|  |
| --- |
| Java Final Project |
| Blood Donation Management System. API Documentation |
| 17-Dec-2015 |

|  |
| --- |
| Submitted To Submitted By  Sabbir Ahmed Shubho, Seemanta Ahmed  Faculty, CS Dept. AIUB Bhuiyan, Md Mahbub Islam  Banani, Dhaka. Hasnat, Md Waliul  Section: G |

# Introduction:

This is a Blood Donation blood donation management system , it provides a Query interface and Donor login and administrator login

# Principals

The Principle of this application is too provide a simple interface to the client (Blood Bank) to operate their daily activities a bit swiftly

# Source Files Given by Soft Copy Package

Two packages has been used in this app, one is default package another db.table containing DbTable class

This program includes the following packages.

**import**javafx.application.Application;

**import**javafx.event.EventHandler;

**import**javafx.geometry.Pos;

**import**javafx.scene.Cursor;

**import**javafx.scene.Scene;

**import**javafx.scene.control.Button;

**import**javafx.scene.control.ChoiceBox;

**import**javafx.scene.control.Tooltip;

**import**javafx.scene.image.Image;

**import**javafx.scene.image.ImageView;

**import**javafx.scene.input.MouseEvent;

**import**javafx.scene.layout.HBox;

**import**javafx.scene.layout.VBox;

**import**javafx.stage.Stage;

**import**java.util.HashMap;

**import**java.util.Map;

**import**java.sql.ResultSet;

**import**java.sql.SQLException;

**import**java.util.ArrayList;

**import**java.sql.ResultSet;

**import**java.sql.SQLException;

**import**java.sql.ResultSet;

**import**java.sql.SQLException;

**import**javax.swing.\*;

**import**java.sql.Connection;

**import**java.sql.DriverManager;

**import**java.sql.ResultSet;

**import**java.sql.Statement;

# Class

In this Program we have the following classes.

* Main
* PopUpWindow
* Query
* Layout
* InltButtons
* DonorPanel
* ConnectorModified
* ButtonEffect
* BloodBankAdminPanel

Methods

The methods included in the classes are shown below.

1. Main

* configureChoicekbox(); // it configures the check box
* public static void cout(); // prints in console
* configure\_Buttons(); // it configures the Button

1. PopUpWindow

* popUp(); // this creates a popup window
* error\_popUp(); // gives an error alert if there is an error
* initButtons(); // initializes Buttons
* LayoutConfiguration(); // Configures the layout of the stages

1. Query

* queryWindow(); // this creates the query window
* getQueryLayout(); // returns Layout as a BorderPane object
* initBackButtonToMainWindow(); // Initializes a “Back to Home ” Button
* getSqlQurey(); // Generates Sql Query
* buttonEffect(); // handles and gives effects to Buttons

1. InltButtons

* initButtons(); // Initializes Buttons

1. DonorPanel

* donorControlPanel(); // gives an interface to the donor
* initComponents(); // initializes the components in DonorPanel
* setDefaultValue(); // sets a default value to the Sql search
* checkInputs(); // checks the inputs of the donor

1. ConnectorModified

* getTableData(); // generates the table data ofResultSet
* theQuery(String query); // executes the Sql commands and updates the database
* getSingleRow(String sql); //

1. ButtonEffect

There are no methods in this class. This class only initializes some Effect Objects.

1. BloodBankAdminPanel

* initButtons(); // this method initializes Buttons for the Admin Panel
* getValues(); // This method gets the results for the Sql search
* getSceneBuild(); // creates a scene for the admin panel
* getAdminPanelLayout(); //generates a layout for the Admin Panel
* BankControlPanel(); // Creates a new window for the Bank Management.

Conclusion: The reusability and efficiency of this program can be maximized, Maintainability is quite easy to handle.