**Group Member: Syed Ans Fraz Subhani**

**Fa21-BCS-030**

***Visual Programming***

**Sir MAzhar Bukhari**

**Super Market Management System**

**Documentation**

**1. Introduction**

The Super Market Management System is designed to streamline the management of a supermarket by providing efficient tools for handling products, suppliers, users, and cashier operations. This documentation serves as a guide for understanding and utilizing the various features of the system.

**2. System Architecture**

The system comprises the following components:

**Login:** Get your info from the database and decide your form accordingly.

**Dashboard:** Provides an overview of key metrics.

**Product Management:** Allows the addition, editing, and removal of products.

**Supplier Management:** Facilitates the management of supplier information.

**User Management:** Enables the addition, editing, and removal of users.

**Cashier Module:** Includes a cart system, product search, discount application, and automatic GST calculation.

**Announcement System:** Displays admin announcements in the Cashier Form.

**3. Technologies Used**

C#: Main programming language for application development.

SQL Server: Database management system.

.NET Framework: Required runtime for the application.

**4. Installation**

* Install .NET Framework on the target machine.
* Set up a SQL Server database for the Super Market Management System.

**5. Database Design**

**Entity-Relationship Diagram (ERD)**

The database consists of tables such as Products, Brands, Suppliers, Categories, Users, Announcements, Cart, etc.

**6. Database Connection**

string connectionString = "your\_database\_connection\_string";

SqlConnection connection = new SqlConnection(connectionString);

**7. Code Structure**

**The project is organized into modules:**

* Dashboard
* Product Management
* Supplier Management
* User Management
* Cashier Module
* Announcement System

**8. User Interfaces (UI)**

**Login Form:**

namespace Super\_Market\_Management\_Store

{

public partial class login : Form

{

public login()

{

InitializeComponent();

}

SqlDataReader dr;

bool found = false;

string role;

bool active;

string passwordd;

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

{

}

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

{

}

private void submitbtn\_Click\_1(object sender, EventArgs e)

{

try

{

string conStr = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con = new SqlConnection(conStr);

con.Open();

string query = "SELECT \* FROM users WHERE username = @username AND password = @password";

SqlCommand cmd = new SqlCommand(query, con);

// Add parameters

cmd.Parameters.AddWithValue("@username", nametb.Text);

cmd.Parameters.AddWithValue("@password", passwrdtb.Text);

dr = cmd.ExecuteReader();

dr.Read();

if (dr.HasRows)

{

found = true;

role = dr["role"].ToString();

active = bool.Parse(dr["isactive"].ToString());

passwordd = dr["password"].ToString();

}

else

{

found = false;

MessageBox.Show("Invalid Name or Password");

}

con.Close();

dr.Close();

if(found)

{

if(!active)

{

MessageBox.Show("Account INACTIVE!", "ACCESS DENIED", MessageBoxButtons.OK, MessageBoxIcon.Information);

return;

}

if(role=="Cashier")

{

MessageBox.Show("Welcome!", "ACCESS GRANTED", MessageBoxButtons.OK, MessageBoxIcon.Information);

this.Hide();

Cashier cashier = new Cashier();

cashier.ShowDialog();

}

if(role=="Admin")

{

MessageBox.Show("Welcome!", "ACCESS GRANTED", MessageBoxButtons.OK, MessageBoxIcon.Information);

this.Hide();

mainForm mainForm = new mainForm();

mainForm.ShowDialog();

}

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

throw;

}

}

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

{

if (MessageBox.Show("Are you sure you want to Exit?", "", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

Environment.Exit(0);

}

}

private void pictureBox1\_Click(object sender, EventArgs e)

{

}

}

**A screenshot of a computer

Description automatically generated**

**Admin Side Dashboard:**

namespace Super\_Market\_Management\_Store

{

public partial class Dashboard : Form

{

private string[] quotes = {

"The only way to do great work is to love what you do. - Steve Jobs",

"Success is not final, failure is not fatal: It is the courage to continue that counts. - Winston Churchill",

"Your time is limited, don't waste it living someone else's life. - Steve Jobs",

"Believe you can and you're halfway there. - Theodore Roosevelt",

"The future belongs to those who believe in the beauty of their dreams. - Eleanor Roosevelt",

"Strive not to be a success, but rather to be of value. - Albert Einstein",

"I have not failed. I've just found 10,000 ways that won't work. - Thomas A. Edison",

"It is during our darkest moments that we must focus to see the light. - Aristotle",

"The only limit to our realization of tomorrow will be our doubts of today. - Franklin D. Roosevelt",

"Do not wait to strike till the iron is hot, but make it hot by striking. - William Butler Yeats",

"In the end, we will remember not the words of our enemies, but the silence of our friends. - Martin Luther King Jr.",

"The best way to predict the future is to create it. - Peter Drucker",

"The only person you are destined to become is the person you decide to be. - Ralph Waldo Emerson",

"Success usually comes to those who are too busy to be looking for it. - Henry David Thoreau",

"The only impossible journey is the one you never begin. - Tony Robbins"

};

private Random random = new Random();

public Dashboard()

{

InitializeComponent();

int hour = DateTime.Now.Hour;

if (hour >= 10 && hour < 17)

{

//Open 10:00am through 4:59pm

timinglbl.Text = "Store is Open";

}

else

{

//Closed 5:00pm through 9:59am

timinglbl.Text = "Store is Closed";

}

if (hour >= 5 && hour < 12)

{

greetinglbl.Text = "Good Morning Sir!";

}

else if (hour >= 12 && hour < 15)

{

greetinglbl.Text = "Good Noon Sir!";

}

else if (hour >= 15 && hour < 19)

{

greetinglbl.Text = "Good Afternoon Sir!";

}

else

{

greetinglbl.Text = "Good Night Sir!";

}

}

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

{

DisplayRandomQuote();

string conStr = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con = new SqlConnection(conStr);

con.Open();

string query = "SELECT SUM(Quantity) FROM Nameofproduct;";

SqlCommand cmd = new SqlCommand(query, con);

object result = cmd.ExecuteScalar();

// Check if the result is not null before converting to a string

if (result != null)

{

string sumTotal = result.ToString();

// Assuming you have a label named 'label1' on your form

totalproductlbl.Text = "" + sumTotal;

}

else

{

totalproductlbl.Text = "No data found"; // Or handle the case when there is no data

}

con.Close();

string conStr2 = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con2 = new SqlConnection(conStr2);

con2.Open();

string query2 = "SELECT SUM(Total) FROM salesrecord;";

SqlCommand cmd2 = new SqlCommand(query2, con2);

object result2 = cmd2.ExecuteScalar();

// Check if the result is not null before converting to a string

if (result2 != null)

{

string sumTotal = result2.ToString();

// Assuming you have a label named 'label1' on your form

earninglbl.Text = "" + sumTotal;

}

else

{

earninglbl.Text = "No data found"; // Or handle the case when there is no data

}

con2.Close();

string conStr23 = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con23 = new SqlConnection(conStr23);

con23.Open();

string query23 = "SELECT COUNT(Description) FROM Nameofproduct;";

SqlCommand cmd23 = new SqlCommand(query23, con23);

object result23 = cmd23.ExecuteScalar();

// Check if the result is not null before converting to a string

if (result23 != null)

{

string sumTotal = result23.ToString();

// Assuming you have a label named 'label1' on your form

productslbl.Text = "" + sumTotal;

}

else

{

productslbl.Text = "No data found"; // Or handle the case when there is no data

}

con2.Close();

timer1.Start();

}

private void DisplayRandomQuote()

{

// Get a random index to select a quote from the array

int randomIndex = random.Next(quotes.Length);

// Display the selected quote on the label

qoutelbl.Text = quotes[randomIndex];

}

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

{

checkedListBox1.Items.Add(textBox1.Text);

textBox1.Text = "";

}

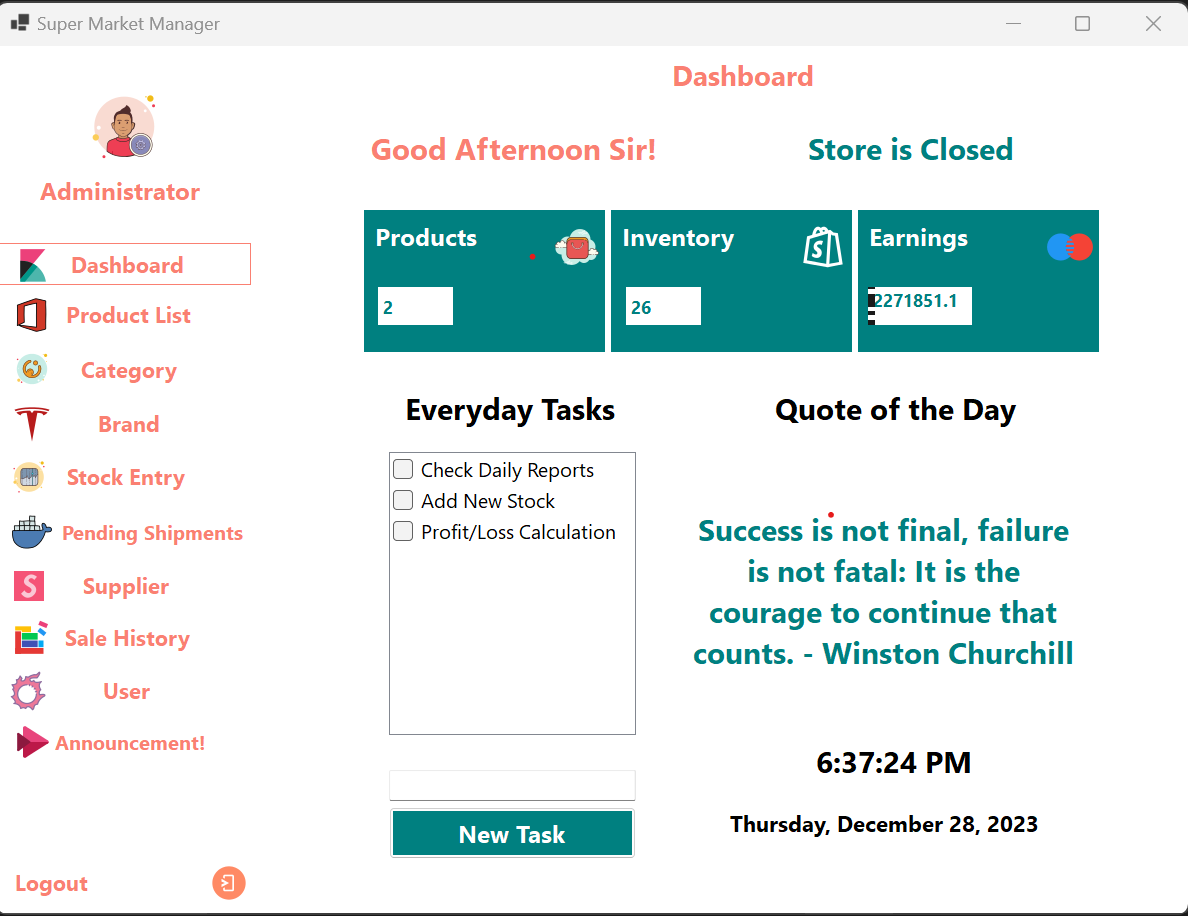
private void timer1\_Tick(object sender, EventArgs e)

{

timelbl.Text = DateTime.Now.ToLongTimeString();

datelbl.Text = DateTime.Now.ToLongDateString();

}

****

**Product Form**

namespace Super\_Market\_Management\_Store

{

public partial class Products : Form

{

public Products()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataRow dRow;

int MaxRows;

int inc = 0;

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL3;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadproduct();

}

public void loadproduct()

{

dataGridView1.Rows.Clear();

inc = 0;

while (inc < MaxRows)

{

dRow = ds.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void button1\_Click(object sender, EventArgs e)

{

this.Dispose();

}

private void button3\_Click(object sender, EventArgs e)

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds);

MaxRows = ds.Tables[0].Rows.Count;

}

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

{

ProductModule productModule = new ProductModule();

productModule.ShowDialog();

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

private void button2\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL3;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadproduct();

}

**A screenshot of a computer

Description automatically generated**

**Product Module**

namespace Super\_Market\_Management\_Store

{

public partial class ProductModule : Form

{

public ProductModule()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataSet ds2;

DataSet ds3;

DataRow dRow;

DataRow dRow2;

DataRow newRow;

int MaxRows = 0;

int inc = 0;

private void brandcombobox\_SelectedIndexChanged(object sender, EventArgs e)

{

}

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL;

ds = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds2 = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.SQL3;

ds3 = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadbrandincombobox();

loadcategoryincombobox();

}

private void loadbrandincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

brandcombobox.Items.Add(row["Brand"].ToString());

}

}

}

public void loadcategoryincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds2.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

categorycombobox.Items.Add(row["Category"].ToString());

}

}

}

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

{

this.Close();

}

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

{

inc = 0;

if (MessageBox.Show("Are you sure you want to add this brand?", "", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

newRow = ds3.Tables[0].NewRow();

newRow["Description"] = descriptiontb.Text;

newRow["Barcode"] = barcodetb.Text;

newRow["Brand"] = brandcombobox.Text.ToString(); // Assuming Brand is the column name

newRow["Category"] = categorycombobox.Text.ToString(); // Assuming Category is the column name

newRow["Price"] = float.Parse(pricebox.Text);

newRow["Quantity"] = quantitybox.Value;

ds3.Tables[0].Rows.Add(newRow);

try

{

objConnect.UpdateDatabase(ds3);

MaxRows = MaxRows + 1;

inc = MaxRows - 1;

MessageBox.Show("Database updated");

this.Dispose();

Products products = new Products();

mainForm mainform = new mainForm();

mainform.openChildForm(new Products());

}

catch (Exception err)

{

MessageBox.Show(err.Message);

}

}

}

}

}

A screenshot of a computer

Description automatically generated

**Category**

namespace Super\_Market\_Management\_Store

{

public partial class Category : Form

{

public Category()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataRow dRow;

int MaxRows;

int inc = 0;

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

public void loadcategory()

{

while (inc < MaxRows)

{

dRow = ds.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

}

private void btnnewbrand\_Click\_1(object sender, EventArgs e)

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click\_1(object sender, EventArgs e)

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds);

MaxRows = ds.Tables[0].Rows.Count;

}

private void button2\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

this.Dispose();

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

}

}

A screenshot of a computer

Description automatically generated

**Category Module**

namespace Super\_Market\_Management\_Store

{

public partial class Category : Form

{

public Category()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataRow dRow;

int MaxRows;

int inc = 0;

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

public void loadcategory()

{

while (inc < MaxRows)

{

dRow = ds.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

}

private void btnnewbrand\_Click\_1(object sender, EventArgs e)

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click\_1(object sender, EventArgs e)

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds);

MaxRows = ds.Tables[0].Rows.Count;

}

private void button2\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

this.Dispose();

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

}

}

A screenshot of a computer

Description automatically generated

**Brand**

namespace Super\_Market\_Management\_Store

{

public partial class Category : Form

{

public Category()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataRow dRow;

int MaxRows;

int inc = 0;

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

public void loadcategory()

{

while (inc < MaxRows)

{

dRow = ds.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

}

private void btnnewbrand\_Click\_1(object sender, EventArgs e)

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click\_1(object sender, EventArgs e)

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds);

MaxRows = ds.Tables[0].Rows.Count;

}

private void button2\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

this.Dispose();

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

}

}

A screenshot of a computer

Description automatically generated

**Brand Module**

namespace Super\_Market\_Management\_Store

{

public partial class Category : Form

{

public Category()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataRow dRow;

int MaxRows;

int inc = 0;

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

public void loadcategory()

{

while (inc < MaxRows)

{

dRow = ds.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

}

private void btnnewbrand\_Click\_1(object sender, EventArgs e)

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click\_1(object sender, EventArgs e)

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds);

MaxRows = ds.Tables[0].Rows.Count;

}

private void button2\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

this.Dispose();

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

}

}

**A screenshot of a computer

Description automatically generated**

**Stock Entry**

using Super\_Market\_Management\_Store.Properties;

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;

namespace Super\_Market\_Management\_Store

{

public partial class Category : Form

{

public Category()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataRow dRow;

int MaxRows;

int inc = 0;

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

public void loadcategory()

{

while (inc < MaxRows)

{

dRow = ds.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

}

private void btnnewbrand\_Click\_1(object sender, EventArgs e)

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click\_1(object sender, EventArgs e)

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds);

MaxRows = ds.Tables[0].Rows.Count;

}

private void button2\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

this.Dispose();

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

}

}

**A screenshot of a computer

Description automatically generated**

**Supplier**

using Super\_Market\_Management\_Store.Properties;

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;

namespace Super\_Market\_Management\_Store

{

public partial class Category : Form

{

public Category()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataRow dRow;

int MaxRows;

int inc = 0;

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

public void loadcategory()

{

while (inc < MaxRows)

{

dRow = ds.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

}

private void btnnewbrand\_Click\_1(object sender, EventArgs e)

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click\_1(object sender, EventArgs e)

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds);

MaxRows = ds.Tables[0].Rows.Count;

}

private void button2\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

this.Dispose();

}

private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

}

}

**A screenshot of a computer

Description automatically generated**

**Supplier Module**

using Super\_Market\_Management\_Store.Properties;

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;

namespace Super\_Market\_Management\_Store

{

public partial class Category : Form

{

public Category()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataRow dRow;

int MaxRows;

int inc = 0;

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

public void loadcategory()

{

while (inc < MaxRows)

{

dRow = ds.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

this.Hide();

}

private void btnnewbrand\_Click\_1(object sender, EventArgs e)

{

CategoryModule categoryModule = new CategoryModule();

categoryModule.ShowDialog();

}

private void button3\_Click\_1(object sender, EventArgs e)

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds);

MaxRows = ds.Tables[0].Rows.Count;

}

private void button2\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadcategory();

}

private void button1\_Click\_1(object sender, EventArgs e)

{

this.Dispose();

}

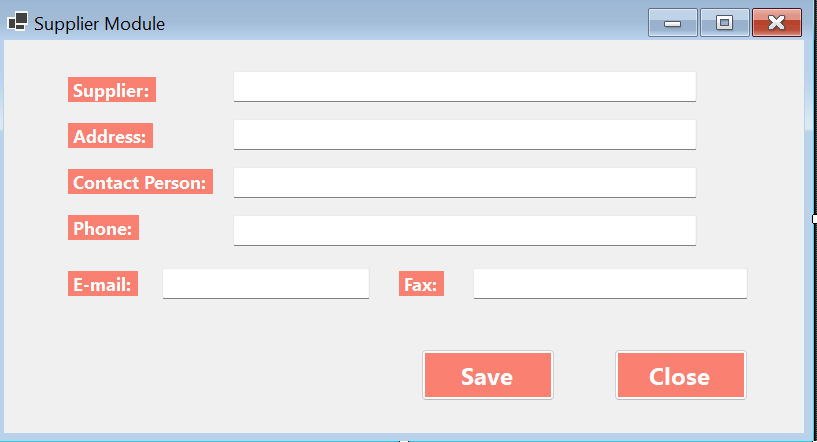
private void dataGridView1\_CellContentClick\_1(object sender, DataGridViewCellEventArgs e)

{

}

}

}

****

**Upcoming Shipments**

using Super\_Market\_Management\_Store.Properties;

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Security.Cryptography;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Super\_Market\_Management\_Store

{

public partial class UpcomingShipments : Form

{

public UpcomingShipments()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataSet ds2;

DataSet ds3;

DataSet ds4;

DataRow dRow;

DataRow dRow2;

DataRow newRow;

int MaxRows = 0;

int inc = 0;

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

inc = 0;

if (MessageBox.Show("Are you sure you want to add this brand?", "", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

newRow = ds3.Tables[0].NewRow();

newRow["Reference No"] = rntb.Text;

newRow["Product"] = ptb.Text;

newRow["Quantity"] = qbox.Value;// Assuming Brand is the column name

newRow["Brand"] = brandcombobox.Text.ToString();// Assuming Category is the column name

newRow["Category"] = categorycombobox.Text.ToString();

newRow["Supplier"] = suppliercombobox.Text.ToString();

newRow["Date"] = datebox.Value;

ds3.Tables[0].Rows.Add(newRow);

try

{

objConnect.UpdateDatabase(ds3);

MaxRows = MaxRows + 1;

inc = MaxRows - 1;

MessageBox.Show("Database updated");

}

catch (Exception err)

{

MessageBox.Show(err.Message);

}

}

}

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL;

ds = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds2 = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.Supplier;

ds4 = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.upcomingstock;

ds3 = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadbrandincombobox();

loadcategoryincombobox();

loadsupplierincombobox();

loadproduct();

}

public void loadproduct()

{

dataGridView1.Rows.Clear();

inc = 0;

MaxRows = ds3.Tables[0].Rows.Count;

while (inc < MaxRows)

{

dRow = ds3.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void loadsupplierincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds4.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

suppliercombobox.Items.Add(row["Supplier"].ToString());

}

}

}

private void loadbrandincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

brandcombobox.Items.Add(row["Brand"].ToString());

}

}

}

public void loadcategoryincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds2.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

categorycombobox.Items.Add(row["Category"].ToString());

}

}

}

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

{

this.Dispose();

}

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

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds3.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds3);

MaxRows = ds3.Tables[0].Rows.Count;

}

private void button1\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.upcomingstock;

ds3 = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadproduct();

}

}

}

**A screenshot of a computer

Description automatically generated**

**Users**

using Super\_Market\_Management\_Store.Properties;

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Security.Cryptography;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Super\_Market\_Management\_Store

{

public partial class UpcomingShipments : Form

{

public UpcomingShipments()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataSet ds2;

DataSet ds3;

DataSet ds4;

DataRow dRow;

DataRow dRow2;

DataRow newRow;

int MaxRows = 0;

int inc = 0;

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

inc = 0;

if (MessageBox.Show("Are you sure you want to add this brand?", "", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

newRow = ds3.Tables[0].NewRow();

newRow["Reference No"] = rntb.Text;

newRow["Product"] = ptb.Text;

newRow["Quantity"] = qbox.Value;// Assuming Brand is the column name

newRow["Brand"] = brandcombobox.Text.ToString();// Assuming Category is the column name

newRow["Category"] = categorycombobox.Text.ToString();

newRow["Supplier"] = suppliercombobox.Text.ToString();

newRow["Date"] = datebox.Value;

ds3.Tables[0].Rows.Add(newRow);

try

{

objConnect.UpdateDatabase(ds3);

MaxRows = MaxRows + 1;

inc = MaxRows - 1;

MessageBox.Show("Database updated");

}

catch (Exception err)

{

MessageBox.Show(err.Message);

}

}

}

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL;

ds = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds2 = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.Supplier;

ds4 = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.upcomingstock;

ds3 = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadbrandincombobox();

loadcategoryincombobox();

loadsupplierincombobox();

loadproduct();

}

public void loadproduct()

{

dataGridView1.Rows.Clear();

inc = 0;

MaxRows = ds3.Tables[0].Rows.Count;

while (inc < MaxRows)

{

dRow = ds3.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void loadsupplierincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds4.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

suppliercombobox.Items.Add(row["Supplier"].ToString());

}

}

}

private void loadbrandincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

brandcombobox.Items.Add(row["Brand"].ToString());

}

}

}

public void loadcategoryincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds2.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

categorycombobox.Items.Add(row["Category"].ToString());

}

}

}

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

{

this.Dispose();

}

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

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds3.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds3);

MaxRows = ds3.Tables[0].Rows.Count;

}

private void button1\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.upcomingstock;

ds3 = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadproduct();

}

}

}

**A screenshot of a computer

Description automatically generated**

**Announcement**

using Super\_Market\_Management\_Store.Properties;

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Security.Cryptography;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Super\_Market\_Management\_Store

{

public partial class UpcomingShipments : Form

{

public UpcomingShipments()

{

InitializeComponent();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataSet ds2;

DataSet ds3;

DataSet ds4;

DataRow dRow;

DataRow dRow2;

DataRow newRow;

int MaxRows = 0;

int inc = 0;

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

inc = 0;

if (MessageBox.Show("Are you sure you want to add this brand?", "", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

newRow = ds3.Tables[0].NewRow();

newRow["Reference No"] = rntb.Text;

newRow["Product"] = ptb.Text;

newRow["Quantity"] = qbox.Value;// Assuming Brand is the column name

newRow["Brand"] = brandcombobox.Text.ToString();// Assuming Category is the column name

newRow["Category"] = categorycombobox.Text.ToString();

newRow["Supplier"] = suppliercombobox.Text.ToString();

newRow["Date"] = datebox.Value;

ds3.Tables[0].Rows.Add(newRow);

try

{

objConnect.UpdateDatabase(ds3);

MaxRows = MaxRows + 1;

inc = MaxRows - 1;

MessageBox.Show("Database updated");

}

catch (Exception err)

{

MessageBox.Show(err.Message);

}

}

}

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.SQL;

ds = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.SQL2;

ds2 = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.Supplier;

ds4 = objConnect.GetConnection;

objConnect.Sql = Properties.Settings.Default.upcomingstock;

ds3 = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadbrandincombobox();

loadcategoryincombobox();

loadsupplierincombobox();

loadproduct();

}

public void loadproduct()

{

dataGridView1.Rows.Clear();

inc = 0;

MaxRows = ds3.Tables[0].Rows.Count;

while (inc < MaxRows)

{

dRow = ds3.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

private void loadsupplierincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds4.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

suppliercombobox.Items.Add(row["Supplier"].ToString());

}

}

}

private void loadbrandincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

brandcombobox.Items.Add(row["Brand"].ToString());

}

}

}

public void loadcategoryincombobox()

{

inc = 0;

{

// Assuming ds2 is the DataSet for the second database

DataTable table2 = ds2.Tables[0];

for (int i = 0; i < table2.Rows.Count; i++)

{

DataRow row = table2.Rows[i];

categorycombobox.Items.Add(row["Category"].ToString());

}

}

}

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

{

this.Dispose();

}

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

{

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds3.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds3);

MaxRows = ds3.Tables[0].Rows.Count;

}

private void button1\_Click(object sender, EventArgs e)

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.upcomingstock;

ds3 = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadproduct();

}

}

}

**A screen shot of a message

Description automatically generated**

* *After logout you will go to login form from where you can go to cashier by putting right username and password.*

**CASHIER FORM**

using Super\_Market\_Management\_Store.Properties;

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Data.SqlClient;

using System.Drawing;

using System.Linq;

using System.Security.Cryptography.X509Certificates;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Super\_Market\_Management\_Store

{

public partial class Cashier : Form

{

public Cashier()

{

InitializeComponent();

// lbldate.Text = DateTime.Now.ToString();

//setdesvalue();

}

DatabaseConnection objConnect;

string conString;

DataSet ds;

DataRow dRow;

int MaxRows;

int inc = 0;

int priceofoneproduct = 0;

double priceofwholecart = 0;

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.Announcement;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

loadAnnouncement();

timer2.Start();

}

public void loadAnnouncement()

{

dRow = ds.Tables[0].Rows[0];

lblannouncement.Text = dRow["Announcement"].ToString();

}

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

{

}

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

{

}

private void middlepanel\_Paint(object sender, PaintEventArgs e)

{

}

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

{

Search\_Product search = new Search\_Product();

search.Show();

}

public void SetTextBoxValues(string description, string brand, string category, string price, string barcode)

{

}

//private void btnnewproduct\_Click(object sender, EventArgs e)

//{

// Search\_Product newsearch = new Search\_Product();

// pricecalculation();

// this.Hide();

//newsearch.ShowDialog(this);

// }

public void pricecalculation()

{

try

{

// priceofoneproduct = Convert.ToInt32(pricebox.Text) \* Convert.ToInt32(quantitybox.Value);

// salestotallbl.Text = priceofoneproduct.ToString();

// discountlbl.Text = discountbox.Value.ToString();

double taxcalculate = Convert.ToDouble(priceofoneproduct) \* 0.05;

// double discountedprice = (priceofoneproduct / 100) \* Convert.ToDouble(discountbox.Value);

// taxlbl.Text = taxcalculate.ToString();

// priceofwholecart = priceofoneproduct + taxcalculate - discountedprice;

// maintotallbl.Text = priceofwholecart.ToString();

}

catch

{

MessageBox.Show("Please Fill this first");

}

}

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

{

pricecalculation();

}

private void endshoppingcheckbox\_CheckedChanged(object sender, EventArgs e)

{

// if (endshoppingcheckbox.Checked)

{

// proceedtocart.Enabled = true;

}

}

private void newproductcheckbox\_CheckedChanged(object sender, EventArgs e)

{

// if (newproductcheckbox.Checked)

{

/// proceedtocart.Enabled = true;

}

}

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

{

pricecalculation();

}

private void button1\_Click(object sender, EventArgs e)

{

}

private void proceedtocart\_Click\_1(object sender, EventArgs e)

{

}

private void paneltitle\_Paint(object sender, PaintEventArgs e)

{

}

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

{

if (MessageBox.Show("Are you sure you want to Logout?", "", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

this.Hide();

login login = new login();

login.ShowDialog();

}

}

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

{

ChangePassword changePassword = new ChangePassword();

changePassword.ShowDialog();

}

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

{

conString = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con2 = new SqlConnection(conString);

con2.Open();

string query2 = "DELETE From DummyCart;";

SqlCommand cmd2 = new SqlCommand(query2, con2);

cmd2.ExecuteNonQuery();

MessageBox.Show("Action Performed");

con2.Close();

}

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

{

Cart cart = new Cart();

cart.ShowDialog();

}

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

{

TotalSales totalSales = new TotalSales();

totalSales.ShowDialog();

}

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

{

}

private void label9\_Click(object sender, EventArgs e)

{

}

private void label4\_Click(object sender, EventArgs e)

{

}

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

{

}

private void timer2\_Tick(object sender, EventArgs e)

{

timmelbl.Text = DateTime.Now.ToLongTimeString();

datelbl.Text = DateTime.Now.ToLongDateString();

}

}

}

A screenshot of a phone

Description automatically generated

**Cart**

namespace Super\_Market\_Management\_Store

{

public partial class Cart : Form

{

public Cart()

{

InitializeComponent();

}

DataRow dRow;

int inc = 0;

DatabaseConnection objConnect;

string conString;

DataSet ds;

// DataRow dRow;

int MaxRows;

int rowindexx;

// int inc = 0;

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

{

string constr = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con = new SqlConnection(constr);

con.Open();

SqlCommand cmd = new SqlCommand("SELECT \* FROM DummyCart", con);

SqlDataAdapter adapter = new SqlDataAdapter(cmd);

DataSet dataTable = new DataSet();

adapter.Fill(dataTable);

int maxrow = dataTable.Tables[0].Rows.Count;

if (dataTable.Tables[0].Rows.Count != 0)

{

while (inc < maxrow)

{

dRow = dataTable.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

con.Close();

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

rowindexx = e.RowIndex;

}

private void button1\_Click(object sender, EventArgs e)

{

string conStr = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con = new SqlConnection(conStr);

con.Open();

string query = "UPDATE DummyCart SET Quantity = @newquantity WHERE Id=@productid";

SqlCommand cmd = new SqlCommand(query, con);

cmd.Parameters.AddWithValue("@newquantity", quantitybox.Value);

cmd.Parameters.AddWithValue("@productid", idtextbox.Text);

cmd.ExecuteNonQuery();

MessageBox.Show("Quantity Updated");

}

private void label6\_Click(object sender, EventArgs e)

{

}

public void loadproduct()

{

inc = 0;

while (inc < MaxRows)

{

dRow = ds.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

}

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

{

objConnect = new DatabaseConnection();

conString = Properties.Settings.Default.EmployeesConnectionString;

objConnect.connection\_string = conString;

objConnect.Sql = Properties.Settings.Default.removefromcart;

ds = objConnect.GetConnection;

MaxRows = ds.Tables[0].Rows.Count;

int rowIndex = dataGridView1.CurrentCell.RowIndex;

dataGridView1.Rows.RemoveAt(rowIndex);

ds.Tables[0].Rows[rowIndex].Delete();

objConnect.UpdateDatabase(ds);

MaxRows = ds.Tables[0].Rows.Count;

}

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

{

quantityupdation();

string conStr = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con = new SqlConnection(conStr);

con.Open();

string query = "INSERT INTO salesrecord (Description, Brand, Category, Price, Quantity, Barcode, Discount, Total) SELECT Description, Brand, Category, Price, Quantity, Barcode, @Discount AS Discount,((Price\*Quantity)-((Price\*Quantity)\*(@Discount/100)))\*(1+0.07) AS Total FROM DummyCart;";

SqlCommand cmd = new SqlCommand(query, con);

cmd.Parameters.AddWithValue("@Discount", discountbox.Value);

cmd.ExecuteNonQuery();

con.Close();

MessageBox.Show("Data Entered in salesrecord");

SqlConnection con2 = new SqlConnection(conStr);

con2.Open();

string query2 = "DELETE From DummyCart;";

SqlCommand cmd2 = new SqlCommand(query2, con2);

cmd2.ExecuteNonQuery();

con2.Close();

this.Close();

string conStr12 = Properties.Settings.Default.EmployeesConnectionString;

using (SqlConnection con12 = new SqlConnection(conStr12))

{

con12.Open();

string query12 = "UPDATE Nameofproduct SET Nameofproduct.Quantity = Nameofproduct.Quantity - DummyCart.Quantity FROM Nameofproduct JOIN DummyCart ON Nameofproduct.Id = DummyCart.Id";

using (SqlCommand cmd12 = new SqlCommand(query12, con12))

{

cmd12.ExecuteNonQuery();

}

con12.Close();

}

}

private void button2\_Click(object sender, EventArgs e)

{

string conStr2 = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con2 = new SqlConnection(conStr2);

con2.Open();

string query2 = "UPDATE DummyCart SET Totalcart= Totalcart\*Quantity";

SqlCommand cmd2 = new SqlCommand(query2, con2);

cmd2.Parameters.AddWithValue("@newquantity", quantitybox.Value);

cmd2.Parameters.AddWithValue("@productid", idtextbox.Text);

cmd2.ExecuteNonQuery();

con2.Close();

string conStr23 = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con23 = new SqlConnection(conStr2);

con23.Open();

string query23 = "UPDATE DummyCart SET Discount = @newquantity, Totalcart = ((Totalcart)-(Totalcart \*(@newquantity/100)))\*(1+0.07)";

SqlCommand cmd23 = new SqlCommand(query23, con23);

cmd23.Parameters.AddWithValue("@newquantity", discountbox.Value);

cmd23.ExecuteNonQuery();

con23.Close();

MessageBox.Show("Quantity and Discount performed");

string conStr = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con = new SqlConnection(conStr);

con.Open();

string query = "SELECT SUM(Totalcart) FROM DummyCart;";

SqlCommand cmd = new SqlCommand(query, con);

object result = cmd.ExecuteScalar();

// Check if the result is not null before converting to a string

if (result != null)

{

string sumTotal = result.ToString();

// Assuming you have a label named 'label1' on your form

totalamounttb.Text = "" + sumTotal;

MessageBox.Show(sumTotal);

}

else

{

totalamounttb.Text = "No data found"; // Or handle the case when there is no data

}

con.Close();

}

public void quantityupdation()

{

try {

string conStr = Properties.Settings.Default.EmployeesConnectionString;

using (SqlConnection con = new SqlConnection(conStr))

{

con.Open();

string query = "UPDATE Nameofproduct SET Nameofproduct.Quantity = Nameofproduct.Quantity - DummyCart.Quantity FROM Nameofproduct JOIN DummyCart ON Nameofproduct.Id = DummyCart.Id";

using (SqlCommand cmd = new SqlCommand(query, con))

{

cmd.ExecuteNonQuery();

}

con.Close();

}

}

catch {

}

}

private void button1\_Click\_1(object sender, EventArgs e)

{

//quantityupdation();

try

{

string conStr2 = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con2 = new SqlConnection(conStr2);

con2.Open();

string query2 = "UPDATE DummyCart SET Quantity= @newquantity WHERE Id=@productid AND @newquantity <= (SELECT Quantity FROM Nameofproduct WHERE Id = @productid)";

SqlCommand cmd2 = new SqlCommand(query2, con2);

cmd2.Parameters.AddWithValue("@newquantity", quantitybox.Value);

cmd2.Parameters.AddWithValue("@productid", idtextbox.Text);

int rowsAffected = cmd2.ExecuteNonQuery();

if (rowsAffected > 0)

{

// The update was successful

MessageBox.Show("Quantity updated");

}

else

{

// No rows were affected, indicating the condition was not met

MessageBox.Show("Inventory doesnot contain that much quantity");

}

con2.Close();

string constr12 = Properties.Settings.Default.EmployeesConnectionString;

SqlConnection con12 = new SqlConnection(constr12);

con12.Open();

SqlCommand cmd12 = new SqlCommand("SELECT \* FROM DummyCart", con12);

SqlDataAdapter adapter = new SqlDataAdapter(cmd12);

DataSet dataTable = new DataSet();

adapter.Fill(dataTable);

int maxrow = dataTable.Tables[0].Rows.Count;

if (dataTable.Tables[0].Rows.Count != 0)

{

inc = 0;

dataGridView1.Rows.Clear();

while (inc < maxrow)

{

dRow = dataTable.Tables[0].Rows[inc];

dataGridView1.Rows.Add(dRow.ItemArray);

inc++;

}

con12.Close();

}

catch (Exception)

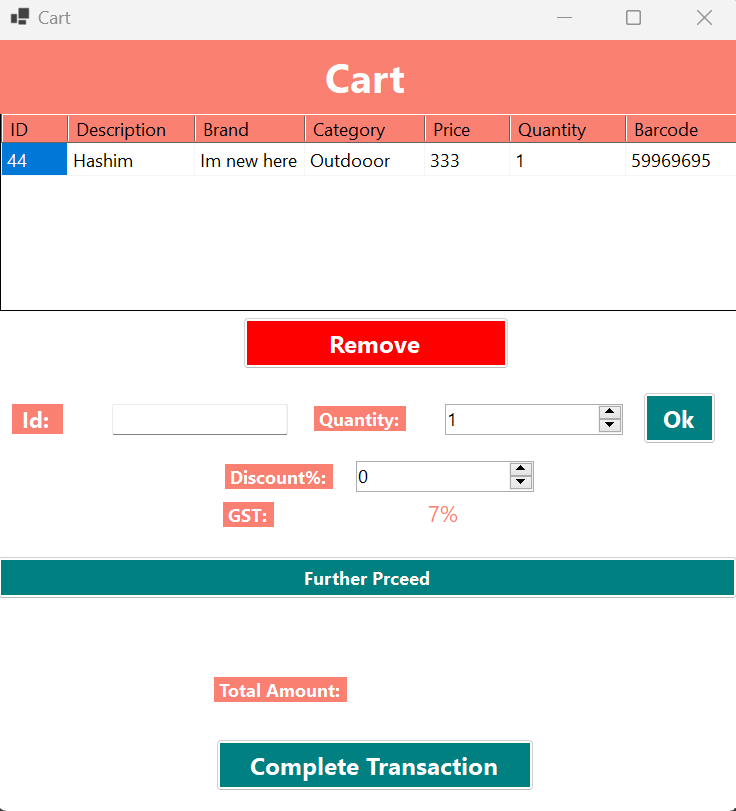
{

throw;

}

}

}}



**Database Tables:**

* **Product Table**
* **Category’s Table**
* **Brands Table**
* **New Shipments Table**
* **Users Table**
* **Sales Record Table**

*Further I will show you in the class and explain during viva.*

**THE END**