1. **INTRODUCTION**

Today the Computer is in almost every field of life. The Computers have made the life very easy fast and comfortable. So the reason we have developed project on blood bank management system, to manage the details of Blood, Donors, Blood Group, Blood Banks and Stock. It manages all the information. Blood Bank Management System (BBMS) is a browser based system 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 group available in each blood bank and help them manage in better way .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 effective...

**1.1 Features of the System :**

\* Blood Donation Camp and Camp Organizer Management.

\* Donor Management - Donor Registration, Managing donor database, recording their

physical and medical statistics.

\* Inventory management in blood bank for storage and issuance of blood.

\* Blood requisition and issuance of blood.

\* Online transfer of blood from one blood bank to another.

**2. SYSTEM DEVELOPMENT LIFE CYCLE**

**2.1 System/Information Engineering and Modeling :**

As software is always of a large system (or business), work begins by establishing the requirements for all system elements and then allocating some subset of these requirements to software. This system view is essential when the software must interface with other elements such as hardware, people and other resources. System is the basic and very critical requirement for the existence of software in any entity. So if the system is not in place, the system should be engineered and put in place. In some cases, to extract the maximum output, the system should be re-engineered and spruced up. Once the ideal system is engineered or tuned, the development team studies the software requirement for the system.

**2.2 Software Requirement Analysis :**

This process is also known as feasibility study. In this phase, the development team visits the customer and studies their system. They investigate the need for possible software automation in the given system. By the end of the feasibility study, the team furnishes a document that holds the different specific recommendations for the candidate system. It also includes the personnel assignments, costs, project schedule, target dates etc.... The requirement gathering process is intensified and focused specially on software. To understand the nature of the program(s) to be built, the system engineer or "Analyst" must understand the information domain for the software, as well as required function, behavior, performance and interfacing. The essential purpose of this phase is to find the need and to define the problem that needs to be solved.

**3. SYSTEM ANALAYSIS**

**3.1 Identification Of The Needs :**

The Management of the blood is an important factors which will save the life of persons, in earlier days the management of blood bank is done manually, if any persons ask whether such blood is available then blood bank managing official has to search the records and say whether it is available or not. The information related to blood bank providing to the patients is a very tedious task, to overcome such difficulties one automated software which will manage the data related to blood bank and gives the information easily and quickly to cater to the needs of large patients and hospitals, given rise to develop this software.

**Need of Computerization :**

Blood is very important commodity to save the life pouf process. The important blood has to be maintained in efficient, easily and proving accurate information, the blood bank system has to be computerized.

Easy to Use

* Provide More Information about donors
* Quick access to information
* More accurate, reliable information

**Current System :**

The present system is certainly siring the cause of humanity, but it is in a sorry state of affairs. There are several shortcoming associated with the present system. Some of the shortcomings are listed below.

* The transaction viz. blood donation and receipt are very laborious and tedious.
* The entire process is highly time consuming
* There is no proper maintenance of records.
* Data Reports may not be accurate.
* Availability of blood of a particular type may not be confirmed.
* The system, basically is not Transparent.

The present system is certainly good, but it can be made better with the help of proper database maintenance. The following new features can be incorporated in the new system to make it more efficient.

* The system should be automated with the help of proper back end front end tools.
* Data reports must be generated automatically.

In the new system, SQL server 2019 has been used as the back end tool. Microsoft Visual Studio 2019 was the obvious choice for the development of the front end.

**3.2 Preliminary Investigation :**

Preliminary investigation is a process of extracting the needs of system and what the system must do to satisfy the user requirements. The goal of the analysis is to understand the domain of the problem and the system responsibilities by understanding, how the user can use system elements must be elected, analyzed, specified, software engineering which involves the system analysis and design implication

Studied in details of the Donors, recipient, Doctors and Hospital data required in a blood bank. If the blood bank is of large then multiple clients can be used with a server.

**4. FEASIBILITY STUDY**

The feasibility was carried out determine is it possible to develop the system with the existing environment. There are three aspects

* Operational feasibility
* Technical feasibility
* Economic feasibility

**4.1 Operational Feasibility :**

The operation of the proposed system is computerizing the manual work of maintaining blood bank related data such as Donor, Recipient, and Hospitals. The earlier system user will maintain the same data with register, the same process is computerized and users have to be given little bit training in operating computers. Since, the proposed software is more user friendlier and he has to enter very less information and can provide quickly information to the persons who are in need of blood such as Hospitals and Recipients this system may be brought into operation with little bit of training to staff who handles the blood bank related information.

**4.2 Technical Feasibility :**

Development this software with the help of Visual studio 2010 as front end and SQL server as back end. The earlier technology was manual system. The school requires hardware, UPS and furniture’s and installation of the proposed software only once, and then it will run smoothly.

**4.3 Economic Feasibility :**

Economic feasibility is an evaluation of development cost weighted against the ultimate income or benefit derived from the proposed system. So it is cost benefit analysis. As the proposed system Blood Bank Management System is developed with less expected investment and with better information quantity, quality and timeliness, it is economic feasibility

**5. PROJECT PLANNING**

**5.1 Project Planning :**

The project planned in concern with important steps involved a standard project life cycle. The risk management is studied in depth and hence planned for a special proof of concept is designed to test the individual system technological supports and their architecture. With the clarification document of proof of concept the total plan of the project is built on papers to eradicate logical errors.

**5.2 Project Scheduling :**

The scheduling of the project is designed in such a way that individual modules will be tested with different testing techniques and also interdependent modules. Per day minimum of 4 hours is required to complete all the modules in three months of time for implementation.

**6. SOFTWARE REQUIREMENT SPECIFICATION ( SRS )**

Hardware Specifications :

This system has been developed on the following hardware configuration.

Process : i5 processer

RAM : 8 GB

HDD : 2 TB

Software Specification :

This system is developed using the following software

Operating System : win 10

Front End : Visual Studio 2019

Back End : SQL server 2019

**6.1 Software Engineering Para Diagram Applied :** SQL server

Using SQL server we can manage all information within single database file. Within the file, we can divide data in separate storage container called tables.

**Features of Server** :

* Working on the internet or internet
* Working on other applications
* Working with sample application
* Working in the database window
* Using and customizing toolbars, menu bars and shortcut menus.
* Working with data
* Working with filters
* Printing and previewing.
* Designing a database.
* Creating and customizing tables.
* Creating and customizing query.
* Creating and customizing forms and reports.
* Working in the module window, debug window and object browser.
* Programming with visual studio.
* Securing and administering a database.

**6.2 DATABASE OBJECTS :**

A database object is any thing defined and stored in a database. The following is list of database objects like Tables, Views, Indexes, Synonyms and Database links, Roles, Snapshots and Users.

TABLES

The basic unit of data in a SQL server All data in a SQL server is stored in tables. Every table has a table – name and a set of columns and rows in which the data stored. Each columns in different tables contain the same information. In this way, the tables can refer to one another.

VIEWS

A view is a customized presentation of the data from one or more tables views derive their data from the tables, on which they are based which are knows as base tables. All operations performed on a view for several purposes.

* To get an additional level of tables security restricting accesses to a pre – determined set of table rows and columns.
* To hide the data complexity. SQL Server database usually include many tables, and by creating a view combining information from two or more tables, make it easier for other users to access information from the database.
* To present the data in a different perspective from that of the base tables. Views provide a means to rename columns without affecting the base tables.
* To store complex queries.

**6.3 Code Efficiency :**

Code efficiency is the amount of computing resource and code required by a program to perform its function. The following considerations have brought our code efficient we have used very limited coding in our project. It will not take more time for exception.

1. Reliability : The project is very reliable as it gives a consistent result without any failure for a specific time period.
2. Functionality : The project functions very efficiently.
3. Flexibility : This project adapts to the changing needs of the customer. It is highly flexible.
4. Usability : This project support very nice GUI, hence it is user friendly.
5. Security : We have used security mechanism.
6. Documentation : It has a good documentation. It contains detailed study about the project or website. It contains detailed information about no of modules and the information present in each module. It contains information about how the web pages are interlinked with each other. ( by railway diagrams, ER diagrams etc. )
7. Performance : Delivers the result as expected.

Has the code written in the project “ Blood Bank Management System “ [BBMS] satisfies all the above features. We can conclude that the code written is very efficient.

**6.4 Data Integrity And Constraints :**

**Registration:**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr no | Field name | Data type | Description |
| 1 | Name | nvarchar(50) | Primary key |
| 2 | Email | nvarchar(50) |  |
| 3 | Mobile | nvarchar(50) |  |
| 4 | gender | nvarchar(50) |  |
| 5 | age | nvarchar(50) |  |
| 6 | bg | nvarchar(50) |  |
| 7 | weight | nvarchar(50) |  |
| 8 | height | nvarchar(50) |  |
| 9 | image | nvarchar(50) |  |
| 10 | password | nvarchar(50) |  |
| 11 | address | nvarchar(50) |  |

**Donor Registration :**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr no | Field name | Data type | Description |
| 1 | Name | nvarchar(50) |  |
| 2 | Email | nvarchar(50) |  |
| 3 | Mobile | nvarchar(50) |  |
| 4 | bg | nvarchar(50) |  |
| 5 | image | nvarchar(50) |  |
| 6 | address | nvarchar(50) |  |

**Blood Stock :**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr no | Field name | Data type | Description |
| 1 | **Id** | nvarchar(50) | Primary key |
| 2 | **Ap** | nvarchar(50) |  |
| 3 | An | nvarchar(50) |  |
| 4 | Bp | nvarchar(50) |  |
| 5 | Bn | nvarchar(50) |  |
| 6 | Op | nvarchar(50) |  |
| 7 | On | nvarchar(50) |  |
| 8 | ABp | nvarchar(50) |  |
| 9 | ABn | nvarchar(50) |  |

**Blood Request :**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr no | Field name | Data type | Description |
| 1 | Name | nvarchar(50) | Primary key |
| 2 | Email | nvarchar(50) |  |
| 3 | Mobile | nvarchar(50) |  |
| 4 | Blood\_Group | nvarchar(50) |  |
| 5 | Address | nvarchar(50) |  |

**7. E-R SYMBOLS AND MEANING**

The overall logical structure of the database can be expressed graphically by an E-R diagram shows the entities, their attributes, relationship and cardinality among the entities. Following are some of the notations used in E-R diagram

|  |  |  |
| --- | --- | --- |
| **Sl. No** | **Symbol** | **Meaning** |
| 1 |  | Entity |
| 2 |  | Attribute |
| 3 |  | Key Attribute |
| 4 |  | Relationship |
| 5 |  | Relations |
| 6 |  | **Weak Entity** |
| 7 |  | Multi Valued Attribute |
| 8 |  | Identifying Relationship |
| 9 |  | Composite Attribute |
| 10 |  | Derived Attribute |

**7.1 Entity Relationship Diagram :**

Name Email Mobile

gender

age bg

Donor

weight height

password

image address

Login

Adminlogin Donorlogin

Username Password Username Password

**8. IMPLEMENTATION**

Implementation is the stage where the theoretical design is turned into a working system. Once the design is complete, most of the major decisions about the system have been made. The goal of coding phase is to translate the design of the system into code in a given programming language. For a given design, the aim in this phase is to implement the design in the best possible manner.

The coding phase affects both exam and maintenance profoundly. Since the exam and maintenance costs of software are much higher than the coding cost, the goal of the coding should be to reduce the exam and maintenance effort. Hence, during coding the focus should be on developing the programs that are easy to read and understand, and not simply on developing programs that are easy to write.

## Introduction to Visual Studio.NET

Microsoft.Net is a very exciting and major technology change. While it builds on the concepts introduced by the COM. It is quite different from anything we today promising to make us more productive as it merges Windows GUI and browser- based Internet development environment closer to together.

Visual Studio.Net will increase productivity, and effectively address the demands of the rapidly changing business Environment. The .NET development platform has made it easier for programmers using different languages to quickly create robust Internet Applications.

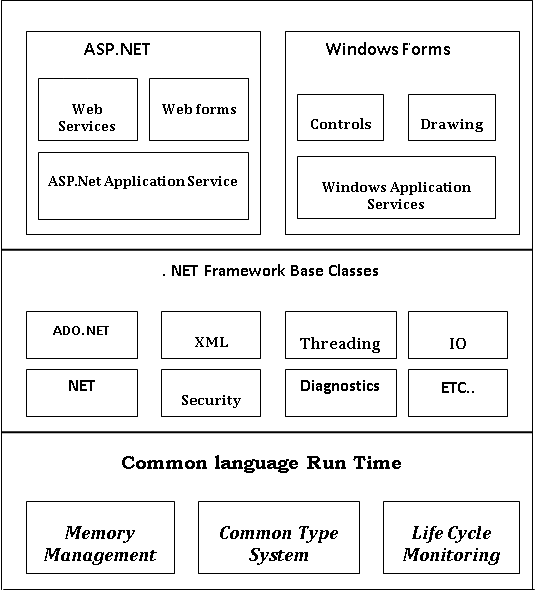
* **Overview of .NET Framework**

The Operating System is actually wrapped by the .NET Framework, insulating Software developed with .NET from Operating System specifies such as file handling and memory allocations.

**8.1 Components of .NET Framework :**

The .NET Framework consist of Common Language Runtime (CLR) .NET Framework Class Libraries, known as Base Class Libraries (BCL). The CLR is a virtual machine in which all .NET applications and Languages run. NET Framework Class Libraries include support for everything from file I/O and database I/O to XML and SOAP.

The CLR is at the base is considered as the heart of the .NET Framework. It is the engine that drives key functionalities, which include Cross Languages inheritance, allocation & management of memory, Reference tracking for object and handle garbage collection. The next generations of standard system service such as ADO.NET & XML are included in the middle layer. These services are brought under control of the framework, making them universally available and making their usage consistence across languages. The Major Components of the Microsoft .NET Framework is shown in the following diagram.



The above .NET Architecture – Three layers at the base is the Common Language Runtime, often abbreviated to CLR. This is the heart of the .NET Framework, the engine that drives key functionality. It includes, for example, a common system of data types. These common types, plus a standard interface convention, make cross-language inheritance possible. In addition to allocation and management of memory, the CLR also does reference counting for objects, and handles garbage collection.

The middle layer includes the next generation of standard system services such as ADO.NET and XML. These services are brought under the control of the framework making them universally available and standardizing their usage across languages.

The top layer includes user programs and interfaces. Windows forms (often informally referred, as Win Forms are a new way to create standard Win 32 applications, based on Windows Foundation Classes (WFC). Web Forms provide powerful form base UI for Internet applications. The Web Services, which are perhaps the most revolutionary mechanism for the programs to communicate over the Internet.

* **. NET Frame Work Class Library**

The .NET Framework class library is a collection of reusable classes, or types that tightly integrates the CLR. Class library builds on the Object-Oriented nature of the runtime, providing types from which your own managed code can derive functionality. This not only makes the .NET framework types easy to use, but also reduces the learning curve associated with using a new piece of code. In addition Third party component can integrate seamlessly with classes in the .NET framework.

**8.2 C# :**

The C# is a modern language derived from the C and C++. It simplifies and modernizes C++ in the areas of Classes, Namespaces, Method overloading and exception handling. Much of the complexity of C++ is removed from C# to make it easier to use and less error prone. C# eliminates certain features of C++ like macros, templates and multiple inheritances. New features added to C# are Strict Type casting, Versioning, Garbage collection, and many more.

C# is an integral part of the entire .NET framework. .NET represents not only a revolution in facilities available for general-purpose Windows programming, but it will also be the immediate environment seen in .NET code (known as managed code) when it runs.

C# is a genuine object-oriented programming language. C# (or more correctly .NET also provided these features) gives you the best of both feature of C++ and VB. It fully supports inheritance, but also of the .NET framework insures that enough information is included in the compiled library files (the assemblies) that your classes can be inherited from and used by other .NET-aware code without requiring access to your source files. To this extent .NET arguably should provide real cross-language code reuse using true object-oriented principles for the first time in the history of programming.

**Features of C#**

* Simple
* Modern
* Object Oriented
* Type Safe
* Version able
* Compatible

**8.3 ASP.NET :**

Active Server Pages is a new technology from Microsoft than enables web server to process application logic & then deliver standard HTML to the client browser. The biggest benefits of ASP.NET over ASP is that it has a language independent way of creating components and dynamic web applications that can produce out put any platform or devise. Everybody will be able to access object, properties and methods no matter what programming language we use. ASP.NET is largely syntax compatible with ASP, it also provides a new programming model and infrastructure for more secure, scalable, and stable applications.ASP.NET is a compiled, NET-based environment; you can author applications in any .NET compatible language, including Visual Basic .NET, C#, and Scripts .NET.

Additionally, the entire .NET Framework is available to any ASP.NET application. Developers can easily access the benefits of these technologies, which include the managed common language runtime environment, type safety, inheritance, and so on.

ASP.NET provides a simple model that enables Web developers to write logic that runs at the application level. Developers can write this code in the global. Sax text files or in a compiled class deployed as an assembly. This logic can include application-level events, but developers can easily extend this model to suit the needs of their Web application.

**ASP.NET Also has Following Advantages:**

ASP.NET reduces the amount of code that is required to write an Application

Compiled code is converted in to classes

Strong typing is followed

Use of the Components made easy.

**The Key Design Goals of the .NET Framework:**

* Simpler and faster development.
* Automatic handling of memory management.
* Good tool support.
* Scalability.
* Huge base class hierarchy, which is common to all languages.

**8.4 Microsoft Visual Studio 2019 :**

Visual Studio is an **Integrated Development Environment(IDE)** developed by Microsoft to develop GUI(Graphical User Interface), console, Web applications, web apps, mobile apps, cloud, and web services, etc. With the help of this IDE, you can create managed code as well as native code. It uses the various platforms of Microsoft software development software like Windows store, Microsoft Silverlight, and Windows API, etc. It is not a language-specific IDE as you can use this to write code in C#, C++, VB(Visual Basic), Python, JavaScript, and many more languages. It provides support for 36 different programming languages. It is available for Windows as well as for macOS. programming aims at providing the user with an interface that is intuitive and easy – to – use. While developing such an interface, the programmer employs user – friendly features such as windows, menus, buttons and list programmer employs user – friendly features such as windows, means, buttons and list boxes. A visual programming environment provides all features that are required to develop a graphical users interface as ready – to – use commonly required user friendly features each time around.

When the programmer needs a specific user interface features such as a button, he selects the appropriate ready – to – use components provided by the visual programming environment. These components can be moved, resized and renamed as required.

A visual programming environment automates the process of creating a user interface and also provides a means of associating code with each components. Microsoft visual studio, version 2019 contains visual programming tools, which is called Microsoft Visual Studio 2019.

**9. SYSTEM DESIGN**

**9.1 Modularization Details :**

In structured design a program is segmented into small independent modules. These are arranged in hierarchy that approximates a module of the school era and is organized in a top down manner in structured design we try to minimize the complexity of the problem and make it manageable by sub – dividing it into smaller segments which is called module.

* Home page
* Login page
* Registration page
* Admin page
* Admin Blood Donation Request page
* Admin Blood Stock Update page
* Blood Bank Stock page
* Blood Request page
* Contact Us page
* Index page
* Master Page
* Master Page 2
* Plz Login page
* Req Blood Donation page
* Search Blood Donor page

**Index Page :**

This is the Index page for the blood bank Management System where it has got the connectivity to different phases which are useful in terms information about the keeping records of the recipient, donors and general information it connected with Master Page or Log In Page.

**Home Page :**

This is the home page for the blood bank Management System where it has got the connectivity to different phases which are useful in terms information about the keeping records of the recipient, donors and general information it connected with Master Page 2 or Log Out Page.

**Registration Page :**

Here any person can register and become member of Blood Bank Management System But not a Blood Donor. He can have Password.

* Name
* Email
* Mobile
* Gender
* Age
* Blood Group
* Weight
* Height
* Image
* Password
* Address

**Blood Bank Stock Page :**

Blood Stock Availablity. Get the complete blood stocks on each blood group.The blood bank details with blood stocks on each blood group will be displayed on search results. Here's where you can find out exactly how much blood we've got in our blood banks at the moment.

**Blood Request :**

Here the member of Blood Bank Donation can request blood.By giving

* Name
* Email
* Mobile
* Blood Group
* Address

**Contact Us :**

Here you can get Developer Details there Name, Mail ID and Mobile Number.

**Login :**

Here member of Blood Bank Donation can login

* User Name
* Password

**Registration :**

Here User can register and can become member

* Name
* Email
* Mobile
* Gender
* Age
* Blood Group
* Weight
* Height
* Image
* Password
* Address

**Search Blood Donor :**

Here we can see Blood Donor who registerd in Request for Blood Donatation

* Name
* Email
* Mobile
* Blood Group
* Image
* Address

**Admin Page :**

In Admin Page s are those who provide support to a company. This support might include Update Blood Bank Stock, See Blood Request and See Blood Request Blood Donation.

**Admin Blood Stock Update Page :**

Here Only Admin can Change\Update the Blood Stock on daily basis.

**Admin Blood Donation Request Page :**

Here Only Admin can See Blood Donor who registered on Request for Blood Donation.

**9.2 Database Design :**

Good design is the key to effective engineering. The design phase included the following three stages.

* Studying and Understanding the problems
* Identifying the features of at least one possible solution
* Describing each abstraction used in the solution

The problem was examined from a number of viewpoints. This provided different insights into the design requirements.

It is often useful to identity a number of solutions and evaluate them all. Here the different solutions that could be provided were identified and evaluated. But the gross features of one particular solution, which was implemented, were identified.

There errors and omissions in the high level design were discovered and corrected.

**Normalization :**

Normalization technique is a bottom up design that is a more purist approach and views relational database schema design strictly in terms of functional and other types of dependencies specified on the database attributes. After the database designer specifies the dependencies, a normalization algorithm is applied to synthesize the relation schemas. Each individual relation schema should possess the measures of goodness associated with 3NF or BCNF or with some higher normal form.

**Types of Normalization :**

There are five types of normal forms. We shall look into three of them, here

First Normal form ( 1NF )

It is considered to be a part of the formal definition of the relation, in the basic ( flat ) relational model. It is defined to disallow multi – valued attributes, composite attributes and their combinations. It states that the domain of an attributes must include only atomic ( simple, indivisible ) values and that value of any attribute in a tuple must be a single value from the domain of the attribute.

Second Normal form ( 2NF )

This is based on the concept of full functional dependency. A functional dependency X 🡪 Y is a full functional dependency if removal of any attribute A from X means that dependency does not hold any more. That is, for any attribute A belonging to X,(X 🡪(a) ) does not functionally determine Y.

Third Normal form ( 3NF )

This is based on the concept of transitive dependency. A functional dependency X🡪Y in a relational schema R, is a transitive dependency, if there is a set of attributes Z, that is neither a candidate key nor a subset of any key of R, and both X🡪Z and Z🡪 Y hold. According to Cord’s original definition, a relational schema R is in 3NF if it satisfies 2NF and no – prime attribute of R is transitively dependent on the primary key. The system database is built into 6 tables. All the tables are in 1NF.

**9.3 User Interface Design :**

User interface design crease an effective communication medium between a human and a computer, following a set of interface design principles, design identifies interface objectives and the actions and then creates a screen a layout that form the basis for user interface prototype.

User interface design begins with the identification of user, task, and environment requirements. Once user tasks have been identified, user scenarios are created and analyzed to define a set of interface objects and actions. These form the basis for the creation of screen layouts that depict graphical design and placement of icons, definition of description of descriptive screen text, specification and titling for windows.

It is important because, if the software id difficult to use, if it forces you into mistake, or if it frustrates our effort to accomplish our goals.

Although text based interfaces are still widely users, especially in large system, computer users now expect application systems to systems form of graphical user interface. The advantages of GUI are they are relatively easy to learn and use with no computing experience can learn to user the interface. The user has multiple screens for system interface. Switching from one task to another is possible without losing of information generated during the first task.

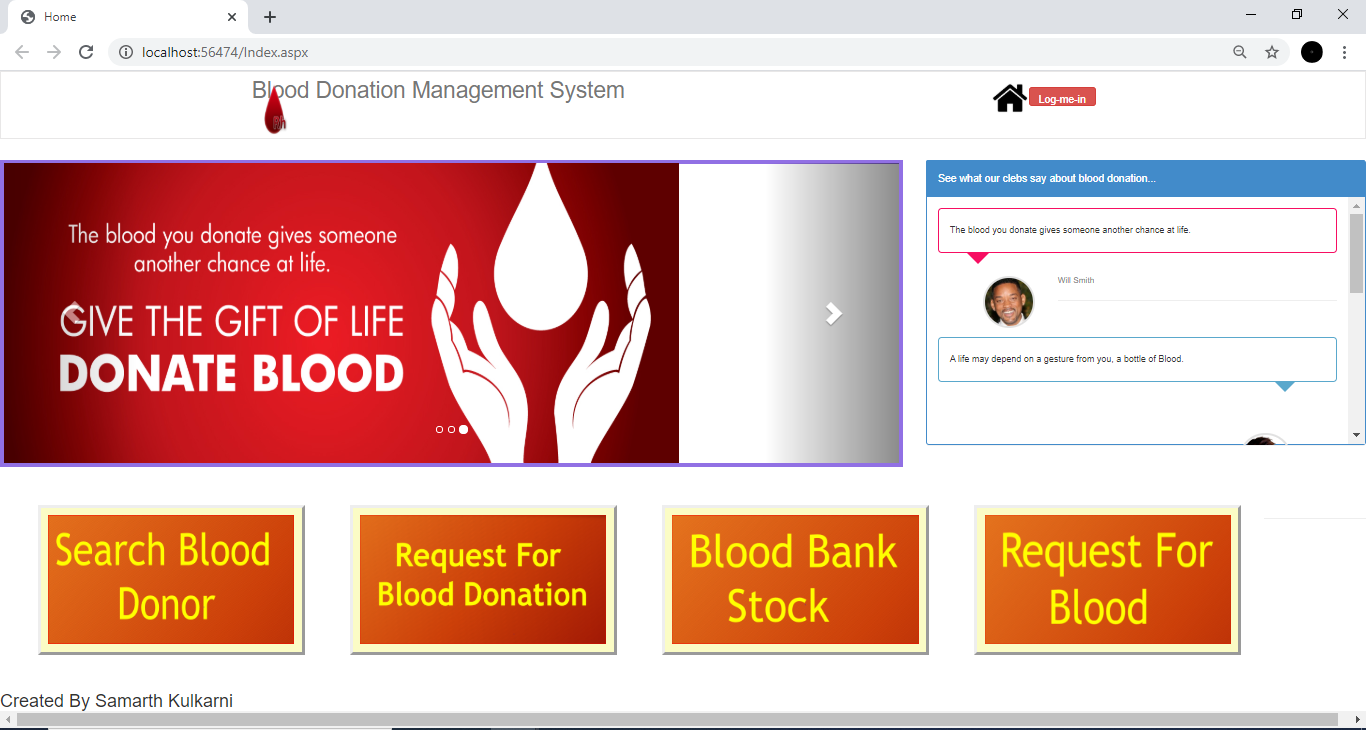
In our project we gave given better GUI that a normal man can access the LAN facility and we have normal English which can be understandable by everyone.

While designing we have taken the physical and mental capabilities of the people who use software. The project on “ Blood Bank Management System “ [ BBMS ] supports the user interface design principle they are :

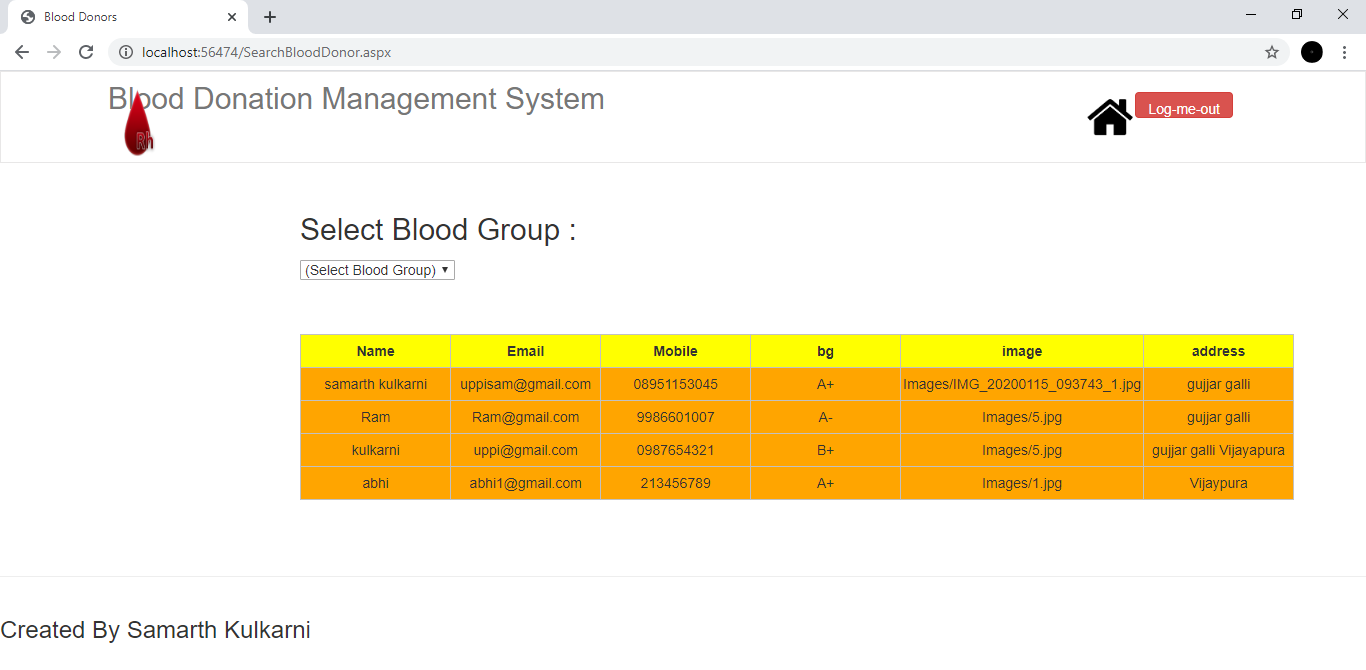
1. **User Familiarity :** The project is user familiarity with minimum computer knowledge, users can access the system
2. **Consistency :** The nomenclature used for identifying various button and fields are very common. So that , user can easily enter right information.
3. **User diversity :** The interface provides the appropriate interaction facilities for the different types of users.
4. **User guidance :** Whenever necessary it provides the information

**Output :**

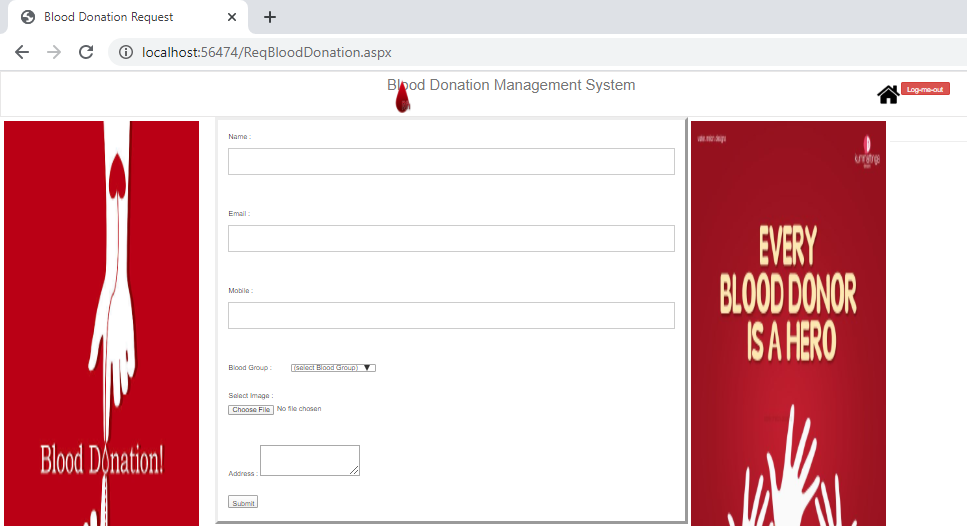
Main Form :



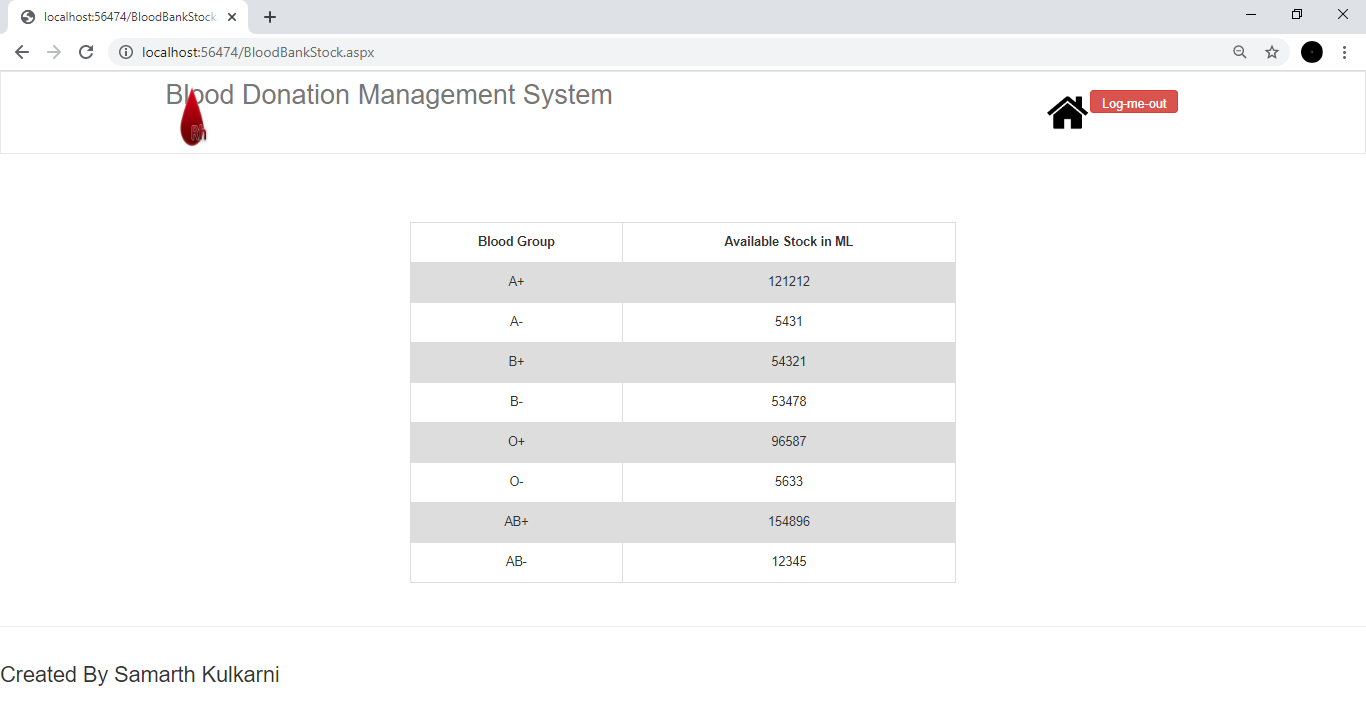
Search Blood Donor :



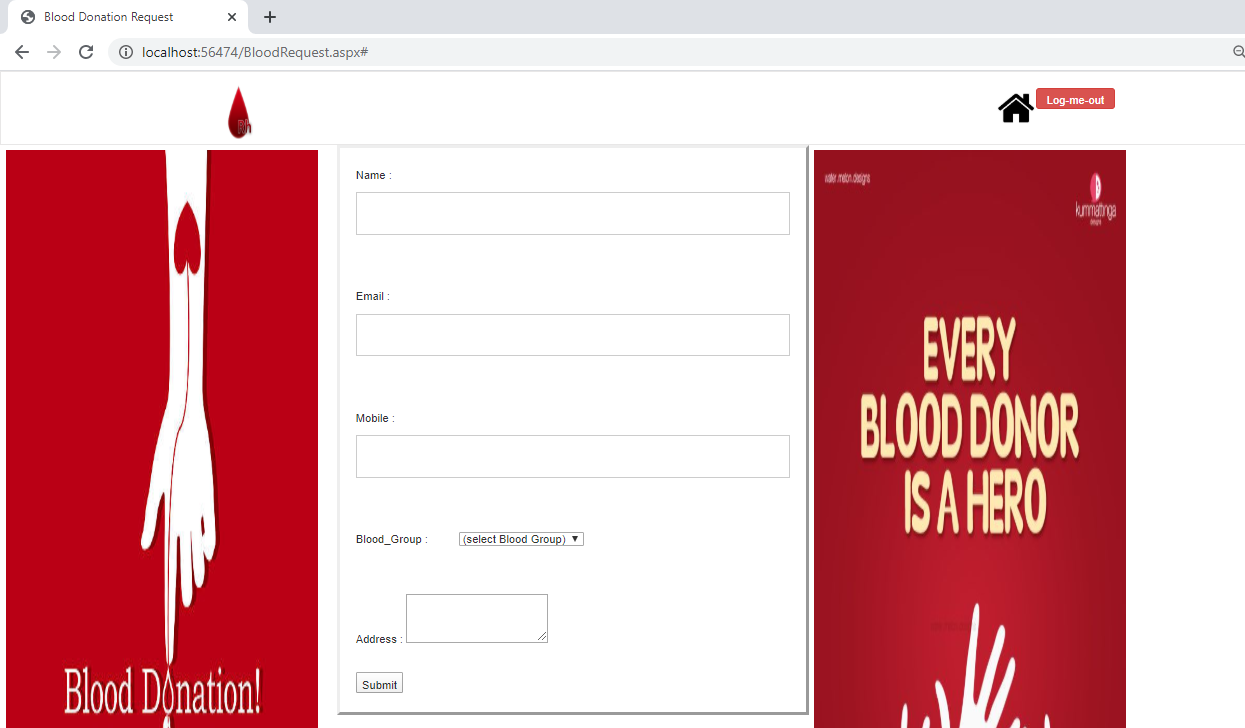
Request for Blood Donation :



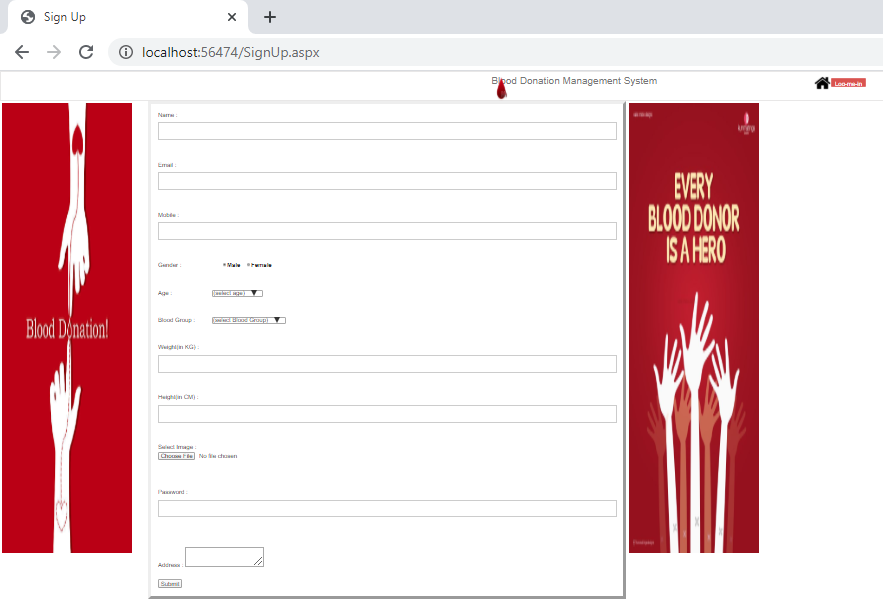
Blood Bank Stock :



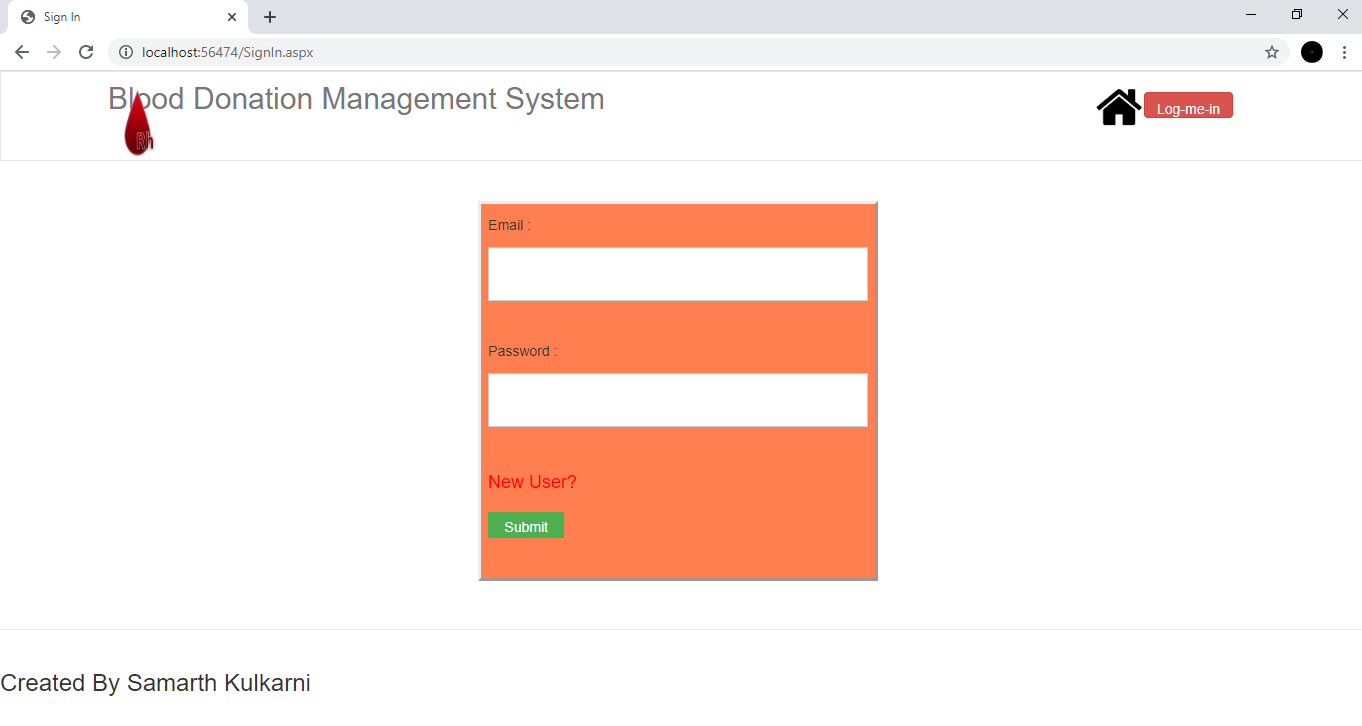
Request for Blood :



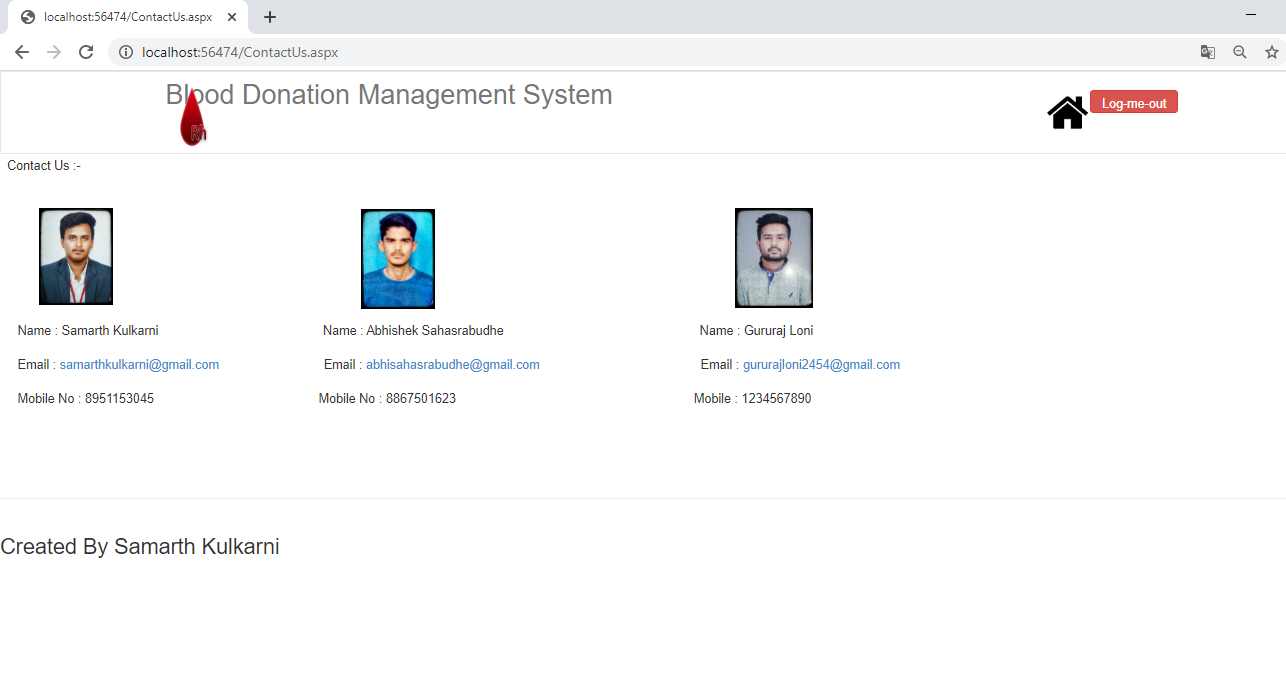
Registration :



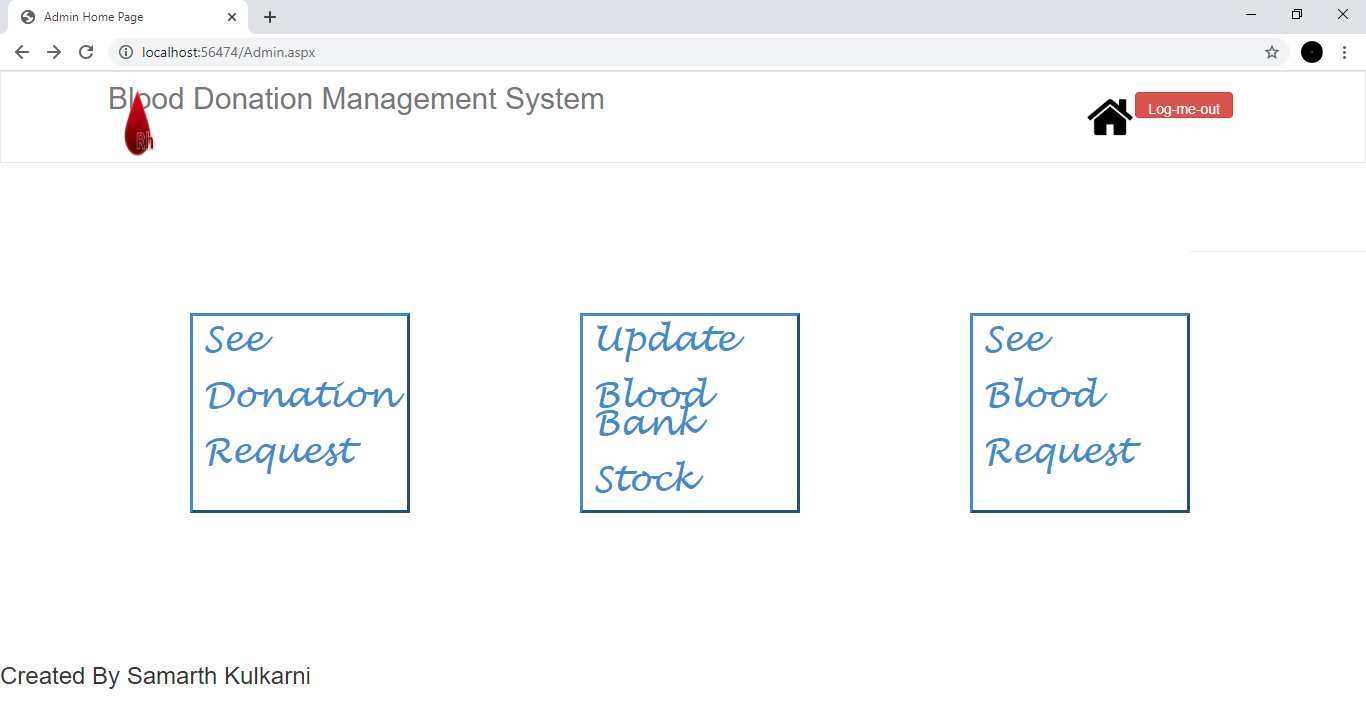
Sign In :



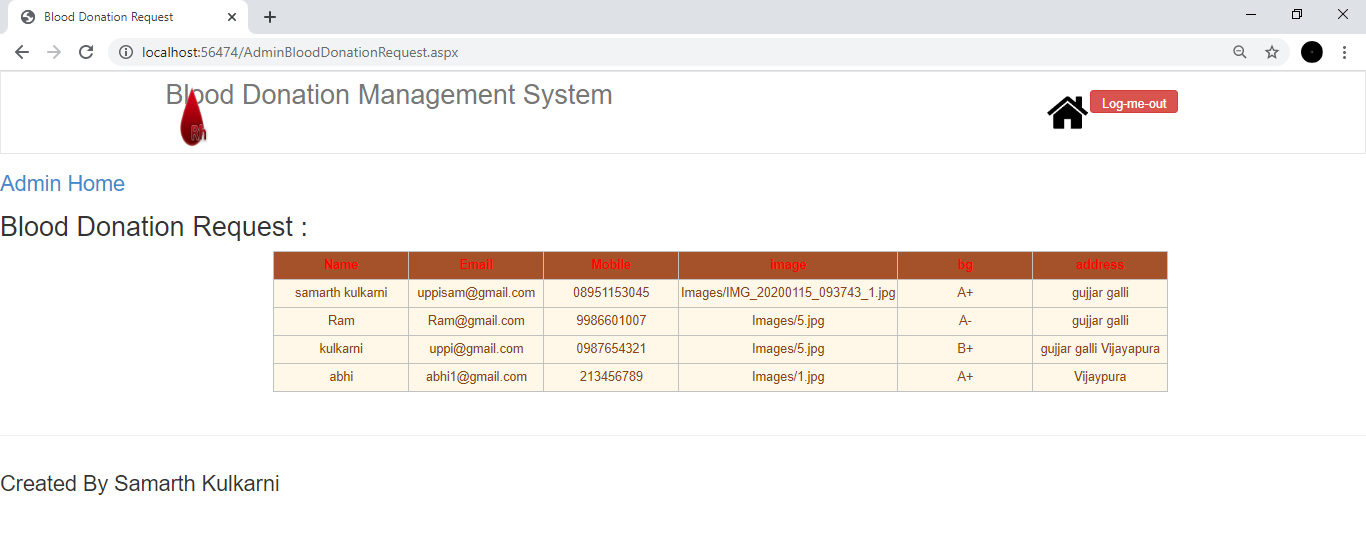
Contact Us :



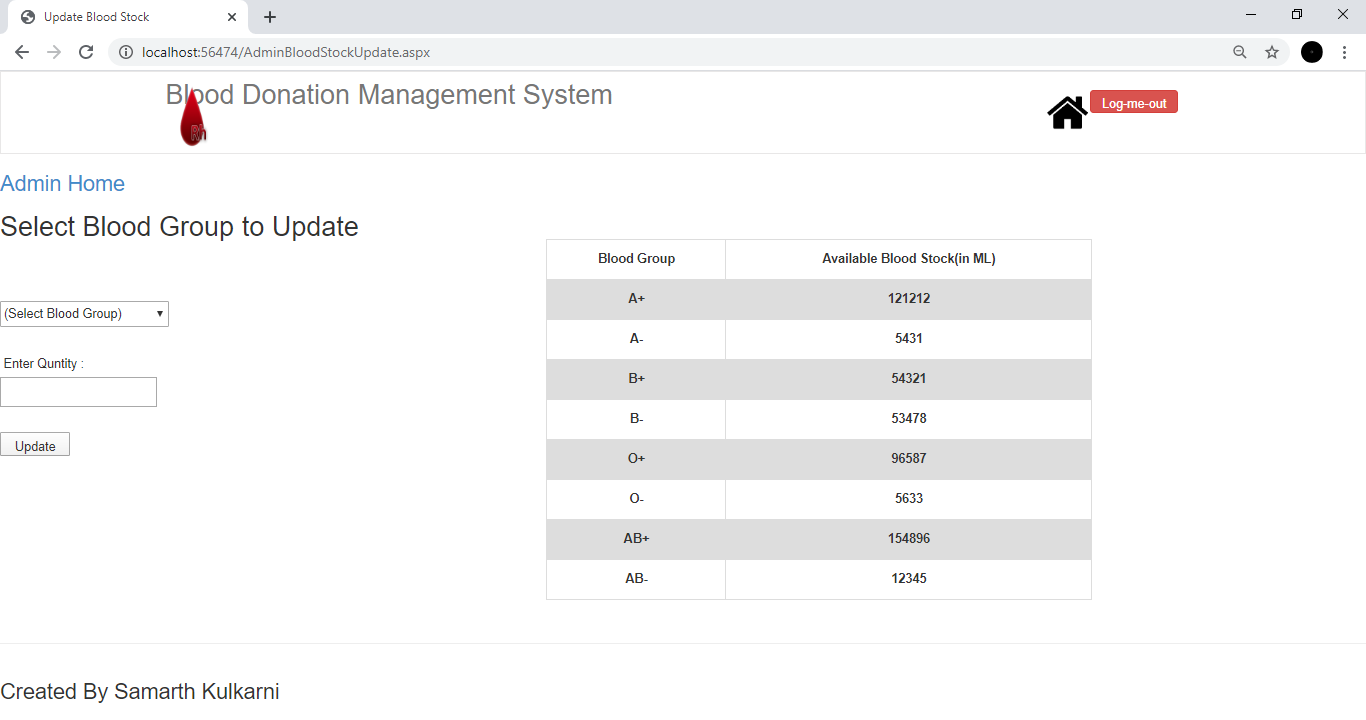
Admin page :



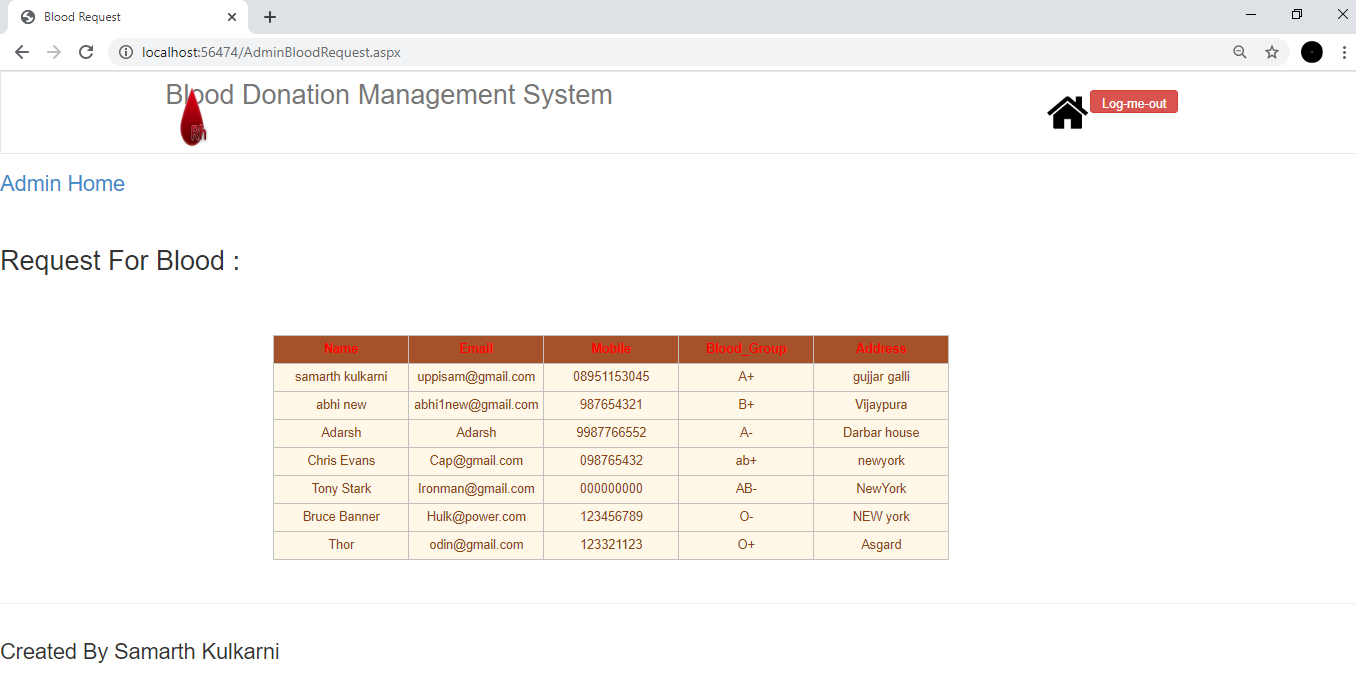
Admin Blood Donation Request page :



Admin Blood Stock Update page :



Admin Blood Request page :



**9.4 Evaluation and Maintenance :**

Evaluation :

In this part we judge the positive and negative sides of the system. Another word we check the strengths and weakness of the system.

Maintenance :

The last part of the system life cycle is system maintenance which is actually the implementation review plan. When system is installed they are generally used long periods The maintenance adapt ion or perfect maintenance, means repairing process or performance. Adapt ion maintenance means changing the program functions, exchanging the performance or modifying the program according to users requirements or changing needs are included in perfective maintenance. It is to keep with changing needs and operational environment. Low level maintenance for this software package as there are many consideration taken to reduce the maintenance, they are

1. Accurately define the user’s requirements during system development.
2. Prepare system documentation in a better way.
3. Use more effective way for designing, processing logic.
4. Make better use of exiting tools and techniques.
5. Manage the systems engineering process effectively.

**10. SYSTEM TESTING**

**10.1 Testing Techniques and Strategies :**

1. Unit Testing :

Unit testing begins at the vertex of the spiral and concentrates on each unit of the software as implemented in source code. Initially test focus on each module individually, assuring that it functions properly as a unit. Hence the name Unit Testing. Unit testing makes heavy use of white box testing techniques, exercising specific paths in a modules control structure complete coverage and maximum error detection. Unit testing focuses on verification efforts of the smallest unit of software design module. Using the procedural design description as a guide important control paths are tested to uncover the errors within the boundary of the module.

1. Integration testing :

Integration testing a systematic technique for constructing the program structure while conducting tests to uncover errors associated with interfacing. The objective is to take unit tested module and build a program structure that has been dictated by design. There is often tendency to attempt non – incremental integration, that is to construct the program using “ big bang “ approach. All modules are combined in advance. The entire program is tested as a whole. A set of errors is encountered. Correction is difficult because isolation of causes is complicated by the vast expanse of the entire program.

**System Testing :**

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer – based system. Although each test has a different purpose, all work to verify that all system elements have been properly integrated and perform allocated functions.

**System Testing Type :**

1. Recovery Testing
2. Security Testing
3. Stress Testing
4. Performance Testing

**11. COST ESTIMATION OF THE PROJECT**

**Introduction To Cost Estimation :**

Several techniques for cost and effort estimation have been proposed over the last 30 years in software engineering, falling into three general categories ( Sheppard, Schofield, Kitchenham, 1996 );

1. Experts opinion :

Expert opinion has been widely used. However, the mean of deriving an estimate are not explicit and therefore not repeatable. Expert opinion, although always difficult to quantity, can be an effective estimating tool on its own or as an adjusting factor for algorithm models

( Gary, MacDonnell, Sheppard, 1999. )

1. Algorithm Models :

Algorithmic models, to date the most popular in the literature. Attempt to represent the relationship between effort and one or more project Characteristics. The man “ cost driver “ used in such a model is usually taken to be some notion of software size ( such as the number of lines of source code, number of pages ). Algorithmic models need calibration or adjustment to local circumstances. Examples of the algorithmic models are the COCOCMO models ( Boehm, 1981 ) and the SLIM model ( Putnam, 1978 )

**12. FEATURE SCOPE AND FURTHER ENHANCEMENT OF THE PROJECT**

This project is standalone it can be made web based technology by using the same logic and front end may be changed and some other additional features may also incorporated so that all blood bank in the district may be interconnected for proper co – ordination.

**13. CONCLUSION**

The scope of our project on “ **Blood Bank Management System [ BBMS ]** “ is highlighted by the following aspects.

* The project is user friendly and very easy for beginners to understand.
* Facilities such as retrieving donor information, adding new donor information, deleting, and listing have been provided.
* Calculation of transaction is very easy.
* Database management for donor, recipient, supervisor and the general public is provided. Printing of the report is also made very easy.
* This package makes the entire Blood Bank control room process easy.

Limitations come across each very work done by human beings. We all have our own limitations and our package also has some limitations. This package does not have any facility for checking whether or nor there is sufficient time gap between two successive blood donations by a donor.

The system may be enhanced by incorporating this facility and a few more.

1. **.BIOGRAPHY AND REFERENCES**
2. Head first Visual :- By Basham, Sierra and Bates.
3. Complete Reference :- 5th Edition by Herbert Schildt, Published by MeGraw – Hill Publishing Company Limited.
4. Software Engineering :- A Practitioner’s Approach, Fifth Edition by Roger S Pressman, Published by McGraw – Hill International Edition
5. Beginning Visual Basic :- By Kelvin Muker and James L weaver with James P srume published by a! Express
6. Google for problem solving
7. http://www.javaworld.com/javaworld/jw-01-1998/jw-01-Credentialreview.html
8. Database Programming with JDBC and Java by O'Reilly
9. Head First Java 2nd Edition
10. http://www.jdbc-tutorial.com/
11. Java and Software Design Concepts by Apress
12. https://www.tutorialspoint.com/java/
13. http://www.javatpoint.com/java-tutorial
14. https://docs.oracle.com/javase/tutorial/
15. http://www.wampserver.com/en/
16. http://www.JSP.net/
17. http://www.tutorialspoint.com/mysql/