

HOSPITAL MANAGEMENT SYSTEM

A

Mini Project Report

Submitted in partial fulfilment of the

Requirements for the award of the Degree of

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

By

GUGGILLA SHRI PALLAVI - 1602-19-737-171

NATHADI NANDITHA - 1602-19-737-141



Department of Information Technology

Vasavi College of Engineering(Autonomous)

(Affiliated to Osmania University)

Ibrahimbagh, Hyderabad-31

2020

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

2020

Hyderabad-500 031

Department of Information Technology



DECLARATION BY THE CANDIDATE

We, GUGGILLA SHRI PALLAVI and NATHADI NANDITHA bearing hall ticket numbers, 1602-19-737-171 and 1602-19-737-141 respectively, hereby declare that the project report entitled "HOSPITAL MANAGEMENT SYSTEM" is submitted in partial fulfillment of the requirement for the award of the degree of Bachelor of Engineering in Information Technology.

This is a record of bonafide work carried out by us and the results embodied in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

GUGGILLA SHRI PALLAVI
1602-19-737-171

NATHADI NANDITHA
1602-19-737-141

(Faculty In-Charge)

(Head, Dept. of IT)

AKNOWLEDGEMENTS

“Task successful” makes everyone happy. But the happiness will be gold without glitter if we didn’t state the persons who have supported us to make it a success. We are extremely thankful to our college, Vasavi College of Engineering, for providing the opportunity to implement our mini-project in C.

We would like to express our sincere thanks and gratitude to our Project Coordinator, Ms. Divya Lingineni, Assistant Professor, Department of Information Technology, Vasavi College of Engineering and Dr. Ramesh Vasappanavara, Professor, Department of Information Technology, Vasavi College of Engineering, for their esteemed guidance, moral support and invaluable advice provided by them for the success of the Mini Project.

On a more personal note we thank our friends Soma Santosh Kumar, Naveen Manda and A. Sravya Reddy for providing seamless help and support.

Sincerely,

GUGGILLA SHRI PALLAVI
1602-19-737-171

NATHADI NANDITHA
1602-19-737-141

ABSTRACT

Hospitals play an essential role in our lives, as they provide best medical facilities to the people who are suffering from various medical conditions. It is necessary for the hospitals to keep track of their day-to-day activities and records of their patients, doctors, staff members etc. that keep the hospital running smoothly and successfully. As the population is increasing, the number of people visiting the hospital is also increasing. Due to the current COVID-19 situation, numerous people visit hospitals daily. But, keeping track of these records on paper is very inefficient and a time-consuming process which may sometimes even result in errors. It is also not economically and technically convenient to maintain records on paper. Thus, keeping the working and difficulties of manual system as the basis of our project, we want to develop an automated version of the manual system which makes it easier and less stressful for the users to use. The main aim is to provide a project which keeps track of all the records and enables the user get the required information within seconds. Hence, it saves time and is more efficient.

Table of Contents

1. Introduction	1
1.1 Problem Domain	1
1.2 About the Project	1
1.3 Features	2
1.3.1 Patient	2
1.3.2 Admin	2
1.3.3 Receptionist	2
1.3.4 Doctor	3
2. Technology	4
2.1 Software Requirements	4
2.2 Hardware Requirements	4
3. Proposed Work	5
3.1 Design	5
Use Case Diagram	5
3.1.1 Patient Use Cases	6
3.1.2 Admin Use Cases	8
3.1.3 Receptionist Use Cases	11
3.1.4 Doctor Use Cases	11
3.2 Implementation	12
3.2.1 Module-Wise Code	12
3.2.2 Github/Folder Structure	66
3.3 Testing	67
3.3.1 Test Plan	67
3.3.2 Patient Test Cases	67
3.3.3 Admin Test Cases	71
3.3.4 Receptionist Test Cases	77
3.3.5 Doctor Test Cases	79

4. Results	81
4.1 Patient Test Results	81
4.2 Admin Test Results	88
4.3 Receptionist Test Results	98
4.4 Doctor Test Results	103
5. Additional Knowledge Acquired	108
6. Conclusion and Future Work	109
7. References	110

1. INTRODUCTION

1.1. PROBLEM DOMAIN

Healthcare is the most critical aspect of our society, and many health care providers face challenges to offer practical and active services to patients. Considering a multi specialty hospital, many people enter and exit the hospital in a day and maintaining their records safely is tedious. Apart from that automating your hospital's processes and implementing them cannot be done too easily, you need an efficient hospital management system to take care of everything that is happening around the labs and hospitals. It is really difficult to maintain medical and contact details of every single patient, to keep track of appointments and many manually. Lots of errors takes place in such a huge network, be it economical or material wastage. Even a small error in pieces of information is misplaced creates a major problem.

1.2. ABOUT THE PROJECT

“Hospital Management System is a console-based C Project which acts as a **system** for managing the **Hospital** functions and events. It has different modules such as adding new doctors, managing patients and their details and managing appointments etc. to reduce all the types of burdens and to manage hospital administration, Hospital management system came into existence. Our project can be used by anyone in the Hospital i.e. Patients, Admins, Receptionists and Doctors easily and effectively.

1.3. FEATURES

Our vision for this project was to provide a platform for everyone in the Hospital to use it so that there would be no waste of money and energy

To be simple, the features that we have offered can be accessed by Patients, Admin, Receptionists and Doctors for doing their respective tasks smoothly and successfully.

1.3.1. PATIENT

In our C project, the first Actor is a Patient. The Patient can Register or Login to do his tasks. They can then view the List of Doctors and their Specializations, Make and Check Appointments and even Unregister whenever they want.

1.3.2. ADMIN

Our second Actor is an Admin. The Admin has more number of tasks to do as he has to do all the additions and modifications of the details of both the Patients and the Doctors. An Admin can add, modify, view and delete the details of a Patient. He can also add a Doctor, and also Give and Check Appointments.

1.3.3. RECEPTIONIST

A Receptionist is an Actor in our project where he/she can only view the details of the List of Registered Patients, Details of a Particular Patient, Appointments of a Patient and the Appointments of Doctors left. A Receptionist cannot make any changes in the details.

1.3.4. DOCTOR

The Last Actor of our Project is a Doctor. The Doctors can Login, Check the Patients assigned to them and even Cancel the Appointments whenever they are busy or have an emergency.

2. TECHNOLOGY

All computer software needs certain hardware components or other software resources to be present, in order for computers to be used efficiently. These prerequisites are known as System Requirements. Within this, we have two types – Software Requirements and Hardware Requirements.

2.1. SOFTWARE REQUIREMENTS

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

In order to use HMS, one should have the following:

- **Operating System:** Windows 7 and above
- **C Compiler:** GNU Compiler Collection(GCC)
- **Editor:** Any text editor (preferably Visual Studio Code or Notepad++)

2.2. HARDWARE REQUIREMENTS

Hardware requirements refer to the common set requirements defined by any operating system or software application and are usually the physical computer resources. In this, we look into the architecture, processing power, memory, secondary memory, display adapter and peripherals.

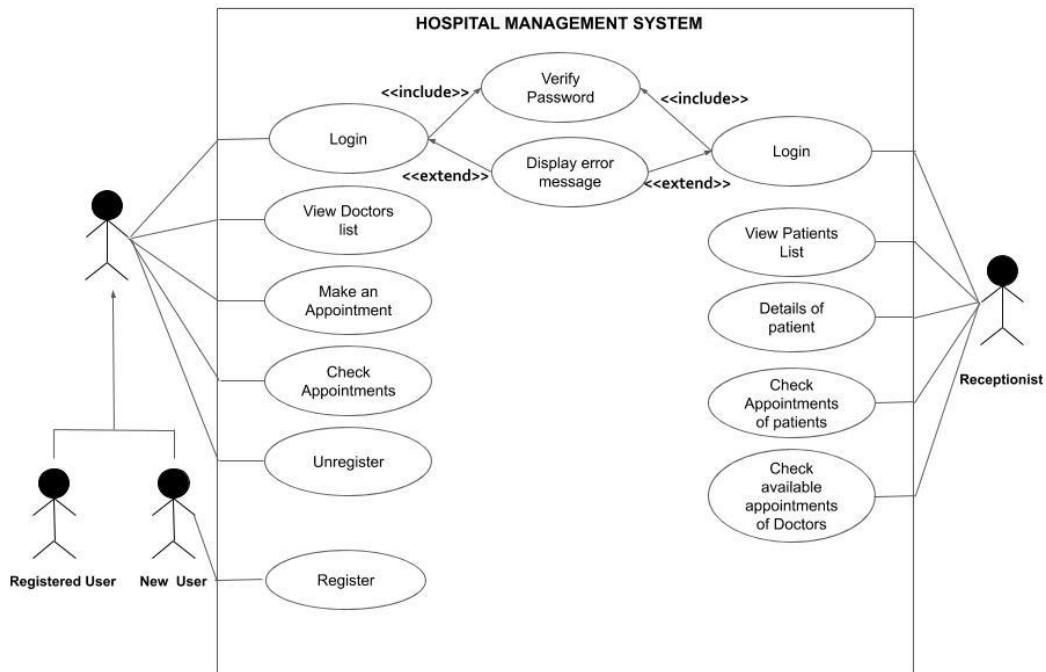
In order to use HMS, one should have the following:

- **Processor:** Intel Core i3 and above
- **Memory:** 4 GB RAM and above

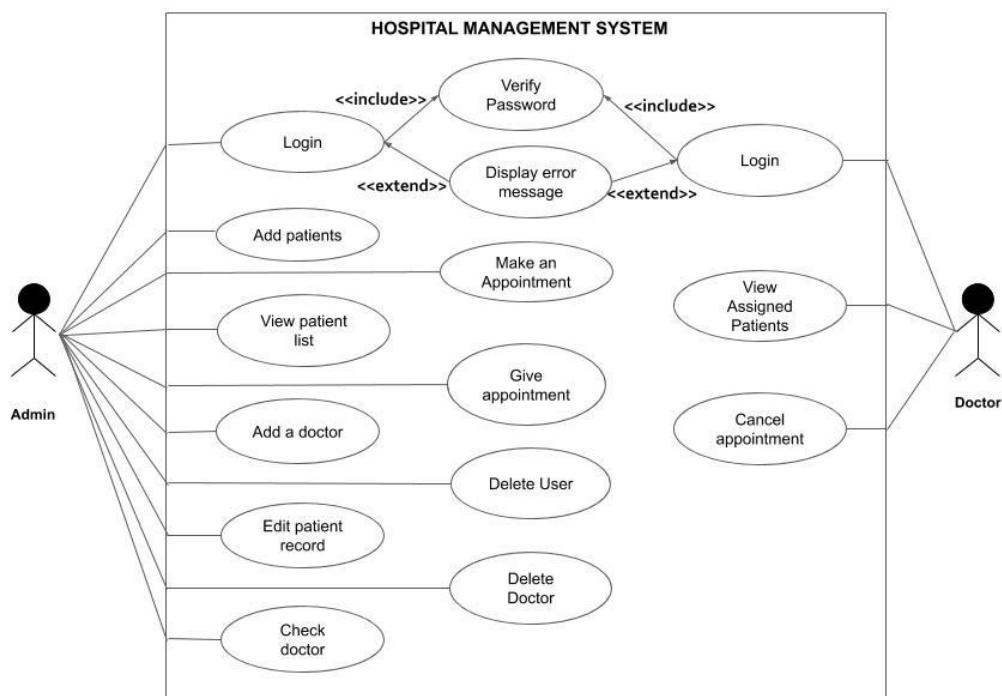
3. PROPOSED WORK

3.1 DESIGN USECASE DIAGRAMS

➤ Patient and Receptionist



➤ Admin and Doctor



3.1.1. PATIENT USECASES

3.1.1.1. REGISTER

Anyone can register as a Patient in the Hospital Management System. A new patient when chooses that he/she is new to the Hospital must consult the Admin for creating an account and the Admin adds the Details of the Patient after Creating an Account.

3.1.1.2. LOGIN

In order for a user to login to HMS, he/she must be a registered user. First, the system will prompt the user to enter his/her credentials (username and password). Once, the Patient enters his/her credentials, we check whether it matches with the information in the files. If a match is found, the user gets access to the extra features accessible by the Patients in the Hospital.

3.1.1.3. VIEW DOCTOR'S LIST

Once the Patient is logged in, he/she can choose to view the list of all the doctors present in the hospital along with their specializations. It helps patients choosing the doctor accordingly, like in case of making an appointment.

3.1.1.4. MAKE AN APPOINTMENT

After logging in, the patient can choose to make an Appointment. First, the patient should choose the doctor they want to be treated by and then they can confirm it by typing YES. It displays that an appointment is booked if there are vacant slots else it will display a message that the doctor doesn't have vacant slots. It even displays a message if he/she enters a wrong doctor name.

3.1.1.5. CHECK APPOINTMENTS

Once the patient logs in, he/she can choose to check their Appointments. It will display the names of the doctors if there are any appointments, and displays a No appointments message if there are no Appointments.

3.1.1.6. UNREGISTER

A user can unregister from HMS, whenever they want to. When a patient wants to get out of Hospital (HMS) they can unregister from HMS. Thereby, the details of the patient will be deleted from the database. Next time if the same patient again newly registers and logs in then their previous details cannot be accessed

3.1.2. ADMIN USE CASES

3.1.2.1. LOGIN

The Admin can login if he/she has the required credentials. The system first verifies whether the credentials entered match the Admin credentials. If they do, then the Admin gets complete access to the HMS. If they don't match, the required message is displayed on the console.

3.1.2.2. ADD A PATIENT

After the Admin logs in, he/she can add a new patient by creating him/her an account in the hospital by entering all the required details and finally display the credentials of the patient, so that he/she can login and use the HMS whenever necessary.

3.1.2.3. PATIENT LIST

After the Admin logs in, he/she can view the list which contains all the information of all registered patients. This information includes full name, username, password, address, age, gender, contact, email and problem. We have included this functionality for ensuring easy access of patient information for the Admin.

3.1.2.4. ADD A DOCTOR

After the Admin logs in, he/she can add a new doctor by creating him/her an account in the hospital by entering all the required details and finally display the credentials of the doctor, so that he/she can login and use the HMS.

3.1.2.5. EDIT PATIENT RECORD

Once logged in as the Admin, he/she can choose to edit any patients' details. First, the Admin needs to enter the username of the patient whose details he/she has to edit. Then the Admin has to enter the new details and make the changes in their records.

3.1.2.6. CHECK DOCTORS

After the Admin logs in, he/she can view the list which contains all the information of all registered doctors. This information includes name of the doctor and their specialization. We have included this functionality for ensuring easy access of doctor information for the Admin.

3.1.2.7. GIVE APPOINTMENT

After logging in, the admin can choose to give an Appointment. First, he/she should enter the username of the patient and then choose the doctor of their choice and then they can confirm it by typing YES. It displays that an appointment is booked if there are vacant slots else it will display a message that the doctor doesn't have vacant slots. It even displays a message if he/she enters a wrong doctor name.

3.1.2.8. DELETE USER

The Admin has the authority to delete a user. The Admin may have different reasons for deleting a user. Whatever the reason, if the Admin needs to delete a user, he/she has to enter the username of the patient they want to remove from the hospital permanently. Then, the system deletes the chosen patient's data.

3.1.2.9. DELETE A DOCTOR

The Admin has the authority to delete a user. The Admin may have different reasons for deleting a doctor. Whatever the reason, if the Admin needs to delete a doctor, he/she has to enter the username of the doctor they want to remove from the hospital permanently.

Then, the system deletes the chosen doctor's data.

3.1.3. RECEPTIONIST USE CASES

3.1.3.1. PATIENT'S LIST

After the receptionist logs in, he/she can view the list which contains all the information of all registered patients. This information includes full name, username, password, address, age, gender, contact, email and problem. We have included this functionality for ensuring easy access of patient information for the Receptionist.

3.1.3.2. VIEW PATIENT'S DETAILS

After the receptionist logs in, he/she can view the details of a particular patient which contains all the information of all registered patients. This information includes full name, username, password, address, age, gender, contact, email and problem. We have included this functionality for ensuring easy access of patient information for the Receptionist.

3.1.3.3. CHECK PATIENT APPOINTMENTS

Once the receptionist logs in, he/she can choose to check the Appointments of a particular patient. He/she should enter the name of a patient and it will display the names of the doctors if there are any appointments, and displays a No appointments message if there are no Appointments.

3.1.3.4. CHECK DOCTOR'S APPOINTMENTS KEFT

Once the receptionist logs in, he/she can choose to check the Appointments of a particular patient. He/she should enter the name of a doctor and it will display the names of the doctors and the number of their appointments for that day.

3.1.4. DOCTOR USE CASES

3.1.4.1. MY PATIENTS

A logged in Doctor can view the list of his patients. List of the patients for the day, thereby he can search their details if required. Through my patients doctor can have an idea about his patients which helps in better interaction and solving the problem.

3.1.4.2. CANCEL APPOINTMENT

After logging in, doctor can cancel appointments. Due to any issue if the doctor isn't going to be available then he/she can cancel appointments as required .This helps patients save time ,by not visiting hospital for appointment on that day. This improves efficiency of HMS.

3.2. IMPLEMENTATION

3.2.1. MODULE-WISECODE

```
//MINI PROJECT - HOSPITAL MANAGEMENT SYSTEM  
//1602-19-737-171 - G. Shri Pallavi  
//1602-19-737-141 - N. Nanditha Reddy
```

```
//HEADER FILES  
#include<windows.h>  
#include<stdlib.h>  
#include<unistd.h> //For using sleep()  
#include<string.h> //For strcmp(),strcpy(),strlen(),etc  
#include<stdio.h> //For standard I/O Operation  
#include<conio.h> //For delay(),getch(),gotoxy(),etc.  
#include<ctype.h> //For toupper(), tolower(),etc  
#include<time.h> //For date, time etc
```

```
//FUNCTION DECLARATIONS
```

```
//WelcomeScreen Function  
void WelcomeScreen(void);
```

```
//Title Function  
void Title(void);
```

```
//RegisterLogin Function  
void RegisterLogin(void);
```

```
//Exit Function  
void Exit(void);
```

```
//Actor1 - Patient
void PLogin();
void Pat_Register();
void Pat_Login();
void PatientMenu(char[]);
void SearchDoctor();
void MakeAppointment(char[]);
void CheckAppointment();
void Unregister(char[]);

//Actor2 - Admin
void ALogin();
void AdminMenu();
void AddPatient();
void PatientList();
void AddDoctor();
void EditPatientRecord();
void CheckDoctors();
void GiveAppointment();
void DeleteUser();
void DeleteDoctor();

//Actor3 - Receptionist
void RLogin();
void ReceptionistMenu();
void PatientsList();
void ViewPatientDetails();
void CheckPatientAppointments();
void CheckDoctorsAppLeft();

//Actor4 - Doctor
void DLogin();
void DoctorMenu();
void MyPatients(char[]);
void CancelAppointment();
```

```

//Defines gotoxy() for ANSI C compilers.
void gotoxy(short x, short y)
{
    COORD pos = {x, y}; //sets co-ordinates in (x,y).
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), pos);
}

//Structure for Patient
struct Patient
{
    char Username[30];
    char Password[20];
    int Age;
    char Gender;
    char Name[20];
    char Contact[15];
    char Address[30];
    char Email[30];
    char Problem[50];
};

struct Patient p, temp_c, l, iter;

//Structure for Doctor
struct doctor
{
    char Username[30];
    char Password[20];
    int Age;
    char Gender;
    char Name[30];
    char Contact[15];
    char Address[30];
    char spec[30];
    char appointments;
};

struct doctor d, iter1;

```



```

        break;
    }
}

strcpy(pwd,password);
fp = fopen("Patients.txt", "r");
while(fscanf(fp,"%s %s %s %c %d %s %s %s %s\n", iter.Username,iter.Password, it
er.Name, &iter.Gender, &iter.Age, iter.Address, iter.Contact, iter.Email, iter.Problem)! =EOF)
{
    if(strcmp(iter.Username, username)==0)
    {
        flag = 1;
        if (strcmp(iter.Password, pwd)==0)
        {
            printf("\n\n\n\n\t\t\t\t\t-----Logging in-----
---");
            Sleep(1000);
            fclose(fp);
            PatientMenu(username);
        }
        else
        {
            fclose(fp);
            printf("\n\n\n\n\n\n\t\t\t-----
WRONG PASSWORD...PLEASE ENTER AGAIN -----");
            Sleep(1000);
            Pat_Login();
        }
    }
    if(flag == 0)
    {
        printf("\n\n\n\n\n\t\t\t-----
USERNAME NOT FOUND...Please Login Again...-----");
        fclose(fp);
        Sleep(2000);
        Pat_Login();
    }
}

```



```
    default:  
        printf("\n\n\n\n\t\t\t-----Invalid Entry...Choose Again----  
-----");  
        getch();  
        break;  
    }  
}
```

//Function for SearchDoctor -

Displays the list of all Doctors with their Specialisations and the Patients can view it
void SearchDoctor(char username[30])

//Function for MakeAppointment =

Patients can Make an Appointment with a particular Doctor after viewing the List

```
void MakeAppointment(char username[30])
```

f

```
system("cls");
```

Title():


```

left--;
itoa(left,left1,10);
printf("\n\n\n\t\t\t\t\t-----APPOINTMENT BOOKED-----
-----");
fprintf(fp2,"%s %s\n",doctorName,left1);
for(int i = 0;i<25;i++)
{
    if(ctime(&t)[i] == ' ')
    {
        time[i] = '-';
    }
    else
    {
        time[i] = ctime(&t)[i];
    }
}
fprintf(fp,"%s %s %s\n",username,doctorName,time);
}
else
{
    printf("\n\n\n\t\t\t\t\t-----NO APPOINTMENTS LEFT FOR THIS DOCTOR-----");
    fprintf(fp2,"%s %s\n",doctorName,appLeft);
}
}
else
{
    fprintf(fp2,"%s %s\n",doctorName,appLeft);
}
}
if(flag != 1)
{
    printf("\n\n\n\t\t\t\t\t-----INVALID DOCTOR NAME-----
-----");
}
fclose(fp1);
fclose(fp2);

```



```

        printf("\n\n\t\t-----\nYOU DID NOT BOOK ANY APPOINTMENTS-----");
    }
    fclose(fp);
    getch();
    PatientMenu(username);
}

//Function for Unregister -
Patients can Unregister from the Hospital whenever they need
void Unregister(char username[30])
{
    system("cls");
    Title();
    FILE *fp,*fp1;
    fp = fopen("Patients.txt","r");
    fp1 = fopen("temp.txt","w+");
    while(fscanf(fp,"%s %s %s %c %d %s %s %s %s\n", iter.Username,iter.Password, it
er.Name, &iter.Gender, &iter.Age, iter.Address, iter.Contact, iter.Email, iter.Problem)!
=EOF)
    {
        if(strcmp(iter.Username,username) == 0)
        {
            printf("\n\n\n\n\t\t-----\nYOUR ACCOUNT IS SUCCESSFULLY DELETED-----\n");
            printf("\n\n\n\n\t\t----- Enter any key to go back to Main Menu...\"");
        }
        else
        {
            fprintf(fp1,"%s %s %s %c %d %s %s %s %s\n", iter.Username, iter.Password, it
er.Name, iter.Gender, iter.Age, iter.Address, iter.Contact, iter.Email, iter.Problem);
        }
    }
    fclose(fp);
    fclose(fp1);
    remove("Patients.txt");
    rename("temp.txt","Patients.txt");
}

```



```

    {
        l--;
        password[l] = '\0';
        printf("\b \b");
    }
}

if(ch == 13)
{
    break;
}

strcpy(Password,password);

if (strcmp(Username,original_Username)==0 && strcmp(Password,original_Password)==0)
{
    printf("\n\n\n\n\t\t\t----- Login Successful -----");
    Sleep(1000);
    AdminMenu();
}
else
{
    printf("\n\n\n\n\t\t\t-----Invalid Credentials...Enter Again-----");
    getch();
    ALogin();
}
}

void AdminMenu()
{
    system("cls");
    // system("color 70");
    int select;
    // call Title function
    Title();
}

```



```

gotoxy(3,15);
printf("UserName");
gotoxy(28,15);
printf("FullName");
gotoxy(53,15);
printf("Gender");
gotoxy(63,15);
printf("Age");
gotoxy(68,15);
printf("Address");
gotoxy(98,15);
printf("Contact");
gotoxy(113,15);
printf("Email");
gotoxy(153,15);
printf("Problem\n");
printf("=====\\n\\n");
=====
=====\\n\\n");

row=18;
while(fscanf(fp,"%s %s %s %c %d %s %s %s %s\n", p.Username, p.Password, p.
Name, &p.Gender, &p.Age, p.Address, p.Contact, p.Email, p.Problem)!=EOF)
{
    gotoxy(3,row);
    printf("%s",p.Username);
    gotoxy(28,row);
    printf("%s",p.Name);
    gotoxy(53,row);
    printf("%c",p.Gender);
    gotoxy(63,row);
    printf("%d",p.Age);
    gotoxy(68,row);
    printf("%s",p.Address);
    gotoxy(98,row);
    printf("%s",p.Contact);
    gotoxy(113,row);
    printf("%s",p.Email);
}

```


B.


```

while(fscanf(f,"%s %s %s %c %d %s %s %s %s %s\n", p.Username, p.Password, p.N
ame, &p.Gender, &p.Age, p.Address, p.Contact, p.Email, p.Problem) != EOF)
{
    if(strcmp(username,p.Username) == 0)
    {
        temp = 1;
        fp1 = fopen("AvailableAppointments.txt","r");
        fp2 = fopen("NewAppointments.txt","w+");
        while(fscanf(fp1,"%s %s\n",doctorName,appLeft) != EOF)
        {
            if(strcmp(doctorName,doct) == 0)
            {
                flag = 1;
                left = atoi(appLeft);
                if(left>0)
                {
                    left--;
                    itoa(left,left1,10);
                    printf("\n\n\n\t\t\t\t\t\t\t-----\n");
                    APPPOINTMENT BOOKED-----");
                    Sleep(1000);
                    fprintf(fp2,"%s %s\n",doctorName,left1);
                    for(int i = 0;i<strlen(ctime(&t))-1;i++)
                    {
                        if(ctime(&t)[i] == ' ')
                        {
                            time[i] = '-';
                        }
                        else
                        {
                            time[i] = ctime(&t)[i];
                        }
                    }
                    fprintf(fp,"%s %s %s\n",username,doctorName,time);
                }
            }
        }
    }
}

```

```

        printf("\n\n\t\t\t\t-----");
NO APPOINTMENTS LEFT FOR THIS DOCTOR-----");
        fprintf(fp2,"%s %s\n",doctorName,appLeft);
    }
}
else
{
    fprintf(fp2,"%s %s\n",doctorName,appLeft);
}
}

if(flag != 1)
{
    printf("\n\n\t\t\t\t-----");
INVALID DOCTOR NAME-----");
}

fclose(fp1);
fclose(fp2);
remove("AvailableAppointments.txt");
rename("NewAppointments.txt","AvailableAppointments.txt");
}

}

if(temp != 1)
{
    printf("\n\n\t\t\t\t-----");
PATIENT WITH THE SPECIFIED USERNAME IS NOT FOUND-----");
}

}

fclose(f);
fclose(fp);
getch();
AdminMenu();
}

void DeleteUser()
{

```



```
    fprintf(fp1,"%s %s %s %c %d %s %s %s %c\n", iter1.Username, iter1.Password
, iter1.Name, iter1.Gender,iter1.Age, iter1.Address, iter1.Contact,iter1.spec, iter1.appoi
ntments);
}
}
if(k==0)
{
    printf("\n\n\t\t-----Username doesn't exist-----
-----");
}
fclose(fp);
fclose(fp1);
remove("Doctors.txt");
rename("temp.txt","Doctors.txt");
getch();
AdminMenu();
}
```

```

while(ch != 13)      //to print '*' when entering password
{
    fflush(stdin);
    ch = getch();
    if(ch != 13 && ch != 8) //printing '*' when a character other than enter and ba
ckspace is entered
    {
        printf("*");
        password[l] = ch;
        l++;
    }
    else if(ch == 8)      //when user enters backspace
    {
        if(l>0)
        {
            l--;
            password[l] = '\0';
            printf("\b \b");
        }
    }
    if(ch == 13)
    {
        break;
    }
}
strcpy(Password,password);

if (strcmp(Username,original_Username)==0 && strcmp(Password,original_Passwor
d)==0)
{
    printf("\n\n\n\n\t\t\t-----Logging in-----");
    getch();
    ReceptionistMenu();//call Menu function
}
else
{

```



```

case 5:
    RegisterLogin()// MainMenu function is called
    break;
default:
    printf("\n\n\n\n\t\t\t-----Invalid Entry...Choose Again-----
-----");
    getch()//holds screen
    ReceptionistMenu();
    break;
}//end of switch
}

void PatientsList()
{
    int row;
    system("cls");
    Title();
    FILE *fp;
    fp=fopen("Patients.txt","r");
    printf("\n\n\t\t\t===== LIST OF PA
TIENTS =====");
    gotoxy(1,15);
    printf("UserName");
    gotoxy(25,15);
    printf("FullName");
    gotoxy(45,15);
    printf("Gender");
    gotoxy(55,15);
    printf("Age");
    gotoxy(60,15);
    printf("Address");
    gotoxy(85,15);
    printf("Contact");
    gotoxy(100,15);
    printf("Email");
    gotoxy(125,15);
    printf("Problem\n");
}

```

```

printf("=====-----\n\n=====\n");
row=17;
while(fscanf(fp,"%s %s %s %c %d %s %s %s %s\n", p.Username, p.Password, p.
Name, &p.Gender, &p.Age, p.Address, p.Contact, p.Email, p.Problem)!=EOF)
{
    gotoxy(1,row);
    printf("%s",p.Username);
    gotoxy(25,row);
    printf("%s",p.Name);
    gotoxy(45,row);
    printf("%c",p.Gender);
    gotoxy(55,row);
    printf("%d",p.Age);
    gotoxy(60,row);
    printf("%s",p.Address);
    gotoxy(85,row);
    printf("%s",p.Contact);
    gotoxy(100,row);
    printf("%s",p.Email);
    gotoxy(125,row);
    printf("%s",p.Problem);
    row++;
}
fclose(fp);
getch();
ReceptionistMenu();
}

void ViewPatientDetails()
{
    char username[30];
    int flag = 0;
    system("cls");
    Title();// call Title function
}

```



```
void CheckPatientAppointments()
{
    char username[30];
    FILE *fp;
    int counter = 0;
    char PatName[30],DocName[30],time[50];
    printf("\n Enter Patient's UserName to Check : ");
    gets(username);
    fp = fopen("PatientAppointments.txt","r");
    while(fscanf(fp,"%s %s %s\n",PatName,DocName,time) != EOF )
    {
        if(strcmp(PatName,username) == 0)
        {
            counter++;
            printf("\n\n\tPATIENT HAVE APPOINTMENT WITH -
% s\n\tAPPOINTMENT BOOKED TIME : % s \n",DocName,time);
        }
    }
    if(counter == 0)
    {
        printf("\n\tPATIENT DON'T HAVE ANY APPOINTMENTS !");
    }
    fclose(fp);
    getch();
    ReceptionistMenu();
}
```



```

        }
    }
    if(ch == 13)
    {
        break;
    }
}

strcpy(pwd,password);
fp = fopen("Doctors.txt","r");
while(fscanf(fp,"%s %s %s %c %d %s %s %s %c\n", iter1.Username, iter1.Password
, iter1.Name, &iter1.Gender, &iter1.Age, iter1.Address, iter1.Contact, iter1.spec, &iter1
.appointments)!=EOF)
{
    if(strcmp(iter1.Username, username)==0)
    {
        flag = 1;
        if (strcmp(iter1.Password, pwd)==0)
        {
            printf("\n\n\t\t\t-----Logging in-----");
            Sleep(1000);
            fclose(fp);
            DoctorMenu(iter1.Username);
        }
        else
        {
            fclose(fp);
            printf("\n\n\t\t-----");
            Wrong Password...Please Give Correct Details-----");
            Sleep(1000);
            DLogin();
        }
    }
}
if(flag == 0)
{
    printf("\n\n\t\t-----");
    Username not found...Please Give Correct Details-----");
}

```



```

else
{
    fprintf(fp1,"%s %s %s\n",PatID,DocName,time);
}
}

if(flag == 1)
{
    printf("\n\n\t\t-----");
    YOUR APPOINTMENT FOR THIS PATIENT IS DELETED-----");
}
else
{
    printf("\n\n\t\t-----");
    PATIENT NOT FOUND IN YOUR APPOINTMENTS LIST-----");
}
fclose(fp);
fclose(fp1);
remove("PatientAppointments.txt");
rename("temp.txt","PatientAppointments.txt");
getch();
DoctorMenu(doctorname);
}

```

3.2.2. GITHUB/FOLDER STRUCTURE

In the Github repository “Hospital Management System”, We have a folder and a file along with a README. We have uploaded all the text files into a folder named Data Files. Source code “hms.c” is uploaded onto the repository. README has a brief description of what the project is all about.

<https://github.com/shripallavi/HospitalManagementSystem>

 shripallavi	Add files via upload	1cd656d now	 2 commits
 Code	Add files via upload	now	
 Files	Add files via upload	now	
 README.md	Initial commit	14 hours ago	
 shripallavi	Add files via upload	1cd656d 16 seconds ago	 History
<hr/>			
 hms.c	Add files via upload	16 seconds ago	
 AvailableAppointments.txt	Add files via upload	27 seconds ago	
 Doctors.txt	Add files via upload	27 seconds ago	
 DoctorsList.txt	Add files via upload	27 seconds ago	
 PatientAppointments.txt	Add files via upload	27 seconds ago	
 Patients.txt	Add files via upload	27 seconds ago	

I
a

3.3. TESTING

Testing is a method to check whether the actual product matches the expected requirements and to ensure that the product is defect-free. This process involves execution of various parts of the product either using manual or automated tools. The purpose is to identify errors, gaps or missing requirements in contrast to the actual requirements.

3.3.1. TESTPLAN

We approached testing our console application by analysing each module separately. First, we coded the requirements and then manually tested each feature present in the module to cover any gaps that might occur.

3.3.2 PATIENT TEST CASES

3.3.2.1 REGISTER

Test Case ID: TC01	Use Case ID	
Test Case Title: Patient– Register	UC01	
Test Case Description: When new patient enters into the application he is directed to admin.		
Test Steps	Expected Result	Actual Result
1. System prompts the user to select if he is a new patient or an existing one. 2. Patient selects “new patient” option.	System should display “Please consult admin for creating an account”.	System displays “Please consult admin for creating an account”.

3.3.2.2 LOGIN

Test Case ID: TC02	Use Case ID	
Test Case Title: Patient– Login	UC02	
Test Case Description: New patient attempts to login into HMS with invalid credentials.		
Test Steps	Expected Result	Actual Result
<p>1. System displays prompt for user to enter username and password.</p> <p>2. Patient enters username and password where username isn't registered.</p>	System should display a message "Username not found".	System displays a message "Username not found".

Test Case ID: TC03	Use Case ID	
Test Case Title: Patient– Login	UC02	
Test Case Description: User attempts to login into HMS without registering.		
Test Steps	Expected Result	Actual Result
<p>1. System displays prompt for user to enter username and password.</p> <p>2. User enters a username and password where password is incorrect</p>	System should display that the password entered is wrong.	System displays "Wrong Password...Please Enter again".

3.3.2.3 VIEW DOCTOR'S LIST

Test Case ID: TC04		Use Case ID
Test Case Title: Patient- View Doctor's List		UC03
Test Case Description: User attempts to view doctors in HMS		
Test Steps	Expected Result	Actual Result
1.System displays prompt for user to choose any option in the menu 2.User chooses view doctor's list option	System should display the list of doctors	System displays doctor's names along with their specializations.

3.3.2.4 MAKE APPOINTMENT

Test Case ID: TC05		Use Case ID
Test Case Title: Patient– Make Appointment		UC04
Test Case Description: Patient can make an appointment		
Test Steps:	Expected Result:	Actual Result:
1. Patient should enter the name of a doctor (wrong name) 2. He/She must type YES for confirming	System should display that the name is invalid	System displays “Invalid Doctor Name”.

Test Case ID: TC06		Use Case ID
Test Case Title: Patient– Make Appointment		UC04
Test Case Description: Patient can make an appointment		
Test Steps:	Expected Result:	Actual Result:
1. Patient should enter the	System should display	System displays

name of a doctor 2. He/She must type NO for confirming	that the Verification is failed.	“Verification failed”.
---	----------------------------------	------------------------

3.3.2.5 CHECK APPOINTMENT

Test Case ID: TC07	Use Case ID	
Test Case Title: Patient– Check Appointment	UC05	
Test Case Description: Patient can check his/her appointments		
Test Steps:	Expected Result:	Actual Result:
1. Patient should choose the check appointment option (when there are appointments)	System should display the appointments of the patient.	System displays the doctor name and the date and time of when the appointment is booked.

Test Case ID: TC08	Use Case ID	
Test Case Title: Patient– Make Appointment	UC05	
Test Case Description: Patient can make an appointment		
Test Steps:	Expected Result:	Actual Result:
1. Patient should choose the check appointment option (when there are no appointments)	System should display that there are no appointments.	System displays “No Appointments”.

3.3.2.6 UNREGISTER

Test Case ID: TC09	Use Case ID	
Test Case Title: Patient– Unregister		
Test Case Description: Patient can unregister from the hospital.	UC06	
Test Steps:	Expected Result:	Actual Result:
1. Patient should choose the unregister option	System should delete the records of the patient.	System displays “Unregistered” and deletes the records of the patient.

3.3.3 ADMIN TESTCASES

3.3.3.1 LOGIN

Test Case ID: TC10	Use Case ID	
Test Case Title: Admin– Login		
Test Case Description: Admin Logs in to the system with correct credentials	UC07	
Test Case Title: Admin– Login	Test Case Title: Admin– Login	Test Case Title: Admin– Login
Test Case Description: Admin Logs in to the system with correct credentials	Test Case Description: Admin Logs in to the system with correct credentials	Test Case Description: Admin Logs in to the system with correct credentials

Test Case ID: TC11	Use Case ID
Test Case Title: Admin– Login	
Test Case Description: Admin Logs in to the system with correct credentials	UC07

wrong corrections		
Test Steps:	Expected Result:	Actual Result:
1. System asks for username and password 2. Admin enters wrong credentials.	System should display that the credentials entered are wrong.	System displays “Invalid Credentials”.

3.3.3.2 ADD A PATIENT

Test Case ID: TC12	Use Case ID	
Test Case Title:	Admin– Add a Patient	
Test Case Description:	Admin Logs in to the system and adds details of a patient (Username)	
Test Steps:	Expected Result:	Actual Result:
1. System asks to enter the username of the patient to add. 2. Admin enters username of existing patient.	System should display that the username already exists	System displays a message “Username already exists”

Test Case ID: TC13	Use Case ID	
Test Case Title:	Admin– Add a Patient	
Test Case Description:	Admin Logs in to the system and adds details of a patient (Contact)	
Test Steps:	Expected Result:	Actual Result:
1. System asks to enter the contact of the patient to add. 2. Admin enters contact with digits less than 10.	System should display that the contact is invalid	System displays a message that the contact is invalid and gives asks the admin to enter 10 digits.

3.3.3.3 PATIENT'S LIST

Test Case ID: TC14	Use Case ID	
Test Case Title: Admin- Patient's List		
Test Case Description: Admin attempts to view patients in HMS	Test Steps:	Expected Result:
1.System displays prompt for user to choose any option in the menu 2.Admin chooses patient's list option	System should display the list of patients	System displays doctor's names along with their details .

3.3.3.4 ADD A DOCTOR

Test Case ID: TC15	Use Case ID	
Test Case Title: Admin– Add a Doctor		
Test Case Description: Admin Logs in to the system and adds details of a doctor (Username)	Test Steps:	Expected Result:
1. System asks to enter the username of the doctor 2. Admin enters existing doctor's username.	System should display that the username already exists	System displays a message “Username already exists”

3.3.3.5 EDIT PATIENT RECORD

Test Case ID: TC16	Use Case ID
---------------------------	--------------------

Test Case Title: Admin– Edit Patient Record		UC11
Test Case Description: Admin Logs in to the system and edits details of a patient (Username)		
Test Steps:	Expected Result:	Actual Result:
1. System asks to enter the username of the doctor 2. Admin enters existing doctor's username.	System should enable admin to edit the details of the patient	System asks for new details of the patient

Test Case ID: TC17	Use Case ID	
Test Case Title: Admin– Edit Patient Record	UC11	
Test Case Description: Admin Logs in to the system and edits details of a patient		
Test Steps:	Expected Result:	Actual Result:
1. System asks to enter the username of the patient to add. 2. Admin enters new username which is not registered.	System should display that the username is not found	System displays a message that there are no records of patients with that username.

3.3.3.6 CHECK DOCTOR

Test Case ID: TC18	Use Case ID	
Test Case Title: Patient- View Doctor's List	UC12	
Test Case Description: Admin attempts to view doctors in HMS		
Test Steps:	Expected Result:	Actual Result:
1. System displays prompt for user to choose any option in the menu 2. Admin chooses check doctor's list option	System should display the list of doctors	System displays doctor's names along with their specializations.

3.3.3.7 GIVE APPOINTMENT

Test Case ID: TC19		Use Case ID
Test Case Title: Admin - Give Appointment		
Test Case Description: Admin can give an appointment		UC13
Test Steps:	Expected Result:	Actual Result:
1. Admin should enter the name of a doctor (wrong name) 2. He/She must type YES for confirming	System should display that the name is invalid	System displays “Invalid Doctor Name”.

Test Case ID: TC20		Use Case ID
Test Case Title: Admin - Give Appointment		
Test Case Description: Admin can give an appointment		UC13
Test Steps:	Expected Result:	Actual Result:
1. Admin should enter the name of a doctor 2. He/She must type NO for confirming	System should display that the Verification is failed.	System displays “Verification failed”.

3.3.3.8 DELETE USER

Test Case ID: TC21		Use Case ID
Test Case Title: Admin - Delete an user		
Test Case Description: Admin can delete records of patients		UC14
Test Steps:	Expected Result:	Actual Result:
1. System displays prompt for user to username of patient 2. Admin enters correct username	System should delete the records of that patient.	System displays a message that the details are deleted and removes them from

		the records
--	--	-------------

Test Case ID: TC22		Use Case ID
Test Case Title: Admin - Delete an user		UC14
Test Case Description: Admin can delete records of patients		
Test Steps:	Expected Result:	Actual Result:
1. System displays prompt for user to username of patient 2. Admin enters incorrect username	System should display that there is no patient with that username	System displays “No patient records found”.

3.3.3.9 DELETE DOCTOR

Test Case ID: TC23		Use Case ID
Test Case Title: Admin - Delete a doctor		UC15
Test Case Description: Admin can delete records of doctors		
Test Steps:	Expected Result:	Actual Result:
1. System displays prompt for user to username of doctor 2. Admin enters correct username	System should delete the records of that doctor.	System displays a message that the details are deleted and removes them from the records

Test Case ID: TC24		Use Case ID
Test Case Title: Admin - Delete a doctor		UC15
Test Case Description: Admin can delete records of doctors		
Test Steps:	Expected Result:	Actual Result:
1. System displays prompt for user to username of patient 2. Admin enters incorrect username	System should display that there is no doctor with that username	System displays “No Doctor records found”.

3.3.4 RECEPTIONIST TESTCASES

3.3.4.1 PATIENT'S LIST

Test Case ID: TC25		Use Case ID
Test Case Title: Receptionist- Patient's List		UC16
Test Case Description: Receptionist attempts to view patients in HMS		
Test Steps:	Expected Result:	Actual Result:
1.System displays prompt for user to choose any option in the menu 2.Admin chooses patient's list option	System should display the list of all patients	System displays all patients' names along with their details.

3.3.4.2 VIEW PATIENT'S DETAILS

Test Case ID: TC26		Use Case ID
Test Case Title: Receptionist–View Patient's Records		UC17
Test Case Description: Receptionist Logs in to the system and views details of a patient		
Test Steps:	Expected Result:	Actual Result:
1. System asks to enter the username of the patient to add. 2. Receptionist enters registered username	System should display details of that particular patient.	System displays details of that particular patient.

Test Case ID: TC27		Use Case ID
Test Case Title: Receptionist–View Patient's Records		UC17
Test Case Description: Receptionist Logs in to the system		

and views details of a patient		
Test Steps:	Expected Result:	Actual Result:
1. System asks to enter the username of the patient to add. 2. Receptionist enters invalid username	System should display that the username is not found	System displays a message that there are no record of a patient with that username.

3.3.4.3 CHECK PATIENT APPOINTMENT

Test Case ID: TC28	Use Case ID
Test Case Title: Receptionist – Check appointments of patient	UC18
Test Case Description: Receptionist can check his/her appointments	
Test Steps:	Expected Result:
1.Receptionist should choose the check patient appointment option 2. Enters a valid username	System should display the appointments of that particular patient.
	System displays the doctor name and the date and time of when the appointment is booked.

Test Case ID: TC29	Use Case ID
Test Case Title: Receptionist – Check appointments of patient	UC18
Test Case Description: Receptionist can check his/her appointments	
Test Steps:	Expected Result:
1.Receptionist should choose the check patient appointment option	System should display an error message saying patient doesn't have any
	System displays an error message saying patient doesn't have

2. Enters a invalid username	appointments	any appointments
------------------------------	--------------	------------------

3.3.4.4 CHECK DOCTORS APPOINTMENTS LEFT

Test Case ID: TC29	Use Case ID	
Test Case Title: Receptionist – Check doctors' appointments left	UC19	
Test Case Description: Receptionist can check all the remaining slots for all the doctors		
Test Steps	Expected Result	Actual Result
1.Receptionist selects check doctors' appointments left option	System should display all the doctors along with the slots available	System displays list of all the doctors and the slots available

3.3.5 DOCTOR TEST CASES

3.3.5.1 MY PATIENTS

Test Case ID: TC30	Use Case ID	
Test Case Title: Doctor – Check assigned patients	UC20	
Test Case Description: Doctors can view all the patients who booked appointment with them.		
Test Steps	Expected Result	Actual Result
1.Doctor selects check my patients option	System should display all the patients along with their details assigned to the doctor if there are any.	System displays all the patients along with their details assigned to that particular doctor

Test Case ID: TC30	Use Case ID	
Test Case Title: Doctor – Check assigned patients	UC20	
Test Case Description: Doctors can view all the patients who booked appointment with them.		
Test Steps	Expected Result	Actual Result
1.Doctor selects check my patients option (when there are no assigned patients)	System should display “No Consultations” message	System displays “No Consultations” message

3.3.5.2 CANCEL APPOINTMENTS

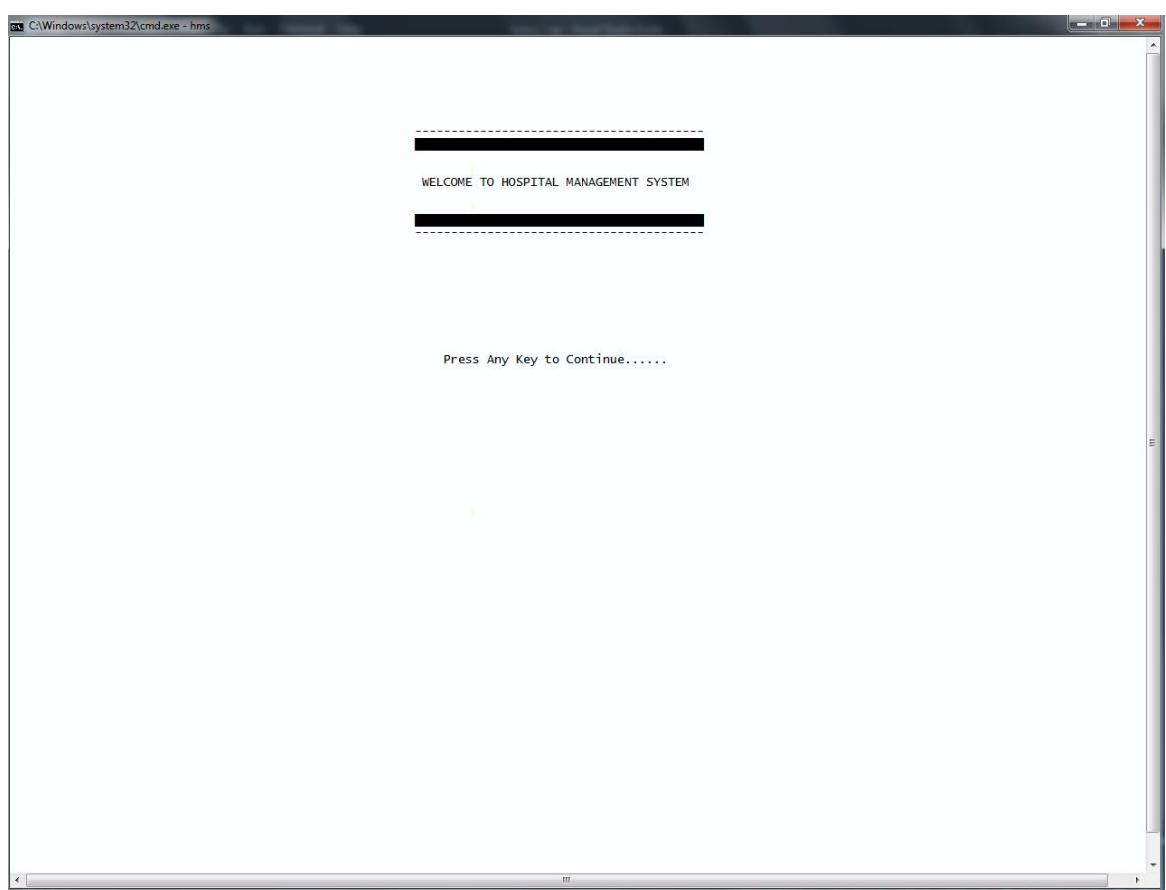
Test Case ID: TC31	Use Case ID	
Test Case Title: Doctor – Cancel appointments	UC21	
Test Case Description: Doctor can cancel appointments		
Test Steps	Expected Result	Actual Result
1.Doctor selects cancel appointment option 2. Enters valid patient username	System should successfully cancel appointment and show a confirmation message	System displays a confirmation message saying “Appointment is cancelled”.

Test Case ID: TC232	Use Case ID	
Test Case Title: Doctor – Cancel appointments	UC21	
Test Case Description: Doctor can cancel appointments		
Test Steps	Expected Result	Actual Result
1.Doctor selects cancel appointment option 2. Enters invalid patient username	System should display an error message.	System displays a error message saying “Patient not found”.

4. RESULTS

We have successfully developed a platform for users of all age groups to utilize and improve their skill-set. Below are the output screenshots of the test cases mentioned.

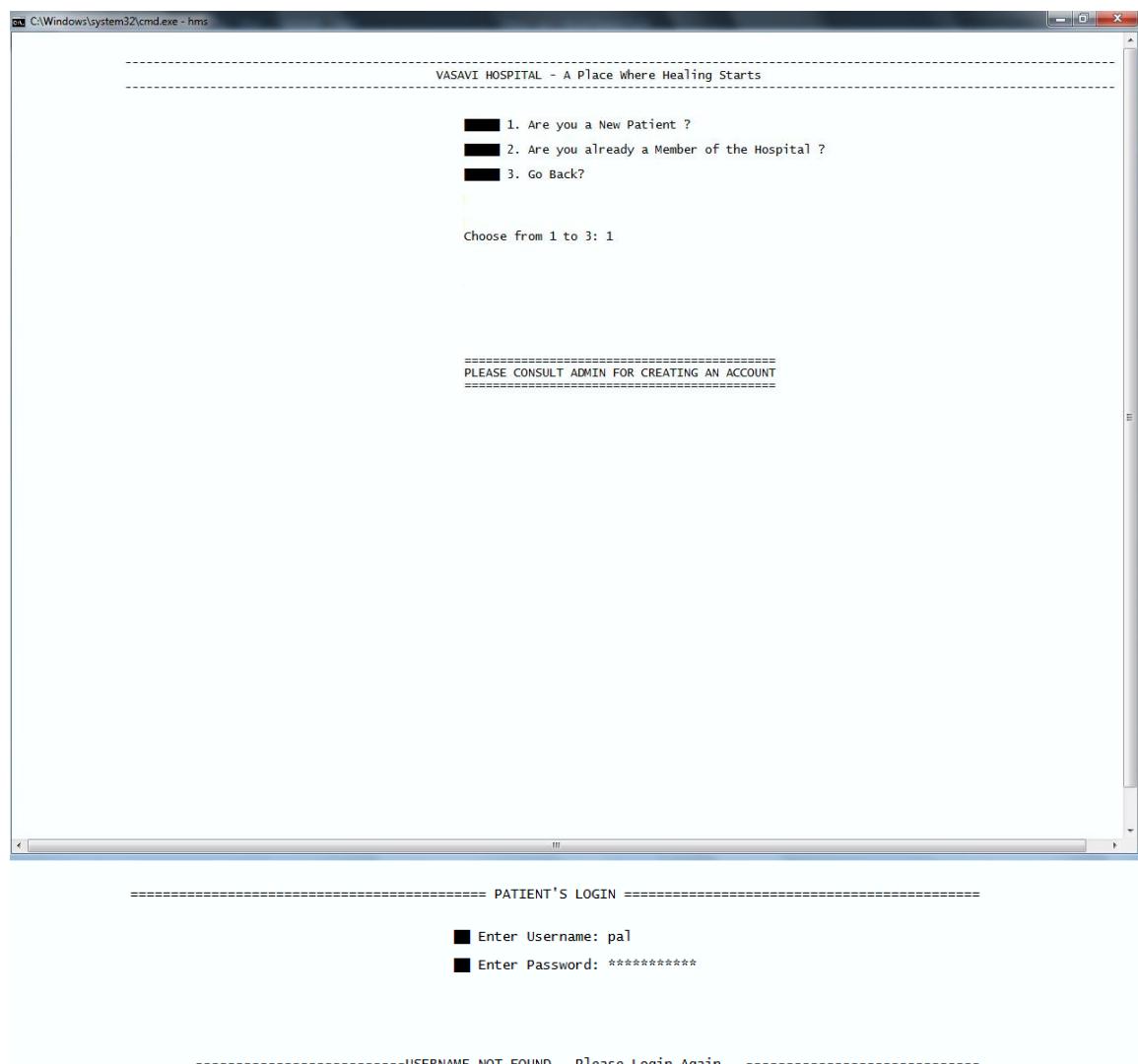
4.1. PATIENT TEST CASE RESULTS



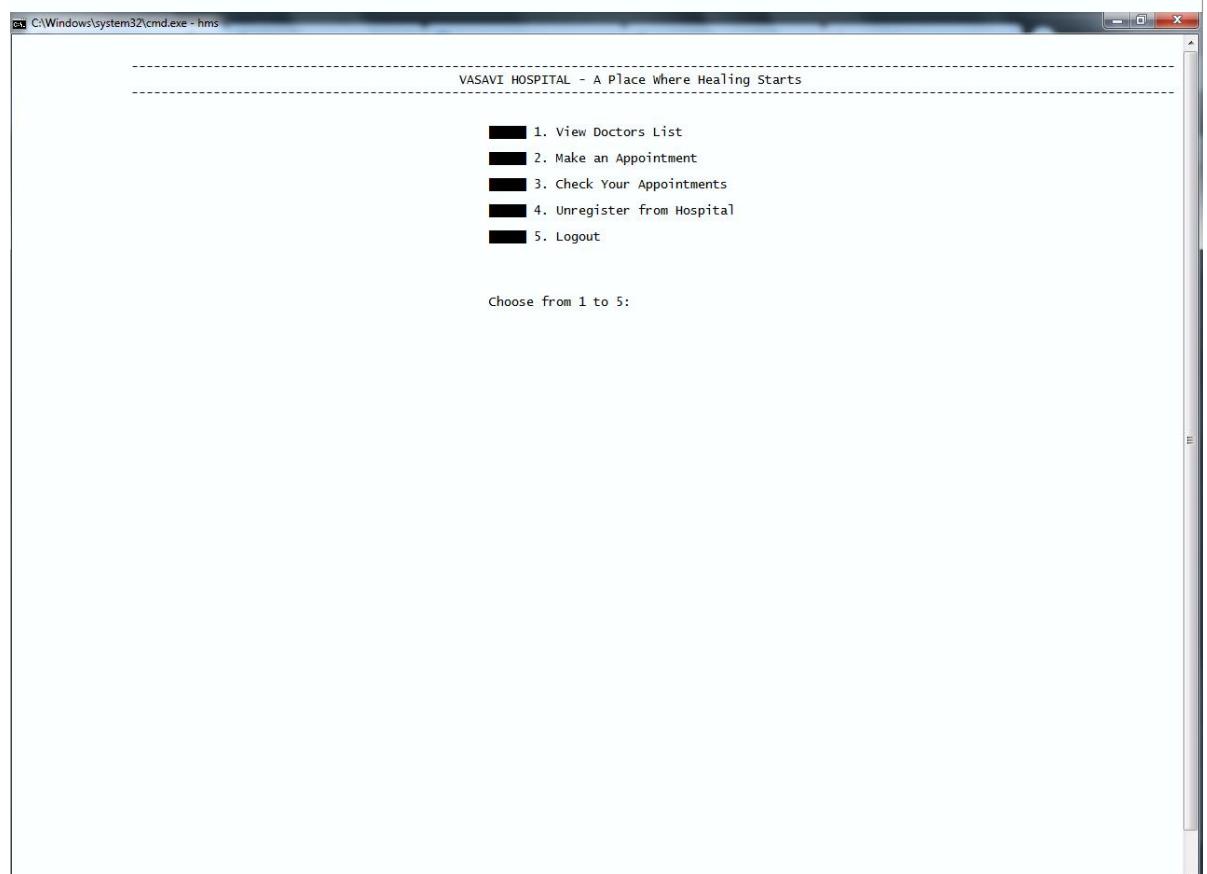
```
C:\Windows\system32\cmd.exe - hms

-----
VASAVI HOSPITAL - A Place Where Healing Starts

-----  
1. Are you a New Patient ?  
2. Are you already a Member of the Hospital ?  
3. Go Back?  
  
Choose from 1 to 3:  
  
-----  
VASAVI HOSPITAL - A Place where Healing Starts  
-----  
1. Are you a Patient ?  
2. Are you an Admin ?  
3. Are you a Receptionist ?  
4. Are you a Doctor ?  
5. Exit  
  
Choose from 1 to 5: _
```



```
===== PATIENT'S LOGIN =====
█ Enter Username: Siddarth
█ Enter Password: *****
-----Logging in-----
```



```
C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts

===== DOCTOR'S LIST =====

DOCTOR NAME          SPECIALIZATION
-----
Dr.Venkat             Physician
Dr.Shraddha           Dermatologist
Dr.Prabhakaran        Cardiologist
Dr.Meenakshi           Gynecologist
Dr.Seshadri            Psychiatrist
Dr.Ravichandran        ENTSpecialist
Dr.Sanketh              Dentist
Dr.Srinivas             Diabetologist
Dr.Sangeetha            Cardiologist
Dr.Rameshwari          Ophthalmologist
Dr.Sravya               Cardiologist
Dr.Naveen                Orthopedist
Dr.Drake                 Neurologist
Dr.Naveen               Physician

Enter any key to continue
```

```
C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts

===== MAKE APPOINTMENT =====

-----DOCTORS AND APPOINTMENTS LEFT-----
Dr.Ramesh      0
Dr.Venkat       0
Dr.Shraddha     0
Dr.Prabhakaran 3
Dr.Meenakshi    5
Dr.Seshadri     0
Dr.Ravichandran 4
Dr.Sanketh      0
Dr.Srinivas     0
Dr.Sangeetha    0
Dr.Rameshwari   0
Dr.Naveen        0
Dr.Drake         0
Dr.Naveen        6

-----■ Enter Doctor Name : Dr.Raju
■ Enter 'YES' to Confirm Appointment : YES
-----INVALID DOCTOR NAME-----
```

```
C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts
===== MAKE APPOINTMENT =====
-----DOCTORS AND APPOINTMENTS LEFT-----
Dr.Ramesh 0
Dr.Venkat 0
Dr.Shraddha 0
Dr.Prabhakaran 3
Dr.Meenakshi 5
Dr.Seshadri 0
Dr.Ravichandran 4
Dr.Sanketh 0
Dr.Srinivas 0
Dr.Sangeetha 0
Dr.Rameshwar 0
Dr.Naveen 0
Dr.Drake 0
Dr.Naveen 6

■ Enter Doctor Name : Dr.Ravichandran
■ Enter 'YES' to Confirm Appointment : YES

-----APPOINTMENT BOOKED-----
```

```
C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts
===== MAKE APPOINTMENT =====
-----DOCTORS AND APPOINTMENTS LEFT-----
Dr.Ramesh 0
Dr.Venkat 0
Dr.Shraddha 0
Dr.Prabhakaran 3
Dr.Meenakshi 5
Dr.Seshadri 0
Dr.Ravichandran 3
Dr.Sanketh 0
Dr.Srinivas 0
Dr.Sangeetha 0
Dr.Rameshwar 0
Dr.Naveen 0
Dr.Drake 0
Dr.Naveen 6

■ Enter Doctor Name : Dr.Prabhakaran
■ Enter 'YES' to Confirm Appointment : NO

-----VERIFICATION FAILED! Enter any key to go back-----
```

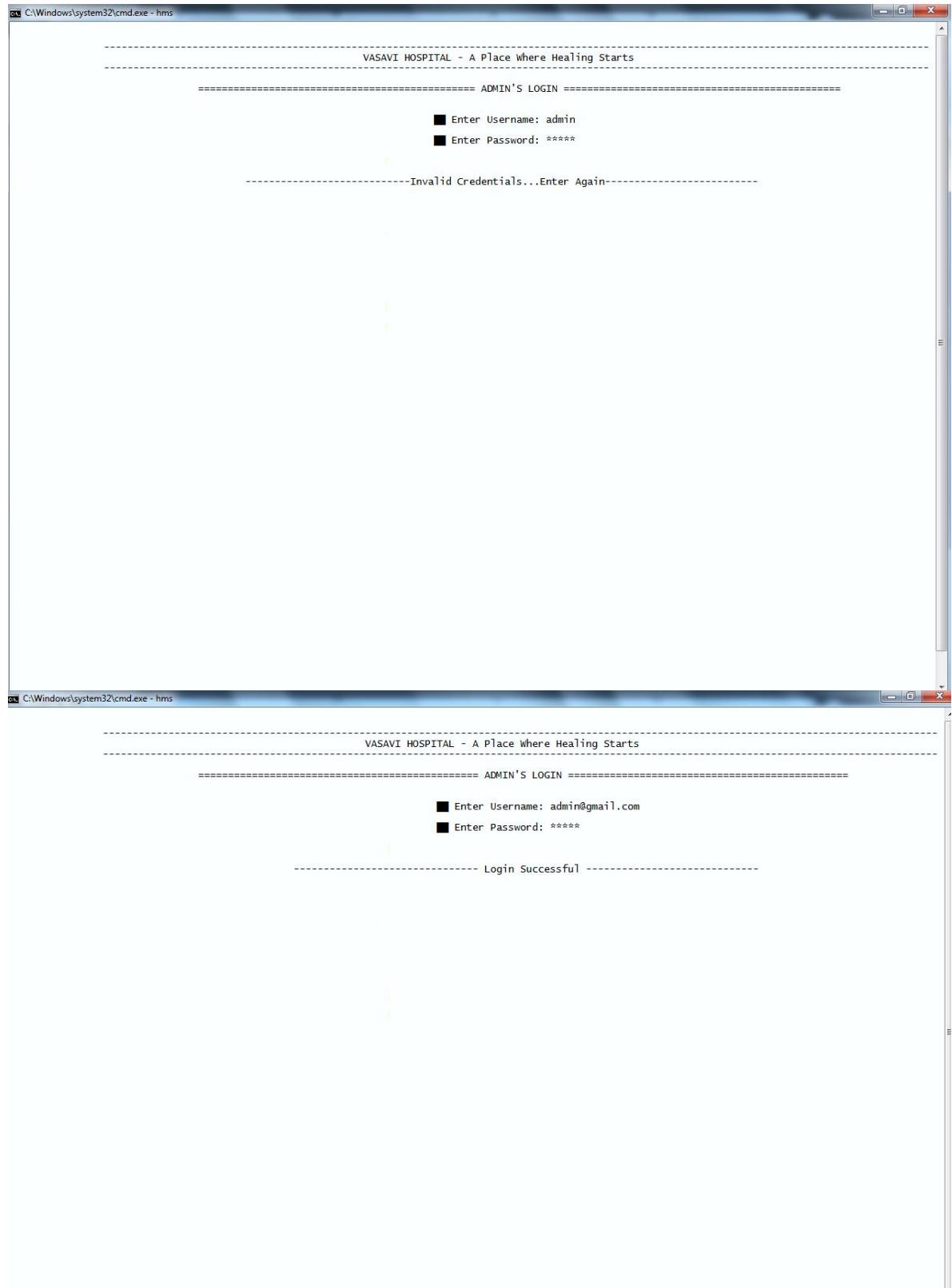
```
C:\Windows\system32\cmd.exe - hms

-----
VASAVI HOSPITAL - A Place Where Healing Starts
=====
===== CHECK APPOINTMENTS =====
■ YOU HAVE AN APPOINTMENT WITH - Dr.Ravichandran
You Booked this Appointment on : Tue-Dec-22-09:11:09-2020
```

```
C:\Windows\system32\cmd.exe - hms

-----
VASAVI HOSPITAL - A Place Where Healing Starts
-----
-----YOUR ACCOUNT IS SUCCESSFULLY DELETED-----
-----
Enter any key to go back to Main Menu...
```

4.2 ADMIN TEST CASE RESULTS



The image contains two side-by-side screenshots of a Windows command-line window titled "C:\Windows\system32\cmd.exe - hms".

Screenshot 1 (Left): Invalid Credentials

VASAVI HOSPITAL - A Place Where Healing Starts
===== ADMIN'S LOGIN =====
■ Enter Username: admin
■ Enter Password: *****
-----Invalid Credentials...Enter Again-----

Screenshot 2 (Right): Login Successful

VASAVI HOSPITAL - A Place Where Healing Starts
===== ADMIN'S LOGIN =====
■ Enter Username: admin@gmail.com
■ Enter Password: *****
----- Login Successful -----

```
C:\Windows\system32\cmd.exe - hms

-----
VASAVI HOSPITAL - A Place Where Healing Starts
-----
[■] 1. Add a Patient
[■] 2. List of Patients
[■] 3. Edit List of Patients
[■] 4. Add Doctor
[■] 5. Check List of Doctors
[■] 6. Give an Appointment
[■] 7. Delete an User
[■] 8. Delete an Doctor
[■] 9. Logout

Choose from 1 to 9:
```

```
C:\Windows\system32\cmd.exe - hms

-----
VASAVI HOSPITAL - A Place Where Healing Starts
-----
===== ADD PATIENT'S RECORD =====
[■] Enter Username: Pallavi
-----USERNAME ALREADY EXISTS-----
[■] Enter Username: Sravya
[■] Enter Password: asa
-----Invalid Password...The max range for Password is 8. Please Enter Again-----
[■] Enter Password: 12345678
[■] Full Name: Srav
-----Invalid FullName...The max range for FullName is 25 and min range is 5-----
[■] Full Name: Sravya
[■] Gender[F/M/O]: k
-----Gender contains Invalid character...Enter either F or M or O-----
[■] Gender[F/M/O]: F
[■] Age: 20
[■] Address: gen
-----Invalid Address...The max range for address is 20 and min range is 8-----
[■] Address: Hyderabad
[■] Contact: 2345
-----Invalid...Contact must contain 10 numbers...Enter again-----
[■] Contact: 9876543218
[■] Email: sravya@gmail.com
[■] Problem: Headache
-----INFORMATION RECORD SUCCESSFULL-----
[■] Patient Username : Sravya
[■] Patient Password : 12345678
[■] Do you want to add more[Y/N]? N
----- THANK YOU -----
```

```

C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts
===== LIST OF PATIENTS =====

UserName      FullName      Gender   Age    Address          Contact      Email           Problem
=====  

Pallavi       PallaviGuggila F        23    Alwal_crossroads_HYD 8974234233  pallavi@gmail.com Stomach
ravikumar     Ravikumar      M        19    Qwertyuiooas      9234359734  ravi@gmail.com Headach
Sravya        Sravya         F        20    Hyderabad        9876543218  sravya@gmail.com Headach

```



```

C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts

1. Add a Patient
2. List of Patients
3. Edit List of Patients
4. Add Doctor
5. Check List of Doctors
6. Give an Appointment
7. Delete an User
8. Delete an Doctor
9. Logout

Choose from 1 to 9: \2

-----Invalid Entry...Choose Again-----

```

```
C:\Windows\system32\cmd.exe - hms
=====
===== NEW RECORD =====
■ Enter New Name : sravyareddy
■ Enter Address : Nampally
■ Enter Phone number : 9873278732
■ Enter Gender : F
■ Enter Email : sra@gmail.com
■ Enter Age : 25
■ Enter Problem : Gas
-----PATIENT DETAILS SUCCESSFULLY UPDATED-----

```



```
C:\Windows\system32\cmd.exe - hms
-----
----- VASAVI HOSPITAL - A Place Where Healing Starts -----
===== LIST OF PATIENTS =====

UserName      FullName      Gender      Age      Address      Contact      Email      Problem
Pallavi      PallaviGuggila   F          23      Alwal_crossroads_HYD    8974234323  pallavi@gmail.com  Stomach
Ravikumar    Ravikumar       M          19      Qwertyuiooooas    9234359734  ravi@gmail.com   Headach
Sravya        Sravyareddy    F          25      Nampally       9873278732  sra@gmail.com   Gas

```

```

C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts

===== ADD DOCTOR'S RECORD =====
■ User Name: Dr.Sravya
USERNAME ALREADY EXISTS !
■ User Name: Dr.Santosh
■ Password : 12345678
■ Full Name of Doctor : Santosh
■ Gender[F/M]: W
Gender contains Invalid character Enter either F or M or O
■ Gender[F/M]: m
■ Age:35
■ Address: 1
Invalid... The max range for address is 20 and min range is 8
■ Address: Kairathabad
■ Contact: 234
Invalid....Contact must contain 10 numbers. Enter again
■ Contact: 9876543218
■ Specialisation: Gastrologist
■ No. of Appointments per day : 5
-----INFORMATION RECORD SUCCESSFULL-----
■ Doctor Username : Dr.Santosh
■ Doctor Password : 12345678
■ Do you want to add more[Y/N]??


-----LIST OF DOCTORS-----
DOCTOR NAME                               SPECIALIZATION
-----+
Dr.Venkat                                Physician
Dr.Shraddha                               Dermatologist
Dr.Prabhakaran                            Cardiologist
Dr.Meenakshi                               Gynecologist
Dr.Seshadri                               Psychiatrist
Dr.Ravichandran                           ENTSpecialist
Dr.P.Srinivas                             Diabetologist
Dr.Srinivas                               Cardiologist
Dr.Sangeetha                               Ophthalmologist
Dr.Rameshwar                               Cardiologist
Dr.Sravya                                  Orthopedist
Dr.Naveen                                   Neurologist
Dr.Drake                                    Physician
Dr.Naveen                                   Gastrologist
Dr.Santosh

Enter any key to continue

```

```

C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place where Healing Starts
===== GIVE APPOINTMENT =====
-----DOCTORS AND APPOINTMENTS LEFT-----
Dr.Ramesh      0
Dr.Venkat      0
Dr.Shraddha    0
Dr.Prabhakaran 3
Dr.Meenakshi   5
Dr.Seshadri    0
Dr.Ravichandran 3
Dr.Sanketh     0
Dr.Srinivas    0
Dr.Sangeetha   0
Dr.Rameshwar   0
Dr.Naveen       0
Dr.Drake       0
Dr.Naveen       6
Dr.Santosh     5

----- Enter username of Patient : pal
----- Enter doctor name for appointment : Dr.Meenakshi
----- ENTER 'YES' TO CONFIRM APPOINTMENT : YES
-----PATIENT WITH THE SPECIFIED USERNAME IS NOT FOUND-----

C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place where Healing Starts
===== GIVE APPOINTMENT =====
-----DOCTORS AND APPOINTMENTS LEFT-----
Dr.Ramesh      0
Dr.Venkat      0
Dr.Shraddha    0
Dr.Prabhakaran 3
Dr.Meenakshi   5
Dr.Seshadri    0
Dr.Ravichandran 3
Dr.Sanketh     0
Dr.Srinivas    0
Dr.Sangeetha   0
Dr.Rameshwar   0
Dr.Naveen       0
Dr.Drake       0
Dr.Naveen       6
Dr.Santosh     5

----- Enter username of Patient : Sravya
----- Enter doctor name for appointment : Dr.asa
----- ENTER 'YES' TO CONFIRM APPOINTMENT : YES
-----INVALID DOCTOR NAME-----

```

```
C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts
===== GIVE APPOINTMENT =====
-----DOCTORS AND APPOINTMENTS LEFT-----
Dr .Ramesh      0
Dr .Venkat      0
Dr .Shradhha    0
Dr .Prabhakaran 3
Dr .Meenakshi   5
Dr .Seshadri    0
Dr .Ravichandran 3
Dr .Sanketh     0
Dr .Srinivas    0
Dr .Sangeetha   0
Dr .Rameshwar   0
Dr .Naveen       0
Dr .Drake       0
Dr .Naveen       6
Dr .Santosh      5

----- Enter username of Patient : Sravya
----- Enter doctor name for appointment : Dr.Shraddha
----- ENTER 'YES' TO CONFIRM APPOINTMENT : YES
-----NO APPOINTMENTS LEFT FOR THIS DOCTOR-----
```

```
C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts
===== GIVE APPOINTMENT =====
-----DOCTORS AND APPOINTMENTS LEFT-----
Dr .Ramesh      0
Dr .Venkat      0
Dr .Shradhha    0
Dr .Prabhakaran 3
Dr .Meenakshi   5
Dr .Seshadri    0
Dr .Ravichandran 3
Dr .Sanketh     0
Dr .Srinivas    0
Dr .Sangeetha   0
Dr .Rameshwar   0
Dr .Drake       0
Dr .Naveen       5
Dr .Santosh      5

----- Enter username of Patient : Sravya
----- Enter doctor name for appointment : Dr.Naveen
----- ENTER 'YES' TO CONFIRM APPOINTMENT : YES
-----APPOINTMENT BOOKED-----
```

```
C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
=====
===== GIVE APPOINTMENT =====
-----DOCTORS AND APPOINTMENTS LEFT-----
Dr.Ramesh      0
Dr.Venkat      0
Dr.Shraddha    0
Dr.Prabhakaran 3
Dr.Meenakshi   5
Dr.Seshadri    0
Dr.Ravichandran 3
Dr.Sanketh     0
Dr.Srinivas    0
Dr.Sangeetha   0
Dr.Rameshwar   0
Dr.Drake       0
Dr.Naveen       4
Dr.Santosh     5
-----
■ Enter username of Patient :
```



```
C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
=====
===== DELETE PATIENT RECORD =====
■ ENTER USERNAME OF THE PATIENT : sas
-----
-Username doesn't exist-----
```

```

C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts
===== DELETE PATIENT RECORD =====
■ ENTER USERNAME OF THE PATIENT : Pallavi
-----YOUR ACCOUNT IS SUCCESSFULLY DELETED-----
-----ENTER ANY KEY TO GO BACK TO MAINMENU-----

```



```

C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts
===== LIST OF PATIENTS =====

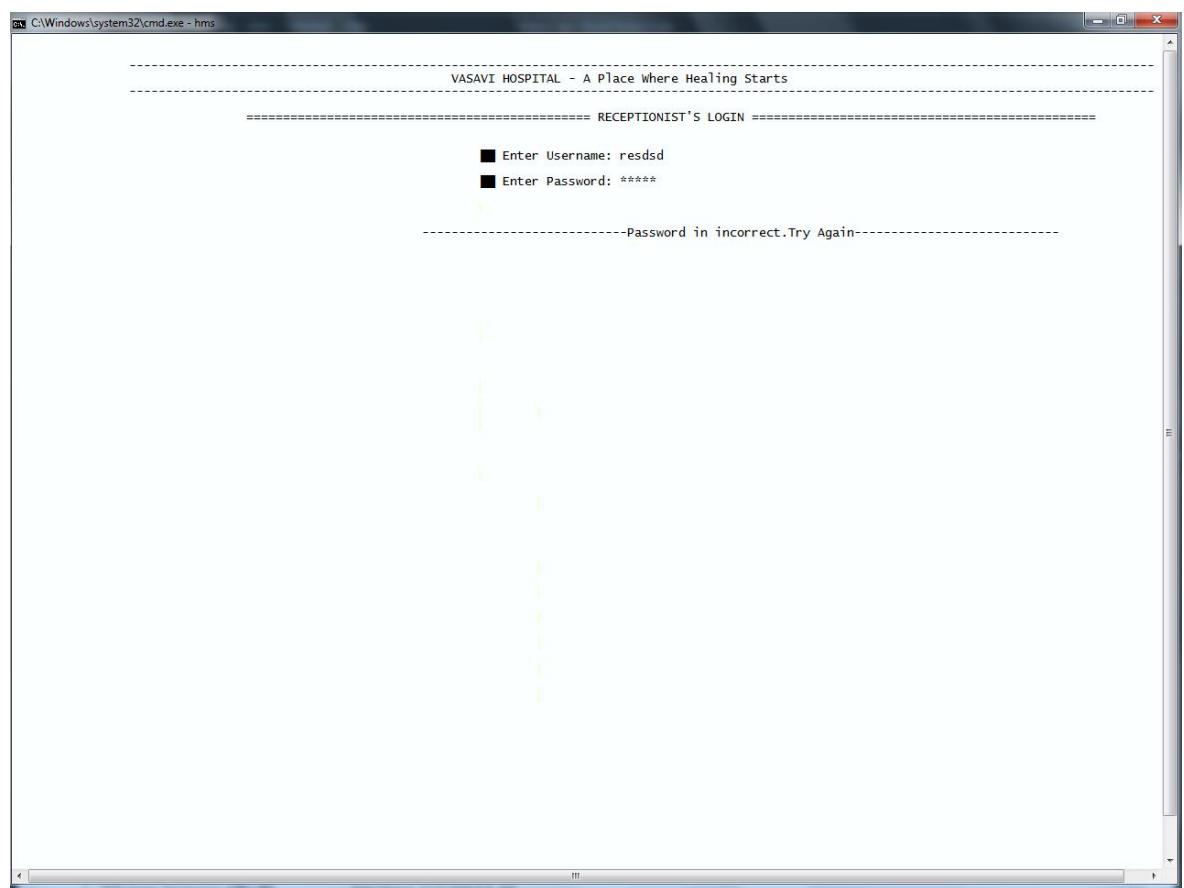
UserName      FullName      Gender   Age    Address          Contact        Email           Problem
=====  

ravikumar     Ravikumar     M        19    Qwertyuiooooas  9234359734    ravi@gmail.com  Headach  

Sravya        Sravyareddy   F        25    Nampally       9873278732    sra@gmail.com  Gas

```


4.3 RECEPTIONIST TEST CASE RESULTS



```
C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
=====
===== RECEPTIONIST'S LOGIN =====
[ ] Enter Username: receptionist@gmail.com
[ ] Enter Password: *****
-----
-----Logging in-----
C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
-----
[ ] 1. View List of Patients
[ ] 2. View Details of a Patient
[ ] 3. Check Appointments of a patient
[ ] 4. Check Available Appointments of Doctors
[ ] 5. Logout
[ ] Choose from 1 to 3:
```

```
C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
=====
===== LIST OF PATIENTS =====
-----
UserName      FullName      Gender   Age    Address          Contact      Email           Problem
ravikumar     Ravikumar     M        19    Qvertyuioooas  9234359734  ravi@gmail.com  Headache
Sravya        Sravyareddy   F        25    Nampally       9873278732  sra@gmail.com  Gas
-----
```

```
C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
=====
===== DETAILS OF A PATIENTS =====
-----
■■■ Enter Patient's UserName to be viewed : aps
-----
-----PATIENT RECORDS NOT FOUND-----
```

```
C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
=====
===== DETAILS OF A PATIENTS =====
[ ] Enter Patient's UserName to be viewed : Sravya
USERNAME      : Sravya
PATIENT NAME  : Sravyareddy
GENDER        : F
AGE           : 25
ADDRESS        : Nampally
CONTACT       : 9873278732
EMAIL          : sra@gmail.com
PROBLEM        : Gas
=====

C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
-----
[ ] 1. View List of Patients
[ ] 2. View Details of a Patient
[ ] 3. Check Appointments of a patient
[ ] 4. Check Available Appointments of Doctors
[ ] 5. Logout
[ ] Choose from 1 to 3: 3
Enter Patient's UserName to Check : rav
PATIENT DON'T HAVE ANY APPOINTMENTS !
```

```
C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts

1. View List of Patients
2. View Details of a Patient
3. Check Appointments of a patient
4. Check Available Appointments of Doctors
5. Logout
Choose from 1 to 3: 3

Enter Patient's UserName to Check : ravikumar

PATIENT HAVE APPOINTMENT WITH - Dr.Venkat
APPOINTMENT BOOKED TIME : Mon-Dec-21-23:11:25-2020

PATIENT HAVE APPOINTMENT WITH - Dr.Naveen
APPOINTMENT BOOKED TIME : Tue-Dec-22-00:38:22-2020

PATIENT HAVE APPOINTMENT WITH - Dr.Naveen
APPOINTMENT BOOKED TIME : Tue-Dec-22-00:43:59-2020

PATIENT HAVE APPOINTMENT WITH - Dr.Seshadri
APPOINTMENT BOOKED TIME : Tue-Dec-22-00:44:42-2020

PATIENT HAVE APPOINTMENT WITH - Dr.Srinivas
APPOINTMENT BOOKED TIME : Tue-Dec-22-00:46:26-2020

PATIENT HAVE APPOINTMENT WITH - Dr.Seshadri
APPOINTMENT BOOKED TIME : Tue-Dec-22-01:02:24-2020

PATIENT HAVE APPOINTMENT WITH - Dr.Sangeetha
APPOINTMENT BOOKED TIME : Tue-Dec-22-01:15:11-2020

PATIENT HAVE APPOINTMENT WITH - Dr.Rameshwar
APPOINTMENT BOOKED TIME : Tue-Dec-22-01:16:24-2020
```

```
C:\Windows\system32\cmd.exe - hms

VASAVI HOSPITAL - A Place Where Healing Starts

1. View List of Patients
2. View Details of a Patient
3. Check Appointments of a patient
4. Check Available Appointments of Doctors
5. Logout
Choose from 1 to 3: 4

-----DOCTORS AND APPOINTMENTS LEFT-----
Dr.Ramesh          0
Dr.Venkat          0
Dr.Shradha         0
Dr.Prabhakaran    3
Dr.Meenakshi       5
Dr.Seshadri        0
Dr.Ravichandran   3
Dr.Rakesh           0
Dr.Srinivas         0
Dr.Sangeetha        0
Dr.Rameshwar        0
Dr.Drake            0
Dr.Naveen           4
Dr.Santosh          5
```

4.4 DOCTOR TEST CASE RESULTS

===== DOCTOR'S LOGIN =====

[] Enter Username: daer
[] Enter Password: *****

-----Username not found...Please Give Correct Details-----

===== DOCTOR'S LOGIN =====

[] Enter Username: Dr.Naveen
[] Enter Password: ****

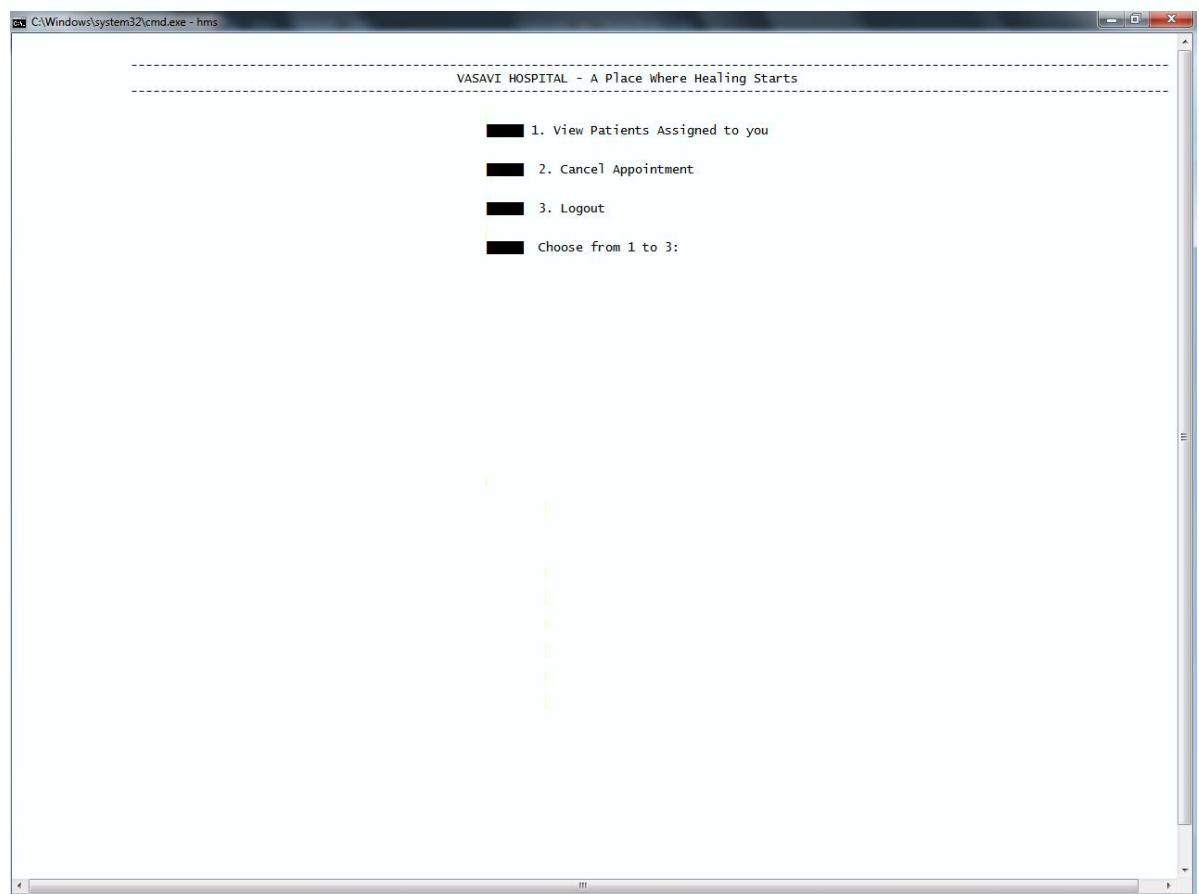
-----Wrong Password...Please Give Correct Details-----

===== DOCTOR'S LOGIN =====

[] Enter Username: Dr.Naveen

[] Enter Password: *****

-----Logging in-----



```
C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
-----
    ■■■ 1. View Patients Assigned to you
    ■■■ 2. Cancel Appointment
    ■■■ 3. Logout
    ■■■ Choose from 1 to 3: 1

    ■■■ DOCTOR NAME : NaveenManda
=====
    ■■■ NO CONSULTATIONS

C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
-----
    ■■■ 1. View Patients Assigned to you
    ■■■ 2. Cancel Appointment
    ■■■ 3. Logout
    ■■■ Choose from 1 to 3: 1

    ■■■ DOCTOR NAME : Dr.Naveen
=====
    ■■■ YOUR APPOINTMENT HAS BEEN WITH PATIENT OF USERNAME - ravikumar
    ■■■ DETAILS OF PATIENT :
NAME : Ravikumar
GENDER : M
AGE : 19
CONTACT : 9234359734
EMAIL : ravi@gmail.com
PROBLEM : Headache
TIME : Tue-Dec-22-09:38:29-2020
=====
    ■■■ YOUR APPOINTMENT HAS BEEN WITH PATIENT OF USERNAME - Sravya
    ■■■ DETAILS OF PATIENT :
NAME : Sravyareddy
GENDER : F
AGE : 25
CONTACT : 9873278732
EMAIL : sra@gmail.com
PROBLEM : Gas
TIME : Tue-Dec-22-09:33:29-2020
```

-----VASAVI HOSPITAL - A Place Where Healing Starts-----
[REDACTED] ENTER PATIENT USERNAME TO DELETE APPOINTMENT : David
-----PATIENT NOT FOUND IN YOUR APPOINTMENTS LIST-----

```
PS C:\Windows\system32\cmd.exe - hms
-----
VASAVI HOSPITAL - A Place Where Healing Starts
----- █ ENTER PATIENT USERNAME TO DELETE APPOINTMENT : Sravya
----- YOUR APPOINTMENT FOR THIS PATIENT IS DELETED-----
```

VASAVI HOSPITAL - A Place Where Healing Starts

[] 1. View Patients Assigned to you

[] 2. Cancel Appointment

[] 3. Logout

[] Choose from 1 to 3: `

Invalid entry. Please Select the Right Option. _

THANK YOU FOR VISITING

Press Any Key.....

5. ADDITIONAL KNOWLEDGE ACQUIRED

This C project has introduced us to different libraries such as: ‘conio.h’, ‘time.h’ and ‘windows.h’. We used the ‘windows.h’ library for controlling the display colours in a controlled manner. We explored the ‘time.h’ and ‘conio.h’ libraries for achieving a look-and-feel of an actual window application by constructing our own time delay function.

We

Also, we have further improved in our knowledge in file-handling because of the vast amount of data manipulation we have done using text files.

Other than this, we have learnt the value of team spirit and have understood the intention behind working in teams. We have learnt to be team players.

6. CONCLUSION AND FUTUREWORK

To conclude, Hospital Management System is a console-based C Project which acts as a **system** for managing the **Hospital** functions and events.

As this project is used for the present day to book appointments and is consoled based, our future work includes incorporating more concepts such as Data Structures in our project. Also, we would like to include a feature which allows the users to make and give appointments for any day in the month.

This project can be further improved by converting it into a Web Application using Python and the Django Framework.

7. REFERENCES

Stack Overflow	https://stackoverflow.com/
C Documentation	https://devdocs.io/c/
Geeks for Geeks	https://www.geeksforgeeks.org/