

TRƯỜNG ĐẠI HỌC THỦY LỢI
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO BÀI TẬP LỚN MÔN HỌC
CÔNG NGHỆ PHẦN MỀM

Đề tài: ***Nostimo – Chia sẻ Nấu Ăn***

Nhóm sinh viên thực hiện: Nhóm 2

Thành viên nhóm:

1. Nguyễn Tấn Tài - 2451261056
2. Đỗ Yến Nhi - 2451261042
3. Vi Thị Phương Nhung - 2451261044
4. Cao Khánh Toàn - 2451261065
5. Vũ Thị Bảo Nhi - 2451261043
6. Nguyễn Duy Mạnh - 2451261027
7. Nguyễn Hoàng Anh - 2451260973
8. Đặng Phương Nam - 2451261033
9. Đặng Đình Phước - 2451261048

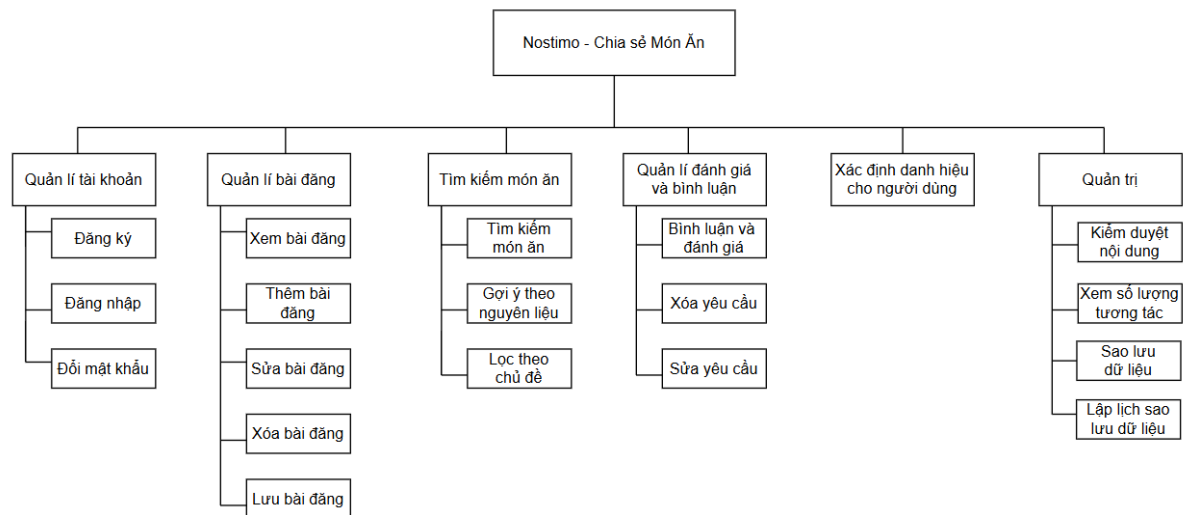
Giảng viên hướng dẫn: Nguyễn Thị Thu Hương

Hà Nội, tháng 12 năm 2025

NHIỆM VỤ 4/4: SPRINT 2/2

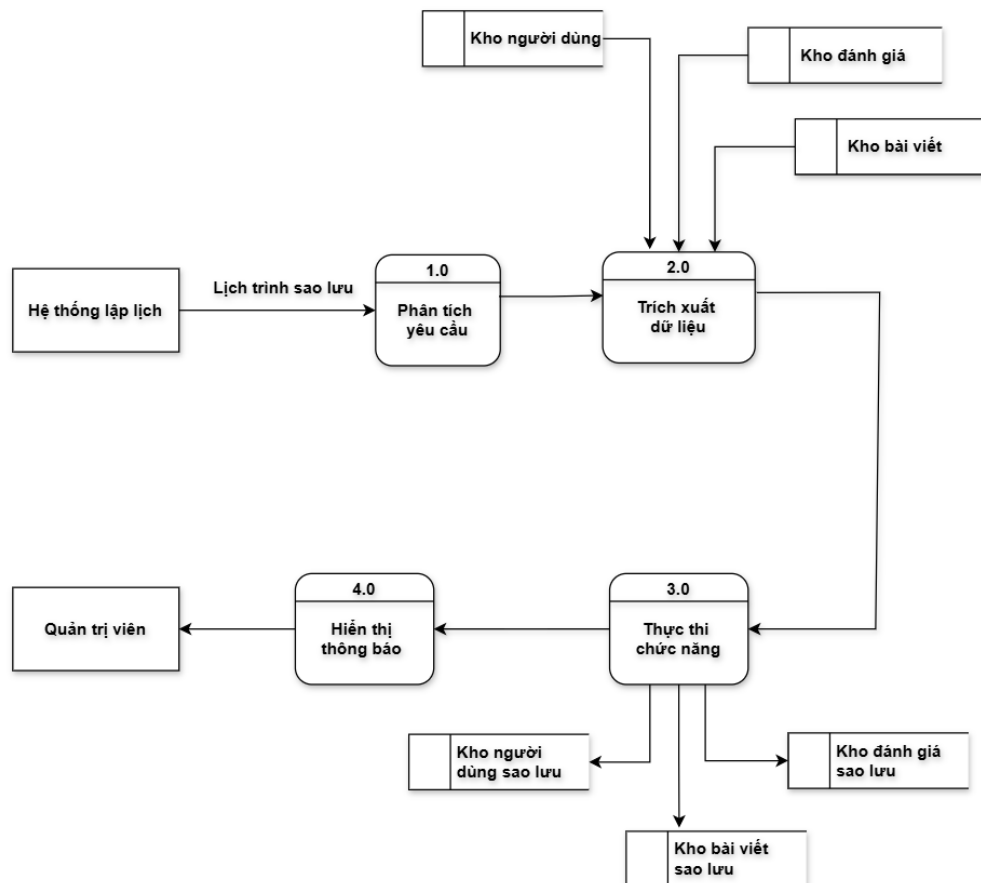
I. Tài liệu đặc tả yêu cầu phần mềm

1.1 Biểu đồ Phân rã chức năng (FDD)

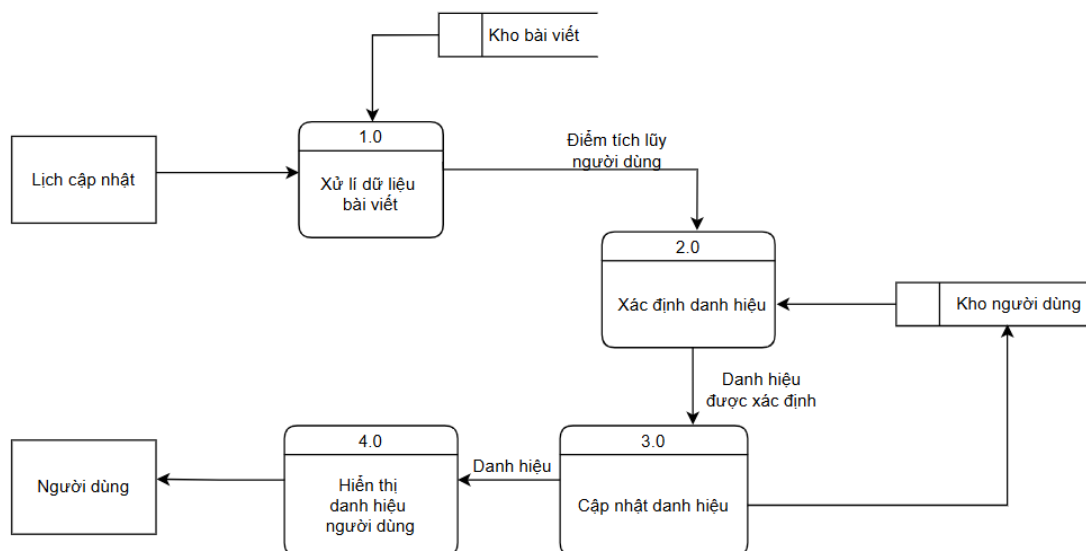


1.2 Biểu đồ Luồng dữ liệu (DFD)

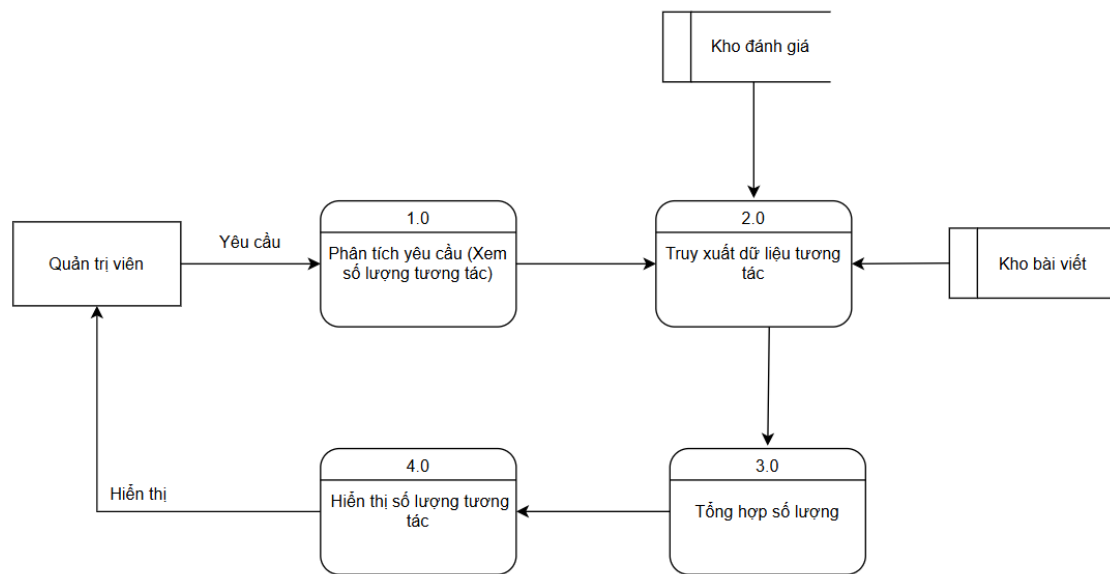
1.2.1 DFD Sao lưu dữ liệu



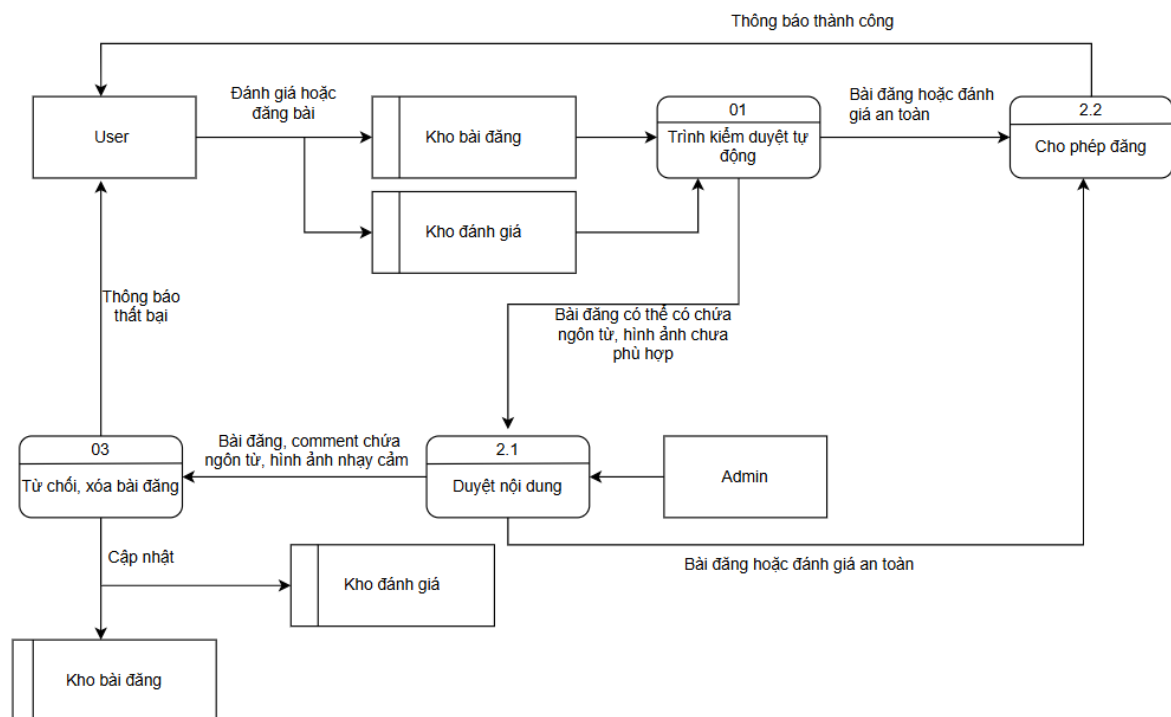
1.2.2 DFD Xác định danh hiệu người dùng



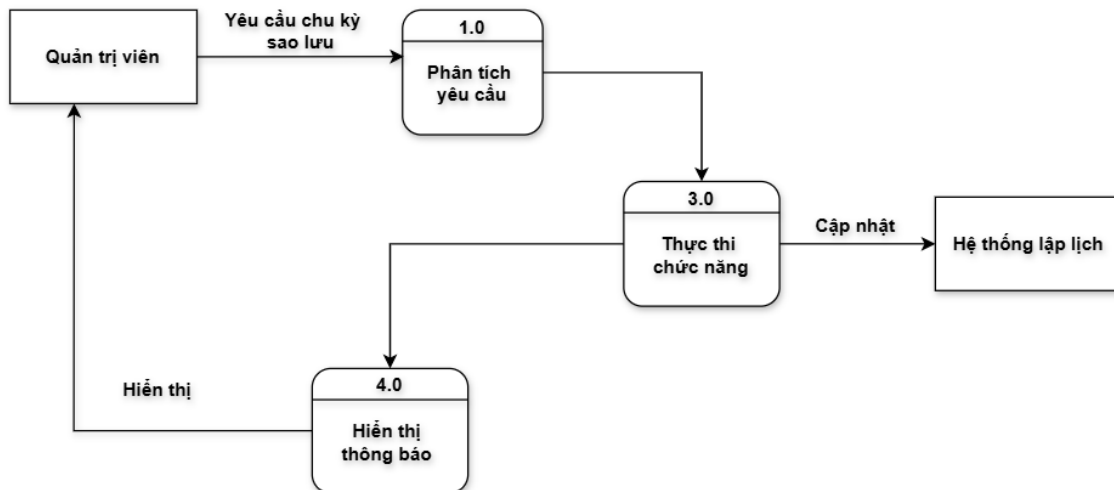
1.2.3 DFD Xem thống kê lượng tương tác



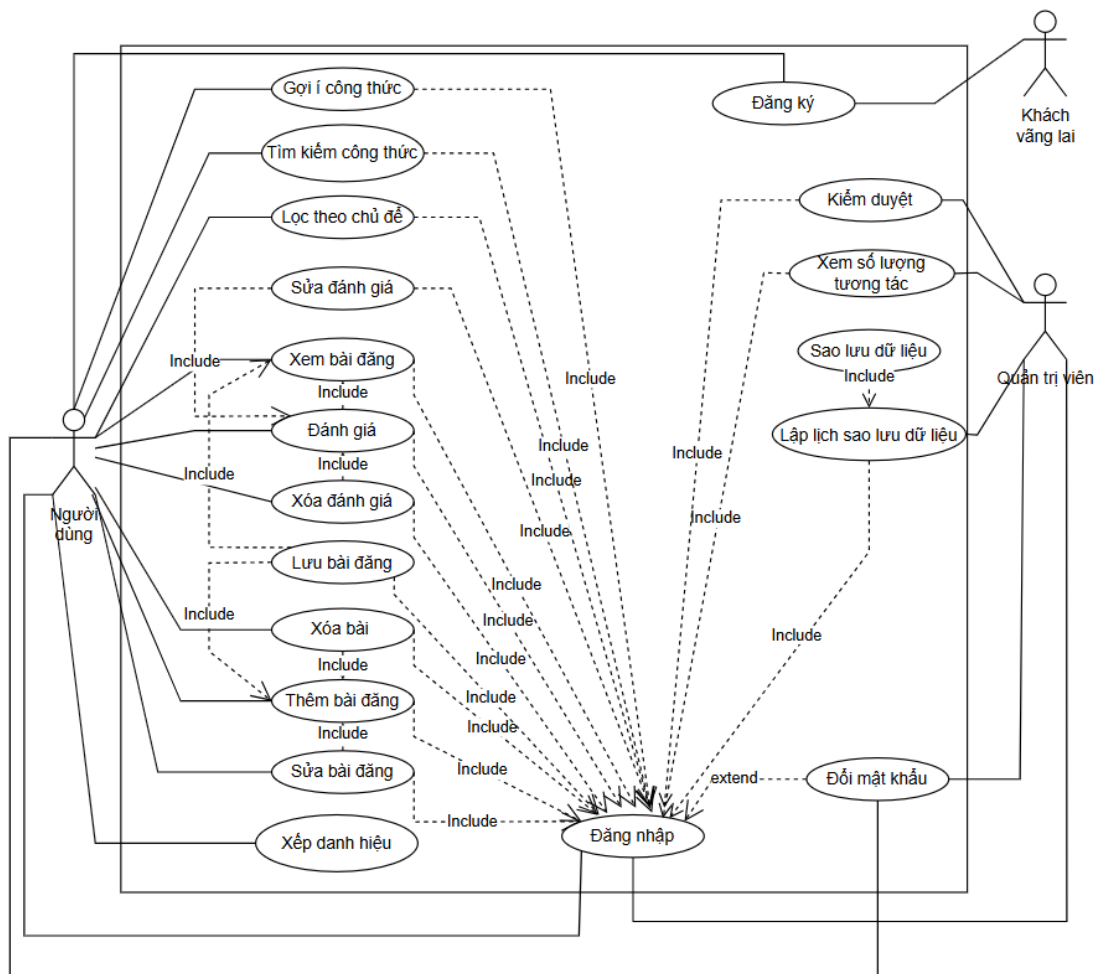
1.2.4 DFD Kiểm duyệt nội dung



1.2.5 DFD Lập lịch sao lưu dữ liệu



1.3 Sơ đồ Use case tổng quát



1.4 Miêu tả chi tiết các Use cases

1.4.1 Use case Sao lưu dữ liệu

| | | | |
|----------------|------------------|--------------------|------------------|
| Use case ID: | 01 | | |
| Use Case Name: | Sao lưu dữ liệu | | |
| Created By: | Nguyễn Hoàng Anh | Last Updated By: | Nguyễn Hoàng Anh |
| Date Created: | 17/12/2025 | Date Last Updated: | 22/12/2025 |

| | |
|------------------------|---|
| <i>Actor</i> | Hệ thống lập lịch |
| <i>Description</i> | Chức năng sao lưu dữ liệu cho phép hệ thống thực hiện sao lưu dữ liệu theo chu kỳ đã được thiết lập nhằm tạo bản sao dữ liệu phục vụ cho việc bảo vệ và khôi phục. |
| <i>Trigger</i> | Chức năng này được sử dụng khi đến thời điểm sao lưu theo chu kỳ đã lập. |
| <i>Pre-conditions</i> | Hệ thống đã có lịch sao lưu hợp lệ. |
| <i>Post conditions</i> | Dữ liệu được sao lưu vào kho lưu trữ sao lưu và gửi thông báo (thành công/thất bại) cho Quản trị viên. |
| <i>Flow of Events</i> | <ol style="list-style-type: none"> 1. Hệ thống xác định thời điểm sao lưu theo lịch. 2. Hệ thống tiến hành sao lưu dữ liệu. 3. Hệ thống lưu bản sao dữ liệu vào kho lưu trữ. 4. Hệ thống ghi lại trạng thái sao lưu và gửi thông báo cho Quản trị viên. |

1.4.2 Use case Xác định danh hiệu người dùng

| | | | |
|---------------|-------------------------------|-------------------|----------------|
| Use Case ID | 02 | | |
| Use Case Name | Xác định danh hiệu người dùng | | |
| Created By: | Vũ Thị Bảo Nhi | Last Updated By | Vũ Thị Bảo Nhi |
| Date Created: | 16/12/2025 | Date Last Updated | 18/12/2025 |

| | |
|-------|----------------|
| Actor | Lịch cập nhật. |
|-------|----------------|

| | |
|-------------------------|---|
| <i>Description</i> | -Chức năng xác định danh hiệu người dùng mô tả quá trình hệ thống tự động xác định và cập nhật danh hiệu cho người dùng dựa trên dữ liệu bài viết và đánh giá. Khi người dùng đạt tới mốc điểm tích lũy nhất định, hệ thống sẽ gán danh hiệu tương ứng và hiển thị danh hiệu đó cho người dùng. |
| <i>Trigger</i> | -Hệ thống tự động xác định danh hiệu định kỳ cuối ngày (00h) dựa trên dữ liệu người dùng. |
| <i>Pre-conditions</i> | -Người dùng đã đăng nhập vào hệ thống. -Dữ liệu bài viết và người dùng tồn tại trong hệ thống. -Hệ thống đã được định nghĩa sẵn các quy tắc xét danh hiệu. |
| <i>Post conditions</i> | -Danh hiệu người dùng được xác định chính xác theo điểm tích lũy. -Danh hiệu mới (nếu có) được cập nhật vào kho người dùng. -Danh hiệu được hiển thị trên giao diện người dùng. |
| <i>Flow of Events</i> | 1.Hệ thống truy vấn vào kho bài viết để lấy thông tin số lượng bài đăng và trung bình sao theo lịch định kỳ 00h mỗi ngày. 2. Hệ thống xử lý dữ liệu để tính toán điểm tích lũy của người dùng. 3. Hệ thống đối chiếu điểm tích lũy với các quy tắc xét danh hiệu. 4. Hệ thống xác định danh hiệu phù hợp cho người dùng, và lưu vào kho người dùng. 5. Sau đó, hệ thống cập nhật danh hiệu người dùng . 6. Hệ thống hiển thị danh hiệu trên giao diện người dùng, và khi người dùng đạt danh hiệu mới sẽ có thông báo chúc mừng. |
| <i>Alternative Flow</i> | 1.Người dùng chưa đạt mốc danh hiệu mới. + Tại bước 2.0, nếu điểm tích lũy của người dùng chưa đạt mốc danh hiệu tiếp theo: - Hệ thống không xác định danh hiệu mới. - Giữ nguyên danh hiệu hiện tại. - Chuyển sang bước 4.0 để hiển thị danh hiệu hiện tại. 2.Người dùng đã có danh hiệu cao nhất. + Tại bước 2.0, xác định danh hiệu, nếu người dùng đã sở hữu danh hiệu cao nhất trong hệ thống thì: - Hệ thống không xác định danh hiệu mới. |

| | |
|--|--|
| | <ul style="list-style-type: none"> - Giữ nguyên danh hiệu hiện tại. - Chuyển sang bước hiển thị danh hiệu. <p>3. Người dùng chưa có tương tác nào với Nostimo.</p> <ul style="list-style-type: none"> + Tại bước 2.0, nếu người dùng chưa có bất kỳ tương tác nào (không có bài đăng, không có đánh giá) thì: <ul style="list-style-type: none"> - Hệ thống xác định danh hiệu mặc định cho người dùng. (Ví dụ là Newbie) - Chuyển sang bước 3 để cập nhật danh hiệu mặc định cho người dùng. - Chuyển sang bước 4 để hiển thị danh hiệu “Newbie” trên giao diện. |
|--|--|

1.4.3 Use case Xem số lượng tương tác

| | | | |
|----------------------|------------------------|--------------------------|------------|
| <i>Use Case ID</i> | 03 | | |
| <i>Use Case Name</i> | Xem số lượng tương tác | | |
| <i>Created By:</i> | Đỗ Yến Nhi | <i>Last Updated By</i> | Đỗ Yến Nhi |
| <i>Date Created:</i> | 17/12/2025 | <i>Date Last Updated</i> | 22/12/2025 |

| | |
|------------------------|---|
| <i>Actor</i> | Quản trị viên. |
| <i>Description</i> | Chức năng này cho phép quản trị viên xem số lượng tương tác của người dùng với ứng dụng Nostimo. |
| <i>Trigger</i> | Trên thanh công cụ có nút “Trang chủ” . Quản trị viên click vào chức năng đó. |
| <i>Pre-conditions</i> | Chức năng này được sử dụng khi quản trị viên đã đăng nhập hệ thống và hệ thống đã có dữ liệu tương tác của người dùng. |
| <i>Post conditions</i> | Dữ liệu tương tác hiển thị. Quản trị viên có thể dựa vào đó để đánh giá lượng tương tác của người dùng với app Nostimo. |

| | |
|--------------------------|--|
| <i>Flow of Events</i> | 1. Quản trị viên truy cập Trang chủ . 2. Hệ thống hiển thị giao diện. 3. Quản trị viên chọn khoảng thời gian: ngày/tuần/tháng (ví dụ: “1 ngày” hoặc “1 tuần”, “1 tháng”) 4. Hệ thống kiểm tra dữ liệu, xử lý dữ liệu tương tác. 5. Hệ thống hiển thị tổng số lượng tương tác và biểu đồ số lượng tương tác theo thời gian. 6. Quản trị viên xem kết quả. |
| <i>Alternative Flows</i> | 3.1. Quản trị viên chọn mốc thời gian sai 3.1.1 Hệ thống hiển thị thông báo lỗi kiểm tra đầu vào ("Xin lỗi, vui lòng chọn mốc thời gian khác. "). |

1.4.4 Use case Kiểm duyệt nội dung

| | | | |
|---------------|---------------------|-------------------|-----------------|
| Use Case ID | 04 | | |
| Use Case Name | Kiểm duyệt nội dung | | |
| Created By: | Đặng Phương Nam | Last Updated By | Đặng Phương Nam |
| Date Created: | 16/12/2025 | Date Last Updated | 17/12/2025 |

| | |
|-----------------|---|
| Actor | Quản trị viên |
| Description | Chức năng Kiểm duyệt nội dung cho phép Quản trị viên kiểm duyệt, xử lý các bài đăng hay đánh giá có khả năng vi phạm. Quản trị viên có thể duyệt hoặc xóa các bài đăng hay đánh giá khi cần thiết |
| Trigger | Chức năng này được sử dụng khi Quản trị viên click vào giao diện chức năng “Kiểm duyệt” của trang quản lí |
| Pre-conditions | Quản trị viên xem xét duyệt hoặc xóa bài đăng hay đánh giá có dấu hiệu vi phạm |
| Post conditions | Các bài đăng hay đánh giá đã được duyệt hoặc bị xóa nếu kiểm duyệt tự động sai |
| Flow of Events | 1. Quản trị viên chọn vào mục “Kiểm duyệt” 2. Hệ thống hiển thị giao diện của kiểm duyệt, Quản trị viên có thể trở vào nút “duyet” |

| | |
|--|--|
| | <p>hoặc “xóa” các bài đăng hay đánh giá nếu có dấu hiệu vi phạm</p> <p>3. Quản trị viên có thể click vào nút “Chọn” để duyệt hoặc xóa nhiều bài đăng hoặc đánh giá cùng lúc</p> <p>4. Sau khi duyệt hoặc xóa mỗi bài đăng hay đánh giá, hệ thống hiện thị thông báo xác nhận muốn duyệt hoặc xóa.</p> <p>5. Hệ thống thông báo đã duyệt hoặc xóa bài đăng/đánh giá thành công.</p> |
|--|--|

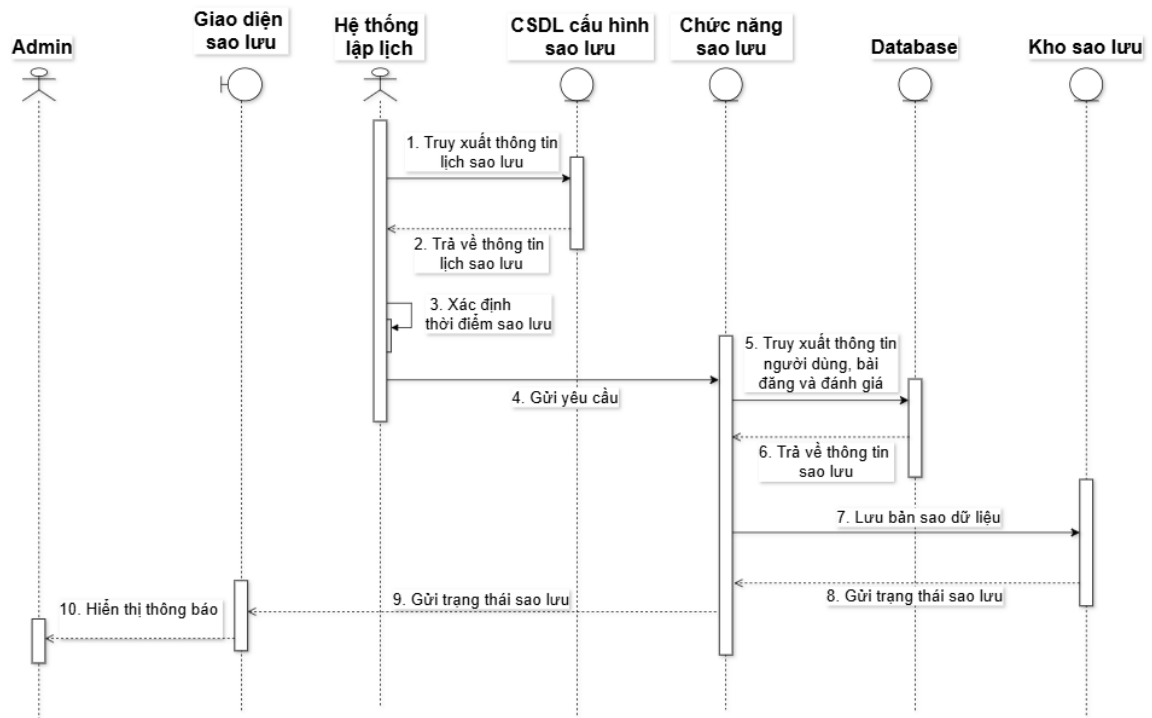
1.4.5 Use case Lập lịch sao lưu dữ liệu

| | | | |
|----------------|--------------------------|--------------------|------------------|
| Use case ID: | 05 | | |
| Use Case Name: | Lập lịch sao lưu dữ liệu | | |
| Created By: | Vi Thị Phương Nhung | Last Updated By: | Nguyễn Hoàng Anh |
| Date Created: | 16/12/2025 | Date Last Updated: | 17/12/2025 |

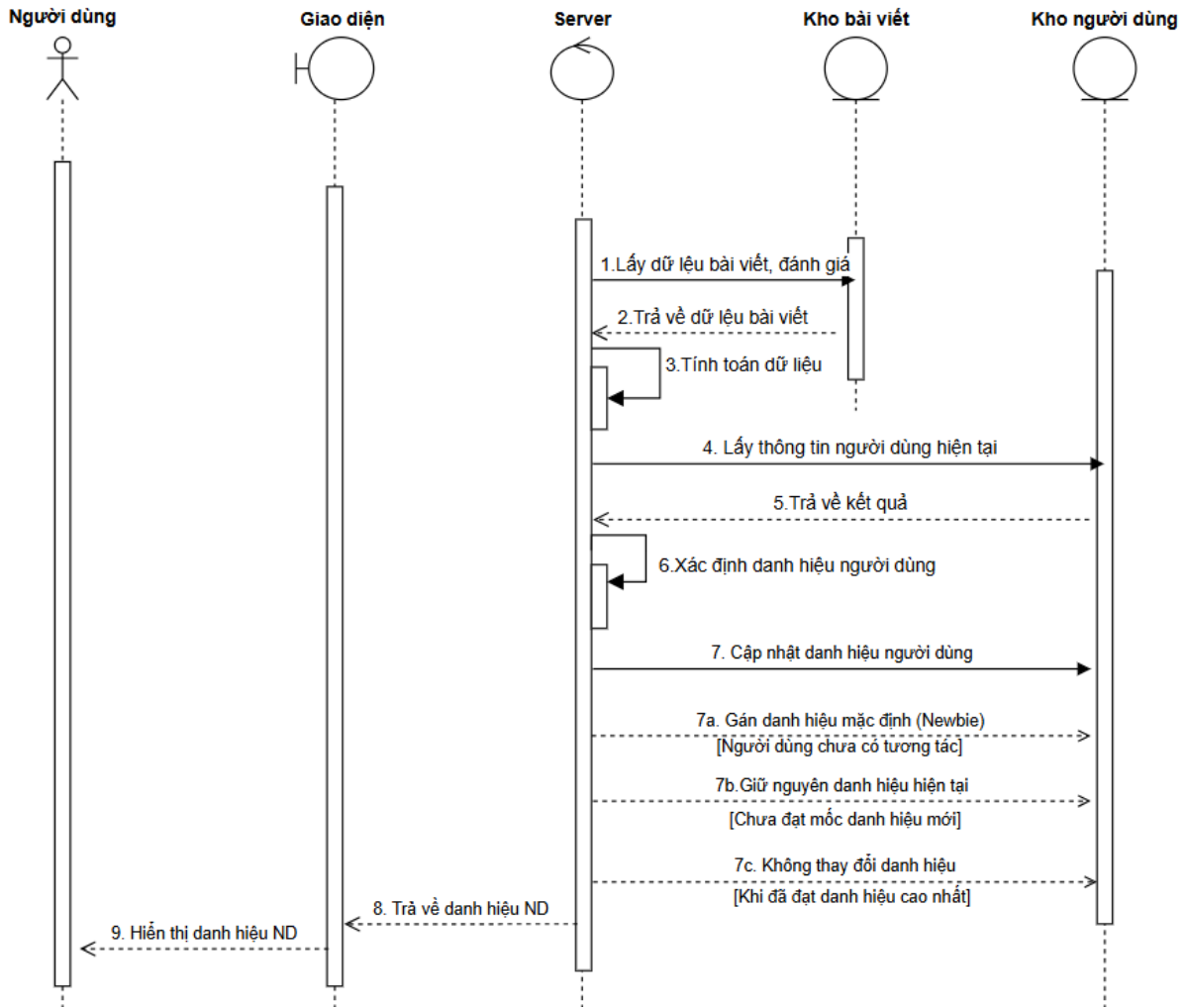
| | |
|------------------------|--|
| <i>Actor</i> | Quản trị viên. |
| <i>Description</i> | Chức năng cho phép Quản trị viên thiết lập chu kỳ và các thông tin liên quan đến việc sao lưu dữ liệu. Sau khi được thiết lập, hệ thống sẽ tự động thực hiện sao lưu dữ liệu theo lịch đã định trước nhằm đảm bảo an toàn và tránh mất mát dữ liệu. |
| <i>Trigger</i> | Chức năng này được sử dụng khi Quản trị viên mở giao diện sao lưu dữ liệu và điền vào form để thiết thời gian sao lưu định kỳ. |
| <i>Pre-conditions</i> | Quản trị viên đã đăng nhập và có quyền quản trị hệ thống. |
| <i>Post conditions</i> | Quản trị viên thiết lập lịch trình sao lưu thành công, hệ thống sẵn sàng tự động sao lưu theo lịch đã thiết lập. |
| <i>Flow of Events</i> | <ol style="list-style-type: none"> 1. Mở giao diện sao lưu dữ liệu. 2. Quản trị viên chọn tần suất sao lưu (hàng ngày, hàng tuần, hàng tháng, tùy chỉnh) 3. Quản trị viên chọn thời gian sao lưu. 4. Quản trị viên chọn ngày bắt đầu sao lưu. 5. Quản trị viên chọn “Lưu” để lưu lịch sao lưu. 6. Hệ thống tự động sao lưu theo lịch đã lập. |

1.5 Sơ đồ Trình tự (SD)

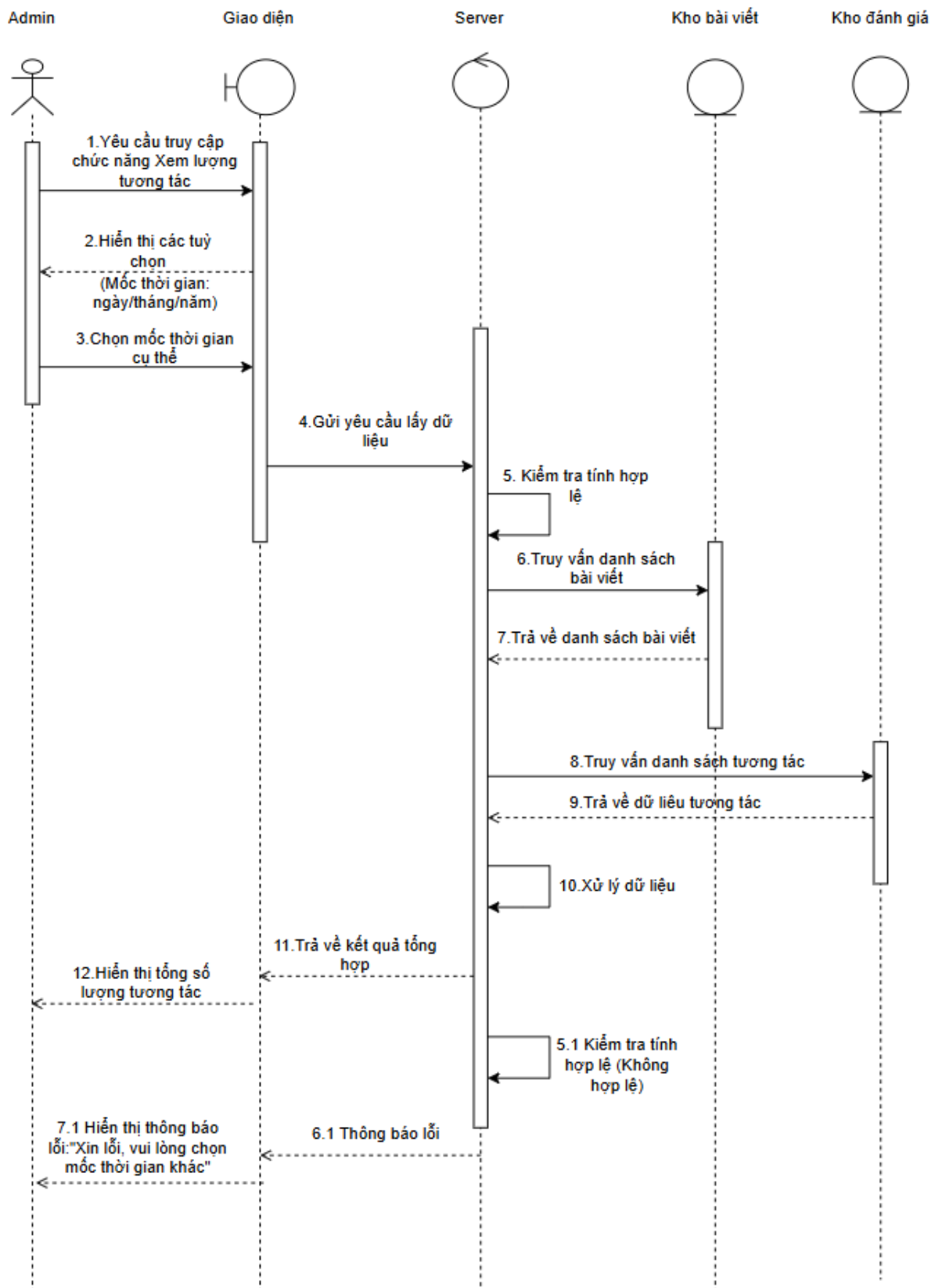
1.5.1 SD Sao lưu dữ liệu



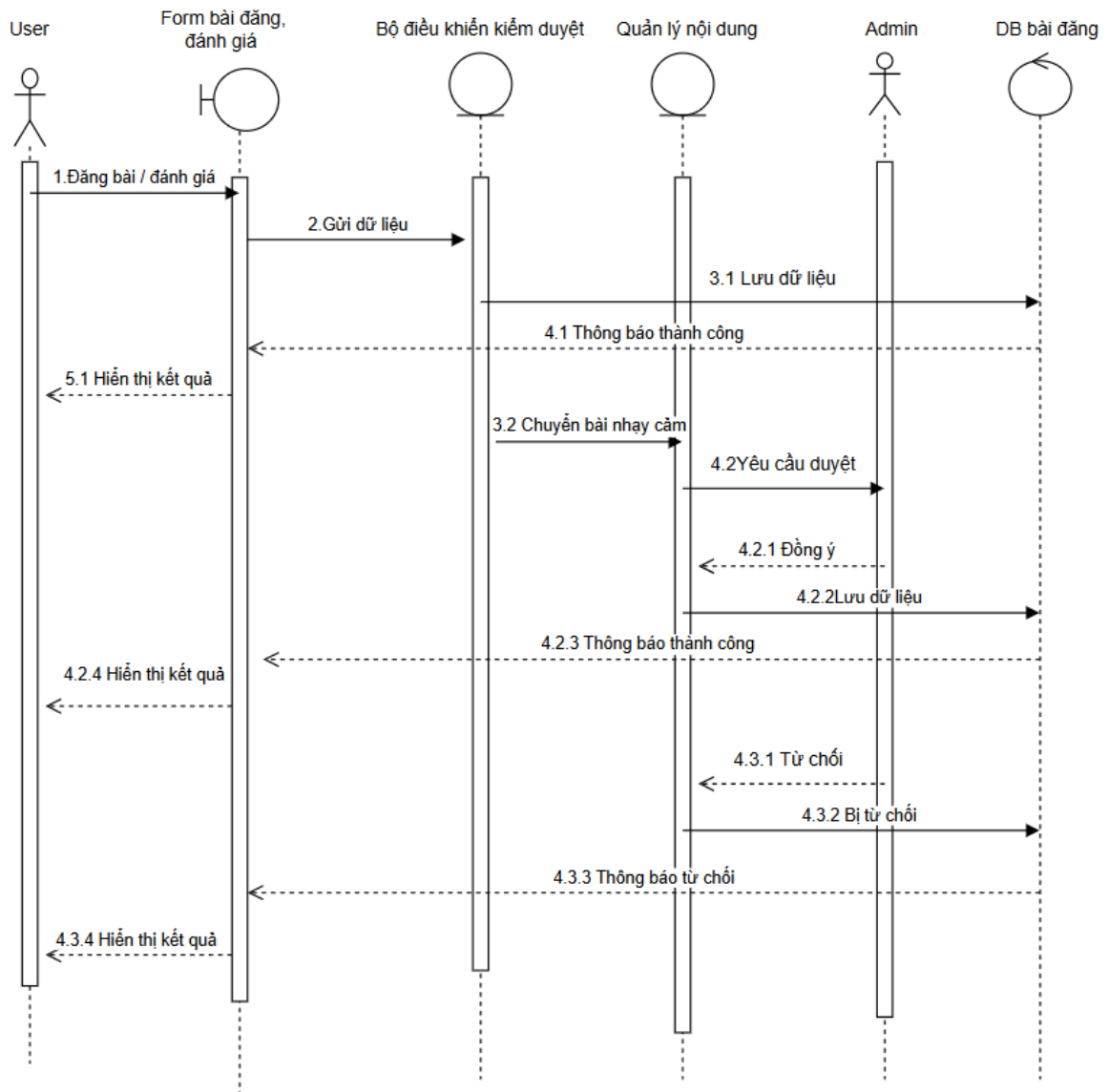
1.5.2 SD Xác định danh hiệu người dùng



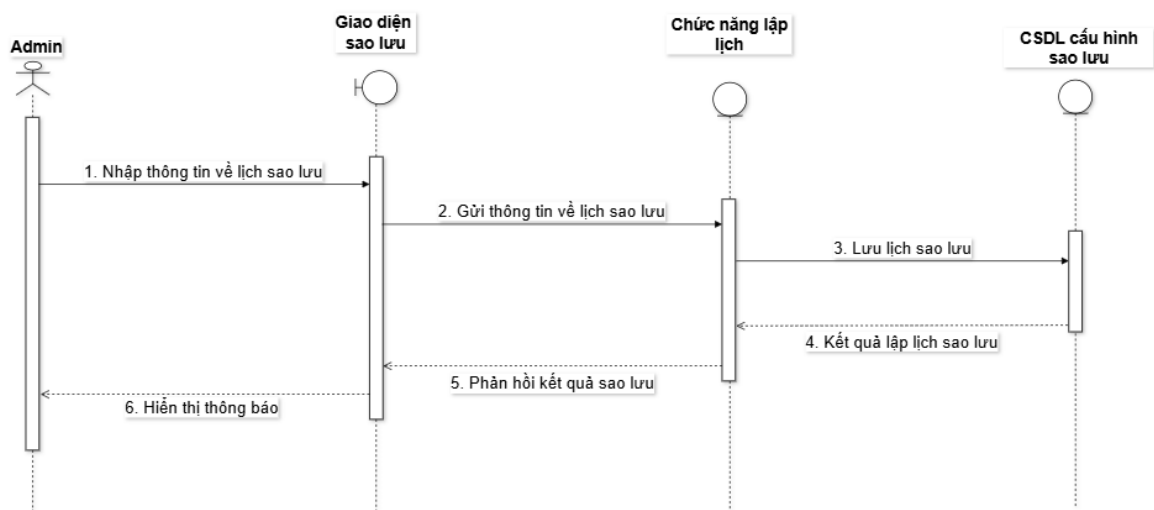
1.5.3 SD Xem số lượng tương tác



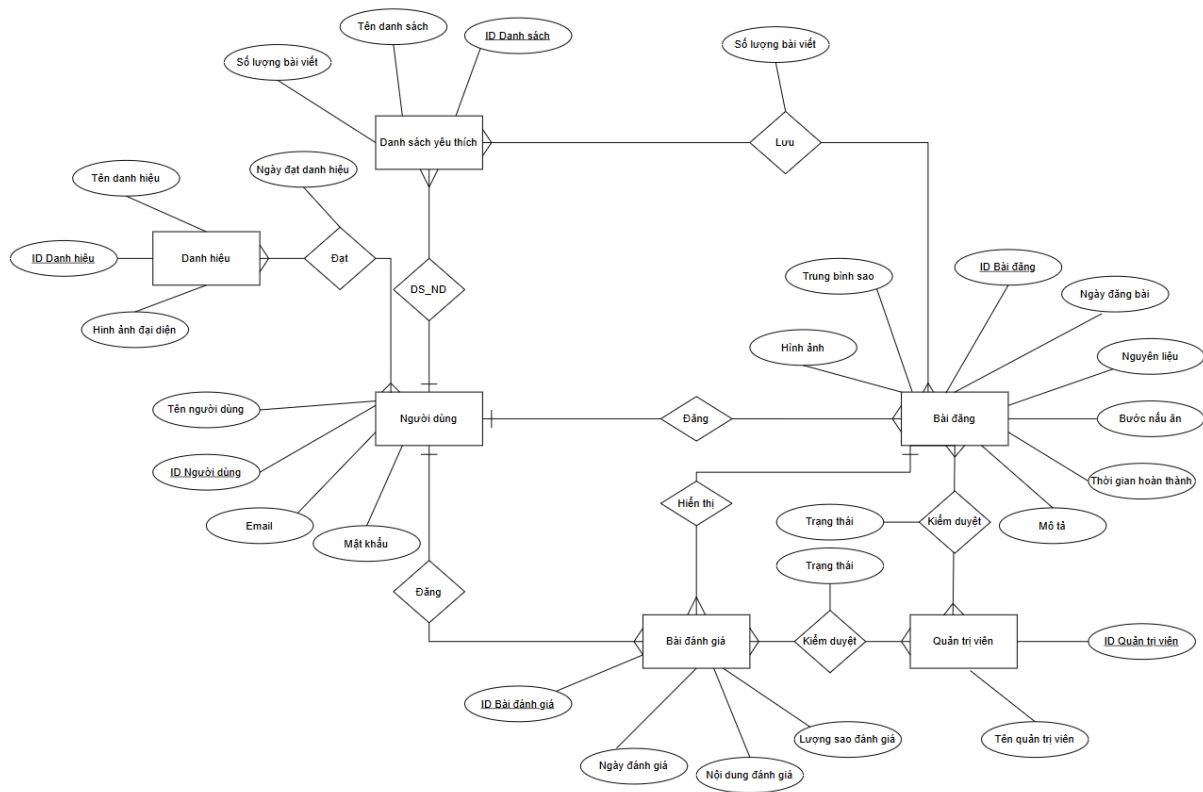
1.5.4 SD Kiểm duyệt nội dung



1.5.5 SD Lập lịch sao lưu dữ liệu

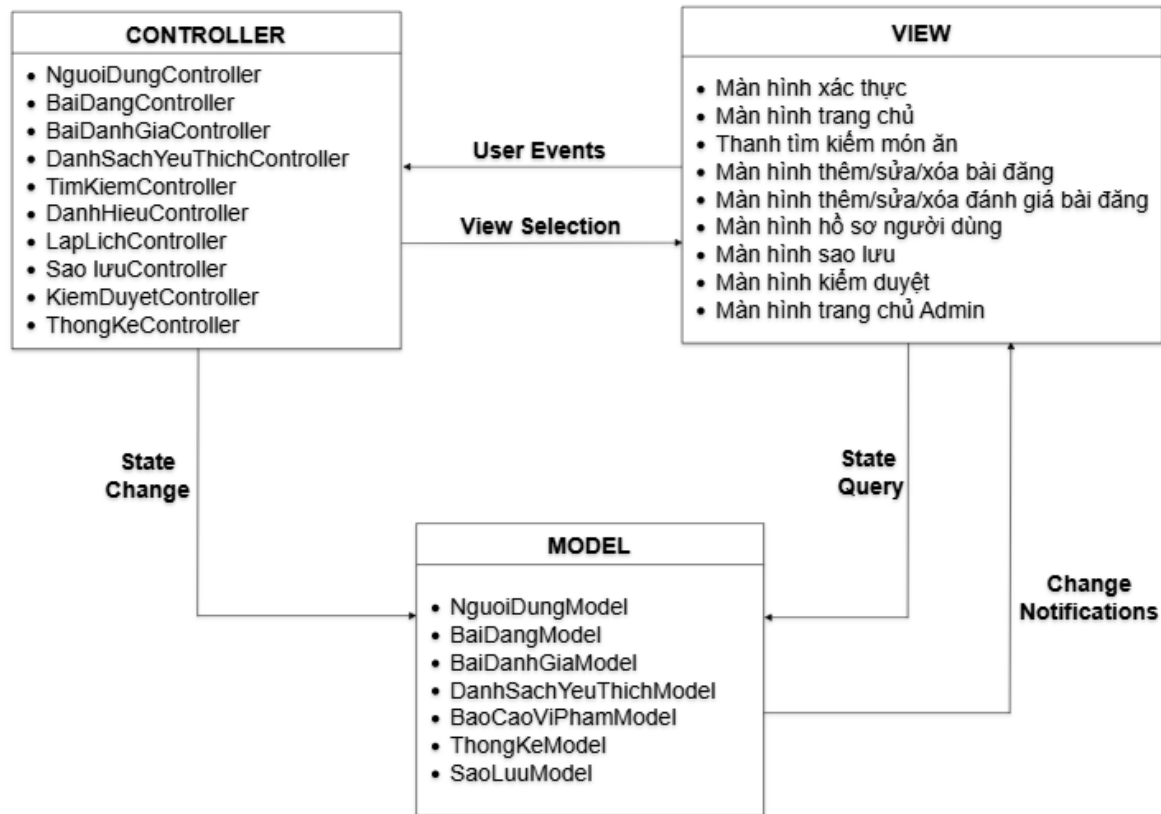


1.6 Biểu đồ quan hệ thực thể (ERD)



II. Thiết kế phần mềm

1. Thiết kế Kiến trúc



A. VIEW - Tầng giao diện người dùng

Chức năng chính:

- Hiển thị dữ liệu từ hệ thống cho người dùng và tiếp nhận các thao tác từ họ.

Thành phần cụ thể trong hệ thống:

- Màn hình xác thực (Giao diện đăng ký/ đăng nhập/ đăng xuất)
- Trang chủ (hiển thị bài đăng, công thức)
- Chức năng thêm, sửa, xóa bài đăng
- Chức năng thêm, xóa, sửa đánh giá bài đăng
- Trang hồ sơ người dùng (có thông tin người dùng, danh sách món ăn yêu thích, ...)
- Màn hình sao lưu (Hiển thị lịch sử sao lưu, khung lập lịch sao lưu)
- Màn hình kiểm duyệt (Hiển thị các bài đăng có dấu hiệu sai phạm và bình luận tương ứng chờ Admin xét duyệt.
- Màn hình trang chủ Admin (Hiển thị thông tin tổng số lượng tương tác của người dùng và biểu đồ số lượng tương tác theo thời gian).

Vai trò trong mô hình:

- Khi người dùng thao tác (click, nhập dữ liệu, ...) trên View, View sẽ tạo User Event gửi đến Controller.
- Sau khi dữ liệu trong Model thay đổi, View nhận Change Notification từ Model để cập nhật lại giao diện hiển thị cho người dùng.

B. Controller - Tầng điều khiển

Chức năng chính:

- Là bộ phận trung gian giữa View và Model, chịu trách nhiệm điều phối luồng dữ liệu giữa View và Model.

Thành phần cụ thể trong hệ thống:

- `NguoiDungController`: Xử lý đăng ký, đăng nhập, xác thực, đăng xuất.
- `BaiDangController`: Xử lý thêm/sửa/xóa bài đăng.
- `BaiDanhGiaController`: Xử lý thêm/sửa/xóa bài đánh giá.
- `DanhSachYeuThichController`: Thêm/xóa món ăn yêu thích.
- `TimKiemController`: Xử lý tìm kiếm món ăn bằng từ khóa, theo chủ đề và gợi ý món ăn bằng cách nhập/chọn nguyên liệu.
- `DanhHieuController`: Xác định và gán danh hiệu người dùng.
- `LapLichController`: Lập và quản lý lịch sao lưu.
- `SaoLuuController`: Thực hiện sao lưu dữ liệu định kỳ.
- `KiemDuyetController`: Kiểm duyệt bài đăng, bình luận, báo cáo vi phạm.
- `ThongKeController`: Tổng hợp số liệu, tương tác (số lượt lưu bài đăng, số lượt đánh giá, số bài đăng) phục vụ thống kê.

Vai trò trong mô hình:

- Xử lý logic ở mức điều phối (User chọn tìm món bằng tìm nguyên liệu → Controller sẽ gọi Model xử lý).
- Nhận User Events từ View (click, nhập dữ liệu, yêu cầu tìm kiếm...).
- Gửi lệnh State Change đến Model để truy vấn hoặc cập nhật dữ liệu.
- Gọi các hàm tương ứng của Model
- Khi cần hiển thị, yêu cầu Model trả dữ liệu hiển thị cho View.

C. Model -Tầng dữ liệu

Chức năng chính:

- Quản lý, lưu trữ và xử lý dữ liệu của toàn hệ thống.

Thành phần cụ thể trong hệ thống:

- `NguoIDungModel`: Lưu trữ thông tin tài khoản, email, mật khẩu, ...
- `BaiDangModel`: Lưu trữ công thức, nguyên liệu, lượt tương tác, ...
- `BaiDanhGiaModel`: Lưu trữ nội dung đánh giá, lượng sao đánh giá, ...
- `DanhSachYeuThichModel`: Lưu trữ bài đăng, các thể loại danh sách yêu thích, ...
- `BaoCaoViPhamModel`: Lưu trữ thông tin báo cáo vi phạm của bài đăng, bình luận.
- `ThongKeModel`: Lưu trữ dữ liệu phục vụ thống kê.
- `SaoLuuModel`: Lưu trữ dữ liệu sao lưu hệ thống.

Vai trò trong mô hình:

- Model cung cấp dữ liệu cho Controller và View khi có yêu cầu (State Query)
- Model tiếp nhận yêu cầu thay đổi dữ liệu từ Controller (State Change)
- Thực hiện cập nhật dữ liệu vào cơ sở dữ liệu (ví dụ là thêm, xóa, sửa bài đăng...)
- Gửi thông báo cập nhật (Change Notification) lên View khi dữ liệu thay đổi để View hiển thị thông tin mới nhất.

Luồng hoạt động chi tiết giữa các thành phần:

1. Người dùng (bao gồm người dùng thường và quản trị viên) thao tác trên giao diện (View)
- Người dùng thực hiện các hành động trên ứng dụng như:
- Đăng nhập/Đăng ký
 - Tìm kiếm món ăn/Nguyên liệu
 - Thêm/Sửa/Xóa bài đăng
 - Đánh giá món ăn
 - Lưu món ăn yêu thích

- Xem danh hiệu cá nhân
- Quản trị hệ thống (sao lưu, kiểm duyệt, thống kê).

=> Các thao tác này tạo ra User Event tại View.

2. View gửi User Event đến Controller

- View không xử lý logic nghiệp vụ.
- View chỉ chuyển sự kiện người dùng (User Event) đến Controller tương ứng.

Ví dụ: Nhấn “Đánh giá” → gửi sự kiện đến BaiDanhGiaController

Nhấn “Tìm kiếm” → gửi sự kiện đến TimKiemController

Nhấn “Xem thống kê” → gửi sự kiện tới ThongKeController

3. Controller xử lý sự kiện và thực hiện nghiệp vụ

- Controller tiếp nhận User Event.
- Phân tích yêu cầu, kiểm tra dữ liệu đầu vào.
- Xác định hành động cần thực hiện (thêm, xóa, sửa...)

→ Controller không truy vấn dữ liệu để hiển thị

→ Controller chỉ gửi yêu cầu thay đổi trạng thái đến Model.

4. Controller tương tác với Model

- Gửi yêu cầu đọc, ghi hoặc cập nhật dữ liệu (State Change).

5. Model thực hiện thay đổi dữ liệu

- Model xử lý logic nghiệp vụ liên quan đến dữ liệu.
- Cập nhật trạng thái dữ liệu hệ thống, sau đó gửi Change Notification đến View.

6. View nhận thông báo từ Model

- View nhận được thông báo rằng đã dữ liệu đã cập nhật.

7. View thực hiện State Query để lấy dữ liệu mới

- Khi nhận Change Notification, View sẽ:

- Gửi yêu cầu State Query đến Model
- Lấy dữ liệu mới nhất

Ví dụ: Lấy danh sách đánh giá mới, lấy danh sách món ăn phù hợp, lấy thông tin hồ sơ người dùng, ..

8. View cập nhật giao diện người dùng

- View hiển thị dữ liệu mới lên màn hình.
- Người dùng nhìn thấy kết quả cập nhật ngay lập tức.

2. Thiết kế Hướng đối tượng (NULL)

3. Thiết kế Dữ liệu

3.1 Các thực thể và thuộc tính

Nguoidung (ID_nguoidung, Tennguoidung, Email, Matkhau)

Baidang (ID_baidang, Ngaydangbai, Trungbinhsao, Nguyenlieu, Buocnauan, Thoigianhoanthanh, Hinhanh, Mota)

Danhsachyeuthich (ID_danhsach, Tendanhsach, Soluongbaiviet)

Baidanhgia (ID_baidanhgia, Ngaydanhgia, Luongsaodanhgia, Noidungdanhgia)

Danhchieu (ID_Danhchieu, Tendanhchieu, Hinhanhdaidien)

Quantrivien (ID_Quantrivien, Tenquantrivien)

3.2 Mối liên kết

1. Nguoidung và Baidang

- Liên kết: 1 - n (Một người dùng có thể đăng nhiều bài đăng)

2. Nguoidung và Danhsachyeuthich

- Liên kết: 1 - n (Một người dùng có thể có nhiều danh sách yêu thích)

3. Danhsachyeuthich và Baidang

- Liên kết: m - n

4. Nguoidung và Baidanhgia

- Liên kết: 1 - n (Một người dùng có thể đăng nhiều bài đánh giá)

5. Baidang và Baidanhgia

- Liên kết: 1 - n (Một bài đăng có thể có nhiều đánh giá)

6. Quantrivien và Baidang

- Liên kết: m-n

7. Quantrivien và Baidanhgia

- Liên kết: m-n

8. Nguoidung và Danhchieu

- Liên kết: m - n (Một người dùng có thể có nhiều danh hiệu, một danh hiệu có thể dành cho nhiều người dùng)

3.3 Chuyển sang quan hệ

Kết quả:

Nguoidung(ID_nguoidung, Tennguoidung, Email, Matkhau)

Baidang(ID_baidang, ID_nguoidung, Ngaydangbai, Trungbinhsao, Nguyenlieu, Buocnauan, Thoigianhoanthanh, Hinhanh, Mota)

Danhsachyeuthich(ID_danhsach, Tendanhsach, Soluongbaiviet, ID_nguoidung)

Baidanhgia(ID_baidanhgia, ID_baidang, ID_nguoidung, Ngaydanhgia, Luongsaoanhgia, Noidunganhgia)

Danhhiệu(ID_Danhhiệu, Tendanhhiệu, Hinhanhdaidien)

Quantrivien(ID_quantrivien, Tenquantrivien)

Kiemduyetbaidang(ID_baidang, ID_quantrivien, Trangthai)

Kiemduyetbaidanhgia(ID_baidanhgia, ID_quantrivien, Trangthai)

ChitietDSyeuthich (ID_danhsach, ID_baidang, Soluongbaidang)

Danhhiệunguoidung (ID_nguoidung, ID_danhhiệu, Ngaydatdanhhiệu)

3.4 Chuẩn hóa dữ liệu

3.4.1 Xác định phụ thuộc hàm

- ID Người dùng xác định một người dùng duy nhất bao gồm tên người dùng, mật khẩu, email.
- ID Bài đăng xác định một bài đăng duy nhất bao gồm các thông tin về bài đăng: ngày đăng bài, nguyên liệu, bước nấu ăn, thời gian hoàn thành, mô tả, hình ảnh, trung bình sao.
- ID Bài đánh giá xác định một bài đánh giá duy nhất gồm các thông tin: lượng sao đánh giá, nội dung đánh giá, ngày đánh giá.
- ID Danh sách yêu thích xác định một danh sách yêu thích bao gồm: Tên danh sách, ID người dùng.
- ID Chi tiết danh sách xác định một danh sách bao gồm: ID bài đăng và số lượng bài đăng.
- ID Danhhiệu xác định duy nhất các thông tin về danh hiệu như Tên danh hiệu, hình ảnh đại diện.
- ID Quản trị viên xác định duy nhất một tên quản trị viên.
- ID Bài đăng và ID Quản trị viên xác định trạng thái của bài đăng.
- ID Bài đánh giá và ID Quản trị viên xác định trạng thái của bài đăng.
- ID Người dùng và ID Danh hiệu xác định ngày đạt danh hiệu, từ ID Người dùng cũng xác định được ID Danh hiệu.

3.4.2 Xét các dạng chuẩn

- **Người dùng**(ID Người dùng, Tên người dùng, Email, Mật khẩu)

Đặt ID người dùng = A, Tên người dùng = B, Email = C, Mật khẩu = D;

Phụ thuộc hàm: $F = \{A \rightarrow B, A \rightarrow C, A \rightarrow D, C \rightarrow A\}$

=> Đạt 1NF vì không có thuộc tính đa trị

=> Đạt 2NF vì các thuộc tính không khóa phụ thuộc đầy đủ vào khóa chính A

=> Đạt 3NF vì các thuộc tính không khóa phụ thuộc trực tiếp vào khóa chính A

=> Không đạt BCNF vì có thuộc tính khóa nào phụ thuộc vào thuộc tính không khóa

- **Bài đăng**(ID bài đăng, Ngày đăng bài, ID người dùng, Nguyên liệu, Bước nấu ăn, Thời gian hoàn thành, Mô tả, Hình ảnh, Trung bình sao);

Đặt ID bài đăng = A, Ngày đăng bài = B, ID người dùng = C, Nguyên liệu = D, Bước nấu ăn = E, Thời gian hoàn thành = F, Mô tả = G, Hình ảnh = H, Trung bình sao = I)

$F = \{A \rightarrow B, A \rightarrow C, A \rightarrow D, A \rightarrow E, A \rightarrow F, A \rightarrow G, A \rightarrow H, A \rightarrow I, A \rightarrow K\}$

=> Không đạt 1NF vì có thuộc tính đa trị (có thể có nhiều nguyên liệu)

=> Không đạt 2NF vì không đạt 1NF

=> Không đạt 3NF

=> Không đạt BCNF

- **Bài đánh giá**(ID bài đăng, Lượng sao đánh giá, Nội dung đánh giá, Ngày đánh giá, ID bài đánh giá);

Đặt ID bài đánh giá = A, Lượng sao đánh giá = B, Nội dung đánh giá = C, Ngày đánh giá = D, ID bài đăng = E; $F = \{A \rightarrow B, A \rightarrow C, A \rightarrow D, A \rightarrow E\}$

=> Đạt 1NF vì không có thuộc tính đa trị

=> Đạt 2NF vì các thuộc tính không khóa phụ thuộc đầy đủ vào khóa chính A

=> Đạt 3NF vì các thuộc tính không khóa phụ thuộc trực tiếp vào khóa chính A

=> Đạt BCNF vì không có thuộc tính khóa nào phụ thuộc vào thuộc tính không khóa

- **Danh sách yêu thích**(ID người dùng, Tên danh sách, ID danh sách)

Đặt ID danh sách = A, ID người dùng = B, Tên danh sách = C

$F = \{A \rightarrow B, A \rightarrow C\}$

=> Đạt 1NF vì không có thuộc tính đa trị

=> Đạt 2NF vì các thuộc tính không khóa phụ thuộc đầy đủ vào khóa chính A

=> Đạt 3NF vì các thuộc tính không khóa phụ thuộc trực tiếp vào khóa chính A

=> Đạt BCNF vì không có thuộc tính khóa nào phụ thuộc vào thuộc tính không khóa.

- **Chi tiết danh sách**(ID danh sách, ID bài đăng, Số lượng bài đăng)

Đặt ID danh sách = A, ID bài đăng = B, Số lượng bài đăng = C;

$F = \{AB \rightarrow C\}$

=> Đạt 1NF vì không có thuộc tính đa trị

=> Đạt 2NF vì các thuộc tính không khóa phụ thuộc đầy đủ vào khóa chính

=> Đạt 3NF vì các thuộc tính không khóa phụ thuộc trực tiếp vào khóa chính

=> Đạt BCNF vì không có thuộc tính khóa nào phụ thuộc vào thuộc tính không khóa

- **Danh hiệu**(ID_Danh hiệu, Tendanh hiệu, Hinhanhdaidien)

Đặt ID_Danh hiệu = A , Tendanh hiệu = B , Hinhanhdaidien = C

$F = \{A \rightarrow B, A \rightarrow C\}$

=> Đạt 1NF vì không có thuộc tính đa trị

=> Đạt 2NF vì các thuộc tính không khóa phụ thuộc đầy đủ vào khóa chính A

=> Đạt 3NF vì các thuộc tính không khóa phụ thuộc trực tiếp vào khóa chính A

=> Đạt BCNF vì không có thuộc tính khóa nào phụ thuộc vào thuộc tính không khóa

- **Quantrivien**(ID_quantrivien, Tenquantrivien)

Đặt ID_quantrivien = A, Ten quantrivien = B

$F = \{A \rightarrow B\}$

=> Đạt 1NF vì không có thuộc tính đa trị

=> Đạt 2NF vì các thuộc tính không khóa phụ thuộc đầy đủ vào khóa chính A

=> Đạt 3NF vì các thuộc tính không khóa phụ thuộc trực tiếp vào khóa chính A

=> Đạt BCNF vì không có thuộc tính khóa nào phụ thuộc vào thuộc tính không khóa

Kiemduyetbaidang(ID_baidang, ID_quantrivien, Trangthai)

Đặt ID_baidang = A, ID_quantrivien = B, Trangthai = C

$F = \{AB \rightarrow C\}$

=> Đạt 1NF vì không có thuộc tính đa trị

=> Đạt 2NF vì các thuộc tính không khóa phụ thuộc đầy đủ vào khóa chính A

=> Đạt 3NF vì các thuộc tính không khóa phụ thuộc trực tiếp vào khóa chính A

=> Đạt BCNF vì không có thuộc tính khóa nào phụ thuộc vào thuộc tính không khóa

Danhhieunguoidung (ID_nguoidung, ID_danhchieu, Ngaydatdanhchieu)

Đặt ID_nguoidung = A, ID_danhchieu = B, Ngaydatdanhchieu = C;

$F = \{AB \rightarrow C, A \rightarrow B\}$

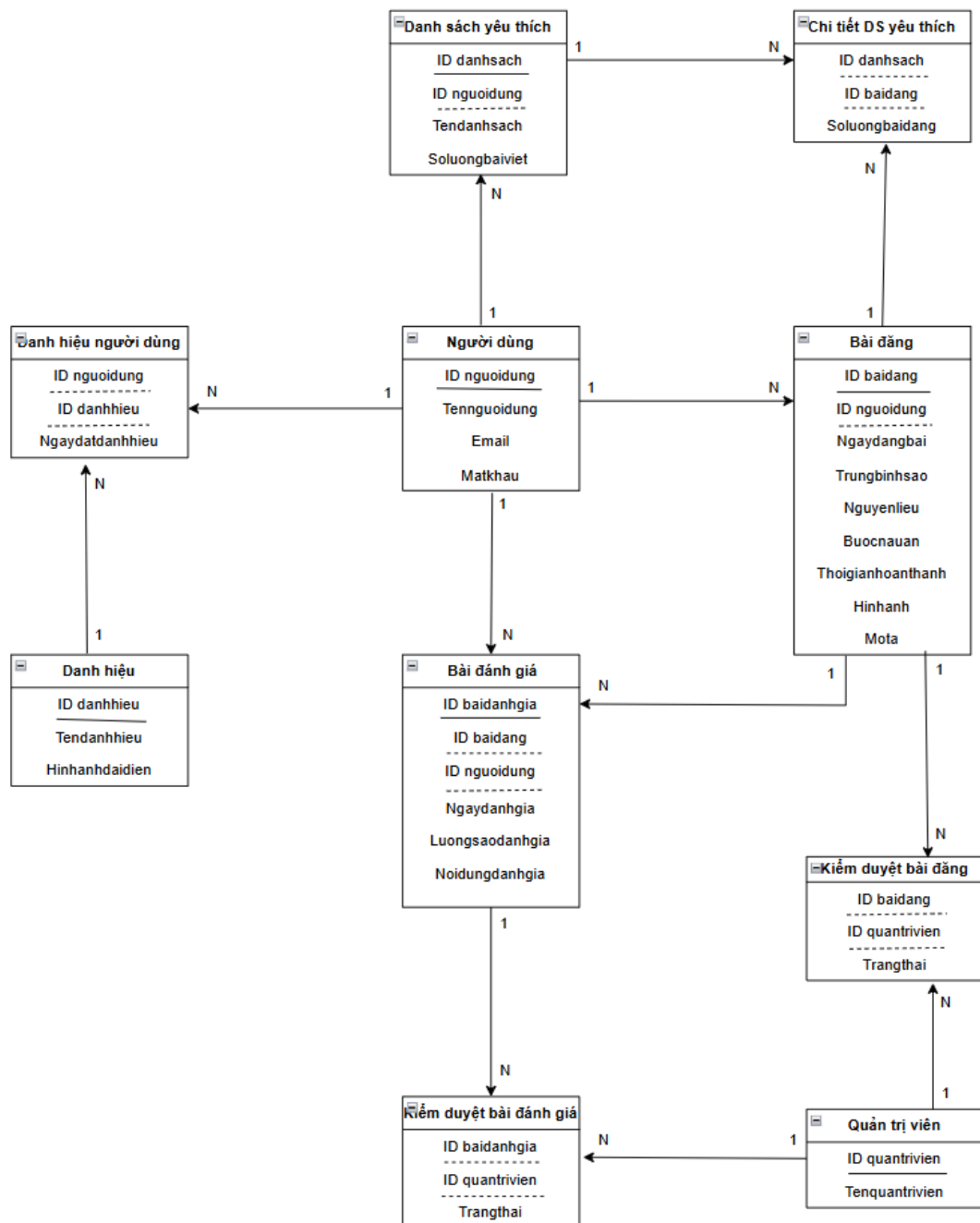
=> Đạt 1NF vì không có thuộc tính đa trị

=> Đạt 2NF vì các thuộc tính không khóa phụ thuộc đầy đủ vào khóa chính A

=> Đạt 3NF vì các thuộc tính không khóa phụ thuộc trực tiếp vào khóa chính A

=> Đạt BCNF vì không có thuộc tính khóa nào phụ thuộc vào thuộc tính không khóa

3.5 Sơ đồ quan hệ



III. Mã nguồn

[Link Code](#)

IV. Tài liệu kiểm thử

- Tham khảo báo cáo [Kiểm Thử](#)