

ĐẠI HỌC ĐÀ NẴNG  
TRƯỜNG ĐẠI HỌC SƯ PHẠM



**BÁO CÁO CUỐI KÌ**  
**PHÂN TÍCH THIẾT KẾ HƯỚNG ĐỐI TƯỢNG**

ĐỀ TÀI: PHÂN TÍCH VÀ THIẾT KẾ  
HỆ THỐNG BÁN GIÀY MIA SHOES

**Sinh viên thực hiện:** Nhóm 10

1. Trần Trọng Tuyển – 19CNTTD
2. Trần Nhật Khánh Lan – 19CNTTD
3. Nguyễn Duy Thắng – 19CNTTD

**GVHD:** Nguyễn Văn Vương

**Lớp học phần:** 19-0203

## BẢNG PHÂN CÔNG CÔNG VIỆC

Tên	Công việc	Hoàn thành
1. Trần Trọng Tuyển	Context Container Component Web	100%
2. Trần Nhật Khánh Lan	Context Use Case Diagram Sequence Diagram State Diagram	100%
3. Nguyễn Duy Thắng	Context Container	100%

# MỤC LỤC

<b>MỞ ĐẦU .....</b>	<b>1</b>
<b>CHƯƠNG 1: CƠ SỞ LÝ THUYẾT .....</b>	<b>3</b>
<b>1.1. Phân tích thiết kế hướng đối tượng (OOAD) .....</b>	<b>3</b>
1.1.1. Phân tích hướng đối tượng (OOA) .....	3
1.1.2. Thiết kế hướng đối tượng (OOD) .....	3
1.1.3. Các khái niệm cơ bản.....	4
1.1.4. Các nguyên tắc cơ bản của phương pháp Hướng đối tượng .....	4
1.1.5. Ưu điểm của Hướng đối tượng .....	5
<b>1.2. C4 Model.....</b>	<b>5</b>
1.2.1. Định nghĩa.....	5
1.2.2. Các mức độ của mô hình C4.....	5
<b>1.3. Biểu đồ UML .....</b>	<b>6</b>
1.3.1. Biểu đồ Use Case .....	6
<i>1.3.1.1. Giới thiệu biểu đồ Use Case .....</i>	<i>6</i>
<i>1.3.1.2. Các thành phần đặc tả biểu đồ Use Case.....</i>	<i>7</i>
1.3.2. Biểu đồ tuần tự (Sequence Diagram).....	9
<i>1.3.2.1. Giới thiệu biểu đồ tuần tự.....</i>	<i>9</i>
<i>1.3.2.2. Các thành phần của biểu đồ tuần tự.....</i>	<i>9</i>
<i>1.3.2.3. Các loại thông điệp trong biểu đồ tuần tự.....</i>	<i>10</i>
1.3.3. Biểu đồ trạng thái (State Diagram).....	11
<i>1.3.3.1. Giới thiệu biểu đồ trạng thái .....</i>	<i>11</i>
<i>1.3.3.2. Các thành phần của biểu đồ trạng thái .....</i>	<i>11</i>
<b>CHƯƠNG 2: KHẢO SÁT HỆ THỐNG .....</b>	<b>13</b>

<b>2.1. Phạm vi đề tài.....</b>	<b>13</b>
<b>2.2. Khảo sát thực tế .....</b>	<b>13</b>
2.2.1. Kế hoạch khảo sát .....	13
2.2.2. Mục tiêu .....	13
2.2.3. Kết quả khảo sát.....	13
<b>2.3. Quy trình xử lý .....</b>	<b>14</b>
<b>CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG .....</b>	<b>15</b>
<b>3.1. System Context Diagram.....</b>	<b>15</b>
<b>3.2. Container Diagram .....</b>	<b>15</b>
<b>3.3. Component Diagram .....</b>	<b>16</b>
<b>3.4. Biểu đồ UML .....</b>	<b>18</b>
3.4.1. Biểu đồ Use Case .....	18
3.4.2. Biểu đồ tuần tự.....	20
3.4.3. Biểu đồ trạng thái.....	21
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>23</b>

## DANH MỤC HÌNH ẢNH

Hình 1. Các mức độ của mô hình C4.....	5
Hình 2. Biểu đồ Use Case – Mối quan hệ Extend.....	8
Hình 3. Biểu đồ tuần tự - Đối tượng.....	9
Hình 4. Biểu đồ tuần tự - Đường đời đối tượng.....	9
Hình 5. Biểu đồ tuần tự - Thông điệp.....	9
Hình 6. Biểu đồ tuần tự - Xử lý bên trong đối tượng.....	10
Hình 7. Biểu đồ tuần tự - Thông điệp đồng bộ.....	10
Hình 8. Biểu đồ tuần tự - Thông điệp không đồng bộ.....	10
Hình 9. Biểu đồ tuần tự - Thông điệp chính mình.....	10
Hình 10. Biểu đồ tuần tự - Thông điệp trả về.....	11
Hình 11. Biểu đồ tuần tự - Thông điệp tạo mới.....	11
Hình 12. Biểu đồ tuần tự - Thông điệp xoá.....	11
Hình 13. Biểu đồ trạng thái – Trạng thái bắt đầu.....	11
Hình 14. Biểu đồ trạng thái – Trạng thái kết thúc.....	12
Hình 15. Biểu đồ trạng thái – Sự kiện hoặc chuyển đổi.....	12
Hình 16. Biểu đồ trạng thái – Trạng thái đối tượng.....	12
Hình 17. System Context Diagram.....	15
Hình 18. Container Diagram.....	15
Hình 19. Component Diagram – Single-Page Application Admin.....	16
Hình 20. Component Diagram – Web Application.....	17
Hình 21. Biểu đồ Use Case – Admin.....	18
Hình 22. Biểu đồ Use Case – User.....	19
Hình 23. Biểu đồ tuần tự - Đăng Nhập.....	20

<i>Hình 24. Biểu đồ tuần tự - Đăng Ký.....</i>	<i>20</i>
<i>Hình 25. Biểu đồ trạng thái – Gió hàng.....</i>	<i>21</i>
<i>Hình 26. Biểu đồ trạng thái – Đăng Ký.....</i>	<i>21</i>
<i>Hình 27. Biểu đồ trạng thái – Đăng Nhập.....</i>	<i>22</i>

## MỞ ĐẦU

### 1. Lý do chọn đề tài

Chúng ta đang sống trong thế giới mà công nghệ thông tin và truyền thông đóng vai trò hết sức quan trọng, nó ngày càng trở nên phổ biến và có ảnh hưởng sâu sắc đến cuộc sống của mỗi người. Sự xuất hiện của mạng máy tính là một dấu mốc trong lịch sử phát triển của loài người, mạng máy tính hỗ trợ cho chúng ta rất nhiều trong cuộc sống, nhờ có mạng máy tính thông tin được truyền đi một cách nhanh chóng làm cho con người ở khắp mọi nơi trên thế giới có thể giao lưu hợp tác trao đổi thông tin với nhau thuận tiện rất nhiều.

Giờ đây, mọi việc liên quan đến thông tin trở nên thật dễ dàng cho người sử dụng: chỉ cần có một máy tính kết nối internet và một dòng dữ liệu truy tìm thì gần như lập tức... cả thế giới về vấn đề mà bạn đang quan tâm sẽ hiện ra, có đầy đủ thông tin, hình ảnh và thậm chí đôi lúc có cả những âm thanh nếu bạn cần... Bằng internet, chúng ta đã thực hiện được nhiều công việc với tốc độ nhanh hơn và chi phí thấp hơn nhiều so với cách thức truyền thống. Chính điều này, đã thúc đẩy sự khai sinh và phát triển của thương mại điện tử và chính phủ điện tử trên khắp thế giới, làm biến đổi đáng kể bộ mặt văn hóa, nâng cao chất lượng cuộc sống con người.

Hiện nay bán hàng trực tuyến đã rất phổ biến, có rất nhiều website bán hàng online đáp ứng cho nhu cầu của người mua. Không thể phủ nhận sự tiện lợi, nhanh chóng thuận tiện, cùng với xu thế phát triển ngày càng cao của công nghệ thông tin. Khi xã hội ngày càng phát triển, mức sống của người dân được nâng cao thu nhập kinh tế ngày càng được cải thiện thì nhu cầu làm đẹp của con người cũng nhiều hơn. Ai cũng muốn tìm cho mình những đôi giày chất lượng, phù hợp với phong cách của mình. Vì thế, chúng em đã chọn đề tài “Phân tích và thiết kế website bán hàng trực tuyến Mía Shoes”.

### 2. Mục tiêu đề tài

Với sự phát triển của công nghệ phần mềm, nhiều công cụ lập trình đã được phát triển và có khả năng hỗ trợ mạnh cho người sử dụng, giảm thiểu công sức của lập trình viên khi xây dựng chương trình. Khi triển khai tin học hóa các bài toán trong thực tế, chúng ta không còn gặp nhiều khó khăn ở khâu viết chương trình mà vấn đề chính là ở khâu thực hiện phân tích, thiết kế hệ thống cho bài toán đó.

Phương pháp phân tích, thiết kế hệ thống hướng đối tượng giúp chúng ta hiểu rõ hơn những công việc phải làm, những yêu cầu thực tế về số liệu cần phải đáp ứng, và trên cơ sở đó xây dựng các mô hình cần thiết để mô tả mối tương quan giữa các thành phần trong hệ thống, từ đó dễ dàng hơn rất nhiều trong việc triển khai lập trình cũng như bảo trì, nâng cấp cho hệ thống hoạt động tốt, đáp ứng các yêu cầu thay đổi của người sử dụng.



## CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

### 1.1. Phân tích thiết kế hướng đối tượng (OOAD)

#### 1.1.1. Phân tích hướng đối tượng (OOA)

Là giai đoạn phát triển một mô hình chính xác và súc tích của vấn đề, có thành phần là các đối tượng và khái niệm đời thực, dễ hiểu đối với người sử dụng.

Trong giai đoạn OOA, vấn đề được trình bày bằng các thuật ngữ tương ứng với các đối tượng có thực. Thêm vào đó, hệ thống cần phải được định nghĩa sao cho người không chuyên Tin học có thể dễ dàng hiểu được.

Dựa trên một vấn đề có sẵn, nhà phân tích cần ánh xạ các đối tượng hay thực thể có thực như khách hàng, ô tô, người bán hàng, ... vào thiết kế để tạo ra được bản thiết kế gần cận với tình huống thực. Mô hình thiết kế sẽ chứa các thực thể trong một vấn đề có thực và giữ nguyên các mẫu hình về cấu trúc, quan hệ cũng như hành vi của chúng. Nói một cách khác, sử dụng phương pháp hướng đối tượng chúng ta có thể mô hình hóa các thực thể thuộc một vấn đề có thực mà vẫn giữ được cấu trúc, quan hệ cũng như hành vi của chúng.

#### 1.1.2. Thiết kế hướng đối tượng (OOD)

Là giai đoạn tổ chức chương trình thành các tập hợp đối tượng cộng tác, mỗi đối tượng trong đó là thực thể của một lớp. Các lớp là thành viên của một cây cấu trúc với mối quan hệ thừa kế.

Mục đích của giai đoạn OOD là tạo thiết kế dựa trên kết quả của giai đoạn OOA, dựa trên những quy định phi chức năng, những yêu cầu về môi trường, những yêu cầu về khả năng thực thi, .... OOD tập trung vào việc cải thiện kết quả của OOA, tối ưu hóa giải pháp đã được cung cấp trong khi vẫn đảm bảo thỏa mãn tất cả các yêu cầu đã được xác lập.

Trong giai đoạn OOD, nhà thiết kế định nghĩa các chức năng, thủ tục (operations), thuộc tính (attributes) cũng như mối quan hệ của một hay nhiều lớp (class) và quyết định chúng cần phải được điều chỉnh sao cho phù hợp với môi trường phát triển. Đây cũng là giai đoạn để thiết kế ngân hàng dữ liệu và áp dụng các kỹ thuật tiêu chuẩn hóa.

Về cuối giai đoạn OOD, nhà thiết kế đưa ra một loạt các biểu đồ (diagram) khác nhau. Các biểu đồ này có thể được chia thành hai nhóm chính là Tĩnh và Động. Các biểu đồ tĩnh biểu thị các lớp và đối tượng, trong khi biểu đồ động biểu thị tương tác giữa các lớp và phương thức hoạt động

chính xác của chúng. Các lớp đó sau này có thể được nhóm thành các gói (Packages) tức là các đơn vị thành phần nhỏ hơn của ứng dụng.

#### 1.1.3. Các khái niệm cơ bản

- Đối tượng (Object): Là khái niệm cho phép mô tả các sự vật/ thực thể trong thế giới thực.
- Lớp (Class): Mô tả một tập hợp các đối tượng có cùng một cấu trúc, cùng hành vi và có cùng những mối quan hệ với các đối tượng khác. Như vậy, một đối tượng là thể hiện của một lớp và một lớp là một định nghĩa trừu tượng của đối tượng.
- Gói (Package): Là một cách tổ chức các thành phần, phần tử trong hệ thống thành các nhóm. Nhiều gói có thể được kết hợp với nhau để trở thành một hệ thống con.
- Kế thừa (Inheritance): Trong phương pháp hướng đối tượng, một lớp có thể sử dụng lại các thuộc tính và phương thức của một hoặc nhiều lớp khác.

#### 1.1.4. Các nguyên tắc cơ bản của phương pháp Hướng đối tượng

- Trừu tượng hóa (abstraction):
  - Các thực thể phần mềm được mô hình hoá dưới dạng các đối tượng.
  - Các đối tượng được trừu tượng hóa ở mức cao hơn dựa trên thuộc tính và phương thức mô tả đối tượng để tạo thành lớp.
  - Các lớp được trừu tượng hóa ở mức cao hơn nữa để tạo thành một sơ đồ các lớp được kế thừa lẫn nhau. Trong phương pháp hướng đối tượng có thể tồn tại những lớp không có đối tượng tương ứng gọi là lớp trừu tượng.
- Tính đóng gói (encapsulation): che dấu mọi chi tiết hiện thực của đối tượng không cho bên ngoài thấy và truy xuất. Đây chính là tính độc lập cao giữa các đối tượng.
  - Che dấu các thuộc tính dữ liệu : nếu cần cho phép truy xuất một thuộc tính dữ liệu, ta tạo hai phương thức set/get tương ứng để giám sát việc truy xuất và che dấu chi tiết hiện thực bên trong (thuộc tính private).
  - Che dấu chi tiết hiện thực các phương thức.
  - Che dấu các hàm và sự hiện thực của chúng.
- Tính mô-đun hóa (modularity): các bài toán sẽ được phân chia thành những vấn đề nhỏ hơn, đơn giản và quản lý được.
- Tính phân cấp (hierarchy): cấu trúc chung của một hệ thống hướng đối tượng là dạng phân cấp theo các mức độ trừu tượng từ cao đến thấp.

### 1.1.5. Ưu điểm của Hướng đối tượng

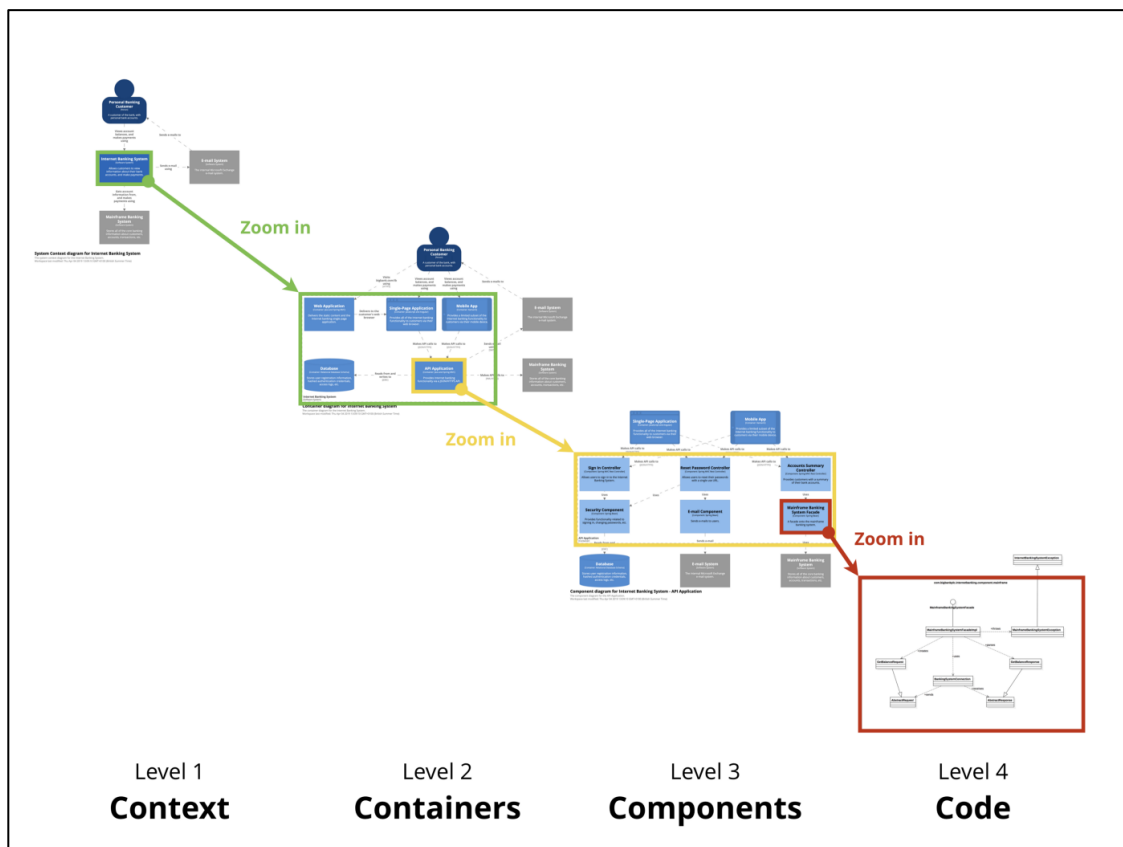
- Gần gũi với thế giới thực
- Tái sử dụng dễ dàng
- Đóng gói, che dấu thông tin làm cho hệ thống tin cậy hơn
- Thừa kế làm giảm chi phí, hệ thống có tính mở cao hơn
- Xây dựng hệ thống lớn và phức tạp

## 1.2. C4 Model

### 1.2.1. Định nghĩa

Mô hình C4 là một phương pháp mô tả kiến trúc phần mềm một cách tinh gọn nhất. Nó bao gồm một tập hợp 4 sơ đồ mô tả cấu trúc tĩnh của một hệ thống phần mềm.

### 1.2.2. Các mức độ của mô hình C4



Hình 1. Các mức độ của mô hình C4

- **System Context:** Sơ đồ này mô tả tổng quát về hệ thống theo hướng che đi thành phần bên trong mà chỉ làm nổi bật thành phần bên ngoài (blackbox); bao gồm các yếu tố phụ thuộc chính (dependencies) của hệ thống, các giao thức (interfaces) để giao tiếp giữa những hệ thống với nhau và con người (người dùng / theo vai trò / theo phòng ban / v.v). Biểu đồ ngữ cảnh này là tiêu chuẩn trong kỹ thuật phần mềm.
- **Container:** Sơ đồ này mô tả về hệ thống theo hướng hiển thị thành phần bên ngoài cùng với việc phóng to những thành phần bên trong của hệ thống (whitebox); do đó hiển thị được các container (building block), mục đích và nhiệm vụ của từng container này, cùng với giao thức giao tiếp giữa những container đó. Sơ đồ này thường được gọi là sơ đồ khối cấp 1 (first level building block). Vậy container cụ thể là gì? Tùy theo phạm vi hệ thống các bạn đang làm (có thể là hệ thống, có thể là 1 component, cũng có thể là 1 hệ thống con), container có thể là cái máy tính, vi xử lý, cũng có thể là 1 dịch vụ (service) nào đó gọi email, lưu trữ, APIs...
- **Component:** Sơ đồ này phóng to container để mô tả về thành phần bên trong; do đó hiển thị được các components (building block), mục đích và nhiệm vụ, mối quan hệ của từng component cũng như là những giao thức kết nối. Sơ đồ này thường được gọi là sơ đồ khối cấp 2.
- **Code (hay class):** Sơ đồ này mô tả bên trong một component; theo đó hiển thị được các phần hiện thực như là class, package... cùng với mối quan hệ giữa những thành phần này.

### 1.3. Biểu đồ UML

#### 1.3.1. Biểu đồ Use Case

##### 1.3.1.1. Giới thiệu biểu đồ Use Case

Use Case mô tả sự tương tác giữa người dùng và hệ thống ở trong một môi trường cụ thể, vì một mục đích cụ thể. Môi trường nằm trong một bối cảnh, phạm vi hoặc hệ thống phần mềm cụ thể. Mục đích cụ thể là diễn tả được yêu cầu theo góc nhìn từ phía người dùng.

Sự tương tác giữa người dùng và hệ thống có 2 cách thức phổ biến:

- Cách thức mà người dùng tương tác với hệ thống.
- Cách thức mà hệ thống tương tác với các hệ thống khác.

### *1.3.1.2. Các thành phần đặc tả biểu đồ Use Case*

#### **Actor (Người sử dụng)**

Actor là thành phần chỉ người dùng hoặc một đối tượng nào đó bên ngoài tương tác với hệ thống. Để xác nhận đó có phải là Actor hay không thì cần xem xét dựa vào những câu hỏi sau:

- Ai là người sử dụng chức năng chính của hệ thống (tác nhân chính)?
- Ai sẽ là admin của hệ thống – Người cài đặt, quản lý và bảo trì hệ thống (tác nhân phụ)?
- Ai sẽ cần hệ thống hỗ trợ để thực hiện các tác vụ hằng ngày?
- Hệ thống này có cần phải tương tác với các hệ thống nào khác không?
- Ai là người input dữ liệu vào hệ thống (trường hợp hệ thống lưu trữ dữ liệu)?
- Ai hay cái gì quan tâm đến giá trị mà hệ thống sẽ mang lại?

#### **Use Case (Chức năng tương tác)**

Use Case là các chức năng mà các Actor sẽ sử dụng hay thể hiện sự tương tác giữa người dùng và hệ thống. Để tìm ra được các Use Case, ta cần trả lời những câu hỏi sau:

- Actor cần những chức năng nào của hệ thống?
- Actor có hành động chính là gì?
- Actor có cần đọc, thêm mới, hủy bỏ, chỉnh sửa hay lưu trữ loại thông tin nào trong hệ thống không?
- Hệ thống có cần thông báo những thay đổi bất ngờ trong nội bộ cho Actor không?
- Công việc hàng ngày của Actor có thể được đơn giản hóa hoặc hữu hiệu hóa qua các chức năng của hệ thống?
- Use Case có thể được tạo ra bởi sự kiện nào khác không?
- Hệ thống cần những thông tin đầu vào/đầu ra nào? Những thông tin đó sẽ đi từ đâu đến đâu?
- Những khó khăn và thiếu hụt của hệ thống hiện tại nằm ở đâu?

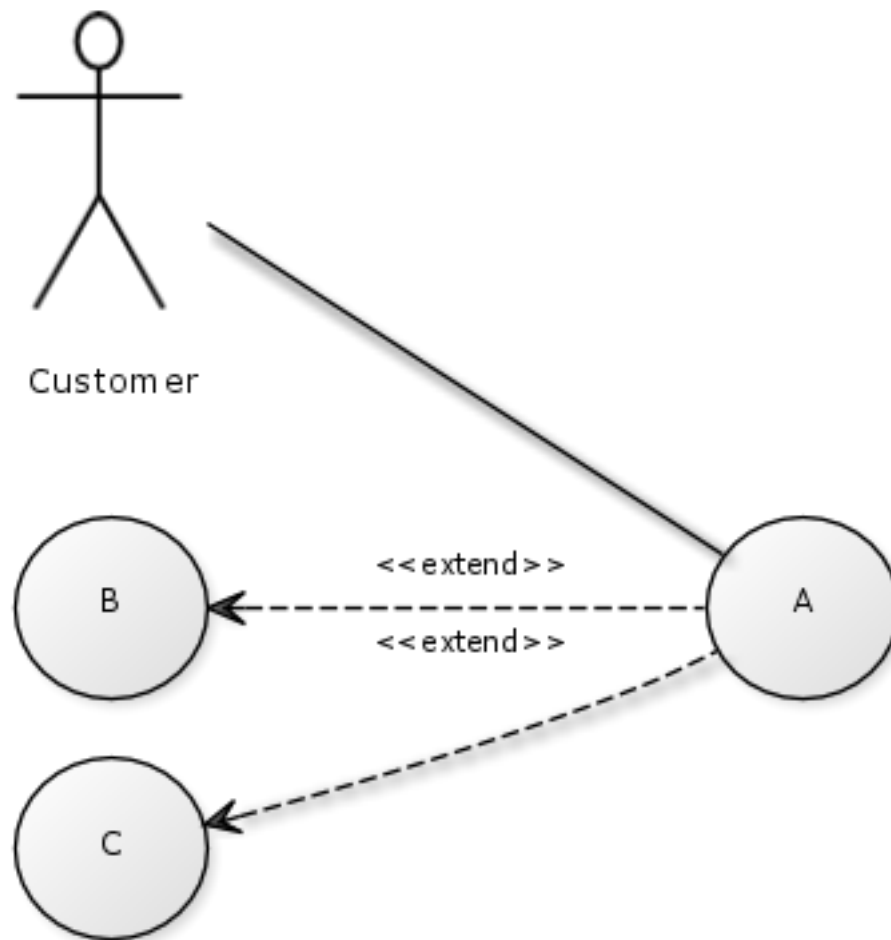
#### **Relationship (Các quan hệ trong Use Case)**

Các quan hệ trong Use Case gồm 3 loại: Include, Extend & Generalization.

- Mỗi quan hệ Include: Include là mối quan hệ bao gồm hoặc bắt buộc phải có giữa các Use Case với nhau. Hiểu đơn giản hơn: Để Use Case A xảy ra thì phải đạt được Use Case B.

- Mối quan hệ Extend

Extend biểu diễn mối quan hệ mở rộng, không bắt buộc, có thể có hoặc không giữa các Use Case với nhau.



Hình 2. Biểu đồ Use Case – Mối quan hệ Extend

Mối quan hệ extend thể hiện mối quan hệ không bắt buộc giữa các Use Case với nhau

- Mối quan hệ Generalization: Generalization là mối quan hệ cha con giữa các Use Case với nhau. Generalization còn thể hiện khả năng thể hiện mối quan hệ giữa các Actor với nhau.

### 1.3.2. Biểu đồ tuần tự (Sequence Diagram)

#### 1.3.2.1. Giới thiệu biểu đồ tuần tự

Biểu đồ tuần tự là biểu đồ dùng để xác định các trình tự diễn ra sự kiện của một nhóm đối tượng nào đó. Nó miêu tả chi tiết các thông điệp được gửi và nhận giữa các đối tượng đồng thời cũng chú trọng đến việc trình tự về mặt thời gian gửi và nhận các thông điệp đó.

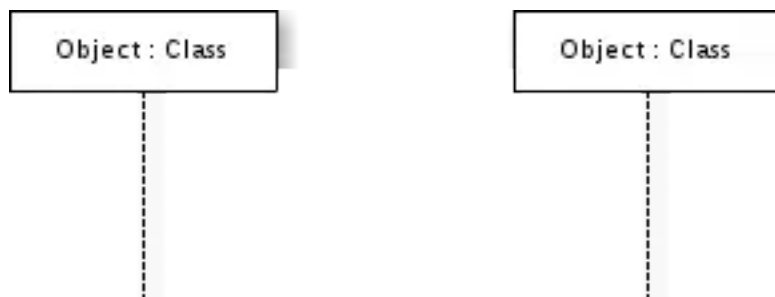
#### 1.3.2.2. Các thành phần của biểu đồ tuần tự

- Đối tượng (object or class): biểu diễn bằng các hình chữ nhật.



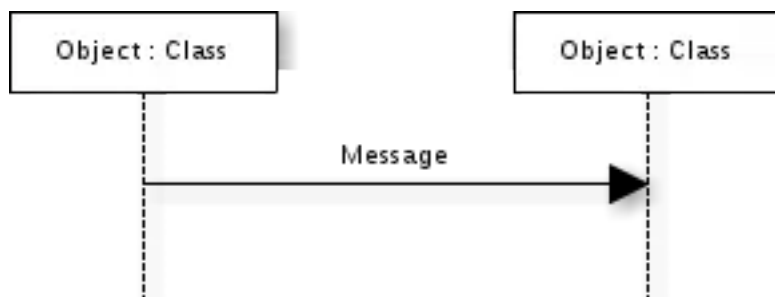
Hình 3. Biểu đồ tuần tự - Đối tượng

- Đường đời đối tượng (Lifelines): biểu diễn bằng các đường gạch rời thẳng đứng bên dưới các đối tượng.



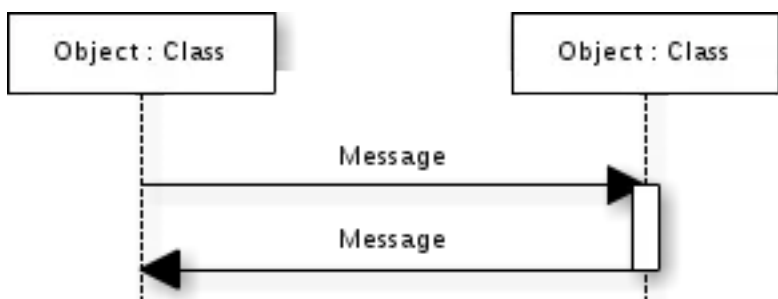
Hình 4. Biểu đồ tuần tự - Đường đời đối tượng

- Thông điệp (Message): biểu diễn bằng các đường mũi tên. Thông điệp được dùng để giao tiếp giữa các đối tượng và lớp. Có nhiều loại thông điệp được định nghĩa ở phần 1.2.3.3.



Hình 5. Biểu đồ tuần tự - Thông điệp

- Xử lý bên trong đối tượng (biểu diễn bằng các đoạn hình chữ nhật rỗng nối với các đường đời đối tượng).



Hình 6. Biểu đồ tuần tự - Xử lý bên trong đối tượng

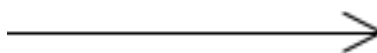
#### 1.3.2.3. Các loại thông điệp trong biểu đồ tuần tự

- Thông điệp đồng bộ (Synchronous Message): Thông điệp đồng bộ cần có một request trước hành động tiếp theo.



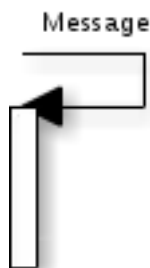
Hình 7. Biểu đồ tuần tự - Thông điệp đồng bộ

- Thông điệp không đồng bộ (Asynchronous Message): Thông điệp không đồng bộ không cần có một request trước hành động tiếp theo.



Hình 8. Biểu đồ tuần tự - Thông điệp không đồng bộ

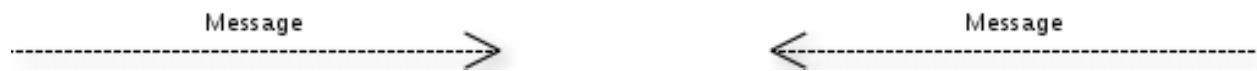
- Thông điệp chính mình (Self Message): Là thông điệp mà đối tượng gửi cho chính nó để thực hiện các hàm nội tại.



Hình 9. Biểu đồ tuần tự - Thông điệp chính mình

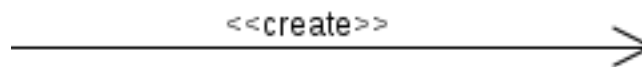


- Thông điệp trả lời hoặc trả về (Reply or Return Message): Là thông điệp trả lời lại khi có request hoặc sau khi kiểm tra tính đúng đắn của một điều kiện nào đó. Ví dụ thông điệp loại này như tin nhắn trả về là success hoặc fail.



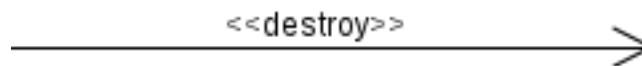
Hình 10. Biểu đồ tuần tự - Thông điệp trả về

- Thông điệp tạo mới (Create Message): Là thông điệp được trả về khi tạo mới một đối tượng.



Hình 11. Biểu đồ tuần tự - Thông điệp tạo mới

- Thông điệp xóa (Delete Message) Là thông điệp được trả về khi xóa một đối tượng.



Hình 12. Biểu đồ tuần tự - Thông điệp xóa

### 1.3.3. Biểu đồ trạng thái (State Diagram)

#### 1.3.3.1. Giới thiệu biểu đồ trạng thái

Biểu đồ trạng thái là dạng biểu đồ mô tả các trạng thái có thể có và sự chuyển đổi giữa các trạng thái đó khi có các sự kiện tác động của một đối tượng.

Đối với các đối tượng có nhiều trạng thái thì biểu đồ trạng thái là sự lựa chọn tốt nhất giúp chúng ta có thể hiểu rõ hơn về hệ thống.

#### 1.3.3.2. Các thành phần của biểu đồ trạng thái

- Trạng thái bắt đầu: (Initial State)



Hình 13. Biểu đồ trạng thái – Trạng thái bắt đầu

- Trạng thái kết thúc: (Final State)



Hình 14. Biểu đồ trạng thái – Trạng thái kết thúc

Trong biểu đồ, đường mũi tên chỉ ra sự biến đổi từ một trạng thái sang trạng thái khác.

- Sự kiện (Event) hoặc Chuyển đổi (Transition)



Hình 15. Biểu đồ trạng thái – Sự kiện hoặc chuyển đổi

- Trạng thái đối tượng (State)



Hình 16. Biểu đồ trạng thái – Trạng thái đối tượng

## CHƯƠNG 2: KHẢO SÁT HỆ THỐNG

### 2.1. Phạm vi đề tài

Mia Shoes là cửa hàng chuyên cung cấp các mặt hàng giày Bitis. Với lượng khách càng ngày càng tăng, để phục vụ khách được tốt hơn, chính xác hơn và nhanh chóng hơn thì chủ cửa hàng có ý định xây dựng một trang bán hàng trực tuyến cho cửa hàng. Mong muốn của chủ cửa hàng bao gồm: Giới thiệu sản phẩm, mua hàng trực tuyến, thông báo hàng mới về và các chương trình khuyến mãi.

### 2.2. Khảo sát thực tế

#### 2.2.1. Kế hoạch khảo sát

Khảo sát trực tiếp tại cửa hàng Mia Shoes.

Đối tượng khảo sát: Chủ cửa hàng.

#### 2.2.2. Mục tiêu

- Hiểu được quá trình quản lý và tổ chức phân công công việc cho nhân viên.
- Nắm bắt được quá trình mua bán giữa khách hàng và nhân viên.
- Hiểu rõ được mong muốn của chủ cửa hàng đối với trang web.

#### 2.2.3. Kết quả khảo sát

Câu hỏi	Câu trả lời
Câu 1: Cửa hàng tổ chức hệ thống nhân viên như thế nào?	Cửa hàng bao gồm: người quản lý và nhân viên
Câu 2: Quy trình mua hàng và thanh toán tại cửa hàng như thế nào?	Khách hàng chọn hàng mà mình mong muốn. Nếu không tìm thấy hàng hoặc có thắc mắc gì sẽ hỏi nhân viên để được nhân viên tư vấn. sau khi chọn được mặt hàng mình mong muốn thì đưa đến quầy thu ngân để tính tiền. Thu ngân nhận sản phẩm quét mã vạch rồi nhập số lượng và tiến hành thanh toán. Thu ngân nhận tiền từ

	khách hàng và in hóa đơn sau đó đưa hóa đơn và hàng cho khách.
Câu 3: Cửa hàng thực hiện quản lý nhân viên như nào?	Tôi quản lý nhân viên của mình bằng hệ thống quản lý nhân viên của cửa hàng
Câu 4: Khách hàng khi mua hàng thì có những hình thức thanh toán nào?	Khách hàng có thể thanh toán bằng thẻ và bằng tiền mặt
Câu 5: Mong muốn đối với trang bán hàng trực tuyến là gì?	<ul style="list-style-type: none"> <li>• Giao diện dễ nhìn</li> <li>• Giới thiệu chi tiết sản phẩm</li> <li>• Mua hàng trực tuyến</li> <li>• Gửi tin nhắn hàng mới về và chương trình khuyến mãi</li> </ul>

### 2.3. Quy trình xử lý

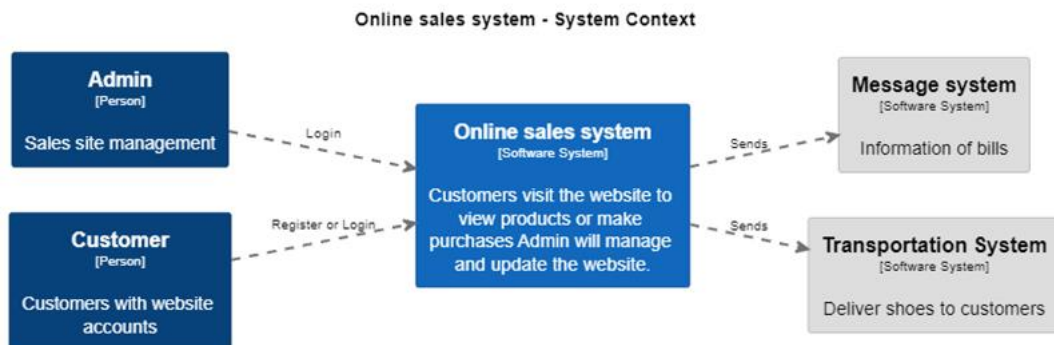
Yêu cầu một trang bán hàng trực tuyến giới thiệu chi tiết các sản phẩm của cửa hàng với người dùng. Cho phép người dùng đăng ký tài khoản và đăng nhập để xem sản phẩm và mua hàng.

Sau khi khách hàng chọn được sản phẩm ưng ý thì chọn phương thức thanh toán và nhập địa chỉ giao hàng để thanh toán. Sau khi thanh toán thì sẽ hiển thị hoá đơn của đơn hàng vừa mua. Hoá đơn gồm có: Tên sản phẩm, Số lượng đặt, Đơn giá, Thành tiền.

Khi có hàng mới về hoặc chương trình khuyến mãi thì sẽ gửi tin nhắn cho khách hàng.

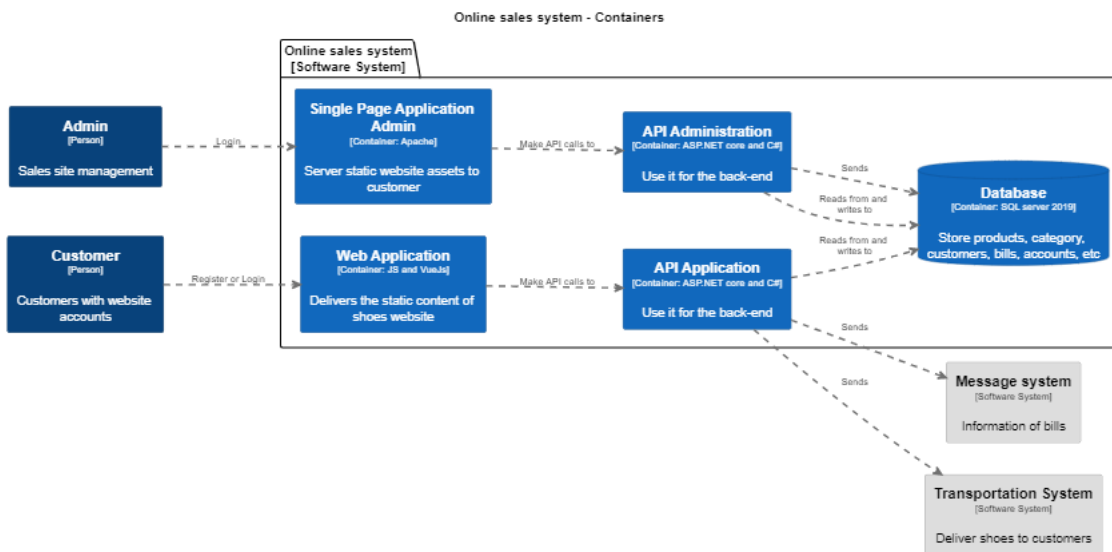
## CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

### 3.1. System Context Diagram



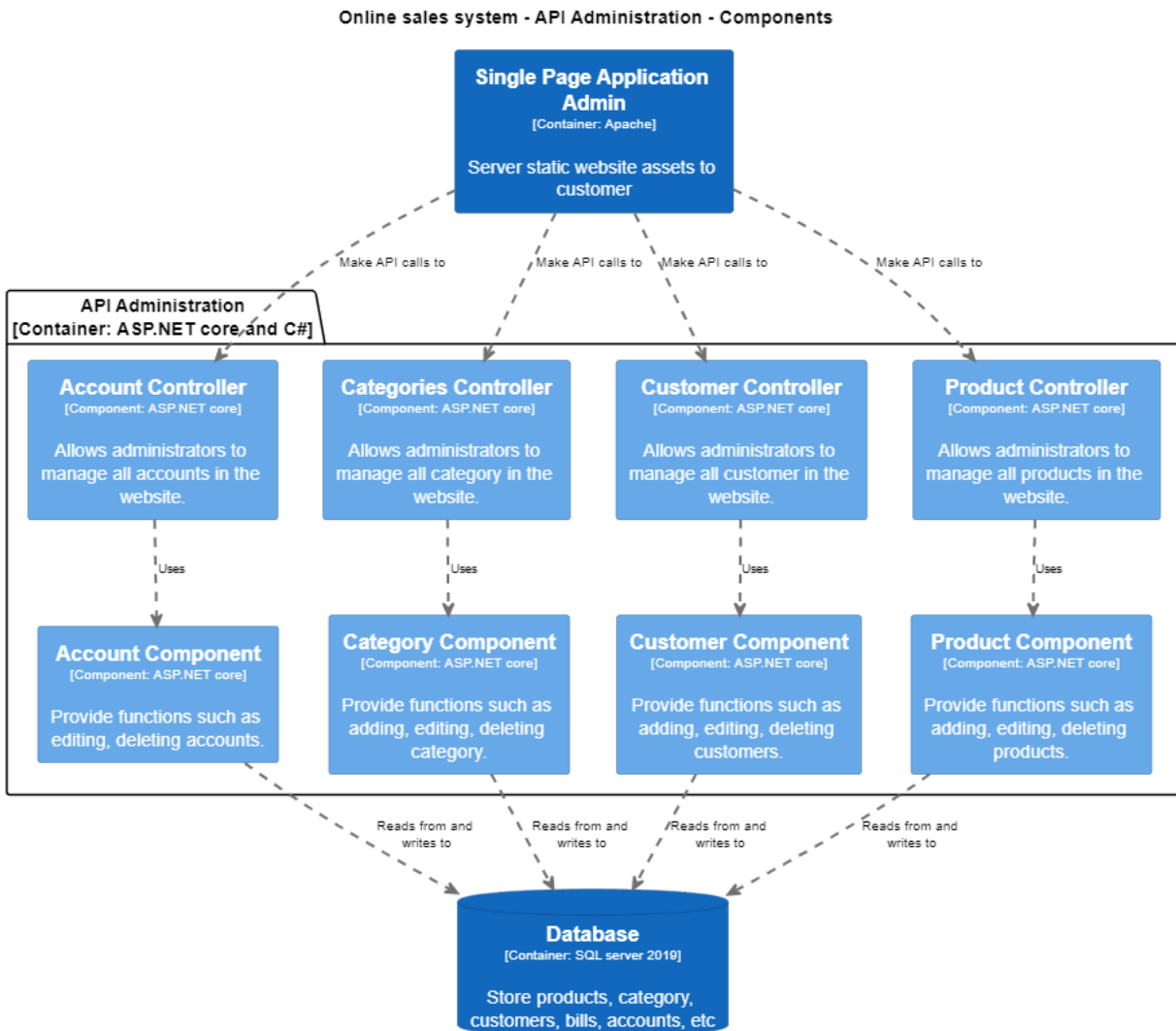
Hình 17. System Context Diagram

### 3.2. Container Diagram

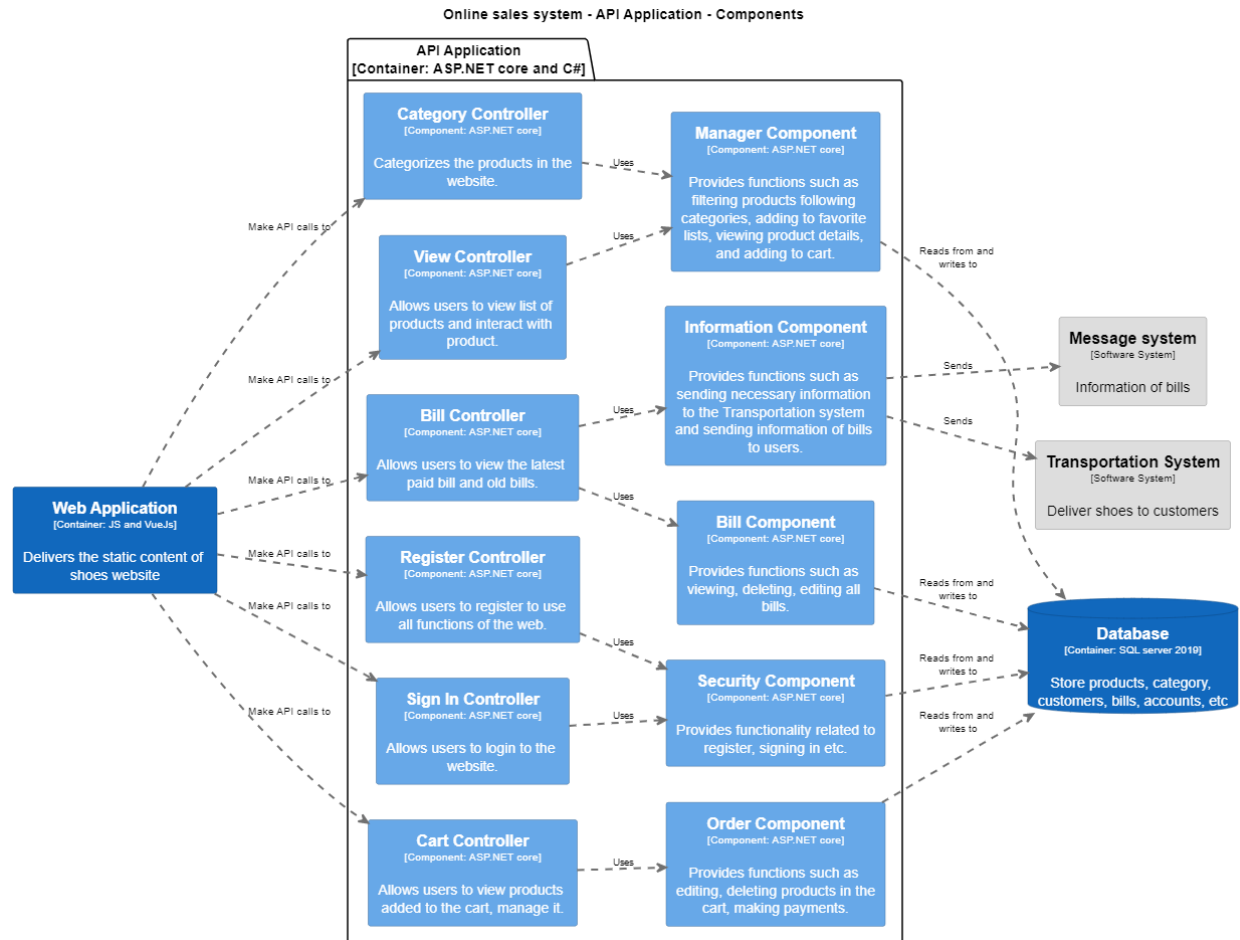


Hình 18. Container Diagram

### 3.3. Component Diagram



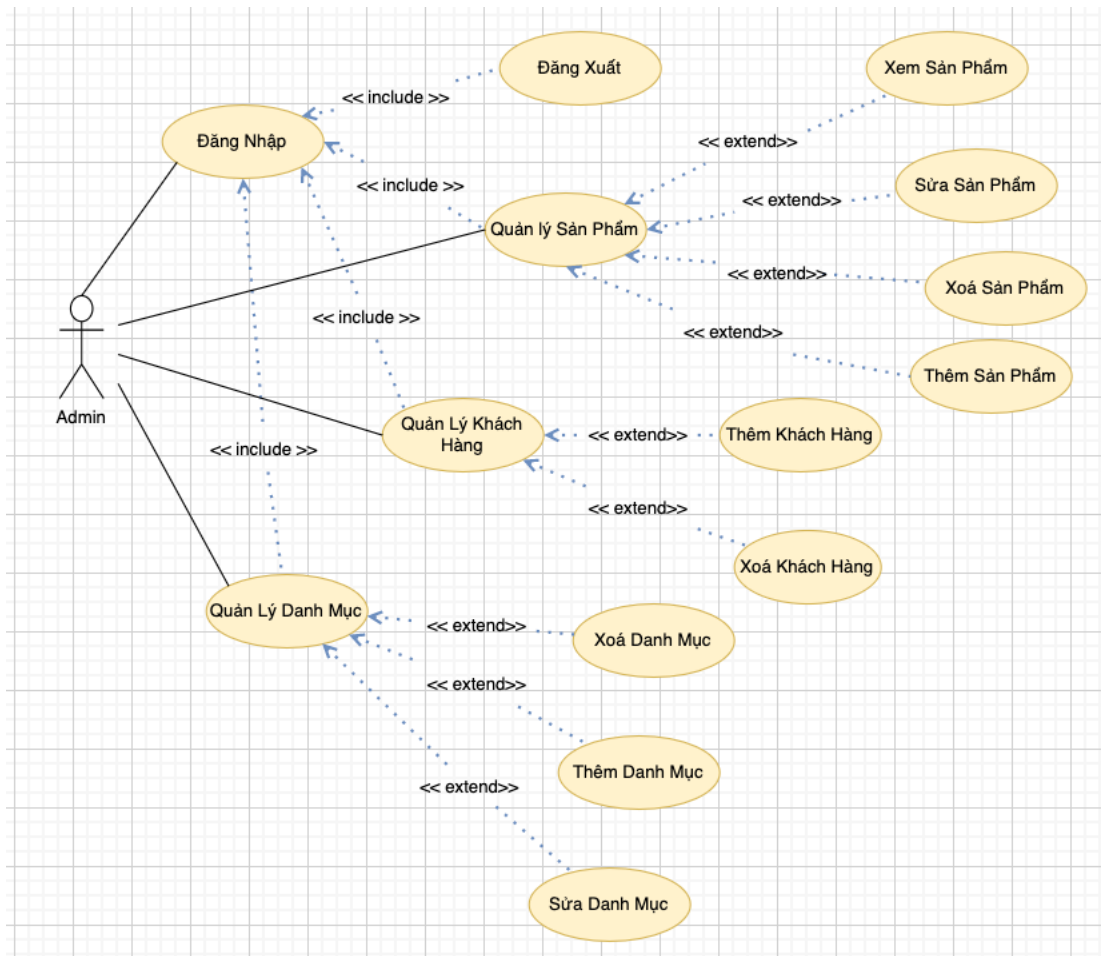
Hình 19. Component Diagram – Single-Page Application Admin



Hình 20. Component Diagram – Web Application

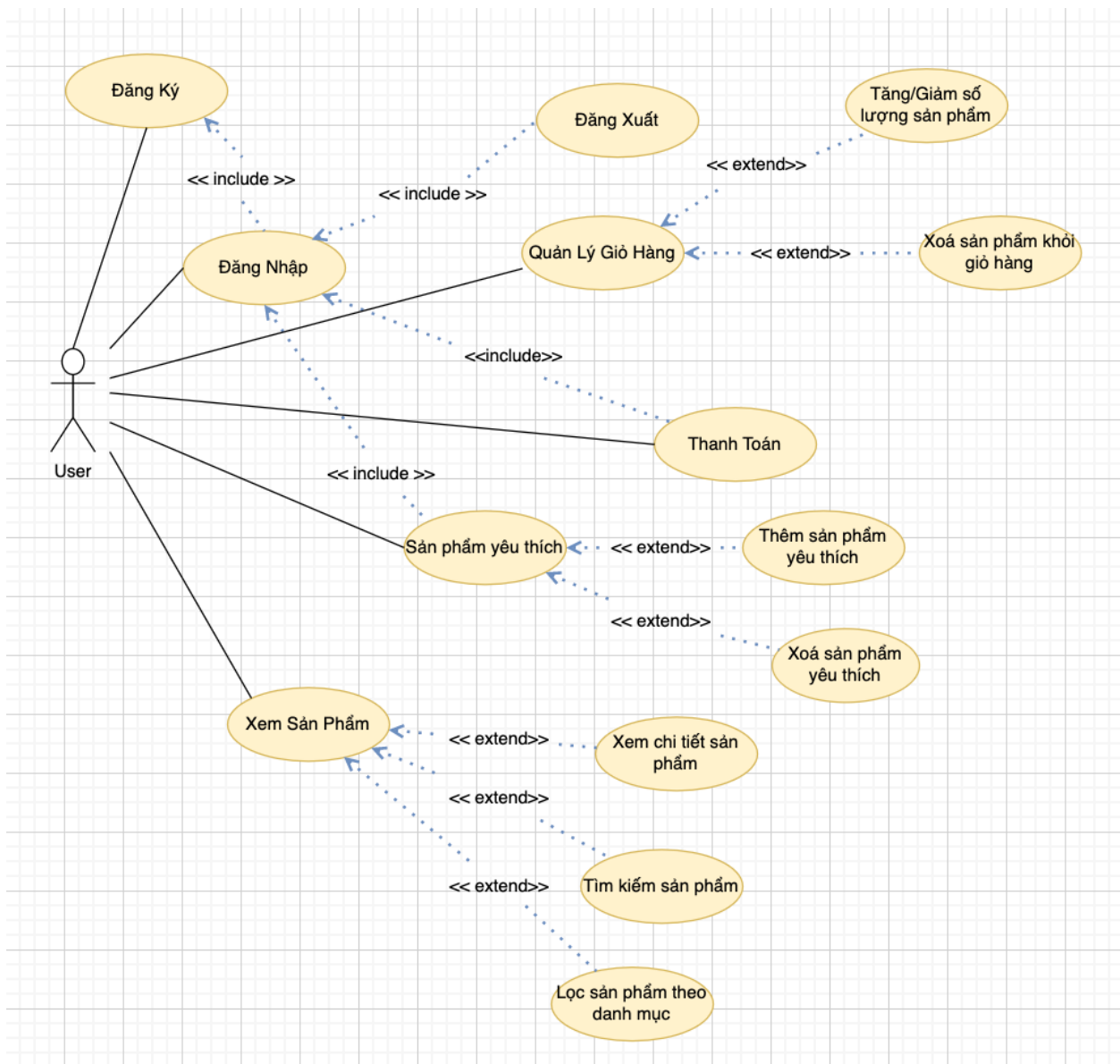
### 3.4. Biểu đồ UML

#### 3.4.1. Biểu đồ Use Case



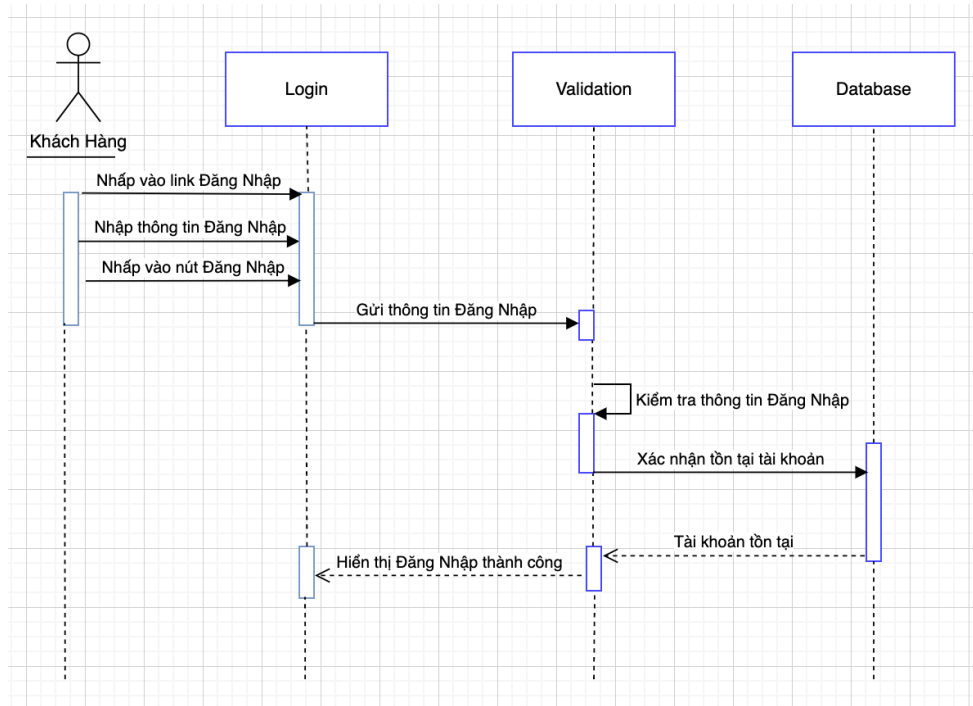
Hình 21. Biểu đồ Use Case – Admin



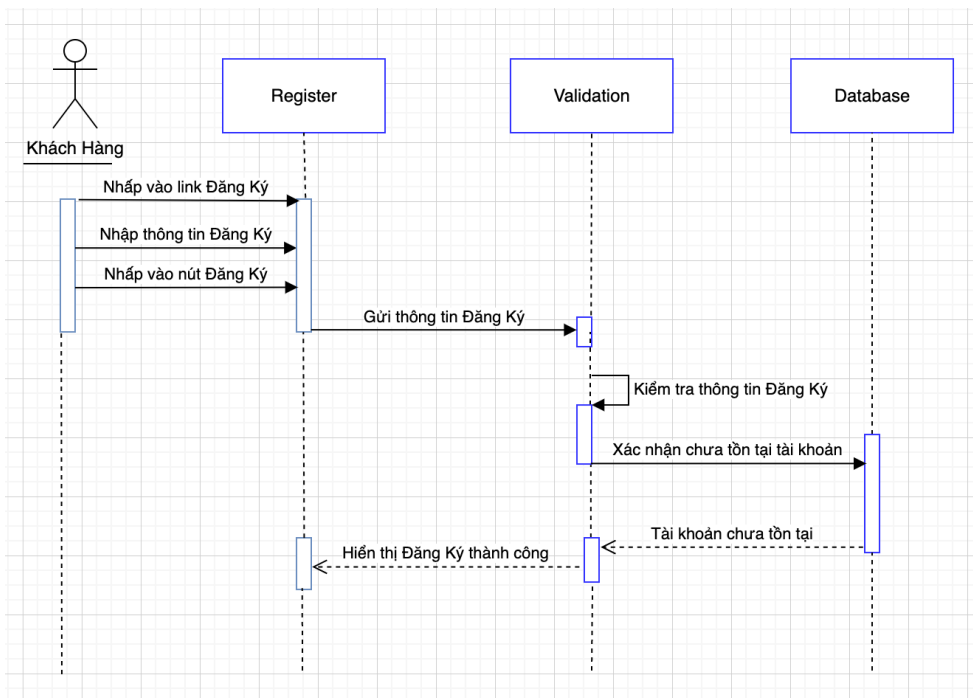


Hình 22. Biểu đồ Use Case – User

## 3.4.2. Biểu đồ tuần tự

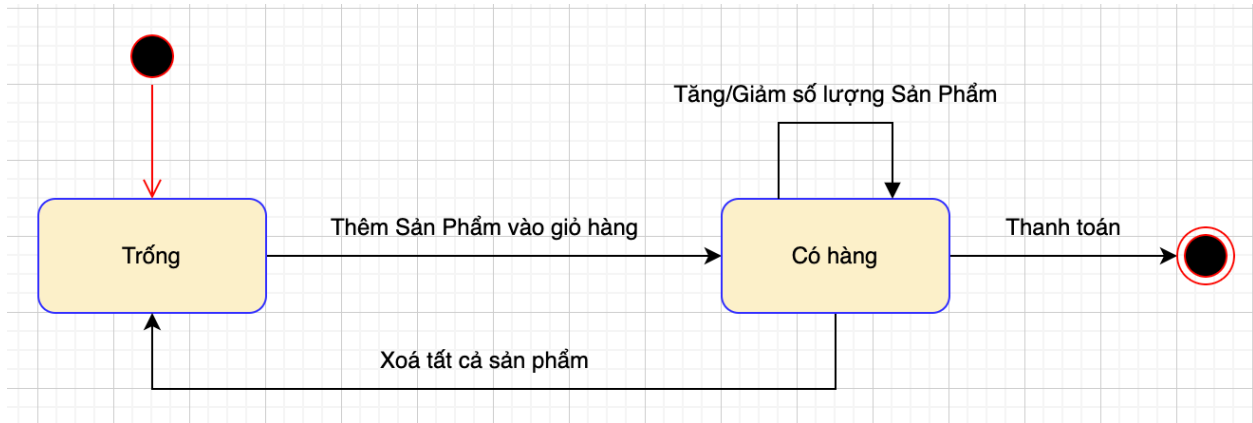


Hình 23. Biểu đồ tuần tự - Đăng Nhập

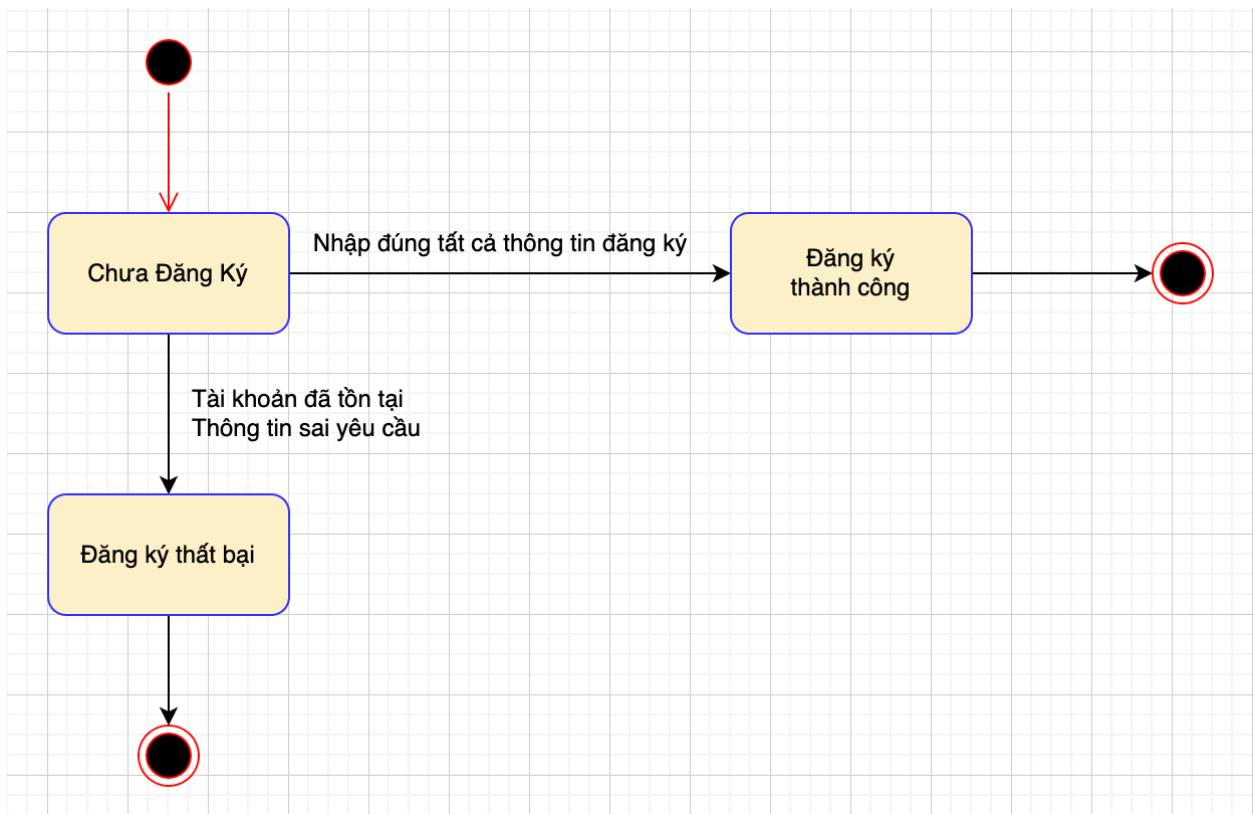


Hình 24. Biểu đồ tuần tự - Đăng Ký

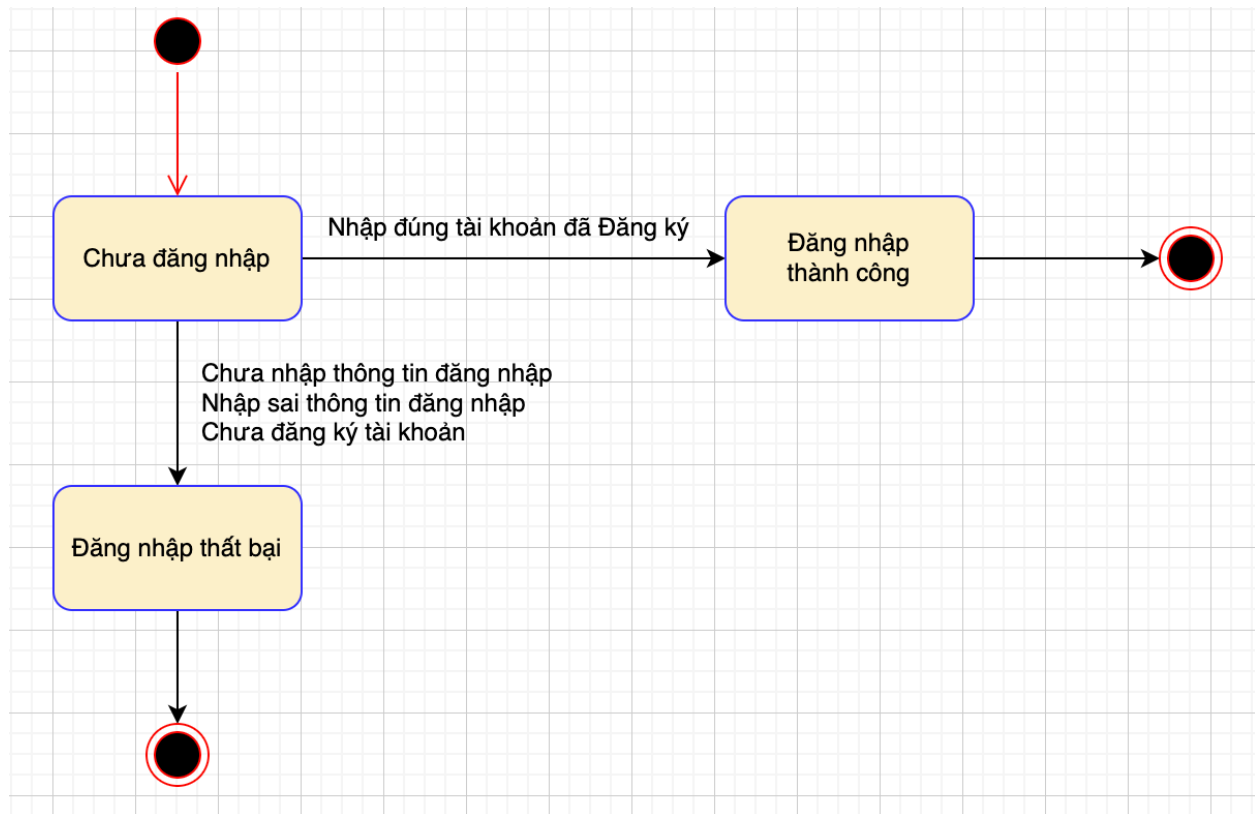
## 3.4.3. Biểu đồ trạng thái



Hình 25. Biểu đồ trạng thái – Giỏ hàng



Hình 26. Biểu đồ trạng thái – Đăng Ký



Hình 27. Biểu đồ trạng thái – Đăng Nhập

## TÀI LIỆU THAM KHẢO

- [1] “The C4 model for visualising software architecture”. Link: <https://www.c4model.com/>.
- [2] “Structurizr”. Link: <https://www.structurizr.com/>.