**HOSPITAL MANAGEMENT**

*A case study in particular fullfillment of the requiremet For the award of the degree of*

***Bachelor of computer science***

***Subbmitted***

***By***

***U.ARUN REDDY PRASHANTH KUMAR***

***V.SANDEEP REDDY U.VINOD***

***A.SHIVA KUMAR G.VAMSHI***

***K.SATHISH P.BALA PRANADEEP***

***Mathamatics , statistics and computer science***

***Under the guidance of***

***Mr PM.NAIDU***

***Lecturer***

***Department of computer science***

***DEPARTMENT OF COMPUTER SCIENCE***

***SIVA SIVANI DEGREE COLLEGE***

***(Affiliated to OSUMINA UNIVERSITY)***

***KOMPALLY , MEDCHAL***

***(2016-19)***

***Acknowledgement***

***I consider as my duty to acknowledge some people, without their assistance this work could not have been undertaken at all. I convey deep sense of gratitude to:***

***P.M.NAIDU***

***Head of computer science department***

***SIVA SIVANI DEGREE COLLEGE***

***For providing me the opportunity to undergo the project training.The most drivers and the deepest gratitude I owe to my guides,who has taken the time and effort to help me to prepare this project report.***

**INDEX**

* **INTRODUCTION**
* **EXPLANATION**

1. **KEY FEATURES**
2. **ERD HOSPITAL MANAGEMENT SYSTEM**
3. **MODULES DESCRIPTION**
4. **ADMIN MODULE OF HOSPITAL MANAGEMENT SYSTEM**
5. **ENTERING INTO THE SYSTEM**

* **SOURCE CODE**

1. **ADD NEW PATIENT RECORD**
2. **EDIT OR SEARCH PATIENT RECORD**
3. **LIST OF PATIENT RECORD**
4. **DELETE PATIENT RECORD**

* **OUTPUT SCREENS**
* **SYSTEM USED**

1. **WINDOWS 8.1**

* **CONCLUSION**

**Introduction:**

Hospitals play a major role in a human's life. There is no human on earth that is not going to be suffering some sort of sickness.   
  
  
 It is a very harsh but truth of the life. Nothing is perfect on this earth so does a human. Hospitals provide the best medical facility to the people who are not well.  
  
 It may be due to some sort of stress, changes in climates, work-overload (this is one of the major issue who work in corporate world) or someone might hurt himself / herself.

## HOW CAN IT HELP?

## **There are several other reasons that a person needs medical assistance. And, to provide best medical assistance, the management of the hospital must be disciplined, well-versed in its service providing techniques. They should be able to keep track of the records of the doctors, patients, nurses, and other hospital staffs. But if these records are maintained on the paper, it will not be a cup of tea that can be sipped without burning the lips. It is not very efficient, is not reliable and is very time consuming process. In today’s highly technological era, it is not feasible not by also technically but also economically. So, I thought of making an automated system for keeping the tracks of all the activities and maintaining their records. It is called “Hospital Management System”. My main aim is to minimize the paperwork of the hospital as minimum as possible, if not completely.**

## OBJECTIVE:

* **Computerization** - All the details regarding hospitals, whether it is small or big, will be computerised.
* **Automated inventory** – If the medicines are provided to the patients, the stock will be reduced in the inventory, and will help in to know the status the available medicines.
* **No redundancy** - For every test that is conducted of the patients, an automated report will be generated and will be available to the patients and his / her concerned doctor uniformly.
* **Keep the Records**– It will be easier task for the management to keep the record of the patients for historical purpose.
* **Appointment**– It will be easier for both the doctors and the patients to have the appointments. It is just two clicks away.

**EXPLANATION:**

**Key Features:**

* Multi user account system
* Monitoring the whole hospital system
* Management of all type of user’s account
* Notice Board
* Appointment Management
* View Appointments
* Notifications
* Medical History
* Invoice Management
* Medical Report Management
* Internal Communication
* Responsive User Interfaces

How does it sound now? I know your eyes will be shining like a star in the sky at night. Now, allow me to explain the whole project and the modules involved in it

## ERD FOR HOSPITAL MANAGEMENT SYSTEM:

## 

**MODULES DESCRIPTION:**

There are eight types of users involved in a hospital management system.

* Admin
* Doctor
* Patient
* Nurse
* Receptionist
* Pharmacist
* Accountant
* Laboratorist

Each of the above plays an important role in hospital management system.  
  
If a single of them does not function properly, the management will not be successful.

**ADMIN MODULE OF HOSPITAL MANAGEMENT SYSTEM:**

* It is the most powerful user of the system.
* There will be only one admin into the system.
* The admin can create and manage all other 7 user accounts. He / she can delete any account according to need.
* The admin can monitor all the activities of the hospital. Whatever is going on into the hospital, will be available in the admin panel.
* If other violates the codes of the hospital, he can take immediate actions and can charge that user.

Before discussing about other 7 modules, let me tell you some common features, which will be involved into these modules and their functionalities will be same across all of the seven modules.

**ENTERING INTO THE SYSTEM (REGISTRATION):**

Every single type of user except admin will go through initial phase i.e. registration in order to start with the system.  
  
The registration process will not be exactly same for all type of users. The following are the common requirement fields:

* **Name** – The user will be asked to enter his / her full name.
* **Password** – It is the key field which will help the user to have secure account in the system.
* **Contact No.**– The concerned user should also provide his personal mobile number or residence contact number so that he / she can be contacted at the time of need.
* **Email ID** – Now, email address has become for the for the communication purpose, verifying the users involved in the system. It helps in making the authenticated system.

**VERIFICATION:**

When the user enters all above details in the valid form, he / she should read the terms and conditions, and policy.  
  
If he / she is satisfied with all the statements stated in those terms and conditions, policy page, he / she should tick the checkbox.

When this registration process is completed successfully, a mail will be sent to the email address for the verification purpose.  
  
The user will have to follow simple order of instructions. After completion of these steps, a unique id will be generated and given to the concerned user, which will be mainly used later on as patient ID, doctor ID, accountant ID, nurse ID etc.

**LOGIN:**

This is also the main common feature of the hospital management system.  
 Every type of user will have to enter his / her unique id as login id, which was given to her / him after completing the email verification, and the password. If both login id and password are matched with the credential stored into the system, the user will be granted to access into the system. For every type of user there will be different view and different privileges of the system.

**FORGOT LOGIN ID OR PASSWORD:**

Forgetting is a human nature. None can deny this fact. There might be possible that a user is trying to access the system after a long period of time, and he or she may forget his / her registered login id or password or both of them. I have taken the users’ that concern too into the account so that it is smoother for him / her to recover the login id or password according to need.

* **Via Email**- The users just have to enter their respective valid registered email ids into the field. An immediate email will be sent to their respective email ids with some set of instructions to recover the login id and password.
* **Via Mobile** – In order to recover the password and login id, the user just have enter the registered email id and registered mobile number (both). A 6-digit PIN combination will be sent to the respective mobile number and that PIN must be entered into the appropriate field. If it matches with the generated PIN number, the user will be able to recover login id and password.

**CHANGE PASSWORD:**

Security plays a vital role into any system. Suppose, someone knows the login id and password combination, that would be a threat.

That burglar may misuse the system, steal the private information etc. So, if any user suspects his / her login credentials are compromised, that user must change his / her password immediately.

**Edit Profile**

It is another common feature of this management system. It is not possible to ask all the important information from the user at the time of registration.

This would make the difficult time for the user who is trying to create the account for the first time into the system.  
  
So, a user can enter his / her personal information anytime they want. One of them is address. It is mandatory for some future communications.

It consists of the apartment number, area name, street name, city name, zip code, country name etc. The following are some additional information that needs to be updated:

* Date of Birth
* Age
* Gender
* Height and Weight – (For patients)
* Profile Photo – The user must upload his / her profile photo after first login.
* Designation – (For Hospital staffs)

And, the users can update any of these fields or their contact numbers any time they want.   
Now, I have detailed all the common features for seven modules. Let us get into the other different features of each module.

**Receptionist**  
  
 Receptionist will be the user that will keep tabs of the appointments. This user will maintain the time table of each doctor. This time table of each doctor will be available to the patients.  
  
 The other task the receptionist can do is, filtering the appointments based on doctors and time periods so that it is easy to access the appointment record.

**Accountant**

Accountant is the user who deals with financial transactions of the hospitals. All the payment information and invoices will be managed by the accountant.

**Pharmacist**  
 All the medicines and their information in the hospital are managed by the pharmacists. He / she can view which medicines are in the stocks and which are needed. He / she can filter the medicines based on date i.e. expired date so that it is easier to know which medicines are going to expire so that those can be arranged.  
  
 When the patient provides the prescription to the pharmacist, the pharmacist can provide the appropriate medications to the concerned patient.

**Patient:**

This is the user why hospital management system is introduced. Neither any person will get sick nor will we need a hospital. But you and I both know, this is next to impossible.  
  
 A patient can make an appointment with the available doctors. The time table will be shown which is directed by receptionist. When the doctor approves the appointment, the patient will be notified via the sms.   
  
 The patient can view the doctor’s report about his / her appointment and his / her prescription details. Thus the patient can clearly know what has happened to him / her and how she or he should take care of herself or himself.  
  
 If there are some serious health issues and the patient is needed to be admitted into the hospital, the bed would be allotted to him or her. The patient can view the information about the allotted bed.   
  
 The patient can also have private message session with the doctors.

**Nurse:**

Nurse is an assistant of the doctor. The nurse is helps the doctors in performing the operations. She takes the diagnostic report and provide to the doctor.

**Laboratorist:**

Whenever a patient is asked by doctor to have some tests, so that the doctor can prescribe some medications according the test result, the patient goes to the lab.

There lab assistant diagnoses some tests and generate the reports. This report further is handed over to the nurse’s account.

**Doctor**

Doctor is the user who will help the patients with their physical health. The doctor will be able to approve the appointments whether he or she wants to take it or not.   
  
 If the doctor wants, he or she can cancel the appointments. The concerned patient will be notified via sms.

The doctor can view the medication history of the patients. The doctor can provide prescriptions to the patients.  
  
 The doctor will be able to have private conversation with the patients. There might be chance when the patient needs medical assistance at home or anywhere but hospital. So, I will provide the private message room for the doctor and patient.

# Hospital Management System Project in C

This is a somewhat long but simple mini project in C programming language. Hospital management system project is just a console application without graphics, designed for Alka Hospital, situated in Lalitpur, Nepal.

This project mainly uses file handling to perform basic operations like how to add, edit, search and delete record using file. The source code of hospital management system is over 1100 lines, so the code has not been posted in this article.

You can directly download the source code along with the application file of this project from the link below. The C code is complete and totally error-free. Please make sure that you compile the source code in Code::Blocks.

**Source code**

#include<stdio.h>

#include<conio.h>

#include<string.h>

#include<stdlib.h>

#define rw 80

#define cl 50

FILE\*fp,\*fp1,\*f1,\*f2;

int s,z;

char fn1[]="opd12.patient";

char fn2[]="oopd12.patient";

struct hospital

{

char name[20],address[20],ch;

int age,roomno,sn;

char disease[30],department[20],date[15];

char recommendation[50],category[30];

char test[15][20];

float testfee[15];

float totalfee;

float balance;

}p,q;

char string[20];

typedef struct hospital alka;

void newrecord(int l);

void print();

void displaytest();

void mainscreen();

void newrecord1();

void displaydepartment();

void edit1();

void editrecord();

void switch1();

void main()

{

int a,i,n,y;

char c,d;

char date1[15],date2[15],string1[20];

unsigned int tsz;

clrscr();

mainscreen();

label3:

textcolor(3);

gotoxy(23,15);

cprintf("Enter today's Date(yyyy/mm/dd)");

fflush(stdin);gotoxy(28,19);

scanf("%[^\n]",date2);

if((date2[4]!='/')||(date2[7]!='/')||(date2[5]>'3')||(date2[8]>'3'))

{

clrscr();

mainscreen();

gotoxy(23,13);textcolor(4+128);

cprintf("Wrong Entry");

goto label3;

}

clrscr();

mainscreen();

label1:

textcolor(15);

lowvideo();gotoxy(22,15);textcolor(14);

cprintf("Enter the corresponding no");gotoxy(22,19);textcolor(10);

cprintf("1.Add new patient record");gotoxy(22,21);

cprintf("2.Search or edit record");gotoxy(22,23);

cprintf("3.Know the records of patients");gotoxy(22,25);

cprintf("4.Delete the records");gotoxy(22,27);

cprintf("5.Exit from the program");gotoxy(25,30);

fflush(stdin);

scanf("%c",&d);

switch(d)

{

case '1':

{

{

if((fp=fopen(fn1,"rb"))==NULL)

s=1;

else

{

while(fread(&p,sizeof(alka),1,fp))

s=1+p.sn;

}

fclose(fp);

}

clrscr();

mainscreen();

label:

gotoxy(22,19);textcolor(7);

cprintf("Enter `o' for O.P.D. & `e'for Emergency");

gotoxy(35,21);

fflush(stdin);

scanf("%c",&c);

if(c=='o')

{

clrscr();

mainscreen();

textcolor(11);gotoxy(23,11);

cprintf("ADDING NEW O.P.D.PATIENT RECORD");textcolor(15);

gotoxy(21,12);

cprintf("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");

newrecord(s);

newrecord1();

if((fp=fopen(fn1,"ab+"))==NULL)

{

printf("Cannot open the file f1");

getch();

exit(1);

}

for(a=0;a<15;a++)

p.testfee[a]=0;

strcpy(&p.test[1][0],"0");

p.totalfee=0;p.balance=0;

strcpy(&p.test[0][0],"O.P.D. charge");

p.testfee[0]=200;

p.totalfee=200;

strcpy(p.category,"O.P.D.Patient");

p.balance=200;

strcpy(p.recommendation,"Admitted to O.P.D.");

strcpy(p.date,date2);

fwrite(&p,sizeof(p),1,fp);

fclose(fp);

if((fp=fopen(fn2,"ab+"))==NULL)

{

printf("Cannot open the file f1");

getch();

exit(1);

}

fwrite(&p,sizeof(p),1,fp);

fclose(fp);

}

else if(c=='e')

{

clrscr();

mainscreen();

textcolor(11);

gotoxy(23,11);

cprintf("ADDING NEW EMERGENCY PATIENT RECORD");

textcolor(15);

gotoxy(23,12);

cprintf("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");

newrecord(s);

newrecord1();

if((fp=fopen(fn1,"ab+"))==NULL)

{

printf("Cannot open the file f1");

getch();

exit(1);

}

for(a=0;a<15;a++)

p.testfee[a]=0;

strcpy(&p.test[1][0],"0");

p.totalfee=0;p.balance=0;

p.totalfee=250;

strcpy(p.date,date2);

strcpy(&p.test[0][0],"Emergency Charge");

strcpy(p.category,"Emergency Patient");

strcpy(p.recommendation,"Admitted to Emergency");

p.testfee[0]=250;

p.balance=250;

fwrite(&p,sizeof(p),1,fp);

fclose(fp);

if((fp=fopen(fn2,"ab+"))==NULL)

{

printf("Cannot open the file f1");

getch();

exit(1);

}

fwrite(&p,sizeof(p),1,fp);

fclose(fp);

}

else

{

clrscr();

mainscreen();

gotoxy(22,15);textcolor(128+5);

cprintf("Wrong choice");

textcolor(15);

goto label;

}

break;

}

case '2':

{

clrscr();

mainscreen();

editrecord();

break;

}

case '5':

{

clrscr();

mainscreen();

textcolor(14); gotoxy(30,24);

cprintf("THANK U");gotoxy(30,26);

// cprintf("SAVING UR SETTINGS"); gotoxy(30,28);

cprintf("BYE...........");

getch();

exit(1);

break;

}

case '4':

{

if((fp=fopen(fn1,"rb"))==NULL)

{

printf("cannot open the file");

getch();

exit(1);

}

if((f1=fopen("delete","wb"))==NULL)

{

printf("cannot open the file");

getch();

exit(1);

}

clrscr();

mainscreen();

gotoxy(30,25);

cprintf("Enter the patient no");

gotoxy(40,27);

fflush(stdin);

scanf("%d",&y);

while(fread(&p,sizeof(alka),1,fp))

if(p.sn!=y)

fwrite(&p,sizeof(alka),1,f1);

clrscr();

mainscreen();

fseek(fp,(y-1)\*sizeof(alka),SEEK\_SET);

fread(&p,sizeof(alka),1,fp);

print();

edit1();

gotoxy(25,46);

cprintf("Press `ENTER' to delete this record");

getch();

fclose(fp);

fclose(f1);

remove(fn1);

rename("delete",fn1);

if((fp=fopen(fn2,"rb"))==NULL)

{

printf("cannot open the file");

getch();

exit(1);

}

if((f1=fopen("delete","wb"))==NULL)

{

printf("cannot open the file");

getch();

exit(1);

}

while(fread(&p,sizeof(alka),1,fp))

if(p.sn!=y)

fwrite(&p,sizeof(alka),1,f1);

fclose(fp);

fclose(f1);

remove(fn2);

rename("delete",fn2);

clrscr();

mainscreen();

gotoxy(25,25);textcolor(3);

cprintf("Record succesfully Deleted");

getch();

gotoxy(37,30);

break;

}

case '3':

{ label6:

clrscr();

mainscreen();

gotoxy(22,15);textcolor(12);

cprintf("Enter the corresponding no");gotoxy(22,19);textcolor(3);

cprintf("1.Records of patients in alphabatecal order");gotoxy(22,21);

cprintf("2.Records of Emergency patients");gotoxy(22,23);

cprintf("3.Records of O.P.D. patients");gotoxy(22,25);

cprintf("4.Recordsin paricular date");gotoxy(22,27);

cprintf("5.Return to main menu");gotoxy(25,30);

fflush(stdin);

scanf("%c",&d);

switch(d)

{

case '1':

{

if((fp=fopen(fn2,"rb+"))==NULL)

{

printf("Cannot open the file");

getch();

exit(1);

}

fseek(fp,0,SEEK\_END);

tsz=ftell(fp);

n=(int)(tsz/sizeof(alka));

for(i=0;i<(n-1);i++)

{

for(a=i+1;a<n;a++)

{

fseek(fp,i\*sizeof(alka),SEEK\_SET);

fread(&p,sizeof(alka),1,fp);

fseek(fp,a\*sizeof(alka),SEEK\_SET);

fread(&q,sizeof(alka),1,fp);

if(strcmp(p.name,q.name)>0)

{

fseek(fp,i\*sizeof(alka),SEEK\_SET);

fwrite(&q,sizeof(alka),1,fp);

fseek(fp,a\*sizeof(alka),SEEK\_SET); fflush(stdin);

fwrite(&p,sizeof(alka),1,fp);

}

}

}

rewind(fp);

clrscr();

mainscreen();

gotoxy(3,20);

textcolor(11);

cprintf("Ready to Display the patient records according to alphabatecal order of names");

gotoxy(27,25);textcolor(3);

cprintf("Press");textcolor(15+128);

cprintf(" `Enter' "); textcolor(3);

cprintf("to continue");

getch();

while(fread(&p,sizeof(alka),1,fp))

{

clrscr();

mainscreen();

print();

gotoxy(17,10); textcolor(7);

cprintf("DISPLAYING-RECORD-ACCORDING-TO-PATIENTS-NAMES");

gotoxy(16,11);textcolor(15);

cprintf("---------------------------------------------");

edit1();

textcolor(11);

gotoxy(20,46);

cprintf("Press");textcolor(15+128);

cprintf(" `Enter'"); textcolor(11);

cprintf(" for next and `r' to quit: ");

scanf("%c",&c);

if(c=='r')

{

goto label6;

}

gotoxy(60,46);

getch();

}

clrscr();

mainscreen();

textcolor(11);

gotoxy(30,25);

cprintf("::No Further Records::"); gotoxy(40,30);

getch();

fclose(fp);

break;

}

case '5':

{

clrscr();

mainscreen();

goto label1;

}

case '2':

{

clrscr();

mainscreen();

if((fp=fopen(fn1,"rb"))==NULL)

{

printf("Cannot open the file");

getch();

exit(1);

}

gotoxy(15,20);

textcolor(2);

cprintf("Ready to Display records of Emergency Patients");

gotoxy(27,25);textcolor(3);

cprintf("Press");textcolor(15+128);

cprintf(" `Enter' "); textcolor(3);

cprintf("to continue");

getch();

while(fread(&p,sizeof(alka),1,fp))

{

if(strcmp(p.category,"Emergency Patient")==NULL)

{

clrscr();

mainscreen();

print();

gotoxy(17,10); textcolor(7);

cprintf("::DISPLAYING-RECORDS-OF-EMERGENCY-PATIENTS::");

gotoxy(16,11);textcolor(15);

cprintf("---------------------------------------------");

edit1();

textcolor(11);

gotoxy(20,45);

cprintf("Press");textcolor(15+128);

cprintf(" `Enter'"); textcolor(11);

cprintf(" for next and `r' to quit: ");

scanf("%c",&c);

if(c=='r')

{

goto label6;

}

gotoxy(60,46);

getch();

}

}

clrscr();

mainscreen();

textcolor(11);

gotoxy(30,25);

cprintf("::No Further Records::"); gotoxy(40,30);

getch();

fclose(fp);

break;

}

case '3':

{

clrscr();

mainscreen();

if((fp=fopen(fn1,"rb"))==NULL)

{

printf("Cannot open the file");

getch();

exit(1);

}

gotoxy(15,20);

textcolor(2);

cprintf("Ready to Display records of O.P.D Patients");

gotoxy(27,25);textcolor(3);

cprintf("Press");textcolor(15+128);

cprintf(" `Enter' "); textcolor(3);

cprintf("to continue");

getch();

while(fread(&p,sizeof(alka),1,fp))

{

if(strcmp(p.category,"O.P.D.Patient")==NULL)

{

clrscr();

mainscreen();

print();

gotoxy(17,10); textcolor(7);

cprintf("::DISPLAYING-RECORDS-OF-OPD-PATIENTS::");

gotoxy(16,11);textcolor(15);

cprintf("---------------------------------------------");

edit1();

textcolor(11);

gotoxy(20,46);

cprintf("Press");textcolor(15+128);

cprintf(" `Enter'"); textcolor(11);

cprintf(" for next and `r' to quit: ");

scanf("%c",&c);

if(c=='r')

{

goto label6;

}

gotoxy(60,46);

getch();

}

}

clrscr();

mainscreen();

textcolor(11);

gotoxy(30,25);

cprintf("::No Further Records::"); gotoxy(40,30);

getch();

fclose(fp);

break;

}

case '4':

{

clrscr();

mainscreen();

if((fp=fopen(fn1,"rb"))==NULL)

{

printf("Cannot open the file");

getch();

exit(1);

}

label8:

gotoxy(27,20);

textcolor(3);

cprintf("Enter the `Date' of a paricular day(yyyy/mm/dd)");

gotoxy(35,25);fflush(stdin);

scanf("%s",string);

if((string[4]!='/')||(string[7]!='/')||(string[5]>'3')||(string[8]>'3'))

{

clrscr();

mainscreen();

gotoxy(23,13);textcolor(4+128);

cprintf("Wrong Entry");

goto label8;

}

// getch();

while(fread(&p,sizeof(alka),1,fp))

{

if(strcmp(string,p.date)==NULL)

{

clrscr();

mainscreen();

print();

gotoxy(17,10); textcolor(7);

cprintf("::DISPLAYING-RECORDS-OF-");

cprintf("DATE >%s",p.date);

gotoxy(16,11);textcolor(15);

cprintf("---------------------------------------------");

edit1();

textcolor(11);

gotoxy(20,46);

cprintf("Press");textcolor(15+128);

cprintf(" `Enter'"); textcolor(11);

cprintf(" for next and `r' to quit: ");

scanf("%c",&c);

if(c=='r')

{

goto label6;

}

gotoxy(60,46);

getch();

}

}

clrscr();

mainscreen();

textcolor(11);

gotoxy(30,25);

cprintf("::No Further Records::"); gotoxy(40,30);

getch();

fclose(fp);

break;

}

default:

{

clrscr();

mainscreen();

textcolor(12+128);gotoxy(22,11);

cprintf("Wrong choice");gotoxy(22,13);textcolor(15);

cprintf("Retype choice");

goto label6;

}

}

}break;

default:

{

clrscr();

mainscreen();

textcolor(12+128);gotoxy(22,11);

cprintf("Wrong choice");gotoxy(22,13);textcolor(15);

cprintf("Retype choice");

goto label1;

}

}

clrscr();

mainscreen();

goto label1;

}

void newrecord(int l)

{

char q;

p.sn=l;

displaydepartment();

gotoxy(5,14);textcolor(10);

cprintf("Record of patient no:");

printf(" %d",l);

gotoxy(5,17);

cprintf("Name:");

gotoxy(5,20);

cprintf("Address:");

gotoxy(5,23);

cprintf("Age: ");

gotoxy(5,26);

cprintf("Sex(m/f): ");

gotoxy(5,29);

cprintf("Disease Descrp:");

gotoxy(9,30);

cprintf("(In Short)");

gotoxy(5,33);

cprintf("Reff. Specialist no:");

fflush(stdin);gotoxy(10,17);

scanf("%[^\n]",p.name);

p.name[0]=toupper(p.name[0]);

gotoxy(14,20);

fflush(stdin);

scanf("%[^\n]",p.address);

gotoxy(10,23);

fflush(stdin);

scanf("%d",&p.age);

gotoxy(15,26);

fflush(stdin);

scanf("%c",&p.ch);

fflush(stdin);gotoxy(22,29);

scanf("%[^\n]",p.disease);

}

void newrecord1()

{

char q;

fflush(stdin);

gotoxy(25,33);

scanf("%c",&q);

switch(q)

{

case '1':

{ gotoxy(5,36);

cprintf("Reff.Specialist:");

printf("Generalphysician");

strcpy(p.department,"General Physician");

gotoxy(5,39);

cprintf("Room no:");

fflush(stdin);

scanf("%d",&p.roomno);

// getch();

break;

}

case '2':

{ gotoxy(5,36);

cprintf("Reff.Specialist:");

printf("E.N.T.");

gotoxy(5,39);

cprintf("Room no:");

printf("302");

strcpy(p.department,"E.N.T");

p.roomno=302; getch();

break;

}

case '3':

{ gotoxy(5,36);

cprintf("Reff.Specialist:");

printf("Cardiologist");

gotoxy(5,39);

cprintf("Room no:");

printf("509");

strcpy(p.department,"cardiologist");

p.roomno=509; getch();

break;

}

case '4':

{ gotoxy(5,36);

cprintf("Reff.Specialist:");

printf("Dermatologist");

gotoxy(5,39);

cprintf("Room no:");

printf("406");

strcpy(p.department,"Dermatologist");

p.roomno=406; getch();

break;

}

case '5':

{ gotoxy(5,36);

cprintf("Reff.Specialist:");

printf("Gastroenteroiogist");

gotoxy(5,39);

cprintf("Room no:");

printf("308");

strcpy(p.department,"Gastroentrologist");

p.roomno=308; getch();

break;

}

case '6':

{ gotoxy(5,36);

cprintf("Reff.Specialist:");

printf("Pediatrician");

gotoxy(5,39);

cprintf("Room no:");

printf("207");

strcpy(p.department,"Padiatrician");

p.roomno=207; getch();

break;

}

case '7':

{ gotoxy(5,36);

cprintf("Reff.Specialist:");

printf("EYE Specialist");

gotoxy(5,39);

cprintf("Room no:");

printf("102");

strcpy(p.department,"EYE Specialist");

p.roomno=102; getch();

break;

}

case '8':

{ gotoxy(5,36);

cprintf("Reff.Specialist:");

printf("Nephrologist");

gotoxy(5,39);

cprintf("Room no:");

printf("109");

strcpy(p.department,"Nephrologist");

p.roomno=109; getch();

break;

}

case '9':

{ gotoxy(5,36);

cprintf("Reff.Specialist:");

printf("General Surgeon");

strcpy(p.department,"General Surgeon");

gotoxy(5,39);

cprintf("Room no:");

fflush(stdin);

scanf("%d",&p.roomno);

// getch();

break;

}

case '10':

{ gotoxy(5,36);

cprintf("Reff.Specialist:");

printf("Accumpunturist");

strcpy(p.department,"Accumpunturist");

gotoxy(5,39);

cprintf("Room no:");

fflush(stdin);

scanf("%d",&p.roomno);

// getch();

break;

}

default:

{

gotoxy(5,36);

cprintf("Reff.Specialist:");

fflush(stdin);

scanf("%[^\n]",p.department);

gotoxy(5,39);

cprintf("Room no:");

fflush(stdin);

scanf("%d",&p.roomno);

return;

}

}

}

void displaydepartment()

{

int i;

gotoxy(48,14);textcolor(6);

cprintf(":Specialists:");

gotoxy(47,15);textcolor(15);

cprintf("---------------");

gotoxy(70,14);textcolor(6);

cprintf("Room No.");

gotoxy(67,15);textcolor(15);

cprintf("----------");

gotoxy(48,17);textcolor(5);

cprintf("1.General Physician");

gotoxy(70,17);textcolor(7);

cprintf("201,202");

gotoxy(48,19);textcolor(5);

cprintf("2.E.N.T");

gotoxy(70,19);textcolor(7);

cprintf("302");

gotoxy(48,21);textcolor(5);

cprintf("3.Cardiologist");

gotoxy(70,21);textcolor(7);

cprintf("509");

gotoxy(48,23);textcolor(5);

cprintf("4.Dermatologist");

gotoxy(70,23);textcolor(7);

cprintf("406");

gotoxy(48,25);textcolor(5);

cprintf("5.Gastroenterologist");

gotoxy(70,25);textcolor(7);

cprintf("308");

gotoxy(48,27);textcolor(5);

cprintf("6.Pediatrician");

gotoxy(70,27);textcolor(7);

cprintf("207");

gotoxy(48,29);textcolor(5);

cprintf("7.EYE Specialist");

gotoxy(70,29);textcolor(7);

cprintf("102");

gotoxy(48,31);textcolor(5);

cprintf("8.Nephrologist");

gotoxy(70,31);textcolor(7);

cprintf("109");

gotoxy(48,33);textcolor(5);

cprintf("9.General Surgeon");

gotoxy(70,33);textcolor(7);

cprintf("407,408");

gotoxy(70,35);textcolor(7);

cprintf("412,413");

gotoxy(48,37);textcolor(5);

cprintf("10.Accupuncturist");

gotoxy(70,37);textcolor(7);

cprintf("123,119");

textcolor(15);

for(i=14;i<=45;i++)

{

gotoxy(46,i);

cprintf("|");

}

}

void print()

{

gotoxy(29,10); textcolor(7);

cprintf("DISPLAYING RECORDS");

gotoxy(28,11);textcolor(15);

cprintf("--------------------");

gotoxy(10,14);textcolor(2);

cprintf("Patient no:");gotoxy(21,14);textcolor(11);

cprintf("%d",p.sn);gotoxy(10,17);textcolor(2);

cprintf("Name:");gotoxy(15,17);textcolor(11);

cprintf(" %s",p.name);gotoxy(10,20);textcolor(2);

cprintf("Address:");gotoxy(18,20);textcolor(11);

cprintf(" %s",p.address);gotoxy(10,23);textcolor(2);

cprintf("Age:");gotoxy(15,23);textcolor(11);

cprintf("%d",p.age);gotoxy(10,26);textcolor(2);

cprintf("Sex:");gotoxy(14,26);textcolor(11);

cprintf(" %c",p.ch);gotoxy(10,29);textcolor(2);

cprintf("Date of Reg.:");gotoxy(24,29);textcolor(11);

cprintf("%s",p.date);gotoxy(10,32);textcolor(2);

cprintf("Room no:");gotoxy(18,32);textcolor(11);

cprintf(" %d",p.roomno);gotoxy(10,35);textcolor(2);

cprintf("Department:");gotoxy(21,35);textcolor(11);

cprintf(" %s",p.department);gotoxy(10,38);textcolor(2);

cprintf("Category:");gotoxy(19,38);textcolor(11);

cprintf(" %s",p.category);

}

void edit1()

{

int i,k;

gotoxy(10,41);textcolor(2);

cprintf("Details:");gotoxy(18,41);textcolor(11);

cprintf(" %s",p.recommendation);gotoxy(10,45);textcolor(4);

cprintf("Press");textcolor(15+128);

cprintf(" `Enter'");textcolor(4);

cprintf(" for financial records");

getch();

clrscr();

mainscreen();

textcolor(14); gotoxy(20,8);

cprintf("The list of expenditure of patient no:");

printf("%d",p.sn);

for(z=0,i=0,k=11;p.testfee[z]!=NULL;i++,z++)

{

textcolor(15);gotoxy(20,k);

cprintf("%d)",i+1); textcolor(11);

cprintf(" %s",&p.test[z][0]);textcolor(15);gotoxy(40,k);

cprintf("Rs.%0.2f",p.testfee[z]);

k=k+2;

}

gotoxy(20,k+2); textcolor(11);

cprintf("Total charge=");textcolor(15);

cprintf("Rs.%0.2f",p.totalfee);

gotoxy(20,k+4);textcolor(11);

cprintf("Total Deposited:");textcolor(15);

cprintf("Rs.%0.2f",p.balance);

if(p.totalfee>p.balance)

{

textcolor(11); gotoxy(20,k+6);

cprintf("Total money to pay=");

textcolor(15);

cprintf("%0.2f",p.totalfee-p.balance);

}

else

{

textcolor(11); gotoxy(20,k+6);

cprintf("Total money to return=");

textcolor(15);

cprintf("%0.2f",p.balance-p.totalfee);

}

}

void switch1()

{

int x,i;

float d,lk;

char v;

label2:

gotoxy(30,15);textcolor(11);

cprintf("Enter");gotoxy(30,18);

cprintf("1.Add Details");

gotoxy(30,20);

cprintf("2.Change Specialist");

gotoxy(30,22);

cprintf("3.Deposit balance");gotoxy(30,24);

cprintf("4.Add test");

gotoxy(30,26);

cprintf("5.Return to main menu");

gotoxy(35,39);

fflush(stdin);

scanf("%d",&x);

switch(x)

{

case 1:

{

clrscr();

mainscreen();

gotoxy(31,10); textcolor(7);

cprintf("ADDING DETAILS");

gotoxy(30,11);textcolor(15);

cprintf("----------------");

gotoxy(15,25);

textcolor(2);

cprintf("Previous Details:");

gotoxy(32,25);textcolor(11);

cprintf(" %s ",p.recommendation);

gotoxy(15,28);textcolor(2);

strcat(p.recommendation,"-> ");

printf("New Details:");

fflush(stdin);

scanf("%[^\n]",string);

strcat(p.recommendation,string);

break;

}

case 2:

{

clrscr();

mainscreen();

displaydepartment();

gotoxy(28,10); textcolor(7);

cprintf("CHANGING SPECIALIST");

gotoxy(27,11);textcolor(15);

cprintf("---------------------");

gotoxy(5,20);

textcolor(2);

cprintf("Previous Specialist:");

gotoxy(26,20);textcolor(11);

cprintf(" %s ",p.department);

textcolor(15);gotoxy(5,27);

cprintf(":Enter the new changed Specialist;");

gotoxy(5,33); textcolor(2);

cprintf("Reff. Specialist no:");

newrecord1();

break;

}

case 3:

{

clrscr();

mainscreen();

gotoxy(28,10); textcolor(7);

cprintf("DEPOSITING MONEY");

gotoxy(27,11);textcolor(15);

cprintf("---------------------");

gotoxy(20,20);

textcolor(2);

cprintf("Deposited balance: Rs.");

fflush(stdin);

scanf("%f",&d);

p.balance=p.balance+d;

break;

}

case 4:

{

clrscr();

mainscreen();

displaytest();

gotoxy(45,13);textcolor(3);

cprintf("Type `esc' tn content to stop");

gotoxy(45,15);textcolor(15);

cprintf("Contents");gotoxy(70,15);

cprintf("Rs.");lk=0;

for(i=17;1;i++,z++)

{

gotoxy(45,i);

fflush(stdin);

cprintf("%c ",16);

scanf("%s",&p.test[z][0]);

if(strcmp(&p.test[z][0],"esc")==0)

{ labelh:

gotoxy(48,i+3);textcolor(15);

cprintf("Paid/Unpaid (p/u):");

scanf("%c",&v);

if(v=='p')

{

p.balance=p.balance+lk;

}

else if(v!='u')

{ gotoxy(48,i+2);

// textcolor(4+128);

// cprintf("Wrong Choice");

goto labelh;

}

break;

}

gotoxy(70,i);

fflush(stdin);

scanf("%f",&p.testfee[z]);

lk=lk+p.testfee[z];

p.totalfee=p.totalfee+p.testfee[z];

i++;

}

break;

}

case 5:

{

return;

}

default:

{

clrscr();

mainscreen();gotoxy(30,13);textcolor(4+128);

cprintf("Wromg choice");

goto label2;

}

}

clrscr();

mainscreen();

goto label2;

}

void editrecord()

{

int y,x;

gotoxy(20,20);textcolor(11);

cprintf("Enter the patient `record no' or `Full name': ");gotoxy(25,24);

fflush(stdin);

scanf("%[^\n]",string);

string[0]=toupper(string[0]);

y=atoi(string);

clrscr();

mainscreen();

if((fp=fopen(fn1,"rb+"))==NULL)

{

printf("\n cannot open the record file 1");

getch();

exit(1);

}

// rewind(fp);

while(fread(&p,sizeof(p),1,fp))

{

if((p.sn==y)||(strcmp(p.name,string)==0))

{

print();

edit1();gotoxy(22,46);textcolor(9);

printf("Press `1'to edit or add and any key togo to main menu");

fflush(stdin);

scanf("%d",&x);

if(x==1)

{

clrscr();

mainscreen();

switch1();

}

else

return;

rewind(fp);

fseek(fp,(p.sn-1)\*sizeof(p),SEEK\_SET);

fwrite(&p,sizeof(p),1,fp);

fclose(fp);

break;

}

}

if(p.sn!=y)

{

gotoxy(22,25);textcolor(3);

cprintf("There is no record available"); gotoxy(22,30);

textcolor(15);

cprintf("Press Enter to continue:"); getch();

}

}

void displaytest()

{ int i;

gotoxy(50,10);textcolor(14);

cprintf("::ADDING SERVICES::");gotoxy(45,11);textcolor(15);

cprintf("----------------------------");

gotoxy(9,10);textcolor(6);

cprintf(":PRICE LIST:");

textcolor(15);

gotoxy(4,11);

cprintf("---------------------");

gotoxy(5,14);textcolor(10);

cprintf("1.X-Ray");

gotoxy(30,14);textcolor(11);

cprintf("Rs. 300.00");

gotoxy(5,16);textcolor(10);

cprintf("2.Ultra-sound");

gotoxy(30,16);textcolor(11);

cprintf("Rs. 500.00");

gotoxy(5,18);textcolor(10);

cprintf("3.C.T.Scan");

gotoxy(30,18);textcolor(11);

cprintf("Rs.1800.00");

gotoxy(5,20);textcolor(10);y

cprintf("4.TC,DC,HB");

gotoxy(30,20);textcolor(11);

cprintf("Rs. 90.00");

gotoxy(5,22);textcolor(10);

cprintf("5.Urine R/E");

gotoxy(30,22);textcolor(11);

cprintf("Rs. 80.00");

gotoxy(5,24);textcolor(10);

cprintf("6.Blood C/S");

gotoxy(30,24);textcolor(11);

cprintf("Rs. 250.00");

gotoxy(5,26);textcolor(10);

cprintf("7.Biliribin D/T");

gotoxy(30,26);textcolor(11);

cprintf("Rs. 200.00");

gotoxy(5,28);textcolor(10);

cprintf("8.Stool Test");

gotoxy(30,28);textcolor(11);

cprintf("Rs. 60.00");

gotoxy(5,30);textcolor(10);

cprintf("9.Bed Charge");

gotoxy(30,30);textcolor(11);

cprintf("Rs. 150.00");

gotoxy(5,32);textcolor(10);

cprintf("10.Sugar Test");

gotoxy(30,32);textcolor(11);

cprintf("Rs. 400.00");

textcolor(15);

for(i=10;i<=46;i++)

{

gotoxy(41,i);

cprintf("|");

}

}

void mainscreen()

{

int i,j;

clrscr();

for(i=2,j=2;i<rw;j++)

{

if(j>15)

j=2;

textcolor(j);

gotoxy(i,2);

cprintf("%c",15);

gotoxy(i,cl-1);

cprintf("%c",15);

i++;

}

for(i=2,j=2;i<cl;i++,j++)

{

if(j>15)

j=2;

textcolor(j);

gotoxy(2,i);

cprintf("%c",15);

gotoxy(rw-1,i);

cprintf("%c",15);

}

gotoxy(30,4);textcolor(3);

cprintf("siva sivani hispital");

gotoxy(27,6);textcolor(6);

cprintf("vensai,kompally");

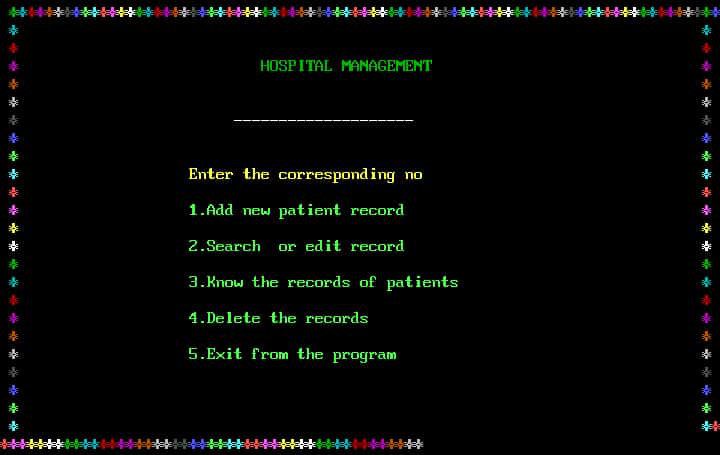
gotoxy(27,7); textcolor(15);

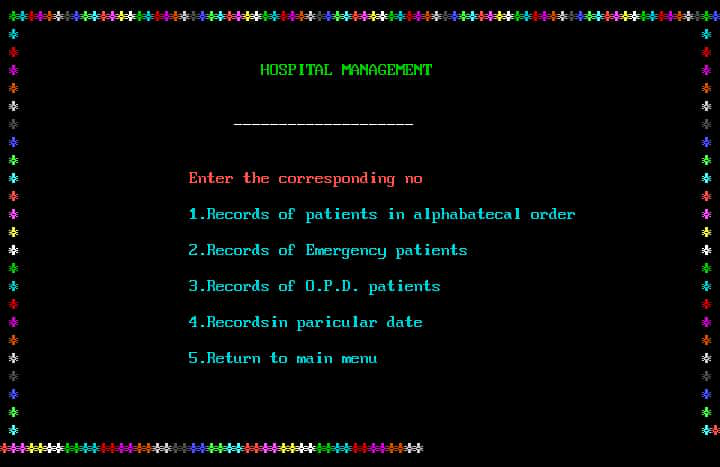
cprintf("--------------------");

}

**Output screens**

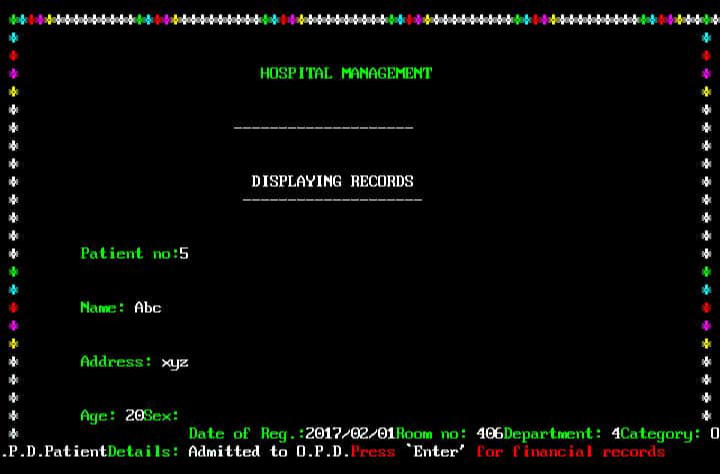
****

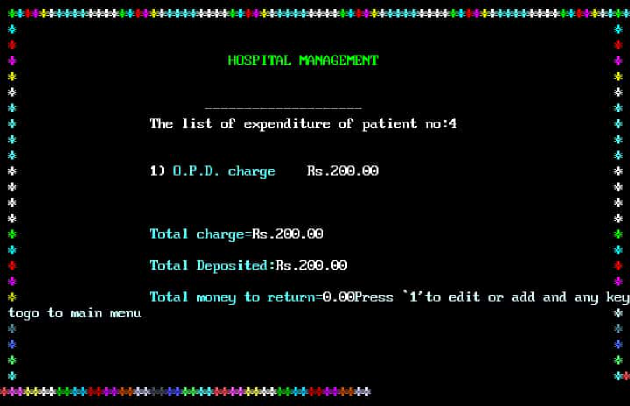














**SYSTEM USED:**

**WINDOWS 8.1:**

**Windows 8.1** (codenamed **Blue**) is an upgrade for windows 8, a computer operation system released by Microsoft. First unveiled and released as a public beta in June 2013, it was released to manufacturing on August 27, 2013, and reached general availability on October 17, 2013, almost a year after the retail release of its predecessor. Windows 8.1 is available free of charge for retail copies of Windows 8 and Windows RT users via the Windows Store. Unlike service packs on previous versions of Windows, users who obtained Windows 8 outside of retail copies or pre-loaded installations (i.e., volume licensing) must obtain Windows 8.1 through new installation media from their respective subscription or enterprise channel. Microsoft's support lifecycle policy treats Windows 8.1 similar to previous service packs of Windows: It is part of Windows 8's support lifecycle, and installing Windows 8.1 is required to maintain access to support and Windows updates after January 12, 2016. However, unlike previous service packs, Windows 8.1 cannot be acquired via Windows Update.

Released as part of a shift by Microsoft towards regular yearly major updates for its software platforms and services, Windows 8.1 aims to address complaints of Windows 8 users and reviewers on launch. Visible enhancements include an improved Start screen, additional snap views, additional bundled apps, tighter OneDrive (formerly SkyDrive) integration, Internet Explorer 11, a Bing-powered unified search system, restoration of a visible Start button on the taskbar, and the ability to restore the previous behavior of opening the user's desktop on login instead of the Start screen. Windows 8.1 also added support for such emerging technologies as high-resolution displays, 3D printing,Wi-Fi Direct,and Miracast streaming, as well as the ReFS

Files used

**Conclusion**

**About Hospital Management System C Project:**

This mini project, like any other, is built without graphics and comes with simple file handling operations. The key features in hospital management system are:

**1. Add new patient record:**

In this feature, user can add a new patient record choosing between O.P.D. service and Emergency service. In O.P.D. service (shown in output screen below), name, address, age, sex, disease description and specialist room number to be referred are available. Thus given information is stored in file. The information to be given are same in Emergency service.

**2. Search or edit patient record:**

In this mini project, two features – searching and editing patient records are placed under a single feature. User can search or edit via. record number of the patient or his/her full name.

All the information corresponding to the respective patient are displayed. These include the ones provided while adding a new patient record. If wrong information about record number or patient full name is provided, the program displays a message saying that no records were available.

Also, user can view the list of expenditures of the particular patient whose record is sought. In the program, the financial information include total charge, total deposited and total money to return.

**3. List record of patients:**

In hospital management system c project, users can list patient records by choosing any one of the four options listed below:

* Records of patients in alphabetical order
* Records of Emergency patients
* Records of O.P.D. patients
* Records of patients in a particular date

As in the two features mentioned above, user can view financial records corresponding to any particular patient listed according to any one of the four options mentioned above. Pressing ‘Enter’ user can view the records of next patients.

**4. Delete patient records:**

This features allows user to delete added record of any patient. For this the patient number to be removed is to be provided. Upon ‘Enter’, user can view the patient record and the financial records of the patient. To delete the record, press ‘Enter’ and the respective patient record will be deleted from the file.