

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG



ĐỒ ÁN TỐT NGHIỆP

NGÀNH : CÔNG NGHỆ THÔNG TIN

Sinh viên : Đào Anh Ngọc

GVHD : Th.S Đỗ Văn Tuyên

HẢI PHÒNG – 2024

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG

**NGHIÊN CỨU XÂY DỰNG PHẦN MỀM QUẢN LÝ
KHÁCH SẠN SINH VIÊN(PHÂN HỆ TÀI CHÍNH)**

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY
NGÀNH: CÔNG NGHỆ THÔNG TIN

Sinh viên : Đào Anh Ngọc

GVHD : Th.S Đỗ Văn Tuyên

HẢI PHÒNG – 2024

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG

NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP

Sinh viên: Đào Anh Ngọc

Mã SV: 2012111005

Lớp : CT2401C

Ngành : Công nghệ thông tin

Tên đề tài: Xây dựng ứng dụng nền tảng web quản lý tài chính khu khách sạn sinh viên.

NHIỆM VỤ ĐỀ TÀI

1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp

a. Mô tả tóm tắt đề tài

Tìm hiểu về bài toán, phân tích thiết kế hệ thống và xây dựng ứng dụng nền tảng web quản lý tài chính khu khách sạn sinh viên.

b. Nội dung hướng dẫn

- Tìm hiểu về hiện trạng và bài toán quản lý tài chính khu khách sạn sinh viên.
- Phân tích, thiết kế cơ sở dữ liệu, hệ thống.
- Cài đặt ứng dụng và thử nghiệm.
- Nhận xét, đánh giá và kết luận.

c. Kết quả cần đạt được

- Tài liệu mô tả các kết quả đã thực hiện.
- Ứng dụng nền tảng web quản lý tài chính khu khách sạn sinh viên.

2. Các tài liệu, số liệu cần thiết

- Tài liệu tham khảo về hệ quản trị cơ sở dữ liệu, ngôn ngữ lập trình web
- Tài liệu tham khảo về phân tích và thiết kế hệ thống thông tin
- Một số hệ thống quản lý khu ký túc xá cho sinh viên.

3. Địa điểm thực tập tốt nghiệp

- Trường Đại học Quản lý và Công nghệ Hải Phòng

CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Họ và tên :

Học hàm, học vị :

Cơ quan công tác :

Nội dung hướng dẫn:

- Tìm hiểu về bài toán
- Thực hiện phân tích thiết kế hệ thống
- Tìm hiểu một hệ quản trị cơ sở dữ liệu, một ngôn ngữ lập trình web để xây dựng hệ thống.
- Cài đặt ứng dụng và thử nghiệm.
- Nhận xét, đánh giá và kết luận

Kết quả cần đạt được

- Tài liệu mô tả các kết quả đã thực hiện
- Ứng dụng nền tảng web quản lý tài chính khu khách sạn sinh viên.

Đề tài tốt nghiệp được giao ngày tháng năm 2024

Yêu cầu phải hoàn thành xong trước ngày tháng năm 2024

Đã nhận nhiệm vụ ĐTTN

Sinh viên

Đã giao nhiệm vụ ĐTTN

Giảng viên hướng dẫn

Th.S Đỗ Văn Tuyên

Hải Phòng, ngày tháng năm 2024

TRƯỞNG KHOA

CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN TỐT NGHIỆP

Họ và tên giảng viên:

Đơn vị công tác:

Họ và tên sinh viên: Đào Anh Ngọc

Ngành: Công nghệ Thông tin

Nội dung hướng dẫn:

- Tìm hiểu về bài toán
- Thực hiện phân tích thiết kế hệ thống
- Tìm hiểu hệ quản trị cơ sở dữ liệu, ngôn ngữ lập trình web để xây dựng ứng dụng nền tảng web quản lý tài chính khu khách sạn sinh viên.
- Nhận xét, đánh giá và kết luận

1. Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp

2. Đánh giá chất lượng của đồ án/khóa luận (so với nội dung yêu cầu đó đề ra trong nhiệm vụ Đ.T.T.N trên các mặt lý luận, thực tiễn, tính toán số liệu...)

3. Ý kiến của giảng viên hướng dẫn tốt nghiệp

Đạt ☐ Không đạt ☐ Điểm:

Hải Phòng, ngày tháng năm 2024.

Giảng viên hướng dẫn

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN CHẤM PHẢN BIỆN

Họ và tên giảng viên:

Đơn vị công tác:

Họ và tên sinh viên: Ngành: Công nghệ thông tin

Đề tài tốt nghiệp:

1. Phần nhận xét của giảng viên chấm phản biện

.....

.....

.....

.....

.....

.....

2. Những mặt còn hạn chế

.....

.....

.....

.....

.....

.....

3. Ý kiến của giảng viên chấm phản biện

Được bảo vệ ☐ Không được bảo vệ ☐ Điểm:.....

Hải Phòng, ngày.....tháng năm 2024

Giảng viên chấm phản biện

(Ký và ghi rõ họ tên)

LỜI CẢM ƠN

Với lòng kính trọng và biết ơn chân thành, em xin được gửi lời cảm ơn sâu sắc tới thầy Đỗ Văn Tuyên đã định hướng và hướng dẫn em tận tình trong suốt quá trình nghiên cứu để em hoàn thành được bài tiểu luận tốt nghiệp này.

Em xin chân thành cảm ơn các thầy cô trong Khoa Công nghệ thông tin của Trường Đại học Quản Lý và Công Nghệ Hải Phòng đã hết lòng giúp đỡ, tạo điều kiện cho em được học hỏi và trao dồi kiến thức, kỹ năng để hoàn thành bài tiểu luận tốt nghiệp.

Trong quá trình thực hiện tiểu luận tốt nghiệp, mặc dù đã cố gắng nhưng khó tránh khỏi những thiếu sót, em rất mong nhận được ý kiến đóng góp của các thầy cô giáo để bài tiểu luận tốt nghiệp của em thêm hoàn thiện.

Em xin chân thành cảm ơn!

Hải Phòng, ngày tháng năm 2024.

Sinh viên

LỜI CAM ĐOAN

Em xin cam đoan rằng công trình nghiên cứu này là của riêng em, được thực hiện dưới sự hướng dẫn của thầy Đỗ Văn Tuyên. Các nội dung và kết quả trong đề tài đều trung thực và chưa từng được công bố dưới bất kỳ hình thức nào trước đây.

Em xin chịu hoàn toàn trách nhiệm về mọi gian lận (nếu có) trong nội dung bài tiểu luận. Trường Đại học Quản lý và Công nghệ Hải Phòng không chịu trách nhiệm về bất kỳ vi phạm tác quyền hay bản quyền nào do em gây ra trong quá trình thực hiện đề tài này.

Hải Phòng, ngày tháng năm 2024.

Sinh viên

MỤC LỤC

LỜI CẢM ƠN.....	v
LỜI CAM ĐOAN	vi
MỤC LỤC.....	vii
DANH MỤC VIẾT TẮT.....	ix
DANH SÁCH HÌNH VẼ.....	x
DANH MỤC BẢNG.....	xiii
MỞ ĐẦU	1
CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI.....	4
1.1. Hiện trạng quản lý khách sạn sinh viên.....	4
1.2. Phát biểu bài toán	5
1.3. Giải pháp	5
1.4. Yêu cầu đạt được của hệ thống	5
1.5. Giới thiệu các công nghệ được sử dụng trong chương trình	6
CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG	15
2.1. Khảo sát hệ thống.....	15
2.1.1. Cách thức khảo sát.....	15
2.1.2. Phân tích yêu cầu hệ thống	15
2.1.3. Quy trình nghiệp vụ chung	16
2.2. Sơ đồ phân rã chức năng	35
2.3. Biểu đồ Usecase	36
2.3.1. Biểu đồ Use Case tổng quát	36
2.3.2. Biểu đồ Use Case chi tiết	37
2.4. Đặc tả User Case	38
2.4.1. Use Case “Tạo tài khoản”	38
2.4.2. Use Case “Phân quyền”	38
2.4.3. Use Case “Thêm mới đơn giá nước”	38
2.4.4. Use Case “Sửa đơn giá nước”	39
2.4.5. Use Case “Thêm mới đơn giá điện”	39
2.4.6. Use Case “Sửa đơn giá điện”	40
2.4.7. Use Case “Đăng nhập”	40
2.4.8. Use Case “Nhập chỉ số điện, nước”	41
2.4.9. Use Case “Sửa chỉ số điện, nước”	41
2.4.10. Use Case “Thông báo hoá đơn hàng tháng cho người thuê phòng”	42

2.5.	Biểu đồ tuần tự	43
2.6.	Biểu đồ cộng tác thực thi.....	53
2.7.	Mô hình quan hệ.....	58
CHƯƠNG 3. LẬP TRÌNH VÀ THỬ NGHIỆM.....		61
3.1.	Xây dựng cơ sở dữ liệu	61
3.2.	Tạo API (Application Programming Interface).....	66
3.3.	Xây dựng giao diện	66
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN		73
TÀI LIỆU THAM KHẢO		74

DANH MỤC VIẾT TẮT

DANH SÁCH HÌNH VẼ

Hình 2.1.1: Sơ đồ tiến trình nghiệp vụ đăng nhập của người dùng hệ thống

Hình 2.1.2: Sơ đồ tiến trình nghiệp vụ “Tạo tài khoản”

Hình 2.1.3: Sơ đồ tiến trình nghiệp vụ “Phân quyền tài khoản”

Hình 2.1.4: Sơ đồ tiến trình nghiệp vụ “Thêm đơn giá nước”

Hình 2.1.5: Sơ đồ tiến trình nghiệp vụ “Sửa đơn giá nước”

Hình 2.1.6: Sơ đồ tiến trình nghiệp vụ “Thêm đơn giá điện”

Hình 2.1.7: Sơ đồ tiến trình nghiệp vụ “Sửa đơn giá điện”

Hình 2.1.8: Sơ đồ tiến trình nghiệp vụ “Tạo tài khoản”

Hình 2.1.9: Sơ đồ tiến trình nghiệp vụ “Thông báo hoá đơn tháng cho người thuê phòng”

Hình 2.1.10: Sơ đồ tiến trình nghiệp vụ “Sửa chỉ số điện, nước tháng”

Hình 2.2.1: Sơ đồ phân rã chức năng

Hình 2.3.1: Usecase tổng quát.

Hình 2.3.2: Usecase chi tiết

Hình 2.5.1: Biểu đồ tuần tự thực thi Use Case "Đăng nhập"

Hình 2.5.2: Biểu đồ tuần tự thực thi Use Case "Tạo tài khoản"

Hình 2.5.3: Biểu đồ tuần tự thực thi Use Case "Phân quyền"

Hình 2.5.4: Biểu đồ tuần tự thực thi Use Case "Thêm mới đơn giá nước"

Hình 2.5.5: Biểu đồ tuần tự thực thi Use Case "Sửa đơn giá nước"

Hình 2.5.6: Biểu đồ tuần tự thực thi Use Case "Thêm mới đơn giá điện"

Hình 2.5.7: Biểu đồ tuần tự thực thi Use Case "Sửa đơn giá điện"

Hình 2.5.8: Biểu đồ tuần tự thực thi Use Case "Nhập chỉ số điện nước"

Hình 2.5.9: Biểu đồ tuần tự thực thi Use Case "Sửa chỉ số điện nước"

Hình 2.5.10: Biểu đồ tuần tự thực thi Use Case "Thông báo hoá đơn tháng"

Hình 2.6.1: Biểu đồ cộng tác thực thi Use Case "Đăng nhập"

Hình 2.6.2: Biểu đồ cộng tác thực thi Use Case "Tạo tài khoản"

Hình 2.6.3: Biểu đồ cộng tác thực thi Use Case "Phân quyền"

Hình 2.6.4: Biểu đồ cộng tác thực thi Use Case "Thêm đơn giá nước"

Hình 2.6.5: Biểu đồ cộng tác thực thi Use Case "Sửa đơn giá nước"

Hình 2.6.6: Biểu đồ cộng tác thực thi Use Case "Thêm đơn giá điện"

Hình 2.6.7: Biểu đồ cộng tác thực thi Use Case "Sửa đơn giá điện"

Hình 2.6.8: Biểu đồ cộng tác thực thi Use Case "Nhập chỉ số điện, nước"

Hình 2.6.9: Biểu đồ cộng tác thực thi Use Case "Sửa chỉ số điện, nước"

Hình 2.6.10: Biểu đồ cộng tác thực thi Use Case "Thông báo hoá đơn tháng"

Hình 2.7.1: Mô hình quan hệ giữa người dùng và quyền

Hình 2.7.2: Mô hình quan hệ giữa khu vực, phòng, hợp đồng thuê nhà, hoá đơn tháng

Hình 2.7.3: Mô hình quan hệ giữa người thuê, loại người thuê, giới tính, hợp đồng thuê phòng

Hình 3.1: Mô hình tổng quát của hệ thống

Hình 3.1.1: Bảng danh mục khu vực

Hình 3.1.2: Bảng danh mục giới tính

Hình 3.1.3: Bảng danh mục phòng

Hình 3.1.4: Bảng danh mục loại người thuê

Hình 3.1.5: Bảng đơn giá điện

Hình 3.1.6: Bảng đơn giá nước

Hình 3.1.7: Bảng người thuê

Hình 3.1.8: Bảng hợp đồng thuê

Hình 3.1.9: Bảng hợp hoá đơn hàng tháng

Hình 3.1.10: Bảng danh mục quyền

Hình 3.1.11: Bảng người dùng

Hình 3.1.12: Bảng quyền của người dùng

Hình 3.2.1: Mô hình hoạt động của hasura

Hình 3.3.1: Giao diện đăng nhập

Hình 3.3.2: Tạo tài khoản người dùng

Hình 3.3.3: Giao diện phân quyền

Hình 3.3.4: Giao diện phân quyền

Hình 3.3.5: Giao diện đơn giá điện

Hình 3.3.6: Giao diện đơn giá nước

Hình 3.3.7: Giao diện chỉ số điện nước

Hình 3.3.8: Giao diện chỉnh sửa chỉ số điện nước

Hình 3.3.9: Giao diện hoá đơn hàng tháng

Hình 3.3.10: Giao diện hoá đơn hàng tháng chi tiết

Hình 3.3.11: Giao diện email thông báo hoá đơn hàng tháng cho người thuê

DANH MỤC BẢNG

Bảng 2.4.1: Use Case “Tạo tài khoản”

Bảng 2.4.2: Use Case “Phân quyền”

Bảng 2.4.3: Use Case “Thêm mới đơn giá nước”

Bảng 2.4.4: Use Case “Sửa đơn giá nước”

Bảng 2.4.5: Use Case “Thêm mới đơn giá điện”

Bảng 2.4.6: Use Case “Sửa đơn giá điện”

Bảng 2.4.7: Use Case “Đăng nhập”

Bảng 2.4.8: Use Case “Nhập chỉ số điện, nước”

Bảng 2.4.9: Use Case “Sửa chỉ số điện, nước”

Bảng 2.4.10: Use Case “Thông báo hoá đơn tháng đến người thuê”

MỞ ĐẦU

1. Lý do chọn đề tài

Quản lý tài chính cho khu ký túc xá sinh viên đòi hỏi sự tổ chức và minh bạch để đảm bảo hoạt động hàng ngày diễn ra một cách hiệu quả và linh hoạt. Việc quản lý các khoản thu và chi, bao gồm tiền phòng, tiền điện, tiền nước và các khoản chi phí khác, là một thách thức không nhỏ. Dự án này được tạo ra với mục đích giải quyết những thách thức này và cung cấp một hệ thống quản lý tài chính toàn diện và tiện lợi cho người quản lý ký túc xá.

2. Nội dung nghiên cứu

- Tìm hiểu bài toán: Nắm bắt và hiểu rõ các quy trình và yêu cầu quản lý tài chính cho khu ký túc xá sinh viên, bao gồm việc quản lý tiền phòng, tiền điện, tiền nước và các khoản phí khác.
- Phân tích thiết kế hệ thống: Xác định và thiết kế các chức năng cần thiết để quản lý và theo dõi các khoản thu và chi phí, cũng như tổ chức dữ liệu một cách có tổ chức và dễ quản lý.
- Tìm hiểu công nghệ: Sử dụng các công nghệ phù hợp để phát triển hệ thống, bao gồm ngôn ngữ lập trình, cơ sở dữ liệu và framework phát triển phần mềm.

3. Mục đích chọn đề tài

Mục đích của dự án là cung cấp một hệ thống quản lý tài chính toàn diện và tiện lợi cho người quản lý ký túc xá sinh viên. Hệ thống này sẽ giúp họ quản lý và theo dõi các khoản thu và chi phí một cách hiệu quả, bao gồm tiền phòng, tiền điện, tiền nước và các khoản phí khác, từ việc thu và ghi nhận thông tin đến việc phân tích và báo cáo kết quả.

4. Phương pháp nghiên cứu

Dự án sẽ tiến hành nghiên cứu bằng cách tổng hợp thông tin từ người quản lý ký túc xá và các bên liên quan, phân tích các quy trình và yêu cầu cụ thể, và áp dụng các kiến thức về thiết kế hệ thống và công nghệ web để xây dựng ứng dụng có tính ứng dụng cao và đáp ứng được nhu cầu thực tế.

❖ Về mặt lý thuyết

- Hiểu được cách thức hoạt động của Client – Server sử dụng ngôn ngữ Javascript

- Hiểu được hệ quản trị cơ sở dữ liệu PostgreSQL
- Hiểu được cách phân tích thiết kế hệ thống phần mềm

❖ Về mặt lập trình

- Sử dụng PostgreSQL, một hệ quản trị cơ sở dữ liệu mạnh mẽ và phổ biến, để lưu trữ và quản lý dữ liệu cho ứng dụng của bạn.

- Sử dụng Node.js và Hasura để tạo và quản lý API cho ứng dụng của bạn. Hasura là một platform cho phép bạn tạo ra các API dựa trên GraphQL một cách nhanh chóng và dễ dàng.

- Sử dụng Vite và ReactJS để phát triển giao diện người dùng của ứng dụng. Vite là một công cụ build nhanh và hiệu quả cho các ứng dụng web, trong khi ReactJS là một thư viện JavaScript phổ biến cho việc xây dựng giao diện người dùng tương tác.

- Sử dụng Tailwind CSS để thiết kế và tạo giao diện cho ứng dụng của bạn. Tailwind CSS là một framework CSS linh hoạt và dễ sử dụng, cho phép bạn tạo ra các giao diện đẹp mắt và hiệu quả một cách nhanh chóng.

- Clerk: Clerk cung cấp các công cụ xác thực và quản lý người dùng mạnh mẽ, giúp bạn dễ dàng tích hợp tính năng đăng nhập, đăng ký và quản lý tài khoản người dùng vào ứng dụng của mình mà không cần phải xây dựng lại từ đầu.

- Vercel: Vercel là một nền tảng hosting và triển khai phổ biến cho các ứng dụng web và API. Nó cung cấp các công cụ linh hoạt để triển khai ứng dụng React và Node.js của bạn một cách dễ dàng, kèm theo tính năng như auto-scaling và continuous deployment.

❖ Về mặt hoạt động

- Đảm bảo rằng website thực hiện đầy đủ các chức năng theo yêu cầu, bao gồm việc xử lý dữ liệu đầu vào từ người dùng và đưa ra các kết quả xử lý chính xác và đáng tin cậy.

5. Phạm vi, đối tượng nghiên cứu của đề tài

- Phạm vi: Hệ thống quản lý tài chính sẽ được thiết kế để hỗ trợ người quản lý ký túc xá trong việc quản lý và theo dõi các khoản thu và chi phí, bao gồm tiền phòng, tiền điện, tiền nước và các khoản phí khác.

- Đối tượng: Người quản lý ký túc xá và nhân viên liên quan có nhu cầu quản lý tài chính và theo dõi các khoản thu và chi phí.

CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

1.1. Hiện trạng quản lý khách sạn sinh viên

Hiện tại, việc quản lý các khoản thu và chi, bao gồm tiền phòng, tiền điện, tiền nước và các khoản chi phí khác, đang gặp nhiều khó khăn. Các quy trình quản lý tài chính thường được thực hiện thủ công hoặc bằng các phần mềm cũ kỹ, không đáp ứng được yêu cầu hiện đại. Điều này dẫn đến nhiều vấn đề như sai sót trong ghi chép, thiếu minh bạch trong quản lý tài chính, và mất nhiều thời gian và công sức cho việc nhập liệu và theo dõi. Trong thời đại công nghệ ngày nay, việc áp dụng các giải pháp công nghệ vào quản lý tài chính khu ký túc xá sinh viên trở nên cần thiết để tối ưu hoá quá trình này và đảm bảo tính chính xác, hiệu quả.

Cụ thể, một số khó khăn chính trong quản lý tài chính ký túc xá hiện tại bao gồm:

Sai sót trong ghi chép và theo dõi: Việc quản lý tài chính bằng tay hoặc qua các phần mềm cũ có thể dẫn đến nhiều sai sót trong ghi chép các khoản thu và chi, gây khó khăn trong việc theo dõi chính xác tình hình tài chính.

Thiếu minh bạch: Thông tin tài chính không được cập nhật kịp thời và rõ ràng, gây khó khăn cho việc kiểm tra và đối chiếu các khoản thu chi, làm giảm độ tin cậy trong quản lý tài chính.

Tốn nhiều thời gian và công sức: Việc nhập liệu và theo dõi các khoản thu chi bằng phương pháp thủ công hoặc các phần mềm không hiệu quả mất nhiều thời gian và công sức của người quản lý, làm giảm hiệu suất công việc.

Khó khăn trong việc tổng hợp và báo cáo: Các phần mềm cũ thường thiếu tính năng tổng hợp và tạo báo cáo tự động, gây khó khăn cho việc tạo ra các báo cáo tài chính chi tiết và chính xác.

Không tích hợp thanh toán trực tuyến: Việc thiếu các phương thức thanh toán trực tuyến gây bất tiện cho sinh viên trong việc thực hiện các giao dịch tài chính, dẫn đến tình trạng chậm trễ trong thanh toán và quản lý nợ.

1.2. Phát biểu bài toán

Bài toán mà dự án này đặt ra là làm thế nào để giải quyết những thách thức nêu trên và cung cấp một hệ thống quản lý tài chính toàn diện và tiện lợi cho người quản lý ký túc xá. Phần mềm mới sẽ tự động hóa các quy trình quản lý tài chính, giảm thiểu sự can thiệp thủ công và sai sót, cung cấp báo cáo chi tiết và minh bạch, tiết kiệm thời gian và nguồn lực, và nâng cao trải nghiệm người dùng thông qua giao diện thân thiện.

1.3. Giải pháp

Để giải quyết bài toán trên, em đề xuất xây dựng một ứng dụng web sử dụng các công nghệ hiện đại để quản lý tài chính cho khu ký túc xá sinh viên. Ứng dụng này sẽ được thiết kế để dễ sử dụng và linh hoạt, đồng thời đảm bảo tính minh bạch và chính xác trong quản lý tài chính.

1.4. Yêu cầu đạt được của hệ thống

❖ Yêu cầu chung

- Trang chủ
- Cài đặt của hệ thống
- Quản lý tài chính

❖ Yêu cầu của các chức năng

- Trang chủ:
 - + Giới thiệu về hệ thống
- Cài đặt của hệ thống
 - + Tạo tài khoản cho người dùng
 - + Phân quyền cho tài khoản người dùng
 - + Cài đặt đơn giá điện: đơn giá của 1 số điện áp dụng trong 1 khoảng thời gian
 - + Cài đặt đơn giá nước: đơn giá của 1 số nước áp dụng trong 1 khoảng thời gian
- Quản lý tài chính

- + Nhập các chỉ số điện, nước hàng tháng
- + Tính toán số tiền hàng tháng của các phòng
- + Báo cáo về số tiền hàng tháng của các phòng
- + Thông báo cho sinh viên về số tiền hàng tháng

1.5. Giới thiệu các công nghệ được sử dụng trong chương trình

❖ Database4

- Database4 là một ứng dụng giao diện người dùng đồ họa (GUI) cho PostgreSQL, giúp bạn quản lý cơ sở dữ liệu PostgreSQL từ xa một cách thuận tiện và trực quan. Dưới đây là một số tính năng chính của Database4:

- + Khám phá cơ sở dữ liệu: Database4 cho phép bạn duyệt qua các cơ sở dữ liệu và các bảng, xem cấu trúc của chúng và tìm hiểu về dữ liệu một cách trực quan thông qua giao diện người dùng.
- + Thực thi truy vấn SQL: Bạn có thể tạo và thực thi các truy vấn SQL trực tiếp từ Database4. Điều này cho phép bạn thao tác với dữ liệu và cấu trúc cơ sở dữ liệu một cách linh hoạt và hiệu quả.
- + Xem và chỉnh sửa dữ liệu: Database4 cho phép bạn xem và chỉnh sửa dữ liệu trực tiếp từ giao diện người dùng. Điều này giúp bạn thực hiện các tác vụ quản lý dữ liệu một cách dễ dàng và nhanh chóng.
- + Quản lý chỉ mục: Bạn có thể xem và quản lý các chỉ mục trên các bảng của mình, và Database4 cung cấp các công cụ để đảm bảo rằng chỉ mục được sử dụng một cách hiệu quả.
- + Kết nối dễ dàng: Database4 cho phép bạn kết nối với cơ sở dữ liệu PostgreSQL của mình từ xa một cách dễ dàng thông qua các kết nối đã được thiết lập trước.

- Với Database4, bạn có thể quản lý và tương tác với cơ sở dữ liệu PostgreSQL của mình một cách tiện lợi và hiệu quả, giúp tối ưu hóa quá trình phát triển và quản lý dữ liệu của bạn.

❖ PostgreSQL

- PostgreSQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở, phát triển và duy trì bởi một cộng đồng toàn cầu của các nhà phát triển. Đây là một trong những hệ quản trị cơ sở dữ liệu (HDCSDL) phổ biến nhất, nổi tiếng với tính mở rộng, độ tin cậy cao, và tính năng mạnh mẽ. Dưới đây là một số điểm nổi bật về PostgreSQL:

- + **Quản lý dữ liệu quan hệ:** PostgreSQL cho phép lưu trữ và quản lý dữ liệu quan hệ thông qua bảng và mối quan hệ giữa chúng, giúp tổ chức dữ liệu một cách logic và có tổ chức.
- + **Tính ACID:** PostgreSQL hỗ trợ các tính chất ACID (Atomicity, Consistency, Isolation, Durability), đảm bảo tính toàn vẹn và nhất quán của dữ liệu trong quá trình thực hiện các thao tác.
- + **Hỗ trợ đa ngôn ngữ:** PostgreSQL hỗ trợ nhiều ngôn ngữ lập trình như SQL, Python, Java, Perl, C/C++, và nhiều ngôn ngữ khác, giúp dễ dàng tích hợp với các ứng dụng và dịch vụ khác.
- + **Tính mở rộng và mạnh mẽ:** PostgreSQL có khả năng mở rộng từ các cấu hình đơn giản đến các cấu hình phức tạp với hàng trăm máy chủ. Nó cung cấp nhiều tính năng mạnh mẽ như truy vấn phức tạp, định nghĩa hàm, và giao dịch song song.
- + **Bảo mật cao:** PostgreSQL có các tính năng bảo mật như quản lý người dùng và phân quyền, SSL/TLS để mã hóa dữ liệu trong quá trình truyền tải, và khả năng kiểm soát truy cập dựa trên địa chỉ IP.
- + **Cộng đồng lớn và hỗ trợ chuyên sâu:** PostgreSQL có một cộng đồng phát triển rộng lớn và hoạt động tích cực, cung cấp tài liệu, hướng dẫn và hỗ trợ từ các chuyên gia trong lĩnh vực.

- Tóm lại, PostgreSQL là một HDCSDL mạnh mẽ và linh hoạt, phù hợp cho các ứng dụng từ nhỏ đến lớn, từ đơn giản đến phức tạp. Đặc tính mở rộng, tính bảo mật cao và tính toàn vẹn dữ liệu là những yếu tố quan trọng khiến PostgreSQL trở thành một lựa chọn phổ biến trong cộng đồng phát triển phần mềm.

❖ Hasura

- Hasura là một nền tảng phát triển ứng dụng web và dịch vụ GraphQL mã nguồn mở. Nó cung cấp một cách nhanh chóng và dễ dàng để xây dựng các API GraphQL từ các cơ sở dữ liệu có sẵn, đặc biệt là PostgreSQL. Dưới đây là một số điểm nổi bật về Hasura:

- + **Tích hợp nhanh chóng với cơ sở dữ liệu:** Hasura cho phép bạn nhanh chóng tạo ra một lớp API GraphQL từ các cơ sở dữ liệu đã có mà không cần viết mã nhiều.
- + **Tự động tạo schema GraphQL:** Hasura tự động phát hiện cấu trúc của cơ sở dữ liệu và tạo ra một schema GraphQL tương ứng, giúp bạn tiết kiệm thời gian và công sức.
- + **Phát triển nhanh chóng và hiệu quả:** Với Hasura, bạn có thể phát triển ứng dụng web nhanh chóng và hiệu quả hơn nhờ vào việc sử dụng GraphQL làm giao diện lập trình ứng dụng.
- + **Bảo mật và quản lý truy cập:** Hasura cung cấp các tính năng bảo mật như xác thực người dùng, quản lý quyền truy cập và kiểm soát truy cập dựa trên vai trò, giúp bảo vệ dữ liệu của bạn.
- + **Tích hợp dễ dàng:** Hasura có thể tích hợp với nhiều công nghệ và dịch vụ khác nhau như Docker, Kubernetes, Auth0, Firebase, và nhiều hơn nữa.
- + **Cộng đồng phát triển mạnh mẽ:** Hasura có một cộng đồng phát triển lớn và tích cực, cung cấp sự hỗ trợ, tài liệu và các tài nguyên hữu ích cho người dùng.

- Tóm lại, Hasura là một công cụ mạnh mẽ cho việc phát triển ứng dụng web dựa trên GraphQL, giúp bạn nhanh chóng xây dựng API từ cơ sở dữ liệu có sẵn một cách dễ dàng và hiệu quả.

❖ Nodejs

- Node.js là một nền tảng mã nguồn mở dựa trên JavaScript, được xây dựng dựa trên runtime JavaScript của Chrome (V8 Engine) để phát triển các ứng dụng mạng và máy chủ. Dưới đây là một số điểm nổi bật về Node.js:

- + Linh hoạt và hiệu quả: Node.js cho phép viết mã JavaScript để xây dựng cả máy chủ và phần mềm máy khách, tạo điều kiện cho việc chia sẻ mã giữa các phần của ứng dụng.
- + Non-blocking và Asynchronous I/O: Node.js sử dụng mô hình không chặn và I/O không đồng bộ, giúp xử lý hàng loạt yêu cầu mà không cần tạo ra các luồng mới, tăng hiệu suất và mở rộng.
- + Package ecosystem mạnh mẽ: Node.js có một hệ sinh thái mô-đun (module) phong phú trên npm (Node Package Manager), cho phép bạn dễ dàng cài đặt, quản lý và chia sẻ mã nguồn mở.
- + Xây dựng ứng dụng thời gian thực: Với khả năng xử lý sự kiện trên máy chủ và trong các ứng dụng mạng thời gian thực, Node.js thích hợp cho việc xây dựng các ứng dụng trò chơi, trò chuyện và ứng dụng phát trực tiếp.
- + JavaScript Everywhere: Node.js cho phép sử dụng cùng một ngôn ngữ lập trình (JavaScript) trên cả máy chủ và máy khách, giúp giảm thời gian và công sức cần thiết cho việc phát triển ứng dụng.
- + Cộng đồng phát triển lớn mạnh: Node.js có một cộng đồng phát triển lớn và tích cực, cung cấp sự hỗ trợ, tài liệu và các tài nguyên hữu ích cho người dùng.

- Tóm lại, Node.js là một nền tảng mạnh mẽ và linh hoạt, được sử dụng rộng rãi cho việc xây dựng các ứng dụng mạng và máy chủ hiệu quả và mở rộng. Với khả năng xử lý không chặn, hệ sinh thái mô-đun phong phú và cộng đồng hỗ trợ lớn, Node.js là một công cụ mạnh mẽ cho các nhà phát triển.

❖ **ReactJSX**

- ReactJSX là một phần của React, một thư viện JavaScript mã nguồn mở được phát triển bởi Facebook, được sử dụng để xây dựng giao diện người dùng cho các ứng dụng web và di động. ReactJSX là một phần mở rộng của JavaScript, cho phép bạn viết mã HTML tương tự như cú pháp HTML bên trong mã JavaScript.

- Dưới đây là một số điểm nổi bật về ReactJSX:

- + Tích hợp dễ dàng: ReactJSX cho phép bạn viết mã HTML trực tiếp trong các thành phần React, làm cho việc phát triển giao diện người dùng trở nên dễ dàng và rõ ràng hơn.
- + JavaScript expressions: Bạn có thể sử dụng biểu thức JavaScript bên trong ReactJSX để định nghĩa và tính toán nội dung của các phân tử HTML.
- + Components-based approach: ReactJSX hỗ trợ việc phát triển giao diện người dùng theo cách tiếp cận dựa trên các thành phần (components), cho phép tái sử dụng mã và tạo ra một cấu trúc ứng dụng có tổ chức và dễ bảo trì.
- + Tích hợp với JavaScript: ReactJSX được tích hợp chặt chẽ với JavaScript, giúp tạo ra các ứng dụng mạnh mẽ và linh hoạt, và cho phép sử dụng tất cả các tính năng của ngôn ngữ này.
- + Công cụ hỗ trợ: Có nhiều công cụ và trình biên tập mã hỗ trợ ReactJSX, giúp tối ưu hóa quá trình phát triển và debug.
- Tóm lại, ReactJSX là một cú pháp mở rộng của JavaScript cho phép viết mã HTML trong các thành phần React, giúp tạo ra các ứng dụng web và di động linh hoạt, mạnh mẽ và dễ bảo trì.

❖ Vite

- Vite là một công cụ phát triển web tốc độ cao, được xây dựng bởi Evan You, người tạo ra Vue.js. Nó được thiết kế để cung cấp một quá trình phát triển web nhanh chóng và mượt mà bằng cách sử dụng ESM (ECMAScript Modules) natively, điều này giúp tận dụng sự nhanh chóng của các trình duyệt hiện đại trong việc tải và xử lý mã JavaScript.
- Dưới đây là một số điểm nổi bật về Vite:
 - + Tích hợp với Vue.js và React: Vite được tối ưu hóa cho việc phát triển các ứng dụng sử dụng Vue.js hoặc React, nhưng cũng có thể được sử dụng cho các dự án sử dụng các framework JavaScript khác.

- + Dev Server nhanh chóng: Vite sử dụng một Dev Server được xây dựng dựa trên ES Modules, giúp tạo ra một quá trình phát triển web mượt mà và nhanh chóng. Nó sử dụng các kỹ thuật như server-side rendering và hoạt động chéo để tối ưu hóa tải trang và tạo ra trải nghiệm phát triển tốt nhất.
 - + Hot Module Replacement (HMR) nhanh chóng: Vite tích hợp HMR để cập nhật các thay đổi mã nguồn của bạn mà không cần làm mới trang, giúp tiết kiệm thời gian phát triển và tăng hiệu suất làm việc.
 - + Build nhanh chóng: Vite tạo ra các bản dựng (builds) nhanh chóng và hiệu quả, tận dụng sức mạnh của ESM và công nghệ cache để tối ưu hóa quá trình này.
 - + Đa nhiệm (Multi-Tasking): Vite cho phép bạn chạy các công việc đồng thời, điều này có thể giúp tăng hiệu suất trong quá trình phát triển và xây dựng ứng dụng của bạn.
- Tóm lại, Vite là một công cụ phát triển web tối ưu và mạnh mẽ, được tạo ra để cung cấp trải nghiệm phát triển web nhanh chóng và mượt mà, đặc biệt là khi sử dụng với Vue.js hoặc React.

❖ Clerk

- Clerk là một dịch vụ quản lý xác thực và quyền truy cập cho ứng dụng web và di động. Nó cung cấp một cách tiếp cận dễ dàng và an toàn để xác thực người dùng, quản lý phiên làm việc và phân quyền cho các tài nguyên trong ứng dụng của bạn. Dưới đây là một số điểm nổi bật về Clerk:
- + Xác thực dựa trên API: Clerk cung cấp các API để sử dụng để xác thực người dùng, bao gồm đăng ký, đăng nhập, xác minh email, đặt lại mật khẩu và nhiều hơn nữa.
- + Quản lý phiên và trạng thái xác thực: Clerk giúp bạn quản lý phiên làm việc của người dùng, đảm bảo tính bảo mật và tính toàn vẹn của hệ thống xác thực.
- + Phân quyền và quản lý quyền truy cập: Clerk cho phép bạn quản lý quyền truy cập của người dùng vào các tài nguyên và tính năng

trong ứng dụng của bạn, đảm bảo rằng họ chỉ có thể truy cập vào những gì được phép.

- + Tích hợp linh hoạt: Clerk tích hợp dễ dàng với các framework và thư viện phổ biến như React, Vue.js, Angular và Node.js, giúp bạn tích hợp xác thực vào ứng dụng của mình một cách dễ dàng.
- + Tính an toàn và bảo mật: Clerk đảm bảo tính an toàn và bảo mật của dữ liệu người dùng thông qua các biện pháp bảo mật như mã hóa mật khẩu, xác thực hai yếu tố và quản lý phiên làm việc an toàn.
- + Tích hợp với các dịch vụ khác: Clerk có thể tích hợp với các dịch vụ lưu trữ dữ liệu như Firebase, MongoDB và MySQL, giúp bạn quản lý dữ liệu người dùng một cách linh hoạt và hiệu quả.

- Tóm lại, Clerk là một dịch vụ xác thực và quản lý quyền truy cập mạnh mẽ và linh hoạt, giúp bạn tích hợp xác thực an toàn và quản lý quyền truy cập vào ứng dụng của mình một cách dễ dàng và hiệu quả.

❖ Vercel

- Vercel là một nền tảng điện toán đám mây được thiết kế để tối ưu hóa quá trình phát triển, triển khai và quản lý ứng dụng web. Dưới đây là một số điểm nổi bật về Vercel:

- + Triển khai tự động: Vercel cung cấp quy trình triển khai tự động từ các kho lưu trữ git như GitHub, GitLab và Bitbucket. Bất kỳ thay đổi nào được push lên kho lưu trữ sẽ kích hoạt một quá trình triển khai mới mà không cần can thiệp thủ công.
- + Hosting tĩnh và độ mở rộng: Vercel cung cấp dịch vụ hosting tĩnh với độ mở rộng tự động, cho phép ứng dụng của bạn tự động mở rộng để đáp ứng nhu cầu của người dùng mà không cần phải lo lắng về việc quản lý máy chủ.
- + Hỗ trợ nhiều loại ứng dụng: Vercel hỗ trợ nhiều loại ứng dụng bao gồm các ứng dụng React, Angular, Vue.js, Node.js và nhiều nền tảng khác, giúp bạn dễ dàng triển khai các dự án khác nhau trên cùng một nền tảng.

- + Phân phối toàn cầu: Vercel sử dụng mạng lưới CDN toàn cầu để cung cấp tốc độ tải trang nhanh cho người dùng trên khắp thế giới, giảm thiểu độ trễ và cải thiện trải nghiệm người dùng.
 - + Quản lý môi trường: Vercel cung cấp các tính năng quản lý môi trường để quản lý biến môi trường, khóa API và các cấu hình khác dễ dàng từ bảng điều khiển.
 - + Liên kết với Vercel Edge Network: Vercel cung cấp sự tích hợp với Vercel Edge Network, giúp tối ưu hóa và bảo mật các ứng dụng web của bạn thông qua các tính năng như serverless functions và route-based caching.
- Tóm lại, Vercel là một nền tảng điện toán đám mây mạnh mẽ và linh hoạt, giúp tối ưu hóa quá trình phát triển và triển khai các ứng dụng web một cách nhanh chóng và dễ dàng.

❖ Visual Studio Code

- Visual Studio Code (VSCode) là một trình soạn thảo mã nguồn miễn phí và mã nguồn mở, phát triển bởi Microsoft từ năm 2015. Nhanh chóng trở thành một trong những công cụ phổ biến nhất trong cộng đồng phát triển phần mềm, VSCode mang lại những ưu điểm đặc trưng sau:

- + Đa ngôn ngữ: VSCode hỗ trợ nhiều ngôn ngữ lập trình như JavaScript, TypeScript, Python, Java, C++, và nhiều ngôn ngữ khác, làm cho nó trở thành sự lựa chọn hàng đầu cho các dự án đa ngôn ngữ.
- + Mở rộng linh hoạt: Với hệ thống mở rộng mạnh mẽ, người dùng có thể dễ dàng cài đặt các tiện ích mở rộng để tùy chỉnh trình biên tập theo nhu cầu cụ thể. Từ themes đến debuggers, VSCode cung cấp một loạt các tiện ích mở rộng từ cộng đồng.
- + IntelliSense và Debugging: VSCode cung cấp tính năng IntelliSense mạnh mẽ giúp hoàn thành mã tự động và gợi ý biến, hàm, và API từ ngôn ngữ đang sử dụng. Tích hợp tính năng debugging giúp người dùng dễ dàng gỡ lỗi mã nguồn.

- + Hỗ trợ Git tích hợp: VSCode tích hợp sẵn với Git, giúp người dùng quản lý phiên bản mã nguồn một cách hiệu quả với các tính năng như xem thay đổi, commit, pull, push và nhiều hơn nữa.
 - + Đa nền tảng và nhẹ nhàng: Tính sẵn có trên nhiều hệ điều hành như Windows, macOS và Linux, VSCode là một ứng dụng nhẹ nhàng, hoạt động mượt mà trên cả các máy tính có cấu hình thấp.
 - + Cộng đồng phát triển lớn mạnh: Với cộng đồng phát triển rộng lớn, VSCode cung cấp nguồn tài nguyên phong phú, hướng dẫn và hỗ trợ từ cộng đồng, giúp người dùng nhanh chóng giải quyết vấn đề và tối ưu hóa quy trình làm việc.
- Tóm lại, Visual Studio Code là một trình soạn thảo mã linh hoạt, mạnh mẽ và dễ sử dụng, được ưa chuộng bởi các nhà phát triển phần mềm trên toàn thế giới.

CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

2.1. Khảo sát hệ thống

2.1.1. Cách thức khảo sát

Trong quá trình khảo sát hệ thống, em tiến hành tìm hiểu cụ thể về quy trình quản lý tài chính hiện tại trong khu ký túc xá sinh viên(địa điểm cụ thể : Khu Khách sạn sinh viên trường Đại học Quản lý và Công nghệ Hải Phòng). Các bước bao gồm:

1. Phỏng vấn các bên liên quan: Em tiến hành phỏng vấn người quản lý ký túc xá, nhân viên quản lý tài chính và sinh viên để hiểu rõ hơn về các quy trình, thách thức và yêu cầu cụ thể.
2. Thu thập dữ liệu: Em sẽ thu thập thông tin từ các tài liệu hiện có như biên bản ghi chép, báo cáo tài chính và các hệ thống đã tồn tại.
3. Quan sát trực tiếp: Em tham gia quan sát trực tiếp các hoạt động và quy trình trong khu ký túc xá để nắm bắt thêm thông tin.

2.1.2. Phân tích yêu cầu hệ thống

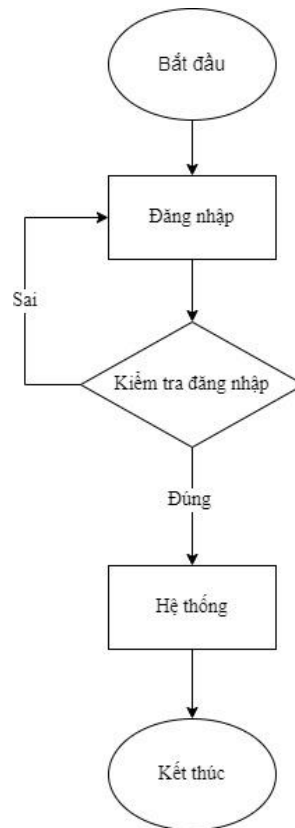
Dựa trên thông tin thu thập được quá trình khảo sát, em tiến hành phân tích yêu cầu hệ thống. Các yêu cầu này bao gồm:

1. Quản lý người dùng: Hệ thống cần có chức năng quản lý người dùng, bao gồm tạo tài khoản, phân quyền và quản lý thông tin cá nhân của người dùng.
2. Cài đặt hệ thống: Hệ thống cần cung cấp giao diện cho người quản trị để có thể cài đặt đơn giá điện, đơn giá nước và các cài đặt khác.
3. Quản lý tài chính: Hệ thống cần có chức năng nhập thông tin chỉ số điện, nước hàng tháng từ các phòng, tính toán số tiền hàng tháng của mỗi phòng và tạo báo cáo về số tiền hàng tháng của các phòng.

Các yêu cầu này sẽ được phân tích cụ thể hơn trong các phần tiếp theo của chương trình.

2.1.3. Quy trình nghiệp vụ chung

❖ Quy trình nghiệp vụ chung



Hình 2.1.1: Sơ đồ tiến trình nghiệp vụ đăng nhập của người dùng hệ thống

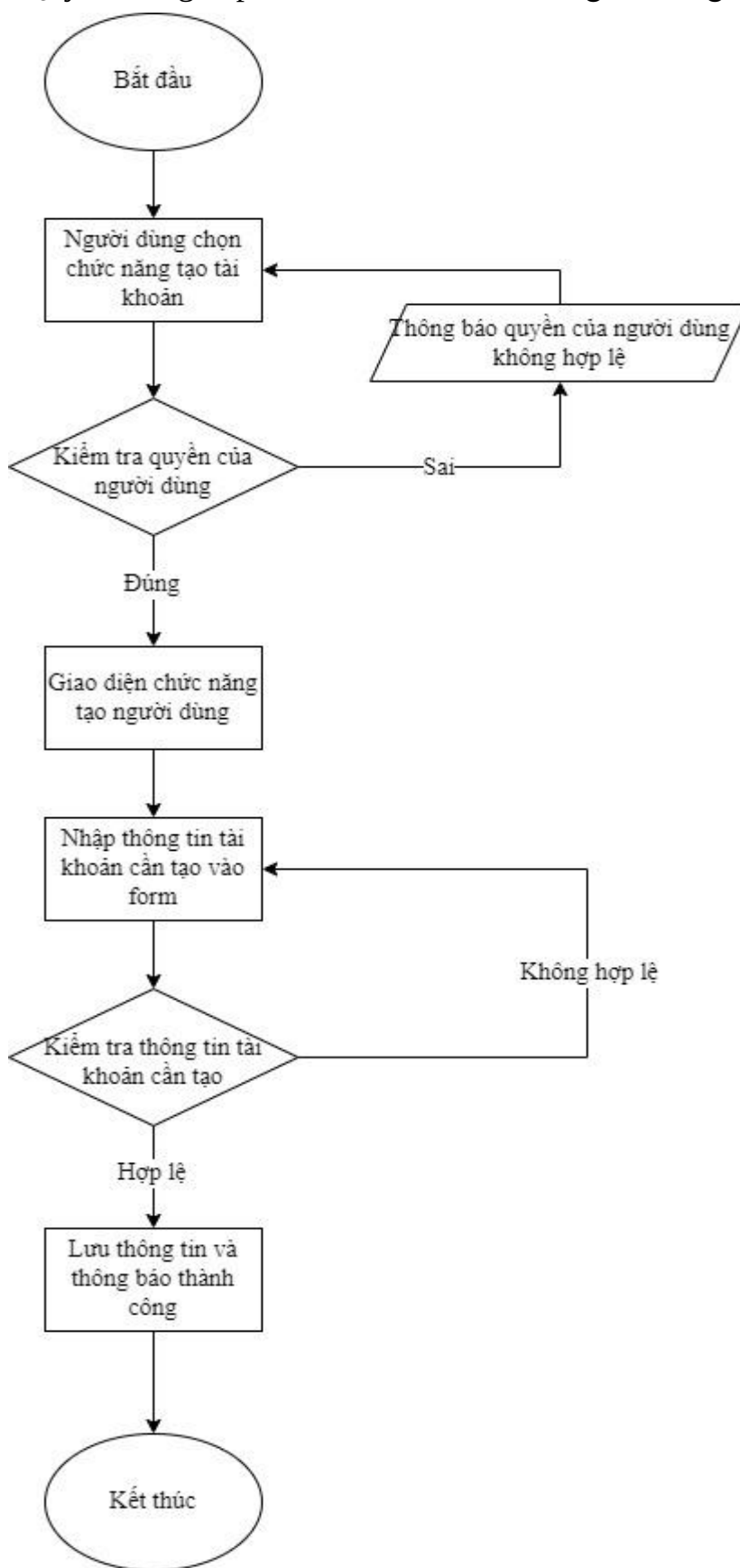
Người dùng hệ thống bao gồm 2 tác nhân chính : người quản trị hệ thống (admin) và người quản lý ký túc xá (manager).

Trong quá trình đăng nhập, người dùng sẽ nhập tên người dùng và mật khẩu vào trang đăng nhập. Sau đó hệ thống sẽ kiểm tra tên tài khoản và người dùng, nếu đăng nhập thành công sẽ được chuyển hướng vào trong hệ thống, nếu đăng nhập thất bại sẽ giữ nguyên ở trang đăng nhập và yêu cầu đăng nhập lại.

Chỉ sau khi đăng nhập thành công vào hệ thống mới có thể sử dụng tất cả các chức năng của hệ thống.

❖ Quy trình nghiệp vụ của người quản trị hệ thống

a. Quy trình nghiệp vụ “Tạo tài khoản cho người dùng”



Hình 2.1.2: Sơ đồ tiến trình nghiệp vụ “Tạo tài khoản”

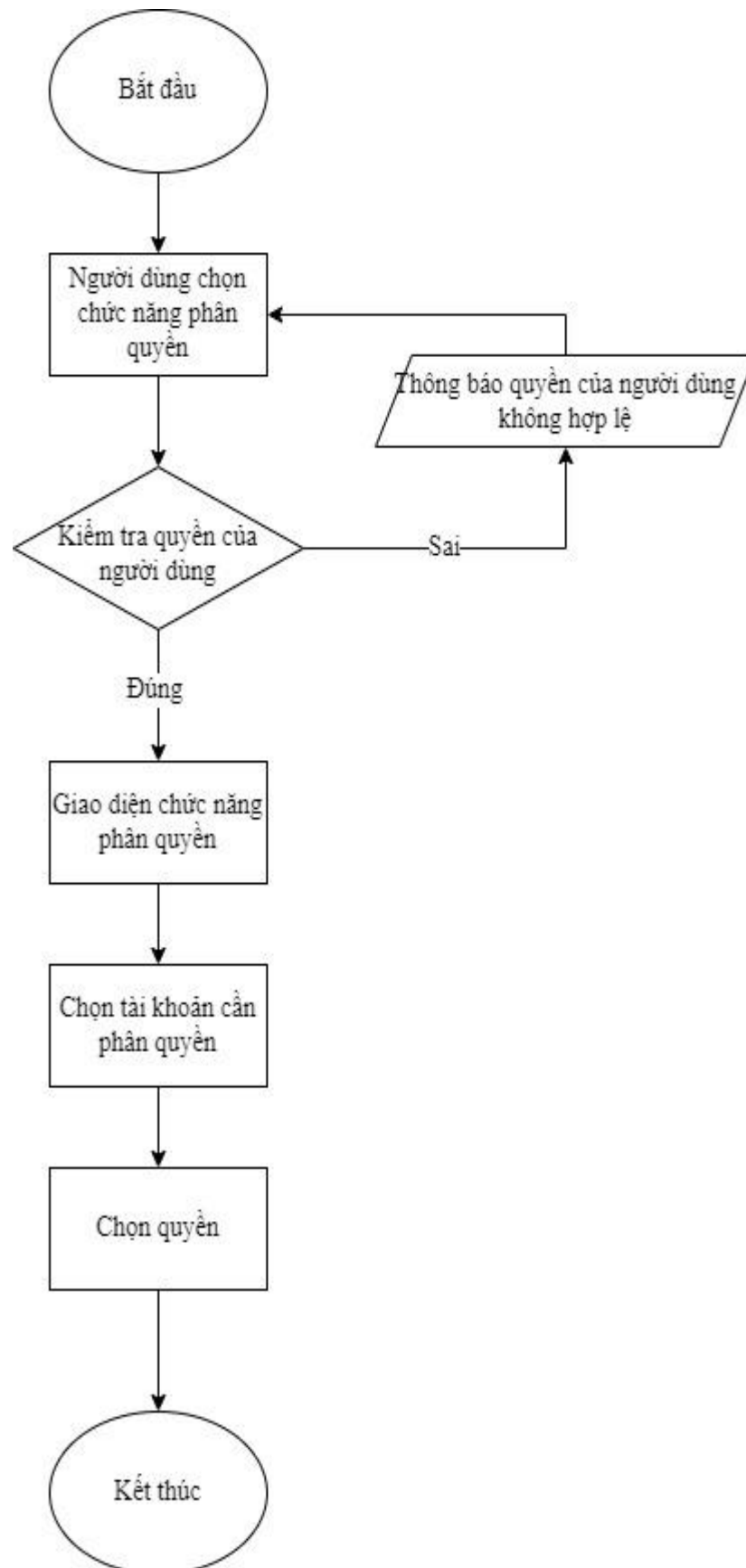
Sau khi người dùng đăng nhập, chọn chức năng tạo tài khoản, hệ thống sẽ kiểm tra quyền có tài khoản người dùng có đúng là quản trị hệ thống (admin) hay chưa.

Nếu sai sẽ thông báo ra màn hình “Tài khoản không có quyền sử dụng chức năng này”.

Nếu đúng sẽ cho phép sử dụng chức năng Tạo tài khoản.

Người dùng nhập các thông tin cần thiết để tạo tài khoản sau đó bấm xác nhận. Hệ thống sẽ kiểm tra các thông tin nhập vào, nếu hợp lệ sẽ lưu thông tin và thông báo thành công. Nếu không hợp lệ sẽ thông báo không thành công và yêu cầu nhập lại.

b. Quy trình nghiệp vụ “Phân quyền tài khoản cho người dùng”



Hình 2.1.3: Sơ đồ tiến trình nghiệp vụ “Phân quyền tài khoản”

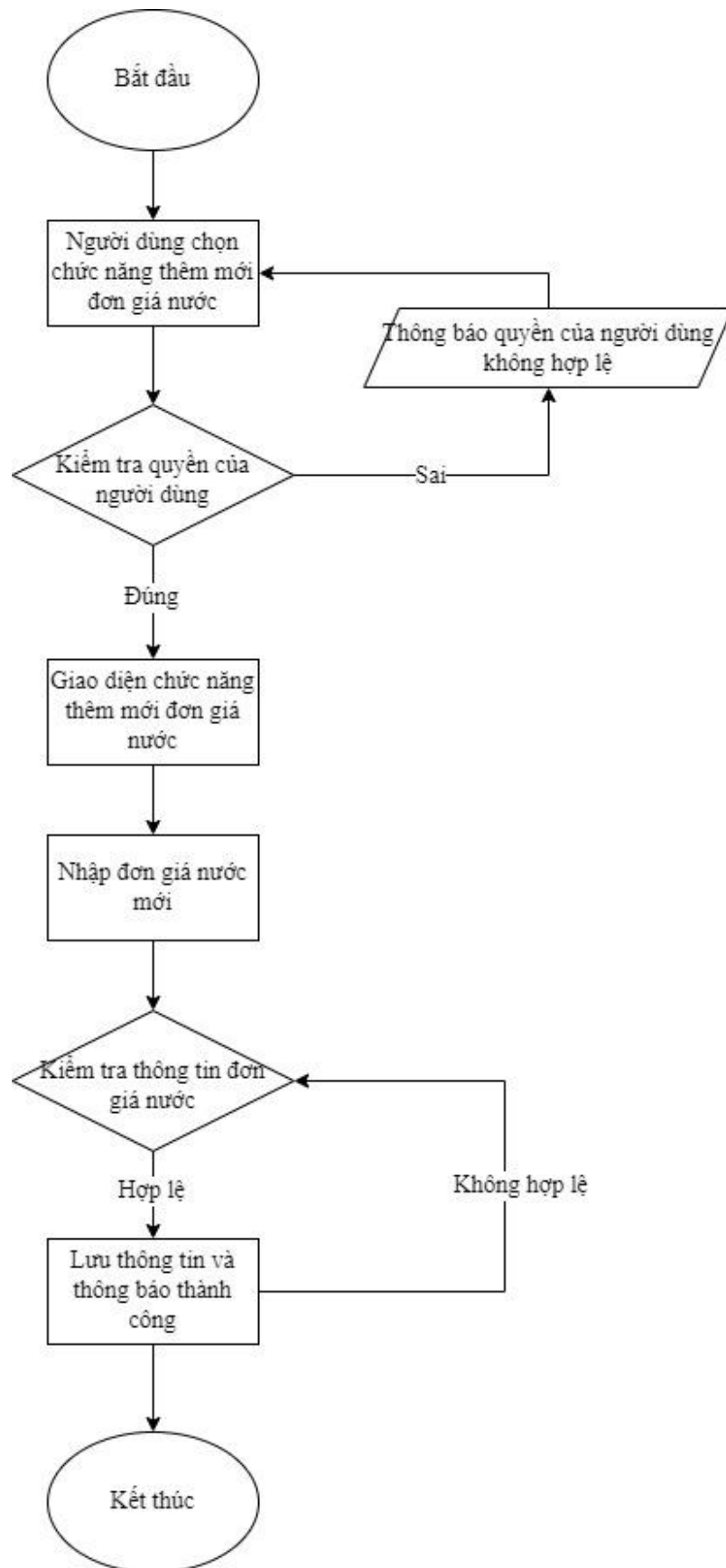
Sau khi người dùng đăng nhập, chọn chức năng phân quyền tài khoản, hệ thống sẽ kiểm tra quyền có tài khoản người dùng có đúng là quản trị hệ thống (admin) hay chưa.

Nếu sai sẽ thông báo ra màn hình “Tài khoản không có quyền sử dụng chức năng này”.

Nếu đúng sẽ cho phép sử dụng chức năng Tạo tài khoản.

Người dùng chọn tài khoản cần phân quyền. Sau đó chọn quyền cần phân cho tài khoản. Bấm xác nhận để lưu lại. Sau đó sẽ thông báo đã lưu thành công hay thất bại.

c. Quy trình nghiệp vụ “Thêm đơn giá nước”



Hình 2.1.4: Sơ đồ tiến trình nghiệp vụ “Thêm đơn giá nước”

Người dùng đăng nhập vào hệ thống bằng tài khoản và mật khẩu của họ.

Sau khi đăng nhập thành công, người dùng chọn chức năng “Thêm mới đơn giá nước”.

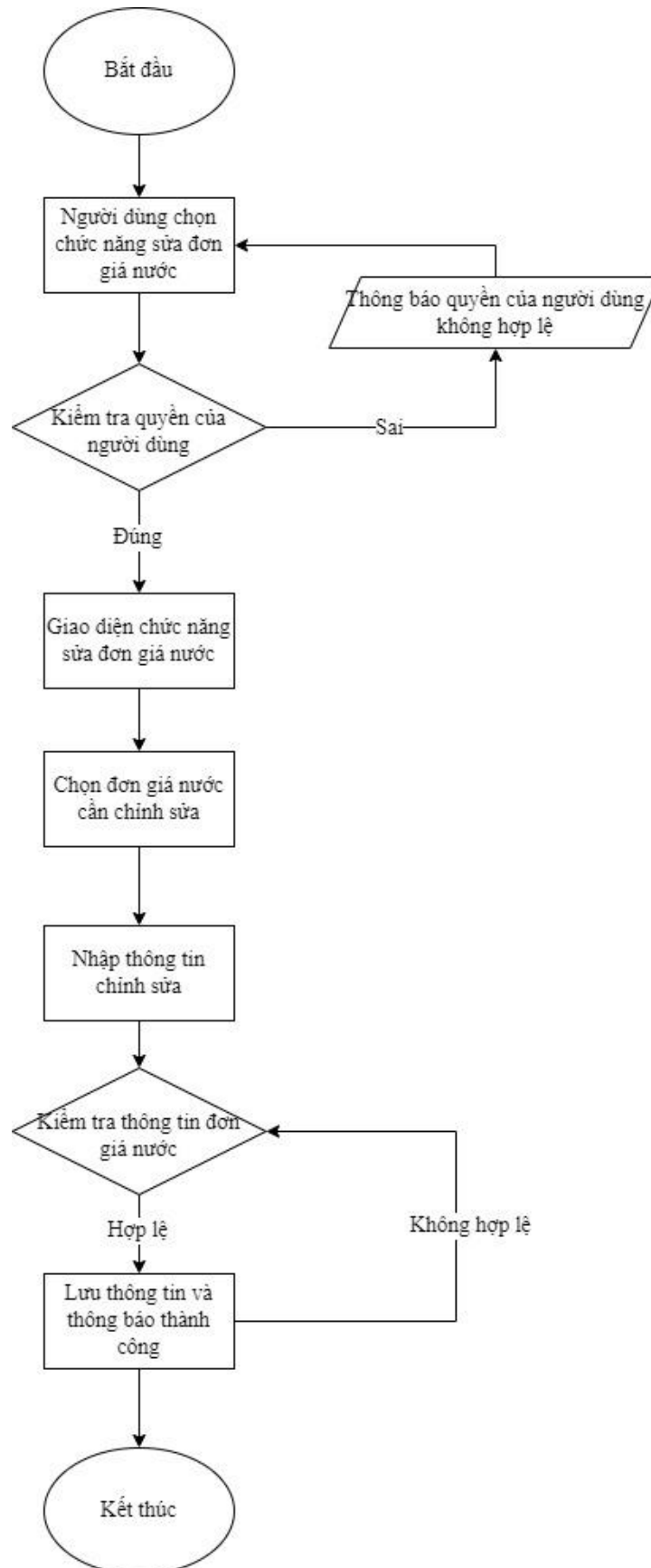
Hệ thống kiểm tra quyền tài khoản của người dùng để có thể cho phép sử dụng chức năng hay không.

Người dùng sẽ được chuyển đến màn hình mẫu nhập thông tin đơn giá nước. Sau khi điền đầy đủ thông tin, bấm lưu để lưu thông tin đơn giá nước mới.

Hệ thống sẽ kiểm tra thông tin được nhập và thêm mới vào cơ sở dữ liệu nếu thông tin hợp lệ.

Sau khi thêm mới đơn giá nước thành công, hệ thống sẽ hiển thị thông báo xác nhận cho người dùng.

d. Quy trình nghiệp vụ “Sửa đơn giá nước”



Hình 2.1.5: Sơ đồ tiến trình nghiệp vụ “Sửa đơn giá nước”

Người dùng đăng nhập vào hệ thống bằng tài khoản và mật khẩu của họ.

Sau khi đăng nhập thành công, người dùng chọn chức năng “Sửa đơn giá nước”.

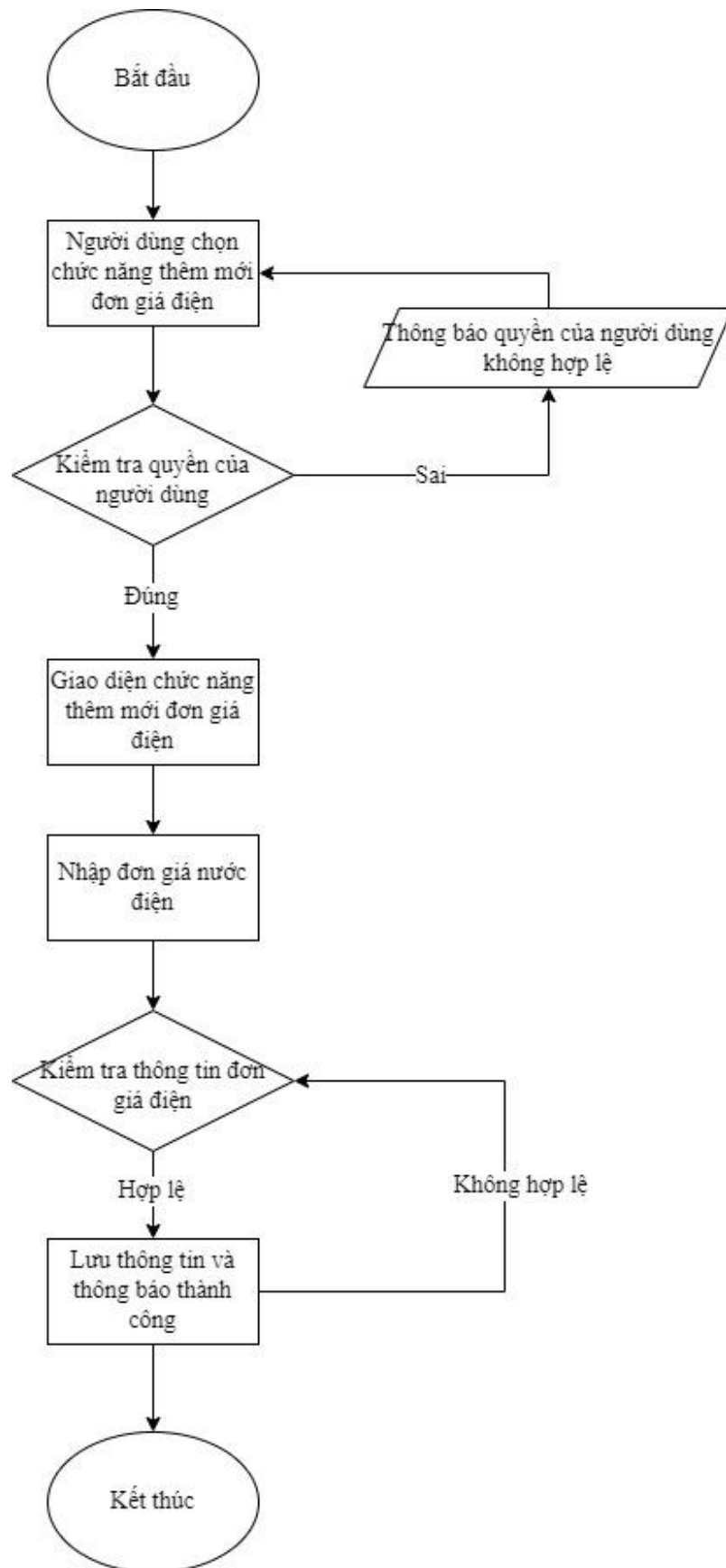
Hệ thống kiểm tra quyền tài khoản của người dùng để có thể cho phép sử dụng chức năng hay không.

Người dùng chọn đơn giá nước cần chỉnh sửa. Sau khi chỉnh sửa thông tin, bấm lưu để lưu thông tin đơn giá nước đã được sửa.

Hệ thống sẽ kiểm tra thông tin được nhập và cập nhật vào cơ sở dữ liệu nếu thông tin hợp lệ.

Sau khi thêm sửa đơn giá nước thành công, hệ thống sẽ hiển thị thông báo xác nhận cho người dùng.

e. Quy trình nghiệp vụ “Thêm đơn giá điện”



Hình 2.1.6: Sơ đồ tiến trình nghiệp vụ “Thêm đơn giá điện”

Người dùng đăng nhập vào hệ thống bằng tài khoản và mật khẩu của họ.

Sau khi đăng nhập thành công, người dùng chọn chức năng “Thêm mới đơn giá điện”.

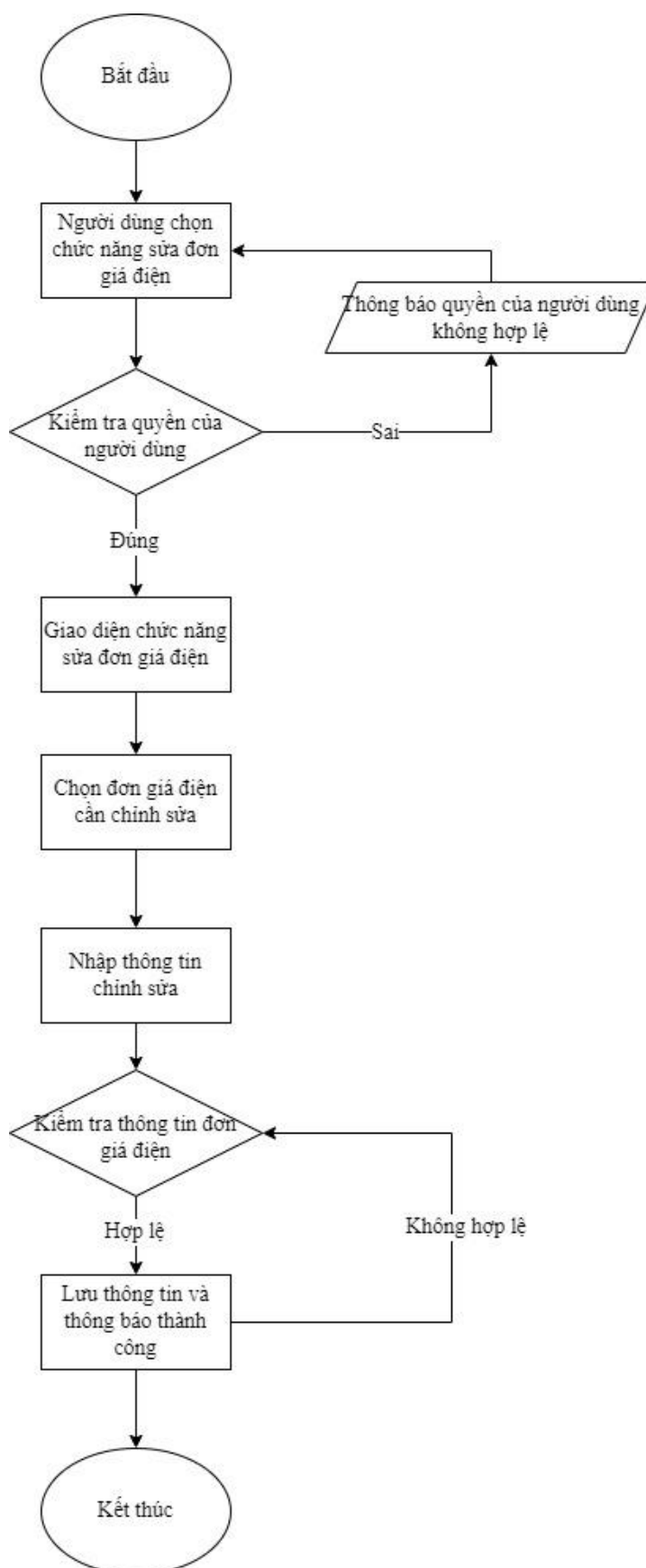
Hệ thống kiểm tra quyền tài khoản của người dùng để có thể cho phép sử dụng chức năng hay không.

Người dùng sẽ được chuyển đến màn hình mẫu nhập thông tin đơn giá điện. Sau khi điền đầy đủ thông tin, bấm lưu để lưu thông tin đơn giá điện mới.

Hệ thống sẽ kiểm tra thông tin được nhập và thêm mới vào cơ sở dữ liệu nếu thông tin hợp lệ.

Sau khi thêm mới đơn giá điện thành công, hệ thống sẽ hiển thị thông báo xác nhận cho người dùng.

f. Quy trình nghiệp vụ “Sửa đơn giá điện”



Hình 2.1.7: Sơ đồ tiến trình nghiệp vụ “Sửa đơn giá điện”

Người dùng đăng nhập vào hệ thống bằng tài khoản và mật khẩu của họ.

Sau khi đăng nhập thành công, người dùng chọn chức năng “Sửa đơn giá điện”.

Hệ thống kiểm tra quyền tài khoản của người dùng để có thể cho phép sử dụng chức năng hay không.

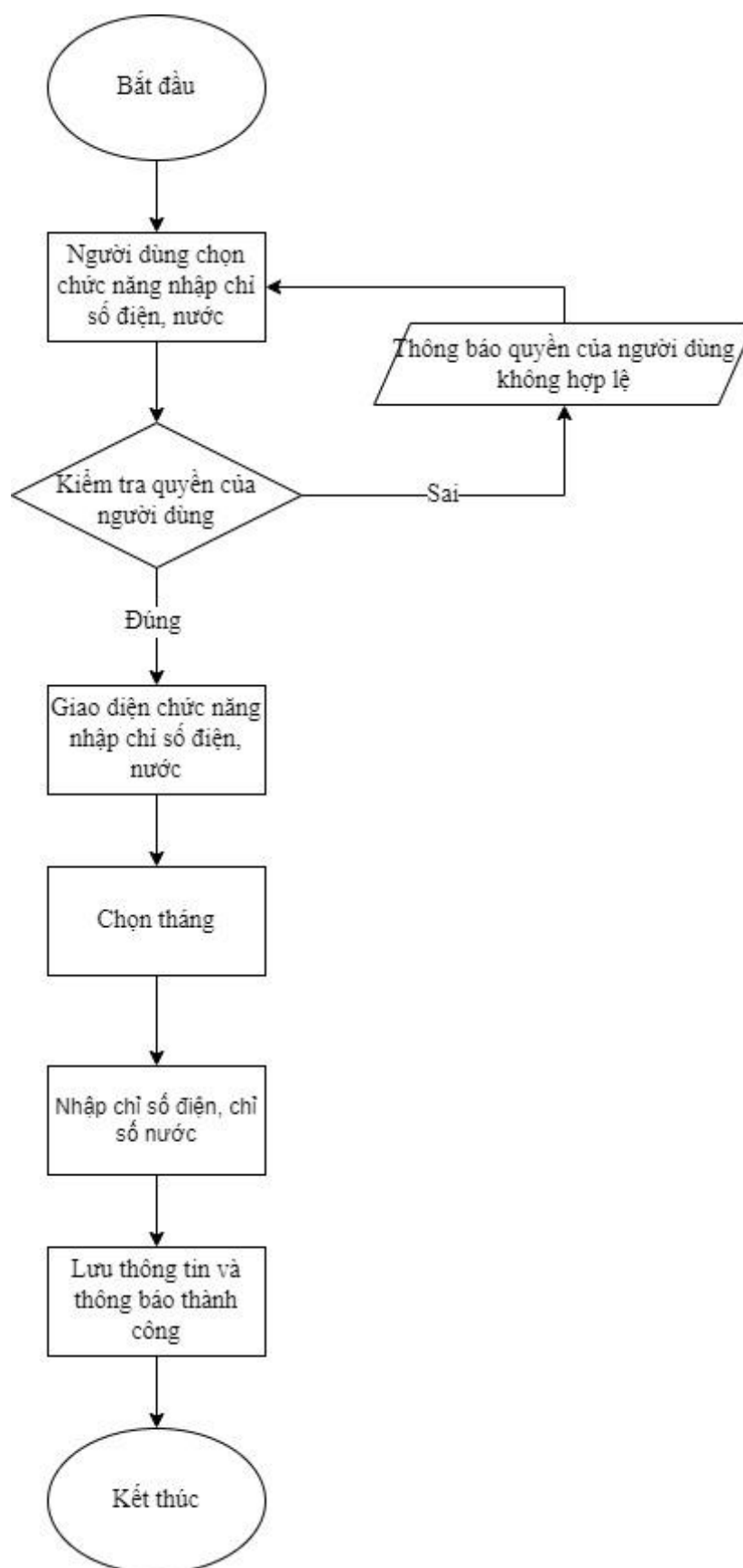
Người dùng chọn đơn giá điện cần chỉnh sửa. Sau khi chỉnh sửa thông tin, bấm lưu để lưu thông tin đơn giá điện đã được sửa.

Hệ thống sẽ kiểm tra thông tin được nhập và cập nhật vào cơ sở dữ liệu nếu thông tin hợp lệ.

Sau khi thêm sửa đơn giá điện thành công, hệ thống sẽ hiển thị thông báo xác nhận cho người dùng.

❖ **Quy trình nghiệp vụ của người quản lý ký túc xá**

a. Quy trình nghiệp vụ “Nhập chỉ số điện, nước hàng tháng”



Hình 2.1.8: Sơ đồ tiến trình nghiệp vụ “Tạo tài khoản”

Người dùng đăng nhập vào hệ thống bằng tài khoản và mật khẩu của họ.

Sau khi đăng nhập thành công, người dùng chọn chức năng “Nhập chỉ số điện, nước”.

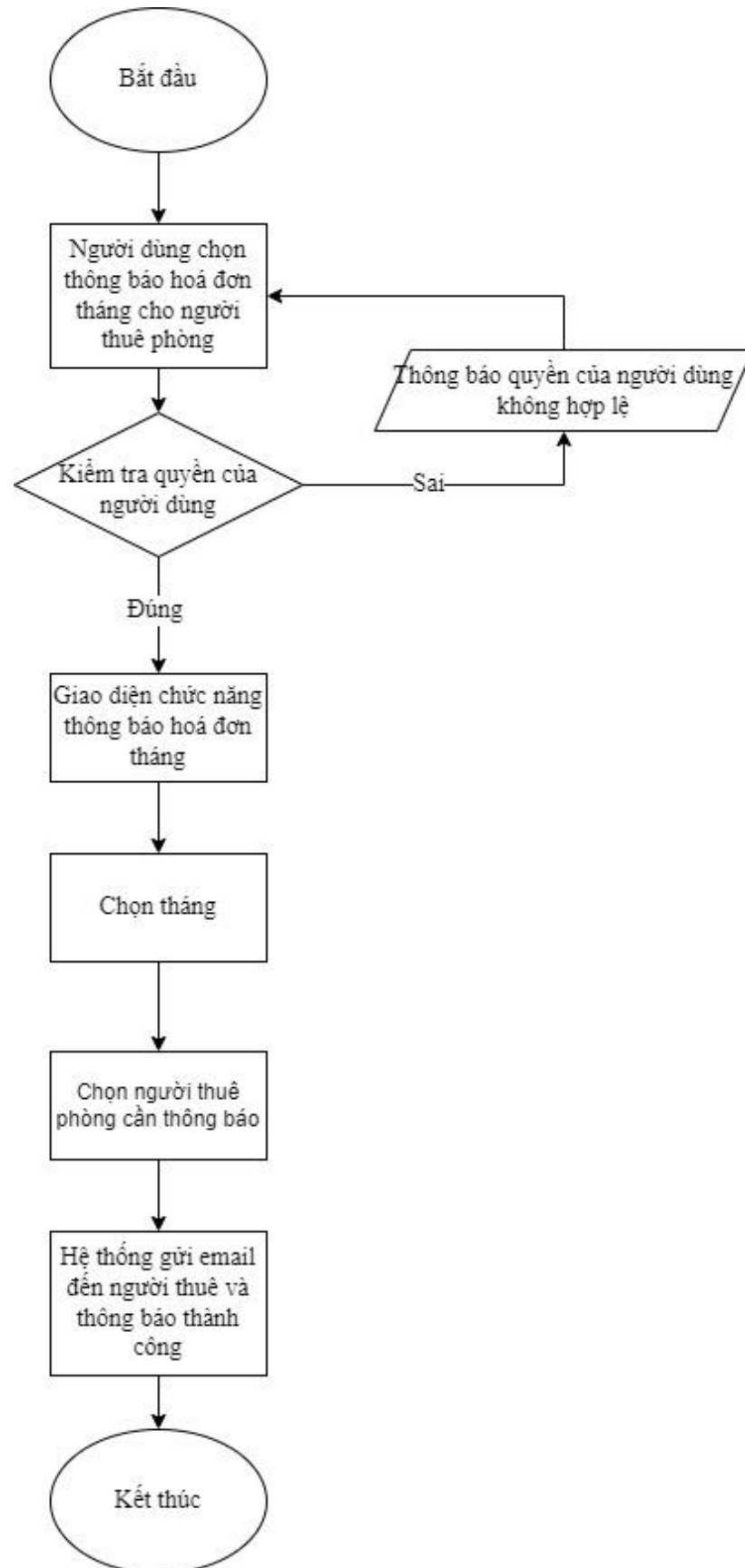
Hệ thống kiểm tra quyền tài khoản của người dùng để có thể cho phép sử dụng chức năng hay không.

Người dùng chọn tháng cần nhập chỉ số điện, nước. Sau khi nhập thông tin bấm lưu.

Hệ thống lưu thông tin được nhập vào cơ sở dữ liệu nếu thông tin hợp lệ.

Sau khi thêm nhập chỉ số điện, nước thành công, hệ thống sẽ hiển thị thông báo xác nhận cho người dùng.

b. Quy trình nghiệp vụ “Thông báo hoá đơn tháng cho người thuê phòng”



Hình 2.1.9: Sơ đồ tiến trình nghiệp vụ “Thông báo hoá đơn tháng cho người thuê phòng”

Người dùng đăng nhập vào hệ thống bằng tài khoản và mật khẩu của họ.

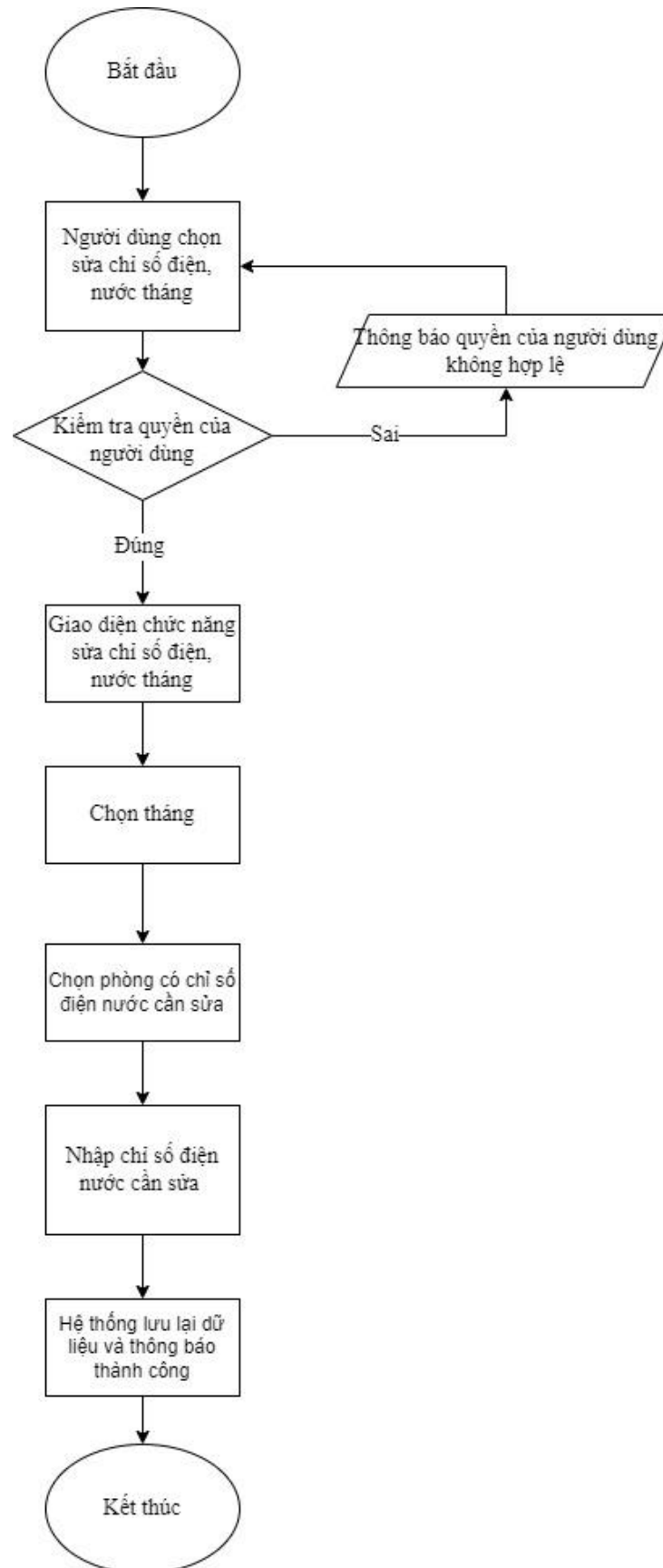
Sau khi đăng nhập thành công, người dùng chọn chức năng “Thông báo hoá đơn tháng cho người thuê phòng”.

Hệ thống kiểm tra quyền tài khoản của người dùng để có thể cho phép sử dụng chức năng hay không.

Người dùng chọn tháng sau đó chọn người thuê cần thông báo sau đó bấm gửi.

Hệ thống gửi thông tin về hoá đơn tháng được chọn cho người thuê được chọn qua email của người thuê phòng được lưu trong cơ sở dữ liệu. Gửi thông báo cho người dùng khi gửi email thành công.

c. Quy trình nghiệp vụ “Sửa chỉ số điện, nước tháng”



Hình 2.1.10: Sơ đồ tiến trình nghiệp vụ “Sửa chỉ số điện, nước tháng”

Người dùng đăng nhập vào hệ thống bằng tài khoản và mật khẩu của họ.

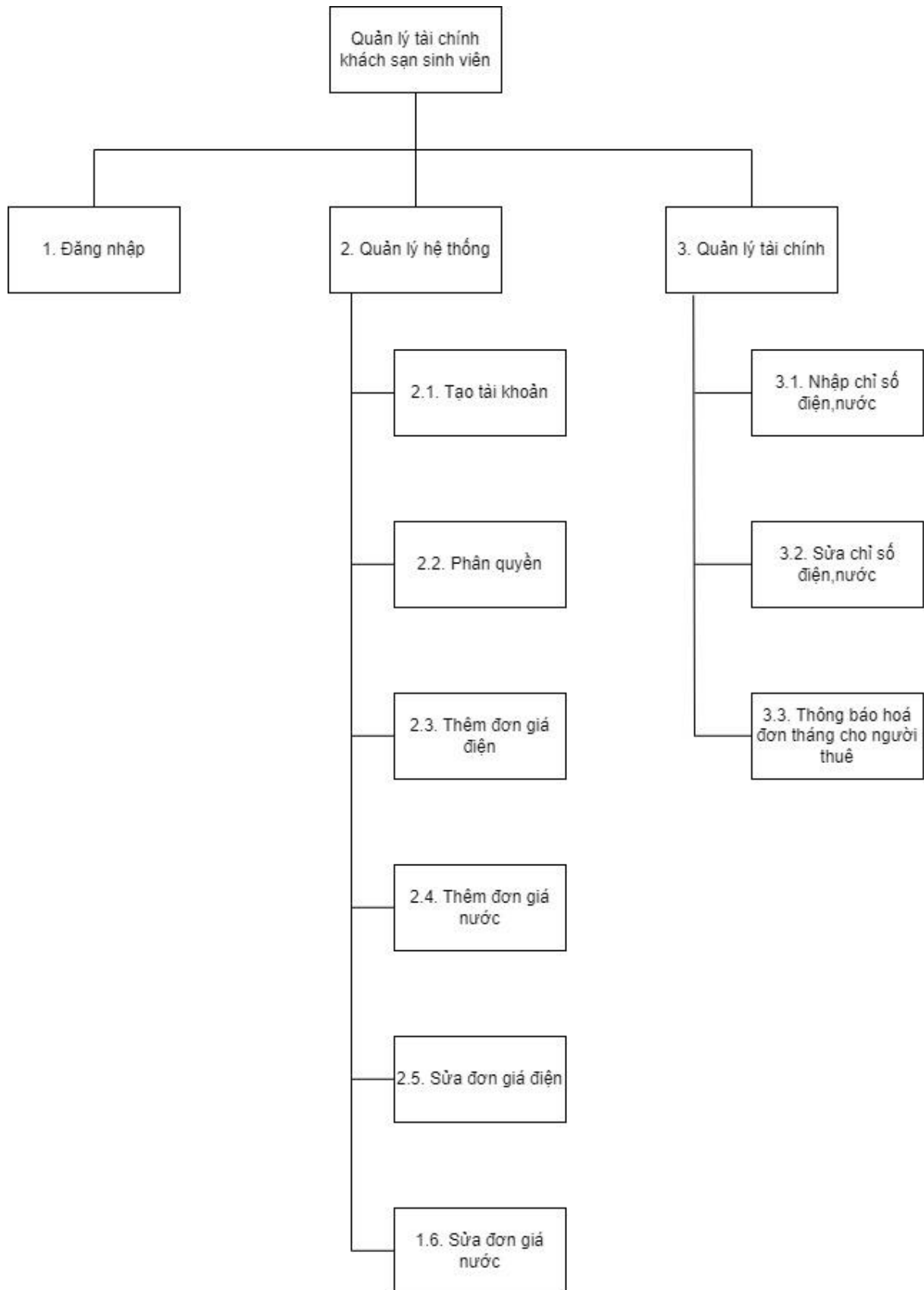
Sau khi đăng nhập thành công, người dùng chọn chức năng “Chỉnh sửa chỉ số điện, nước tháng”.

Hệ thống kiểm tra quyền tài khoản của người dùng để có thể cho phép sử dụng chức năng hay không.

Người dùng chọn tháng sau đó chọn phòng có chỉ số điện, nước cần chỉnh sửa.

Chỉnh sửa chỉ số điện, nước cần thiết, bấm lưu để lưu vào hệ thống.

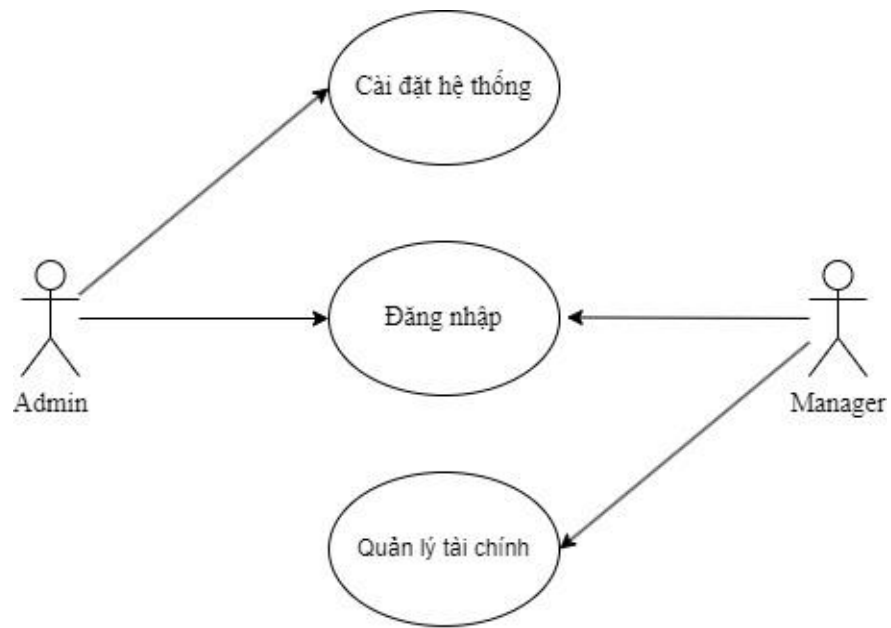
2.2. Sơ đồ phân rã chức năng



Hình 2.2.1: Sơ đồ phân rã chức năng

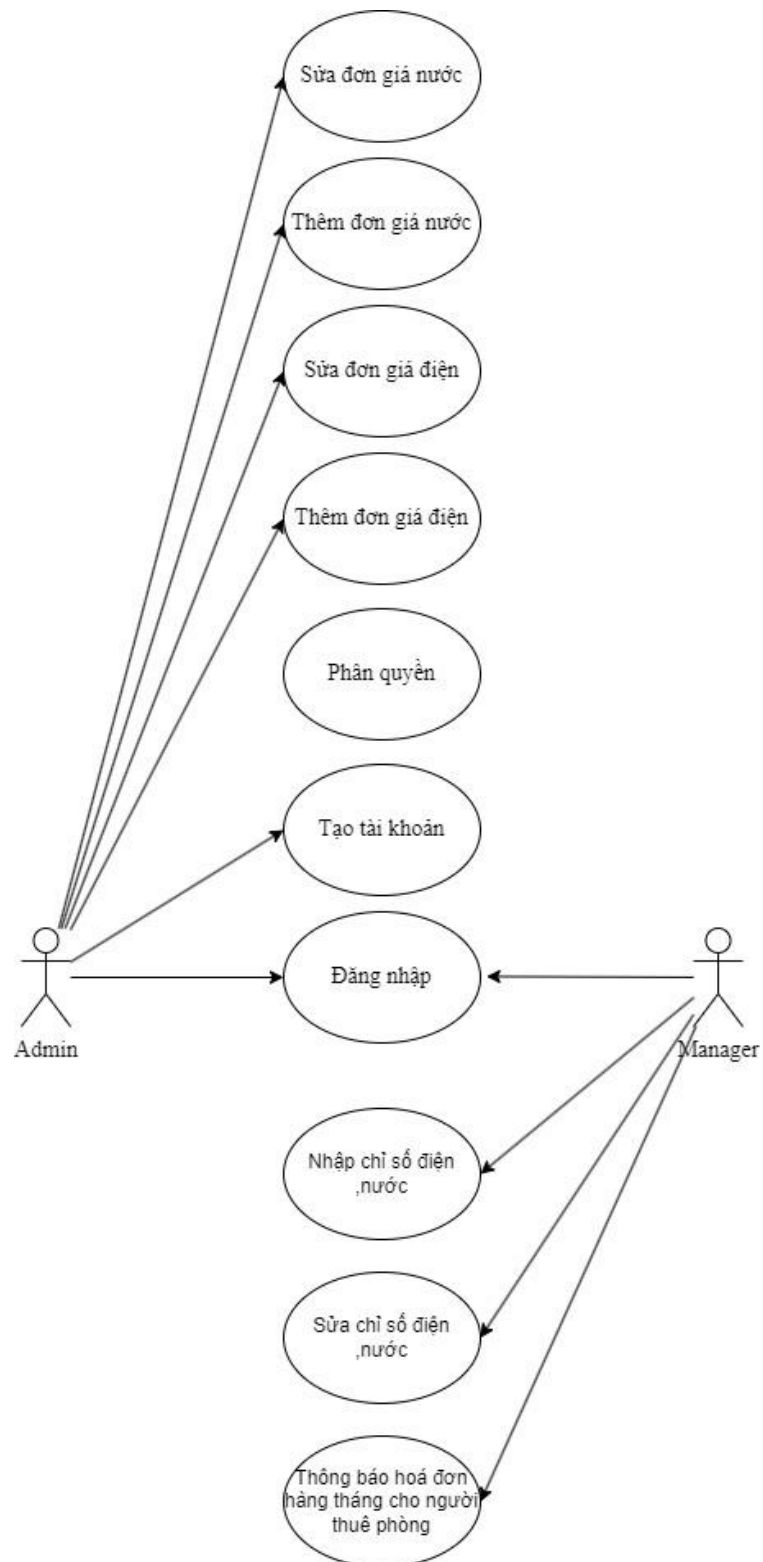
2.3. Biểu đồ Usecase

2.3.1. Biểu đồ Use Case tổng quát



Hình 2.3.1: Usecase tổng quát.

2.3.2. Biểu đồ Use Case chi tiết



Hình 2.3.2: Usecase chi tiết

2.4. Đặc tả User Case

2.4.1. Use Case “Tạo tài khoản”

Tên Use Case	Tạo tài khoản
Tác nhân chính.	Quản trị hệ thống
Điều kiện trước.	Người quản trị hệ thống có nhu cầu tạo tài khoản
Điều kiện sau.	Tạo được tài khoản cho người cần
Các bước tiến hành: 1. Người quản trị hệ thống vào chức năng tạo tài khoản. 2. Người quản trị hệ thống nhập các thông tin của tài khoản. 3. Lưu lại tài khoản đã được tạo	
Ngoại lệ: Tên tài khoản không phù hợp	

2.4.2. Use Case “Phân quyền”

Tên Use Case	Phân quyền
Tác nhân chính.	Quản trị hệ thống
Điều kiện trước.	Người quản trị hệ thống có nhu cầu phân quyền cho tài khoản
Điều kiện sau.	Tài khoản đã được phân quyền
Các bước tiến hành: 1. Người quản trị hệ thống vào chức năng phân quyền. 2. Người quản trị hệ thống chọn tài khoản cần phân quyền. 3. Phân quyền cho tài khoản và lưu lại	
Ngoại lệ:	

2.4.3. Use Case “Thêm mới đơn giá nước”

Tên Use Case	Thêm mới đơn giá nước
Tác nhân chính.	Quản trị hệ thống.
Điều kiện trước.	Quản trị hệ thống muốn thêm đơn giá nước mới.

Điều kiện sau.	Đơn giá nước được thêm thành công.
Các bước tiến hành: 1. Quản trị hệ thống chọn chức năng thêm mới đơn giá nước. 2. Quản trị hệ thống nhập đơn giá nước từ giao diện. 3. Chọn Thêm mới để lưu lại. 4. Giao diện thông báo thêm mới thành công.	
Ngoại lệ: 1. Giao diện thông báo thêm mới đơn giá nước không thành công.	

2.4.4. Use Case “Sửa đơn giá nước”

Tên Use Case	Sửa đơn giá nước
Tác nhân chính.	Quản trị hệ thống.
Điều kiện trước.	Quản trị hệ thống muốn sửa đơn giá nước.
Điều kiện sau.	Đơn giá nước được sửa thành công.
Các bước tiến hành: 1. Quản trị hệ thống chọn chức năng sửa đơn giá nước. 2. Quản trị hệ thống chọn đơn giá nước cần sửa từ giao diện. 3. Nhập thông tin chỉnh sửa vào giao diện. 4. Giao diện thông báo thêm mới thành công.	
Ngoại lệ: 1. Giao diện thông báo sửa đơn giá nước không thành công.	

2.4.5. Use Case “Thêm mới đơn giá điện”

Tên Use Case	Thêm mới đơn giá điện
Tác nhân chính.	Quản trị hệ thống.
Điều kiện trước.	Quản trị hệ thống muốn thêm đơn giá điện mới.
Điều kiện sau.	Đơn giá điện được thêm thành công.
Các bước tiến hành:	

1. Quản trị hệ thống chọn chức năng thêm mới đơn giá điện. 2. Quản trị hệ thống nhập đơn giá điện từ giao diện. 3. Chọn thêm mới để lưu lại. 4. Giao diện thông báo thêm mới thành công.
Ngoại lệ: 1. Giao diện thông báo thêm mới đơn giá điện không thành công.

2.4.6. Use Case “Sửa đơn giá điện”

Tên Use Case	Sửa đơn giá điện
Tác nhân chính.	Quản trị hệ thống.
Điều kiện trước.	Quản trị hệ thống muốn sửa đơn giá điện.
Điều kiện sau.	Đơn giá điện được sửa thành công.
Các bước tiến hành: 1. Quản trị hệ thống chọn chức năng sửa đơn giá điện. 2. Quản trị hệ thống chọn đơn giá điện cần sửa từ giao diện. 3. Nhập thông tin chỉnh sửa vào giao diện. 4. Giao diện thông báo thêm mới thành công.	
Ngoại lệ: 1. Giao diện thông báo sửa đơn giá điện không thành công.	

2.4.7. Use Case “Đăng nhập”

Tên Use Case	Đăng nhập
Tác nhân chính.	Người dùng : Quản trị hệ thống và quản lý ký túc xá.
Điều kiện trước.	Người dùng có nhu cầu sử dụng hệ thống.
Điều kiện sau.	Người dùng đăng nhập thành công.
Các bước tiến hành: 1. Người dùng vào trang đăng nhập. 2. Người dùng nhập thông tin tài khoản và mật khẩu.	

3. Giao diện thông báo đăng nhập thành công.
4. Chuyển người dùng đến giao diện hệ thống.
Ngoại lệ:
1. Người dùng nhập sai tài khoản hoặc mật khẩu.

2.4.8. Use Case “Nhập chỉ số điện, nước”

Tên Use Case	Nhập chỉ số điện, nước
Tác nhân chính.	Người quản lý ký túc xá
Điều kiện trước.	Người quản lý ký túc xá muốn nhập chỉ số điện, nước cuối tháng.
Điều kiện sau.	Chỉ số điện, nước được thêm thành công
Các bước tiến hành: 1. Người quản lý ký túc xá chọn chức năng “Nhập chỉ số điện, nước”. 2. Người quản lý ký túc xá chọn tháng cần nhập. 3. Người quản lý ký túc xá chọn phòng cần nhập. 4. Nhập chỉ số điện, nước vào giao diện.	
Ngoại lệ: 1. Hệ thống thông báo nhập chỉ số điện, nước không thành công.	

2.4.9. Use Case “Sửa chỉ số điện, nước”

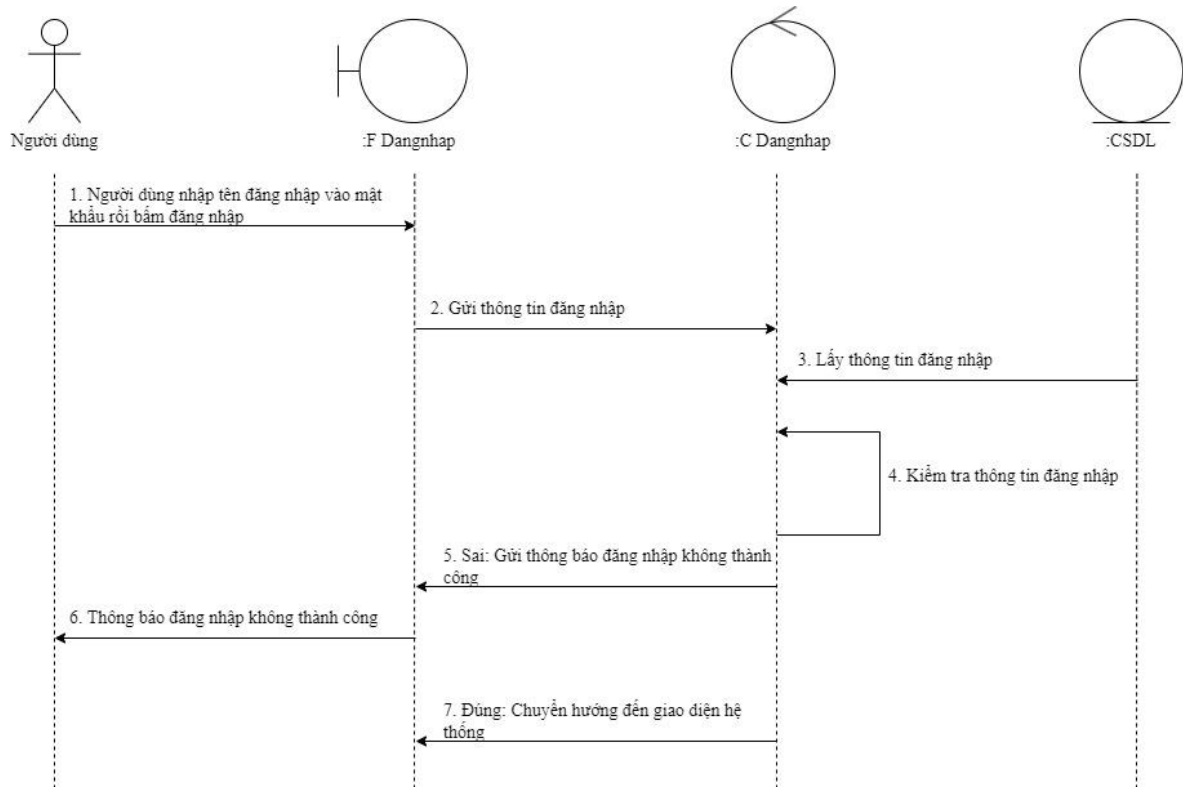
Tên Use Case	Sửa chỉ số điện, nước
Tác nhân chính.	Người quản lý ký túc xá
Điều kiện trước.	Người quản lý ký túc xá có nhu cầu sửa chỉ số điện, nước.
Điều kiện sau.	Người quản lý ký túc xá sửa chỉ số điện nước thành công.
Các bước tiến hành: 1. Người quản lý ký túc xá chọn chức năng “Sửa chỉ số điện, nước”. 2. Người quản lý ký túc xá chọn tháng cần sửa. 3. Người quản lý ký túc xá chọn phòng cần sửa.	

4. Nhập chỉ số điện, nước mới vào giao diện.
Ngoại lệ: 1. Hệ thống thông báo nhập chỉ số điện, nước không thành công.

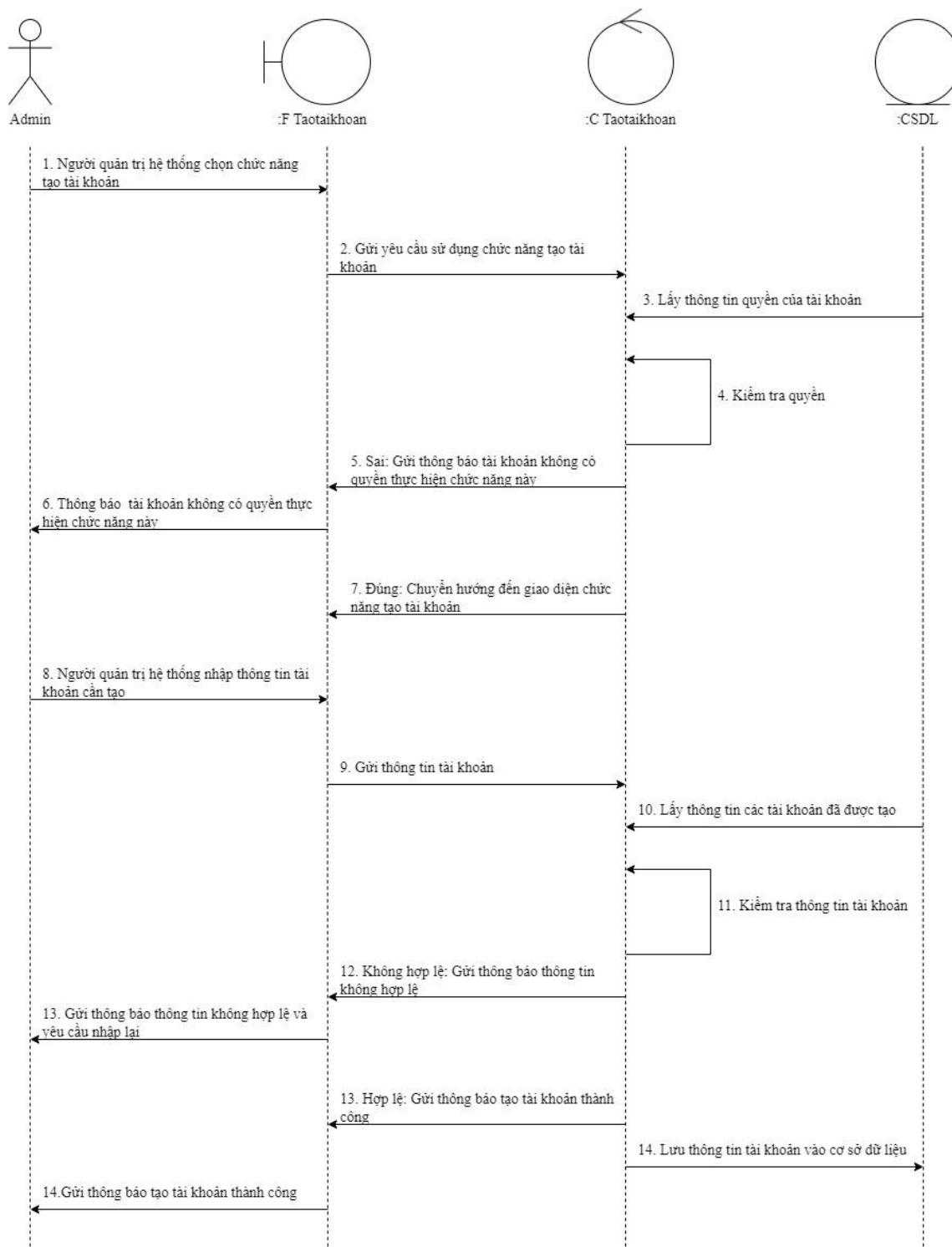
2.4.10. Use Case “Thông báo hoá đơn hàng tháng cho người thuê phòng”

Tên Use Case	Thông báo hoá đơn hàng tháng cho người thuê phòng
Tác nhân chính.	Người quản lý ký túc xá
Điều kiện trước.	Người quản lý ký túc xá có nhu cầu thông báo hoá đơn hàng tháng cho người thuê phòng
Điều kiện sau.	Người quản lý ký túc xá thông báo hoá đơn hàng tháng cho người thuê phòng thành công.
Các bước tiến hành: 1. Người quản lý ký túc xá chọn chức năng “Thông báo hoá đơn hàng tháng cho người thuê phòng”. 2. Người quản lý ký túc xá chọn người thuê phòng cần thông báo. 3. Người quản lý ký túc xá chọn xác nhận thông báo. 4. Hệ thống thông báo đã gửi email thành công.	

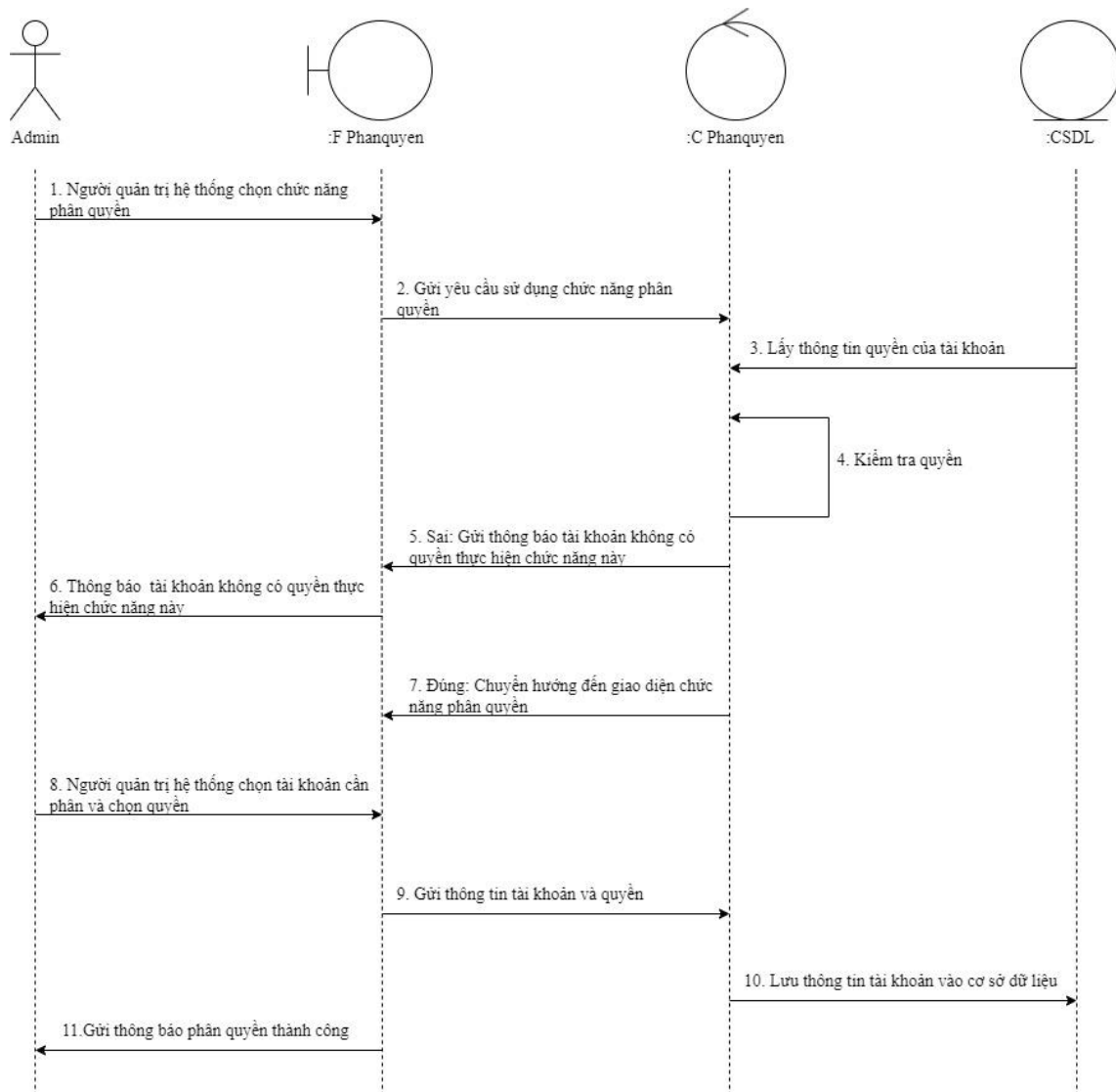
2.5. Biểu đồ tuần tự



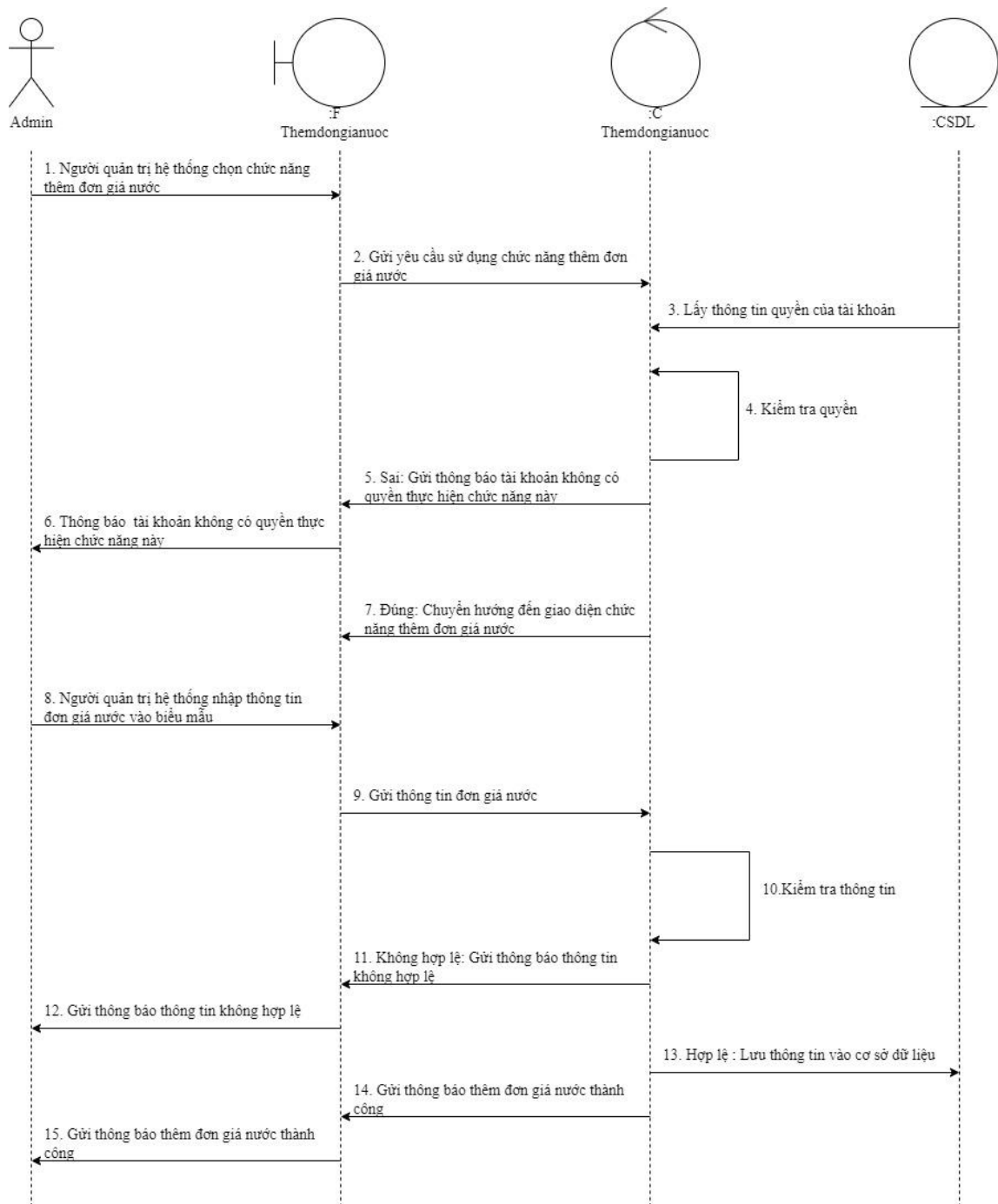
Hình 2.5.1: Biểu đồ tuần tự thực thi Use Case "Đăng nhập"



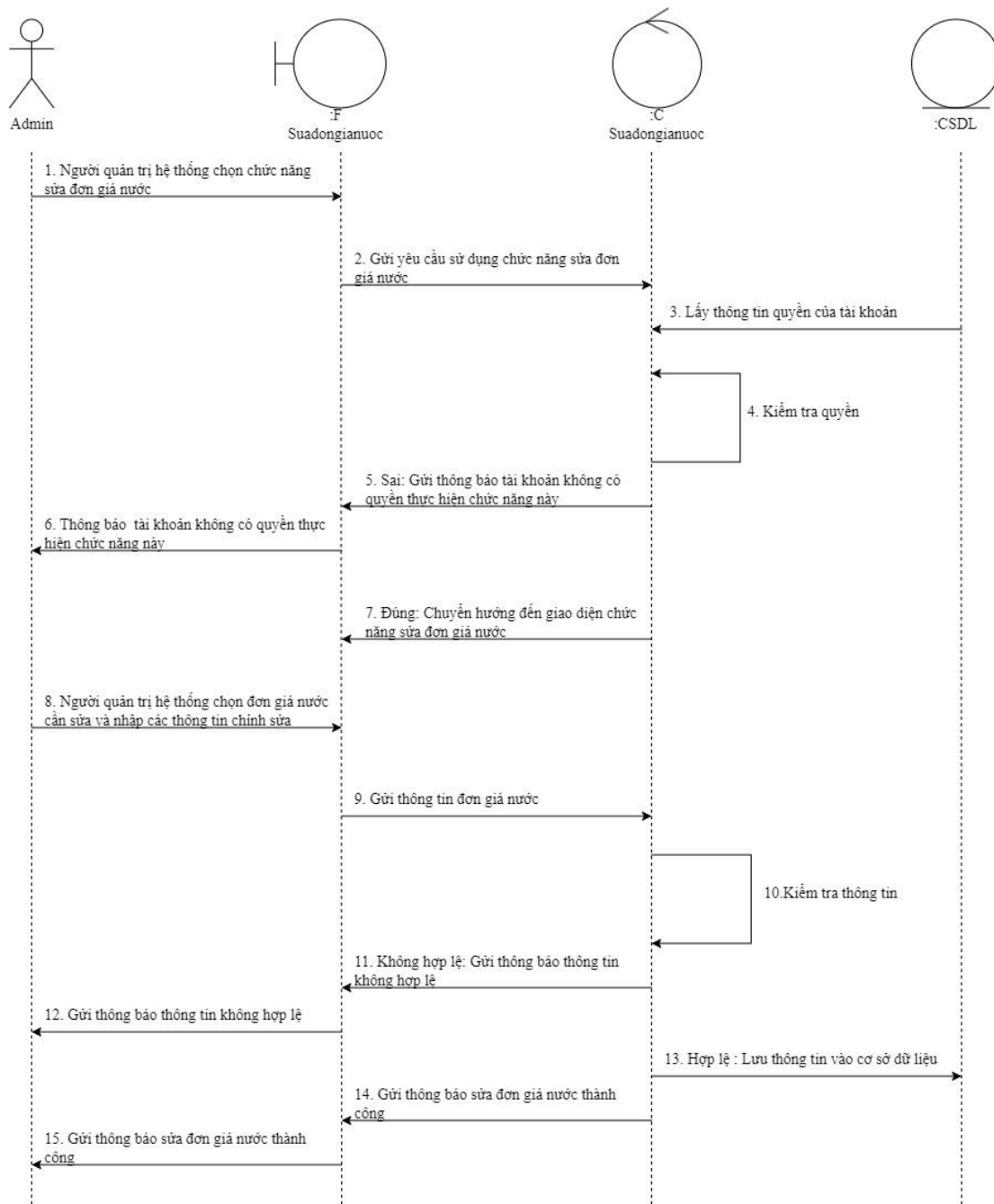
Hình 2.5.2: Biểu đồ tuần tự thực thi Use Case "Tạo tài khoản"



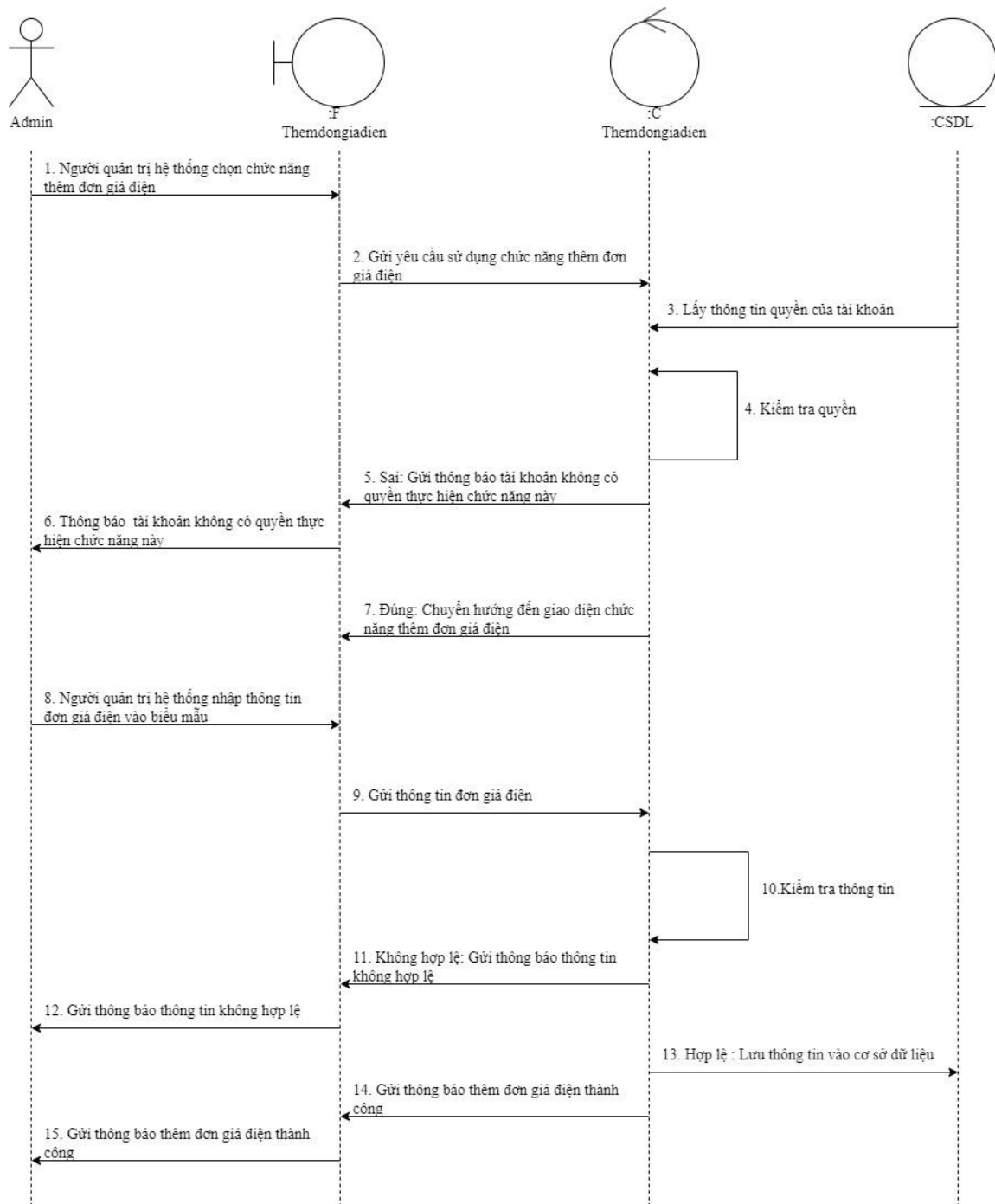
Hình 2.5.3: Biểu đồ tuần tự thực thi Use Case "Phân quyền"



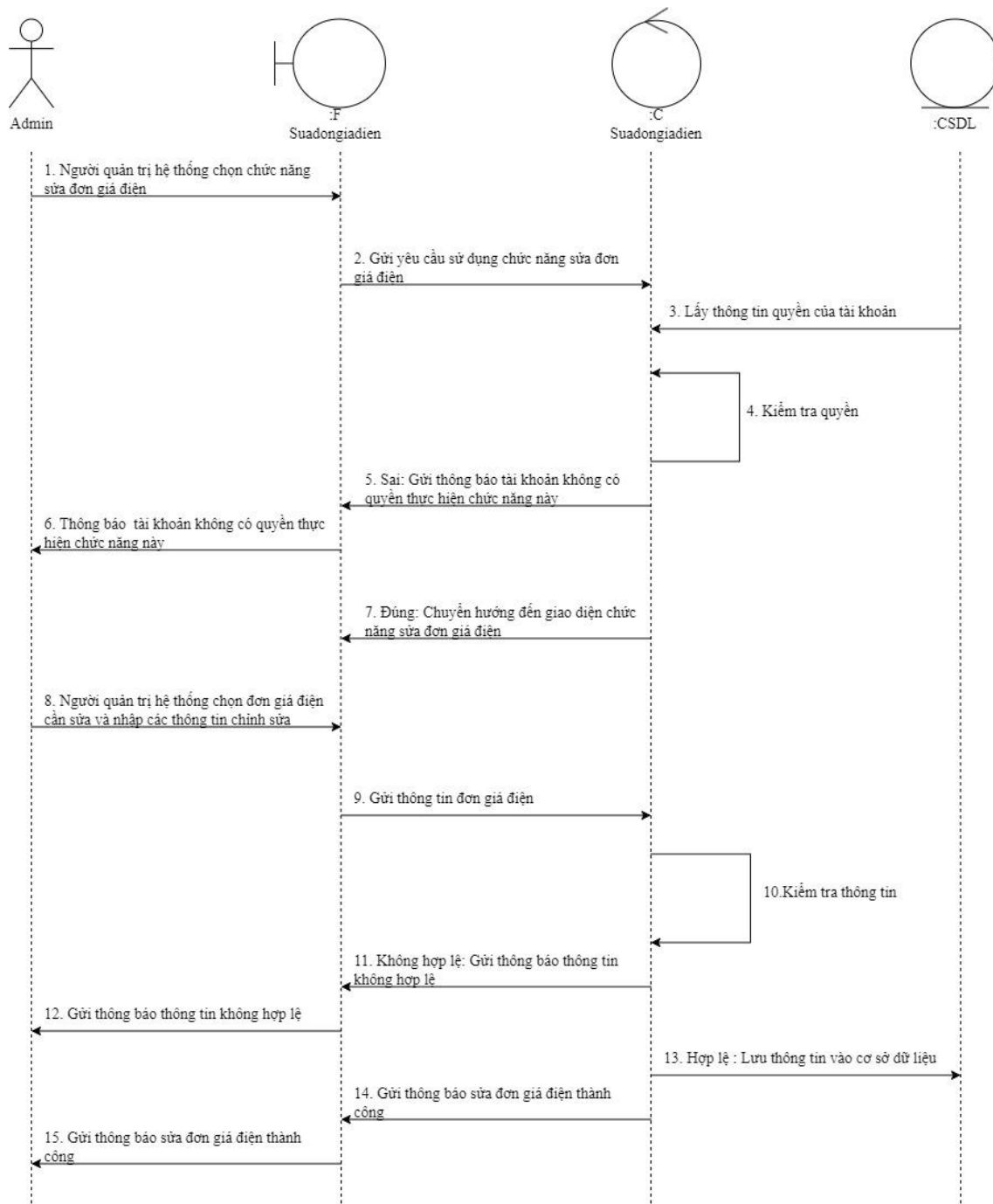
Hình 2.5.4: Biểu đồ tuần tự thực thi Use Case "Thêm mới đơn giá nước"



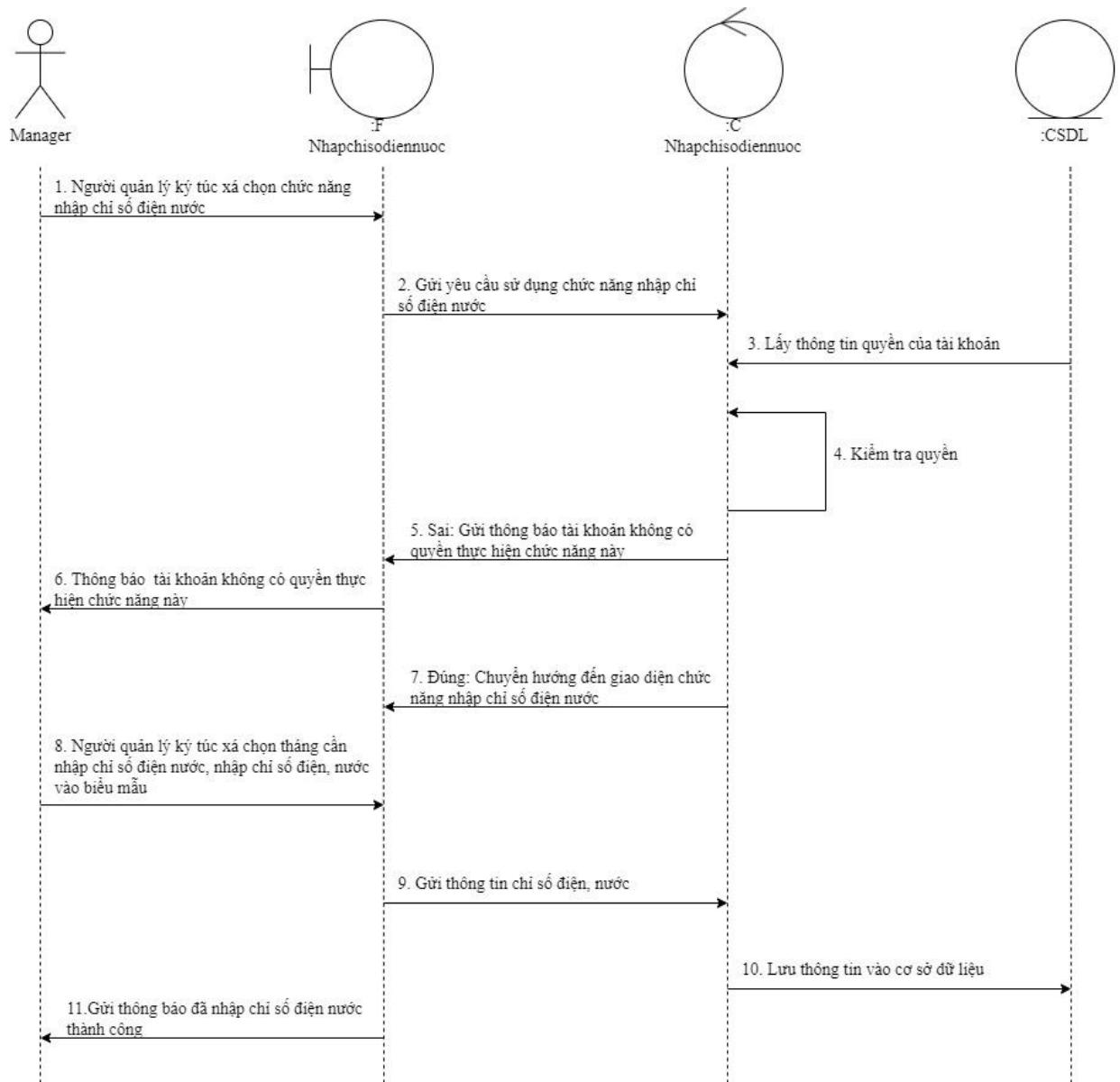
Hình 2.5.5: Biểu đồ tuần tự thực thi Use Case "Sửa đơn giá nước"



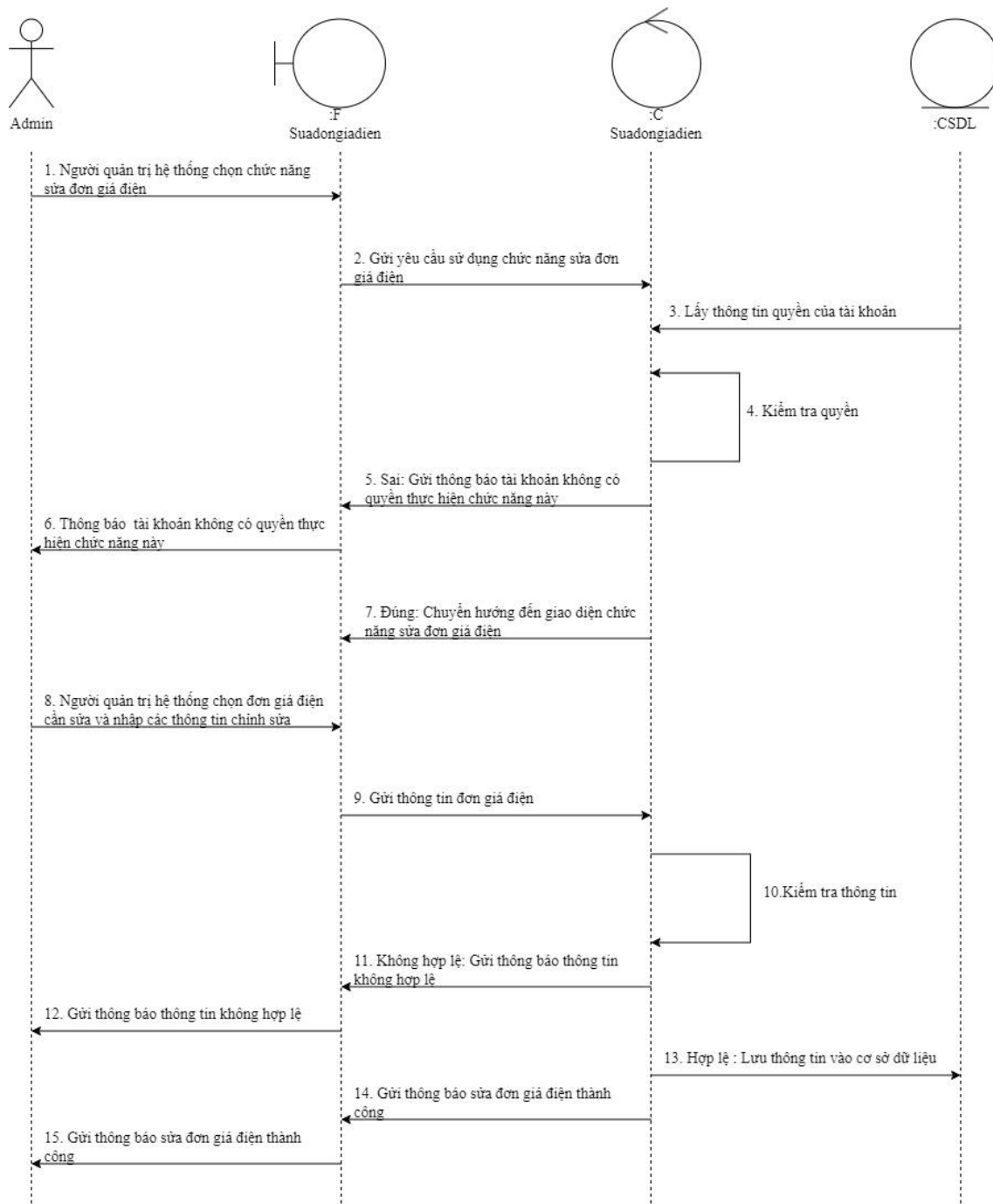
Hình 2.5.6: Biểu đồ tuần tự thực thi Use Case "Thêm mới đơn giá điện"



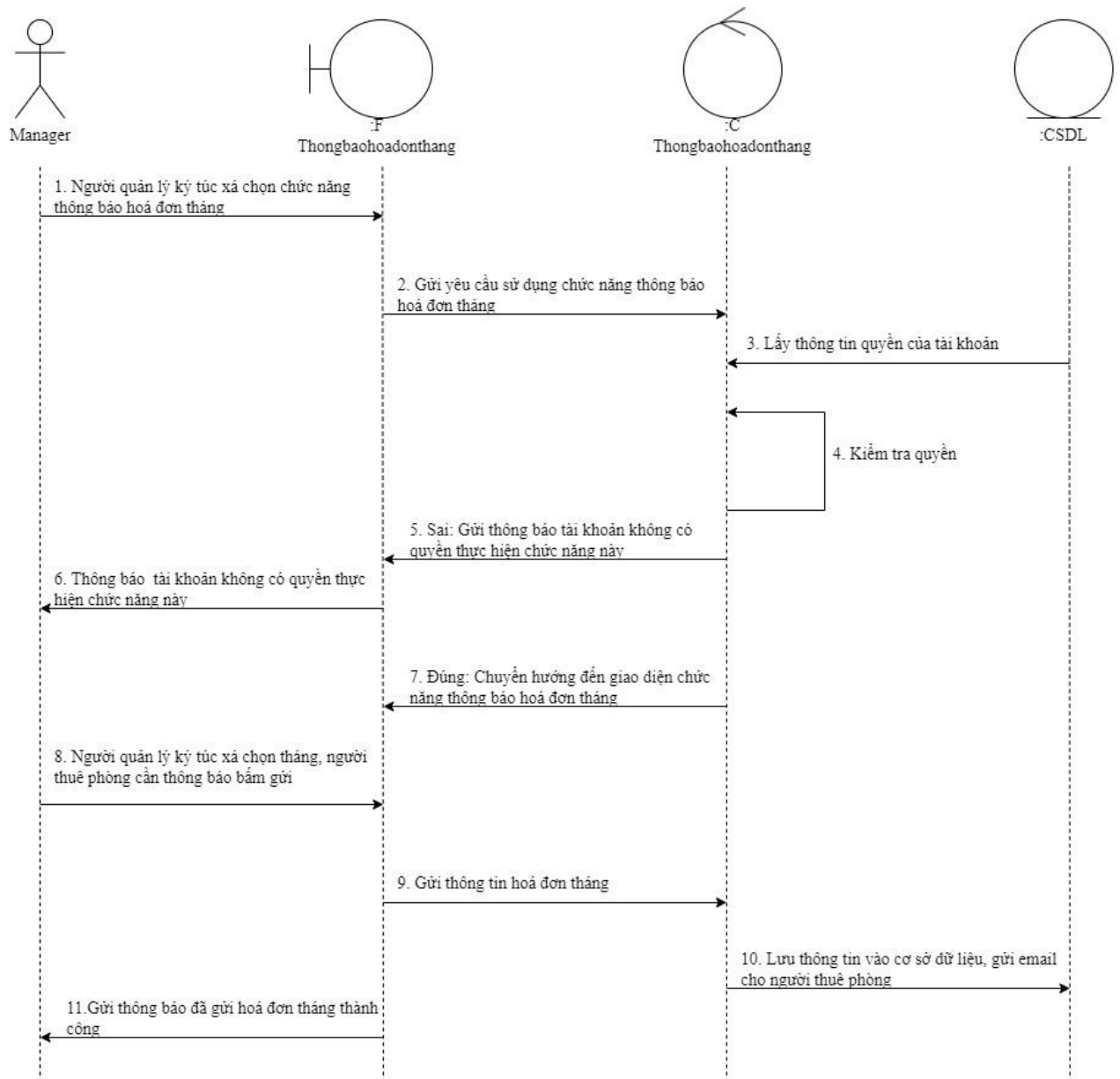
Hình 2.5.7: Biểu đồ tuần tự thực thi Use Case "Sửa đơn giá điện"



Hình 2.5.8: Biểu đồ tuần tự thực thi Use Case "Nhập chỉ số điện nước"

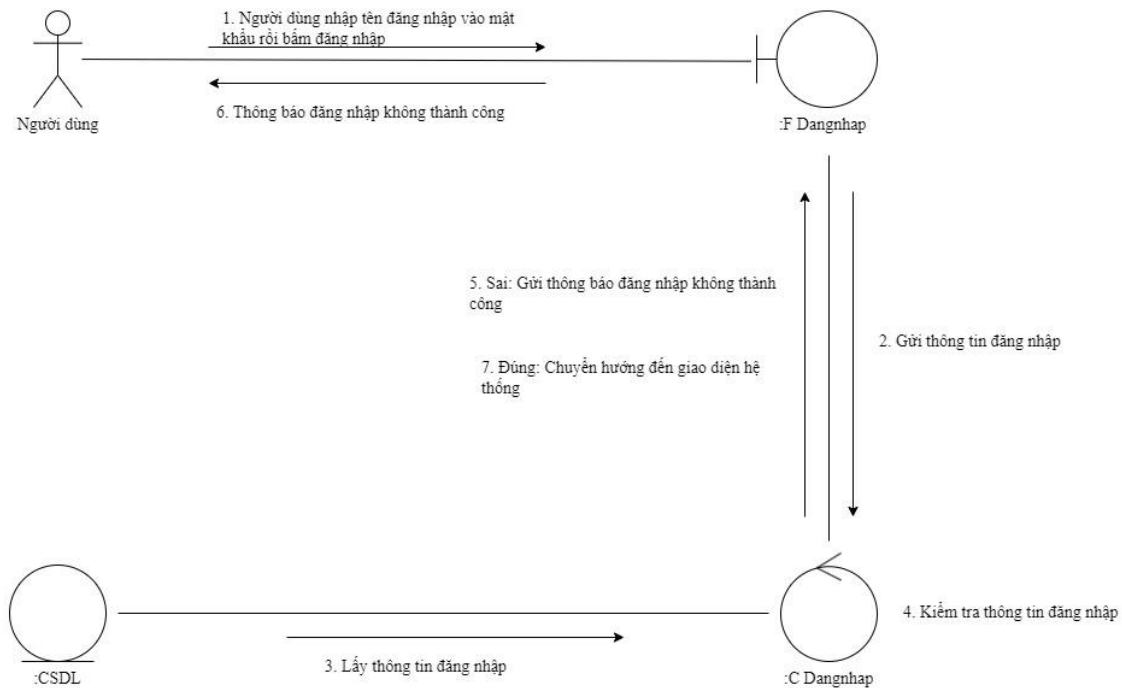


Hình 2.5.9: Biểu đồ tuần tự thực thi Use Case "Sửa chỉ số điện nước"

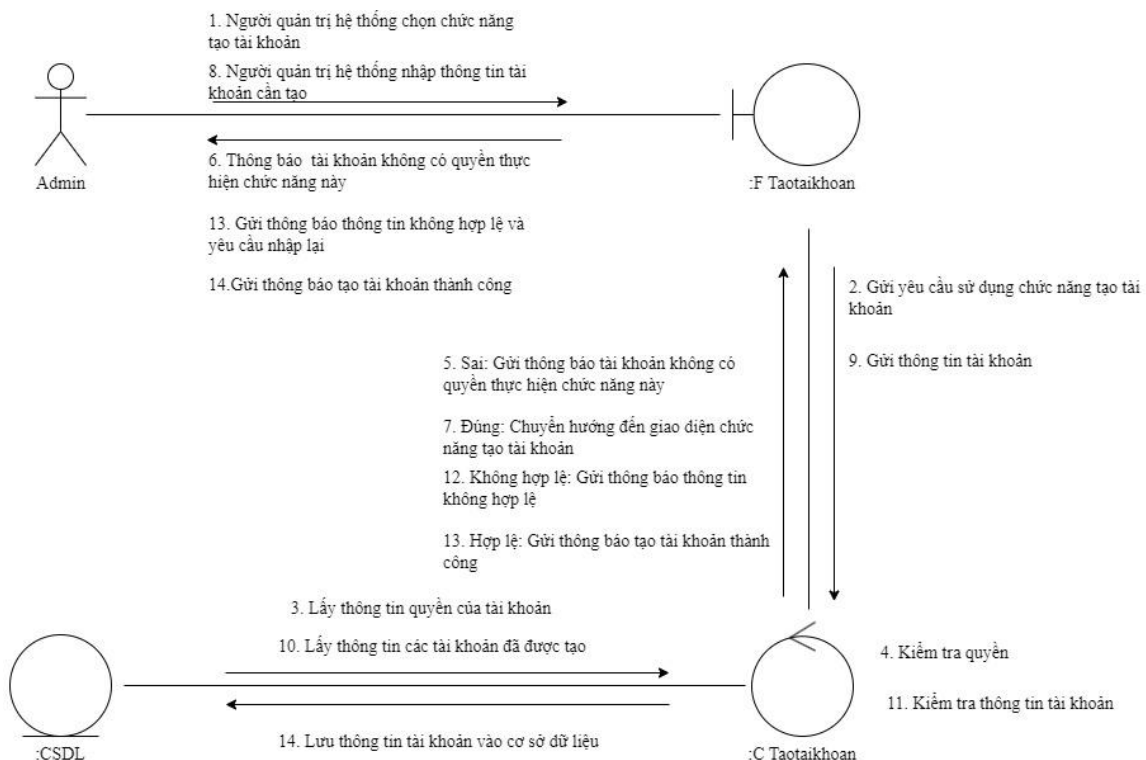


Hình 2.5.10: Biểu đồ tuần tự thực thi Use Case "Thông báo hoá đơn tháng"

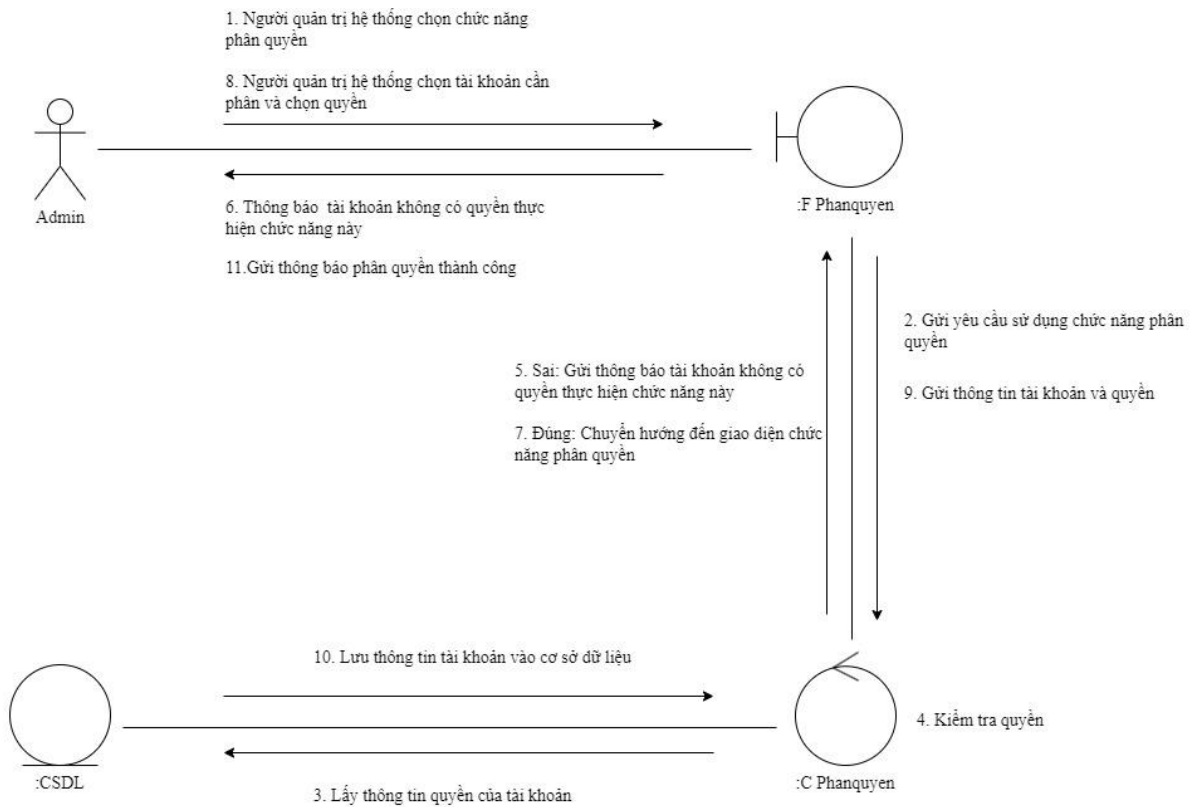
2.6. Biểu đồ cộng tác thực thi



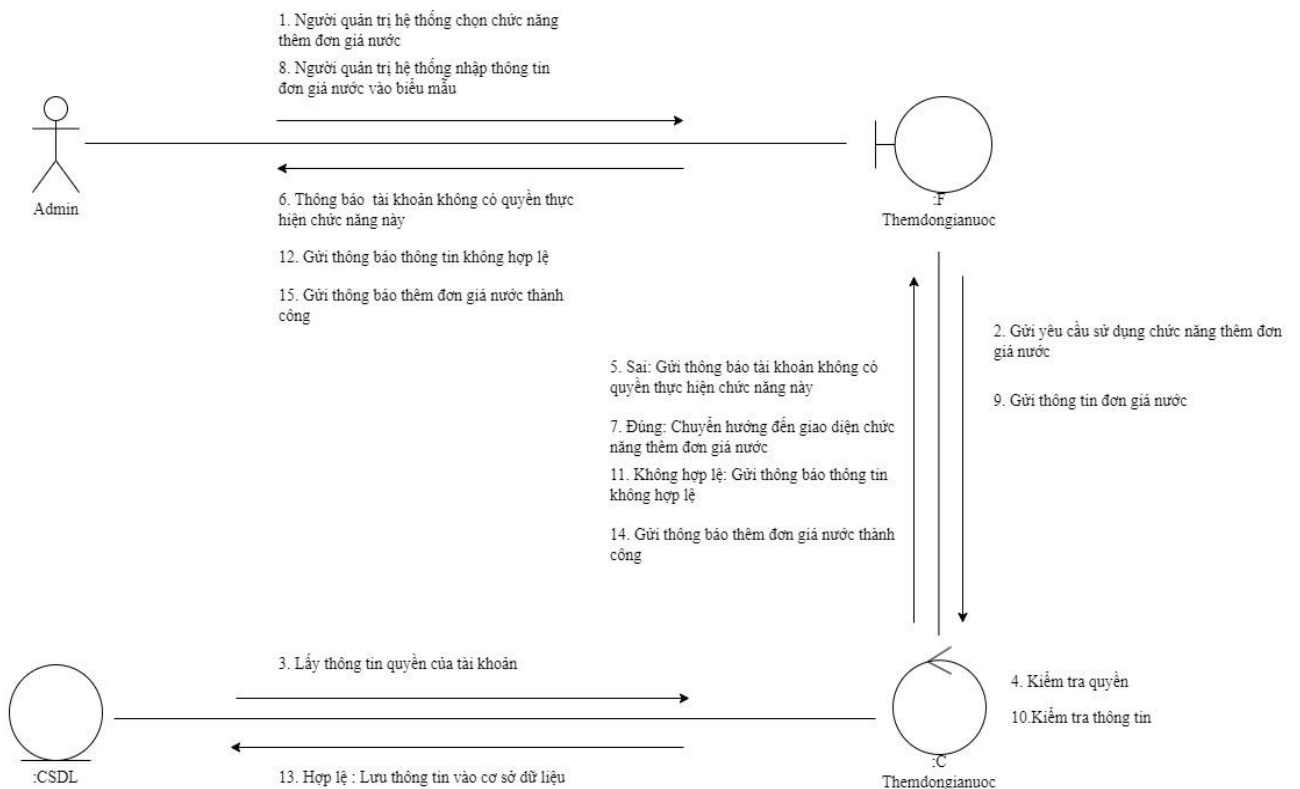
Hình 2.6.1: Biểu đồ cộng tác thực thi Use Case "Đăng nhập"



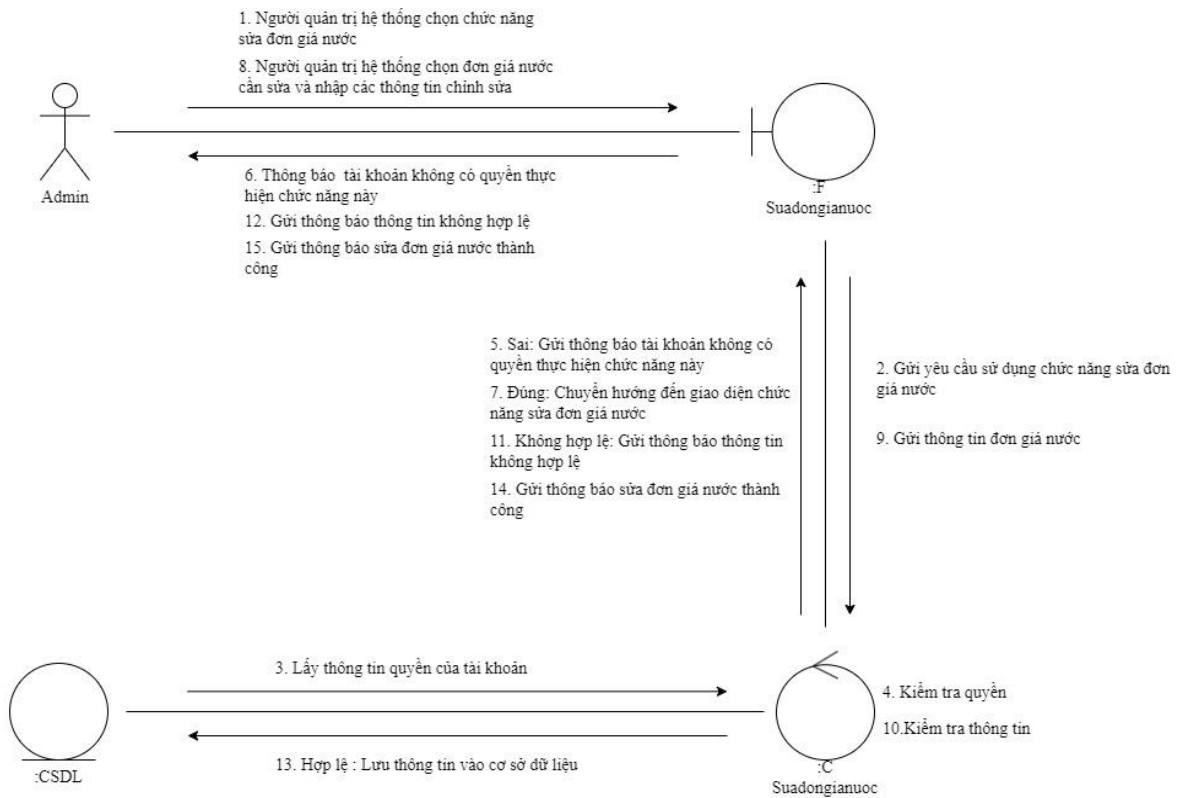
Hình 2.6.2: Biểu đồ cộng tác thực thi Use Case "Tạo tài khoản"



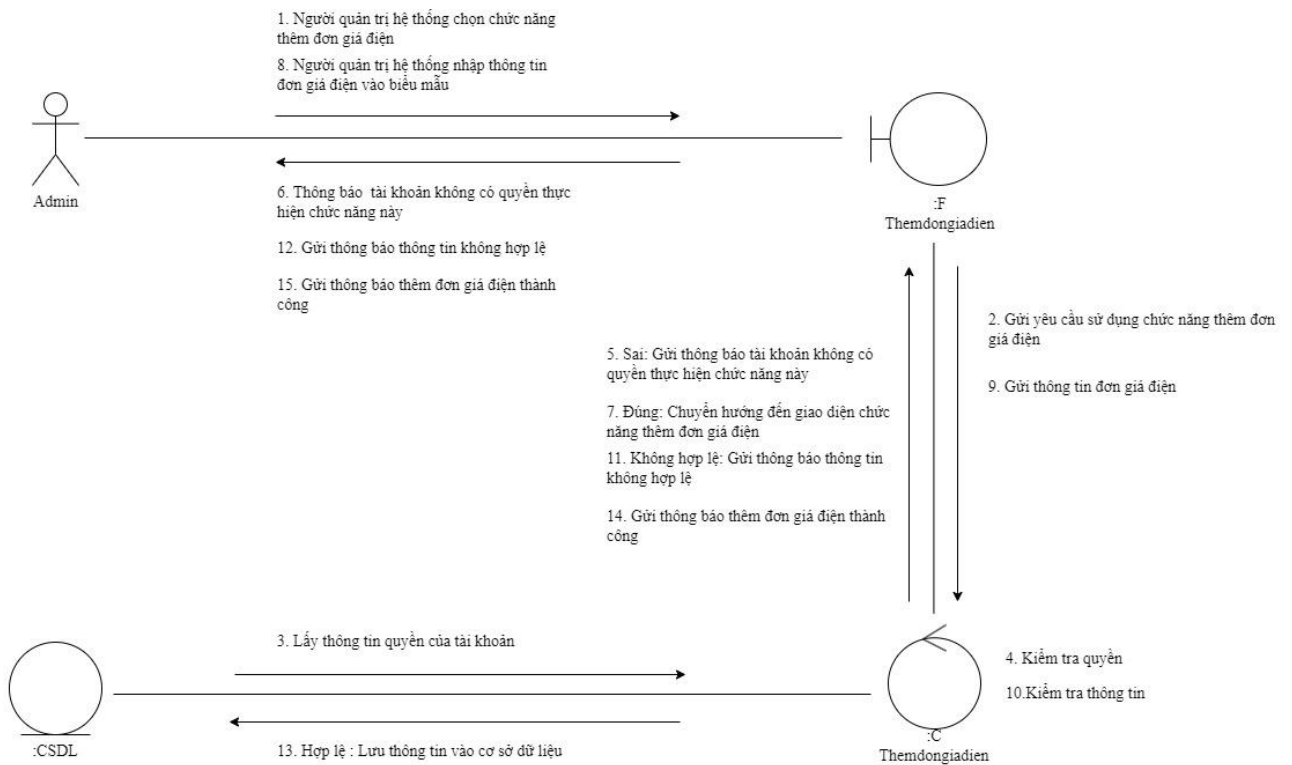
Hình 2.6.3: Biểu đồ cộng tác thực thi Use Case "Phân quyền"



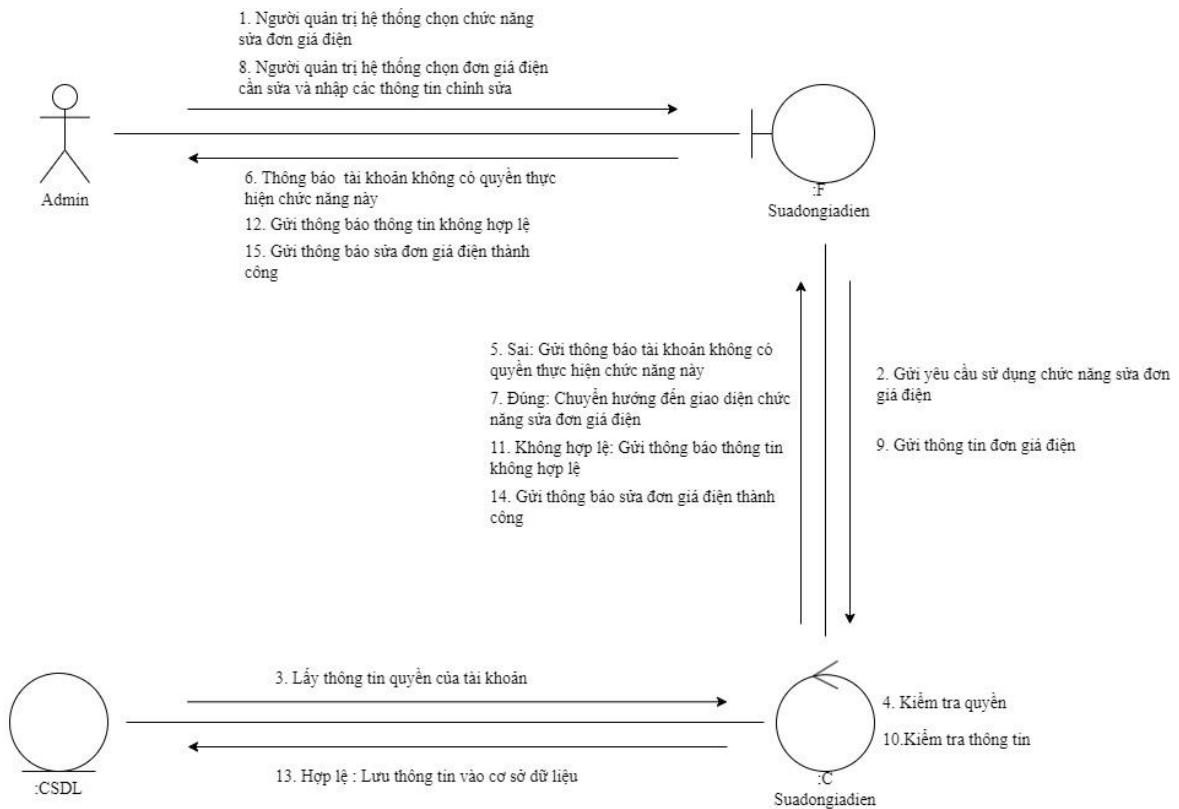
Hình 2.6.4: Biểu đồ cộng tác thực thi Use Case "Thêm đơn giá nước"



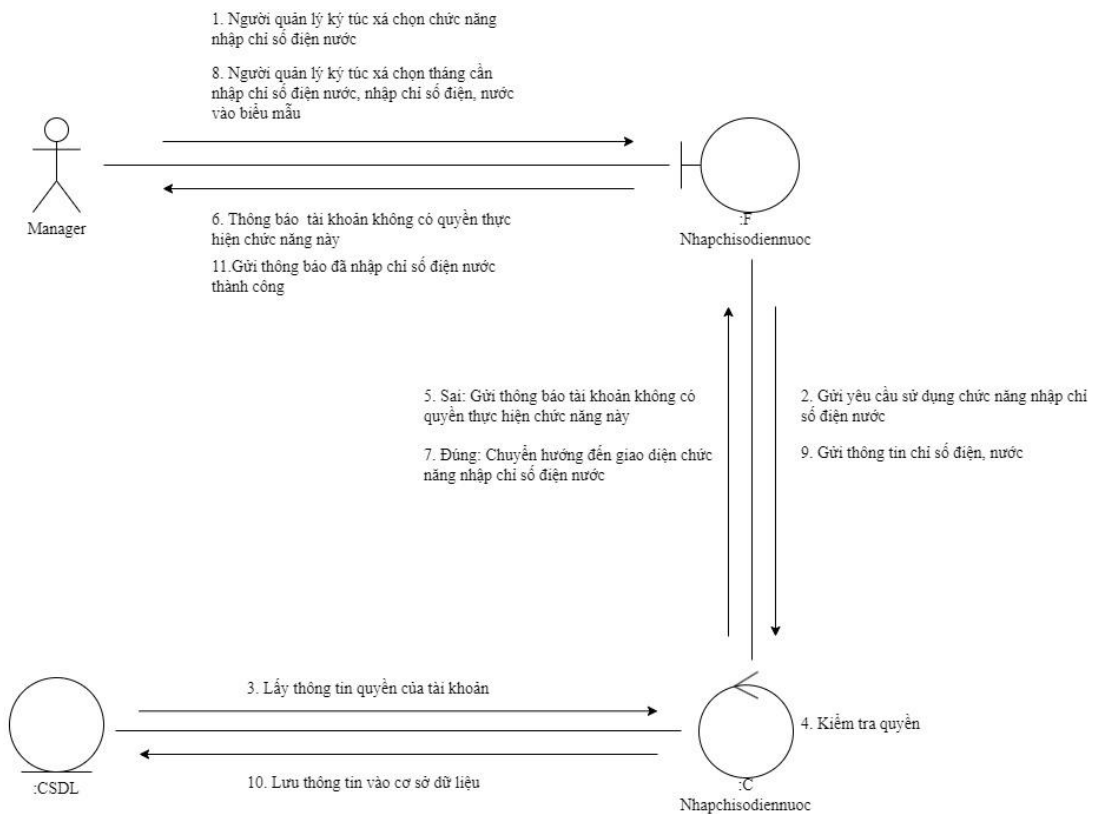
Hình 2.6.5: Biểu đồ cộng tác thực thi Use Case "Sửa đơn giá nước"



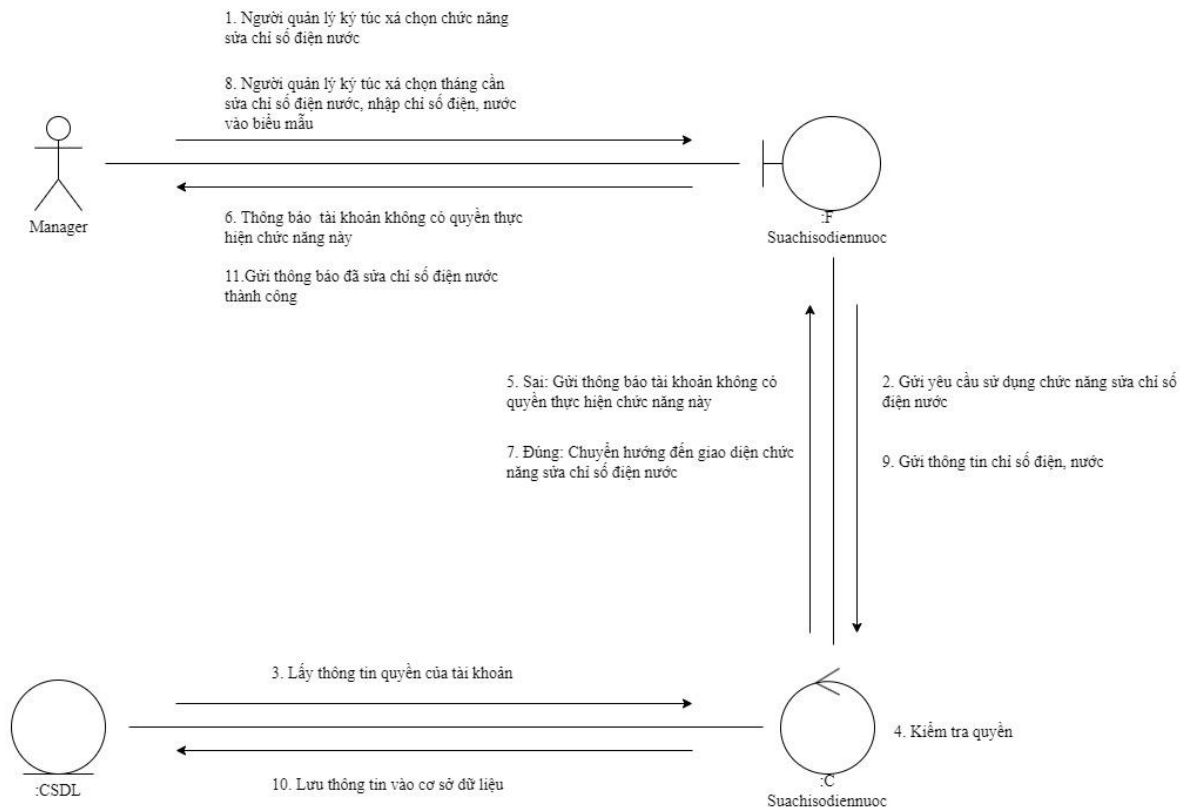
Hình 2.6.6: Biểu đồ cộng tác thực thi Use Case "Thêm đơn giá điện"



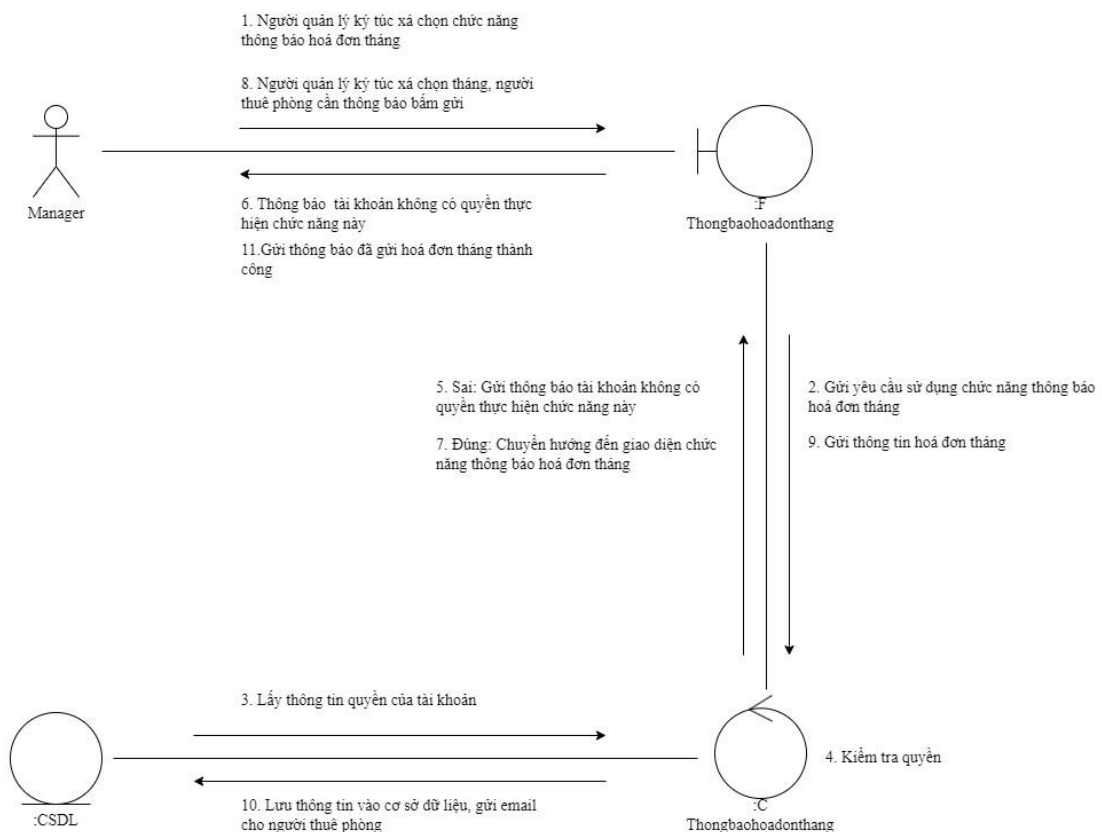
Hình 2.6.7: Biểu đồ cộng tác thực thi Use Case "Sửa đơn giá điện"



Hình 2.6.8: Biểu đồ cộng tác thực thi Use Case "Nhập chỉ số điện, nước"

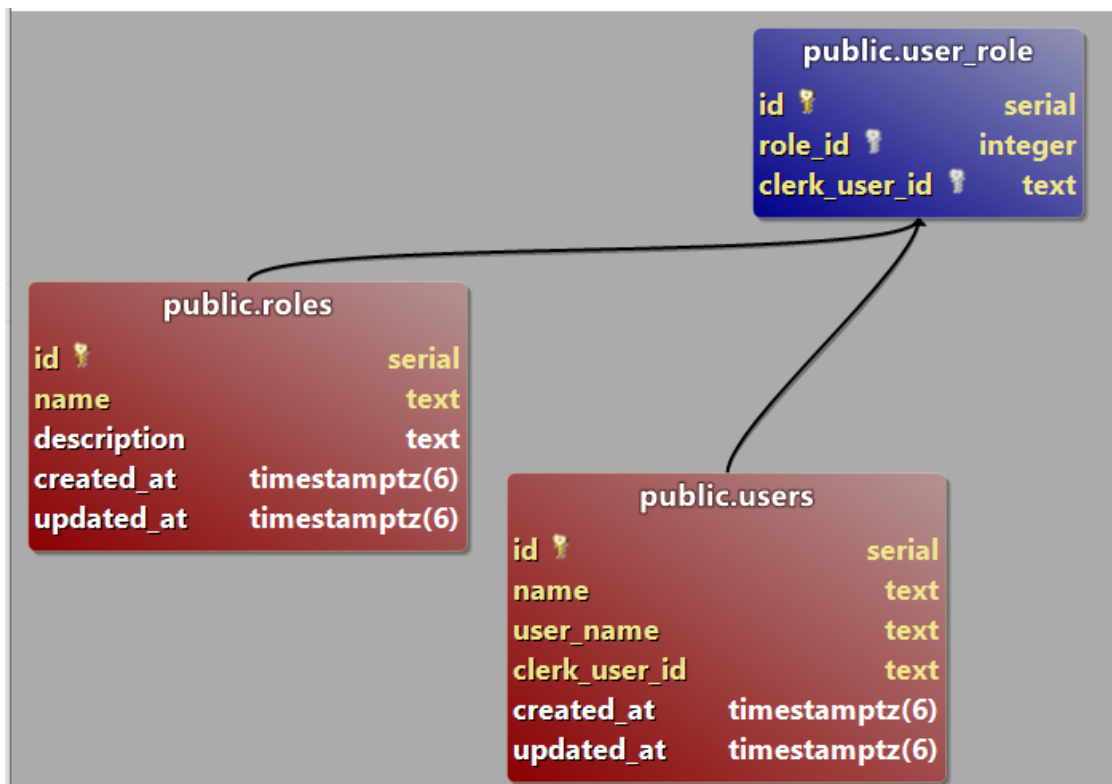


Hình 2.6.9: Biểu đồ cộng tác thực thi Use Case "Sửa chỉ số điện, nước"



Hình 2.6.10: Biểu đồ cộng tác thực thi Use Case "Thông báo hoá đơn tháng"

2.7. Mô hình quan hệ



Hình 2.7.1: Mô hình quan hệ giữa người dùng và quyền

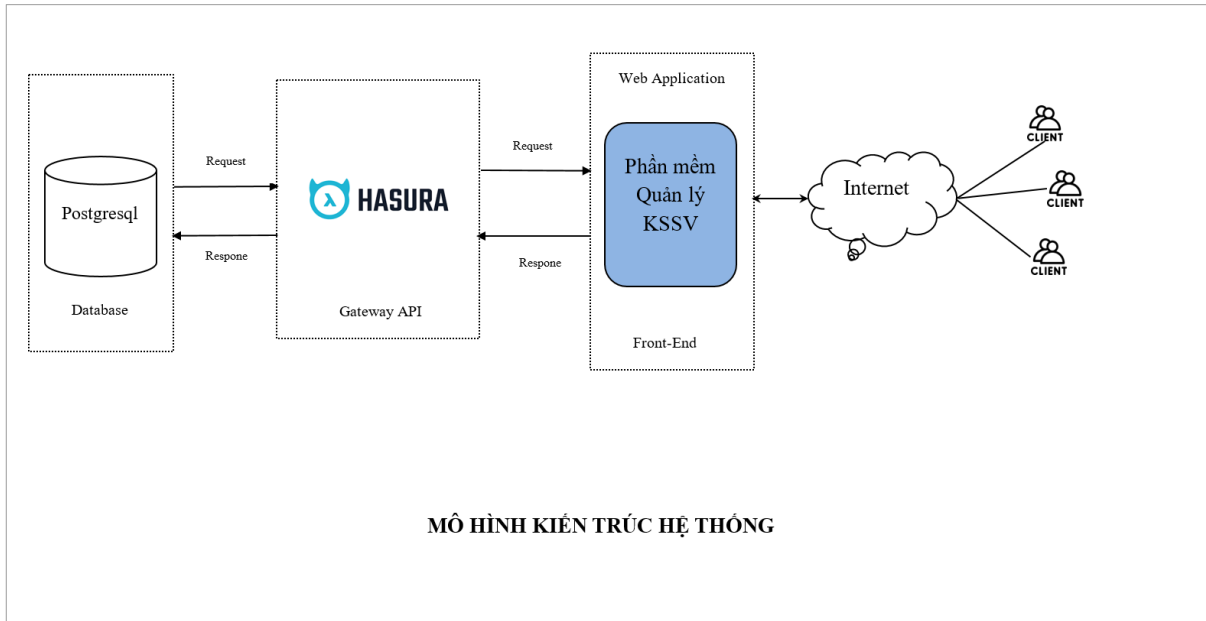


Hình 2.7.2: Mô hình quan hệ giữa khu vực, phòng, hợp đồng thuê nhà, hoá đơn tháng



Hình 2.7.3: Mô hình quan hệ giữa người thuê, loại người thuê, giới tính, hợp đồng thuê phòng

CHƯƠNG 3. LẬP TRÌNH VÀ THỬ NGHIỆM



Hình 3.1: Mô hình tổng quát của hệ thống

3.1. Xây dựng cơ sở dữ liệu

3.1.1. Tạo các bảng dữ liệu

Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment	Expression	Collation
id	serial	☐	serial	nextval('categor...	☒			
name	text	☐	text		☐			
description	text	☐	text		☐	Mô tả		
total_room	integer	☐	integer		☐	Tổng số phòng trong khu vực		
created_at	timestampz	☒	timestampz(6)		☐			
updated_at	timestampz	☒	timestampz(6)		☐			

Hình 3.1.1: Bảng danh mục khu vực

Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment	Expression	Collation
id	serial	☐	serial	nextval('categor...	☒			
name	text	☐	text		☐			
description	text	☒	text		☐			
created_at	timestampz	☒	timestampz(6)		☐			
updated_at	timestampz	☒	timestampz(6)		☐			

Hình 3.1.2: Bảng danh mục giới tính

Design Table - public.category_rooms							
	Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment
9	id	serial	☐	serial	nextval('categor...	☒	
	code	text	☐	text		☐	
	name	text	☐	text		☐	
	max_tenant	integer	☐	integer		☐	
	area_id	integer	☐	integer		☐	
	allow_female	boolean	☐	boolean	false	☐	
	created_at	timestampz	☒	timestampz(6)		☐	
	updated_at	timestampz	☒	timestampz(6)		☐	
	room_price	double precision	☒	double precision		☐	Giá phòng

Hình 3.1.3: Bảng danh mục phòng

Design Table - public.category_tenant_types								
	Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment	Collation
5	id	serial	☐	serial	nextval('categor...	☒		
	name	text	☐	text		☐		
	description	text	☒	text		☐		
	created_at	timestampz	☒	timestampz(6)		☐		
	updated_at	timestampz	☒	timestampz(6)		☐		

Hình 3.1.4: Bảng danh mục loại người thuê

Design Table - public.electricity_rates							
	Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment
6	id	serial	☐	serial	nextval('electrici...	☒	
	rate	double precision	☐	double precision		☐	
	start_date	date	☐	date		☐	
	end_date	date	☐	date		☐	
	created_at	timestampz	☒	timestampz(6)		☐	
	updated_at	timestampz	☒	timestampz(6)		☐	

Hình 3.1.5: Bảng đơn giá điện

Design Table - public.water_rates							
	Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment
6	id	serial	☐	serial	nextval('water_r...	☒	
	rate	double precision	☐	double precision		☐	
	start_date	date	☐	date		☐	
	end_date	date	☐	date		☐	
	created_at	timestampz	☒	timestampz(6)		☐	
	updated_at	timestampz	☒	timestampz(6)		☐	

Hình 3.1.6: Bảng đơn giá nước

Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment
id	serial	☐	serial	nextval('tenants...')	☒	
tenants_type_id	integer	☐	integer		☐	
gender_id	integer	☐	integer		☐	
name	text	☐	text		☐	họ và tên
date_of_birth	date	☒	date		☐	Ngày tháng năm sinh
address	text	☐	text		☐	Địa chỉ
phone_number	varchar	☒	varchar(10)		☐	Số điện thoại
email	text	☐	text		☐	tên email
cccd	varchar	☐	varchar(12)		☐	Số căn cước công dân
work_place	text	☐	text		☐	Nơi công tác
created_at	timestampz	☒	timestampz(6)		☐	
updated_at	timestampz	☒	timestampz(6)		☐	

Hình 3.1.7: Bảng người thuê

Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment
id	serial	☐	serial	nextval('rent_de...')	☒	
tenant_id	integer	☐	integer		☐	Mã người thuê
room_id	integer	☐	integer		☐	Mã phòng
start_date	date	☐	date		☐	Ngày vào
end_date	date	☒	date		☐	Ngày ra
advance_payment	double precision	☐	double precision		☐	Khoản tạm thu
status	boolean	☐	boolean	false	☐	Trạng thái khoản tạm thu (đóng ...
initial_electricity	double precision	☒	double precision		☐	Số điện khi bắt đầu vào
initial_water	double precision	☒	double precision		☐	Số nước khi bắt đầu vào
remaining_amount	double precision	☒	double precision		☐	
created_at	timestampz	☒	timestampz(6)		☐	
updated_at	timestampz	☒	timestampz(6)		☐	

Hình 3.1.8: Bảng hợp đồng thuê

Design Table - public.month_bill							
	Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment
18	id	serial	☐	serial	nextval('month_...)	☒	
	room_id	integer	☐	integer		☐	Mã phòng
	current_num_tenants	integer	☐	integer		☐	Số lượng người đang thuê phòng
	month	date	☒	date		☐	Tháng
	start_date	date	☒	date		☐	Ngày bắt đầu
	end_date	date	☒	date		☐	Ngày kết thúc
	room_price	double precision	☒	double precision		☐	Giá phòng
	start_electricity	double precision	☒	double precision		☐	Số điện tháng trước
	end_electricity	double precision	☐	double precision		☐	Số điện tháng này
	electricity_price	double precision	☒	double precision		☐	Đơn giá điện
	start_water	double precision	☒	double precision		☐	Số nước tháng trước
	end_water	double precision	☐	double precision		☐	Số nước tháng này
	water_price	double precision	☒	double precision		☐	Đơn giá nước
	total_water_charge	double precision	☒	double precision		☐	Tổng tiền nước
	total_electricity_charge	double precision	☒	double precision		☐	Tổng tiền điện
	total_revenue	double precision	☒	double precision		☐	Tổng tiền cả phòng
	created_at	timestampz	☒	timestampz(6)		☐	
	updated_at	timestampz	☒	timestampz(6)		☐	

Hình 3.1.9: Bảng hợp hoá đơn hàng tháng

Design Table - public.roles							
	Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment
5	id	serial	☐	serial	nextval('roles_id_...)	☒	
	name	text	☐	text		☐	
	description	text	☒	text		☐	
	created_at	timestampz	☒	timestampz(6)		☐	
	updated_at	timestampz	☒	timestampz(6)		☐	

Hình 3.1.10: Bảng danh mục quyền

Design Table - public.users							
	Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment
6	id	serial	☐	serial	nextval('users_i_...)	☒	
	name	text	☐	text		☐	
	user_name	text	☐	text		☐	
	clerk_user_id	text	☐	text		☐	
	created_at	timestampz	☒	timestampz(6)		☐	
	updated_at	timestampz	☒	timestampz(6)		☐	

Hình 3.1.11: Bảng người dùng

	Column Name	Data Type	Null	Data Type(Length)	Default value	Auto	Comment
9	id	serial	☐	serial	nextval('user_rol...)	☒	
3	role_id	integer	☐	integer		☐	
	clerk_user_id	text	☐	text		☐	

Hình 3.1.12: Bảng quyền của người dùng

3.1.2. Tạo các ràng buộc về dữ liệu

Một số ràng buộc được tạo:

- Một người dùng chỉ được có 1 “clerk_user_id”
- Tên tài khoản (username) của người dùng không được trùng
- Mỗi 1 phòng 1 tháng chỉ được phép có 1 hoá đơn
- Các dữ liệu ngày bắt đầu (start_date) phải nhỏ hơn ngày kết thúc (end_date)
- Các đơn giá điện, đơn giá nước có khoảng thời gian áp dụng không được đan xe đè lên nhau
- Số người trong phòng không vượt quá số người giới hạn
- Một người thuê chỉ có thể thuê 1 phòng trong 1 khoảng thời gian

3.1.3. Tạo trigger tính toán

Trigger là một thủ tục SQL được thực thi ở phía server khi có một sự kiện như Insert, Delete, hay Update. Trigger là một loại stored procedure đặc biệt (không có tham số) được thực thi (execute) một cách tự động khi có một sự kiện thay đổi dữ liệu (data modification). Trigger được lưu trữ và quản lý trong Server DB, được dùng trong trường hợp ta muốn kiểm tra các ràng buộc toàn vẹn trong DB.

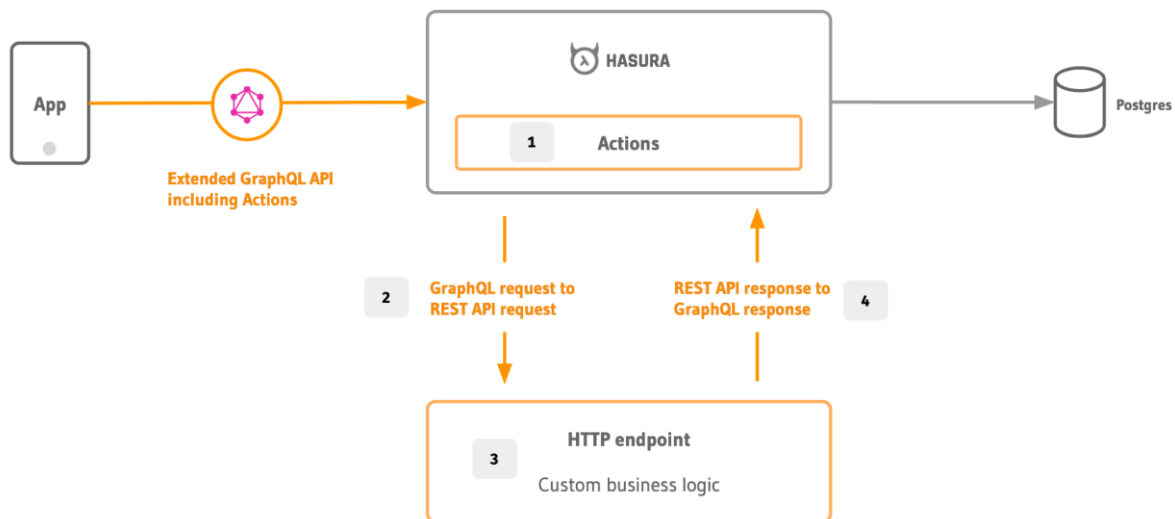
Trong đây, ta sử dụng trigger để:

- Tự động lấy chỉ số điện, chỉ số nước của tháng trước
- Tự động lấy đơn giá nước, đơn giá điện đang được áp dụng
- Tự động tính toán tiền nước, tiền điện, tổng tiền của phòng

Mỗi khi người dùng thêm hoặc chỉ sửa dữ liệu của bảng hoá đơn tháng (month_bill). Do đó, khi sử dụng, người dùng chỉ cần nhập các chỉ số điện, nước vào cuối tháng. Các thông tin khác trong hoá đơn sẽ được tự động tính.

3.2. Tạo API (Application Programming Interface)

API là các phương thức, giao thức kết nối với các thư viện và ứng dụng khác. Nó là viết tắt của Application Programming Interface – giao diện lập trình ứng dụng. API cung cấp khả năng truy xuất đến một tập các hàm hay dùng. Và từ đó có thể trao đổi dữ liệu giữa các ứng dụng.



Hình 3.2.1: Mô hình hoạt động của hasura

Hasura được sử dụng để tạo các API phục vụ tương tác giữa giao diện và cơ sở dữ liệu bằng GraphQL. GraphQL là ngôn ngữ dùng để thao tác và truy vấn dữ liệu cho API, cung cấp cho client 1 cách thức dễ dàng để request chính xác những gì họ cần, giúp việc phát triển API dễ dàng hơn.

Với GraphQL, chúng ta chỉ cần expose một API cho thông tin sinh viên, client có thể sử dụng API này để query đúng thông tin cần thiết.

3.3. Xây dựng giao diện

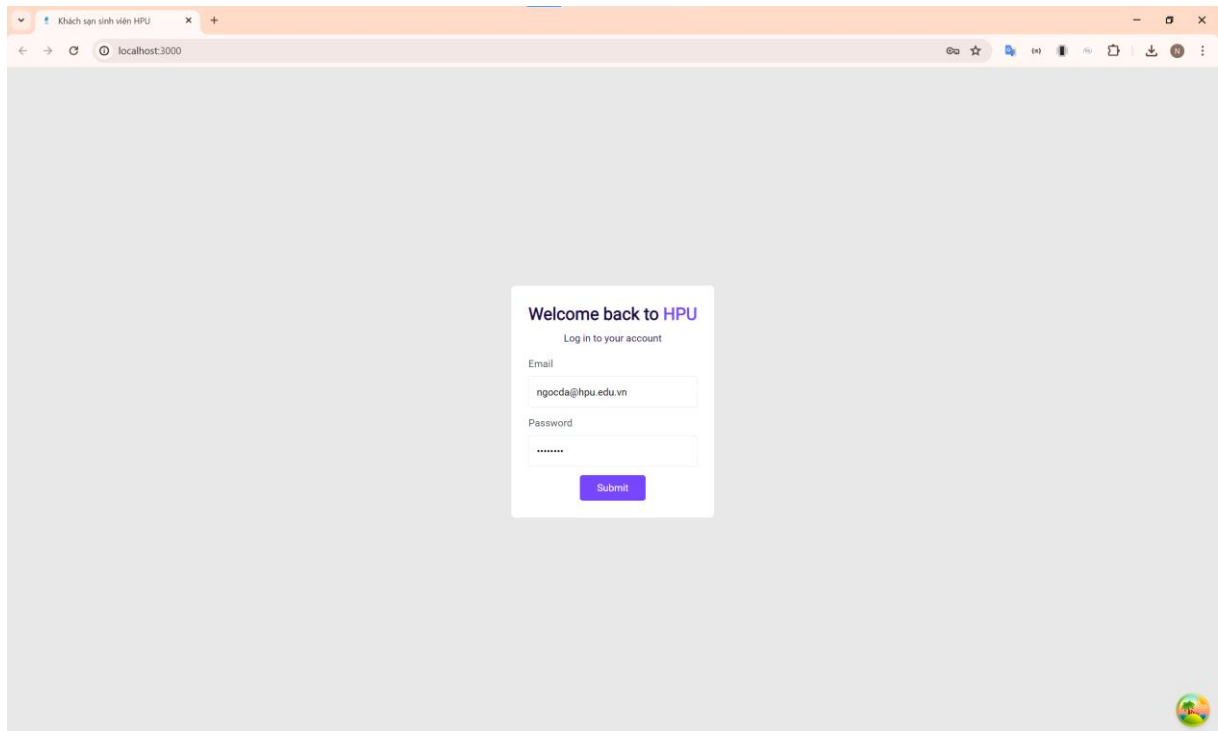
ReactJS là một thư viện JavaScript được phát triển bởi Facebook nhằm xây dựng giao diện người dùng(UI).

JSX là một phần quan trọng trong ReactJS. JSX giống như một syntax extension giúp code trở lên dễ mượt và dễ đọc hơn.

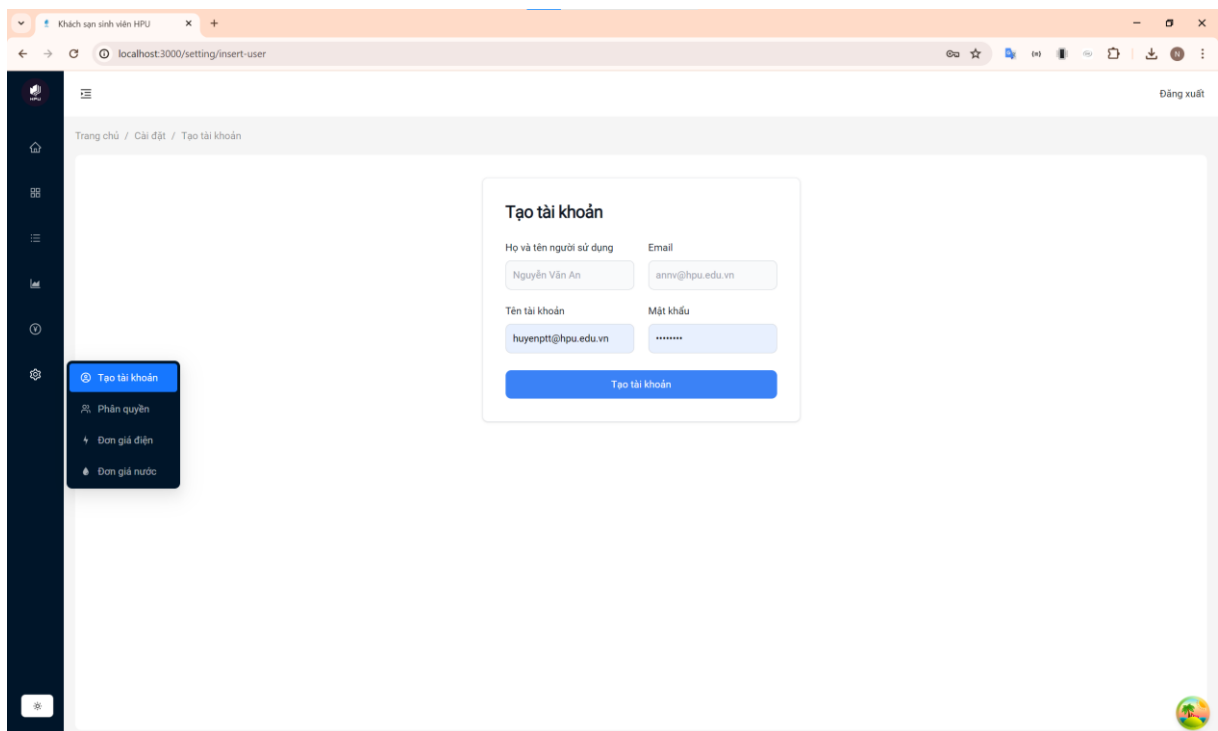
JSX = Javascript + XML. Nó transform cú pháp dạng gần như XML về thành Javascript. Giúp người lập trình có thể code ReactJS bằng cú pháp của

XML thay vì sử dụng Javascript. Các XML elements, attributes và children được chuyển đổi thành các đối số truyền vào React.createElement.

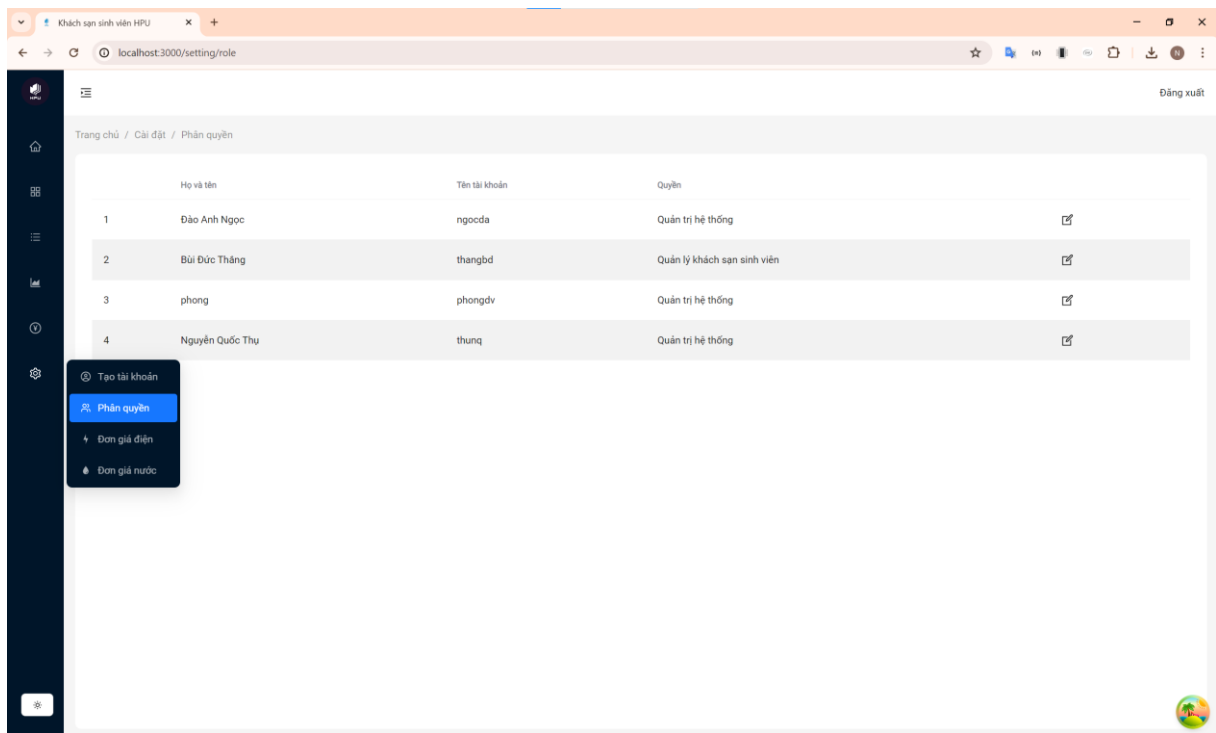
Các hình ảnh giao diện :



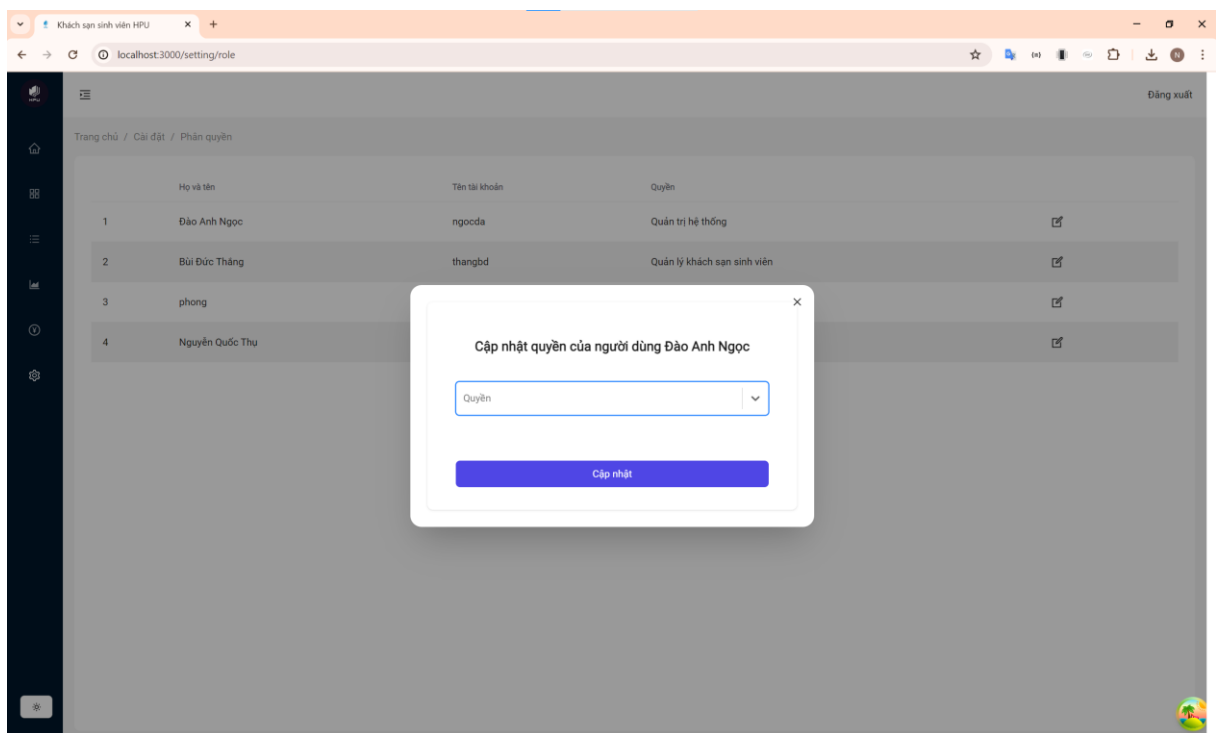
Hình 3.3.1: Giao diện đăng nhập



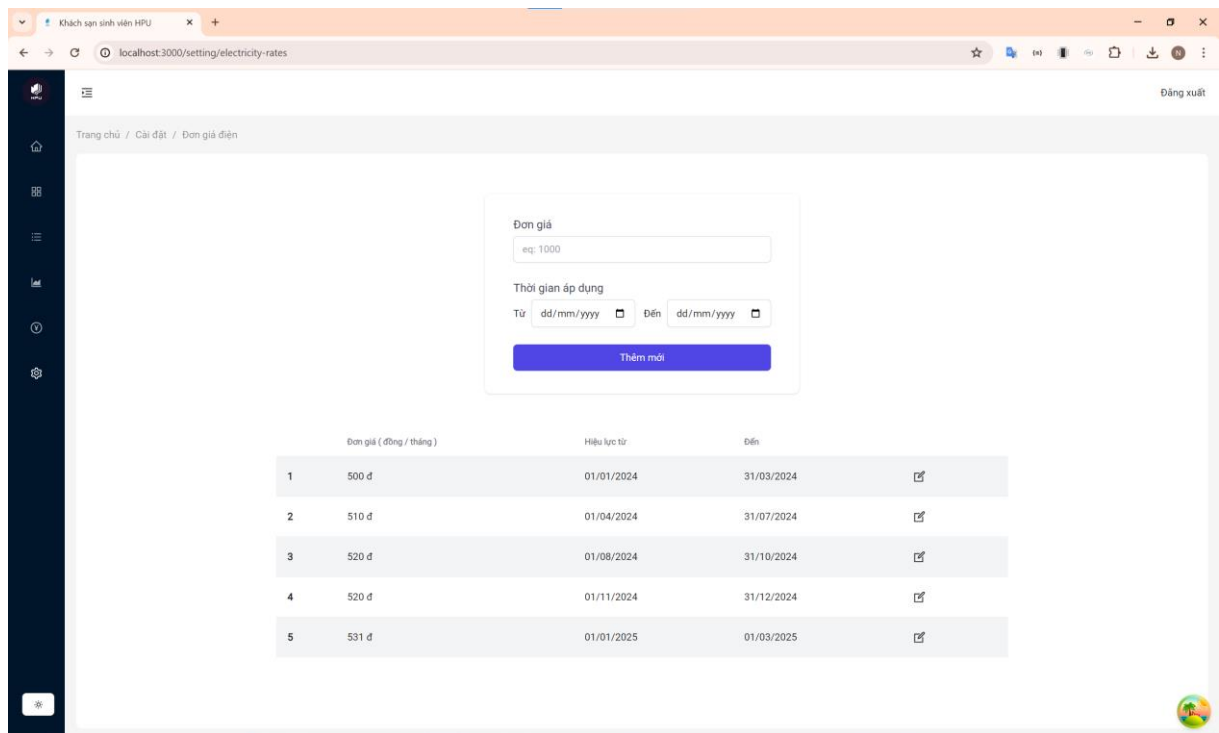
Hình 3.3.2: Tạo tài khoản người dùng



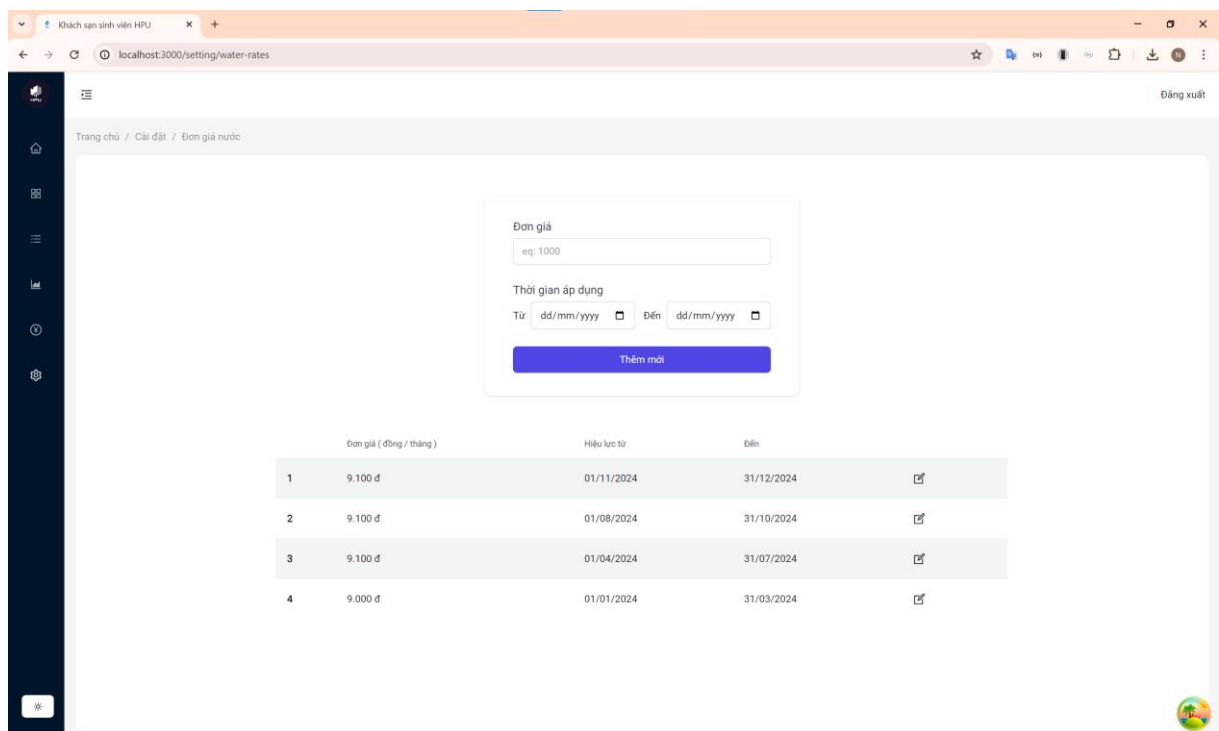
Hình 3.3.3: Giao diện phân quyền



Hình 3.3.4: Giao diện phân quyền



Hình 3.3.5: Giao diện đơn giá điện



Hình 3.3.6: Giao diện đơn giá nước

Trang chủ / Chỉ số điện, nước

Đăng xuất

Chọn tháng: Tháng Năm 2024

Chọn ngày bắt đầu: 01/05/2024

Chọn ngày kết thúc: 31/05/2024

Xác nhận

Phòng	Số điện	Số nước
1 B201	501	501
2 B202	502	502

Hình 3.3.7: Giao diện chỉ số điện nước

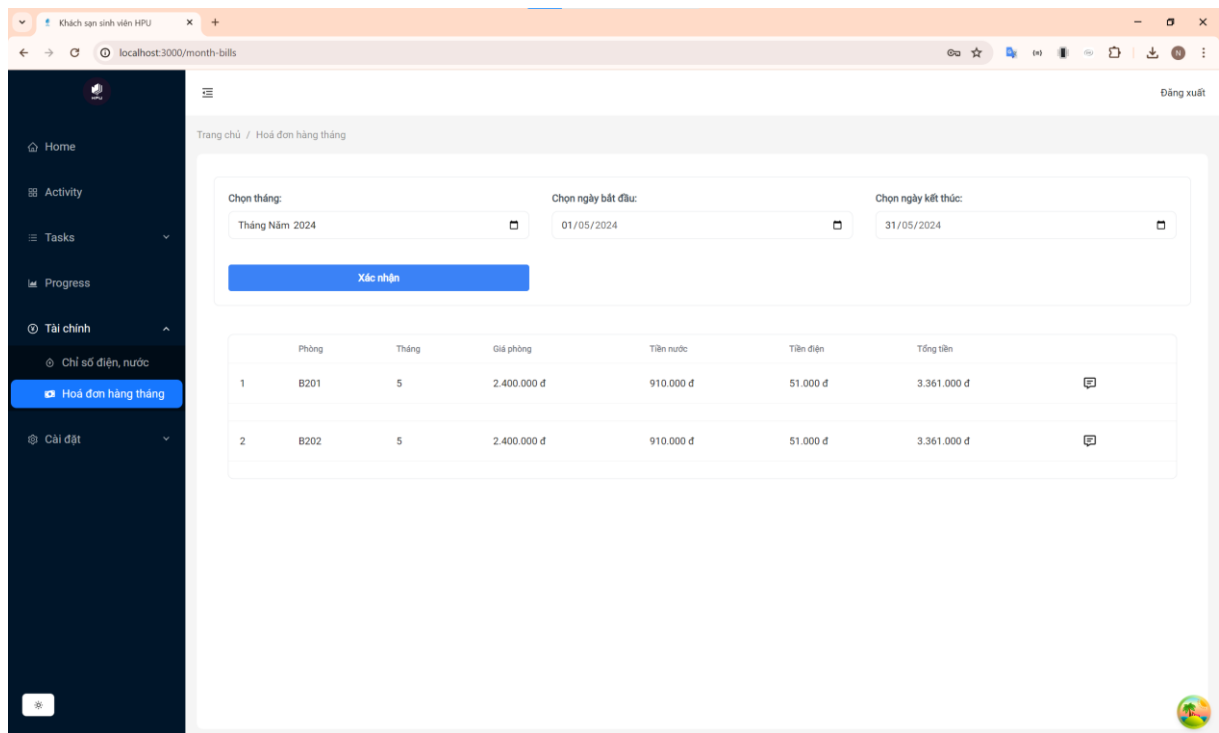
Chỉnh sửa chỉ số điện, nước

Số nước: 501

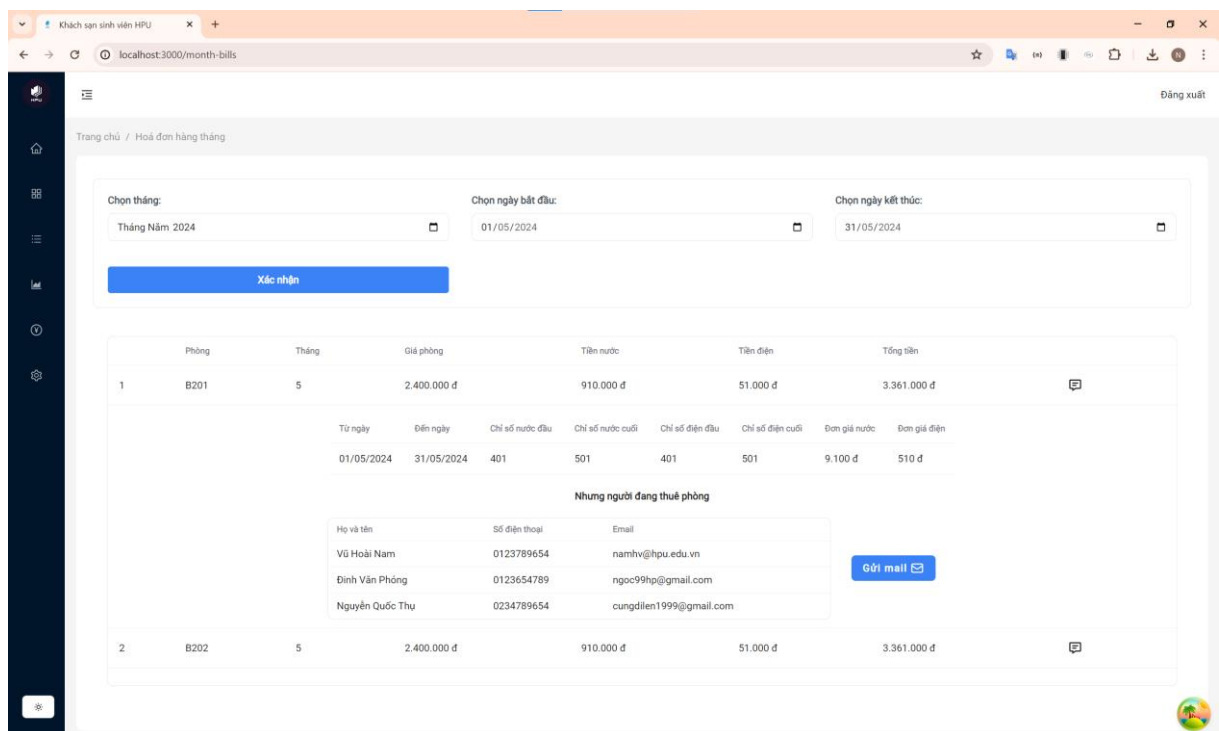
Số nước: 501

Cập nhật

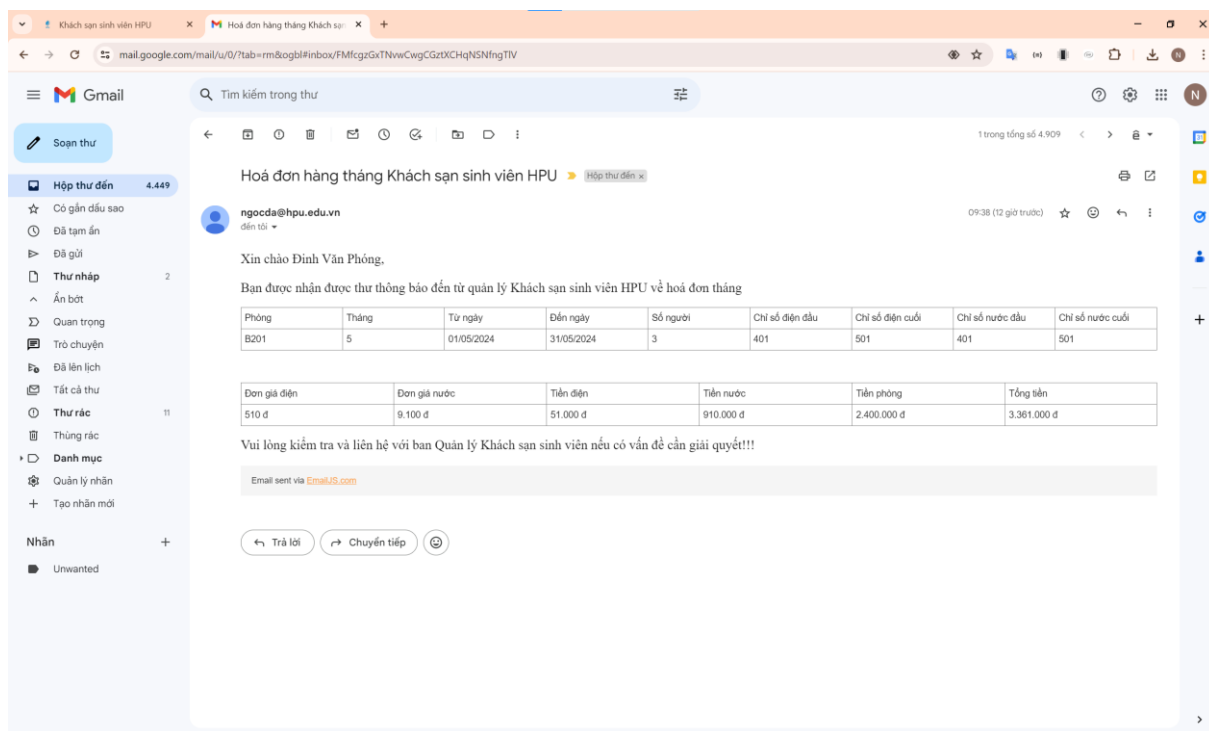
Hình 3.3.8: Giao diện chỉnh sửa chỉ số điện nước



Hình 3.3.9: Giao diện hoá đơn hàng tháng



Hình 3.3.10: Giao diện hoá đơn hàng tháng chi tiết



Hình 3.3.11: Giao diện email thông báo hoá đơn hàng tháng cho người thuê

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. Kết quả đạt được

- Xây dựng trang web hoàn chỉnh: Đã nắm vững quy trình xây dựng một trang web hoàn chỉnh, bao gồm cả phần frontend(giao diện người dùng) và backend(xử lý logic và dữ liệu).
- Phát triển các chức năng chính của trang web.
- Biết cách sử dụng các công nghệ mới vào trong dự án như : Hasura, Clerk, Vercel, ...

2. Hạn chế

- Thiếu chức năng: Do thời gian làm bài có hạn, trang web hiện tại vẫn chưa đầy đủ các chức năng cần thiết để hoàn thiện một hệ thống quản lý khách sạn sinh viên ví dụ như : các chức năng tìm kiếm, tính tiền cho đối tượng được miễn giảm,...
- Tối ưu hoá hiệu năng của hệ thống còn chưa được tối ưu

3. Hướng phát triển

- Hoàn thiện các chức năng còn thiếu : Tiếp tục phát triển và bổ sung các chức năng cần thiết còn thiếu cho trang web, đảm bảo hệ thống quản lý tài chính của khách sạn sinh viên được hoàn thiện và hiệu quả.
- Tìm hiểu và học tập thêm các kiến thức lập trình web:
 - + Nâng cao kiến thức và kỹ năng lập trình web, học thêm nhiều công nghệ mới như Nextjs,...
 - + Học tập về các kỹ thuật bảo mật, tối ưu hiệu suất và quản lý dự án phần mềm.
- Cải thiện ràng buộc dữ liệu: Áp dụng các biện pháp kiểm tra và ràng buộc dữ liệu đầu vào, đảm bảo dữ liệu nhập vào hệ thống là chính xác và hợp lệ, giảm thiểu rủi ro và lỗi.
- Tích hợp thêm các dịch vụ và API: Khám phá và tích hợp các dịch vụ và API bên thứ ba để mở rộng tính năng và khả năng của trang web, như thanh toán trực tuyến, và các dịch vụ phân tích dữ liệu.
- Cải thiện giao diện người dùng (UI/UX) : Tập trung vào cải thiện giao diện và trải nghiệm người dùng, đảm bảo trang web dễ sử dụng, thân thiện và tương tác tốt trên nhiều thiết bị.

Bằng cách tiếp tục phát triển và hoàn thiện trang web quản lý khách sạn sinh viên, không chỉ nâng cao kỹ năng lập trình mà còn đóng góp vào việc cải thiện chất lượng dịch vụ và hiệu quả quản lý trong môi trường giáo dục.

TÀI LIỆU THAM KHẢO

- [1]. Ramez A. Elmasri, Shamkant Navathe, Fundamentals of Database Systems, 7th edition, John Wiley & Sons, Inc., 2016.
- [2]. Đoàn Thanh Nghị, Giáo trình Lập trình Web, Đại học Quốc gia TP.HCM, 2022.
- [3]. Nguyễn Tuệ, Giáo trình nhập môn hệ cơ sở dữ liệu , Nhà xuất bản giáo dục (tái bản lần thứ nhất), 2009.
- [4]. Trương Ninh Thuận, Đặng Đức Hạnh, Giáo trình Phân tích thiết kế hướng đối tượng, Đại học Quốc gia Hà Nội, 2016.
- [5]. <https://vercel.com/>
- [6]. <https://react.dev/learn>
- [7]. <https://tailwindcss.com/docs/installation>
- [8]. <https://clerk.com/docs>