MINOR PROJECT ON E HEALTHCARE MANAGEMENT SYSTEM

A minor project report for the evaluation and partial fulfilment of the requirement for the award of degree

Master in Computer Application in Artificial Intelligence

Submitted By:
Sumit Kumar(225/PCA/014)
Ritik Chauhan(225/PCA/022)
Pankaj Kumar(225/PCA/008)

Under the Supervision:
Mr. Manish Kumar Singh

UNIVERSITY SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY, GAUTAM BUDDHA UNIVERSITY, GREATER NOIDA-201312, UTTAR PRADESH INDIA

NOVEMBER,2023

ACKNOWLEDGEMENT

In the journey of exploring the intricate realms of “E Healthcare Management System using Java", We are deeply indebted to us supervisor, **Mr. Manish Kumar Singh**. His unwavering guidance and mentorship have been the compass that guided me through the java concepts. Sir, your expertise and support have been invaluable, and I am sincerely grateful for the effort you invested in enriching our understanding.

We extend us heartfelt appreciation to java for providing comprehensive and user-friendly documentation. This has been a beacon of knowledge, especially for students who may face challenges but aspire to learn and enhance their skills. The accessibility and clarity in the documentation have been pivotal in shaping my understanding of java core concepts.

A special acknowledgment goes to the creators of the course that has been nothing short of excellent. The course's global availability has opened doors for countless learners worldwide, making complex concepts accessible through an engaging and interesting approach. Thank you for fostering an environment that encourages learning and growth.

Lastly, I want to express my gratitude to my team. Their unwavering support and encouragement have been the pillars of strength throughout this endeavor. Together, we have faced challenges, celebrated victories, and fostered a collaborative spirit that has made this learning experience even more enriching.

Thank you, one and all, for contributing to this journey of knowledge and growth. Your support has been instrumental, and I am truly fortunate to have such incredible individuals alongside us.

SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

GAUTAM BUDDHA UNIVERSITY, GREATER NOIDA, 201312, U. P., (INDIA)

Candidate's Declaration

We, hereby, certify that the work embodied in this project report entitled "E Healthcare Management System using Java " in partial fulfilment of the requirements for the award of the Degree of MCA (AI) submitted to the School of Information and Communication Technology, Gautam Buddha University, Greater Noida is an authentic record of our own work carried out under the supervision of Mr. Manish Kumar Singh, School of ICT. The matter presented in this report has not been submitted in any other University / Institute for the award of any other degree or diploma. Responsibility for any plagiarism related issue stands solely with us.

Signature

Sumit Kumar	225/PCA014
Ritik Chauhan	225/PCD/022
Pankaj Kumar	225/PCD/008

This is to certify that the above statement made by the candidates is correct to the best of my knowledge and belief. However, responsibility for any plagiarism related issue solely stands with the students.

Signature of Supervisor

Mr. Manish Kumar Singh

ABSTRACT

Introduction

This project deals with the Corporate Medicare Management. This project is very helpful to both Medicare staff as well as to the public. It is having mainly Administration and Client modules.

The growing quality demand in the hospital sector makes it necessary to exploit the whole potential of stored data efficiently, not only the clinical data, in order to improve diagnoses and treatments, but also on management, in order to minimize costs and improve the care given to the patients.

This project adds the details of every individual patient and the staff appointed to them, it stores the schedule of doctors and their operation timings. It is a user-friendly system which can be used by any person. It can also store the duration of a patient till the time of discharge. It helps in the satisfaction of the user and the public itself.

Existing System Features

1. Integration of Corporate Medicare centers is very difficult while it is having different branches.
2. In most of the cases the database is similar from one hospital to another hospital. In those cases, also we can’t easily adapt a new technology in the new hospital.
3. It is very difficult to analyze the usage percentage of hospital resources, Bed occupation Ratio, Administration, Laboratory information even in a single center. Then we can expect the complexity while integrating multi-specialty Medicare Centers.
4. Room Reservations, Doctor Appointment Schedules, Operation Schedules, and Medicine indentation information is very difficult to maintain and share among the different Medicare Centers.

Lack of generic and unique model we have to implement the same set of data model for every newly established Medicare Center.

PROPOSED SYSTEM

In Medicare management situations we are dealing with Data Mining objectives such as:

1. To optimize bed occupation.
2. To improve the use of operating theatres, avoiding the cancellation of operations.
3. To know how emergencies affect to the administration of the hospital departments or services (cancellation of operations, etc).
4. To detect the influence of certain diseases in the hospital's services.
5. To find clusters of patients.

Hardware & Software Requirement Specification

OPERATING SYSTEM	:	WINDOWS 7 and more
SOFTWARE	:	NET BEAN 8.2
LANGUAGE	:	JAVA
PROCESSOR	:	Intel® Core (TM) i3 and more
PROCESSOR SPEED	:	2.4GHZ
HARD DISK	:	40GB
RAM	:	Minimum 2GB
DATA BASE	:	.DAT file

INDEX

Chapter I: Introduction

This project deals with the Corporate Medicare Management. This project is very helpful to both Medicare staff as well as to the public. It is having mainly Administration and Client modules. The growing quality demand in the hospital sector makes it necessary to exploit the whole potential of stored data efficiently, not only the clinical data, in order to improve diagnoses and treatments, but also on management, in order to minimize costs and improve the care given to the patients.

It is a process of implementing all the activities of the hospital in a computerized automated way to fasten the performance. This project is to maintain the patient details, lab reports and to calculate the bill of the patient. You can also manually edit any patient details and issue bill receipt to patient within few seconds.

This project gives the procedural approach how a patient gets treatment, details about date of treatment and finally depending on different criteria like room allocated, lab reports, treatment and medicine take etc., how billing is calculated.

This project adds the details of every individual patient and the staff appointed to them, it stores the schedule of doctors and their operation timings. It is a user friendly system which can be used by any person. It can also store the duration of a patient till the time of discharge. It helps in the satisfaction of the user and the public itself.

Chapter II: LITERATURE SURVEY

Case 1:

E Healthcare management is a growing profession with increasing opportunities in both direct and no direct care settings. As defined by Buchbinder and Thompson (2010), direct care settings are those organizations that provide care directly to a patient, resident or client who seeks services from the organization.

Disadvantage: Existing system does not include non-direct care settings. Non-direct care settings are not directly involved in providing care to persons needing health services, but rather support the care of individuals through products and services made available to direct care settings.

Case 2:

The construction of medical information is important to improve the hospital medical care capability, the management decision-making level of health and the hospital operational efficiency. Nowadays, comprehensive hospital information services and management platform have been established, centering on electronic medical records and clinical pathway.

Disadvantage: The establishment and use of these information systems played an important role in improving the degree of patient satisfaction, enhancing hospital efficiency and healthcare quality, protecting the safety of healthcare, and reducing healthcare costs which is not there in existing system.

Case 3:

E Healthcare Management System (computerized) is increasingly becoming an emerging tool in health care arena to efficiently enable delivery of high-quality health services. These systems have large computerized data bases intended primarily for communication and storing health and administrative information. EHMS has different components and includes broad scope and level of systems from departmental.

Disadvantage: Previously the Data was handled manually using record, pen etc. That were difficult to handle physically as the burden on the user increases.

Chapter III: SYSTEM ANALYSIS

3.1 Objectives:

The main objective while implementing the project E-Health care management System was to minimize the work and at the same time increase the speed of the work done.

3.2 Problem specification:

Hospitals currently use a manual system for the management and maintenance of critical information. The current system requires numerous paper forms, with data stores spread throughout the hospital management infrastructure. Often information is incomplete or does not follow management standards.

1. Integration of Corporate Medicare centers is very difficult while it is having different branches.
2. In most of the cases the database is similar from one hospital to another hospital. In those cases, also we can't easily adapt a new technology in the new hospital.
3. It is very difficult to analyze the usage percentage of hospital resources, Bed occupation Ratio, Administration, Laboratory information even in a single center. Then we can expect the complexity while integrating multi-specialty Medicare Centers.
4. Room Reservations, Doctor Appointment Schedules, Operation Schedules, and Medicine indentation information is very difficult to maintain and share among the different Medicare Centers.
5. Lack of generic and unique model we have to implement the same set of data model for every newly established Medicare Center.

Forms are often lost in transit between departments requiring a comprehensive auditing process to ensure that no vital information is lost. Multiple copies of the same information exist in the hospital and may lead to inconsistencies in data in various data stores

3.3 PROPOSED SYSTEM

The E-Health care 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. Room availability, staff and operating room schedules and patient invoices. 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.

In E-Health care Management System situations we are dealing with objectives such as:

1. To optimize bed occupation.
2. To improve the use of operating theatres, avoiding the cancellation of operations.
3. To know how emergencies affect to the administration of the hospital departments or services (cancellation of operations, etc.).
4. To detect the influence of certain diseases in the hospital's services.
5. To find clusters of patients.
6. This system is mainly built for the purpose to reduce the work and improve the efficiency in the hospital's management.
7. Recording information about the Patients that come.
8. Generating bills at any instance of time.
9. Recording information related to diagnosis given to patients.
10. Keeping record of the immunization provided to children/patients.
11. Keeping information about various diseases and medicines available to cure them.

These are the various jobs that need to be done in a hospital by the operational staff and Doctors. The above facts, figures and drawbacks clearly indicate that there is need for computerization and thus decided to computerize the "E-Health care management system".

3.4 Applications

- Consistent user interface with high economic features built into it.
- System design is modular and structured way so as to make the integration with other subsystems easier.

- User has complete control as it provides and accept only appropriate and valid data.
- User-friendly error messages are provided wherever necessary.
- Addition of new patient record, deletion of the existing record and modification of existing records as when needed.
- Provision of saving the info for new patients.
- Generate bills for the patients respectively.

3.5. Modules and their functionalities

Patient Module: -

Patient's personal information can be store by using this module. Details such as Patient ID, Name, Age, Sex, Address, Phone Number, gender can be saved using this module. It is used to store information about patients who were admitted in the hospital on doctor's advice.

History Module: -

Here we can see previous or existing details of particular patient such as Patient ID, Dept depending on disease, Doctor, Ward, Date of admitted, Date of discharge. Updation like deletion and modification is done.

Diagnosis Module: -

This module used to store or produce the laboratory reports. Patient ID, ward Category, Doctor, Date, Medicines or drugs. Updating like deletion and modification is done.

Billing Module: -

This module works as its name itself says, to generate the bill of the patient with displaying the details such as Patient ID, doctor's charge, health card amount, room bill, medicine bill, total amount, no of days, Service charge, Operation theatre charge, Nursing care, Lab bill.

Information Module: -

This module gives information about the working of every module.

3.6 Software and Hardware requirements

Hardware Environment

- The HDD required for the application is minimum 40GB.
- The application works on minimum Intel core i3.
- It requires 2GB RAM.

Software Environment

- The application is designed using Net Bean IDE 8.2.
- The technology used is Java.
- The database has been designed on **.DAT file.**

3.7 Packages

It looks like you've started a Java program and imported some necessary packages. The imported packages include:

java.util.*: This package provides the utility classes for collection framework, events, date, and time manipulation, etc.

java.lang.String: This is the core class for handling strings in Java. The **String** class is part of the **java.lang** package, so it's automatically imported, and you don't need to explicitly import it.

java.io.*: This package provides classes for input and output operations, such as reading and writing to files.

java.time.format.DateTimeFormatter and **java.time.LocalDateTime**: These classes are part of the Java Date and Time API introduced in Java 8. They are used for formatting and parsing date-time objects.

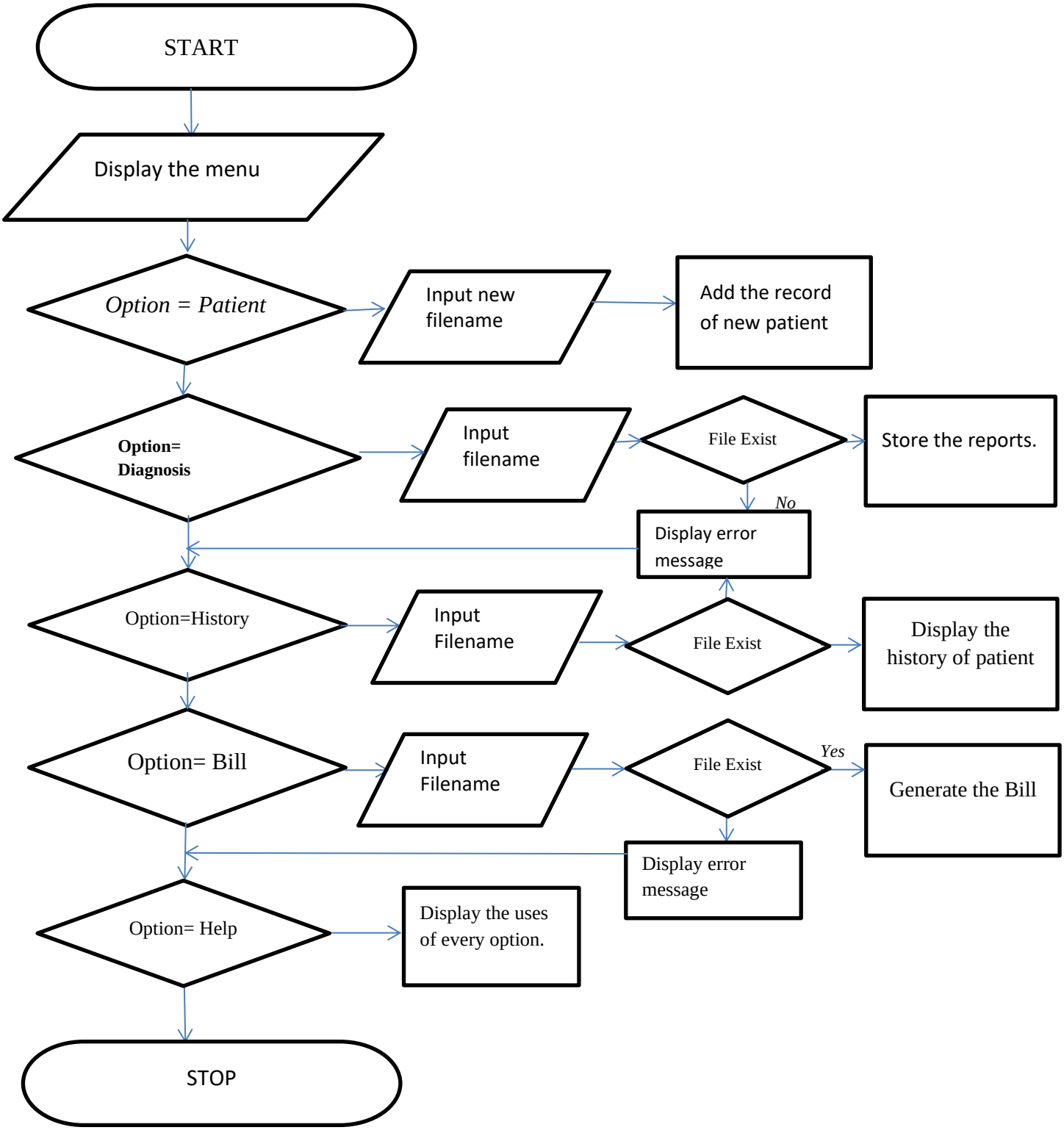
Here's a brief explanation of the classes you've imported related to date and time:

- **DateTimeFormatter**: This class provides formatting and parsing for date-time objects. You can use it to define a pattern for formatting and parsing dates and times.
- **LocalDateTime**: This class represents a date-time without a time zone in the ISO-8601 calendar system. It's often used to represent a specific date and time, and you can perform various operations on it.

If you have specific questions or if you want assistance with the development of your Java program, feel free to ask!

Chapter IV: DESIGN

4.1 Data Flow Chart



4.2 Algorithm

Step 1: Enter the correct password.

Step 2: Menu will be displayed choose the required option you want to perform.

Step 3: Enter Option 1 i.e. the administration module which enables us to perform the following operation illustrate below:

1. Insert the patient information
2. Update the patient information
3. View the patient information
4. Delete the patient information

Step 4: If number of patients>=max patients the display patient can't be added and return

If person is outpatient go to step 6 else step 5**.**

Step 5: option 2 provides us to store patient personal information here we need to enter the following details:

1. Name:
2. Age:
3. Sex:
4. Address:
5. Weight:

Step 6 Now enter option 3 i.e., outpatient module which take the information of the outpatient i.e., illustrated below:

1. Enter the ID:
2. Enter the dept depending on the disease:
3. Room number:
4. Date of admitted date of discharge:

Step 7: Now press 4 to create patient report which ask us to

1. Enter patient ID number:
2. Enter Weight
3. Enter Name of Doctor appointed to
4. Date
5. Enter Precipitation to be followed.

Step 8: Now enter 5 to generate bill of the patient, to generate bill

1. Enter the id number of patient
2. Name Doctor appointed to
3. Doctor fee
4. Health card number (not compulsory)
5. Room bill
6. Medicine bill
7. Lab bill
8. Operation theatre charges
9. Nursing bill

Step 9: Now press enter which will display you the total cost you charged.

Step 10: Exit

Chapter V: IMPLEMENTATION

5.1. Partial code

```
import java.util.*;
import java.lang.String;
import java.io.*;
import java.time.format.DateTimeFormatter;
import java.time.LocalDateTime;


class Info
{
   Info()
   {
      System.out.print("\t\t\t\t\t
_____
_____\n");
      System.out.print("\t\t\t\t\t|                                                                    |\n");
      System.out.print("\t\t\t\t\t|                                                                    |\n");
      System.out.print("\t\t\t\t\t|                                                                    |\n");
      System.out.print("\t\t\t\t\t|                                                                    |\n");
      System.out.print("\t\t\t\t\t|                                                                    |\n");
      System.out.print("\t\t\t\t\t|                                                                    |\n");
      System.out.print("\t\t\t\t\t|          WELCOME TO E-HEALTH CARE
                                    MANAGEMENT SYSTEM                  |\n");
      System.out.print("\t\t\t\t\t|                                                                   |\n");
      System.out.print("\t\t\t\t\t|                                                                   |\n");
```

```
        System.out.print("\t\t\t\t\t|                                                                |\n");
        System.out.print("\t\t\t\t\t|                                                                |\n");
        System.out.print("\t\t\t\t\t|                                                                |\n");
        System.out.print("\t\t\t\t\t|                                                                |\n");
        System.out.print("\t\t\t\t\t|                -Brought To You by
                                                                        |\n");
        System.out.print("\t\t\t\t\t|                 Sumit Kumar, Ritik Chauhan and Pankaj kumar
                                                                        |\n");
System.out.print("\t\t\t\t\t|_____
_____|\n");
    }
    //Declaring variales to be used
    Scanner scan=new Scanner(System.in);
    String name;
    String address;
    long contact;
    int age;
    String bg;
    String sex;
    String disease;
    long id;
    String dadm;
    //To take a pause
    private void pressAnyKeyToContinue()
    {
```

```
System.out.print("\t\t\t\t\t_____
_____\n");

    System.out.println("\t\t\t\t\t  Press Enter key to continue...");
    try
    {
        System.in.read();
    }
    catch(Exception e)
    {
        System.out.println("Press 'Enter' key to continue!");
    }
}
//Taking multiple words
String readString()
{
    Scanner scanner = new Scanner(System.in);
    return scanner.nextLine();
}
//Log in Module
void login()
{
    int a;
    String pass;
```

```
System.out.print("\t\t\t\t\t________________________________________________________________________________________________\n");

        System.out.print("\n\t\t\t\t\t\t\t            E-HEALTH CARE MANAGEMENT SYSTEM \n");

System.out.print("\t\t\t\t\t________________________________________________________________________________________________\n");

System.out.print("\t\t\t\t\t________________________________________________________________________________________________\n");
        System.out.print("\n\t\t\t\t\t\t\t\t                      LOGIN \n");

System.out.print("\t\t\t\t\t________________________________________________________________________________________________\n");
        System.out.print("\t\t\t\t\tEnter the Password:");
        pass=scan.next();
        if(pass.equals("12345"))
        {
            System.out.print("\t\t\t\t\t____________________________________________Access Granted!______________________________________________\n");
            pressAnyKeyToContinue();
        }
        else
        {
            System.out.print( "\n\n\t\t\t\t\t\t\t\tAccess Aborted...\n\t\t\t\t\t\t\t1.Try Again\n\n\t\t\t\t\t\t\t2.Exit");
            System.out.print("\n\n\t\t\t\t\t\tEnter the option: ");
            try{
```

```
            a=scan.nextInt();
            if(a==1)
               login();
            else if(a==2)
               System.exit(0);
            else
            {
               System.out.print("\n\n\t\t\t\t\t\tInvalid Choice\n");
            }
         }
         catch(InputMismatchException e){
            System.out.println("\n\n\t\t\t\t\t\t*Enter the correct option!*\n\n");
            login();
         }
      }
}
//To get Date
public void getCurrentTimeUsingDate()
{
   DateTimeFormatter dtf = DateTimeFormatter.ofPattern("dd/MM/yyyy HH:mm:ss");
   LocalDateTime now = LocalDateTime.now();
   dadm=dtf.format(now);

}
```

```
    void menu()
    {
        char k;
//giving option to the user for their choice

System.out.print("\t\t\t\t\t___________________________________________________________________
__________________________________\n");
        System.out.print("\n\t\t\t\t\t\t\t          E-HEALTH CARE MANAGEMENT SYSTEM \n");

System.out.print("\t\t\t\t\t___________________________________________________________________
__________________________________\n");
        System.out.print("\n\t\t\t\t     Please, Choose from the following Options: \n\n");
        System.out.print("\t\t\t\t\t\t
______________________________________________________________________\n");
        System.out.print("\t\t\t\t\t\t|                                                |\n");
        System.out.print("\t\t\t\t\t\t|          1  >> Add New Patient Record          |\n");
        System.out.print("\t\t\t\t\t\t|          2  >> Add Diagnosis Information       |\n");
        System.out.print("\t\t\t\t\t\t|          3  >> History of the Patient          |\n");
        System.out.print("\t\t\t\t\t\t|          4  >> Bill of the patient             |\n");
        System.out.print("\t\t\t\t\t\t|          5  >> Help                            |\n");
        System.out.print("\t\t\t\t\t\t|          6  >> Exit                            |\n");

System.out.print("\t\t\t\t\t\t|____________________________________________________________
_________|\n\n");
        System.out.print("\t\t\t\t\t  Enter your choice: ");
        k=scan.next().charAt(0);
//if inputed choice is other than given choice
        switch(k)
```

```
    {
        case '1': patient();
            break;
        case '2': diagnos();
            break;
        case '3': history();
            break;
        case '4': bill();
            break;
        case '5': info();
            break;
        case '6': exit();
            break;
        default: System.out.print("\n\n\t\t\t\t\t\tInvalid Choice\n");
            System.out.print("\t\t\t\t\tTry again.....\n\n");
            menu();
    }
    menu();
}
void patient()
{
    System.out.print("\t\t\t\t  Enter the patient's file name: ");

    String fileName =scan.next();
    getCurrentTimeUsingDate();
```

```
        try {
            // Assume default encoding.
            FileWriter fileWriter = new FileWriter(fileName+".txt");

            // Always wrap FileWriter in BufferedWriter.
            BufferedWriter bufferedWriter = new BufferedWriter(fileWriter);

            // Note that write() does not automatically
            // append a newline character.

System.out.print("\t\t\t\t\t**********************************************************
**********************************\n");

            Scanner scan = new Scanner(System.in);
            String date;
            System.out.print("\t\t\t\t\t    Enter a date (dd/mm/yyyy): ");date =
scan.next();bufferedWriter.write("Date: " + date);
            bufferedWriter.newLine();
            System.out.print("\t\t\t\t\t    Name : ");name=readString();bufferedWriter.write("Name :
"+name);
            bufferedWriter.newLine();
            System.out.print("\t\t\t\t\t    Address :
");address=readString();bufferedWriter.write("Address : "+address);
            bufferedWriter.newLine();
            System.out.print("\t\t\t\t\t    Contact Number :
");contact=scan.nextLong();bufferedWriter.write("Contact Number : "+contact);
```

```
        bufferedWriter.newLine();

        System.out.print("\t\t\t\t\t    Age : ");age=scan.nextInt();bufferedWriter.write("Age :
"+age);

        bufferedWriter.newLine();

        System.out.print("\t\t\t\t\t    Sex : ");sex=scan.next();bufferedWriter.write("Sex : "+sex);

        bufferedWriter.newLine();

        System.out.print("\t\t\t\t\t    Blood Group : ");bg=scan.next();bufferedWriter.write("Blood
Group : "+bg);

        bufferedWriter.newLine();

        System.out.print("\t\t\t\t\t    Any Major disease suffered earlier :
");disease=readString();bufferedWriter.write("Any Major disease suffered earlier : "+disease);

        bufferedWriter.newLine();

        System.out.print("\t\t\t\t\t    Patient ID :
");id=scan.nextLong();bufferedWriter.write("Patient ID : "+id);

        bufferedWriter.newLine();


System.out.print("\t\t\t\t\t**********************************************************
**********************************\n");


bufferedWriter.write("\n**********************************************************
**********\n\n");

        bufferedWriter.newLine();System.out.print("\n\t\t\t\t\t  Information Saved
Successfully\n");


        // Always close files.

        bufferedWriter.close();

    }

    catch(IOException ex)
```

```
    {
        System.out.println("Error writing to file '"+ fileName +"'");
    }
    pressAnyKeyToContinue();
}
void diagnos()
{
    String symptom;
    String diagnosis;
    String medicine;
    String addmission;
    String ward;
    String doctor;
    getCurrentTimeUsingDate();
    System.out.print("\n\nEnter the patient's file name to be opened : ");
    String fileName=scan.next();
    // This will reference one line at a time
    String line = null;
    try {
        // FileReader reads text files in the default encoding.
        FileReader fileReader = new FileReader(fileName+".txt");

        // Always wrap FileReader in BufferedReader.
        BufferedReader bufferedReader = new BufferedReader(fileReader);
```

```
System.out.println("\n\n\n\n\t\t\t\t..... Information about '"+fileName+"' .....\n\n\n\n");

while((line = bufferedReader.readLine()) != null)

{

   System.out.println(line);

}


// Always close files.

bufferedReader.close();

//Appending data in file

PrintWriter out = new PrintWriter(new BufferedWriter(new FileWriter(fileName+".txt", true)));

System.out.print("Adding more information in patient's file.....on : "+dadm);

out.println("Description of the day:"+dadm);

System.out.print("\nDoctor appointed:");doctor=readString();out.println("Doctor appointed: "+doctor);

System.out.print("\nSymptoms : ");symptom=readString();out.println("Symptoms : "+symptom);

System.out.print("\nDiagnosis : "); diagnosis=readString();out.println("Diagnosis : "+diagnosis);

System.out.print("\nMedicines : ");medicine=readString(); out.println("Medicines : "+medicine);

System.out.print("\nAddmission Required? : ");

addmission=readString();out.println("Addmission Required? : "+addmission);

if(addmission.equals("Yes")||addmission.equals("Y"))

{

   System.out.print("\nType of ward : ");ward=readString(); out.println("Type of ward : "+ward);
```

```
out.println("\n**********************************************************************
*******\n");
            System.out.print(ward+" ward is alloted Successfully\n");
        }
        else

{out.println("**********************************************************************
******\n");}
        out.close();
        pressAnyKeyToContinue();
      }
      catch(FileNotFoundException ex) {
        System.out.println(
              "Unable to open file '" + fileName + "'");
      }
      catch (IOException e) {
        System.out.println("Error writing or Reading to file '"+ fileName +"'");
      }
   }

   void history()
   {
      System.out.print("\n\nEnter the patient's file name to be opened : ");
      String fileName=scan.next();
      // This will reference one line at a time
      String line = null;
```

```
        try {
            // FileReader reads text files in the default encoding.
            FileReader fileReader = new FileReader(fileName+".txt");


            // Always wrap FileReader in BufferedReader.
            BufferedReader bufferedReader = new BufferedReader(fileReader);
            System.out.print("\n\n\n\n\t\t\t\t..... Full Medical History of
'"+fileName+"' .....\n\n\n\n");
            while((line = bufferedReader.readLine()) != null)
            {
               System.out.println(line);
            }
            pressAnyKeyToContinue();
            // Always close files.
            bufferedReader.close();
        }
        catch(FileNotFoundException ex) {
            System.out.println(
                    "Unable to open file '" + fileName + "'");
        }
        catch(IOException ex) {
            System.out.println(
                    "Error reading file '" + fileName + "'");
        }
    }
```

```
   void bill()
   {
      getCurrentTimeUsingDate();
      int days;
      double wcharge;
      double doc;
      double ser;
      System.out.println("\t\t\t\t\t\t
_____ ");
      System.out.print("\n\n\t\t\t\t\t\t\t\t  E-HEALTH CARE MANAGEMENT SYSTEM \n\n");
      System.out.println("\t\t\t\t\t\t
_____ ");
      System.out.print("Enter the patient's file name to get the Bill: ");
      String fileName=scan.next();
      // This will reference one line at a time
      String line = null;
      try {
         // FileReader reads text files in the default encoding.
         FileReader fileReader = new FileReader(fileName+".txt");
         PrintWriter out = new PrintWriter(new BufferedWriter(new FileWriter(fileName+".txt",
true)));

         // Always wrap FileReader in BufferedReader.
         BufferedReader bufferedReader = new BufferedReader(fileReader);
         System.out.print("Date: "+dadm);
         System.out.println("\n\n\n\n\t\t\t\tDetails of the patient:\n\n\n\n");
```

```
        while((line = bufferedReader.readLine()) != null)
        {
            System.out.println(line);
        }
        System.out.print("\t\t\t\t\t\t
_____ ");
        System.out.println("\n\t\t\t\t\t\t\t\t\t\tBill of the patient:");
        System.out.print("\t\t\t\t\t\t
_____ \n\n");
        System.out.print("\t\t\t\t\t\tNo.of days patient stayed:");days=scan.nextInt();
        System.out.print("\t\t\t\t\t\tWard charges/day:");wcharge=scan.nextDouble();
        System.out.print("\n\t\t\t\t\t\tTotal ward charges:"+(wcharge*days));
        System.out.print("\n\t\t\t\t\t\tDoctor's fee:");doc=scan.nextDouble();
        System.out.print("\n\t\t\t\t\t\tService charges:");ser=scan.nextDouble();
        System.out.print("\n\t\t\t\t\t\tTOTAL AMOUNT:"+((wcharge*days)+doc+ser));
        out.println("Total Bill:"+((wcharge*days)+doc+ser));
        System.out.print("\n\t\t\t\t\t\t
_____ \n");
        // Always close files.
        out.close();
        bufferedReader.close();
    }
    catch(FileNotFoundException ex) {
        System.out.println(
            "Unable to open file '" + fileName + "'");
    }
```

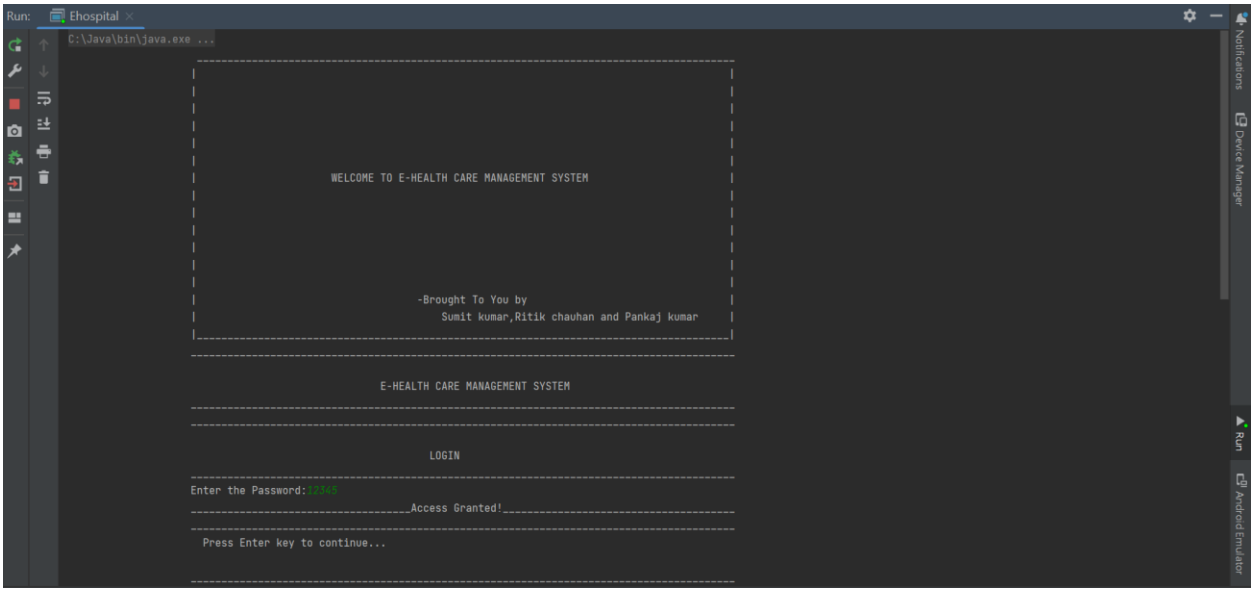
```
    catch(IOException ex) {
        System.out.println(
                "Error reading file '" + fileName + "'");
    }
    pressAnyKeyToContinue();
  }
  void exit()
  {
    System.out.print("\n\n\n\n\n\n\n\n\n\n\n\n\t\t\t\t\t\n");
    System.out.print("\t\t\t\t\t
_____ \n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
    System.out.print("\t\t\t\t\t|              THANK YOU FOR USING
                                                       |\n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
    System.out.print("\t\t\t\t\t|        E-HEALTH CARE MANAGEMENT SYSTEM
                                                        |\n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
    System.out.print("\t\t\t\t\t|                                                    |\n");
```

```
        System.out.print("\t\t\t\t\t|                    -Brought To You by sumit,ritik and pankaj
                                                                                |\n");

System.out.print("\t\t\t\t\t|_____
_____|\n");

        System.out.print("\t\t\t\t\t\n\n\n\n\t\t\t\t\t");

        System.exit(0);

    }

    void info()

    {

        System.out.println("\n\nE-Health care management system is command-driven application
that helps to manage the record of the patient's as well as generate their hospital bills.");


System.out.print("\t\t\t\t_____
_____\n");

        System.out.print("\t\t\t\t\tOptions                    | Description\n");


System.out.print("\t\t\t\t_____
_____\n");

        System.out.println("\t\t\t\t 1 >> Add New Patient Record       | A new patient file can be
created by using this module.");

        System.out.println("\t\t\t\t 2 >> Add Diagnosis Information   | Adding day-to-day evaluation
of the patient or the diagnosis report in the existing file.");

        System.out.println("\t\t\t\t 3 >> History of the Patient        | Check the previous or existing
                                                                          details of particular patient
                                                                            by using their file name.");
System.out.println("\t\t\t\t 4 >> Bill of the patient                  | Generates the bill of the
patient with displaying the details.");

        System.out.println("\t\t\t\t 5 >> Help                          | Provide the help
documentation.");

        System.out.println("\t\t\t\t 6 >> Exit                          | Exits the application.");
```

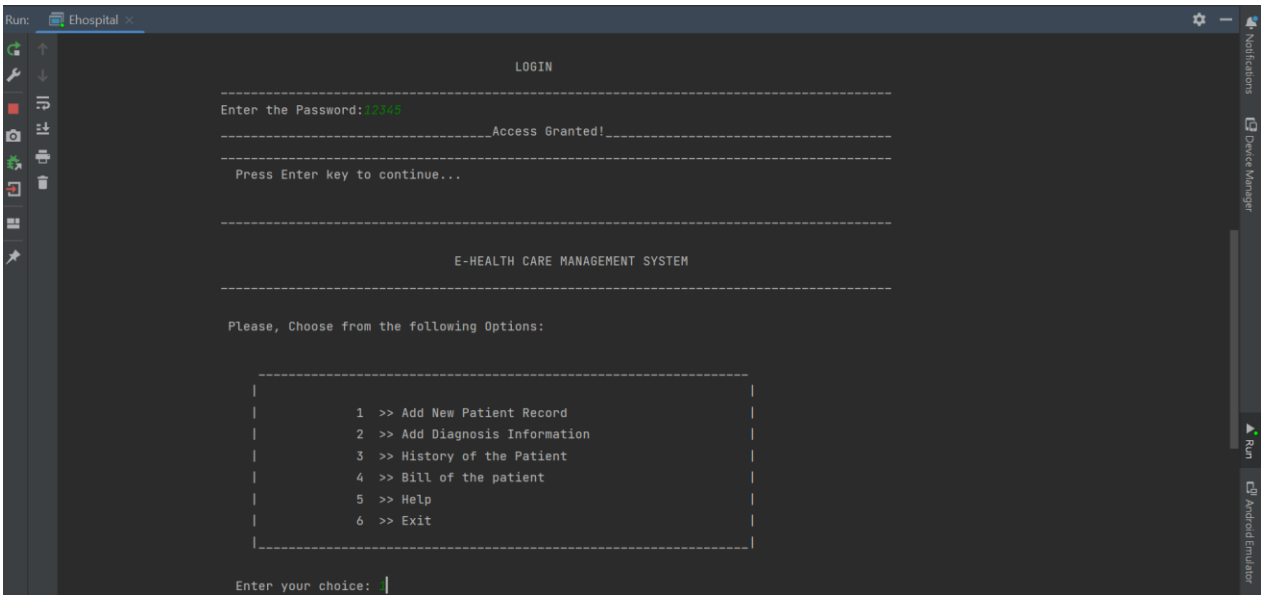
```
        pressAnyKeyToContinue();
    }
}
class Ehospital
{
    public static void main (String args[])
    {
        Info i=new Info();
        i.login();
        i.menu();
    }
}
```

5.2. Screenshots

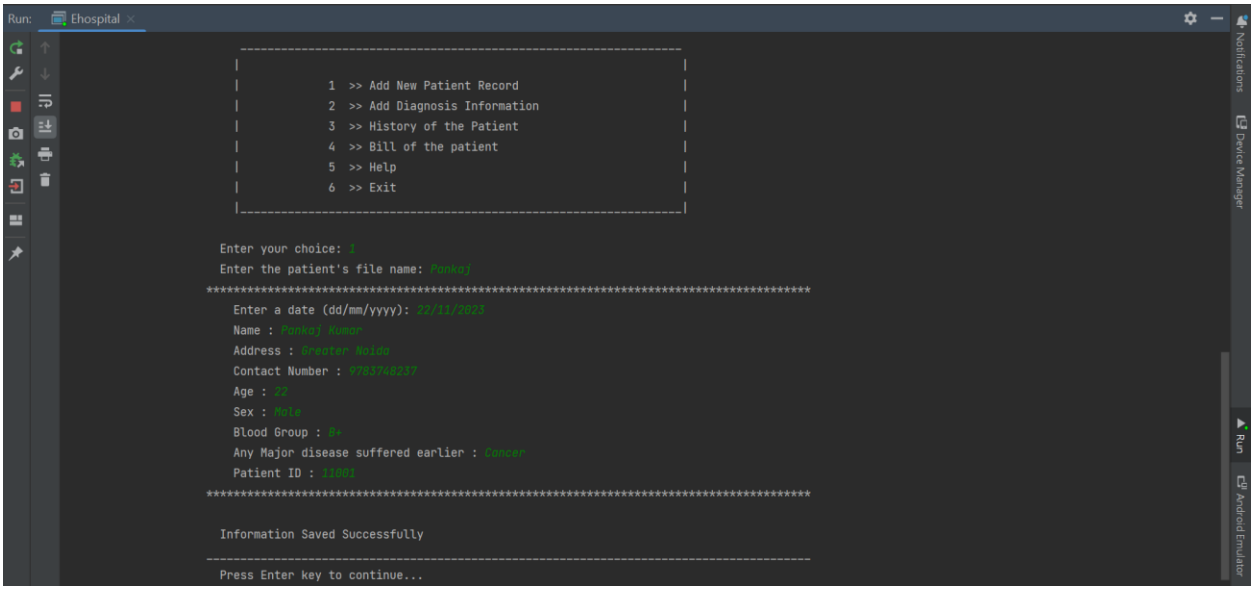
1.Login menu



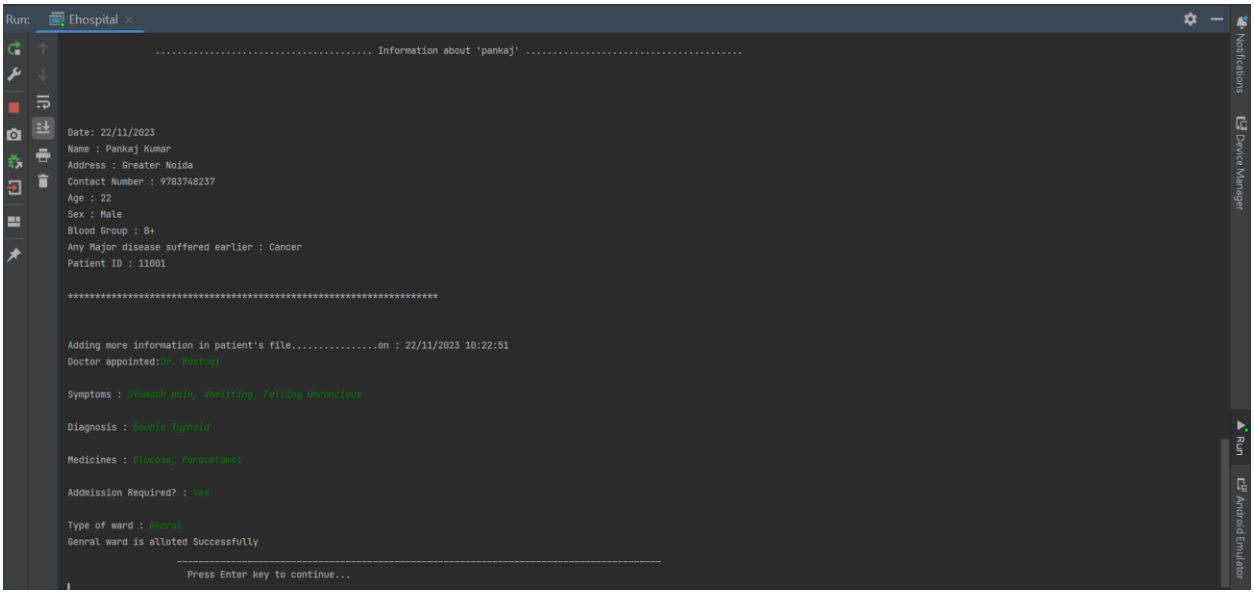
2.Main menu



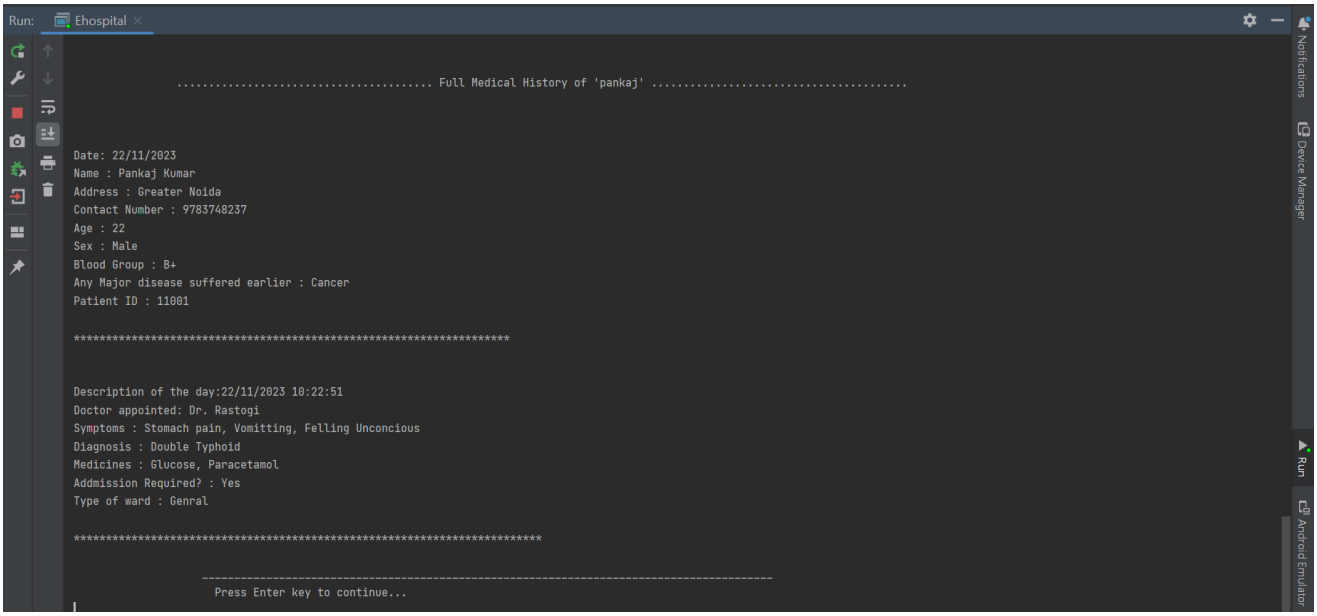
3.Adding patient record



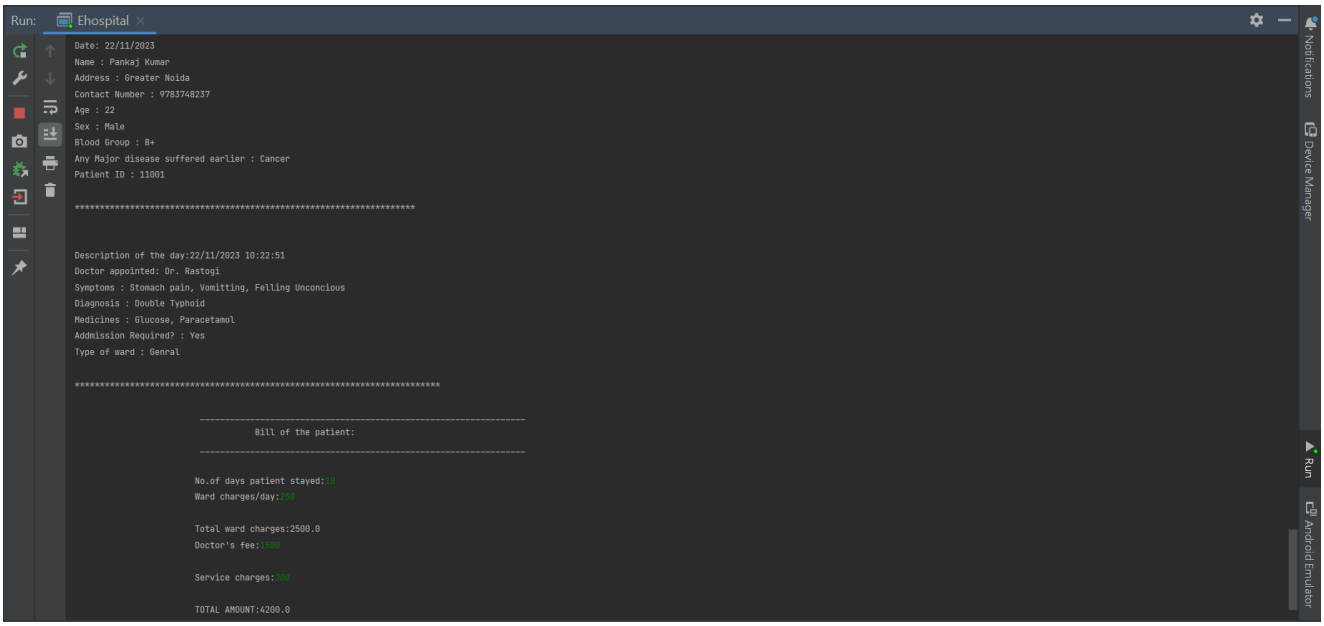
4.Adding Diagnosis report



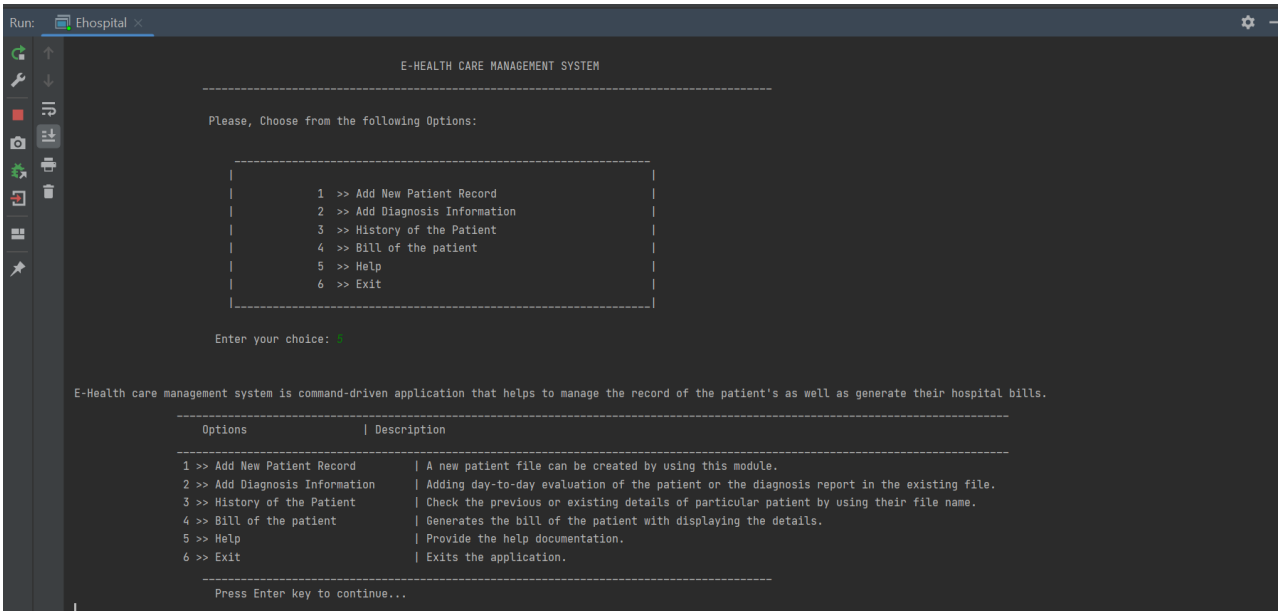
5. History of the Patient



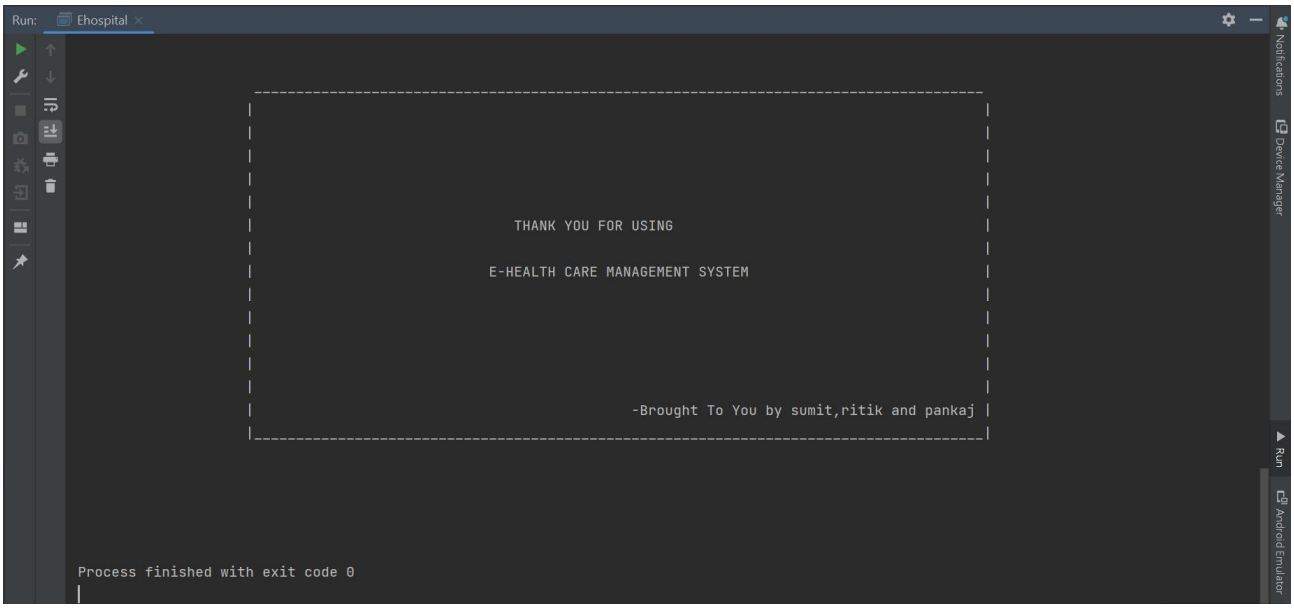
6. Generate Bill



7. Help


Run:
Ehospital
E-HEALTH CARE MANAGEMENT SYSTEM
Please, Choose from the following Options:
1 >> Add New Patient Record
2 >> Add Diagnosis Information
3 >> History of the Patient
4 >> Bill of the patient
5 >> Help
6 >> Exit
Enter your choice: 5
E-Health care management system is command-driven application that helps to manage the record of the patient's as well as generate their hospital bills.
Options | Description
1 >> Add New Patient Record | A new patient file can be created by using this module.
2 >> Add Diagnosis Information | Adding day-to-day evaluation of the patient or the diagnosis report in the existing file.
3 >> History of the Patient | Check the previous or existing details of particular patient by using their file name.
4 >> Bill of the patient | Generates the bill of the patient with displaying the details.
5 >> Help | Provide the help documentation.
6 >> Exit | Exits the application.
Press Enter key to continue...

8. Exit


Run:
Ehospital
THANK YOU FOR USING
E-HEALTH CARE MANAGEMENT SYSTEM
-Brought To You by sumit,ritik and pankaj
Process finished with exit code 0

Chapter VI: Future enhancement

Future Enhancement

Every project whether large or small has some limitations no matter however diligently developed. In some cases, limitations is small while in other cases they may be broad also. The new system has got some limitations. Major areas where modifications can be done are as follows:

- Our system is not online so further it can be improved.
- The security is limited so some additional arrangement could be made to provide more security to the system.

Chapter VII: Conclusion

Conclusion

- A fully menu driven user-friendly computerized system has been developed where the user can perform task like entering data and appending the information with great ease.
- All the operations are carried automatically preventing a lot of manual work.
- Additional checks have also been incorporated into the system to avoid duplications of data as far as possible.