

**Software Requirements**

**Specification**

**for**

**Smart Blood**™

Prepared by:

**Alishba Qadir (20K-1616)**

**Areeba Batool (20K-0341) Hafsa Farhan (20K-0127)**

**Misbah Shakil (20K-0480)**

**FAST-NUCES KARACHI**

**22-10-2022**

**Table of Contents**

**1.Introduction**

[1.1 Purpose 3](#_Toc9046)

[1.2 Document Conventions 3](#_Toc9047)

[1.3 Intended Audience and Reading Suggestions 3](#_Toc9048)

[1.4 Product Scope 3](#_Toc9049)

[1.5 References 3](#_Toc9050)

[2.Overall Description](#_Toc9051)

[2.1 Product Perspective 4](#_Toc9052)

[2.2 Product Functions 4](#_Toc9053)

[2.3 User Classes and Characteristics 5](#_Toc9054)

[2.4 Operating Environment 5](#_Toc9055)

[2.5 Design and Implementation Constraints 5](#_Toc9056)

[2.6 User Documentation 6](#_Toc9057)

[2.7 Assumptions and Dependencies 6](#_Toc9058)

[3. External Interface Requirements](#_Toc9059)

[3.1 User Interfaces 6](#_Toc9060)

[3.2 Software Interfaces 8](#_Toc9061)

[3.3 Communications Interfaces 8](#_Toc9062)

[4. System Features](#_Toc9063)

[4.1 Login Portal/Logging Out 8](#_Toc9064)

[4.2 Dashboard 9](#_Toc9065)

[4.3 Blood Donation History 10](#_Toc9066)

[4.4 Book/Manage Appointmen 10](#_Toc9067)

[4.5 View and Update Test Results/View Medications 11](#_Toc9068)

[5. Other Nonfunctional Requirements](#_Toc9069)

[5.1 Performance Requirements 11](#_Toc9070)

[5.2 Safety Requirements 11](#_Toc9071)

[5.3 Security Requirements 11](#_Toc9072)

[5.4 Software Quality Attributes 12](#_Toc9073)

[6. Testing6.1 Unit Testing 12](#_Toc9074)

[6.2 Component Testing 14](#_Toc9075)

[6.3 System Testing 16](#_Toc9076)

[6.4 Load Testing 18](#_Toc9077)

[6.5 Stress Testing 19](#_Toc9078)

1. Introduction

## 1.1 Purpose

Smart Blood is a software aimed towards the healthcare industry to help patients and donors under one application. Patients will be able to evaluate their blood test reports and donors will have a platform that would allow them to complete donor related tasks.

## 1.2 Document Conventions

This document is following IEEE format. The format of the headings is as follows: Major headings are in bold 18pt font, and concurrent headings in bold 14 pt font. Sections are in the format of x.y, where x and y are real, positive integers.

## 1.3 Intended Audience and Reading Suggestions

As the software requirement specification (SRS) is written for a more general audience, this document is intended for individuals directly involved in the development of Smart Blood. This includes medical facility staff (management, information technology, quality, validation, and laboratory), auditors, software developers, project consultants, and team managers, as well as testers such as Doctors, Patients, Donors and Staff. This document need not be read sequentially, users are encouraged to jump to any section they find relevant. Anyone can use Smart Blood to donate as well as who need blood, i.e patients, hospitals, blood banks, etc.

## 1.4 Product Scope

This software will enable patients to understand their test reports and based on their reports they will be recommended with medicines that could potentially treat them. The blood reports our application will be able to evaluate are vitamins, iron, complete blood count and the basic metabolic panel.Donors will be able to make an account, book appointments, view their donor history etc. Doctors can have a benefit as the evaluation process has been decreased, doctors will be able to just take the reports and all the medicines/treatments will be listed.

## 1.5 References

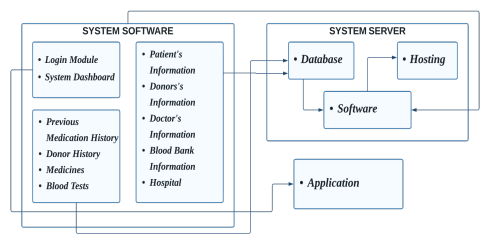
[1] IEEE Software Engineering Standard Committee, “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirement Specifications”. [2] Software Requirement Specification template. [Online]. Available:

https://cs.gmu.edu/~rpettit/files/project/SRS-template.doc .

# 2. Overall Description

## 2.1 Product Perspective

The software will be able to check the patient’s blood test report, analyze it, and then it will suggest whether the patient needs Blood or he/she can be cured by Medicines. It can be primarily used by patients, doctors, and blood banks. It is a modified version of Smart Blood Analytics where we can only get an analysis of our blood report, and you have to figure on your own what step you will take next, but in our software, the experts will guide you about your condition and it will help you to save your money.



## 2.2 Product Functions

**Login Interface** - User should enter valid username and password to get access to its profile. The system provides security features through username-password matching where only authorized users can access the system with different authorization levels.

**Donor Profile Registrations** - User will be able to see its Account No. , receipts of blood donated, need and request of blood. This allows the healthy public to register as volunteer donors.

INPUT: Donor/ Recipient Id, Name, Date of Birth, Sex, Blood Group,Address, Contact Number, Email Address, Diseases (if any). OUTPUT: Successfully Registered.

**Patient Profile Registration** - User will be able to see the information of their blood donor, report of their blood

**Blood Bank -** Blood Bank staff can manage the blood stock starting from the blood collection, to blood screening, processing, storage, transference and transfusion through this system. Each process or work-flow can be traced from the database. The system will also alert the personnel whenever the blood quantity is below its par level or when the blood in stock has expired.

## 2.3 User Classes and Characteristics

The system will support three user privileges: Users(Patients, Donors), Doctors, and Admin.

Users include patients and donors, patients and donors will have access to only their respective functionalities, doctors will have access to users and doctors information but they can only read them, and administrators will have access to everything.

The Users should be able to do the following functions:

*Patients*:

* Upload personal information.
* History of previous medications or drugs.
* Test results*Donors*:
* Upload personal information.
* History of previous medications or drugs.
* He can check how many times he has donated blood previously.

The doctor should be able to do the following functions:

* Get all donors who have previously donated the blood.● Decide whether the donor can donate or not.
* Prescribe medicines to patients based on their reports.
* Have access to the medical history of both Donors and Patients.
* Recommend hospital to patients.

The administrator should be able to do the following functions:

* Add/Delete donors information.
* Add/Delete patients information.
* Add/Delete doctors information.

## 2.4 Operating Environment

Operating environment for the Smart Blood software are as listed below:

* Database(Sql)
* Client/Server system.
* Operating System (Windows).
* Platform(Node js, React)

## 2.5 Design and Implementation Constraints

* The donor and recipient are constrained to create an account first to avail the services.
* Internet connection is also a constraint for this software.
* This software is also constrained by the database capacity so it works well with a smallernumber of donors and hospitals.
* End user's will not be able to get the information about the availability of the blood in thebank.
* Only the Admin has all the right to edit the things in the End user's profile.

## 2.6 User Documentation

1. Node JS backend technologyhttps://www.youtube.com/watch?v=TlB\_eWDSMt4 https://nodejs.org/en/docs/
2. React Frontend technologyhttps://www.youtube.com/watch?v=Ke90Tje7VS0 https://reactjs.org/docs/getting-started.html

## 2.7 Assumptions and Dependencies

* Every donor has mobile phones.
* The system does not keep the details of the gathering stock of blood.
* The donor doesn’t submit any fake reports of this system.
* The donors who want to contribute to a donation will definitely reply to the request of thesystem.
* A health practitioner or a patient can request for an actual blood group. But the request comes through blood bank authorities to the system admin. Therefore, doctor, patients are not direct customers of the system.

# 3. External Interface Requirements

## 3.1 User Interfaces

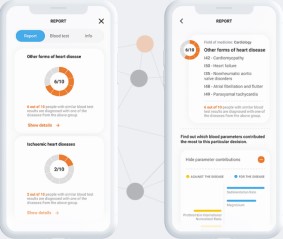
The software components that we are going to use in this project are:

* Input Controls.
* Navigation Components.
* Informational Components.
* Containers.

Tentative login portal:

The application will have basic buttons like:

* Donation History
* Personal Information
* Previous Test Results History
* Medication History
* Blood Bank locationsTentative Report Page:



How the donor information will be stored(Tentative):



## 3.2 Software Interfaces

Smart Blood can be used on the following softwares:

* Windows
* Internet Explorer
* Chrome
* Firefox

## 3.3 Communications Interfaces

There are:

* Communication functions like email for login portal will be used.
* Should have proper internet connection.
* The response time for a change will be more than 4 seconds.
* The response time for accessing the database will be no more than 5 seconds.

# 4. System Features

## 4.1 Login Portal/Logging out

**4.1.1**

### Description and Priority

This is one of the features that would reassure the user that the personal information they enter will not be changed and is extremely confidential.

**4.1.2**

### Stimulus/Response Sequences

Entering in their login credentials/signing up

**4.1.3**

### Functional Requirements

* Users will be able to type in their login credentials and if an account is not already registered they can do that as well.
* As soon as the app will be closed without pressing the ‘log out’ button our software will be designed in a way that it would automatically be logged out to ensure integrity and confidentiality.

## 4.2 Dashboard

**4.2.1**

### Description and Priority

This will act as the ‘homepage’ of the software. Here you will be able to view all the standard buttons. This will be different for a patient and donor. The patient will be able to view their test results, medications that they need, and hospitals near them in case of an emergency. Whereas the donors will be able to view buttons like donation history, blood bank locations etc. This will be at a higher priority as the dashboard would be considered as the base of our user interface.

**4.2.2**

### Stimulus/Response Sequences

Users will click onto the specific board they want to view and by the click of one button, their desired board will be displayed.

**4.2.3**

### Functional Requirements

* Users should be able to click and view all the buttons in the dashboard interface.
* Users should be able to see important notifications like next appointment,manage appointment, blood type and units donated in the dashboard.

## 4.3 Blood Donation History

**4.3.1**

### Description and Priority

This will show all the donor’s history for example; when was the last time they donated and where the blood was donated. The donation history will only be visible to the donor itself. This has a high priority as this is one of the main features of Smart Blood.

**4.3.2**

### Stimulus/Response Sequences

Users should be able to click on the donation history tab and view all their previous donations for future reference.

**4.4.3**

### Functional Requirements

➔ This is also a way for our software to track where the donor is most likely to donate blood so we can recommend blood banks near to the area where they live.

## 4.4 Book/Manage appointments

**4.4.1**

### Description and Priority

The donor will be able to book their appointment as well as make any changes as they please.This is also at a high priority.

**4.4.2**

### Stimulus/Response Sequences

Users will be able to enter in their information on when and where to donate their blood.

**4.4.3**

### Functional Requirements

* The user will be able to keep track and make sure that they do not miss their appointment.
* The system will always have a tab that would make managing their appointment easily accessible.
* The user should also be able to choose from a list of blood banks where they want to donate

## 4.5 View and update Test Results/View Medications

**4.5.1**

### Description and Priority

Patients will be able to enter in and upload their test results. The test results will then be able to determine what medications the patient has to intake, the dosage they need to take and how long it can take for the patient to reach normal blood levels.

**4.5.2**

### Stimulus/Response Sequences

Users will be able to input their test results. View all their previous results and also see what medications they will need to take.

**4.5.3**

### Functional Requirements

* The patients should be able to view all their blood test history.
* They should be able to track their medication dosage and make checkpoints in order to keep the system updated.
* This test result will also be shared with the doctor of the patient to make the decision-making process (if needed) quicker. All the doctor has to do is validate and make sure that all given medications are of use.

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

The aim we have kept for our software is that it should be efficient and work as fast as possible. The user should not have a large amount of loading time. If a user presses a tab, it should take a maximum of 2 or 3 seconds for the interface to be updated. As for user inputs, as soon as the user clicks on the text box, the keyboard should be enabled.

## 5.2 Safety Requirements

● Will try to prevent unexpected or improper shutdowns in order to prevent data loss. ● Will try to have a good memory management in the software, better error handling, network management and software development process

.

## 5.3 Security Requirements

User authentication is also one of our main priorities which is why we will be using a login portal to make sure that no unauthorized access is taking place. For example if a user closes our application without logging out, it will automatically be logged out and the user will always have to re-enter their login credentials.

## 5.4 Software Quality Attributes

Our software will be extremely correct and maintainable for the user as well as the developers. The user would be satisfied with all the results and medications that we will recommend as they would already be cross-verified by doctors users will also have the opportunity to visit and talk to our recommended doctors in-case they need any reassurance.