

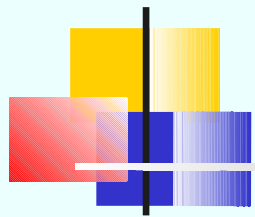


# Công nghệ phần mềm

## Pha thiết kế

---

*Giảng viên: TS. Nguyễn Mạnh Hùng*  
*Học viện Công nghệ Bưu chính Viễn thông (PTIT)*

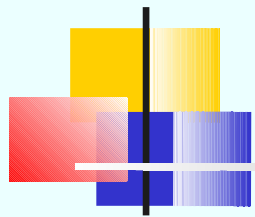


# Pha thiết kế (1)

---

Mục đích:

- Chuyển tài liệu phân tích dữ dạng đặc tả nghiệp vụ hệ thống, sang dạng có thể cài đặt và kiểm thử được



# Pha thiết kế (2)

---

Thực hiện:

- B1: Thiết kế CSDL (nếu có)
- B2: Hoàn thiện sơ đồ lớp có được trong pha phân tích → sơ đồ lớp chi tiết
- B3: Thiết kế chi tiết hoạt động bên trong của các lớp, các phương thức của lớp



# Thiết kế CSDL (1)

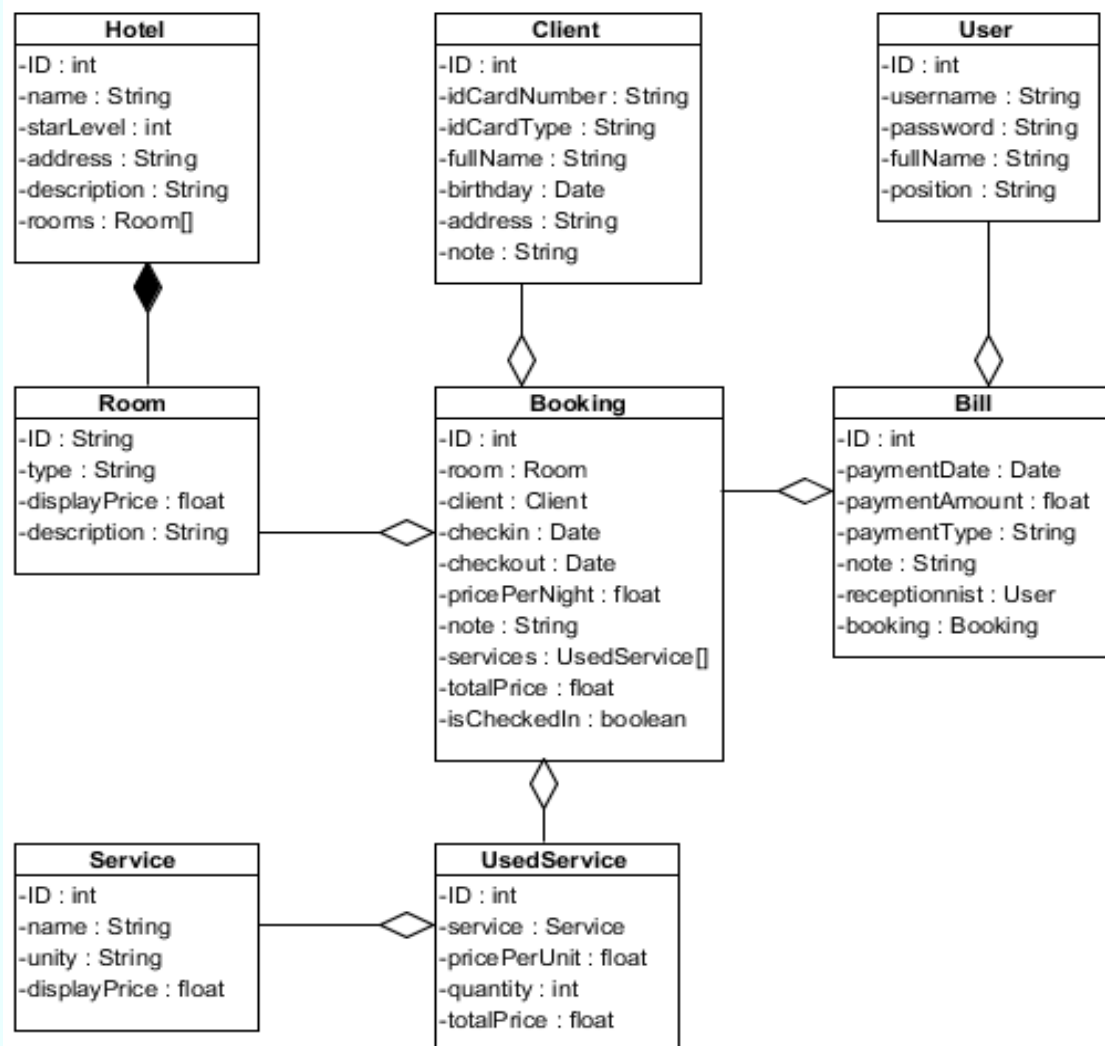
---

Xây dựng CSDL từ sơ đồ lớp thực thể của hệ thống:

- B1: Mỗi lớp thực thể đề xuất thành một bảng tương ứng trong CSDL
- B2: Với mỗi lớp thực thể, lấy các thuộc tính kiểu cơ bản (không phải kiểu lớp thực thể khác) làm thuộc tính cho bảng tương ứng với lớp thực thể đó.
- B3: Giữa hai lớp có quan hệ thành phần, liên kết, hợp thì giữa hai bảng tương ứng phải có quan hệ n-n, 1-n hoặc 1-1, tùy từng trường hợp.
- B4: Định nghĩa khóa chính và khóa ngoài tương ứng với các quan hệ giữa các bảng
- B5: Gộp bảng nếu có quan hệ 1-1, tách bảng nếu có quan hệ n-n.

# Thiết kế CSDL (2)

Ví dụ với phần mềm quản lí đặt phòng khách sạn:





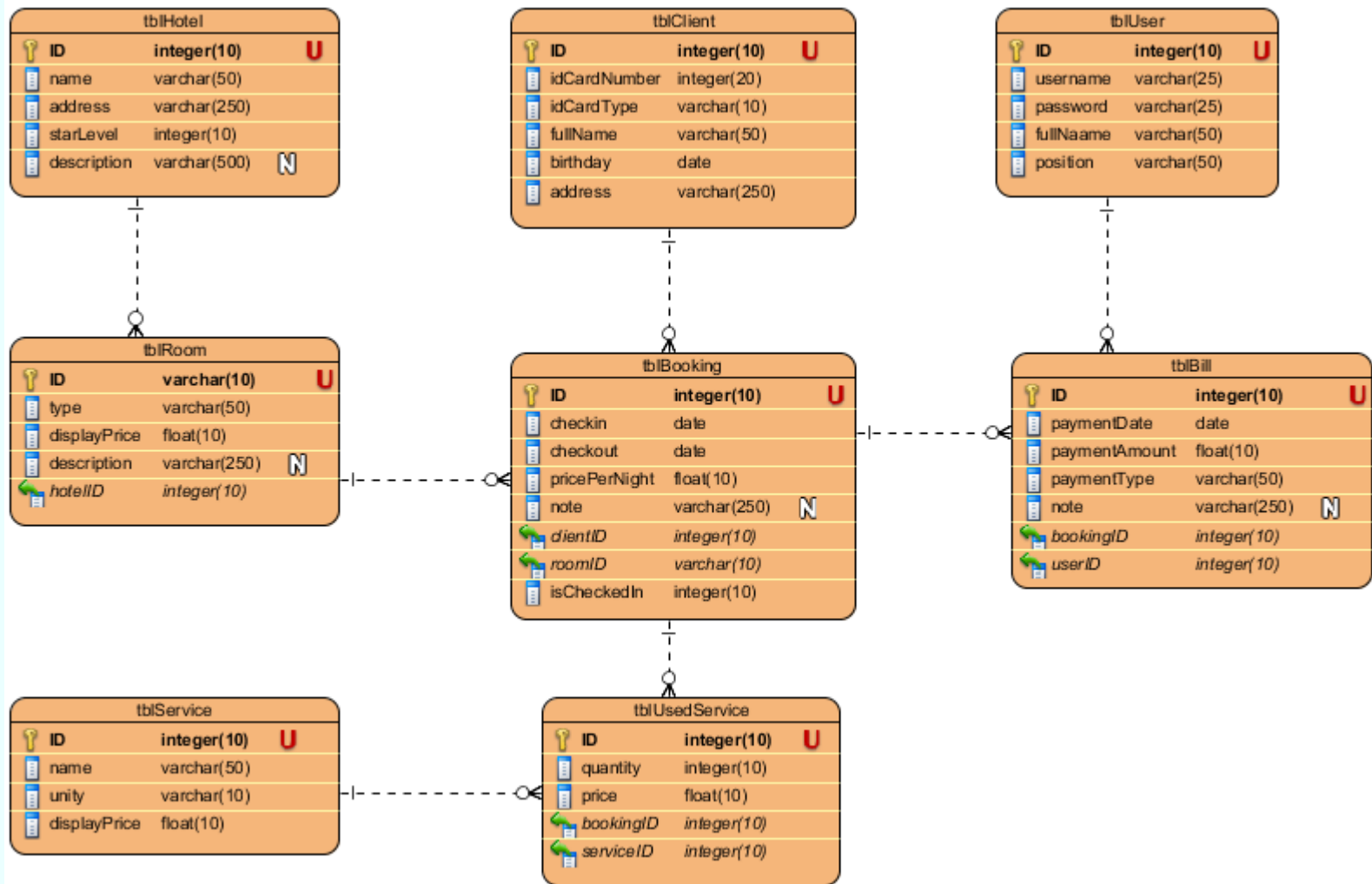
# Thiết kế CSDL (3)

---

B1: Mỗi lớp thực thể đề xuất thành một bảng tương ứng trong CSDL:

- Lớp Hotel → bảng tblHotel
- Lớp Room → bảng tblRoom
- Lớp Client → bảng tblClient
- Lớp User → bảng tblUser
- Lớp Service → bảng tblService
- Lớp UsedService → bảng tblUsedService
- Lớp Booking → bảng tblBooking
- Lớp Bill → bảng tblBill

# Thiết kế CSDL (4)





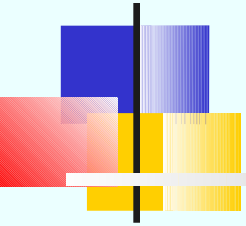
# Bài tập (1)

---

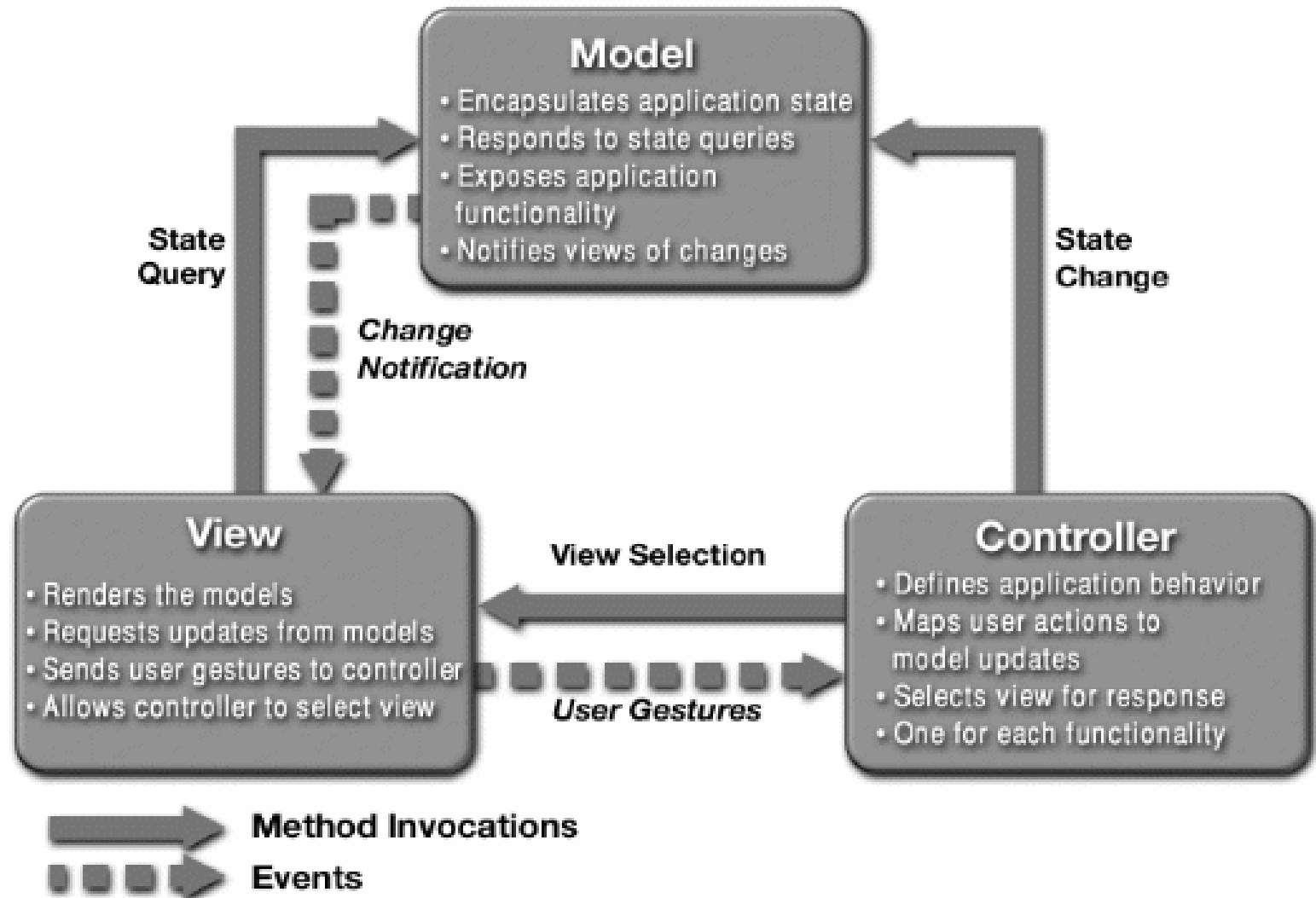
Xây dựng CSDL cho bài tập nhóm:

- Trình bày lại (đã chỉnh sửa theo kết quả pha phân tích) sơ đồ lớp thực thể của toàn hệ thống
- Trình bày sơ đồ quan hệ giữa các bảng trong CSDL sau khi áp dụng các bước chuyển đổi trong bài
- Cả nhóm nộp chung

# Thiết kế hệ thống theo mô hình MVC



# Mô hình MVC (1)





# Mô hình MVC (2)

---

M - model:

- Đóng gói dữ liệu, thông tin
- Chức năng biểu diễn, vận chuyển thông tin để trình diễn (view) và xử lý (control)



# Mô hình MVC (3)

---

C - control:

- Định nghĩa các hành vi, hoạt động, xử lý của hệ thống
- Đối chiếu hành động của user (nhận từ view), vào tập chức năng để xử lý, đồng thời chọn hành động đưa view ra để show



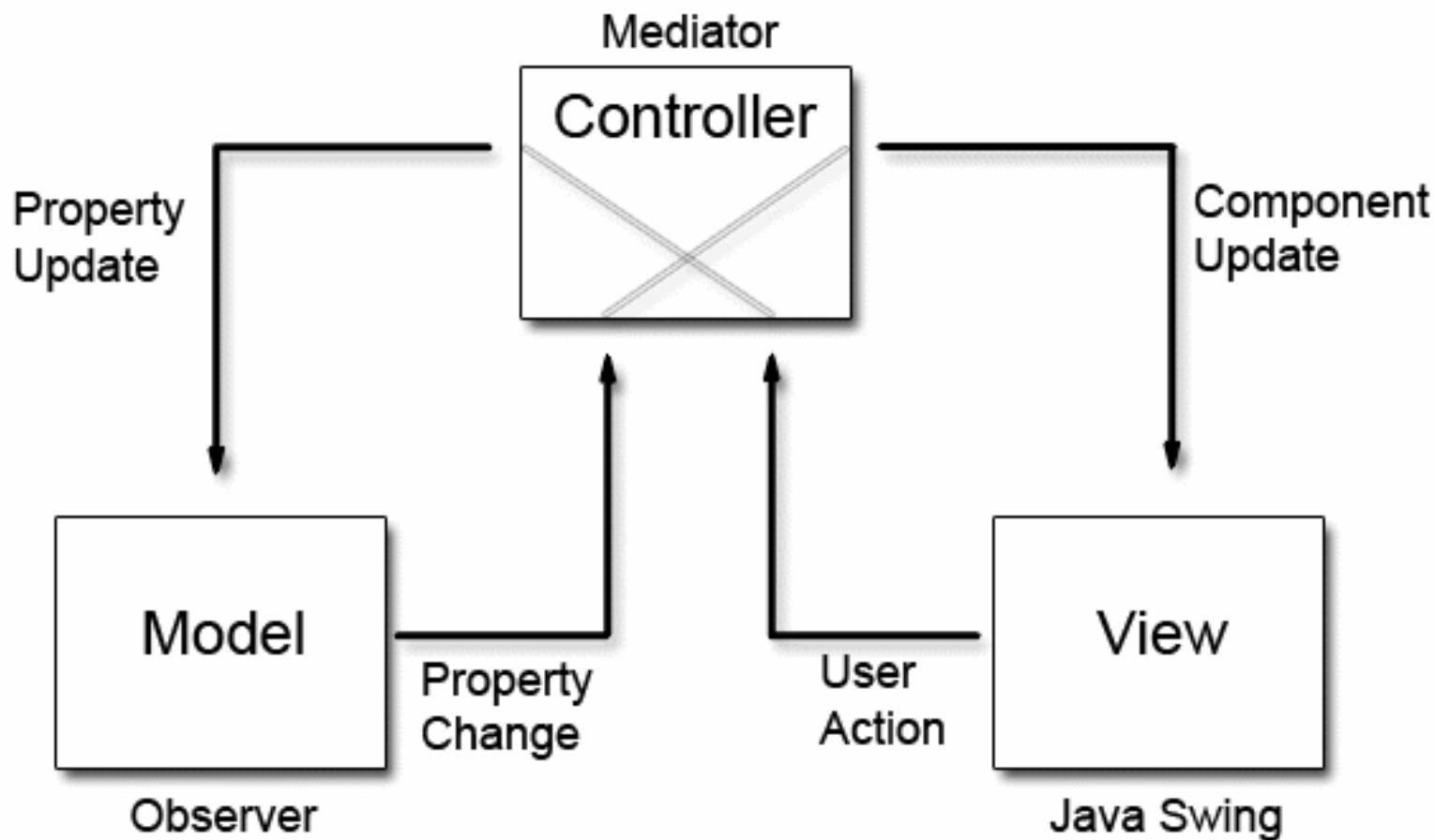
# Mô hình MVC (4)

---

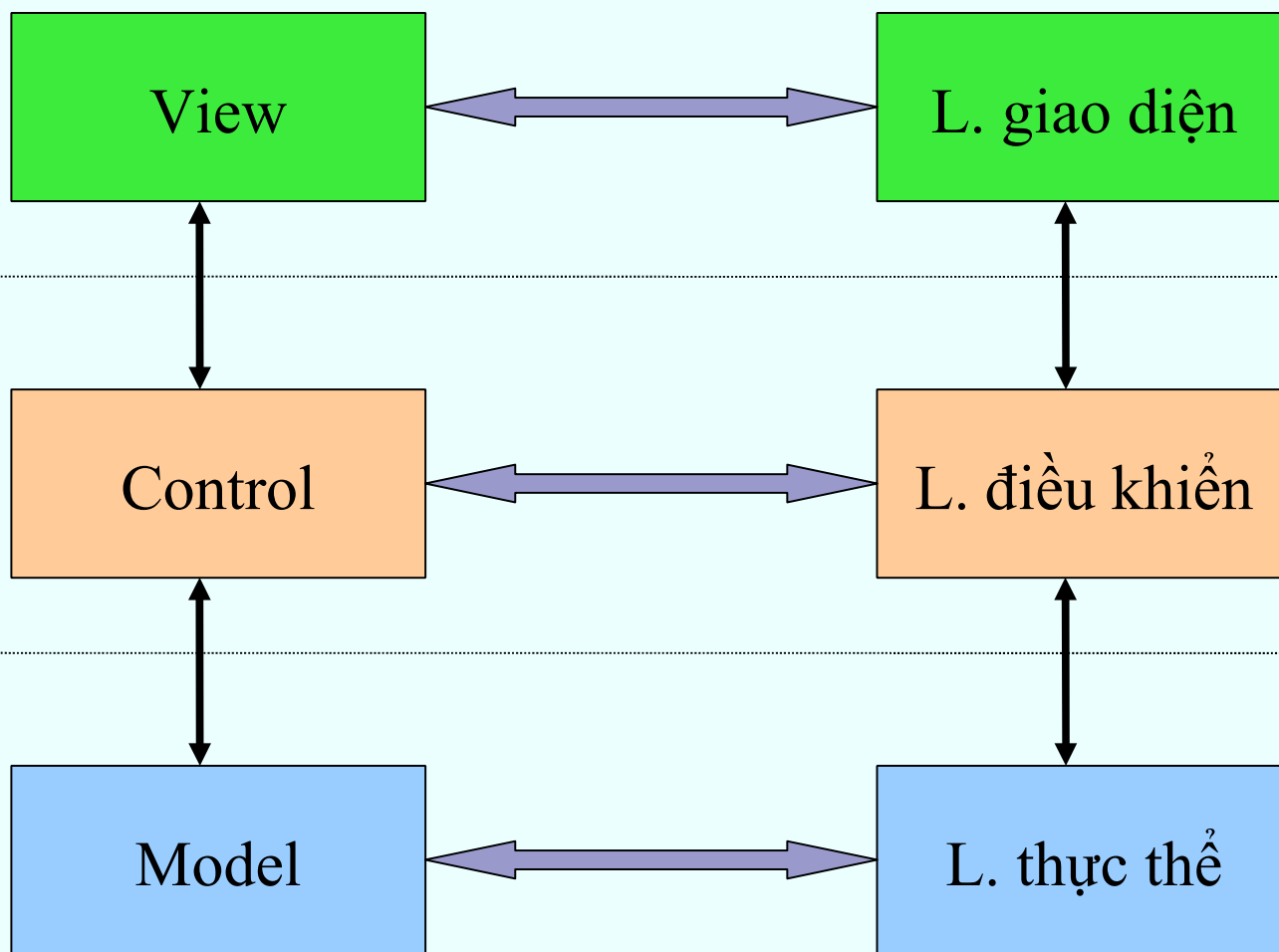
V - view:

- Giao diện với người sử dụng
- Show các kết quả xử lý của tầng control
- Thu nhận các hoạt động, yêu cầu của người sử dụng và chuyển cho tầng control xử lý

# MVC cải tiến (1)



# MVC cải tiến (2)





# Các lớp thực thể

---

- Đóng gói dữ liệu, thông tin
- Chỉ chứa các thuộc tính và các phương thức truy cập các thuộc tính (javaBean)
- Chức năng biểu diễn, vận chuyển thông tin để trình diễn (view) và xử lí (control)



# Các lớp điều khiển

---

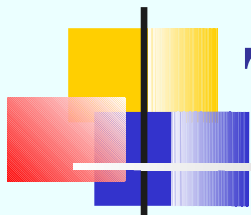
- Cập nhật thông tin vào DB (thông tin chứa trong các thực thể)
- Thực hiện các tính toán, xử lý trung gian
- Đối chiếu hành động của user (nhận từ view), vào tập chức năng để xử lý, đồng thời chọn hành động đưa view ra để show



# Các lớp giao diện

---

- Các frame, cửa sổ của ứng dụng (javaSwing)
- Các trang giao diện web: html, jsp
- Các bảng, mẫu biểu, báo cáo in ra



# Thiết kế hệ thống theo MVC

---

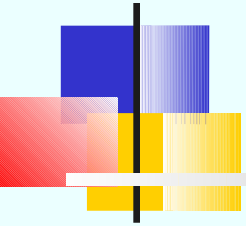
Có thể áp dụng một số dạng mô hình MVC phổ biến:

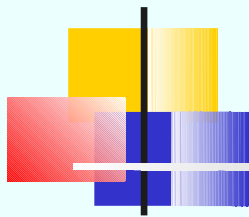
- Mô hình MVC dùng thực thể thuần (cổ điển)
- Mô hình MVC dùng bean
- Mô hình MVC cải tiến (hiện đại)

Lưu ý:

Với bài tập lớn, các nhóm nên chọn thiết kế theo 1 trong 3 dạng kiến trúc trên.

# Thiết kế theo mô hình MVC với thực thể thuần





# MVC với thực thể thuần (1)

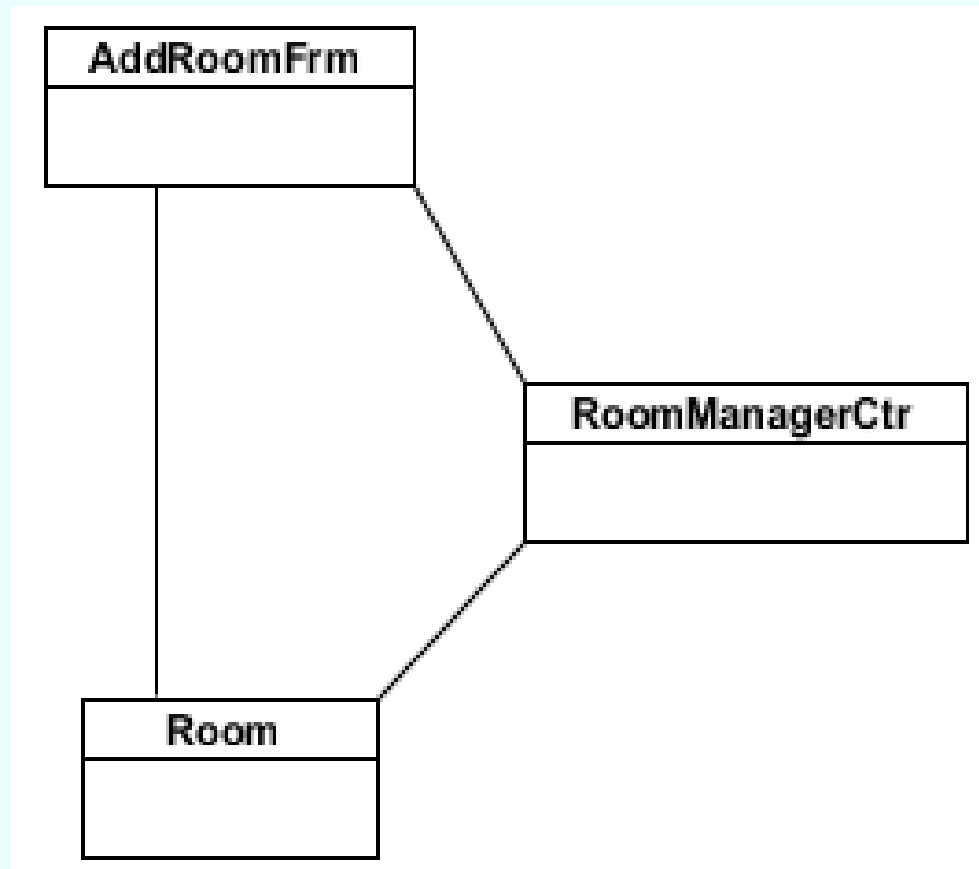
---

## Đặc trưng:

- Lớp thực thể chỉ chứa các thuộc tính và các phương thức get/set cho mỗi thuộc tính (còn gọi là các lớp thực thể thuần)
- Các thao tác liên quan đến CSDL đều đặt trong lớp điều khiển (dạng lớp DAO – Data Access Object)

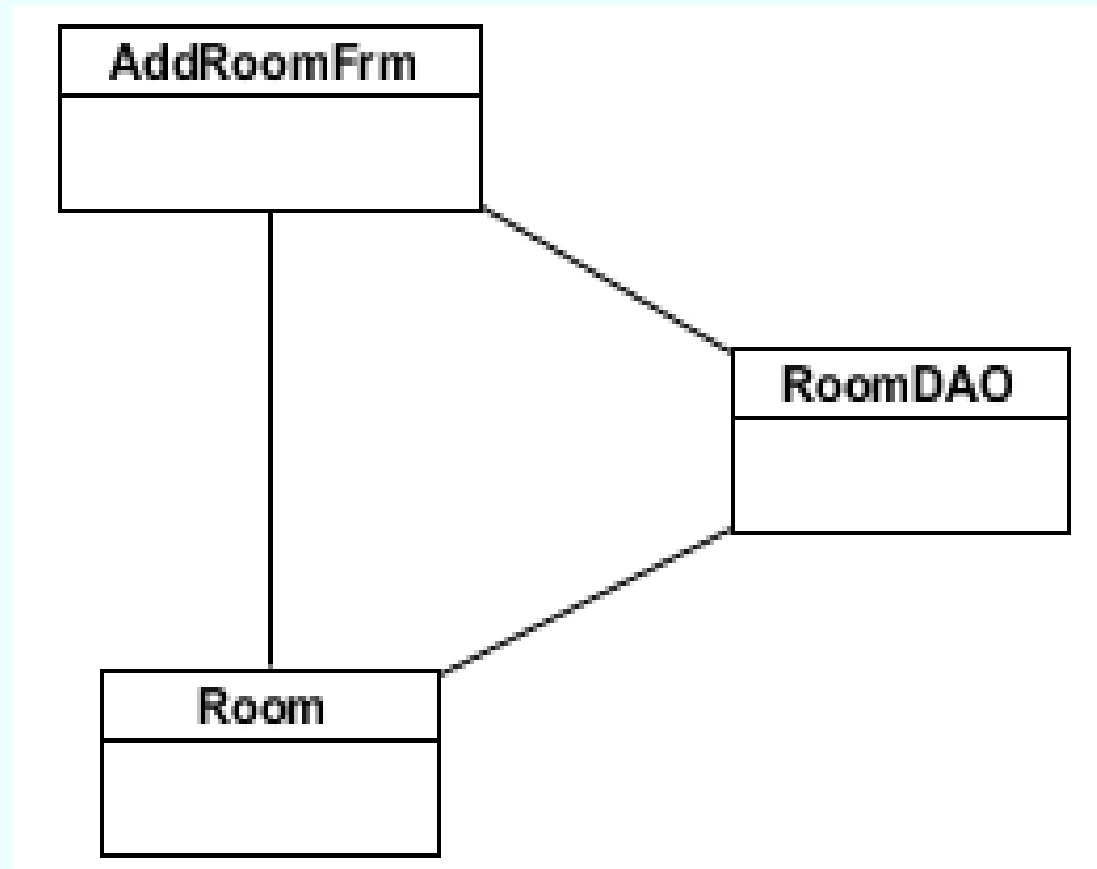
# MVC với thực thể thuần (2)

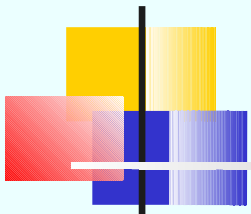
Ví dụ modul quản lí phòng của Manager, sơ đồ lớp cuối pha phân tích của chức năng thêm phòng:



# MVC với thực thể thuần (3)

Ví dụ chức năng thêm phòng của Manager, sơ đồ lớp theo MVC dùng thực thể thuần:



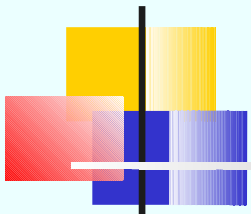


# Hoàn thiện sơ đồ lớp (1)

---

Thực hiện:

- Định nghĩa kiểu thuộc tính cho lớp
- Định nghĩa khuôn mẫu các phương thức cho lớp

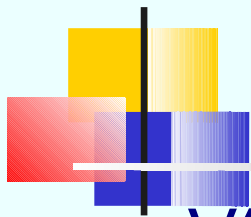


# Hoàn thiện sơ đồ lớp (2)

---

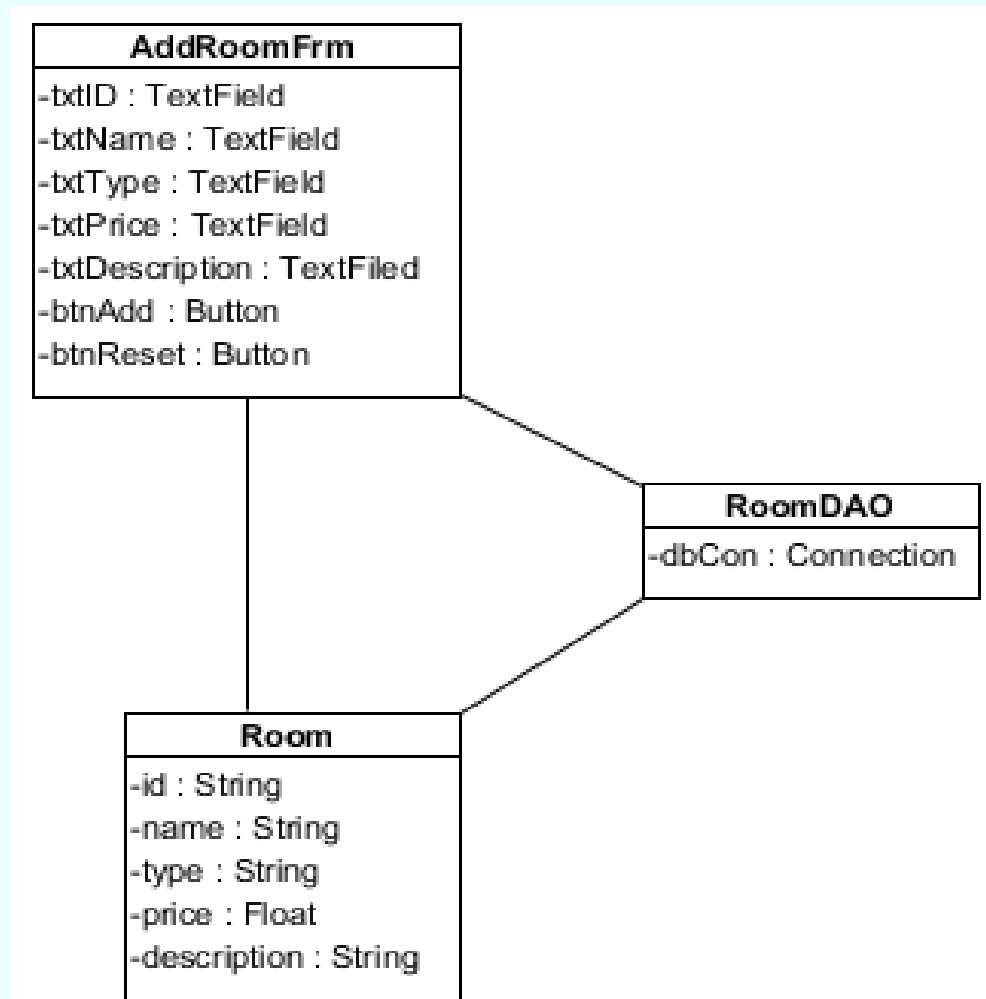
Định nghĩa kiểu thuộc tính cho lớp:

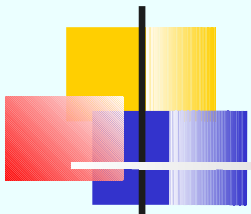
- Tên thuộc tính đã xác định trong pha phân tích
- Chọn kiểu dữ liệu cụ thể cho từng thuộc tính dựa vào giới hạn lưu trữ của thuộc tính
- Điền thuộc tính (tên:kiểu) vào sơ đồ lớp



# Hoàn thiện sơ đồ lớp (3)

Ví dụ với chức năng thêm phòng:



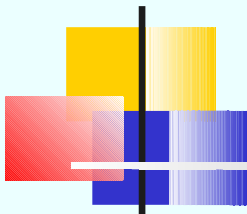


# Hoàn thiện sơ đồ lớp (4)

---

Định nghĩa khuôn mẫu phương thức cho lớp:

- Dùng thẻ CRC để xác định phương thức nào nên gán cho lớp nào
- Định nghĩa khuôn mẫu cho từng phương thức

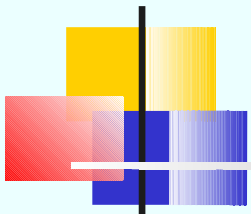


# Hoàn thiện sơ đồ lớp (5)

---

## Gán phương thức cho lớp:

- **Nguyên lí A: Che giấu thông tin.** Các thuộc tính của lớp phải để dạng private → cần các phương thức **get/set** tương ứng cho phép các đối tượng khác truy nhập vào các thuộc tính này
- Áp dụng cho các lớp thực thể: các thuộc tính để private, mỗi thuộc tính có một cặp phương thức get/set tương ứng



# Hoàn thiện sơ đồ lớp (6)

---

Ví dụ với lớp thực thể Room:

- Các thuộc tính để chế độ private (có dấu “-” đầu tên thuộc tính)
- Mỗi thuộc tính có một cặp phương thức get/set tương ứng: ví dụ thuộc tính name có các phương thức:

```
public void setName(String name)
```

```
public String getName()
```

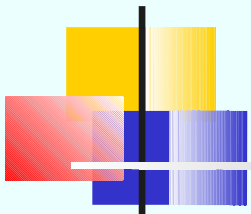


# Hoàn thiện sơ đồ lớp (7)

---

Gán phương thức cho lớp (tt):

- **Nguyên lí B:** Nếu có nhiều đối tượng **X** gọi đến một hành động **k** của đối tượng **Y**, thì phương thức để thực hiện hành động **k** nên gán cho lớp của đối tượng **Y**, mà không nên gán cho lớp của đối tượng **X**

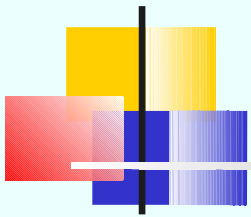


# Hoàn thiện sơ đồ lớp (8)

---

Gán phương thức cho lớp (tt):

- **Nguyên lí C:** Thiết kế hướng trách nhiệm. Nếu một hành động mà không thể gán thành phương thức cho lớp khác, thì lớp của đối tượng cần thực hiện hành động đó phải chứa phương thức tương ứng hành động đó



# Hoàn thiện sơ đồ lớp (9)

Định nghĩa khuôn mẫu các phương thức:

- Kiểu dữ liệu trả về
- Số lượng, thứ tự và kiểu dữ liệu truyền vào

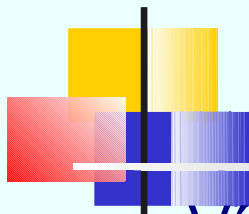
Ví dụ:

Không nên định nghĩa phương thức lưu thông tin một **Room** với các tham số là các thuộc tính:

```
public void saveRoom(int id, string name...)
```

Mà nên truyền vào là một kiểu đối tượng của lớp **Room** đã đóng gói:

```
public void saveRoom(Room room)
```

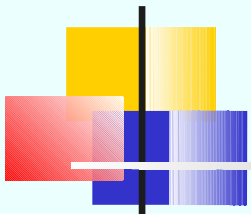


# Hoàn thiện sơ đồ lớp (10)

Ví dụ thẻ CRC của lớp RoomDAO và AddRoomFrm cho chức năng thêm phòng:

RoomDAO	
Responsibilities:	
Name	Collaborator
Yêu cầu lớp AddRoomFrm hiển thị giao diện nhập thông tin phòng	AddRoomFrm
Lưu thông tin phòng vào CSDL	
Yêu cầu lớp AddRoomFrm hiển thị thông báo thêm thành công	AddRoomFrm

AddRoomFrm	
Responsibilities:	
Name	Collaborator
Hiển thị form nhập thông tin phòng	
Yêu cầu lớp Room đóng gói thông tin nhập trên form thành đối tượng	Room
Yêu cầu lớp RoomDAO lưu thông tin phòng vào CSDL	RoomDAO
Xử lý sự kiện khi nút Add bị click	
Xử lý sự kiện khi nút Reset bị click	

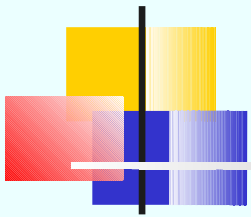


# Hoàn thiện sơ đồ lớp (11)

---

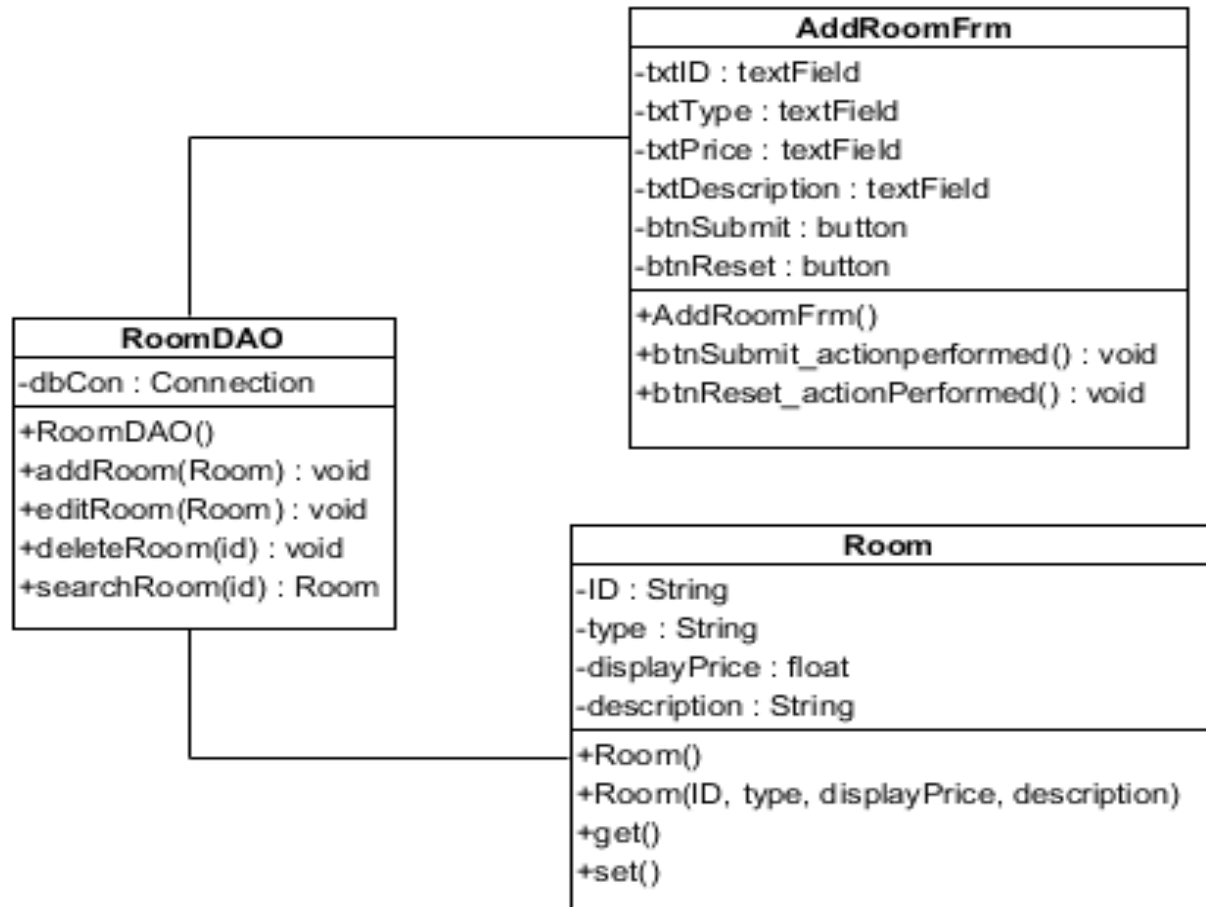
Áp dụng nguyên lí B và C:

- Lớp AddRoomFrm phải có các phương thức: hiển thị form (hàm khởi tạo), xử lý sự kiện nút Add và nút Reset bị click, thông báo thành công
- Lớp RoomDAO phải có phương thức lưu thông tin phòng vào CSDL
- Lớp Room phải có các phương thức đóng gói thông tin đối tượng (hàm khởi tạo hoặc các phương thức set)



# Hoàn thiện sơ đồ lớp (12)

Kết quả thu được sơ đồ lớp như sau:





# Sơ đồ tuần tự pha thiết kế (1)

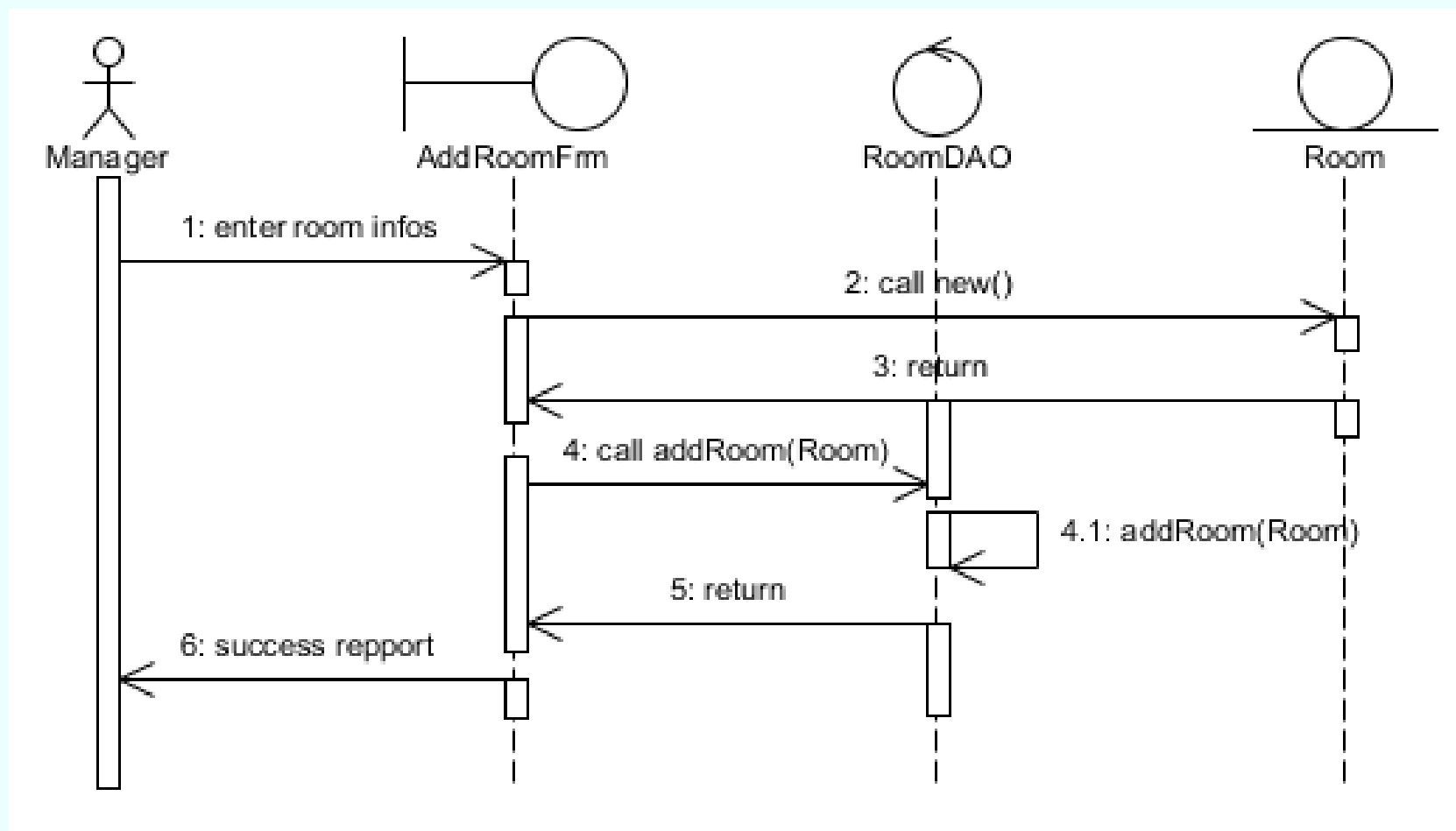
---

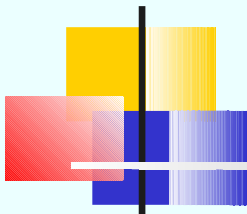
Đặc trưng:

- Tên các lớp theo đúng sơ đồ lớp đã thiết kế
- Nhãn các mũi tên phải là tên các phương thức của các lớp trong sơ đồ lớp đã thiết kế

# Sơ đồ tuần tự pha thiết kế (2)

Ví dụ chức năng thêm phòng của Manager:





# Bài tập

---

Thiết kế tương tự cho các chức năng và modul:

- Sửa thông tin phòng
- Xóa thông tin phòng
- Khách hàng đặt chỗ tại quầy với nhân viên tiếp tân
- Khách hàng checkin tại quầy với nhân viên tiếp tân
- Khách hàng trả phòng và thanh toán tại quầy với nhân viên tiếp tân



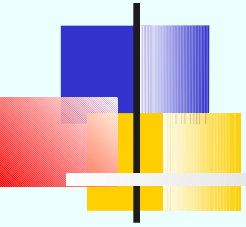
# Bài tập về nhà

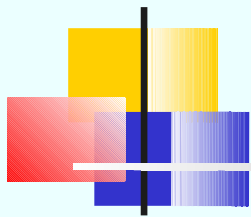
---

Với mỗi modul cá nhân:

- Vẽ lại sơ đồ UC chi tiết của hệ thống và của modul
- Vẽ lại sơ đồ các lớp sau pha phân tích
- Định nghĩa các thuộc tính và kiểu thuộc tính của mỗi lớp
- Dùng kĩ thuật thẻ CRC và 3 nguyên lí thiết kế phương thức để gán các phương thức cho các lớp
- Định nghĩa khuôn mẫu cho từng phương thức
- Điền tất cả vào sơ đồ lớp để thu được sơ đồ lớp chi tiết (theo mô hình MVC dùng thực thể thuần)
- Vẽ lại sơ đồ tuần tự sau pha thiết kế

# Thiết kế theo mô hình MVC với thực thể bean





# MVC với thực thể bean (1)

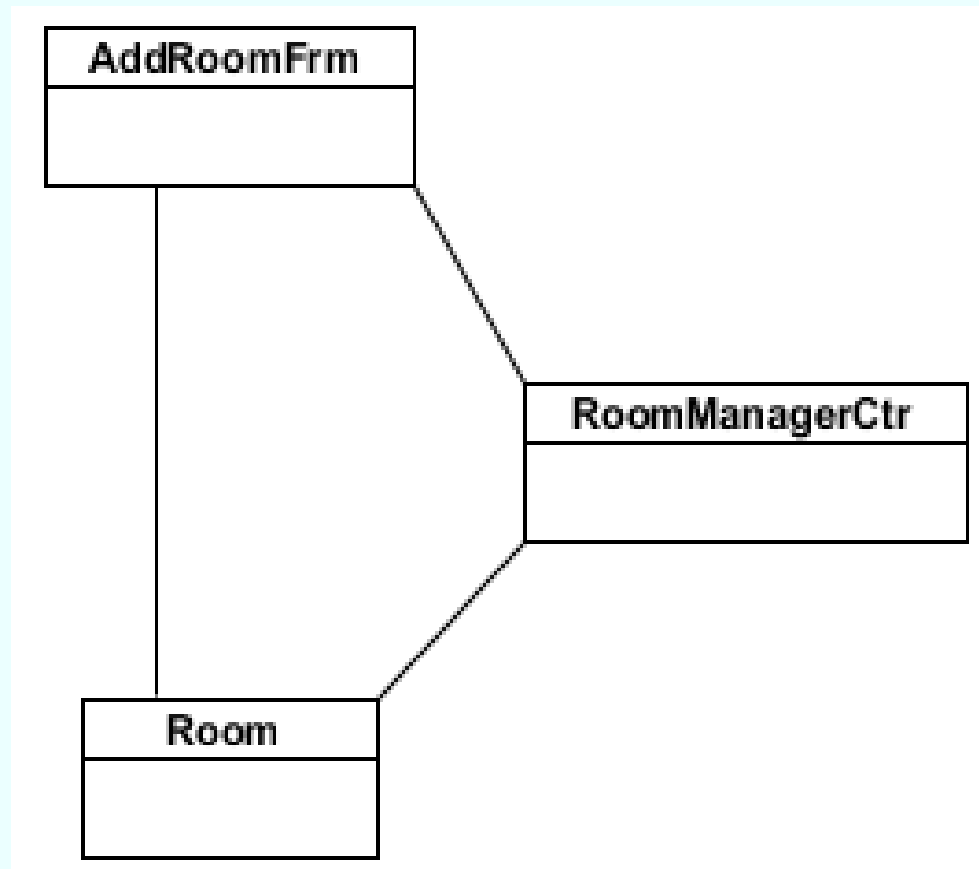
---

## Đặc trưng:

- Lớp thực thể chứa các thuộc tính và các phương thức get/set cho mỗi thuộc tính, và
- Các thao tác liên quan đến CSDL mà liên quan đến lớp thực thể nào thì đều đặt trong lớp điều khiển đó
- Các lớp thực thể kiểu này được gọi là lớp bean
- Trong nhiều trường hợp, không còn cần đến lớp điều khiển nữa vì lớp bean đã kiêm luôn vai trò của lớp điều khiển

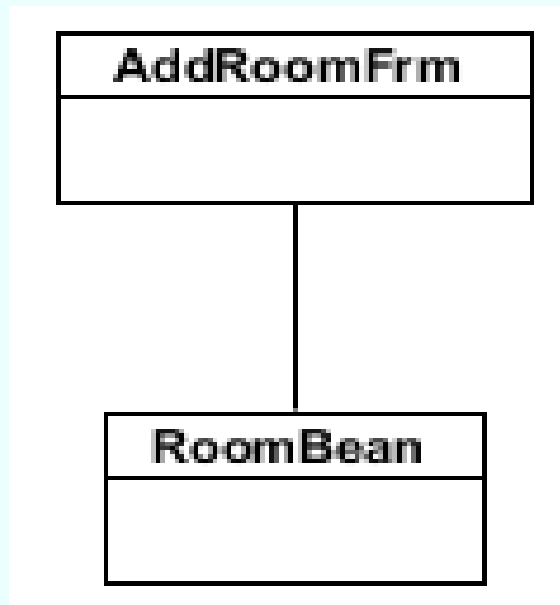
# MVC với thực thể bean (2)

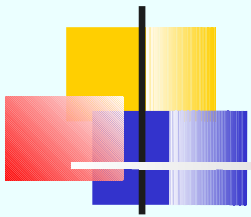
Ví dụ modul quản lí phòng của Manager, sơ đồ lớp cuối pha phân tích của chức năng thêm phòng:



# MVC với thực thể bean (3)

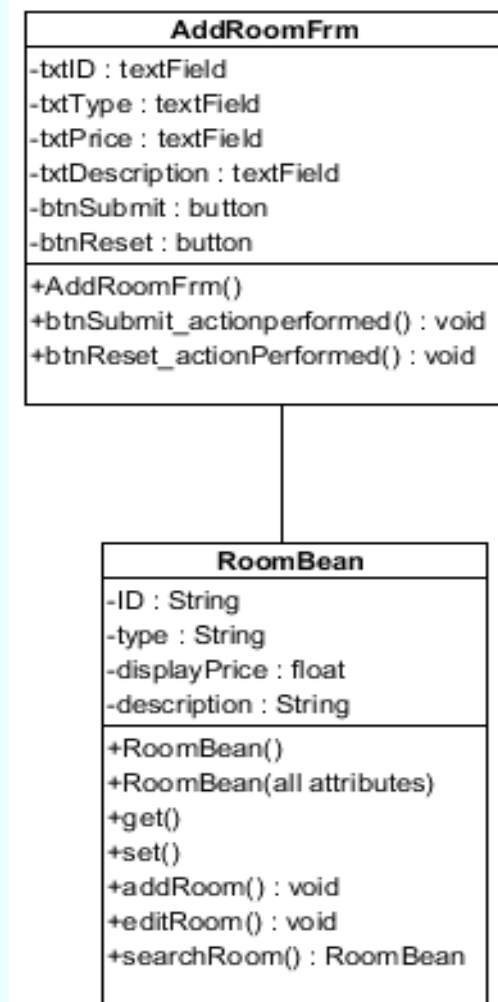
Ví dụ chức năng thêm phòng của Manager, sơ đồ lớp theo MVC dùng thực thể bean:





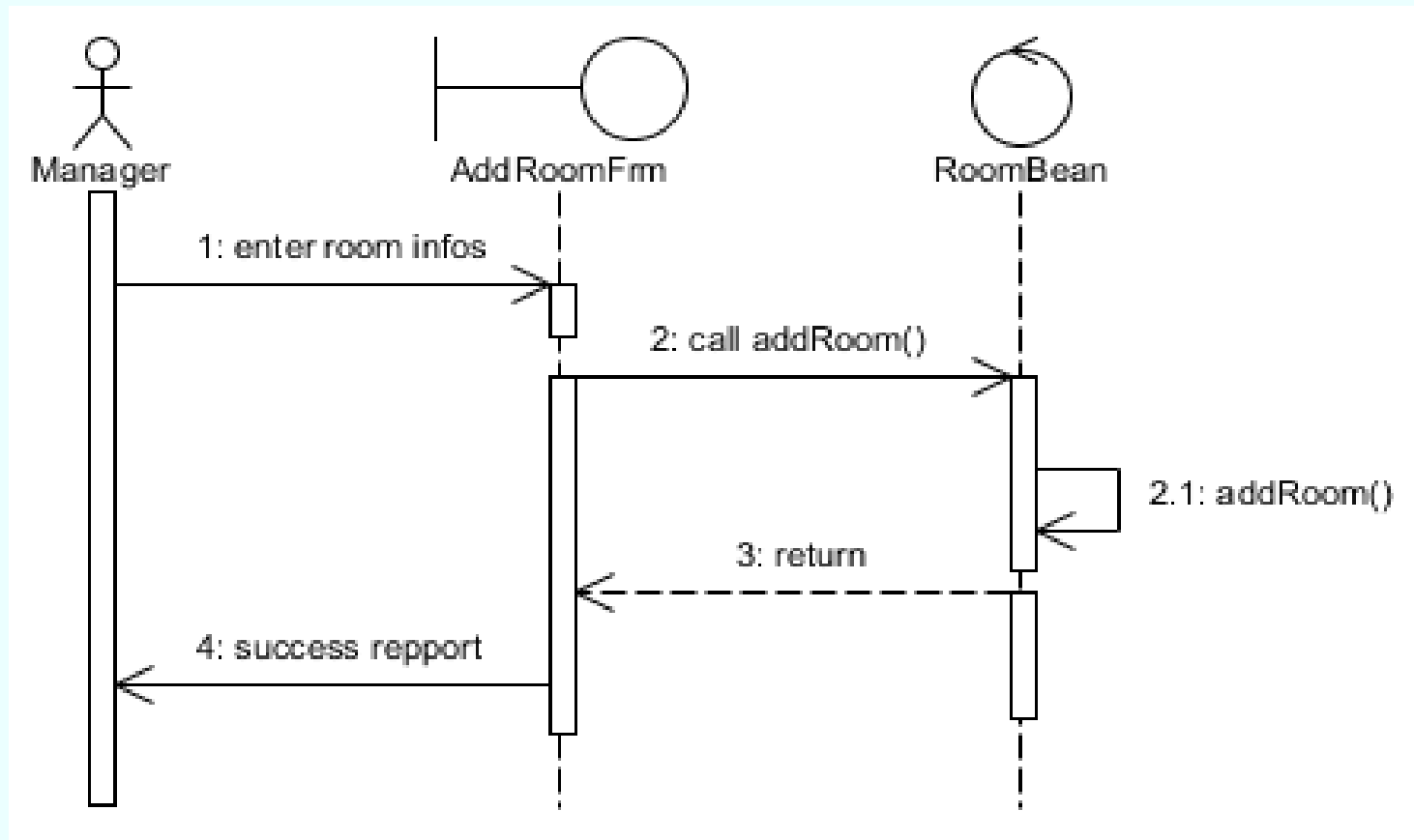
# Hoàn thiện sơ đồ lớp

Kết quả thu được sơ đồ lớp sau khi áp dụng các nguyên lí A, B, và C như sau:

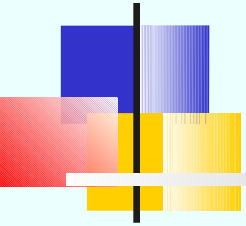


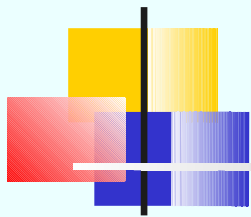
# Sơ đồ tuần tự pha thiết kế

Sơ đồ tuần tự cho cách thiết kế dùng bean:



# Thiết kế theo mô hình MVC cải tiến (hiện đại)





# MVC cải tiến (1)

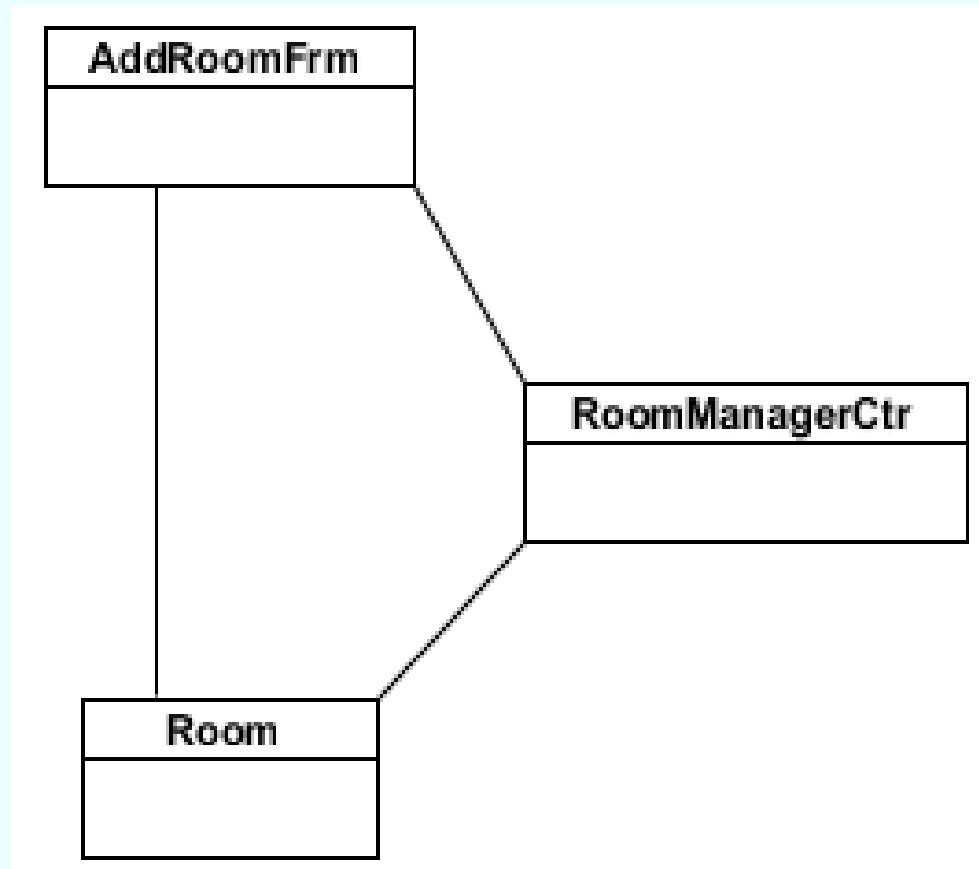
---

## Đặc trưng:

- Lớp thực thể thuần
- Các thao tác liên quan đến CSDL đều đặt trong lớp điều khiển
- Các lớp điều khiển giành quyền điều khiển toàn bộ các lớp view và control

# MVC cải tiến (2)

Ví dụ modul quản lí phòng của Manager, sơ đồ lớp cuối pha phân tích của chức năng thêm phòng:



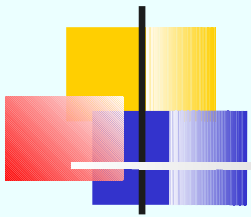


# MVC cải tiến (3)

---

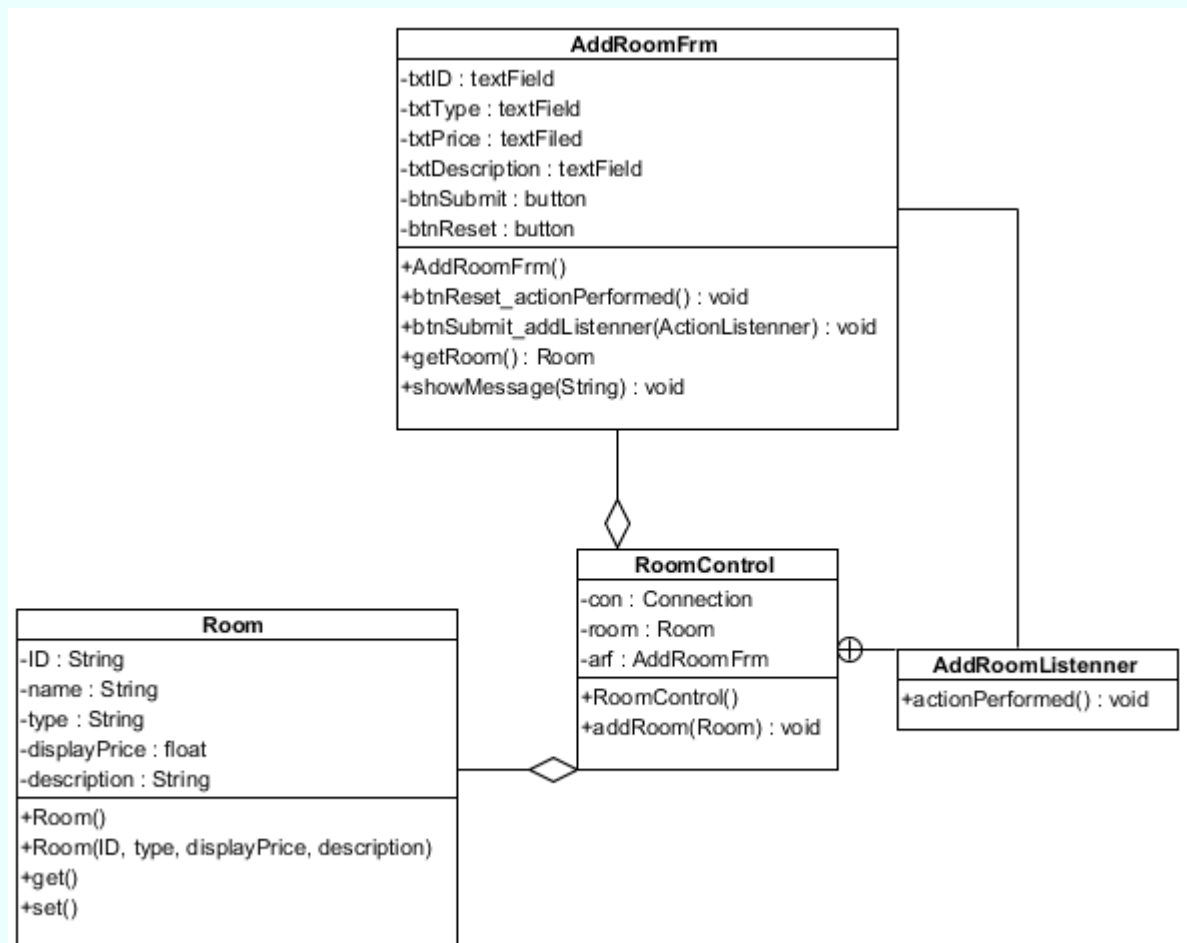
Ví dụ chức năng thêm phòng của Manager, sơ đồ lớp theo MVC cải tiến:

- Khi nút Add trên form của lớp biên AddRoomFrm bị click thì lớp biên này không được xử lí gì mà phải truyền sự kiện này xuống cho lớp điều khiển
- Để làm được việc này, trên lớp biên có phương thức truyền quyền điều khiển cho lớp điều khiển, và
- Trong lớp điều khiển có một lớp nội tại (inner class) dùng để nhận sự kiện do lớp biên truyền xuống để xử lí



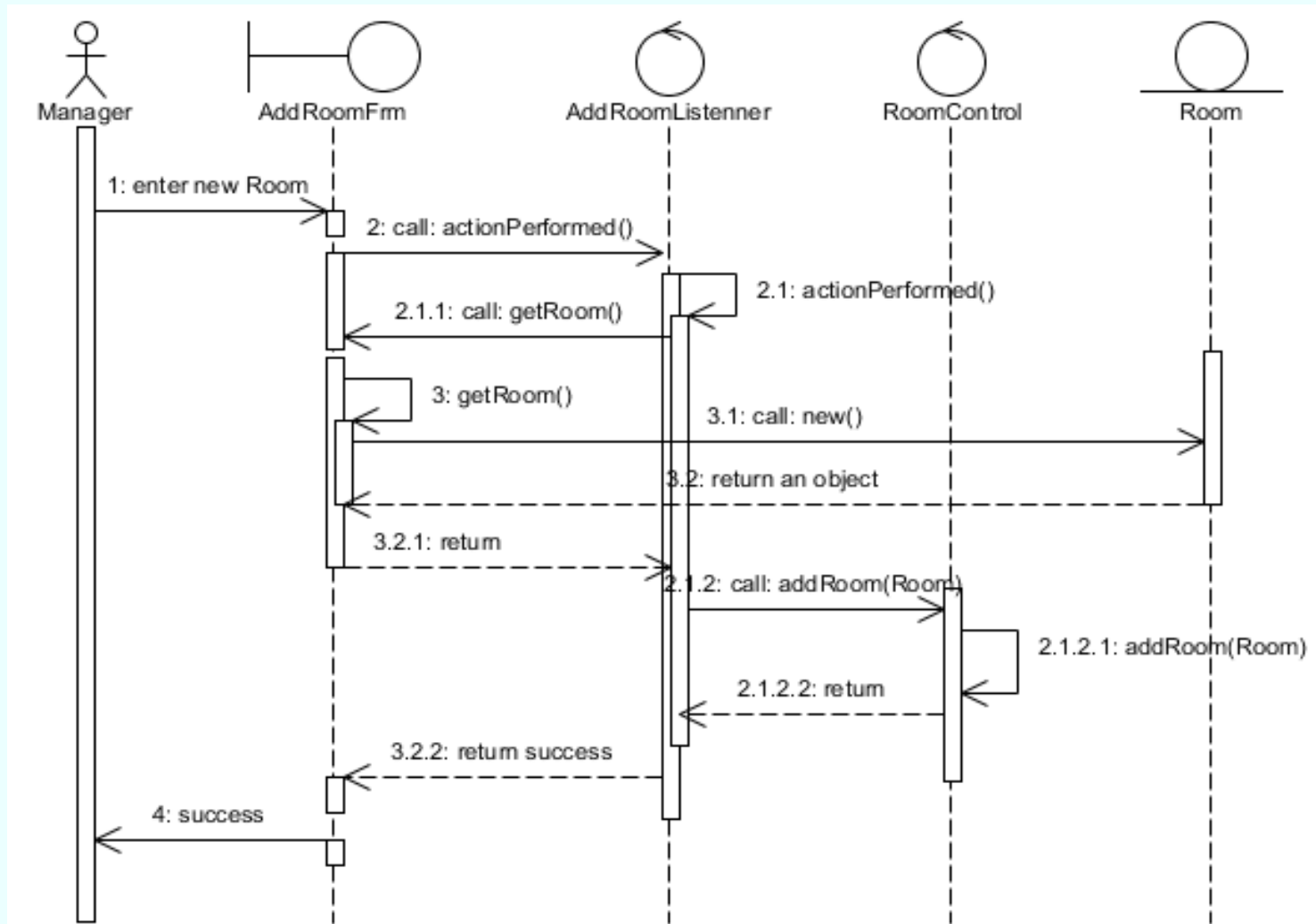
# Hoàn thiện sơ đồ lớp

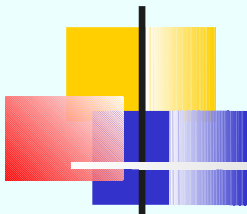
Kết quả thu được sơ đồ lớp sau khi áp dụng các nguyên lý A, B, và C như sau:



# Sơ đồ tuần tự pha thiết kế

Sơ đồ tuần tự cho cách thiết kế dùng MVC cải tiến:





# Bài tập

---

Thiết kế tương tự (dung MVC với thực thể bean và dùng MVC cải tiến) cho các chức năng và modul:

- Sửa thông tin phòng
- Xóa thông tin phòng
- Khách hàng đặt chỗ tại quầy với nhân viên tiếp tân
- Khách hàng checkin tại quầy với nhân viên tiếp tân
- Khách hàng trả phòng và thanh toán tại quầy với nhân viên tiếp tân



# Bài tập về nhà

---

Với mỗi modul cá nhân:

- Vẽ lại sơ đồ UC chi tiết của hệ thống và của modul
- Vẽ lại sơ đồ các lớp sau pha phân tích
- Thiết kế hệ thống theo mô hình MVC dùng thực thể bean và theo mô hình MVC cải tiến
- Vẽ lại sơ đồ tuần tự sau pha thiết kế (cho mỗi loại)



# Questions?

---



# Công nghệ phần mềm

## Pha cài đặt

---

*Giảng viên: TS. Nguyễn Mạnh Hùng*  
*Học viện Công nghệ Bưu chính Viễn thông (PTIT)*