Software Requirements Specification
Version 1.1
September 8, 2015

Blood Donation System

Milan Patel (U12CO083)
Jay Panchal (U12CO073)

Table of Contents

1.0. Purpose

1.1. Introduction

This Software Requirements Specification provides a complete description of all the functions and specifications of the blood donation system.

The expected audience of this document is operators who manage this system.

1.2. Scope

The Blood Donation System is designed to run on the Blood Bank server and to allow the donors who wishes to donate blood to get information they require and for system operators to create a new database entry, update an existing database entry, or contact donor for blood requirements. The data will be held in an Access database on the server.

1.3. Glossary

Term	**Definition**
BDS	Blood Donation System
BDSD	Blood Donation System Database
CI	Configuration Item
Entry	Donor details stored in database
Html	Hyper text markup language
IEEE	Institute of Electrical and Electronic Engineers
QA	Quality assurance
SCMP	Software Configuration Management Plan
SDD	Software Design Document
SQAP	Software Quality Assurance Plan
SRS	Software Requirements Specification
Tbd	To be decided
Tbn	To be named
Web Site	A place on the world wide web

1.4. References

[IEEE] The applicable IEEE standards are published in “IEEE Standards Collection,” 2001 edition.

[Bruade] The principal source of textbook material is “Software Engineering: An Object-Oriented Perspective” by Eric J. Bruade (Wiley 2001).

1.5. Document overview

The remainder of this document is two chapters, the first providing a full description of the project for the owners of the Red Blood Bank. It lists all the functions performed by the system. The final chapter concerns details of each of the system functions and actions in full for the software developers’ assistance. These two sections are cross-referenced by topic; to increase understanding by both groups involved.

2.0. Overall description

The BDS encompasses numerous files and information from the BDSD. This system will be completely web-based, linking to BDSD and the remote web server from a standard web browser. An Internet connection is necessary to access the system.

2.1. System environment

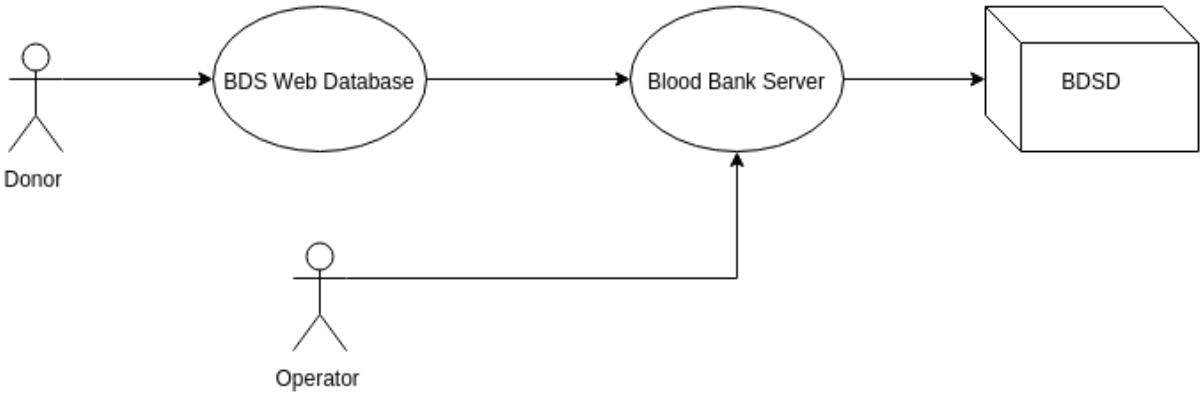


Figure 1 System Design

The BDS web site will be operated from the blood bank server. When an Donor connects to the BDS Web Server, the BDS Web Server will pass the donor to the Blood Bank Server. The Blood Bank Server will then interact with the BDSD, which allows the Windows type program to transfer data to and from a database.

2.2. Functional requirements definitions

Functional Requirements are those that refer to the functionality of the system, i.e., what services it will provide to the user. Nonfunctional (supplementary) requirements pertain to other information needed to produce the correct system and are detailed separately.

2.3. Use cases

The system will consist of Home page with three sections.

The first section is for operator authentication, which requires operator_id and password. The second section is to the Entries section. It is to add a new entry for donor. A form is presented to the operator to be filled in. Certain fields in the form will be required, and list boxes will be used where appropriate. This section can also be used to update entries.

The third section is to search Donor. This will give three options to query the database based on donor name, blood-type and city. A table will be presented providing the requested details.

All pages will return the Operator to the Home Page.

2.3.1. Use Case: Access Home Page

Brief Description
The BDS Web Server is waiting on an Operator or donor to connect.

Initial step-by-step description
For this use case to be initiated, the operator must be connected to the Internet and connected to the BDS Web Server.

1. The Operator connects to the BDS Web Server.
2. The Operator selects the home page.
3. The BDS Web Server passes the Operator to the Home Page.

Reference SRS 3.2.1

2.3.2. Use Case: Create New Entry

Brief Description:
The Operator chooses to create a new entry on the Entries page.

Initial step-by-step description.

For this use case to be initiated the Operator must be connected to the Internet and on the Entries page.

1. The Operator selects the “Add Entry” link.
2. The BDS Server returns the “Add a New Donor Form.”
3. The Operator fills in the form.
4. The Operator can choose which fields to make public or private.
5. The Operator clicks submit.
6. The BDSD checks to see if all required fields contain data.
7. If all required fields contain data the BDS Server adds the data to the BDSD.
8. If a required filed is empty the BDS Server returns the form to the Operator with a message.
9. The BDS Server returns the Operator to the Home Page.

Reference: SRS 3.2.3

2.3.3. Use Case: Update an Entry.

Brief Description:
The Operator chooses to update an existing entry in BDSD.

Initial step-by-step description:
For this use case to be initiated the Operator must be connected to the Internet and on the Search page.

1. The Operator chooses the “Search” option.
2. The BDS Server presents the Operator with a form.
3. The Operator fills in the name of the donor.
4. The BDS Server returns a table with all the details of that donor.
5. The Operator double clicks to update information.

6. The BDS Server makes changes to the BDSD.

7. The BDS Server returns the Operator to the Home Page.

Reference: SRS 3.2.4

2.3.4. Use Case: Search for a Donor

Brief description:
The Operator chooses to search donor.

Initial step-by-step description:
For this use case to be initiated the Operator must be connected to the Internet and on the Home Page.

1. The Operator chooses “Search.”

2. The BDS Server presents a form requesting name, city or blood type of the donor.

3. The Operator fills in the form and clicks submit.

4. The BDS Server queries the BDSD for the requested information.

5. The BDS Server returns requested details in tabular form.

6. The BDS Server will return the Operator to the Home Page.

Reference: SRS 3.2.5

2.4. Non-functional requirements

There are requirements that are not functional in nature. Specifically, these are the constraints the system must work within.

The web site must be compatible with both the Netscape and Internet Explorer web browsers. This system will use the same type of Internet security presently being used by Red Blood Bank.

3.0. Requirement specifications

3.1. External interface specifications

None

3.2. Functional Requirements

3.2.1. Access Home Page

Use Case Name:	Access Home Page
Priority	Essential
Trigger	Menu selection
Precondition	Operator is connected to the Internet and on the home page
Basic Path	1. BDS Web Server sends the Operator to the Blood Bank Server. 2. The BDS Server presents the Operator with the Home Page.
Alternate Path	N/A
Postcondition	The Operator is on the Home Page
Exception Path	If there is a connection failure the BDS Server returns to the wait state
Other	
Reference	SRS 2.3.1

3.2.2. Create a new entry

Use Case Name:	Create a new entry
Priority	Essential
Trigger	Menu selection
Precondition	The Operator must be connected to the Internet and on the Entries page.
Basic Path	1. The Operator clicks on add a new entry. 2. The BDS Server returns a form. 3. The Operator fills in the form and clicks submit. 4. The BDS Server checks to see if any required field is empty. 5. If any required field is empty the BDS Server will send a message and return the Operator to the new entry form page. 6. If no required field is empty the BDS Server will create a new record in the Donor Table in the BDSD, and return the Operator to the Home Page. 7. The Operator may select Cancel.

	8. If the Operator selects Cancel, the form is cleared and the Operator is returned to the Home page.
Alternate Path	N/A
Postcondition	A record is created in the Donor Table of the BDSD.
Exception Path	1. If the connection is terminated before the form is submitted, the fields are cleared and the BDS Server is returned to the wait state. 2. If the connection is terminated after the form is submitted, but before the Operator is returned to the Home Page, the record is created in the Donor Table of the BDSD.
Other	
Reference:	SRS 2.3.3

3.2.3 Update an Entry

Use Case Name:	Update an Entry
Priority	Essential
Trigger	Menu selection
Precondition	The Operator must be connected to the Internet and on the Search Page.
Basic Path	1. The Operator clicks on search link. 2. The BDS Server returns a form. 3. The Operator enters donor name. 4. The BDS Server queries the BDSD for that particular donor and returns a table of all details of that donor. 5. The operator double clicks particular field and updates information. 6. The BDS Server replaces the old data with the new data and returns the Operator to the Home Page.
Alternate Path	None.
Postcondition	The record in the Donor Table of the BDSD has been updated and the Operator is returned to the Home Page.
Exception Path	1. If the connection is terminated before the form is submitted, the fields are cleared and the BDS Server is returned to the wait state. 2. If the connection is terminated after the

	form is submitted, but before the Operator is returned to the Home Page, the record in the Donor Table of the BDSD is updated and the BDS Server is returned to the wait state
Other	
Reference:	SRS 2.3.4

3.2.4 Send Message Alerts

Use Case Name:	Message Alerts
Priority	Optional
Trigger	Automatic
Precondition	Donor must provide his/her contact number.
Basic Path	None
Alternate Path	None.
Postcondition	The Donor gets Message Alert when there is requirement of blood of his type or when he is able to donate blood again after previous donation.
Exception Path	
Other	
Reference:	SRS 2.3.4

3.3. Detailed non-functional requirements

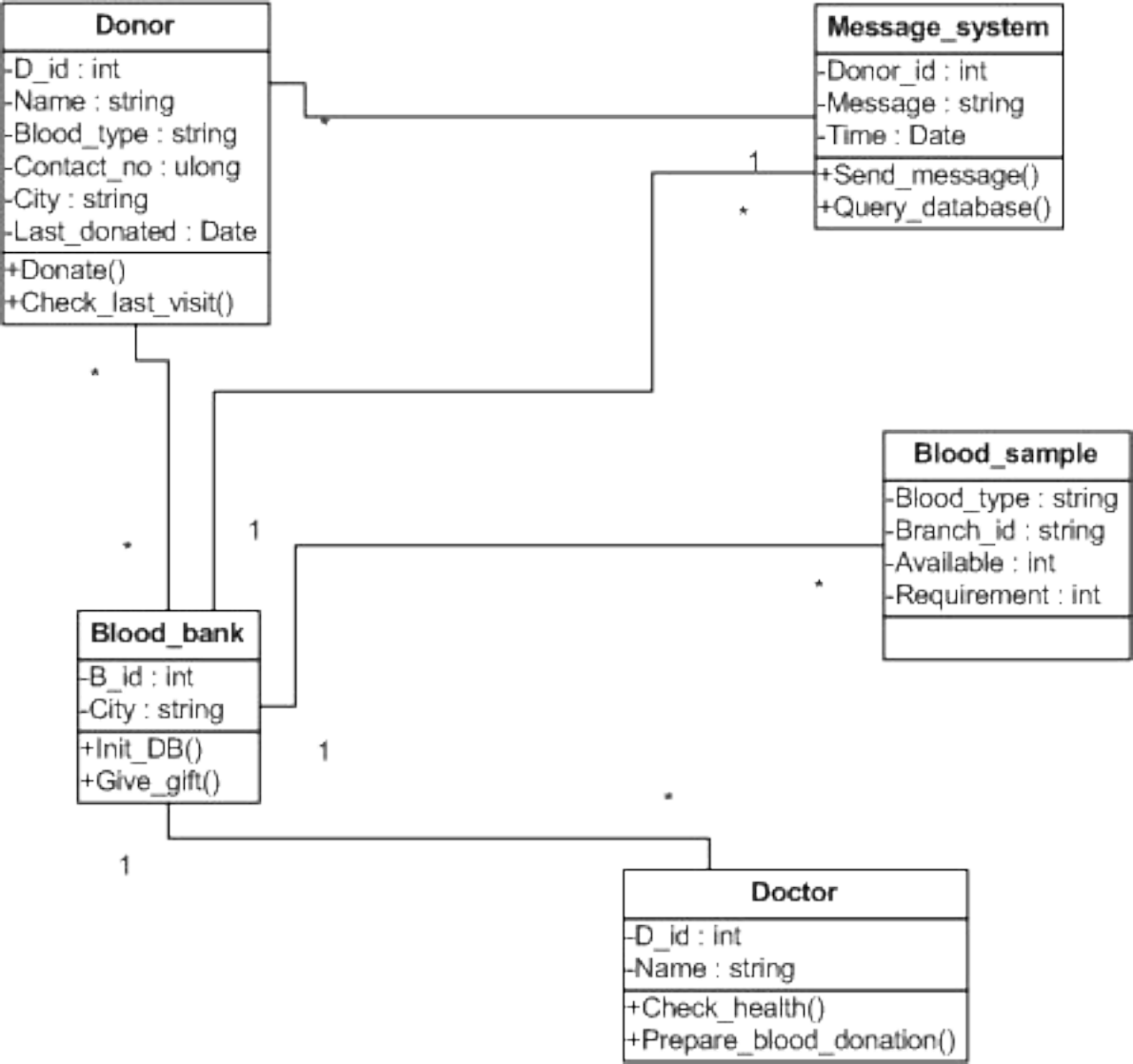
Attribute Name	Attribute Type	Attribute Size
LastName*#	String	30
FirstName*#	String	30
Address1*#	String	50
Address2#	String	50
City*#	String	30
State*#	String	2
Zip*#	Int	6
Children#	String	50
EmailAddress#	String	20
ReceiveEmails#^	Boolean	1

Fields marked with an ‘*’ are required fields. Fields marked with a ‘#’ can be visible or not visible and is determined by the Operator. Fields marked with a ‘^’ are never visible to anyone other than the Operator.

Hardware:	BDS Server
Operation System	Window 98 or above
Internet Connection	Existing telephone lines
Code Standard	The web pages will be coded in html by using Front Page. The forms will be done in Java Server Pages. The connection to the BDSD will be done with Windows BDE. Each page of the web site will be fully documented.
Performance	The system should generate the records in the appropriate table of the BDSD 100% of the time.

3.4. System Evolution

In the future this system will be update to allow donors to query the database and when they require blood for their relatives or friends then they can get them with their credits which they secured when they donated blood.


Donor
-D_id : int
-Name : string
-Blood_type : string
-Contact_no : ulong
-City : string
-Last_donated : Date
+Donate()
+Check_last_visit()
Message_system
-Donor_id : int
-Message : string
-Time : Date
+Send_message()
+Query_database()
Blood_sample
-Blood_type : string
-Branch_id : string
-Available : int
-Requirement : int
Blood_bank
-B_id : int
-City : string
+Init_DB()
+Give_gift()
Doctor
-D_id : int
-Name : string
+Check_health()
+Prepare_blood_donation()
1
*
1
*
*
1
*
1
1
*

Blood Donation System

Donor

Donate Blood

Check last visit

Visit center

Add donor entry in system

Delete donor entry

Modify donor entry

Generate monthly report

Operator

Message System

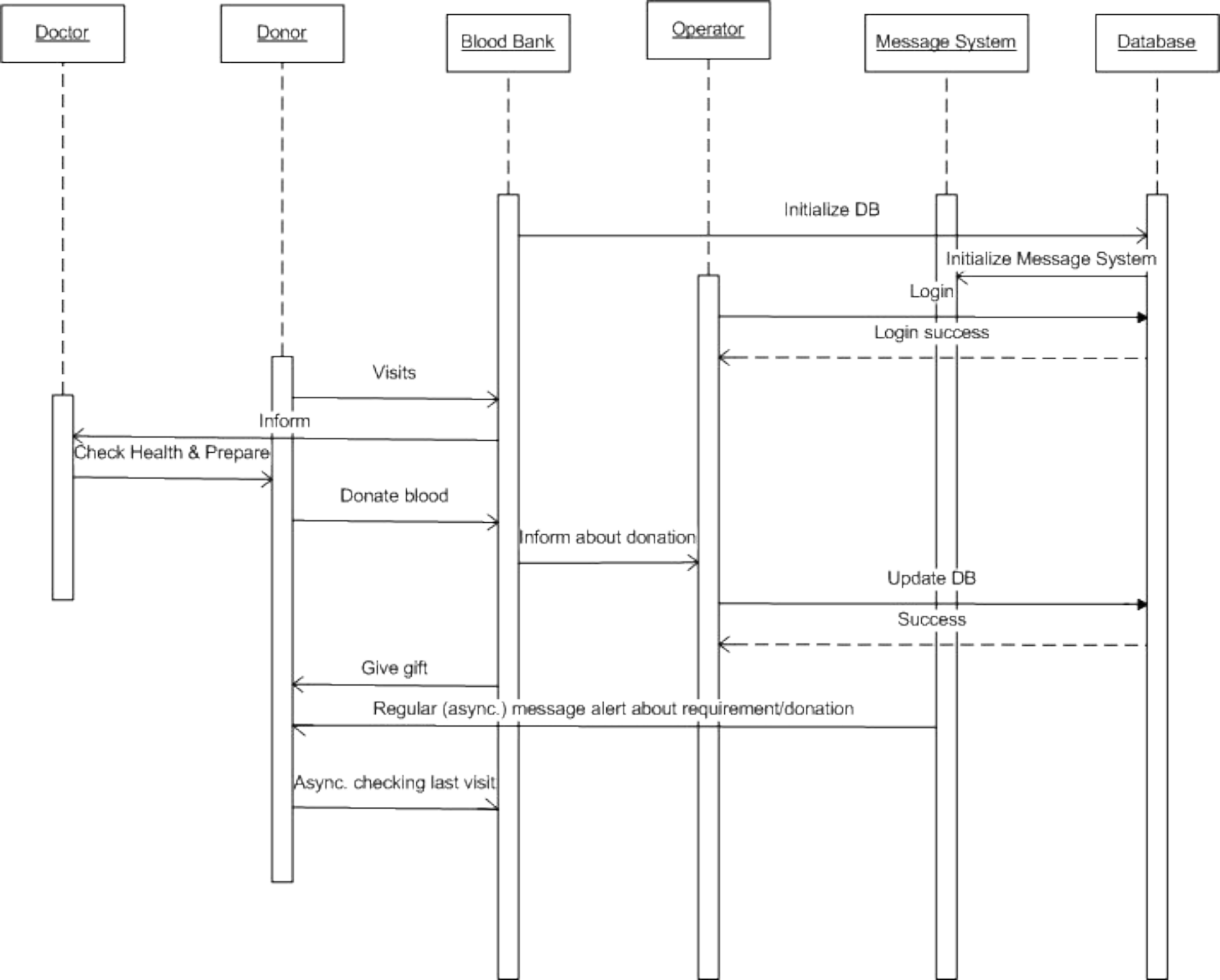
Generate Alert Message

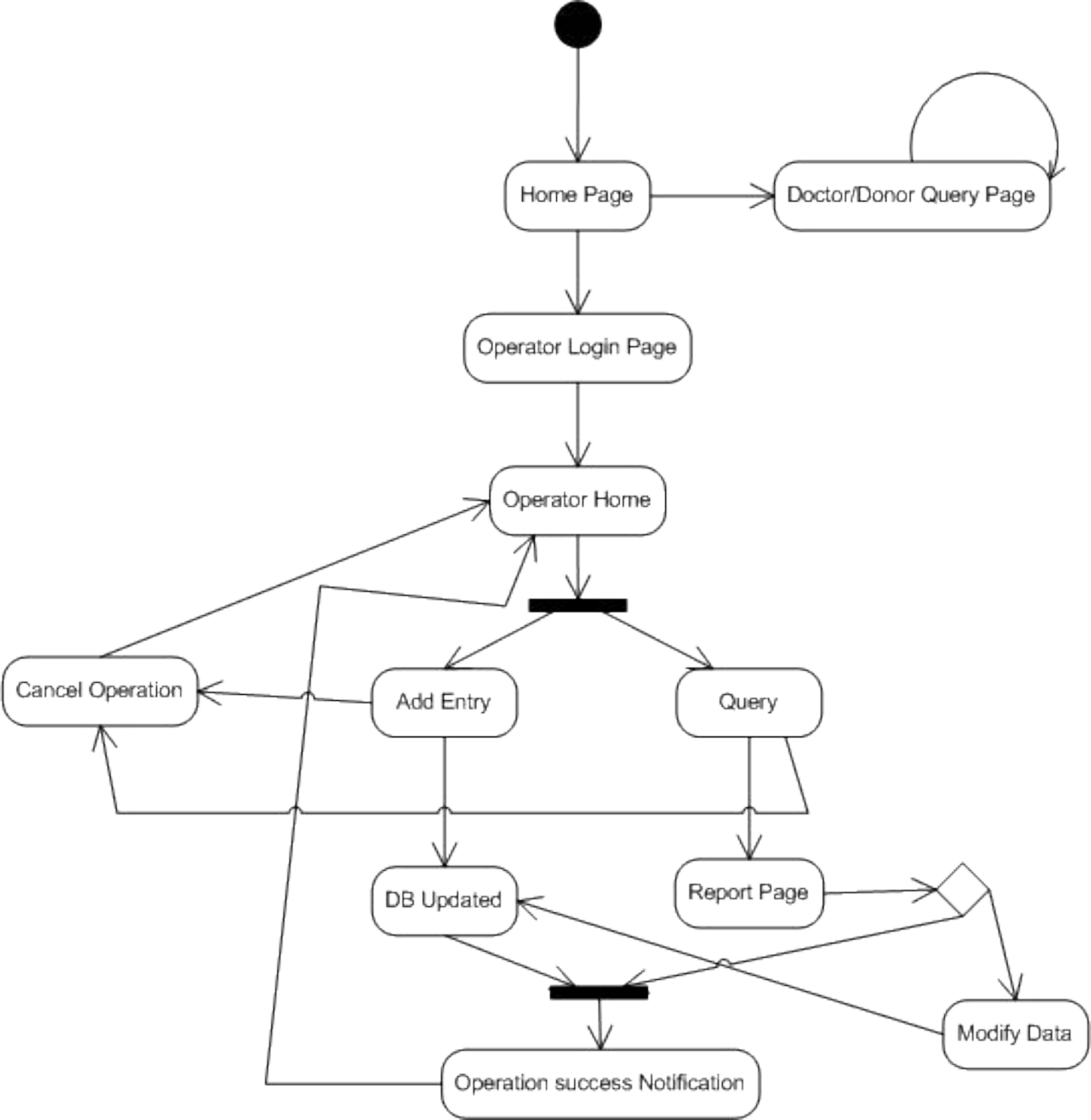
Query System

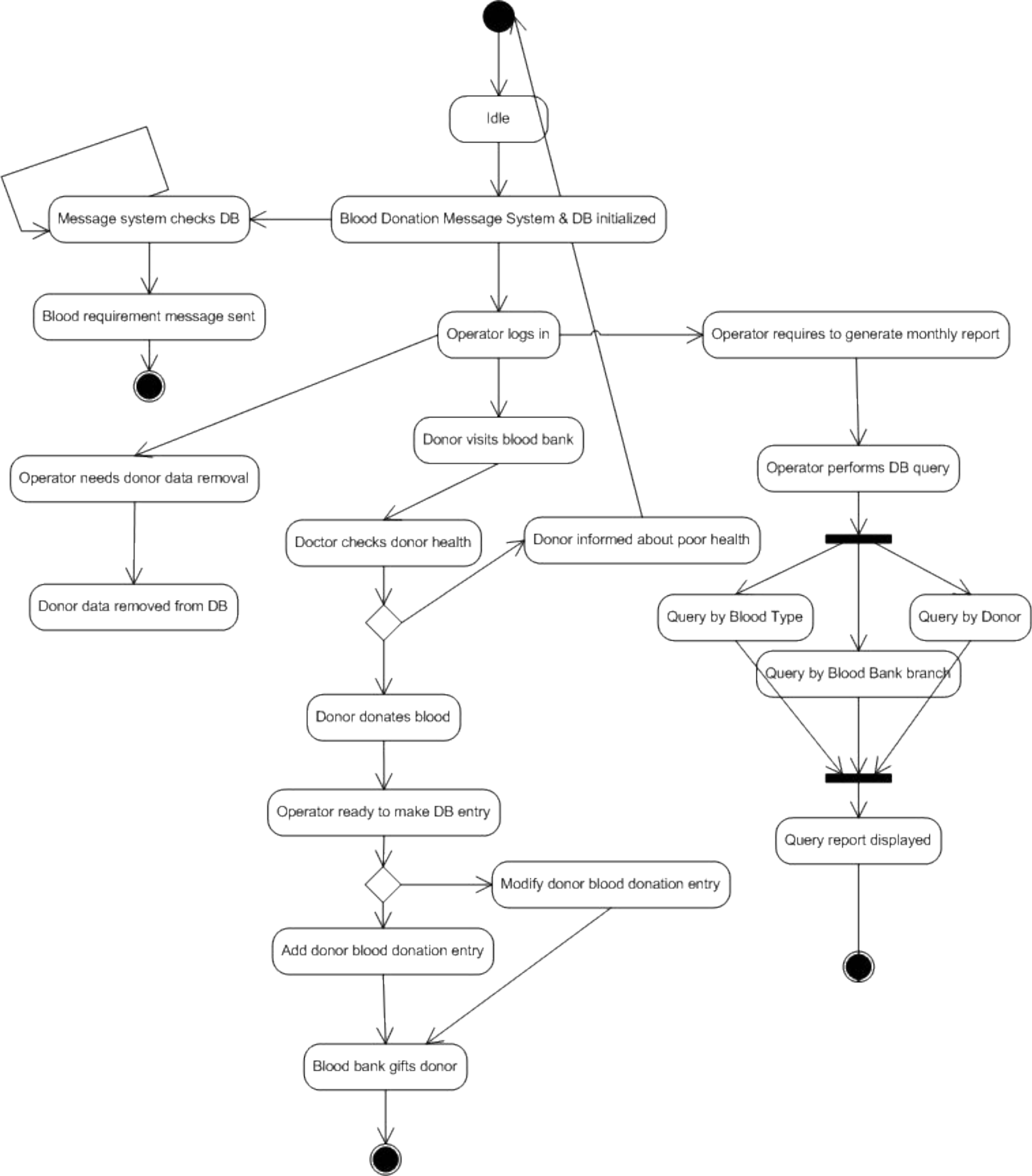
Prepare donor for blood donation

Check donor health

Doctor


Doctor
Donor
Blood Bank
Operator
Message System
Database
Initialize DB
Initialize Message System
Login
Login success
Visits
Inform
Check Health & Prepare
Donate blood
Inform about donation
Update DB
Success
Give gift
Regular (async.) message alert about requirement/donation
Async. checking last visit


Home Page
Doctor/Donor Query Page
Operator Login Page
Operator Home
Cancel Operation
Add Entry
Query
DB Updated
Report Page
Modify Data
Operation success Notification


Idle
Blood Donation Message System & DB initialized
Message system checks DB
Blood requirement message sent
Operator logs in
Operator requires to generate monthly report
Operator needs donor data removal
Donor visits blood bank
Operator performs DB query
Donor data removed from DB
Doctor checks donor health
Donor informed about poor health
Query by Blood Type
Query by Donor
Query by Blood Bank branch
Donor donates blood
Query report displayed
Operator ready to make DB entry
Modify donor blood donation entry
Add donor blood donation entry
Blood bank gifts donor