

North Western University, Khulna

**Department of Computer Science and Engineering**

# Project Title: "Blood Bank Management System"

**Course Title:** Database Systems Sessional

**Course Code:** CSE-3108

# Submitted By: Team – 3 Bumbling

**Tarek Ahmmed ID: 20181023010**

**Erina Momtaz ID: 20181054010**

**Md. Hasanul Banna Khan Abir ID: 20181004010**

**Submitted To:**

**M. Raihan**

Lecturer

Computer Science and Engineering Department,

**North Western University, Khulna.**

Date of Submission: 13. 03. 2020 …………………………………………..

Signature

# Introduction:

The project blood bank management system is known to be a pilot project that is designed for the blood bank to gather blood from various sources and distribute it to the needy people who have high requirements for it. The software is designed to handle the daily transactions of the blood bank and search the details when required. It also helps to register the details of donors, blood collection details as well as blood issued reports. This software application is password protected and it is designed in such a way that some individual users have their login access only. The software application is designed in such a manner that it can suit the needs of all the blood bank requirements in the course of future.

# Description:

The BLOOD BANK MANAGEMENT SYSTEM project is designed for successful completion of a project on blood bank management system. The main aim of developing this system is to provide blood to the people who are in need of blood. The basic building aim is to provide blood donation service to the city recently. The Blood Bank Management System (BBMS) is a password protected application that is designed to store, process, retrieve and analyze information concerned with the administrative and inventory management within a blood bank. This project aims at maintaining all the information pertaining to blood donors, different blood groups available in each blood bank and help them manage in a better way.

Project Aim is to provide transparency in this field, make the process of obtaining blood from a blood bank hassle-free and corruption-free and make the system of blood bank management effective. Blood Bank donation system can collect blood from many donators in short from various sources and distribute that blood to needy people who require blood.

# Features:

1. Add, Update and Delete Users
2. Register Donors and Keep Their Details.
3. Admins can manage (Add, Update and Delete) Donors
4. Login Function
5. Splash Screen (Animation before loading application)

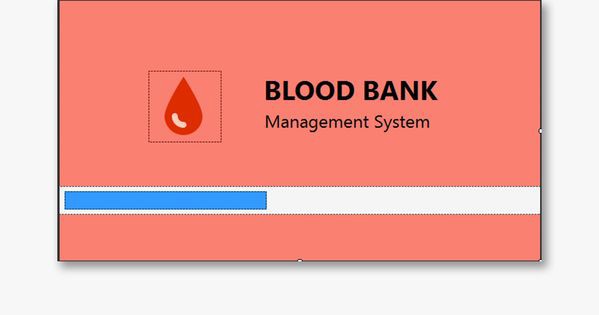
# Technology Used

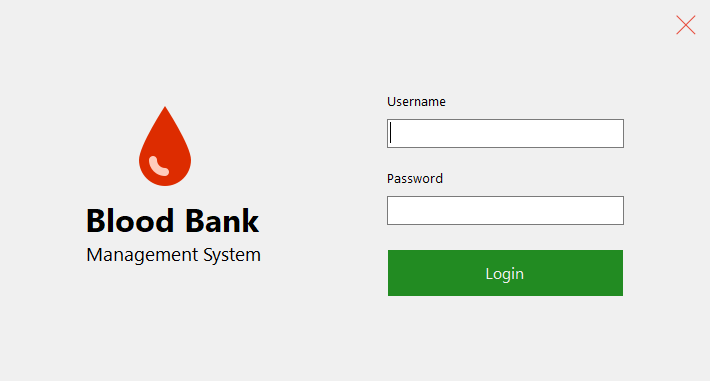
1. C# (C Sharp) Programming Language and 3-tier architecture of Software Development
2. MS SQL Server Database and Relational Database Management System (RDBMS)

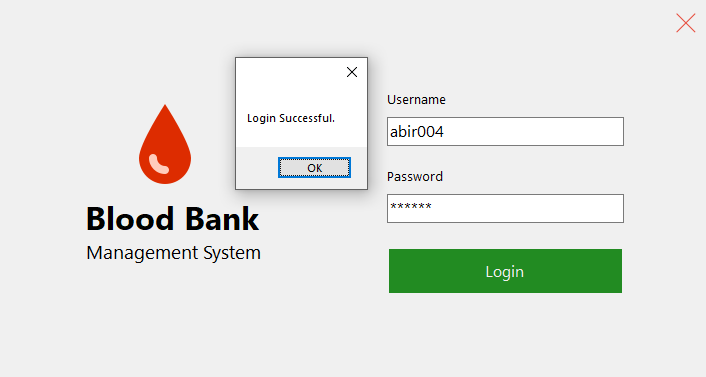
# Tools Used:

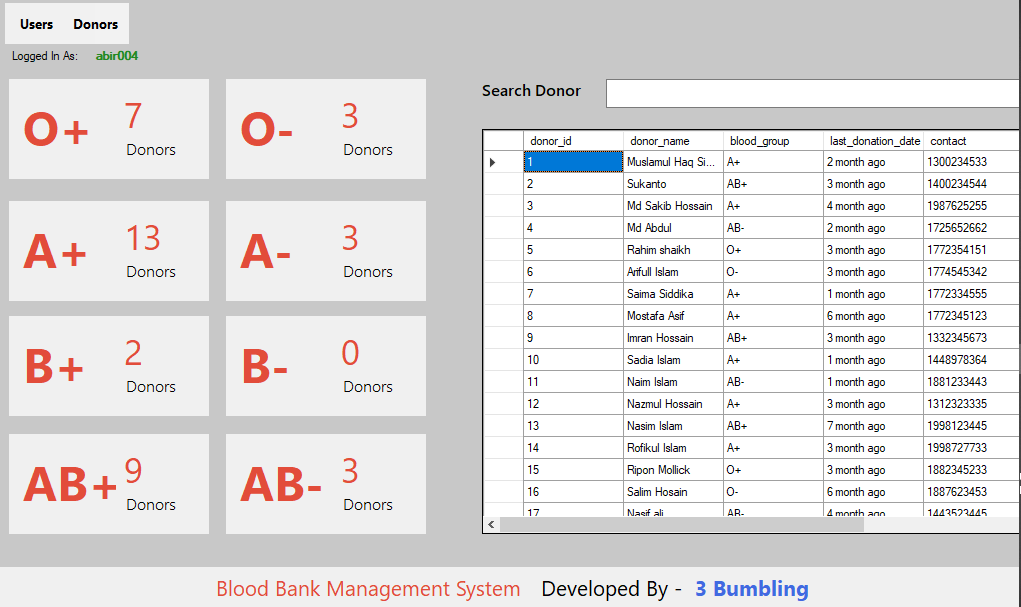
1. SQL Server Management Studio
2. Visual Studio

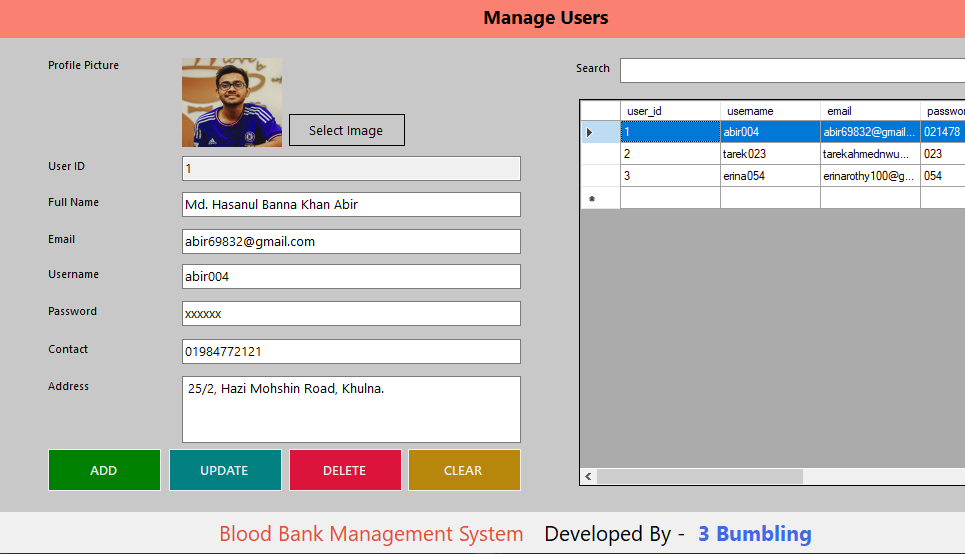
# ScreenShots:

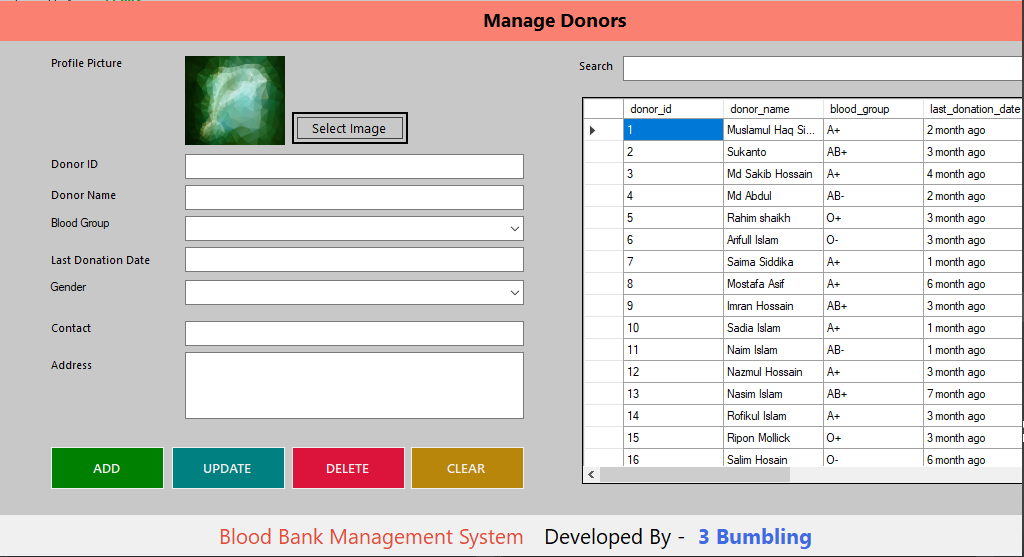












**Queries (SQL Commands):**

USE [master] GO

/\*\*\*\*\*\* Object: Database [BloodBankManagementSystem] Script Date: 28-Feb-20 9:30:24 AM \*\*\*\*\*\*/

CREATE DATABASE [BloodBankManagementSystem] CONTAINMENT = NONE

ON PRIMARY

( NAME = N'BloodBankManagementSystem', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL11.AUNGSTROMESQL\MSSQL\DATA\BloodBankManagementSystem.mdf' , SIZE = 11264KB , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )

LOG ON

( NAME = N'BloodBankManagementSystem\_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL11.AUNGSTROMESQL\MSSQL\DATA\BloodBankManagementSystem\_log.ldf' , SIZE = 1024KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)

GO

ALTER DATABASE [BloodBankManagementSystem] SET COMPATIBILITY\_LEVEL = 110 GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

begin

EXEC [BloodBankManagementSystem].[dbo].[sp\_fulltext\_database] @action = 'enable' end

GO

ALTER DATABASE [BloodBankManagementSystem] SET ANSI\_NULL\_DEFAULT OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET ANSI\_NULLS OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET ANSI\_PADDING OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET ANSI\_WARNINGS OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET ARITHABORT OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET AUTO\_CLOSE OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET AUTO\_CREATE\_STATISTICS ON GO

ALTER DATABASE [BloodBankManagementSystem] SET AUTO\_SHRINK OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET AUTO\_UPDATE\_STATISTICS ON GO

ALTER DATABASE [BloodBankManagementSystem] SET CURSOR\_CLOSE\_ON\_COMMIT OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET CURSOR\_DEFAULT GLOBAL GO

ALTER DATABASE [BloodBankManagementSystem] SET CONCAT\_NULL\_YIELDS\_NULL OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET NUMERIC\_ROUNDABORT OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET QUOTED\_IDENTIFIER OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET RECURSIVE\_TRIGGERS OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET DISABLE\_BROKER GO

ALTER DATABASE [BloodBankManagementSystem] SET AUTO\_UPDATE\_STATISTICS\_ASYNC OFF

GO

ALTER DATABASE [BloodBankManagementSystem] SET DATE\_CORRELATION\_OPTIMIZATION OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET TRUSTWORTHY OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET ALLOW\_SNAPSHOT\_ISOLATION OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET PARAMETERIZATION SIMPLE GO

ALTER DATABASE [BloodBankManagementSystem] SET READ\_COMMITTED\_SNAPSHOT OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET HONOR\_BROKER\_PRIORITY OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET RECOVERY FULL GO

ALTER DATABASE [BloodBankManagementSystem] SET MULTI\_USER GO

ALTER DATABASE [BloodBankManagementSystem] SET PAGE\_VERIFY CHECKSUM GO

ALTER DATABASE [BloodBankManagementSystem] SET DB\_CHAINING OFF GO

ALTER DATABASE [BloodBankManagementSystem] SET FILESTREAM( NON\_TRANSACTED\_ACCESS = OFF ) GO

ALTER DATABASE [BloodBankManagementSystem] SET TARGET\_RECOVERY\_TIME = 0 SECONDS GO

EXEC sys.sp\_db\_vardecimal\_storage\_format N'BloodBankManagementSystem', N'ON' GO

USE [BloodBankManagementSystem] GO

/\*\*\*\*\*\* Object: Table [dbo].[donor] Script Date: 28-Feb-20 9:30:25 AM \*\*\*\*\*\*/ SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON GO

CREATE TABLE [dbo].[donor]( [donor\_id] [int] NOT NULL,

[donor\_name] [nvarchar](255) NULL, [blood\_group] [nvarchar](255) NULL, [last\_donation\_date] [nvarchar](255) NULL, [contact] [int] NULL,

[gender] [nvarchar](255) NULL, [address] [nvarchar](255) NULL, [added\_date] [nvarchar](255) NULL, [added\_by] [nvarchar](255) NULL,

CONSTRAINT [PK\_donor] PRIMARY KEY CLUSTERED (

[donor\_id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tbl\_donors] Script Date: 28-Feb-20 9:30:25 AM \*\*\*\*\*\*/ SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON GO

CREATE TABLE [dbo].[tbl\_donors](

[donor\_id] [int] IDENTITY(1,1) NOT NULL, [donor\_name] [nvarchar](50) NULL, [email] [nvarchar](150) NULL,

[contact] [nvarchar](20) NULL, [gender] [nvarchar](10) NULL, [address] [nvarchar](250) NULL, [blood\_group] [nvarchar](10) NULL, [added\_date] [datetime] NULL, [added\_by] [int] NULL,

CONSTRAINT [PK\_tbl\_donor] PRIMARY KEY CLUSTERED (

[donor\_id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[tbl\_users] Script Date: 28-Feb-20 9:30:25 AM \*\*\*\*\*\*/ SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON GO

CREATE TABLE [dbo].[tbl\_users](

[user\_id] [int] IDENTITY(1,1) NOT NULL,

[username] [nvarchar](50) NULL, [email] [nvarchar](150) NULL, [password] [nvarchar](150) NULL, [full\_name] [nvarchar](150) NULL, [contact] [nvarchar](20) NULL, [address] [nvarchar](250) NULL, [added\_date] [datetime] NULL, [image\_name] [nvarchar](50) NULL,

CONSTRAINT [PK\_tbl\_user] PRIMARY KEY CLUSTERED (

[user\_id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (1, N'Muslamul Haq Siam', N'A+', N'2 month ago', 1300234533, N'Male', N'103, Hazi Ismail Road, Khulna.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (2, N'Sukanto', N'AB+', N'3 month ago', 1400234544, N'Male', N'11/b, Hazi Mohasheen Road, Khulna.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (3, N'Md Sakib Hossain', N'A+', N'4 month ago', 1987625255, N'Male', N'112, KDA Avenue, Khulna.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (4, N'Md Abdul', N'AB-', N'2 month ago', 1725652662, N'Male', N'26/a, Bashupara Road, khulna.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (5, N'Rahim shaikh',

N'O+', N'3 month ago', 1772354151, N'Male', N'273, Hazi Ismail Road, Khulna.', N'2020-02- 21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (6, N'Arifull Islam', N'O-', N'3 month ago', 1774545342, N'Male', N'32/c, Boira Cross Road, Khulna.', N'2020- 02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (7, N'Saima Siddika', N'A+', N'1 month ago', 1772334555, N'Female', N'45, Ikbal Nagor Moshjid Road, Khulna.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (8, N'Mostafa Asif', N'A+', N'6 month ago', 1772345123, N'Male', N'31/4, Bashupara lane, Khulna.', N'2020-02- 21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (9, N'Imran Hossain', N'AB+', N'3 month ago', 1332345673, N'Male', N'14, Hazi Ismail Road, khulna.', N'2020-02- 21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (10, N'Sadia Islam', N'A+', N'1 month ago', 1448978364, N'Female', N'19/1, SS College Road, Banorgati, Khulna.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (11, N'Naim Islam', N'AB-', N'1 month ago', 1881233443, N'Male', N'245, Hazi Ismail Road, Khulna.', N'2020- 02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (12, N'Nazmul Hossain', N'A+', N'3 month ago', 1312323335, N'Male', N'12, Farazi Para Lane, Khulna.', N'2020-02- 21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (13, N'Nasim Islam', N'AB+', N'7 month ago', 1998123445, N'Male', N'32, Bashupara More, Khulna.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (14, N'Rofikul Islam', N'A+', N'3 month ago', 1998727733, N'Male', N'110/a, Hazi Meher Ali Road, Khulna.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (15, N'Ripon Mollick', N'O+', N'3 month ago', 1882345233, N'Male', N'16/b, Hazi Mohasheen Road, Khulna.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (16, N'Salim Hosain', N'O-', N'6 month ago', 1887623453, N'Male', N'48/6, Ahsn Ahmed Road, Khulna.', N'2020-02- 21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (17, N'Nasif ali', N'AB- ', N'4 month ago', 1443523445, N'Male', N'69,cometery Road,Dak Bangla,Khulna.', N'2020- 02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (18, N'Riyaj Shaikh', N'A+', N'3 month ago', 1323432987, N'Male', N'Moulvi Para RD,Khulna 1100.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (19, N'Md Masum Sk', N'O+', N'5 month ago', 1923339499, N'Male', N'Shere-Bangla-Road, Power House Mor, Khulna 9100.', N'2020-02-21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (20, N'Youbair Islam', N'A-', N'1 month ago', 1987234645, N'Male', N'81, Boyra Bazar Road, Khulna.', N'2020-02- 21 15:20:00.000', N'tarek023')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (21, N'Karim islam', N'AB+', N'6 month ago', 1771263553, N'Male', N'11/b, Hazi Mohasheen Road, Khulna.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (22, N'Rokib Hassain', N'A+', N'1 month ago', 1342525454, N'Male', N'112, KDA Avenue, Khulna.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (23, N'Babul Khan', N'AB+', N' 4 month ago', 1909875646, N'Male', N'26/a, Bashupara Road, khulna.', N'2020- 02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (24, N'Sumaiya Hassain', N'B+', N'8 month ago', 1323143332, N'Female', N'273, Hazi Ismail Road, Khulna.', N'2020- 02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (25, N'Rani Islam', N'O+', N'2 month ago', 1775434344, N'Female', N'32/c, Boira Cross Road, Khulna.', N'2020- 02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (26, N'Md Nadim', N'A+', N'1 month ago', 1978654363, N'Male', N'45, Ikbal Nagor Moshjid Road, Khulna.', N'2020-02- 21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (27, N'Md Miraz', N'O+', N'7 month ago', 1887654673, N'Male', N'31/4, Bashupara lane, Khulna.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (28, N'Marufa katun', N'B+', N'4 month ago', 1342356743, N'Male', N'14, Hazi Ismail Road, khulna.', N'2020-02- 21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (29, N'Kona Halder', N'AB+', N'2 month ago', 1967543762, N'Female', N'19/1, SS College Road, Banorgati, Khulna.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (30, N'Abdullah', N'O-', N'5 month ago', 1453765286, N'Male', N'245, Hazi Ismail Road, Khulna.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (31, N'Md Billah', N'A+', N'1 month ago', 1876766760, N'Male', N'12, Farazi Para Lane, Khulna.', N'2020-02- 21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (32, N'Md Rubel', N'O+', N'7 month ago', 1765435876, N'Male', N'32, Bashupara More, Khulna.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (33, N'Ali Akbar', N'A- ', N'5 month ago', 1776545448, N'Male', N'110/a, Hazi Meher Ali Road, Khulna.', N'2020- 02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (34, N'Mostafa', N'A-',

N'3 month ago', 1776545449, N'Male', N'16/b, Hazi Mohasheen Road, Khulna.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (35, N'Md Rahim', N'AB+', N'4 month ago', 1654554455, N'Male', N'48/6, Ahsn Ahmed Road, Khulna.', N'2020- 02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (36, N'saddam hossain', N'O+', N'3 month ago', 1923876647, N'Male', N'69,cometery Road,Dak Bangla,Khulna.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (37, N'Aslam Shaikh', N'A+', N'2 month ago', 1887656255, N'Male', N'Moulvi Para RD,Khulna 1100.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (38, N'Billal Hossain', N'AB+', N'1 month ago', 1776355378, N'Male', N'Shere-Bangla-Road, Power House Mor, Khulna 9100.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (39, N'Ariful Islam', N'A+', N'2 month ago', 1712339848, N'Male', N'81, Boyra Bazar Road, Khulna.', N'2020-02- 21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (40, N'Billal Hossain', N'AB+', N'1 month ago', 1880928782, N'Male', N'Moulvi Para RD,Khulna 1100.', N'2020-02-21 15:20:00.000', N'erina054')

INSERT [dbo].[donor] ([donor\_id], [donor\_name], [blood\_group], [last\_donation\_date], [contact], [gender], [address], [added\_date], [added\_by]) VALUES (41, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL)

SET IDENTITY\_INSERT [dbo].[tbl\_users] ON

INSERT [dbo].[tbl\_users] ([user\_id], [username], [email], [password], [full\_name], [contact], [address], [added\_date], [image\_name]) VALUES (1, N'abir004', N'abir69832@gmail.com', N'021478', N'Md. Hasanul Banna Khan Abir', N'01984772121', N'25/2, Hazi Mohshin Road, Khulna.', CAST(0x0000AB6800FCAF80 AS DateTime), N'Blood\_Bank\_MS\_001.jpg')

INSERT [dbo].[tbl\_users] ([user\_id], [username], [email], [password], [full\_name], [contact], [address], [added\_date], [image\_name]) VALUES (2, N'tarek023', N'tarekahmednwu@gmail.com', N'023', N'Tarek Ahmed', N'01400652322', N'Farazaipara Road, Khulna.', CAST(0x0000AB6B0125E260 AS DateTime), N'Blood\_Bank\_MS\_002.jpg')

INSERT [dbo].[tbl\_users] ([user\_id], [username], [email], [password], [full\_name], [contact], [address], [added\_date], [image\_name]) VALUES (3, N'erina054', N'erinarothy100@gmail.com', N'054', N'Erina Momtaz', N'01309188935', N'Ghop, Jessore, Khulna.', CAST(0x0000AB6E013BDB60 AS DateTime), N'Blood\_Bank\_MS\_003.jpg')

SET IDENTITY\_INSERT [dbo].[tbl\_users] OFF USE [master]

GO

ALTER DATABASE [BloodBankManagementSystem] SET READ\_WRITE GO

# Source Code:

* **frmSplash.cs:**

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 BloodBankManagementSystem.UI

{

public partial class frmSplash : Form

{

public frmSplash()

{

InitializeComponent();

}

int move = 0;

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

{

//Write the code to show Loading Animation timerSplash.Interval = 20;

panelMovable.Width += 5;

move += 5;

form

//If the loading is complete then display login form and close this

if(move==640)

{

//Stop the Timer and Close this Form timerSplash.Stop();

this.Hide();

//Display the Login Form frmLogin login = new frmLogin(); login.Show();

}

}

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

{

//Load the Timer timerSplash.Start();

}

}

}

## frmHome.cs:

using BloodBankManagementSystem.DAL; using BloodBankManagementSystem.UI; 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 BloodBankManagementSystem

{

public partial class frmHome : Form

{

public frmHome()

{

InitializeComponent();

}

//Create the Object of Donor Dal donorDAL dal = new donorDAL();

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

{

//Code to Close this Application this.Hide();

}

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

{

//Open Users Form

frmUsers users = new frmUsers(); users.Show();

}

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

{

//Open Manage Donors Form

frmDonors donors = new frmDonors(); donors.Show();

}

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

{

//Load all the Blood Donors Count When Form is Loaded

//Call allDonorCountMethod allDonorCount();

//Display all the Donors DataTable dt = dal.Select(); dgvDonors.DataSource = dt;

//Display the username of Logged In user lblUser.Text = frmLogin.loggedInUser;

}

public void allDonorCount()

{

//Get the Donor Count from DAtabase and SEt in respective label lblOpositiveCount.Text = dal.countDonors("O+"); lblOnegativeCount.Text = dal.countDonors("O-");

lblApositiveCount.Text = dal.countDonors("A+"); lblAnegativeCount.Text = dal.countDonors("A-"); lblBpositiveCount.Text = dal.countDonors("B+"); lblBnegativeCount.Text = dal.countDonors("B-"); lblABpositiveCount.Text = dal.countDonors("AB+"); lblABnegativeCount.Text = dal.countDonors("AB-");

}

private void frmHome\_Activated(object sender, EventArgs e)

{

//Call allDonorCount Method allDonorCount();

}

private void txtSearch\_TextChanged(object sender, EventArgs e)

{

//Get the Keywords from the SEarch TextBox string keywords = txtSearch.Text;

//Check whether the TextBox is Empty or Not if(keywords != null)

{

}

else

{

}

}

//Filter the Donors based on Keywords DataTable dt = dal.Search(keywords); dgvDonors.DataSource = dt;

//DIsplay all the Donors DataTable dt = dal.Select(); dgvDonors.DataSource = dt;

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

{

}

}

}

## frmLogin.cs:

using BloodBankManagementSystem.BLL; using BloodBankManagementSystem.DAL; 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 BloodBankManagementSystem.UI

{

public partial class frmLogin : Form

{

public frmLogin()

{

InitializeComponent();

}

//Create the Object of BLL and DAL loginBLL l = new loginBLL(); loginDAL dal = new loginDAL();

//Create a Static String method to save the username public static string loggedInUser;

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

{

//Write the Code to Cose the Application this.Close();

}

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

{

//Write the Code to Login our Application

//1. Get the username and password from login form l.username = txtUsername.Text;

l.password = txtPassword.Text;

//Check the Login Credentials

bool isSuccess = dal.loginCheck(l);

//Check whehter the login is success or not

//If login is success then isSuccess will be true else it will be false

if(isSuccess==true)

{

//Login Success

//Display Success Message MessageBox.Show("Login Successful.");

//Save the username in loggedInuser Stattic MEthod loggedInUser = l.username;

}

else

{

}

}

}

}

//Display home Form

frmHome home = new frmHome(); home.Show();

this.Hide(); //To CLose Login Form

//Login Failed

//Display the Error Message MessageBox.Show("Login Failed. Try Again.");

## frmUsers.cs:

using BloodBankManagementSystem.BLL; using BloodBankManagementSystem.DAL; using System;

using System.Collections.Generic; using System.ComponentModel; using System.Data;

using System.Drawing; using System.IO; using System.Linq; using System.Text;

using System.Threading.Tasks; using System.Windows.Forms;

namespace BloodBankManagementSystem.UI

{

public partial class frmUsers : Form

{

public frmUsers()

{

InitializeComponent();

}

//Create Objects of userBLL and userDAL userBLL u = new userBLL();

userDAL dal = new userDAL(); string imageName = "no-image.jpg";

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

{

//Add functionality to close this form this.Hide();

}

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

{

//Step 1: Get the Values from UI u.full\_name = txtFullName.Text; u.email = txtEmail.Text; u.username = txtUsername.Text; u.password = txtPassword.Text; u.contact = txtContact.Text; u.address = txtAddress.Text; u.added\_date = DateTime.Now; u.image\_name = imageName;

//Step2: Adding the Values from UI to the Database

//Create a Boolean Variable to check whether the data is inserted successfully or not

bool success = dal.Insert(u);

//Step 3: Check whether the Data is Inserted Successfully or Not if(success==true)

{

//Data or User Added Successfully MessageBox.Show("New User added Successfully.");

//Display the user in DataGrid View DataTable dt = dal.Select(); dgvUsers.DataSource = dt;

}

else

{

}

}

//Clear TextBoxes Clear();

//Failed to Add User MessageBox.Show("Failed to Add New User.");

//Method or Function to Clear TextBoxes public void Clear()

{

txtFullName.Text = ""; txtEmail.Text = ""; txtUsername.Text = ""; txtContact.Text = ""; txtAddress.Text = ""; txtPassword.Text = ""; txtUserID.Text = "";

//Path to Destination Folder

//Get the Image path

string paths = Application.StartupPath.Substring(0, (Application.StartupPath.Length - 10));

string imagePath = paths + "\\images\\no-image.jpg";

//Diplay in Picture Box

pictureBoxProfilePicture.Image = new Bitmap(imagePath);

}

private void dgvUsers\_RowHeaderMouseClick(object sender, DataGridViewCellMouseEventArgs e)

{

//Find the Row Index of the Row Clicked on Users Data Frid View int RowIndex = e.RowIndex;

txtUserID.Text = dgvUsers.Rows[RowIndex].Cells[0].Value.ToString(); txtUsername.Text = dgvUsers.Rows[RowIndex].Cells[1].Value.ToString(); txtEmail.Text = dgvUsers.Rows[RowIndex].Cells[2].Value.ToString(); txtPassword.Text = dgvUsers.Rows[RowIndex].Cells[3].Value.ToString(); txtFullName.Text = dgvUsers.Rows[RowIndex].Cells[4].Value.ToString(); txtContact.Text = dgvUsers.Rows[RowIndex].Cells[5].Value.ToString(); txtAddress.Text = dgvUsers.Rows[RowIndex].Cells[6].Value.ToString(); imageName = dgvUsers.Rows[RowIndex].Cells[8].Value.ToString();

//Display the Image of SElected User

//Get the Image path

string paths = Application.StartupPath.Substring(0, (Application.StartupPath.Length - 10));

if(imageName != "no-image.jpg")

{

}

else

{

}

}

//Path to Destination Folder

string imagePath = paths + "\\images\\" + imageName;

//Diplay in Picture Box

pictureBoxProfilePicture.Image = new Bitmap(imagePath);

//Path to Destination Folder

string imagePath = paths + "\\images\\no-image.jpg";

//Diplay in Picture Box

pictureBoxProfilePicture.Image = new Bitmap(imagePath);

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

{

//Display the Users in DAtagrid View When the Form is Loaded DataTable dt = dal.Select();

dgvUsers.DataSource = dt;

}

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

{

//Step1: Get the Values from UI u.user\_id = int.Parse(txtUserID.Text); u.full\_name = txtFullName.Text; u.email = txtEmail.Text;

u.username = txtUsername.Text; u.password = txtPassword.Text; u.contact = txtContact.Text; u.address = txtAddress.Text; u.added\_date = DateTime.Now; u.image\_name = imageName;

//Step 2: Create a Boolean variable to check whether the data is updated successfully or not

bool success = dal.Update(u);

//Let's check whether the data is updated or not if(success==true)

{

//Data Udated Successfully MessageBox.Show("User Updated Successfully.");

//Refresh DAta Grid View DataTable dt = dal.Select(); dgvUsers.DataSource = dt;

//Clear the TextBoxes Clear();

}

}

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

{

//Step 1: Get the UserID from Text Box to Delete the User u.user\_id = int.Parse(txtUserID.Text);

not

//Step Create the Boolean value to check whether the user deleted or bool success = dal.Delete(u);

//Let's check whteher the user is Deleted or Not if(success==true)

{

//User Deleted Successfully MessageBox.Show("User Deleted Successfully.");

//Refresh DataGrid View DataTable dt = dal.Select(); dgvUsers.DataSource = dt;

//Clear the TextBoxes Clear();

}

}

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

{

//Call the user Function Clear();

}

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

{

//Write the Code to Upload the Image of User

//Open Dialog Box t Select Image OpenFileDialog open = new OpenFileDialog();

//Filter the File Type, Only Allow Image File Types open.Filter = "Image Files (\*.jpg; \*.jpeg; \*.png; \*.PNG;

\*.gif;)|\*.jpg; \*.jpeg; \*.png; \*.PNG; \*.gif;";

//Check if the file is selected or Not if(open.ShowDialog()==DialogResult.OK)

{

//Check if the file exists or not if(open.CheckFileExists)

{

//DIsplay the Selected File on Picture Box pictureBoxProfilePicture.Image = new Bitmap(open.FileName);

//Rename the Image we selected

//1. Get the Extension of Image

string ext = Path.GetExtension(open.FileName);

//2. Generate Random Integer Random random = new Random();

int RandInt = random.Next(0, 1000);

//3. REname the Image

imageName = "Blood\_Bank\_MS\_" + RandInt + ext;

//4. Get the path of SElected Image string sourcePath = open.FileName;

//5. Get the Path of Destination

string paths = Application.StartupPath.Substring(0, Application.StartupPath.Length - 10);

//Paths to Destination Folder

string destinationPath = paths + "\\images\\" + imageName;

//6. Copy image to the Destination Folder File.Copy(sourcePath, destinationPath);

//7. Display Message

MessageBox.Show("Image Successfully Uploaded.");

}

}

}

private void txtSearch\_TextChanged(object sender, EventArgs e)

{

//Write the Code to Get the users BAsed on Keywords

//1. Get the Keywords from the TextBox String keywords = txtSearch.Text;

//Check whether the textbox is empty or not if(keywords!=null)

{

the keywords

}

}

else

{

}

//TextBox is not empty, display users on DAta Grid View based on

DataTable dt = dal.Search(keywords); dgvUsers.DataSource = dt;

//TExtbox is Empty and display all the users on DAta Grid View DataTable dt = dal.Select();

dgvUsers.DataSource = dt;

}

}

## frmDonors.cs:

dgvUsers.DataSource = dt; using BloodBankManagementSystem.BLL; using BloodBankManagementSystem.DAL;

using System;

using System.Collections.Generic; using System.ComponentModel; using System.Data;

using System.Drawing; using System.IO; using System.Linq; using System.Text;

using System.Threading.Tasks; using System.Windows.Forms;

namespace BloodBankManagementSystem.UI

{

public partial class frmDonors : Form

{

public frmDonors()

{

InitializeComponent();

}

//Create object of Donor BLL and Donor DAL donorBLL d = new donorBLL();

donorDAL dal = new donorDAL(); userDAL udal = new userDAL();

//Global Variable for Image

string imageName = "no-image.jpg";

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

{

//Display Donors in DataGrid View DataTable dt = dal.Select(); dgvDonors.DataSource = dt;

//First we need to get the image Path

string path = Application.StartupPath.Substring(0, (Application.StartupPath.Length) - 10);

string imagepath = path + "\\images\\no-image.jpg";

//Display Image in PictureBox pictureBoxProfilePicture.Image = new Bitmap(imagepath);

}

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

{

//Close this form this.Hide();

}

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

{

//We will write the code to Add new Donor

//Step 1. Get the DAta from Manage Donors Form d.donor\_name = txtFirstName.Text; d.last\_donation\_date = txtEmail.Text; d.gender = cmbGender.Text;

d.blood\_group = cmbBloodGroup.Text; d.contact = txtContact.Text; d.address = txtAddress.Text; d.added\_date = DateTime.Now;

//Get The ID of Logged In USer

string loggedInUser = frmLogin.loggedInUser; userBLL usr = udal.GetIDFromUsername(loggedInUser);

d.added\_by = usr.user\_id; d.image\_name = imageName;

//Step2: Inserting Data into Database

//Create a Boolean Variable to Isnert DAta into DAtabase and check whether the data inserted successfully of not

bool isSuccess = dal.Insert(d);

//if the Data is inserted successfully then the values of isSuccess will be True else it will be false

if(isSuccess==true)

{

//Data Inserted Successfully MessageBox.Show("New Donor Added Successfully");

//Refresh Datagrid View DataTable dt = dal.Select(); dgvDonors.DataSource = dt;

}

else

{

}

}

//Clear all the Textboxes Clear();

//FAiled to Insert Data MessageBox.Show("Failed to Add new Donor.");

//Create a Method to Clear all the Textboxes public void Clear()

{

//Clear all the TExtboxes txtFirstName.Text = ""; txtEmail.Text = ""; txtDonorID.Text = ""; cmbGender.Text = ""; cmbBloodGroup.Text = ""; txtContact.Text = ""; txtAddress.Text = "";

//Clear the PictureBox

//First we need to get the image Path

string path = Application.StartupPath.Substring(0, (Application.StartupPath.Length) - 10);

string imagepath = path + "\\images\\no-image.jpg";

//Display Image in PictureBox pictureBoxProfilePicture.Image = new Bitmap(imagepath);

}

private void dgvDonors\_RowHeaderMouseClick(object sender, DataGridViewCellMouseEventArgs e)

{

//SElect the DAta from DAtagrid View and Display in our Form

//1. Find the Row Selected int RowIndex = e.RowIndex;

txtDonorID.Text = dgvDonors.Rows[RowIndex].Cells[0].Value.ToString(); txtFirstName.Text =

dgvDonors.Rows[RowIndex].Cells[1].Value.ToString();

txtEmail.Text = dgvDonors.Rows[RowIndex].Cells[3].Value.ToString();

txtContact.Text = dgvDonors.Rows[RowIndex].Cells[4].Value.ToString(); cmbGender.Text = dgvDonors.Rows[RowIndex].Cells[5].Value.ToString(); txtAddress.Text = dgvDonors.Rows[RowIndex].Cells[6].Value.ToString(); cmbBloodGroup.Text =

dgvDonors.Rows[RowIndex].Cells[7].Value.ToString();

/\*imageName = dgvDonors.Rows[RowIndex].Cells[9].Value.ToString();

//Display The image of Selected Donor

//Get the image path

string paths = Application.StartupPath.Substring(0, (Application.StartupPath.Length) - 10);

string imagePath = paths + "\\images\\" + imageName;

//Display the Image of SElected User pictureBoxProfilePicture.Image = new Bitmap(imagePath);\*/

}

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

{

//Add the Functionality to Update the Donors

//1. Get the Values from Form d.donor\_id = int.Parse(txtDonorID.Text); d.donor\_name = txtFirstName.Text; d.last\_donation\_date = txtEmail.Text; d.gender = cmbGender.Text;

d.blood\_group = cmbBloodGroup.Text; d.contact = txtContact.Text; d.address = txtAddress.Text;

//Get The ID of Logged In USer

string loggedInUser = frmLogin.loggedInUser; userBLL usr = udal.GetIDFromUsername(loggedInUser);

d.added\_by = usr.user\_id; d.image\_name = imageName;

//Create a Boolean Variable to Check whether the data updated successfully or not

bool isSuccess = dal.Update(d);

//If the data updated successfully then the value of isSuccess will be true else it will be false

if(isSuccess == true)

{

//Donor Updated Successfully MessageBox.Show("Donor updated Successfully."); Clear();

//Refresh Datagridview DataTable dt = dal.Select(); dgvDonors.DataSource = dt;

}

else

{

}

}

//Failed to Update

MessageBox.Show("Failed to update donors.");

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

{

//Get the value from form

d.donor\_id = int.Parse(txtDonorID.Text);

//Create a Boolean Variable to Check whether the donor deleted or not bool isSuccess = dal.Delete(d);

if(isSuccess==true)

{

//Donor Deleted Successfully MessageBox.Show("Donor Deleted Successfully.");

Clear();

}

else

{

}

}

//Refresh Datagrid View DataTable dt = dal.Select(); dgvDonors.DataSource = dt;

//Failed to Delete Donor MessageBox.Show("Failed to Delete Donor");

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

{

//Clear the TExtboxes Clear();

}

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

{

//Code to Select Image and Upload

//1. Open the Dialog Box to SElect Image OpenFileDialog open = new OpenFileDialog();

//2. FIlter the File Type (allow only image files)

open.Filter = "Image Files Only (\*.jpg; \*.jpeg; \*.png; \*.gif| \*.jpg;

\*.jpeg; \*.png; \*.gif)";

//3. Check whether the image is selected or not if(open.ShowDialog()==DialogResult.OK)

{

//Check if the file exists or not if(open.CheckFileExists)

{

//Display the Selected Image in PictureBox pictureBoxProfilePicture.Image = new Bitmap(open.FileName);

//Rename the selected image

//1. Get the Extension of SElected Image string ext = Path.GetExtension(open.FileName);

string name = Path.GetFileNameWithoutExtension(open.FileName);

//Generate Random but Globally Unique Identifier

Guid g = new Guid(); g = Guid.NewGuid();

//Finally REname our Image

imageName = "Blood\_Bank\_MS\_"+ name + g + ext;

//Get the Source Path (Path of Image) string sourcePath = open.FileName;

//Get the Destination Path

string paths = Application.StartupPath.Substring(0, Application.StartupPath.Length - 10);

//Path to Desitnation

string destinationPath = paths + "\\images\\" + imageName;

//Upload the Image to Destination Folder File.Copy(sourcePath, destinationPath);

//Display Message after the image is uploaded successfully MessageBox.Show("Image successfully uploaded.");

}

}

}

private void txtSearch\_TextChanged(object sender, EventArgs e)

{

//Let's Add the Dunctionality to Search the Donors

//1. Get the Keywords Typed on the Search TExt Box string keywords = txtSearch.Text;

// Check Whether the Search TExtBox is Empty or Not if(keywords != null)

{

}

else

{

}

}

//Display the information of Donors Based on Keywords DataTable dt = dal.Search(keywords); dgvDonors.DataSource = dt;

//DIsplay all the Donors DataTable dt = dal.Select(); dgvDonors.DataSource = dt;

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

{

}

}

}

# Conclusion:

A Blood Bank is a software product suite designed to improve the quality and management of bloodbank and hospital health care management in the areas of health process analysis and activity-based costing. Blood Bank Manager enables to develop organization and improve its effectiveness and quality of work, Managing the key processes efficiently is critical to the success of the pharmacy.

Blood Bank Manager helps manage processes. A Blood Bank Manager provides all process management tool elements:

Modeling, analysis, and simulation. Documentation though an important part of a blood bank management, is a nonproductive exercise for the intellectual human being, whose ability lies in core areas of excellence. Hence a systematic approach to the way documents are managed,

Blood Bank Management System is an efficient project. Effective analysis of data can help a lot in medical field as many other objects and fields can be added to this system for the different blood groups.