**Super Market Management System**



Session: 2021 – 2024

**Submitted by:**

M Yaqoob 2021-CS-118

**Supervised by:**

Sir Awais

Department of Computer Science

**University of Engineering and Technology**

**Lahore Pakistan**

**Description:**

Super Market Management has three users:

1- Manager

2- Staff

3- Customer

Manager can do following things:

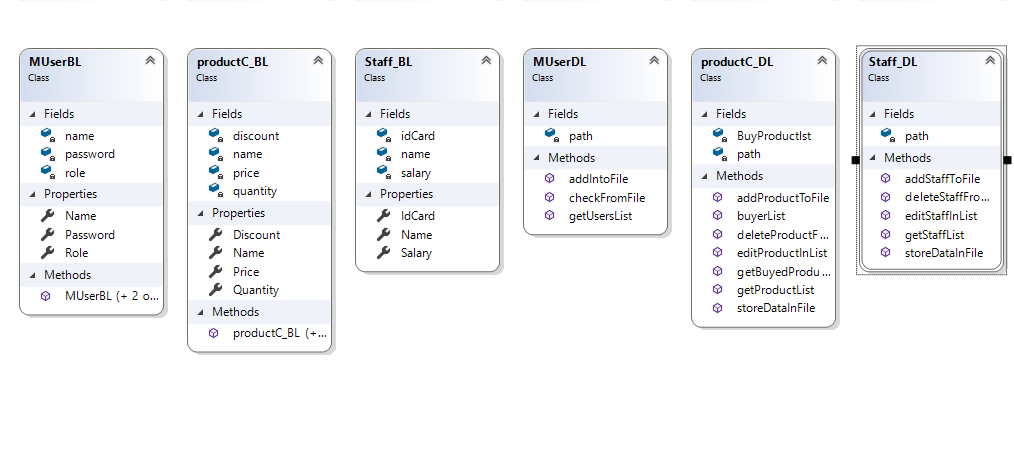
* can add product detail
* can edit product detail
* can delete product detail
* can view product list
* can add Staff member info
* can edit staff member info
* can delete staff member info
* can send messages to staff members
* can pay salary to staff members
* can edit the quantity of product
* can view short listed products
* can edit and delete short listed products

Staff member can do following things:

* he can check messages
* he can get info about salary
* he can send request to manager

Customer can do following things:

* he can buy products
* he can set quantity to buked products
* he can calculate bill

**Class Diagram:** 

**1-Register Form:**

namespace bilingSystemGUI

{

public partial class frmlogin : Form

{

public frmlogin()

{

InitializeComponent();

}

private void btnRegister\_Click(object sender, EventArgs e)

{

//SignIN

this.Hide();

Form f = new frmRegisterNewUser();

f.Show();

}

private void btnSignUP\_Click(object sender, EventArgs e)

{

//SignUP

this.Hide();

Form f = new frmSignUp();

f.Show();

}

}

}

**2-Add New Product Form:**

namespace bilingSystemGUI

{

public partial class frmAddNewProduct : Form

{

public frmAddNewProduct()

{

InitializeComponent();

}

private void btnBack\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmManager();

f.Show();

}

private void btnAdd\_Click(object sender, EventArgs e)

{

try

{

if ((float.Parse(txtPrice.Text) >= 0) && (float.Parse(txtDiscount.Text) >= 0) && (float.Parse(txtQuantity.Text) >= 0) && (txtName.Text) != string.Empty)

{

productC\_BL product = new productC\_BL();

product.Name = txtName.Text;

product.Price = float.Parse(txtPrice.Text);

product.Discount = float.Parse(txtDiscount.Text);

product.Quantity = float.Parse(txtQuantity.Text);

productC\_DL.addProductToFile(product);

MessageBox.Show("Product Info is added Successfully");

}

else

{

throw new Exception("Enter Valid Information");

}

}

catch (Exception exp)

{

MessageBox.Show(exp.Message);

}

}

}

**3- Add New Staff Member Form:**

namespace bilingSystemGUI

{

public partial class frmAddNewStaffMember : Form

{

public frmAddNewStaffMember()

{

InitializeComponent();

}

private void btnBack\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmManager();

f.Show();

}

private void btnAdd\_Click(object sender, EventArgs e)

{

try

{

if ((float.Parse(txtIDCard.Text) >= 0) && (float.Parse(txtSalary.Text) >= 0) && (txtName.Text) != string.Empty)

{

Staff\_BL member = new Staff\_BL();

member.Name = txtName.Text;

member.IdCard = txtIDCard.Text;

member.Salary = float.Parse(txtSalary.Text);

Staff\_DL.addStaffToFile(member);

MessageBox.Show("Staff Member Info is added Successfully");

}

else

{

throw new Exception("Enter Valid Information");

}

}

catch (Exception exp)

{

MessageBox.Show(exp.Message);

}

}

}

}

**3-Customer Form:**

namespace bilingSystemGUI

{

public partial class frmCustumer : Form

{

public frmCustumer()

{

InitializeComponent();

}

private void btnExit\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmSignUp();

f.Show();

}

private void frmCustumer\_Load(object sender, EventArgs e)

{

List<productC\_BL> lst = productC\_DL.getProductList();

gvProducts.DataSource = lst;

}

private void gvProducts\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

productC\_BL product = (productC\_BL)gvProducts.CurrentRow.DataBoundItem;

if (gvProducts.Columns["Buy"].Index == e.ColumnIndex)

{

productC\_DL.buyerList(product);

MessageBox.Show("Product is Added to Your List Successfully");

}

}

public void dataBind()

{

gvProducts.DataSource = null;

gvProducts.DataSource = productC\_DL.getProductList();

gvProducts.Refresh();

}

private void bynCalculateBill\_Click(object sender, EventArgs e)

{

List<productC\_BL> lst = productC\_DL.getBuyedProductList();

/\*Form f = new frmShowBuyedProducts();

f.Show();\*/

}

}

}

**4- Manager Form:**

namespace bilingSystemGUI

{

public partial class frmManager : Form

{

public frmManager()

{

InitializeComponent();

}

private void addProductToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmAddNewProduct();

f.Show();

}

private void viewAllProductsInfoToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmViewAllProducts();

f.Show();

}

private void deleteProductInfoToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmDeleteProduct();

f.Show();

}

private void editProductInfoToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmjustEditProduct();

f.Show();

}

private void productsQuantityRecordToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmViewAllProducts();

f.Show();

}

private void addStaffMemberToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmAddNewStaffMember();

f.Show();

}

private void viewAllStaffMembersToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmShowAllStaffMembers();

f.Show();

}

private void deleteStaffMemberInfoToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmDeleteStaff();

f.Show();

}

private void editStaffMemberInfoToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmjustEditStaffInfo();

f.Show();

}

private void sendMessageToStaffToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmSendMessage();

f.Show();

}

private void btnBack\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmSignUp();

f.Show();

}

}

}

**5-Show All Products & Edit Products & Delete Products:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using bilingSystemGUI.BL;

using bilingSystemGUI.DL;

namespace bilingSystemGUI

{

public partial class frmShowAllStaffMembers : Form

{

public frmShowAllStaffMembers()

{

InitializeComponent();

}

private void btnBack\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmManager();

f.Show();

}

private void gvProducts\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

Staff\_BL member = (Staff\_BL)gvStaff.CurrentRow.DataBoundItem;

if (gvStaff.Columns["Delete"].Index == e.ColumnIndex)

{

List<Staff\_BL> lst = Staff\_DL.deleteStaffFromList(member);

Staff\_DL.storeDataInFile(lst);

dataBind();

MessageBox.Show("Staff info is deleted Successfully");

}

else if (gvStaff.Columns["Edit"].Index == e.ColumnIndex)

{

frmEditStaff f = new frmEditStaff(member);

f.ShowDialog();

dataBind();

MessageBox.Show("Staff info is edited Successfully");

}

}

public void dataBind()

{

gvStaff.DataSource = null;

gvStaff.DataSource = Staff\_DL.getStaffList();

gvStaff.Refresh();

}

private void frmShowAllStaffMembers\_Load(object sender, EventArgs e)

{

List<Staff\_BL> lst = Staff\_DL.getStaffList();

gvStaff.DataSource = lst;

}

}

}

**6-Show Staff Members & Edit Staff Members & Delete Staff Members:**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using bilingSystemGUI.BL;

using bilingSystemGUI.DL;

namespace bilingSystemGUI

{

public partial class frmViewAllProducts : Form

{

public frmViewAllProducts()

{

InitializeComponent();

}

private void frmViewAllProducts\_Load(object sender, EventArgs e)

{

List<productC\_BL> lst = productC\_DL.getProductList();

gvProducts.DataSource = lst;

}

private void btnBack\_Click(object sender, EventArgs e)

{

this.Hide();

Form f = new frmManager();

f.Show();

}

private void gvProducts\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

productC\_BL product = (productC\_BL)gvProducts.CurrentRow.DataBoundItem;

if (gvProducts.Columns["Delete"].Index == e.ColumnIndex)

{

List<productC\_BL> lst = productC\_DL.deleteProductFromList(product);

productC\_DL.storeDataInFile(lst);

dataBind();

MessageBox.Show("Product is deleted Successfully");

}

else if(gvProducts.Columns["Edit"].Index == e.ColumnIndex)

{

frmEditProduct f = new frmEditProduct(product);

f.ShowDialog();

dataBind();

MessageBox.Show("Product is edited Successfully");

}

}

public void dataBind()

{

gvProducts.DataSource = null;

gvProducts.DataSource = productC\_DL.getProductList();

gvProducts.Refresh();

}

}

}

**BL:**

**1-MUser BL**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace bilingSystemGUI.BL

{

class MUserBL

{

private string name;

private string password;

private string role;

public string Name { get => name; set => name = value; }

public string Password { get => password; set => password = value; }

public string Role { get => role; set => role = value; }

public MUserBL(string name, string password, string role)

{

this.name = name;

this.password = password;

this.Role = role;

}

public MUserBL(string name, string password)

{

this.name = name;

this.password = password;

Role = null;

}

public MUserBL() { }

}

}

**2-Product BL:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace bilingSystemGUI.BL

{

public class productC\_BL

{

private string name;

private float price;

private float discount;

private float quantity;

public string Name { get => name; set => name = value; }

public float Price { get => price; set => price = value; }

public float Discount { get => discount; set => discount = value; }

public float Quantity { get => quantity; set => quantity = value; }

public productC\_BL(string name, float price, float discount)

{

this.Name = name;

this.Price = price;

this.Discount = discount;

}

public productC\_BL()

{

}

}

}

**3- Staff BL:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace bilingSystemGUI.BL

{

public class Staff\_BL

{

private string name;

private string idCard;

private float salary;

public string Name { get => name; set => name = value; }

public string IdCard { get => idCard; set => idCard = value; }

public float Salary { get => salary; set => salary = value; }

}

}

**DL:**

**1- MUser DL:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

using bilingSystemGUI.BL;

namespace bilingSystemGUI.DL

{

class MUserDL

{

private static string path = "dataSource.txt";

public static void addIntoFile(MUserBL newUser)

{

StreamWriter f = new StreamWriter(path, true);

f.WriteLine(newUser.Name + "," + newUser.Password + "," + newUser.Role);

f.Flush();

f.Close();

}

public static List<MUserBL> getUsersList()

{

List<MUserBL> lst = new List<MUserBL>();

StreamReader f = new StreamReader(path);

string line;

string[] splittedLine;

while ((line = f.ReadLine()) != null)

{

MUserBL user = new MUserBL();

splittedLine = line.Split(',');

user.Name = splittedLine[0];

user.Password = splittedLine[1];

user.Role = splittedLine[2];

lst.Add(user);

}

f.Close();

return lst;

}

public static MUserBL checkFromFile(MUserBL temp)

{

List<MUserBL> lst = getUsersList();

foreach (MUserBL user in lst)

{

if (user.Name == temp.Name && user.Password == temp.Password)

{

return user;

}

}

return null;

}

}

}

**2-Product DL:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using bilingSystemGUI.BL;

using System.IO;

namespace bilingSystemGUI.DL

{

class productC\_DL

{

private static List<productC\_BL> BuyProductlst = new List<productC\_BL>();

private static string path = "productsFile.txt";

public static void addProductToFile(productC\_BL product)

{

StreamWriter f = new StreamWriter(path, true);

f.WriteLine(product.Name + "," + product.Price + "," + product.Discount + "," + product.Quantity);

f.Close();

}

public static List<productC\_BL> getProductList()

{

List<productC\_BL> lst = new List<productC\_BL>();

StreamReader f = new StreamReader(path);

string line;

string[] productInfo;

while ((line = f.ReadLine()) != null)

{

productInfo = line.Split(',');

productC\_BL p = new productC\_BL();

p.Name = productInfo[0];

p.Price = float.Parse(productInfo[1]);

p.Discount = float.Parse(productInfo[2]);

p.Quantity = float.Parse(productInfo[3]);

lst.Add(p);

}

f.Close();

return lst;

}

public static List<productC\_BL> deleteProductFromList(productC\_BL temp)

{

List<productC\_BL> lst = getProductList();

for(int i =0; i< lst.Count; i++)

{

productC\_BL product = lst[i];

if(temp.Name == product.Name && temp.Price == product.Price && temp.Discount == product.Discount && temp.Quantity == temp.Quantity)

{

lst.RemoveAt(i);

}

}

return lst;

}

public static void storeDataInFile(List<productC\_BL> lst)

{

StreamWriter f = new StreamWriter(path);

foreach(productC\_BL product in lst)

{

f.WriteLine(product.Name + "," + product.Price + "," + product.Discount + "," + product.Quantity);

}

f.Flush();

f.Close();

}

public static List<productC\_BL> editProductInList(productC\_BL previous, productC\_BL update)

{

List<productC\_BL> lst = getProductList();

for ( int i =0; i< lst.Count; i++)

{

productC\_BL temp = new productC\_BL();

temp = lst[i];

if (temp.Name == previous.Name && temp.Price == previous.Price && temp.Discount == previous.Discount && temp.Quantity == previous.Quantity)

{

lst.RemoveAt(i);

lst.Insert(i, update);

}

}

return lst;

}

public static void buyerList(productC\_BL product)

{

BuyProductlst.Add(product);

}

public static List<productC\_BL> getBuyedProductList()

{

return BuyProductlst;

}

}

}

**3- Staff DL:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using bilingSystemGUI.BL;

using bilingSystemGUI.DL;

using System.IO;

namespace bilingSystemGUI.DL

{

class Staff\_DL

{

private static string path = "staff\_File.txt";

public static void addStaffToFile(Staff\_BL member)

{

StreamWriter f = new StreamWriter(path, true);

f.WriteLine(member.Name + "," + member.IdCard + "," + member.Salary);

f.Flush();

f.Close();

}

public static List<Staff\_BL> getStaffList()

{

List<Staff\_BL> lst = new List<Staff\_BL>();

StreamReader f = new StreamReader(path);

string line;

string[] staffInfo;

while ((line = f.ReadLine()) != null)

{

staffInfo = line.Split(',');

Staff\_BL s = new Staff\_BL();

s.Name = staffInfo[0];

s.IdCard = staffInfo[1];

s.Salary = float.Parse(staffInfo[2]);

lst.Add(s);

}

f.Close();

return lst;

}

public static List<Staff\_BL> deleteStaffFromList(Staff\_BL member)

{

List<Staff\_BL> lst = getStaffList();

for (int i = 0; i < lst.Count; i++)

{

Staff\_BL temp = lst[i];

if (temp.Name == member.Name && temp.IdCard == member.IdCard && temp.Salary == member.Salary)

{

lst.RemoveAt(i);

}

}

return lst;

}

public static void storeDataInFile(List<Staff\_BL> lst)

{

StreamWriter f = new StreamWriter(path);

foreach (Staff\_BL staff in lst)

{

f.WriteLine(staff.Name + "," + staff.IdCard + "," + staff.Salary);

}

f.Flush();

f.Close();

}

public static List<Staff\_BL> editStaffInList(Staff\_BL previous, Staff\_BL update)

{

List<Staff\_BL> lst = getStaffList();

for (int i = 0; i < lst.Count; i++)

{

Staff\_BL temp = new Staff\_BL();

temp = lst[i];

if (temp.Name == previous.Name && temp.IdCard == previous.IdCard && temp.Salary == previous.Salary)

{

lst.RemoveAt(i);

lst.Insert(i, update);

}

}

return lst;

}

}

}