

|  |
| --- |
|  |

|  |
| --- |
| **FPT POLYTECHNIC** |
| **DỰ ÁN 1** |
| **NGÀNH CÔNG NGHỆ THÔNG TIN (ỨNG DỤNG PHẦN MỀM) – PRO1041** |

GV hướng dẫn: Thầy **Lê Văn Phụng**

Lớp IT17101

Kỳ Spring 2022

Thành viên nhóm:

**(Trưởng nhóm)**

**Trần Hoàng Thúy Vân – PS19229**

**Đỗ Phi Hùng – PS19239**

**Cao Hoài Bảo Ngọc – PS19105**

**Nhóm 3**

MỤC LỤC

[1 Phân tích 3](#_Toc101387519)

[1.1 Hiện trạng 3](#_Toc101387520)

[1.2 Yêu cầu hệ thống 3](#_Toc101387534)

[1.3 Use case 4](#_Toc101387535)

[2 Thiết kế 7](#_Toc101387536)

[2.1 Mô hình triển khai 7](#_Toc101387537)

[2.2 Thiết kế CSDL 8](#_Toc101387538)

[2.2.1 Sơ đồ quan hệ thực thể 8](#_Toc101387539)

[2.2.2 Thiết kế chi tiết các thực thể 10](#_Toc101387540)

[2.3 Thiết kế giao diện 16](#_Toc101387541)

[2.3.1 Sơ đồ tổ chức giao diện 21](#_Toc101387542)

[2.3.2 Thiết kế giao diện cho các chức năng nghiệp vụ 22](#_Toc101387543)

[3 Thực hiện viết mã 51](#_Toc101387544)

[3.1 Viết mã tạo CSDL 51](#_Toc101387545)

[3.1.1 Tạo CSDL 51](#_Toc101387546)

[3.1.2 SQL truy vấn và thao tác 56](#_Toc101387547)

[3.1.3 Các thủ tục lưu, trigger, function 63](#_Toc101387548)

[3.2 Lập trình JDBC 96](#_Toc101387549)

[3.2.1 Lớp hỗ trợ 97](#_Toc101387550)

[3.2.2 Model class - Các lớp mô tả dữ liệu 133](#_Toc101387551)

[3.2.3 DAO Class - Các lớp truy xuất dữ liệu 157](#_Toc101387552)

[4 Kiểm thử 210](#_Toc101387553)

[4.1 Kiểm thử form LoginFrame 210](#_Toc101387554)

[4.2 Kiểm thử form qLDoAnKemFrame 210](#_Toc101387555)

[4.3 Kiểm thử form qldonthanhtoanframe 211](#_Toc101387556)

[4.4 Kiểm thử form khachhangFrame 212](#_Toc101387557)

[4.5 Kiểm thử form qlnhanvienframe 213](#_Toc101387558)

[4.6 Kiểm thử form qlphimframe 217](#_Toc101387559)

[4.7 Kiểm thử form qlprofileframe 217](#_Toc101387560)

[4.8 Kiểm lỗi tracuulichchieuframe 219](#_Toc101387561)

[4.9 Kiểm lỗi thongkeframe 220](#_Toc101387562)

[5 Đóng gói và triển khai 220](#_Toc101387563)

[5.1 Hướng dẫn chuyển đổi jar thành exe 220](#_Toc101387564)

[5.2 Hướng dẫn cài đặt triển khai 221](#_Toc101387565)

[5.3 Hướng dẫn sử dụng phần mềm 221](#_Toc101387566)

# Phân tích

## Hiện trạng

## Trong các rạp chiếu phim hiện nay, việc quản lý và tổ chức chiếu phim, bán vé luôn là một vấn đề được quan tâm. Người quản lý luôn gặp khó khăn trong vấn đề quản lý quá trình hoạt động của từng bộ phân như: phim, lịch chiếu,… đặc biệt là quá trình bán vé. Khách hàng phải xếp hàng chờ đợi để mua được vé. Chính vì lý do đó mà số lượng khách hàng đi xem phim cũng ít hơn, thu nhập của rạp cũng bị ảnh hưởng. Vì vậy, tạo ra một phần mềm quản lý rạp chiếu phim là một nhu cầu tất yếu.

## Hệ thống quản lý rạp chiếu phim được xây dựng trên những nhu cầu thực tế của khách hàng và nhà quản lý nhằm giải quyết những khó khăn gặp phải, giảm rủi ro trong quá trình quản lý rạp. Hệ thống hướng tới đối tượng nhân viên quản lý rạp, cũng như nhân viên bán hàng với giao diện thân thiện, dễ dàng tương tác.

## Các chức năng chính bao gồm:

## - Quản lý phim.

## - Quản lý, tra cứu lịch chiếu.

## - Quản lý suất chiếu.

## - Quản lý vé, đặt vé.

## - Quản lý đồ ăn, đặt đồ ăn.

## - Quản lý đơn thanh toán.

## - Quản lý thông tin khách hàng.

## - Quản lý nhân viên.

## - Thống kê báo cáo.

## Các chức năng này giúp người quản lý dễ dàng điều khiển quá trình hoạt động của rạp và rất thuận tiện đẻ khách hàng có thể mua được vé.

## Yêu cầu hệ thống

Rạp chiếu phim mong muốn xây dựng một phần mềm quản lý gồm các chức năng cơ bản :

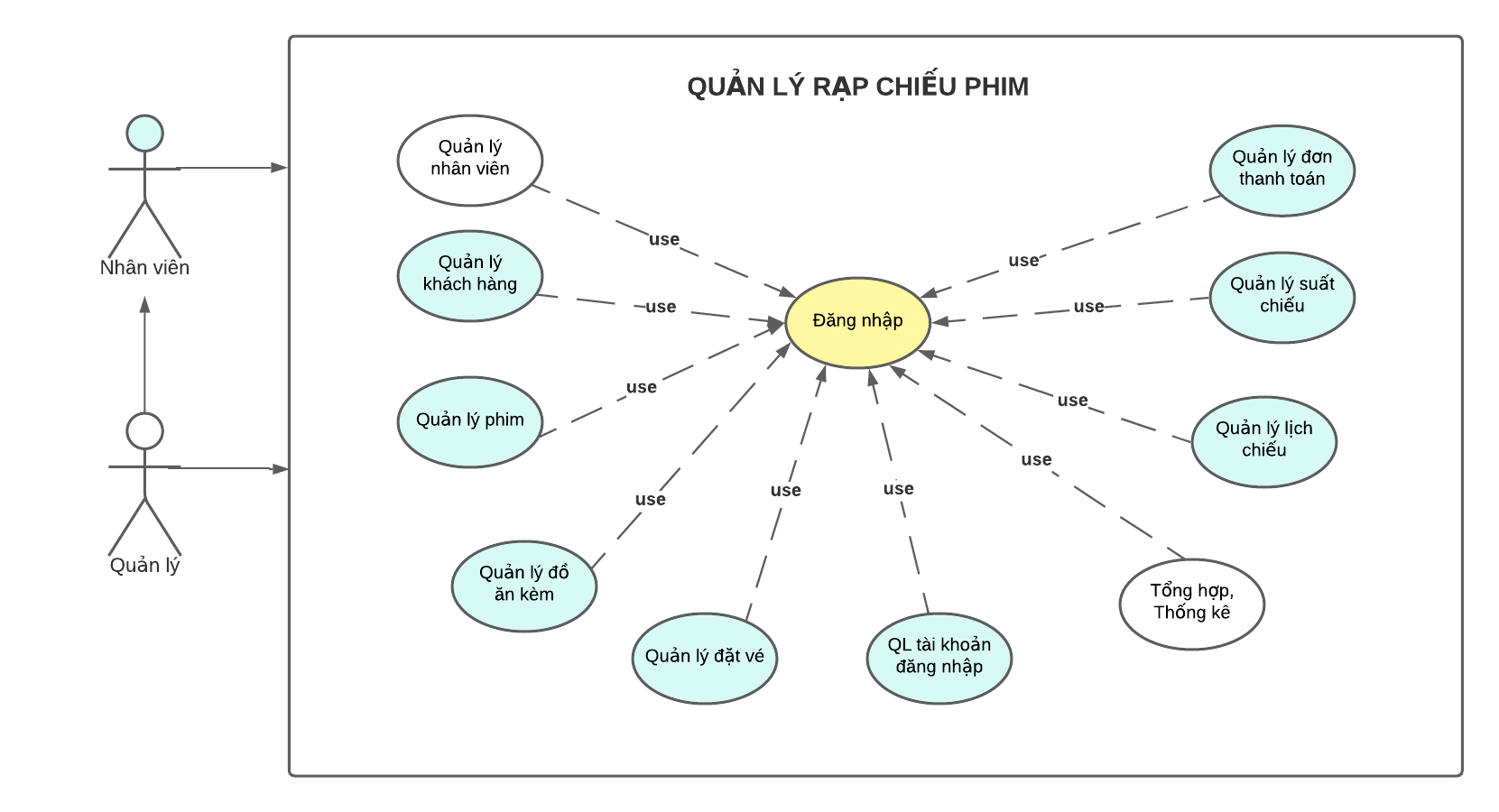
* **Yêu cầu chức năng nghiệp vụ**
  + Quản lý phim.
  + Quản lý suất chiếu (với phim, phòng).
  + Tra cứu lịch chiếu.
  + Quản lý thành viên.
  + Quản lý đơn thanh toán.
  + Quản lý đồ ăn, đặt đồ ăn
  + Quản lý vé, đặt vé.
  + Quản lý nhân viên.
  + Thống hợp – tổng kê.
  + Quản lý tài khoản (đổi mật khẩu, đổi thông tin người dùng).
* **Yêu cầu về bảo mật**
  + Tất cả mọi thành viên phải đăng nhập mới sử dụng được phần mềm.
  + Quản lý được thực hiện tất cả các chức năng.
  + Nhân viên không được phép truy cập chức năng quản lý nhân viên và thống kê.
* **Yêu cầu về môi trường công nghệ**
  + Ứng dụng phải được thực với công nghệ Swing và JDBC chạy trên mọi hệ điều hành với môi trường JDK tối thiểu 1.8
  + Hệ quản trị CSDL SQL Server 2008 trở lên

**PHẠM VI CỦA ĐỀ TÀI:** Toàn bộ chức năng được nêu trong mô tả nghiệp vụ trên.

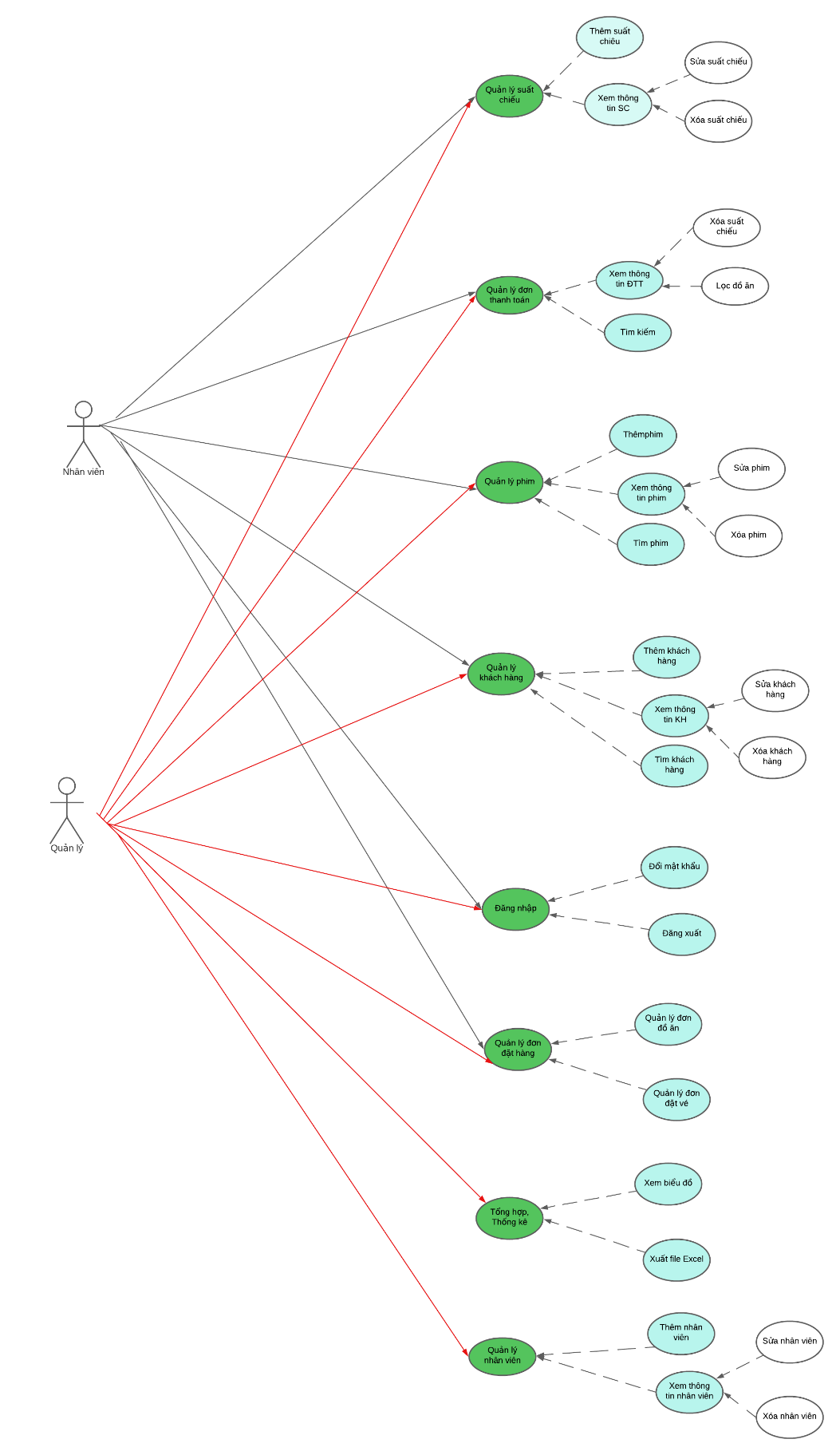
## Use case

Use case là sơ đồ tổng quan về mặt chức năng và phân vai trò người sử dụng. Dựa vào yêu cầu hệ thống của khách hàng, chúng ta có thể phác thảo sơ đồ use case như sau.

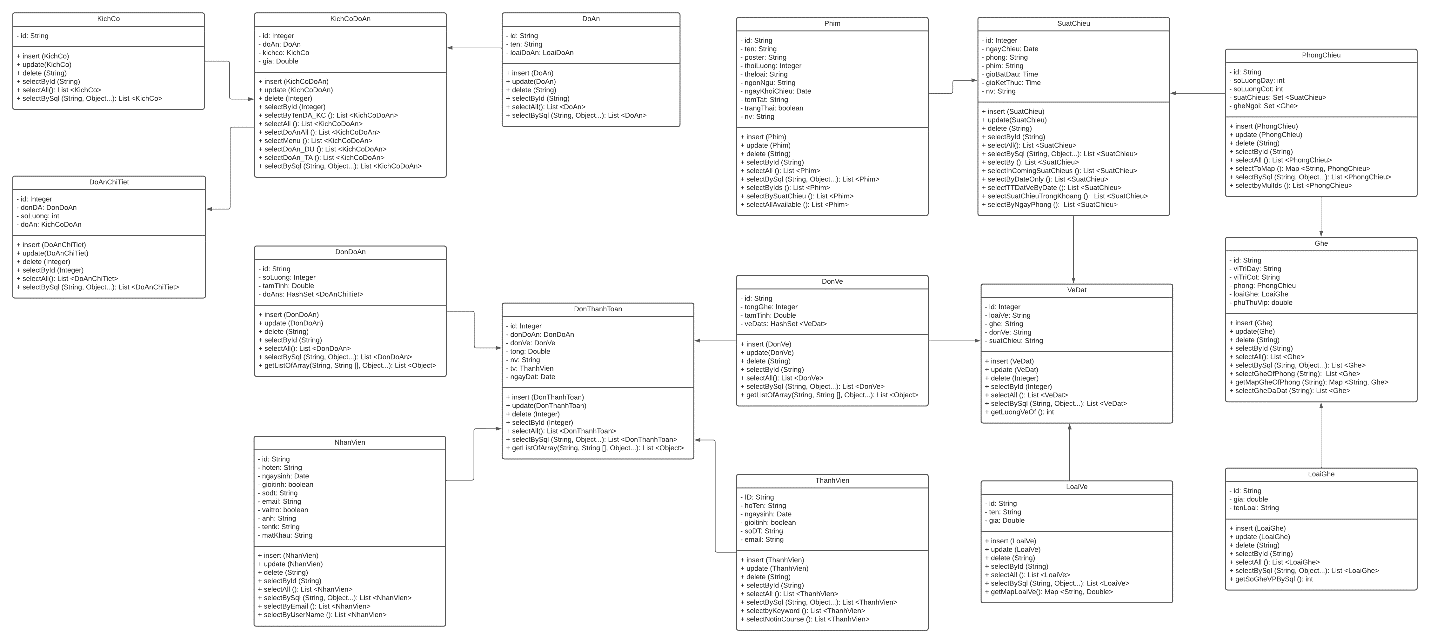
**Sơ đồ use case tổng quát**

****

**Sơ đồ uscase chi tiết**

****

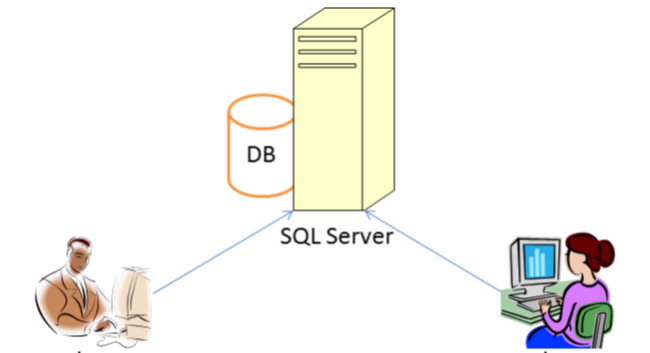
**Sơ đồ use case class diagram**



# Thiết kế

## Mô hình triển khai

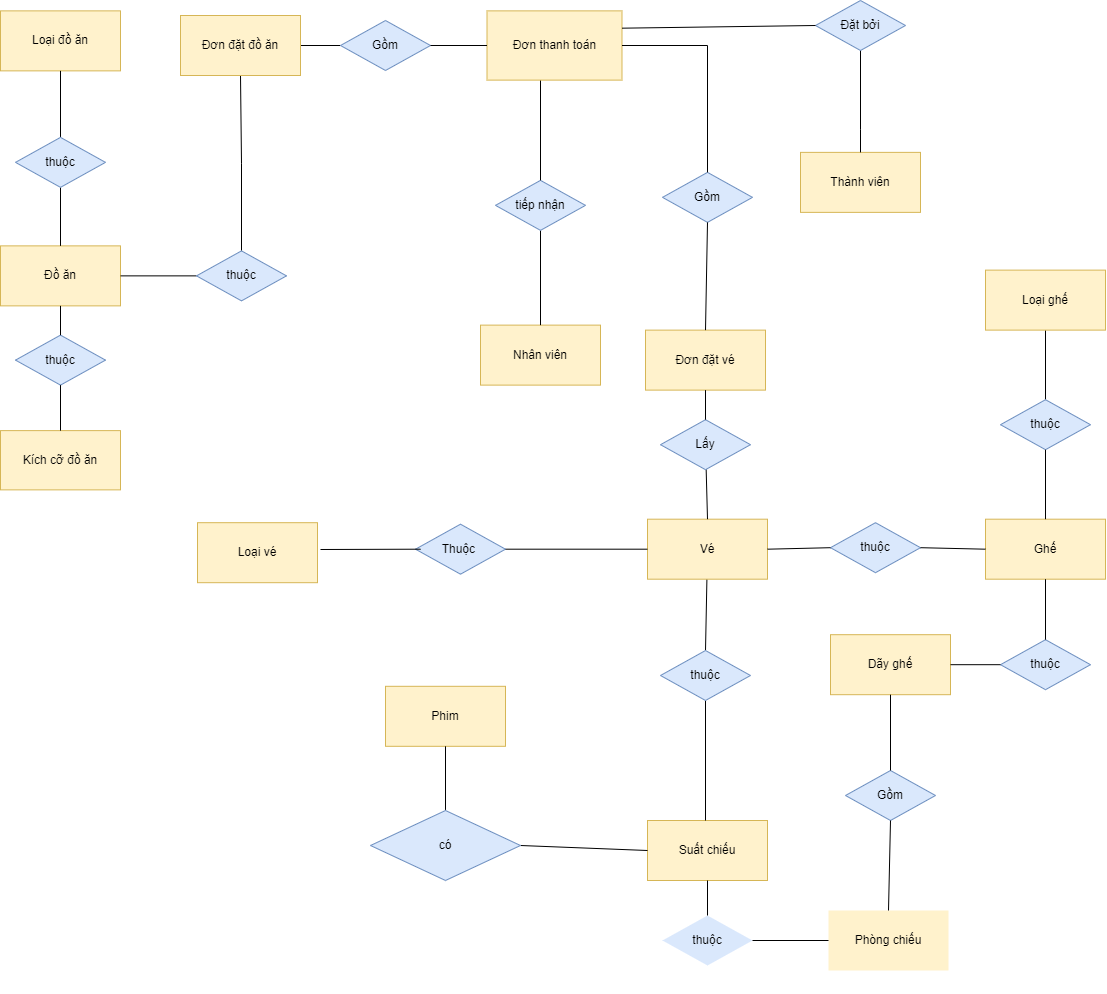
Ứng dụng phần mềm được xây dựng để phục vụ cho nhiều người dùng nhưng cơ sở dữ liệu thì lưu trữ tập trung.



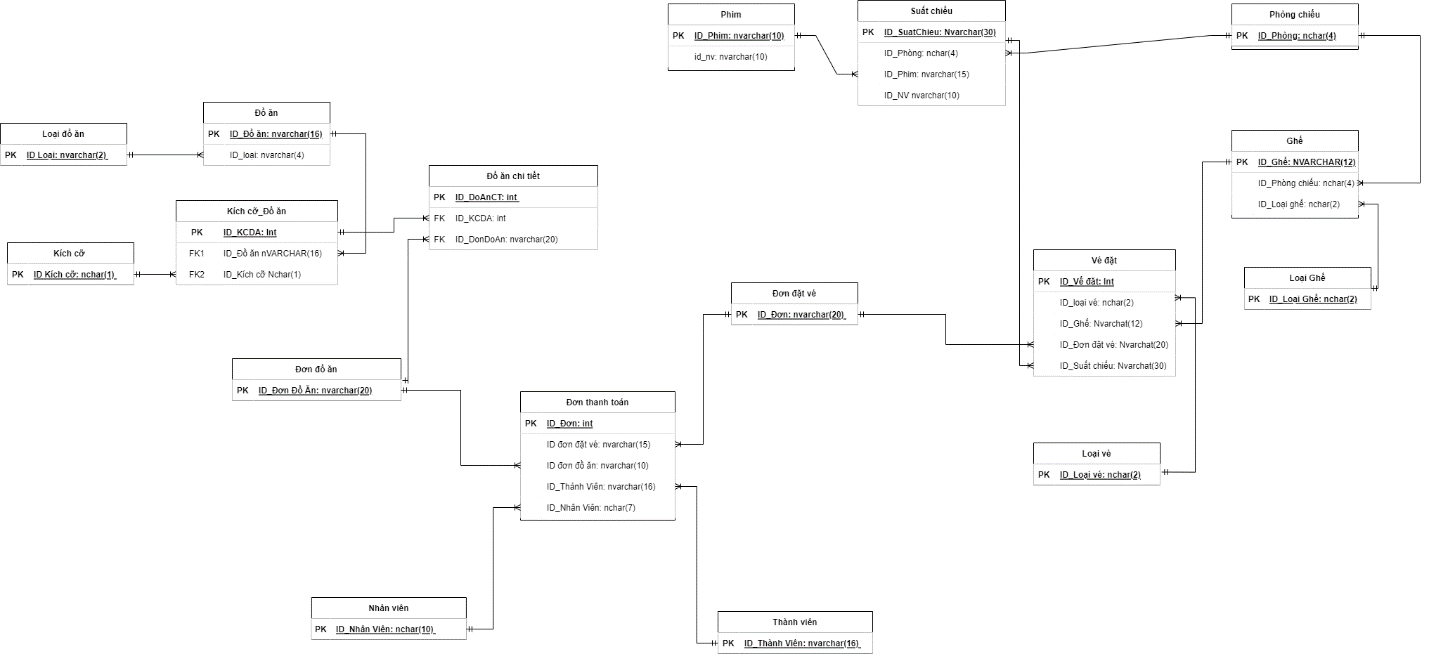
* Cần một máy cài SQL Server 2008+. Máy này cần hệ điều hành window xp trở lên.
* Các máy nhân viên cần JDK 1.8+ với hệ điều hành bất kỳ.

## Thiết kế CSDL

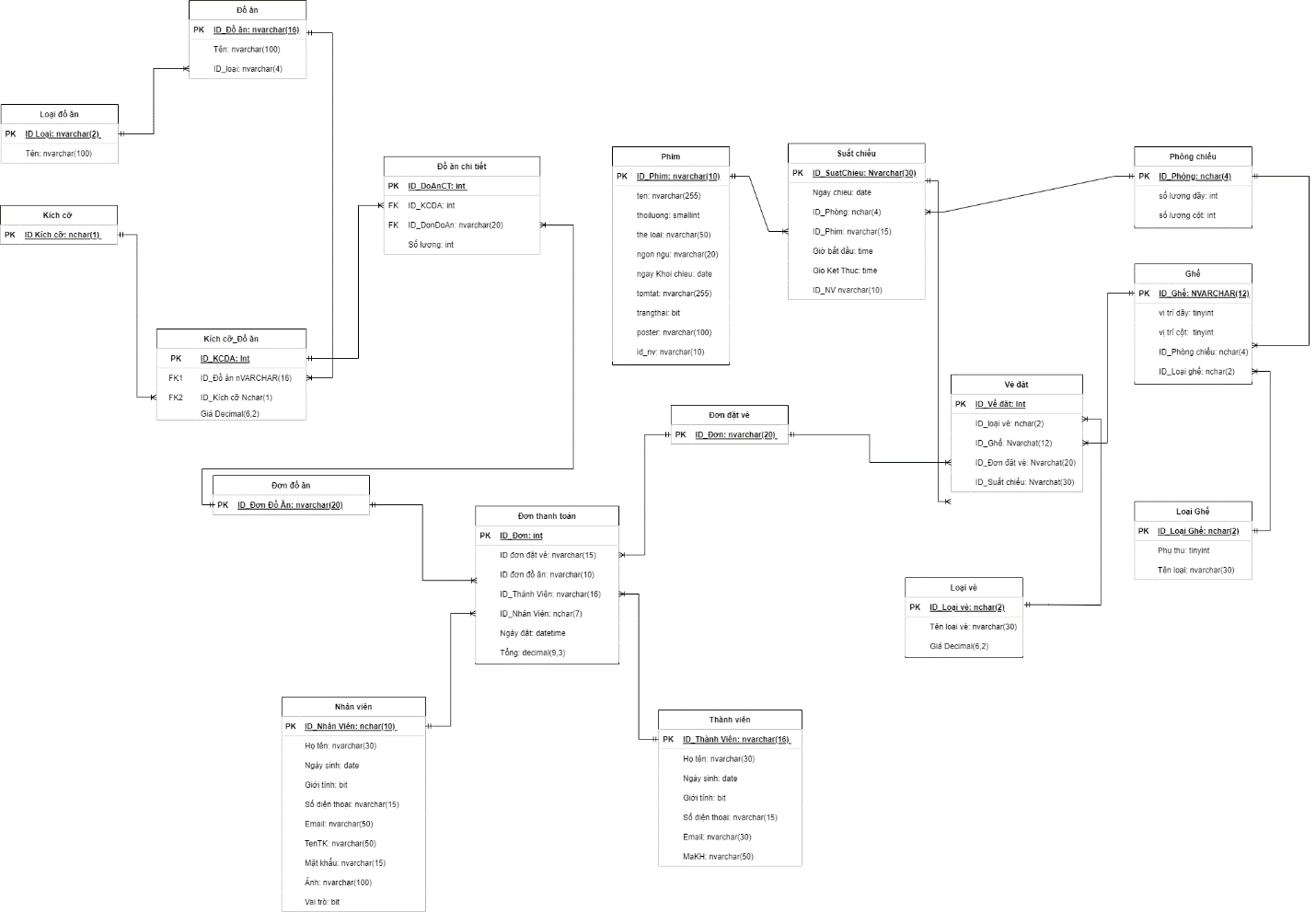
### Sơ đồ quan hệ thực thể



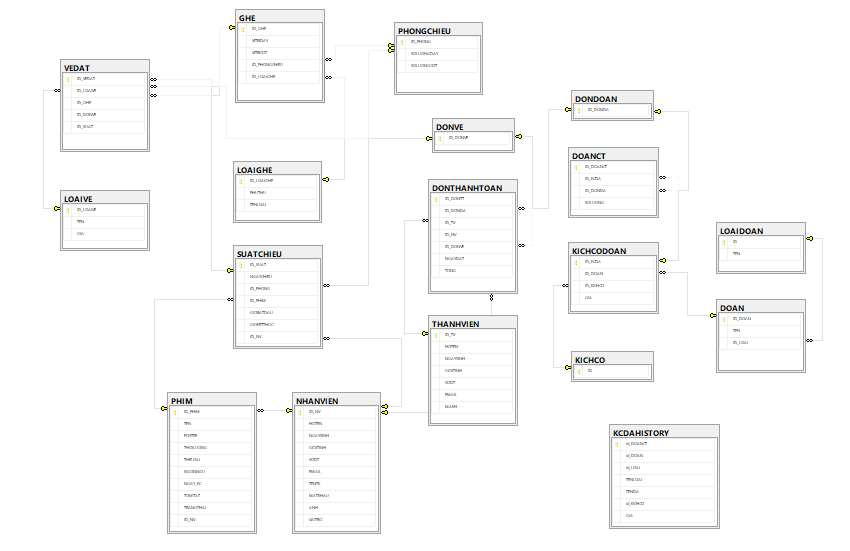
#### ERD Diagram level 1



#### ERD Diagram level 2



#### Database Diagram



### Thiết kế chi tiết các thực thể

* + - Các bảng có khỏa chính tự tăng viết bằng trigger: Nhân viên, thành viên, đồ ăn, đơn đồ ăn, đơn vé, phim, phòng chiếu, ghế

#### Nhân viên

Bảng NHANVIEN lưu thông tin nhân viên có cấu trúc như sau:

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_NV | NVARCHAR(10) | PK, NOT NULL | Mã nhân viên |
| HOTEN | NVARCHAR(50) | NOT NULL | Tên nhân viên |
| NGAYSINH | DATE | NOT NULL | Ngày sinh |
| GIOITINH | BIT | NOT NULL | Giới tính |
| SODT | NVARCHAR(15) | NOT NULL | Số điện thoại |
| EMAIL | NVARCHAR(50) | UNIQUE,NOT NULL | Email |
| TENTK | NVARCHAR(30) | UNIQUE,NOT NULL | Tên tài khoản |
| MATKHAU | NVARCHAR(15) | NOT NULL | Mật Khẩu |
| ANH | NVARCHAR(100) | NOT NULL | Hình ảnh |
| VAITRO | BIT | NOT NULL | Vai trò |

#### Thành viên

Bảng THANHVIEN lưu thông tin các khách hàng có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_TV | NVARCHAR(16) | PK, NOT NULL | Mã khách hàng |
| HOTEN | NVARCHAR(50) | NOT NULL | Tên khách hàng |
| NGAYSINH | DATE | NOT NULL | Ngày Sinh |
| GIOITINH | BIT | NOT NULL | Giới Tính |
| SODT | NVARCHAR(15) | NOT NULL | Số điện thoại |
| EMAIL | NVARCHAR(50) | NOT NULL | Email |
| MAKH | NVARCHAR(12) | NULL | Mã khách hàng |

#### Loại đồ ăn

Bảng LOAIDOAN lưu thông tin loại đồ ăn có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID | NVARCHAR(4) | PK, NOT NULL | Mã loại |
| TEN | NVARCHAR(100) | NOT NULL | Tên loại |

#### Kích cỡ

Bảng KICHCO lưu thông tin kích cỡ đồ ăn có cấu trúc như sau

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên cột** | | **Kiểu** | **Ràng buộc** | | **Ghi chú** |
| ID | NCHAR(1) | | PK, NOT NULL | Kích cỡ đồ ăn | |

#### Đồ ăn

Bảng DOAN lưu thông tin đồ ăn có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_DOAN | NVARCHAR(16) | PK, NOT NULL | Mã đồ ăn |
| TEN | NVARCHAR(100) | NOT NULL | Tên đồ ăn |
| ID\_LOAI | NVARCHAR(4) | FK, NOT NULL | Mã |

#### Kích cỡ đồ ăn

Bảng KICHCODOAN lưu thông tin chi tiết các loại đồ ăn có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_KCDA | INT | PK, Tự tăng | Mã kích cỡ |
| ID\_DOAN | NVARCHAR(16) | FK, NOT NULL | Mã đồ ăn |
| ID\_KICHCO | NCHAR(1) | FK, NOT NULL | Kích cỡ đồ ăn |
| GIA | DECIMAL | NOT NULL | Giá đồ ăn |

#### Đơn đồ ăn

Bảng DONDOAN lưu thông tin đơn có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_DONDA | NVARCHAR(20) | PK, NOT NULL | Mã đơn |

#### Đồ ăn chi tiết

Bảng DOANCT lưu thông tin có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_DOANCT | INT | PK, Tự tăng | Mã đơn |
| ID\_KCDA | INT | FK, NOT NULL | Mã kích cỡ |
| ID\_DONDA | NVARCHAR(20) | FK, NOT NULL | Mã đơn |
| SOLUONG | INT | NOT NULL | Số lượng |

#### Phim

Bảng PHIM lưu thông tin phim có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_PHIM | VARCHAR(10) | PK, NOT NULL | Mã phim |
| TEN | NVARCHAR(255) | FK, NOT NULL | Tên phim |
| POSTER | NVARCHAR(100) | NOT NULL | Poster phim |
| THOILUONG | SMALLINT | NOT NULL | Thời lượng phim |
| THELOAI | NVARCHAR(50) | NOT NULL | Thể loại phim |
| NGONNGU | NVARCHAR(20) | NOT NULL | Ngôn ngữ phim |
| NGAY\_KC | DATE | NOT NULL | Ngày khởi chiếu |
| TOMTAT | NVARCHAR(MAX) | NOT NULL | Tóm tắt phim |
| TINHTRANG | BIT | NOT NULL | Tình trạng phim |
| ID\_NV | NVARCHAR(10) | FK, NOT NULL | Mã nhân viên |

#### Phòng chiếu

Bảng PHONGCHIEU lưu thông tin phòng có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_PHONG | NCHAR(4) | PK, NOT NULL | Mã phòng |
| SOLUONGDAY | INT | NOT NULL | Số lượng dãy |
| SOLUONGCOT | INT | NOT NULL | Số lượng cột |

#### Suất chiếu

Bảng SUATCHIEU lưu thông tin suất chiếu có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_SUAT | NVARCHAR(30) | PK, NOT NULL | Mã suất chiếu |
| NGAYCHIEU | DATE | NOT NULL | Ngày chiếu |
| ID\_PHONG | NCHAR(4) | FK, NOT NULL | Mã phòng |
| ID\_PHIM | NVARCHAR(15) | FK, NOT NULL | Mã phim |
| GIOBATDAU | TIME | NOT NULL | Giờ bắt đầu |
| GIOKETTHUC | TIME | NOT NULL | Giờ kết thúc |
| ID\_NV | NVARCHAR(10) | FK, NOT NUL | Mã nhân viên |

#### Loại ghế

Bảng LOAIGHE lưu thông tin ghế có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_LOAIGHE | NCHAR(2) | PK, NOT NULL | Mã loại ghế |
| TEN | NVARCHAR(30) | NOT NULL | Tên lại ghế |
| GIA | DECIMAL(6,2) | NOT NULL | Giá loại ghế |

#### Ghế

Bảng GHE lưu thông tin ghế có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_GHE | NVARCHAR(12) | PK, NOT NULL | Mã ghế |
| VITRIDAY | TINYINT | NOT NULL | Vị trí dãy |
| VITRICOT | TINYINT | NOT NULL | Vị trí cột |
| ID\_PHONGCHIEU | NCHAR(4) | FK, NOT NULL | Mã phòng chiếu |
| ID\_LOAIGHE | NCHAR(2) | FK, NOT NULL | Mã loại ghế |

#### Loại vé

Bảng LOAIVE lưu thông tin loại vé có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_LOAIVE | NCHAR(2) | PK, NOT NULL | Mã loại vé |
| TEN | NVARCHAR(30) | NOT NULL | Tên vé |
| GIA | DECIMAL(6,2) | NOT NULL | Giá vé |

#### Đơn vé

Bảng DONVE lưu thông tin đơn vé có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_DONVE | NVARCHAR(20) | PK, NOT NULL | Mã đơn vé |

#### Vé đặt

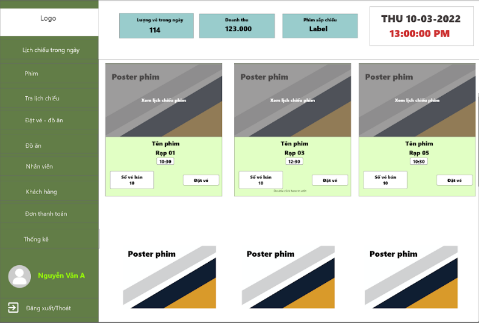
Bảng VEDAT lưu thông tin vé đặt có cấu trúc như sau

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_VEDAT | INT IDENTITY(1,1) | PK, NOT NULL | Mã vé đặt |
| ID\_LOAIVE | NCHAR(2) | FK, NOT NULL | Mã loại vé |
| ID\_GHE | NVARCHAR(12) | FK, NOT NULL | Mã ghế |
| ID\_DONVE | NVARCHAR(20) | FK, NOT NULL | Mã đơn vé |
| ID\_SUAT | NVARCHAR(30) | FK, NOT NULL | Mã suất chiếu |

#### Đơn thanh toán

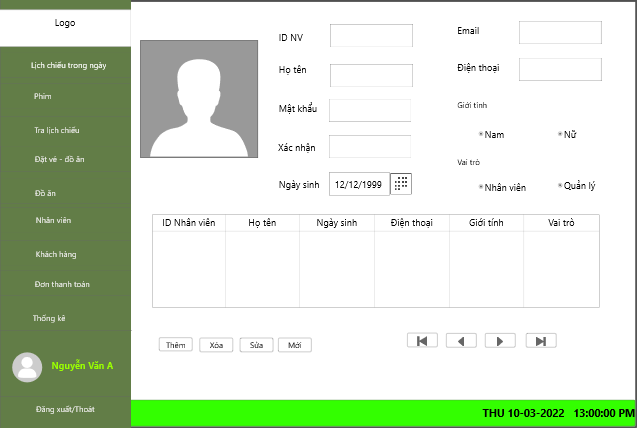
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu** | **Ràng buộc** | **Ghi chú** |
| ID\_DONTT | INT | PK, NOT NULL | Mã đơn thanh toán |
| ID\_DONDA | NVARCHAR(20) | FK, NOT NULL | Mã đơn đồ ăn |
| ID\_TV | NVARCHAR(16) | FK, NOT NULL | Mã thành viên |
| ID\_DONVE | NVARCHAR(20) | FK, NOT NULL | Mã đơn vé |
| NGAYDAT | DATETIME(30) | NOT NULL | Ngày đặt đơn |
| TONG | DECIMAL(9,3) | NOT NULL | Tổng tiền |

## Thiết kế giao diện



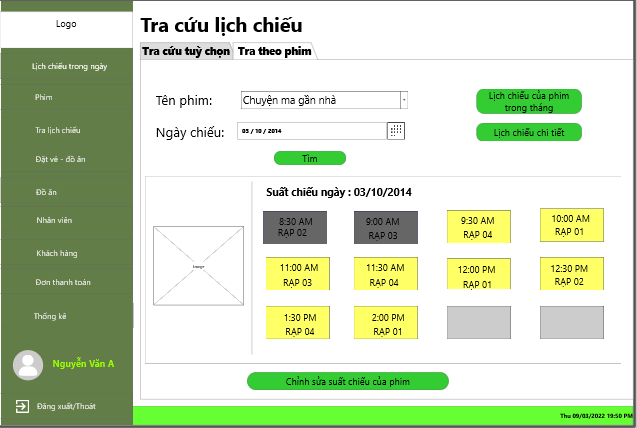
Giao diện quản lý nhân viên

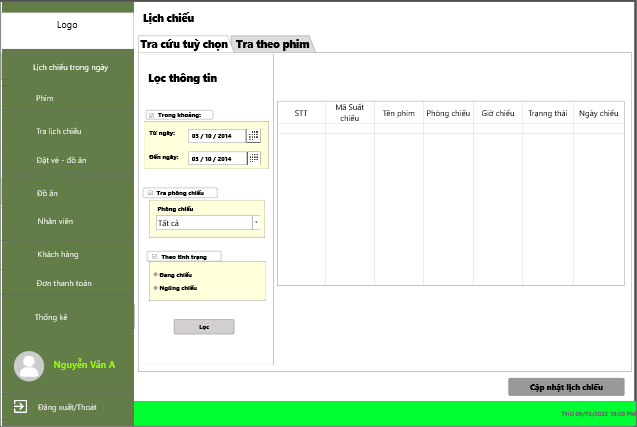
Màn hình chính



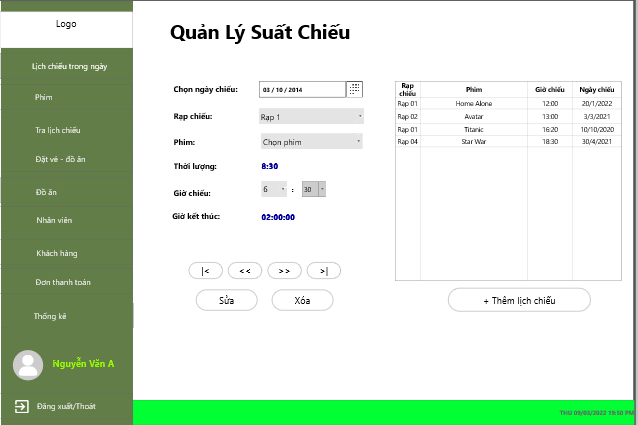
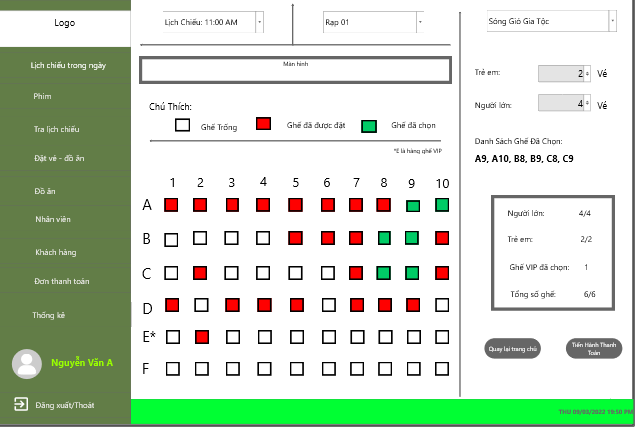
Tra cứu lịch chiếu tổng hợp

Tra cứu lịch chiếu theo phim





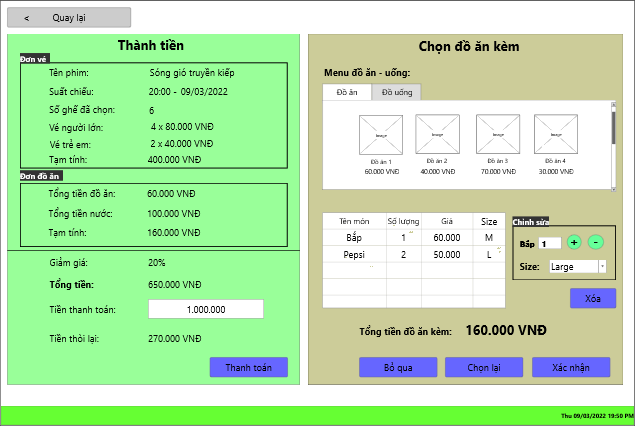
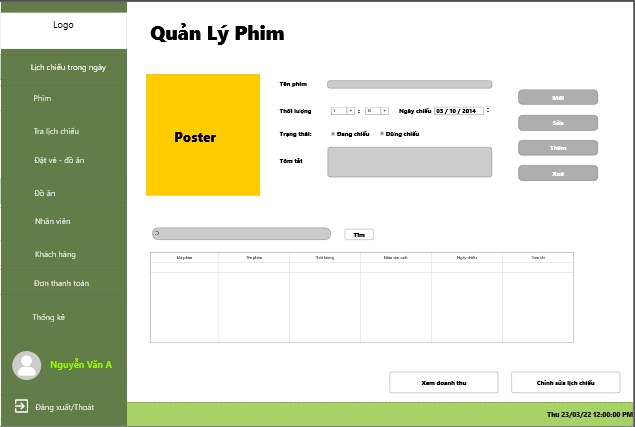
Quản lý suất chiếu



Quản lý đặt vé

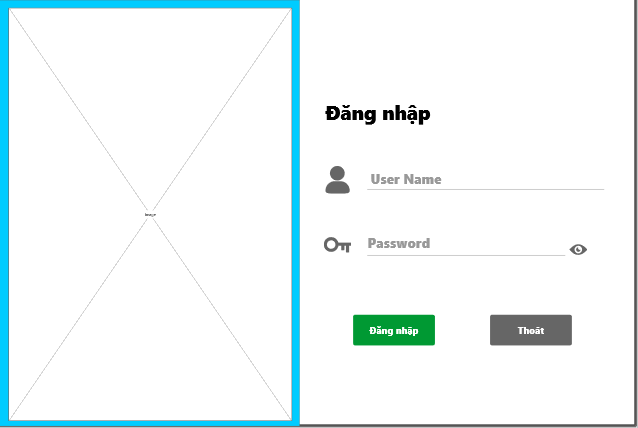
Quản lý phim

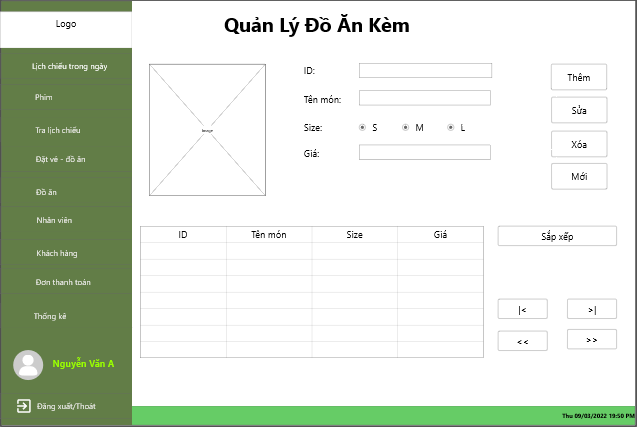
Đơn thanh toán



Form đăng nhập

Quản lý khách hàng





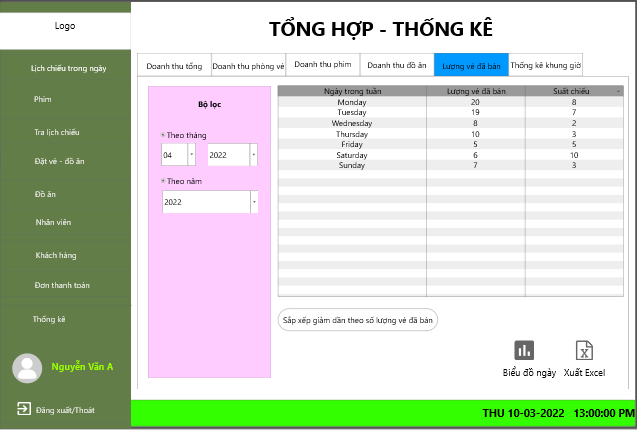
Quản lý đồ ăn kèm

Cập nhật thông tin tài khoản



Thống kê lượng vé

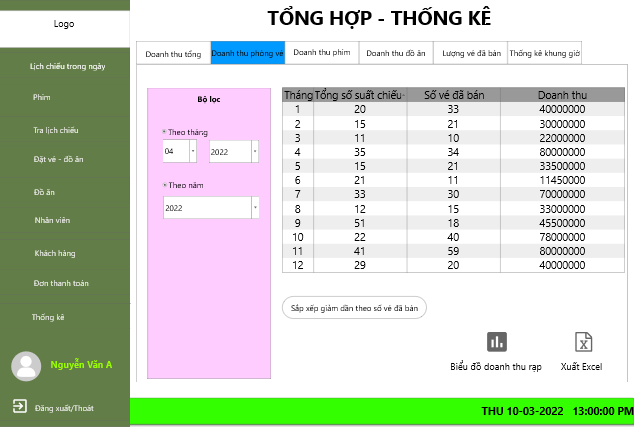
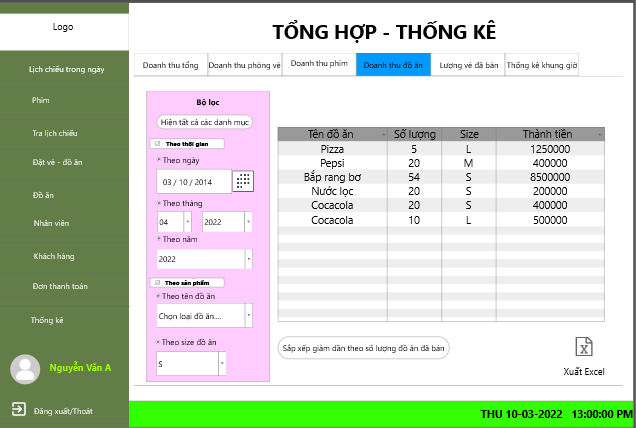
Thống kê doanh thu tổng





Thống kê doanh thu đồ ăn

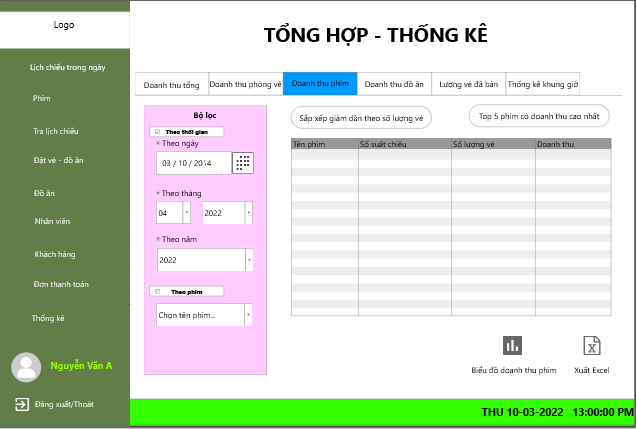
Thống kê doanh thu phòng vé



Thống kê khung giờ

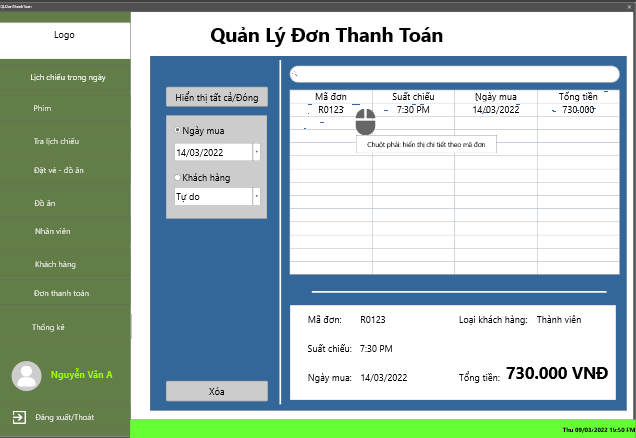
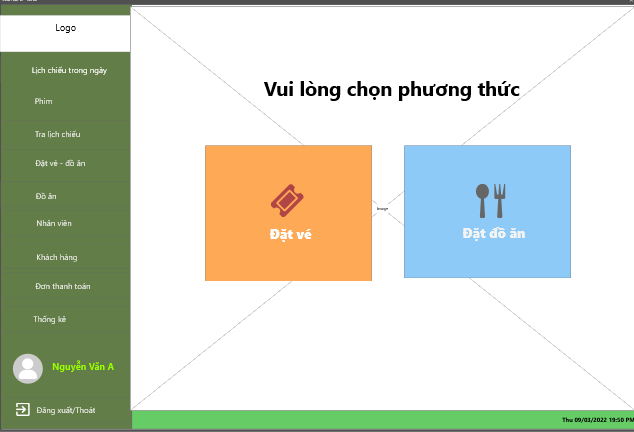


Thống kê doanh thu phim



Quản lý đặt vé – đồa ăn

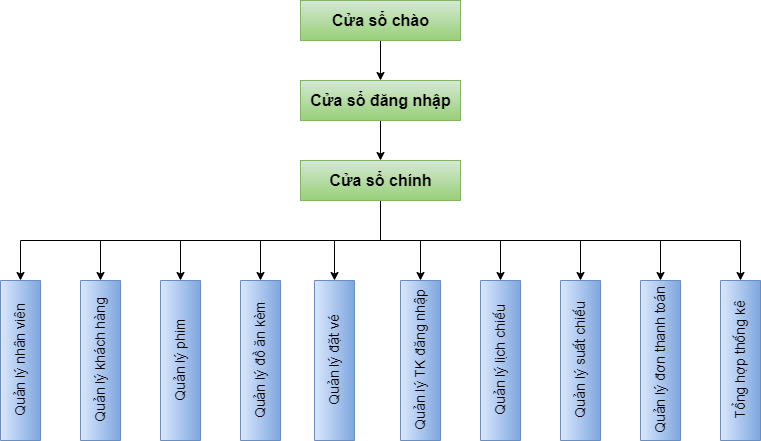
Quản lý đơn thanh toán



### Sơ đồ tổ chức giao diện

Giao diện chính là một cửa sổ chứa menu chính. Thông qua đó để đi đến các giao diện thành viên để thực hiện các chức năng trong hệ thống.

Theo yêu cầu thì mọi nhân viên phải đăng nhập trước khi sử dụng ứng dụng nên cửa sổ đăng nhập xuất hiện trước để yêu cầu đăng nhập.



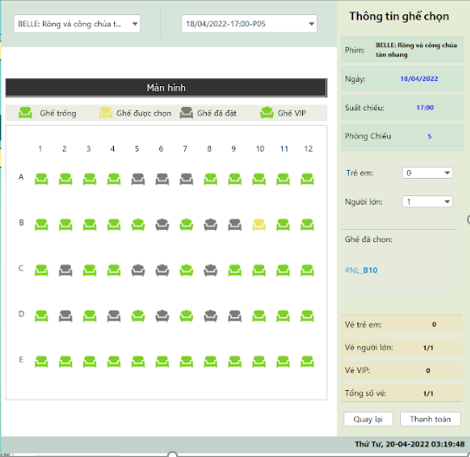
### Thiết kế giao diện cho các chức năng nghiệp vụ

#### Thiết kế DatMenuFrame



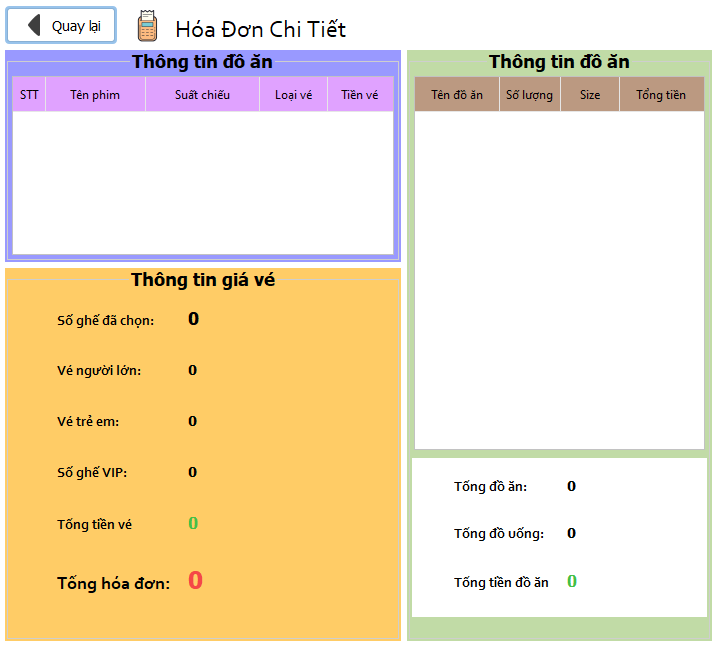
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | DatMenuFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JFrame | ptnContent | Background: [218,237,237]  CardLayout |
| 3 | JPanel | ptnHome | Background: [217,229,235]  Null |
| 4 | ButtonGradient | btnDatVe | Text: Đặt vé  Icon: ticket\_menu\_icon.png  Color1: [238,235,128]  Color2: [240,226,180] |
| 5 | ButtonGradient | btnDatDoAn | Text: Đặt đồ ăn  Icon: food\_menu\_icon\_100.png  Color1: [147,220,219]  Color2: [108,179,184] |
| 6 | JPanel | pnlDatVe | BorderLayout  Card Name: card3 |
| 7 | JPanel | pnlDatDoAn | BorderLayout  Card Name: card4 |
| 8 | JPanel | pnlThanhToan | BorderLayout  Card Name: card5 |

#### Thiết kế DatVeFrame



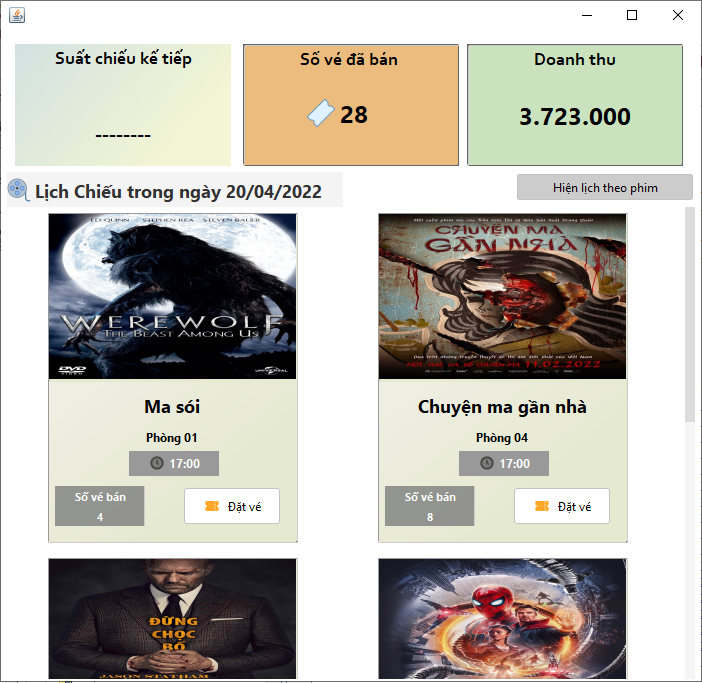
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | DatVeFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JPanel | pnlContent | Background: [255,255,255] |
| 3 | JPanel | pnlLichRap | Background: [214,229,217]  AbsoluteLayout |
| 4 | JPanel | pnlSuatPhong | Background: [198,217,201] |
| 5 | JComboBox | cboSuat |  |
| 6 | JPanel | pnlPhim | Background: [198,217,201] |
| 7 | JComboBox | cboPhim |  |
| 8 | JPanel | pnlPhong | Background: [255,255,255]  Null |
| 9 | JLabel | lblManHinh | Text: Màn hình  Background: [51,51,51] |
| 10 | JPanel | pnlChuThichGhe | GridLayout |
| 11 | JLabel | lblGheTrong | Text: Ghế Trống  Icon: seat\_available.png |
| 12 | JLabel | lblGheDuocChon | Text: Ghế được chọn  Icon: seat\_selected.png |
| 13 | JLabel | lblGheDaDat | Text: Ghế Đã đặt  Icon: seat\_unavailable.png |
| 14 | JLabel | lblGheVIP | Text: Ghế VIP  Icon: seat\_vp\_available.png |
| 15 | JPanel | pnlGheMAp | CardLayout |
| 16 | JPanel | pnlGhe | Background: [153,204,255]  BorderLayout |
| 17 | JPanel | pnlEmpty | Background: [255,153,153] |
| 18 | JPanel | pnlGheChon | Background: [229,236,213] |
| 19 | JPanel | pnlThongTin | GridLayout |
| 20 | JPanel | PnlTenPhim | Background: [210,230,211] |
| 21 | JLabel | lblTenPhim | Text: Phim |
| 22 | JLabel | lblPhim |  |
| 23 | JPanel | pnlNgayChieu | Background: [210,230,211] |
| 24 | JLabel | lblNgay | Text: Ngày |
| 25 | JLabel | lblNgayChieu |  |
| 26 | JPanel | pnlSuat | Background: [210,230,211] |
| 27 | JLabel | lblSuat | Text: Suất |
| 28 | JLabel | lblSuatChieu |  |
| 29 | JPanel | pnlPhong | Background: [210,230,211] |
| 30 | JLabel | lblPhongChieu | Text: Phòng |
| 31 | JLbael | lblPhong |  |
| 32 | JPanel | pnlSelectedGhe | Background: [229,234,218] |
| 33 | JLabel | lblGhe | Text: Ghế |
| 34 | JLabel | lblGheChon |  |
| 35 | JPanel | pnlDetail | GridLayout |
| 36 | JLabel | lblVeTreEm | Text: Vé trẻ em |
| 37 | JLabel | lblSoVeTESelected |  |
| 38 | JLabel | lblVeNguoiLon | Text: Vé người lớn |
| 39 | JLabel | lblVeNLSelected |  |
| 40 | JLabel | lblVip | Text: Vé VIP |
| 41 | JLabel | lblSoVeVip |  |
| 42 | JLabel | lblTongVe | Text: Tổng số vé |
| 43 | JLabel | lblSoVeSelected |  |
| 44 | JButton | btnQuayLai | Text: Quay lại |
| 45 | JButton | btnThanhToan | Text: Thanh toán |

#### Thiết kế HoaDonChiTietFrame



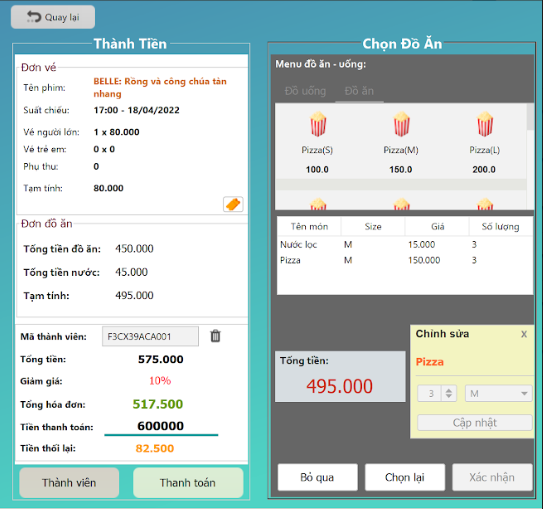
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | HoaDonChiTietFrame | DefaultCloseOperation: DISPONSE |
| 2 | JButton | btnQuayLai | Text: Quay lại  Icon: icons8\_sort\_left\_26px\_5.png |
| 3 | JLabel | lblHoaDon | Text: Hóa đơn chi tiết  Icon: icons8\_billing\_machine\_48px.png |
| 4 | JPanel | pnlDonVe | Titled Border: Thông tin đơn vé |
| 5 | JTabel | tblDonVe |  |
| 6 | JPanel | pnlGiaVe | Titled Border: Tong tin giá vé |
| 7 | JLabel | lblGheDaChon | Text: Số ghế đã chọn |
| 8 | JLabel | lblSoGheDC |  |
| 9 | JLabel | lblVeNL | Text: Vé người lớn |
| 10 | JLabel | lblTienVeNL |  |
| 11 | JLabel | lblVeTE | Text: Vé trẻ em |
| 12 | JLabel | lblTienVeTE |  |
| 13 | JLabel | lblGheVip | Text: Số ghế Vip |
| 14 | JLabel | lblSoGheVip |  |
| 15 | JLabel | lblGiamGia | Text: Giảm giá |
| 16 | JLabel | lblGiamGia |  |
| 17 | JLabel | lblTongTien | Text: Tổng tiền vé |
| 18 | JLabel | lblTongTien |  |
| 19 | JPanel | pnlChonDoAn | Titled Border: Thông tin đồ ăn |
| 20 | Jtable | tblDonDoAn |  |

#### Thiết kế LichChieuTrongNgayFrame



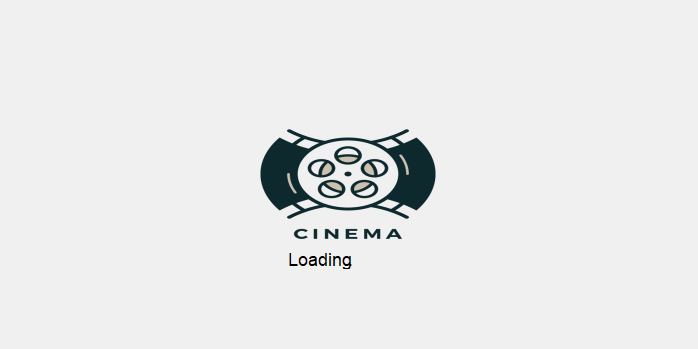
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | LichChieuTrongNgayFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JPanel | pnlSuatChieu | Background: [239,224,120] |
| 3 | JLabel | lblSuatChieu | Text: Suất chiếu kế tiếp |
| 4 | JLabel | lblCountDown |  |
| 5 | JLabel | lblPhim |  |
| 6 | JPanel | pnlSoVe | Background: [236,188,127] |
| 7 | JLabel | lblSoVe | Text: Số vé đã bán |
| 8 | JLabel | lblLuongVe | Icon: icons8\_ticket\_30px\_2.png |
| 9 | JPanel | pnlDoanhThu |  |
| 10 | JLabel | lblDoanhthu | Text: Doanh thu |
| 11 | JLabel | lblDoanhThu |  |
| 12 | JLabel | lblLichChieu | Text: Lich chiếu trong ngày  Icon: icons8\_film\_reel\_24px.png |
| 13 | JTextField | txtTimPhim |  |
| 14 | JToggleButton | tglState | Text: Hiện lịch theo phim |

#### Thiết kế ThanhToanFrame



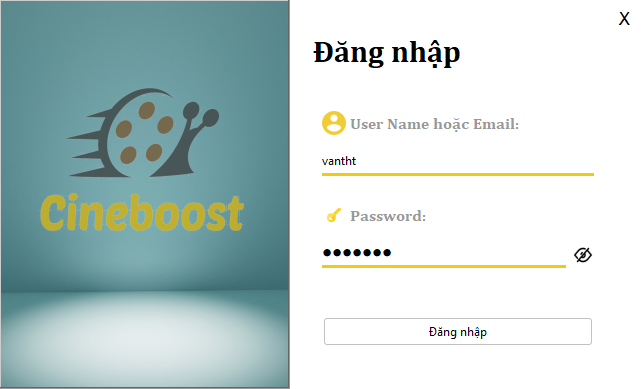
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | ThanhToanFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JButton | btnQuayLai | Text: Quay lại  Icon: icons8\_return\_20px.png |
| 3 | JPanel | pnlThanhTien | Titled Border: Thành tiền |
| 4 | JPanel | pnlDonVe | Titled Border: Đơn vé |
| 5 | JLabel | lblPhim | Text: Tên phim |
| 6 | JLabel | lblTenPhim |  |
| 7 | JLabel | lblSuat | Text: Suất chiếu |
| 8 | JLabel | lblSuatChieu | Text: (Truyền vào suất chiếu) |
| 9 | JLabel | lblVeNL | Text: Vé người lớn |
| 10 | JLabel | lblVeNguoiLon |  |
| 11 | JLabel | lblVeTE | Text: Vé trẻ em |
| 12 | JLabel | lblVeTreEm |  |
| 13 | JLabel | lblTamTinhDonVe | Text: Tạm tính |
| 14 | JLabel | lblTamTinhVe |  |
| 15 | JPanel | pnlDoAn | Titled Border: Đơn đồ ăn |
| 16 | JLabel | lblTongTienDA | Text: Tổng tiền đồ ăn |
| 17 | JLabel | lblTongTienDoAn |  |
| 18 | JLabel | lblTongTienNc | Text: Tổng tiền nước |
| 19 | JLabel | lblTongTienNuoc |  |
| 20 | JLabel | lblTamTinhDoAn | Text: Tạm tính |
| 21 | JLabel | lblTamTinhDA |  |
| 22 | Jpanel | pnlTongTien |  |
| 23 | JLabel | lblMãTV | Text: Mã thành viên |
| 24 | JTextField | txtIDThanhVien |  |
| 25 | JLabel | lblGiam\_Gia | Giảm giá |
| 26 | JLabel | lblGiamGia |  |
| 27 | JLabel | lblTong | Text: Tổng tiền |
| 28 | JLabel | lblTongTien |  |
| 29 | JLabel | lblTienThanhToan | Text: Tiền thanh toán |
| 30 | JTextField | txtTienThanhToan |  |
| 31 | JLabel | lblTienThoi | Text: Tiền thói lại |
| 32 | JLabel | lblTienThoiLai |  |
| 33 | JButton | btnThanhVien | Text: Thành viên |
| 34 | JButton | btnThanhToan | Text: Thanh toán |
| 35 | JPanel | pnlChonDoAn |  |

#### Thiết kế IntroFrame



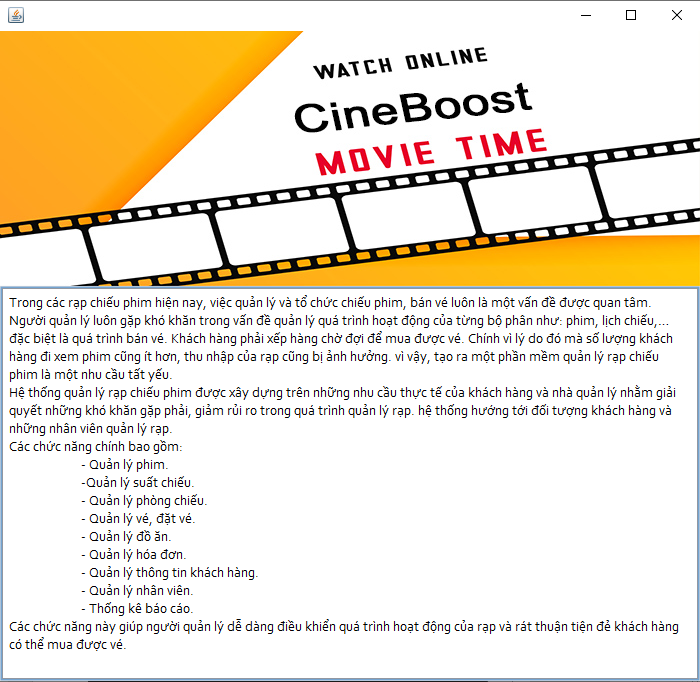
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | IntroFrame | DefaultCloseOperation: DISPONSE |
| 2 | JLabel | lblDot |  |
| 3 | JLabel | lblLoading |  |
| 4 | JLabel | lblBackground |  |

#### Thiết kế LoginFrame



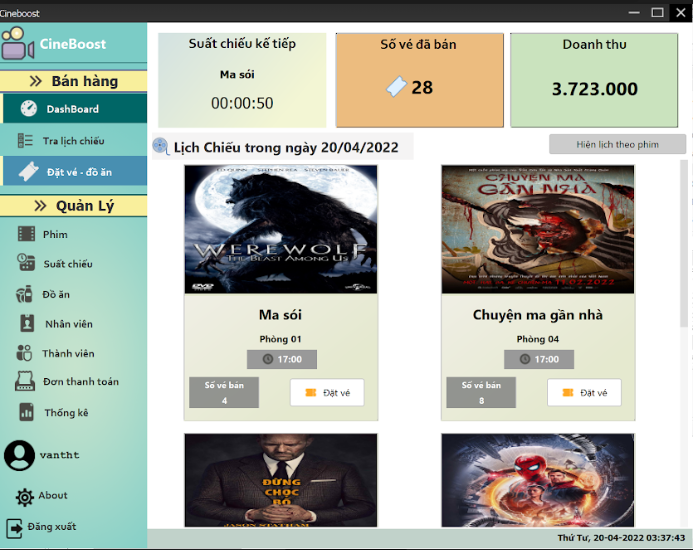
|  |  |  |  |
| --- | --- | --- | --- |
| TT | ĐIỀU KHIỂN | TÊN | GIÁ TRỊ |
| 1 | JFrame | LoginFrame | DefaultCloseOperation: DISPONSE |
| 2 | JPanel | plLeft |  |
| 3 | JLabel | lblBackground | Icon: |
| 4 | JPanel | pnlRight |  |
| 5 | JLabel | lblTitle | Text: Đăng nhập |
| 6 | JLabel | lblUserEmail | Text: User Name hoặc Email  Icon: account\_24.png |
| 7 | JTextField | txtUserName |  |
| 8 | JLabel | lblPass | Text: Password  Icon: key\_24.png |
| 9 | JPasswordField | txtPassword |  |
| 10 | JLabel | lblHile | Icon: |
| 11 | JButton | btnLogin | Text: Đăng nhập |
| 12 | JButton | btnExit | Text: X |

#### Thiết kế AboutFrame



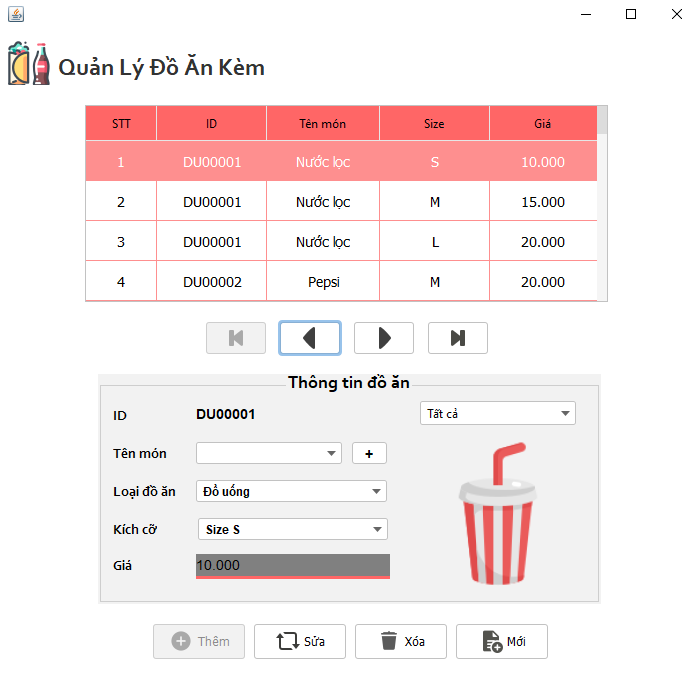
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | AboutFrame | DefaultCloseOperation: DISPONSE |
| 2 | JTextPane | txtText | Text: |
| 3 | JLabel | lblPoster | Icon: poster.jpg |

#### Thiết kế MainFrame



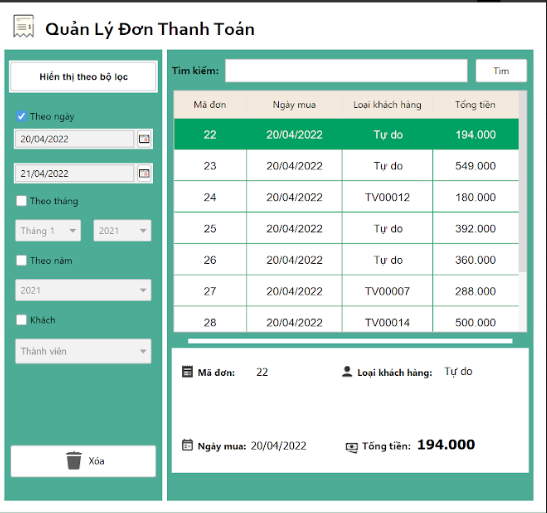
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | MainFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE  Title: Mainframe |
| 2 | JPanel | pnlHome | BorderLayout  Background: [102,102,102] |
| 3 | JPanel | pnlTilte | FlowLayout |
| 4 | JLabel | lblTitle | Text: Title |
| 5 | JPanel | pnlActions | GridLayout  Background: [51,51,51]  Direction: After |
| 6 | JPanel | pnlMinimize | Background: [51,51,51] |
| 7 | JLabel | lblMinimize | HorizontalAlignment: CENTER  Icon: icons8\_minus\_18px\_1.png  Cursor: Hand Cursor |
| 8 | JPanel | pnlMaximize | Background: [51,51,51] |
| 9 | JLabel | lblMaximize | HorizontalAlignment: CENTER  Icon: icons8\_rectangle\_stroked\_18px.png  Cursor: Hand Cursor |
| 10 | JPanel | pnlClose | Background: [51,51,51] |
| 11 | JLabel | lblClose | HorizontalAlignment: CENTER  Icon: icons8\_multiply\_18px\_1.png  Cursor: Hand Cursor |
| 12 | JPanel | pnlLeft | AbsoluteLayout  Background: [255,255,255] |
| 13 | JLabel | lblLogo | Icon: icons8\_documentary\_48px.png |
| 14 | JLabel | lblBanHang | Text: Bán hàng  Icon: double\_arrow.png  HorizontalAlignment: CENTER  Border: MatteBorder  Border.Insets: [2,0,2,0]  Border.Color: [98,107,150]  Cursor: Hand Cursor |
| 15 | JPanel | pnlBanHanh | GridLayout |
| 16 | JLayeredPane | lpnl2 |  |
| 17 | JLabel | btnLichChieu | Text: DashBoard  Icon: icons8\_dashboard\_26px\_1.png  HorizontalAlignment: LEFT  Cursor: Hand Cursor |
| 18 | JLayeredPane | lpnl1 |  |
| 19 | JLabel | btnTraLichChieu | Text: Tra lịch chiếu  Icon: icons8\_content\_26px.png  HorizontalAlignment: LEFT  Cursor: Hand Cursor |
| 20 | JLayeredPane | lpnl3 |  |
| 21 | JLabel | btnTraLichChieu | Text: Đặt vé – đồ ăn  Icon: icons8\_ticket\_26px.png  HorizontalAlignment: LEFT  Cursor: Hand Cursor |
| 22 | JLabel | lblQuanLy | Text: Quản Lý  Icon: double\_arrow.png  HorizontalAlignment: CENTER  Border: Matte Border  Border.Insets: [2,0,2,0]  Border.Color: [0,102,102]  Cursor: Hand Cursor |
| 23 | JPanel | pnlQuanLy | GirdLayout |
| 24 | JLayeredPane | lpnl4 |  |
| 25 | JLabel | btnQLPhim | Text: Phim  Icon: icons8\_movie\_26px.png  Background: [240,239,239]  HorizontalAlignment: LEFT  Cursor: Hand Cursor  HorizontalTextPosition: RIGHT  VerticalTextPosition: BOTTOM |
| 26 | JLayeredPane | lpnl10 |  |
| 27 | JLabel | btnQLSuatChieu | Text: Suất chiếu  Icon: icons8\_schedule\_26px.png  Background: [240,239,239]  Foreground: [51,51,51]  HorizontalAlignment: LEFT  Cursor: Hand Cursor  HorizontalTextPosition: RIGHT  VerticalTextPosition: BOTTOM |
| 28 | JLayeredPane | lpnl5 |  |
| 29 | JLabel | btnQLDoAn | Text: Đồ ăn  Icon: icons8\_Street\_Food\_26px.png  Background: [240,239,239]  Foreground: [51,51,51]  HorizontalAlignment: LEFT  Cursor: Hand Cursor  HorizontalTextPosition: RIGHT  VerticalTextPosition: BOTTOM |
| 30 | JLayeredPane | lpnl6 |  |
| 31 | JLabel | btnQLNhanVien | Text: Nhân viên  Icon: icons8\_name\_tag\_26px.png  Background: [102,102,102]  Foreground: [51,51,51]  HorizontalAlignment: LEFT  Cursor: Hand Cursor  HorizontalTextPosition: RIGHT  VerticalTextPosition: BOTTOM |
| 32 | JLayeredPane | lpnl7 |  |
| 33 | JLabel | btnQLThanhVien | Text: Thành viên  Icon: icons8\_member\_26px.png  Background: [102,102,102]  Foreground: [51,51,51]  HorizontalAlignment: LEFT  Cursor: Hand Cursor  HorizontalTextPosition: RIGHT  VerticalTextPosition: BOTTOM |
| 34 | JLayeredPane | lpnl9 |  |
| 35 | JLabel | btnQLThongKe | Text: Thống kê  Icon: icons8\_report\_file\_26px.png  Background: [240,240,240]  Foreground: [51,51,51]  HorizontalAlignment: LEFT  Cursor: Hand Cursor  VerticalTextPosition: BOTTOM |
| 36 | JPanel | pnlProfile |  |
| 37 | JLabel | lblProfile | Icon:  Cursor: Hand Cursor |
| 38 | JLabel | lblTenTK | Text: (Truyền vào tên tài khoản đăng nhập) |
| 39 | JLayeredPane | lpnl12 |  |
| 40 | JLabel | btnAbout | Text: About  Icon: icons8\_settings\_26px.png  Cursor: Hand Cursor |
| 41 | JLayeredPane | lpnl11 |  |
| 42 | JLabel | BtnExit | Text: Đăng xuất  Icon: icons8\_Logout\_26px.png  Cursor: Hand Cursor |
| 43 | JPanel | pnlRight | CardLayout  Background: [204,255,255] |
| 44 | JPanel | pnlClock | Background: [193,210,202] |
| 45 | JLabel | lblClock | Text: (Truyền vào giá trị thời gian) |

#### Thiết kế QLDoAnKemFrame



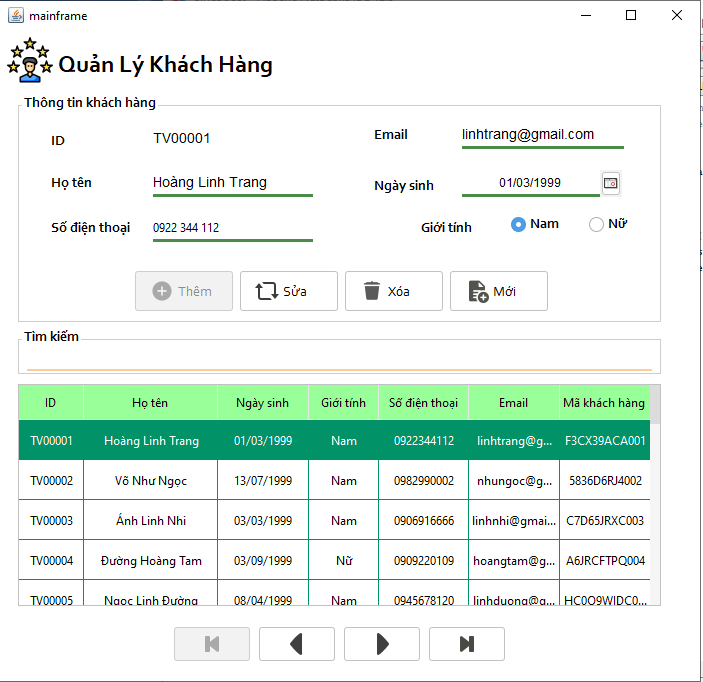
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | QLDoAnKemFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JPanel | pnlMain | Background: [255,255,255] |
| 3 | JLabel | lblTitle | Text: Quản lý đồ ăn kèm |
| 4 | JTable | tblDoAn | GridColor: [169,197,80]  SelectionBackground: [61,144,106]  SelectionForeground: [244,244,244] |
| 5 | JPanel | pnlDiChuyen | GridLayout |
| 6 | JButton | btnFirst | Icon: icons8\_skip\_to\_start\_26px.png  Cursor: Hand Cursor |
| 7 | JButton | btnPre | Icon: icons8\_skip\_to\_start\_26px.png  Cursor: Hand Cursor |
| 8 | JButton | btnNext | Icon: icons8\_sort\_right\_26px\_2.png  Cursor: Hand Cursor |
| 9 | JButton | btnLast | Icon: icons8\_end\_26px.png  Cursor: Hand Cursor |
| 10 | JPanel | pnlThongTin | Border.Title: Thông tin đô ăn  Border. Justification: Center |
| 11 | JLabel | lblId | Text: ID |
| 12 | JLabel | lblID\_DoAn |  |
| 13 | JLabel | lblTenmon | Text: Tên món |
| 14 | JTextField | txtTenmon |  |
| 15 | JLabel | lblLoaiDA | Text: Loại đô ăn |
| 16 | JComboBox | cboLoaiDoAn |  |
| 17 | JLabel | lblKichCo | Text: Kích cỡ |
| 18 | JComboBox | cboSize |  |
| 19 | JLabel | lblGia | Text: Giá |
| 20 | JTextField | txtGia |  |
| 21 | JComboBox | cboLocDoAn |  |
| 22 | JPanel | pnlAnh | Background: [204,204,204] |
| 23 | JLabel | lblAnhDoAn |  |
| 24 | JPanel | pnlThaoTac | GridLayout |
| 25 | Jbutton | btnThem | Text: Thêm  Icon: icons8\_add\_24px\_1.png  Cursor: Hand Cursor |
| 26 | Jbutton | btnSua | Text: Sửa  Icon: icons8\_refresh\_24px\_1.png  Cursor: Hand Cursor |
| 27 | Jbutton | btnXoa | Text: Xóa  Icon: icons8\_remove\_24px\_1.png  Cursor: Hand Cursor |
| 28 | Jbutton | btnMoi | Text: Mới  Icon: icons8\_new\_copy\_24px\_1.png  Cursor: Hand Cursor |

#### Thiết kế QLDonThanhToanFrame



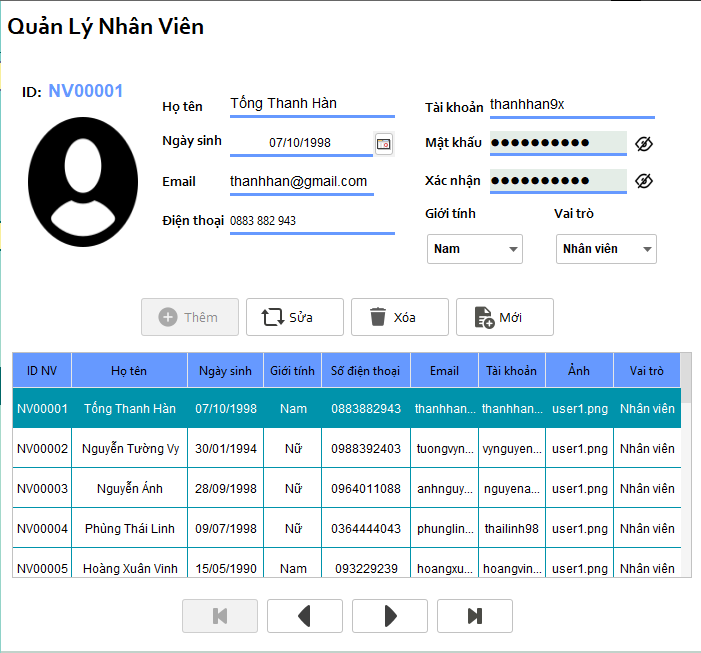
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | QLDonThanhToanFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JPanel | pnlMain | Background: [255,255,255] |
| 3 | JLabel | lblTitle | Text: Quản lý đơn thanh toán  Icon: icons8\_bill\_48px\_1.png |
| 4 | JPanel | pnlBoLoc | Background: [204,204,255] |
| 5 | JButton | btnShowAll | Text: Hiển thị tất cả |
| 6 | JPanel | pnlLoc | Background: [204,204,204] |
| 7 | JRadioButton | rdoNgayMua | Text: Ngày mua  ButtonGroup: btgLoc |
| 8 | JRadioButton | rdoKhachHang | Text: Khách hàng  ButtonGroup: btgLoc |
| 9 | JDateChooser | dcsNgayMua | DateFormatString: dd/MM/yyyy |
| 10 | JComboBox | cboLoaiKH |  |
| 11 | JButton | btnXoa | Text: Xóa  Icon:  icons8\_remove\_30px\_2.png |
| 12 | JLabel | lblTim | Text: Tìm kiếm |
| 13 | JTextField | txtTim | ToolTipText: Nhập mã đơn thanh toán |
| 14 | JTabel | tblDonThanhToan | ToolTipText: Chọn 1 dòng và click chuột phải để xem hóa đơn chi tiết |
| 15 | JPanel | pnlThongTinDTT |  |
| 16 | JLabel | lblMa | Text: Mã đơn  Icon: icons8\_bill\_18px.png |
| 17 | JLabel | lblMaDon |  |
| 18 | JLabel | lblSuat | Text: Suất chiếu  Icon: icons8\_time\_18px.png |
| 19 | JLabel | lblSuatChieu |  |
| 20 | JLabel | lblNgay | Text: Ngày mua  Icon: icons8\_timeline\_week\_18px.png |
| 21 | JLabel | lblNgayMua |  |
| 22 | JLabel | lblKhachHang | Text: Loại khách hàng  Icon: icons8\_user\_18px.png |
| 23 | JLabel | lblLoaiKH |  |
| 24 | JLabel | lblTong\_Tien | Text: Tổng tiền  Icon: icons8\_money\_18px.png |
| 25 | JLabel | lblTongTien |  |

#### Thiết kế QLKhachHangFrame



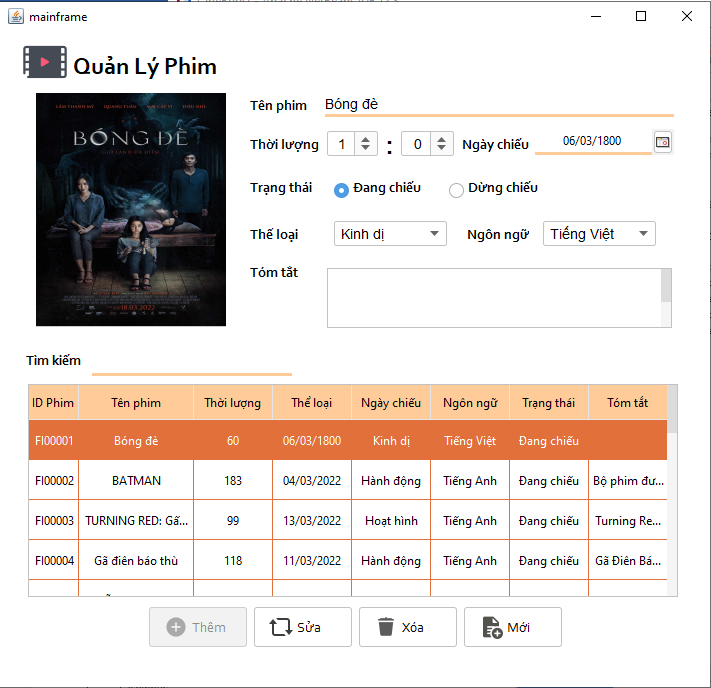
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | QLKhachHangFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JPanel | pnlMain | Background: [255,255,255] |
| 3 | JLabel | lblTitle | Text: Quản lý khách hàng |
| 4 | JPanel | pnlThongTinKhach | Border.Title: Thông tin khách hàng  Border.Justification: Left |
| 5 | JLabel | lblID | Text: ID |
| 6 | JLabel | lblIDKhachHang |  |
| 7 | JLabel | lblHoTen | Text: Họ tên |
| 8 | JTextField | txtHoTen | Border.Insets: [0, 0, 3, 0]  Border.Color: [153,255,153] |
| 9 | JLabel | lblSDT | Text: Số điện thoại |
| 10 | JTextField | txtSoDT | Border.Insets: [0, 0, 3, 0]  Border.Color: [153,255,153] |
| 11 | Jlabel | lblEmail | Text: Email |
| 12 | JTextField | txtEmail | Border.Insets: [0, 0, 3, 0]  Border.Color: [153,255,153] |
| 13 | JLabel | lblNgaySinh | Text: Ngày sinh |
| 14 | JDateChooser | dcsNgaySinh | DateFormatString: dd/MM/yyyy |
| 15 | JradioButton | rdoNam | Text: Nam  ButtonGroup: btgGioiTinh |
| 16 | JradioButton | rdoNu | Text: Nu  ButtonGroup: btgGioiTinh |
| 17 | JPanel | pnlDiChuyen | GridLayout |
| 18 | JButton | btnThem | Text: Thêm  Icon: icons8\_add\_24px\_1.png |
| 19 | JButton | btnSua | Text: Sửa  Icon: icons8\_refresh\_24px\_1.png |
| 20 | JButton | btnXoa | Text: Xóa  Icon: icons8\_remove\_24px\_1.png |
| 21 | JButton | btnMoi | Text: Mới  Icon: icons8\_new\_copy\_24px\_1.png |
| 22 | JPanel | pnlTim | Border. Title: Tìm kiếm  Border .Justification: Left |
| 23 | JTextField | txtTim |  |
| 24 | Jtabel | tblKhachHang | GridColor: [169,197,80]  SelectionBackground: [61,144,106]  SelectionForeground: [244,244,244]  SelectionMode: SINGLE\_INTERVAL |
| 25 | JPanel | pnlThaoTac | FlowLayout |
| 26 | JButton | btnFirst | Icon: icons8\_skip\_to\_start\_26px.png |
| 27 | JButton | btnPre | Icon: icons8\_sort\_left\_26px\_5.png |
| 28 | JButton | btnNext | Icon: icons8\_sort\_right\_26px\_2.png |
| 29 | JButton | btnLast | Icon: icons8\_end\_26px.png |

#### Thiết kế QLNhanVienFrame



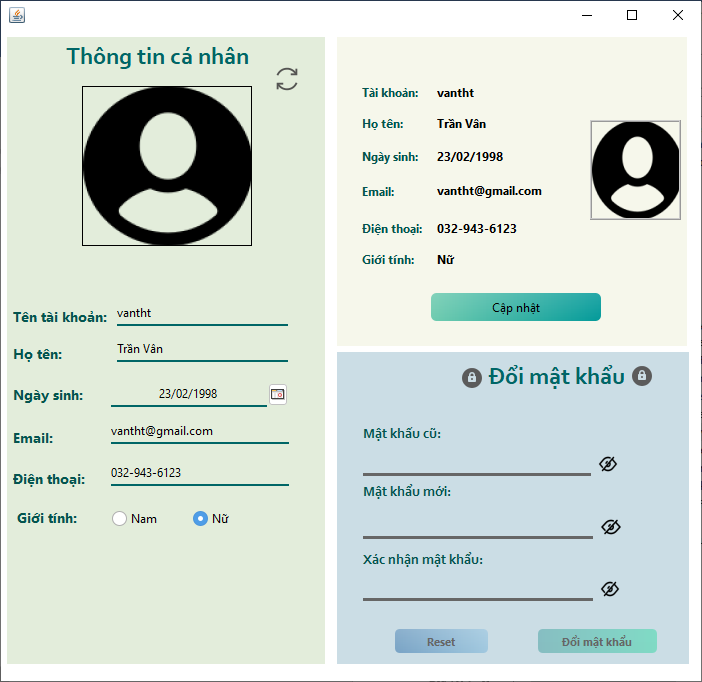
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | QLNhanVienFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JPanel | pnlMain | Background: [255,255,255] |
| 3 | JLabel | lblTitle | Text: Quản lý nhân viên |
| 4 | JPanel | pnlThongTin |  |
| 5 | JLabel | lblID | Text: ID |
| 6 | JLabel | lblIDNhanVien |  |
| 7 | JLabel | lblHoTen | Text: Họ tên |
| 8 | JTextField | txtHoTen | Border.Insets: [0, 0, 3, 0]  Border.Color: [204,153,255] |
|  | JLabel | lblNgaySinh | Text: Ngày sinh |
|  | JDateChooser | dcsNgaySinh | DateFormatString: dd/MM/yyyy |
| 9 | JLabel | lblSDT | Text: Số điện thoại |
| 10 | JTextField | txtSoDT | Border.Insets: [0, 0, 3, 0]  Border.Color: [204,153,255] |
| 11 | Jlabel | lblEmail | Text: Email |
| 12 | JTextField | txtEmail | Border.Insets: [0, 0, 3, 0]  Border.Color: [204,153,255] |
| 13 | Jlabel | lblTenTK | Text: Mật khẩu |
| 14 | JTextField | txtTaiTK | Border.Insets: [0, 0, 3, 0]  Border.Color: [204,153,255] |
| 15 | Jlabel | lblMatKhau | Text: Mật khẩu |
| 16 | JPasswordField | txtMatKhau |  |
| 17 | JLabel | lblHide | ToolTipText: Nhấn giữ để hiện Password  Cursor: Hand Cursor |
| 18 | Jlabel | lblXacNhan | Text: Xác nhận |
| 19 | JPasswordField | txtXacNhanMK |  |
| 20 | JLabel | lblHide\_XN | ToolTipText: Nhấn giữ để hiện Password  Cursor: Hand Cursor |
| 21 | JPanel | pnlThaoTac | GridLayout |
| 22 | JButton | btnThem | Text: Thêm  Icon: icons8\_add\_24px\_1.png |
| 23 | JButton | btnSua | Text: Sửa  Icon: icons8\_refresh\_24px\_1.png |
| 24 | JButton | btnXoa | Text: Xóa  Icon: icons8\_remove\_24px\_1.png |
| 25 | JButton | btnMoi | Text: Mới  Icon: icons8\_new\_copy\_24px\_1.png |
| 26 | JLayeredPane | pnlHinh |  |
| 27 | JLabel | lblHover |  |
| 28 | JLabel | lblAnh |  |
| 29 | Jtabel | tblNhanVien | GridColor: [169,197,80]  SelectionBackground: [61,144,106]  SelectionForeground: [244,244,244] |
| 30 | JPanel | pnlDiChuyen | FlowLayout |
| 31 | JButton | btnFirst | Icon: icons8\_skip\_to\_start\_26px.png |
| 32 | JButton | btnPre | Icon: icons8\_sort\_left\_26px\_5.png |
| 33 | JButton | btnNext | Icon: icons8\_sort\_right\_26px\_2.png |
| 34 | JButton | btnLast | Icon: icons8\_end\_26px.png |

#### Thiết kế QlPhimFrame



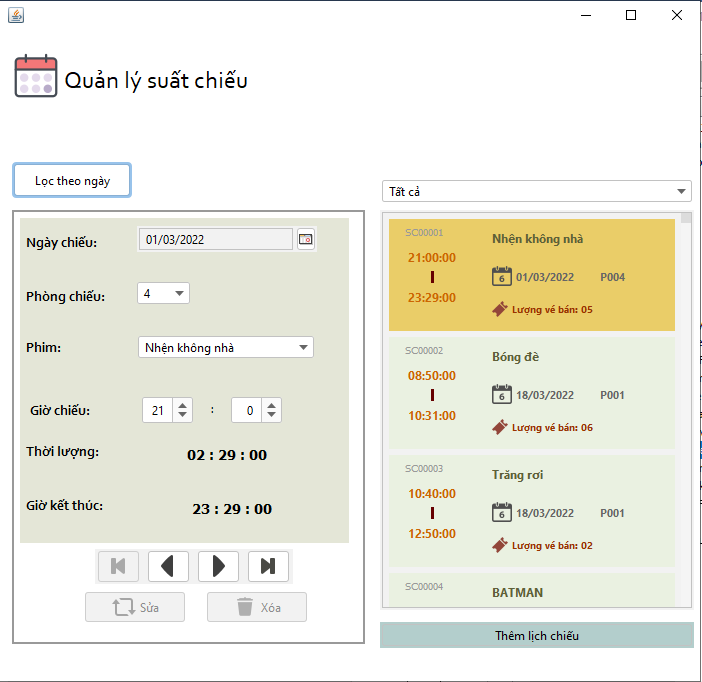
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | QlPhimFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JPanel | pnlMain | Background: [240,240,240] |
| 3 | JLabel | lblTitle | Text: Quản lý phim  Icon: icons8\_movie\_beginning\_48px\_1.png |
| 4 | JLayeredPane | pnlPoster | Null |
| 5 | JLabel | lblHover | Text: Double click để chọn poster |
| 6 | JLabel | lblPoster |  |
| 7 | JPanel | pnlThongTinPhim |  |
| 8 | JLabel | lblTenPhim | Text: Tên phim |
| 9 | JTextField | txtTenPhim | Border.Insets: [0, 0, 3, 0]  Border.Color: [204,153,255] |
| 10 | JLabel | lblThoiLuong | text Thời lượng |
| 11 | Jspinner | spnGio | ToolTipText: Giờ |
| 12 | Jspinner | spnPhut | ToolTipText: Phút |
| 13 | JLabel | lblNgayChieu | Text: Ngày chiếu |
| 14 | JDateChooser | dcsNgayChieu | DateFormatString: dd/MM/yyyy |
| 15 | JLabel | lblTrangThai | Text: Trạng thái |
| 16 | JRadioButton | rdoDangChieu | Text: Đang chiếu  ButtonGroup: btgTrangThai |
| 17 | JRadioButton | rdoDungChieu | Text: Dừng chiếu  ButtonGroup: btgTrangThai |
| 18 | JLabel | lblTheLoai | Text: Thể loại |
| 19 | JComboBox | cboTheLoai |  |
| 20 | JLabel | lblNgonNgu | Text: Ngôn ngữ |
| 21 | JComboBox | cboNgonNgu |  |
| 22 | JLabel | lblTomTat | Text: Tóm tắt |
| 23 | JtextArea | txtTomTat | Columns: 20  Rows: 5 |
| 24 | JLabel | lblTim | Text: Tìm kiếm |
| 25 | JTextField | txtTim | Border.Insets: [0, 0, 3, 0]  Border.Color: [204,153,255] |
| 26 | JButton | btnChinhSuaLichChieu | Text: Chỉnh sửa lịch chiếu  Cursor: Hand Cursor |
| 27 | Jtabel | tblPhim | GridColor: [169,197,80]  SelectionBackground: [61,144,106]  SelectionForeground: [244,244,244] |
| 28 | JPanel | pnlThaoTac | GridLayout |
| 29 | JButton | btnThem | Text: Thêm  Icon: icons8\_add\_24px\_1.png |
| 30 | JButton | btnSua | Text: Sửa  Icon: icons8\_refresh\_24px\_1.png |
| 31 | JButton | btnXoa | Text: Xóa  Icon: icons8\_remove\_24px\_1.png |
| 32 | JButton | btnMoi | Text: Mới  Icon: icons8\_new\_copy\_24px\_1.png |
| 33 | JButton | btnXemDoanhThu | Text: Xem doanh thu  Cursor: Hand Cursor |

#### Thiết kế QLProfileFrame



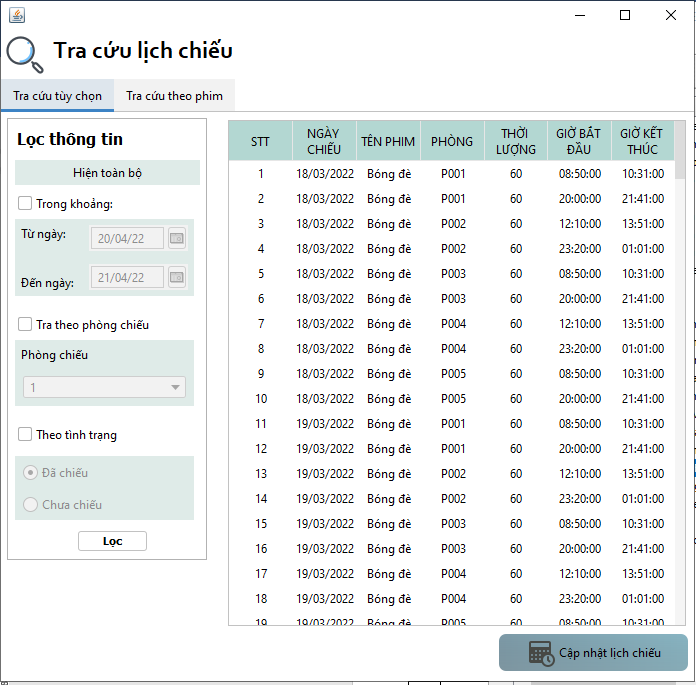
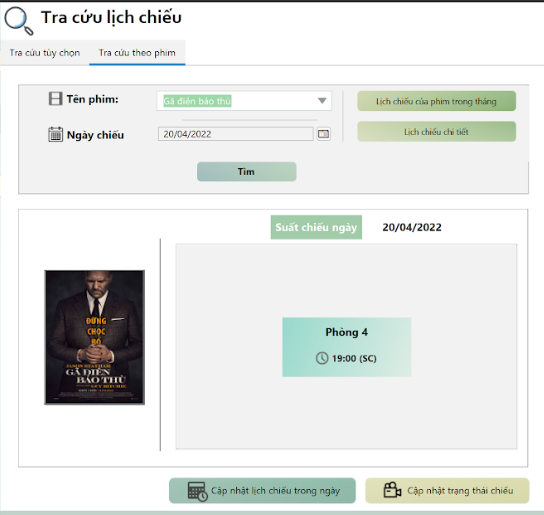
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | QLProfileFra | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JPanel | pnlContent | Background: [153,153,153] |
| 3 | JPanel | pnlUpdateProfile | Background: [227,228,207] |
| 4 | JLabel | lblThongTinCaNhan | Text: Thông tin cá nhân  Foreground: [0,102,102] |
| 5 | JLayeredPane | pnlJinh | Null |
| 6 | JLabel | lblHover | Text: Double click để chọn hình  Border: LineBorder |
| 7 | JLabel | lblHinh | Border: LineBorder |
| 8 | JLabel | lblClear | Icon: refresh.png |
| 9 | JLabel | lblHoTen | Text: Họ tên  Foreground: [3,85,79] |
| 10 | JtextField | txtTen |  |
| 11 | JLabel | lblNgaySinh | Text: Ngày sinh |
| 12 | JDateChooser | dcsNgaySinh | DateFormatString: dd/MM/yyyy |
| 13 | JLabel | lblEmail | Text: Email  Foreground: [3,85,79] |
| 14 | JtextField | txtEmail |  |
| 15 | JLabel | lblDienThoai | Text: Điện thoại  Foreground: [3,85,79] |
| 16 | JFormattedTextField | txtPhone | FormatterFactory: AbstracFormatterFactory |
| 17 | JLabel | lblGioiTinh | Text: Giới tính  Foreground: [3,85,79] |
| 18 | JradioButton | rdoNam | Text: Nam  ButtonGroup: bgrGioiTinh |
| 19 | JradioButton | rdoNu | Text: Nữ  ButtonGroup: bgrGioiTinh |
| 20 | JPanel | pnlProfile | Background: [255,255,255] |
| 21 | JLabel | lblHoTen | Text: Họ tên  Foreground: [3,85,79] |
| 22 | JLabel | lblTen |  |
| 23 | JLabel | lblNgaysinh | Text: Ngày sinh  Foreground: [3,85,79] |
| 24 | JLabel | lblNgaySinh |  |
| 25 | JLabel | lblemail | Text: Email  Foreground: [3,85,79] |
| 26 | JLabel | lblEmail |  |
| 27 | JLabel | lblDienthoai | Text: Điện thoại  Foreground: [3,85,79] |
| 28 | JLabel | lblPhone |  |
| 29 | JLabel | lblGioitinh | Text: Giới tính  Foreground: [3,85,79] |
| 30 | JLabel | lblGioiTinh |  |
| 31 | JLabel | lblHinh | Border. Highlight Color: [0,153,153]  Border. Shadow Color: [0,204,204] |
| 32 | JButton | btnCapNhat | Text: Cập nhật |
| 33 | JPanel | pnlChangePassword |  |
| 34 | JPanel | pnlDoiMK | Null |
| 35 | JLabel | lblDoiMK | Text: Đổi mật khẩu  Icon: icons8\_secure\_24px.png  Background: [0,102,102]  Foreground: [0,102,102] |
| 36 | JLabel | lblicon | Icon: icons8\_secure\_24px.png |
| 37 | JLabel | lblMatKhauCu | Text: Mật khẩu cũ  Foreground: [3,85,79] |
| 38 | JPasswordField | txtMatKhauCu |  |
| 39 | JLabel | lblHideMK | ToolTipText: Nhấn giữ để hiện Password |
| 40 | JLabel | lblMatKhauMoi | Text: Mật khẩu mới  Foreground: [3,85,79] |
| 41 | JPasswordField | txtMatKhauMoi |  |
| 42 | JLabel | lblHideMKMoi | ToolTipText: Nhấn giữ để hiện Password |
| 43 | JLabel | lblXacNhan | Text: Xác nhận  Foreground: [3,85,79] |
| 44 | JPasswordField | txtXacNhan |  |
| 45 | JLabel | lblHideXacNhan | ToolTipText: Nhấn giữ để hiện Password |
| 46 | JButton | btnReset | Text: Reset |
| 47 | JButton | btnDoiMK | Text: Đổi mật khẩu |

#### Thiết kế QLSuatChieuFrame



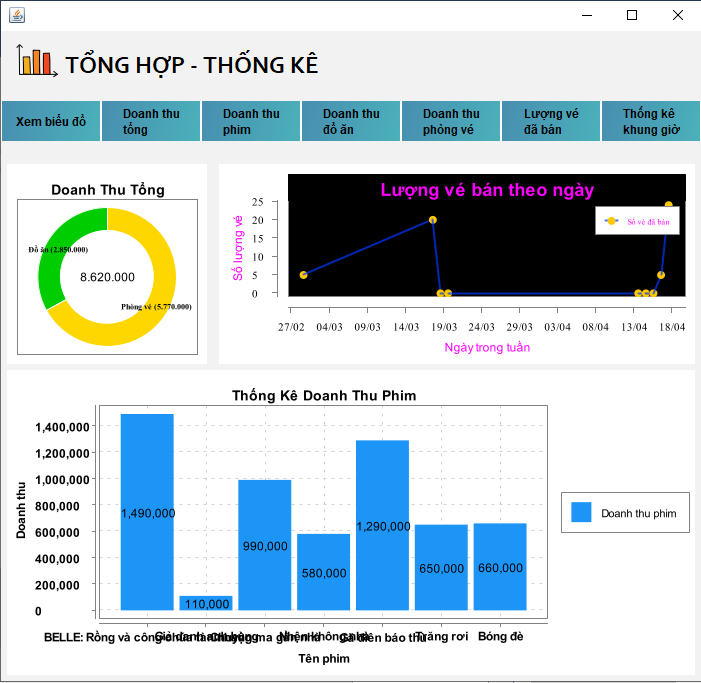
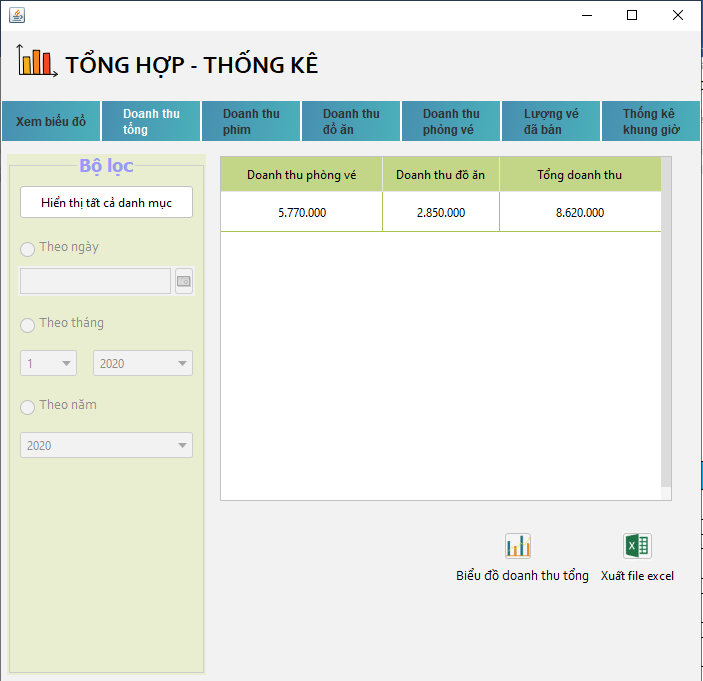
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | QLSuatChieuFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE  BorderLayout |
| 2 | JPanel | pnlContent | Background: [255,255,255] |
| 3 | JLabel | lblTitle | Text: Quản lý suất chiếu  Icon: icons8\_planner\_48px.png |
| 4 | JPanel | pnlLoc |  |
| 5 | JButton | btnNgayFilter | Text: Lọc theo ngày |
| 6 | JdateChooser | dcNgayFilter | DateFormatString: dd/MM/yyyy |
| 7 | JPanel | pnlLeft | Border.Highlight Color: [153,153,153]  Border.Shadow Color: [153,153,153] |
| 8 | JPanel | pnlThongTin | Background: [228,230,215]  GridLayout |
| 9 | JLayeredPane | pnlNgay |  |
| 10 | Jlabel | lblNgay | Text: Ngày chiếu |
| 11 | JdateChooser | dcNgay | DateFormatString: dd/MM/yyyy |
| 12 | JLayeredPane | pnlPhong |  |
| 13 | Jlabel | lblPhong | Text: Phòng chiếu |
| 14 | JComboBox | cboPhongChieu |  |
| 15 | JLayeredPane | pnlPhim |  |
| 16 | Jlabel | lblPhim | Text: Phim |
| 17 | JComboBox | cboPhim |  |
| 18 | JPanel | pnlGio | AbsoluteLayout |
| 19 | Jlabel | lblGio | Text: Giờ chiếu |
| 20 | Jspinner | spnHour |  |
| 21 | JLabel | lblPhanCach | Text: “:” |
| 22 | Jspinner | spnMintue |  |
| 23 | JLayeredPane | pnlThoiLuong |  |
| 24 | Jlabel | lblThoiluong | Text: Thòi lượng |
| 25 | Jlabel | lblThoiluong |  |
| 26 | JLayeredPane | pnlGioKetThuc |  |
| 27 | Jlabel | lblGioKT | Text: Giờ kết thúc |
| 28 | JLabel | lblGioKetThuc |  |
| 29 | JPanel | pnlControButton | GridLayout |
| 30 | JButton | btnFirst | Icon: icons8\_skip\_to\_start\_26px.png  Cursor: Hand Cursor |
| 31 | JButton | btnPre | Icon: icons8\_sort\_left\_26px\_5.png  Cursor: Hand Cursor |
| 32 | JButton | btnNext | Icon: icons8\_sort\_right\_26px\_2.png  Cursor: Hand Cursor |
| 33 | JButton | btnLast | Icon: icons8\_end\_26px.png  Cursor: Hand Cursor |
| 34 | JButton | btnSua | Text: Sửa  Icon: icons8\_refresh\_24px\_1.png  Cursor: Hand Cursor |
| 35 | JButton | btnXoa | Text: Xóa  Icon: icons8\_remove\_24px\_1.png  Cursor: Hand Cursor |
| 36 | JComboBox | cboPhòngilter |  |
| 37 | JPanel | pnlTabel |  |
| 38 | JToggleButton | btnChangeState | Text: Thêm lịch chiếu  Background: [179,206,204]  Cursor: Hand Cursor |

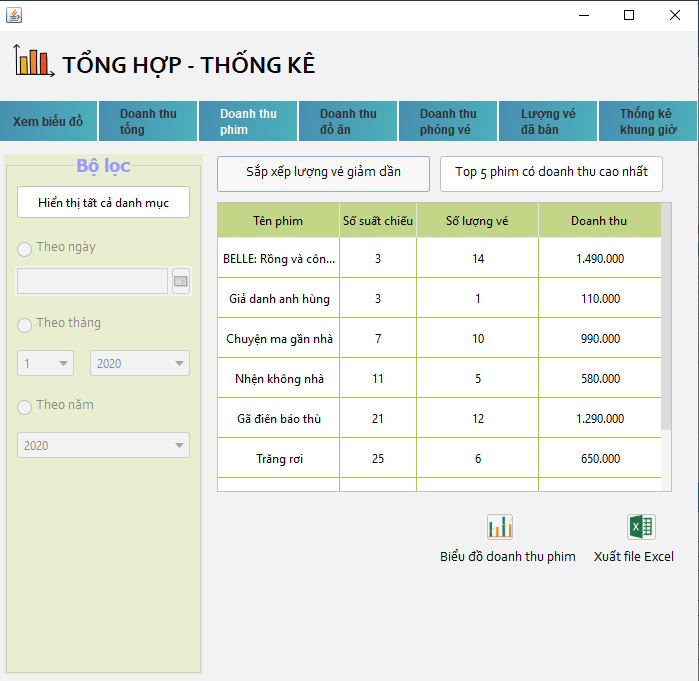
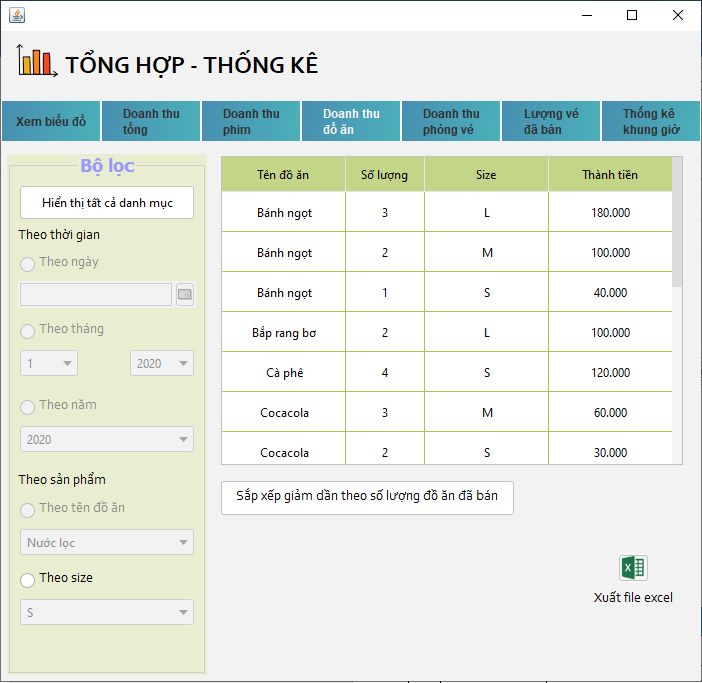
#### Thiết kế TraCuuLichChieuFrame

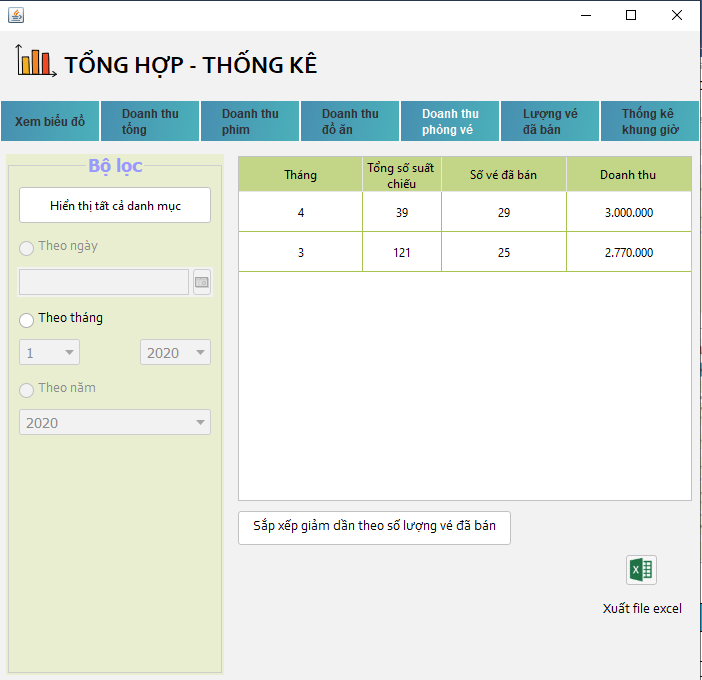
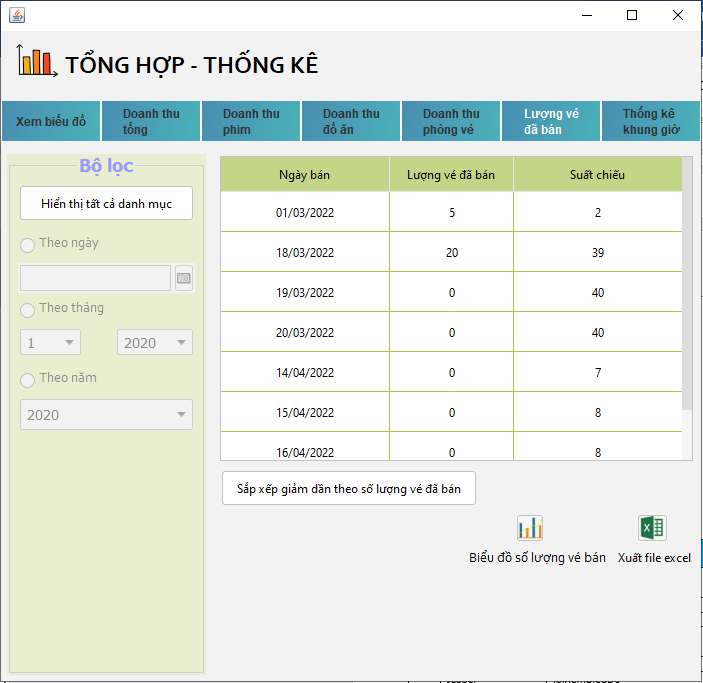
 

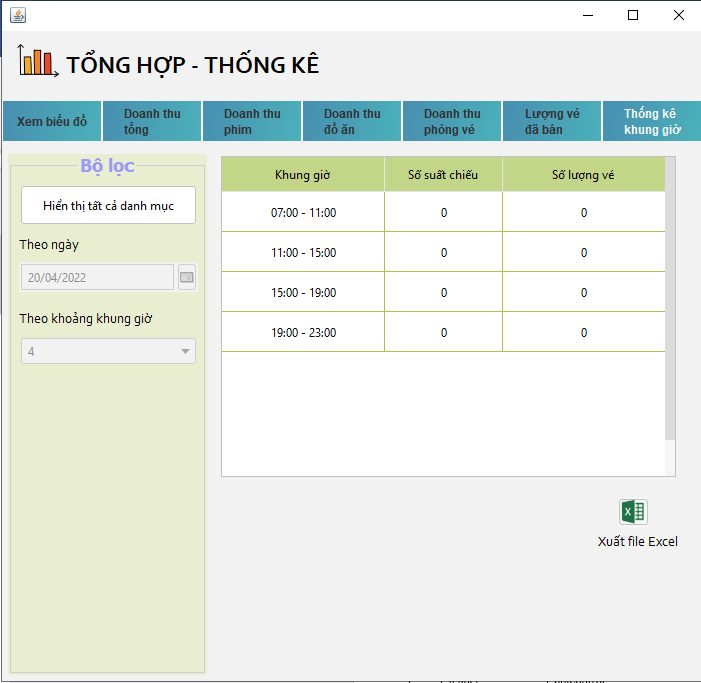
|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | TraCuuLichChieuFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JPanel | pnlTraCuuLichChieu | BorderLayout |
| 3 | JLabel | lblTracuulichchieu | Text: Tra cứu lịch chiếu  Icon: icons8\_search\_48px.png |
| 4 | JTabbedPane | tabsTracuulichchieu |  |
| 5 | JPanel | pnlTuyChon | Tab Title: Tra cứu tùy chọn |
| 6 | JPanel | pnlLoc | Background: [255,255,255]  Border: EtchedBorder  Foreground: [204,204,204] |
| 7 | JLabel | lblLocThongTin | Text: Lọc thông tin |
| 8 | JToggleButton | btnFilterPhim | Text: Hiện theo phim |
| 9 | JComboBox | cboPhimTuyChon |  |
| 10 | JLayeredPane | pnlTrongKhoang |  |
| 11 | JCheckBox | chkTrongKhoang | Text: Trong khoảng  ButtonGroup: btgTraCuuTuyChon |
| 12 | JPanel | pnlNgay | Background: [255,255,204]  Border: LineBorder |
| 13 | JLabel | lblTuNgay | Text: Từ ngày |
| 14 | JdateChooser | dscNgayBatDau |  |
| 15 | JLabel | lblDenNgay | Text: Đến ngày |
| 16 | JdateChooser | dscNgayKetThuc |  |
| 17 | JLayeredPane | pnlPhong |  |
| 18 | JCheckBox | chkTraTheoPhongChieu | Text: Tra theo phòng chiếu  ButtonGroup: btgTraCuuTuyChon |
| 19 | JPanel | pnlPhongChieu | Background: [255,255,204]  Border: LineBorder |
| 20 | JLabel | lblPhongChieu | Text: Phòng chiếu |
| 21 | JComboBox | cboPhongChieu |  |
| 22 | JLayeredPane | pnlTinhtrang |  |
| 23 | JCheckBox | chkTheoTinhTrang | Text: Tra tình trạng  ButtonGroup: btgTraCuuTuyChon |
| 24 | JPanel | pnlTinhTrang | Background: [255,255,204]  Border: LineBorder |
| 25 | JRadioButton | rdoDangChieu | text: Đang chiếu  ButtonGroup: btgTinhTrang |
| 26 | JRadioButton | rdoNgungChieu | text: Ngưng chiếu  ButtonGroup: btgTinhTrang |
| 27 | JButton | btnLoc | Text: Lọc |
| 28 | JTabel | tblTraCuuTuyChon | Text: Cập nhật lịch chiếu |
| 29 | JPanel | pnlTheoPhim | Tab Title: Tra cứu theo phim |
| 30 | JLayeredPane | pnlPhim |  |
| 31 | JLabel | lblTenPhim | Text: Tên phim |
| 32 | JComboBox | cboTenPhim |  |
| 33 | JLabel | lblNgayChieu | Text: Ngày chiếu |
| 34 | JdateChooser | dscNgayChieu |  |
| 35 | JButton | btnLichChieuPhim | Text: Lịch chiếu của phim trong tháng  Background: [102,255,102] |
| 36 | JButton | btnLichChieuChiTiet | Text: Lịch chiếu chi tiết  Background: [102,255,102] |
| 37 | JButton | btnTim | Text: Tìm |
| 38 | JPanel | pnlDssuatchieu | Border: EtchedBorder |
| 39 | JLabel | kbkTenNgayChieu |  |
| 40 | JLabel | lblPhim |  |
| 41 | Jseparator | sptPhanCach | Foreground: [0,0,0]  Orientation: VERTICAL |
| 42 | JPanel | pnlDsPhim | Border: EtchedBorder  GridLayout |
| 43 | JButton | btnChinhSuaSuatChieuCuaPhim | Text: Chỉnh sửa suất chiếu của phim  Background: [102,255,102] |

#### Thiết kế TongHopThongKeFrame



|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **ĐIỀU KHIỂN** | **TÊN** | **GIÁ TRỊ** |
| 1 | JFrame | TongHopThongKeFrame | DefaultCloseOperation: EXIT\_ON\_CLOSE |
| 2 | JPanel | pnlMenu | Null |
| 3 | JLabel | lblTongHopThongKe | Text: TỔNG HỢP - THỐNG KÊ |
| 4 | JPanel | pnlControl | Cursor: Default Cursor  GridLayout |
| 5 | JPanel | pnlBD | Background: [102,102,102] |
| 6 | JLabel | lblXemBieuDo | Text: Xem biểu đồ  Cursor: Hand Cursor |
| 7 | JPanel | pnlTong | Background: [102,102,102] |
| 8 | JLabel | lblDTTong | Text: Doanh thu tổng  Cursor: Hand Cursor |
| 9 | JPanel | pnlPhim | Background: [102,102,102] |
| 10 | JLabel | lblDTPhim | Text: Doanh thu phim  Cursor: Hand Cursor |
| 11 | JPanel | pnlPV | Background: [102,102,102] |
| 12 | JLabel | lblPTPhongVe | Text: Phòng Vé  Cursor: Hand Cursor |
| 13 | JPanel | pnlLV | Background: [102,102,102] |
| 14 | JLabel | lblDTLuongVe | Text: Lượng vé đã bán  Cursor: Hand Cursor |
| 15 | JPanel | pnlKG | Background: [102,102,102] |
| 16 | JLabel | lblKhungGio | Text: Khung giờ  Cursor: Hand Cursor |
| 17 | JTabbedPane | tabsTonghopthongke |  |
| 18 | JPanel | pnlBieuDo | Tab Title: BieuDo |
| 19 | JPanel | pnlBD\_DTTong | BorderLayout |
| 20 | JPanel | pnlBD\_DTLuongVe | BorderLayout |
| 21 | JPanel | pnlBD\_DTPhim | BorderLayout |
| 22 | JPanel | pnlDTTong |  |
| 23 | JPanel | pnl1 | Border. Title: Bộ lọc  Border. Color; [153,153,255]  Border. Justification: Center |
| 24 | JToggleButton | btnHienThiTatCaDanhMuc | Text: Hiển thị tất cả danh mục |
| 25 | JRadioButton | rdoTheoNgay\_DTTong | Text: Theo ngày  ButtonGroup: btgDoanhThuTong |
| 26 | JdateChooser | dscTheoNgay\_DTTong | DateFormatString: dd/MM/yyyy |
| 27 | JRadioButton | rdoTheoThang\_DTTong | Text: Theo tháng  ButtonGroup: btgDoanhThuTong |
| 28 | JcomboBox | cboThangTheoThang\_DTTong |  |
| 29 | JComboBox | cboNamTheoThang\_DTTong |  |
| 30 | JRadioButton | rdoTheoNam\_DTTong | Text: Theo năm  ButtonGroup: btgDoanhThuTong |
| 31 | JComboBox | cboTheoNam\_DTTong |  |
| 32 | JTable | tblDoanhThuTong |  |
| 33 | JButton | btnBieuDoNgay\_DTTong | Icon: bieudo.jpg |
| 34 | JLabel | lblBieuDoNgayDTT | Text: Biểu đồ doanh thu tổng |
| 35 | JButton | btnXuatExcel\_DTTong | Icon: xuatfileexel.jpg |
| 36 | JLabel | lblXuatExcelDTT | Text: Xuất file excel |
| 37 | JPanel | pnlDTPhim |  |
| 38 | JPanel | pnl3 | Border. Title: Bộ lọc  Border. Color; [153,153,255]  Border. Justification: Center |
| 39 | JToggleButton | btnHienThiTatCaDanhMuc\_DTPhim | Text: Hiển thị tất cả danh mục |
| 40 | JRadioButton | rdoTheoNgay\_DTPhim | Text: Theo ngày  ButtonGroup: btgPhim |
| 41 | JdateChooser | dscTheoNgay\_DTPhim | DateFormatString: dd/MM/yyyy |
| 42 | JRadioButton | rdoTheoThang\_DTPhim | Text: Theo tháng  ButtonGroup: btgPhim |
| 43 | JcomboBox | cboThangTheoThang\_DTPhim |  |
| 44 | JComboBox | cboNamTheoThang\_DTPhim |  |
| 45 | JRadioButton | rdoTheoNam\_DTPhim | Text: Theo năm  ButtonGroup: btgPhim |
| 46 | JComboBox | cboTheoNam\_DTPhim |  |
| 47 | JButton | btnSLVeGiam | Text: Sắp xếp lượng vé giảm dần |
| 48 | JButton | btnTop5DoanhThu | Text: Top 5 phim có doanh thu cao nhất |
| 49 | JTable | tblDoanhThuPhim |  |
| 50 | JButton | btnBieuDoNgay\_DTPhim | Icon: bieudo.jpg |
| 51 | JLabel | lblBieuDoNgayDTPhim | Text: Biểu đồ doanh thu phim |
| 52 | JButton | btnXuatExcel\_DTPhim | Icon: xuatfileexel.jpg |
| 53 | JLabel | lblXuatExcelDTPhim | Text: Xuất file excel |
| 54 | JPanel | pnlDTDoAn |  |
| 55 | JPanel | Pnl6 | Border. Title: Bộ lọc  Border. Color; [153,153,255]  Border. Justification: Center |
| 56 | JToggleButton | btnHienThiTatCaDanhMuc\_DTDoAn | Text: Hiển thị tất cả danh mục |
| 57 | JLabel | lblThgian | Text: Theo thời gian |
| 58 | JRadioButton | rdoTheoNgay\_DTDoAn | Text: Theo ngày  ButtonGroup: btgDoAn\_TG |
| 59 | JdateChooser | dscTheoNgay\_DTDoAn | DateFormatString: dd/MM/yyyy |
| 60 | JRadioButton | rdoTheoThang\_DTDoAn | Text: Theo tháng  ButtonGroup: btgDoAn\_TG |
| 61 | JcomboBox | cboThangTheoThang\_DTDoAn |  |
| 62 | JComboBox | cboNamTheoThang\_DTDoAn |  |
| 63 | JRadioButton | rdoTheoNam\_DTDoAn | Text: Theo năm  ButtonGroup: btgDoAn\_TG |
| 64 | JComboBox | cboTheoNam\_ DTDoAn |  |
| 65 | JLabel | lblSp | Text: Theo sản phẩm |
| 66 | JRadioButton | rdoTheoTenDoAn | Text: Theo tên đồ ăn  ButtonGroup: btgDoAn\_SP |
| 67 | JComboBox | cboTheoTenDoAn |  |
| 68 | JRadioButton | rdoTheoSize | Text: Theo size  ButtonGroup: btgDoAn\_SP |
| 69 | JComboBox | cboTheoSize |  |
| 70 | JTable | tblDoanhThuDoAn |  |
| 71 | JButton | btnBieuDoNgay\_DTDoAn | Icon: bieudo.jpg |
| 72 | JLabel | lblBieuDoNgayDTTDA | Text: Biểu đồ doanh thu tổng |
| 73 | JButton | btnXuatExcel\_DTDoAn | Icon: xuatfileexel.jpg |
| 74 | JLabel | lblXuatExcelDTDA | Text: Xuất file excel |
| 75 | JButton | btnSapXepTheoLuongDoAnDaBan | Text: Sắp xếp giảm dần theo số lượng đồ ăn đã bán |
| 76 | JPanel | pnlPhongVe |  |
| 77 | JPanel | Pnl2 | Border. Title: Bộ lọc  Border. Color; [153,153,255]  Border. Justification: Center |
| 78 | JToggleButton | btnHienThiTatCaDanhMuc\_DTPhongVe | Text: Hiển thị tất cả danh mục |
| 79 | JRadioButton | rdoTheoNgay\_ DTPhongVe | Text: Theo ngày  ButtonGroup: btgDoAn\_TG |
| 80 | JdateChooser | dscTheoNgay\_ DTPhongVe | DateFormatString: dd/MM/yyyy |
| 81 | JRadioButton | rdoTheoThang\_ DTPhongVe | Text: Theo tháng  ButtonGroup: btgDoAn\_TG |
| 82 | JcomboBox | cboThangTheoThang\_ DTPhongVe |  |
| 83 | JComboBox | cboNamTheoThang\_ DTPhongVe |  |
| 84 | JRadioButton | rdoTheoNam\_ DTPhongVe | Text: Theo năm  ButtonGroup: btgDoAn\_TG |
| 85 | JComboBox | cboTheoNam\_ DTPhongVe |  |
| 86 | JTable | tblDoanhThuPhongVe |  |
| 87 | JButton | btnXuatExcel\_ DTPhongVe | Icon: xuatfileexel.jpg |
| 88 | JLabel | lblXuatExcelDTPV | Text: Xuất file excel |
| 89 | JButton | btnSapXepTheoLuongVeDaBanPV | Text: Sắp xếp giảm dần theo số lượng vé đã bán |
| 90 |  |  |  |
| 91 | JPanel | pnlVeTheoNgay |  |
| 92 | JPanel | Pnl4 | Border. Title: Bộ lọc  Border. Color; [153,153,255]  Border. Justification: Center |
| 93 | JToggleButton | btnHienThiTatCaDanhMuc\_DTLuongVe | Text: Hiển thị tất cả danh mục |
| 94 | JRadioButton | rdoTheoNgay\_LuongVeBD | Text: Theo ngày  ButtonGroup: btgLuongVe |
| 95 | JdateChooser | dscTheoNgay\_ LuongVeBD | DateFormatString: dd/MM/yyyy |
| 96 | JRadioButton | rdoTheoThang\_ LuongVeBD | Text: Theo tháng  ButtonGroup: btgLuongVe |
| 97 | JcomboBox | cboThangTheoThang\_ LuongVeBD |  |
| 98 | JComboBox | cboNamTheoThang\_ LuongVeBD |  |
| 99 | JRadioButton | rdoTheoNam\_ LuongVeBD | Text: Theo năm  ButtonGroup: btgLuongVe |
| 100 | JComboBox | cboTheoNam\_ LuongVeBD |  |
| 101 | JButton | btnSapXepTheoLuongVeDaBan | Text: Sắp xếp giảm dần theo số lượng vé đã bán |
| 101 | JTable | tblDoanhThuPhim |  |
| 103 | JButton | btnBieuDoNgay\_ LuongVeBD | Icon: bieudo.jpg |
| 104 | JLabel | lblBieuDoNgayLuongVeDB | Text: Biểu đồ số lượng vé đã bán |
| 105 | JButton | btnXuatExcel\_ LuongVeBD | Icon: xuatfileexel.jpg |
| 106 | JLabel | lblXuatExcelLuongVrDB | Text: Xuất file excel |
| 107 |  |  |  |
| 108 | JPanel | pnlKhungGio |  |
| 109 | JPanel | Pnl5 | Border. Title: Bộ lọc  Border. Color; [153,153,255]  Border. Justification: Center |
| 110 | JToggleButton | btnHienThiTatCaDanhMuc\_KhungGio | Text: Hiển thị tất cả danh mục |
| 111 | JLabel | lblTheoNgay | Text: Theo ngày |
| 112 | JdateChooser | dscTheoNgay\_ KhungGio | DateFormatString: dd/MM/yyyy |
| 113 | JLabel | lblTheo KhungGio | Text: Theo khoảng khung giờ |
| 114 | JcomboBox | cbo ChonKhungGio |  |
| 115 | JButton | btnXuatExcel\_ LuongVeBD | Icon: xuatfileexel.jpg |
| 116 | JLabel | lblXuatExcelLuongVrDB | Text: Xuất file excel |

# Thực hiện viết mã

## Viết mã tạo CSDL

### Tạo CSDL

|  |
| --- |
| /\*  DATABASE: CINEBOOST - PRO1041 - IT17101 - NHOM 3  NGAY UPDATE: 20/03/2022  NGAY TAO: 17/03/2022  \*/  --TAO DATABASE  CREATE DATABASE CINEBOOST  GO  USE CINEBOOST  GO  -- CREATE TABLE NhanVien  CREATE TABLE NHANVIEN  (  ID\_NV NVARCHAR(10) PRIMARY KEY, --QL00001 - NV00001  HOTEN NVARCHAR(50),  NGAYSINH DATE,  GIOITINH BIT,  SODT NVARCHAR(15),  EMAIL NVARCHAR(50) UNIQUE NOT NULL,  TENTK NVARCHAR(30) UNIQUE NOT NULL,  MATKHAU NVARCHAR(15), --6 DEN 10 KI TU  ANH nvarchar(100),  VAITRO BIT  )  GO  --CREATE TABLE ThanhVien  CREATE TABLE THANHVIEN  (  ID\_TV NVARCHAR(16) PRIMARY KEY, --TV00001  HOTEN NVARCHAR(50),  NGAYSINH DATE,  GIOITINH BIT,  SODT NVARCHAR(15),  EMAIL NVARCHAR(50),  MAKH NVARCHAR(12),  constraint UQ\_SoDT\_TV UNIQUE(SODT)  )  GO  --CREATE TABLE LoaiDoAn  CREATE TABLE LOAIDOAN  (  ID NVARCHAR(4) PRIMARY KEY, --TA OR DU  TEN NVARCHAR(100)  )  GO  --CREATE TABLE KICHCO  CREATE TABLE KICHCO  (  ID NCHAR(1) PRIMARY KEY --(S,M,L)  )  GO  --CREATE TABLE DOAN  CREATE TABLE DOAN  (  ID\_DOAN NVARCHAR(16) PRIMARY KEY, --TA00001 OR DU  TEN NVARCHAR(100),  ID\_LOAI NVARCHAR(4),  CONSTRAINT FK\_DOAN\_LOAI FOREIGN KEY (ID\_LOAI) REFERENCES LOAIDOAN(ID) ON DELETE NO ACTION ON UPDATE CASCADE    )  GO  --CREATE TABLE KICHCODOAN  CREATE TABLE KICHCODOAN  (  ID\_KCDA INT IDENTITY(1,1) PRIMARY KEY,  ID\_DOAN NVARCHAR(16) NOT NULL,  ID\_KICHCO NCHAR(1) NOT NULL,  GIA DECIMAL(6,2) NOT NULL,  CONSTRAINT FK\_KICHCO\_DOAN FOREIGN KEY (ID\_DOAN) REFERENCES DOAN(ID\_DOAN) ON DELETE NO ACTION ON UPDATE CASCADE,  CONSTRAINT FK\_KC\_KCDA FOREIGN KEY (ID\_KICHCO) REFERENCES KICHCO(ID) ON DELETE NO ACTION ON UPDATE CASCADE,  CONSTRAINT UQ\_DA\_KC UNIQUE (ID\_DOAN, ID\_KICHCO)  )  GO  --CREATE TABLE DONDOAN  CREATE TABLE DONDOAN  (  ID\_DONDA NVARCHAR(20) PRIMARY KEY, --DA00001  )  GO  --CREATE TABLE DOANCT  CREATE TABLE DOANCT  (  ID\_DOANCT INT IDENTITY(1,1) PRIMARY KEY,  ID\_KCDA INT,  ID\_DONDA NVARCHAR(20),  SOLUONG INT NOT NULL,  CONSTRAINT FK\_CT\_KCDA FOREIGN KEY (ID\_KCDA) REFERENCES KICHCODOAN(ID\_KCDA) ON DELETE SET NULL ON UPDATE CASCADE,  CONSTRAINT FK\_CT\_DONDA FOREIGN KEY (ID\_DONDA) REFERENCES DONDOAN(ID\_DONDA) ON DELETE CASCADE ON UPDATE CASCADE  )  GO  --CREATE TABLE PHIM  CREATE TABLE PHIM  (  ID\_PHIM NVARCHAR(15) PRIMARY KEY, --FI00001  TEN NVARCHAR(255),  POSTER NVARCHAR(100),  THOILUONG SMALLINT,  THELOAI NVARCHAR(50),  NGONNGU NVARCHAR(20),  NGAY\_KC DATE,  TOMTAT NVARCHAR(MAX),  TRANGTHAI BIT,  ID\_NV NVARCHAR(10),  CONSTRAINT FK\_PHIM\_NV FOREIGN KEY (ID\_NV) REFERENCES NHANVIEN(ID\_NV) ON UPDATE CASCADE  )  GO  --CREATE TABLE PHONGCHIEU  CREATE TABLE PHONGCHIEU  (  ID\_PHONG NCHAR(4) PRIMARY KEY, --P001  SOLUONGDAY INT NOT NULL,  SOLUONGCOT INT NOT NULL  )  GO  --CREATE TABLE SUATCHIEU  CREATE TABLE SUATCHIEU  (  ID\_SUAT NVARCHAR(30) PRIMARY KEY, --SC00001  NGAYCHIEU DATE NOT NULL,  ID\_PHONG NCHAR(4) NOT NULL,  ID\_PHIM NVARCHAR(15) NOT NULL,  GIOBATDAU TIME NOT NULL,  GIOKETTHUC TIME NOT NULL,  ID\_NV NVARCHAR(10),  CONSTRAINT FK\_SC\_PHONG FOREIGN KEY (ID\_PHONG) REFERENCES PHONGCHIEU(ID\_PHONG) ON DELETE NO ACTION ON UPDATE CASCADE,  CONSTRAINT FK\_SC\_PHIM FOREIGN KEY (ID\_PHIM) REFERENCES PHIM(ID\_PHIM) ON DELETE CASCADE ON UPDATE CASCADE ,  CONSTRAINT FK\_SC\_NV FOREIGN KEY (ID\_NV) REFERENCES NHANVIEN(ID\_NV) on delete set null  )  GO  --CREATE TABLE LOAIGHE  CREATE TABLE LOAIGHE  (  ID\_LOAIGHE NCHAR(2) PRIMARY KEY, --TH OR VP  PHUTHU TINYINT NOT NULL,  TENLOAI NVARCHAR(30) NOT NULL  )  GO  --CREATE TABLE GHE  CREATE TABLE GHE  (  ID\_GHE NVARCHAR(12) PRIMARY KEY, --GS01A01 -- Tien to GS + SO PHONG + KI TU DAY + KI TU COT  VITRIDAY TINYINT NOT NULL,  VITRICOT TINYINT NOT NULL,  ID\_PHONGCHIEU NCHAR(4) NOT NULL,  ID\_LOAIGHE NCHAR(2) NOT NULL,  CONSTRAINT FK\_GHE\_PHONG FOREIGN KEY (ID\_PHONGCHIEU) REFERENCES PHONGCHIEU (ID\_PHONG) ON DELETE CASCADE ON UPDATE CASCADE,  CONSTRAINT FK\_GHE\_LOAI FOREIGN KEY (ID\_LOAIGHE) REFERENCES LOAIGHE (ID\_LOAIGHE) ON DELETE NO ACTION ON UPDATE CASCADE  )  GO  --CREATE TABLE LOAIVE  CREATE TABLE LOAIVE  (  ID\_LOAIVE NCHAR(2) PRIMARY KEY, --TE OR NL  TEN NVARCHAR(30),  GIA DECIMAL(6,2)  )  GO  --CREATE TABLE DONVE  CREATE TABLE DONVE  (  ID\_DONVE NVARCHAR(20) PRIMARY KEY, --DV00001  )  GO  --CREATE TABLE VEDAT  CREATE TABLE VEDAT  (  ID\_VEDAT INT IDENTITY(1,1) PRIMARY KEY,  ID\_LOAIVE NCHAR(2) NOT NULL,  ID\_GHE NVARCHAR(12) NOT NULL,  ID\_DONVE NVARCHAR(20) NOT NULL,  ID\_SUAT NVARCHAR(30) NOT NULL,  CONSTRAINT FK\_VEDAT\_LOAIVE FOREIGN KEY (ID\_LOAIVE) REFERENCES LOAIVE (ID\_LOAIVE) ON DELETE NO ACTION ON UPDATE CASCADE,  CONSTRAINT FK\_VEDAT\_GHE FOREIGN KEY (ID\_GHE) REFERENCES GHE (ID\_GHE) ON DELETE CASCADE ON UPDATE CASCADE,  CONSTRAINT FK\_VEDAT\_DONDATVE FOREIGN KEY (ID\_DONVE) REFERENCES DONVE (ID\_DONVE) ON DELETE CASCADE ON UPDATE CASCADE,  CONSTRAINT FK\_VEDAT\_SUAT FOREIGN KEY (ID\_SUAT) REFERENCES SUATCHIEU (ID\_SUAT) ON DELETE CASCADE, --need delete ON DELETE CASCADE  CONSTRAINT U\_G\_S UNIQUE(ID\_GHE, ID\_SUAT)  )  GO  -----------------------------------------  --CREATE TABLE DONTHANHTOAN  CREATE TABLE DONTHANHTOAN  (  ID\_DONTT INT IDENTITY(1,1) PRIMARY KEY,  ID\_DONDA NVARCHAR(20),  ID\_TV NVARCHAR(16),  ID\_NV NVARCHAR(10) NULL,  ID\_DONVE NVARCHAR(20),  NGAYDAT DATETIME NOT NULL,  TONG DECIMAL(9,3),  -- Đã chỉnh sửa  CONSTRAINT FK\_DON\_NV FOREIGN KEY (ID\_NV) REFERENCES NHANVIEN(ID\_NV) ON DELETE SET NULL ON UPDATE CASCADE,  CONSTRAINT FK\_DON\_TV FOREIGN KEY (ID\_TV) REFERENCES THANHVIEN (ID\_TV) ON DELETE SET NULL ON UPDATE CASCADE,  CONSTRAINT FK\_DON\_DA FOREIGN KEY (ID\_DONDA) REFERENCES DONDOAN (ID\_DONDA) ON DELETE NO ACTION ON UPDATE CASCADE,  CONSTRAINT FK\_DON\_VE FOREIGN KEY (ID\_DONVE) REFERENCES DONVE (ID\_DONVE) ON DELETE NO ACTION ON UPDATE CASCADE  )  GO  create table KCDAHISTORY(  id\_DOANCT int primary key,  id\_DOAN nvarchar(16),  id\_LOAI nvarchar(4),  TENLOAI nvarchar(100),  TENDA nvarchar(100),  id\_KICHCO nchar(1),  GIA decimal(6,2)  ) |

### SQL truy vấn và thao tác

#### SQL đối với bảng nhân viên

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT NHANVIEN (HOTEN, NGAYSINH, GIOITINH, SODT, EMAIL, TENTK, MATKHAU, ANH, VAITRO) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?) |
| Cập nhật theo mã | UPDATE NHANVIEN SET HOTEN =?, NGAYSINH =?, GIOITINH =?, SODT =?, EMAIL =?, TENTK =?, MATKHAU =?, ANH =?, VAITRO =? WHERE ID\_NV =? |
| Xóa theo mã | DELETE FROM NHANVIEN WHERE ID\_NV =? |
| Truy vấn tất cả | SELECT \* FROM NHANVIEN |
| Truy vấn theo mã | SELECT \* FROM NHANVIEN WHERE ID\_NV =? |

#### SQL đối với bảng Thành viên

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT THANHVIEN (HOTEN, NGAYSINH, GIOITINH, SODT, EMAIL) VALUES (?, ?, ?, ?, ?) |
| Cập nhật theo mã | UPDATE THANHVIEN SET HOTEN =?, NGAYSINH =?, GIOITINH =?, SODT =?, EMAIL WHERE ID\_TV =? |
| Xóa theo mã | DELETE FROM THANHVIEN WHERE ID\_TV =? |
| Truy vấn tất cả | SELECT \* FROM THANHVIEN |
| Truy vấn theo mã | SELECT \* FROM THANHVIEN WHERE ID\_TV =? |

#### SQL đối với bảng loại đồ ăn

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT LOAIDOAN (TEN) VALUES ( ?) |
| Cập nhật theo mã | UPDATE LOAIDOAN SET TEN =?, WHERE ID =? |
| Xóa theo mã | DELETE FROM LOAIDOAN WHERE ID =? |
| Truy vấn tất cả | SELECT \* FROM LOAIDOAN |
| Truy vấn theo mã | SELECT \* FROM LOAIDOAN WHERE ID =? |

#### SQL đối với bảng kích cỡ

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT KICHCO (ID) VALUES (?) |
| Cập nhật theo mã | UPDATE KICHCO SET ID =?, WHERE ID =? |
| Xóa theo mã | DELETE FROM KICHCO WHERE ID =? |
| Truy vấn tất cả | SELECT \* FROM KICHCO |
| Truy vấn theo mã | SELECT \* FROM KICHCO WHERE ID =? |

#### SQL đối với bảng đồ ăn

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT DOAN (TEN, ID\_LOAI) VALUES (?,?) |
| Cập nhật theo mã | UPDATE DOAN SET TEN =?, ID\_LOAI =? WHERE ID\_ DOAN =? |
| Xóa theo mã | DELETE FROM DOAN WHERE ID\_ DOAN =? |
| Truy vấn tất cả | SELECT \* FROM DOAN |
| Truy vấn theo mã | SELECT \* FROM DOAN WHERE ID\_ DOAN =? |

#### SQL đối với bảng kích cỡ đồ ăn

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT KICHCODOAN (ID\_DOAN, ID\_KICHCO, GIA) VALUES (?, ?, ?) |
| Cập nhật theo mã | UPDATE KICHCODOAN SET ID\_DOAN =?, ID\_KICHCO =?, GIA =? WHERE ID\_ KCDA =? |
| Xóa theo mã | DELETE FROM KICHCODOAN WHERE ID\_ KCDA =? |
| Truy vấn tất cả | SELECT \* FROM KICHCODOAN |
| Truy vấn theo mã | SELECT \* FROM KICHCODOAN WHERE ID\_ KCDA =? |

#### SQL đối với bảng đơn đồ ăn

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT DONDOAN () VALUES (?) |
| Cập nhật theo mã | UPDATE DONDOAN WHERE ID\_DONDA =? |
| Xóa theo mã | DELETE FROM DONDOAN WHERE ID\_ DONDA =? |
| Truy vấn tất cả | SELECT \* FROM DONDOAN |
| Truy vấn theo mã | SELECT \* FROM DONDOAN WHERE ID\_ DONDA =? |

#### SQL đối với bảng đồ ăn chi tiết

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT DOANCT (ID\_KCDA, ID\_DONDA, SOLUONG) VALUES (?, ?, ?) |
| Cập nhật theo mã | UPDATE DOANCT SET ID\_ KCDA =?, ID\_ DONDA =?, SOLUONG =? WHERE ID\_DOANCT =? |
| Xóa theo mã | DELETE FROM DOANCT WHERE ID\_ DOANCT =? |
| Truy vấn tất cả | SELECT \* FROM DOANCT |
| Truy vấn theo mã | SELECT \* FROM DOANCT WHERE ID\_ DOANCT =? |

#### SQL đối với bảng phim

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT PHIM (TEN, POSTER, THOILUONG, THELOAI, NGONNGU, NGAY\_KC, TOMTAT, TRANGTHAI, ID\_NV) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?) |
| Cập nhật theo mã | UPDATE PHIM SET TEN =?, POSTER =?, THOILUONG =?, THELOAI =?, NGONNGU =?, NGAY\_KC =?, TOMTAT =?, TRANGTHAI =?, ID\_NV =? WHERE ID\_ PHIM =? |
| Xóa theo mã | DELETE FROM PHIM WHERE ID\_ PHIM =? |
| Truy vấn tất cả | SELECT \* FROM PHIM |
| Truy vấn theo mã | SELECT \* FROM PHIM WHERE ID\_ PHIM =? |

#### SQL đối với bảng phòng chiếu

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT PHONGCHIEU (SOLUONGDAY, SOLUONGCOT) VALUES (?, ?) |
| Cập nhật theo mã | UPDATE PHONGCHIEU SET SOLUONGDAY =?, SOLUONGCOT WHERE ID\_ PHONG =? |
| Xóa theo mã | DELETE FROM PHONGCHIEU WHERE ID\_ PHONG =? |
| Truy vấn tất cả | SELECT \* FROM PHONGCHIEU |
| Truy vấn theo mã | SELECT \* FROM PHONGCHIEU WHERE ID\_ PHONG =? |

#### SQL đối với bảng suất chiếu

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT SUATCHIEU (NGAYCHIEU, ID\_PHONG, ID\_PHIM, GIOBATDAU, GIOKETTHUC, ID\_NV) VALUES (?, ?, ?, ?, ?, ?) |
| Cập nhật theo mã | UPDATE PHONGCHIEU SET NGAYCHIEU =?, ID\_PHONG =?, ID\_PHIM =?, GIOBATDAU =?, GIOKETTHUC =?, ID\_NV =? WHERE ID\_ SUAT =? |
| Xóa theo mã | DELETE FROM PHONGCHIEU WHERE ID\_ SUAT =? |
| Truy vấn tất cả | SELECT \* FROM PHONGCHIEU |
| Truy vấn theo mã | SELECT \* FROM PHONGCHIEU WHERE ID\_ SUAT =? |

#### SQL đối với bảng loại ghế

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT LOAIGHE (ID\_LOAIGHE, PHUTHU, TENLOAI) VALUES (?, ?, ?) |
| Cập nhật theo mã | UPDATE LOAIGHE SET PHUTHU =?, TEN =? WHERE ID\_ LOAIGHE =? |
| Xóa theo mã | DELETE FROM LOAIGHE WHERE ID\_ LOAIGHE =? |
| Truy vấn tất cả | SELECT \* FROM LOAIGHE |
| Truy vấn theo mã | SELECT \* FROM LOAIGHE WHERE ID\_ LOAIGHE =? |

#### SQL đối với bảng ghế

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT GHE (VITRIDAY, VITRICOT, ID\_PHONGCHIEU, ID\_LOAIGHE) VALUES (?, ?, ?, ?) |
| Cập nhật theo mã | UPDATE GHE SET VITRIDAY =?, VITRICOT =?, ID\_PHONGCHIEU =?, ID\_LOAIGHE =? WHERE ID\_ GHE =? |
| Xóa theo mã | DELETE FROM GHE WHERE ID\_ GHE =? |
| Truy vấn tất cả | SELECT \* FROM GHE |
| Truy vấn theo mã | SELECT \* FROM GHE WHERE ID\_ GHE =? |

#### SQL đối với bảng loại vé

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT LOAIVE (ID\_LOAIVE, TEN, GIA) VALUES (?, ?, ?) |
| Cập nhật theo mã | UPDATE LOAIVE SET ID\_LOAIVE =?, TEN =?, GIA =? WHERE ID\_ LOAIVE =? |
| Xóa theo mã | DELETE FROM LOAIVE WHERE ID\_ LOAIVE =? |
| Truy vấn tất cả | SELECT \* FROM LOAIVE |
| Truy vấn theo mã | SELECT \* FROM LOAIVE WHERE ID\_ LOAIVE =? |

#### SQL đối với bảng đơn vé

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT DONVE (ID\_DONVE) VALUES (?) |
| Cập nhật theo mã | UPDATE DONVE SET WHERE ID\_ DONVE =? |
| Xóa theo mã | DELETE FROM DONVE WHERE ID\_ DONVE =? |
| Truy vấn tất cả | SELECT \* FROM DONVE |
| Truy vấn theo mã | SELECT \* FROM DONVE WHERE ID\_ DONVE =? |

#### SQL đối với bảng vé đặt

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT VEDAT (ID\_LOAIVE, ID\_GHE, ID\_DONVE, ID\_SUAT) VALUES (?, ?, ?, ?) |
| Cập nhật theo mã | UPDATE VEDAT SET ID\_LOAIVE =?, ID\_GHE =?, ID\_DONVE =?, ID\_SUAT =? WHERE ID\_ VEDAT =? |
| Xóa theo mã | DELETE FROM VEDAT WHERE ID\_ VEDAT =? |
| Truy vấn tất cả | SELECT \* FROM VEDAT |
| Truy vấn theo mã | SELECT \* FROM VEDAT WHERE ID\_ VEDAT =? |

#### SQL đối với bảng đơn thanh toán

|  |  |
| --- | --- |
| **Hành động** | **Câu lệnh SQL** |
| Thêm mới | INSERT DONTHANHTOAN (ID\_DONDA, ID\_TV, ID\_NV, ID\_DONVE, NGAYDAT, TONG) VALUES (?, ?, ?, ?, ?, ?) |
| Cập nhật theo mã | UPDATE DONTHANHTOAN SET ID\_DONDA =?, ID\_TV =?, ID\_NV =?, ID\_DONVE =?, NGAYDAT =?, TONG =? WHERE ID\_ DONTT =? |
| Xóa theo mã | DELETE FROM DONTHANHTOAN WHERE ID\_ DONTT =? |
| Truy vấn tất cả | SELECT \* FROM DONTHANHTOAN |
| Truy vấn theo mã | SELECT \* FROM DONTHANHTOAN WHERE ID\_ DONTT =? |

### Các thủ tục lưu, trigger, function

#### Trigger tạo khóa chính Nhân viên

|  |
| --- |
| if(OBJECT\_ID('tg\_nv')) is not null  drop trigger tg\_nv  go  create trigger tg\_nv on NHANVIEN  INSTEAD OF INSERT  AS BEGIN  SET NOCOUNT ON;  DECLARE @HOTEN NVARCHAR(50) = (SELECT HOTEN FROM inserted)  DECLARE @NGAYSINH DATE =(SELECT NGAYSINH FROM inserted)  DECLARE @GIOITINH BIT = (SELECT GIOITINH FROM inserted)  DECLARE @SODT NVARCHAR(15) = (SELECT SODT FROM inserted)  DECLARE @EMAIL NVARCHAR(50) = (SELECT EMAIL FROM INSERTED)  DECLARE @TENTK NVARCHAR(30) = (SELECT TENTK FROM INSERTED)  DECLARE @MATKHAU NVARCHAR(15) = (SELECT MATKHAU FROM INSERTED)  DECLARE @VT BIT = (SELECT VAITRO FROM INSERTED)  DECLARE @ANH NVARCHAR(100) = (SELECT ANH FROM inserted)  --SO ID GAN NHAT DUOC THEM VAO TRONG BANG  DECLARE @PRE\_ID NVARCHAR(16) = (SELECT TOP 1 CAST(REPLACE(ID\_NV,left(ID\_NV,2),'') AS INT) IDN FROM NHANVIEN WHERE VAITRO LIKE @VT ORDER BY IDN DESC)  DECLARE @NEXT INT  -- NEU LA RECORD DAU TIEN SO LA 1 HOAC @PRE\_ID + 1  IF(@PRE\_ID IS NULL) SET @NEXT = 1  ELSE SET @NEXT = @PRE\_ID + 1  -- TAO ID TU TANG CHO HANG CHUAN BI THEM  DECLARE @CUR\_ID NVARCHAR(16) = CASE WHEN @VT = 1 THEN ('QL' + RIGHT('0000'+CONVERT(NVARCHAR(10),@NEXT),5))  WHEN @VT = 0 THEN ('NV'+ RIGHT('0000'+CONVERT(NVARCHAR(10),@NEXT),5)) END  -- INSERT  INSERT INTO NHANVIEN VALUES(@CUR\_ID, @HOTEN, @NGAYSINH, @GIOITINH, @SODT, @EMAIL,@TENTK,@MATKHAU,@ANH,@VT)  END  GO |

#### Function tự tạo mã khách hàng cho bảng thành viên

|  |
| --- |
| --------------------LAY MOT KY TU BAT KY TRONG 1 CHUOI  CREATE FUNCTION [dbo].[fn\_PickRandomChar]  (  @chars VARCHAR(MAX)  )  RETURNS CHAR(1)  AS  BEGIN  DECLARE @result CHAR(1) = NULL;  DECLARE @resultIndex INT = NULL;  SET @resultIndex = [dbo].[fn\_RandIntBetween](1, LEN(@chars));  SET @result = SUBSTRING(@chars, @resultIndex, 1);  RETURN @result;  END  GO  ----- TAO MAKH CHO THANH VIEN  ------------------------THAY THE RAND() TRONG USER-FUNCTION  create view vw\_rand  as  SELECT RAND() AS Value  go  --------------LAY 1 SO TRONG KHOANG TU MIN-MAX  CREATE FUNCTION fn\_RandIntBetween  (  @lower INT,  @upper INT  )  RETURNS INT  AS  BEGIN  DECLARE @result INT  DECLARE @RAND FLOAT = (SELECT \* FROM vw\_rand)  DECLARE @range INT = @upper - @lower + 1;  SET @result = FLOOR(@rand \* @range + @lower);  RETURN @result;  END  go  --------------------TAO MA\_KH VOI SO LUONG KY TU, GIOI HAN CAC KY TY, LAY SO LUONG KY TU DUOI TRONG ID\_TV  CREATE FUNCTION FN\_GENERATE\_MATV(  @LENGHT INT,  @ID\_TV NVARCHAR(12),  @CHARACTER NVARCHAR(Max),  @SUB\_OF\_ID INT  )  RETURNS NVARCHAR(12) AS  BEGIN  DECLARE @MATV NVARCHAR(12) = ''  DECLARE @ID\_TV\_SUB nchar(3) = right(@ID\_TV,@SUB\_OF\_ID)  DECLARE @TIMES INT = @LENGHT - @SUB\_OF\_ID  DECLARE @CHAR NCHAR(1) = ''  WHILE @TIMES > 0  BEGIN  SET @CHAR = (SELECT dbo.fn\_PickRandomChar(@CHARACTER))  SET @MATV = @MATV + CAST(@CHAR AS nvarchar(1))  SET @TIMES = @TIMES -1  END  SET @MATV = @MATV + @ID\_TV\_SUB  RETURN @MATV + @ID\_TV\_SUB  END  GO  -----------------------HAM GOI LAY MA\_KH  create function fn\_MaKH  (@ID\_TV nvarchar(16))  returns nvarchar(12) as  begin  DECLARE @MA\_LENGHT INT = 12;  DECLARE @ID\_TV\_SUB nchar(3) = right(@ID\_TV,3)  DECLARE @CHARACTER NVARCHAR(Max) = 'ABCDEFGHJKLIMNOPQRSTUWXYZ0123456789'  DECLARE @MAKH NVARCHAR(12) = ''  DECLARE @TIMES INT = @MA\_LENGHT - LEN(@ID\_TV\_SUB)  DECLARE @CHAR NCHAR(1) = ''  DECLARE @MaDaRand nvarchar(50) = (select Value from vw\_rand)  DECLARE @isin nvarchar(50) = (select count(\*) from THANHVIEN where MAKH like @MaDaRand)    SET @MAKH = (SELECT DBO.FN\_GENERATE\_MATV(@MA\_LENGHT, @ID\_TV, @CHARACTER,3))  -- D130  --SELECT COUNT(\*) FROM THANHVIEN WHERE MA\_KH = @MATV  --IF TRUNG SET @MAKH LAI  while @isin != 0  begin  SET @MAKH = (SELECT DBO.FN\_GENERATE\_MATV(@MA\_LENGHT, @ID\_TV, @CHARACTER,3))  set @isin = (select count(MAKH) from THANHVIEN where MAKH like @MaDaRand)  end  RETURN @MAKH  end  go |

#### Trigger tự tạo khóa chính Thành viên

|  |
| --- |
| if(OBJECT\_ID('tg\_tv')) is not null  drop trigger tg\_tv  go  CREATE TRIGGER tg\_tv ON THANHVIEN  INSTEAD OF INSERT  AS BEGIN  SET NOCOUNT ON  DECLARE @HOTEN NVARCHAR(50) = (SELECT HOTEN FROM inserted)  DECLARE @NGAYSINH DATE =(SELECT NGAYSINH FROM inserted)  DECLARE @GIOITINH BIT = (SELECT GIOITINH FROM inserted)  DECLARE @SODT NVARCHAR(15) = (SELECT SODT FROM inserted)  DECLARE @EMAIL NVARCHAR(50) = (SELECT EMAIL FROM inserted)  DECLARE @MAKH NVARCHAR(50)    --SO ID GAN NHAT DUOC THEM VAO TRONG BANG  DECLARE @PRE\_ID NVARCHAR(16) = (SELECT TOP 1 CAST(REPLACE(ID\_TV,left(ID\_TV,2),'') AS INT) IDN FROM THANHVIEN ORDER BY IDN DESC)  DECLARE @NEXT INT  -- NEU LA RECORD DAU TIEN SO LA 1 HOAC @PRE\_ID + 1  IF(@PRE\_ID IS NULL) SET @NEXT = 1  ELSE SET @NEXT = @PRE\_ID + 1  --TAO ID TU TANG CHO HANG CHUAN BI THEM  DECLARE @CUR\_ID NVARCHAR(16) = ('TV' + RIGHT('00000'+CAST((@NEXT) AS NVARCHAR(5)),5))  SET @MAKH = (SELECT DBO.FN\_MAKH(@CUR\_ID))  --INSERT  INSERT INTO THANHVIEN VALUES(@CUR\_ID,@HOTEN,@NGAYSINH,@GIOITINH,@SODT,@EMAIL, @MAKH)  END  GO |

#### Trigger tự tạo khóa chính Phòng chiếu

|  |
| --- |
| if(OBJECT\_ID('tg\_pc')) is not null  drop trigger tg\_pc  go  create trigger tg\_pc on PHONGCHIEU  INSTEAD OF INSERT  AS BEGIN  SET NOCOUNT ON  DECLARE @DAY INT = (SELECT SOLUONGDAY FROM inserted)  DECLARE @COT INT = (SELECT SOLUONGCOT FROM inserted)  DECLARE @PRE\_ID NVARCHAR(16) = (SELECT TOP 1 CAST(REPLACE(ID\_PHONG,LEFT(ID\_PHONG,1),'') AS INT) ID FROM PHONGCHIEU ORDER BY ID DESC)  -- NEU LA RECORD DAU TIEN SO LA 1 HOAC @PRE\_ID + 1  DECLARE @NEXT NVARCHAR(16)  IF(@PRE\_ID IS NULL) SET @NEXT = 1  ELSE SET @NEXT = @PRE\_ID + 1    --TAO ID TU TANG CHO HANG CHUAN BI THEM  DECLARE @CUR\_ID NVARCHAR(16) = ('P' + RIGHT('000'+CAST((@NEXT) AS NVARCHAR(5)),3))  --INSERT  INSERT INTO PHONGCHIEU VALUES(@CUR\_ID,@DAY,@COT)  END  GO |

#### Trigger tự tạo khóa chính Phim

|  |
| --- |
| if(OBJECT\_ID('tg\_phim')) is not null  drop trigger tg\_phim  go  -- create trigger before insert phim  create trigger tg\_phim on PHIM  INSTEAD OF INSERT  AS BEGIN  SET NOCOUNT ON;  DECLARE @TEN NVARCHAR(50) = (SELECT TEN FROM INSERTED)  DECLARE @THOILUONG SMALLINT = (SELECT THOILUONG FROM INSERTED)  DECLARE @THELOAI NVARCHAR(50) = (SELECT THELOAI FROM INSERTED)  DECLARE @NGONNGU NVARCHAR(20) = (SELECT NGONNGU FROM INSERTED)  DECLARE @NGAY\_KC DATE = (SELECT NGAY\_KC FROM INSERTED)  DECLARE @TOMTAT NVARCHAR(1024) = (SELECT TOMTAT FROM INSERTED)  DECLARE @TRANGTHAI BIT = (SELECT TRANGTHAI FROM INSERTED)  DECLARE @ID\_NV NVARCHAR(10) = (SELECT ID\_NV FROM INSERTED)  DECLARE @POSTER NVARCHAR(100) = (SELECT POSTER FROM inserted)  --SO ID GAN NHAT DUOC THEM VAO TRONG BANG  DECLARE @PRE\_ID NVARCHAR(16) = (SELECT TOP 1 CAST(REPLACE(ID\_PHIM,left(ID\_PHIM,2),'') AS INT) IDN FROM PHIM ORDER BY IDN desc)  DECLARE @NEXT INT  -- NEU LA RECORD DAU TIEN SO LA 1 HOAC @PRE\_ID + 1  IF(@PRE\_ID IS NULL) SET @NEXT = 1  ELSE SET @NEXT = @PRE\_ID + 1  -- NEXT ID  DECLARE @CUR\_ID NVARCHAR(16) = ('FI' + RIGHT('0000'+CONVERT(NVARCHAR(10),@NEXT),5))  -- INSERT  INSERT INTO PHIM VALUES(@CUR\_ID, @TEN, @POSTER ,@THOILUONG, @THELOAI, @NGONNGU, @NGAY\_KC, @TOMTAT, @TRANGTHAI, @ID\_NV)  END  GO |

#### Trigger tự tạo khóa chính Suất chiếu

|  |
| --- |
| if(OBJECT\_ID('tg\_sc')) is not null  drop trigger tg\_sc  go  CREATE TRIGGER tg\_sc ON SUATCHIEU  INSTEAD OF INSERT  AS BEGIN  SET NOCOUNT ON  DECLARE @NGAYCHIEU DATE = (SELECT NGAYCHIEU FROM inserted)  DECLARE @ID\_PHONG NCHAR(4) = (SELECT ID\_PHONG FROM inserted)  DECLARE @ID\_PHIM NVARCHAR(15) = (SELECT ID\_PHIM FROM inserted)  DECLARE @GIOBATDAU TIME = (SELECT GIOBATDAU FROM inserted)  DECLARE @GIOKETTHUC TIME = (SELECT GIOKETTHUC FROM inserted)  DECLARE @ID\_NV NVARCHAR(10) = (SELECT ID\_NV FROM inserted)  --SO ID GAN NHAT DUOC THEM VAO TRONG BANG  DECLARE @PRE\_ID NVARCHAR(16) = (SELECT TOP 1 CAST(REPLACE(ID\_SUAT,left(ID\_SUAT,2),'') AS INT) IDN FROM SUATCHIEU ORDER BY IDN DESC)  DECLARE @NEXT INT  -- NEU LA RECORD DAU TIEN SO LA 1 HOAC @PRE\_ID + 1  IF(@PRE\_ID IS NULL) SET @NEXT = 1  ELSE SET @NEXT = @PRE\_ID + 1  --TAO ID TU TANG CHO HANG CHUAN BI THEM  DECLARE @CUR\_ID NVARCHAR(16) = ('SC' + RIGHT('00000'+CAST((@NEXT) AS NVARCHAR(5)),5))  --INSERT  INSERT INTO SUATCHIEU VALUES(@CUR\_ID,@NGAYCHIEU,@ID\_PHONG,@ID\_PHIM,@GIOBATDAU,@GIOKETTHUC,@ID\_NV)  END  GO |

#### Trigger tự tạo khóa chính Đồ ăn

|  |
| --- |
| if(OBJECT\_ID('tg\_Doan')) is not null  drop trigger tg\_Doan  go  -- create trigger before insert  create trigger tg\_doan on DOAN  INSTEAD OF INSERT  AS BEGIN  SET NOCOUNT ON;  DECLARE @LDA NVARCHAR(50) = (SELECT ID\_LOAI FROM INSERTED)  DECLARE @TENDA NVARCHAR(100) = (SELECT TEN FROM INSERTED)  --SO ID GAN NHAT DUOC THEM VAO TRONG BANG  DECLARE @PRE\_IDDA NVARCHAR(16) = (SELECT TOP 1 CAST(REPLACE(ID\_DOAN,left(ID\_DOAN,2),'') AS INT) IDoan FROM DOAN WHERE ID\_LOAI LIKE @LDA ORDER BY IDoan desc)  DECLARE @NEXT INT  -- NEU LA RECORD DAU TIEN SO LA 1 , NEU KHONG THI ++  IF(@PRE\_IDDA IS NULL) SET @NEXT = 1  ELSE SET @NEXT = @PRE\_IDDA + 1    -- NEXT ID  DECLARE @DA\_ID NVARCHAR(16) = @LDA + RIGHT('0000'+CONVERT(NVARCHAR(10),@NEXT),5)    -- INSERT  INSERT INTO DOAN VALUES(@DA\_ID, @TENDA, @LDA)  END  GO |

#### Trigger tự tạo khóa chính Đơn đồ ăn

|  |
| --- |
| if(OBJECT\_ID('tg\_DonDoan')) is not null  drop trigger tg\_DonDoan  go  -- create trigger before insert  create trigger tg\_DonDoan on DONDOAN  INSTEAD OF INSERT  AS BEGIN  SET NOCOUNT ON;  DECLARE @ID\_DONDA NVARCHAR(50) = (SELECT ID\_DONDA FROM INSERTED)  --SO ID GAN NHAT DUOC THEM VAO TRONG BANG  DECLARE @PRE\_IDDA NVARCHAR(16) = (SELECT TOP 1 CAST(REPLACE(ID\_DONDA,left(ID\_DONDA,2),'') AS INT) IDDoan FROM DONDOAN ORDER BY IDDoan desc)  DECLARE @NEXT INT  -- NEU LA RECORD DAU TIEN SO LA 1 , NEU KHONG THI ++  IF(@PRE\_IDDA IS NULL) SET @NEXT = 1  ELSE SET @NEXT = @PRE\_IDDA + 1    -- NEXT ID  DECLARE @DA\_ID NVARCHAR(50) = 'DA'+ RIGHT('0000'+CONVERT(NVARCHAR(10),@NEXT),5)    -- INSERT  INSERT INTO DONDOAN VALUES(@DA\_ID)  END  GO |

#### Trigger tự tạo khóa chính Ghế

|  |
| --- |
| if(OBJECT\_ID('tg\_donve')) is not null  drop trigger tg\_donve  go  CREATE TRIGGER tg\_donve ON DONVE  INSTEAD OF INSERT  AS BEGIN  SET NOCOUNT ON  --SO ID GAN NHAT DUOC THEM VAO TRONG BANG  DECLARE @PRE\_ID NVARCHAR(16) = (SELECT TOP 1 CAST(REPLACE(ID\_DONVE,left(ID\_DONVE,2),'') AS INT) IDN FROM DONVE ORDER BY IDN DESC)  DECLARE @NEXT INT  -- NEU LA RECORD DAU TIEN SO LA 1 HOAC @PRE\_ID + 1  IF(@PRE\_ID IS NULL) SET @NEXT = 1  ELSE SET @NEXT = @PRE\_ID + 1  --TAO ID TU TANG CHO HANG CHUAN BI THEM  DECLARE @CUR\_ID NVARCHAR(16) = ('DV' + RIGHT('00000'+CAST((@NEXT) AS NVARCHAR(5)),5))  --INSERT  INSERT INTO DONVE VALUES(@CUR\_ID)  END  GO |

#### Thủ tục Quản lý đồ ăn

|  |
| --- |
| if OBJECT\_ID('sp\_qldoan') is not null  drop proc sp\_qldoan  go  create proc sp\_qldoan  as  begin  select DOAN.ID\_DOAN, DOAN.TEN, DOAN.ID\_LOAI, KICHCODOAN.ID\_KICHCO, KICHCODOAN.GIA  from KICHCODOAN join DOAN on KICHCODOAN.ID\_DOAN = DOAN.ID\_DOAN  end  go |

#### Thủ tục thống kê doanh thu đồ ăn

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthudoan') is not null  drop proc sp\_doanhthudoan  go  create proc sp\_doanhthudoan  as  begin  select DOAN.TEN, SUM(DOANCT.SOLUONG) AS SOLUONG, KICHCODOAN.ID\_KICHCO, (SUM(DOANCT.SOLUONG) \* KICHCODOAN.GIA) AS THANHTIEN  from KICHCODOAN join DOAN on KICHCODOAN.ID\_DOAN = DOAN.ID\_DOAN  join DOANCT on DOANCT.ID\_KCDA = KICHCODOAN.ID\_KCDA  GROUP BY DOAN.TEN,KICHCODOAN.ID\_KICHCO, KICHCODOAN.GIA  end  go |

#### Thủ tục Sắp xếp số lượng đồ ăn

|  |
| --- |
| if OBJECT\_ID('sp\_sapxepdoan') is not null  drop proc sp\_sapxepdoan  go  create proc sp\_sapxepdoan  as  begin  select DOAN.TEN, SUM(DOANCT.SOLUONG) AS SOLUONG, KICHCODOAN.ID\_KICHCO, (SUM(DOANCT.SOLUONG) \* KICHCODOAN.GIA) AS THANHTIEN  from KICHCODOAN join DOAN on KICHCODOAN.ID\_DOAN = DOAN.ID\_DOAN  join DOANCT on DOANCT.ID\_KCDA = KICHCODOAN.ID\_KCDA  GROUP BY DOAN.TEN,KICHCODOAN.ID\_KICHCO, KICHCODOAN.GIA  order by SOLUONG desc  end  go |

#### Thủ tục Lọc doanh thu đồ ăn theo ngày

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthudoan\_theongay') is not null  drop proc sp\_doanhthudoan\_theongay  go  create proc sp\_doanhthudoan\_theongay (@Ngay date)  as  begin  select DOAN.TEN, SUM(DOANCT.SOLUONG) AS SOLUONG, KICHCODOAN.ID\_KICHCO, (SUM(DOANCT.SOLUONG) \* KICHCODOAN.GIA) AS THANHTIEN  from KICHCODOAN join DOAN on KICHCODOAN.ID\_DOAN = DOAN.ID\_DOAN  join DOANCT on DOANCT.ID\_KCDA = KICHCODOAN.ID\_KCDA  join DONTHANHTOAN on DONTHANHTOAN.ID\_DONDA = DOANCT.ID\_DONDA  where NGAYDAT BETWEEN (convert(varchar, @Ngay) +' 00:00:00') AND (convert(varchar, @Ngay) + ' 23:59:59')  GROUP BY DOAN.TEN,KICHCODOAN.ID\_KICHCO, KICHCODOAN.GIA  end  go |

#### Thủ tục Lọc doanh thu đồ ăn theo tháng – năm

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthudoan\_theothangnam') is not null  drop proc sp\_doanhthudoan\_theothangnam  go  create proc sp\_doanhthudoan\_theothangnam (@Thang int, @Year int)  as  begin  select DOAN.TEN, SUM(DOANCT.SOLUONG) AS SOLUONG, KICHCODOAN.ID\_KICHCO, (SUM(DOANCT.SOLUONG) \* KICHCODOAN.GIA) AS THANHTIEN  from KICHCODOAN join DOAN on KICHCODOAN.ID\_DOAN = DOAN.ID\_DOAN  join DOANCT on DOANCT.ID\_KCDA = KICHCODOAN.ID\_KCDA  join DONTHANHTOAN on DONTHANHTOAN.ID\_DONDA = DOANCT.ID\_DONDA  WHERE MONTH(NGAYDAT) = @Thang and YEAR(NGAYDAT) = @Year  GROUP BY DOAN.TEN,KICHCODOAN.ID\_KICHCO, KICHCODOAN.GIA  end  go |

#### Thủ tục Lọc doanh thu đồ ăn theo năm

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthudoan\_theonam') is not null  drop proc sp\_doanhthudoan\_theonam  go  create proc sp\_doanhthudoan\_theonam (@Year int)  as  begin  select DOAN.TEN, SUM(DOANCT.SOLUONG) AS SOLUONG, KICHCODOAN.ID\_KICHCO, (SUM(DOANCT.SOLUONG) \* KICHCODOAN.GIA) AS THANHTIEN  from KICHCODOAN join DOAN on KICHCODOAN.ID\_DOAN = DOAN.ID\_DOAN  join DOANCT on DOANCT.ID\_KCDA = KICHCODOAN.ID\_KCDA  join DONTHANHTOAN on DONTHANHTOAN.ID\_DONDA = DOANCT.ID\_DONDA  WHERE YEAR(NGAYDAT) = @Year  GROUP BY DOAN.TEN,KICHCODOAN.ID\_KICHCO, KICHCODOAN.GIA  end  go |

#### Thủ tục Lọc doanh thu đồ ăn theo tên

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthudoan\_theoten') is not null  drop proc sp\_doanhthudoan\_theoten  go  create proc sp\_doanhthudoan\_theoten (@ten nvarchar(100))  as  begin  select DOAN.TEN, SUM(DOANCT.SOLUONG) AS SOLUONG, KICHCODOAN.ID\_KICHCO, (SUM(DOANCT.SOLUONG) \* KICHCODOAN.GIA) AS THANHTIEN  from KICHCODOAN join DOAN on KICHCODOAN.ID\_DOAN = DOAN.ID\_DOAN  join DOANCT on DOANCT.ID\_KCDA = KICHCODOAN.ID\_KCDA  join DONTHANHTOAN on DONTHANHTOAN.ID\_DONDA = DOANCT.ID\_DONDA  WHERE DOAN.TEN like @ten  GROUP BY DOAN.TEN,KICHCODOAN.ID\_KICHCO, KICHCODOAN.GIA  end  go |

#### Thủ tục Lọc doanh thu đồ ăn theo size

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthudoan\_theosize') is not null  drop proc sp\_doanhthudoan\_theosize  go  create proc sp\_doanhthudoan\_theosize (@size nchar(1))  as  begin  select DOAN.TEN, SUM(DOANCT.SOLUONG) AS SOLUONG, KICHCODOAN.ID\_KICHCO, (SUM(DOANCT.SOLUONG) \* KICHCODOAN.GIA) AS THANHTIEN  from KICHCODOAN join DOAN on KICHCODOAN.ID\_DOAN = DOAN.ID\_DOAN  join DOANCT on DOANCT.ID\_KCDA = KICHCODOAN.ID\_KCDA  join DONTHANHTOAN on DONTHANHTOAN.ID\_DONDA = DOANCT.ID\_DONDA  WHERE KICHCODOAN.ID\_KICHCO like @size  GROUP BY DOAN.TEN,KICHCODOAN.ID\_KICHCO, KICHCODOAN.GIA  end  go |

#### Thủ tục Quản lý suất chiếu

|  |
| --- |
| if OBJECT\_ID('sp\_qlsuatchieu') is not null  drop proc sp\_qlsuatchieu  go  create proc sp\_qlsuatchieu  as  begin  SELECT PHONGCHIEU.ID\_PHONG, PHIM.TEN, CONCAT(convert(varchar, GIOBATDAU, 0), ' ', NGAYCHIEU) as NgayGio  FROM PHONGCHIEU JOIN SUATCHIEU ON PHONGCHIEU.ID\_PHONG = SUATCHIEU.ID\_PHONG  JOIN PHIM ON PHIM.ID\_PHIM = SUATCHIEU.ID\_PHIM  END  GO |

#### Thủ tục Thống kê doanh thu phòng vé

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthuphongve') is not null  drop proc sp\_doanhthuphongve  go  create proc sp\_doanhthuphongve  as  begin  declare @SoSuatChieu\_Thang table (thang int, tongsosuatchieu int)  insert into @SoSuatChieu\_Thang  select DATEPART(MM, NGAYCHIEU) ,count(ID\_SUAT) from SUATCHIEU group by DATEPART(MM, NGAYCHIEU)  SELECT DATEPART(MM, NGAYCHIEU) as thang, tongsosuatchieu, COUNT(ID\_VEDAT) as sovedaban, sum(GIA + PHUTHU) as doanhthu  from SUATCHIEU SC left join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT  join @SoSuatChieu\_Thang SSC on SSC.thang = MONTH(SC.NGAYCHIEU)  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  group by DATEPART(MM, NGAYCHIEU), tongsosuatchieu  end  go |

#### Thủ tục Sắp xếp doanh thu phòng vé giảm dần theo số lượng vé đã bán

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthuphongve\_sapxep') is not null  drop proc sp\_doanhthuphongve\_sapxep  go  create proc sp\_doanhthuphongve\_sapxep  as  begin  declare @SoSuatChieu\_Thang table (thang int, tongsosuatchieu int)  insert into @SoSuatChieu\_Thang  select DATEPART(MM, NGAYCHIEU) ,count(ID\_SUAT) from SUATCHIEU group by DATEPART(MM, NGAYCHIEU)  SELECT DATEPART(MM, NGAYCHIEU) as thang, tongsosuatchieu, COUNT(ID\_VEDAT) as sovedaban, sum(GIA + PHUTHU) as doanhthu  from SUATCHIEU SC left join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT  join @SoSuatChieu\_Thang SSC on SSC.thang = MONTH(SC.NGAYCHIEU)  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  group by DATEPART(MM, NGAYCHIEU), tongsosuatchieu  order by sovedaban desc  end  go |

#### Thủ tục Lọc doanh thu phòng vé theo ngày

|  |
| --- |
| -- THEO NGÀY  if OBJECT\_ID('sp\_doanhthuphongve\_theongay') is not null  drop proc sp\_doanhthuphongve\_theongay  go  create proc sp\_doanhthuphongve\_theongay (@Ngay date)  as  begin  declare @SoSuatChieu\_Thang table (thang int, tongsosuatchieu int)  insert into @SoSuatChieu\_Thang  select DATEPART(MM, NGAYCHIEU) ,count(ID\_SUAT) from SUATCHIEU group by DATEPART(MM, NGAYCHIEU)  SELECT DATEPART(MM, NGAYCHIEU) as thang, tongsosuatchieu, COUNT(ID\_VEDAT) as sovedaban, sum(GIA + PHUTHU) as doanhthu  from SUATCHIEU SC left join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT  join @SoSuatChieu\_Thang SSC on SSC.thang = MONTH(SC.NGAYCHIEU)  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  where NGAYCHIEU BETWEEN (convert(varchar, @Ngay) +' 00:00:00') AND (convert(varchar, @Ngay) + ' 23:59:59')  group by DATEPART(MM, NGAYCHIEU), tongsosuatchieu  end  go |

#### Thủ tục Lọc doanh thu phòng vé theo tháng – năm

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthuphongve\_theothangnam') is not null  drop proc sp\_doanhthuphongve\_theothangnam  go  create proc sp\_doanhthuphongve\_theothangnam (@Thang int, @Year int)  as  begin  declare @SoSuatChieu\_Thang table (thang int, tongsosuatchieu int)  insert into @SoSuatChieu\_Thang  select DATEPART(MM, NGAYCHIEU) ,count(ID\_SUAT) from SUATCHIEU group by DATEPART(MM, NGAYCHIEU)  SELECT DATEPART(MM, NGAYCHIEU) as thang, tongsosuatchieu, COUNT(ID\_VEDAT) as sovedaban, sum(GIA + PHUTHU) as doanhthu  from SUATCHIEU SC left join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT  join @SoSuatChieu\_Thang SSC on SSC.thang = MONTH(SC.NGAYCHIEU)  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  where MONTH(NGAYCHIEU) like @Thang and YEAR(NGAYCHIEU) like @Year  group by DATEPART(MM, NGAYCHIEU), tongsosuatchieu  end  go |

#### Lọc doanh thu phòng vé theo năm

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthuphongve\_theonam') is not null  drop proc sp\_doanhthuphongve\_theonam  go  create proc sp\_doanhthuphongve\_theonam (@Year int)  as  begin  declare @SoSuatChieu\_Thang table (thang int, tongsosuatchieu int)  insert into @SoSuatChieu\_Thang  select DATEPART(MM, NGAYCHIEU) ,count(ID\_SUAT) from SUATCHIEU group by DATEPART(MM, NGAYCHIEU)  SELECT DATEPART(MM, NGAYCHIEU) as thang, tongsosuatchieu, COUNT(ID\_VEDAT) as sovedaban, sum(GIA + PHUTHU) as doanhthu  from SUATCHIEU SC left join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT  join @SoSuatChieu\_Thang SSC on SSC.thang = MONTH(SC.NGAYCHIEU)  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  where YEAR(NGAYCHIEU) like @Year  group by DATEPART(MM, NGAYCHIEU), tongsosuatchieu  end  go |

#### Thủ tục Thống kê lượng vé đã bán

|  |
| --- |
| if OBJECT\_ID('sp\_luongvedaban') is not null  drop proc sp\_luongvedaban  go  create proc sp\_luongvedaban  as  begin  declare @SoSuatChieu\_Ve table(ngaychieu date, tongsuatchieu int)  insert into @SoSuatChieu\_Ve  select NGAYCHIEU ,count(ID\_SUAT) from SUATCHIEU group by NGAYCHIEU  select SC.NGAYCHIEU, COUNT(ID\_VEDAT) as sovedaban, tongsuatchieu from SUATCHIEU SC  left join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT join @SoSuatChieu\_Ve SSC on SSC.ngaychieu = SC.NGAYCHIEU  group by SC.NGAYCHIEU, tongsuatchieu  end  go |

#### Thủ tục Sắp xếp lượng vé đã bán giảm dần theo số vé đã bán

|  |
| --- |
| if OBJECT\_ID('sp\_luongvedaban\_sapxep') is not null  drop proc sp\_luongvedaban\_sapxep  go  create proc sp\_luongvedaban\_sapxep  as  begin  declare @SoSuatChieu\_Ve table(ngaychieu date, tongsuatchieu int)  insert into @SoSuatChieu\_Ve  select NGAYCHIEU ,count(ID\_SUAT) from SUATCHIEU group by NGAYCHIEU  select SC.NGAYCHIEU, COUNT(ID\_VEDAT) as sovedaban, tongsuatchieu from SUATCHIEU SC  left join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT join @SoSuatChieu\_Ve SSC on SSC.ngaychieu = SC.NGAYCHIEU  group by SC.NGAYCHIEU, tongsuatchieu  order by sovedaban desc  end  go |

#### Thủ tục Lọc lượng vé đã bán theo ngày

|  |
| --- |
| if OBJECT\_ID('sp\_luongvedaban\_theongay') is not null  drop proc sp\_luongvedaban\_theongay  go  create proc sp\_luongvedaban\_theongay (@ngay varchar(20))  as  begin  declare @SoSuatChieu\_Ve table(ngaychieu date, tongsuatchieu int)  insert into @SoSuatChieu\_Ve  select NGAYCHIEU ,count(ID\_SUAT) from SUATCHIEU group by NGAYCHIEU  select SC.NGAYCHIEU, COUNT(ID\_VEDAT) as sovedaban, tongsuatchieu from SUATCHIEU SC  left join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT  join @SoSuatChieu\_Ve SSC on SSC.ngaychieu = SC.NGAYCHIEU  where @ngay like SC.NGAYCHIEU  group by SC.NGAYCHIEU, tongsuatchieu  end  go |

#### Thủ tục Lọc lượng vé đã bán theo tháng – năm

|  |
| --- |
| if OBJECT\_ID('sp\_luongvedaban\_theothangnam') is not null  drop proc sp\_luongvedaban\_theothangnam  go  create proc sp\_luongvedaban\_theothangnam (@Thang int, @Year int)  as  begin  declare @SoSuatChieu\_Ve table(ngaychieu date, tongsuatchieu int)  insert into @SoSuatChieu\_Ve  select NGAYCHIEU ,count(ID\_SUAT) from SUATCHIEU group by NGAYCHIEU  select SC.NGAYCHIEU, COUNT(ID\_VEDAT) as sovedaban, tongsuatchieu  from SUATCHIEU SC left join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT  join @SoSuatChieu\_Ve SSC on SSC.ngaychieu = SC.NGAYCHIEU  where MONTH(SC.NGAYCHIEU) like @Thang and YEAR(sc.NGAYCHIEU) like @Year  group by SC.NGAYCHIEU, tongsuatchieu  end  go |

#### Thủ tục Lọc lượng vé đã bán theo năm

|  |
| --- |
| if OBJECT\_ID('sp\_luongvedaban\_theonam') is not null  drop proc sp\_luongvedaban\_theonam  go  create proc sp\_luongvedaban\_theonam (@Year int)  as  begin  declare @SoSuatChieu\_Ve table(ngaychieu date, tongsuatchieu int)  insert into @SoSuatChieu\_Ve  select NGAYCHIEU ,count(ID\_SUAT) from SUATCHIEU group by NGAYCHIEU  select SC.NGAYCHIEU, COUNT(ID\_VEDAT) as sovedaban, tongsuatchieu  from SUATCHIEU SC left join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT  join @SoSuatChieu\_Ve SSC on SSC.ngaychieu = SC.NGAYCHIEU  where YEAR(sc.NGAYCHIEU) like @Year  group by SC.NGAYCHIEU, tongsuatchieu  end  go |

#### Thủ tục Thống kê doanh thu phim

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthuphim') is not null  drop proc sp\_doanhthuphim  go  create proc sp\_doanhthuphim  as  begin  declare @SoSuatChieu\_Phim table(id nvarchar(20), sosuatchieu int)  insert into @SoSuatChieu\_Phim  select PHIM.ID\_PHIM, count(ID\_SUAT) from SUATCHIEU join PHIM  on SUATCHIEU.ID\_PHIM = PHIM.ID\_PHIM group by PHIM.ID\_PHIM  select P.TEN, sosuatchieu, COUNT(VD.ID\_VEDAT) as sovedat, SUM(GIA + PHUTHU) as tonggia  from PHIM P join SUATCHIEU SC on P.ID\_PHIM = SC.ID\_PHIM  left join VEDAT VD on VD.ID\_SUAT = SC.ID\_SUAT  join @SoSuatChieu\_Phim SSC on SSC.id = P.ID\_PHIM  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  group by P.TEN, sosuatchieu  end  go |

#### Thủ tục Sắp xếp doanh thu phim giảm dần theo lượng vé đã bán

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthuphim\_sapxep') is not null  drop proc sp\_doanhthuphim\_sapxep  go  create proc sp\_doanhthuphim\_sapxep  as  begin  declare @SoSuatChieu\_Phim table(id nvarchar(20), sosuatchieu int)  insert into @SoSuatChieu\_Phim  select PHIM.ID\_PHIM, count(ID\_SUAT) from SUATCHIEU join PHIM  on SUATCHIEU.ID\_PHIM = PHIM.ID\_PHIM group by PHIM.ID\_PHIM  select P.TEN, sosuatchieu, COUNT(VD.ID\_VEDAT) as sovedat, SUM(GIA + PHUTHU) as tonggia  from PHIM P join SUATCHIEU SC on P.ID\_PHIM = SC.ID\_PHIM  left join VEDAT VD on VD.ID\_SUAT = SC.ID\_SUAT  join @SoSuatChieu\_Phim SSC on SSC.id = P.ID\_PHIM  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  group by P.TEN, sosuatchieu  order by sovedat desc  end  go |

#### Thủ tục lấy Top 5 doanh thu phim cao nhất

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthuphim\_top') is not null  drop proc sp\_doanhthuphim\_top  go  create proc sp\_doanhthuphim\_top  as  begin  declare @SoSuatChieu\_Phim table(id nvarchar(20), sosuatchieu int)  insert into @SoSuatChieu\_Phim  select PHIM.ID\_PHIM, count(ID\_SUAT) from SUATCHIEU join PHIM  on SUATCHIEU.ID\_PHIM = PHIM.ID\_PHIM group by PHIM.ID\_PHIM  select top (5) P.TEN, sosuatchieu, COUNT(VD.ID\_VEDAT) as sovedat, SUM(GIA + PHUTHU) as tonggia  from PHIM P join SUATCHIEU SC on P.ID\_PHIM = SC.ID\_PHIM  left join VEDAT VD on VD.ID\_SUAT = SC.ID\_SUAT  join @SoSuatChieu\_Phim SSC on SSC.id = P.ID\_PHIM  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  group by P.TEN, sosuatchieu  order by tonggia desc  end  go |

#### Thủ tục Lọc doanh thu phim theo ngày

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthuphim\_theongay') is not null  drop proc sp\_doanhthuphim\_theongay  go  create proc sp\_doanhthuphim\_theongay (@Ngay date)  as  begin  declare @SoSuatChieu\_Phim table(id nvarchar(20), sosuatchieu int)  insert into @SoSuatChieu\_Phim  select PHIM.ID\_PHIM, count(ID\_SUAT) from SUATCHIEU join PHIM  on SUATCHIEU.ID\_PHIM = PHIM.ID\_PHIM group by PHIM.ID\_PHIM  select P.TEN, sosuatchieu, COUNT(VD.ID\_VEDAT) as sovedat, SUM(GIA + PHUTHU) as tonggia  from PHIM P join SUATCHIEU SC on P.ID\_PHIM = SC.ID\_PHIM  left join VEDAT VD on VD.ID\_SUAT = SC.ID\_SUAT  join @SoSuatChieu\_Phim SSC on SSC.id = P.ID\_PHIM  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  where sc.NGAYCHIEU BETWEEN (convert(varchar, @Ngay) +' 00:00:00') AND (convert(varchar, @Ngay) + ' 23:59:59')  group by P.TEN, sosuatchieu  end  go |

#### Thủ tục Lọc doanh thu phim theo tháng – năm

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthuphim\_theothangnam') is not null  drop proc sp\_doanhthuphim\_theothangnam  go  create proc sp\_doanhthuphim\_theothangnam (@Thang int, @Year int)  as  begin  declare @SoSuatChieu\_Phim table(id nvarchar(20), sosuatchieu int)  insert into @SoSuatChieu\_Phim  select PHIM.ID\_PHIM, count(ID\_SUAT) from SUATCHIEU join PHIM  on SUATCHIEU.ID\_PHIM = PHIM.ID\_PHIM group by PHIM.ID\_PHIM  select P.TEN, sosuatchieu, COUNT(VD.ID\_VEDAT) as sovedat, SUM(GIA + PHUTHU) as tonggia  from PHIM P join SUATCHIEU SC on P.ID\_PHIM = SC.ID\_PHIM  left join VEDAT VD on VD.ID\_SUAT = SC.ID\_SUAT  join @SoSuatChieu\_Phim SSC on SSC.id = P.ID\_PHIM  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  where MONTH(sc.NGAYCHIEU) like @Thang and YEAR(sc.NGAYCHIEU) like @Year  group by P.TEN, sosuatchieu  end  go |

#### Thủ tục Lọc doanh thu phim theo năm

|  |
| --- |
| if OBJECT\_ID('sp\_doanhthuphim\_theonam') is not null  drop proc sp\_doanhthuphim\_theonam  go  create proc sp\_doanhthuphim\_theonam (@Year int)  as  begin  declare @SoSuatChieu\_Phim table(id nvarchar(20), sosuatchieu int)  insert into @SoSuatChieu\_Phim  select PHIM.ID\_PHIM, count(ID\_SUAT) from SUATCHIEU join PHIM  on SUATCHIEU.ID\_PHIM = PHIM.ID\_PHIM group by PHIM.ID\_PHIM  select P.TEN, sosuatchieu, COUNT(VD.ID\_VEDAT) as sovedat, SUM(GIA + PHUTHU) as tonggia  from PHIM P join SUATCHIEU SC on P.ID\_PHIM = SC.ID\_PHIM  left join VEDAT VD on VD.ID\_SUAT = SC.ID\_SUAT  join @SoSuatChieu\_Phim SSC on SSC.id = P.ID\_PHIM  join LOAIVE on VD.ID\_LOAIVE = LOAIVE.ID\_LOAIVE  join GHE on VD.ID\_GHE = GHE.ID\_GHE  join LOAIGHE on GHE.ID\_LOAIGHE = LOAIGHE.ID\_LOAIGHE  where YEAR(sc.NGAYCHIEU) like @Year  group by P.TEN, sosuatchieu  end  go |

#### Thủ tục Tra cứu lịch chiếu

|  |
| --- |
| if OBJECT\_ID('sp\_tracuulichchieu') is not null  drop proc sp\_tracuulichchieu  go  create proc sp\_tracuulichchieu  as  begin  declare @stt table (stt int identity(1,1), idsuat nvarchar(50) )  insert into @stt  select ID\_SUAT from SUATCHIEU  select stt, sc.ID\_SUAT,p.TEN,sc.ID\_PHONG,sc.GIOBATDAU,p.TRANGTHAI,sc.NGAYCHIEU  from SUATCHIEU sc join PHIM p on sc.ID\_PHIM=p.ID\_PHIM join @stt s on s.idsuat = sc.ID\_SUAT  end  go |

#### Thủ tục Quản lý đơn thanh toán

|  |
| --- |
| CREATE PROC [dbo].[sp\_donthanhtoan] @NGAYBD DATE = NULL, @NGAYKT DATE = NULL, @THANG INT = NULL, @NAM INT = NULL, @ISKHACHHANG INT = -1, @KEYWORD NVARCHAR(100) = NULL  as  begin  DECLARE @TABLE TABLE(ID\_DONTT INT,NGAYDAT DATE, ID\_TV NVARCHAR(16), TONG DECIMAL(9,3));  insert @TABLE select ID\_DONTT, NGAYDAT, TV.ID\_TV, ROUND(TONG, 0)  from DONTHANHTOAN DTT LEFT JOIN THANHVIEN TV ON DTT.ID\_TV = TV.ID\_TV  WHERE ((@NGAYBD IS NULL AND @NGAYKT IS NULL) OR (NGAYDAT >= @NGAYBD AND NGAYDAT <= @NGAYKT) )  AND (@THANG IS NULL OR MONTH(NGAYDAT) like @THANG) AND (@NAM IS NULL OR YEAR(NGAYDAT) like @NAM)  AND (@KEYWORD IS NULL OR (SODT LIKE @KEYWORD OR HOTEN LIKE @KEYWORD))  ORDER BY ID\_DONTT ASC  IF(@ISKHACHHANG != -1)  BEGIN  declare @now date = getdate();  if(@ISKHACHHANG = 0)  select \* from @TABLE where ((NGAYDAT < @now ) or (NGAYDAT = @now)) and ID\_TV IS NOT NULL order by NGAYDAT desc, ID\_DONTT desc  else select \* from @TABLE where (NGAYDAT > @now) or (NGAYDAT = @now) and ID\_TV IS NULL order by NGAYDAT desc, ID\_DONTT desc  END  else select \* from @TABLE  end  GO |

#### Thủ tục Lọc đơn thanh toán theo ngày mua

|  |
| --- |
| if OBJECT\_ID('sp\_donThanhToan\_NgayMua') is not null  drop proc sp\_donThanhToan\_NgayMua  go  create proc sp\_donThanhToan\_NgayMua (@ngayMua date)  as  begin  select ID\_DONTT, GIOBATDAU as SUATCHIEU, convert(varchar, NGAYDAT, 105) as NGAYDAT, THANHVIEN.ID\_TV, TONG  from DONTHANHTOAN JOIN DONVE ON DONTHANHTOAN.ID\_DONVE = DONVE.ID\_DONVE  JOIN THANHVIEN ON THANHVIEN.ID\_TV = DONTHANHTOAN.ID\_TV  JOIN VEDAT ON VEDAT.ID\_DONVE = DONVE.ID\_DONVE  JOIN SUATCHIEU ON SUATCHIEU.ID\_SUAT = VEDAT.ID\_SUAT  where NGAYDAT BETWEEN (convert(varchar, @ngayMua) +' 00:00:00') AND (convert(varchar, @ngayMua) + ' 23:59:59')  GROUP BY ID\_DONTT, SUATCHIEU.GIOBATDAU,SUATCHIEU.NGAYCHIEU, NGAYDAT, TONG, THANHVIEN.ID\_TV  end  go |

#### Thủ tục Lọc đơn thanh toán theo loại khách hàng

|  |
| --- |
| if OBJECT\_ID('sp\_donThanhToan\_KhachHang') is not null  drop proc sp\_donThanhToan\_KhachHang  go  create proc sp\_donThanhToan\_KhachHang(@loaiKH nvarchar(10))  as  begin  declare @loaiKH\_search varchar  if @loaiKH like N'Thành viên'  set @loaiKH\_search = '%TV%'  else  set @loaiKH\_search = 'NULL'  select ID\_DONTT, GIOBATDAU as SUATCHIEU, convert(varchar, NGAYDAT, 105) as NGAYDAT, THANHVIEN.ID\_TV, TONG  from DONTHANHTOAN JOIN DONVE ON DONTHANHTOAN.ID\_DONVE = DONVE.ID\_DONVE  JOIN THANHVIEN ON THANHVIEN.ID\_TV = DONTHANHTOAN.ID\_TV  JOIN VEDAT ON VEDAT.ID\_DONVE = DONVE.ID\_DONVE  JOIN SUATCHIEU ON SUATCHIEU.ID\_SUAT = VEDAT.ID\_SUAT  where THANHVIEN.ID\_TV like @loaiKH\_search  GROUP BY ID\_DONTT, SUATCHIEU.GIOBATDAU,SUATCHIEU.NGAYCHIEU, NGAYDAT, TONG, THANHVIEN.ID\_TV  end  go |

#### Thủ tục Đơn vé

|  |
| --- |
| if OBJECT\_ID('sp\_donVe') is not null  drop proc sp\_donVe  go  create proc sp\_donVe (@id\_dontt int)  as  begin  select P.TEN,  convert(varchar, GIOBATDAU, 0) + ' ' + convert(varchar, NGAYCHIEU, 105) as SUATCHIEU,  VDAT.ID\_LOAIVE,  convert(float,(GIA + PHUTHU)) as GIA  from DONTHANHTOAN DTT join DONVE DV on DTT.ID\_DONVE = DV.ID\_DONVE  join VEDAT VDAT on VDAT.ID\_DONVE = DV.ID\_DONVE  join SUATCHIEU SC on SC.ID\_SUAT = VDAT.ID\_SUAT  join PHIM P on P.ID\_PHIM = SC.ID\_PHIM  join LOAIVE LV on LV.ID\_LOAIVE = VDAT.ID\_LOAIVE  join GHE G on G.ID\_GHE = VDAT.ID\_GHE  join LOAIGHE LG on LG.ID\_LOAIGHE = G.ID\_LOAIGHE  where ID\_DONTT = @id\_dontt  group by ID\_VEDAT, P.TEN, GIOBATDAU, NGAYCHIEU, VDAT.ID\_LOAIVE, GIA, PHUTHU  end  go |

#### Thủ tục Đơn đồ ăn

|  |
| --- |
| if OBJECT\_ID('sp\_donDoAn') is not null  drop proc sp\_donDoAn  go  create proc sp\_donDoAn (@id\_dontt int)  as  begin  select TEN, SOLUONG, ID\_KICHCO, convert(float,sum(GIA)) as TONGTIEN  from DONTHANHTOAN DTT join DONDOAN DDA on DTT.ID\_DONDA = DDA.ID\_DONDA  join DOANCT DACT on DACT.ID\_DONDA = DDA.ID\_DONDA  join KICHCODOAN KCDA on KCDA.ID\_KCDA = DACT.ID\_KCDA  join DOAN DA on DA.ID\_DOAN = KCDA.ID\_DOAN  where ID\_DONTT = @id\_dontt  group by TEN, SOLUONG, ID\_KICHCO  end  go |

#### Thủ tục Quản lý đồ ăn kèm (Thêm đồ ăn)

|  |
| --- |
| if OBJECT\_ID('sp\_themDoAn') is not null  create proc [dbo].[sp\_themDoAn] (@tenDA nvarchar(50), @id\_loaiDA varchar(2), @id\_kichCo varchar(1), @gia float)  as  begin  if (select count(\*) from DOAN where TEN like @tenDA and ID\_LOAI like @id\_loaiDA) = 0  begin  insert into DOAN(TEN, ID\_LOAI)  values (@tenDA, @id\_loaiDA)  end  declare @id\_da varchar(10)  select @id\_da = (select ID\_DOAN from DOAN where TEN like @tenDA and ID\_LOAI like @id\_loaiDA)  insert into KICHCODOAN(ID\_DOAN, ID\_KICHCO, GIA)  values (@id\_da, @id\_kichCo, @gia)  end  GO |

#### Thủ tục Quản lý đồ ăn kèm (Sửa đồ ăn)

|  |
| --- |
| if OBJECT\_ID('sp\_suaDoAn') is not null  drop proc sp\_suaDoAn  go  create proc sp\_suaDoAn (@tenDA nvarchar(50), @id\_loaiDA varchar(2), @id\_kichCo varchar(1), @gia float, @id\_DoAn varchar(10), @id\_KCDA int)  as  begin  update DOAN  set TEN = @tenDA, ID\_LOAI = @id\_loaiDA  where ID\_DOAN like @id\_DoAn  update KICHCODOAN  set ID\_KICHCO = @id\_kichCo, GIA = @gia  where ID\_KCDA like @id\_KCDA  end  go |

#### Trigger Quản lý đồ ăn kèm (Xóa đồ ăn)

|  |
| --- |
| if OBJECT\_ID('tg\_xoaDoAn') is not null  drop trigger tg\_xoaDoAn  go  create trigger tg\_xoaDoAn on DOAN instead of delete  as  begin  delete from KICHCODOAN where ID\_DOAN like (select ID\_DOAN from deleted)  delete from DOAN where ID\_DOAN like (select ID\_DOAN from deleted)  end  go |

#### Hàm lấy Khung giờ

|  |
| --- |
| create function [dbo].[fn\_KhungGio](@giobd\_ngay time, @khoangcachgio int)  returns @bangkhunggio table (khunggio varchar(15))  as  begin  declare @giokt\_ngay time = '23:00'  declare @giotemp time = @giobd\_ngay  while CONVERT(time, @giotemp) < CONVERT(time, @giokt\_ngay)  begin  set @giotemp = DATEADD(HOUR, @khoangcachgio, CONVERT(time, @giotemp))  insert into @bangkhunggio  select CONVERT(VARCHAR(15), LEFT(@giobd\_ngay, 5)) + ' - ' + CONVERT(VARCHAR(15),LEFT(@giotemp, 5))  set @giobd\_ngay = DATEADD(HOUR, @khoangcachgio, CONVERT(time, @giobd\_ngay))  end  return  end  GO |

#### Thủ tục Doanh thu tổng

|  |
| --- |
| create proc [dbo].[sp\_doanhthutong]  as  begin  -- DOANH THU DO AN  declare @doanhthuDA table (id int identity(1,1), doanhthuDA decimal(6, 2))  insert into @doanhthuDA  select SUM(SOLUONG\*GIA) as DT\_DOAN  from DOANCT DACT join KICHCODOAN KCDA on DACT.ID\_KCDA = KCDA.ID\_KCDA  join DONTHANHTOAN DTT on DTT.ID\_DONDA = DACT.ID\_DONDA  -- DOANH THU PHONG VE  declare @doanhthuPV table (id int identity(1,1), doanhthuPV decimal(6, 2))  insert into @doanhthuPV  select SUM(GIA + PHUTHU) as DT\_PHIM  from VEDAT VD join LOAIVE LV on VD.ID\_LOAIVE = LV.ID\_LOAIVE  join GHE G on G.ID\_GHE = VD.ID\_GHE  join LOAIGHE LG on LG.ID\_LOAIGHE = G.ID\_LOAIGHE  join PHONGCHIEU PC on PC.ID\_PHONG = G.ID\_PHONGCHIEU  join SUATCHIEU SC on SC.ID\_SUAT = VD.ID\_SUAT  select iif(doanhthuPV is NULL, 0, doanhthuPV) as DTPV,  iif(doanhthuDA is NULL, 0, doanhthuDA) as DTDA,  (iif(doanhthuDA is NULL, 0, doanhthuDA) + iif(doanhthuPV is NULL, 0, doanhthuPV)) as tongDoanhThu  from @doanhthuDA DTDA join @doanhthuPV DTPV on DTDA.id = DTPV.id  end  GO |

#### Thủ tục Lọc oanh thu tổng theo năm

|  |
| --- |
| create proc [dbo].[sp\_doanhthutong\_nam] (@Nam int)  as  begin  declare @doanhthuDA table (id int identity(1,1), doanhthuDA decimal(6, 2))  insert into @doanhthuDA  select SUM(SOLUONG\*GIA) as DT\_DOAN  from DOANCT DACT join KICHCODOAN KCDA on DACT.ID\_KCDA = KCDA.ID\_KCDA  join DONTHANHTOAN DTT on DTT.ID\_DONDA = DACT.ID\_DONDA  where YEAR(NGAYDAT) like @Nam  -- DOANH THU PV  declare @doanhthuPV table (id int identity(1,1), doanhthuPV decimal(6, 2))  insert into @doanhthuPV  select SUM(GIA + PHUTHU) as DT\_PHIM  from VEDAT VD join LOAIVE LV on VD.ID\_LOAIVE = LV.ID\_LOAIVE  join GHE G on G.ID\_GHE = VD.ID\_GHE  join LOAIGHE LG on LG.ID\_LOAIGHE = G.ID\_LOAIGHE  join PHONGCHIEU PC on PC.ID\_PHONG = G.ID\_PHONGCHIEU  join SUATCHIEU SC on SC.ID\_SUAT = VD.ID\_SUAT  where YEAR(NGAYCHIEU) like @Nam  select iif(doanhthuPV is NULL, 0, doanhthuPV) as DTPV,  iif(doanhthuDA is NULL, 0, doanhthuDA) as DTDA,  (iif(doanhthuDA is NULL, 0, doanhthuDA) + iif(doanhthuPV is NULL, 0, doanhthuPV)) as tongDoanhThu  from @doanhthuDA DTDA join @doanhthuPV DTPV on DTDA.id = DTPV.id  end  GO |

#### Thủ tục Lọc doanh thu tổng theo ngày

|  |
| --- |
| create proc [dbo].[sp\_doanhthutong\_ngay] (@Ngay date)  as  begin  declare @doanhthuDA table (id int identity(1,1), doanhthuDA decimal(6, 2))  insert into @doanhthuDA  select SUM(SOLUONG\*GIA) as DT\_DOAN  from DOANCT DACT join KICHCODOAN KCDA on DACT.ID\_KCDA = KCDA.ID\_KCDA  join DONTHANHTOAN DTT on DTT.ID\_DONDA = DACT.ID\_DONDA  where NGAYDAT like @Ngay  -- DOANH THU PV  declare @doanhthuPV table (id int identity(1,1), doanhthuPV decimal(6, 2))  insert into @doanhthuPV  select SUM(GIA + PHUTHU) as DT\_PHIM  from VEDAT VD join LOAIVE LV on VD.ID\_LOAIVE = LV.ID\_LOAIVE  join GHE G on G.ID\_GHE = VD.ID\_GHE  join LOAIGHE LG on LG.ID\_LOAIGHE = G.ID\_LOAIGHE  join PHONGCHIEU PC on PC.ID\_PHONG = G.ID\_PHONGCHIEU  join SUATCHIEU SC on SC.ID\_SUAT = VD.ID\_SUAT  where NGAYCHIEU like @Ngay  select iif(doanhthuPV is NULL, 0, doanhthuPV) as DTPV,  iif(doanhthuDA is NULL, 0, doanhthuDA) as DTDA,  (iif(doanhthuDA is NULL, 0, doanhthuDA) + iif(doanhthuPV is NULL, 0, doanhthuPV)) as tongDoanhThu  from @doanhthuDA DTDA join @doanhthuPV DTPV on DTDA.id = DTPV.id  end  GO |

#### Thủ tục Lọc doanh thu tổng theo tháng

|  |
| --- |
| create proc [dbo].[sp\_doanhthutong\_thang] (@Thang int, @Nam int)  as  begin  declare @doanhthuDA table (id int identity(1,1), doanhthuDA decimal(6, 2))  insert into @doanhthuDA  select SUM(SOLUONG\*GIA) as DT\_DOAN  from DOANCT DACT join KICHCODOAN KCDA on DACT.ID\_KCDA = KCDA.ID\_KCDA  join DONTHANHTOAN DTT on DTT.ID\_DONDA = DACT.ID\_DONDA  where MONTH(NGAYDAT) like @Thang and YEAR(NGAYDAT) like @Nam  -- DOANH THU PV  declare @doanhthuPV table (id int identity(1,1), doanhthuPV decimal(6, 2))  insert into @doanhthuPV  select SUM(GIA + PHUTHU) as DT\_PHIM  from VEDAT VD join LOAIVE LV on VD.ID\_LOAIVE = LV.ID\_LOAIVE  join GHE G on G.ID\_GHE = VD.ID\_GHE  join LOAIGHE LG on LG.ID\_LOAIGHE = G.ID\_LOAIGHE  join PHONGCHIEU PC on PC.ID\_PHONG = G.ID\_PHONGCHIEU  join SUATCHIEU SC on SC.ID\_SUAT = VD.ID\_SUAT  where MONTH(NGAYCHIEU) like @Thang and YEAR(NGAYCHIEU) like @Nam  select iif(doanhthuPV is NULL, 0, doanhthuPV) as DTPV,  iif(doanhthuDA is NULL, 0, doanhthuDA) as DTDA,  (iif(doanhthuDA is NULL, 0, doanhthuDA) + iif(doanhthuPV is NULL, 0, doanhthuPV)) as tongDoanhThu  from @doanhthuDA DTDA join @doanhthuPV DTPV on DTDA.id = DTPV.id  end  GO |

#### Thủ tục Tìm thời gian

|  |
| --- |
| CREATE PROC [dbo].[SP\_FINDTIMESLOT] @NGAY DATE, @PHONG NCHAR(4), @GIOBD TIME, @GIOKT TIME  AS BEGIN  select \* from SUATCHIEU where ID\_PHONG = @PHONG and ngaychieu = @ngay  and (( GIOBATDAU between convert(time, @giobd) and convert(time, @gioKT)) or ( GIOBATDAU <= @gioKT ) AND (GIOKETTHUC >= @giobd ))  END  GO |

#### Thủ tục Tính tiền tổng đơn thanh toán

|  |
| --- |
| CREATE PROC [dbo].[SP\_GETTONGDON] @ID\_DONTT INT, @TONGTIEN DECIMAL(9,3) OUTPUT  AS BEGIN  DECLARE @ID\_VE NVARCHAR(20) = (SELECT ID\_DONVE FROM DONTHANHTOAN WHERE @ID\_DONTT = ID\_DONTT)  DECLARE @ID\_DONDA NVARCHAR(20) = (SELECT ID\_DONDA FROM DONTHANHTOAN WHERE @ID\_DONTT = ID\_DONTT)  DECLARE @TONG DECIMAL(9,3) = 0  IF(@ID\_VE is NOT null)  BEGIN  EXEC SP\_TINHTONGDONVE @ID\_VE, @TONG OUTPUT  END  IF(@ID\_DONDA is NOT null)  BEGIN  DECLARE @SUB DECIMAL(9,3) = 0  EXEC SP\_TINHTONGDONDA @ID\_DONDA, @SUB OUTPUT  SET @TONG = @TONG + @SUB  END  select @TONGTIEN = @Tong  END  GO |

#### Thủ tục Lịch chiếu theo ngày

|  |
| --- |
| CREATE PROC [dbo].[SP\_LICHCHIEUNGAY] @NGAY DATE, @GIO TIME = NULL, @CONDITION TINYINT = 1, @PHIM NVARCHAR(15) = NULL  AS BEGIN  DECLARE @Lichchieu TABLE(  ID\_SUAT NVARCHAR(30),  NGAYCHIEU DATE,  ID\_PHONG NCHAR(4),  ID\_PHIM NVARCHAR(15),  GIOBATDAU TIME,  GIOKETTHUC TIME,  ID\_NV NVARCHAR(10)  )  declare @maxNgay date = DATEADD(day, 2, @NGAY)  IF(@CONDITION = 1) BEGIN  INSERT INTO @Lichchieu  SELECT SC.\*  FROM PHIM P  JOIN SUATCHIEU SC ON SC.ID\_PHIM = P.ID\_PHIM  WHERE TRANGTHAI = 1 AND ( NGAYCHIEU >= @NGAY AND NGAYCHIEU <= @maxNgay)  GROUP BY SC.ID\_PHIM, TEN, SC.ID\_SUAT, SC.NGAYCHIEU, SC.ID\_PHONG, SC.GIOBATDAU, SC.GIOKETTHUC, SC.ID\_NV  ORDER BY NGAYCHIEU, GIOBATDAU OFFSET 0 ROWS  END  IF(@CONDITION = 0) BEGIN  INSERT INTO @Lichchieu  SELECT SC.\* FROM PHIM P  JOIN SUATCHIEU SC ON SC.ID\_PHIM = P.ID\_PHIM  WHERE TRANGTHAI = 1 AND NGAYCHIEU = @NGAY  GROUP BY SC.ID\_PHIM, TEN, SC.ID\_SUAT, SC.NGAYCHIEU, SC.ID\_PHONG, SC.GIOBATDAU, SC.GIOKETTHUC, SC.ID\_NV  ORDER BY NGAYCHIEU, GIOBATDAU OFFSET 0 ROWS  END  IF(@CONDITION = -1) BEGIN  INSERT INTO @Lichchieu  SELECT SC.\* FROM PHIM P  JOIN SUATCHIEU SC ON SC.ID\_PHIM = P.ID\_PHIM  WHERE TRANGTHAI = 1 AND NGAYCHIEU < @NGAY  GROUP BY SC.ID\_PHIM, TEN, SC.ID\_SUAT, SC.NGAYCHIEU, SC.ID\_PHONG, SC.GIOBATDAU, SC.GIOKETTHUC, SC.ID\_NV  ORDER BY NGAYCHIEU, GIOBATDAU OFFSET 0 ROWS  END  IF(@GIO IS NOT NULL)  BEGIN  IF(@CONDITION = 1)  SELECT \* FROM @Lichchieu WHERE (NGAYCHIEU = @NGAY and GIOBATDAU > @GIO) or(NGAYCHIEU > @NGAY)  ELSE  SELECT \* FROM @Lichchieu WHERE GIOBATDAU > @GIO  END  ELSE IF(@PHIM IS NOT NULL)  BEGIN  SELECT \* FROM @Lichchieu WHERE ID\_PHIM = @PHIM  END  ELSE SELECT \* FROM @Lichchieu  END  GO |

#### Thủ tục Lượng vé đã bán của suất chiếu

|  |
| --- |
| CREATE PROC [dbo].[SP\_LUONGVEBANCUASUAT] @ID\_SUAT NVARCHAR(30)  AS BEGIN  select count(\*) LUONGVE from suatchieu sc join VEDAT vd on sc.ID\_SUAT = vd.ID\_SUAT where sc.ID\_SUAT = @Id\_suat  END  GO |

#### Thủ tục Quản lý đơn thanh toán

|  |
| --- |
| create proc [dbo].[sp\_qldonthanhtoan]  as  begin  select ID\_DONTT, GIOBATDAU as SUATCHIEU, convert(varchar, NGAYDAT, 105) as NGAYDAT, THANHVIEN.ID\_TV, TONG  from DONTHANHTOAN JOIN DONVE ON DONTHANHTOAN.ID\_DONVE = DONVE.ID\_DONVE  JOIN THANHVIEN ON THANHVIEN.ID\_TV = DONTHANHTOAN.ID\_TV  JOIN VEDAT ON VEDAT.ID\_DONVE = DONVE.ID\_DONVE  JOIN SUATCHIEU ON SUATCHIEU.ID\_SUAT = VEDAT.ID\_SUAT  GROUP BY ID\_DONTT, SUATCHIEU.GIOBATDAU,SUATCHIEU.NGAYCHIEU, NGAYDAT, TONG, THANHVIEN.ID\_TV  end  GO |

#### Thủ tục SP\_SELECTLICH

|  |
| --- |
| CREATE PROC [dbo].[SP\_SELECTLICH] @ID\_PHIM nvarchar(15) = Null, @NGAYBD DATE = NULL, @NGAYKT DATE = NULL, @Phong nchar(4) = NULL, @ISCHUACHIEU INT = -1  as begin    DECLARE @TABLE TABLE(ID\_SUAT NVARCHAR(30),NGAYCHIEU DATE, TEN NVARCHAR(MAX), PHONG NCHAR(4), THOILUONG SMALLINT, GIOBATDAU TIME, GIOKETTHUC TIME);  insert @TABLE select ID\_SUAT, ngaychieu, p.TEN TENPHIM, ID\_PHONG PHONG, THOILUONG , GIOBATDAU, GIOKETTHUC from SUATCHIEU sc join PHIM p on sc.ID\_PHIM = p.ID\_PHIM  WHERE( @ID\_PHIM IS NULL OR @ID\_PHIM = SC.ID\_PHIM)  AND ((@NGAYBD IS NULL AND @NGAYKT IS NULL) OR (NGAYCHIEU >= @NGAYBD AND NGAYCHIEU <= @NGAYKT) )  AND (@PHONG IS NULL OR ID\_PHONG = @Phong)  ORDER BY sc.NGAYCHIEU ASC, TENPHIM asc  IF(@ISCHUACHIEU != -1)  BEGIN  declare @now date = getdate();  declare @timeNow time = CAST(GETdATE() AS TIME)  if(@ISCHUACHIEU = 0)  select \* from @TABLE where (NGAYCHIEU < @now ) or (NGAYCHIEU = @now and GIOBATDAU < @timeNow) order by NGAYCHIEU desc, TEN asc  else select \* from @TABLE where (NGAYCHIEU > @now) or (NGAYCHIEU = @now and GIOBATDAU > @timeNow) order by NGAYCHIEU desc, ten asc  END  else select \* from @TABLE  end  GO |

#### Thủ tục SP\_SELECTLICHBYTRANGTHAI

|  |
| --- |
| CREATE PROC [dbo].[SP\_SELECTLICHBYTRANGTHAI] @ID\_PHIM NVARCHAR(15) NULL, @PHONG NCHAR(4) = NULL, @ISCHUACHIEU BIT = 1  as begin  DECLARE @TABLE TABLE(ID\_SUAT NVARCHAR(30),NGAYCHIEU DATE, TEN NVARCHAR(MAX), PHONG NCHAR(4), THOILUONG SMALLINT, GIOBATDAU TIME, GIOKETTHUC TIME);  insert @TABLE EXEC SP\_SELECTLICH @id\_phim = @ID\_PHIM, @PHONG = @PHONG  declare @now date = getdate();  declare @timeNow time = CAST(GETdATE() AS TIME)    if(@ISCHUACHIEU = 0)  select \* from @TABLE where (NGAYCHIEU < @now ) or (NGAYCHIEU = @now and GIOBATDAU < @timeNow)  else select \* from @TABLE where (NGAYCHIEU > @now) or (NGAYCHIEU = @now and GIOBATDAU > @timeNow)  end  GO |

#### Thủ tục SP\_SELECTLICHPHONG

|  |
| --- |
| CREATE PROC [dbo].[SP\_SELECTLICHPHONG] @ID\_PHONG NCHAR(4), @NGAY DATE, @GIO TIME  as begin  DECLARE @SUATCHUACHIEU INT = (SELECT COUNT(SC.ID\_SUAT) from PHONGCHIEU pc join SUATCHIEU sc on pc.ID\_PHONG = sc.ID\_PHONG join PHIM p on p.ID\_PHIM = sc.ID\_PHIM where sc.ID\_PHONG = @ID\_PHONG and NGAYCHIEU = @NGAY and GIOBATDAU > @GIO)    IF(@SUATCHUACHIEU IS NULL OR @SUATCHUACHIEU = 0)  begin  select top 3 pc.ID\_PHONG, TEN, GIOBATDAU, GIOKETTHUC from PHONGCHIEU pc join SUATCHIEU sc on pc.ID\_PHONG = sc.ID\_PHONG join PHIM p on p.ID\_PHIM = sc.ID\_PHIM where sc.ID\_PHONG = @ID\_PHONG and NGAYCHIEU = @NGAY and GIOBATDAU < @GIO order by GIOBATDAU desc  end  else  begin  select top 3 pc.ID\_PHONG, TEN, GIOBATDAU, GIOKETTHUC from PHONGCHIEU pc join SUATCHIEU sc on pc.ID\_PHONG = sc.ID\_PHONG join PHIM p on p.ID\_PHIM = sc.ID\_PHIM where sc.ID\_PHONG = @ID\_PHONG and NGAYCHIEU = @NGAY and GIOBATDAU > @GIO order by GIOBATDAU  end    END  GO |

#### Thủ tục v Sửa suất chiếu của phòng

|  |
| --- |
| CREATE PROC [dbo].[SP\_SUATDACHIEUCUAPHONG] @ID\_PHONG NCHAR(4), @NGAY DATE, @GIO TIME  AS BEGIN  DECLARE @TONG INT = (select count(\*) from PHONGCHIEU pc join SUATCHIEU sc on pc.ID\_PHONG = sc.ID\_PHONG where NGAYCHIEU = @NGAY and sc.ID\_PHONG = @ID\_PHONG)  DECLARE @DACHIEU INT = (select count(\*) from PHONGCHIEU pc join SUATCHIEU sc on pc.ID\_PHONG = sc.ID\_PHONG where NGAYCHIEU = @NGAY and sc.ID\_PHONG = @ID\_PHONG and GIOBATDAU <= @GIO)  select @DACHIEU as dachieu, @tong as tongsuat  END  GO |

#### Thủ tục Thêm đơn mới

|  |
| --- |
| CREATE PROC [dbo].[SP\_THEMDON] @ISDONVE BIT = 1  AS BEGIN  IF(@ISDONVE = 1)  INSERT INTO DONVE VALUES('')  ELSE INSERT INTO DONDOAN VALUES('')  END  GO |

#### Thủ tục Thủ tục Thêm đơn

|  |
| --- |
| create proc [dbo].[sp\_themdonOf] @id\_donthanhtoan INT, @ISDONVE BIT,  @ID\_Don nvarchar(20) output  as begin  IF(@ISDONVE = 1)  BEGIN  EXEC SP\_THEMDON 1  DECLARE @ID\_DONVE NVARCHAR(20) = (select TOP 1 \* from DONVE ORDER BY ID\_DONVE DESC)  select @ID\_don = @ID\_DONVE  END  ELSE  BEGIN  EXEC SP\_THEMDON 0  DECLARE @ID\_DONDA NVARCHAR(20) = (select TOP 1 \* from DONDOAN ORDER BY ID\_DONDA DESC)  select @ID\_don = @ID\_DONDA  END  end  GO |

#### Thủ tục Thêm đơn thanh toán

|  |
| --- |
| create proc [dbo].[sp\_themdontt] @loaiDon int = 0, @id\_tv nvarchar(16) = null, @id\_nv nvarchar(10) = null  as begin  DECLARE @MYDON TABLE (IDDONTT INT ,DONDA NVARCHAR(20), DONVE NVARCHAR(20))  insert into DONTHANHTOAN (ID\_TV, ID\_NV, NGAYDAT,TONG) VALUES  (@id\_tv,@id\_nv,GETDATE(), 0)  declare @ID\_DONTT INT  declare @id\_ve nvarchar(20) = NULL  declare @id\_da nvarchar(20) = NULL  SET @ID\_DONTT = ( SELECT top 1 ID\_DONTT from DONTHANHTOAN order by ID\_DONTT desc)  INSERT INTO @MYDON(IDDONTT) VALUES(@ID\_DONTT)  ---  ---don ve  IF(@loaiDon = 0)  BEGIN  exec sp\_themdonOf @ID\_DONTT,1, @id\_ve output  END  --don do an  ELSE IF(@loaiDon = 1)  BEGIN  exec sp\_themdonOf @ID\_DONTT,0, @id\_da output  END  --ca 2  ELSE  BEGIN  exec sp\_themdonOf @ID\_DONTT,1, @id\_ve output  exec sp\_themdonOf @ID\_DONTT,0, @id\_da output    END  UPDATE @MYDON SET DONVE = @ID\_ve, DONDA = @id\_da WHERE IDDONTT = @ID\_DONTT    update DONTHANHTOAN set ID\_DONDA = @id\_da, ID\_DONVE = @id\_ve where ID\_DONTT = @ID\_DONTT  SELECT \* FROM @MYDON  end  GO |

#### Thủ tục Thống kê khung giờ

|  |
| --- |
| create proc [dbo].[sp\_thongke\_khunggio] (@khoangcach int, @ngay date)  as  begin  declare @table1 table (stt int identity(1,1), khunggio varchar(15))  insert into @table1  select \* from dbo.fn\_KhungGio('7:00', @khoangcach)  declare @table2 table (stt int identity(1,1), sosuatchieu int, sovedat int)    declare @dem int = 1  while @dem <= (select count(\*) from dbo.fn\_KhungGio('7:00', @khoangcach))  begin  declare @khunggio varchar(15) = (select khunggio from @table1 where stt = @dem)  insert into @table2  select (select count(ID\_SUAT) from SUATCHIEU  where (GIOBATDAU >= convert(time, left(@khunggio,5)) and GIOBATDAU < convert(time, right(@khunggio,5))) and NGAYCHIEU like @ngay) as SOSUATCHIEU,  count(ID\_VEDAT) as SOVEDAT from SUATCHIEU SC join VEDAT VD on SC.ID\_SUAT = VD.ID\_SUAT  where (GIOBATDAU >= convert(time, left(@khunggio,5)) and GIOBATDAU < convert(time, right(@khunggio,5))) and NGAYCHIEU like @ngay  set @dem = @dem + 1  end  select khunggio, sosuatchieu, sovedat from @table1 TB1 join @table2 TB2 on TB1.stt = TB2.stt  end  GO |

#### Thủ tục Tính tông đơn đồ ăn

|  |
| --- |
| CREATE PROC [dbo].[SP\_TINHTONGDONDA] @ID\_DONDA NVARCHAR(20), @TONG DECIMAL(9,3) OUTPUT  AS BEGIN  DECLARE @TONGDON DECIMAL(9,3) = 0  set @TONGDON = (select sum(sub.tong) as tong from (SELECT DDA.ID\_DONDA, CT.ID\_KCDA, KC.ID\_DOAN, KC.GIA, CT.SOLUONG, (KC.GIA \* CT.SOLUONG)TONG FROM DONDOAN DDA JOIN DOANCT CT ON DDA.ID\_DONDA = CT.ID\_DONDA JOIN KICHCODOAN KC ON KC.ID\_KCDA = CT.ID\_KCDA where DDA.ID\_DONDA = @ID\_DONDA ) as sub)  select @TONG = @TONGDON  END  GO |

#### Thủ tục Tính tổng đơn vé

|  |
| --- |
| CREATE PROC [dbo].[SP\_TINHTONGDONVE] @ID\_DONVE NVARCHAR(20), @TONG DECIMAL(9,3) OUTPUT  AS BEGIN  DECLARE @TONGDON DECIMAL(9,3) = 0  ---TINH TONG CAC GHE TRONG DONVE  SET @TONGDON = (SELECT SUM(GHEDON.TONGTIEN) TONG FROM  (SELECT V.ID\_GHE,V.ID\_LOAIVE,GIA, PHUTHU, (GIA+PHUTHU)TONGTIEN from VEDAT V  JOIN LOAIVE LV ON V.ID\_LOAIVE = LV.ID\_LOAIVE JOIN GHE G ON V.ID\_GHE = G.ID\_GHE  JOIN LOAIGHE LG ON LG.ID\_LOAIGHE = G.ID\_LOAIGHE WHERE ID\_DONVE = @ID\_DONVE) AS GHEDON)  ---  SELECT @TONG = @TONGDON  END  GO |

#### Thủ tục Cập nhật tổng đơn

|  |
| --- |
| CREATE PROC [dbo].[SP\_UPDATETONGDON] @ID\_DONTT INT  AS BEGIN  DECLARE @T DECIMAL(9,3)  EXEC SP\_GETTONGDON @ID\_DONTT, @T OUTPUT  UPDATE DONTHANHTOAN SET TONG = @T WHERE ID\_DONTT = @ID\_DONTT  END  GO |

#### Thủ tục Xóa đơn thanh toán

|  |
| --- |
| create trigger [dbo].[tg\_xoa\_donThanhToan] on [dbo].[DONTHANHTOAN] instead of delete  as  begin  delete from DONTHANHTOAN where ID\_DONTT like (select ID\_DONTT from deleted)  delete from VEDAT where ID\_DONVE like (select ID\_DONVE from deleted)  delete from DONVE where ID\_DONVE like (select ID\_DONVE from deleted)  delete from DOANCT where ID\_DONDA like (select ID\_DONDA from deleted)  delete from DONDOAN where ID\_DONDA like (select ID\_DONDA from deleted)  end  GO  ALTER TABLE [dbo].[DONTHANHTOAN] ENABLE TRIGGER [tg\_xoa\_donThanhToan]  GO |

#### Trigger Xóa kích cỡ đồ ăn

|  |
| --- |
| create trigger [dbo].[tg\_xoaKCDA] on [dbo].[KICHCODOAN] instead of delete  as  begin  declare @idKC int = (select ID\_KCDA from deleted)  declare @table table (id\_doanct int, id\_doan nvarchar(16), id\_loai nvarchar(4),  tenloai nvarchar(100), tenda nvarchar(100), id\_kichco nchar(1), gia decimal(6,2))  insert @table select ID\_DOANCT, DA.ID\_DOAN, ID\_LOAI, LDA.TEN, DA.TEN, KCDA.ID\_KICHCO, GIA  from DOANCT DACT left join KICHCODOAN KCDA on DACT.ID\_KCDA = KCDA.ID\_KCDA  left join DOAN DA on DA.ID\_DOAN = KCDA.ID\_DOAN  left join LOAIDOAN LDA on LDA.ID = DA.ID\_LOAI  where DACT.ID\_KCDA like @idKC  -- them do an vao bang tam  insert into KCDAHISTORY select \* from @table  -- xoa do an  delete from KICHCODOAN where ID\_KCDA like @idKC  if (select count(\*) from KICHCODOAN where ID\_DOAN like (select ID\_DOAN from deleted where ID\_KCDA like @idKC)) = 0  delete from DOAN where ID\_DOAN like (select ID\_DOAN from deleted where ID\_KCDA like @idKC)  end  GO  ALTER TABLE [dbo].[KICHCODOAN] ENABLE TRIGGER [tg\_xoaKCDA]  GO |

#### Trigger Xóa nhân viên

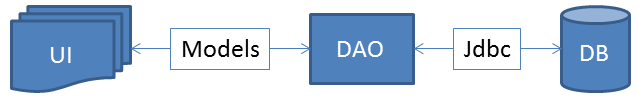
|  |
| --- |
| create trigger [dbo].[tg\_xoa\_nv] on [dbo].[NHANVIEN] instead of delete  as  begin  update PHIM set ID\_NV = NULL where ID\_NV like (select ID\_NV from deleted)  update DONTHANHTOAN set ID\_NV = NULL where ID\_NV like (select ID\_NV from deleted)  delete from NHANVIEN where ID\_NV like (select ID\_NV from deleted)  end  GO  ALTER TABLE [dbo].[NHANVIEN] ENABLE TRIGGER [tg\_xoa\_nv]  GO |

#### Trigger Xóa vé dặt

|  |
| --- |
| create trigger [dbo].[tg\_xoavedat] on [dbo].[VEDAT]  AFTER DELETE  AS BEGIN  DECLARE @T TABLE(ID\_DONVE NVARCHAR(20), LUONGVE INT )  -- DECLARE @ID\_DONVE NVARCHAR(20) = (SELECT TOP 1 ID\_DONVE FROM deleted)  INSERT @T(ID\_DONVE) SELECT ID\_DONVE FROM deleted group by ID\_DONVE;  --DECLARE @VEDAT\_NUM INT = (SELECT COUNT(\*) FROM VEDAT WHERE ID\_DONVE = @ID\_DONVE)    UPDATE @T SET T1.LUONGVE = T2.LUONGVE  from @T AS T1 INNER JOIN  (SELECT ID\_DONVE, COUNT(ID\_VEDAT) AS LUONGVE FROM VEDAT WHERE ID\_DONVE IN (SELECT ID\_DONVE FROM @T) GROUP BY ID\_DONVE) AS T2  ON T1.ID\_DONVE = T2.ID\_DONVE  DELETE FROM DONVE WHERE ID\_DONVE IN (SELECT ID\_DONVE FROM @T WHERE LUONGVE is null or LUONGVE = 0)  SELECT \* FROM @T  --PRINT 'LUONG VE TRONG DON: ' + CAST(@VEdAT\_NUM AS NVARCHAR(40)) + ' DONVE: ' + @ID\_DONVE  END  GO |

## Lập trình JDBC

Trong dự án này chúng ta chọn mô hình lập trình với CSDL như hình sau



Trong đó:

* UI: các thành phần giao diện
* DB: cơ sở dữ liệu
* Jdbc: là lớp tiện ích cung cấp các hàm tiện ích làm việc với CSDL thông qua câu lệnh sql hoặc lời gọi thủ tục lưu
* Model: là các lớp mô tả dữ liệu theo cấu trúc các bảng trong CSDL
* DAO: là các lớp thao tác và truy vấn dữ liệu. Nó có nhiệm vụ chuyển đổi Model sang SQL và ngược lại.

Theo mô hình này thì lập trình giao diện làm việc với CSDL thông qua các lớp model. Model như là một gói dữ liệu được sử dụng để giao tiếp giữa giao diện và các thành phần DAO. DAO mới thực sự là lập trình CSDL nó làm nhiệm vụ nhận model từ giao diện, chuyển đổi sang SQL và sử dụng Jdbc để thực hiện các thao tác dữ liệu. Ngược lại thì truy vấn dữ liệu từ CSDL, chuyển đổi thành model để hiển thị lên các thành phần giao diện.

*Việc tách thành các phần riêng biết giúp quản lý dự án tốt hơn, dễ bảo trì nâng cấp hơn.*

### Lớp hỗ trợ

Các lớp hỗ trợ chứa các phương thức tiện ích giúp chúng ta thực hiện các xử lý trong ứng dụng đơn giản hơn, ngắn gọn hơn, dễ hiểu hơn. Trong phần này chúng ta xây dựng các lớp tiện ích như sau.

#### Lớp tiện ích Authenticator

Hỗ trợ đăng nhập, kiểm tra

|  |
| --- |
| package com.org.app.helper;  import com.org.app.entity.NhanVien;  public class Authenticator {  public static NhanVien USER;  public static void logOut() {  Authenticator.USER = null;  }  public static boolean isLogin() {  return Authenticator.USER != null;  }    public static boolean isQuanLy() {  return USER.getVaitro();  }    } |

#### Lớp tiện ích DateHelper

Hỗ trợ kiểu dữ liệu ngày tháng năm

|  |
| --- |
| package com.org.app.helper;  public class DateHelper {  static SimpleDateFormat formater = new SimpleDateFormat();    public static DateFormat DATE\_FORMAT = new SimpleDateFormat ("dd-MM-yyyy");  public static DateFormat DATE\_SLASH\_FORMAT = new SimpleDateFormat ("dd/MM/yyyy");  public static DateFormat DEFAULT\_DATE\_FORMAT = new SimpleDateFormat ("yyyy-MM-dddd");    public static DateFormat TIME\_FORMAT = new SimpleDateFormat ("HH:mm:ss");  public static DateFormat DATETIME\_FORMAT = new SimpleDateFormat("dd-MM-yyyy HH:mm:ss");  public static DateFormat TIMESTAMP\_FORMAT = new SimpleDateFormat("dd-MM-yyyy HH:mm:ss.SSS");    public static DateTimeFormatter DATE\_FROMATER = DateTimeFormatter.ofPattern("dd-MM-yyyy");  public static DateTimeFormatter DATE\_SLASH\_FROMATER = DateTimeFormatter.ofPattern("dd/MM/yyyy");  public static DateTimeFormatter DEFAULT\_DATE\_FORMATER = DateTimeFormatter.ofPattern("yyyy-MM-dd");    public static Integer DDMMYY\_FORMAT = 1;  public static Integer DDMMYYY\_SLASH\_FORMAT = 2;  public static Integer YYYYMMDD\_FORMAT = 3;    public static java.sql.Date now() {  return toDate(LocalDate.now());  }    //Chuyển đổi String sang Date -- delfaut format = DATE\_SLASH\_FORMAT  public static java.sql.Date toDate(String date) {  return java.sql.Date.valueOf(convertToLocalDate(date));  }    public static java.sql.Date toDate(Object date) {  if(date instanceof java.util.Date)  return java.sql.Date.valueOf(convertToLocalDate((Date)date));  else if(date instanceof LocalDate)  return java.sql.Date.valueOf((LocalDate) date);    return null;  }  //Chuyển đổi từ Date sang String  public static String toString(Date date) {  return toString(date,DateHelper.DDMMYYY\_SLASH\_FORMAT);  }    public static String toString(Date date, int type) {  return geDatetFormat(type).format(date);  }    public static String toString(LocalDateTime ldt) {  return getDatePart(ldt).toString();  }    public static String toString(LocalDateTime ldt, int type) {  return toString(getDatePart(ldt),type);  }    private static DateFormat geDatetFormat(int type) {  switch(type) {  case 1: return DATE\_FORMAT;  case 2: return DATE\_SLASH\_FORMAT;  case 3: return DEFAULT\_DATE\_FORMAT;    default : return DATE\_SLASH\_FORMAT;  }  }    public static java.sql.Date toDate(String value, int type) {  if (value == null) {  throw new RuntimeException("Chuỗi cần parse là null");  }  return java.sql.Date.valueOf(convertToLocalDate(value, type));  }    public static LocalDate convertToLocalDate(String value) {  return convertToLocalDate(value, DDMMYYY\_SLASH\_FORMAT);  }        public static LocalDate convertToLocalDate(String value, int type) {  try {  DateTimeFormatter f = getFormatter(type);  return LocalDate.parse(value, f);    }catch(DateTimeParseException ex) {  ex.printStackTrace();  throw new RuntimeException("Không thể chuyển đổi chuỗi sang ngày");  }  }    private static DateTimeFormatter getFormatter(int type) {  switch(type) {  case 1: return DATE\_FROMATER;  case 2: return DATE\_SLASH\_FROMATER;  case 3: return DEFAULT\_DATE\_FORMATER;    default : return DATE\_SLASH\_FROMATER;  }  }    public static LocalDate convertToLocalDate(Date d) {  return Instant.ofEpochMilli(d.getTime())  .atZone(ZoneId.systemDefault())  .toLocalDate();  }    public static java.sql.Date getDatePart(LocalDateTime ldt) {  return DateHelper.toDate(ldt.toLocalDate());  }    public static void main(String[] args) {  Date d = DateHelper.toDate("23/02/1998");  System.out.println(d);  }    public static String toStringForQuery(Date date) {  return toString(date,YYYYMMDD\_FORMAT);  }    } |

#### Lớp tiện ích ImageUtil

Hỗ trợ lưu hình ảnh

|  |
| --- |
| package com.org.app.helper;  import java.awt.Graphics;  import java.awt.Image;  import java.awt.Rectangle;  import java.awt.image.BufferedImage;  import java.io.File;  import java.io.IOException;  import java.nio.file.Files;  import java.nio.file.Path;  import java.nio.file.Paths;  import java.nio.file.StandardCopyOption;  import javax.imageio.ImageIO;  import javax.swing.ImageIcon;  public class ImageUtil {  //filePath la resources/images/ poster | profilepic  public static boolean saveImage(String filePath, String fileName, File file) {  File dir = new File(filePath);  // Tạo thư mục nếu chưa tồn tại  if (!dir.exists()) {  dir.mkdirs();  }  File newFile = new File(dir, fileName);  try {  Path source = Paths.get(file.getAbsolutePath());  Path destination = Paths.get(newFile.getAbsolutePath());  Files.copy(source, destination, StandardCopyOption.REPLACE\_EXISTING);  return true;  } catch (Exception ex) {  ex.printStackTrace();  return false;  }  }  public static ImageIcon readImage(String filePath, String fileName) {  File file = new File(filePath, fileName);  return new ImageIcon(file.getAbsolutePath());  }  public static boolean deleteImage(String filePath, String fileName) {  try {  File file = new File(filePath, fileName);  return file.delete();  } catch (Exception e) {  e.printStackTrace();  }  return false;  }  public static ImageIcon getPosterImage(String fileName) {  try {  String path = "/images/posters/";  return new ImageIcon(ImageUtil.class.getResource(path + fileName));  } catch (Exception e) {  e.printStackTrace();  throw new RuntimeException("Lỗi đọc hình ảnh.");  }  }    public static ImageIcon getAvtImage(String fileName) {  try {  String path = "/images/avt/";  return new ImageIcon(ImageUtil.class.getResource(path + fileName));  } catch (Exception e) {  e.printStackTrace();  throw new RuntimeException("Lỗi đọc hình ảnh.");  }  }  } |

#### Lớp tiện ích InputValiHelper

Hỗ trợ kiểm tra dữ liệu đưa vào ô nhập trên form

|  |
| --- |
| package com.org.app.helper;  import javax.swing.JTextField;  import javax.swing.text.JTextComponent;  public class InputValidlHelper {  public static boolean isEmpty(JTextComponent... txt) {  for(int i = 0; i<txt.length; i++) {  if(txt[i].getText().isBlank()) return true;  }  return false;  }    public static void clear(JTextComponent txt) {  txt.setText("");  }    public static void clear(JTextComponent... txts) {  for(int i = 0; i<txts.length; i++) {  clear(txts[i]);  }  }    public static boolean isValidEmail(String input){  return input.matches("^([\\w\\-\\.]+)\\@(\\w+)(\\.[a-z]{2,10}){1,2}$");  }  } |

#### Lớp tiện ích JDBC

Hỗ trợ kết nối cơ sở dữ liệu

|  |
| --- |
| package com.org.app.helper;  import java.sql.Connection;  import java.sql.DriverManager;  import java.sql.PreparedStatement;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.logging.Level;  import java.util.logging.Logger;  public class JDBCHelper {  private static String driver = "com.microsoft.sqlserver.jdbc.SQLServerDriver";  private static String dburl = "jdbc:sqlserver://localhost;databaseName=cineboost\_fix\_2";  private static String username = "sa";  private static String password = "123456";  private static String errorMess = JDBCHelper.class.getName();  static {  try {  Class.forName(driver);  } catch (ClassNotFoundException ex) {  throw new RuntimeException(errorMess + ": " + "Lỗi Driver");  }  }  public static PreparedStatement prepareStatement(String sql, Object... args) throws SQLException {  Connection connection = DriverManager.getConnection(dburl, username, password);  PreparedStatement pstmt = null;  if (sql.trim().startsWith("{")) {  pstmt = connection.prepareCall(sql);  } else {  pstmt = connection.prepareStatement(sql);  }  for (int i = 0; i < args.length; i++) {  pstmt.setObject(i + 1, args[i]);  }  return pstmt;  }  public static int excecuteStatement(String sql, Object... args) {  try {  PreparedStatement stmt = prepareStatement(sql, args);  return stmt.executeUpdate();  } catch (SQLException e) {  e.printStackTrace();  throw new RuntimeException(errorMess + ": " + "Lỗi execute querry SQL");  }  }  public static ResultSet getResultSetByQuery(String sql, Object... args) {  try {  PreparedStatement stmt = prepareStatement(sql, args);  return stmt.executeQuery();  } catch (SQLException e) {  e.printStackTrace();  throw new RuntimeException(errorMess + ": " + "Lỗi execute querry SQL");  }  }  public static void main(String[] args) {  JDBCHelper.getResultSetByQuery("SELECT \* FROM NHANVIEN");  }  } |

#### Lớp tiện ích MessageHelper

Hỗ trợ đưa ra thông báo

|  |
| --- |
| package com.org.app.helper;  import com.org.app.util.ColorAndIconBank;  import com.org.app.util.ScaleImageIconGenerator;  import java.awt.Color;  import java.awt.Component;  import java.net.URL;  import javax.swing.ImageIcon;  import javax.swing.JOptionPane;  public class MessageHelper {  private static ImageIcon SuccessIcon = ColorAndIconBank.Icon.SUCCESS.getIcon();  private static ImageIcon ErrorIcon = ColorAndIconBank.Icon.ERROR.getIcon();  private static ImageIcon InfoIcon = ColorAndIconBank.Icon.INFO.getIcon();  public static final int INFORMATION\_MESSAGE = 1;  public static final int ERROR\_MESSAGE = 0;  public static void message(Component parent, String message, int messageType) {  message(parent, "Thông báo", message,messageType);  }    public static void message(Component parent, String message, ImageIcon icon) {  JOptionPane.showMessageDialog(parent, message,  "Thông báo", 1,icon);  }    public static void message(Component parent, String message) {  JOptionPane.showMessageDialog(parent, message,  "Thông báo", 1,InfoIcon);  }    public static void message(Component parent, String message, ColorAndIconBank.Icon icon) {  JOptionPane.showMessageDialog(parent, message,  "Thông báo", 1,icon.getIcon());  }    public static void message(Component parent, String title, String message, int messageType) {  ImageIcon icon = messageType == INFORMATION\_MESSAGE ? SuccessIcon : ErrorIcon;  JOptionPane.showMessageDialog(parent, message,  title, messageType, icon);  }  public static boolean confirm(Component parent, String message) {  return confirm(parent, "Thông báo", message);  }    public static boolean confirm(Component parent, String title, String message) {  int result = JOptionPane.showConfirmDialog(parent, message,  title,  JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE, InfoIcon);  return result == JOptionPane.YES\_OPTION;  }  } |

#### Lớp tiện ích SettingIcnHelper

Hỗ trợ chỉnh kích thước đặt icon cho label ảnh

|  |
| --- |
| package com.org.app.helper;  import com.org.app.util.ScaleImageIconGenerator;  import java.awt.Image;  import javax.swing.ImageIcon;  import javax.swing.JComponent;  import javax.swing.JLabel;  public class SettingIconHelper {  public static void setIconFor(JLabel comp, ImageIcon image) {  comp.setIcon(ScaleImageIconGenerator.getScaledIconOf(image, comp));  }    public static void setPosterFor(JLabel comp, String name) {  ImageIcon poster = ImageUtil.getPosterImage(name);  System.out.println("pso = "+poster.toString());  setIconFor(comp, poster);  }    public static void setPosterFor(JLabel comp, int w, int h, String name) {  ImageIcon poster = null;  try {  poster = ImageUtil.getPosterImage(name);    }catch(Exception e){  poster = ImageUtil.getPosterImage("poster1.png");  Image img = poster.getImage().getScaledInstance(w, h, Image.SCALE\_SMOOTH);  comp.setIcon(new ImageIcon(img));    }  }    public static void setIconSourceFor(JLabel comp, ImageIcon image) {    }    public static void setIconSourceFor(JLabel comp, String name) {  comp.setIcon(ScaleImageIconGenerator.getScaledSourcesIcon(name, comp));  }    public static void setIconFor(JLabel comp, String name) {  comp.setIcon(ScaleImageIconGenerator.getScaledIconOf(name, comp));  }    } |

#### Lớp tiện ích TimeHelper

Hỗ trợ thao tác với kiểu dữ liệu thời gian, chuyển đổi các kiểu dữ liệu thời gian liên quan

|  |
| --- |
| package com.org.app.helper;  public class TimeHelper {  static DateTimeFormatter HH\_MM\_FORMAT = DateTimeFormatter.ofPattern("HH:mm");  public static Time now() {  LocalTime now = LocalTime.now();  return Time.valueOf(now);  }    public static LocalDateTime getDateTimeNow() {  return LocalDateTime.now();  }    public static Time getTime(LocalDateTime ldt) {  return toSQLTime(ldt.toLocalTime());  }    public static Time getTime(Date date) {  return toSQLTime(convertToLocalDateTime(date).toLocalTime());  }    public static String toStringWithHourMintues(Time time) {  return time.toLocalTime().format(HH\_MM\_FORMAT);  }    public static String toString(Time t) {return t.toString();}    public static String toString(Date d) {return convertToLocalDateTime(d).toLocalTime().toString();}    public static String toString(LocalDateTime ldt) {return getTime(ldt).toString();}  private static LocalDateTime convertToLocalDateTime(Date d) {  return d.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime();  }    public static String convertMinuteToTimeFormat(Integer minutes) {  String m = String.valueOf(minutes);  Duration d = Duration.ofMinutes(Long.valueOf(m));  System.out.println("ThoiLuong: " + d.toHours() + ": " + d.toMinutesPart() + ":" + d.toSecondsPart());  return d.toHours() + ":" + d.toMinutesPart() + ":" + d.toSecondsPart();  }  public static String convertMinuteToTimeFormat(String minutes) {  return convertMinuteToTimeFormat(Integer.parseInt(minutes));  }  public static Time add(Time start, Integer minutes) {  if(start == null) throw new RuntimeException("Thoi gian khong the null");  return toSQLTime(start.toLocalTime().plusMinutes(minutes));  }  public static Time toTime(Integer hours, Integer minutes) {  try {  return toSQLTime(LocalTime.of(hours, minutes));  } catch (Exception e) {  e.printStackTrace();  throw new RuntimeException("Thời gian không hợp lệ");  }  }  public static Time toTime(Integer hours, Integer minutes, Integer seconds) {  try {  return toSQLTime(LocalTime.of(hours, minutes, seconds));  } catch (Exception e) {  e.printStackTrace();  throw new RuntimeException("Thoi gian khong hop le");  }  }  public static Time toSQLTime(LocalTime lt) {  return Time.valueOf(lt);  }  public static LocalTime toLocalTime(Time t) {  return t.toLocalTime();  }  public static Integer compare(Time t1, Time t2) {  if(t1 == null || t2 == null) throw new RuntimeException("Thoi gian khong the null");  LocalTime ta = toLocalTime(t1);  LocalTime tb = toLocalTime(t2);  return ta.compareTo(tb);  }  } |

#### Lớp tiện ích ColorAndIconBank

Lưu trữ một số màu mặc định cho giao diện và icon thông báo cho messagehelper

|  |
| --- |
| package com.org.app.util;  import java.awt.Color;  import java.util.HashMap;  import java.util.Map;  import javax.swing.ImageIcon;  public class ColorAndIconBank {  public static String path = "";  public static ImageIcon CONTENT\_PANEL\_BACKGROUND = new ImageIcon("resources/icon/pic.jpg");  public static ImageIcon SIDE\_PANEL\_BACKGROUND = new ImageIcon(ColorAndIconBank.class.getResource("/icon/sideBar\_bckg.png"));  public static final Color SIDEMENU\_COLOR\_HOVER = Color.decode("#05595B");    public static final Color SIDEMENU\_COLOR\_SELECTED = new Color(0,102,102);  public static final Color SIDEMENU\_COLOR\_HOVERS = new Color(72,143,177);    public static final Color SIDEMENU\_TEXT\_DEFAULT = new Color(51,51,51);  public static final Color SIDEMENU\_TEXT\_SELECTED = new Color(255,255,255);  public static final Color SIDEMENU\_TEXT\_TITLED = new Color(251,240,155,250);  public static final Color SIDEMENU\_TEXT\_BORDER = new Color(251,248,192,255);    public static final Color INPUT\_COLOR\_FOCUS = new Color(199,199,199,100);    public static final Color FILTER\_BAR\_BACKGROUND = new Color(234,238,208);  public static final Color FILTER\_BAR\_BORDER\_TEXT = Color.RED;    public static final Color REP\_COLOR\_PART = new Color(223,237,186, 150);    public static final Color TOOLBAR\_BACKGROUND\_COLOR\_DEFAULT = Color.decode("#345368");  public static final Color TOOLBAR\_BACKGROUND\_COLOR\_SELECTED = Color.decode("#FFFFFF");  public static final Color TOOLBAR\_BACKGROUND\_COLOR\_ENTERED = Color.decode("#4E6B7E");      public static final Color TOOLBAR\_FOREGROUND\_COLOR\_DEFAULT = Color.decode("#FFFFFF");  public static final Color TOOLBAR\_FOREGROUND\_COLOR\_SELECTED = Color.decode("#345368");    public static final Color SEAT\_COLOR\_SELECTED = Color.decode("#F4AA24");  public static final Color SEAT\_FOREGROUND\_COLOR\_SELECTED = Color.WHITE;    public static final Color NORMAL\_SEAT\_COLOR = Color.decode("#DDDDDD");  public static final Color NORMAL\_SEAT\_FOREGROUND\_COLOR = Color.BLACK;    public static final Color VIP\_SEAT\_COLOR = Color.decode("#4D648D");  public static final Color VIP\_SEAT\_FOREGROUND\_COLOR = Color.WHITE;    public static final Color RESERVED\_SEAT\_COLOR = Color.decode("#A52B24");  public static final Color RESERVED\_SEAT\_FOREGROUND\_COLOR = Color.WHITE;    public enum Icon {  ADD(1),  SAVE(2),  DELETE(3),  UPDATE(5),  SUCCESS(6),  SEARCH(7),  EXIT(8),  OPEN(9),  INFO(11),  LOGIN(12),  LOADING(13),  ERROR(-1),  VALID\_ERROR(-2);    Map<Integer,String> iconWithMess = new HashMap<>();  private int index;    Icon(int index) {  this.index = index;  createMap();  }    public ImageIcon getIcon() {  return new ImageIcon(ColorAndIconBank.class.getResource(iconWithMess.get(index)));  }    public boolean isError() {  return index < 0;  }    private void createMap() {  String packageName = "/icon/";  String extension = ".png";  iconWithMess.put(1, packageName + "add" + extension);  iconWithMess.put(2, packageName + "save" +extension);  iconWithMess.put(3, packageName + "trash\_can" + extension);  iconWithMess.put(5, packageName + "update\_32" + extension);  iconWithMess.put(6, packageName + "success\_icon" +extension);  iconWithMess.put(7, packageName + "search" +extension);  iconWithMess.put(8, packageName + "exit" +extension);  iconWithMess.put(9, packageName + "open" +extension);  iconWithMess.put(11, packageName + "info\_icon" +extension);  iconWithMess.put(12, packageName + "login\_success" +extension);  iconWithMess.put(13, packageName + "loading.gif");  iconWithMess.put(-1, packageName + "error\_icon" + extension);  iconWithMess.put(-2, packageName + "valid\_error"+extension);    }  }  } |

#### Lớp tiện ích CustomGridlayOut

Hỗ trợ căn chỉnh kích thước cho GridLayout

|  |
| --- |
| package com.org.app.util;  import java.awt.GridLayout;  public class CustomGridLayOut {  GridLayout layout;  public CustomGridLayOut(int total, int defaultColNum) {  layout = new GridLayout(getRowNum(total, defaultColNum),defaultColNum);  }    public CustomGridLayOut(int total, int defaultColNum, int hGap, int vGap) {  this(total,defaultColNum);  getLayOut().setVgap(vGap);  getLayOut().setHgap(hGap);  }    private int getRowNum(int total, int defaultCol) {  if( total % defaultCol > 0) return total/defaultCol + 1;  return total/defaultCol;  }    public GridLayout getLayOut() {  return layout;  }  } |

#### Lớp tiện ích ExcelExporter

Hỗ trợ xuất Excel

|  |
| --- |
| package com.org.app.util;  import org.apache.poi.ss.usermodel.Cell;  import org.apache.poi.ss.usermodel.HorizontalAlignment;  import org.apache.poi.ss.usermodel.Row;  import org.apache.poi.ss.usermodel.VerticalAlignment;  import org.apache.poi.ss.util.CellRangeAddress;  import org.apache.poi.xssf.usermodel.XSSFCellStyle;  import org.apache.poi.xssf.usermodel.XSSFFont;  import org.apache.poi.xssf.usermodel.XSSFRow;  import org.apache.poi.xssf.usermodel.XSSFSheet;  import org.apache.poi.xssf.usermodel.XSSFWorkbook;  import javax.swing.\*;  import java.io.File;  import java.io.FileNotFoundException;  import java.io.FileOutputStream;  import java.io.IOException;  import java.text.SimpleDateFormat;  import java.util.Date;  public class ExcelExporter {  public String getCurrentDateAsString() {  SimpleDateFormat dt = new SimpleDateFormat("dd-MM-yyyy | hh:mm:ss");  return dt.format(new Date());  }  public void exportTable(JTable table, File file, String headerTitle, int numOfColumns) throws IOException {  XSSFWorkbook wb = new XSSFWorkbook();  XSSFSheet sheet = wb.createSheet("Data");  createHeader(headerTitle, numOfColumns, wb, sheet);  createTitle(wb, sheet, table);  insertData(sheet, table);  setColumnWidth(sheet);  saveSheet(wb, file);  }  public void saveSheet(XSSFWorkbook wb, File file) {  try {  FileOutputStream outputStream = new FileOutputStream(file);  wb.write(outputStream);  wb.close();  } catch (FileNotFoundException e) {  e.printStackTrace();  } catch (IOException e) {  e.printStackTrace();  }  }  public void setColumnWidth(XSSFSheet sheet) {  sheet.setColumnWidth(0, 7000);  sheet.setColumnWidth(1, 7000);  sheet.setColumnWidth(2, 7000);  sheet.setColumnWidth(3, 7000);  sheet.setColumnWidth(4, 7000);  sheet.setColumnWidth(5, 7000);  sheet.setColumnWidth(6, 7000);  }  public void insertData(XSSFSheet sheet, JTable table) {  int currentRow = 4;  for (int i = 0; i < table.getRowCount(); i++) {  Row row = sheet.createRow(currentRow++);  for (int j = 0; j < table.getColumnCount(); j++) {  Cell cell = row.createCell(j);  Object value = table.getValueAt(i, j);  String content = "";  if (value != null) {  content = value.toString();  }  cell.setCellValue(content);  }  }  }  public void createTitle(XSSFWorkbook wb, XSSFSheet sheet, JTable table) {  XSSFRow rowCol3 = sheet.createRow(3);  XSSFCellStyle style1 = wb.createCellStyle();  XSSFFont font1 = wb.createFont();  font1.setFontName("Time New Roman");  font1.setFontHeightInPoints((short) 15);  font1.setBold(true);  style1.setFont(font1);  style1.setAlignment(HorizontalAlignment.CENTER);  for (int i = 0; i < table.getColumnCount(); i++) {  Cell cell = rowCol3.createCell(i);  cell.setCellValue(table.getColumnName(i));  rowCol3.getCell(i).setCellStyle(style1);  }  }  public void createHeader(String headerValue, int numOfColumns, XSSFWorkbook wb, XSSFSheet sheet) {  XSSFRow rowCol = sheet.createRow(0);  XSSFCellStyle style = wb.createCellStyle();  XSSFFont font = wb.createFont();  font.setFontName("Time New Roman");  font.setFontHeightInPoints((short) 12);  font.setBold(true);  font.setItalic(true);  style.setFont(font);  style.setWrapText(true);  style.setAlignment(HorizontalAlignment.CENTER);  style.setVerticalAlignment(VerticalAlignment.CENTER);  Cell cell0 = rowCol.createCell(0);  cell0.setCellValue(headerValue + "\n Ngày lập báo cáo: " + getCurrentDateAsString());  cell0.setCellStyle(style);  sheet.addMergedRegion(new CellRangeAddress(0, 2, 0, numOfColumns - 1));  }  } |

#### Lớp tiện ích FixedSizeCoatingPanel

Hỗ trợ căn chỉnh lại vị trí cho Component đặt vào GridLayout

|  |
| --- |
| package com.org.app.util;  import java.awt.Color;  import java.awt.GridBagLayout;  import javax.swing.JPanel;  import javax.swing.border.MatteBorder;  public class FixedSizeCoatingPanel {    public static JPanel getCoatingJPanel(JPanel panel) {  JPanel p = new JPanel(new GridBagLayout());  p.setOpaque(false);  p.add(panel);  return p;  }    } |

#### Lớp tiện ích InputFocusGUIUtil

Hỗ trợ thay đổi màu khi đưa chuột vào các TextFField và khi bỏ chọn TextField

|  |
| --- |
| package com.org.app.util;  import java.awt.Color;  import java.awt.event.FocusAdapter;  import java.awt.event.FocusEvent;  import javax.swing.JComponent;  import javax.swing.JLabel;  import javax.swing.JTextField;  public class InputFocusGUIUtil {    static Color lblO;    public static void setFocusEffect(Color focusColor, Color unFocusColor, JTextField[] texts, JLabel...lbl) {  for (int i = 0; i < texts.length; i++) {  setFocusEffect(focusColor, unFocusColor, texts[i], lbl[i]);    }  }  public static void setFocusEffect(Color focusColor, Color unFocusColor, JTextField textField, JLabel lbl) {  lblO = lbl.getForeground();  textField.addFocusListener(new FocusAdapter() {  @Override  public void focusGained(FocusEvent e) {  textField.setBackground(focusColor);  setForegroundLabel(Color.BLACK, lbl);    }  @Override  public void focusLost(FocusEvent e) {  textField.setBackground(unFocusColor);  setForegroundLabel(lblO, lbl);  }    });    }    public static void setForegroundLabel( Color or, JLabel lbl) {  lbl.setForeground(or);      }  public static void setFocusEffect(Color focusColor, Color unFocusColor, JTextField textField, JLabel lbl, Color lblFocusColor) {  lblO = lbl.getForeground();  textField.addFocusListener(new FocusAdapter() {  @Override  public void focusGained(FocusEvent e) {  textField.setBackground(focusColor);  setForegroundLabel(lblFocusColor, lbl);    }  @Override  public void focusLost(FocusEvent e) {  textField.setBackground(unFocusColor);  setForegroundLabel(lblO, lbl);  }    });    }      public static void setFocusEffect(Color focusColor, Color unFocusColor, JTextField[] texts, Color focusLabelColor, JLabel...lbl) {  for (int i = 0; i < texts.length; i++) {  setFocusEffect(focusColor, unFocusColor, texts[i], lbl[i], focusLabelColor);    }  }  } |

#### Lớp tiện ích InvalidInputExceoption

Lớp Exceoption được viết để báo lỗi định dạng của các thực thể

|  |
| --- |
| package com.org.app.util;  public class InvalidInputException extends Exception{  public InvalidInputException() {  }  public InvalidInputException(String message) {  super(message);  }  } |

#### Lớp tiện ích MouseHoverEffect

Hỗ trợ thay đổi màu nền của nút khi đưa chuột vào label

|  |
| --- |
| package com.org.app.util;  import com.formdev.flatlaf.ui.FlatBorder;  import com.formdev.flatlaf.ui.FlatLineBorder;  import java.awt.Color;  import java.awt.Font;  import java.awt.Insets;  import java.awt.event.MouseAdapter;  import java.awt.event.MouseEvent;  import java.awt.event.MouseListener;  import javax.swing.JButton;  import javax.swing.JComponent;  import javax.swing.JLabel;  import javax.swing.border.MatteBorder;  import javax.swing.plaf.basic.BasicBorders;  public class MouseHoverEffect {  static Color or;  public static void buttonMouseOverBoldEffect(Color c, JButton btn) {  Color or = btn.getBackground();  btn.addMouseListener(  new MouseAdapter() {  @Override  public void mouseEntered(MouseEvent e) {  btn.setBackground(c);  boldText(true, btn);  }  @Override  public void mouseExited(MouseEvent e) {  btn.setBackground(or);  boldText(false, btn);  }  }  );  }  public static void boldText(boolean bold, JButton btn) {  Font f = btn.getFont();  if (bold) {  btn.setFont(new Font(f.getName(), Font.BOLD, f.getSize()));  } else {  btn.setFont(new Font(f.getName(), Font.PLAIN, f.getSize()));  }  }  public static void buttonMouseOverEffect(Color c, JButton btn, boolean textOnly) {  or = btn.getBackground();  if (textOnly) {  or = btn.getForeground();  btn.addMouseListener(  new MouseAdapter() {  @Override  public void mouseEntered(MouseEvent e) {  btn.setForeground(c);  }  @Override  public void mouseExited(MouseEvent e) {  btn.setForeground(or);  }  }  );  } else {  mouseOverEffect(c, btn);  }  }  public static void mouseOverEffect(Color c, JComponent lbl) {  Color or = lbl.getBackground();  lbl.addMouseListener(  new MouseAdapter() {  @Override  public void mouseEntered(MouseEvent e) {  lbl.setBackground(c);  }  @Override  public void mouseExited(MouseEvent e) {  lbl.setBackground(or);  }  }  );  }  public static void mouseOverEffect(Color c, JComponent... lbls) {  for (int i = 0; i < lbls.length; i++) {  mouseOverEffect(c, lbls[i]);  }  }  public static void mouseOverEffect(Color c, JComponent[] parents, JComponent[] children, boolean needLeftBorder) {  if (parents.length != children.length) {  throw new RuntimeException("Arrays of Component not matched");  } else {  for (int i = 0; i < parents.length; i++) {  mouseOverEffect(c, parents[i], children[i],needLeftBorder);  }  }  }  public static void mouseOverEffect(Color c, JComponent parent, JComponent child, boolean needLeftBorder) {  parent.repaint();  parent.validate();  child.addMouseListener(  new MouseAdapter() {  @Override  public void mouseEntered(MouseEvent e) {  parent.setBackground(c);  parent.repaint();  parent.validate();  parent.setOpaque(true);  if(needLeftBorder) setBorder(true, parent);  child.setForeground(ColorAndIconBank.SIDEMENU\_TEXT\_SELECTED);  child.validate();  child.repaint();  }  @Override  public void mouseExited(MouseEvent e) {  parent.repaint();  parent.validate();  parent.setOpaque(false);  if(needLeftBorder) setBorder(false, parent);    child.validate();  child.repaint();  child.setForeground(ColorAndIconBank.SIDEMENU\_TEXT\_DEFAULT);  }  }  );  }    public static void setBorder(boolean border, JComponent c){  if(border)  c.setBorder(new MatteBorder(new Insets(0,5,0,0),ColorAndIconBank.SIDEMENU\_TEXT\_BORDER));  else c.setBorder(null);    }  } |

#### Lớp tiện ích ScaleImageIconGenerator

Căn chỉnh kích thước icon phù hợp với JLabel

|  |
| --- |
| package com.org.app.util;  import java.awt.Image;  import javax.swing.ImageIcon;  import javax.swing.JComponent;  public class ScaleImageIconGenerator {  public static ImageIcon getScaledSourcesIcon(String name, JComponent comp) {  ImageIcon i = getImageFromResources(name);  Image img = i.getImage().getScaledInstance(comp.getWidth(), comp.getHeight(), Image.SCALE\_SMOOTH);  return new ImageIcon(img);  }    public static ImageIcon getScaledIconOf(ImageIcon icon, JComponent comp) {  Image img = icon.getImage().getScaledInstance(comp.getWidth(), comp.getHeight(), Image.SCALE\_SMOOTH);  return new ImageIcon(img);  }    public static ImageIcon getImageFromResources(String name) {  return new ImageIcon(ScaleImageIconGenerator.class.getResource("/icon/"+name));  }    public static ImageIcon getScaledIconOf(String name, JComponent comp) {  ImageIcon i = new ImageIcon(ScaleImageIconGenerator.class.getResource(name));  Image img = i.getImage().getScaledInstance(comp.getWidth(), comp.getHeight(), Image.SCALE\_SMOOTH);  return new ImageIcon(img);  }  public static void main(String[] args) {  System.out.println(ScaleImageIconGenerator.class.getResource("/com/org/app/resources/").toExternalForm());  }  } |

#### Lớp tiện ích SideMenuButton

Lớp hỗ trợ cho side menu

|  |
| --- |
| package com.org.app.util;  import static com.org.app.util.MouseHoverEffect.setBorder;  import java.awt.Color;  import java.awt.event.MouseAdapter;  import java.awt.event.MouseEvent;  import java.util.Arrays;  import java.util.Objects;  import javax.swing.ImageIcon;  import javax.swing.JComponent;  import javax.swing.JLabel;  import javax.swing.JPanel;  public class SideMenuButton extends MouseHoverEffect{  JLabel lbl;  JComponent binding;  boolean isSelected;  boolean isEnable;    ImageIcon[] icons = new ImageIcon[2];      public SideMenuButton(JLabel lbl, JComponent binding) {  this.lbl = lbl;  this.binding = binding;  isSelected = false;  }    public SideMenuButton(JLabel lbl, JComponent binding,Color c,boolean border) {  this(lbl,binding);  mouseOverEffect(c,border);  }    public JLabel getButton() {  return lbl;  }  public boolean isIsSelected() {  return isSelected;  }  public void setIsSelected(boolean selected) {    this.isSelected = selected;  if(!this.isSelected || !this.lbl.isEnabled()) {  clearState();  }else {  binding.setBackground(ColorAndIconBank.SIDEMENU\_COLOR\_SELECTED);  binding.repaint();  binding.validate();  binding.setOpaque(true);  setBorder(true, binding);  lbl.repaint();  lbl.validate();  lbl.setForeground(ColorAndIconBank.SIDEMENU\_TEXT\_SELECTED);  }    }  public void clearState() {  binding.repaint();  binding.validate();  binding.setOpaque(false);  setBorder(false, binding);  lbl.validate();  lbl.repaint();  lbl.setForeground(ColorAndIconBank.SIDEMENU\_TEXT\_DEFAULT);  }    public void mouseOverEffect(Color c, boolean needLeftBorder) {  binding.repaint();  binding.validate();  lbl.addMouseListener(  new MouseAdapter() {  @Override  public void mouseEntered(MouseEvent e) {  if(isSelected) return;  binding.setBackground(c);  binding.repaint();  binding.validate();  binding.setOpaque(true);  if(needLeftBorder) setBorder(true, binding);  lbl.setForeground(ColorAndIconBank.SIDEMENU\_TEXT\_SELECTED);  lbl.validate();  lbl.repaint();  }  @Override  public void mouseExited(MouseEvent e) {  if(isSelected) return;  binding.repaint();  binding.validate();    binding.setOpaque(false);  binding.setBackground(new Color(0,0,0,0));  if(needLeftBorder) setBorder(false, binding);  lbl.setForeground(ColorAndIconBank.SIDEMENU\_TEXT\_DEFAULT);  }  }  );  }  @Override  public int hashCode() {  int hash = 7;  hash = 97 \* hash + Objects.hashCode(this.lbl);  hash = 97 \* hash + Objects.hashCode(this.binding);  hash = 97 \* hash + (this.isSelected ? 1 : 0);  hash = 97 \* hash + Arrays.deepHashCode(this.icons);  return hash;  }  @Override  public boolean equals(Object obj) {  if (this == obj) {  return true;  }  if (obj == null) {  return false;  }  if (getClass() != obj.getClass()) {  return false;  }  final SideMenuButton other = (SideMenuButton) obj;  if (this.isSelected != other.isSelected) {  return false;  }  if (!Objects.equals(this.lbl, other.lbl)) {  return false;  }  if (!Objects.equals(this.binding, other.binding)) {  return false;  }  if (!Arrays.deepEquals(this.icons, other.icons)) {  return false;  }  return true;  }  } |

#### Lớp tiện ích ShowPasswordInterfaceUtil

Ẩn hiện Password

|  |
| --- |
| package com.org.app.util;  import javax.swing.JPasswordField;  public interface ShowPasswordInterfaceUtil {  default public void showPassword(JPasswordField p) {  p.setEchoChar('\u0000');  }    default public void hidePassword(JPasswordField p) {  p.setEchoChar('\*');  }    } |

#### Lớp tiện ích SubFrame

Hỗ trợ gọi subFrame

|  |
| --- |
| package com.org.app.util;  import java.awt.Container;  import javax.swing.JComponent;  import javax.swing.JFrame;  import javax.swing.JPanel;  public interface SubFrame<F extends JFrame> extends SubPanelCreatorInterfaces<SubFrame>{    default public Container getContentPane() {  return ((JFrame)this).getContentPane();  }    public javax.swing.JPanel getContentPanelFor(JComponent panel);    public SubPanelCreator getSubPanelCreator();    } |

#### Lớp tiện ích SubPanelCreator

Hỗ trợ lấy Panel và thêm vào cửa sổ chính

|  |
| --- |
| package com.org.app.util;  import java.awt.Dimension;  import javax.swing.JComponent;  import javax.swing.JFrame;  import javax.swing.JPanel;  public abstract class SubPanelCreator<E extends SubFrame> {    E frame;  JPanel contentPanel;    public JPanel createPanelFor(JComponent comp) {  resize(comp.getWidth(), comp.getHeight());  return contentPanel;  }  public SubPanelCreator (E frame) {  this.frame = frame;  contentPanel = getContentPanel();  }    public JPanel getContentPanel() {  return (JPanel) frame.getContentPane();  }    public void resize(int width, int height) {  contentPanel.setPreferredSize(new Dimension(width,height));  }  public abstract void render();    public SubPanelCreator getSubPanel() {return this;};  } |

#### Lớp tiện ích ValidationUtil

Hỗ trợ thông báo lỗi

|  |
| --- |
| package com.org.app.util;  import java.text.SimpleDateFormat;  public class ValidationUtil {  public static boolean isEmpty(String input){  return input.equals("");  }  public static boolean isLenghtLess(String input, int lenght){  return input.length() < lenght;  }    public static boolean isLengthMore(String input, int lenght){  return input.length() > lenght;  }      public static boolean isLenghtEqual(String input, int lenght){  return input.length() == lenght;  }  //định dạng email  public static boolean isValidEmail(String input){  return input.matches("^([\\w\\-\\.]+){2,64}\\@(\\w){2,255}(\\.[a-z]{2,10}){1,2}$");  }    public static boolean isValidDate(String date){  SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");  try {  sdf.format(date);  } catch (Exception e) {  return false;  }  return true;  }  } |

#### Lớp tiện ích ValueOfInQuerryBuilder

Lớp hỗ trợ truy vấn, định dạng một mảng chuỗi thành một chuỗi sử dụng cho câu lệnh query với in (select \* form ThanhVien Where Id in (?,?,? …..))

|  |
| --- |
| package com.org.app.util;  import java.util.Arrays;  import java.util.Collections;  import java.util.HashSet;  import java.util.Set;  public class ValueOfInQuerryBuilder {    public static String getValue(Object[] arr) {  String[] str = Arrays.copyOf(arr, arr.length,String[].class);  StringBuilder values = new StringBuilder();  for(String s : str) {  String i = "'" + s + "'";  values.append(i).append(",");  }    values = values.deleteCharAt(values.lastIndexOf(","));  System.out.println(values.toString());  return values.toString();  }    public static String any(String sql, int num) {  final StringBuilder sb = new StringBuilder(  String.join(", ", Collections.nCopies(num, "?")));  System.out.println(sb.toString());  if (sb.length() > 1) {  sql = sql.replace("(?)", "(" + sb + ")");  }  return sql;  }    public static void main(String[] args) {  String[] array = {"a", "b", "c","c"};  java.util.Set<String> set = new java.util.HashSet<>(Arrays.asList(array));  }  } |

#### Lớp tiện ích GradientPanel

Hỗ trợ đỗ màu Gradient cho panel

|  |
| --- |
| package com.org.app.weblaf;  import java.awt.Color;  import java.awt.Component;  import java.awt.GradientPaint;  import java.awt.Graphics;  import java.awt.Graphics2D;  import java.awt.RenderingHints;  import java.awt.event.MouseMotionListener;  import javax.swing.JCheckBox;  import javax.swing.JFormattedTextField;  import javax.swing.JLabel;  import javax.swing.JPanel;  import javax.swing.JPasswordField;  import javax.swing.JRadioButton;  import javax.swing.JTextField;  import javax.swing.JToolBar;  public class GradientPanel extends JPanel {  public Color kStartColor = Color.decode("#488FB1");  public Color kEndColor = Color.decode("#4FD3C4");  public boolean kTransparentControls = true;  public int kGradientFocus = 200;  public Color getStartColor() {  return kStartColor;  }  public void setkStartColor(Color kStartColor) {  this.kStartColor = kStartColor;  }  public Color getEndColor() {  return kEndColor;  }  public void setkEndColor(Color kEndColor) {  this.kEndColor = kEndColor;  }  public boolean isTransparentControls() {  return kTransparentControls;  }  public void setkTransparentControls(boolean kTransparentControls) {  this.kTransparentControls = kTransparentControls;  }  public int getGradientFocus() {  return kGradientFocus;  }  public void setkGradientFocus(int kGradientFocus) {  this.kGradientFocus = kGradientFocus;  }    public GradientPanel() {  if (kTransparentControls) {  setBg(true);  } else {  setBg(false);  }  }    public GradientPanel(String startColor, String focusColor, int range) {  setkStartColor(Color.decode(startColor));  setkEndColor(Color.decode(focusColor));  setkGradientFocus(range);  }  @Override  public synchronized void addMouseMotionListener(MouseMotionListener l) {  super.addMouseMotionListener(l); //To change body of generated methods, choose Tools | Templates.  }  @Override  protected void paintComponent(Graphics g) {  super.paintComponent(g);  Graphics2D g2d = (Graphics2D) g;  g2d.setRenderingHint(RenderingHints.KEY\_RENDERING, RenderingHints.VALUE\_RENDER\_QUALITY);  int w = getWidth();  int h = getHeight();  GradientPaint gp = new GradientPaint(0, 0, kStartColor, kGradientFocus, h, kEndColor);;  g2d.setPaint(gp);  g2d.fillRect(0, 0, w, h);  }  private void setBg(boolean isOpaque) {  Component[] components = this.getComponents();  for (Component component : components) {  ((JLabel) component).setOpaque(isOpaque);  ((JCheckBox) component).setOpaque(isOpaque);  ((JTextField) component).setOpaque(isOpaque);  ((JPasswordField) component).setOpaque(isOpaque);  ((JFormattedTextField) component).setOpaque(isOpaque);  ((JToolBar) component).setOpaque(isOpaque);  ((JRadioButton) component).setOpaque(isOpaque);  }  }  } |

### Model class - Các lớp mô tả dữ liệu

Để tránh các sai sót về kiểu dữ liệu khi làm việc với CSDL, chúng ta cần phải xây dựng các lớp mô tả dữ liệu với kiểu phù hợp. Điều này giúp làm việc với CSDL chặt chẽ hơn, dễ hơn.

#### Đồ ăn

|  |
| --- |
| package com.org.app.entity;  public class DoAn {  private String id;  private String ten;  private LoaiDoAn loaiDoAn;  public DoAn(String id, String ten, LoaiDoAn loaiDoAn) {  this(ten,loaiDoAn);  this.id = id;  }  public DoAn(String ten, LoaiDoAn loaiDoAn) {  this.ten = ten;  this.loaiDoAn = loaiDoAn;  }  public String getId() {  return id;  }  public void setId(String id) {  this.id = id;  }  public String getTen() {  return ten;  }  public void setTen(String ten) {  this.ten = ten;  }  public LoaiDoAn getLoaiDoAn() {  return loaiDoAn;  }  public void setLoaiDoAn(LoaiDoAn loaiDoAn) {  this.loaiDoAn = loaiDoAn;  }    } |

#### Đồ ăn chi tiết

|  |
| --- |
| package com.org.app.entity;  public class DoAnChiTiet {  private Integer id;  private DonDoAn donDA;  private int soLuong;  private KichCoDoAn doAn;  public DoAnChiTiet(Integer id, DonDoAn donDA, int soLuong, KichCoDoAn doAn) {  this.id = id;  this.donDA = donDA;  this.soLuong = soLuong;  this.doAn = doAn;  }  public Integer getId() {  return id;  }  public void setId(Integer id) {  this.id = id;  }    public DonDoAn getDonDA() {  return donDA;  }  public void setDonDA(DonDoAn donDA) {  this.donDA = donDA;  }  public int getSoLuong() {  return soLuong;  }  public void setSoLuong(int soLuong) {  this.soLuong = soLuong;  }  public KichCoDoAn getDoAn() {  return doAn;  }  public void setDoAn(KichCoDoAn doAn) {  this.doAn = doAn;  }  } |

#### Đơn đồ ăn

|  |
| --- |
| package com.org.app.entity;  import java.util.HashSet;  public class DonDoAn {  private String id;  private Integer soluong;  private Double tamTinh;  private HashSet<DoAnChiTiet> doAns;  public DonDoAn(String id, Integer soluong, Double tamTinh, HashSet<DoAnChiTiet> doAns) {  this.id = id;  this.soluong = soluong;  this.tamTinh = tamTinh;  this.doAns = doAns;  }  public DonDoAn(String id, Integer soluong, Double tamTinh) {  this.id = id;  this.soluong = soluong;  this.tamTinh = tamTinh;  doAns = new HashSet<>();  }  public String getId() {  return id;  }  public void setId(String id) {  this.id = id;  }  public Integer getSoluong() {  return soluong;  }  public void setSoluong(Integer soluong) {  this.soluong = soluong;  }  public Double getTamTinh() {  return tamTinh;  }  public void setTamTinh(Double tamTinh) {  this.tamTinh = tamTinh;  }    } |

#### Đơn thanh toán

|  |
| --- |
| package com.org.app.entity;  import java.sql.Date;  public class DonThanhToan {  private Integer id;  private DonDoAn donDoAn;  private DonVe donVe;  private Double tong;  private String nv;  private ThanhVien tv;  private Date ngayDat;  public DonThanhToan(Integer id, DonDoAn donDoAn, DonVe donVe, Double tong, String nv, ThanhVien tv, Date ngayDat) {  this.id = id;  this.donDoAn = donDoAn;  this.donVe = donVe;  this.tong = tong;  this.nv = nv;  this.tv = tv;  this.ngayDat = ngayDat;  }  public DonThanhToan(DonDoAn donDoAn, DonVe donVe, Double tong, String nv, ThanhVien tv, Date ngayDat) {  this.donDoAn = donDoAn;  this.donVe = donVe;  this.tong = tong;  this.nv = nv;  this.tv = tv;  this.ngayDat = ngayDat;  }  public Integer getId() {  return id;  }  public void setId(Integer id) {  this.id = id;  }    public DonDoAn getDonDoAn() {  return donDoAn;  }  public void setDonDoAn(DonDoAn donDoAn) {  this.donDoAn = donDoAn;  }  public DonVe getDonVe() {  return donVe;  }  public void setDonVe(DonVe donVe) {  this.donVe = donVe;  }  public Double getTong() {  return tong;  }  public void setTong(Double tong) {  this.tong = tong;  }  public String getNv() {  return nv;  }  public void setNv(String nv) {  this.nv = nv;  }  public ThanhVien getTv() {  return tv;  }  public void setTv(ThanhVien tv) {  this.tv = tv;  }  public Date getNgayDat() {  return ngayDat;  }  public void setNgayDat(Date ngayDat) {  this.ngayDat = ngayDat;  }  } |

#### Đơn vé

|  |
| --- |
| package com.org.app.entity;  import java.util.HashSet;  public class DonVe {  private String id;  private Integer tongGhe;  private Double tamTinh;  private HashSet<VeDat> veDats;  public DonVe(String id, Integer tongGhe, Double tamTinh) {  this.id = id;  this.tongGhe = tongGhe;  this.tamTinh = tamTinh;  veDats = new HashSet<>();  }  public DonVe(String id, Integer tongGhe, Double tamTinh, HashSet<VeDat> veDats) {  this.id = id;  this.tongGhe = tongGhe;  this.tamTinh = tamTinh;  this.veDats = veDats;  }  public String getId() {  return id;  }  public void setId(String id) {  this.id = id;  }  public Integer getTongGhe() {  return tongGhe;  }  public void setTongGhe(Integer tongGhe) {  this.tongGhe = tongGhe;  }  public Double getTamTinh() {  return tamTinh;  }  public void setTamTinh(Double tamTinh) {  this.tamTinh = tamTinh;  }  public HashSet<VeDat> getVeDats() {  return veDats;  }  public void setVeDats(HashSet<VeDat> veDats) {  this.veDats = veDats;  }    } |

#### Ghế

|  |
| --- |
| package com.org.app.entity;  public class Ghe {  private String id;  private String viTriDay;  private String viTriCot;  private PhongChieu phong;  private LoaiGhe loaiGhe;  public Ghe(String id, String viTriDay, String viTriCot, PhongChieu phong, LoaiGhe loaiGhe) {  this.id = id;  this.viTriDay = viTriDay;  this.viTriCot = viTriCot;  this.phong = phong;  this.loaiGhe = loaiGhe;  }  public Ghe() {  }  public String getId() {  return id;  }  public void setId(String id) {  this.id = id;  }  public String getViTriDay() {  return viTriDay;  }  public void setViTriDay(String viTriDay) {  this.viTriDay = viTriDay;  }  public String getViTriCot() {  return viTriCot;  }  public void setViTriCot(String viTriCot) {  this.viTriCot = viTriCot;  }  public PhongChieu getPhong() {  return phong;  }  public void setPhong(PhongChieu phong) {  this.phong = phong;  }  public LoaiGhe getLoaiGhe() {  return loaiGhe;  }  public void setLoaiGhe(LoaiGhe loaiGhe) {  this.loaiGhe = loaiGhe;  }    } |

#### Kích cỡ

|  |
| --- |
| package com.org.app.entity;  public class KichCo {  String id;  public KichCo(String id) {  this.id = id;  }  public String getId() {  return id;  }  public void setId(String id) {  this.id = id;  }    } |

#### Kích cỡ đồ ăn

|  |
| --- |
| package com.org.app.entity;  import java.util.Objects;  public class KichCoDoAn {  Integer id;  DoAn doAn;  KichCo kichco;  Double gia;  public KichCoDoAn(Integer id, KichCo kichco, Double gia) {  this.id = id;  this.kichco = kichco;  this.gia = gia;  }    public KichCoDoAn(Integer id, DoAn doAn, KichCo kichco, Double gia) {  this.id = id;  this.doAn = doAn;  this.kichco = kichco;  this.gia = gia;  }  public Integer getId() {  return id;  }  public void setId(Integer id) {  this.id = id;  }  public DoAn getDoAn() {  return doAn;  }  public void setDoAn(DoAn doAn) {  this.doAn = doAn;  }  public KichCo getKichco() {  return kichco;  }  public void setKichco(KichCo kichco) {  this.kichco = kichco;  }  public Double getGia() {  return gia;  }  public void setGia(Double gia) {  this.gia = gia;  }  @Override  public int hashCode() {  int hash = 3;  return hash;  }  @Override  public boolean equals(Object obj) {  if (this == obj) {  return true;  }  if (obj == null) {  return false;  }  if (getClass() != obj.getClass()) {  return false;  }  final KichCoDoAn other = (KichCoDoAn) obj;  if (!Objects.equals(this.id, other.id)) {  return false;  }  return true;  }    } |

#### Loại đồ ăn

|  |
| --- |
| package com.org.app.entity;  public class LoaiDoAn {  private String id;  private String ten;  public LoaiDoAn(String id, String ten) {  this.id = id;  this.ten = ten;  }  public String getId() {  return id;  }  public void setId(String id) {  this.id = id;  }  public String getTen() {  return ten;  }  public void setTen(String ten) {  this.ten = ten;  }    } |

#### Loại ghế

|  |
| --- |
| package com.org.app.entity;  public class LoaiGhe {  private String id;  private double gia;  private String tenLoai;  public LoaiGhe(String id, double gia, String tenLoai) {  this.id = id;  this.gia = gia;  this.tenLoai = tenLoai;  }    } |

#### Loại vé

|  |
| --- |
| package com.org.app.entity;  public class LoaiVe {  private String id;  private String ten;  private Double gia;  public LoaiVe(String id, String ten, Double gia) {  this.id = id;  this.ten = ten;  this.gia = gia;  }  public String getId() {  return id;  }  public String getTen() {  return ten;  }  public Double getGia() {  return gia;  }  } |

#### Nhân viên

|  |
| --- |
| package com.org.app.entity;  import java.sql.Date;  public class NhanVien {  private String id;  private String hoten;  private Date ngaysinh;  private boolean gioitinh;  private String sodt;  private String email;  private boolean vaitro;  private String anh;  private String tentk;  private String matkhau;  public NhanVien(String id) {  this.id = id;  }  public NhanVien() {  }  public NhanVien(String hoten, Date ngaysinh, boolean gioitinh, String sodt, String email, boolean vaitro, String anh,String tentk, String matkhau) {  this.hoten = hoten;  this.ngaysinh = ngaysinh;  this.gioitinh = gioitinh;  this.sodt = sodt;  this.email = email;  this.vaitro = vaitro;  this.anh = anh;  this.tentk = tentk;  this.matkhau = matkhau;  }    public NhanVien(String id, String hoten, Date ngaysinh, boolean gioitinh, String sodt, String email, boolean vaitro,String anh, String tentk, String matkhau) {  this(hoten,ngaysinh,gioitinh,sodt,email,vaitro,anh,tentk,matkhau);  this.id = id;  }  public String getId() {  return id;  }  public void setId(String id) {  this.id = id;  }  public String getHoten() {  return hoten;  }  public void setHoten(String hoten) {  this.hoten = hoten;  }  public Date getNgaysinh() {  return ngaysinh;  }  public void setNgaysinh(Date ngaysinh) {  this.ngaysinh = ngaysinh;  }  public boolean getGioitinh() {  return gioitinh;  }  public void setGioitinh(boolean gioitinh) {  this.gioitinh = gioitinh;  }  public String getSodt() {  return sodt;  }  public void setSodt(String sodt) {  this.sodt = sodt;  }  public String getEmail() {  return email;  }  public void setEmail(String email) {  this.email = email;  }  public boolean getVaitro() {  return vaitro;  }  public void setVaitro(boolean vaitro) {  this.vaitro = vaitro;  }  public String getAnh() {  return anh;  }  public void setAnh(String anh) {  this.anh = anh;  }  public String getTentk() {  return tentk;  }  public void setTentk(String tentk) {  this.tentk = tentk;  }  public String getMatkhau() {  return matkhau;  }  public void setMatkhau(String matkhau) {  this.matkhau = matkhau;  }  public boolean getGioiTinh() {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }    } |

#### Phim

|  |
| --- |
| package com.org.app.entity;  import java.sql.Date;  public class Phim {  private String id;  private String ten;  private String poster;  private Integer thoiLuong;  private String theloai;  private String ngonNgu;  private Date ngayKhoiChieu;  private String tomTat;  private boolean trangThai;  private String nv;  public Phim(String id, String ten, String poster, Integer thoiLuong, String theloai, String ngonNgu, Date ngayKhoiChieu, String tomTat, boolean trangThai, String nv) {  this(ten,poster,thoiLuong,theloai,ngonNgu,ngayKhoiChieu,tomTat,trangThai,nv);  this.id = id;  }  public Phim(String ten, String poster, Integer thoiLuong, String theloai, String ngonNgu, Date ngayKhoiChieu, String tomTat, boolean trangThai, String nv) {  this.ten = ten;  this.poster = poster;  this.thoiLuong = thoiLuong;  this.theloai = theloai;  this.ngonNgu = ngonNgu;  this.ngayKhoiChieu = ngayKhoiChieu;  this.tomTat = tomTat;  this.trangThai = trangThai;  this.nv = nv;  }  public String getId() {  return id;  }  public void setId(String id) {  this.id = id;  }  public String getTen() {  return ten;  }  public void setTen(String ten) {  this.ten = ten;  }    public String getPoster() {  return poster;  }  public void setPoster(String poster) {  this.poster = poster;  }  public Integer getThoiLuong() {  return thoiLuong;  }  public void setThoiLuong(Integer thoiLuong) {  this.thoiLuong = thoiLuong;  }  public String getTheloai() {  return theloai;  }  public void setTheloai(String theloai) {  this.theloai = theloai;  }  public String getNgonNgu() {  return ngonNgu;  }  public void setNgonNgu(String ngonNgu) {  this.ngonNgu = ngonNgu;  }  public Date getNgayKhoiChieu() {  return ngayKhoiChieu;  }  public void setNgayKhoiChieu(Date ngayKhoiChieu) {  this.ngayKhoiChieu = ngayKhoiChieu;  }  public String getTomTat() {  return tomTat;  }  public void setTomTat(String tomTat) {  this.tomTat = tomTat;  }  public boolean getTrangThai() {  return trangThai;  }  public void setTrangThai(boolean trangThai) {  this.trangThai = trangThai;  }  public String getNv() {  return nv;  }  public void setNv(String nv) {  this.nv = nv;  }  @Override  public String toString() {  return ten;  }    public String toTen() {  return ten;  }  } |

#### Phòng chiếu

|  |
| --- |
| package com.org.app.entity;  import java.util.HashSet;  import java.util.Set;  public class PhongChieu {  private String id;  private int soLuongDay;  private int soLuongCot;  private Set<SuatChieu> suatChieus = new HashSet<SuatChieu>(0);  private Set<Ghe> gheNgois = new HashSet<Ghe>(0);  public PhongChieu(String id, int soLuongDay, int soLuongCot) {  this.id = id;  this.soLuongDay = soLuongDay;  this.soLuongCot = soLuongCot;  }  public PhongChieu() {    }  public String getId() {  return id;  }  public void setId(String id) {  this.id = id;  }  public int getSoLuongDay() {  return soLuongDay;  }  public void setSoLuongDay(int soLuongDay) {  this.soLuongDay = soLuongDay;  }  public int getSoLuongCot() {  return soLuongCot;  }  public void setSoLuongCot(int soLuongCot) {  this.soLuongCot = soLuongCot;  }  public Set<SuatChieu> getSuatChieus() {  return suatChieus;  }  public void setSuatChieus(Set<SuatChieu> suatChieus) {  this.suatChieus = suatChieus;  }  public Set<Ghe> getGheNgois() {  return gheNgois;  }  public void setGheNgois(Set<Ghe> gheNgois) {  this.gheNgois = gheNgois;  }  @Override  public String toString() {  return Integer.valueOf(id.substring(1,id.length())) + "";  }      } |

#### Suất chiếu

|  |
| --- |
| package com.org.app.entity;  import com.org.app.helper.DateHelper;  import com.org.app.helper.TimeHelper;  import java.sql.Date;  import java.sql.Time;  public class SuatChieu {  private String id;  private Date ngayChieu;  private String phong;  private String phim;  private Time gioBatDau;  private Time gioKetThuc;  private String nv;  public SuatChieu(String id, Date ngayChieu, String phong, String phim, Time gioBatDau, Time gioKetThuc, String nv) {  this(ngayChieu, phong,phim,gioBatDau,gioKetThuc,nv);  this.id = id;  }  public SuatChieu(Date ngayChieu, String phong, String phim, Time gioBatDau, Time gioKetThuc, String nv) {  this.ngayChieu = ngayChieu;  this.phong = phong;  this.phim = phim;  this.gioBatDau = gioBatDau;  this.gioKetThuc = gioKetThuc;  this.nv = nv;  }  public SuatChieu() {  }  public String getId() {  return id;  }  public void setId(String id) {  this.id = id;  }  public Date getNgayChieu() {  return ngayChieu;  }  public void setNgayChieu(Date ngayChieu) {  this.ngayChieu = ngayChieu;  }  public String getPhong() {  return phong;  }  public void setPhong(String phong) {  this.phong = phong;  }  public String getPhim() {  return phim;  }  public void setPhim(String phim) {  this.phim = phim;  }  public Time getGioBatDau() {  return gioBatDau;  }  public void setGioBatDau(Time gioBatDau) {  this.gioBatDau = gioBatDau;  }  public Time getGioKetThuc() {  return gioKetThuc;  }  public void setGioKetThuc(Time gioKetThuc) {  this.gioKetThuc = gioKetThuc;  }  public String getNv() {  return nv;  }  public void setNv(String nv) {  this.nv = nv;  }  @Override  public String toString() {  }    } |

#### Thành viên

|  |
| --- |
| package com.org.app.entity;  import java.util.Date;  public class ThanhVien {  private String ID;  private String hoTen;  private Date ngaySinh;  private boolean gioiTinh;  private String soDT;  private String email;  public ThanhVien() {  }    public ThanhVien(String hoTen, Date ngaySinh, boolean gioiTinh, String soDT, String email) {  this.hoTen = hoTen;  this.ngaySinh = ngaySinh;  this.gioiTinh = gioiTinh;  this.soDT = soDT;  this.email = email;  }    public ThanhVien(String ID, String hoTen, Date ngaySinh, boolean gioiTinh, String soDT, String email) {  this(hoTen,ngaySinh,gioiTinh,soDT,email);  this.ID = ID;  }  public String getID() {  return ID;  }  public void setID(String ID) {  this.ID = ID;  }  public String getHoTen() {  return hoTen;  }  public void setHoTen(String hoTen) {  this.hoTen = hoTen;  }  public Date getNgaySinh() {  return ngaySinh;  }  public void setNgaySinh(Date ngaySinh) {  this.ngaySinh = ngaySinh;  }  public boolean getGioiTinh() {  return gioiTinh;  }  public void setGioiTinh(boolean gioiTinh) {  this.gioiTinh = gioiTinh;  }  public String getSoDT() {  return soDT;  }  public void setSoDT(String soDT) {  this.soDT = soDT;  }  public String getEmail() {  return email;  }  public void setEmail(String email) {  this.email = email;  }    } |

#### Vé đặt

|  |
| --- |
| package com.org.app.entity;  public class VeDat {  private Integer id;  private LoaiVe loaiVe;  private Ghe ghe;  private DonVe donVe;  private SuatChieu suatChieu;  private Double gia;  public VeDat(Integer id, LoaiVe loaiVe, Ghe ghe, DonVe donVe, SuatChieu suatChieu, Double gia) {  this(loaiVe,ghe,donVe,suatChieu,gia);  this.id = id;    }    public VeDat(LoaiVe loaiVe, Ghe ghe, DonVe donVe, SuatChieu suatChieu, Double gia) {  this.loaiVe = loaiVe;  this.ghe = ghe;  this.donVe = donVe;  this.suatChieu = suatChieu;  this.gia = gia;  }  public VeDat() {    }    public Integer getId() {  return id;  }  public void setId(Integer id) {  this.id = id;  }  public LoaiVe getLoaiVe() {  return loaiVe;  }  public void setLoaiVe(LoaiVe loaiVe) {  this.loaiVe = loaiVe;  }  public Ghe getGhe() {  return ghe;  }  public void setGhe(Ghe ghe) {  this.ghe = ghe;  }  public DonVe getDonVe() {  return donVe;  }  public void setDonVe(DonVe donVe) {  this.donVe = donVe;  }  public SuatChieu getSuatChieu() {  return suatChieu;  }  public void setSuatChieu(SuatChieu suatChieu) {  this.suatChieu = suatChieu;  }  public Double getGia() {  return gia;  }  public void setGia(Double gia) {  this.gia = gia;  }    } |

### DAO Class - Các lớp truy xuất dữ liệu

DAO (Data Access Object) là các lớp giúp giúp làm việc với các bảng trong CSDL. Mỗi lớp được định nghĩa một cách đầy đủ nhất về các thao tác và truy vấn dữ liệu sẽ được sử dụng trong ứng dụng. DAO sau đó sẽ được sử dụng trong các điều khiển sự kiện của giao diện để làm việc với CSDL một cách thuận tiện.

#### Lớp DAO làm việc với bảng đồ ăn chi tiết

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.DoAnChiTiet;  import com.org.app.entity.DonDoAn;  import com.org.app.entity.KichCo;  import com.org.app.entity.KichCoDoAn;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.util.ArrayList;  import java.util.List;  public class DoAnChiTietDao extends Dao<Integer, DoAnChiTiet>{  String INSERT\_SQL = "INSERT INTO DOANCT (ID\_KCDA, ID\_DONDA, SOLUONG) VALUES (?, ?, ?)";  String UPDATE\_SQL = "UPDATE DOANCT SET ID\_KCDA = ?, ID\_DONDA = ?, SOLUONG = ? WHERE ID\_DOANCT = ?";  String DELETE\_SQL = "DELETE FROM DOANCT WHERE ID\_DOANCT = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM DOANCT";  String SELECT\_BY\_ID\_SQL = "SELECT DCT.\*, DDA.SOLUONG, DDA.TAMTINH, KCDA.ID\_KICHCO, KCDA.GIA FROM DOANCT DCT INNER JOIN DONDOAN "  + "DDA ON DCT.ID\_DONDA = DDA.ID\_DONDA" +  "INNER JOIN KICHCODOAN KCDA ON KCDA.ID\_KCDA = DCT.ID\_KCDA WHERE ID\_DOANCT = ?";    @Override  public void insert(DoAnChiTiet entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getDoAn(), entity.getDonDA(), entity.getSoLuong());  }  @Override  public void update(DoAnChiTiet entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE\_SQL, entity.getDoAn(), entity.getDonDA(), entity.getSoLuong(), entity.getId());  }  @Override  public void delete(Integer value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  @Override  public DoAnChiTiet selectById(Integer value) throws Exception {  List<DoAnChiTiet> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<DoAnChiTiet> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public List<DoAnChiTiet> selectBySql(String sql, Object... args) throws Exception {  List<DoAnChiTiet> list = new ArrayList<DoAnChiTiet>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  list.add(objectFromRs(rs));  return list;  } catch (Exception e) {  e.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  @Override  public DoAnChiTiet objectFromRs(ResultSet rs) throws Exception {  return null;  }    public DonDoAn createDonDoAn(String id\_DonDA, int soLuong, double tamTinh){  return new DonDoAn(id\_DonDA, soLuong, tamTinh);  }    public KichCoDoAn createKichCoDoAn(int id\_KichCoDoAn, String id\_KichCo, double gia){  return new KichCoDoAn(id\_KichCoDoAn, createKichCo(id\_KichCo), gia);  }    public KichCo createKichCo(String id){  return new KichCo(id);  }  } |

#### Lớp DAO làm việc với bảng đồ ăn

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.DoAn;  import com.org.app.entity.LoaiDoAn;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  public class DoAnDao extends Dao<String, DoAn>{  String INSERT\_SQL = "INSERT INTO DOAN (TEN, ID\_LOAI) VALUES (?, ?)";  String UPDATE\_SQL = "UPDATE DOAN SET TEN = ?, ID\_LOAI = ? WHERE ID\_DOAN = ?";  String DELETE\_SQL = "DELETE FROM DOAN WHERE ID\_DOAN = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM DOAN";  String SELECT\_BY\_ID\_SQL = "SELECT DA.\*, LDA.TEN FROM DOAN DA INNER JOIN LOAIDOAN LDA ON DA.ID\_LOAI = LDA.ID WHERE ID\_DOAN = ?";  String SELECT\_TENDA = "SELECT TEN FROM DOAN GROUP BY TEN";  String SELECT\_ID\_DOAN = "SELECT ID\_DOAN FROM DOAN WHERE TEN LIKE ?";    @Override  public void insert(DoAn entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getTen(), entity.getLoaiDoAn());  }    @Override  public void update(DoAn entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getTen(), entity.getLoaiDoAn(), entity.getId());  }  @Override  public void delete(String value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  @Override  public DoAn selectById(String value) throws Exception {  List<DoAn> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<DoAn> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public List<DoAn> selectBySql(String sql, Object... args) throws Exception {  List<DoAn> list = new ArrayList<DoAn>();  ResultSet rs = JDBCHelper.getResultSetByQuery(sql, args);  try {  list.add(objectFromRs(rs));  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  @Override  public DoAn objectFromRs(ResultSet rs) throws Exception {  return new DoAn(  rs.getString("ID\_DOAN"),  rs.getString("TEN"),  createLoaiDoAn(rs.getString("ID\_LOAI"), rs.getString("TEN"))  );  }    public LoaiDoAn createLoaiDoAn(String id\_loai, String ten){  return new LoaiDoAn(id\_loai, ten);  }    public void insert\_SP(String tenDA, String id\_loaiDA, String id\_kichCo, double gia) throws Exception {  JDBCHelper.excecuteStatement("{CALL sp\_themDoAn(?,?,?,?)}", tenDA, id\_loaiDA, id\_kichCo, gia);  }    public void update\_SP(String tenDA, String id\_loaiDA, String id\_kichCo, double gia, String id\_DoAn, int id\_KCDA) throws Exception {  JDBCHelper.excecuteStatement("{CALL sp\_suaDoAn(?,?,?,?,?,?)}", tenDA, id\_loaiDA, id\_kichCo, gia, id\_DoAn, id\_KCDA);  }    public List<String> getTenDA() throws SQLException{  List<String> list = new ArrayList<>();  ResultSet rs = JDBCHelper.getResultSetByQuery(SELECT\_TENDA);  while (rs.next()) {  String tenDA = rs.getString("TEN");  list.add(tenDA);  }    return list;  }    public String getIdDoAn(String value){  try {  ResultSet rs = JDBCHelper.getResultSetByQuery(SELECT\_ID\_DOAN, value);  String id\_DA = null;  while (rs.next()) {  id\_DA = rs.getString("ID\_DOAN");  }  rs.getStatement().getConnection().close();  return id\_DA;  } catch (Exception e) {  throw new RuntimeException(e);  }  }  } |

#### Lớp DAO làm việc với bảng đơn đồ ăn

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.DonDoAn;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  public class DonDoAnDao extends Dao<String, DonDoAn>{  String INSERT\_SQL = "INSERT INTO DONDOAN (ID\_DONDA) VALUES (?)";  String UPDATE\_SQL = "UPDATE DONDOAN SET ID\_DONDA WHERE ID\_DONDA = ?";  String DELETE\_SQL = "DELETE FROM DONDOAN WHERE ID\_DONDA = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM DONDOAN";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM DONDOAN WHERE ID\_DONDA = ?";  String SELECT\_BY\_SL\_DU = "select count(ID\_LOAI) as SOLUONGDU from DONTHANHTOAN DTT join DONDOAN DDA on DTT.ID\_DONDA = DDA.ID\_DONDA " +  "join DOANCT DACT on DDA.ID\_DONDA = DACT.ID\_DONDA " +  "join KICHCODOAN KCDA on KCDA.ID\_KCDA = DACT.ID\_KCDA " +  "join DOAN DA on DA.ID\_DOAN = KCDA.ID\_DOAN " +  "where ID\_DONTT like ? and ID\_LOAI like 'DU'";  String SELECT\_BY\_SL\_TA = "select count(ID\_LOAI) as SOLUONGTA from DONTHANHTOAN DTT join DONDOAN DDA on DTT.ID\_DONDA = DDA.ID\_DONDA " +  "join DOANCT DACT on DDA.ID\_DONDA = DACT.ID\_DONDA " +  "join KICHCODOAN KCDA on KCDA.ID\_KCDA = DACT.ID\_KCDA " +  "join DOAN DA on DA.ID\_DOAN = KCDA.ID\_DOAN " +  "where ID\_DONTT like ? and ID\_LOAI like 'TA'";    @Override  public void insert(DonDoAn entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getId());  }  @Override  public void update(DonDoAn entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getId(), entity.getId());  }  @Override  public void delete(String value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  @Override  public DonDoAn selectById(String value) throws Exception {  List<DonDoAn> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<DonDoAn> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public List<DonDoAn> selectBySql(String sql, Object... args) throws SQLException {  List<DonDoAn> list = new ArrayList<DonDoAn>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  list.add(objectFromRs(rs));  return list;  } catch (Exception e) {  e.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  @Override  public DonDoAn objectFromRs(ResultSet rs) throws Exception {  return new DonDoAn(  rs.getString("ID\_DONDA"),  rs.getInt("SOLUONG"),  rs.getDouble("TAMTINH")  );  }    private List<Object[]> getListOfArray(String sql, String[] cols, Object...args) throws SQLException{  ResultSet rs = null;  try {  List<Object[]> list = new ArrayList<>();  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  Object[] vals = new Object[cols.length];  for (int i = 0; i < cols.length; i++) {  vals[i] = rs.getObject(cols[i]);  }  list.add(vals);  }  return list;  } catch (Exception e) {  e.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  } finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }    }    public List<Object[]> getThongTinDonDA(int id\_dontt) throws SQLException{  String sql = "{CALL sp\_donDoAn(?)}";  String[] cols = {"TEN", "SOLUONG", "ID\_KICHCO", "TONGTIEN"};  return this.getListOfArray(sql, cols, id\_dontt);  }    public int getSoLuongDUBySQL(Integer value) throws SQLException{  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(SELECT\_BY\_SL\_DU, value);  int soLuongDU = 0;  while (rs.next()) {  soLuongDU = rs.getInt("SOLUONGDU");  }  return soLuongDU;  } catch (SQLException ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }    public int getSoLuongTABySQL(Integer value) throws SQLException{  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(SELECT\_BY\_SL\_TA, value);  int soLuongTA = 0;  while (rs.next()) {  soLuongTA = rs.getInt("SOLUONGTA");  }  return soLuongTA;  } catch (Exception e) {  throw new RuntimeException(e);  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }    } |

#### Lớp DAO làm viẹce với bảng đơn thanh toán

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.DonDoAn;  import com.org.app.entity.DonThanhToan;  import com.org.app.entity.DonVe;  import com.org.app.entity.NhanVien;  import com.org.app.entity.ThanhVien;  import com.org.app.helper.DateHelper;  import com.org.app.helper.JDBCHelper;  import com.org.app.util.DinhDangTienTe;  import java.sql.ResultSet;  import java.util.ArrayList;  import java.sql.Date;  import java.sql.SQLException;  import java.time.LocalDate;  import java.util.List;  import java.util.Vector;  import java.util.logging.Level;  import java.util.logging.Logger;  public class DonThanhToanDao extends Dao<Integer, DonThanhToan>{  String INSERT\_SQL = "INSERT INTO DONTHANHTOAN (ID\_DONDA, ID\_TV, ID\_NV, ID\_DONVE, NGAYDAT, TONG) VALUES (?, ?, ?, ?, ?, ?)";  String UPDATE\_SQL = "UPDATE DONTHANHTOAN SET ID\_DONDA = ?, ID\_TV = ?, ID\_NV = ?, ID\_DONVE = ?, NGAYDAT = ?, TONG = ? WHERE ID\_DONTT = ?";  String DELETE\_SQL = "DELETE FROM DONTHANHTOAN WHERE ID\_DONTT = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM DONTHANHTOAN";  String SELECT\_BY\_ID\_SQL = "SELECT DTT.\*, DV.TONGGHE, DV.TAMTINH, TV.HOTEN, TV.NGAYSINH, TV.GIOITINH, TV.SODT, TV.EMAIL FROM DONTHANHTOAN DTT INNER JOIN DONVE DV ON DTT.ID\_DONVE = DV.ID\_DONVE INNER JOIN THANHVIEN TV ON TV.ID\_TV = DTT.ID\_TV WHERE ID\_DONTT = ?";  String ADD\_DONTT = "{CALL SP\_THEMDONTT (?,?,?)}";  String UPDATE\_TONGTIEN = "UPDATE DONTHANHTOAN SET TONG = ? WHERE ID\_DONTT = ?";  String UPDATE\_TONGTIEN\_NO\_GIAM = "{CALL SP\_UPDATETONGDON (?)}";  String GET\_DOANH\_THU = "select sum(tong) from DONTHANHTOAN where cast(NGAYDAT as date) = ?";  String SELECT\_YEAR = "select YEAR(NGAYDAT) as NAM from DONTHANHTOAN group by YEAR(NGAYDAT) order by NAM";  String SELECT\_DONTT\_ALL = "{CALL SP\_DONTHANHTOAN (?,?,?,?,?,?)}";  String SELECT\_DON\_SDT\_NAME = "select ID\_DONTT, NGAYDAT, TV.ID\_TV, TONG from DONTHANHTOAN DTT JOIN THANHVIEN TV ON DTT.ID\_TV = TV.ID\_TV where SODT like ? or HOTEN like ?";    @Override  public void insert(DonThanhToan entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getDonDoAn(), entity.getTv(), entity.getNv(),  entity.getDonVe(), entity.getNgayDat(), entity.getTong());  }  @Override  public void update(DonThanhToan entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getDonDoAn(), entity.getTv(), entity.getNv(),  entity.getDonVe(), entity.getNgayDat(), entity.getTong(), entity.getId());  }  public void delete(Integer value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  public DonThanhToan selectById(Integer value) throws Exception {  List<DonThanhToan> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<DonThanhToan> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }    @Override  public List<DonThanhToan> selectBySql(String sql, Object... args) throws Exception {  List<DonThanhToan> list = new ArrayList<DonThanhToan>();  ResultSet rs = JDBCHelper.getResultSetByQuery(sql, args);  try {  list.add(objectFromRs(rs));  rs.getStatement().close();  return list;  } catch (Exception e) {  throw new RuntimeException(e);  }  }  @Override  public DonThanhToan objectFromRs(ResultSet rs) throws Exception {  return null;  }    public DonDoAn createDonDoAn(String id\_DonDA, int soLuong, double tamTinh){  return new DonDoAn(id\_DonDA, soLuong, tamTinh);  }    public DonVe createDonVe(String id\_DonVe, int tongGhe, double tamTinh){  return new DonVe(id\_DonVe, tongGhe, tamTinh);  }    public ThanhVien createThanhVien(String id\_ThanhVien){  return new ThanhVien(id\_ThanhVien);  }    private List<Object[]> getListOfArray(String sql, String[] cols, Object...args) throws SQLException{  ResultSet rs = null;  try {  List<Object[]> list = new ArrayList<>();  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  Object[] vals = new Object[cols.length];  for (int i = 0; i < cols.length; i++) {  vals[i] = rs.getObject(cols[i]);  }  list.add(vals);  }  return list;  } catch (Exception e) {  throw new RuntimeException(e);  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }      //loai don 0 la ve, 1 la doan , con lai la ca 2  public DonThanhToan addDonThanhToan(int loaiDon, String tv, String nv) throws SQLException {  ResultSet rs = null;  try {  DonThanhToan don = new DonThanhToan(nv, tv);  rs = JDBCHelper.getResultSetByQuery(ADD\_DONTT, loaiDon, tv, nv);  //ID\_DTT, ID\_DA, ID\_VE  while(rs.next()) {  don.setId(rs.getInt(1));  don.setDonDoAn(rs.getString(2));  don.setDonVe(rs.getString(3));  }  return don;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }    public void updateTongTien(int id\_don, double tong) {  JDBCHelper.excecuteStatement(UPDATE\_TONGTIEN, tong, id\_don);  }    public void updateTongTien(int id\_don) {  JDBCHelper.excecuteStatement(UPDATE\_TONGTIEN\_NO\_GIAM, id\_don);  }      public double getDoanhThuNgay(LocalDate date) throws SQLException {  ResultSet rs = null;  try {  java.util.Date d = DateHelper.convertToDate(date);  String df = DateHelper.toString(d, DateHelper.YYYYMMDD\_FORMAT);  rs = JDBCHelper.getResultSetByQuery(GET\_DOANH\_THU, df);  Double dt = 0.0;  while (rs.next()){dt = rs.getDouble(1);};    return dt==null? 0.0 : dt;  } catch (SQLException ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  public List<Integer> getNamThanhToan() throws SQLException{    List<Integer> list = new ArrayList<>();  ResultSet rs = JDBCHelper.getResultSetByQuery(SELECT\_YEAR);  while (rs.next()) {  int nam = rs.getInt("NAM");  list.add(nam);  }    return list;  }    public List<Vector<Object>> selectByAll(Object...args) throws Exception {  ResultSet rs = JDBCHelper.getRSOfProc(SELECT\_DONTT\_ALL, new String[]{"NGAYBD","NGAYKT", "THANG", "NAM", "ISKHACHHANG", "KEYWORD"}, args);    List<Vector<Object>> list = new ArrayList<>();  while(rs.next()) {  Vector<Object> v = new Vector<>();  v.addAll(List.of(  rs.getInt("ID\_DONTT"),  rs.getDate("NGAYDAT"),  rs.getString("ID\_TV") == null ?"Tự do": rs.getString("ID\_TV"),  DinhDangTienTe.chuyenThanhTienVN(rs.getDouble("TONG")))  );  list.add(v);  }  return list;  }    } |

#### Lớp DAO làm việc với bảng đơn vé

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.DonVe;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  public class DonVeDao extends Dao<String, DonVe>{  String INSERT\_SQL = "INSERT INTO DONVE (ID\_DONVE) VALUES (?)";  String UPDATE\_SQL = "UPDATE DONVE SET ID\_DONVE = ? WHERE ID\_DONVE = ?";  String DELETE\_SQL = "DELETE FROM DONVE WHERE ID\_DONVE = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM DONVE";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM DONVE WHERE ID\_DONVE = ?";  @Override  public void insert(DonVe entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getId());  }  @Override  public void update(DonVe entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE\_SQL, entity.getId(), entity.getId());    }  @Override  public void delete(String value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  @Override  public DonVe selectById(String value) throws Exception {  List<DonVe> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<DonVe> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public List<DonVe> selectBySql(String sql, Object... args) throws Exception {  List<DonVe> list = new ArrayList<>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  list.add(objectFromRs(rs));  }  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  return list;  }  @Override  public DonVe objectFromRs(ResultSet rs) throws Exception {  return new DonVe(rs.getString("ID\_DONVE"),  rs.getInt("TONGGHE"),  rs.getDouble("TAMTINH")  );  }    private List<Object[]> getListOfArray(String sql, String[] cols, Object...args) throws Exception{  ResultSet rs = null;  try {  List<Object[]> list = new ArrayList<>();  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  Object[] vals = new Object[cols.length];  for (int i = 0; i < cols.length; i++) {  vals[i] = rs.getObject(cols[i]);  }  list.add(vals);  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }    public List<Object[]> getThongTinDonVe(int id\_dontt) throws Exception{  String sql = "{CALL sp\_donVe(?)}";  String[] cols = {"TEN", "SUATCHIEU", "ID\_LOAIVE", "GIA"};  return this.getListOfArray(sql, cols, id\_dontt);  }    } |

#### Lớp DAO làm việc với bảng ghế

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.Ghe;  import com.org.app.entity.LoaiGhe;  import com.org.app.entity.PhongChieu;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.util.ArrayList;  import java.util.LinkedHashMap;  import java.util.List;  import java.util.Map;  import java.util.function.Function;  import static java.util.stream.Collectors.toMap;  public class GheDao extends Dao<String, Ghe>{  String INSERT\_SQL = "INSERT INTO GHE(VITRIDAY, VITRICOT, ID\_PHONGCHIEU, ID\_LOAIGHE) VALUES (?, ?, ?, ?)";  String UPDATE\_SQL = "UPDATE GHE SET VITRIDAY = ?, VITRICOT = ?, ID\_PHONGCHIEU=?, ID\_LOAIGHE=? WHERE ID\_GHE = ?";  String DELETE\_SQL = "DELETE FROM GHE WHERE ID\_GHE = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM GHE G JOIN PHONGCHIEU ON G.ID\_PHONGCHIEU = PHONGCHIEU.ID\_PHONG " +  "JOIN LOAIGHE LG ON G.ID\_LOAIGHE = LG.ID\_LOAIGHE";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM GHE WHERE ID\_GHE = ?";  String SELECT\_GHEPHONG\_SQL ="SELECT \* FROM GHE JOIN PHONGCHIEU ON GHE.ID\_PHONGCHIEU = PHONGCHIEU.ID\_PHONG";    private String selectGheBySuat ="SELECT g.\* FROM GHE g JOIN SUATCHIEU s ON s.ID\_PHONG = g.ID\_PHONGCHIEU = ?";  private String selectByPhong = "SELECT \* FROM GHE G JOIN PHONGCHIEU ON G.ID\_PHONGCHIEU = PHONGCHIEU.ID\_PHONG " +  "JOIN LOAIGHE LG ON G.ID\_LOAIGHE = LG.ID\_LOAIGHE WHERE ID\_PHONGCHIEU = ?" ;    private String SELECTED\_BOOKED\_GHE = "SELECT G.\*, PC.\*, LG.\* FROM VEDAT " +  "LEFT JOIN GHE G ON VEDAT.ID\_GHE = G.ID\_GHE " +  "LEFT JOIN LOAIGHE LG ON LG.ID\_LOAIGHE = G.ID\_LOAIGHE " +  "LEFT JOIN SUATCHIEU SC ON VEDAT.ID\_SUAT = SC.ID\_SUAT " +  "LEFT JOIN PHONGCHIEU PC ON PC.ID\_PHONG = SC.ID\_PHONG WHERE SC.ID\_SUAT = ?";  @Override  public void insert(Ghe entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getViTriDay(), entity.getViTriCot(), entity.getPhong(), entity.getLoaiGhe());  }  @Override  public void update(Ghe entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE\_SQL, entity.getViTriDay(), entity.getViTriCot(), entity.getPhong(), entity.getLoaiGhe(), entity.getId());  }  @Override  public void delete(String value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  @Override  public Ghe selectById(String value) throws Exception {  List<Ghe> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<Ghe> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public List<Ghe> selectBySql(String sql, Object... args) throws Exception {  List<Ghe> list = new ArrayList<>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  list.add(objectFromRs(rs));  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  @Override  public Ghe objectFromRs(ResultSet rs) throws Exception {  return new Ghe(rs.getString("ID\_GHE"),  rs.getString("VITRIDAY"),  rs.getString("VITRICOT"),  createPhongChieu(rs.getString("ID\_PHONGCHIEU"), rs.getInt("SOLUONGDAY"), rs.getInt("SOLUONGCOT")),  createLoaiGhe(rs.getString("ID\_LOAIGHE"), rs.getDouble("PHUTHU"), rs.getString("TENLOAI"))  );  }    public PhongChieu createPhongChieu(String id, int slDay, int slCot){  return new PhongChieu(id, slDay, slCot);  }    public LoaiGhe createLoaiGhe(String id, double gia, String tenLoai){  return new LoaiGhe(id, gia, tenLoai);  }    public List<Ghe> selectGheOfPhong(String idPhong) throws Exception {  return selectBySql(selectByPhong, idPhong);  }    public Map<String, Ghe> getMapGheOfPhong(String idPhong) throws Exception {  List<Ghe> ghes = selectGheOfPhong(idPhong);  LinkedHashMap<String, Ghe> map = new LinkedHashMap<>();  for(Ghe p : ghes) {  map.put(p.getId(), p);  }  return map;  }    public List<Ghe> selectGheDaDat(String idSuat) throws Exception {  return selectBySql(SELECTED\_BOOKED\_GHE, idSuat);    }  } |

#### Lớp DAO làm việc với bảng kích cỡ

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.KichCo;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.util.ArrayList;  import java.util.List;  public class KichCoDao extends Dao<String, KichCo>{  String INSERT\_SQL = "INSERT INTO KICHCO (ID) VALUES (?)";  String UPDATE\_SQL = "UPDATE KICHCO SET ID =? WHERE ID=?";  String DELETE\_SQL = "DELETE FROM KICHCO WHERE ID = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM KICHCO";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM KICHCO WHERE ID = ?";  @Override  public void insert(KichCo entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getId());  }  @Override  public void update(KichCo entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE\_SQL, entity.getId(), entity.getId());  }  @Override  public void delete(String value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  @Override  public KichCo selectById(String value) throws Exception {  List<KichCo> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<KichCo> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public List<KichCo> selectBySql(String sql, Object... args) throws Exception {  List<KichCo> list = new ArrayList<>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  list.add(objectFromRs(rs));  }  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  return list;  }  @Override  public KichCo objectFromRs(ResultSet rs) throws Exception {  return new KichCo(rs.getString("ID"));  }    } |

#### Lớp DAO làm việc với bảng kích cỡ đồ ăn

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.DoAn;  import com.org.app.entity.KichCo;  import com.org.app.entity.KichCoDoAn;  import com.org.app.entity.LoaiDoAn;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  public class KichCoDoAnDao extends Dao<Integer, KichCoDoAn>{  String INSERT\_SQL = "INSERT INTO KICHCODOAN (ID\_DOAN, ID\_KICHCO, GIA) VALUES (?, ?, ?)";  String UPDATE\_SQL = "UPDATE KICHCODOAN SET ID\_DOAN = ?, ID\_KICHCO = ?, GIA=? WHERE ID\_KCDA = ?";  String DELETE\_SQL = "DELETE FROM KICHCODOAN WHERE ID\_KCDA = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM KICHCODOAN";  String SELECT\_BY\_ID\_SQL = "select ID\_KCDA , DA.ID\_DOAN, DA.TEN, ID\_LOAI, LDA.TEN, ID\_KICHCO, GIA " +  "from DOAN DA join KICHCODOAN KCDA on DA.ID\_DOAN = KCDA.ID\_DOAN join LOAIDOAN LDA on LDA.ID = DA.ID\_LOAI " +  "where ID\_KCDA = ?";  String SELECT\_GET\_ID\_SQL = "select ID\_KCDA , DA.ID\_DOAN, DA.TEN, ID\_LOAI, LDA.TEN, ID\_KICHCO, GIA " +  "from DOAN DA join KICHCODOAN KCDA on DA.ID\_DOAN = KCDA.ID\_DOAN join LOAIDOAN LDA on LDA.ID = DA.ID\_LOAI " +  "where DA.TEN like ? and ID\_KICHCO like ?";    String SELECT\_DOAN\_SQL = "SELECT ID\_KCDA, DOAN.ID\_DOAN, DOAN.TEN, DOAN.ID\_LOAI, LDA.TEN, ID\_KICHCO, GIA FROM KICHCODOAN KCDA JOIN DOAN DOAN ON KCDA.ID\_DOAN = DOAN.ID\_DOAN JOIN LOAIDOAN LDA ON DOAN.ID\_LOAI = LDA.ID";    String SELECT\_DOAN\_DU\_SQL = "SELECT ID\_KCDA, DOAN.ID\_DOAN, DOAN.TEN, DOAN.ID\_LOAI, LDA.TEN, ID\_KICHCO, GIA " +  "FROM KICHCODOAN KCDA JOIN DOAN DOAN ON KCDA.ID\_DOAN = DOAN.ID\_DOAN " +  " JOIN LOAIDOAN LDA ON DOAN.ID\_LOAI = LDA.ID " +  "WHERE ID\_LOAI like 'DU'";    String SELECT\_DOAN\_TA\_SQL = "SELECT ID\_KCDA, DOAN.ID\_DOAN, DOAN.TEN, DOAN.ID\_LOAI, LDA.TEN, ID\_KICHCO, GIA " +  "FROM KICHCODOAN KCDA JOIN DOAN DOAN ON KCDA.ID\_DOAN = DOAN.ID\_DOAN " +  " JOIN LOAIDOAN LDA ON DOAN.ID\_LOAI = LDA.ID " +  "WHERE ID\_LOAI like 'TA'";    String SELECT\_MENU = "select ID\_KCDA , DA.ID\_DOAN, DA.TEN, ID\_LOAI, LDA.TEN, ID\_KICHCO, GIA " +  "from DOAN DA join KICHCODOAN KCDA on DA.ID\_DOAN = KCDA.ID\_DOAN join LOAIDOAN LDA on LDA.ID = DA.ID\_LOAI " +  "where ID\_LOAI like ?";    @Override  public void insert(KichCoDoAn entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL,entity.getDoAn(), entity.getKichco(), entity.getGia());  }    @Override  public void update(KichCoDoAn entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE\_SQL,entity.getDoAn(), entity.getKichco(), entity.getGia(), entity.getId());  }    @Override  public void delete(Integer value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }    @Override  public KichCoDoAn selectById(Integer value) throws Exception {  List<KichCoDoAn> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }    public KichCoDoAn selectByTenDA\_KC(String tenDA, String kichCo) throws Exception {  List<KichCoDoAn> list = this.selectBySql(SELECT\_GET\_ID\_SQL, tenDA, kichCo);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<KichCoDoAn> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }    public List<KichCoDoAn> selectDoAnAll() throws Exception {  return this.selectBySql(SELECT\_DOAN\_SQL);  }    public List<KichCoDoAn> selectMenu(String loaiDA) throws Exception {  return this.selectBySql(SELECT\_MENU, loaiDA);  }    public List<KichCoDoAn> selectDoAn\_DU() throws Exception {  return this.selectBySql(SELECT\_DOAN\_DU\_SQL);  }    public List<KichCoDoAn> selectDoAn\_TA() throws Exception {  return this.selectBySql(SELECT\_DOAN\_TA\_SQL);  }    @Override  public List<KichCoDoAn> selectBySql(String sql, Object... args) throws SQLException, Exception {  List<KichCoDoAn> list = new ArrayList<>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  list.add(objectFromRs(rs));  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  @Override  public KichCoDoAn objectFromRs(ResultSet rs) throws Exception {  return new KichCoDoAn(  rs.getInt("ID\_KCDA"),  createDoAn(rs.getString("ID\_DOAN"), rs.getString("TEN"),  new LoaiDoAn(rs.getString("ID\_LOAI"), rs.getString("TEN"))),  new KichCo(rs.getString("ID\_KICHCO")),  rs.getDouble("GIA")  );  }  public DoAn createDoAn(String idDoAn, String ten, LoaiDoAn loaiDoAn){  return new DoAn(idDoAn, ten, loaiDoAn);  }    } |

#### Lớp DAO làm việc với bảng loại ghế

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.Ghe;  import com.org.app.entity.LoaiGhe;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  public class LoaiGheDao extends Dao<String, LoaiGhe>{  private String selectAll = "SELECT \* FROM LOAIGHE";  private String selecById = "SELECT \* FROM LOAIGHE WHERE ID\_LOAIGHE = ?";  private String SELECT\_SOGHEVP\_SQL = "select count(\*) as SOGHEVIP\n" +  "from DONTHANHTOAN DTT join DONVE DV on DTT.ID\_DONVE = DV.ID\_DONVE " +  " join VEDAT VD on VD.ID\_DONVE = DV.ID\_DONVE " +  " join GHE G on VD.ID\_GHE = G.ID\_GHE " +  " join LOAIGHE LG on LG.ID\_LOAIGHE = G.ID\_LOAIGHE " +  "where ID\_DONTT = ? and G.ID\_LOAIGHE like 'VP'";  private String INSERT = "";  private String UPDATE = "";  private String DELETE = "";    @Override  public void insert(LoaiGhe entity) throws Exception {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  @Override  public void update(LoaiGhe entity) throws Exception {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  @Override  public void delete(String value) throws Exception {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }  @Override  public LoaiGhe selectById(String value) throws Exception {  List<LoaiGhe> l = selectBySql(selecById, value);  return l.size() == 0? null : l.get(0);  }  @Override  public List<LoaiGhe> selectAll() throws Exception {  return selectBySql(selectAll);  }  @Override  public List<LoaiGhe> selectBySql(String sql, Object... args) throws Exception {  List<LoaiGhe> list = new ArrayList<>();  ResultSet rs = null;    try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while(rs.next()) {  list.add(objectFromRs(rs));  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }    }  }  }  @Override  public LoaiGhe objectFromRs(ResultSet rs) throws Exception {  return new LoaiGhe(  rs.getString("ID\_LOAIGHE"),  Integer.parseInt(rs.getDouble("PHUTHU")+""),  rs.getString("TENLOAI")  );    }    public int getSoGheVPBySQL(Integer value) throws RuntimeException, SQLException{  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(SELECT\_SOGHEVP\_SQL, value);  int soGheVP = 0;  while (rs.next()) {  soGheVP = rs.getInt("SOGHEVIP");  }  return soGheVP;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }    }  }  }  } |

#### Lớp DAO làm việc với bảng loại vé

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.LoaiVe;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  import java.util.Map;  import java.util.function.Function;  import static java.util.stream.Collectors.toMap;  public class LoaiVeDao extends Dao<String, LoaiVe> {  private String selectAll = "SELECT \* FROM LOAIVE";  private String selecById = "SELECT \* FROM LOAIVE WHERE ID\_LOAIVE = ?";  private String INSERT = "";  private String UPDATE = "";  private String DELETE = "";    @Override  public void insert(LoaiVe entity) throws Exception {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }    @Override  public void update(LoaiVe entity) throws Exception {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }    @Override  public void delete(String value) throws Exception {  throw new UnsupportedOperationException("Not supported yet."); //To change body of generated methods, choose Tools | Templates.  }    @Override  public LoaiVe selectById(String value) throws Exception {  List<LoaiVe> l = selectBySql(selecById, value);  return l.size() == 0? null : l.get(0);  }    @Override  public List<LoaiVe> selectAll() throws Exception {  return selectBySql(selectAll);  }    @Override  public List<LoaiVe> selectBySql(String sql, Object... args) throws Exception {  List<LoaiVe> list = new ArrayList<>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  list.add(objectFromRs(rs));  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }    @Override  public LoaiVe objectFromRs(ResultSet rs) throws Exception {  return new LoaiVe(rs.getString("ID\_LOAIVE"),  rs.getString("TEN"),  rs.getDouble("GIA"));  }    public Map<String, Double> getMapLoaiVe() throws Exception {  List<LoaiVe> l = selectAll();  return l.stream().collect(toMap(LoaiVe::getId, LoaiVe::getGia));  }  } |

#### Lớp DAO làm việc với bảng nhân viên

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.NhanVien;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.List;  public class NhanVienDao extends Dao<String, NhanVien> {  private String selectAll = "SELECT \* FROM NHANVIEN";  private String selecById = "SELECT \* FROM NHANVIEN WHERE ID\_NV = ?";  private String SELECT\_BY\_USER\_NAME = "SELECT \* FROM NHANVIEN WHERE TENTK = ?";  private String INSERT = "INSERT INTO NHANVIEN (HOTEN,NGAYSINH,GIOITINH,SODT,EMAIL,TENTK,MATKHAU,ANH,VAITRO) VALUES(?,?,?,?,?,?,?,?,?)";  private String UPDATE = "UPDATE NHANVIEN SET HOTEN=?, NGAYSINH=?, GIOITINH=?, SODT=?, EMAIL=?, TENTK=?,MATKHAU=?,ANH=?,VAITRO=? WHERE ID\_NV=?";  private String DELETE = "DELETE FROM NHANVIEN WHERE ID\_NV=?";  private String SELECT\_BY\_EMAIL = "SELECT \* FROM NHANVIEN WHERE EMAIL = ?";    @Override  public void insert(NhanVien entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT,  entity.getHoten(),entity.getNgaysinh(),entity.getGioitinh(),entity.getSodt(),entity.getEmail(),entity.getTentk(), entity.getMatkhau(),entity.getAnh(),entity.getVaitro());  }  @Override  public void update(NhanVien entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE,  entity.getHoten(),entity.getNgaysinh(),entity.getGioitinh(),entity.getSodt(),entity.getEmail(),entity.getTentk(), entity.getMatkhau(),entity.getAnh(),entity.getVaitro(),entity.getId());  }  @Override  public void delete(String value) throws Exception {  JDBCHelper.excecuteStatement(DELETE, value);  }  @Override  public NhanVien selectById(String value) throws Exception {  List<NhanVien> l = selectBySql(selecById, value);  return l.size() == 0? null : l.get(0);  }  @Override  public List<NhanVien> selectAll() throws Exception {  return selectBySql(selectAll);  }  @Override  public List<NhanVien> selectBySql(String sql, Object... args) throws Exception {  List<NhanVien> list = new ArrayList<>();  ResultSet rs = JDBCHelper.getResultSetByQuery(sql, args);  try {  while(rs.next()) {  list.add(objectFromRs(rs));  }  } catch (SQLException ex) {  ex.printStackTrace();  }finally {  rs.getStatement().getConnection().close();  return list;  }  }  @Override  public NhanVien objectFromRs(ResultSet rs) throws Exception {  return new NhanVien(rs.getString("ID\_NV"),  rs.getString("HOTEN"),  rs.getDate("NGAYSINH"),  rs.getBoolean("GIOITINH"),  rs.getString("SODT"),  rs.getString("EMAIL"),  rs.getBoolean("VAITRO"),  rs.getString("ANH"),  rs.getString("TENTK"),  rs.getString("MATKHAU")  );  }    public NhanVien selectByEmail(String value) throws Exception {  List<NhanVien> l = selectBySql(SELECT\_BY\_EMAIL, value);  return l.size() == 0? null : l.get(0);    }    public NhanVien selectByUserName(String value) throws Exception {  List<NhanVien> l = selectBySql(SELECT\_BY\_USER\_NAME, value);  return l.size() == 0? null : l.get(0);    }    } |

#### Lớp DAO làm việc với bảng phim

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.NhanVien;  import com.org.app.entity.Phim;  import com.org.app.helper.JDBCHelper;  import com.org.app.util.ValueOfInQuerryBuilder;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.Arrays;  import java.util.List;  public class PhimDao extends Dao<String, Phim> {  private String SELECT\_ALL = "SELECT \* FROM PHIM";  private String SELECT\_BY\_ID = "SELECT \* FROM PHIM WHERE ID\_PHIM = ?";  private String INSERT = "INSERT INTO PHIM (TEN,POSTER,THOILUONG,THELOAI,NGONNGU,NGAY\_KC,TOMTAT,TRANGTHAI,ID\_NV) VALUES(?,?,?,?,?,?,?,?,?)";  private String UPDATE = "UPDATE PHIM SET TEN=?, POSTER=?, THOILUONG=?, THELOAI=?, NGONNGU=?, NGAY\_KC=?,TOMTAT=?,TRANGTHAI=?,ID\_NV = ? WHERE ID\_PHIM=?";  private String DELETE = "DELETE FROM PHIM WHERE ID\_PHIM=?";  private String SELECT\_BY\_MUL\_IDS = "SELECT \* FROM PHIM WHERE ID\_PHIM IN(?)";  private String SELECT\_BY\_MUL\_SC = " SELECT DISTINCT P.\* FROM PHIM P JOIN SUATCHIEU SC ON SC.ID\_PHIM = P.ID\_PHIM " +  "WHERE SC.ID\_SUAT IN (?) ";  private String SELECT\_AVAILABLE\_PHIM = "SELECT \* FROM PHIM WHERE TRANGTHAI = 1";  @Override  public void insert(Phim entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT,  entity.getTen(), entity.getPoster(), entity.getThoiLuong(), entity.getTheloai(), entity.getNgonNgu(), entity.getNgayKhoiChieu(), entity.getTomTat(), entity.getTrangThai(), entity.getNv());  }  @Override  public void update(Phim entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE,  entity.getTen(), entity.getPoster(), entity.getThoiLuong(), entity.getTheloai(), entity.getNgonNgu(), entity.getNgayKhoiChieu(), entity.getTomTat(), entity.getTrangThai(), entity.getNv(), entity.getId());  }  @Override  public void delete(String value) throws Exception {  JDBCHelper.excecuteStatement(DELETE, value);  }  @Override  public Phim selectById(String value) throws Exception {  List<Phim> l = selectBySql(SELECT\_BY\_ID, value);  return l.size() == 0 ? null : l.get(0);  }  @Override  public List<Phim> selectAll() throws Exception {  return selectBySql(SELECT\_ALL);  }  @Override  public List<Phim> selectBySql(String sql, Object... args) throws Exception {  List<Phim> list = new ArrayList<>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  list.add(objectFromRs(rs));  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  @Override  public Phim objectFromRs(ResultSet rs) throws Exception {  return new Phim(  rs.getString("ID\_PHIM"),  rs.getString("TEN"),  rs.getString("POSTER"),  rs.getInt("THOILUONG"),  rs.getString("THELOAI"),  rs.getString("NGONNGU"),  rs.getDate("NGAY\_KC"),  rs.getString("TOMTAT"),  rs.getBoolean("TRANGTHAI"),  rs.getString("ID\_NV")  );  }    public List<Phim> selectByIds(String...args) throws Exception {  String sql = ValueOfInQuerryBuilder.any(SELECT\_BY\_MUL\_IDS, args.length);  System.out.println("ids = " + sql);  return selectBySql(sql, args);  }    public List<Phim> selectBySuatChieu(String...args) throws Exception {  if(args.length == 0) return null;  String sql = ValueOfInQuerryBuilder.any(SELECT\_BY\_MUL\_SC, args.length);  return selectBySql(sql, args);  }    public List<Phim> selectAllAvailable() throws Exception {  return selectBySql(SELECT\_AVAILABLE\_PHIM);  }  } |

#### Lớp DAO làm việc với bảng phòng chiếu

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.PhongChieu;  import com.org.app.helper.JDBCHelper;  import com.org.app.util.ValueOfInQuerryBuilder;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.util.ArrayList;  import java.util.LinkedHashMap;  import java.util.List;  import java.util.Map;  import java.util.function.Function;  import static java.util.stream.Collectors.toMap;  public class PhongChieuDao extends Dao<String, PhongChieu> {  String INSERT\_SQL = "INSERT INTO PHONGCHIEU (SOLUONGDAY, SOLUONGCOT) VALUES (?, ?, ?)";  String UPDATE\_SQL = "UPDATE PHONGCHIEU SET SOLUONGDAY = ?, SOLUONGCOT = ? WHERE ID\_PHONG = ?";  String DELETE\_SQL = "DELETE FROM PHONGCHIEU WHERE ID\_PHONG = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM PHONGCHIEU ORDER BY ID\_PHONG";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM PHONGCHIEU WHERE ID\_PHONG = ?";  String SELECT\_BY\_MUL\_ID = "SELECT \* FROM PHONGCHIEU WHERE ID\_PHONG IN (?)";  @Override  public void insert(PhongChieu entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getId(), entity.getSoLuongDay(), entity.getSoLuongCot());  }  @Override  public void update(PhongChieu entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE\_SQL, entity.getSoLuongDay(), entity.getSoLuongCot());  }  @Override  public void delete(String value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  @Override  public PhongChieu selectById(String value) throws Exception {  List<PhongChieu> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<PhongChieu> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }  public Map<String, PhongChieu> selectToMap(List<PhongChieu> phongs) throws Exception {  LinkedHashMap<String, PhongChieu> map = new LinkedHashMap<>();  for(PhongChieu p : phongs) {  map.put(p.getId(), p);  }  return map;  }    @Override  public List<PhongChieu> selectBySql(String sql, Object... args) throws Exception {  List<PhongChieu> list = new ArrayList<PhongChieu>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  list.add(objectFromRs(rs));  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  @Override  public PhongChieu objectFromRs(ResultSet rs) throws Exception {  return new PhongChieu(  rs.getString("ID\_PHONG"),  rs.getInt("soLuongDay"),  rs.getInt("soLuongCot")  );  }    public List<PhongChieu> selectbyMulIds(String...args) throws Exception {  List<PhongChieu> l = new ArrayList<>();  String sql = ValueOfInQuerryBuilder.any(SELECT\_BY\_MUL\_ID, args.length);  return selectBySql(sql, args);  }  } |

#### Lớp DAO làm việc với bảng suất chiếu

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.SuatChieu;  import com.org.app.helper.DateHelper;  import com.org.app.helper.JDBCHelper;  import com.org.app.helper.TimeHelper;  import java.sql.CallableStatement;  import java.sql.Date;  import java.sql.PreparedStatement;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.sql.Time;  import java.time.LocalDate;  import java.util.ArrayList;  import java.util.HashMap;  import java.util.List;  import java.util.Map;  import java.util.Vector;  import java.util.logging.Level;  import java.util.logging.Logger;  public class SuatChieuDao extends Dao<String, SuatChieu> {  private String INSERT\_SQL = "INSERT INTO SUATCHIEU (NGAYCHIEU, ID\_PHONG, ID\_PHIM, GIOBATDAU, GIOKETTHUC, ID\_NV) VALUES (?, ?, ?, ?, ?,?)";  private String UPDATE\_SQL = "UPDATE SUATCHIEU SET NGAYCHIEU = ?, ID\_PHONG = ?, ID\_PHIM = ?, GIOBATDAU = ?, GIOKETTHUC = ?, ID\_NV = ? WHERE ID\_SUAT = ?";  private String DELETE\_SQL = "DELETE FROM SUATCHIEU WHERE ID\_SUAT = ?";  private String SELECT\_ALL\_SQL = "SELECT \* FROM SUATCHIEU ORDER BY NGAYCHIEU, GIOBATDAU";  private String SELECT\_BY\_ID\_SQL = "SELECT \* FROM SUATCHIEU WHERE ID\_SUAT = ?";  private String SELECT\_WITH\_DATETIME\_AFTER = "SELECT \* FROM SUATCHIEU WHERE NGAYCHIEU = ? AND GIOBATDAU > ? ORDER BY NGAYCHIEU, GIOBATDAU";  private String SELECT\_WITH\_DATETIME\_BEFORE = "SELECT \* FROM SUATCHIEU WHERE NGAYCHIEU = ? AND GIOBATDAU < ? ORDER BY NGAYCHIEU, GIOBATDAU";  private String SELECT\_WITH\_DATE = "SELECT \* FROM SUATCHIEU WHERE NGAYCHIEU = ? ORDER BY GIOBATDAU";  private String SELECT\_TTDATVE = "{CALL SP\_LICHCHIEUNGAY(?,?,?,?)}";    private String SELECT\_BY\_NGAY\_PHONG = "SELECT \* FROM SUATCHIEU WHERE NGAYCHIEU = ? AND ID\_PHONG = ? ORDER BY GIOBATDAU";  private String SELECT\_SUAT\_TRONG\_KHUNG = "{CALL SP\_FINDTIMESLOT (?,?,?,?)}";  private String SELECT\_SUAT\_TRONG\_NGAY\_PHIM = "{CALL SP\_LICHCHIEUNGAY (?,?,?,?)}";  private String SELECT\_SUAT\_ALL = "{CALL SP\_SELECTLICH (?,?,?,?,?)}";  private String SELECT\_SUAT\_TRANG\_THAI = "{CALL SP\_SELECTLICHBYTRANGTHAI (?,?,?)}";  private String SELECT\_PHONG\_SUAT\_TOP3 = "{CALL SP\_SELECTLICHPHONG (?,?,?)}";  private String SELECT\_SUAT\_DA\_CHIEU\_PHONG = "{CALL SP\_SUATDACHIEUCUAPHONG (?,?,?)}";  private String SELECT\_LUONG\_VE\_SUAT = "select sc.ID\_SUAT, count(ID\_VEDAT) luongve from SUATCHIEU sc " +  "left join VEDAT vd on vd.ID\_SUAT = sc.ID\_SUAT " +  "group by sc.ID\_SUAT " +  "having count(ID\_VEDAT) > 0";      @Override  public void insert(SuatChieu entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getNgayChieu(), entity.getPhong(),  entity.getPhim(), entity.getGioBatDau(), entity.getGioKetThuc(), entity.getNv());  }  @Override  public void update(SuatChieu entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE\_SQL, entity.getNgayChieu(), entity.getPhong(),  entity.getPhim(), entity.getGioBatDau(), entity.getGioKetThuc(), entity.getNv(),entity.getId());  }  @Override  public void delete(String value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  @Override  public SuatChieu selectById(String value) throws Exception {  List<SuatChieu> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<SuatChieu> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public List<SuatChieu> selectBySql(String sql, Object... args) throws Exception {  List<SuatChieu> list = new ArrayList<SuatChieu>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  list.add(objectFromRs(rs));  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  @Override  public SuatChieu objectFromRs(ResultSet rs) throws Exception {  return new SuatChieu(rs.getString("ID\_SUAT"),  rs.getDate("NGAYCHIEU"),  rs.getString("ID\_PHONG"),  rs.getString("ID\_PHIM"),  rs.getTime("GIOBATDAU"),  rs.getTime("GIOKETTHUC"),  rs.getString("ID\_NV"));  }    public List<SuatChieu> selectBy(Date ngay, Time t) throws Exception {  String n = DateHelper.toStringForQuery(ngay);  String g = t.toString();  return selectBySql(n, g);  }    public List<SuatChieu> selectInComingSuatChieus(java.time.LocalDateTime now) throws Exception {  try {  String date = DateHelper.getDatePart(now).toString();  String time = TimeHelper.toString(now);  System.out.println("date = "+date +" time = "+time);  return selectBySql(SELECT\_WITH\_DATETIME\_AFTER,date,time);  }catch(Exception e) {  e.printStackTrace();  throw new RuntimeException("Loi ngay thang");  }  }    public List<SuatChieu> selectByDateOnly(java.sql.Date date) throws Exception{  return selectBySql(SELECT\_WITH\_DATE, date.toString());  }    // condition : 1 lon hon @ngay | 0 bang @ngay | -1 nho hon ngay  public List<SuatChieu> selectTTDatVeByDate(java.sql.Date date, java.sql.Time time, int condition) throws Exception{  //{ID\_PHIM, TEN, ID\_SUAT, ID\_PHONG, NGAYCHIEU, GIOBATDAU}  List<SuatChieu> l = new ArrayList<>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(SELECT\_TTDATVE, date, time, condition, null);  while(rs.next()) {  l.add(objectFromRs(rs));  }  return l;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }    public List<SuatChieu> selectByPhimNgay(java.sql.Date date, String id\_Phim) throws SQLException, Exception {  List<SuatChieu> l = new ArrayList<>();    Time t = null;  String[] para = {"NGAY","PHIM","CONDITION","GIO"};  ResultSet rs = null;  try {    rs = JDBCHelper.getRSOfProc(SELECT\_SUAT\_TRONG\_NGAY\_PHIM, para, date,id\_Phim, 0, t);  while(rs.next()) l.add(objectFromRs(rs));  return l;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  public List<SuatChieu> selectSuatChieuTrongKhoang(Date ngay, String phong, Time gioBatDau, Time gioKetThuc) throws Exception {  return selectBySql(SELECT\_SUAT\_TRONG\_KHUNG, ngay, phong, gioBatDau, gioKetThuc);  }    public List<SuatChieu> selectByNgayPhong(Date ngay, String phong) throws Exception {  return selectBySql(SELECT\_BY\_NGAY\_PHONG, ngay, phong);  }      //ID\_SUAT, ngaychieu, p.TEN TENPHIM, ID\_PHONG PHONG, THOILUONG , GIOBATDAU, GIOKETTHUC  public List<Vector<Object>> selectByAll(Object...args) throws Exception {  ResultSet rs = null;  try {  rs = JDBCHelper.getRSOfProc(SELECT\_SUAT\_ALL, new String[]{"ID\_PHIM","NGAYBD", "NGAYKT", "PHONG", "ISCHUACHIEU"}, args);  List<Vector<Object>> list = new ArrayList<>();  int count = 0;  while(rs.next()) {  Vector<Object> v = new Vector<>();  v.addAll(List.of(rs.getDate(2),  rs.getString(3),  rs.getString(4),  rs.getInt(5),  rs.getTime(6),  rs.getTime(7),  rs.getString(1))  );  list.add(v);  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }    public Integer[] selectSuatDaChieuCuaPhong(String id\_phong, java.sql.Date date, java.sql.Time time) throws SQLException, Exception {  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(SELECT\_SUAT\_DA\_CHIEU\_PHONG, id\_phong, date, time);  Integer[] a = new Integer[2];  while(rs.next()) {  a[0] = rs.getInt(1);  a[1] = rs.getInt(2);  }  return a;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }    public List<Object[]> selectTopSuatCuaPhong(String id\_phong, java.sql.Date date, java.sql.Time time) throws SQLException, Exception {  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(SELECT\_PHONG\_SUAT\_TOP3, id\_phong, date, time);  List<Object[]> list = new ArrayList<>();  //pc.ID\_PHONG, TEN, GIOBATDAU, GIOKETTHUC  while(rs.next()) {  list.add(new Object[] {  rs.getString(1),  rs.getString(2),  rs.getTime(3),  rs.getTime(4)  });  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }    public Map<String, Integer> selectLuongVeOfSuat() throws SQLException, Exception {  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(SELECT\_LUONG\_VE\_SUAT);  Map<String, Integer> map = new HashMap<>();  while(rs.next()) {  map.put(rs.getString(1), rs.getInt(2));  }  return map;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  } |

#### Lớp DAO làm việc với bảng thành viên

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.ThanhVien;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.util.ArrayList;  import java.util.List;  import javax.swing.text.html.parser.Entity;  public class ThanhVienDao extends Dao<String, ThanhVien>{  String INSERT\_SQL = "INSERT INTO THANHVIEN (HOTEN, NGAYSINH, GIOITINH, SODT, EMAIL) VALUES (?, ?, ?, ?, ?)";  String UPDATE\_SQL = "UPDATE THANHVIEN SET HOTEN = ?, NGAYSINH = ?, GIOITINH = ?, SODT = ?, EMAIL = ? WHERE ID\_TV = ?";  String DELETE\_SQL = "DELETE FROM THANHVIEN WHERE ID\_TV = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM THANHVIEN";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM THANHVIEN WHERE ID\_TV = ?";  @Override  public void insert(ThanhVien entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getHoTen(), entity.getNgaySinh(),  entity.getGioiTinh(), entity.getSoDT(), entity.getEmail());  }  @Override  public void update(ThanhVien entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE\_SQL, entity.getHoTen(), entity.getNgaySinh(),  entity.getGioiTinh(), entity.getSoDT(), entity.getEmail(), entity.getID());  }  @Override  public void delete(String value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  @Override  public ThanhVien selectById(String value) throws Exception {  List<ThanhVien> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<ThanhVien> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public List<ThanhVien> selectBySql(String sql, Object... args) throws Exception {  List<ThanhVien> list = new ArrayList<ThanhVien>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  list.add(objectFromRs(rs));    }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  @Override  public ThanhVien objectFromRs(ResultSet rs) throws Exception {  return new ThanhVien(  rs.getString("id\_tv"),  rs.getString("hoten"),  rs.getDate("ngaysinh"),  rs.getBoolean("gioitinh"),  rs.getString("soDT"),  rs.getString("email"),  rs.getString("makh")  );  }    public List<ThanhVien> selectbyKeyword(String keyword, String keyword1) throws Exception {  String sql = "SELECT \*FROM THANHVIEN WHERE HOTEN LIKE ? ";  return this.selectBySql(sql, "%" + keyword + "%");  }  public List<ThanhVien> selectNotinCourse(int makh, String keyword) throws Exception {  String sql = "SELECT \*FROM THANHVIEN WHERE HOTEN LIKE ? AND ID\_TV NOT IN "  + "(SELECT ID\_TV FROM THANHVIEN WHERE ID\_TV = ?)";  return this.selectBySql(sql, "%" + keyword + "%", makh);  }  } |

#### Lớp DAO làm việc với bảng thống kê

|  |
| --- |
| package com.org.app.controller;  import com.org.app.helper.JDBCHelper;  import com.org.app.helper.TimeHelper;  import java.sql.Date;  import java.util.ArrayList;  import java.util.List;  import java.sql.ResultSet;  import java.sql.SQLException;  public class ThongKeDAO {  String TAO\_KHUNG\_GIO\_SQL = "SELECT \* FROM DBO.FKHUNGGIO(?,?)";    private List<Object[]> getListOfArray(String sql, String[] cols, Object... args) throws Exception {  ResultSet rs = null;  List<Object[]> list = null;  try {  list = new ArrayList<>();  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  Object[] vals = new Object[cols.length];  for (int i = 0; i < cols.length; i++) {  vals[i] = rs.getObject(cols[i]);  }  list.add(vals);    }  System.out.println("aha finish");  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  return list;  }  // DOANH THU PHÒNG VÉ  public List<Object[]> getDoanhThuPhongVe() throws Exception {  String sql = "{CALL sp\_doanhthuphongve}";  String[] cols = {"thang", "tongsosuatchieu", "sovedaban", "doanhthu"};  return this.getListOfArray(sql, cols);  }  // DOANH THU PHÒNG VÉ  public List<Object[]> getDoanhThuPhongVe\_SapXep() throws Exception {  String sql = "{CALL sp\_doanhthuphongve\_sapxep}";  String[] cols = {"thang", "tongsosuatchieu", "sovedaban", "doanhthu"};  return this.getListOfArray(sql, cols);  }    // ;ỌC DOANH THU PHÒNG VÉ THEO NGÀY  public List<Object[]> getDoanhThuPhongVe\_TheoNgay(Date ngay) throws Exception {  String sql = "{CALL sp\_doanhthuphongve\_theongay(?)}";  String[] cols = {"thang", "tongsosuatchieu", "sovedaban", "doanhthu"};  return this.getListOfArray(sql, cols, ngay);  }  // LỌC DOANH THU PHÒNG VÉ THEO THÁNG  public List<Object[]> getDoanhThuPhongVe\_TheoThang(int thang, int nam) throws Exception {  String sql = "{CALL sp\_doanhthuphongve\_theothangnam(?,?)}";  String[] cols = {"thang", "tongsosuatchieu", "sovedaban", "doanhthu"};  return this.getListOfArray(sql, cols, thang, nam);  }  // LỌC DOANH THU PHÒNG VÉ THEO THÁNG  public List<Object[]> getDoanhThuPhongVe\_TheoNam(int nam) throws Exception {  String sql = "{CALL sp\_doanhthuphongve\_theonam(?)}";  String[] cols = {"thang", "tongsosuatchieu", "sovedaban", "doanhthu"};  return this.getListOfArray(sql, cols, nam);  }  // DOANH THU PHIM  public List<Object[]> getDoanhThuPhim() throws Exception {  String sql = "{CALL sp\_doanhthuphim}";  String[] cols = {"TEN", "sosuatchieu", "sovedat", "tonggia"};  return this.getListOfArray(sql, cols);  }  // DOANH THU PHIM  public List<Object[]> getDoanhThuPhim\_SapXep() throws Exception {  String sql = "{CALL sp\_doanhthuphim\_sapxep}";  String[] cols = {"TEN", "sosuatchieu", "sovedat", "tonggia"};  return this.getListOfArray(sql, cols);  }    // DOANH THU PHIM  public List<Object[]> getDoanhThuPhim\_Top() throws Exception {  String sql = "{CALL sp\_doanhthuphim\_top}";  String[] cols = {"TEN", "sosuatchieu", "sovedat", "tonggia"};  return this.getListOfArray(sql, cols);  }    // LỌC DOANH THU PHIM THEO NGÀY  public List<Object[]> getDoanhThuPhim\_TheoNgay(Date ngay) throws Exception {  String sql = "{CALL sp\_doanhthuphim\_theongay(?)}";  String[] cols = {"TEN", "sosuatchieu", "sovedat", "tonggia"};  return this.getListOfArray(sql, cols, ngay);  }  // LỌC DOANH THU PHIM THEO THÁNG  public List<Object[]> getDoanhThuPhim\_TheoThang(int thang, int nam) throws Exception {  String sql = "{CALL sp\_doanhthuphim\_theothangnam(?,?)}";  String[] cols = {"TEN", "sosuatchieu", "sovedat", "tonggia"};  return this.getListOfArray(sql, cols, thang, nam);  }  // LỌC DOANH THU PHIM THEO THÁNG  public List<Object[]> getDoanhThuPhim\_TheoNam(int nam) throws Exception {  String sql = "{CALL sp\_doanhthuphim\_theonam(?)}";  String[] cols = {"TEN", "sosuatchieu", "sovedat", "tonggia"};  return this.getListOfArray(sql, cols, nam);  }  // LƯỢNG VÉ ĐÃ BÁN  public List<Object[]> getLuongVeDB() throws Exception {  String sql = "{CALL sp\_luongvedaban}";  String[] cols = {"NGAYCHIEU", "sovedaban", "tongsuatchieu"};  return this.getListOfArray(sql, cols);  }  // SẮP XẾP LƯỢNG VÉ ĐÃ BÁN  public List<Object[]> getLuongVeDB\_SapXep() throws Exception {  String sql = "{CALL sp\_luongvedaban\_sapxep}";  String[] cols = {"NGAYCHIEU", "sovedaban", "tongsuatchieu"};  return this.getListOfArray(sql, cols);  }  // LỌC LƯỢNG VÉ ĐÃ BÁN THEO NGÀY  public List<Object[]> getLuongVeDB\_TheoNgay(Date ngay) throws Exception {  String sql = "{CALL sp\_luongvedaban\_theongay(?)}";  String[] cols = {"NGAYCHIEU", "sovedaban", "tongsuatchieu"};  return this.getListOfArray(sql, cols, ngay);  }  // LỌC LƯỢNG VÉ ĐÃ BÁN THEO THÁNG  public List<Object[]> getLuongVeDB\_TheoThang(int thang, int nam) throws Exception {  String sql = "{CALL sp\_luongvedaban\_theothangnam(?,?)}";  String[] cols = {"NGAYCHIEU", "sovedaban", "tongsuatchieu"};  return this.getListOfArray(sql, cols, thang, nam);  }  // LỌC LƯỢNG VÉ ĐÃ BÁN THEO NĂM  public List<Object[]> getLuongVeDB\_TheoNam(int nam) throws Exception {  String sql = "{CALL sp\_luongvedaban\_theonam(?)}";  String[] cols = {"NGAYCHIEU", "sovedaban", "tongsuatchieu"};  return this.getListOfArray(sql, cols, nam);  }  // DOANH THU ĐỒ ĂN  public List<Object[]> getDTDoAN() throws Exception {  String sql = "{CALL sp\_doanhthudoan}";  String[] cols = {"TEN", "SOLUONG", "ID\_KICHCO", "THANHTIEN"};  return this.getListOfArray(sql, cols);  }  // LỌC DOANH THU ĐỒ ĂN THEO NGÀY  public List<Object[]> getDTDoAN\_TheoNgay(Date ngay) throws Exception {  String sql = "{CALL sp\_doanhthudoan\_theongay(?)}";  String[] cols = {"TEN", "SOLUONG", "ID\_KICHCO", "THANHTIEN"};  return this.getListOfArray(sql, cols, ngay);  }  // LỌC DOANH THU ĐỒ ĂN THEO THÁNG  public List<Object[]> getDTDoAN\_TheoThang(int thang, int nam) throws Exception {  String sql = "{CALL sp\_doanhthudoan\_theothangnam(?,?)}";  String[] cols = {"TEN", "SOLUONG", "ID\_KICHCO", "THANHTIEN"};  return this.getListOfArray(sql, cols, thang, nam);  }  // LỌC DOANH THU ĐỒ ĂN THEO NĂM  public List<Object[]> getDTDoAN\_TheoNam(int nam) throws Exception {  String sql = "{CALL sp\_doanhthudoan\_theonam(?)}";  String[] cols = {"TEN", "SOLUONG", "ID\_KICHCO", "THANHTIEN"};  return this.getListOfArray(sql, cols, nam);  }  // LỌC DOANH THU ĐỒ ĂN THEO TÊN  public List<Object[]> getDTDoAN\_TheoTen(String ten) throws Exception {  String sql = "{CALL sp\_doanhthudoan\_theoten(?)}";  String[] cols = {"TEN", "SOLUONG", "ID\_KICHCO", "THANHTIEN"};  return this.getListOfArray(sql, cols, ten);  }  // LỌC DOANH THU ĐỒ ĂN THEO SIZE  public List<Object[]> getDTDoAN\_TheoSize(String size) throws Exception {  String sql = "{CALL sp\_doanhthudoan\_theosize(?)}";  String[] cols = {"TEN", "SOLUONG", "ID\_KICHCO", "THANHTIEN"};  return this.getListOfArray(sql, cols, size);  }    // SẮP XẾP DOANH THU ĐỒ ĂN  public List<Object[]> getDTDoAN\_SapXep() throws Exception {  String sql = "{CALL sp\_sapxepdoan}";  String[] cols = {"TEN", "SOLUONG", "ID\_KICHCO", "THANHTIEN"};  return this.getListOfArray(sql, cols);  }    // DOANH THU TỔNG  public List<Object[]> getDoanhThuTong() throws Exception {  String sql = "{CALL sp\_doanhthutong}";  String[] cols = {"DTPV", "DTDA", "tongDoanhThu"};  return this.getListOfArray(sql, cols);  }    // DOANH THU TỔNG THEO NGÀY  public List<Object[]> getDoanhThuTong\_TheoNgay(Date ngay) throws Exception {  String sql = "{CALL sp\_doanhthutong\_ngay(?)}";  String[] cols = {"DTPV", "DTDA", "tongDoanhThu"};  return this.getListOfArray(sql, cols, ngay);  }    // DOANH THU TỔNG THEO THÁNG  public List<Object[]> getDoanhThuTong\_TheoThang(int thang, int nam) throws Exception {  String sql = "{CALL sp\_doanhthutong\_thang(?,?)}";  String[] cols = {"DTPV", "DTDA", "tongDoanhThu"};  return this.getListOfArray(sql, cols, thang, nam);  }    // DOANH THU TỔNG THEO NĂM  public List<Object[]> getDoanhThuTong\_TheoNam(int nam) throws Exception {  String sql = "{CALL sp\_doanhthutong\_nam(?)}";  String[] cols = {"DTPV", "DTDA", "tongDoanhThu"};  return this.getListOfArray(sql, cols, nam);  }    // THONG KE THEO KHUNG GIO  public List<Object[]> getThongKeKG(int khoangCachKG, Date ngay) throws Exception {  String sql = "{CALL sp\_thongke\_khunggio(?,?)}";  String[] cols = {"KHUNGGIO","SOSUATCHIEU", "SOVEDAT"};  return this.getListOfArray(sql, cols, khoangCachKG, ngay);  }  } |

#### Lớp DAO làm việc với bảng vé đặt

|  |
| --- |
| package com.org.app.controller;  import com.org.app.entity.VeDat;  import com.org.app.helper.DateHelper;  import com.org.app.helper.JDBCHelper;  import java.sql.ResultSet;  import java.sql.SQLException;  import java.time.LocalDate;  import java.util.ArrayList;  import java.util.List;  import java.util.logging.Level;  import java.util.logging.Logger;  public class VeDatDao extends Dao<Integer, VeDat> {  String INSERT\_SQL = "INSERT INTO VEDAT (ID\_LOAIVE, ID\_GHE, ID\_DONVE, ID\_SUAT) VALUES (?, ?, ?, ?)";  String UPDATE\_SQL = "UPDATE VEDAT SET ID\_LOAIVE = ?, ID\_GHE = ?, ID\_DONVE = ?, ID\_SUAT = ? WHERE ID\_VEDAT = ?";  String DELETE\_SQL = "DELETE FROM VEDAT WHERE ID\_VEDAT = ?";  String SELECT\_ALL\_SQL = "SELECT \* FROM VEDAT";  String SELECT\_BY\_ID\_SQL = "SELECT \* FROM VEDAT WHERE ID\_VEDAT = ?";  String SELECT\_LUONG\_VE\_SUAT = "{CALL SP\_LUONGVEBANCUASUAT (?)}";  String SELECT\_LUONG\_VE\_BAN\_NGAY = "SELECT count(ID\_VEDAT)\n" +  "FROM DONTHANHTOAN dtt JOIN DONVE dv" +  " ON dv.ID\_DONVE = dtt.ID\_DONVE " +  "JOIN VEDAT vd ON vd.ID\_DONVE = dv.ID\_DONVE " +  "JOIN SUATCHIEU SC ON vd.ID\_SUAT = SC.ID\_SUAT " +  "WHERE cast(ngaydat as date) = ? or sc.NGAYCHIEU = ?";  @Override  public void insert(VeDat entity) throws Exception {  JDBCHelper.excecuteStatement(INSERT\_SQL, entity.getLoaiVe(), entity.getGhe(), entity.getDonVe(),  entity.getSuatChieu());  }  @Override  public void update(VeDat entity) throws Exception {  JDBCHelper.excecuteStatement(UPDATE\_SQL, entity.getLoaiVe(), entity.getGhe(),  entity.getDonVe(), entity.getSuatChieu());  }  @Override  public void delete(Integer value) throws Exception {  JDBCHelper.excecuteStatement(DELETE\_SQL, value);  }  @Override  public VeDat selectById(Integer value) throws Exception {  List<VeDat> list = this.selectBySql(SELECT\_BY\_ID\_SQL, value);  if (list.isEmpty()) {  return null;  }  return list.get(0);  }  @Override  public List<VeDat> selectAll() throws Exception {  return this.selectBySql(SELECT\_ALL\_SQL);  }  @Override  public List<VeDat> selectBySql(String sql, Object... args) throws Exception {  List<VeDat> list = new ArrayList<VeDat>();  ResultSet rs = null;  try {  rs = JDBCHelper.getResultSetByQuery(sql, args);  while (rs.next()) {  list.add(objectFromRs(rs));  }  return list;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  /\*[ID\_VEDAT],[ID\_LOAIVE],[ID\_GHE],[ID\_DONVE],[ID\_SUAT]\*/  @Override  public VeDat objectFromRs(ResultSet rs) throws Exception {  return new VeDat(rs.getInt("ID\_VEDAT"),  rs.getString("ID\_LOAIVE"),  rs.getString("ID\_GHE"),  rs.getString("ID\_DONVE"),  rs.getString("ID\_SUAT"));  }  public int getLuongVeOf(String id\_suat) throws Exception {  ResultSet rs = null;  try{  rs = JDBCHelper.getResultSetByQuery(SELECT\_LUONG\_VE\_SUAT, id\_suat);  rs.next();  return rs.getInt("LUONGVE");  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }    public int getLuongVeBanNgay(LocalDate date) throws Exception {  Integer luongVe = 0;  ResultSet rs = null;  try {  java.util.Date d = DateHelper.convertToDate(date);  String df = DateHelper.toString(d, DateHelper.YYYYMMDD\_FORMAT);  System.out.println("df = "+ df);  rs = JDBCHelper.getResultSetByQuery(SELECT\_LUONG\_VE\_BAN\_NGAY, df,df);  while (rs.next()){  luongVe = rs.getInt(1);  }  return luongVe == null? 0 : luongVe;  }catch (Exception ex) {  ex.printStackTrace();  throw new RuntimeException("Lỗi truy suất dữ liệu");  }finally {  if(rs != null) {  rs.getStatement().getConnection().close();  if(!rs.isClosed()) {  rs.getStatement().close();  rs.close();  }  }  }  }  } |

# Kiểm thử

## Kiểm thử form LoginFrame

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Miêu tả tình huống** | **Dữ liệu vào** | **Kết quả trông đợi** | **Chi tiết các bước** |
| 1 | Để trống Usename/Email và Password | 1/ Username/Email:  2/ Password: | Hiển thị thông báo không được để trống Usename/Email hoặc Password | 1/ Để trống Username/Email  2/ Để trống Password  3/ Nhấn “Đăng nhập” |
| 2 | Nhập sai Username/Email | 1/ Username/Email: vanth  2/ Password: Abcd123 | Hiển thị thông báo Usename hoặc Email không tồn tại | 1/ Nhập sai Uesname/Email  2/ Nhập Password  3/ Nhấn “Đăng nhập” |
| 3 | Nhập sai Password | 1/ Username/Email: vantht  2/ Password: Abcd12 | Hiển thị thông báo sai Password | 1/ Nhập Uesname/Email  2/ Nhập sai Password  3/ Nhấn “Đăng nhập” |
| 4 | Nhập đúng Username/Email và Password | 1/ Username/Email: vantht  2/ Password: Abcd123 | Hiển thị thông báo đăng nhập thành công | 1/ Nhập đúng Uesname/Email  2/ Nhập đúng Password  3/ Nhấn “Đăng nhập” |
| 5 | Nhấn nút Kết thúc | 1/ Nhấn vào icon dấu X | Hiển thị hộp thoại đóng chương trình | 1/ Nhấn vào icon dấu X |
| 6 | Nhấn nút Hiện mật khẩu | 1/ Nhấn vào icon hiện mật khẩu | Nhấn giữ: Hiển thị mật khẩu tại ô mật khẩu  Thả: Ẩn mật khẩu | 1/ Nhấn vào icon hiện mật khẩu |

## Kiểm thử form qLDoAnKemFrame

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Miêu tả tình huống** | **Dữ liệu vào** | **Kết quả trông đợi** | **Chi tiết các bước** |
| 1 | Để trống tên món và giá khi thêm/sửa | 1/ Tên món:  2/ Giá: | Hiển thị thông báo không được để trống ô nhập | 1/ Để trống tên món  2/ Để trống giá  3/ Nhấn nút “Thêm”/”Sửa” |
| 2 | Nhập giá là chữ | 1/ Tên món: Hotdog  2/ Giá: abc | Hiển thị thông báo giá phải là số | 1/ Nhập tên món  2/ Nhập giá là chữ  3/ Nhấn “Thêm”/”Sửa” |
| 3 | Nhập giá < 0 | 1/ Tên món: Hotdog  2/ Giá: -10 | Hiện thông báo giá phải lớn hơn 0 | 1/ Nhập tên món  2/ Nhập giá < 0  3/ Nhấn “Thêm”/”Sửa” |
| 4 | Thêm tên món và size đã có | 1/ Tên món: Sushi  2/ Size: S | Hiện thông báo món ăn và kích cỡ này đã tồn tại | 1/ Nhập tên món trùng  2/ Chọn size trùng  3/ Nhấn “Thêm” |
| 5 | Xóa 1 dòng bất kì | 1/ Chọn dòng bất kỳ trên bảng | Hiển thị thông báo xóa thành công | 1/ Nhấn chọn 1 dòng trên bảng  2/ Nhấn “Xóa” |
| 6 | Xóa trắng form |  | 1/ Xóa trắng các TextFiled  2/ Khóa lại nút “Sửa” và “Xóa” | 1/ Nhấn “Mới” |
| 7 | Nhấn chọn các nút di chuyển |  | 1/ Hiển thị chi tiết dòng được chọn lên form  2/ Khóa nút “First”/”Last” khi dòng được chọn ở đầu/cuối bảng | 1/ Nhấn chọn các nút di chuyển |
| 8 | Nhập đúng thông tin khi thêm/sửa | 1/ Tên món: Hotdog  2/ Size: S  3/ Giá: 10.000 | Hiện thông báo thêm/sửa thành công | 1/ Nhập tên món  2/ Chọn size  3/ Nhập giá  4/ Nhấn “Thêm”/”Sửa” |

## Kiểm thử form qldonthanhtoanframe

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Miêu tả tình huống** | **Dữ liệu vào** | **Kết quả trông đợi** | **Chi tiết các bước** |
| 1 | Nhập từ khóa cần tìm | 1/ txtTim: (từ khóa cần tìm) | Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhập từ khóa cần tìm vào TextField  2/ Nhấn “Tìm” |
| 2 | Lọc theo ngày | 1/ Nhấn “Theo ngày” | 1/ Khóa các lựa chọn ”Theo tháng”/ ”Theo năm”/ “Khách”  2/ Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhấn “Theo ngày”  2/ Chọn ngày bắt đầu  3/ Chọn ngày kết thúc |
| 3 | Lọc theo tháng | 1/ Nhấn “Theo tháng” | 1/ Khóa các lựa chọn ”Theo ngày”/ ”Theo năm”/ “Khách”  2/ Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhấn “Theo tháng”  2/ Chọn tháng - năm |
| 4 | Lọc theo năm | 1/ Nhấn “Theo năm | 1/ Khóa các lựa chọn ”Theo ngày”/ ”Theo tháng”/ “Khách”  2/ Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhấn “Theo năm”  2/ Chọn năm |
| 5 | Lọc theo khách | 1/ Nhấn “Khách” | 1/ Khóa các lựa chọn “Theo ngày”/ ”Theo tháng”/ ”Theo năm”  2/ Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhấn “Khách”  2/ Chọn loại khách hàng |
| 6 | Xóa 1 dòng bất kì | 1/ Chọn 1 dòng bất kỳ trên bảng | Hiển thị thông báo xóa thành công | 1/ Nhấn chọn 1 dòng trên bảng  2/ Nhấn “Xóa” |

## kiểm thử form khachhangFrame

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Miêu tả tình huống** | **Dữ liệu vào** | **Kết quả trông đợi** | **Chi tiết các bước** |
| 1 | Xóa trắng form |  | 1/ Xóa trắng các TextFiled  2/ Khóa lại nút “Sửa” và “Xóa” | 1/ Nhấn “Mới” |
| 2 | Để trống các ô nhập khi thêm/sửa | 1/ Họ tên:  2/ Số điện thoại:  3/ Emai: | Hiện thông báo không được để trống ô nhập | 1/ Để trống họ tên  2/ Để trống số điện thoại  3/ Để trống email  4/ Nhấn “Thêm” "/ “Sửa” |
| 3 | Nhập họ tên không đúng định dạng khi thêm/sửa | 1/ Họ tên: Ab  2/ Số điện thoại: 09174385938  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 2/2/2002  5/ Giới tính: Nam/Nữ | Hiện thông báo họ tên phải từ 3 – 50 ký tự | 1/ Nhập sai định dạng họ tên  2/ Nhập số điện thoại  3/ Nhập email  4/ Nhập ngày sinh  5/ Chọn giới tính  6/ Nhấn “Thêm""/ “Sửa” |
| 4 | Nhập điện thoại không đúng định dạng khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09235731a  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 2/2/2002  5/ Giới tính: Nam/Nữ | Hiện thông báo số điện thoại không đúng định dạng | 1/ Nhập họ tên  2/ Nhập sai định dạng số điện thoại  3/ Nhập email  4/ Nhập ngày sinh  5/ Chọn giới tính  6/ Nhấn “Thêm""/ “Sửa” |
| 5 | Nhập email không đúng định dạng khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385938  3/ Email: [abcgmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 2/2/2002  5/ Giới tính: Nam/Nữ | Hiện thông báo email không đúng định dạng | 1/ Nhập họ tên  2/ Nhập số điện thoại  3/ Nhập sai định dạng email  4/ Nhập ngày sinh  5/ Chọn giới tính  6/ Nhấn “Thêm"/ “Sửa” |
| 6 | Nhập ngày sinh sai định dạng khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385938  3/ Email: [abcgmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 2/2/2012  5/ Giới tính: Nam/Nữ | Hiện thông báo ngày sinh phải từ 18 tuổi trở lên | 1/ Nhập họ tên  2/ Nhập số điện thoại  3/ Nhập email  4/ Nhập sai định dạng ngày sinh  5/ Chọn giới tính  6/ Nhấn “Thêm"/ “Sửa” |
| 7 | Xóa 1 dòng bất kì | 1/ Chọn dòng bất kỳ trên bảng | Hiển thị thông báo xóa thành công | 1/ Nhấn chọn 1 dòng trên bảng  2/ Nhấn “Xóa” |
| 8 | Nhấn chọn các nút di chuyển |  | 1/ Hiển thị chi tiết dòng được chọn lên form  2/ Khóa nút “First”/”Last” khi dòng được chọn ở đầu/cuối bảng | 1/ Nhấn chọn các nút di chuyển |
| 9 | Nhập từ khóa cần tìm | 1/ txtTim: (từ khóa cần tìm) | Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhập từ khóa cần tìm vào TextField |
| 10 | Nhập đúng thông tin khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385938  3/ Email: [abcgmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 2/2/2002  5/ Giới tính: Nam/Nữ | Hiện thông báo thêm/sửa thành công | 1/ Nhập họ tên  2/ Nhập số điện thoại  3/ Nhập email  4/ Nhập định dạng ngày sinh  5/ Chọn giới tính  6/ Nhấn “Thêm"/ “Sửa” |

## kiểm thử form qlnhanvienframe

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Miêu tả tình huống** | **Dữ liệu vào** | **Kết quả trông đợi** | **Chi tiết các bước** |
| 1 | Xóa trắng form |  | 1/ Xóa trắng các TextFiled  2/ Khóa lại nút “Sửa” và “Xóa” | 1/ Nhấn “Mới” |
| 2 | Để trống các ô nhập khi thêm/sửa | 1/ Họ tên:  2/ Số điện thoại:  3/ Emai:  4/ Tài khoản:  5/ Mật khẩu:  6/ Xác nhận | Hiện thông báo không được để trống ô nhập | 1/ Để trống họ tên  2/ Để trống số điện thoại  3/ Để trống email  4/ Để trống tài khoản  5/ Để trống mật khẩu  6/ Để trống xác nhận  7/ Nhấn “Thêm” "/ “Sửa” |
| 3 | Nhập họ tên không đúng định dạng khi thêm/sửa | 1/ Họ tên: Ab  2/ Số điện thoại: 09174385938  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 1/12/1992  5/ Tài khoản: hoathien9x  6/ Mật khẩu: HoaThien92  7/ Xác nhận: HoaThien92  8/ Giới tính: Nam/Nữ  9/ Vai trò: Nhân viên/ Quản lý | Hiện thông báo họ tên phải từ 3 – 50 ký tự | 1/ Nhập sai định dạng họ tên  2/ Nhập số điện thoại  3/ Nhập email  4/ Nhập ngày sinh  5/ Nhập tài khoản  6/ Nhập mật khẩu  7/ Nhập xác nhận  8/ Chọn giới tính  9/ Chọn vai trò  10/ Nhấn “Thêm""/ “Sửa” |
| 4 | Nhập điện thoại không đúng định dạng khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 0917438593a  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 1/12/1992  5/ Tài khoản: hoathien9x  6/ Mật khẩu: HoaThien92  7/ Xác nhận: HoaThien92  8/ Giới tính: Nam/Nữ  9/ Vai trò: Nhân viên/ Quản lý | Hiện thông báo số điện thoại không đúng định dạng | 1/ Nhập sai định dạng họ tên  2/ Nhập sai định dạng số điện thoại  3/ Nhập email  4/ Nhập ngày sinh  5/ Nhập tài khoản  6/ Nhập mật khẩu  7/ Nhập xác nhận  8/ Chọn giới tính  9/ Chọn vai trò  10/ Nhấn “Thêm""/ “Sửa” |
| 5 | Nhập email không đúng định dạng khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385938  3/ Email: [abcgmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 1/12/1992  5/ Tài khoản: hoathien9x  6/ Mật khẩu: HoaThien92  7/ Xác nhận: HoaThien92  8/ Giới tính: Nam/Nữ  9/ Vai trò: Nhân viên/ Quản lý | Hiện thông báo email không đúng định dạng | 1/ Nhập họ tên  2/ Nhập số điện thoại  3/ Nhập sai định dạng email  4/ Nhập ngày sinh  5/ Nhập tài khoản  6/ Nhập mật khẩu  7/ Nhập xác nhận  8/ Chọn giới tính  9/ Chọn vai trò  10/ Nhấn “Thêm""/ “Sửa” |
| 6 | Nhập ngày sinh sai định dạng khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385938  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 1/12/2002  5/ Tài khoản: hoathien9x  6/ Mật khẩu: HoaThien92  7/ Xác nhận: HoaThien92  8/ Giới tính: Nam/Nữ  9/ Vai trò: Nhân viên/ Quản lý | Hiện thông báo ngày sinh phải từ 16 – 65 tuổi | 1/ Nhập họ tên  2/ Nhập số điện thoại  3/ Nhập email  4/ Nhập sai định dạng ngày sinh  5/ Nhập tài khoản  6/ Nhập mật khẩu  7/ Nhập xác nhận  8/ Chọn giới tính  9/ Chọn vai trò  10/ Nhấn “Thêm""/ “Sửa” |
| 7 | Nhập tài khoản sai định dạng khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385938  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 1/12/2002  5/ Nhập thông tin  - Tài khoản: hoa  - Tài khoản: Hoathien  6/ Mật khẩu: HoaThien92  7/ Xác nhận: HoaThien92  8/ Giới tính: Nam/Nữ  9/ Vai trò: Nhân viên/ Quản lý | **TH1:** Hiện thông báo tài khoản phải từ 6 – 15 ký tự  **TH2:** Hiện thông báo tài khoản không chứa ký tự in hoa và ký tự đặc biệt | 1/ Nhập họ tên  2/ Nhập số điện thoại  3/ Nhập email  4/ Nhập ngày sinh  5/ Nhập sai định dạng tài khoản  6/ Nhập mật khẩu  7/ Nhập xác nhận  8/ Chọn giới tính  9/ Chọn vai trò  10/ Nhấn “Thêm""/ “Sửa” |
| 8 | Nhập tài khoản đã tồn tại khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385938  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 1/12/2002  5/ Tài khoản: vynguyen97  6/ Mật khẩu: HoaThien92  7/ Xác nhận: HoaThien92  8/ Giới tính: Nam/Nữ  9/ Vai trò: Nhân viên/ Quản lý | Hiện thông báo tên tài khoản trùng | 1/ Nhập họ tên  2/ Nhập số điện thoại  3/ Nhập email  4/ Nhập ngày sinh  5/ Nhập trùng tài khoản  6/ Nhập mật khẩu  7/ Nhập xác nhận  8/ Chọn giới tính  9/ Chọn vai trò  10/ Nhấn “Thêm""/ “Sửa” |
| 9 | Nhập mật khẩu sai dạng khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385938  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 1/12/2002  5/ Tài khoản: hoathien9x  6/ Nhập thông tin:  - Mật khẩu: hoa  - Mật khẩu: hoathien/ Hoathien/ hoaThien1  7/ Xác nhận: HoaThien92  8/ Giới tính: Nam/Nữ  9/ Vai trò: Nhân viên/ Quản lý | **TH1:** Hiện thông báo tài khoản phải từ 6 – 10 ký tự  **TH2:** Hiện thông báo mật khẩu phải ít nhất 1 chữ hoa, thường và số | 1/ Nhập họ tên  2/ Nhập số điện thoại  3/ Nhập email  4/ Nhập ngày sinh  5/ Nhập tài khoản  6/ Nhập sai định dạng mật khẩu  7/ Nhập xác nhận  8/ Chọn giới tính  9/ Chọn vai trò  10/ Nhấn “Thêm""/ “Sửa” |
| 10 | Nhập xác nhận sai khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385938  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 1/12/2002  5/ Tài khoản: hoathien9x  6/ Mật khẩu: HoaThien92  7/ Xác nhận: HoaThien  8/ Giới tính: Nam/Nữ  9/ Vai trò: Nhân viên/ Quản lý | Hiện thông báo xác nhận mật khẩu không trùng khớp | 1/ Nhập họ tên  2/ Nhập số điện thoại  3/ Nhập email  4/ Nhập ngày sinh  5/ Nhập tài khoản  6/ Nhập mật khẩu  7/ Nhập sai xác nhận  8/ Chọn giới tính  9/ Chọn vai trò  10/ Nhấn “Thêm""/ “Sửa” |
| 11 | Xóa 1 dòng bất kì | 1/ Chọn dòng bất kỳ trên bảng | Hiển thị thông báo xóa thành công | 1/ Nhấn chọn 1 dòng trên bảng  2/ Nhấn “Xóa” |
| 12 | Nhấn chọn các nút di chuyển |  | 1/ Hiển thị chi tiết dòng được chọn lên form  2/ Khóa nút “First”/”Last” khi dòng được chọn ở đầu/cuối bảng | 1/ Nhấn chọn các nút di chuyển |
| 13 | Chọn ảnh |  | Hiện ảnh tưng ứng lên label | 1/ Double click vào label chứa ảnh  2/ Chọn ảnh |
| 14 | Nhập đúng thông tin khi thêm/sửa | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385938  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 1/12/2002  5/ Tài khoản: hoathien9x  6/ Mật khẩu: HoaThien92  7/ Xác nhận: HoaThien92  8/ Giới tính: Nam/Nữ  9/ Vai trò: Nhân viên/ Quản lý | Hiện thông báo thêm/sửa thành công | 1/ Nhập họ tên  2/ Nhập số điện thoại  3/ Nhập email  4/ Nhập ngày sinh  5/ Nhập tài khoản  6/ Nhập mật khẩu  7/ Nhập xác nhận  8/ Chọn giới tính  9/ Chọn vai trò  10/ Nhấn “Thêm""/ “Sửa” |

## kiểm thử form qlphimframe

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Miêu tả tình huống** | **Dữ liệu vào** | **Kết quả trông đợi** | **Chi tiết các bước** |
| 1 | Xóa trắng form |  | 1/ Xóa trắng các TextFiled  2/ Khóa lại nút “Sửa” và “Xóa” | 1/ Nhấn “Mới” |
| 2 | Để trống tên phim khi thêm/sửa | 1/ Tên phim: | Hiện thông báo không được để trống ô nhập | 1/ Để trống tên phim  2/ Nhấn “Thêm” "/ “Sửa” |
| 3 | Để trống thời lượng khi thêm/sửa | 1/ Thời lượng: 0:0 | Hiện thông báo chưa nhập thời lượng | 1/ Để trống thời lượng  2/ Nhấn “Thêm” "/ “Sửa” |
| 4 | Xóa 1 dòng bất kì | 1/ Chọn dòng bất kỳ trên bảng | Hiển thị thông báo xóa thành công | 1/ Nhấn chọn 1 dòng trên bảng  2/ Nhấn “Xóa” |
| 5 | Nhập từ khóa cần tìm | 1/ txtTim: (từ khóa cần tìm) | Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhập từ khóa cần tìm vào TextField |
| 6 | Chọn ảnh |  | Hiện ảnh tưng ứng lên label | 1/ Double click vào label chứa ảnh  2/ Chọn ảnh |
| 7 | Nhập đúng thông tin khi thêm/sửa | 1/ Tên phim: Conan  2/ Thời lượng: 1:0 | Hiển thị thông báo thêm/cập nhật thành công | 1/ Nhập tên  2/ Nhập thời lượng  3/ Nhấn “Thêm”/ ”Sửa” |

## kiểm thử form qlprofileframe

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Miêu tả tình huống** | **Dữ liệu vào** | **Kết quả trông đợi** | **Chi tiết các bước** |
| 1 | Nhập tên tài khoản không đúng định dạng | 1/ Nhập thông tin  - Tài khoản: hoa  - Tài khoản: Hoathien  2/ Họ tên: Hoa Thiên  3/ Số điện thoại: 09174385938  4/ Email: [abc@gmail.com](mailto:abc@gmail.com)  5/ Ngày sinh: 1/12/1992  6/ Giới tính: Nam/Nữ | **TH1:** Hiện thông báo tài khoản phải từ 6 – 15 ký tự  **TH2:** Hiện thông báo tài khoản không chứa ký tự in hoa và ký tự đặc biệt | 1/ Nhập sai định dạng tên tài khoản  2/ Nhập họ tên  3/ Nhập số điện thoại  4/ Nhập email  5/ Nhập ngày sinh  6/ Chọn giới tính  7/ Nhấn “Cập nhật” |
| 2 | Nhập họ tên không đúng định | 1/ Tên tài khoản: hoathien  2/ Họ tên: Hoa  3/ Số điện thoại: 09174385938  4/ Email: [abc@gmail.com](mailto:abc@gmail.com)  5/ Ngày sinh: 1/12/1992  6/ Giới tính: Nam/Nữ | Hiện thông báo họ tên phải từ 3 – 50 ký tự | 1/ Nhập tài khoản  2/ Nhập sai định dạng họ tên  3/ Nhập số điện thoại  4/ Nhập email  5/ Nhập ngày sinh  6/ Chọn giới tính  7/ Nhấn “Cập nhật” |
| 3 | Nhập điện thoại không đúng định dạng | 1/ Họ tên: Hoa Thiên  2/ Số điện thoại: 09174385  3/ Email: [abc@gmail.com](mailto:abc@gmail.com)  4/ Ngày sinh: 1/12/1992  5/ Giới tính: Nam/Nữ | Hiện thông báo số điện thoại phải từ 9 – 10 số | 1/ Nhập tài khoản  2/ Nhập tên họ tên  3/ Nhập sai định dạng số điện thoại  4/ Nhập email  5/ Nhập ngày sinh  6/ Chọn giới tính  7/ Nhấn “Cập nhật” |
| 4 | Nhập email không đúng định dạng | 1/ Tên tài khoản: hoathien  2/ Họ tên: Hoa Thiên  3/ Số điện thoại: 09174385938  4/ Email: [abcgmail.com](mailto:abc@gmail.com)  5/ Ngày sinh: 1/12/1992  6/ Giới tính: Nam/Nữ | Hiện thông báo email không đúng định dạng | 1/ Nhập tài khoản  2/ Nhập tên họ tên  3/ Nhập số điện thoại  4/ Nhập sai định dạng email  5/ Nhập ngày sinh  6/ Chọn giới tính  7/ Nhấn “Cập nhật” |
| 5 | Nhập ngày sinh sai định dạng | 1/ Tên tài khoản: hoathien  2/ Họ tên: Hoa Thiên  3/ Số điện thoại: 09174385938  4/ Email: [abc@gmail.com](mailto:abc@gmail.com)  5/ Ngày sinh: 1/12/2012  6/ Giới tính: Nam/Nữ | Hiện thông báo ngày sinh phải từ 16 – 65 tuổi | 1/ Nhập tài khoản  2/ Nhập tên họ tên  3/ Nhập số điện thoại  4/ Nhập email  5/ Nhập sai định dạng ngày sinh  6/ Chọn giới tính  7/ Nhấn “Cập nhật” |
| **PHẦN ĐỔI MẬT KHẨU** | | | | |
| 6 | Xóa trắng ô nhập |  | Xóa trắng các TextField | 1/ Nhấn “Reset” |
| 7 | Để trống các ô nhập | 1/ Mật khẩu cũ:  2/ Mật khẩu mới:  3/ Xác nhận mật khẩu: | 1/ Hiện thông báo không để trống ô nhập trên “lblErrorMess”  2/ Khóa nút “Đổi mật khẩu” |  |
| 8 | Nhập sai mật khẩu cũ | 1/ Mật khẩu cũ: Abcd12  2/ Mật khẩu mới: Edf123  3/ Xác nhận mật khẩu: Edf123 | 1/ Hiện thông mật khẩu cũ không đúng trên “lblErrorMess”  2/ Khóa nút “Đổi mật khẩu” | 1/ Nhập sai mật khẩu cũ  2/ Nhập mật khẩu mới  3/ Nhập xác nhận mật khẩu |
| 9 | Nhập sai định dạng mật khẩu mới | 1/ Mật khẩu cũ: Abcd123  2/ Nhập thông tin:  Mật khẩu: hoa  - Mật khẩu: hoathien/ Hoathien/ hoaThien1  3/ Xác nhận mật khẩu: Edf123 | 1/  **- TH1:** Hiện thông báo tài khoản phải từ 6 – 10 ký tự trên “lblErrorMess”  **- TH2:** Hiện thông báo mật khẩu phải ít nhất 1 chữ hoa, thường và số trên “lblErrorMess”  2/ Khóa nút “Đổi mật khẩu” | 1/ Nhập mật khẩu cũ  2/ Nhập sai mật khẩu mới  3/ Nhập xác nhận mật khẩu |
| 10 | Nhập sai xác nhận mật khẩu | 1/ Mật khẩu cũ: Abcd12  2/ Mật khẩu mới: Edf123  3/ Xác nhận mật khẩu: Edf12 | 1/ Hiện thông báo xác nhận mật khẩu không trùng khớp trên “lblErrorMess”  2/ Khóa nút “Đổi mật khẩu” | 1/ Nhập mật khẩu cũ  2/ Nhập mật khẩu mới  3/ Nhập sai xác nhận mật khẩu |
| 11 | Nhập đúng thông tin | 1/ Mật khẩu cũ: Abcd12  2/ Mật khẩu mới: Edf123  3/ Xác nhận mật khẩu: Edf123 | Hiện thông báo đổi mật khẩu thành công | 1/ Nhập mật khẩu cũ  2/ Nhập mật khẩu mới  3/ Nhập xác nhận mật khẩu  4/ Nhấn “Đổi mật khẩu” |

## kiểm lỗi tracuulichchieuframe

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Miêu tả tình huống** | **Dữ liệu vào** | **Kết quả trông đợi** | **Chi tiết các bước** |
| 1 | Nhấn chọn các nút di chuyển |  | 1/ Hiển thị chi tiết dòng được chọn lên form  2/ Khóa nút “First”/”Last” khi dòng được chọn ở đầu/cuối bảng | 1/ Nhấn chọn các nút di chuyển |
| 2 | Lọc theo ngày | 1/ Nhấn “Lọc theo ngày”  2/ Chọn ngày | 1/ Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhấn “Lọc theo ngày”  2/ Chọn ngày |
| 3 | btnChangeState: Text “Cập nhật xuất chiếu” |  | Ẩn nút “Xóa” và “Sửa” |  |
| 4 | btnChangeState: Text “Thêm xuất chiếu” |  | Ẩn nút “Thêm” và “Mới” |  |

## kiểm lỗi thongkeframe

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Miêu tả tình huống** | **Dữ liệu vào** | **Kết quả trông đợi** | **Chi tiết các bước** |
| **Bộ lọc các tab Doanh thu tổng, phim, đồ ăn, phòng vé và Lượng vé đã bán** | | | | |
| 1 | Lọc theo ngày | 1/ Nhấn “Theo ngày” | 1/ Khóa các lựa chọn ”Theo tháng”/ ”Theo năm”  2/ Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhấn “Theo ngày”  2/ Chọn ngày bắt đầu  3/ Chọn ngày kết thúc |
| 2 | Lọc theo tháng | 1/ Nhấn “Theo tháng” | 1/ Khóa các lựa chọn ”Theo ngày”/ ”Theo năm”  2/ Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhấn “Theo tháng”  2/ Chọn tháng - năm |
| 3 | Lọc theo năm | 1/ Nhấn “Theo năm” | 1/ Khóa các lựa chọn ”Theo ngày”/ ”Theo tháng”  2/ Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhấn “Theo năm”  2/ Chọn năm |
|  | | | | |
| 4 | Lọc theo tên đồ ăn | 1/ Nhấn “Theo tên đồ ăn” | 1/ Khóa các lựa chọn ”Theo ngày”/ ”Theo tháng”/ “Theo năm”/ “Theo Size”  2/ Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhấn “Theo tên đồ ăn”  2/ Chọn tên |
| 5 | Lọc theo size đồ ăn | 1/ Nhấn “Theo Size” | 1/ Khóa các lựa chọn ”Theo ngày”/ ”Theo tháng”/ “Theo năm”/ “Theo tên đồ ăn”  2/ Hiển thị dữ liệu tương ứng lên bảng | 1/ Nhấn “Theo Size”  2/ Chọn size |
| 6 | Thống kê khung giờ | 1/ Chọn ngày  2/ Chọn lựa khung giờ | Hiện dữ liệu tương ứng lên bảng |  |

# Đóng gói và triển khai

## Hướng dẫn chuyển đổi jar thành exe

* Sử dụng phần mềm EXE4J để chuyển đổi jar sang exe
* Sử dụng phần mềm InnoSetup để đóng gói

## Hướng dẫn cài đặt triển khai

* Cài SQL Server 2008 trở lên
* Tạo CSDL bằng cách chạy file CSDL cinemaboost\_setup\_db.sql
* Cài đặt JDK 1.8 trở lên
* Chạy file setup

## Hướng dẫn sử dụng phần mềm

* Đăng nhập với
* Tài khoản quản lý
* Tài khoản của nhân viên
* Sử dụng phần mềm theo từng vai trò.