



American International University -Bangladesh

CareBook

Software Project-2

Submitted By

16-31985-2	Bhuiyan, Md. Mahmudul Hasan
16-32084-2	Irfan, Iraz
16-32025-2	Fahim, Hasan Shahria

**Department of Computer Science
Faculty of Science and Technology
American International University -Bangladesh**

19 Nov, 2019

Fall 2018-2019

Declaration

We declare that the submitted project is our original work and has not been submitted in any form for another degree or diploma at any university or other institute of tertiary education. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

We declare that this project does not contain any content that discloses the secret of any organization or related parties. American International University – Bangladesh (AIUB) will not be held liable for any such activities, as for the project is presented as our original work.

Bhuiyan, Md. Mahmudul Hasan
ID: 16-31985-2
Software Engineering

Irfan, Iraz
ID: 16-32084-2
Software Engineering

Fahim, Hasan Shahria
ID: 16-32025-2
Computer Science & Software Engineering

Approval

The project titled “CareBook” has been submitted to the following respected members of the board of examiners of the department of computer science in partial fulfilment of the requirements for the degree of Bachelor of Science in Computer Science & Software Engineering and Bachelor of Science in Software Engineering and has been accepted as satisfactory.

Md. Asiful Islam

Supervisor, Lecturer
Department of Computer Science
American International University-
Bangladesh

Victor Stany Rozario

Assistant Professor
Department of Computer Science
American International University-
Bangladesh

Dr. Mahbubul Syeed

Associate Professor and Head
Department of Computer Science
American International University-
Bangladesh

Prof. Dr. Tafazzal Hossain

Dean, Faculty of Science and Technology
American International University- Bangladesh

Dr. Carmen Z. Lamagna

Vice Chancellor
American International University-Bangladesh

Acknowledgement

This project is the result of our cordial effort and our supervisor's initiative and constant motivation. But first of all, we would like to be grateful to the almighty Allah, who gives us the effort to work on this project. We want to thank our honorable supervisor **Md. Asiful Islam** for his outstanding supervision and support to make this project possible. We would also like to show our gratitude to our honorable department head **Dr. Mahbubul Syeed**, honorable director **Dr. Dip Nandi**, honorable associate dean **Mr. Mashiour Rahman**, honorable dean **Professor Dr. Tafazzal Hossain** and honorable Vice Chancellor **Dr. Carmen Z. Lamagna** of AIUB for their genuine support. We are immensely grateful to them for their comments on an earlier version of the manuscript, although any errors are our own and should not tarnish the reputations of these esteemed persons.

Table of Contents:

Declaration	2
Approval	3
Acknowledgement	4
1.Introduction	10
1.1 Purpose	10
1.1.1 Documentation and history distribution	10
1.1.2 Objectives	11
1.1.3 Benefits	11
1.1.4 Software/Technology Proposed	11
1.1.5 Requirements of the project	12
1.2 Document Conventions	12
1.3 Intended Audience and Reading Suggestions	12
1.4 Product Scope	12
1.5 Deliverable includes in scope	13
1.6 SDLC methods	13
1.6.1 Phase of SDLC	13
1.6.2 SDLC Models	15
1.7 Our chosen SDLC model	19
1.8 Our Related work studies	20
2 Overall Description	20
2.1 Product Perspective	20
2.2 Product Features	22
2.3 User Classes and Characteristics	22
2.4 Operating Environment	22
2.5 Design and Implementation Constraints	22
2.6 User Documentation	23
2.7 Assumptions and Dependencies	23

3	External Interface Requirements	23
3.1	User Interfaces	23
3.2	User Interface Design	23
3.3	Minimum Hardware Requirement	32
4	System Feature	32
4.0	Overall Description	32
4.0.1	User Option	32
4.0.2	Stimulus/Response Sequence	32
4.0.3	Interface Requirement	32
4.0.4	User Interface	32
4.1	System Environment	33
4.2	Functional Requirements Specifications	33
4.2.1	Doctor Use Case	33
4.2.2	Login & Register Use Case	34
4.2.3	Patient Use Case	35
4.2.4	Technician Use Case	36
4.2.5	Admin Use Case	37
4.2.6	Receptionist Use Case	38
4.3	E-R Diagram	39
4.4	Activity Diagram	40
4.5	Schema Diagram	50
4.6	Sequence Diagram	50
4.7	User Story	55
5.0	Software Testing Plan	57
5.1	Test Plan	57
5.1.1	Test plan for registration	57
5.1.2	Test plan for login	57
5.1.3	Test plan for admin	58

5.1.4 Test plan for Doctor	58
5.1.5 Test plan for Patient	59
5.1.6 Test plan for Technician	59
5.1.7 Test plan for Dashboard	60
5.1.8 Test plan for Doctor's Promotion ad upload	62
5.1.9 Test plan for view page	62
5.1.10 Test plan for additional test	63
5.2 Testing Methodology	63
5.2.1 System Testing	64
5.2.2 Regression Testing	68
5.2.3 User Acceptance Testing	69
6. Other Non-functional Requirements	71
6.1 Performance Requirements	71
6.2 Security Requirements	71
6.3 Software Quality Attributes	71
6.4 Business Plan	71
6.4.1 Marketing	71
6.4.2 Objectives	71
6.4.3 Key to Success	72
6.4.4 Mission	72
6.5 Business Analysis	72
6.5.1 Stakeholder	72
6.5.2 Business Objectives	72
6.5.3 Evaluate Options	73
6.5.4 Scope	73
6.5.5 Future Marketing	73
7 Other Requirements	73
8 Project Scheduling	73

Conclusion	74
References	75

Table of Figures:

Figure 1: Phase of SDLC	13
Figure 2: Waterfall Model	15
Figure 3: Incremental life cycle model	16
Figure 4: Prototype Model	17
Figure 5: V Model	18
Figure 6: Agile Model	19
Figure 7: Homepage & login-1	24
Figure 8: Homepage & login-2	24
Figure 9: Sign Up page	25
Figure 10: Sign-In Options	25
Figure 11: Patient Homepage	26
Figure 12: Patient Notification	26
Figure 13: Prescription	27
Figure 14: Doctor Homepage	27
Figure 15: Prescription Form	28
Figure 16: Prescription PDF Format	28
Figure 17: Previous Patient	29
Figure 18: Technician Homepage	29
Figure 19: Technician Profile	30
Figure 20: Admin Homepage	30
Figure 21: Doctor Details	31
Figure 22: Patient Details	31

Figure 23: Full system use case diagram	33
Figure 24: Doctor use case diagram	33
Figure 25: Login & register Use case diagram	34
Figure 26: Patient Use case diagram	35
Figure 27: Technician Use case diagram	36
Figure 28: Admin Use case diagram	37
Figure 29: Receptionist Use case diagram	38
Figure 30: E-R Diagram of the system	39
Figure 31: Doctor Activity Diagram of the system	41
Figure 32: Patient Activity Diagram of the system	43
Figure 33: Technician Activity Diagram of the system	45
Figure 34: Receptionist Activity Diagram of the system	47
Figure 35: Admin Activity Diagram of the system	49
Figure 36: Schema Diagram of the system	50
Figure 37: Sequence Diagram of admin	51
Figure 38: Sequence Diagram of doctor	52
Figure 39: Sequence Diagram of patient	53
Figure 40: Sequence Diagram of technician	54

1. Introduction

CareBook is an ecosystem project for those who are connected to the medical industry. This system will be accessible by anyone related to medical industry. But it will primarily be used by patients, doctors, technicians. Patients normally have to save their medical history and data manually via analogue methods and when sharing their medical data with their doctors they often fail to give them proper information for various reasons. And from the doctors' point of view, a doctor cannot remember all of his patients' details and there is no common communication platform between doctors and patients. Patients always face difficulties searching for the right doctor for their sickness. And technician normally store the patients' test results manually. As there is a chance that patient's test results can be lost or damaged. This proposal includes how we can solve these problems by building a common platform.

1.1 Purpose

The purpose of this document is to build an online platform for all medical professionals. This document will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external systems.

1.1.1 Documentation and history distribution:

Time Duration	Revision date	Descriptions of changes	Contributors
4 months	15.07.19	N/A	Bhuiyan, Md. Mahmudul Hasan Irfan, Iraz Fahim, Hasan Shahria

Table 1.1.1

Recipient Name	Recipient Organization	Distribution Method
Md. Asiful Islam	AIUB	Web Based Platform

Table 1.1.2

1.1.2 Objectives

“CareBook” specifically will be developed for the medical professionals. Throughout this document, “CareBook” will sometimes be referred to as "this system" or "our website system". That’s all our main objective.

Besides, other objectives are:

- To build an online platform for medical professionals.
- To identify proper disease.
- To save time.
- Online medical services from the doctors.
- Decrease the communication gap between the doctors and patients.
- Developing web software.
- Don’t misuse the data of the users.
- To provide proper treatment.
- Centralized system for all related users.
- To deliver the service whatever data will be used, all that data on the website will be up to date, trusted and verified.

1.1.3 Benefits

- Saves time.
- Proper identification.
- Properly stored.
- Easy to manage.
- User friendly.
- Cost effective.
- Automated.
- Centralized system for all.

1.1.4 Software/Technology Proposed

Scripting language: Laravel, PHP, JS

Markup language: HTML, CSS

Online resources used: Bootstrap

Database: MySQL

Architecture: MVC

SDLC Model: Prototyping and Scrum

Server Software: Xampp

1.1.5 Requirements of the project

Computers

MySQL Database

Apache web server

Operating System: Windows, Linux, iOS and android

Internet Browsers

1.2 Document Conventions

Title	Font Name	Font Size
Heading1	Times New Roman	18
Heading2	Times New Roman	14
Content	Times New Roman	12

Table 1.2.1

1.3 Intended Audience and Reading Suggestions

This project is a beta version of the CareBook system. This has been implemented under the guidance of Md. Asiful Islam, Faculty, CS department, AIUB to fulfill the requirements for the Software Project-2 course, Summer 2018-19 semester. This project is useful for stakeholders of our project and as well as considering the perspective of Bangladesh its useful for medical professionals, clinics, hospitals and patients of our country.

1.4 Product Scope

CareBook is an ecosystem platform for those who are connected with the medical industry. This system will be accessible by anyone. But it will primarily be used by patients, doctors and technicians.

It's very effective to search doctors of any specific area or any specific field. CareBook provides the opportunity to find out the best doctor of any specific field or any specific area with doctors

contact and necessary details. Patients need to store their medical data for the purpose of better treatment. But in typical system they cannot properly store the data because of various reasons like missing files, damaged documents etc. Generally, patients cannot easily categorize their medical history. Categorization and organization will make it easier for doctors to gather their patient's medical history, drug history, history of the past illness, history of their operation, history of vaccination and the past test reports.

There is a major problem that has always existed between doctors and their patients. Receiving personal calls or messages from all their patients is not a practical or manageable idea for a doctor. It's not always beneficial for the patients either. In most cases, patients find it difficult to determine which doctor meets their requirements. And they also find it troublesome to find the doctor's address, appointment details and doctor's schedule. Most of the time, it is difficult for the patients to know if and where the doctor is available which can help to reduce the waiting time.

1.5 Deliverable includes in scope

- Beta version of full software
- Technical documentation
- Testing documentation

1.6 SDLC methods:

1.6.1 Phase of SDLC:

SDLC is a process for planning, creating, testing, and deploying high qualified software. SDLC stands for system develop life cycle and have six stages. The following image is a representation of the stages of a typical SDLC.

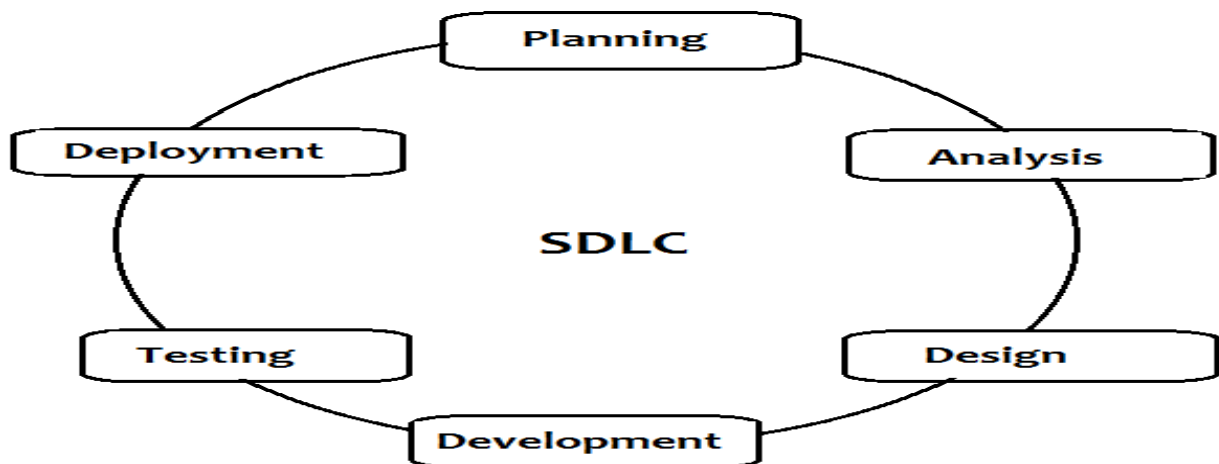


Figure 1: Phase of SDLC

Description of the phases:

Stage 1: Planning and Requirement Analysis

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the doctor, patients and industry experts. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas. Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

Stage 2: Defining Requirements analysis

Once the requirement analysis is done the next step is to clearly define and document the product requirements and approve from the patient and the doctors. This is done through an SRS (Software Requirement Specification) document which consists of all the product requirements to be designed and developed during the project life cycle.

Stage 3: Designing the Product Architecture

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification. This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints, the best design approach is selected for the product. A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third-party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

Stage 4: Building or Developing the Product

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle. Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Markup languages such as HTML and stylesheet language CSS and scripting languages JavaScript, PHP, Laravel are used for coding. The programming language is chosen with respect to the type of software being developed.

Stage 5: Testing the Product

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

Stage 6: Deployment in the Market and Maintenance

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the business strategy of that organization. The product may first be released in a limited segment and tested in the real business environment (UAT- User acceptance testing). Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

1.6.2 SDLC Models: There are different kinds of SDLC process model. Some are traditional and some agile and other.

Traditional models:

1. Waterfall model:

If all requirements are very clear then waterfall model should be chosen. The main thing is that when the working process is started in phase by phase there is no way to move back to the previous phases. This model is chosen for small projects, when the requirements are very clear and fixed and there is no option to change it. There are advantage and disadvantages in this model.

Advantages:

- Waterfall model is simple and easy to use.
- This model can be easy to implement and manage because each phase has a specific purpose.
- This model development occurs in only one phase at a time.

Disadvantages:

- Waterfall model is inappropriate for complex projects.
- It should not be used for developing object-oriented software, for long-term or ongoing projects.
- This model should not be used for projects in which requirements are unknown or subject to change.

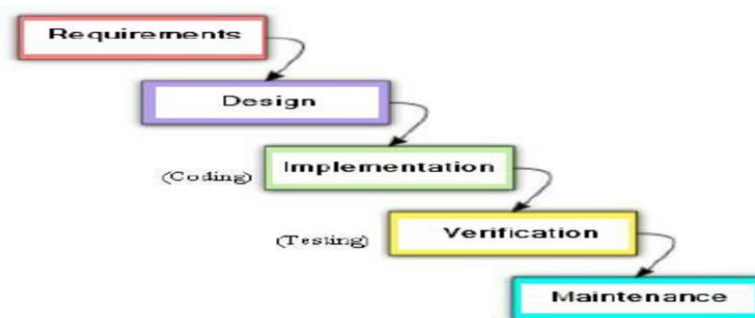


Figure 2: Waterfall Model

2. Incremental life cycle Model:

The Incremental life cycle model builds an iterative approach into the Waterfall model. Development projects are divided into several smaller iterations and more manageable iterations. Each iteration passes through a mini-Waterfall process. Requirement, design, implementation and testing phases are completed for each iteration. It starts implement a subset of the software. This method is based on repeated cycle (iterative). The main plot of this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).

Advantages:

- The Incremental model generates a working prototype early in the development process.
- The iterative nature of the Incremental model makes it more flexible when adding or changing requirements.
- This model is also easier to test and debug because testing is performed incrementally during each iteration on a relatively small amount of new code.

Disadvantages:

- The Incremental model is less risky than the Waterfall or V-Shaped models
- This model is inappropriate for large, long-term projects.

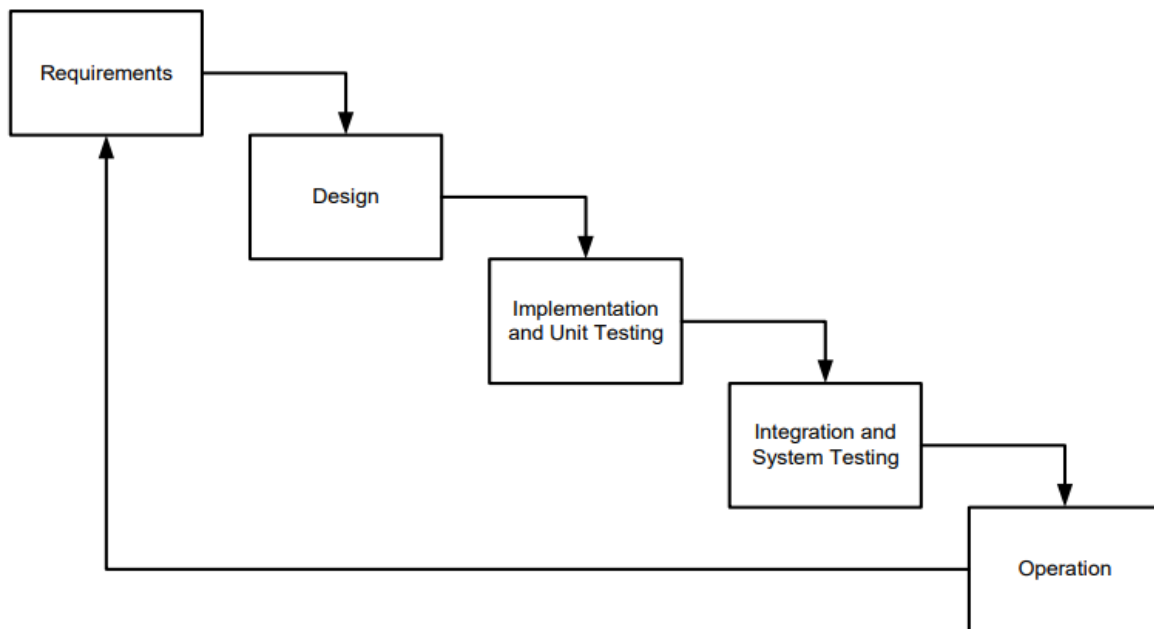


Figure 3: Incremental life cycle model

3. Prototyping model:

The prototyping refers to building software application prototypes which displays the functionality of the product under development but does not hold the exact logic of the original software. Iteration occurs as the prototype is tuned to satisfy the needs of the customer. This model is mainly used for understanding the user requirements clearly. This one helps developer to understand what functionality and system look customer is expecting to build.

Advantages:

- This is a step by step process; it makes the work of a developer more specific and clearer.
- It is safer than other models.

Disadvantages:

- It takes time to implement the project as it is based on the customer demand.

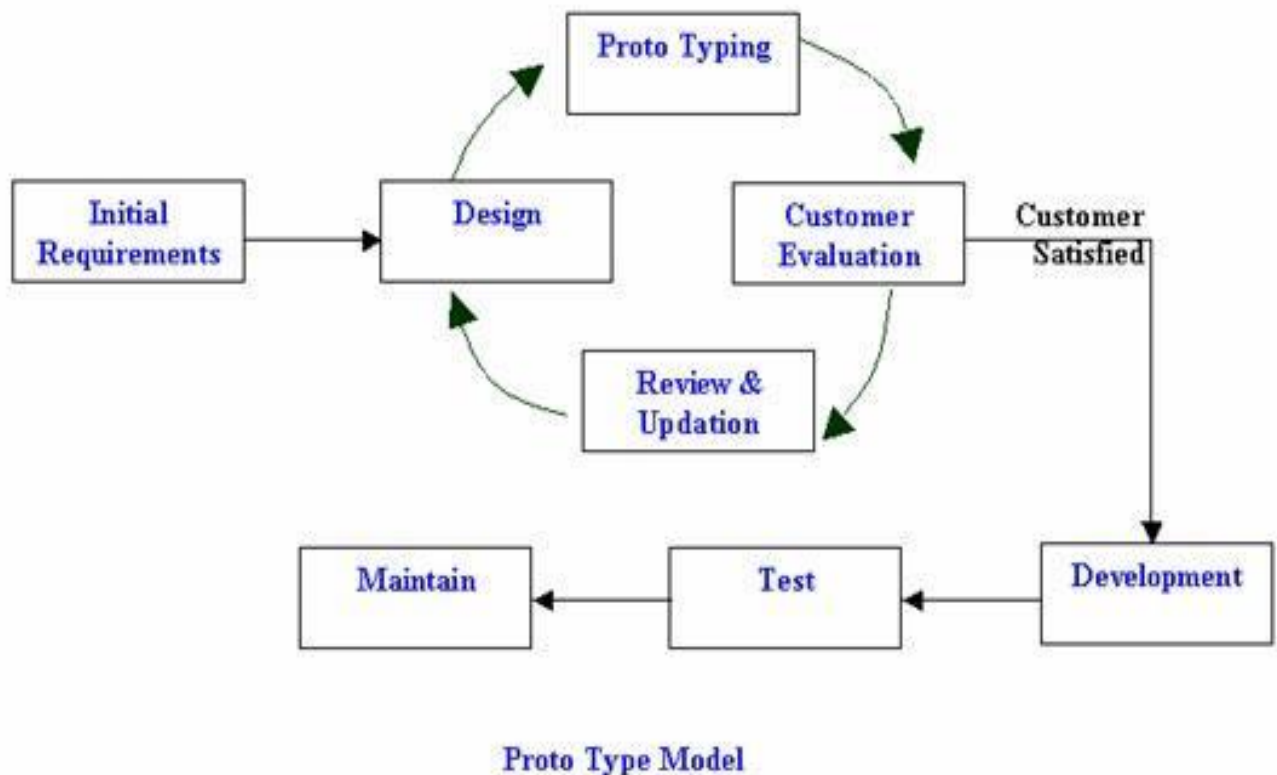


Figure 4: Prototype model

4. V – Model:

This model is also known as Verification and Validation Model. The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding

development stage. This means that for every single phase in the development cycle, there is a directly associated testing phase. Like waterfall model in this model requirements are well defined and fixed and there will be no undefined or ambiguous requirements. This is basically used for short projects.

Advantages:

- The V-Shaped model is relatively simple and easy to use.
- Specific goals are defined for each phase. The focus on preparing test plans early in the process gives the V-Shaped model a higher chance for success. Like the Waterfall model, the V-Shaped model is appropriate for small development projects in which requirements are well understood.

Disadvantages:

Like the Waterfall model, all requirements must be stated at the beginning of the project, making it difficult to add or change requirements later in the development process. All software development occurs in a single phase, so there are no early working versions or prototypes.

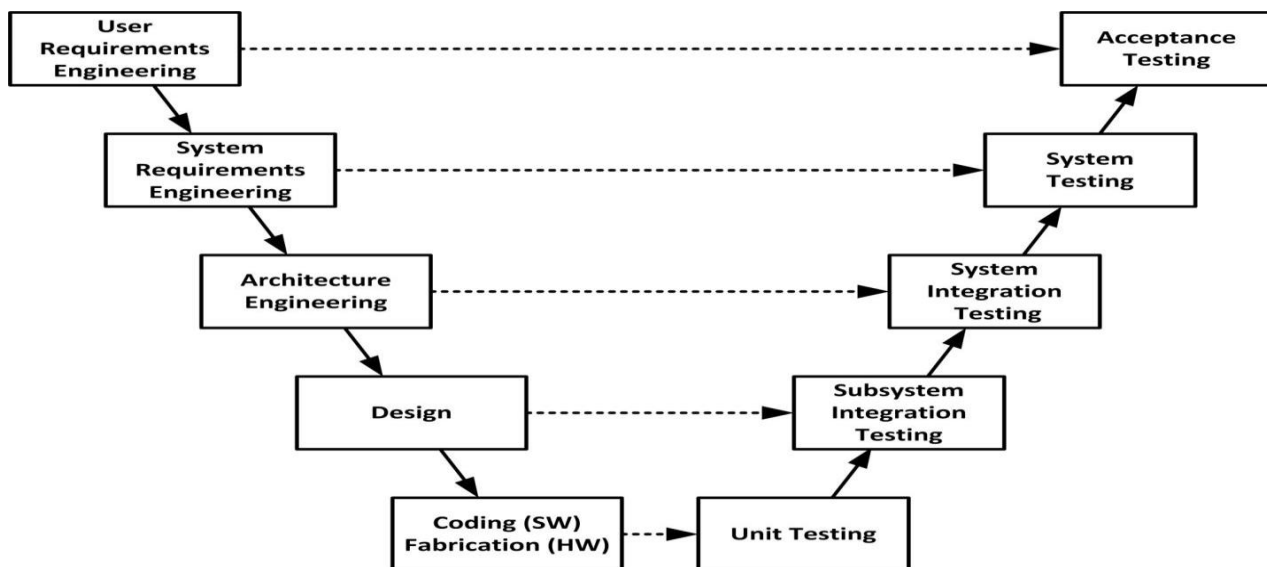


Figure 5: V model

5. Agile methods:

In Agile model, the tasks are divided to time boxes (small time frames) to deliver specific features for a release. These are done in iteration process. Each iteration process has time limit from 1 week to 4 weeks. Basically now-a-days this model is used in most of the projects. Agile model believes that every project needs to be handled differently and the existing methods need

to be tailored to best suit the project requirements. In agile the tasks are divided to time boxes small time frames to deliver specific features for a release.

Advantages:

It is a very realistic approach to software development Promotes teamwork and cross training. Functionality can be developed rapidly and demonstrated. Resource requirements are minimum and suitable for fixed requirements deliver early partial working solutions.

Disadvantages:

Not suitable for handling complex dependencies. More risk of sustainability, maintainability and extensibility. An overall plan, an agile leader and agile PM practice is a must without which it will not work. Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadline.

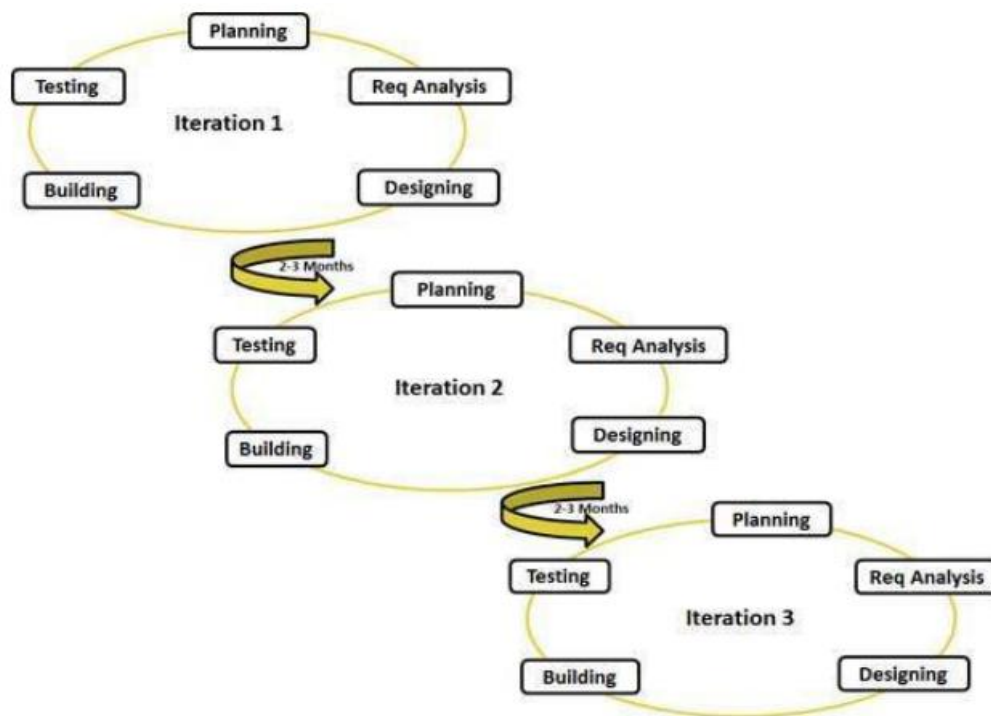


Figure 6: Agile model

1.7 Our chosen SDLC model:

Our chosen SDLC model is a combination of Prototyping model & Agile model's scrum methodology. This model is most useful in development of systems having high level of user interactions. In this model functionality can be developed rapidly and demonstrated. In this model, resource requirements are minimum and suitable for fixed requirements deliver early partial working solutions.

Why we chose Prototyping-model & Agile-model's Scrum Methodology:

There are many advantages of Prototyping-model and Agile-Model, as it is interactive to customers and developers. They are:

- This one helps developer to understand what functionality and system look customer is expecting to build.
- Iteration occurs as the prototype is tuned to satisfy the needs of the customer.
- Understanding of the system being developed.
- Reduces time and cost as the defects can be detected much earlier.
- Resource requirements are minimum and suitable for fixed requirements deliver early partial working solutions.
- Quicker user feedback is available leading to better solutions.
- Missing functionality can be identified easily.
- Every stage of the SDLC can be tested individually.

There are some lacking in using our chosen model. Like in Prototyping model, it takes time to implement the project as it is based on the customer demand. And in Agile model, Not suitable for handling complex dependencies. More risk of sustainability, maintainability and extensibility. An overall plan, an agile leader and agile PM practice is a must without which it will not work. Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadline.

1.8 Our Related work studies:

Before coming up with the idea we tried to find sites like this but we only found doctor-patients messaging platform like Klara, doctor media etc. In this part, we analysis their software and find out pros and cons of those software. That will help our project development.

2. Overall Description

2.1 Product Perspective

This platform is basically for medical professionals and the stakeholders of this system. We want to build a community depending on this platform. Patient and doctor can use this system fully free. In this beta version, only doctor, patient and technician can use this platform and admin will be responsible for maintaining the whole system.

This CareBook system can do the following things:

- **Storing patients' documents:** Patients can store all the documents in the system.
- **Searching doctors:** Patients can search doctor of any field and any specific area.
- **Categorizing history of patients:** All the history of patients will be categorized.

- **Prescription:** Doctor can also write a prescription via system.
- **Uploading test results:** Technicians can upload all the test results of the patients.
- **Task Manager:** A task manager to show daily tasks in doctors home page.
- **Recent diseases:** System will notify doctor about various new diseases of any part in the world and related necessary information's of that disease.
- **Summarized result:** From the all data system will show a summarized overview of a patient, so that doctor will be able to take quick decision from that summary.
- **Drugs information's:** System will show various drugs information's to doctors' profile.
- **Profile update:** dis All users are able to update and edit their profile information.
- **System maintenance:** dis Admins are responsible for maintaining the whole system, checking every user's entity, update, delete etc.

2.2 Product Features

The major features of CareBook system as shown in below-

Patients:

- **Database for storing patients' documents:** In this system, we will use database for storing patient data. Patients can store their medical data there. So, the patient does not have to carry a hard copy of each of the documents, there is no chance to lose the documents. Patients can download or view their data from the database anytime and anywhere.
- **Automatic categorization of the data:** When patients input their data, it will be automatically saved in a certain category based on the data type stored in the database. So that patients don't need to sort their data manually.
- **Searching doctor's information:** Patients can search for a doctor by name or ID and the system will show the relative information's about the specific doctor. Patients can also learn whether the doctor is available using the doctor's current status feature.
- **Previous consultation:** Patients are able to check previous consultation with any doctor and also provided instructions and prescriptions of that doctor.

Doctors:

- **Patient's history:** Doctors can find all the history of their patients by providing the patient's id and name in the system. All the history's will be categorized. A doctor can see what others doctor suggested to the patients like the previous drugs or previous treatments. And also, can check all the treatments or drugs that patients have taken.
- **Doctors information's:** All necessary information's of a doctor will be stored in database to show and use in need.
- **Disease database:** Top emerging diseases currently effected in various parts of the world will be showed with all necessary information's of that disease like origin, symptom's,

treatments, medicine's etc. will be showed in doctors homepage so that doctors will be able to keep themselves updated about various diseases and will be able to provide better treatment without delaying time. At the same time doctors also capable to save any disease information's if he/ she wish to save.

- **Prescription:** Doctors can also write a prescription via system. Prescription is system generated and it will be stored in patients' profile and can be downloaded in pdf format.
- **Checking the history of their own patients:** Doctors can check his patients' details and their treatment history that he suggested.

Technician:

- **Uploading all the test reports:** Technician can upload all the test reports. It will be uploaded to specific patients' profile so that doctors will be able to check all the test reports without any hassle.

Receptionist:

- **Serial number:** Receptionist issues serial number for patients.
- **Total payment:** Receptionist will give update about the total payment details that a patient needs to pay.

Admin:

- **Approval of new accounts:** Admin can approve all the new accounts of patients/doctors/receptionist and technicians. After proper verification users will get access of the system. If admin wants, they can block any specific user from the system.
- **Trending diseases:** Admin can see all the diseases and can figure out which is the most trending one on that mean time.

2.3 User Classes and Characteristics

- Patient
- Doctor
- Technician
- Admin

2.4 Operating Environment

The software will run in a hosting service and this domain can be accessed via any windows or android platform through any browser like Google Chrome, Mozilla, Microsoft Edge etc.

2.5 Design and Implementation Constraints

- SQL commands for above queries/applications.
- Implement the database at least using a centralized database management system.

- “CareBook” can be run in any processing system like dual-core, quad-core, hexa-core, octa-core etc.
- Much higher configured system and well-developed operating system will help to run this system more smoothly.

2.6 User Documentation

- User manual.
- Online help center.
- Help forum.

2.7 Assumptions and Dependencies

- A good structured and secured hosting service to store the data & files.
- Good internet connection.

3 External Interface Requirements

3.1 User Interfaces

Scripting language: Laravel, PHP, JS

Markup language: HTML, CSS

Online resources used: Bootstrap

Database: MySQL

Architecture: MVC

Server Software: Xampp

3.2 User Interface Design

Homepage & Login:

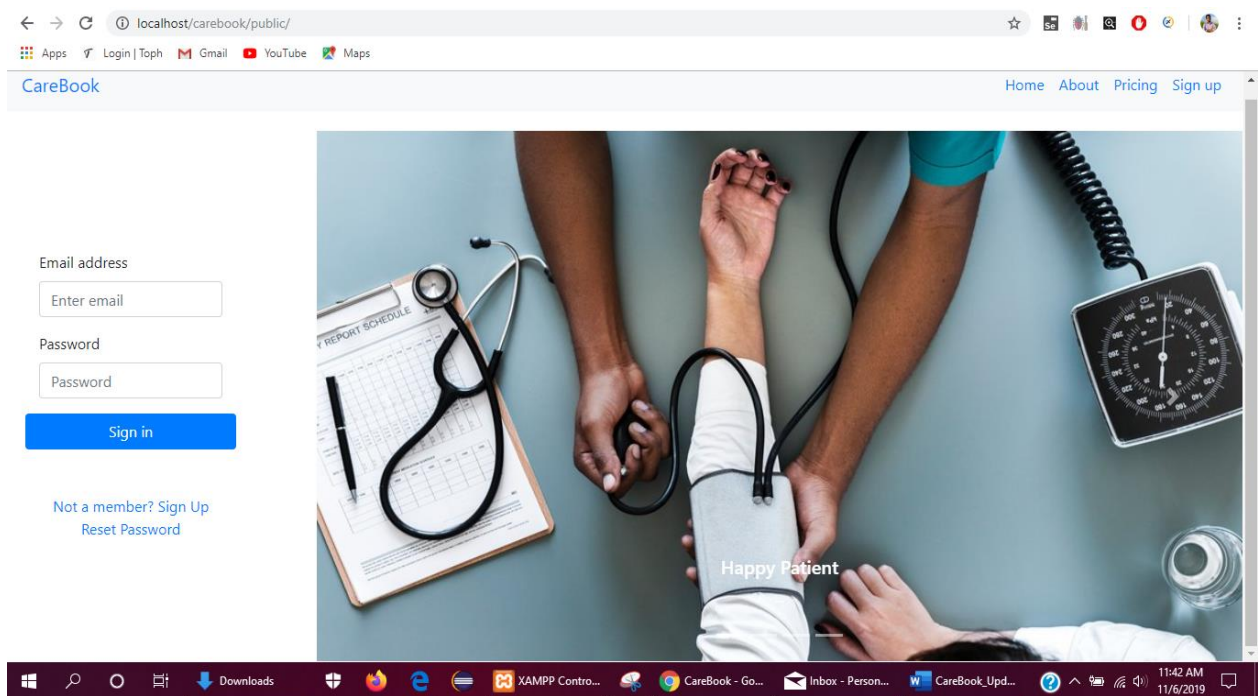


Figure 07: Homepage & login-1

Homepage & Login:

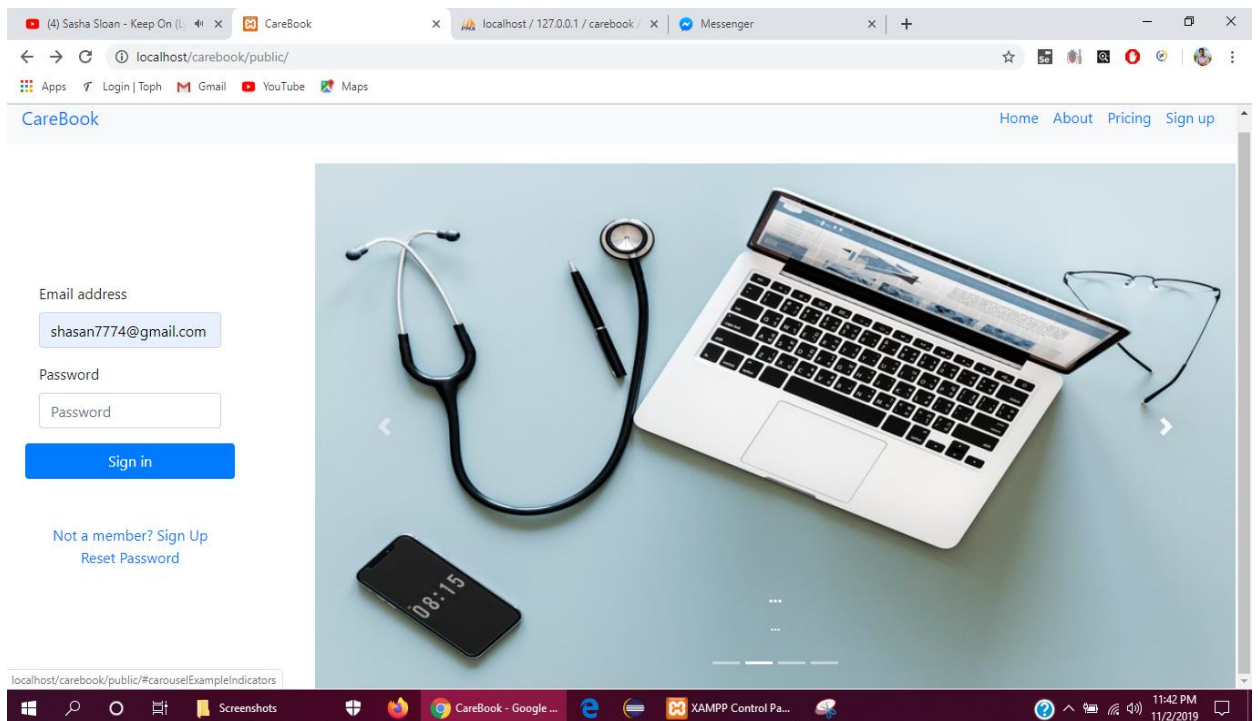


Figure 08: Homepage & login-2

Sign-Up:

REGISTRATION

First Name

Last Name

Email*

Password*

Confirm Password*

Date of Birth*

Phone number

Your phone number won't be disclosed anywhere

Address

Gender
☐ Female ☐ Male

Sign Up As*

*Required fields

[Next](#)

page 1 of 2

Figure 09: Common Sign Up page

Sign-In Options:

CareBook

Home About Pricing Sign up

☐ Login As Patient

☐ Login As Doctor

[Login](#)

Figure 10: Sign-in options

Patient Homepage:

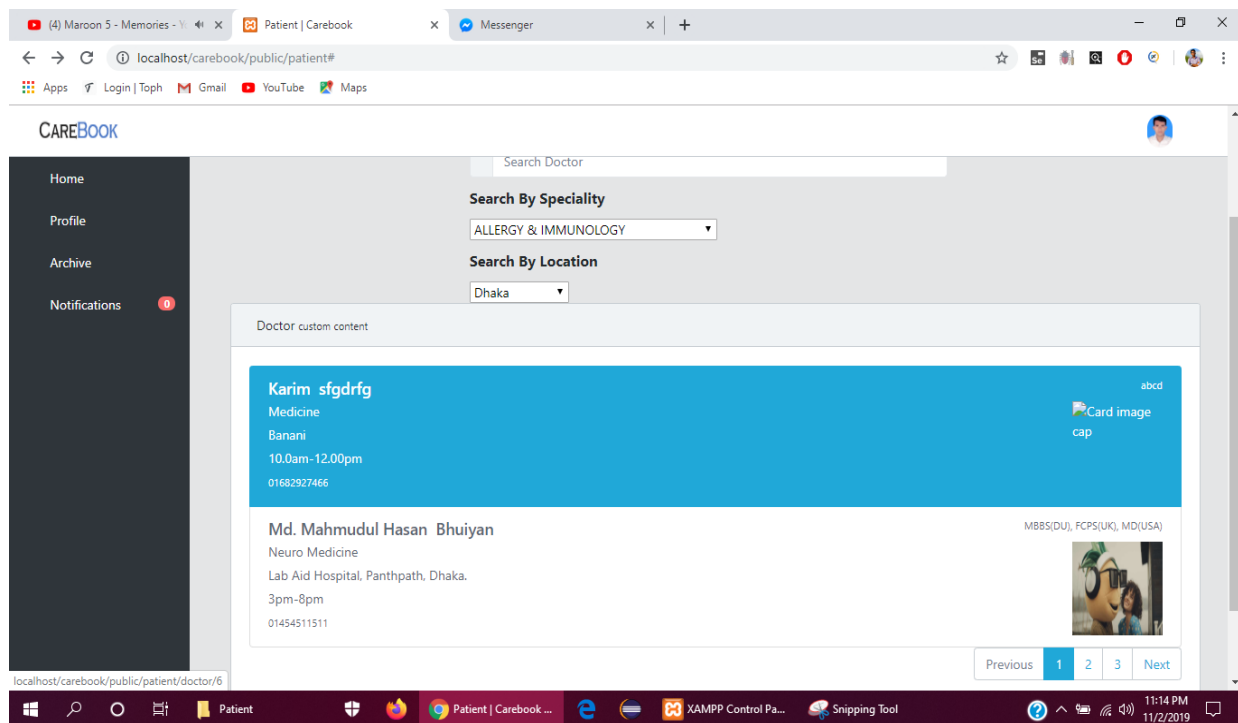


Figure 11: Patient's homepage

Patient Notification:

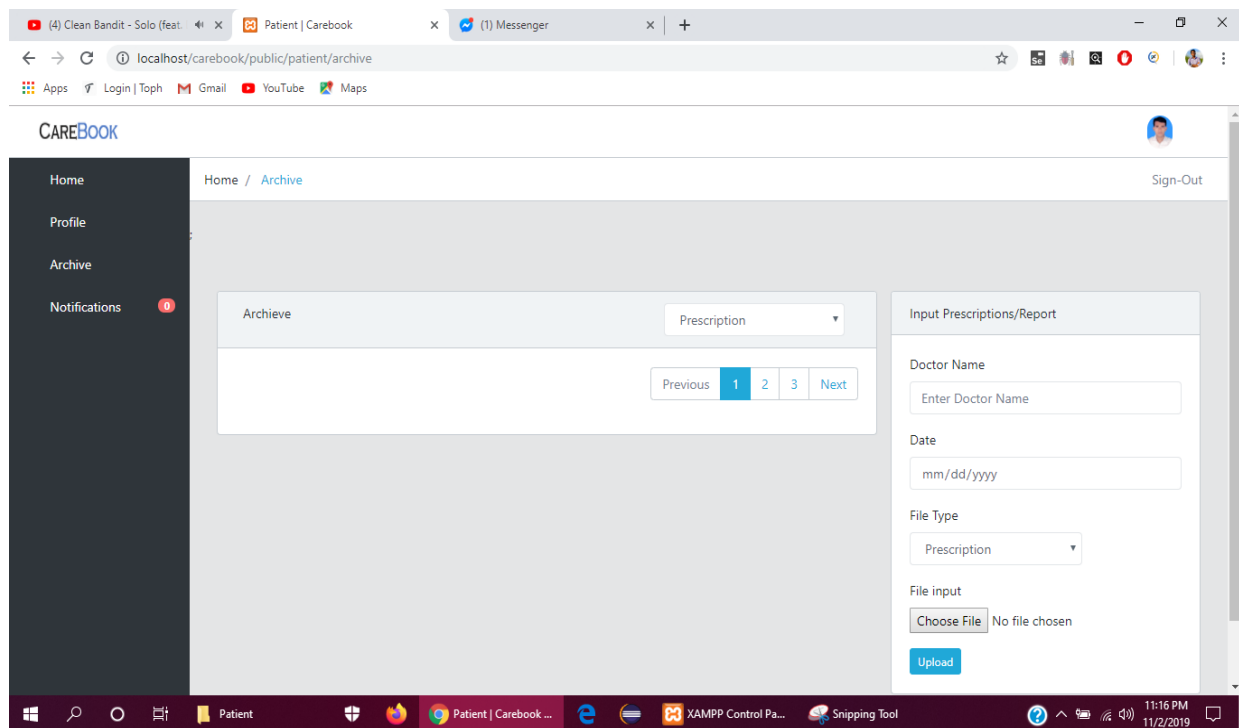
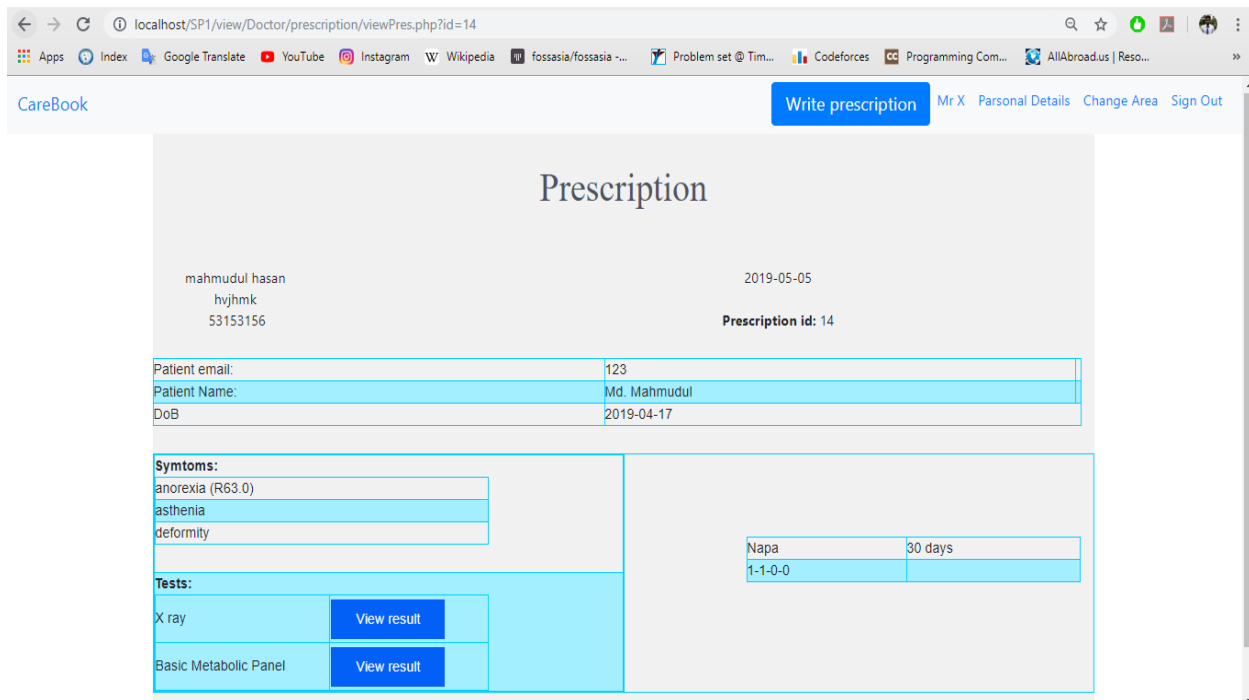


Figure 12: Patient Notification

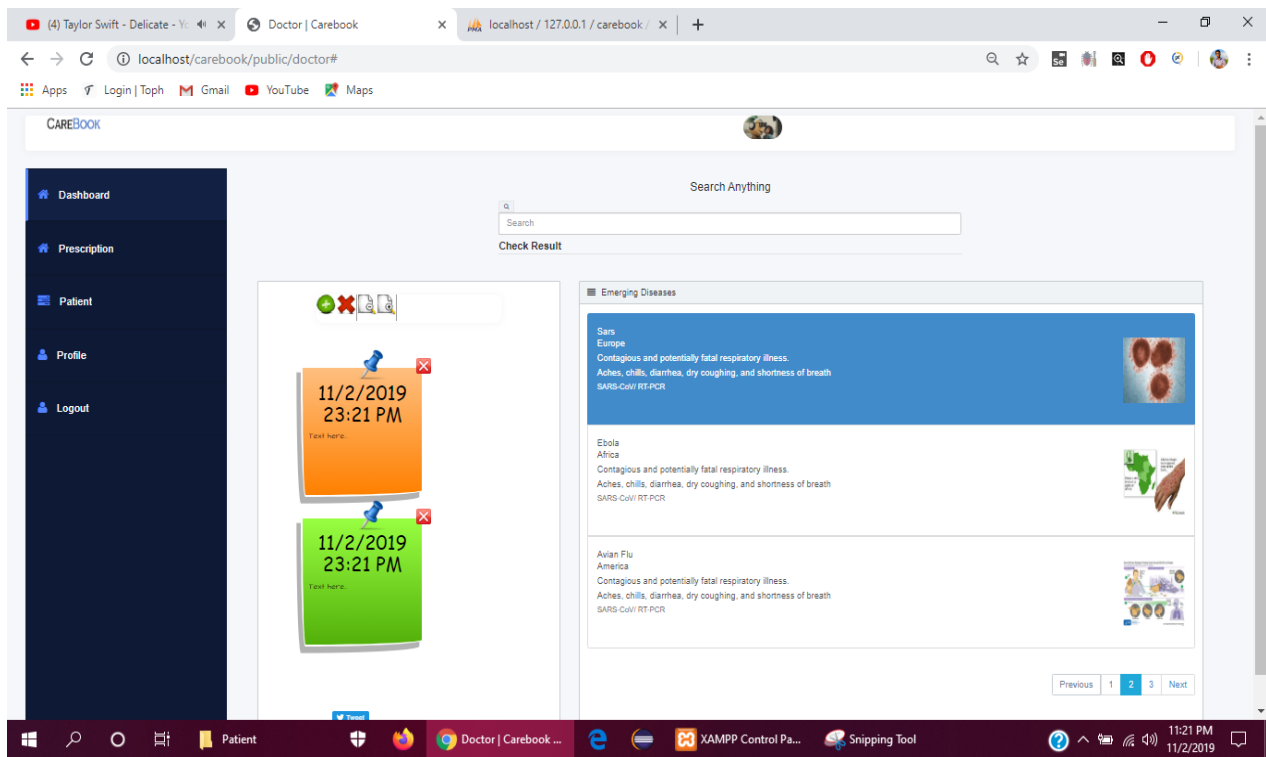
Prescription:



The screenshot shows a web browser at the URL `localhost/SP1/view/Doctor/prescription/viewPres.php?id=14`. The page is titled "Prescription" and displays patient information for "mahmudul hasan" (ID: 53153156) on "2019-05-05". The "Prescription id" is 14. The form includes fields for "Patient email" (123), "Patient Name" (Md. Mahmudul), and "DoB" (2019-04-17). Under "Symptoms", there are checkboxes for "anorexia (R63.0)", "asthenia", and "deformity". Under "Tests", there are checkboxes for "X ray" and "Basic Metabolic Panel", each with a "View result" button. On the right, there is a section for "Napa" with a value of "30 days" and a "1-1-0-0" indicator.

Figure 13: Prescription

Doctor Homepage:



The screenshot shows the "Doctor | Carebook" homepage. The left sidebar contains navigation links: "Dashboard", "Prescription", "Patient", "Profile", and "Logout". The main content area has a "Search Anything" bar and a "Check Result" button. Below this, there are two sticky notes: an orange one dated "11/2/2019 23:21 PM" and a green one dated "11/2/2019 23:21 PM". On the right, there is a section titled "Emerging Diseases" with three entries: "Sars" (Europe), "Ebola" (Africa), and "Avian Flu" (America). Each entry includes a brief description and a "SARS-CoV-2 RT-PCR" test result. The bottom of the page shows a Windows taskbar with various open applications, including "Doctor | Carebook ...", "XAMPP Control Pa...", and "Snipping Tool". The system clock indicates "11:21 PM 11/2/2019".

Figure 14: Doctor's homepage

Prescription Form:

The screenshot shows the 'Doctor | Carebook' interface in a web browser. The URL is `localhost/carebook/public/doctor/prescription`. The left sidebar contains links for Dashboard, Prescription, Patient, Profile, and Logout. The main content area displays patient information: Karim, sfgdrfg, Medicine, abcd, Banani. The date and time are Sat Nov 02 2019 23:22:39 GMT+0600 (Bangladesh Standard Time). The Prescription id is displayed. Below this, there are tabs for Dashboard, Symptoms, Tests, Medicine, and Prescription. A search bar is present with fields for Patient email, Patient Name, and Age, and a Search button. Navigation buttons Previous and Next are also visible.

Figure 15: Doctor's prescription form

Prescription PDF Format:

The screenshot shows the 'CareBook Prescription' PDF format. The header includes the CareBook logo and the text 'CareBook professional team is always here to help you We care about you'. The main content area displays patient information: Name, ID No, Date/Time: Sat Nov 02 2019 23:22:39 Gi, Age/Sex, and Consultant: Karim. Below this, there are sections for Symptoms, Clinical diagnosis, Treatment, Test, and Visit after. The footer contains Save and Discard buttons.

Figure 16: Prescription's pdf format

Previous Patient:

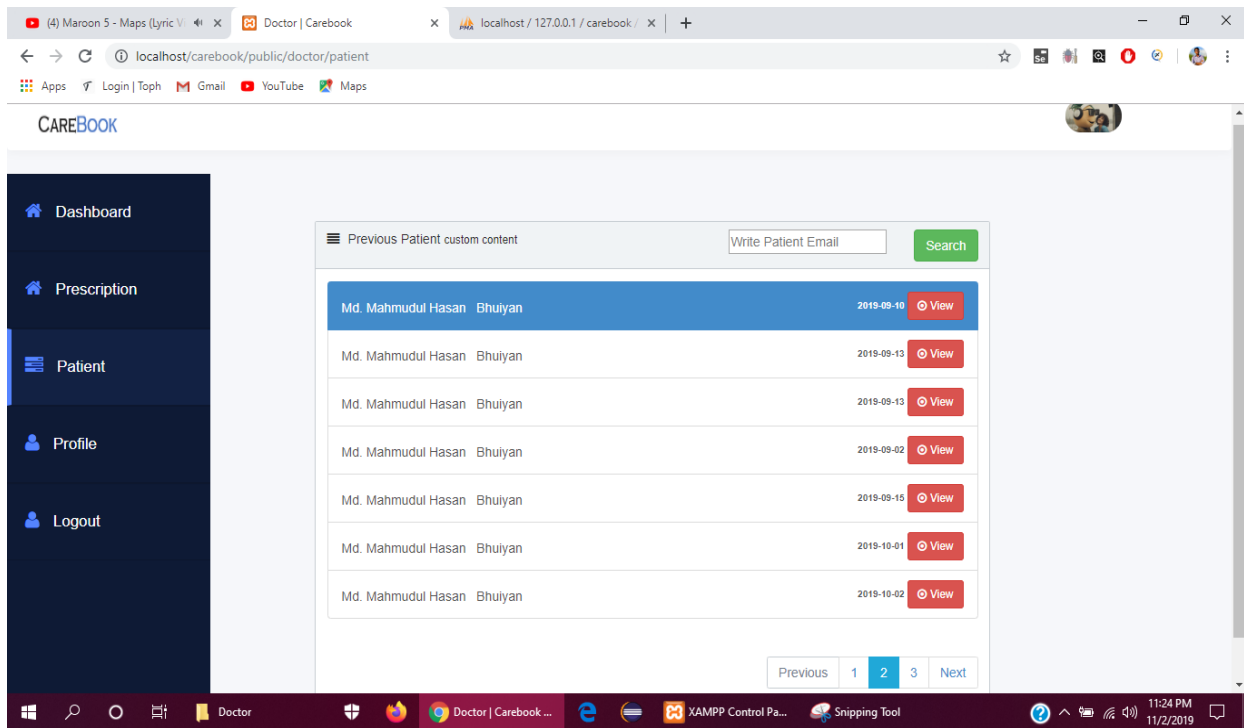


Figure 17: Previous patient checklist

Technician Homepage:

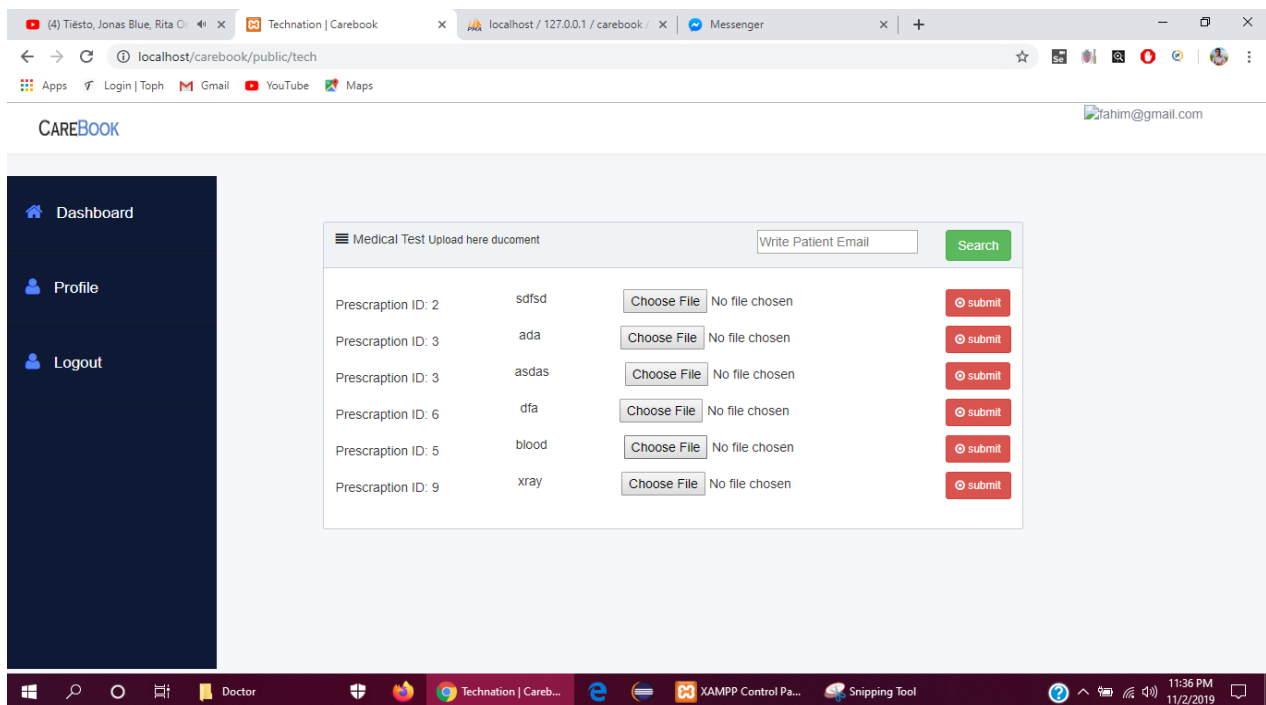


Figure 18: Technician's homepage

Technician Profile:

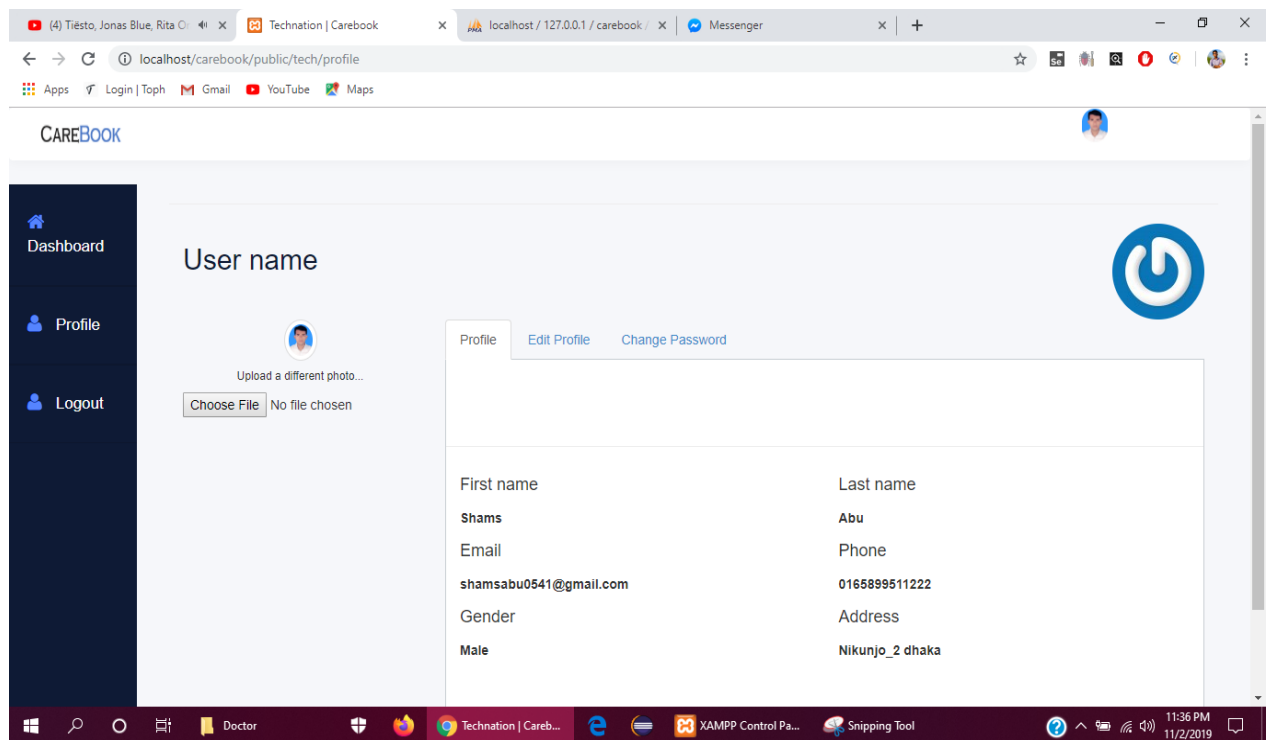


Figure 19: Technician's profile

Admin Homepage:

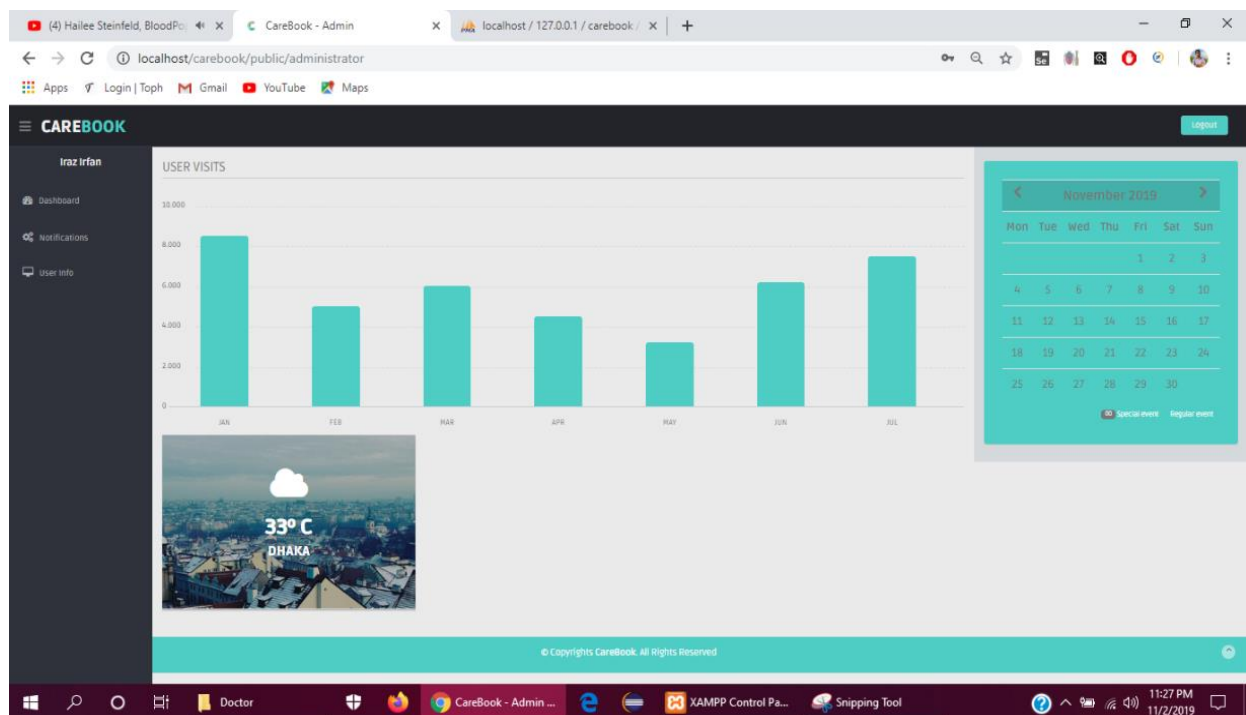


Figure 20: Admin's homepage

Doctor Details:

Email	BMDC No	Degree	Specialized	Consulting	Location	Contact
shaheb@mail.com	51464186	MBBS(DU), FCPS(UK), MD(USA)	Neuro Medicine	3pm-8pm	Lab Aid Hospital, Panthpath, Dhaka.	01454511511
karim1@gmail.com	561256156	abcd	Medicine	10.0am-12.00pm	Banani	01682927466

Copyrights CareBook. All Rights Reserved

Figure 21: Doctor details

Patient Details:

Email	Phone	First Name	Last Name	Address	NID
123@gmail.com	fdvgdf	efwrfw	sfgdrfg	gergv	48949
abdul@gmail.com	01682927466	Karim	Abdul	Abdur Sadek Road	561561
azij@gmail.com	01682927466	Karim	Azij	Abdur Sadek Road	6325156
bhuiyanmahmudul@gmail.com	sdfsd	sdfsdcfs	sdvsd	sdfsd	156165
hbjh@gmail.com	asdfc	dfgdrf	sfdgsd	ascscdc	1561858
hbuhbu1@gmail.com	01682927466	sfs	sfgdrfg	Abdur Sadek Road	015415646
karim1@gmail.com	01682927466	Karim	sfgdrfg	Abdur Sadek Road	5616516
karim2@gmail.com	01682927466	dasdasd	Azij	Abdur Sadek Road	56156
karim@gmail.com	01682927466	Abdul	Karim	abdur Sadek Road	05166262
mahmudulhasan661@gmail.com	01682927466	Md. Mahmudul Hasan	Bhuiya	Abdur Sadek Road	48949
sdfvsdfv1@gmail.com	sdfs	cfc	cfcdrf	sdrsd	41861896

Copyrights CareBook. All Rights Reserved

Figure 22: Patient details

3.3 Minimum Hardware Requirement

Server Side:

OS	Linux/Windows Server
CPU	Minimum Intel Xenon or higher
RAM	2 GB or higher
Hard Drive	1024 GB or more
Client Side	Operating System: any operating system with webpage browsing accessibilities
CPU (PC)	Minimum Intel Pentium or higher
Operating System (Android)	any operating system with web page browsing accessibilities
CPU (Android)	Mali-400MP4 or higher
RAM	512 mb or higher

Table 3.4.1

4 System Feature

4.0 Overall Description

4.0.1 User Option

There are multiple login systems in the software. Every user has a particular homepage. To access the system the need to login through id and password. If same user has multiple role in the system by using same credential, he/ she can access both profile by only identifying the role.

4.0.2 Stimulus/Response Sequence

User can login through the id and password or log out. For login, when user give id and password, the software will verify them. If it is allowed it will give access to the user.

4.0.3 Interface Requirement

Graphical user interface will demonstrate the page that will appear to user. The front-end part contains lots of form that will be used by the users.

4.0.4 User Interface

User interface will contain lots of features that used by users. They will appear step by step while accessing the system. There must be some rules that should follow the user.

The major features are already written in section 2.2. Here are some use cases of the system.

4.1 System Environment

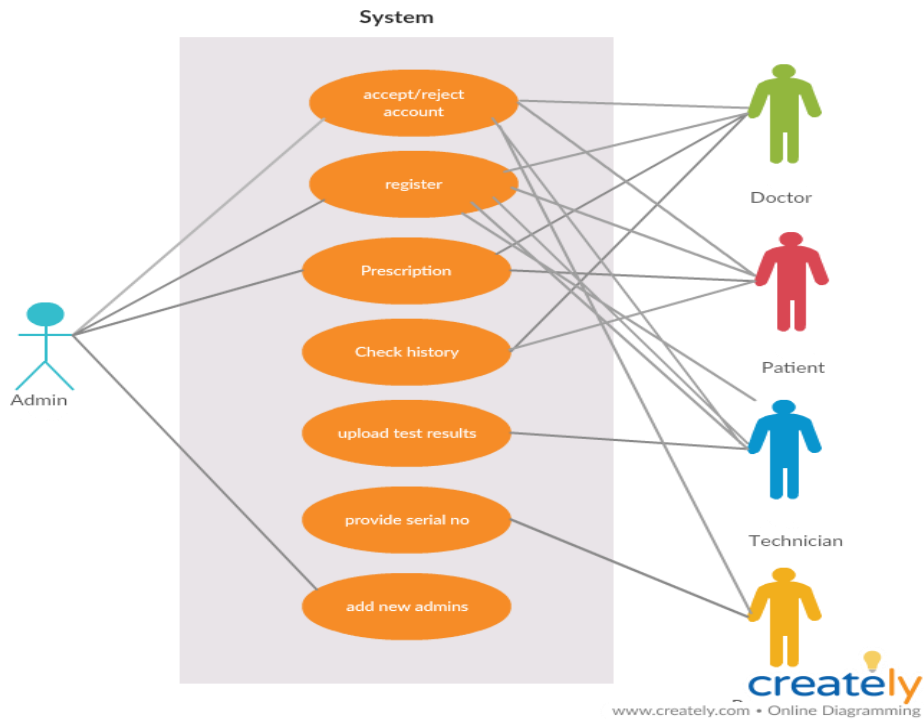


Figure 23: Full system Use case diagram'

4.2 Functional requirements Specifications

4.2.1 Doctor Use Case:

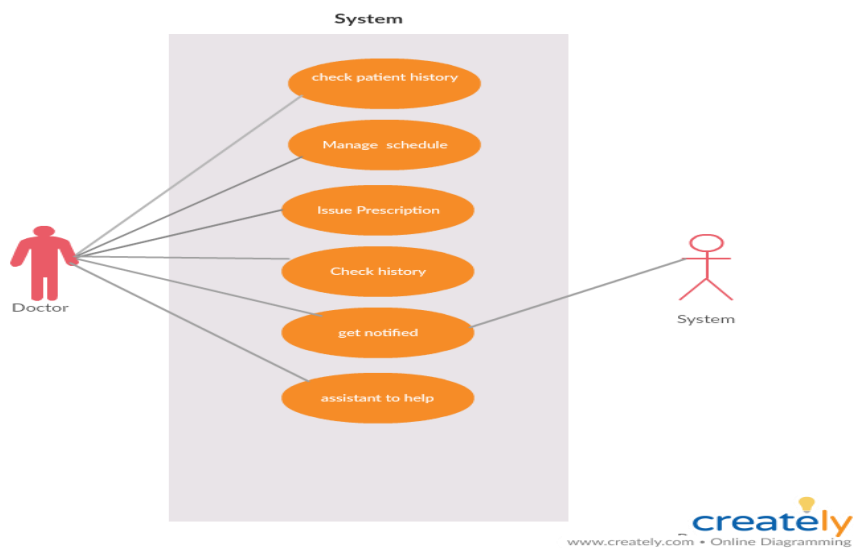


Figure 24: Doctor Use case Diagram

Initial Step-By-Step Description:

1. Doctor can check patient's history.
2. Doctor can manage his/her schedule.
3. Doctor can issue a prescription.
4. Doctor can check recent emerging disease.
5. Doctor get notified from system if emergency patients come by.
6. Doctor get an assistant for help.

4.2.2 Login & Register Use Case:

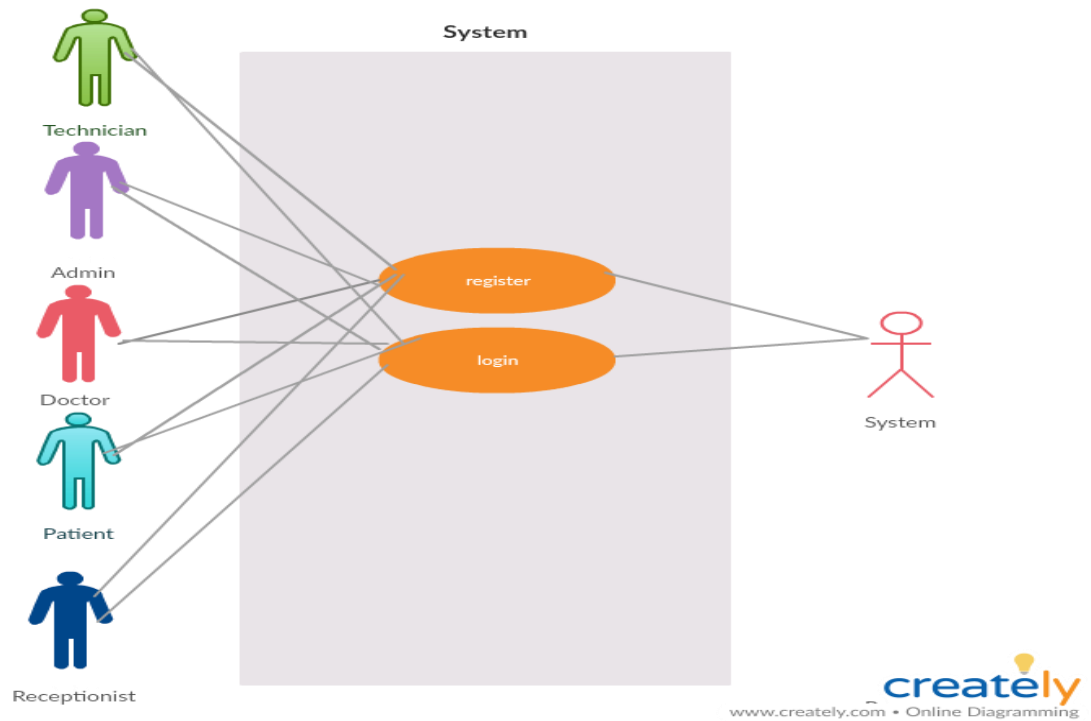


Figure 25: Login & Register Use case Diagram

Initial Step-By-Step Description:

1. Technician can register & login into the system.
2. Admin can register & login into the system.
3. Receptionist can register & login into the system.
4. Patient can register & login into the system.
5. Doctor can register & login into the system.

4.2.3 Patient Use Case:

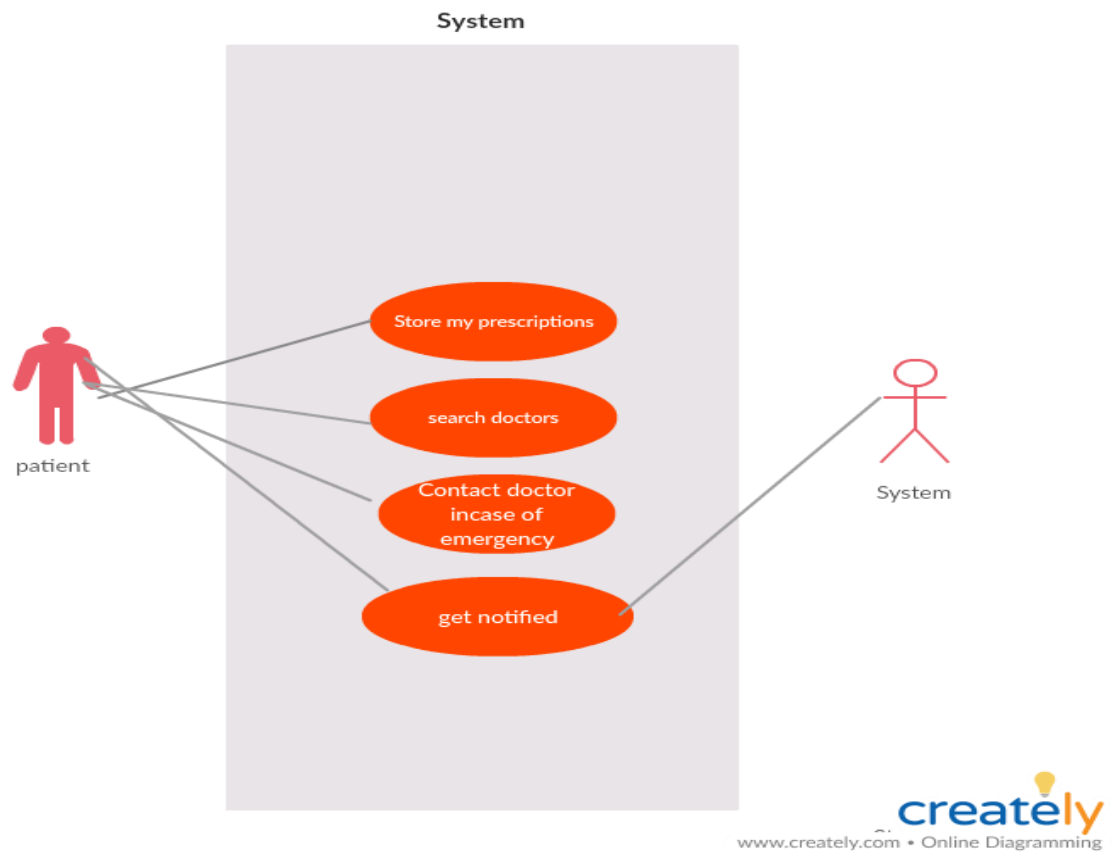


Figure 26: Patient Use case Diagram

Initial Step-By-Step Description:

1. Patient can store their prescription in the system.
2. Patient can search doctors.
3. Patient can contact doctor in case of emergency.
4. Patient get notified if doctor give prescription.

4.2.4 Technician Use Case:

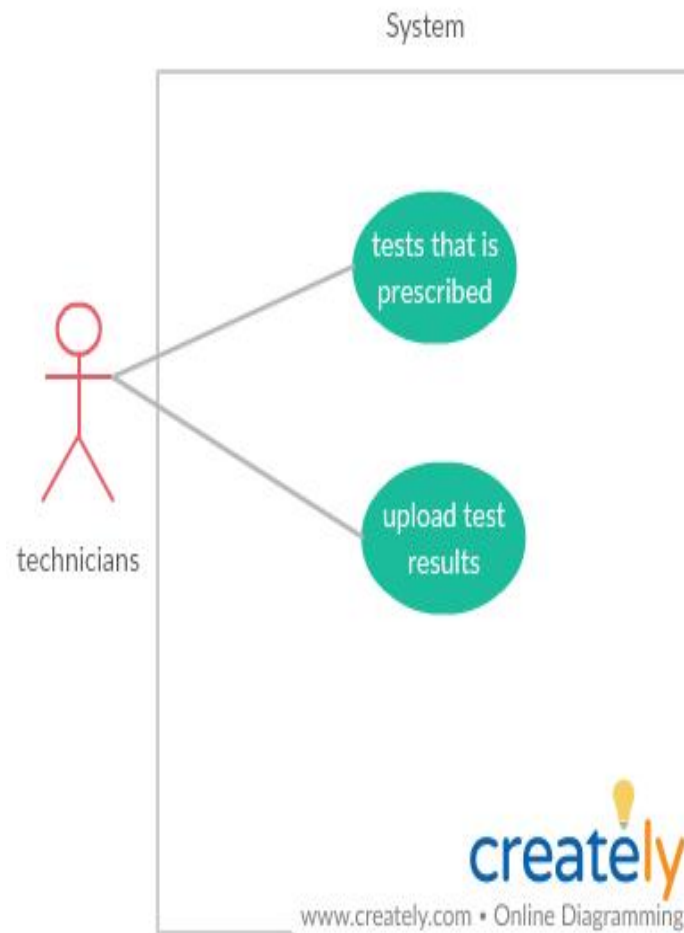


Figure 27: Technician Use case Diagram

Initial Step-By-Step Description:

1. Technician upload test results in the system.
2. Technician tests that is prescribed.

4.2.5 Admin Use Case:

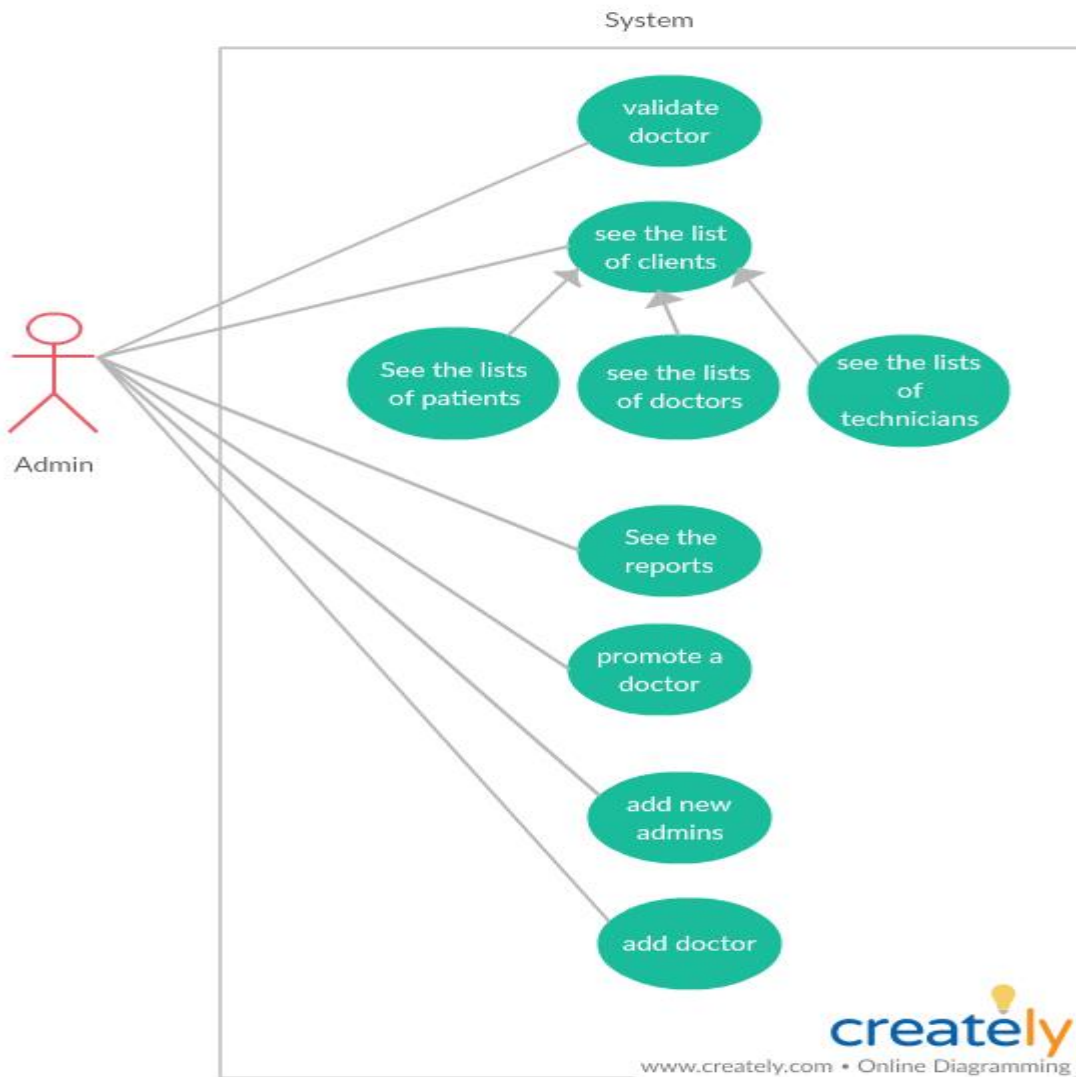


Figure 28: Admin Use case Diagram

Initial Step-By-Step Description:

1. Admin can add new doctors.
2. Admin can add new admins.
3. Admin can see the reports.
4. Admin can see the list of patients.
5. Admin can see the list of doctors.
6. Admin can see the list of technicians.

4.2.6 Receptionist Use Case:

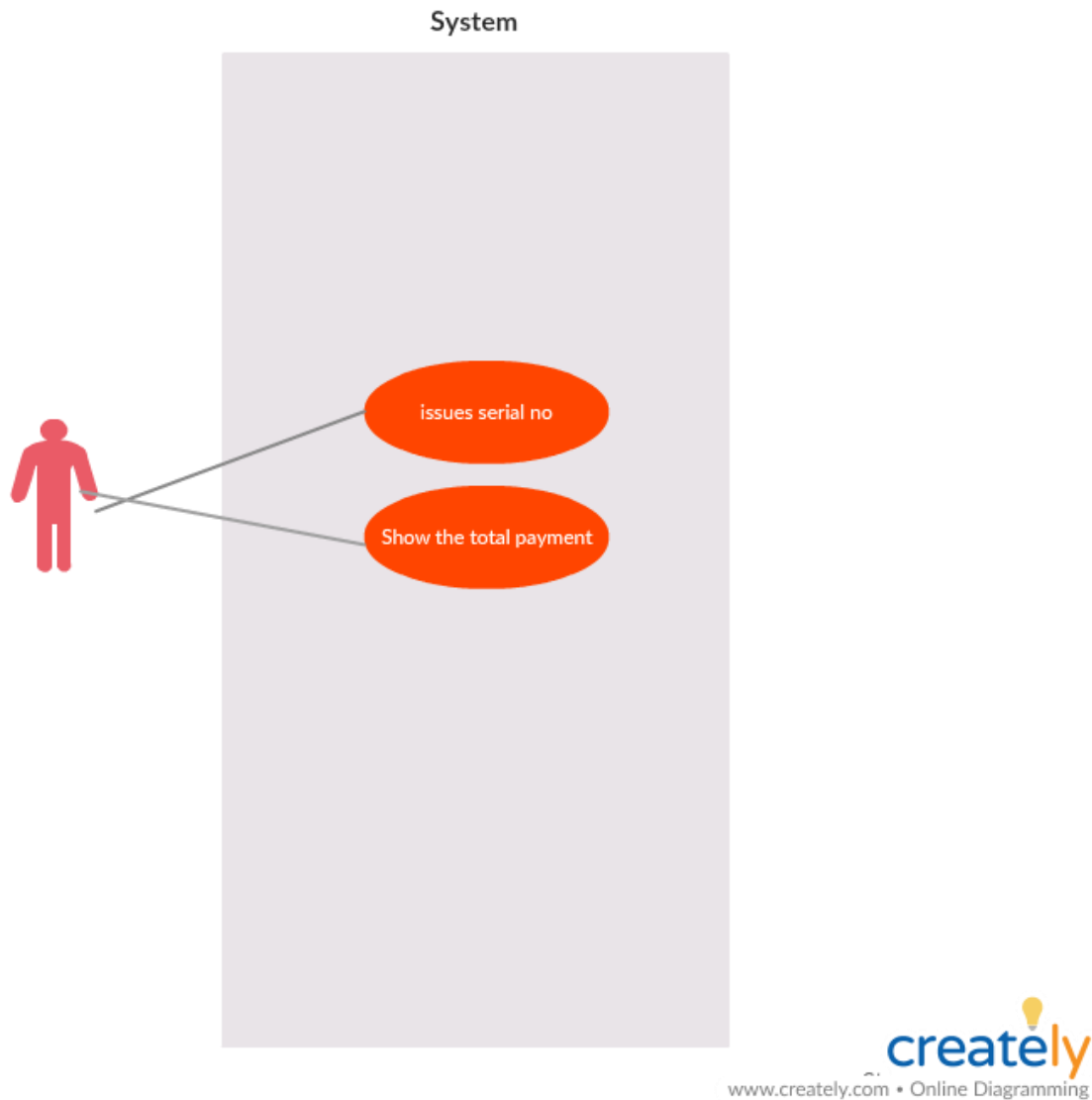


Figure 29: Receptionist Use case Diagram

Initial Step-By-Step Description:

1. Receptionist can issue serial number for patients.
2. Receptionist can show the total payment.

4.3 E-R Diagram

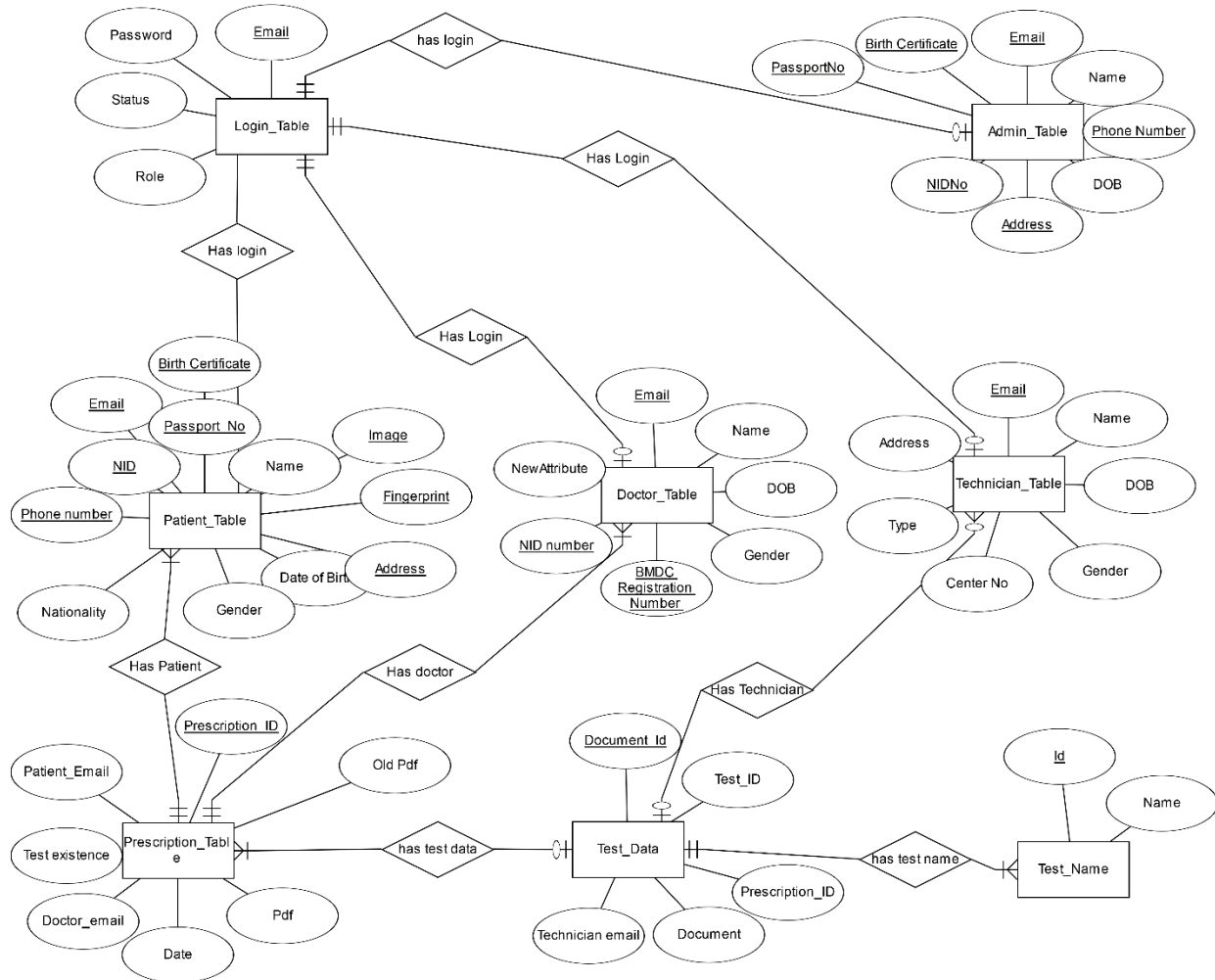
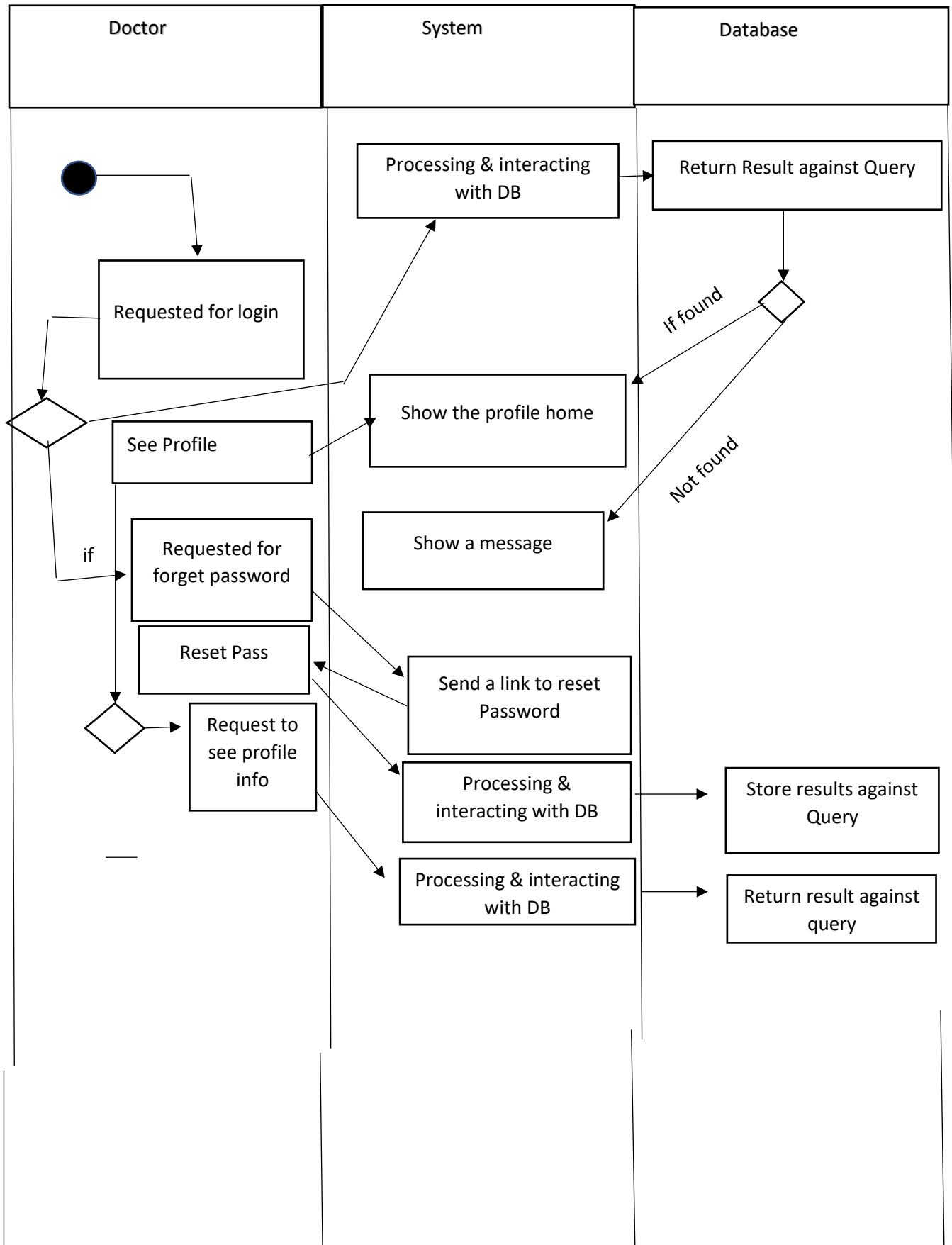


Figure 30: E-R Diagram of the system

4.4 Activity Diagram

Doctor:



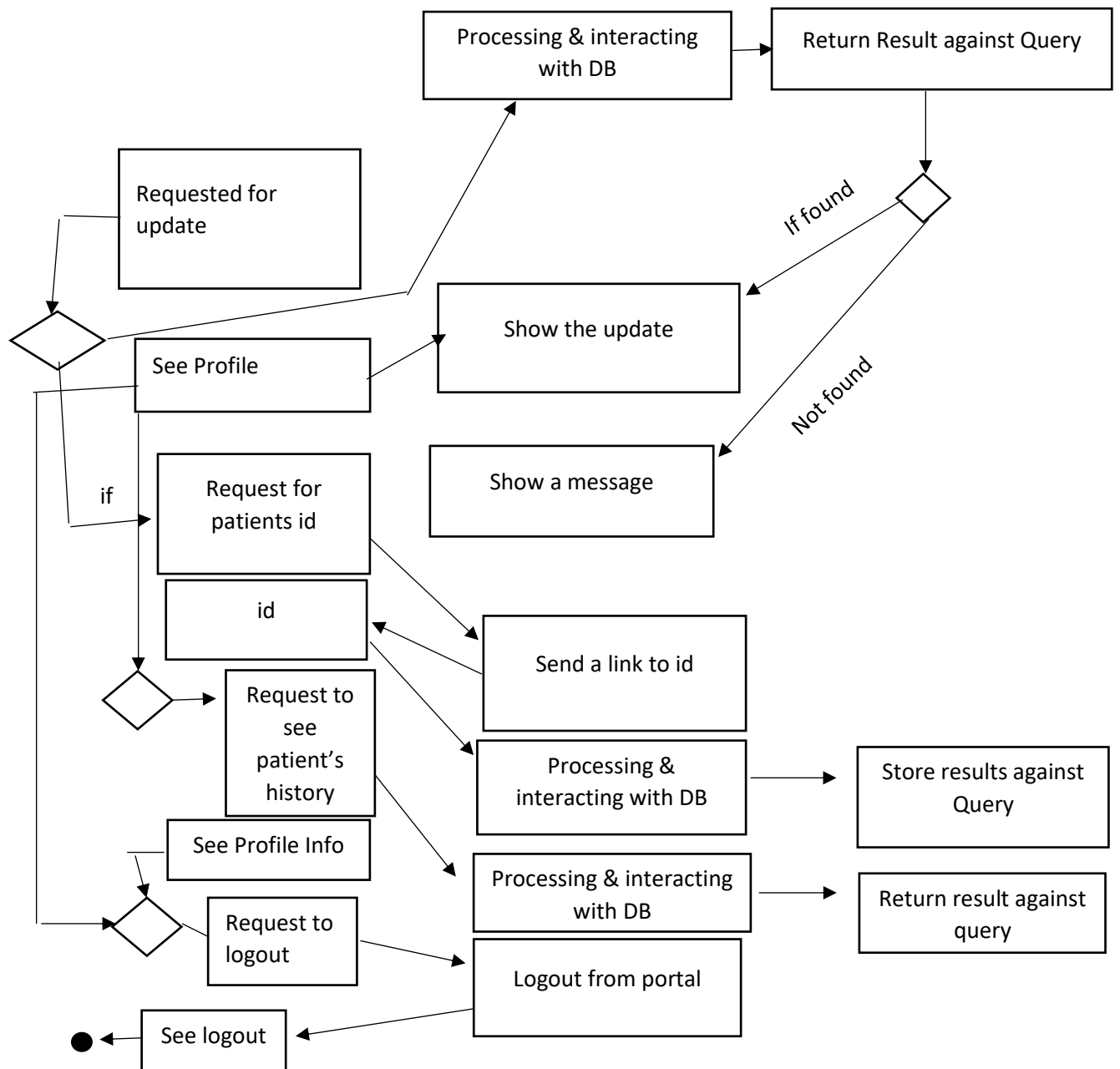
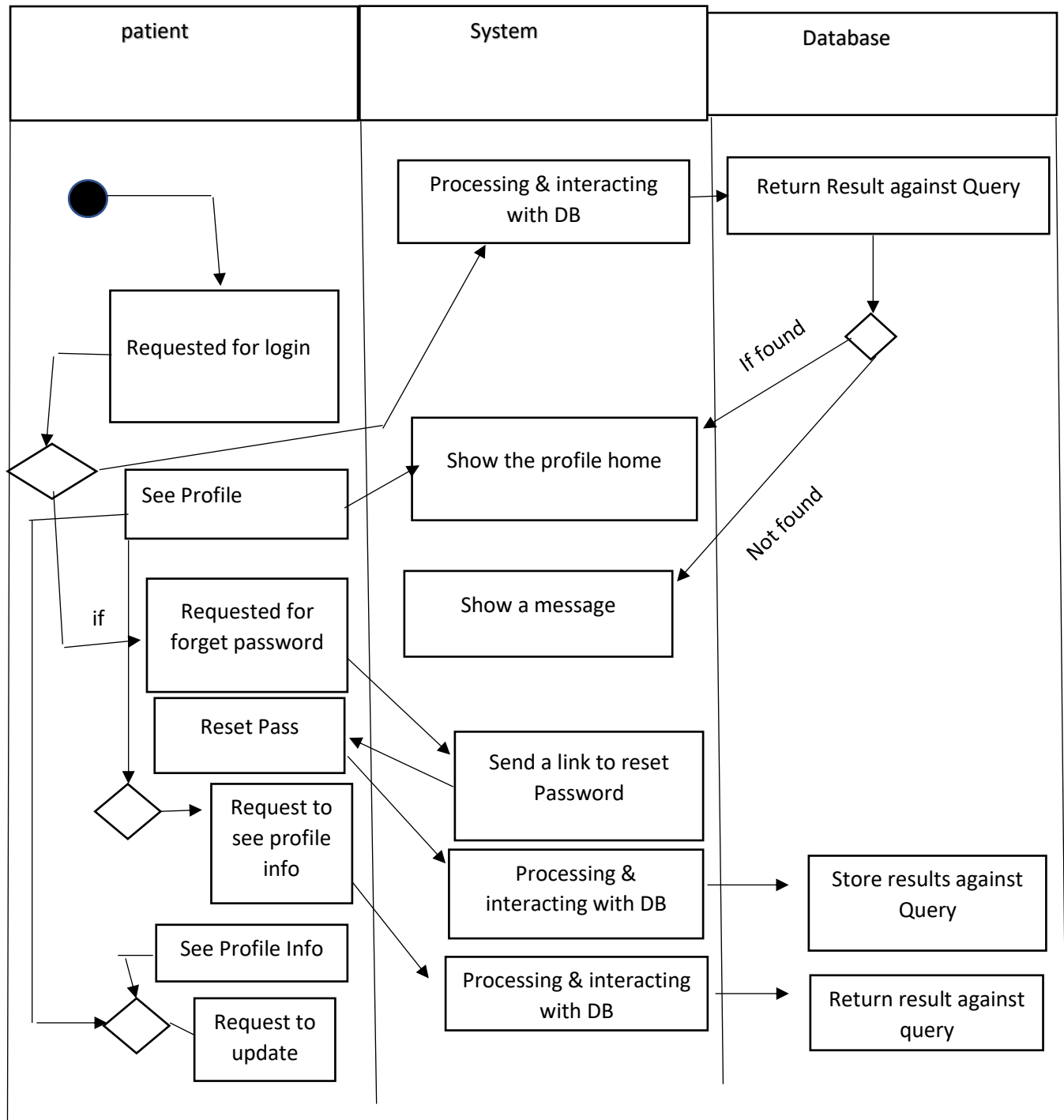


Figure 31: Doctor Activity Diagram of the system

Patient:



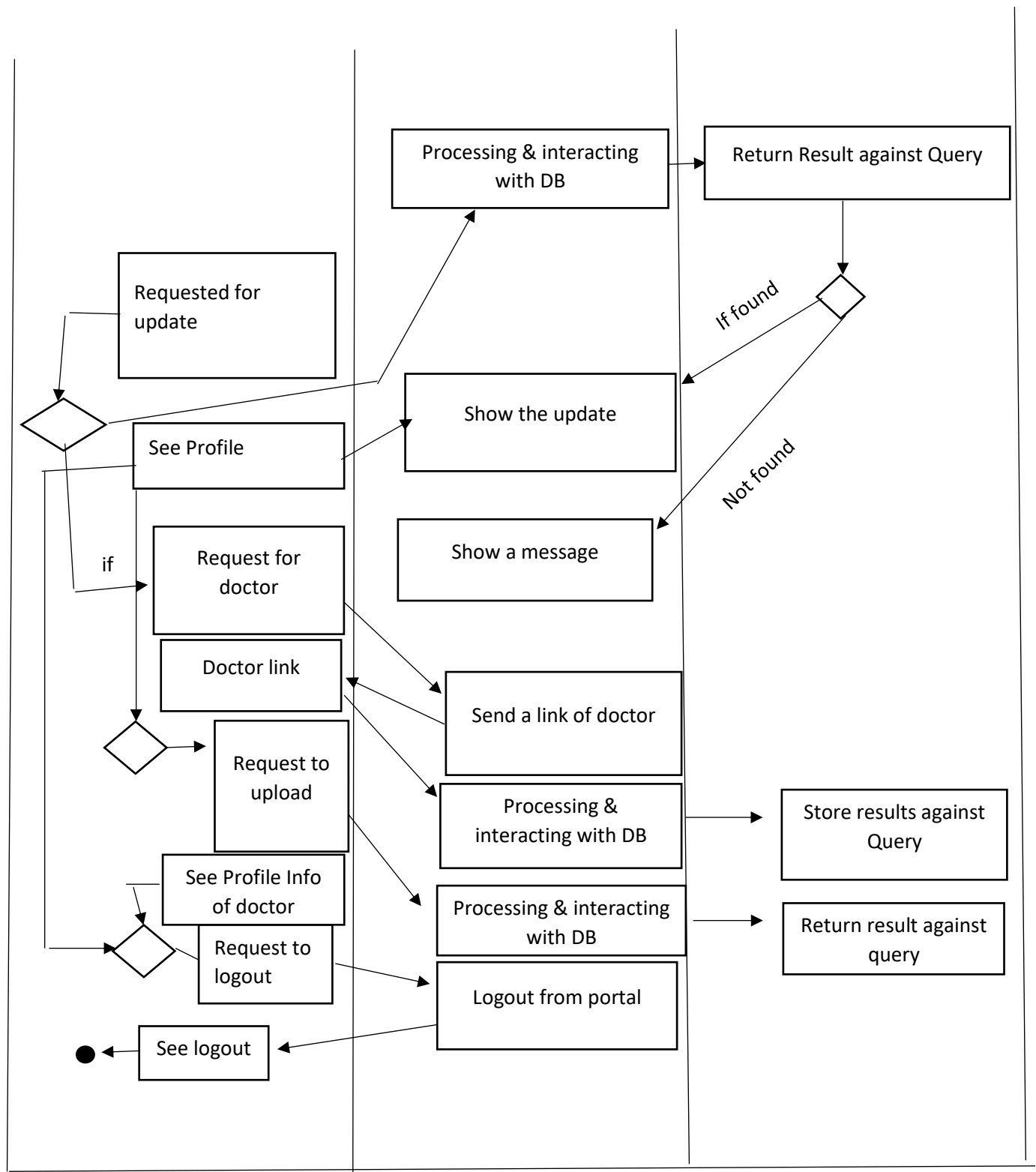
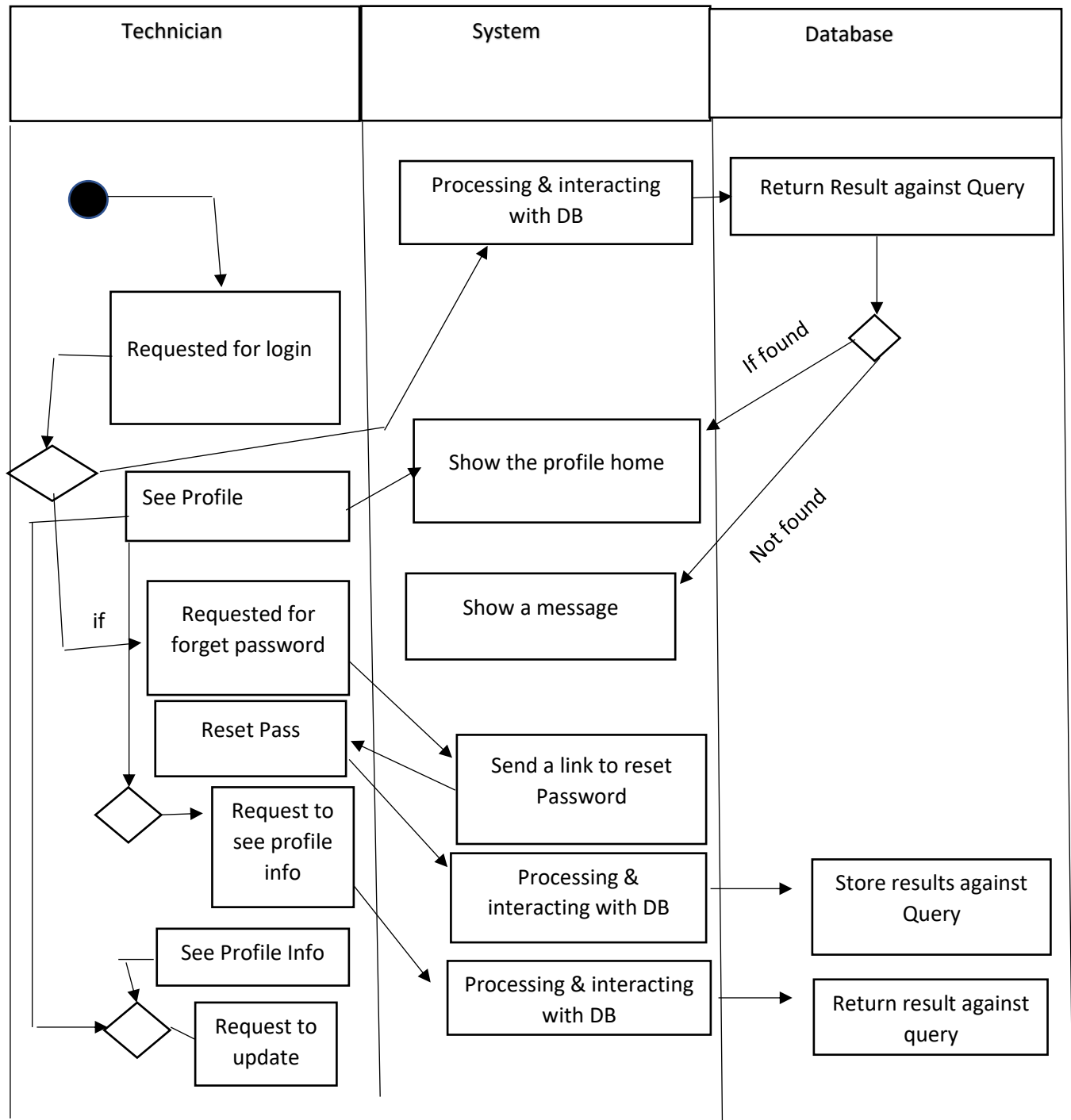


Figure 32: Patient Activity Diagram of the system

Technician:



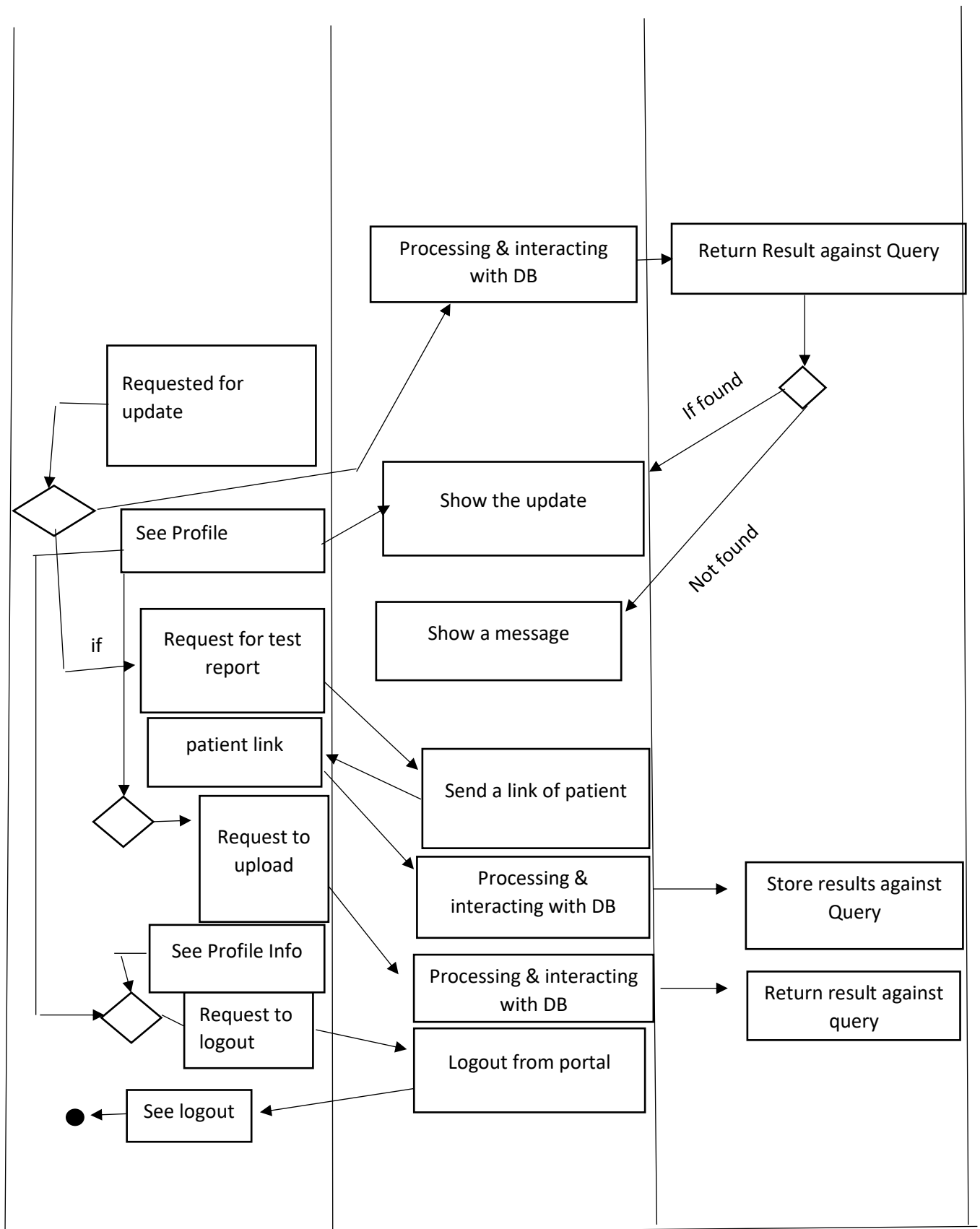
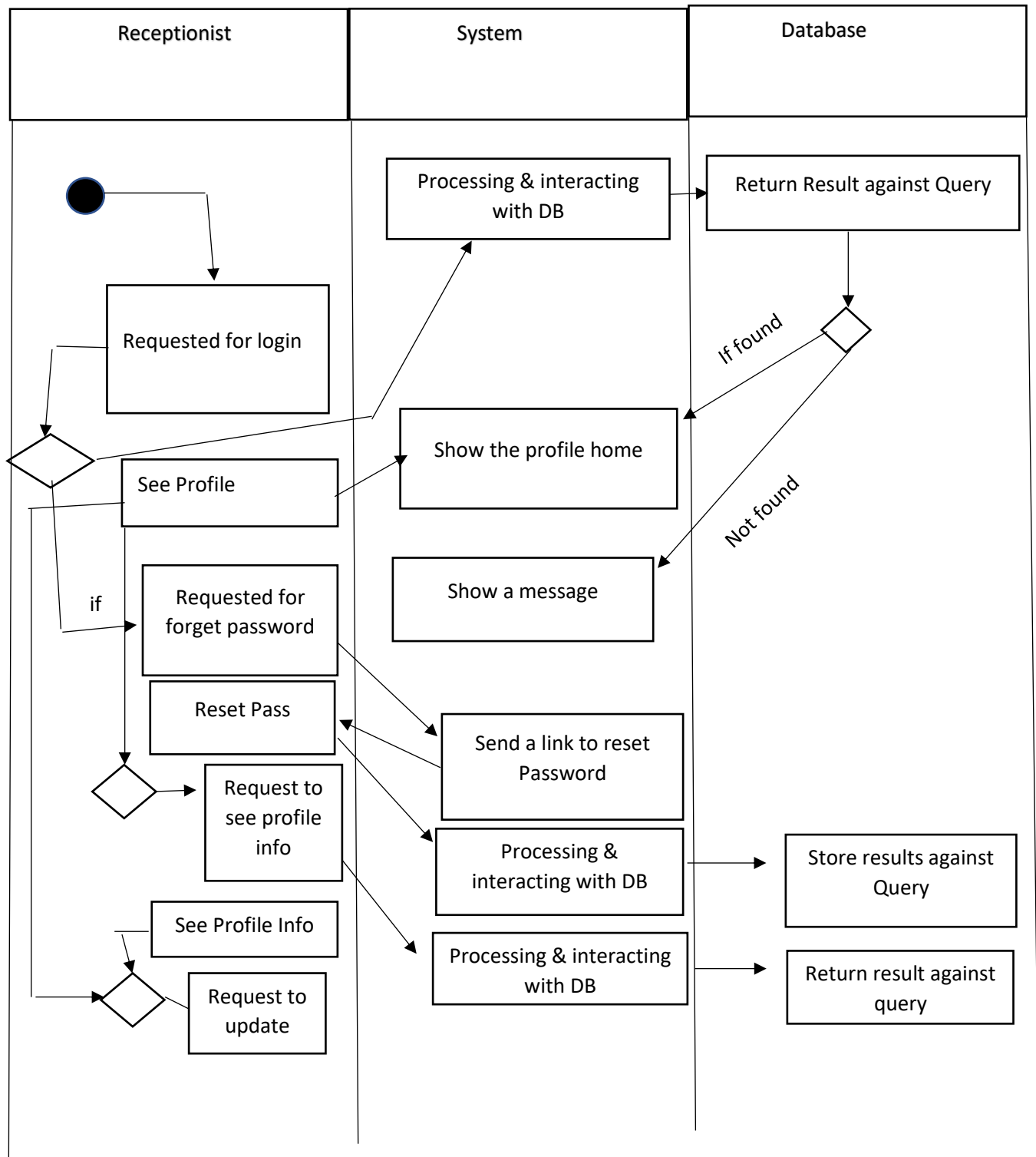


Figure 33: Technician Activity Diagram of the system

Receptionist:



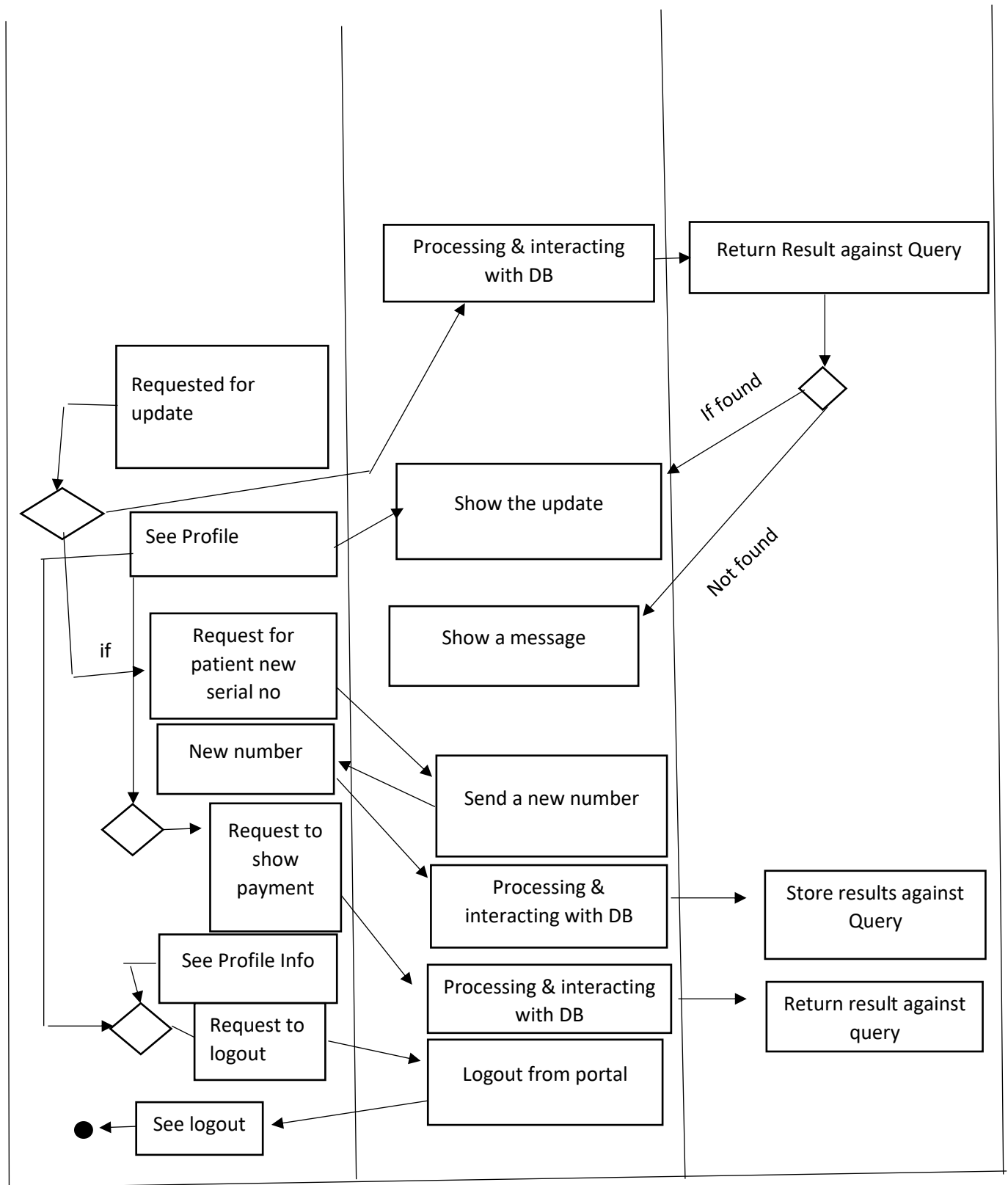
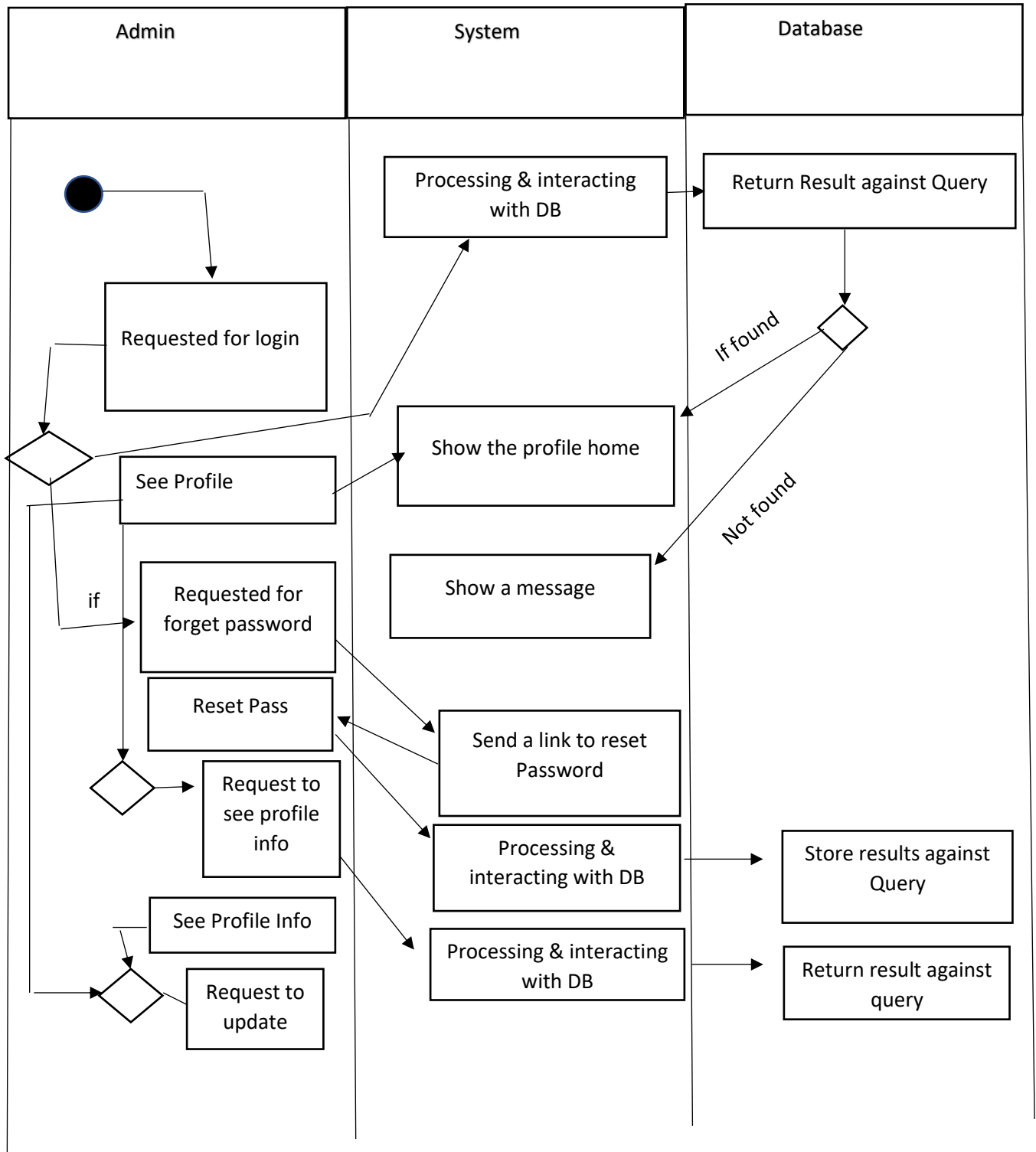


Figure 34: Receptionist Activity Diagram of the system

Admin:



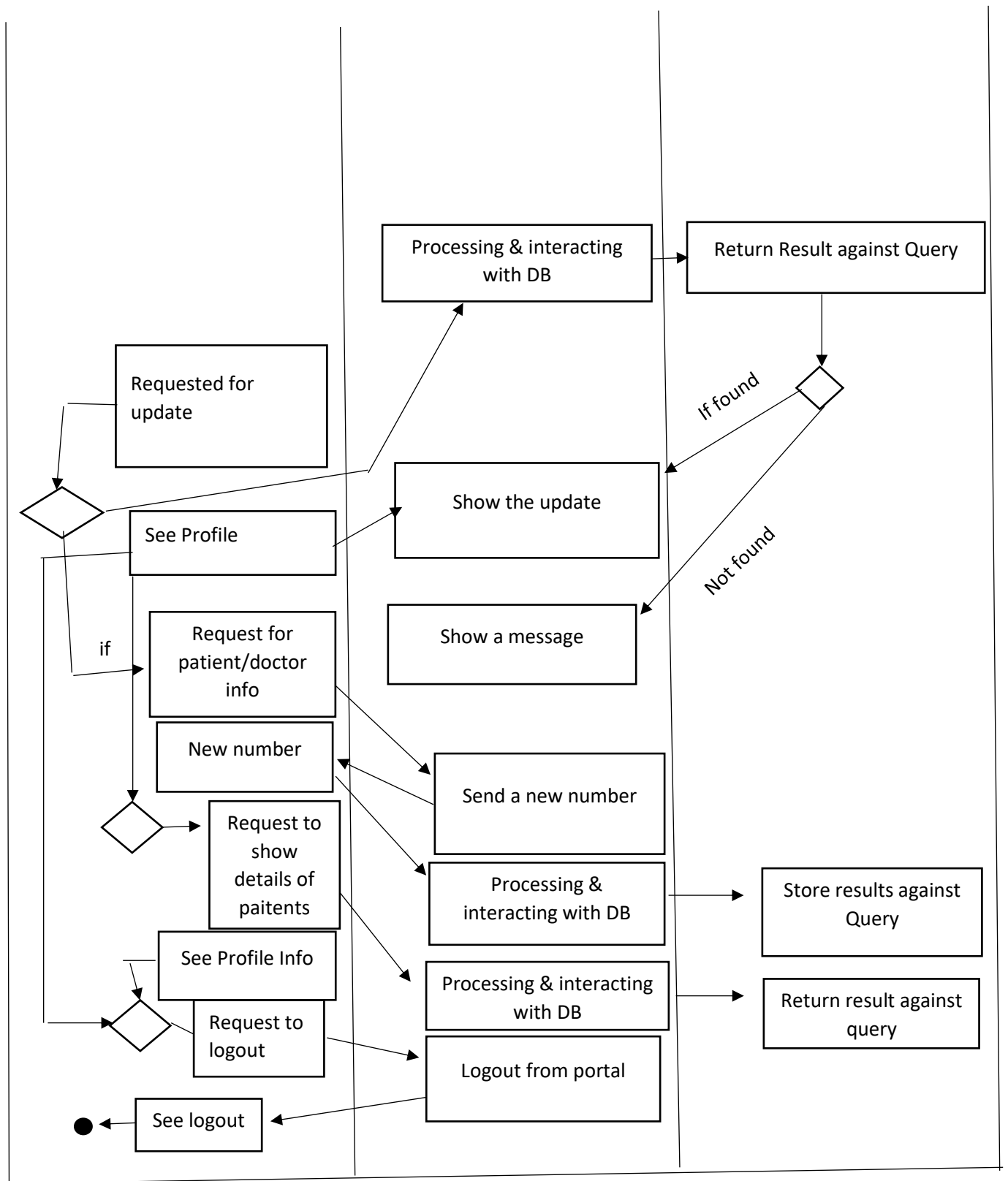


Figure 35: Admin Activity Diagram of the system

4.5 Schema diagram:

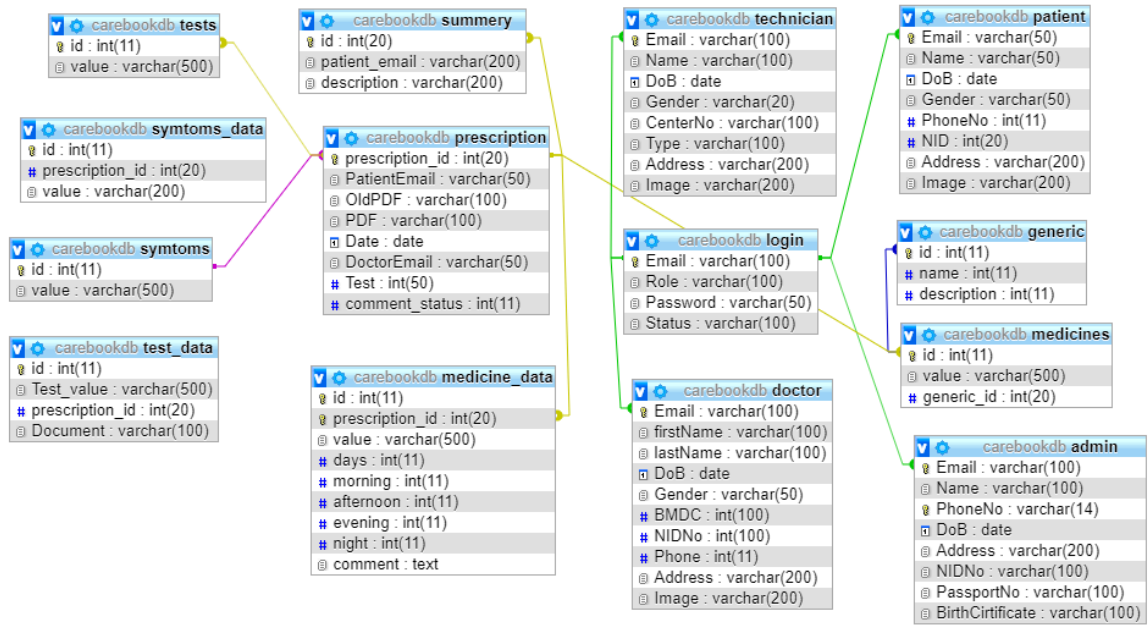


Figure 36: Schema Diagram of the system

4.6 Sequence diagram:

For Admin:

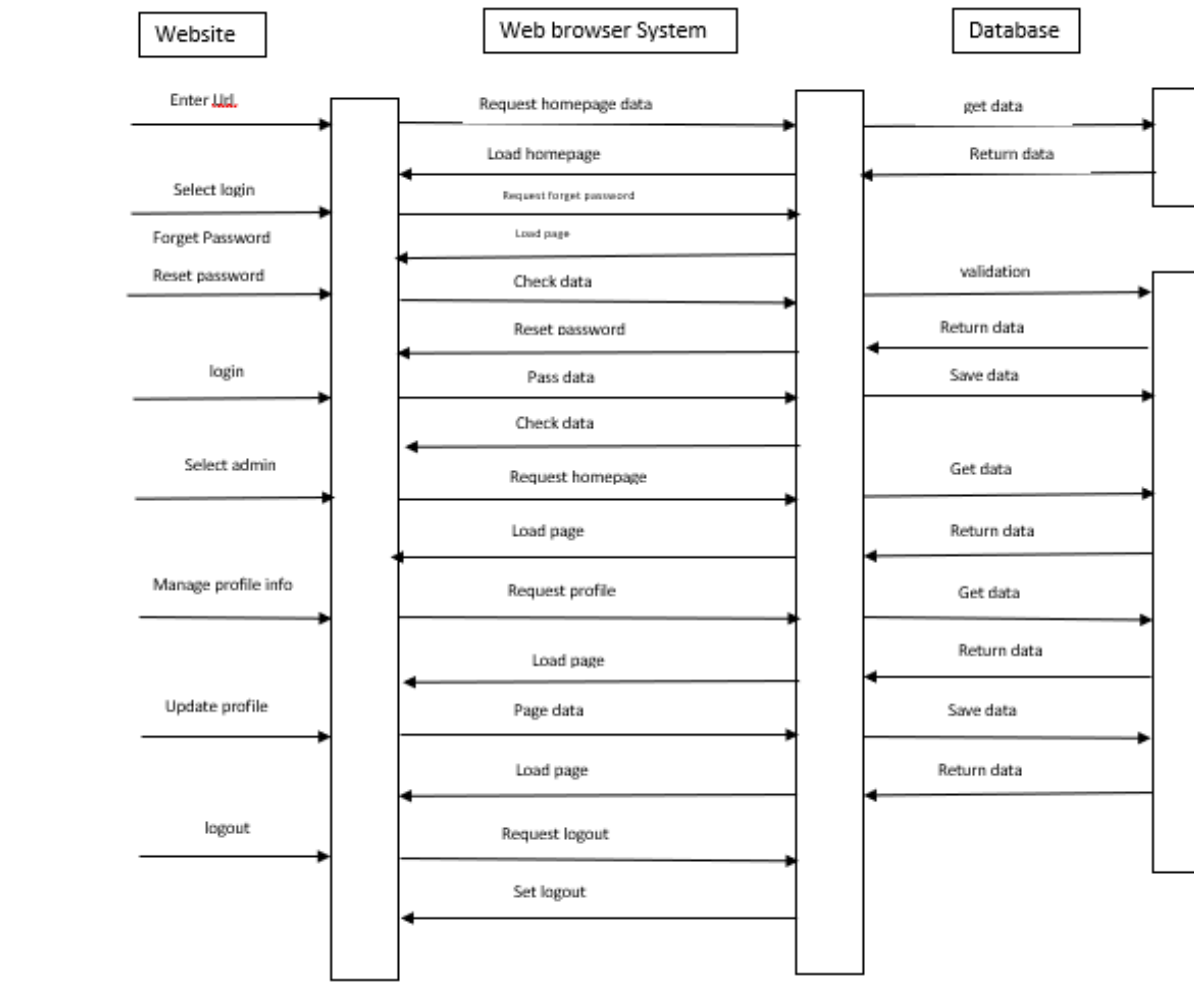


Figure 37: Sequence Diagram of admin

For Doctor:

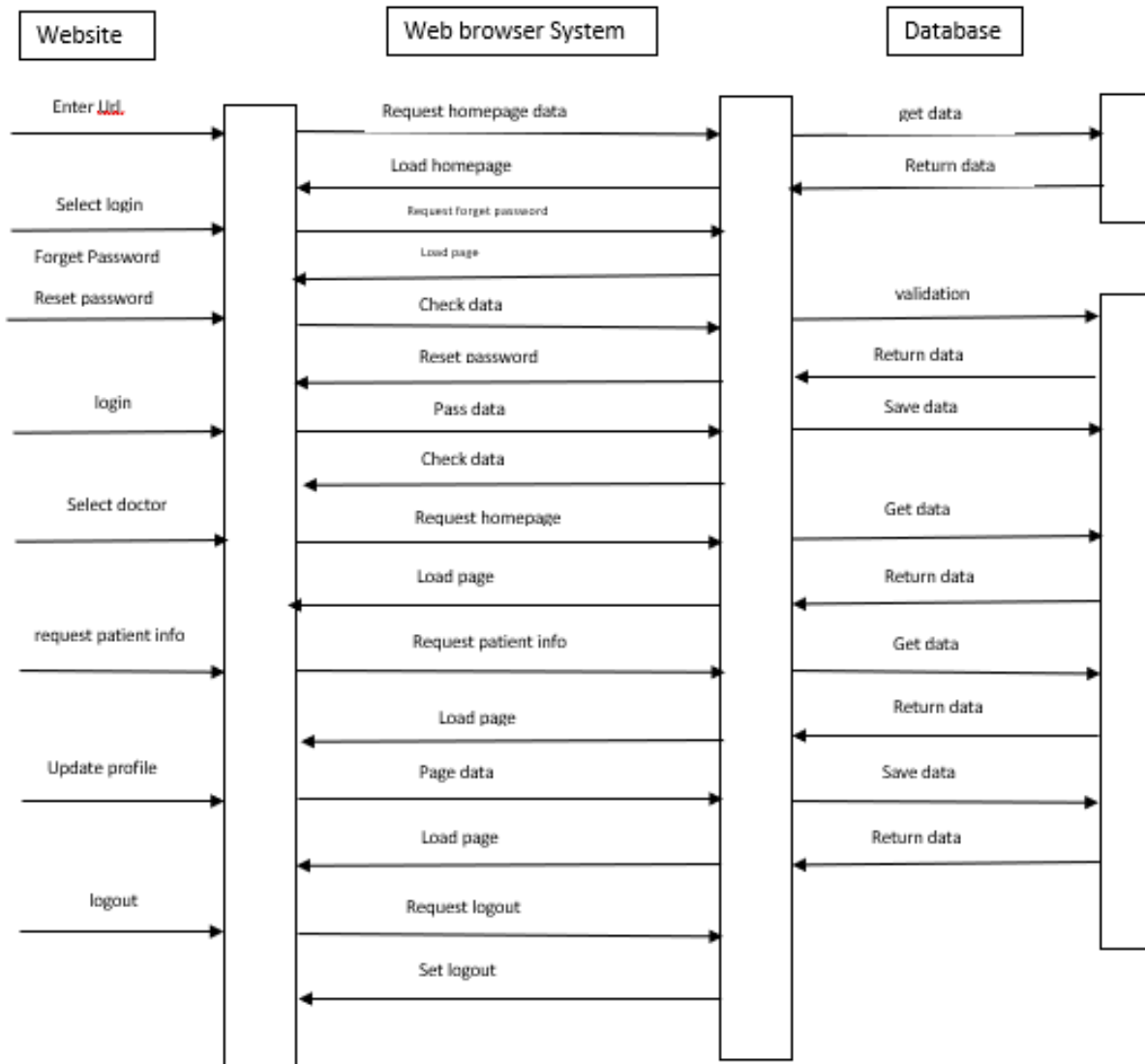


Figure 38: Sequence Diagram of doctor

For Patient:

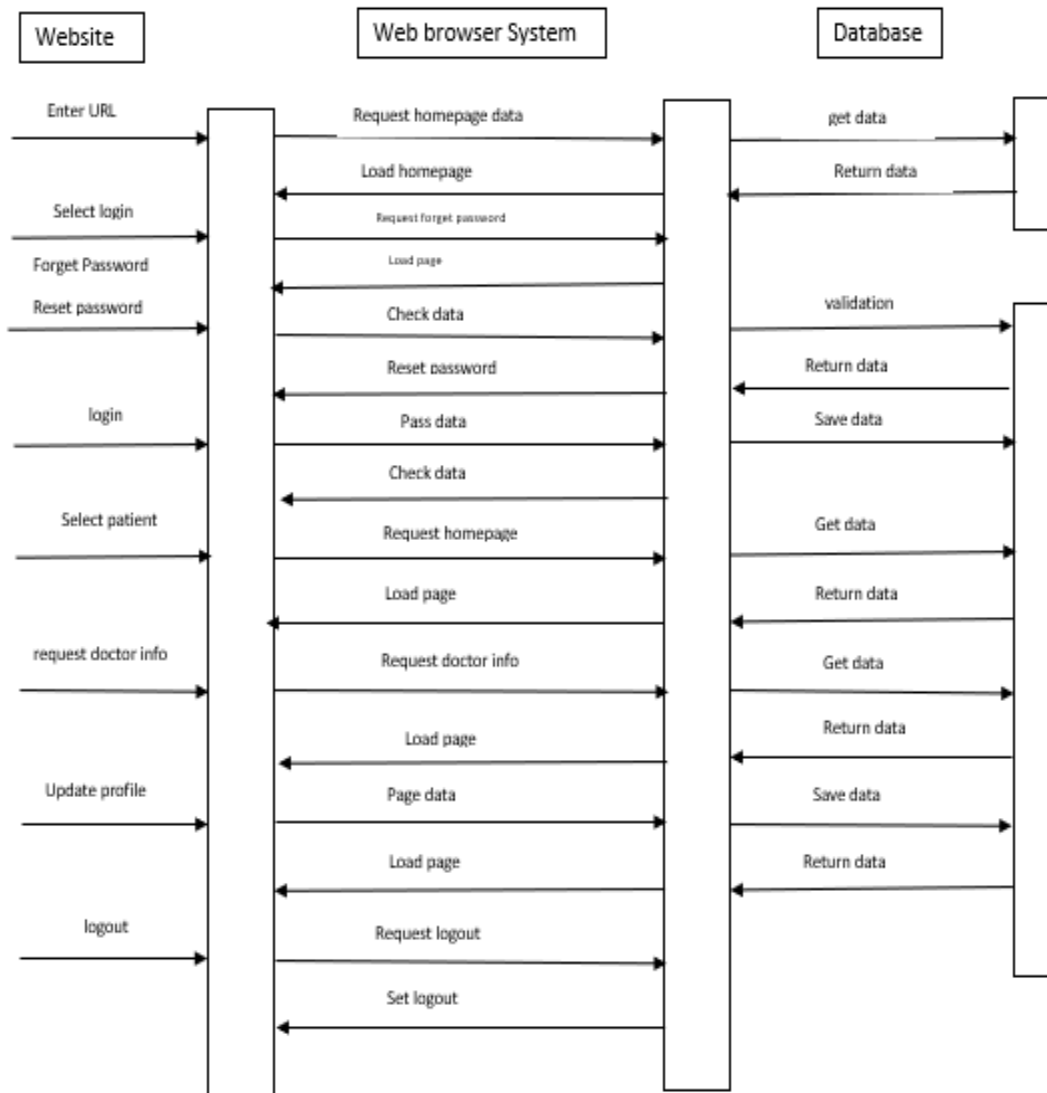


Figure 39: Sequence Diagram of patient

For Technician:

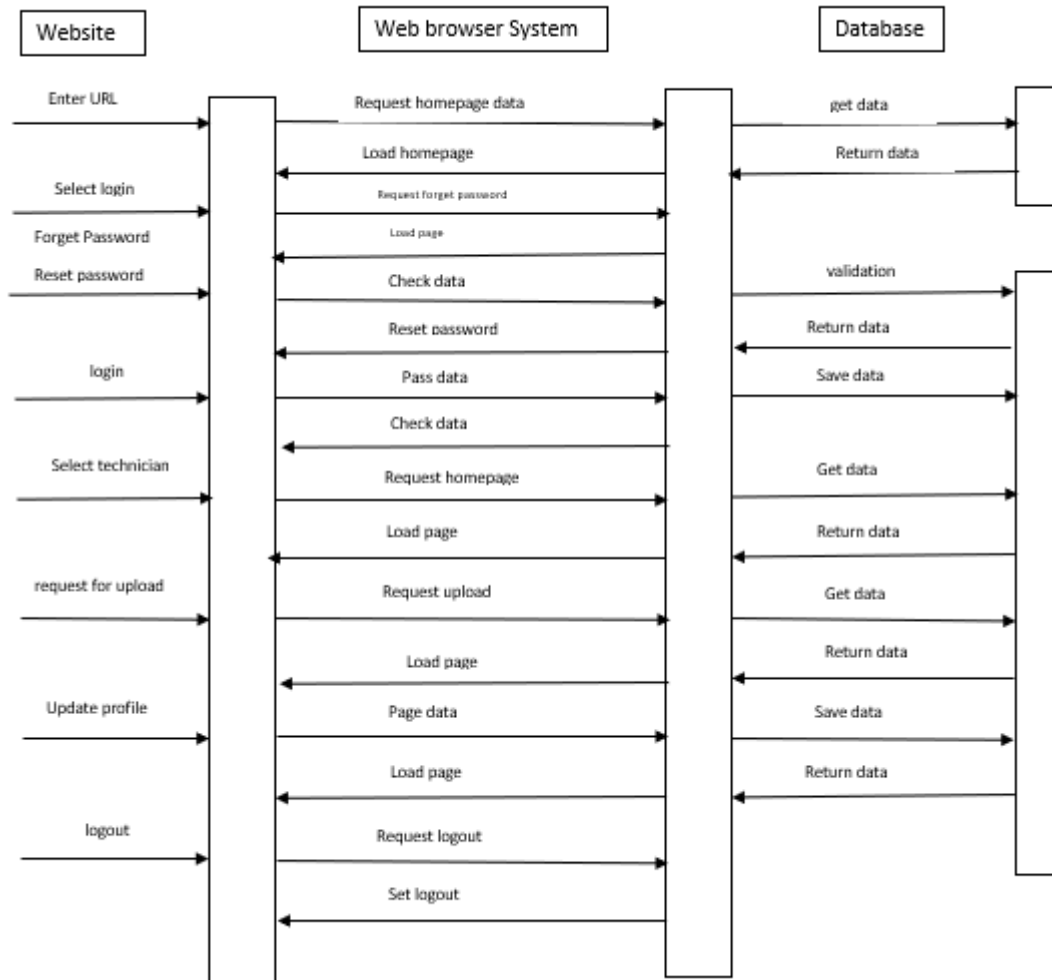


Figure 40: Sequence Diagram of technician

4.7 User Story:

SL	As a/an	I want to	So that	Acceptance Criteria
1	Admin/patient/technician/ Doctor/receptionist	login	I can use the system.	Login form
2	Admin	Add new doctor	I can add new doctor's information in the system.	Add new doctor's page
3	Admin	Approve new accounts	I can approve patient/doctors new accounts	Approval page
4	Admin	Add/update/delete	I can add/update/delete any information	Add/update/delete page
5	Admin	manage	I can manage the advertises	Managing the advertises
6	Admin	Edit	I can edit my account	Edit/reset account page
7	Admin	See all the doctor	I can see the list of doctors	All the doctor's info in pages
8	Admin	See all the patients	I can see the list of patients	All the patient's info in pages
9	Admin	Add new admins	New admins can register into the system.	Registration form for admin
10	Patient/ Admin/ Doctor/ Technician/ Receptionist	registration	I can registration into the system.	registration form
11	Patient	Store data	I can store my data into the system	Data storing
12	Patient	edit	I can edit my profile information	Edit profile page
13	Patient	search	I can search doctor according my requirements	Search in the page
14	Receptionist	Show the total payment	I can show the total payment	Show total payment in page

15	Patient	Upload	I can upload all the previous data	Upload page
16	Patient	Check notification	I can know the upgrade information	Notification page
17	Doctor	Check patient's history	I want to check patient's history	By providing Id name in the search page
18	Doctor	Add assistant	I can manage all my patients through my assistant	Receptionist page
19	Doctor	Write prescription	I can issue prescription for my patients	Writing prescription page
20	Doctor	Edit	I can edit my profile	Edit profile page
21	Doctor	Schedule update	I can update my schedule	Update my schedule in page
22	Doctor	View the test reports that I prescribed	I can give proper treatment according to the test reports	View button for every test .
23	Technician	See all the test that has been prescribed	I can see all the test that has been prescribed to patients in a prescription	By providing id name can see all the prescribed test
24	Technician	Upload test results	I can upload test results, so that patients can see their results in their account.	Upload box in page
25	Technician/Doctor/patient/Admin/receptionist	Logout	I can logout from the system	Logout button
26	Receptionist	Issue serial number	I can issue serial number for patients	Serial number page for every patient

Table 4.6.1

5.SOFTWARE TESTING PLAN

5.1 Test plan

5.1.1 Test plan for registration:

No	Test	Case	Coverage
1.1	Registration	Check all the text boxes that accept text and numbers	All the text boxes are checked with valid and invalid inputs
1.2	Registration	Check the signup button	Sign up button navigation
1.3	Registration	Check system's Registration functionality	Registration successfully
1.4	Registration	Sign up as different users	Select one user for further registration
1.5	Registration	Check the next button	Next button navigation

5.1.2 Test plan for login:

No	Test	Case	Coverage
2.1	login	Check all the text boxes that accept text and numbers	all the text boxes are checked with valid and invalid inputs
2.2	login	Check all the text boxes that accept text and numbers	Match and successfully log in With the database
2.3	login	Check all the text boxes that accept text and numbers	Login typing required data
2.4	login	Check the sign in button	Login button navigation
2.5	login	Check user interface	System user interface
2.6	Login	Reset password	all the text boxes are checked with valid and invalid inputs
2.7	Login	Reset password	Match the successful login with the database

5.1.3. Test plan for admin:

No	Test	Case	Coverage
3.1	Admin	Check all the text boxes that accept text and numbers	all the text boxes are checked with valid and invalid inputs
3.2	Admin	Check system's Registration functionality	Registration successfully
3.3	Admin	Check the signup button	Sign up button navigation
3.4	Admin	Check the profile link	View profile navigation
3.5	Admin	Check the dashboard	View the dashboard
3.6	Admin	Edit profile link	Edit profile navigation
3.7	Admin	See all the doctors link	View all the doctor's dashboard
3.8	Admin	See all the patients	View all the patient's dashboard
3.9	Admin	See rest of the admins	View rest of the admins in the dashboard
3.10	Admin	Approval/rejection of new accounts	approval/rejection navigation

5.1.4. Test plan for Doctor:

No	Test	Case	Coverage
4.1	Doctor	Check the patient's history	By providing name, id
4.2	Doctor	Check the patient's history	Match with database
4.3	Doctor	Check all the appointments	Appointment page navigation
4.4	Doctor	Edit the profile	Edit profile navigation
4.5	Doctor	Issuing prescription	Prescription navigation

4.6	Doctor	Check prescription	all the text boxes are checked with valid and invalid inputs
4.7	Doctor	Edit prescription	Edit prescription navigation
4.8	Doctor	Schedule update	Update the schedule in the dashboard

5.1.5. Test plan for Patient:

No	Test	Case	Coverage
5.1	Patient	Upload the history	Check history uploaded in the upload page
5.2	Patient	Store data	Check data store in the store data navigation
5.3	Patient	Edit the profile	Edit profile navigation
5.4	Patient	Search doctor	Check the filtered requirements
5.5	Patient	Search doctor	Match with database
5.6	Patient	Contact doctor	Check the contact information
5.7	Patient	Set appointment	Check the appointment with the doctor
5.8	Patient	Set appointment	Check the filtered requirements
5.9	Patient	Set appointment	Match with database
5.10	Patient	Notification	Check the notification navigation

5.1.6. Test plan for Technician:

No	Test	Case	Coverage
6.1	Technician	See all the test that has been prescribed	See all the test navigation
6.2	Technician	Upload test results	Upload the results in the upload page navigation

5.1.7. Test plan for Dashboard:

No	Test	Case	Coverage
7.1	Dashboard	Admin Dashboard Test	Check all the buttons
7.2	Dashboard	Admin Dashboard Test	Sizing and colours
7.3	Dashboard	Admin Dashboard Test	Check all the button's navigation
7.4	Dashboard	Admin Dashboard Test	Check all the links navigation
7.5	Dashboard	Admin Dashboard Test	Check to access
7.6	Dashboard	Admin Dashboard Test	Check view profile
7.7	Dashboard	Admin Dashboard Test	Check the UI design in web view
7.8	Dashboard	Admin Dashboard Test	Check functionalities
7.9	Dashboard	Admin Dashboard Test	Check all the doctors, all the patients, rest of the admins, approval, rejection
7.10	Dashboard	Patient Dashboard Test	Check notification, profile, store data
7.11	Dashboard	Patient Dashboard Test	Check the UI design in web view
7.12	Dashboard	Patient Dashboard Test	Check functionalities
7.13	Dashboard	Patient Dashboard Test	Check to access
7.14	Dashboard	Patient Dashboard Test	Check all the buttons
7.15	Dashboard	Patient Dashboard Test	Sizing and colours

7.16	Dashboard	Patient Dashboard Test	Check all the button's navigation
7.17	Dashboard	Patient Dashboard Test	Check all the links navigation
7.18	Dashboard	Patient Dashboard Test	Check view profile
7.19	Dashboard	Doctor Dashboard Test	Check the UI design in web view
7.20	Dashboard	Doctor Dashboard Test	Check to access
7.21	Dashboard	Doctor Dashboard Test	Check all the buttons
7.22	Dashboard	Doctor Dashboard Test	Sizing and colours
7.23	Dashboard	Doctor Dashboard Test	Check view profile
7.24	Dashboard	Doctor Dashboard Test	Check all the links navigation
7.25	Dashboard	Doctor Dashboard Test	Check schedule update, patient's history, appointments
7.26	Dashboard	Technician Dashboard Test	Check the UI design in web view
7.27	Dashboard	Technician Dashboard Test	Check to access
7.28	Dashboard	Technician Dashboard Test	Check all the buttons
7.29	Dashboard	Technician Dashboard Test	Sizing and colours
7.30	Dashboard	Technician Dashboard Test	Check view profile
7.31	Dashboard	Technician Dashboard Test	Check all the links navigation

7.32	Dashboard	Technician Dashboard Test	Check all the test, upload test
------	-----------	---------------------------	---------------------------------

5.1.8. Test plan for Doctor's Promotion ad upload:

No	Test	Case	Coverage
8.1	Ad upload	Check in as an admin of the system	Check the update
8.2	Ad upload	Check in as an admin of the system	File format
8.3	Ad upload	Check in as an admin of the system	Check the expired date of ad
8.4	Ad upload	Check in as an admin of the system	Check the uploading buttons
8.5	Ad upload	Check in as an admin of the system	Check the box with all format of files

5.1.9. Test plan for view page:

No	Test	Case	Coverage
9.1	View page	UI design & properties	Check button properties
9.2	View page	UI design & properties	Check attached photo sizing
9.3	View page	UI design & properties	Check file container size
9.4	View page	UI design & properties	Properties
9.5	View page	General functionalities	Check button navigations & correct functionalities

9.6	View page	General functionalities	Check the database values & the output of the posts
-----	-----------	-------------------------	---

5.1.10. Test plan for additional test:

No	Test	Case	Coverage
10.1	Additional	Navbar Properties	Check the link director
10.2	Additional	Navbar Properties	texts
10.3	Additional	Navbar Properties	Check the username properties and dropdown menu
10.4	Additional	UI design in web	Check high prioritized
10.5	Additional	View	Sections
10.6	Additional	functional	Functionalities performance
10.7	Additional	Requirements performance	Constraints and system dependencies

5.2 Testing Methodology

Our project CareBook is a web-based application. Here we are performing three types of testing. They are:

- System testing
- Regression testing
- User Acceptance testing.

5.2.1 System Testing

We tested to the program by system testing to find the defects and noted severity of defects, summary of defects and commented on them. System testing is the testing of a complete and fully integrated software product. System tests the overall system operations as a whole, typically from a customer's perspective. System testing presumes that all components have been previously, successfully, integrated. System testing is done after integration testing is complete. System testing main focus is to verify that the customer requirements are fulfilled. System testing should test functional and non-functional requirements of the software. System testing is black-box testing and often done by independent testers.

Test Case ID	Test Area	Test Case	Tester	Severity Of Defects	Summary Of Defects	Comments
1.1	Registration	Check all the text boxes that accept text and numbers	Fahim	None	None	Passed
1.2	Registration	Check the submit button	Fahim	None	None	Passed
1.3	Registration	Check system's Registration functionality	Fahim	None	None	Passed
1.4	Registration	Sign up as different users	Fahim	None	None	Passed
1.5	Registration	Check the all the buttons	Fahim	None	None	Passed
2.1	login	Check all the text boxes that accept text and numbers (valid and invalid inputs)	Fahim	None	None	Passed
2.2	login	Check all the text boxes that accept text and numbers (Match and successfully log in with the database)	Fahim	None	None	Passed
2.3	login	Check all the text boxes that accept text and numbers (Login typing required data)	Fahim	None	None	Passed
2.4	login	Check the sign in button	Fahim	Medium	No error message given while e-mail and password fields are empty	Failed
2.5	login	Check user interface	Fahim			Passed
2.6	login	Reset password (valid and invalid inputs)	Fahim	High	Not Working	Failed
2.7	login	Reset password (Match the successful login with the database)	Fahim	High	Not Working	Failed
3.1	Admin	Check all the text boxes that accept text and numbers (valid and invalid inputs)	Fahim	None	None	Passed
3.2	Admin	Check system's Registration functionality (Registration successfully)	Fahim	None	None	Passed

3.3	Admin	Check the signup button	Fahim	None	None	Passed
3.4	Admin	Check the profile link	Fahim	None	None	Passed
3.5	Admin	Check the dashboard	Fahim	None	None	Passed
3.6	Admin	Edit profile link	Fahim	None	None	Passed
3.7	Admin	See all the doctors list	Fahim	None	None	Passed
3.8	Admin	See all the patients	Fahim	None	None	Passed
3.9	Admin	See rest of the admins	Fahim	High	Not Available	Failed
3.10	Admin	Approval/rejection of new accounts	Fahim	None	None	Passed
4.1	Doctor	Check the patient's history (By providing name, id)	Fahim	None	None	Passed
4.2	Doctor	Check the patient's history (Match with database)	Fahim	None	None	Passed
4.3	Doctor	Check all the appointments	Fahim	Low	Not Available	Failed
4.4	Doctor	Edit the profile	Fahim	None	None	Passed
4.5	Doctor	Issuing prescription	Fahim	None	None	Passed
4.6	Doctor	Check prescription	Fahim	High	PDF format is not supporting	Failed
4.7	Doctor	Edit prescription	Fahim	None	None	Passed
4.8	Doctor	Schedule update	Fahim	None	None	Passed
5.1	Patient	Upload the history	Fahim	None	None	Passed
5.2	Patient	Store data	Fahim	None	None	Passed
5.3	Patient	Edit the profile	Fahim	None	None	Passed
5.4	Patient	Search doctor (Check the filtered requirements)	Fahim	None	None	Passed
5.5	Patient	Search doctor (Match with database)	Fahim	None	None	Passed
5.6	Patient	Contact doctor	Fahim	Medium	Not Available	Failed
5.7	Patient	Set appointment (Check the appointment with the doctor)	Fahim	Medium	Not Available	Failed

5.8	Patient	Set appointment (Check the filtered requirements)	Fahim	Medium	Not Available	Failed
5.9	Patient	Set appointment (Match with database)	Fahim	Medium	Not Available	Failed
5.10	Patient	Notification	Fahim			Passed
6.1	Technician	See all the test that has been prescribed	Fahim	Medium	Not Available	Failed
6.2	Technician	Upload test results	Fahim	None	None	Passed
7.1	Dashboard	Admin Dashboard Test (Check all the buttons)	Fahim	None	None	Passed
7.2	Dashboard	Admin Dashboard Test (Sizing and colours)	Fahim	None	None	Passed
7.3	Dashboard	Admin Dashboard Test (Check all the button's navigation)	Fahim	None	None	Passed
7.4	Dashboard	Admin Dashboard Test (Check all the links navigation)	Fahim	None	None	Passed
7.5	Dashboard	Admin Dashboard Test (Check to access)	Fahim	None	None	Passed
7.6	Dashboard	Admin Dashboard Test (Check view profile)	Fahim	None	None	Passed
7.7	Dashboard	Admin Dashboard Test (Check the UI design in web view)	Fahim	None	None	Passed
7.8	Dashboard	Admin Dashboard Test (Check functionalities)	Fahim	None	None	Passed
7.9	Dashboard	Admin Dashboard Test (Check all the doctors, all the patients, rest of the admins, approval, rejection)	Fahim	None	None	Passed
7.10	Dashboard	Patient Dashboard Test (Check notification, profile, store data)	Fahim	Medium	Patient notification is not showing properly	Failed
7.11	Dashboard	Patient Dashboard Test (Check the UI design in web view)	Fahim	None	None	Passed
7.12	Dashboard	Patient Dashboard Test (Check functionalities)	Fahim	Medium	In archive file is not uploading	Failed

7.13	Dashboard	Patient Dashboard Test (Check to access)	Fahim	None	None	Passed
7.14	Dashboard	Patient Dashboard Test (Check all the buttons)	Fahim	None	None	Passed
7.15	Dashboard	Patient Dashboard Test (Sizing and colours)	Fahim	Low	Profile photo is very small, Colour of Sign-out is not focusing	Failed
7.16	Dashboard	Patient Dashboard Test (Check all the button's navigation)	Fahim	None	None	Passed
7.17	Dashboard	Patient Dashboard Test (Check all the links navigation)	Fahim	None	None	Passed
7.18	Dashboard	Patient Dashboard Test (Check view profile)	Fahim	None	None	Passed
7.19	Dashboard	Doctor Dashboard Test (Check the UI design in web view)	Fahim	None	None	Passed
7.20	Dashboard	Doctor Dashboard Test (Check to access)	Fahim	None	None	Passed
7.21	Dashboard	Doctor Dashboard Test (Check all the buttons)	Fahim	High	Discard button is not working	Failed
7.22	Dashboard	Doctor Dashboard Test (Sizing and colours)	Fahim	None	None	Passed
7.23	Dashboard	Doctor Dashboard Test (Check view profile)	Fahim	None	None	Passed
7.24	Dashboard	Doctor Dashboard Test (Check all the links navigation)	Fahim	None	None	Passed
7.25	Dashboard	Doctor Dashboard Test (Check schedule update, patient's history, appointments)	Fahim	None	None	Passed
7.26	Dashboard	Technician Dashboard Test (Check the UI design in web view)	Fahim	None	None	Passed
7.27	Dashboard	Technician Dashboard Test (Check to access)	Fahim	None	None	Passed
7.28	Dashboard	Technician Dashboard Test (Check all the buttons)	Fahim	None	None	Passed

7.29	Dashboard	Technician Dashboard Test (Sizing and colours)	Fahim	None	None	Passed
7.30	Dashboard	Technician Dashboard Test (Check view profile)	Fahim	None	None	Passed
7.31	Dashboard	Technician Dashboard Test (Check all the links navigation)	Fahim	None	None	Passed
7.32	Dashboard	Technician Dashboard Test (Check all the test, upload test)	Fahim	None	None	Passed
8.1	Ad upload	Check in as an admin of the system (Check the update)	Fahim	Medium	Not Available	Failed
8.2	Ad upload	Check in as an admin of the system (File format)	Fahim	Medium	Not Available	Failed
8.3	Ad upload	Check in as an admin of the system (Check the expired date of ad)	Fahim	None	Future Development	None
8.4	Ad upload	Check in as an admin of the system (Check the uploading buttons)	Fahim	None	Future Development	None
8.5	Ad upload	Check in as an admin of the system (Check the box with all format of files)	Fahim	None	Future Development	None

Table-15: System testing

5.2.2 Regression Testing

Regression tests determine that the system remains stable as it cycles through the integration of other subsystems and through maintenance tasks. Testing that is performed after making a functional improvement or repair to the program. Its purpose is to determine if the change has regressed other aspects of the program. In this category, new tests are not designed; instead, test cases are selected from the existing pool and executed. Finally, we tested our project by an own user and write report and also found the defects.

The table below summarizes the test cases employed for regression testing and the test results obtained for each test case:

Test Case ID	Date Tested	Tester	Pass /Fail	Severity of Defect	Summary of Defect	Closed prior to Production Release ?	Comments
11.1	20/10/2019	Fahim	Passed	High	Without filling up any requirement but add item.	Yes	Problem Solved.
11.2	20/10/2019	Fahim	Passed	Medium	Guest browser rated which is not in the requirement	Yes	Problem Solved
11.3	20/10/2019	Fahim	Passed	High	Validation problem in Patients and doctor's login	Yes	Problem solved
11.4	20/1/2019	Fahim	Passed	Medium	Now can delete the pictures	Yes	Problem solved

Table-16: Regression testing

5.2.3 User Acceptance Testing

Acceptance testing is Formal testing for product evaluation. Formal testing conducted to determine whether a system satisfies its acceptance criteria. Performed by customers/end users (preferably). Verifies functionality and usability of the software. Conducted prior to software being released to live operation. At this level of testing, the software should be in a sufficiently defect-free state to permit the emphasis to change. Acceptance testing is usually the last step before the user/customer takes the possession of the software.

The table below summarizes the test cases employed for user acceptance testing and the test results obtained for each test case:

Test Case ID	Test Area	Test Case	Tester	Severity Of Defects	Summary Of Defects	Comments
9.1	View page	UI design & properties (Check button properties)	Fahim	None	None	Passed
9.2	View page	UI design & properties (Check attached photo sizing)	Fahim	Low	Profile photo of user is very small	Failed
9.3	View page	UI design & properties (Check file container size)	Fahim	None	None	Passed
9.4	View page	UI design & properties (Properties)	Fahim	None	None	Passed
9.5	View page	General functionalities (Check button navigations & correct functionalities)	Fahim	Medium	Next page buttons are not working	Failed
9.6	View page	General functionalities (Check the database values & the output of the posts)	Fahim	High	Prescriptions PDF format is not showing correctly	Failed
10.1	Additional	Navbar Properties (Check the link director)	Fahim	None	None	Passed
10.2	Additional	Navbar Properties (texts)	Fahim	None	None	Passed
10.3	Additional	Navbar Properties (Check the username properties and dropdown menu)	Fahim	None	None	Passed
10.4	Additional	UI design in web (Check high prioritized)	Fahim	None	None	Passed
10.5	Additional	View (Sections)	Fahim	None	None	Passed
10.6	Additional	Functional (Functionalities performance)	Fahim	High	Page loading and response time higher	Failed
10.7	Additional	Requirements performance (Constraints and system dependencies)	Fahim	High	Not Satisfactory	Failed

Table-17: Acceptance testing

6. Other Non-functional Requirements

6.1 Performance Requirements

1. System cannot loss its data.
2. The system shall be available 99% of the time unless previously announced for scheduled maintenance or backup.
3. System should be available 24*7, in case of development and maintenance issues user should be notified.

6.2 Security Requirements

1. Software shall not be accessible from external networks.
2. Hardware shall be secure from external tampering.

6.3 Software Quality Attributes

1. **System Availability:** The system shall be available 99% of the time unless previously announced for scheduled maintenance or backup.
2. **System Reset:** The system shall provide the ability to reset all settings to the default or a saved configuration.
3. **Developer Access:** The system administrator shall be able to grant developer privileges to users.
4. **Interface is accessible from more than one convenient device:** If the system interface is not readily accessible, then it will not be as easy to control and will offer little convenience.
5. **System restore:** The system shall provide the ability to restore all settings and data from the database in case of system failure.

6.4 Business Plan

6.4.1 Marketing:

CareBook.com is a platform to connect between doctor, patient and technician in Bangladesh. Mainly we will focus in Dhaka city first then step by step all over in Bangladesh. First, CareBook.com will provide a platform for doctor, patient and technician, in future we will provide online medical service by whom customer (patient) get their service from home. Now initially our platform has three users' doctor, patient and technician. In future development we want to involve more users like receptionist, pharmacist, pathologist, radiologist etc.

6.4.2 Objectives

- CareBook.com serves the patient, doctor and technician.
- Doctor working process will be easier than traditional process.

- Technician will also get facility from our platform.
- Patient does not have to carry their documents, only they need to remember their id/email.
- Doctor have a review system, which is based on by patient's vote.
- Patient will easily get good doctor address his/her nearest place from the rating of a doctor.

6.4.3 Key to Success

- Professional quality of service offered
- Reliability – being available that user personal information will not be disclosed.
- Effective collaboration with other community professionals (physicians, hospital, and other organizations)

6.4.4 Mission

Our values are simple. CareBook.com Services strives to offer excellent and affordable health care and community-based social services to individuals and families of Bangladeshi people.

It is our goal to give good service, caring, and authentic doctor and technician who are responsive to the needs of our platform, their families, and the communities we serve. Each staff member of platform administrative will meet the Government of Bangladesh educational and training requirements for the services they provide. We encourage and support continued education of each service provider. In turn, our platform will provide staff with competitive compensation, an inviting work environment, and knowledgeable, trustworthy management and direction.

6.5 Business Analysis

6.5.1 Stakeholder: CareBook.com will serve the community free. In business field our main stakeholder is medicine/Drug Company. Our secondary stakeholder is patient who will get service from CareBook.com

6.5.2 Business Objectives

- CareBook.com main objective to serve the community.
- Getting sponsor from market.
- Doctors and patient will use the system to easily interact with each other.
- Our strength is Bangladesh is a developing country and government is trying to develop our system to digital where this type of concept is very new in Bangladesh.
- Our weakness is the people, who are not comfortable to interact with machine.
- Ensuring that our all stakeholder is getting good service from our system.
- Our platform will offer a review system, by whom there will be, no injustice to select good/required doctor for patient.
- Drug/Medicine company will post their advertise in our platform in doctor page.

6.5.3 Evaluate Options

To achieve our goal our system will be more user-friendly, so that users are happy, more interacting with the system and not discouraged to use the system. And the other stakeholder, by whom we get sponsor and advertise, we also focus them that our system will be more used in future and show our system architecture and marketing plan. There has an opportunity for drug and Pharmacy Company to post advertise their product in our platform, we supposed to decrease the time waste of doctor and increase our business value and revenue from this option.

6.5.4 Scope

At the very beginning our main task is to introduce community that our system is helpful for community and them. In addition, show the advantage to use the system, as patient will not carry their documents, doctor not to write prescription cause all this part will be digital and patient getting their good doctor easily.

6.5.5 Future Marketing:

- We will be trying to add the instant treatment in our platform.
- Our most effective feature will be medical services from home.
- Patient will book appointments for doctor.
- Centralized systems for various hospitals and clinics.

7. Other Requirements

Appendix A: Glossary

Term	Full form
DB	Database
Pass	Password
UI	User Interface
Pay	Payment
CB	CareBook

8. Project Scheduling

Task	Time (week)
Field study	0-1
Study on previous works	0-1
Study on SDLC	1-2
Prepare user story	1-2
Identify user requirements	2-3
Develop Use case diagram	3-4
Develop Activity diagram	3-4
Develop Class diagram	3-4
Develop E-R diagram	4-5
Develop Schema diagram	4-5
V-Model	5-6
Create Database	5-6
Create user interface	6-8
Software Development	8-12
Prepare Test plan	12-13
Software Testing and Debugging	13-15
Documentation	15-16

Total Project Time: 16 Weeks

Start Date: 15.07.19

Release Date: 19.11.19

Conclusion:

The purpose of this document is to build a common platform for the medical industry which will be accessible by anyone. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external systems.

References

1. OMG. "Unified Modeling Language Specification", Superstructure Version 2.1.1, February 2007
2. "Redirect" *Omg.org*, 2019. [Online]. Available: <http://www.omg.org/technology/documents/formal/uml.htm>.
3. "Creating Online Stores Worldwide | osCommerce", *Oscommerce.com*, 2019. [Online]. Available: <http://www.oscommerce.com/solutions/documentation>.
4. Object-oriented design with applications [Grady Booch, 2nd Edition]
5. Model-based software engineering [G. Bruno]
6. Cutter Consortium, Software project success and failure
7. For coding Architecture
8. "Architecture Guide: ASP.NET MVC Framework + N-tier + Entity Framework and Many More", *Codeproject.com*, 2019. [Online]. Available: <https://www.codeproject.com/Articles/70061/Architecture-Guide-ASP-NET-MVC-Framework-N-tier-En>.
9. "SDLC-Agile Model- Tutorialspoint", *Tutorialspoint.com*, 2019. [Online]. Available: https://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm.
10. *Lucidchart.com*, 2019. [Online]. Available: <https://www.lucidchart.com/>.
11. "ERDPlus", *Erdplus.com*, 2019. [Online]. Available: <https://erdplus.com/#/>.
12. "Flowchart Maker & Online Diagram Software", *Draw.io*, 2019. [Online]. Available: <https://www.draw.io/>.
13. "Diagram Maker | Online Diagram Software", *Creately.com*, 2019. [Online]. Available: <https://creately.com/>.
14. R. Smith et al., "Symptom Checker, Health Information and Medicines Guide", *Patient.info*, 2019. [Online]. Available: <https://patient.info/>.
15. B. Doctors, "Best Doctors | Made available by Trustmark", *Bestdoctors.com*, 2019. [Online]. Available: <https://bestdoctors.com/>.
16. "Login - Patient Portal", *Nextmd.com*, 2019. [Online]. Available: <https://www.nextmd.com/ud2/Login/Login.aspx>.
17. "PatientConnect360- A complete digital health card", *Patientconnect360.com*, 2019. [Online]. Available: <http://patientconnect360.com/>.