**BÁO CÁO PROJECT**

# Thế Giới Phim Trên Đầu Ngón Tay

|  |  |
| --- | --- |
| **Học kỳ:** | Học Kỳ 1 |
| **Lớp:** | F07 |
| **Nhóm:** | Group 2 |
| **Giảng viên hướng dẫn:** | Đào Văn Đức |
| **Các thành viên:** | NDE17060 - Trần Mạnh Đạt  NDE17071 - Tống Văn Phòng  NDE17069 - Cao Xuân Huy |
|  |  |

# 

# Mục Lục

[**Thế Giới Phim Trên Đầu Ngón Tay**](#_gjdgxs) **1**

[**Mục Lục**](#_iipoy1hzt3vi) **2**

[**Giới thiệu dự án**](#_xxaj5sqv0rrp) **4**

[**Phân tích yêu cầu hệ thống**](#_3znysh7) **5**

[1. Use Case](#_s5aofq49jq1p) 5

[Use Case Diagram](#_3ppbrufk0gsu) 5

[Mô tả Use Case](#_ly5bby2p0ins) 5

[Mô tả chức năng đăng nhập](#_93ih2xszifjs) 5

[- Mô tả chức năng xem các phim đang chiếu.](#_hzvmd7fsg0um) 7

[Mô tả chức năng đặt vé](#_1cj4wqoaog2m) 9

[2. Sơ đồ hoạt động (Activity Diagram)](#_83ve21jxxy55) 11

[Đăng nhập (Login)](#_69g9sm67elva) 11

[Xem các phim đang chiếu (Movie Now Showing)](#_6gi5mqhdh5ps) 12

[Đặt vé ( Booking Ticket )](#_82cpq95talop) 13

[3. Sequence Diagram](#_60xri4oub0ge) 14

[Đăng nhập (Login)](#_mj0fu5sue7zo) 14

[Phim đang chiếu. ( Now Showing )](#_qksoq69hvxzs) 15

[Đặt vé ( Booking Ticket )](#_x943k98xw2g7) 16

[**Thiết kế chi tiết**](#_yfvwqzhcrwzn)

[Thiết kế giao diện (GUI)](#_nn4mka1l4udz) 17

[Mô tả chức năng giao diện](#_kvyqpxphfj99) 17

[Thiết kế giao diện](#_untd3fgg64gi) 17

[Giao diện đăng nhập](#_cmvhms6nwku6) 17

[Giao diện rạp chiếu phim.](#_rohvzv17sbvn) 18

[Giao diện danh sách phim đang chiếu](#_q3tkx888natj) 18

[Giao diện Đặt vé](#_riqjvk4wipdy) 20

[Giao diện đặt ghế](#_adnvmtz4lzxs) 22

[Thông tin vé đặt trước](#_iall9m1dkx56) 24

[Giao diện quản lý thông tin cá nhân](#_6svff91til3x) 25

[Thiết kế mã (Class Diagram)](#_6yjdmjck5lep) 27

[Class Diagram](#_21rxpfwmci) 27

[MÔ HÌNH 3 LỚP (3 – LAYER)](#_6uiuivwpmi4a) 28

[Thiết kế CSDL (Database)](#_q8zoxh1ngvpt) 29

[Entity Relationship Diagram](#_akaa0zshe1og) 29

[Thiết kế CSDL chi tiết](#_jsxwyfl59idj) 29

[**Kiểm thử**](#_cu3q6qwdx616)32

[4.1. Kiểm thử tầng DAL](#_4qmrzl5l4gxk) 32

[Class CustomerDAL](#_6znk526luxcc) 32

[Kiểm thử Class MovieDAL](#_jt3k7dmd3z9s) 33

[- Kiểm thử Class Room](#_g6ibw7tj96hn) 35

[- Kiểm thử Class Schedule](#_18cwbac4arul) 36

[Kiểm thử Class Reservation](#_86sajoo8k0sx) 40

[4.2 Kiểm thử tầng BL](#_vmctwbfvf35p) 41

[**Phân công công việc cho từng thành viên trong nhóm**](#_2wwfmnw40uh0)42

[**Hướng dẫn cài đặt**](#_1t3h5sf) **46**

[Biểu đồ triển khai (Deployment Diagram)](#_bl10u3ld7w8s) 46

[Các bước cài đặt.](#_lv34myam6r1f) 47

[**Phụ lục**](#_6o44jbmhnbdz) **47**

[Tổng Kết](#_kbv1rdyg0n1w) 48

[Kết quả thu được.](#_jyuew5dndcb7) 48

[Hạn chế .](#_xsd3ak7i8ez) 48

# Giới thiệu dự án

1. Hệ thống dự định sẽ làm,

Đặt vé xem phim trực tuyến.

1. Mục đích của dự án

Hệ thống đặt vé xem phim trực tuyến cơ bản được thực hiện để cung cấp cho khách hàng dịch vụ mọi lúc mọi nơi để xem thông tin về các bộ phim đang chiếu qua đó họ có thể đặt vé ở bất kì đâu, bất kỳ lúc nào.

1. Phạm vi dự án được ứng dụng

Sử dụng cho rạp chiếu phim thế giới.

Ứng dụng tốt cho mô hình cá nhân

1. Tên hệ thống

Thế Giới Phim Trên Đầu Ngón Tay

1. Xác định yêu cầu khách hàng

Khách hàng muốn xem thông tin danh sách các phim đang chiếu.

Khách hàng muốn đăng ký thành viên để dễ dàng quản lý và đặt vé xem phim.

6. Môi trường triển khai (phần cứng, phần mềm)

Phần mềm : Hệ điều hành windows, macOS

Phần cứng : Laptop, PC

7. Công cụ kỹ thuật

Hệ quản trị cơ sở dữ liệu : MySQL Server.

Công cụ tạo báo cáo : Microsoft Word,Microsoft PowerPoint, Microsoft Excel.

Công cụ sử dụng để phát triển : Visual Studio Code, Violet UML.

<https://www.lucidchart.com/> , [https://www.draw.io/ ,](https://www.draw.io/)

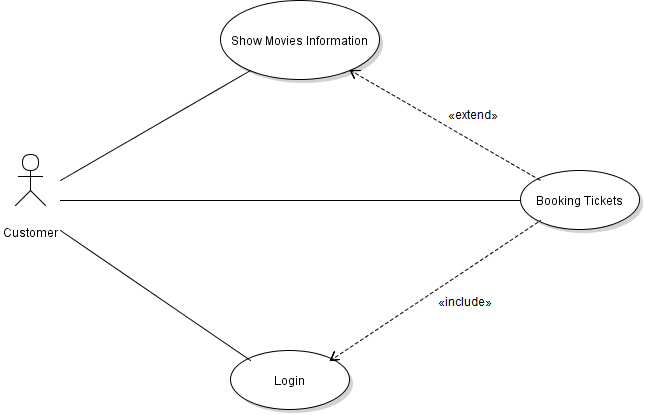
[https://www.google.com/intl/vi\_vn/docs/about/ .](https://www.draw.io/)

# Phân tích yêu cầu hệ thống

Hệ thống đặt vé xem phim online được xây dựng dựa trên những nhu cầu thực tế của khách hàng. Khánh hàng có thể mua vé xem phim trực tuyến mọi lúc mọi nơi mà không phải xếp hàng chờ đợi. Hệ thống có các chức năng là đặt vé, xem lịch chiếu phim và xem thông tin của khách hàng. Các chức năng này giúp người dùng dễ dàng xem được thông tin phim, lịch chiếu phim ,đặt vé, xem và thay đổi thông tin cá nhân.

## 1. Use Case

#### Use Case Diagram



#### *Mô tả Use Case*

##### Mô tả chức năng đăng nhập

|  |  |
| --- | --- |
| **Tiêu đề** | **Nội dung** |
| **Tên chức năng** | **Đăng nhập** |
| **Mô tả** | Chức năng này cho phép một người dùng đăng nhập vào hệ thống. |
| **Sự kiện kích hoạt** | Khách hàng muốn đăng nhập vào hệ thống. |
| **Actor** | Khách hàng. |
| **Tiền điều kiện** | Khách hàng muốn đăng nhập để vào hệ thống. |
| **Hậu điều kiện** | Thành công : Người dùng đăng nhập được vào hệ thống.  Lỗi : Người dùng không đăng nhập được vào hệ thống. |
| **Luồng sự kiện chính.** | Chức năng này bắt đầu khi người dùng muốn đăng nhập vào hệ thống.   1. Hệ thống yêu cầu người dùng nhập email hoặc user name và mật khẩu của mình. 2. Người dùng nhập email hoặc user name và mật khẩu. 3. Hệ thống xác thực thông tin đăng nhập và cho phép người dùng vào hệ thống. 4. Kết thúc chức năng. |
| **Luồng sự kiện phụ.** | Người dùng nhập Email hoặc mật khẩu không hợp lệ.  Người dùng nhập Email mật khẩu không đúng định dạng, không được có ký tự đặc biệt.   1. Hệ thống hiển thị màn hình đăng nhập kèm theo thông báo đăng nhập sai. 2. Người dùng có thể quay lại bước 2 ở luồng sự kiện chính hoặc chọn huỷ đăng nhập. |
| **Ngoại lệ** | 1. Mất kết nối cơ sở dữ liệu  * Hiển thị thông báo mất kết nối máy chủ. * Quay lại màn hình đăng nhập. |

##### - Mô tả chức năng xem các phim đang chiếu.

|  |  |
| --- | --- |
| **Tiêu đề** | **Nội dung** |
| **Tên chức năng** | **Xem thông tin phim.** |
| **Mô tả** | Chức năng này cho phép khách hàng xem thông tin các phim đang chiếu tại rạp. Khách hàng có thể xem thông tin chi tiết phim. |
| **Sự kiện kích hoạt** | Khách hàng chọn chức năng phim đang chiếu từ màn hình chính. |
| **Actor** | Khách hàng. |
| **Các chức năng liên quan.** | Đặt vé.  Đăng nhập. |
| **Tiền điều kiện** | Khách hàng muốn xem thông tin các phim đang chiếu tại rạp. |
| **Luồng sự kiện chính.** | Chức năng này sẽ bắt đầu khi khách hàng muốn xem thông tin phim.   1. Mỗi lần khách hàng yêu cầu xem thông tin phim hệ thống sẽ tìm và hiển thị một danh sách các phim đang chiếu tại rạp.   2. Khách hàng chọn một trong 2 chức năng :   * Xem thông tin chi tiết của phim. * Quay lùi lại menu chính.   3. Hệ thống đưa ra thông tin chi tiết phim đã chọn.  4. Khách hàng có thể lùi lại menu chính để chọn một chức năng khác  5. Kết thúc chức năng. |
| **Luồng sự kiện phụ.** | Luồng 1:   1. Khách hàng xem thông tin chi tiết của phim rồi có thể chọn đặt vé. 2. Khách hàng hàng xem thông tin chi tiết của phim rồi có thể quay lại menu chính thực hiện các chức năng khác. |
| **Ngoại lệ** | Ngoại lệ 1 :   1. Những sự kiện nhập số thứ tự để xem chức năng, số thứ tự phim người dùng nhập sai. 2. Hệ thống sẽ báo sai. 3. Người dùng nhập lại.   Ngoại lệ 2 :   1. Mất kết nối cơ sở dữ liệu . 2. Hiển thị thông báo mất kết nối với máy chủ. 3. Quay lại màn hình chính |

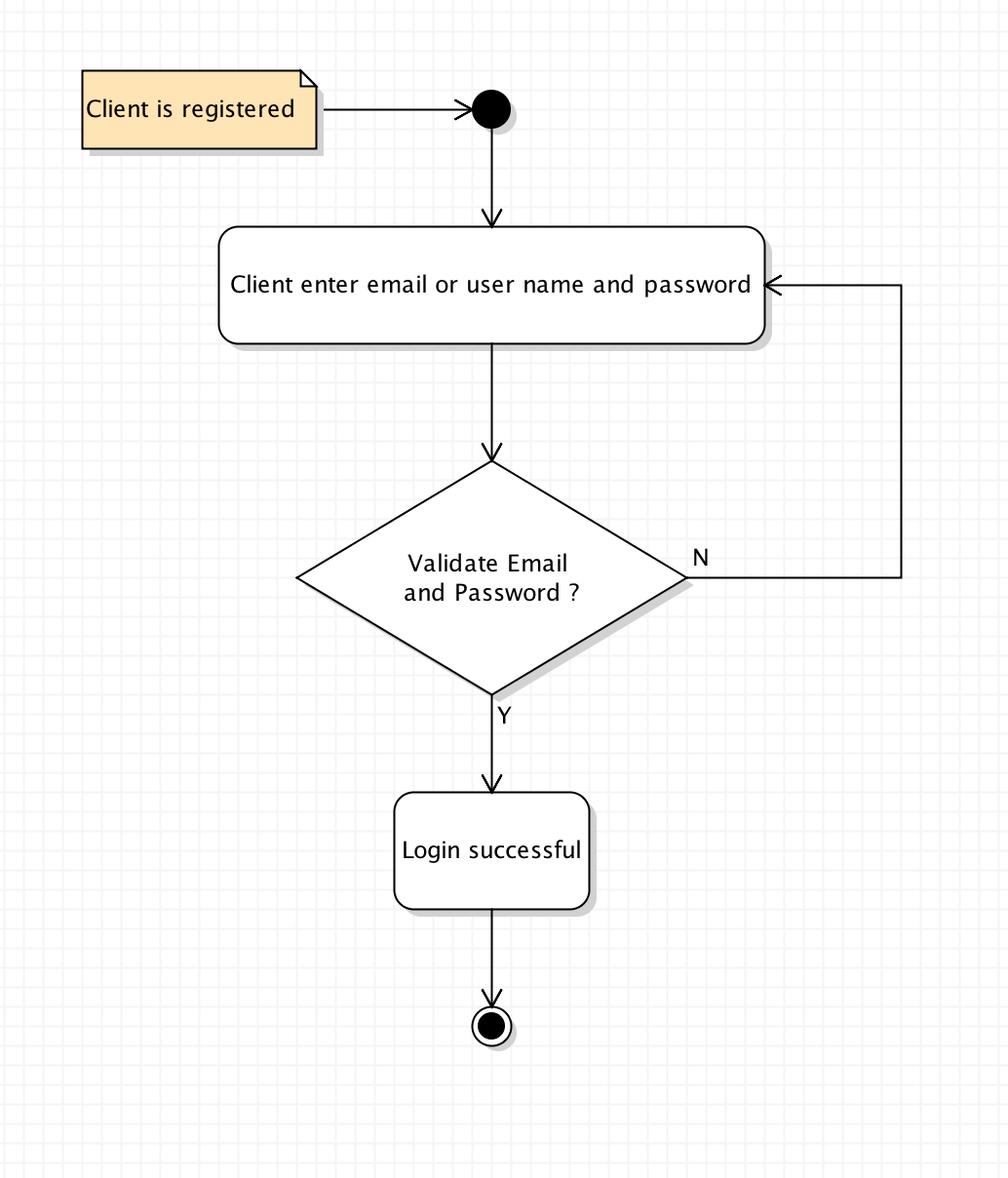
##### Mô tả chức năng đặt vé

|  |  |
| --- | --- |
| **Tiêu đề** | **Nội dung** |
| **Tên chức năng** | **Đặt vé.** |
| **Mô tả** | Chức năng này cho phép khách hàng đặt mua được vé khi muốn xem phim. |
| **Sự kiện kích hoạt** | Người dùng chọn chức năng đặt vé từ màn hình. |
| **Actor** | Người dùng. |
| **Các chức năng liên quan.** | Đăng nhập.  Xem chi tiết phim. |
| **Tiền điều kiện** | Khách hàng có nhu cầu mua vé để xem phim |
| **Hậu điều kiện** | Thành công : Thông tin vé đặt trước sẽ được gửi cho khách hàng.  Lỗi : Phiếu đặt vé sẽ không được lập, khách hàng sẽ nhận thông báo lỗi. |
| **Luồng sự kiện chính.** | Chức năng này bắt đầu khi khách hàng xem thông tin lịch chiếu của từng bộ phim và chọn đặt vé.   1. Hệ thống hiển thị toàn bộ danh sách các phim đang được chiếu tại rạp. 2. Khách hàng chọn bộ phim muốn xem. 3. Khách hàng chọn lịch chiếu phù hợp. 4. Khách hàng chọn vị trí ghế ngồi. 5. Hệ thống cập nhập các thông tin vào phiếu đặt. 6. Khách hàng có thể chọn chức năng :  * Đặt thêm vé cho bộ phim khác. Quá trình được lặp lại từ đầu. * Thanh toán để hoàn tất quá trình.   7. Hệ thống sẽ gửi thông tin vé đặt trước đến khách hàng. |
| **Luồng sự kiện phụ.** | Luồng 1 : Khi hiển thị phiếu đặt vé   1. Huỷ : Khách hàng sẽ hủy bỏ quá trình đặt mua vé. 2. Thanh toán : Tài khoản của khách hàng sẽ bị trừ tiền.   Luồng 2: Nếu khách hàng đặt ghế mà đã có người đặt rồi.   1. Hệ thống sẽ hiển thị một thông báo, ghế này đã có người khác chọn hoặc không có trong danh sách ghế, vui lòng chọn lại. 2. Quay lại bước đặt ghế hoặc quay lại menu chính. 3. Tiếp tục thực hiện. |
| **Ngoại lệ** | Ngoại lệ 1 : Trong lúc thanh toán mất kết nội mạng, cơ sở dữ liệu   1. Hiển thị thông báo mất kết nối máy chủ. 2. Quay lại màn hình chính. |

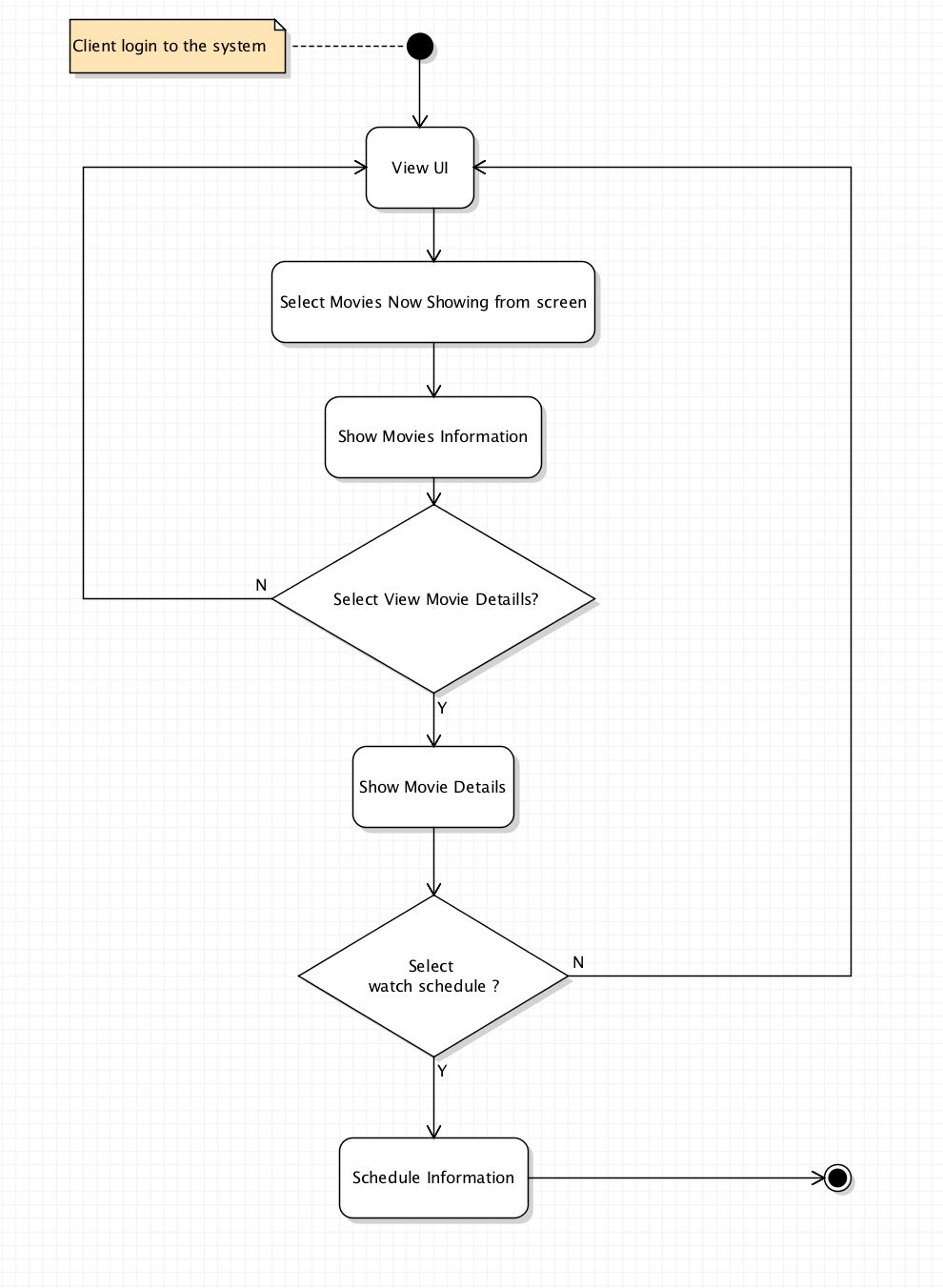
### 

## 2. Sơ đồ hoạt động (Activity Diagram)

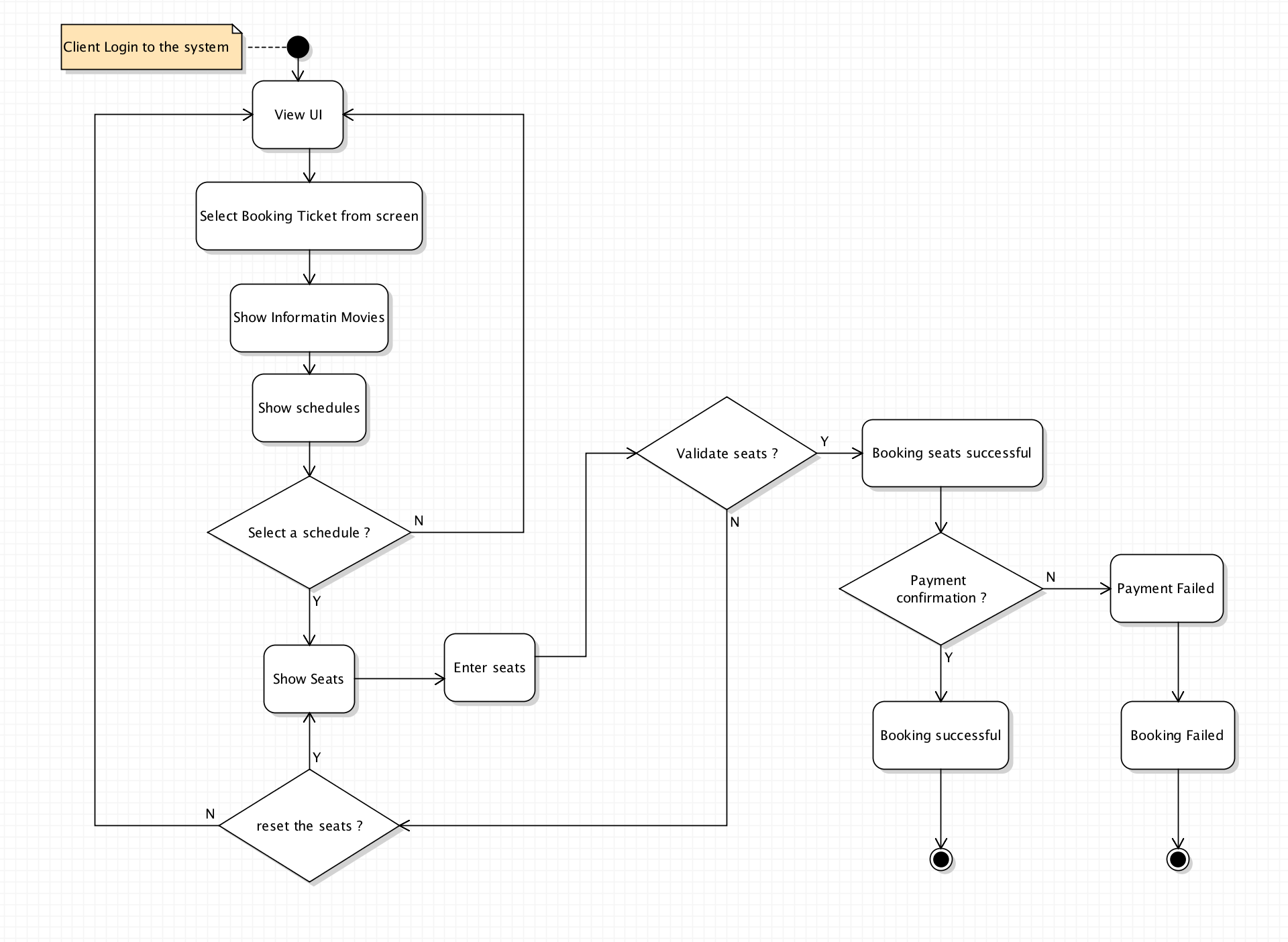
#### Đăng nhập (Login)



#### Xem các phim đang chiếu (Movie Now Showing)

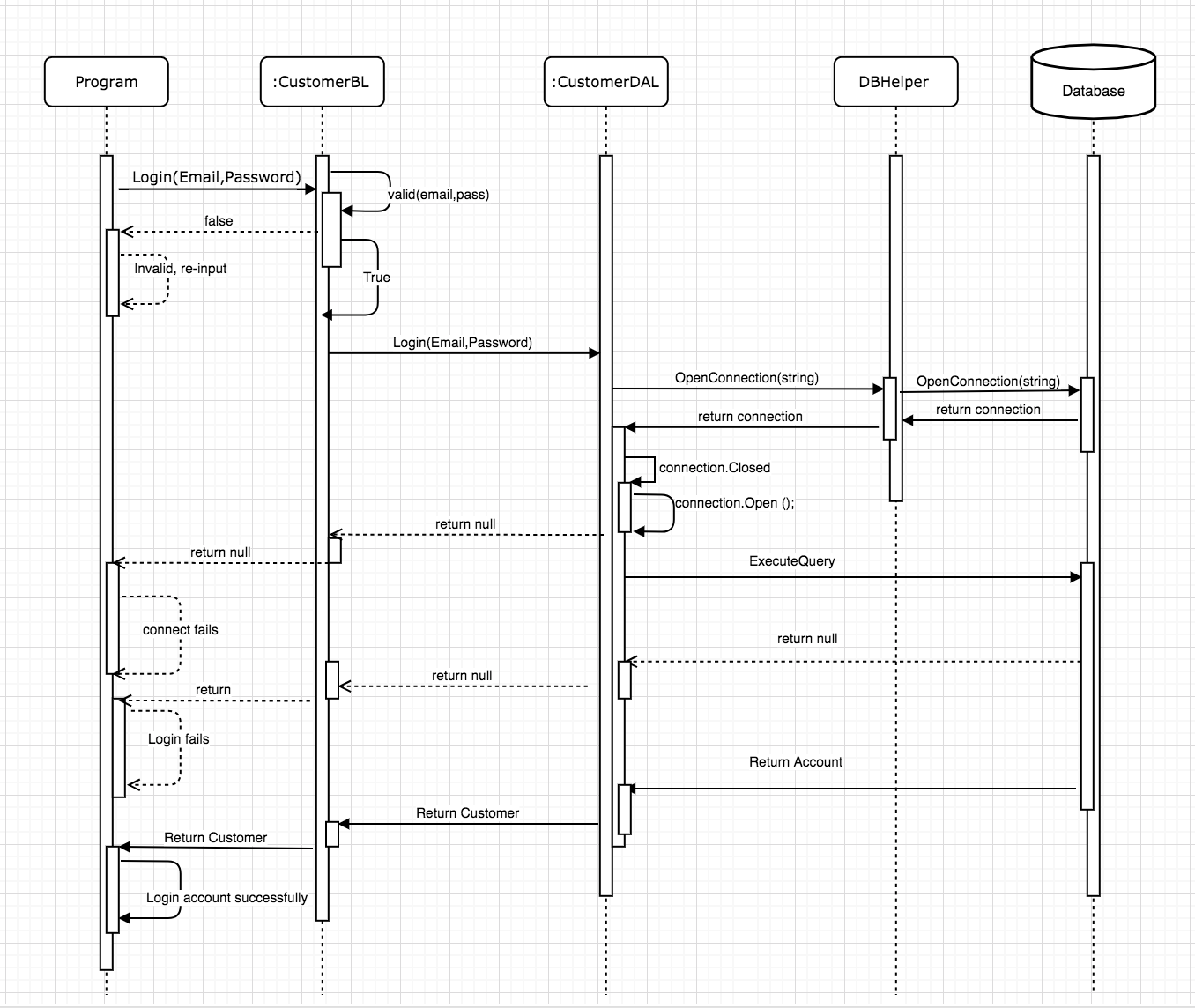


#### Đặt vé ( Booking Ticket )

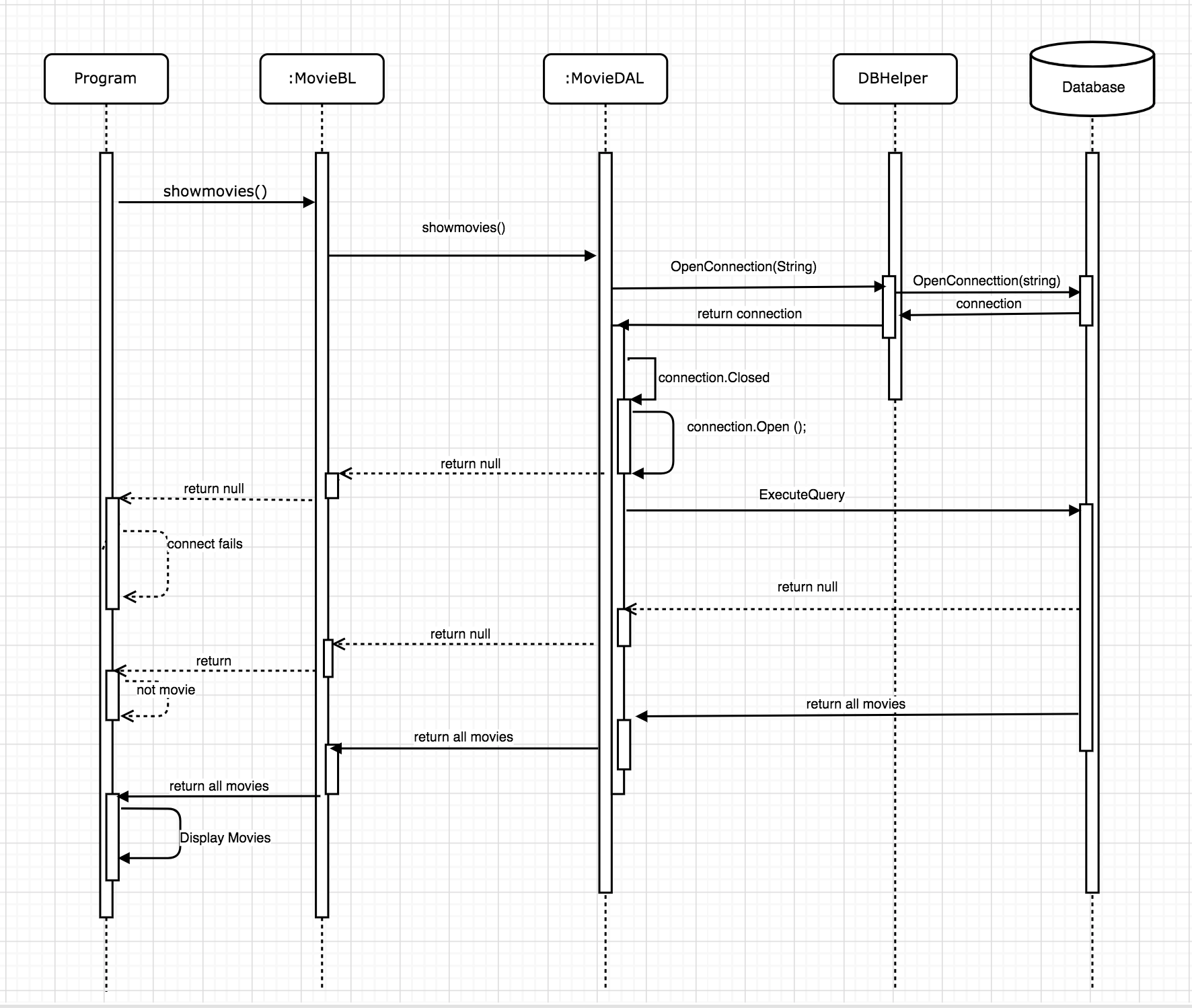


## 3. Sequence Diagram

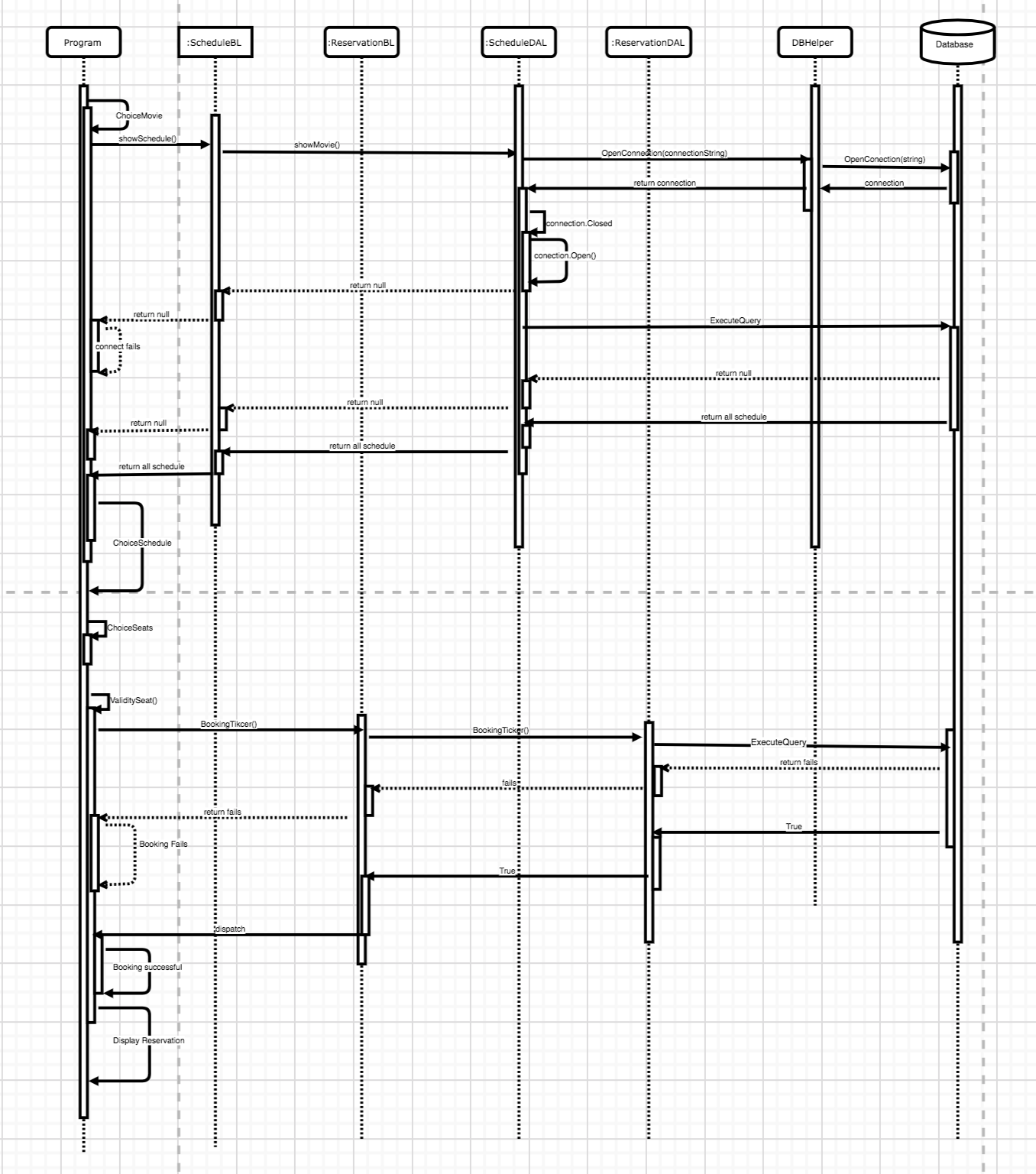
### Đăng nhập (Login)



### Phim đang chiếu. ( Now Showing )



### Đặt vé ( Booking Ticket )



# Thiết kế chi tiết

### Thiết kế giao diện (GUI)

#### Mô tả chức năng giao diện

Chương trình gồm có các chức năng sau :

• **Đăng nhập.**

• **Xem thông tin phim.**

• **Đặt vé**

**• Quản lý thông tin cá nhân.**

Giao diện và chi tiết chức năng.

Chương trình có menu chính như bên dưới và khi chọn các lựa chọn thì chương  
 trình sẽ hoạt động như sau.

#### Thiết kế giao diện

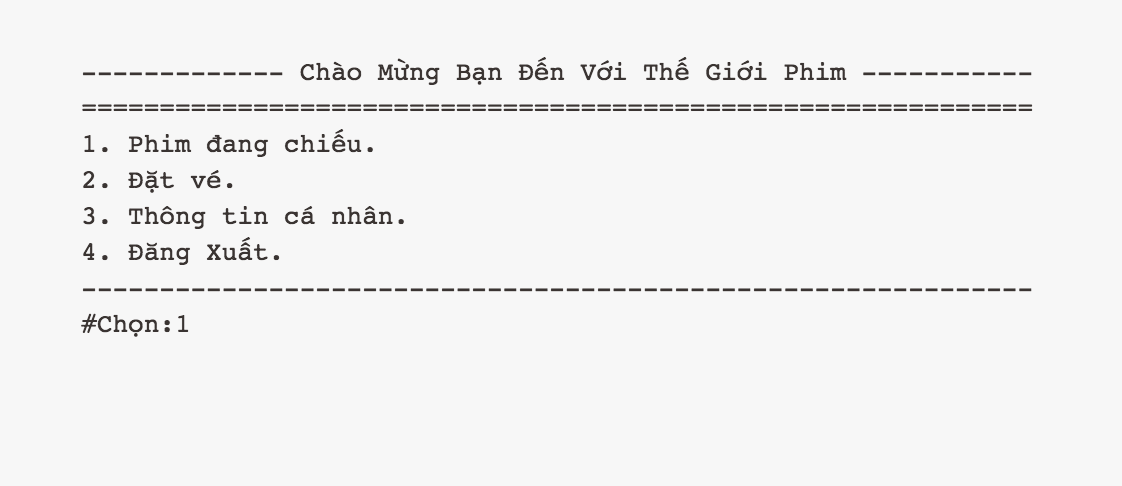
##### Giao diện đăng nhập

Chọn 1, chức năng đăng nhập sẽ như sau

****

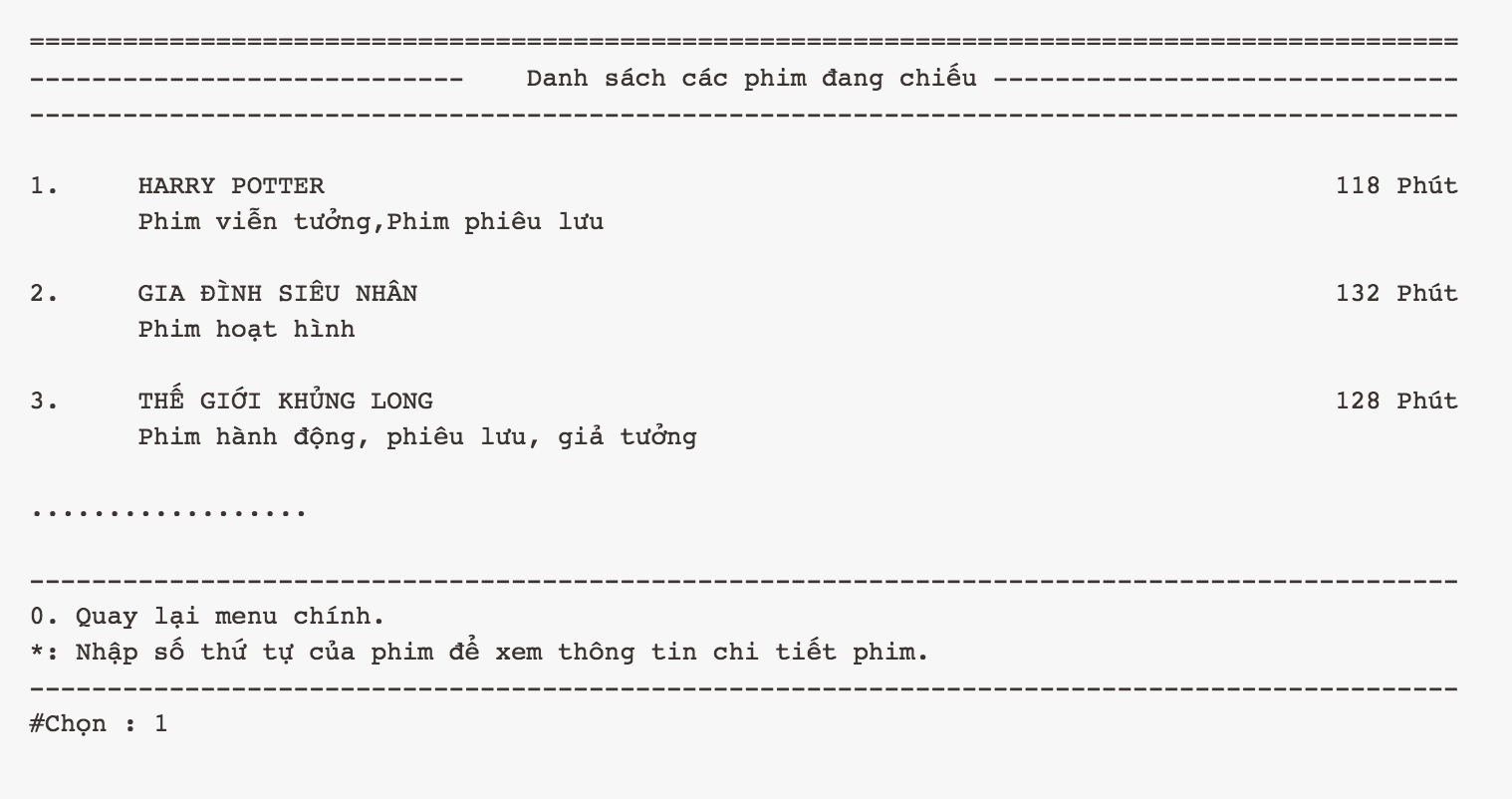
##### Giao diện rạp chiếu phim.

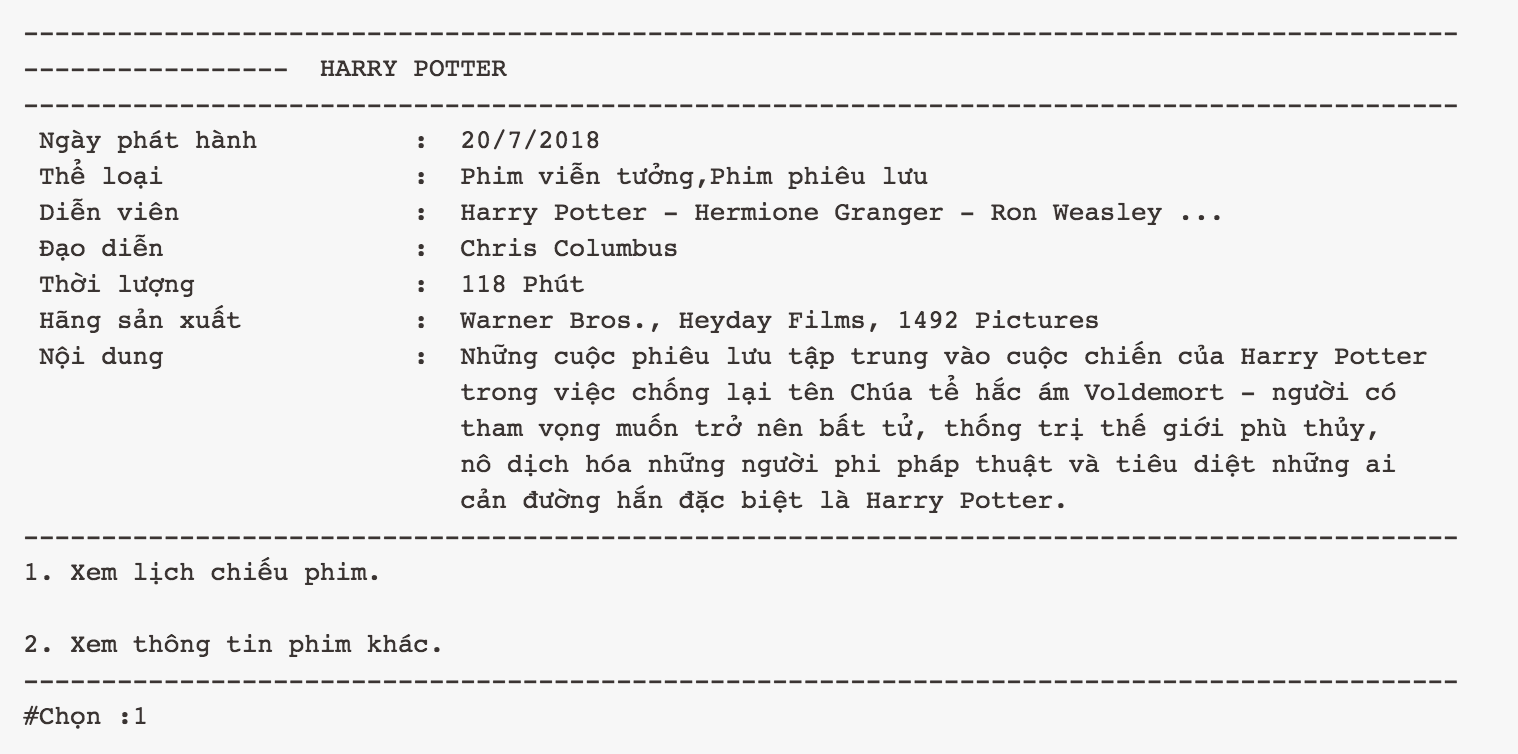
**Đăng nhập** vào hệ thống sẽ hiển thị menu như sau:



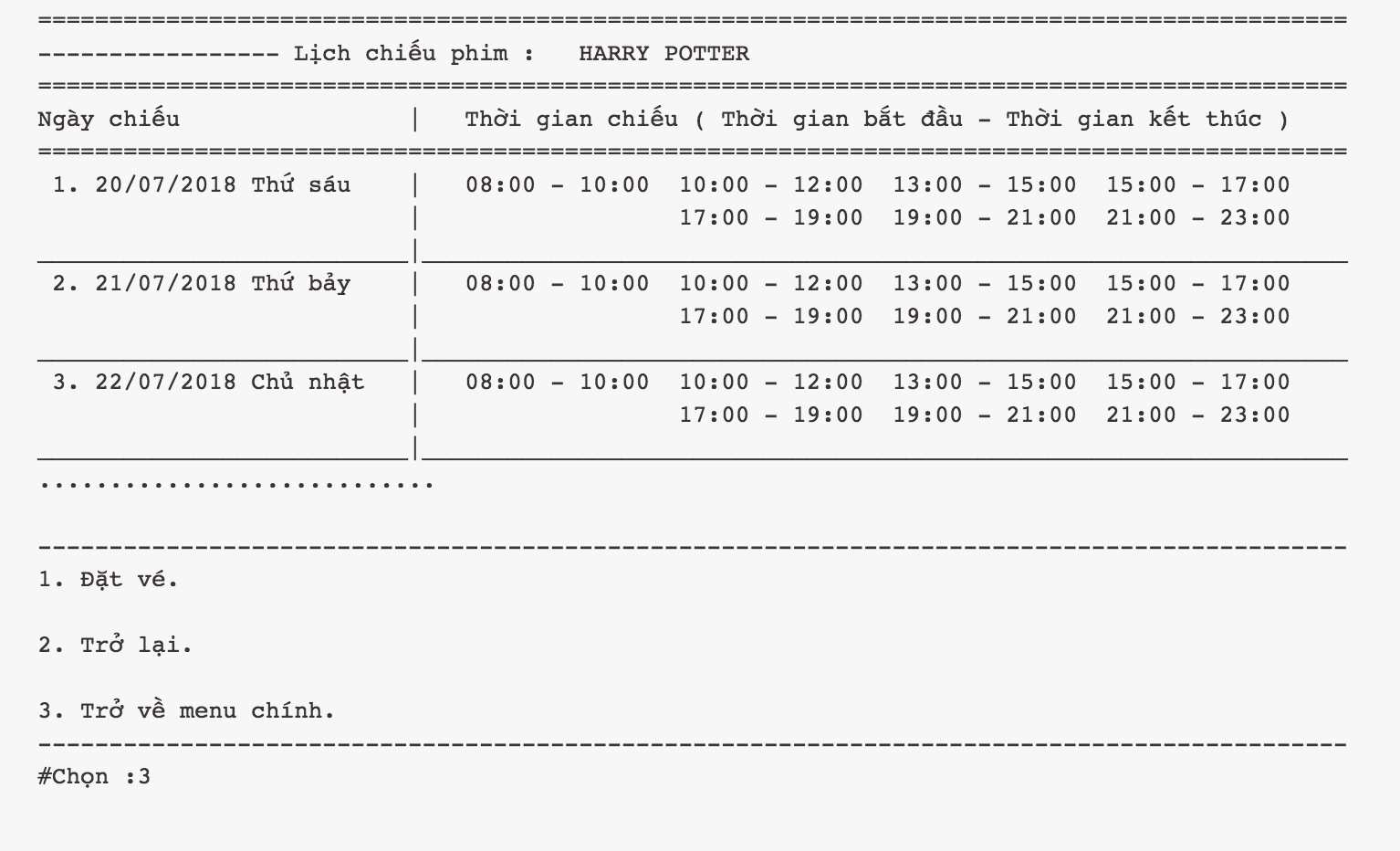
##### Giao diện danh sách phim đang chiếu

Tiếp tục chọn chức năng 1, Menu **Phim** sẽ hiển thị danh sách các phim.

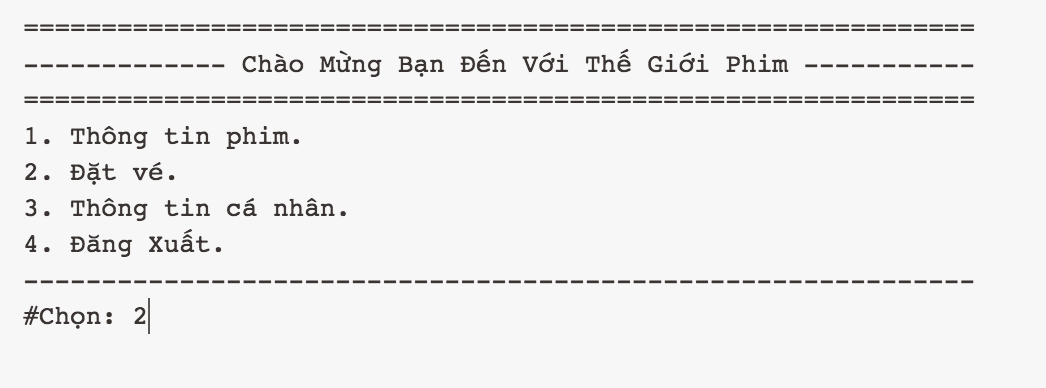




Tiếp tục chọn 1 **Xem** **Lịch chiếu** sẽ hiển thị như sau.

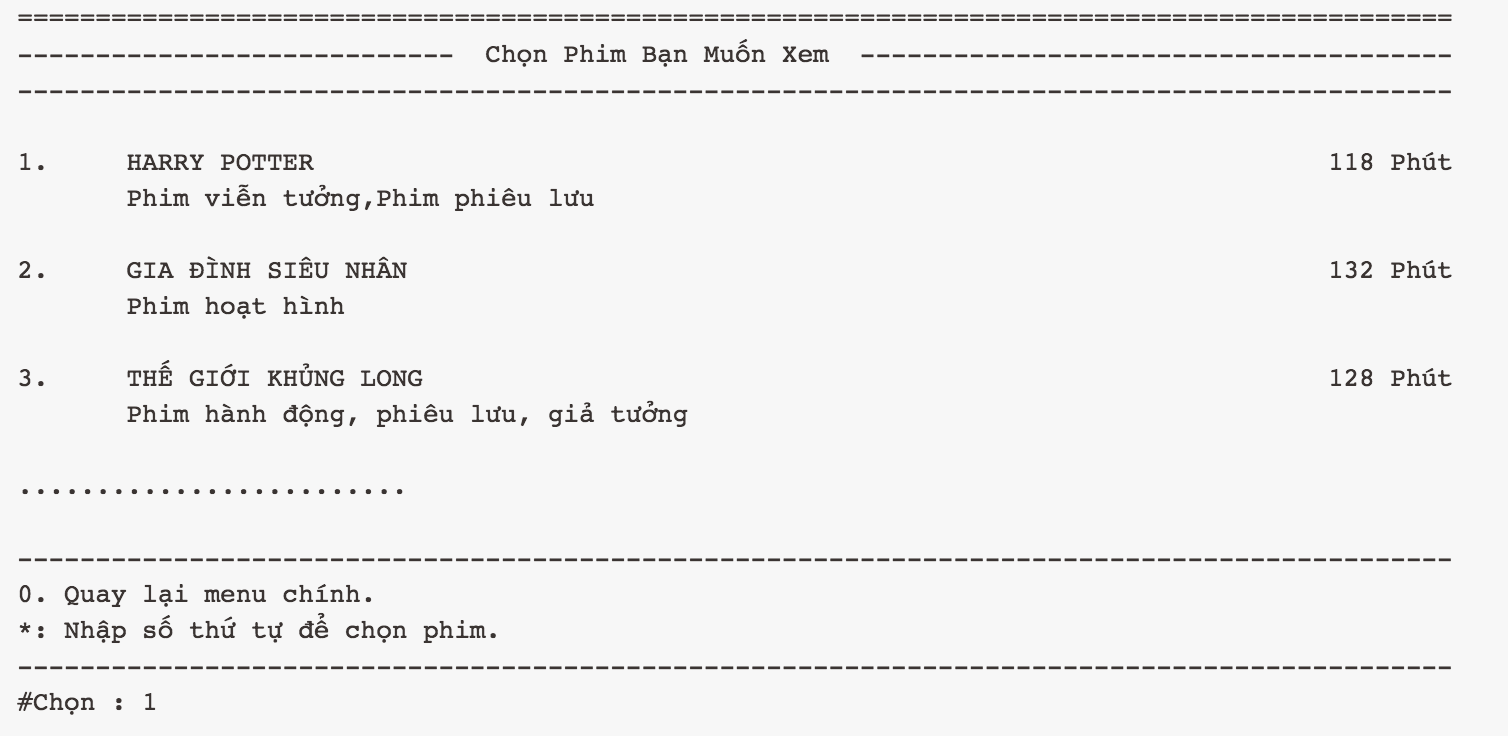


Trở lại với Menu chính khi vừa **Đăng Nhập** xong :

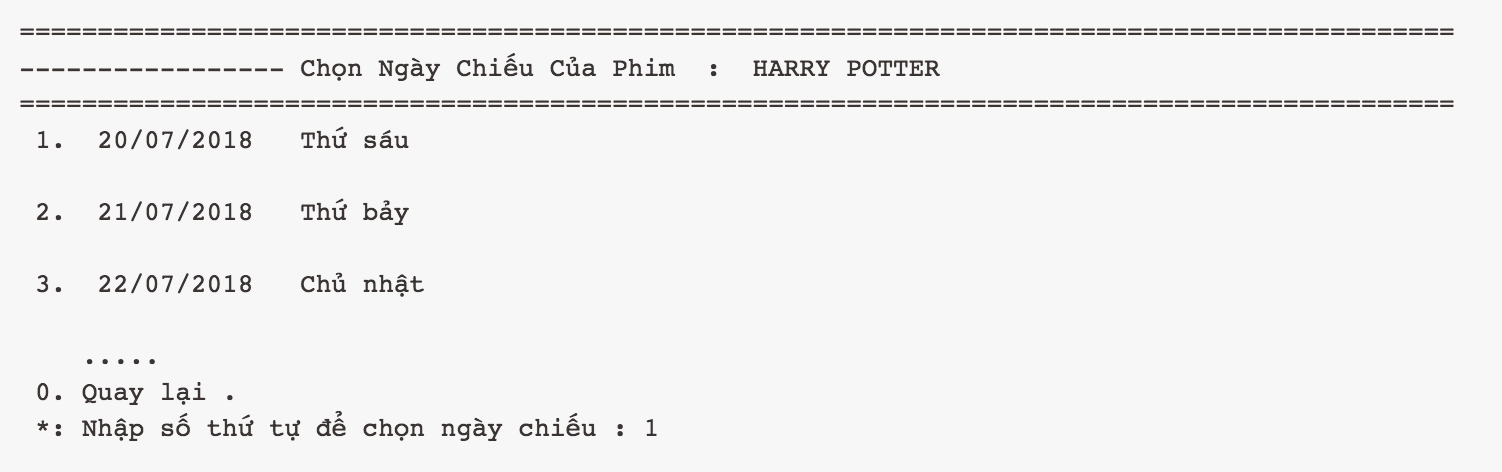
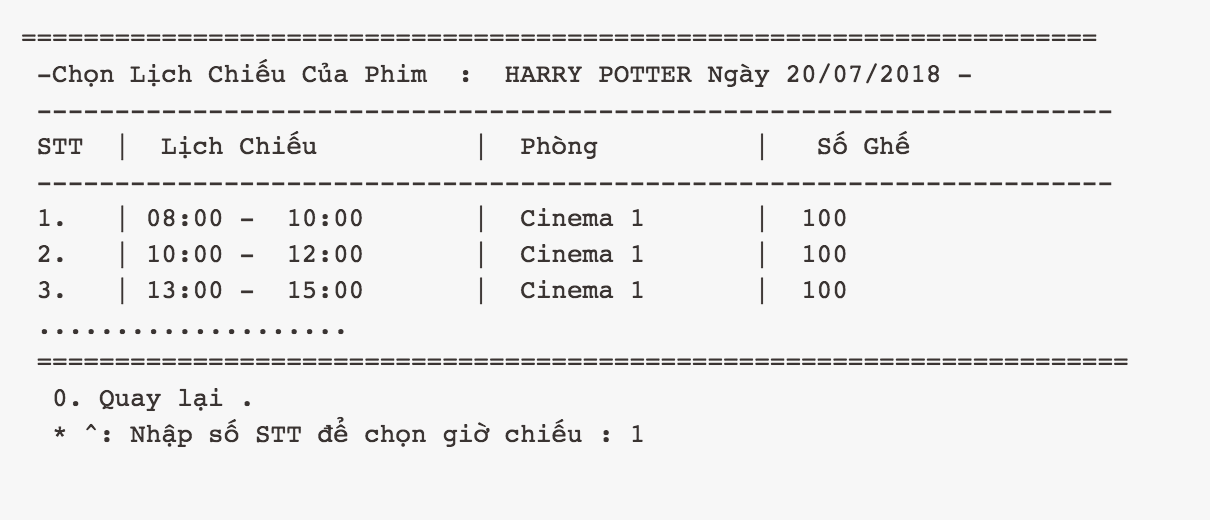


##### Giao diện Đặt vé

Chọn 2, **Đặt Vé** giao diện sẽ hiển thị như sau.



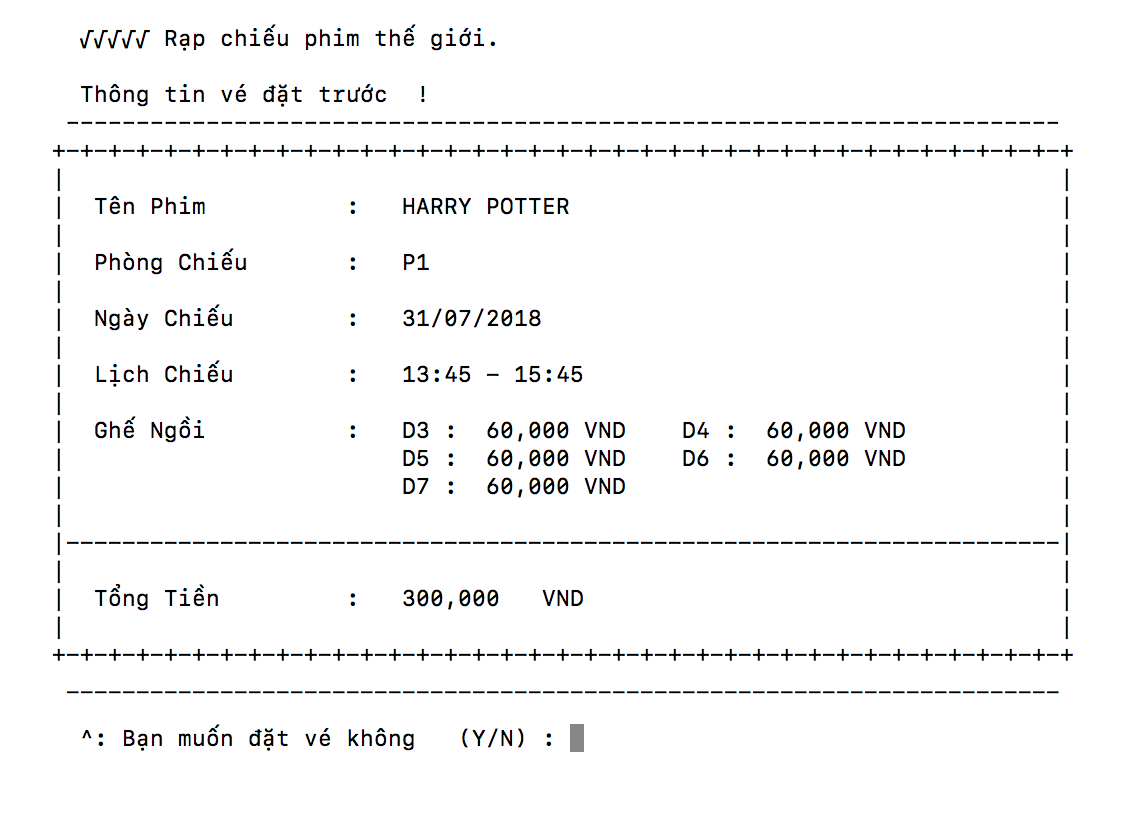
* Giao diện hiển thị lịch chiếu phim



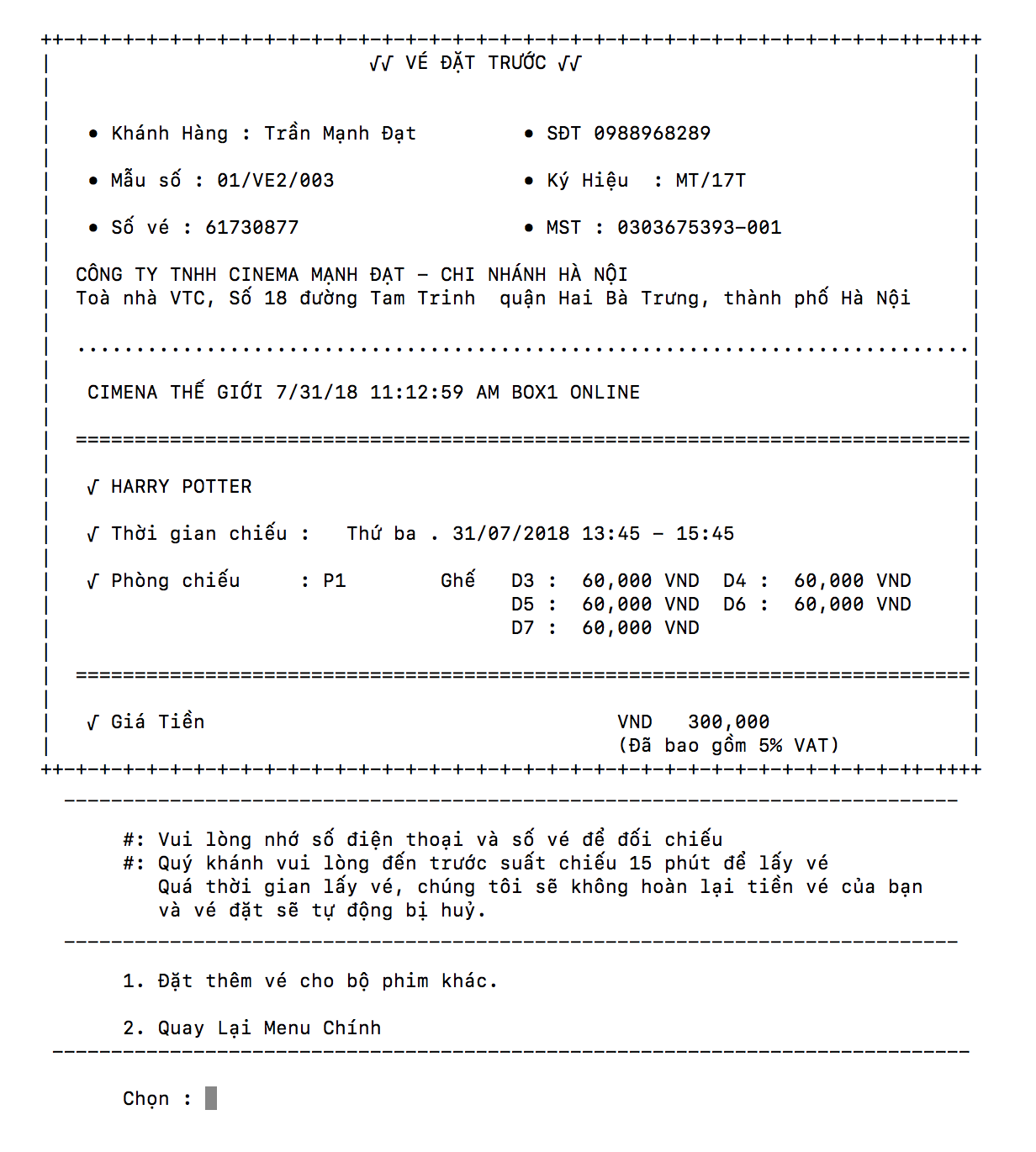
##### Giao diện đặt ghế



Chọn 1: **Đặt vé các ghế đã chọn**!

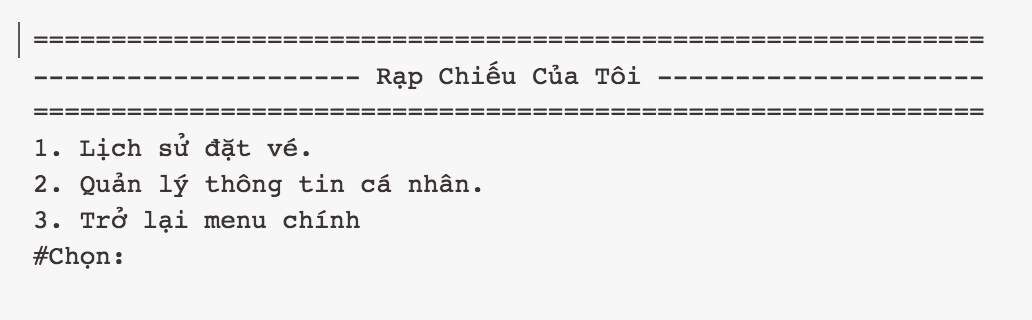


##### Thông tin vé đặt trước

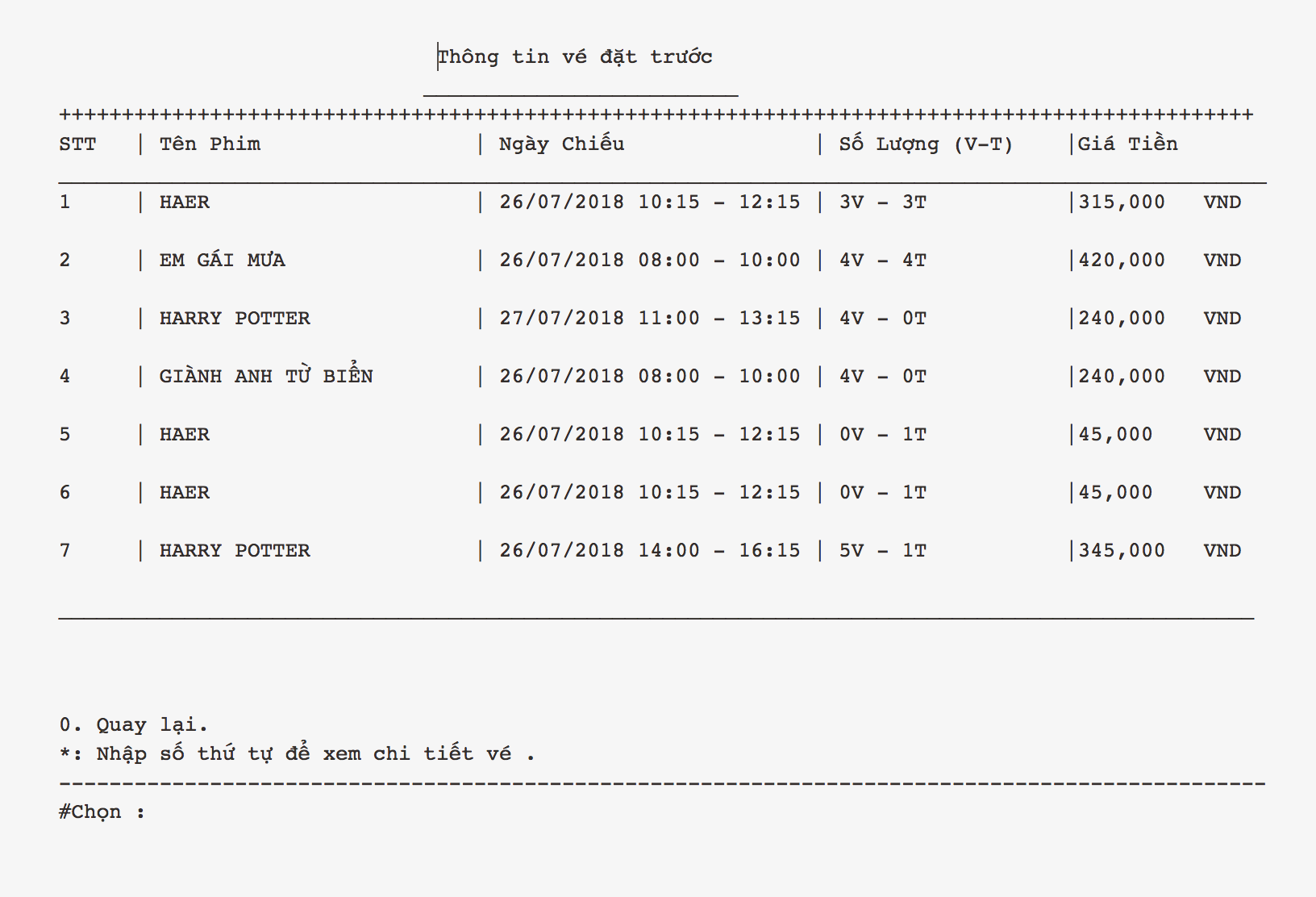


Chọn 2 quay lại menu chính.

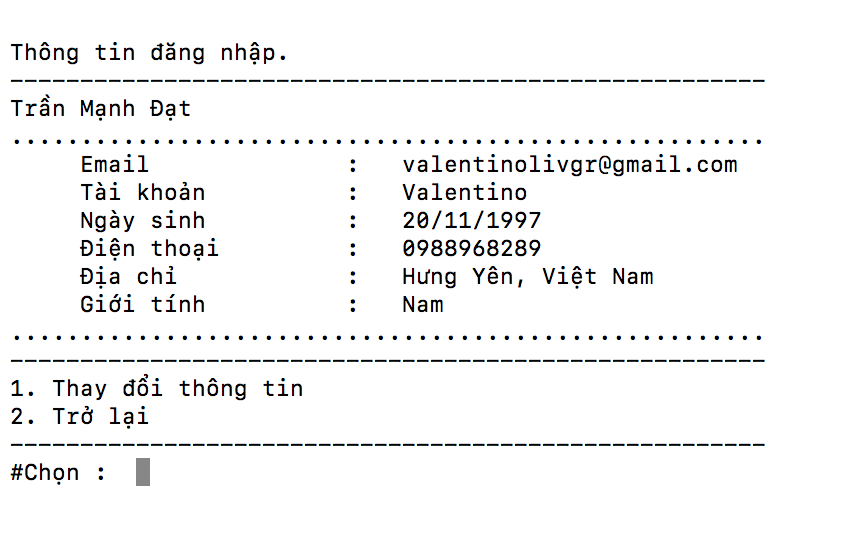
##### Giao diện quản lý thông tin cá nhân

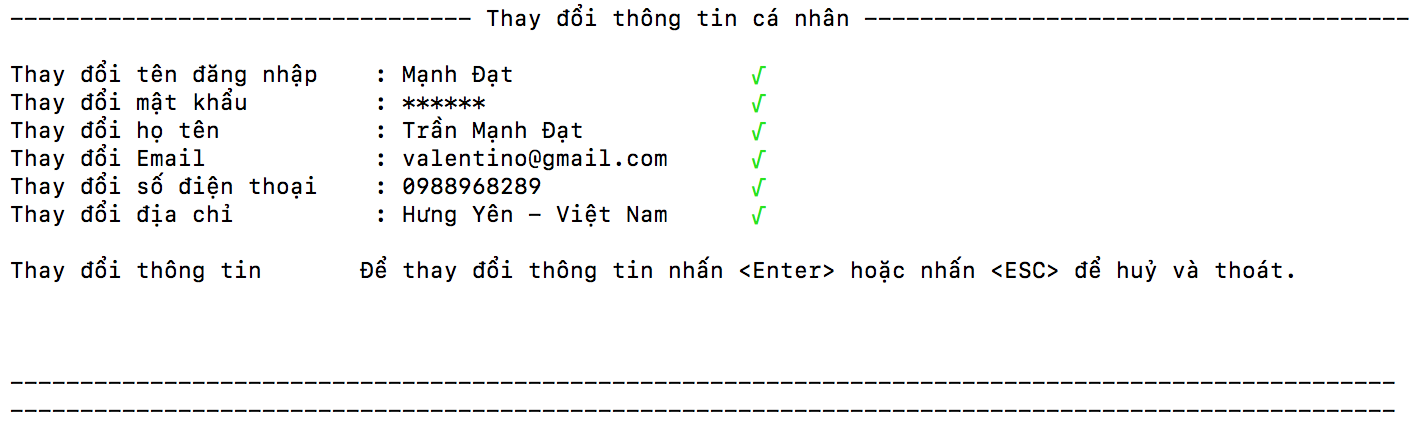


Chọn 1:



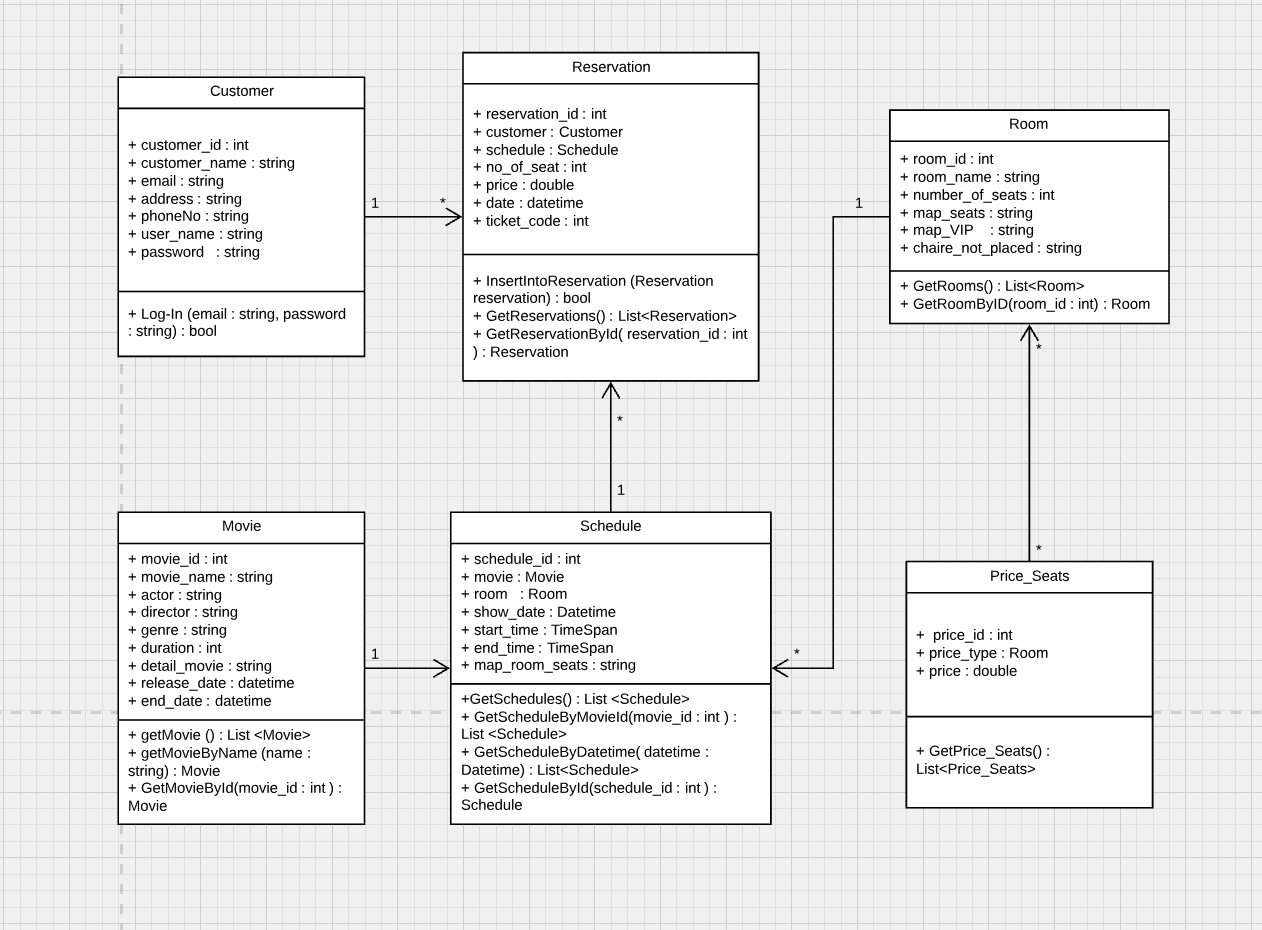
Chọn 2:



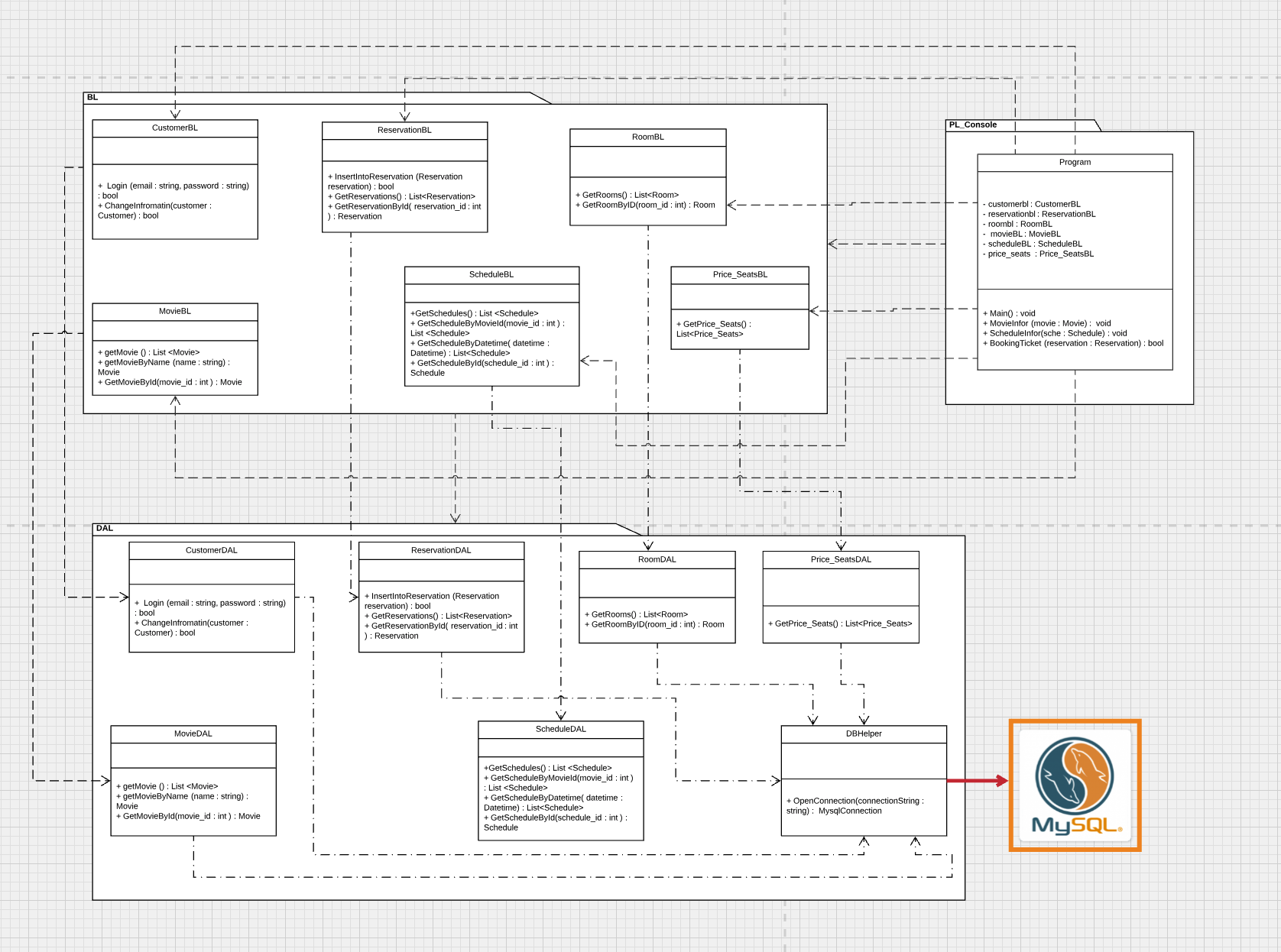


### Thiết kế mã (Class Diagram)

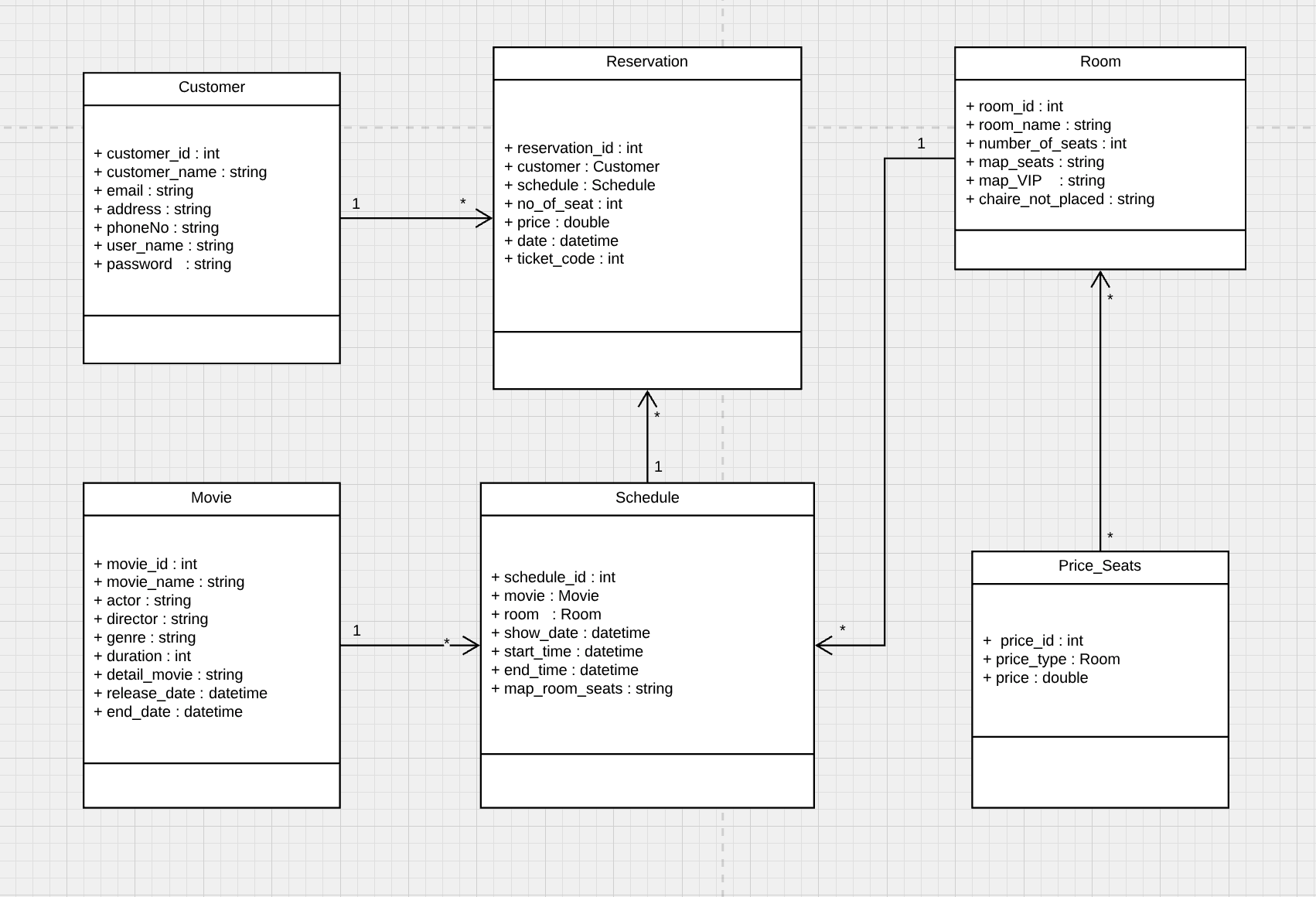
#### Class Diagram



#### MÔ HÌNH 3 LỚP (3 – LAYER)

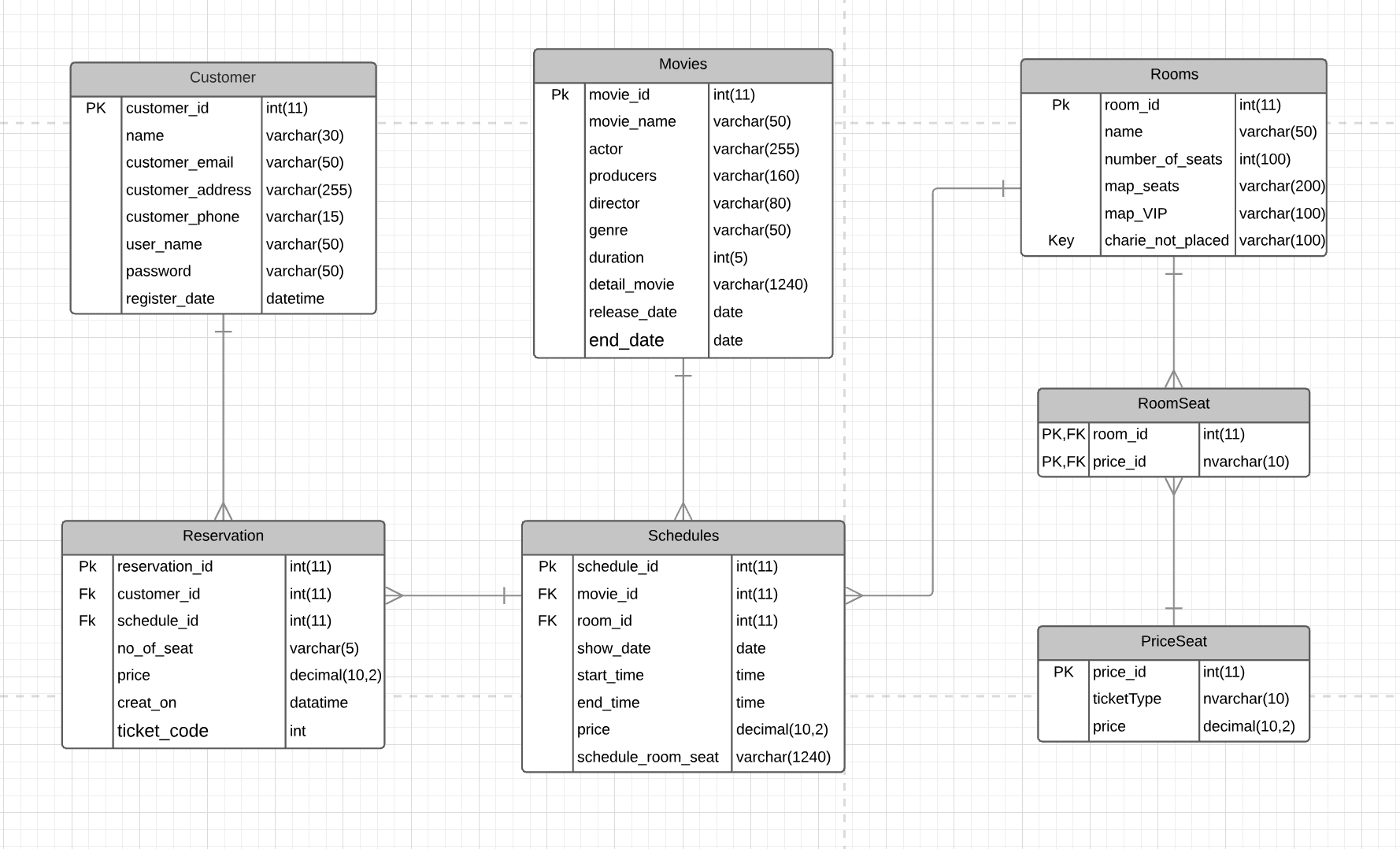


Persistence



### Thiết kế CSDL (Database)

#### Entity Relationship Diagram



#### Thiết kế CSDL chi tiết

|  |  |  |  |
| --- | --- | --- | --- |
| **Customer** | | | |
| **Column Name** | **Data Type** | **Constraints** | **Description** |
| customer\_id | int(11) | Primary Key, Autoincrement  UNIQUE | ID của khánh hàng |
| name | varchar(30) | NOT NULL | Tên của khách hàng |
| customer\_email | varchar(50) | NOT NULL | Địa chỉ Email |
| customer\_address | varchar(255) | NOT NULL | Địa chỉ. |
| customer\_phone | varchar(15) |  | Số điện thoại của khách hàng |
| user\_name | varchar(50) | NOT NULL | Tên hiển thị |
| password | varchar(50) | NOT NULL | Mật khẩu đăng nhập. |
| register\_date | datetime | NOT NULL current\_timestamp | Ngày đăng ký |

|  |  |  |  |
| --- | --- | --- | --- |
| **Reservation** | | | |
| **Column Name** | **Data Type** | **Constraints** | **Description** |
| reservation\_id | int(11) | Primary Key, Autoincrement  UNIQUE | ID của bảng đặt trước |
| customer\_Id | int(11) | NOT NULL | Là khoá ngoại tham chiếu đến **id** của **Customer.** |
| schedule\_id | int(11) | NOT NULL | Là khoá ngoại tham chiếu đến **id** của **Schedule** |
| no\_of\_seat | varchar(5) | NOT NULL | Số ghế ngồi. |
| price | decimal(10,2) | Not NULL | Thành tiền. |
| create\_on | datetime | current\_timestamp | Thời gian tạo. |
| ticket\_code | int | not null | Mã vé |

|  |  |  |  |
| --- | --- | --- | --- |
| **Movies** | | | |
| **Column Name** | **Data Type** | **Constraints** | **Description** |
| movie\_id | int(11) | Primary Key, Autoincrement  UNIQUE | ID của phim. |
| movie\_name | varchar(50) | NOT NULL, UNIQUE | Tên của phim. |
| actor | varchar(255) | NOT NULL | Diễn viên trong phim. |
| producers | varchar(160) | NOT NULL | Nhà sản xuất. |
| director | varchar(80) | NOT NULL | Đạo diễn. |
| genre | varchar(50) | NOT NULL | Thể loại. |
| detail\_movie | varchar(1240) | NOT NULL | Nội dung của phim. |
| duration | int(5) | NOT NULL | Thời gian của phim. |
| release\_date | date | NOT NULL | Thời gian khởi chiếu. |
| end\_date | date | NOT NULL | Thời gian kết thúc. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Schedules** | | | |
| **Column Name** | **Data Type** | **Constraints** | **Description** |
| schedule\_id | int(11) | Primary Key, Autoincrement  UNIQUE | ID của lịch chiếu |
| room\_id | int(11) | NOT NULL | Là khoá ngoại tham chiếu đến **id** của bảng **Rooms** |
| movie\_id | int(11) | NOT NULL | Là khoá ngoại tham chiếu đến **id** của bảng **Movie** |
| show\_date | date | NOT NULL | Ngày chiếu của phim. |
| start\_time | time | NOT NULL | Là khoá ngoại tham chiếu tới id của bảng **Cinemas**. |
| end\_time | time | NOT NULL | Là khoá ngoại tham chiếu tới id của **Movie** |
| price | decimal(10,2) | NOT NULL | Giá. |
| schedule\_room\_seat | varchar(1240) | NOT NULL | Bản đồ ghế ngồi. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Rooms** | | | |
| **Column Name** | **Data Type** | **Constraints** | **Description** |
| **room\_id** | int(11) | Primary Key, Autoincrement  UNIQUE | ID của phòng chiếu |
| room\_name | varchar(50) | NOT NULL, UNIQUE | Tên phòng chiếu. |
| number\_of\_seats | int(100) | NOT NULL | Sức chứa của phòng. |
| map\_seats | varchar(200) | NOT NULL | Bản đồ ghế của phòng. |
| map\_VIP | varchar(200) |  | Ghế VIP của phòng |
| chaire\_not\_placed | varchar(200) |  | Ghế lỗi không thể đặt |

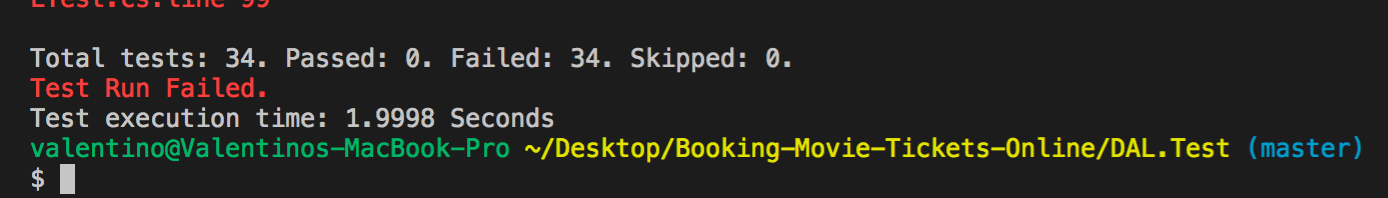
|  |  |  |  |
| --- | --- | --- | --- |
| **Price\_Seats** | | | |
| **Column Name** | **Data Type** | **Constraints** | **Description** |
| **price\_id** | int(11) | Primary Key, Autoincrement  UNIQUE | ID của giá ghế |
| price\_Type | varchar(10) | NOT NULL, UNIQUE | Kiểu giá. |
| price | double(10,2) | NOT NULL | Giá ghế. |

# Kiểm thử

Test run Successful

# 

Test Run Failed



### 4.1. Kiểm thử tầng DAL

#### Class CustomerDAL

|  |  |
| --- | --- |
| **ID** | Test\_Case 1 |
| **Tên kiểm thử** | TestLogin |
| **Mô tả kiểm thử** | Kiểm thử phương thức Login() trong Class CustomerDAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương thức login() trong Class CustomerDAL tầng DAL. |
| **Đầu vào kiểm thử** | 1. login(“valentinolivgr@gmail.com", "123456”) 2. login("abc d e f'", "123") 3. login(",,'1`12", "as''dfg") |
| **Đầu ra kiểm thử** | 1. Trả về NOTNULL vào giao diện người dùng. 2. Trả về NULL (Email, Password đã nhập không đúng). 3. Trả về NULL(Email, Password không được nhập ký tự đặc biệt). |
| **Các bước kiểm thử** | 1. Nhập Email, Password 2. Email, Password là tham số truyền vào phương thức Login() trong Class CustometDAL 3. Trả về NULL nếu trong cơ sở dữ liệu không có Email và Password như đã nhập.   Trả về một Customer ( NOT NULL ) nếu tìm thấy trong cơ sở dữ liệu có Email và Password giống như đã nhập. |

#### Kiểm thử Class MovieDAL

|  |  |
| --- | --- |
| **ID** | Test\_Case 2.1 |
| **Tên kiểm thử** | Test GetMovieDAL |
| **Mô tả kiểm thử** | Kiểm thử phương thức GetMovies trong tầng DAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu, đã viết hàm GetMovies trong tầng DAL. |
| **Đầu vào kiểu thử** | 1. Lấy ra một List<Movie> trong cơ sở dữ liệu. 2. Lấy ra một Movie ngẫu nhiên trong cơ sở dữ liệu. 3. Lấy ra vị trí Top của Movie trong cơ sở dữ liệu. 4. Lấy ra Movie Bottom trong cơ sở dữ liệu |
| **Đầu ra kiểm thử** | 1. Trả về một List<Movie> (NotNull). 2. Trả về một Movie ngẫu nhiên ( Not Null ).Kiểm tra Movide Rand trong cơ sở dữ liệu có trong List<Movie> hay không ? (Contains). 3. Trả về Movie Top (NotNull). So sánh Movie Top trong cơ sở dữ liệu có giống Movie đầu tiên trong List<Movie> vừa lấy ?(True). 4. Trả về Movie Bottom (Not Null ). So sánh Movie Bottom trong cơ sở dữ liệu có giống Movie cuối cùng trong List<Movie> ? (True). |
| **Các bước kiểm thử** | 1. Gọi hàm GetMovies trong tầng DAL lấy ra 1 List<Movie> 2. Viết một query để lấy một Movie ngẫu nhiên trong cơ sở dữ liệu. 3. Viết một query để lấy Movies Top trong cơ sở dữ liệu. 4. Viết một query để lấy Movies Bottom trong cơ sở dữ liệu |

|  |  |
| --- | --- |
| **ID** | Test\_Case 2.2 |
| **Tên kiểm thử** | Test Get Movie By Name |
| **Mô tả kiểm thử** | Kiểm thử phương thức GetMovieByName trong Class Movie\_DAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương GetMovieByName() trong DAL. |
| **Đầu vào kiểm thử** | 1. GetMovieByName(“HARRY POTTER”); 2. GetMovieByName(“”); 3. GetMovieByName(“Khong co ten phim”); |
| **Đầu ra kiểm thử** | 1. Trả về Movie có tên Harry Potter ( NOT NULL ). 2. Trả về NULL , (Nhập tên phim không được để trống) 3. Trả về NULL , ( Không tìm thấy tên phim trong DATABASE) |
| **Các bước kiểm thử** | 1. Nhập tên phim. 2. Tìm kiếm tên Phim đấy có trong cơ sở dữ liệu không ? Bằng phương thức GetMovieByName(name). 3. - Nếu Name = “” hoặc Name = NULL : Trả về NULL , tham số đầu vào không được rỗng.   - Nếu tìm kiếm trong cơ sở dữ liệu không thấy tên phim như vừa nhập thì trả về NULL.  - Nếu tìm thấy trong cơ sở dữ liệu có tên giống với tên vừa nhập thì trả về một Movie (NOT NULL) |

# 

|  |  |
| --- | --- |
| **ID** | Test\_Case 2.3 |
| **Tên kiểm thử** | Test Get Movie By ID |
| **Mô tả kiểm thử** | Kiểm thử phương thức GetMovieById() trong DAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương thức GetMovieById() trong DAL. |
| **Đầu vào kiểm thử** | 1. GetMovieById(1); 2. GetMovieById(0) |
| **Đầu ra kiểm thử** | 1. Trả về Movie có ID = 1 ( NOT NULL ) 2. Trả về NULL (Không tìm thấy movie\_id = 0 trong Database) |
| **Các bước kiểm thử** | 1. Nhập ID của phim 2. Tìm kiếm ID của phim có trong cơ sở dữ liệu không ? Bằng phương thức GetMovieById(movie\_id); 3. - Nếu movie\_id bằng 0 hoặc movie\_id bằng null. Trả về Null tham số đầu vào không được bằng 0 hoặc null.   - Nếu tìm kiếm trong cơ sở dữ liệu không có movie\_id vừa nhập thì trả về Null.  - Nếu tìm kiếm trong cơ sở dữ liệu có MovieID vừa nhập thì trả về một Movie (NOT NULL) |

#### - Kiểm thử Class Room

|  |  |
| --- | --- |
| **ID** | Test\_Case 3.1 |
| **Tên kiểm thử** | TestGetRoomDAL |
| **Mô tả kiểm thử** | Kiểm thử phương thức GetRooms trong tầng DAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu, đã viết phương thức Get Room trong tầng DAL. |
| **Đầu vào kiểu thử** | 1. Lấy ra một List<Room> trong Database. 2. Lấy ra một Room ngẫu nhiên trong Database 3. Lấy ra vị trí Top của Room trong Database 4. Lấy ra Room Bottom trong Database |
| **Đầu ra kiểm thử** | 1. Trả về một List<Room> (NotNull). 2. Trả về một Movie Rand (NotNull ). Kiểm tra Room Rand trong Database có trong List<Room> hay không ? (Contains). 3. Trả về Room Top ( NOT NULL ). So sánh Room Top trong Database có giống Room đầu tiên trong List<Room> vừa lấy ?(True). 4. Trả về Room Bottom (NOT NULL ). So sánh Room Bottom trong Database có giống Room cuối cùng trong List<Movie> ? (True). |
| **Các bước kiểm thử** | 1. Gọi phương thức GetRooms trong tầng DAL lấy ra 1 List<Room>. 2. Viết một query để lấy một Room ngẫu nhiên trong Database. 3. Viết một query để lấy Room Top trong Database. 4. Viết một query để lấy Room Bottom trong Database |

|  |  |
| --- | --- |
| **ID** | Test\_Case 3.2 |
| **Tên kiểm thử** | TestGetRoomByID |
| **Mô tả kiểm thử** | Kiểm thử phương thức GetRoomById() trong DAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương thức GetRoomById() trong DAL. |
| **Đầu vào kiểm thử** | 1. GetRoomById(1); 2. GetRoomById(0) |
| **Đầu ra kiểm thử** | 1. Trả về Room có ID = 1 ( NOT NULL ) 2. Trả về NULL (Không tìm thấy room\_id = 0 trong Database) |
| **Các bước kiểm thử** | 1. Nhập ID phòng chiếu. 2. Tìm kiếm Room\_ID đấy có trong cơ sở dữ liệu hay không ? bằng phương thức GetRoomByID(Id) ; |

#### - Kiểm thử Class Schedule

|  |  |
| --- | --- |
| **ID** | Test\_Case 4.1 |
| **Tên kiểm thử** | TestGetSchedules |
| **Mô tả kiểm thử** | Kiểm thử phương thức GetSchedules trong tầng DAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu, đã viết phương thức GetSchedules trong tầng DAL. |
| **Đầu vào kiểu thử** | 1. Lấy ra một List<Schedule> trong Database. 2. Lấy ra một Schedule ngẫu nhiên trong Database. 3. Lấy ra vị trí Top của Schedule trong Database. 4. Lấy ra Schedule Bottom trong Database |
| **Đầu ra kiểm thử** | 1. Trả về một List<Schedule> (NotNull). 2. Trả về một Schedule Rand (NotNull).Kiểm tra Schedule Rand trong Database có trong List<Schedule> hay không ? (Contains). 3. Trả về Schedule Top (NotNull). So sánh Schedule Top trong Database có giống Schedule đầu tiên trong List<Schedule> vừa lấy ?(True). 4. Trả về Schedule Bottom (NotNull ). So sánh Schedule Bottom trong Database có giống Schedule cuối cùng trong List<Schedule> ? (True). |
| **Các bước kiểm thử** | 1. Gọi hàm GetSchedules trong tầng DAL lấy ra 1 List<Schedule>. 2. Viết một query để lấy một Schedule ngẫu nhiên trong Database. 3. Viết một query để lấy Schedule Top trong Database. 4. Viết một query để lấy Schedule Bottom trong Database |

|  |  |
| --- | --- |
| **ID** | Test\_Case 4.2 |
| **Tên kiểm thử** | TestGetScheduleByIDSchedule |
| **Mô tả kiểm thử** | Kiểm tra phương thức GetScheduleByIDSchedule trong Class ScheduleDAL. |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương thức GetScheduleByIDSchedule () trong DAL. |
| **Đầu vào kiểu thử** | 1. GetScheduleByIdSchedule (1 ); 2. GetScheduleByIdSchedule (0); |
| **Đầu ra kiểm thử** | 1. Trả về Schedule có ID bằng 1 ( Not Null ) 2. Trả về Null, không tìm được ID bằng 0 trong Database |
| **Các bước kiểm thử** | 1. Nhập ID của lịch chiếu |

|  |  |
| --- | --- |
| **ID** | Test\_Case 4.3 |
| **Tên kiểm thử** | TestGetScheduleByMovieID |
| **Mô tả kiểm thử** | Kiểm tra phương thức GetScheduleByMovieID trong Class ScheduleDAL. |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương thức GetScheduleByMovieID () trong DAL. |
| **Đầu vào kiểu thử** | 1. GetScheduleByMovieID (1 ); 2. GetScheduleByMovieID (0); |
| **Đầu ra kiểm thử** | 1. Trả về List<Schedule> có Movie ID bằng 1 ( Not Null ) 2. Trả về Null, không tìm được Movie ID bằng 0 trong Database |
| **Các bước kiểm thử** | 1. Nhập ID của Movie ID |

|  |  |
| --- | --- |
| **ID** | Test\_Case 4.4 |
| **Tên kiểm thử** | Test Get Time By Movie ID and Date |
| **Mô tả kiểm thử** | Kiểm thử phương thức SelectTime() trong Class Shedule\_DAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương thức SelectTime() trong Class Schedule\_DAL. |
| **Đầu vào kiểm thử** | 1. DateTime timeStart = new DateTime(2018,7,20,0,0,0);   SelectTime(1,timeStart.ToString($“{timeStart:yyyy-MM-dd}”));  2. DateTime timeStart = new DateTime(2018,7,20,0,0,0);  SelectTime(1,timeStart.ToString($“{timeStart:dd-yyyy-dd}”));  3. DateTime timeStart = new DateTime(2018,7,20,0,0,0);  SelectTime(0,timeStart.ToString($“{timeStart:dd-yyyy-dd}”));  4. DateTime timeStart = new DateTime(2017,7,20,0,0,0);  SelectTime(1,timeStart.ToString($“{timeStart:yyyy-MM-dd}”));  5. SelectTime(1,””); |
| **Đầu ra kiểm thử** | 1. Trả về một List<Schedule> với Movie\_ID = 1 và DateTime là 2018-7-20 2. Trả về NULL (Nhập sai định dạng DateTime) 3. Trả về NULL (Không có Movie\_id = 0 trong cơ sở dữ liệu) 4. Trả về NULL (Không tìm thấy trong cơ sở dữ liệu bảng Schedule có movie\_ id = 1 và Datetime là 2017-7-20) 5. Trả về NULL không được để Datetime = null. |
| **Các bước kiểm thử** | 1. Nhập Movie\_id 2. Nhập Datetime |

|  |  |
| --- | --- |
| **ID** | Test\_Case 4.5 |
| **Tên kiểm thử** | TestSelectScheduleByMovieIDAndDateTime |
| **Mô tả kiểm thử** | Kiểm tra phương thức SelectScheduleByDateTimeAndMovieID trong Class ScheduleDAL. |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương thức SelectScheduleByDateTimeAndMovieID () trong DAL. |
| **Đầu vào kiểu thử** | 1. DateTime ShowDate = new DateTime(2018,7,20,0,0,0);   TimeSpan timeSpan = new TimeSpan(8,0,0);  string Date = ShowDate.ToString($“{ShowDate:yyyy-MM-dd};  string Time = string.Format ($"{timeSpan.Hours:D2}:{timeSpan.Minutes:D2}:  {timeSpan.Seconds:D2}");  • SelectScheduleByDateTimeAndMovieID(1,Date,Time);  2. DateTime ShowDate = new ShowDate(2017,7,20,0,0,0);  TimeSpan StartTime = new TimeSpan(1,0,0);  string Date = ShowDate.ToString($“{ShowDate:yyyy-MM-dd};  string Time = string.Format ($"{StartTime.Hours:D2}:{StartTime.Minutes:D2}:  {timeSpan.Seconds:D2}");  • SelectScheduleByDateTimeAndMovieID(1,Date,Time);  3. DateTime ShowDate = new DateTime(2018,7,20,0,0,0);  TimeSpan StartTime = new TimeSpan(8,0,0);  string Date = ShowDate.ToString($“{ShowDate:yyyy-MM-dd};  string Time = string.Format ($"{StartTime.Hours:D2}:{StartTime.Minutes:D2}:  {StartTime.Seconds:D2}");  • SelectScheduleByDateTimeAndMovieID(0,Date,Time);  4. SelectScheduleByDateTimeAndMovieID(1,””,””); |
| **Đầu ra kiểm thử** | 1. Trả về một Schedule có MovieID = 1 , ShowDate = ‘2018-07-20’, start\_time = ‘08:00:00’. (NOT NULL) 2. Trả về NULL không tìm thấy Schedule. 3. Trả về NULL , MovieID không được bằng 0 4. Trả về NULL , ShowDate và Start\_Time không được để NULL. |
| **Các bước kiểm thử** | 1. Nhập MovideID 2. Nhập ShowDate 3. Nhập StartTime |

|  |  |
| --- | --- |
| **ID** | Test\_Case 4.6 |
| **Tên kiểm thử** | TestAddMapSeatsForSchedule |
| **Mô tả kiểm thử** | Kiểm thử phương thức AddMapSeats () trong Class Schedule DAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương thức AddMapSeats () trong Class Schedule DAL |
| **Đầu vào kiểm thử** | 1. Schedules schedule = new Schedules (1, 1, 1, new DateTime (2018, 7, 20), new TimeSpan (21, 0, 0), new TimeSpan (21, 0, 0), "MapSeat", 45000);   string mapSeats = "A B C D F E G H J K L M;10;"  • AddMapSeats(schedule,mapSeats);  2. • AddMapSeats(NULL,NULL);  3. • AddMapSeats(NULL,” ”);  4. Schedules schedule = new Schedules (0, 0, 0, new DateTime  (2018, 7, 20, 0, 0, 0), new TimeSpan (21, 0, 0), new TimeSpan  (21, 0, 0), "", 45000);  • AddMapSeats(schedule,””); |
| **Đầu ra kiểm thử** | 1. Trả về True , Insert Into thành công ghế vào cơ sở dữ liệu. 2. Trả về False, Insert Into không thành công do dữ liệu nhập vào không được NULL. 3. Trả về False, Insert Into không thành công do dữ liệu nhập vào không được NULL và MapSeats phải != “”. 4. Trả về False, Insert Into không thành công do dữ liệu Schdule.Schedule\_ID phải != 0, Schdule.Movie\_id phải != 0, Schedule.Schedule\_room\_seat phải != “”. |
| **Các bước kiểm thử** | 1. Nhập dữ liệu Schedule 2. Nhập một MapSeats. |

#### Kiểm thử Class Reservation

|  |  |
| --- | --- |
| **ID** | Test\_Case 5.1 |
| **Tên kiểm thử** | Test GetReservationByCustomerID |
| **Mô tả kiểm thử** | Kiểm thử phương thức GetReservationByCustomerID() trong Class ReservationDAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương thức GetReservationByCustomerID() trong Class ReservationDAL. |
| **Đầu vào kiểm thử** | 1. GetReservationByCustomerID(NULL). 2. GetReservationByCustomerID(0). |
| **Đầu ra kiểm thử** | 1. Kết quả trả về NULL. 2. Kết quả trả về NULL. |
| **Các bước kiểm thử** | 1. Nhập CustomerID 2. CustomerID là tham số truyền vào GetReservationByCustomerID(). 3. - Trả về NULL nếu CustomerID = NULL hoặc CustomerID = 0 hoặc CustomerID không có trong cơ sở dữ liệu.   - Trả về NOT NULL nếu tìm thầy CustomerID trong bảng Reservation. |

|  |  |
| --- | --- |
| **ID** | Test\_Case 5.2 |
| **Tên kiểm thử** | Test TestInsertIntoReservation() |
| **Mô tả kiểm thử** | Kiểm thử phương thức TestInsertIntoReservation() trong Class ReservationDAL |
| **Tiền điều kiện** | Có dữ liệu trong cơ sở dữ liệu , đã khai báo phương thức TestInsertIntoReservation() trong Class ReservationDAL. |
| **Đầu vào kiểm thử** | 1. Reservation reservation = new Reservation (1, 1, 1, "A1 A2 A3 A4 A5", 1234567,225000.0, DateTime.Now);   • TestInsertIntoReservation(reservation);  2. Reservation reservation = new Reservation (0, 0, 0,  "A1 A2 A3 A4 A5", 1234567,225000.0, DateTime.Now);  • TestInsertIntoReservation(reservation);  3. Reservation reservation = new Reservation (1, 1, 1,  NULL, 1234567,225000.0, DateTime.Now);  • TestInsertIntoReservation(reservation);  4. Reservation reservation = new Reservation (1, 1, 1,  "", 1234567,225000.0, DateTime.Now);  • TestInsertIntoReservation(reservation); |
| **Đầu ra kiểm thử** | 1. Trả về True , Khách hàng đặt vé thành công. 2. Trả về False, Đặt vé không thành công do không có CustomerID và ScheduleID. 3. Trả về False, Đặt vé không thành công do ghế = NULL. 4. Trả về False, Đặt vé không thành công do ghế = “”. |
| **Các bước kiểm thử** | 1. Nhập đầy đủ dữ liệu Reservation |

### 4.2 Kiểm thử tầng BL

|  |  |
| --- | --- |
| **ID** | Test\_CaseBL 1 |
| **Tên kiểm thử** | TestLoginCinema |
| **Mô tả kiểm thử** | Kiểm thử phương thức Login() trong Class LoginCinema (Email, Password) |
| **Tiền điều kiện** | Đã khai báo phương thức LoginCinema() trong tầng DAL. |
| **Đầu vào kiểm thử** | 1. LoginCinema(“”"", "”) 2. LoginCinema(“‘’’’@#$%^”,”...121233”); 3. LoginCinema(“Manh Dat”,”123456”); |
| **Đầu ra kiểm thử** | 1. Input đầu vào không hợp lệ, không được rỗng. 2. Input đầu vào không hợp lệ, không được chứa ký tự đặc biệt. 3. Input đầu vào hợp lệ, cho vào tầng DAL. |

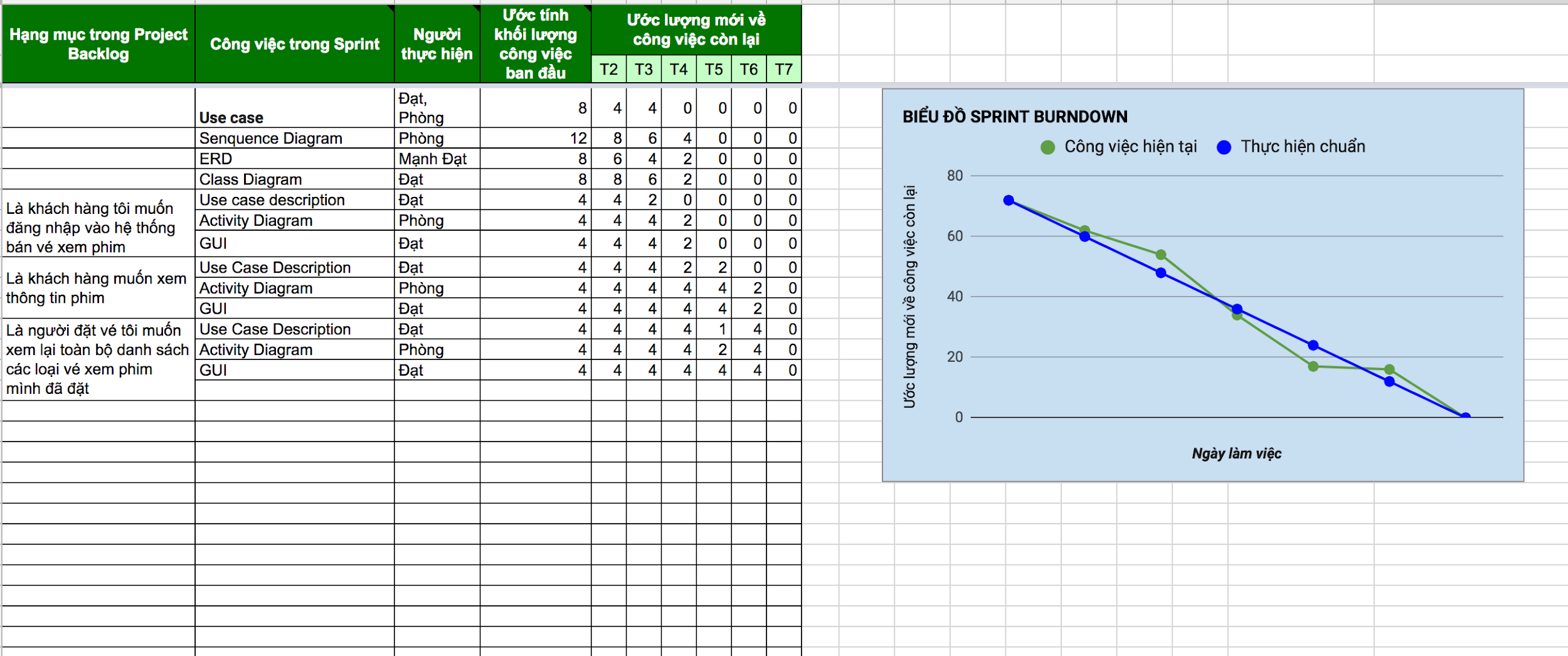
|  |  |
| --- | --- |
| **Tên kiểm thử** | TestScheduleTimeByDateAndMovieID |
| **Mô tả kiểm thử** | Kiểm thử phương thức SelectTimeBy |
| **Tiền điều kiện** | Đã khai báo phương thức SelectTimeBy() trong tầng BL. |
| **Đầu vào kiểm thử** | 1. SelectTimeBy(1,’121212’,’1111’); 2. SelectTimeBy(1,”2018-07-26”,”08:00:00”) 3. SelectTimeBy(1,”26-7-2018”,”8:0:0”) |
| **Đầu ra kiểm thử** | 1. Đầu vào không hợp lệ, không đúng định dạng ngày 2. Đầu vào hợp lệ, chuyển sang tầng DAL 3. Đầu vào không hợp lệ, không đúng định dạng giờ. |
| **Các bước kiểm thử** | 1. Nhập Movie ID, Nhập ngày, Nhập giờ 2. Kiểm tra định dạng   Date = @"(?<year>\d{2,4})-(?<month>\d{1,2})-(?<day>\d{1,2})";  Time = @"^(\d{1,2}|\d\.\d{2}):([0-5]\d):([0-5]\d)(\.\d+)?$"  3.  Trả về Matches nếu đúng định dạng, Cho sang tầng DAL  Trả về DoesNotMatch nếu sai định dạng. |

# Phân công công việc cho từng thành viên trong nhóm

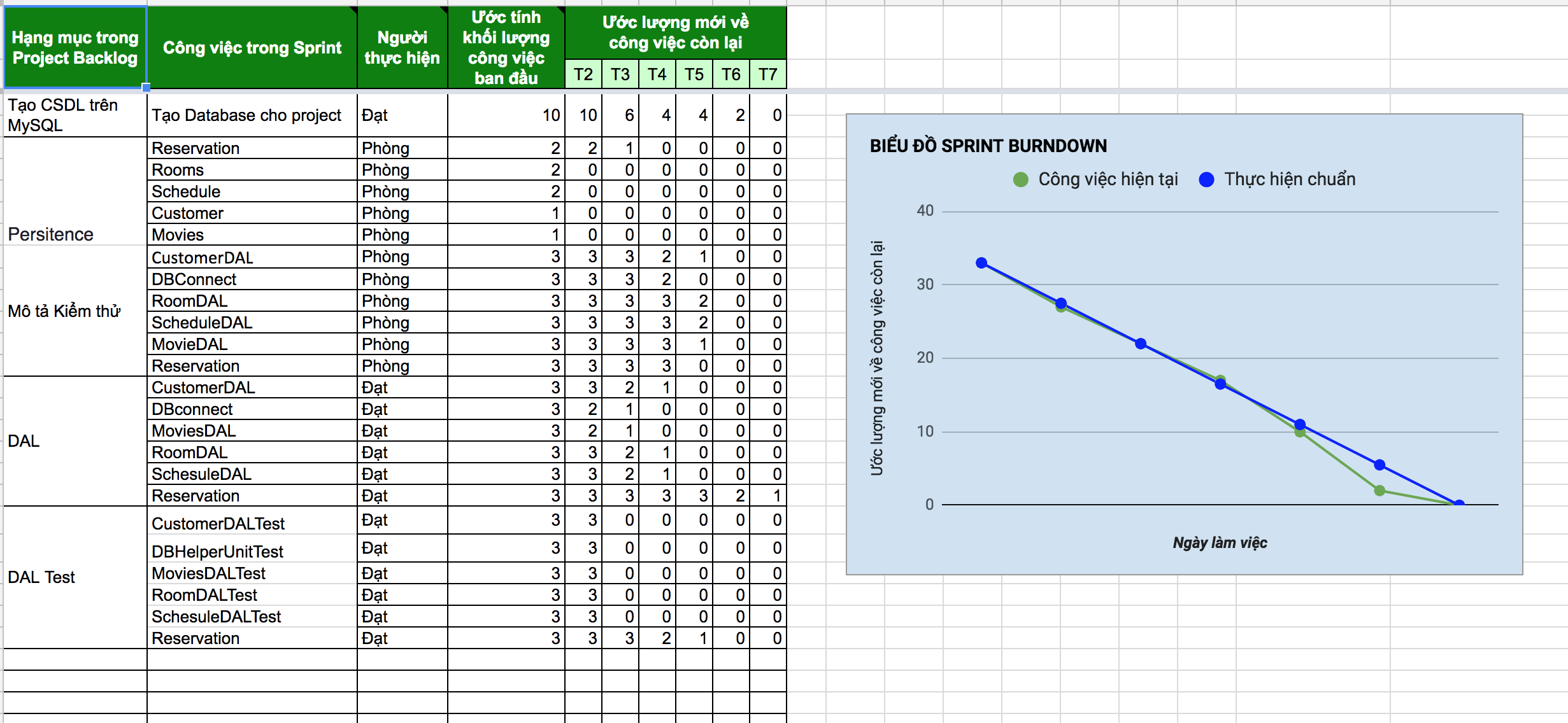
Product Backlog

# 

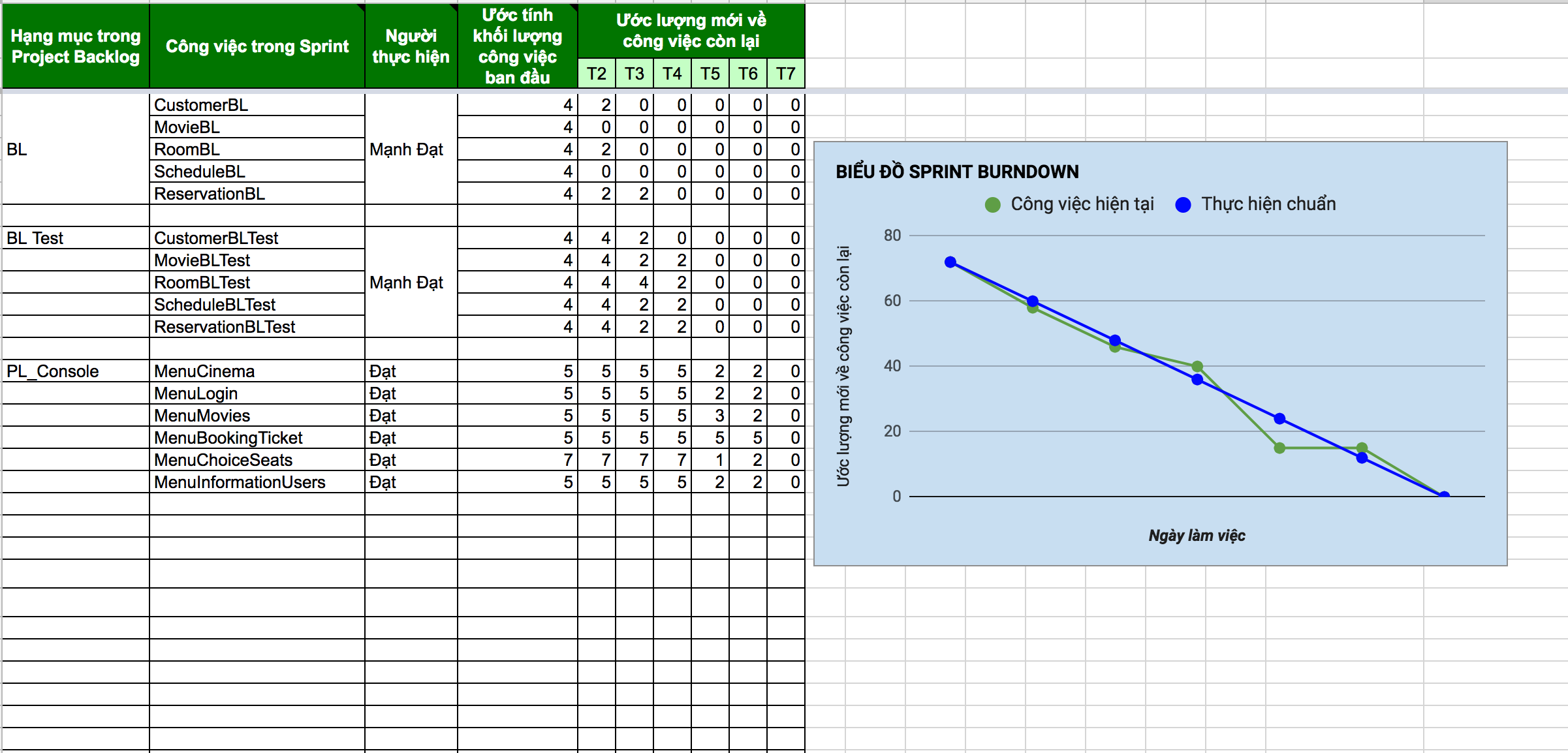
Sprint 1:



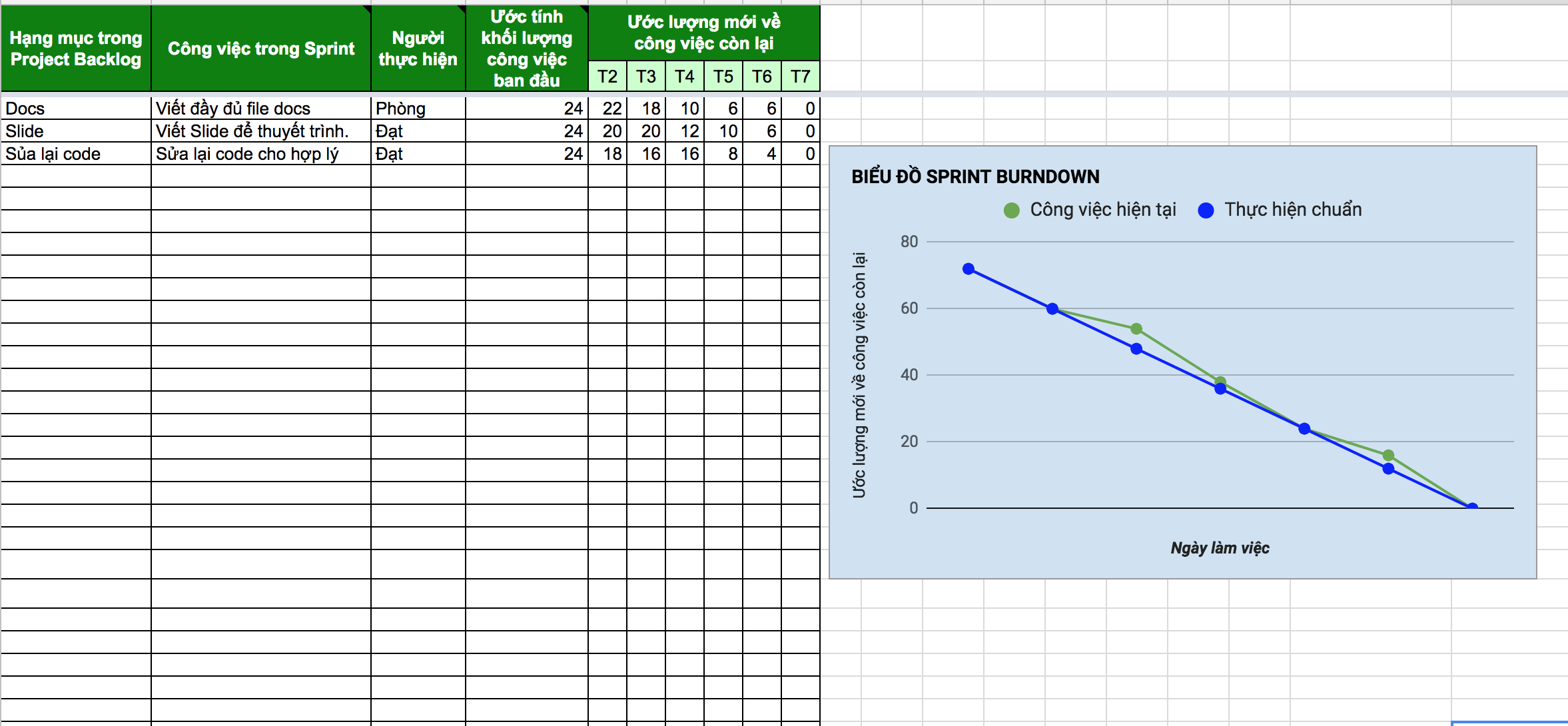
Sprint 2:



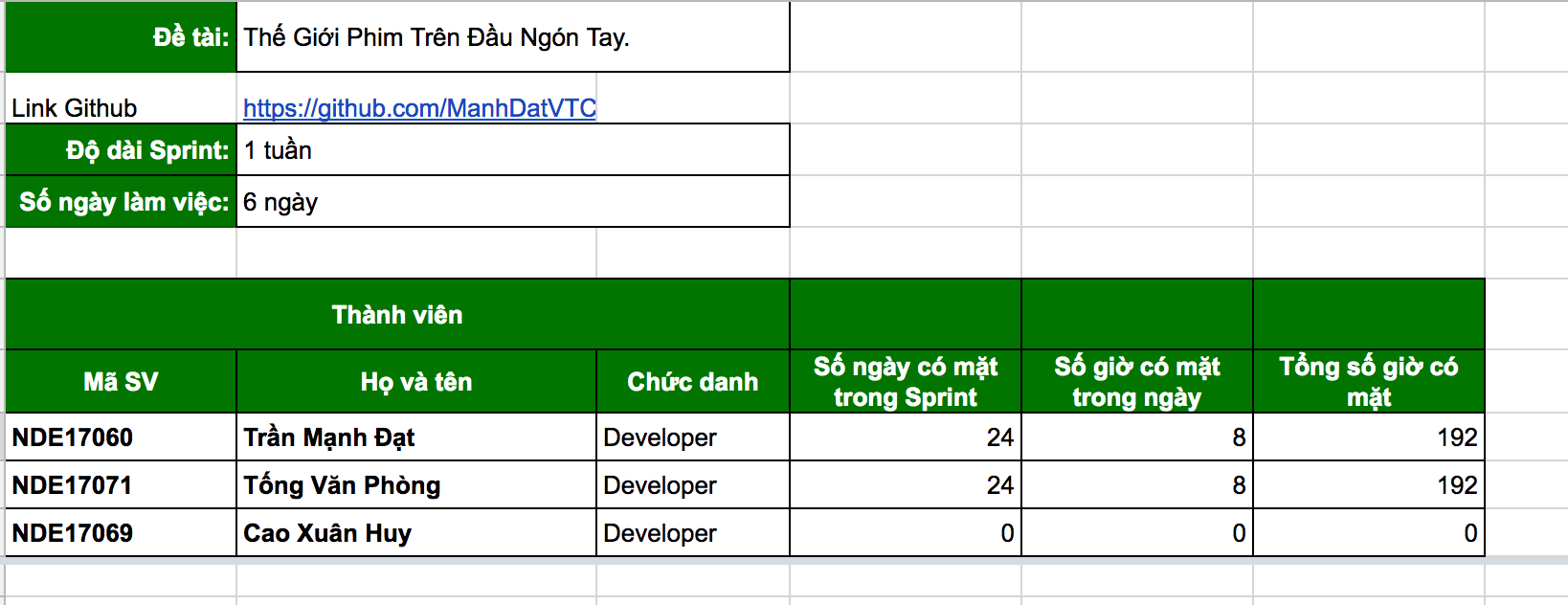
Sprint 3:



Sprint 4:



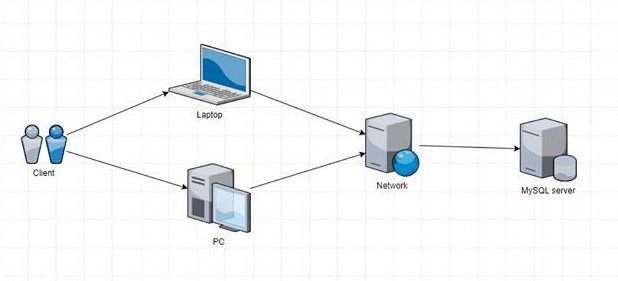
Project- Info :



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nhóm dự án:**  **01** | **Tên dự án: Dự án mua vé xem phim online** | | | | | |
| **STT** | **Tên công việc** | **Mô tả** | **Ngày bắt đầu** | **Ngày kết thúc** | **Thành viên liên quan** | **Tự đánh giá kết quả** |
| 1 | Sprint 1 | 1.Use case diagram  2. Activity diagram  3.Sequence diagram  4.Thiết kế database | 7/6 | 14/6 | Mạnh Đạt , Phòng | Chưa hoàn thành hết được yêu cầu đề ra từ đầu. |
| 2 | Sprint 2 | 1. Hoàn thành những công việc (chưa xong) trong Sprint 1  2. Chuyển hết code lên GitHub và đưa thông tin vào sheet 'Project Info'  3. Tạo CSDL trên MySQL Workbenck  4. Hoàn thành Code của tầng Persistence và DAL | 3/7 | 10/7 | Mạnh Đạt , Phòng | Hoàn thành tốt yêu cầu được đưa ra từ đầu |
| 3 | Sprint 3 | 1.Hoàn thành code của tầng BL 2.Hoàn thành code của tầng BL\_Test 3.Hoàn thành code của tầng PL Console | 11/7 | 18/7 | Đạt | Hoàn thành tốt yêu cầu |
| 4 | Sprint 4 | 1.Viết đầy đủ files Docs 2.Viết siles để thuyết trình 3.Sửa lại code, tối ưu code cho hợp lý | 7/7 | 23/7 | Đạt Phòng | Hoàn thành các yêu cầu. |

# Hướng dẫn cài đặt

#### Biểu đồ triển khai (Deployment Diagram)



#### Các bước cài đặt.

* Bước 1 : Cài MySQL theo đường link <https://www.mysql.com/products/workbench/>
* Chọn download -> Community -> MySql Community Server -> Chọn phiên bản phù hợp với hệ điều hành.
* Bước 2 : Cài đặt .NET Core
* Vào đường link : <https://www.microsoft.com/net/learn/get-started-with-dotnet-tutorial>
* Chọn Download .NET SDK
* Bật lên và cài đặt.
* Bước 3 : Download dự án về máy : <https://drive.google.com/open?id=1ksH2-t5XlVJtpLGZreB-87f5aDbgv_6h>
* Bước 4 : Cài đặt CSDL
* Trong quá trình cài đặt MySQL , bạn đã tạo ra mật khẩu ngẫu nhiên tạm thời.
* Mở Cmd với Windows hoặc Terminal với MacOs
* Khởi chạy MySQL

Hệ điều hành Windows "C:\Program Files\MySQL\MySQL Server 8.0\bin\mysqld"

Hệ điều hành MacOS **:** sudo /usr/local/mysql/support-files/mysql.server start

* Đăng nhập vào MySQL : mysql -u root -p , nhập mật khẩu.
* Chạy và lưu cơ sở dữ liệu vào máy : Vào forder ./Database trong dự án đã tải về ở bước 3 chạy MovieEngagement.sql
* Bước 5 : Chạy ứng dụng
* Mở file ConnectionString.txt

server=localhost;user id=root;password=123456;port=3306;database=MovieEngagement;SslMode=none

Thay đổi password = mật khẩu Mysql của bạn

* Mở thư mục osx.10.12-x64 -> chạy dự án trên terminal (MacOS)

Mở thư mục win10x86 -> chạy dự án trên Cmd (Windows)

Nếu ứng dụng lên thì bạn đã cài đặt thành công !

# Phụ lục

***Các thuật ngữ và từ viết tắt***

*- CSDL : Cơ sở dữ liệu  
- DAL : Data Access Layer*

*- BL : Business Logic Layer*

***Tài liệu tham khảo :***

* [***https://msdn.microsoft.com/en-us***](https://msdn.microsoft.com/en-us)
* [***https://learn.vtc.edu.vn/***](https://learn.vtc.edu.vn/)
* [***https://www.w3schools.com/sql/***](https://www.w3schools.com/sql/)
* [***https://stackoverflow.com/questions/28584634/cinema-booking-application***](https://stackoverflow.com/questions/28584634/cinema-booking-application)
* [***https://lottecinemavn.com/en-us/default.aspx***](https://lottecinemavn.com/en-us/default.aspx)

### Tổng Kết

#### Kết quả thu được.

* **Học được trách nhiệm khi làm việc nhóm.**
* **Học được quy trình làm việc của dự án thật.**
* **Hiểu kĩ và sâu hơn về C# (C Sharp) .**
* **Vẽ sơ đồ Activity Diagram, Sequence Diagram, Class Diagram.**
* **Viết Test Case theo quy trình TDD (Test Driven Development) tăng hiệu suất công việc, giảm thiểu lỗi.**
* **Thiết kế cơ sở dữ liệu chính xác, phù hợp , tối ưu , tiết kiệm dữ liệu.**
* **Kỹ năng viết báo cáo, làm slide.**
* **Tăng khả năng tự học của bản thân.**

#### Hạn chế .

* **Lần đầu làm việc nhóm nên các thành viên chưa liên kết được với nhau nhiều.**
* **Chưa giải quyết được toàn bộ yêu cầu lúc đầu đặt ra**
* **Chưa hiểu hết về nghiệp vụ yêu cầu bài toán, không lường trước được những vấn đề phát sinh khi viết code và phân tích cơ sở dữ liệu .**