

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 Tân 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

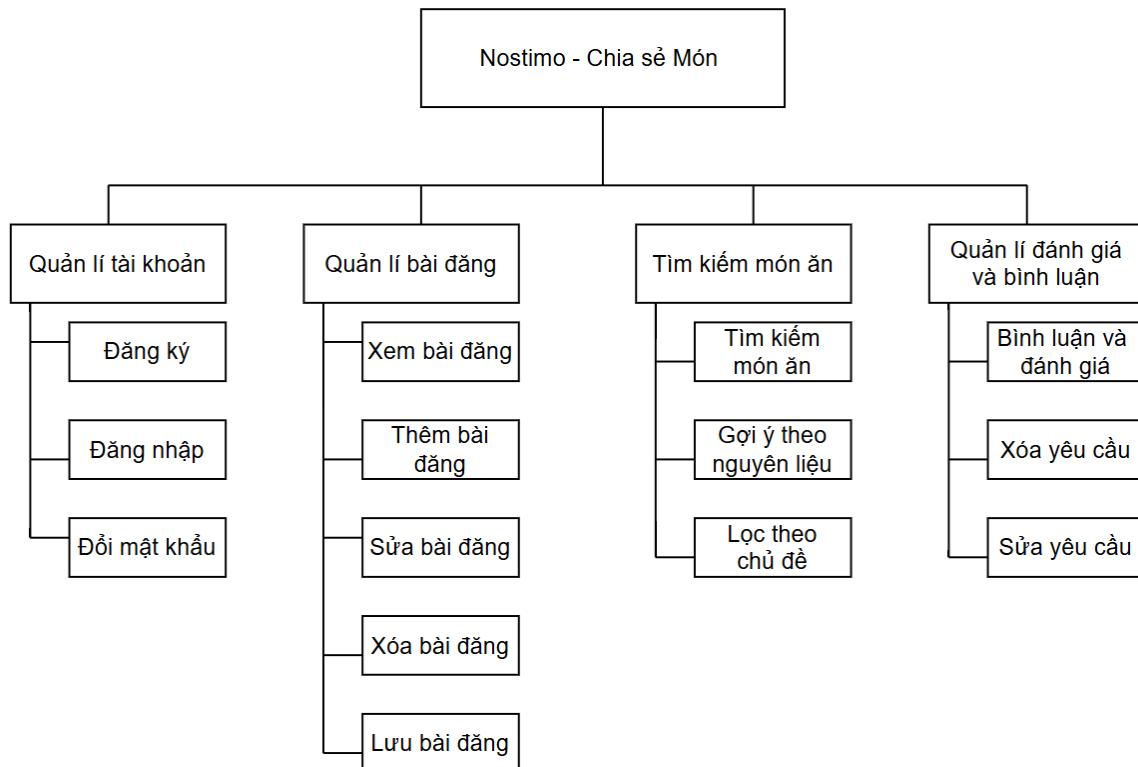
**Giảng viên hướng dẫn:** Nguyễn Thị Thu Hương

*Hà Nội, tháng 12 năm 2025*

# NHIỆM VỤ 3/4: SPRINT 1/2

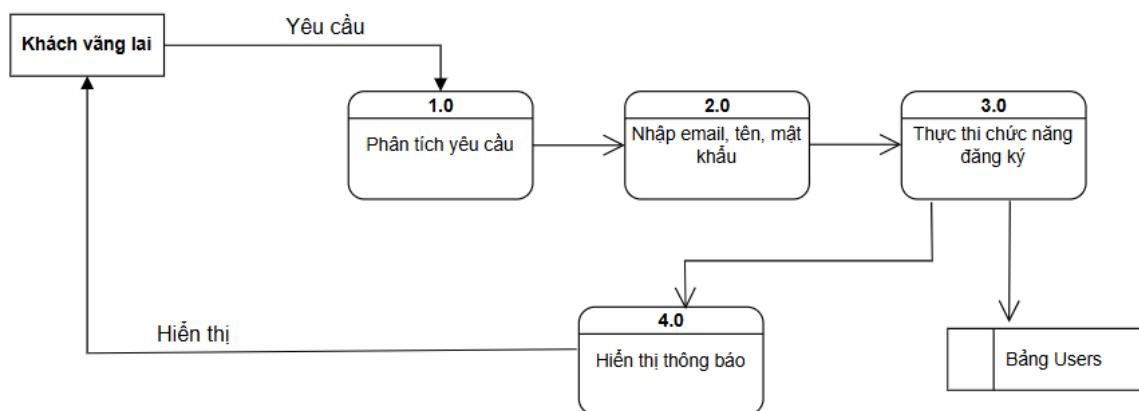
## I. Tài liệu đặc tả yêu cầu phần mềm

### 1.1 Biểu đồ Phân rã chức năng (FDD)

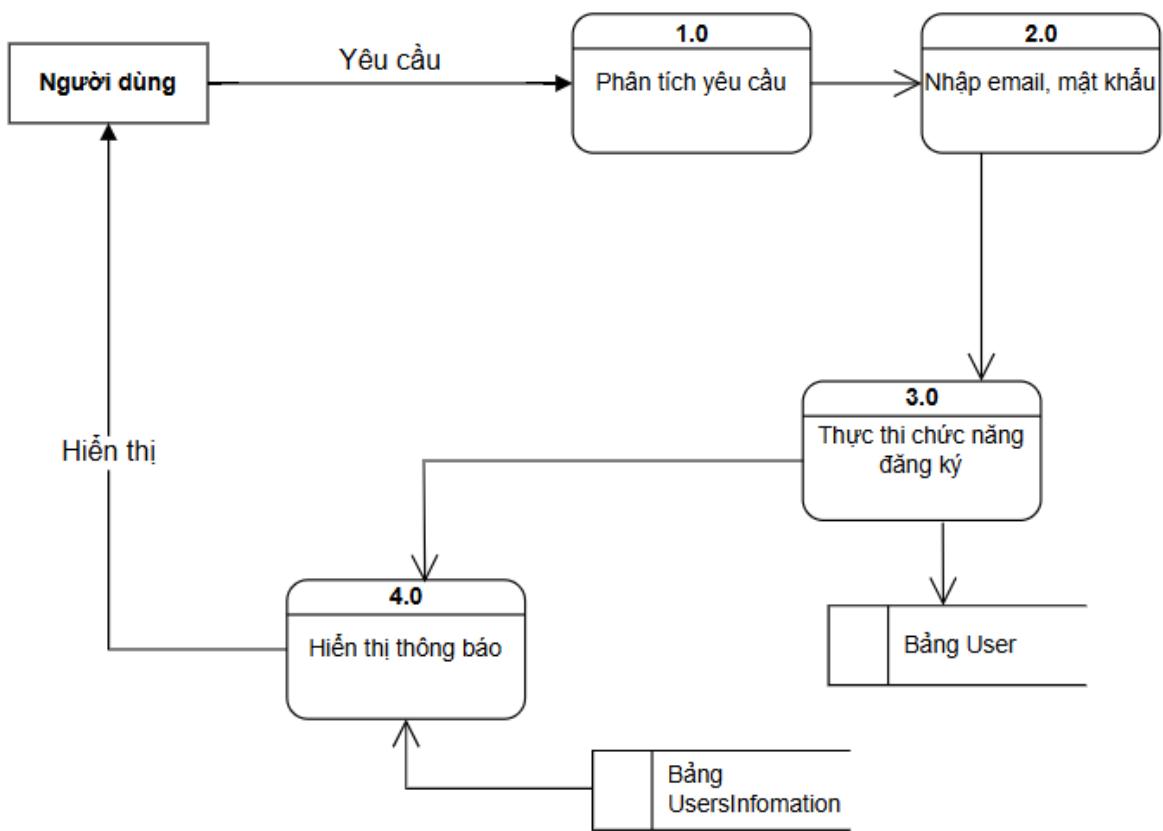


### 1.2 Biểu đồ Luồng dữ liệu (DFD)

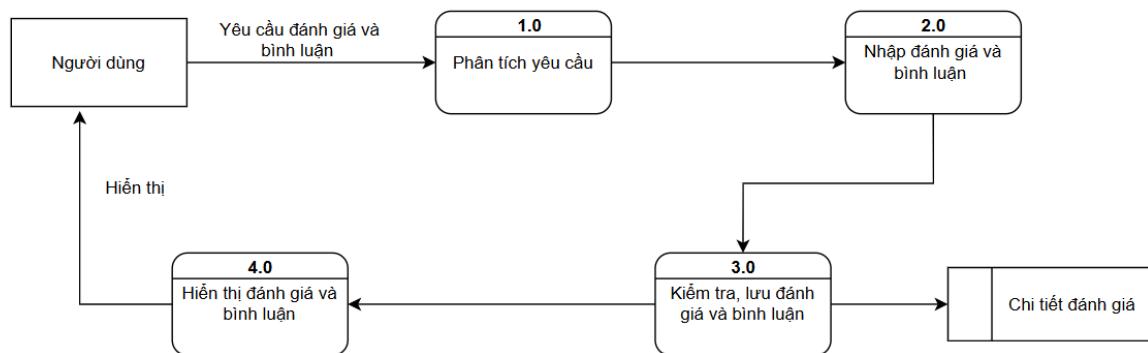
#### 1.2.1 DFD Đăng ký



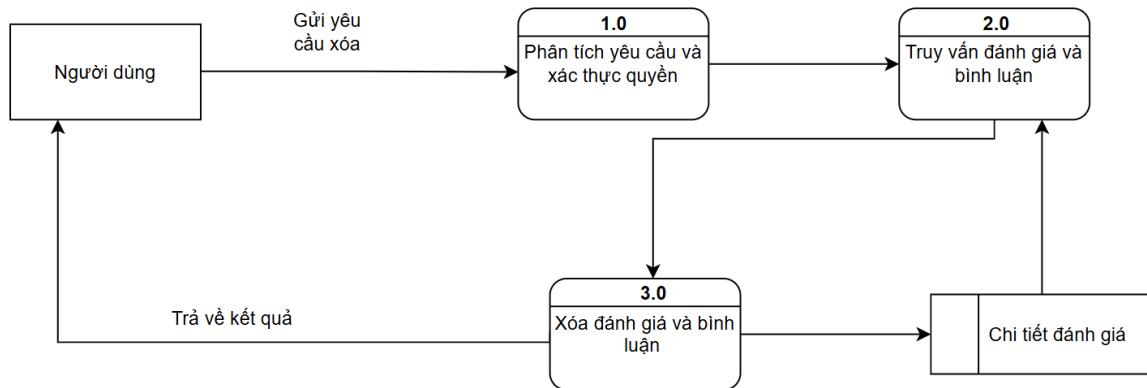
### 1.2.2 DFD Đăng nhập



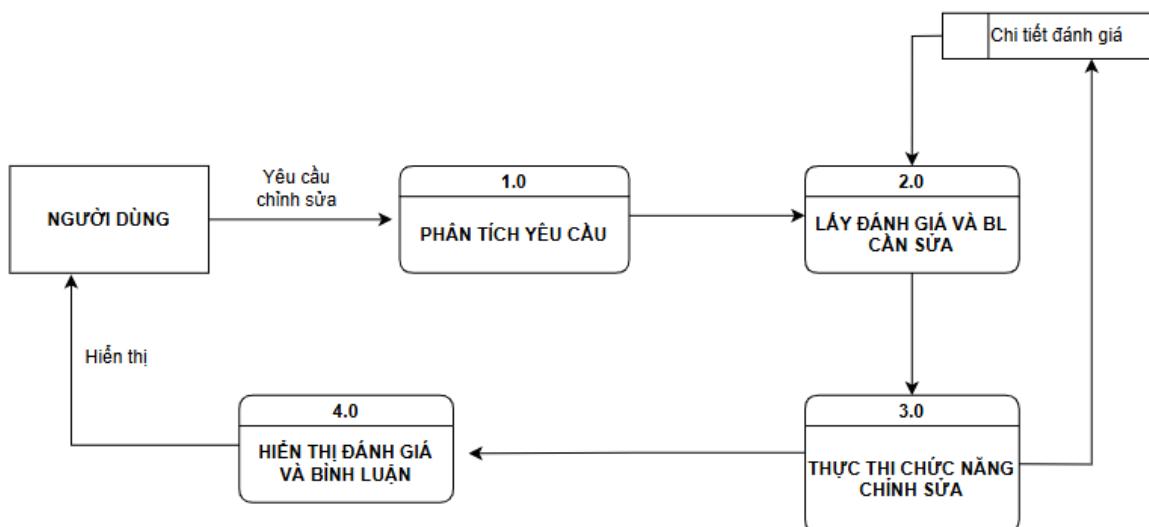
### 1.2.3 DFD Yêu cầu đánh giá và bình luận



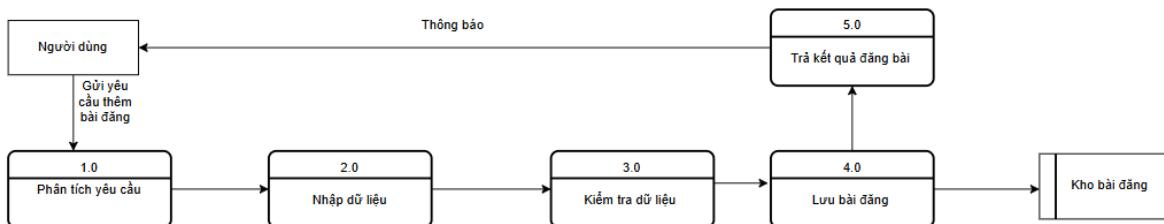
#### 1.2.4 DFD Yêu cầu xóa bình luận và đánh giá



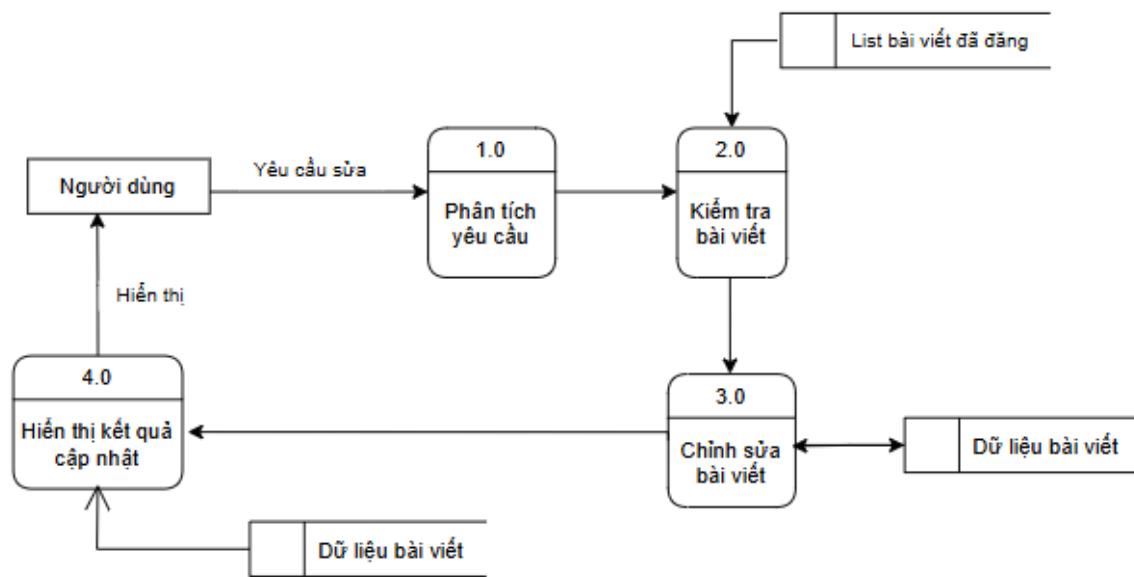
#### 1.2.5 DFD Sửa yêu cầu đánh giá và bình luận



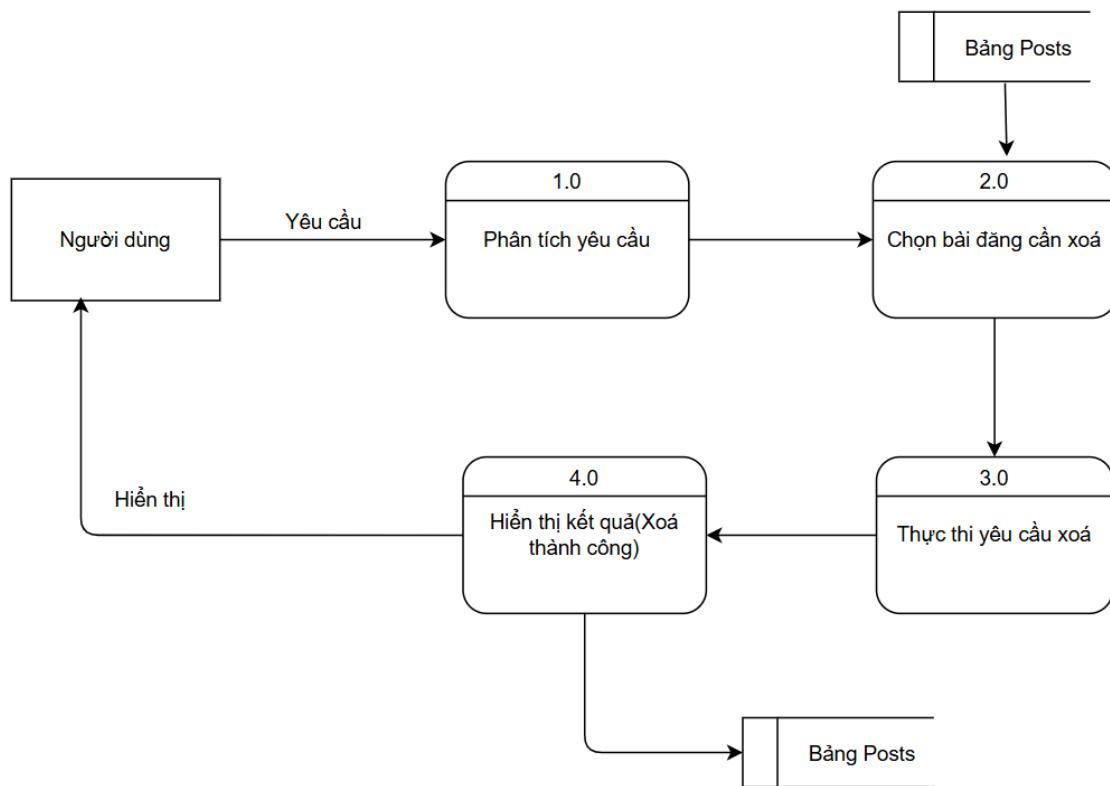
#### 1.2.6 DFD Thêm bài đăng



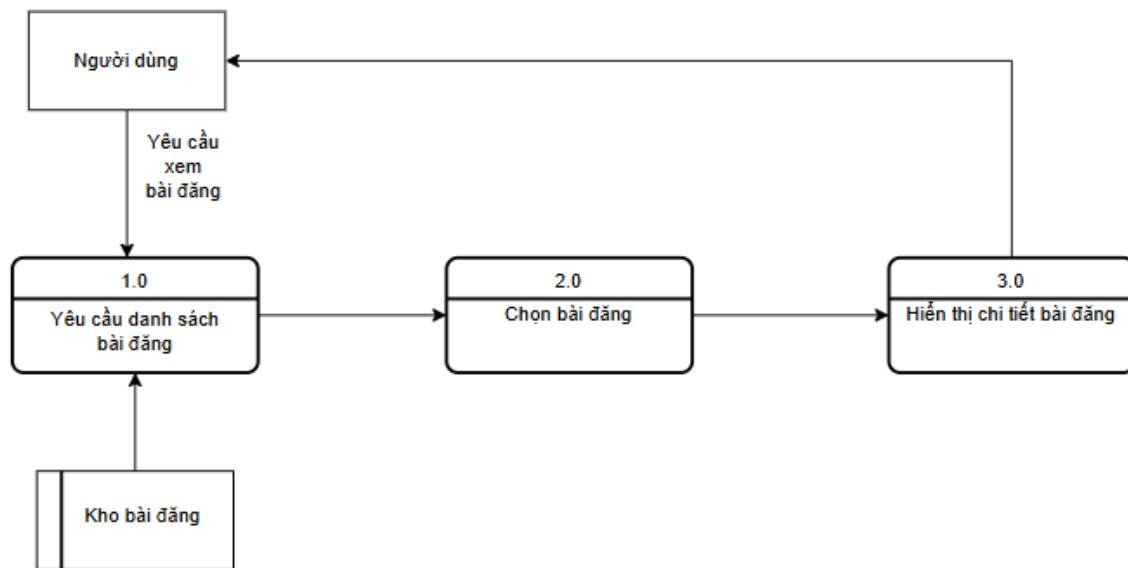
### 1.2.7 DFD Sửa bài đăng



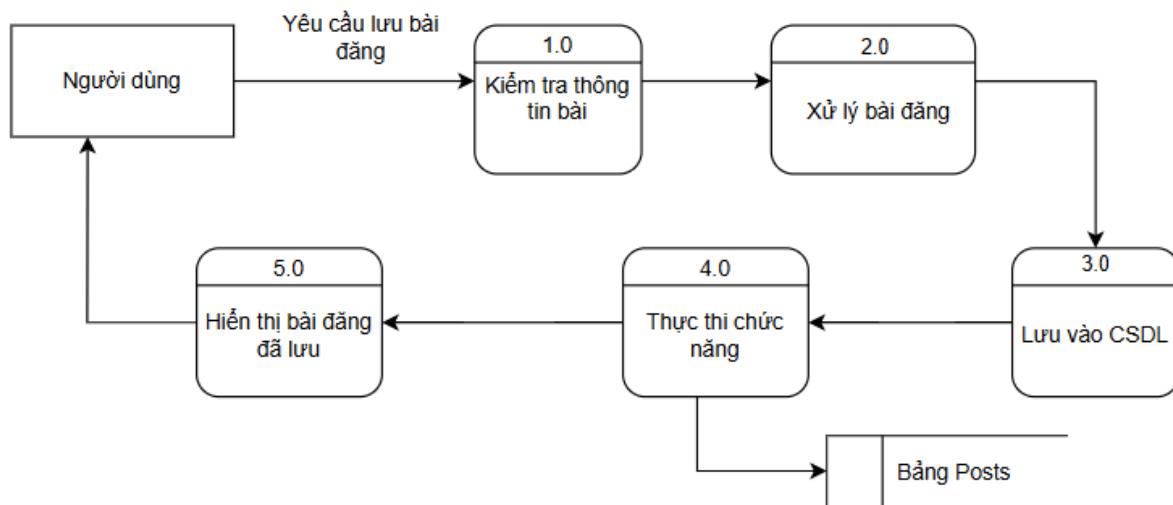
### 1.2.8 DFD Xóa bài đăng



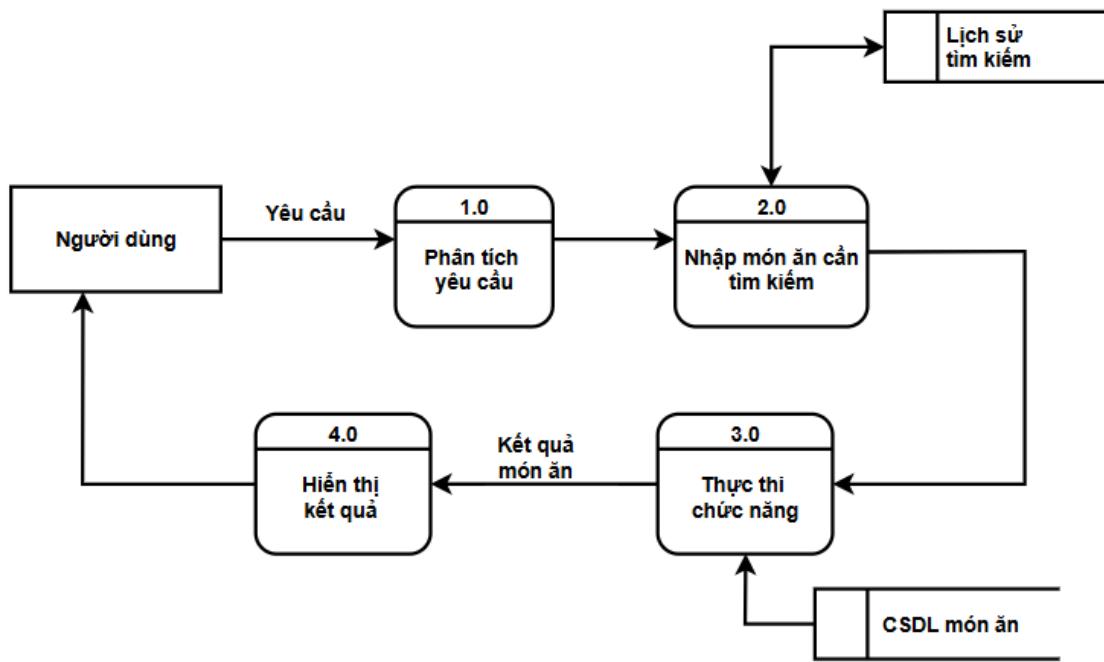
### 1.2.9 DFD Xem bài đăng



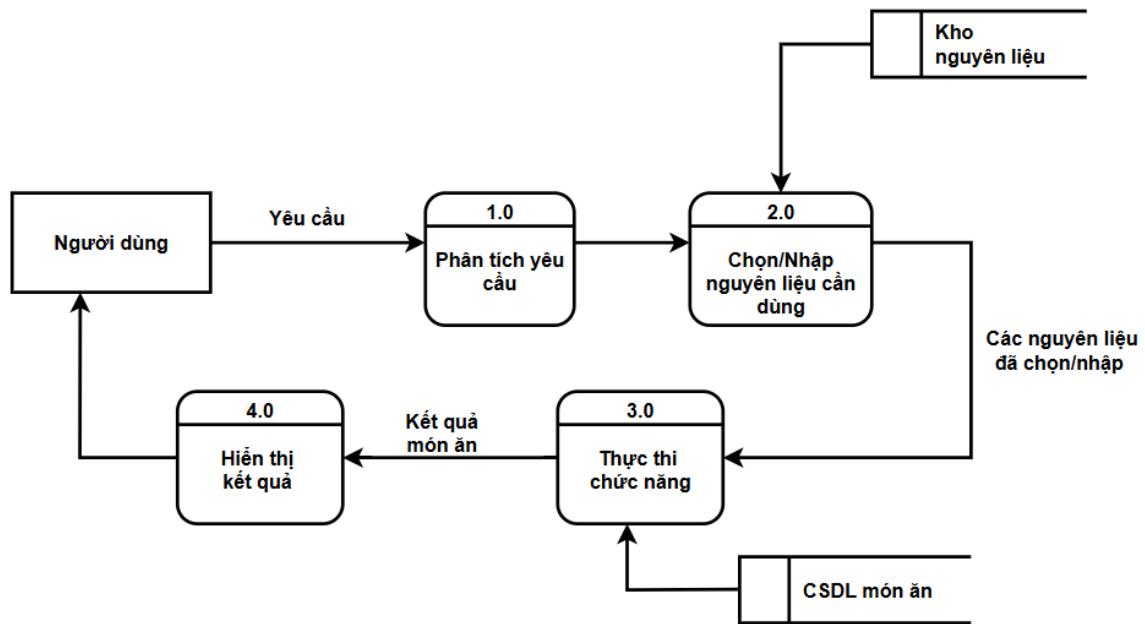
### 1.2.10 DFD Lưu bài đăng



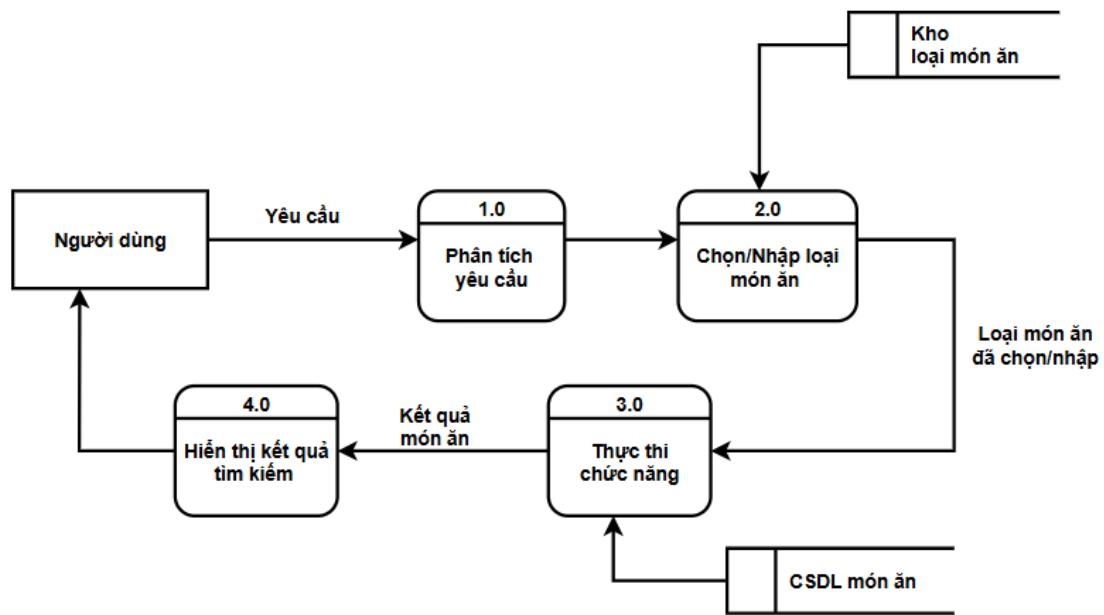
### 1.2.11 DFD Tìm kiếm món ăn



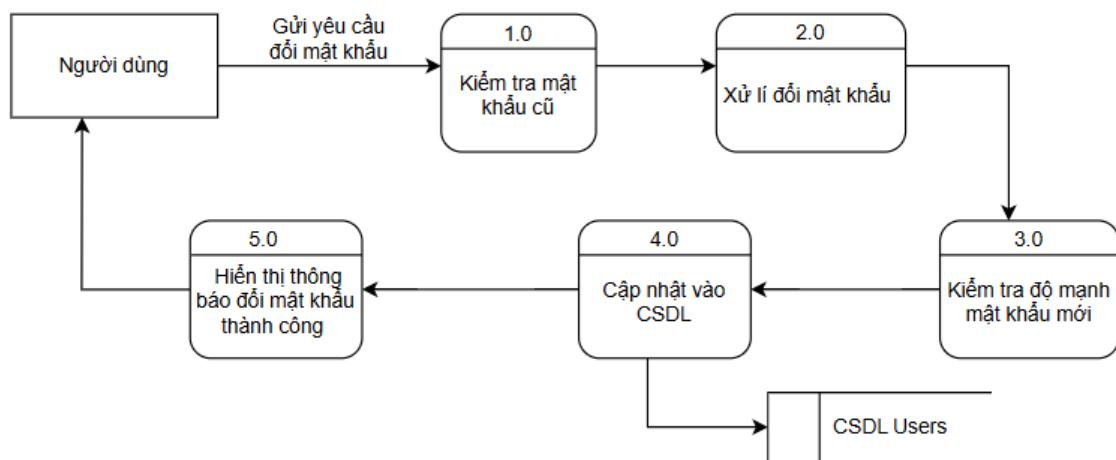
### 1.2.12 DFD Gợi ý theo nguyên liệu



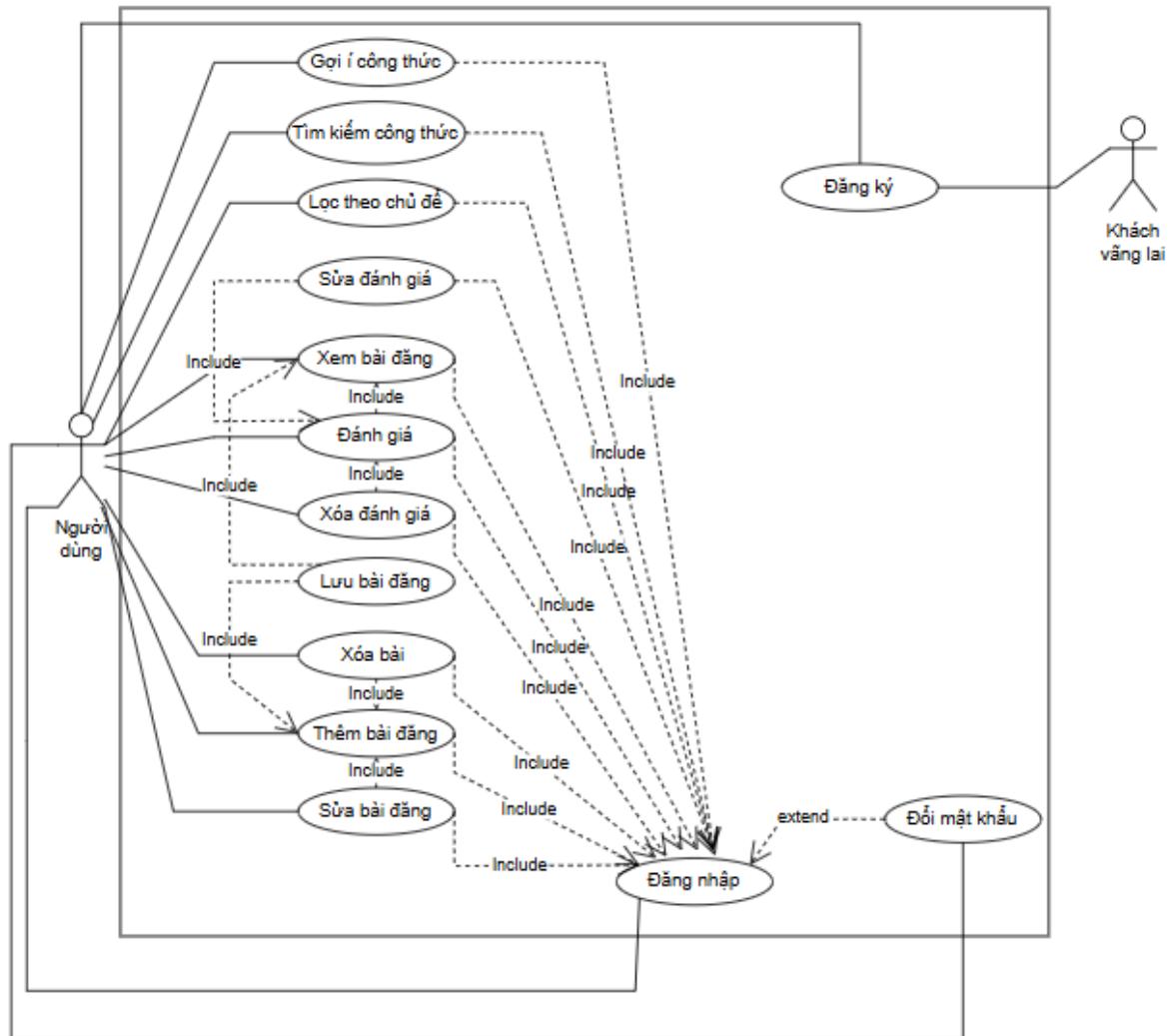
### 1.2.13 DFD Gợi ý theo chủ đề



#### 1.2.14 DFD Đổi mật khẩu



### 1.3 Sơ đồ Use case tổng quát



### 1.4 Miêu tả chi tiết các Use cases

#### 1.4.1 Use case Đăng ký

Use Case ID	01		
Use Case Name	Đăng ký		
Created By:	Nguyễn Tân Tài	Last Updated By	Nguyễn Tân Tài
Date Created:	6/12/2025	Date Last Updated	14/12/2025

Actor	Khách vãng lai
-------	----------------

<i>Description</i>	Chức năng đăng ký cho phép người dùng đăng ký để có tài khoản và đăng nhập được trên nền tảng Nostimo
<i>Trigger</i>	Chức năng này được sử dụng khi khách vãng lai click vào nút “Đăng ký”
<i>Pre-conditions</i>	Khách vãng lai truy cập vào Ứng dụng
<i>Post conditions</i>	Khách vãng lai đăng ký thành công
<i>Flow of Events</i>	<ol style="list-style-type: none"> <li>1. Khách vãng lai truy đăng nhập vào Ứng dụng</li> <li>2. Khách vãng lai chọn “Đăng ký”</li> <li>3. Hệ thống hiển thị giao diện nhập tên, email, mật khẩu</li> <li>4. Khách vãng lai nhập tên, email, mật khẩu và chọn “Đăng ký”</li> <li>5. Hệ thống hiển thị thông báo đăng ký thành công</li> <li>6. Chuyển sang màn hình đăng nhập</li> </ol>

#### 1.4.2 Use case Đăng nhập

Use Case ID	02		
Use Case Name	Đăng nhập		
Created By:	Nguyễn Tân Tài	Last Updated By	Nguyễn Tân Tài
Date Created:	6/12/2025	Date Last Updated	14/12/2025

Actor	Người dùng
<i>Description</i>	Chức năng này cho phép người dùng có thể truy cập vào ứng dụng để sử dụng các chức năng như đăng bài, bình luận và đánh giá, chia sẻ công thức,...
<i>Trigger</i>	Chức năng này được sử dụng khi người dùng vào ứng dụng lần đầu tiên hoặc là sau khi đăng xuất
<i>Pre-conditions</i>	Người dùng truy cập vào Ứng dụng
<i>Post conditions</i>	Người dùng đăng nhập thành công
<i>Flow of Events</i>	<ol style="list-style-type: none"> <li>1. Người dùng truy cập vào Ứng dụng</li> <li>2. Hệ thống hiển thị giao diện nhập email, mật khẩu</li> <li>3. Người dùng nhập email, mật khẩu và chọn “Đăng nhập”</li> </ol>

	4. Hệ thống hiển thị thông báo đăng nhập thành công và có thể sử dụng các tính năng của ứng dụng.
--	---

#### 1.4.3 Use case Xóa bài đăng

Use Case ID	03		
Use Case Name	Xóa bài đăng		
Created By:	Nguyễn Duy Mạnh	Last Updated By	Nguyễn Duy Mạnh
Date Created:	04/12/2025	Date Last Updated	14/12/2025

Actor	Người dùng.
Description	Chức năng này cho phép người dùng có thể tùy ý xóa bài đăng của mình khỏi trang cá nhân.
Trigger	Chức năng này được sử dụng khi người dùng bấm vào nút “Xóa” trên bài đăng.
Pre-conditions	Người dùng đã đăng nhập được vào ứng dụng, trên trang cá nhân đã có bài đăng và bài đăng đó thuộc về chính người dùng đó.
Post conditions	Bài đăng sẽ được xóa khỏi trang cá nhân. Mọi người dùng khác không thể xem và đánh giá được nữa
Flow of Events	<ol style="list-style-type: none"> <li>1. Người dùng truy cập vào app và đăng nhập vào tài khoản của mình.</li> <li>2. Người dùng mở trang cá nhân hoặc danh sách bài đăng của mình.</li> <li>3. Người dùng chọn bài đăng muốn xoá.</li> <li>4. Nhấn nút “Xóa” trên bài đăng.</li> <li>5. Hệ thống yêu cầu xác nhận xoá.</li> </ol>

	<p>6. Người dùng xác nhận việc xoá.</p> <p>7. Hệ thống xóa bài đăng và thông báo “Xóa thành công”.</p>
--	--

#### 1.4.4 Use case Đổi mật khẩu

Use Case ID	04		
Use Case Name	Đổi mật khẩu		
Created By:	Nguyễn Duy Mạnh	Last Updated By	Nguyễn Duy Mạnh
Date Created:	04/12/2025	Date Last Updated	14/12/2025

Actor	Người dùng.
Description	Chức năng này cho phép người dùng tự thay đổi mật khẩu của mình trong ứng dụng. Khi mật khẩu được cập nhật thành công người dùng sẽ phải đăng nhập bằng mật khẩu mới trong các lần tiếp theo.
Trigger	Người dùng chọn chức năng “Đổi mật khẩu” trong phần cài đặt tài khoản.
Pre-conditions	Người dùng đã có tài khoản và đã đăng nhập vào ứng dụng.
Post conditions	Mật khẩu mới của người dùng được cập nhật thành công. Người dùng cần phải sử dụng mật khẩu mới khi đăng nhập ở những lần tiếp theo.
Flow of Events	1. Người dùng truy cập vào app và đăng nhập vào tài khoản của mình.

	<p>2. Người dùng truy cập vào cài đặt tài khoản.</p> <p>3. Người dùng chọn “Đổi mật khẩu”.</p> <p>4. Hệ thống hiển thị màn hình đổi mật khẩu, yêu cầu nhập mật khẩu hiện tại, mật khẩu mới và xác nhận mật khẩu mới.</p> <p>5. Hệ thống sẽ kiểm tra xem người dùng đã nhập đúng mật khẩu hiện tại hay chưa? Mật khẩu mới có hợp lệ không (độ dài, ký tự,...)? Mật khẩu xác nhận đã chính xác hay chưa?</p> <p>6. Nếu chính xác người dùng nhấn “Lưu” để cập nhật thay đổi mật khẩu.</p> <p>7. Hệ thống thông báo đổi mật khẩu thành công.</p>
Alternative Flows	<ul style="list-style-type: none"> <li>Mật khẩu hiện tại không đúng:</li> </ul> <p>1.Tại bước 5 khi người dùng nhập mật khẩu hiện tại sai, hệ thống sẽ phát hiện và thông báo “Mật khẩu hiện tại không đúng”.</p> <p>2. Hệ thống sẽ yêu cầu người dùng nhập lại mật khẩu.</p> <ul style="list-style-type: none"> <li>Mật khẩu mới không hợp lệ:</li> </ul> <p>1.Tại bước 5 khi người dùng nhập không đúng định dạng, hệ thống sẽ phát hiện và thông báo “Vd: Mật khẩu phải có ít nhất 8 ký tự, bao gồm chữ hoa, chữ thường, số và ký tự đặc biệt ...”</p> <p>2.Hệ thống sẽ yêu cầu người dùng nhập lại mật khẩu mới.</p> <ul style="list-style-type: none"> <li>Mật khẩu mới và mật khẩu xác nhận không trùng khớp:</li> </ul> <p>1.Tại bước 5 khi người dùng nhập mật khẩu xác nhận không chính xác, hệ thống sẽ thông báo ”Xác nhận mật khẩu không trùng khớp”.</p> <p>2.Hệ thống sẽ yêu cầu người dùng nhập lại mật khẩu xác nhận.</p>

#### 1.4.5 Use case Thêm bài đăng

Use Case ID	05		
Use Case Name	Thêm bài đăng		
Created By:	Đặng Đình Phước	Last Updated By	Đặng Đình Phước
Date Created:	06/12/2025	Date Last Updated	14/12/2025

<i>Actor</i>	Người dùng
<i>Description</i>	Chức năng đăng bài cho phép dùng có thể đăng bài chia sẻ công thức nấu ăn kèm hình ảnh hoặc video
<i>Trigger</i>	Chức năng này được sử dụng khi người dùng click vào nút “thêm bài đăng”
<i>Pre-conditions</i>	Sau khi người dùng đăng nhập vào app
<i>Post conditions</i>	Người dùng đăng bài thành công
<i>Flow of Events</i>	<ol style="list-style-type: none"> <li>1. Người dùng truy cập vào app</li> <li>2. Người dùng đăng nhập và đăng nhập thành công</li> <li>3. Người dùng bấm vào nút thêm bài đăng</li> <li>3. Hệ thống hiển thị giao diện cho phép soạn thảo văn bản hoặc thêm ảnh và video</li> <li>4. Người dùng nhập văn bản theo ý muốn hoặc add thêm hình ảnh</li> <li>5. Người dùng click vào xác nhận đăng bài</li> <li>7. Hệ thống hiển thị giao diện xác nhận đăng bài thành công</li> </ol>

#### 1.4.6 Use case Sửa bài đăng

<i>Use Case ID</i>	06		
<i>Use Case Name</i>	Sửa bài đăng		
<i>Created By:</i>	Đặng Phương Nam	<i>Last Updated By</i>	Đặng Phương Nam
<i>Date Created:</i>	04/12/2025	<i>Date Last Updated</i>	5/12/2025

<i>Actor</i>	Người dùng
<i>Description</i>	Chức năng sửa bài đăng cho phép người dùng có thể chỉnh sửa nội dung những bài viết đã đăng theo ý muốn
<i>Trigger</i>	Chức năng này được sử dụng khi người dùng click vào nút “Chỉnh sửa”
<i>Pre-conditions</i>	Người dùng muốn sửa nội dung thông tin bài đăng theo ý muốn
<i>Post conditions</i>	Người dùng chỉnh sửa bài đăng thành công. Những người dùng khác sẽ xem được những bài đăng đã được chỉnh sửa
<i>Flow of Events</i>	<ol style="list-style-type: none"> <li>1. Người dùng chọn vào mục “Trang cá nhân”</li> <li>2. Người dùng lựa chọn bài đăng đã đăng muốn chỉnh sửa và click vào nút “Chỉnh sửa”</li> <li>3. Hệ thống hiển thị giao diện của bài đăng, người dùng có thể trỏ để thông tin bài đăng để chỉnh sửa</li> <li>4. Sau khi chỉnh sửa, người dùng click vào nút “Lưu”</li> </ol>

	5. Hệ thống thông báo người dùng đã sửa bài đăng thành công
--	---

#### 1.4.7 Use case Xem bài đăng

Use Case ID	07		
Use Case Name	Xem bài đăng		
Created By:	Đặng Phương Nam	Last Updated By	Đặng Phương Nam
Date Created:	04/12/2025	Date Last Updated	5/12/2025

Actor	Người dùng
Description	Chức năng xem bài đăng cho phép người dùng có thể xem bài đăng của mình hoặc của những người dùng khác trong app
Trigger	Chức năng này được sử dụng khi người dùng xem thông tin bài viết của những người khác
Pre-conditions	Người dùng muốn xem các bài đăng của người khác
Post conditions	Người dùng nắm bắt được thông tin bài đăng sau khi xem
Flow of Events	<ol style="list-style-type: none"> <li>1. Click vào bài đăng mà người dùng muốn xem</li> <li>2. Màn hình hiển thị thông tin của bài đăng đã chọn</li> </ol>

#### 1.4.8 Use case Sửa yêu cầu đánh giá, bình luận

Use Case ID	08		
Use Case Name	Sửa yêu cầu đánh giá/ bình luận		
Created By:	Vũ Thị Bảo Nhi	Last Updated By	Vũ Thị Bảo Nhi
Date Created:	08/12/2025	Date Last Updated	08/12/2025

Actor	Người dùng
Description	Chức năng sửa yêu cầu cho phép người dùng có thể chỉnh sửa lại nội dung đánh giá và bình luận đã đăng theo ý muốn.
Trigger	Chức năng này được sử dụng khi người dùng click vào nút “Chỉnh sửa đánh giá”
Pre-conditions	<ul style="list-style-type: none"> <li>-Người dùng đã đăng nhập vào hệ thống.</li> <li>-Đánh giá/ Bình luận cần sửa đã tồn tại trên hệ thống.</li> </ul>

	-Đánh giá/ Bình luận thuộc về người dùng đó và vẫn còn cho phép chỉnh sửa.
<i>Post conditions</i>	-Nội dung đánh giá được cập nhật lại trong Data store: Chi tiết đánh giá. -Người dùng nhìn thấy đánh giá, bình luận đã được sửa thành công.
<i>Flow of Events</i>	6. Người dùng gửi yêu cầu chỉnh sửa đánh giá/ bình luận. 7. Hệ thống tiếp nhận và phân tích yêu cầu chỉnh sửa từ người dùng. 8. Hệ thống truy xuất dữ liệu cũ từ Kho “Chi tiết đánh giá” để lấy đánh giá, bình luận cần sửa và hiển thị form sửa ( có sẵn nội dung cũ) cho người dùng. 9. Sau khi lấy đánh giá, bình luận cần sửa thì người dùng chỉnh sửa lại đánh giá, bình luận trên giao diện. 10. Gửi yêu cầu lưu: Người dùng gửi yêu cầu lưu thông tin thay đổi mới( bằng cách nhấn nút Lưu/Cập nhật). 11. Hệ thống lưu thông tin thay đổi mới vào kho “Chi tiết đánh giá”. 12. Hệ thống hiển thị lại đánh giá , bình luận đã sửa cho người dùng xem.

#### 1.4.9 Use case Tìm kiếm món ăn

<i>Use Case ID</i>	09		
<i>Use Case Name</i>	Tìm kiếm món ăn		
<i>Created By:</i>	Đỗ Yên Nhi	Last Updated By	Đỗ Yên Nhi
<i>Date Created:</i>	04/12/2025	Date Last Updated	15/12/2025

<i>Actor</i>	Người dùng.
<i>Description</i>	Chức năng này cho phép người dùng tìm kiếm công thức món ăn.
<i>Trigger</i>	Chức năng này được sử dụng khi người dùng nhấn chọn thanh tìm kiếm sau đó tìm kiếm công thức món ăn mình mong muốn.
<i>Pre-conditions</i>	Người dùng đã đăng nhập được vào ứng dụng. Nếu là khách vãng lai phải đăng ký để được cấp tài khoản sau đó giao diện ứng dụng hiển thị.
<i>Post conditions</i>	Danh sách công thức món ăn phù hợp sẽ hiển thị. Người dùng chọn công thức mong muốn.

<i>Flow of Events</i>	<ol style="list-style-type: none"> <li>1. Người dùng chọn thanh <b>Tìm kiếm</b>.</li> <li>2. Người dùng nhập công thức cần tìm (ví dụ: "<b>Sườn xào chua ngọt</b>", "<b>Thịt nướng</b>",...).</li> <li>3. Người dùng nhấn nút <b>Tìm kiếm</b>.</li> <li>4. Hệ thống kiểm tra dữ liệu, phân tích và xử lý từ khóa.</li> <li>5. Hệ thống hiển thị danh sách công thức món ăn phù hợp.</li> <li>6. Người dùng chọn một công thức trong kết quả.</li> </ol>
<i>Alternative Flows</i>	<p>3.1 Người dùng tìm tên món ăn nhưng không có trong <b>CSDL món ăn</b>.</p> <p>3.1.1 Hệ thống hiển thị <b>thông báo kết quả rỗng</b> ("Xin lỗi, không tìm thấy món ăn nào khớp với từ khóa của bạn."). </p>

#### 1.4.10 Use case Gọi ý món ăn theo nguyên liệu

<i>Use Case ID</i>	10		
<i>Use Case Name</i>	Tìm kiếm món ăn theo nguyên liệu		
<i>Created By:</i>	Nguyễn Hoàng Anh	Last Updated By	Nguyễn Hoàng Anh
<i>Date Created:</i>	05/12/2025	Date Last Updated	15/12/2025

<i>Actor</i>	Người dùng.
<i>Description</i>	Chức năng này cho phép người dùng tìm kiếm món ăn theo các nguyên liệu được chọn.
<i>Trigger</i>	Chức năng này được sử dụng khi người dùng nhấn chọn vào mục " <b>Tìm kiếm món ăn theo nguyên liệu</b> " trong giao diện chính sau đó chọn các nguyên liệu mà mình cần.

<i>Pre-conditions</i>	Người dùng phải có tài khoản và đã đăng nhập được vào ứng dụng.
<i>Post conditions</i>	Danh sách các món ăn có đủ các nguyên liệu mà người dùng chọn và gợi ý các món chỉ cần thêm một vài nguyên liệu.
<i>Flow of Events</i>	<ol style="list-style-type: none"> <li>1. Người dùng chọn mục “<b>Tìm kiếm món ăn theo nguyên liệu</b>”.</li> <li>2. Hệ thống hiển thị giao diện tìm kiếm kèm danh sách các nguyên liệu.</li> <li>3. Người dùng chọn/xóa các nguyên liệu mong muốn và nhấn nút “<b>Tìm kiếm</b>”.</li> <li>4. Hệ thống xử lý dữ liệu đầu vào, thực hiện chuẩn hóa và truy vấn cơ sở dữ liệu để tìm kiếm các món ăn phù hợp.</li> <li>5. Hệ thống hiển thị danh sách các món ăn phù hợp với nguyên liệu đã chọn.</li> <li>6. Người dùng chọn một món ăn từ danh sách kết quả để xem thông tin chi tiết.</li> </ol>
<i>Alternative Flows</i>	<p>3.1 Người dùng không nhập nguyên liệu.</p> <p>3.1.1 Hệ thống hiển thị thông báo lỗi ("Vui lòng chọn ít nhất một nguyên liệu.") trên giao diện tìm kiếm.</p>

#### 1.4.11 Use case Gợi ý món ăn theo chủ đề

<i>Use Case ID</i>	11		
<i>Use Case Name</i>	Gợi ý món ăn theo chủ đề		
<i>Created By:</i>	Nguyễn Hoàng Anh	Last Updated By	Nguyễn Hoàng Anh
<i>Date Created:</i>	06/12/2025	Date Last Updated	15/12/2025

<i>Actor</i>	Người dùng.
<i>Description</i>	Chức năng này cho phép người dùng tìm kiếm món ăn theo chủ đề (món nước, món chay, ...).
<i>Trigger</i>	Chức năng này được sử dụng khi người dùng nhấn chọn vào mục “ <b>Các chủ đề món ăn</b> ” trong giao diện chính.
<i>Pre-conditions</i>	Người dùng phải có tài khoản và đã đăng nhập được vào ứng dụng.
<i>Post conditions</i>	Danh sách các món ăn đã được hệ thống sắp xếp theo chủ mà người dùng đã chọn.
<i>Flow of Events</i>	<ol style="list-style-type: none"> <li>Người dùng chọn mục “<b>Các chủ đề món ăn</b>”.</li> <li>Người dùng chọn loại món ăn (món nước, món chay, ...) mình mong muốn.</li> <li>Hệ thống tiếp nhận lựa chọn và truy vấn cơ sở dữ liệu để tìm các món ăn thuộc phân loại đó.</li> <li>Hệ thống hiển thị danh sách món ăn tương ứng với loại người dùng vừa chọn.</li> <li>Người dùng chọn một món ăn trong kết quả để xem chi tiết.</li> </ol>

#### 1.4.12 Use case Lưu bài đăng

<i>Use Case ID</i>	12		
<i>Use Case Name</i>	Lưu bài viết		
<i>Created By:</i>	Cao Khánh Toàn	Last Updated By	Cao Khánh Toàn
<i>Date Created:</i>	12/4/2025	Date Last Updated	12/4/2025

<i>Actor</i>	Người dùng đã đăng nhập.	
<i>Description</i>	Chức năng lưu bài viết cho phép người dùng lưu bài đăng yêu thích nào đó vào hệ thống.	
<i>Trigger</i>	Chức năng bắt đầu khi người dùng nhấn nút “Lưu bài viết”.	

<i>Pre-conditions</i>	Người dùng đã đăng nhập vào ứng dụng. Người dùng đang xem một bài viết bất kì nào đó.
<i>Post conditions</i>	Bài viết được lưu thành công vào hệ thống. Bài viết xuất hiện trong danh sách bài viết của người dùng.
<i>Flow of Events</i>	<ol style="list-style-type: none"> <li>1. Người dùng truy cập vào ứng dụng và xem một bài viết bất kì.</li> <li>2. Người dùng nhấn nút “<b>Lưu bài viết</b>”.</li> <li>3. Hệ thống kiểm tra dữ liệu bài viết (nội dung, đúng định dạng hình ảnh,...).</li> <li>4. Hệ thống xử lý và lưu bài viết vào bảng Posts.</li> <li>5. Hệ thống thông báo “Lưu bài viết thành công”.</li> <li>6. Hệ thống hiển thị bài viết vừa lưu trong danh sách bài viết của người dùng.</li> </ol>

#### 1.4.13 Use case Đánh giá và bình luận

User case ID:	13		
Use Case Name:	Đánh giá và bình luận		
Created By:	Vi Thị Phương Nhung	Last Updated By:	Vi Thị Phương Nhung
Date Created:	5/12/2025	Date Last Updated:	5/12/2025

<i>Actor</i>	Người dùng.
<i>Description</i>	Chức năng đánh giá và bình luận cho phép người dùng đánh giá và bình luận dưới mỗi bài viết.
<i>Trigger</i>	Chức năng này được sử dụng khi người dùng click vào nút “Đánh giá”.
<i>Pre-conditions</i>	Người dùng có tài khoản, đã đăng nhập
<i>Post conditions</i>	Người dùng đánh giá và bình luận thành công

<i>Flow of Events</i>	<ol style="list-style-type: none"> <li>1. Dưới mỗi bài viết, người dùng click vào “Đánh giá”</li> <li>2. Hệ thống hiển thị giao diện để đánh giá số sao từ 1 đến 5 sao, bên dưới kèm bình luận.</li> <li>2. Người dùng nhập sao đánh giá và nhập bình luận.</li> <li>3. Người dùng click “Gửi”</li> <li>4. Hệ thống hiển thị chi tiết đánh giá, bình luận phía dưới bài viết.</li> </ol>
-----------------------	--

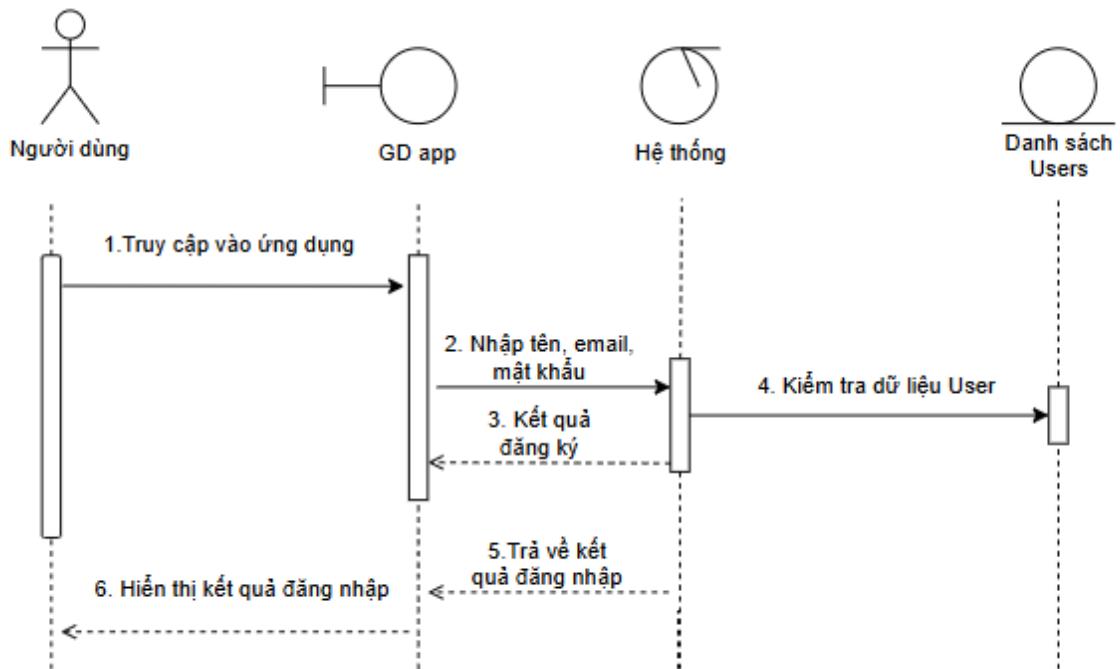
#### 1.4.14 Xóa đánh giá và bình luận

User case ID:	14		
Use Case Name:	Xóa đánh giá và bình luận		
Created By:	Võ Thị Phương Nhung	Last Updated By:	Võ Thị Phương Nhung
Date Created:	5/12/2025	Date Last Updated:	5/12/2025

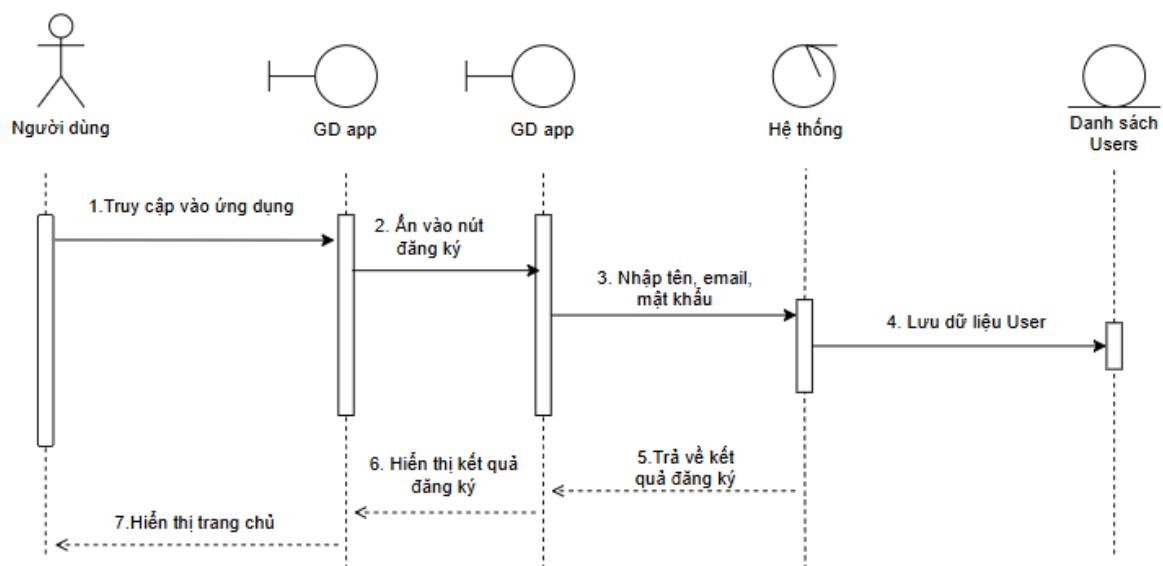
<i>Actor</i>	Người dùng.
<i>Description</i>	Chức năng xóa đánh giá, bình luận cho phép người dùng xóa những đánh giá và bình luận mà mình đã đánh giá, bình luận trước đó.
<i>Trigger</i>	Người dùng truy cập vào phần đánh giá, bình luận mà họ đã viết, lựa chọn “Xóa”.
<i>Pre-conditions</i>	Người dùng có tài khoản, đã đăng nhập. Đánh giá, bình luận phải thuộc về người dùng hiện tại.
<i>Post conditions</i>	Đánh giá được xóa khỏi hệ thống.
<i>Flow of Events</i>	<ol style="list-style-type: none"> <li>1. Người dùng mở bài viết có chứa đánh giá của mình.</li> <li>2. Người dùng nhấn nút “Xóa”.</li> <li>3. Hệ thống hiển thị “Xác nhận xóa”.</li> <li>4. Người dùng chọn “Xác nhận xóa”. Hệ thống sẽ xóa đánh giá và bình luận</li> <li>5. Thông báo xóa thành công, cập nhật lại giao diện.</li> </ol>

## 1.5 Sơ đồ Trình tự (SD)

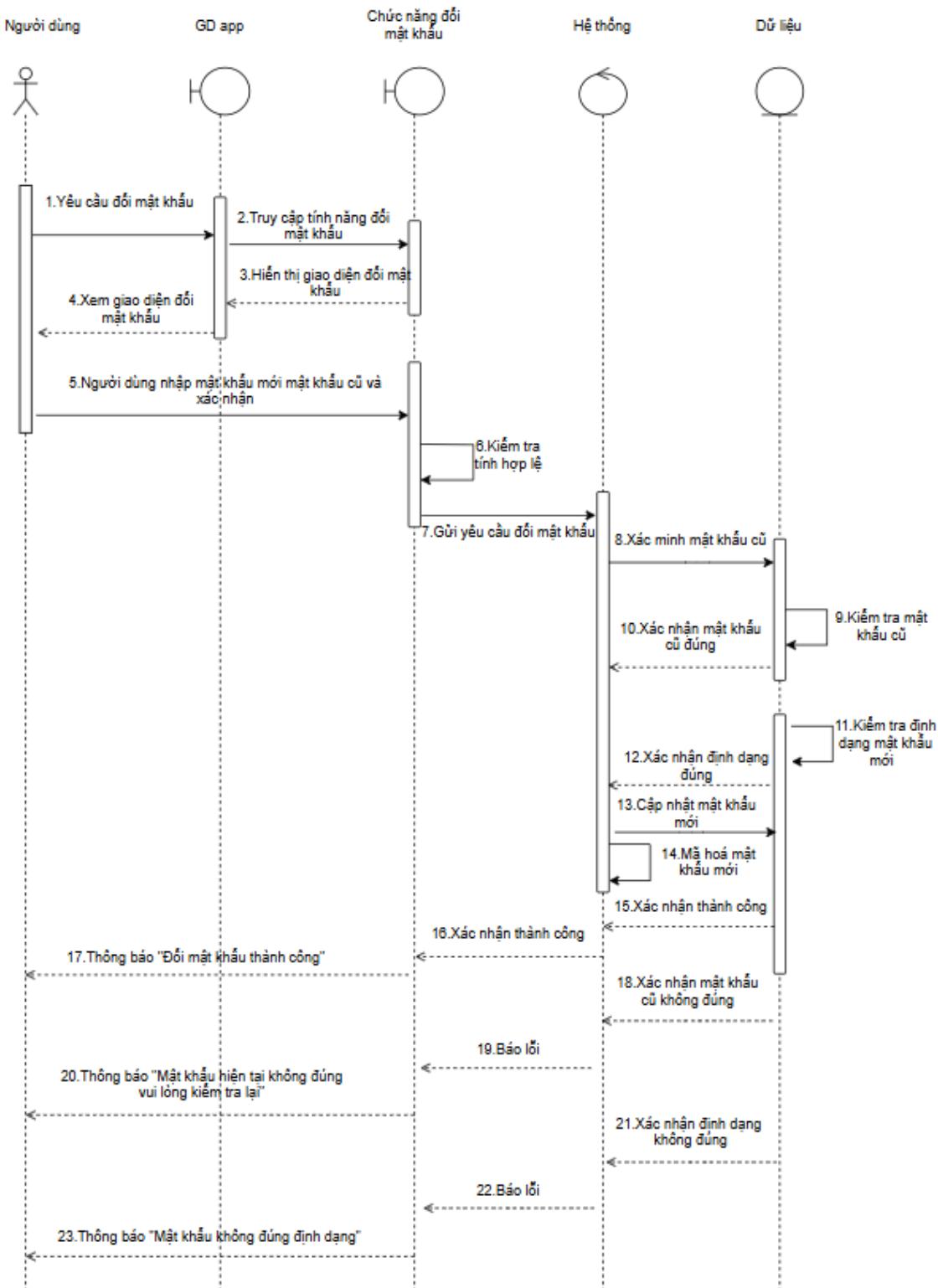
### 1.5.1 SD Đăng Nhập



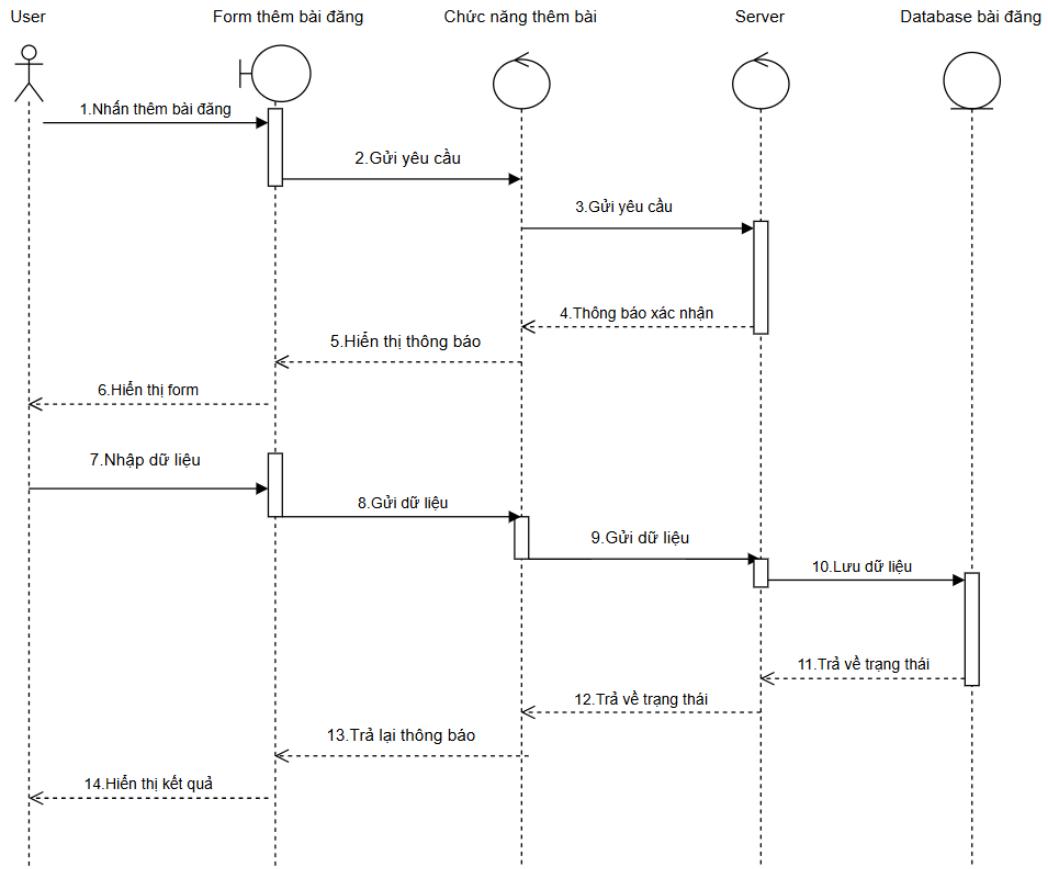
### 1.5.2 SD Đăng ký



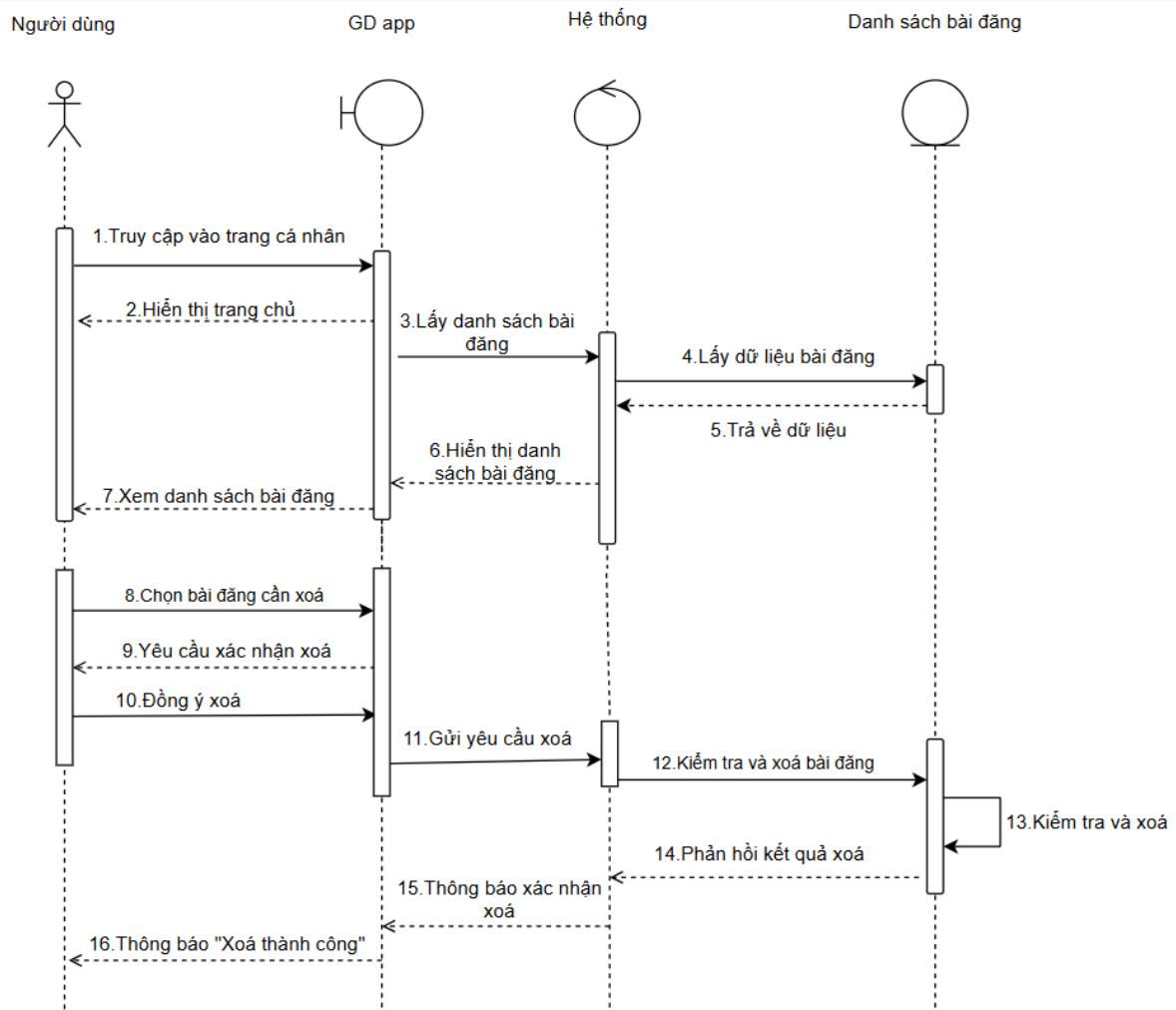
### 1.5.3 SD Đổi mật khẩu



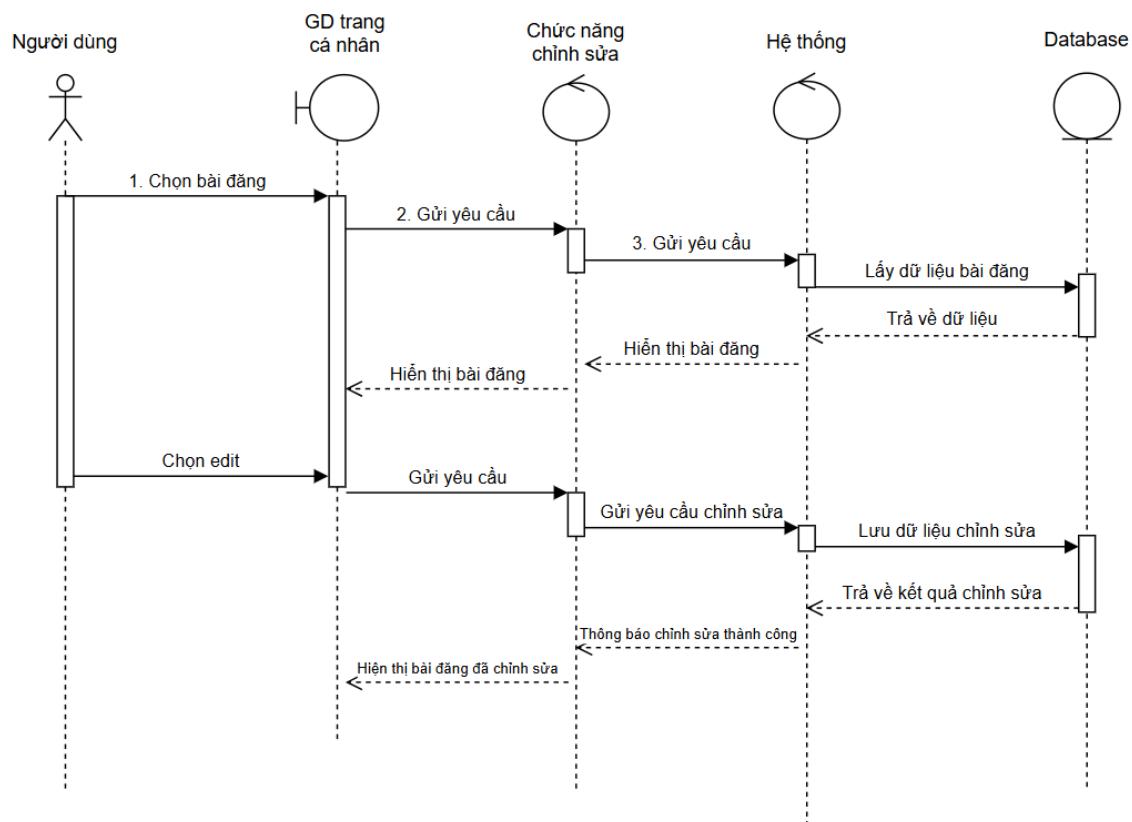
### 1.5.4 SD Thêm bài đăng



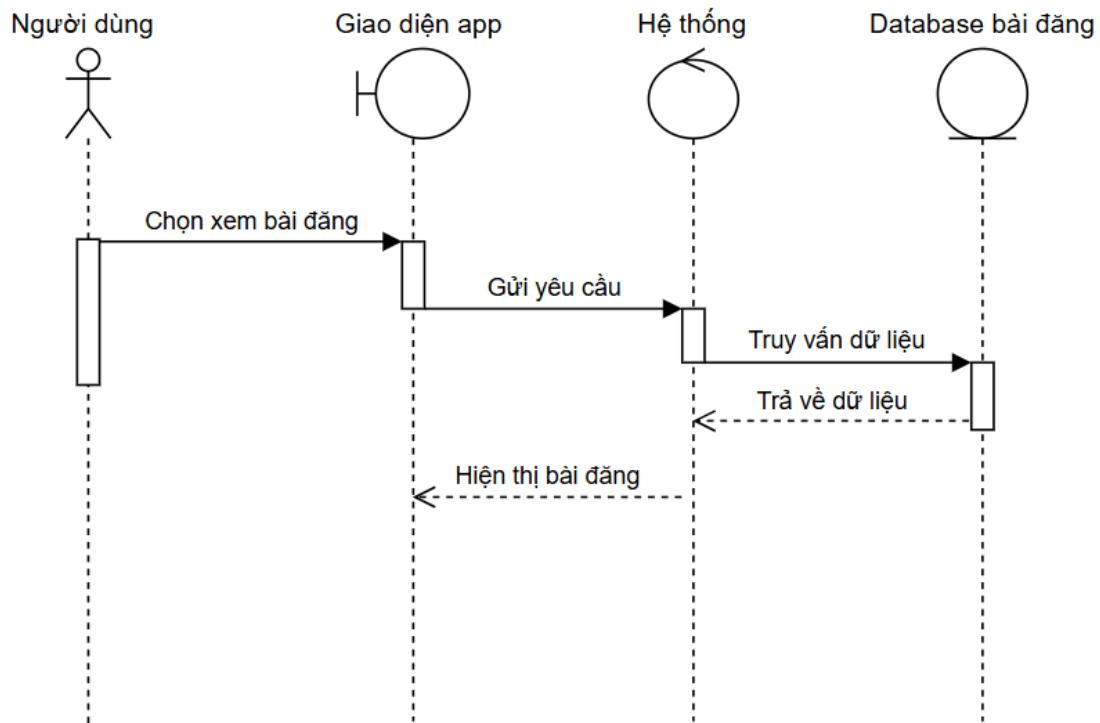
### 1.5.5 SD Xóa bài đăng



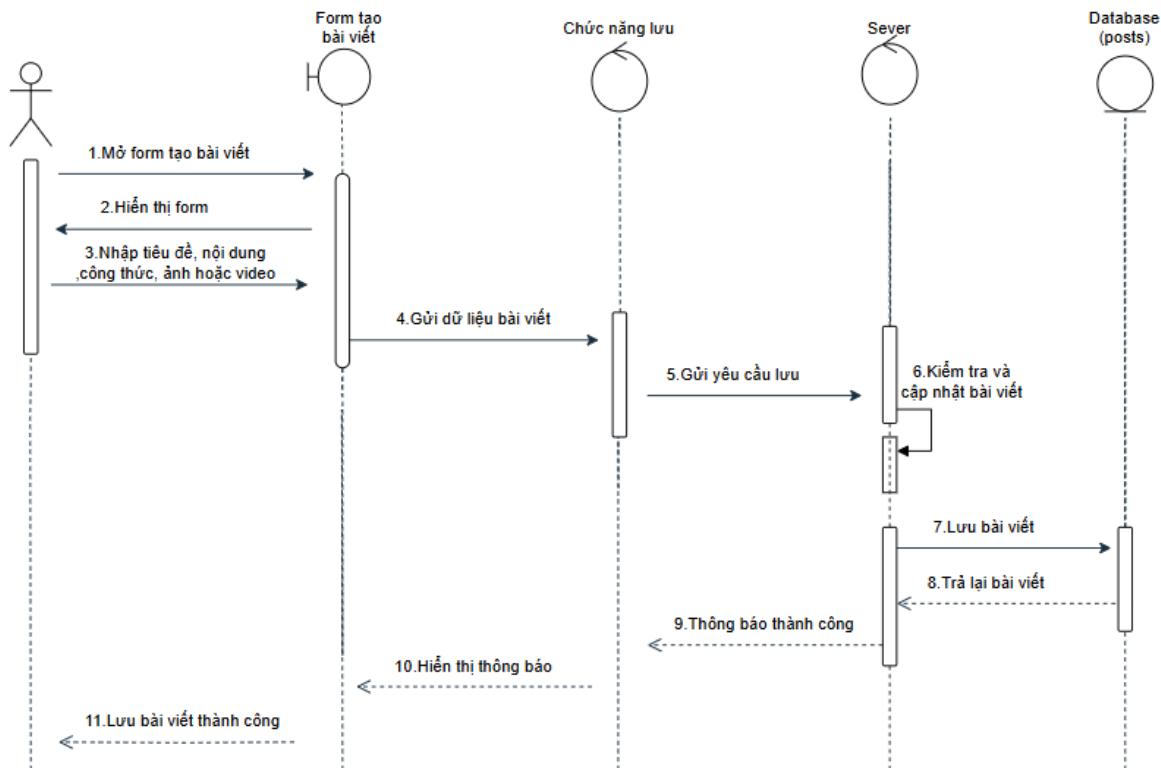
### 1.5.6 SD Sửa bài đăng



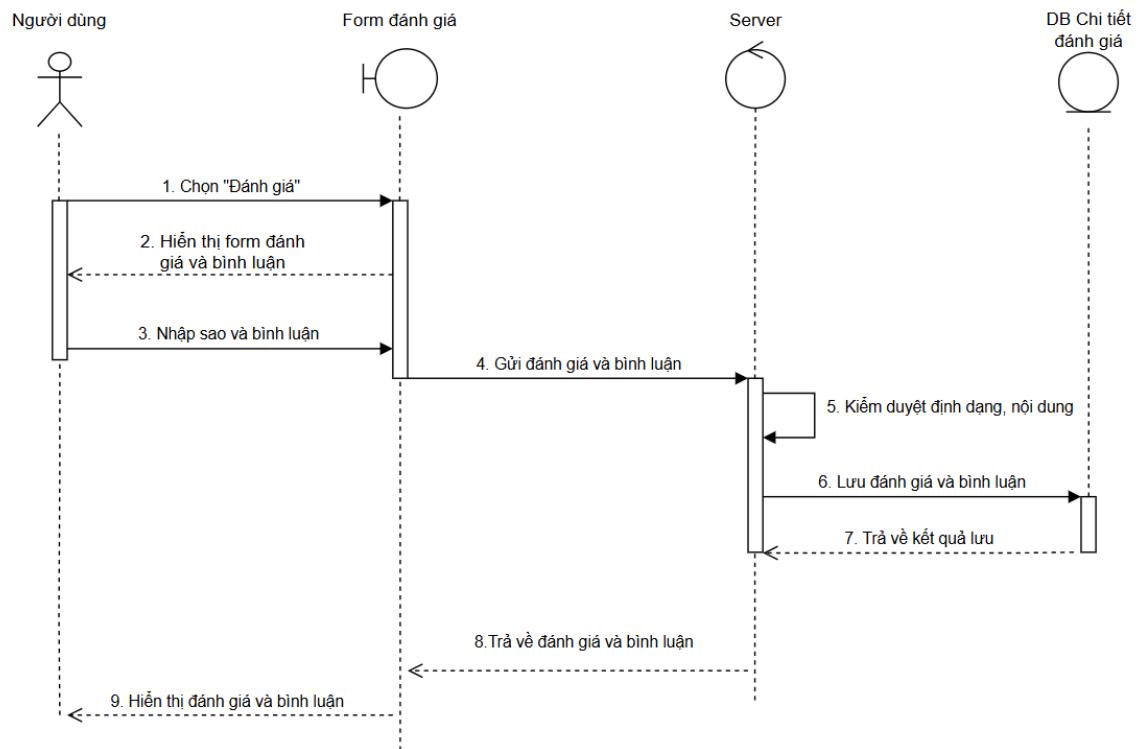
### 1.5.7 SD Xem bài đăng



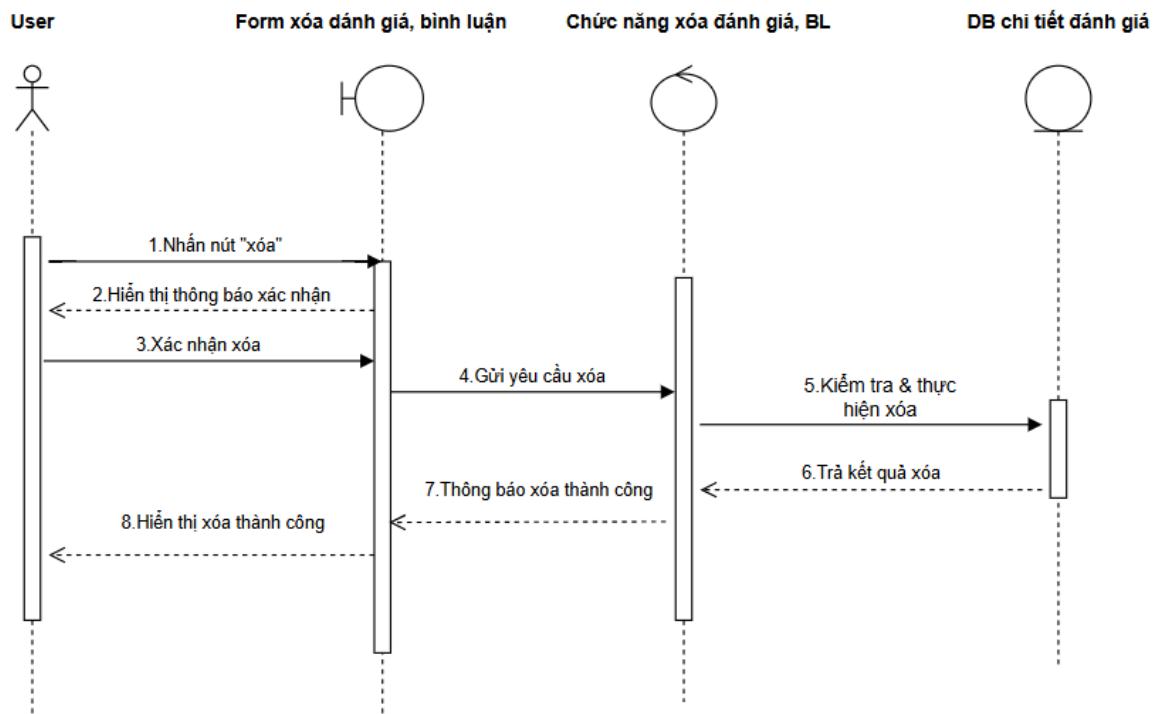
### 1.5.8 SD Lưu bài đăng



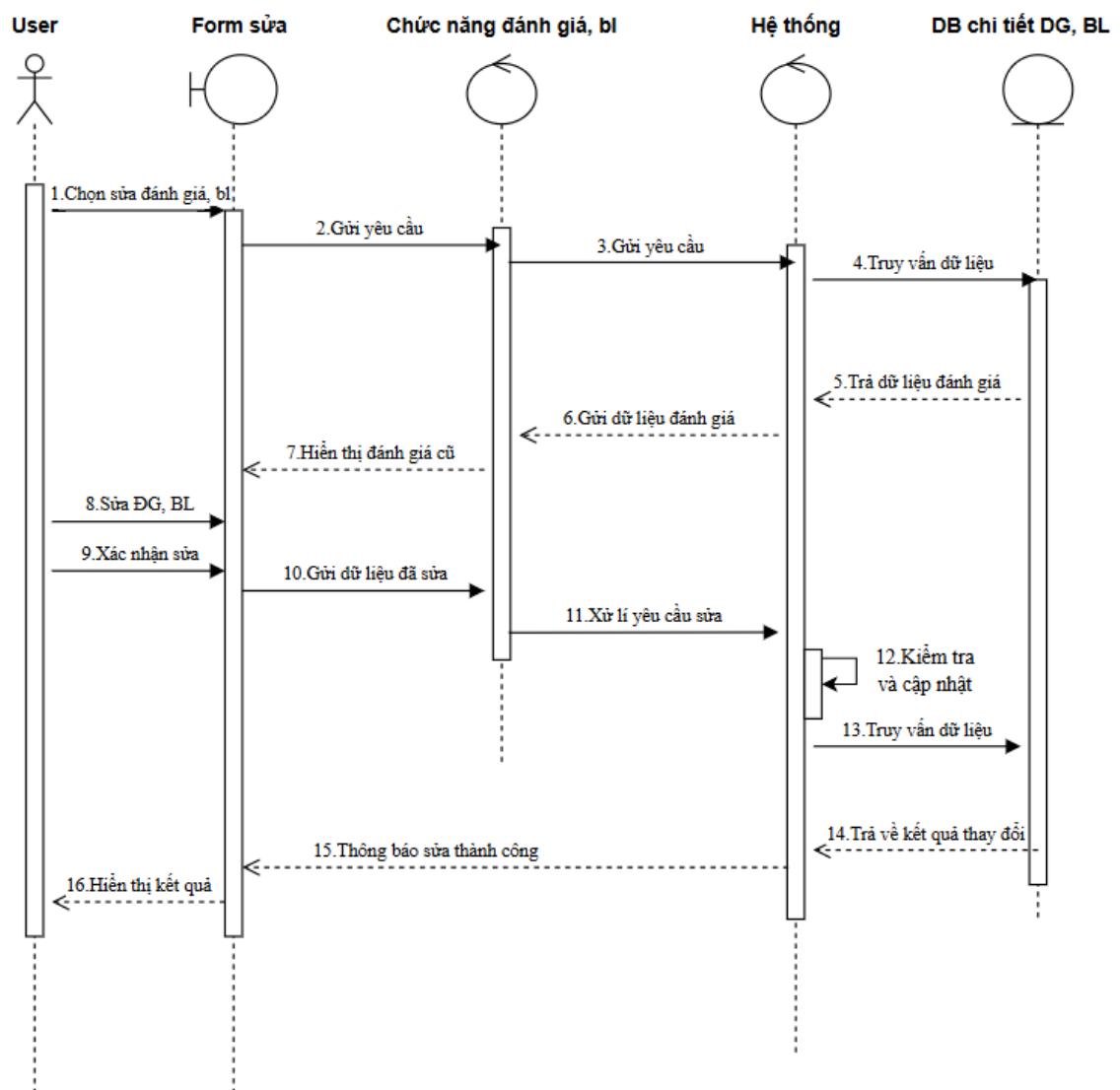
### 1.5.9 SD Bình luận và đánh giá



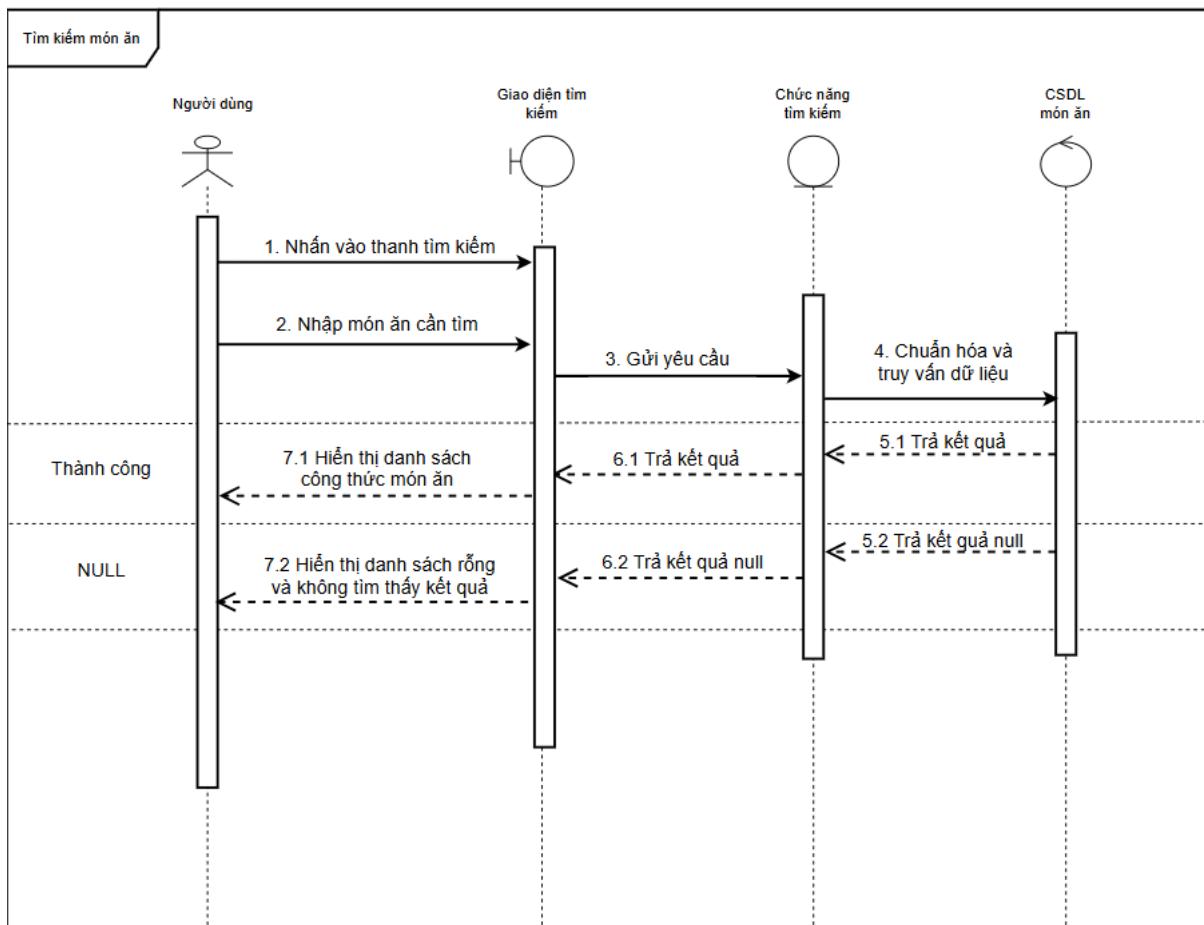
### 1.5.10 SD Xóa bình luận và đánh giá



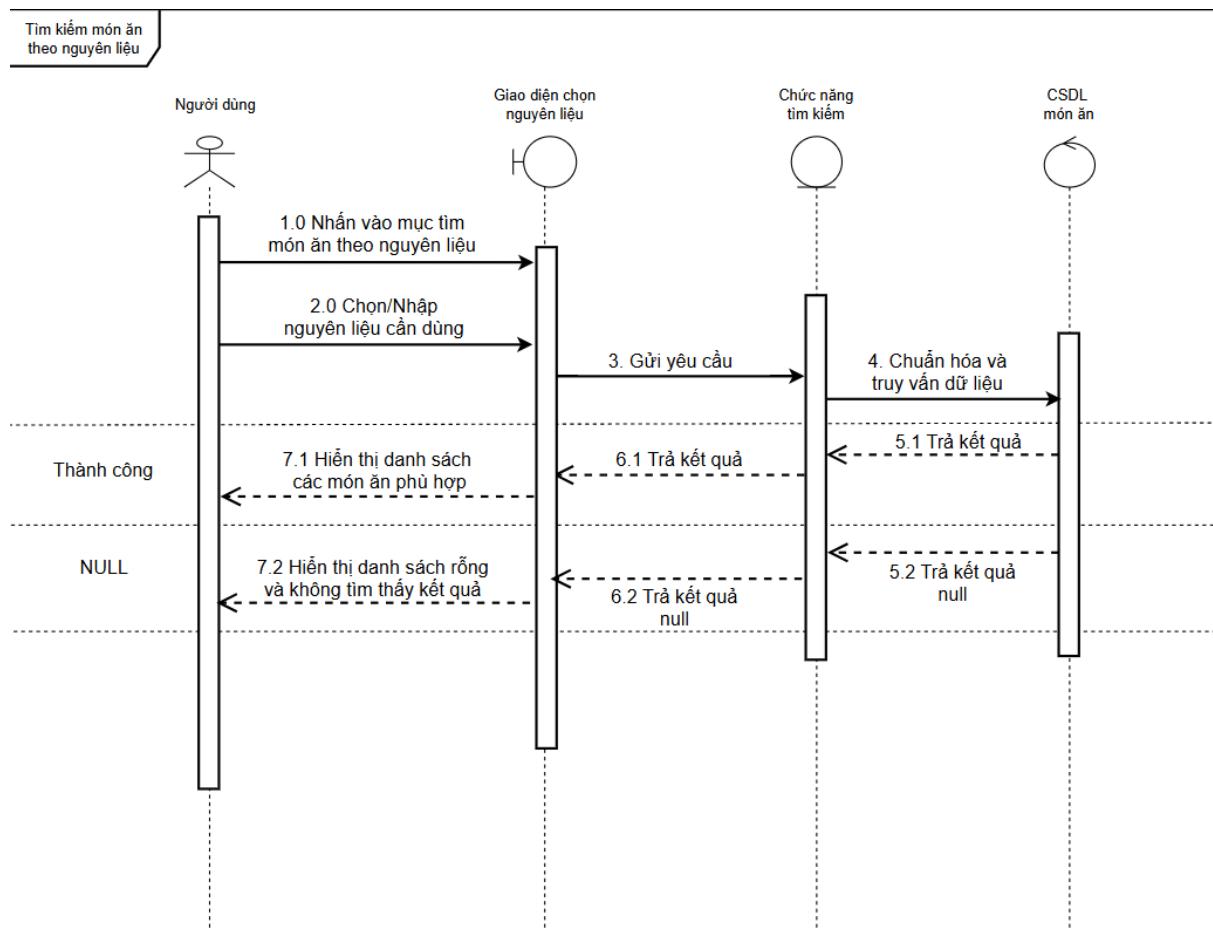
### 1.5.11 SD Sửa bình luận và đánh giá



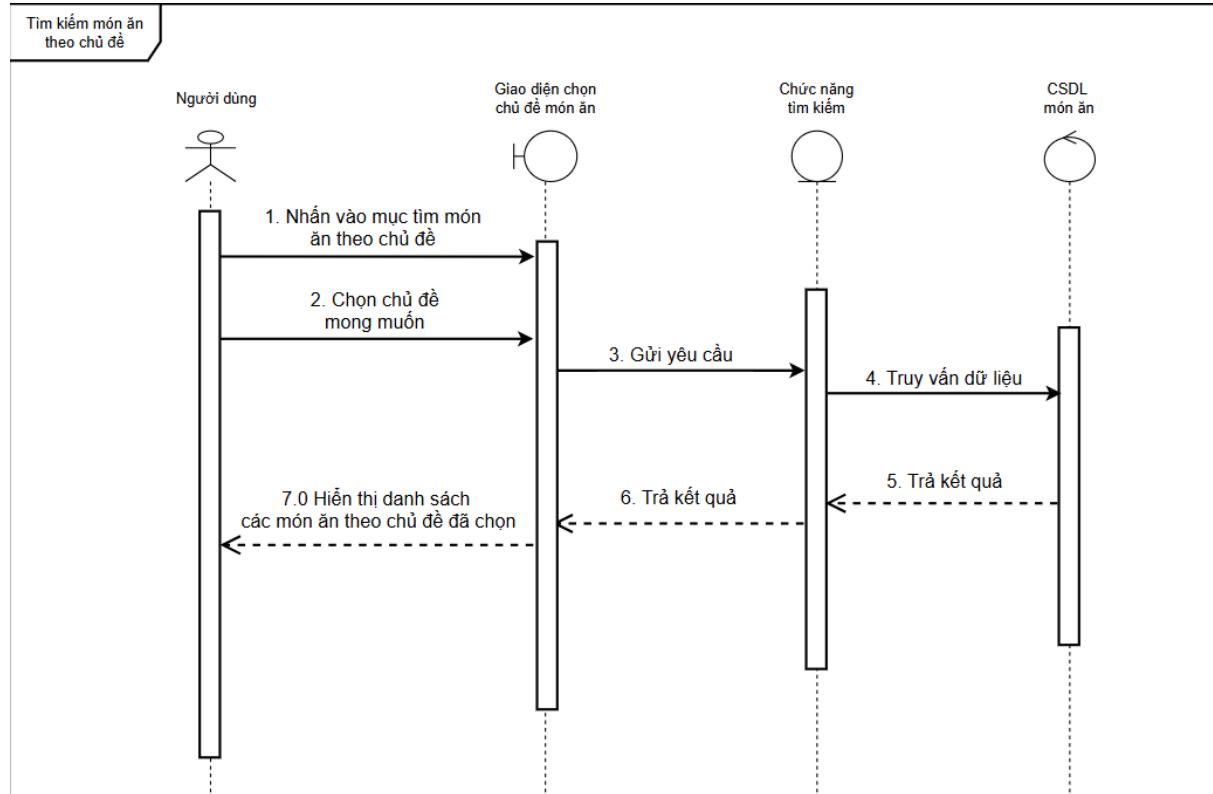
### 1.5.12 SD Tìm kiếm món ăn



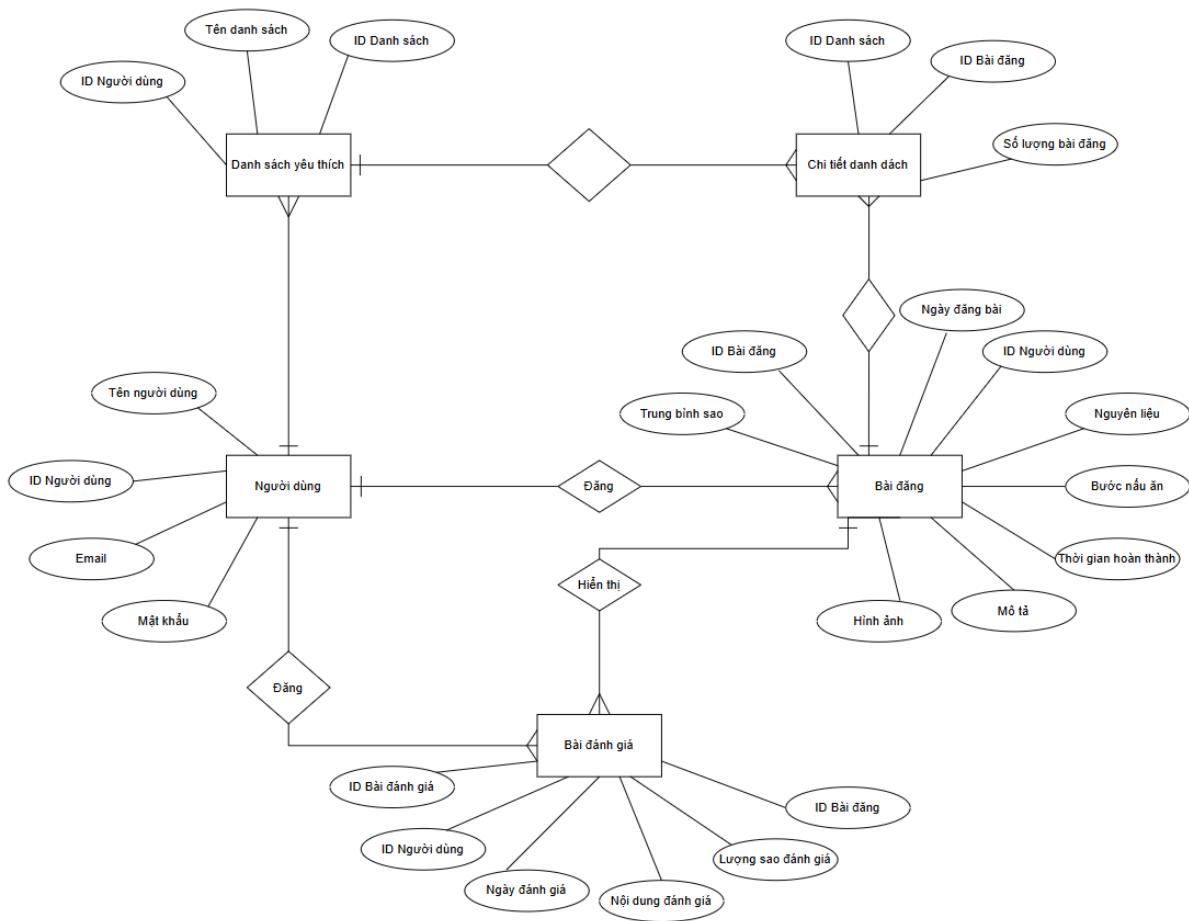
### 1.5.13 SD Gọi ý món ăn theo nguyên liệu



### 1.5.14 SD Gọi ý món ăn theo chủ đề

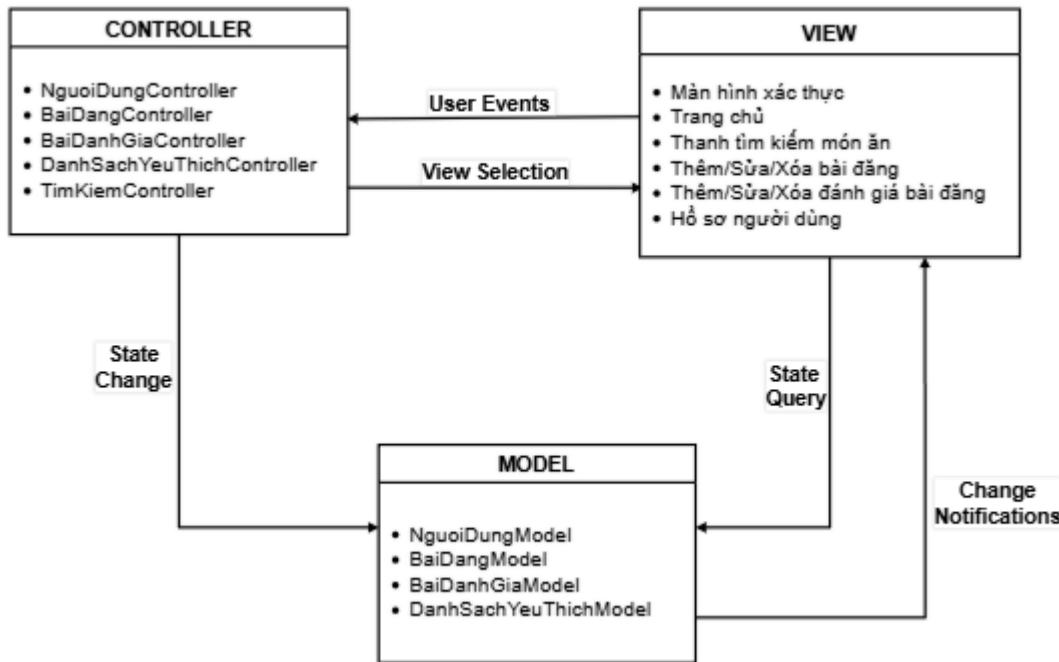


## 1.6 Biểu đồ quan hệ thực thể (ERD)



## III. 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, ...)

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.

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, ...
- DanhSachYeuThich: lưu trữ bài đăng, các thẻ loại danh sách yêu thích, ...

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 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

=> 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

### 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)
- **Baidanhgia**(ID\_baidanhgia, Ngaydanhgia, Luongsaodanhgia, Noidungdanhgia)

#### 3.2 Mối liên kết

- **Nguoidung**(ID\_nguoidung, Tennguoidung, Email, Matkhau)
- **Baidang**(ID\_baidang, Ngaydangbai, Trungbinhsao, Nguyenlieu, Buocnauan, Thoigianhoanthanh, Hinhanh, Mota)

- **Danhsachyeuthich**(ID\_danhsach, Tendanhsach)
- **Baidanhgia**(ID\_baidanhgia, Ngaydanhgia, Luongsaodanhgia, Noidungdanhgia)

### 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, ID\_nguoidung)
- **Chitietdanhsach**(ID\_danhsach, ID\_baidang, Soluongbaidang)
- **Baidanhgia**(ID\_baidanhgia, ID\_baidang, ID\_nguoidung, Ngaydanhgia, Luongsaodanhgia, Noidungdanhgia)

### 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.

#### 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 nâu 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 nâu 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

- 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 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;

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

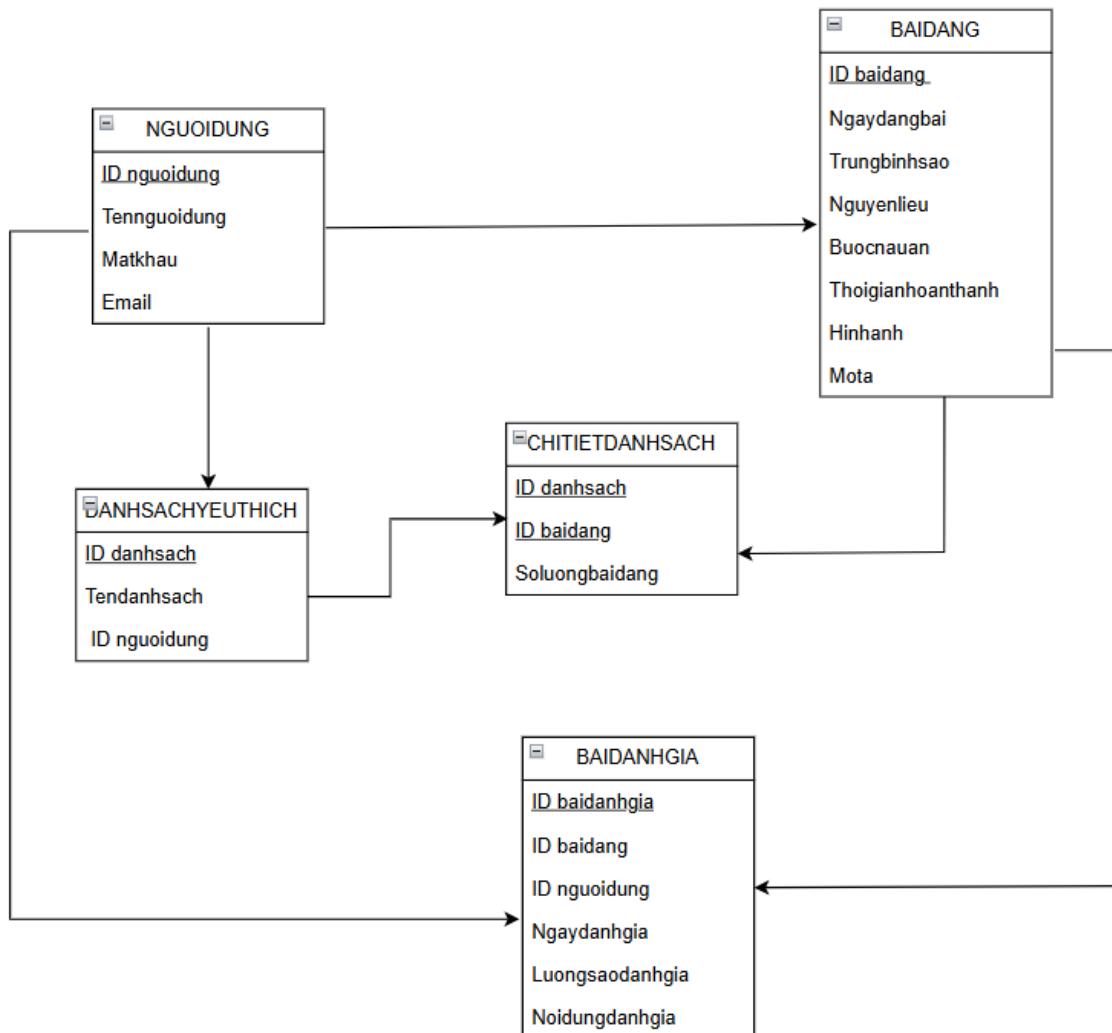
=> Đạt 1NF vì không có thuộc tính đa trị

=> 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.

### 3.5 Sơ đồ quan hệ



## 3. Thiết kế Giao diện Người dùng

[Link Figma](#)

### **III. Mã nguồn**

Link [Github](#)

### **IV. Tài liệu kiểm thử**

- Tham khảo báo cáo [kiểm thử giao diện](#)