

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



BÁO CÁO BÀI TẬP LỚN
ĐỀ TÀI :QUẢN LÝ SỔ TIẾT KIỆM

Giảng viên hướng dẫn: ThS.TRẦN PHONG NHÃ

Sinh viên thực hiện: Lê Công Thuận

Nguyễn Ngô Thành Tài

Trịnh Nguyễn Vy Na

Lớp: CQ.62.CNTT

Khóa: 62

Tp. Hồ Chí Minh, năm 2024

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



BÁO CÁO BÀI TẬP LỚN
ĐỀ TÀI : QUẢN LÝ SỔ TIẾT KIỆM

Giảng viên hướng dẫn: ThS. TRẦN PHONG NHÃ

Sinh viên thực hiện: Lê Công Thuận - 6251071100

Trịnh Nguyễn Vy Na - 6251071001

Nguyễn Ngô Thành Tài - 6251071003

Lớp: CQ.62.CNTT

Khóa: 62

Tp. Hồ Chí Minh, năm 2024

NHIỆM VỤ THIẾT KẾ BÀI TẬP LỚN
BỘ MÔN: CÔNG NGHỆ THÔNG TIN

-----***-----

1. Đề tài.

Thiết kế và xây dựng chương trình quản lý sổ tiết kiệm trên website

2. Mục đích, yêu cầu.

a. Mục đích.

- Tự động hóa quy trình quản lý: Giảm thiểu sai sót do nhập liệu thủ công và tối ưu hóa thời gian xử lý giao dịch, thay thế phương pháp quản lý truyền thống bằng một hệ thống số hóa toàn diện
- Nâng cao hiệu quả hoạt động: Cung cấp công cụ hỗ trợ quản lý thông tin sổ tiết kiệm, giao dịch gửi tiền, rút tiền và báo cáo chi tiết, dễ dàng tra cứu và tổng hợp thông tin khi cần thiết
- Tăng tính chính xác và bảo mật: Đảm bảo thông tin tài khoản và giao dịch được quản lý an toàn và bảo mật, ngăn ngừa các trường hợp gian lận hoặc sai sót trong giao dịch
- Cung cấp báo cáo quản trị: Hỗ trợ quản trị viên trong việc theo dõi doanh thu, quản lý sổ tiết kiệm, và thống kê chi tiết theo từng tháng.

b. Yêu cầu.

- Giao diện: Thân thiện, dễ sử dụng, cấu hình nhẹ.
- Tổ chức dữ liệu rõ ràng, nhất quán

3. Nội dung và phạm vi đề tài.

a. Nội dung đề tài.

- Tổng quan về bài toán
- Tổng quan về ngôn ngữ lập trình C#
- Tổng quan về SQL

- Tổng quan về SQL Server
- Tổng quan về Visual Studio
- b. Phạm vi đề tài.
 - Tạo ra chương trình website quản lý sổ tiết kiệm.

4. Công nghệ, công cụ và ngôn ngữ lập trình.

- a. Công nghệ: Github.
- b. Công cụ: Visual Studio, SQL Server.
- c. Ngôn ngữ lập trình: C#.

5. Các kết quả chính dự kiến sẽ đạt được và ứng dụng.

Quyển báo cáo đề tài bài tập lớn được chia thành 4 phần như sau:

- Chương 1: Tổng quan về bài toán
 - + Tổng quan về đề tài.
 - + Mục tiêu nghiên cứu.
 - + Đối tượng nghiên cứu
 - + Phạm vi.
- Chương 2: Cơ sở lí thuyết.
 - + Tổng quan về ngôn ngữ lập trình C#.
 - + Tổng quan về SQL
 - + Tổng quan về SQL Server
 - + Tổng quan về Visual Studio
 - + Tổng quan về Bootstrap
- Chương 3: Phân tích bài toán
 - + Mô tả bài toán
 - + Phân tích yêu cầu
 - + Sơ đồ phân rã chức năng BFD
 - + Sơ đồ luồng dữ liệu DFD
 - + Sơ đồ Usecase
 - + Sơ đồ hoạt động - activity diagram
 - + Sơ đồ tuần tự
 - + Sơ đồ class
- Chương 4: Thiết kế và cài đặt chương trình

- + Kiến trúc hệ thống
- + Thiết kế và xây dựng hệ thống.
- + Thiết kế giao diện
- + Test Case
- Chương 5: Kết luận và hướng phát triển:
 - + Kết quả thực hiện được
 - + Đề xuất phát triển

6. Giáo viên và cán bộ hướng dẫn.

Họ tên: Trần Phong Nhã

Đơn vị công tác: Bộ môn Công Nghệ Thông Tin – Trường Đại học Giao thông Vận tải phân hiệu tại TP HCM

Điện thoại:

Email

LỜI CẢM ƠN

Lời đầu tiên em xin phép gửi đến Quý Thầy Cô của Bộ môn Công Nghệ Thông Tin trường Đại học Giao thông Vận tải - Phân hiệu tại Tp Hồ Chí Minh lời chúc sức khỏe và lời cảm ơn sâu sắc nhất. Thầy cô đã tận tình hướng dẫn, truyền đạt kiến thức, kinh nghiệm cho chúng em trong suốt quá trình học tập và rèn luyện ở trường, chúc thầy cô luôn mạnh khỏe và thu được nhiều thành công hơn nữa trong tương lai.

Đặc biệt, em xin gửi lời cảm ơn đến thầy ThS.Trần Phong Nhã, thầy đã tận tình hướng dẫn chỉ bảo để em có thể tiếp cận với nhiều bài toán thực tế để hoàn thành bài tập lớn ‘Quản lý sổ tiết kiệm’.

Mặc dù nhóm em đã cố gắng hết sức trong quá trình thực hiện bài tập lớn này, nhưng vì thời gian có hạn và nhóm em cũng chưa có kinh nghiệm làm việc với bài toán thực tế, nên chắc chắn sẽ có nhiều thiếu sót.

Lời sau cùng, em xin kính chúc thầy cô trong bộ môn Công Nghệ Thông Tin trường Đại học Giao thông Vận tải - Phân hiệu tại Tp Hồ Chí Minh luôn mạnh khỏe và thu được nhiều thành công hơn nữa trong tương lai.

Em xin chân thành cảm ơn!

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

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

Giảng viên hướng dẫn

MỤC LỤC

NHIỆM VỤ THIẾT KẾ BÀI TẬP LỚN	1
LỜI CẢM ƠN	4
NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN.....	5
MỤC LỤC.....	6
DANH MỤC HÌNH ẢNH	10
DANH MỤC THUẬT NGỮ	12
CHƯƠNG 1. MỞ ĐẦU	1
1.1 Tổng quan về đề tài.	1
1.2. Mục tiêu nghiên cứu.....	1
1.3 Đối tượng nghiên cứu.....	2
1.4. Phạm vi.....	2
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	3
2.1. Tổng quan về ngôn ngữ lập trình C#.....	3
2.1.1. Khái niệm.....	3
2.1.2. Ưu điểm	3
2.1.3. Nhược điểm.....	4
2.2. Tổng quan về SQL.....	4
2.2.1. Khái niệm.....	4
2.2.2. Ưu điểm	5
2.3. Tổng quan về SQL Server	6
2.3.1. Khái niệm.....	6
2.3.2. Ưu điểm	7
2.3.3. Nhược điểm.....	8
2.4. Tổng quan về Microsoft Visual Studio	9
2.4.1. Khái niệm.....	9
2.4.2. Ưu điểm	9
2.4.3. Nhược điểm.....	10
2.5. Tổng quan về Bootstrap.....	10
2.5.1 Khái niệm	10
2.5.2 Ưu điểm	11
2.5.3 Nhược điểm	12
CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG.....	13
3.1. Mô tả bài toán:.....	13
3.2. Phân tích yêu cầu.....	13
3.2.1. Yêu cầu chức năng.....	13

3.2.2. Yêu cầu phi chức năng.....	15
3.3. Sơ đồ phân rã chức năng (BFD).....	16
3.4. Sơ đồ luồng dữ liệu DFD	16
3.4.1. Sơ đồ DFD mức khung cảnh (mức 0).....	16
3.4.2. Sơ đồ DFD mức đỉnh (mức 1)	17
3.4.2. Sơ đồ DFD mức dưới đỉnh (mức 2).....	17
3.5. Sơ đồ Usecase.....	20
3.5.1. Sơ đồ Usecase tổng quát	20
3.5.2 Sơ đồ Usecase đăng ký	20
3.5.3 Sơ đồ Usecase đăng nhập	22
3.5.4 Sơ đồ Usecase mở sổ tiết kiệm	25
3.5.5 Sơ đồ UC lập phiếu gửi tiền.....	28
3.5.6 Sơ đồ Usecase lập phiếu rút tiền.....	31
3.5.7 Sơ đồ Usecase tra cứu thông tin.....	34
3.5.8 Sơ đồ Usecase lập báo cáo.....	37
3.5.9 Sơ đồ Usecase thay đổi quy định.....	40
3.6.Sơ đồ hoạt động - AD.....	43
3.6.1. Sơ đồ hoạt động miêu tả người dùng đăng ký	43
3.6.2. Sơ đồ hoạt động miêu tả người dùng đăng nhập	44
3.6.3. Sơ đồ hoạt động miêu tả người dùng mở sổ tiết kiệm.....	45
3.6.4. Sơ đồ hoạt động miêu tả người dùng Lập Phiếu Gửi Tiền.....	46
3.6.5. Sơ đồ hoạt động miêu tả người dùng Lập Phiếu Rút Tiền:	47
3.6.6. Sơ đồ hoạt động miêu tả người dùng Tra Cứu Thông Tin:	48
3.6.7. Sơ đồ hoạt động miêu tả người dùng Lập Báo Cáo:.....	49
3.6.8. Sơ đồ hoạt động miêu tả người dùng Thay Đổi Quy Định:.....	50
3.7. Sơ đồ tuần tự.....	51
3.7.1. Sơ đồ tuần tự đăng ký	51
3.7.2. Sơ đồ tuần tự đăng nhập	52
3.7.3. Sơ đồ tuần tự Mở sổ.....	53
3.7.4. Sơ đồ tuần tự Lập Phiếu Gửi Tiền	54
3.7.5. Sơ đồ tuần tự Lập Phiếu Rút Tiền	54
3.7.6. Sơ đồ tuần tự Tra Cứu Thông Tin.....	55
3.7.7. Sơ đồ tuần tự Lập Báo Cáo.....	56
3.7.8. Sơ đồ tuần tự thay đổi quy định.....	56
3.8.Class Diagram:	57
CHƯƠNG 4: THIẾT KẾ HỆ THỐNG.....	61
4.1. Kiến trúc hệ thống	61
4.1.1 Tổng quan kiến trúc	61

4.1.2 Lớp Giao Diện Người Dùng (Presentation Layer)	61
4.1.3 Lớp Xử Lý Nghiệp Vụ (Business Logic Layer)	62
4.1.4 Lớp Dữ Liệu (Data Layer)	62
4.1.5 Lợi ích của kiến trúc 3 lớp	63
4.2. Thiết kế giao diện:	64
4.2.1. Giao diện khi đăng kí	64
4.2.2. Giao diện khi đăng nhập	64
4.2.3. Giao diện chọn loại báo cáo	65
4.2.4: Giao diện báo cáo theo ngày	65
4.2.5 Giao diện báo cáo theo tháng	66
4.2.6: Giao diện phiếu rút tiền	66
4.2.7: Giao diện tìm kiếm phiếu rút tiền	67
4.2.8: Giao diện tạo phiếu rút tiền	68
4.2.9: Giao diện chỉnh sửa phiếu rút tiền	68
4.2.10: Giao diện chi tiết phiếu rút tiền	69
4.2.11: Giao diện xóa phiếu rút tiền	69
4.2.12 Giao diện Passbook	70
4.2.13. Giao diện chỉnh sửa Passbook	70
4.2.14. Giao diện chi tiết Passbook	71
4.2.15. Giao diện xóa Passbook	71
4.2.16 Giao diện chính loại tiết kiệm	72
4.2.17 Giao diện chỉnh sửa thông tin loại tiết kiệm	73
4.2.18 Giao diện thông tin loại tiết kiệm	73
4.2.19 Giao diện xóa loại tiết kiệm	74
4.2.20 Giao diện chỉnh sửa thông tin gửi thêm tiết kiệm	75
4.2.21: Giao diện xóa thông tin gửi thêm tiết kiệm	75
4.2.22: Giao diện gửi thêm tiết kiệm	75
4.2.23: Giao diện tìm kiếm phiếu thêm tiết kiệm	76
4.2.24: Giao diện tạo mới phiếu gửi thêm tiết kiệm	77
4.2.25: Giao diện thông tin cá nhân	77
4.2.26: Giao diện chỉnh sửa thông tin cá nhân	78
4.2.27: Giao diện chính chương trình	78
4.3 Test Case	79
4.3.1 Register - Người dùng đăng ký thành công	79
4.3.2. Register - Email đã tồn tại	79
4.3.3. Login - Đăng nhập thành công	80
4.3.4 Login - Đăng nhập với quyền quản trị	80
4.3.5 Login - Thông tin không hợp lệ	81

4.3.6. Login - Thiếu thông tin bắt buộc	81
KẾT LUẬN VÀ KIẾN NGHỊ	83
TÀI LIỆU THAM KHẢO	85

DANH MỤC HÌNH ẢNH

3.2.1.Sơ đồ phân rã chức năng BFD	16
3.4.1.Sơ đồ DFD mức 0	16
3.4.2.Sơ đồ DFD mức 1	17
3.4.2.1 Sơ đồ DFD mức 2: Mở sổ tiết kiệm.....	17
3.4.2.2 Sơ đồ DFD mức 2: Lập phiếu gửi tiền.....	18
3.4.2.3 Sơ đồ DFD mức 2: Lập phiếu rút tiền	18
3.4.2.4 Sơ đồ DFD mức 2: Tra cứu thông tin	18
3.4.2.5 Sơ đồ DFD mức 2: Lập báo cáo	19
3.4.2.6 Sơ đồ DFD mức 2: Thay đổi quy định.....	19
3.5.1.Sơ đồ UC tổng quát.....	20
3.5.2: Bảng Đặc tả Usecase đăng ký tài khoản.....	22
3.5.2.Sơ đồ UC đăng ký	22
3.5.3: Bảng Đặc tả Usecase đăng ký tài khoản.....	24
3.5.3.Sơ đồ UC đăng nhập	25
3.5.4:Bảng Đặc tả Usecase mở sổ tiết kiệm.....	27
3.5.4.Sơ đồ UC mở sổ tiết kiệm.....	27
3.5.5: Bảng Đặc tả Usecase lập phiếu gửi tiền	30
3.5.5.Sơ đồ UC lập phiếu gửi tiền.....	31
3.5.6: Bảng Đặc tả Usecase lập phiếu rút tiền	33
3.5.6.Sơ đồ UC lập phiếu rút tiền	34
3.5.7: Bảng Đặc tả Usecase Tra cứu thông tin.....	36
3.5.7.Sơ đồ UC tra cứu thông tin	37
3.5.8: Bảng Đặc tả Usecase Lập báo cáo	39
3.5.8.Sơ đồ UC lập báo cáo	40
3.5.9: Bảng Đặc tả Usecase Thay đổi quy định	42
3.5.9.Sơ đồ UC thay đổi quy định.....	43
3.6.1. Sơ đồ hoạt động miêu tả người dùng đăng nhập bằng tài khoản.....	44
3.6.2 Sơ đồ hoạt động miêu tả người dùng nghe nhạc offline	45
3.6.3. Sơ đồ hoạt động miêu tả người dùng mở sổ tiết kiệm	45
3.6.4. Sơ đồ hoạt động miêu tả người dùng Lập Phiếu Gửi Tiền	46
3.6.5 Sơ đồ hoạt động miêu tả người dùng Lập Phiếu Rút Tiền	47
3.6.6 Sơ đồ hoạt động miêu tả người dùng Tra Cứu Thông Tin.....	48
3.6.7 Sơ đồ hoạt động miêu tả người dùng Lập Báo Cáo.....	49
3.6.8 Sơ đồ hoạt động miêu tả người dùng Thay Đổi Quy Định.....	50
3.7.1. Sơ đồ tuần tự đăng ký	51
3.7.2. Sơ đồ tuần tự đăng nhập.....	52
3.7.3. Sơ đồ tuần tự Mở sổ.....	53

3.7.4. Sơ đồ tuần tự Lập Phiếu Gửi Tiền	54
3.7.5. Sơ đồ tuần tự Lập Phiếu Rút Tiền.....	55
3.7.6. Sơ đồ tuần tự Tra Cứu Thông Tin.....	55
3.7.7. Sơ đồ tuần tự Lập Báo Cáo	56
3.7.8. Sơ đồ tuần tự thay đổi quy định	57
3.8. Class Diagram	57
Bảng 3.8.1: Bảng Đặc tả lớp Passbook	58
Bảng 3.8.2: Bảng Đặc tả lớp SavingsAccountType	59
Bảng 3.8.3: Bảng Đặc tả lớp SavingsDeposit.....	59
Bảng 3.8.4: Bảng Đặc tả lớp users	60
Bảng 3.8.5: Bảng Đặc tả lớp WithdrawalSlip.....	61
4.2.1. Giao diện khi đăng kí	64
4.2.2. Giao diện khi đăng nhập	64
4.2.3. Giao diện chọn loại báo cáo	65
4.2.4. Giao diện báo cáo theo ngày.....	66
4.2.5. Giao diện báo cáo theo tháng.....	66
4.2.6. Giao diện phiếu rút tiền	67
4.2.7. Giao diện tìm kiếm phiếu rút tiền	67
4.2.8. Giao diện tạo phiếu rút tiền.....	68
4.2.9. Giao diện chỉnh sửa phiếu rút tiền	68
4.2.10: Giao diện chi tiết phiếu rút tiền	69
4.2.11: Giao diện chi tiết phiếu rút tiền	69
4.2.12: Giao diện Passbook.....	70
4.2.13: Giao diện chỉnh sửa Passbook	71
4.2.14: Giao diện chỉnh sửa Passbook	71
4.2.15: Giao diện chỉnh sửa Passbook	72
4.2.16: Giao diện chỉnh sửa Passbook	72
4.2.17: Giao diện chỉnh sửa thông tin loại tiết kiệm.....	73
4.2.18: Giao diện thông tin loại tiết kiệm	73
4.2.19: Giao diện xóa loại tiết kiệm	74
4.2.20: Giao diện chỉnh sửa thông tin gửi thêm tiết kiệm.....	75
4.2.21: Giao diện xóa thông tin gửi thêm tiết kiệm	75
4.2.22: Giao diện gửi thêm tiết kiệm	76
4.2.23: Giao diện tìm kiếm phiếu thêm tiết kiệm	77
4.2.24: Giao diện tạo mới phiếu gửi thêm tiết kiệm	77
4.2.25: Giao diện thông tin cá nhân	78
4.2.26: Giao diện chỉnh sửa thông tin loại tiết kiệm.....	78
4.2.27: Giao diện chính chương trình	78

DANH MỤC THUẬT NGỮ

STT	Thuật ngữ	Ý nghĩa
1	MVC	Model-View-Controller, mô hình kiến trúc phần mềm phân tách logic nghiệp vụ, giao diện và dữ liệu.
2	SQL	Structured Query Language, ngôn ngữ lập trình tiêu chuẩn để quản lý và thao tác cơ sở dữ liệu quan hệ.
3	SQL Server	Hệ quản trị cơ sở dữ liệu quan hệ do Microsoft phát triển, sử dụng SQL để xử lý và lưu trữ dữ liệu.
4	Use Case	Mô tả cách người dùng tương tác với hệ thống để thực hiện một chức năng cụ thể.
5	Activity Diagram (AD)	Sơ đồ biểu diễn luồng công việc hoặc hoạt động trong hệ thống, thường dùng trong UML.
6	Sequence Diagram	Sơ đồ trình tự thể hiện cách các đối tượng trong hệ thống tương tác với nhau qua các thông điệp.
7	BFD	Business Function Diagram, sơ đồ phân rã chức năng mô tả các chức năng chính của hệ thống.
8	DFD	Data Flow Diagram, sơ đồ luồng dữ liệu mô tả cách dữ liệu di chuyển qua các quy trình của hệ thống.
9	Bootstrap	Framework HTML, CSS, và JavaScript mã nguồn mở hỗ trợ thiết kế giao diện web đáp ứng (responsive design).

10	C#	Ngôn ngữ lập trình hướng đối tượng do Microsoft phát triển, được sử dụng phổ biến trong các ứng dụng .NET.
11	Visual Studio	Môi trường phát triển tích hợp (IDE) do Microsoft cung cấp, hỗ trợ phát triển nhiều loại ứng dụng phần mềm.
12	Entity Framework (EF)	Công cụ ORM (Object-Relational Mapping) của Microsoft, giúp tương tác với cơ sở dữ liệu dễ dàng qua mô hình đối tượng.
13	Primary Actor	Tác nhân chính tương tác trực tiếp với hệ thống trong một Use Case.
14	Secondary Actor	Tác nhân phụ, hỗ trợ hoặc cung cấp tài nguyên để thực hiện Use Case.
15	CRUD	Create, Read, Update, Delete - các thao tác cơ bản trong quản lý cơ sở dữ liệu.
16	Authentication	Quá trình xác minh danh tính của người dùng để cho phép truy cập vào hệ thống.
17	Authorization	Quá trình kiểm tra quyền hạn của người dùng để thực hiện các hành động cụ thể trong hệ thống.
18	TDE	Transparent Data Encryption, cơ chế mã hóa toàn bộ cơ sở dữ liệu để đảm bảo an toàn dữ liệu.
19	Mock	Công cụ hoặc đối tượng giả lập được sử dụng trong kiểm thử phần mềm để thay thế các đối tượng thật.
20	Unit Test	Phương pháp kiểm thử phần mềm tập trung vào các đơn vị nhỏ nhất như hàm hoặc lớp.

CHƯƠNG 1. MỞ ĐẦU

1.1 Tổng quan về đề tài.

Trong bối cảnh ngày càng phát triển của công nghệ thông tin, nhu cầu quản lý tài chính cá nhân và doanh nghiệp một cách hiệu quả, chính xác đang trở thành một xu hướng tăng trưởng mạnh mẽ. Trong đó, quản lý sổ tiết kiệm là một phần quan trọng trong việc quản trị tài chính, giúp người dùng theo dõi, đầu tư và đạt được các mục tiêu tài chính dài hạn.

Hiện nay, phần lớn các công cụ quản lý tài chính trên thị trường đã được đầu tích hợp với các nền tảng số, nhưng một số ứng dụng vẫn có hạn chế về giao diện, tính linh hoạt và bảo mật. Đề tài "Quản lý sổ tiết kiệm" trên website sử dụng MVC .NET nhằm tạo ra một hệ thống tích hợp cao, giao diện thân thiện, và đảm bảo tính đồng bộ trong xử lý và bảo mật dữ liệu người dùng.

Đề tài này đặc biệt hướng tới việc tính hợp lý trong việc tối ưu hóa giao diện người dùng, cải thiện hiệu suất xử lý của hệ thống và cung cấp các chức năng tiên tiến như tra cứu giao dịch, quản lý tài khoản, và đề xuất báo cáo.

1.2. Mục tiêu nghiên cứu.

Xây dựng một hệ thống quản lý sổ tiết kiệm trên nền tảng web sử dụng MVC .NET, với giao diện người dùng thân thiện và tính năng đầy đủ. Tích hợp các công nghệ hiện đại như SQL Server để lưu trữ và quản lý dữ liệu một cách an toàn và hiệu quả. Đáp ứng nhu cầu báo cáo tài chính và quản lý giao dịch, giúp người sử dụng tối ưu hoá các quy trình tài chính cá nhân hoặc doanh nghiệp. Cung cấp mô hình bài học đáng tin cậy cho các ứng dụng web trong lĩnh vực tài chính.

1.3 Đối tượng nghiên cứu

Đối tượng nghiên cứu của đề tài bao gồm hệ thống quản lý sổ tiết kiệm, các chức năng liên quan đến quản lý tài khoản, giao dịch, báo cáo tài chính và bảo mật dữ liệu. Các công nghệ cần nghiên cứu bao gồm: Mô hình MVC .NET, cơ sở dữ liệu SQL Server và các giao thức API cho hệ thống web. Hướng tới người dùng cá nhân và doanh nghiệp nhỏ, nhằm cung cấp giải pháp quản lý tài chính hiệu quả.

1.4. Phạm vi.

Phạm vi nghiên cứu gồm việc thiết kế và phát triển một website quản lý sổ tiết kiệm sử dụng ngôn ngữ C# và nền tảng .NET. Dữ liệu sẽ được quản lý trên SQL Server, bao gồm các bảng dữ liệu về người dùng, giao dịch, và báo cáo. Các tính năng bao gồm: quản lý tài khoản, theo dõi giao dịch, tính lãi suất, và sinh báo cáo tài chính. Hạn chế phạm vi: Đề tài tập trung vào mô hình giao diện và các chức năng cơ bản, không bao gồm tính năng phức tạp như machine learning hoặc blockchain.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Tổng quan về ngôn ngữ lập trình C#

2.1.1. Khái niệm

C# (C-Sharp) là một ngôn ngữ lập trình hiện đại, hướng đối tượng được Microsoft phát triển và giới thiệu lần đầu tiên vào năm 2000 như một phần của nền tảng .NET Framework. Ngôn ngữ này được thiết kế nhằm tối ưu hóa cho sự đơn giản, an toàn và khả năng mở rộng, giúp lập trình viên dễ dàng triển khai các ứng dụng từ nhỏ đến lớn, bao gồm cả ứng dụng web, desktop, di động và game.

C# kế thừa nhiều đặc điểm mạnh mẽ từ các ngôn ngữ như C++, Java và Pascal, đồng thời tích hợp các tính năng tiên tiến như quản lý bộ nhớ tự động (garbage collection), kiểu dữ liệu mạnh (strong typing) và khả năng xử lý dữ liệu linh hoạt thông qua LINQ (Language Integrated Query). Những tính năng này khiến C# trở thành một lựa chọn phổ biến trong phát triển phần mềm hiện đại.

2.1.2. Ưu điểm

C# nổi bật với sự đơn giản và dễ tiếp cận nhờ cú pháp rõ ràng và thân thiện, phù hợp cho cả người mới bắt đầu lẫn lập trình viên có kinh nghiệm. Đặc biệt, C# được sử dụng rộng rãi trong việc phát triển ứng dụng web nhờ sự tích hợp chặt chẽ với ASP.NET và các công nghệ khác của Microsoft.

Bên cạnh đó, tính linh hoạt cao của C# cho phép xây dựng nhiều loại ứng dụng khác nhau, từ hệ thống quản lý doanh nghiệp, ứng dụng di động đến các trò chơi phức tạp trên Unity. Một ưu điểm khác là khả năng bảo mật vượt

trội, nhờ vào việc kiểm tra kiểu dữ liệu chặt chẽ, giảm thiểu rủi ro lỗi runtime và bảo vệ hệ thống khỏi các lỗ hổng phổ biến.

Hiệu suất của C# cũng rất ấn tượng, nhờ vào sự tối ưu trong cách thức xử lý của nền tảng .NET. Cộng đồng lập trình viên lớn cùng với sự hỗ trợ mạnh mẽ từ Microsoft đã giúp C# trở thành một ngôn ngữ đáng tin cậy cho các dự án dài hạn.

2.1.3. Nhược điểm

Mặc dù C# có nhiều ưu điểm, nhưng nó vẫn tồn tại một số nhược điểm đáng lưu ý. Trước hết, ngôn ngữ này phụ thuộc lớn vào hệ sinh thái Microsoft, điều này có thể là một hạn chế đối với các dự án yêu cầu tính độc lập nền tảng hoặc không muốn phụ thuộc vào công nghệ của một nhà cung cấp cụ thể.

Ngoài ra, mặc dù hiệu suất của C# rất tốt, nhưng nó không thể sánh bằng các ngôn ngữ lập trình cấp thấp như C++ hoặc Rust trong những trường hợp yêu cầu hiệu suất tối đa. Hơn nữa, các ứng dụng được viết bằng C# thường tiêu tốn nhiều tài nguyên hơn, điều này có thể là một trở ngại đối với các thiết bị phần cứng hạn chế.

Tuy vậy, với sự phát triển không ngừng của công nghệ và các cải tiến trong hệ sinh thái .NET, C# vẫn là một lựa chọn lý tưởng cho đa số các dự án phát triển phần mềm hiện đại.

2.2. Tổng quan về SQL

2.2.1. Khái niệm

SQL (Structured Query Language) là ngôn ngữ lập trình chuẩn được sử dụng để quản lý và thao tác cơ sở dữ liệu quan hệ (RDBMS). Ngôn ngữ này

được thiết kế để truy vấn, chèn, cập nhật và xóa dữ liệu trong cơ sở dữ liệu một cách hiệu quả và dễ dàng. SQL không chỉ cho phép người dùng xử lý dữ liệu mà còn hỗ trợ việc định nghĩa cấu trúc cơ sở dữ liệu thông qua các câu lệnh như CREATE, ALTER và DROP.

SQL đã trở thành một tiêu chuẩn công nghiệp được hầu hết các hệ quản trị cơ sở dữ liệu như MySQL, SQL Server, Oracle và PostgreSQL áp dụng. Với sự phổ biến của dữ liệu lớn và phân tích dữ liệu, SQL ngày càng đóng vai trò quan trọng trong việc quản lý và khai thác thông tin từ các hệ thống dữ liệu phức tạp.

2.2.2. Ưu điểm

SQL nổi bật với khả năng truy vấn dữ liệu một cách nhanh chóng và chính xác nhờ vào cú pháp đơn giản và dễ học. Các lệnh trong SQL thường sử dụng ngôn ngữ tự nhiên, giúp người dùng dễ dàng hiểu và viết các câu truy vấn, ngay cả khi không có nền tảng lập trình chuyên sâu. Một trong những ưu điểm quan trọng nhất của SQL là tính chuẩn hóa, đảm bảo rằng các hệ thống khác nhau có thể sử dụng chung một ngôn ngữ để giao tiếp và xử lý dữ liệu.

Bên cạnh đó, SQL rất linh hoạt và mạnh mẽ khi xử lý các tập dữ liệu lớn, nhờ vào các công cụ tối ưu hóa truy vấn và các phương pháp lập chỉ mục (indexing) hiện đại. Ngôn ngữ này cũng hỗ trợ đầy đủ các phép toán dữ liệu phức tạp, từ việc thực hiện các phép nối (JOIN) đến phân tích dữ liệu đa chiều thông qua các hàm tổng hợp.

2.2.3. Nhược điểm

Tuy có nhiều ưu điểm, SQL vẫn tồn tại một số hạn chế. Một trong những nhược điểm lớn nhất của SQL là phụ thuộc vào các hệ quản trị cơ sở dữ

liệu cụ thể. Mặc dù cú pháp SQL cơ bản là chuẩn hóa, nhưng nhiều hệ thống như SQL Server, Oracle hay MySQL lại có những phần mở rộng riêng, gây khó khăn cho việc di chuyển ứng dụng giữa các nền tảng.

Ngoài ra, SQL có thể trở nên không hiệu quả khi xử lý các cơ sở dữ liệu phi quan hệ hoặc dữ liệu không cấu trúc, nơi mà các công nghệ như NoSQL hoặc Big Data thường hoạt động tốt hơn. Cuối cùng, việc tối ưu hóa truy vấn SQL đòi hỏi kiến thức chuyên sâu và có thể trở thành một thách thức đối với các lập trình viên mới làm quen.

2.3. Tổng quan về SQL Server

2.3.1. Khái niệm

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System - RDBMS) do Microsoft phát triển. SQL Server được thiết kế để lưu trữ và quản lý khối lượng lớn dữ liệu, cung cấp các công cụ mạnh mẽ để xử lý, phân tích, và truy xuất dữ liệu. Hệ thống hỗ trợ ngôn ngữ SQL (Structured Query Language) tiêu chuẩn, đồng thời bổ sung các tính năng mở rộng để tương thích tốt hơn với các ứng dụng doanh nghiệp. SQL Server có khả năng làm việc với nhiều loại dữ liệu khác nhau, từ văn bản đơn giản, hình ảnh, đến dữ liệu không gian địa lý và dữ liệu phi cấu trúc.

SQL Server thường được sử dụng trong các ứng dụng doanh nghiệp nhờ vào khả năng tích hợp chặt chẽ với các công nghệ khác của Microsoft, như .NET Framework, Power BI, và Azure. Phiên bản SQL Server Express còn được cung cấp miễn phí, thích hợp cho các ứng dụng nhỏ và học tập.

2.3.2. Ưu điểm

SQL Server mang lại nhiều ưu điểm vượt trội, khiến nó trở thành một trong những hệ quản trị cơ sở dữ liệu phổ biến và đáng tin cậy nhất hiện nay. Trước hết, SQL Server có khả năng mở rộng và hiệu năng cao, cho phép xử lý hiệu quả cả trong các ứng dụng nhỏ lẫn các hệ thống lớn với khối lượng dữ liệu khổng lồ. Với các tính năng tối ưu hóa truy vấn tự động, chỉ số phân cụm (clustered index), và khả năng phân vùng bảng, SQL Server hỗ trợ việc truy xuất và xử lý dữ liệu nhanh chóng, đáp ứng tốt nhu cầu của các tổ chức và doanh nghiệp lớn.

Bên cạnh đó, SQL Server nổi bật với tính bảo mật cao. Hệ thống tích hợp nhiều cơ chế bảo mật tiên tiến như xác thực dựa trên Windows, mã hóa dữ liệu toàn diện thông qua Transparent Data Encryption (TDE), và quản lý quyền truy cập chi tiết đến từng cấp độ dữ liệu (row-level security). Điều này đảm bảo rằng dữ liệu không chỉ được bảo vệ trước các cuộc tấn công từ bên ngoài, mà còn ngăn ngừa rủi ro do các lỗi truy cập không mong muốn từ nội bộ. Các công cụ này giúp SQL Server đáp ứng được các yêu cầu bảo mật khắt khe trong các lĩnh vực như tài chính, y tế, và chính phủ.

Một trong những lợi thế lớn khác của SQL Server là khả năng tích hợp dễ dàng với các công cụ và dịch vụ khác trong hệ sinh thái Microsoft, như Visual Studio, Power BI, và Azure. Điều này không chỉ giúp các lập trình viên nhanh chóng xây dựng, thử nghiệm, và triển khai các ứng dụng mà còn mang lại giải pháp phân tích và trực quan hóa dữ liệu mạnh mẽ. Ngoài ra, SQL Server cung cấp các tính năng sao lưu và phục hồi dữ liệu toàn diện. Điều này cho phép các tổ chức dễ dàng bảo vệ dữ liệu khỏi mất mát hoặc hư hỏng và nhanh chóng khôi phục hệ thống trong các tình huống khẩn cấp.

Giao diện quản lý thân thiện của SQL Server Management Studio (SSMS) cũng là một điểm cộng lớn. SSMS cung cấp môi trường trực quan cho quản trị viên cơ sở dữ liệu, giúp thực hiện các thao tác quản lý, tối ưu hóa, và giám sát hệ thống một cách dễ dàng. Với sự hỗ trợ từ tài liệu chi tiết và cộng đồng người dùng rộng lớn, SQL Server trở thành một lựa chọn lý tưởng không chỉ cho các chuyên gia mà còn cho các lập trình viên mới bắt đầu làm việc với cơ sở dữ liệu. Những ưu điểm này giúp SQL Server không chỉ đóng vai trò như một công cụ quản trị dữ liệu mà còn trở thành nền tảng quan trọng trong việc xây dựng và vận hành các hệ thống doanh nghiệp hiện đại.

2.3.3. Nhược điểm

Mặc dù SQL Server mang lại nhiều lợi ích, nhưng nó cũng tồn tại một số hạn chế. Chi phí bản quyền cao là một trong những nhược điểm chính, khiến SQL Server trở thành lựa chọn không khả thi cho các doanh nghiệp nhỏ hoặc tổ chức phi lợi nhuận có ngân sách hạn chế. Ngoài ra, mặc dù SQL Server hoạt động tốt trên các môi trường Windows, nhưng sự hỗ trợ cho các hệ điều hành khác như Linux vẫn còn hạn chế, gây khó khăn cho các tổ chức sử dụng đa nền tảng.

Hơn nữa, SQL Server có thể trở nên phức tạp đối với những người mới bắt đầu, đặc biệt khi cần triển khai các tính năng nâng cao như tích hợp dữ liệu hoặc phân tích dữ liệu lớn. Việc tối ưu hóa hiệu suất và quản lý hệ thống cũng đòi hỏi kỹ năng và kinh nghiệm, điều này có thể làm tăng chi phí nhân sự và thời gian triển khai.

2.4. Tổng quan về Microsoft Visual Studio

2.4.1. Khái niệm

Microsoft Visual Studio là một môi trường phát triển tích hợp (IDE) mạnh mẽ, được thiết kế và phát triển bởi Microsoft. Đây là công cụ hàng đầu dành cho lập trình viên trong việc xây dựng các ứng dụng phần mềm trên nhiều nền tảng, bao gồm web, desktop, di động và điện toán đám mây. Visual Studio hỗ trợ nhiều ngôn ngữ lập trình như C#, Visual Basic, C++, Python và JavaScript, đồng thời tích hợp chặt chẽ với hệ sinh thái công cụ và dịch vụ của Microsoft như Azure, .NET, và SQL Server.

Visual Studio không chỉ cung cấp các tính năng như trình biên tập mã nguồn, trình gỡ lỗi mạnh mẽ, và công cụ kiểm tra mã, mà còn hỗ trợ các dịch vụ tiên tiến như kiểm soát phiên bản (Version Control) và triển khai liên tục (Continuous Deployment). Với giao diện trực quan và các tính năng tối ưu hóa, Visual Studio giúp lập trình viên tiết kiệm thời gian và nâng cao hiệu suất phát triển.

2.4.2. Ưu điểm

Microsoft Visual Studio nổi bật với giao diện thân thiện và khả năng hỗ trợ đa nền tảng, cho phép lập trình viên dễ dàng chuyển đổi giữa các dự án với nhiều ngôn ngữ và công nghệ khác nhau. IDE này được trang bị đầy đủ các công cụ từ cơ bản đến nâng cao, đáp ứng tốt các yêu cầu của cả lập trình viên mới bắt đầu và chuyên gia.

Một ưu điểm lớn khác là khả năng tích hợp sâu với các công cụ và dịch vụ khác của Microsoft, như Azure DevOps và GitHub, giúp dễ dàng quản lý mã nguồn và triển khai ứng dụng. Visual Studio cũng đi kèm với trình gỡ lỗi

manh mẽ, giúp lập trình viên phát hiện và khắc phục lỗi một cách nhanh chóng. Đặc biệt, Visual Studio có một cộng đồng lớn, cung cấp nhiều tài nguyên học tập, plugin và tiện ích mở rộng, giúp lập trình viên mở rộng khả năng của IDE theo nhu cầu riêng.

2.4.3. Nhược điểm

Mặc dù Visual Studio là một công cụ mạnh mẽ, nhưng chi phí bản quyền cao có thể là một rào cản đối với các lập trình viên tự do hoặc doanh nghiệp nhỏ. Ngoài ra, Visual Studio có dung lượng lớn và yêu cầu tài nguyên hệ thống cao, điều này có thể gây khó khăn trên các máy tính cấu hình thấp. Một nhược điểm khác là sự phức tạp trong giao diện và các tính năng nâng cao, khiến người mới bắt đầu có thể gặp khó khăn trong việc làm quen và sử dụng hiệu quả.

Bên cạnh đó, mặc dù Microsoft đã cải thiện tính đa nền tảng của Visual Studio, nhưng sự hỗ trợ này vẫn chưa hoàn toàn tương đồng với các IDE khác, đặc biệt trên các hệ điều hành không phải Windows. Điều này khiến Visual Studio trở thành một lựa chọn ít phù hợp hơn trong các môi trường phát triển phi Windows hoặc yêu cầu tính linh hoạt cao.

2.5. Tổng quan về Bootstrap

2.5.1 Khái niệm

Bootstrap là một framework HTML, CSS và JavaScript mã nguồn mở được thiết kế để hỗ trợ phát triển giao diện web nhanh chóng và dễ dàng. Được giới thiệu lần đầu bởi Twitter, Bootstrap cung cấp các thành phần giao diện sẵn có như nút bấm, thanh điều hướng, bảng, biểu mẫu, và các công cụ thiết kế web đáp ứng (responsive design) nhằm tối ưu hóa trải nghiệm người dùng trên nhiều loại thiết bị khác nhau. Bootstrap hoạt động dựa trên một hệ thống lưới

(grid system) linh hoạt, cho phép lập trình viên xây dựng giao diện web có cấu trúc rõ ràng và nhất quán.

Framework này thường được sử dụng trong các dự án web hiện đại nhờ vào khả năng tích hợp nhanh, hỗ trợ đa nền tảng, và tài liệu chi tiết. Hiện nay, Bootstrap đã trở thành một công cụ tiêu chuẩn trong việc thiết kế giao diện web, đặc biệt là khi cần triển khai các ứng dụng web đáp ứng hoặc giao diện người dùng phức tạp.

2.5.2 Ưu điểm

Bootstrap mang lại nhiều lợi ích cho lập trình viên và các dự án web. Một trong những ưu điểm lớn nhất của Bootstrap là khả năng hỗ trợ thiết kế giao diện đáp ứng (responsive design). Framework này tích hợp hệ thống lưới linh hoạt, giúp giao diện tự động điều chỉnh kích thước và bố cục để phù hợp với các màn hình khác nhau, từ máy tính để bàn, máy tính bảng đến điện thoại di động. Điều này giúp tiết kiệm thời gian và công sức trong việc tạo các phiên bản giao diện khác nhau cho từng thiết bị.

Thêm vào đó, Bootstrap cung cấp nhiều thành phần giao diện sẵn có như nút bấm, biểu mẫu, thanh điều hướng, và bảng, giúp lập trình viên không cần phải thiết kế từ đầu. Tất cả các thành phần đều được tối ưu hóa về mặt thẩm mỹ và hiệu suất, phù hợp với các xu hướng thiết kế hiện đại. Ngoài ra, Bootstrap có cộng đồng người dùng lớn và tài liệu phong phú, cho phép người dùng dễ dàng tra cứu và tìm giải pháp khi gặp vấn đề.

Framework này còn hỗ trợ khả năng tùy chỉnh cao, cho phép lập trình viên thay đổi các biến số (variables) hoặc lớp CSS theo yêu cầu dự án mà không làm mất đi tính nhất quán của hệ thống. Khả năng tích hợp nhanh với

các công cụ và thư viện JavaScript cũng là một lợi thế, giúp tối ưu hóa quy trình phát triển web.

2.5.3 Nhược điểm

Mặc dù có nhiều ưu điểm, Bootstrap cũng tồn tại một số nhược điểm cần lưu ý. Một trong những hạn chế lớn nhất là giao diện mặc định dễ bị nhận diện. Vì nhiều lập trình viên sử dụng các thành phần mặc định mà không tùy chỉnh, giao diện web dễ bị giống nhau và thiếu đi tính độc đáo. Điều này đòi hỏi lập trình viên phải đầu tư thời gian để chỉnh sửa hoặc bổ sung các phong cách riêng nhằm tạo sự khác biệt cho dự án.

Thêm vào đó, việc sử dụng Bootstrap có thể dẫn đến tăng kích thước tệp CSS và JavaScript, đặc biệt là khi không tối ưu hóa hoặc loại bỏ các thành phần không cần thiết. Điều này có thể làm giảm hiệu suất tải trang, đặc biệt đối với các ứng dụng web lớn hoặc yêu cầu tốc độ cao.

Ngoài ra, hệ thống lưới của Bootstrap có thể gây hạn chế trong việc tạo các thiết kế phức tạp hoặc không tuân theo bố cục truyền thống. Với những dự án yêu cầu giao diện sáng tạo, lập trình viên thường phải can thiệp sâu vào mã nguồn hoặc sử dụng các framework khác để đạt được kết quả mong muốn. Cuối cùng, Bootstrap có thể không phù hợp với những lập trình viên mới làm quen, bởi số lượng lớn các lớp và thành phần có thể gây nhầm lẫn và khó kiểm soát.

Bootstrap vẫn là một lựa chọn mạnh mẽ và phổ biến trong thiết kế giao diện web, đặc biệt đối với các dự án cần triển khai nhanh và đảm bảo tính nhất quán trên nhiều loại thiết bị. Tuy nhiên, để tận dụng tối đa lợi ích của Bootstrap, lập trình viên cần hiểu rõ các tính năng, biết cách tối ưu hóa và tùy chỉnh giao diện theo yêu cầu cụ thể của dự án.

CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

3.1. Mô tả bài toán:

Quản lý sổ tiết kiệm là một nhu cầu phổ biến đối với các cá nhân và tổ chức nhằm theo dõi, quản lý tài chính một cách hiệu quả. Tuy nhiên, việc quản lý thông qua các phương pháp thủ công hoặc hệ thống cũ kỹ không còn đáp ứng được yêu cầu hiện đại, đặc biệt là trong bối cảnh công nghệ số phát triển mạnh mẽ. Do đó, việc xây dựng một hệ thống quản lý sổ tiết kiệm trên nền tảng web, ứng dụng mô hình MVC .NET, không chỉ đáp ứng được nhu cầu quản lý mà còn mang lại hiệu quả về thời gian, độ chính xác và tính bảo mật.

Bài toán đặt ra là thiết kế và triển khai một hệ thống quản lý sổ tiết kiệm, cho phép người dùng tạo và quản lý tài khoản, theo dõi giao dịch, tính lãi suất và tạo các báo cáo tài chính chi tiết. Hệ thống cần đảm bảo tính năng dễ sử dụng, giao diện thân thiện và khả năng mở rộng để đáp ứng các yêu cầu trong tương lai. Đồng thời, bài toán yêu cầu hệ thống phải tích hợp công nghệ hiện đại như SQL Server để lưu trữ và quản lý dữ liệu một cách an toàn, và Microsoft Visual Studio để phát triển và kiểm thử hệ thống.

3.2. Phân tích yêu cầu

3.2.1. Yêu cầu chức năng

Mở sổ tiết kiệm: Hệ thống cho phép khách hàng mở sổ tiết kiệm mới thông qua biểu mẫu BM1. Trong quá trình mở sổ, khách hàng cần cung cấp các thông tin như mã số sổ, loại tiết kiệm (không kỳ hạn, kỳ hạn 3 tháng hoặc kỳ hạn 6 tháng), thông tin cá nhân (tên, CMND, địa chỉ), ngày mở sổ và số tiền gửi ban đầu. Quy định liên quan đến việc mở sổ bao gồm số tiền gửi tối thiểu là 1.000.000đ. Điều này giúp đảm bảo tính đồng nhất trong quy trình và khả năng quản lý hiệu quả.

Lập phiếu gửi tiền: Hệ thống hỗ trợ khách hàng gửi thêm tiền vào sổ tiết kiệm thông qua biểu mẫu BM2. Tính năng này yêu cầu khách hàng cung cấp mã số sổ, thông tin cá nhân, ngày gửi và số tiền gửi. Theo quy định QĐ2, hệ thống chỉ cho phép gửi thêm tiền tại các thời điểm kỳ hạn của từng loại tiết kiệm và số tiền gửi thêm tối thiểu là 100.000đ. Quy định này đảm bảo các giao dịch tuân thủ các điều kiện của sản phẩm tiết kiệm và giảm thiểu lỗi.

Lập phiếu rút tiền: Hệ thống cung cấp biểu mẫu BM3 để khách hàng thực hiện rút tiền. Biểu mẫu này yêu cầu thông tin mã số sổ, tên khách hàng, ngày rút, và số tiền rút. Các quy định rút tiền bao gồm: loại tiết kiệm không kỳ hạn chỉ được phép rút sau 15 ngày kể từ ngày gửi, trong khi loại có kỳ hạn chỉ cho phép rút toàn bộ số dư sau khi kỳ hạn kết thúc. Nếu rút trước kỳ hạn, lãi suất sẽ được tính theo mức không kỳ hạn. Tính năng này đảm bảo tuân thủ các quy định tài chính và giảm thiểu tranh chấp.

Tra cứu sổ tiết kiệm: Hệ thống cung cấp chức năng tra cứu thông tin sổ tiết kiệm thông qua biểu mẫu BM4. Người dùng có thể nhập các tiêu chí như mã số sổ hoặc tên khách hàng để tìm kiếm thông tin chi tiết. Kết quả trả về bao gồm mã số sổ, loại tiết kiệm, tên khách hàng và số dư hiện tại. Tính năng này giúp khách hàng dễ dàng theo dõi tình hình tài khoản của mình và hỗ trợ quản lý dữ liệu một cách hiệu quả.

Lập báo cáo tháng: Hệ thống hỗ trợ quản trị viên lập báo cáo hoạt động và báo cáo tài chính theo tháng thông qua các biểu mẫu BM5. Cụ thể:

- BM5.1: Báo cáo doanh số hoạt động theo ngày, thống kê tổng thu, tổng chi và chênh lệch giữa các loại tiết kiệm.
- BM5.2: Báo cáo mở/đóng sổ theo tháng, cung cấp số liệu về số sổ mở mới, số sổ đóng và chênh lệch. Các báo cáo này cung cấp

cái nhìn tổng quan về hiệu suất hoạt động của hệ thống và hỗ trợ ra quyết định.

Thay đổi quy định: Hệ thống cho phép quản trị viên cập nhật các quy định liên quan đến sản phẩm tiết kiệm như số lượng loại tiết kiệm, số tiền gửi tối thiểu, thời gian gửi tối thiểu hoặc mức lãi suất. Tính năng này giúp hệ thống duy trì tính linh hoạt và đáp ứng các yêu cầu kinh doanh mới.

3.2.2. Yêu cầu phi chức năng

Hiệu suất: Hệ thống phải xử lý nhanh chóng các giao dịch quan trọng như mở sổ, gửi tiền và rút tiền. Để đạt được điều này, các truy vấn cơ sở dữ liệu cần được tối ưu hóa nhằm giảm thiểu thời gian phản hồi.

Bảo mật: Hệ thống cần triển khai cơ chế xác thực người dùng và phân quyền rõ ràng. Các thông tin nhạy cảm như thông tin cá nhân và giao dịch phải được mã hóa để đảm bảo an toàn.

Khả năng mở rộng: Hệ thống phải dễ dàng tích hợp thêm các chức năng mới, xử lý lượng lớn giao dịch và hỗ trợ nhiều người dùng đồng thời trong tương lai.

Tính dễ sử dụng: Giao diện cần thân thiện, trực quan và hỗ trợ trên nhiều thiết bị, cho phép người dùng dễ dàng thao tác mà không cần kỹ năng chuyên sâu.

Đảm bảo tính toàn vẹn dữ liệu: Mọi giao dịch và thao tác trên hệ thống phải được ghi nhận chính xác, đảm bảo dữ liệu không bị mất mát hoặc sai lệch.

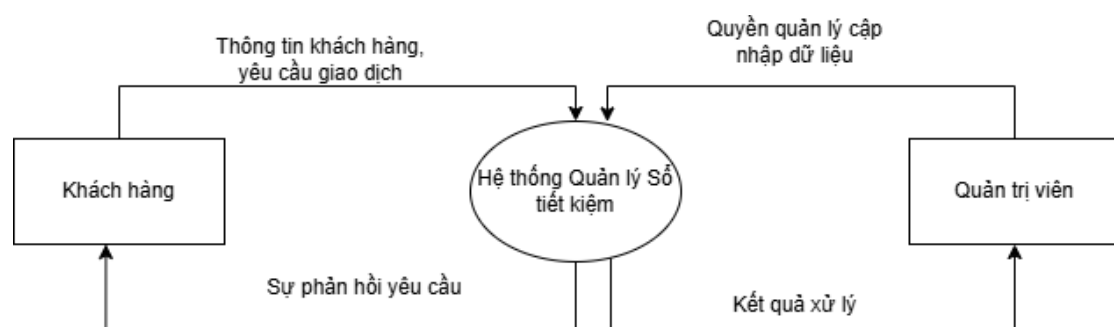
3.3. Sơ đồ phân rã chức năng (BFD)



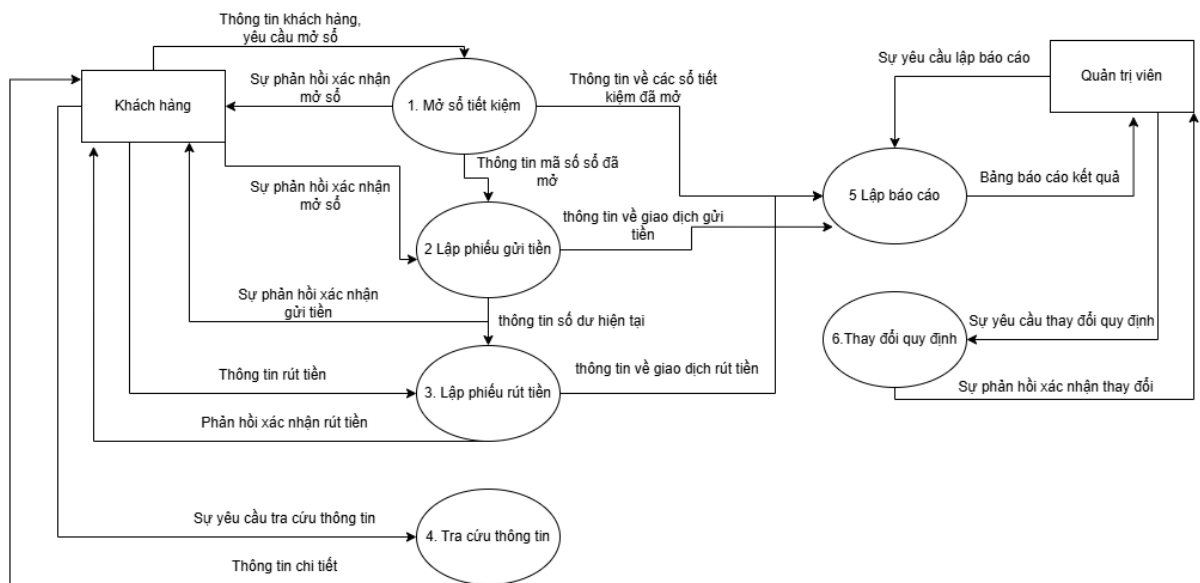
3.2.1. Sơ đồ phân rã chức năng BFD

3.4. Sơ đồ luồng dữ liệu DFD

3.4.1. Sơ đồ DFD mức khung cảnh (mức 0)



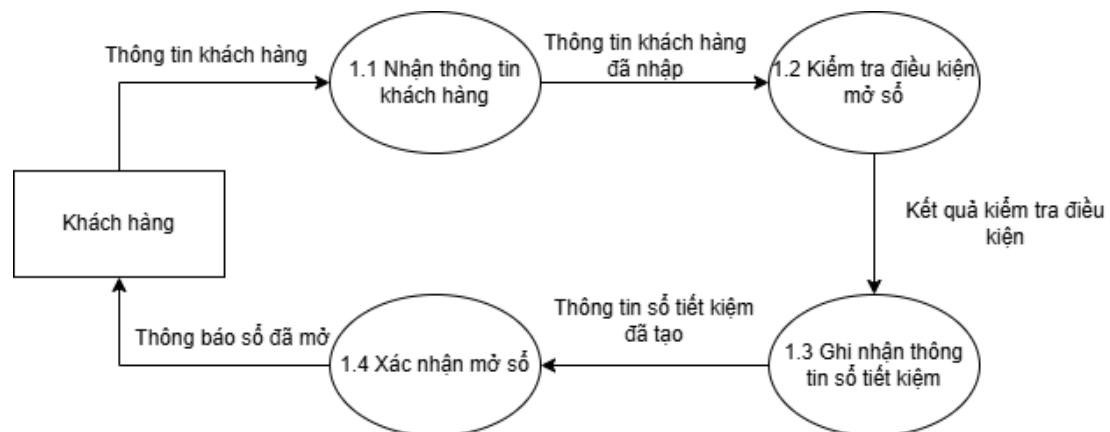
3.4.1. Sơ đồ DFD mức 0



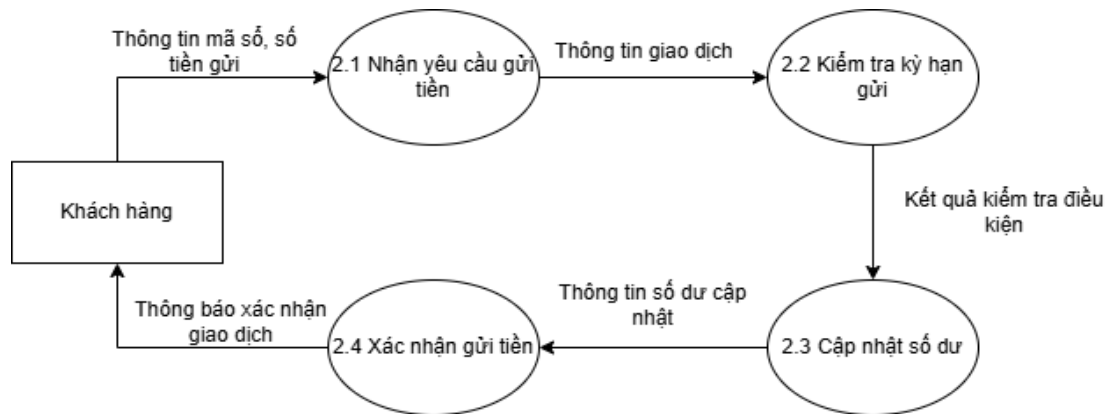
3.4.2. Sơ đồ DFD mức đỉnh (mức 1)

3.4.2.Sơ đồ DFD mức 1

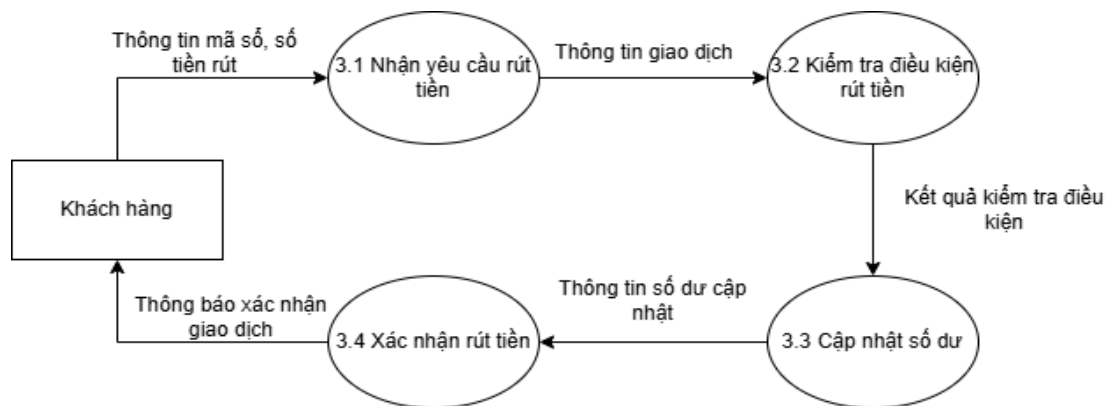
3.4.2. Sơ đồ DFD mức dưới đỉnh (mức 2)



3.4.2.1 Sơ đồ DFD mức 2: Mở sổ tiết kiệm



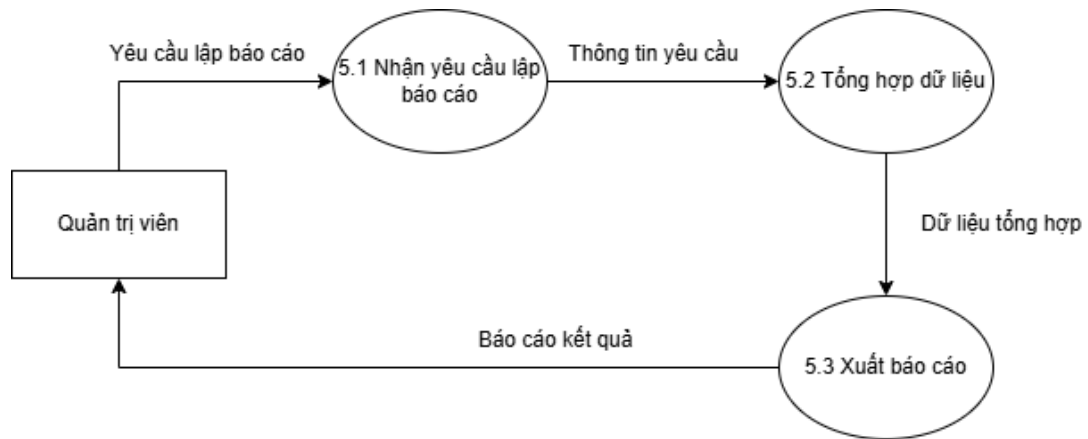
3.4.2.2 Sơ đồ DFD mức 2: Lập phiếu gửi tiền



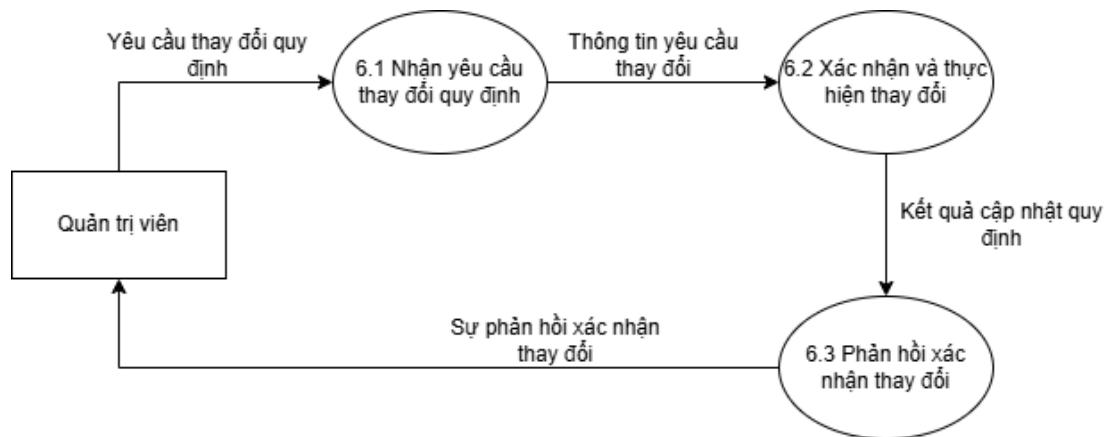
3.4.2.3 Sơ đồ DFD mức 2: Lập phiếu rút tiền



3.4.2.4 Sơ đồ DFD mức 2: Tra cứu thông tin



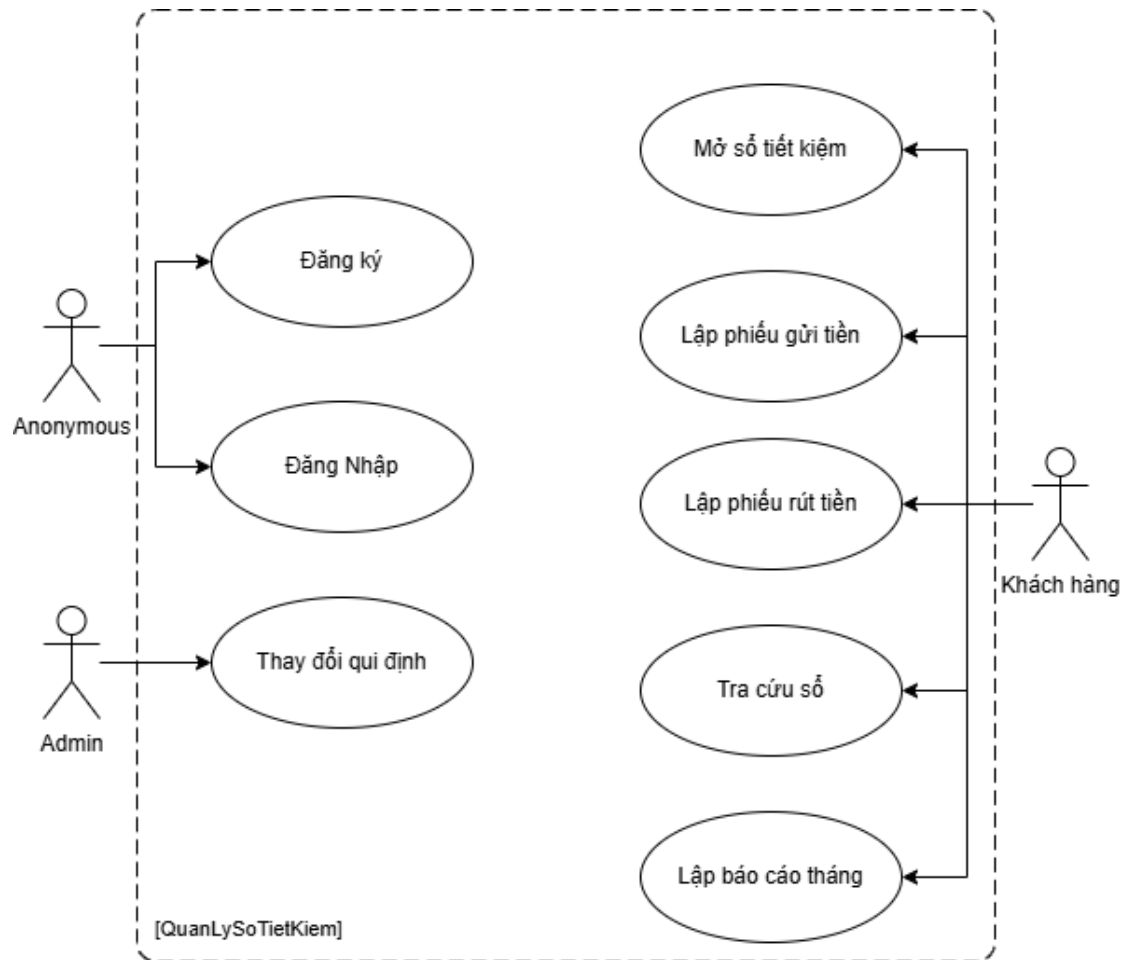
3.4.2.5 Sơ đồ DFD mức 2: Lập báo cáo



3.4.2.6 Sơ đồ DFD mức 2: Thay đổi quy định

3.5. Sơ đồ Usecase

3.5.1. Sơ đồ Usecase tổng quát



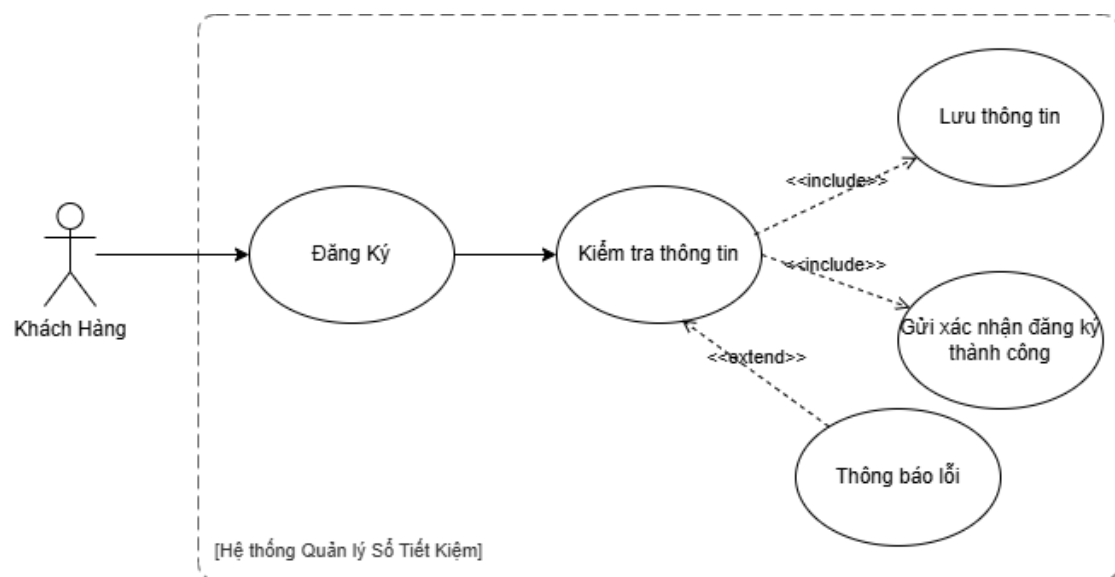
3.5.1.Sơ đồ UC tổng quát

3.5.2 Sơ đồ Usecase đăng ký

Tên Use Case	Đăng Ký Tài Khoản
Tên Tác Nhân	- Người dùng mới (Primary Actor): Người muốn tạo tài khoản.
Điều kiện tiên quyết	- Người dùng chưa có tài khoản trên hệ thống - Người dùng có thông tin đăng ký hợp lệ (email, tên người dùng, mật khẩu).
Đảm bảo tối thiểu	- Nếu thông tin không hợp lệ, hệ thống hiển thị thông báo lỗi tương ứng và không tạo tài khoản.
Đảm bảo thành công	- Tài khoản mới được lưu vào cơ sở dữ liệu. - Người dùng nhận được thông báo đăng ký thành công và được chuyển đến giao diện đăng nhập
Kích hoạt	Người dùng nhấn nút "Đăng Ký" sau khi nhập thông tin
Chuỗi Sự Kiện Chính	<p>- Người dùng mở giao diện đăng ký trên hệ thống.</p> <p>-Người dùng nhập các thông tin đăng ký bao gồm:</p> <ul style="list-style-type: none"> + Tên người dùng. + Email. + Mật khẩu. <p>-Người dùng nhấn nút "Đăng Ký" để gửi yêu cầu đăng ký.</p> <p>Hệ thống kiểm tra thông tin đăng ký:</p> <ul style="list-style-type: none"> + Tên người dùng: Không trùng lặp. + Email: Hợp lệ theo định dạng. <p>-Nếu thông tin hợp lệ:</p>

	<ul style="list-style-type: none"> + Hệ thống lưu thông tin tài khoản vào cơ sở dữ liệu. + Gửi thông báo đăng ký thành công cho người dùng. + Chuyển người dùng đến giao diện đăng nhập.
--	---

3.5.2: Bảng Đặc tả Usecase đăng ký tài khoản



3.5.2.Sơ đồ UC đăng ký

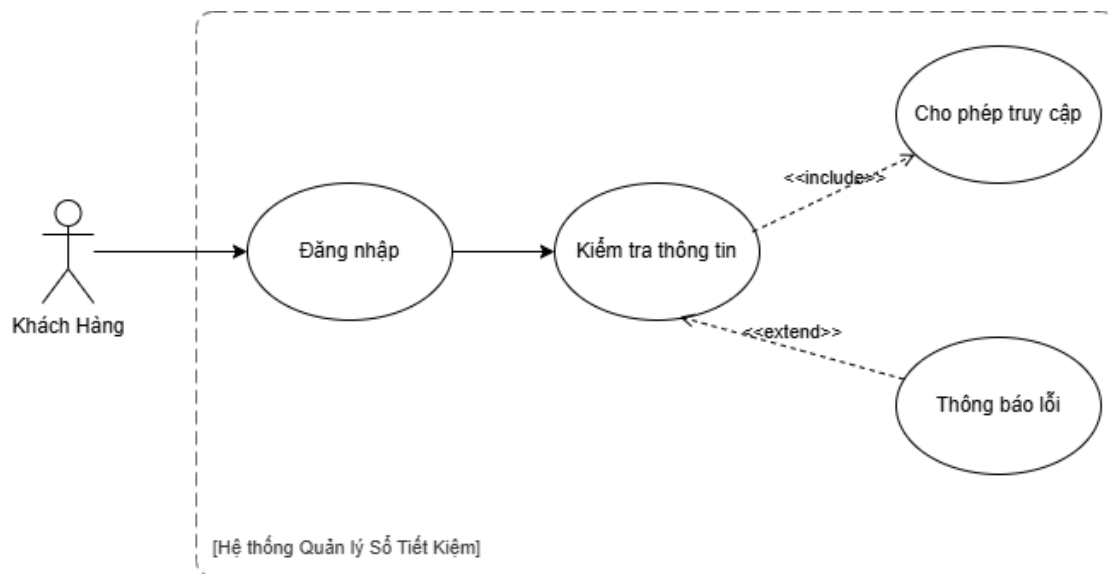
3.5.3 Sơ đồ Usecase đăng nhập

Thành phần	Mô tả
Tên Use Case	Đăng Nhập

Mã Use Case	UC002
Tác nhân	- Người dùng (Primary Actor): Người sử dụng muốn đăng nhập vào hệ thống.
Mục tiêu	Cho phép người dùng truy cập vào hệ thống bằng tài khoản đã được đăng ký.
Điều kiện tiên quyết	- Người dùng đã có tài khoản trong hệ thống. - Hệ thống hoạt động bình thường và kết nối cơ sở dữ liệu ổn định.
Đảm bảo tối thiểu	- Nếu thông tin đăng nhập không hợp lệ, hệ thống hiển thị thông báo lỗi và không cho phép truy cập.
Đảm bảo thành công	- Người dùng được xác thực thành công và truy cập vào giao diện chính của hệ thống.
Kích hoạt	Người dùng nhấn nút "Đăng Nhập" sau khi nhập tên tài khoản và mật khẩu.
Chuỗi sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng mở giao diện đăng nhập. 2. Người dùng nhập tên tài khoản và mật khẩu. 3. Người dùng nhấn nút "Đăng Nhập" để gửi yêu cầu. 4. Hệ thống kiểm tra thông tin đăng nhập: <ul style="list-style-type: none"> - Xác minh tên tài khoản có tồn tại không. - Kiểm tra mật khẩu có chính xác không. 5. Nếu thông tin hợp lệ: <ul style="list-style-type: none"> - Hệ thống cho phép người dùng truy cập vào hệ thống.

	<ul style="list-style-type: none"> - Chuyển hướng người dùng đến giao diện chính.
Ngoại lệ	<ol style="list-style-type: none"> 1. Tên tài khoản không tồn tại: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Tên tài khoản không tồn tại." 2. Mật khẩu không chính xác: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Mật khẩu không đúng." 3. Lỗi kết nối cơ sở dữ liệu: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Hệ thống gặp sự cố. Vui lòng thử lại sau." 4. Tài khoản bị khóa: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Tài khoản của bạn đã bị khóa. Vui lòng liên hệ quản trị viên."

3.5.3: Bảng Đặc tả Usecase đăng ký tài khoản



3.5.3.Sơ đồ UC đăng nhập

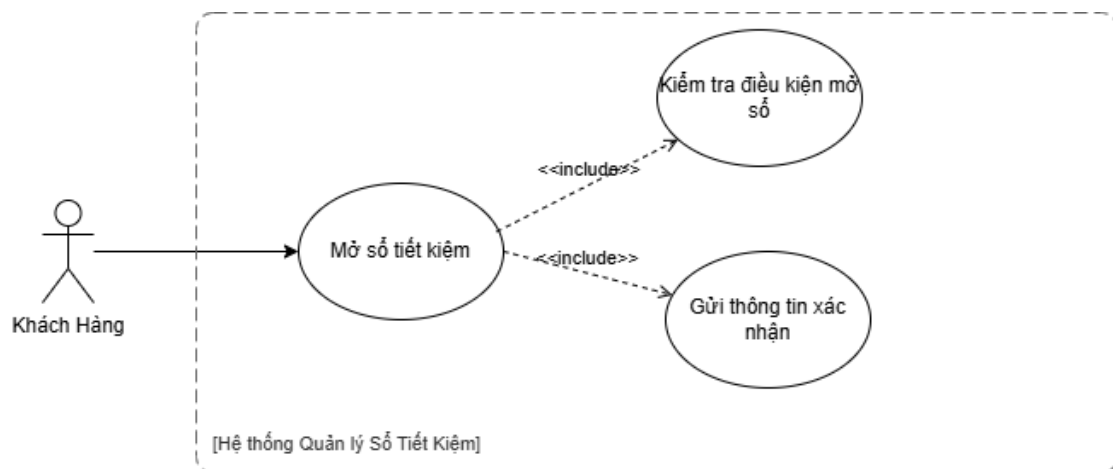
3.5.4 Sơ đồ Usecase mở sổ tiết kiệm

Thành phần	Mô tả
Tên Use Case	Mở Sổ Tiết Kiệm
Mã Use Case	UC003
Tác nhân	<ul style="list-style-type: none"> - Khách hàng (Primary Actor): Người muốn mở sổ tiết kiệm. - Nhân viên ngân hàng (Secondary Actor): Hỗ trợ khách hàng trong quá trình mở sổ.
Mục tiêu	Tạo một sổ tiết kiệm mới cho khách hàng, lưu thông tin và cập nhật dữ liệu trong hệ thống.

Điều kiện tiên quyết	<ul style="list-style-type: none"> - Khách hàng phải cung cấp đầy đủ thông tin cần thiết như họ tên, CMND/CCCD, và địa chỉ. - Số tiền gửi ban đầu phải đáp ứng mức tối thiểu theo quy định của hệ thống.
Đảm bảo tối thiểu	- Nếu thông tin không hợp lệ hoặc số tiền gửi không đạt yêu cầu, sổ tiết kiệm không được tạo.
Đảm bảo thành công	- Sổ tiết kiệm được tạo thành công, thông tin được lưu trong hệ thống và khách hàng nhận được thông báo.
Kích hoạt	Nhân viên hoặc khách hàng nhấn nút "Mở Sổ Tiết Kiệm" sau khi nhập thông tin và số tiền.
Chuỗi sự kiện chính	<ol style="list-style-type: none"> 1. Khách hàng cung cấp thông tin cá nhân và lựa chọn loại tiết kiệm (không kỳ hạn, 3 tháng, 6 tháng). 2. Khách hàng nhập số tiền gửi ban đầu. 3. Nhân viên kiểm tra thông tin và nhấn nút "Mở Sổ Tiết Kiệm". 4. Hệ thống kiểm tra các điều kiện: <ul style="list-style-type: none"> - Thông tin khách hàng phải hợp lệ (CMND/CCCD, họ tên, địa chỉ). - Số tiền gửi ban đầu phải đạt tối thiểu 1.000.000 VNĐ theo quy định. 5. Nếu thông tin hợp lệ: <ul style="list-style-type: none"> - Hệ thống lưu thông tin sổ tiết kiệm mới vào cơ sở dữ liệu. - Gửi thông báo tạo sổ thành công cho khách hàng.

Ngoại lệ	<ol style="list-style-type: none"> 1. Thông tin khách hàng không hợp lệ: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Thông tin khách hàng không hợp lệ. Vui lòng kiểm tra lại." 2. Số tiền gửi không đạt yêu cầu: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Số tiền gửi tối thiểu là 1.000.000 VNĐ." 3. Lỗi hệ thống: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Hệ thống gặp sự cố. Vui lòng thử lại sau."
----------	---

3.5.4:Bảng Đặc tả Usecase mở sổ tiết kiệm



3.5.4.Sơ đồ UC mở sổ tiết kiệm

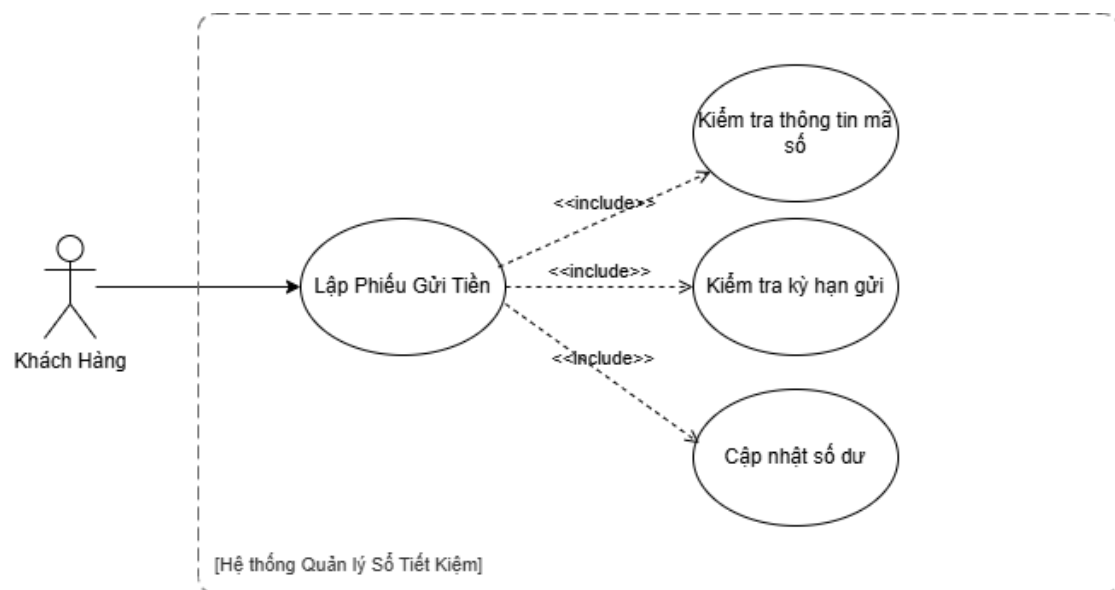
3.5.5 Sơ đồ UC lập phiếu gửi tiền

Thành phần	Mô tả
Tên Use Case	Lập Phiếu Gửi Tiền
Mã Use Case	UC004
Tác nhân	<ul style="list-style-type: none">- Khách hàng (Primary Actor): Người muốn gửi thêm tiền vào sổ tiết kiệm.- Nhân viên ngân hàng (Secondary Actor): Thực hiện thao tác lập phiếu gửi tiền cho khách hàng.
Mục tiêu	Gửi thêm tiền vào sổ tiết kiệm hiện có của khách hàng, cập nhật số dư và ghi nhận giao dịch.
Điều kiện tiên quyết	<ul style="list-style-type: none">- Sổ tiết kiệm của khách hàng phải tồn tại và hợp lệ.- Số tiền gửi thêm phải đạt mức tối thiểu theo quy định (100.000 VNĐ).- Loại tiết kiệm có kỳ hạn chỉ nhận gửi tiền khi đến kỳ hạn tính lãi.
Đảm bảo tối thiểu	<ul style="list-style-type: none">- Nếu thông tin không hợp lệ hoặc không đáp ứng yêu cầu, giao dịch gửi tiền không được thực hiện.

Đảm bảo thành công	- Giao dịch gửi tiền được thực hiện thành công, số dư sổ tiết kiệm được cập nhật, và phiếu gửi tiền được lưu trữ.
Kích hoạt	Nhân viên hoặc khách hàng nhấn nút "Gửi Tiền" sau khi nhập thông tin sổ tiết kiệm và số tiền gửi.
Chuỗi sự kiện chính	<ol style="list-style-type: none"> 1. Khách hàng cung cấp thông tin sổ tiết kiệm (mã số hoặc số CMND/CCCD liên kết). 2. Khách hàng nhập số tiền muốn gửi thêm. 3. Nhân viên kiểm tra thông tin và nhấn nút "Gửi Tiền". 4. Hệ thống kiểm tra các điều kiện: <ul style="list-style-type: none"> - Sổ tiết kiệm tồn tại và hợp lệ. - Số tiền gửi thêm đạt tối thiểu 100.000 VNĐ. - Loại tiết kiệm có kỳ hạn chỉ cho phép gửi thêm khi đến kỳ hạn tính lãi. 5. Nếu thông tin hợp lệ: <ul style="list-style-type: none"> - Hệ thống cập nhật số dư sổ tiết kiệm. - Lưu thông tin giao dịch gửi tiền vào cơ sở dữ liệu. - In phiếu gửi tiền và thông báo giao dịch thành công.
Ngoại lệ	<ol style="list-style-type: none"> 1. Sổ tiết kiệm không hợp lệ: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Sổ tiết kiệm không tồn tại hoặc đã bị khóa."

	<p>2. Số tiền gửi không đạt yêu cầu:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Số tiền gửi thêm tối thiểu là 100.000 VNĐ." <p>3. Loại tiết kiệm không cho phép gửi thêm:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Chỉ gửi thêm tiền khi đến kỳ hạn." <p>4. Lỗi hệ thống:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Hệ thống gặp sự cố. Vui lòng thử lại sau."
--	---

3.5.5: Bảng Đặc tả Usecase lập phiếu gửi tiền



3.5.5.Sơ đồ UC lập phiếu gửi tiền

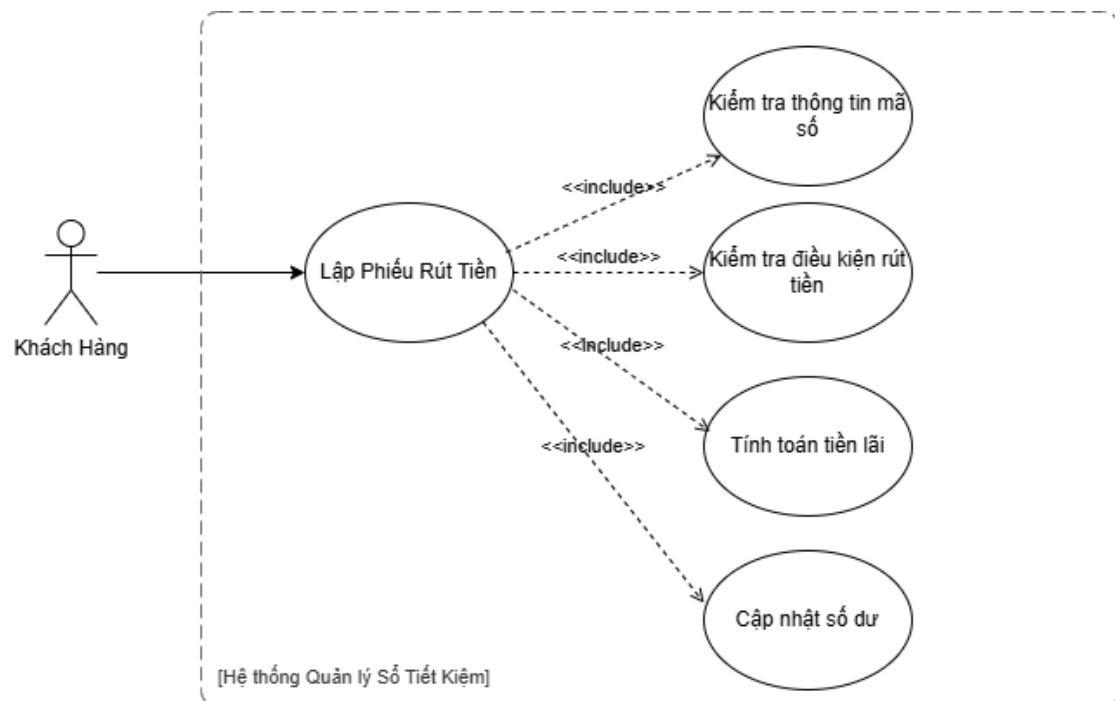
3.5.6 Sơ đồ Usecase lập phiếu rút tiền

Thành phần	Mô tả
Tên Use Case	Lập Phiếu Rút Tiền
Mã Use Case	UC005
Tác nhân	<ul style="list-style-type: none">- Khách hàng (Primary Actor): Người muốn rút tiền từ sổ tiết kiệm.- Nhân viên ngân hàng (Secondary Actor): Thực hiện thao tác lập phiếu rút tiền cho khách hàng.
Mục tiêu	Thực hiện giao dịch rút tiền từ sổ tiết kiệm của khách hàng, cập nhật số dư và ghi nhận giao dịch.
Điều kiện tiên quyết	<ul style="list-style-type: none">- Sổ tiết kiệm của khách hàng phải tồn tại và hợp lệ.- Số tiền rút phải thỏa mãn điều kiện theo loại sổ tiết kiệm (kỳ hạn hoặc không kỳ hạn).- Sổ tiết kiệm có kỳ hạn chỉ được rút sau khi hết kỳ hạn và phải rút toàn bộ số tiền.
Đảm bảo tối thiểu	- Nếu thông tin không hợp lệ hoặc không đáp ứng điều kiện rút, giao dịch sẽ không được thực hiện.
Đảm bảo thành công	- Giao dịch rút tiền được thực hiện thành công, số dư sổ tiết kiệm được cập nhật và phiếu rút tiền được lưu trữ.

Kích hoạt	Nhân viên hoặc khách hàng nhấn nút "Rút Tiền" sau khi nhập thông tin sổ tiết kiệm và số tiền rút.
Chuỗi sự kiện chính	<ol style="list-style-type: none"> 1. Khách hàng cung cấp thông tin sổ tiết kiệm (mã số hoặc số CMND/CCCD liên kết). 2. Khách hàng nhập số tiền muốn rút. 3. Nhân viên kiểm tra thông tin và nhấn nút "Rút Tiền". 4. Hệ thống kiểm tra các điều kiện: <ul style="list-style-type: none"> - Sổ tiết kiệm tồn tại và hợp lệ. - Số tiền rút không vượt quá số dư hiện tại. - Loại tiết kiệm có kỳ hạn chỉ cho phép rút toàn bộ số tiền sau khi hết kỳ hạn. - Loại tiết kiệm không kỳ hạn chỉ cho phép rút nếu thời gian gửi ≥ 15 ngày. 5. Nếu thông tin hợp lệ: <ul style="list-style-type: none"> - Hệ thống cập nhật số dư sổ tiết kiệm. - Lưu thông tin giao dịch rút tiền vào cơ sở dữ liệu. - In phiếu rút tiền và thông báo giao dịch thành công.
Ngoại lệ	<ol style="list-style-type: none"> 1. Sổ tiết kiệm không hợp lệ: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Sổ tiết kiệm không tồn tại hoặc đã bị khóa."

	<p>2. Số tiền rút vượt số dư:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Số tiền rút không được vượt quá số dư hiện tại." <p>3. Loại tiết kiệm không cho phép rút:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Số tiết kiệm có kỳ hạn chỉ được rút toàn bộ sau khi hết kỳ hạn." - Hệ thống hiển thị thông báo "Số tiết kiệm không kỳ hạn chỉ được rút nếu thời gian gửi ≥ 15 ngày." <p>4. Lỗi hệ thống:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Hệ thống gặp sự cố. Vui lòng thử lại sau."
--	---

3.5.6: Bảng Đặc tả Usecase lập phiếu rút tiền



3.5.6.Sơ đồ UC lập phiếu rút tiền

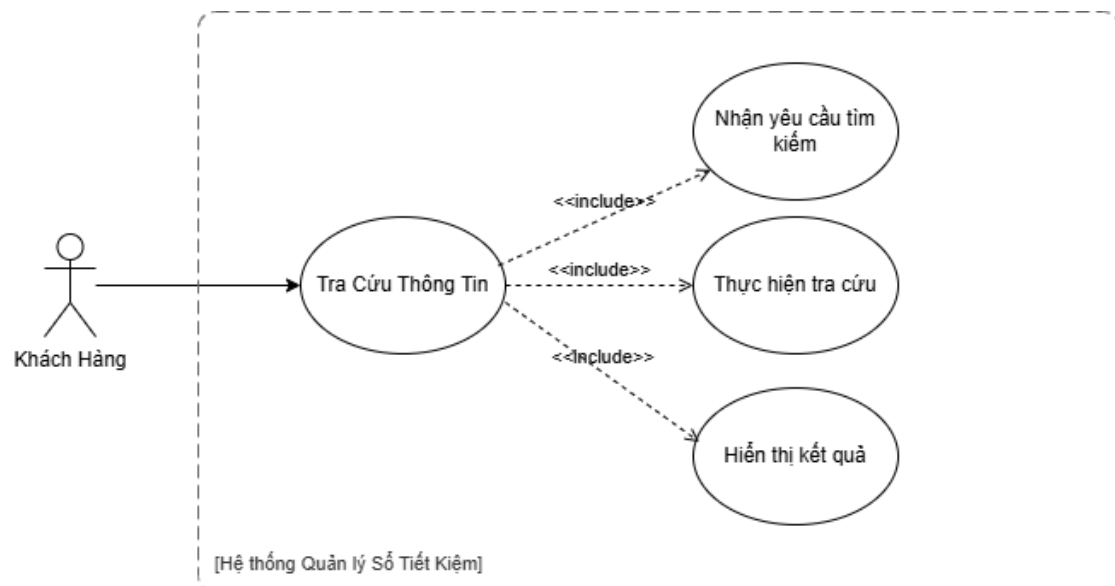
3.5.7 Sơ đồ Usecase tra cứu thông tin

Thành phần	Mô tả
Tên Use Case	Tra Cứu Thông Tin
Mã Use Case	UC006
Tác nhân	<ul style="list-style-type: none">- Khách hàng (Primary Actor): Người cần tra cứu thông tin về sổ tiết kiệm của mình.- Nhân viên ngân hàng (Secondary Actor): Hỗ trợ khách hàng thực hiện tra cứu nếu cần.
Mục tiêu	Hiển thị thông tin chi tiết của sổ tiết kiệm, bao gồm loại tiết kiệm, số dư, lịch sử giao dịch và trạng thái hiện tại.
Điều kiện tiên quyết	<ul style="list-style-type: none">- Sổ tiết kiệm phải tồn tại và thông tin tra cứu phải chính xác (mã số sổ tiết kiệm hoặc thông tin khách hàng).- Khách hàng hoặc nhân viên phải có quyền truy cập vào thông tin cần tra cứu.
Đảm bảo tối thiểu	- Nếu không tìm thấy thông tin hoặc thông tin không hợp lệ, hệ thống hiển thị thông báo lỗi.
Đảm bảo thành công	- Thông tin chi tiết của sổ tiết kiệm được hiển thị đầy đủ và chính xác cho tác nhân tra cứu.

Kích hoạt	Khách hàng hoặc nhân viên nhập thông tin sổ tiết kiệm và nhấn nút "Tra Cứu".
Chuỗi sự kiện chính	<ol style="list-style-type: none"> 1. Khách hàng hoặc nhân viên mở giao diện tra cứu thông tin sổ tiết kiệm. 2. Nhập thông tin tra cứu: mã số sổ tiết kiệm, CMND/CCCD hoặc số tài khoản liên kết. 3. Nhấn nút "Tra Cứu" để gửi yêu cầu. 4. Hệ thống kiểm tra thông tin tra cứu: <ul style="list-style-type: none"> - Xác nhận mã số sổ tiết kiệm có tồn tại không. - Kiểm tra quyền truy cập thông tin của tác nhân. 5. Nếu thông tin hợp lệ: <ul style="list-style-type: none"> - Hệ thống truy xuất dữ liệu từ cơ sở dữ liệu. - Hiển thị thông tin chi tiết của sổ tiết kiệm bao gồm: loại tiết kiệm, số dư, trạng thái hiện tại, và lịch sử giao dịch.
Ngoại lệ	<ol style="list-style-type: none"> 1. Thông tin không hợp lệ: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Thông tin tra cứu không hợp lệ. Vui lòng kiểm tra lại." 2. Sổ tiết kiệm không tồn tại: <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Sổ tiết kiệm không tồn tại hoặc đã bị đóng." 3. Lỗi quyền truy cập:

	<p>- Hệ thống hiển thị thông báo "Bạn không có quyền truy cập thông tin này."</p> <p>4. Lỗi hệ thống:</p> <p>- Hệ thống hiển thị thông báo "Hệ thống gặp sự cố. Vui lòng thử lại sau."</p>
--	--

3.5.7: Bảng Đặc tả Usecase Tra cứu thông tin



3.5.7.Sơ đồ UC tra cứu thông tin

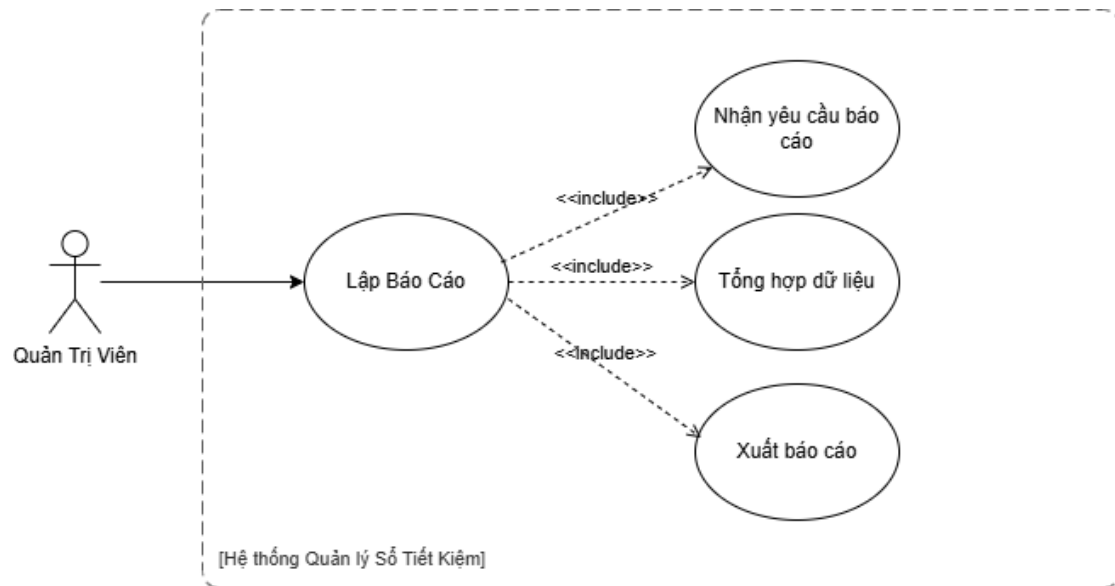
3.5.8 Sơ đồ Usecase lập báo cáo

Thành phần	Mô tả
Tên Use Case	Lập Báo Cáo

Mã Use Case	UC007
Tác nhân	<ul style="list-style-type: none"> - Quản trị viên (Primary Actor): Người chịu trách nhiệm quản lý và theo dõi báo cáo tài chính. - Hệ thống (Secondary Actor): Thực hiện tổng hợp và xuất báo cáo dựa trên dữ liệu có sẵn.
Mục tiêu	Tạo báo cáo chi tiết theo các tiêu chí định sẵn (doanh thu, số lượng sổ mở/đóng, giao dịch...) để phục vụ công tác quản lý.
Điều kiện tiên quyết	<ul style="list-style-type: none"> - Dữ liệu giao dịch và sổ tiết kiệm phải tồn tại và chính xác trong cơ sở dữ liệu. - Quản trị viên phải có quyền truy cập chức năng lập báo cáo.
Đảm bảo tối thiểu	- Nếu không đủ dữ liệu hoặc lỗi xảy ra trong quá trình xử lý, hệ thống hiển thị thông báo lỗi.
Đảm bảo thành công	- Báo cáo được tạo và hiển thị đầy đủ, có thể xuất ra các định dạng như PDF hoặc Excel nếu cần.
Kích hoạt	Quản trị viên chọn loại báo cáo cần lập và nhấn nút "Tạo Báo Cáo".
Chuỗi sự kiện chính	<ol style="list-style-type: none"> 1. Quản trị viên mở giao diện lập báo cáo. 2. Chọn loại báo cáo cần lập (doanh thu ngày, tháng, hoặc thống kê sổ mở/đóng). 3. Nhập các thông tin lọc báo cáo (ví dụ: khoảng thời gian, loại tiết kiệm).

	<p>4. Nhấn nút "Tạo Báo Cáo" để gửi yêu cầu.</p> <p>5. Hệ thống xử lý yêu cầu:</p> <ul style="list-style-type: none"> - Truy xuất dữ liệu từ cơ sở dữ liệu theo các tiêu chí lọc. - Tổng hợp và tính toán thông tin cần thiết. <p>6. Nếu xử lý thành công:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị báo cáo chi tiết trên giao diện. - Quản trị viên có thể xuất báo cáo ra định dạng mong muốn (PDF hoặc Excel).
Ngoại lệ	<p>1. Không có dữ liệu phù hợp:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Không tìm thấy dữ liệu phù hợp với tiêu chí lọc." <p>2. Lỗi quyền truy cập:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Bạn không có quyền thực hiện chức năng này." <p>3. Lỗi trong xử lý dữ liệu:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Hệ thống gặp sự cố. Vui lòng thử lại sau." <p>4. Lỗi xuất báo cáo:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Không thể xuất báo cáo. Vui lòng kiểm tra lại định dạng tệp."

3.5.8: Bảng Đặc tả Usecase Lập báo cáo



3.5.8.Sơ đồ UC lập báo cáo

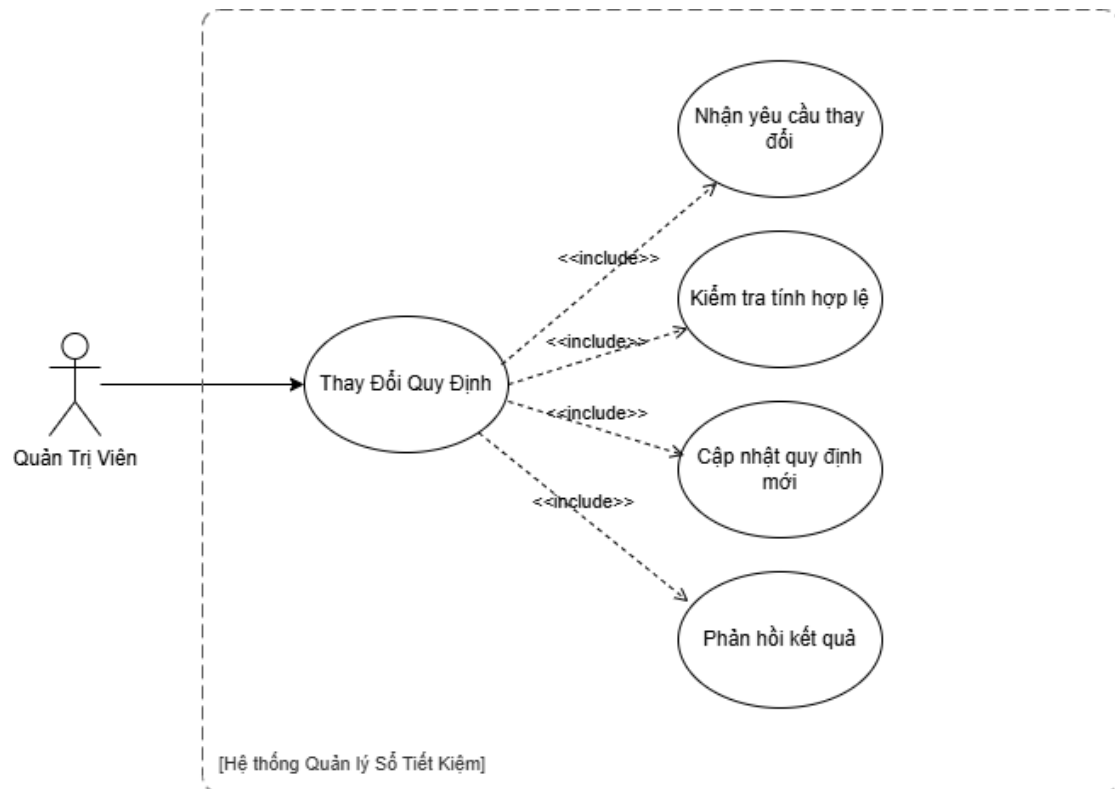
3.5.9 Sơ đồ Usecase thay đổi quy định

Thành phần	Mô tả
Tên Use Case	Thay Đổi Quy Định
Mã Use Case	UC008
Tác nhân	- Quản trị viên (Primary Actor): Người chịu trách nhiệm thay đổi các quy định của hệ thống.
Mục tiêu	Cập nhật các quy định trong hệ thống (lãi suất, kỳ hạn, số tiền gửi tối thiểu...) để phù hợp với yêu cầu quản lý.

Điều kiện tiên quyết	<ul style="list-style-type: none"> - Quản trị viên phải có quyền truy cập chức năng thay đổi quy định. - Các quy định mới phải tuân thủ các ràng buộc logic của hệ thống.
Đảm bảo tối thiểu	- Nếu quy định không hợp lệ, hệ thống sẽ từ chối cập nhật và hiển thị thông báo lỗi.
Đảm bảo thành công	- Quy định mới được cập nhật thành công và lưu vào cơ sở dữ liệu.
Kích hoạt	Quản trị viên chọn chức năng "Thay Đổi Quy Định" trên giao diện quản trị.
Chuỗi sự kiện chính	<ol style="list-style-type: none"> 1. Quản trị viên mở giao diện thay đổi quy định trong hệ thống. 2. Nhập thông tin quy định mới cần thay đổi (ví dụ: lãi suất, số tiền gửi tối thiểu, kỳ hạn tiết kiệm). 3. Nhấn nút "Lưu Quy Định" để gửi yêu cầu thay đổi. 4. Hệ thống kiểm tra các điều kiện: <ul style="list-style-type: none"> - Dữ liệu nhập vào hợp lệ (ví dụ: số tiền phải là số dương, lãi suất trong phạm vi cho phép). - Không vi phạm logic nghiệp vụ của hệ thống (ví dụ: kỳ hạn hợp lệ phải là số tháng). 5. Nếu thông tin hợp lệ: <ul style="list-style-type: none"> - Hệ thống cập nhật các quy định mới vào cơ sở dữ liệu. - Gửi thông báo xác nhận thay đổi thành công. 6. Quản trị viên nhận thông báo kết quả.

Ngoại lệ	<p>1. Quy định nhập không hợp lệ:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Thông tin nhập không hợp lệ. Vui lòng kiểm tra lại." <p>2. Lỗi quyền truy cập:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Bạn không có quyền thực hiện chức năng này." <p>3. Lỗi cập nhật cơ sở dữ liệu:</p> <ul style="list-style-type: none"> - Hệ thống hiển thị thông báo "Không thể lưu quy định. Vui lòng thử lại sau."

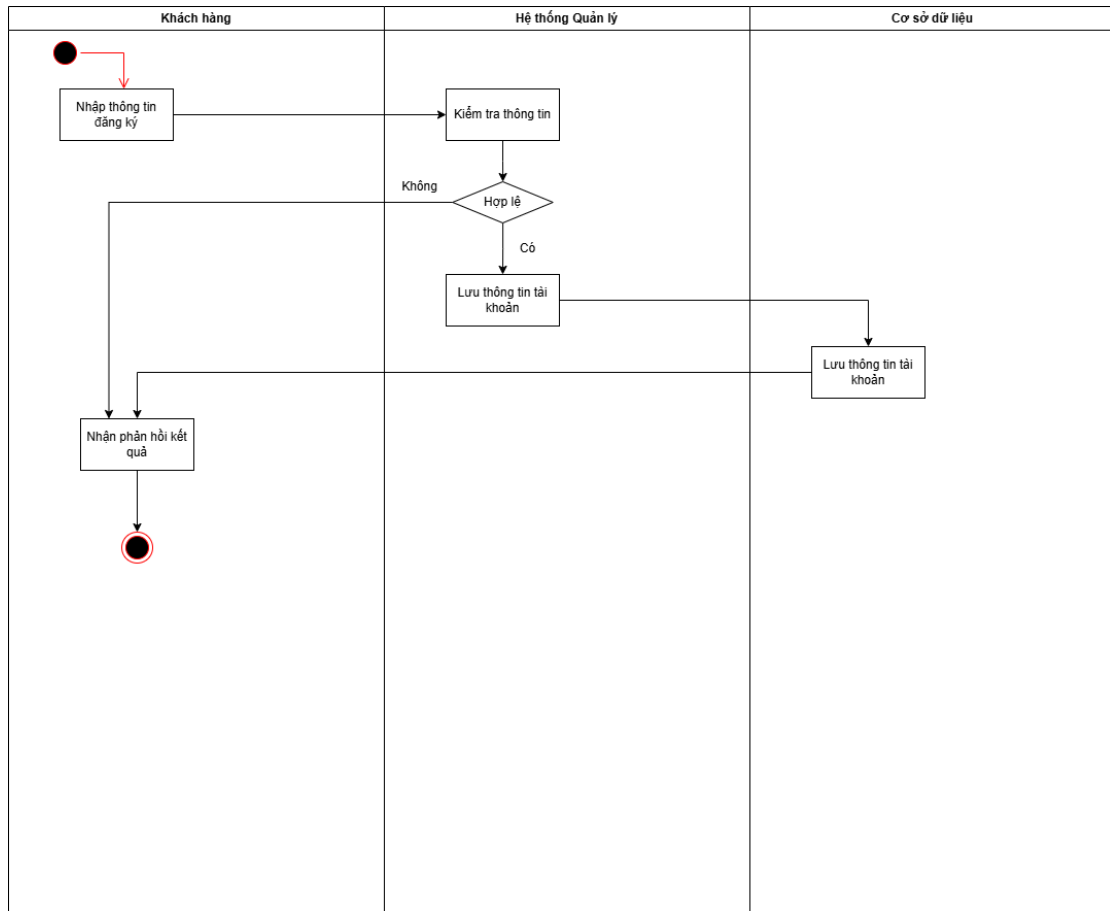
3.5.9: Bảng Đặc tả Usecase Thay đổi quy định



3.5.9.Sơ đồ UC thay đổi quy định

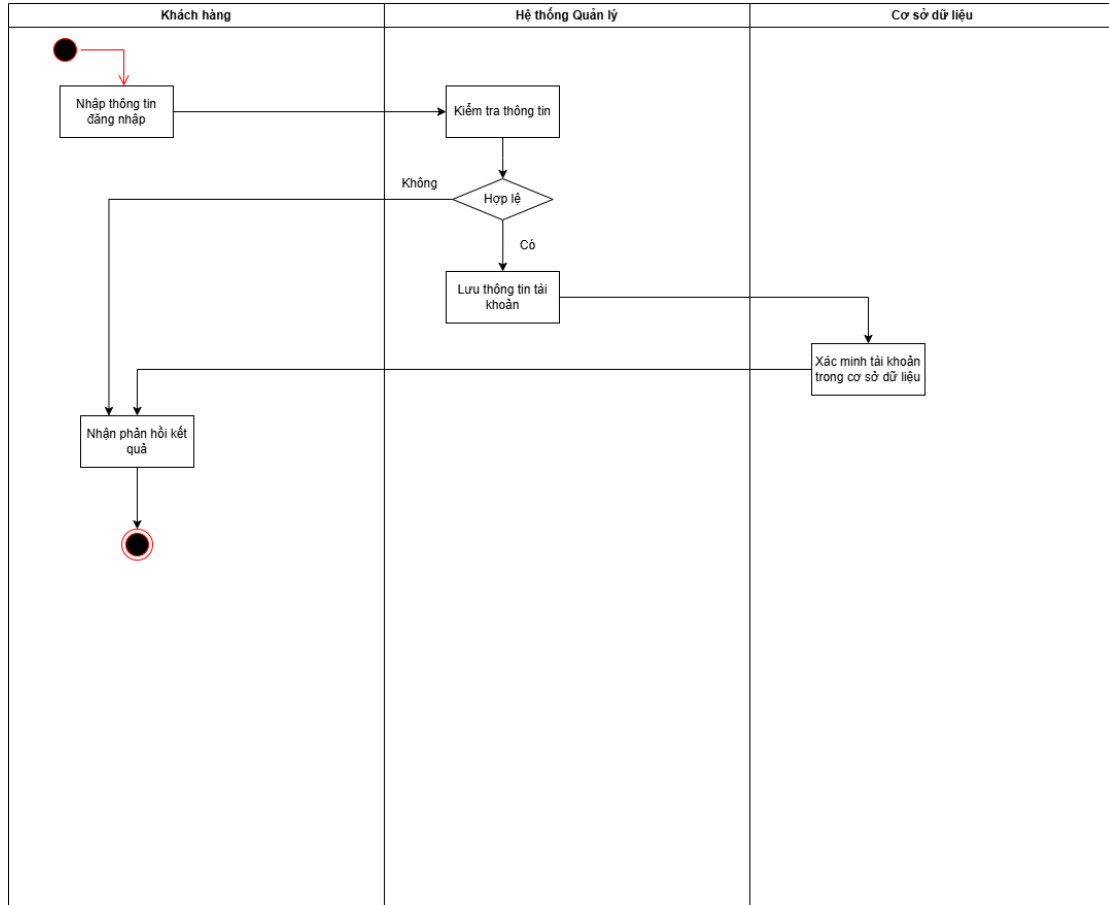
3.6.Sơ đồ hoạt động - AD

3.6.1. Sơ đồ hoạt động miêu tả người dùng đăng ký



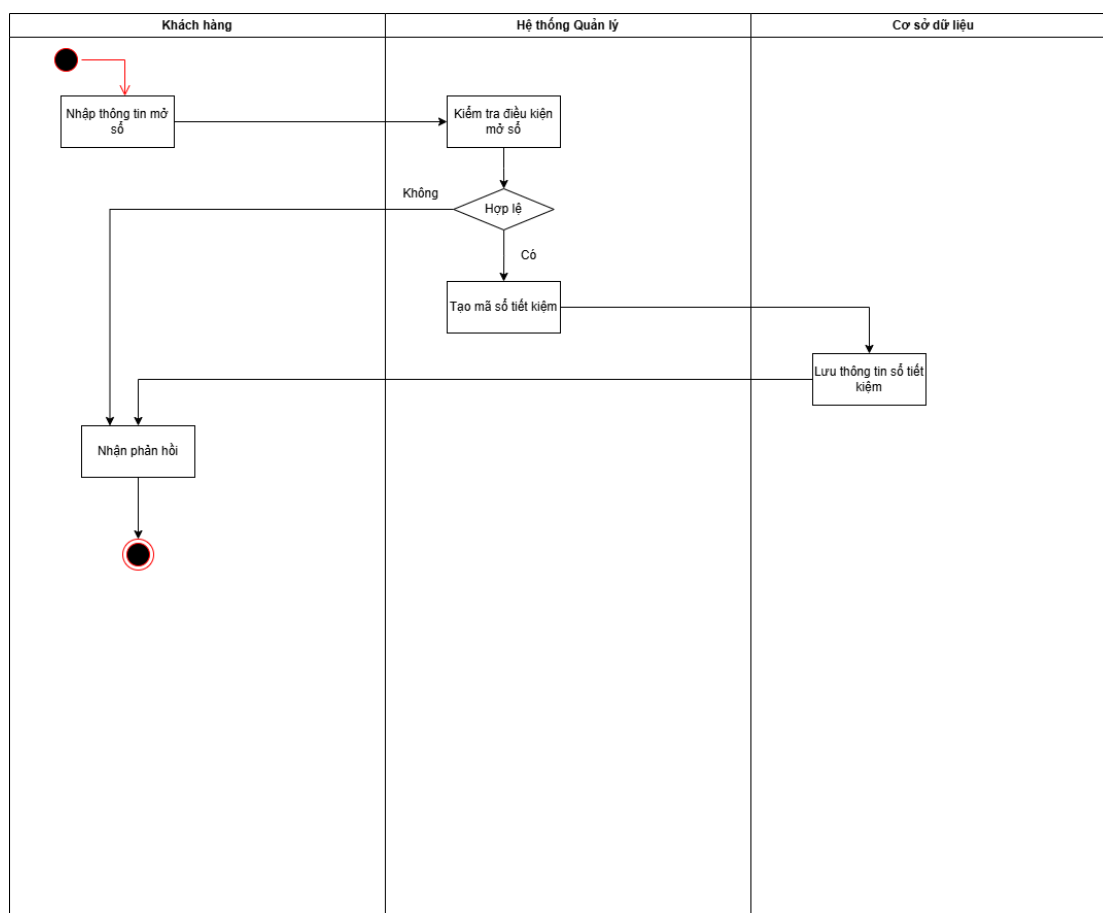
3.6.1. Sơ đồ hoạt động miêu tả người dùng đăng nhập bằng tài khoản

3.6.2. Sơ đồ hoạt động miêu tả người dùng đăng nhập



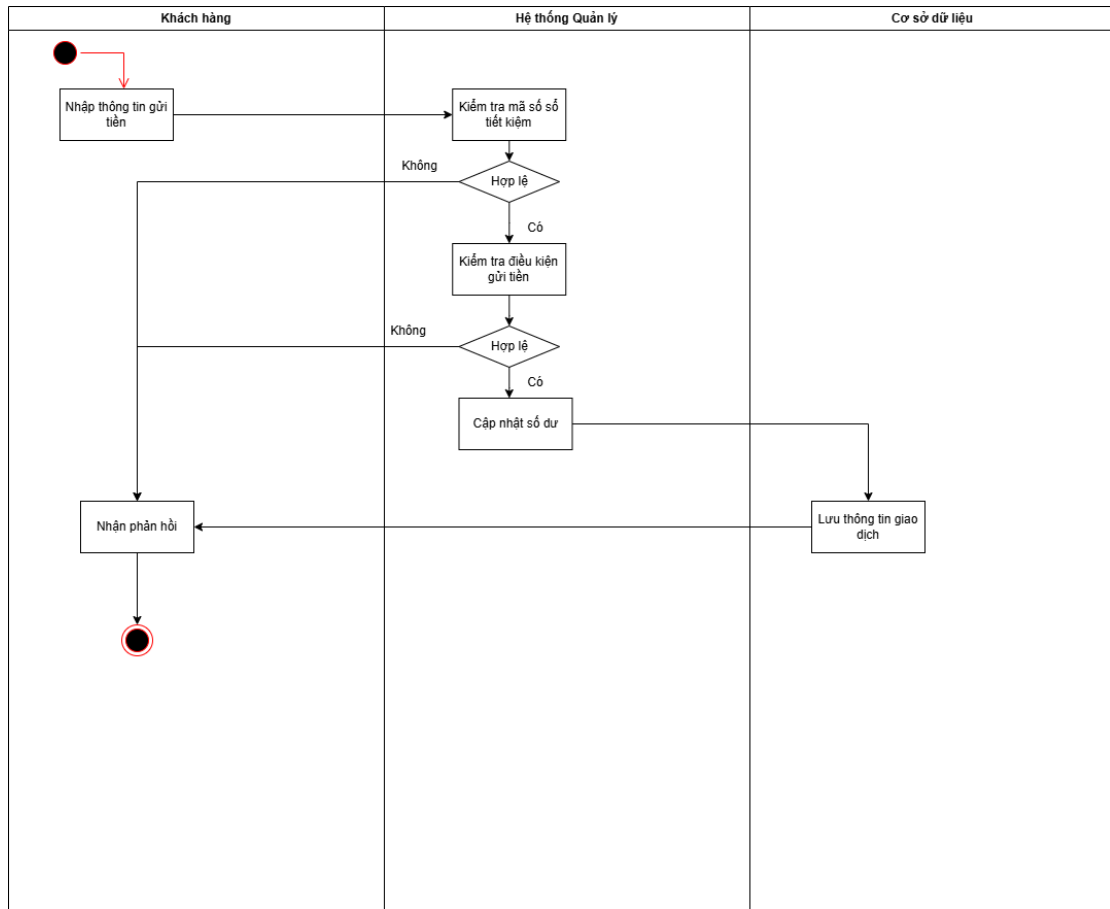
3.6.2 Sơ đồ hoạt động miêu tả người dùng nghe nhạc offline

3.6.3. Sơ đồ hoạt động miêu tả người dùng mở sổ tiết kiệm



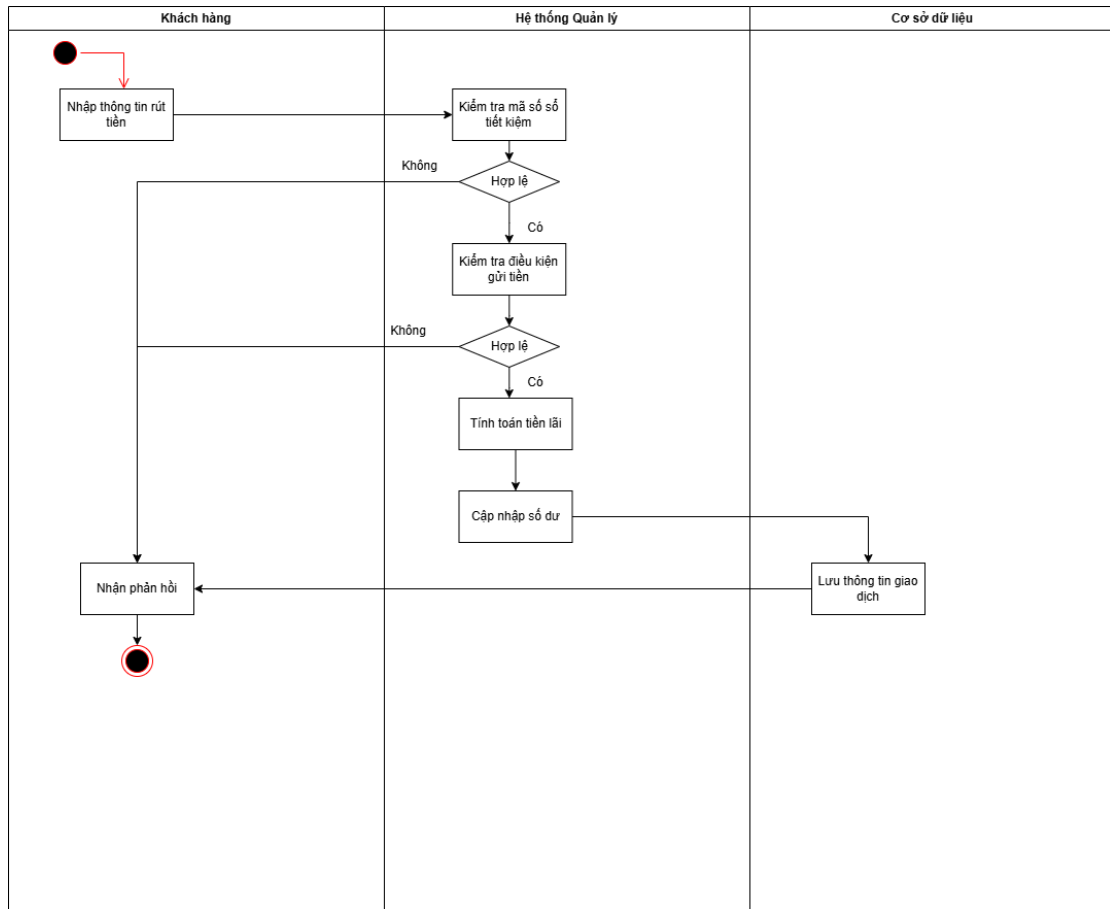
3.6.3. Sơ đồ hoạt động miêu tả người dùng mở sổ tiết kiệm

3.6.4. Sơ đồ hoạt động miêu tả người dùng Lập Phiếu Gửi Tiền



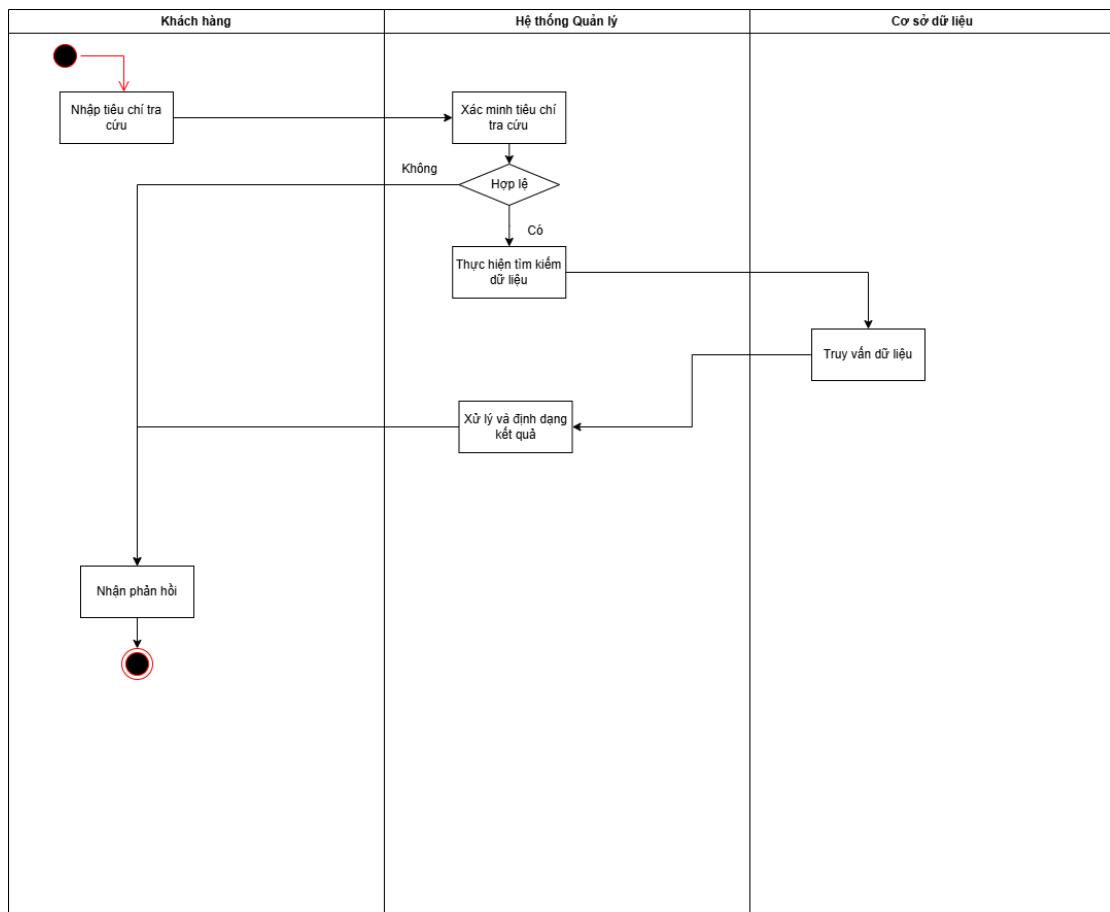
3.6.4. Sơ đồ hoạt động miêu tả người dùng Lập Phiếu Gửi Tiền

3.6.5. Sơ đồ hoạt động miêu tả người dùng Lập Phiếu Rút Tiền:



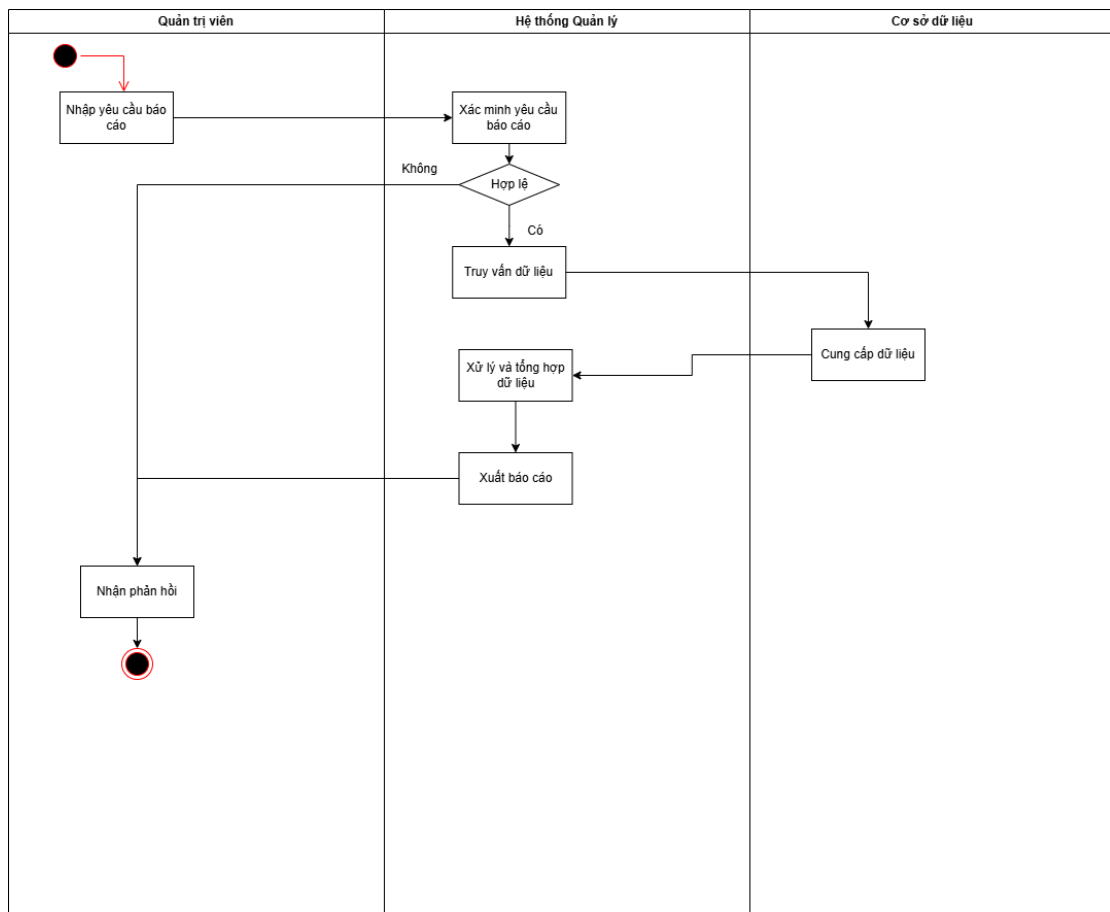
3.6.5 Sơ đồ hoạt động miêu tả người dùng Lập Phiếu Rút Tiền

3.6.6. Sơ đồ hoạt động miêu tả người dùng Tra Cứu Thông Tin:



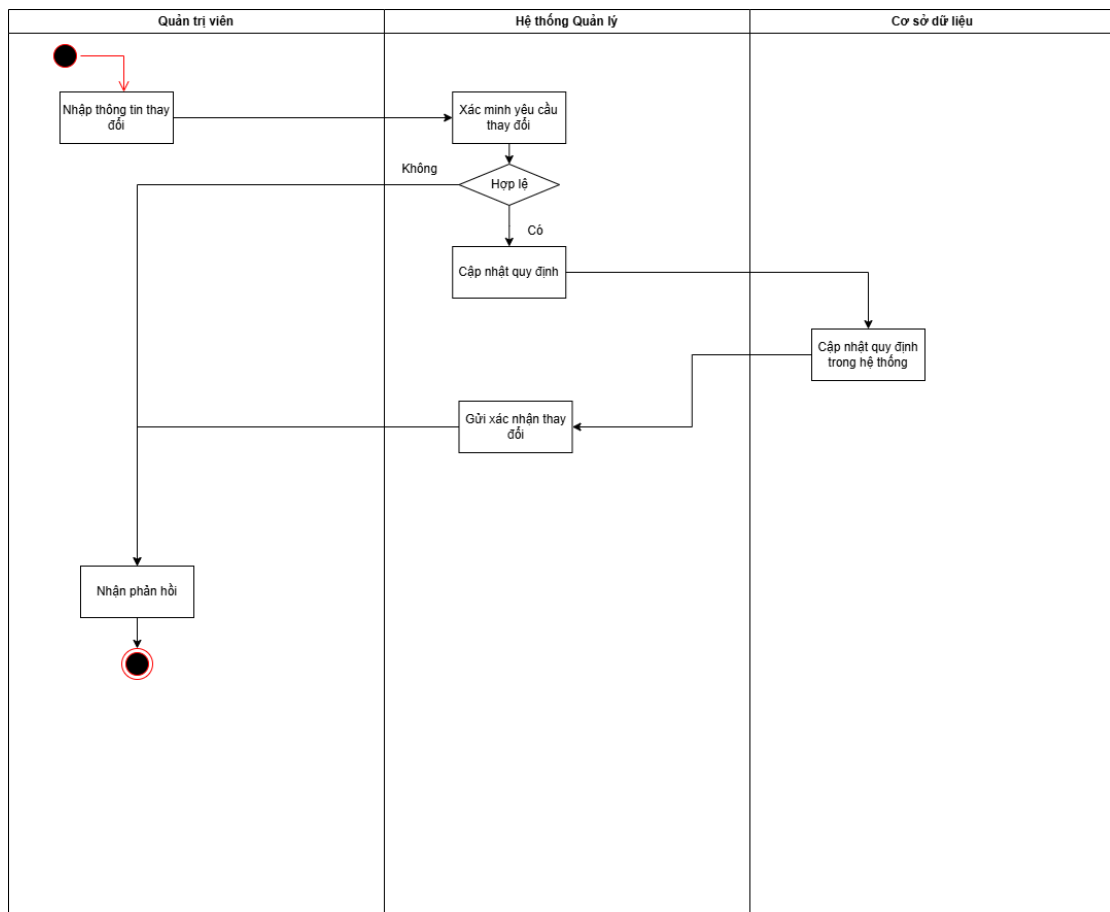
3.6.6 Sơ đồ hoạt động miêu tả người dùng Tra Cứu Thông Tin

3.6.7. Sơ đồ hoạt động miêu tả người dùng *Lập Báo Cáo*:



3.6.7 Sơ đồ hoạt động miêu tả người dùng *Lập Báo Cáo*

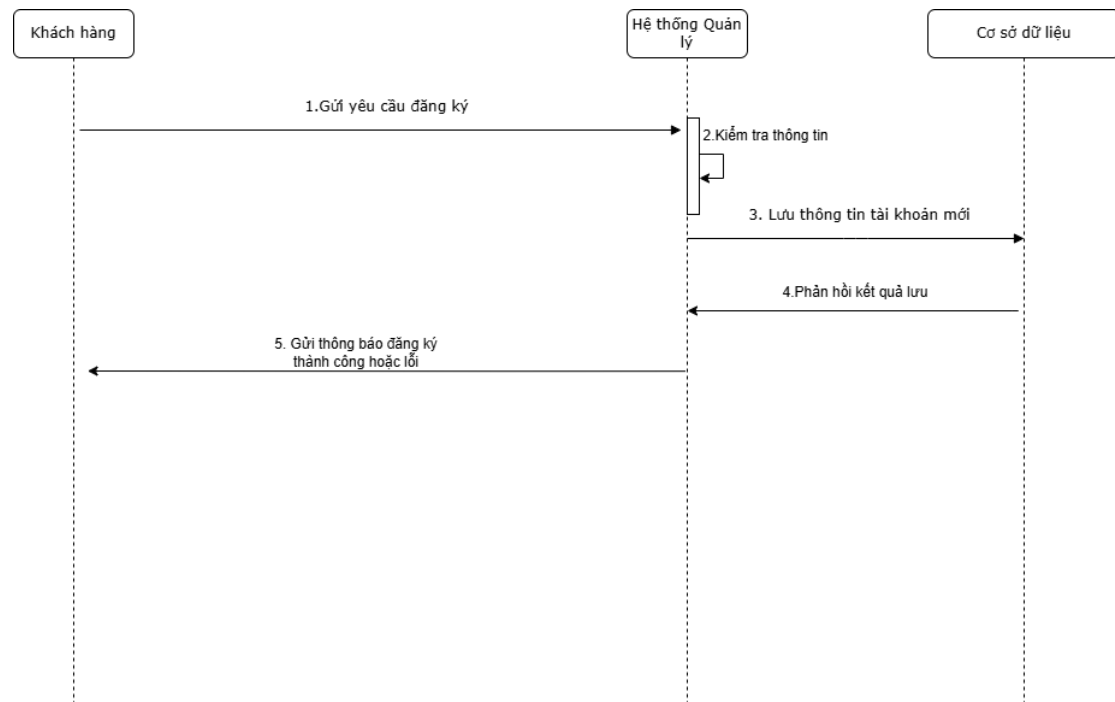
3.6.8. Sơ đồ hoạt động miêu tả người dùng Thay Đổi Quy Định:



3.6.8 Sơ đồ hoạt động miêu tả người dùng Thay Đổi Quy Định

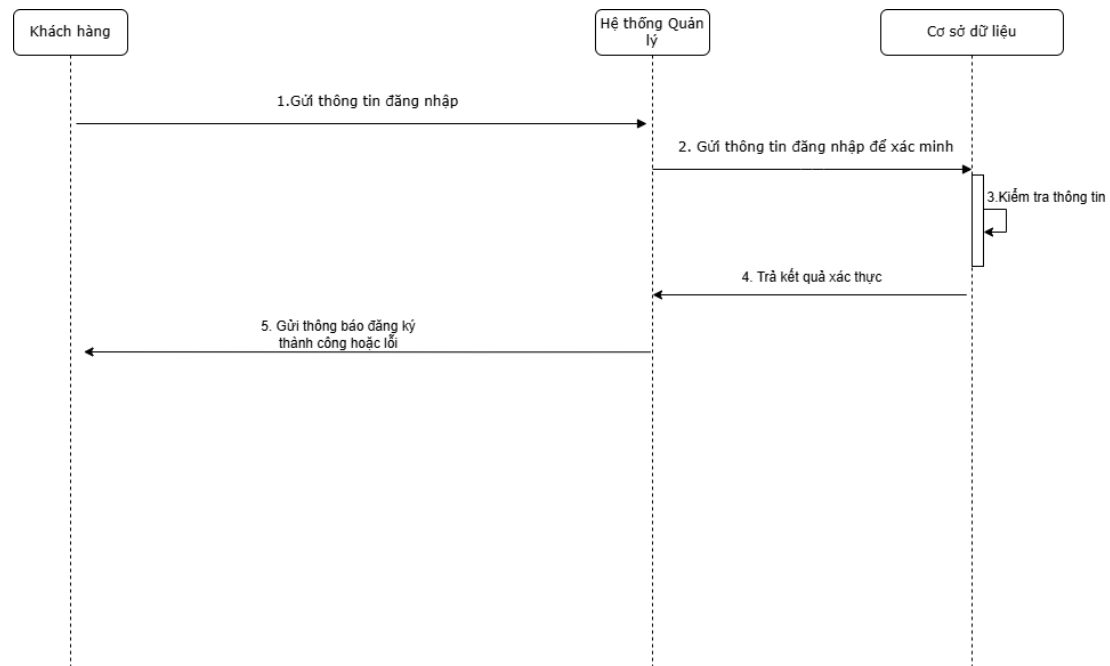
3.7. Sơ đồ tuần tự

3.7.1. Sơ đồ tuần tự đăng ký



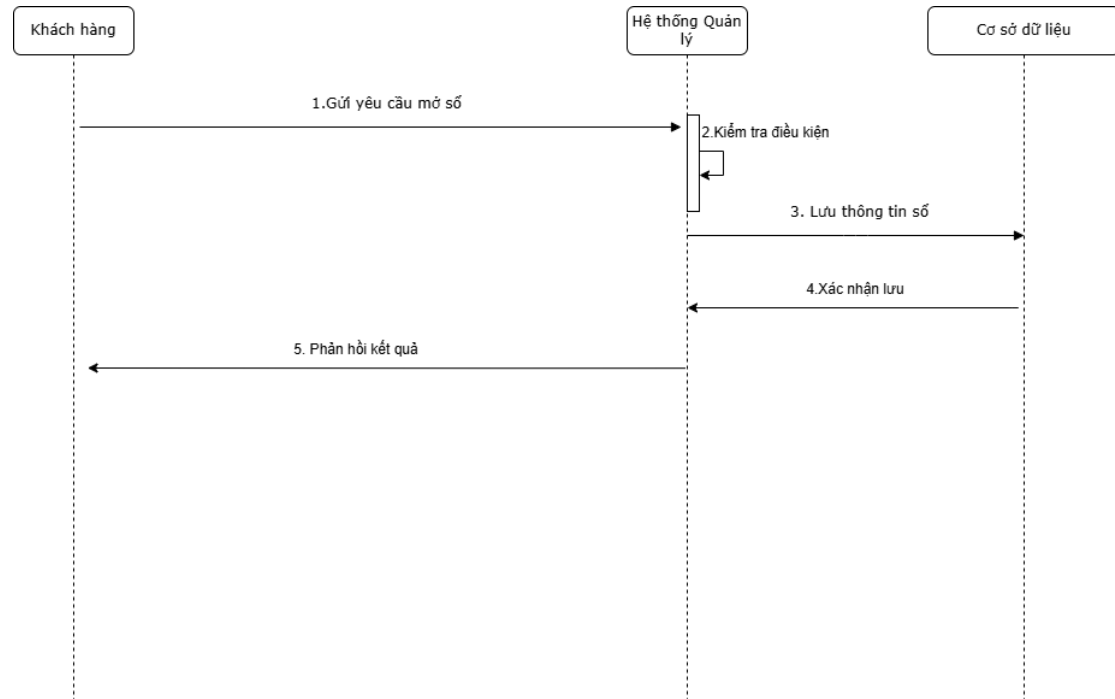
3.7.1. Sơ đồ tuần tự đăng ký

3.7.2. Sơ đồ tuần tự đăng nhập



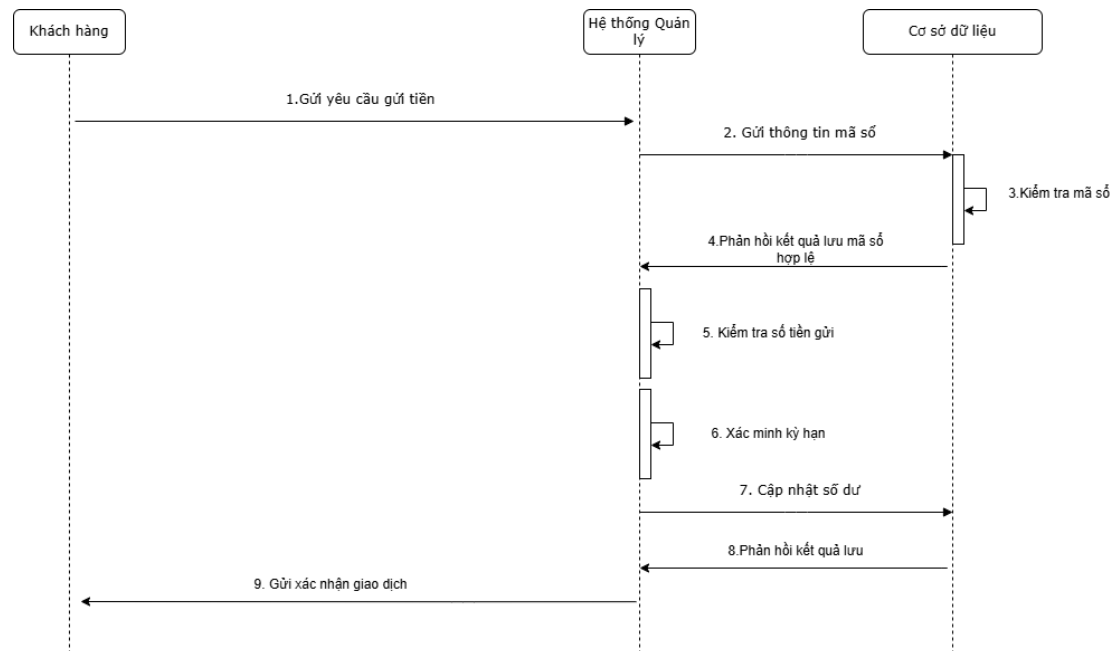
3.7.2. Sơ đồ tuần tự đăng nhập

3.7.3. Sơ đồ tuần tự Mở sổ



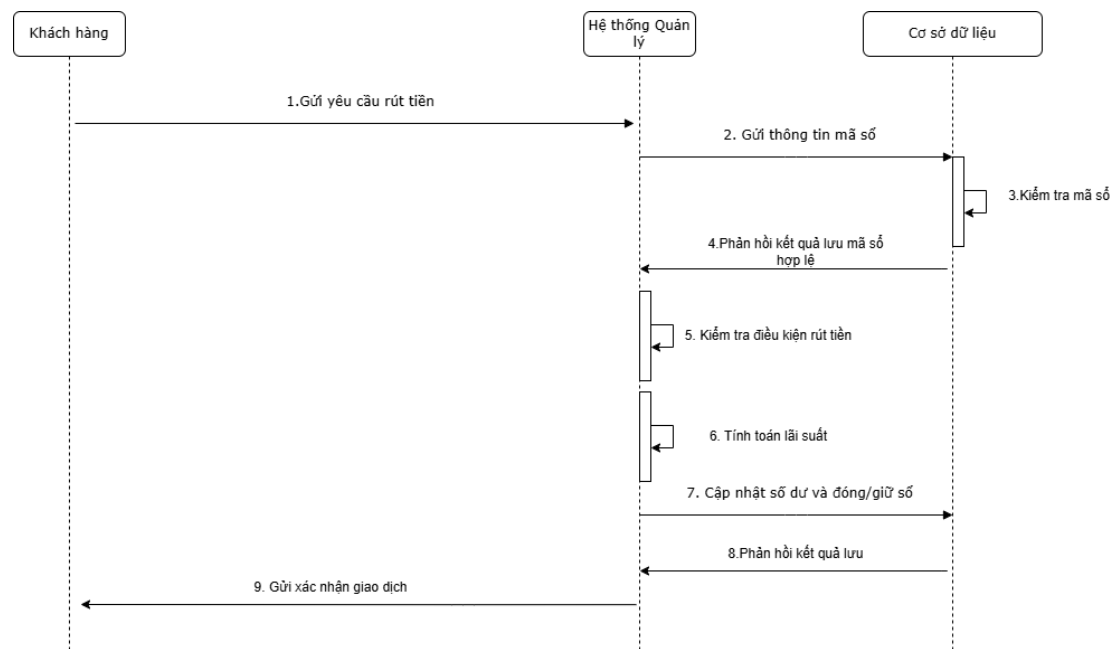
3.7.3. Sơ đồ tuần tự Mở sổ

3.7.4. Sơ đồ tuần tự Lập Phiếu Gửi Tiền



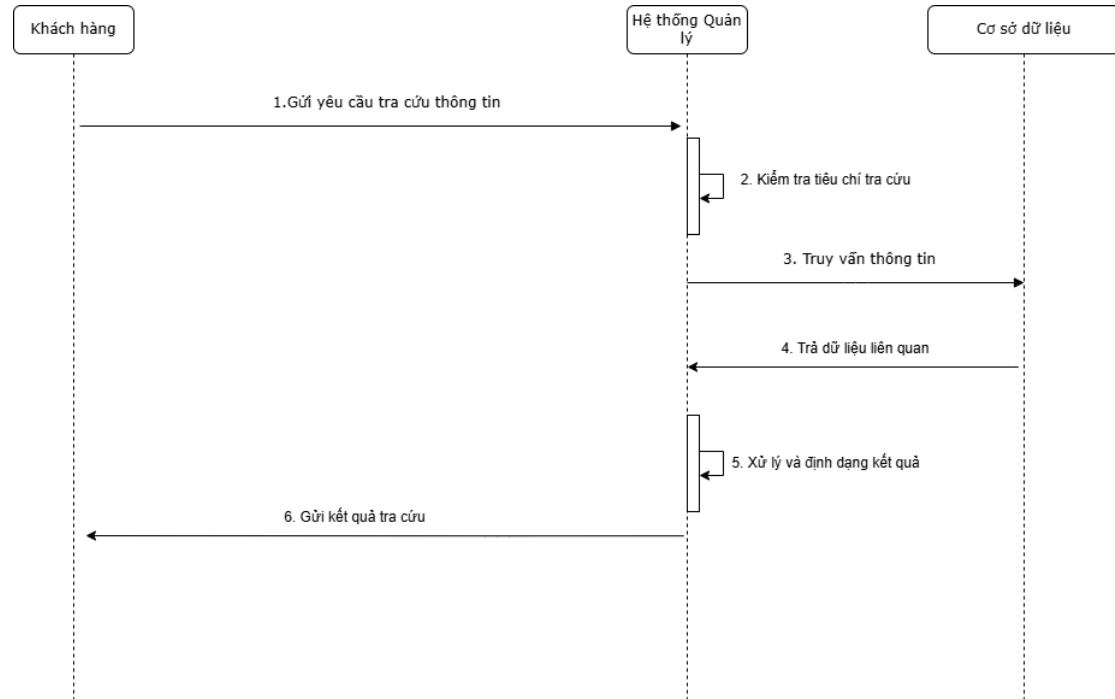
3.7.4. Sơ đồ tuần tự Lập Phiếu Gửi Tiền

3.7.5. Sơ đồ tuần tự Lập Phiếu Rút Tiền



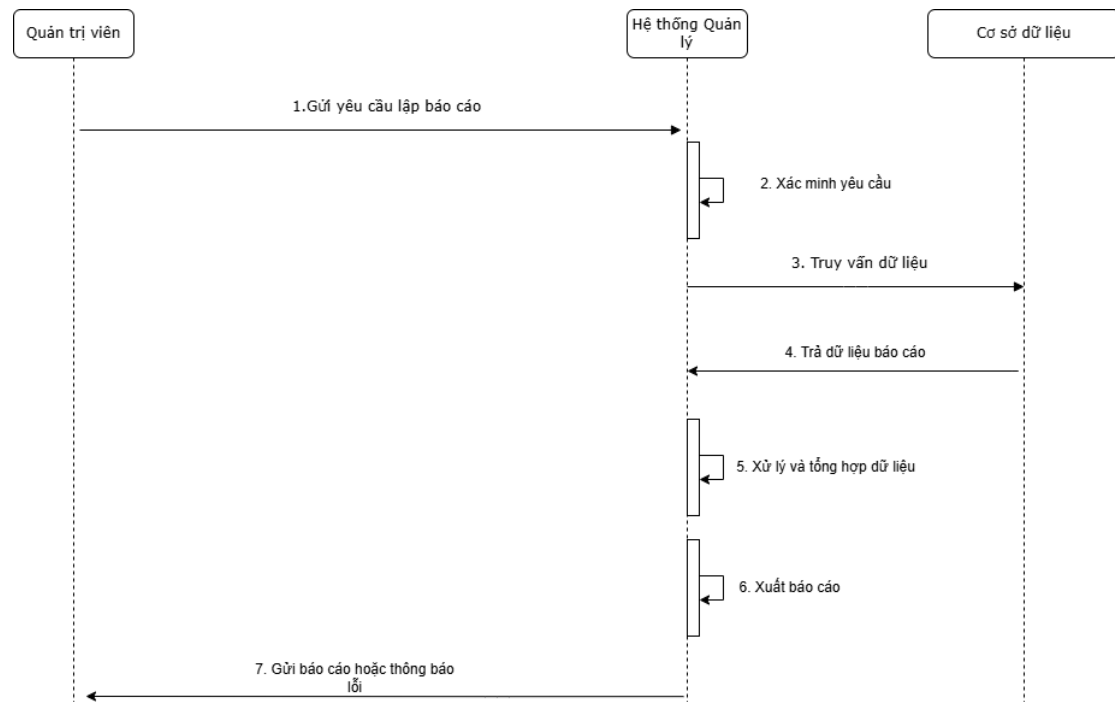
3.7.5. Sơ đồ tuần tự Lập Phiếu Rút Tiền

3.7.6. Sơ đồ tuần tự Tra Cứu Thông Tin



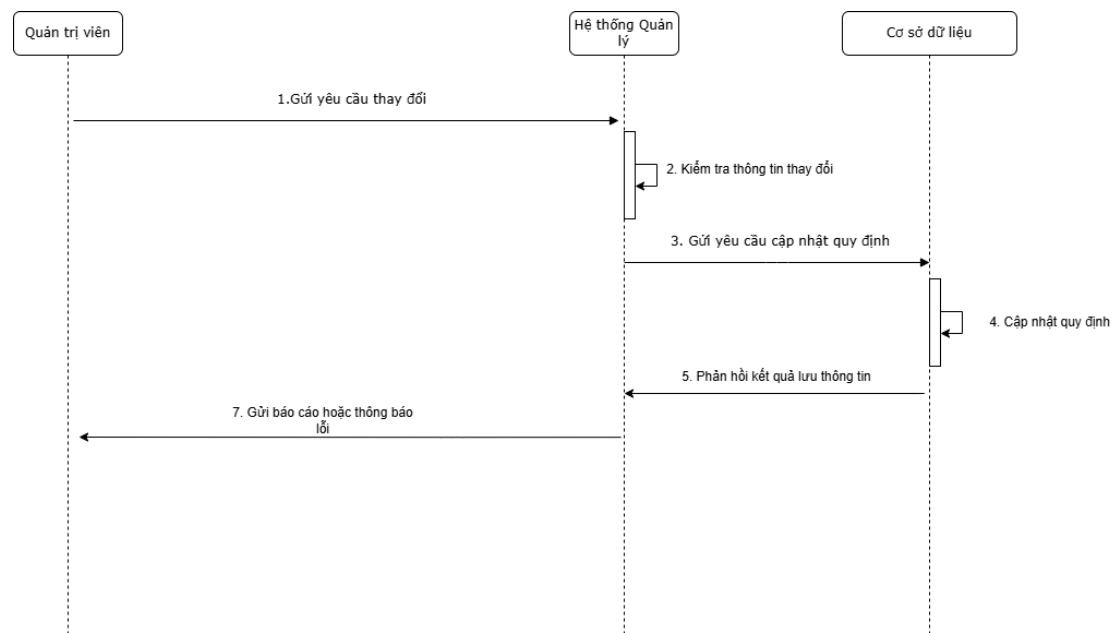
3.7.6. Sơ đồ tuần tự Tra Cứu Thông Tin

3.7.7. Sơ đồ tuần tự Lập Báo Cáo



3.7.7. Sơ đồ tuần tự Lập Báo Cáo

3.7.8. Sơ đồ tuần tự thay đổi quy định



3.7.8. Sơ đồ tuần tự thay đổi quy định

3.8. Class Diagram:



3.8. Class Diagram

Thành phần	Chi tiết
Tên lớp	passbook
Mô tả	Lớp đại diện cho thông tin sổ tiết kiệm của khách hàng.
Thuộc tính	<ul style="list-style-type: none"> - SavingsBookID: Mã định danh duy nhất của sổ tiết kiệm. - user_id: Mã người dùng sở hữu sổ tiết kiệm (liên kết với lớp users). - OpeningDate: Ngày mở sổ tiết kiệm. - InitialDepositAmount: Số tiền gửi ban đầu.

	<ul style="list-style-type: none"> - DepositAmount: Tổng số tiền hiện có trong sổ tiết kiệm. - InterestRate: Lãi suất áp dụng cho sổ tiết kiệm. - SavingsType: Loại tiết kiệm (liên kết với lớp SavingsAccountType). - IsClosed: Trạng thái của sổ tiết kiệm (đã đóng hoặc mở).
Quan hệ	<ul style="list-style-type: none"> - 1-nhiều với lớp SavingsDeposit. - 1-nhiều với lớp WithdrawalSlip. - Nhiều-1 với lớp SavingsAccountType.

Bảng 3.8.1: Bảng Đặc tả lớp Passbook

Thành phần	Chi tiết
Tên lớp	SavingsAccountType
Mô tả	Lớp lưu thông tin các loại hình tiết kiệm.
Thuộc tính	<ul style="list-style-type: none"> - SavingsTypeID: Mã định danh loại tiết kiệm. - AccountTypeName: Tên loại tiết kiệm (ví dụ: không kỳ hạn, 3 tháng, 6 tháng). - InterestRate: Lãi suất mặc định của loại tiết kiệm. - Term: Kỳ hạn tiết kiệm (số tháng).

	<ul style="list-style-type: none"> - WithdrawalDays: Số ngày tối thiểu trước khi được rút tiền. - MinimumDeposit: Số tiền gửi tối thiểu. - IsActive: Trạng thái hoạt động của loại tiết kiệm. - AllowsAdditionalDeposits: Cho phép gửi thêm tiền hay không.
Quan hệ	- 1-nhiều với lớp passbook.

Bảng 3.8.2: Bảng Đặc tả lớp SavingsAccountType

Thành phần	Chi tiết
Tên lớp	SavingsDeposit
Mô tả	Lớp quản lý các giao dịch gửi tiền của khách hàng.
Thuộc tính	<ul style="list-style-type: none"> - DepositID: Mã định danh giao dịch gửi tiền. - SavingsBookID: Liên kết với sổ tiết kiệm (lớp passbook). - user_id: Mã người dùng thực hiện giao dịch. - DepositAmount: Số tiền được gửi thêm. - InterestRate: Lãi suất áp dụng cho giao dịch. - DepositDate: Ngày thực hiện giao dịch gửi tiền.
Quan hệ	- Nhiều-1 với lớp passbook.

	- Nhiều-1 với lớp users.
--	--------------------------

Bảng 3.8.3: Bảng Đặc tả lớp SavingsDeposit

Thành phần	Chi tiết
Tên lớp	users
Mô tả	Lớp đại diện cho thông tin của người dùng trong hệ thống.
Thuộc tính	<ul style="list-style-type: none"> - user_id: Mã định danh của người dùng. - user_name: Tên đầy đủ của người dùng. - user_birth: Ngày sinh của người dùng. - user_address: Địa chỉ của người dùng. - user_identity: Số CMND/CCCD của người dùng. - user_phone: Số điện thoại liên lạc của người dùng. - email: Địa chỉ email của người dùng. - password: Mật khẩu đăng nhập (được mã hóa). - role: Vai trò của người dùng (khách hàng hoặc quản trị viên).
Quan hệ	- 1-nhiều với lớp passbook.

	<ul style="list-style-type: none"> - 1-nhiều với lớp SavingsDeposit. - 1-nhiều với lớp WithdrawalSlip.
--	--

Bảng 3.8.4: Bảng Đặc tả lớp users

Thành phần	Chi tiết
Tên lớp	WithdrawalSlip
Mô tả	Lớp lưu thông tin các giao dịch rút tiền từ sổ tiết kiệm.
Thuộc tính	<ul style="list-style-type: none"> - WithdrawalID: Mã định danh giao dịch rút tiền. - SavingsBookID: Liên kết với sổ tiết kiệm (lớp passbook). - user_id: Mã người dùng thực hiện giao dịch. - WithdrawalAmount: Số tiền được rút từ sổ tiết kiệm. - WithdrawalDate: Ngày thực hiện giao dịch rút tiền.
Quan hệ	<ul style="list-style-type: none"> - Nhiều-1 với lớp passbook. - Nhiều-1 với lớp users.

Bảng 3.8.5: Bảng Đặc tả lớp WithdrawalSlip

CHƯƠNG 4: THIẾT KẾ HỆ THỐNG

4.1. Kiến trúc hệ thống

4.1.1 Tổng quan kiến trúc

Hệ thống Quản Lý Sổ Tiết Kiệm được thiết kế dựa trên mô hình kiến trúc 3 lớp (Three-Tier Architecture), một mô hình phổ biến trong phát triển phần mềm hiện đại. Mô hình này phân chia hệ thống thành ba lớp chính: Lớp Giao Diện Người Dùng (Presentation Layer), Lớp Xử Lý Nghiệp Vụ (Business Logic Layer), và Lớp Dữ Liệu (Data Layer). Việc phân tách này giúp hệ thống dễ bảo trì, mở rộng, và tăng tính linh hoạt.

4.1.2 Lớp Giao Diện Người Dùng (Presentation Layer)

Lớp giao diện người dùng chịu trách nhiệm tương tác trực tiếp với người dùng thông qua giao diện web. Đây là lớp tiếp nhận yêu cầu từ người dùng và hiển thị kết quả trả về từ hệ thống.

- Các công nghệ sử dụng:
 - ASP.NET MVC: Cung cấp giao diện người dùng thân thiện, trực quan.
 - HTML, CSS, JavaScript, Bootstrap: Tạo giao diện web động, tương thích với nhiều thiết bị.
- Nhiệm vụ chính:
 - Hiển thị thông tin sổ tiết kiệm và các giao dịch.
 - Nhận yêu cầu từ người dùng, bao gồm đăng ký tài khoản, đăng nhập, mở sổ, gửi tiền, rút tiền...
 - Chuyển dữ liệu từ người dùng đến lớp xử lý nghiệp vụ.

4.1.3 Lớp Xử Lý Nghiệp Vụ (Business Logic Layer)

Lớp xử lý nghiệp vụ là thành phần trung gian, chịu trách nhiệm thực hiện tất cả các logic nghiệp vụ của hệ thống.

Các công nghệ sử dụng: C# với ASP.NET MVC: Thực hiện các yêu cầu xử lý, tính toán và áp dụng các quy định.

Nhiệm vụ chính:

- Kiểm tra tính hợp lệ của dữ liệu (như số dư, kỳ hạn, hoặc thông tin giao dịch).
- Tính toán lãi suất, cập nhật số dư và xử lý các giao dịch như gửi tiền hoặc rút tiền.
- Thực hiện các quy định được quản trị viên thiết lập, ví dụ: thay đổi kỳ hạn, lãi suất.
- Kết nối với cơ sở dữ liệu để lưu trữ hoặc truy vấn thông tin.

4.1.4 Lớp Dữ Liệu (Data Layer)

Lớp dữ liệu chịu trách nhiệm quản lý và lưu trữ toàn bộ dữ liệu của hệ thống, đảm bảo thông tin được lưu trữ an toàn và truy xuất nhanh chóng.

Các công nghệ sử dụng:

- SQL Server: Cơ sở dữ liệu quan hệ dùng để lưu trữ thông tin khách hàng, sổ tiết kiệm, giao dịch, và các quy định.

Nhiệm vụ chính:

- Lưu trữ dữ liệu liên quan đến sổ tiết kiệm, giao dịch tài chính, và quy định hệ thống.
- Đảm bảo an toàn dữ liệu thông qua cơ chế phân quyền và mã hóa.

- Thực hiện các truy vấn lấy dữ liệu hoặc cập nhật thông tin khi có yêu cầu từ lớp xử lý nghiệp vụ.

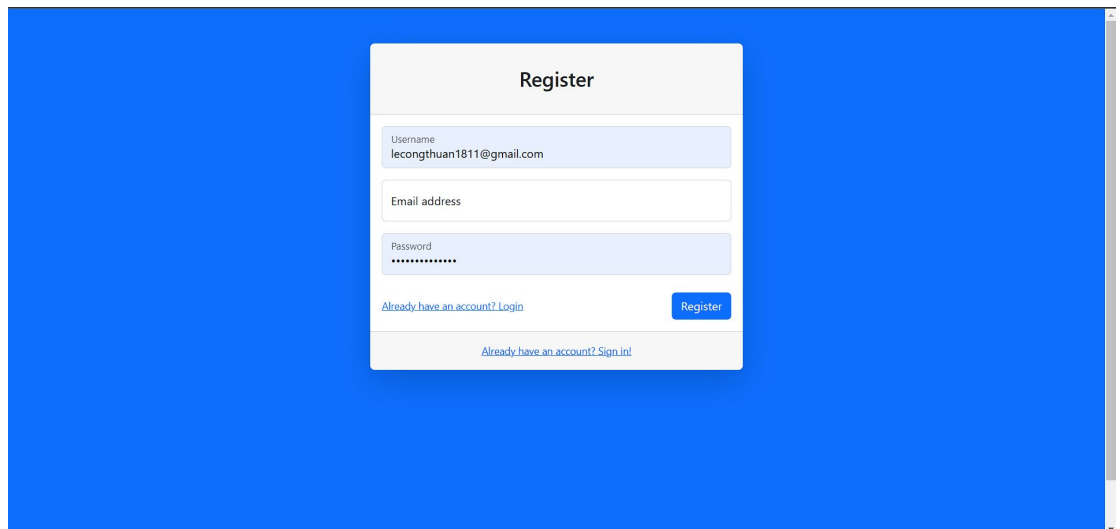
4.1.5 Lợi ích của kiến trúc 3 lớp

Kiến trúc 3 lớp mang lại nhiều lợi ích quan trọng, giúp tối ưu hóa việc phát triển và vận hành hệ thống:

1. Dễ bảo trì và nâng cấp:
 - Việc phân chia hệ thống thành các lớp riêng biệt giúp cô lập từng phần, giảm thiểu tác động khi cần sửa đổi hoặc nâng cấp một lớp cụ thể.
2. Tăng tính tái sử dụng:
 - Các logic nghiệp vụ được thiết kế độc lập, có thể tái sử dụng trong các chức năng khác hoặc trong các dự án tương tự.
3. Bảo mật cao:
 - Lớp xử lý nghiệp vụ đóng vai trò trung gian, ngăn chặn truy cập trực tiếp từ người dùng đến dữ liệu, đảm bảo tính bảo mật.
4. Khả năng mở rộng:
 - Hệ thống dễ dàng mở rộng về cả quy mô lẫn chức năng mà không ảnh hưởng đến cấu trúc hiện tại.
5. Hiệu suất cao:
 - Mỗi lớp được tối ưu hóa để xử lý nhiệm vụ riêng, giúp cải thiện tốc độ và hiệu quả của hệ thống.

4.2. Thiết kế giao diện:

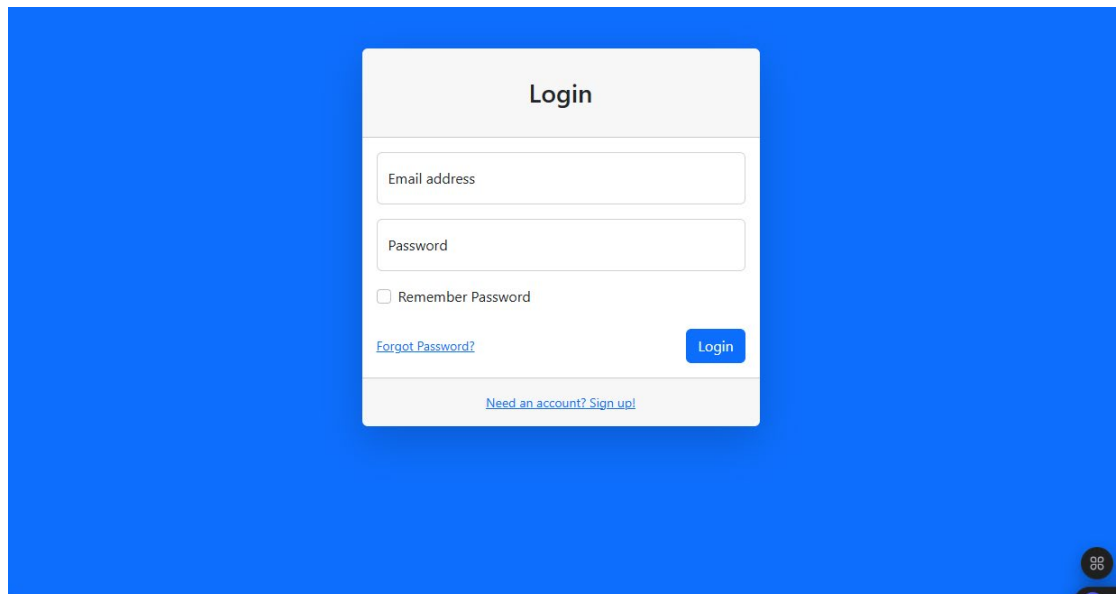
4.2.1. Giao diện khi đăng kí



The image shows a 'Register' form centered on a solid blue background. The form is a white rectangle with a light gray header containing the title 'Register'. It features three input fields: 'Username' (containing 'lecongthuan1811@gmail.com'), 'Email address' (empty), and 'Password' (masked with dots). Below the password field is a link 'Already have an account? Login'. At the bottom left is another link 'Already have an account? Sign in!', and at the bottom right is a blue 'Register' button.

4.2.1. Giao diện khi đăng kí

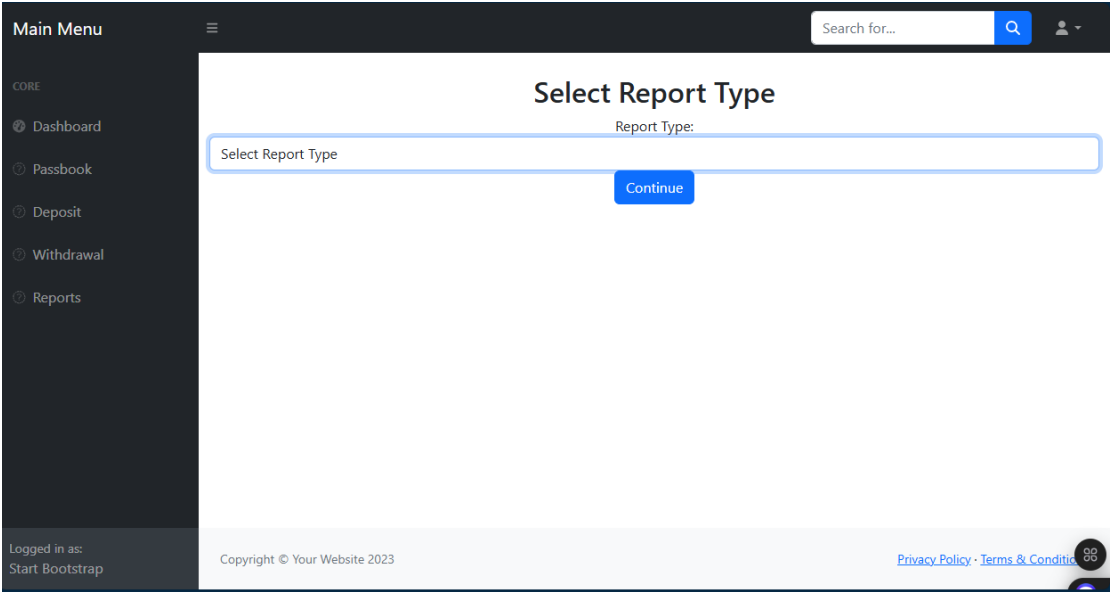
4.2.2. Giao diện khi đăng nhập



The image shows a 'Login' form centered on a solid blue background. The form is a white rectangle with a light gray header containing the title 'Login'. It features two input fields: 'Email address' and 'Password'. Below the password field is a checkbox labeled 'Remember Password'. At the bottom left is a link 'Forgot Password?', and at the bottom right is a blue 'Login' button. At the very bottom of the form is a link 'Need an account? Sign up!'. A small circular icon is visible in the bottom right corner of the blue background.

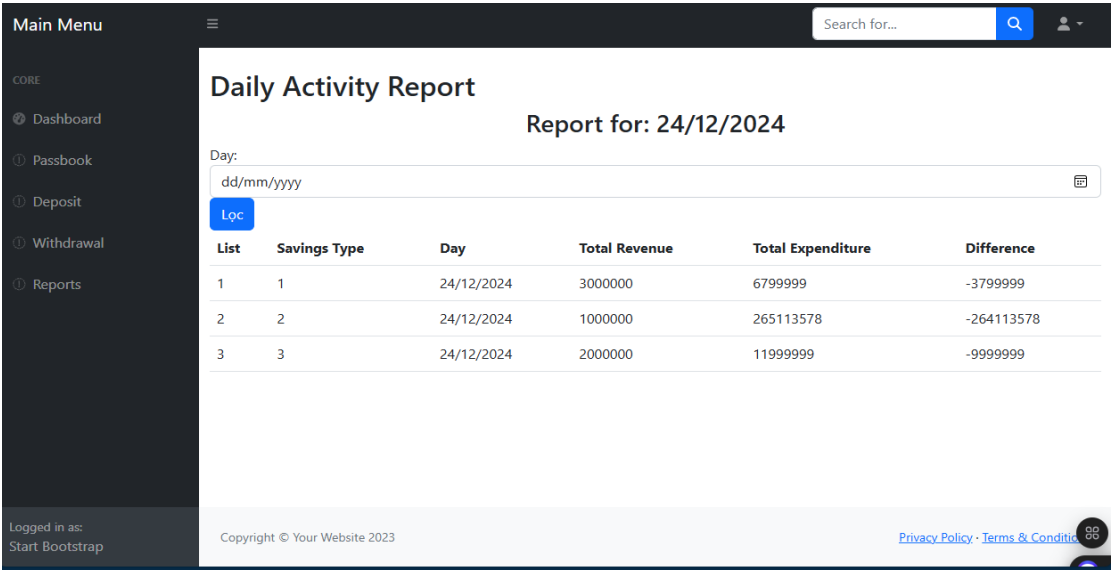
4.2.2. Giao diện khi đăng nhập

4.2.3. *Giao diện chọn loại báo cáo*



4.2.3. Giao diện chọn loại báo cáo

4.2.4:*Giao diện báo cáo theo ngày*



4.2.4: Giao diện báo cáo theo ngày

4.2.5 Giao diện báo cáo theo tháng

Main Menu

CORE

- Dashboard
- Passbook
- Deposit
- Withdrawal
- Reports

Logged in as:
Start Bootstrap

Search for...

Monthly Passbooks opening/closing report

Month:

Savings Type:

1

Lọc

Report for: 12/2024

Savings Type: 1

List	Savings Type	Day	Total Revenue	Total Expenditure	Difference
1	1	23/12/2024	1000000	0	1000000
2	1	24/12/2024	3000000	6799999	-3799999

Copyright © Your Website 2023

[Privacy Policy](#) · [Terms & Conditions](#)

4.2.5: Giao diện báo cáo theo tháng

4.2.6: Giao diện phiếu rút tiền

Main Menu

CORE

- Dashboard
- Passbook
- Deposit
- Withdrawal
- Reports

Logged in as:
Start Bootstrap

Search for...

Withdrawal Slips

Tim kiếm theo PassBook ID:

Nhập PassBook ID

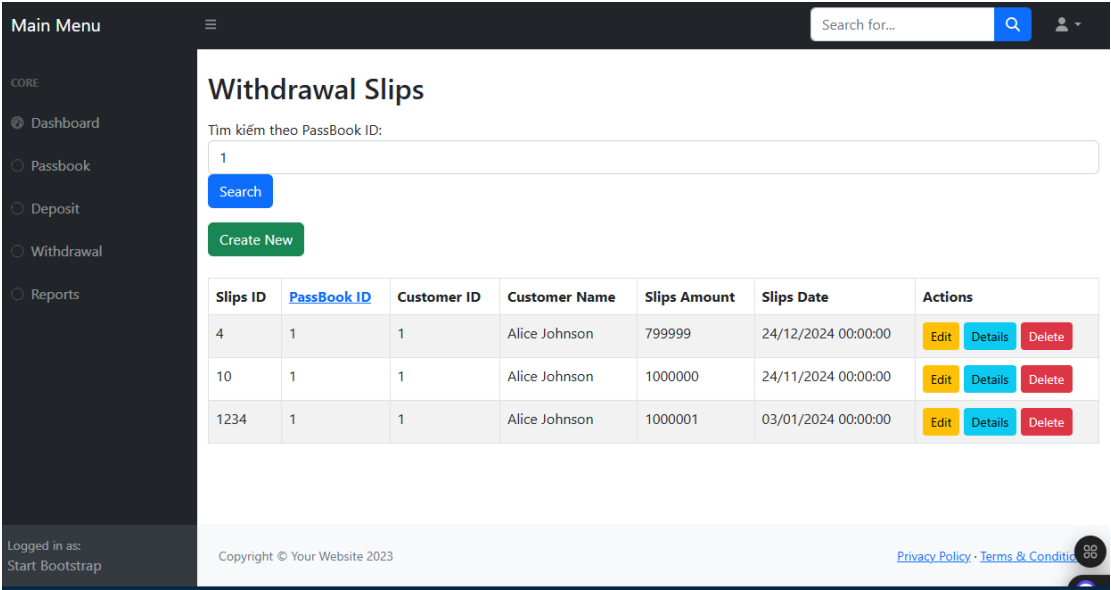
Search

Create New

Slips ID	PassBook ID	Customer ID	Customer Name	Slips Amount	Slips Date	Actions
4	1	1	Alice Johnson	799999	24/12/2024 00:00:00	<div>EditDetailsDelete</div>
10	1	1	Alice Johnson	1000000	24/11/2024 00:00:00	<div>EditDetailsDelete</div>
1234	1	1	Alice Johnson	1000001	03/01/2024 00:00:00	<div>EditDetailsDelete</div>
11	2	2	Bob Smith	2000000	11/11/2024 00:00:00	<div>EditDetailsDelete</div>
5	2	2	Bob Smith	246913578	24/12/2024 00:00:00	<div>EditDetailsDelete</div>
8	3	3	Charlie Brown	3999999	24/12/2024 00:00:00	<div>EditDetailsDelete</div>

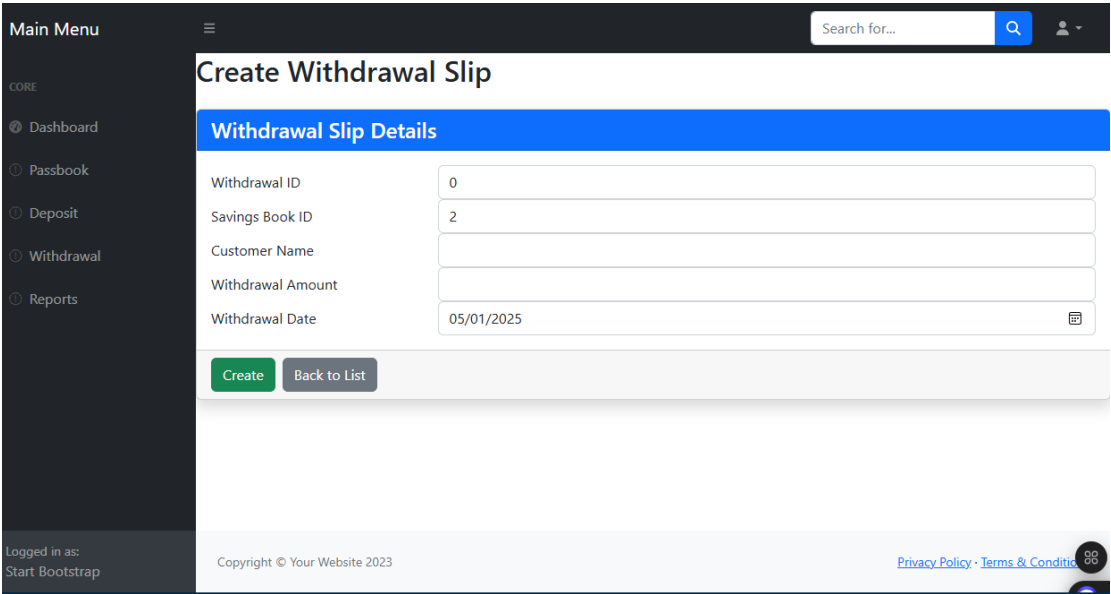
4.2.6: Giao diện phiếu rút tiền

4.2.7: Giao diện tìm kiếm phiếu rút tiền



4.2.7: Giao diện tìm kiếm phiếu rút tiền

4.2.8: Giao diện tạo phiếu rút tiền



4.2.8: Giao diện tạo phiếu rút tiền

4.2.9: Giao diện chỉnh sửa phiếu rút tiền

Main Menu

CORE

Dashboard

Passbook

Deposit

Withdrawal

Reports

Search for...

Q

Edit Withdrawal Slip

Edit Withdrawal Slip Details

SavingsBookID

1

user_id

123 Main St

WithdrawalAmount

799999

WithdrawalDate

24/12/2024 00:00:00

Save

Back to List

Logged in as:
Start Bootstrap

Copyright © Your Website 2023

[Privacy Policy](#) · [Terms & Conditions](#)

4.2.9: Giao diện chỉnh sửa phiếu rút tiền

4.2.10: Giao diện chi tiết phiếu rút tiền

Main Menu

CORE

Dashboard

Passbook

Deposit

Withdrawal

Reports

Search for...

Q

Withdrawal Slip Details

Withdrawal Slip Information

WithdrawalAmount:

799999

WithdrawalDate:

24/12/2024 00:00:00

SavingsBookID:

1

user_address:

123 Main St

Edit

Back to List

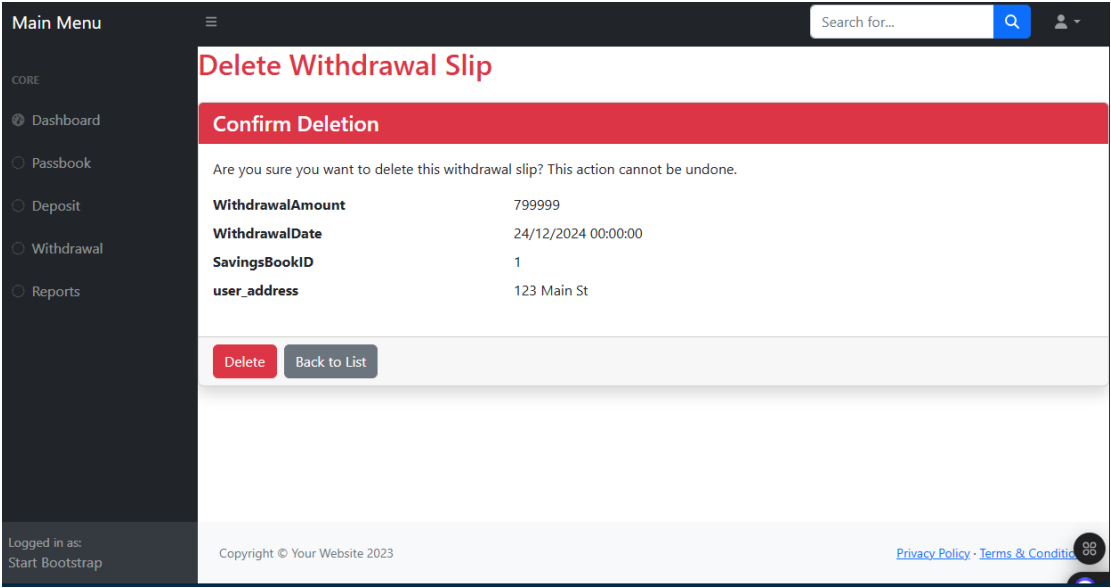
Logged in as:
Start Bootstrap

Copyright © Your Website 2023

[Privacy Policy](#) · [Terms & Conditions](#)

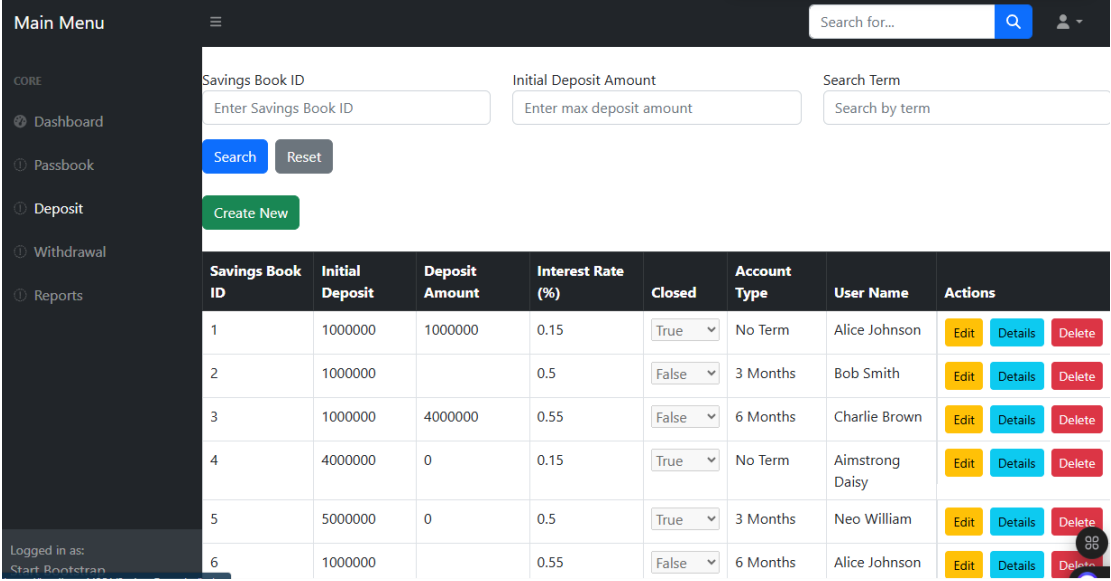
4.2.10: Giao diện chi tiết phiếu rút tiền

4.2.11: Giao diện xóa phiếu rút tiền



4.2.11: Giao diện chi tiết phiếu rút tiền

4.2.12 Giao diện Passbook



4.2.12: Giao diện Passbook

4.2.13. Giao diện chỉnh sửa Passbook

Main Menu

CORE

- Dashboard
- Passbook
- Deposit
- Withdrawal
- Reports

Logged in as:
Start Bootstrap

Search for...

Edit Passbook Details

user_id

123 Main St

OpeningDate

01/01/2024 00:00:00

InitialDepositAmount

1000000

DepositAmount

1000000

InterestRate

0.15

SavingsType

No Term

IsClosed

☐

Save

Back to List

Copyright © Your Website 2023

[Privacy Policy](#) · [Terms & Conditions](#)

4.2.13: Giao diện chỉnh sửa Passbook

4.2.14. Giao diện chi tiết Passbook

Main Menu

CORE

- Dashboard
- Passbook
- Deposit
- Withdrawal
- Reports

Logged in as:
Start Bootstrap

Search for...

Passbook Information

OpeningDate:

01/01/2024 00:00:00

InitialDepositAmount:

1000000

DepositAmount:

1000000

InterestRate:

0.15

IsClosed:

True

AccountTypeName:

No Term

user_address:

123 Main St

Edit

Back to List

Copyright © Your Website 2023

[Privacy Policy](#) · [Terms & Conditions](#)

4.2.14: Giao diện chỉnh sửa Passbook

4.2.15. Giao diện xóa Passbook

Main Menu

CORE

Dashboard

Passbook

Deposit

Withdrawal

Reports

Logged in as: Start Bootstrap

Copyright © Your Website 2023

[Privacy Policy](#) · [Terms & Conditions](#)

Search for...

Delete Passbook

Confirm Deletion

Are you sure you want to delete this passbook? This action cannot be undone.

OpeningDate01/01/2024 00:00:00

InitialDepositAmount1000000

DepositAmount1000000

InterestRate0.15

IsClosedTrue

AccountTypeNameNo Term

user_address123 Main St

Delete

Back to List

4.2.15: Giao diện chỉnh sửa Passbook

4.2.16 Giao diện chỉnh loại tiết kiệm

Admin

CORE

Dashboard

Passbook

Deposit

Withdrawal

Reports

localhost:44351/SavingsAccountTypes/Index

Search for...

Savings Account Types

Create New

AccountTypeName	InterestRate	Term	WithdrawalDays	MinimumDeposit	IsActive	AllowsAdditionalDeposits	Actions
No Term	0.15	0	15	100000	True	True	<div>EditDetailsDelete</div>
3 Months	0.5	3	90	100000	True	False	<div>EditDetailsDelete</div>
6 Months	0.55	6	180	100000	True	False	<div>EditDetailsDelete</div>

4.2.16: Giao diện chỉnh sửa Passbook

4.2.17 Giao diện chỉnh sửa thông tin loại tiết kiệm

The screenshot shows the 'Edit SavingsAccountType' form in the Admin interface. The form is titled 'Edit SavingsAccountType' and contains several input fields for editing account type details. The fields are: AccountTypeName (No Term), InterestRate (0.15), Term (0), WithdrawalDays (15), MinimumDeposit (100000), IsActive (True), and AllowsAdditionalDeposits (True). There is a 'Save' button and a 'Back to List' link at the bottom of the form. The left sidebar shows the 'Admin' menu with options like Dashboard, Passbook, Deposit, Withdrawal, and Reports.

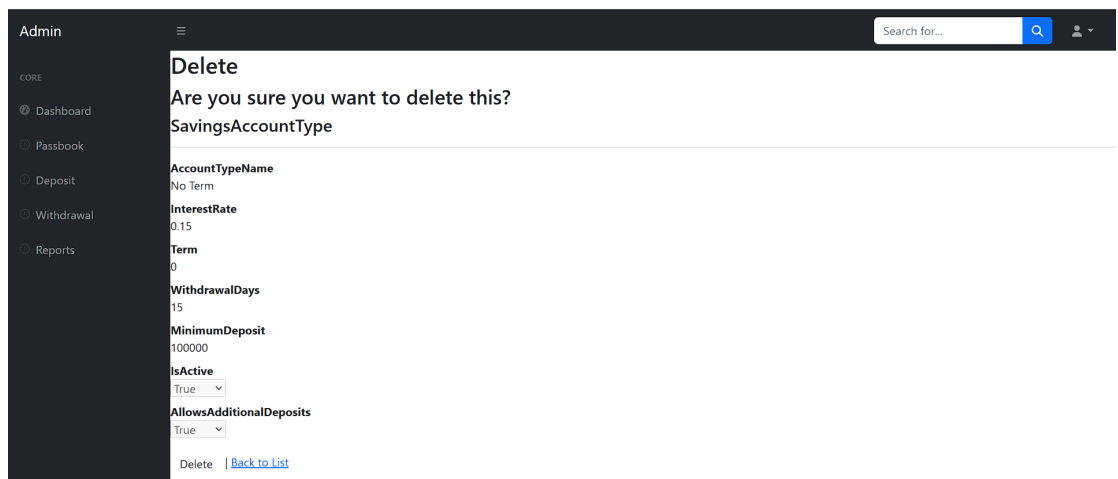
4.2.17: Giao diện chỉnh sửa thông tin loại tiết kiệm

4.2.18 Giao diện thông tin loại tiết kiệm

The screenshot shows the 'Details SavingsAccountType' form in the Admin interface. The form is titled 'Details SavingsAccountType' and displays the details of a specific account type. The details are: AccountTypeName (No Term), InterestRate (0.15), Term (0), WithdrawalDays (15), MinimumDeposit (100000), IsActive (True), and AllowsAdditionalDeposits (True). There is an 'Edit' link and a 'Back to List' link at the bottom of the form. The left sidebar shows the 'Admin' menu with options like Dashboard, Passbook, Deposit, Withdrawal, and Reports.

4.2.18: Giao diện thông tin loại tiết kiệm

4.2.19 Giao diện xóa loại tiết kiệm



4.2.19: Giao diện xóa loại tiết kiệm

4.2.20 Giao diện chỉnh sửa thông tin gửi thêm tiết kiệm

Main Menu

CORE

- Dashboard
- Passbook
- Deposit
- Withdrawal
- Reports

Logged in as: Start Bootstrap

Copyright © Your Website 2023

[Privacy Policy](#) · [Terms & Conditions](#)

Savings Deposit Details

Savings Deposit Information

DepositAmount: 1000000	InterestRate: 0.15
DepositDate: 24/12/2024 00:00:00	SavingsBookID: 1
user_name: Alice Johnson	

[Edit](#) [Back to List](#)

4.2.20: Giao diện chỉnh sửa thông tin gửi thêm tiết kiệm

4.2.21: Giao diện xóa thông tin gửi thêm tiết kiệm

Main Menu

CORE

- Dashboard
- Passbook
- Deposit
- Withdrawal
- Reports

Logged in as: Start Bootstrap

Copyright © Your Website 2023

[Privacy Policy](#) · [Terms & Conditions](#)

Delete Savings Deposit

Confirm Deletion

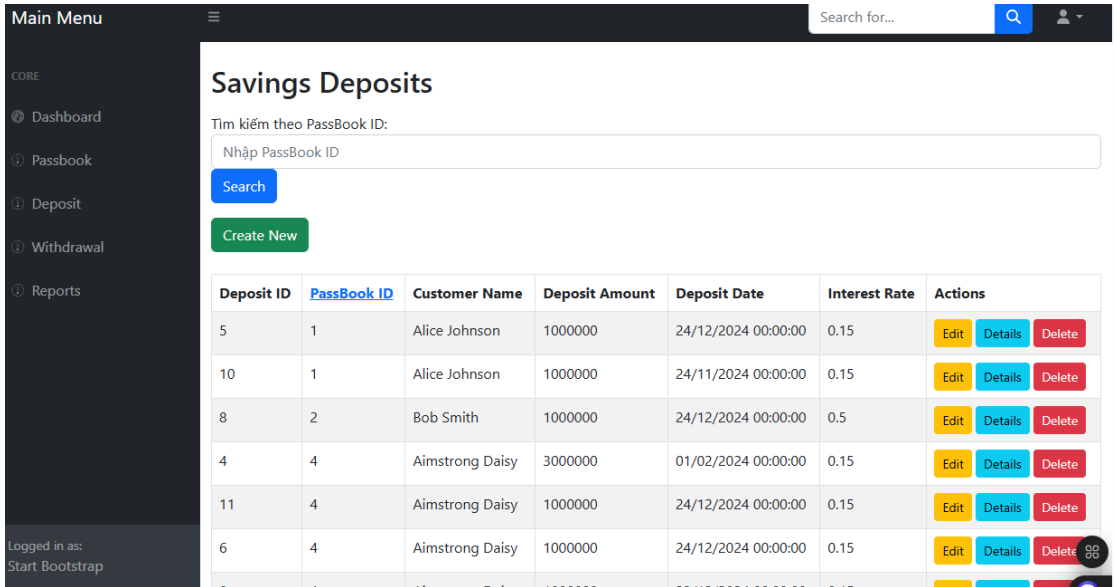
Are you sure you want to delete this savings deposit? This action cannot be undone.

DepositAmount	1000000
InterestRate	0.15
DepositDate	24/12/2024 00:00:00
SavingsBookID	1
user_name	Alice Johnson

[Delete](#) [Back to List](#)

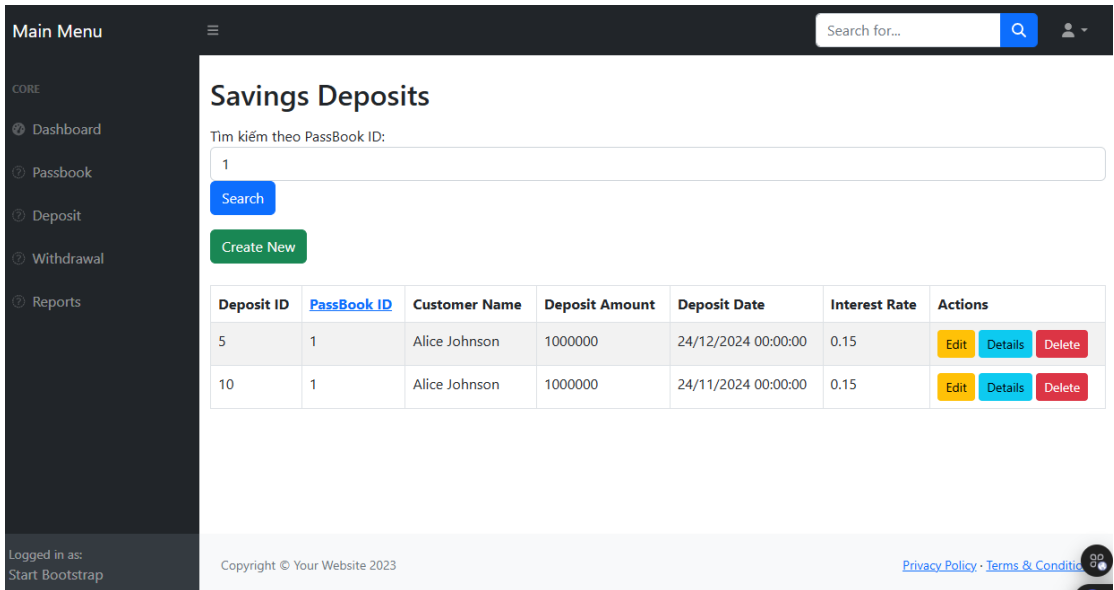
4.2.21: Giao diện xóa thông tin gửi thêm tiết kiệm

4.2.22: Giao diện gửi thêm tiết kiệm



4.2.22: Giao diện gửi thêm tiết kiệm

4.2.23: Giao diện tìm kiếm phiếu thêm tiết kiệm



4.2.23: Giao diện tìm kiếm phiếu thêm tiết kiệm

4.2.24: Giao diện tạo mới phiếu gửi thêm tiết kiệm

The screenshot shows a web application interface for creating a savings deposit. On the left is a dark sidebar with a 'Main Menu' and a 'CORE' section containing links to Dashboard, Passbook, Deposit, Withdrawal, and Reports. The top header has a search bar and a user profile icon. The main content area is titled 'Create Savings Deposit' and features a blue header for 'Savings Deposit Details'. Below this, there are input fields for Deposit ID (0), Savings Book ID (2), Customer Name, Deposit Amount, Interest Rate, and Deposit Date (05/01/2025). At the bottom of the form are 'Create' and 'Back to List' buttons. The footer includes a login status 'Logged in as: Start Bootstrap', a copyright notice 'Copyright © Your Website 2023', and links to 'Privacy Policy' and 'Terms & Conditions'.

4.2.24: Giao diện tạo mới phiếu gửi thêm tiết kiệm

4.2.25: Giao diện thông tin cá nhân

The screenshot displays the 'Account Settings' page. The sidebar and top header are identical to the previous page. The main content area is titled 'Account Settings' and contains a 'Personal Info' section. This section lists the following details: Full Name: Tai, Email: taipknguyen101@gmail.com, Date of Birth: Not provided, Phone Number: 767214193, and Address: HaNoi. An 'Edit' button is located at the bottom left of the personal info box. The footer is consistent with the other pages, showing the login status, copyright, and policy links.

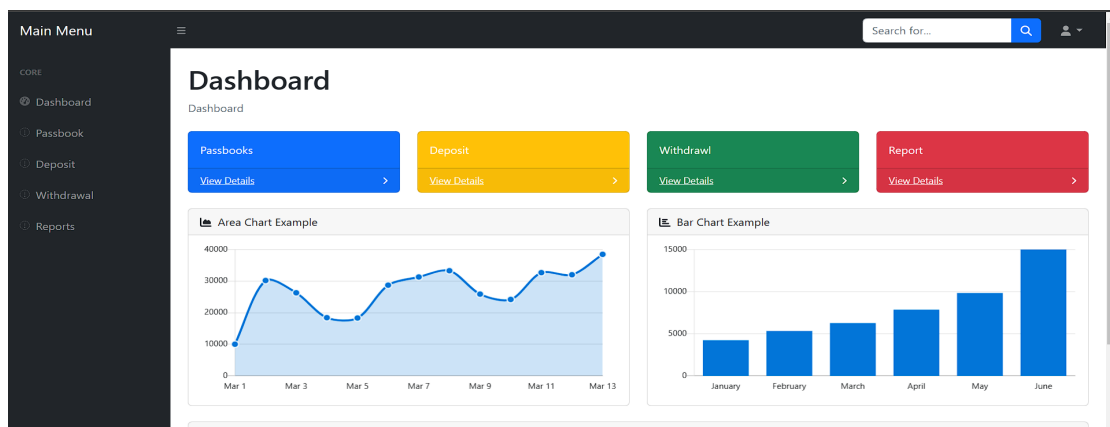
4.2.25: Giao diện thông tin cá nhân

4.2.26: Giao diện chỉnh sửa thông tin cá nhân

The screenshot shows a web application interface with a dark sidebar menu on the left containing 'Main Menu', 'CORE', and a list of links: 'Dashboard', 'Passbook', 'Deposit', 'Withdrawal', and 'Reports'. The main content area is titled 'Account Settings' and contains a 'Personal Info' section with fields for 'Full Name', 'Email', 'Date of Birth', 'Phone Number', and 'Address'. An 'Edit' button is located below this section. A modal window titled 'Edit Personal Information' is open, displaying the same fields with pre-filled data: Full Name: Tai, Email: taipknguyen101@gmail.com, Date of Birth: mm/dd/yyyy, Phone Number: 767214193, and Address: HaNoi. At the bottom of the modal are 'Close' and 'Save Changes' buttons. The footer of the page includes 'Logged in as: Start Bootstrap', 'Copyright © Your Website 2023', and links for 'Privacy Policy' and 'Terms & Conditions'.

4.2.26: Giao diện chỉnh sửa thông tin loại tiết kiệm

4.2.27: Giao diện chính chương trình



4.2.27: Giao diện chính chương trình

4.3 Test Case

4.3.1 Register - Người dùng đăng ký thành công

Tên Test Case	Người dùng đăng ký thành công.
Mô tả	Xác minh rằng khi người dùng cung cấp thông tin hợp lệ, tài khoản được thêm và chuyển hướng đến trang đăng nhập.
Điều kiện tiên quyết	Cơ sở dữ liệu không có tài khoản với email đã nhập.
Dữ liệu đầu vào	Email: test@example.comMật khẩu: Password123Tên người dùng: Test User
Kỳ vọng	- Tài khoản được thêm vào cơ sở dữ liệu.- Hệ thống chuyển hướng đến Login.

4.3.2. Register - Email đã tồn tại

Tên Test Case	Email đã tồn tại.
Mô tả	Xác minh rằng khi người dùng nhập email đã tồn tại, hệ thống trả về thông báo lỗi.
Điều kiện tiên quyết	Cơ sở dữ liệu đã có tài khoản với email test@example.com.
Dữ liệu đầu vào	Email: test@example.comMật khẩu: Password123Tên người dùng: Test User

Kỳ vọng	- Hệ thống không thêm tài khoản mới.- Hiện thị thông báo lỗi: Email already exists.
----------------	---

4.3.3. Login - Đăng nhập thành công

Tên Test Case	Đăng nhập thành công với quyền người dùng.
Mô tả	Xác minh rằng người dùng hợp lệ được chuyển hướng đến giao diện chính sau khi đăng nhập thành công.
Điều kiện tiên quyết	Cơ sở dữ liệu chứa tài khoản với email test@example.com và mật khẩu hợp lệ.
Dữ liệu đầu vào	Email: test@example.comMật khẩu: password
Kỳ vọng	- Hệ thống chuyển hướng đến Home/Index.

4.3.4 Login - Đăng nhập với quyền quản trị

Tên Test Case	Đăng nhập thành công với quyền quản trị viên.
Mô tả	Xác minh rằng quản trị viên hợp lệ được chuyển hướng đến giao diện quản lý sau khi đăng nhập thành công.
Điều kiện tiên quyết	Cơ sở dữ liệu chứa tài khoản admin với email admin@example.com.

Dữ liệu đầu vào	Email: admin@example.comMật khẩu: password
Kỳ vọng	- Hệ thống chuyển hướng đến SavingsAccountTypes/Index.

4.3.5 Login - Thông tin không hợp lệ

Tên Test Case	Đăng nhập thất bại do thông tin không hợp lệ.
Mô tả	Xác minh rằng người dùng nhập sai email hoặc mật khẩu sẽ nhận được thông báo lỗi.
Điều kiện tiên quyết	Cơ sở dữ liệu không chứa tài khoản tương ứng.
Dữ liệu đầu vào	Email: test@example.comMật khẩu: wrongpassword
Kỳ vọng	- Hiện thị thông báo lỗi: Invalid email or password.

4.3.6. Login - Thiếu thông tin bắt buộc

Tên Test Case	Thiếu thông tin đăng nhập.
Mô tả	Xác minh rằng khi người dùng không nhập email hoặc mật khẩu, hệ thống hiển thị thông báo lỗi.
Điều kiện tiên quyết	Không có điều kiện tiên quyết.

Dữ liệu đầu vào	Email: ""Mật khẩu: ""
Kỳ vọng	- Hiện thị thông báo lỗi: Email and password are required.

KẾT LUẬN VÀ KIẾN NGHỊ

1. Kết quả đạt được

Dự án Quản Lý Sổ Tiết Kiệm đã hoàn thành với nhiều kết quả tích cực, đáp ứng tốt các yêu cầu và mục tiêu đề ra. Hệ thống đã tích hợp đầy đủ các chức năng cốt lõi như mở sổ tiết kiệm, gửi tiền, rút tiền, tra cứu thông tin, lập báo cáo và thay đổi quy định. Quy trình quản lý đã được tự động hóa, giúp giảm thiểu sai sót thủ công, tăng hiệu quả hoạt động, đồng thời đảm bảo tính chính xác và bảo mật cho dữ liệu khách hàng. Giao diện hệ thống được thiết kế trực quan và dễ sử dụng, mang lại trải nghiệm tốt cho người dùng. Ngoài ra, hệ thống còn hỗ trợ lập báo cáo chi tiết, giúp quản trị viên dễ dàng phân tích và quản lý hoạt động tài chính.

2. Tồn tại

Tuy nhiên, hệ thống vẫn tồn tại một số hạn chế cần được cải thiện. Hiện tại, phạm vi chức năng chủ yếu tập trung vào quản lý sổ tiết kiệm, chưa tích hợp các dịch vụ tài chính khác như quản lý khoản vay hoặc thanh toán trực tuyến. Hệ thống cũng thiếu tính năng đa nền tảng, chưa có ứng dụng di động riêng biệt, gây khó khăn cho người dùng trên thiết bị di động. Hiệu năng của hệ thống chưa được kiểm thử kỹ với khối lượng dữ liệu lớn, và khả năng tích hợp với các dịch vụ bên ngoài như ngân hàng trực tuyến hoặc ví điện tử vẫn còn hạn chế. Việc kiểm tra bằng manual test vẫn còn trên dự kiến và chưa triển khai do sự bất đồng bộ về phiên bản của .Net và NUnit Test

3. Hướng phát triển

Để phát triển hệ thống trong tương lai, cần mở rộng phạm vi chức năng bằng cách tích hợp các dịch vụ tài chính bổ sung và xây dựng ứng dụng di động để tăng tính linh hoạt trong sử dụng. Hiệu năng hệ thống cần được nâng

cao bằng cách tối ưu hóa truy vấn và cải thiện khả năng xử lý dữ liệu lớn. Đồng thời, việc xây dựng các API mở sẽ giúp hệ thống dễ dàng tích hợp với các dịch vụ bên ngoài, mở rộng khả năng hợp tác. Ngoài ra, nên cải tiến chức năng lập báo cáo bằng cách tích hợp các công cụ phân tích dữ liệu trực quan, dự đoán xu hướng tài chính thông qua AI và hỗ trợ giao diện đa ngôn ngữ để đáp ứng nhu cầu của nhiều đối tượng người dùng.

TÀI LIỆU THAM KHẢO

1. Microsoft, *SQL Server Documentation and Resources*, [Online].
Available: <https://learn.microsoft.com/en-us/sql/>. [Accessed: Dec. 22, 2024].
2. J. Galloway, B. Wilson, K. S. Allen, and D. Matson, *Professional ASP.NET MVC 5*. Wrox, 2014.
3. C. Coronel and S. Morris, *Database Systems: Design, Implementation, and Management*, 14th ed. Cengage Learning, 2020.
4. E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1994.
5. Microsoft Patterns & Practices, *Building Secure ASP.NET Applications*. [Online]. Available: [https://learn.microsoft.com/en-us/previous-versions/msp-n-p/ff649858\(v=pandp.10\)](https://learn.microsoft.com/en-us/previous-versions/msp-n-p/ff649858(v=pandp.10)). [Accessed: Dec. 22, 2024].
6. M. Winand, *SQL Performance Explained*. Self-published, 2012.
[Online]. Available: <https://use-the-index-luke.com/>. [Accessed: Dec. 22, 2024].
7. M. Richards and N. Ford, *Modern Software Architecture: Domain-Driven Design, Event-Driven Architecture, and Microservices*. O'Reilly Media, 2020.

8. Microsoft, *Entity Framework Core Documentation*, [Online]. Available:
<https://learn.microsoft.com/en-us/ef/core/>. [Accessed: Dec. 22, 2024].
9. International Organization for Standardization, *ISO/IEC 27001: Information Security Management*, [Online]. Available:
<https://www.iso.org/isoiec-27001-information-security.html>. [Accessed:
Dec. 22, 2024].
10. S. Hoberman, *Data Modeling Made Simple: A Practical Guide for Business and IT Professionals*. Technics Publications, 2015.