

Test Plan Document

Blood Donation Management App

Project Code:

Internal Advisor:

Dr. Muhammad Ilyas

External Advisor:

Dr. Muhammad Ilyas

Project Manager:

Shumaila Zafar (BSAI51S25R028)

Project Team:

Rameen Fatima (BSAI51S25R039)

Zobia Shaheen (BSAI51S25R018)

Submission Date:

Project Manager's Signature

Document Information

Category	Information
Customer	University of Sargodha, Department of Computer Science
Project	Blood Donation Management App
Document	Test Plan
Document Version	1.0
Identifier	PGBH01-2003-TP
Status	Final
Author(s)	Shumaila Zafar, Rameen Fatima, Zobia Shaheen
Approver(s)	PM
Issue Date	Sept. 15, 2003
Document Location	
Distribution	1. Advisor 2. PM 3. Project Office

Definition of Terms, Acronyms and Abbreviations

This section should provide the definitions of all terms, acronyms, and abbreviations required to interpret the terms used in the document properly.

Term	Description
BDMS	Blood Donation Management System
SRS	Software Requirements Specifications

Table of Contents

1. Introduction.....	4
1.1 <i>Purpose of Document</i>	4
1.2 <i>Project Overview</i>	4
2. Scope of Testing	4
3. Test Plan Strategy	4
3.1 <i>Unit Testing</i>	4
3.2 <i>Integration Testing</i>	5
3.3 <i>System Testing</i>	5
4. Test Environment.....	5
5. Schedule	6
6. Control Activities.....	6
7. Functions to be Tested.....	6
8. Functions not to be Tested.....	7
9. Test Case Design and Description.....	7
10.Traceability Matrix	8
11.Major Deliverables	8
12.Risks and Assumptions.....	9
13.Exit Criteria.....	9
14.References	9
15.Appendices.....	Error! Bookmark not defined.

1. Introduction

The BDMS is designed to efficiently manage the entire process of blood donation from registration to blood distribution. This system is designed to reduce manual errors and for making timely availability of blood.

1.1 Purpose of Document

The purpose of this document is to test the BDMS, its functions and features. This document will be read by testing personnel, developers and managers.

1.2 Project Overview

The BDMS is designed to efficiently manage the entire process of blood donation from registration to blood distribution. This system is designed to reduce manual errors.

Software Usage:

- *This software will be used by donors, recipients, admins, hospitals and blood banks.*

Goals and Benefits:

- *The system aims to provide the efficient delivery of blood in normal and critical cases. It reduces manual errors, saves time and ensures timely availability of blood.*

2. Scope of Testing

The BDMS is tested only for donor-recipient management, blood group matching, data handling and admin functionalities while payments and emergency handling are out of scope.

What will be tested in BDMS:

- *Donor registration and login*
- *Recipient registration and login*
- *Admin registration and login*
- *Login authentication*
- *Adding new donors*
- *Updating donors' data*
- *Verification of users*
- *Searching for donors by*
 1. *Blood group*
 2. *City, Area*
 3. *Availability status*

What will be tested in BDMS:

- *Medical verification (Medical eligibility of donors)*
- *live connection with hospitals and blood banks*
- *Real time blood stock update*
- *Live tracking of donation*

3. Test Plan Strategy

The following test plan is therefore primarily designed to confirm that BDMS meets the requirements defined in the SRS. Manual testing has been used in this project. At each stage testing is done such that any error detected might be corrected before the next stage is approached.

3.1 Unit Testing

Definition:

Unit testing checks individual modules of the system separately. For example, registration, login and search functions.

Participants: *Project team members*

Methodology:

It follows the following approach:

- *Each module is tested separately.*
- *Both valid and invalid inputs are used.*
- *Errors are corrected before integrating modules.*

3.2 Integration Testing

Definition:

The goal of integration testing is to ensure that various modules interact properly with each other.

Participants:

Project team

Methodology:

It follows the following approach:

- *Combine related modules*
- *Check data flow between donor, recipient and admin modules*

3.3 System Testing

Definition:

System testing checks the complete system to ensure that it meets the requirements mentioned in SRS.

Participants:

Testing team

Methodology:

It follows the following approach:

- *Perform end to end test cases*
- *Restate all functional and non-functional requirements*
- *Check system behavior for normal and wrong input*

4. Test Environment

Hardware:

- *Personal computer (Desktop or Laptop)*
- *Smart Phone (Android or iOS)*

Software:

- *Window 10*
- *C++ Compiler*

Network:

- *Stable network required*

5. Schedule

Testing Activities End	Begin	
Designing Test Cases	01-12-2025	03-12-2025
Executing Test Cases	04-12-2025	05-12-2025
Unit Testing	06-12-2025	07-12-2025
Integration Testing	08-12-2025	09-12-2025
System Testing	10-12-2025	10-12-2025

6. Control Activities

- *Regular review meetings with the project advisor*
- *Test case review before execution and tracking of identical defects*

7. Functions to be Tested

This section describes all the functions of the system which will be tested to ensure they meet the specified requirements.

7.1. User registration

- *Donor registration.*
- *Receiver registration.*
- *Validation to mandatory fields: Name, ID, Blood group*

7.2. User Login and Authentication

- *Login with valid credentials.*
- *Handling invalid login attempts.*
- *Role-based access: Donors, Recipients, Administration.*

7.3. Donor Management

- *Inserting new records of donors.*
- *Updating status of donor availability.*
- *Access to donor profile information.*
- *Blood group search amongst potential blood donor.*

7.4. Receiver Management

- *Creating blood request.*
- *Status of blood request.*
- *Cancellation of blood request.*

7.5. Blood Request Management

- *Creates Blood request.*
- *Admins view all the requests.*
- *Requests acceptance or cancellation by admin.*
- *Status Updates: pending, accepted or cancelled.*

7.6. Blood Stock Management

- Showing the available blood stock.
- Updating blood stock levels.
- Check stocks by blood group.

7.7. Appointments Management

- Scheduling donor appointments.
- Displaying appointments information.

7.8. Administrative Functions

- Verification of users.
- Viewing all users.
- Generating basic system report.

8. Functions not to be Tested

Testing out of scope includes the following functions, which are either not included in the present system requirements or scheduled for future versions.

8.1. Medical verification of Donor

- No medical eligibility check.
- **Reason:** Require medical systems and expert validation.

8.2. Real-time Integration of Hospitals and Blood Banks

- No live interactions with any hospitals and blood banks.
- **Reason:** External system is not integrated into this system.

8.3. Online Payments or Financial Transactions

- No functionality for payment or billing exists.
- **Reason:** Not mentioned within the SRS and design scope.

8.4. Real time Blood Stock Updates

- The stocks are updates manually.
- **Reason:** It is out of scope with the present system to allow live tracking.

8.5. Emergency Handling and Alarms

- Emergency notifications are not included.
- **Reason:** Planned as future enhancements.

9. Test Case Design and Description

<Summarize the common attributes of test cases. This may include input constraints that must be true for every input in the set of associated test cases, any shared environmental needs, any shared special procedural requirements, and any shared case dependencies. The following scheme is recommended for describing test cases in detail.

12.1. Test Case 1

12.2. Test Case 2

.

.

12.n. Test Case n

>

< *Example for a test case template* >

Test Case ID:	Reference number	QA Test Engineer	Name of personnel
Test case Version:	Version number	Reviewed By	Testing Team lead
Test Date:	Date	Use Case Reference(s)	Relation to use cases
Revision History	Refer to previous test case identity (if any)		
Objective	Need and scope of the testing		

Product/Ver/Module	<i>Refer to overall system being built and the place of this test case in it.</i>	
Environment:	<i>Necessary and desired properties of the test environment. (hardware/software)</i>	
Assumptions:	<i>Assumptions that might affect the testing process.</i>	
Pre-Requisite:	<i>Necessary condition that needs to be fulfilled prior to the test case.</i>	
Step No.	Execution description	Procedure result
	<i>Events being tested.</i>	<i>Mention software response.</i>
Comments		
<input type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed		

10. Traceability Matrix

<Build a traceability matrix that shows relationship between use cases and test cases to ensure forward and backward traceability.>

Example for Traceability matrix template

<Test case name/Id>

Sr. no.	Requirement ID	Use Case ID	GUI ID	Test case ID	Status	Date Completed
1	Reference number of the Requirement	Reference number of the use case	Reference number of GUI (if any)	Reference number of pre-requisite use case	Complete /incomplete	dd-mm-yyyy
2						
.						
.						
n						

11. Major Deliverables

The following deliverables will be produced at the end of the testing phase of the Blood Donation Management System (BDMS):

- **Test plan**
A complete test plan defining testing scope, objectives, testing strategy, resources, schedule, risks and exit criteria for BDMS.
- **Test cases**
Detailed test cases covering all functional modules of BDMS including donor management, blood inventory, requests, authentication and reporting.

- **Test summary report**

A final report highlighting testing outcomes, quality assessment of BDMS, known issues and recommendations for release.

12. Risks and Assumptions

Assumptions:

The following assumptions were considered while planning and executing testing for BDMS:

- *Unit testing has been completed before system testing begins.*
- *The requirements document is stable, complete and approved.*
- *Testing personnel have sufficient knowledge of BDMS and basic testing components.*
- *Required tools are available during the testing phase.*
- *Test environment was suitable for testing.*

Risks:

- *Requirements' creeping.*
- *Ambiguous requirements.*
- *Limited availability of resources for testing (time, money, members).*
- *Delays in bug fixing by the development team.*

13. Exit Criteria

The testing document will be considered complete when the following conditions are met:

- *All planned test cases have been executed.*
- *All errors are fixed.*
- *All core functionalities (donor registration, blood requests, inventory management, authentication and reporting) work as per requirements.*
- *Test summary report is prepared and approved by stakeholders.*
- *The system is stable and ready for deployment.*

14. References

- *Course Instructor Guidelines*
- *Blood Donation Management System (BDMS). (2025) Software Requirements Specifications (SRS)*
- *Blood Donation Management System (BDMS). (2025) Systems Design Document (SDD)*
- *Roger S. Pressman, Software Engineering*

