**A PROJECT REPORT**

**ON**

# "E-RATION MANAGEMENT"

*Submitted in partial fulfilment of the requirements*
*for the degree of*

## BACHELOR OF COMPUTER APPLICATION

*Prescribed By*

**Bengaluru City University**

By

ASHIQU SAJI (R1818214)

**V Semester**

**KLE SOCIETY's**

**S. NIJALINGAPPA COLLEGE**

**RAJAJINAGAR, BANGALORE-560010**

***Under the guidance of:***

**SHWETA S.M**

**Academic Year**

**2020-2021**

BENGALURU CITY UNIVERSITY

**KLE SOCIETY's S. NIJALINGAPPA COLLEGE**

**BACHELOR OF COMPUTER APPLICATION**

**RAJAJINAGAR, BANGALORE-560010**

## CERTIFICATE

This is to certify that project work entitled "**E-RATION MANAGEMENT**" has been successfully carried out by **ASHIQU SAJI (R1818214)** in partial fulfilment for the award of V semester BCA during the academic year 2020-21.

**Signature of the Guide** — **Signature of the Co-Ordinator**

(Prof.Shweta S.M) — (Dr. Parvati N Angadi)

**Name of the Examiners:** — **Signature with date**

**1.**

**2.**

# ACKNOWLEDGEMENT

I consider it is our privilege to express my sincere gratitude and respect to all those who guided us in the completion of this project.

I thank our beloved Principal **Dr. Arun Kumar Sonappanavar**, for providing us with a congenial environment to carry out our project.

I consider it is a privilege to place my deepest sense of gratitude and sincere thanks to our beloved Co-Ordinator **Dr. Parvati N Angadi**, for her sincere co-operation, guidance and supervision during the course of this project work undertaken by us.

I am thankful and wish to express my gratitude to our project guide **Ms. SHWETA S.M**, her kind co-operation and for providing valuable suggestions and constant encouragement for this project.

I express my sincere thanks and gratitude to the guidance, timely advices, support, sincere co-operation, suggestions, ideas and also constructive criticisms provided by all the teaching staff of the Computer Science Department, KLE Society's Nijalingappa College.

ASHIQU SAJI (R1818214)

# ABSTRACT

The information storing methodology includes large number of Books such as Registers, Ledgers (Purchase, Sales and Stock). These methods are time consuming and require more storage space. It is very difficult to maintain and manage these large numbers of books. Moreover there is a high risk of data loss such as that the books may catch fire or may be lost somewhere or stolen.

The main aim of developing software is to put an end to the black marketing and corruption of ration going on in various cities at present. Users must be given the right to check the price list of the ration and the quantity of ration allotted to them by the government.

This might help to put an end to the practice of holding back rations as there will be transparency in data and the services provided by the government. The present day information storing problem can be solved only by implementing digitalization and modern digital computing methods, to convert the available information into digital format and store it in a single place using computer software.

# INDEX

**1** Introduction

**2** System Analysis

**3** Software Requirement Specification

- **3.1** Existing System.
- **3.2** Proposed System.
- **3.3** Requirement Specification.
  - **3.3.1** About the technologies used.
  - **3.3.2** Hardware and Software Requirement.

**4** System Design

- **4.1** Architecture
- **4.2** Data Flow Diagrams.
- **4.3** ER Diagrams.
- **4.4** Database Design.

**5** Implementation

**6** Testing

- **6.1** About testing.
- **6.2** Types of Testing.
- **6.3** Test Cases.

**7** Application Screenshots

**8** Future Enhancements

**9** Conclusion

**10** Bibliography

# 1. INTRODUCTION

Ration cards are an official document issued by state governments in India to households that are eligible to purchase subsidized food grain from the Public Distribution System (under the National Food Security Act). They also serve as common form of identification for many Indians. Under the National Food Security Act, all state governments in India have to identify households that are eligible for purchasing subsidized food grain from the Public Distribution System and provide them with ration cards. There are two types of ration cards.

- Above Poverty Line (APL) ration cards that were issued to households living above the poverty line (as estimated by the Planning Commission). These households received 15 kilogram of food grain (based on availability).

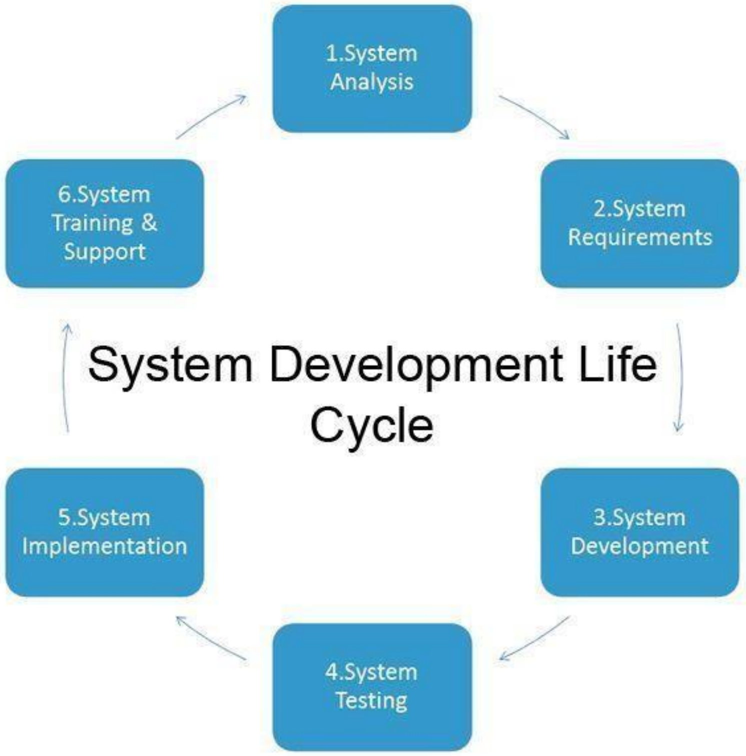
- Below Poverty Line (BPL) ration cards that were issued to households living below the poverty line. These households received 25-35 kilograms of food grain.

E-Ration Management application will help us to manage the ration available at the E-Ration Shops. The data stored in this application is monitored and moderated by a specially designated person called administrator who has the right to:

- Add New User, Delete or Modify Existing User under User Management section.
- Add New Stock or View Total Stock Available under Stock Management section.
- Edit and Update Price List of the ration under Price List section.
- Edit and Update the quantity of ration allotted under Ration Allotment section.
- Supply Ration to registered users under Purchase section.
- View Purchase History of the users under Purchase History section.

The users can login into the application by entering their username and password provided by the admin. The users can check their card details, price list of all the ration, quantity of ration allotted and purchase history of the ration purchased by them under Customer Home Page.

# 2. SYSTEM ANALYSIS



System analysis is a project management technique and a phase of system development life cycle that divides complex projects into smaller, more easily managed segments or phases. Segmenting projects allows managers to verify the successful completion of project phases before allocating resources to subsequent phases. The purpose of the systems analysis phase is to understand the requirement and build a logical model of the new system. During the next phases, data modelling, process modelling, and object modelling, we develop a logical model of business process that the system must support. The end product called as systems requirements document describes management and user requirements, alternative plans and costs, and analysis your recommendation.

System engineering and analysis encompasses requirement gathering at the system level with a small amount of top-level design and analysis. This process of analysing and gathering requirements is known as software requirement specification (SRS).The requirement gathering process intensified and focused specifically, on software. The preliminary investigation, feasibility study and the detailed investigation allows the system to comprehend the full scope of this project. Soon after testing, implementation of the developed system is followed by training.

# 3. SOFTWARE REQUIREMENT SPECIFICATION

Software Requirement Specification (SRS) is a fundamental document, which forms the foundation of the software development process. SRS not only lists the requirements of a system but also has a description of its major features. These recommendations extend the IEEE standards. The recommendations would form the basis for providing clear visibility of the product to be developed serving as baseline for execution of a contract between client and the developer.

A system requirement is one of the main steps involved in the development process. It follows after a resource analysis phase that is the task to determine what a particular software product does. The focus in this stage is one of the users of the system and not the system solutions. The result of the requirement specification document States the intention of the software, properties and constraints of the desired system.

SRS constitutes the agreement between clients and developers regarding the contents of the software product that is going to be developed. SRS should accurately and completely represent the system requirements as it makes a huge contribution to the overall project plan.

## 3.1 Existing System

The existing information storing methodology includes large number of Books such as Registers, Ledgers (Purchase, Sales and Stock). These traditional methods are time consuming and require more storage space. It is very difficult to maintain and manage these large numbers of books. Moreover there is a high risk of data loss such as that the books may catch fire or may be lost somewhere or stolen. The administrator must write down all the information in these books without making any mistakes which is very difficult and time consuming. This method does not give rights to the user to view all the information regarding ration and the services provided by the department. This traditional system is outdated and needs to be improved such that it would be easy for the administrator to store and retrieve all the information easily in an efficient and quick manner. This can be done by implementing new methodologies that supports the present digital world.

## 3.2 Proposed System

The existing traditional information storing methodology can be improved by the process of digitalization. This can be easily done using a computer-based system. Our proposed system is a user-friendly computer-based system which is more effective and efficient in managing, maintaining and accessing large information using computer system. The main features of the proposed system which overcomes the drawbacks of the existing system are as follows:-

- The proposed system is very Fast, Efficient and Portable.
- The digital information can be easily backed up and restored whenever required which avoids loss of data.
- This system serves both the administrators and the users.
- It provides a user-friendly interface which helps the users to easily use the application.
- The administrators are those E-Ration shopkeepers who use this application to store, retrieve, manage and maintain all the information regarding user management, stock management, price list, ration allotment, purchase details and purchase history.
- The users are those who come to take away ration regularly every month and use the application to view their card details, ration price list, ration allotment and purchase history.

## 3.3 Requirement Specification.

### 3.3.1 About the Technologies Used.

**Introduction to the .NET Framework**

The .Net Framework is one of the most widely used software development environment in today’s programming world. Before its introduction, programmers had to face a lot of difficulties to integrate the code written using different programming languages. This was due to the reason that each language used a different execution environment to execute the code written in that language. For Example, code written using Visual Basic 6.0 requires a different execution environment for execution than that is required by code written using Visual C++. With the .NET Framework, Microsoft has provided programmers a single platform for developing applications using different programming languages, such as Visual Basic, Visual C# and Visual C++. The .NET Framework reduces the difficulty involved in building large, reliable applications.

**Microsoft SQL Server 2008 Express Edition**

Microsoft SQL Server 2008 is the latest version of Microsoft SQL Server, which is a Relational Database Management System

Allows you to store any kind of data, such as audio files, images, rich media. In addition, SQL Server 2008 provides you rich set of integrated services that enables you to perform various types of data manipulation operations, such as querying, searching, reporting, and analysing.

**Features of SQL Server 2008**

- New data types
- Large User-Defined Types (UDTs)
- Sparse columns
- Table-valued parameters
- Integrated full-text search
- IntelliSense

### 3.3.2 Hardware and Software requirements.

## Hardware Requirements

- ⇨ **Processor**
    - Intel Core 2 duo (minimum)
    - Intel Core i3 and above (recommended)
- ⇨ **RAM**
    - 2GB (minimum)
    - 4GB (recommended)
- ⇨ **Disk Space**
    - 90GB (minimum)
    - 180GB (recommended)

## Software Requirements

- ⇨ **Operating System**
    - Windows 7 or above
- ⇨ **Front End**
    - Microsoft Visual Studio 2010
- ⇨ **Back End**
    - Microsoft SQL Server 2008

# 4. SYSTEM DESIGN

## 4.1 Architecture

The type of architecture used in this application is two-tier architecture which is similar to Client-Server architecture, where communication takes place between client and server.
In this type of software architecture, the presentation layer or user interface layer runs on the client side while dataset layer gets executed and stored on server side. There is no Business logic layer or immediate layer in between client and server.

The main reasons for considering two-tier architecture for the application are as follows:

- Applications can be easily developed due to simplicity.
- Maximum user satisfaction is gained with accurate and fast prototyping of applications through robust tools
- Since this contains static business rules it's more applicable for homogenous environments.
- Database server and business logic is physically close, which offers higher performance.

## 4.2 Data Flow Diagrams (DFD's)

DFD graphically representing the functions, or processes, which capture, manipulate, store and distribute data between a system and its environment and between components of a system. The visual representation makes it a good communication tool between User and System designer. A structure of DFD allows starting from a broad overview and expands it to a hierarchy of detailed diagrams. DFD has often been used due to the following reasons:

- Logical information flow of the system.
- Determination Of physical system construction requirements.
- Simplicity of notation.
- Establishment of manual and automated systems requirements.

The following symbols are used in the Data Flow Diagram:

1)

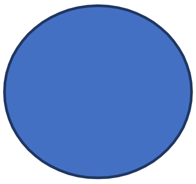
This arrow diagram represents the flow of data in one direction.

2)

This arrow diagram represents the flow of data in bi-direction.

3)

This rectangular symbol represents the activity of the application.

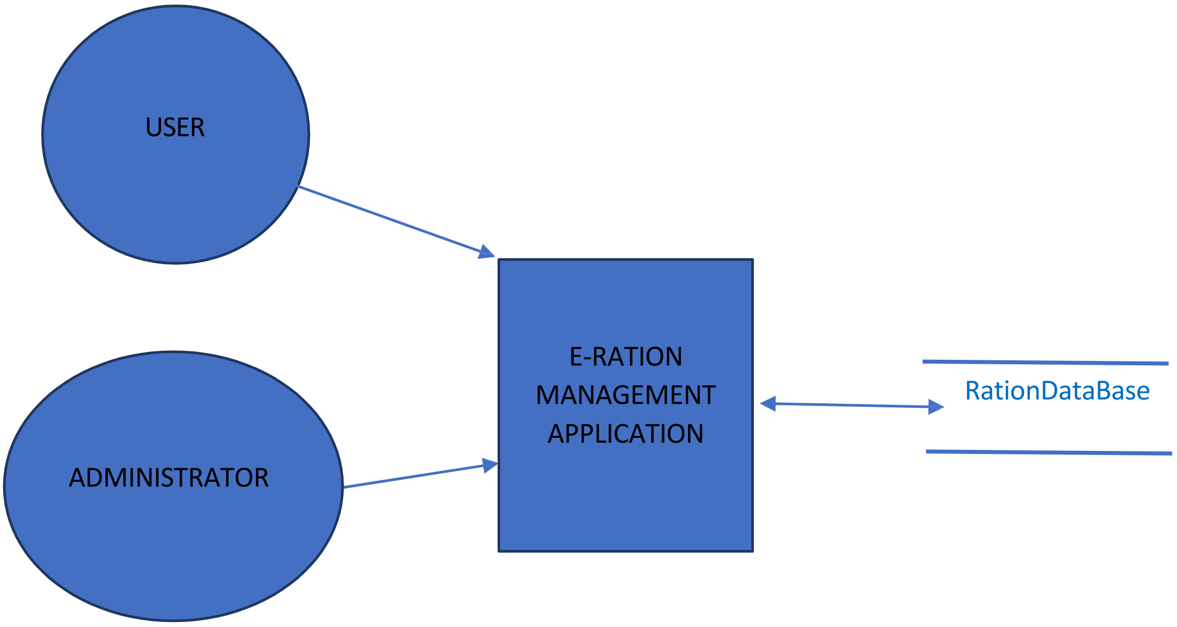
4) 

This circular symbol denotes the users / source of inputs to the activity of the application.

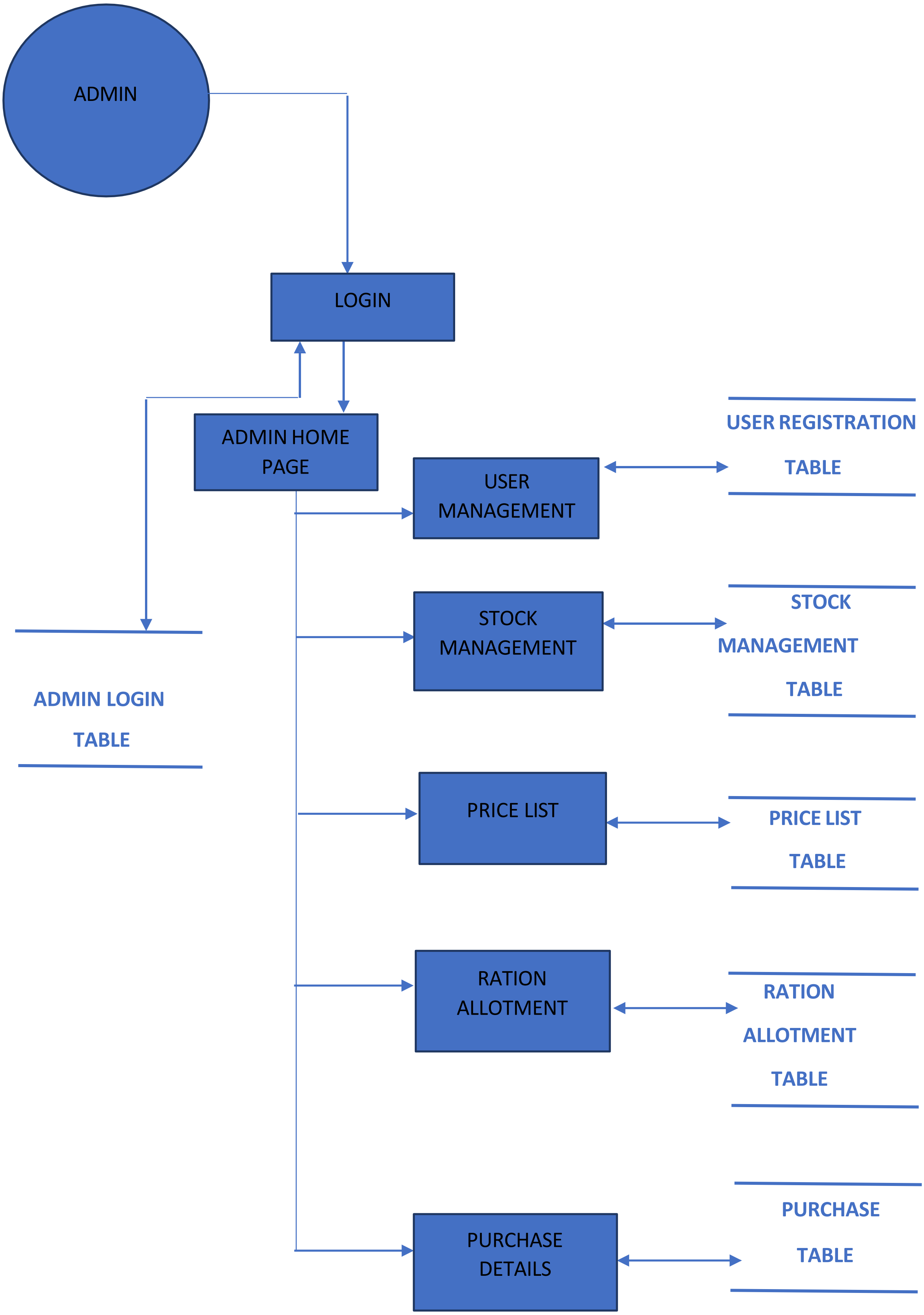
5)

This symbol denotes the Database and Database Tables of the application.

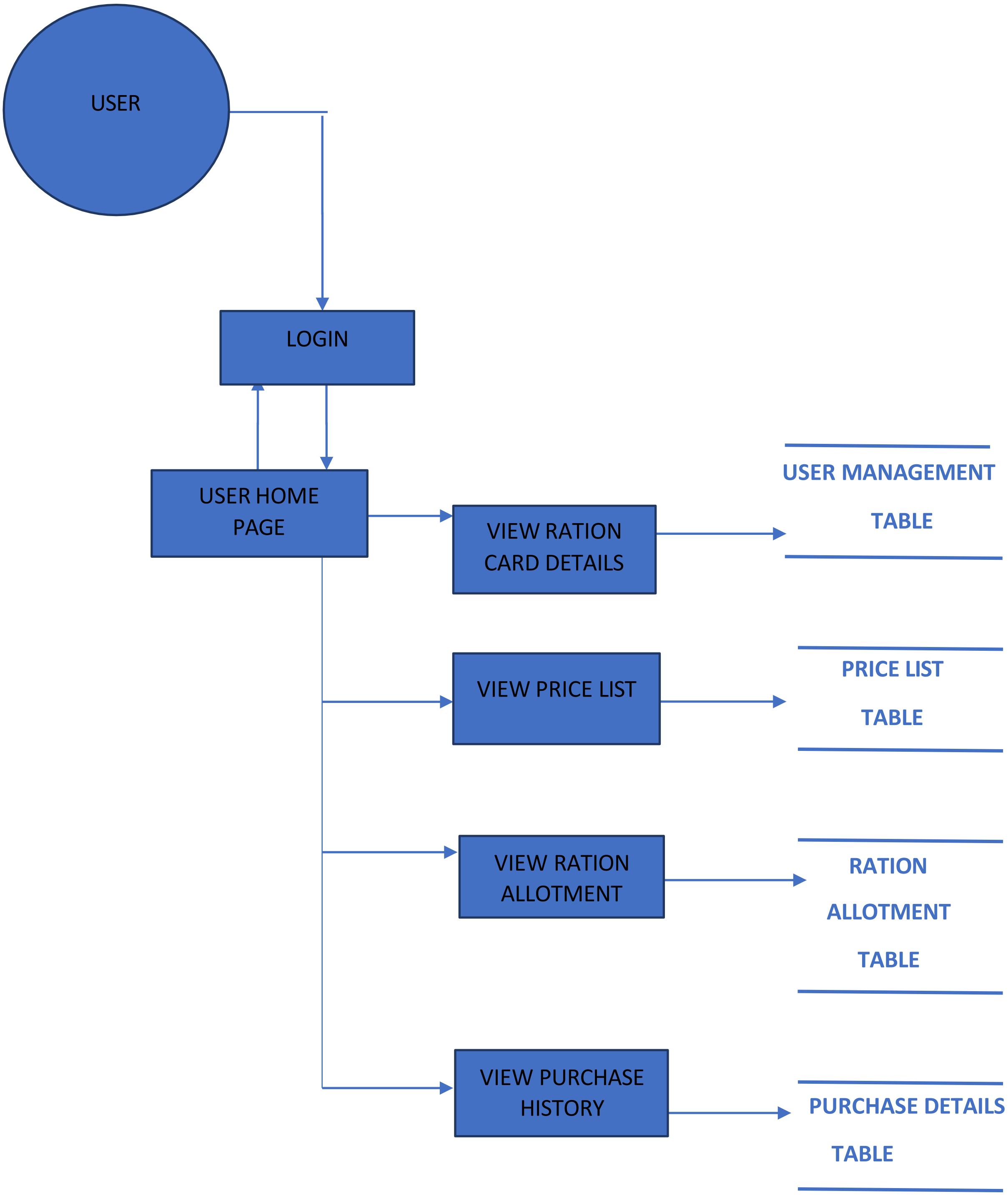
**DATAFLOW DIAGRAM: LEVEL 0**



## DATAFLOW DIAGRAM: LEVEL 1 A – ADMIN ACTIVITY



## DATAFLOW DIAGRAM: LEVEL 1 A – USER ACTIVITY



## 4.3 E-R Diagrams (Entity-Relationships Diagrams)

Entity Relationship Diagram depicts the various relationships among entities, considering each objective as entity. Entity relationships are described by their dependence on each other, as well as the extent of the relationship between the data stores. It depicts the relationship between data Objects. While ER models are mostly developed for designing relational databases in terms of concept visualization and in terms of physical database design, there are still other situations when ER diagrams can help. The ER diagram is a notation that is used to conduct the data modelling activity.

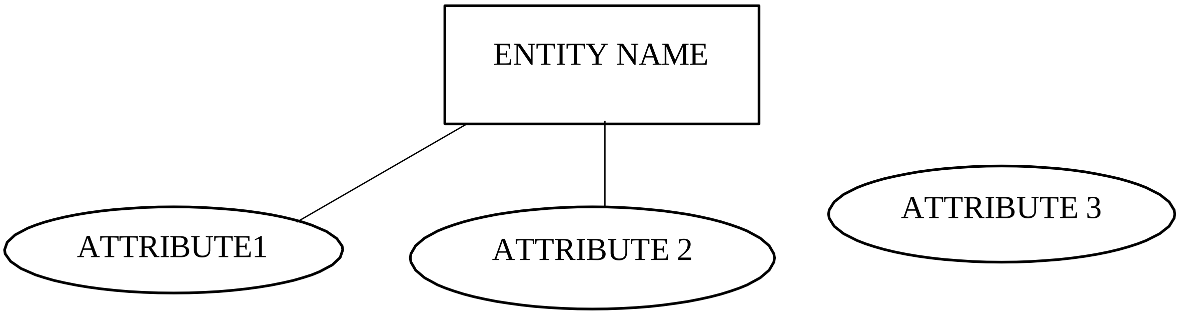
### i) Entity

An ERD entity is a **definable thing or concept within a system**. In ER models, an entity is shown as a rectangle, with its name on top and its attributes listed in the body of the entity shape. i.e.

ENTITY NAME

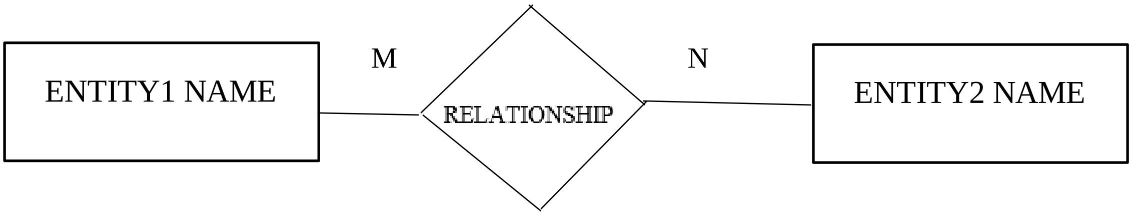
### ii) Attributes

Also known as a column, an attribute is a **property or characteristic of the entity that holds it**. An attribute has a name that describes the property and a type that describes the kind of attribute it is, such as varchar for a string, and int for integer. It is represented by an oval shape in the ER Diagram. i.e.

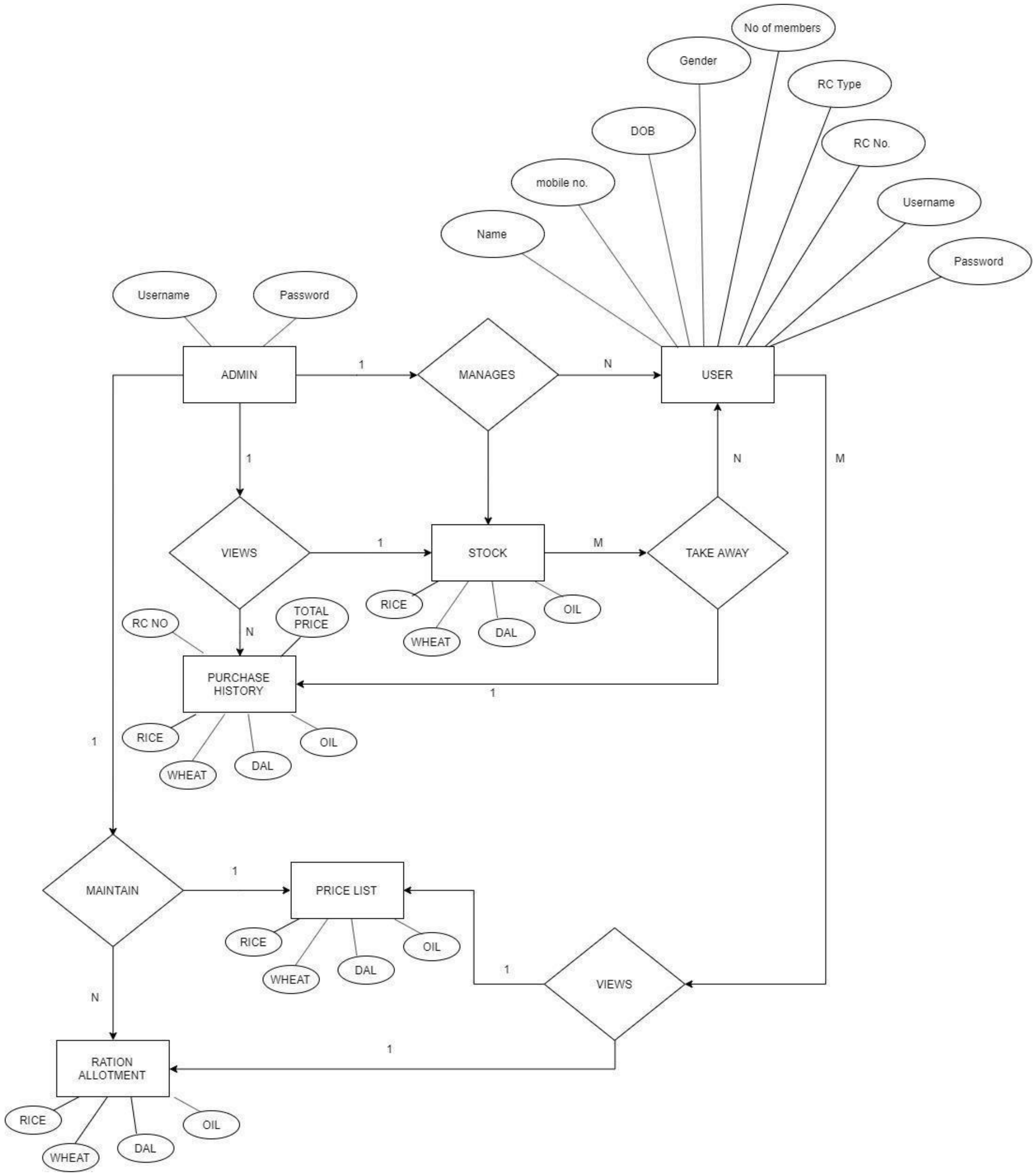


## iii) Relationship

A relationship between two entities signifies that the **two entities are associated with each other**. It is represented by a Rhombus shape in ER Diagram.



# ENTITY RELATION DIAGRAM OF THE APPLICATION



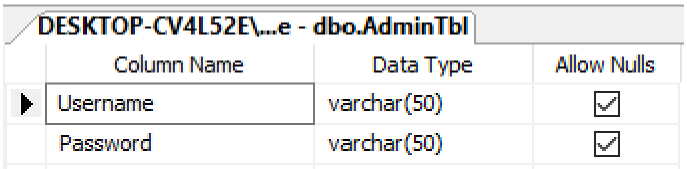
## 4.4 Database Design

It is a process of designing the database file, which is the key source of the information in the system. The objective of database is to design is to provide storage and it contributes to the overall efficiency of the system. The file should properly design and planned for collection, accumulation, editing and retrieving the required information.

The primary objective of a database design are fast response time to inquiries, more information at low cost, control of redundancy, clarity and ease of use, accuracy and integrity of the system, fast recovery and availability of powerful end-user languages. The theme behind a database is to handle information as an integrated whole thus the main objective is to make information as access easy, quick, inexpensive and flexible for the users.

The following database tables are created in Microsoft SQL Server 2008 and accessed in our application.

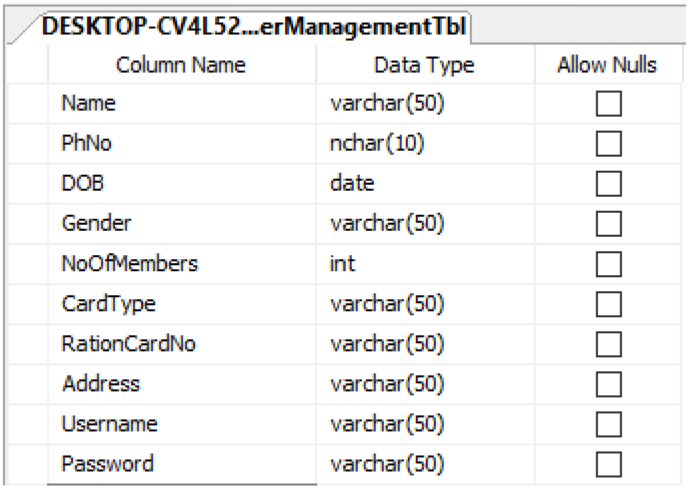
- ADMIN LOGIN TABLE



DESKTOP-CV4L52E\...e - dbo.AdminTbl

| Column Name | Data Type | Allow Nulls |
|---|---|---|
| Username | varchar(50) | ☑ |
| Password | varchar(50) | ☑ |

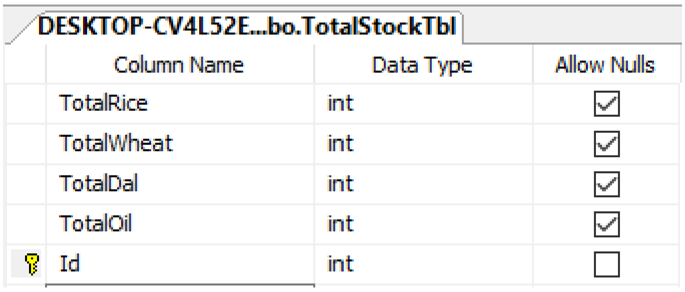
- USER MANAGEMENT TABLE



DESKTOP-CV4L52...erManagementTbl

| Column Name | Data Type | Allow Nulls |
|---|---|---|
| Name | varchar(50) | ☐ |
| PhNo | nchar(10) | ☐ |
| DOB | date | ☐ |
| Gender | varchar(50) | ☐ |
| NoOfMembers | int | ☐ |
| CardType | varchar(50) | ☐ |
| RationCardNo | varchar(50) | ☐ |
| Address | varchar(50) | ☐ |
| Username | varchar(50) | ☐ |
| Password | varchar(50) | ☐ |

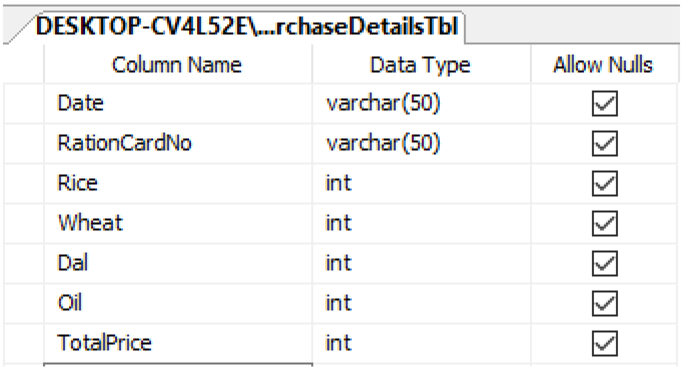
- STOCK MANAGEMENT TABLE



DESKTOP-CV4L52E...bo.TotalStockTbl

| | Column Name | Data Type | Allow Nulls |
|---|---|---|---|
| | TotalRice | int | ☑ |
| | TotalWheat | int | ☑ |
| | TotalDal | int | ☑ |
| | TotalOil | int | ☑ |
| 🔑 | Id | int | □ |

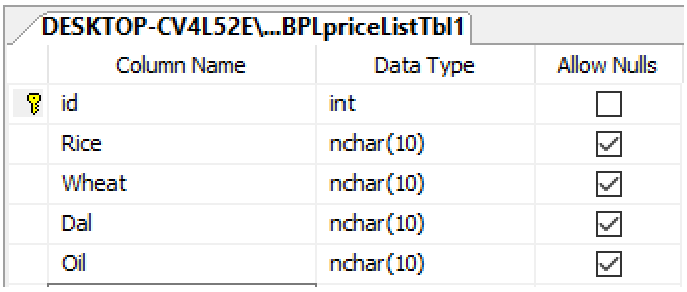
- PURCHASE DETAILS TABLE



DESKTOP-CV4L52E\...rchaseDetailsTbl

| | Column Name | Data Type | Allow Nulls |
|---|---|---|---|
| | Date | varchar(50) | ☑ |
| | RationCardNo | varchar(50) | ☑ |
| | Rice | int | ☑ |
| | Wheat | int | ☑ |
| | Dal | int | ☑ |
| | Oil | int | ☑ |
| | TotalPrice | int | ☑ |

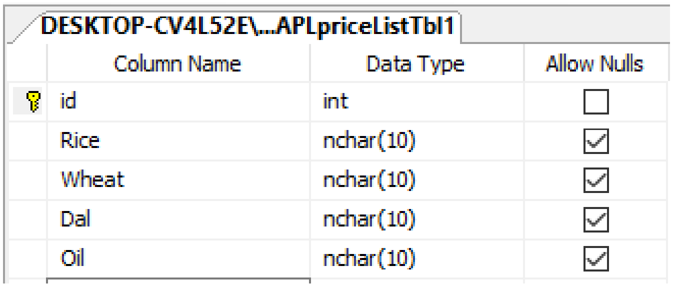
- BPL PRICE LIST TABLE



DESKTOP-CV4L52E\...BPLpriceListTbl1

| | Column Name | Data Type | Allow Nulls |
|---|---|---|---|
| 🔑 | id | int | □ |
| | Rice | nchar(10) | ☑ |
| | Wheat | nchar(10) | ☑ |
| | Dal | nchar(10) | ☑ |
| | Oil | nchar(10) | ☑ |

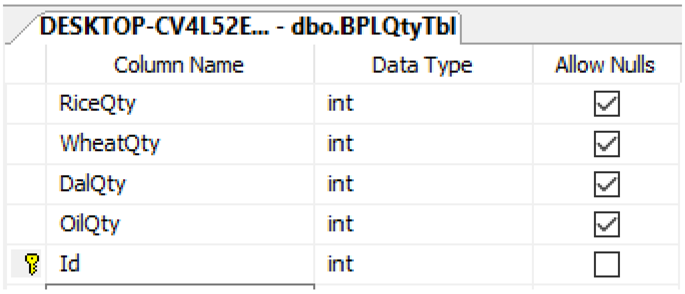
- APL PRICE LIST TABLE



DESKTOP-CV4L52E\...APLpriceListTbl1

| | Column Name | Data Type | Allow Nulls |
|---|---|---|---|
| 🔑 | id | int | □ |
| | Rice | nchar(10) | ☑ |
| | Wheat | nchar(10) | ☑ |
| | Dal | nchar(10) | ☑ |
| | Oil | nchar(10) | ☑ |

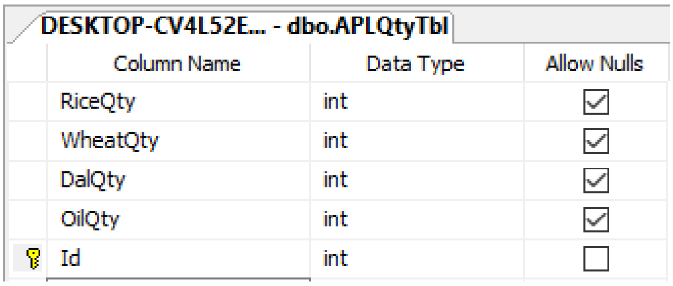
- BPL RATION ALLOTMENT TABLE



DESKTOP-CV4L52E... - dbo.BPLQtyTbl

| | Column Name | Data Type | Allow Nulls |
|---|---|---|---|
| | RiceQty | int | ☑ |
| | WheatQty | int | ☑ |
| | DalQty | int | ☑ |
| | OilQty | int | ☑ |
| 🔑 | Id | int | □ |

- APL RATION ALLOTMENT TABLE



DESKTOP-CV4L52E... - dbo.APLQtyTbl

| | Column Name | Data Type | Allow Nulls |
|---|---|---|---|
| | RiceQty | int | ☑ |
| | WheatQty | int | ☑ |
| | DalQty | int | ☑ |
| | OilQty | int | ☑ |
| 🔑 | Id | int | □ |

# 5. IMPLEMENTATION

Implementation is the process of converting a or a revised system design into an operational one. The Objective is to put the new or revised system that has been tested into operation while holding costs, risks, and personal irritation to the minimum. A critical aspect or the implementation process is to ensure that there will be no disrupting the functioning or the organization. The best method for gaining control while implanting any new system would be to use well planned test for testing all new programs. Before production files are used to test live data, text files must be created on the old system, copied over to the new system, and used for the initial test of each program.

- **Coding of Admin Login Form**

```
Public Class Adminlogin
    Dim connection As New SqlConnection("Server = DESKTOP-CV4L52E\SQLEXPRESS ;
Database = RationDataBase ; Integrated security = true")
    Private Sub AdminLoginBtn_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles AdminLoginBtn.Click
        Dim command As New SqlCommand("select * from AdminTbl where Username
COLLATE Latin1_General_CS_AS = @username and Password COLLATE
Latin1_General_CS_AS= @password", connection)
        command.Parameters.Add("@username", SqlDbType.VarChar).Value =
AdminUsername.Text
        command.Parameters.Add("@password", SqlDbType.VarChar).Value =
AdminPassword.Text
        Dim adapter As New SqlDataAdapter(command)
        Dim table As New DataTable()
        adapter.Fill(table)
        If table.Rows.Count() <= 0 Then
            MessageBox.Show("Invalid Username Or Password", "Error")
        Else
            Dim Admin As New Adminhomepage()
            Me.Hide()
            Admin.Show()
        End If
    End Sub

    Private Sub eye_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles eye.Click
        If AdminPassword.UseSystemPasswordChar = True Then
            AdminPassword.UseSystemPasswordChar = False
        Else
            AdminPassword.UseSystemPasswordChar = True
        End If
    End Sub

    Private Sub PictureBox1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox1.Click
        Home.Show()
        Me.Hide()
    End Sub

    Private Sub Label4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Label4.Click
        Me.Hide()
        AdminPasswordchange.Show()
    End Sub

    Private Sub Label4_MouseHover(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Label4.MouseHover
        Label4.ForeColor = Color.GreenYellow
    End Sub

    Private Sub Label4_MouseLeave(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Label4.MouseLeave
        Label4.ForeColor = Color.Black
    End Sub

    Private Sub PictureBox4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox4.Click
        End
    End Sub
End Class
```

- **Coding of User Login Form**

```
Public Class Userlogin
    Dim connection As New SqlConnection("Server = DESKTOP-CV4L52E\SQLEXPRESS ;
Database = RationDataBase ; Integrated security = true")
    Private Sub Login_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Login.Click
        Dim command As New SqlCommand("select * from UserManagementTbl where
Username COLLATE Latin1_General_CS_AS= @username and Password COLLATE
Latin1_General_CS_AS= @password", connection)
        command.Parameters.Add("@username", SqlDbType.VarChar).Value = username.Text
        command.Parameters.Add("@password", SqlDbType.VarChar).Value = password.Text
        Dim adapter As New SqlDataAdapter(command)
        Dim table As New DataTable()
        adapter.Fill(table)
        If table.Rows.Count() <= 0 Then
            MessageBox.Show("Invalid Username Or Password", "Error")
        Else
            Dim Customer As New Customerhomepage()
            Me.Hide()
            Customer.Show()
        End If
    End Sub
    Private Sub exitfrm_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles exitfrm.Click
        Home.Show()
        Me.Hide()
    End Sub

    Private Sub PictureBox3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles eye.Click
        If password.UseSystemPasswordChar = True Then
            password.UseSystemPasswordChar = False
        Else
            password.UseSystemPasswordChar = True
        End If
    End Sub

    Private Sub PictureBox4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox4.Click
        Home.Show()
        Me.Hide()
    End Sub

    Private Sub PictureBox3_Click_1(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox3.Click
        End
    End Sub
End Class
```

- **Coding of User Management Form**

```
Imports System.Data.SqlClient
Public Class UserManageForm
```

```
    Dim connection As New SqlConnection("Server = DESKTOP-CV4L52E\SQLEXPRESS ;
Database = RationDataBase ; Integrated security = true")
    Dim name As String
    Sub filterrecords()
        Dim command As New SqlCommand("select * from UserManagementTbl", connection)
        Dim adapter = New SqlDataAdapter(command)
        Dim table As New DataTable
        adapter.Fill(table)
        DataGridView1.DataSource = table
    End Sub


    Private Sub UserManageForm_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
        filterrecords()
    End Sub


    Private Sub DataGridView1_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles DataGridView1.Click

        Try
            Dim i As Integer
            i = DataGridView1.CurrentRow.Index
            Me.name1.Text = DataGridView1.Item(0, i).Value
            Me.phno.Text = DataGridView1.Item(1, i).Value
            Me.dob.Value = DataGridView1.Item(2, i).Value
            Me.gender.Text = DataGridView1.Item(3, i).Value
            Me.nom.Text = DataGridView1.Item(4, i).Value
            Me.Cardtype.Text = DataGridView1.Item(5, i).Value
            Me.rcno.Text = DataGridView1.Item(6, i).Value
            Me.address.Text = DataGridView1.Item(7, i).Value
            Me.username.Text = DataGridView1.Item(8, i).Value
            Me.password.Text = DataGridView1.Item(9, i).Value
            name = username.Text
        Catch ex As Exception
        End Try
    End Sub

    Private Sub addbtn_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles addbtn.Click
        If phno.Text.Length < 10 Then
            MsgBox("Enter Complete Mobile Number!")
        Else
            Try
                Dim command As New SqlCommand("insert into UserManagementTbl(Name,
PhNo, DOB, Gender, NoOfMembers,CardType, RationCardNo, Address,Username,
Password )
values(@name,@phno,@dob,@gender,@mno,@cardtype,@Rcard,@add,@username,@pass
word)", connection)
                command.Parameters.Add("@name", SqlDbType.VarChar).Value = name1.Text
                command.Parameters.Add("@phno", SqlDbType.NChar).Value = phno.Text
                command.Parameters.Add("@dob", SqlDbType.Date).Value = dob.Value
                command.Parameters.Add("@gender", SqlDbType.VarChar).Value =
gender.SelectedItem
                command.Parameters.Add("@mno", SqlDbType.Int).Value = Val(nom.Text)
                command.Parameters.Add("@cardtype", SqlDbType.VarChar).Value =
Cardtype.SelectedItem
                command.Parameters.Add("@Rcard", SqlDbType.VarChar).Value = rcno.Text
                command.Parameters.Add("@add", SqlDbType.VarChar).Value = address.Text
                command.Parameters.Add("@username", SqlDbType.VarChar).Value =
username.Text
                command.Parameters.Add("@password", SqlDbType.VarChar).Value =
password.Text
                connection.Open()
                Dim command2 As New SqlCommand("select * from UserManagementTbl where
Username=@user", connection)
                command2.Parameters.Add("@user", SqlDbType.VarChar).Value = username.Text
                Dim adapter As New SqlDataAdapter(command2)
                Dim table As New DataTable()
                adapter.Fill(table)

                If table.Rows.Count() > 0 Then
                    MessageBox.Show("Username Already Exists", "Error")
                Else
                    If command.ExecuteNonQuery() = 1 Then
                        MessageBox.Show("User Added Successfully!!", "Success")
                        filterrecords()
                    Else
                        MessageBox.Show("Sorry , User Cannot Be Added!! Try again Later", "Try
    Again")
                    End If

                End If
                connection.Close()
            Catch ex As Exception
                MessageBox.Show("Enter Data Into All The Fields", "Error")
            End Try
        End If
        name1.Text = ""
        phno.Text = ""
        dob.Text = ""
        gender.SelectedIndex = -1
        nom.Text = ""
        Cardtype.SelectedIndex = -1
        rcno.Text = ""
        address.Text = ""
        username.Text = ""
        password.Text = ""
        GroupBox1.Enabled = False
    End Sub
```

```
    Private Sub deletebtn_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles deletebtn.Click
        Dim name As String
        name = Me.username.Text
        Try
            Dim command As New SqlCommand("delete from UserManagementTbl where
Username=@username ", connection)
            connection.Open()
            command.Parameters.Add("@username", SqlDbType.VarChar).Value = name
            If command.ExecuteNonQuery() = 1 Then
                MessageBox.Show("Sucessfully Deleted!!", "Deleted")
                filterrecords()
            Else
                MessageBox.Show("Sorry , Cannot Be Deleted", "Error")
            End If
            connection.Close()
        Catch ex As Exception
            MessageBox.Show("No More Rows To Delete!!!!!", "Error")
        End Try
        name1.Text = ""
        phno.Text = ""
        dob.Text = ""
        gender.SelectedIndex = -1
        nom.Text = ""
        Cardtype.SelectedIndex = -1
        rcno.Text = ""
        address.Text = ""
        username.Text = ""
        password.Text = ""
        GroupBox1.Enabled = False
    End Sub

    Private Sub updatebtn_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles updatebtn.Click
```

```
Dim command As New SqlCommand("update UserManagementTbl set Name=@name, PhNo=@phno, DOB=@dob, Gender=@gender, NoOfMembers=@mno,CardType=@cardtype, RationCardNo=@Rcard, Address=@add,Username=@username1, Password=@password where Username=@username2 ", connection)
command.Parameters.Add("@name", SqlDbType.VarChar).Value = name1.Text
command.Parameters.Add("@phno", SqlDbType.NChar).Value = phno.Text
command.Parameters.Add("@dob", SqlDbType.Date).Value = dob.Value
command.Parameters.Add("@gender", SqlDbType.VarChar).Value = gender.SelectedItem
command.Parameters.Add("@mno", SqlDbType.Int).Value = Val(nom.Text)
command.Parameters.Add("@cardtype", SqlDbType.VarChar).Value = Cardtype.SelectedItem
command.Parameters.Add("@Rcard", SqlDbType.VarChar).Value = rcno.Text
command.Parameters.Add("@add", SqlDbType.VarChar).Value = address.Text
command.Parameters.Add("@username1", SqlDbType.VarChar).Value = username.Text
command.Parameters.Add("@username2", SqlDbType.VarChar).Value = name
command.Parameters.Add("@password", SqlDbType.VarChar).Value = password.Text
connection.Open()
Dim command2 As New SqlCommand("select * from UserManagementTbl where Username = @username ", connection)
command2.Parameters.Add("@username", SqlDbType.VarChar).Value = username.Text
Dim adapter As New SqlDataAdapter(command2)
Dim table As New DataTable()
adapter.Fill(table)
If table.Rows.Count() > 1 Then
    MessageBox.Show("Username Already Taken!!!!", "Error")
Else
    If command.ExecuteNonQuery() = 1 Then
        MessageBox.Show("Sucessfully Updated!!", "Updated")
        filterrecords()
```

```
            Else
                MessageBox.Show("Sorry , Cannot Be Updated", "Error")
            End If
        End If
        connection.Close()
        name1.Text = ""
        phno.Text = ""
        dob.Text = ""
        gender.SelectedIndex = -1
        nom.Text = ""
        Cardtype.SelectedIndex = -1
        rcno.Text = ""
        address.Text = ""
        username.Text = ""
        password.Text = ""
        GroupBox1.Enabled = False
        updatebtn.Enabled = False
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles addbtn1.Click
        GroupBox1.Enabled = True
        name1.Text = ""
        phno.Text = ""
        dob.Text = ""
        gender.SelectedIndex = -1
        nom.Text = ""
        Cardtype.SelectedIndex = -1
        rcno.Text = ""
        address.Text = ""
        username.Text = ""
        password.Text = ""
    End Sub
    Private Sub modifybtn_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles modifybtn.Click
        MessageBox.Show("Select An User From The Datagrid To Modify It", "Select User")
        GroupBox1.Enabled = True
        addbtn.Enabled = False
        updatebtn.Enabled = True
        addbtn1.Enabled = False
        deletebtn.Enabled = False
        modifybtn.Enabled = False
    End Sub

    Private Sub cancelbtn_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles cancelbtn.Click
        GroupBox1.Enabled = False
        name1.Text = ""
        phno.Text = ""
        dob.Text = ""
        gender.SelectedIndex = -1
        nom.Text = ""
        Cardtype.SelectedIndex = -1
        rcno.Text = ""
        address.Text = ""
        username.Text = ""
        password.Text = ""
    End Sub

    Private Sub PictureBox1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox1.Click
        Adminhomepage.Show()
        Me.Hide()
    End Sub
    Private Sub phno_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles phno.KeyPress
        If phno.Text = "" Then
            MobNo1stDigit(e)
        Else
            If Not (Asc(e.KeyChar) = 8) Then
                If (phno.Text.Length <= 9) Then
                    MobileNumberValidation(e)
                Else
                    e.KeyChar = ChrW(0)
                    MsgBox("Invalid Mobile Number, Please Enter Correctly!!")
                End If
            End If
        End If
    End Sub
    Private Sub name1_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles name1.KeyPress
        If name1.Text = " " Then
            MsgBox("Please Enter Your Name Correctly!!")
            name1.Text = ""
        Else
            TextBoxValidation(e)
        End If
    End Sub
    Private Sub nom_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles nom.KeyPress
        NOMValidation(e)
    End Sub
    Private Sub PictureBox2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles PictureBox2.Click
        End
    End Sub
End Class
```

# 6. TESTING

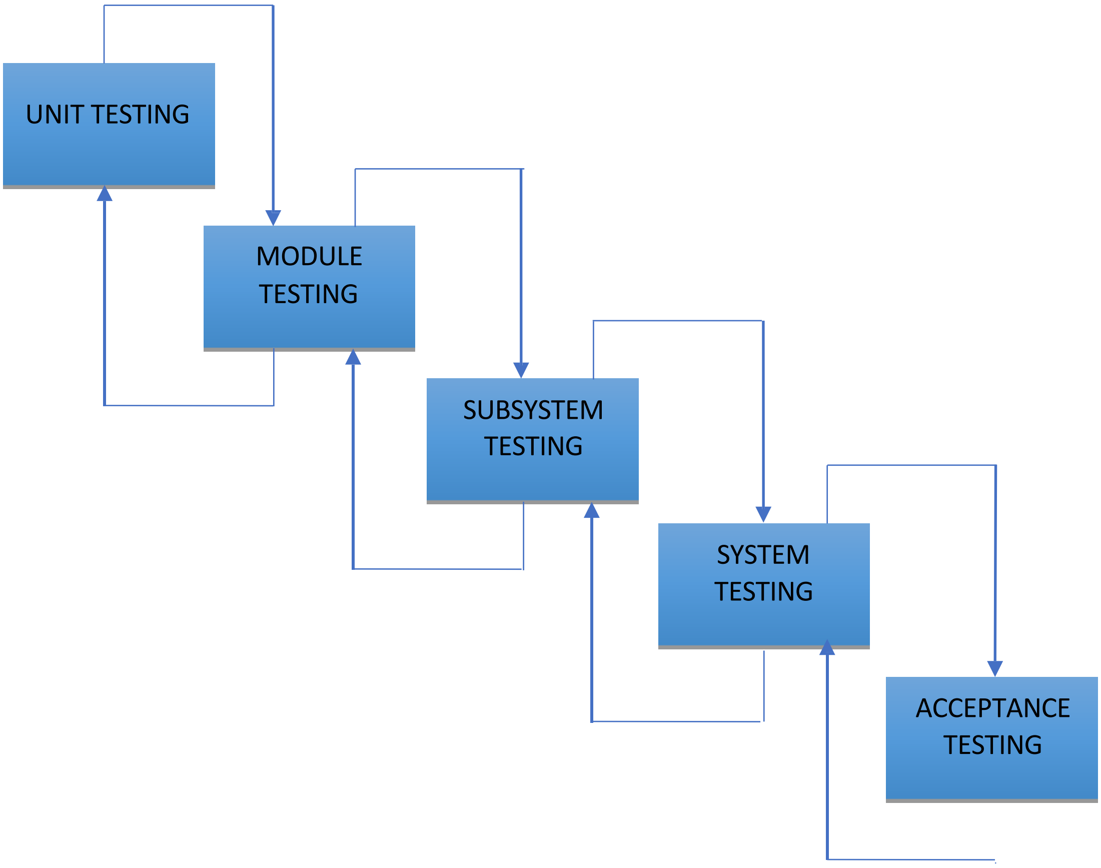
## 6.1 About Testing

Testing is the major process involved in software quality assurance (QA) is iterative Here test data is prepared and is used to test the modules individually. System testing ensures that all components or the system function - as a unit by actually forcing the system to fail.

The test causes should be planned before testing begins. Then as the testing progresses, testing shifts focus in an attempt to find errors in integrated clusters of modules. The philosophy behind testing is to find errors. Actually, testing is the state of implementation that is aimed at ensuring that the system works actually and efficiently before implementation.

Testing is done for each module. After testing all the modules, the modules are integrated and testing of the final system is done with the test data, specially designed to show that the system will operate successfully in all its aspects conditions. The procedure level testing is made first. By giving improper inputs, the errors occurred are noted and eliminated. Thus, the system testing is a confirmation that all is correct and an opportunity to show the user that the system works. The final step involves Validation testing, which determines whether the software function as the user expected. The end-user rather than the system developer conduct this test most software developers as a process called "Alpha and Beta test" to uncover that only the end user seems able to find. This is the final Step in system life cycle. Here we implement the tested error-free system into real-life environment and make necessary changes, which runs in an online fashion. Here system maintenance is done every month or year based on company policies, and is checked for errors like runtime errors, long run errors and other maintenances like table verification and during the requirement analysis and design, the output is a document that is usually textual and non-executable. After the coding phase, computer programs are available that can be executed for testing purpose. This implies that testing not only has to uncover errors introduced during coding, but also errors introduced during the previous phases.

## 6.2 Types of Testing



**The various types of testing done are:**

- Unit testing
- Integration testing
- Validation testing
- System testing
- Acceptance testing

### 1) Unit Testing

Unit testing verification efforts on the smallest unit of software design, module. This is known as “Module Testing". The modules are tested separately. This testing is carried out during stage itself. In these testing Steps, each module is found to be working satisfactorily as regard to the expected output from the module.

## 2) Integration Testing

Integration testing is a systematic technique for constructing tests to uncover error within the interface. In the project, all the modules are combined and then the entire is tested as a whole. In the integration-testing Step, all the error uncovered is corrected for the next testing steps.

## 3) Validation Testing

To uncover functional errors, that is, to check whether functional characteristics confirm to specification or not specified.

## 4) System Testing

Once individual module testing completed, modules are assembled to perform as a system. Then the top down testing, which begins from upper level to lower level module testing, to done to check whether the entire system is performing satisfactorily. After unit and integration testing are over then the system as whole is tested. There are two general strategies for system testing. They are:

- Code Testing
- Specification Testing

### • Code Testing

This strategy examines the logic of the program. A path is a specific combination of conditions handled by the program. Using this strategy, every path through the program is tested.

### • Specification Testing

This strategy examines the specifications stating what the program should do and how it should perform under various conditions. The test cases are developed for each condition of developed System and processed. It is found that the system developed perform according to its specified requirements. The system is used experimentally to ensure that the software will run according to tits specification and in the way user expects. Specification Testing is done successfully by entering various types of end data. It is checked for both valid and invalid data and found System is working properly as per requirement.

### 5) Acceptance Testing

When the system has no measure problem with its accuracy, the system passes through a final acceptance test. This test confirms that the system needs the original goal, Objective and requirements established during analysis. If the system fulfils all the requirements, it is finally acceptable and ready for operation.

## 6.3 Test Cases

### Testing Module: Login Form

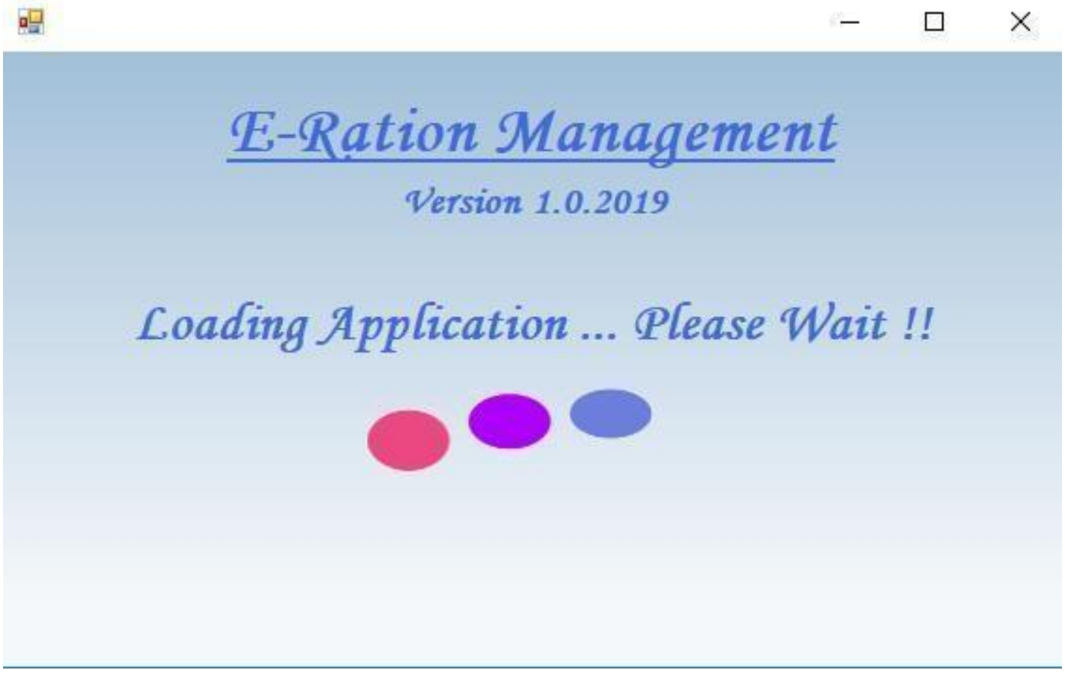
| SL NO. | TEST CASE NAME | INPUT | EXPECTED OUTPUT | ACTUAL OUTPUT | VALID / INVALID |
|---|---|---|---|---|---|
| 1 | Username | --BLANK-- | If Username Field is not blank then it is valid. | Username Field is not blank. | INVALID |
|  | Username | admin | If Username Field is not blank then it is valid. | Username Field is not blank. | VALID |
| 2 | Password | --BLANK-- | If Password Field is not blank then it is valid. | Password Field is not blank. | INVALID |
|  | Password | admin | If Password Field is not blank then it is valid. | Password Field is not blank. | VALID |

## Testing Module: User Management Form

| SL NO. | TEST CASE NAME | INPUT | EXPECTED OUTPUT | ACTUAL OUTPUT | VALID / INVALID |
|---|---|---|---|---|---|
| 1 | Mobile No | 82961173 | If Mobile No. Field contains exactly 10 digits then it is valid. | Mobile No contains 10 digits. | INVALID |
|  | Mobile No | 8296117328 | If Mobile No. Field contains exactly 10 digits then it is valid. | Mobile No contains 10 digits. | VALID |
| 2 | Mobile No | 0829611732 | If Mobile No. does not start with digit 0 then it is valid. | Mobile No does not start with digit 0. | INVALID |
|  | Mobile No | 8296117328 | If Mobile No. does not start with digit 0 then it is valid. | Mobile No does not start with digit 0. | VALID |
| 3 | Username already exists | Username : rajesh<br>New username:rajesh | If User name does not exists then it is valid | User name doesn’t exist | INVALID |
|  | Username already exists | Username : rajesh<br>New username: raj | If User name does not exists then it is valid | User name already exists | VALID |

# 7. APPLICATION SCREENSHOTS.

- **Welcome Screen Form**

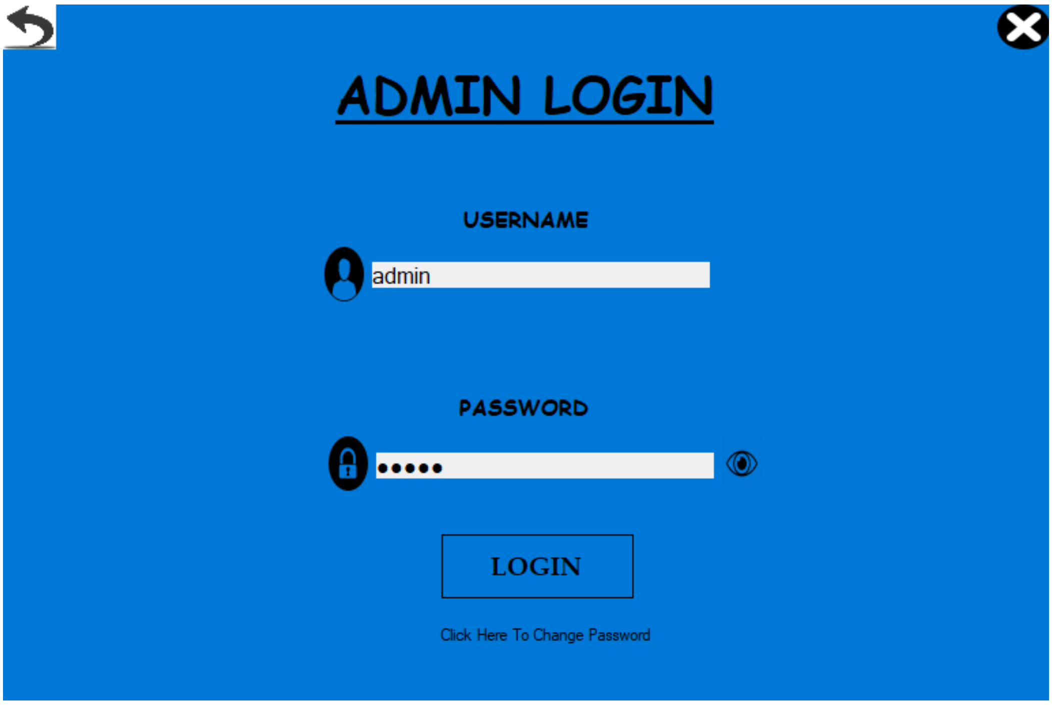


E-Ration Management

Version 1.0.2019

Loading Application ... Please Wait !!

- **Admin Login Form**



ADMIN LOGIN

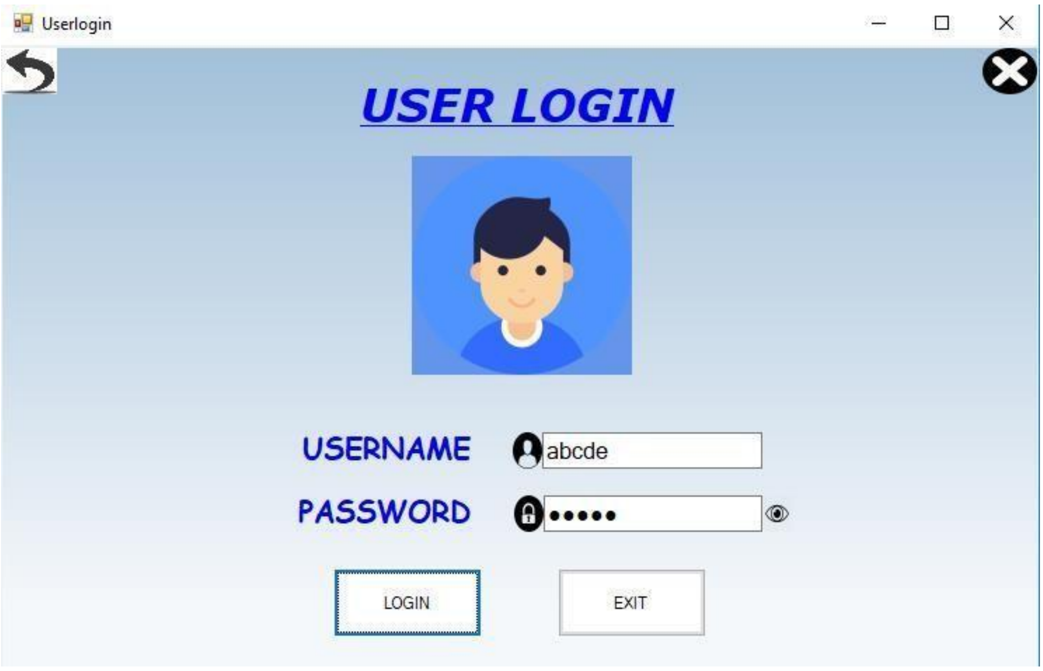
USERNAME

admin

PASSWORD

•••••

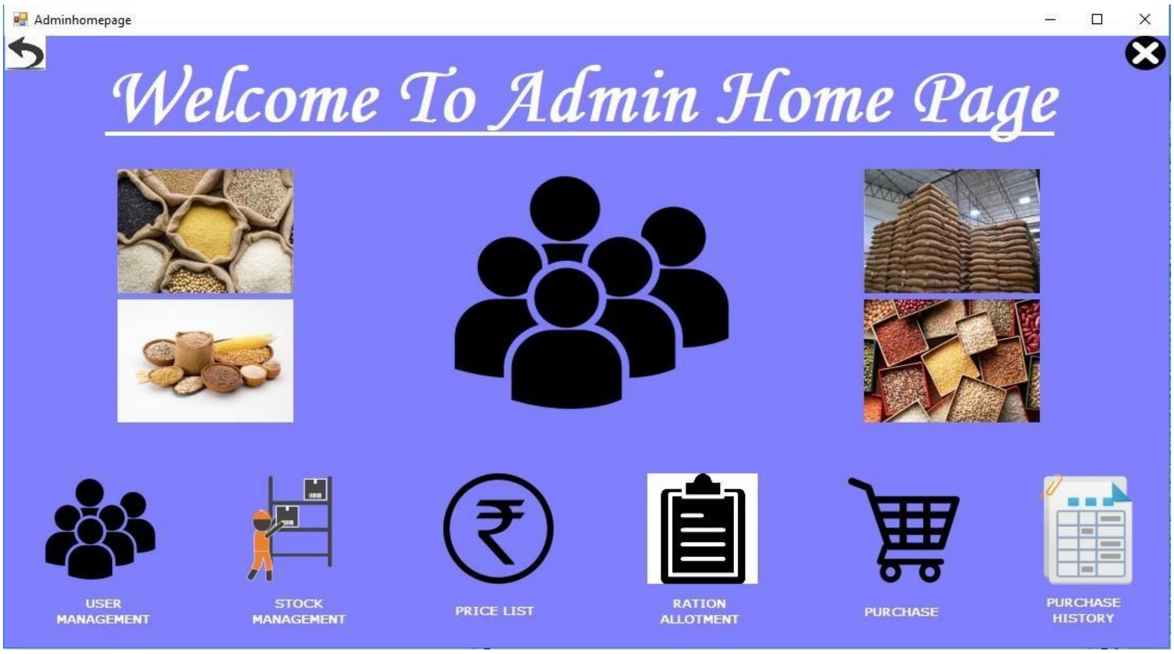
- ## User Login Form



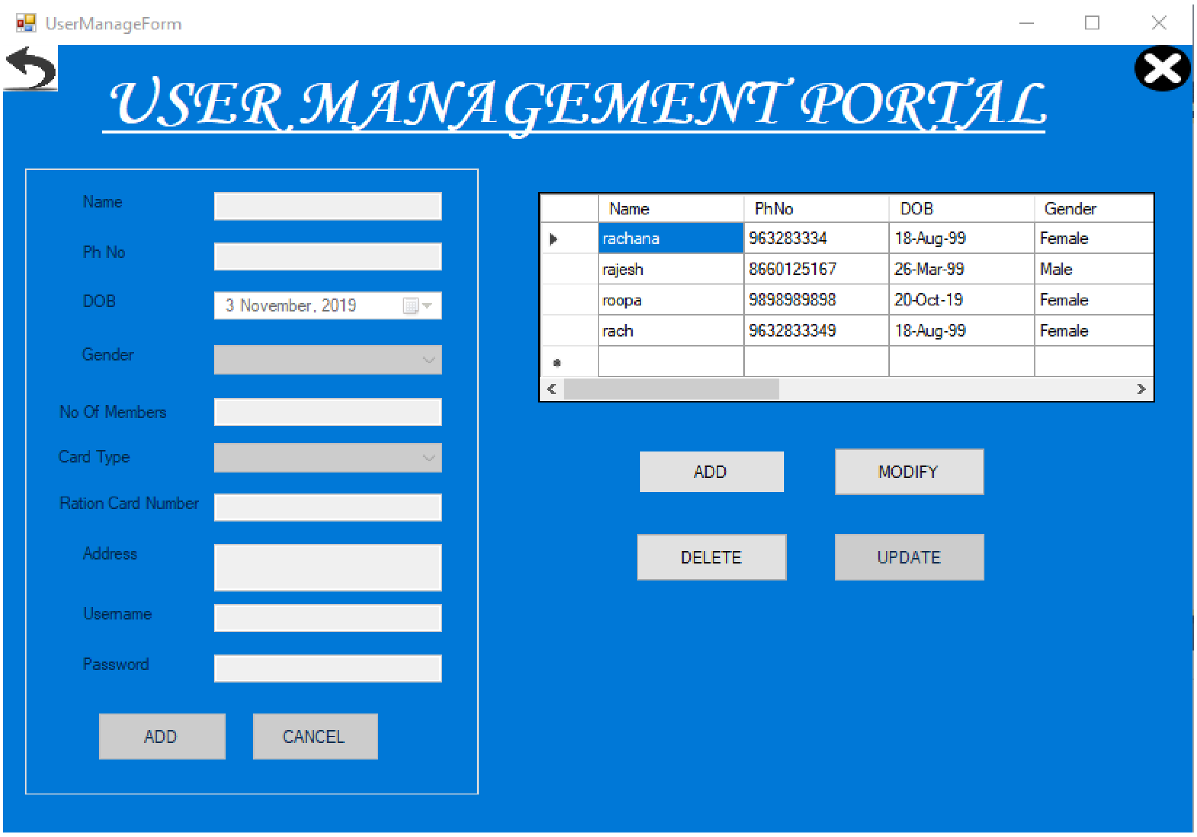
USERNAME abcde

PASSWORD •••••

- ## Admin Home Page



- ## User Management Form



USER MANAGEMENT PORTAL

Name

Ph No

DOB 3 November, 2019

Gender

No Of Members

Card Type

Ration Card Number

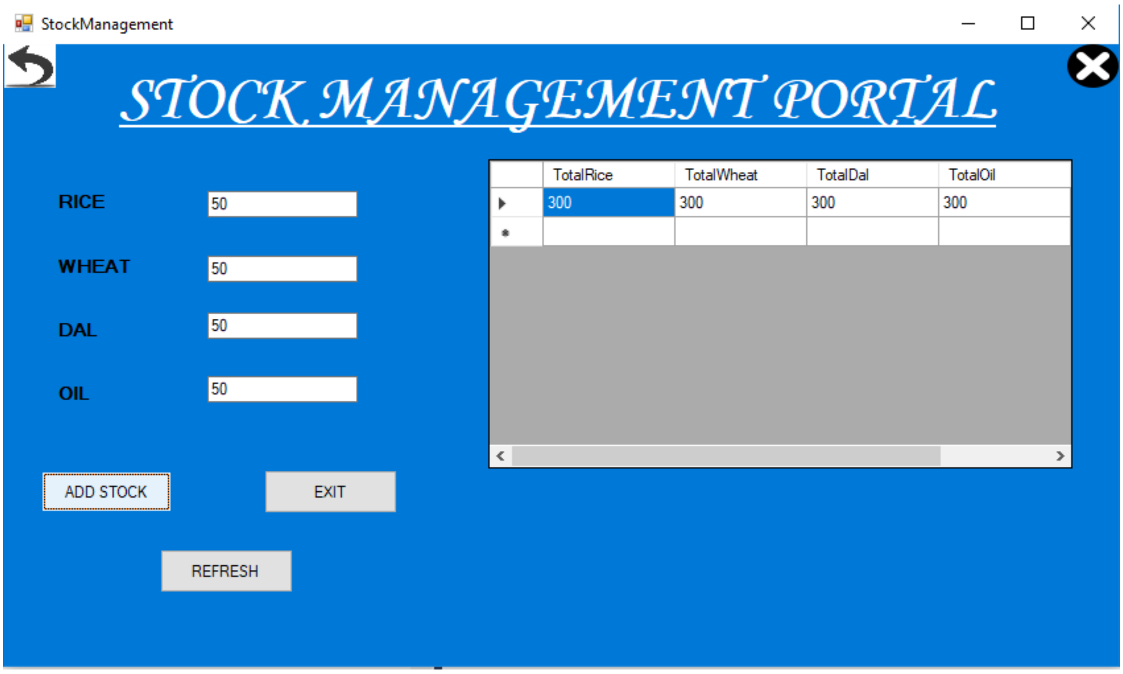
Address

Username

Password

| Name | PhNo | DOB | Gender |
|---|---|---|---|
| rachana | 963283334 | 18-Aug-99 | Female |
| rajesh | 8660125167 | 26-Mar-99 | Male |
| roopa | 9898989898 | 20-Oct-19 | Female |
| rach | 9632833349 | 18-Aug-99 | Female |

- ## Stock Management Form



STOCK MANAGEMENT PORTAL

RICE 50

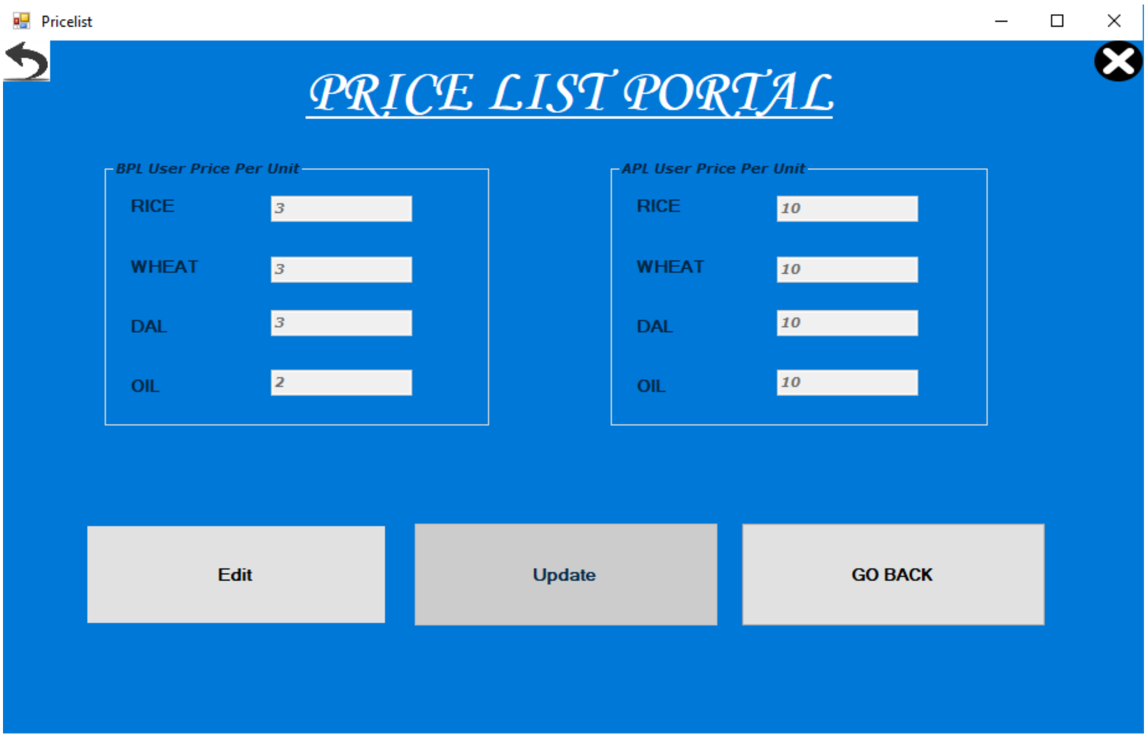
WHEAT 50

DAL 50

OIL 50

| TotalRice | TotalWheat | TotalDal | TotalOil |
|---|---|---|---|
| 300 | 300 | 300 | 300 |

- ## Price List Portal Form



Pricelist

PRICE LIST PORTAL

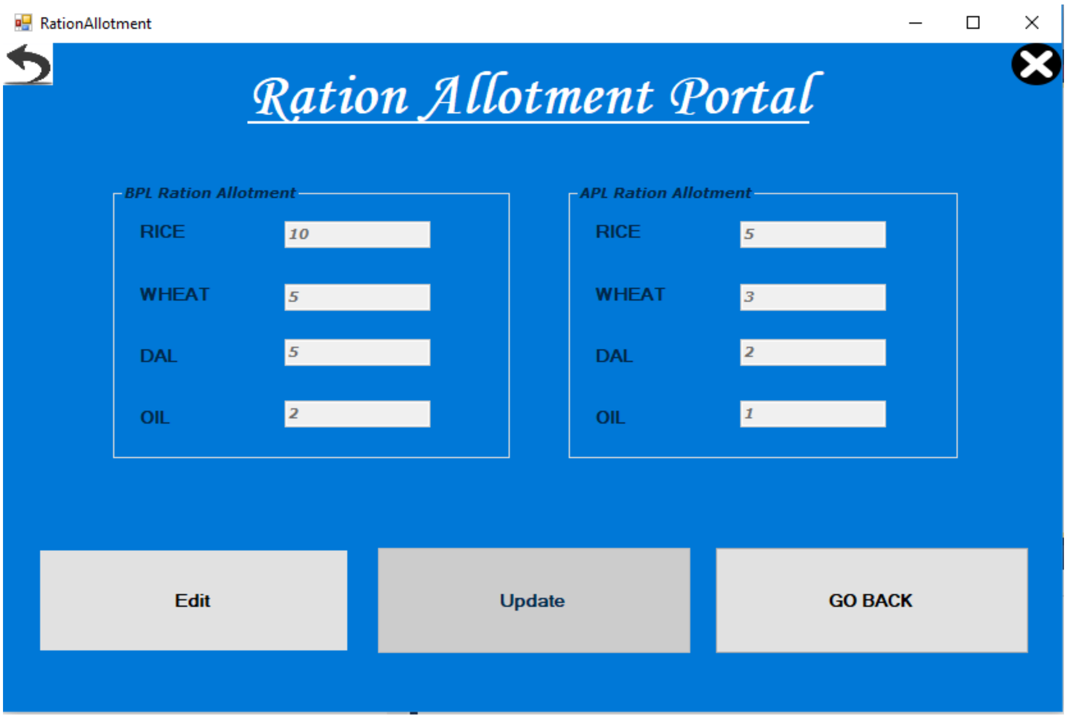
BPL User Price Per Unit

| | |
|---|---|
| RICE | 3 |
| WHEAT | 3 |
| DAL | 3 |
| OIL | 2 |

APL User Price Per Unit

| | |
|---|---|
| RICE | 10 |
| WHEAT | 10 |
| DAL | 10 |
| OIL | 10 |

- ## Ration Allotment Form



RationAllotment

Ration Allotment Portal

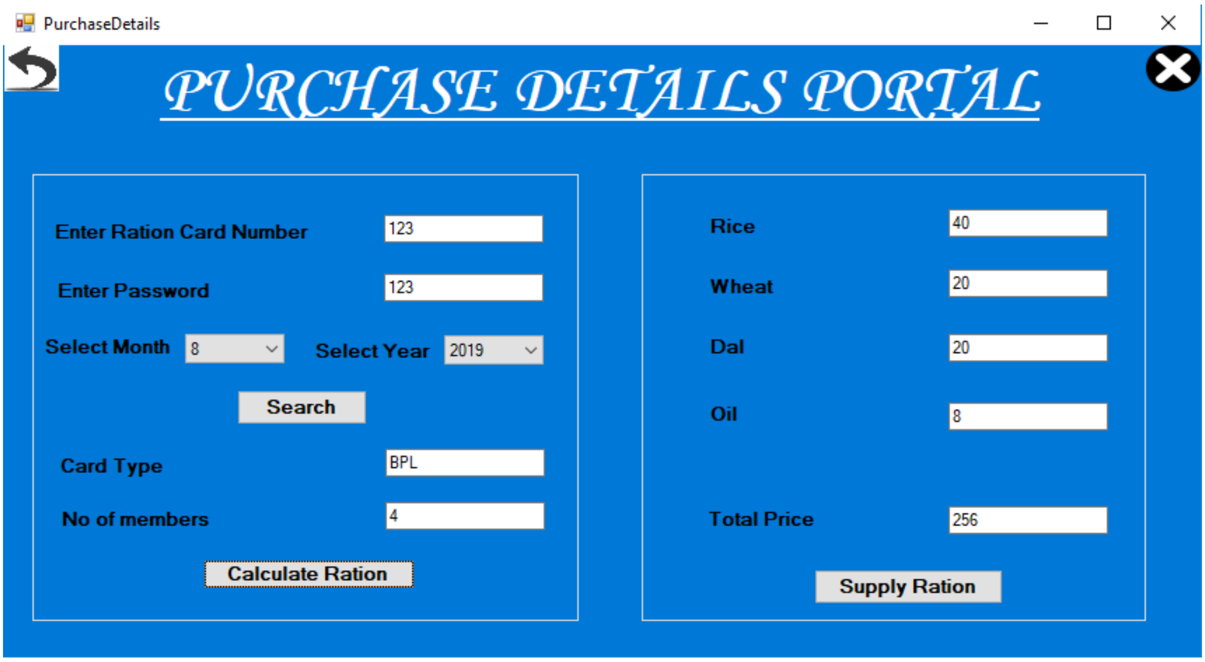
BPL Ration Allotment

| | |
|---|---|
| RICE | 10 |
| WHEAT | 5 |
| DAL | 5 |
| OIL | 2 |

APL Ration Allotment

| | |
|---|---|
| RICE | 5 |
| WHEAT | 3 |
| DAL | 2 |
| OIL | 1 |

- ## Purchase Details Form

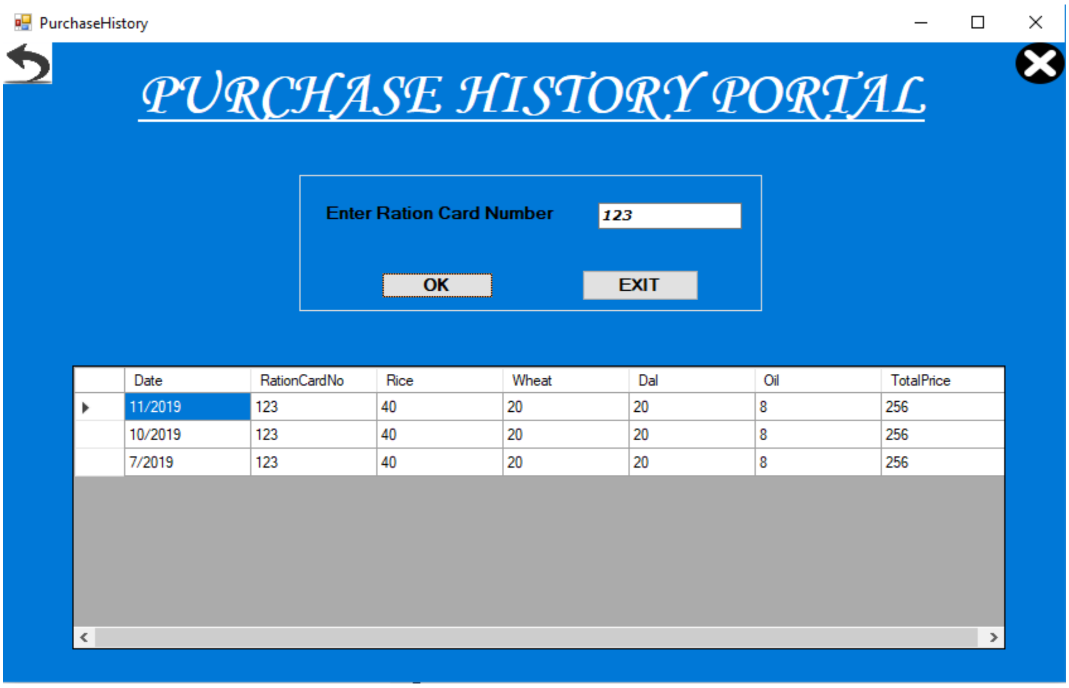


PurchaseDetails

PURCHASE DETAILS PORTAL

|  |  |
|---|---|
| Enter Ration Card Number | 123 |
| Enter Password | 123 |
| Select Month | 8 |
| Select Year | 2019 |
| Card Type | BPL |
| No of members | 4 |
| Rice | 40 |
| Wheat | 20 |
| Dal | 20 |
| Oil | 8 |
| Total Price | 256 |

- ## Purchase History Form



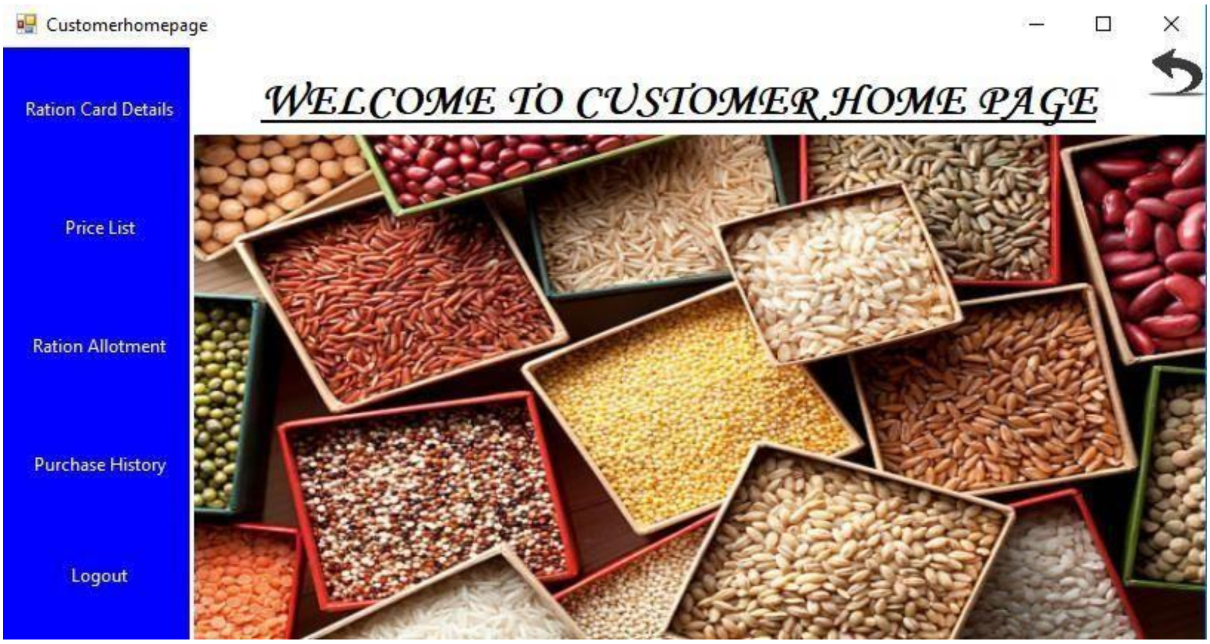
PurchaseHistory

PURCHASE HISTORY PORTAL

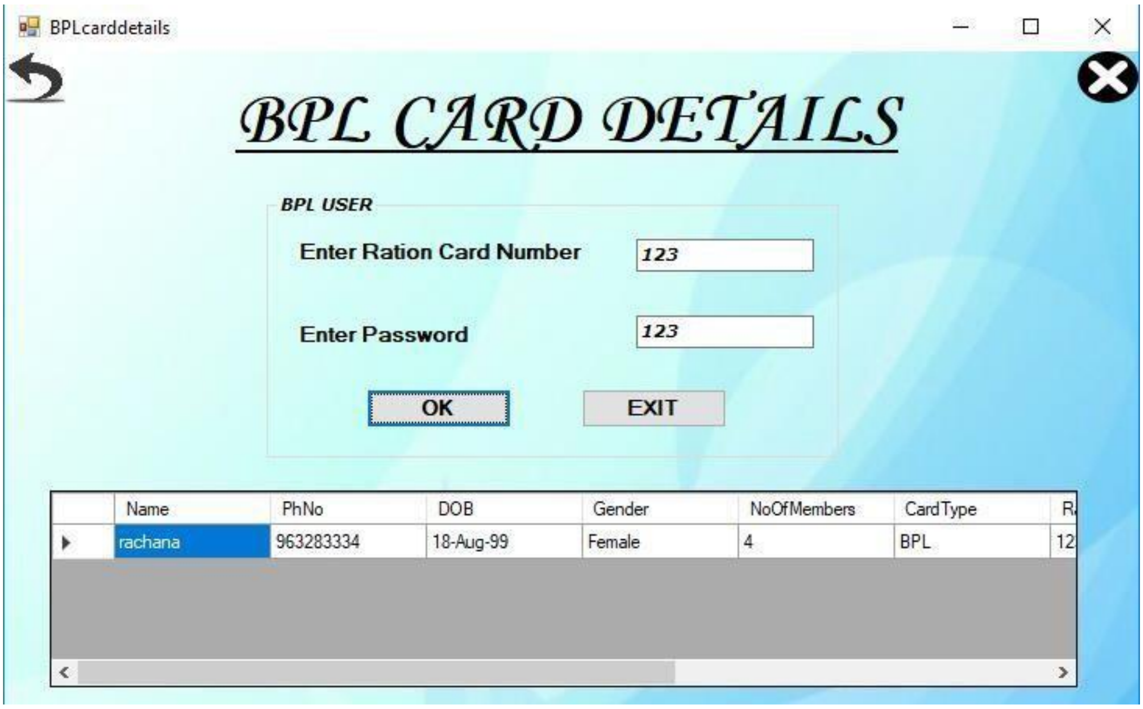
Enter Ration Card Number 123

| Date | RationCardNo | Rice | Wheat | Dal | Oil | TotalPrice |
|---|---|---|---|---|---|---|
| 11/2019 | 123 | 40 | 20 | 20 | 8 | 256 |
| 10/2019 | 123 | 40 | 20 | 20 | 8 | 256 |
| 7/2019 | 123 | 40 | 20 | 20 | 8 | 256 |

- **User Home Page**



- **BPL Card Details Form**

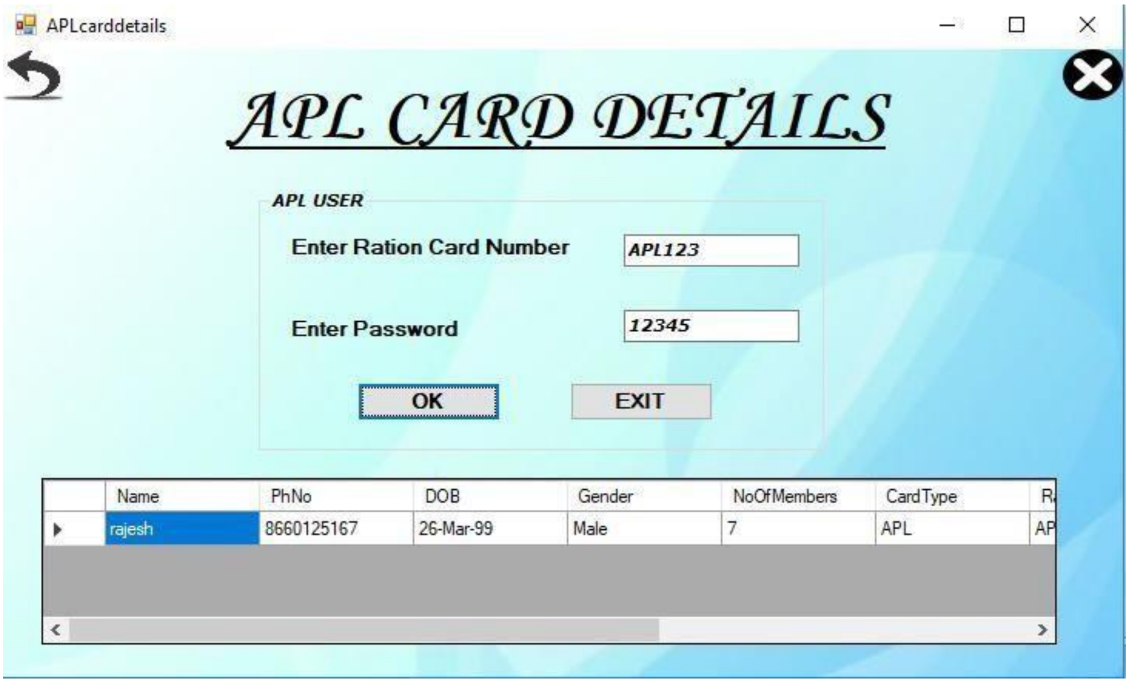


Enter Ration Card Number 123

Enter Password 123

| | Name | PhNo | DOB | Gender | NoOfMembers | CardType | R |
|---|---|---|---|---|---|---|---|
| ▸ | rachana | 963283334 | 18-Aug-99 | Female | 4 | BPL | 12 |

- **APL Card Details Form**

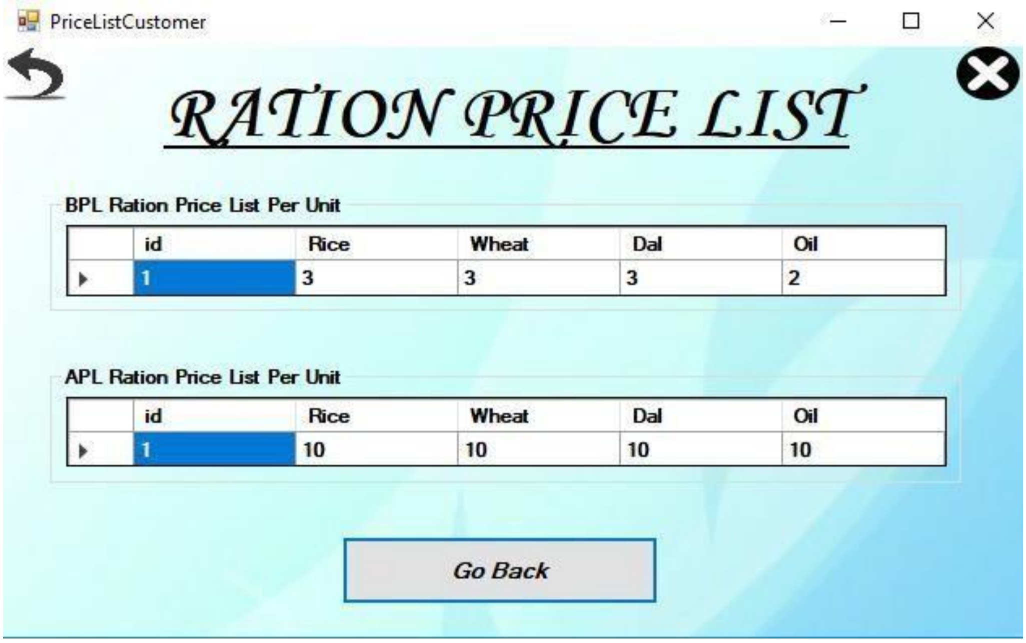


Enter Ration Card Number: APL123

Enter Password: 12345

| | Name | PhNo | DOB | Gender | NoOfMembers | CardType | R |
|---|---|---|---|---|---|---|---|
| ▸ | rajesh | 8660125167 | 26-Mar-99 | Male | 7 | APL | AP |

- **User Price List Form**



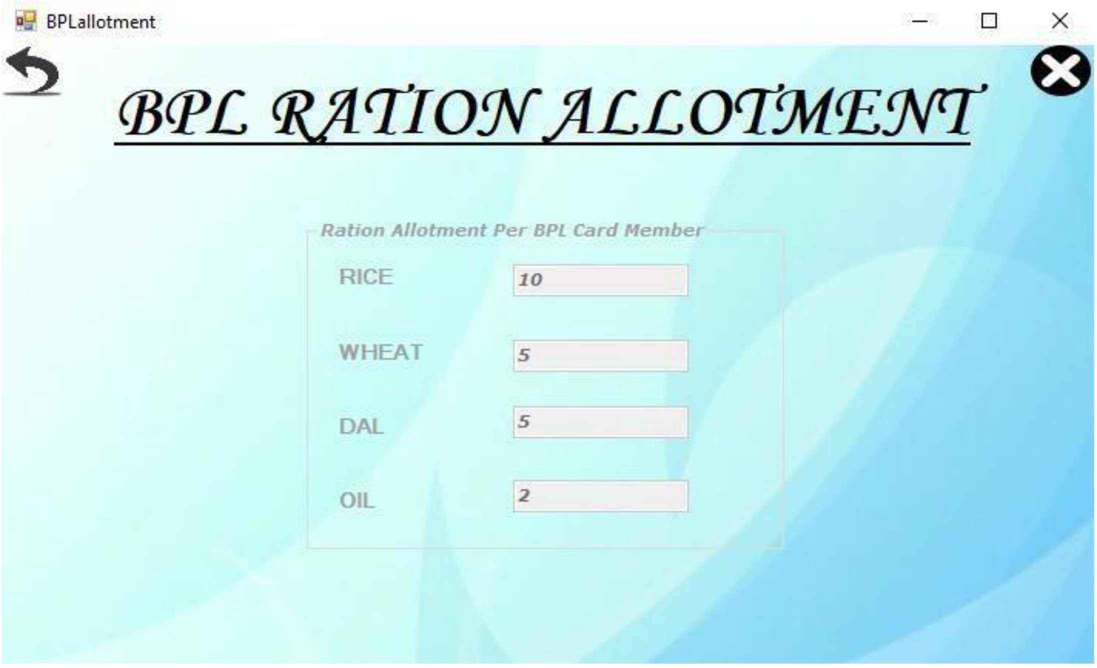
BPL Ration Price List Per Unit

| | id | Rice | Wheat | Dal | Oil |
|---|---|---|---|---|---|
| ▸ | 1 | 3 | 3 | 3 | 2 |

APL Ration Price List Per Unit

| | id | Rice | Wheat | Dal | Oil |
|---|---|---|---|---|---|
| ▸ | 1 | 10 | 10 | 10 | 10 |

- **BPL Ration Allotment Form**

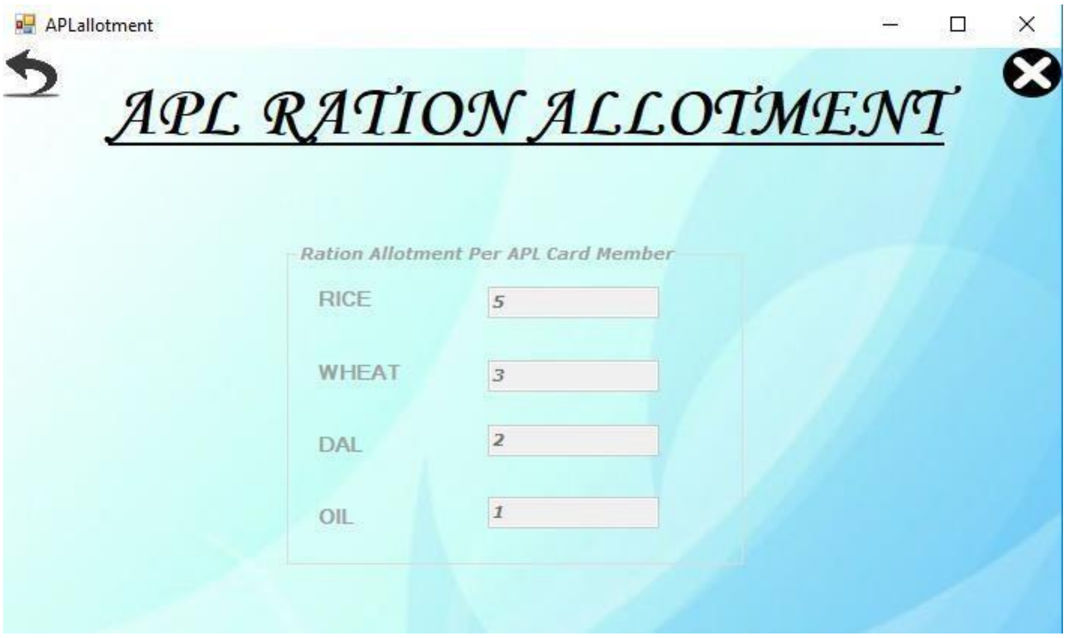


BPL RATION ALLOTMENT

Ration Allotment Per BPL Card Member

| | |
|---|---|
| RICE | 10 |
| WHEAT | 5 |
| DAL | 5 |
| OIL | 2 |

- **APL Ration Allotment Form**

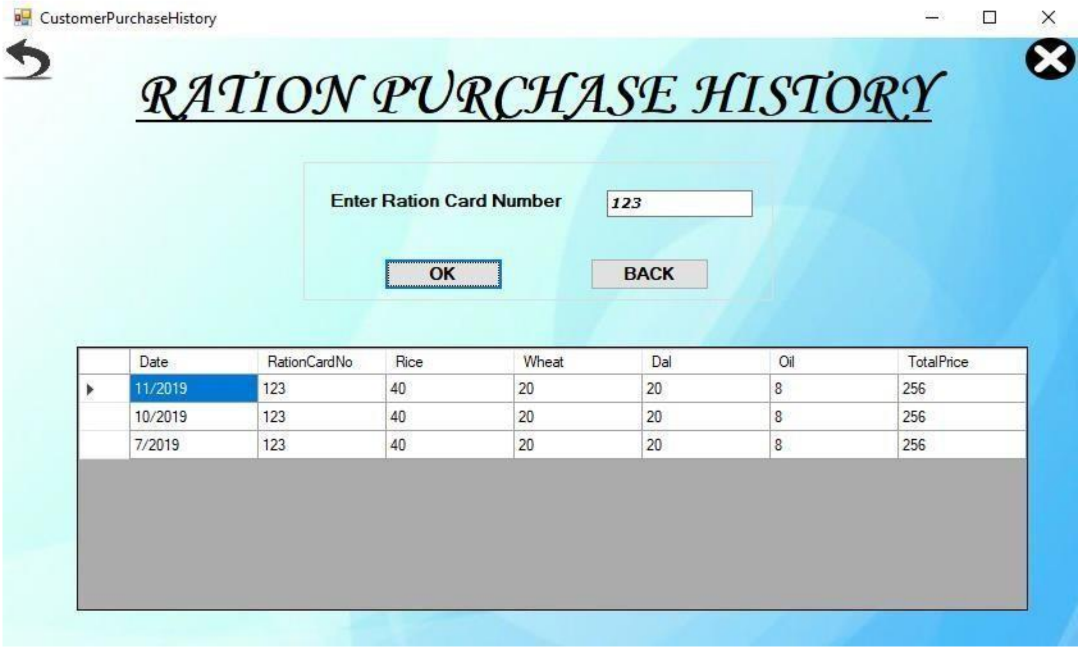


APL RATION ALLOTMENT

Ration Allotment Per APL Card Member

| | |
|---|---|
| RICE | 5 |
| WHEAT | 3 |
| DAL | 2 |
| OIL | 1 |

- **User Purchase History Form**



| Date | RationCardNo | Rice | Wheat | Dal | Oil | TotalPrice |
|---|---|---|---|---|---|---|
| 11/2019 | 123 | 40 | 20 | 20 | 8 | 256 |
| 10/2019 | 123 | 40 | 20 | 20 | 8 | 256 |
| 7/2019 | 123 | 40 | 20 | 20 | 8 | 256 |

# 8. FUTURE ENHANCEMENTS

The software application is developed in such a way that application can be enhanced with further requirements without any much change in the system flow. The additional modules can be implemented to the same application without disturbing the existing modules.

A number of enhancements can be added to the existing software application like,

- Security enhancements can be made by implementing face recognition login, OTP (one-time password) and fingerprint login techniques.
- Mobile applications will be developed so that the users can gain access to our services from their mobile phones.
- Information will be made available in different languages so that language would not be a boundary for learning.
- Online version of the application will be developed which helps the users to browse required information such as ration card details, price list details, ration allotment details, purchase history and other related information and services from their home itself without visiting any E-Ration centers.

# 9. CONCLUSION

E-Ration Management Application has been designed and developed according to the current requirements of the users. The benefit expected from this application is that it could reduce the burden of traditional methods and overcomes all the drawbacks of the existing system.

This application reduces the work load of the administrator by implementing digital methodologies and techniques. Administrator need not have to calculate the total stock and the total amount to be paid by the users manually as our application does these calculations automatically which reduces time and are error free.

The major goal of the software is to put an end to the black marketing and corruption of ration going on in various cities. Users will now be able to check the price list of the ration and the quantity of ration allotted to them by the government and this will put an end to the practice of holding back rations and ensures that they reach the needy.

# 10. BIBLIOGRAPHY

## Reference Books

- **Database Management System:** Ashwini S Diwakar, Skywardpublishers.
- **Visual programming:** Dr M.A Rama, Subhas Stores & publishers.

## Web Sites

- **www.code.msdn.microsoft.com**
  MSDN Code Gallery is your destination for downloading sample applications and code snippets, as well as sharing your own resource
- **www.tutorialspoint.com**
  A free online learning platform for learning programming languages.
- **www.vbtutor.com**
  A free online learning platform for visual programming.
- **www.stackoverflow.com**
  A free online help portal for software developers.

## YouTube Channels

- **ProgrammingKnowledge**
  https://www.youtube.com/user/ProgrammingKnowledge