Please be precise in all sections. Also, avoid to write the same statements again and again.

1. Abstract

Every system needs management to operate normally. So does a hospital for it's smooth work flow. To keep its data intact and accessible, this program has been made. Just think because of data mishap, how many people can end up going to wrong medical specialist. How many appointments would not have taken place just because of slow flow of data.

Thus, we applied the knowledge of C++ OOP gained in class to make a program that covers up such problems in an hospital, clinic, dispensary, etc.

2. Introduction of your project

The project is designed to manage record of three main bodies of a hospital. Through this project, keeping records and their safety is easy. This project provides a clean user interface for patients, doctors, and administration. Doctors can track down their patients, upcoming appointments, and departments, etc. Vice versa for patients. Administration can easily manage other bodies through this.

3. Methodology/ Plan development to solve the Problem

We applied the knowledge we gained in object-oriented programming. The main three modules of this management system are:

- Administration Module.
- Doctor Module.
- Patient Module.

All these modules will have login, add, view and deleting functionalities. Data will be handled by file handling.

1) Administrator Module:

- Sign in to system
- Add a Doctor
- Add a Patient
- Add an appointment
- View all patient details
- View all doctor details
- View all appointment details
- Delete data

2)Patient Module:

- Sign in to system
- Add an appointment
- View appointments

3)Doctor Module:

- Sign in to system
- View patients
- View appointments

4. Specifications

1. We have used different libraries for various purposes.
2. Bool datatype has been used to keep a check on the conditions on various points.
3. We have used global variables that are common in different classes and functions, for ease of accessing and using.
4. Classes have been used for each members with different member functions for functionality. Concepts of inheritance have been demonstrated in the project.
5. File handling is done to manage and store all data records.
6. For choice choosing interface, if-else statements and switch statements have been used.
7. Goto statement has been used to go back to homepage purpose.

5. Code

```
//OOP Project

#include <iostream>
#include <string>
#include <fstream>
#include <windows.h>
using namespace std;
string ch_us, ch_pas, ch_doc, ch_did, ch_ps; int ch_id;
bool ispatient;
bool isadmin;
bool isdoc;
class Doctor {
public:
    void Doc_Login()
    {
        string ds, pss, gen, ag, dep, d_id;
        int n{};
        ifstream read;
        cout << "Welcome to Login Info" << endl;
        cout << "Enter Your Name Dr :"; cin >> ch_doc;
        cout << "Enter your password :"; cin >> ch_ps;
        cout << "Enter your Department :\n"
            "1. Dental\n"
            "2. Accident & Emergency\n"
            "3. Orthopaedics\n"
            "4. Psychiatry\n"
            "5. Physiotherapy\n"
            "-->"; cin >> ch_did;
        if (ch_did == "Dental")
        {
            n = 1;
        }
        else if (ch_did == "Accident & Emergency")
```

```
{
    n = 2;
}
else if (ch_did == "Orthopaedics")
{
    n = 3;
}
else if (ch_did == "Psychiatry")
{
    n = 4;
}
else if (ch_did == "Physiotherapy")
{
    n = 5;
}
else
{
    cout << "Invalid Department";
}

switch (n)
{
case 1:
{

    read.open("Dental.txt");
    while (!read.eof())
    {
        read >> ds;
        read >> pss;
        read >> gen;
        read >> ag;
        read >> dep;
        read >> d_id;

        if (ch_us == ds && ch_pas == pss && ch_did == d_id)
        {
            isdoc = true;
            break;
        }
        else
        {
            cout << "Wrong Input" << endl;

            isdoc = false;
        }
    }
    if (isdoc)
    {
        cout << "You have been successfully logged in!" << endl;
    }
    else
    {
        cout << "Wrong Input" << endl;
    }
    read.close();

}break;
case 2:
{
    read.open("Accident&Emergency.txt");
    while (!read.eof())
    {
        read >> ds;
        read >> pss;
        read >> gen;
        read >> ag;
        read >> dep;
        read >> d_id;

        if (ch_us == ds && ch_pas == pss && ch_did == d_id)
        {
            isdoc = true;
            break;
        }
        else
        {
            cout << "Wrong Input" << endl;

            isdoc = false;
        }
    }
    if (isdoc)
    {
        cout << "You have been sucessfully logged in!" << endl;
    }
    else
    {
        cout << "Wrong Input" << endl;
    }
    read.close();
}break;
case 3:
{
    read.open("Orthopaedics.txt");
    while (!read.eof())
    {
        read >> ds;
        read >> pss;
        read >> gen;
        read >> ag;
        read >> dep;
        read >> d_id;
        if (ch_us == ds && ch_pas == pss && ch_did == d_id)
        {
            isdoc = true;
            break;
        }
        else
        {
            cout << "Wrong Input" << endl;

            isdoc = false;
        }
    }
    if (isdoc)
    {
        cout << "You have been sucessfully logged in!" << endl;
    }
    else
    {
        cout << "Wrong Input" << endl;
    }
    read.close();
}break;
case 4:
{
    read.open("Psychiatry.txt");
    while (!read.eof())
    {
        read >> ds;
        read >> pss;
        read >> gen;
        read >> ag;
        read >> dep;
        read >> d_id;

        if (ch_us == ds && ch_pas == pss && ch_did == d_id)
        {
            isdoc = true;
            break;
        }
        else
        {
            cout << "Wrong Input" << endl;

            isdoc = false;
        }
    }
    if (isdoc)
    {
        cout << "You have been sucessfully logged in!" << endl;
    }
    else
    {
            cout << "Wrong Input" << endl;
        }
        read.close();
    }break;
    case 5:
    {
        read.open("Physiotherapy.txt");
        while (!read.eof())
        {
            read >> ds;
            read >> pss;
            read >> gen;
            read >> ag;
            read >> dep;
            read >> d_id;

            if (ch_us == ds && ch_pas == pss && ch_did == d_id)
            {
                isdoc = true;
                break;
            }
            else
            {
                cout << "Wrong Input" << endl;

                isdoc = false;
            }
        }
        if (isdoc)
        {
            cout << "You have been sucessfully logged in!" << endl;
        }
        else
        {
            cout << "Wrong Input" << endl;
        }
        read.close();
    }break;
    default:
    {
        cout << "Invalid option.";
    }
    break;

    }



}

void view_doctor_appointment()
{
    int departmentno;
    cout << "Enter The Department Of Which Doctor Whose Appointment You
Want To Check :\n";
    cout << "Press 1. Dental\n"
        "Press 2. Accident & Emergency\n"
        "Press 3. Orthopaedics\n"
        "Press 4. Psychiatry\n"
        "Press 5. Physiotherapy\n";
    cout << "-> ";
    cin >> departmentno;
    switch (departmentno)
    {
    case 1:
    {

        string doc1, doc2; int Named;
        ifstream read;
        read.open("Dental.txt");
        cout << "Doctor(s) Available :" << endl;
        while (!read.eof()) {
            if (doc1 == "\0" && doc2 == "\0") {
                cout << "No Doctor Now. \n";
                break;
            }
            read >> doc1;
            read >> doc2;
            cout << endl;
        }
        cout << "Press 1 to check the Appointment of Doctor 1\n And 2 to check the
Appointment of 2 Doctor"; cin >> Named;
        string username, password, Gender; int Age, Time;

        if (Named == 1)
        {
            read.open(doc1 + "txt");

        }

        else
        {
            read.open(doc2 + "txt");

        }
        cout << "Checking the Appointment with Patients :";
        read >> username;
        read >> Gender;
        read >> Age;
        read >> Time;
        read.close();


    }
    case 2:
    {

      string doc1, doc2; int Named;
      ifstream read;
      read.open("Accident&Emergency.txt");
      cout << "Doctor(s) Availabe :" << endl;
      while (!read.eof()) {
        if (doc1 == "\0" && doc2 == "\0") {
          cout << "No Doctor Now. \n";
          break;
        }
        read >> doc1;
        read >> doc2;
        cout << endl;
      }
      cout << "Press 1 to check the Appointment of Doctor 1\n And 2 to check the
Appointment of 2 Doctor"; cin >> Named;
      string username, password, Gender; int Age, Time;

      if (Named == 1)
      {
        read.open(doc1 + "txt");

      }

      else
      {
        read.open(doc2 + "txt");

      }
      cout << "Checking the Appointment with Patients :";
      read >> username;
      read >> Gender;
      read >> Age;
      read >> Time;
      read.close();


    }
    case 3:
    {

      string doc1, doc2; int Named;
      ifstream read;
      read.open("Orthopaedics.txt");
      cout << "Doctor(s) Availabe :" << endl;
      while (!read.eof()) {
        if (doc1 == "\0" && doc2 == "\0") {
          cout << "No Doctor Now. \n";
          break;
        }
```

```
                read >> doc1;
                read >> doc2;
                cout << endl;
            }
            cout << "Press 1 to check the Appointment of Doctor 1\n And 2 to check the
Appointment of 2 Doctor"; cin >> Named;
            string username, password, Gender; int Age, Time;

            if (Named == 1)
            {
                read.open(doc1 + "txt");

            }

            else
            {
                read.open(doc2 + "txt");

            }
            cout << "Checking the Appointment with Patients :";
            read >> username;
            read >> Gender;
            read >> Age;
            read >> Time;
            read.close();


        }
        case 4:
        {

            string doc1, doc2; int Named;
            ifstream read;
            read.open("Psychiatry.txt");
            cout << "Doctor(s) Availabe :" << endl;
            while (!read.eof()) {
                if (doc1 == "\0" && doc2 == "\0") {
                    cout << "No Doctor Now. \n";
                    break;
                }
                read >> doc1;
                read >> doc2;
                cout << endl;
            }
            cout << "Press 1 to check the Appointment of Doctor 1\n And 2 to check the
Appointment of 2 Doctor"; cin >> Named;
            string username, password, Gender; int Age, Time;

            if (Named == 1)
            {
                read.open(doc1 + "txt");
            }

            else
            {
                read.open(doc2 + "txt");

            }
            cout << "Checking the Appointment with Patients :";
            read >> username;
            read >> Gender;
            read >> Age;
            read >> Time;
            read.close();


        }
        case 5:
        {

            string doc1, doc2; int Named;
            ifstream read;
            read.open("Physiotherapy.txt");
            cout << "Doctor(s) Availabe :" << endl;
            while (!read.eof()) {
                if (doc1 == "\0" && doc2 == "\0") {
                    cout << "No Doctor Now. \n";
                    break;
                }
                read >> doc1;
                read >> doc2;
                cout << endl;
            }
            cout << "Press 1 to check the Appointment of Doctor 1\n And 2 to check the
Appointment of 2 Doctor"; cin >> Named;
            string username, password, Gender; int Age, Time;

            if (Named == 1)
            {
                read.open(doc1 + "txt");

            }

            else
            {
                read.open(doc2 + "txt");

            }
            cout << "Checking the Appointment with Patients :";
            read >> username;
            read >> Gender;
            read >> Age;
            read >> Time;
            read.close();


        }
        default:
        {
            cout << "Invalid option.";
        }
        }





    }

};
class Patient {
public:

    void Paitient_Login()
    {
        string us, ps, ge, dep; int id, ag;
        ifstream read;
        read.open("Patient.txt");
        cout << "Welcome to Login Info :" << endl;
        cout << "Enter Your Name :"; cin >> ch_us;
        cout << "Enter your password :"; cin >> ch_pas;
        cout << "Enter your patient ID :"; cin >> ch_id;
        while (!read.eof())
        {
            read >> us;
            read >> ps;
            read >> ge;
            read >> ag;
            read >> dep;
            read >> id;
            if (ch_us == us && ch_pas == ps && ch_id == id)
            {

                ispatient = true;
                break;
            }
            else
            {
                ispatient = false;
            }
        }
        if (ispatient)
        {
            cout << "You Have Been Sucessfully Logged In!" << endl;
        }
        else
        {
            cout << "Wrong Input" << endl;
        }

        read.close();
    }


};


class Administration : public Doctor, public Patient
{
public:

    void Adminstration_Login()
    {

        string adm = "adminadmin";
        string pass = "1234";
        string adm_us, adm_ps;
        cout << "\t\tWelcome to login Info" << endl;
        cout << "Enter Username :"; cin >> adm_us;
        cout << "Enter Your password :"; cin >> adm_ps;
        if (adm_us == adm && adm_ps == pass)
        {
            cout << "You Have Been Logged in Sucessfully!" << endl;
            isadmin = true;
        }
        else
        {
            cout << "Incorrect Username or Password";
            isadmin = false;
        }

    }

    void Add_Doctor()
    {
        string username, password, Gender, Department, D_id;
        int age;
        int departmentno;
        cout << "Note: Add Only Two Doctors Of Each Department\n";
        cout << "Enter The Department of which doctor you want to add:\n";
        cout << "Press 1. Dental\n"
            "Press 2. Accident & Emergency\n"
            "Press 3. Orthopaedics\n"
            "Press 4. Psychiatry\n"
            "Press 5. Physiotherapy\n";
        cout << "-> ";
        cin >> departmentno;
```

```
switch (departmentno)
{
case 1:
{
    cout << "Add Doctor Info\n";
    cout << "Enter His Name :";
    cin >> username;
    cout << "Enter His Password :";
    cin >> password;
    cout << "Enter His Gender :";
    cin >> Gender;
    cout << "Enter His Age :";
    cin >> age;

    ofstream create;
    create.open("Dental.txt", ios::app);
    create << username << " ";
    create << password << " ";
    create << Gender << " ";
    create << age << " ";
    create << Department << endl;

    create.close();
}break;
case 2:
{
    cout << "Add Doctor Info\n";
    cout << "Enter His Name :";
    cin >> username;
    cout << "Enter His Password :";
    cin >> password;
    cout << "Enter His Gender :";
    cin >> Gender;
    cout << "Enter His Age :";
    cin >> age;

    ofstream create;
    create.open("Accident&Emergency.txt", ios::app);
    create << username << " ";
    create << password << " ";
    create << Gender << " ";
    create << age << " ";
    create << Department << endl;

    create.close();
}break;
case 3:
{
    cout << "Add Doctor Info\n";
    cout << "Enter His Name :";
    cin >> username;
    cout << "Enter His Password :";
    cin >> password;
    cout << "Enter His Gender :";
    cin >> Gender;
    cout << "Enter His Age :";
    cin >> age;

    ofstream create;
    create.open("Orthopaedics.txt", ios::app);
    create << username << " ";
    create << password << " ";
    create << Gender << " ";
    create << age << " ";
    create << Department << endl;

    create.close();
}break;
case 4:
{
    cout << "Add Doctor Info\n";
    cout << "Enter His Name :";
    cin >> username;
    cout << "Enter His Password :";
    cin >> password;
    cout << "Enter His Gender :";
    cin >> Gender;
    cout << "Enter His Age :";
    cin >> age;

    ofstream create;
    create.open("Psychiatry.txt", ios::app);
    create << username << " ";
    create << password << " ";
    create << Gender << " ";
    create << age << " ";
    create << Department << endl;

    create.close();
}break;
case 5:
{
    cout << "Add Doctor Info\n";
    cout << "Enter His Name :";
    cin >> username;
    cout << "Enter His Password :";
    cin >> password;
    cout << "Enter His Gender :";
    cin >> Gender;
    cout << "Enter His Age :";
    cin >> age;

    ofstream create;
    create.open("Physiotherapy.txt", ios::app);
```

```
        create << username << " ";
        create << password << " ";
        create << Gender << " ";
        create << age << " ";
        create << Department << endl;

        create.close();
    }break;

    default:
    {
        cout << "Invalid option.";
    }break;


    }

}
void Add_patient()
{
    string username, password, Gender, patientoff;
    int ID, Age;

    cout << "Adding patient Info\n";
    cout << "Enter Your Name :"; cin >> username;
    cout << "Enter Your Password :"; cin >> password;
    cout << "Enter Your Gender :"; cin >> Gender;
    cout << "Enter Your Age :"; cin >> Age;
    cout << "You want to concern the doctor of which Department :\n";
    cout << "1. Dental\n"
        "2. Accident & Emergency\n"
        "3. Orthopaedics\n"
        "4. Psychiatry\n"
        "5. Physiotherapy\n";
    cout << "-> ";
    cin >> patientoff;
    ID = (rand() % 1000) + 1;
    cout << "Your Patient ID is  = " << ID << endl;
    ofstream create;
    create.open("Patient.txt", ios::app);
    create << username << " ";
    create << password << " ";
    create << Gender << " ";
    create << Age << " ";
    create << ID << " ";
    create << patientoff << endl;
    create.close();

}
```

```
    void Add_Appointment()
    {
        int departmentno;
        cout << "Enter The Department of which doctor you want to Make an
appointment:\n";
        cout << "Press 1. Dental\n"
            "Press 2. Accident & Emergency\n"
            "Press 3. Orthopaedics\n"
            "Press 4. Psychiatry\n"
            "Press 5. Physiotherapy\n";
        cout << "-> ";
        cin >> departmentno;

        switch (departmentno) {
        case 1: {

            string doc1, doc2; int Named;
            ifstream read;
            read.open("Dental.txt");
            cout << "Doctor(s) Availabe :" << endl;
            while (!read.eof()) {
                if (doc1 == "\0" && doc2 == "\0") {
                    cout << "No Doctor Availabe Now. \n";
                    break;
                }
                read >> doc1;
                read >> doc2;
                cout << endl;
            }
            cout << "Press 1 to make the Appointment with Doctor 1\n And 2 to make
Appointment with Doctor 2"; cin >> Named;
            string username, password, Gender; int Age, Time;
            ofstream create;
            if (Named == 1)
            {
                create.open(doc1 + "txt");

            }

            else
            {
                create.open(doc2 + "txt");

            }
            cout << "Enter Patient Detail and Time\n";

            cout << "Adding patient Info\n";
            cout << "Enter His Name :"; cin >> username;
            cout << "Enter His Gender :"; cin >> Gender;
            cout << "Enter His Age :"; cin >> Age;
            cout << "Enter the Time :"; cin >> Time;
            create << username << " ";
            create << Gender << " ";
            create << Age << " ";
            create << Time << endl;
            create.close();

        }break;
        case 2: {
            string doc1, doc2; int  Named;
            ifstream read;
            read.open("Accident&Emergency.txt");
            cout << "Doctor(s) Availabe :" << endl;
            while (!read.eof()) {
                if (doc1 == "\0" && doc2 == "\0") {
                    cout << "No Doctor Availabe Now. \n";
                    break;
                }
                read >> doc1;
                read >> doc2;
                cout << endl;
            }
            cout << "Press 1 to make the Appointment with Doctor 1\n And 2 to make
Appointment with Doctor 2"; cin >> Named;
            string username, password, Gender; int Age, Time;
            ofstream create;
            if (Named == 1)
            {
                create.open(doc1 + "txt");

            }

            else
            {
                create.open(doc2 + "txt");

            }
            cout << "Enter Patient Detail and Time\n";

            cout << "Adding patient Info\n";
            cout << "Enter His Name :"; cin >> username;
            cout << "Enter His Gender :"; cin >> Gender;
            cout << "Enter His Age :"; cin >> Age;
            cout << "Enter the Time :"; cin >> Time;
            create << username << " ";
            create << Gender << " ";
            create << Age << " ";
            create << Time << endl;
            create.close();

        }
              break;
        case 3: {
            string doc1, doc2; int  Named;
            ifstream read;
            read.open("Orthopaedics.txt");
            cout << "Doctor(s) Availabe are :" << endl;
            while (!read.eof()) {
                if (doc1 == "\0" && doc2 == "\0") {
                    cout << "No Doctor Availabe Now. \n";
                    break;
                }
                read >> doc1;
                read >> doc2;
                cout << endl;
            }cout << "Press 1 to make the Appointment with Doctor 1\n And 2 to make
Appointment with Doctor 2"; cin >> Named;
            string username, password, Gender; int Age, Time;
            ofstream create;
            if (Named == 1)
            {
                create.open(doc1 + "txt");

            }

            else
            {
                create.open(doc2 + "txt");

            }
            cout << "Enter Patient Detail and Time\n";

            cout << "Adding patient Info\n";
            cout << "Enter His Name :"; cin >> username;
            cout << "Enter His Gender :"; cin >> Gender;
            cout << "Enter His Age :"; cin >> Age;
            cout << "Enter the Time :"; cin >> Time;
            create << username << " ";
            create << Gender << " ";
            create << Age << " ";
            create << Time << endl;
            create.close();



        }
              break;
        case 4: {

            string doc1, doc2; int  Named;
            ifstream read;
            read.open("Psychiatry.txt");
            cout << "Doctor(s) Availabe are :" << endl;
            while (!read.eof()) {
                if (doc1 == "\0" && doc2 == "\0") {
```

```
                cout << "No Doctor Availabe Now. \n";
                break;
            }
            read >> doc1;
            read >> doc2;
            cout << endl;
        }cout << "Press 1 to make the Appointment with Doctor 1\n And 2 to make Appointment with Doctor 2"; cin >> Named;
        string username, password, Gender; int Age, Time;
        ofstream create;
        if (Named == 1)
        {
            create.open(doc1 + "txt");

        }

        else
        {
            create.open(doc2 + "txt");

        }
        cout << "Enter Patient Detail and Time\n";

        cout << "Adding patient Info\n";
        cout << "Enter His Name :"; cin >> username;
        cout << "Enter His Gender :"; cin >> Gender;
        cout << "Enter His Age :"; cin >> Age;
        cout << "Enter the Time :"; cin >> Time;
        create << username << " ";
        create << Gender << " ";
        create << Age << " ";
        create << Time << endl;
        create.close();


    }
          break;
    case 5: {

        string doc1, doc2; int Named;
        ifstream read;
        read.open("Physiotherapy.txt");
        cout << "Doctor(s) Availabe are :" << endl;
        while (!read.eof()) {
            if (doc1 == "\0" && doc2 == "\0") {
                cout << "No Doctor Availabe Now. \n";
                break;
            }
            read >> doc1;
            read >> doc2;
            cout << endl;
        }
```

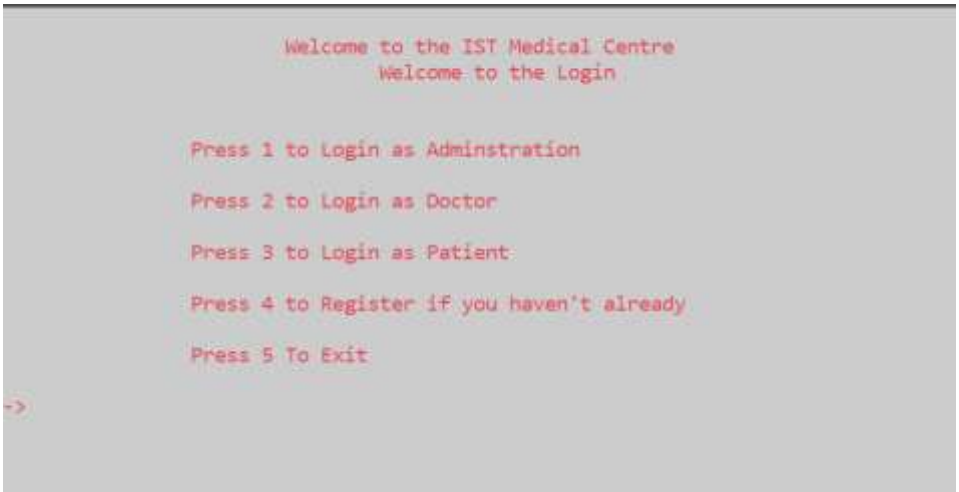
```
      cout << "Press 1 to make the Appointment with Doctor 1\n And 2 to make
Appointment with Doctor 2"; cin >> Named;
      string username, password, Gender; int Age, Time;
      ofstream create;
      if (Named == 1)
      {
        create.open(doc1 + "txt");

      }

      else
      {
        create.open(doc2 + "txt");

      }
      cout << "Enter Patient Detail and Time\n";

      cout << "Adding patient Info\n";
      cout << "Enter His Name :"; cin >> username;
      cout << "Enter His Gender :"; cin >> Gender;
      cout << "Enter His Age :"; cin >> Age;
      cout << "Enter the Time :"; cin >> Time;
      create << username << " ";
      create << Gender << " ";
      create << Age << " ";
      create << Time << endl;
      create.close();

    }
       break;
    default:
    {
      cout << "Invalid option.";
    }
    break;
    }


  }
  void View_Doctors()
  {
    ifstream read;
    int departmentno;
    cout << "Enter The Department of which doctor you want to add:\n";
    cout << "Press 1. Dental\n"
      "Press 2. Accident & Emergency\n"
      "Press 3. Orthopaedics\n"
      "Press 4. Psychiatry\n"
      "Press 5. Physiotherapy\n";
    cout << "-> ";
    cin >> departmentno;
```

```
switch (departmentno)
{
case 1:
{   string username, password, Gender, Department, D_id; int age;

read.open("Dental.txt");
while (!read.eof())
{
    read >> username;
    read >> password;
    read >> Gender;
    read >> age;
    read >> Department;

    read.close();
}

}break;
case 2:
{
    string username, password, Gender, Department, D_id; int age;

    read.open("Accident&Emergency.txt");
    while (!read.eof())
    {
        read >> username;
        read >> password;
        read >> Gender;
        read >> age;
        read >> Department;

        read.close();
    }

}break;
case 3:
{
    string username, password, Gender, Department, D_id; int age;

    read.open("Orthopaedics.txt");
    while (!read.eof())
    {
        read >> username;
        read >> password;
        read >> Gender;
        read >> age;
        read >> Department;

        read.close();
    }

}break;
```

```
    case 4:
    {
        string username, password, Gender, Department, D_id; int age;

        read.open("Psychiatry.txt");
        while (!read.eof())
        {
            read >> username;
            read >> password;
            read >> Gender;
            read >> age;
            read >> Department;
            read.close();
        }

    }break;
    case 5:
    {
        string username, password, Gender, Department, D_id; int age;

        read.open("Physiotherapy.txt");
        while (!read.eof())
        {
            read >> username;
            read >> password;
            read >> Gender;
            read >> age;
            read >> Department;
            read.close();
        }

    }break;
    default:
    {
        cout << "Invalid option.";
    }
    break;

    }


}
void View_Patient()
{
    string username, password, Gender, Age, ID, patientoff;
    ifstream read;
    read.open("Patient.txt");
    while (!read.eof())
    {
        read >> username;
        read >> password;
        read >> Gender;
```

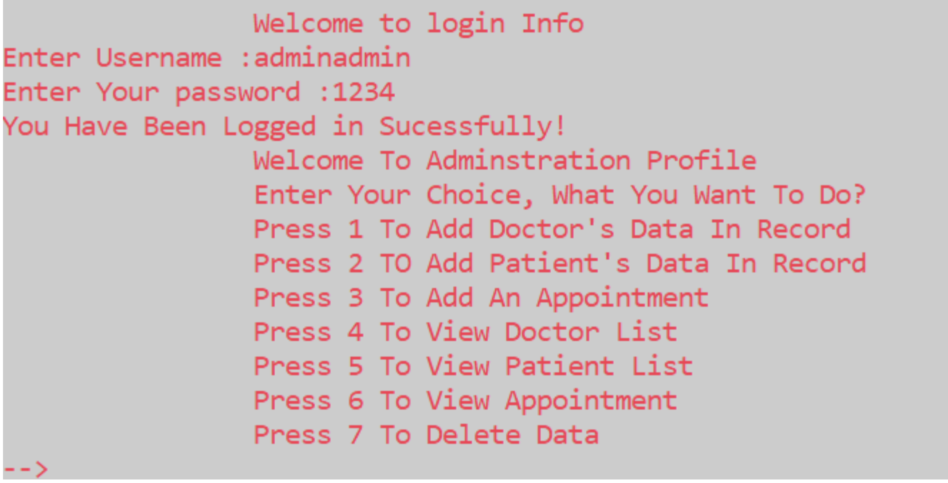
```
            read >> Age;
            read >> ID;
            read >> patientoff;
            read.close();
        }


    }
    void del_data()
    {
        int decision;
        cout << "\t\tWhose Data You Want To Remove?\n";
        cout << "\t\tPress 1 to Remove Doctor Data\n"
            "\t\tPress 2 to Remove Patient Data\n";
        cout << "-->"; cin >> decision;
        switch (decision)
        {
        case 1:
        {
            int departmentno;
            cout << "Enter The Department of Which Doctor Data You Want To
Remove:\n";
            cout << "Press 1. Dental\n"
                "Press 2. Accident & Emergency\n"
                "Press 3. Orthopaedics\n"
                "Press 4. Psychiatry\n"
                "Press 5. Physiotherapy\n";
            cout << "-> ";
            cin >> departmentno;
            switch (departmentno)
            {
            case 1:
            {
                cout << "\t\tRemoving The Dental Doctor's Data.\n";
                remove("Dental");
            }
            break;
            case 2:
            {
                cout << "\t\tRemoving The Accident&Emergency Doctor's Data.\n";
                remove("Accident&Emergency");
            }
            break;
            case 3:
            {
                cout << "\t\tRemoving The Orthopaedics Doctor's Data.\n";
                remove("Orthopaedics");
            }
            break;
            case 4:
            {
                cout << "\t\tRemoving The Psychaitry Doctor's Data.\n";
                remove("Psychiatry");
            }
            break;
            case 5:
            {
                cout << "\t\tRemoving The Physiotherapy Doctor's Data.\n";
                remove("Physiotherapy");
            }
            break;
            }

        }break;
        case 2:
        {
            cout << "\t\tRemoving Patient Data From Records\n";
            remove("Patient.txt");

        }break;
        default:
        {
            cout << "Wrong Input";
        }break;
        }

    }
} Adm;


void adm_fun()
{
    int decision, choice;
    do {


        cout << "\t\tWelcome To Adminstration Profile\n";
        cout << "\t\tEnter Your Choice, What You Want To Do?\n";
        cout << "\t\tPress 1 To Add Doctor's Data In Record\n"
            "\t\tPress 2 TO Add Patient's Data In Record\n"
            "\t\tPress 3 To Add An Appointment\n"
            "\t\tPress 4 To View Doctor List\n"
            "\t\tPress 5 To View Patient List\n"
            "\t\tPress 6 To View Appointment\n"
            "\t\tPress 7 To Delete Data\n";
        cout << "-->"; cin >> decision;
        switch (decision)
        {
        case 1:
        {
            Adm.Add_Doctor();
        }
        case 2:
        {
            Adm.Add_patient();
        }
        case 3:
        {
            Adm.Add_Appointment();
        }
        case 4:
        {
            Adm.View_Doctors();
        }
        case 5:
        {
            Adm.View_Patient();
        }
        case 6:
        {
            Adm.view_doctor_appointment();
        }
        case 7:
        {
            Adm.del_data();
        }
        default:
        {
            cout << "Invalid option.";
        }
        }


        cout << "Do You want to do Another Task :\n"
            "If Yes Press 1\n"
            "If No Press 0\n";
        cout << "->"; cin >> choice;
    } while (choice != 0);
}
void pat_fun()
{

    int decision, choice;
    do {

        cout << "\t\tWelcome to Patient Profile\n";
        cout << "\t\tPress 1 To Make an Appointment\n"
            "\t\tPress 2 to View Appointment\n";
        cout << "-->"; cin >> decision;
        if (decision == 1)
        {
            Adm.Add_Appointment();
        }
        else if (decision == 2)
        {
            Adm.view_doctor_appointment();
        }

        else
        {
            cout << "Wrong Input\n";
        }

        cout << "Do You want to do Another Task :\n"
            "If Yes Press 1\n"
            "If No Press 0\n";
        cout << "->"; cin >> choice;
    } while (choice != 0);


}
void doc_fun()
{

    int decision, choice;
    do {


        cout << "\t\tWelcome to Doctor Profile\n";
        cout << "\t\tPress 1 To View Patient\n"
            "\t\tPress 2 To View Appointment\n";
        cout << "-->"; cin >> decision;

        if (decision == 1)
        {
            Adm.View_Patient();
        }
        else if (decision == 2)
        {
            Adm.view_doctor_appointment();
        }
        else
        {
            cout << "Wrong Input\n";
        }


        cout << "Do You want to do Another Task :\n"
            "If Yes Press 1\n"
            "If No Press 0\n";
        cout << "->"; cin >> choice;
    } while (choice != 0);

}


int main()
{
    int choice;

    cout << "\n\t\t\tWelcome to the IST Medical Centre";
    cout << endl;
    system("Color 7C");
Home:

    cout << "\t\t\t\tWelcome to the Login\n";
    cout << endl;
    cout << "\n\t\tPress 1 to Login as Adminstration\n"
        "\n\t\tPress 2 to Login as Doctor\n"
        "\n\t\tPress 3 to Login as Patient\n"
        "\n\t\tPress 4 to Register if you haven't already\n"
        "\n\t\tPress 5 To Exit\n" << endl;
    cout << "->"; cin >> choice;
    switch (choice)
    {
    case 1:
    {
        Adm.Adminstration_Login();
    }break;
    case 2:
    {
        Adm.Doc_Login();
    }break;
    case 3:
    {
        Adm.Paitient_Login();
    }break;
    case 4:
    {
        Adm.Add_patient();
    }break;
    case 5:
    {
        exit(0);
    }
    default:
    {
        cout << "Invalid option.";
    }break;
    }
    if (isadmin)
    {
        adm_fun();
        goto Home;
    }
    if (isdoc)
    {
        doc_fun();
        goto Home;
    }
    if (ispatient)
    {
        pat_fun();
        goto Home;
    }


}
```

6. Results Discussion



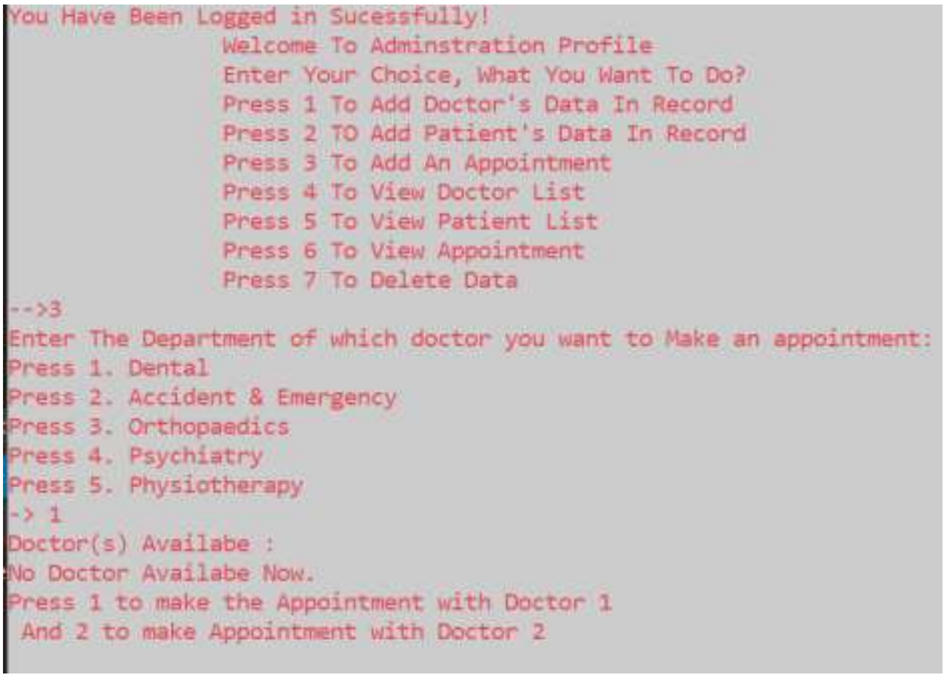
```
                    Welcome to the IST Medical Centre
                          Welcome to the Login


            Press 1 to Login as Adminstration

            Press 2 to Login as Doctor

            Press 3 to Login as Patient

            Press 4 to Register if you haven't already

            Press 5 To Exit

->
```

When programs start, user sees such interface. The user can register himself or herself. Login as a patient, doctor, or admin. Or ultimately close the program. The reason for choosing red color on white background is that this color coordination a norm in medical field.



```
                Welcome to login Info
Enter Username :adminadmin
Enter Your password :1234
You Have Been Logged in Sucessfully!
                Welcome To Adminstration Profile
                Enter Your Choice, What You Want To Do?
                Press 1 To Add Doctor's Data In Record
                Press 2 TO Add Patient's Data In Record
                Press 3 To Add An Appointment
                Press 4 To View Doctor List
                Press 5 To View Patient List
                Press 6 To View Appointment
                Press 7 To Delete Data
-->
```

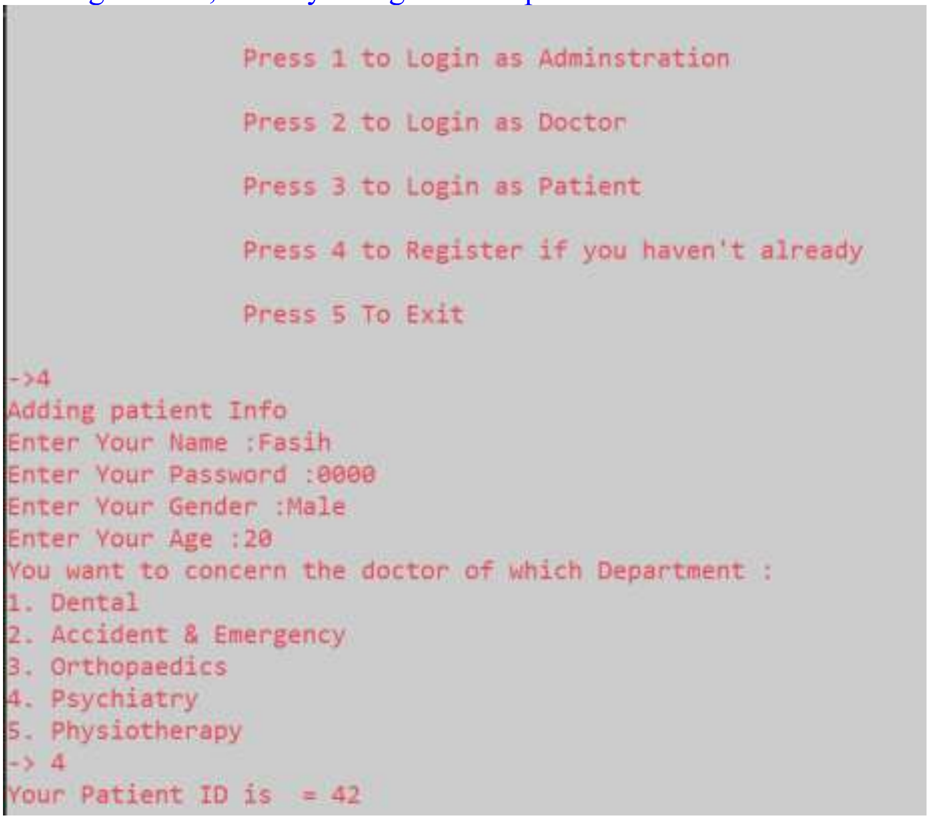
The login entries are set as default:
Username: adminadmin
Password: 1234
When logged in successfully as an admin, there are options related to admin like adding data of doctors or admin, viewing the data or delete the data.



```
You Have Been Logged in Sucessfully!
                Welcome To Adminstration Profile
                Enter Your Choice, What You Want To Do?
                Press 1 To Add Doctor's Data In Record
                Press 2 TO Add Patient's Data In Record
                Press 3 To Add An Appointment
                Press 4 To View Doctor List
                Press 5 To View Patient List
                Press 6 To View Appointment
                Press 7 To Delete Data
-->3
Enter The Department of which doctor you want to Make an appointment:
Press 1. Dental
Press 2. Accident & Emergency
Press 3. Orthopaedics
Press 4. Psychiatry
Press 5. Physiotherapy
-> 1
Doctor(s) Availabe :
No Doctor Availabe Now.
Press 1 to make the Appointment with Doctor 1
 And 2 to make Appointment with Doctor 2
```

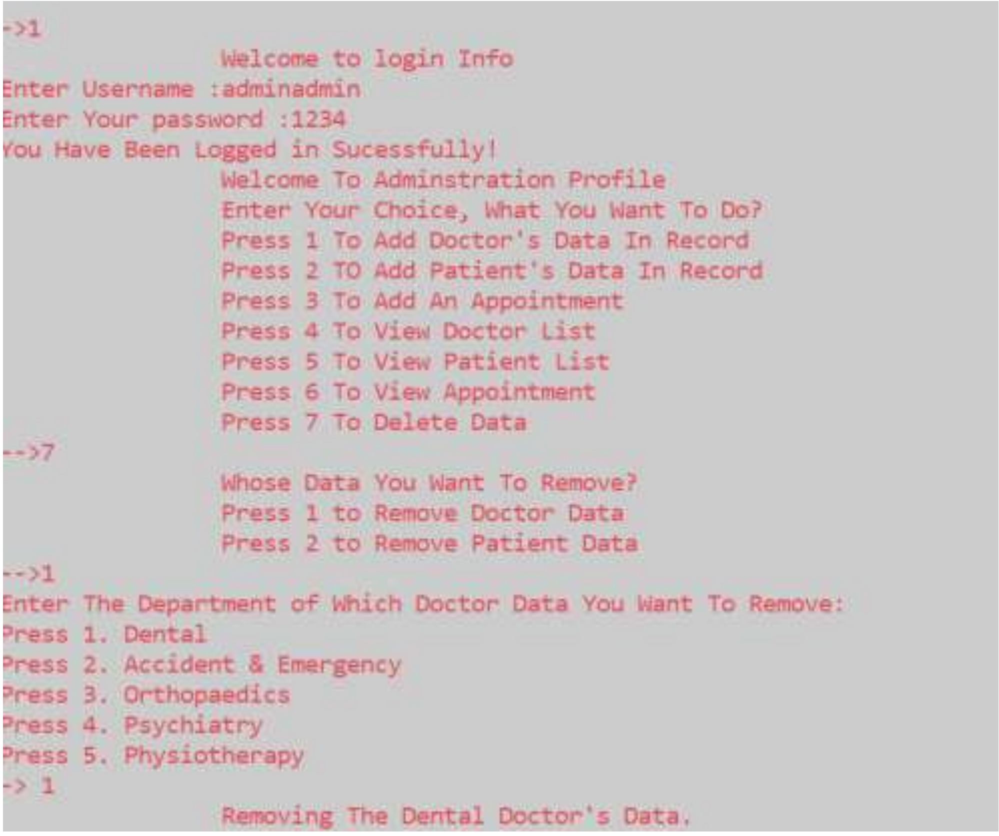
As, we have logged in as admin and checked that there are no doctors available, the programs ask us to add doctors for appointment.

Moving onward, let’s try to register as a patient



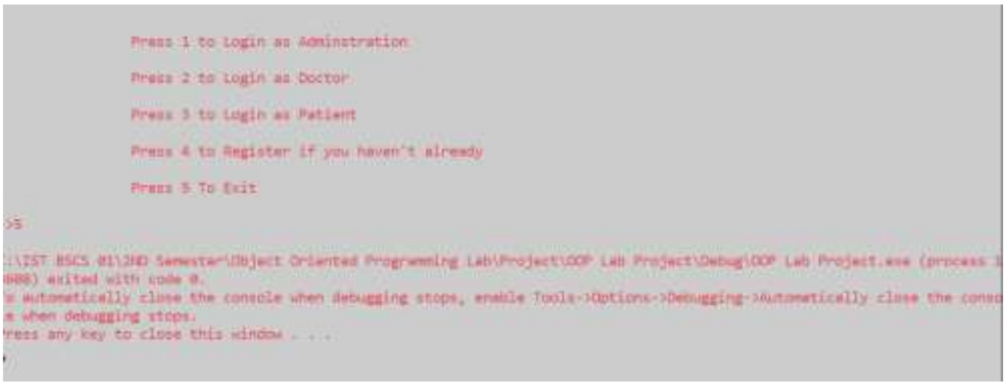
```
                Press 1 to Login as Adminstration

                Press 2 to Login as Doctor

                Press 3 to Login as Patient

                Press 4 to Register if you haven't already

                Press 5 To Exit

->4
Adding patient Info
Enter Your Name :Fasih
Enter Your Password :0000
Enter Your Gender :Male
Enter Your Age :20
You want to concern the doctor of which Department :
1. Dental
2. Accident & Emergency
3. Orthopaedics
4. Psychiatry
5. Physiotherapy
-> 4
Your Patient ID is  = 42
```

Upon selecting registration, we can see that program asked us to enter name, password, gender and age. Upon entering data, the program gave us patient ID through random function.



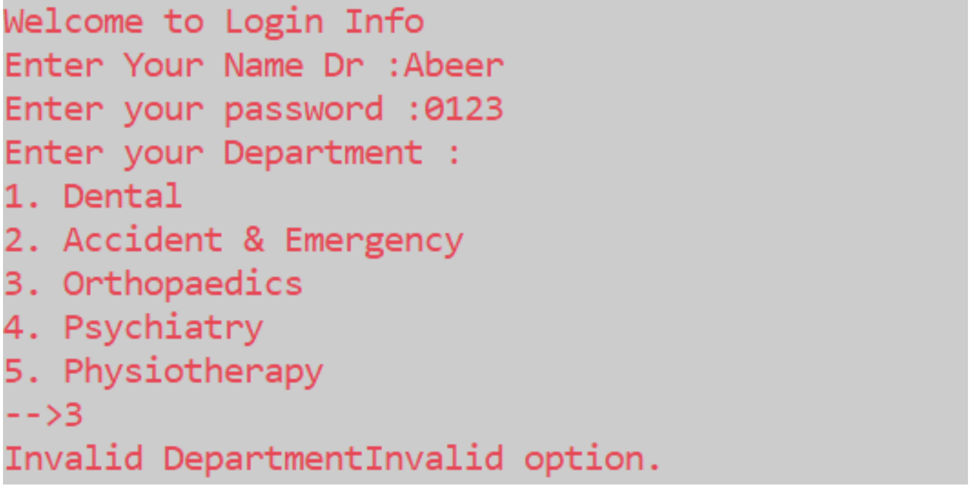
```
->1
                Welcome to login Info
Enter Username :adminadmin
Enter Your password :1234
You Have Been Logged in Sucessfully!
                Welcome To Adminstration Profile
                Enter Your Choice, What You Want To Do?
                Press 1 To Add Doctor's Data In Record
                Press 2 TO Add Patient's Data In Record
                Press 3 To Add An Appointment
                Press 4 To View Doctor List
                Press 5 To View Patient List
                Press 6 To View Appointment
                Press 7 To Delete Data
-->7
                Whose Data You Want To Remove?
                Press 1 to Remove Doctor Data
                Press 2 to Remove Patient Data
-->1
Enter The Department of Which Doctor Data You Want To Remove:
Press 1. Dental
Press 2. Accident & Emergency
Press 3. Orthopaedics
Press 4. Psychiatry
Press 5. Physiotherapy
-> 1
                Removing The Dental Doctor's Data.
```

Demonstrating delete function.



```
          Press 1 to Login as Adminstration

          Press 2 to Login as Doctor

          Press 3 to Login as Patient

          Press 4 to Register if you haven't already

          Press 5 To Exit

>5
:\IST BSCS 01\2ND Semester\Object Oriented Programming Lab\Project\OOP Lab Project\Debug\OOP Lab Project.exe (process 1
808) exited with code 0.
o automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the conso
e when debugging stops.
ress any key to close this window . . .
```

Exiting program.



```
Welcome to Login Info
Enter Your Name Dr :Abeer
Enter your password :0123
Enter your Department :
1. Dental
2. Accident & Emergency
3. Orthopaedics
4. Psychiatry
5. Physiotherapy
-->3
Invalid DepartmentInvalid option.
```

Invalidity is given upon wrong login.

7. Conclusion

This project is computerizing the work in a hospital, it is more convenient than the manual system. The computerization makes the managing process fast, safe and reliable. The program is thoroughly checked by entering dummy data, it worked as per the requirement, performing all functions. The program is great for a small, normal hospital. In future it can have features like surgery record management, pharmacy management, research management, blood and organ donation management, equipment management and salary management, etc.

8. References

1. https://medium.com/@auedbaki/c-program-for-hospital-management-system-240623fc048f
2. https://www.geeksforgeeks.org/hospital-management-system-in-c/