**DAIRY MANAGEMENT SYSTEM**

A project submitted to

**UKA TARSADIA UNIVERSITY**

in partial fulfillment of the requirements for the degree of

**Bachelor of Science**

in

**Information Technology**

for

**5 Years Integrated M.Sc. (IT)**

By

**[Rajan Prajapati, Priyank Lad]**

[201606100110041, 201606100110007]

Guided by

**Mr. Bhavik Sarang**



**Babu Madhav Institute of Information Technology**

### Uka Tarsadia University

### Bardoli – 394350

### May 2019

**CERTIFICATE**



This is to certify that **Rajan Prajapati (201606100110041) and Priyank Lad (201606100110007)** has submitted project entitled “**Dairy Management System”** as the partial fulfillment for the award of the degree of Bachelor of Science in Information Technology for 5 Years Integrated M.Sc. (IT) in 2018 – 2019.

**Date:**

**Place:**

**Mr. Bhavik Sarang Dr. Jitendra Nasriwala**

Guide Programme Coordinator

**External Examiner**



**Babu Madhav Institute of Information Technology,**

**Uka Tarsadia University,**

**Bardoli - 394350**

**Acknowledgement**

We express thanks and gratitude to almighty **god**, my **parent** and other **family** members without whose unconditional support we could not have made this career in M.Sc. (IT).

We also extend my gratitude to internal project guide **Mr. Bhavik Sarang**, who assisted us in completing the project.

We would like to express sincere thanks to program coordinator **Dr. Jitendra Nasriwala**, who gave an opportunity to undertaken such a great challenging and innovative work.

Rajan Prajapati

Priyank Lad

**Table of Content**

|  |  |  |  |
| --- | --- | --- | --- |
| **Chapters** | **Particulars** | | **Page no.** |
| **1** | **Introduction** | |  |
| 1.1 | Project Definition | 1 |
| 1.2 | Purpose | 1 |
| 1.3 | Scope | 1 |
| 1.4 | Technology Used | 2 |
|  | | | |
| **2** | **System Analysis** | |  |
| 2.1 | Identification of need | 3 |
| 2.2 | Functional Requirement | 3 |
| 2.3 | Non-functional Requirement | 4 |
| 2.4 | Modules | 4 |
| 2.5 | User Characteristics | 5 |
| 2.6 | Use Case description form | 6 |
| 2.7 | Use-Case diagrams | 6 |
| 2.8 | Activity diagrams | 7 |
|  | | | |
| **3** | **System Design** | |  |
| 3.1 | Data Dictionary | 12 |
| 3.2 | Interface Design | 19 |
|  | | | |
| **4** | **Testing** | |  |
| 4.1 | Test Cases | 25 |
|  | | | |
| **5** | **Future Enhancement** | | 27 |
|  | | | |
| **6** | **Reference** | | 27 |
|  | | | |
| **7** | **Bibliography** | | 27 |

**List of Figures**

[Figure 1:Use case Diagram of Dairy Management System 6](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775520)

[Figure 2:Activity- Login 7](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775521)

[Figure 3:Activity- Forget Password 8](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775522)

[Figure 4:Activity- Manage Cart 9](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775523)

[Figure 5:Activity- Manage Customer 10](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775524)

[Figure 6:Activity- Manage Product 11](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775525)

[Figure 7:Login Page 19](#_Toc7775526)

[Figure 8:Registration Page 20](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775527)

[Figure 9:Home Page 21](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775528)

[Figure 10:Product List 22](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775529)

[Figure 11:Single Product 23](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775530)

[Figure 12:Cart Page 24](file:////Users/priyanklad/Desktop/Dairy%20Management%20system.docx#_Toc7775531)

**List of Tables**

[Table 1:tblArea 12](#_Toc7776484)

[Table 2:tblLandmark 12](#_Toc7776485)

[Table 3:tblStaff 13](#_Toc7776486)

[Table 4:tblUnit 13](#_Toc7776487)

[Table 5:tblMeasurement 14](#_Toc7776488)

[Table 6:tblPcategory 14](#_Toc7776489)

[Table 7:tblProduct 14](#_Toc7776490)

[Table 8:tblMilkcollection 15](#_Toc7776491)

[Table 9:tblCustomer 15](#_Toc7776492)

[Table 10:tblMembership 16](#_Toc7776493)

[Table 11:tblCart 16](#_Toc7776494)

[Table 12:tblBill 16](#_Toc7776495)

[Table 13:tblCustomerOrder 17](#_Toc7776496)

[Table 14:tblRetailer 17](#_Toc7776497)

[Table 15:tblRetailerOrder 17](#_Toc7776498)

[Table 16:tblContactus 18](#_Toc7776499)

[Table 17:Test Case for Login 25](#_Toc7776500)

[Table 18:Test case for Change Password 26](#_Toc7776501)

**Ch-1 Introduction**

**1.1 Project Definition**

Dairy Milk management System is a web-based software application to maintain day to day transaction of milk collection and sell milk products. This system is easy to operate and understand. This system collects the milk from many different sources and distribute to customer or milk buyers.

**1.2 Purpose**

The system describes the selling of MILK & various MILK PRODUCTS according to customer need. It provides the customer to create order and also cancel order. And it’s provides easy to use system.

**1.3 Scope**

This Web-based software is available in limited area.

The aim is to focus on MILK & MILK PRODUCTS in which user can view product and order according to their need.

Only Admin can generate sales reports.

**1.4 Technology Used**

**Front End**

PHP - PHP: Hypertext Preprocessor is a general-purpose programming language originally designed for web development.

HTML – Hyper Text Markup Language is a system used to mark text for World Wide Webpages in order to obtain colors, style, picture, etc.

**Back End**

MYSQL – It is an Oracle-backed open source relational database management system based on structured Query Language.

**Tools**

WampServer - It refers to a software stack for the Microsoft Windows operating system and consisting of the Apache web server, OpenSSL for SSL support, MySQL database and PHP programming language.

Net Beans IDE 8.2 – It is an open-source integrated development environment (IDE) for developing with Java, PHP, C++, and other programming languages.

Umlet – It is an open-source Java- based UML tool designed for teaching the Unified Modeling Language and for quickly creating UML diagrams.

**Scripting Language**

JavaScript - It is an object-oriented computer programming language commonly used to create interactive effects within web browsers.

AJAX – It is a set of web development techniques using many web technologies on the client side to create asynchronous web applications.

CSS – Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language like HTML.

BOOTSTRAP – It is a framework to help you design websites faster and easier.

**Ch-2 System Analysis**

**2.1 Identification of Need**

The System is useful for the user who wants to purchase MILK & MILK PRODUCTS as per their requirement in easy manner without any physical visit to the DAIRY SHOP.

**2.2 Functional Requirements**

* + Admin can add new staff member, update its details according to its role.
  + Admin can add new product, Update Product details, and also remove product according to its selling.
  + Admin maintain records of daily milk collection and also update record if any error occurs.
* Admin can maintain the record of those customer which have create account in system.
  + Admin can see customer orders which they have place.
* Admin confirm the order and maintain the order of customer which they have purchased.
  + Admin can manage retailers like admin can see records how much milk is collected by which retailer.
  + Admin can generate reports of daily milk collection, products, selling report etc.
  + Customer can create account in system.
  + Customer can add product to cart which they want to purchase and also remove product from cart which they do not want to purchase.
  + Customer can order products and cancel products according their needs.
  + Customer can generate bill of products which they have purchased.
* Customer can shall be able to modify cart and, also maintain product in cart according to quantity wise.

**2.3 Non-functional Requirements**

**1. Availability**

System will be available 24x7.

**2. Responsive**

System will be responsive to any browser.

**3. Portability**

This system can be used in different platforms.

## 2.4 Modules

* Staff Module
  + This Functionality is Maintain by Admin.
  + In this functionality Admin can add new staff member, update its details according to its role.
* Product Module
  + This Functionality is Maintain by Admin.
* Manage Milk Product.
  + Admin can add new product, Update Product details, and also remove product according to its selling.
* Manage Daily Milk collection.
  + Admin maintain records of daily milk collection and also update record if any error occurs.
* Customer Module
  + This Functionality is Maintain by Admin.
  + In this functionality Admin Can Manage customers.
* Customer Order Module
  + This functionality is Maintain by Admin and customer.
  + Customer order products and cancel products according their needs.
  + Admin can see customer orders which they have place.
* Cart Module
  + This functionality is used by customer.
  + Customer add product to cart which they want to purchase and also remove product from cart which they do not want to purchase.
* Retailer Module
  + This functionality is used by Admin and Retailer.
  + Admin can manage retailers like admin can see records how much milk is collected by which retailer.
  + Retailer can collect milk from customer according to category of cattle and keep record of each customer. And also update record if any error occurs.
* Generate Bill Module
  + This functionality is used by Customer.
  + Customer generate bill of products which they have purchased.
* Generate Report Module
  + This Functionality is Maintain by Admin.
  + Admin can generate reports of daily milk collection, products, selling report etc.

## 2.5 User Characteristics

* **Admin**
  + Admin can add new staff member, update its details according to its role.
  + Admin can add new product, Update Product details, and also remove product according to its selling.
  + Admin maintain records of daily milk collection and also update record if any error occurs.
* Admin can maintain the record of those customer which have create account in system.
  + Admin can see customer orders which they have place.
* **Customer**
  + Customer can create account in system.
  + Customer can add product to cart which they want to purchase and also remove product from cart which they do not want to purchase.
  + Customer can order products and cancel products according their needs.
  + Customer can generate bill of products which they have purchased.
* Customer can shall be able to modify cart and, also maintain product in cart according to quantity wise.

**2.7 Use Case Diagram**



Figure 1:Use case Diagram of Dairy Management System

## 2.8 Activity Diagrams

1. **Login**

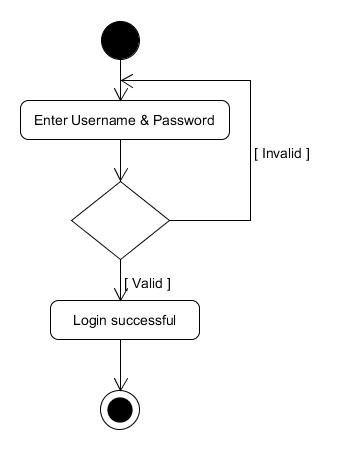


Figure 2:Activity- Login

1. **Forget Password**

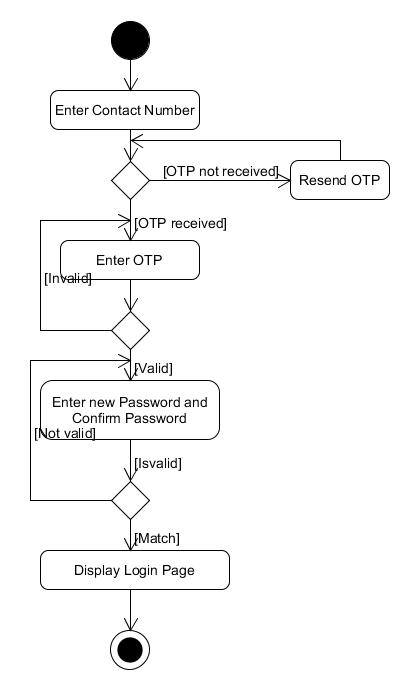


Figure 3:Activity- Forget Password

1. **Manage Cart**

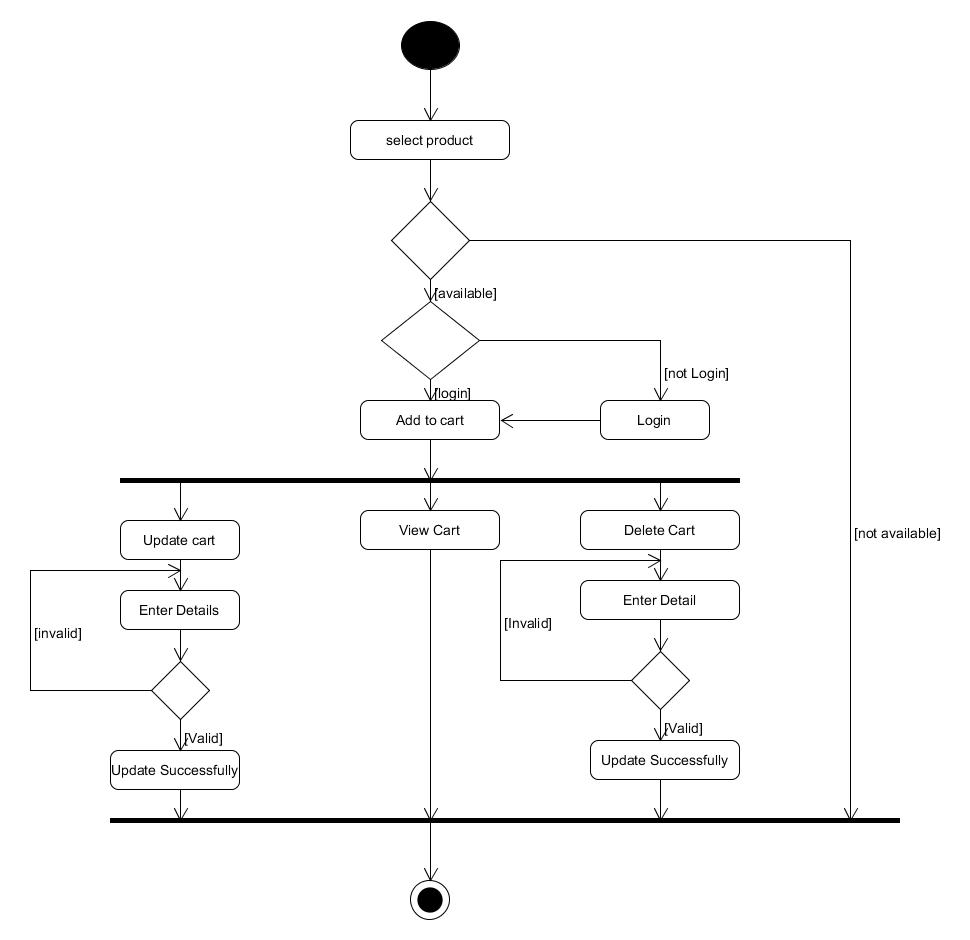


Figure 4:Activity- Manage Cart

1. **Manage Customer**

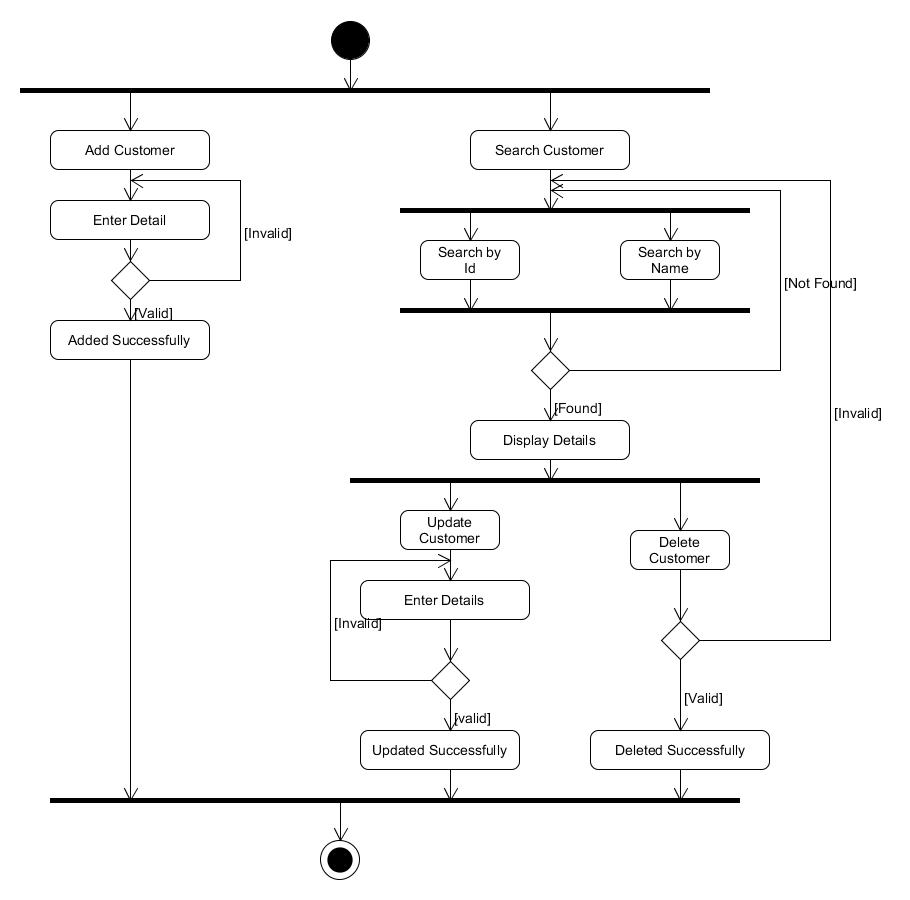


Figure 5:Activity- Manage Customer

1. **Manage Product**

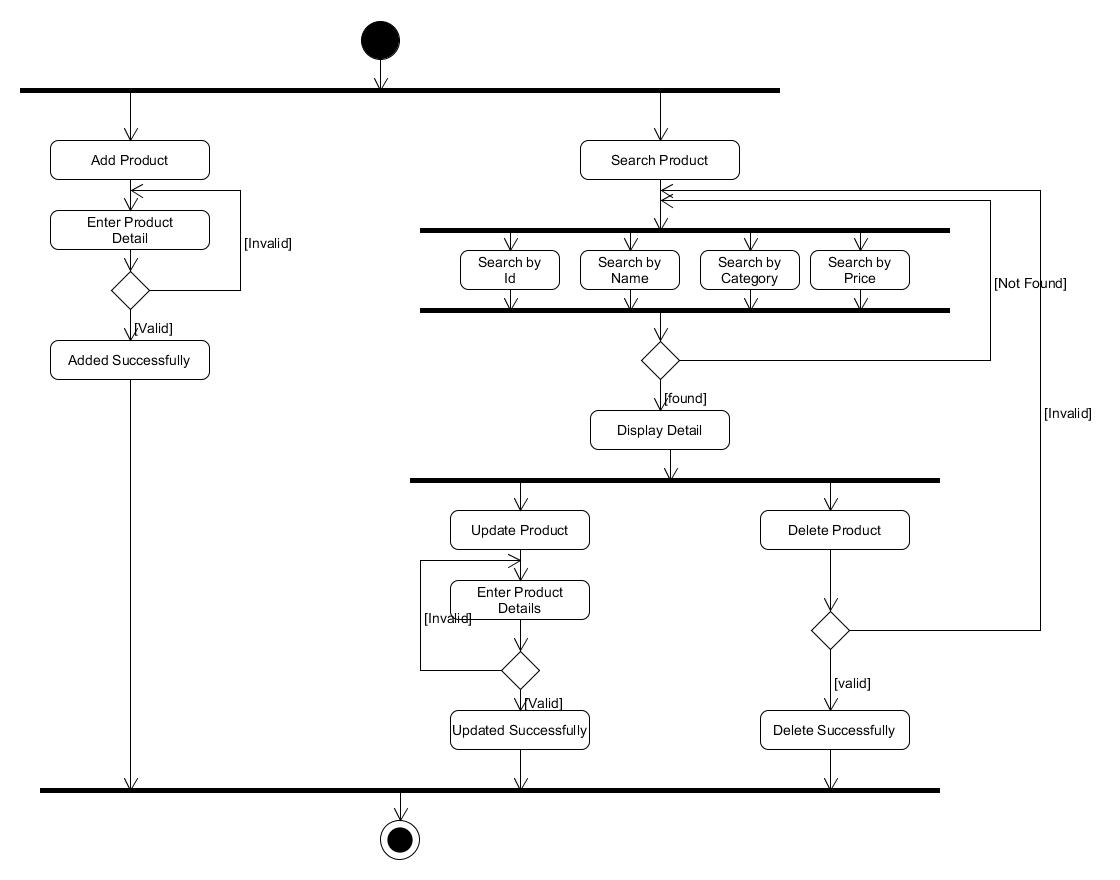


Figure 6:Activity- Manage Product

**Ch-3 System Design**

* 1. **Data Dictionary**

1. **tblArea**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Id | Int | Primary key, Autoincrement | Generate unique areaid. |
| name | Varchar (100) | unique | Name of area. |
| Pincode | Int | Not null | Pin code of area. |

Table :tblArea

## tblLandmark

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Id | int | Primary key  Autoincrement | Id of landmark. |
| Name | Varchar (100) | unique | Name of landmark. |
| Areaid | Int | Foreign key | References to “Id” from tblArea. |
| Status | Int | Not null | 0 for not available and 1 for available area. |

Table :tblLandmark

1. **tblStaff**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| id | Int | Primarykey,  Autoincrement | Generate unique staffid. |
| Fname | Varchar (100) | Not null | First name of staff person. |
| Lname | Varchar (100) | Not null | Last name of staff person. |
| Address | Varchar (200) | Not null | Address of staff person. |
| Landmarkid | Int | Foreign key | References to “Id” from tblLandmark. |
| Gender | Char (6) | Not null | Gender of staff person. |
| Contactno | Numeric | Not null | Phone number of staff person. |
| Role | Varchar (50) | Not null | Designation of staff person. |
| Emailid | Varchar (100) | Unique | Email id of staff person. |
| Password | Varchar (100) | Not null | Password of staff person. |
| update\_datetime | Datetime | Not null | Updated date and time. |

Table :tblStaff

1. **tblUnit**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description. |
| id | Int | Primary key,  Autoincrement | Id of unit. |
| Type | Varchar (20) | Not null | Type of unit (kg, ml, litres etc). |

Table :tblUnit

1. **tblMeasurement**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Id | Int | Primary key, not null,  Autoincrement | Id of unit measurement. |
| Type | Varchar (20) | Not null | Name of measurement. |
| United | int | Foreign key | Unit id. |

Table :tblMeasurement

1. **tblPcategory**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description. |
| Id | int | Primary key, not null,  Autoincrement | id of category. |
| Name | Varchar (100) | Not null | Name of product category. |
| Imagepath | Path | Not null | Image of product. |

Table :tblPcategory

1. **tblProduct**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Id | Int | Primary key, not null,  Autoincrement | Id of product. |
| Pcategoryid | Int | Foreign key | id of product category. |
| measurementid | Int | Foreign key | Id of measurement . |
| Price | number | Not null | Price of product. |
| manufacture\_date | Date | Not null | Date of manufacture product. |
| expire\_date | Date | Not null | Date of expire product. |
| updatedatetime | Datetime | Not null | Update date and time. |
| Status | Int | Not null | To check product is available or not. |

Table :tblProduct

## tblMilkcollection

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Id | Int | Primary key, not null, autoincrement | Id of milk collection. |
| cattletype[C/B] | Char (10) | Not null | Type of cattle from milk collected. |
| Litre | Float | Not null | Litre of collected. |
| Date | Date | Not null | Date of collected milk. |
| Time | Time | Not null | Time of collected milk. |
| Staffed | Int | Foreign key | Id of staff. |
| updatedatetime | Datetime | Not null | Update date and time. |

Table :tblMilkcollection

1. **tblCustomer**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Id | Int | Primary key, not null, autoincrement | Id of customer. |
| Fname | Varchar (100) | Not null | Firstname of customer. |
| Lname | Varchar (100) | Not null | lastname of customer. |
| Address | Varchar (200) | Not null | Address of customer |
| Landmarked | Int | Foreign key | Id of landmark |
| Gender | Char (7) | Not null | Gender of customer. |
| Contactno | Number | Not null | Contactno of customer. |
| Emailed | Varchar (100) | Not null | Email of customer. |
| Password | Varchar (100) | Not null | Password of customer. |
| updatedatetime | Datetime | Not null | Update date and time. |

Table :tblCustomer

**10.tblMembership**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Id | Int | Primary key, not null, autoincrement | Id of membership. |
| Customerid | Int | Foreign key | Id of customer. |
| Duration | Int | Not null | Duration of membership. |
| advance\_amount | Number | Not null | Advance amount by customer. |
| advance\_date | Date | Not null | Date of given advance amount. |
| total\_amount | Number | Not null | Total amount given by customer. |
| date | Date | Not null | Date of given amount. |

Table :tblMembership

**11.tblCart**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Id | Int | Primary key, not null, autoincrement | Id of cart. |
| Customerid | Int | Foreign key | Customer id. |
| Productid | Int | foreign key | Product id. |
| Qty | Int | Not null | Number of product. |

Table :tblCart

**12.tblBill**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| No | Int | Primary key, not null, autoincrement | Bill no. |
| Customerid | Int | Foreign key | Id of customer. |
| Date | Date | Not null | Date of bill |
| total\_amout | Number | Not null | Total amount of product. |

Table :tblBill

**13.tblCustomerOrder**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| No | Int | Primary key, not null, autoincrement | No of customer order. |
| Billno | Int | Foreign key | Number of bill. |
| Productid | Int | Foreign key | Id of product. |
| qty | Int | Not null | Quantity of product. |

Table :tblCustomerOrder

**14.tblRetailer**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Id | Int | Primary key, not null, autoincrement. | Id of retailer. |
| Name | Varchar (100) | Not null | Retailer name. |
| Address | Varchar (200) | Not null | Address of retailer. |
| Landmarked | Int | Foreign key | Id of landmark. |
| Contactno | Number | Not null | Contact number of retailer. |
| Emailed | Varchar (100) | Not null | Email id of retailer. |
| Password | Varchar (100) | Not null | Password of retailer. |

Table :tblRetailer

**15.tblRetailerOrder**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| no | Int | Primary key, not null, autoincrement. | No of retailer order. |
| Retailerid | Int | Foreign key | Id of retailer. |
| Liter | Int | Not null | Liter of milk given. |
| Price | Number | Not null | Price of per liter. |
| Date | Date | Not null | Milk given date. |

Table :tblRetailerOrder

**16.tblContactus**

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| Id | int | Primary key, not null, autoincrement | Id of visitor. |
| full name | Varchar (100) | Not null | Name of visitor. |
| Landmarked | Int | Foreign key | Id of landmark |
| Comment | Varchar (200) | Not null | comment by visitors. |
| updatedatetime | Datetime | Not null | Update date and time. |

Table :tblContactus

* 1. **Interface Design**

1. **Login**

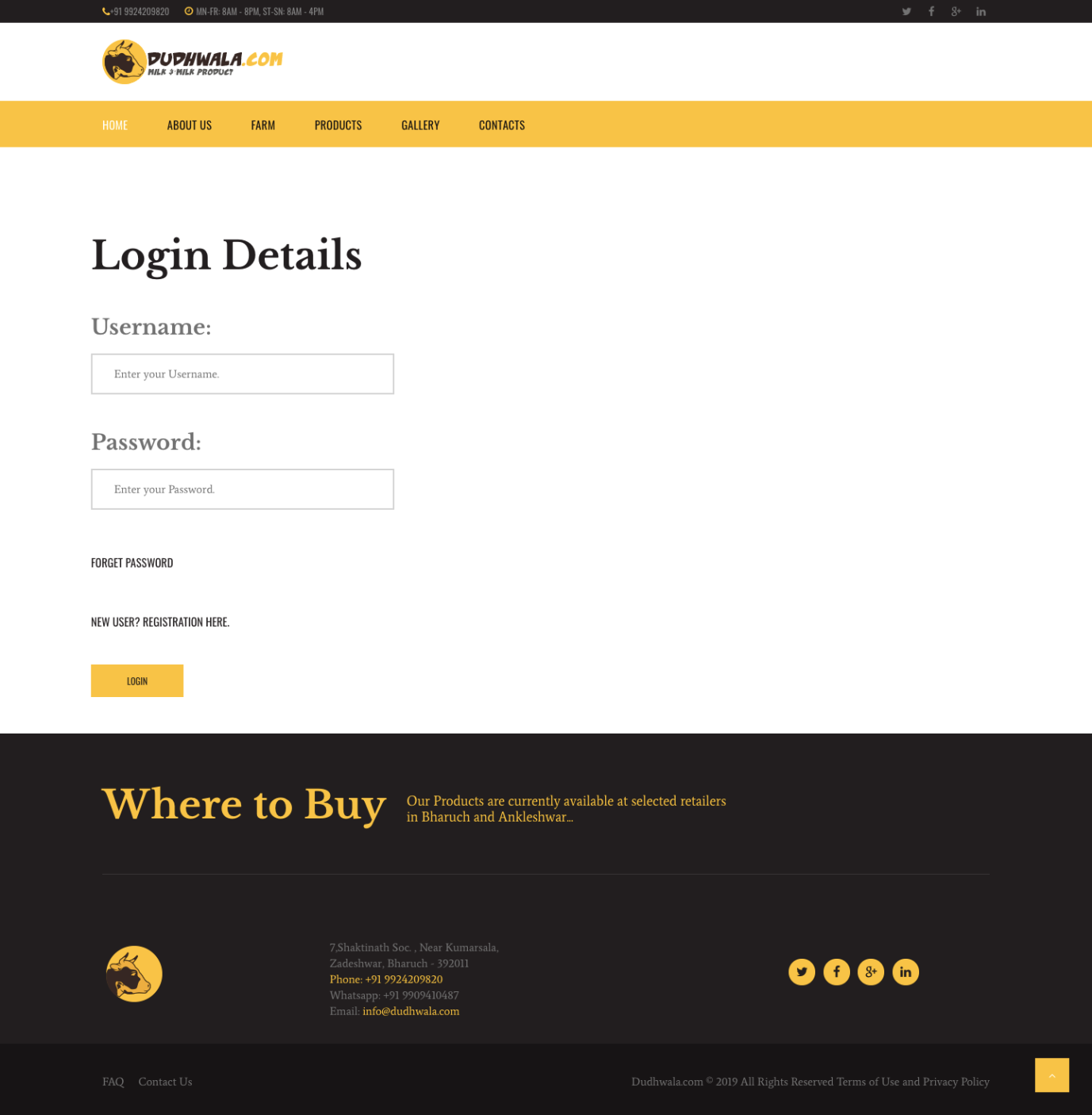
****

Figure :Login Page

1. **Registration**

Figure 8:Registration Page

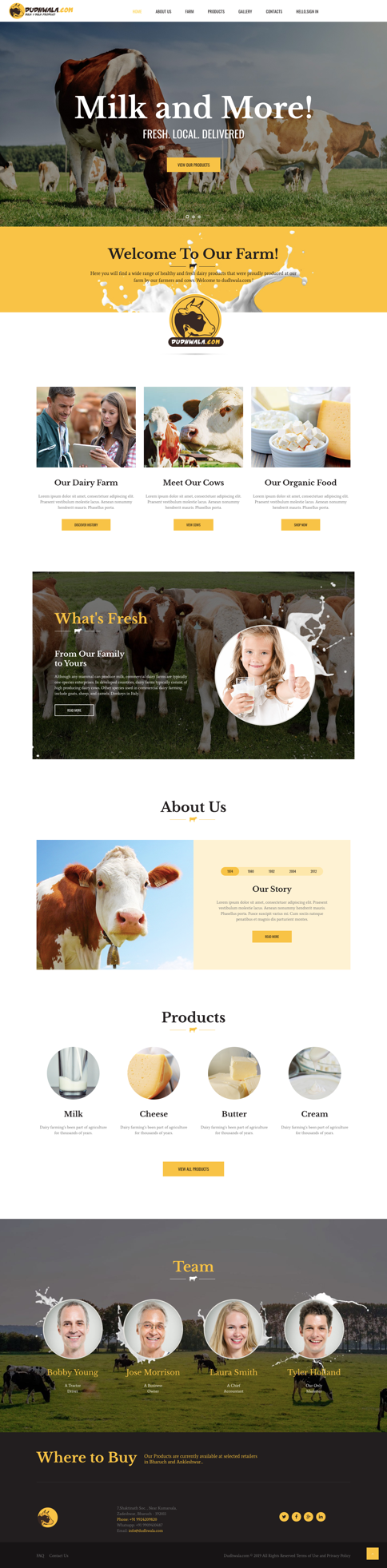
1.  **Home Page**

Figure 9:Home Page

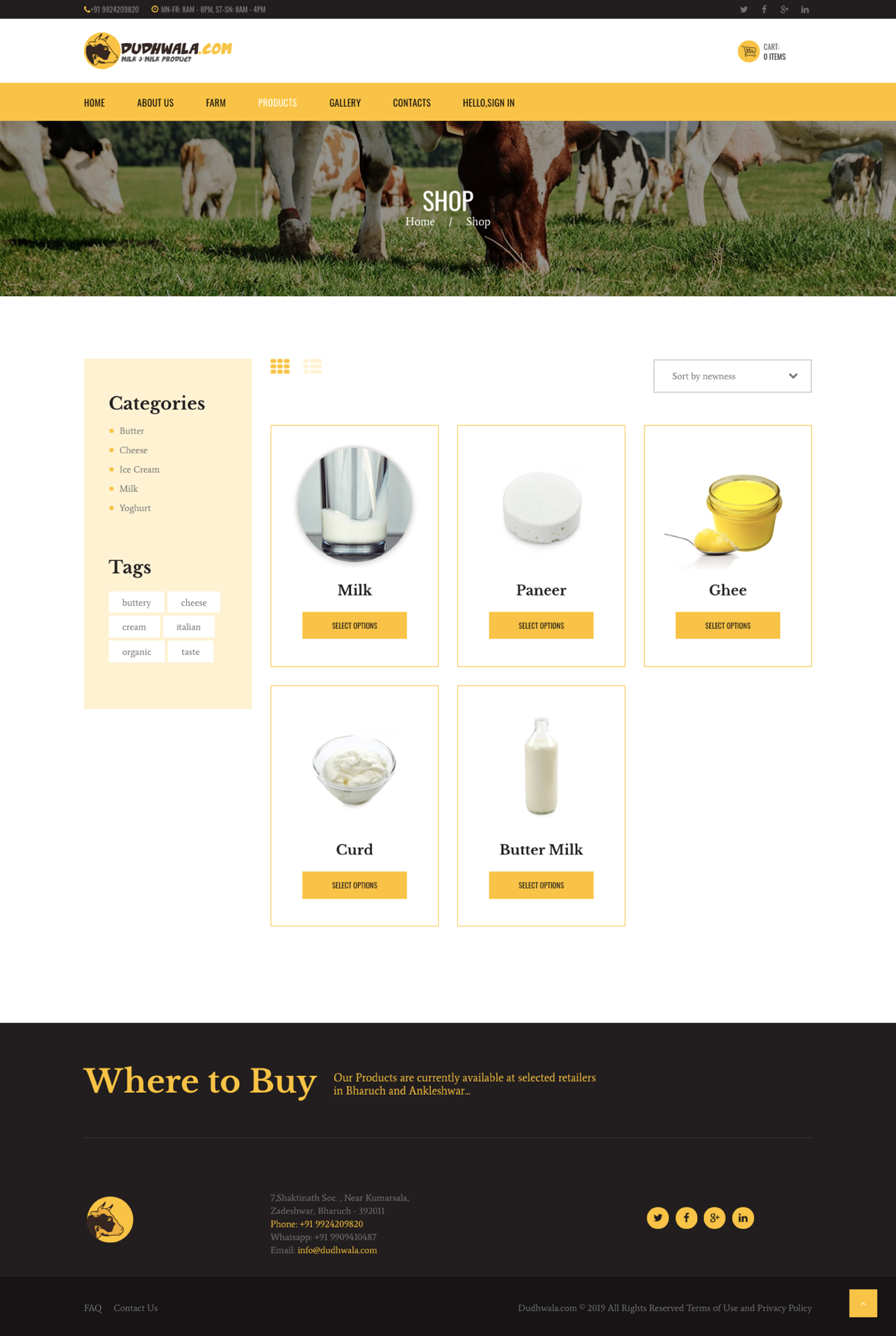
1. **Product List**

Figure 10:Product List

1. **Single Product**

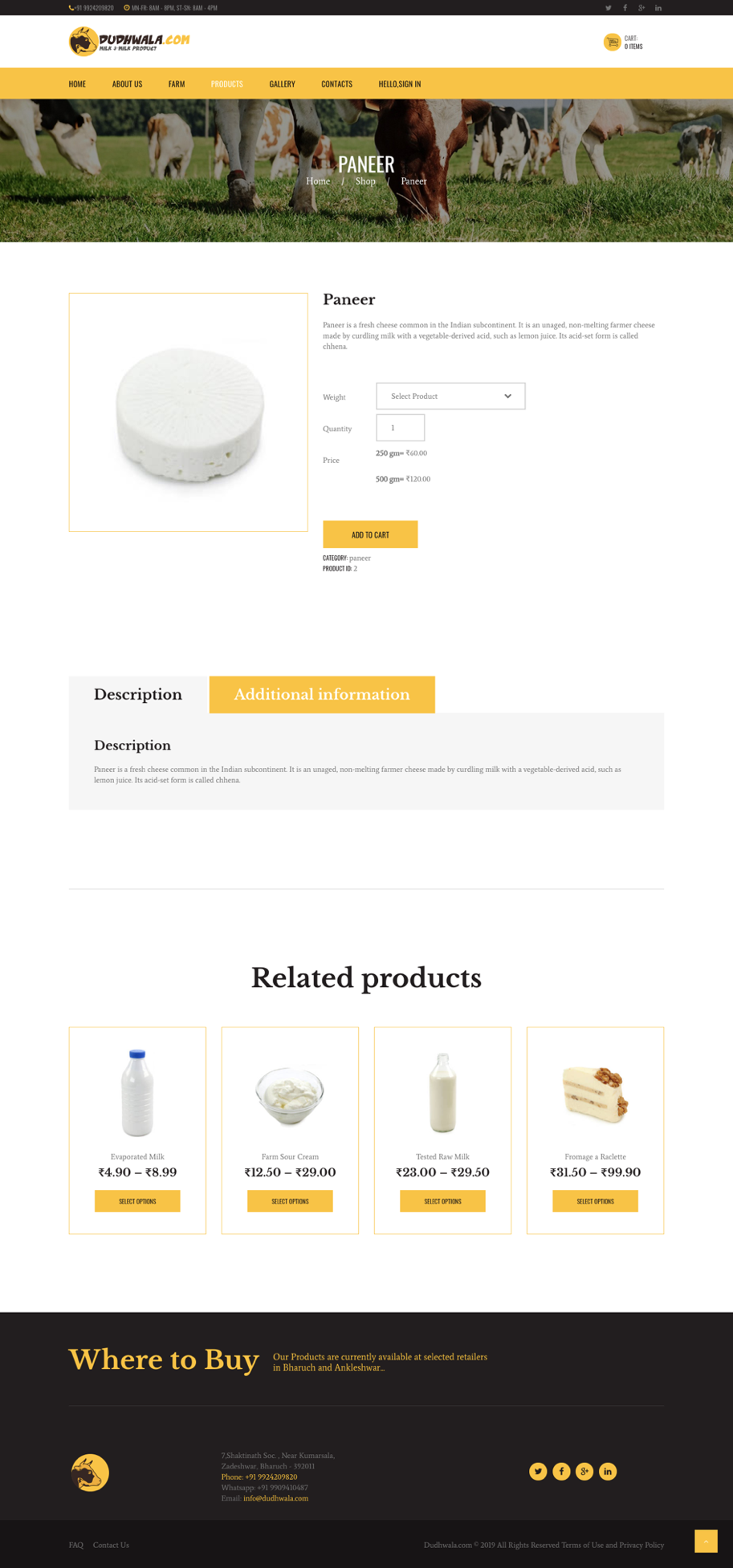


Figure 11:Single Product

**6.Cart**

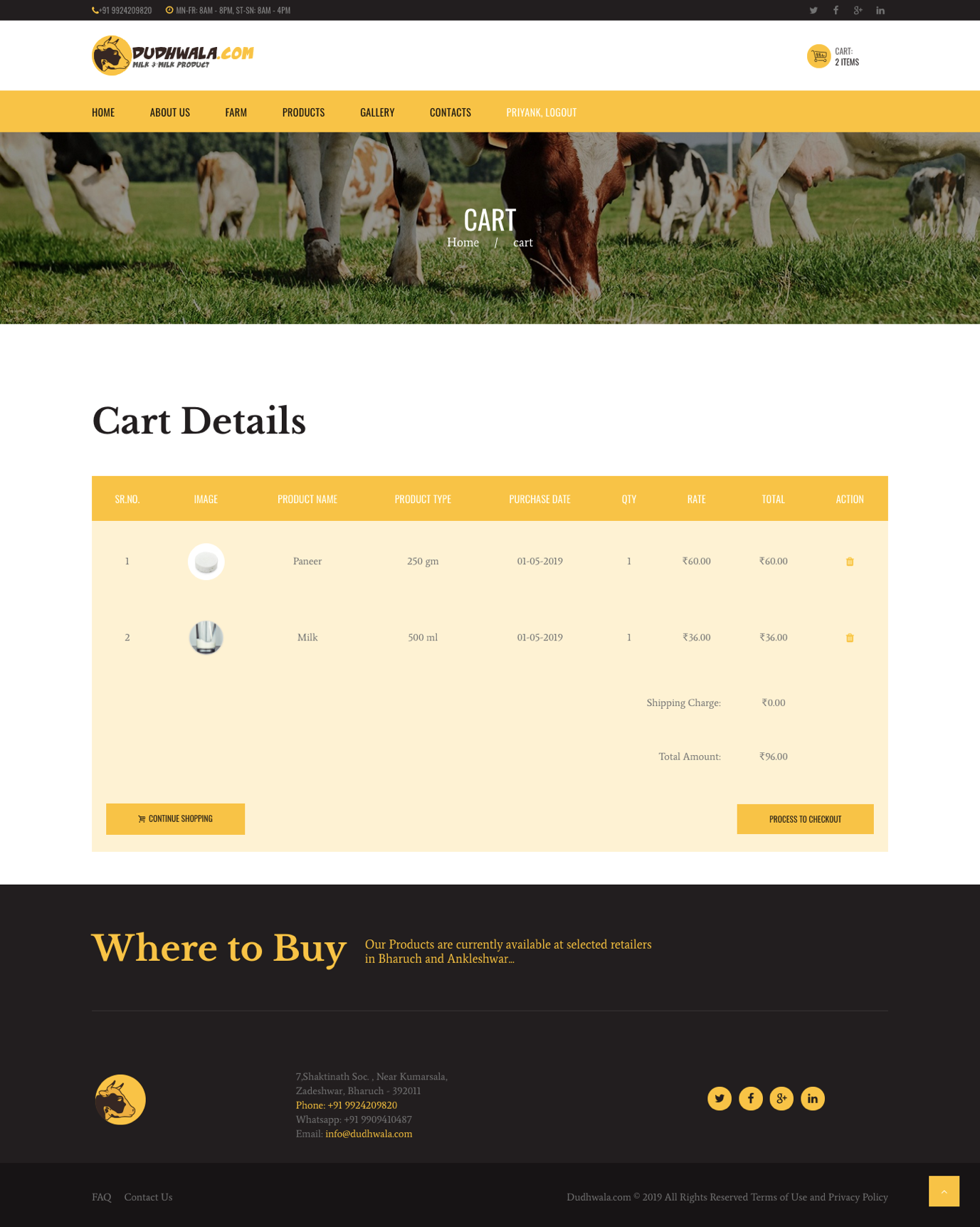


Figure 12:Cart Page

**Ch-4 Testing**

**4.1 Test Cases**

1. **Test case for Login**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **TCID** | **Scenario name/Description** | **Username** | **Password** | **Excepted**  **Result** | **Actual Result** | **Remarks** |
| TC1 | Username and password is valid | valid | valid | Redirect to another page | No Redirection | Fail |
| TC1 | Username and password is valid | valid | valid | Redirect to another page | Redirection | Pass |
| TC1 | Username is invalid | Invalid | valid | Show Error Message | No message  Display | Fail |
| TC1 | Username is invalid | Invalid | valid | Show Error Message | Message Displayed | Pass |
| TC1 | Password is Invalid | valid | Invalid | Show Error Message | Message Displayed | Pass |
| TC1 | Password is Wrong | valid | Invalid | Show Error Message | No Message displayed | Fail |
| TC1 | Password is Wrong | valid | Invalid | Show Error Message | Message Displayed | Pass |
| TC1 | Both is Incorrect | Invalid | Invalid | Show Error Message | Message  Displayed | Pass |

Table :Test Case for Login

1. **Test case for Change Password**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **TCID** | **Scenario name/Description** | **Current Password** | **New Password** | **Re-enter new Password** | **Expected Result** | **Actual Result** | **Remarks** |
| TC2 | Old, new and re-enter password are valid | Valid | Valid | Valid | Redirect to another Page | No Redirection | Fail |
| TC2 | Old, new and re-enter password are valid | Valid | Valid | Valid | Redirect to another page | Redirection done | Pass |
| TC2 | Old password is invalid | Invalid | Valid | Valid | Error Message Display | No Error message | Fail |
| TC2 | Old password is invalid | Invalid | Valid | Valid | Error Message Display | Error message  Display | Pass |
| TC2 | New or Re-enter password are wrong | Valid | Valid | Invalid | Error Message Display | Error message display | Pass |
| TC2 | New or Re-enter password are wrong | Valid | Invalid | Invalid | Error Message Display | No Error message | Fail |
| TC2 | New or Re-enter password are wrong | Valid | Invalid | Invalid | Error Message Display | Error message display | Pass |

Table :Test case for Change Password

**Ch-7 Future Enhancement**

* Payment using PAYPAL gateway.

**Ch 8 – References**

[1] Steverholzner, the Complete Reference PHP, McGraw Hill.

[2] Steve Suehring, Tim Converse and Joyce Park, PHP6 and Mysql, Wiley India Pvt. Ltd.

[3] Singh Y., Malhotra R., Object oriented Software Engineering, PHI.

**Ch 9 - Bibliography**

1. [www.google.com](file:///C:\Users\Rashmi\Desktop\www.google.com)

2. [www.stackoverflow.com](http://www.stackoverflow.com)

3. [www.w3schools.com](http://www.w3schools.com)

4. [www.tutorialpoints.com](http://www.tutorialpoints.com)