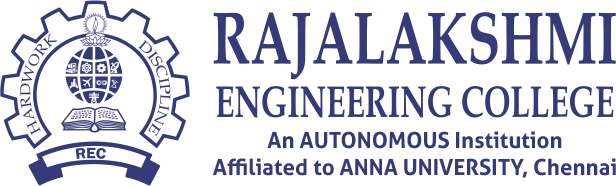
# 

# 



**Clinic Management System**

**Submitted by**

**Hemanth Kumar D (231801057)**

**CS23333-Object oriented Programming using java**

**Department of Artificial Intelligence and Data Science**

**Rajalakshmi Engineering College, Thandalam**

**Nov 202**

# ABSTRACT

The project **“Clinical Management System”** in Java is a [simple project](https://code-projects.org/tank-war-game-in-java-with-source-code/) for the management of small clinics and medical shops. The whole system is in **Netbeans IDE**. The system does have **MySQL** as database support. The database keeps all records of patient and transaction details for a particular clinic and medical shop. A particular clinic and medical shop can handle this by just using the features provided in the window. This software reduces the amount of manual data entry and gives greater efficiency which in turn saves the time. The system supports all the features that meet the project requirement and is doing all the work accurately.

.

# 

# TABLE OF CONTENTS

[ABSTRACT i](#_Toc183111782)

[TABLE OF CONTENTS ii](#_Toc183111783)

[1. INTRODUCTION 5](#_Toc183111784)

[1.1 PROJECT SUMMARY 5](#_Toc183111785)

[1.2 PURPOSE 5](#_Toc183111786)

[1.3 SCOPE 5](#_Toc183111787)

[1.4 OBJECTIVES 5](#_Toc183111788)

[2. SYSTEM REQUIREMENTS STUDY 6](#_Toc183111789)

[2.1 USER CHARACTERISTICS 6](#_Toc183111790)

[2.2 HARDWARE REQUIREMENTS 6](#_Toc183111791)

[2.3 SOFTWARE REQUIREMENTS 6](#_Toc183111792)

[3.1 STUDY OF CURRENT SYSTEM 7](#_Toc183111793)

[3.2 FLOW CHART 7](#_Toc183111794)

[4. LIMITATIONS AND FUTURE ENHANCEMENT 8](#_Toc183111795)

[4.1 LIMITATIONS 8](#_Toc183111796)

[4.2 FUTURE ENHANCEMENTS 8](#_Toc183111797)

[5. CONCLUSION AND DISCUSSION 9](#_Toc183111798)

[5.1 SELF ANALYSIS OF PROJECT VIABILITIES 9](#_Toc183111799)

[5.2 PROBLEM ENCOUNTERED AND POSSIBLE SOLUTIONS 9](#_Toc183111800)

[6.3 SUMMARY OF PROJECT WORK 9](#_Toc183111801)

[6. SCREENSHOTS 10](#_Toc183111802)

[6.1 INPUT / OUTPUT AND INTERFACE DESIGN 10](#_Toc183111803)

[7. REFERENCES 16](#_Toc183111804)

LIST OF FIGURES

Fig. 1.1 Progress Graph (Week vs Time)…………………………………………….. 8

Fig. 2.1 Flow Chart 10

Fig. 3.1 Main Screen 13

Fig. 3.2 Reception Module 13

Fig. 3.3 Reception Login 14

Fig. 3.4 Reception Change Password 14

Fig. 3.5 Add Patient ………………………………………………………………… 14

Fig. 3.6 Appointment………………………………………………………………... 14

Fig. 3.7 Update Patient Details……………………………………………………… 15

Fig. 3.8 Update Contact………………………………………………………………15

Fig. 3.9 Appointment List………………………………………………………….... 15

Fig. 3.10 Doctor Module…………………………………………………………….. 16

Fig. 3.11 Doctor Login……………………………………………………………..... 16

Fig. 3.12 Doctor Change Password……………………..………………………….... 16

Fig. 3.13 Patient Info…………………………...……………………………………. 16

Fig. 3.14 Add Report…………...……………………………………………………. 17

Fig. 3.15 Past Report……………………………………………………………….... 17

Fig. 3.16 Chemist Module……………………………………...……………………. 18

Fig. 3.17 Chemist Login…...………………………………………………………….18

Fig. 3.18 Chemist Change Password………………………………………………….18

Fig. 3.19 Medicine Info……………………………………………………………….18

Fig. 3.20 Add Medicine……………………………………………………………….18

Fig. 3.21 About Us……………………………………………………………………19

Fig. 3.22 Contact Us………………………………………………………………......19

**LIST OF TABLES**

Table 1.1 Project Effort and time 7

Table 1.2 Roles and responsibilties 8

# INTRODUCTION

## PROJECT SUMMARY

* In today’s digital era, everything is becoming digital and paperless but still there are few small clinics in which a file is given to us which contain all our information on bunch of papers.
* But there are several disadvantages of this traditional file system. So to overcome those problems, we have made a desktop application, which will keep those records saved in our system.
* So, basically it is a software which helps the user to work with the clinical work easily.

## PURPOSE

* Our main purpose behind making this project is to overcome the difficulties that some small clinics are facing nowadays.
* Talking about this system, it comes with three modules. One for the Receptionist, second for the Doctor and the third for the Chemist.
* Design of this is so simple that the one won’t find difficulties while working on it. In just few clicks, one can get whole history of the patient. This project is easy to operate and is very user-friendly.

## 1.3 SCOPE

* This application is not for all clinics or hospitals. It only fulfills some basic problems occurring during clinical work.
* There is no Patient Module in it, so the patient cannot install this application and see his/her past reports.
* This application is stick to only one doctor. One cannot add new doctor in it.

## 1.4 OBJECTIVES

* Our objective is to reduce the amount of manual data entry and give greater efficiency and increase service effectiveness.
* Also with this, we have made the billing system automatic. No need to manually add medicine names in bill and calculate. The system automatically generates bill based on the prescription prescribed by the doctor.
* At the end, we can say that this software is performing all the tasks accurately and is doing the work for which it is made.

# 2. SYSTEM REQUIREMENTS STUDY

## 2.1 USER CHARACTERISTICS

* **Receptionist :** He/She needs to enter patient’s data into our database and give him/her appointment.
* **Doctor :** He/She needs to consult the patient and fill up the medicine and symptom details for the further treatment.
* **Chemist :** He/She needs to give the patient the medicines prescribed by the doctor and take the billing amount of medicines given including doctor’s consulting fees.

## 2.2 HARDWARE REQUIREMENTS

* RAM – Minimum 2 GB
* Hard Disk –Minimum 20 GB

## 2.3 SOFTWARE REQUIREMENTS

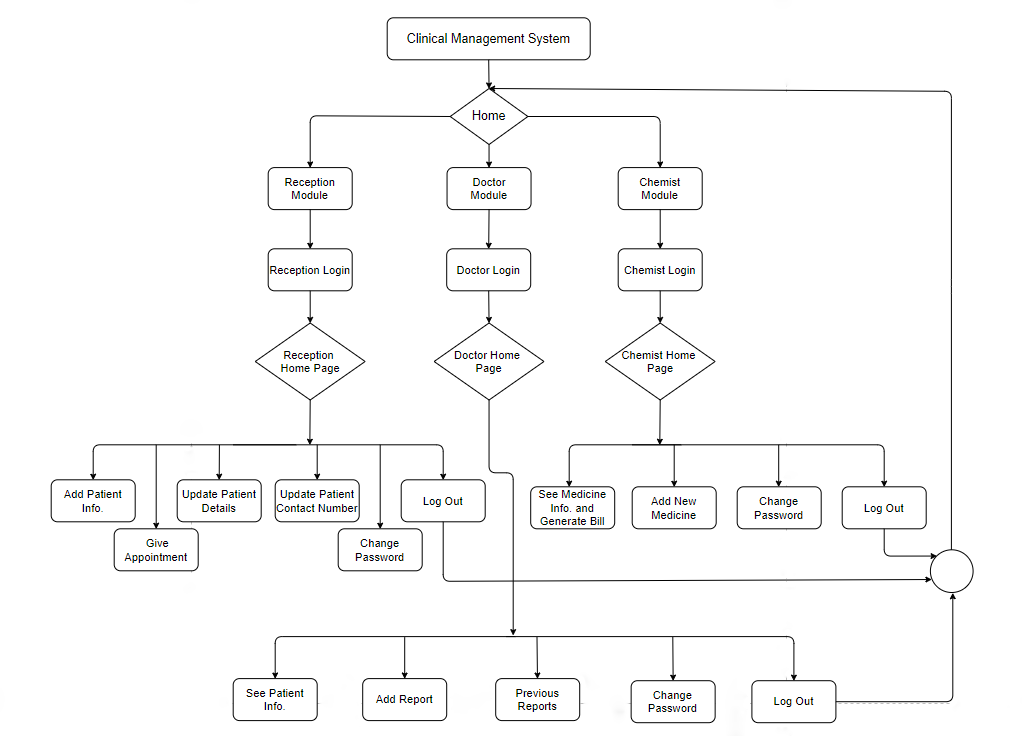
* Windows XP or higher versions
* Netbeans IDE
* MySQL Database

**3. SYSTEM ANALYSIS**

## 3.1 STUDY OF CURRENT SYSTEM

* When patient comes to clinic or hospital first of all he/she will approach to the receptionist and will take appointment from receptionist. As we are given file from receptionist instead of it receptionist will register patient in their database. Receptionist will take and add data of patient into their database and give him/her appointment. Now this data will be added in database.
* When patient goes to the doctor, the doctor will be able to see all his/her details. If patient has come first time then prescription and medicines given to him/her will be added in database for further treatment. And if patient has taken treatment before that day then his/her previous record will be shown to doctor. As per the previous record and currant situation doctor can do further treatment.
* A digital copy of the prescription prescribed to the patient will be sent to the chemist so that he/she can provide the medicines to the patient. At the chemist’s shop, a digital bill will also be generated including doctor’s consulting fees. The patient needs to pay the amount at chemist’s shop only.

## 3.2 FLOW CHART



**Fig. 2.1 Flow Chart**

# 4. LIMITATIONS AND FUTURE ENHANCEMENT

## 4.1 LIMITATIONS

* If a patient don’t have a mobile then it is difficult to store his/her data into this system.
* If doctor prescribes any medicine which needs to be taken at some specific reason or time then this system is not able to add such kind of report.
* At the end of each day, the receptionist have to manually reset the appointment list.
* Appointment cannot be taken at some specific time. The patient needs to visit the clinic and take the appointment.

## 4.2 FUTURE ENHANCEMENTS

* In future, we will try to add some more functionalities to it which can still increase the overall efficiency of this system.
* We will also look into the all those problems encountered or faced by the user while using this application and try to get best possible solution for it.
* In future, we can also expand this application for multi specialty hospitals by adding more functionalities to it according to their needs.

## 

# 5. CONCLUSION AND DISCUSSION

## 5.1 SELF ANALYSIS OF PROJECT VIABILITIES

* Our main motive behind making this project was to learn and implement Java and its features. We also came to know that how data is handled in MySQL database.
* Clinical Management system is an application which covers all parts of management and tasks of facilities. This application covers Doctors module, Reception Module, Chemist Module, Scheduling Appointments, Patient treatment reports, Prescription report, billings, and so forth.
* This is a good project and it boosted up our career with this beginner-level java project.

## 5.2 PROBLEM ENCOUNTERED AND POSSIBLE SOLUTIONS

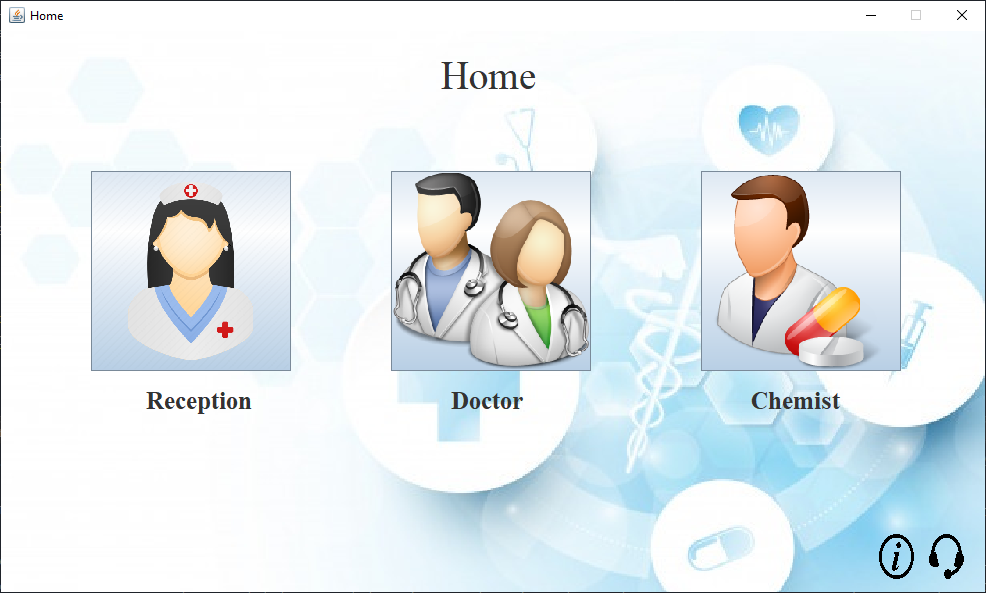
* At the end of each day, the receptionist have to manually reset the appointment list. The solution is that we can create new table everyday.
* If doctor prescribes any medicine which needs to be taken at some specific reason or time then this system is not able to add such kind of report. To overcome this, we can add one textarea in which doctor adds when to take that medicine.
* If a patient don’t have a mobile then it is difficult to store his/her data into this system. Possible solution is that we can allocate unique Patient ID to each patient.

## 6.3 SUMMARY OF PROJECT WORK

* Nowadays Java is often a default choice for scientific applications, including natural language processing. Main reason of this is because Java is more safe, portable, maintainable and comes with better high-level concurrency tools than C++ or any other language.
* In this Project, we tried to make a small application using Java which may solve our daily problems like less use of paper, saving the time, etc. and giving us the relevant information whenever we want. All this information is stored in MySQL database.
* So we can say that this application is performing all the tasks accurately and is doing all the work for which it is made.
* We can further improve our application by overcoming the current problems encounterd and by adding more and more functionalities to it.

# 6. SCREENSHOTS

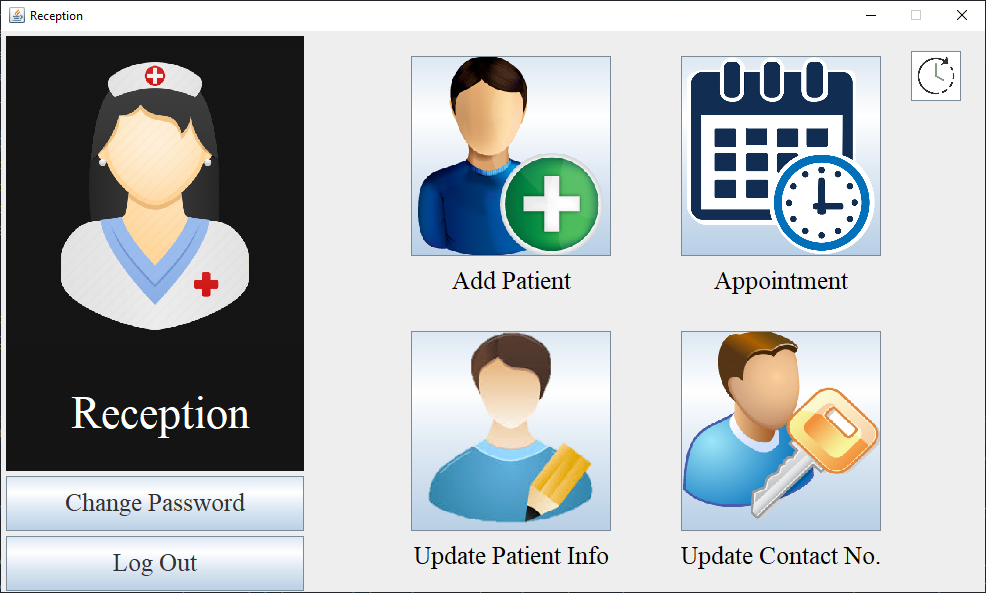
## 6.1 INPUT / OUTPUT AND INTERFACE DESIGN



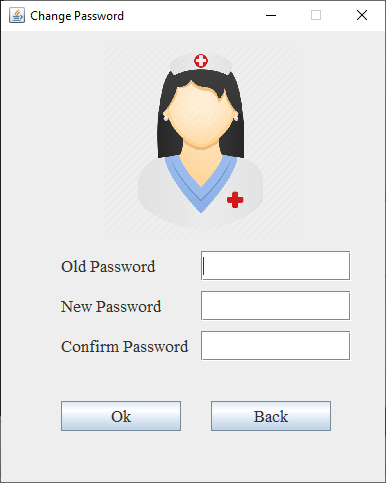
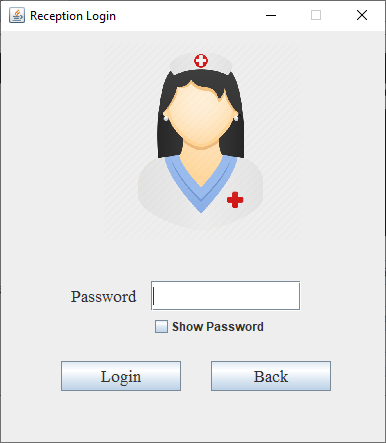
**Fig. 3.1 Main Screen**

**6.1.1 Module Preview**

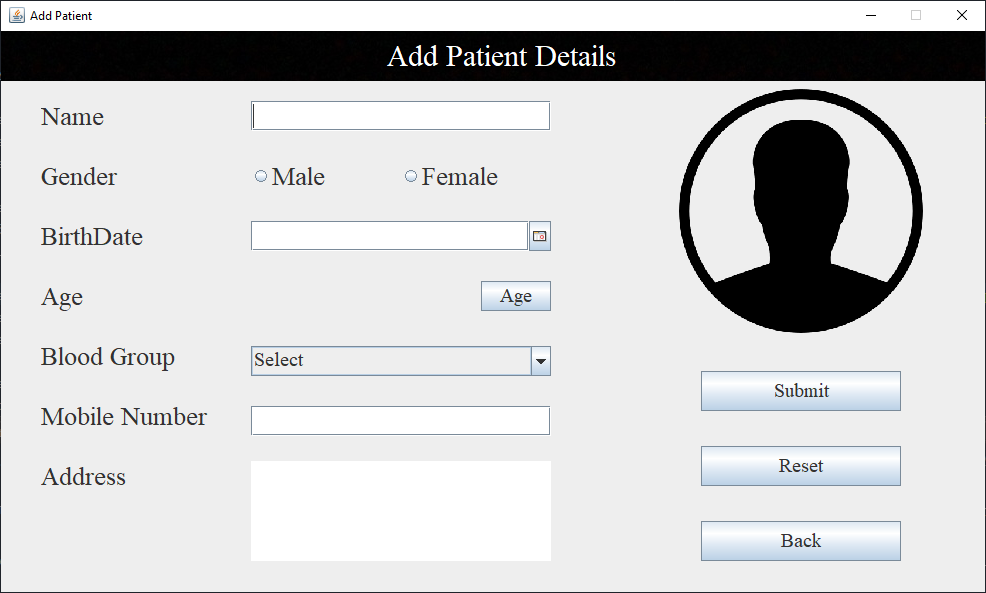
**Reception Module Features :** *Add patient, Appointment, Update Patient Info, Update Contact, Appointment List*



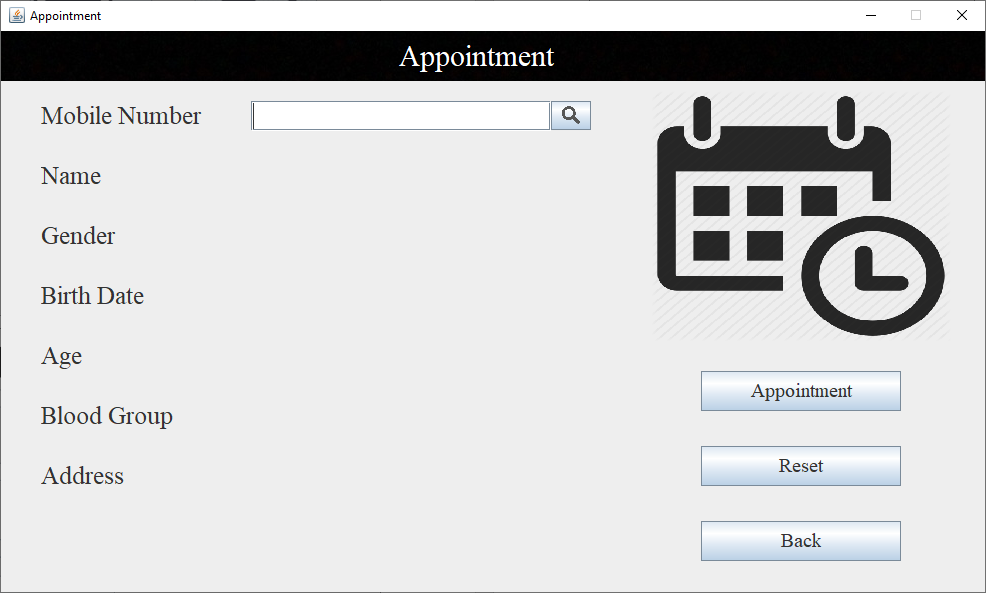
**Fig. 3.2 Reception Module**



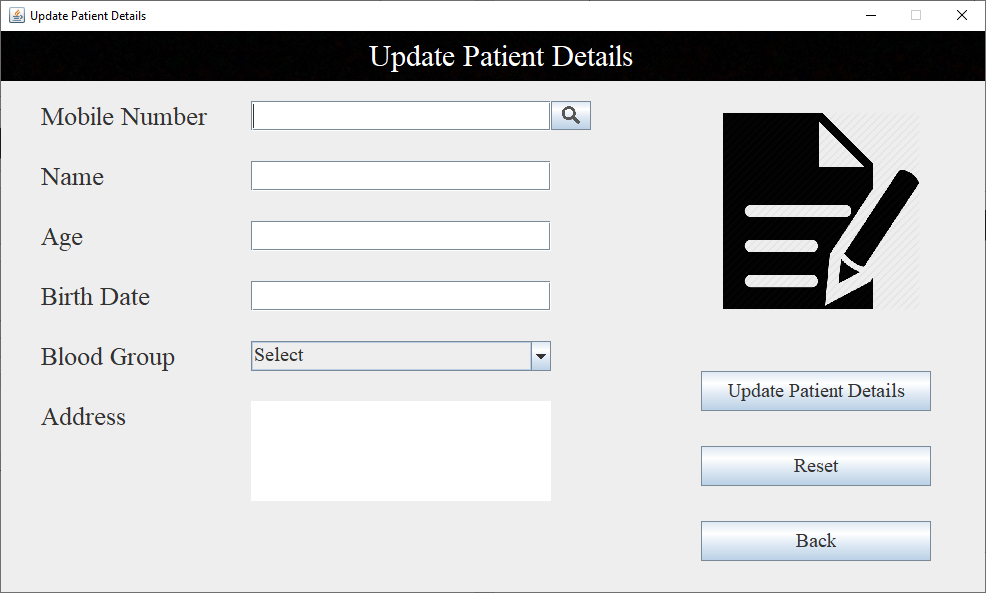
**Fig. 3.3 Reception Login Fig. 3.4 Reception Change Password**



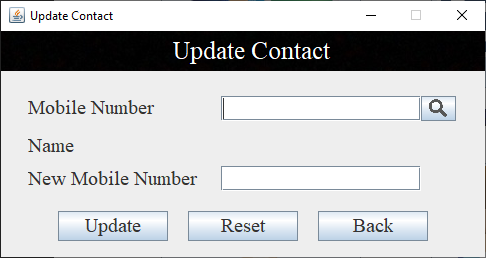
**Fig. 3.5 Add Patient**



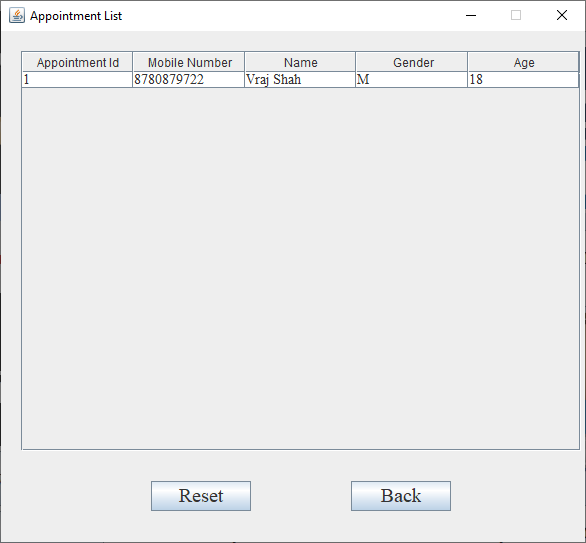
**Fig. 3.6 Appointment**



**Fig. 3.7 Update Patient Details**

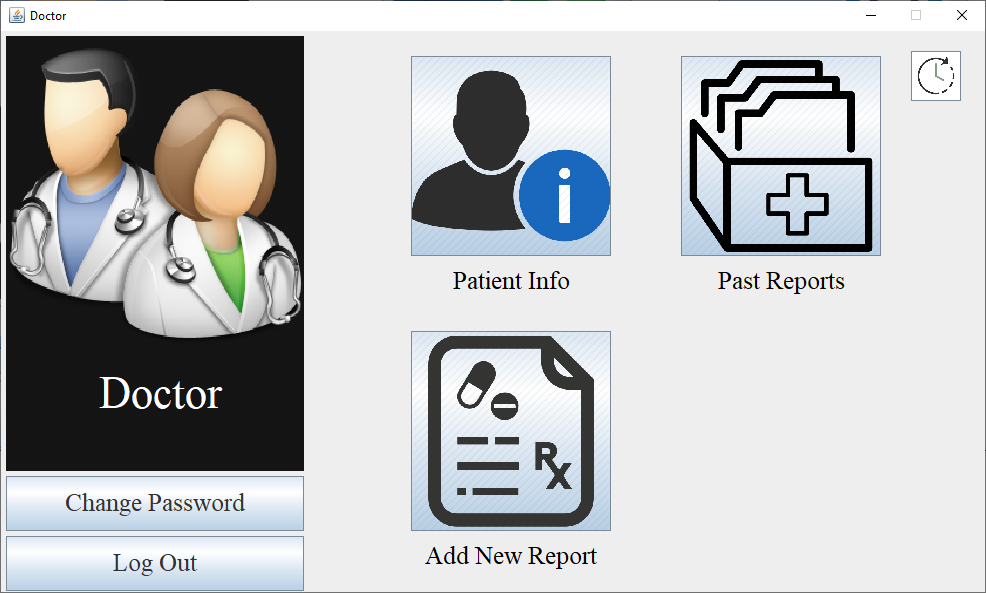


**Fig. 3.8 Update Contact**

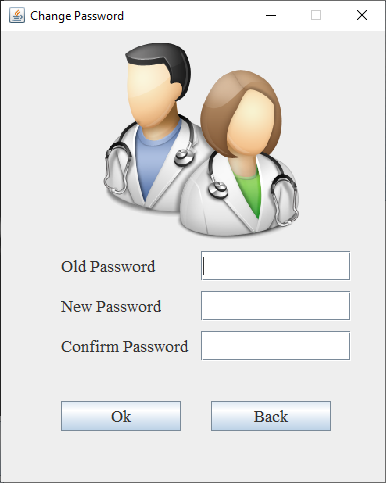
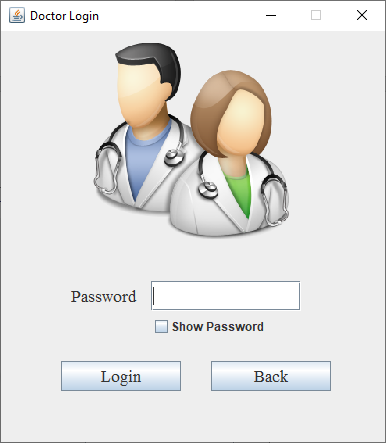


**Fig. 3.9 Appointment List**

**Doctor Module Features :** *Patient Info, Add Report, Past Reports, Appointment List*



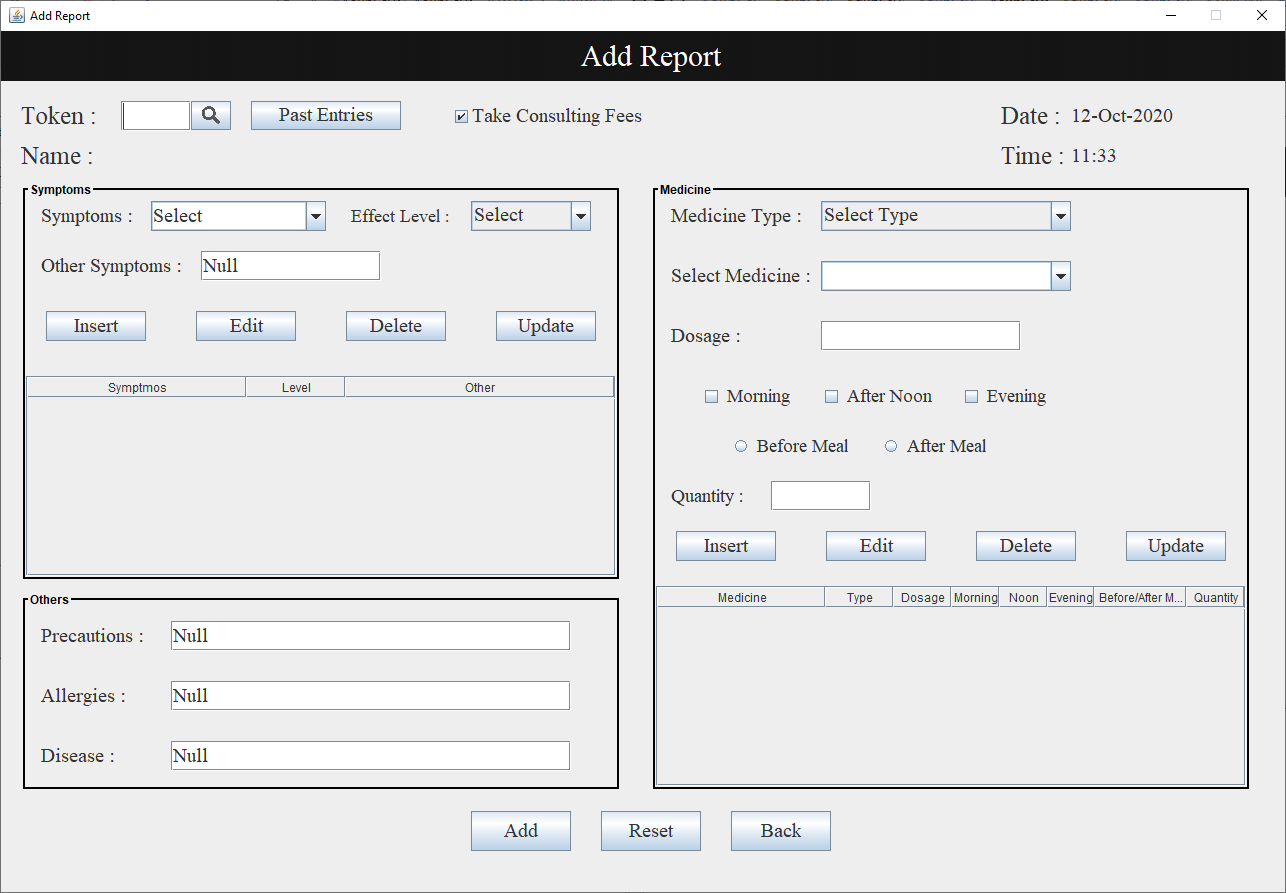
**Fig. 3.10 Doctor Module**



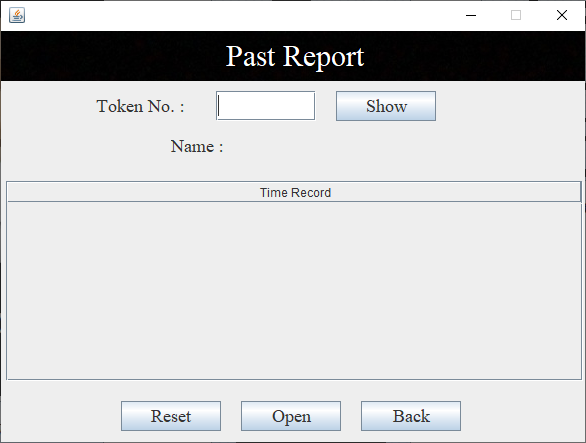
**Fig. 3.11 Doctor Login Fig. 3.12 Doctor Change Password**



**Fig. 3.13 Patient Info**

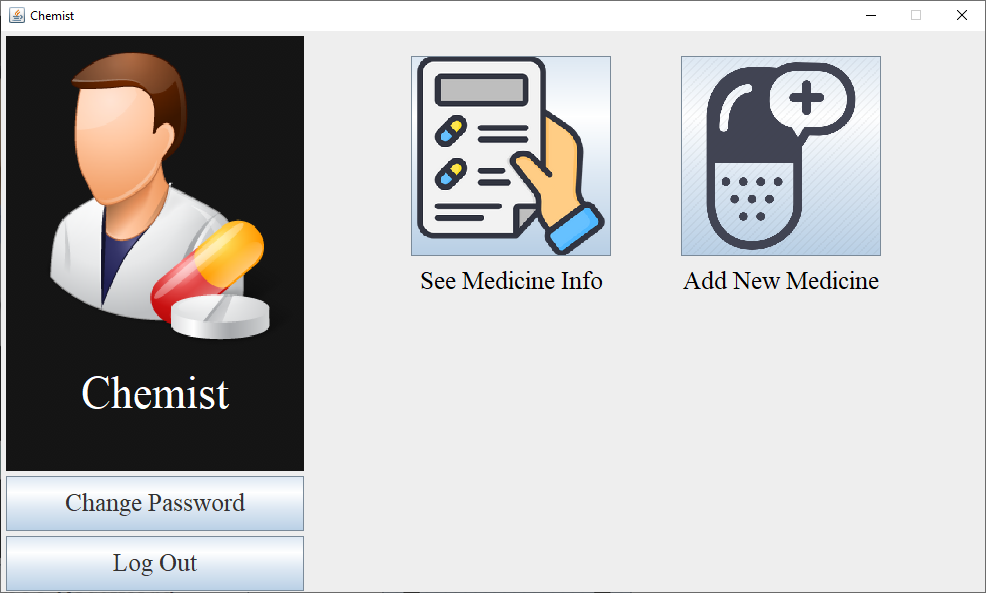


**Fig. 3.14 Add Report**

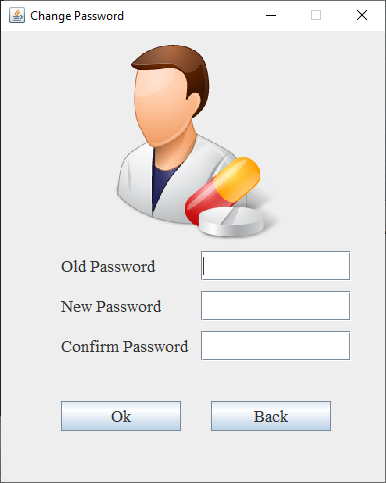
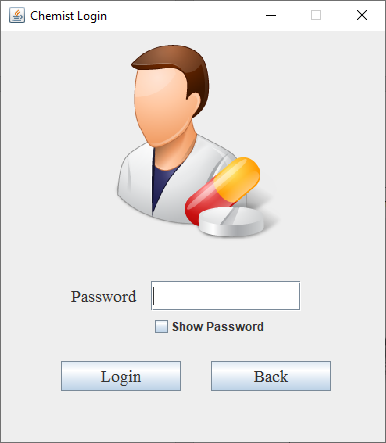


**Fig. 3.15 Past Report**

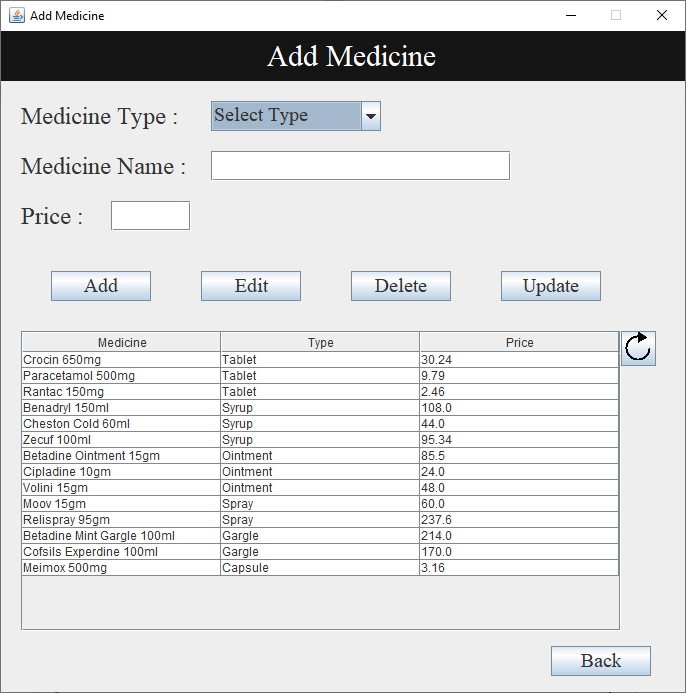
**Chemist Module Features :** *See Medicine Info, Add New Medicine*



**Fig. 3.16 Chemist Module**



**Fig. 3.17 Chemist Login Fig. 3.18 Chemist Change Password**



# 7. REFERENCES

**Web reference:**

[i] <https://www.edureka.co/blog/java-swing>[/](https://www.edureka.co/blog/java-swing/)

[ii] https://docs.oracle.com/javase/tutorial/uiswing/components/table.html

[iii] <https>[://www.tutorialspoint.com/swing/index.htm](https://www.tutorialspoint.com/swing/index.htm)

[iv] <https://www.w3schools.com/sql/>

[v] <https://youtu.be/mgw1sAvhjy8>

[vi] <https://www.javatpoint.com/java-jdbc>

[vii] <https://youtu.be/eYTJM0emDes>

[viii] https://youtu.be/Zg7lS5sPN0M