Project Task: Hospital Management System

Applications and software comprise of computer or web-based software facilitating management/ functions within in the hospital or any medical company. Basically, it is likely alternative paper-records which are difficult to manage and get information. These software integrate important information from people involved such as doctors, patients and administration staff.

1.Who are the User of an HMS

+ Hospital Administration: People who fall under this category belong to Finance, Accounts, etc. These people should only access basic information about any patient.

+Doctors and employees: Doctors, officers, Nurses, Surgeons, and Laboratory staff deal with more critical and detailed patient information. Doctors and other paramedical staff are the main audiences of an HMS.

+Patients: A patient generally uses the HMS to book appointments with doctors and to check the laboratory test results. A Hospital Management System also helps to speed up the communication and exchange of information between the patient and the hospital personnel.

Basic Components of Hospital Management System

It usually includes some basic hospital management system modules. The number of them can depend on the clinic needs.

+ Appointment and Schedule: This module in hospital management arranges the schedule of doctors due to the patients' application. It helps to organize the availability of medical specialists at any convenient time.

+ Facility Management: This function is responsible for tracking and maintaining the room availability, the occupancy status as well as various kind of medical equipment, or documentary.

+ **Staff Management:** Staff management module provides the human resources administration. It updates the job description of employees, updates the hospital structure, tracks the recruiting records.

+ Accounting: This function organizes information affairs both customer and the medical institution. With the patients, it stores and display all patient's payment detail and medical history. With the administration staff, it will present the hospital financial records on expense and overall profit.

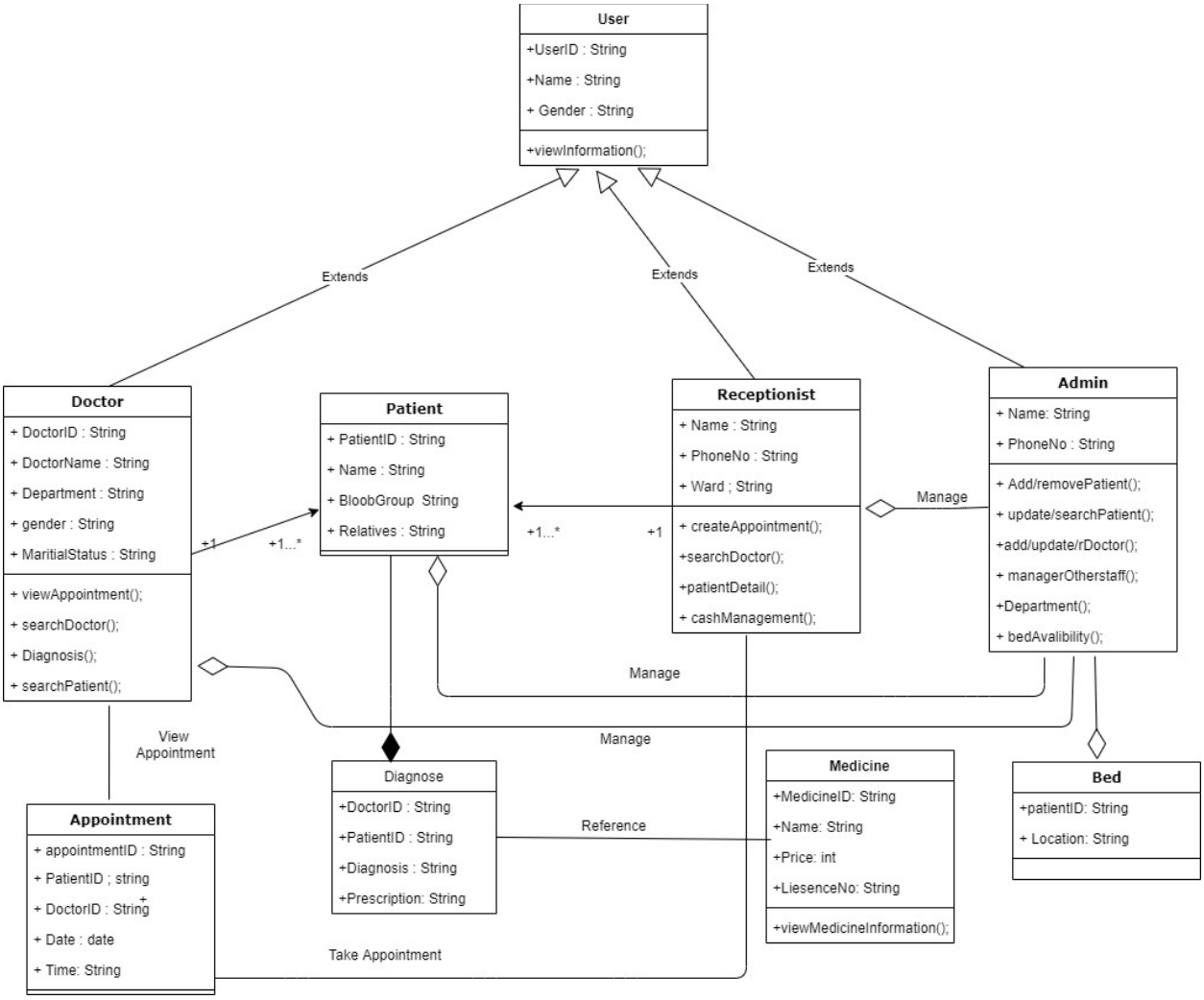
+ **Insurance services integration:** This module can store user's insurance detail such as policy, insurance company, or grace time of it. It helps users register or cancel services of hospital easily and immediately.

+ Medicine Management: Medicine management module contains the list of drugs that usually used for the specific treatments. It keeps records of every patient's drugs used during the treatment.

3. Hospital management system UML diagram

Class diagram shows a set of classes, interface, and collaborations and their relationship. They are important for visualizing, specifying, and documenting structural models and also for constructing executable systems through forward and reverse engineering.

In class diagram of hospital management there are many classes like Patient, Doctor, Receptionist, etc. The relationship among classes are association, generalization, composition, aggregation.



4. Modules

+ **Admin**: admin will log in the system with the user “admin” and password “admin”. With this system, admin can manage all of the users such as Doctor, Receptionist, Pharmacy…These functions is add/ update/ remove/search users and manage bed. There are some sample code for this module.

```
private void browserActionPerformed(java.awt.event.ActionEvent evt) {
    JFileChooser fileChooser = new JFileChooser();
    fileChooser.setCurrentDirectory(new File(System.getProperty("user.home")));
    FileNameExtensionFilter filter = new FileNameExtensionFilter("*.IMAGE", "jpg", "gif", "png");
    fileChooser.addChoosableFileFilter(filter);
    int result = fileChooser.showSaveDialog(null);
    if (result == JFileChooser.APPROVE_OPTION) {
        File selectedFile = fileChooser.getSelectedFile();
        String path = selectedFile.getAbsolutePath();
        doctorImage.setIcon(ResizeImage(path));
        s = path;
    } else if (result == JFileChooser.CANCEL_OPTION) {
        System.out.println("No Data");
    }
    //convert Image to byte[] array
    try {
        File image = new File(s);
        FileInputStream fis = new FileInputStream(image);
        ByteArrayOutputStream bos = new ByteArrayOutputStream();
        byte[] buf = new byte[1024];
        for (int readNum; (readNum = fis.read(buf)) != -1;) {
            bos.write(buf, 0, readNum);
        }
        person_image = bos.toByteArray();
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, e);
    }

}
```

This code allow user to upload the image from the devices.

```
public ImageIcon ResizeImage(String imgPath) {
        ImageIcon MyImage = new ImageIcon(imgPath);
        Image img = MyImage.getImage();
        Image newImage = img.getScaledInstance(doctorImage.getWidth(),
                doctorImage.getHeight(), Image.SCALE_SMOOTH);
        ImageIcon image = new ImageIcon(newImage);
        return image;
    }
}
```

Resize the avatar.

```
conn = db.getConnection();
PreparedStatement pst2 = conn.prepareStatement("Select*from Doctor"
        + " Where DoctorID =?");
pst2.setString(1, DocID.getText());
ResultSet rs = pst2.executeQuery();
```

Search doctor.

+ Doctor: In the doctor's interface, doctor can view appointments and search them. Then Doctor can search the other Doctors and patients. With a patient, Doctor can store the diagnosis of them and search the diagnosis to understand the patient's disease and prescription**.**

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        conn = db.getConnection();
        PreparedStatement pst = conn.prepareStatement("Insert into diagnose "
                + "(PatientID,DoctorID,Diagnose,Date,Prescription,DoctorName)"
                + "values(?,?,?,?,?,?)");
        pst.setString(1, pid.getText());
        pst.setString(2, did.getText());
        pst.setString(3, dia.getText());
        pst.setDate(4, convertFromJAVADateToSQLDate(datel.getDate()));
        pst.setString(5, pre.getText());
        pst.setString(6, name.getText());
        pst.execute();
        JOptionPane.showMessageDialog(this, "Save Successfully");

    } catch (Exception ex) {
        Logger.getLogger(Diagnosel.class.getName()).log(Level.SEVERE, null, ex);
    }

}
```

Insert the Diagnosis.

```
try {
    // TODO add your handling code here:
    conn = db.getConnection();
    PreparedStatement pst = conn.prepareStatement("Select Date from "
            + "diagnose where PatientID= ?");
    pst.setString(1, pid.getText());
    ResultSet rs = pst.executeQuery();
    while (rs.next()) {
        date.addItem(convertStringToDate(rs.getDate("Date")));
    }

} catch (Exception ex) {
    Logger.getLogger(Diagnosel.class.getName()).log(Level.SEVERE, null, ex);
```

Choose the previous diagnosis depending on the time of these.

+ **Receptionist**: The receptionist can search the Doctor, search/ update patients and create the appointment.

```
private boolean check() throws SQLException, Exception{
    int count =0;
    conn= db.getConnection();
    PreparedStatement pst = conn.prepareStatement("Select* from Appointment where DoctorID=? and Hour =?");
    pst.setString(1, DocID());
    pst.setString(2, (String) timing.getSelectedItem());

    ResultSet rs = pst.executeQuery();
    while(rs.next()){
        count++;
    }
    return count == 0;

}
```

Check Appointment ID.

```
public void showTableData(){
    List<Doctor> doctorList = DoctorModify.findAll();
    tableModel.setRowCount(0);
    doctorList.forEach(doctor -> {
        tableModel.addRow(new Object[]{doctor.getDoctorID(),doctor.getName(),
            doctor.getDepartment(),doctor.getVisitTiming(),doctor.getDays()});
    });

}
```

Show Doctor's Information in the table.

```
  try {
  // TODO add your handling code here:
  conn = db.getConnection();
  PreparedStatement pst = conn.prepareStatement("Insert into Appointment "
          + "(DoctorID, PatientID,AppointmentID,Date,Hour)"
          + "values(?,?,?,?,?)");
  pst.setString(1, docl.getText());
  pst.setString(2, id.getText());
  pst.setString(3, aID.getText());
  pst.setDate(4, convertFromJAVADateToSQLDate(date.getDate()));
  pst.setString(5, (String) timing.getSelectedItem());
  pst.execute();
  JOptionPane.showMessageDialog(this, "Save sucessfully");
catch (Exception ex) {
  Logger.getLogger(createAppointment.class.getName()).log(Level.SEVERE, null, ex);
```

Saving the appointment.

+ **Pharmacy:** pharmacy can view patient's prescriptions. Moreover, the pharmacy can search the medicine information by medicine's name or medicine ID. To do this, we catch the textField event (key release).

```
private void jTextField1KeyReleased(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    String searchl=jTextFieldl.getText();
    TableRowSorter<DefaultTableModel> tr=new TableRowSorter<DefaultTableModel>(tableModel);
    table.setRowSorter(tr);
    tr.setRowFilter(RowFilter.regexFilter(searchl));
}
```

Search medicine.

```
private void showTable(){
    try {
        conn = db.getConnection();
        PreparedStatement pst = conn.prepareStatement("Select*from medicine");
        ResultSet rs = pst.executeQuery();
        tableModel.setRowCount(0);
        while(rs.next()){
            tableModel.addRow(new Object[]{
               rs.getString("MedicineID"),rs.getString("Name"),rs.getString("LiesenceNo"),rs.getString("Price")
            });

        }
    } catch (Exception ex) {
        Logger.getLogger(DoctorActivity.class.getName()).log(Level.SEVERE, null, ex);
    }
}
```

Show the medicine information.

5. Difficulties about Data types

Because, we do not have a lot of knowledge about Data types such as String and Date, SQL date and Java date, Image and Bytes. Thanks to the Internet, we used some methods to convert them.

```
public String convertStringToDate(Date indate) {
    String dateString = null;
    SimpleDateFormat sdfr = new SimpleDateFormat("dd/MMM/yyyy");
    try {
        dateString = sdfr.format(indate);
    } catch (Exception ex) {
        System.out.println(ex);
    }
    return dateString;
}
```

Convert Date to String.

```
public static java.sql.Date convertFromJAVADateToSQLDate(
        java.util.Date javaDate) {
    java.sql.Date sqlDate = null;
    if (javaDate != null) {
        sqlDate = new java.sql.Date(javaDate.getTime());
    }
    return sqlDate;
}
```

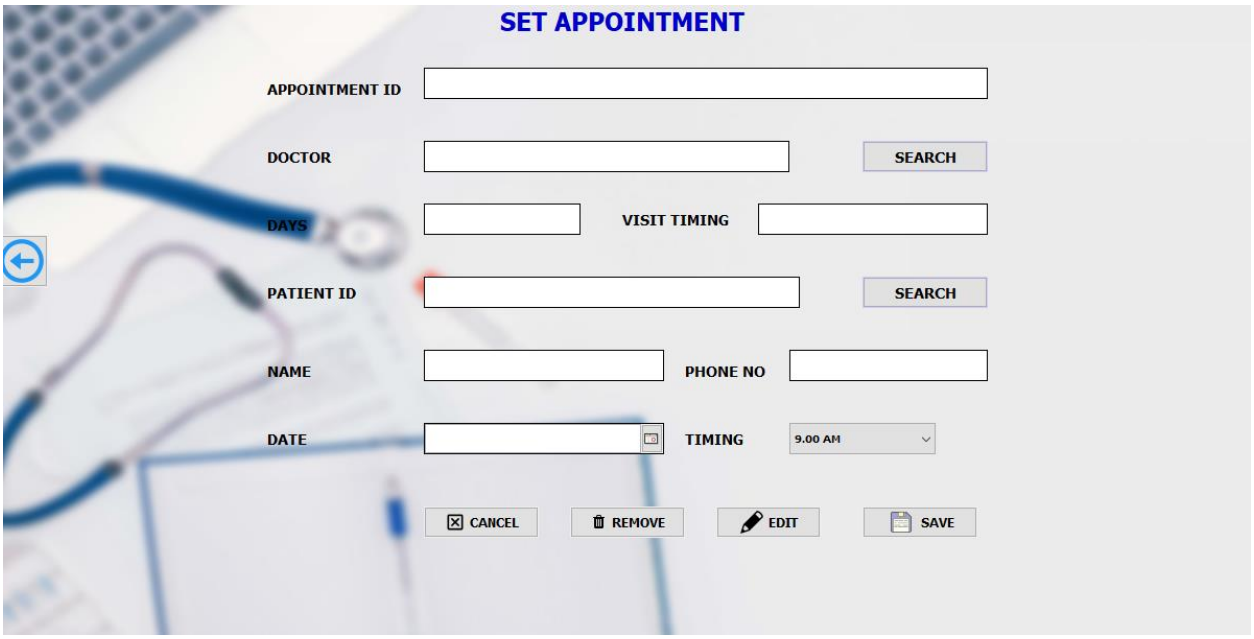
Convert Java Date to SQL Date.

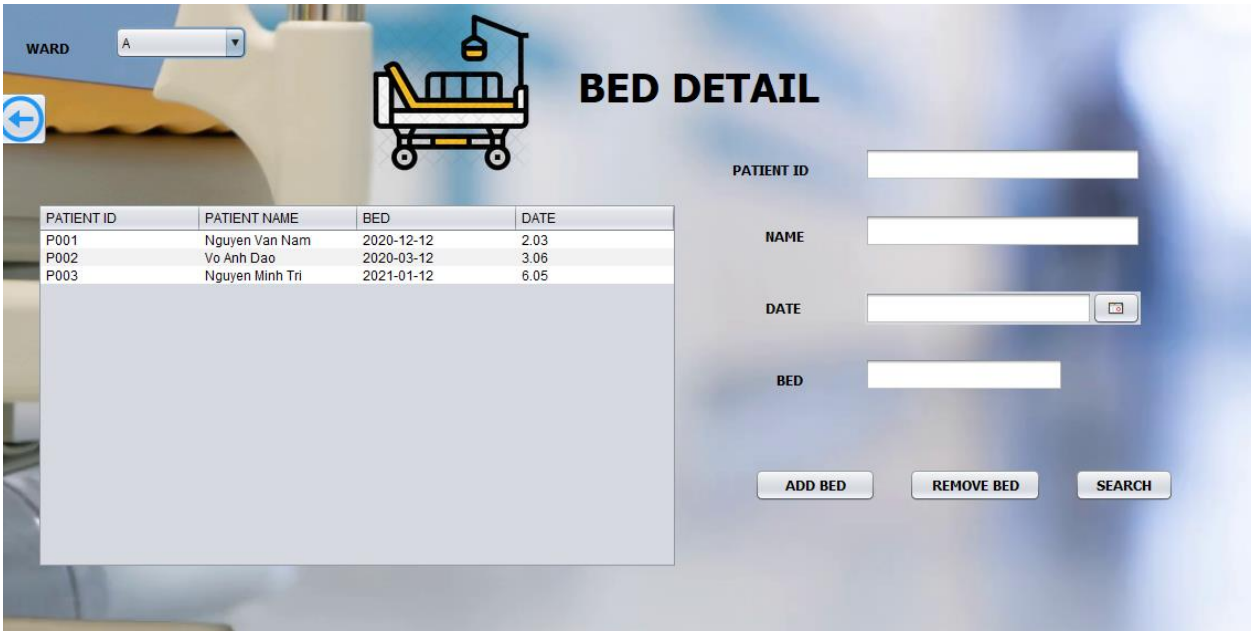
```
try {
    File image = new File(s);
    FileInputStream fis = new FileInputStream(image);
    ByteArrayOutputStream bos = new ByteArrayOutputStream();
    byte[] buf = new byte[1024];
    for (int readNum; (readNum = fis.read(buf)) != -1;) {
        bos.write(buf, 0, readNum);
    }
    person_image = bos.toByteArray();
} catch (Exception e) {
    JOptionPane.showMessageDialog(null, e);
}
```

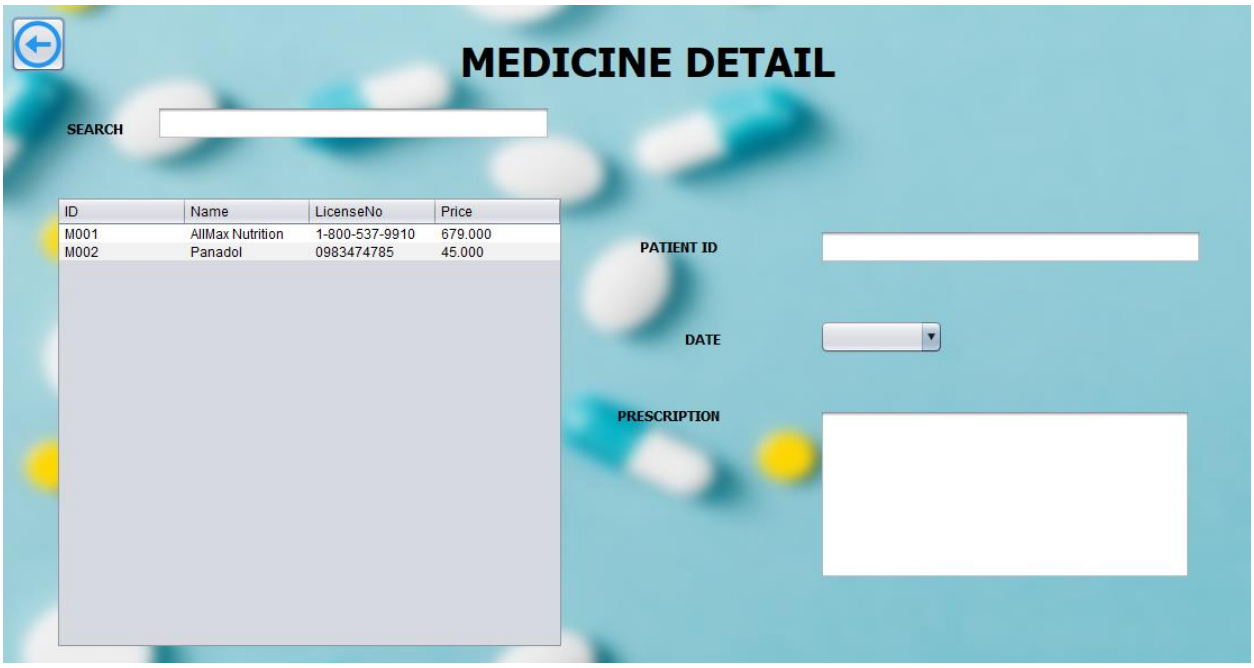
Convert Image to Bytes array.

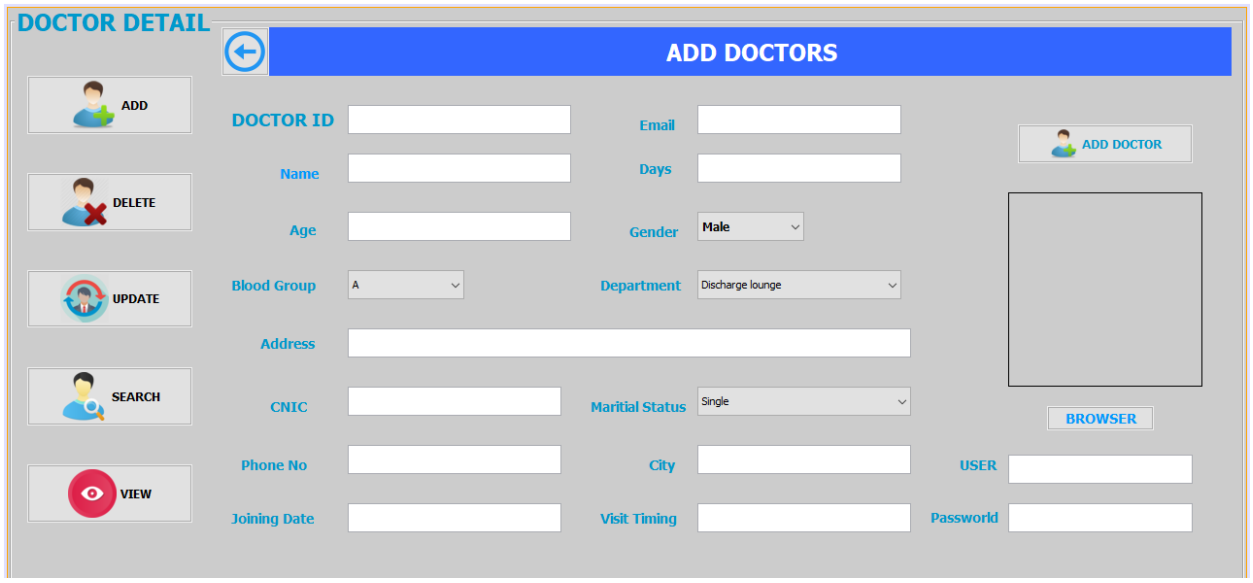
6. The other images about this project.

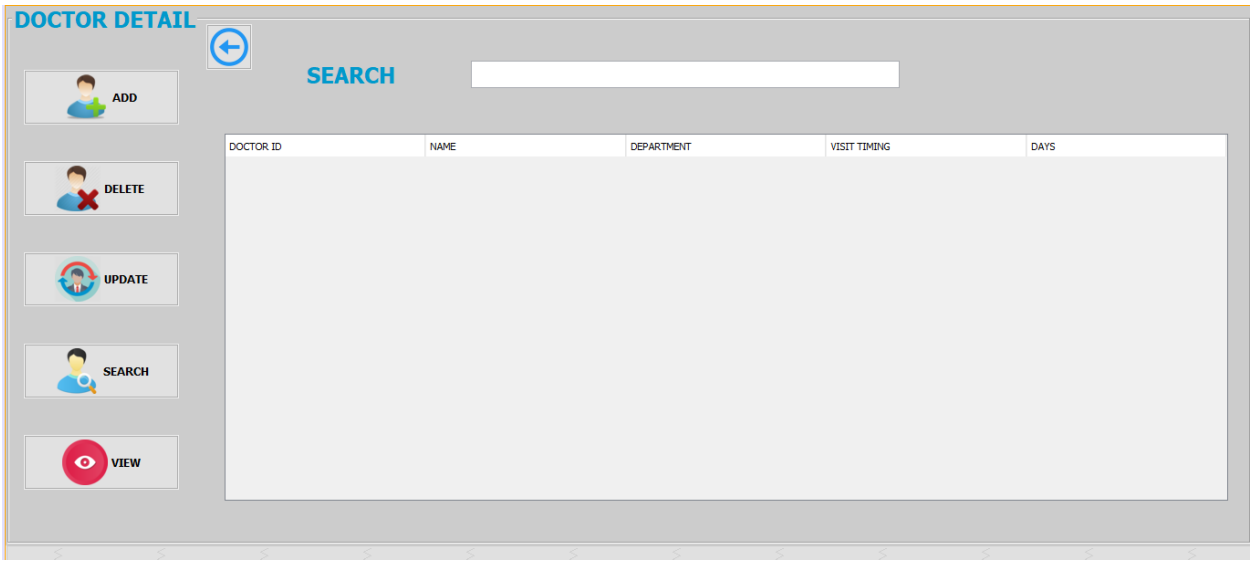

Hospital Management System
Admin Panel
DOCTOR DETAIL
LOWERSTAFF DETAIL
RECEPTIONIST DETAIL
PATIENT DETAIL
BED AVALIBILITY
Department
DEPARTMENT


SET APPOINTMENT
APPOINTMENT ID
DOCTOR
SEARCH
DAYS
VISIT TIMING
PATIENT ID
SEARCH
NAME
PHONE NO
DATE
TIMING
9.00 AM
CANCEL
REMOVE
EDIT
SAVE


WARD
A
BED DETAIL
PATIENT ID
NAME
DATE
BED
ADD BED
REMOVE BED
SEARCH
PATIENT ID
PATIENT NAME
BED
DATE
P001
Nguyen Van Nam
2020-12-12
2.03
P002
Vo Anh Dao
2020-03-12
3.06
P003
Nguyen Minh Tri
2021-01-12
6.05


MEDICINE DETAIL
SEARCH
PATIENT ID
DATE
PRESCRIPTION
ID
Name
LicenseNo
Price
M001
AllMax Nutrition
1-800-537-9910
679.000
M002
Panadol
0983474785
45.000





7. References

+ Image and Icon: Java Hub.

+ Stackoverflow.com.

+ Slideshare.com (https://www.slideshare.net/PradeepBhosale/uml-diagram-forhospitalmanagementsystem).