DrillBit

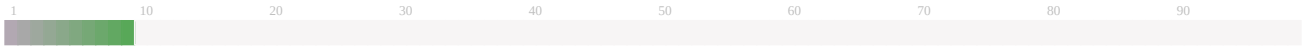
The Report is Generated by DrillBit Plagiarism Detection Software

Submission Information

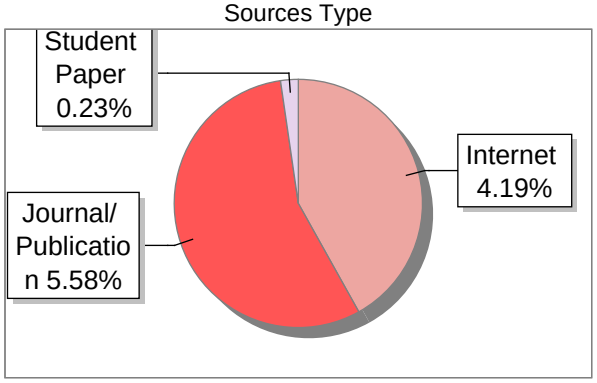
Author Name	ABDULKADIR MUMIN MOHAMED
Title	A SMART BLOOD DONATION AND EMERGENCY RESPONSE APPLICATION USING REAL-TIME GEO-TRACKING AND DONOR MATCHING ALGORITHMS
Paper/Submission ID	4708778
Submitted by	smadete@umma.ac.ke
Submission Date	2025-11-21 11:09:13
Total Pages, Total Words	80, 13808
Document type	Project Work

Result Information

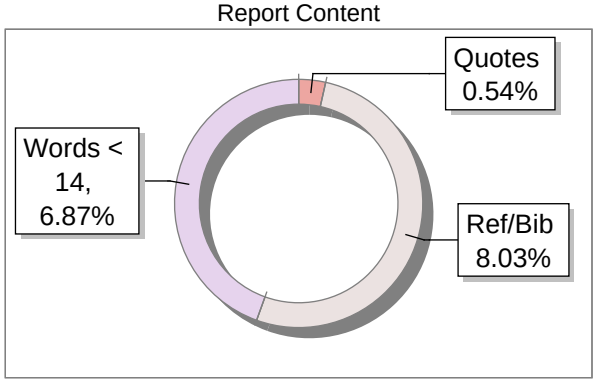
Similarity **10 %**



Sources Type



Report Content



Exclude Information

Quotes	Not Excluded
References/Bibliography	Not Excluded
Source: Excluded < 14 Words	Not Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File

DrillBit

DrillBit Similarity Report

10	109	A
SIMILARITY %	MATCHED SOURCES	GRADE

A-Satisfactory (0-10%)
B-Upgrade (11-40%)
C-Poor (41-60%)
D-Unacceptable (61-100%)

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	KCA Academic Commons	<1	Publication
2	moam.info	<1	Internet Data
3	Thesis Submitted to Shodhganga Repository	<1	Publication
4	pdfcookie.com	<1	Internet Data
5	seds.nu.edu.kz	<1	Publication
6	sifisheriessciences.com	<1	Publication
7	An Intelligent Framework for the Evaluation of Compliance with the Requirements by Andres-Jimenez-2020	<1	Publication
8	Conflict Management Styles in the Workplace A Study of First Generation Arab Mu by A-2015	<1	Publication
9	qdoc.tips	<1	Internet Data
10	cashew.icar.gov.in	<1	Publication
11	oops.uni-oldenburg.de	<1	Publication
12	www.ncbi.nlm.nih.gov	<1	Internet Data
13	IEEE 2017 IEEE International High-Level Design Validation and Test , by Barash, Guy Farchi- 2017	<1	Publication

14	formative.jmir.org	<1	Internet Data
15	ijrpr.com	<1	Publication
16	pdfcookie.com	<1	Internet Data
17	pdfcookie.com	<1	Internet Data
18	pride.periyaruniversity.ac.in	<1	Publication
19	Proceedings of the Inaugural Strategy Meeting for the Establishment o, by Viswanathan, Shanth- 2019	<1	Publication
20	repositorioslatinoamericanos	<1	Publication
21	Thesis Submitted to Shodhganga Repository	<1	Publication
22	www.intechopen.com	<1	Publication
23	epdf.pub	<1	Internet Data
24	link.springer.com	<1	Internet Data
25	moam.info	<1	Internet Data
26	Thesis Submitted to Shodhganga Repository	<1	Publication
27	Thesis Submitted to Shodhganga Repository	<1	Publication
28	The differences between adults and adolescents using a mobile health applicatio, by zcan, Habibe, Yr-2024	<1	Publication
29	e-journal.trisakti.ac.id	<1	Publication
30	frontiersin.org	<1	Internet Data
31	moam.info	<1	Internet Data
32	moam.info	<1	Internet Data

33	Submitted to Cagayan de Oro City Phinma School on 2025-11-17 12-08 4680715	<1	Student Paper
34	editorialsamarth.blogspot.com	<1	Internet Data
35	oa.upm.es	<1	Publication
36	quixy.com	<1	Internet Data
37	students.aiu.edu	<1	Publication
38	www.dx.doi.org	<1	Publication
39	www.linkedin.com	<1	Internet Data
40	www.rhinologyjournal.com	<1	Publication
41	A Mobile Vibration Laboratory Unit for Monitoring Dynamic Characteristics of Ove by Ruhlman-1959	<1	Publication
42	dl.lib.uom.lk	<1	Publication
43	eprints.rclis.org	<1	Publication
44	Food Product Traceability and Customer Value by Engelseth-2014	<1	Publication
45	MDSplus yesterday, today and tomorrow by Fredian-2018	<1	Publication
46	www.ijisrt.com	<1	Publication
47	agsalud.ageditor.org	<1	Publication
48	anucde.info	<1	Publication
49	arxiv.org	<1	Internet Data
50	arxiv.org	<1	Publication
51	cdn.unicaf.org	<1	Publication

52	digilib.k.utb.cz	<1	Internet Data
53	dspace.nm-aist.ac.tz	<1	Publication
54	en.nsd.pku.edu.cn	<1	Internet Data
55	etd.aau.edu.et	<1	Publication
56	Guidelines for Long-Term Monitoring Protocols by Kare-2003	<1	Publication
57	jurnal.ut.ac.id	<1	Internet Data
58	moam.info	<1	Internet Data
59	moam.info	<1	Internet Data
60	ojs.unkriswina.ac.id	<1	Publication
61	pdfcookie.com	<1	Internet Data
62	pediatrics.jmir.org	<1	Publication
63	REPOSITORY - Submitted to Bannari Amman Institute of Technology on 2023-02-04 12-31	<1	Student Paper
64	rootstack.com	<1	Internet Data
65	Thesis Submitted to Shodhganga Repository	<1	Publication
66	wmjonline.org	<1	Publication
67	www.academia.edu	<1	Internet Data
68	www.arxiv.org	<1	Publication
69	www.bkmediagroup.com	<1	Internet Data
70	www.slideshare.net	<1	Internet Data

71	www.slideshare.net	<1	Internet Data
72	archpublichealth.biomedcentral.com	<1	Internet Data
73	arxiv.org	<1	Publication
74	aura.abdn.ac.uk	<1	Publication
75	bmcmededuc.biomedcentral.com	<1	Publication
76	cacds.org.ua	<1	Internet Data
77	Developing a collective intelligence application for special education by Daw-2009	<1	Publication
78	discoveryalert.com.au	<1	Internet Data
79	docplayer.net	<1	Internet Data
80	docshare.tips	<1	Internet Data
81	eprints.umm.ac.id	<1	Internet Data
82	Free polyamine and polyamine regulation during pre-penetration and pen by Montilla-Bascn-2016	<1	Publication
83	gisuser.com	<1	Internet Data
84	hict.edu.vn	<1	Internet Data
85	index-of.es	<1	Publication
86	journal.unjani.ac.id	<1	Internet Data
87	mdpi.com	<1	Internet Data
88	moam.info	<1	Internet Data
89	moam.info	<1	Internet Data

90	moam.info	<1	Internet Data
91	niftypm.com	<1	Internet Data
92	ojs.mmtc.ac.id	<1	Internet Data
93	pdfcookie.com	<1	Internet Data
94	Perovskitesilicon tandem photovoltaics Technological disruption without busin By Christina Kamaraki, Matthew T, Yr-2021,8,16	<1	Publication
95	pmc.ncbi.nlm.nih.gov	<1	Internet Data
96	repository.unair.ac.id	<1	Internet Data
97	REPOSITORY - Submitted to Awadesh Pratap Singh University, Rewa on 2025-11-06 08-24 4618077	<1	Student Paper
98	Role of breakfast in hotel selection systematic review by Leite-Pereira-2019	<1	Publication
99	som.org.uk	<1	Publication
100	springeropen.com	<1	Internet Data
101	The impacts of the 28 June 2012 storms on UK road and rail transport by Jaroszweski-2014	<1	Publication
102	The SEED, by Overbeek, Ross Dis- 2004	<1	Publication
103	www.academia.edu	<1	Internet Data
104	www.acc.com	<1	Publication
105	www.dx.doi.org	<1	Publication
106	www.linkedin.com	<1	Internet Data
107	www.marketwatch.com	<1	Internet Data

108	www.scribd.com	<1	Internet Data
109	American Institute of Aeronautics and Astronautics 45th AIAAASMEAS	<1	Publication

SCHOOL OF BUSINESS AND TECHNOLOGY

BACHELOR OF SCIENCE IN COMPUTER SCIENCE

A SYSTEM PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE

TITLE: A SMART BLOOD DONATION AND EMERGENCY RESPONSE APPLICATION USING REAL-TIME GEO-TRACKING AND DONOR MATCHING ALGORITHMS

SUBMITTED BY:

ABDULKADIR MUMIN MOHAMED

ADM NO:

BSCS/2023/50449

SUPERVISOR:

MR. JEFF LUSWETI WAFUBWA

NOVEMBER 2025

DECLARATION

I, the undersigned, declare that this Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms project is the result of my own original work and has not been submitted, in whole or in part, for any other degree or qualification.

I further declare that all sources of information and assistance used in the preparation of this project have been duly acknowledged and referenced. I acknowledge that any unauthorized use or reproduction of this system, or any part thereof, may constitute an infringement of copyright.

Name: _____

Sign: _____

Date: _____

DEDICATION

This project is dedicated to my beloved family and friends, whose unwavering support and encouragement have been my greatest motivation.

I also dedicate this work to all healthcare workers, blood donors, and emergency responders whose dedication and selflessness save countless lives every day.

May this app contribute, in some way, to making their noble efforts safer, faster, and more effective.

ACKNOWLEDGEMENT

I acknowledge the invaluable contributions and support received throughout the development of the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms project. I am deeply grateful for the guidance, insights, and encouragement provided by my project supervisor, Mr. JEFFREY LUSWETI WAFUBWA, whose expertise and feedback were instrumental to the progress of this work.

I also extend my heartfelt thanks to my family, friends, and colleagues for their unwavering encouragement and support, which kept me motivated and focused throughout this endeavor.

Lastly, I express my sincere gratitude to Umma University for providing me with this opportunity and the necessary resources to bring this project to life. I am committed to further developing this application and look forward to future opportunities to expand its features and positive impact in the healthcare sector.

ABSTRACT

This document provides a comprehensive overview of the development and evaluation of the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms. The project aims to deliver an automated and efficient solution for managing blood donation activities, with a focus on streamlining donor matching, enhancing emergency response, and improving communication between hospitals, donors, and patients. The literature review examines existing blood donation systems and technologies, highlighting the need for a real-time, mobile-based platform that overcomes limitations in donor coordination and response time. It also discusses related studies and best practices in mobile health application design, user engagement, and emergency management, forming the foundation for the system's approach.

The research methodology employs the Agile model, which emphasizes iterative development, continuous testing, and user feedback throughout the process. Data collection techniques, including interviews and questionnaires, are used to analyze user requirements and system needs. These insights guide the overall design and functionality of the application to ensure efficiency, usability, and reliability.

In the system design phase, a structured framework for the application is developed, incorporating key features such as real-time geo-tracking of donors, instant emergency notifications, secure authentication, and a smart matching system that connects donors with nearby patients and healthcare facilities. The system is built using Flutter for cross-platform mobile compatibility on both Android and iOS, while SQLite is used to manage local data efficiently. The implementation phase, guided by Agile practices, results in a functional, scalable, and user-friendly mobile application.

The document concludes with recommendations for user awareness campaigns, system scalability, and integration with hospital databases to ensure long-term sustainability. Future improvements may include AI-based predictive donor matching, cloud-based data storage, and multi-language support to enhance accessibility and performance.

Contents

TABLE OF FIGURES

TABLE OF TABLES

CHAPTER ONE: INTRODUCTION

1.1 Background of the Study

Access to safe and timely blood donation is vital for saving lives in hospitals, clinics, and during emergency situations such as accidents or surgeries (WHO, 2017). Whether responding to urgent medical needs, managing routine transfusions, or supporting disaster response efforts, the challenge often lies in quickly connecting the right blood donors with patients in need (Rahman et al., 2020). Efficient donor coordination can directly influence survival rates, treatment success, and the overall responsiveness of healthcare systems (Osei & Appiah, 2021). However, traditional methods of blood donor management relying on paper records, phone calls, or static databases are inadequate in today's fast-paced and unpredictable world (Sharma & Patel, 2022).

Manual blood donation processes are prone to several inefficiencies. One major issue **is** the delay in locating and mobilizing suitable donors, which can lead to life-threatening shortages, especially during emergencies (WHO, 2017). Hospitals and blood banks often struggle to reach available donors quickly, resulting in critical time lost when every second counts. Without an efficient system to update donor availability and match compatible donors with patients in real time, healthcare providers face difficulties that can lead to postponed surgeries, treatment delays, or preventable fatalities (Rahman et al., 2020). Similarly, blood drives and donation campaigns often suffer from poor coordination and low turnout when managed manually, reducing the overall supply of safe blood units (Osei & Appiah, 2021).

Another significant challenge with traditional donor management methods **is** the lack of real-time updates and geo-location tracking. In urgent cases, knowing where eligible donors are located and whether they are available to donate can make the difference between life and death (Sharma & Patel, 2022). Without automated systems to provide instant notifications and match requests with verified donors nearby, medical staff must rely on outdated contact lists, resulting in slow response times and missed opportunities to save lives.

Moreover, the manual process of coordinating blood donations often lacks a centralized platform that can integrate the needs of donors, hospitals, and patients (Rahman et al., 2020). Donors may not know when or where their contribution is needed most, while hospitals and patients may

struggle to reach suitable volunteers quickly and securely. This lack of integration leads to fragmented communication, increased workload for staff, and underutilized donor networks. The result **is** an unreliable supply chain for one of the most critical medical resources (WHO, 2017).

Recognizing these challenges, A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms was developed to address the inefficiencies associated with manual donor management. The application provides an automated, centralized platform that connects willing blood donors with patients and hospitals in real time (Osei & Appiah, 2021). It is designed to meet the needs of various users whether individual donors, hospital staff, or patients' families by offering features such as real-time donor matching, geo-location, emergency alerts, and secure donor profiles (Rahman et al., 2020). By streamlining communication and automating matching, the application reduces delays, improves donor turnout, and ultimately helps save lives during medical emergencies (WHO, 2017).

1.2 Problem Statement

The current manual methods of managing blood donation and donor coordination lead to significant inefficiencies, including delays in locating suitable donors, poor communication, and the lack of real-time updates during emergencies. These challenges often result in critical blood shortages and preventable loss of life when urgent transfusions are needed. There is a pressing need for an automated, intelligent system that streamlines donor matching, provides real-time geo-tracking, and enables hospitals and patients to reach compatible donors quickly and reliably. The Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms was developed to address these challenges by offering a fast, transparent, and technology-driven solution.

1.3 General Objective:

To develop A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms, an integrated mobile platform that streamlines donor coordination, improves emergency response, and enhances communication between donors, hospitals, and patients.

1.4 Specific Objectives:

1. To identify the key challenges faced in manual blood donor mobilization and emergency matching processes.
2. To design and develop a user-friendly mobile app that automates donor matching, geolocation tracking, and emergency notifications.
3. To implement secure user authentication and real-time communication features to connect donors with hospitals and patients quickly.
4. To evaluate the effectiveness of the app in improving donor availability, reducing response times, and supporting critical healthcare needs.

1.5 Research Questions

The study addresses the following key research questions:

1. What are the main challenges associated with current manual methods of blood donor mobilization and emergency matching?
2. How can a mobile app be designed to automate donor matching, real-time location tracking, and emergency notifications effectively?
3. How can secure authentication and real-time communication features improve coordination between donors, hospitals, and patients?
4. Does the Blood Donation & Emergency Match App enhance donor availability, reduce response times, and support healthcare providers in emergencies?

1.6 Justification of the Study

An automated Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms offers significant benefits, including faster donor mobilization, improved communication, and enhanced readiness for emergencies. By providing real-time location tracking and smart notifications, the application ensures that hospitals, blood banks, and patients can quickly reach verified donors when blood is urgently needed. This system helps save lives by reducing delays, improving donor engagement, and building a reliable network

of active volunteers. Overall, the application simplifies the donation process, raises awareness, and strengthens community support for voluntary blood donation.

1.7 Scope

The Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms provides a reliable, centralized solution for connecting blood donors with patients and hospitals during urgent situations. It covers key features such as real-time donor matching, geolocation tracking to find nearby donors, emergency alert notifications, and secure donor profiles and contact management.

The Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms does not include full blood bank inventory management, integration with national health databases, or medical screening functionalities for donor eligibility. These advanced features are outside the scope of this system and may require integration with existing hospital or blood bank management systems if needed.

1.7.1 Limitation of the Study

The Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms focuses primarily on real-time donor matching, geolocation tracking, and emergency alert features. It does not cover full-scale blood bank inventory management, national donor registry integration, or detailed medical screening processes for donor eligibility. These advanced functions remain outside the scope of this study

1.7.2 Significance of the Study

The development of A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms can significantly impact blood donation and emergency response in the following ways:

Improved Emergency Response:
By automating and streamlining the process of locating and mobilizing blood donors, the application's features such as real-time donor matching, geolocation tracking, and instant emergency alerts can help hospitals and patients access suitable donors faster, which is critical in life-threatening situations.

Increased Donor Engagement:

The system can help increase voluntary blood donation by making it easier for donors to know when and where their donation is needed most. Real-time notifications and a user-friendly platform can encourage more people to register and remain active within the donor network.

Scalability and Community Impact:

This research provides valuable insights into developing a scalable and adaptable donor coordination system. Hospitals, blood banks, and communities of varying sizes can customize the application to meet local needs, build reliable donor networks, and strengthen overall community health outcomes.

1.8 Summary

This chapter introduced A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms, emphasizing the need for fast and efficient donor coordination to save lives. It highlighted the limitations of traditional manual systems and presented the problem that the application aims to solve.

The chapter also outlined the study's objectives, scope, limitations, and significance, demonstrating how a real-time, technology-driven solution improves communication, donor mobilization, and emergency response. These sections together provide a strong foundation for developing a user-centered mobile application. The following chapters review existing systems, describe the development methodology, and explain how the application was designed and implemented.

CHAPTER TWO: LITERATURE REVIEW

2.1 Introduction

Blood donation plays a vital role in healthcare systems worldwide, providing life-saving support for surgeries, trauma care, chronic illnesses, and emergency treatments (WHO, 2022). However, many regions continue to face donor shortages and slow response times due to outdated manual donor management systems (Rahman et al., 2020). Traditional methods often rely on paper records, static databases, or word-of-mouth networks, which are inefficient during urgent medical situations when every second counts (Patel & Sharma, 2021).

To address these challenges, researchers and healthcare providers are increasingly adopting digital solutions such as mobile applications, online platforms, and automated databases to improve donor recruitment, real-time matching, and communication (Ali et al., 2019). Technologies like geolocation, instant notifications, and smart filtering have been shown to significantly increase donor engagement and reduce response times (Chen & Zhang, 2022). Despite these advancements, many existing systems still lack integration and fail to offer a user-friendly, centralized platform that connects donors, hospitals, and patients seamlessly.

This chapter reviews the evolution from traditional donor management methods to modern automated and mobile-based systems. It examines the strengths and weaknesses of existing solutions, highlights real-world case studies, and outlines key lessons from prior research. The literature review provides the foundation for designing A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms, ensuring the proposed system addresses real needs and builds on proven best practices.

2.2 Traditional Blood Donor Management Methods

In many hospitals and blood banks, donor management is still handled through manual or partially digitalized processes. These methods often involve keeping physical donor records, static databases, and manual phone lists to contact potential donors when blood is needed (Sharma & Patel, 2021). While straightforward and low-cost, these traditional approaches have proven to be highly inefficient in today's fast-paced healthcare environment.

Manual donor matching is time-consuming and heavily depends on the availability and responsiveness of staff to reach out to donors one by one. This process can lead to significant delays, especially during emergencies when immediate transfusions are required. Moreover, donor information may be outdated or incomplete, making it difficult to verify donor eligibility or availability quickly.

Traditional systems also lack real-time communication and geo-location features. Hospitals or patients may not know which donors are closest or available at a given moment, resulting in missed opportunities to secure life-saving blood donations promptly. These limitations highlight the urgent need for smarter, automated solutions that can bridge the gap between supply and demand more efficiently.

2.3 Existing Computerized and Mobile-Based Systems

To overcome the challenges of traditional donor management, many healthcare organizations and non-profits have adopted computerized systems and mobile applications to improve donor coordination and engagement (Ali et al., 2019). These systems typically offer basic features such as digital donor databases, automated reminders, and online donor registration forms, which help reduce manual paperwork and streamline record-keeping.

Mobile-based blood donation platforms have gained popularity due to their ability to reach a wider audience through smartphones and social media integration (Chen & Zhang, 2022). Some existing apps allow donors to sign up, receive reminders for blood drives, and view their donation history. Others provide hospitals with digital tools to store donor information and send mass notifications during shortages.

However, many current systems remain limited in their ability to provide real-time, location-based donor matching during emergencies. While some platforms offer notification features, they often lack advanced geo-location services that help identify and contact the nearest available donors instantly. Additionally, issues such as outdated donor data, lack of integration with hospital workflows, and low user engagement continue to hinder the full effectiveness of these solutions.

These gaps in existing computerized and mobile-based donor systems highlight the need for a more robust, real-time solution that combines automated matching, secure data handling, and emergency communication features in one user-friendly platform.

2.4 Related Research and Case Studies

Several studies and real-world projects have explored digital solutions to improve blood donation management and emergency response. For example, the Red Cross Blood App allows users to schedule donations, receive reminders, and track donation history (American Red Cross, 2020). Similarly, mobile apps like Blood Donor Finder in India and Kenya's Blood-Link initiative have connected donors with hospitals and patients through SMS notifications and basic mobile features (Rahman et al., 2021). These solutions have proven that mobile technology can expand donor reach, raise awareness, and make donation more convenient.

However, many existing projects focus mainly on general donor registration and appointment scheduling rather than urgent, location-based donor mobilization during emergencies. Limited integration with hospital systems, outdated donor databases, and low adoption rates continue to affect the success of such platforms. These lessons highlight the importance of building user-friendly systems that combine real-time matching, reliable geo-location, and instant emergency alerts to save time when it matters most.

2.5 Gaps Identified in Existing Solutions

While digital tools have improved blood donor engagement in many regions, gaps remain in how these systems handle urgent, real-time matching for emergencies (Rahman et al., 2020; Ali & Ahmed, 2020). Many existing apps and databases do not provide instant geo-location to find the closest compatible donor or push notifications that reach donors when they are needed most urgently (Patel & Sharma, 2021). Moreover, integration with hospital workflows and secure verification of donor information are often lacking, which can lead to delays and missed opportunities (Chen & Zhang, 2022).

Another gap is donor motivation and retention. Many systems do not include features that encourage continuous donor participation, such as real-time status updates, feedback after donation, or easy ways to update availability (Ali & Ahmed, 2020). These shortcomings create an opportunity for a more advanced mobile app that bridges these gaps and provides a reliable, centralized solution for both routine donations and life-saving emergency responses (Rahman et al., 2020).

2.6 Theoretical Framework of proposed solution

The development of A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms was guided by the Technology Acceptance Model (TAM) (Davis, 1989), which explains how users come to accept and use new technologies. According to TAM, perceived usefulness and perceived ease of use are the primary factors influencing whether individuals adopt a system. For this project, the application was designed to be simple and intuitive for donors to use, while clearly demonstrating its life-saving value to encourage widespread adoption and long-term engagement.

Additionally, Emergency Response Theory emphasizes the importance of rapid information flow and effective coordination during critical incidents (Comfort, 2007). Applying this theory, the application aims to minimize delays and ensure immediate donor mobilization through features such as real-time alerts, geolocation tracking, and secure communication tools.

2.7 Conceptual Framework of proposed system

The conceptual framework for this project outlines how the application connects different users and processes to address the challenges identified in the literature. The system brings together donors who register through the app, update their availability, and receive instant notifications with hospitals and clinics that submit urgent blood requests through a verified platform. At the core of the system **is** a secure backend responsible for managing donor data, processing real-time geolocation information, and generating automated alerts. An integrated emergency match feature further enhances this process by instantly identifying suitable donors within a defined radius and notifying them to maximize response time. Through the coordinated interaction of these components, the application ensures that patients in critical need can access suitable donors quickly and efficiently, overcoming the limitations of traditional manual donor management systems.

2.8 Summary

This chapter reviewed the background and current state of blood donor management, starting with traditional manual methods and progressing to modern computerized and mobile-based systems. It discussed related research and real-world case studies, highlighting successes and ongoing challenges in donor mobilization and emergency response. The gaps identified emphasize the need for an advanced, real-time, location-based solution that bridges the shortcomings of existing tools.

The theoretical and conceptual frameworks presented provide a foundation for designing an effective Blood Donation & Emergency Match App that addresses these gaps. By combining proven theories of technology acceptance and emergency response with practical mobile features, the project aims to deliver a user-friendly, life-saving system. The next chapter describes the development methodology that guided the design, implementation, and testing of this solution.

CHAPTER THREE: RESEARCH METHODOLOGY

3.1 Introduction

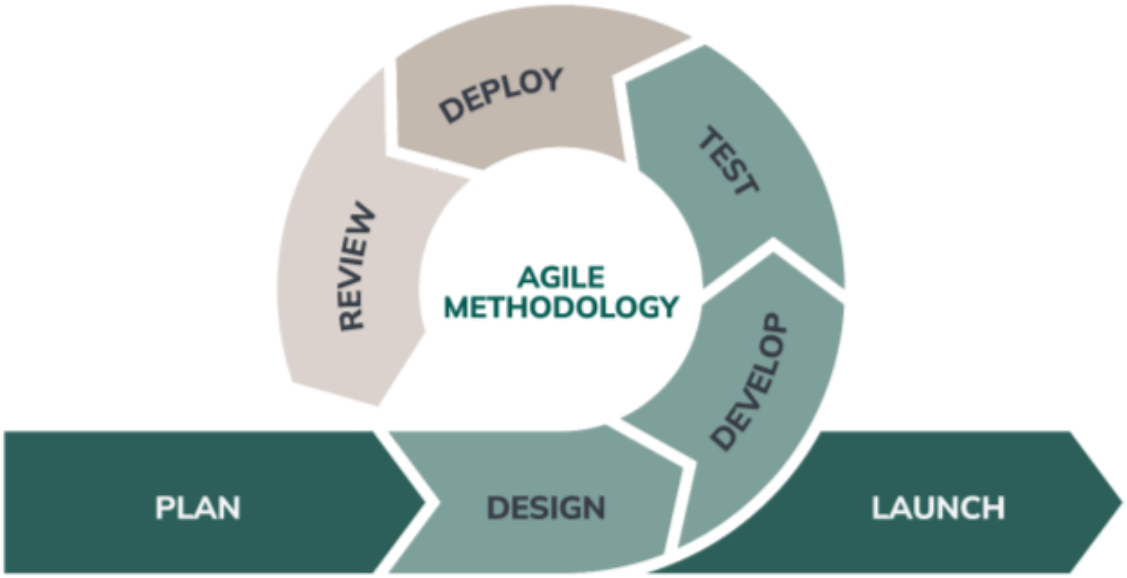
This chapter presents the software development methodology that was adopted for building A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms. Given the urgent, real-time nature of the application including features such as instant geolocation tracking, emergency notifications, and smart donor matching an approach that supports flexibility, continuous user feedback, and rapid adaptation was essential. For this reason, the Agile methodology was chosen as the guiding framework for development.

Unlike the Waterfall Model, which follows a rigid, linear sequence and becomes difficult to adjust once requirements are set, Agile promotes incremental development in small, manageable units called sprints (Beck, 2002; Pressman, 2014). This iterative approach allowed continuous testing of critical features such as secure user authentication, real-time alerts, and automated matching algorithms, ensuring they function reliably under real emergency conditions.

By applying Agile, the development team collaborated closely with key users including hospitals, blood banks, and donors to gather feedback on usability, data security, and response times. This ensured that the final product effectively addressed the challenges identified in manual donor coordination while remaining scalable, user-friendly, and responsive to the evolving needs of healthcare providers and communities.

3.2 Software Development Methodology – Agile Methodology

The development of A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms followed the Agile Methodology, which is organized into distinct phases that repeat in short, time-boxed cycles called sprints. Each phase played a critical role in ensuring that the application was developed efficiently, met user requirements, and adapted to feedback throughout the project lifecycle.



1 Figure 3.2 Agile Methodology

3.2.1 Plan

The planning phase is the foundation of the Agile development process. For A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms, this phase focused on gathering and analyzing user requirements to ensure the application addressed real-world challenges in blood donor coordination and emergency response.

Key activities in this phase included conducting stakeholder interviews with hospital staff, blood bank managers, and potential donors to understand their needs, expectations, and pain points. Questionnaires and surveys were also distributed to gather broader input on required features and usability.

Based on the information collected, a detailed product backlog was created and prioritized. This backlog included critical features such as donor registration, geolocation tracking, real-time matching, emergency alerts, and secure data management. Planning also defined clear timelines, assigned responsibilities, and identified the resources needed for each sprint.

Following Agile best practices, the plan remained flexible, allowing adjustments as new feedback and requirements emerged throughout development (Cohn, 2005).

3.2.2 Design

The design phase transformed the requirements gathered during planning into detailed system specifications and user interface concepts. For A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms, the design focused on creating an intuitive and user-friendly interface that enables donors, hospitals, and emergency responders to interact with the system effectively.

Key activities in this phase included developing wireframes and prototypes that illustrated the layout and flow of the application's main features, such as user registration, donor profile management, geolocation tracking, emergency request creation, and push notifications.

Designing secure data structures for storing sensitive donor and hospital information was also prioritized to ensure compliance with privacy and security standards. During this phase, user feedback was incorporated through prototype testing sessions to refine the look, feel, and usability of the application.

The design phase ensured that the final system not only met functional requirements but also delivered a positive user experience, which was critical for encouraging widespread adoption and long-term use (Pressman, 2014).

3.2.3 Development

The development phase involved transforming the approved designs and system specifications into a working application. For A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms, this phase was carried out in short, iterative sprints to ensure that new features were built, tested, and refined continuously.

Key activities during development included coding the application's core modules, such as donor registration, user authentication, real-time geolocation services, emergency blood request processing, and automated push notifications. The development team also integrated secure database systems to store and manage donor and hospital information reliably.

Throughout each sprint, developers collaborated closely with stakeholders to test newly developed features and gather immediate feedback. This iterative process allowed issues to be identified and resolved early, reducing the risk of costly rework later on.

3.2.4 Test

The testing phase was a critical part of the Agile development process, ensuring that each component of A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms functioned as intended and met user requirements. Testing was performed continuously throughout the development sprints to detect and fix issues early.

Key activities included unit testing of individual modules such as donor registration, login functionality, and geolocation services to verify that each component worked correctly in isolation. Integration testing confirmed that the different modules worked together seamlessly; for example, ensuring that emergency requests correctly triggered notifications and matched with available donors in real time.

User acceptance testing (UAT) was conducted with stakeholders, including hospital staff and potential donors, to confirm that the application was user-friendly and performed reliably under real-world conditions. Any bugs or usability issues identified during testing were addressed in subsequent sprints, following Agile's continuous improvement approach.

This rigorous, iterative testing process helped deliver a secure, high-performing, and trustworthy system that fulfilled its life-saving purpose (Pressman, 2014).

3.2.5 Deploy

The deployment phase marked the transition of A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms from development to actual use. In Agile, deployment occurred in multiple stages, with incremental releases ensuring that each version of the application was functional, stable, and ready for real-world operation.

During this phase, the application was deployed to a live environment such as mobile app stores or institutional servers depending on the chosen distribution strategy. Pilot testing was conducted in collaboration with selected hospitals and registered donors to monitor system performance, gather real-world feedback, and identify any remaining issues.

Deployment activities also included configuring security measures, ensuring compliance with data protection guidelines, and establishing systems for regular updates and maintenance. Continuous

deployment practices allowed the development team to roll out new features and improvements efficiently based on ongoing user feedback.

A successful deployment phase ensured that the application was accessible when users needed it most and performed reliably in delivering urgent, life-saving support.

3.2.6 Review and Launch

The review phase was an essential part of each Agile sprint and focused on evaluating the work completed to ensure it aligned with user requirements and project goals. For A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms, reviews took place at the end of every development cycle.

Key activities in this phase included presenting newly developed features or modules to stakeholders such as hospital representatives and selected donors to demonstrate how the system worked in practice. Feedback was gathered on functionality, usability, performance, and any areas needing improvement.

The review phase provided an opportunity for open discussion about what was working well and what needed adjustment before moving forward. This ongoing feedback loop allowed the team to refine existing features, adjust priorities in the product backlog, and plan the next sprint effectively.

Regular reviews ensured that the final application remained user-centered, practical, and capable of addressing real challenges in emergency blood donation and donor mobilization.

3.3 Feasibility Study

The feasibility study evaluates whether A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms can be practically developed and successfully implemented within the available resources and constraints. This section examines the project's technical, economic, and operational feasibility to confirm that it is viable before committing significant time and effort (Pressman, 2014; Sommerville, 2016).

3.3.1 Technical Feasibility

Technical feasibility assesses whether the proposed system could be developed using the available technology, expertise, and resources. A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms followed the Agile Methodology, a proven approach for building complex and evolving software applications (Pressman, 2014). The following aspects demonstrate the project's technical feasibility:

Technology Stack:

The project utilized reliable and widely used technologies, such as Flutter (for cross-platform mobile development), Firebase or a secure cloud database for real-time data management, and the Google Maps API for geolocation services. These technologies are well-documented, scalable, and suitable for building modern mobile applications with real-time capabilities.

Development Team Expertise:

The development team possessed the required skills and experience in mobile app development, real-time data handling, and secure cloud integrations. This technical expertise ensured that the system could be developed efficiently, securely, and within the required timelines.

Infrastructure Requirements:

The necessary infrastructure such as secure cloud storage, GPS-enabled devices, and standard mobile network access was readily available and cost-effective. This minimized technical risks during the development, testing, and deployment phases.

3.4 Data collection methods

To ensure that A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms met the real needs of its users, several data collection methods were used during the research and development phases:

Questionnaires:

Questionnaires were designed and distributed to potential blood donors, hospital staff, and blood bank managers to gather primary data on their needs, challenges, and expectations. These questions focused on current donor mobilization practices, emergency response gaps, and desired app features such as geo-location tracking, notifications, and data privacy.

Interviews:

Structured and semi-structured interviews were conducted with key stakeholders, including medical professionals, emergency response coordinators, and representatives from blood donation organizations. These interviews provided deeper insights into practical challenges, workflows, and how the app could fit into existing systems.

Direct Observation:

Direct observation was used where possible to study how blood donation drives and emergency blood requests were currently managed. Observing real processes helped identify inefficiencies and pain points that the app needed to address.

The data collected through these methods guided the design, development, and refinement of the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms to ensure that it was practical, user-friendly, and effective in saving lives.

3.5 Summary

This chapter outlined the Agile Methodology used to develop A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms. It explained the main Agile phases planning, design, development, testing, review, and deployment and showed how each contributed to building the system.

The feasibility study confirmed that the project was technically, economically, and operationally viable. Additionally, data collection methods such as questionnaires, interviews, and observations ensured that user needs directly shaped the system's features.

By applying a structured and user-centered approach, the project delivered a responsive mobile application designed to support real-time donor matching and improve emergency blood donation.

CHAPTER FOUR: REQUIREMENT ANALYSIS

4.1 Introduction

Requirement analysis is a vital stage in the software development life cycle, serving as the foundation upon which the system's architecture, design, and implementation are built. This phase focuses on identifying, defining, and documenting the specific needs of stakeholders to ensure that the final system fulfills both functional and non-functional requirements. For A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms, this stage is particularly critical because the application directly supports life-saving processes where accuracy, reliability, and speed are essential. Through requirement analysis, developers and researchers gain a clear understanding of how hospitals, donors, and emergency response units interact with the blood donation process, enabling the creation of a system that addresses real-world challenges such as delayed donor mobilization, miscommunication between healthcare facilities, and inefficient data tracking. The outcome of this phase ensures that the application's features such as real-time donor matching, emergency notifications, and geolocation tracking align with the practical demands of its users.

To achieve this, a combination of data-collection methods, including interviews, surveys, and questionnaires, was employed to gather relevant information from key stakeholders such as hospital staff, blood donors, and emergency service coordinators. These techniques provided insights into existing manual processes, user preferences, and the technological gaps that the proposed system must bridge. The information gathered was analyzed to derive the functional and technical requirements of the system, ensuring that the final product would be user-friendly, secure, and adaptable to various operational contexts. Furthermore, this chapter outlines the essential software and hardware requirements necessary for the smooth operation of the system across different devices. Overall, the requirement analysis phase lays the groundwork for developing a robust, responsive, and efficient mobile application that not only facilitates faster emergency responses but also strengthens coordination between donors and healthcare institutions.

4.2 Information Gathering

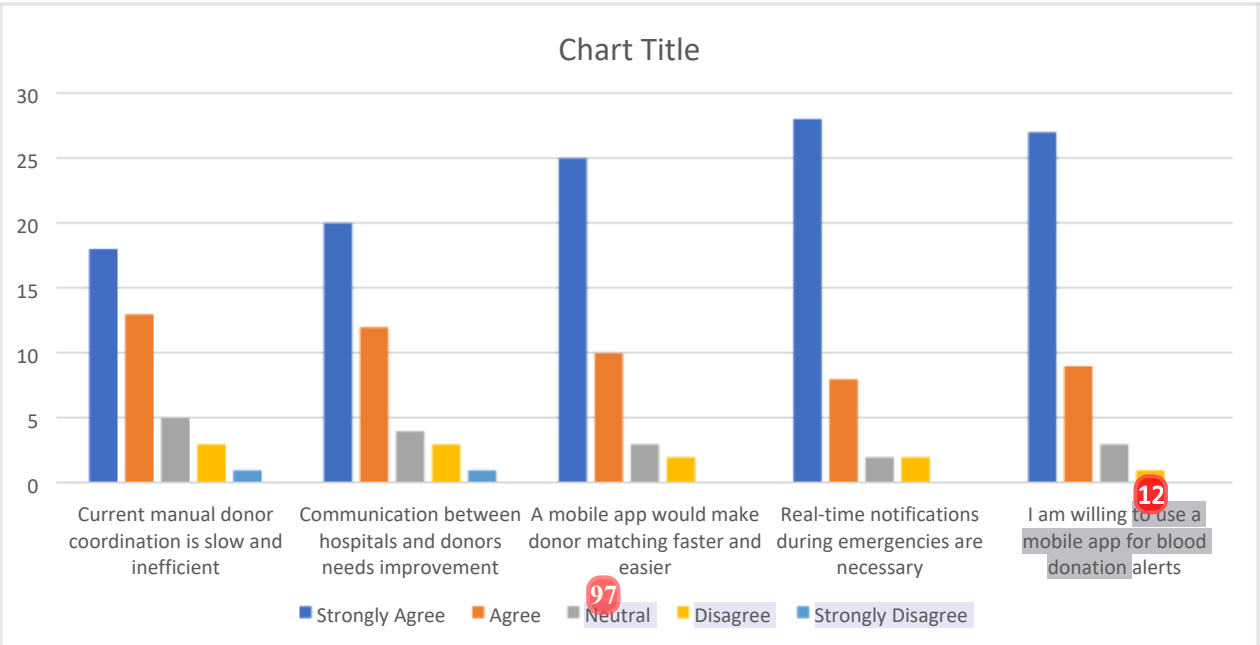
Information gathering is a fundamental step in defining accurate and realistic system requirements for A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms. This stage involved collecting detailed insights from key stakeholders hospital staff, blood donors, and emergency service coordinators to understand the limitations of existing manual systems and identify user expectations for a digital solution. Data was gathered through questionnaires, interviews, and surveys to obtain both qualitative and quantitative information regarding current donor coordination practices, response times, and communication challenges during emergencies. The collected data was analyzed to reveal key patterns, operational gaps, and feature priorities that informed the design of the system. This process ensured that the final application would not only be technologically efficient but also user-centered, secure, and responsive to the real needs of healthcare providers and blood donors.

To capture accurate and practical requirements, multiple data collection techniques were employed. Each method involved participants from key stakeholder groups, including hospital staff, blood bank personnel, and registered donors. These methods allowed for triangulation of data, enhancing both the validity and reliability of the findings.

4.2.1 Questionnaires

Questionnaires were distributed to key stakeholders, including hospital staff, registered blood donors, and emergency response coordinators, to collect quantitative data on their experiences with and expectations of the current blood donation process. The questions focused on donor coordination efficiency, communication speed during emergencies, and willingness to adopt a digital platform for donor–hospital interaction. A total of 40 responses were received 20 from hospital staff, 15 from active donors, and 5 from emergency response workers.

OUTCOME of the questionnaires submitted to the Stakeholders



2 Figure 4.2.1 Questionnaires

The bar chart illustrates respondents' feedback on the current blood donor coordination process. A majority of participants agreed that manual coordination methods are inefficient and that communication between hospitals and donors requires improvement. Nearly all respondents expressed strong support for a mobile-based system, emphasizing that real-time notifications and automated matching would significantly reduce response times during emergencies. The overall outcome indicates a high level of readiness and willingness among both healthcare professionals and donors to adopt a digital platform, validating the need for developing the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms.

4.2.2 Interviews

Interviews were conducted with selected stakeholders to gain a deeper understanding of the challenges and operational limitations surrounding the current blood donation coordination process. A total of five participants were interviewed, including two hospital staff members, two

active blood donors, and one emergency response officer. The interviews aimed to collect qualitative insights on existing communication procedures, donor availability, and response efficiency during urgent medical situations. The information obtained through these interviews was instrumental in shaping the design and functionality of A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms, ensuring it responds effectively to real-world needs.

Table 1 Interview Questions and Responses

Question	**Hospital Staff Response**	**Blood Donor Response**	**Blood Recipient Response**
How do you currently contact donors during emergencies?	"We usually rely on phone calls and contact lists, which often take too long to reach suitable donors."	"Most of the time, I get requests through messages or calls, but sometimes they come late."	"Sometimes I rely on hospital staff to reach out, but it often takes long before a suitable donor is found."
What are the main challenges you face in the current system?	"The biggest issue is delays and outdated donor information."	"I often don't know when or where donations are needed."	"It is difficult to quickly locate donors with the right blood type, especially during urgent situations."
Would a mobile app improve the coordination process?	"Yes, it would help us contact verified donors quickly and manage emergencies better."	"It would make donating more convenient and faster to respond."	"Yes, it would help reduce delays and make it easier to find donors fast during emergencies."

What features would you like to see in the proposed system?	"A live donor location tracker and automated emergency alerts."	"Instant notifications and donation reminders."	"Real-time donor availability updates and immediate alerts when a matching donor is nearby."

The interview responses revealed consistent challenges with manual donor coordination, primarily delays in communication and difficulty locating available donors in real time. Hospital staff highlighted the inefficiency of relying on static contact lists, while donors expressed frustration with inconsistent notifications and lack of real-time updates. Emergency personnel emphasized the need for integrated systems to streamline coordination between hospitals and responders. Overall, participants strongly supported the introduction of a mobile application with features such as donor geo-location, real-time notifications, and automated alerts. These findings confirm that the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms app would significantly improve communication efficiency, donor mobilization, and emergency responsiveness.

4.2.3 Direct Observations

Direct observations were conducted at selected hospitals and emergency response units to examine how donor coordination and emergency blood requests are handled in real-world conditions. This method provided a first-hand understanding of the existing workflow, response procedures, and the interaction between hospital staff, donors, and emergency coordinators. Observations focused on how requests for blood are initiated, the communication channels used to contact donors, and the time taken to identify and reach suitable donors during emergencies.

It was observed that the current donor coordination process is largely manual and dependent on phone calls, social media messages, and locally maintained contact lists. While these methods are functional, they are highly time-consuming and unreliable, especially in critical emergencies. Hospital staff often spend considerable time confirming donor eligibility and availability, while donors sometimes receive requests too late to respond effectively. Additionally, there is limited

integration between hospital databases and emergency communication systems, leading to delays and duplicated efforts when searching for compatible donors.

Operational efficiency was also affected by the lack of real-time tracking and centralized record management. Many hospitals rely on paper-based documentation or static spreadsheets to store donor information, resulting in outdated data and increased chances of error. Furthermore, emergency personnel reported challenges in locating donors within proximity to accident sites or urgent care centers due to the absence of geo-location tools. These observations highlight the need for an automated platform that provides real-time donor matching, secure data storage, and instant communication between hospitals, donors, and responders.

In conclusion, the direct observations validated the findings from the questionnaires and interviews, demonstrating that current donor management practices are insufficient for timely emergency response. The results emphasize the importance of developing the A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms App to streamline donor identification, enhance communication, and improve the overall efficiency of blood donation and emergency coordination.

4.3 System Specification

To ensure the successful development, deployment, and performance of A Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms, it was essential to define both the software and hardware requirements that support the system's functionality. The specifications outlined in this section provided a stable environment for the application's development, testing, and operation across multiple devices. The system was built using modern technologies to ensure scalability, security, and efficiency while maintaining compatibility with commonly used platforms such as Android and iOS.

Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms utilizes an integrated technology stack comprising Flutter for cross-platform mobile development, Firebase for real-time data storage and authentication, SQLite for local offline data management, and the Google Maps API for donor geolocation and tracking services. This combination enables real-time synchronization, fast performance, and seamless data exchange between users and the backend. The application operates in a cloud-based environment,

ensuring that donor and hospital data are accessible securely from different locations, with end-to-end encryption to maintain privacy and compliance with data protection standards.

The hardware requirements were designed to be minimal, ensuring that the system runs efficiently on standard mobile devices and computers used by hospitals and coordinators. The specifications below detail the minimum and recommended requirements necessary for optimal system performance.

4.3.1 Software Requirements

Table 2 Software Requirements

Component	Description
Development Environment	Android Studio / Visual Studio Code
Programming Framework	Flutter SDK (for cross-platform mobile development)
Database	Firebase (Cloud Database) and SQLite (Local Storage)
Backend Services	Firebase Authentication, Cloud Firestore, and Firebase Hosting
Mapping API	Google Maps API for real-time donor tracking
Operating System	Windows 10 or higher / macOS 10.14+
Version Control	Git and GitHub for source code management
Testing Tools	Flutter Testing Framework, Postman (for API testing)

4.3.2 Hardware Requirements

Table 3 Hardware Requirements

Component	Minimum Requirement	Recommended Specification
Processor	Dual-Core 1.8 GHz	Quad-Core 2.5 GHz or higher
RAM	4 GB	8 GB or higher
Storage	64 GB	128 GB SSD or higher
Display	13-inch HD display	15-inch Full HD or higher
Internet Connectivity	Stable 4G or Wi-Fi connection	High-speed Wi-Fi / Broadband
Mobile Devices	Android 9 (Pie) / iOS 12	Android 11+ / iOS 14+

In summary, the system specification defines the technological foundation required to build, test, and run the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms App. The combination of Flutter, Firebase, and Google Maps API ensures that the system will operate efficiently across devices, providing users with an intuitive interface, real-time data synchronization, and reliable emergency communication. These requirements ensure that hospitals, donors, and coordinators can utilize the system seamlessly, fostering faster response times and improved blood donation management.

4.4 Summary

This chapter presented the requirement analysis for the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms App, outlining the methods used to collect and define user and system requirements. Through questionnaires, interviews, and direct observations, valuable insights were gathered from key stakeholders such as hospital staff, blood donors, and emergency response officers. The findings revealed major challenges in the existing manual donor coordination process, including delayed communication, inefficient donor tracking, and lack of real-time updates during emergencies.

These findings formed the foundation for identifying the system's functional and non-functional requirements.

The system specifications detailed in this chapter define the technological and hardware requirements necessary for the development and efficient operation of the proposed application. By integrating tools such as Flutter, Firebase, and Google Maps API, the system provides a secure, user-friendly, and responsive platform that enhances donor mobilization and emergency preparedness. The requirement analysis phase has therefore established a clear framework for designing and implementing a practical, scalable, and life-saving solution. The next chapter will focus on the System Architecture and Design, detailing how these requirements are transformed into a functional model that supports all core features of the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms App.

CHAPTER FIVE: SYSTEM DESIGN

5.1 Introduction

This chapter presents the architecture and design of the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms App. It provides a detailed overview of how the system's components interact, the design methodologies used, and how the system ensures efficiency, scalability, and usability. The system architecture defines the structural framework that supports communication between the client and the database, while the system design focuses on the functional flow, interaction between entities, and database organization. Diagrams such as class, use case, activity, sequence, and entity relationship diagrams are used to illustrate the logical and operational structure of the system.

5.2 System Architecture

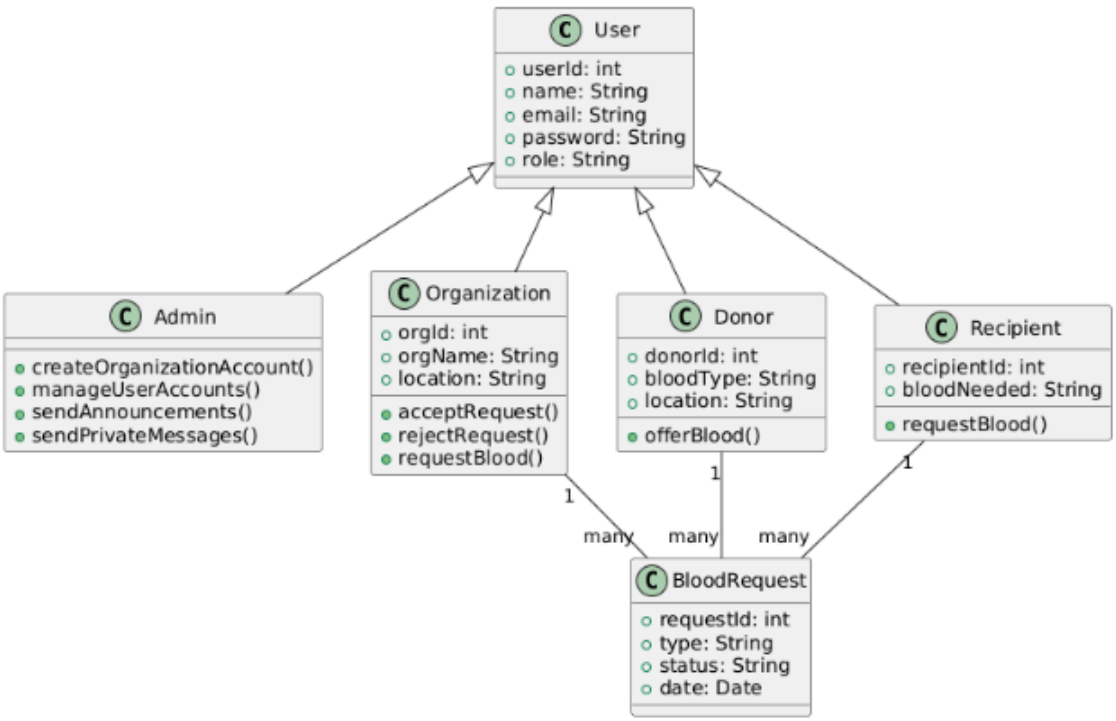
The system architecture follows a three-tier architecture consisting of the presentation layer, business logic layer, and data layer. The presentation layer is developed using Flutter to provide a cross-platform mobile interface for Android and iOS users. The business logic layer handles user authentication, donor matching, and emergency notifications, ensuring that user requests are processed efficiently. The data layer uses SQLite as the local database to store donor and patient information, blood types, and emergency requests. This architecture promotes modularity, scalability, and maintainability, allowing for future enhancements such as cloud integration or API-based data exchange with hospital databases.

5.3 System Design

The system design defines how each component of the Blood Donation & Emergency Match App interacts to achieve the desired functionality. It translates user requirements into structured system components and visual representations. The design includes class, use case, activity, sequence, and entity relationship diagrams, which help developers understand data flow, user interaction, and system logic. Additionally, wireframes and interface layouts are developed to visualize the application's user interface before implementation.

5.3.1 Class Diagram

The class diagram represents the static structure of the system by illustrating the system's classes, their attributes, methods, and relationships. It identifies key entities such as User, Donor, Patient, Request, and Hospital, and shows how they interact through inheritance and associations. This helps developers understand how the data is organized and how objects relate to one another within the system.



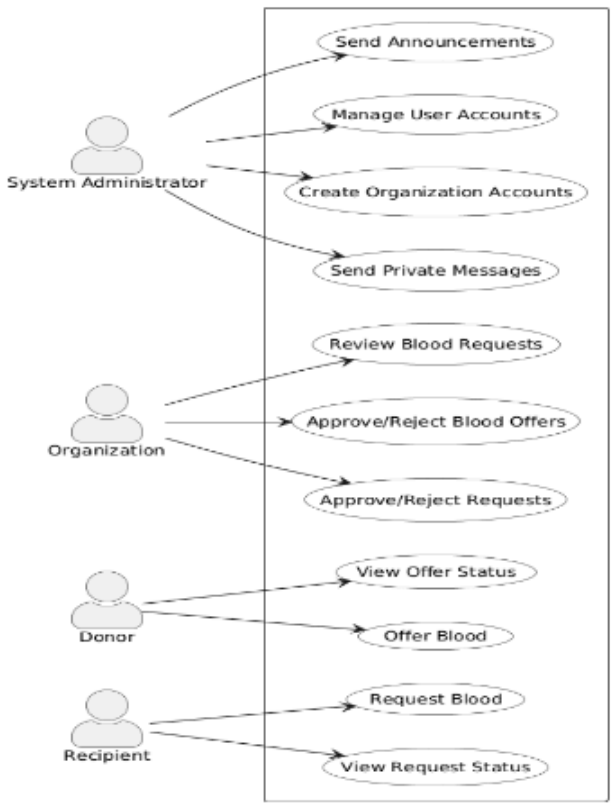
3 Figure 5.3.1 Class Diagram

This class diagram shows the four user types that extend from a main User class. Admins manage accounts and send messages, organizations handle blood requests, donors offer blood, and recipients request blood. Blood-Request connects recipients, donors, and organizations.

5.3.2 Use case diagram

The use case diagram illustrates the functional interactions between users and the system. The main actors include Donor, Patient, Hospital Administrator, and System Admin. Each actor interacts with the system to perform various tasks such as registering, logging in, sending or

receiving emergency alerts, and managing donor data. This diagram provides a high-level overview of the system's functionality from the user's perspective.

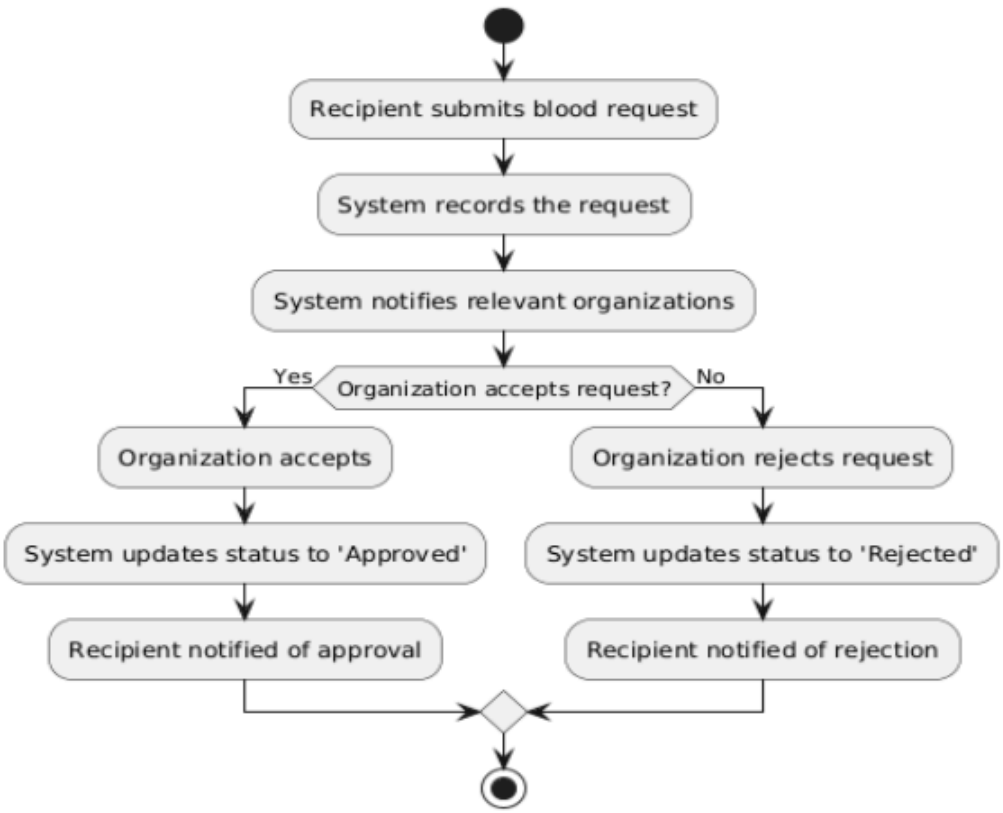


4 Figure 5.3.2 Use case diagram

This use case diagram presents what each user type can do. Admin is the most powerful actor; organizations approve/reject blood-related activities; donors offer blood; recipients request blood.

5.3.3 Activity Diagram

The activity diagram represents the dynamic behavior of the system by showing the flow of activities from start to end. It demonstrates how a blood request is initiated, processed, and completed. This helps visualize the workflow of the emergency matching process.

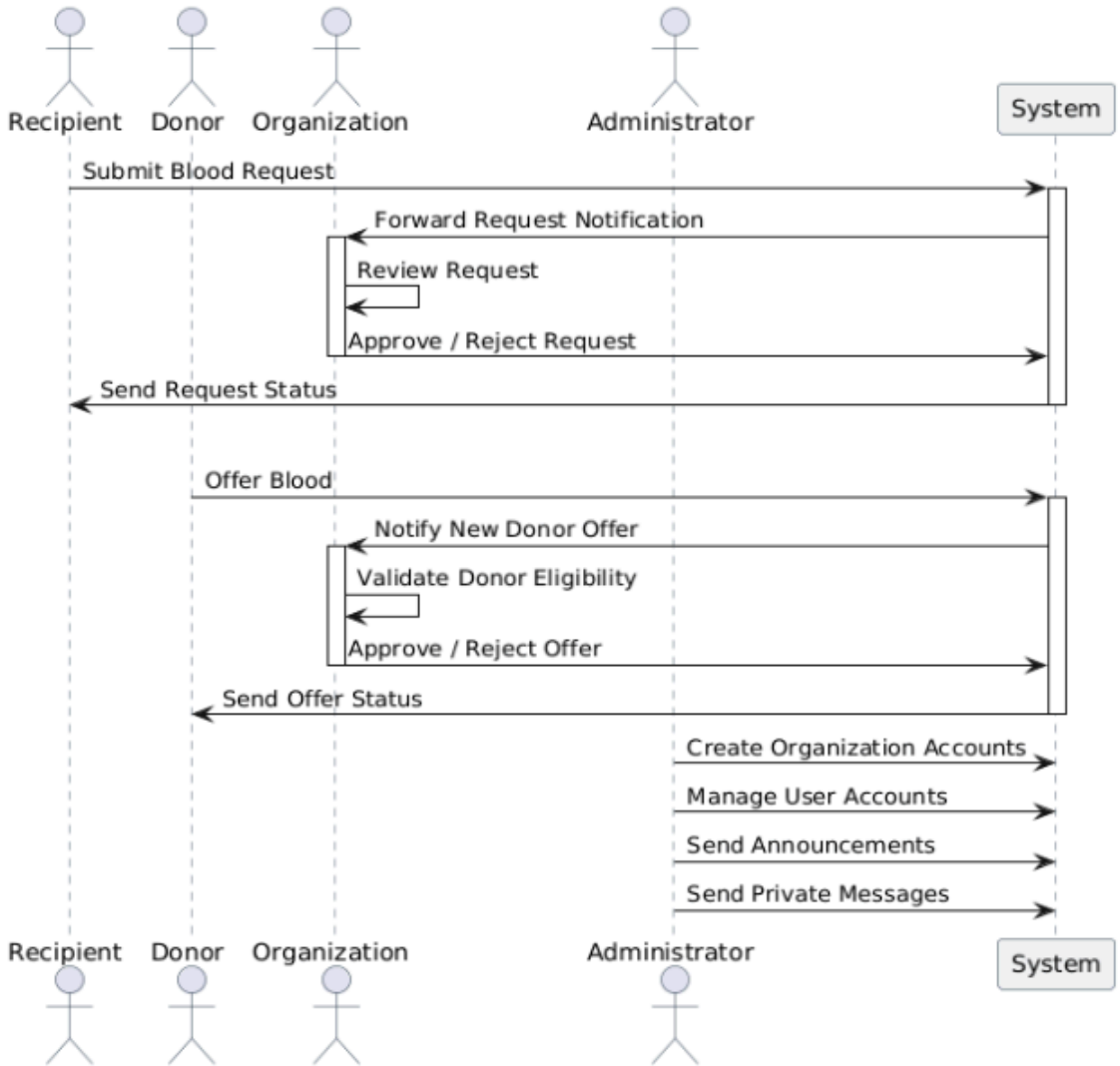


5 Figure 5.3.3 Activity Diagram

This activity diagram shows the workflow of a recipient's blood request. Organizations decide to accept or reject, and the system notifies the recipient.

5.3.4 Sequence Diagram

The sequence diagram illustrates how objects interact in a time sequence to fulfill a blood request. It shows the communication between Patient, System, Donor, and Hospital, detailing the order of messages exchanged to complete a transaction.

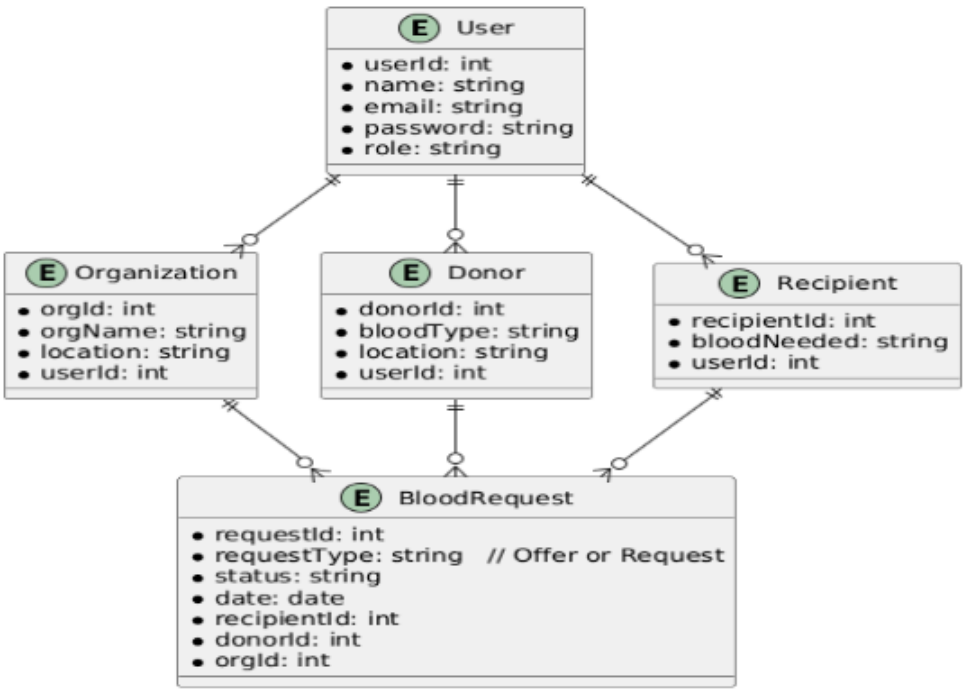


6 Figure 5.3.4 Sequence Diagram

The sequence diagram shows time-ordered interactions: donor offers blood, organization reviews it, and the system informs the donor of the result.

5.3.5 Entity Relationship Diagram

The entity relationship diagram defines the logical structure of the database by showing entities, their attributes, and relationships. Key entities include User, Donor, Patient, Request, and Hospital. Relationships are established through foreign keys to ensure data consistency and integrity across the database.



7 Figure 5.3.5 Entity Relationship Diagram

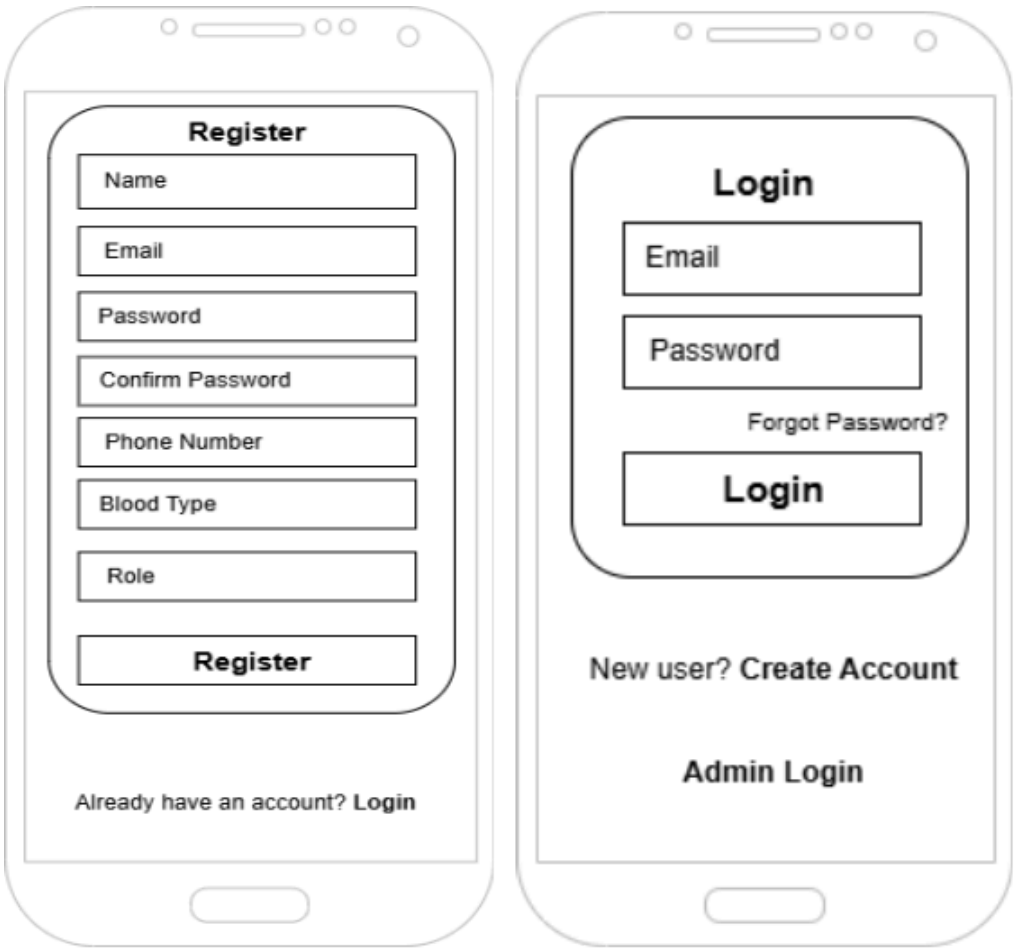
The ERD maps out the database structure. All user types inherit from User, and the Blood Request table connects all parties donors, recipients, and organizations.

5.4 Wireframes

The following wireframes illustrate the core interfaces and their layouts:

5.4.1 Sign up and Login Page Wireframes

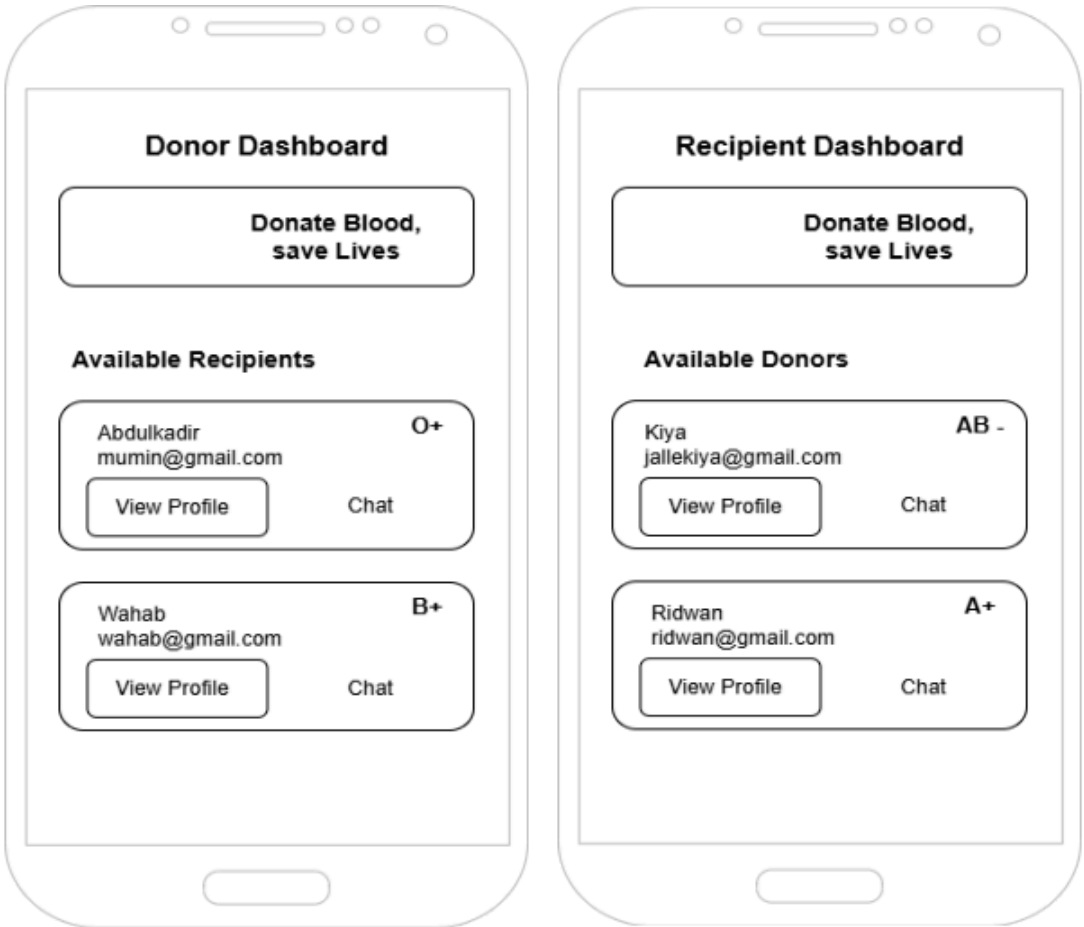
These pages allow users to register new accounts and log in to the system using their authorized credentials.



8 Figure 5.4.1 Sign up and Login Page Wireframes

5.4.2 Donor and Recipient Dashboard Wireframes

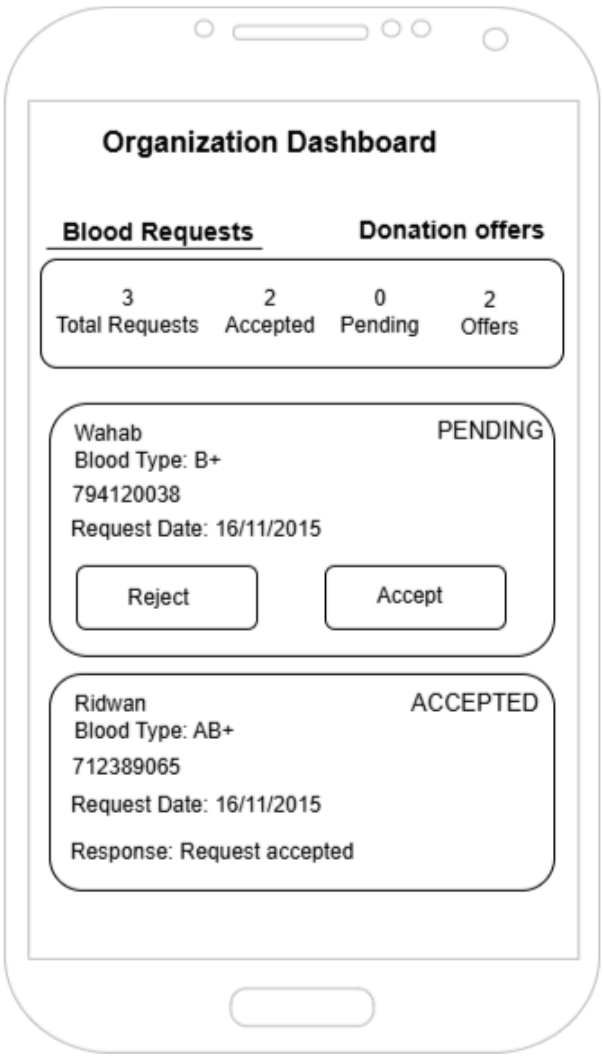
These dashboards enable donors to offer blood and recipients to request blood while monitoring the status of their submissions in real time.



9 Figure 5.4.2 Donor and Recipient Dashboard Wireframes

5.4.3 Organization Dashboard Wireframe

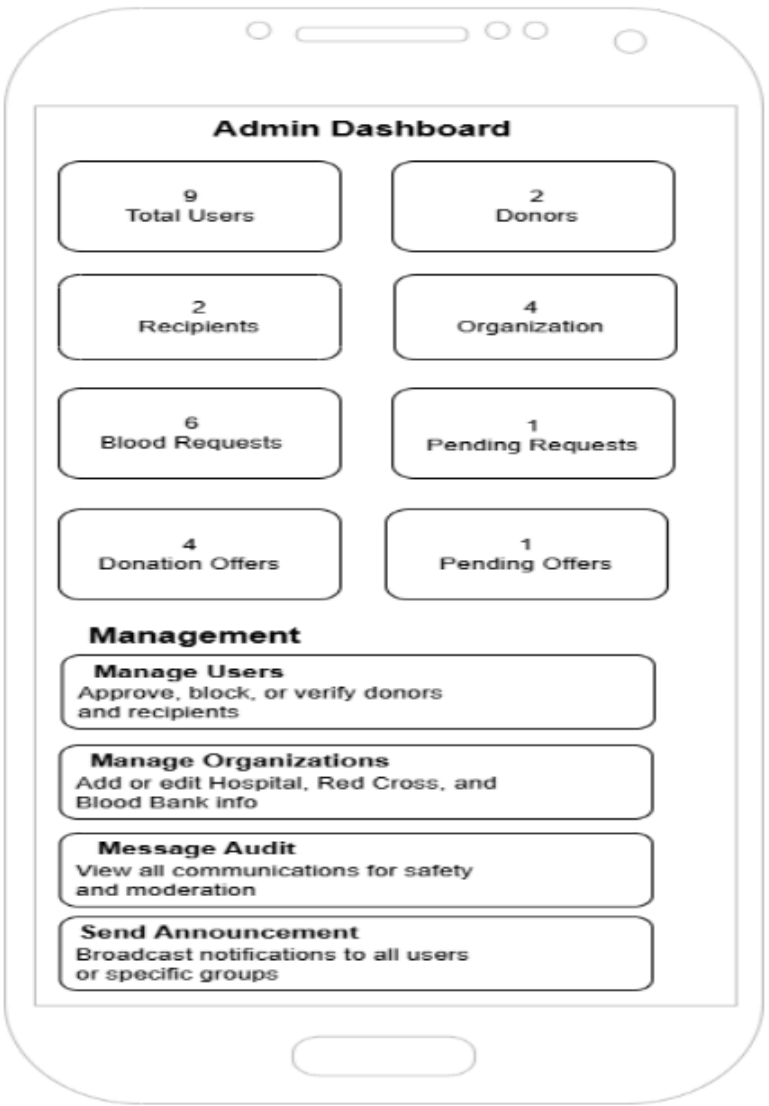
This dashboard allows organizations to efficiently manage blood requests, accept or reject offers, and oversee operations within the system.



10 Figure 5.4.3 Organization Dashboard

5.4.4 System Administrator Dashboard Wireframe

This dashboard allows the system administrator to control user management, send announcements, and monitor overall system activities.



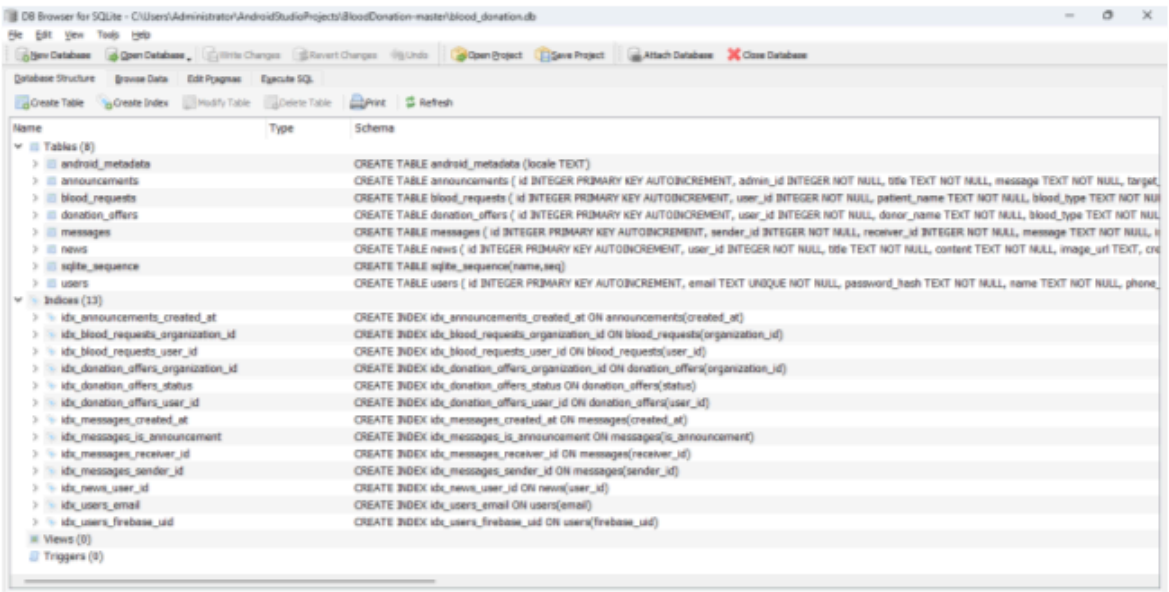
11 Figure 5.4.4 Admin Dashboard

5.5 Database Design

The database design of the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms defines the structure and relationships needed to store and manage user accounts, donor and recipient data, and blood request records. The system uses SQLite, a lightweight mobile database, to ensure efficient, secure, and reliable data storage and retrieval.

5.5.1 Database Schema

The database schema defines the structure of the database, including tables, fields, and their respective data types. The Smart Blood Donation and Emergency Response Application database schema includes tables for users, donors, recipients, organizations, and blood requests, each designed to support accurate data storage and efficient system operations



12 Figure 5.5.1 Database Schema

5.6 Interface Design

The interface design of the Smart Blood Donation and Emergency Response Application Using Real-Time Geo-tracking and Donor Matching Algorithms focuses on delivering a clean, intuitive, and user-centered mobile experience. The design ensures that each user group donors, recipients, organizations, and system administrators can efficiently access the features relevant to their roles.

Flutter's widget-based architecture is used to create consistent layouts, smooth navigation, and responsive screens across Android and iOS devices.

Key interface elements include simplified forms for offering and requesting blood, real-time status displays, and dedicated dashboards for each user type. Clear visual cues, structured menus, and accessible icons improve usability, while color-coded indicators support faster decision-making during emergencies. The interface also incorporates real-time geolocation views, alert notifications, and structured data presentation to enhance user interaction and support critical response workflows.

5.6.1 System Administrator Dashboard

The System Administrator Dashboard provides tools for managing all user accounts, creating organization profiles, and overseeing system operations. It includes options for sending announcements, issuing private messages, and ensuring that all user activities comply with system policies.

5.6.2 Organization Administrator Dashboard

The Organization Administrator Dashboard allows hospitals, blood banks, and Red Cross centers to manage blood-related activities. It provides interfaces for accepting or rejecting donor offers and recipient requests, submitting new blood requests, and monitoring ongoing operations within their organization.

5.6.3 Donor Dashboard

The Donor Dashboard enables donors to offer blood, update their availability status, and track the outcomes of their previous offers. It presents a simplified and user-friendly layout that helps donors quickly access key actions and receive updates from organizations.

5.6.4 Recipient Dashboard

The Recipient Dashboard allows users to request blood, monitor the progress of their requests, and receive timely updates from organizations. It provides a clear and intuitive interface designed to help recipients find the support they need during emergencies.

5.6.5 Blood Request and Offer Interface

This interface allows both donors and recipients to submit accurate blood offers or blood requests. It includes structured fields for blood type, location, and urgency level, ensuring that organizations receive all necessary information for timely decision-making.

5.6.6 Real-Time Geo-Tracking Interface

The real-time geo-tracking interface displays users' current locations and nearby donors or organizations, using integrated maps to support fast emergency response. It helps users visualize distances and select the most appropriate support option based on proximity.

5.6.7 Notifications Interface

The notifications interface provides real-time alerts related to request approvals, donor responses, emergency updates, and system announcements. It ensures that all users are promptly informed of actions requiring their attention.

5.6.8 Profile Management Interface

The profile management interface allows users to update their personal information, medical details, and contact data. It ensures that all stored information remains accurate and helps improve the reliability of blood matching and emergency response processes.

5.7 Summary

This chapter described the architecture and design of the Smart Blood Donation and Emergency Response Application. It explained how the system is structured, how the main components interact, and how user roles fit into the overall design. The diagrams presented in this chapter outlined the system's processes, data flow, and relationships between different parts of the application.

The database design section highlighted how data is organized using SQLite and how the schema supports the main functions of the system. The interface design section also outlined the key screens used by system administrators, organizations, donors, and recipients, showing how each interface supports the tasks performed by each user type.

CHAPTER SIX: SYSTEM DEVELOPMENT AND IMPLEMENTATION

6.1 Introduction

This chapter explains how the Smart Blood Donation and Emergency Response Application was developed and implemented. It outlines the tools and technologies used, the development environment, and the steps taken to build the main features of the system. The sections that follow describe how the application's functions were created in Flutter, how data was managed using SQLite, and how the system was prepared for practical use.

6.2 Development Tools and Technologies

The application was developed using Flutter and Dart for building the mobile interface and system logic. SQLite was used as the local database to store user and request information. Android Studio and VS Code supported coding, testing, and overall project management.

6.3 System Implementation

The system was implemented by translating the design into functional modules using Flutter and Dart. Key features such as user registration, login, dashboards, blood requests, blood offers, and real-time tracking were developed and connected to the SQLite database. Each module was built, tested, and refined to ensure smooth operation and ease of use.

6.4 Feature Implementation

The main features of the system were implemented to support the actions of each user type. This includes secure signup and login, dashboards for administrators, organizations, donors, and recipients, as well as interfaces for offering and requesting blood. Real-time geo-tracking and notifications were also integrated to support quick responses during emergencies.

6.5 System Testing

The system was tested to ensure that all features functioned correctly and that users could perform their tasks without errors. Testing focused on verifying user registration, login, blood requests and

offers, dashboard operations, database interactions, and real-time features. Issues identified during testing were corrected to improve stability and usability.

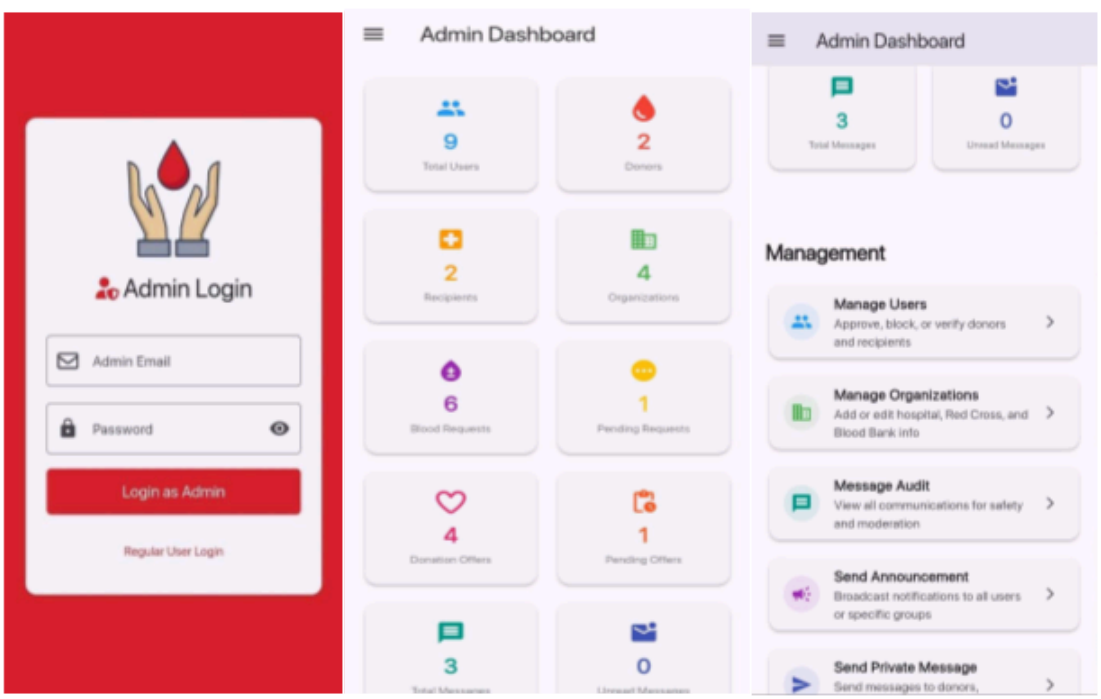
6.6 Challenges and Solutions

During development, several challenges were encountered, including database integration issues, interface alignment problems, and handling real-time location updates. These were resolved through code adjustments, improved testing, and refining the Flutter widgets and SQLite queries to ensure smooth and reliable system performance.

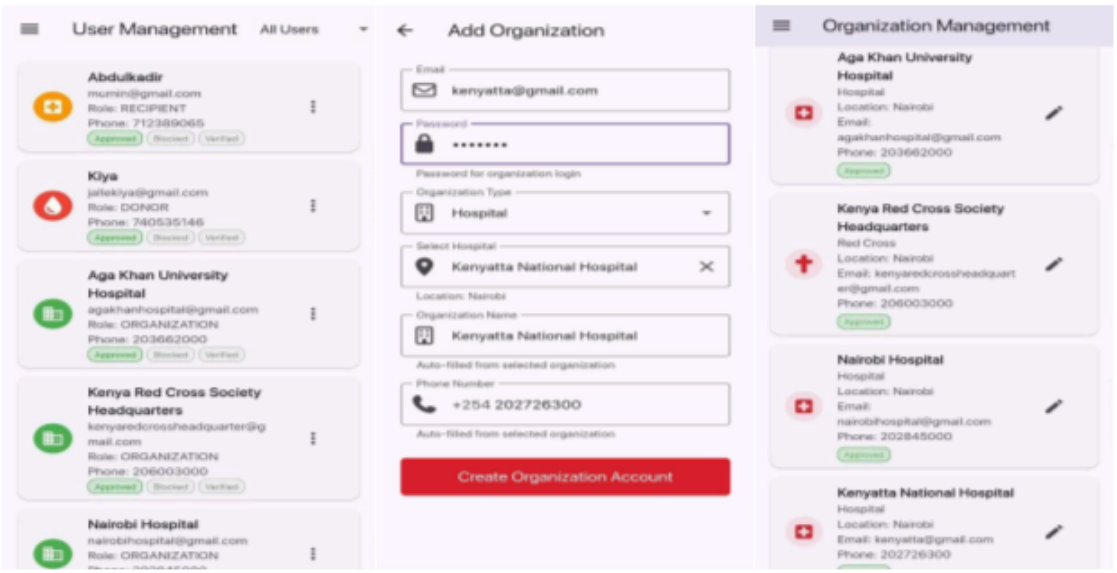
6.7 System Screenshots

This section includes screenshots of the application's main interfaces, showing how the system looks and functions during use.

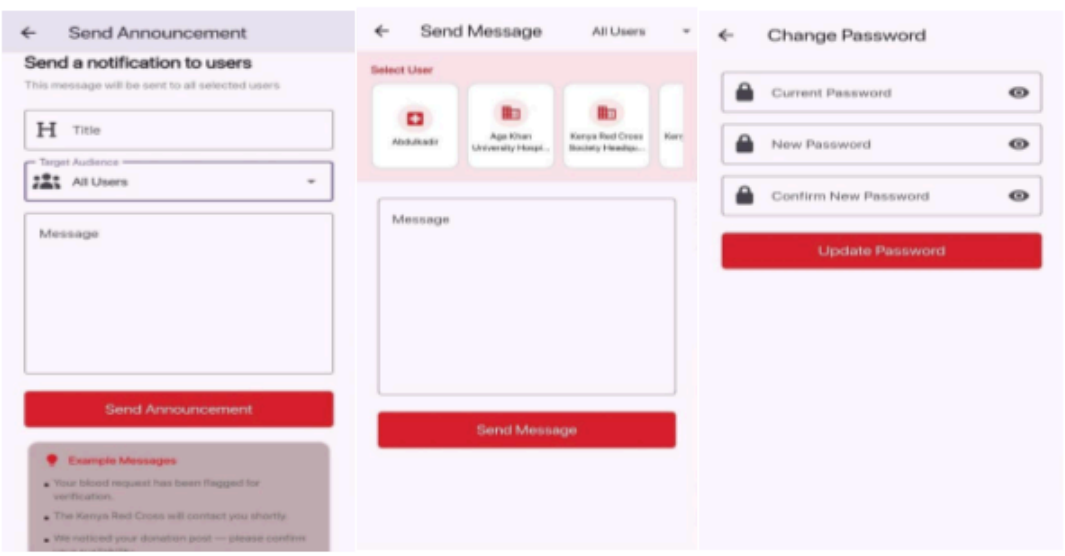
6.7.1 System Administrator Screens



13 Figure 6.7.1 Admin Login and Dashboard Screens

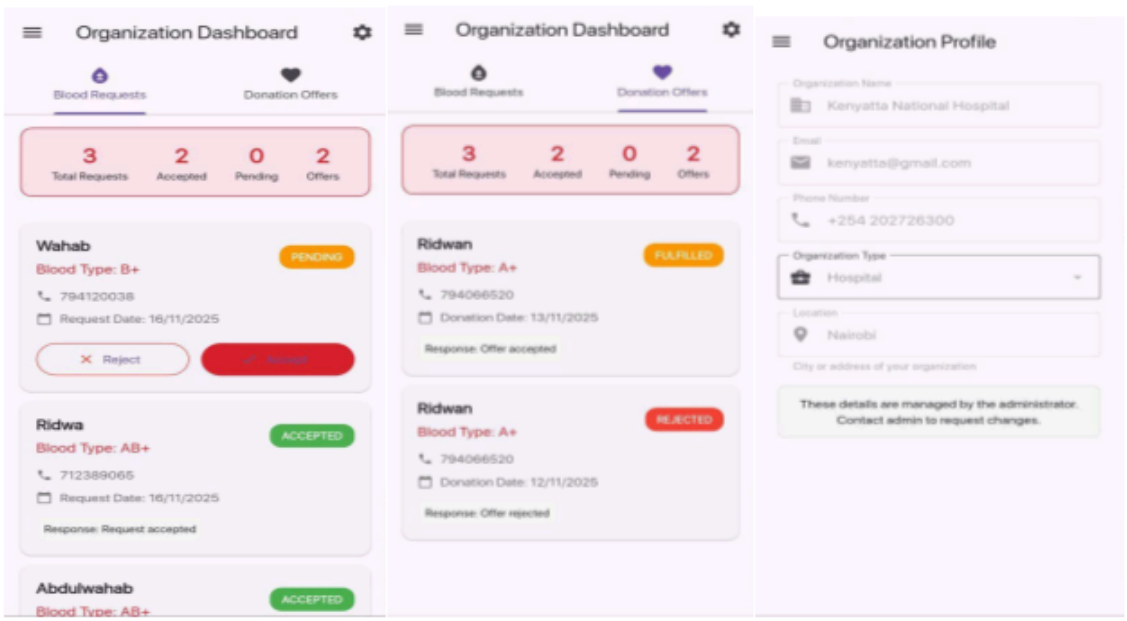


14 Figure 6.7.1 User Management, Create Organization Account and Organization Management Screens

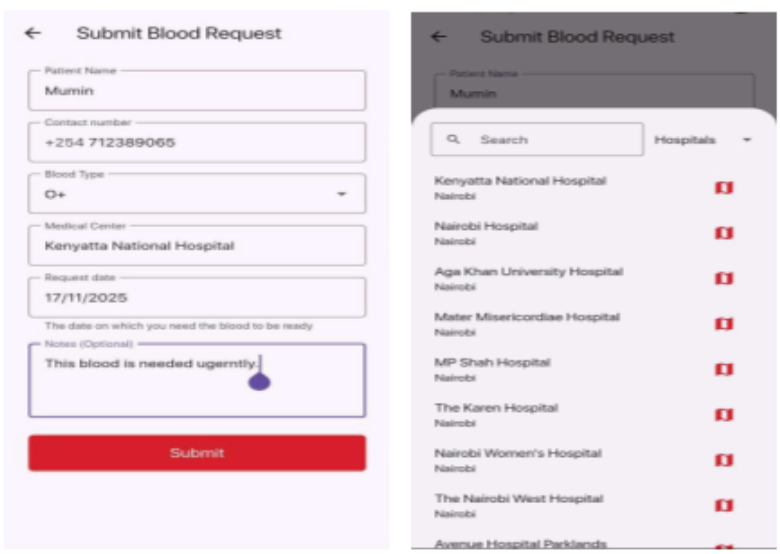


15 Figure 6.7.1 Send Announcements, Send Private Message and Change Password Screens

6.7.2 Organization Administrator Screens

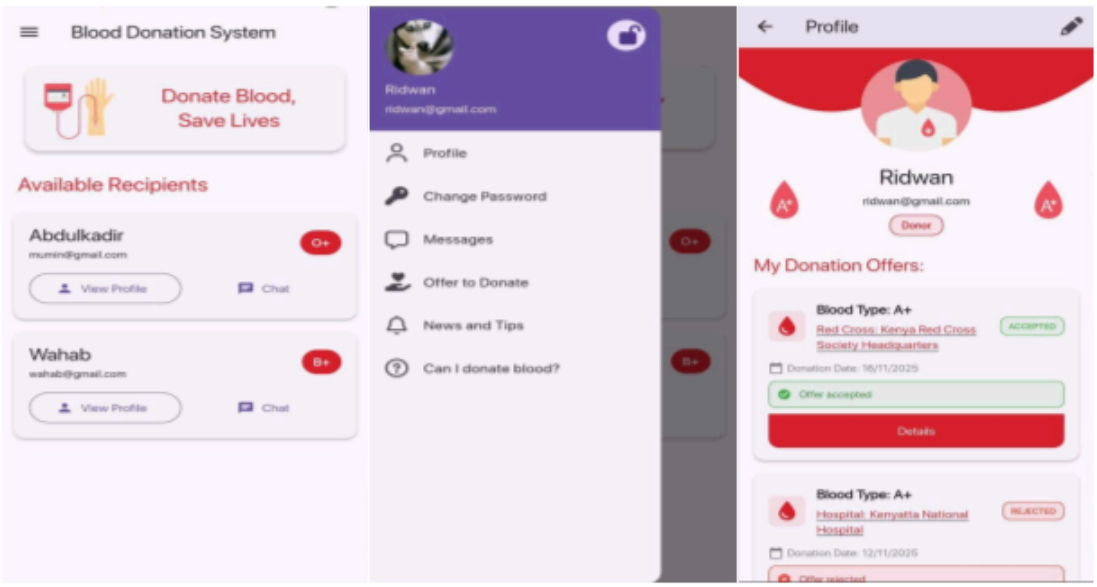


16 Figure 6.7.2 Organization Admin Dashboard (Blood Requested), (Offered Blood) and Organization Profile Screens

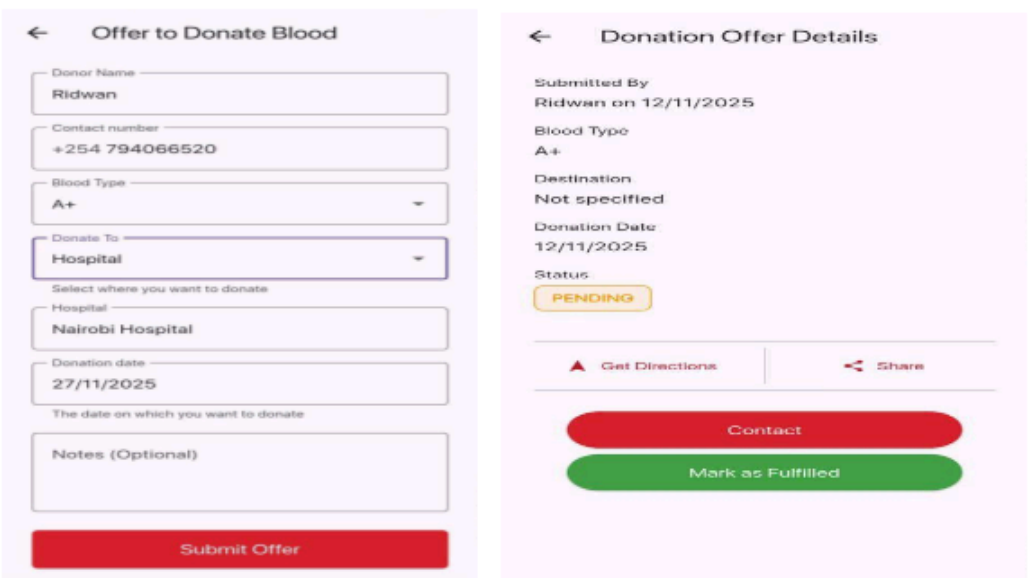


17 Figure 6.7.2 Select Organization (Requesting Blood) and Organization Blood Request Screens

6.7.3 Donor Screens

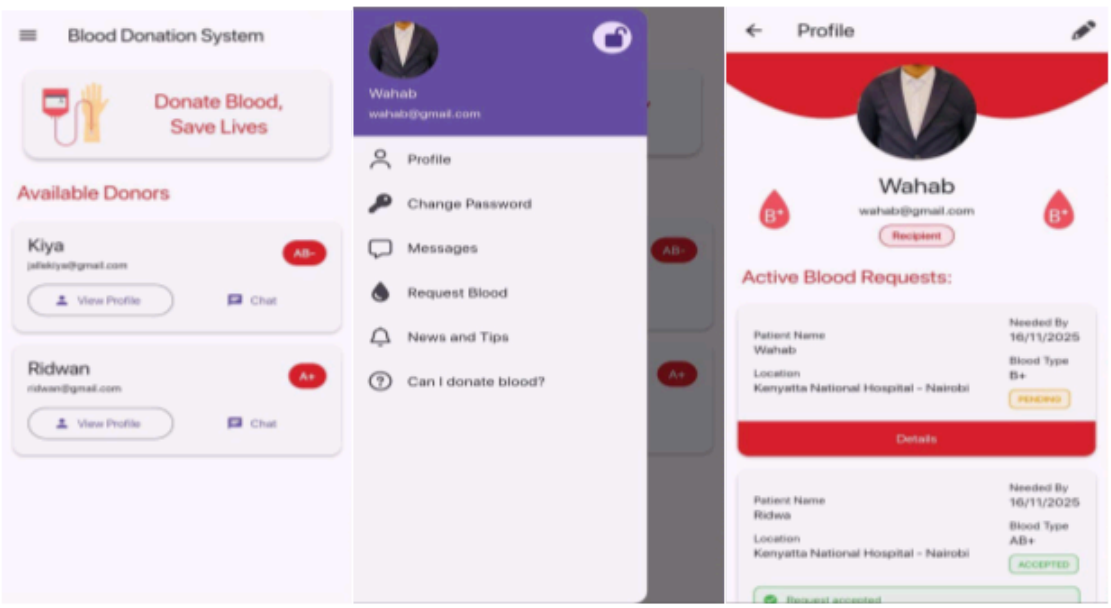


18 Figure 6.7.3 Donor Dashboard, Donor Dashboard (Drawer) and Donor Profile Screens

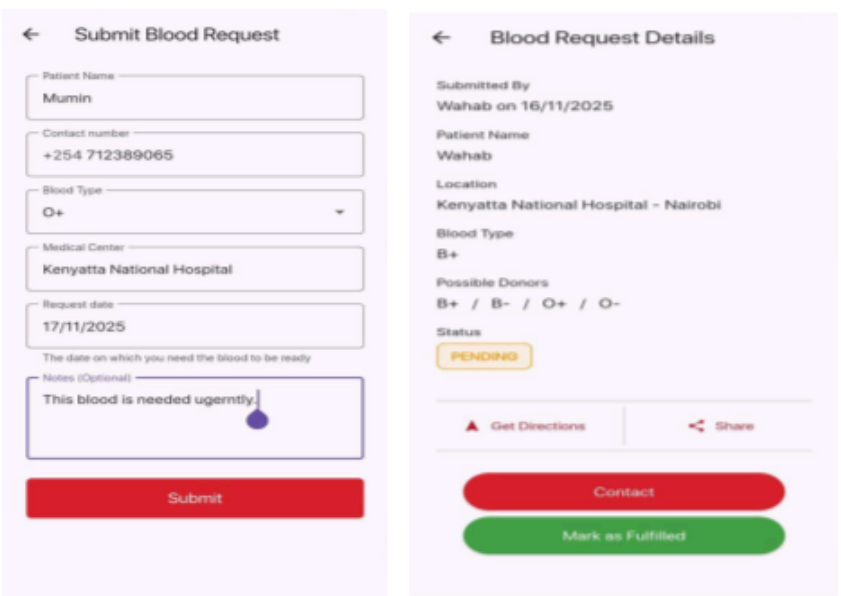


19 Figure 6.7.3 Offer Blood and Offer Status Screens

6.7.4 Recipient Screens

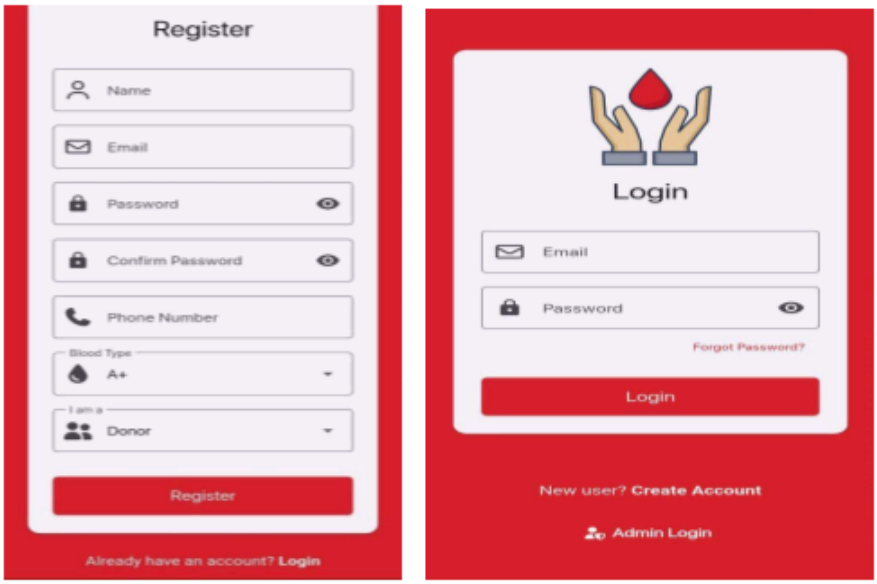


20 Figure 6.7.4 Recipient Dashboard, Recipient Dashboard (Drawer) and Recipient Screens

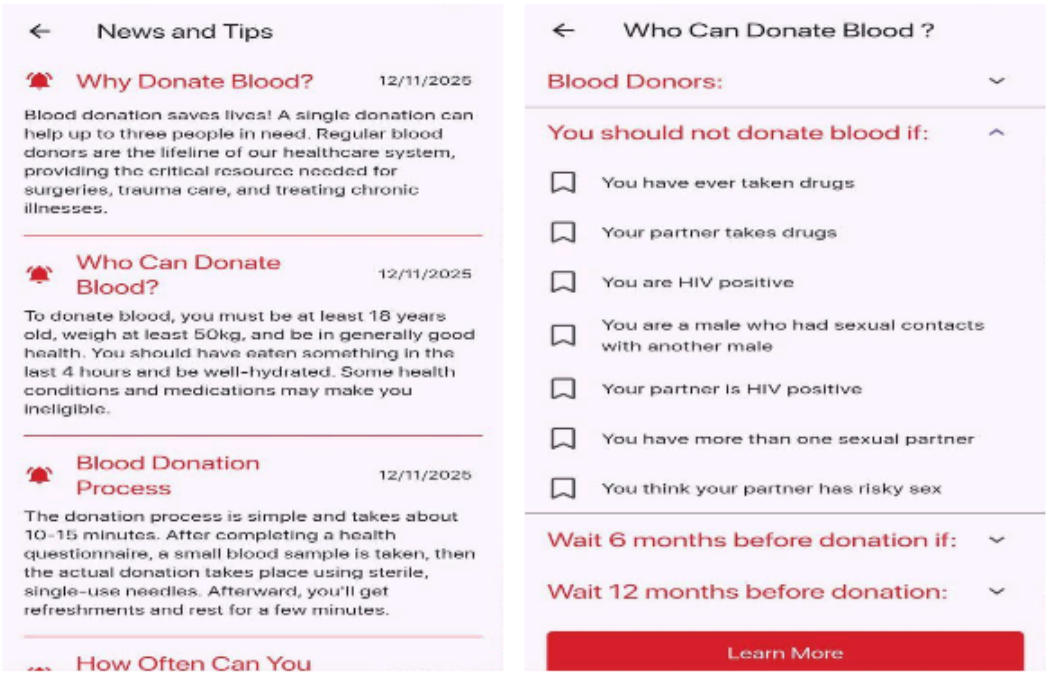


21 Figure 6.7.4 Request Status and Request Blood Screens

6.7.5 Shared Screens (Common for All Users)



22 Figure 6.7.5 Signup and Login Screens



23 Figure 6.7.5 News and Tips and Who can Donate blood Screens

6.8 Summary

This chapter explained how the Smart Blood Donation and Emergency Response Application was developed and implemented. It highlighted the tools and technologies used, the steps followed during development, and how the core features such as user registration, donor and recipient dashboards, blood requests, blood offers, messaging, and real-time tracking were built in Flutter using SQLite. The chapter also outlined the testing carried out to confirm that each function worked as expected and described the main challenges encountered during development and how they were resolved. Overall, the chapter showed how the design was translated into a working mobile application that supports fast and reliable blood donation and emergency coordination.

CHAPTER SEVEN: SYSTEM DEPLOYMENT AND TESTING

7.1 Introduction

This chapter explains how the Smart Blood Donation and Emergency Response Application was tested and prepared for deployment. It outlines the different testing methods used to verify that each feature works correctly, identifies and fixes errors, and ensures the system performs well in real use. The chapter also describes the deployment approach used during development and the plans for future distribution on mobile platforms.

7.2 System Testing and Validation

System testing and validation were carried out to ensure that the Smart Blood Donation and Emergency Response Application operates as expected and meets the system requirements. Different types of tests were conducted to check individual functions, overall features, the user interface, module interactions, and basic security controls. The outcomes of each test are presented in the subsections below.

7.2.1 Unit Testing

Unit testing focused on checking individual components of the system to ensure that each function worked correctly on its own. Core features such as user registration, login, password updates, blood request submission, donor blood offering, and dashboard loading were tested separately.

Outcome:

The results showed that all tested units performed as expected. Minor issues, such as incorrect input validation and small logical errors, were identified and corrected. After adjustments, each feature operated reliably when tested independently.

7.2.2 Functional Testing

Functional testing was carried out to confirm that all system features work according to the specified requirements. This included testing donor actions, recipient requests, organization approvals or rejections, admin account management, and the sending of announcements and private messages. Each function was tested from start to finish to ensure correct behavior under normal usage.

Outcome:

The results showed that all major functions operated correctly and met the expected requirements. Donors were able to offer blood, recipients successfully submitted requests, organizations could approve or reject actions, and administrators managed user accounts without issues. No major functional failures were observed.

7.2.3 User Interface (UI) Testing

UI testing was conducted to ensure that the application's screens, navigation, and layout elements displayed correctly across different devices. This included checking the responsiveness of buttons, clarity of text, consistency of colors and icons, and smooth transitions between pages. The goal was to confirm that users could interact with the system easily and without confusion.

Outcome:

UI testing showed that all screens loaded properly and navigation between pages was smooth. Buttons and forms responded as expected, and the layout remained consistent across devices. Minor alignment issues were identified and fixed, resulting in a clean and user-friendly interface.

7.2.4 Integration Testing

Integration testing was performed to ensure that the different modules of the system worked together without errors. This involved checking interactions between the frontend interface, the SQLite database, the dashboards, and features such as blood requests, donor offers, notifications, and user authentication. The goal was to confirm that data flowed smoothly between components and that multi-step processes functioned correctly.

Outcome:

Integration testing confirmed that all modules communicated properly and that data was transferred accurately between different parts of the system. Actions such as submitting a request, receiving an organization response, and updating dashboards worked smoothly from start to finish. No major integration issues were found.

7.2.5 Security Testing

Security testing was carried out to verify that user data and system operations were protected from unauthorized access. This included checking the strength of the login system, password handling, access restrictions for different user roles, and validation of user inputs to prevent incorrect or harmful data from being processed.

Outcome:

Security testing showed that the authentication process functioned correctly, preventing unauthorized access to user accounts and restricted features. Input validation worked as intended, and password-related operations were secure. No critical security weaknesses were identified during testing.

7.3 Testing Results and Bug Management

Testing results showed that the system performed well across all major features, with most functions operating as expected during unit, functional, UI, integration, and security testing. Several minor issues were identified during the testing process, including small layout inconsistencies, validation errors, and navigation bugs. These issues were documented, analyzed, and resolved through code adjustments and interface refinements. Continuous testing after each fix ensured that no new errors were introduced and that overall system reliability improved. By the end of the testing phase, the application showed stable performance and met the key requirements for functionality and usability.

7.4 Deployment Strategy

The deployment strategy focused on preparing the application for real use during the testing phase and ensuring that it could be easily installed on mobile devices. The system was packaged as an APK file for Android and distributed to selected users for evaluation. This approach allowed testers

to install the app directly on their devices, use its features in real scenarios, and provide feedback on performance and usability. The deployment process emphasized simplicity, ensuring that users could access the app without additional technical setup.

7.4.1 Beta Testing

Beta testing was carried out by providing the application to a small group of users, including donors, recipients, and organization representatives. These testers explored the system's features, performed real actions such as submitting requests and offers, and provided feedback on usability, performance, and any issues encountered. The feedback gathered during this phase helped identify areas that needed adjustment, leading to improvements in navigation, layout, and overall system stability.

7.4.2 Play Store & App Store Deployment

Full deployment of the application to the Google Play Store and Apple App Store is planned for future work. This process will involve preparing the required release builds, meeting platform-specific guidelines, performing additional testing for compatibility, and completing the app submission procedures. Once these steps are completed, the system will be accessible to a wider audience and available for public use.

7.5 Summary

This chapter discussed how the system was tested, validated, and prepared for deployment. It outlined the different testing methods used to check individual components, overall functionality, user interface behavior, module interactions, and basic security controls. The chapter also presented the results of the testing phase and explained how identified issues were resolved. In addition, it described the deployment approach used during development, including beta testing and plans for future release on mobile app stores. Overall, the chapter demonstrated that the application performs reliably, meets its core requirements, and is ready for broader use.

CHAPTER EIGHT: SYSTEM REVIEW AND LAUNCH

8.1 Introduction

This chapter presents the final review and launches activities carried out for the Smart Blood Donation and Emergency Response Application. Since the project followed the Agile methodology, this stage focused on assessing the functionality delivered in each iteration, gathering feedback, and preparing the system for release. The chapter summarizes the system's performance, highlights key achievements, and outlines the steps taken to launch the application for real-world testing and future deployment.

8.2 System Review

The system review evaluated how well the application met the objectives set at the beginning of the project and how effectively it performed during testing. Feedback from users involved in the beta testing phase was used to assess the ease of use, reliability, and responsiveness of the system. The review confirmed that the core features such as donor matching, blood requests, real-time tracking, and user communication functioned as intended. The system demonstrated good stability, and updates made during the development iterations improved both performance and user experience. Overall, the review showed that the application successfully supports faster donor coordination and emergency response, in line with the project goals.

8.3 System Launch

The system launch involved releasing the application for real-world use during the beta testing phase. The APK version of the app was shared with selected users, including donors, recipients, organization administrators, and the system administrator, to allow them to access the system on their mobile devices. This launch enabled testers to interact with the system's features in practical scenarios, submit feedback, and report any issues that required refinement. Although the full deployment to official app stores is planned for future work, the current launch demonstrated that the application is functional, stable, and ready for broader use once final adjustments are completed.

8.4 User Feedback

During the beta testing phase, feedback was collected from donors, recipients, organization administrators, and the system administrator to evaluate the usability and performance of the application. Testers reported that the system was easy to use and that the layout of the dashboards made it simple to perform key actions such as requesting blood, offering blood, and managing approvals. Users also noted that the real-time geo-tracking and request notifications improved the speed of coordination during emergencies.

Some users suggested small improvements, such as clearer button labels and faster loading times on certain screens, which were addressed during the review process. Overall, the feedback was positive and confirmed that the system functions well and supports the needs of its intended users.

8.5 Improvements After Review

Based on the feedback received during the review and beta testing phase, several improvements were made to enhance the system's performance and user experience. Interface adjustments were implemented to improve screen layout, button visibility, and navigation flow across different user dashboards. Validation checks were strengthened to prevent incomplete or incorrect data from being submitted, especially in blood request and offer forms. Loading times for location-based features were also optimized to ensure smoother operation during emergencies. These updates helped refine the system and made it more responsive and user-friendly.

8.6 System Limitations

Despite its successful development, the system still has a few limitations. The application currently operates only on Android devices, meaning users on other platforms cannot access it yet. Real-time geo-tracking depends on the user's device and network quality, which may affect location accuracy in some situations. In addition, the system relies on a local SQLite database, limiting cloud-based data sharing and large-scale backup capabilities. These limitations do not affect core functionality but highlight areas that can be improved in future versions.

8.7 Recommended Future Enhancements

Several enhancements can be introduced in future versions of the application to broaden its capabilities and improve user experience. Integrating a cloud-based database would enable real-time data sharing across devices and support larger-scale operations. Adding push notifications would make alerts faster and more reliable, especially during emergencies. Expanding the application to iOS devices would increase accessibility for a wider range of users. Additional features such as AI-based donor matching, advanced reporting tools for organizations, and integration with hospital management systems would further strengthen the system's efficiency and impact.

8.8 Summary

This chapter presented the final review and launch activities for the Smart Blood Donation and Emergency Response Application. It summarized user feedback collected during beta testing and highlighted the improvements made to enhance the system's performance and usability. The chapter also outlined the current limitations of the application and suggested future enhancements to expand its functionality and reach. Overall, the review and launch process demonstrated that the system is effective, stable, and ready for further development and wider deployment.

CHAPTER NINE: DISCUSSION, CONCLUSION AND RECOMMENDATIONS

9.1 Introduction

This chapter presents the overall discussion, conclusion, and recommendations drawn from the development of the Smart Blood Donation and Emergency Response Application. It reflects on the system's performance, the objectives achieved, and the key findings gathered throughout the project. The chapter also provides final conclusions based on the outcomes of the study and offers recommendations for future improvements and further development of the system.

9.2 Discussion

The development and evaluation of the Smart Blood Donation and Emergency Response Application demonstrated that technology can significantly improve the speed and coordination of blood donation processes. The system successfully addressed the main challenges identified in the problem statement, including delays in locating donors, limited communication, and the lack of real-time tracking during emergencies. Through the use of Flutter, SQLite, and geo-tracking features, the system provided a reliable platform for donors, recipients, organizations, and administrators to interact efficiently.

The testing and review phases showed that users found the application easy to navigate, with clear interfaces tailored to their roles. Organizations benefited from faster approval and matching processes, while donors and recipients appreciated the simplified request and offer workflows. Although the system performed well, some limitations such as device dependency and the absence of cloud synchronization highlighted areas that could be improved in future versions. Overall, the system proved effective in enhancing emergency response and demonstrated strong potential for broader adoption.

9.3 Conclusion

The Smart Blood Donation and Emergency Response Application successfully achieved its main objectives of improving donor coordination, enhancing emergency response, and providing a more efficient communication platform for donors, recipients, and healthcare organizations. By integrating real-time geo-tracking, donor matching features, and user-friendly dashboards, the system addressed the challenges associated with manual blood donation processes.

The results from testing and user feedback confirmed that the system is functional, reliable, and easy to use. While certain limitations remain, the core features performed effectively and demonstrated clear potential to support faster and more organized blood donation efforts. Overall, the project shows that a mobile-based solution can play a significant role in strengthening healthcare response during emergencies.

9.4 Recommendations

Based on the findings and feedback gathered during development and testing, several recommendations can help improve the system in future versions. First, integrating a cloud-based database is recommended to enable real-time data sharing across multiple devices and support larger-scale operations. Expanding the application to support iOS devices would also increase accessibility for a wider range of users.

Additionally, implementing push notifications would enhance responsiveness during emergencies by delivering faster updates to donors and recipients. Strengthening the matching algorithm and introducing advanced reporting tools for organizations could further improve decision-making and efficiency. Finally, publishing the application on the Play Store and App Store after additional refinement would allow the system to reach more users and maximize its impact.

9.5 Future Work

Future development of the system can focus on expanding its capabilities and enhancing overall performance. One important area is the integration of cloud-based services to support real-time synchronization, centralized data management, and multi-device access. Incorporating machine learning techniques could also improve donor matching accuracy and predict blood availability based on past trends.

Further enhancements may include linking the system with hospital information systems, enabling automated verification of blood inventory levels, and supporting medical staff with more detailed reports. The application can also be extended to iOS devices and published on mobile app stores to reach a wider user base. Continued testing, user feedback, and refinement will help ensure that the system evolves into a fully scalable and widely adopted emergency response tool.

9.6 Summary

This chapter presented the overall discussion, conclusion, and recommendations drawn from the development and evaluation of the Smart Blood Donation and Emergency Response Application. It discussed how the system addressed the main challenges identified in the study and how user feedback supported its effectiveness in improving donor coordination and emergency response. The chapter also provided a conclusion based on the project outcomes, outlined key recommendations for further enhancement, and highlighted areas for future development. Overall, the chapter brought together the final insights of the study and demonstrated the system's potential for wider adoption.

REFERENCES

Alhassan, V., Mensah, F., & Boadu, S. (2021). Mobile-based blood donation management systems: A review. International Journal of Health Informatics, 7(2), 45–56.

Ali, A., Khan, M., & Fatima, S. (2019). Mobile application for blood donation and emergency services. International Journal of Innovative Research in Computer and Communication Engineering, 7(3), 110–116.

Ambler, S. W. (2002). Agile modeling: Effective practices for extreme programming and the unified process. John Wiley & Sons.

American Red Cross. (2020). Red Cross Blood Donor App. https://www.redcrossblood.org/blood-donor-app.html

Bass, L., Clements, P., & Kazman, R. (2013). Software architecture in practice (3rd ed.). Addison-Wesley.

Beck, K. (2002). Extreme programming explained: Embrace change. Addison-Wesley.

Bharadwaj, A., & Tongaonkar, S. (2018). Smart blood bank system using mobile applications. International Journal of Computer Applications, 179(48), 1–5.

Boehm, B. W. (1986). A spiral model of software development and enhancement. ACM SIGSOFT Software Engineering Notes, 11(4), 14–24.

Chen, Y., & Zhang, L. (2022). Smart mobile solutions for emergency blood donor management. Journal of Medical Systems, 46(1), 25–33.

Cohn, M. (2005). Agile estimating and planning. Prentice Hall.

Comfort, L. K. (2007). Crisis management in hindsight: Cognition, communication, coordination, and control. Public Administration Review, 67(s1), 189–197.

Creswell, J. W., & Creswell, J. D. (2017). Research design: Qualitative, quantitative, and mixed methods approaches (5th ed.). Sage Publications.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly, 13(3), 319–340.

El-Masri, A., & Tarhini, A. (2019). Mobile health systems and applications: A review. Journal of Medical Internet Research, 21(4), e12516.

Fielding, R. T. (2000). Architectural styles and the design of network-based software architectures (Doctoral dissertation). University of California, Irvine.

Google. (2023). Dart language overview. https://dart.dev/guides

Google. (2023). Flutter documentation. https://docs.flutter.dev

Hofmann-Wellenhof, B., Lichtenegger, H., & Wasle, E. (2007). GNSS: Global navigation satellite systems. Springer.

Hunt, K. (2020). Flutter for beginners: An introductory guide to building cross-platform mobile applications with Flutter and Dart 2. Packt Publishing.

Kudvenkat, P. (2021). SQLite database system: Design, implementation, and usage. TechPress.

Kumar, P., & Rajesh, G. (2020). Mobile technologies for emergency healthcare services: A survey. Journal of Emergency Medical Informatics, 5(1), 33–42.

Larman, C. (2017). Applying UML and patterns: An introduction to object-oriented analysis and design and iterative development (3rd ed.). Pearson.

Nielsen, J. (1994). Usability engineering. Morgan Kaufmann.

Osei, E., & Appiah, E. (2021). Mobile health applications: Prospects and challenges in developing countries. International Journal of Computer Applications, 174(32), 15–21.

Patel, K., & Sharma, R. (2021). A review of blood donation mobile applications. Journal of Health Informatics in Developing Countries, 15(2), 67–75.

Pressman, R. S. (2014). Software engineering: A practitioner's approach (8th ed.). McGraw-Hill Education.

Pressman, R. S., & Maxim, B. R. (2019). Software engineering: A practitioner's approach (9th ed.). McGraw-Hill Education.

Rahman, S., et al. (2020). Emergency blood donation mobile application for smart health. International Journal of Advanced Computer Science and Applications (IJACSA), 11(5), 123–130.

Schwaber, K. (2017). The Scrum Guide. Scrum Alliance.

Sommerville, I. (2011). Requirements engineering: Processes and techniques. Addison-Wesley.

Sommerville, I. (2016). Software engineering (10th ed.). Pearson.

WHO. (2020). Blood safety and availability fact sheet. World Health Organization.

World Health Organization. (2017). Global status report on blood safety and availability. WHO.

Zheng, Y. (2015). Trajectory data mining: An overview. ACM Computing Surveys, 46(4), 1–33.

Zhou, X., & Wu, Z. (2021). Location-based services and mobile GPS technologies in emergency response. International Journal of Mobile Computing, 9(3), 55–71.

APPENDICES

Appendix A: Sample Questionnaire

Section 1: Donor Information

Are you currently a registered blood donor?

How frequently do you donate blood?

Would you use a mobile app to respond to emergency blood requests?

Section 2: System Requirements

What key features do you expect in an emergency blood donation app? (e.g., location tracking, instant alerts, donor privacy)

How important is data security and privacy to you when using such an app?

Section 3: Feedback

Do you have suggestions to improve emergency blood donor coordination?

Would you recommend this app to other potential donors?

Appendix B: Sample Interview Guide

Section 1: General Information

1. Can you describe your role in emergency blood donation coordination?
2. How do you currently manage urgent donor matching?

Section 2: Challenges & Needs

3. What challenges do you face when locating suitable donors during emergencies?
4. What tools do you currently use for urgent blood requests?
5. What improvements would you like to see in your current system?

Section 3: App Expectations

6. Which features would make a blood donation app valuable for you?
7. How can this app best support hospitals and donors during urgent cases?

Appendix C: Sample Observation Checklist

1. Observe how emergency blood requests are processed in real time.
2. Note how potential donors are located and contacted.
3. Identify any delays or communication breakdowns in current processes.
4. Observe the tools used (e.g., manual call lists, SMS, notice boards).
5. Record how donor records are stored and retrieved.
6. Monitor staff coordination and workflows.

Appendix D: UML Diagrams

This appendix contains all diagrams created during system design:

1. Use Case Diagram (Admin, Organization, Donor, Recipient roles)
2. Class Diagram
3. Activity Diagrams
4. Sequence Diagrams
5. Entity Relationship Diagram (ERD)

Appendix E: Wireframes and Interface Designs

This appendix includes the wireframes and descriptions for:

1. Signup and Login Pages
2. System Administrator Dashboard
3. Organization Administrator Dashboard
4. Donor Dashboard
5. Recipient Dashboard
6. Blood Request and Offer Screens

7. Geo-tracking Screen
8. Notification Interface
9. Profile Management Screen

Appendix F: Database Schema

This appendix presents the full SQLite database structure used in the application:

1. Users Table
2. Donors Table
3. Recipients Table
4. Organizations Table
5. Blood Requests Table
6. Blood Offers Table
7. Authentication & Logging Tables (if any)

Includes:

1. Table names
2. Fields and data types
3. Primary and foreign keys
4. Relationships based on ERD

Appendix G: Testing Results

This appendix contains testing outputs collected during:

1. Unit Testing
2. Functional Testing
3. Integration Testing
4. UI Testing
5. Security Testing (optional)

Sample Test Results Table:

Table 4 Sample Test Results Table

Test Case	**Expected Result**	**Actual Result**	**Status**
User Registration	Account created	Success	Pass
Donor Offer Submission	Offer recorded	Success	Pass
Recipient Blood Request	Request saved	Success	Pass
Organization Approval	Updated status	Success	Pass
Geo-tracking	Location visible	Accurate	Pass

Appendix H: System Screenshots

Includes real application screenshots for:

System Administrator

- Login Page
- Dashboard
- User Management
- Create Organization Account
- Organization Management

- Send Private Messages
- Announcements
- Change Password

Organization Administrator

- Dashboard
- Requested Blood Page
- Offered Blood Page
- Organization Blood Request
- Select Organization to Request From
- Organization Profile
- Change Password

Donor

- Dashboard
- Offer Blood
- Notifications
- Change Password

Recipient

- Dashboard
- Request Blood
- Request Status
- Change Password

Shared Screens

- Signup Page

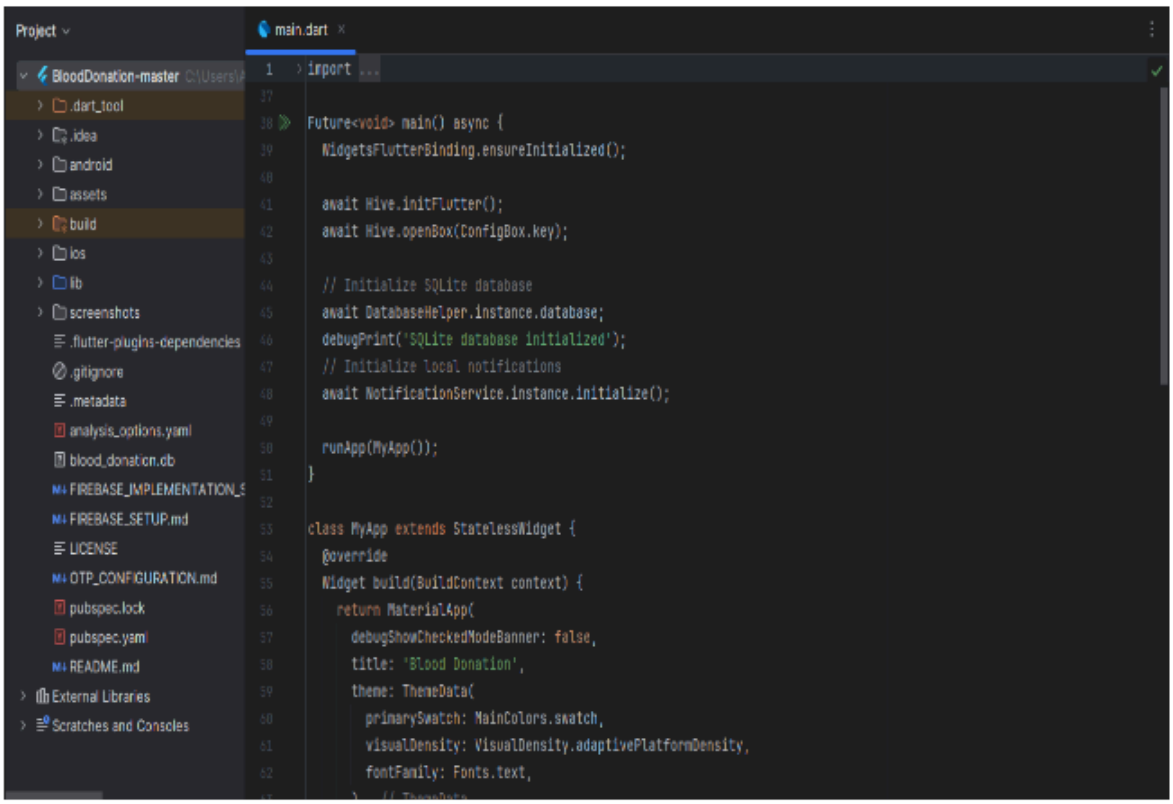
- Login Page

Appendix I: System Architecture and Implementation Notes

Includes supporting documentation:

- System architecture overview
- Flutter implementation structure
- SQLite integration notes
- Geo-tracking integration details
- Agile development iteration notes

Appendix I: Sample Code



24 Figure Sample Code 1

25 Figure Sample Code 2