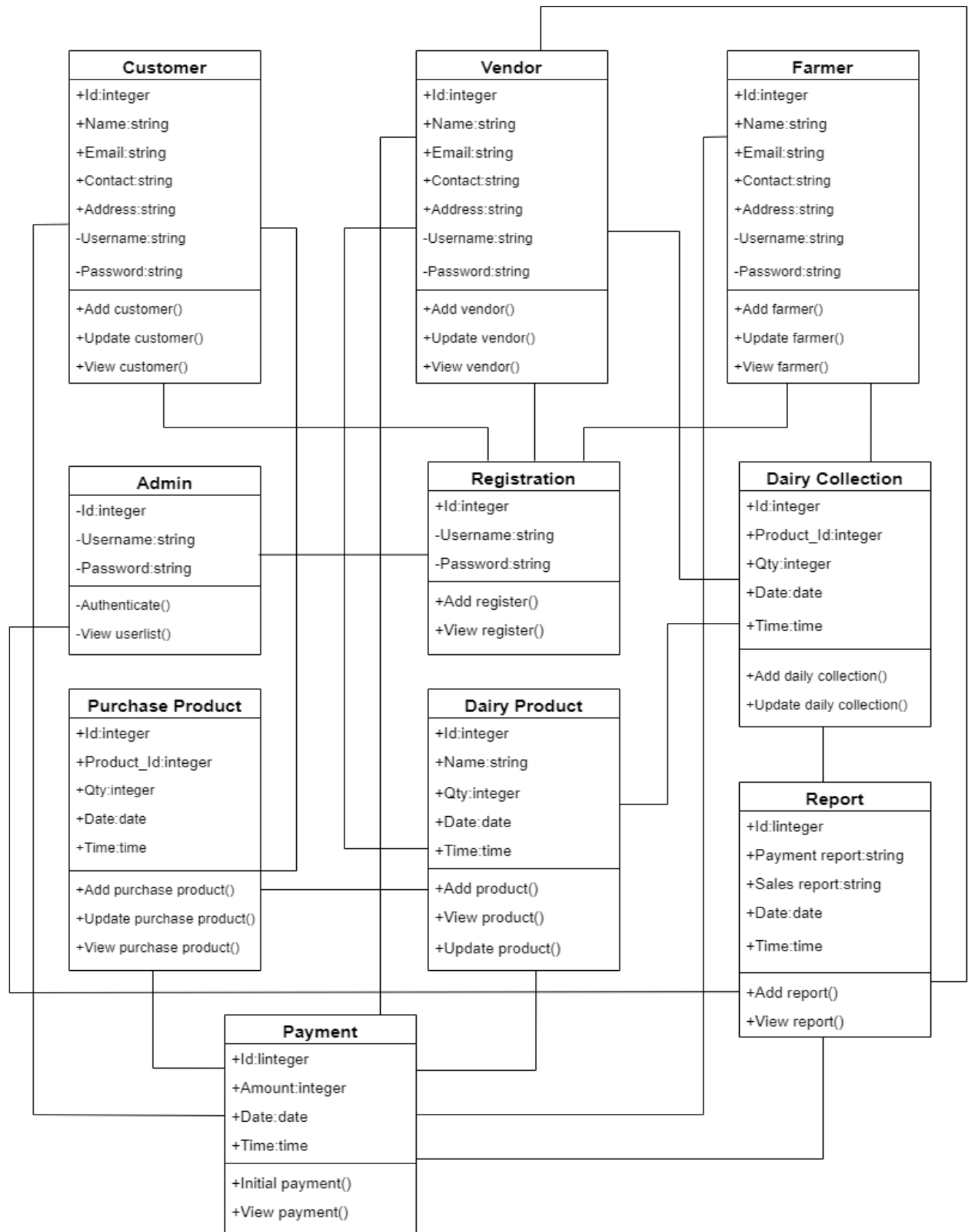


Figure 2.8: Class Diagram

3. SYSTEM DESIGN

3.1. Database Design

1) Table name: **Farmer**

Description: To store farmer details

Field name	Data type	Description	Constrains
Farmer_id	Int(20)	Id for farmer	Primary key
Name	varchar(100)	Name for farmer	Not null
Contact no	varchar(100)	Contact number of farmer	Not null
Email	Varchar(100)	Email for customer	Not null
Address	Varchar(100)	Address for farmer	Not null
Location_id	Int(20)	Id for location	Foreign key
Society no	Varchar(100)	Society number of farmer	Not null
Photo	Blob	Photo of farmer	Not null
Proof	Blob	Proof of farmer	Not null
Username	Varchar(100)	username for farmer	Not null
Password	Varchar(100)	password for farmer	Not null

2) Table name: **Vendor**

Description: To store vendor details

Field name	Data type	Description	Constrains
Vendor_id	Int(20)	Id for vendor	Primary key
Name	varchar(100)	Name for vendor	Not null
Contact no	varchar(100)	Contact number of vendor	Not null
Email	Varchar(100)	Email for vendor	Not null
Address	Varchar(100)	Address for vendor	Not null
Location_id	Int(20)	Id for location	Foreign key
License no	Varchar(100)	License number of vendor	Not null
Photo	Blob	Photo of vendor	Not null
Proof	Blob	Proof of vendor	Not null
Username	Varchar(100)	username for vendor	Not null
Password	Varchar(100)	password for vendor	Not null

3) Table name: **Customer**

Description: To store customer details

Field name	Data type	Description	Constrains
Customer_id	Int(20)	Id for customer	Primary key
Name	varchar(100)	Name for customer	Not null
Contact no	varchar(100)	Contact number of customer	Not null
Email	Varchar(100)	Email for customer	Not null
Address	varchar(100)	Address for customer	Not null
Location_id	Int(20)	Id for location	Foreign key
Photo	Blob	Photo of customer	Not null
Proof	Blob	Proof of customer	Not null
Username	varchar(100)	Username for customer	Not null
Password	varchar(100)	Password for customer	Not null

4) Table name: **Dairy Products**

Description: To store dairy products details

Field name	Data type	Description	Constrains
Product_id	Int(20)	Id for product	Primary key
Product_name	Varchar(100)	Name of product	Not null
Quantity	Varchar(100)	Quantity of product	Not null
Image	Blob	Image of product	Not null
Amount	Double	Amount of milk	Not null
Vendor_id	Int(20)	Id for Vendor	Foreign key
product type_id	Int(20)	Product type Id for product	Foreign key

5) Table name: **Product Type**

Description: To store product type details

Field name	Data type	Description	Constrains
Product type_id	Int(20)	Id for product type	Primary key
Product type name	Varchar(100)	Name for product type	Not null
Quantity	Varchar(100)	Quantity of product type	Not null
Image	Blob	Image of product type	Not null
Amount	Double	Amount of product type	Not null
Vendor_id	Int(20)	Id for Vendor	Foreign key

6) Table name: **Dairy Collection**

Description: To store dairy collection details

Field name	Data type	Description	Constrains
Dc_id	Int(20)	Id for dairy collection	Primary key
Quantity	Double	Quantity of milk	Not null
Date	Date	Date for dairy collection	Not null
Time	Time	Time for dairy collection	Not null
Amount	Double	Amount of milk	Not null
Vendor_id	Int(20)	Id for Vendor	Foreign key
Farmer_id	Int(20)	Id for farmer	Foreign key

7) Table name: **Location**

Description: To store location details

Field name	Data type	Description	Constrains
Location_id	Int(20)	Id for location	Primary key
Location	Varchar(100)	Name for location	Not null

8) Table name: **Purchase Products**

Description: To store purchase products details

Field name	Data type	Description	Constrains
Purchase_id	Int(20)	Id for purchase product	Primary key
Quantity	Double	Quantity of milk	Not null
Date	Date	Date for purchase product	Not null
Time	Time	Time for purchase product	Not null
Amount	Double	Amount of milk	Not null
Product_id	Int(20)	Id for product	Foreign key
Product type_id	Int(20)	Id for product type	Foreign key
Customer_id	Int(20)	Id for customer	Foreign key
Vendor_id	Int(20)	Id for Vendor	Foreign key

9) Table name: **Payment**

Description: To store payment details

Field name	Data type	Description	Constrains
Payment_id	Int(20)	Id for payment	Primary key
Amount	Double	Amount of milk	Not null
Date	Date	Date for product	Not null
Time	Time	Time for product	Not null
Dc_id	Int(20)	Id for dairy collection	Foreign key
Purchase_id	Int(20)	Id for purchase product	Foreign key
Customer_id	Int(20)	Id for customer	Foreign key
Vendor_id	Int(20)	Id for Vendor	Foreign key
Farmer_id	Int(20)	Id for farmer	Foreign key

10) Table name: **Report**

Description: To store report details

Field name	Data type	Description	Constrains
Report_id	Int(20)	Id for report	Primary key
Date	Date	Date for report	Not null
Time	Time	Time for report	Not null
Description	Varchar	Description about report	Not null
Payment_id	Int(20)	Id for payment	Foreign key
Vendor_id	Int(20)	Id for vendor	Foreign key
Admin_id	Int(20)	Id for admin	Foreign key
Dc_id	Int(20)	Id for dairy collection	Foreign key

3.2. UI Design

3.2.1. Home Page

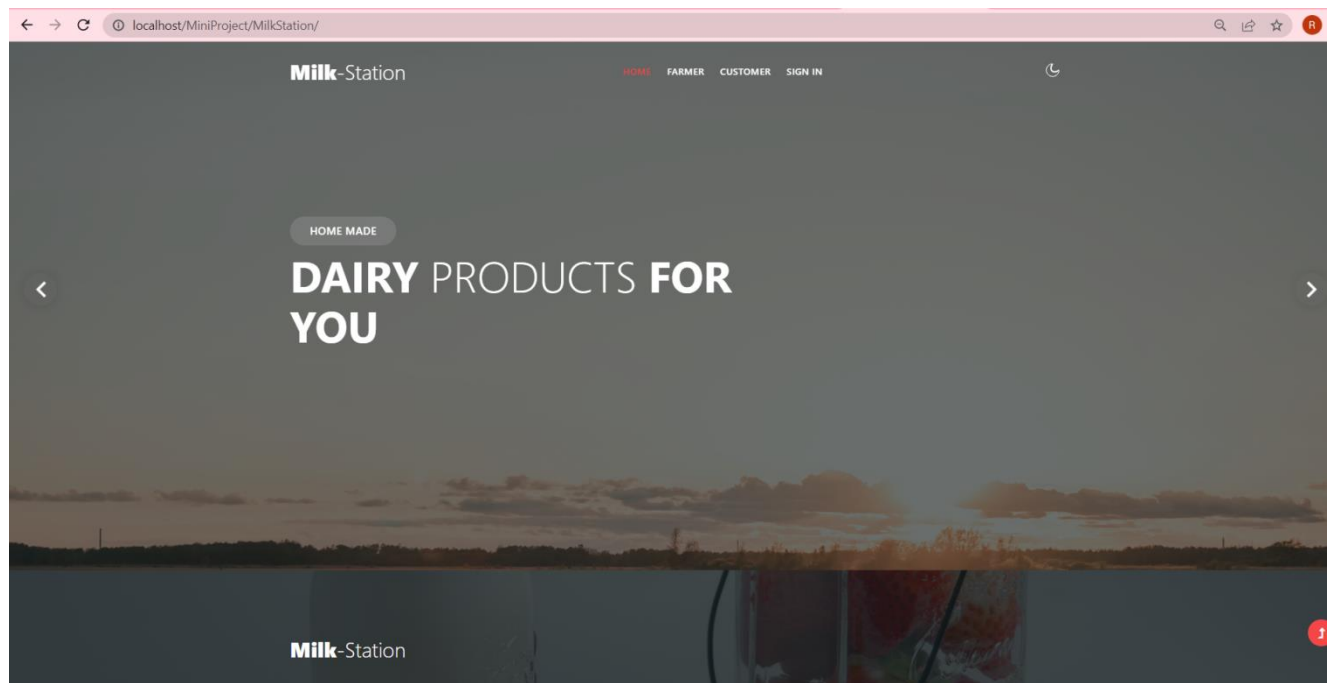
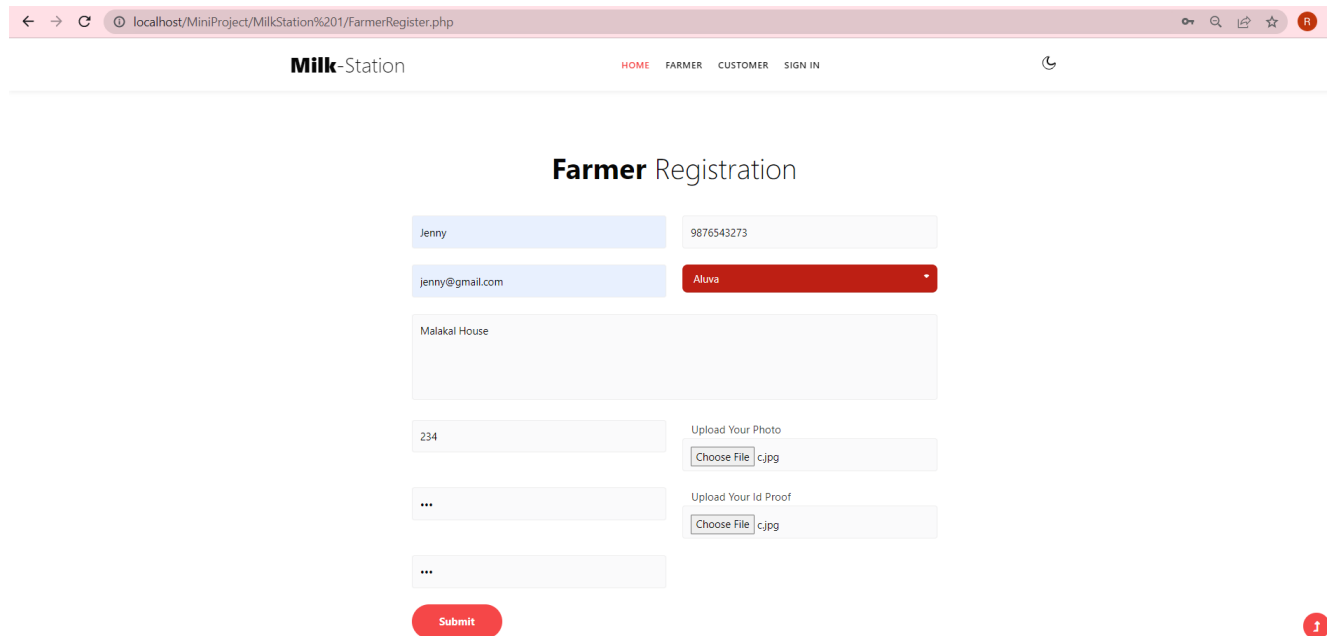


Figure 3.2.1: Home Page

3.2.2. Farmer Registration page

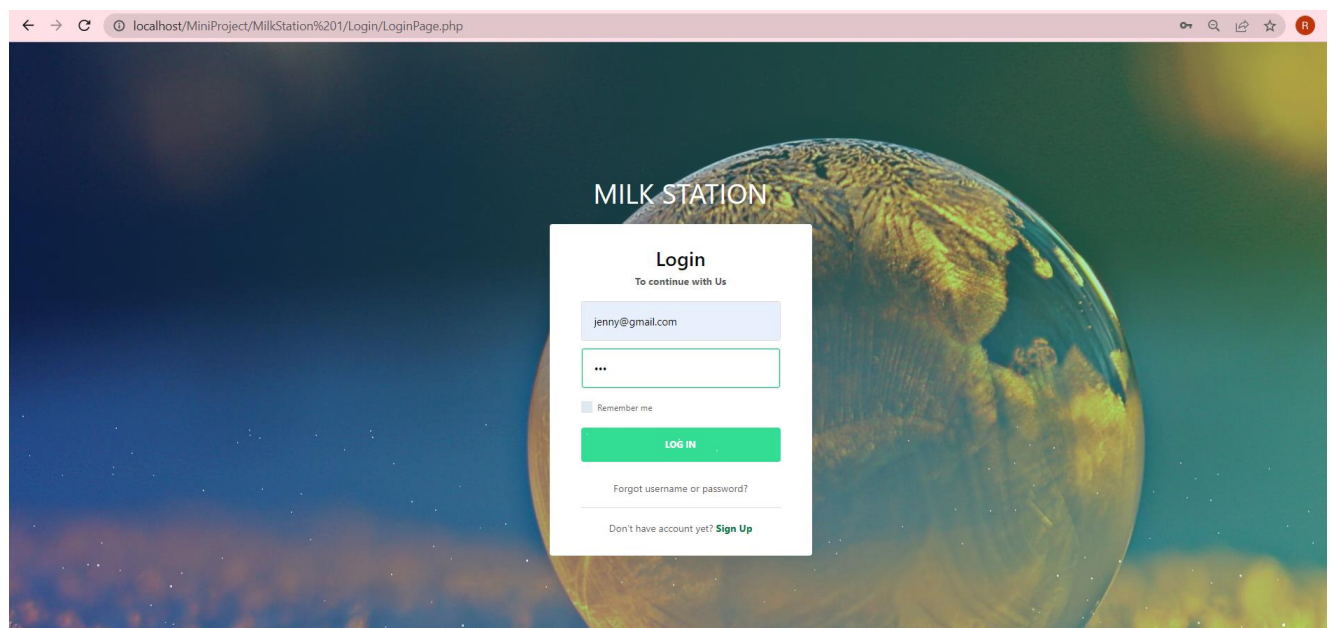


The screenshot shows a web browser window with the address bar displaying `localhost/MiniProject/MilkStation%201/FarmerRegister.php`. The page title is "Milk-Station". The navigation bar includes links for HOME, FARMER, CUSTOMER, and SIGN IN. The main heading is "Farmer Registration". The registration form contains the following fields and elements:

- First Name: Jenny
- Phone Number: 9876543273
- Email: jenny@gmail.com
- Gender: Aluva (selected from a dropdown menu)
- Address: Malakal House
- Age: 234
- Upload Your Photo: Choose File (cjpg)
- Upload Your Id Proof: Choose File (cjpg)
- Submit button

Figure 3.2.2: Farmer Registration Page

3.2.3. Login Page



The screenshot shows a web browser window with the address bar displaying `localhost/MiniProject/MilkStation%201/Login/LoginPage.php`. The page features a background image of a globe. The login form is centered and includes the following elements:

- Page Title: MILK STATION
- Section Title: Login
- Text: To continue with Us
- Username Field: jenny@gmail.com
- Password Field: ***
- Remember me checkbox
- LOG IN button
- Forgot username or password? link
- Don't have account yet? Sign Up link

Figure 3.2.3: Login Page

3.2.4. Supply Milk

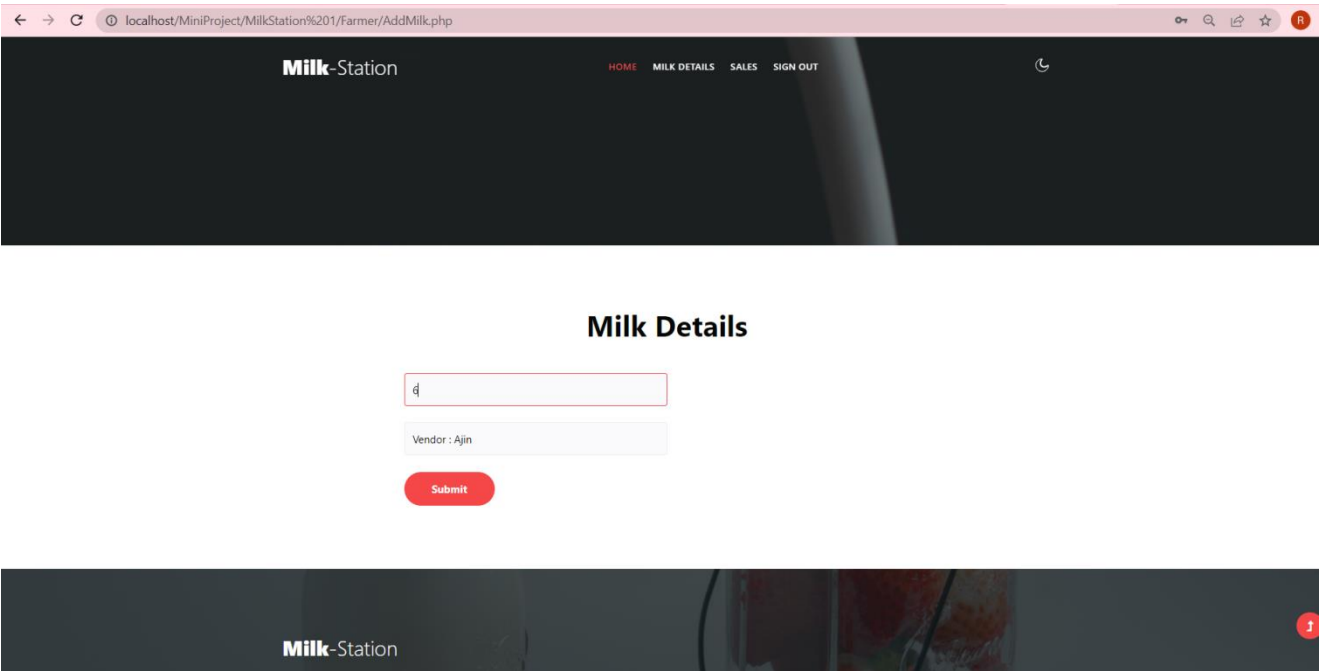


Figure 3.2.4: Supply Milk

3.2.5. Sales Details

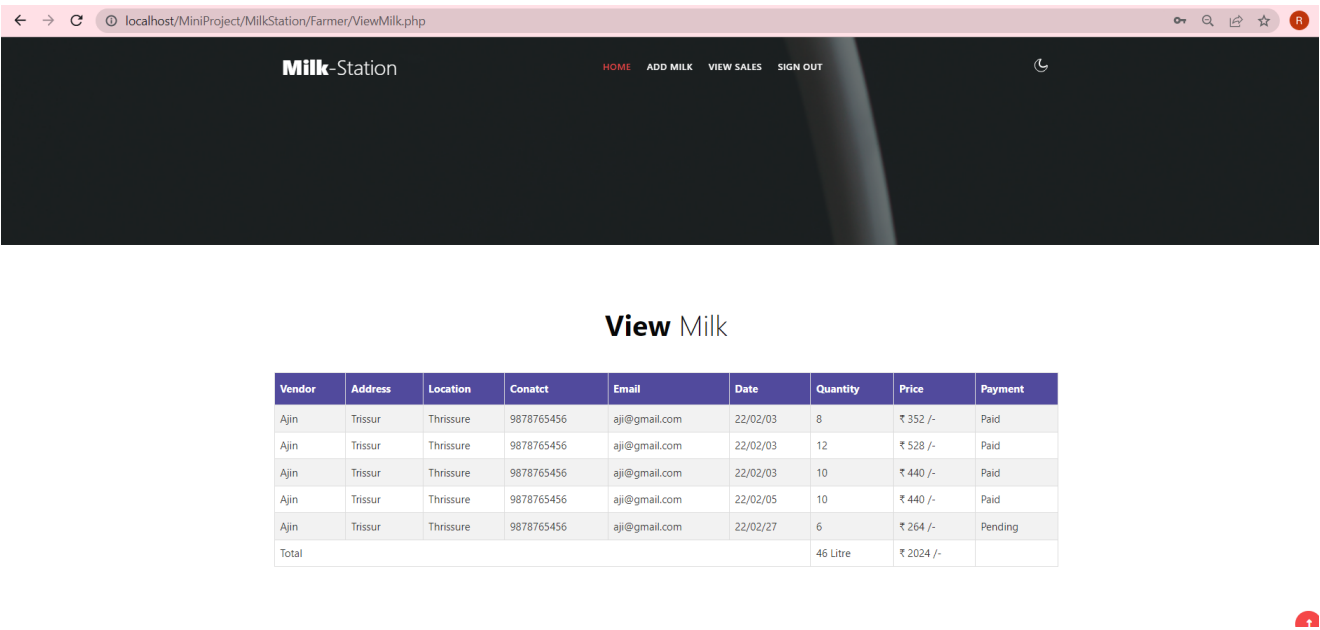


Figure 3.2.5: Sales Details

3.2.6. Payment for Farmer

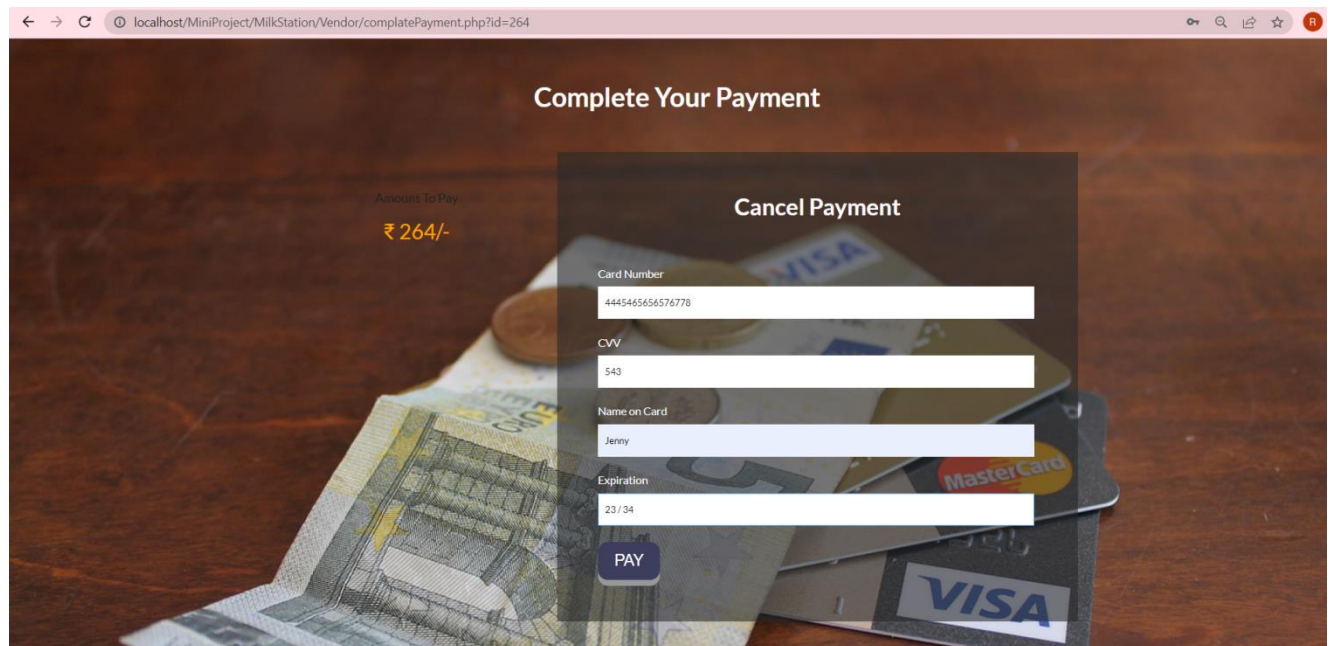
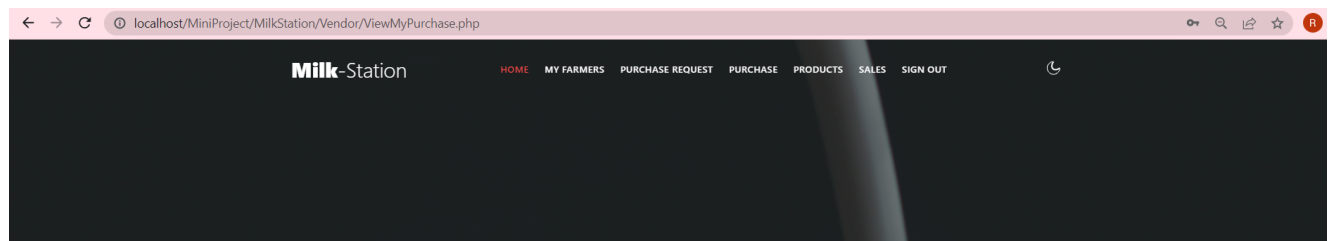


Figure 3.2.6: Payment for Farmer

3.2.7. Purchase Details



View Milk

Farmer	Address	Location	Contact	Email	Date	Quantity	Price	Payment
Jeena	Ernakulam	Ernakulam	8775546757	jeena@gmail.com	22/02/03	8	₹ 352 /-	Paid
Jeena	Ernakulam	Ernakulam	8775546757	jeena@gmail.com	22/02/03	12	₹ 528 /-	Paid
Jeena	Ernakulam	Ernakulam	8775546757	jeena@gmail.com	22/02/03	10	₹ 440 /-	Paid
Jeena	Ernakulam	Ernakulam	8775546757	jeena@gmail.com	22/02/05	10	₹ 440 /-	Paid
Abi	Abi Adrs	Aluva	9090909090	abi@mail.com	22/02/12	15	₹ 660 /-	Paid
Jeena	Ernakulam	Ernakulam	8775546757	jeena@gmail.com	22/02/27	6	₹ 264 /-	Paid
Total						61 Litre	₹ 2684 /-	

Figure 3.2.7: Purchase Details

3.2.8. Add Product

The screenshot shows a web browser at the URL `localhost/MiniProject/MilkStation/Vendor/vendorAddProducts.php`. The page has a dark header with the 'Milk-Station' logo and navigation links: HOME, MY FARMERS, PURCHASE REQUEST, PURCHASE, PRODUCTS, SALES, and SIGN OUT. The main content area is titled 'Add Products' and contains a form with the following fields: a text input with 'Ghee', a text input with '200', a text input with '6', a text input with 'Pure Ghee', and a file upload section with a 'Choose File' button and the filename 'ghee.jpg'. A red 'Submit' button is at the bottom of the form. A red notification icon is visible in the bottom right corner.

Figure 3.2.8: Add Product

3.2.9. Request Milk for Customer

The screenshot shows a web browser at the URL `localhost/MiniProject/MilkStation/Customer/RequestMilk.php`. The page has a dark header with the 'Milk-Station' logo and navigation links: HOME, REQUEST MILK, PRODUCTS, MY PURCHASES, and SIGN OUT. The main content area is titled 'Request Milk' and contains a form with the following fields: a text input with '5', a date-time input with '27-02-2022 21:46', and a red 'Submit' button. A red notification icon is visible in the bottom right corner.

Figure 3.2.5: Request Milk for Customer

3.2.10. Milk Products

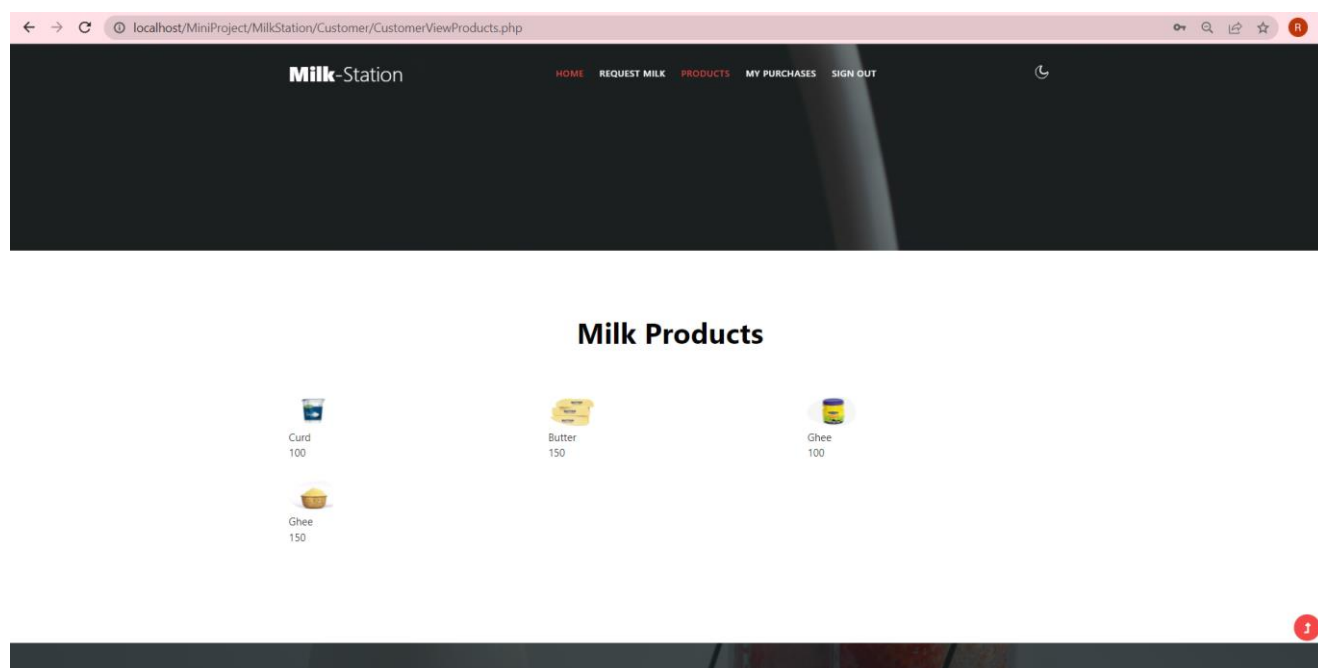


Figure 3.2.10: Milk Products

3.2.11. Report

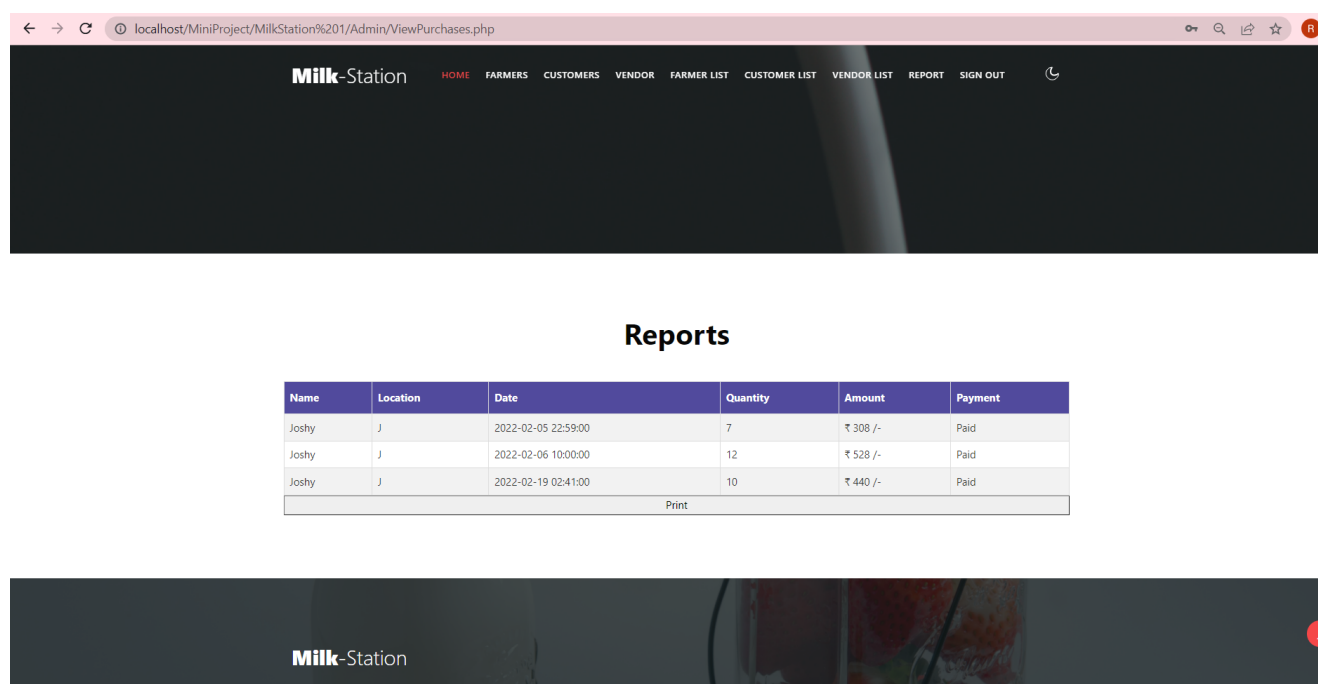


Figure 3.2.11: Report

4. TESTING AND IMPLEMENTATION

4.1. Testing

Software testing is a critical element of software quality assurance and represents ultimate review of specification, design and code generation. Once the source code has been generated the program should be executed before the customer gets it with the specific intend of finding and removing all errors, test must be conducted systematically and test must be designed using disciplined techniques.

Test Cases

Testing is based on test cases. It describes which feature or service test attempts to cover. In test cases specify what you are testing and which particular feature it tests.

- Test the normal use of system
- Test the abnormal, but reasonable use of system
- Test the abnormal and reasonable use of system
- Test the boundary Conditions

Unit Testing

Unit testing is a level of software testing where individual units/ components of the software are tested. The purpose is to validate that each unit of the software performs as designed. In the system the units testing perform separately and each module passes the test cases successfully. In each unit, its working is monitored so that it works safely and accurately. In each modules the unit testing is performed for smooth working of each modules.

Integration Testing

Integration Testing is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. During the Integration testing phase of my project its clear that the combined components/units don't affect the expected processes and other modules. Here we mixed up the units that have passed the test cases during unit test. And check-out the workflow of the modules.

Unit test cases

Sl no	Procedure	Expected result	Actual result	Status
1.	User registration	Registration successful	User details saved and registered successfully	Pass
2.	User login	Login successful	User logged in	Pass
3.	Add milk	Milk added successfully	Milk details is saved and added successfully	Pass
4.	Purchase milk	Milk purchased successfully	Purchased details is saved and added successfully	Pass
5.	Add product	Product added successfully	Product details is saved and added successfully	Pass
6.	Purchase product	Product purchased successfully	Product purchased details is saved and added successfully	Pass
7.	Add payment	Payment added successfully	Payment details added successfully	Pass
8.	Generate report	Report generated successfully	Details such as sales, purchase, payment saved successfully	Pass

Integration test cases

Sl no	Procedure	Test case description	Expected result	Actual result	Status
1.	Integrating registration and login	Users give their details in registration module. After successful registration users can login.	Registration and login must work properly.	Successful working of registration and login.	Pass
2.	Integrating milk and payment	Farmer adds milk details. Vendor view them. When they make order and they can make payment.	Milk and payment must work properly.	Successful working of milk and payment.	Pass
3.	Integrating milk products	Vendor adds milk products. From vendor the customer can order the milk products.	Milk products order must work properly.	Successful working of milk products order.	Pass

System Testing

The process of testing an integrated system to verify that it meets specified requirements in my project it satisfied all the requirements such as supply milk, request milk, adding milk products, order milk products etc...

4.2. Implementation

Implementation simply means carrying out the activities described in requirement. After testing, the system “Milk Station” is ready for the implementation. Implementation is the stage of the project when the theoretical design is turned in to a working system. Implementation is the process of bringing a newly developed system or revised into operational one. The new system and its components are to be tested in a structured and planned manner. There are some challenges faced by me while implementing the software. Some of them

- Code-reuse: There are huge issues faced by me for compatibility checks and deciding how much code to re-use.
- Version Management: Every time a new release is developed, I have to maintain version and configuration related documentation. In the Git hub helps me to solve such version control issues.

The implementation stage of a project is often very complex and time consuming. This involves careful planning, investigation of the current system and constraints of implementation, training the operating users in the changeover procedures before the system is setup and running. So, “Milk Station” is easy to use. It would be very easy to run also.

4.3. Git History

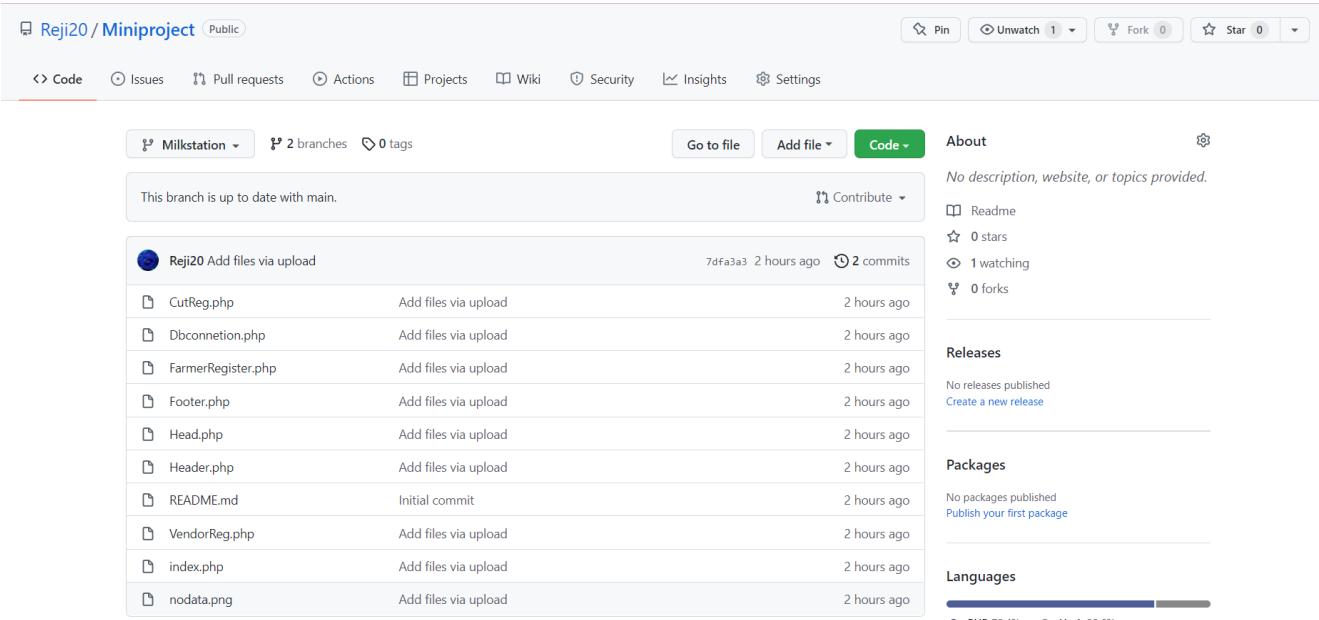


Figure 4.3: Git History

5. CONCLUSION

The project can ease a variety of a dairy process like member management, report analysis, accounting and finance, milk supply. This website will be showing the details of employees in the milk station information system, it will show quality of milk sold in the dairy, it will help them to know their frequenters customers and their information details in their dairy database. It will help to know daily sale each day enable to access the business progress, It will enable them to their know suppliers who brought for them in daily time.

This Project is based on the concept of managing dairy products and their records. Design of this is so simple that the user won't find difficulties while working on it. This project helps the user for easy management of transaction activities. This web application helps to register all the suppliers, purchase, Sales details etc. This project deals with the purchase and sale of milk by making records in the database. The milk station is a comprehensive approach to manage the Dairy products collection, making payments. The admin of the website can manage employees and payroll. This is software developed with the purpose of helping the dairy farm mitigate their losses and increase their productivity.

The results provided by system are truly close to what was expected of the system. Since the system aims at providing maximum output from minimum input, the system is coded in such a way that it tries to provide results in most of the cases. The system always has the required requirements to produce the expected results. It is set up in such a way that the controller of the system has to give his minimum effort. The system has a certain flow.

The main advantage of the system is its user-friendly interface and the ability to produce maximum output from minimum input. The system is capable of updating information in more than one database without affecting the integrity constraints. In many of the applications users have to give most of the information in one or more scenarios, The "Milk Station" tries to ensure that user does not feel that kind of boredom. This system provides new features like online payment. This project provides new facility to the user, and helps them to save their time and money.

The milk station required less time for completion of any work. This system is reduces the chances of error. This system should work smoothly and very fast. It saves time and manpower. The system is user friendly and anyone having computer knowledge can handle it easily. Suitable for computerized data entry. It maintains sales information, customer information and milk Rate Information. The initial investment in dairy farming business is low in comparison to other Industry. It is environment-friendly. The demand for milk product is increasing rapidly. Cow dung is good organic manure it increases soil fertility. Cow dung can be used for the production of biogas.

The system certainly possesses certain disadvantages. Productivity of Indian dairy animals is lower compared to other country. It is due to the lack of scientific livestock feeding practice, inadequacy and unavailability of livestock healthcare. Some other problems associated with the system are improper milk marketing facilities and uncertain price of milk for producers, lack of infrastructure for milk collection, transportation, processing, lack of veterinary and extension services, milk losses due to lack of cold chain facilities and lack of clean milk production practices. These days in India production of milk and its products are more hence the dairy associations are decreasing the price given to farmers per liter. The dairy products needs to wider marketing facility.

6. SCOPE AND FUTURE ENHANCEMENTS

Changes in software engineering technology are indeed rapid. By the time a decision is made to adopt a new method, conduct the training necessary to understand its application and introduced technology into the software development culture. Something new has come along and the process begins. The application has been developed in such a way that any user can access at any time.

This application is built such a way that it should suits for all type of Milk Distributors in future. So every effort is taken to implement this project in this Milk station, on successful implementation in this Milk station, we can target other Milk Distributors in the city. Milk station project is a software application useful for dairy forms for managing daily activities like receiving of milk from various sources.

7. APPENDIX

7.1. Minimum Software Requirement

Software : Xampp Server, Visual Studio Code

Operating System : Windows

7.2. Minimum Hardware Requirement

Hardware capacity : 100 GB (minimum)

RAM : 2 GB

Processor : Intel Core i3 preferred

Display : 1366 * 768

8. REFERENCES

- <https://www.coursehero.com/>
- <https://sites.google.com/>
- <https://www.wikipedia.org/>
- <https://w3layouts.com/>
- <https://www.geeksforgeeks.org/>
- <https://javapoint.com/>