

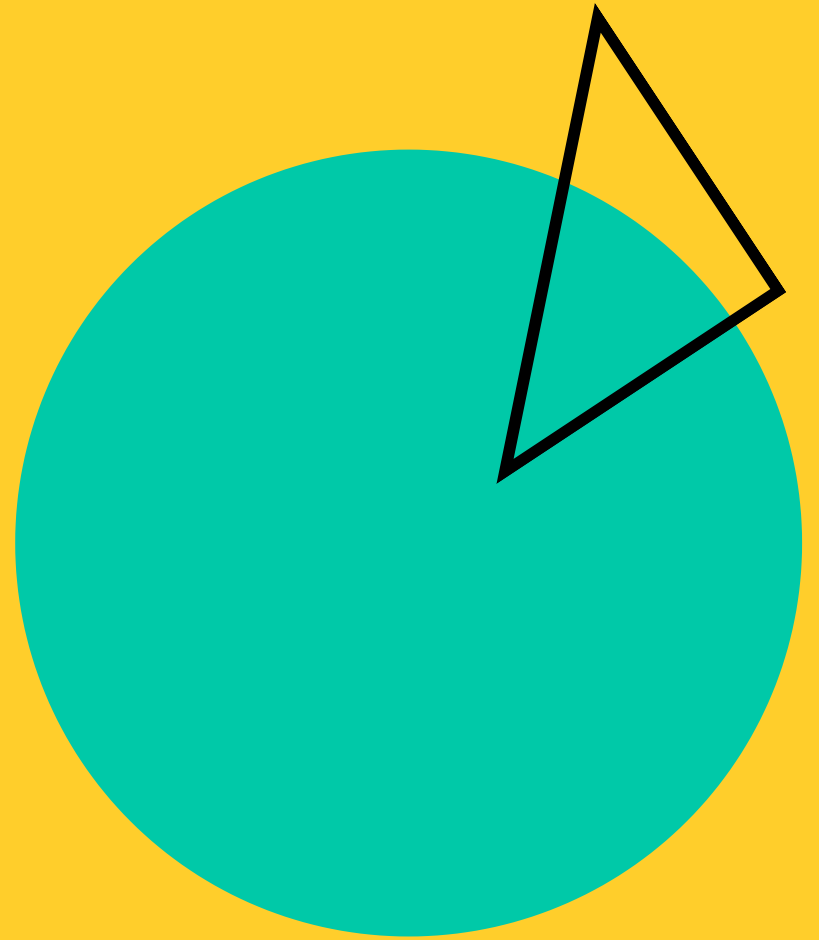
INVENTORY MANAGEMENT SYSTEM(IMS)

By:

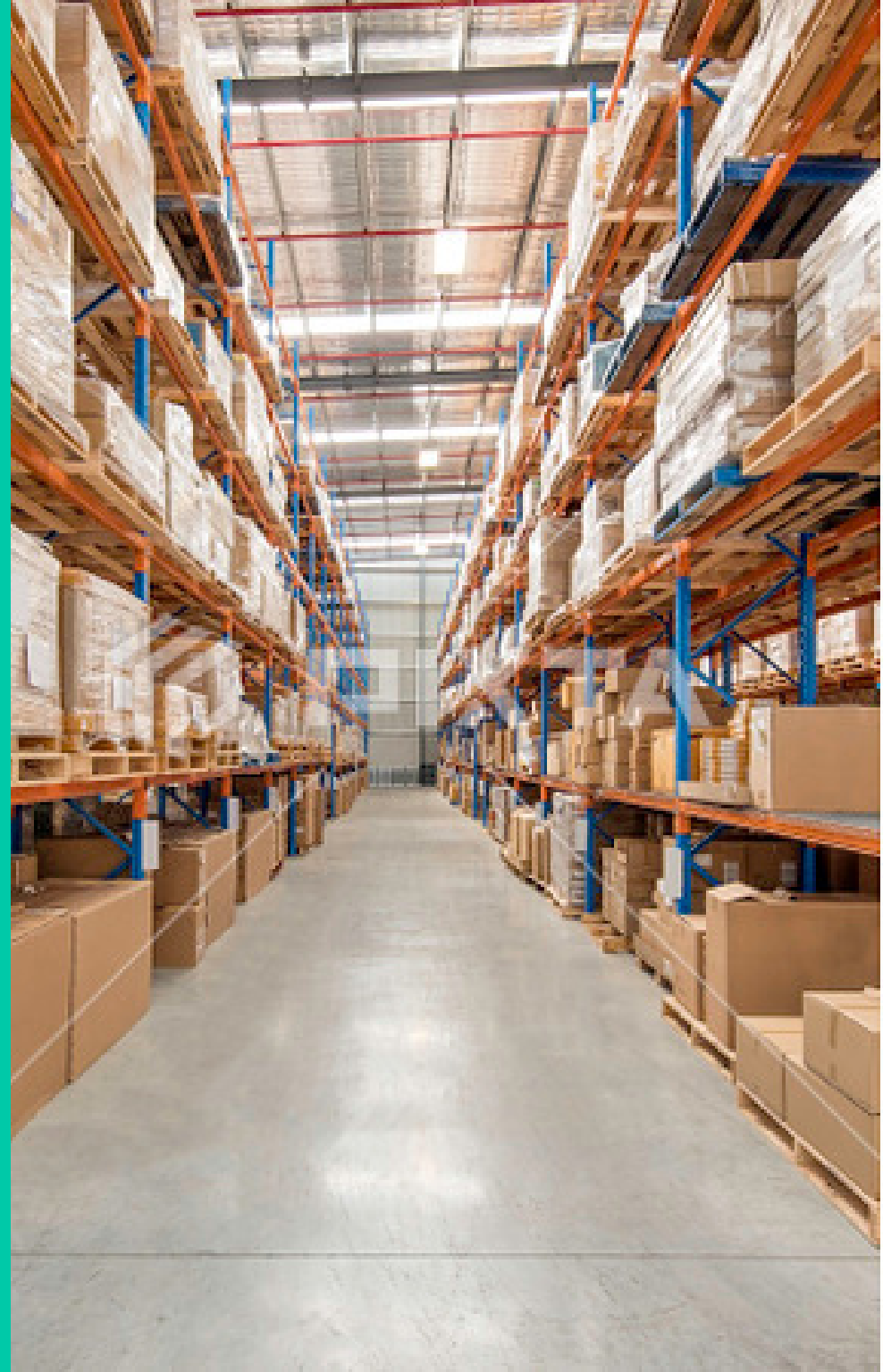
Ilnaz Alizadeh

Hossein Parsa

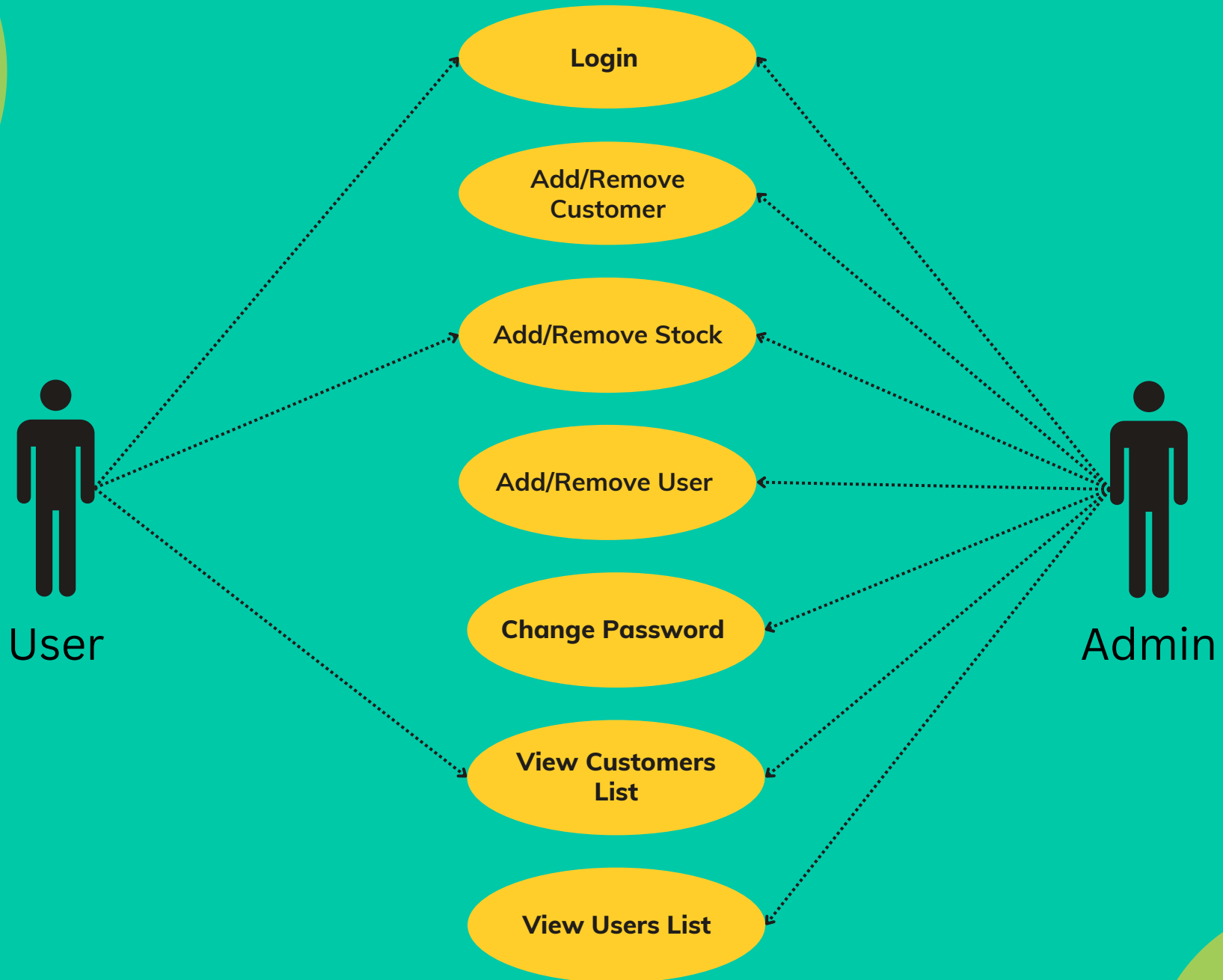
Amir Sharifi



- 
- **What is IMS?**
 - **Types of IMS.**
 - **IMS applications.**

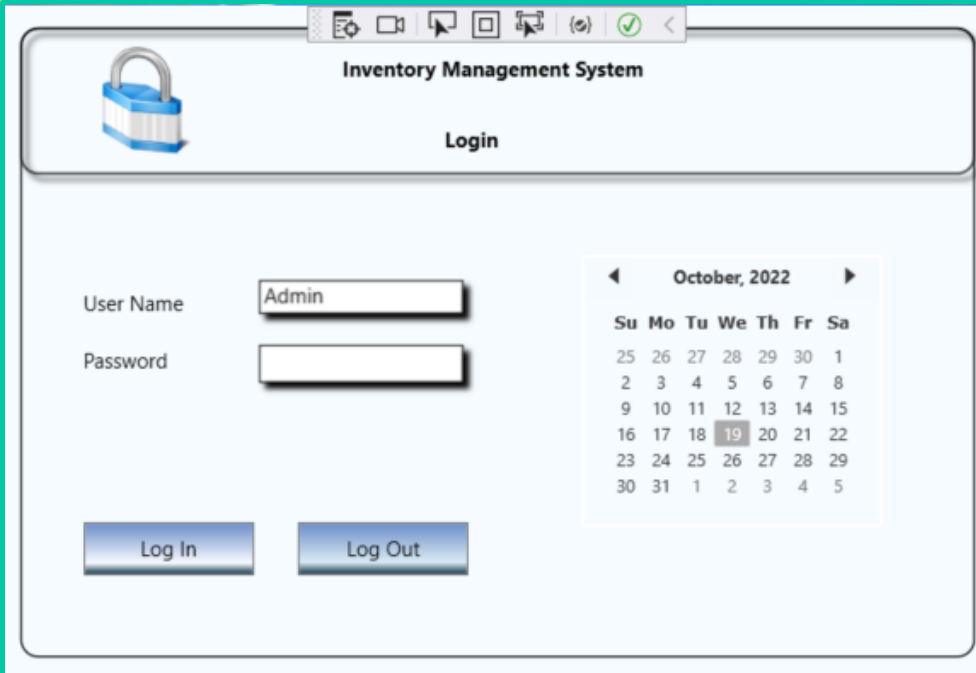


Use-case Diagram



Our Solutions

Login Window



The screenshot shows the login window of the Inventory Management System. It features a title bar with standard window controls. Below the title bar, there is a header section with a lock icon and the text "Inventory Management System" and "Login". The main area contains two input fields: "User Name" with the value "Admin" and "Password" which is empty. To the right of these fields is a calendar widget for "October, 2022". At the bottom, there are two buttons: "Log In" and "Log Out".

Inventory Management System

Login

User Name: Admin

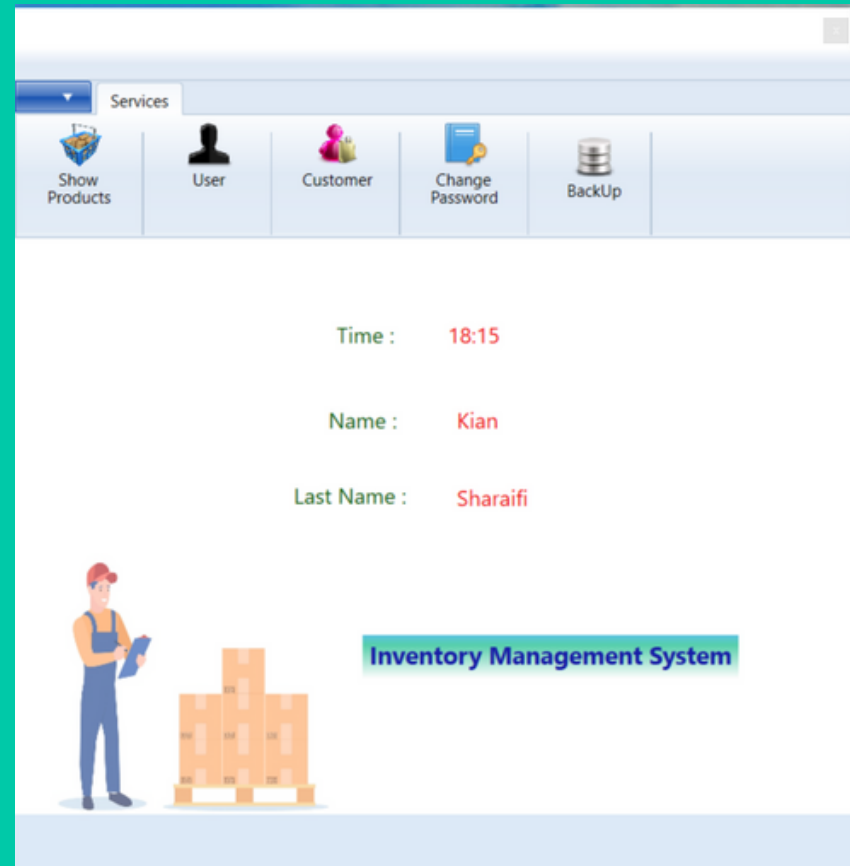
Password:

October, 2022

Su	Mo	Tu	We	Th	Fr	Sa
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

Log In Log Out

Main Window



The screenshot shows the main window of the Inventory Management System. It features a title bar with a "Services" dropdown menu. Below the title bar, there is a navigation bar with five icons: "Show Products", "User", "Customer", "Change Password", and "BackUp". The main area displays the following information:

- Time : 18:15
- Name : Kian
- Last Name : Sharaifi

At the bottom, there is an illustration of a worker in overalls and a red cap standing next to a stack of orange boxes. To the right of the illustration, there is a label "Inventory Management System".

Services

Show Products User Customer Change Password BackUp

Time : 18:15



Name : Kian










Last Name : Sharaifi

Inventory Management System

Our Solutions


Products list view

**Products**

Product ID	Name	Description	Price	Quantity	ProductActive	Registered By	Image
8	Canon Camera	LensLess	1100	8	1	Kian Sharaifi	
	 Price History						
	 Product Quantity						
	 Active Product						
		Design	90	20	1	Kian Sharaifi	
10	Chocolate	Sweet	25	30	1	Kian Sharaifi	
11	bear	very soft	15	3	1	Kian Sharaifi	
12	Bike	Some Desc	120	0	1	Amir Sharifi	
13	chair	good	10	30	1	Admin Parsa	

Add ProductEdit ProductDelete Product

Users list view

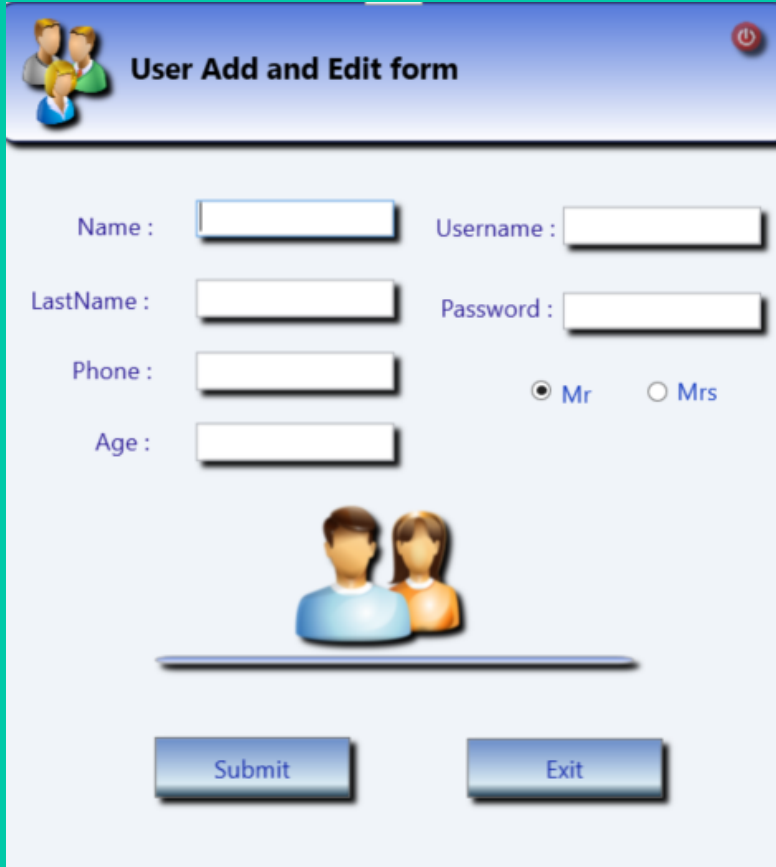
**Users**

User ID	Name	Last Name	Tel	Age	Status
1	ilnaz	Alizadeh	4388332414	36	1
2	Amir	Sharifi	514676324	35	1
4	Admin	Parsa	1234567896	34	1
5	Nick	Pa	5149731552	35	2
6	Kian	Sharaifi	5146663333	6	2
7		sa	5149633222	35	1
8		arifi	51499999	33	1
9	elnaz	Alizadeh	4388332414	38	1

Add UserEdit UserDelete User

Our Solutions

Add/Edit Users




User Add and Edit form

Name : Username :

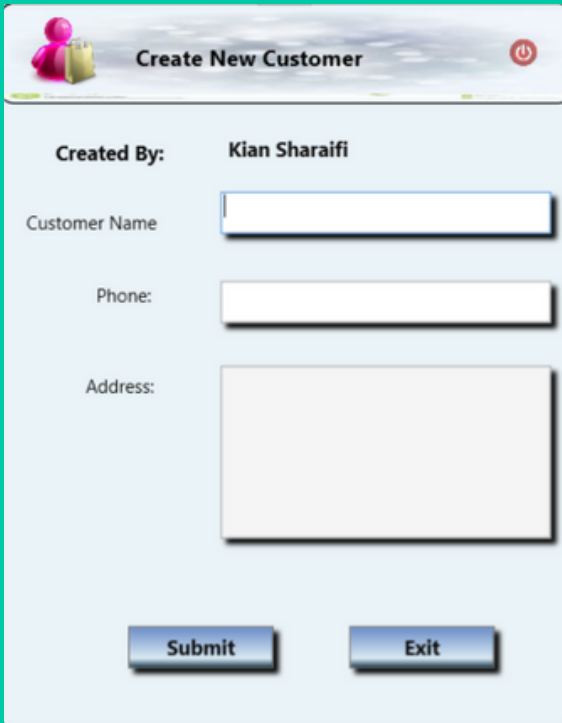
LastName : Password :

Phone : ☒ Mr ☐ Mrs

Age :



Add/Edit Customers



Create New Customer

Created By: Kian Sharaifi

Customer Name :

Phone:

Address:

Add/Edit Products



Add and Edit Product

Created By: Kian Sharaifi

Product Name :

Description :

Image : 


Our Solutions

Add new operation

Products list view



A dialog box titled "Change Password" with a key icon and a power button. It contains fields for Username (Kian Sharaifi), Old Password, and New Password. At the bottom are "Submit" and "Exit" buttons.



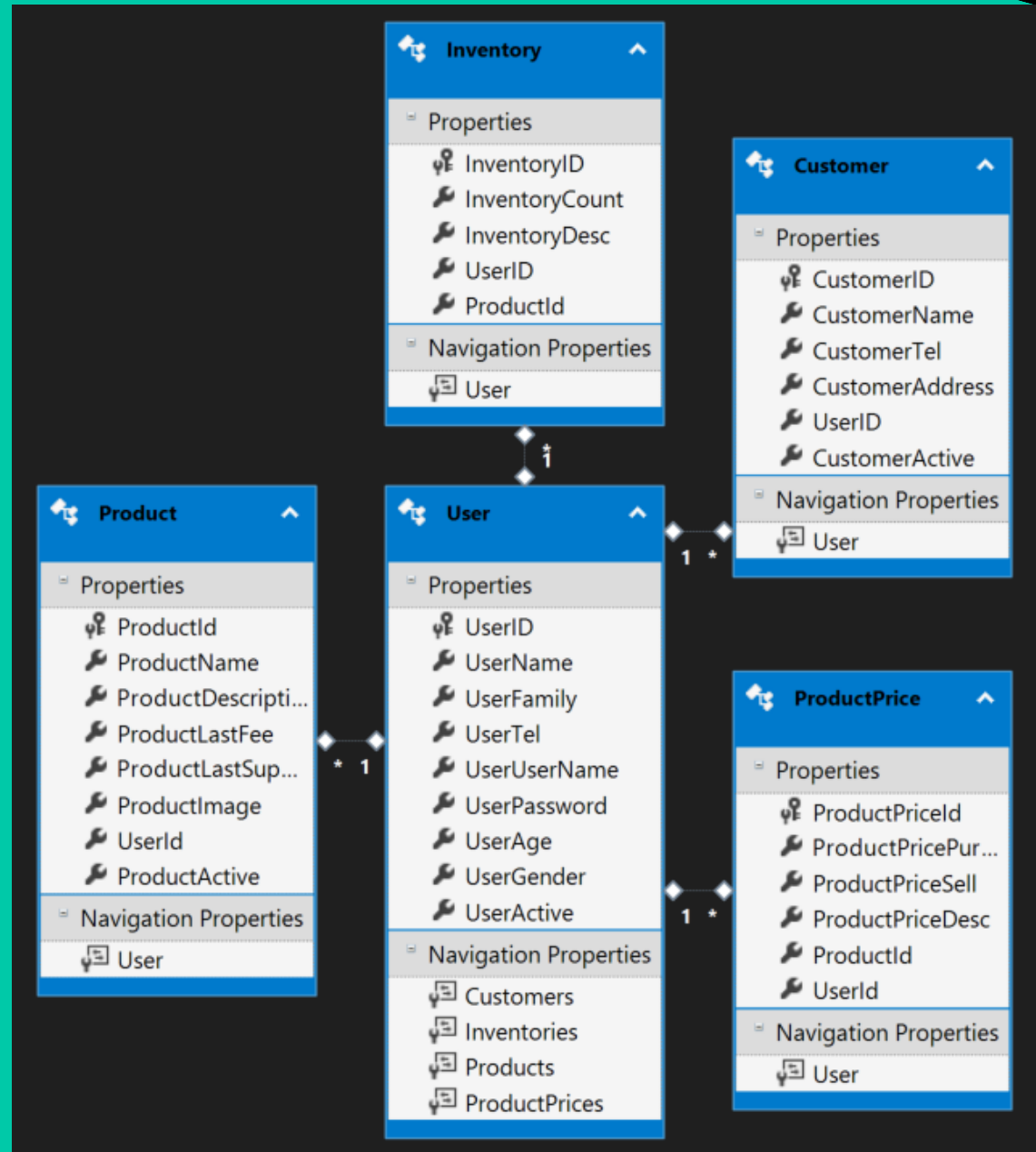
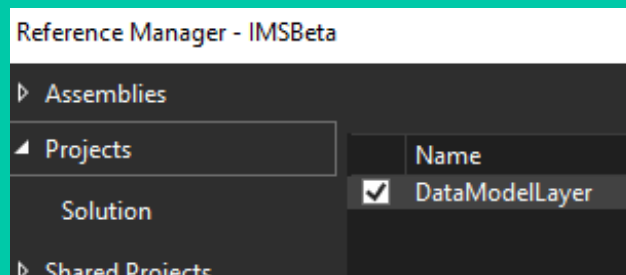
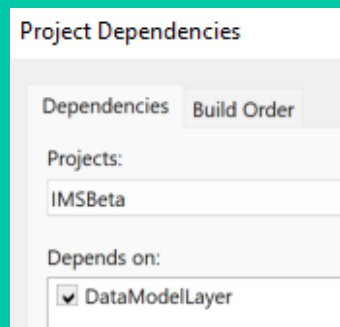
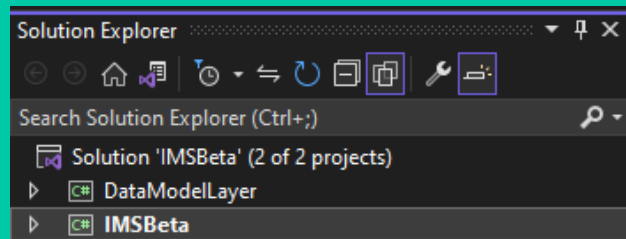
A dialog box titled "Add New Operation" with a 3D cube icon and a power button. It displays "Canon Camera" and "Created By : Kian Sharaifi". It has fields for Number, Transaction Type (with a dropdown menu showing "Add to inventory" and "Remove from inventory"), and Description. At the bottom are "Add" and "Exit" buttons.

UserID	UserName	UserFamily	UserTel	UserUserName	UserPassword
1	linaz	Alizadeh	4388332414	Ila	03-AC-67-42-16-F3-E1-5C-76-1E-E1-A5-E2-55-F0-67-...
2	Amir	Sharifi	514676324	ASH	03-AC-67-42-16-F3-E1-5C-76-1E-E1-A5-E2-55-F0-67-...
4	Admin	Parsa	1234567896	Admin	03-AC-67-42-16-F3-E1-5C-76-1E-E1-A5-E2-55-F0-67-...
5	Nick	Pa	5149731552	Nick	A6-65-A4-59-20-42-2F-9D-41-7E-48-67-EF-DC-4F-B8-...
6	Kian	Sharaifi	5146663333	Kian12	03-AC-67-42-16-F3-E1-5C-76-1E-E1-A5-E2-55-F0-67-...

Database

Tables

Two separate Layer



Database

Views

Vw_Customer	
Properties	
UserName	
UserFamily	
CustomerID	
CustomerName	
CustomerTel	
CustomerAddress	
UserID	
FullName	
CustomerActive	
Navigation Properties	

Vw_Product	
Properties	
ProductId	
ProductName	
ProductDescripti...	
ProductLastFee	
ProductLastSup...	
ProductImage	
UserId	
UserFamily	
UserName	
FullName	
ProductActive	
Navigation Properties	

Vw_Users	
Properties	
UserID	
UserName	
UserFamily	
UserTel	
UserUserName	
UserAge	
UserPassword	
UserGender	
UserActive	
UserGenderShow	
UserActiveShow	
Navigation Properties	

Vw_Inventory	
Properties	
InventoryID	
InventoryCount	
InventoryDesc	
UserID	
ProductId	
ProductName	
UserName	
UserFamily	
FullName	
status	
Navigation Properties	

Vw_ProductPr...	
Properties	
ProductPriceId	
ProductPricePur...	
ProductPriceSell	
ProductPriceDesc	
ProductId	
UserId	
ProductName	
UserName	
UserFamily	
FullName	
Navigation Properties	

database_fire...	
Properties	
id	
name	
start_ip_address	
end_ip_address	
create_date	
modify_date	
Navigation Properties	

Technology used



What we learned

How to Save image in DB

```
if (imageName != null)
{
    //// change type of image to String
    FileStream fs = new FileStream(imageName, FileMode.Open, FileAccess.Read);
    byte[] imgByteArr = new byte[fs.Length];
    fs.Read(imgByteArr, 0, Convert.ToInt32(fs.Length));
    fs.Close();

    ////////////Save in DB
    database.Sp_ins_product(txt_productname.Text.Trim(), txt_desc.Text.Trim(), imgByteArr, PublicVariable.gUserId);
    database.SaveChanges();
    MessageBox.Show("Information saved successfully", "Save error", MessageBoxButtons.OK, MessageBoxIcon.Information);
}
else
{
    MessageBox.Show("Please select an image for the new item", "Save error", MessageBoxButtons.OK, MessageBoxIcon.Error);
    return;
}
```

What we learned

Hash the password

```
//////////Hashing Password//////////  
SHA256CryptoServiceProvider Sha2 = new SHA256CryptoServiceProvider();  
Byte[] B1;  
Byte[] B2;  
B1 = UTF8Encoding.UTF8.GetBytes(txt_pass.Password.Trim());  
B2 = Sha2.ComputeHash(B1);  
  
string UserPasswordHashed = BitConverter.ToString(B2);
```

UserID	UserName	UserFamily	UserTel	UserUserName	UserPassword
1	ilinaz	Alizadeh	4388332414	Ila	03-AC-67-42-16-F3-E1-5C-76-1E-E1-A5-E2-55-F0-67-...
2	Amir	Sharifi	514676324	ASH	03-AC-67-42-16-F3-E1-5C-76-1E-E1-A5-E2-55-F0-67-...
4	Admin	Parsa	1234567896	Admin	03-AC-67-42-16-F3-E1-5C-76-1E-E1-A5-E2-55-F0-67-...
5	Nick	Pa	5149731552	Nick	A6-65-A4-59-20-42-2F-9D-41-7E-48-67-EF-DC-4F-B8-...
6	Kian	Sharaifi	5146663333	Kian12	03-AC-67-42-16-F3-E1-5C-76-1E-E1-A5-E2-55-F0-67-...

What we learned

Inactive instead of remove

1 reference

```
private void Btn_EditProduct_Click(object sender, RoutedEventArgs e)
{
    object item = DataGridProduct.SelectedItem;
    win_add_edit_product w_p = new win_add_edit_product();
    w_p.win_Type = 2;////Editing Mode

    w_p.ProductId = Convert.ToInt32((DataGridProduct.SelectedCells[0].Column.GetCellContent(item) as TextBlock).Text);
    w_p.ProductName = (DataGridProduct.SelectedCells[1].Column.GetCellContent(item) as TextBlock).Text;
    w_p.ProductDesc = (DataGridProduct.SelectedCells[2].Column.GetCellContent(item) as TextBlock).Text;

    w_p.ShowDialog();
    ShowProductInfo();
}
```

What we learned

Procedure & Transaction

```
1 reference
private void btn_ok_Click(object sender, RoutedEventArgs e)
{
    using (TransactionScope ts = new TransactionScope())
    {
        try
        {
            ////////////////////////////////////////Add New Price
            ProductPrice PP = new ProductPrice();
            PP.ProductPricePurch = Convert.ToInt64(txt_purch.Text.Trim());
            PP.ProductPriceSell = Convert.ToInt64(txt_sale.Text.Trim());
            PP.ProductPriceDesc = txt_desc.Text.Trim();
            PP.ProductId = this.productid;
            PP.UserId = PublicVariable.gUserId;

            database.ProductPrices.Add(PP);
            database.SaveChanges();
            //////////////////////////////////////// Update Price
            database.Sp_UpdateFee_Product(this.productid, Convert.ToInt64(txt_sale.Text.Trim()));
            database.SaveChanges();

            ts.Complete();
            MessageBox.Show("Information saved successfully", "Save error", MessageBoxButton.OK, MessageBoxImage.Information);
        }
    }
}
```

```
Create Procedure Sp_UpdateFee_Product
    @ProductId int,
    @ProductLastFee bigint
    As
Update Product
    Set ProductLastFee = @ProductLastFee
    Where ProductID = @ProductId
```

Future Work

User access Tree

Connect to sales

Print/save reports

Charts

Summary

- what was done

Did research and made decision about technologies

Planned the work flow and divided the tasks

Examined different solutions for each task

- what is the result

Got one step closer to understand how
a complete application works

Got better in doing research and solution finding