



Daffodil
International
University

TITLE OF THE PROJECT

**Online Medicine Donation System
(OMDS)**

Submitted By

Robiul Islam

ID: 181-16-254

Supervised By

Mr. Abdullah Bin Kasem Bhuiyan

Lecturer(CIS)

**Department of Computing and Information System
(CIS)**

DAFFODIL INTERNATIONAL UNIVERSITY

Fall 2021

APPROVAL

This Project titled “Online Medicine Donation System”, Submitted by Robiul Islam, ID No: 181-16-254 to the Department of Computing & Information Systems, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information Systems and approved as to its style and contents. The presentation has been held on- 13-02-2022.

BOARD OF EXAMINERS



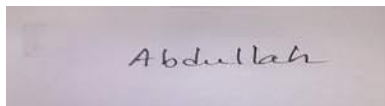
*Mr. Md Sarwar Hossain Mollah
Associate Professor and Head
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University*

Chairman



*Ms. Nayeema Rahman
Sr. Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University*

Internal Examiner



*Mr. Abdullah Bin Kasem Bhuiyan
Lecturer
Department of Computing & Information Systems
Faculty of Science & Information Technology
Daffodil International University*

Internal Examiner



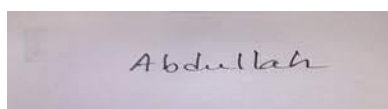
*Dr. Mohammad Shorif Uddin
Professor
Department of Computer Science and Engineering
Jahangirnagar University, Savar, Dhaka*

External Examiner

DECLARATION

I hereby declare that; this project has been done by me under supervision of Mr. Abdullah Bin Kasem Bhuiyan, Lecturer, department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this project or any part of there has never been submitted anywhere else for the award of any educational degree like, B.Sc., M.Sc., Diploma or other qualifications.

Supervised By



Mr. Abdullah Bin Kasem Bhuiyan

Lecturer

Department of CIS

Daffodil International University

Submitted By



Robiul Islam

ID: 182-16-254

Department of CIS

Daffodil International University

ACKNOWLEDGEMENT

First of all, I'm so much grateful to my almighty Allah, he makes me possible to complete this project successfully.

Then I want to thanks my parents and family for their support, courage and sympathy they shown to me during my entire course. I cannot successfully complete this project without their heartiest support. I am especially indebted to Mr. Abdullah Bin Kasem Bhuiyan, Lecturer, Department of Computing Information System, Daffodil International University, Dhaka, who have been supportive of my career goals and who work actively to make this happen. His patience, guidance, encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

Finally, I would like to thank my entire course mates in Daffodil International University, who took part in this discuss while completing the course work.

ABSTRACT

The “Online Medicine Donation System” is a web-based project that will help the poor people by storing donated medicine. The development process of “Online Medicine Donation System” is started from February, 2021. I had to think about a relevant problem regarding to our poverty problem. Then I have tried to develop my idea as detail as much. This documentation is the final result of that idea.

This project is to prepare an online portal which is the Online Medicine Donation System for the collection of unused medicines. So that they can be given to the people who are in need the medicine. So the question is, what this project will do is, collect the unused medicines from the people by the agent who has completely recovered from the illness or disease and does not require the medicines anymore. After the collection of these medicines, we would check the medicines. and After checking, if the medicine is ok then it will be given to the poor needy people directly giving out these medicines for free. There must be a log-in for the User, admin, Agent. Admin can see all details of users who donated medicine.

Table of Contents

Chapter 1 - Introduction.....	1
1.1 Introduction	1
1.2 Document Contents	1
1.2.1 Overview.....	1
1.2.2 Stakeholders & Characteristics.....	2
1.2.2.1 Admin	2
1.2.2.2 Doctor.....	2
1.2.2.3 Donor.....	2
1.2.2.4 Agent.....	2
Chapter 2 - Initial Study.....	2
2.1 Project Proposal.....	2
2.1.1 Introduction	2
2.1.2 Background Study	3
2.1.3 Description of the proposed system.....	3
2.1.4 Feasibility Study.....	3
2.1.4.1 Operational Feasibility	3
2.1.4.2 Technical Feasibility	4
2.1.4.3 Economic Feasibility.....	4
2.1.4.3.1 Web-based Application	4
2.1.5 Market Research Analysis Based on the Feasibility Factors	4
2.1.6 Foundations	5
2.1.6.1 Aims, Objectives & Scope	5
2.1.6.1.1 Aims and Objective of the proposed project.....	5
2.1.6.1.2 Scope of the proposed project	5
2.1.7 Main Goal	5
2.1.8 Priorities feature according to MoSCoW.....	5
2.1.9 Exploration & Engineering	6
2.1.9.1 Based on PRL the final requirements are:.....	6
2.1.10 Deployment.....	6
2.1.11 Conclusion	7
2.2 Background of the project	7

2.3 Problem Area	7
2.4 Problem Solution.....	7
Chapter 3 - Literature Review	8
3.1 Discussion about Problem Domain	8
3.2 Recommended Approach	8
Chapter 4 - Methodology	9
4.1 What to Use	9
4.1.1 Choosing Methodology	9
4.2 Why to Use	9
4.3 Section of Methodology.....	9
4.3.1 Pre-project Phase	10
4.3.2 Feasibility Study Phase	10
4.3.3 Requirement Gathering Phase	10
4.3.4 Requirement Analysis and Prioritization Phase	10
4.3.5 Exploration and Engineering Phase.....	10
4.3.6 Review Phase.....	10
4.4 Implementations Plan.....	10
Chapter 5 - Planning	11
5.1 Project Plan.....	11
5.1.1 Work Breakdown Structure	11
5.1.2 Resource Allocation.....	12
5.1.3 Time Boxing:.....	13
5.1.4 Gantt Chart	13
5.2 Test Plan.....	14
5.2.1 Required Test	14
5.2.1.1 Functional Testing:	14
5.2.1.2 Non-Functional Testing:	14
5.2.2 Test Case	15
5.2.2.1 Black Box Testing.....	15
5.2.2.1.1 Registration	15
5.2.2.1.2 Login	16
5.2.2.1.3 Store Medicine Information	17

5.2.2.1.4 View Prescription	18
5.2.2.1.5 Update Medicine Info	19
5.2.2.1.6 Store Patient Info	20
5.2.2.1.7 Giving Prescription	21
5.2.2.1.8 Logout	22
5.3 Risk Management	23
5.3.1 Risk Identify	23
5.3.2 Risk Assessment	24
5.3.3 Action Plan.....	24
5.3.4 Steps Taken for Possible Risks	25
5.4 Quality Management	25
5.4.1 Rules For Maintaining Quality	25
5.4.1.1 Quality Control.....	25
5.4.1.2 Quality Assurance	25
Chapter 6 - Feasibility	26
6.1 All Possible Feasibility.....	26
6.1.1 Operational Feasibility	26
6.1.2 Technical Feasibility	26
6.1.3 Economic Feasibility	26
6.2 Cost Benefit Analysis	27
6.3 Is DSDM Good or Bad for this Project	28
Chapter 7 - Foundation	28
7.1 Problem Area Identification	28
7.1.1 Interview	29
7.1.1.1 For Donor	29
7.1.1.2 For Seeker.....	29
7.1.1.3 For Doctor:	29
7.1.2 Observations.....	29
7.1.2.1 For Donor	29
7.1.2.2 For Seeker.....	29
7.1.2.3 For Doctor	29
7.1.3 Questionnaires.....	30

7.1.3.1 For Donor	30
7.1.3.2 For Seeker.....	30
7.1.3.3 For Doctor	30
7.2 Specific Problem Area Identification	30
7.3 Possible Solutions.....	30
7.4 Overall Requirement List.....	31
7.4.1 Functional Requirement.....	31
7.4.2 Non-functional Requirements	31
7.5 Technology to be Implemented	31
7.5.1 Client-Server Application Technology	31
7.5.2 Web Application Technology	32
7.6 Recommendation and Justification	32
Chapter 8 - Exploration	33
8.1 Full System Use Case.....	33
8.2 Activity Diagram	34
8.2.1 Activity Diagram for Registration.....	34
8.2.2 Activity Diagram for Login	34
8.2.3 Activity Diagram for Admin.....	35
8.2.4 Activity Diagram for Doctor	35
8.2.5 Activity Diagram for Donor	36
8.3 Requirement Catalogue	36
8.3.1 Registration and Login Requirement Catalogue	36
8.3.2 Medicine Request Requirement Catalogue	37
8.3.3 Medicine Donor Requirement Catalogue	37
8.3.4 Doctor Requirement Catalogue	38
8.3.5 Agent Requirement Catalogue.....	38
8.4 Prioritized Requirement List (PRL).....	39
Chapter 9 - Engineering	40
9.1 New System Modules	40
9.1.1 Donor Login Module	40
9.1.2 Medicine Donation Module	40
9.1.3 View Prescription Module	40

9.1.4 Doctor Give Prescription	40
9.1.5 Admin Store Patients Info Module	41
9.2 Use Case Diagram.....	41
9.3 Class Diagram	42
9.4 Entity Relationship Diagram.....	43
9.5 Sequence Diagram	44
9.5.1 Sequence Diagram for Admin.....	44
9.5.2 Sequence Diagram for Agent.....	45
9.5.3 Sequence Diagram for Doctor	46
9.6 System Interface Design	47
9.6.1 Homepage	47
9.6.2 Login.....	47
9.6.3 Admin Manage Medicine Information	48
9.6.4 Medicine Info List.....	48
9.6.5 Donor Give Medicine Information	49
9.6.6 Agent Homepage.....	49
9.6.7 Super Admin Manage Agent List	50
9.6.8 Super Admin Homepage.....	50
9.6.9 Super Admin Manage Donor List.....	51
9.6.10 Super Admin Manage Doctor List	51
9.6.11 Doctor List.....	52
9.6.12 Doctor Give Prescription.....	52
9.6.13 Doctor View Prescription List.....	53
9.6.14 Appointment for Patient	53
Chapter 10 - Deployment	54
10.1 Core Module Coding Sample	54
10.1.1 Donor Registration.....	54
10.1.2 Login	54
10.1.3 Agent List.....	55
10.1.4 Admin Manage Medicine Information	55
10.1.5 Donor Give Medicine Information	56
10.1.6 Manage Doctor	56

10.1.7 Doctor Give Prescription	57
10.1.8 Give Medicine Info	57
10.1.9 Make an Appointment.....	57
10.2 Problem Breakdown.....	58
10.2.1 Requirements Analysis	58
10.2.2 Database Design	58
10.2.3 Admin Panel Design and Development	58
10.2.4 Agent Module Design and Development	58
10.2.5 Donors Module Design and Development	58
10.2.6 Doctors Module Design and Development.....	59
10.2.7 Patients Module Design and Development.....	59
10.3 Prioritization of Tasks.....	59
Chapter 11 - Testing.....	60
11.1 Acceptance Testing	60
11.1.1 Functional Testing	60
11.1.1.1 Unit Testing	60
11.1.1.2 Integration Testing.....	60
11.1.2 Non- Functional Testing.....	60
11.1.2.1 Acceptance Testing	60
11.1.2.2 Usability Testing	60
11.1.2.3 Performance Testing	60
11.2 Test Case.....	61
11.2.1 Black Box Testing	61
11.2.1.1 Registration	61
11.2.1.2 Login.....	62
11.2.1.3 Store Medicine Information.....	63
11.2.1.4 View Prescription.....	64
11.2.1.5 Update Medicine Info.....	65
11.2.1.6 Store Patient Info.....	66
11.2.1.7 Giving Prescription	67
11.2.1.8 Logout	68
Chapter 12 - Implementation.....	69

12.1 Training	69
12.2 Big Bang	69
12.3 Scaling	70
12.4 Load Balancing	70
Chapter 13 - Critical Appraisal and Evaluation.....	70
13.1 Objective Could be Met.....	70
13.1.1 Objective-1.....	70
13.1.1.1 Success rate and others for Registration and login systems for users ..	70
13.1.2 Objective-2.....	71
13.1.2.1 Success rate and others for Medicine donor and seeker management .	71
13.1.3 Objective-3.....	71
13.1.3.1 Success rate and others for Agent and doctor management.....	71
13.2 Objectives totally don't meet or touched	71
13.3 The reasons why the feature could not be touch	71
Chapter 14 - Lessons Learned	72
14.1 Pre Project Review & closing	72
14.2 What I Have Learnt	72
14.3 Problems I have faced	72
14.4 Solution Occurred in project.....	72
Chapter 15 - Conclusion.....	73
15.1 Summary of this Project	73
15.2 Goal of this project	73
15.3 The success of the Project.....	73
15.4 What I have done in my documentation	73
15.5 Value of the Project.....	74
15.6 My Experience	74
Chapter 16 - References	75

List of Tables

Table 1: Cost Estimation of the Web-based solution.....	4
Table 2: Table of the Prioritized Features	6
Table 3: Work Breakdown Structure of the OMDS.....	11
Table 4: Resource allocation list for OMDS.....	12
Table 5: Time Boxing list for OMDS	13
Table 6: Test for Registration	15
Table 7: Test for Login	16
Table 8: Test for Store Medicine Information	17
Table 9: Test for View Prescription.....	18
Table 10: Test for Update Medicine Info	19
Table 11: Test for Store Patient Info.....	20
Table 12: Test for Giving Prescription	21
Table 13: Test for Logout	22
Table 14: Risk Identification Table	23
Table 15: Risk Assessment Table	24
Table 16: Table of Action Plan	24
Table 17: Table of Step Taken for Possible Risks.....	25
Table 18: Cost Estimation of the Web-based solution.....	26
Table 19: Cost Estimation for the project	27
Table 20: Earning Estimation for the project	27
Table 21: Estimated Revenue for five years.....	28
Table 22: Requirement Catalogue for Registration and Login.....	37
Table 23: Requirement Catalogue for Medicine Request.....	37
Table 24: Requirement Catalogue for Medicine Donor	38
Table 25: Requirement Catalogue for Doctor	38
Table 26: Requirement Catalogue for Agent.....	39
Table 27: Prioritized Features list.....	39
Table 28: Login Module.....	40
Table 29: Medicine Donation Module	40
Table 30: View Prescription Module	40
Table 31: Doctor Give Prescription	40
Table 32: Admin Store Patients Info Module	41
Table 33: Black Box Testing for Registration	61
Table 34: Black Box Testing for Login.....	62
Table 35: Black Box Testing for Store Medicine Information	63
Table 36: Black Box Testing for View Prescription.....	64
Table 37: Black Box Testing for Update Medicine Info.....	65
Table 38: Black Box Testing for Store Patient Info.....	66
Table 39: Black Box Testing for Giving Prescription.....	67
Table 40: Black Box Testing for Logout.....	68
Table 41: User Training.....	69

List of Figures

Figure 1: Project Cycle Gantt Chart	14
Figure 2: Use Case Diagram.....	33
Figure 3: Activity diagram for Registration.....	34
Figure 4: Activity Diagram for Login	34
Figure 5: Activity Diagram for Admin	35
Figure 6: Activity Diagram for Doctor.....	35
Figure 7: Activity Diagram for Donor	36
Figure 8: Use Case Diagram.....	41
Figure 9: Class Diagram.....	42
Figure 10: Entity Relationship Diagram	43
Figure 11: Sequence Diagram for Admin	44
Figure 12: Sequence Diagram for Agent.....	45
Figure 13: Sequence Diagram for Doctor	46
Figure 14: System Interface Home Design.....	47
Figure 15: System Interface Login Design	47
Figure 16: System Interface Admin Manage Medicine Information Design	48
Figure 17: System Interface Medicine Info List Design	48
Figure 18: System Interface Donor Give Medicine Info Design.....	49
Figure 19: System Interface Agent Home Design	49
Figure 20: System Interface Manage Agent List Design	50
Figure 21: System Interface Admin Home Design	50
Figure 22: System Interface Manage Donor List Design	51
Figure 23: System Interface Manage Doctor List Design	51
Figure 24: System Interface List of Doctor Design	52
Figure 25: System Interface Donor Give Prescription Design	52
Figure 26: System Interface Doctor View Prescription Design	53
Figure 27: System Interface Appointment for patient Design	53
Figure 28: Code Sample for Donor Registration	54
Figure 29: Code Sample for Login	54
Figure 30: Code Sample for Agent List	55
Figure 31: Code Sample for Manage Medicine Info	55
Figure 32: Code Sample for Donor Give Medicine Info	56
Figure 33: Code Sample for Manage Doctor	56
Figure 34: Code Sample for Donor Give Prescription	57
Figure 35: Code Sample for Give Medicine Info.....	57
Figure 36: Code Sample for Make an Appointment	57

Chapter 1 - Introduction

1.1 Introduction

At present, using of computer science is advancing very fast in the whole world. just like we can say, in this age of computer science, the world is like a small city. Due to this, medical science is also growing on a large scale. So the first priority of the technology is the safety of the human. With a donation of tablets or medicine, by well organizing and controlling, we can save lives and ease suffering. But medicine wastage has become a common issue in this world. The World Health Organization (WHO) claims that more than 50% of medicines are inappropriately prescribed and dispensed. Therefore, WHO (2011) calculated that about 24.9% of the total health expenditure or 1.4 to 1.63% of Gross Domestic Product worldwide was spent on medicines. In Asia, a large proportion of out-of-pocket payments is spent on medicines. For example, in India, Bangladesh, and Vietnam, this share is calculated as 70%. Thus, many medicines are wastage due to non-use. On the other hand, many people cannot afford to buy medicine for recovering their disease due to lack of money, and also many are dying. that's why someone needs to rely on the donors. The internet user is increasing rapidly in the last several years. and that will be increased day by day so that it becomes easier for users to use any website. That's why it is needed a system that will save our time, will automate the entire process, and ensure the record-keeping system. As there are no current donation sites for pharmaceutical items to donate free and get the medicine free in our country. So we can develop a new project to overcome this problem for future which name is 'Online Medicine Donation System or OMDS. With this OMDS, we can easily solve all the above-mentioned trouble and have a user-friendly website.

1.2 Document Contents

1.2.1 Overview

This system is designed to be a facility for maintaining donated medicine, patient information, and helping needy people. The rest of the specifications of the Online Medicine Donation System Portal are in detail.

1.2.2 Stakeholders & Characteristics

1.2.2.1 Admin

Admin plays a major and significant role in this system. Admin will store patient information, store medicine information, after viewing the patient's prescription he will also provide available medicine.

1.2.2.2 Doctor

The doctor will sign up in the system and will see patient information then prescribe medicine to them. The system will save all this information.

1.2.2.3 Donor

The donor of the system will register and upload their unused medicine information with their addresses.

1.2.2.4 Agent

The agent of this system will register donors and store doctor information, patient information, and medicine information.

Chapter 2 - Initial Study

2.1 Project Proposal

2.1.1 Introduction

Online medicine donation system aims to develop an internet-based centralized solution for donating medicines all over the country. It's a common sight that most of the people don't recover from disease for lack of medicine due to money, even many people die. This project is aimed to provide a safe and fast way to take medicine from donors and donate the medicine. This project is set to develop into a system that will make it easier for agents, doctors, donors, and recipients to collaborate and donate the medicines hassle-free.

2.1.2 Background Study

There are no organizations working in our country to collect medicine and distribute the medicine free. There are some international online medicine donation organizations that donated the medicine free like- <https://www.sirum.org> and <https://www.transparenthands.org>. In our country, there are some online medicines organizations that sell their medicine for money to the people. The site is- <https://www.arogga.com>, <https://www.onlinemedicineshop.com>, <https://osudpotro.com>. Since these organizations can't collect free medicines, that's why also they cannot distribute free medicine. There are many people in our country who cannot afford to buy medicine for money, they will get those medicine for free and will get well. Through this site, they will have the opportunity to see a doctor online. This OMDS technology will make the entire process easier, more dependable, and efficient, even while assuring that donors' privacy is protected. So, this system has great demand in the market.

2.1.3 Description of the proposed system

This online Medicine Donation System is a web-based service that gives support and the essential medicine to those in need for free. Because not everyone has the financial backing to acquire medicines. Using this system, this is available for users, with the latest features added at the moment. This system has an easy navigation system for users, that's why the system will become friendly, which will also make the users comfortable for donating and receiving medicines from this system. They can collect medicine from a particular place. There will must be a login for the user, admin, Agent, and Doctor. Admin and Agent can see all details of user donated medicine. There will must be a form, where the new user can register first. Medicine donate form. it is must be tracked in the SQL Server database. When users will donate the medicine, then they have to input the medicine deadline. Also, who receive the medicine, can see the deadline. Users can make an appointment and also get a chance to see a doctor for their disease if they want. Also, user can give their feedback on which thing they liked or something they didn't like. The database and the structure of this system will be reliable and coordinative.

2.1.4 Feasibility Study

2.1.4.1 Operational Feasibility

Operational feasibility is concerned with the proposed system's usability, flexibility of use, and essential functionalities. The proposed system, OMDS, gives sufficient validation and verification for user inputs, a very easy navigation system, and an easy request option, which all make the system much easier to use and understanding. With valid credentials, an administrator, agent, doctor, medicine donor, and medication seeker may easily get access to their portal and perform their responsibilities. (Kenton, n.d.).

2.1.4.2 Technical Feasibility

Users of OMDS can easily manage their operations, according to a technical feasibility study. For development and documentation purposes, I'll be using MySQL database server, Django web framework, HTML - CSS, and Microsoft Office. The system was developed with the most up-to-date and widely used web technologies. This assures that the system will run with any web browser. The development platform for a web-based application is platform independent. We may say that the project OMDS will be technically feasible as resources are reasonable and cost-effective (Kenton, n.d.).

2.1.4.3 Economic Feasibility

2.1.4.3.1 Web-based Application

The whole system and data are stored on a remote server, which can be accessed through the Internet from anywhere at any time (Kenton, n.d.).

Equipment	Cost Per Unit	Cost
2 Desktop (Core i5, 2.4 GHz Processor, 8 GB RAM, 1TB HDD)	1,00,000 /-	2,00,000 /-
Server	10,000 /- per month	10,000 /-
Broadband Internet	1000 /- per month	1000 /-
Total		2,11,000 /-

Table 1: Cost Estimation of the Web-based solution

2.1.5 Market Research Analysis Based on the Feasibility Factors

It depends the demands of the users on the market. Bangladesh is a poor country. In this country, there are many poor and middle-class families who can't afford to buy medicine for their disease. According to WHO (World Health Organizations), there are 25% of people who died can't buy the medicines. Again, according to the WHO, 50% of medicines are discarded or the expiration date will over. For this reason, there is a huge demand for an automated system for managing safe and free medicine donations. Because in Bangladesh, there isn't any website of organizations, that distributes the medicine for free yet. A user can easily take a doctor's appointment and will be able to dispense the medicine prescribed by the doctor free of cost. As a result, the OMDS will be a new and more effective platform for our country's citizens.

2.1.6 Foundations

2.1.6.1 Aims, Objectives & Scope

2.1.6.1.1 Aims and Objective of the proposed project

The goal of this Online Medicine Donation System is to create a portal for the collection of unused medicine for those people, who need those medicine free. This site must be like users can donate unused medicine to this portal.

That can help the needy people and also the user can donate their unused medicines.

It can be claimed by the portal. They can collect those medicines from a particular place where the donor will give the address.

2.1.6.1.2 Scope of the proposed project

This system is mainly focused on collecting donated medicine and storing medicines information in this system to provide medicine for needy people.

Where donors can donate their medicine for needy people, and system will collect those medicine.

There will be registered doctors for needy people and doctors will prescribe them medicine which medicine will provide from the portal through this system.

Admin and Agent will store all information and update information daily.

2.1.7 Main Goal

- Make an easy-access system for donors and recipients.
- Where people can donate medicine for needy people.
- Make sure users get the medicine for free after seeing a doctor.

2.1.8 Priorities feature according to MoSCoW

The MoSCoW prioritization method is a technique for organizing the project requirements based on the priority level. So, MoSCoW is a prioritization technique used in the requirement analysis stage. It will help to understand the importance of the requirements.

According to MoSCoW method prioritized requirements are given below:

SL	Requirement	Priority
01	Registration & Login for all users	Must Have
02	Admin Login	Must Have
03	Manage Medicine Name	Must Have

04	Manage Medicine Type	Must Have
05	Manage Medicine Info	Must Have
06	Manage Appointments	Must Have
07	Manage Agent	Must Have
08	Manage Doctor	Must Have
09	Manage Services	Should Have
10	Make an Appointment to Doctor	Must Have
11	Give Medicine Info	Must Have
12	Contact to Admin/Agent	Should Have
13	About	Should Have
14	Newsletter	Could Have

Table 2: Table of the Prioritized Features

2.1.9 Exploration & Engineering

2.1.9.1 Based on PRL the final requirements are:

- ✦ Medicine donor management system
- ✦ Medicine seeker management facilities
- ✦ Doctor and Agent manipulation
- ✦ Medicine list management
- ✦ Location-based medicine donor searching facility by Agent
- ✦ Make an appointment to doctor
- ✦ Manage Doctor and Agent list

2.1.10 Deployment

In this stage, I'm going to implemented the recommended solution. I will to design and develop the database and web application for the proposed system. I will to develop the system using the DSDM approach. I'm going to develop this system using MySQL for database implementation, the Django web framework, HTML - CSS.

2.1.11 Conclusion

In terms of the above discussion, we can say that the "Online Medicine Donation System" will be very helpful for people who can't afford to buy medicines. From where the donor will give the address, an agent will collect the medicine very carefully and cautiously. So, we can say that this system will have a huge impact on our country.

2.2 Background of the project

There are no organizations working in our country to collect medicine and distribute the medicine free. There are some international online medicine donation organizations that donated the medicine free like sirum and transparent hands. In our country, there are some online medicines organizations that sell their medicine for money to the people. The organization is- arogga, onlinemedicineshop, osudpotro. Since these organizations can't collect free medicines, that's why also they cannot distribute free medicine. There are many people in our country who cannot afford to buy medicine for money, they will get those medicine for free and will get well. And through this site, they will have the opportunity to see a doctor online. This system will make the whole process easy, reliable, and effective and also ensure the safety of donors' privacy. So, this system has high demand in the market. For this reason, there is more demand for integrated and centralized free medicine donation systems like this Online Medicine Donation System that offers individual Medicine donors and seekers for free, agent organization facilities, and appointments to a doctor. Because this kind of web application reduces time consumption and also provides an easier way to accomplish the work.

2.3 Problem Area

As soon as the patients recover from their disease, the remaining medicine is of no use. That's why the medicine will be wastage and the cost of the medicine will be very high.

2.4 Problem Solution

To reduce the wastage unused medicine, we will be developing a web application where the unused medicine can be donated to the poor and needy people. These medicine will be donated to the people for free.

Chapter 3 - Literature Review

3.1 Discussion about Problem Domain

This website Online Medicine Donation System comprises of the use of left-over stuff or item which can be used by poor people. But in our country, there is no website or application that donated medicine free for needy people. But in the outside world, very few organizations developed websites like- Sirum, transparenthand. But in our country, there is no any kind of complete solution for free medicine donation and transfusion. Each of the above international organizations develops their system just considering their organizational activities. But in our country, the key issue is there are no organizations and actions, due to a lack of analysis and in-depth evaluation of who collected the unused medicine from the donor and those medicine donated to the needy people. So, the main problem is- creating the whole thing and arranging it according to the process is a huge challenge.

3.2 Recommended Approach

In terms of the above discussion, it is clear that there are no existing websites in our country. So, I have to clear the functionalities and important ideas for the new system. The new system should provide this recommended approach those are given below:

- The system should be User friendly and attractive outlook design.
- Secured and confined access to the donor's, doctor's, agent information.
- Take an appointment with the doctor.
- Contact the Agent or admin about any information.
- Medicines verification identified and approval process.
- Recognition program for the Medicine donors and seeker.

Chapter 4 - Methodology

One of the most efficient ways to manage a successful project is to choose a software development method that the software team can perform very easily. Each methodology has its own strength and weaknesses. Almost all successful development depends on the methodology chosen for any development purpose. This chapter will discuss, which methodology has been chosen for the project and why are chosen

4.1 What to Use

A software development life cycle methodology plays a very important role in a software development process. There are many software development life cycle methodologies such as dynamic systems development model, spiral model, waterfall model, prototype model, agile software development, rapid application development, joint application development, etc.

4.1.1 Choosing Methodology

In terms of the above discussion, Considering the criteria of the project and each software development life cycle methodology, I have decided to use the Dynamic System Development Model (DSDM) methodology for Online Medicine Donation System. Because DSDM methodology always ensures that, the project quality can be completed within the time table and budget also. It is flexible for use in any kind of company or organization. If the client thinks that any change is required for his project during the implementation process, then they can allow changes and respond quickly (GeeksforGeeks, n.d.).

4.2 Why to Use

It is really tough to choose a perfect methodology for this type of project but DSDM helps to achieve the main part of goal for the project. Because DSDM met the requirements on time and budget. Any changes requirements by the client can be changed during the development process. So it will help a lot to complete the project which is Online Medicine Donation System and deliver the output correctly (GeeksforGeeks, n.d.).

4.3 Section of Methodology

This methodology which is DSDM has certain sections. Following this, the development team can easily develop the project. The certain sections are given below-

4.3.1 Pre-project Phase

In this section, the concept of the initial project, as well as the time frame, budget, and basic requirements for the project are created.

4.3.2 Feasibility Study Phase

Feasibility is measured for the proposed system in this section. There are basically three types of feasibilities in this section. Which is - operational, technical, and economic feasibility.

4.3.3 Requirement Gathering Phase

In this section, all requirements for the proposed system which is both the functional and non-functional requirements are gathered from the users.

4.3.4 Requirement Analysis and Prioritization Phase

This section uses a prioritizing approach such as MoSCoW. Because after collecting all the requirements, the collected requirements are analyzed and priority is given to this category through MoSCoW.

4.3.5 Exploration and Engineering Phase

Here all the requirements are explored and the solution for this requirement is developed in an iterative way. That's why this is the main iterative section of this methodology.

4.3.6 Review Phase

At this stage, the developer team provides the system. Users will be testing the system. If there need any kind of change, they return the issue or problem the change to the developer team.

4.4 Implementations Plan

The effectiveness of a project system is primarily determined by implementation planning, because except it, the strategic goals stay inactive. As a result, implementation is a critical phase in the project system since it transforms strategic ideas into action in order to accomplish the objectives. In this testing, the new arrived problem associated with the system is identified and resolved so that the new system will be available for use. in this section the release criteria, configuration, and all plans are decided. after testing these criteria if everything goes well then the system is ready to launch.

Chapter 5 - Planning

5.1 Project Plan

This part is combined with planning of work and time period.

5.1.1 Work Breakdown Structure

In this section we break this project into some small tasks completed within fixed period.

The below chart is description of WBS:

SL	Task Name	Duration(Days)	Start Date	Finish Date
1	Introduction	4 days	Wed 3/11/2021	Sat 06/12/2021
2	Initial Study	5 days	Sun 7/11/2021	Thu 11/11/2021
3	Literature Review	3 days	Fri 12/11/2021	Sun 14/11/2021
4	Methodology	5 days	Mon 15/11/2021	Fri 19/12/2021
5	Planning	6 days	Sat 20/11/2021	Thu 25/11/2021
6	Feasibility Study	5 days	Fri 26/11/2021	Tues 30/11/2021
7	Foundation	3 days	Wed 01/12/2021	Sun 03/12/2021
8	Exploration	8 days	Sat 04/12/2021	Sat 11/12/2021
9	Engineering	8 days	Sun 12/12/2021	Sun19/12/2021
10	Deployment / Development	13 days	Mon 20/12/2021	Sat 01/01/2022
11	Testing	3 days	Sun 02/01/2022	Tues 04/01/2022
12	Implementation	1 days	Wed 05/01/2022	Wed 05/01/2022
13	Critical Appraisal and Evaluation	1 days	Thu 06/01/2022	Thu 06/01/2022
14	Lessons Learned	1 days	Fri 07/01/2022	Fri 07/01/2022
15	Conclusion	1 days	Sat 08/01/2022	Sat 08/01/2022
16	Total	67 days		

Table 3: Work Breakdown Structure of the OMDS

5.1.2 Resource Allocation

It is one of the most significant part of planning phase. I perform all the roles because this is an academic project. Resource allocation for the OMDS project as follows:

SL	Task	Duration(Days)	Resource
1	Introduction	4	Analyst, User
2	Initial Study	5	Analyst
3	Literature Review	3	Analyst
4	Methodology	5	Analyst, User
5	Planning	6	Analyst, Team Leader
6	Feasibility Study	5	Analyst, User
7	Foundation	3	Analyst, Designer, Developer
8	Exploration	8	Analyst, Designer, Developer
9	Engineering	8	Analyst, Designer, Developer
10	Deployment / Development	13	Designer, Developer
11	Testing	3	Developer, Tester, User
12	Implementation	1	Analyst, Developer, User
13	Critical Appraisal and Evaluation	1	Analyst
14	Lessons Learned	1	Analyst, Designer, Developer, Tester
15	Conclusion	1	Analyst

Table 4: Resource allocation list for OMDS

5.1.3 Time Boxing:

In this part, all of the tasks of the project are divided into some time boxes with a time duration given below here-

Time Box	Task	Resource
TB1	Introduction	Analyst, User
	Initial Study	Analyst
	Literature Review	Analyst
TB2	Methodology	Analyst, User
	Planning	Analyst, Team Leader
	Feasibility Study	Analyst, User
TB3	Foundation	Analyst, Designer, Developer
TB4	Exploration	Analyst, Designer, Developer
	Engineering	Analyst, Designer, Developer
TB5	Deployment / Development	Designer, Developer
	Testing	Developer, Tester, User
TB6	Implementation	Analyst, Developer, User
TB7	Critical Appraisal and Evaluation	Analyst
	Lessons Learned	Analyst, Designer, Developer, Tester
15	Conclusion	Analyst

Table 5: Time Boxing list for OMDS

5.1.4 Gantt Chart

Graphical representation of the project schedule is Gantt Chart. It shows the progress of the project from beginning to the end.

Gantt chart of Online Medicine Donation Portal:

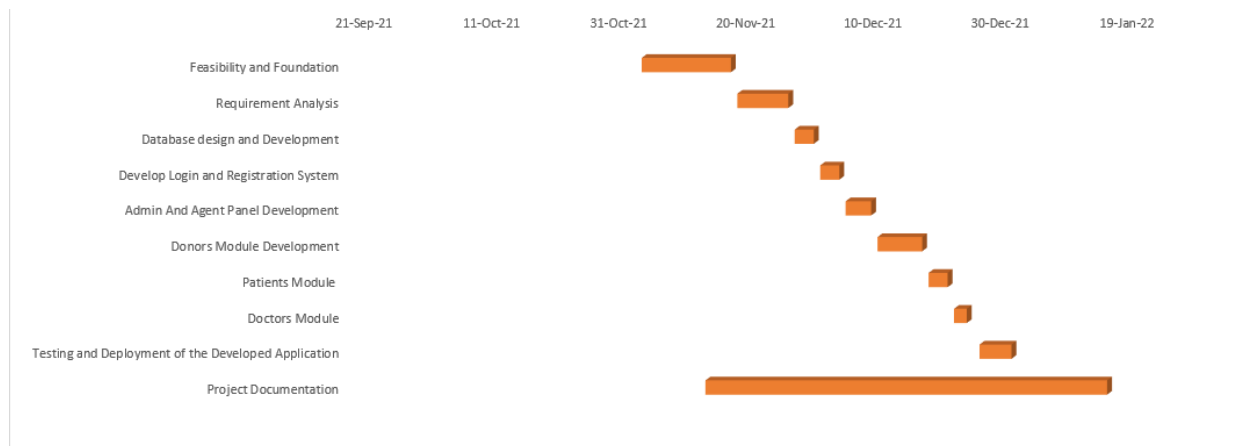


Figure 1: Project Cycle Gantt Chart

5.2 Test Plan

Test plan describe the scopes of testing. Also the activities that need to be performed during project development.

5.2.1 Required Test

There are two types of testing methodology, Functional & non-functional.

5.2.1.1 Functional Testing:

- Unit Testing: Tried on some components of the project during development by developer.
- Integration Testing: This is done during development phase of the project while integrate different components.
- System Testing: After the whole system has been developed, it is tested to ensure that it meets the requirement specification.
- System testing involves evaluating the overall system rather than particular aspects of the design.
- Acceptance Testing: It checks that; the system delivers what was requested. Acceptance Testing is a type of software testing in which a system's acceptability is evaluated.

5.2.1.2 Non-Functional Testing:

- Security Testing: It validates user access and protect application from internal attack like SQL Injection.
- Usability Testing: Usability Testing ensures whether system is usable or not that means it checks the system user-friendliness.

5.2.2 Test Case

5.2.2.1 Black Box Testing

5.2.2.1.1 Registration

Test Case:		Test Case Name: Registration			
System: Online Medicine Donation System		Subsystem:			
Designed by: Robiul Islam		Design Date:			
Executed by: Robiul Islam		Execution Date:			
Description: Unregistered users can register for the system access.					
Pre-condition: - Email address must be unique - Password must be 8 characters					
Step	Email	Password	Response	Pass/Fail	Comment
1	Robiul@gmail.com	181-16-254	Registration Successful	Pass	
2	Raziulgmail.com	181-16-254	Invalid email	Fail	
3	Robiul@gmail.com	1234	Password must contain 6 characters	Fail	
Post-condition: User has been registered successfully.					

Table 6: Test for Registration

5.2.2.1.2 Login

Test Case:		Test Case Name: Login			
System: Online Medicine Donation System		Subsystem:			
Designed by: Robiul Islam		Design Date:			
Executed by:		Execution Date:			
Description: Registered users can Login to the system.					
Pre-condition: Registration Successful.					
Step	Email	Password	Response	Pass/Fail	Comment
1	Robiul@gmail.com	181-16-254	Registration Successful	Pass	
2	Raziul@gmail.com	181-16-254	Invalid email	Fail	
3	Robiul@gmail.com	1234	Password must contain 6 characters	Fail	
Post-condition: Login successful.					

Table 7: Test for Login

5.2.2.1.3 Store Medicine Information

Test Case:		Test Case Name: Store Medicine Info.		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin/Agent can store medicine info.				
Pre-condition: Login to System				
Step	Form Fill-up	Response	Pass/Fail	Comment
1	Fill-up required field	Information Stored	Pass	
2	Fill-up the required form except one/more field.	All field must be filled.	Fail	
Post-condition: Information stored successfully.				

Table 8: Test for Store Medicine Information

5.2.2.1.4 View Prescription

Test Case:		Test Case Name: View Prescription.		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin can view prescription.				
Pre-condition: • Login to System • Prescription must be given				
Step	Action	Response	Pass/Fail	Comment
1	Click ‘view Prescription’ button	View Prescription	Pass	
Post-condition: Admin can view prescription.				

Table 9: Test for View Prescription

5.2.2.1.5 Update Medicine Info

Test Case:		Test Case Name: Update Medicine Info.		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin can update medicine info.				
Pre-condition: - Login to System. - Medicine information must be stored.				
Step	Action	Response	Pass/Fail	Comment
1	Update info	Updated Successfully	Pass	
2	Nothing Updated	Nothing Updated	Fail	
Post-condition: Information should be update successfully.				

Table 10: Test for Update Medicine Info

5.2.2.1.6 Store Patient Info

Test Case:		Test Case Name: Store Patient Info.		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin can store patient info.				
Pre-condition: Login to System.				
Step	Action	Response	Pass/Fail	Comment
1	Input in required field. Click on ‘Save Info’ button	Stored Successfully	Pass	
2	Fill-up the required form except one/more field.	All field must be filled.	Fail	
Post-condition: Information should be store successfully.				

Table 11: Test for Store Patient Info

5.2.2.1.7 Giving Prescription

Test Case:		Test Case Name: Give Prescription.		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin can store patient info.				
Pre-condition: Login to System.				
Step	Action	Response	Pass/Fail	Comment
1	Input in required field. Click on ‘Save Info’ button	Stored Successfully	Pass	
Post-condition: Information should be store successfully.				

Table 12: Test for Giving Prescription

5.2.2.1.8 Logout

Test Case:		Test Case Name: Logout		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin can update medicine info.				
Pre-condition: Login to System.				
Step	Action	Response	Pass/Fail	Comment
1	User Clicks on ‘Logout’ Button	Logged Out and take back to the login page.	Pass	
Post-condition: Logged out from system.				

Table 13: Test for Logout

5.3 Risk Management

Risk management is the process of recognizing and evaluating any risks that may arise after the development stage of a project, in order to meet the project's objectives. A project's risk might involve the project's timeframe, money, and overall performance (Mindtools, n.d.). There are many types of risk management those are included-

- Risk Identify
- Risk Assessment
- Action Plan
- Possible Mitigation Process

5.3.1 Risk Identify

- Possible risk finding and track them
- Risk description and cause of the risk
- Impact of the risk

Possible risk descriptions are given below:

Types of Risk	Causes	Impact
Scheduling	• Unintentional Delay • Features Failure or Data Loss	• Submission Late
Database Lost/Error	• Unintentionally Delete Database/Unrecognized Database Error	• Can't get patient info or stored medicine info
Technical Issue	• Not having enough speed • Configuration problem	• Can't run the application • Project delay
Backup	• Unintentionally not backup the project on GitHub or other cloud services or drive	• Lose the project development or the whole project

Table 14: Risk Identification Table

5.3.2 Risk Assessment

Risk assessment explains quantitative and qualitative value of risk included threats.

Types of Risks	Likelihood	Impact	Restore Time
Scheduling	6	8	10
Database Lost/Error	5	5	5
Technical Issue	2	4	3
Backup	4	4	4

Table 15: Risk Assessment Table

5.3.3 Action Plan

Action Plan is done when risk identification and assessment are completed.

Types of Risks	Action	Action Taken by	Action Required
Scheduling	Accurate estimation using Gantt Chart. Using Project Management Tool: Trello	Developer	From the beginning of the development
Database Lost/Error	Normalization/Accurate Relation on table	Developer	During database design and development
Technical Issue	Configure before starting development and connect a secured broadband line	Project Manager	Frequently
Backup	Backup project on GitHub/cloud/drive	Developer	Frequently

Table 16: Table of Action Plan

5.3.4 Steps Taken for Possible Risks

Types of Risks	Likelihood	Impact	Steps Taken
Scheduling	Rare	Medium	Follow Gantt Chart
Database Lost/Error	Likely	High	Used Normalization & ERD
Technical Issue	Frequent	Medium	Ensure secure connection and configuration
Backup	Rare	Low	Backed up frequently on GitHub

Table 17: Table of Step Taken for Possible Risks

5.4 Quality Management

Maintaining the standard during project initiation is quality management. Several characteristic are included in this term-

- ☐ Medicine and patient's information
- ☐ Communication among agents and admin and client
- ☐ Frequent changes on functional and non-functional requirements

5.4.1 Rules for Maintaining Quality

Rules are followed to assure that quality standards are maintained from system to system and from user to user. Quality control and quality assurance are two principles that must be followed while preserving quality.

5.4.1.1 Quality Control

By evaluating of donor registration, donor information, medicine information, patients helping service quality maintained by donor registration and comments.

5.4.1.2 Quality Assurance

By evaluating negative and positive steps, quality assurance has been maintained.

Chapter 6 - Feasibility

6.1 All Possible Feasibility

6.1.1 Operational Feasibility

The term "operational feasibility" refers to the usability of an application's features. OMDP has full operational feasibility and very use to use. It has smooth UI and navigation. Easy self-registration process, which gives donor a hassle free system. Agent can input information so easily (Simplilearn, n.d.).

6.1.2 Technical Feasibility

Some aspects of technical feasibilities are:

Users of OMDS can easily manage their operations, according to a technical feasibility study. For development and documentation purposes, I'll be using MySQL database server, Django web framework, HTML - CSS, and Microsoft Office. The system was developed with the most up-to-date and widely used web technologies. This assures that the system will run with any web browser. The development platform for a web-based application is platform independent. We may say that the project OMDS will be technically feasible as resources are reasonable and cost-effective (Simplilearn, n.d.).

6.1.3 Economic Feasibility

Web-based Application

The entire software and data that is stored on a remote server, which can be connected to internet from anywhere at any time.

Equipment	Cost Per Unit	Cost
2 Desktop (Core i5, 2.4 GHz Processor, 8 GB RAM, 1TB HDD)	1,00,000 /-	2,00,000 /-
Server	10,000 /- per month	10,000 /-
Broadband Internet	1000 /- per month	1000 /-
Total		2,11,000 /-

Table 18: Cost Estimation of the Web-based solution

6.2 Cost Benefit Analysis

Cost benefit of this project are given below:

Total cost

SL no.	Equipment	1 st Year	2 nd Year	3 rd Year	4 th Year	5 th Year	Total
1	Cost of Web Based Application	90,000	-	-	-	-	90,000
2	Cost of Domain & Hosting	25,000	25,000	25,000	25,000	25,000	1,25,000
3	Expenses for Employee	30,000	30,000	30,000	30,000	30,000	1,50,000
4	Cost of Other Things	20,000	20000	20000	20,000	20,000	1,00000
	Total Cost	1,65,000	75,000	75,000	75,000	75,000	4,65,000

Table 19: Cost Estimation for the project

Total Earn

SL no.	Earn Sector	1 st Year	2 nd Year	3 rd Year	4 th Year	5 th Year	Total
1	Government Tax	1,25,000	1,30,000	1,50,000	1,80,000	2,10,000	7,95,000

Table 20: Earning Estimation for the project

Total Revenue

SL no.	Sectors	1 st Year	2 nd Year	3 rd Year	4 th Year	5 th Year	Total
--------	---------	----------------------	----------------------	----------------------	----------------------	----------------------	-------

1	Total Earnings	₹1,25,000	₹1,30,000	₹1,50,000	₹1,80,000	₹2,10,000	₹7,95,000
2	Total Cost of Equipment	₹1,65,000	₹75,000	₹75,000	₹75,000	₹75,000	₹4,65,000
	Total Revenue	-₹40,000	₹55,000	₹75,000	₹1,05,000	₹1,35,000	₹4,10,000

Table 21: Estimated Revenue for five years

By analyzing the cost-benefit it is shows that the promoting of my Online Medicine Donation System the government will earn a big amount from collecting VAT per year. Also, the revenue will be increased every year. So the project will be helpful and beneficial both for humans and the government of our country.

6.3 Is DSDM Good or Bad for this Project

DSDM is an iterative software development technique that also adds some discipline and structure to the project's process. Because the project may need to accept modifications during development, it should be developed iteratively. It ensures that the work is completed on time and that users are engaged throughout the project's lifecycle.

This is an educational project, and projects that have all essential features must be delivered in a specific amount of time. (GeeksforGeeks, n.d.). So, it is evident that the most effective methodology to follow is DSDM for Online Medicine Donation System.

Chapter 7 - Foundation

7.1 Problem Area Identification

Identify problems for a project is a great improvement for that project. Depending on the usage of this project, where the problem is. So the users of this project will be able to tell where the problem of this project is. So there seems to be some problem from collect of the user which is mentioned below-

7.1.1 Interview

The interview is the best option to know the real problem of the project. When the problems are found, they become easier to solve. So, for the proposed system, some questions that have been raised through interviews with users are given below here-

7.1.1.1 For Donor

- When they donate the medicine, what kind of problem do they face
- When contacting the site for a problem. reply available in time

7.1.1.2 For Seeker

- What is the problem faced when they collecting the medicines.
- When seeing a doctor, facing any problem
- The search process of medicine collection can be identified or need more user-friendly.

7.1.1.3 For Doctor:

- When seeing a patient, facing any problem
- Is there any problem to get the prescription

7.1.2 Observations

Observation is another option for collecting the user requirements and knowing the actual problem they faced. So, the main points of observations are given below:

7.1.2.1 For Donor

- In the existing medicine donation system, check the medicine donation system.
- In the existing medicine donation system, check the contact system.

7.1.2.2 For Seeker

- In the existing medicine donation system, check doctor appointments for the seeker portion.
- In the existing medicine donation system, check the medicine search portion.

7.1.2.3 For Doctor

- In the existing medicine donation system, checking the prescription option.

7.1.3 Questionnaires

Questionnaires are another good option to collect information and ask questions to the users and other people. So the questions are given below here

7.1.3.1 For Donor

- How fast and frequently can you donate medicine?
- How much safety do you feel when you have to share your location for the donation?
- Give your thoughts and options about the donation management system.

7.1.3.2 For Seeker

- How do you collect medicine in an emergency?
- How to see a doctor in case of emergency?
- How do you contact with the organization?

7.1.3.3 For Doctor

- Is it easy to handle patients seeing and prescriptions?
- Would it be easier for you to manage things if the central system is developed?

7.2 Specific Problem Area Identification

Some problems are point out in the problem area identification section by using the information gathering method. Those specific problems are given below which are-

- There is no online medicine donation system for free in the country.
- The rest of the medicines are of no use as soon as the patients recover from the disease. That is why medicines will be wasted and the price of medicines will be much higher.
- Some expensive medicines are not affordable to buy for the poor people.

7.3 Possible Solutions

The proposed solutions of the identified problems are:

- To reduce the wastage unused medicine, I will be developing a web application where the unused medicine can be donated to the poor and needy people.
- These medicine will be donated to the people for free.

7.4 Overall Requirement List

The overall requirement lists are given below here-

7.4.1 Functional Requirement

- Medicine donor manipulation system
- Medicine seeker management facility
- Agent and doctor manipulation system
- Written Location-wise donor searching facility by agent
- Doctor appointment management for patients
- Message sending facility for user
- Medicine List management system

7.4.2 Non-functional Requirements

- User-friendly interface design.
- Maintain proper data security and privacy
- Ensure the data and resources backup.
- To restrict the access, implement the authentication and authorization.
- Maintain Validation and verification system

7.5 Technology to be Implemented

It is very important to become successful through choose the right and accurate technology. For implementing the proposed system, there are some implementation technology options. These are-

7.5.1 Client-Server Application Technology

Client-Server technology is one kind of desktop-based technology that requires the client's device to be installed and the data to be kept on a remote server. (www.microfocus.com., n.d.).

There are some features of the client-server application are given below here-

- ❖ The server and client application need to be maintained.
- ❖ The Application needs to be physically installed.
- ❖ The application isn't portable and always accessible.
- ❖ The users need to purchase the application which is costly.

7.5.2 Web Application Technology

This application program is stored on a remote server and in the client devices there is no need to install anything. So the application is accessed delivered through a browser interface over the Internet (What is Web Application (Web Apps) and its Benefits., n.d.).

There are some features of a web-based application are given here-

- ❖ This can be accessed through various platforms. such as- a desktop, laptop, or mobile.
- ❖ No need to be installed.
- ❖ Can be accessed through multiple browsers.
- ❖ Comparatively low cost.
- ❖ This is accessible from anywhere at any time.

7.6 Recommendation and Justification

In terms of the above discussion, it can be said that for the proposed Online Medicine Donation System, it is necessary to make sure that it is always accessible from all over the world. It is important to ensure that, access should not be a hassle for the user and also it may be accessed by a huge number of people at the same time. And also make sure that the user can access the system with smartphones, laptops, and desktops.

As a result, the client-server application will not be able to fulfill all requirements. So, for the suggested system, the web application is recommended.

Chapter 8 - Exploration

8.1 Full System Use Case

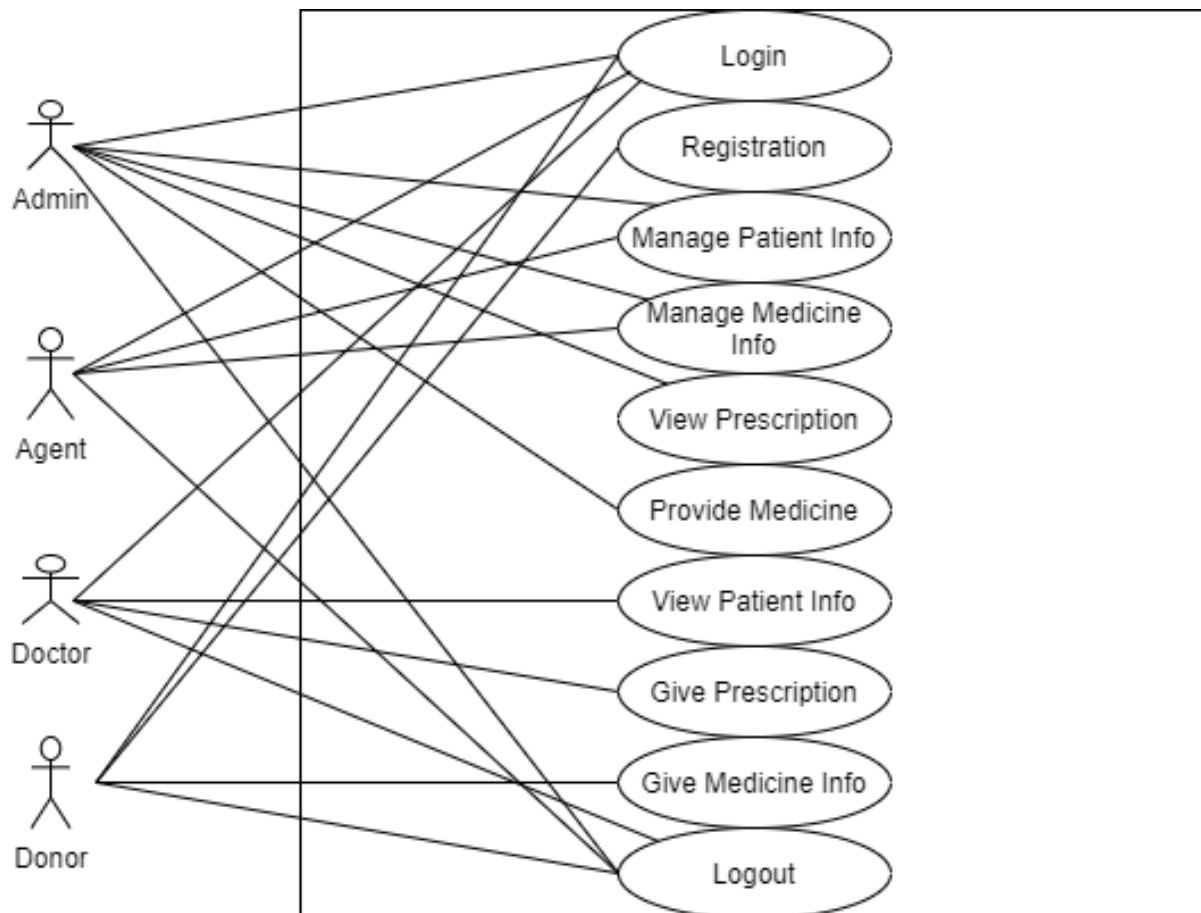


Figure 2: Use Case Diagram

8.2 Activity Diagram

8.2.1 Activity Diagram for Registration

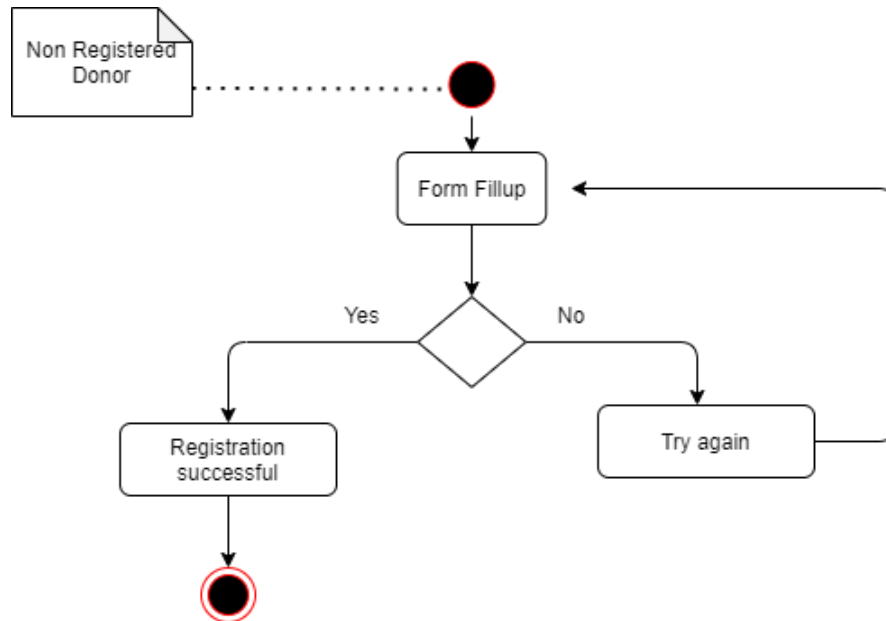


Figure 3: Activity diagram for Registration

8.2.2 Activity Diagram for Login

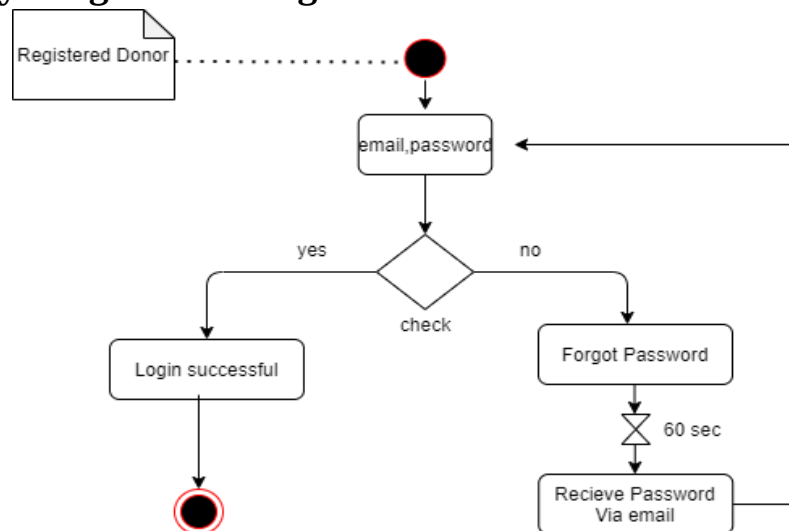


Figure 4: Activity Diagram for Login

8.2.3 Activity Diagram for Admin

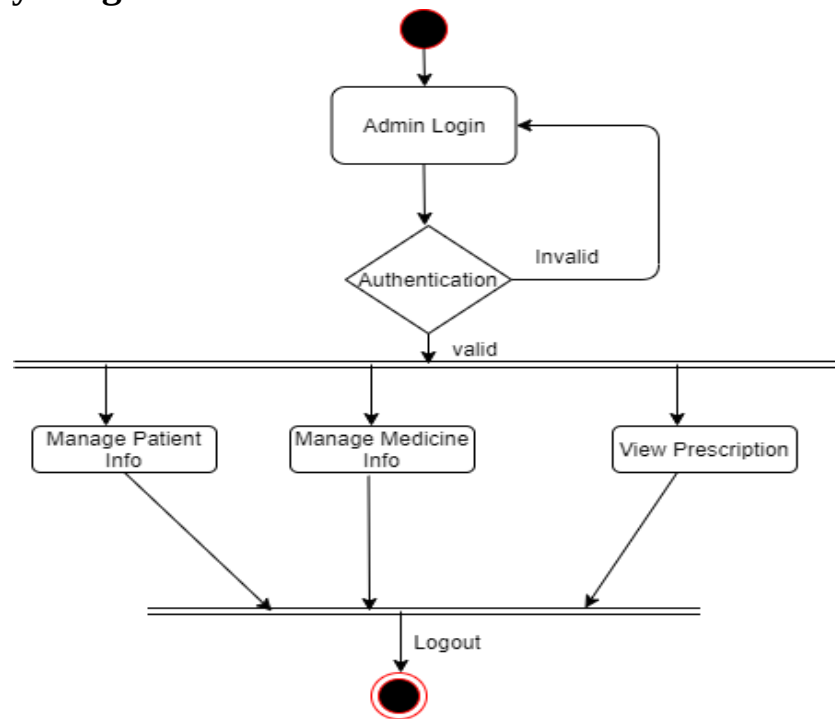


Figure 5: Activity Diagram for Admin

8.2.4 Activity Diagram for Doctor

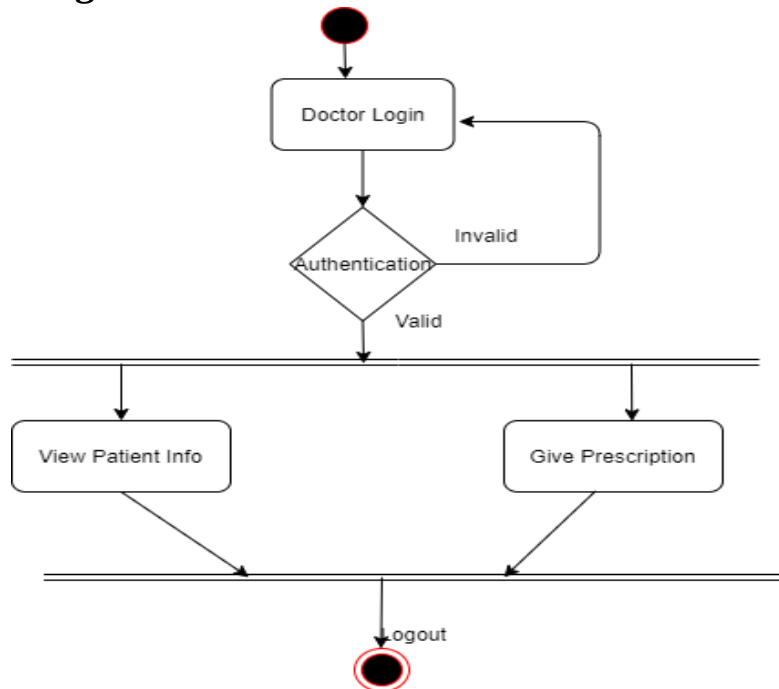


Figure 6: Activity Diagram for Doctor

8.2.5 Activity Diagram for Donor

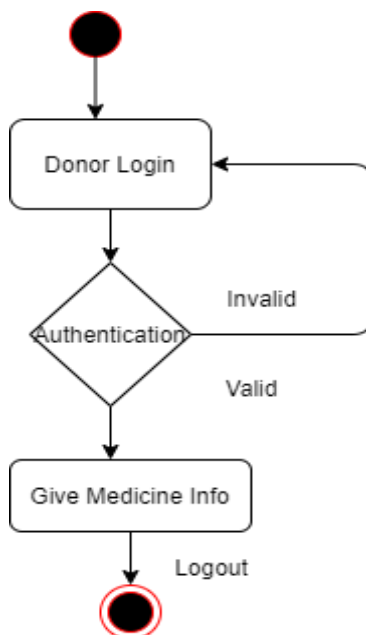


Figure 7: Activity Diagram for Donor

8.3 Requirement Catalogue

Requirement Catalogue is the identified all the requirements of the project. The Requirement catalogue are given below here-

8.3.1 Registration and Login Requirement Catalogue

Source	Sing off:	Priority	Requirement ID:
Admin	All Users	Must	M-001
Functional Requirement Registration and Login for users The system users need to registration first and log in with their id. No users can registered second time with the same id and also they cannot perform without any authentication.			
Non-Functional Requirement			

Description	Target value	Acceptance Value	Comment
Login/registration per day	30000	23000	

Table 22: Requirement Catalogue for Registration and Login

8.3.2 Medicine Request Requirement Catalogue

Source	Sing off:	Priority	Requirement ID:
Seeker	Medicine Seeker	Must	M-002
Functional Requirement Medicine Request for the Seeker The Medicine seeker can make appoint to doctor for their problem. Doctor will give the prescription and agent will give medicine as easily			
Non-Functional Requirement			
Description	Target value	Acceptance Value	Comment
Medicine Request per day	5000	3500	

Table 23: Requirement Catalogue for Medicine Request

8.3.3 Medicine Donor Requirement Catalogue

Source	Sing off:	Priority	Requirement ID:
Donor	Medicine Donor	Must	M-003
Functional Requirement Medicine Donor for the system The Medicine Donor can give their medicine as easily			
Non-Functional Requirement			

Description	Target value	Acceptance Value	Comment
Medicine Donation per day	30000	28000	

Table 24: Requirement Catalogue for Medicine Donor

8.3.4 Doctor Requirement Catalogue

Source	Sing off:	Priority	Requirement ID:
Doctor	Doctor	Must	M-004
Functional Requirement Doctor of the system If any medicine seeker need medicine for their illness, then they make an appoint to doctor. The doctor will give them medicine prescription for their problem.			
Non-Functional Requirement			
Description	Target value	Acceptance Value	Comment
Doctor's prescription per day	1000	600	

Table 25: Requirement Catalogue for Doctor

8.3.5 Agent Requirement Catalogue

Source	Sing off:	Priority	Requirement ID:
Agent	Agent	Must	M-005
Functional Requirement Agent of the system When the doctor prescribes a medicine for a patient's problem, the agent will look at that prescription and give the medicine to the seeker.			

Non-Functional Requirement			
Description	Target value	Acceptance Value	Comment
Agent give medicine per day	20000	18000	

Table 26: Requirement Catalogue for Agent

8.4 Prioritized Requirement List (PRL)

The MoSCoW prioritizing approach is a way for categorizing project requirements according to priority level. In the requirement analysis stage, MoSCoW is a prioritizing method. It will assist in comprehending the significance of the requirements.

Prioritized requirements, according to the MoSCoW technique are given below:

SL	Requirement	Priority
01	Registration & Login for all users	Must Have
02	Admin Login	Must Have
03	Manage Medicine Name	Must Have
04	Manage Medicine Type	Must Have
05	Manage Medicine Info	Must Have
06	Manage Appointments	Must Have
07	Manage Agent	Must Have
08	Manage Doctor	Must Have
09	Manage Services	Should Have
10	Make an Appointment to Doctor	Must Have
11	Give Medicine Info	Must Have
12	Contact to Admin/Agent	Should Have
13	About	Should Have
14	Newsletter	Could Have

Table 27: Prioritized Features list

Chapter 9 - Engineering

9.1 New System Modules

9.1.1 Donor Login Module

User Action	System Action
User click on the login link	A login form will be shown
User have to provide login credentials	System will validate the data
User will click login button	System authenticate the user and if the credentials are accurate then user can go to the dashboard otherwise it will take back to the login page.

Table 28: Login Module

9.1.2 Medicine Donation Module

User Action	System Action
User click on the donate medicine button	A form will be shown
User have to provide accurate information	System will store the data

Table 29: Medicine Donation Module

9.1.3 View Prescription Module

User Action	System Action
Admin will click on the view prescription button	System will show the prescription of patients

Table 30: View Prescription Module

9.1.4 Doctor Give Prescription

User Action	System Action
Doctor click on the give prescription button	System will show prescription form
Doctor have to provide information	System will store the data

Table 31: Doctor Give Prescription

9.1.5 Admin Store Patients Info Module

User Action	System Action
Admin click on the add patients info button	System will show form of patients info
Admin have to provide accurate info	System will store the data

Table 32: Admin Store Patients Info Module

9.2 Use Case Diagram

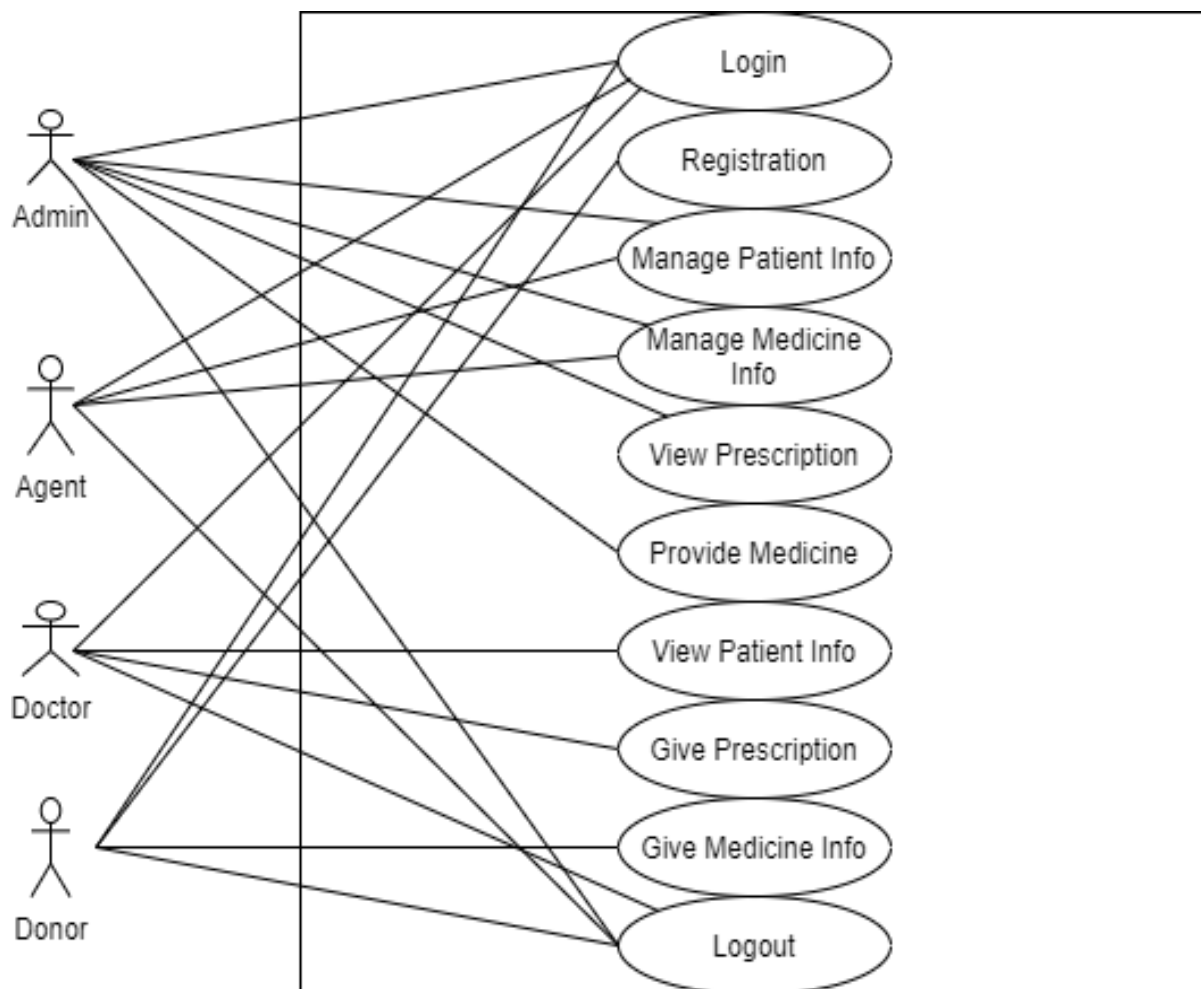


Figure 8: Use Case Diagram

9.3 Class Diagram

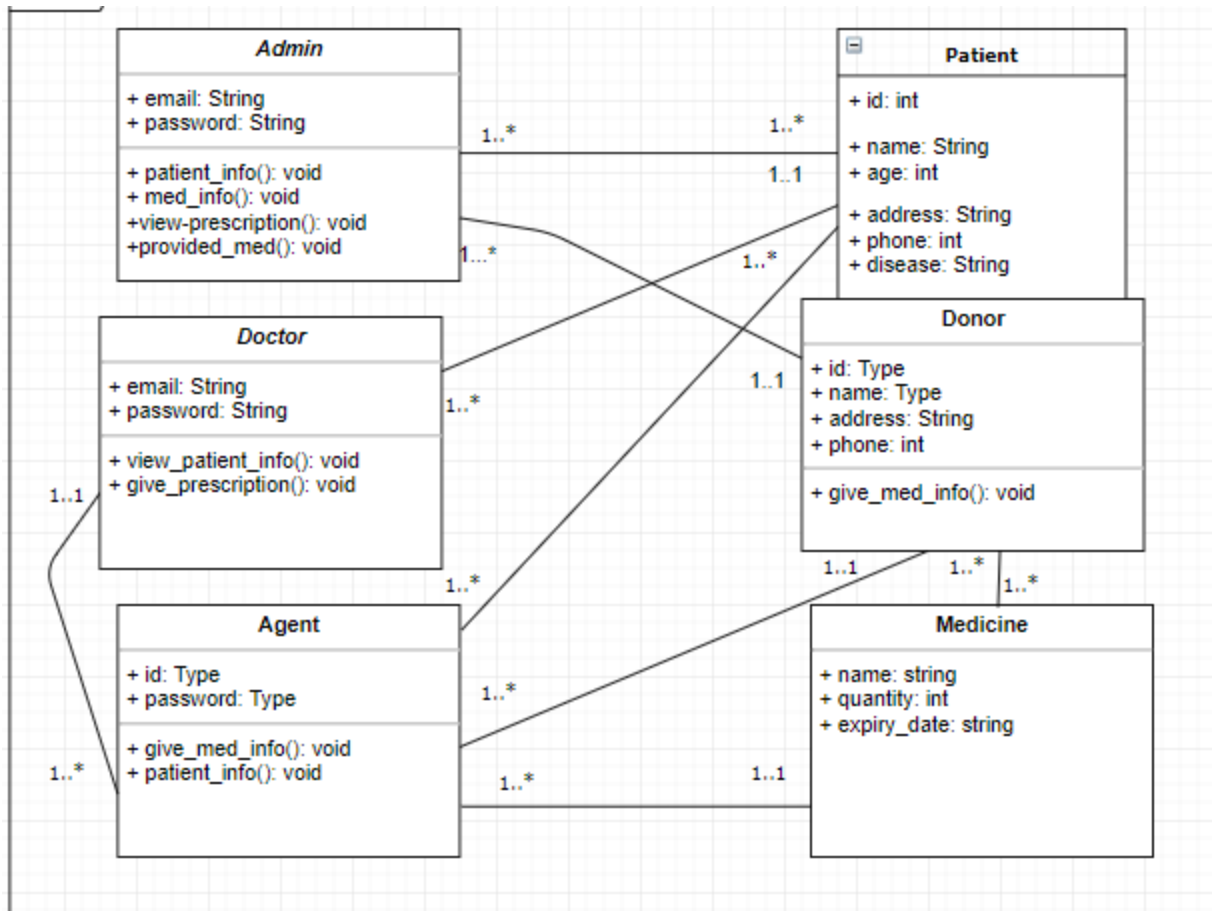


Figure 9: Class Diagram

9.4 Entity Relationship Diagram

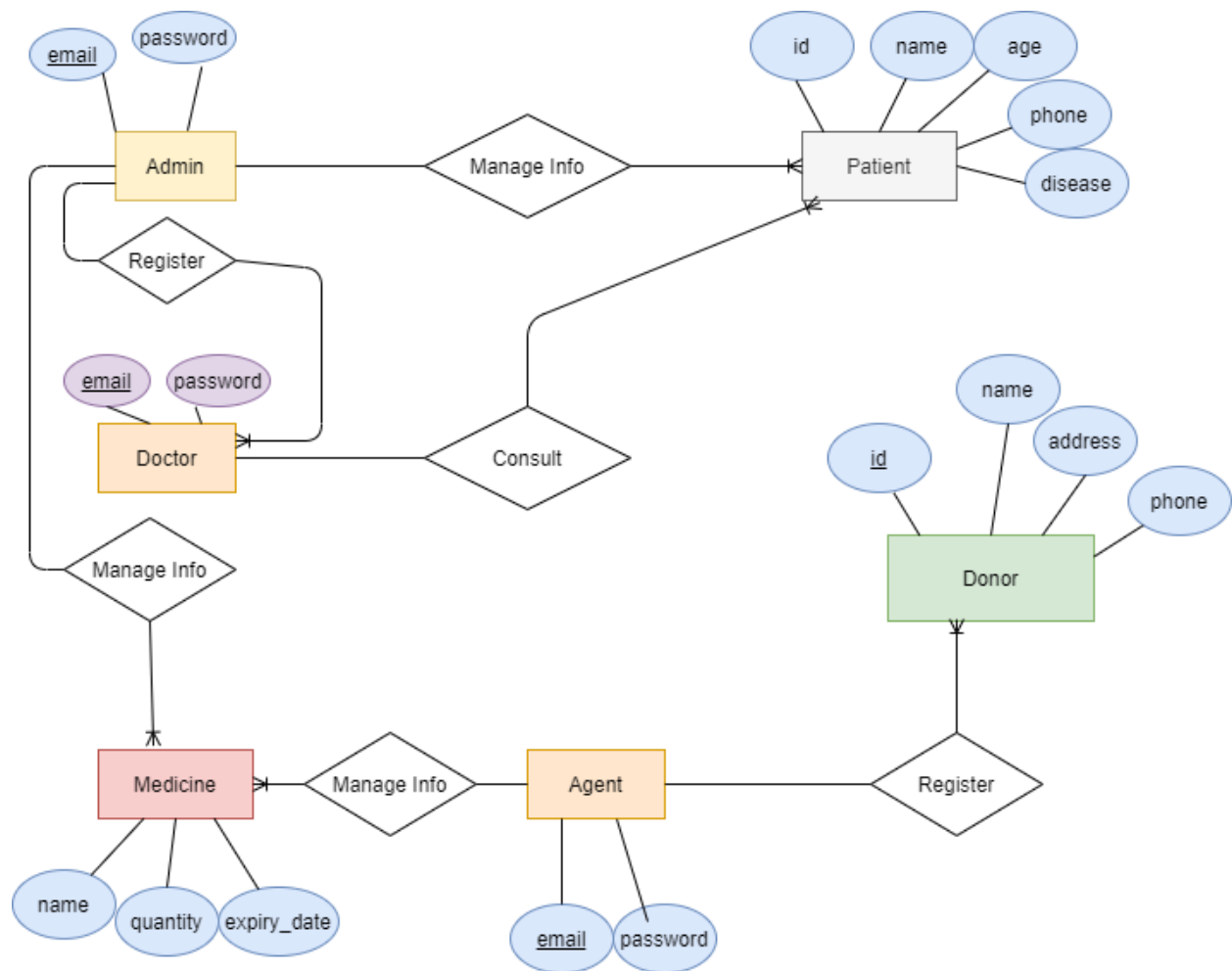


Figure 10: Entity Relationship Diagram

9.5 Sequence Diagram

9.5.1 Sequence Diagram for Admin

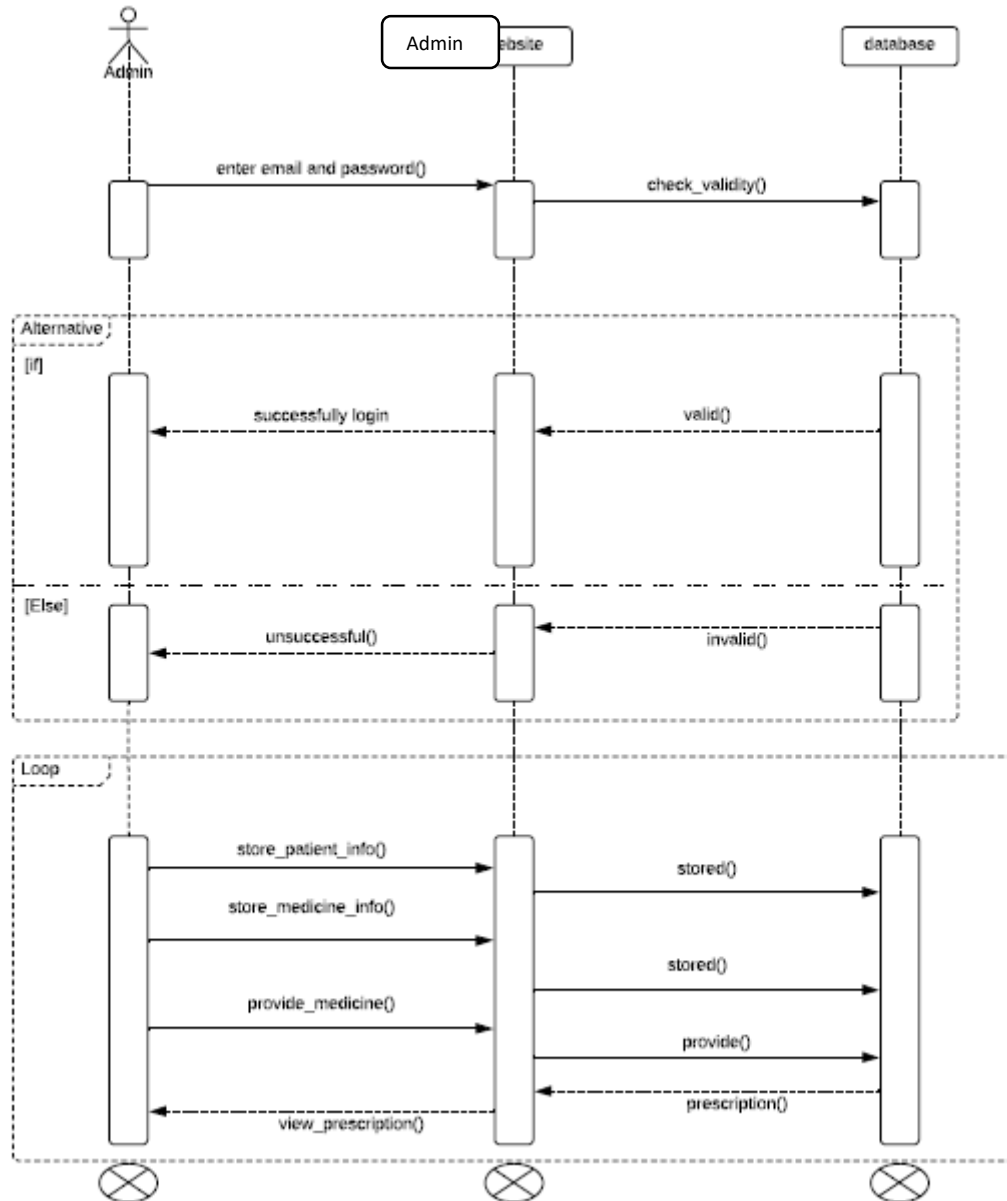


Figure 11: Sequence Diagram for Admin

9.5.2 Sequence Diagram for Agent

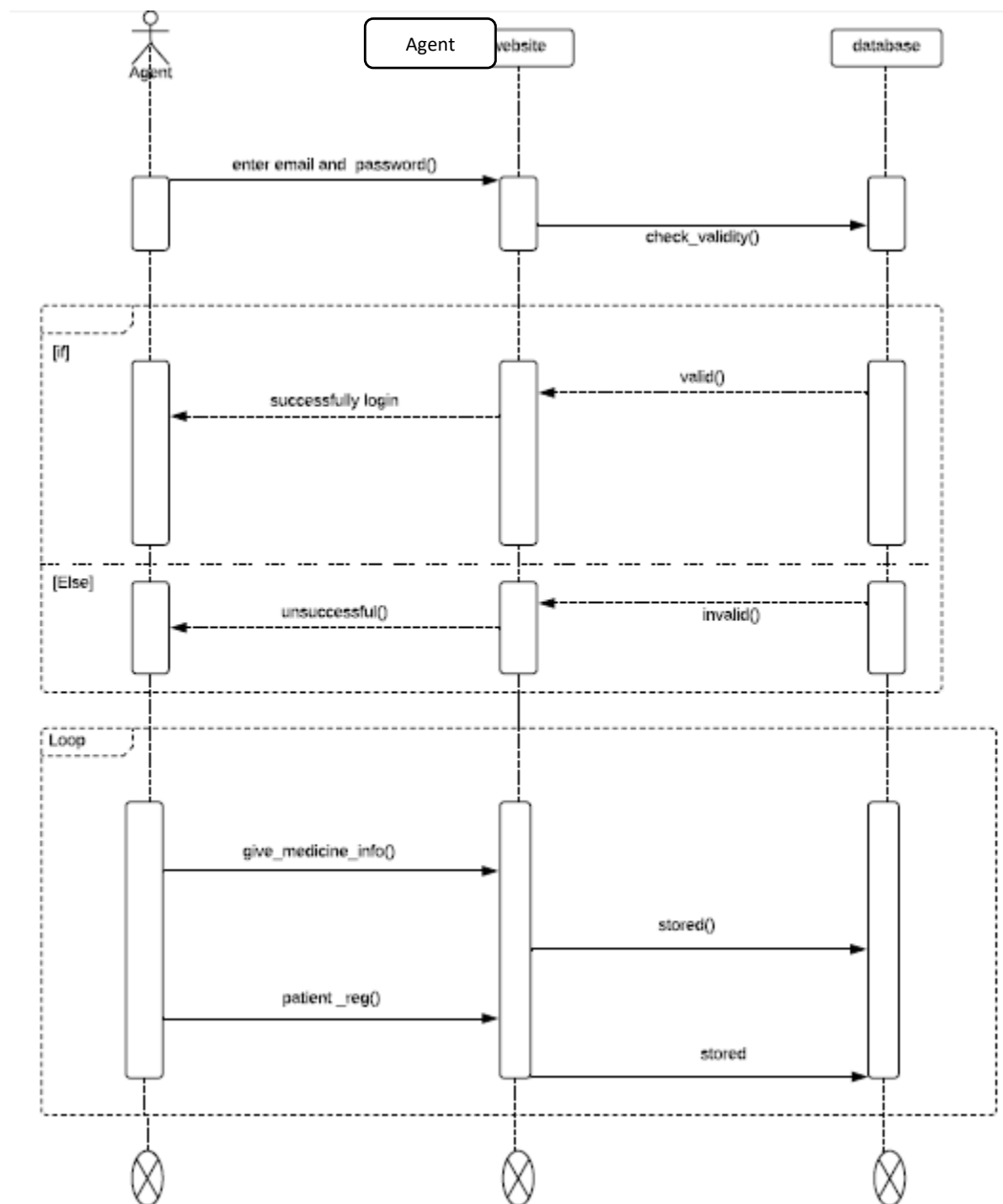


Figure 12: Sequence Diagram for Agent

9.5.3 Sequence Diagram for Doctor

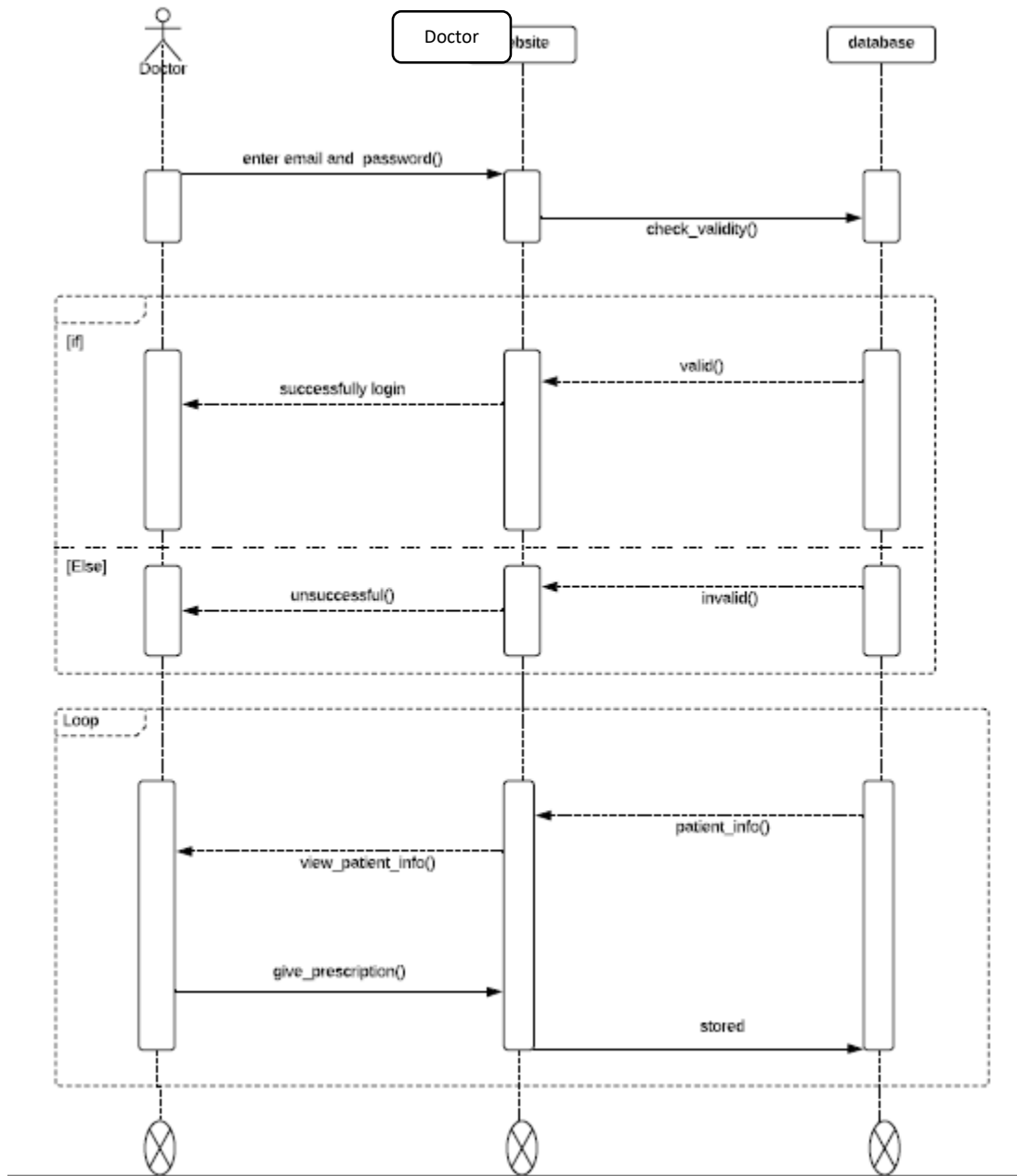


Figure 13: Sequence Diagram for Doctor

9.6 System Interface Design

9.6.1 Homepage

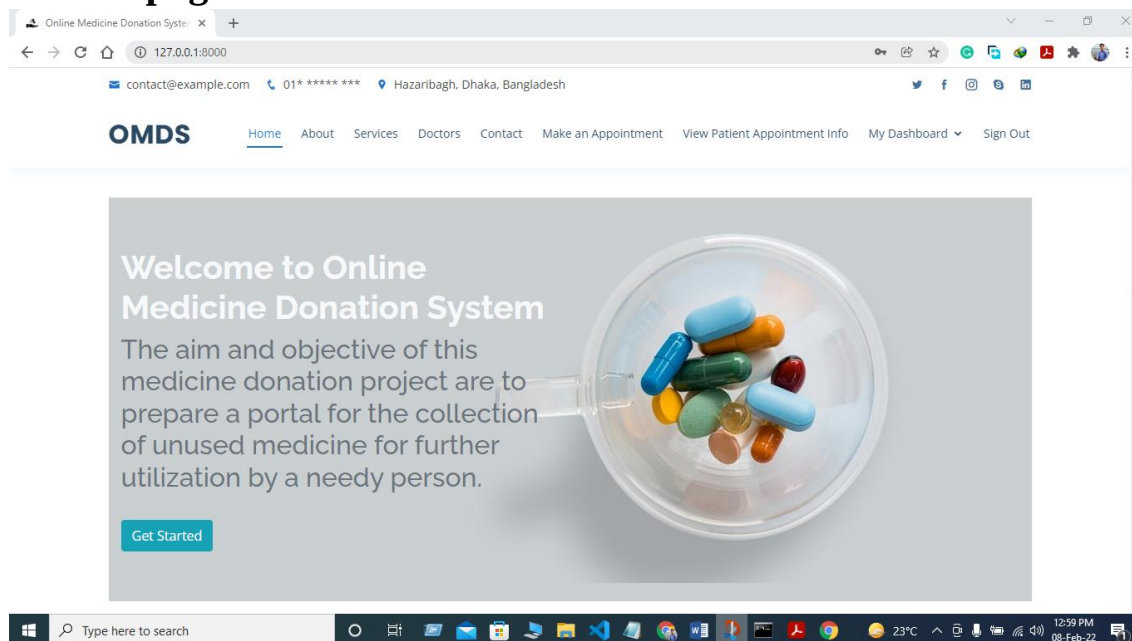


Figure 14: System Interface Home Design

9.6.2 Login

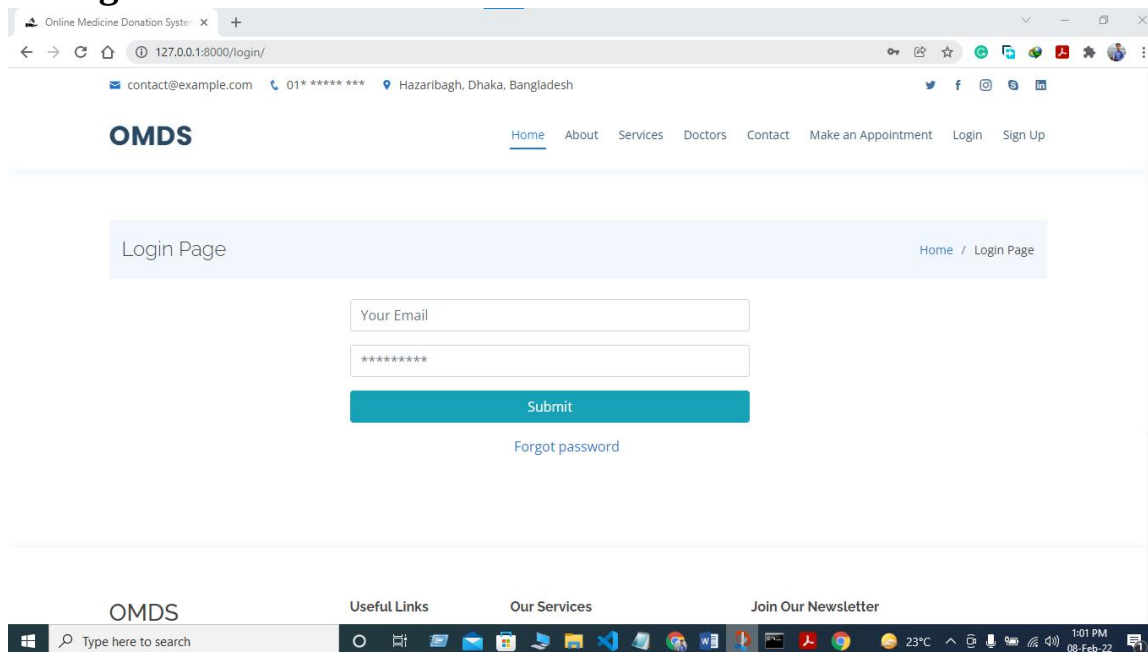


Figure 15: System Interface Login Design

9.6.3 Admin Manage Medicine Information

Super Admin
Robiul_Islam

Home
Manage Medicine Name
Manage Medicine Type
Manage Medicine Info
Manage Doctor
Manage Appointments
Agent
Donor
Contact
Profile

Expiry Date Medicine Info

Add Medicine Info

All Medicine Info List!!!

S. No.	Username	Medicine Name	Medicine Type	No. Of Medicine	Medicine MG	Medicine Expiry Date	Running Date	Update	Delete
1	Robiul_Islam	Napa	Liquid	10	120	Jan. 16, 2023	Feb. 8, 2022	Update	Delete
2	Robiul_Islam	Napa EXTRA	Liquid	25	500	June 16, 2024	Feb. 8, 2022	Update	Delete

Donated Medicine Info List

S. No.	Username	Medicine Name	Medicine Type	No. Of Medicine	Medicine MG	Medicine Expiry Date	Donor Location	Collected
--------	----------	---------------	---------------	-----------------	-------------	----------------------	----------------	-----------

Figure 16: System Interface Admin Manage Medicine Information Design

9.6.4 Medicine Info List

Online Medicine Donation System

contact@example.com 01* ***** Hazaribagh, Dhaka, Bangladesh

Home About Services Doctors Contact Make an Appointment Give Medicine Info My Dashboard Sign Out

Donor Gives Medicine Info Page

Home / Donor Gives Medicine Info

All Medicine Info List!

S. No.	Username	Medicine Name	Medicine Type	No. Of Medicine	Medicine MG	Medicine Expiry Date	Donor Location	Collected
1	rafid	Napa EXTEND	Tablet	15	250	May 27, 2022	Azimpur, Dhaka, Bangladesh	Yes Collected

OMDS Useful Links Our Services Join Our Newsletter

Figure 17: System Interface Medicine Info List Design

9.6.5 Donor Give Medicine Information

The screenshot shows a web browser window with the URL `127.0.0.1:8000/create_donor_give_medicine_info/`. The page is titled "OMDS" and has a navigation bar with links: Home, About, Services, Doctors, Contact, Make an Appointment, Give Medicine Info, My Dashboard, and Sign Out. The main form contains the following fields:

- Medicine name* (text input)
- Medicine type* (dropdown menu)
- Number of Medicine* (text input)
- Medicine MG* (text input)
- Expiry date* (date picker showing dd-mm-yyyy)
- Donor Location (text area)
- Submit (button)

The Windows taskbar at the bottom shows the search bar, task view, and various application icons. The system clock indicates 1:07 PM on 08-Feb-22.

Figure 18: System Interface Donor Give Medicine Info Design

9.6.6 Agent Homepage

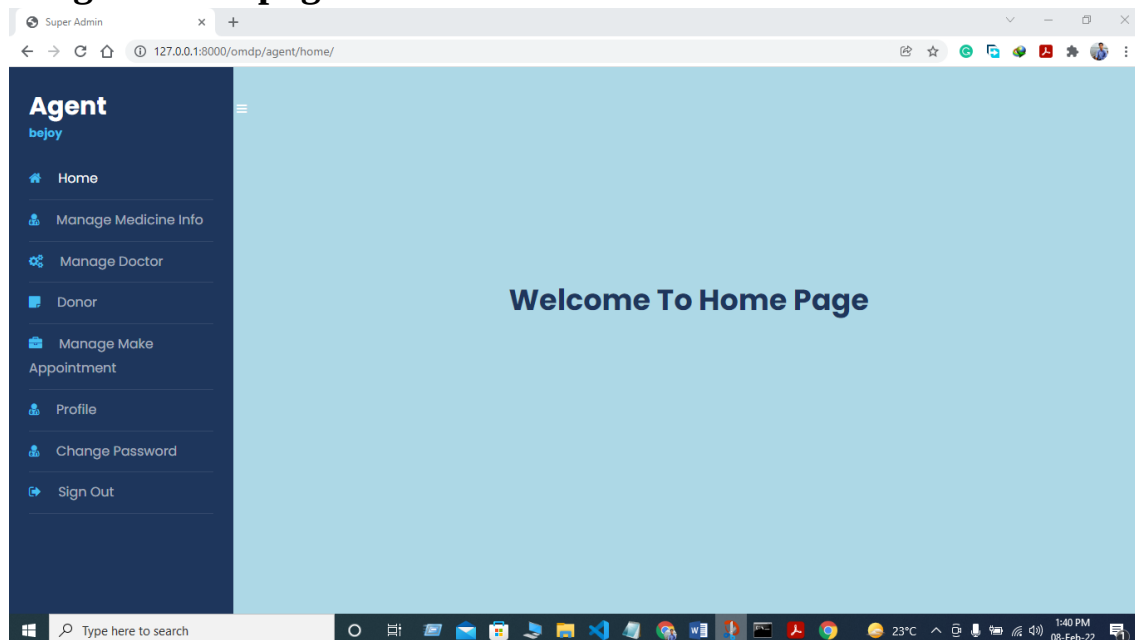


Figure 19: System Interface Agent Home Design

9.6.7 Super Admin Manage Agent List

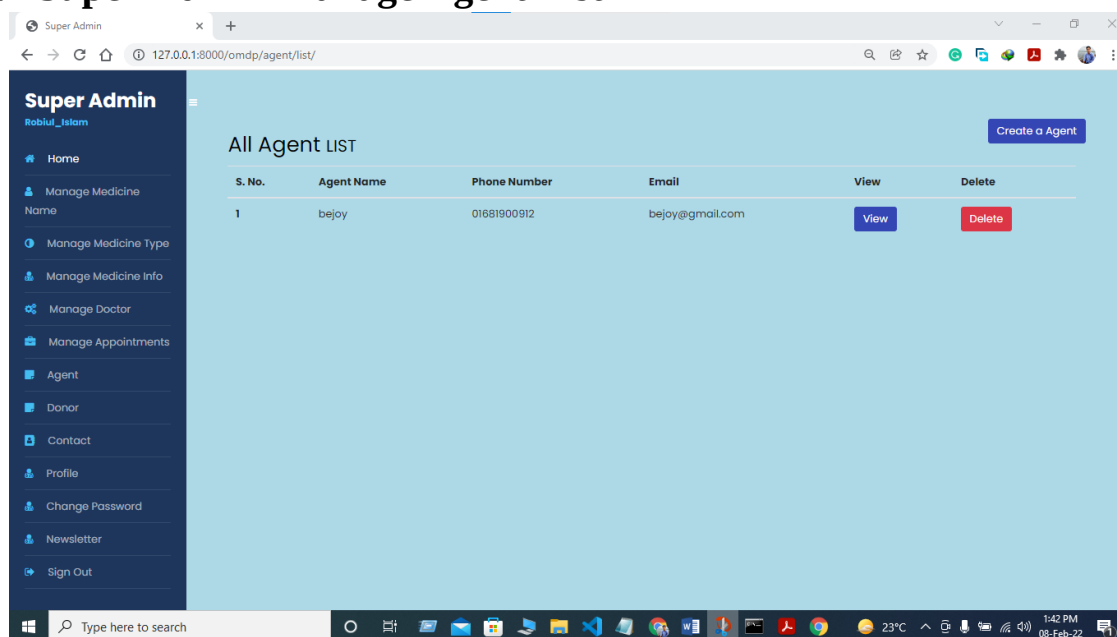


Figure 20: System Interface Manage Agent List Design

9.6.8 Super Admin Homepage

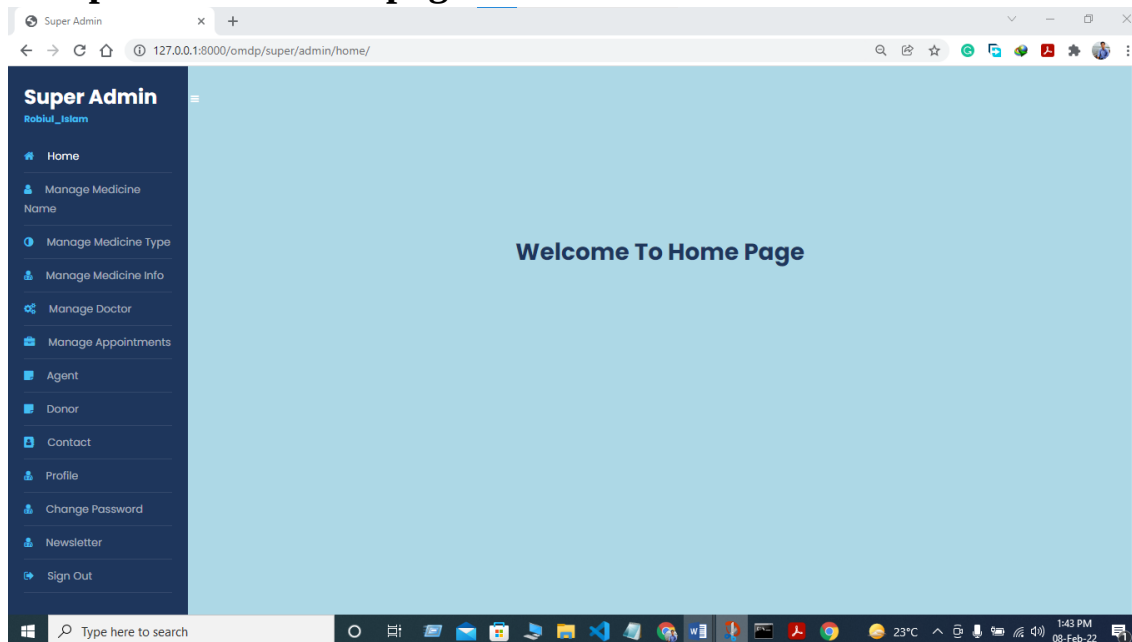


Figure 21: System Interface Admin Home Design

9.6.9 Super Admin Manage Donor List

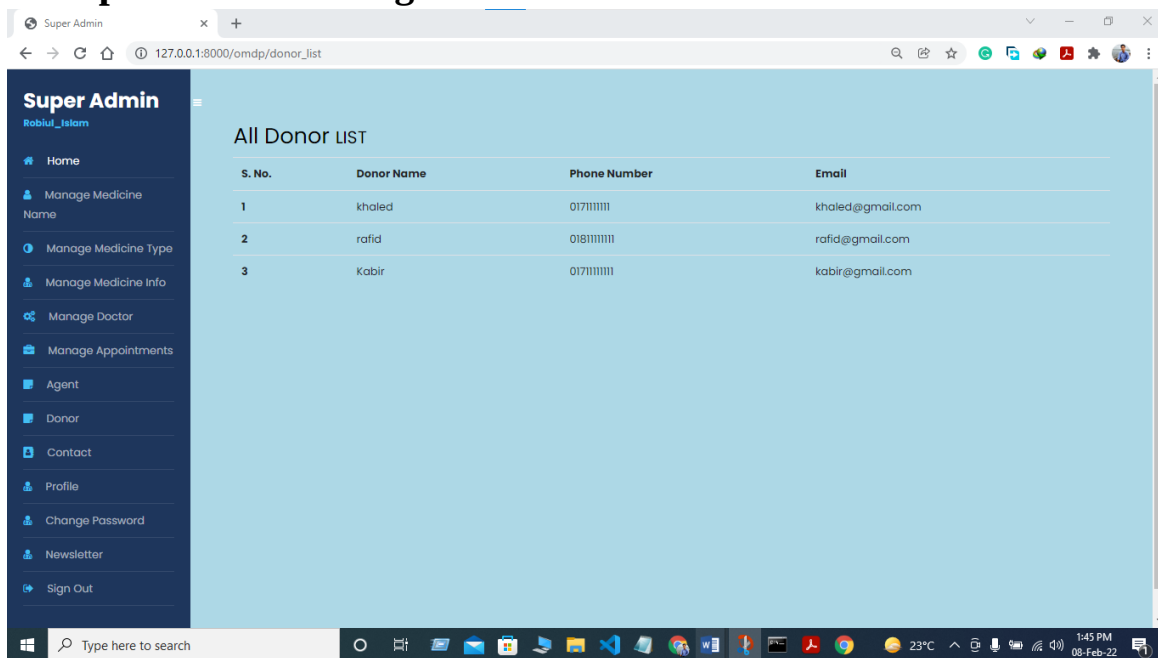


Figure 22: System Interface Manage Donor List Design

9.6.10 Super Admin Manage Doctor List

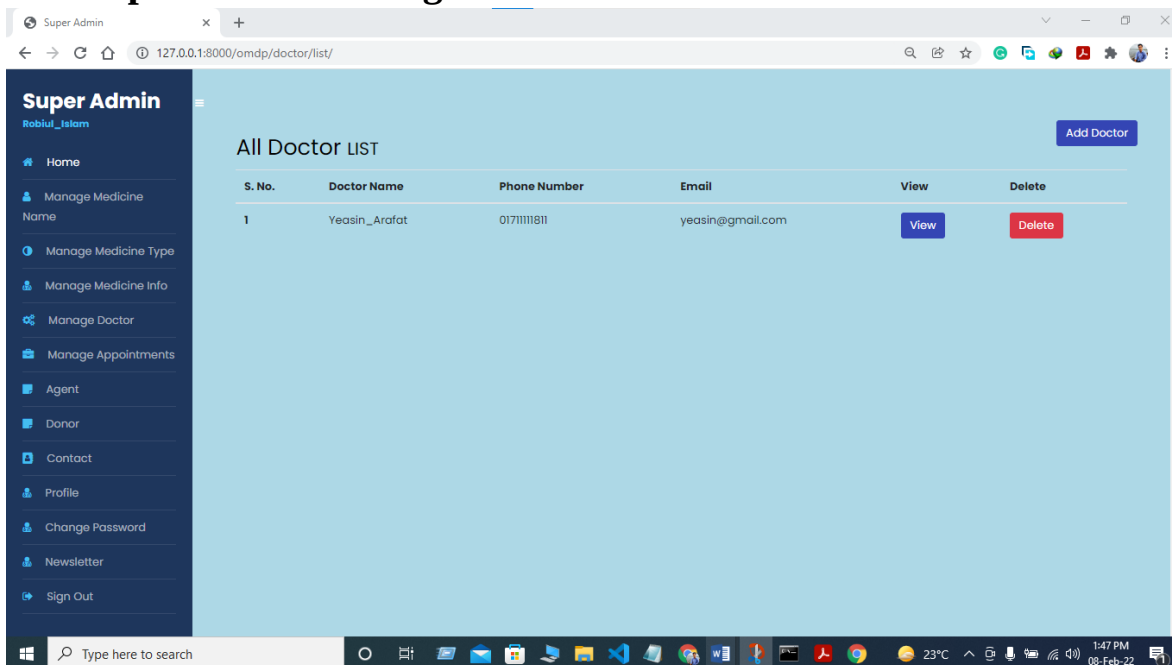


Figure 23: System Interface Manage Doctor List Design

9.6.11 Doctor List

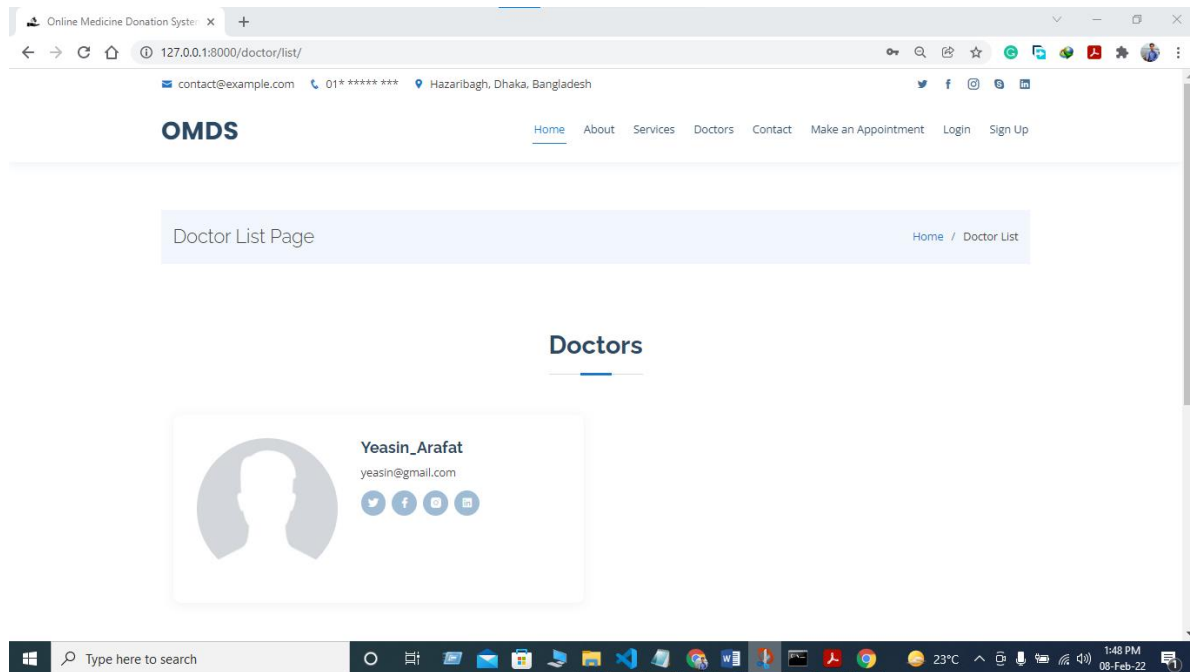


Figure 24: System Interface List of Doctor Design

9.6.12 Doctor Give Prescription

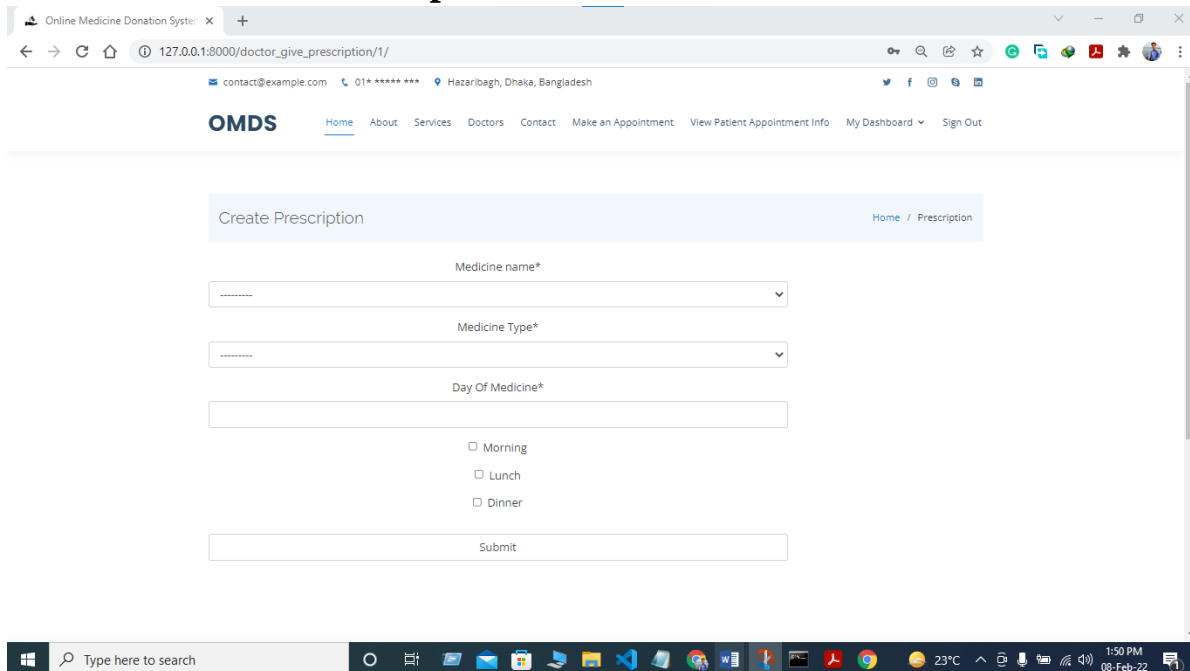


Figure 25: System Interface Donor Give Prescription Design

9.6.13 Doctor View Prescription List

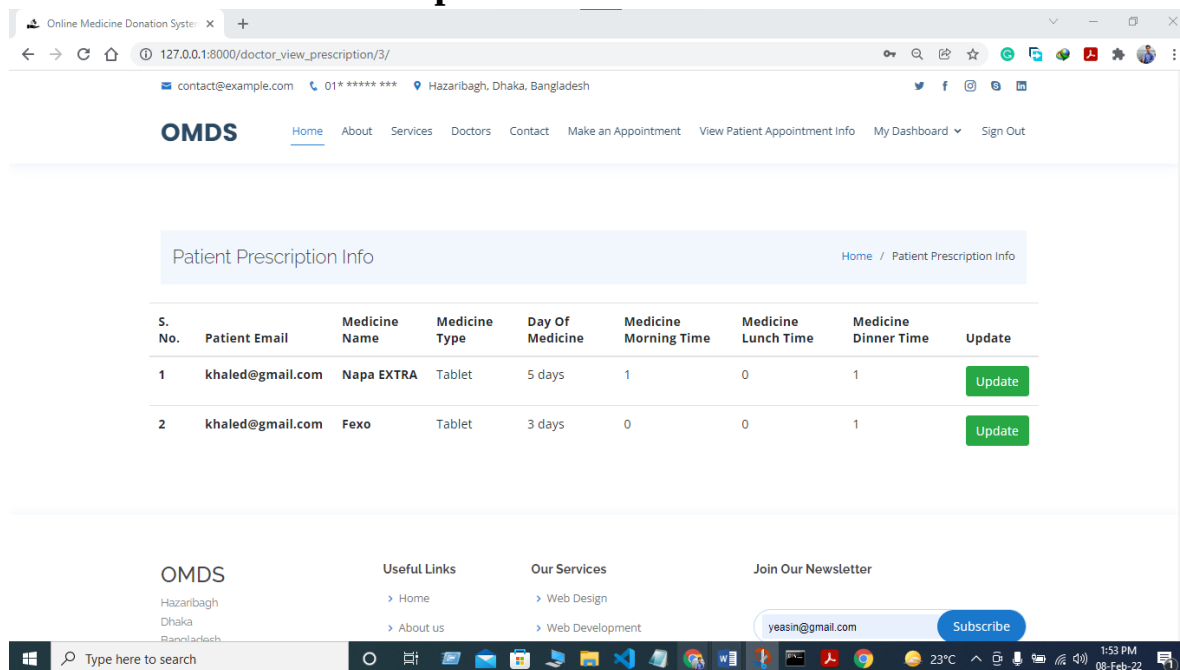


Figure 26: System Interface Doctor View Prescription Design

9.6.14 Appointment for Patient

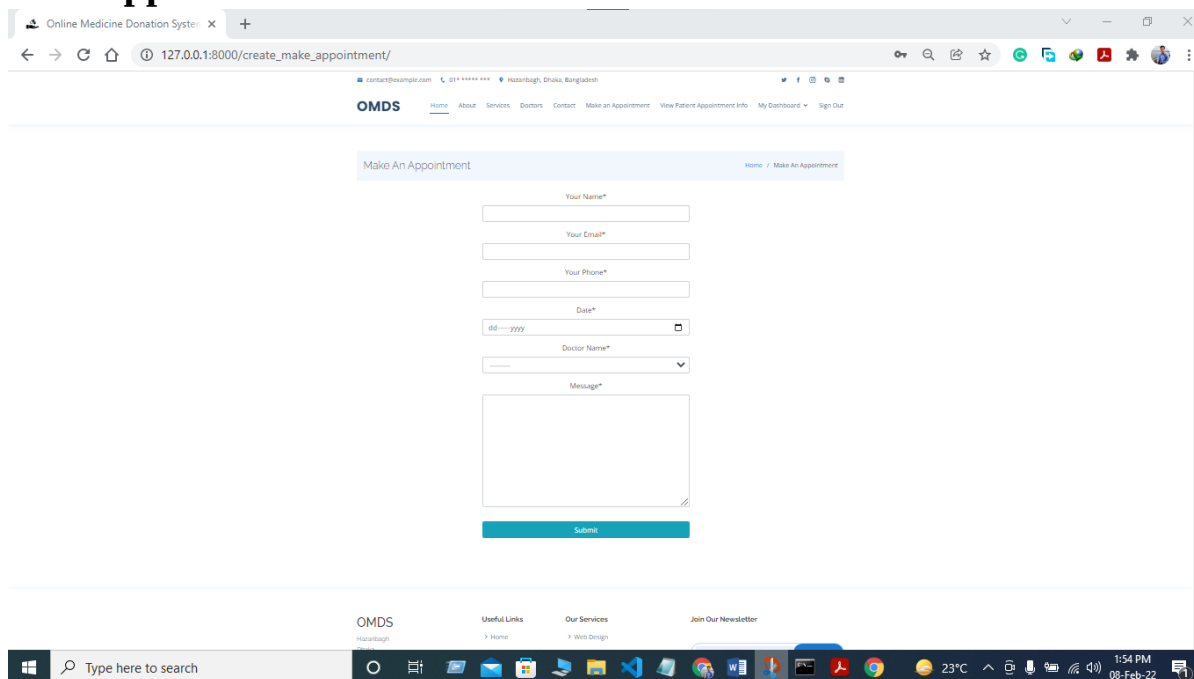


Figure 27: System Interface Appointment for patient Design

Chapter 10 - Deployment

10.1 Core Module Coding Sample

10.1.1 Donor Registration

```

54
55 def donor_registration(request):
56     form = RegistrationForm()
57     if request.method == "POST":
58         form = RegistrationForm(request.POST or None)
59         if form.is_valid():
60             form.save()
61             return redirect(reverse('login'))
62
63     if request.method == "GET":
64         context = {
65             'form': form
66         }
67     return render(request, 'donor_sign_up.html', context=context)
68

```

Figure 28: Code Sample for Donor Registration

10.1.2 Login

```

22
23 def login(request):
24     if request.method == "POST":
25         email = request.POST['email']
26         password = request.POST['password']
27         user = User.objects.filter(email=email)
28         username = user[0].username
29         user_authentication = authenticate(request, username=username, password=password)
30         if user_authentication is not None:
31             auth_login(request, user_authentication)
32             if user_authentication.is_superuser:
33                 return redirect(reverse('super_admin_home'))
34             elif user_authentication.is_agent:
35                 return redirect(reverse('agent_home'))
36             elif user_authentication.is_donor:
37                 return redirect(reverse('base'))
38             else:
39                 return redirect(reverse('base'))
40         else:
41             message = "Your email And Password Is Wrong"
42             context = {
43                 'message': message,
44             }
45             return render(request, 'login.html', context=context)
46     if request.method == "GET":
47         return render(request, 'login.html')
48

```

Figure 29: Code Sample for Login

10.1.3 Agent List

```

73 def list_agent(request):
74     agent = User.objects.filter(is_agent=True)
75     page = request.GET.get('page', 1)
76     paginator = Paginator(agent, 5)
77     try:
78         agent = paginator.page(page)
79     except PageNotAnInteger:
80         agent = paginator.page(1)
81     except EmptyPage:
82         agent = paginator.page(paginator.num_pages)
83
84     context = {
85         'agent': agent
86     }
87     return render(request, 'superadmin/Agent/agent_list.html', context=context)
88

```

Figure 30: Code Sample for Agent List

10.1.4 Admin Manage Medicine Information

```

232
233 def list_medicine_info(request):
234     now_date = date.today()
235     medicine_info = MedicineInfo.objects.filter(username=request.user, expiry_date__gte=now_date)
236     donor = MedicineInfo.objects.filter(username__is_donor=True, expiry_date__gte=now_date)
237     page = request.GET.get('page', 1)
238     paginator = Paginator(medicine_info, 5)
239     paginator1 = Paginator(donor, 5)
240     try:
241         medicine_info = paginator.page(page)
242         donor = paginator1.page(page)
243     except PageNotAnInteger:
244         medicine_info = paginator.page(1)
245         donor = paginator1.page(1)
246     except EmptyPage:
247         medicine_info = paginator.page(paginator.num_pages)
248         donor = paginator1.page(paginator1.num_pages)
249     context = {
250         'medicine_info': medicine_info,
251         'donor': donor,
252         'today': now_date
253     }
254     return render(request, 'superadmin/MedicineInfo/list_medicine_info.html', context=context)
255

```

Figure 31: Code Sample for Manage Medicine Info

10.1.5 Donor Give Medicine Information

```

69
70 def create_donor_give_medicine_info(request):
71     form = DonorMedicineInfoForm()
72     if request.method == 'POST':
73         form = DonorMedicineInfoForm(request.POST)
74         if form.is_valid():
75             m = form.save(commit=False)
76             m.username = request.user
77             m.save()
78             return redirect('base')
79     context = {
80         'form': form
81     }
82     return render(request, 'Donor/GiveMedicineInfo/create_give_medicine_info.html', context=context)
83

```

Figure 32: Code Sample for Donor Give Medicine Info

10.1.6 Manage Doctor

```

20
21 def list_doctor(request):
22     doctor = User.objects.filter(is_doctor=True)
23     page = request.GET.get('page', 1)
24     paginator = Paginator(doctor, 5)
25     try:
26         doctor = paginator.page(page)
27     except PageNotAnInteger:
28         doctor = paginator.page(1)
29     except EmptyPage:
30         doctor = paginator.page(paginator.num_pages)
31
32     context = {
33         'doctor': doctor
34     }
35     return render(request, 'superadmin/Doctor/doctor_list.html', context=context)
36

```

Figure 33: Code Sample for Manage Doctor

10.1.7 Doctor Give Prescription

```

129
130 def doctor_give_prescription(request, patient_id):
131     patient = MakeAppointment.objects.get(id=patient_id)
132     form = GivePrescriptionForm()
133     if request.POST:
134         form = GivePrescriptionForm(request.POST or None)
135         if form.is_valid():
136             p = form.save(commit=False)
137             p.patient_name = patient
138             p.doctor_name = request.user
139             p.save()
140             return redirect(doctor_list_make_appointment_info)
141     context = {
142         'form': form,
143         'patient': patient
144     }
145     return render(request, 'Doctor/GivePrescription/create_prescription.html', context=context)
146

```

Figure 34: Code Sample for Donor Give Prescription

10.1.8 Give Medicine Info

```

70 def create_donor_give_medicine_info(request):
71     form = DonorMedicineInfoForm()
72     if request.method == 'POST':
73         form = DonorMedicineInfoForm(request.POST)
74         if form.is_valid():
75             m = form.save(commit=False)
76             m.username = request.user
77             m.save()
78             return redirect('base')
79     context = {
80         'form': form
81     }
82     return render(request, 'Donor/GiveMedicineInfo/create_give_medicine_info.html', context=context)
83

```

Figure 35: Code Sample for Give Medicine Info

10.1.9 Make an Appointment

```

100
101 def create_make_appointment(request):
102     form = MakeAppointmentForm()
103     if request.method == 'POST':
104         form = MakeAppointmentForm(request.POST)
105         if form.is_valid():
106             form.save()
107             return redirect('base')
108     context = {
109         'form': form
110     }
111     return render(request, 'create_make_appointment.html', context=context)
112

```

Figure 36: Code Sample for Make an Appointment

10.2 Problem Breakdown

To make an easier and efficient system needed break into small tasks as follows –

- * Requirement Analysis
- * Database Design
- * Admin Panel Design & Development
- * Agent Module Design and Development
- * Donors Module Design and Development
- * Doctors Module Design and Development
- * Patients Module Design and Development

10.2.1 Requirements Analysis

- * Gather all the needed requirements
- * Design Prototype

10.2.2 Database Design

- * Create Table and all the schemas
- * Create all the needed keys

10.2.3 Admin Panel Design and Development

- * Design dashboard
- * Login system
- * Management system

10.2.4 Agent Module Design and Development

- * Design dashboard
- * Login system
- * Management system

10.2.5 Donors Module Design and Development

- * Design front-end
- * Develop registration and login system

10.2.6 Doctors Module Design and Development

- * Design doctor's homepage
- * Login
- * Management system

10.2.7 Patients Module Design and Development

- * Patient's appointments page design
- * Develop designed page
- * Appointments list design and develop

10.3 Prioritization of Tasks

Priority list while developing the system to track and not to lose core functionalities development.

List of priorities as follows:

- ✦ Requirements Analysis
- ✦ Database Design and Development
- ✦ Develop Login and Registration System
- ✦ Admin and Agent Panel Development
- ✦ Donors Module Development
- ✦ Patients Module
- ✦ Doctors Module

Chapter 11 - Testing

11.1 Acceptance Testing

11.1.1 Functional Testing

11.1.1.1 Unit Testing

- ✚ Form input fields
- ✚ Saving forms

11.1.1.2 Integration Testing

- ✚ Add patient module
- ✚ Add doctor module
- ✚ Add agent module

11.1.2 Non- Functional Testing

11.1.2.1 Acceptance Testing

- ✚ Beta test by end user

11.1.2.2 Usability Testing

- ✚ Donor module test
- ✚ Patient module test

11.1.2.3 Performance Testing

- ✚ 5000 users can access at a time, anytime without any lag

11.2 Test Case

11.2.1 Black Box Testing

11.2.1.1 Registration

Test Case:		Test Case Name: Registration			
System: Online Medicine Donation System		Subsystem:			
Designed by: Robiul Islam		Design Date:			
Executed by: Robiul Islam		Execution Date:			
Description: Unregistered users can register for the system access.					
Pre-condition: - Email address must be unique - Password must be 8 characters					
Step	Email	Password	Response	Pass/Fail	Comment
1	Robiul@gmail.com	181-16-254	Registration Successful	Pass	
2	Raziulgmail.com	181-16-254	Invalid email	Fail	
3	Robiul@gmail.com	1234	Password must contain 6 characters	Fail	
Post-condition: User has been registered successfully.					

Table 33: Black Box Testing for Registration

11.2.1.2 Login

Test Case:		Test Case Name: Login			
System: Online Medicine Donation System		Subsystem:			
Designed by: Robiul Islam		Design Date:			
Executed by:		Execution Date:			
Description: All registered users can Login to the system.					
Pre-condition: Registration Successful.					
Step	Email	Password	Response	Pass/Fail	Comments
1	Robiul@gmail.com	181-16-254	Registration Successful	Pass	
2	Razuil@gmail.com	181-16-254	Invalid email	Fail	
3	Robiul@gmail.com	1234	Password must contain 6 characters	Fail	
Post-condition: Login successful.					

Table 34: Black Box Testing for Login

11.2.1.3 Store Medicine Information

Test Case:		Test Case Name: Store Medicine Info.		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin/Agent can store medicine info.				
Pre-condition: Login to the System				
Step	Form Fill-up	Response	Pass/Fail	Comment
1	Fill-up all required fields	Information Stored	Pass	
2	Fill-up the required form except one/more field.	All field must be filled.	Fail	
Post-condition: Information stored successfully.				

Table 35: Black Box Testing for Store Medicine Information

11.2.1.4 View Prescription

Test Case:		Test Case Name: View Prescription.		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin can view prescription.				
Pre-condition: • Login to the System • Prescription must be given				
Step	Action	Response	Pass/Fail	Comment
1	Click ‘view Prescription’ button	View Prescription	Pass	
Post-condition: Admin can view prescription.				

Table 36: Black Box Testing for View Prescription

11.2.1.5 Update Medicine Info

Test Case:		Test Case Name: Update Medicine Info.		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin can update medicine info.				
Pre-condition: • Login to the System. • Medicine information must be stored.				
Step	Action	Response	Pass/Fail	Comment
1	Update information	Updated Successfully	Pass	
2	Nothing Updated	Nothing Updated	Fail	
Post-condition: Information should be update successfully.				

Table 37: Black Box Testing for Update Medicine Info

11.2.1.6 Store Patient Info

Test Case:		Test Case Name: Store Patient Info.		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin can store patient info.				
Pre-condition: Login to the System.				
Step	Action	Response	Pass/Fail	Comment
1	Input in required field. Click on ‘Save Info’ button	Stored Successfully	Pass	
2	Fill-up the required form except one/more field.	All field must be filled.	Fail	
Post-condition: Information should be store successfully.				

Table 38: Black Box Testing for Store Patient Info

11.2.1.7 Giving Prescription

Test Case:		Test Case Name: Give Prescription.		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin can store patient info.				
Pre-condition: Login to the System.				
Step	Action	Response	Pass/Fail	Comment
1	Input in required field. Click on ‘Save Info’ button	Stored Successfully	Pass	
Post-condition: Information should be store successfully.				

Table 39: Black Box Testing for Giving Prescription

11.2.1.8 Logout

Test Case:		Test Case Name: Logout		
System: Online Medicine Donation System		Subsystem:		
Designed by: Robiul Islam		Design Date:		
Executed by:		Execution Date:		
Description: Admin can update medicine info.				
Pre-condition: Login to the System.				
Step	Action	Response	Pass/Fail	Comments
1	User Clicks on ‘Logout’ Button	Logged Out and take back to the login page.	Pass	
Post-condition: Logged out from system.				

Table 40: Black Box Testing for Logout

Chapter 12 - Implementation

12.1 Training

SL-no.	Users	Scope of training	Time period	Comments
01	Users or Donators or Seekers	Registering as a donator, log in as a donator, submitted the medicine info which will be given. Registering as a seeker, log in as a seeker, Make an appointment to doctor.	02 hour	The users will be able to tamed this procedure step by step
02	Super Admin	Login as a admin, manage medicine info, manage doctor, manage agent, manage appointment	01 hour	Admin will understand this operation easily.
03	Agent	Log in as an agent, manage appointment, manage medicine info, manage doctor	1.30 hour	Agent will understand this operation easily
04	Doctor	Log in as a doctor, view patient appointment info, give prescription	02 hour	The doctor will be able to understand this procedure gradually

Table 41: User Training

12.2 Big Bang

Big Bang changes the old system to the new system very rapidly. It can be said that the deadline is one day for changing the system. When everyone starts using the new system, they will never use the old system again. The problem is that when the new system has to be replaced within a day, the data from the old management system can be lost and any system can crash. Again, if the system changes, users may not be able to understand the new system. However, the advantage of this system is that there is no need to connect the new system with the old system (Wikipedia, n.d.).

Since I don't have an older version of this system, it has been included on a single site.

12.3 Scaling

The application scaling into consideration no plans were devised to take as it happens to be an academic project (PrimePM., n.d.).

12.4 Load Balancing

Load balancing is a basic networking solution that is used to distribute multiple traffic across server firm. These load balancers improve the availability and responsiveness of applications and prevent server overload. The load balancing process is shared with different servers, so the system runs very fast.

Since the system has four types of users and receive a large number of hits per day of hits per day, a proper load balancing method should be devised to avoid downtime (Contributors, n.d.).

Chapter 13 - Critical Appraisal and Evaluation

13.1 Objective Could be Met

The functionalities which are initially declared those are listed below.

- Registration system and login system for all users
- Medicine donor management and the seeker management
- Agent and doctor management system

13.1.1 Objective-1

13.1.1.1 Success rate and others for Registration and login systems for users

Successfully implemented of registration and login system for all types of users. The donor and seeker can the registration process and the agent and doctor have to be registered by admin. Then all of the candidates can log in into the system. When each user logs in, his/her intent is directed to the dashboard and will only be able to access those pages that are allowed. In this system, for users, I have used the Django authentication system for authenticating and authorizing and the success rate is 100 percent.

13.1.2 Objective-2

13.1.2.1 Success rate and others for Medicine donor and seeker management

The medicine donor and the seeker management system which are fully implemented. first of all, Medicine Seekers can search for the medicine by requesting to the system for an appointment with a doctor. When doctors prescribe them medicine then, by the system, the agent will give them those prescribed medicines. Donors are able to donate medicine through the system at any moment. The success rate of this system is 100 percent.

13.1.3 Objective-3

13.1.3.1 Success rate and others for Agent and doctor management

The agents and doctors are registered by the admin. Doctors prescribe medicine according to the problems of their appointed patients. The agents collect the donated medicine and give the medicine to the patient according to the doctor's prescription. This system is working with a 100 % success rate.

13.2 Objectives totally don't meet or touched

I have implemented most of the functional requirements for this project. But a single target that doesn't contact is AI image recognition for medicine.

13.3 The reasons why the feature could not be touch

The main reason is the short time to finish the activities. If there was time, the work would have been smoother if I had used AI image recognition. Because of this, the users would get a lot of benefits. But due to lack of time, the work could not be done.

Chapter 14 - Lessons Learned

14.1 Pre Project Review & closing

This OMDS is a web-based application. From the beginning, I have to follow predefined structures such as project proposal submission, project title defense, development, and documentation. The system is primarily concerned with safe, timely, and give free instance medicine.

14.2 What I Have Learnt

during this project system, I have learned something huge thing. I have figured out how to deal with structure, arrangement, and database areas and also improved my ability to manage the initiative. Through these projects, I learned about the testing methodology like- the black box testing and the white box testing, the unit testing and the acceptance testing, usability testing, and accessibility testing that will help me in my future professional life. It also a huge increase my programming skills. I have learned the Django framework of Python. This initiative allows me to gather a lot of basic information, which is very necessary to improve life skills.

14.3 Problems I have faced

During my project work, I have faced some difficulties and challenges. As I have followed the DSDM methodology for this system, that's why I have to face and overcome problems iteratively. This means that a problem occurs in the one-time box which solves another time box and task. I also had difficulty creating multi-level authentication and CRUD development in Django. The most suffering problem was maintaining the time boxes which were very extremely difficult to do. After all, the most painful problem was maintaining time boxes which were so suffering for me.

14.4 Solution Occurred in project

When any problem of this project came to me, I tried my best to figure out the solution for it. To implement the multi-authentication, I have used Django authentication system. I have used the Django authentication system that handles both authentication and authorization for passport drivers to allow access over the AP and I also worked hard to maintain the time box by following the deadline.

Chapter 15 - Conclusion

15.1 Summary of this Project

This online Medicine Donation System is an online platform to ensure the safe and timely to take from user unused medicine by this system. All the aspects of the Donor, seeker, agent and doctor are fulfilled via the developed system. If the user wants to donate medicine, the agent will be able to monitor the whole thing. The medicine seeker should make an appointment with the doctor to talk about his problem. Then the doctor will read the appointment message through the system and give the prescription to the patients. Seeing that prescription, the agent will distribute free medicine if the stock is available. The system has developed with Python Django framework, HTML, CSS, JavaScript, jQuery. All relevant diagrams of this system and charts are included with the documents of this system.

15.2 Goal of this project

The ultimate goal and objective of this project were to take a convenient centralized measure to collect instant medicine through an agent in a safe, proper way through the system. The main goals of the project are-

- ❖ The entire system monitoring over the center.
- ❖ An easy-access system for donors and recipients.
- ❖ People can donate medicine to needy people.
- ❖ The users get the medicine for free after seeing a doctor.

15.3 The success of the Project

The development purpose of the project has been successful. Because it is able to meet every goal. When the user wants to donate medicine, the agent can be able to monitor the whole thing. A medicine seeker can make an appointment with the doctor to talk about his problem. Then the doctor can read the appointment message through the system and can give the prescription to the patients. Seeing that prescription, the agent can distribute free medicine if the stock is available. So, the program meets all the initial criteria. That's why this is a well-developed project.

15.4 What I have done in my documentation

For documentation, at first I understood the matter all the required task from beginning of the documentation. So I did time boxing, numerous necessary images, and analyzing. The various goals in the tasks have been properly addressed. The medicine service system will enhance the

experience of collecting medicines from the right donor without the need for extra effort or transportation.

15.5 Value of the Project

Each project has its own value. The medicine service system will enhance the experience of collecting medicines from the right donor without the need for extra effort or transportation. This will depend on how the project is used. However, it has a unique value in the marketplace. The experience gained through the project will enable me to easily work on larger projects, even in professional jobs.

15.6 My Experience

I have gathered a lot of experience by working this project. I have dealt with a number of issues regarding this project, which has given me a lot more experience. I have learned how to manage a complete project very well. It has given me a great experience of how to meet all the criteria of the project in a short period of time.

Chapter 16 - References

- Contributors, W. (n.d.). *Load balancing (computing)*. Retrieved from [https://en.wikipedia.org/wiki/Load_balancing_\(computing\)](https://en.wikipedia.org/wiki/Load_balancing_(computing)).
- GeeksforGeeks. (n.d.). *Dynamic Systems Development Method (DSDM) - GeeksforGeeks*. Retrieved from <https://www.geeksforgeeks.org/dynamic-systems-development-method-dsdm/>.
- <https://www.investopedia.com/terms/f/feasibility-study.asp>. (n.d.).
- Kenton. (n.d.). *Considering a New Venture?* Retrieved from <https://www.investopedia.com/terms/f/feasibility-study.asp>.
- Mindtools. (n.d.). *Risk Analysis and Risk ManagementEvaluating and Managing Risks*. Retrieved from https://www.mindtools.com/pages/article/newTMC_07.htm.
- PrimePM. (n.d.). *Creating a framework for scaling project control practices*. Retrieved from <https://www.primepm.net/scaling-project-control-practices/#:~:text=As%20discussed%20in%20a%20previous> [Accessed 16 Feb. 2022].
- Simplilearn. (n.d.). *Why a Feasibility Study is Important in Project Management*. Retrieved from <https://www.simplilearn.com/feasibility-study-article>.
- What is Web Application (Web Apps) and its Benefits*. (n.d.). Retrieved from <https://searchsoftwarequality.techtarget.com/definition/Web-application-Web-app>.
- Wikipedia. (n.d.). *Big bang adoption*. Retrieved from https://en.wikipedia.org/wiki/Big_bang_adoption.
- www.microfocus.com. (n.d.). *Client/Server Binding*. Retrieved from <https://www.microfocus.com/documentation/object-cobol/oc41books/cmbind.htm> [Accessed 16 Feb. 2022].

Turnitin Originality Report

Processed on: 17-Feb-2022 21:46 +06
ID: 1764656213
Word Count: 9593
Submitted: 1

181-16-254 Robiul Islam By Robiul Islam

Similarity Index	Similarity by Source
20%	Internet Sources: 11% Publications: 3% Student Papers: 13%

9% match (student papers from 29-Dec-2019)
Class: Final_project_Fall2019
Assignment: Project_Fall_2019
Paper ID: [1238624034](#)

4% match (Internet from 05-Jan-2022)
http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5724/171-35-1981%20%2815_%29.pdf?isAllowed=y&sequence=1

1% match (Internet from 01-Apr-2020)
<https://www.slideshare.net/RaihanMahmud5/remote-doctor-project-report>

1% match (student papers from 28-Jan-2021)
[Submitted to Deptford Township High School on 2021-01-28](#)

1% match ()
[Adineh Jafarzadeh, Alireza Mahboub-Ahari, Moslem Najafi, Mahmood Yousefi, Koustuv Dalal. "Medicine storage, wastage, and associated determinants among urban households: a systematic review and meta-analysis of household surveys", BMC Public Health](#)

< 1% match (student papers from 30-Jul-2020)
Class: Final_Project_Spring_2020
Assignment: Final_Project_Spring_2020
Paper ID: [1363960027](#)

< 1% match (student papers from 30-Jul-2020)
Class: Final_Project_Spring_2020
Assignment: Final_Project_Spring_2020
Paper ID: [1363960066](#)

< 1% match (student papers from 29-Dec-2019)
Class: Final_project_Fall2019
Assignment: Project_Fall_2019
Paper ID: [1238620674](#)

< 1% match (Internet from 05-Jan-2022)
http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5799/191-16-429%20%2823_%29.pdf?isAllowed=y&sequence=1

< 1% match (Internet from 07-Apr-2021)
<http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5392/181-16-279%20%2813%25%29.pdf?isAllowed=y&sequence=1>

< 1% match (Internet from 17-Nov-2021)
http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5793/191-16-426%20%2818_%29.pdf?isAllowed=y&sequence=1

< 1% match (student papers from 21-Jan-2014)
[Submitted to University of Greenwich on 2014-01-21](#)

< 1% match (student papers from 28-Apr-2016)
[Submitted to University of Greenwich on 2016-04-28](#)

< 1% match (student papers from 22-May-2018)
[Submitted to University of Greenwich on 2018-05-22](#)

< 1% match (student papers from 28-Apr-2016)
[Submitted to University of Greenwich on 2016-04-28](#)

< 1% match (student papers from 03-Jun-2019)
[Submitted to University of Greenwich on 2019-06-03](#)

< 1% match (student papers from 09-Nov-2016)
[Submitted to Higher Education Commission Pakistan on 2016-11-09](#)