MEDICAL ASSISTANT

FINAL PROJECT

Asad Ashraf

Syed Usama Shah

Muhammad Rizwan Malik

Supervised by

Muhammad Arif

Submitted for the partial fulfillment of BS Software Engineering degree to the Faculty of Engineering and Computer Science

NATIONAL UNIVERSITY OF MODERN LANGUAGES

ISLAMABAD

December, 2022

ABSTRACT

Remembering the exact time of taking exact prescribed medicines can be very challenging for some especially for those who don't have anyone to look after them. And in the times of current pandemic, this issue seems to be a common problem for many people. So, there should be some sort of assistant for these patients who can remind them to take proper medicines and right on time. There are some formal ways like through an alarm clock app or hiring a care taker. But these methods either don't fulfill the needs of a patient or can be very challenging. To overcome these problem, we have proposed a system named as "Medical Assistant".

Medical assistant will be an android based platform that will help the people in remembering the time and the medicine they need to take. As the problem of remembering the time and the medicine is mostly faced by older patients, so our system will have simple interfaces where they either choose to set an alarm for a reminder themselves or by using and OCR text reader that would read the test from the doctor's prescription script and tell the patient about what's written and if there is time written, then it will automatically set a reminder of that time. It will also have a QR scanner that will scan the QR code of a medicine if that medicine was bought from a medical store or purchased online and has a QR code. That QR code would give the information about the medicine and general timing of when and how they can be taken. Patients will also be able to check their records of medicines they took. A chat bot will also be available for patients to use that will help identify the disease according to the symptoms provided by the patient.

Our System will address the users the major aspects that have to be considered when developing a single platform for two different kind of users. Our system is a Mobile Application where user will be able to perform tasks on a friendly and easy-to-use UI. From Mobile Application Patient can perform their respective tasks.

FINAL PROJECT APPROVAL CERTIFICATE

It is certified that project report titled ‘Medical Assistant’ submitted by Muhammad Rizwan Malik, Asad Ashraf, Syed Usama Shah for the complete fulfilment of the requirement of “Bachelor’s Degree in Software Engineering” is approved.

COMMITTEE

Dean Engineering & CS — Signature: _____

Dr. Basit Shahzad

HoD Software Engineering — Signature: _____

Dr. Muzafar Khan

Head Project Committee — Signature: _____

Naveed Ahmed

Supervisor — Signature _____

Muhammad Arif

Dated: _____

UNDERTAKING

We hereby declare that our dissertation is entirely our work and genuine / original. We understand that in case of discovery of any PLAGIARISM at any stage, our group will be assigned an F (FAIL) grade and it may result in withdrawal of our Bachelor's degree.

Group Members	**Signature**
1. Muhammad Rizwan Malik	_____
2. Asad Ashraf	_____
3. Syed Usama Shah	_____

TABLE OF CONTENT

CHAPTER 1
INTRODUCTION

1.1 INTRODUCTION

In this chapter we will provide brief introduction of the project and the report. It includes problem statement, goals and objectives and scope of studies. The chapter provides the information about the feasibility of the project. In the following chapter along with the proposed system's features, the software and hardware requirement are also discussed. The tool and technologies used for the system also discussed at the end of the chapter.

1.2 MOTIVATION

Many people those are suffering from blood pressure or sugar level diseases have to take medicines on regular basis according to given precautions but they face difficulty in remembering the time of medicine and forget to take those medicines on time which cause effect on their health and level of diseases does not remains balanced and get lowers or higher. Medical Assistant application will help patients through various digital features to remember medicine timings and appointment dates also patient can use chatbot to know use of a medicine which will help them to confirm whether they are taking the right pill or not. To know about the progress of the treatment, system will generate weekly reports of every medicine to check how much medicines patient has taken on time.

1.3 PROBLEM STATEMENT

Remembering the exact time of taking exact prescribed medicines can be very challenging for some especially for those who don't have anyone to look after them. And in the times of current pandemic, this issue seems to be a common problem for many people. So, there should be some sort of assistant for these patients who can remind them to take proper medicines and right on time. There are some formal ways like through an alarm clock app or hiring a care taker. But these methods either don't fulfill the needs of a patient or can be very challenging. To overcome these problem, we have proposed a system named as "Medical Assistant".

Most of the times, the hand writing of the doctor is not clear or understandable to the normal human which makes it very difficult to see the time of the prescribed medicine. To overcome that problem, we have included an OCR text reader in our system. Sometimes people need to just identify the use of a medicine just by entering the medicine name. For that, we have

introduced a chat bot that would give disease name by just putting medicine name on the system.

1.4 MEDICAL ASSISTANT APPLICATION

There are some systems regarding Medicine reminder but they lack many important features like they do not provide Chatbot feature to identify use of a medicine by entering medicine name. They do not provide the feature of OCR and QR scanning that help patient in various ways. This system generates weekly report that can help patient and doctor to analyze progress of treatment.

Patients are increasing day by day and by having disease doctor prescribe patient a lot of medicines to get recover from disease and patient do not remember to take medicine at right time and the purpose of the medicine. Some medicines have different milligrams and by taking heavy dose medicine can also get patient in serious condition. So, to overcome these kinds of problems and situation our application can help them. Patient take medicines regularly at given time which is prescribed by the doctor and still they are not feeling better than they can search the nearby doctors and can save their appointment meeting timings and doctor details so they can contact with him by using our application.

1.5 GOALS AND OBJECTIVES

Medical Assistant system helps the patient to check on their medicine, prescriptions and their daily routine of medicines. If they do not know about the medicine knowledge and purpose of the medicine then he can scan the medicine and gets the information of the medicine from the application. The goal is to provide an easy manageable system to the patients to take care of their health. The application can also help the patient to analyze progress of the treatment where system will generate weekly reports of every medicine to check whether the patient has taken all medicines on time or not, so that patient can share that report with their doctor to analyze progress of treatment. Also, patient can add appointment timing of the doctors on the system to get notification reminder.

1.6 DEVELOPED SYSTEM FEATURES

Features of the developed system are as follows:

1.0.1 Alarm Reminder:

The system sets alarm for the provided medicine details.

1.0.2 Add Prescription:

The system saves prescription details entered by patient.

1.0.3 OCR Scanning:

The system will scan prescription image and will convert it into a digital image by recognizing the characters.

1.0.4 QR Scanning:

The system will scan QR code of any medicine to show medicine details.

1.0.5 Generate Report:

The system will generate a weekly or monthly report of medicines added to analyze progress of the treatment.

1.0.6 Add doctor details:

The system will save the details of the doctors entered by patient.

1.0.7 Add an Appointment:

The system will set appointment reminder of the provided doctor's meeting.

1.0.8 Medicine Checker:

The system check medicine usage and details through Chatbot.

1.0.9 Feedback:

The system will allow user to enter feedback and suggestions for the system.

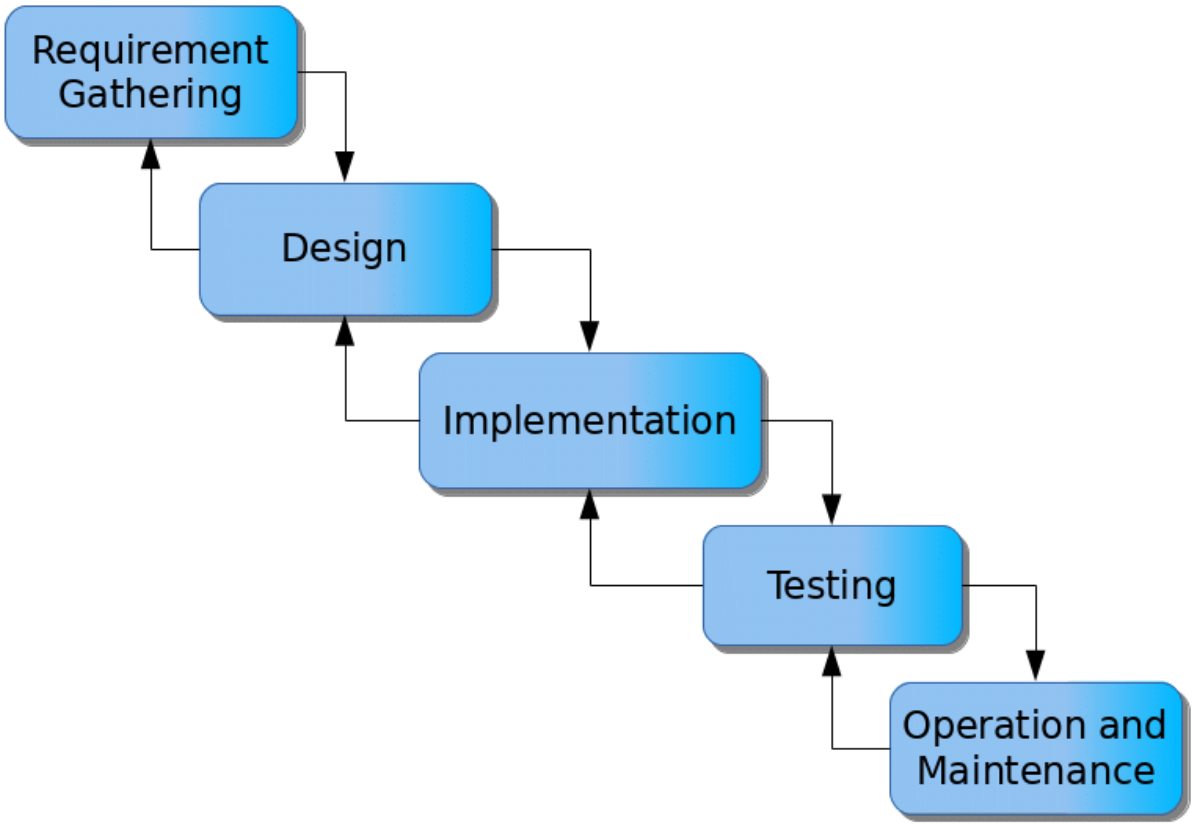
1.7 SCOPE OF THE STUDY

The Medical Assistant system is to facilitate patients in various ways. The target of the application is to help the patients to maintain about the medicine information and purpose of the medicine so that he can scan the medicine for its purpose and take the particular medicine to get rid of the pain and if the person has to take medicine on regular basis then he can also get the alarm to take the medicine on regular basis. The aim is to help patient to keep track of

their medicines. It will help them in both when to take their medicine and which medicine is for which disease or what purpose it serves. The system has a feature of Medicine Checker where patient will enter medicine name and system will show what is the purpose of that medicine and how they can use it. The system will also generate weekly reports of every medicine to check whether the patient has taken all medicines on time or not, this will help patient to know how much punctual he is in taking medicines.

1.8 PROCESS MODEL

The different software companies choose the kind of development model which suits to their system. The modified water fall model is followed for the development of the system. The modified water fall model is considered as traditional approach for describing software development life cycle, the modified water fall model is a linear approach for development of software products and by this approach user also stay touch with the system that is under development and enabled to understand what is actually happening.



The modified water fall model is followed because:

- Comprehensive and functional.
- Handled simply as model is rigid.
- Saves effort and resources.
- Provide easy way for analysis and testing**.**

1.9 NATURE OF THE PROJECT

This project is a android based application. Machine learning is also involved to train the data and to check medicine usage and details through AI.

1.10 OVERVIEW OF THE REPORT

The target of the application is to help the patients to maintain about the medicine information and purpose of the medicine so that he can scan the medicine for its purpose and take the particular medicine to get rid of the pain and if the person has to take medicine on regular basis then he can also get the alarm to take the medicine on regular basis.

The chapter is about to the problem statement, introduction, developed system purpose, developed system functionalities, possibility of the developed system, what resources software's and hardware are required. The approaches and tools that are used to develop this app are discussed in the following chapter. The second chapter is about the reason behind the idea. The chapter describes existing system and their functionalities, and the about the flaws in developed system, its assessment, comparison with other existing system and its limitation are also conferred.

CHAPTER 2

BACKGROUND AND EXISTING WORK

2.1 INTRODUCTION

In this chapter we will discuss a synopsis of present systems. The detail analysis of My pill, My Therapy and Take your pill is given with their pros and cons are enlisted.

2.2 IMPORTANT CONSTRUCTS OF THE APPLICATION

Following are some of the important constructs of the system.

2.1.1 Machine Learning

In this domain the data is trained and image processing is done. Emgu.CV library is used. This library called open CV usually used for image processing and is used to detect the human face from an image and crop that face. The HaarCascade Frontal Face framework is used. Haarcascade is a framework that classifies the frontal face of human. OpenCV cascade classifier uses this framework to classify frontal face from an image. Dlib library is used. Dlib is a machine learning algorithm written in C++ is used to get the facial landmarks using given shape predictors i.e. 68 landmarks shape predictor.

2.1.2 Android App Development

For the GUI the .NET framework is used. The main IDE is designed in .NET framework using c#. In this domain the distributable desktop applications for Windows, Mac and Linux.

2.1.3 Mongo dB Database

MongoDB is an open source NoSQL database management program. NoSQL is used as an alternative to traditional relational databases. NoSQL databases are quite useful for working with large sets of distributed data. MongoDB is a tool that can manage document-oriented information, store or retrieve information.

2.3 EXISTING SYSTEMS AND THEIR LIMITATIONS

There are many systems that provides the Medicine tracking facilities but they have a lot of restrictions and flaws. Those systems are discussed below:

2.1.4 Take your pill

Take your pill is one of the system that provides medicine tracking facilities. Take your pill provides simple functionality like medicine alarm reminder and checking history.

2.1.4.1 Features

- Medicine alarm reminder
- History Report.
- Feedback.

2.1.5 My Therapy

My Therapy have friendly interface and it is easily used by anyone. My Therapy provides functionalities like Medicine alarm reminder, add appointment, add healthcare and progress report.

2.1.5.1 Features

- Medicine alarm reminder
- Progress Report
- Add appointment
- Add healthcare professional
- Add pharmacy

2.1.6 Med Control

Med Control also have user friendly interface and it is easily used by anyone. Med Control provides a basic functionality of Medicine alarm reminder.

2.1.6.1 Features

- Medicine alarm reminder
- Feedback

2.4 LIMITATIONS OF THE EXISTING SYSTEMS

The limitation of the current systems are as follows:

2.1.7 Limitations of Take your pill

This application has basic functionalities. It cannot add appointment meetings and healthcare professionals for future use. This application cannot Scan OCR to convert prescription image into digital text. It does not provide medicine usage and details.

2.1.8 Limitations of My Therapy

This application cannot Scan OCR to convert prescription image into digital text. It does not provide medicine usage and details by scanning QR scanner. It does not allow user to send feedback and suggest any improvements for the system.

2.1.9 Limitations of Med Control

This application only allows user to add medicine details to set alarm reminder. It does not provide facility of generating progress report. It cannot add appointment meetings and healthcare professionals for future use. This application cannot Scan OCR to convert prescription image into digital text. It does not provide medicine usage and details.

2.5 COMPARISON OF EXISTING SYSTEMS AND PROPOSED SYSTEM

Features	Take your pills	My pills	My therapy	Med control	Our system
QR scanner	no	no	no	no	yes
OCR scanner	no	no	no	no	yes
Analytics report	yes	yes	yes	no	yes
Add prescription	yes	no	no	yes	yes
Review/Feedback	yes	no	no	yes	yes

Add Healthcare Professional	no	yes	yes	no	yes
Appointment reminder	no	no	yes	no	yes
Medicine Checker	no	no	no	no	yes
Alarm Reminder	yes	yes	yes	yes	yes

2.6 FEATURES

Table 2.1 describe the comparison between proposed system and existing system functionalities. The proposed system covers all the features while the existing system fails to perform all the features.

2.7 SUMMARY

In this chapter the important constructs of application domain that includes machine learning, android application development, client-server architecture and mongo Db database are explained. The existing systems which includes My Therapy, my pills, and Med Control are explained. The features of existing system are explained. And table 2.1 describes similarities and dissimilarities between the present systems.

CHAPTER 3
REQUIREMENTS SPECIFICATION

3.0 Introduction

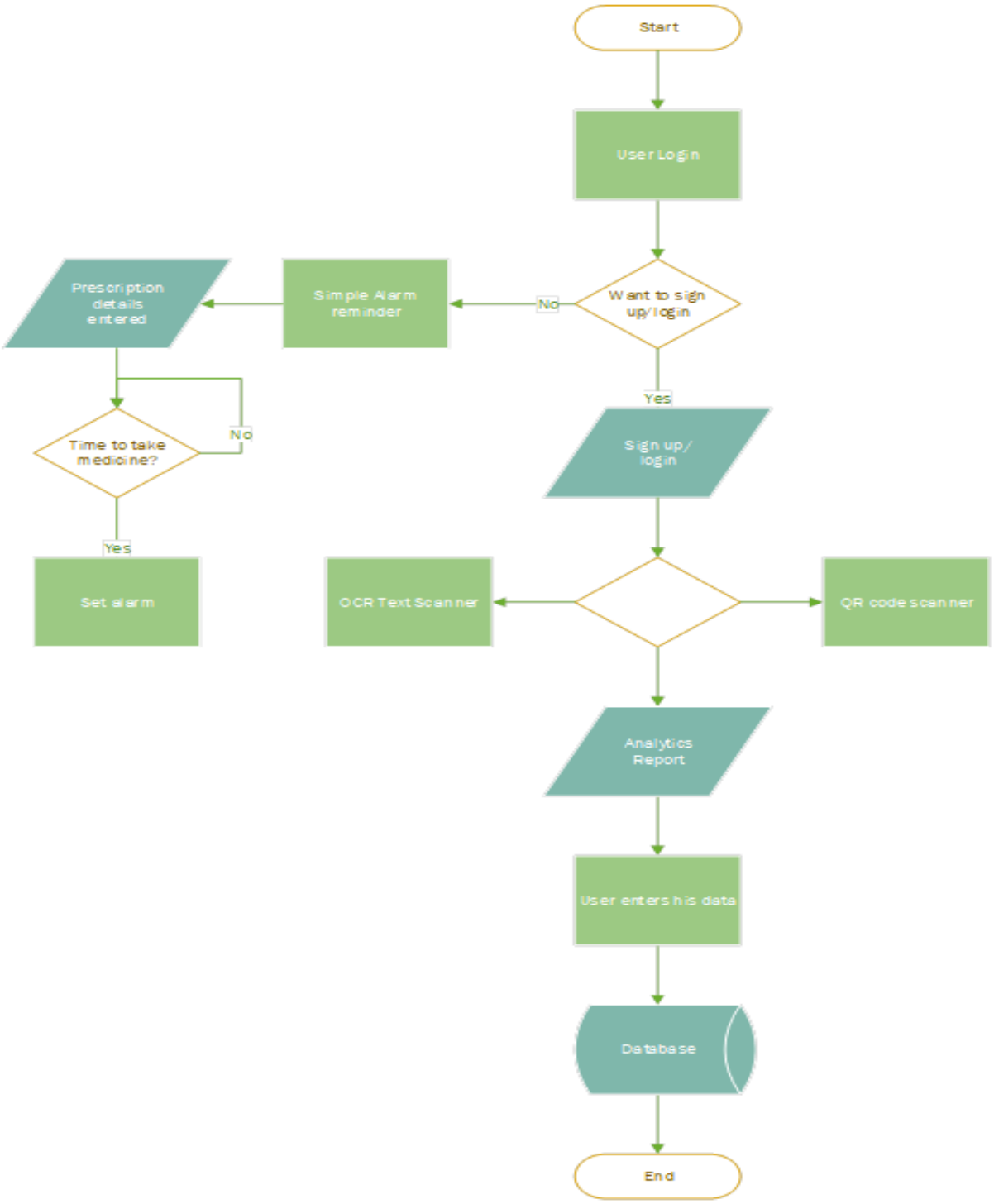
System modelling is the process of designing different models of a system in order to predict various perspectives or views of that system. In this chapter, system modelling is done by using a graphical notation based on Unified Modelling Language (UML). This will enable us to design a use case diagram.

3.1 Developed System Flow

The system flow has been thoroughly discussed in the system flow diagram. The system flow diagram depicts what the system is capable of and how it will be executed. All of the software's features are divided into sub-features and similar features are categorized together to make differentiating between them simpler and more comprehensible. Each feature of the system is explained separately to lower the complexity of the system and to heighten the understandability of the system.

The system starts by login in through email and password required. If the user is new then first of all the registration is done by sign up to the system. After logging in the system will display add medication screen where user can add medicine details to set alarm for that medicine accordingly. User can use add prescription and add doctor option where system will save prescription details entered by patient into the database.

There is an option of OCR on the main screen where system will scan prescription image selected by user and will convert it into a digital image by recognizing the characters. There is an option of QR scanner on the side bar of the main screen where user can scan QR code of the medicine to get medicine details. There is an option of Report where system will generate a weekly or monthly report of medicines added to analyze progress of the treatment.



3. 1 System Flow Diagram

3.2 Interface Requirements

The main interface is designed using React, React-native API's and JavaScript. The machine learning algorithm are coded in python which work as a server.

3.2.1 Hardware Interface Requirements

Hardware	**Requirement**
CPU	Intel core i5 5th gen or higher
Languages	JavaScript, Python
Database	Mongo DB
Mobile Devices	Android supported
RAM	8GB or above
OS	Windows
Compilers	Visual Studio Code
Libraries	React, React-native, Json

3.2.2 Software Interface Requirements

The required software interface requirements for the system are:

3.2.2.1 Visual Studio code

Microsoft's Visual Studio is an integrated development environment (IDE) that can be used to develop a wide range of applications, including graphical user interfaces (GUIs), console apps, web apps, mobile apps, cloud services, and web services. It supports managed and native code development and can be used with various programming languages such as C#, C++, VB (Visual Basic), Python, JavaScript, and other languages.

3.2.2.2 Mongo DB

MongoDB is an open source NoSQL database management program. NoSQL is used as an alternative to traditional relational databases. NoSQL databases are quite useful for working with large sets of distributed data. MongoDB is a tool that can manage document-oriented information, store or retrieve information.

3.2.2.3 Python

Python is a powerful, versatile programming language that is easy to read and write. Developed in 1989 by Guido van Rossum, Python has become one of the most popular languages among developers due to its easy syntax and dynamic semantics. Python also supports modules and packages, which encourages code reuse and modularity. With Python, you can create a wide variety of applications, including GUI applications, web applications, mobile apps server-side code, artificial intelligence (AI) algorithms, and machine learning algorithms.

3.3 Functional Requirements

The Basic functionalities of the system are:

System shall set the medicine alarm reminder. System shall allow to scan OCR of the prescription and convert it into digital text. System must be able to Scan QR code of the medicine and show results. Application should be able to generate weekly report of the medicines added for reminder. System shall set appointment meeting reminder. System should be able to add doctor details and prescription details in database.

3.4 Use Case

A use case is the communication between an external actor and the system. It explains how a user interacts with the system and what actions are taken. A use case is a collection of specified sequences that cover every possible state of software for passing along implementation control. This results in all potential actions that a system could take.

In order to write a use case, we must adhere to the steps listed below: Name, Brief Description, Dependency, Actors, Preconditions, Basic Flow, Alternate Flows, Exception Flows and Post Conditions. The communication between external actors and the system is explained by a use case in order to achieve a goal. It's important to note that actors don't necessarily have to be human - an actor can be anything that asks the system for a certain action.

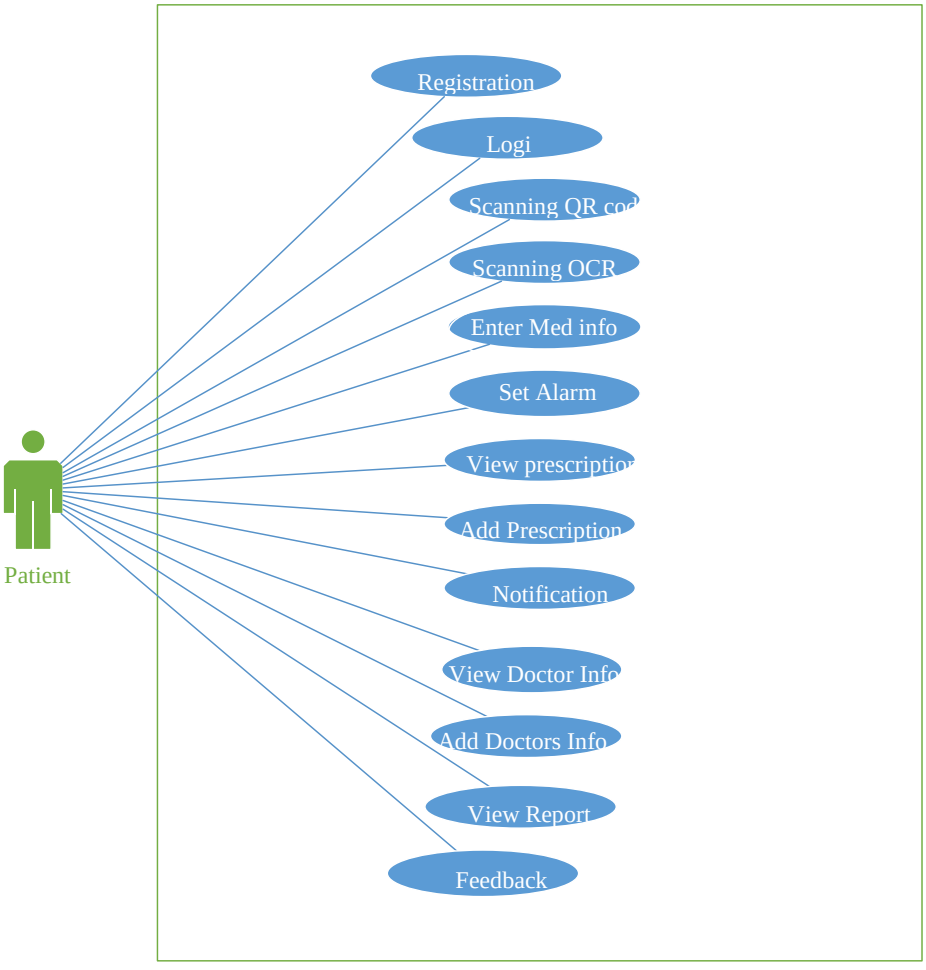
Use case analysis usually starts by demonstrating use case diagrams. For agile development, UML requirement model for many diagram showing use cases and some textual explanations, notes or use case briefs would be very insignificant and just enough for little or simple project use. As good complements to use case texts, the visual diagram representations of use cases are also effective facilitating tools for the better comprehension, communication and design of complex system behavioral requirements. Less defined use case detail tend to be more like business processes. In other words, when it comes to agile development, all you really need is

a few diagrams and a few sentences to explain each one - anything more than that would be overkill and not necessary for such a project.

3.5 Use Case Diagram

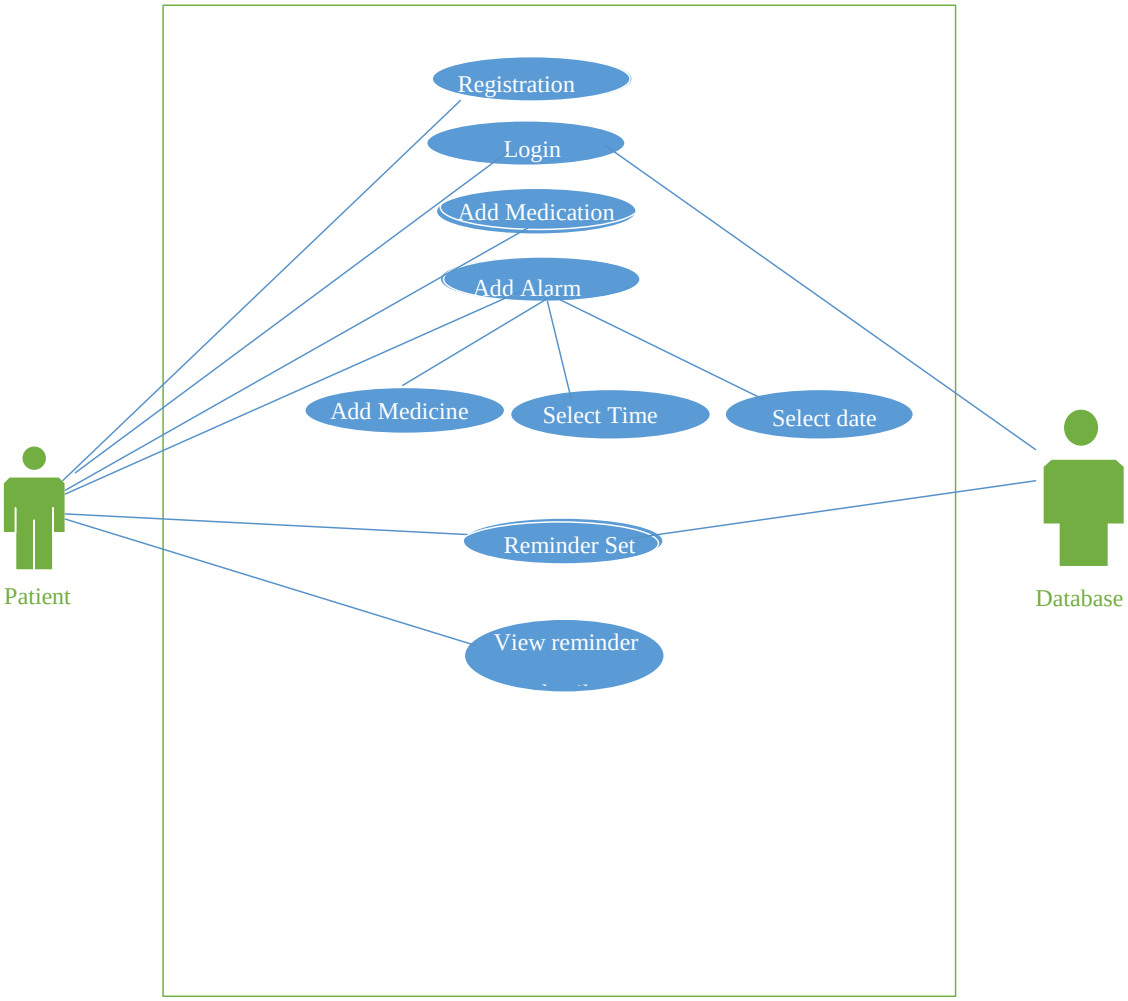
Following are the use case diagrams which shows the communication of external actors with the system

3.5.1 System Use Case Diagram



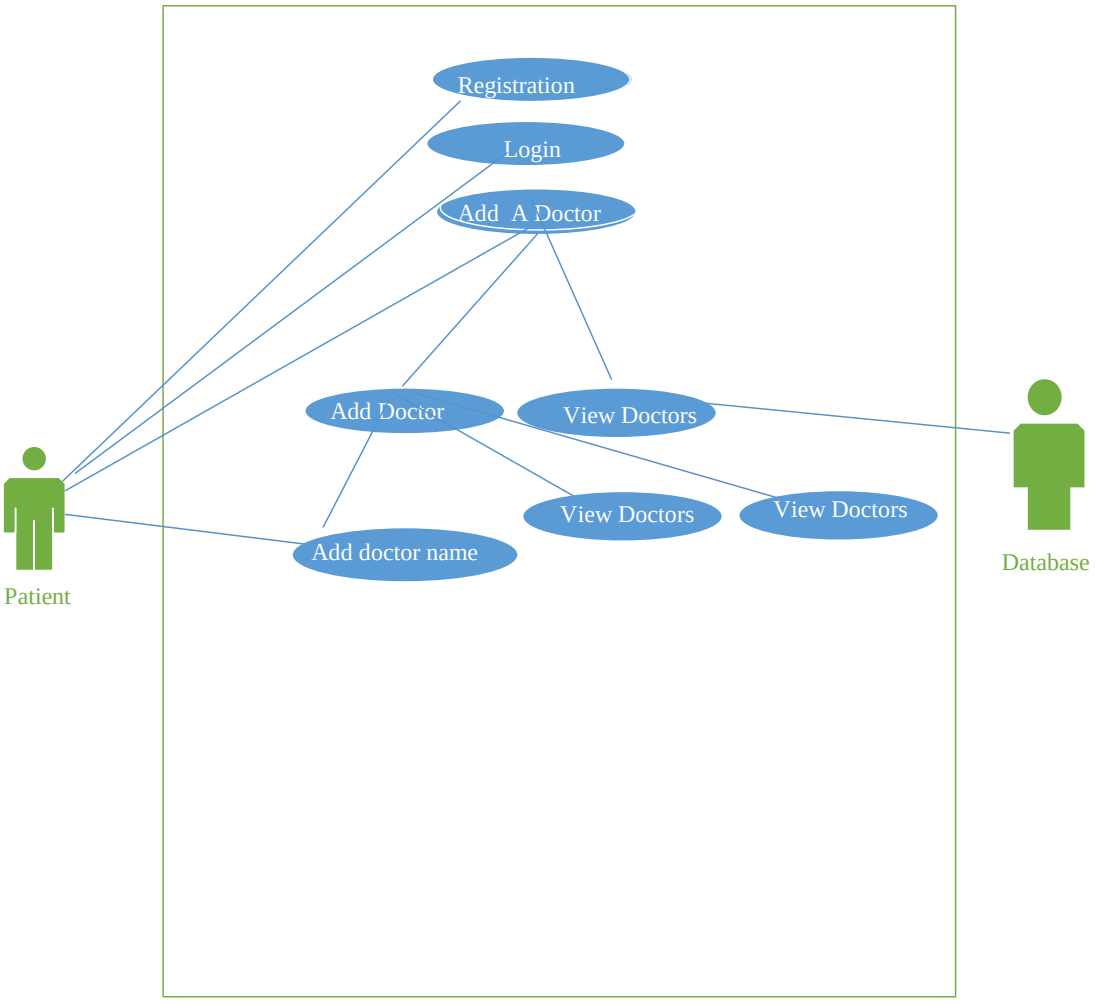
3. 1 System Use Case Diagram

3.5.2 Reminder set Use Case Diagram



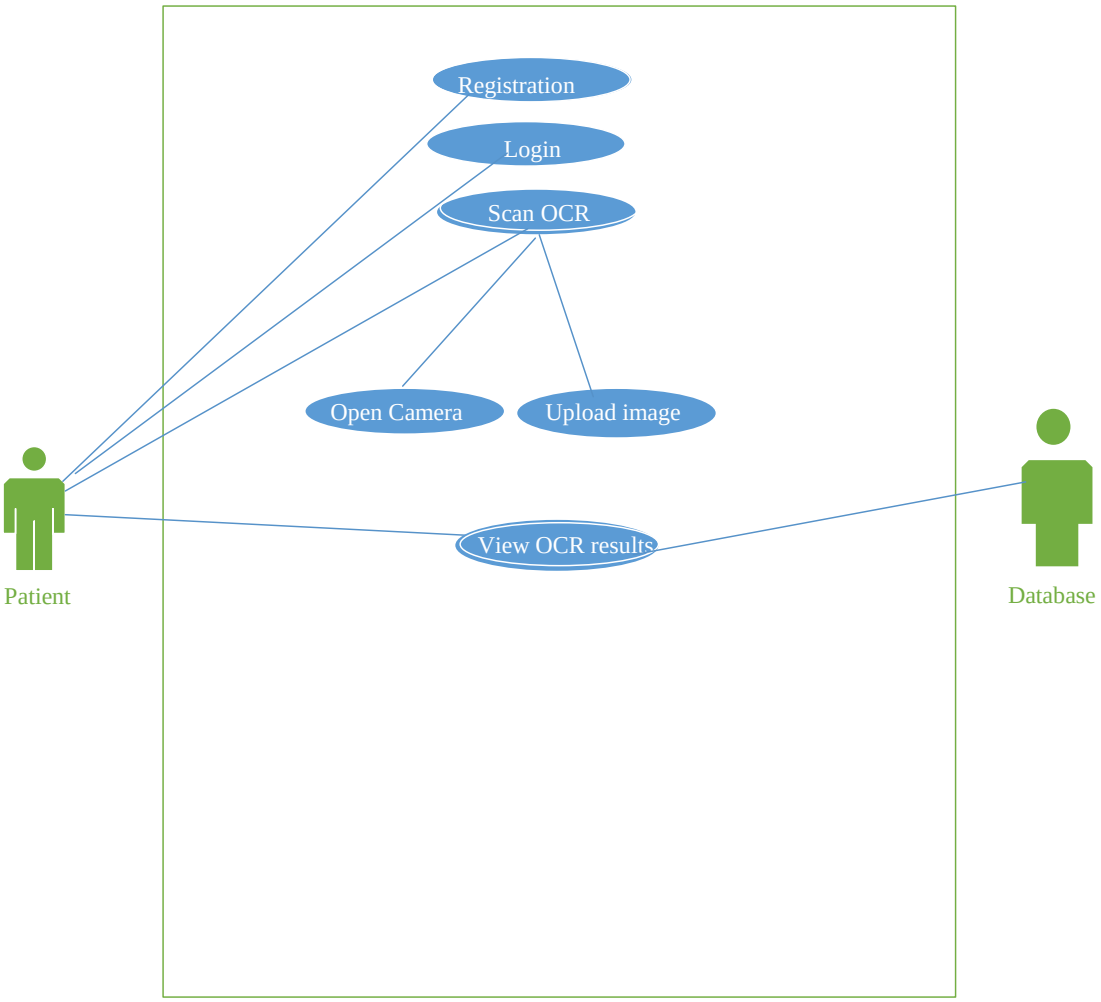
3. 2 Reminder Set Use Case Diagram

3.5.3 **Add Doctor Use Case Diagram**



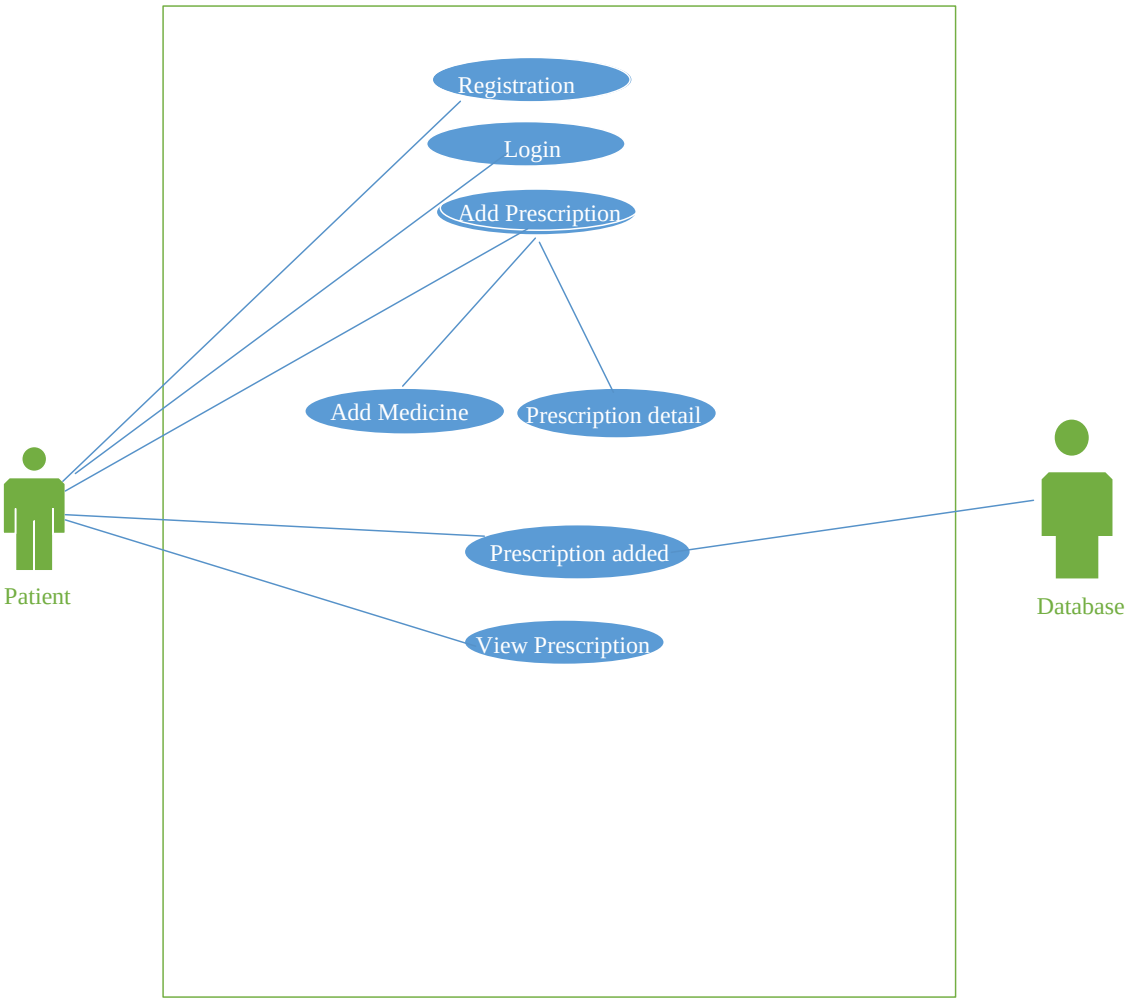
3. 5 Add Doctor Use Case Diagram

3.5.4 **OCR Use Case Diagram**



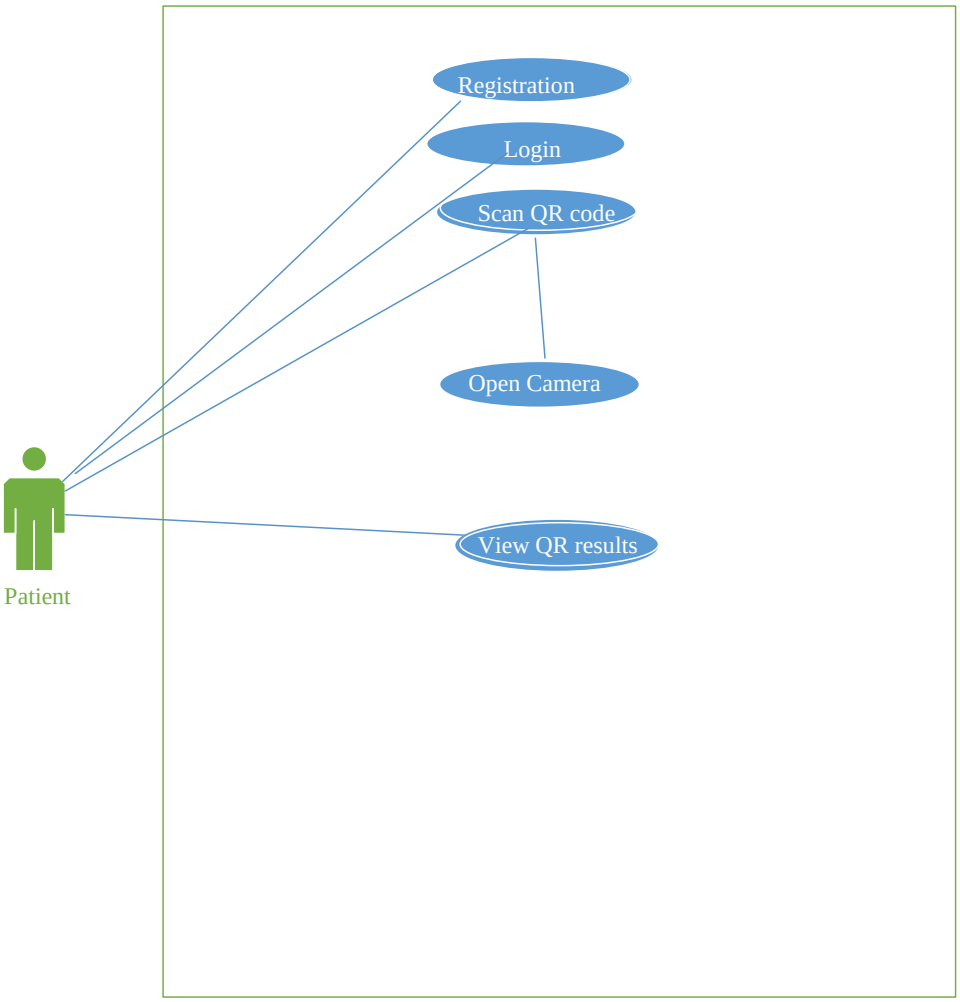
3.6 OCR Use Case Diagram

3.5.5 **Add prescription Use Case Diagram**



3.7 OCR Use Case Diagram

3.5.6 **QR scanner Use Case Diagram**



3.8 QR Use Case Diagram

3.5.7 **Registration Use Case**

Use Case ID:	UC-01
Use Case Name:	Registration
Actors:	Patient
Description:	Patient will register him/herself to create the profile.
Trigger:	User click the “Register” Button.
Preconditions:	PRE-1. User must have an android device. PRE-2. User must be connected to internet. PRE-3. Application should be installed in device.
Postconditions:	POST-1. User is logged in the system. POST-2. User Profile is created. POST-3. Home page is displayed to the user.

Normal Flow:	1. User will open the application, after installing it in device. 2. User will be asked to register him/herself in the system. 3. User will enter a unique username. 4. User will create a password. 5. User click the Register Button. 6. System will verify username and password. 7. User Profile will be created after the verification. Home page is displayed.
Alternative Flows:	1. If username is not unique, system displays a message "Re-enter username". 2. If password is not verified, system displays a message "Re-enter your password". Normal flow continues from step1.
Exceptions:	None
Business Rules	None
Assumptions:	None

3.5.8 **Login Use Case**

Use Case ID:	UC-02
Use Case Name:	Login
Actors:	Patient
Description:	Patient will login to the system.
Trigger:	User click the "Login" Button.
Preconditions:	PRE-1. User Should be registered.
Postconditions:	POST-1. User is logged in the system.
	POST-2. Home page is Displayed.
Normal Flow:	1. User will open the application. 2. User will be asked to login the system. 3. User will enter username. 4. User will enter password. 5. User Click Login Button. 6. System will verify username and password. 7. User will be logged in, after the verification. 8. Home page is displayed.

Alternative Flows:	1. If username is not unique, system displays a message “Re-enter username”. 2. If password is not verified, system displays a message “Re-enter your password”. Normal flow continues from step1.
Exceptions:	None
Business Rules	None
Assumptions:	None

3.5.9 **Alarm Reminder Use Case**

Use Case ID:	UC-03
Use Case Name:	Alarm reminder
Actors:	Patient
Description:	User will select add medication option and system will display the page where patient will select date and time with medicine details and confirm details. System will set alarm reminder for that medicine and save details into the database
Goal:	Patient wants to add alarm reminder for a medicine.
Preconditions:	User should select future date and time.
Postconditions:	Reminder is successfully set for the medicine
Normal Flow:	1. User will open the application. 2. User will select add medication button. 3. System will ask for medicine and timing details. 4. User will enter medicine info and select date and time. 5. User will confirm details to set reminder. 6. System will add and save reminder details in database. 7. System will display added reminder details on the screen.
Alternative Flows:	3. If selected date is not of future, system displays a message “please choose future date”. 4. If user do not confirm medicine details, system will not add reminder”. Normal flow continues from step1.
Exceptions:	None
Business Rules	None
Assumptions:	None

3.5.10 **OCR Use Case**

Use Case ID:	UC-04
Use Case Name:	OCR
Actors:	Patient
Description:	User will select scan OCR option and system will display the options to open camera or to upload image. User will choose option to scan image. System will scan image and convert it into digital text and will save it into database.
Goal:	Patient wants to convert prescription image into digital text.
Preconditions:	User should scan clear image that is visible on the camera.
Postconditions:	Image is successfully converted into digital text.
Normal Flow:	1. User will open the application. 2. User will select OCR scan option. 3. System will ask for option to open camera or upload image. 4. User will scan image to convert it. 5. System will scan image and display results on the screen. 6. System will save result details in database.
Alternative Flows:	4. If selected image is not clear, system displays a message “please scan image again”. 5. If mobile camera is not working properly, system will not show any results.
Exceptions:	None
Business Rules	None
Assumptions:	None

3.5.11 **QR Scanner Use Case**

Use Case ID:	UC-05
Use Case Name:	QR scanner
Actors:	Patient

Description:	User will select scan QR code option and system will display the options to open camera. User will choose option to scan code. System will scan QR code and show results on the screen.
Goal:	Patient wants to scan QR code to get medicine details.
Preconditions:	User should scan code with clear and working camera.
Postconditions:	QR code is successfully scanned and results are displayed.
Normal Flow:	1. User will open the application. 2. User will select QR scan option. 3. System will ask for option to open camera. 4. User will scan code to convert it. 5. System will scan QR code and display results on the screen.
Alternative Flows:	5. If mobile camera is not working properly, system will not show any results.
Exceptions:	None
Business Rules	None
Assumptions:	None

3.5.12 **OCR Use Case**

Use Case ID:	UC-06
Use Case Name:	Report
Actors:	Patient
Description:	User will select Report option and system will display screen to add medicine details. User will add medicine details. System will save details in database and will prepare weekly report for that medicine.
Goal:	Patient wants to generate weekly progress report.
Preconditions:	User should add medicine details and reminder to get weekly report.
Postconditions:	Weekly progress report is in progress to be generated successfully.

Normal Flow:	1. User will open the application. 2. User will select report option. 3. System will ask for add medicine details and reminder. 4. User will add medicine details and reminder. 5. System will save details in database and report get in progress.
Alternative Flows:	3. If user does not save medicine details in report, system will not generate report.
Exceptions:	None
Business Rules	None
Assumptions:	None

3.6 Gant chart:

Features	2021-2022									
	Dec	Jan	Feb	Mar	Apr	May	Jun	Oct	Nov	Dec
Proposal Writing										
Requirement Analysis										
Black Box Testing for Interface										
Database Design										
Building Database										
Test Database										
Build System										
Integration of system										
System testing										
Black Box Testing of System										
Final Report										
Presentation										

CHAPTER 4
SYSTEM MODELING

4.0 Introduction

In this chapter the developed system's modeling is performed with the help of system design, design approach, interface design and 4+1 view model of architecture. The process of developing abstract models of a system is known as system modeling, with every model shows a different view or perspective of the system. System modeling here is to indicate this system using some kind of graphical notation, which is based on notations in the Unified Modeling Language (UML).

4.0.1 Top down Design Approach

In Top-Down design approach, all the requirements are gathered and listed, system is then divided into modules based on requirements. Top down design approach allows the structural control of the system and is used where the requirements are not properly available.

4.0.2 Bottom up Design Approach

Bottom up design approach requires the very basic knowledge of system components and modules and leads towards a product. In this approach, we have all the basic components of the system known, and we develop and integrate them which leads towards a product. In bottom up approach the low-level modules should be designed before moving to design the high modules of the system. The top down design approach is adopted to develop the system.

4.1 Architectural design

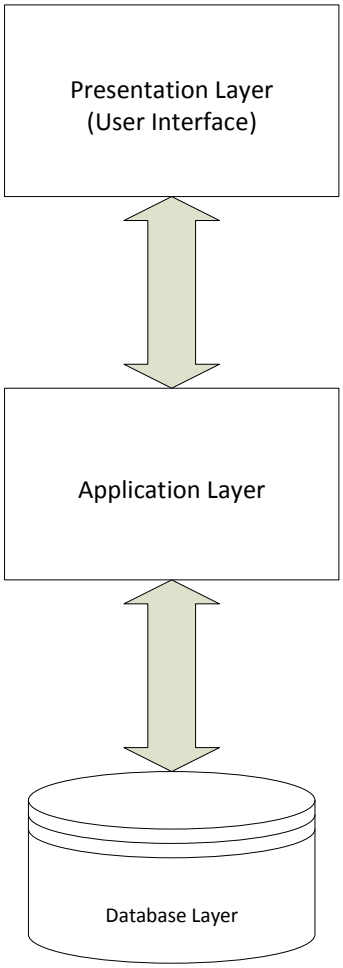
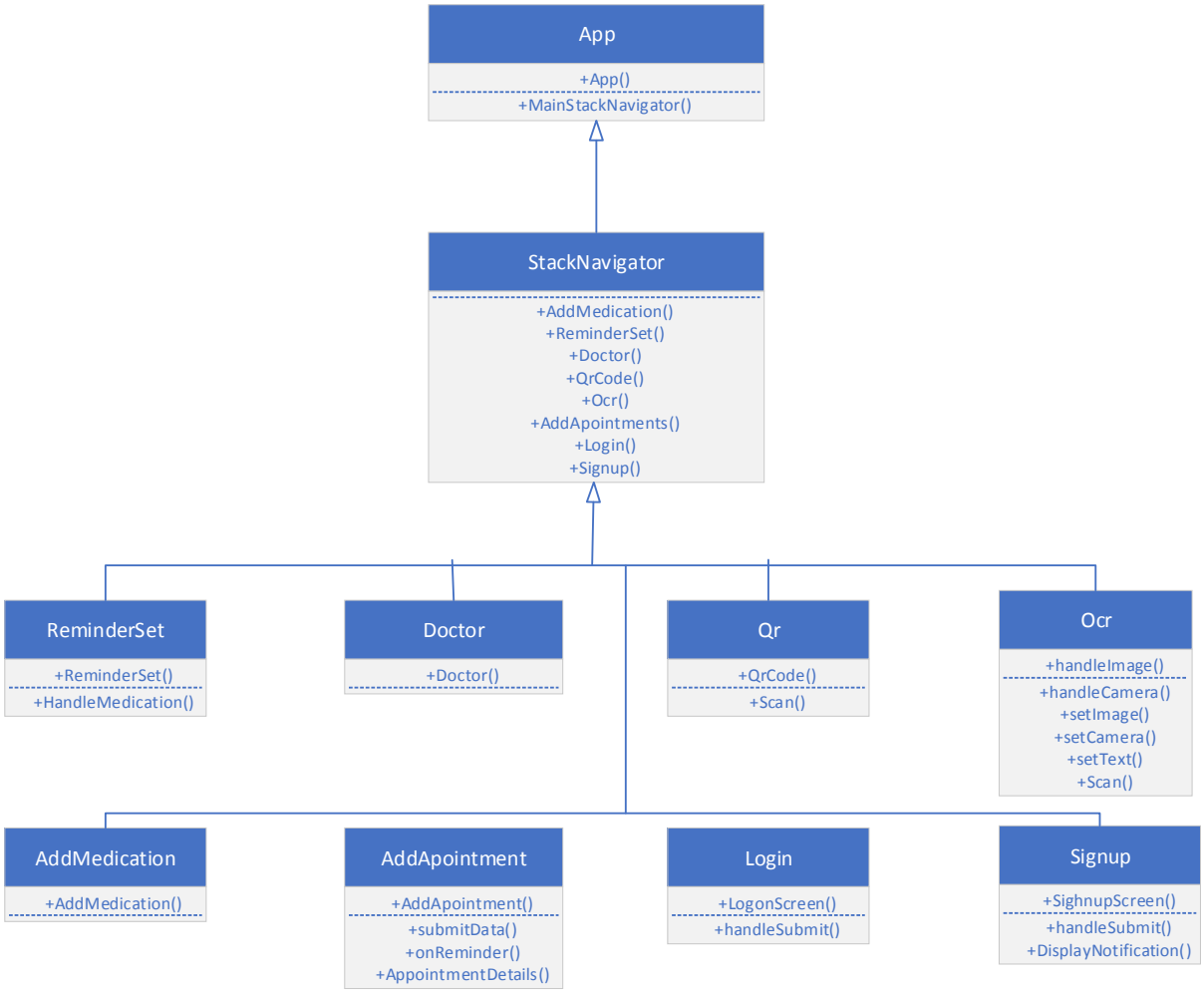


Figure 1: Architecture Diagram

4.2 Class Diagram:


App
+App()
+MainStackNavigator()
StackNavigator
+AddMedication()
+ReminderSet()
+Doctor()
+QrCode()
+Ocr()
+AddApointments()
+Login()
+Signup()
ReminderSet
+ReminderSet()
+HandleMedication()
Doctor
+Doctor()
Qr
+QrCode()
+Scan()
Ocr
+handleImage()
+handleCamera()
+setImage()
+setCamera()
+setText()
+Scan()
AddMedication
+AddMedication()
AddApointment
+AddApointment()
+submitData()
+onReminder()
+AppointmentDetails()
Login
+LogonScreen()
+handleSubmit()
Signup
+SighnupScreen()
+handleSubmit()
+DisplayNotification()

4.3 Process Flow Diagram:

Scan Medicine

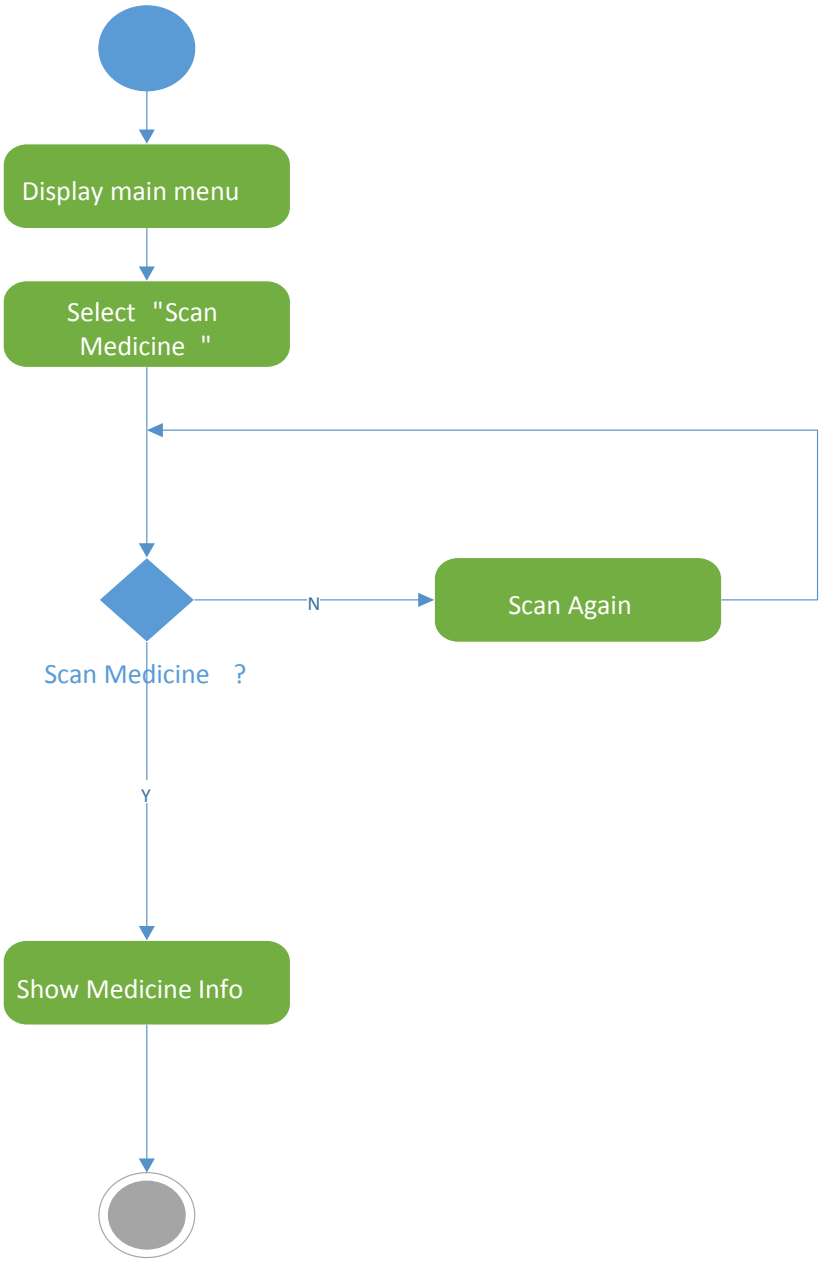
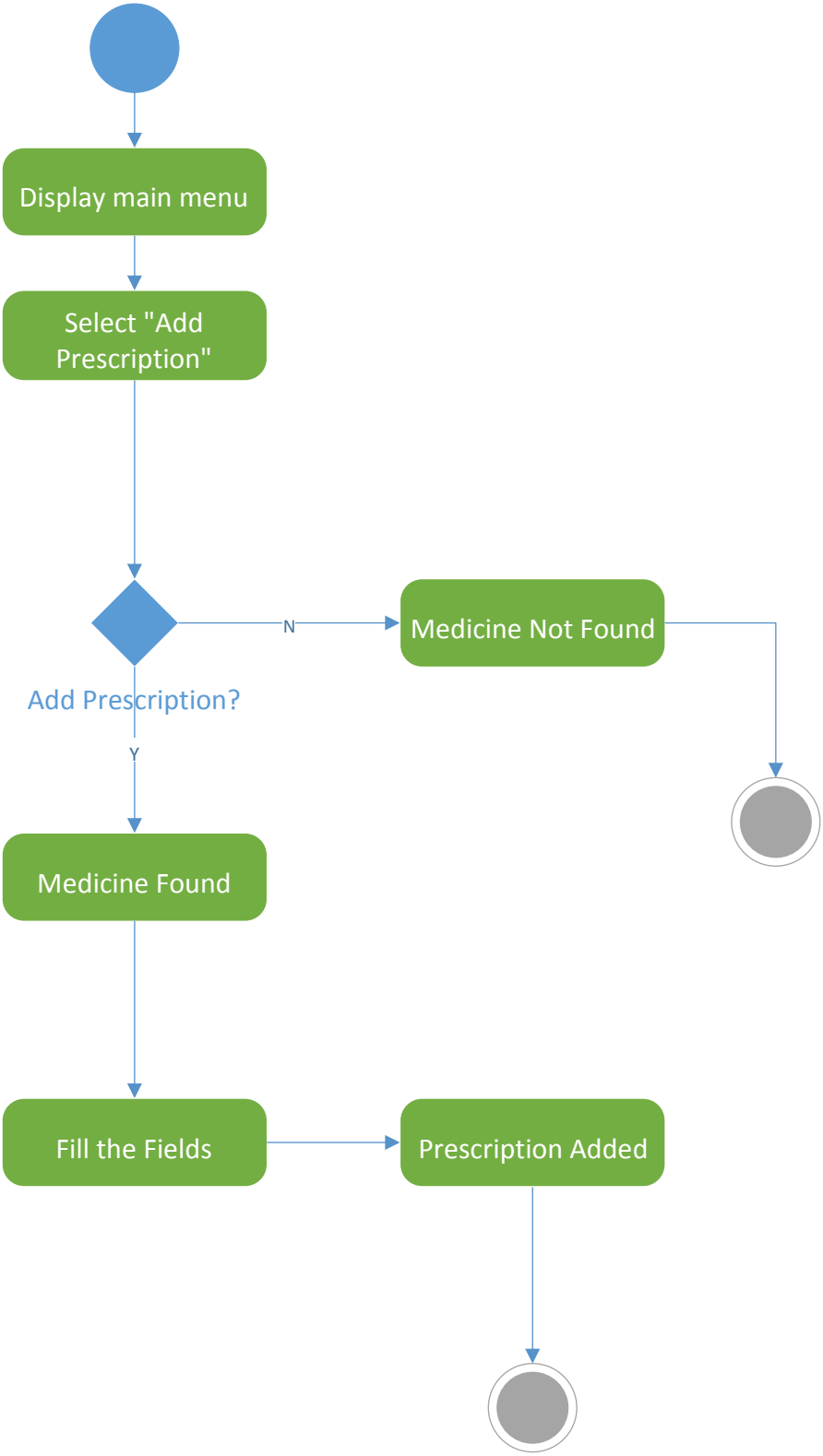


Figure 2: Activity Diagram for Scan Medicine

Add Prescription:



Set Alarm:

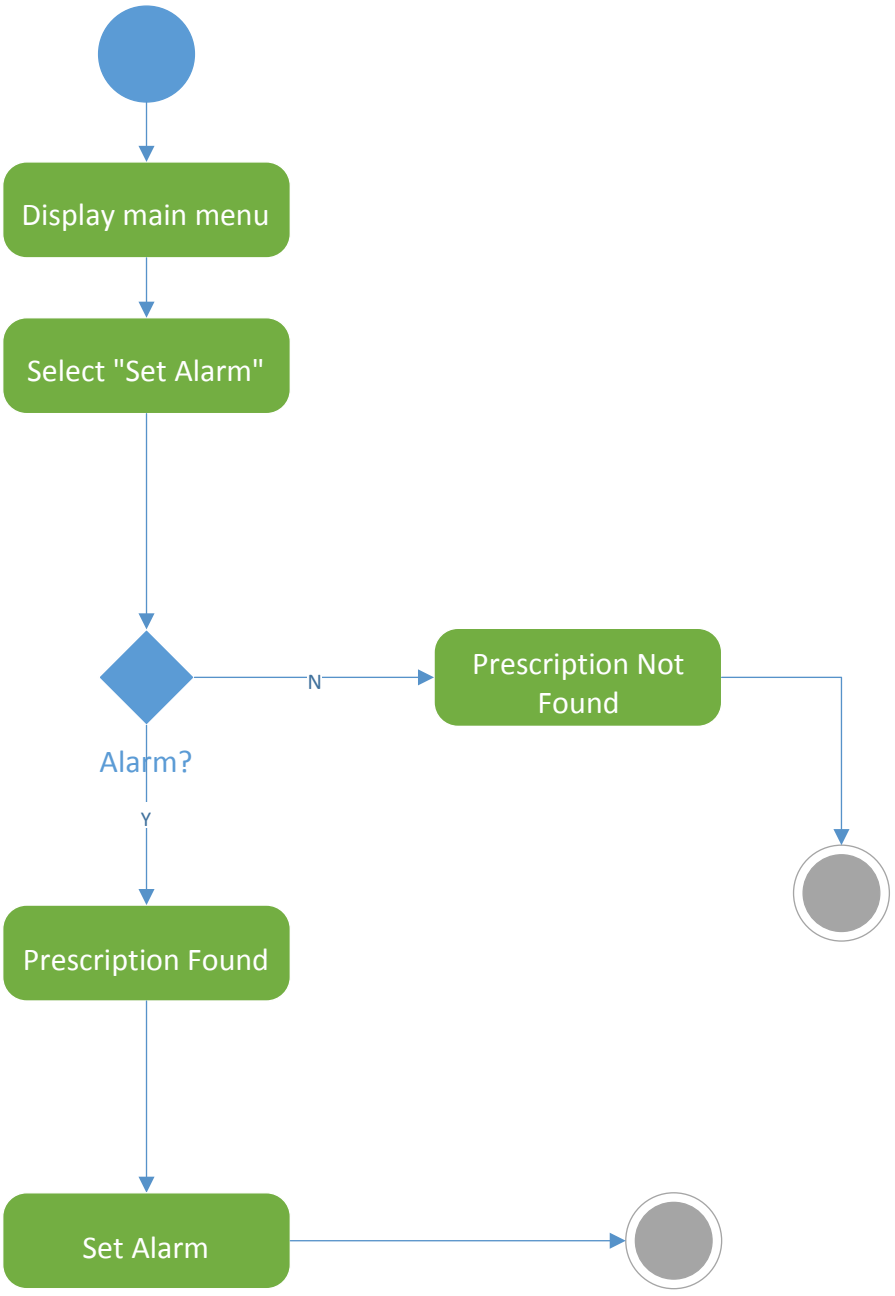


Figure 4: Activity Diagram for Set Alarm

Update Prescription:

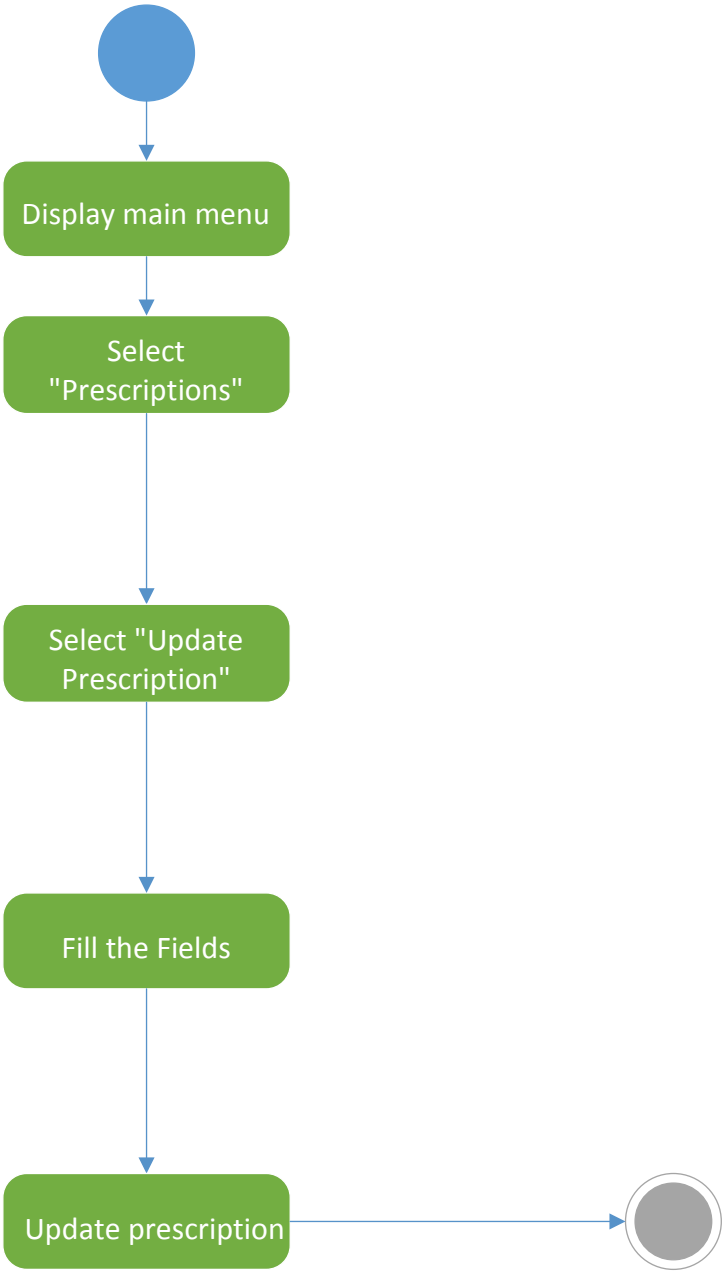


Figure 6: Activity Diagram for Update Prescription

4.4 Sequence Diagram:

Add Prescription:

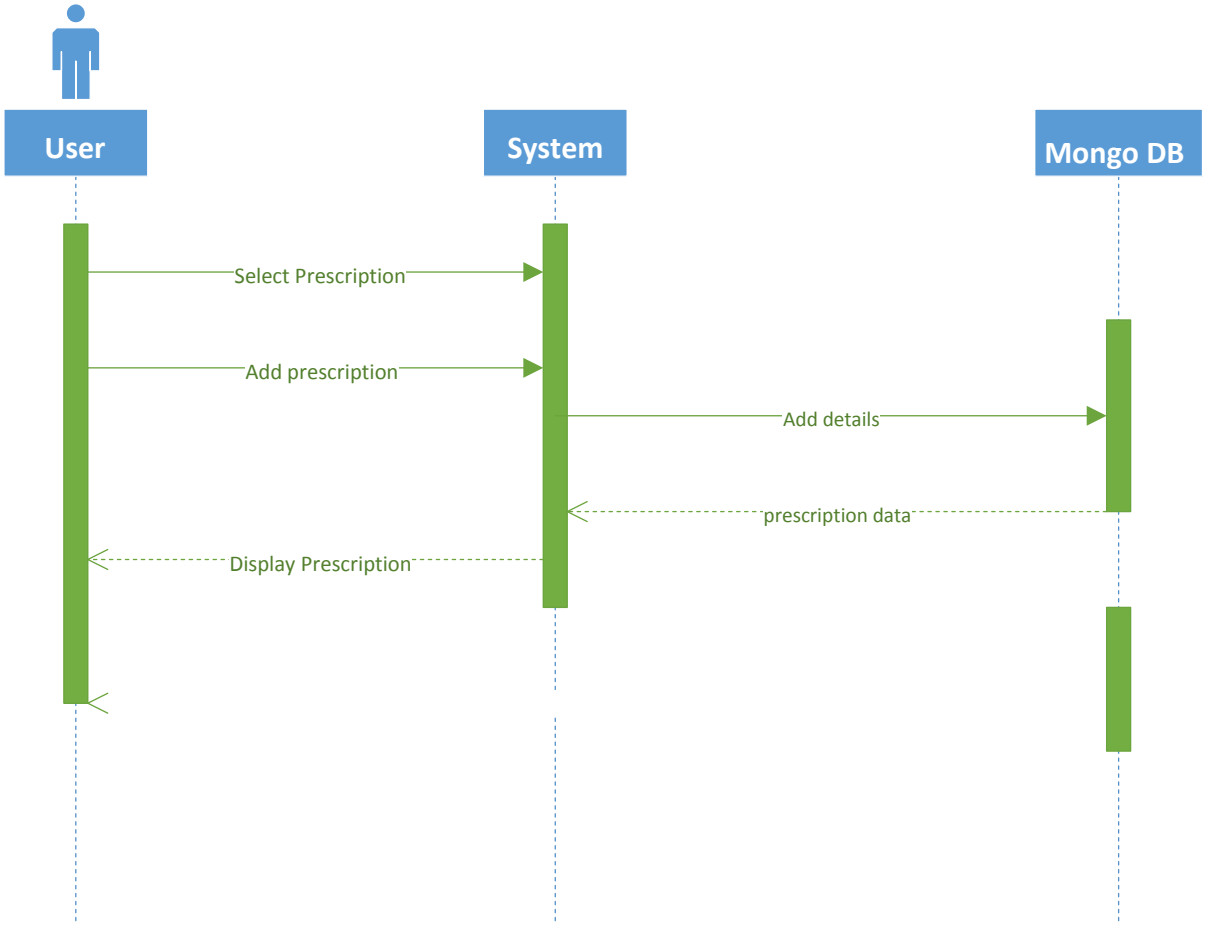


Figure 8: Sequence Diagram for Add Prescription

Set Alarm:

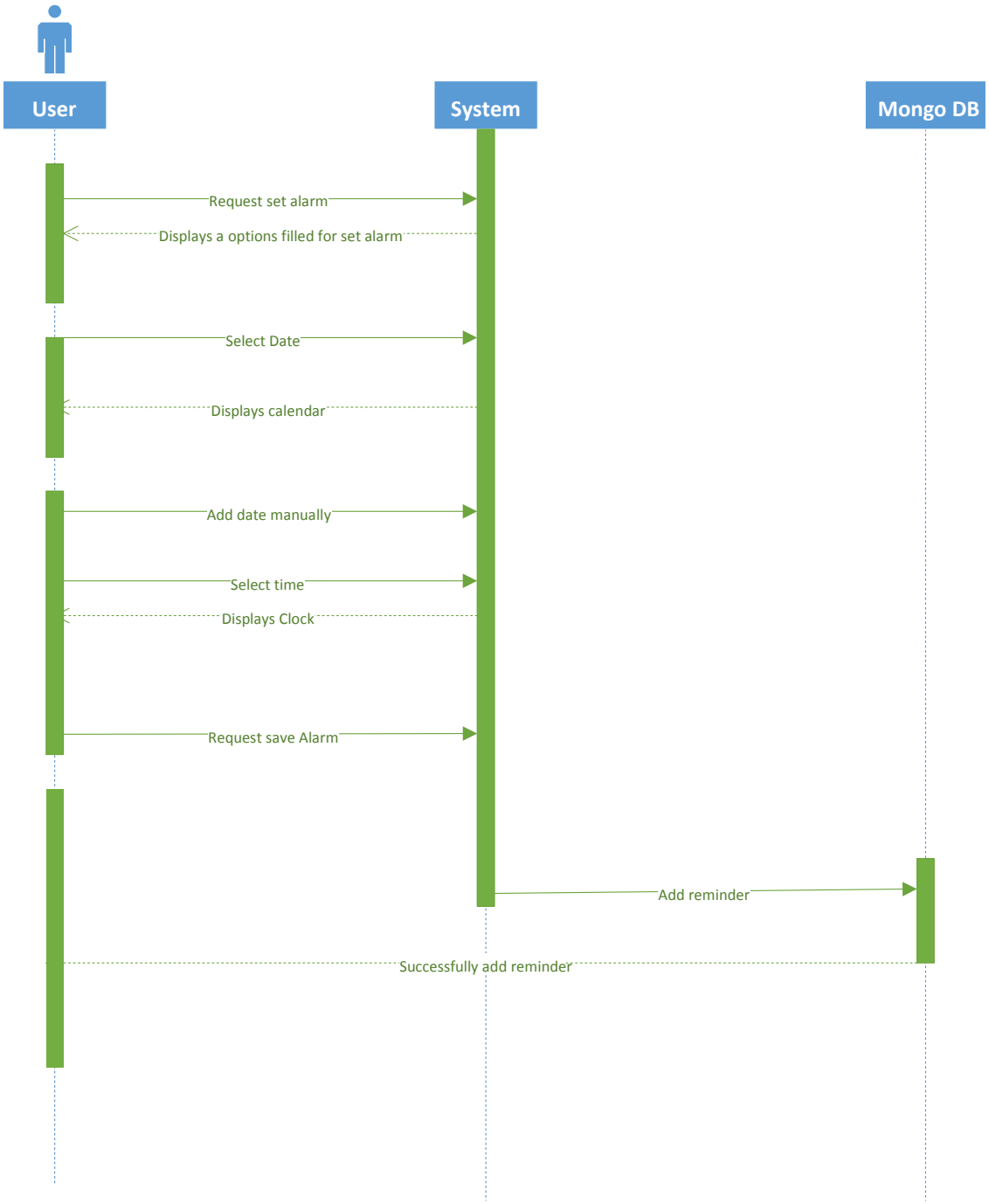


Figure 9: Sequence Diagram for Set Alarm

CHAPTER 5
IMPLEMENTATION

5.0 Introduction

The chapter gives a brief explanation of modules of this system. It includes libraries, frameworks and algorithms and their details that are used in each module. This chapter gives how modules are integrated with each other to perform the functionalities of the system. It briefly describes how the developed system is working and module wise integration with each other.

5.1 Modules

Following are some of the modules of the system which shows the workflow of the system.

5.0.1 Developed System Features

Features of the developed system are as follows:

5.0.2 Alarm Reminder

The system sets alarm for the provided medicine details.

5.0.3 Add Prescription

The system saves prescription details entered by patient.

5.0.4 OCR Scanning

The system will scan prescription image and will convert it into a digital image by recognizing the characters.

5.0.5 QR Scanning

The system will scan QR code of any medicine to show medicine details.

5.0.6 Generate Report

The system will generate a weekly or monthly report of medicines added

to analyze progress of the treatment.

5.0.7 Add doctor details

The system will save the details of the doctors entered by patient.

5.0.8 Add an Appointment

The system will set appointment reminder of the provided doctor’s meeting.

5.0.9 Medicine Checker

The system check medicine usage and details through Chatbot.

5.0.10 Feedback

The system will allow user to enter feedback and suggestions for the system.

CHAPTER 6
SYSTEM TESTING

6.0 INTRODUCTION

This chapter provides the overview of the testing techniques that are used to test the system. In this chapter many testing methodologies are discussed. The main functionality of the system are tested by developing test cases. Black box, unit and integration testing are explained and is used to test the functionality and performance. Each and every test case is explained with the help of table.

6.1 TEST METHODOLOGY

The testing is performed to checks that system functionalities works accurately or not. All features of the application are tested in software testing. The application's response time to the user's input is observed. The output that is generated by the system against every input is analyzed. Each feature of the system is tested in real world environment. Functionalities are checked by analyzing the output of the system against user different inputs. Test cases are generated for each and every single feature of application to test that weather system satisfies all the requirements or not. Several testing techniques used to perform testing for example unit testing, white box, black box, integration testing and system testing. In black box testing the internal mechanism or structure of the system is unknown. The user/tester is not familiar with the internal working of the system. Black box testing is performed to test the system. In white box testing the internal mechanism or organization of the system is known. Unit testing is performed when the system contains module in separate form to test whether each module of system developed is working well or not according to design. When all the modules are integrated then the system capture any error while performing integration testing. Test cases are tested by giving several inputs and commands to every part/module/component of system so that the whole system can be tested. For testing of system PC (personal computer) is used to test the application. Testing confirms that all the features are working properly. During testing bugs/failures/defects are identified. These bugs are removed by making some changes and improvements in system. Selenium tool is used for automated testing.

6.2 TEST BED

The process of testing of specific module of software like function, class or object in a specific method is called bed. In test bed software, operating system and hardware

are required to test the complete Application. In this system PC (personal computer) is needed to check and run the application which runs on Window 8.1 or higher.

6.3 SYSTEM TEST CASE

Testing ensures that all the requirements are working properly. During testing bugs/failures/defects are identified. These bus/errors are removed later.

6.4 TEST CASES

Testing is fundamentally collection of certain actions applied on system to check that whether the system is performing tasks accurately or not according to the requirements. After performing the testing, the software tester choose which test case is passed and which test case is failed. The testing of every module is performed separately. Every test case is specifically generated for a certain situation. Test cases contain different characteristics like test case ID, QA test engineer, and use cases reference, name of personnel, revision history, test date, objective, assumptions, and pre-conditions, testing environment, steps and the rank of test case. The test cases are planned to ensure the quality of the system to determines that whether the system functionalities work accurately or not what more improvements are required.

6.4.1 User Registration Test Case

The registering new user is a simple process. The user is required to provide needed information. Once the user provides all the required information for registration. The system performs processing on the data and validate the data. If the user provides the valid data then the user is successfully registered otherwise, the user will be not registered. The test case determines whether the system performs registration as per requirements and most importantly, does the system give out the expected outcomes against the values entered by the user. Table 6.1 describe the registration test case for user. The user is tested for the registration. The test case shows that first of all the user open the registration page and then the user provide all the required data and then the system process that data. The outcome of the current test case is that the system verifying the data provided by the user and successfully registered the user. This test case follows the black box testing methodology and functional testing type. In black box testing the internal mechanism is unknown, it only focuses on the external working of the system that is checking the output against the given

input. In this test case, the system response is checked against the given input in the text fields for the User Registration form.

6. 1 User Registration Test Case

Test Case ID	**Use Case Reference**	
TC-1	UC-2	
Test Date	28-11-2019	
Revision History	None	
Objective	To register the new user.	
Environment	User Mode	
Assumptions	User is registered successfully.	
Pre-Requisite	User is at login page.	
Steps #	**Execution Description**	**Procedure Result**
1. **2.** **3.**	User opens the registration page User provides the required information Submitting details by selecting submit option	Registration page is displayed. Registration form is filled successfully Details submitted successfully

6.4.2 User/Admin Login Test Case

Table 6.2 explains the successful sign in of a user/admin so that user/admin is authorized and is able to use the Application. In the present test case the user provides the login information. The system validates the information. If the user is registered already.

In the system and admin, the system successfully login the user and takes to the main page. Otherwise if the username and password matches with the user data, the system successfully login the user and opens to user page. Black box testing methodology is used in this test case.

6. 2 User/Admin Login Test Case

Test Case ID	**Use Case Reference**	
TC-2	UC-1	
Test Date	28-11-2019	
Revision History	None	
Objective	User/Admin want to login to the system.	
Environment	User Mode, Admin Mode	
Assumptions	User/Admin logged in successfully.	
Pre-Requisite	User/Admin is at login page.	
Steps #	**Execution Description**	**Procedure Result**
1.	User/Admin Open the login form	Login page is displayed.
2.	User/Admin enters correct "User name" and "Password".	Registration form is filled successfully.
3.	System verify the information entered by the user. The system successfully validate the information.	The waiting screen is displayed to the user.
4.		

4.	User/Admin login by Clicking on login button	Successfully logged in
Comments Only list of available Items will be displayed and then further steps will be taken accordingly.		

6.4.3 QR Scanning Test Case

Table 6.3 explains loading an image. This test case also determines the performance testing as well. It ensures that the application performing fast to select the image and displays the results.

6. 3 QR Scanning Test Case

Test Case ID	**Use Case Reference**
TC-3	UC-3
Test Date	28-11-2019
Revision History	None
Objective	Admin wants to scan image.
Environment	Admin/User Mode.
Assumptions	Image is loaded successfully.

Pre-Requisite	System is in running condition.	
Steps #	**Execution Description**	**Procedure Result**
1.	Admin select on browse image/camera button.	System ask for image path from admin/user.
2.	Admin select image/camera to load.	Image selected/camera loaded successfully
3.	Admin click load to load image.	Image loading is started and after few moments loaded successfully.
Comments System allows all image formats to select.		

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Overview of Chapter

This chapter provides conclusion of the all the aspects of project report and also it is signifying the future work that is possible and which can be added in application in near future. The chapter also describe that what a system can do and what can't. The system restriction are the challenges for the developers which they have to overcome in future. Future work is planned so that it can add more value to the developed product.

7.2 System Overview

The project, Android Based Application for medicine tracking, lets the user to sets alarm for the provided medicine details, save prescription details entered by patient, scan prescription image and will convert it into a digital image by recognizing the characters, scan QR code of any medicine to show medicine details, generate a weekly or monthly report of medicines added to analyze progress of the treatment, save the details of the doctors entered by patient, set appointment reminder of the provided doctor's meeting, check medicine usage and details through Chatbot, will allow user to enter feedback and suggestions for the system. This Android based application is developed using React, React-Native, react API's and Python.

7.3 Milestones Achieved

Milestone of the developed system are as follows:

7.3.1 Alarm Reminder

The system sets alarm for the provided medicine details.

7.3.2 Add Prescription

The system saves prescription details entered by patient.

7.3.3 OCR Scanning

The system will scan prescription image and will convert it into a digital image by recognizing the characters.

7.3.4 QR Scanning

The system will scan QR code of any medicine to show medicine details.

7.3.5 Generate Report

The system will generate a weekly or monthly report of medicines added to analyze progress of the treatment.

7.3.6 Add doctor details

The system will save the details of the doctors entered by patient.

7.3.7 Add an Appointment

The system will set appointment reminder of the provided doctor’s meeting.

7.3.8 Medicine Checker

The system check medicine usage and details through Chatbot.

7.3.9 Feedback

The system will allow user to enter feedback and suggestions for the system.

7.4 Limitations

1. Only available in English language
2. Phone must be charged to get alerts
3. Phone OS should be compatible with system software.

7.5 Future Work

In the future, the IOS version of this application is also possible to develop as React native acts as a native platform for different OS. And with that being made, IOS users can also use this application in their devices. Moreover, a tutorial guide can be added to it as well if new users find it difficult to understand the UI.

7.6 Summary

This is an android based project. Remembering the exact time of taking exact prescribed medicines can be very challenging for some especially for those who don't have anyone to look after them. And in the times of current pandemic, this issue seems to be a common problem for many people. So, there should be some sort of assistant for these patients who can remind them to take proper medicines and right on time. There are some formal ways like through an alarm clock app or hiring a care taker. But these methods either don't fulfill the needs of a patient or can be very challenging. To overcome these problem, we have proposed a system named as "Medical Assistant".

Most of the times, the hand writing of the doctor is not clear or understandable to the normal human which makes it very difficult to see the time of the prescribed medicine. To overcome that problem, we have included an OCR text reader in our system.

Sometimes people need to just identify the use of a medicine just by entering the medicine name. For that, we have introduced a chat bot that would give disease name by just putting medicine name on the system.