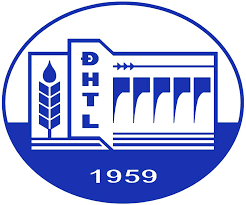
**TRƯỜNG ĐẠI HỌC THỦY LỢI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**----- 🙢🕮🙠 -----**

****

**BÀI TẬP LỚN**

**MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**ĐỀ TÀI:**

**XÂY DỰNG PHẦN MỀM QUẢN LÝ THƯ VIỆN**

***Nhóm sinh viên thực hiện:***

1. **Nguyễn Văn Sang (Lớp 60TH2)**
2. **Tào Phương Quỳnh (Lớp 60TH2)**
3. **Trần Thị Thu Liễu (Lớp 60TH2)**

**Giảng viên hướng dẫn: TS Lương Thị Hồng Lan**

# LỜI NÓI ĐẦU

Duới sự tác động của công cuộc đổi mới, sự phát triển của xã hội và nền kinh tế thị truờng ngày càng được mở rộng. Như vậy trong công tác quản lý, kiểm soát tình hình doanh thu, số luợng,... nếu chỉ làm trên giấy tờ thì rất khó khăn trong việc quản lý, sửa đổi dữ liệu và dễ gây mất mát, bảo mật thông tin. Mô hình chung đòi hỏi cần phải có một hệ thống chặt chẽ, hợp lý mới đảm bảo được việc quản lý dễ dàng đáp ứng được nhu cầu cần thiết hiện nay.

Ngày nay, với sự bùng nổ mạnh mẽ của ngành khoa học kỹ thuật nói chung và ngành CNTT nói riêng. Việc ứng dụng tin học trong công tác quản lý phát triển rất mạnh, ứng dụng tin học giúp cho công tác quản lý ngày càng hiệu quả hơn, góp phần nâng cao hiệu quả trong công việc, đưa ra các báo cáo, số liệu thống kê,... Đồng thời, trong công việc của các nhà quản lý tiết kiệm được nhiều thời gian, công sức, việc quản lý dễ dàng và bảo mật hơn.

Nhờ có sự phát triển của CNTT mà hiện nay đã có nhiều phần mềm ứng dụng quản lý được ra đời ví dụ như: Quản lý nhân sự, quản lý khách sạn, quản lý ngân hàng,... Vậy tại sao ta không thể kết hợp tin học trong việc quản lý thư viện để khiến cho việc quản lý trở nên dễ dàng hơn.

Với những lý do trên nhóm đã quyết định chọn đề tài “Quản lý thư viện” với mong muốn tìm hiểu về lĩnh vực quản lý nói chung mà mục đích chính là quản lý sách của thư viện Đại học Thủy Lợi nói riêng để đáp ứng được nhu cầu thực tế của con người. Mục tiêu của bài báo cáo là nêu rõ các đặc tả yêu cầu của hệ thống, phân tích, thiết kế để xây dựng được phần mềm quản lý thư viện trên phục vụ cho việc quản lý hoàn toàn tự động trên máy tính.

**MỤC LỤC**

[**LỜI NÓI ĐẦU 2**](#_Toc55517953)

[**CHƯƠNG 1: BÀI TOÁN ĐẶT RA VÀ YÊU CẦU CỦA HỆ THỐNG 4**](#_Toc55517954)

[**1.1 Giới thiệu bài toán: 4**](#_Toc55517955)

[**1.2 Yêu cầu chung của bài toán: 4**](#_Toc55517956)

[**1.3 Cách thức giải thích bài toán: 4**](#_Toc55517957)

[**1.4 Phát biểu nghiệp vụ của bài toán: 4**](#_Toc55517958)

[**CHƯƠNG 2:PHÂN TÍCH HỆ THỐNG 5**](#_Toc55517959)

[**2.1 Phân tích chức năng của hệ thống: 5**](#_Toc55517960)

[**2.2 Người sử dụng hệ thống: 5**](#_Toc55517961)

[**CHƯƠNG 3: XÂY DỰNG MÔ HÌNH LIÊN KẾT THỰC THỂ 6**](#_Toc55517962)

[**3.1 Xác định các thực thể toàn vẹn: 6**](#_Toc55517963)

[**3.2 Xây dựng mô hình ER 6**](#_Toc55517964)

[**3.3 Xây dựng mô hình quan hệ 7**](#_Toc55517965)

[**3.4 Mô hình hệ thống 8**](#_Toc55517966)

[**CHƯƠNG 4 : MỘT SỐ CÂU LỆNH SQL 9**](#_Toc55517967)

[**4.1 PROCEDURE (Thủ tục) 9**](#_Toc55517968)

[**4.2 FUNCTION (Hàm) 12**](#_Toc55517969)

[**4.3 VIEW 14**](#_Toc55517970)

[**4.4 TRIGGER 17**](#_Toc55517971)

[**4.5 PHÂN QUYỀN 20**](#_Toc55517972)

[**CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 21**](#_Toc55517973)

[**5.1 Kết luận: 21**](#_Toc55517974)

[**5.2 Demo giao diện 21**](#_Toc55517975)

[**5.3 Hướng phát triển 21**](#_Toc55517976)

[**CHƯƠNG 6: TÀI LIỆU THAM KHẢO 21**](#_Toc55517977)

[**6.1 Tài liệu tham khảo online: 21**](#_Toc55517978)

[**6.2 Giáo trình: 21**](#_Toc55517979)

# CHƯƠNG 1: BÀI TOÁN ĐẶT RA VÀ YÊU CẦU CỦA HỆ THỐNG

## Giới thiệu bài toán:

Như chúng ta đã biết, trong một hệ thống thư viện lớn như thư viện của trường Đại Học Thủy Lợi thì số lượng sách là rất nhiều đồng thời để quản lý được những số sách đó thì cũng cần số lượng nhân viên nhiều. Đặc biệt những ngày đầu kì học thì số lượng nhân viên đến mượn giáo trình rất đông.Vì vậy yêu cầu đặt ra ở đây là**:”Phải làm sao để tiết kiệm được thời gian, nguồn nhân lực, thông tin tin cậy, bảo mật được hệ thống quản lý một cách hợp lí nhất ?”**

## Yêu cầu chung của bài toán:

Đặc tả các yêu cầu,chức năng: Quản lý độc giả, mượn - trả sách, quản lý các đầu sách, thống kê độc giả mượn sách quá hạn trả sách, sắp đến hạn trả và tìm kiếm sách.

## Cách thức giải thích bài toán:

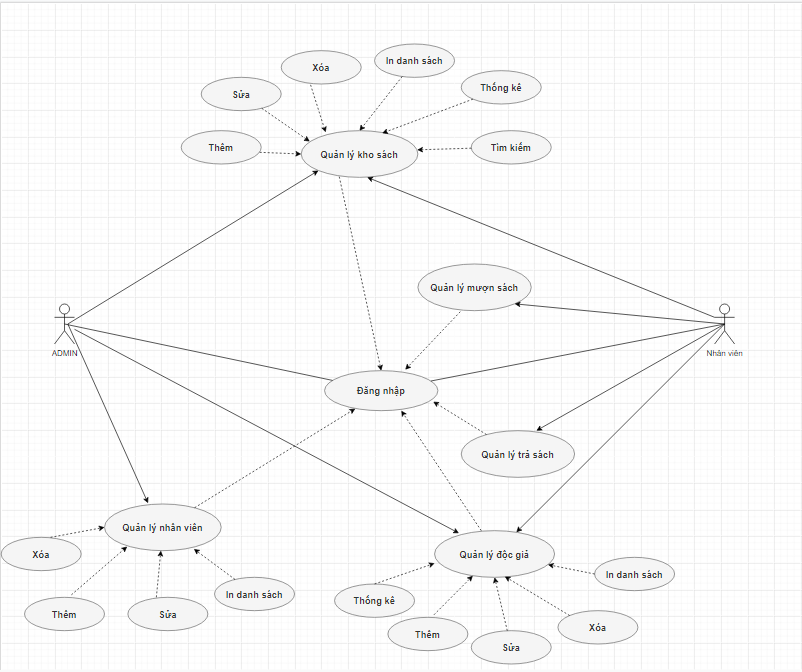
* Tìm hiểu quy tắc nghiệp vụ, các yêu cầu đặt ra của đề tài
* Các ngôn ngữ lập trình bậc cao và cơ sở dữ liệu
* Bắt tay vào xây dựng ứng dụng
* Tiến hành kiểm tra và chạy thử
* Thay đổi và hoàn thiện

## Phát biểu nghiệp vụ của bài toán:

* **Quản lý mượn trả sách**: Khi độc giả (sinh viên) đến mượn sách. Nếu thông tin không đáp ứng yêu cầu (chưa đóng phí) thì đưa ra thông báo từ chối, nếu đáp ứng được thì lập phiếu mượn cho độc giả (sinh viên). Khi độc giả trả sách, phiếu trả sẽ được lập cho người mượn dựa trên sự so khớp và tự động cập nhật số lượng khi phiếu mượn – trả được lập.
* **Quản lý độc giả:** Độc giả (sinh viên) đang học tại trường sẽ được lưu thông tin ở trên hệ thống. Đồng thời hệ thống sẽ quản lý các giao dịch thông tin liên quan tới độc giả.
* **Quản lý sách**: Nhập mới thông tin sách ,sửa chữa thông tin, xóa bỏ những sách không còn được sử dụng.
* **Quản lý nhân viên**: Admin có quyền thao tác quản lý thêm, sửa, xóa thông tin nhân viên.
* **Quản lý thống kê**: Lấy thông tin từ các bảng cần thiết để đưa ra số liệu: Tổng độc giả, số độc giả đang mượn, số độc giả quá hạn trả, số độc giả chưa đóng phí, sách đang mượn, sách tồn, sách quá hạn chưa trả, sách sắp đến hạn trả…

# CHƯƠNG 2:PHÂN TÍCH HỆ THỐNG

## Phân tích chức năng của hệ thống:



Hình 1:Mô Hình chức năng hệ thống

## Người sử dụng hệ thống:

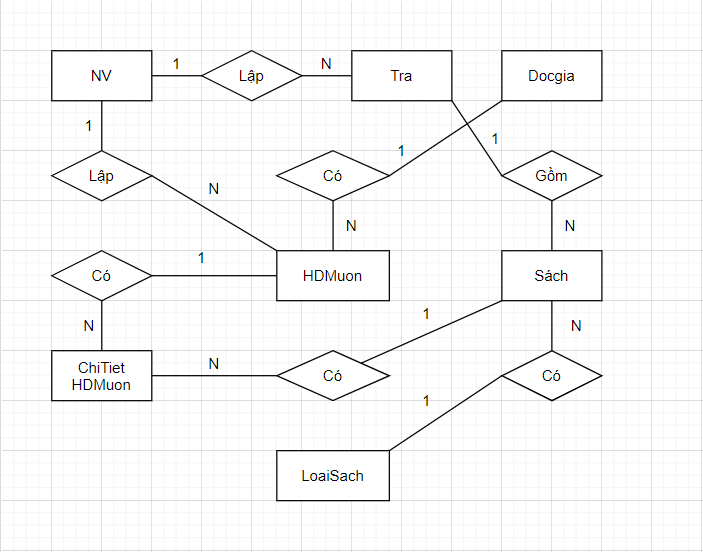
* **Quản lý (ADMIN):** Quản lý chịu trách nhiệm chính trong việc quản lý ,bao gồm quản lý nhân viên, quản lý kho sách, độc giả (sinh viên), thống kê, cập nhật bổ sung, sửa đổi hoặc xóa khi cần thiết.
* **Nhân viên:** Nhân viên chịu trách nhiệm chính trong việc quản lý Mượn – Trả sách. Đồng thời quản lý thêm, sửa, xóa các đầu sách và thêm độc giả mới, sửa, xóa thông tin các độc giả. Nhân viên sẽ tiếp nhận thông tin yêu cầu Mượn – Trả sách của độc giả (sinh viên). Hướng dẫn cho sinh viên sử dụng phần mềm của hệ thống để tìm kiếm sách khi có yêu cầu.
* **Độc giả(sinh viên):** Sinh viên tương tác với hệ thống qua ID đôc giả(mã sinh viên) của mình đã được cấp sẵn, xem trực tiếp trạng thái của mình đã đóng phí chưa, đang mượn bao nhiêu cuốn sách và sách sắp đến hạn trả.

# CHƯƠNG 3: XÂY DỰNG MÔ HÌNH LIÊN KẾT THỰC THỂ

## Xác định các thực thể toàn vẹn:

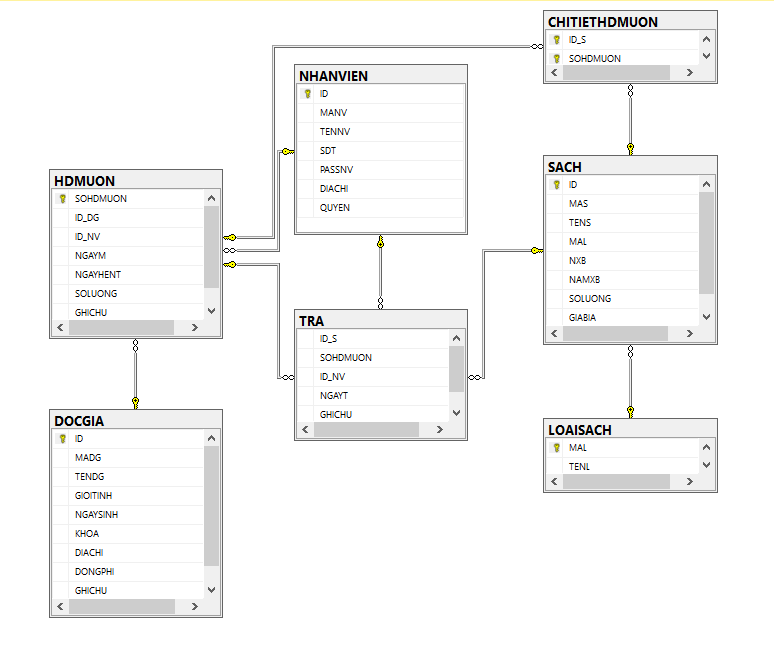
* **LoaiSach** (**MaL**, TenS)
* **Sach** (**ID, MaL**, MaS, TenS, NXB, NamXB, SoLuong, Giabia)
* **NV** (**ID**, MaNV, TenNV, SDT, PassNV, DiaChi, Quyen)
* **DocGia** (**ID**, MaDG, TenDG, GioiTinh, NgaySinh, Khoa, DiaChi, DongPhi, GhiChu)
* **Tra** (**ID\_NV, ID\_S**, **SoHDMuon**, NgayT, GhiChu)
* **HDMuon** **(SoHDMuon**, ID\_DG, ID\_NV,NgayM, NgayHenT, SoLuong, GhiChu)
* **ChiTietHDMuon** (**ID\_S, SoHDMuon**)

## Xây dựng mô hình ER:



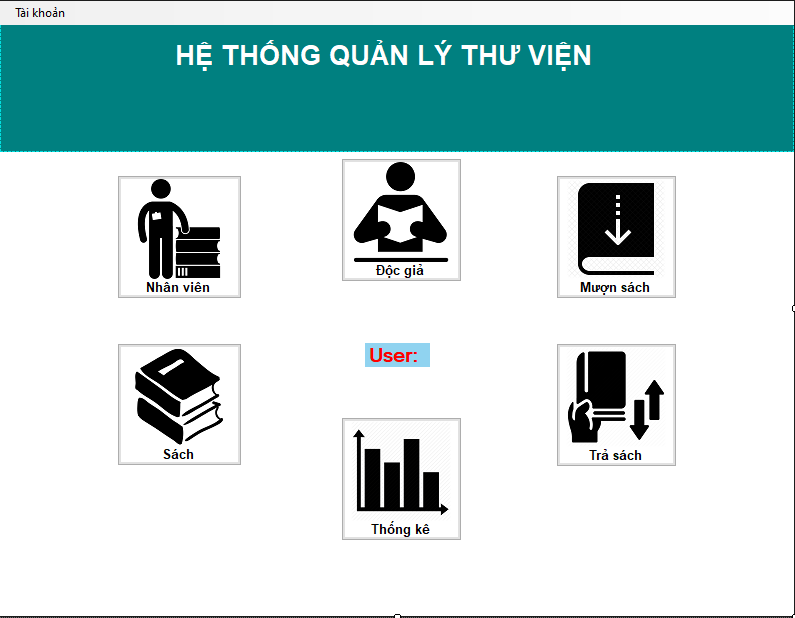
Hình 2: Mô hình liên kết thực thể ER

## Xây dựng mô hình quan hệ



Hình 3 : Mô hình quan hệ

## Mô hình hệ thống



# CHƯƠNG 4 : MỘT SỐ CÂU LỆNH SQL

## PROCEDURE (Thủ tục)

* **Thủ tục 1: Tạo thủ tục thống kê các độc giả chưa đóng phí mượn sách**

GO

CREATE PROCEDURE TT\_CHUADP

AS

BEGIN

SELECT ID,MADG,TENDG,NGAYSINH,KHOA,CASE WHEN GIOITINH=1 THEN N'NAM' WHEN GIOITINH=0 THEN N'NỮ' END AS GIOITINH,

CASE WHEN DONGPHI=1 THEN N'ĐÃ ĐÓNG PHÍ' WHEN DONGPHI=0 THEN N'CHƯA ĐÓNG PHÍ' END AS DONGPHI,DIACHI,GHICHU

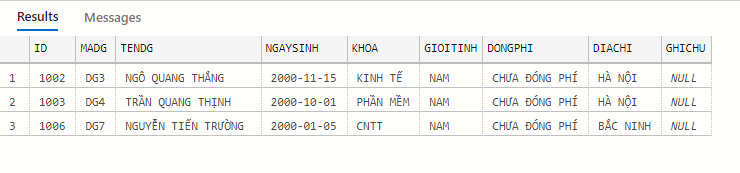
FROM DOCGIA

WHERE DONGPHI=0

END

--CHẠY THỬ:

EXECUTE TT\_CHUADP



* **Thủ tục 2: Tạo thủ tục thống kê các sách được mượn bởi 1 SV nào đó**

GO

CREATE PROCEDURE TK\_S @IDDG INT

AS

BEGIN

SELECT DOCGIA.ID,DOCGIA.MADG,DOCGIA.TENDG,SACH.MAS,SACH.MAL,SACH.TENS,SACH.NXB,SACH.NAMXB,SACH.SOLUONG,SACH.GIABIA FROM HDMUON

INNER JOIN DOCGIA ON HDMUON.ID\_DG=DOCGIA.ID

INNER JOIN CHITIETHDMUON ON CHITIETHDMUON.SOHDMUON=HDMUON.SOHDMUON

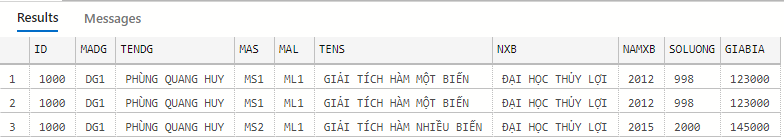
INNER JOIN SACH ON CHITIETHDMUON.ID\_S=SACH.ID

WHERE DOCGIA.ID=@IDDG

END

--CHẠY THỬ:

EXECUTE TK\_S 1000



* **Thủ tục 3: Tạo thủ tục xóa 1 nhân viên theo ID**

GO

CREATE PROCEDURE PROC\_DELETE\_NV

@ID INT

AS BEGIN

DELETE FROM NHANVIEN

WHERE ID=@ID

END

--CHẠY THỬ:

GO

EXECUTE PROC\_DELETE\_NV 5

* **Thủ tục 4: Tạo thủ tục tìm kiếm 1 quyển sách nào đó theo mã sách**

GO

CREATE PROCEDURE PROC\_TK\_SACH @IDS INT

AS

BEGIN

IF(@IDS NOT IN (SELECT ID FROM SACH))

SELECT ID AS N'NO DATA',MAS AS N'NO DATA',MAL AS N'NO DATA',TENS AS N'NO DATA',NXB AS N'NO DATA',NAMXB AS N'NO DATA',SOLUONG AS N'NO DATA',GIABIA AS N'NO DATA' FROM SACH

WHERE @IDS=ID

ELSE

SELECT ID,MAS,MAL,TENS,NXB,NAMXB,SOLUONG,GIABIA FROM SACH

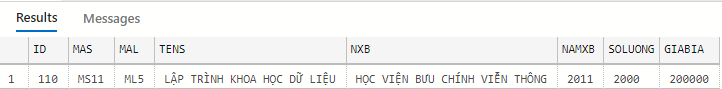
WHERE @IDS =ID

END

--CHẠY THỬ:

GO

EXECUTE PROC\_TK\_SACH 110



* **Thủ tục 5: Tạo thủ tục thống kê danh sách sinh viên mượn sách trong một ngày bất kì**

GO

CREATE PROCEDURE PROC\_THONGKE\_NGAY(@DATE DATE)

AS

BEGIN

SELECT HDMUON.SOHDMUON,DOCGIA.MADG,DOCGIA.TENDG,HDMUON.SOLUONG FROM HDMUON,DOCGIA

WHERE HDMUON.ID\_DG=DOCGIA.ID AND HDMUON.NGAYM =@DATE

END

--CHẠY THỬ:

GO

EXECUTE PROC\_THONGKE\_NGAY '9/26/2020'



* **Thủ tục 6: Tạo thủ tục đưa ra danh sách các độc giả(sinh viên) khoa bất kì mượn một cuốn sách nào đó(theo tên sách). Mặc định là khoa CNTT sách mạng máy tính**

GO

CREATE PROCEDURE PROC\_DSDG(@TENK NVARCHAR(50)='CNTT',@TENS NVARCHAR(100)=N'MẠNG MÁY TÍNH')

AS

BEGIN

SELECT DOCGIA.MADG,DOCGIA.TENDG FROM DOCGIA

INNER JOIN HDMUON ON DOCGIA.ID=HDMUON.ID\_DG

INNER JOIN CHITIETHDMUON ON HDMUON.SOHDMUON=CHITIETHDMUON.SOHDMUON

INNER JOIN SACH ON CHITIETHDMUON.ID\_S=SACH.ID

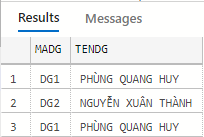
WHERE SACH.TENS=@TENS AND DOCGIA.KHOA=@TENK

END

--CHẠY THỬ:

GO

EXECUTE PROC\_DSDG N'CNTT',N'GIẢI TÍCH HÀM MỘT BIẾN'



## FUNCTION (Hàm)

* **Hàm 1: Hàm trả về giá trị tính số lần mượn một quyển sách nào đó của một độc giả (sinh viên) bất kì**

GO

CREATE FUNCTION FUN\_SOLUONGSACH(@IDSACH INT,@IDDG INT )

RETURNS INT

AS

BEGIN

DECLARE @SL INT

SELECT @SL=(SELECT COUNT(ID\_S) FROM CHITIETHDMUON INNER JOIN HDMUON ON CHITIETHDMUON.SOHDMUON=HDMUON.SOHDMUON

INNER JOIN DOCGIA ON HDMUON.ID\_DG=DOCGIA.ID WHERE HDMUON.ID\_DG=@IDDG AND CHITIETHDMUON.ID\_S=@IDSACH)

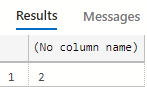
RETURN @SL

END

--CHẠY THỬ:

GO

SELECT DBO.FUN\_SOLUONGSACH(100,1000)



* **Hàm 2: Hàm trả về giá trị tính tổng số sách đang được mượn**

GO

CREATE FUNCTION FUN\_TONG\_MUON()

RETURNS INT

AS BEGIN

DECLARE @TONGMUON INT

SELECT @TONGMUON=(SELECT SUM (SOLUONG) FROM HDMUON)

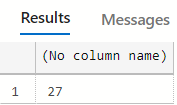
RETURN @TONGMUON

END

--CHẠY THỬ:

GO

SELECT DBO.FUN\_TONG\_MUON()



* **Hàm 3: Hàm trả về giá trị tính tổng số sách còn lại trong kho**

GO

CREATE FUNCTION FUN\_TONGKHO()

RETURNS INT

AS BEGIN

DECLARE @TONGKHO INT

SELECT @TONGKHO=(SELECT SUM(SACH.SOLUONG) FROM SACH)-(SELECT DBO.FUN\_TONG\_MUON())

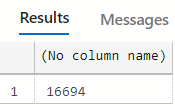
RETURN @TONGKHO

END

--CHẠY THỬ:

GO

SELECT DBO.FUN\_TONGKHO()



* **Hàm 4: Hàm trả về bảng tính số lượng sách theo từng loại sách**

GO

CREATE FUNCTION FUN\_TONG\_LOAISACH()

RETURNS @KQ TABLE(TENL NVARCHAR(50),SOLUONG INT)

AS BEGIN

INSERT INTO @KQ

SELECT MAL,SUM(SACH.SOLUONG) AS SOLUONG FROM SACH

GROUP BY MAL

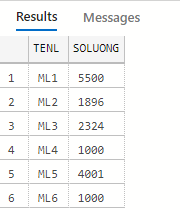
RETURN

END

--CHẠY THỬ:

GO

SELECT \* FROM FUN\_TONG\_LOAISACH()



* **Hàm 5: Hàm trả về danh sách của một loại sách bất kỳ. Nếu mã loại không có trong CSDL thì hiển thị tất cả sách theo từng loại**

GO

CREATE FUNCTION FUN\_DS\_SACH(@MAL NVARCHAR(10))

RETURNS @KQ TABLE(MAS NVARCHAR(10),TENS NVARCHAR(100),MAL NVARCHAR(10))

AS BEGIN

IF(@MAL NOT IN (SELECT DISTINCT MAL FROM SACH))

INSERT INTO @KQ

SELECT MAS,TENS,MAL FROM SACH

ELSE

INSERT INTO @KQ

SELECT MAS,TENS,MAL FROM SACH

WHERE @MAL=MAL

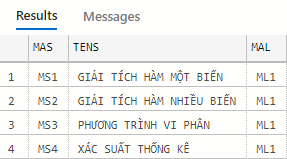
RETURN

END

--CHẠY THỬ:

GO

SELECT \* FROM DBO.FUN\_DS\_SACH('ML1')



## VIEW

* **Khung nhìn 1: Tạo view liệt kê thông tin các độc giả mượn sách quá hạn**

GO

CREATE VIEW TT\_DG\_QUAHAN1

AS

SELECT DOCGIA.ID,DOCGIA.MADG,DOCGIA.TENDG FROM DOCGIA

INNER JOIN HDMUON ON DOCGIA.ID=HDMUON.ID\_DG

INNER JOIN TRA ON HDMUON.SOHDMUON=TRA.SOHDMUON

WHERE NGAYHENT<NGAYT

--CHẠY THỬ:

GO

SELECT \* FROM TT\_DG\_QUAHAN1



* **Khung nhìn 2: Tạo view chứa thông tin sách đang được mượn bởi độc giả**

GO

CREATE VIEW TT\_SACH\_MUON\_SV

AS

SELECT DG.ID,DG.MADG,DG.TENDG,SACH.TENS,SACH.MAL,SACH.NXB,SACH.SOLUONG,SACH.GIABIA FROM DOCGIA AS DG

INNER JOIN HDMUON ON DG.ID=HDMUON.ID\_DG

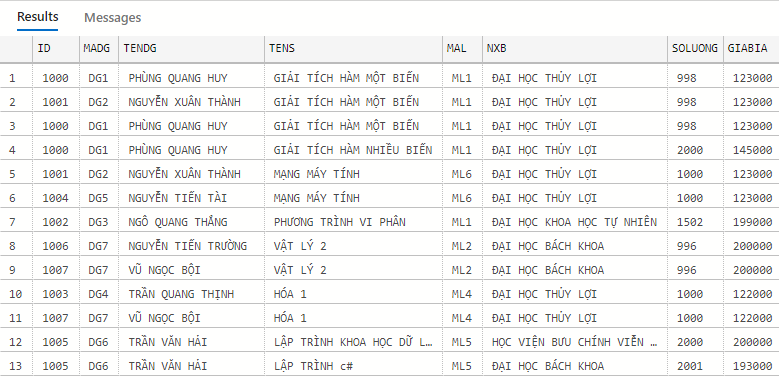
INNER JOIN CHITIETHDMUON ON HDMUON.SOHDMUON=CHITIETHDMUON.SOHDMUON

INNER JOIN SACH ON CHITIETHDMUON.ID\_S=SACH.ID

--CHẠY THỬ:

GO

SELECT \* FROM TT\_SACH\_MUON\_SV



* **Khung nhìn 3: Tạo view chứa thông tin về các sách sắp đến hạn trả**

GO

CREATE VIEW TT\_SACH\_SAPDENHANTRA

AS

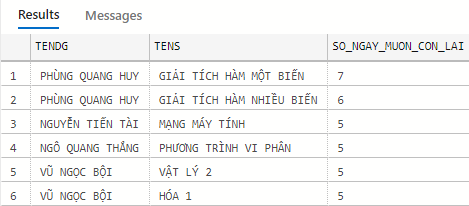
SELECT DOCGIA.TENDG,SACH.TENS,DATEDIFF(DD,NGAYHENT,GETDATE()) AS SO\_NGAY\_MUON\_CON\_LAI FROM HDMUON

INNER JOIN DOCGIA ON HDMUON.ID\_DG=DOCGIA.ID

INNER JOIN CHITIETHDMUON ON HDMUON.SOHDMUON=CHITIETHDMUON.SOHDMUON

INNER JOIN SACH ON CHITIETHDMUON.ID\_S=SACH.ID

WHERE DATEDIFF(DD,NGAYHENT,GETDATE())<=7 AND DATEDIFF(DD,NGAYHENT,GETDATE())>=0



* **Khung nhìn 4: Tạo view chứa thông tin sách chưa từng được mượn**

GO

CREATE VIEW TT\_SACH\_CHUAMUON

AS

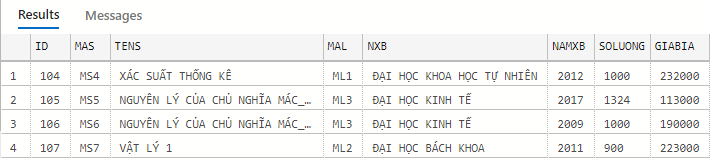
SELECT \* FROM SACH

WHERE ID NOT IN (SELECT DISTINCT ID\_S FROM CHITIETHDMUON)

--CHẠY THỬ:

GO

SELECT \* FROM TT\_SACH\_CHUAMUON



* **Khung nhìn 5: Tạo view chứa thông tin sách mượn quá hạn chưa trả**

GO

CREATE VIEW TT\_SACH\_QH

AS

SELECT SACH.ID,SACH.MAS,SACH.TENS,SACH.MAL FROM SACH INNER JOIN TRA ON SACH.ID=TRA.ID\_S

INNER JOIN HDMUON ON TRA.SOHDMUON=HDMUON.SOHDMUON

WHERE NGAYHENT<NGAYT

--CHẠY THỬ:

GO

SELECT \* FROM TT\_SACH\_QH



## TRIGGER

* **Trigger 1: Trigger không cho phép thêm phiếu mượn nếu ngày mượn lớn hơn ngày hiện tại ngược lại chèn 1 bản ghi mới của bảng HDMuon**

GO

CREATE TRIGGER CHECK\_DATE

ON HDMUON

FOR INSERT

AS

IF((SELECT NGAYM FROM inserted)>GETDATE())

BEGIN

PRINT(N'VUI LÒNG KIỂN TRA CỘT NGÀY MƯỢN CỦA PHIẾU MƯỢN')

ROLLBACK TRANSACTION

END

ELSE

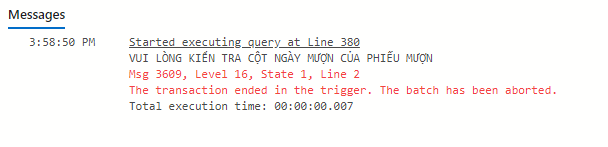
PRINT(N'THÊM PHIẾU MƯỢN THÀNH CÔNG')

--CHẠY THỬ:

GO

INSERT HDMUON

VALUES(23,1000,1,'11/26/2020','12/26/2020',1,'GHICHU100')



* **Trigger 2: Viết một trigger để đảm bảo rằng khi nhập số lượng sách vào kho phải lớn hơn 0 và giá bìa phải lớn hơn 50.000**

GO

CREATE TRIGGER CHECK\_SL\_SACH

ON SACH

FOR INSERT

AS

IF((SELECT SOLUONG FROM inserted)<=0 OR (SELECT GIABIA FROM inserted)<50000)

BEGIN

PRINT(N'VUI LÒNG KIỂM TRA LẠI SỐ LƯỢNG SÁCH HOẶC GIÁ BÌA SÁCH!!!')

ROLLBACK TRANSACTION

END

ELSE

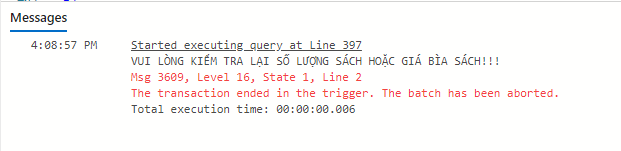
PRINT(N'THÊM SÁCH THÀNH CÔNG !!!')

--CHẠY THỬ:

GO

INSERT SACH

VALUES(131,'MS31',N'SỨC BỀN VẬT LIỆU 2','ML4',N'ĐẠI HỌC THỦY LỢI','2011',0,3)



* **Trigger 3: Viết trigger cập nhật số lượng sách trong kho khi có giao dịch mượn sách**

GO

CREATE TRIGGER UPDATE\_GD\_MUON

ON CHITIETHDMUON

FOR INSERT

AS

BEGIN

DECLARE @SL\_MUON INT

SET @SL\_MUON=(SELECT SOLUONG FROM HDMUON WHERE SOHDMUON=(SELECT SOHDMUON FROM inserted))

DECLARE @ID\_S INT

SET @ID\_S=(SELECT ID\_S FROM inserted)

UPDATE SACH

SET SOLUONG=SOLUONG-@SL\_MUON

WHERE ID=@ID\_S

END

--CHẠY THỬ (Chạy trên winform C#)

* **Trigger 4: Tự động cập nhật số lượng sách trong kho khi có giao dịch trả sách**

GO

CREATE TRIGGER UPDATE\_GD\_TRA

ON TRA

FOR INSERT

AS

BEGIN

DECLARE @SL\_TRA INT

SET @SL\_TRA=(SELECT SOLUONG FROM HDMUON WHERE SOHDMUON=(SELECT SOHDMUON FROM inserted))

DECLARE @ID\_S INT

SET @ID\_S=(SELECT ID\_S FROM inserted)

UPDATE SACH

SET SOLUONG=SOLUONG+@SL\_TRA

WHERE ID=@ID\_S

END

--CHẠY THỬ(Chạy trên winform C#)

* **Trigger 5: Tạo trigger không cho thêm sách nếu Mã Loại(MAL) trong bảng SACH không tồn tại trong bảng LOAISACH(MAL)**

GO

CREATE TRIGGER CHECK\_MAL1

ON SACH

FOR INSERT

AS

IF((SELECT MAL FROM inserted) NOT IN (SELECT MAL FROM LOAISACH))

BEGIN

PRINT(N'MÃ LOẠI KHÔNG TỒN TẠI. VUI LÒNG KIỂM TRA LẠI !!!')

ROLLBACK TRANSACTION

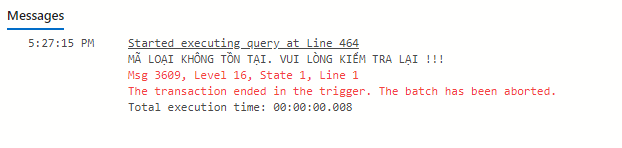
END

--CHẠY THỬ:

GO

INSERT SACH

VALUES(161,'MS60',N'TRÍ TUỆ NHÂN TẠO','ML19',N'ĐẠI HỌC BÁCH KHOA','2012',1000,123000)



## PHÂN QUYỀN

Thực hiện phân quyền trên Winform Application C#:

* **Bước 1: Tạo hàm thực hiện truy vấn lấy loại tài khoản**

public string getQT(string user)

{

return cn.GetValue(@"Select QUYEN from NHANVIEN WHERE ID='" + user + "'");

}

* **Bước 2: Gọi hàm vừa tạo ở bước 1 và lưu vào biến**

//Lấy ra Quyền của người đăng nhập theo tên đăng nhập(ID)

this.ParentForm.Name = tk.getUser(textBox1.Text)+"\_"+tk.getQT(int.Parse(textBox1.Text.Trim()));

* **Bước 3:Thực hiện kiểm tra loại tài khoản và phân quyền truy cập**

//Nếu là nhân viên thì KHÓA chức năng quản lý nhân viên trong FormMain

if(this.ParentForm.Name.Split('\_')[1] =="NV")

{

button1.Enabled = false;

}

//Nếu là ADMIN(Quản trị) thì KHÓA chức năng mượn trả sách trong FormMain

if(this.ParentForm.Name.Split('\_')[1]=="QT")

{

button4.Enabled = false;

button5.Enabled = false;

}

# CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận:

* Sau thời gian phân tích, thiết kế, lập trình nhóm đã hoàn thành cơ bản hệ thống”Quản lý thư viện”, đáp ứng các chức năng quản lý của người quản lý và nhân viên quản lý.
* Mặc dù đã có nhiều cố gắng trong suốt quá trình làm bài tập lớn, nhưng do thời gian có hạn và thiếu kinh nghiệm trong thực tế cũng không thể tránh khỏi những nhược điểm và thiếu sót. Vì vậy, nhóm em mong nhận được sự góp ý của thầy(cô) để nhóm chúng em có thể phát triển hệ thống này hoàn thiện hơn.

## Demo giao diện

## Hướng phát triển

Trong tương lai, nhóm muốn phát triển hệ thống này thêm một số chức năng để đáp ứng yêu cầu thực tế như: quản lý việc nhập xuất, cho sinh viên tự đăng nhập vào hệ thống để xem thông tin, xây dựng website cho phép sinh viên có thể xem và tìm hiểu trước khi đến mượn sách,…đưa hệ thống vào triển khai thực tế tại thư viện.

# CHƯƠNG 6: TÀI LIỆU THAM KHẢO

## Tài liệu tham khảo online:

1. C# Hướng dẫn tạo form thêm sửa xóa bằng kỹ thuật 3 lớp hoàn chỉnh: <https://www.youtube.com/watch?v=860BK3mcp48&ab_channel=69C%C3%A0Mau>
2. Lập trình 3 lớp của thầy Nguyễn Xuân Hùng:

<https://www.youtube.com/watch?v=haYvBlvID4A&list=PLIbobkdH1ftS5_L858q3BXfsexe-HWprG&ab_channel=NguyenXuanHung>

## Giáo trình:

1. TS.Lại Hiền Phương, Bài giảng môn Hệ quản trị cơ sở dữ liệu
2. Giáo trình Hệ quản trị cơ sở dữ liệu Đại học Thủy Lợi