



HỌC VIỆN NGÂN HÀNG
KHOA HỆ THỐNG THÔNG TIN QUẢN LÝ

BÀI TẬP KẾT THÚC HỌC PHẦN **CƠ SỞ LẬP TRÌNH II - IS11A**



ĐỀ TÀI:
PHẦN MỀM QUẢN LÝ
RẠP CHIẾU PHIM



NHÓM THỰC HIỆN: 12
GIẢNG VIÊN HƯỚNG DẪN: TRẦN THỊ HUẾ

Hà Nội, 2020

DANH SÁCH THÀNH VIÊN NHÓM 12

STT	Họ và tên	Mã sinh viên	Tỷ lệ đóng góp
1	Cao Hữu Phước	21A4040096	20%
2	Nguyễn Hồng Phong	21A4040114	20%
3	Phạm Hồng Minh	21A4040112	20%
4	Vũ Văn Thắng	20A4040162	20%
5	Nguyễn Bá Hiếu	21A4040126	20%

Giảng viên hướng dẫn: Trần Thị Huế

Tên đề tài: Quản lý rạp chiếu phim

Kết quả đạt được của đề tài: Xây dựng chương trình hỗ trợ công tác quản lý rạp chiếu phim

Công cụ phát triển:

- Ngôn ngữ lập trình: C#

- Công cụ sử dụng: Visual Studio 2019, Microsoft SQL Server 2008

MỤC LỤC

CHƯƠNG 1: MÔ TẢ HỆ THỐNG	1
1. Mô tả bài toán.....	1
2. Thiết kế chức năng của hệ thống.....	2
3. Sơ đồ liên kết quan hệ các bảng.....	4
CHƯƠNG 2: GIẢI QUYẾT VẤN ĐỀ	5
2.1. Trường Tổng thu, Số phòng, Tổng số ghế, Số vé đã bán, Tổng tiền trong ban đầu có giá trị = 0, không được nhập mà tự động cập nhật.....	5
2.2. Khi cập nhập phòng chiếu thì trường Số phòng và Tổng số ghế trong bảng Rạp được cập nhật theo	5
2.3. Khi cập nhập Phòng chiếu - Ghế thì trường Tổng số ghế trong bảng Phòng chiếu được cập nhật theo.	7
2.4. Khi cập nhập lịch chiếu, kiểm tra các điều kiện	8
a) Mã phòng chỉ hiển thị danh sách các phòng của rạp tương ứng.....	8
b) Mã phòng của một rạp không được chiếu cùng lịch	9
2.5. Khi cập nhập bảng Vé, kiểm tra các điều kiện:	10
a) Mã ghế chỉ hiển thị danh sách các ghế còn trống của lịch chiếu đó	10
b) Các trường Số vé đã bán, Tổng tiền trong bảng Lịch chiếu; trường tổng thu trong bảng Phim được cập nhật theo.....	11
2.6. Tìm kiếm các phim theo: tên, loại phim, hãng sản xuất	12
2.7. Tìm kiếm các lịch chiếu theo: mã rạp, mã phim.....	16
2.8. Báo cáo doanh thu của một bộ phim bất kỳ.....	18
2.9. Báo cáo doanh thu của một rạp bất kỳ	22
2.10. In ra danh sách phim đang chiếu.....	26

CHƯƠNG 1: MÔ TẢ HỆ THỐNG

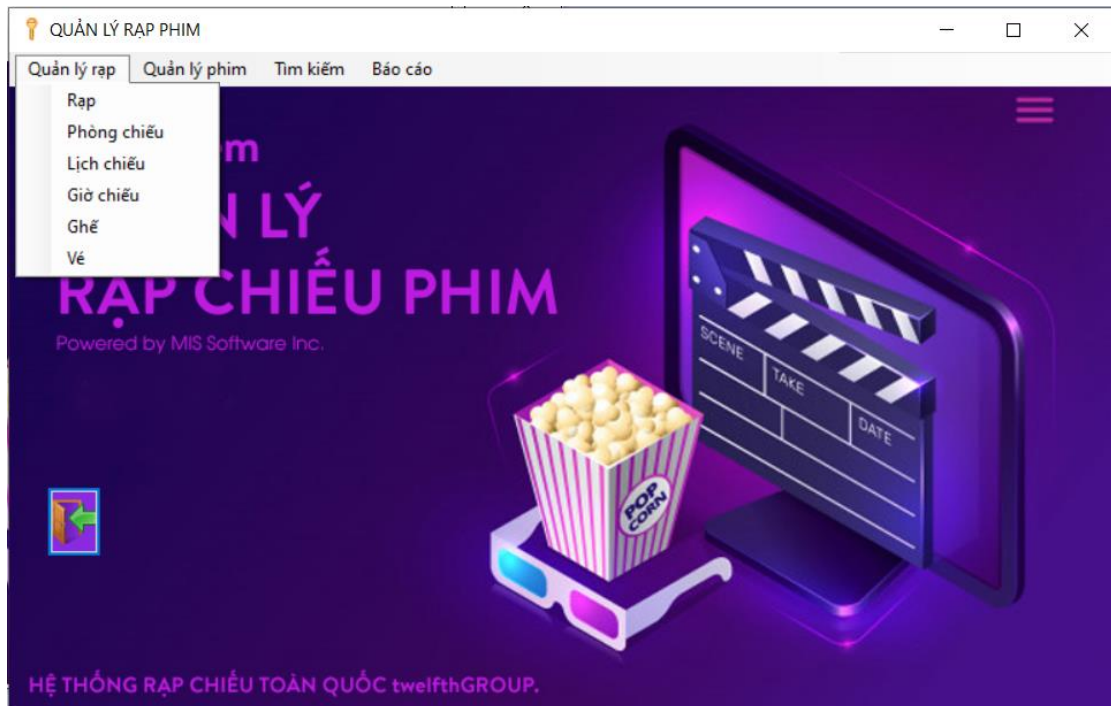
1. Mô tả bài toán

Từ cơ sở dữ liệu ta suy ra nghiệp vụ của bài toán như sau:

- Cơ sở dữ liệu quản lý các rạp trong hệ thống của một công ty. Mỗi rạp trong cơ sở dữ liệu sẽ bao gồm Mã rạp, Tên rạp, Địa chỉ, Điện thoại, Số phòng, Tổng số ghế.
- Mỗi rạp sẽ có nhiều phòng chiếu vì ở bảng phòng chiếu có mã rạp. Mỗi phòng chiếu sẽ có Mã phòng, Tên phòng và Tổng số ghế.
- Cơ sở dữ liệu cũng quản lý các bộ phim, các bộ phim được chiếu trên toàn bộ hệ thống của rạp. Mỗi bộ phim sẽ gồm Tên phim, Mã nước sản xuất, Mã hãng sản xuất, Đạo diễn, Mã thể loại, Ngày khởi chiếu, Ngày kết thúc, Nữ diễn viên chính, Nam diễn viên chính, Nội dung chính, Tổng chi phí, Tổng thu, Ảnh. Các bộ phim có lịch chiếu riêng theo từng rạp
- Các thông tin của phim về Hãng sản xuất, Nước sản xuất, Thể loại đều có bảng riêng gồm Mã và Tên.
- Mỗi rạp cũng có lịch chiếu riêng và lịch chiếu sẽ có mã show. Mỗi show sẽ là một bộ phim được chiếu tại rạp đó với mã phim, mã phòng chiếu, mã rạp và mã giờ chiếu theo từng bộ với giờ chiếu khác nhau trong bảng Giờ chiếu.

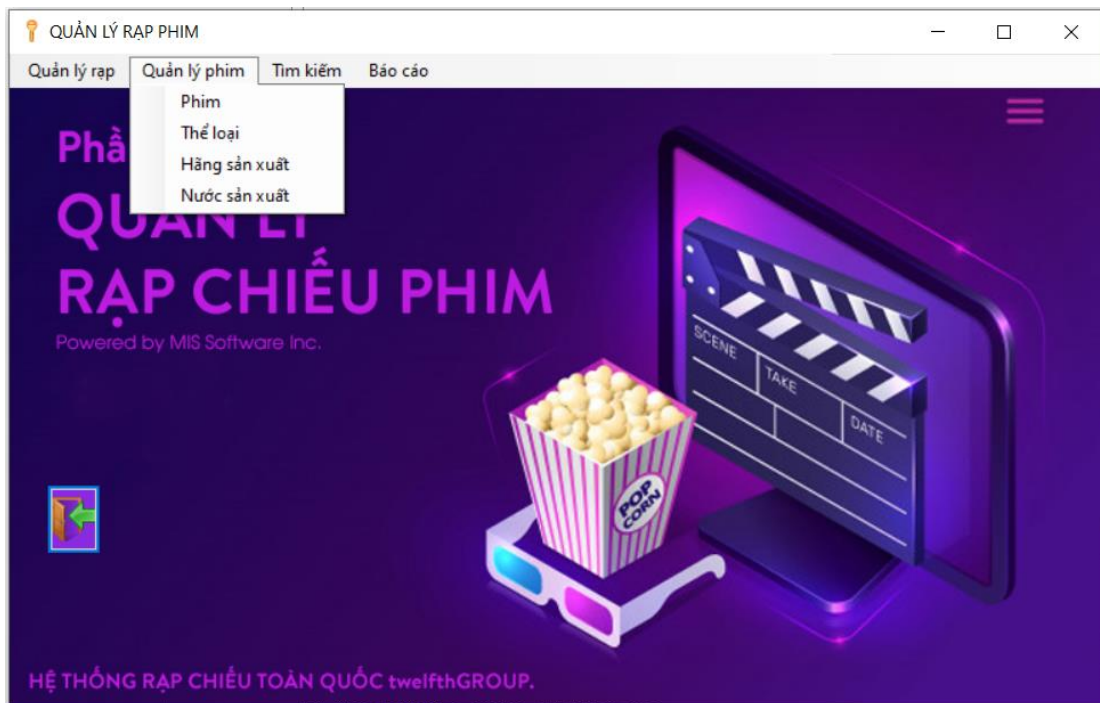
Từ đó suy ra hệ thống có chức năng quản lý các danh mục, tạo lịch chiếu và in các báo cáo doanh thu.

2. Thiết kế chức năng của hệ thống

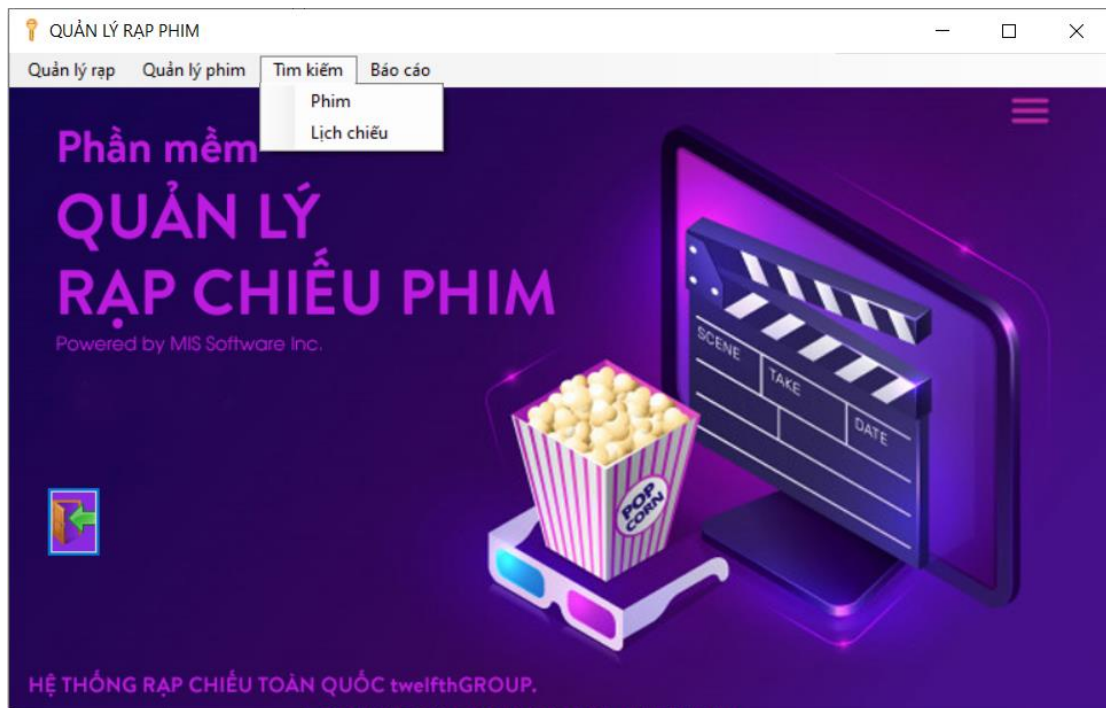


Ở frmMain thì có các danh mục Quản lý rạp, Quản lý phim, Tìm kiếm và Báo cáo là các mục cấp 1.

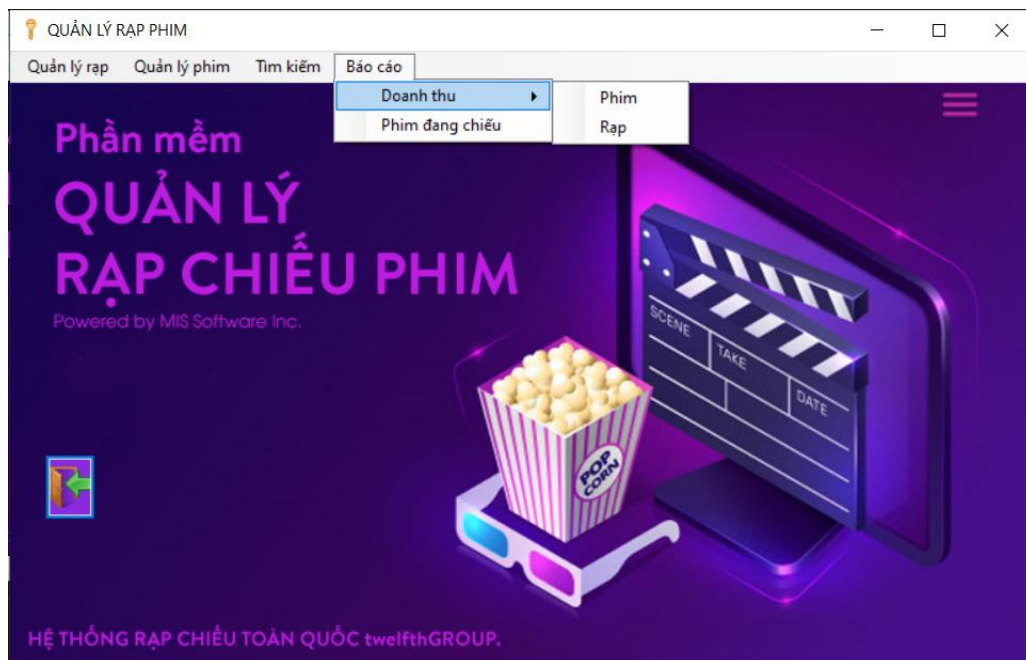
Trong Quản lý rạp thì ta có các mục Rạp, Phòng chiếu, Lịch chiếu, Giờ chiếu, Ghế, Vé là các mục cấp 2.



Ở trong Quản lý phim sẽ bao gồm Phim, Thẻ loại, Hãng sản xuất, Nước sản xuất là các mục cấp 2.

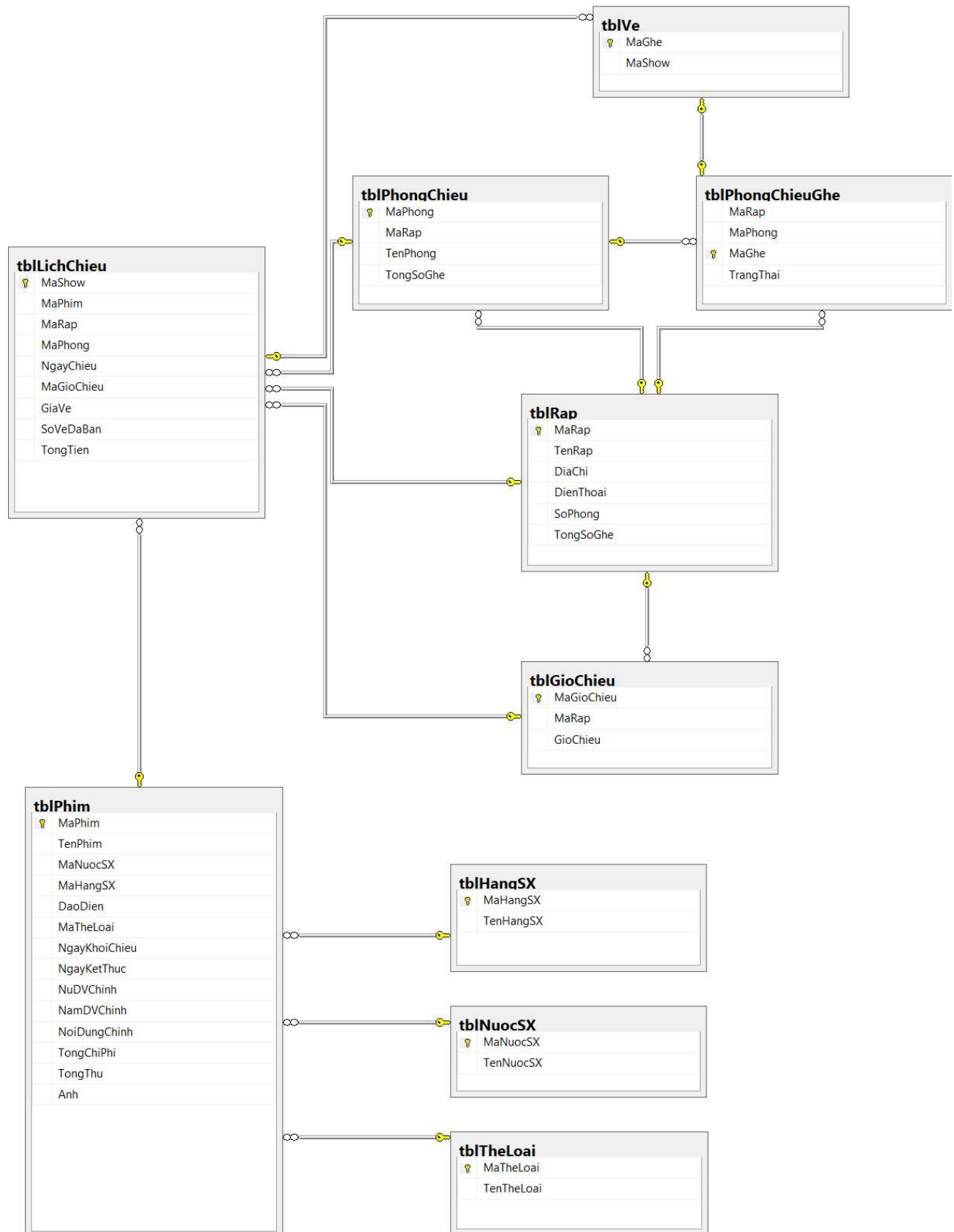


Trong mục Tìm kiếm thì có Phim và Lịch chiếu, mục đích là để tìm kiếm Phim hoặc tìm kiếm Lịch chiếu.



Cuối cùng là danh mục Báo cáo sẽ bao gồm danh mục Doanh thu và Phim đang chiếu ở cấp 2. Trong danh mục Doanh thu sẽ bao gồm 2 danh mục cấp 3 là Phim và Rạp

3. Sơ đồ liên kết quan hệ các bảng



CHƯƠNG 2: GIẢI QUYẾT VẤN ĐỀ

2.1. Trường Tổng thu, Số phòng, Tổng số ghế, Số vé đã bán, Tổng tiền trong ban đầu có giá trị = 0, không được nhập mà tự động cập nhật

- Giải quyết: Khi tạo cơ sở dữ liệu trong SQL, chúng ta để giá trị mặc định default bằng 0 và khi tạo các Form Quản lý danh mục, sẽ không tạo Textbox cho các trường trên. Tự động cập nhật sẽ được giải quyết tại các phần sau đây.

2.2. Khi cập nhật phòng chiếu thì trường Số phòng và Tổng số ghế trong bảng Rạp được cập nhật theo

- Form liên quan: Phòng chiếu, Rạp và chức năng liên quan gồm Thêm, Sửa, Xóa, Lưu
- Ứng dụng: Khi người dùng sử dụng các nút để thêm, bớt bản ghi hay thay đổi Số ghế của bản ghi trong bảng Phòng chiếu thì đồng thời Số phòng và Tổng số ghế của bản ghi có mã rạp tương ứng trong bảng Rạp cũng tự động thay đổi theo.
- Ý tưởng: Tạo các trigger cho database trong SQL đối với bảng tblPhongChieu và có chức năng cập nhật Số phòng và Tổng số ghế trong bảng Rạp bằng với số lượng có Mã rạp tương ứng. Trigger hoạt động theo cách: Mỗi khi người dùng insert, delete, update bản ghi của bảng Phòng chiếu (thông qua các nút Sửa, Xóa, Lưu) thì hệ thống sẽ đếm lại số phòng theo Mã rạp của các bản ghi liên quan và tổng số ghế bằng cách sử dụng hàm.
 - o Các bước tiến hành
 - Bước 1: Tạo trigger trong SQL cho các hành động insert, delete và update tại bảng Phòng chiếu
 - Bước 2: Viết câu lệnh đếm số lượng phòng bằng hàm COUNT, tính tổng số ghế bằng hàm SUM tại các Phòng chiếu của một Rạp.
 - Bước 3: Kiểm tra độ chính xác của trigger. Kết thúc.

- Sơ đồ giải quyết

– Code xử lý

```
-- Cập nhật tổng số phòng một rạp khi có phòng mới
create trigger SoPhongRap
on tblPhongChieu after insert as
begin
update tblRap
set SoPhong = SoPhong +( select count(MaPhong) from inserted where MaRap= tblRap.MaRap) from
tblRap join inserted on tblRap.MaRap =inserted.MaRap
end
-- Cập nhật tổng số ghế một rạp khi có phòng mới
create trigger TongSoGheRap on tblPhongChieu after insert as
begin
update tblRap
set TongSoGhe = tblRap.TongSoGhe +(select TongSoGhe from inserted where MaRap =
tblRap.MaRap) from tblRap join inserted on tblRap.MaRap=inserted.MaRap
end
-- Cập nhật tổng số phòng một rạp sau khi một phòng bị xóa
create trigger XoaSoPhongRap
on tblPhongChieu for delete as
begin
update tblRap
set SoPhong = SoPhong -( select count(MaPhong) from deleted where MaRap= tblRap.MaRap) from
tblRap join deleted on tblRap.MaRap =deleted.MaRap
end
-- Cập nhật tổng số ghế một rạp sau khi một phòng bị xóa
create trigger XoaTongSoGheRap on tblPhongChieu for delete as
begin
update tblRap
set TongSoGhe = tblRap.TongSoGhe -(select TongSoGhe from deleted where MaRap =tblRap.MaRap)
from tblRap join deleted on tblRap.MaRap =deleted.MaRap
end
-- Cập nhật tổng số ghế một rạp sau khi sửa số liệu một phòng
create trigger SuaTongSoGhe on tblPhongChieu after update as
begin
update tblRap
set TongSoGhe = tblRap.TongSoGhe +(select TongSoGhe from inserted where MaRap =tblRap.MaRap
) - (select TongSoGhe from deleted where MaRap =tblRap.MaRap) from tblRap join deleted on
tblRap.MaRap =deleted.MaRap
end
```

2.3. Khi cập nhật Phòng chiếu - Ghế thì trường Tổng số ghế trong bảng Phòng chiếu được cập nhật theo.

- Form liên quan: Phòng chiếu – Ghế, Phòng chiếu và chức năng liên quan gồm Thêm, Sửa, Xóa, Lưu
- Ứng dụng: Khi người dùng sử dụng các nút để thêm, bớt bản ghi hay thay đổi Ghế của bản ghi trong bảng Phòng chiếu – Ghế thì đồng thời Tổng số ghế của bản ghi có Mã phòng chiếu tương ứng trong bảng Phòng chiếu cũng tự động thay đổi theo.
- Ý tưởng: Tạo các trigger cho database trong SQL đối với bảng tblPhongChieuGhe và có chức năng cập nhật Tổng số ghế trong bảng Phòng chiếu bằng với số lượng có Mã phòng chiếu tương ứng. Trigger hoạt động theo cách: Mỗi khi người dùng insert,

delete, update bản ghi của bảng Phòng chiếu – Ghế(thông qua các nút Sửa, Xóa, Lưu) thì hệ thống sẽ đếm lại số ghế theo Mã phòng chiếu của các bản ghi liên quan và tổng số ghế bằng cách sử dụng hàm.

- Các bước tiến hành
- Bước 1: Tạo trigger trong SQL cho các hành động insert, delete tại bảng Phòng chiếu – Ghế. Không cần làm câu lệnh update thì khi sửa 1 bản ghi, số lượng vẫn giữ nguyên.
- Bước 2: Viết câu lệnh Tổng số ghế cũ cộng/trừ số lượng ghế được thêm (đếm bằng hàm COUNT) tại Phòng chiếu của một Rạp.
- Bước 3: Kiểm tra độ chính xác của trigger. Kết thúc.

- Code xử lý yêu cầu:

```
-- Cập nhật tổng số ghế một phòng sau khi một ghế được thêm
create trigger TongSoGhePC on tblPhongChieuGhe after insert as
begin
update tblPhongChieu
set TongSoGhe = tblPhongChieu.TongSoGhe +(select count (MaGhe) from inserted where MaPhong
=tblPhongChieu.MaPhong) from tblPhongChieu join inserted on
tblPhongChieu.MaPhong=inserted.MaPhong
end

-- Cập nhật tổng số ghế một phòng sau khi một ghế bị xóa
create trigger XoaTongSoGhePC on tblPhongChieuGhe for delete as
begin
update tblPhongChieu
set TongSoGhe = tblPhongChieu.TongSoGhe -(select count (MaGhe) from deleted where MaPhong
=tblPhongChieu.MaPhong) from tblPhongChieu join deleted on tblPhongChieu.MaPhong
=deleted.MaPhong
end
```

2.4. Khi cập nhật lịch chiếu, kiểm tra các điều kiện

a) Mã phòng chỉ hiển thị danh sách các phòng của rạp tương ứng

- Form liên quan: Lịch chiếu
- Ứng dụng: Khi người dùng sử dụng danh mục quản lý lịch chiếu, các phòng sẽ được hiển thị tương ứng với rạp của nó, đảm bảo tính thống nhất và logic
- Ý tưởng: Khi đổ dữ liệu vào Combo Box của Mã phòng, ta sẽ dùng event Selecte Index Changed của Combo Box Mã rạp để chỉ hiển thị các Mã phòng tương ứng
- Các bước tiến hành

- Bước 1: Tạo Form Lịch chiếu với Combo Box là các khóa ngoại, trong đó bao gồm Mã phòng và Mã rạp.
 - Bước 2: Kích hoạt event Selecte Index Changed của Combo Box Mã rạp và dùng code xử lý phù hợp.
 - Bước 3: Kiểm tra và kết thúc
- Sơ đồ giải quyết:
- Code xử lý yêu cầu (C#: Tại đây nhóm có thực hiện thêm chỉ hiển thị giờ chiếu tương ứng với Rạp để đảm bảo sự thống nhất)

```
private void cboMarap_SelectedIndexChanged(object sender, EventArgs e)
{
    if (cboMarap.SelectedIndex != -1)
    {
        string str;
        str = "select distinct MaPhong, TenPhong from tblPhongChieu where MaRap = '"
+ cboMarap.SelectedValue + "'";
        // MessageBox.Show(str);
        DAO.FillDataToCombo(str, cboMaphong, "MaPhong", "TenPhong");
    }
    {
        string str1;
        str1 = "select distinct MaGioChieu, GioChieu from tblGioChieu where MaRap = '"
+ cboMarap.SelectedValue + "'";
        // MessageBox.Show(str);
        DAO.FillDataToCombo(str1, cboMagiochieu, "MaGioChieu", "GioChieu");
    }
}
```

b) Mã phòng của một rạp không được chiếu cùng lịch

- Yêu cầu liên quan đến form Lịch chiếu
- Dữ kiện đầu vào: nhập đầy đủ thông tin form lịch chiếu.
 - Điều kiện cần kiểm tra: không tồn tại trùng lặp dữ liệu của các thuộc tính sau: Mã phim, Mã phòng, Mã Giờ chiếu, Ngày chiếu
 - Kết quả trả về: lưu CSDL vào bảng Lịch chiếu thành công.
- Sơ đồ giải quyết:

- Code xử lý:

```
//1 phòng không được chiếu trùng lịch
//Kiểm tra Phim, Phòng, Giờ chiếu, Ngày chiếu có trùng?
```

```

        DAO.OpenConnection();
        string sql = "SELECT MaPhim, MaPhong, MaGioChieu, NgayChieu FROM tblLichChieu WHERE
MaPhim=N'" + cboMaphim.SelectedValue.ToString() + "' AND MaPhong=N'" +
cboMaphong.SelectedValue.ToString() + "' AND MaGioChieu=N'" +
cboMagiochieu.SelectedValue.ToString() + "' AND NgayChieu=N'" +
Convert.ToDateTime(dateNgaychieu.Value).ToString() + "'";
        if (DAO.checkKeyExit(sql) == true)
        {
            MessageBox.Show("Phim '" + cboMaphim.SelectedValue.ToString() + "' đã có
lich chiếu tại phòng vào giờ này!", "Thông báo");
            return;
        }

```

2.5. Khi cập nhật bảng Vé, kiểm tra các điều kiện:

a) Mã ghế chỉ hiển thị danh sách các ghế còn trống của lịch chiếu đó

- Form liên quan: Vé
- Ứng dụng: Khi người dùng sử dụng danh mục quản lý vé, các ghế còn trống sẽ được hiển thị tương ứng với rạp của nó, tránh tình trạng đặt vé đã đặt.
- Ý tưởng: Khi đổ dữ liệu vào Combo Box của Mã ghế, ta sẽ chỉ hiển thị các Mã ghế có Trạng thái tương ứng là “Trống”
- Sơ đồ giải quyết:
 - Code xử lý yêu cầu:
 - o Trước tiên ta sẽ tạo trigger để ghi thêm hoặc xóa tại bảng Vé, cột Trạng thái sẽ tự động được cập nhật là “Đã đặt” hoặc “Trống”

```

Create trigger DatVe on tblVe AFTER INSERT AS
BEGIN UPDATE tblPhongChieuGhe
SET TrangThai= N'Đã Đặt'
From tblPhongChieuGhe join inserted on tblPhongChieuGhe.MaGhe = inserted.MaGhe
where tblPhongChieuGhe.MaGhe = inserted.MaGhe
END
Create trigger HuyVe on tblVe AFTER DELETE AS
BEGIN UPDATE tblPhongChieuGhe
SET TrangThai= N'Trống'
From tblPhongChieuGhe join deleted on tblPhongChieuGhe.MaGhe = deleted.MaGhe
where tblPhongChieuGhe.MaGhe = deleted.MaGhe
END

```

- Sau đó là code C# với combobox

```
private void FrmVe_Load(object sender, EventArgs e)
{
    DAO.OpenConnection();
    LoadDataToGridView();
    DAO.FillDataToCombo("SELECT MaShow FROM tblLichChieu", cboMashow, "MaShow",
        "MaShow");
    DAO.FillDataToCombo("SELECT MaGhe FROM tblPhongChieuGhe where TrangThai =
        N'Trống'", cboGhe, "MaGhe", "MaGhe");
    btnLuu.Enabled = false;
    DAO.CloseConnection();
}
```

b) Các trường Số vé đã bán, Tổng tiền trong bảng Lịch chiếu; trường tổng thu trong bảng Phim được cập nhật theo.

- Form liên quan: Vé, Lịch chiếu, Phim và chức năng liên quan gồm Thêm, Sửa, Xóa, Lưu
- Ứng dụng: Khi người dùng sử dụng các nút để thêm, bớt bản ghi hay thay đổi bản ghi trong bảng Vé thì đồng thời Số vé đã bán và Tổng tiền trong bảng Lịch chiếu có mã show tương ứng thay đổi theo, kéo theo đó là Tổng thu của Phim có mã phim tương ứng thay đổi.
- Ý tưởng: Tạo các trigger cho database trong SQL đối với bảng tblVé, tblLichChieu và có chức năng cập nhật Số vé đã bán, Tổng tiền và Tổng thu. Trigger hoạt động theo cách: Mỗi khi người dùng insert, delete, update bản ghi của bảng Vé (thông qua các nút Sửa, Xóa, Lưu) thì Số vé đã bán sẽ tự động tăng hoặc giảm theo số lượng thêm/xóa; Tổng tiền = Số vé đã bán * Giá vé được cập nhật ngay khi bảng Lịch chiếu được update Số vé đã bán. Đồng thời Tổng thu của một phim bằng tổng Tổng tiền của một Show trong Lịch chiếu

– Sơ đồ giải quyết:

- Code xử lý yêu cầu (Ở đây chúng ta không cần cập nhật tổng thu khi có một lịch chiếu mới được thêm, bởi quy tắc ở đây Tổng tiền của một show sẽ không được thêm khi thêm mới mà tự động cập nhật)

```
-- Cập nhật tổng tiền khi có thêm vé được bán
Create trigger TongTienist on tblVe AFTER INSERT AS
BEGIN UPDATE tblLichChieu
SET TongTien = (GiaVe * SoVeDaBan) FROM tblLichChieu join inserted on
tblLichChieu.MaShow = inserted.MaShow
WHERE tblLichChieu.MaShow = inserted.MaShow
END

-- Cập nhật tổng tiền khi có vé bị hủy
Create trigger TongTiendlt on tblVe AFTER DELETE AS
BEGIN UPDATE tblLichChieu
SET TongTien = (GiaVe * SoVeDaBan) FROM tblLichChieu join deleted on
tblLichChieu.MaShow = deleted.MaShow
WHERE tblLichChieu.MaShow = deleted.MaShow
END

-- Cập nhật tổng tiền khi có vé thay đổi phim
Create trigger TongTienudt on tblVe AFTER UPDATE AS
BEGIN UPDATE tblLichChieu
SET TongTien = (GiaVe * SoVeDaBan) FROM tblLichChieu join inserted on
tblLichChieu.MaShow = inserted.MaShow
WHERE tblLichChieu.MaShow = inserted.MaShow
END

-- Cập nhật tổng thu khi show được cập nhật
Create trigger TongThuupdate on tblLichChieu AFTER UPDATE AS
BEGIN UPDATE tblPhim
SET TongThu = (select SUM(TongTien) from tblPhim join inserted on tblPhim.MaPhim =
inserted.MaPhim
GROUP BY tblPhim.MaPhim)
END

-- Cập nhật tổng tiền khi có một show bị hủy
Create trigger TongThudelete on tblLichChieu AFTER DELETE AS
BEGIN UPDATE tblPhim
SET TongThu = (select SUM(TongTien) from tblPhim join deleted on tblPhim.MaPhim =
deleted.MaPhim
GROUP BY tblPhim.MaPhim)
END
```

2.6. Tìm kiếm các phim theo: tên, loại phim, hãng sản xuất

– Form liên quan: Tìm kiếm phim

- Ứng dụng: Tìm kiếm thêm theo một trong ba tiêu chí: Tên, Loại phim, Hãng sản xuất hoặc cả 3.
- Các bước tiến hành:
 - Bước 1: Tạo form FrmTimKiemPhim với 3 button tìm kiếm riêng cho từng tiêu chí và một button tìm kiếm chung cả 3 tiêu chí.
 - Bước 2: Hoàn thiện code theo yêu cầu đều bài.

– Sơ đồ giải quyết:

– Form Tìm kiếm phim:

– Code xử lí:

```
public FrmTimKiemPhim()
{
    InitializeComponent();
}

private void FrmTimKiemPhim_Load(object sender, EventArgs e)
{
    Load_comboBox();
    ResetValues();
}

private void ResetValues()
{
    cboTheLoai.Text = "";
    cboHangSX.Text = "";
}

private void Load_comboBox()
```

```

{
    DAO.OpenConnection();

    //Load cboTheLoai tìm kiếm
    cboTheLoai.DataSource = DAO.GetDataToTable("SELECT MaTheLoai, TenTheLoai FROM
tblTheLoai");
    cboTheLoai.ValueMember = "MaTheLoai";
    cboTheLoai.DisplayMember = "TenTheLoai";
    cboTheLoai.SelectedIndex = -1;

    //Load cboMaHangSX tìm kiếm
    cboHangSX.DataSource = DAO.GetDataToTable("SELECT MaHangSX, TenHangSX FROM
tblHangSX");
    cboHangSX.ValueMember = "MaHangSX";
    cboHangSX.DisplayMember = "TenHangSX";
    cboHangSX.SelectedIndex = -1;
}

private void btnTimkiem_Click(object sender, EventArgs e)
{
    if (txtTenphim.Text == "")
    {
        MessageBox.Show("Hãy chọn tên Phim cần tìm!", "Thông Báo",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        return;
    }
    if (cboTheLoai.Text == "")
    {
        MessageBox.Show("Hãy chọn tên thể loại cần tìm!", "Thông Báo",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        return;
    }
    if (cboHangSX.Text == "")
    {
        MessageBox.Show("Hãy chọn tên hãng sản xuất cần tìm!", "Thông Báo",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        return;
    }
    else
    {
        DataTable tbl;
        string sql = "SELECT * FROM tblPhim WHERE (TenPhim LIKE '%" +
txtTenphim.Text.Trim() + "%' AND MaTheLoai= N'" + cboTheLoai.SelectedValue + "' AND
MaHangSX = N'" + cboHangSX.SelectedValue + "')";
        tbl = DAO.GetDataToTable(sql);
        GridViewTimKiemPhim.DataSource = tbl;
        GridViewTimKiemPhim.AllowUserToAddRows = false;
        GridViewTimKiemPhim.EditMode = DataGridViewEditMode.EditProgrammatically;
    }
}

private void btnTimkiemphim_Click(object sender, EventArgs e)
{
    if (txtTenphim.Text == "")
    {
        MessageBox.Show("Hãy chọn tên Phim cần tìm!", "Thông Báo",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        return;
    }
    else

```

```

        {
            DataTable tbl;
            string sql = "SELECT * FROM tblPhim WHERE TenPhim LIKE '%" +
txtTenPhim.Text.Trim() + "%'";
            tbl = DAO.GetDataToTable(sql);
            GridViewTimKiemPhim.DataSource = tbl;
            GridViewTimKiemPhim.AllowUserToAddRows = false;
            GridViewTimKiemPhim.EditMode = DataGridViewEditMode.EditProgrammatically;
        }
    }

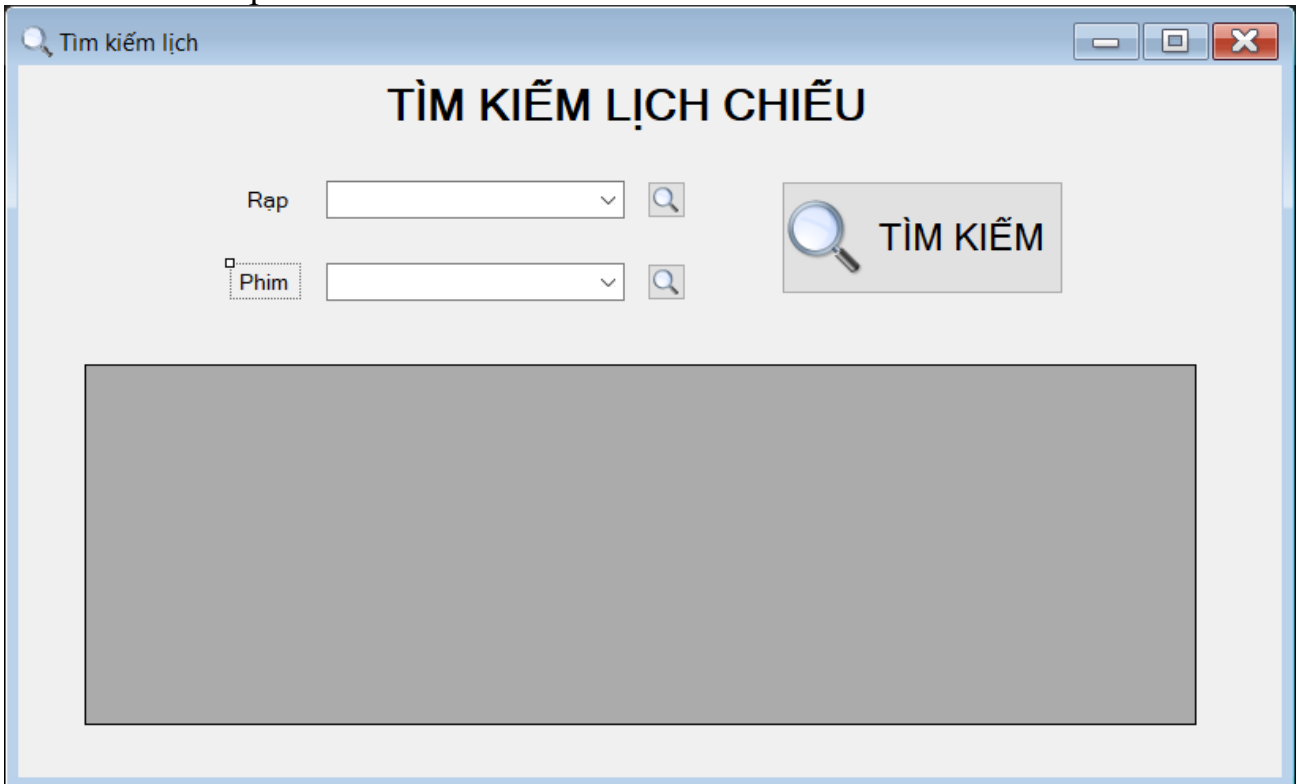
private void btnTimkiemtheloai_Click(object sender, EventArgs e)
{
    if (cboTheLoai.Text == "")
    {
        MessageBox.Show("Hãy chọn tên thể loại cần tìm!", "Thông Báo",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        return;
    }
    else
    {
        DataTable tbl;
        string sql = "SELECT * FROM tblPhim WHERE (MaTheLoai =N'" +
cboTheLoai.SelectedValue + "')";
        tbl = DAO.GetDataToTable(sql);
        GridViewTimKiemPhim.DataSource = tbl;
        GridViewTimKiemPhim.AllowUserToAddRows = false;
        GridViewTimKiemPhim.EditMode = DataGridViewEditMode.EditProgrammatically;
    }
}

private void btnTimkiemhsx_Click(object sender, EventArgs e)
{
    if (cboHangSX.Text == "")
    {
        MessageBox.Show("Hãy chọn tên hãng sản xuất cần tìm!", "Thông Báo",
        MessageBoxButtons.OK, MessageBoxIcon.Information);
        return;
    }
    else
    {
        DataTable tbl;
        string sql = "SELECT * FROM tblPhim WHERE (MaHangSX =N'" +
cboHangSX.SelectedValue + "')";
        tbl = DAO.GetDataToTable(sql);
        GridViewTimKiemPhim.DataSource = tbl;
        GridViewTimKiemPhim.AllowUserToAddRows = false;
        GridViewTimKiemPhim.EditMode = DataGridViewEditMode.EditProgrammatically;
    }
}

```

2.7. Tìm kiếm các lịch chiếu theo: mã rạp, mã phim

- Form liên quan: Tìm kiếm lịch
 - Ứng dụng: Tìm kiếm thêm theo một trong hai tiêu chí: Mã rạp, Mã phim hoặc cả 3.
 - Các bước tiến hành:
 - Bước 1: Tạo form FrmTimKiemLich với 2 button tìm kiếm riêng cho từng tiêu chí và một button tìm kiếm chung cả 2 tiêu chí.
 - Bước 2: Hoàn thiện code theo yêu cầu đề bài.
- Sơ đồ giải quyết:
- Form Tìm kiếm phim:



- Code xử lí:

```
public FrmTimKiemLich()
{
    InitializeComponent();
}
```

```

private void FrmTimKiemLich_Load(object sender, EventArgs e)
{
    Load_comboBox();
    ResetValues();
}
private void ResetValues()
{
    cboRap.Text = "";
    cboPhim.Text = "";
}
private void Load_comboBox()
{
    DAO.OpenConnection();
    //Load cboMaRap tìm kiếm
    cboRap.DataSource = DAO.GetDataToTable("SELECT MaRap, TenRap FROM tblRap");
    cboRap.ValueMember = "MaRap";
    cboRap.DisplayMember = "TenRap";
    cboRap.SelectedIndex = -1;

    //Load cboMaPhim tìm kiếm
    cboPhim.DataSource = DAO.GetDataToTable("SELECT MaPhim, TenPhim FROM tblPhim");
    cboPhim.ValueMember = "MaPhim";
    cboPhim.DisplayMember = "TenPhim";
    cboPhim.SelectedIndex = -1;
}

private void btnTimkiem_Click(object sender, EventArgs e)
{
    if (cboRap.Text == "")
    {
        MessageBox.Show("Chưa chọn mã rạp!", "Thông Báo", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        return;
    }
    if (cboPhim.Text == "")
    {
        MessageBox.Show("Chưa chọn mã phim!", "Thông Báo", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        return;
    }

    DataTable tbl;
    string sql = "SELECT * FROM tblLichChieu WHERE (MaRap =N'"
    + cboRap.SelectedValue + "' AND MaPhim= N'" + cboPhim.SelectedValue + "')";
    tbl = DAO.GetDataToTable(sql);
    GridViewTimKiemLich.DataSource = tbl;
    // Không cho phép thêm mới dữ liệu trực tiếp trên lưới
    GridViewTimKiemLich.AllowUserToAddRows = false;
    // Không cho phép sửa dữ liệu trực tiếp trên lưới
    GridViewTimKiemLichEditMode = DataGridViewEditMode.EditProgrammatically;
}

private void btnTimkiemrap_Click(object sender, EventArgs e)
{
    if (cboRap.Text == "")
    {
        MessageBox.Show("Chưa chọn rạp!", "Thông Báo", MessageBoxButtons.OK,
        MessageBoxIcon.Information);
        return;
    }

```

```

    }
    DataTable tbl;
    string sql = "SELECT * FROM tblLichChieu WHERE (MaRap =N'" +
cboRap.SelectedValue + "')";
    tbl = DAO.GetDataToTable(sql);
    GridViewTimKiemLich.DataSource = tbl;
    // Không cho phép thêm mới dữ liệu trực tiếp trên lưới
    GridViewTimKiemLich.AllowUserToAddRows = false;
    // Không cho phép sửa dữ liệu trực tiếp trên lưới
    GridViewTimKiemLich.EditMode = DataGridViewEditMode.EditProgrammatically;
}

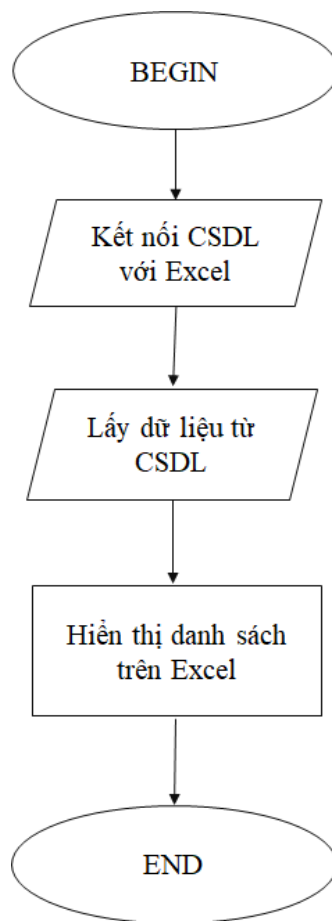
private void btnTimkiemphim_Click(object sender, EventArgs e)
{
    if (cboPhim.Text == "")
    {
        MessageBox.Show("Chưa chọn tên phim!", "Thông Báo", MessageBoxButtons.OK,
MessageBoxIcon.Information);
        return;
    }

    DataTable tbl;
    string sql = "SELECT * FROM tblLichChieu WHERE (MaPhim= N'" +
cboPhim.SelectedValue + "')";
    tbl = DAO.GetDataToTable(sql);
    GridViewTimKiemLich.DataSource = tbl;
    // Không cho phép thêm mới dữ liệu trực tiếp trên lưới
    GridViewTimKiemLich.AllowUserToAddRows = false;
    // Không cho phép sửa dữ liệu trực tiếp trên lưới
    GridViewTimKiemLich.EditMode = DataGridViewEditMode.EditProgrammatically;
}

```

2.8. Báo cáo doanh thu của một bộ phim bất kỳ

- Yêu cầu liên quan đến form In Doanh Thu Phim sử dụng chức năng in
 - Ứng dụng: Sử dụng khi xuất dữ liệu ra file Excel để in.
 - Các bước tiến hành :
 - Bước 1: Kết nối project với Excel.
 - Bước 2: Tạo một combobox mã phim để chọn phim cần in báo cáo. Lấy dữ liệu về doanh thu của một bộ phim từ các bảng tblPhim và tblLichChieu
 - Bước 3: In ra Excel, chỉnh sửa theo ý.
- Sơ đồ giải quyết



– Form In Doanh Thu Phim:

The screenshot shows a software window titled "Doanh thu Phim". The main heading is "BÁO CÁO DOANH THU PHIM". Below the heading, there is a label "Chọn phim" followed by a dropdown menu. At the bottom center, there is a button with a printer icon and the text "In báo cáo".

- Mẫu báo cáo sau khi ấn In

	A	B	C	D	E	F	G	H
1	Tập đoàn twelthGROUP							
2	Hệ thống Rạp chiếu toàn quốc		BÁO CÁO DOANH THU PHIM					
3	Hotline: +8441992809							
4								
5								
6		Mã phim:	MV001					
7		Tên phim:	Inception					
8		Ngày khởi chiếu:	04/02/14 0:00					
9		Ngày kết thúc:	02/01/14 0:00					
10		Số vé đã bán:	1					
11		Doanh thu:	60000					
12								
13								
14								
15								
16								
17								
18								
19								

– Code xử lý:

```
public InDoanhThuPhim()
{
    InitializeComponent();
}

private void InDoanhThuPhim_Load(object sender, EventArgs e)
{
    DAO.OpenConnection();
    DAO.FillDataToCombo("SELECT MaPhim, TenPhim FROM tblPhim", cboPhim, "MaPhim",
"TenPhim");
    DAO.CloseConnection();
}

private void btnInDTP_Click(object sender, EventArgs e)
{
    COMExcel.Application exApp = new COMExcel.Application();
    COMExcel.Workbook exBook; //Trong 1 chương trình Excel có nhiều Workbook
    COMExcel.Worksheet exSheet; //Trong 1 Workbook có nhiều Worksheet
    COMExcel.Range exRange;
    string sql;
    DataTable tblPhim, sove;
    exBook = exApp.Workbooks.Add(COMExcel.XlWBATemplate.xlWBATWorksheet);
    exSheet = exBook.Worksheets[1];
    // Định dạng chung
    exRange = exSheet.Cells[1, 1];
    exRange.Range["A1:B3"].Font.Size = 10;
    exRange.Range["A1:B3"].Font.Name = "Times new roman";
    exRange.Range["A1:B3"].Font.Bold = true;
    exRange.Range["A1:B3"].Font.ColorIndex = 5; //Màu xanh da trời
    exRange.Range["A1:A1"].ColumnWidth = 7;
    exRange.Range["B1:B1"].ColumnWidth = 15;
    exRange.Range["A1:B1"].MergeCells = true;
    exRange.Range["A1:B1"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
    exRange.Range["A1:B1"].Value = "Tập đoàn twelfthGROUP";
    exRange.Range["A2:B2"].MergeCells = true;
    exRange.Range["A2:B2"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
    exRange.Range["A2:B2"].Value = "Hệ thống Rạp chiếu toàn quốc";
    exRange.Range["A3:B3"].MergeCells = true;
    exRange.Range["A3:B3"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
    exRange.Range["A3:B3"].Value = "Hotline: +8441992809";
    exRange.Range["C2:H2"].Font.Size = 16;
    exRange.Range["C2:H2"].Font.Name = "Times new roman";
    exRange.Range["C2:H2"].Font.Bold = true;
    exRange.Range["C2:H2"].Font.ColorIndex = 3; //Màu đỏ
    exRange.Range["C2:H2"].MergeCells = true;
    exRange.Range["C2:H2"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
    exRange.Range["C2:H2"].Value = "BÁO CÁO DOANH THU PHIM";
    sql = "SELECT MaPhim, TenPhim, NgayKhoiChieu, NgayKetThuc, TongThu FROM tblPhim
WHERE MaPhim = N'" + cboPhim.Selected.Value.ToString() + "'";

    tblPhim = DAO.GetDataToTable(sql);
    exRange.Range["B6:C11"].Font.Size = 12;
    exRange.Range["B6:C11"].Font.Name = "Times new roman";
    exRange.Range["B6:B6"].Value = "Mã phim:";
    exRange.Range["C6:E6"].MergeCells = true;
    exRange.Range["C6:E6"].Value = tblPhim.Rows[0][0].ToString();
    exRange.Range["B7:B7"].Value = "Tên phim:";
```

```

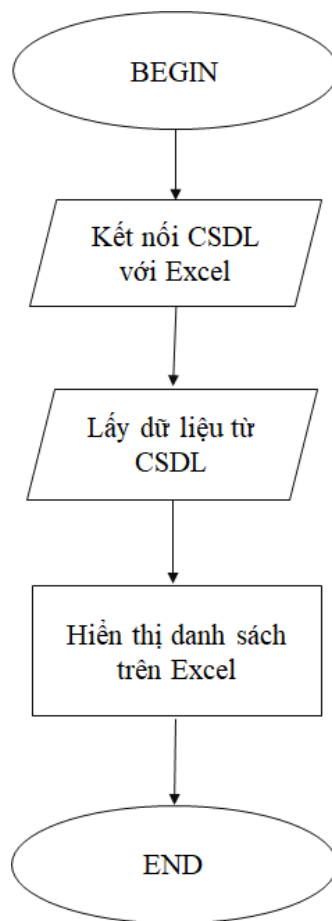
exRange.Range["C7:E7"].MergeCells = true;
exRange.Range["C7:E7"].Value = tblPhim.Rows[0][1].ToString();
exRange.Range["B8:B8"].Value = "Ngày khởi chiếu:";
exRange.Range["C8:E8"].MergeCells = true;
exRange.Range["C8:E8"].Value = tblPhim.Rows[0][2].ToString();
exRange.Range["B9:B9"].Value = "Ngày kết thúc:";
exRange.Range["C9:E9"].MergeCells = true;
exRange.Range["C9:E9"].Value = tblPhim.Rows[0][3].ToString();
exRange.Range["B11:B11"].Value = "Doanh thu:";
exRange.Range["C11:E11"].MergeCells = true;
exRange.Range["C11:E11"].Value = tblPhim.Rows[0][4].ToString();

sql = "SELECT SUM(SoVeDaBan) from tblLichChieu where MaPhim = N'" +
cboPhim.SelectedValue.ToString() + "'";
sove = DAO.GetDataToTable(sql);
exRange.Range["B10:B10"].Value = "Số vé đã bán:";
exRange.Range["C10:E10"].MergeCells = true;
exRange.Range["C10:E10"].Value = sove.Rows[0][0].ToString();
exApp.Visible = true;
}

```

2.9. Báo cáo doanh thu của một rạp bất kỳ

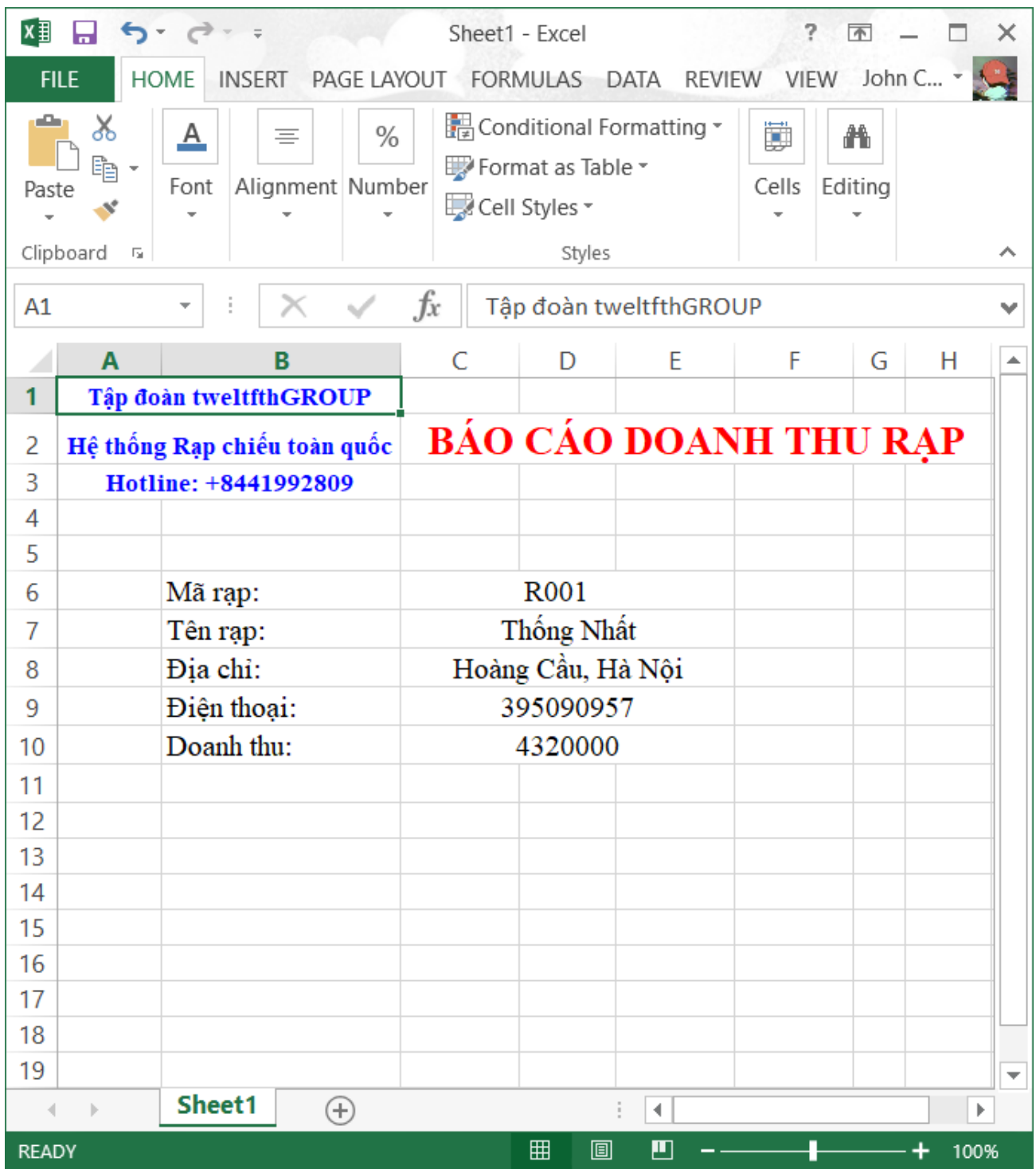
- Yêu cầu liên quan đến form In Doanh Thu Rạp sử dụng chức năng in
 - Ứng dụng: Sử dụng khi xuất dữ liệu ra file Excel để in.
 - Các bước tiến hành :
 - Bước 1: Kết nối project với Excel.
 - Bước 2: Tạo một combobox mã rạp để chọn rạp cần in báo cáo. Lấy dữ liệu về doanh thu của một bộ phim từ bảng tblRap.
 - Bước 3: In ra Excel, chỉnh sửa theo ý.
- Sơ đồ giải quyết



- Form In Doanh Thu Phim:

The screenshot shows a software window titled 'InDoanhThuRap'. The main heading is 'BÁO CÁO DOANH THU RẠP'. Below this, there is a label 'Chọn rạp' followed by a dropdown menu currently showing 'Thống Nhất'. At the bottom center, there is a button with a printer icon and the text 'In báo cáo'.

- Mẫu báo cáo sau khi ấn In:



– Code xử lý:

```
public InDoanhThuRap()
{
    InitializeComponent();
}

private void InDoanhThuRap_Load(object sender, EventArgs e)
{

```

```

        DAO.OpenConnection();
        DAO.FillDataToCombo("SELECT MaRap, TenRap FROM tblRap", cboRap, "MaRap",
"TenRap");
        DAO.CloseConnection();
    }

    private void btnInDTR_Click(object sender, EventArgs e)
    {
        COMExcel.Application exApp = new COMExcel.Application();
        COMExcel.Workbook exBook; //Trong 1 chương trình Excel có nhiều Workbook
        COMExcel.Worksheet exSheet; //Trong 1 Workbook có nhiều Worksheet
        COMExcel.Range exRange;
        string sql;
        DataTable tblRap, sove;
        exBook = exApp.Workbooks.Add(COMExcel.XlWBATemplate.xlWBATWorksheet);
        exSheet = exBook.Worksheets[1];
        // Định dạng chung
        exRange = exSheet.Cells[1, 1];
        exRange.Range["A1:B3"].Font.Size = 10;
        exRange.Range["A1:B3"].Font.Name = "Times new roman";
        exRange.Range["A1:B3"].Font.Bold = true;
        exRange.Range["A1:B3"].Font.ColorIndex = 5; //Màu xanh da trời
        exRange.Range["A1:A1"].ColumnWidth = 7;
        exRange.Range["B1:B1"].ColumnWidth = 15;
        exRange.Range["A1:B1"].MergeCells = true;
        exRange.Range["A1:B1"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
        exRange.Range["A1:B1"].Value = "Tập đoàn twelfthGROUP";
        exRange.Range["A2:B2"].MergeCells = true;
        exRange.Range["A2:B2"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
        exRange.Range["A2:B2"].Value = "Hệ thống Rạp chiếu toàn quốc";
        exRange.Range["A3:B3"].MergeCells = true;
        exRange.Range["A3:B3"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
        exRange.Range["A3:B3"].Value = "Hotline: +8441992809";
        exRange.Range["C2:H2"].Font.Size = 16;
        exRange.Range["C2:H2"].Font.Name = "Times new roman";
        exRange.Range["C2:H2"].Font.Bold = true;
        exRange.Range["C2:H2"].Font.ColorIndex = 3; //Màu đỏ
        exRange.Range["C2:H2"].MergeCells = true;
        exRange.Range["C2:H2"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
        exRange.Range["C2:H2"].Value = "BẢO CÁO DOANH THU RẠP";
        sql = "SELECT MaRap, TenRap, DiaChi, DienThoai FROM tblRap WHERE MaRap = N'" +
cboRap.SelectedValue.ToString() + "'";

        tblRap = DAO.GetDataToTable(sql);
        exRange.Range["B6:C11"].Font.Size = 12;
        exRange.Range["B6:C11"].Font.Name = "Times new roman";
        exRange.Range["B6:B6"].Value = "Mã rạp:";
        exRange.Range["C6:E6"].MergeCells = true;
        exRange.Range["C6:E6"].Value = tblRap.Rows[0][0].ToString();
        exRange.Range["B7:B7"].Value = "Tên rạp:";
        exRange.Range["C7:E7"].MergeCells = true;
        exRange.Range["C7:E7"].Value = tblRap.Rows[0][1].ToString();
        exRange.Range["B8:B8"].Value = "Địa chỉ:";
        exRange.Range["C8:E8"].MergeCells = true;
        exRange.Range["C8:E8"].Value = tblRap.Rows[0][2].ToString();
        exRange.Range["B9:B9"].Value = "Điện thoại:";
        exRange.Range["C9:E9"].MergeCells = true;
        exRange.Range["C9:E9"].Value = tblRap.Rows[0][3].ToString();
    }
}

```

```

        sql = "SELECT SUM(TongTien) from tblLichChieu where MaRap = N'" +
cboRap.SelectedValue.ToString() + "'";
        sove = DAO.GetDataToTable(sql);
        exRange.Range["B10:B10"].Value = "Doanh thu:";
        exRange.Range["C10:E10"].MergeCells = true;
        exRange.Range["C10:E10"].Value = sove.Rows[0][0].ToString();
        exApp.Visible = true;
    }

```

2.10. In ra danh sách phim đang chiếu

- Yêu cầu liên quan đến form In Danh sách phim đang chiếu sử dụng chức năng in
 - Ứng dụng: Sử dụng khi xuất dữ liệu ra file Excel để in.
 - Các bước tiến hành :
 - Bước 1: Kết nối project với Excel.
 - Bước 2: Tạo một DataGridView để có thể xem danh sách ngay các phim đang chiếu. Lấy dữ liệu về danh sách phim đang chiếu từ tblPhim bằng cách đặt điều kiện Ngày khởi chiếu < Ngày hôm nay < Ngày kết thúc.
 - Bước 3: In ra Excel, chỉnh sửa theo ý.
- Sơ đồ giải quyết
- Form In danh sách phim đang chiếu:

The screenshot shows a Windows application window titled "InPhimDangChieu". Inside the window, there is a section titled "DANH SÁCH PHIM ĐANG CHIẾU". Below this title is a table with the following columns: MaPhim, TenPhim, DaoDien, TenTheLoai, NgayKhoiChieu, and NgayKetThu. The first row of data is highlighted in blue and contains the values: MV001, Inception, Christopher Nolan, Hành động, 06/01/20, and 07/23/20. Below the table is a large gray rectangular area, likely a placeholder for a DataGridView. At the bottom center of the form, there is a button with a printer icon and the text "In danh sách".

- Mẫu báo cáo sau khi ấn In:

STT	Mã phim	Tên phim	Đạo diễn	Tên thể loại	Ngày khởi chiếu	Ngày kết thúc
1	MV001	Inception	GBR	LB001	Christopher Nolan	CT001

- Code xử lý:

```
public partial class InPhimDangChieu : Form
{
    public InPhimDangChieu()
    {
        InitializeComponent();
    }

    private void InPhimDangChieu_Load(object sender, EventArgs e)
    {
        DAO.OpenConnection();
        LoadDataToGridView();
        DAO.CloseConnection();
    }

    private void LoadDataToGridView()
    {
        string sql = "select MaPhim, TenPhim, DaoDien, TenTheLoai, NgayKhoiChieu,
        NgayKetThuc from tblPhim inner join tblTheLoai on tblPhim.MaTheLoai = tblTheLoai.MaTheLoai
        where NgayKetThuc > getdate() and NgayKhoiChieu < getdate()";
        SqlDataAdapter adapter = new SqlDataAdapter(sql, DAO.conn);
    }
}
```

```

        DataTable tblPhimDC = new DataTable();
        adapter.Fill(tblPhimDC);
        dataGridView1.DataSource = tblPhimDC;
    }

    private void btnInDTR_Click(object sender, EventArgs e)
    {
        COMExcel.Application exApp = new COMExcel.Application();
        COMExcel.Workbook exBook; //Trong 1 chương trình Excel có nhiều Workbook
        COMExcel.Worksheet exSheet; //Trong 1 Workbook có nhiều Worksheet
        COMExcel.Range exRange;
        int hang = 0, cot = 0;
        string sql = "select * from tblPhim where NgayKetThuc > getdate() and
        NgayKhoiChieu < getdate()";
        SqlDataAdapter adapter = new SqlDataAdapter(sql, DAO.conn);
        DataTable tblPhimDC = new DataTable();
        exBook = exApp.Workbooks.Add(COMExcel.XlWBATemplate.xlWBATWorksheet);
        exSheet = exBook.Worksheets[1];
        // Định dạng chung
        exRange = exSheet.Cells[1, 1];
        exRange.Range["A1:B3"].Font.Size = 10;
        exRange.Range["A1:B3"].Font.Name = "Times new roman";
        exRange.Range["A1:B3"].Font.Bold = true;
        exRange.Range["A1:B3"].Font.ColorIndex = 5; //Màu xanh da trời
        exRange.Range["A1:A1"].ColumnWidth = 7;
        exRange.Range["B1:B1"].ColumnWidth = 15;
        exRange.Range["A1:B1"].MergeCells = true;
        exRange.Range["A1:B1"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
        exRange.Range["A1:B1"].Value = "Tập đoàn twelfthGROUP";
        exRange.Range["A2:B2"].MergeCells = true;
        exRange.Range["A2:B2"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
        exRange.Range["A2:B2"].Value = "Hệ thống Rạp chiếu toàn quốc";
        exRange.Range["A3:B3"].MergeCells = true;
        exRange.Range["A3:B3"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
        exRange.Range["A3:B3"].Value = "Hotline: +8441992809";
        exRange.Range["C2:H2"].Font.Size = 16;
        exRange.Range["C2:H2"].Font.Name = "Times new roman";
        exRange.Range["C2:H2"].Font.Bold = true;
        exRange.Range["C2:H2"].Font.ColorIndex = 3; //Màu đỏ
        exRange.Range["C2:H2"].MergeCells = true;
        exRange.Range["C2:H2"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
        exRange.Range["C2:H2"].Value = "DANH SÁCH PHIM ĐANG CHIẾU";

        tblPhimDC = DAO.GetDataToTable(sql);
        exRange.Range["A6:G6"].Font.Bold = true;
        exRange.Range["A6:G6"].HorizontalAlignment = COMExcel.XlHAlign.xlHAlignCenter;
        exRange.Range["C6:G6"].ColumnWidth = 15;
        exRange.Range["A6:A6"].Value = "STT";
        exRange.Range["B6:B6"].Value = "Mã phim";
        exRange.Range["C6:C6"].Value = "Tên phim";
        exRange.Range["D6:D6"].Value = "Đạo diễn";
        exRange.Range["E6:E6"].Value = "Tên thể loại";
        exRange.Range["F6:F6"].Value = "Ngày khởi chiếu";
        exRange.Range["G6:G6"].Value = "Ngày kết thúc";

        for (hang = 0; hang <= tblPhimDC.Rows.Count - 1; hang++)
        {
            //Điền số thứ tự vào cột 1 từ dòng 7
            exSheet.Cells[1][hang + 7] = hang + 1;
            for (cot = 0; cot <= tblPhimDC.Columns.Count - 1; cot++)

```

```
        //Điền thông tin hàng từ cột thứ 2, dòng 7
        exSheet.Cells[cot + 2][hang + 7] = tblPhimDC.Rows[hang][cot].ToString();
    }
    exApp.Visible = true;
}
```

CHƯƠNG 3: TỔNG KẾT

- Phần mềm đã đáp ứng được cơ bản các yêu cầu quản lý một rạp chiếu phim.
- Các chức năng và yêu cầu đều được cả nhóm chỉnh sửa sao cho khoa học và tốn ít tài nguyên của hệ thống.
- Vận dụng đầy đủ kiến thức của CSDL1 và CSLT2 vào để thiết kế hệ thống một cách tối ưu nhất.
- Còn một số hạn chế về sự hiệu quả, cách trình bày và giao diện. Phần mềm sẽ tiếp tục được hoàn thiện.

Nhóm 12 xin được gửi lời cảm ơn đến cô Trần Thị Huế đã nhiệt tình giảng dạy, hướng dẫn và giải đáp thắc mắc cho chúng em trong suốt thời gian học học phần Cơ sở lập trình II.

Chúc cô có thật nhiều sức khỏe và may mắn!