A

Project Report

On

"HOSPITAL MANAGEMENT SYSTEM"

Submitted to

Department of Computer Engineering,

PDEA's College of Engineering, Manjari BK,

Pune-412307

Submitted by

Swapnil Sawant **(30)**

Abraar Khan **(22)**

Atharva Shrotre **(37)**

Niteen Kale **(36)**

CERTIFICATE

This is to certify that the below mentioned third year engineering students have carried out the necessary a project report work on **"HOSPITAL MANAGEMENT SYSTEM"** in the department of Computer Engineering, PDEA's College of Engineering, Manjari BK, Pune-412307. They have completed this project work under my guidance in satisfactory manner in October 2019 of third year engineering.

Computer Engineering students have successfully completed a project report on **"HOSPITAL MANAGEMENT SYSTEM"** towards the fulfillment of their Degree in Computer Engineering in academic year 2019-2020. The performance of each of these students during the course was very good.

Place:

Date:

Prof. S.V. Phulari — **Guide**

Dr. R.V. Patil — **Principal/H.O.D.**

ACKNOWLEDGEMENT

Apart from the efforts of all the team members, the section of this project report topic depends largely on the encouragement and guidance of our teachers. We take this opportunity to express our gratitude to the teachers who have been instrumental in the approval of this project topic.

We would like to show our greatest appreciation to **Prof. S.V. Phulari** and other staff members. We cannot think them enough for their tremendous support and help. They motivated and encouraged use very time while selecting the proper project topic. Without their encouragement and guidance, we would not have been able to select the proper topic.

The contribution and support received from all the team members including **Swapnil Sawant, Abraar Khan, Atharva Shrotre & Niteen Kale** is vital. The team spirit shown by all has made a project report work successful.

ABSTRACT

The purpose of the project entitled as "Hospital Management System" is to computerize the Front Office Management of Hospital to develop software which is user friendly simple, fast, and cost – effective. It deals with the collection of patient's information like add patient, update patient, delete patient, search patient, view patient diagnosis, etc. Traditionally, it was done manually. The main function of the system is register and store patient details and doctor details and retrieve these details as and when required, and also to manipulate these details meaningfully. The Hospital Management System can be entered using a username and password. It is accessible by an Admin, Doctor & Receptionist. Only they can add data into the database. The data can be retrieved easily. The data are well protected for personal use and makes the data processing very fast.

Government of India has still aimed at providing medical facilities by establishing hospital. The basic working of various hospitals in India is still on paper as compared to hospitals in European countries where computers have been put in to assist the hospital personals their work. The concept of automation of the administration and management of hospital is now being implemented in India also, with large hospitals like APPOLO and AIIMS in Delhi, ESCORTS in Chennai, having automated their existing system.

Computers are not only used to increase the efficiency in all fields ranging from fixing the appointment with the Doctor to keeping the record of the Patient.

INDEX

1. INTRODUCTION

Health of citizen is the wealth of Nation. India has contributes the most ancient Medical science “AYURVEDA” to the world besides other medical sciences. This field had witnessed a rapid metamorphosis in all of its sections. Hospital Management System is designed to improve the quality and management of hospital in the areas of clinical process analysis and activity-based costing. Hospital Management System enables you to develop your organization and improve its effectiveness and quality of work. Managing the key processes efficiently is critical to the success of the hospital helps you manage your processes.

The Hospital Management System can be entered using a username and password. It is accessible by an Admin, Doctor & Receptionist. Only they can add data into the database. The data can be retrieved easily. The data are well protected for personal use and makes the data processing very fast. Hospital Management System is powerful, flexible, and easy to use and is designed and developed to deliver real conceivable benefits to hospitals.

The project Hospital Management system includes registration of patients, storing their details into the system by using database. The software has the facility to give a unique id for every patient and stores the details of every patient and the staff manually. Admin can view availability of a doctor and the details of a patient using the name, id.

Hospital Management System is designed for multi specialist hospitals, to cover a wide range of hospital administration and management processes. It also aims at providing low-cost reliable automation of the existing systems. The system also provides excellent security of data at every level of user-system interaction and also provides robust & reliable storage facilities.

1.1 Modules:

The entire project mainly consists of 6 modules, which are

- Admins
    - AdminActivity
    - ReceptionistManagement
    - doctorManagement
- Doctors
    - Appointment
    - addDoctor
    - deleteDoctor
    - doctorActivity
    - patientDetails
    - searchDoctor
    - updateDoctor
    - viewDoctor
- Main
    - Admin
    - Connector
    - Doctor
    - Hospital
    - Receptionist
- Patient
    - DeletePatient
    - addPatient
    - patientActivity
    - searchPatient
    - updatePatient
    - viewPatient
- Receptionists
    - ReceptionistActivity
    - addReceptionist
    - chooseDoctor
    - choosePatient

- deleteReceptionist
- newPatient
- patientManagement
- searchReceptionist
- viewReceptionist
- updateReceptionist

2. SYSTEM ANALYSIS

2.1 Existing system:

The current manual system has a lot of paper work. To maintain the records of sale and service manually, is a Time-consuming task. With the increase in database, it will become a massive task to maintain the database. Requires large quantities of file cabinets, which are huge and require quite a bit of space in the office, which can be used for storing records of previous details. The retrieval of records of previously registered patients will be a tedious task. Lack of security for the records, anyone disarrange the records of your system. If someone want to check the details of the available doctors the previous system does not provide any necessary detail of this type.

All this work is done manually by the receptionist and other operational staff and lot of papers are needed to be handled and taken care of. Doctors have to remember various medicines available for diagnosis and sometimes miss better alternatives as they can't remember them at that time.

- **Advantages:**
    1. No extra training required.
    2. Easy to implement.
    3. Can be stored anywhere.
    4. Requires minimum effort.

- **Disadvantages:**
    1. Needs lots of paper.
    2. Problem with maintenance.
    3. Volumes of data becomes problem.
    4. Once data is burned it cannot be reproduced easily.
    5. Data handling is problem.

2.2 Proposed system:

The Hospital Management System is designed for any hospital to replace their existing manual paper-based system. The new system is to control the information of patients as well as doctors. These services are to be provided in an efficient, cost effective manner, with the goal of reducing the time and resources currently required for such tasks.

The complete set of rules & procedures related to Hospital's day to day activities and generating report is called "Hospital Management System". It is a computerized management system. This system also keeps the records of hardware assets besides software of this organization. The proposed system will keep a track of Doctors, Patients & Receptionist. This project has GUI based software that will help in storing, updating and retrieving the information through various user-friendly menu-driven modules.

- **Goals of proposed system:**
  i. The system should be easy to operate.
  ii. The working in the organization will be well planned and organized.
  iii. The level of accuracy in the proposed system will be higher.
  iv. The reliability of the proposed system will be high due to proper storage of information.
  v. Provide quick and efficient retrieval of information.

- **Advantages:**
  1. Low maintenance cost.
  2. Volume of data is not an issue.
  3. Data can be converted easily to information.
  4. Data cannot be corrupted easily with proper backup.
  5. It can be expanded as well as data communication is possible.

- **Disadvantages:**
  1. High starting cost requires.
  2. Additional manpower is necessary.
  3. Data communication system will have an additional cost.

3. LITERATURE SURVEY.

One of the major challenges existing hospital management systems face is around operational efficiency and wait times between different processes, departments and persons. This paper highlights such limitations of existing systems and proposes a RFID (Radio Frequency ID) and wireless sensor based, location and information management framework that facilitates real time tracking of hospital assets, personnel and patients as they move through pre-set procedures as part of daily activities of the hospitals. The system covers the visual simulation and providing ability to analyze the ongoing operations so they can be corrected to achieve increased process efficiency and service levels.

Hospitals are complex organizations which, in addition to the technical assistance expected in the context of treatment and prevention of health hazards, also require good management practices aimed at improving their efficiency in their core business. However, in administrative terms, recurrent conflicts arise involving technical and managerial areas.

4. FUTURE SCOPE

All this work is done manually by the receptionist and other operational staff and lot of papers are needed to be handled and taken care of. Doctors have to remember various medicines available for diagnosis and sometimes miss better alternatives as they can’t remember them at that time. The limited time and resources have restricted us to incorporate, in this project, only main activities that are performed in a Hospital Management System, but utmost care has been taken to make the system efficient and user friendly.

Most of the analysis and interpretations, made for this report, are based on secondary data obtained. This data could have some inherent mistakes and errors. Finally, although due care has been taken those can be typing and compilation errors in the report itself. The tasks specified were not well defined because nothing was mentioned regarding validations in the project. Though we gave maximum effort to check the software. But it in no way alters the ultimate aim of the project and because it's highly USER FRIENDLY, it would be the choice of all kinds of personnel.

5. REQUIREMENT SPECIFICATION

5.1 Hardware specification:

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatibility and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

⇒ Hardware requirements for present project:

Processor : Intel core i3
RAM : 4GB
Hard disk : 1TB (Minimum 80GB)

5.2 Software specification:

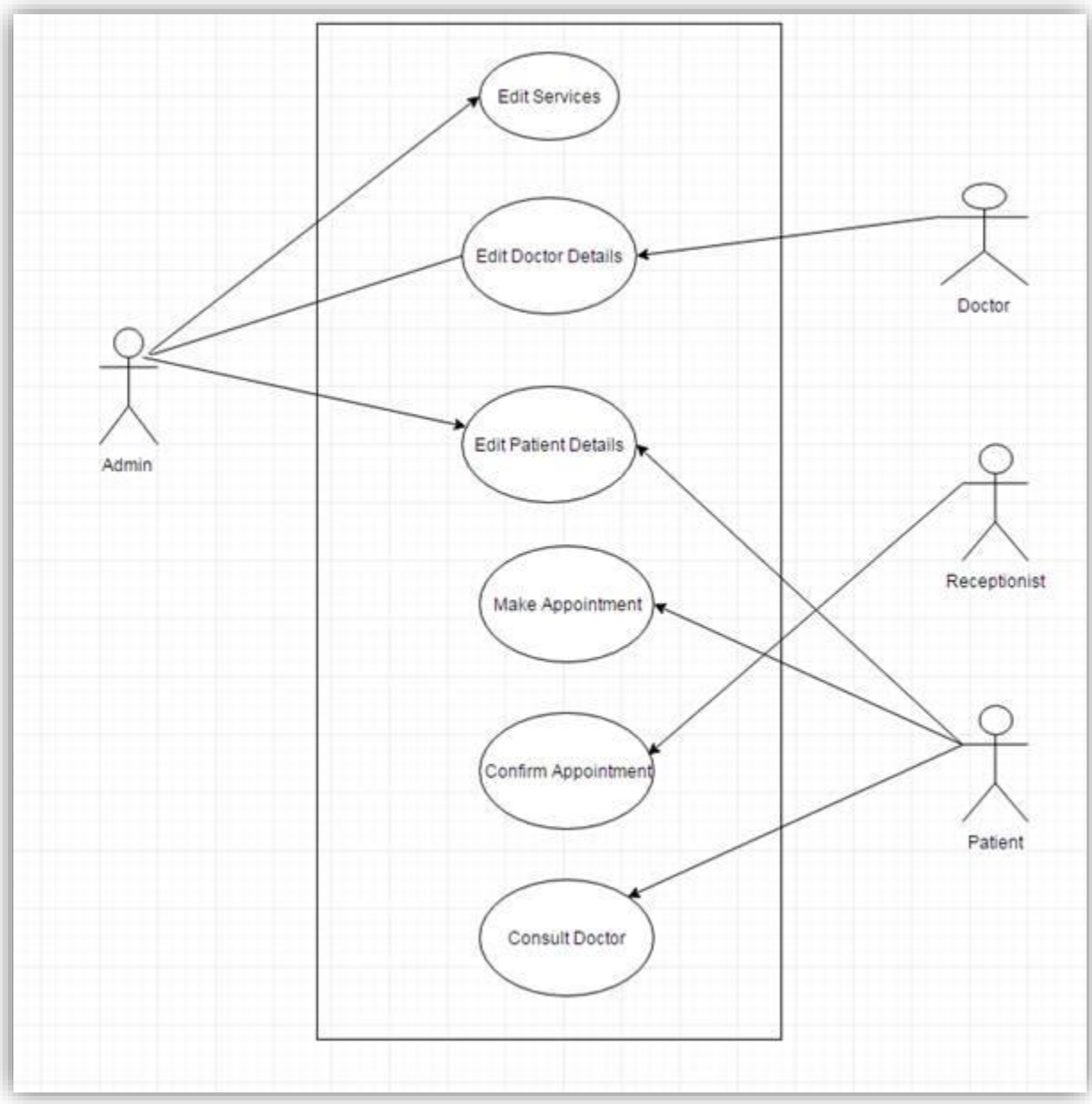
Software Requirements deal with defining software resource requirements and pre-requisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed.

⇒ Software requirements for present project:

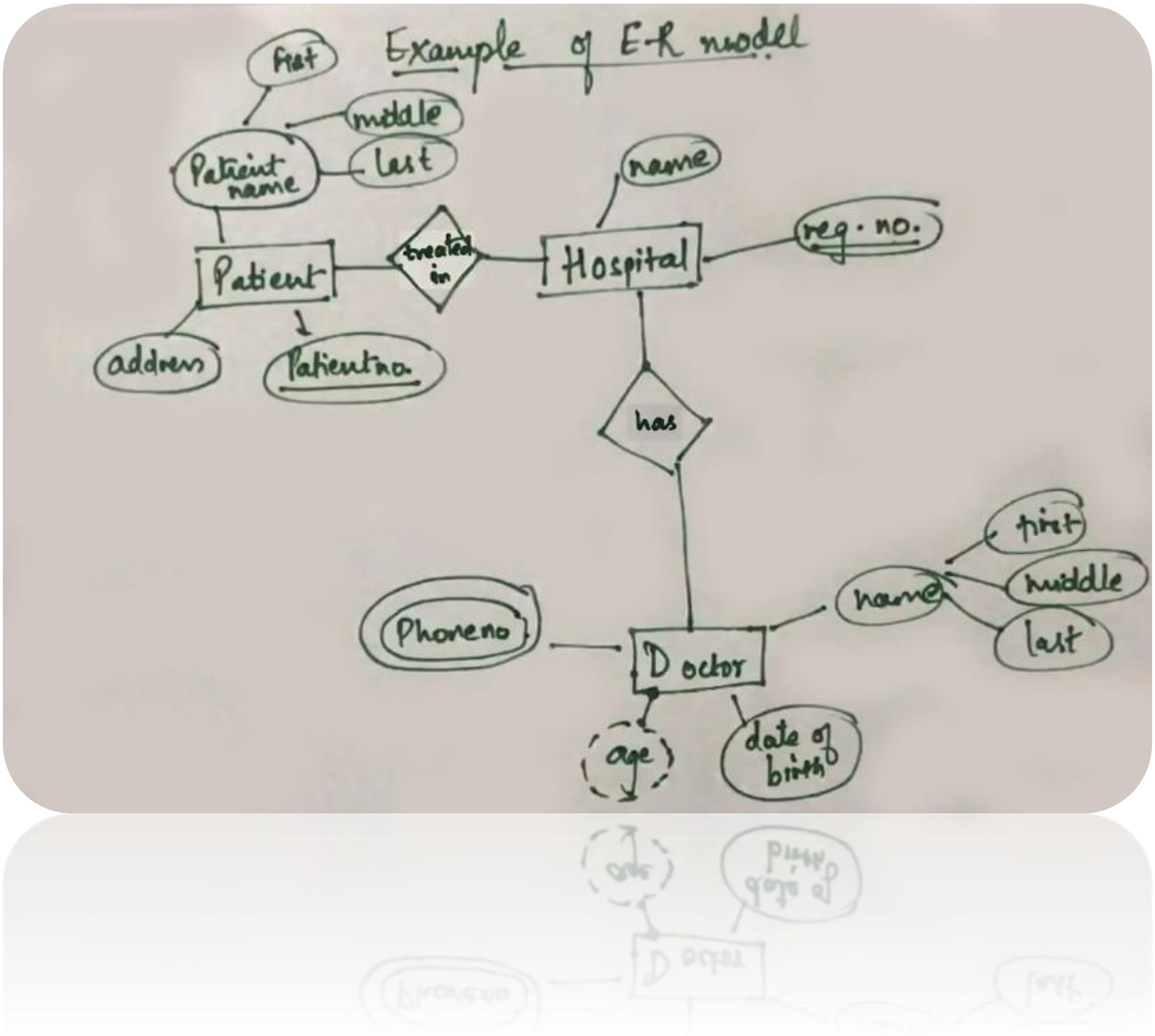
Operating system **:** Windows 7, 10
Front-end **:** Java
Back-end **:** MySQL
Development environment/tools: NetBeans IDE 8.2 & Xampp

6. SYSTEM DESIGN

6.1 Use case diagram:



6.2 E-R diagram:



7. SYSTEM IMPLEMENTATION

7.1 Introduction:

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective. The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

7.2 Sample code for Main screen (Hospital.java):

```
package Main;

import javax.swing.ImageIcon;

public class Hospital extends javax.swing.JFrame {

    public Hospital() {

        initComponents();

        ImageIcon ic = new ImageIcon(getClass().getResource("/Images/hospital.png"));

        this.setIconImage(ic.getImage());

    }

    @SuppressWarnings("unchecked")

    // <editor-fold defaultstate="collapsed" desc="Generated Code">

    private void initComponents() {

        MainF = new javax.swing.JPanel();

        jPanel1 = new javax.swing.JPanel();

        adminIcon = new javax.swing.JLabel();

        adminLabel = new javax.swing.JLabel();
```

```
        mLabel = new java.awt.Label();
        jPanel2 = new javax.swing.JPanel();
        receptionIcon = new javax.swing.JLabel();
        receptionLabel = new javax.swing.JLabel();
        jPanel3 = new javax.swing.JPanel();
        doctorIcon = new javax.swing.JLabel();
        doctorLabel = new javax.swing.JLabel();
        jPanel4 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jPanel5 = new javax.swing.JPanel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setTitle("Welcome To Hospital Management System");
        setResizable(false);
        setSize(new java.awt.Dimension(800, 480));
        MainF.setName("Welcome To Hospital Management System"); // NOI18N
        jPanel1.setBackground(java.awt.Color.lightGray);
        adminIcon.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/Icon/admin.png"))); // NOI18N
        adminIcon.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
                adminIconMouseClicked(evt);
            }
```

```
        });
        adminLabel.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N
        adminLabel.setForeground(new java.awt.Color(255, 255, 255));
        adminLabel.setText("Admin");
        adminLabel.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
                adminLabelMouseClicked(evt);
            }
        });
        javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(54, 54, 54)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILI
NG)
                    .addComponent(adminLabel)
                    .addComponent(adminIcon))
                .addContainerGap(57, Short.MAX_VALUE))
        );
        jPanel1Layout.setVerticalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
```

```
            .addContainerGap()
            .addComponent(adminIcon)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(adminLabel)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );
        mLabel.setAlignment(java.awt.Label.CENTER);
        mLabel.setBackground(new java.awt.Color(0, 153, 153));
        mLabel.setFont(new java.awt.Font("Arial", 0, 24)); // NOI18N
        mLabel.setForeground(new java.awt.Color(255, 255, 255));
        mLabel.setText("Hospital Management System");
        jPanel2.setBackground(java.awt.Color.lightGray);
        receptionIcon.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/Icon/reception.png"))); // NOI18N
        receptionIcon.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
                receptionIconMouseClicked(evt);
            }
        });
        receptionLabel.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N
        receptionLabel.setForeground(new java.awt.Color(255, 255, 255));
        receptionLabel.setText("Receptionist");
        receptionLabel.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
```

```
            receptionLabelMouseClicked(evt);
        }
    });
    javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
    jPanel2.setLayout(jPanel2Layout);
    jPanel2Layout.setHorizontalGroup(
        jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addContainerGap(27, Short.MAX_VALUE)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
                .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel2Layout.createSequentialGroup()
                    .addComponent(receptionLabel)
                    .addGap(24, 24, 24))
                .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel2Layout.createSequentialGroup()
                    .addComponent(receptionIcon)
                    .addGap(50, 50, 50))))
    );
    jPanel2Layout.setVerticalGroup(
        jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addContainerGap()
```

```
            .addComponent(receptionIcon)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
            .addComponent(receptionLabel)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );
        jPanel3.setBackground(java.awt.Color.lightGray);
        doctorIcon.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/Icon/doctor.png"))); // NOI18N
        doctorIcon.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
                doctorIconMouseClicked(evt);
            }
        });
        doctorLabel.setFont(new java.awt.Font("Arial", 0, 18)); // NOI18N
        doctorLabel.setForeground(new java.awt.Color(255, 255, 255));
        doctorLabel.setText("Doctor");
        doctorLabel.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
                doctorLabelMouseClicked(evt);
            }
        });
        javax.swing.GroupLayout jPanel3Layout = new javax.swing.GroupLayout(jPanel3);
        jPanel3.setLayout(jPanel3Layout);
        jPanel3Layout.setHorizontalGroup(
```

```
            jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel3Layout.createSequentialGroup()
                .addGap(51, 51, 51)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
                    .addComponent(doctorLabel)
                    .addComponent(doctorIcon))
                .addContainerGap(46, Short.MAX_VALUE))
        );
        jPanel3Layout.setVerticalGroup(
            jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel3Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(doctorIcon)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(doctorLabel)
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );
        jPanel4.setBackground(new java.awt.Color(0, 204, 204));
        jLabel1.setFont(new java.awt.Font("Arial", 0, 24)); // NOI18N
        jLabel1.setForeground(new java.awt.Color(255, 255, 255));
        jLabel1.setText("Login As");
        javax.swing.GroupLayout jPanel4Layout = new javax.swing.GroupLayout(jPanel4);
```

```
        jPanel4.setLayout(jPanel4Layout);
        jPanel4Layout.setHorizontalGroup(
            jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel4Layout.createSequentialGroup()
                .addContainerGap(88, Short.MAX_VALUE)
                .addComponent(jLabel1)
                .addContainerGap())
        );
        jPanel4Layout.setVerticalGroup(
            jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel4Layout.createSequentialGroup()
                .addContainerGap(23, Short.MAX_VALUE)
                .addComponent(jLabel1)
                .addGap(20, 20, 20))
        );
        jPanel5.setBackground(new java.awt.Color(0, 102, 102));
        jLabel2.setFont(new java.awt.Font("Arial", 1, 30)); // NOI18N
        jLabel2.setForeground(new java.awt.Color(255, 255, 255));
        jLabel2.setText("Welcome");
        jLabel3.setFont(new java.awt.Font("Tahoma", 0, 24)); // NOI18N
        jLabel3.setForeground(new java.awt.Color(255, 255, 255));
        jLabel3.setText("to");
        jLabel4.setFont(new java.awt.Font("Arial Narrow", 0, 30)); // NOI18N
```

```
        jLabel4.setForeground(new java.awt.Color(255, 255, 255));
        jLabel4.setText("My Hospital");
        jLabel4.setToolTipText("");
        javax.swing.GroupLayout jPanel5Layout = new javax.swing.GroupLayout(jPanel5);
        jPanel5.setLayout(jPanel5Layout);
        jPanel5Layout.setHorizontalGroup(
            jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel5Layout.createSequentialGroup()

.addGroup(jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
                    .addGroup(jPanel5Layout.createSequentialGroup()
                        .addGap(88, 88, 88)
                        .addComponent(jLabel3))
                    .addGroup(jPanel5Layout.createSequentialGroup()
                        .addGap(22, 22, 22)
                        .addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE, 157,
javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addGroup(jPanel5Layout.createSequentialGroup()
                        .addGap(35, 35, 35)
                        .addComponent(jLabel2)))
                .addContainerGap(33, Short.MAX_VALUE))
        );
        jPanel5Layout.setVerticalGroup(
            jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
            .addGroup(jPanel5Layout.createSequentialGroup()
                .addGap(67, 67, 67)
                .addComponent(jLabel2)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jLabel3)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jLabel4, javax.swing.GroupLayout.PREFERRED_SIZE, 28,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(96, Short.MAX_VALUE))
        );
        javax.swing.GroupLayout MainFLayout = new javax.swing.GroupLayout(MainF);
        MainF.setLayout(MainFLayout);
        MainFLayout.setHorizontalGroup(
            MainFLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(mLabel, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addGroup(MainFLayout.createSequentialGroup()
                .addComponent(jPanel5, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(48, 48, 48)
                .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(100, 100, 100)
```

```
            .addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 100,
Short.MAX_VALUE)

            .addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addGap(99, 99, 99))

        .addGroup(MainFLayout.createSequentialGroup()

            .addComponent(jPanel4, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))

    );

    MainFLayout.setVerticalGroup(

        MainFLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(MainFLayout.createSequentialGroup()

            .addGap(35, 35, 35)

            .addComponent(mLabel, javax.swing.GroupLayout.PREFERRED_SIZE, 70,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addGap(39, 39, 39)

            .addComponent(jPanel4, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addGroup(MainFLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADIN
G)
            .addGroup(MainFLayout.createSequentialGroup()
              .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
              .addComponent(jPanel5, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(MainFLayout.createSequentialGroup()
              .addGap(30, 30, 30)

.addGroup(MainFLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADIN
G)
                .addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))))
          .addContainerGap(29, Short.MAX_VALUE))
    );
    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
      layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

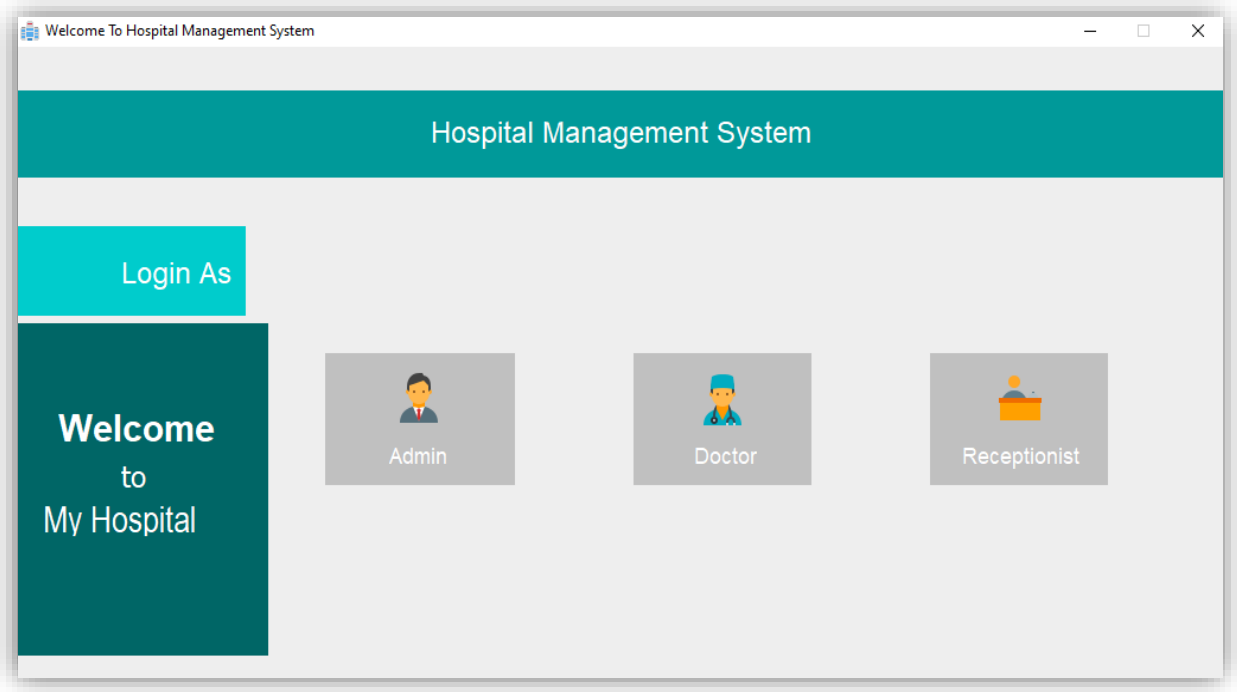
```
            .addComponent(MainF, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );
        layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                .addGap(0, 0, Short.MAX_VALUE)
                .addComponent(MainF, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        );
        pack();
        setLocationRelativeTo(null);
    }// </editor-fold>
    private void adminLabelMouseClicked(java.awt.event.MouseEvent evt) {
        Admin ad = new Admin();
        ad.setVisible(true);
        dispose();
    }
    private void doctorLabelMouseClicked(java.awt.event.MouseEvent evt) {
        Doctor doctor = new Doctor();
        doctor.setVisible(true);
        dispose();
    }
```

```
private void receptionLabelMouseClicked(java.awt.event.MouseEvent evt) {
    Receptionist receptionist = new Receptionist();
    receptionist.setVisible(true);
    dispose();
}
private void adminIconMouseClicked(java.awt.event.MouseEvent evt) {
    Admin ad = new Admin();
    ad.setVisible(true);
    dispose();
}
private void doctorIconMouseClicked(java.awt.event.MouseEvent evt) {
    Doctor doctor = new Doctor();
    doctor.setVisible(true);
    dispose();
}
private void receptionIconMouseClicked(java.awt.event.MouseEvent evt) {
    Receptionist receptionist = new Receptionist();
    receptionist.setVisible(true);
    dispose();
}
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(() -> {
        new Hospital().setVisible(true);
    });
```

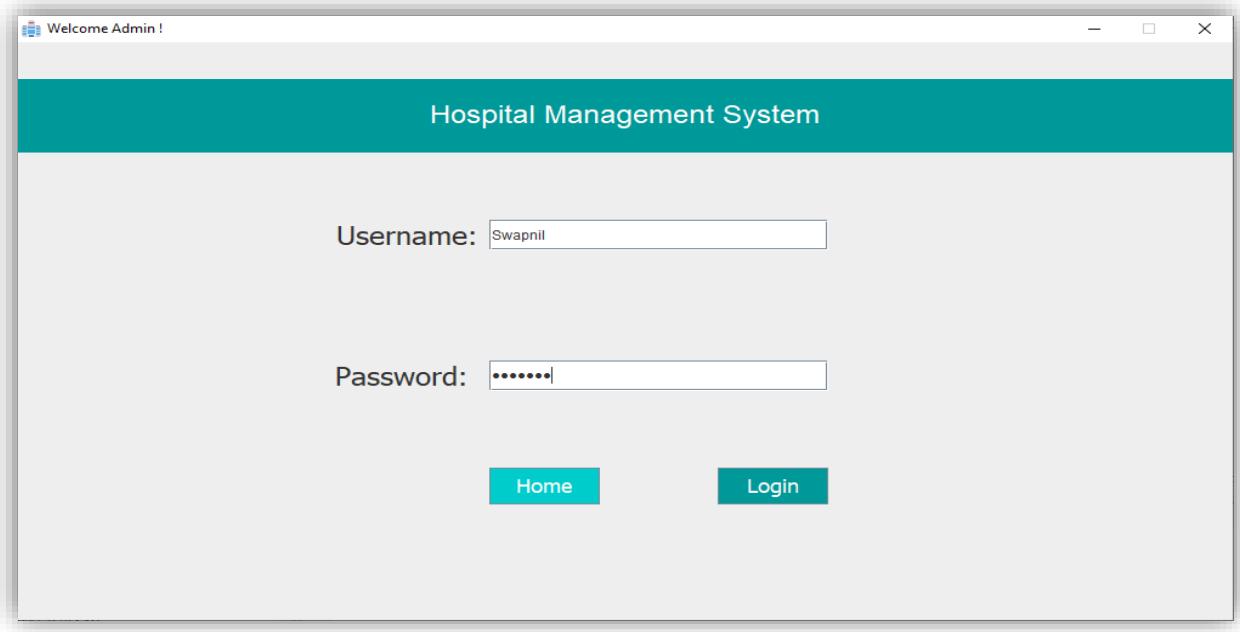
```
    }
    // Variables declaration - do not modify
    private javax.swing.JPanel MainF;
    private javax.swing.JLabel adminIcon;
    private javax.swing.JLabel adminLabel;
    private javax.swing.JLabel doctorIcon;
    private javax.swing.JLabel doctorLabel;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPanel jPanel2;
    private javax.swing.JPanel jPanel3;
    private javax.swing.JPanel jPanel4;
    private javax.swing.JPanel jPanel5;
    private java.awt.Label mLabel;
    private javax.swing.JLabel receptionIcon;
    private javax.swing.JLabel receptionLabel;
    // End of variables declaration
}
```

8. SAMPLE SCREENSHOT

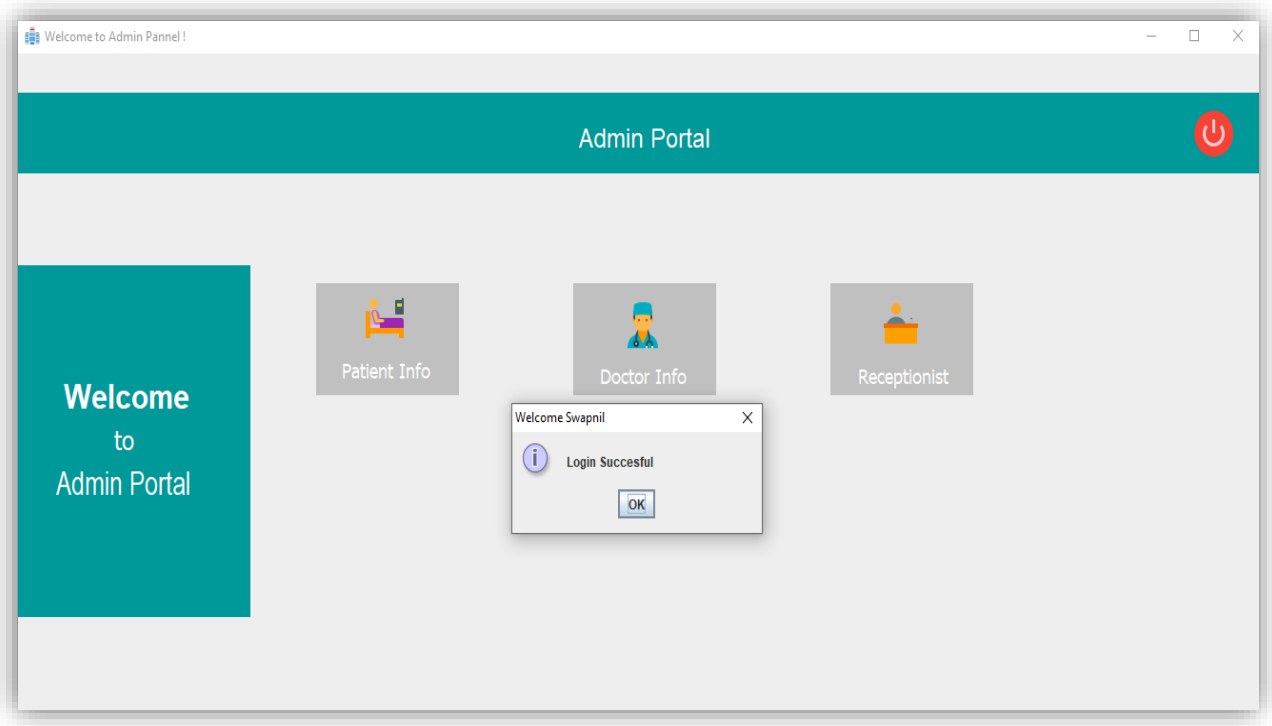
8.1 Main screen:



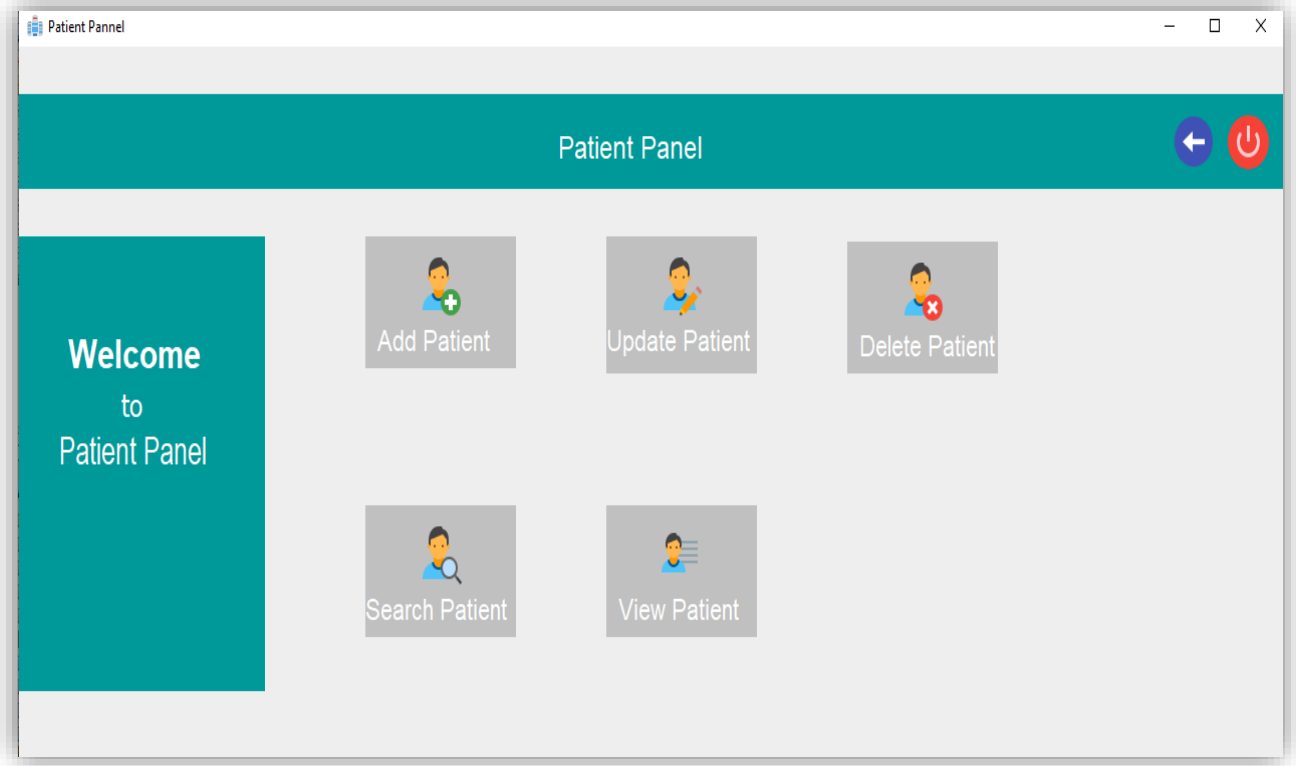
8.2 Admin login:



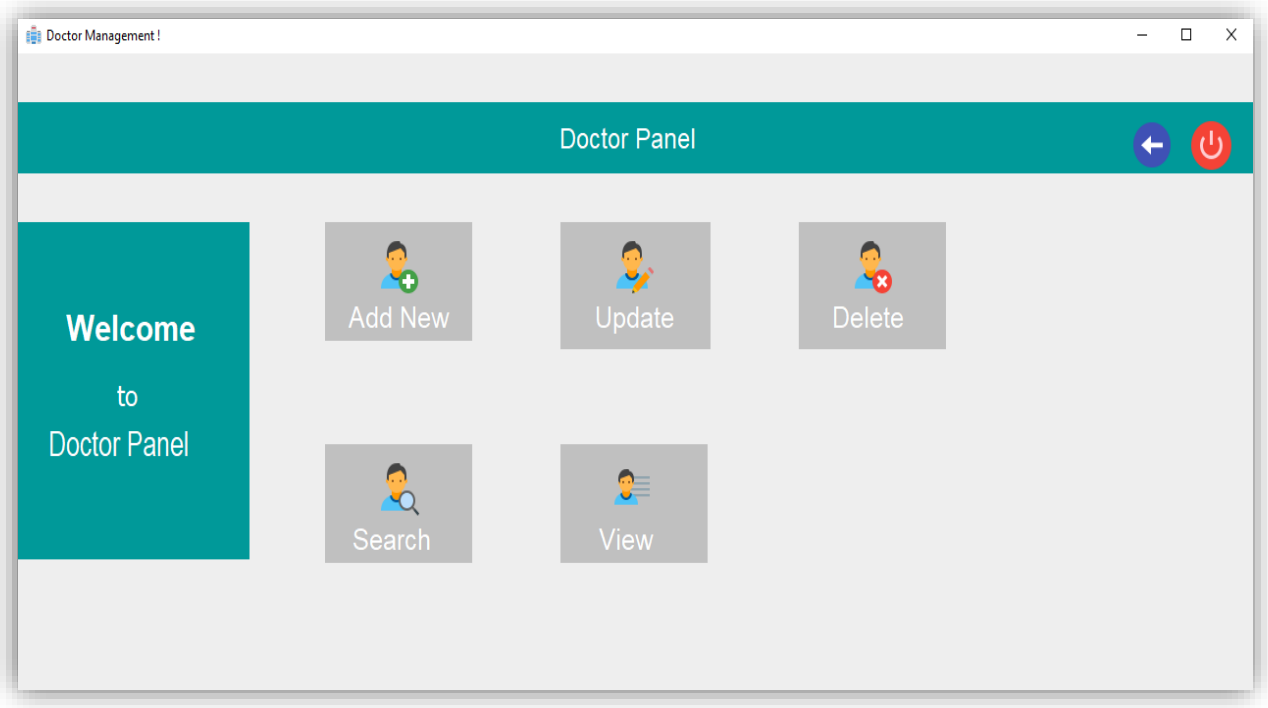
8.2.1 Admin portal:



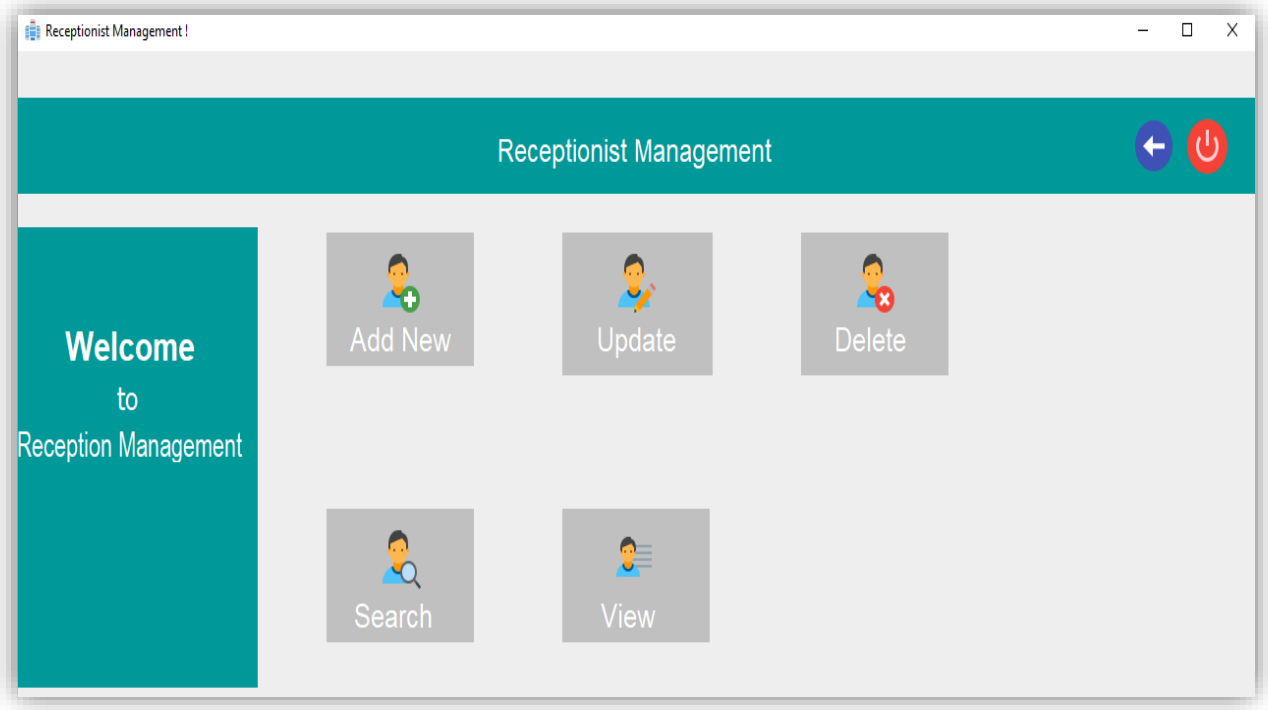
8.2.2 Patient panel:



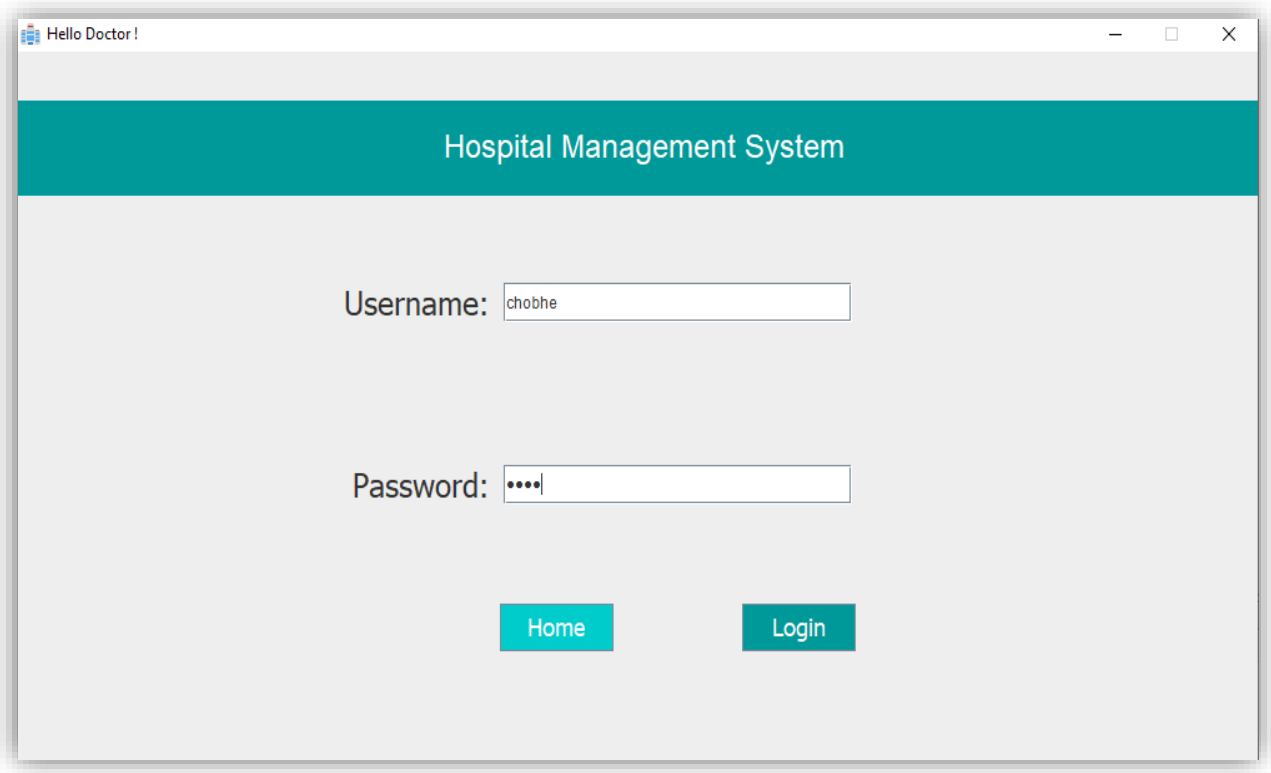
8.2.3 Doctor panel:



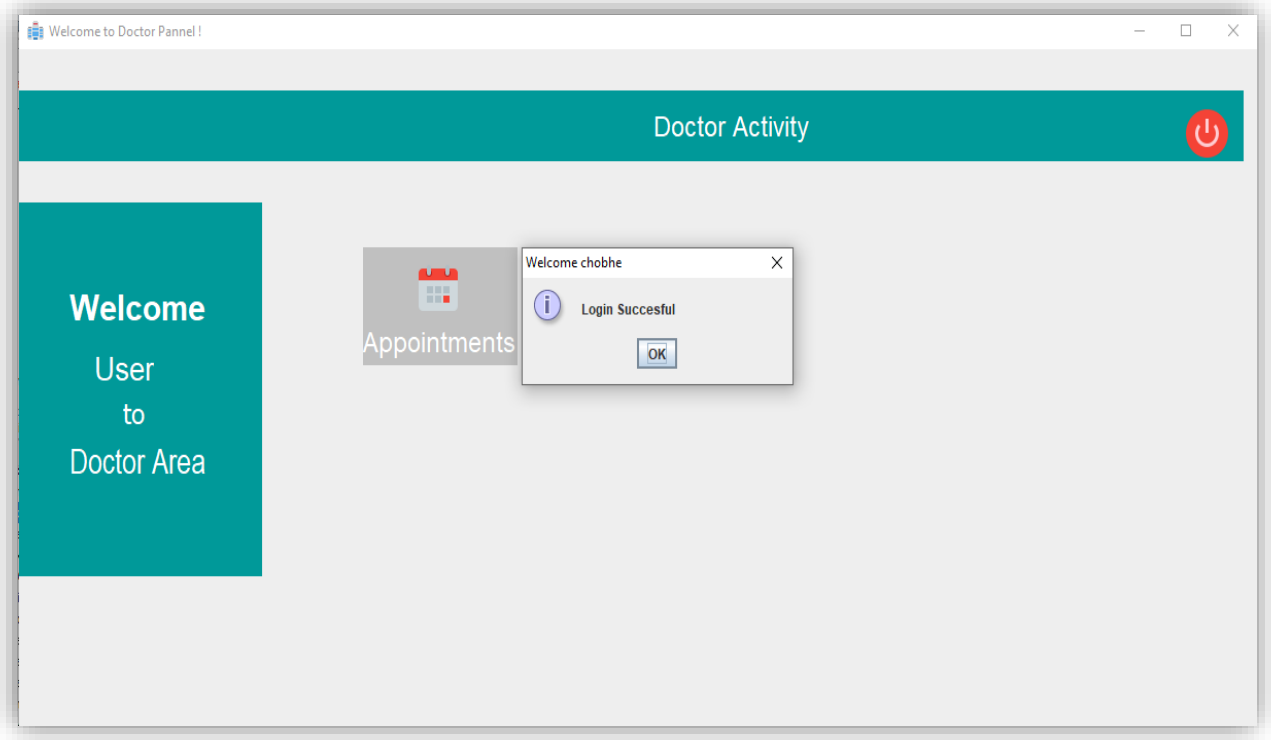
8.2.4 Receptionist management:



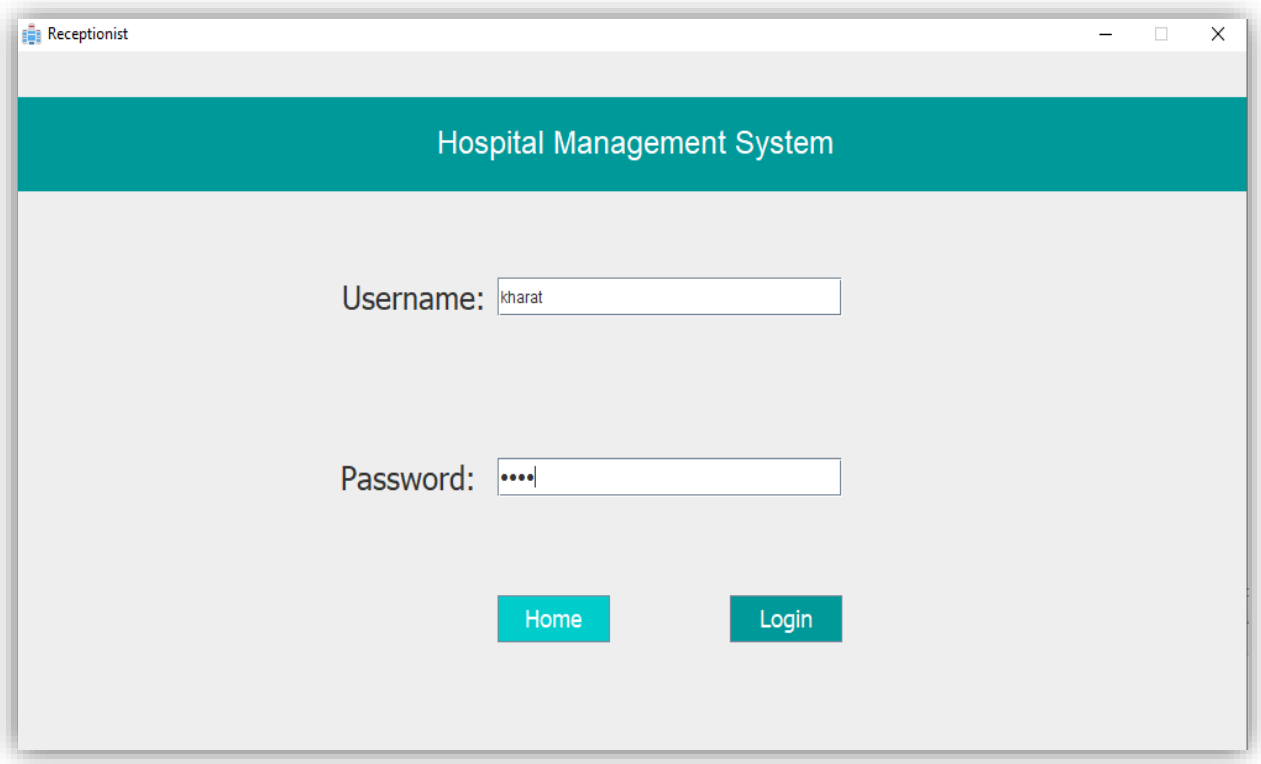
8.3 Doctor login:



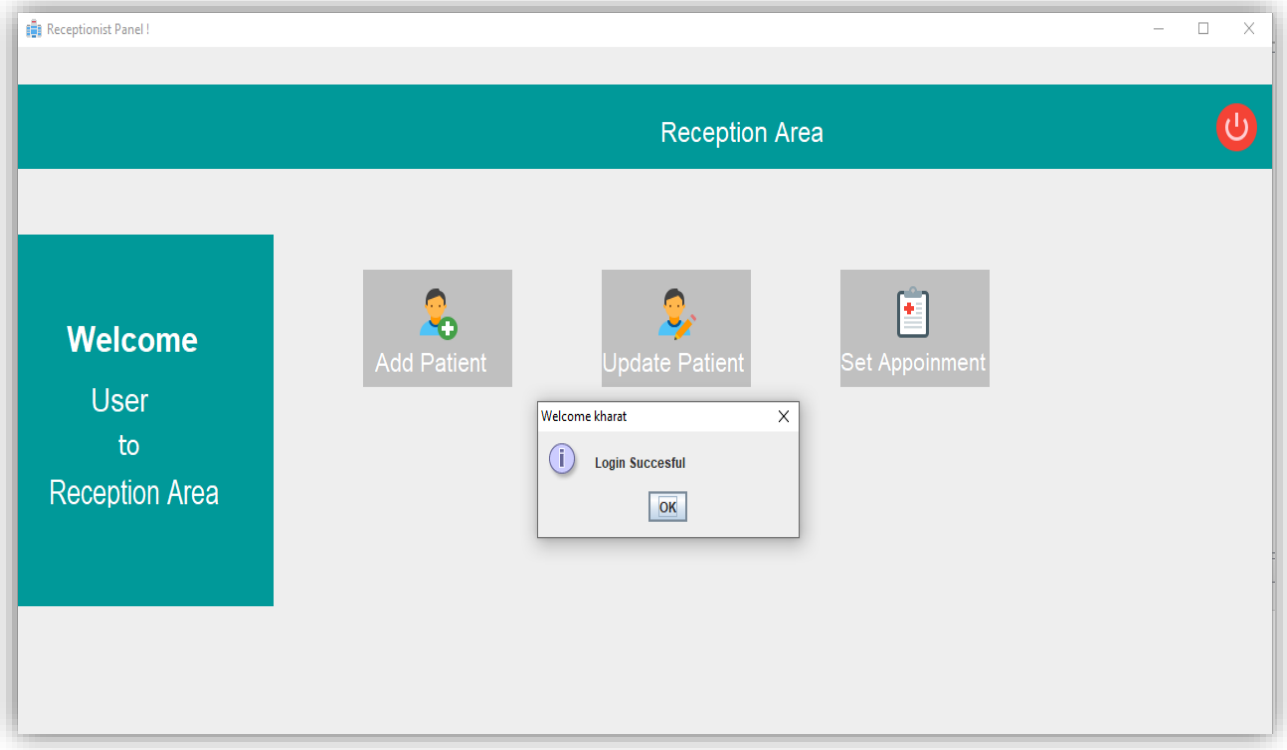
8.3.1 Doctor activity:



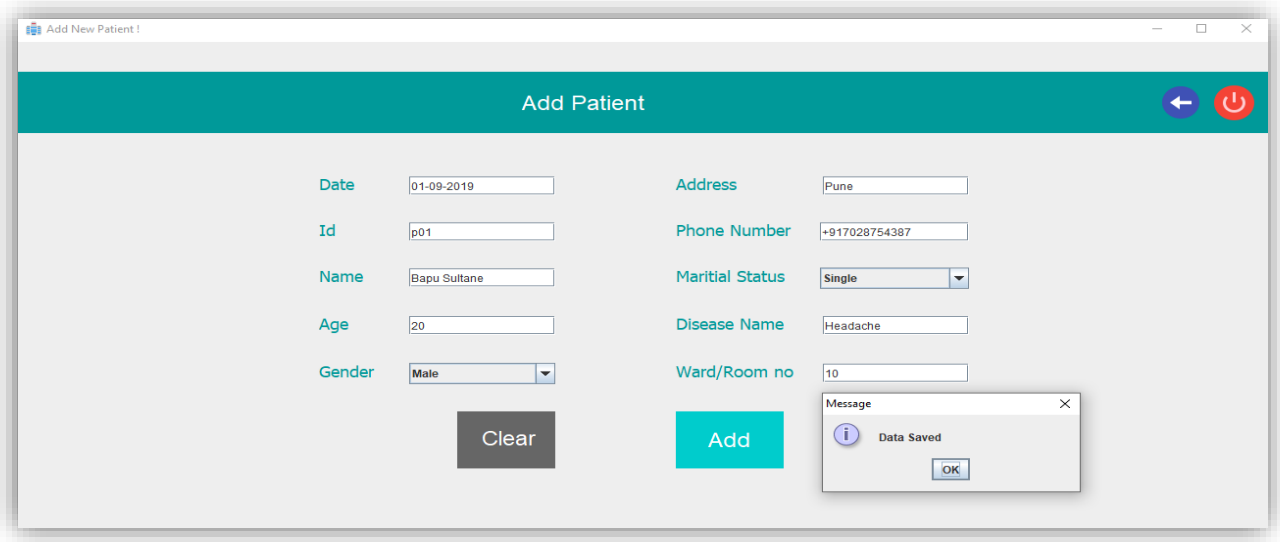
8.4 Receptionist login:



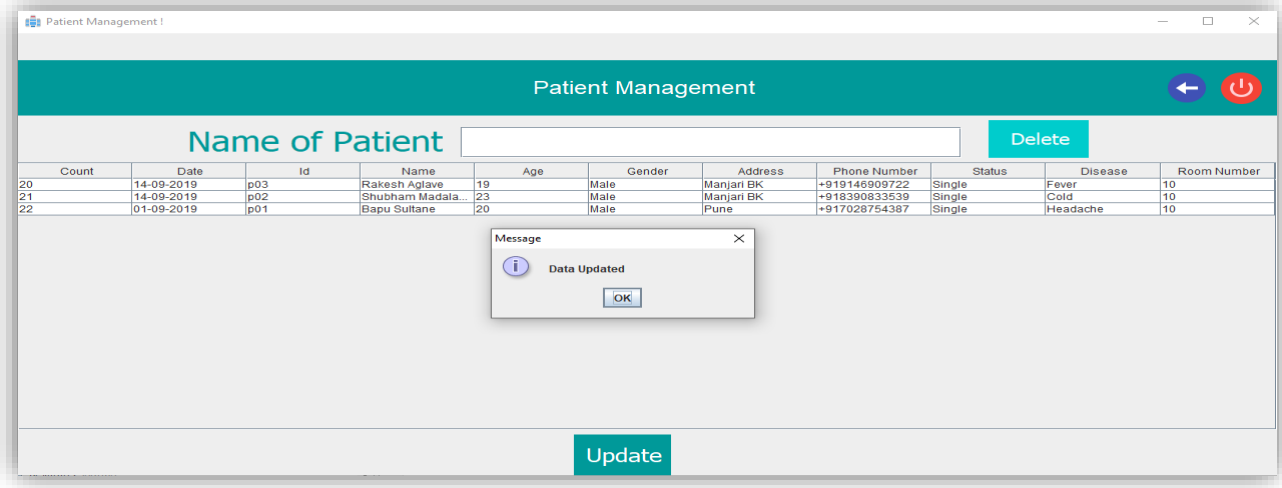
8.4.1 Reception area:



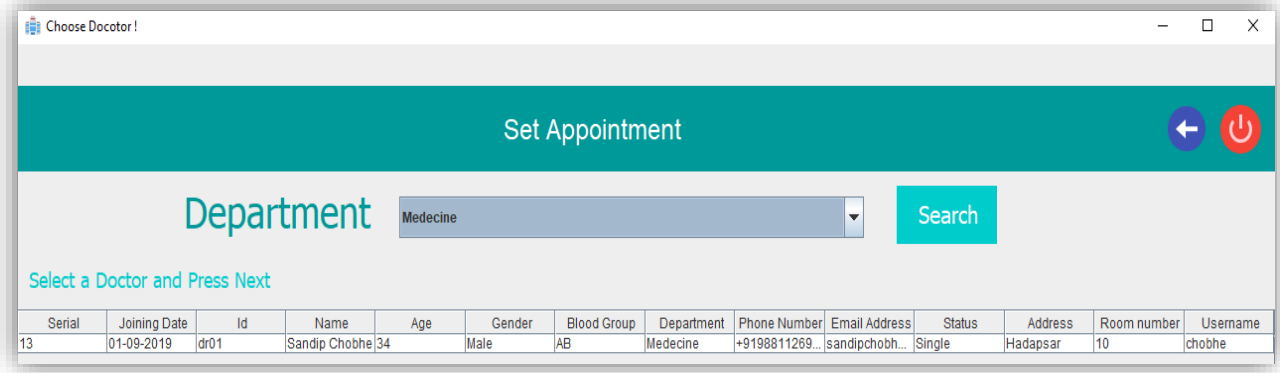
8.4.2 Add patient:



8.4.3 Update patient:



8.4.4 Set appointment:



9. CONCLUSION

Since we are entering details of the patients electronically in the” Hospital Management System”, data will be secured. Using this application, we can retrieve patient’s history with a single click. Thus, processing information will be faster. It guarantees accurate maintenance of Patient details. It easily reduces the book keeping task and thus reduces the human effort and increases accuracy speed.

Hospital Management System is essential for maintaining detail about the Doctor, Patient, Hospital staff etc. we understand that by using of Hospital Management System project the work became very easy and we save lot of time. Hospital administrators would be able to significantly improve the operational control and thus streamline operations. This would enable to improve the response time to the demands of patient care because it automates the process of collecting, collating and retrieving patient information. Accounting sometimes becomes awfully pathetic and complex. This product will eliminate any such complexity.

REFERENCES

- Binstock, Andrew (May 20, 2015). **"Java's 20 Years of Innovation"**. Forbes. **Archived** from the original on March 14, 2016. Retrieved March 18, 2016.
- Herbert Scheldt, **Java Complete Reference,** Fifth Edition, Tata McGraw Hill Edition.
- Gosling, James; Joy, Bill; Steele, Guy; Bracha, Gilad. **"The Java Language Specification, 2nd Edition"**. **Archived** from the original on August 5, 2011. Retrieved February 8, 2008.
- **"MySQL 8.0 Release Notes"**. mysql.com. Retrieved 29 July 2019.
- https://en.wikipedia.org/wiki/NetBeans
- https://www.tutorialspoint.com/mysql/index.htm
- https://www.w3schools.com/java/java_intro.asp