OOPS PROJECT

121CS0026
121CS0061
121CS0068
121CS0071

HOSPITAL MANAGEMENT SYSTEM

Problem Statement

Different classes

Inheritance Diagram

Data Hiding Principles

Summary

PROBLEM STATEMENT

The issue is that hospitals are having difficulty managing their operations in an efficient and effective manner due to a lack of a comprehensive and integrated Hospital Management System (HMS). Existing systems are frequently out of date, disconnected, and lack real-time data, making it difficult for hospital staff to make informed decisions and provide high-quality patient care. This causes delays, errors, and inefficiencies in daily hospital operations such as patient registration, appointment scheduling, clinical documentation, pharmacy management, inventory management, and billing. As a result, a modern and user-friendly HMS that can integrate all hospital functions and provide real-time access to data for improved decision-making, better resource allocation, and improved patient outcomes is required.

OBJECTIVES

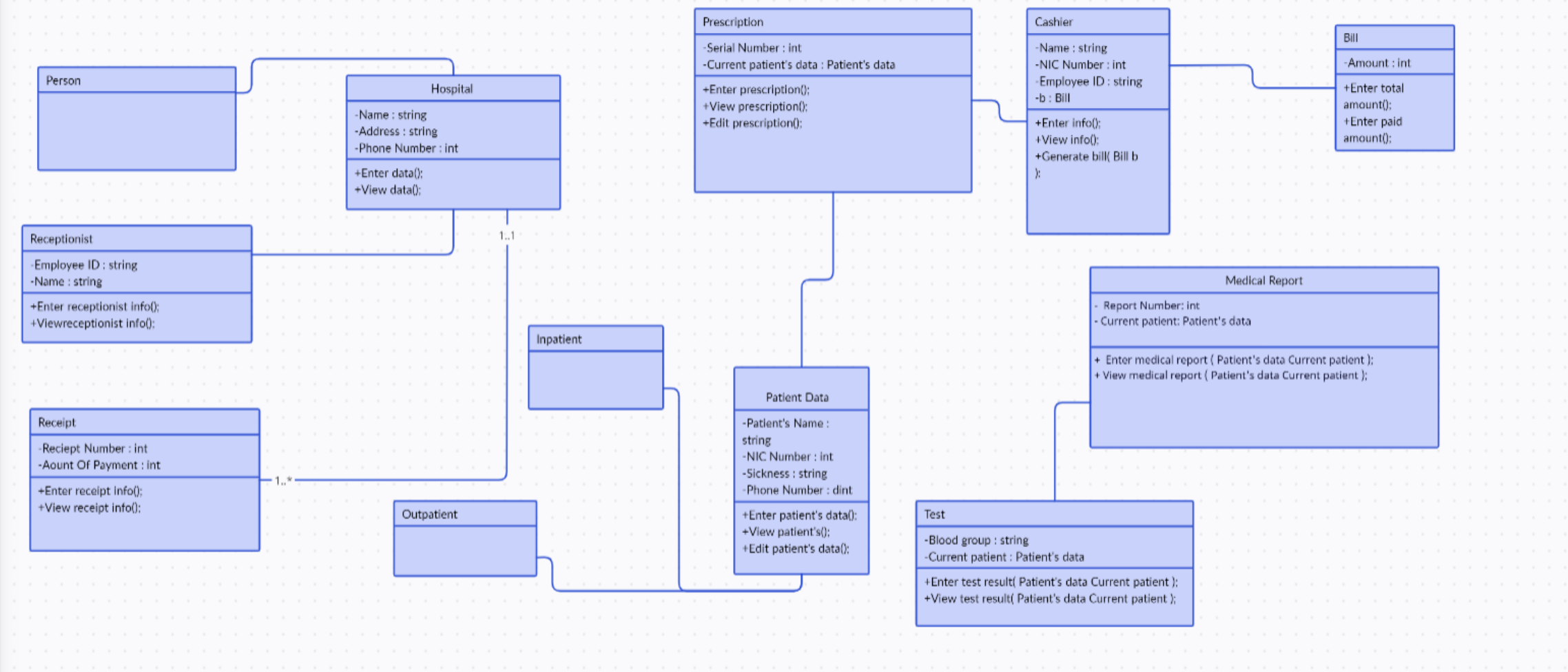
- Patient Records: The system should be able to manage patient records such as personal information, medical history, diagnosis, treatment plans, medication.
- Staff Management: The system should be able to manage employee records, such as personal information, role, availability, and schedule.
- Inventory Management: The system should be capable of managing hospital inventory such as medicine stock, medical equipment, and other supplies.
- Billing : The system should be capable of generating bills for patients.

DIFFERENT CLASSES THAT WE ARE USING TO ACHIEVE THOSE OBJECTIVES

The classes that we are using in this project are:-

1)Admin
2)Person
3)Staff
4)Patient
5)Doctor
6)Nurse
7)Receptionist
8)Store manager

WHAT THE CLASSES ARE DOING JUST A BRIEF OUTLINE



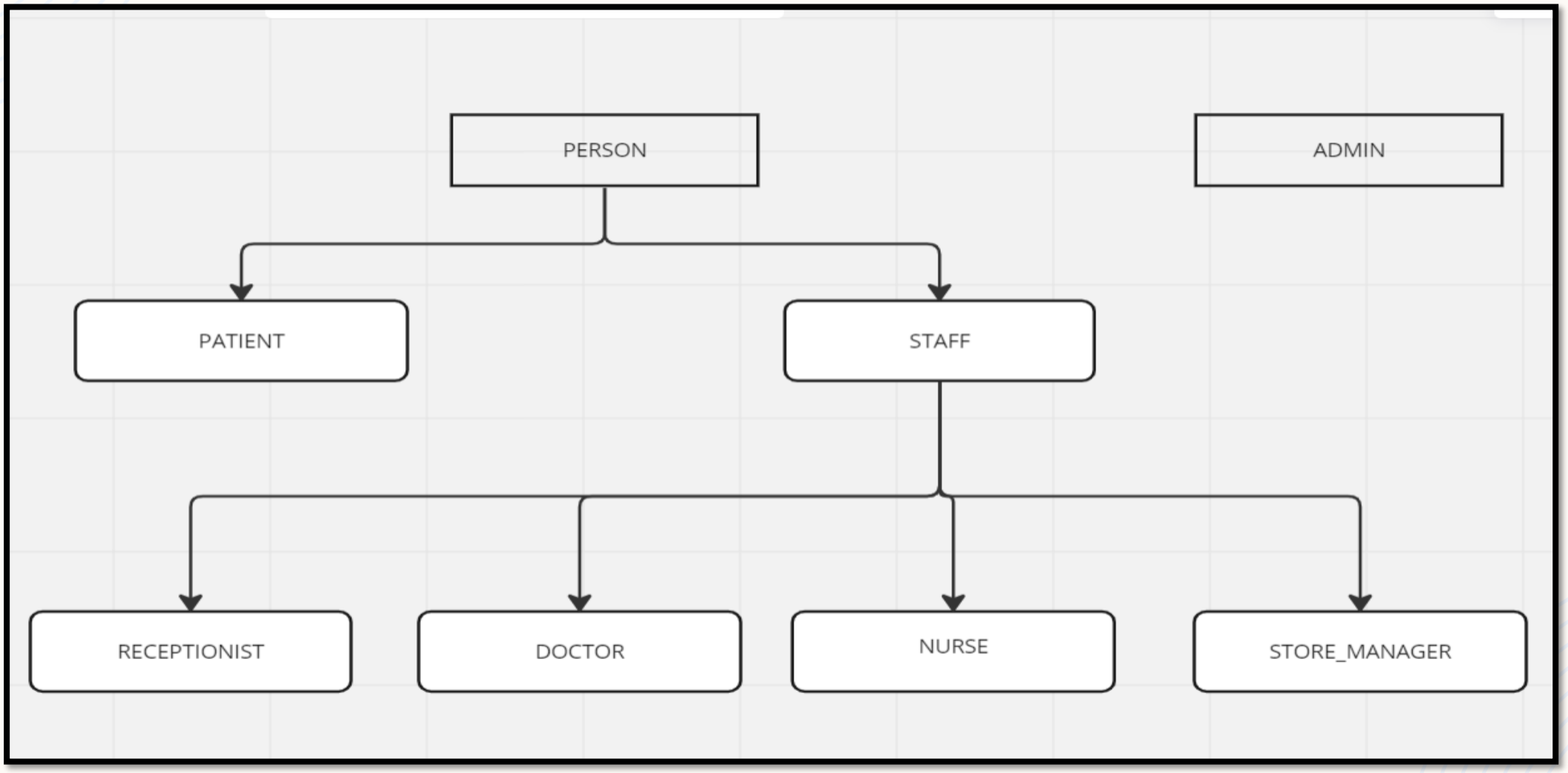
INHERITANCE DIAGRAM

The inheritance feature is very essential in developing a program or an interface.

It allows a way of creating a new class from the existing class. This feature will

help in reducing the redundancy of the code by allowing the feature of Reusability.

As we have listed out all the required classes for our project such as Admin, Person, Staff, Patient, Doctor, Nurse, Receptionist, Store manager. So we are deriving patient, staff class from person and remaining classes from the staff. So basically we are using hierarchical inheritance the more.


PERSON
ADMIN
PATIENT
STAFF
RECEPTIONIST
DOCTOR
NURSE
STORE_MANAGER

DATA HIDING PRINCIPLES

We are using necessary data hiding principles while declaring fields and member functions of the classes as private, public, protected access modes in order to ensure that data is protected from accidental or intentional modification by external code while also simplifying the interface to the data by hiding its complexity.

While deriving the one class from another class we are deriving the classes in the mode of

public or private or protected mode basing on the requirement of the function.

THANK YOU

121CS0026

121CS0061

121CS0068

121CS0071