**MINI PROJECT REPORT**



* **Mini project title:**

“Hospital Management System”

* **Details of group members:**

|  |  |  |  |
| --- | --- | --- | --- |
| **USN** | **NAME** | **SEMESTER** | **BRANCH** |
| 1MV19IS015 | LAHARI C | 3 | ISE |
| 1MV19IS018 | DEEPANSHU BACHLOO | 3 | ISE |
| 1MV19IS039 | NEHA S GAMANAGATTI | 3 | ISE |
| 1MV19IS062 | VAISHNAVI R | 3 | ISE |

**PROJECT OVER VIEW /**

**BACKGROUND**

**INTRODUCTION**

Hospital is the essential part of our lives, providing best medical facilities to people suffering from various ailments, which may be due to change in climatic conditions, due to work load, emotional trauma, stress etc.

It is necessary for the hospital to keep track of its day-today activities and records of its patients, doctors, nurses, ward boys and other staff personals that keep the hospital running smoothly and successfully.

But keeping track of all the activities and their records on the paper is very cumbersome and error prone. It is also is very inefficient and a time-consuming process Observing the continuous increase in population and number of people visiting the hospital.

**Reason for selection of the project**

Recording and maintaining all these records are highly unreliable and manually. Thus, to keeping the working of the manual system as the basis of our project. The main aim of our project is to provide a paper-less hospital up to 90%. It also aims at providing low-cost reliable automation of the existing system.

**Objective**

Main objectives of hospital management system are to design a system for better patient care. To computerize all details regarding patient’s details and hospital details.

Scheduling the appointment of patient with doctors to make it convenient for both. The information of the patients should be kept up to date and their record should be kept in the system for historical purposes.

**Problem statement**

It is very important to maintain efficient software to handle information of a hospital. This application provides a way to record this information and to access these in simple way. There are two system: -

**Existing system:** The existing system provide the basic functionalities needed to be handled management environment. There is no intelligence of the software in such cases. In the existing system all the patient details doctor availability details and regarding test done to the patients prescribed by the doctor is maintaining manually by the receptionist. If the patient has to be admitted we need to cheek the availability of the bed which consumes lots of time if done manually.

**Proposed system:** whereas in this system, we are going to provide solutions to all the above-mentioned problems by automating the whole hospital management system by using an integrated software that handle the whole system

**+Benefits to the surrounding / society:**

Hospital management system is a medical information solution element that mainly focuses on the hospital administration requirements. The HMS is a computer application that takes care of the complete hospital functionalities. The integrated system can be customized and are developed to control all hospital operations like patient details, appointment booking, and billing so on.

Some of the top benefits of implementing an HMS are role-based access control, data accuracy, revenue management, overall cost reduction and data security.

**Advantages of Hospital Management**

**System**

The main advantage of hospital management software is to automate the business process of a hospital and lower the expenses. Same functionalities can be managed with the lower staff efficiently and timely.

* Easy access to patient data.
* Cost effective.
* Improve deficiency.
* Reduces scope of error.
* Increase data security and retrieve ability.
* Improve patient care.
* Improve data security.
* Better revenue management.

**Disadvantages of Hospital Management System**

The main disadvantage is a compromise on the security of information and dependency like in case of a system crash, running operations would seem a difficult task.

* Included reducing access.
* Reducing the rate of hospital admissions.
* Increasing employees work load.
* Dissatisfaction.

**Hardware Requirements:**

* Systems with 8GB RAM, 256GB SSD Storage
* Pen drives with the storage capacity of 2 GB

**Software Requirements:**

* Code Blocks IDE

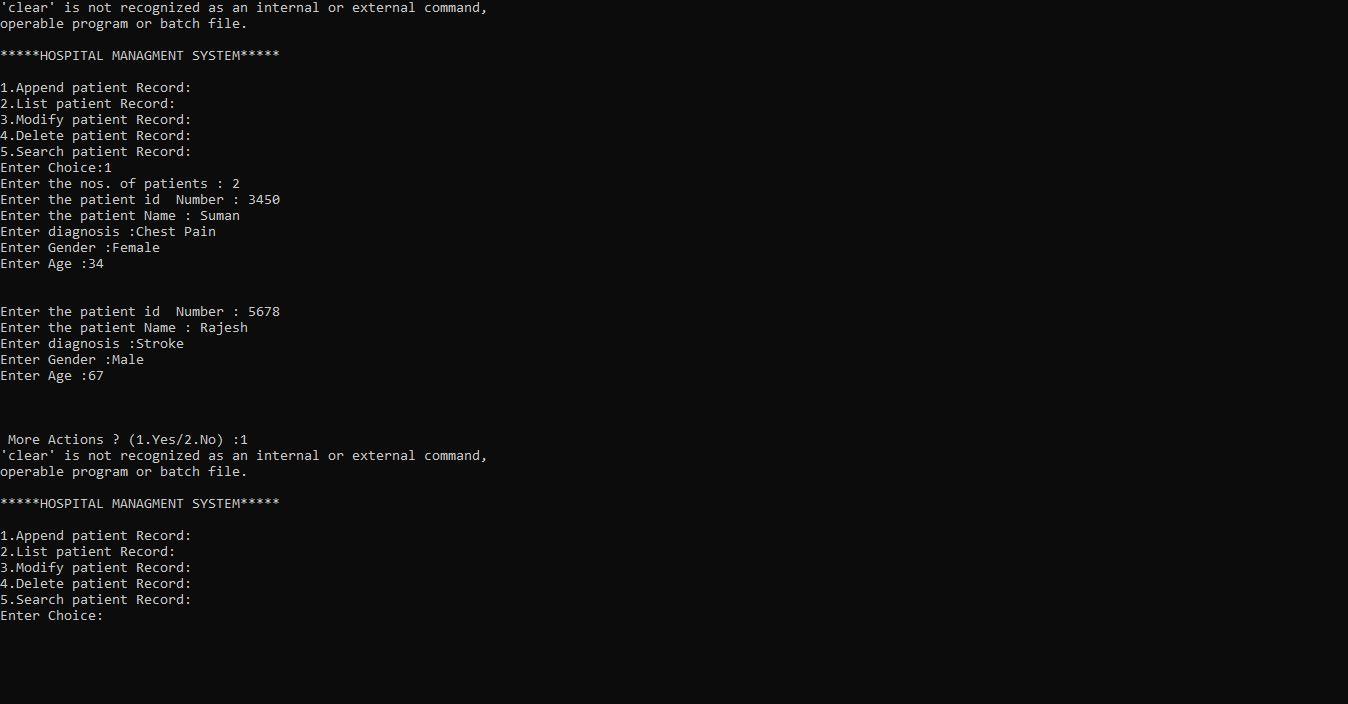
**Code**

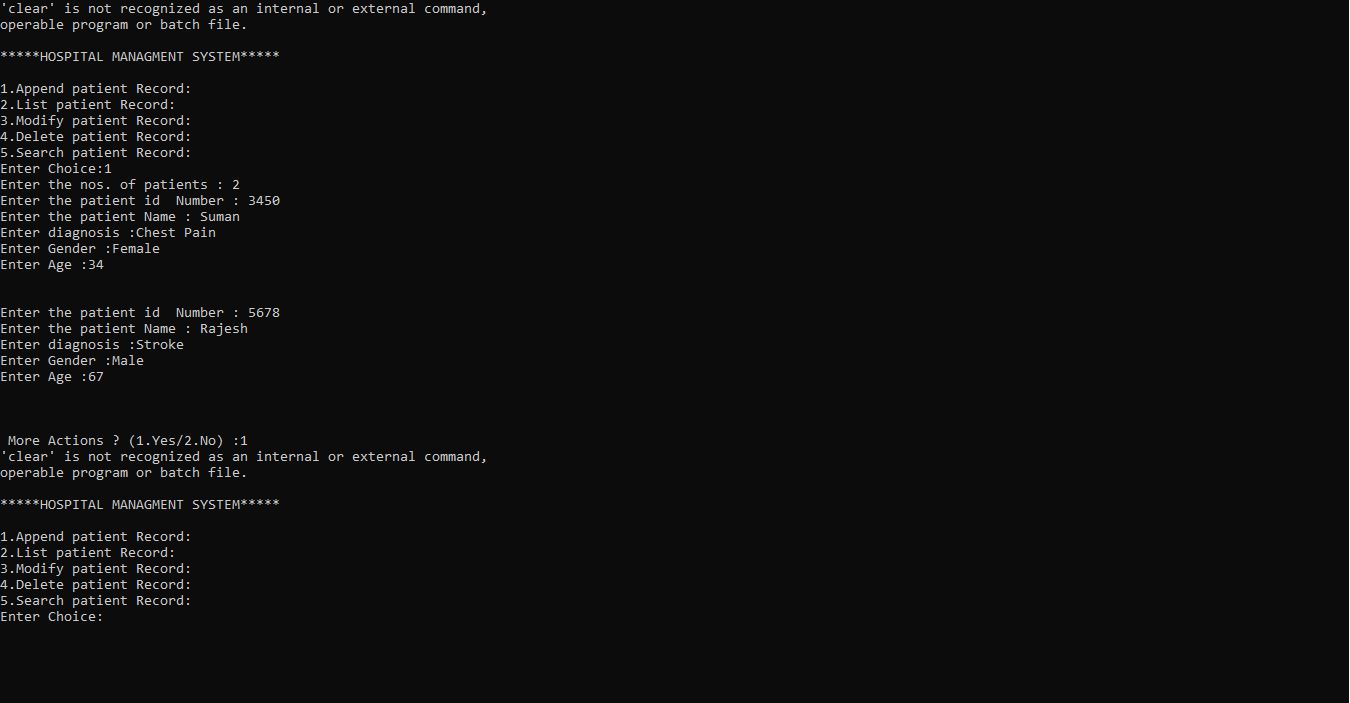
#include<stdio.h>  
void append();  
void view();  
void search();  
void modify();  
void delete();  
struct patient  
{  
char name[20];  
char diag[12];  
char gender[5];  
int no;  
int age;  
};  
  
void main()  
{  
int a;  
int ch;  
do  
{  
system("clear");  
printf("\n\*\*\*\*\*HOSPITAL MANAGMENT SYSTEM\*\*\*\*\*\n\n");  
printf("1.Append patient Record:\n2.List patient Record:\n3.Modify

patient Record:\n");  
printf("4.Delete patient Record:\n5.Search patient Record:\nEnter Choice:");  
scanf("%d",&a);  
switch(a)  
{  
case 1:  
append();  
break;  
case 2:  
view();  
break;  
case 3:  
modify();  
break;  
case 4:  
delete();  
break;  
case 5:  
search();  
break;  
default :  
printf("Invalid Choice!");  
}  
printf("\n More Actions ? (1.Yes/2.No) :");  
scanf("%d", &ch);  
}while(ch==1);  
}  
//APPENDING PATIENT DETAILS TO FILE  
void append()  
{  
int i,n;  
struct patient e;  
FILE \*fp;  
fp=fopen("patient.dat", "a");  
if(fp==NULL)  
{  
printf("File Creation Failed!");  
}  
printf("Enter the nos. of patients : ");  
scanf("%d", &n);  
for(i=0;i<n;i++)  
{  
printf("Enter the patient id  Number : ");  
scanf("%d", &e.no);  
printf("Enter the patient Name : ");  
scanf("%s",[e.name](http://e.name/));  
printf("Enter diagnosis :");  
scanf("%s",e.diag);  
fflush(stdin);  
printf("Enter Gender :");  
scanf("%s",e.gender);  
printf("Enter Age :");  
scanf("%d",&e.age);  
printf("\n\n");  
fwrite((char \*)&e, sizeof(e), 1, fp);  
}  
fclose(fp);  
}  
//VIEWING PATIENT DETAILS  
  
void view()  
{  
int nofrec=0;  
struct patient e;  
FILE \*fp;  
fp=fopen("patient.dat", "r");  
if(fp==NULL)  
printf("\n\tFile doesn’t exist!! try again");  
else  
{  
while((fread((char \*)&e, sizeof(e), 1, fp))==1)  
{  
nofrec++;  
printf("\npatient Number : %d ", [e.no](http://e.no/));  
printf("\npatient Name : %s",[e.name](http://e.name/));  
printf("\npatient diagnosis : %s",e.diag);  
printf("\npatient gender : %s",e.gender);  
printf("\npatient age : %d",e.age);  
printf("\n\n");  
}  
printf("Total number of records present are : %d", nofrec);  
fclose(fp);  
}  
}  
  
//MODIFY THE PATIENT DETAIL  
  
void modify()  
{  
int recno, nofrec=0;  
int ch;  
struct patient e;  
FILE \*fp;  
fp=fopen("patient.dat", "rb+");  
if(fp==NULL)  
                printf("\n\tFile doesn’t exist!! try again");  
else  
{  
view();  
printf("\nEnter the patient id Number to modify : ");  
scanf("%d", &recno);  
while((fread((char \*)&e, sizeof(e), 1, fp))==1)  
{  
nofrec++;  
if([e.no](http://e.no/)==recno)  
{  
printf("\npatient Number : %d", e.no);  
printf("\npatient Name : %s",[e.name](http://e.name/));  
printf("\npatient diagnosis : %s",e.diag);  
printf("\npatient gender : %s",e.gender);  
printf("\npatient age : %d",e.age);  
printf("\n");  
printf("Do you want to modify this record : ? (1.Y/2.N)");  
scanf("%d", &ch);  
fseek(fp, ((nofrec-1)\*sizeof(e)),0);  
if(ch==1)  
{  
printf("Enter the patient id No : ");  
scanf("%d",&[e.no](http://e.no/));  
printf("Enter the patient Name : ");  
scanf("%s",[e.name](http://e.name/));  
printf("Enter diagnosis :");  
scanf("%s",e.diag);  
printf("Enter Gender :");  
scanf("%s",e.gender);  
printf("Enter Age :");  
scanf("%d",&e.age);  
  
fwrite((char \*)&e, sizeof(e), 1, fp);  
printf("Record Modified");  
}  
else  
printf("No modifications were made");  
fclose(fp);  
}  
}  
}  
}  
//DELETE THE RECORD  
void delete()  
{  
int recno;  
int ch;  
struct patient e;  
FILE \*fp, \*ft;  
fp=fopen("patient.dat", "rb");  
ft=fopen("Temp.dat", "wb");  
if(fp==NULL)  
                printf("\n\tFile doesn’t exist!! try again");  
else  
{  
view();  
printf("\nEnter the patient id Number to delete : ");  
scanf("%d", &recno);  
while((fread((char \*)&e, sizeof(e), 1, fp))==1)  
{  
if([e.no](http://e.no/)==recno)  
{  
printf("\npatient Number : %d", [e.no](http://e.no/));  
printf("\npatient Name : %s",[e.name](http://e.name/));  
printf("\npatient diagnosis : %s",e.diag);  
printf("\npatient gender : %s",e.gender);  
printf("\npatient age : %d",e.age);  
printf("\n");  
printf("Do you want to delete this record : ? (1.Y/2.N)");  
scanf("%d", &ch);  
}  
}  
if(ch==1)  
{  
rewind(fp);  
while((fread((char \*)&e, sizeof(e), 1, fp))==1)  
{  
if(recno!=[e.no](http://e.no/))  
{  
fwrite((char \*)&e, sizeof(e), 1, ft);  
}  
}  
printf("\nrecord deleted\n");  
}  
else  
printf("No Record was deleted");  
fclose(fp);  
fclose(ft);  
remove("patient.dat");  
rename("Temp.dat", "patient.dat");  
}  
}  
  
////SEARCH THE PATIENT RECORD  
void search()  
{  
int s,recno;  
char sname[20];  
struct patient e;  
FILE \*fp;  
fp=fopen("patient.dat", "rb");  
if(fp==NULL)  
                printf("\n\tFile doesn’t exist!! try again");  
else  
{  
printf("\n1.Search by Name\n2.Search by patient No.\n Enter choice

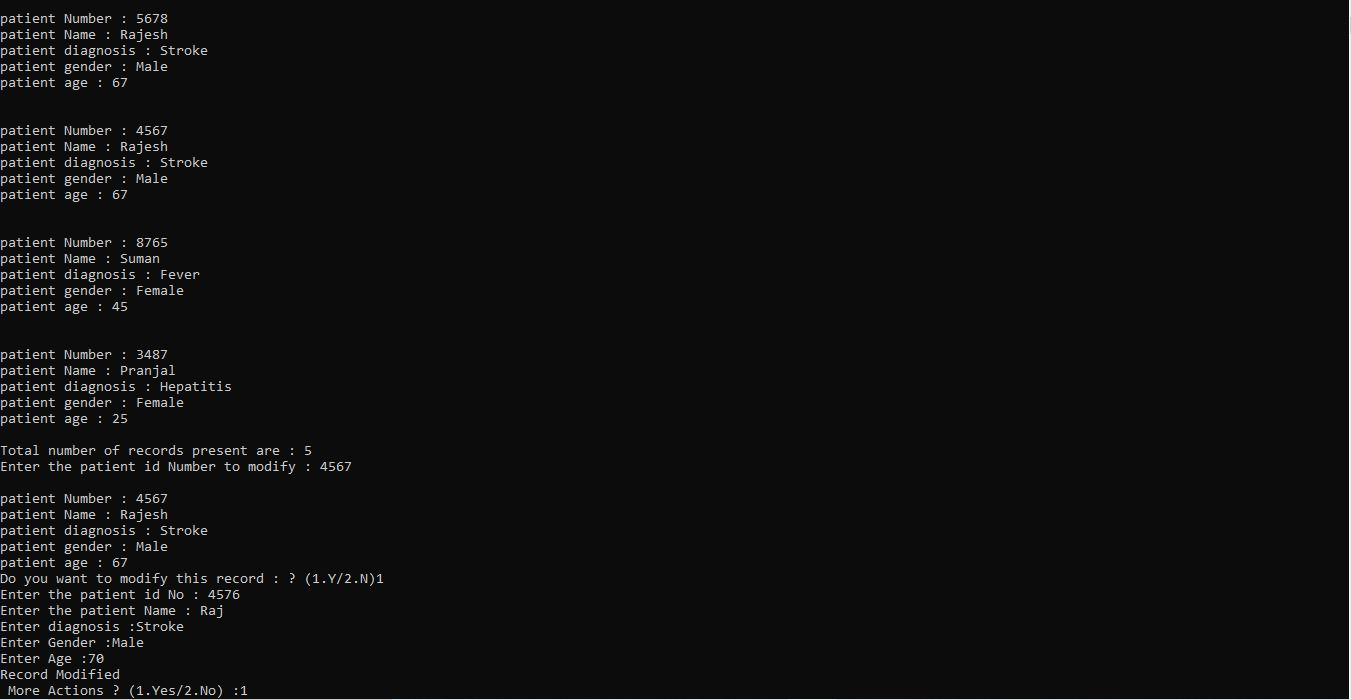
: ");  
scanf("%d", &s);  
switch(s)  
{  
case 1:  
printf("Enter the patient Name to Search : ");  
scanf("%s",sname);  
while((fread((char \*)&e, sizeof(e), 1, fp))==1)  
{  
if(strcmp(sname,[e.name](http://e.name/))==0)  
{  
printf("\npatient Number : %d", [e.no](http://e.no/));  
printf("\npatient Name : %s",[e.name](http://e.name/));  
printf("\npatient diagnosis : %s",e.diag);  
printf("\npatient gender : %s",e.gender);  
printf("\npatient age : %d",e.age);  
printf("\n");  
}  
}  
break;  
case 2:  
printf("Enter the patient Number to Search : ");  
scanf("%d", &recno);  
while((fread((char \*)&e, sizeof(e), 1, fp))==1)  
{  
if([e.no](http://e.no/)==recno)  
{  
printf("\npatient Number : %d", [e.no](http://e.no/));  
printf("\npatient Name : %s",[e.name](http://e.name/));  
printf("\npatient diagnosis : %s",e.diag);  
printf("\npatient gender : %s",e.gender);  
printf("\npatient age : %d",e.age);  
printf("\n");  
}  
}  
break;  
}  
}  
} //end of the program

**Output**

****



****

****

