# Small Clinic Management System - Documentation

## 1. Object-Oriented Analysis (OOA)

Following the 4-step OOA model for the Small Clinic Management System:

### Objects (Nouns)

- Patient  
- ChronicPatient  
- Doctor  
- Appointment  
- Prescription  
- Bill  
- ClinicSystem

### Attributes (Descriptive Nouns)

- Patient: name, id, age, medicalHistory  
- ChronicPatient: inherits Patient + conditionType, lastCheckupDate  
- Doctor: name, id, specialty  
- Appointment: date, time, reason, status, patientID, doctorID, Prescription, Bill  
- Prescription: medicines, notes  
- Bill: services, amount, status, insuranceCovered  
- ClinicSystem: patients, doctors, appointments

### Methods (Verbs)

- Patient: scheduleAppointment(), updateHistory(), displayInfo()  
- ChronicPatient: scheduleAppointment() (override), displayInfo() (override)  
- Doctor: displayInfo()  
- Appointment: updateStatus(), addPrescription(), displayInfo(), getPrescription(), getBill()  
- Prescription: addMedicine(), setNotes(), displayPrescription()  
- Bill: addService(), calculateTotal(), applyInsurance(), markPaid(), displayBill()  
- ClinicSystem: addPatient(), addDoctor(), createAppointment(), cancelAppointment(), showAllAppointments()

### Inheritance

- ChronicPatient inherits from Patient.  
- Appointment has a composition relationship with Prescription and Bill.  
- ClinicSystem manages Patients, Doctors, and Appointments.

## 2. Class Design Explanation

The system is designed using OOP principles:  
- Encapsulation: Attributes are private and accessed via methods.  
- Inheritance: ChronicPatient extends Patient to include additional attributes and overrides methods.  
- Composition: Appointment contains a Prescription and a Bill.  
- Polymorphism: scheduleAppointment() is overridden in ChronicPatient to add extra behavior.  
This design ensures code reuse, flexibility, and clear separation of responsibilities.

## 3. Code Walkthrough

Key parts of the code:  
- Appointment class: Stores appointment details, linked to Prescription and Bill.  
- Patient class: Represents a generic patient with medical history.  
- ChronicPatient class: Specialization of Patient, requires more frequent checkups.  
- Doctor class: Contains doctor information such as name and specialty.  
- Prescription class: Holds prescribed medicines and notes.  
- Bill class: Manages service charges, payment status, and insurance coverage.  
- ClinicSystem class: Central system to manage patients, doctors, and appointments.

## 4. Test Results

Sample test case:  
- Added two doctors (Dr. Smith and Dr. Alice).  
- Added two patients (John Doe as normal patient, Jane Doe as chronic patient).  
- Created two appointments: one for routine checkup, one for diabetes follow-up.  
- Added prescription and insurance coverage for Jane Doe’s appointment.  
Output demonstrates:  
- Appointments with prescription and bills displayed.  
- Insurance applied correctly, showing 'Paid (Covered by Insurance)'.

OUTPUT:

Appointment on 2025-09-10 at 10:00 | Reason: Routine Checkup | Status: Scheduled | PatientID: 101 | DoctorID: 1

----- Bill -----

Consultation Fee: $50

Blood Test: $30

Medicine: $20

Total: $100

Payment Status: Unpaid

Appointment on 2025-09-10 at 10:00 | Reason: Routine Checkup | Status: Scheduled | PatientID: 101 | DoctorID: 1

----- Bill -----

Consultation Fee: $50

Blood Test: $30

Medicine: $20

Total: $100

Payment Status: Paid

Appointment on 2025-09-12 at 14:00 | Reason: Follow-up | Status: Scheduled | PatientID: 102 | DoctorID: 2

----- Bill -----

Consultation Fee: $50

Total: $50

Covered by insurance.

Payment Status: Paid (Covered by Insurance)

PS C:\Users\Admin\Desktop> cd "c:\Users\Admin\Desktop\" ; if ($?) { g++ OOP4\_24110076\_PhamTuanAnh.cpp -o OOP4\_24110076\_PhamTuanAnh } ; if ($?) { .\OOP4\_24110076\_PhamTuanAnh }

Appointment scheduled for patient John Doe

Chronic patient Jane Doe requires frequent checkups.

Appointment on 2025-09-10 at 10:00 | Reason: Routine Checkup | Status: Scheduled | PatientID: 101 | DoctorID: 1

Prescription:

----- Bill -----

Consultation Fee: $50

Total: $50

Appointment on 2025-09-15 at 14:00 | Reason: Diabetes Follow-up (Chronic Care) | Status: Scheduled | PatientID: 102 | DoctorID: 2

Prescription:

- Metformin : 500mg twice a day

- Insulin : 10 units daily

Notes: Check blood sugar regularly

----- Bill -----

Consultation Fee: $50

Chronic Care Fee: $20

Total: $70

Payment Status: Paid (Covered by Insurance)

## 5. LLM Usage

I used ChatGPT to assist in brainstorming ideas for methods and class design. For example, I asked: 'Suggest methods for an Appointment class in a clinic system.' ChatGPT provided ideas such as updateStatus() and displayInfo(), which I adapted into my implementation.  
  
I also used ChatGPT to refine explanations and help with generating documentation structure. However, all source code was written by me and tested independently.