

Flutter & Dart Complete Guide

Prepared by Eng. Yousef Salah Kassem

DAFL 1
Dart OOP Problem Solving

Lecture 8



Chapter
08

Table Of Content

- Hospital Management System



Hospital Management System

Hospital System

Comprehensive Hospital Management System Using OOP Principles

Scenario/Problem Statement

You are hired as a **Software Architect** to design a **Hospital Management System (HMS)** for a large hospital. The HMS will streamline operations, improve efficiency, and ensure proper patient care. This system must incorporate all **OOP Principles** (Encapsulation, Inheritance, Polymorphism, and Abstraction) while addressing the following real-world requirements.



Hospital System

Problem Requirements

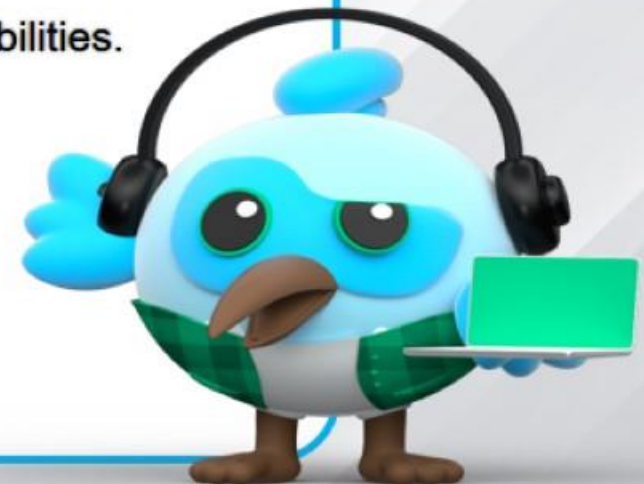
Functional Requirements:

1. Department Management:

- The hospital has multiple departments (e.g., Emergency, Cardiology, Pediatrics).
- Each department has a **Department Head** and **Staff Members** (Doctors, Nurses, and Technicians).

2. Staff Management:

- Staff members (Doctors, Nurses, Technicians) have unique roles and responsibilities.
- Doctors have specializations (e.g., Cardiologist, Pediatrician).
- Nurses have assigned shifts and are responsible for specific wards or units.
- Technicians handle equipment and assist in diagnostic tests.



Hospital System

Patient Management:

- Each patient has a **Medical Record** (including name, age, diagnosis, treatments, and billing).
- Patients can be categorized as **Outpatients** or **Inpatients**.
 - **Outpatients** book appointments with doctors.
 - **Inpatients** are admitted to hospital rooms and monitored daily.

Appointment Scheduling:

- Patients book appointments with available doctors based on time slots.

Room Management:

- Rooms are classified as **General Wards** and **Private Suites** with different charges.
- Admitted patients are assigned to rooms based on availability.



Hospital System

Billing System:

- The system should calculate the total bill for each patient based on:
 - Room charges.
 - Treatments provided.
 - Consultation fees.

Hospital Operations:

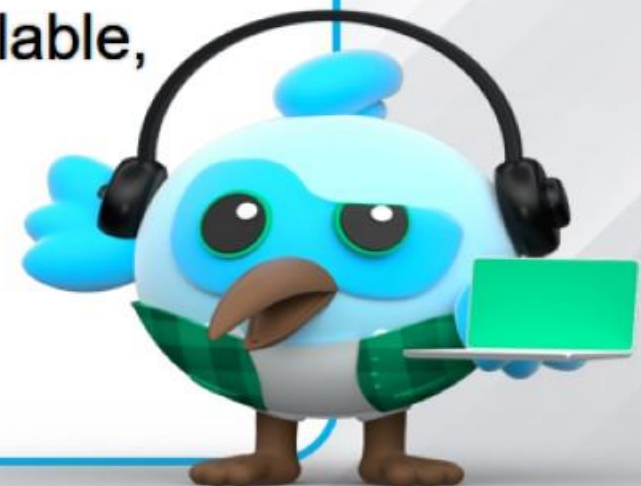
- Generate **daily reports** for:
 - Patients admitted and discharged.
 - Staff on duty.
 - Revenue generated.



Hospital System

Non-Functional Requirements:

- Use **Object-Oriented Design** principles to ensure modularity, reusability, and scalability.
- Code should be **well-documented** and **readable** for future maintainability.
- System should handle **edge cases** (e.g., no rooms available, double booking of doctors).



Hospital System

Guidelines for the Assignment

1. Encapsulation:

- Ensure private data members for all classes.
- Provide getter and setter methods for controlled access.

2. Abstraction:

- Use abstract classes or interfaces for common behaviors, such as **Staff**, **Room**, or **Report**.

3. Inheritance:

- Create specialized classes using base classes.
- Example: **Doctor**, **Nurse**, and **Technician** inherit from **Staff**.

4. Polymorphism:

- Use overridden methods to provide specific implementations.
- Example: **GeneralWard** and **PrivateSuite** override **calculateCharges()**.

5. Bonus: Use composition to create relationships between objects

(e.g., **Department** contains **Staff** and **Patients**).



Hospital System

Deliverables

- A fully implemented solution in **Dart**.
- Include:
 - **Class Diagrams** for the system design.
 - The complete **source code** with clear documentation.
 - **Test cases** to verify functionality.



Hospital System

Dart Implementation Outline

Below is a **brief outline** of the classes you need to implement:

1. Abstract Class: **Staff**

- Common properties: **id**, **name**, **role**.
- Abstract method: **work()**.

2. Classes for Inheritance:

- **Doctor**: Specialization, availability schedule.
- **Nurse**: Shift, assigned ward/unit.
- **Technician**: Assigned equipment, responsibilities.



Hospital System

Abstract Class: **Room**

- Properties: **roomNumber**, **dailyCharge**.
- Methods: **cleanRoom()**, **calculateCharges()**.

Polymorphic Classes:

- **GeneralWard**: Shared room, lower charges.
- **PrivateSuite**: Single room, higher charges.

Class: **Patient**

- Properties: **id**, **name**, **diagnosis**, **medicalRecord**, **roomAssigned**.
- Methods: **assignRoom(Room room)**, **addTreatment(String treatment)**.



Hospital System

Class: **Appointment**

- Properties: **id**, **patient**, **doctor**, **timeSlot**.
- Methods: **schedule()**, **cancel()**.

Class: **Department**

- Properties: **name**, **staff**, **patients**.
- Methods: **addStaff(Staff staff)**, **assignPatient(Patient patient)**.



Hospital System

Class: **Billing**

- Methods: `calculateBill(Patient patient)`.

Class: **Hospital**

- Properties: `name`, `departments`, `patients`, `appointments`.
- Methods: `generateDailyReport()`, `admitPatient()`, `dischargePatient()`.



Department Cardiology added to the hospital.
Department Pediatrics added to the hospital.
Dr. Smith has been added to the Cardiology department.
Nurse Jane has been added to the Pediatrics department.
Tech John has been added to the Pediatrics department.
John Doe has been assigned to room 101.
John Doe has been admitted to the hospital.
Alice Brown has been assigned to room 201.
Alice Brown has been admitted to the hospital.
Appointment scheduled for John Doe with Dr. Smith at 2024-11-18 15:00:00.000.
Total bill for John Doe: \$500.0
Total bill for Alice Brown: \$1500.0
Daily Report for City General Hospital:
Total Patients Admitted: 2
Departments: 2
Appointments: 1

What is Next ?



Thank You