./

Mini Project Report – Blood bank Management System



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **Approved By** | **Remarks/Revision Details** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Document History**

# 

Contents

[Problem Statement 4](#_Toc53175852)

[Problem Description 4](#_Toc53175853)

[**Requirements** 5](#_Toc53175854)

[**High Level Requirement** 5](#_Toc53175855)

[**Low Level Requirements** 5](#_Toc53175856)

[**Hardware Requirements** 5](#_Toc53175857)

[**Software Requirements** 5](#_Toc53175858)

[Design 6](#_Toc53175859)

[Flow Diagram 6](#_Toc53175860)

[User Case Diagram 7](#_Toc53175861)

[Test Plan 7](#_Toc53175862)

[**High Level Test Plan** 7](#_Toc53175863)

[**Low Level Test Plan** 7](#_Toc53175864)

[**Quality Objectives** 8](#_Toc53175865)

[**Test Criteria** 8](#_Toc53175866)

[Suspension Criteria 8](#_Toc53175867)

[Exit Criteria 8](#_Toc53175868)

[**Test Environment** 8](#_Toc53175869)

[Github Link 9](#_Toc53175870)

[Repository 9](#_Toc53175871)

[9](#_Toc53175872)

[Conclusion 9](#_Toc53175873)

# Problem Statement

The objective of this project is to develop blood bank management system, designed to handle the daily transaction of the blood bank and to search details when required. It also helps to register the details of blood doners, blood collection details as well as blood issued reports

# Problem Description

The project aims at maintaining the information pertaining to blood donors, different blood groups available in the blood bank and help them manage in a better way, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. When a person wants to donate blood, he has to register by giving information like blood group, contact details etc.

Blood Bank management system can lead to error-free, secure, reliable and fast management systems.

**Requirements**

**High Level Requirement**

|  |  |
| --- | --- |
| **ID** | **Description** |
| H01 | To manage the operations of the blood bank |
| H02 | To maintain the database of the donors |

**Low Level Requirements**

|  |  |
| --- | --- |
| **ID** | **Description** |
| H01\_L01 | To add a new donor to the database |
| H01\_L02 | To modify the details of an existing donor |
| H01\_L03 | To search for a donor |
| H02\_L04 | To list the donors in the database |
| H02\_L05 | To delete a particular donor |

**Hardware Requirements**

* System: CPU - 2.0 GHZ
* RAM: 1 GB (Minimum)
* Hard Disk: 40 GB

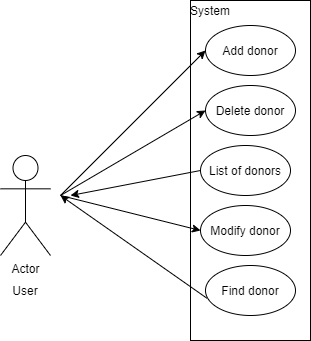
**Software Requirements**

* System: CPU - 2.0 GHZ Operating system – Any operating system
* IDE - CodeBlocks
* Language – C
* Compiler - MINGW

# Design

# Flow Diagram

# User Case Diagram



# Test Plan

## **High Level Test Plan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Description** | **Pre-Condition** | **Expected input** | **Expected Output** | **Actual output** |
| H01\_T01 | To test the operations of the menu-driven system | Choice must be one of the listed | 1,2,3,4,5,’Q’ | Specified choice is selected |  |

## **Low Level Test Plan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Description** | **Pre-Condition** | **Expected input** | **Expected Output** | **Actual output** |
| H01\_L01\_T01 | To add a donor | Choice 1 is selected | Detail of donor | Donor detail added |  |
| H01\_L01\_T02 | To delete a donor detail | Choice 2 is selected | Acc no. of the donor | Donor detail deleted |  |
| H01\_L02\_T03 | To list all donor  detail | Choice 3 is selected | No input required | List of all the donors |  |
| H01\_L02\_T04 | To modify a donor detail | Choice 4 is selected | Detail to be modified | Modify the donor detail |  |
| H01\_L02\_T05 | To find a donor by blood group | Choice 5 is selected | Blood group required | Matching donor detail |  |

## **Quality Objectives**

* Ensure the Application under Test conforms to functional and non-functional requirements.
* Bugs/issues are identified and fixed before they go live.

## **Test Criteria**

### Suspension Criteria

If more than 50% of test cases are failed then the testing is suspended until all fail cases are fixed

### Exit Criteria

100% Test coverage.

All manual test cases are executed.

All open bugs are fixed or will be fixed in the next release.

## **Test Environment**

The test cases are executed in Code Blocks.

# Github Link

<https://github.com/mariateres/105092-BloodBankManagementSystem>

# Repository

## 

# Conclusion

The Blood Bank Management System is very useful for handling the records of all donors in the blood bank. No formal knowledge is needed for the user to use this system. It also provides error messages while entering invalid data. It is very user-friendly. Blood Bank Management System, as described above, can lead to a secure, reliable, and fast management system.