

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH
BỘ MÔN CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO BÀI TẬP LỚN
MÔN: PHÂN TÍCH THIẾT KẾ HƯỚNG ĐỐI TƯỢNG
ĐỀ TÀI:
XÂY DỰNG WEBSITE QUẢN TRỊ DỮ LIỆU THUÊ NHÀ**

Giảng viên hướng dẫn : ThS. PHẠM THỊ MIÊN
Sinh viên thực hiện : HOÀNG GIA KIẾT
Lớp : CÔNG NGHỆ THÔNG TIN
Khoá : 62

TP. Hồ Chí Minh, tháng 4 năm 2024

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH
BỘ MÔN CÔNG NGHỆ THÔNG TIN



BÁO CÁO BÀI TẬP LỚN
MÔN: PHÂN TÍCH THIẾT KẾ HƯỚNG ĐỐI TƯỢNG
ĐỀ TÀI:
XÂY DỰNG WEBSITE QUẢN TRỊ DỮ LIỆU THUÊ NHÀ

Giảng viên hướng dẫn : ThS. PHẠM THỊ MIÊN

Sinh viên thực hiện : HOÀNG GIA KIẾT
: NGUYỄN HƯƠNG DUYÊN

Lớp : CÔNG NGHỆ THÔNG TIN

Khoá : 62

TP. Hồ Chí Minh, tháng 4 năm 2024

LỜI CẢM ƠN

Lời đầu tiên nhóm chúng em xin gửi lời cảm ơn sâu sắc đến cô Phạm Thị Miên đã hỗ trợ và giúp đỡ nhóm chúng em hoàn thành môn học Phân tích thiết kế hướng đối tượng.

Chúng em cũng xin gửi lời cảm ơn đến quý thầy cô Bộ môn Công nghệ thông tin Trường Đại học Giao thông Vận tải Phân hiệu tại TP. Hồ Chí Minh đã truyền đạt kiến thức và giúp đỡ nhóm chúng em trong quá trình học tập để chúng em có thể hoàn thành tốt bài tập lớn của mình.

Ngoài ra, chúng em xin gửi lời cảm ơn đến các anh, chị, bạn đã đồng hành và giúp đỡ chúng em về tài liệu.

Trong quá trình thực hiện đề tài, chúng em còn gặp nhiều khó khăn, thiếu kiến thức chuyên môn nên báo cáo còn nhiều thiếu sót. Mong cô có thể xem xét, đánh giá cho bài tập lớn của nhóm chúng em được cập nhật và chỉnh sửa tốt nhất.

Chúng em xin chân thành cảm ơn cô./.

TP. Hồ Chí Minh, ngày ... tháng ... năm 2024

Sinh viên thực hiện

Hoàng Gia Kiệt

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

Giảng viên hướng dẫn

Hoàng Gia Kiệt – K62

MỤC LỤC

LỜI CẢM ƠN.....	i
NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN.....	ii
MỤC LỤC.....	iii
DANH MỤC CHỮ VIẾT TẮT	v
DANH MỤC BẢNG BIỂU	vi
DANH MỤC HÌNH ẢNH.....	vii
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT	1
1.1. Tổng quan về phân tích thiết kế hướng đối tượng	1
1.1.1. Khái niệm.....	1
1.2. Mô tả chu trình phát triển phần mềm	1
1.2.1. Chu trình phát triển phần mềm (Software Development Life Cycle)	1
1.2.2. Các giai đoạn của chu trình phát triển phần mềm	2
1.3. Giới thiệu về UML	4
1.3.1. Khái niệm.....	4
1.3.2. UML trong phân tích thiết kế phần mềm.....	4
1.3.3. Một số dạng biểu đồ UML phổ biến.....	5
CHƯƠNG 2: KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU	9
2.1. Khảo sát hiện trạng.....	9
2.1.1. Quy trình nghiệp vụ	9
2.1.2. Mô tả bài toán	9
2.2. Phân tích yêu cầu	9
2.2.1. Yêu cầu chức năng.....	9
2.2.2. Yêu cầu phi chức năng.....	10
CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	11
3.1. Các biểu đồ hệ thống	11
3.1.1. Biểu đồ thực thể.....	11
3.1.2. Biểu đồ lớp.....	11
3.1.3. Biểu đồ use case.....	12

3.1.4. Biểu đồ hoạt động	27
3.1.5. Biểu đồ tuần tự.....	39
CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH	46
4.1. Tổng quan các công cụ, công nghệ hỗ trợ xây dựng hệ thống. 46	
4.1.1. Phát triển Backend	46
4.1.2. Phát triển Frontend.....	47
4.2. Giao diện hệ thống	47
KẾT LUẬN.....	53
1. Kết quả đạt được.....	53
2. Hạn chế	53
3. Hướng phát triển.....	53
TÀI LIỆU THAM KHẢO.....	54

DANH MỤC CHỮ VIẾT TẮT

STT	VIẾT TẮT	Ý NGHĨA	DIỄN GIẢI
1	VSCode	Visual Studio Code	
2	OOAD	Object Oriented Analysis and Design	Phân tích và thiết kế hướng đối tượng
3	UML	Unified Modeling Language	
4	IDE	Integrated development environment	Môi trường phát triển tích hợp
5	RDBMS	Relational Database Management System	
6	XML	Extensible markup language	
7	JWT	JSON Web Token	

DANH MỤC BẢNG BIỂU

Bảng 3.1: Đặc tả use case thêm khách hàng	13
Bảng 3.2: Đặc tả use case sửa khách hàng	14
Bảng 3.3: Đặc tả use case import danh sách khách hàng	14
Bảng 3.4: Đặc tả use case export danh sách khách hàng.....	15
Bảng 3.5: Đặc tả use case xoá khách hàng	15
Bảng 3.6: Đặc tả use case thêm hợp đồng	17
Bảng 3.7: Đặc tả use case sửa hợp đồng.....	17
Bảng 3.8: Đặc tả use case import danh sách hợp đồng	18
Bảng 3.9: Đặc tả use case export danh sách hợp đồng	18
Bảng 3.10: Đặc tả use case xoá hợp đồng	18
Bảng 3.11: Đặc tả use case thêm căn hộ.....	20
Bảng 3.12: Đặc tả use case sửa căn hộ	20
Bảng 3.13: Đặc tả use case import danh sách căn hộ	21
Bảng 3.14: Đặc tả use case export danh sách căn hộ.....	21
Bảng 3.15: Đặc tả use case xoá căn hộ.....	21
Bảng 3.16: Đặc tả use case thêm người dùng.....	22
Bảng 3.17: Đặc tả use case sửa người dùng	23
Bảng 3.18: Đặc tả use case chặn người dùng	23
Bảng 3.19: Đặc tả use case thống kê	23
Bảng 3.20: Đặc tả use case đăng nhập.....	24
Bảng 3.21: Đặc tả use case đăng xuất.....	24
Bảng 3.22: Đặc tả use case đổi mật khẩu	25
Bảng 3.23: Đặc tả use case quên mật khẩu.....	25
Bảng 3.24: Đặc tả use case cập nhật thông tin	26

DANH MỤC HÌNH ẢNH

Hình 1.1: Các giai đoạn của Chu trình phát triển phần mềm	4
Hình 1.2: Ví dụ biểu đồ ca sử dụng	5
Hình 1.3: Ví dụ biểu đồ lớp	6
Hình 1.4: Ví dụ biểu đồ tuần tự	6
Hình 1.5: Ví dụ biểu đồ trạng thái	7
Hình 1.6: Ví dụ biểu đồ hoạt động.....	8
Hình 3.1: Biểu đồ lớp quản trị dữ liệu thuê nhà	11
Hình 3.2: Biểu đồ ERD quản trị dữ liệu thuê nhà	11
Hình 3.3: Biểu đồ use case tổng quát	12
Hình 3.4: Biểu đồ use case quản lý khách hàng	13
Hình 3.5: Biểu đồ use case quản lý hợp đồng	16
Hình 3.6: Biểu đồ use case quản lý căn hộ	19
Hình 3.7: Biểu đồ use case quản lý người dùng	22
Hình 3.8: Biểu đồ hoạt động đăng nhập	27
Hình 3.9: Biểu đồ hoạt động đăng xuất	28
Hình 3.10: Biểu đồ hoạt động đổi mật khẩu.....	29
Hình 3.11: Biểu đồ hoạt động quên mật khẩu	30
Hình 3.12: Biểu đồ hoạt động cập nhật thông tin.....	31
Hình 3.13: Biểu đồ hoạt động thêm khách hàng	32
Hình 3.14: Biểu đồ hoạt động cập nhật khách hàng.....	33
Hình 3.15: Biểu đồ hoạt động xem danh sách khách hàng.....	34
Hình 3.16: Biểu đồ hoạt động xóa khách hàng.....	35
Hình 3.17: Biểu đồ hoạt động import khách hàng.....	38
Hình 3.18: Biểu đồ hoạt động export khách hàng	38
Hình 3.19: Biểu đồ tuần tự đăng nhập.....	39
Hình 3.20: Biểu đồ tuần tự đăng xuất.....	39
Hình 3.21: Biểu đồ tuần tự đổi mật khẩu.....	40
Hình 3.22: Biểu đồ tuần tự quên mật khẩu.....	40
Hình 3.23: Biểu đồ tuần tự cập nhật thông tin.....	41

Hình 3.24: Biểu đồ tuần tự thêm khách hàng	42
Hình 3.25: Biểu đồ tuần tự cập nhật khách hàng.....	43
Hình 3.26: Biểu đồ tuần tự xem danh sách khách hàng	43
Hình 3.27: Biểu đồ tuần tự xóa khách hàng	44
Hình 3.28: Biểu đồ tuần tự import khách hàng	44
Hình 3.29: Biểu đồ tuần tự export khách hàng	45
Hình 4.1: Giao diện đăng nhập	47
Hình 4.2: Giao diện quên mật khẩu	48
Hình 4.3: Giao diện gửi mật khẩu mới qua email	48
Hình 4.4: Giao diện trang chủ.....	49
Hình 4.5: Giao diện quản lý khách hàng	49
Hình 4.6: Giao diện quản lý căn hộ	50
Hình 4.7: Giao diện quản lý hợp đồng.....	50
Hình 4.8: Giao diện quản lý người dùng	51
Hình 4.9: Giao diện tài khoản.....	51
Hình 4.10: Giao diện sau khi đổi mail mới.....	52
Hình 4.11: Giao diện đổi mật khẩu.....	52

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

1.1. Tổng quan về phân tích thiết kế hướng đối tượng

1.1.1. Khái niệm

Trong kỹ nghệ phần mềm để sản xuất được một sản phẩm phần mềm người ta chia quá trình phát triển sản phẩm ra nhiều giai đoạn như thu thập và phân tích yêu cầu, phân tích và thiết kế hệ thống, phát triển (coding), kiểm thử, triển khai và bảo trì. Trong đó, giai đoạn phân tích, thiết kế bao giờ cũng là giai đoạn khó khăn và phức tạp nhất. Giai đoạn này giúp chúng ta hiểu rõ yêu cầu đặt ra, xác định giải pháp, mô tả chi tiết giải pháp. Nó trả lời 2 câu hỏi What (phần mềm này làm cái gì?) và How (làm nó như thế nào?).

Để phân tích và thiết kế một phần mềm thì có nhiều cách làm, một trong những cách làm đó là xem hệ thống gồm những đối tượng sống trong đó và tương tác với nhau. Việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng ta hiểu rõ hệ thống và cài đặt được nó. Phương thức này gọi là Phân tích thiết kế hướng đối tượng (OOAD) [1]

1.2. Mô tả chu trình phát triển phần mềm

1.2.1. Chu trình phát triển phần mềm (Software Development Life Cycle)

Chu Trình Phát Triển Phần Mềm là một chuỗi các hoạt động của nhà phân tích (Analyst), nhà thiết kế (Designer), người phát triển (Developer) và người dùng (User) để phát triển và thực hiện một hệ thống thông tin. Những hoạt động này được thực hiện trong nhiều giai đoạn khác nhau.

- **Nhà phân tích (Analyst):** là người nghiên cứu yêu cầu của khách hàng/người dùng để định nghĩa một phạm vi bài toán, nhận dạng nhu cầu của một tổ chức, xác định xem nhân lực, phương pháp và công nghệ máy tính có thể làm sao để cải thiện một cách tốt nhất công tác của tổ chức này.

- **Nhà thiết kế (Designer):** thiết kế hệ thống theo hướng cấu trúc của database, screens, forms và reports – quyết định các yêu cầu về phần cứng và phần mềm cho hệ thống cần được phát triển.

- **Chuyên gia lĩnh vực (Domain Experts):** là những người hiểu thực chất vấn đề cùng tất cả những sự phức tạp của hệ thống cần tin học hoá. Họ không nhất thiết phải là nhà lập trình, nhưng họ có thể giúp nhà lập trình hiểu yêu cầu đặt ra đối với hệ thống cần phát triển. Quá trình phát triển phần mềm sẽ có rất nhiều thuận lợi nếu đội ngũ làm phần mềm có được sự trợ giúp của họ.

- **Lập trình viên (Programmer):** là những người dựa trên các phân tích và thiết kế để viết chương trình (coding) cho hệ thống bằng ngôn ngữ lập trình đã được thống nhất.

- **Người dùng (User):** là đối tượng phục vụ của hệ thống cần được phát triển.

1.2.2. Các giai đoạn của chu trình phát triển phần mềm

Chu trình của một phần mềm có thể được chia thành các giai đoạn như sau:

- Nghiên cứu sơ bộ (Preliminary Investigation hay còn gọi là Feasibility Study)

- Phân tích yêu cầu (Requirements Analysis)

- Thiết kế hệ thống (Design of the System)

- Xây dựng phần mềm (Software Construction)

- Thử nghiệm hệ thống (System Testing)

- Thực hiện, triển khai (System Implementation)

- Bảo trì, nâng cấp (System Maintenance)

1.2.2.1. Nghiên cứu sơ bộ

UML đưa ra khái niệm Use Case để nắm bắt các yêu cầu của khách hàng (người sử dụng). UML sử dụng biểu đồ Use case (Use Case Diagram) để nêu bật mối quan hệ cũng như sự giao tiếp với hệ thống. Qua phương pháp mô hình hóa Use case, các tác nhân (Actor) bên ngoài quan tâm đến hệ thống sẽ được mô hình hóa song song với chức năng mà họ đòi hỏi từ phía hệ thống (tức là Use case). Các tác nhân và các Use case được mô hình hóa cùng các mối quan hệ và được miêu tả trong biểu đồ Use case của UML. Mỗi một Use case được mô tả trong tài liệu, và nó sẽ đặc tả các yêu cầu của khách hàng: Anh ta hay chị ta chờ đợi điều gì ở phía hệ thống mà không hề để ý đến việc chức năng này sẽ được thực thi ra sao.

1.2.2.2. Phân tích yêu cầu

Giai đoạn phân tích quan tâm đến quá trình trừu tượng hóa đầu tiên (các lớp và các đối tượng) cũng như cơ chế hiện hữu trong phạm vi vấn đề. Sau khi nhà phân tích đã nhận biết được các lớp thành phần của mô hình cũng như mối quan hệ giữa chúng với nhau, các lớp cùng các mối quan hệ đó sẽ được miêu tả bằng công cụ biểu đồ lớp (class diagram) của UML. Trong giai đoạn phân tích, chỉ duy nhất các lớp có tồn tại trong phạm vi vấn đề (các khái niệm đời thực) là được mô hình hóa.

1.2.2.3. Thiết kế hệ thống

Một số các công việc thường được thực hiện trong giai đoạn thiết kế:

- Nhận biết form nhập liệu tùy theo các thành phần dữ liệu cần nhập.

- Nhận biết reports và những output mà hệ thống mới phải sản sinh

- Thiết kế forms (vẽ trên giấy hay máy tính, sử dụng công cụ thiết kế)

- Nhận biết các thành phần dữ liệu và bảng để tạo database
- Ước tính các thủ tục giải thích quá trình xử lý từ input đến output.

Kết quả giai đoạn thiết kế là Đặc Tả Thiết Kế (Design Specifications). Bản Đặc Tả Thiết Kế Chi Tiết sẽ được chuyển sang cho các lập trình viên để thực hiện giai đoạn xây dựng phần mềm.

1.2.2.4. Xây dựng hệ thống

Đây là giai đoạn viết lệnh (code) thực sự, tạo hệ thống. Cũng chính người viết code chịu trách nhiệm viết tài liệu liên quan đến chương trình, giải thích thủ tục (procedure).

Để đảm bảo chương trình được viết nên phải thoả mãn mọi yêu cầu có ghi trước trong bản Đặc Tả Thiết Kế Chi Tiết, người viết code cũng đồng thời phải tiến hành thử nghiệm phần chương trình của mình.

1.2.2.5. Thử nghiệm hệ thống

Một hệ thống phần mềm thường được thử nghiệm qua nhiều giai đoạn và với nhiều nhóm thử nghiệm khác nhau. Các nhóm sử dụng nhiều loại biểu đồ UML khác nhau làm nền tảng cho công việc của mình: Thử nghiệm đơn vị sử dụng biểu đồ lớp (class diagram) và đặc tả lớp, thử nghiệm tích hợp thường sử dụng biểu đồ thành phần (component diagram) và biểu đồ cộng tác (collaboration diagram), và giai đoạn thử nghiệm hệ thống sử dụng biểu đồ Use case (use case diagram) để đảm bảo hệ thống có phương thức hoạt động đúng như đã được định nghĩa từ ban đầu trong các biểu đồ này.

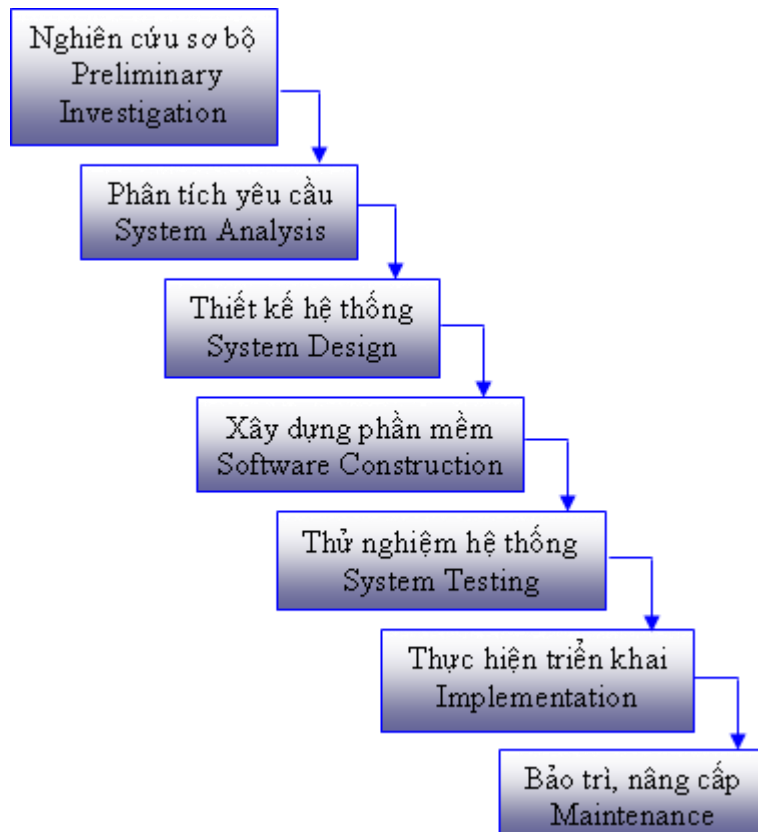
1.2.2.6. Thực hiện, triển khai

Trong giai đoạn này, hệ thống vừa phát triển sẽ được triển khai sao cho phía người dùng. Trước khi để người dùng thật sự bắt tay vào sử dụng hệ thống, nhóm các nhà phát triển cần tạo các file dữ liệu cần thiết cũng như huấn luyện cho người dùng, để đảm bảo hệ thống được sử dụng hữu hiệu nhất.

1.2.2.7. Bảo trì, nâng cấp

Tùy theo các biến đổi trong môi trường sử dụng, hệ thống có thể trở nên lỗi thời hay cần phải được sửa đổi nâng cấp để sử dụng có hiệu quả. Hoạt động bảo trì hệ thống có thể rất khác biệt tùy theo mức độ sửa đổi và nâng cấp cần thiết.

Sơ đồ tổng quát các giai đoạn của Chu Trình Phát Triển Phần Mềm:



Hình 1.1: Các giai đoạn của Chu trình phát triển phần mềm

1.3. Giới thiệu về UML

1.3.1. Khái niệm

UML là ngôn ngữ mô hình hóa hợp nhất dùng để biểu diễn hệ thống. Nói một cách đơn giản là nó dùng để tạo ra các bản vẽ nhằm mô tả thiết kế hệ thống. Các bản vẽ này được sử dụng để các nhóm thiết kế trao đổi với nhau cũng như dùng để thi công hệ thống (phát triển), thuyết phục khách hàng, các nhà đầu tư v.v.. (Giống như trong xây dựng người ta dùng các bản vẽ thiết kế để hướng dẫn và kiểm soát thi công, bán hàng căn hộ v.v..)

1.3.2. UML trong phân tích thiết kế phần mềm

Vì mục đích chính của ngôn ngữ này là dùng các biểu đồ hướng đối tượng để mô tả hệ thống nên miền ứng dụng của UML bao gồm nhiều loại hệ thống khác nhau như:

- **Hệ thống thông tin:** Cát giữ, lấy, biến đổi biểu diễn thông tin cho người sử dụng. Xử lý những khoảng dữ liệu lớn có các quan hệ phức tạp, mà chúng được lưu trữ trong các cơ sở dữ liệu quan hệ hay hướng đối tượng.

- **Hệ thống kỹ thuật:** Xử lý và điều khiển các thiết bị kỹ thuật như viễn thông, hệ thống quân sự, hay các quá trình công nghiệp. Đây là loại thiết bị phải xử lý các giao tiếp đặc biệt, không có phần mềm chuẩn và thường là các hệ thống thời gian thực.

- **Hệ thống nhúng:** Thực hiện trên phần cứng gắn vào các thiết bị như điện thoại di động, điều khiển xe hơi, ... Điều này được thực hiện bằng việc lập trình mức thấp với hỗ trợ thời gian thực. Những hệ thống này thường không có các thiết bị như màn hình đĩa cứng, ...

- **Hệ thống phân bố:** Được phân bố trên một số máy cho phép truyền dữ liệu từ nơi này đến nơi khác một cách dễ dàng. Chúng đòi hỏi các cơ chế liên lạc đồng bộ để đảm bảo toàn vẹn dữ liệu và thường được xây dựng trên một số các kỹ thuật đối tượng như CORBA, COM/DCOM, hay Java Beans/RMI.

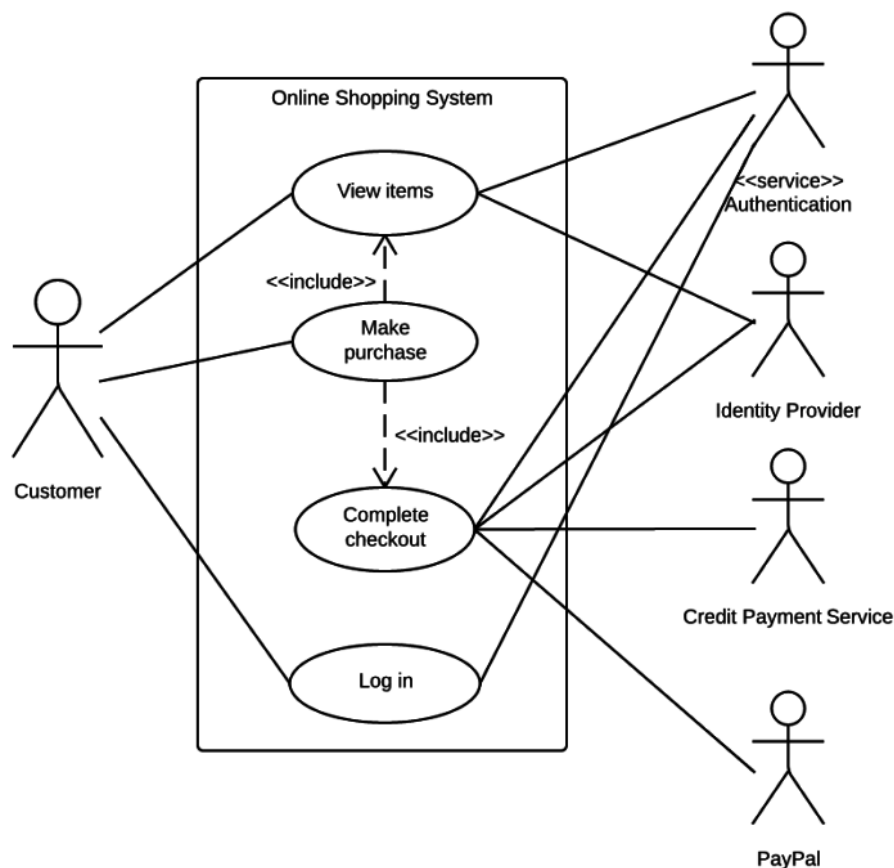
- **Hệ thống Giao dịch:** Mô tả mục đích, tài nguyên (con người, máy tính, ...), các quy tắc (luật pháp, chiến thuật kinh doanh, cơ chế, ...), và công việc hoạt động kinh doanh.

- **Phần mềm hệ thống:** Định nghĩa cơ sở hạ tầng kỹ thuật cho phần mềm khác sử dụng, chẳng hạn như hệ điều hành, cơ sở dữ liệu, giao diện người sử dụng.

1.3.3. Một số dạng biểu đồ UML phổ biến

1.3.3.1. Biểu đồ ca sử dụng (Use case diagram)

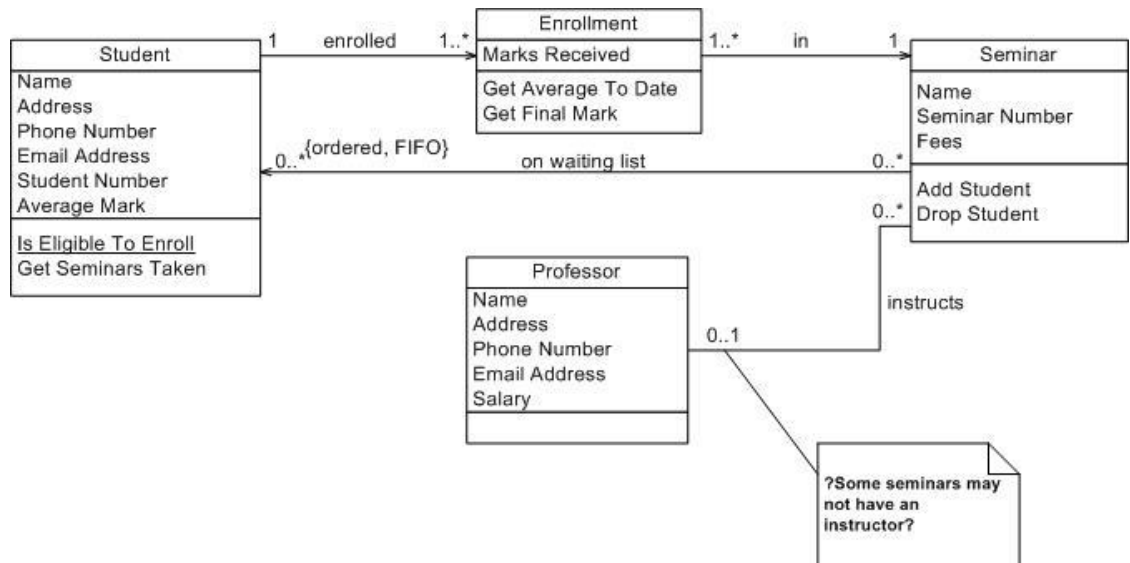
Biểu đồ ca sử dụng mô hình hóa chức năng của một hệ thống sử dụng các tác nhân và ca sử dụng. [2]



Hình 1.2: Ví dụ biểu đồ ca sử dụng

1.3.3.2. Biểu đồ lớp (Class diagram)

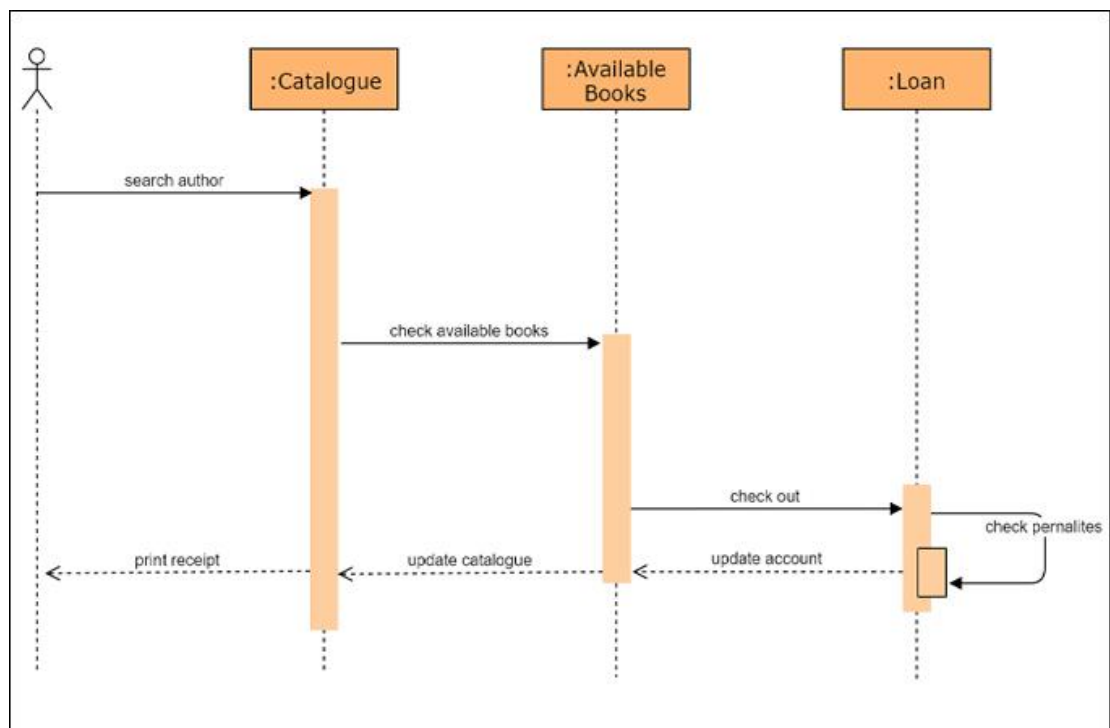
Biểu đồ lớp là xương sống của hầu hết mọi phương pháp hướng đối tượng, bao gồm cả UML. Chúng mô tả cấu trúc tĩnh của một hệ thống.



Hình 1.3: Ví dụ biểu đồ lớp

1.3.3.3. Biểu đồ tuần tự (Sequence diagram)

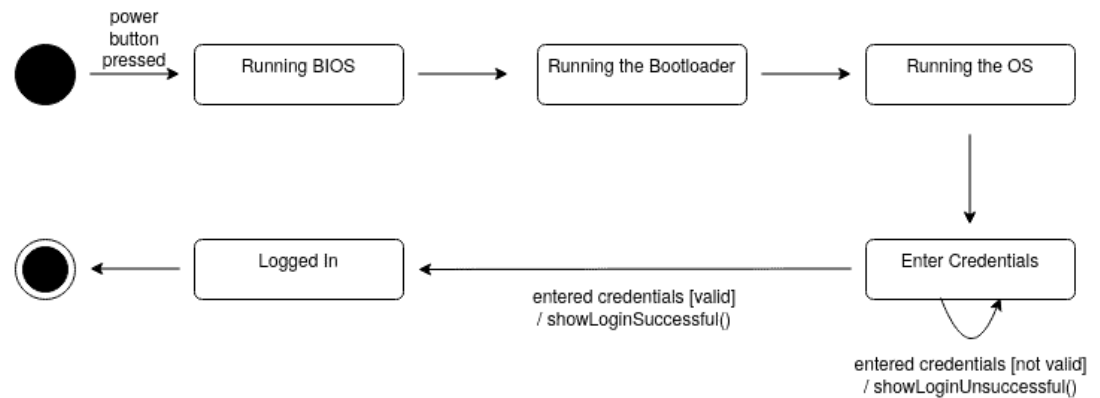
Biểu đồ trình tự mô tả sự tương tác giữa các lớp về mặt trao đổi thông điệp theo thời gian.



Hình 1.4: Ví dụ biểu đồ tuần tự

1.3.3.4. Biểu đồ trạng thái (State diagram)

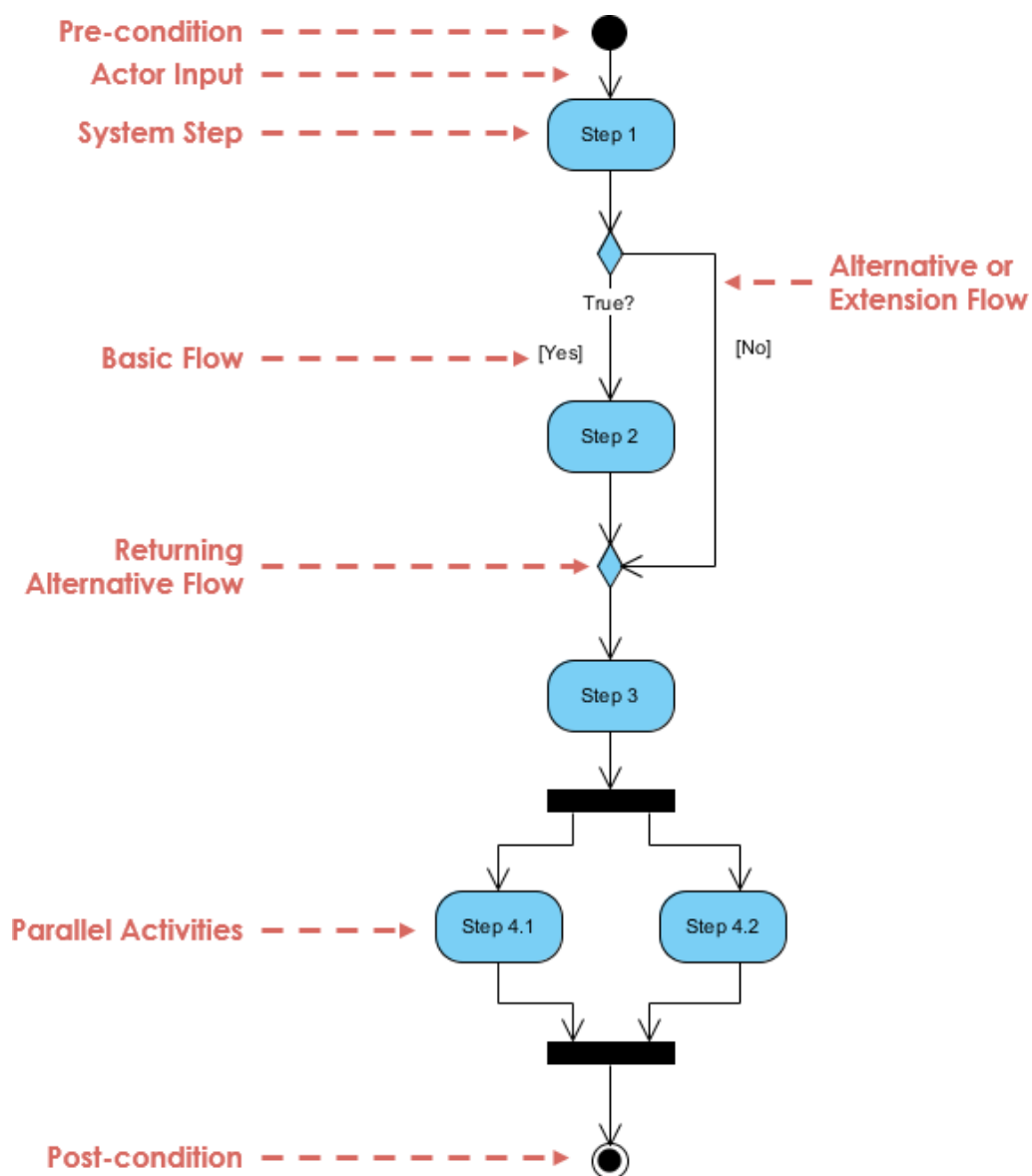
Biểu đồ trạng thái, ngày nay được gọi là biểu đồ trạng thái máy và biểu đồ trạng thái mô tả hành vi động của một hệ thống để đáp ứng với các kích thích bên ngoài. Biểu đồ trạng thái đặc biệt hữu ích trong việc mô hình hóa các đối tượng phản ứng có trạng thái được kích hoạt bởi các sự kiện cụ thể.



Hình 1.5: Ví dụ biểu đồ trạng thái

1.3.3.5. Biểu đồ hoạt động (Activity diagram)

Biểu đồ hoạt động minh họa bản chất động của một hệ thống bằng cách mô hình hóa luồng kiểm soát từ hoạt động này sang hoạt động khác. Một hoạt động đại diện cho một hoạt động trên một số lớp trong hệ thống dẫn đến sự thay đổi trạng thái của hệ thống. Thông thường, sơ đồ hoạt động được sử dụng để mô hình hóa quy trình làm việc hoặc quy trình kinh doanh và hoạt động nội bộ.



Hình 1.6: Ví dụ biểu đồ hoạt động

CHƯƠNG 2: KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU

2.1. Khảo sát hiện trạng

2.1.1. Quy trình nghiệp vụ

- Nhân viên kinh doanh trong phòng ban sẽ tìm khách hàng, căn hộ và hợp đồng cho công ty. Sau đó cập nhật thông tin vào file máy tính cá nhân

- Một nhân viên có thể tạo nhiều hợp đồng, mỗi hợp đồng thì sẽ có thông tin của khách hàng, căn hộ và nhân viên tạo ra hợp đồng đó.

- Cuối tuần sẽ gửi lại file cho người quản lý để cập nhật vào file của máy tính công ty. Có những trường hợp thông tin bị trùng thì rất tốn thời gian để xử lý.

- Khi có chỉnh sửa thì cập nhật thông tin với người quản lý.

- Hầu hết các nghiệp vụ quản lý đều được thực hiện theo cách thủ công, gây chậm trễ, phiền toái.

2.1.2. Mô tả bài toán

Một công ty cần chuyển đổi số vào việc quản lý dữ liệu khách hàng, căn hộ, hợp đồng cho phòng ban kinh doanh. Việc sử dụng hệ thống chung sẽ làm tăng khả năng quản lý, nhanh chóng và tránh trùng lặp dữ liệu. Nhân viên sẽ nhập thông tin vào hệ thống quản lý. Thông tin về khách hàng: Mã, Họ tên, Địa chỉ, Số CCCD, Ngày sinh, Số điện thoại. Thông tin về văn hộ: Mã, Địa chỉ, Giá tiền, Số phòng. Thông tin về hợp đồng: Mã, Ngày bắt đầu, Ngày kết thúc, Tổng tiền, Tiền thuê 1 tháng, Ngày tạo, Mã khách hàng, Mã căn hộ, Mã nhân viên tạo. Khi có yêu cầu tạo hợp đồng thì nhân viên sử dụng chương trình cần nhập thông tin khách hàng (nếu chưa có).

Ngoài ra chương trình cần có thêm các chức năng như:

- Import/Export danh sách thông tin của khách hàng, căn hộ, hợp đồng
- Tra cứu thông tin hợp đồng
- Thống kê

2.2. Phân tích yêu cầu

2.2.1. Yêu cầu chức năng

- Cho phép người dùng đăng nhập vào hệ thống
- Cho phép người dùng cập nhật thông tin cá nhân
- Cho phép người quản lý xem thống kê của hệ thống
- Phân quyền cho nhân viên và quản lý:

+ Quản lý, nhân viên: Thêm, xem và chỉnh sửa dữ liệu

- + Quản lý: Xoá dữ liệu

- Cho phép người quản lý, nhân viên nhập và xuất dữ liệu khách hàng, căn hộ, hợp đồng

2.2.2. Yêu cầu phi chức năng

Yêu cầu về chất lượng

- Tính cập nhật – phát triển:

- + Dễ dàng nâng cấp hoặc thêm các tính năng tiện ích

- + Bảo trì hệ thống

- Tính tiện dụng:

- + Có các đặc tả và hướng dẫn rõ ràng, dễ dàng thao tác

- + Hoạt động ổn định, đáng tin cậy

- Tính tương thích:

- + Hoạt động tốt trên tất cả các phương tiện phổ biến hiện nay

Yêu cầu về giao diện

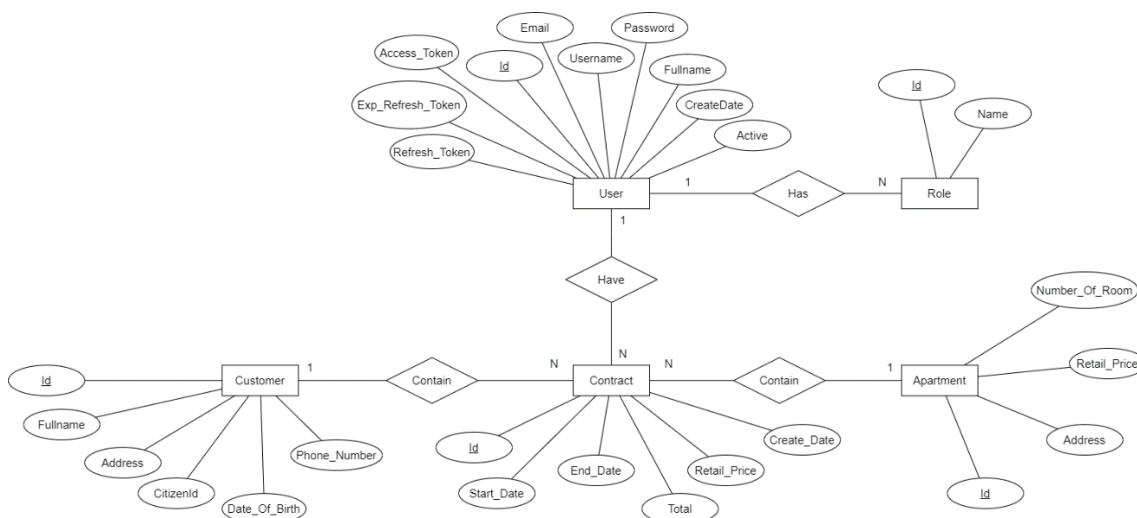
- + Thân thiện, dễ sử dụng

- + Hiện thị đầy đủ và chi tiết các thông tin, tính nhất quán

CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

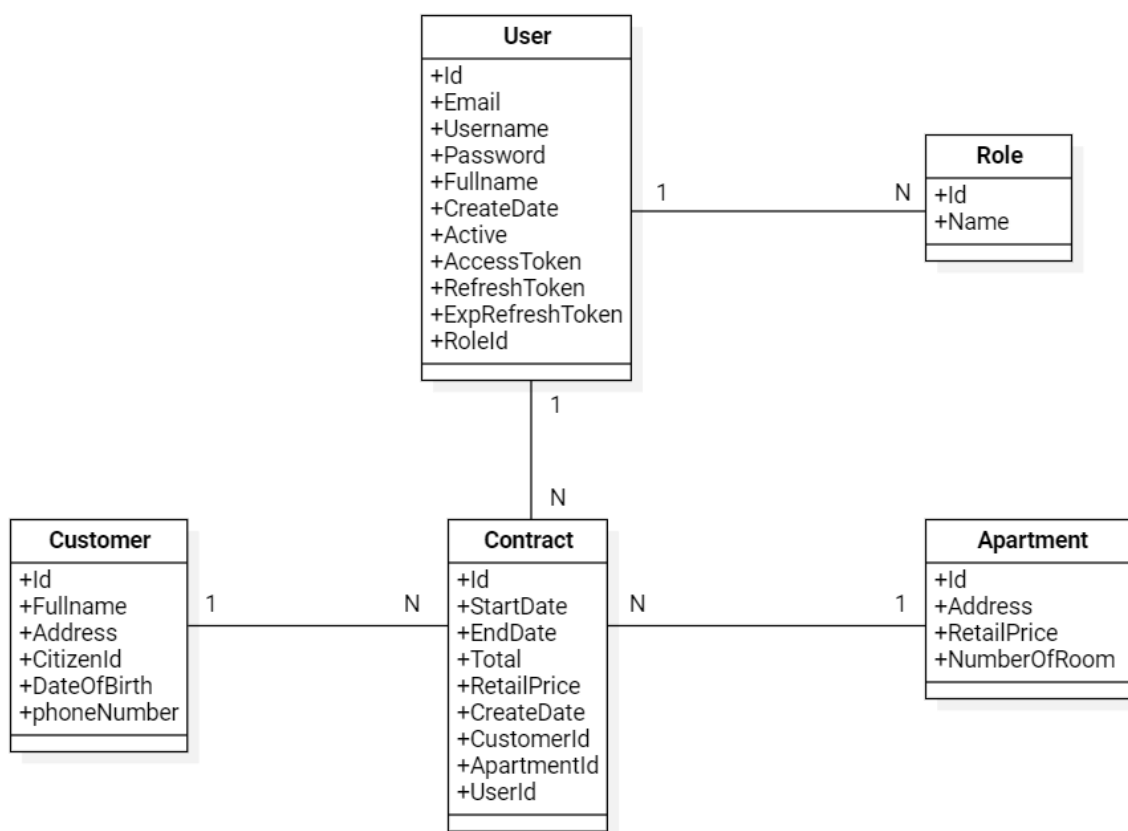
3.1. Các biểu đồ hệ thống

3.1.1. Biểu đồ thực thể



Hình 3.1: Biểu đồ lớp quản trị dữ liệu thuê nhà

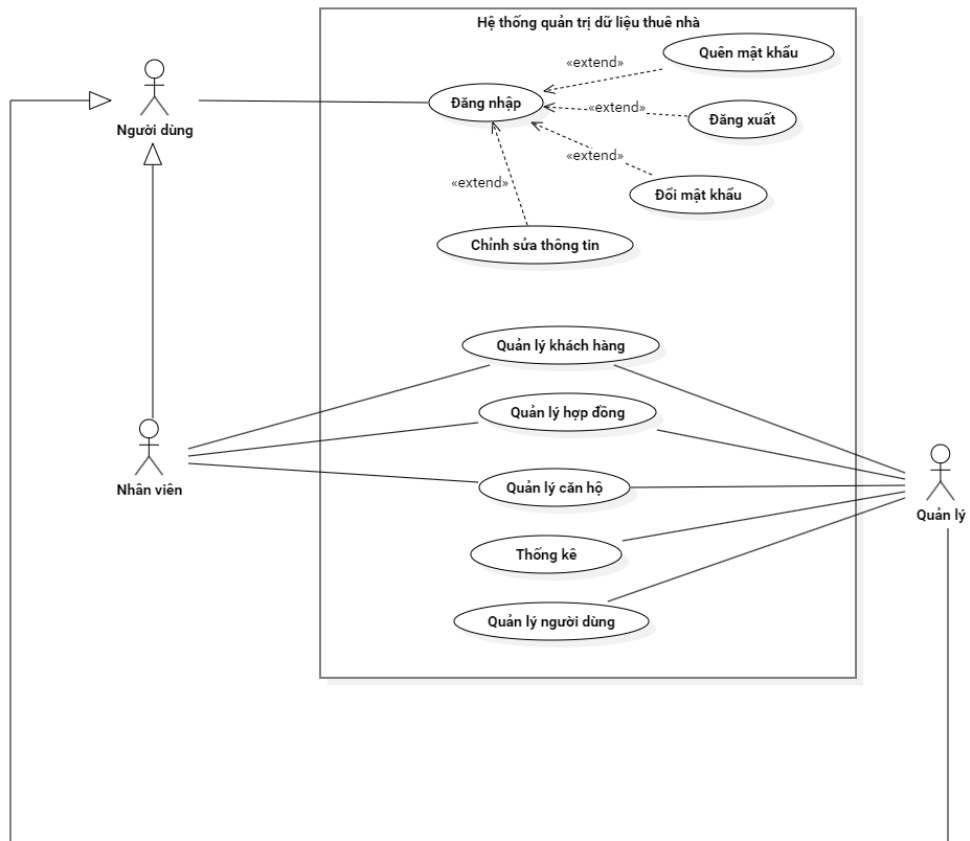
3.1.2. Biểu đồ lớp



Hình 3.2: Biểu đồ ERD quản trị dữ liệu thuê nhà

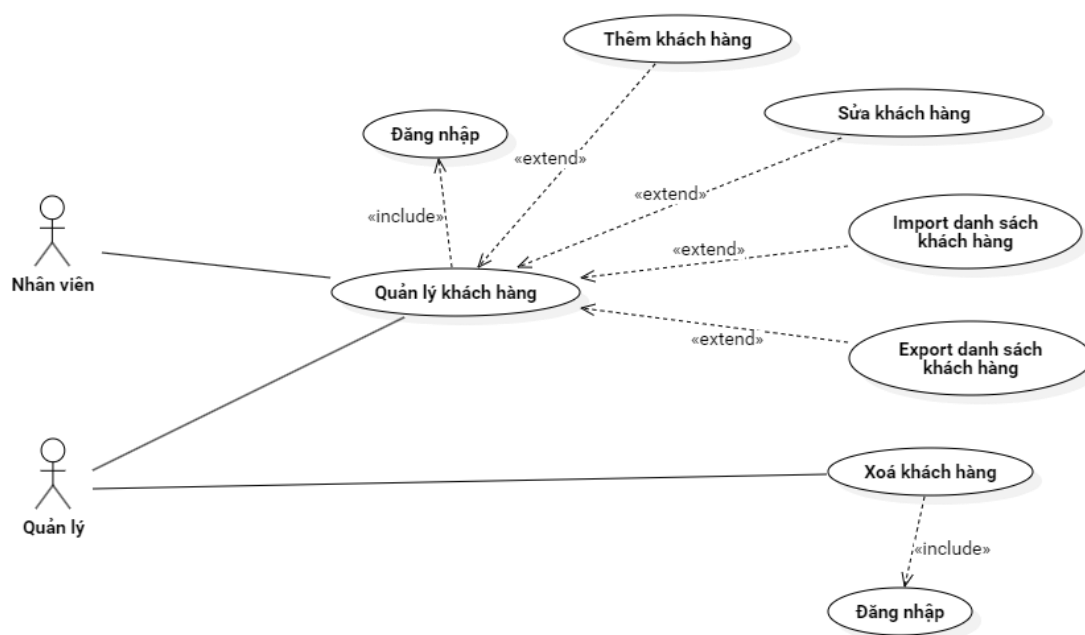
3.1.3. Biểu đồ use case

3.1.3.1. Use case tổng quát hệ thống



Hình 3.3: Biểu đồ use case tổng quát

3.1.3.2. Chức năng quản lý khách hàng



Hình 3.4: Biểu đồ use case quản lý khách hàng

- Đặc tả thêm khách hàng

Tên use case:	Thêm khách hàng
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> Người dùng phải đăng nhập Đang ở trang Quản lý khách hàng
Hậu điều kiện:	<ul style="list-style-type: none"> Thêm thành công khách hàng mới
Luồng sự kiện chính:	<ol style="list-style-type: none"> Người dùng chọn nút “Thêm” Hệ thống hiển thị form thêm khách hàng Người dùng nhập thông tin bắt buộc Người dùng nhấn nút “Lưu” Hệ thống kiểm tra thông tin Hệ thống cho phép thêm khách hàng Hệ thống lưu khách hàng
Luồng sự kiện phụ:	<p>A. Thông tin không đúng yêu cầu</p> <ol style="list-style-type: none"> Hệ thống báo lỗi Người dùng nhập lại thông tin không đúng yêu cầu

Bảng 3.1: Đặc tả use case thêm khách hàng

- Đặc tả sửa khách hàng

Tên use case:	Sửa khách hàng
Tác nhân:	Quản lý, nhân viên

Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng phải đăng nhập • Đang ở trang Quản lý khách hàng
Hậu điều kiện:	<ul style="list-style-type: none"> • Sửa thành công khách hàng
Luồng sự kiện chính:	<ol style="list-style-type: none"> 1. Người dùng chọn khách hàng mong muốn chỉnh sửa 2. Người dùng chọn nút “Sửa” 3. Hệ thống hiển thị form sửa khách hàng 4. Người dùng nhập thông tin chỉnh sửa 5. Người dùng nhấn nút “Lưu” 6. Hệ thống kiểm tra thông tin 7. Hệ thống cho phép cập nhật khách hàng 8. Hệ thống cập nhật khách hàng
Luồng sự kiện phụ:	<p>A. Thông tin không đúng yêu cầu</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi 2. Người dùng nhập lại thông tin không đúng yêu cầu

Bảng 3.2: Đặc tả use case sửa khách hàng

- Đặc tả import danh sách khách hàng

Tên use case:	Import danh sách khách hàng
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng phải đăng nhập • Đang ở trang Quản lý khách hàng
Hậu điều kiện:	<ul style="list-style-type: none"> • Thêm thành công danh sách khách hàng
Luồng sự kiện chính:	<ol style="list-style-type: none"> 1. Người dùng chọn nút “Import” 2. Hệ thống hiển thị form import 3. Người dùng chọn file cần import 4. Người dùng nhấn nút “Gửi” 5. Hệ thống kiểm tra file được gửi lên 6. Hệ thống cho phép import 7. Hệ thống lưu danh sách khách hàng 8. Hệ thống thông báo kết quả import 9. Hệ thống import thành công
Luồng sự kiện phụ:	<p>A. File không đúng yêu cầu</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi 2. Người dùng chọn file lại <p>B. Nội dung danh sách nhập không đúng yêu cầu</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi 2. Người dùng chỉnh sửa lại nội dung

Bảng 3.3: Đặc tả use case import danh sách khách hàng

- Đặc tả export danh sách khách hàng

Tên use case:	Export danh sách khách hàng
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng phải đăng nhập • Đang ở trang Quản lý khách hàng
Hậu điều kiện:	<ul style="list-style-type: none"> • Tải thành công danh sách khách hàng
Luồng sự kiện chính:	<ol style="list-style-type: none"> 1. Người dùng chọn nút “Export” 2. Hệ thống xử lý file 3. Hệ thống gửi file về cho trình duyệt 4. Trình duyệt tự động tải file 5. Hệ thống export thành công
Luồng sự kiện phụ:	Không có

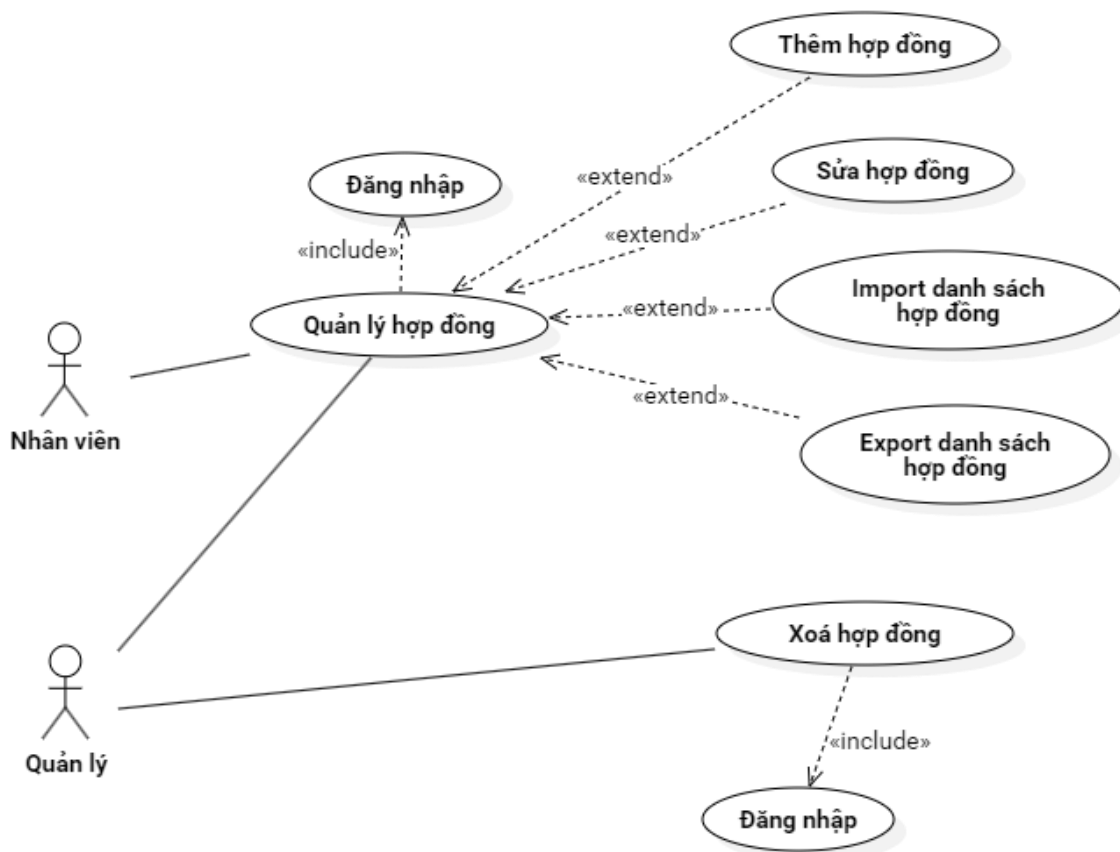
Bảng 3.4: Đặc tả use case export danh sách khách hàng

- Đặc tả xoá khách hàng

Tên use case:	Xoá khách hàng
Tác nhân:	Quản lý
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng phải đăng nhập • Đang ở trang Quản lý khách hàng
Hậu điều kiện:	<ul style="list-style-type: none"> • Xoá thành công khách hàng
Luồng sự kiện chính:	<ol style="list-style-type: none"> 1. Người dùng chọn khách hàng mong muốn xoá 2. Người dùng chọn nút “Xoá” 3. Hệ thống hỏi xác nhận xoá 4. Người dùng xác nhận xoá 5. Hệ thống xoá thành công khách hàng
Luồng sự kiện phụ:	<p>A. Người dùng xác nhận không xoá</p> <ol style="list-style-type: none"> 1. Hệ thống hiển thị trang Quản lý khách hàng

Bảng 3.5: Đặc tả use case xoá khách hàng

3.1.3.3. Chức năng quản lý hợp đồng



Hình 3.5: Biểu đồ use case quản lý hợp đồng

- Đặc tả thêm hợp đồng

Tên use case:	Thêm hợp đồng
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> Người dùng phải đăng nhập Đang ở trang Quản lý hợp đồng
Hậu điều kiện:	<ul style="list-style-type: none"> Thêm thành công hợp đồng mới
Luồng sự kiện chính:	<ol style="list-style-type: none"> Người dùng chọn nút “Thêm” Hệ thống hiển thị form thêm hợp đồng Người dùng nhập thông tin bắt buộc Người dùng nhấn nút “Lưu” Hệ thống kiểm tra thông tin Hệ thống cho phép thêm hợp đồng Hệ thống lưu hợp đồng
Luồng sự kiện phụ:	<p>A. Thông tin không đúng yêu cầu</p> <ol style="list-style-type: none"> Hệ thống báo lỗi Người dùng nhập lại thông tin không đúng yêu cầu

Bảng 3.6: Đặc tả use case thêm hợp đồng

- Đặc tả sửa hợp đồng

Tên use case:	Sửa hợp đồng
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng phải đăng nhập • Đang ở trang Quản lý hợp đồng
Hậu điều kiện:	<ul style="list-style-type: none"> • Thêm thành công hợp đồng
Luồng sự kiện chính:	<ol style="list-style-type: none"> 1. Người dùng chọn hợp đồng mong muốn chỉnh sửa 2. Người dùng chọn nút “Sửa” 3. Hệ thống hiển thị form sửa hợp đồng 4. Người dùng nhập thông tin chỉnh sửa 5. Người dùng nhấn nút “Lưu” 6. Hệ thống kiểm tra thông tin 7. Hệ thống cho phép cập nhật hợp đồng 8. Hệ thống cập nhật hợp đồng
Luồng sự kiện phụ:	<p>A. Thông tin không đúng yêu cầu</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi 2. Người dùng nhập lại thông tin không đúng yêu cầu

Bảng 3.7: Đặc tả use case sửa hợp đồng

- Đặc tả import danh sách hợp đồng

Tên use case:	Import danh sách hợp đồng
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng phải đăng nhập • Đang ở trang Quản lý hợp đồng
Hậu điều kiện:	<ul style="list-style-type: none"> • Thêm thành công danh sách hợp đồng
Luồng sự kiện chính:	<ol style="list-style-type: none"> 1. Người dùng chọn nút “Import” 2. Hệ thống hiển thị form import 3. Người dùng chọn file cần import 4. Người dùng nhấn nút “Gửi” 5. Hệ thống kiểm tra file được gửi lên 6. Hệ thống cho phép import 7. Hệ thống lưu danh sách hợp đồng 8. Hệ thống thông báo kết quả import 9. Hệ thống import thành công

Luồng sự kiện phụ:	A. File không đúng yêu cầu 1. Hệ thống báo lỗi 2. Người dùng chọn file lại B. Nội dung danh sách nhập không đúng yêu cầu 1. Hệ thống báo lỗi 2. Người dùng chỉnh sửa lại nội dung
---------------------------	--

Bảng 3.8: Đặc tả use case import danh sách hợp đồng

- Đặc tả export danh sách hợp đồng

Tên use case:	Export danh sách hợp đồng
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng phải đăng nhập • Đang ở trang Quản lý hợp đồng
Hậu điều kiện:	<ul style="list-style-type: none"> • Tải thành công danh sách hợp đồng
Luồng sự kiện chính:	1. Người dùng chọn nút “Export” 2. Hệ thống xử lý file 3. Hệ thống gửi file về cho trình duyệt 4. Trình duyệt tự động tải file 5. Hệ thống export thành công
Luồng sự kiện phụ:	Không có

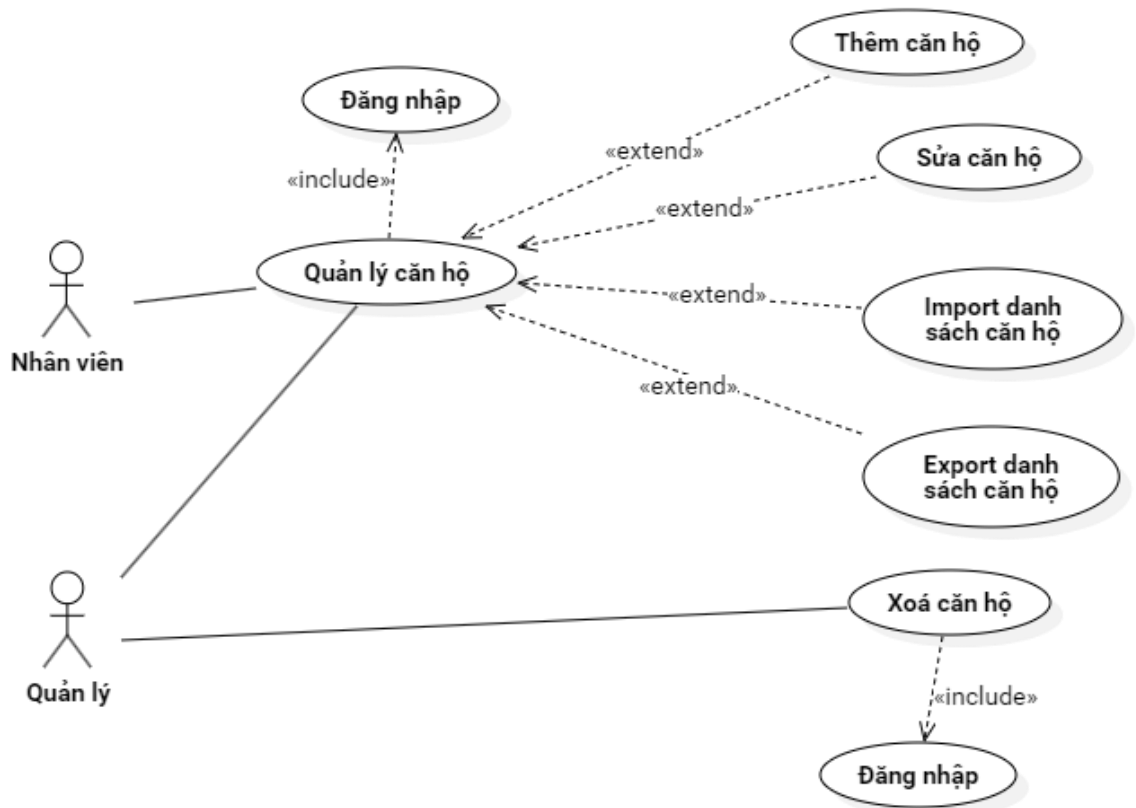
Bảng 3.9: Đặc tả use case export danh sách hợp đồng

- Đặc tả xoá hợp đồng

Tên use case:	Xoá hợp đồng
Tác nhân:	Quản lý
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng phải đăng nhập • Đang ở trang Quản lý hợp đồng
Hậu điều kiện:	<ul style="list-style-type: none"> • Xoá thành công hợp đồng
Luồng sự kiện chính:	1. Người dùng chọn hợp đồng mong muốn xoá 2. Người dùng chọn nút “Xoá” 3. Hệ thống hỏi xác nhận xoá 4. Người dùng xác nhận xoá 5. Hệ thống xoá thành công hợp đồng
Luồng sự kiện phụ:	A. Người dùng xác nhận không xoá 1. Hệ thống hiển thị trang Quản lý hợp đồng

Bảng 3.10: Đặc tả use case xoá hợp đồng

3.1.3.4. Chức năng quản lý căn hộ



Hình 3.6: Biểu đồ use case quản lý căn hộ

- Đặc tả thêm căn hộ

Tên use case:	Thêm căn hộ
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> Người dùng phải đăng nhập Đang ở trang Quản lý căn hộ
Hậu điều kiện:	<ul style="list-style-type: none"> Thêm thành công căn hộ mới
Luồng sự kiện chính:	<ol style="list-style-type: none"> 1. Người dùng chọn nút “Thêm” 2. Hệ thống hiển thị form thêm căn hộ 3. Người dùng nhập thông tin bắt buộc 4. Người dùng nhấn nút “Lưu” 5. Hệ thống kiểm tra thông tin 6. Hệ thống cho phép thêm căn hộ 7. Hệ thống lưu căn hộ

Luồng sự kiện phụ:	A. Thông tin không đúng yêu cầu 1. Hệ thống báo lỗi 2. Người dùng nhập lại thông tin không đúng yêu cầu
---------------------------	---

Bảng 3.11: Đặc tả use case thêm căn hộ

- Đặc tả sửa căn hộ

Tên use case:	Sửa căn hộ
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> Người dùng phải đăng nhập Đang ở trang Quản lý căn hộ
Hậu điều kiện:	<ul style="list-style-type: none"> Thêm thành công căn hộ
Luồng sự kiện chính:	1. Người dùng chọn căn hộ mong muốn chỉnh sửa 2. Người dùng chọn nút “Sửa” 3. Hệ thống hiển thị form sửa căn hộ 4. Người dùng nhập thông tin chỉnh sửa 5. Người dùng nhấn nút “Lưu” 6. Hệ thống kiểm tra thông tin 7. Hệ thống cho phép cập nhật căn hộ 8. Hệ thống cập nhật căn hộ
Luồng sự kiện phụ:	A. Thông tin không đúng yêu cầu 1. Hệ thống báo lỗi 2. Người dùng nhập lại thông tin không đúng yêu cầu

Bảng 3.12: Đặc tả use case sửa căn hộ

- Đặc tả import danh sách căn hộ

Tên use case:	Import danh sách căn hộ
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> Người dùng phải đăng nhập Đang ở trang Quản lý căn hộ
Hậu điều kiện:	<ul style="list-style-type: none"> Thêm thành công danh sách căn hộ
Luồng sự kiện chính:	1. Người dùng chọn nút “Import” 2. Hệ thống hiển thị form import 3. Người dùng chọn file cần import 4. Người dùng nhấn nút “Gửi” 5. Hệ thống kiểm tra file được gửi lên 6. Hệ thống cho phép import 7. Hệ thống lưu danh sách căn hộ 8. Hệ thống thông báo kết quả import 9. Hệ thống import thành công

Luồng sự kiện phụ:	A. File không đúng yêu cầu 1. Hệ thống báo lỗi 2. Người dùng chọn file lại B. Nội dung danh sách nhập không đúng yêu cầu 1. Hệ thống báo lỗi 2. Người dùng chỉnh sửa lại nội dung
---------------------------	--

Bảng 3.13: Đặc tả use case import danh sách căn hộ

- Đặc tả export danh sách căn hộ

Tên use case:	Export danh sách căn hộ
Tác nhân:	Quản lý, nhân viên
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng phải đăng nhập • Đang ở trang Quản lý căn hộ
Hậu điều kiện:	<ul style="list-style-type: none"> • Tải thành công danh sách căn hộ
Luồng sự kiện chính:	1. Người dùng chọn nút “Export” 2. Hệ thống xử lý file 3. Hệ thống gửi file về cho trình duyệt 4. Trình duyệt tự động tải file 5. Hệ thống export thành công
Luồng sự kiện phụ:	Không có

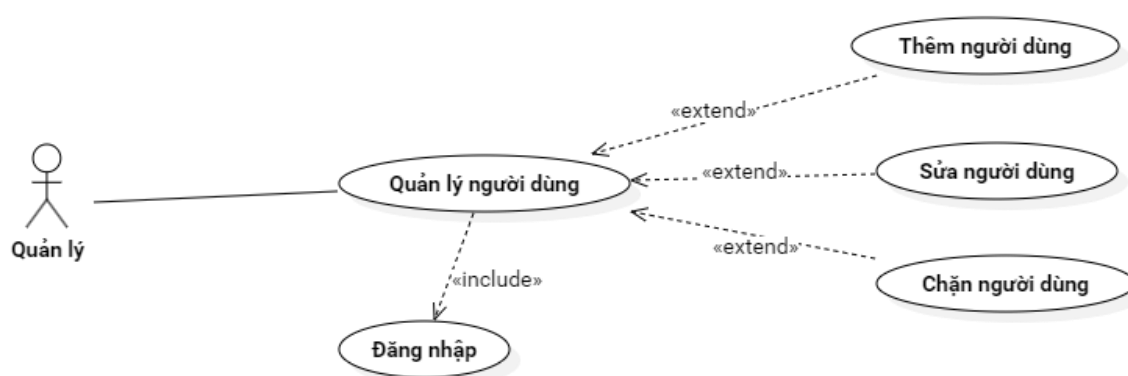
Bảng 3.14: Đặc tả use case export danh sách căn hộ

- Đặc tả xoá căn hộ

Tên use case:	Xoá căn hộ
Tác nhân:	Quản lý
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng phải đăng nhập • Đang ở trang Quản lý căn hộ
Hậu điều kiện:	<ul style="list-style-type: none"> • Xoá thành công căn hộ
Luồng sự kiện chính:	1. Người dùng chọn căn hộ mong muốn xoá 2. Người dùng chọn nút “Xoá” 3. Hệ thống hỏi xác nhận xoá 4. Người dùng xác nhận xoá 5. Hệ thống xoá thành công căn hộ
Luồng sự kiện phụ:	A. Người dùng xác nhận không xoá 1. Hệ thống hiển thị trang Quản lý căn hộ

Bảng 3.15: Đặc tả use case xoá căn hộ

3.1.3.5. Chức năng quản lý người dùng



Hình 3.7: Biểu đồ use case quản lý người dùng

- Đặc tả thêm căn hộ

Tên use case:	Thêm người dùng
Tác nhân:	Quản lý
Điều kiện tiên quyết:	<ul style="list-style-type: none"> Người dùng phải đăng nhập Đang ở trang Quản lý người dùng
Hậu điều kiện:	<ul style="list-style-type: none"> Thêm thành công người dùng mới
Luồng sự kiện chính:	<ol style="list-style-type: none"> Người dùng chọn nút “Thêm” Hệ thống hiển thị form thêm người dùng Người dùng nhập thông tin bắt buộc Người dùng nhấn nút “Lưu” Hệ thống kiểm tra thông tin Hệ thống cho phép thêm người dùng Hệ thống lưu người dùng
Luồng sự kiện phụ:	A. Thông tin không đúng yêu cầu <ol style="list-style-type: none"> Hệ thống báo lỗi Người dùng nhập lại thông tin không đúng yêu cầu

Bảng 3.16: Đặc tả use case thêm người dùng

- Đặc tả sửa người dùng

Tên use case:	Sửa người dùng
Tác nhân:	Quản lý
Điều kiện tiên quyết:	<ul style="list-style-type: none"> Người dùng phải đăng nhập Đang ở trang Quản lý người dùng
Hậu điều kiện:	<ul style="list-style-type: none"> Sửa thành công người dùng
Luồng sự kiện chính:	<ol style="list-style-type: none"> Người dùng chọn người dùng mong muốn chỉnh sửa Người dùng chọn nút “Sửa”

	3. Hệ thống hiển thị form sửa người dùng 4. Người dùng nhập thông tin chỉnh sửa 5. Người dùng nhấn nút “Lưu” 6. Hệ thống kiểm tra thông tin 7. Hệ thống cho phép cập nhật người dùng 8. Hệ thống cập nhật người dùng
Luồng sự kiện phụ:	A. Thông tin không đúng yêu cầu 1. Hệ thống báo lỗi 2. Người dùng nhập lại thông tin không đúng yêu cầu

Bảng 3.17: Đặc tả use case sửa người dùng

- Đặc tả chặn người dùng

Tên use case:	Chặn người dùng
Tác nhân:	Quản lý
Điều kiện tiên quyết:	<ul style="list-style-type: none"> Người dùng phải đăng nhập Đang ở trang Quản lý người dùng
Hậu điều kiện:	<ul style="list-style-type: none"> Chặn thành công người dùng
Luồng sự kiện chính:	1. Người dùng chọn người dùng mong muốn chặn 2. Người dùng chọn nút “Chặn” 3. Hệ thống hỏi xác nhận chặn 4. Người dùng xác nhận chặn 5. Hệ thống chặn thành công người dùng
Luồng sự kiện phụ:	A. Người dùng xác nhận không chặn 1. Hệ thống hiển thị trang Quản lý người dùng

Bảng 3.18: Đặc tả use case chặn người dùng

3.1.3.6. Chức năng thống kê

- Đặc tả thống kê

Tên use case:	Thống kê
Tác nhân:	Quản lý
Điều kiện tiên quyết:	<ul style="list-style-type: none"> Người dùng đã đăng nhập Đang ở Trang chủ
Hậu điều kiện:	<ul style="list-style-type: none"> Người dùng xem thống kê thành công
Luồng sự kiện chính:	1. Người dùng chọn loại thống kê muốn xem 2. Hệ thống hiển thị kết quả người dùng chọn
Luồng sự kiện phụ:	Không có

Bảng 3.19: Đặc tả use case thống kê

3.1.3.7. Chức năng đăng nhập

- Đặc tả đăng nhập

Tên use case:	Đăng nhập
Tác nhân:	Người dùng
Điều kiện tiên quyết:	<ul style="list-style-type: none">• Tài khoản người dùng đã được tạo sẵn• Đang ở trang Đăng nhập
Hậu điều kiện:	<ul style="list-style-type: none">• Người dùng đăng nhập ứng dụng thành công• Hệ thống ghi nhận hoạt động đăng nhập thành công
Luồng sự kiện chính:	<ol style="list-style-type: none">1. Người dùng nhập Email và Password2. Hệ thống kiểm tra thông tin3. Tài khoản đăng nhập đúng4. Hệ thống chuyển người dùng đến Trang chủ
Luồng sự kiện phụ:	A. Đăng nhập thất bại <ol style="list-style-type: none">1. Thông báo lỗi2. Về lại trang Đăng nhập

Bảng 3.20: Đặc tả use case đăng nhập

3.1.3.8. Chức năng đăng xuất

- Đặc tả đăng xuất

Tên use case:	Đăng xuất
Tác nhân:	Người dùng
Điều kiện tiên quyết:	<ul style="list-style-type: none">• Người dùng đã đăng nhập• Đang ở Trang chủ
Hậu điều kiện:	<ul style="list-style-type: none">• Người dùng đăng xuất ứng dụng thành công• Hệ thống ghi nhận hoạt động đăng xuất thành công
Luồng sự kiện chính:	<ol style="list-style-type: none">1. Người dùng chọn nút “Đăng xuất”2. Hệ thống hỏi xác nhận đăng xuất3. Hệ thống xoá phiên đăng nhập4. Hệ thống chuyển người dùng đến trang Đăng nhập
Luồng sự kiện phụ:	A. Người dùng huỷ đăng xuất <ol style="list-style-type: none">1. Hệ thống hiển thị Trang chủ

Bảng 3.21: Đặc tả use case đăng xuất

3.1.3.9. Chức năng đổi mật khẩu

- Đặc tả đổi mật khẩu

Tên use case:	Đổi mật khẩu
Tác nhân:	Người dùng

Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng đã đăng nhập • Đang ở Trang chủ
Hậu điều kiện:	<ul style="list-style-type: none"> • Người dùng đổi mật khẩu thành công
Luồng sự kiện chính:	<ol style="list-style-type: none"> 1. Người dùng chọn vào Ảnh đại diện 2. Hệ thống hiển thị Trang thông tin 3. Người dùng chọn chức năng “Đổi mật khẩu” 4. Hệ thống hiển thị trang “Đổi mật khẩu” 5. Người dùng nhập thông tin đổi mật khẩu 6. Hệ thống kiểm tra thông tin 7. Hệ thống cho phép cập nhật mật khẩu mới 8. Hệ thống cập nhật thông tin 9. Hệ thống cập nhật mật khẩu mới
Luồng sự kiện phụ:	A. Thông tin không đúng yêu cầu <ol style="list-style-type: none"> 1. Hệ thống báo lỗi 2. Người dùng nhập lại thông tin không đúng yêu cầu

Bảng 3.22: Đặc tả use case đổi mật khẩu

3.1.3.10. Chức năng quên mật khẩu

- Đặc tả quên mật khẩu

Tên use case:	Quên mật khẩu
Tác nhân:	Người dùng
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Tài khoản người dùng được tạo sẵn • Người dùng không nhớ mật khẩu • Người dùng đang ở trang Đăng nhập
Hậu điều kiện:	<ul style="list-style-type: none"> • Người dùng đặt lại được mật khẩu mới • Đăng nhập thành công
Luồng sự kiện chính:	<ol style="list-style-type: none"> 1. Chọn Quên mật khẩu 2. Hệ thống hiển thị trang Quên mật khẩu 3. Nhập Email tài khoản quên 4. Hệ thống mật khẩu mới 5. Hệ thống hiển thị trang Đăng nhập
Luồng sự kiện phụ:	A. Email không hợp lệ <ol style="list-style-type: none"> 1. Hệ thống thông báo Email không hợp lệ

Bảng 3.23: Đặc tả use case quên mật khẩu

3.1.3.11. Chức năng cập nhật thông tin

- Đặc tả cập nhật thông tin

Tên use case:	Cập nhật thông tin
----------------------	--------------------

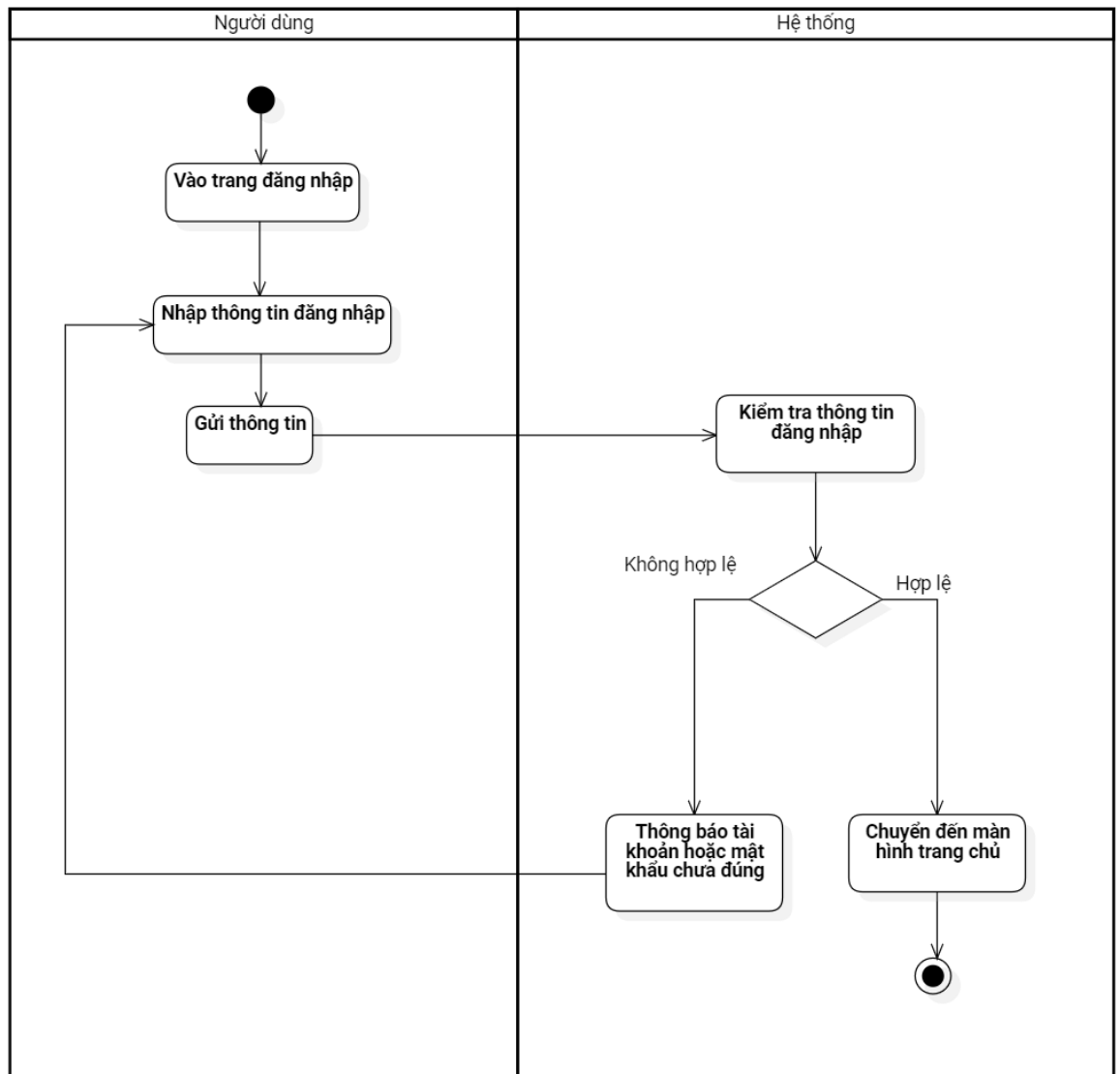
Tác nhân:	Người dùng
Điều kiện tiên quyết:	<ul style="list-style-type: none"> • Người dùng đã đăng nhập • Đang ở Trang chủ
Hậu điều kiện:	<ul style="list-style-type: none"> • Người dùng cập nhật thông tin thành công
Luồng sự kiện chính:	<ol style="list-style-type: none"> 1. Người dùng chọn vào Ảnh đại diện 2. Hệ thống hiển thị Trang thông tin 3. Người dùng nhập thông tin cần chỉnh sửa 4. Hệ thống kiểm tra thông tin 5. Hệ thống cho phép cập nhật thông tin 6. Hệ thống cập nhật thông tin
Luồng sự kiện phụ:	A. Thông tin không đúng yêu cầu <ol style="list-style-type: none"> 1. Hệ thống báo lỗi 2. Người dùng nhập lại thông tin không đúng yêu cầu

Bảng 3.24: Đặc tả use case cập nhật thông tin

3.1.4. Biểu đồ hoạt động

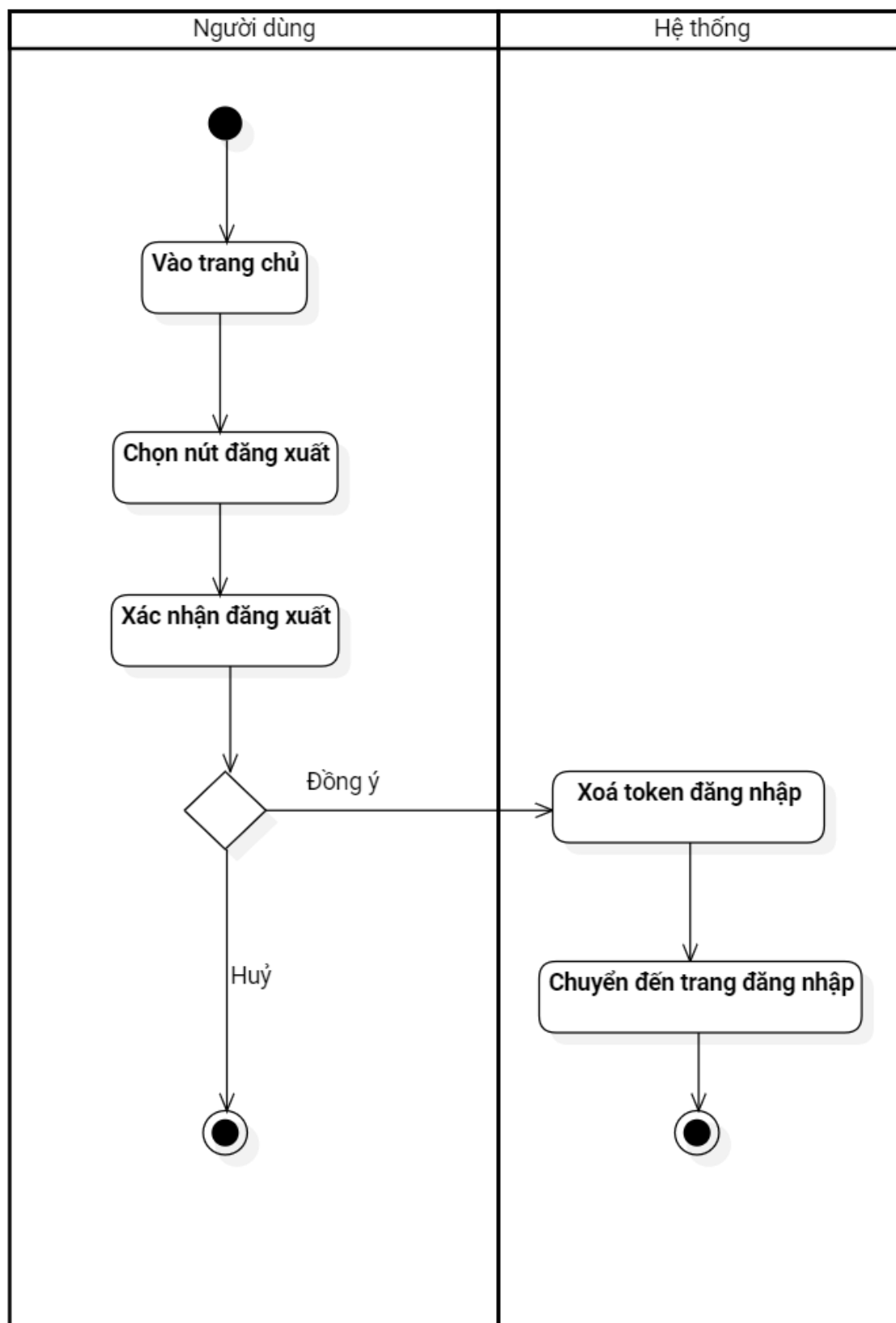
3.1.4.1. Chức năng người dùng chung

3.1.4.1.1. Biểu đồ hoạt động đăng nhập



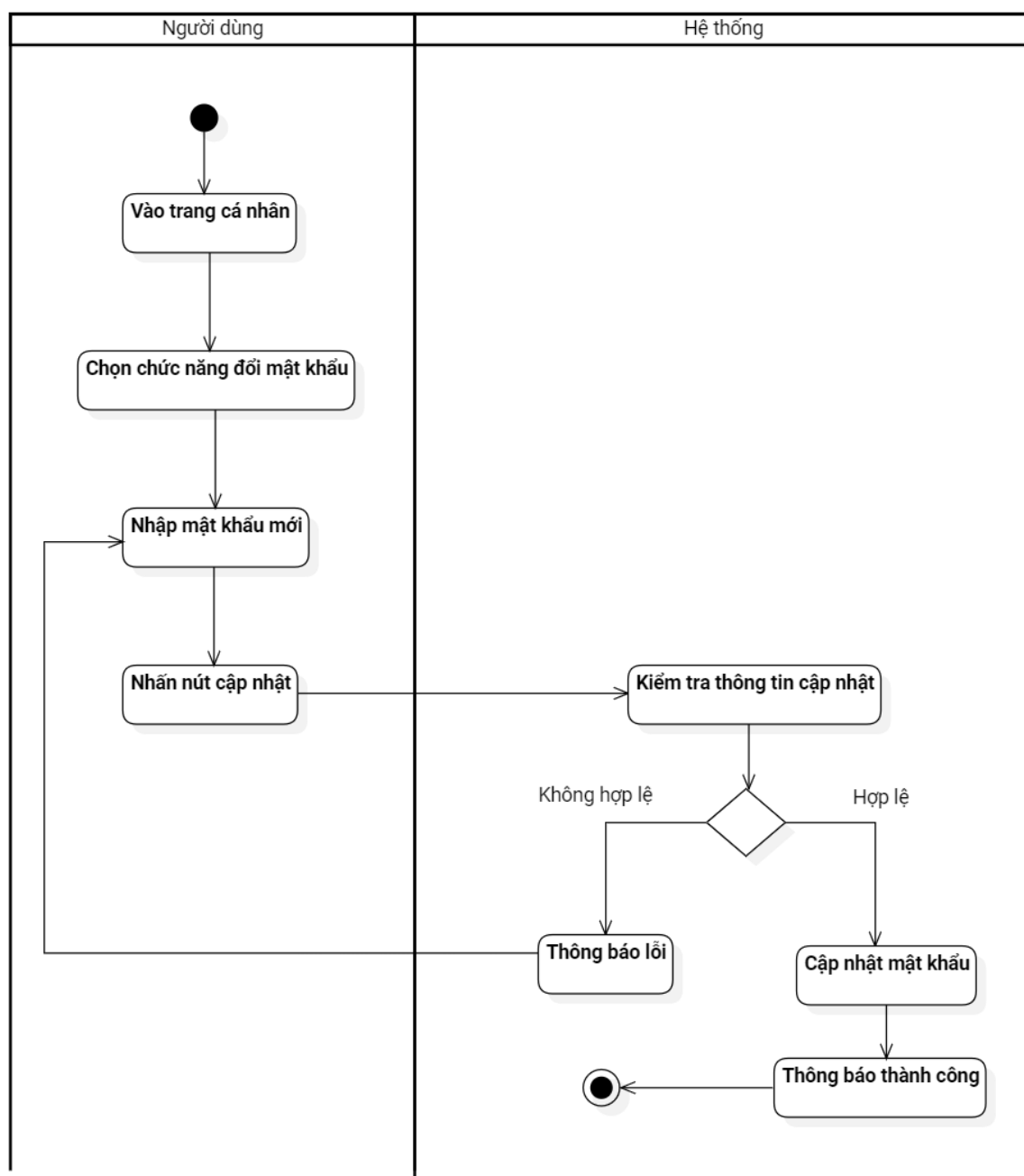
Hình 3.8: Biểu đồ hoạt động đăng nhập

3.1.4.1.2. Biểu đồ hoạt động đăng xuất



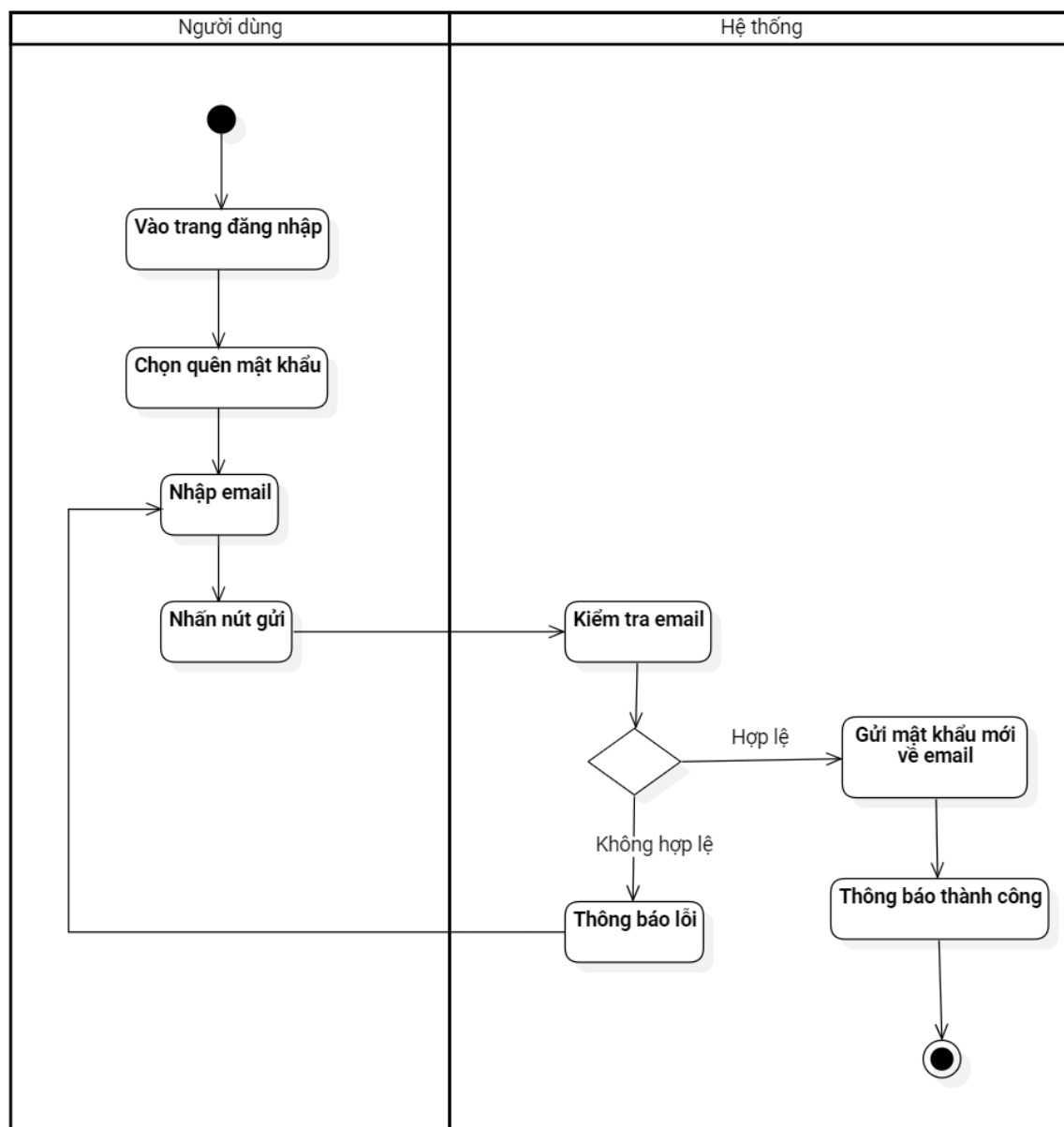
Hình 3.9: Biểu đồ hoạt động đăng xuất

3.1.4.1.3. Biểu đồ hoạt động đổi mật khẩu



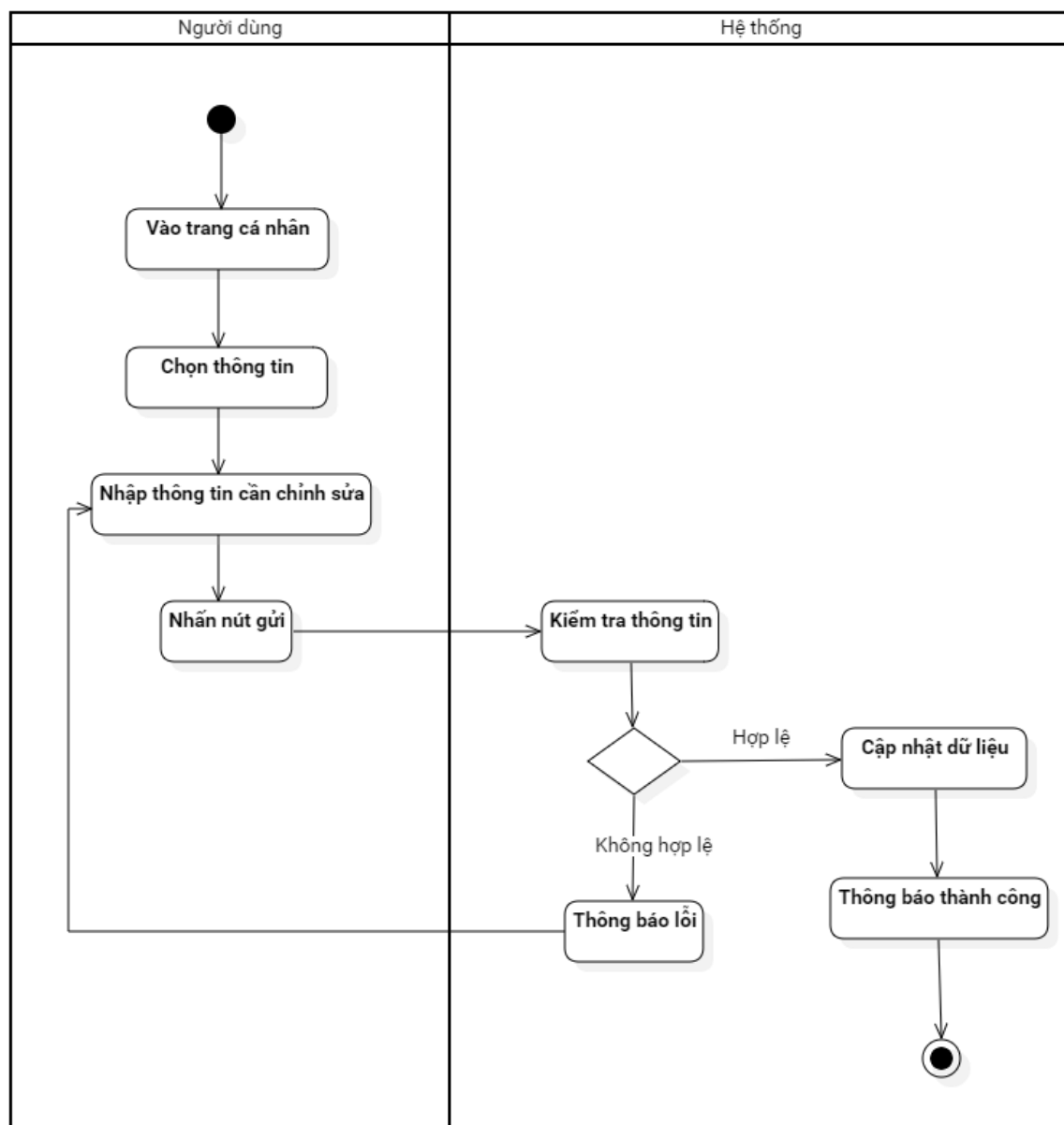
Hình 3.10: Biểu đồ hoạt động đổi mật khẩu

3.1.4.1.4. Biểu đồ hoạt động quên mật khẩu



Hình 3.11: Biểu đồ hoạt động quên mật khẩu

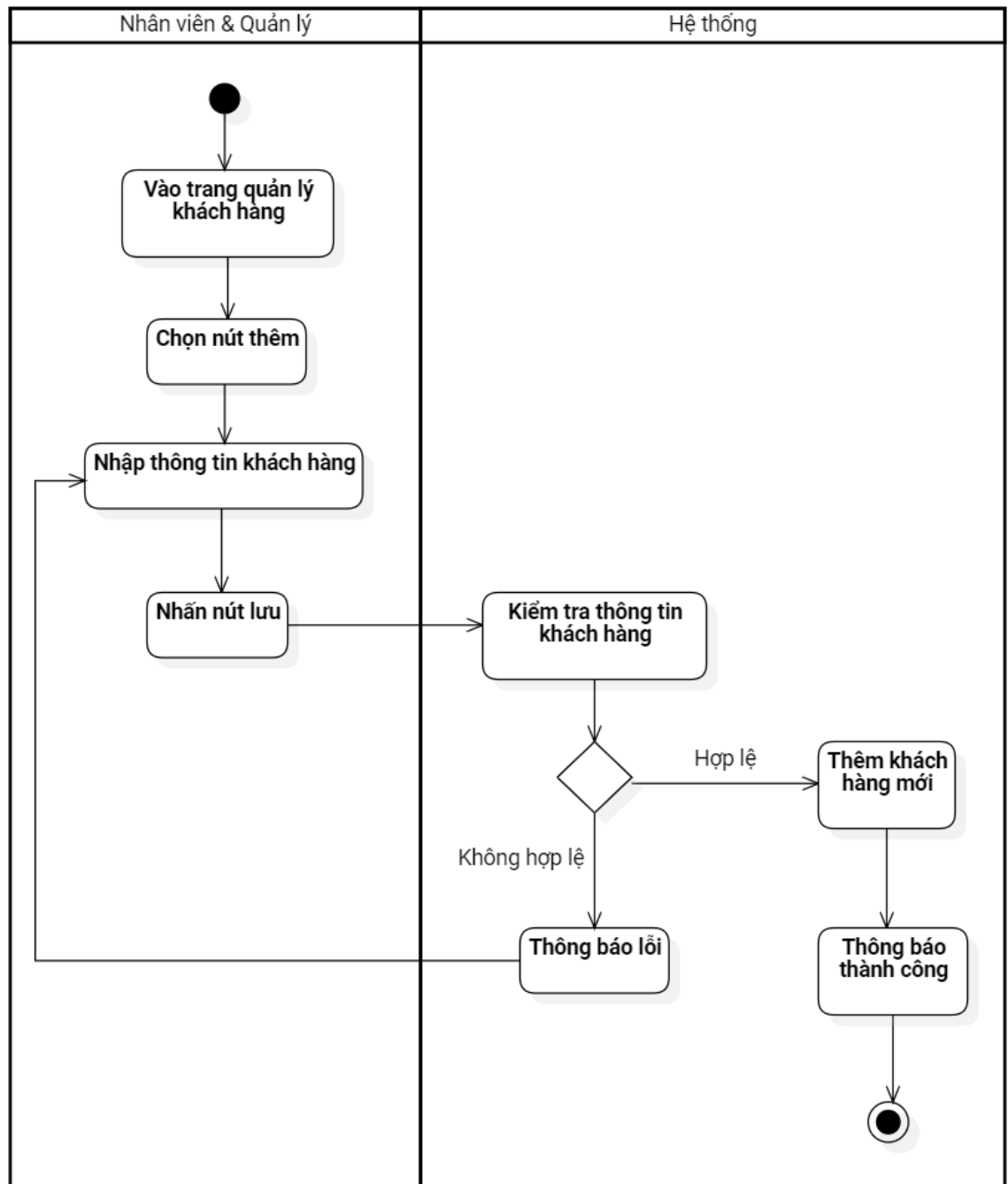
3.1.4.1.5. Biểu đồ hoạt động cập nhật thông tin



Hình 3.12: Biểu đồ hoạt động cập nhật thông tin

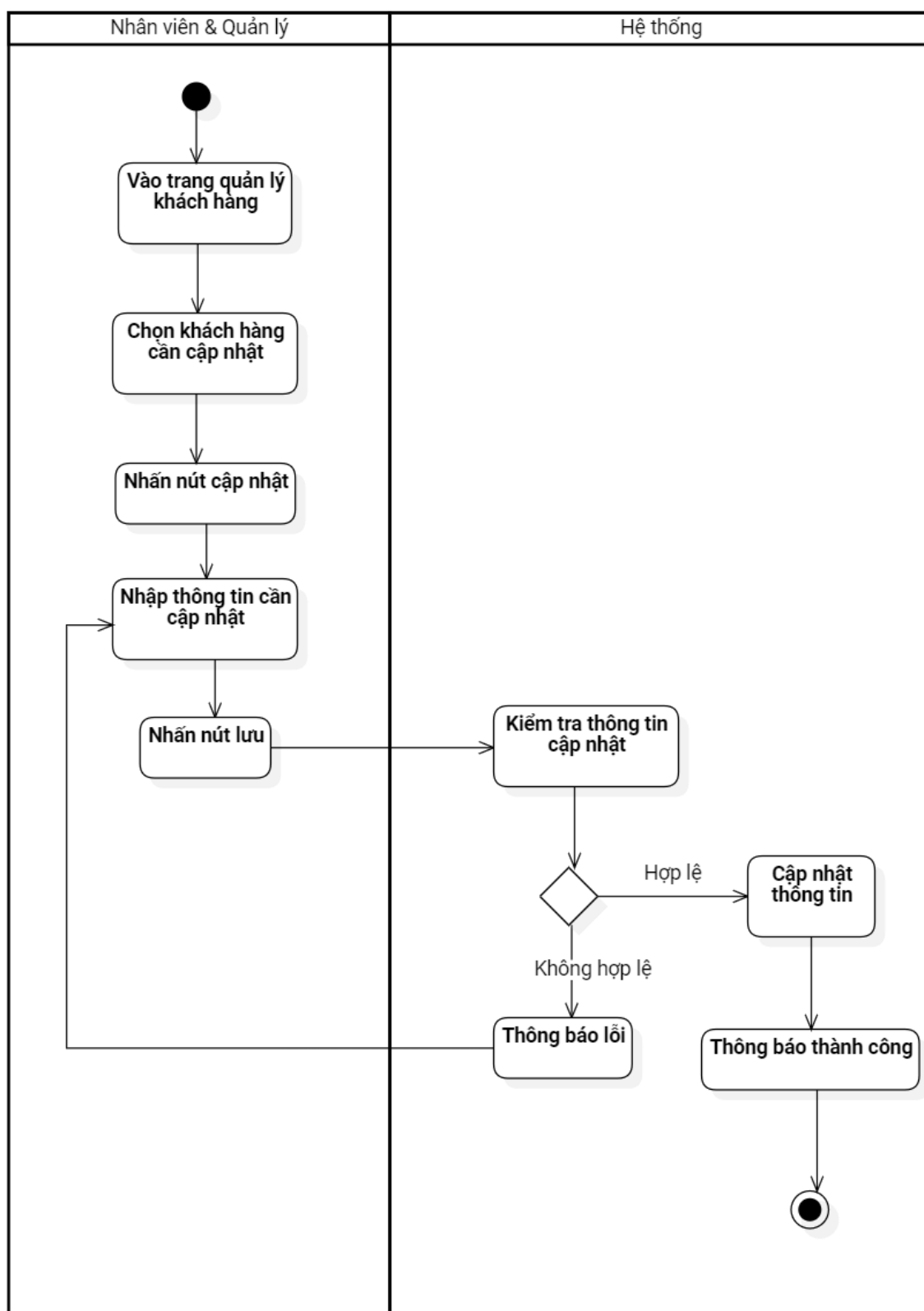
3.1.4.2. Chức năng quản lý khách hàng

3.1.4.2.1. Biểu đồ hoạt động thêm khách hàng



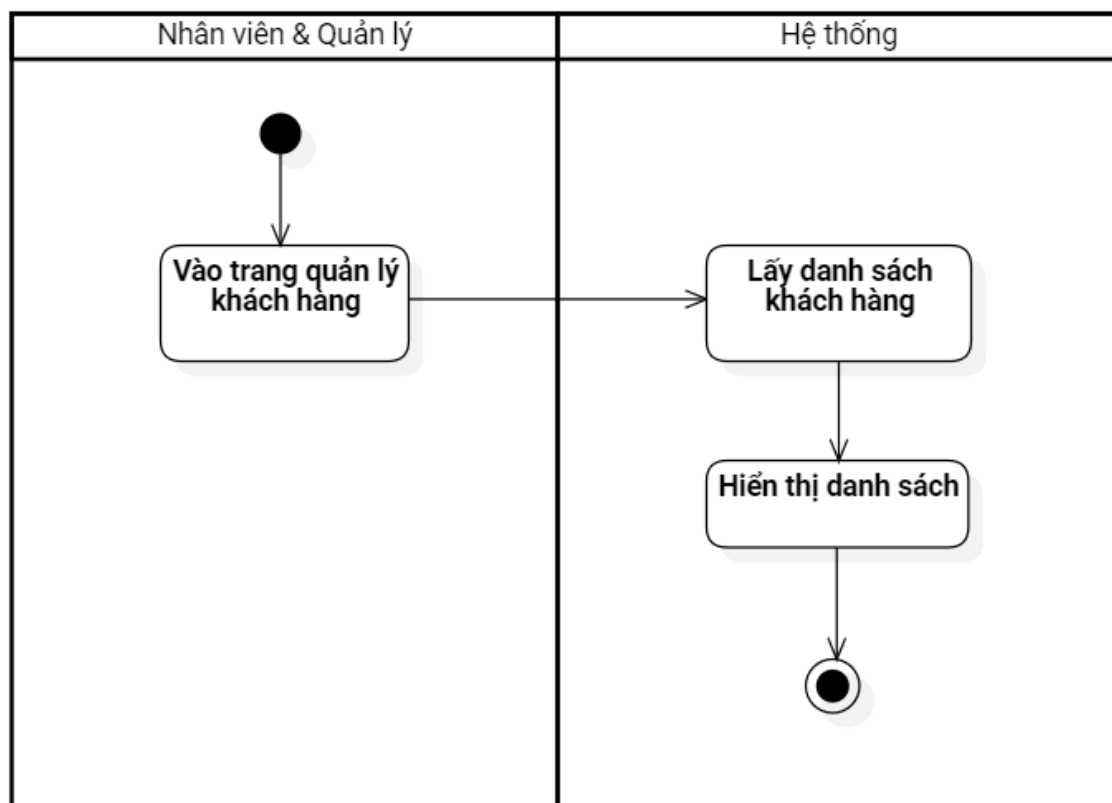
Hình 3.13: Biểu đồ hoạt động thêm khách hàng

3.1.4.2.2. Biểu đồ hoạt động cập nhật khách hàng



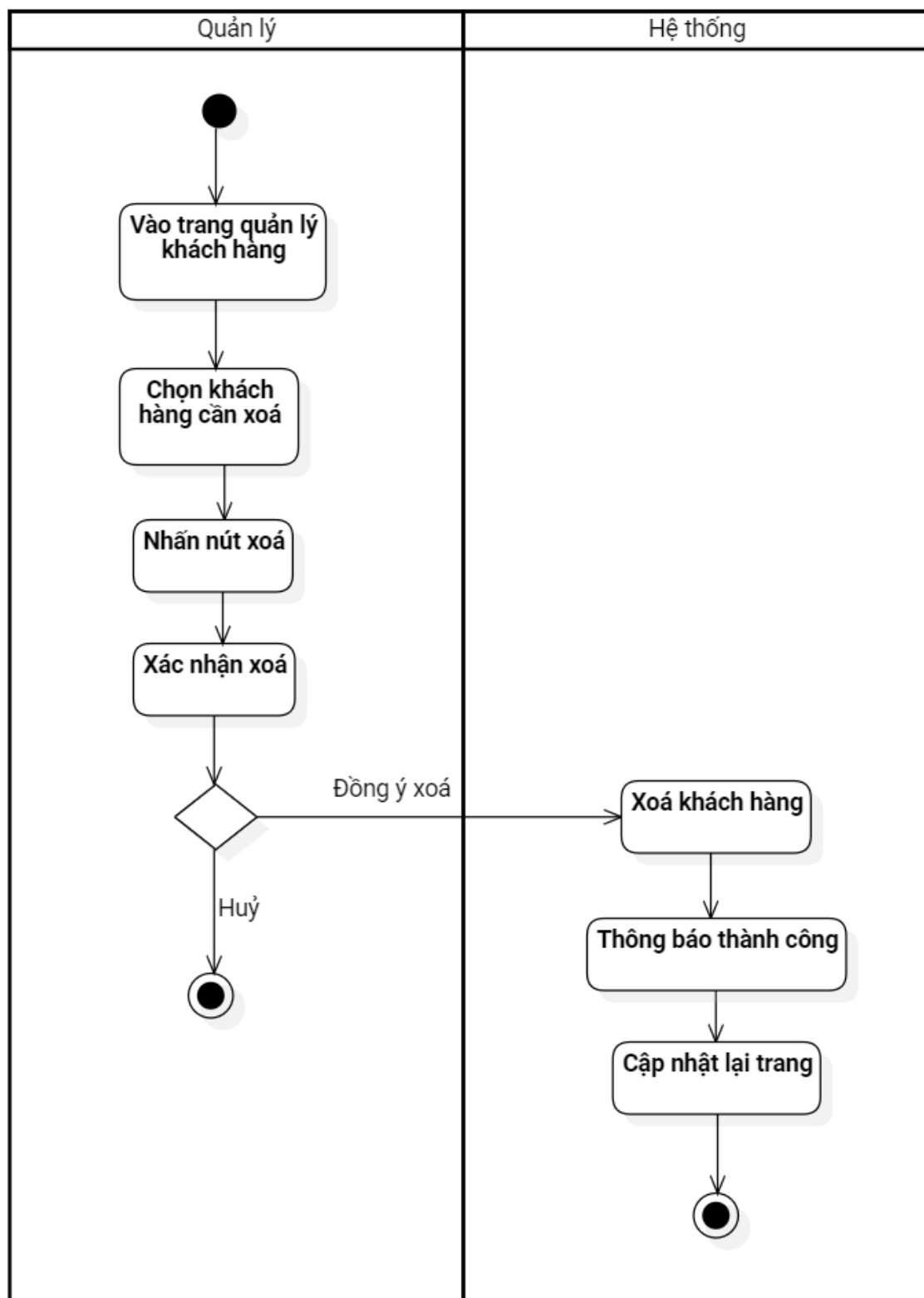
Hình 3.14: Biểu đồ hoạt động cập nhật khách hàng

3.1.4.2.3. Biểu đồ hoạt động xem danh sách khách hàng



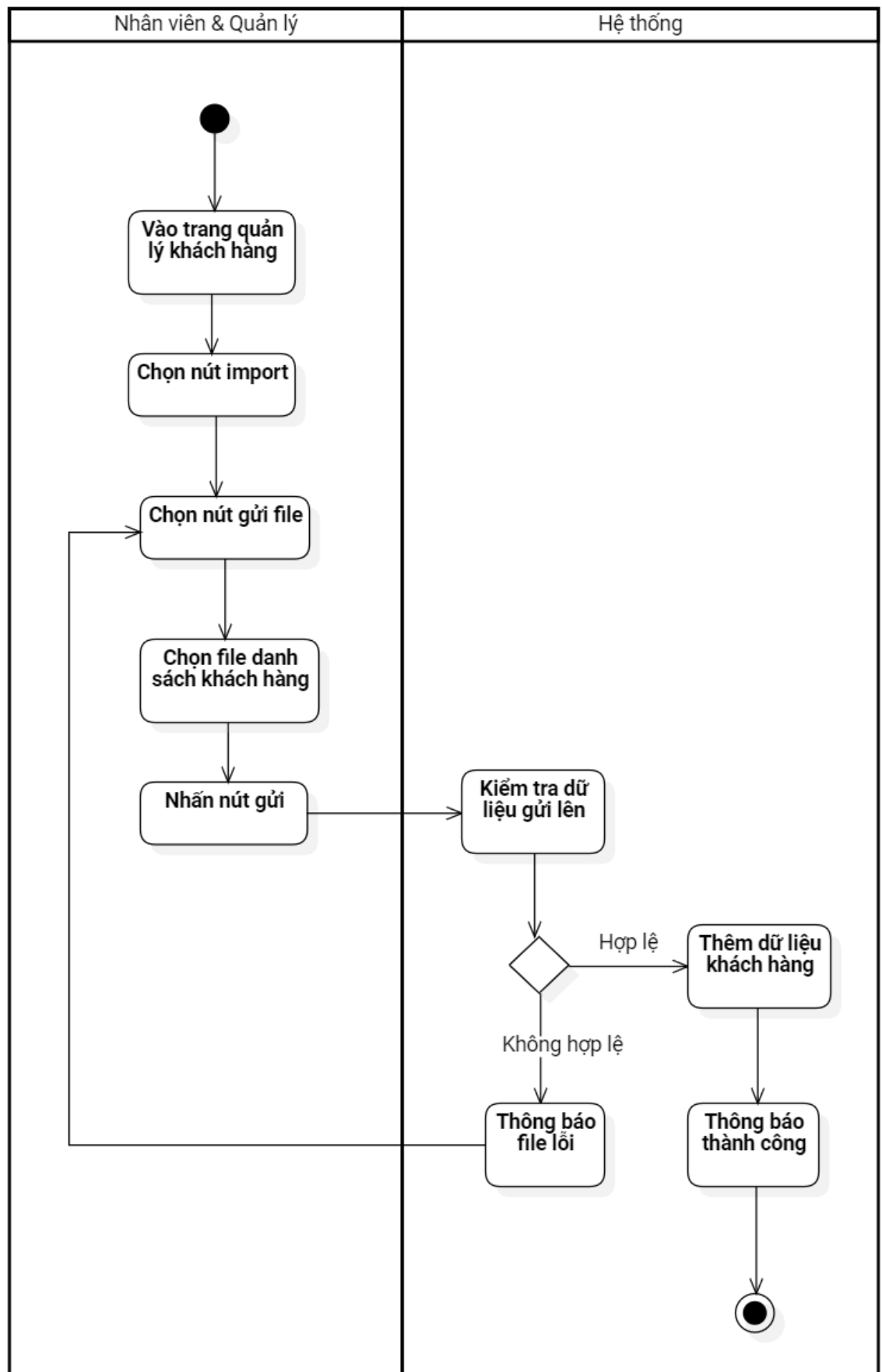
Hình 3.15: Biểu đồ hoạt động xem danh sách khách hàng

3.1.4.2.4. Biểu đồ hoạt động xoá khách hàng



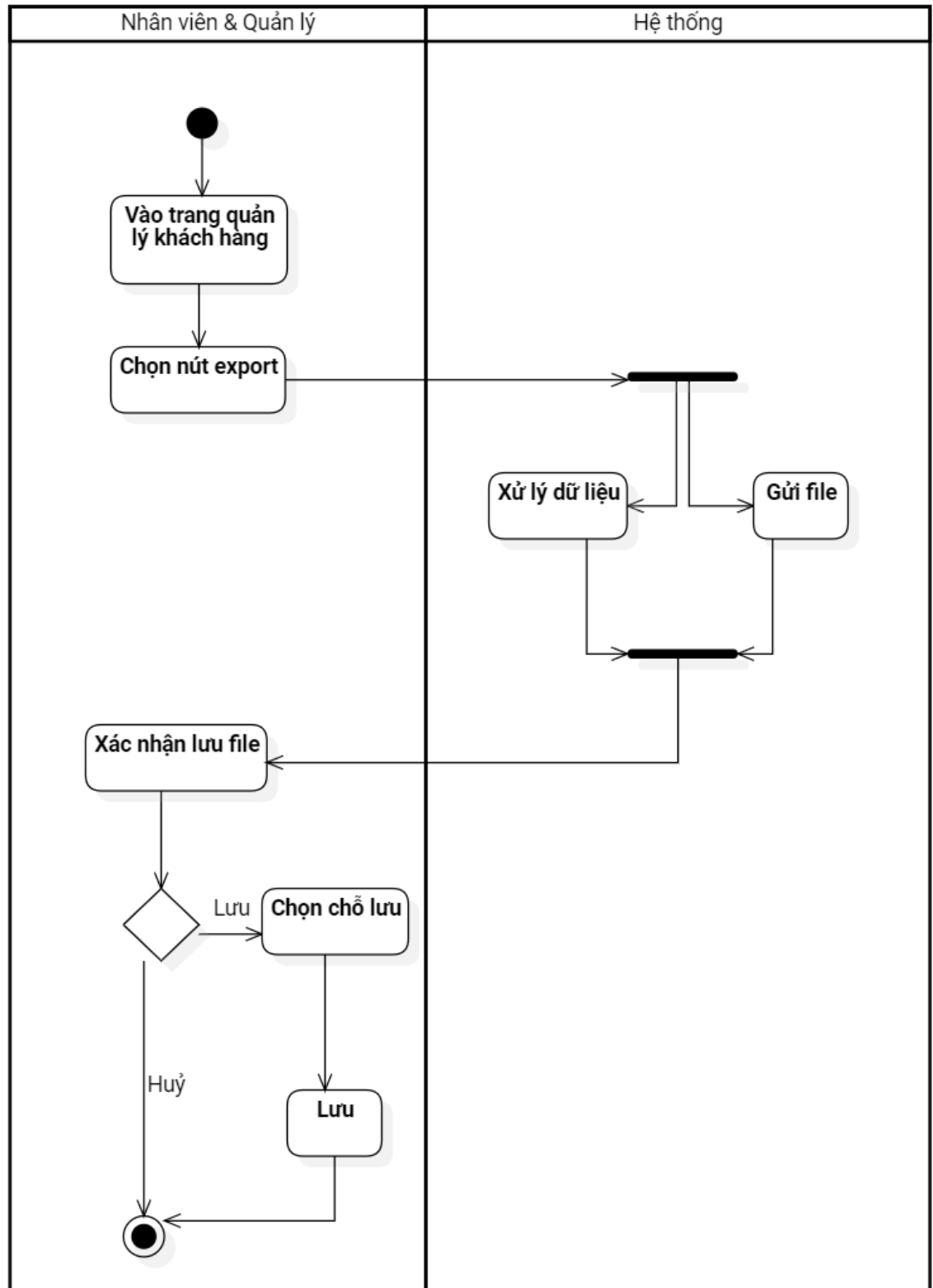
Hình 3.16: Biểu đồ hoạt động xoá khách hàng

3.1.4.2.5. Biểu đồ hoạt động import khách hàng



Hình 3.17: Biểu đồ hoạt động import khách hàng

3.1.4.2.6. Biểu đồ hoạt động export khách hàng

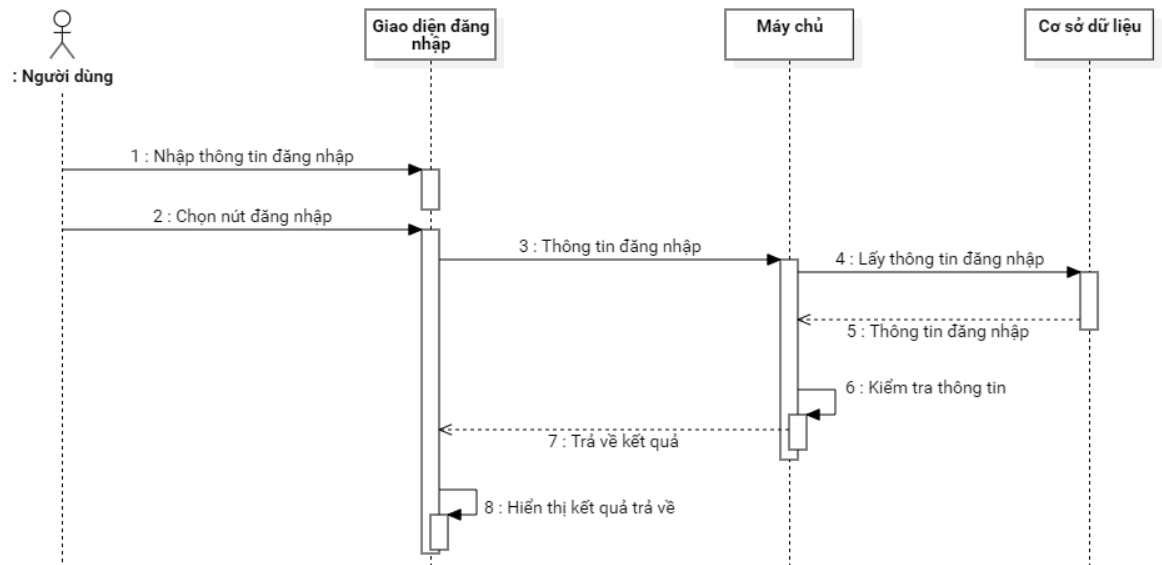


Hình 3.18: Biểu đồ hoạt động export khách hàng

3.1.5. Biểu đồ tuần tự

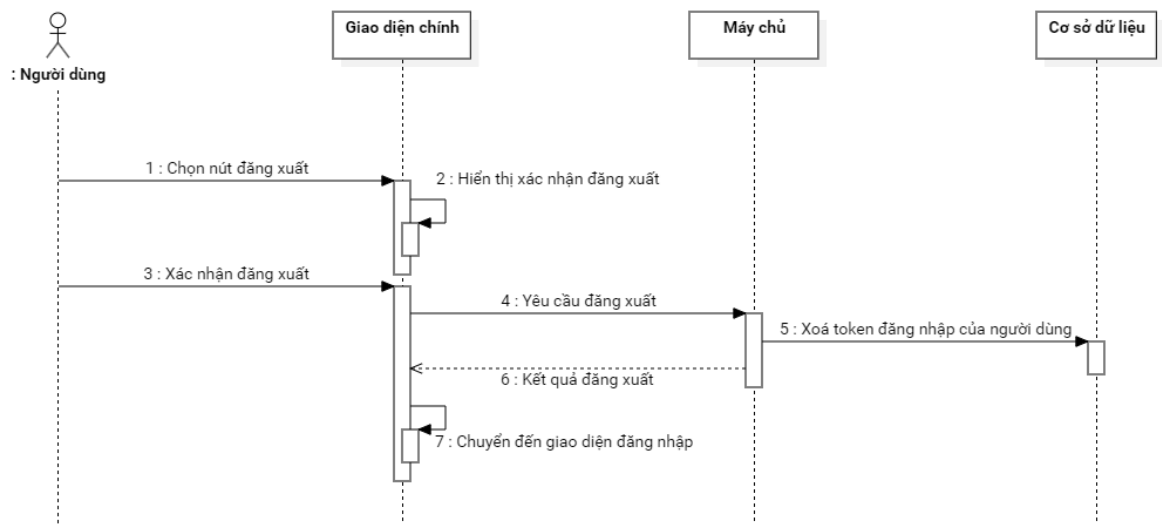
3.1.5.1. Chức năng người dùng chung

3.1.5.1.1. Biểu đồ tuần tự đăng nhập



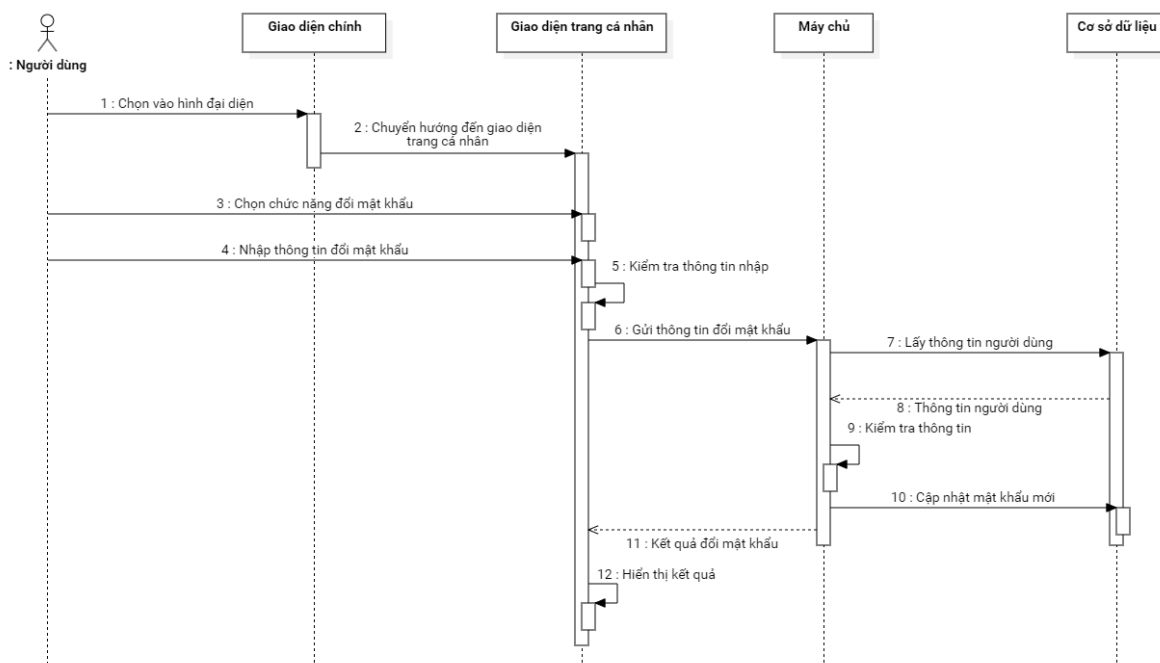
Hình 3.19: Biểu đồ tuần tự đăng nhập

3.1.5.1.2. Biểu đồ tuần tự đăng xuất



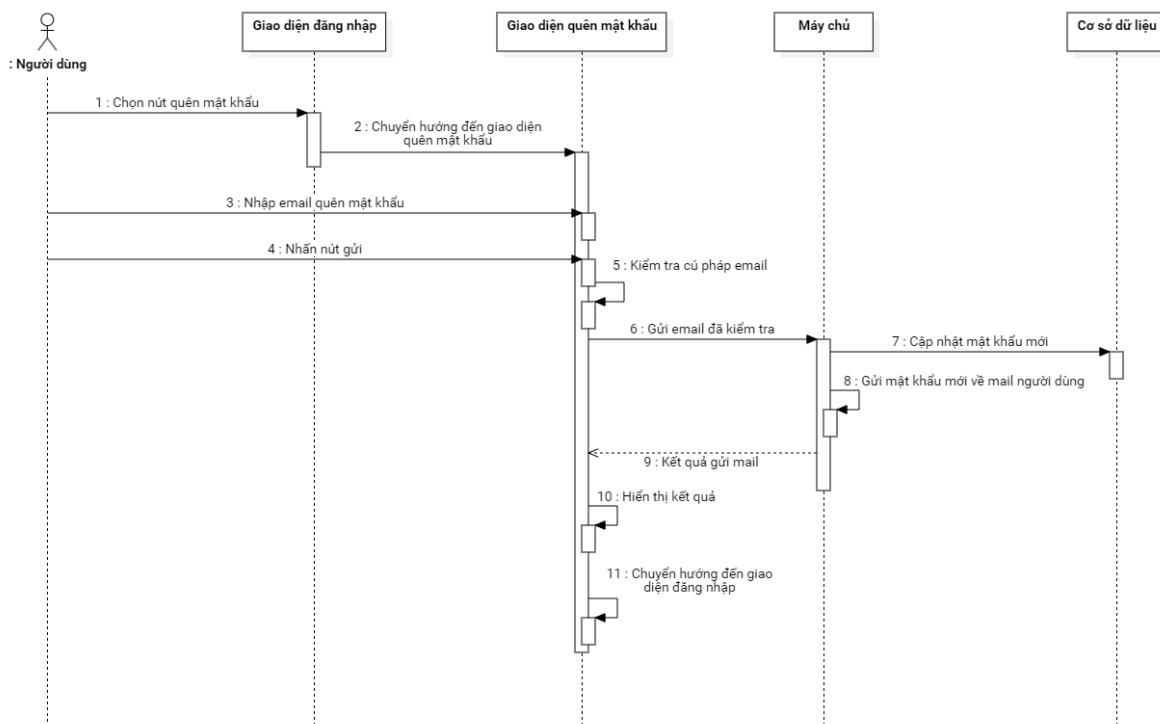
Hình 3.20: Biểu đồ tuần tự đăng xuất

3.1.5.1.3. Biểu đồ tuần tự đổi mật khẩu



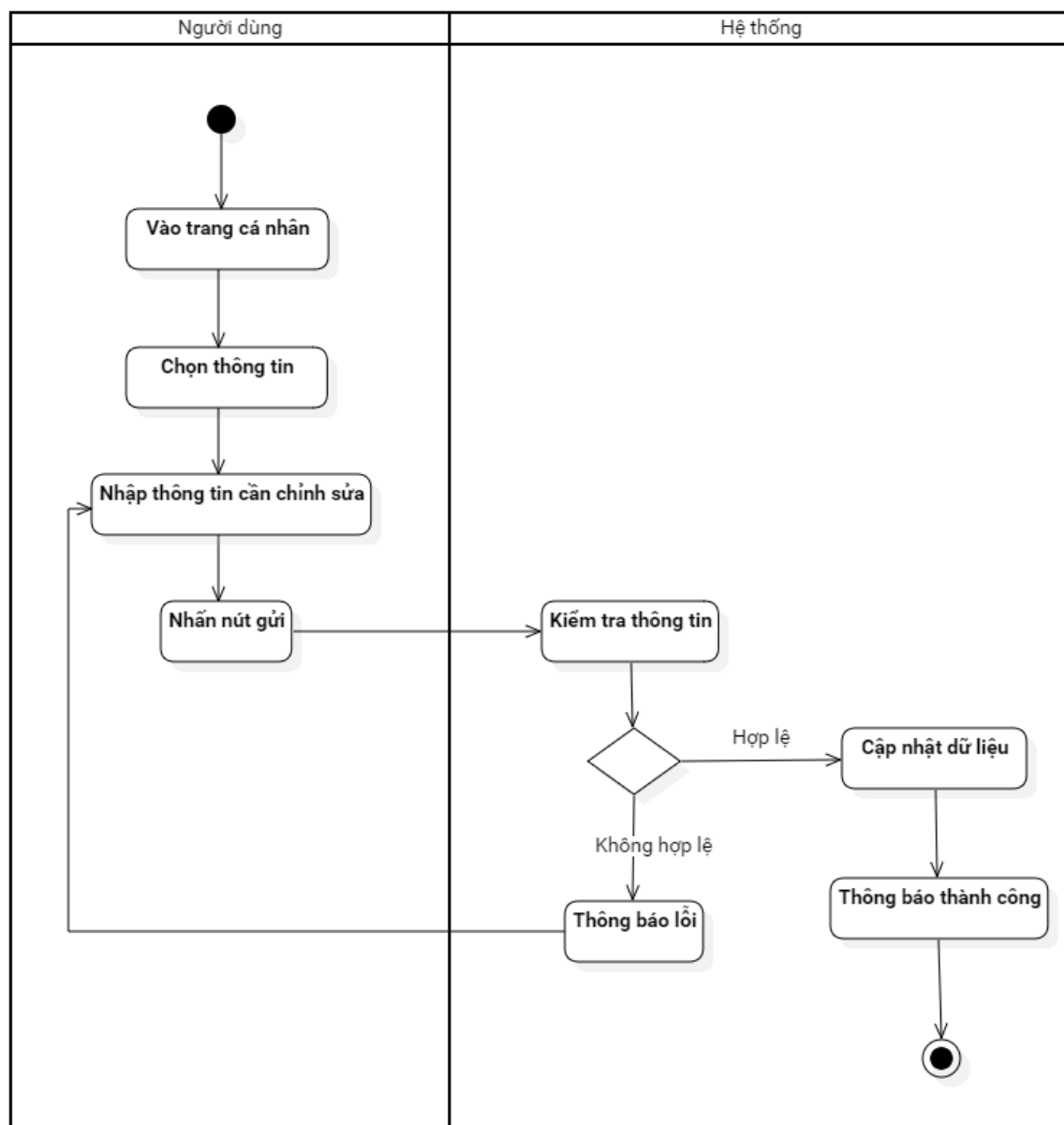
Hình 3.21: Biểu đồ tuần tự đổi mật khẩu

3.1.5.1.4. Biểu đồ tuần tự quên mật khẩu



Hình 3.22: Biểu đồ tuần tự quên mật khẩu

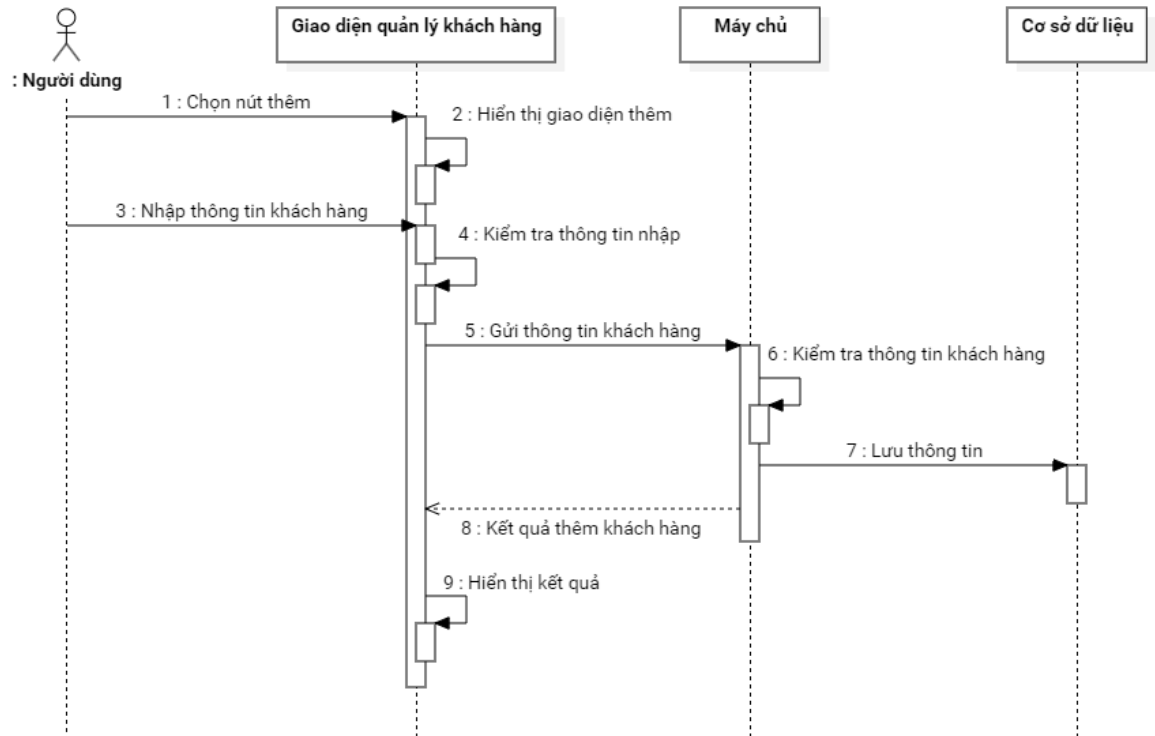
3.1.5.1.5. Biểu đồ tuần tự cập nhật thông tin



Hình 3.23: Biểu đồ tuần tự cập nhật thông tin

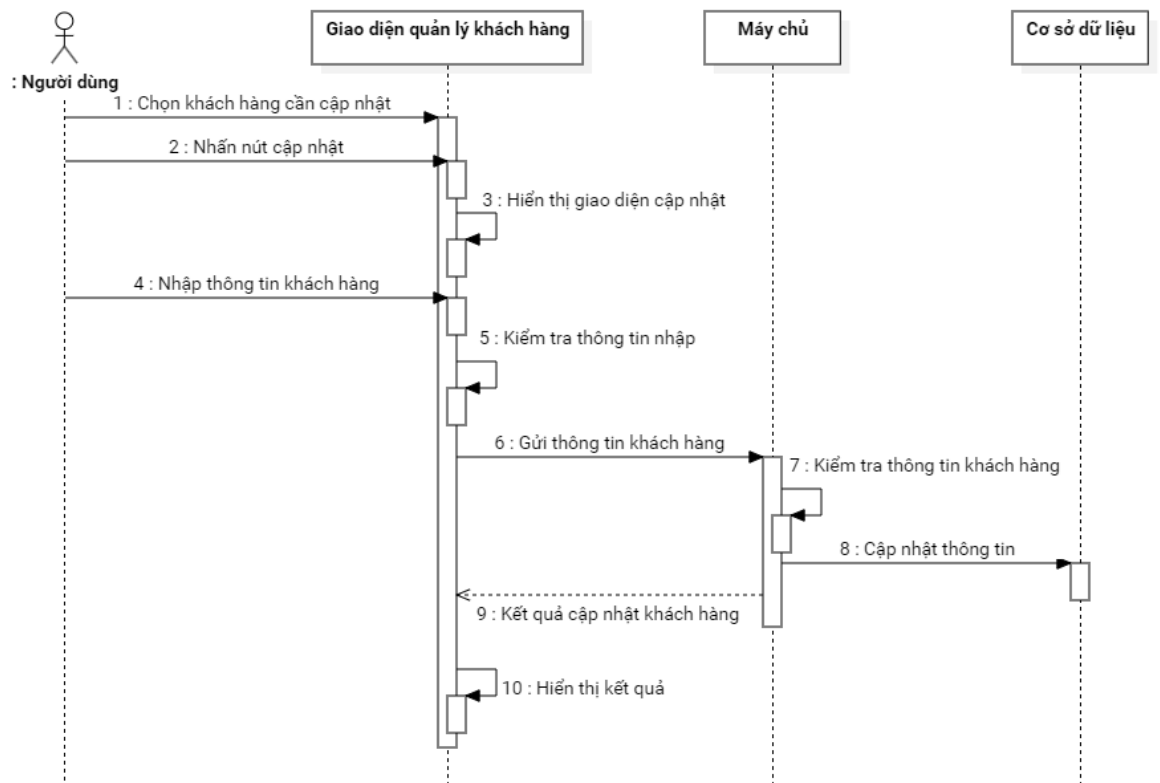
3.1.5.2. Chức năng quản lý khách hàng

3.1.5.2.1. Biểu đồ tuần tự thêm khách hàng



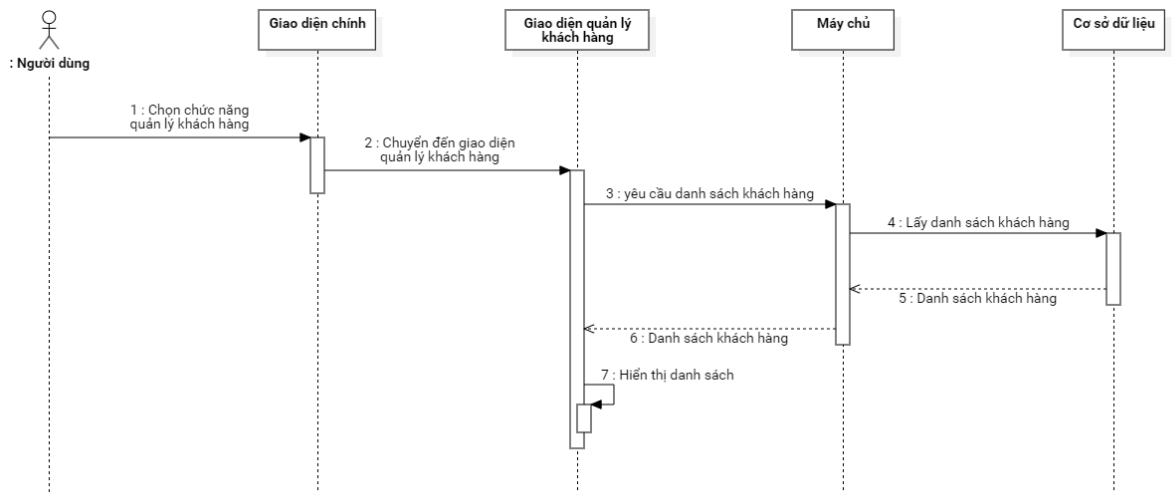
Hình 3.24: Biểu đồ tuần tự thêm khách hàng

3.1.5.2.2. Biểu đồ tuần tự cập nhật khách hàng



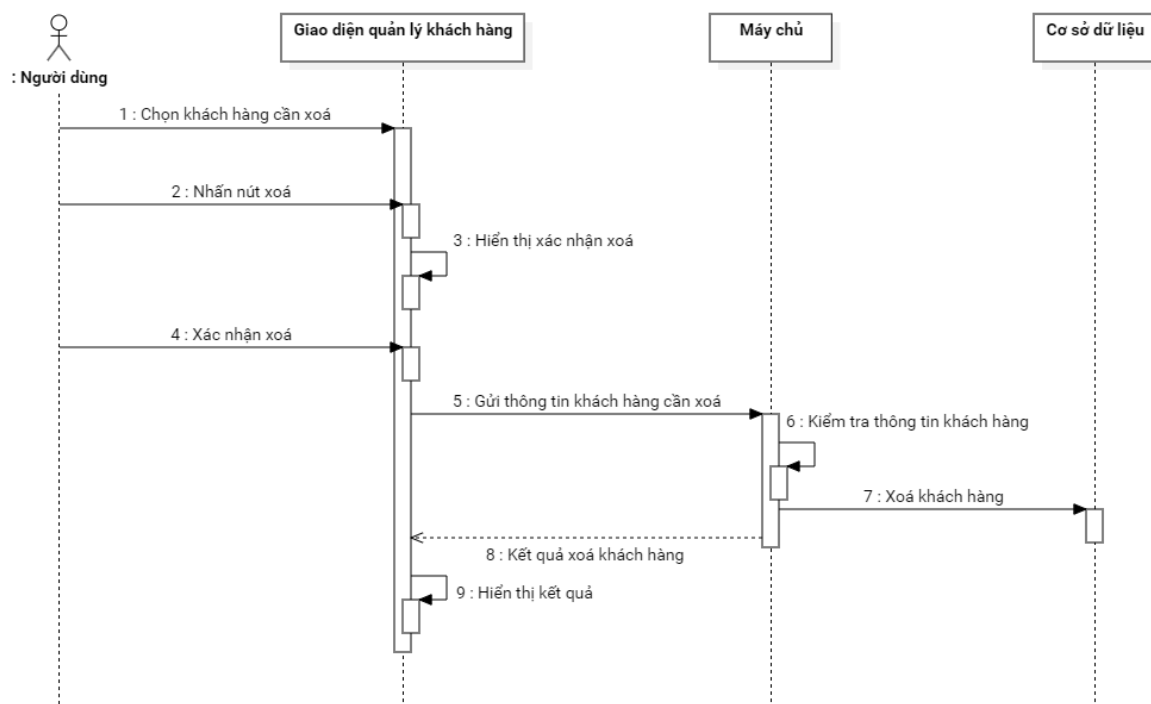
Hình 3.25: Biểu đồ tuần tự cập nhật khách hàng

3.1.5.2.3. Biểu đồ tuần tự xem danh sách khách hàng



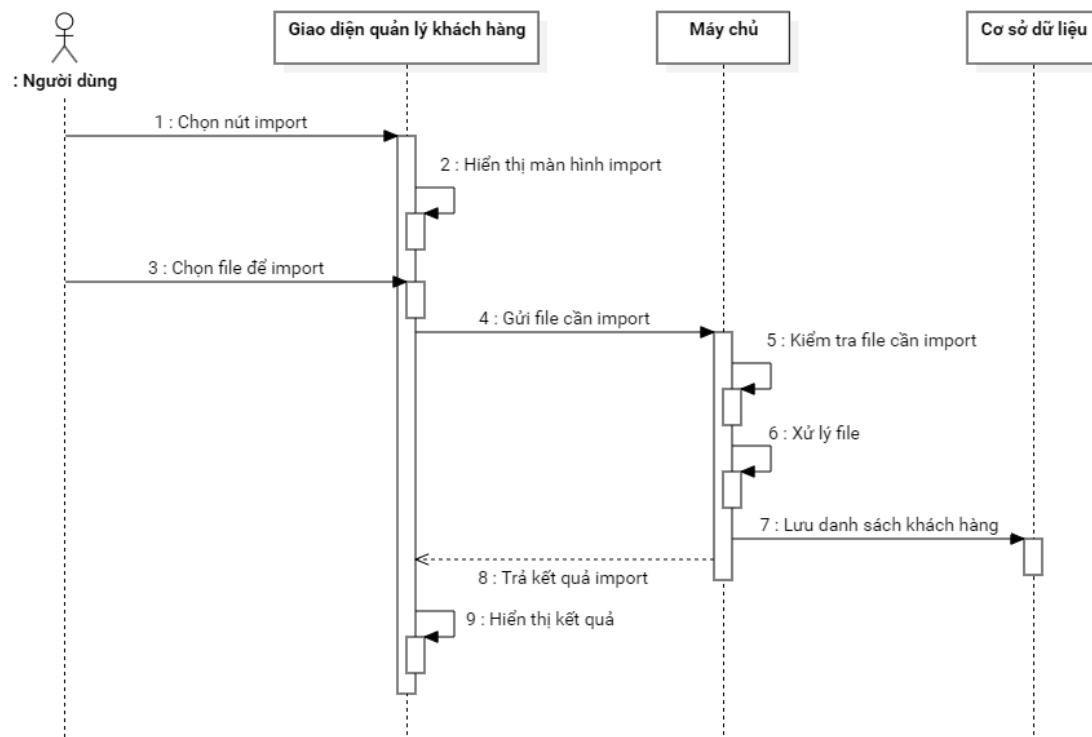
Hình 3.26: Biểu đồ tuần tự xem danh sách khách hàng

3.1.5.2.4. Biểu đồ tuần tự xoá khách hàng



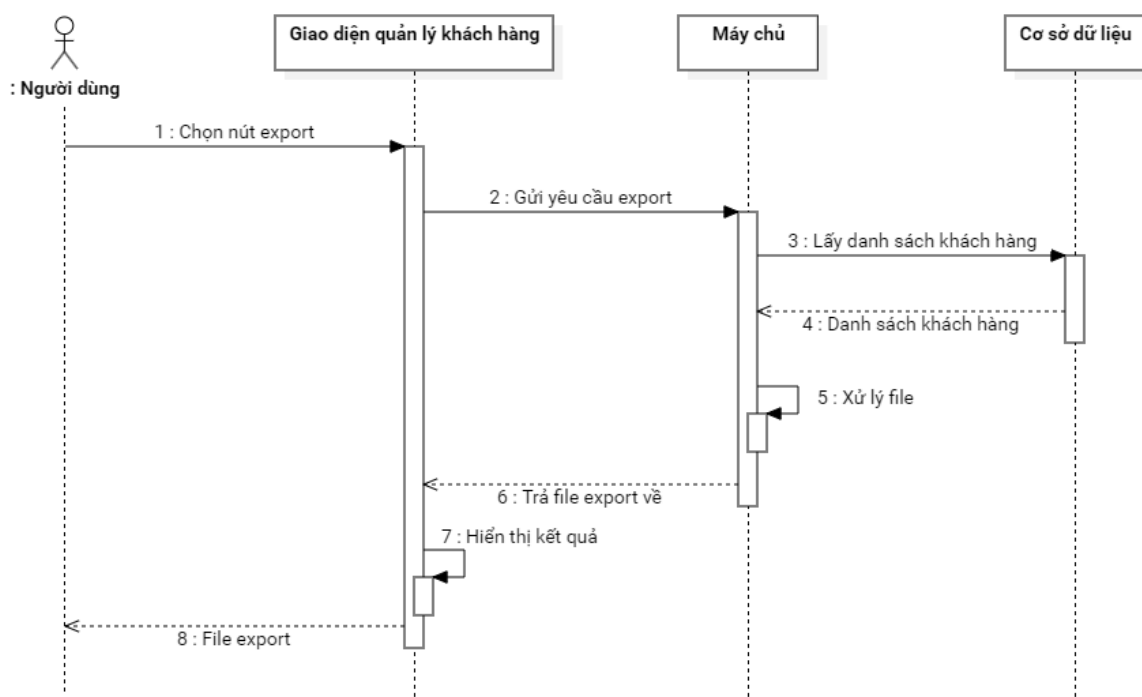
Hình 3.27: Biểu đồ tuần tự xoá khách hàng

3.1.5.2.5. Biểu đồ tuần tự import khách hàng



Hình 3.28: Biểu đồ tuần tự import khách hàng

3.1.5.2.6. Biểu đồ tuần tự export khách hàng



Hình 3.29: Biểu đồ tuần tự export khách hàng

CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH

4.1. Tổng quan các công cụ, công nghệ hỗ trợ xây dựng hệ thống

4.1.1. Phát triển Backend

4.1.1.1. IntelliJ IDEA

IntelliJ IDEA là một môi trường phát triển tích hợp (IDE) mạnh mẽ được thiết kế đặc biệt cho lập trình Java, và cũng có khả năng hỗ trợ một số ngôn ngữ khác như Node.js, Python hay nhiều ngôn ngữ khác. Phiên bản đầu tiên của IntelliJ IDEA được phát hành vào tháng 1/2001, từ đó đến nay nó đã trở thành một trong những IDE hàng đầu trong cộng đồng phát triển phần mềm.

Mặc dù ban đầu được biết đến chủ yếu là một IDE Java, IntelliJ IDEA đã mở rộng sự hỗ trợ của mình cho nhiều ngôn ngữ và framework khác nhau. Nó cung cấp một loạt các tính năng thông minh như trình soạn thảo mã thông minh, công cụ phân tích mã và bộ công cụ refactorings giúp tối ưu hóa quy trình phát triển phần mềm. [3]

4.1.1.2. MySQL

MySQL là hệ quản trị cơ sở dữ liệu mã nguồn mở phổ biến hàng đầu trên thế giới (gọi tắt là RDBMS) và đặc biệt MySQL được ưa chuộng trong quá trình xây dựng, phát triển ứng dụng. MySQL được đánh giá là hệ quản trị cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có khả năng thay đổi mô hình sử dụng phù hợp với điều kiện công việc. MySQL hoạt động trên nhiều hệ điều hành, cung cấp một hệ thống lớn các hàm tiện ích rất mạnh.

MySQL thích hợp với các ứng dụng có truy cập cơ sở dữ liệu trên Internet nhờ vào tốc độ cũng như tính bảo mật cao. MySQL có thể tải miễn phí từ trang chủ với nhiều phiên bản cho các hệ điều hành khác nhau như phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix, FreeBSD,... [4]

4.1.1.3. Java Spring Boot

Spring Boot là một trong số các module của Spring framework chuyên cung cấp các tính năng RAD (Rapid Application Development) cho phép tạo ra và phát triển các ứng dụng độc lập dựa trên Spring một cách nhanh chóng.

Spring Boot ra đời với mục đích loại bỏ những cấu hình phức tạp của Spring, nó không yêu cầu cấu hình XML và nâng cao năng suất cho các nhà phát triển. Với sự góp mặt của Spring Boot, hệ sinh thái Spring đã trở nên mạnh mẽ, phổ biến và hiệu quả hơn bao giờ hết.

4.1.1.4. JSON Web Token (JWT)

JWT là một phương tiện đại diện cho các yêu cầu chuyên giao giữa hai bên Client – Server, các thông tin trong chuỗi JWT được định dạng bằng JSON. Trong

đó chuỗi Token phải có 3 phần là header, payload và signature được ngăn bằng dấu “.”.

4.1.2. Phát triển Frontend

4.1.2.1. Visual Studio Code

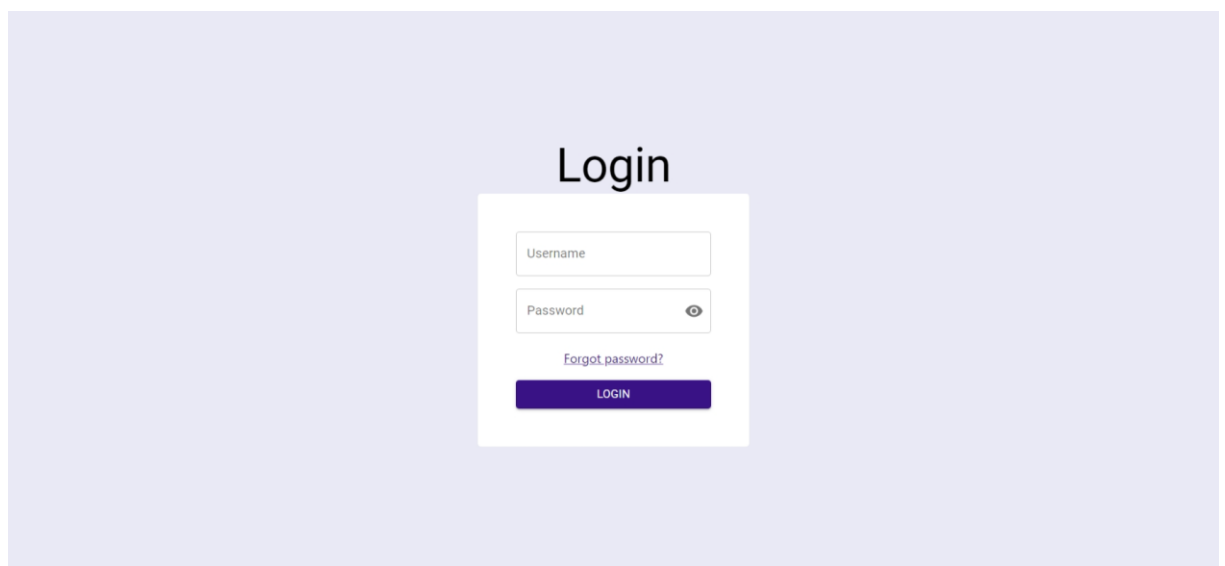
Visual Studio Code là trình soạn thảo, biên tập lập trình mã nguồn miễn phí được sử dụng trên 3 nền tảng đó là: Windows, macOS và Linux được xây dựng, phát triển bởi Microsoft. Visual Studio Code được các chuyên gia công nghệ thông tin đánh giá cao, nó là sự kết hợp hoàn hảo giữa IDE và Code Editor.

4.1.2.2. ReactJs

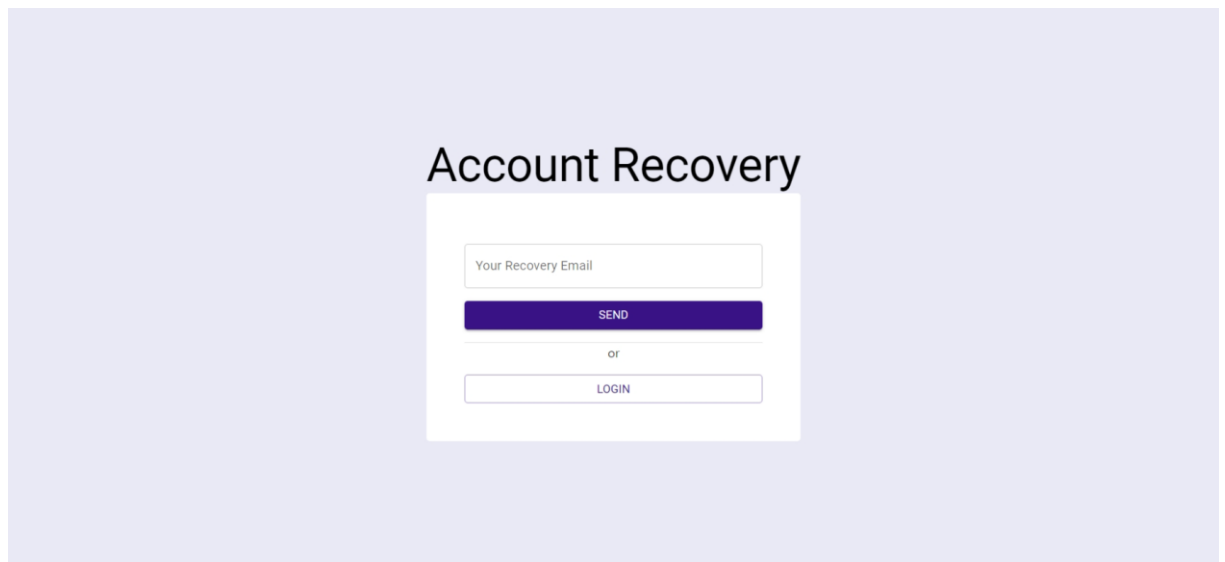
ReactJS được phát triển bởi Facebook và được giới thiệu lần đầu tiên vào năm 2011. Ban đầu, ReactJS được phát triển để xây dựng giao diện người dùng trên trang web Facebook, nhằm cải thiện tốc độ và hiệu suất của ứng dụng web.

Hiện nay, ReactJS đã trở thành một trong những thư viện phát triển web phổ biến nhất, được sử dụng rộng rãi bởi các công ty lớn và nhỏ trên toàn thế giới. Facebook cũng tiếp tục đầu tư phát triển và nâng cấp ReactJS để đáp ứng nhu cầu của cộng đồng phát triển.

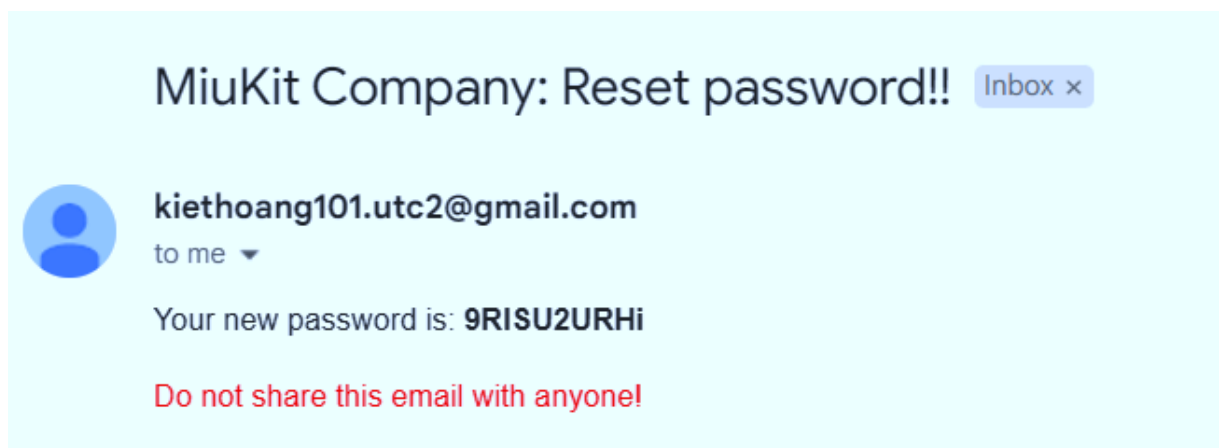
4.2. Giao diện hệ thống



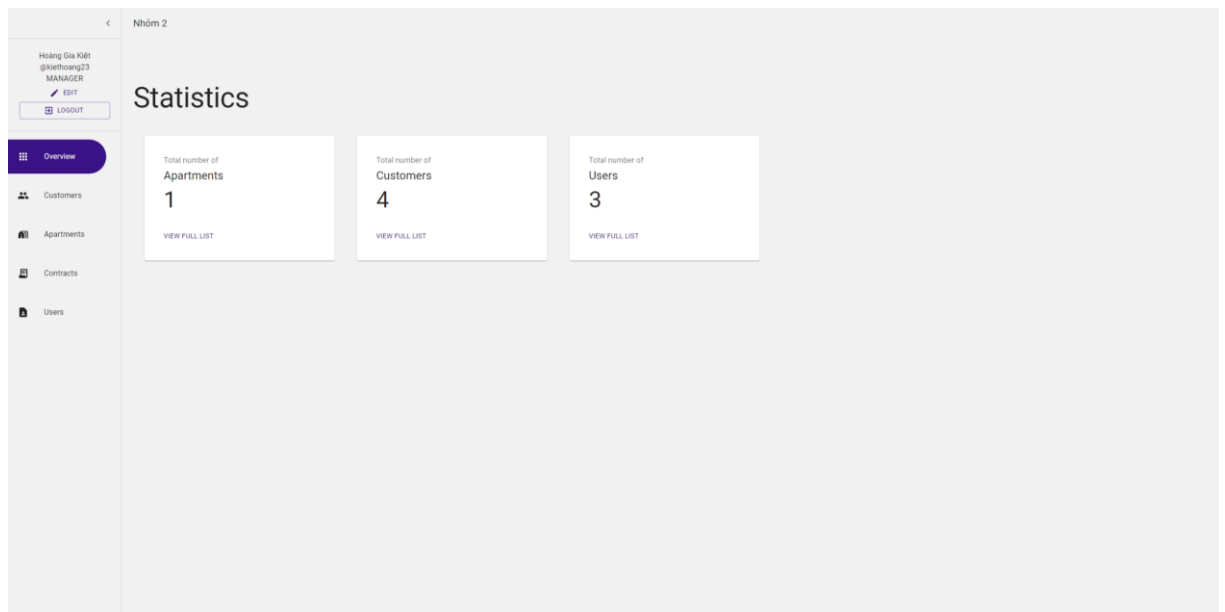
Hình 4.1: Giao diện đăng nhập



Hình 4.2: Giao diện quên mật khẩu



Hình 4.3: Giao diện gửi mật khẩu mới qua email

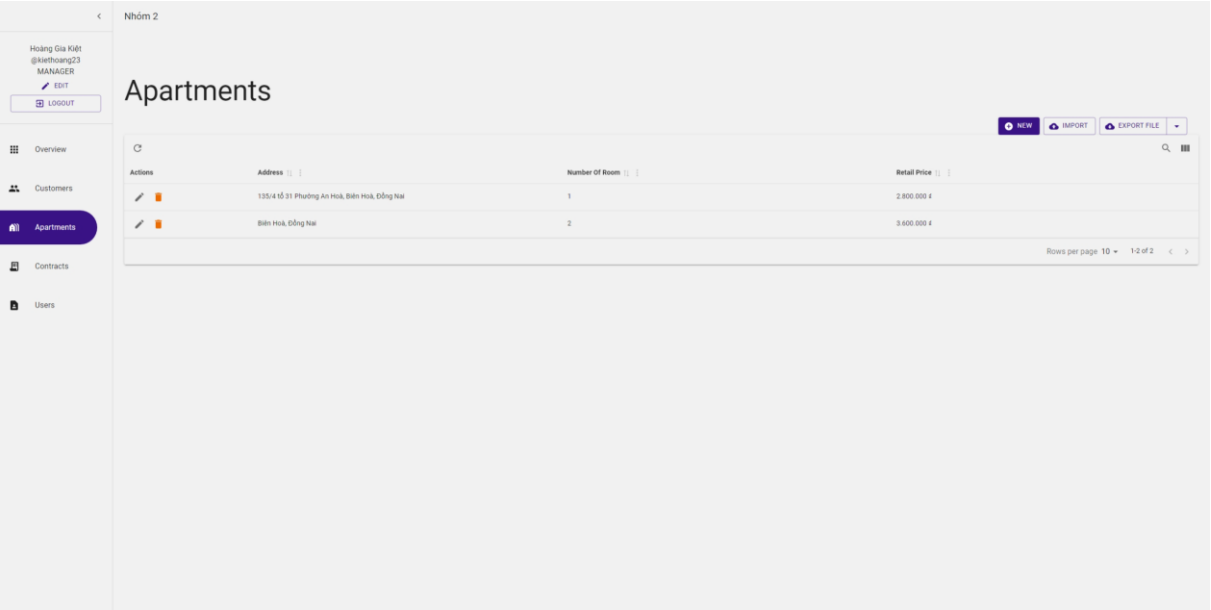


Hình 4.4: Giao diện trang chủ

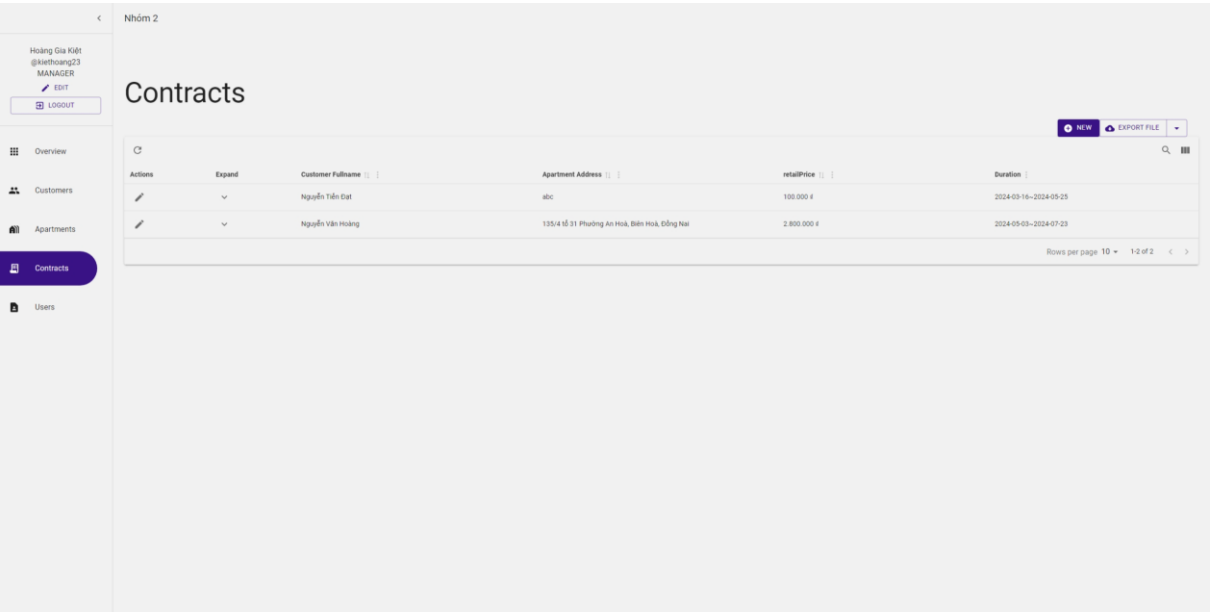
The screenshot shows a "Customers" management interface titled "Nhóm 2". The sidebar is similar to the dashboard but highlights the "Customers" section. The main content area has a "NEW" button and "IMPORT" and "EXPORT FILE" buttons. Below these is a table with columns: "Actions", "Full name", "Citizen Id", and "Phone Number". The table contains four rows of customer data. At the bottom right, there is a pagination control showing "Rows per page 10" and "1-4 of 4".

Actions	Full name	Citizen Id	Phone Number
	Hoàng Gia Kiệt	012345678901	0372234122
	Hoàng Gia Kiệt	123123123321	0784265174
	Nguyễn Văn Hoàng	123123123123	0342232141
	Nguyễn Văn Dũng	123123123124	0324123254

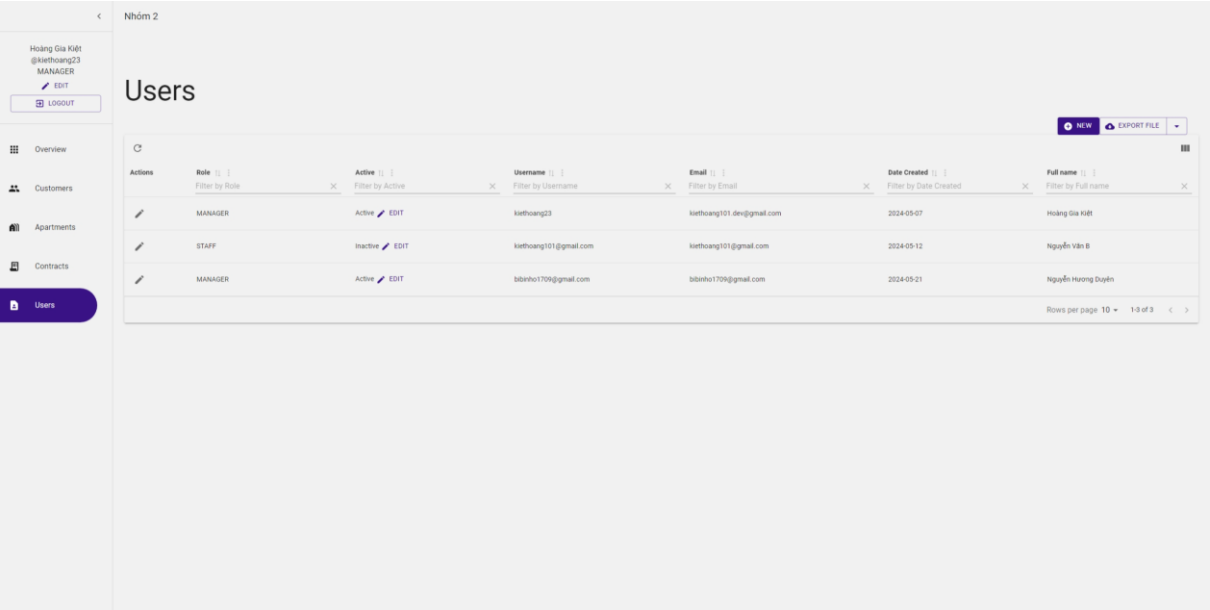
Hình 4.5: Giao diện quản lý khách hàng



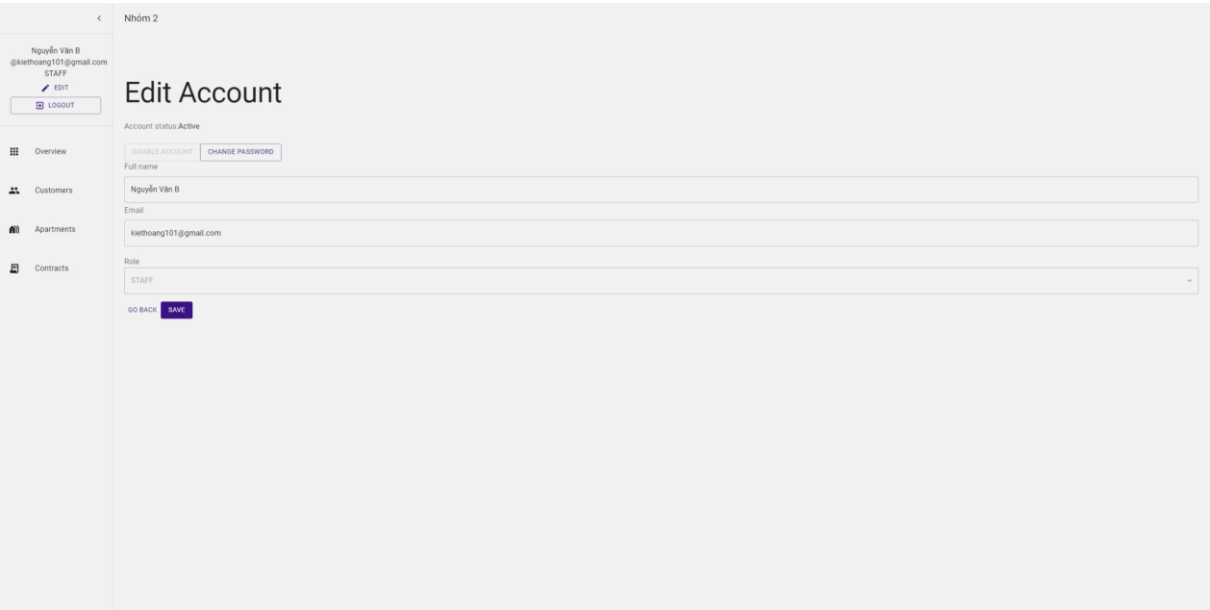
Hình 4.6: Giao diện quản lý căn hộ



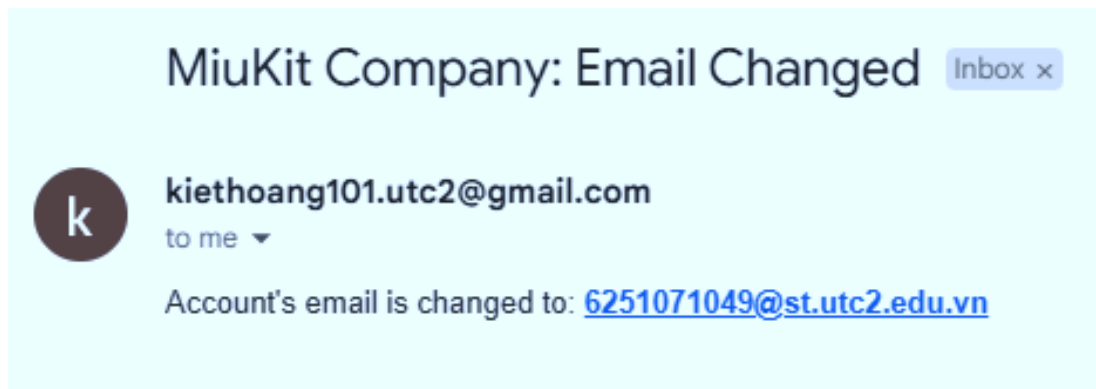
Hình 4.7: Giao diện quản lý hợp đồng



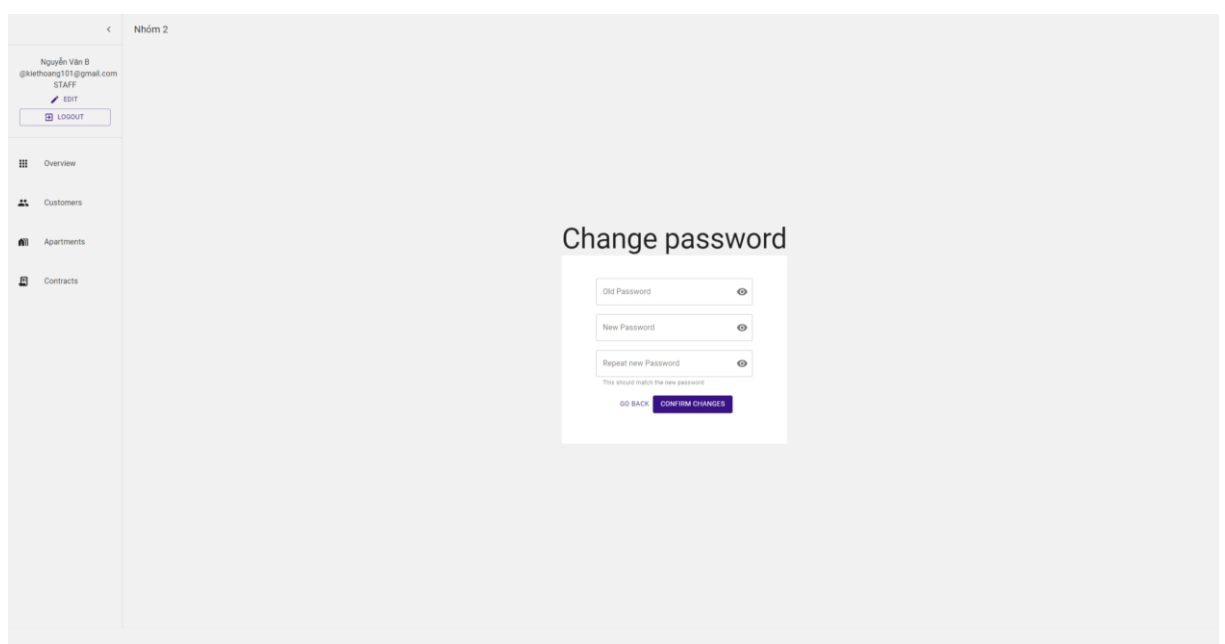
Hình 4.8: Giao diện quản lý người dùng



Hình 4.9: Giao diện tài khoản



Hình 4.10: Giao diện sau khi đổi mail mới



Hình 4.11: Giao diện đổi mật khẩu

KẾT LUẬN

1. Kết quả đạt được

- Quản lý khách hàng.
- Quản lý căn hộ.
- Quản lý hợp đồng.
- Quản lý người dùng.
- Thống kê
- Xác thực và phân quyền người dùng.
- Import/Export danh sách dữ liệu.

2. Hạn chế

- Giao diện còn chưa thân thiện với người dùng.
- Xử lý Backend chưa tối ưu.
- Chưa deploy được website.

3. Hướng phát triển

Trong thời gian tới, em sẽ cố gắng khắc phục những hạn chế trên và mở rộng thêm các tính năng mới của chương trình như:

- Thêm nhiều tính năng
- Tối ưu giao diện
- Tối ưu Backend
- Deploy website

TÀI LIỆU THAM KHẢO

- [1] iviettech, "iviettech," iviettech, 30 06 2014. [Online]. Available: <https://iviettech.vn/blog/302-ooduml.html>. [Accessed 17 04 2024].
- [2] H. V. Trinh, "Viblo," Viblo, 28 03 2016. [Online]. Available: <https://viblo.asia/p/phan-tich-thiet-ke-he-thong-thong-tin-su-dung-bieu-do-uml-phan-1-PjxMe6yNG4YL>. [Accessed 20 4 2024].
- [3] N. Thuý, "FPTShop," 28 02 2024. [Online]. Available: <https://fptshop.com.vn/tin-tuc/danh-gia/phan-mem-intellij-idea-176513s>. [Accessed 05 05 2024].
- [4] NhanHoa, "NhanHoa," 08 04 2021. [Online]. Available: <https://nhanhoa.com/tin-tuc/mysql-la-gi.html>. [Accessed 07 05 2024].