University of Jaffna

Software Engineering

Project: Blood Donation Application

GitHub: https://github.com/miyushan/LifeSource/

Project by Group 2:

Arshath J.M. (2018/E/012)

Nayanajith Y.G.A. (2018/E/083)

Nuwansiri W.D.A. (2018/E/085)

Rodrigo S.M. (2018/E/102)

Table of Contents

1. Customer Problem Statement

a. Problem Statement

Lack of blood reservation is a terrible issue in Sri Lankan Health Department. That can affect the patient who suffers from blood requirements. Sometimes that patient has to pay with their lives for this issue.

When we consider the contributors to the local blood bank, blood donation campaigns take a major role in it. But unfortunately, it can't fulfill the need for blood due to some drawbacks. The main drawback can list down as,

- Insufficient blood storage
- Problems in information flow
- Lack of negative blood donors
- Finding blood in emergencies
- Traffic in blood donation campaigns

We have chosen this topic intending to find a method to improve the efficiency of the blood donation process. We propose a software system that keep details of blood bank and organizations to deliver to people who interested. Also this system allow organizations to get permissions from PHI in their area and notify people about emergency blood requirement.

b. Glossary of Terms

1. **Blood bank:** A place where supplies of blood or plasma for transfusion are stored
2. **Negative blood:** Absence of protein called Rh factor in blood. People who have A-, B-, AB- and O- blood type.
3. **PHI:** Public Health Inspector
4. **Organization:** Organizations that arrange and conduct blood donation camps

2. System of Requirements

a. User Interface Requirements

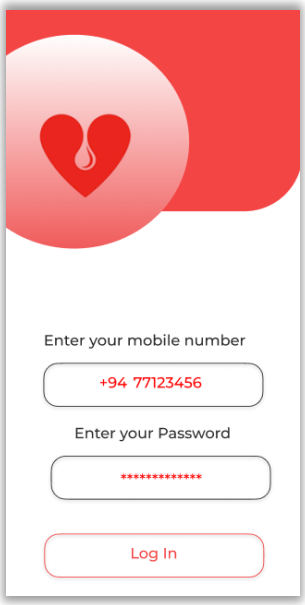
Identifier	Priority Weight	Requirements
REQ - 1	10	System should have secure login
REQ - 2	10	System should have secure landing page to register
REQ - 3	8	System should allow to update the parameters
REQ - 4	10	System should store data in secured manner
REQ - 5	10	System should allow user to unregister and delete data upon unregistering
REQ – 6	10	System should ask the user for consent to share data for statistical comparison
REQ – 7	10	System should allow organizations to post events
REQ – 8	10	System should allow organizations to get confirmation feedback
REQ – 9	10	System should allow users to get details about blood bank and blood donation camp details
REQ – 10	10	System should allow Blood banks to update blood bank data
REQ – 11	4	System should allow users to give feedback and provide support as well
REQ – 12	10	System should give notification for emergency post and events
REQ – 13	10	System should allow organizations to get permission from PHI
REQ – 14	9	System should provide blood bank statistics to users (donors)
REQ – 15	7	System should show past records of users
REQ – 16	10	System should have option to share personal details
REQ – 17	10	System should ask permission to share personal details
REQ – 18	10	System should have questionnaire to take user details
REQ – 19	10	System should have option to track user
REQ – 20	10	System should get permission to track user
REQ – 21	10	System should show user’s personal details
REQ – 22	7	System should show about new application updates

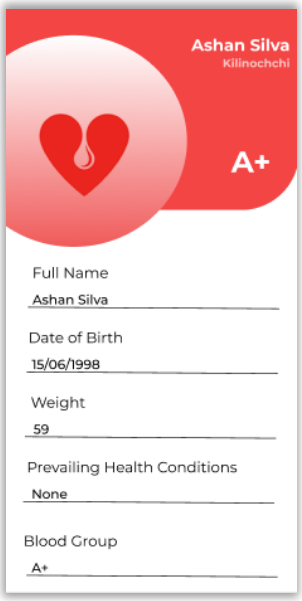
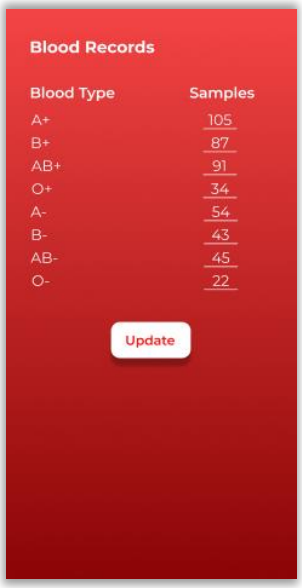
b. Enumerated Non-Functional Requirements

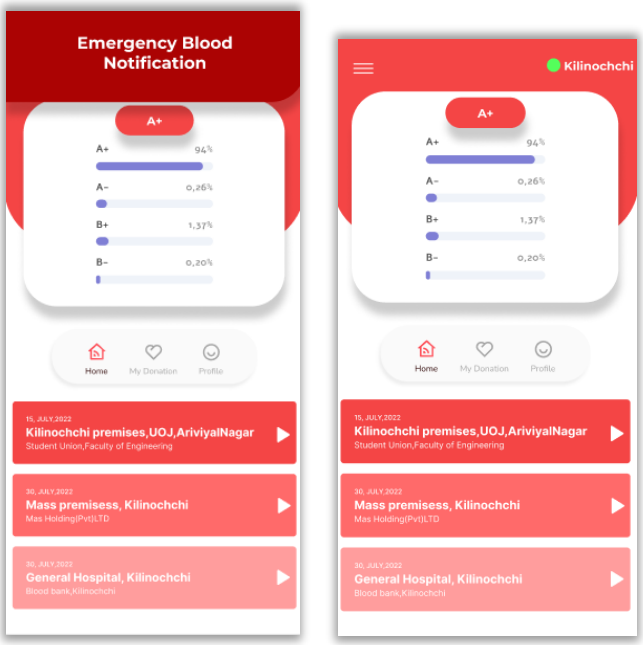
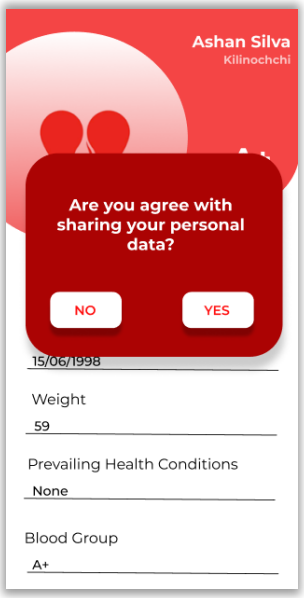
Identifier	Priority Weight	Requirements
REQ – 15	8	As a system, size and generality of the data must be defined
REQ – 16	10	As a system, all the user data must be encrypted
REQ – 17	2	System should be scalable and load balanced
REQ – 18	4	As a system, data of the entire system must be archived periodically

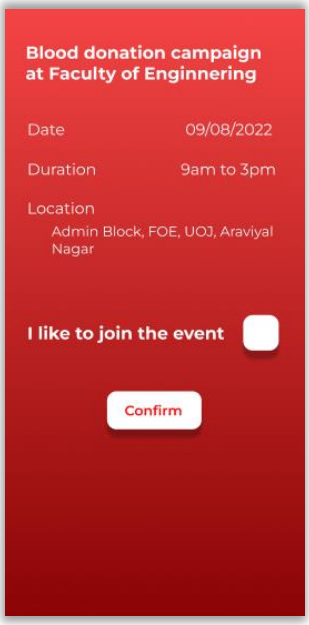
REQ – 19	6	As a system, backups of the data must be taken periodically
REQ – 20	5	As a system, appropriate business continuity policies and disaster recovery strategy must be implemented
REQ – 21	7	As a system, system maintenance should be done regularly in order to keep systems up to date
REQ – 22	10	User manual and Architecture Diagram along with proper Documentation of the system must be provided
REQ – 23	7	As a system, communication between system actors must be secured
REQ – 24	7	As a system, system maintenance should be done regularly in order to keep systems up to date

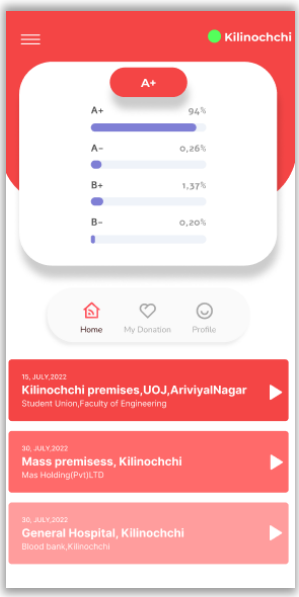
c. User Interface Requirements

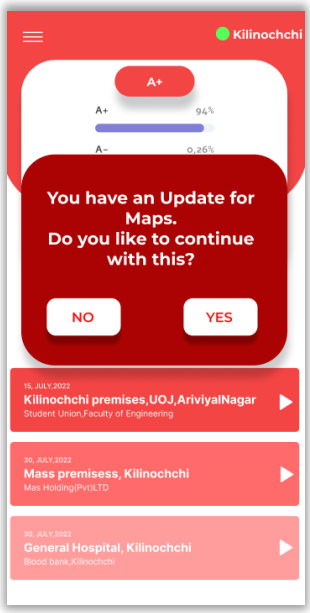
Identifier	Priority Weight	Requirements
REQ-01	10	GUI must have a Landing page to Login. 
REQ-02	10	GUI must have a Landing page to Register. 
REQ-21	10	GUI must have a page for user's personal information.

		
REQ-10	10	GUI must have a page to update data(Blood records). 
REQ-12	10	GUI must show notification for emergency post and events.

		
REQ-06	10	GUI must have consent option for share personal data. 
REQ-19	10	GUI should have an option to track user preference (Participant to the event).

		
REQ-18	10	GUI should include a Questionier for take user data. 
REQ-14	9	GUI should able to provide statictical data.

		
REQ-15	7	GUI must have a page to see user's past records. 
REQ-22	7	GUI should able to show new updates.

		 Kilinochchi A+ A+ 94% A- 0.26% You have an Update for Maps. Do you like to continue with this? NO YES 15, JULY,2022 Kilinochchi premises,UOJ,AriviyalNagar Student Union,Faculty of Engineering 30, JULY,2022 Mass premisess, Kilinochchi Mas Holding(Pvt)LTD 30, JULY,2022 General Hospital, Kilinochchi Blood bank,Kilinochchi

3. Functional Requirement Specification

a. Actors and Goals

Actor	Actor's Goal	Use Case Name
User	To login to the app to check current blood data	Login(UC-1)
Visitor	To register in the system	Register (UC-2)
User	To add data in the system via mobile app	Add data(UC-3)
User	To get upcoming event details	upcoming event(UC-4)
User	To get notified of recommended timeslot for donating	notify donating(UC-5)
User	To have its data validated before adding data in the system	Data validation(UC-6)
User/Blood bank	To request an event	Request event(UC-7)
Blood bank	To maintain blood resources data	maintain data(UC-8)
Blood bank	To contact donors	contact(UC-9)
PHI	To inform the requested event to blood bank	event verify(UC-10)
System Admin	To resolve issues that users might face while user performs any operations or has any other issues	Communication(UC-11)
Database /Repository	To store the user information, login details, data	UC-1,UC-2,UC-3,UC-4, UC-5,UC-6,UC-7,UC-8, UC-9,UC-10,UC-11

b. Use Cases

i. Casual Description

UC#1 Login

The user can enter the application by entering his credentials used for registration. User can perform various operations, like View upcoming blood donation camps details.

UC#2 Register

User can register to the system through app. Then System will inform to the new user date to medical checkup. If user pass the medical checkup then approve his or her registration request. Otherwise system will inform the details for rejected.

UC#3 Add data

The user, once registered into the system, will be able to update his or her personal details, like contact number, personal address, login password

UC#4 Upcoming events

User can get information about upcoming blood donation camps. It inform Date,venue and most requested blood group.

UC#5 Notify donating
Previously blood donated user can get notification of his or her next donation date.

UC#6 Data validation
The data validator will check if the data added by the user into the app is within the appropriate range. if any outliers exist in the data entered, an error message will pop up prompting the user to re-enter his valid data.

UC#7 Request event
The organizer can request a organize a blood donation camp through the app. Then PHI officer can inform to the blood bank.

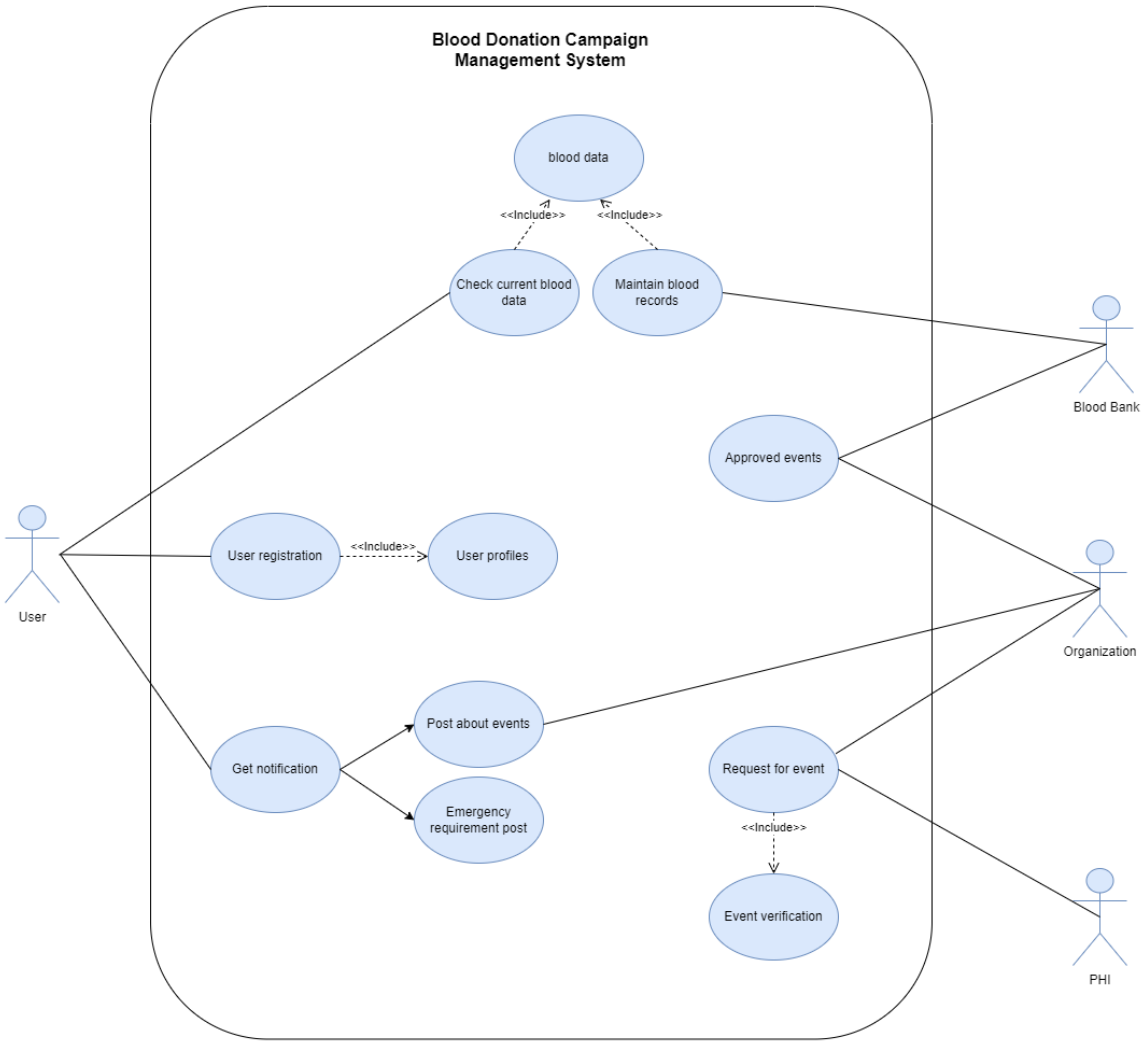
UC#8 maintain data
The blood bank authorized users can update the blood reserve, post approved blood donation camp details, and add or remove permissions for blood bank officers.

UC#9 contact
The blood bank officer can contact the new users to inform their medical checkup results, and for any emergency situation.

UC#10 event verify
PHI officer will validate the requested event primary details and inform to the blood bank about request.

UC#11 Communication
Our system can provide a forum to let users give us, the development team, feedback and report bugs.

ii. Use Case Diagram



iii. Traceability Matrix

The below table depicts the mapping of the various requirements of our system with the use cases defined previously. The requirements are given based on a scale from 1 to 10, 1 being the lowest priority and 10 being the highest priority.

REQ'T	PW	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC 9	UC10	UC11
REQ1	10	x										
REQ2	8		x		x		x		x			
REQ3	10		x	x					x			
REQ4	10			x					x			
REQ5	8				x	x				x		x
REQ6	9						x	x				
REQ7	9				x						x	
REQ8	9				x	x		x				
REQ9	9								x			
REQ10	4											x
REQ11	9	x			x	x						
REQ12	9										x	
REQ13	9	x										
REQ14	7	x										
REQ15	10	x										
REQ16	10	x		x			x					x
REQ18	10		x					x				
REQ19	10									x		x
REQ21	10	x							x	x		x
REQ22	7	x										
Maximum Weight		10	10	10	9	9	10	10	10	10	9	10
Total Weight		72	28	43	43	26	27	28	47	28	18	42

iv. Fully Dressed Description

Use Case UC#2:	New Registration
Related Requirements:	REQ-2, REQ-3, REQ-18
Initiating Actor:	Users
Actor's Goal:	To allow user to have secure login. To allow user to unregister and delete the data.
Participating Actors:	Active user, System
Precondition:	System should be active and running. User should have established connection with the system. User should meet the requirement for registering.
Postcondition:	User can enter his data in the system. User can update or delete his data as per his requirement.
Failed end condition:	User entering wrong credentials, re-enter valid credentials Unable to correct to system, check system status

Flow of events for success scenario

1. → User enters his credentials into the portal
2. ← The credentials are verified by the system.
3. → The user is allowed to access the portal.

Flow of events for extensions:
User enters invalid/out-of-bounds data.

1. ← System detects the error and sends an error message to the user prompting to enter the data again.

Use Case UC#3:	User data addition
Related Requirements:	REQ-3, REQ-4, REQ-16
Initiating Actor:	Users
Actor's Goal:	To add and update data which is within appropriate range.
Participating Actors:	Mobile Application form, System
Precondition:	Mobile Application form, System.
Postcondition:	New data should be appended to the database system.
Failed end condition:	Data input unsuccessful, check data range Unable to connect with system, check connection

Flow of events for success scenario

1. → User enters the mobile application and chooses option to input data.
2. → User inputs data.

User enters invalid/out-of-bounds data.

1. ← System detects error and sends an error message to the user asking to re-input the data.

2. → User inputs correct data.

Use Case UC#4:	User get Upcoming event details
Related Requirements:	REQ-2, REQ-5, REQ-7, REQ-8, REQ-11
Initiating Actor:	Users
Actor's Goal:	To get upcoming event details
Participating Actors:	System, Users
Precondition:	User should be active on the system
Postcondition:	User should able to check and notify upcoming events
Failed end condition:	Unable to connect with system, check connection

Flow of events for success scenario

1. → User enters the mobile application and chooses events.
2. → User can see events.

User enters invalid/out-of-bounds data.

1. ← System detects error and sends an error message to the user check connection

2. → User can see events.

Use Case UC#5:	User get timeslot for donation
Related Requirements:	REQ-5, REQ-8, REQ-11
Initiating Actor:	Users
Actor's Goal:	To get timeslot for donation
Participating Actors:	System, Users
Precondition:	User should be active on the system
Postcondition:	User should able to check timeslot for a confirmed donation camp
Failed end condition:	Unable to connect with system, check connection

Flow of events for success scenario

1. → User enters the mobile application and check event
2. → User can see timeslot for the donation camp.

User enters invalid/out-of-bounds data.

1. ← System detects error and sends an error message to the user check connection.
2. → User can see timeslot

Use Case UC#7:	User get timeslot for donation
Related Requirements:	REQ-6, REQ-8, REQ-18
Initiating Actor: Actor's Goal:	Users To request a event
Participating Actors:	PHI, System, Blood Bank,System
Precondition:	User should be active on the system User could request for a blood donation camp
Postcondition: post it to events.	User should able to check confirm of donation camp and
Failed end condition:	Unable to confirm the camp, unable to post it on events

Flow of events for success scenario

1. → User enters the mobile application and request for blood donation camp.

2. ← PHI can confirm the donation camp.
3. → User could post the event

User enters invalid/out-of-bounds data.

1. ← User couldn't post a request, PHI couldn't confirm the camp.
2. → User input corrected details.
3. → Post event successfully

Use Case UC#8:	Maintain blood bank resources
Related Requirements:	REQ-2, REQ-3, REQ-4, REQ-9, REQ-21
Initiating Actor:	Blood Bank
Actor's Goal:	Maintain the blood bank resources
Participating Actors:	System
Precondition:	Blood Bank should be active on the system Blood Bank can update the details
Postcondition:	Update the current details
Failed end condition:	Update the current details

Flow of events for success scenario

1. → Blood Bank enters the mobile application and update the details
2. ← Update successfully

User enters invalid/out-of-bounds data.

1. → Blood bank couldn't update the details correctly
2. ← System request for input correct details again.
3. → Input correct details again.
4. ← Successfully added the details.

c. System Sequence Diagrams

i. Use Case- 1

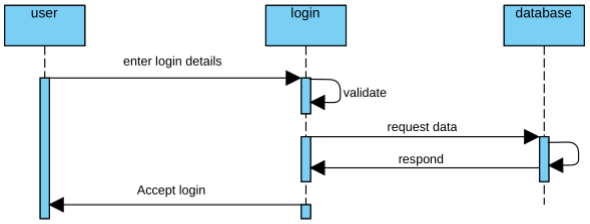


Fig 1: The above figure depicts the sequence diagram for Use Case 1, which is "User login". The user will enter his login username and password into the app, if login details are correct the user redirect to the application main dashboard.

ii. Use Case- 2

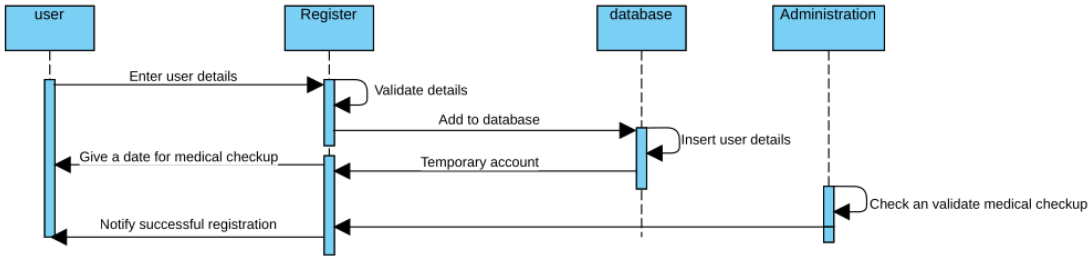


Fig 2 : The above figure depicts the sequence diagram for Use Case 2,which is "New Registration". The user will enter his credentials into the mobile app data will be verified by system and send to the database and notify medical checkup date to the new user. After pass the medical checkup then Administrator confirm the new user registration and notify to the user. Then user can access to the app and get information from mobile app.

iii. Use Case- 4 and 7

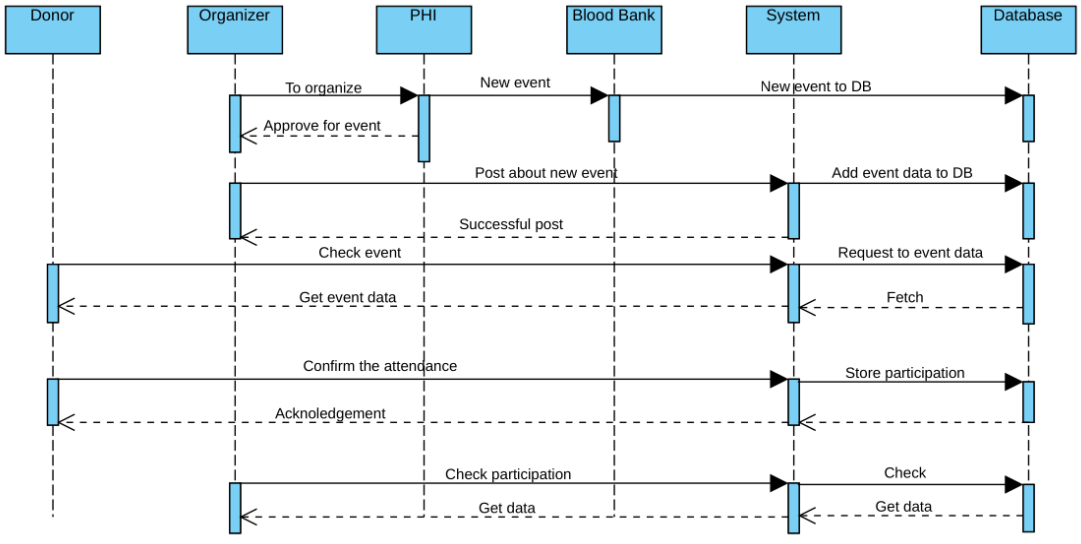


Fig 3 : Above figure shows the sequence diagram for Use Case 7,which is "Request event". The organizer inform to the PHI about new blood donation camp, then PHI inform to the

blood bank and if blood bank approve the request then PHI inform to the organizers then they can post the camp in the App.

Use case 4, which is “Upcoming events”. All donors can view and check upcoming blood donation camps using app. If user wish to participate to the camp he or she can place their participation to the organizers and organizers can get rough idea about total participants.