**A PROJECT REPORT ON**

**"MEDICAL INFORMATION SYSTEM"**

SUBMITTED BY:

**Miss. Pranjal Prafulla Raut(2124UCEF1003)**

SUBJECT:

**CORE C++ PROGRAMMING**

Under the guidance of

**Miss. Ishwari Tirse**



**Department of Computer Science and Engineering**

Sanjivani Rural Education Society's

**SANJIVANI UNIVERSITY**

KOPARGAON-423603, DIST: AHMEDNAGAR

2024-2025

**INDEX**

|  |  |  |
| --- | --- | --- |
| **SR.NO** | **CONTENT** | **PAGE NO.** |
| **1.** | **INTRODUCTION** | **3** |
| **2.** | **CODE** | **4** |
| **3.** | **OUTPUT** | **8** |
| **4.** | **CONCLUSION** | **10** |

**INTRODUCTION**

The Medical Information System can help any hospital or healthcare facility by boosting productivity, lowering paperwork, and enhancing patient care.

The Medical Information System in C++ simplifies the process for hospitals and other healthcare organizations to manage medical records, billing, and appointments. Among the information stored in the system are vital signs, medical history, medication details, laboratory results, and billing information. Medical Information system that keeps records of doctors, their appointments, patients, staff, and more using object-oriented programming and file handling (in C++).

**CODE**

#include <iostream>

#include <vector>

#include <string>

using namespace std;

class Patient {

public:

string name;

int age;

string disease;

Patient(string n, int a, string d) : name(n), age(a), disease(d) {}

};

class Doctor {

public:

string name;

string specialization;

Doctor(string n, string s) : name(n), specialization(s) {}

};

class Appointment {

public:

Patient patient;

Doctor doctor;

string date;

Appointment(Patient p, Doctor d, string dt) : patient(p), doctor(d), date(dt) {}

};

class HospitalManagementSystem {

private:

vector<Patient> patients;

vector<Doctor> doctors;

vector<Appointment> appointments;

public:

void addPatient() {

string name;

int age;

string disease;

cout << "Enter patient name: ";

cin >> name;

cout << "Enter patient age: ";

cin >> age;

cout << "Enter patient disease: ";

cin >> disease;

patients.push\_back(Patient(name, age, disease));

cout << "Patient added successfully!\n";

}

void addDoctor() {

string name;

string specialization;

cout << "Enter doctor name: ";

cin >> name;

cout << "Enter doctor specialization: ";

cin >> specialization;

doctors.push\_back(Doctor(name, specialization));

cout << "Doctor added successfully!\n";

}

void scheduleAppointment() {

string patientName;

string doctorName;

string date;

cout << "Enter patient name: ";

cin >> patientName;

cout << "Enter doctor name: ";

cin >> doctorName;

cout << "Enter appointment date (DD/MM/YYYY): ";

cin >> date;

// Find the patient

Patient\* patientPtr = nullptr;

for (auto& patient : patients) {

if (patient.name == patientName) {

patientPtr = &patient;

break;

}

}

// Find the doctor

Doctor\* doctorPtr = nullptr;

for (auto& doctor : doctors) {

if (doctor.name == doctorName) {

doctorPtr = &doctor;

break;

}

}

if (patientPtr && doctorPtr) {

appointments.push\_back(Appointment(\*patientPtr, \*doctorPtr, date));

cout << "Appointment scheduled successfully!\n";

} else {

cout << "Patient or Doctor not found!\n";

}

}

void showAppointments() {

if (appointments.empty()) {

cout << "No appointments scheduled.\n";

return;

}

cout << "Scheduled Appointments:\n";

for (const auto& appointment : appointments) {

cout << "Patient: " << appointment.patient.name

<< ", Doctor: " << appointment.doctor.name

<< ", Date: " << appointment.date << "\n";

}

}

};

int main() {

HospitalManagementSystem hms;

int choice;

do {

cout << "\nHospital Management System\n";

cout << "1. Add Patient\n";

cout << "2. Add Doctor\n";

cout << "3. Schedule Appointment\n";

cout << "4. Show Appointments\n";

cout << "5. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

hms.addPatient();

break;

case 2:

hms.addDoctor();

break;

case 3:

hms.scheduleAppointment();

break;

case 4:

hms.showAppointments();

break;

case 5:

cout << "Exiting...\n";

break;

default:

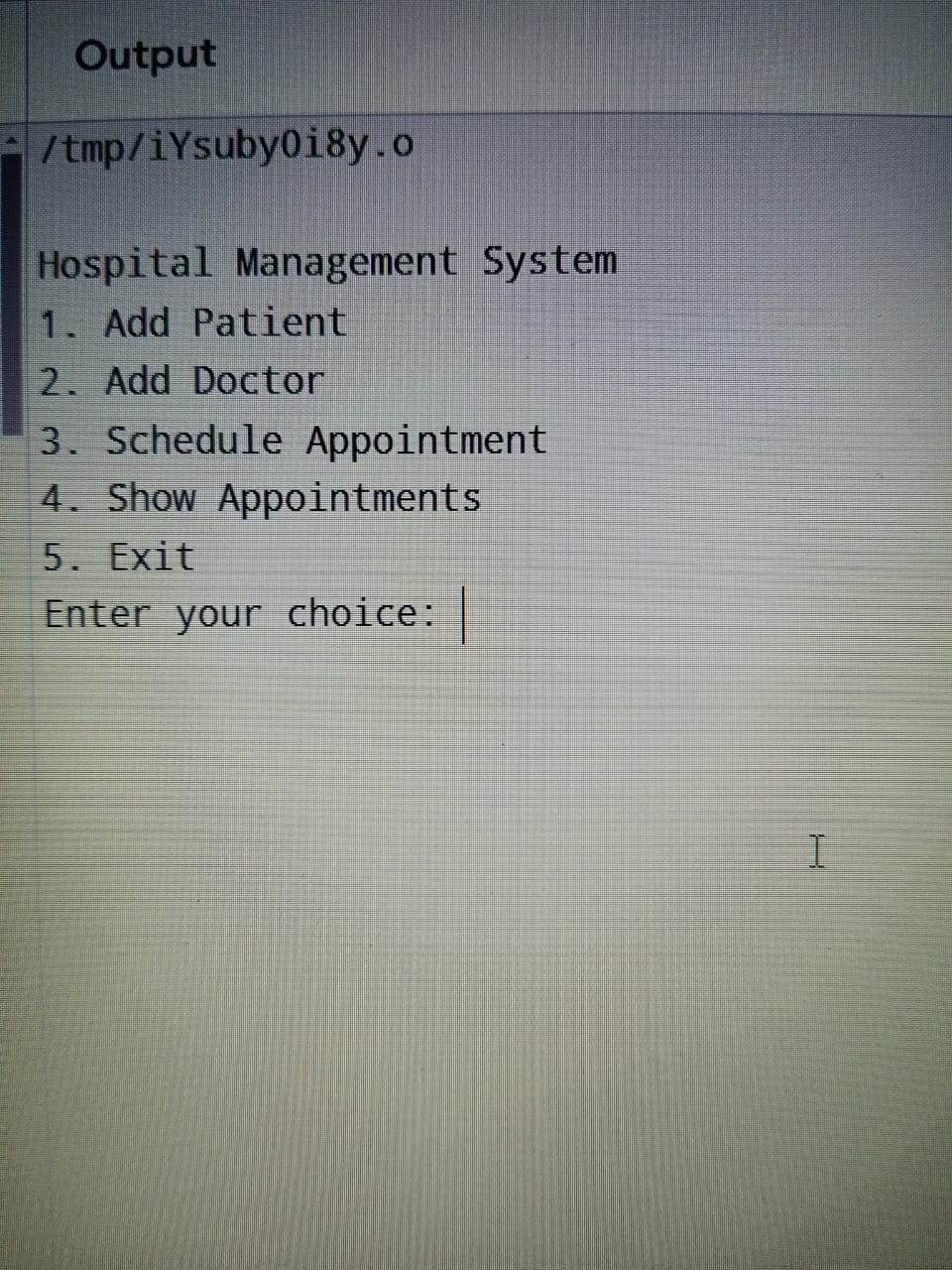
cout << "Invalid choice! Please try again.\n";

}

} while (choice != 5);

return 0;

}



**CONCLUSION**

The Medical Information system mini project in C++ showcases the effective use of programming to streamline hospital operations. Through features such as

patient registration, appointment scheduling, and medical record management, the system enhances efficiency and accessibility for both staff and patients.

Overall, the project serves as a practical demonstration of how technology can improve hospital operations and patient care.