



ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Nhập môn công nghệ phần mềm SE104.L23

**Đề tài**

**Quản lý Sổ tiết kiệm**

ĐỒ ÁN MÔN: NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

Sinh viên thực hiện:		
STT	Họ tên	MSSV
1	Nguyễn Lộc Linh	19521754
2	Trần Duy Quang	19522102
3	Nguyễn Trí Tài	19522572



Thành phố Hồ Chí Minh, Tháng 7 năm 2021

## Mục lục

<b>I.</b>	<b>Giới thiệu.....</b>	<b>4</b>
1.	Chủ đề.....	4
2.	Mục đích tài liệu.....	4
<b>II.</b>	<b>Nội dung .....</b>	<b>5</b>
1.	Xác định và đặc tả yêu cầu phần mềm: .....	5
1.1	Khảo sát hiện trạng:.....	5
1.2	Phân tích & xác định yêu cầu.....	9
2.	Kế hoạch, quy trình, công cụ: .....	22
2.1.	Kế hoạch phát triển: .....	22
2.2.	Quy trình phát triển đồ án:.....	22
2.3.	Công cụ (CASE Tools) hỗ trợ trong suốt quá trình quản lý công việc.....	26
3.	Thiết kế mô hình dữ liệu: .....	28
3.1.	Biểu mẫu 1 – Quy định 1:.....	28
3.2.	Biểu mẫu 2 – Quy định 2:.....	29
3.3.	Biểu mẫu 3 – Quy định 3:.....	30
3.4.	Biểu mẫu 4: .....	32
3.5.	Biểu mẫu 5: .....	33
3.6.	Biểu mẫu 6 – Quy định 6:.....	35
4.	Thiết kế giao diện:.....	37
4.1.	Biểu mẫu 1 – Quy định 1:.....	37
4.3.	Biểu mẫu 3 – Quy định 3:.....	41
4.4.	Biểu mẫu 4: .....	43
4.5.	Biểu mẫu 5: .....	44

4.6.	Biểu mẫu 6 – Quy định 6:.....	46
4.7.	Thiết kế giao diện quản lý.....	47
5.	Thiết kế xử lý:.....	48
5.1.	Biểu mẫu 1 – Quy định 1:.....	48
5.2.	Biểu mẫu 2 – Quy định 2:.....	53
5.3.	Biểu mẫu 3 – Quy định 3:.....	59
5.4.	Biểu mẫu 4 – Quy định 4:.....	63
5.5.	Biểu mẫu 5: .....	66
5.6.	Biểu mẫu 6 – Quy định 6:.....	73
6.	Thiết kế phần mềm:.....	76
6.2.	Application Architecture:.....	76
6.2.	System Architecture:.....	82
7.	Coding Convention:.....	83
7.1.	Lập trình hướng đối tượng .....	83
7.2.	Chuẩn viết mã .....	83
<b>III.</b>	<b>Phần kết luận.....</b>	<b>87</b>
1.	Đánh giá đồ án.....	87
1.1.	Nhận xét.....	87
1.2.	Hướng phát triển.....	87
2.	Phân công công việc .....	88
3.	Tài liệu tham khảo.....	89

## **I. Giới thiệu**

### **1. Chủ đề**

Nhóm thống nhất chọn giải quyết bài toán quản lý sổ tiết kiệm.

Phần mềm quản lý sổ tiết kiệm cung cấp các chức năng giúp hỗ trợ các nhân viên kế toán ngân hàng thực thi các nhiệm vụ liên quan đến quản lý tài khoản ngân hàng, lưu trữ và tính toán các vấn đề quan trọng, thuận tiện trong việc kiểm tra thông tin sổ tiết kiệm của khách hàng, vấn tin số dư, đồng thời có thể rút và gửi tiền dễ dàng.

### **2. Mục đích tài liệu**

- Đối với nhóm phát triển và kiểm thử:

Nắm được quy trình phát triển một phần mềm. Học được các kỹ năng phát triển và quản lý điều hành các dự án phần mềm. Có khả năng phân tích thiết kế, chế tác, triển khai và bảo trì các hệ thống phần mềm cũng như nghiên cứu, phát triển công cụ và ứng dụng liên quan. Hiểu được các yêu cầu người dùng và phát triển công cụ thực hiện yêu cầu đó.

- Đối với người dùng:

Biết được cách hoạt động của phần mềm, các chức năng và công cụ có sẵn trong phần mềm để sử dụng phần mềm hiệu quả hơn.

## II. Nội dung

### 1. Xác định và đặc tả yêu cầu phần mềm:

#### 1.1 Khảo sát hiện trạng:

##### 1.1.1 Hiện trạng nghiệp vụ:

###### a. Danh sách nghiệp vụ:

STT	Tên yêu cầu	Biểu mẫu	Qui định	Ghi chú
1	Mở sổ tiết kiệm	BM1	QĐ1	
2	Lập phiếu gửi tiền	BM2	QĐ2	
3	Lập phiếu rút tiền	BM3	QĐ3	
4	Tra cứu sổ	BM4		
5	Lập báo cáo tháng	BM5		
6	Thay đổi qui định		QĐ6	

###### b. Danh sách biểu mẫu quy định:

- Biểu mẫu 1 và quy định 1

<b>BM1:</b>	<b>Sổ Tiết Kiệm</b>
Mã số: .....	Loại tiết kiệm: .....
Khách hàng: .....	CMND: .....
Địa chỉ: .....	Ngày mở sổ: .....
Số tiền gửi: .....	

**QĐ1: Có 3 loại tiết kiệm (không kỳ hạn, 3 tháng, 6 tháng). Số tiền gửi (ban đầu) tối thiểu là 1.000.000đ**

- Biểu mẫu 2 và quy định 2

<b>BM2:</b>	<b>Phiếu Gửi Tiền</b>
Mã số: .....	Khách hàng: .....
Ngày gửi: .....	Số tiền gửi: .....

**QĐ2: Chỉ nhận gửi thêm tiền khi đến kỳ hạn tính lãi suất của các loại tiết kiệm tương ứng.  
Số tiền gửi thêm tối thiểu là 100.000đ**

- Biểu mẫu 3 và quy định 3

<b>BM3:</b>	<b>Phiếu Rút Tiền</b>	
Mã số: .....	Khách hàng: .....	
Ngày rút: .....	Số tiền rút: .....	

**QĐ3: Lãi suất là 0.5% đối với loại không kỳ hạn, 5% với kỳ hạn 3 tháng và 5.5% với kỳ hạn 6 tháng.**

**Tiền lãi 1 năm = số dư \* lãi suất của loại tiết kiệm tương ứng.**

**Loại tiết kiệm có kỳ hạn chỉ được rút khi quá kỳ hạn và phải rút hết toàn bộ, khi này tiền lãi được tính với mức lãi suất của loại không kỳ hạn.**

**Loại tiết kiệm không kỳ hạn được rút khi gửi trên 15 ngày và có thể rút số tiền <= số dư hiện có.**

**Sổ sau khi rút hết tiền sẽ tự động đóng.**

- Biểu mẫu 4

<b>BM4:</b>	<b>Danh Sách Sổ Tiết Kiệm</b>			
<b>STT</b>	<b>Mã Số</b>	<b>Loại Tiết Kiệm</b>	<b>Khách Hàng</b>	<b>Số Dư</b>
1				
2				

- Biểu mẫu 5

• Biểu mẫu 5.1

<b>BM5.1</b>	<b>Báo Cáo Doanh Số Hoạt Động Ngày</b>			
Ngày:.....				
<b>STT</b>	<b>Loại Tiết Kiệm</b>	<b>Tổng Thu</b>	<b>Tổng Chi</b>	<b>Chênh Lệch</b>
1				
2				

• Biểu mẫu 5.2

<b>BM5.2</b>	<b>Báo Cáo Mở/Đóng Sổ Tháng</b>			
Loại tiết kiệm: .....			Tháng:.....	
<b>STT</b>	<b>Ngày</b>	<b>Số Mở</b>	<b>Số Đóng</b>	<b>Chênh Lệch</b>
1				
2				

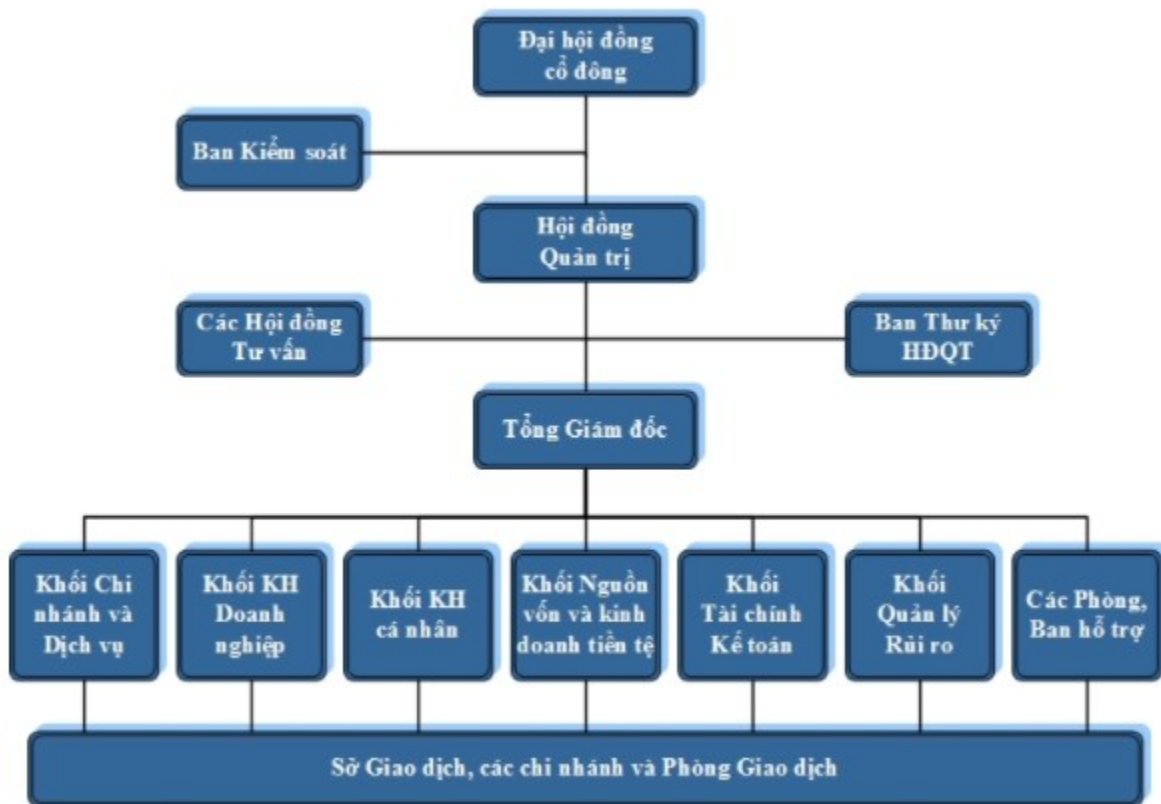
- Quy định 6

**QĐ6: Người dùng có thể thay đổi các qui định như sau:**

+ QĐ1: Thay đổi số lượng các loại kỳ hạn, tiền gửi tối thiểu.

+ QĐ3: Thay đổi thời gian gửi tối thiểu và lãi suất các loại kỳ hạn.

### 1.1.2 Hiện trạng tổ chức



### 1.1.3 Hiện trạng tin học

- Hệ thống phần cứng:  
CPU: Intel Pentium III 800 MHz  
RAM: 2GB RAM
- Hệ thống phần mềm  
OS: Windows 7 minimum  
Microsoft SQL Server 2013 or above



## 1.2 Phân tích & xác định yêu cầu

### 1.2.1. Chức năng mở sổ tiết kiệm

#### Thông tin chung chức năng

<b>Tên chức năng</b>	Mở sổ tiết kiệm
<b>Mô tả</b>	Mở tài khoản tiết kiệm
<b>Tác nhân</b>	Nhân viên tiếp nhận yêu cầu từ người dùng
<b>Điều kiện trước</b>	Nhân viên đăng nhập.
<b>Điều kiện sau</b>	QĐ1: Có 3 loại tiết kiệm (không kỳ hạn, 3 tháng, 6 tháng). Số tiền gửi (ban đầu) tối thiểu là 1.000.000đ
<b>Ngoại lệ</b>	
<b>Các yêu cầu đặc biệt</b>	

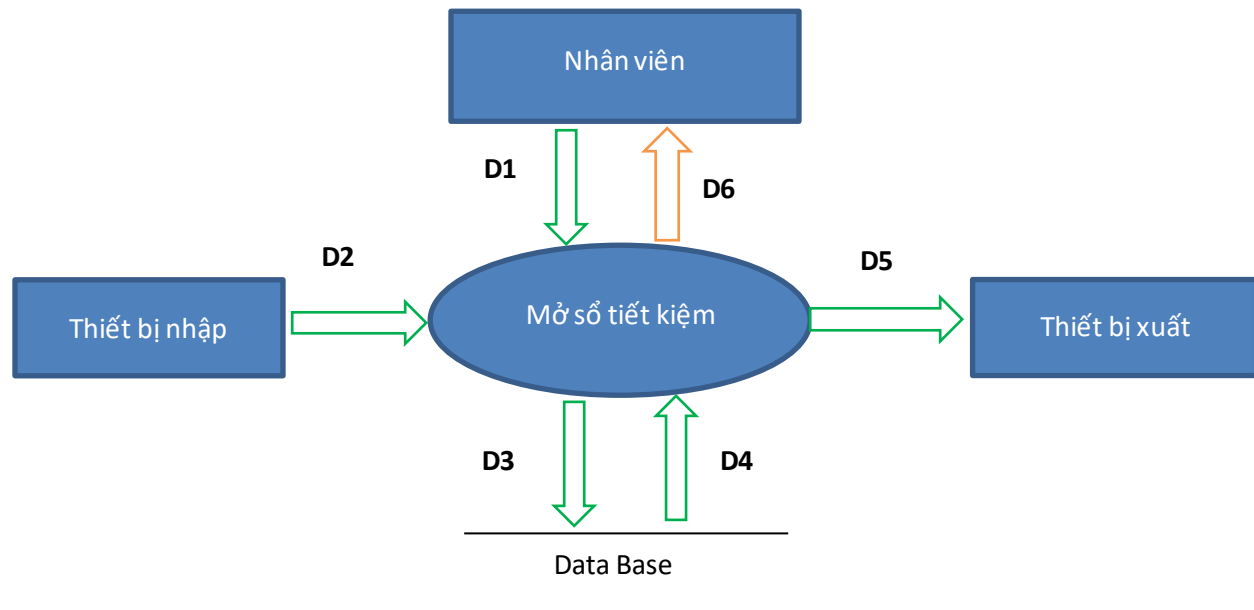
#### Mô tả dòng sự kiện chính

Hành động của tác nhân	Phản ứng của hệ thống
Bấm vào nút “Đăng kí mới”	Hiển thị ra biểu mẫu 1 (BM1) cho người dùng nhập
Chọn nút “Đăng kí”	Kiểm tra thông tin có phù hợp với QĐ1, ngày mở sổ trùng ngày hệ thống. Không nhập mã số
Nếu thỏa điều kiện trên	Lưu thông tin vào bộ nhớ phụ và xuất ra “Đăng kí thành công”
Nếu không thỏa điều kiện hay vì lí do hệ thống trục trặc gì đó	Xuất ra “Đăng kí không thành công”

### Mô tả dòng sự kiện phụ

Hành động của tác nhân	Phản ứng của hệ thống
Bấm vào nút “Hủy”	Hủy bỏ việc đăng kí mở sổ tiết kiệm và quay lại màn hình chính

### DFD (Data Flow Diagram)



### Mô tả các luồng dữ liệu:

D1: Thông tin về khách hàng như mã khách hàng, tên khách hàng, CMND, địa chỉ, loại tiết kiệm, ngày mở sổ, số tiền gửi.

D2: Nhập thông tin

D3, D4: D1

D5: Xuất đăng kí thành công hay không thành công

D6: không có

### Ghi chú

Cho phép hủy, cập nhật lại

### 1.2.2. Chức năng lập phiếu gửi tiền

#### Thông tin chung chức năng

<b>Tên chức năng</b>	Lập phiếu gửi tiền
<b>Mô tả</b>	Cho người dùng đưa tiền vào sổ tiết kiệm
<b>Tác nhân</b>	Nhân viên tiếp nhập yêu cầu của khách hàng
<b>Điều kiện trước</b>	Nhân viên đăng nhập.
<b>Điều kiện sau</b>	QĐ2: Chỉ nhận gửi thêm tiền khi đến kỳ hạn tính lãi suất của các loại tiết kiệm tương ứng. Số tiền gửi thêm tối thiểu là 100.000đ
<b>Ngoại lệ</b>	
<b>Các yêu cầu đặc biệt</b>	

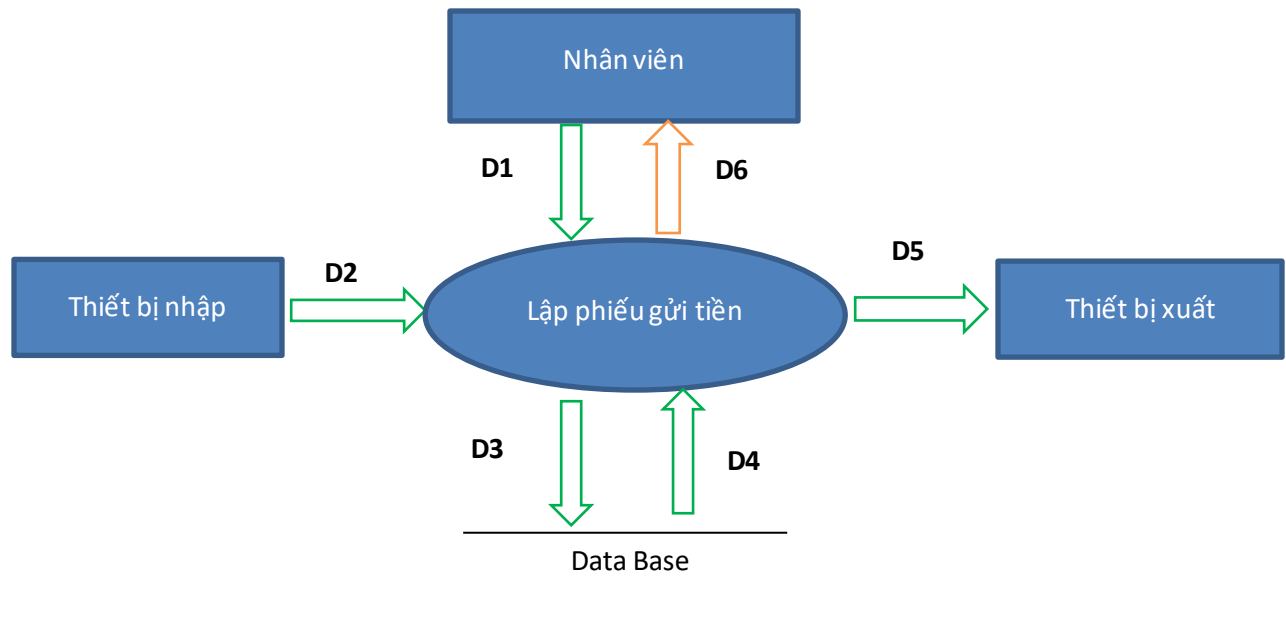
#### Mô tả dòng sự kiện chính

Hành động của tác nhân	Phản ứng của hệ thống
Bấm vào nút “Gửi tiền”	Hiển thị ra biểu mẫu 2 (BM2) cho người dùng nhập
Chọn nút “Xác nhận”	Kiểm tra thông tin có phù hợp với QĐ2.
Nếu thỏa điều kiện trên	Lưu thông tin vào bộ nhớ phụ và in ra hóa đơn gửi tiền đồng thời thông báo gửi tiền thành công.
Nếu không thỏa điều kiện hay vì lí do hệ thống trục trặc gì đó	Xuất ra “gửi tiền không thành công” và cho phép thực hiện lại.

#### Mô tả dòng sự kiện phụ

Hành động của tác nhân	Phản ứng của hệ thống
Bấm vào nút “Hủy”	Hủy bỏ việc gửi tiền vào sổ tiết kiệm và quay lại màn hình chính.

## DFD (Data Flow Diagram)



### Mô tả các luồng dữ liệu:

D1: Thông tin về mã số phiếu, khách hàng gửi, ngày gửi, số tiền gửi.

D2: Nhập thông tin

D3: D1

D4: D1, số dư sổ tiết kiệm của khách hàng.

D5: Xuất nhập thành công và in hóa đơn hoặc xuất ra không thành công

D6: không có

### Ghi chú

Cho phép hủy, cập nhật lại

### 1.2.3. Chức năng lập phiếu rút tiền

#### Thông tin chung chức năng

<b>Tên chức năng</b>	Lập phiếu rút tiền
<b>Mô tả</b>	Khách hàng
<b>Tác nhân</b>	Nhân viên tiếp nhận (Nhân viên ngân hàng)
<b>Điều kiện trước</b>	<p>Người dùng phải đăng nhập thành công tài khoản ngân hàng (Kiểm tra thông tin khách hàng có trong hệ thống hay không)</p> <p>Chỉ được rút sau khi mở sổ ít nhất 15 ngày. Loại tiết kiệm có kỳ hạn chỉ được rút khi quá kỳ hạn và phải rút hết toàn bộ. Tiền lãi = số lần đáo hạn * lãi suất * kỳ hạn (0.5% với kỳ hạn 3 tháng, 0.55% với kỳ hạn 6 tháng). Loại tiết kiệm không kỳ hạn có thể rút với số tiền &lt;= số dư hiện có. Tiền lãi chỉ tính khi gửi ít nhất 1 tháng với lãi suất 0.15%. Sổ sau khi rút hết tiền sẽ tự động đóng.</p>
<b>Điều kiện sau</b>	Không có
<b>Ngoại lệ</b>	
<b>Các yêu cầu đặc biệt</b>	

#### Mô tả dòng sự kiện chính

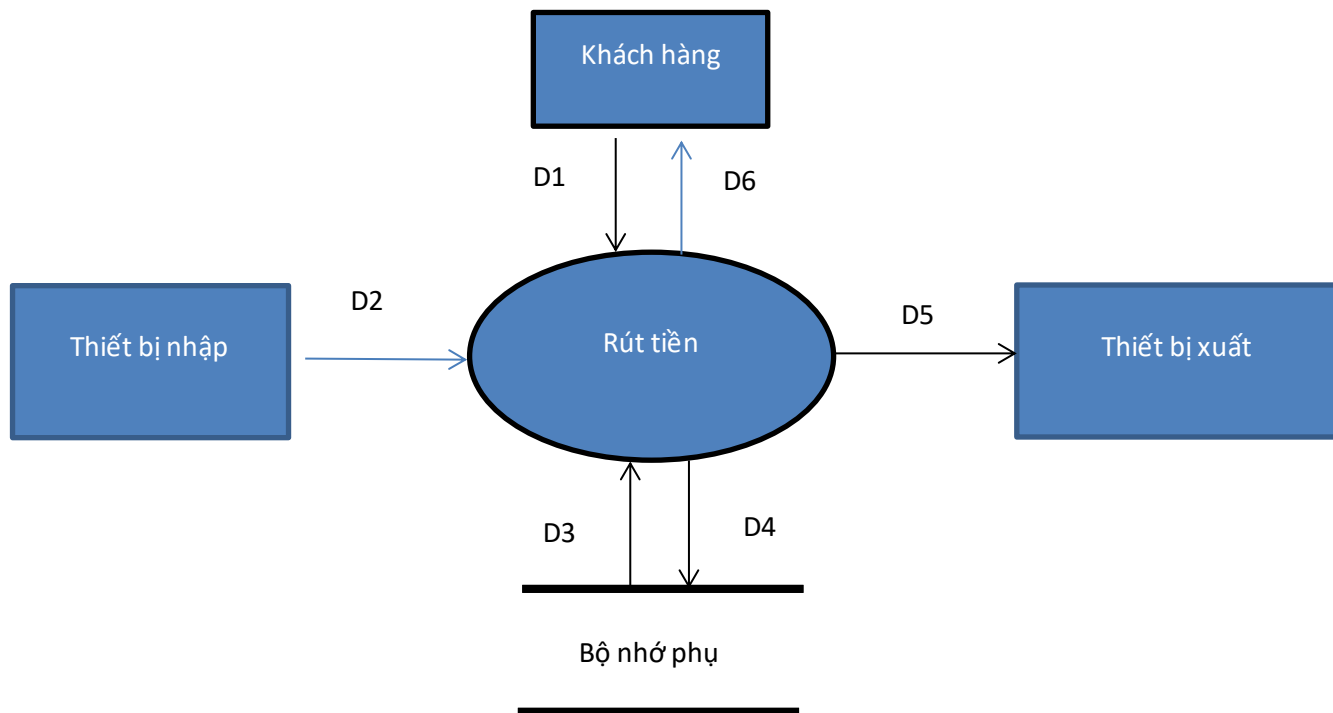
<b>Hành động của tác nhân</b>	<b>Phản ứng của hệ thống</b>
Bấm vào “Lập phiếu rút tiền”	-Hiển thị biểu mẫu 3 (BM3) cho người dùng nhập bao gồm: + Mã số: + Khách hàng: + Ngày rút + Số tiền rút
Chọn “Xác nhận”	-Kiểm tra thông tin có tồn tại trong hệ thống hay không. Kiểm tra có thỏa QĐ3. Kiểm tra số tiền hiện tại. -Nếu thỏa yêu cầu: + Hiển thị “Giao dịch thành công” -Nếu không thỏa yêu cầu: + Hiển thị “Giao dịch thất bại”

	-Hiển thị: “Bạn có muốn in ra thông tin tài khoản không?”
Chọn “Có”	-In ra thông tin khách hàng (Mã số, Khách hàng, Ngày rút, Số tiền rút, Số tiền còn lại)
Chọn “Không”	-Hiển thị “Giao dịch kết thúc”

## Mô tả dòng sự kiện phụ

Không

## DFD (Data Flow Diagram)



**Mô tả các luồng dữ liệu:**

D1: Thông tin về Phiếu rút tiền (Mã số, Khách hàng, Ngày rút, Số tiền rút).

D2: Không có

D3: Danh sách các Phiếu rút tiền (Mã số, Khách hàng, Ngày rút, Số tiền rút), Loại tiết kiệm, Số dư, Ngày mở sổ

D4: D1 + Số tiền còn lại

D5: D4

D6: Không có

**Thuật toán:**

B1: Nhận D1 từ người dùng

B2: Kết nối cơ sở dữ liệu

B3: Đọc D3 từ bộ nhớ phụ

B4: Kiểm tra Phiếu rút tiền (D1) có thuộc danh sách các Phiếu rút tiền (D3) hay không?

B5: Tính Số tiền còn lại

B6: Kiểm tra Số tiền hiện tại  $\geq$  Số tiền rút?

B7: Nếu không thỏa một trong các điều kiện trên thì -> B11

B8: Tính Số tiền còn lại

B9: Lưu D4 xuống bộ nhớ phụ

B10: Xuất D5 ra máy in

B11: Đóng kết nối cơ sở dữ liệu

B12: Kết thúc

Ghi chú: không

#### 1.2.4. Chức năng tra cứu sổ

##### Thông tin chung chức năng

<b>Tên chức năng</b>	Tra cứu sổ
<b>Mô tả</b>	Tra cứu sổ tiết kiệm của khách hàng
<b>Tác nhân</b>	Nhân viên tiếp nhận (Nhân viên ngân hàng)
<b>Điều kiện trước</b>	Người dùng phải đăng nhập thành công tài khoản ngân hàng (Kiểm tra thông tin khách hàng có trong hệ thống hay không)
<b>Điều kiện sau</b>	
<b>Ngoại lệ</b>	
<b>Các yêu cầu đặc biệt</b>	

##### Mô tả dòng sự kiện chính

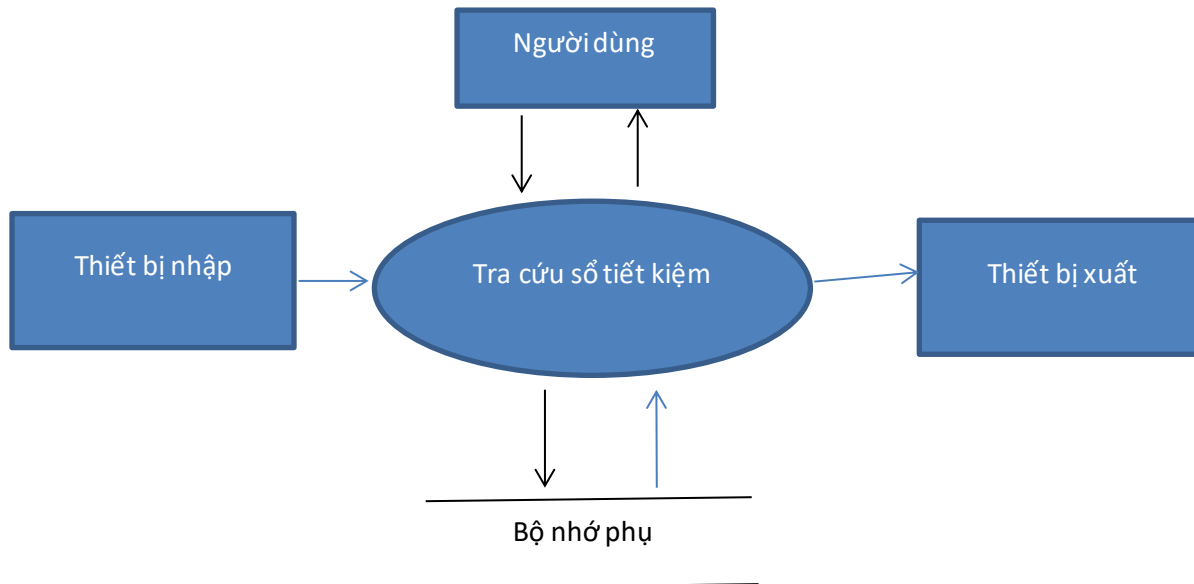
<i>Hành động của tác nhân</i>	<i>Phản ứng của hệ thống</i>
Bấm vào “Tra cứu sổ tiết kiệm”	-Hiển thị biểu mẫu 4 (BM4) cho người dùng nhập bao gồm:
Chọn “Xác nhận”	-Kiểm tra thông tin có tồn tại trong hệ thống hay không. -Nếu tồn tại: + Hiển thị danh sách thông tin sổ tiết kiệm thỏa mãn -Nếu không tồn tại: + Hiển thị “Không tồn tại”

##### Mô tả dòng sự kiện phụ

Không



## DFD (Data Flow Diagram)



### Mô tả luồng dữ liệu:

D1: Thông tin về Sổ tiết kiệm (Mã sổ, Loại tiết kiệm, Khách hàng, Số dư)

D2: Không có

D3: D1 thỏa mãn tiêu chuẩn tra cứu

D4: Không có

D5: D3

D6: D5

### Thuật toán:

B1: Nhận D1 từ người dùng

B2: Kết nối cơ sở dữ liệu

B3: Đọc D3 từ bộ nhớ phụ

B4: Xuất D5 ra máy in

B5: Trả D6 cho người dùng

B6: Đóng kết nối cơ sở dữ liệu

B7: Kết thúc

Ghi chú: không

### 1.2.5. Chức năng lập báo cáo tháng

#### Thông tin chung chức năng

<b>Tên chức năng</b>	Lập báo cáo tháng
<b>Mô tả</b>	Cung cấp cho nhân viên : Ngày tháng đầu cuối , sổ mở , sổ đóng , chênh lệch
<b>Tác nhân</b>	Nhân viên
<b>Điều kiện trước</b>	Nhân viên đăng nhập thành công vào hệ thống
<b>Điều kiện sau</b>	Không
<b>Ngoại lệ</b>	<i>Không</i>
<b>Các yêu cầu đặc biệt</b>	<i>Không</i>

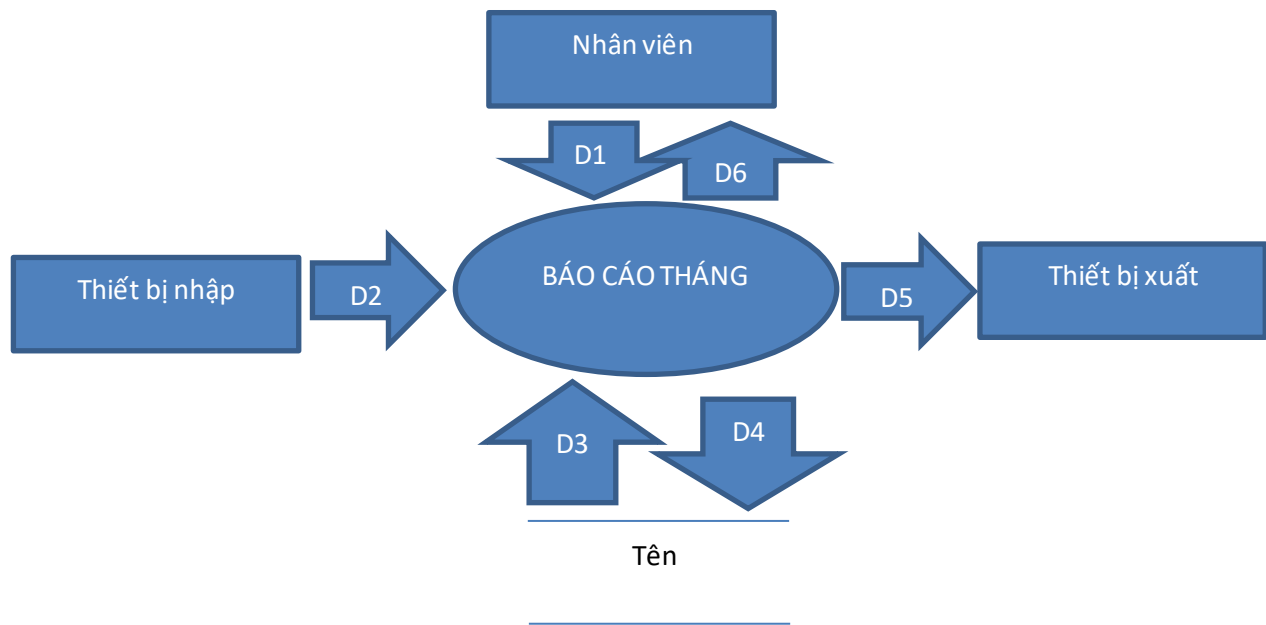
#### Mô tả dòng sự kiện chính

<b>Hành động của tác nhân</b>	<b>Phản ứng của hệ thống</b>
Tra cứu báo cáo tháng	Tìm và xuất ra thông tin về ngày tháng đầu cuối, sổ mở, sổ đóng, chênh lệch.

#### Mô tả dòng sự kiện phụ

Không

## DFD (Data Flow Diagram)



D1: Nhân viên nhập tháng cần báo cáo

D2 : Không có

D3 : Ngày tháng đầu cuối , sổ mở , sổ đóng , chênh lệch

D4 : D1

D5 : D4 + D3

D6 : Không có

### Ghi chú

Select từ bảng Báo Cáo Mở/Đóng Sổ Tháng

Số đóng – số mở = Chênh lệch Hiển thị chênh lệch dương hoặc âm

### 1.2.6. Chức năng thay đổi quy định

#### Thông tin chung chức năng

<b>Tên chức năng</b>	Thay đổi qui định
<b>Mô tả</b>	Thay đổi các qui định đã lập ra.
<b>Tác nhân</b>	Giám đốc , quản lí
<b>Điều kiện trước</b>	Giám đốc , quản lí đăng nhập thành công vào hệ thống
<b>Điều kiện sau</b>	Không
<b>Ngoại lệ</b>	<i>Không</i>
<b>Các yêu cầu đặc biệt</b>	<i>Không</i>

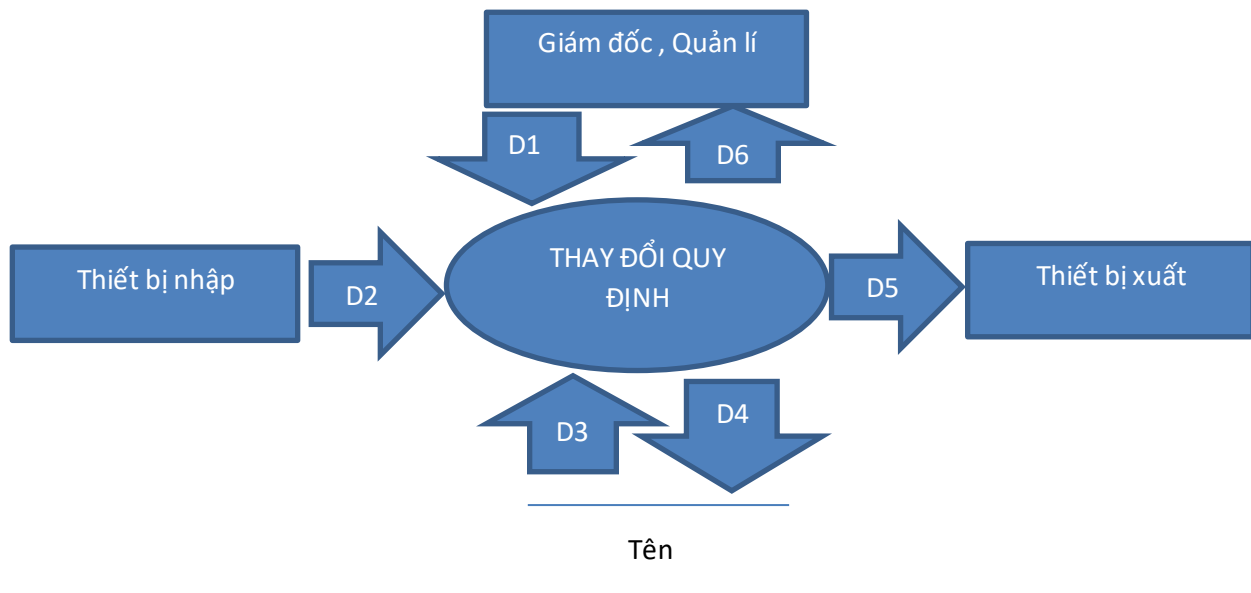
#### Mô tả dòng sự kiện chính

<b>Hành động của tác nhân</b>	<b>Phản ứng của hệ thống</b>
Thêm xoá sửa các quy định	Cho phép nhập vào và in ra màn hình hệ thống

#### Mô tả dòng sự kiện phụ

Không

## DFD (Data Flow Diagram)



D1: Giám đốc , quản lí thêm, xoá , sửa quy định

D2: Không có

D3: Các quy định đang hiện hành

D4: D3 + D1

D5: D4

### Ghi chú

Cho phép hủy, cập nhật lại

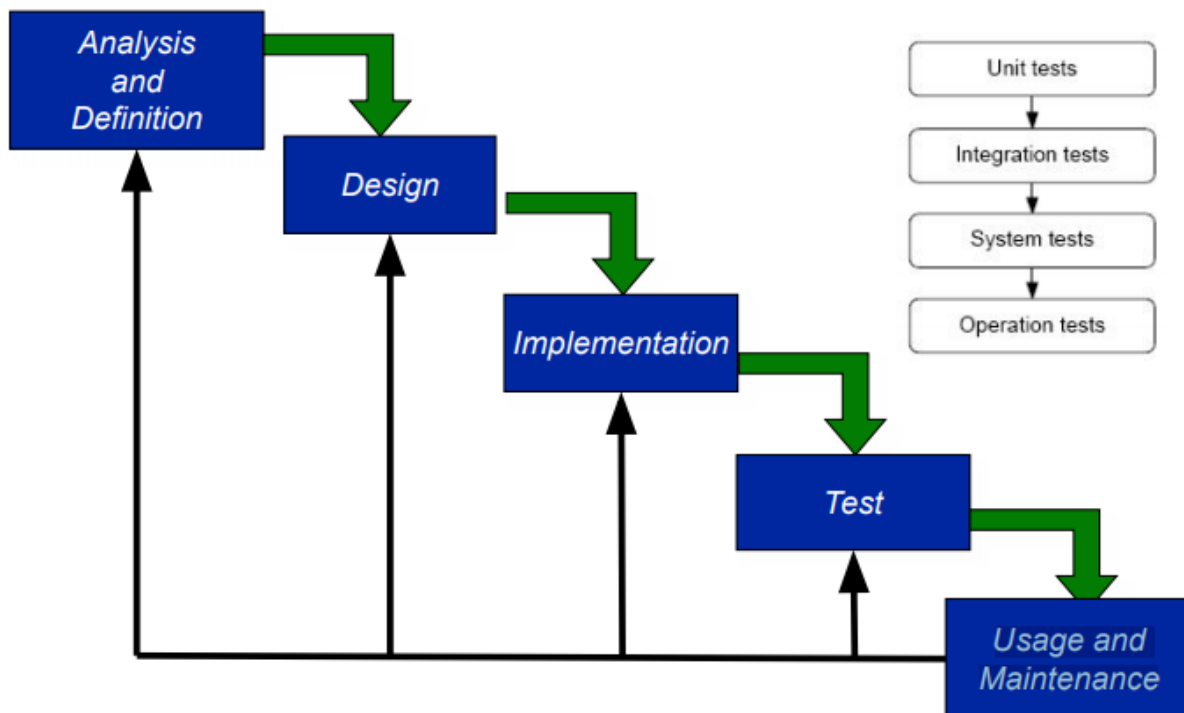
## 2. Kế hoạch, quy trình, công cụ:

### 2.1. Kế hoạch phát triển:

- GIAI ĐOẠN 1: Phân tích các yêu cầu và chức năng mà khách hàng muốn phần mềm có.
- GIAI ĐOẠN 2: Thiết kế phần mềm theo cá phân tích và yêu cầu khách hàng.
- GIAI ĐOẠN 3: Tổng hợp chương trình. Thực thi phần mềm.
- GIAI ĐOẠN 4: Kiểm tra các lỗi, các yêu cầu, chức năng ở phần Phân tích yêu cầu khách hàng.
- GIAI ĐOẠN 5: Kiểm thử, thu thập ý kiến của khách hàng về chương trình. Ở giai đoạn này, nếu khách hàng chưa hài long hay đưa thêm yêu cầu hoặc có sự cố, quay lại các giai đoạn trước. Thực hiện đến khi thoả mãn yêu cầu, từ đó hoàn thiện sản phẩm đầu ra cuối cùng.

### 2.2. Quy trình phát triển đồ án:

Sơ đồ mô hình Thác nước cải tiến (Căn bản):



- Từ mô hình trên, ta có thể chia quy trình thành 6 giai đoạn:

+ **Giai đoạn 1: Analysis and Definition - Phân tích và xác định:**

Bước 1: Phân tích và đặc tả yêu cầu:

\* Gồm có 6 yêu cầu:

1. Mở sổ tiết kiệm
2. Lập phiếu gửi tiền
3. Lập phiếu rút tiền
4. Tra cứu số
5. Lập báo cáo tháng
6. Thay đổi qui định

Dựa vào bản mẫu "Phân tích xác định yêu cầu Đề tài quản lý sổ tiết kiệm" để phân tích 6 yêu cầu trên

Bước 2: Xác định quy trình phát triển:

Xác định sử dụng mô hình Thác nước cải tiến. Xem xét các bước và giai đoạn tiếp theo sau khi đã phân tích các yêu cầu (Gồm 6 yêu cầu)

[ Cụ thể xác định các giai đoạn tổng quát của quy trình như sau:



Bước 3: Phân bổ nguồn lực

Có rất nhiều cách để phân bổ nguồn lực. Ta có thể làm như sau:

Do ta đã xác định sử dụng Mô hình thác nước cải tiến nên ta có thể phân bổ nguồn lực đảm nhiệm 5 vai trò:

1. Analysis and Definition: Bộ phận này nhằm phân tích xác định các yêu cầu
  2. Design and Implementation: Bộ phận này nhằm thiết kế phần mềm phù hợp với các yêu cầu đã được phân tích của bộ phận nắm chức năng Analysis and Definition
  3. Test: Bộ phận này kiểm tra code, kiểm tra lỗi code, sai sót, lập danh sách các lỗi, danh sách các yêu cầu không đáp ứng, ... Nếu có lỗi code thì đưa lại cho bộ phận Design khắc phục code
  4. Usage and Maintenance: Thu thập feedback của user. Thu thập, cập nhật các yêu cầu của user, danh sách các yêu cầu thay đổi, các yêu cầu không được thỏa. Nếu tất cả đã thỏa thì thực hiện nhiệm vụ bảo trì phần mềm
  5. Update: Nếu ở bộ phận Usage and Maintenance thu thập được feedback của user, bộ phận này có chức năng phân loại các feedback của user, sau đó đưa về lại cho từng bộ phận đảm nhiệm phần liên quan đến các feedback đó cho đến khi thỏa hết các yêu cầu user
- \* Như ta biết nhược điểm của Mô hình thác nước là sau khi đã hoàn tất ở bộ phận nào và chuyển giao nhiệm vụ cho bộ phận kế tiếp thì không thể thay đổi hay sửa chữa lỗi của bộ phận trước đó (Nếu có lỗi). Vậy ta nên phân bổ nguồn lực chủ yếu ở các bộ phận đầu tiên (Bộ phận Analysis and Definition và Design and Implementation), trung bình ở bộ phận Test và các bộ phận như Usage and Maintenance và Update thì không cần quá nhiều nguồn nhân lực

### **+ Giai đoạn 2: Design**

Từ những yêu cầu đã được phân tích ở giai đoạn 1, ta thiết kế ra mô hình tổng quát xây dựng phần mềm

Phân chia nguồn lực trong bộ phận có nhiệm vụ Design thành nhiều bộ phận nhỏ hơn, mỗi bộ phận nhỏ đảm nhiệm 1 yêu cầu nhỏ nào đó

\* Trong quá trình thực hiện giai đoạn 2, ta bắt buộc phải thiết kế mô hình đáp ứng các yêu cầu đã được phân tích như trên, sau đó ta mới thiết kế và tự cải tiến phần mềm (Vd: Tốc độ phần mềm, giao diện, độ khả dụng, khả năng tương tác giữa phần mềm và user,...)

### **+ Giai đoạn 3: Implement**

Chạy thử phần mềm. Sau khi các bộ phận đảm nhiệm 1 yêu cầu nhỏ trong bộ phận Design, ta tổng hợp lại 'code' của các bộ phận đó, điều chỉnh và chỉnh sửa để các 'code' đó có thể thực thi



với nhau thành 1 phần mềm hoàn chỉnh. Sau khi phần mềm đã thực thi được thì chuyển sang giai đoạn 4

#### **+ Giai đoạn 4: Test**

Sau khi đã có 1 phần mềm hoàn chỉnh, ta đưa phần mềm đó cho bộ phận Test để kiểm tra phần mềm.

Kiểm tra xem phần mềm đã thỏa các yêu cầu đã được phân tích xác định từ trước hay chưa. Kiểm tra các trường hợp đặc biệt, kiểm tra các trường hợp gây lỗi, tự gây lỗi cho phần mềm, Kiểm tra thời gian, chất lượng phần mềm, ....

Nếu không thỏa thì yêu cầu bộ phận Design and Implementation khắc phục. Sau khi khắc phục thì đưa lại cho bộ phận Test cho tới khi phần mềm được chấp nhận

#### **+ Giai đoạn 5: Usage and Maintenance**

Do bộ phận Usage and Maintenance và bộ phận Update đảm nhiệm. Bộ phận Usage and Maintenance chuyển giao phần mềm 'thử' cho user sử dụng (Phần mềm sau khi đã qua bộ phận Test). Thu thập các feedback của user. Hệ thống các yêu cầu mới của user và cập nhật các yêu cầu của user

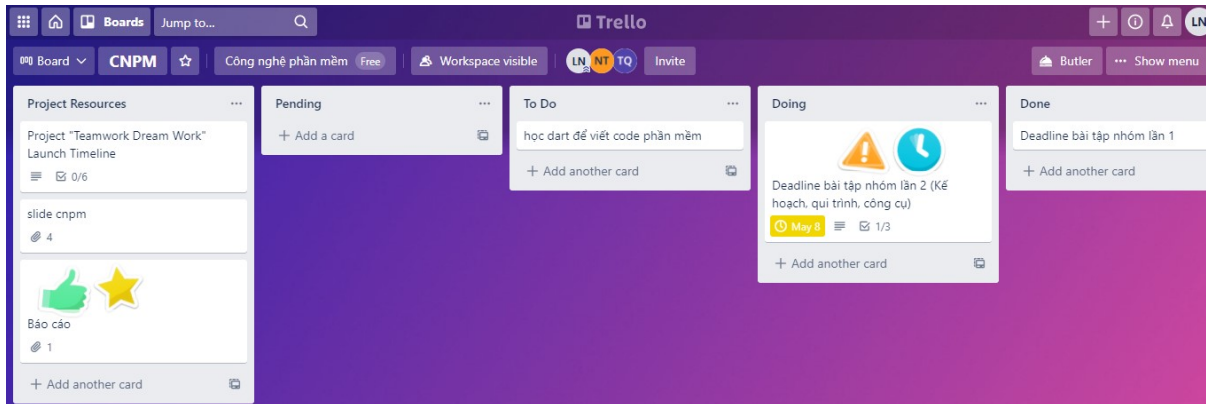
Sau khi đã hệ thống được các feedback của user, bộ phận Update chuyển về cho từng bộ phận các 'lỗi' hay 'không hài lòng' trong feedback của user. Yêu cầu các bộ phận đó chỉnh sửa khắc phục các 'lỗi' đó.

Quá trình này lặp đi lặp lại cho đến phần mềm thỏa các yêu cầu của user. Sau đó ta đưa ra bản hoàn chỉnh của phần mềm

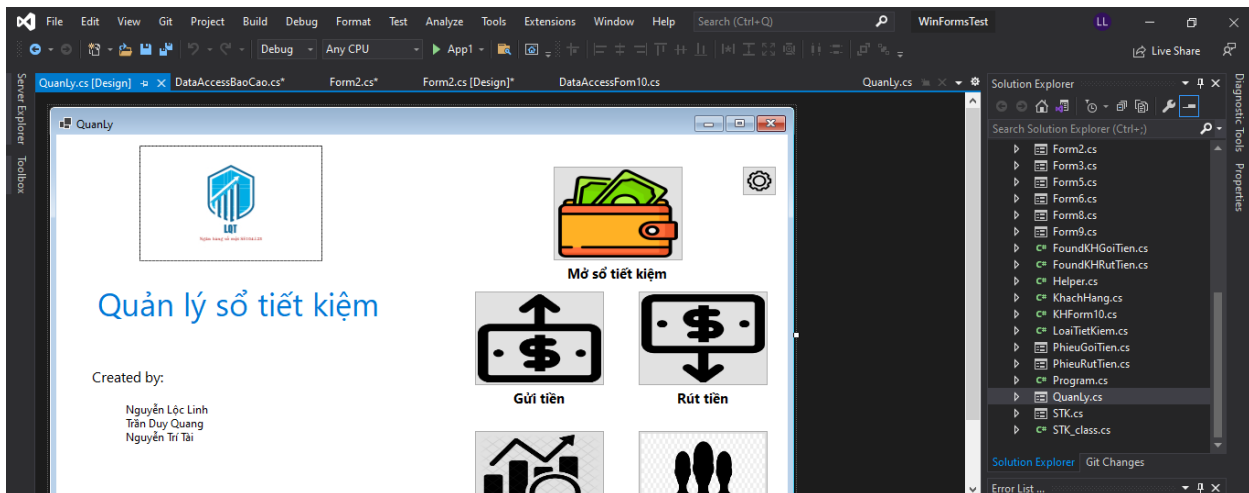
### 2.3. Công cụ (CASE Tools) hỗ trợ trong suốt quá trình quản lý công việc

Các công cụ hỗ trợ bao gồm:

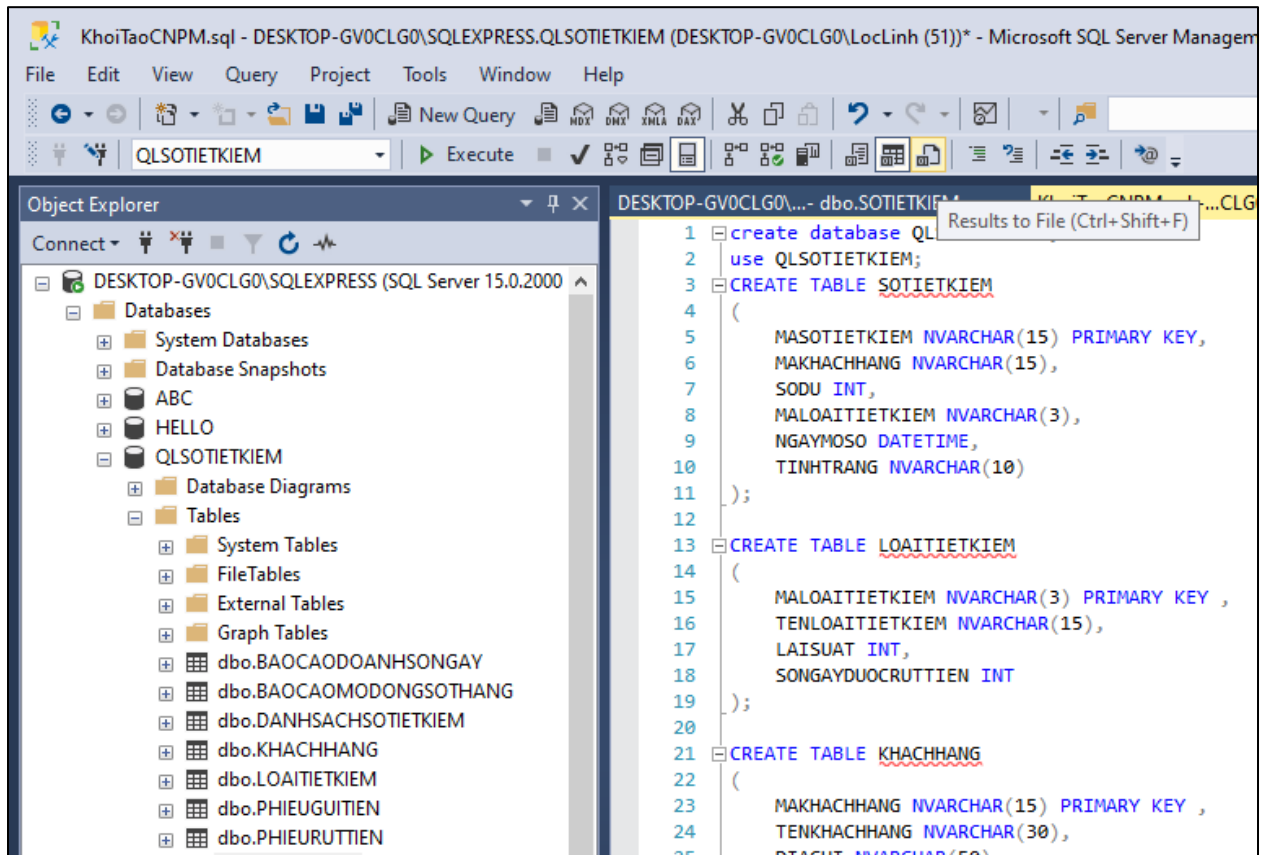
- Roadmap: làm timeline chi tiết về kế hoạch phát triển đồ án
- Trello: công cụ quản lý công việc và thời gian



- Visual studio, ngôn ngữ c#, winform, dotNET platform: Công cụ viết mã chương trình phần mềm.



- SQL server 2019, SQL Server Management Studio : công cụ quản lý dữ liệu



### 3. Thiết kế mô hình dữ liệu:

#### 3.1. Biểu mẫu 1 – Quy định 1:

<b>BM1:</b>	<b>Sổ Tiết Kiệm</b>	
Mã số: .....	Loại tiết kiệm: .....	
Khách hàng: .....	CMND: .....	
Địa chỉ: .....	Ngày mở sổ: .....	
Số tiền gửi: .....		

**QĐ1: Có 3 loại tiết kiệm (không kỳ hạn, 3 tháng, 6 tháng). Số tiền gửi tối thiểu là 100.000**

#### (1) Thiết kế DL đảm bảo tính đúng đắn

- Thuộc tính mới: MaSoTietKiem, *KhachHang*, *DiaChi*, *SoTienGui*, *LoaiTietKiem*, *CMND*, *NgayMoSo*

- Bố trí các thuộc tính vào các bảng đã có

+ MaSo -> **SOTIETKIEM** (MaSoTietKiem)

+ KhachHang -> **KHACHHANG** (MaKhachHang, TenKhachhang)

**SOTIETKIEM → KHACHHANG**

**SOTIETKIEM** (MaSoTietKiem, MaKhachhang)

+ DiaChi -> **KHACHHANG** (MaKhachHang, TenKhachhang, DiaChi)

+ SoTienGui -> **SOTIETKIEM** (MaSoTietKiem, MaKhachHang, SoDu)

+ LoaiTietKiem -> **SOTIETKIEM** (MaSoTietKiem, MaKhachHang, SoTienGui, LoaiTietKiem)

+ CMND -> **KHACHHANG** (MaKhachHang, TenKhachhang, DiaChi, CMND)

+ NgayMoSo -> **SOTIETKIEM** (MaSoTietKiem, MaKhachHang, SoDu, LoaiTietKiem, NgayMoSo)

#### (2) Thiết kế DL đảm bảo tính tiến hoá

- Thuộc tính mới: *TenLoaiTietKiem*

- Bố trí các thuộc tính vào các bảng đã có:

+ TenLoaiTietKiem -> **LOAITIETKIEM** (MaLoaiTietKiem, TenLoaiTietKiem)

**SOTIETKIEM -> LOAITIETKIEM**

**SOTIETKIEM** (MaSoTietKiem, MaKhachHang, SoDu, LoaiTietKiem -> MaLoaiTietKiem, NgayMoSo)

⇒ Bảng đã có:

**SOTIETKIEM** (MaSoTietKiem, MaKhachHang, SoDu, MaLoaiTietKiem, NgayMoSo)

**LOAITIETKIEM** (MaLoaiTietKiem, TenLoaiTietKiem,)

**KHACHHANG** (MaKhachHang, TenKhachhang, DiaChi, CMND)

**THAMSO**(TienGuiToiThieu)

→ Mô hình này là một mô hình đảm bảo tính đúng đắn và tiến hoá.

### 3.2. Biểu mẫu 2 – Quy định 2:

BM2:	Phiếu Gửi Tiền	
Mã số: .....	Khách hàng: .....	
Ngày gửi: .....	Số tiền gửi: .....	

**QĐ2: Chỉ nhận gửi tiền với loại tiết kiệm không kỳ hạn. Số tiền gửi thêm tối thiểu là 100.000**

#### (1) Thiết kế DL đảm bảo tính đúng đắn

- Thuộc tính mới: MaSo, NgayGui, SoTienGui

- Bố trí các thuộc tính vào các bảng đã có:

+ MaSo -> **PHIEUGUITIEN** (MaPhieuGuiTien, )

**PHIEUGUITIEN** -> **SOTIETKIEM**

-> **PHIEUGUITIEN** (MaPhieuGuiTien, MaSoTietKiem)

**PHIEUGUITIEN** -> **KHACHHANG**

-> **PHIEUGUITIEN** (MaPhieuGuiTien, MaSoTietKiem, MaKhachHang)

+ NgayGui -> **PHIEUGUITIEN** (MaPhieuGuiTien, MaSoTietKiem, MaKhachHang, NgayGui)

+ SoTienGui -> **PHIEUGUITIEN** (MaPhieuGuiTien, MaSoTietKiem, MaKhachHang, NgayGui, SoTienGui)

#### (2) Thiết kế DL đảm bảo tính tiến hoá

- Thuộc tính mới: LoaiTietKiemDuocGuiThem, TienGuiThemToiThieu

- Bố trí các thuộc tính vào các bảng đã có:

THAMSO(TienGuiToiThieu, LoaiTietKiemDuocGuiThem, TienGuiThemToiThieu)

⇒ Bảng đã có:

SOTIETKIEM (MaSoTietKiem, MaKhachHang, SoDu, MaLoaiTietKiem, NgayMoSo)

LOAITIETKIEM (MaLoaiTietKiem, TenLoaiTietKiem,)

KHACHHANG (MaKhachHang, TenKhachhang, DiaChi, CMND)

PHIEUGUITIEN (MaPhieuGuiTien, MaSoTietKiem, MaKhachHang, NgayGui, SoTienGui)

### 3.3. Biểu mẫu 3 – Quy định 3:

BM3:	Phiếu Rút Tiền
Mã số: .....	Khách hàng: .....
Ngày rút: .....	Số tiền rút: .....

**QĐ3: Lãi suất là 0.5% đối với loại không kỳ hạn, 5% với kỳ hạn 3 tháng và 5.5% với kỳ hạn 6 tháng.**

**Tiền lãi 1 năm = số dư \* lãi suất của loại tiết kiệm tương ứng.**

**Loại tiết kiệm có kỳ hạn chỉ được rút khi quá kỳ hạn và phải rút hết toàn bộ, khi này tiền lãi được tính với mức lãi suất của loại không kỳ hạn.**

**Loại tiết kiệm không kỳ hạn được rút khi gửi trên 15 ngày và có thể rút số tiền <= số dư hiện có.**

**Số sau khi rút hết tiền sẽ tự động đóng.**

#### (1) Thiết kế DL đảm bảo tính đúng đắn

- Thuộc tính mới: MaSoRutTien, NgayRut, SoTienRut.

- Bố trí các thuộc tính vào các bảng đã có:

+MaSoRutTien→ PHIEURUTTIEN(MaSoRutTien)

PHIEURUTTIEN → SOTIETKIEM

PHIEURUTTIEN (MaSoRutTien, MaSoTietKiem)

PHIEURUTTIEN → KHACHHANG

**PHIEURUTTIEN** (MaSoRutTien, MaSoTietKiem, NgayRut, MaKhachHang)

+ NgayRut → **PHIEURUTTIEN** (MaSoRutTien, MaSoTietKiem, MaKhachHang, NgayRut)

+ SoTienRut → **PHIEURUTTIEN** (MaSoRutTien, MaSoTietKiem, MaKhachHang, NgayRut, SoTienRut)

## (2) Thiết kế DL đảm bảo tính tiến hoá

- Thuộc tính mới: LaiSuat, SoNgayDuocRutTien, SoTienConLai, TinhTrang, SoNgayRutLoaiKhongKiHan

- Bố trí các thuộc tính vào các bảng đã có:

+ LaiSuat → **LOAITIETKIEM** (MaLoaiTietKiem, TenLoaiTietKiem, LaiSuat)

+ SoNgayDuocRutTien → **LOAITIETKIEM** (MaLoaiTietKiem, TenLoaiTietKiem, LaiSuat, SoNgayDuocRutTien)

+ TinhTrang → **SOTIETKIEM** (MaSoTietKiem, MaKhachHang, SoDu, MaLoaiTietKiem, NgayMoSo, TinhTrang)

+ SoNgayRutLoaiKhongKiHan → **THAMSO** (TienGuiToiThieu, LoaiTietKiemDuocGuiThem, TienGuiThemToiThieu, SoNgayRutLoaiKhongKiHan)

⇒ Bảng đã có:

**SOTIETKIEM** (MaSoTietKiem, MaKhachHang, SoDu, MaLoaiTietKiem, NgayMoSo, TinhTrang)

**LOAITIETKIEM** (MaLoaiTietKiem, TenLoaiTietKiem, LaiSuat, SoNgayDuocRutTien)

**KHACHHANG** (MaKhachHang, TenKhachhang, DiaChi, CMND)

**PHIEUGUITIEN** (MaPhieuGuiTien, MaSoTietKiem, MaKhachHang, NgayGui, SoTienGui)

**PHIEURUTTIEN** (MaSoRutTien, MaSoTietKiem, MaKhachHang, NgayRut, SoTienRut)

**THAMSO** (TienGuiToiThieu, LoaiTietKiemDuocGuiThem, TienGuiThemToiThieu, SoNgayRutLoaiKhongKiHan)

### 3.4. Biểu mẫu 4:

BM4:		Danh Sách Sổ Tiết Kiệm		
STT	Mã Số	Loại Tiết Kiệm	Khách Hàng	Số Dư
1				
2				

#### (1) Thiết kế DL đảm bảo tính đúng đắn

- Thuộc tính mới: (Không)
- Bố trí các thuộc tính vào các bảng đã có:

**DSSOTIETKIEM** (MaDS)

**DSSOTIETKIEM** → **SOTIETKIEM**

**DSSOTIETKIEM** (MaDS, MaSoTietKiem)

**DSSOTIETKIEM** → **LOAITIETKIEM**

**DSSOTIETKIEM** (MaDS, MaSoTietKiem, MaLoaiTietKiem)

**DSSOTIETKIEM** → **KHACHHANG**

**DSSOTIETKIEM** (MaDS, MaSoTietKiem, MaLoaiTietKiem, MaKhachHang)

#### (2) Thiết kế DL đảm bảo tính tiến hoá

- Thuộc tính mới: Không
- Bố trí các thuộc tính vào các bảng đã có:

⇒ Bảng đã có:

**SOTIETKIEM** (MaSoTietKiem, MaKhachHang, SoDu, MaLoaiTietKiem, NgayMoSo, TinhTrang)

**LOAITIETKIEM** (MaLoaiTietKiem, TenLoaiTietKiem, LaiSuat, SoNgayDuocRutTien)

**KHACHHANG** (MaKhachHang, TenKhachhang, DiaChi, CMND)

**PHIEUGUITIEN** (MaPhieuGuiTien, MaSoTietKiem, MaKhachHang, NgayGui, SoTienGui)

**PHIEURUTTIEN** (MaSoRutTien, MaSoTietKiem, MaKhachHang, NgayRut, SoTienRut)

**DSSOTIETKIEM** (MaDS, MaSoTietKiem, MaLoaiTietKiem, MaKhachHang)

**THAMSO** (TienGuiToiThieu, LoaiTietKiemDuocGuiThem, TienGuiThemToiThieu, SoNgayRutLoaiKhongKiHan)



### 3.5. Biểu mẫu 5:

<b>BM5.1</b>	<b>Báo Cáo Doanh Số Hoạt Động Ngày</b>			
Ngày:.....				
<b>STT</b>	<b>Loại Tiết Kiệm</b>	<b>Tổng Thu</b>	<b>Tổng Chi</b>	<b>Chênh Lệch</b>
1				
2				

<b>BM5.2</b>	<b>Báo Cáo Mở/Đóng Sổ Tháng</b>			
Loại tiết kiệm: .....		Tháng:.....		
<b>STT</b>	<b>Ngày</b>	<b>Số Mở</b>	<b>Số Đóng</b>	<b>Chênh Lệch</b>
1				
2				

#### Xét yc 5.1:

##### (1) Thiết kế DL đảm bảo tính đúng đắn

- Thuộc tính mới: Ngày, TongThu, TongChi
- Bố trí các thuộc tính vào các bảng đã có:  
 +Ngày → **BAOCAODOANHSONGAY** (MaBaoCaoDoanhSo, Ngày)  
**BAOCAODOANHSONGAY** → **LOAITIETKIEM**  
**BAOCAODOANHSONGAY** (MaBaoCaoDoanhSo, Ngày, MaLoaiTietKiem)  
 +TongThu → **BAOCAODOANHSONGAY** (MaBaoCaoDoanhSo, Ngày, MaLoaiTietKiem, TongThu)  
 +TongChi → **BAOCAODOANHSONGAY** (MaBaoCaoDoanhSo, Ngày, MaLoaiTietKiem, TongThu, TongChi)

##### (2) Thiết kế DL đảm bảo tính tiến hoá

- Thuộc tính mới: Không
- Bố trí các thuộc tính vào các bảng đã có

#### Xét yc 5.2:

##### (1) Thiết kế DL đảm bảo tính đúng đắn

- Thuộc tính mới: Thang, Ngày, SoMo, SoDong

- Bố trí các thuộc tính vào các bảng đã có:

+ Thang → BaoCaoMoDongSoThang (MaBaoCaoMDThang, Thang)

**BaoCaoMoDongSoThang → LOAITIETKIEM**

**BaoCaoMoDongSoThang (MaBaoCaoMDThang, MaLoaiTietKiem, Thang)**

+ Ngay → BaoCaoMoDongSoThang (MaBaoCaoMDThang, MaLoaiTietKiem, Thang, Ngay)

+ SoMo → BaoCaoMoDongSoThang (MaBaoCaoMDThang, MaLoaiTietKiem, Thang, Ngay, SoMo)

+ SoDong → BaoCaoMoDongSoThang (MaBaoCaoMDThang, MaLoaiTietKiem, Thang, Ngay, SoMo, SoDong)

## (2) Thiết kế DL đảm bảo tính tiến hoá

- Thuộc tính mới: (Không)
- Bố trí các thuộc tính vào các bảng đã có

⇒ Bảng đã có:

**SOTIETKIEM (MaSoTietKiem, MaKhachHang, SoDu, MaLoaiTietKiem, NgayMoSo, TinhTrang)**

**LOAITIETKIEM (MaLoaiTietKiem, TenLoaiTietKiem, LaiSuat, SoNgayDuocRutTien)**

**KHACHHANG (MaKhachHang, TenKhachhang, DiaChi, CMND)**

**PHIEUGUITIEN (MaPhieuGuiTien, MaSoTietKiem, MaKhachHang, NgayGui, SoTienGui)**

**PHIEURUTTIEN (MaSoRutTien, MaSoTietKiem, MaKhachHang, NgayRut, SoTienRut)**

**DSSOTIETKIEM (MaDS, MaSoTietKiem, MaLoaiTietKiem, MaKhachHang)**

**BAOCAODOANHSONGAY (MaBaoCaoDoanhSo, Ngay, MaLoaiTietKiem, TongThu, TongChi)**

**BaoCaoMoDongSoThang (MaBaoCaoMDThang, MaLoaiTietKiem, Thang, Ngay, SoMo, SoDong)**

**THAMSO (TienGuiToiThieu, LoaiTietKiemDuocGuiThem, TienGuiThemToiThieu, SoNgayRutLoaiKhongKiHan)**

### 3.6. Biểu mẫu 6 – Quy định 6:

**QĐ6:** Người dùng có thể thay đổi các qui định như sau:

- + QĐ1: Thay đổi số lượng các loại kỳ hạn, tiền gửi tối thiểu.
- + QĐ3: Thay đổi thời gian gửi tối thiểu và lãi suất các loại kỳ hạn.

Xét yc 6:

**(1) Thiết kế DL đảm bảo tính đúng đắn**

- Thuộc tính mới: Không
- Bố trí các thuộc tính vào các bảng đã có

**(2) Thiết kế DL đảm bảo tính tiến hoá**

- Thuộc tính mới: Không
- Bố trí các thuộc tính vào các bảng đã có

⇒ Bảng đã có:

**SOTIETKIEM** (MaSoTietKiem, MaKhachHang, SoDu, MaLoaiTietKiem, NgayMoSo, TinhTrang)

**LOAITIETKIEM** (MaLoaiTietKiem, TenLoaiTietKiem, LaiSuat, SoNgayDuocRutTien)

**KHACHHANG** (MaKhachHang, TenKhachhang, DiaChi, CMND)

**PHIEUGUITIEN** (MaPhieuGuiTien, MaSoTietKiem, MaKhachHang, NgayGui, SoTienGui)

**PHIEURUTTIEN** (MaSoRutTien, MaSoTietKiem, MaKhachHang, NgayRut, SoTienRut)

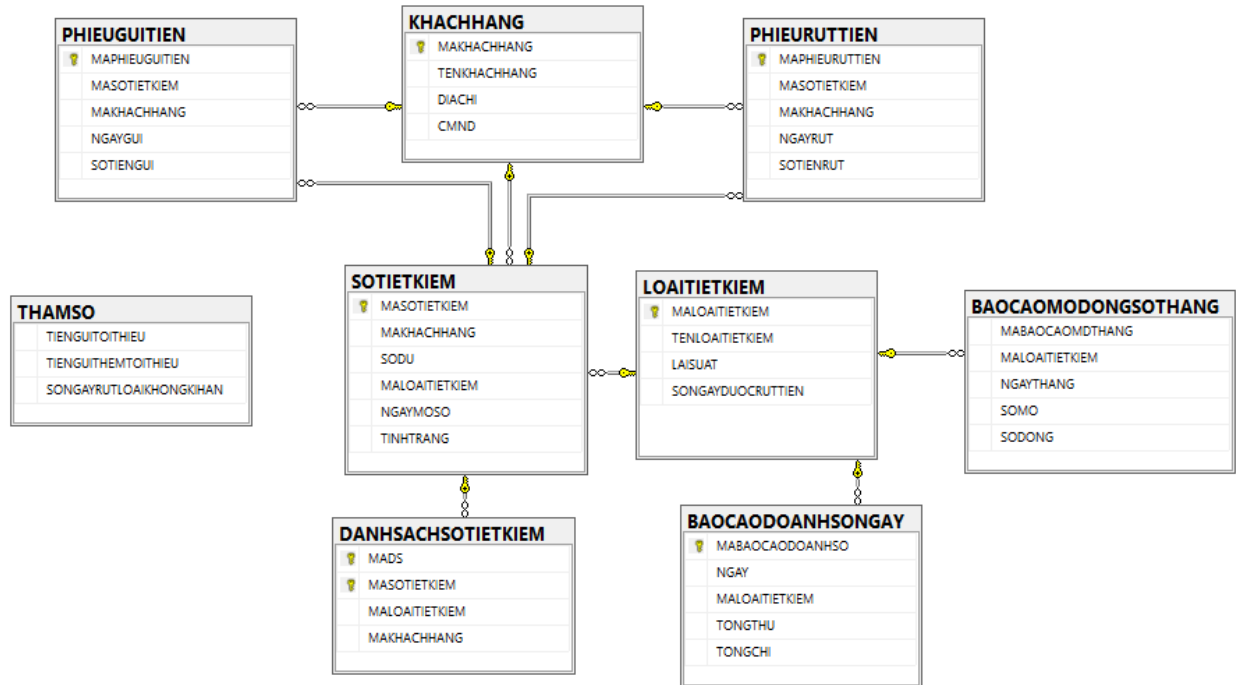
**DSSOTIETKIEM** (MaDS, MaSoTietKiem, MaLoaiTietKiem, MaKhachHang)

**BAOCAODOANHSONGAY** (MaBaoCaoDoanhSo, Ngay, MaLoaiTietKiem, TongThu, TongChi)

**BaoCaoMoDongSoThang** (MaBaoCaoMDThang, MaLoaiTietKiem, Thang, Ngay, SoMo, SoDong)

**THAMSO** (TienGuiToiThieu, LoaiTietKiemDuocGuiThem, TienGuiThemToiThieu,  
SoNgayRutLoaiKhongKiHan)

## Database diagram:



#### 4. Thiết kế giao diện:

##### 4.1. Biểu mẫu 1 – Quy định 1:

###### 1.9.2.1 Biểu mẫu 1 và quy định 1

<b>BM1:</b>	<b>Số Tiết Kiệm</b>
Mã số: .....	Loại tiết kiệm: .....
Khách hàng: .....	CMND: .....
Địa chỉ: .....	Ngày mở số: .....
Số tiền gửi: .....	

**QD1: Có 3 loại tiết kiệm (không kỳ hạn, 3 tháng, 6 tháng). Số tiền gửi tối thiểu là 100.000**

- 1) Kiểm tra ngày sinh: (Ngày sinh có hợp lệ không?)
- 2) Kiểm tra số CMND hoặc CCCD (Hết hạn sử dụng hay chưa?)
- 3) Kiểm tra số tiền
- 4) Kiểm tra người đăng ký đã nhập đầy đủ thông tin Quan trọng (\*) hay chưa?
- 5) Kiểm tra lỗi chính tả

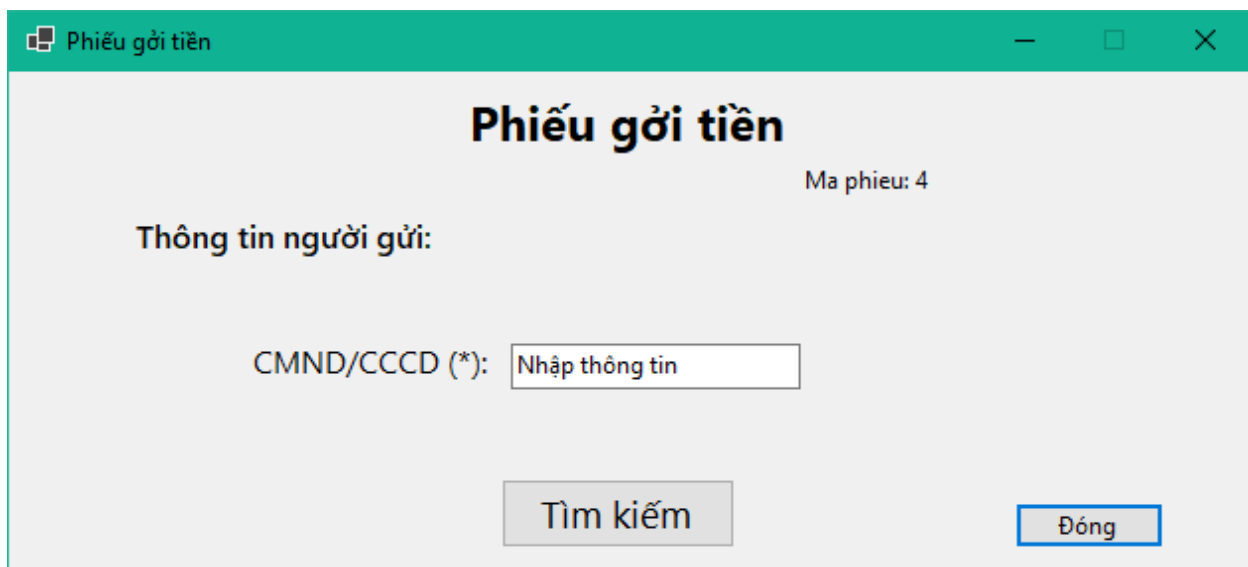


1.9.2.2 Biểu mẫu 2 và quy định 2

<b>BM2:</b>	<b>Phiếu Gửi Tiền</b>	
Mã số: .....	Khách hàng: .....	
Ngày gửi: .....	Số tiền gửi: .....	

**QB2: Chỉ nhận gửi tiền với loại tiết kiệm không kỳ hạn. Số tiền gửi thêm tối thiểu là 100.000**

- 1) Kiểm tra thông tin Họ Tên và CMND có hợp lệ. Nếu hợp lệ -> Hiển thị Thông tin sổ tiết kiệm
- 2) Kiểm tra Loại tiết kiệm. Nếu Loại tiết kiệm là Không kỳ hạn -> Xuất "*Bạn có muốn gửi tiền không?*" -> Chọn "*Yes*" -> Xuất "*Thông tin gửi tiền*"
- 3) Kiểm tra số tiền
- 4) Kiểm tra người đăng ký đã nhập đầy đủ thông tin Quan trọng (\*) hay chưa?
- 5) Kiểm tra lỗi chính tả



Phiếu gửi tiền

**Phiếu gửi tiền**

Ma phiếu: 4

Thông tin người gửi:

CMND/CCCD (\*):

Figure 2 Giao diện tìm thông tin sổ tiết kiệm

Thông tin người gửi tiền

Phiếu gửi tiền

Mã phiếu: 4

Thông tin sổ tiết kiệm:

Họ tên:

Nguyen Loc Linh

CMND/CCCD:

19521754

Địa chỉ:

123 ABC

Số dư hiện tại:

1324467

Loại tiết kiệm:

khong ky han

Bạn có muốn gửi tiền ko?

☐ Có

☐ Không

Xác nhận

Đóng

Figure 3 Giao diện hiển thị thông tin sổ tiết kiệm và đưa lựa chọn gửi tiền

Thông tin gửi tiền

Phiếu gửi tiền

Mã phiếu: 4

Thông tin gửi tiền:

Người gửi:

Nguyen Loc Linh

Số tiền gửi (\*):

vnd

Ngày gửi (\*):

Xác nhận

Đóng

Figure 4 Giao diện nhập thông tin gửi tiền và xác nhận



### 4.3. Biểu mẫu 3 – Quy định 3:

<b>BM3:</b>	<b>Phiếu Rút Tiền</b>	
Mã số: .....	Khách hàng: .....	
Ngày rút: .....	Số tiền rút: .....	

**QĐ3:** Lãi suất là 0.5% đối với loại không kỳ hạn, 5% với kỳ hạn 3 tháng và 5.5% với kỳ hạn 6 tháng.

Tiền lãi 1 năm = số dư \* lãi suất của loại tiết kiệm tương ứng.

Loại tiết kiệm có kỳ hạn chỉ được rút khi quá kỳ hạn và phải rút hết toàn bộ, khi này tiền lãi được tính h với mức lãi suất của loại không kỳ hạn.

Loại tiết kiệm không kỳ hạn được rút khi gửi trên 15 ngày và có thể rút số tiền  $\leq$  số dư hiện có.

Số sau khi rút hết tiền sẽ tự động đóng.

Figure 5 Giao diện tìm thông tin sổ tiết kiệm

Thông tin rút tiền

Phiếu rút tiền

Mã phiếu: 3

Thông tin sổ tiết kiệm:

Họ tên:

Co The Rut

CMND/CCCD:

11223344

Địa chỉ:

111 aaa, Đ?ng Nai

Số dư hiện tại:

100000

Loại tiết kiệm:

3 tháng

Bạn có muốn rút tiền ko?

☐ Có

☐ Không

Xác nhận

Đóng

Figure 6 Giao diện hiển thị thông tin sổ tiết kiệm và cho lựa chọn rút tiền

Thông tin rút tiền

Phiếu rút tiền

Mã phiếu: 3

Thông tin rút tiền:

Người rút:

Co The Rut

Số dư:

100000

Số tiền rút (\*):

vnd

Ngày rút (\*):

Xác nhận

Đóng

Figure 7 Giao diện nhập thông tin rút tiền và xác nhận rút

#### 4.4. Biểu mẫu 4:

BM4:		Danh Sách Sổ Tiết Kiệm		
STT	Mã Số	Loại Tiết Kiệm	Khách Hàng	Số Dư
1				
2				

Form10

×

Danh sách khách hàng

Ngày thống kê: 10/07/2021

Lọc:

Loại tiết kiệm: Tất cả

Số dư: tất cả

Tìm kiếm:

Họ và tên: tìm theo tên

Tìm

STT	Mã số	Loại tiết kiệm	Khách hàng	Số dư
1	01	không kỳ hạn	Nguyen Loc Linh	1324467
2	02	3 tháng	Nguyen Van A	224466
3	03	6 tháng	Tr?n Duy Quang	235689
4	04	3 tháng	Nguyen Tri Tai	4433467
5	05	3 tháng	Co The Rut	100000

Đóng

Figure 8 Giao diện tìm kiếm và hiển thị danh sách khách hàng

#### 4.5. Biểu mẫu 5:

<b>BM5.1</b>	<b>Báo Cáo Doanh Số Hoạt Động Ngày</b>			
Ngày:.....				
<b>STT</b>	<b>Loại Tiết Kiệm</b>	<b>Tổng Thu</b>	<b>Tổng Chi</b>	<b>Chênh Lệch</b>
1				
2				

<b>BM5.2</b>		<b>Báo Cáo Mô/Đóng Sổ Tháng</b>		
Loại tiết kiệm: .....			Tháng:.....	
<b>STT</b>	<b>Ngày</b>	<b>Sổ Mơ</b>	<b>Sổ Đóng</b>	<b>Chênh Lệch</b>
1				
2				

Báo cáo đóng/mở sổ
— □ ×

## BÁO CÁO MỞ/ĐÓNG SỔ THÁNG

Loại tiết kiệm Chọn loại tiết kiệm ▾

THÁNG 06/2021 ▾

STT	Ngày	Số sổ mở	Số sổ đóng	Chênh lệch
1	1/6			
2	2/6			
3	3/6			
4	4/6			

**Tổng cộng**

Tổng số mở:
Tổng số đóng:
Chênh lệch

Export
Đóng
In

Figure 9 Giao diện báo cáo sổ mở và đóng trong tháng

Báo cáo doanh số

## BÁO CÁO DOANH SỐ HOẠT ĐỘNG NGÀY

NGÀY 

Cập nhật

STT	Loại tiết kiệm	Tổng thu	Tổng chi	Chênh lệch
1	Không kì hạn	312233	0	312233
2	hạn 3 tháng	0	23445	-23445
3	hạn 6 tháng	0	0	0
	<b>Tổng cộng</b>	312233	23445	288788

Export

Đóng

In

Figure 10 Giao diện báo cáo doanh số theo ngày

#### 4.6. Biểu mẫu 6 – Quy định 6:

**QĐ6:** Người dùng có thể thay đổi các quy định như sau:

- + QĐ1: Thay đổi số lượng các loại kỳ hạn, tiền gửi tối thiểu.
- + QĐ3: Thay đổi thời gian gửi tối thiểu và lãi suất các loại kỳ hạn.

The screenshot shows a software window titled "Cài đặt" (Settings) with a teal header bar. The main content area is light gray and contains several sections for configuring deposit types:

- Loại kỳ hạn:** (Deposit Type)
  - Thêm loại kỳ hạn: (Add deposit type)
    - Tên: <Nhập> (Name)
    - Lãi suất: <Nhập> (Interest rate)
    - Kỳ hạn: <Nhập> (Term)
    - Thêm mới (Add new)
  - danh sách loại kỳ hạn: (Deposit type list)
    - hạn 3 tháng (3 months term) is selected in a dropdown.
    - Tên: hạn 3 tháng (Name)
    - Lãi suất: 5% (Interest rate)
    - Kỳ hạn: 90 ngày (Term)
    - Tên mới: <Nhập> (New name)
    - Lãi suất mới: <Nhập> (New interest rate)
    - Kỳ hạn mới: <Nhập> (New term)
- Tiền gửi tối thiểu:** (Minimum deposit)
  - Tiền gửi tối thiểu hiện tại: 100.000đ (Current minimum deposit)
  - Tiền gửi tối thiểu mới: <Nhập> (New minimum deposit)
- Thời hạn gửi tối thiểu của loại kỳ hạn:** (Minimum deposit term)
  - Thời hạn gửi hiện tại: 15 ngày (Current deposit term)
  - Thời hạn gửi mới: <Nhập> (New deposit term)

At the bottom right, there are two buttons: "Xác nhận thay đổi" (Confirm change) and "Quay lại" (Back).

Figure 11 Giao diện thay đổi thuộc tính các loại kỳ hạn và các tham số khác

#### 4.7. Thiết kế giao diện quản lý

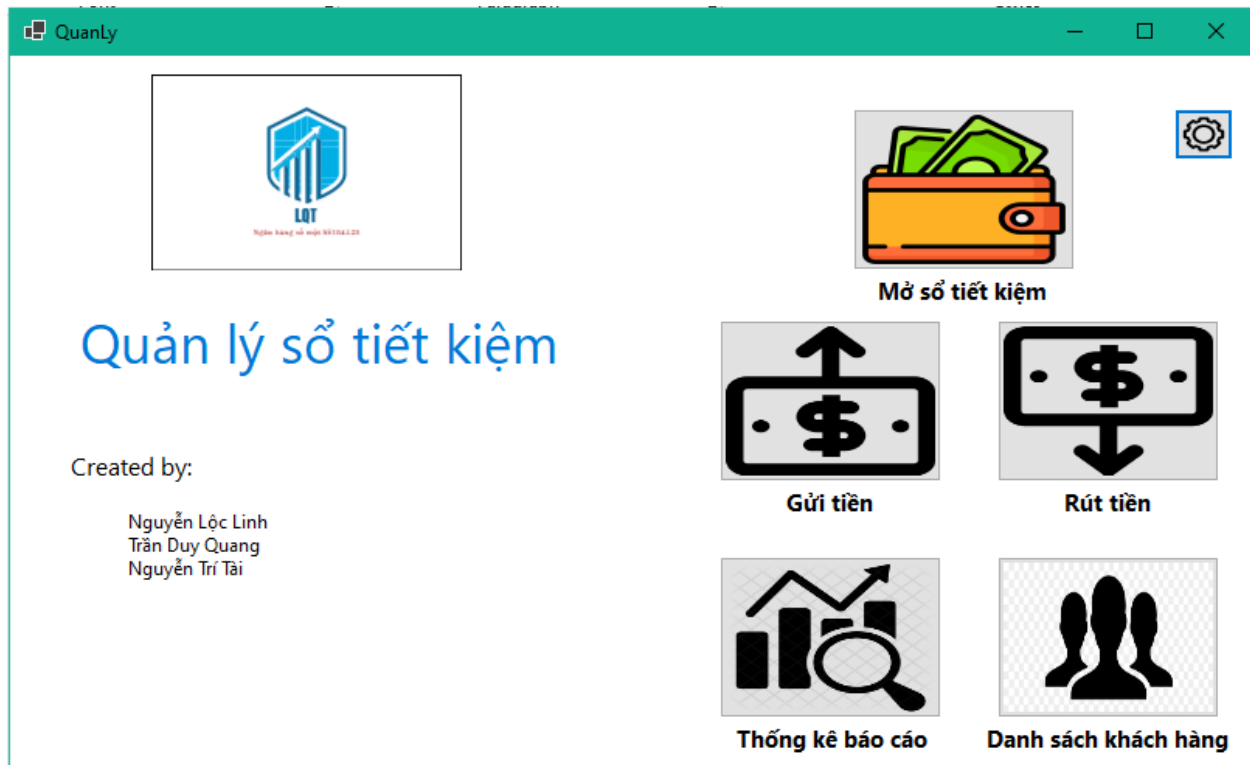


Figure 12 Giao diện lựa chọn các chức năng

## 5. Thiết kế xử lý:

### 5.1. Biểu mẫu 1 – Quy định 1:

#### 1.9.2.1 Biểu mẫu 1 và qui định 1

<b>BM1:</b>	<b>Số Tiết Kiệm</b>
Mã số: .....	Loại tiết kiệm: .....
Khách hàng: .....	CMND: .....
Địa chỉ: .....	Ngày mở số: .....
Số tiền gửi: .....	

**QD1: Có 3 loại tiết kiệm (không kỳ hạn, 3 tháng, 6 tháng). Số tiền gửi tối thiểu là 100.000**

- 1) Kiểm tra ngày sinh: (Ngày sinh có hợp lệ không?)
- 2) Kiểm tra số CMND hoặc CCCD (Hết hạn sử dụng hay chưa?)
- 3) Kiểm tra số tiền
- 4) Kiểm tra người đăng ký đã nhập đầy đủ thông tin Quan trọng (\*) hay chưa?
- 5) Kiểm tra lỗi chính tả



<Thông tin ngân hàng>

Mã Số: <Hệ thống tự lập>

(Tên, địa chỉ, chi nhánh,...)

## Sổ tiết kiệm

### Thông tin khách hàng:

Họ tên (\*): <Nhập từ bàn phím>

Địa chỉ (\*):

(1) Tỉnh (\*): <Bảng chọn tỉnh>

(2) Thành phố (\*): <Bảng chọn Thành phố>

(3) Phường (\*): <Bảng chọn phường>

Địa chỉ nhà (số nhà + đường): <Nhập từ bàn phím>

Địa chỉ cụ thể: <Nhập từ bàn phím>

CMND (\*):

CMND hoặc CCCD (1): <Nhập từ bàn phím>

CMND người giám hộ (2): <Nhập từ bàn phím>

(4) Ngày mở sổ (\*): <Ngày mặc định> hoặc <Bảng chọn dd/mm/yyyy> (Người gửi muốn chọn)

Số điện thoại: <Nhập từ bàn phím>

Địa chỉ email: <Nhập từ bàn phím>

### Thông tin đăng ký sổ:

(5) Số tiền tiết kiệm (\*): <Nhập từ bàn phím>

(Có dấu ‘.’ để phân chia đơn vị tiền – Nghìn, triệu, tỉ, trăm nghìn tỉ,...>)

(Kiểm tra QĐ1 – Nếu trên 100k -> Hiện dòng Số tiền hợp lệ)

(6) Chọn loại tiết kiệm (\*): <Chọn 1 trong 3 bảng>

☐ Kỳ hạn 6 tháng

☐ Kỳ hạn 3 tháng

☐ Không kỳ hạn

(\* ) Bắt buộc (1) Nếu người đăng ký có cmnd (2) Nếu người đăng ký không có cmnd

(7)

Đóng

(8)

Kiểm tra

(9)

Xác nhận

## Thiết kế xử lý:

**Bước 1:** Bổ sung các nút xử lý trên từng màn hình giao diện đã thiết kế

Bổ sung Nút Kiểm tra để kiểm tra thông tin người dùng nhập đã hợp lệ hay chưa

**Bước 2:** Lập danh sách các xử lý trên mỗi màn hình.

STT	Tên xử lý	Ý nghĩa	Mô tả	Điều kiện gọi	Ghi chú
1	XL1	Chọn 1 tỉnh	Chọn 1 tỉnh trong các tỉnh	Khi form được load lên	
2	XL2	Chọn 1 thành phố	Chọn 1 thành phố trong tỉnh đã chọn	Sau khi đã chọn tỉnh	
3	XL3	Chọn 1 phường	Chọn 1 phường trong thành phố đã chọn	Sau khi đã chọn thành phố	
4	XL4	Lấy ngày hệ thống hoặc chọn ngày	Lấy ngày hệ thống hoặc chọn ngày	Khi form được load lên	
5	XL5	Nhập số tiền	Nhập số tiền và tự động phân đơn vị nghìn, triệu, tỉ	Khi form được load lên	
6	XL6	Chọn loại tiết kiệm	Chọn 1 trong 3 loại tiết kiệm	Khi form được load lên	
7	XL7	Đóng đăng ký	Đóng bảng đăng ký	Khi form được load lên	
8	XL8	Kiểm tra thông tin đăng ký	Kiểm tra thông tin có hợp lệ hay không	Khi form được load lên	Hiện vị trí không hợp lệ
9	XL9	Xác nhận đăng ký	Xác nhận đăng ký và đóng bảng đăng ký	Khi form được load lên	

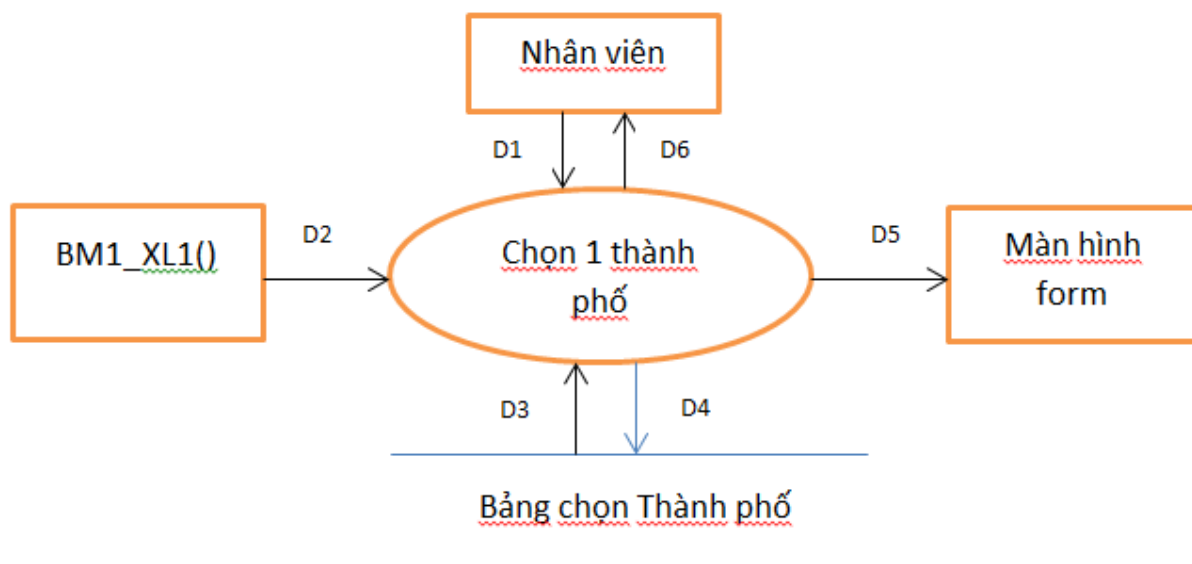
Hàm xử lý tương ứng:

Tên xử lý	Tên hàm	Mô tả	Ghi chú
XL1	BM1_XL1()	Tạo bảng chọn danh sách 63 tỉnh và cho phép người dùng chọn 1 tỉnh trong danh sách đó. Người dùng có thể nhập vài chữ cái và hiện tỉnh liên quan nhất	
XL2	BM1_XL2(Province)	Tạo bảng chọn danh sách các thành phố/quận trong tỉnh mà người dùng đã chọn.	Tham số "Province": Tỉnh mà người dùng đã chọn ở hàm BM1_XL1()
XL3	BM1_XL3(City)	Tạo bảng chọn danh sách các phường trong thành phố/quận mà người dùng đã chọn	Tham số "City": Thành phố/quận mà người dùng đã chọn ở hàm BM1_XL2(Province)
XL4	BM1_XL4()	Tạo 1 bảng chọn dd/mm/yyyy. Tự động chọn Ngày hệ thống và người dùng có thể thay đổi ngày bằng chọn dd/mm/yyyy	
XL5	BM1_XL5()	Tạo bảng cho người dùng nhập số tiền. Tự động phân chia đơn vị tiền nghìn, triệu, tỉ,...	

XL6	BM1_XL6()	Tạo 3 mục chọn Loại tiết kiệm. Chỉ cho phép người dùng chọn 1 loại tiết kiệm	
XL7	BM1_XL7()	Lệnh đóng bảng đăng ký	
XL8	BM1_XL8()	Lệnh kiểm tra bảng đăng ký. Nếu hợp lệ, hiển thị "Hợp lệ!" và trả về 1. Nếu không hợp lệ, hiển thị vị trí không hợp lệ và trả về 0. Vd: "'CMND' không hợp lệ!"	
XL9	BM1_XL9(Check)	Lệnh xác nhận đăng ký và đóng bảng. Chỉ hiển thị được phép chọn sau khi đã nhấn button Kiểm tra và thông tin hợp lệ	Tham số "Check": Giá trị trả về của BM_XL8()

**Bước 3:** Mô tả chi tiết một số xử lý phức tạp, quan trọng:

XL2 (Xử lý 2):



- Mô tả luồng dữ liệu:

D1: Các ký tự Nhân viên nhập để tìm kiếm thành phố

D2: Tỉnh đã được chọn ở hàm BM\_XL1()

D3: Danh sách các thành phố trong tỉnh đã chọn ở hàm BM\_XL1()

D4: Không có

D5: Thành phố liên quan nhất và các thành phố cũng liên quan đến D1

D6: D5

- Thuật toán:

B1: Nhận D1 từ Nhân viên

B2: Nhận D2 từ hàm BM1\_XL1() (Tên tỉnh)

B3: Kết nối cơ sở dữ liệu

B4: Đọc D3 từ Bảng chọn Thành phố

B5: Hiển thị các thành phố (D3) liên quan nhất với các ký tự nhập vào (D1) lên bảng chọn Thành phố của form

B6: Xuất D5 lên form ở bảng Thành phố.

B7: Đóng kết nối cơ sở dữ liệu

**Bước 4:** Rà soát các yêu cầu để cải tiến thuật giải nếu cần.

## 5.2. Biểu mẫu 2 – Quy định 2:

### 1.9.2.2 Biểu mẫu 2 và qui định 2

<b>BM2:</b>	<b>Phiếu Gửi Tiền</b>	
Mã số: .....	Khách hàng: .....	
Ngày gửi: .....	Số tiền gửi: .....	

**QĐ2: Chỉ nhận gửi tiền với loại tiết kiệm không kỳ hạn. Số tiền gửi thêm tối thiểu là 100.000**

- 1) Kiểm tra thông tin Họ Tên và CMND có hợp lệ. Nếu hợp lệ -> Hiện thị Thông tin sổ tiết kiệm
- 2) Kiểm tra Loại tiết kiệm. Nếu Loại tiết kiệm là Không kỳ hạn -> Xuất "*Bạn có muốn gửi tiền không?*" -> Chọn "*Yes*" -> Xuất "*Thông tin gửi tiền*"
- 3) Kiểm tra số tiền
- 4) Kiểm tra người đăng ký đã nhập đầy đủ thông tin Quan trọng (\*) hay chưa?
- 5) Kiểm tra lỗi chính tả

<Thông tin ngân hàng>

(Tên, địa chỉ, chi nhánh,..)

Phiếu gửi tiền

Mã Số: <Hệ thống tự lập>

Thông tin người gửi:

Họ tên (\*): <Nhập từ bàn phím>

CMND (\*): <Nhập từ bàn phím>

Đóng

(1)

(2)

Tìm kiếm

\*Hiện thị



Thông tin hợp lệ!

Thông tin Số tiết kiệm:

(3)

Họ tên: <Hệ thống hiển thị>

Địa chỉ: < Hệ thống hiển thị > (Không hiển thị thông tin cụ thể, chỉ hiển thị tên T.phố- Tỉnh - VN)

Số tiền hiện tại: <Hệ thống hiển thị>

Loại tiết kiệm; <Hệ thống hiển thị>

Bạn có muốn gửi tiền không?

☐ Yes

☐ No

(Chỉ hiển thị khi Loại tiết kiệm là Không thời hạn)

Đóng

(5)

(4)

(6)

Xác nhận

\*Hiện thị



Xác nhận gửi tiền

Thông tin gửi tiền:

(7) Số tiền gửi (\*): <Nhập từ bàn phím>

(Có dấu ‘.’ Để phân chia đơn vị tiền – Nghìn, triệu, tỉ, trăm nghìn tỉ, ...>)

(Kiểm tra QĐ1 – Nếu trên 100k -> Hiện dòng Số tiền hợp lệ)

(8) Ngày gửi (\*): <Ngày mặc định> hoặc <Bảng chọn dd/mm/yyyy> (Người gửi muốn chọn)

(\*) Bắt buộc

Đóng

(9)

(10)

Xác nhận

Thiết kế xử lý:

Bước 1: Bổ sung các nút xử lý trên từng màn hình giao diện đã thiết kế

Không cần bổ sung

Bước 2: Lập danh sách các xử lý trên mỗi màn hình.

STT	Tên xử lý	Ý nghĩa	Mô tả	Điều kiện gọi	Ghi chú
1	XL1	Đóng bảng tra cứu thông tin	Đóng bảng tra cứu thông tin	Khi form được load lên	
2	XL2	Tra cứu thông tin	Tra cứu thông tin. Nếu thông tin hợp lệ hiện Thông tin Sổ tiết kiệm	Khi form được load lên	
3	XL3	Hiện thị bảng Thông tin Sổ tiết kiệm	Hệ thống hiện thị Thông tin Sổ tiết kiệm	Sau khi XL2 thành công	
4	XL4	Hỏi người dùng có muốn gọi tiền hay không	Hệ thống hiện dòng chữ như trên và cho phép người dùng chọn 1 trong 2 (“Yes” or “No”)	Sau khi XL2 thành công và “Loại tiết kiệm” là “Không thời hạn”	Nếu “Loại tiết kiệm” là “Không thời hạn” thì mới hiển thị
5	XL5	Đóng bảng Thông tin Sổ tiết kiệm	Đóng bảng Thông tin Sổ tiết kiệm	Sau khi XL2 thành công	
6	XL6	Xác nhận thao tác tùy thuộc vào XL4	Nếu người dùng chọn “Yes”, hiển thị Thông tin Gởi tiền. Các trường hợp còn lại đóng bảng và kết thúc form Phiếu gởi tiền	Sau khi XL2 thành công	
7	XL7	Nhập số tiền	Nhập số tiền và tự động phân đơn vị nghìn, triệu, tỉ	Sau khi XL6 thành công	
8	XL8	Lấy ngày hệ thống hoặc chọn ngày	Lấy ngày hệ thống hoặc chọn ngày	Sau khi XL6 thành công	
9	XL9	Đóng bảng Thông tin Gởi tiền	Đóng bảng Thông tin Gởi tiền	Sau khi XL6 thành công	
10	XL10	Xác nhận Gởi tiền	Xác nhận Gởi tiền và kết thúc form Phiếu gởi tiền	Sau khi XL6 thành công	

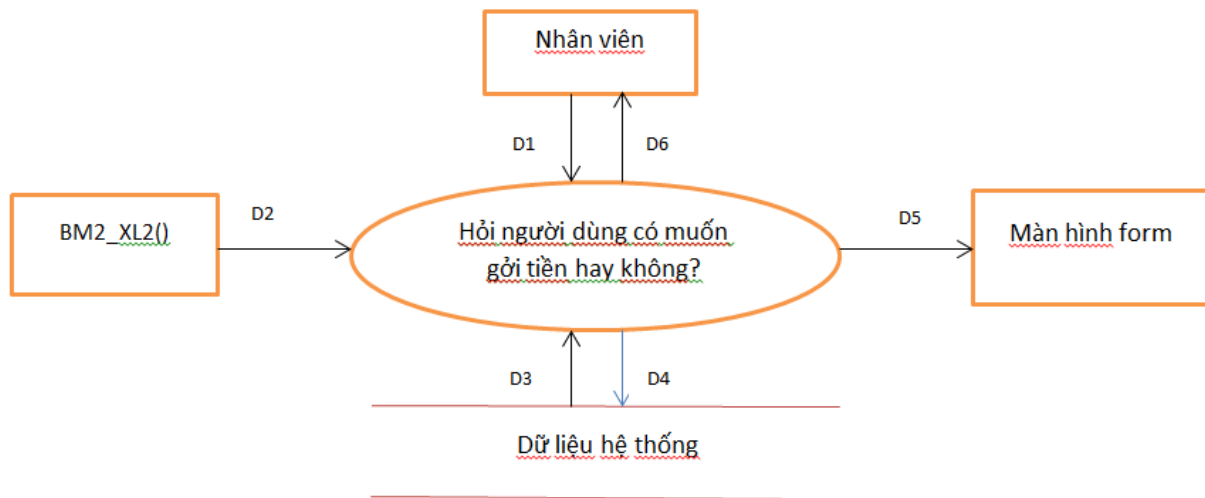
Hàm xử lý thường dùng:

<b>Tên xử lý</b>	<b>Tên hàm</b>	<b>Mô tả</b>	<b>Ghi chú</b>
XL1	BM2_XL1()	Đóng bảng tra cứu thông tin	
XL2	BM2_XL2()	Tra cứu thông tin. Nếu thông tin hợp lệ, trả về 1 xuất bảng Thông tin Sổ tiết kiệm. Không hợp lệ trả về 0	
XL3	BM2_XL3(CheckXL2)	Nếu CheckXL2=1: Hiển thị Thông tin Sổ tiết kiệm sau khi Tra cứu thông tin hợp lệ	Tham số "CheckXL2": Giá trị trả về của XL2
XL4	BM2_XL4(CheckXL2, CheckLTK)	Nếu CheckXL2=1: Hiển thị dòng chữ như trên nếu CheckLTK=1. Nếu chọn "Yes" sẽ trả về 1. Nếu "No" sẽ trả về 0	Tham số "CheckXL2": Giá trị trả về của XL2 Tham số "CheckLTK": Giá trị trả về của ("Loại tiết kiệm" có là "Không thời hạn"?)
XL5	BM2_XL5(CheckXL2)	Nếu CheckXL2=1: Hiển thị button. Đóng bảng Thông tin Sổ tiết kiệm	Tham số "CheckXL2": Giá trị trả về của XL2
XL6	BM2_XL6(CheckXL2, CheckXL4)	Nếu CheckXL2=1: Hiển thị button. Đóng bảng Thông tin Sổ tiết kiệm. Nếu CheckXL4=1 (Ứng với "Yes"), hiện bảng Thông tin gửi tiền và trả về 1. CheckXL4=0 (Ứng với "No"), đóng bảng Thông tin Sổ tiết kiệm và kết thúc form Phiếu gửi tiền. CheckXL4=null (Không thực hiện XL4) thì tương tự CheckXL4=0	Tham số "CheckXL2": Giá trị trả về của XL2 Tham số "CheckXL4": Giá trị trả về của XL4
XL7	BM2_XL7(CheckXL6)	Nếu CheckXL6=1: Tạo bảng cho người dùng nhập số tiền. Tự động phân chia đơn vị tiền nghìn, triệu, tỉ,...	Tham số "CheckXL6: Giá trị trả về của XL6
XL8	BM2_XL8(CheckXL6)	Nếu CheckXL6=1: Tạo 1 bảng chọn dd/mm/yyyy. Tự động chọn Ngày hệ thống và người dùng có thể thay đổi ngày bằng chọn dd/mm/yyyy	Tham số "CheckXL6: Giá trị trả về của XL6
XL9	BM2_XL9(CheckXL6)	Nếu CheckXL6=1: Hiển thị button. Kết thúc form Phiếu gửi tiền	Tham số "CheckXL6: Giá trị trả về của XL6
XL10	BM2_XL10(CheckXL6)	Nếu CheckXL6=1: Hiển thị button. Xác nhận gửi tiền và kết thúc form Phiếu gửi tiền	Tham số "CheckXL6: Giá trị trả về của XL6



**Bước 3:** Mô tả chi tiết một số xử lý phức tạp, quan trọng:

XL4 (Xử lý 4):



- Mô tả luồng dữ liệu:

D1: Lựa chọn “Yes” hoặc “No” của người dùng

D2: Giá trị trả về của BM2\_XL2()

D3: Thông tin Số tiết kiệm

D4: Không có

D5: Hiển thị “Bạn có muốn gọi tiền không?” lên màn hình form hoặc không hiển thị gì cả

D6: Hiển thị Tick vào ô tương đương với D1

- Thuật toán:

B1: Nhận D1 từ Nhân viên

B2: Nhận D2 từ hàm BM2\_XL2()

B3: Kết nối cơ sở dữ liệu

B4: Đọc D3 từ Dữ liệu hệ thống có dữ liệu trùng khớp với D2

B5: Kiểm tra thông tin Loại tiết kiệm

B6: Xuất D5 lên form ở bảng Thành phố nếu Loại tiết kiệm là “Không thời hạn”. Nếu Loại tiết kiệm không phải “Không thời hạn” thì không hiển thị gì. Tới B8

B7: Hiển thị Tick vào ô “Yes” hay “No” tương đương với thông tin D1

B8: Đóng kết nối cơ sở dữ liệu

**Bước 4:** Rà soát các yêu cầu để cải tiến thuật giải nếu cần.

### 5.3. Biểu mẫu 3 – Quy định 3:

<Thông tin ngân hàng>			
(Tên, địa chỉ, chi nhánh,...)	<b>Phiếu rút tiền</b>	Mã Số: <Hệ thống tự lập>	

**Thông tin người rút :**

Họ tên (\*): <Nhập từ bàn phím>

CMND (\*): <Nhập từ bàn phím>

Đóng

(1)

(2)

Tìm kiếm

\*Hiện thị

Thông tin hợp lệ!

**Thông tin Sổ tiết kiệm:** (3)

Họ tên: <Hệ thống hiển thị>

Địa chỉ: < Hệ thống hiển thị > (Không hiển thị thông tin cụ thể, chỉ hiển thị tên T.phố - Tỉnh - VN)

Số tiền hiện tại: <Hệ thống hiển thị>

Loại tiết kiệm; <Hệ thống hiển thị>

Ngày mở sổ : <Hệ thống hiển thị>

Bạn có muốn rút tiền không? Yes (4) No (Chiển thị khi đủ điều kiện rút )

Đóng

(5)

(6)

Xác nhận

\*Hiện thị

Xác nhận rút tiền

**Thông tin rút tiền:**

(7) Số tiền rút (\*): <Nhập từ bàn phím>

(Có dấu ‘.’ Để phân chia đơn vị tiền – Nghìn, triệu,ti, trăm nghìn ti,...>)

Nếu số dư còn lại = 0 => đóng sổ

(8) Ngày rút (\*): <Ngày mặc định> hoặc <Bảng chọn dd/mm/yyyy> (Người gọi muốn chọn)

Đóng

(9)

(10)

Xác nhận

Thiết kế xử lý:

**Bước 1:** Bổ sung các nút xử lý trên từng màn hình giao diện đã thiết kế

Không cần bổ sung

**Bước 2:** Lập danh sách các xử lý trên mỗi màn hình.

STT	Tên xử lý	Ý nghĩa	Mô tả	Điều kiện gọi	Ghi chú
1	XL1	Đóng bảng tra cứu thông tin	Đóng bảng tra cứu thông tin	Khi form được load lên	
2	XL2	Tra cứu thông tin	Tra cứu thông tin. Nếu thông tin hợp lệ hiện Thông tin Sổ tiết kiệm	Khi form được load lên	
3	XL3	Hiện thị bảng Thông tin Sổ tiết kiệm	Hệ thống hiển thị Thông tin Sổ tiết kiệm	Sau khi XL2 thành công	
4	XL4	Hỏi người dùng có muốn gửi tiền hay không	Hệ thống hiện dòng chữ như trên và cho phép người dùng chọn 1 trong 2 (“Yes” or “No”)	Sau khi XL2 thành công và “Loại tiết kiệm” là “Không thời hạn”	Nếu đủ thời gian của mỗi loại kì hạn thì mới hiển thị
5	XL5	Đóng bảng Thông tin Sổ tiết kiệm	Đóng bảng Thông tin Sổ tiết kiệm	Sau khi XL2 thành công	
6	XL6	Xác nhận thao tác tùy thuộc vào XL4	Nếu người dùng chọn “Yes”, hiển thị Thông tin Rút tiền. Các trường hợp còn lại đóng bảng và kết thúc form Phiếu rút tiền	Sau khi XL2 thành công	
7	XL7	Nhập số tiền	Nhập số tiền và tự động phân đơn vị nghìn, triệu, tỉ	Sau khi XL6 thành công	
8	XL8	Lấy ngày hệ thống hoặc chọn ngày	Lấy ngày hệ thống hoặc chọn ngày	Sau khi XL6 thành công	
9	XL9	Đóng bảng Thông tin Rút tiền	Đóng bảng Thông tin rút tiền	Sau khi XL6 thành công	
10	XL10	Xác nhận Rút tiền	Xác nhận Rút tiền và kết thúc form Phiếu Rút tiền	Sau khi XL6 thành công	

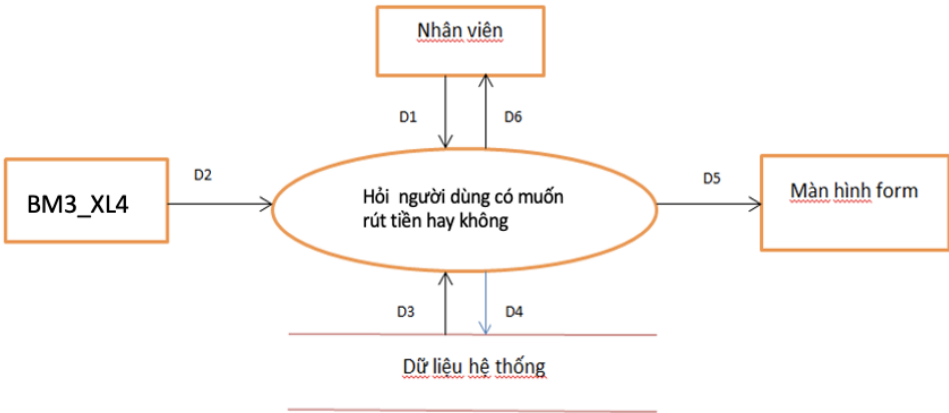
Hàm xử lý thường dùng:

Tên xử lý	Tên hàm	Mô tả	Ghi chú
XL1	BM3_XL1()	Đóng bảng tra cứu thông tin	
XL2	BM3_XL2()	Tra cứu thông tin. Nếu thông tin hợp lệ, trả về 1 xuất bảng Thông tin Sổ tiết kiệm. Không hợp lệ trả về 0	
XL3	BM3_XL3(CheckXL2)	Nếu CheckXL2=1: Hiện thị Thông tin Sổ tiết kiệm sau khi Tra cứu thông tin hợp lệ	Tham số “CheckXL2”: Giá trị trả về của XL2
XL4	BM3_XL4(CheckXL2,	Nếu CheckXL2=1: Hiện thị dòng chữ như trên nếu	Tham số “CheckXL2”: Giá

	CheckLTK)	CheckLTK=1. Nếu chọn “Yes” sẽ trả về 1. Nếu “No” sẽ trả về 0	trị trả về của XL2 Tham số “CheckLTK”: Giá trị trả về của (“Loại tiết kiệm” có là “Không thời hạn”?)
XL5	BM3_XL5(CheckXL2)	Nếu CheckXL2=1: Hiển thị button. Đóng bảng Thông tin Sổ tiết kiệm	Tham số “CheckXL2”: Giá trị trả về của XL2
XL6	BM3_XL6(CheckXL2, CheckXL4)	Nếu CheckXL2=1: Hiển thị button. Đóng bảng Thông tin Sổ tiết kiệm . Nếu CheckXL4=1 (Ứng với “Yes”), hiện bảng Thông tin rút tiền và trả về 1. CheckXL4=0 (Ứng với “No”), đóng bảng Thông tin Sổ tiết kiệm và kết thúc form Phiếu rút tiền. CheckXL4=null (Không thực hiện XL4) thì tương tự CheckXL4=0	Tham số “CheckXL2”: Giá trị trả về của XL2 Tham số “CheckXL4”: Giá trị trả về của XL4
XL7	BM3_XL7(CheckXL6)	Nếu CheckXL6=1: Tạo bảng cho người dùng nhập số tiền. Tự động phân chia đơn vị tiền nghìn, triệu, tỉ,...	Tham số “CheckXL6: Giá trị trả về của XL6
XL8	BM3_XL8(CheckXL6)	Nếu CheckXL6=1: Tạo 1 bảng chọn dd/mm/yyyy. Tự động chọn Ngày hệ thống và người dùng có thể thay đổi ngày bằng chọn dd/mm/yyyy	Tham số “CheckXL6: Giá trị trả về của XL6
XL9	BM3_XL9(CheckXL6)	Nếu CheckXL6=1: Hiển thị button. Kết thúc form Phiếu rút tiền	Tham số “CheckXL6: Giá trị trả về của XL6
XL10	BM3_XL10(CheckXL6)	Nếu CheckXL6=1: Hiển thị button. Xác nhận gửi tiền và kết thúc form Phiếu rút tiền	Tham số “CheckXL6: Giá trị trả về của XL6

**Bước 3:** Mô tả chi tiết một số xử lý phức tạp, quan trọng:

XL4 (Xử lý 4):



- Mô tả luồng dữ liệu:

D1: Lựa chọn “Yes” hoặc “No” của người dùng

D2: Giá trị trả về của BM3\_XL44()

D3: Thông tin Sổ tiết kiệm

D4: Không có

D5: Hiển thị “Bạn có muốn rút tiền không?” lên màn hình form hoặc không hiển thị gì cả

D6: Hiển thị Tick vào ô tương đương với D1

- Thuật toán:

B1: Nhận D1 từ Nhân viên

B2: Nhận D2 từ hàm BM3\_XL44()

B3: Kết nối cơ sở dữ liệu

B4: Đọc D3 từ Dữ liệu hệ thống có dữ liệu trùng khớp với D2

B5: Kiểm tra thông tin Loại tiết kiệm

B6: Xuất D5 lên form ở bảng Thành phố nếu Loại tiết kiệm là “Không thời hạn”. Nếu Loại tiết kiệm không phải “Không thời hạn” thì không hiển thị gì. Tới B8

B7: Hiển thị Tick vào ô “Yes” hay “No” tương đương với thông tin D1

B8: Đóng kết nối cơ sở dữ liệu

#### 5.4. Biểu mẫu 4 – Quy định 4:

<Thông tin ngân hàng>		Mã Số: <Hệ thống tự lập>				
(Tên, địa chỉ, chi nhánh,...)		<b>Danh sách khách hàng</b>				
Lọc :	(1)	<div>Loại tiết kiệm ▼</div>	(2)	<div>Số dư ▼</div>	(3)	<div>Từ ngày ▼</div>
Tìm kiếm :		<div>Họ và Tên :&lt;Nhập từ bàn phím &gt;</div>				
(4)	STT	Mã Số	Loại Tiết Kiệm	Khách Hàng	Số Dư	
	1					
	2					
Ngày xuất phiếu :		< Ngày từ hệ thống >				
<div>Đóng</div>		(5)	(6)		<div>Xác nhận</div>	

## Thiết kế xử lý:

**Bước 1:** Bổ sung các nút xử lý trên từng màn hình giao diện đã thiết kế

**Bước 2:** Lập danh sách các xử lý trên mỗi màn hình.

STT	Tên xử lý	Ý nghĩa	Mô tả	Điều kiện gọi	Ghi chú
1	XL1	Chọn loại tiết kiệm	Chọn 1 hoặc nhiều loại tiết kiệm	Khi form được load lên	
2	XL2	Chọn khoảng số dư	Chọn 1 khoảng số dư mong muốn	Khi form được load lên	
3	XL3	Chọn 1 ngày mở sổ	Chọn 1 ngày mở sổ	Khi form được load lên	
4	XL4	Hiện thị bảng khách hàng	Hệ thống hiện thị bảng khách hàng	Khi form được load lên	
5	XL5	Đóng bảng Thông tin khách hàng	Đóng bảng Thông tin khách hàng	Khi form được load lên	
6	XL6	Xác nhận danh sách	Xác nhận và kết thúc phiếu danh sách	Khi form được load lên	

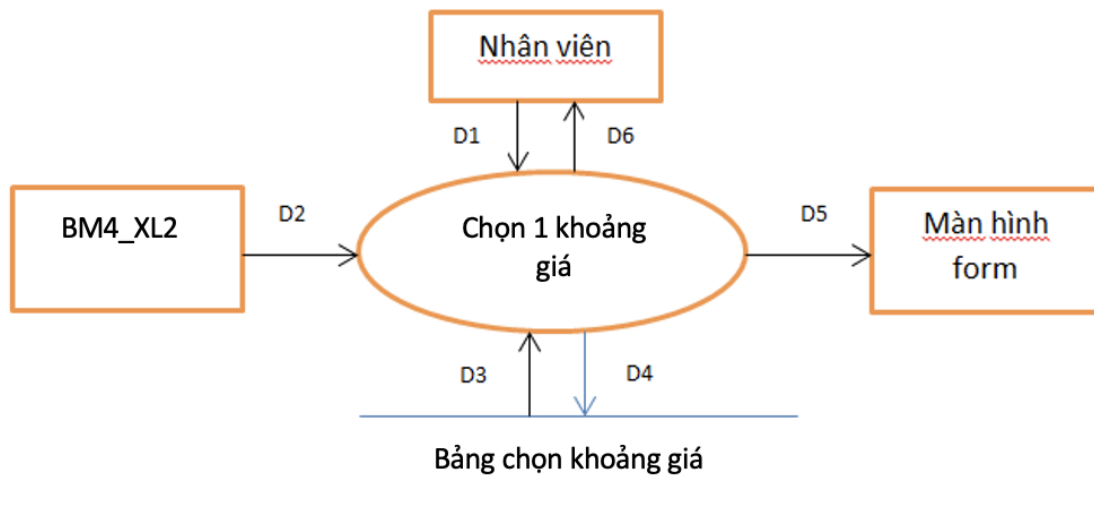
Hàm xử lý tương ứng:

Tên xử lý	Tên hàm	Mô tả	Ghi chú
XL1	BM4_XL1()	Tạo bảng chọn loại tiết kiệm Người dùng có thể nhập vài chữ cái và hiện tính liên quan nhất	
XL2	BM4_XL2	Tạo bảng chọn khoảng số dư , có thể cho nhập khoảng từ bàn phím	
XL3	BM4_XL3	Tạo bảng chọn dd/mm/yyyy.	
XL4	BM4_XL4()	Hiện thị danh sách khách hàng thỏa mãn yêu cầu XL1,2,3	
XL5	BM4_XL5()	Lệnh đóng bảng	
XL6	BM4_XL6()	Lệnh xác nhận cho phép xuất ra file bảng hoặc chuyển đến lệnh in	



**Bước 3:** Mô tả chi tiết một số xử lý phức tạp, quan trọng:

XL2 (Xử lý 2):



- Mô tả luồng dữ liệu:

D1: Các ký tự Nhân viên nhập để tìm kiếm khoảng giá

D2: khoảng giá đã được chọn ở hàm BM4\_XL2()

D3: Danh sách khoảng giá trong hàm BM4\_XL2()

D4: Không có

D5: Khoảng giá liên quan đến D1

D6: D5

- Thuật toán:

B1: Nhận D1 từ Nhân viên

B2: Nhận D2 từ hàm BM4\_XL2

B3: Kết nối cơ sở dữ liệu

B4: Đọc D3 từ Bảng chọn khoảng giá

B5: Hiển thị các khoảng giá (D3) liên quan nhất với các ký tự nhập vào (D1) lên bảng chọn

B6: Xuất D5 lên form ở bảng khoảng giá.

B7: Đóng kết nối cơ sở dữ liệu

### 5.5. Biểu mẫu 5:

<b>BM5.1</b>	<b>Báo Cáo Doanh Số Hoạt Động Ngày</b>			
Ngày:.....				
<b>STT</b>	<b>Loại Tiết Kiệm</b>	<b>Tổng Thu</b>	<b>Tổng Chi</b>	<b>Chênh Lệch</b>
1				
2				

<b>BM5.2</b>	<b>Báo Cáo Mở/Đóng Sổ Tháng</b>			
Loại tiết kiệm: .....		Tháng:.....		
<b>STT</b>	<b>Ngày</b>	<b>Số Mở</b>	<b>Số Đóng</b>	<b>Chênh Lệch</b>
1				
2				

Báo cáo doanh số

— □ ×

**BÁO CÁO DOANH SỐ HOẠT ĐỘNG NGÀY**

NGÀY  (1)

STT	Loại tiết kiệm	Tổng thu	Tổng chi	Chênh lệch
1	Không kì hạn			
2	hạn 3 tháng	(2)		(3)
3	hạn 6 tháng			
<b>Tổng cộng</b>				

Export (4)

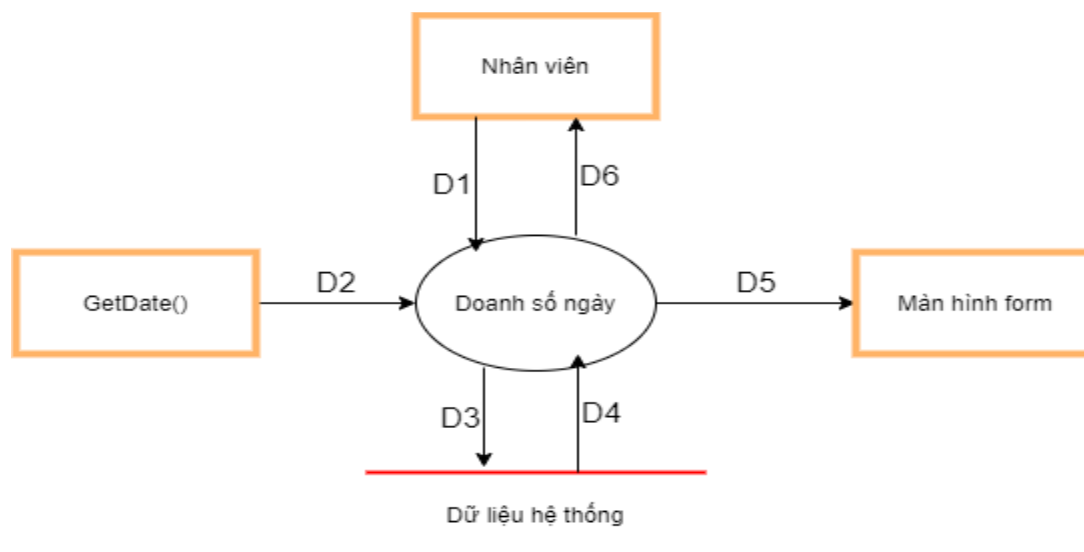
Đóng (5)

In (6)

STT	Tên XL	Ý nghĩa	Mô tả	Điều kiện gọi	Ghi chú
1	XL 1	Chọn ngày muốn xem báo cáo, mặc định ngày hệ thống.	Chọn ngày muốn xem báo cáo	Khi form đc load lên hoặc được người dùng chọn/nhập ngày mong muốn.	
2	XL 2	Hiển thị doanh thu	Tính toán và hiển thị tổng thu và tổng chi theo ngày	Khi ngày ở XL 1 hợp lệ	
3	XL 3	Hiển thị chênh lệch giữa tổng thu và tổng chi	Tính toán và hiển thị chênh lệch giữa tổng thu và tổng chi	Khi có dữ liệu ở XL 2 hay dữ liệu ở XL 2 bị thay đổi.	
4	XL 4	Xuất dữ liệu	Xuất dữ liệu dưới dạng file phần mềm như .xlsx, .doc, .pdf, ...	Khi người dùng click vào nút 'Export'	
5	XL 5	Thoát màn hình	Đóng cửa sổ báo cáo	Khi người dùng click vào nút 'Đóng'	
6	XL 6	In dữ liệu	In toàn bộ dữ liệu có trong cửa sổ	Khi người dùng click vào nút 'In'	Phải có máy in tích hợp

Mô tả chi tiết một số xử lý phức tạp, quan trọng:

- Xử lý 2: Tính tổng thu theo ngày



- Mô tả luồng dữ liệu:

D1: Không

D2: Giá trị trả về của GetDate()

D3: Ngày thống kê báo cáo

D4: Tổng thu nhập của từng loại trong ngày

D5: Hiển thị D4

D6: Không

- Thuật toán:

B1: Nhận giá trị trả về từ GetDate()

B2: Gửi ngày vừa nhận được đến cơ sở dữ liệu

B3: Cơ sở dữ liệu tổng hợp những số tiết kiệm trong ngày và gom nhóm theo từng loại tiết kiệm (không kỳ hạn, hạn 3 tháng, hạn 6 tháng)

B4: Tính tổng thu trong ngày và gom nhóm theo từng loại tiết kiệm

B5: Xuất kết quả tính toán vừa tìm đc ra màn hình tương ứng với từng loại tiết kiệm trong bảng

B6: Đóng kết nối cơ sở dữ liệu

Báo cáo đóng/mở sổ

BÁO CÁO MỞ/ĐÓNG SỔ THÁNG

Loại tiết kiệm

Chọn loại tiết kiệm

(7)

THÁNG

06/2021

(8)

STT	Ngày	Số sổ mở	Số sổ đóng	Chênh lệch
1	1/6			
2	2/6			
3	3/6			
4	4/6			

(11)

(9)

(10)

Tổng cộng

Tổng số mở:

Tổng số đóng:

Chênh lệch

(12)

Export

(13)

Đóng

(14)

In

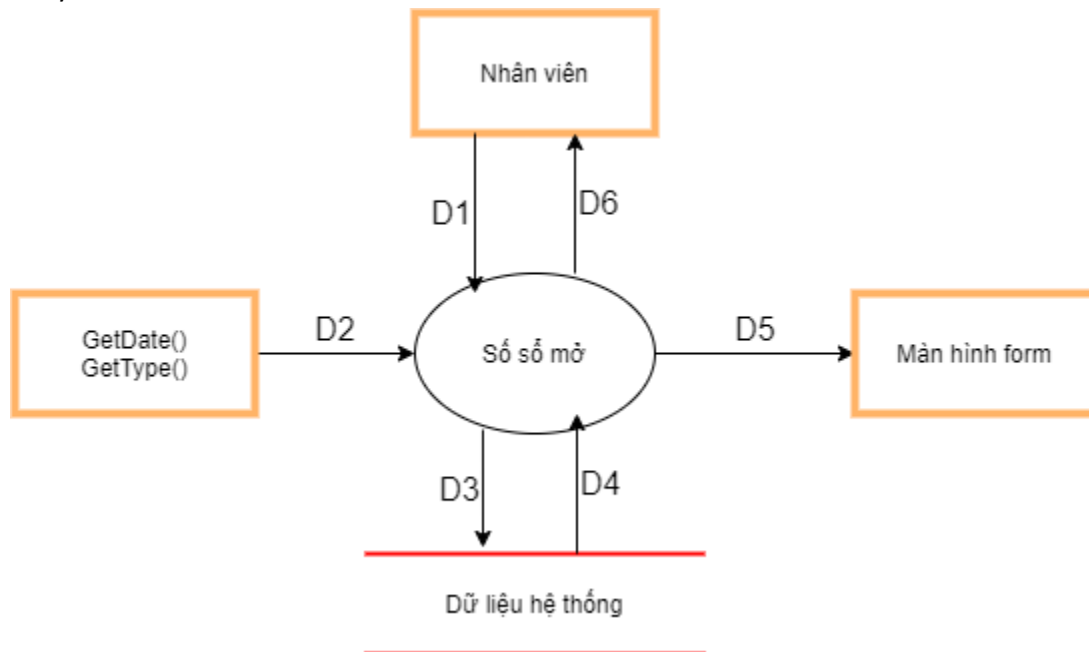
(15)

STT	Tên XL	Ý nghĩa	Mô tả	Điều kiện gọi	Ghi chú
1	XL 7	Chọn loại tiết kiệm	Chọn ngày loại tiết kiệm muốn được thống kê báo cáo	Khi form đc load lên hoặc được người dùng chọn/nhập loại tiết kiệm phù hợp	
2	XL 8	Chọn khoảng thời gian thống kê	Chọn tháng và năm để thống kê báo cáo	Khi form đc gọi lên hoặc được người dùng chọn/nhập tháng năm phù hợp.	
3	XL 9	Hiển thị số sổ mở và số sổ đóng	Tính toán và hiển thị số lượng sổ mở và số lượng sổ đóng của loại tiết kiệm ở XL 7 và trong tháng ở XL 8	Khi XL 7 và XL 8 hợp lệ. Khi XL 7 và XL 8 có thay đổi	Tùy tháng có thể có 28-31 dòng dữ liệu
4	XL 10	Hiển thị chênh lệch giữa số sổ mở và số sổ đóng	Tính toán và hiển thị chênh lệch giữa số sổ mở và số sổ đóng theo dòng	XL 9 có dữ liệu hợp lệ hay XL 9 có sự thay đổi	

5	XL 11	Cuộn chuột	Cuộn chuột để xem từng dòng dữ liệu trong bảng vì ko thể hiện thị hết trên màn hình	Khi người dùng lăn chuột trong bảng dữ liệu	
6	XL 12	Hiển thị thông tin tổng kết	Tính toán và hiển thị toàn bộ số sổ mở và số sổ đóng trong tháng cũng như chênh lệch	Khi có dữ liệu hợp lệ ở XL 9, XL 10 và XL 11.	
7	XL 13	Xuất dữ liệu	Xuất dữ liệu dưới dạng file phần mềm như .xlsx, .doc, .pdf, ...	Khi người dùng click vào nút 'Export'	
8	XL 14	Thoát màn hình	Đóng cửa sổ báo cáo	Khi người dùng click vào nút 'Đóng'	
9	XL 15	In dữ liệu	In toàn bộ dữ liệu có trong cửa sổ	Khi người dùng click vào nút 'In'	Phải có máy in tích hợp

Mô tả chi tiết một số xử lý phức tạp, quan trọng:

- Xử lý 9: Tính số sổ mở



- Mô tả luồng dữ liệu:

D1: Không

D2: Giá trị trả về của GetDate() và GetType()

D3: Tháng thống kê báo cáo, loại tiết kiệm

D4: Số sổ mở của loại tiết kiệm đang xét trong tháng, gom nhóm theo từng ngày

D5: Tổng số sổ mở của loại tiết kiệm đang xét trong tháng, gom nhóm theo từng ngày

D6: Không

- Thuật toán:

B1: Nhận giá trị trả về từ GetDate(), GetType()

B2: Gửi tháng, năm, loại tiết kiệm vừa nhận được đến cơ sở dữ liệu

B3: Cơ sở dữ liệu tổng hợp những sổ tiết kiệm có loại tiết kiệm là loại đang xét, được mở trong tháng, gom nhóm theo từng ngày

B4: Tính tổng số tiết kiệm có loại tiết kiệm là loại đang xét, được mở trong tháng, gom nhóm theo từng ngày.

B5: Xuất kết quả tính toán vừa tìm được ra màn hình tương ứng với từng ngày trong bảng.

B6: Đóng kết nối cơ sở dữ liệu

**Hàm xử lý tương ứng:**

Tên xử lý	Tên hàm	Mô tả	Ghi chú
XL 1	GetDate()	Lấy ngày ở trong text box. (mặc định ngày hệ thống)	
XL 2	prtRevenue()	Tính và in ra tổng thu theo từng loại tiết kiệm. Ở dòng tổng cộng là in ra tổng thu của cả 3 loại tiết kiệm trên	
	prtExpense()	Tính và in ra tổng chi tiêu theo từng loại tiết kiệm. Ở dòng tổng cộng là in ra tổng chi tiêu của cả 3 loại tiết kiệm trên	
XL 3	diffReEx()	Tính và in ra chênh lệch giữa tổng thu và tổng chi	
XL 4	Export()	Lưu lại dữ liệu dưới dạng các file mong muốn.	
XL 5	CloseWindow()	Đóng cửa sổ	
XL 6	Print()	In ra dữ liệu với máy in	
XL 7	getType()	Lấy ra loại tiết kiệm	
XL 8	GetDate()	Lấy ngày ở trong text box	
XL 9	countOpenAcc()	Tính và in ra tổng số sổ mở trong từng ngày.	
	countCloseAcc()	Tính và in ra tổng số sổ đóng trong từng ngày.	
XL 10	diffOpCl()	Tính và in ra chênh lệch giữa số sổ mở và số sổ đóng	
XL 11		Cuộn chuột	
XL 12	sumOpCl()	Tính tổng số sổ mở, số sổ đóng	
	diffOpCl()	Tính và in ra chênh lệch giữa số sổ mở và số sổ đóng	
XL 13	Export()	Lưu lại dữ liệu dưới dạng các file mong muốn.	
XL 14	CloseWindow()	Đóng cửa sổ	
XL 15	Print()	In ra dữ liệu với máy in	



## 5.6. Biểu mẫu 6 – Quy định 6:

**Loại kỳ hạn:**

(1) - Thêm loại kỳ hạn:

Tên: <Nhập> Lãi suất: <Nhập> Kỳ hạn: <Nhập>

Thêm mới

(2) - danh sách loại kỳ hạn: <Chọn loại kỳ hạn> Xóa kỳ hạn (6)

Tên: <Tự động> Lãi suất: <Tự động> Kỳ hạn: <Tự động>

Tên mới: <Nhập> Lãi suất mới: <Nhập> Kỳ hạn mới: <Nhập>

**Tiền gửi tối thiểu:**

(3) - Tiền gửi tối thiểu hiện tại: 100.000đ

- Tiền gửi tối thiểu mới: <Nhập>

**Thời hạn gửi tối thiểu của loại ko kì hạn:**

(4) - Thời hạn gửi hiện tại: 15 ngày

- Thời hạn gửi mới: <Nhập>

Xác nhận thay đổi (5) (7) Quay lại

STT	Tên XL	Ý nghĩa	Mô tả	Điều kiện gọi	Ghi chú
1	XL 1	Thêm loại kì hạn	Thêm loại kì hạn mới	Có dữ liệu hợp lệ ở cả 3 ô (tên, lãi suất, kỳ hạn) và người dùng bấm nút thêm mới	
2	XL 2	Thay đổi tham số của từng loại kì hạn	Thay đổi tên, lãi suất và kỳ hạn của từng loại kỳ hạn	Có dữ liệu hợp lệ ở 1 trong 3 ô (tên, lãi suất, kỳ hạn) và người dùng bấm nút xác nhận thay đổi	
3	XL 3	Thay đổi số tiền gửi thêm tối thiểu của	Thay đổi số tiền gửi thêm tối thiểu	Khi có dữ liệu hợp lệ ở ô Tiền gửi tối thiểu mới và người dùng bấm nút xác nhận thay đổi.	
4	XL 4	Thay đổi thời hạn gửi tối thiểu của loại không kỳ hạn	Thay đổi thời hạn gửi tối thiểu của loại không kỳ hạn. Tức sau 1 thời hạn nhất định mới đc rút	Khi có dữ liệu hợp lệ ở ô Thời hạn gửi mới và người dùng bấm nút xác nhận thay đổi.	

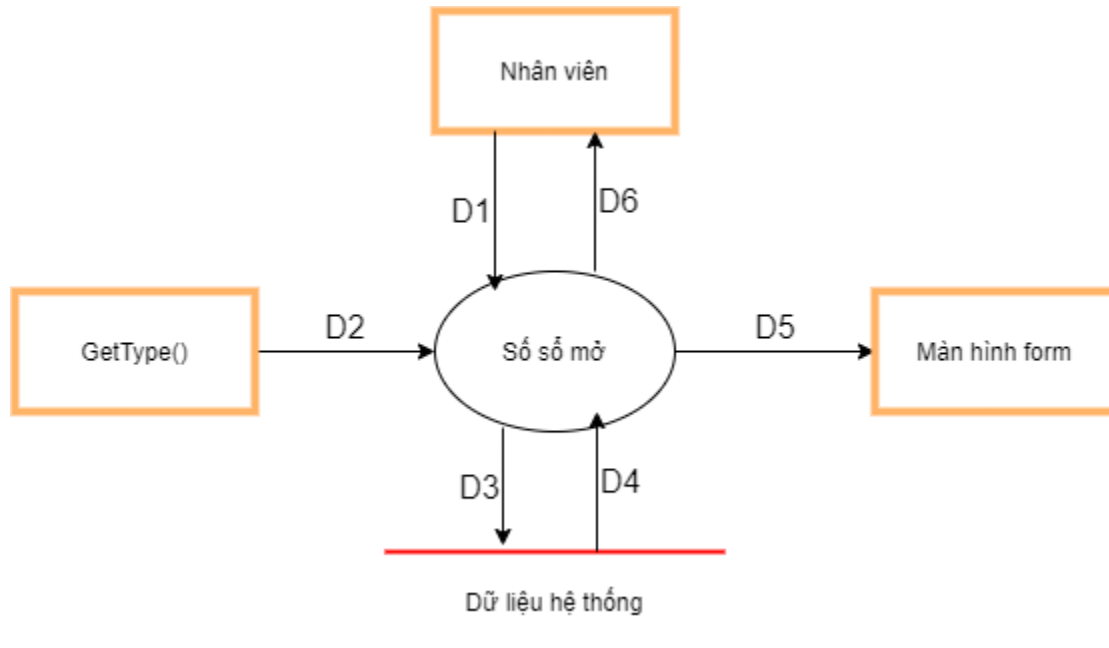
5	XL 5	Xác nhận thay đổi	Chấp nhận những thay đổi ở trên.	Khi người dùng bấm nút xác nhận thay đổi.	
6	XL 6	Xóa loại kỳ hạn	Xóa loại kỳ hạn khỏi hệ thống. Người dùng ko thể đăng ký loại kỳ hạn này nữa	Khi người dùng chọn loại kỳ hạn và bấm nút Xóa kỳ hạn	Dữ liệu về kỳ hạn cũ vẫn trong hệ thống

**Hàm xử lý tương ứng:**

Tên xử lý	Tên hàm	Mô tả	Ghi chú
XL 1	AddType()	Thêm loại kì hạn có các thuộc tính ở 3 ô (tên, lãi suất, kỳ hạn)	
XL 2	GetType()	Lấy ra loại tiết kiệm	
	ChangeParaType()	Thay đổi thuộc tính của loại kỳ hạn được chọn bằng thuộc tính ở 3 ô (tên mới, lãi suất mới, kỳ hạn mới)	Nếu ô nào bỏ trống tức thuộc tính đó được giữ nguyên
XL 3	ChangeMinDeposit()	Thay đổi số tiền gửi thêm tối thiểu bằng số tiền ở ô Tiền gửi tối thiểu mới	
XL 4	ChangeMinDuration()	Thay đổi thời hạn gửi tối thiểu của loại không kỳ hạn bằng thời hạn ở ô Thời hạn gửi mới	
XL 5	SaveChanges()	Xác nhận tất cả các thay đổi trong form	
XL 6	DelType()	Xóa loại kỳ hạn được chọn ở ô chọn loại kỳ hạn	
XL 7	Back()	Xác nhận ko thay đổi gì cả và đóng cửa sổ	

Mô tả chi tiết một số xử lý phức tạp, quan trọng:

- Xử lý 2: Đổi thuộc tính của các loại tiết kiệm



- Mô tả luồng dữ liệu:

D1: Nhân viên nhập vào tên mới, lãi suất mới, kỳ hạn mới

D2: Giá trị trả về của GetType()

D3: D1 và D2

D4: D1

D5: D1

D6: Không

- Thuật toán:

B1: Nhận giá trị trả về từ GetType()

B2: Nhân viên nhập vào tên mới, lãi suất mới, kỳ hạn mới

B3: Thay đổi các thuộc tính của loại tiết kiệm đang xét bằng tên mới, lãi suất mới, kỳ hạn mới

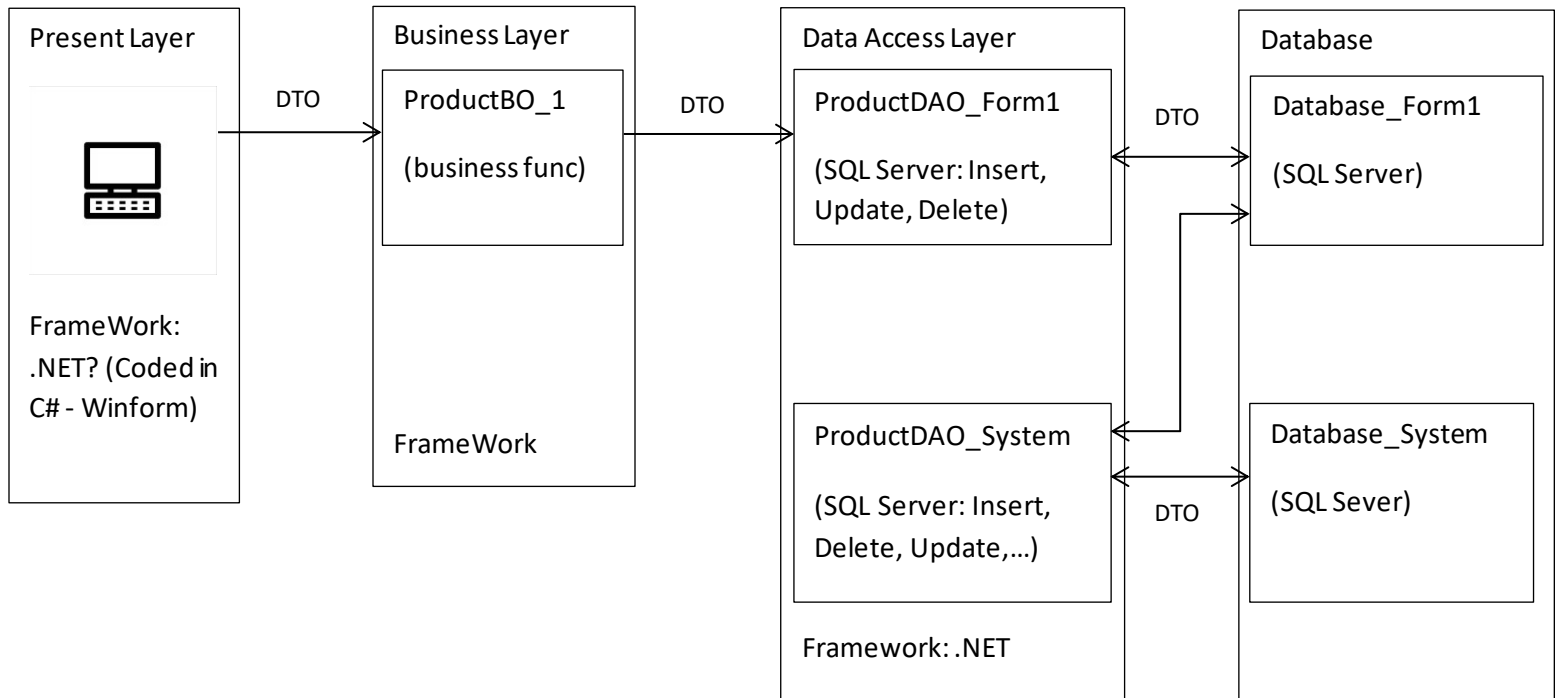
B4: Xuất ra màn hình tên, lãi suất, kỳ hạn sau khi được thay đổi.

B5: Đóng kết nối cơ sở dữ liệu

## 6. Thiết kế phần mềm:

### 6.2. *Application Architecture:*

#### Form1: Mở sổ tiết kiệm



DTO: Data Transfer Object

Database\_Form1: Lưu thông tin đăng ký của khách hàng

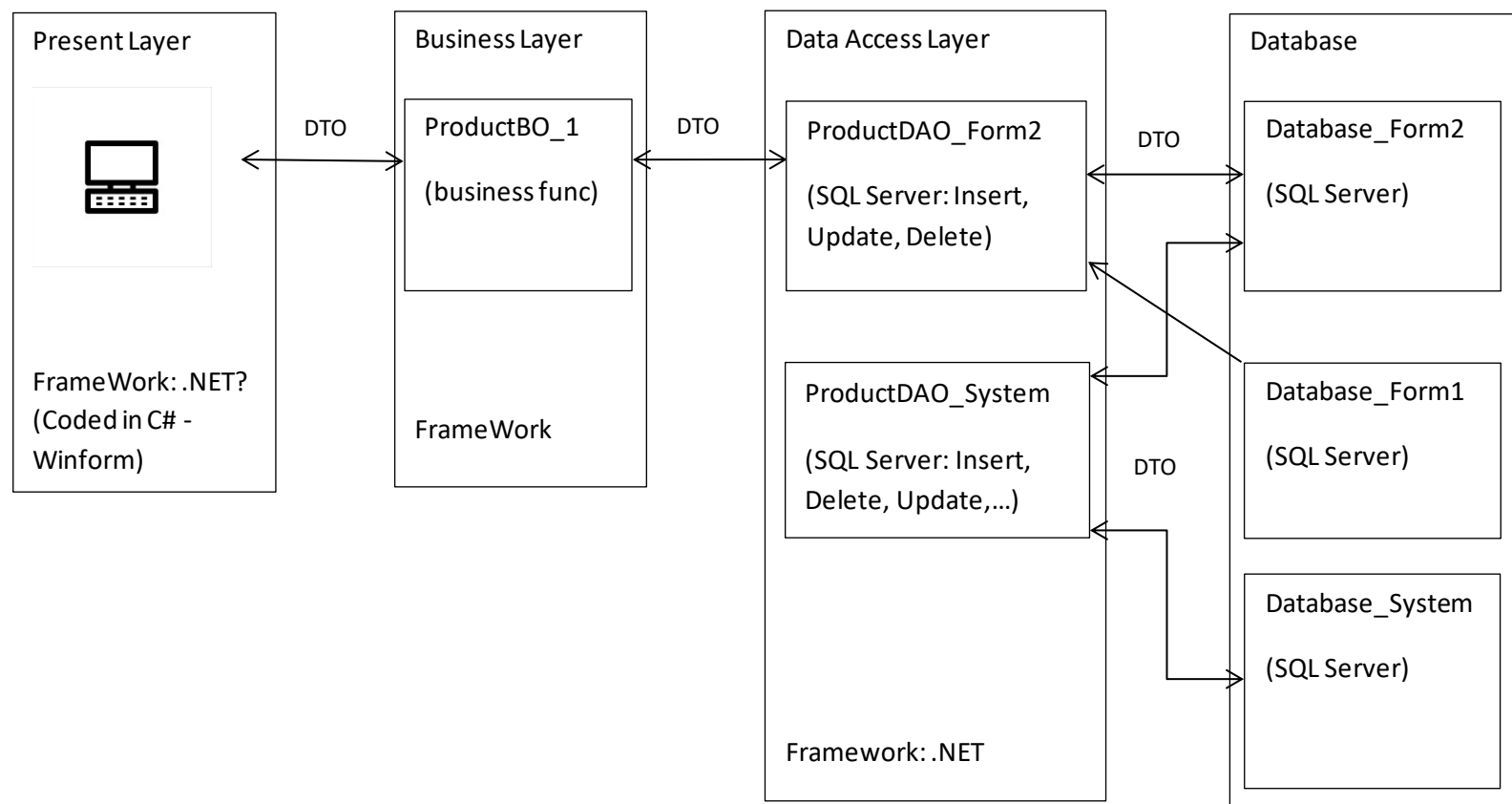
Database\_System: Database của hệ thống. Lưu toàn bộ thông tin mọi hoạt động của hệ thống

ProductDAO\_Form1: Thực hiện các thao tác thêm, sửa, xóa trên Database\_Form1

ProductDAO\_System: Thực hiện các thao tác chỉnh sửa, truy cập trên Database\_System và các Database khác

ProductBO\_1: Thực hiện các nghiệp vụ business logic trên ProductDAO\_1. Nhận thông tin và yêu cầu từ tầng Present Layer. Kiểm tra các yêu cầu đó. Nếu hợp lệ, gửi các yêu cầu về cho tầng Data Access Layer (ProductDAO\_Form1) để xử lý trên Database (Database\_Form1)

## Form2: Gởi tiết kiệm:



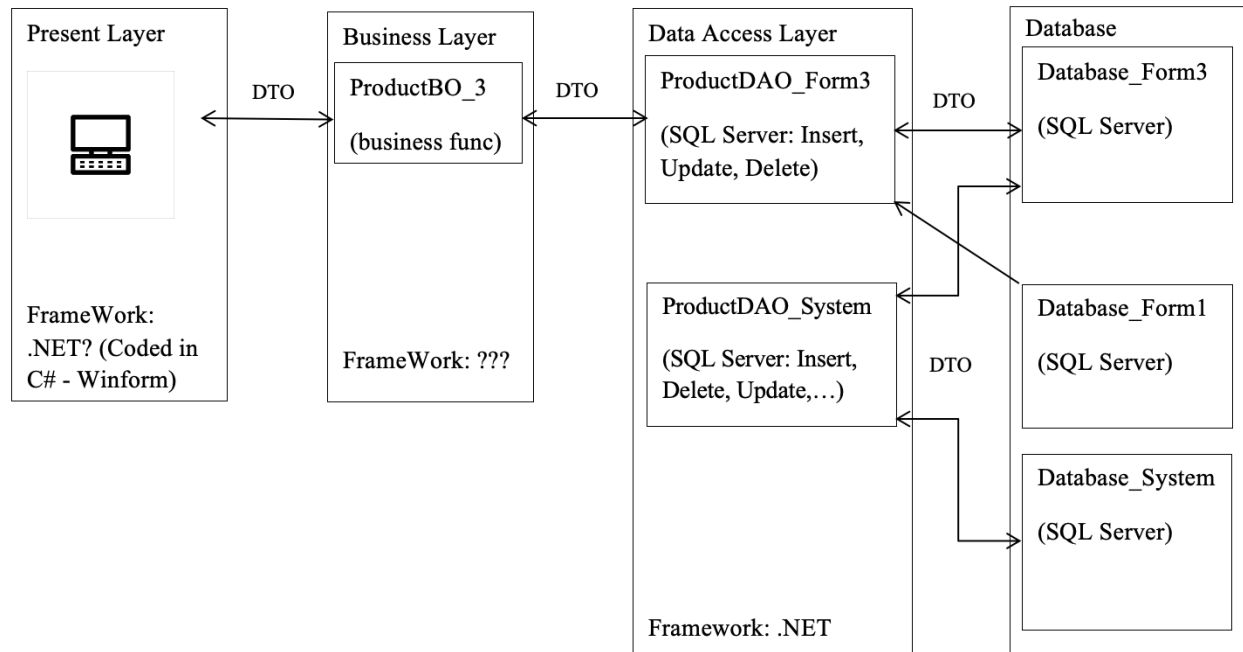
Database\_Form2: Lưu thông tin Gởi tiết kiệm của khách hàng

ProductDAO\_Form2: Thực hiện các thao tác thêm, sửa, xóa trên Database\_Form2

ProductBO\_5\_1: Thực hiện các nghiệp vụ business logic trên ProductDAO\_2. Nhận thông tin và yêu cầu từ tầng Present Layer. Kiểm tra các yêu cầu đó. Nếu hợp lệ, gửi các yêu cầu về cho tầng Data Access Layer (ProductDAO\_Form2) để xử lý trên Database (Database\_Form2) hoặc truy xuất dữ liệu từ Database\_Form1

\*Note: Ở đây, ProductDAO\_System không thực hiện thao tác gì trên Database\_Form1. Nhiệm vụ của ProductDAO\_System là truy cập Database của mỗi yêu cầu (Vd: Yêu cầu đăng ký Sổ tiết kiệm thì thao tác với Database\_Form1, yêu cầu Gởi tiết kiệm thì thao tác với Database\_Form2). Sau đó từ các thông tin từ Database\_FormX, cập nhật, bổ sung, xóa, sửa,... trên Database\_System. Ngược lại, nếu có xóa sửa trên Database\_System, cập nhật lại trên Database\_FormX. Mục đích là để truy xuất dữ liệu nhanh hơn. Vd ở đây, khi khách hàng thực hiện yêu cầu Gởi tiết kiệm, khách hàng sẽ xem thông tin Sổ tiết kiệm. Ta sẽ lấy thông tin Sổ tiết kiệm từ Database\_Form1 thay vì Database\_System để truy xuất dữ liệu nhanh hơn và hệ thống không bị quá tải.

### Form3 : rút tiền sổ tiết kiệm



DTO: Data Transfer Object

Database\_Form3: Lưu thông tin Rút tiết kiệm của khách hàng

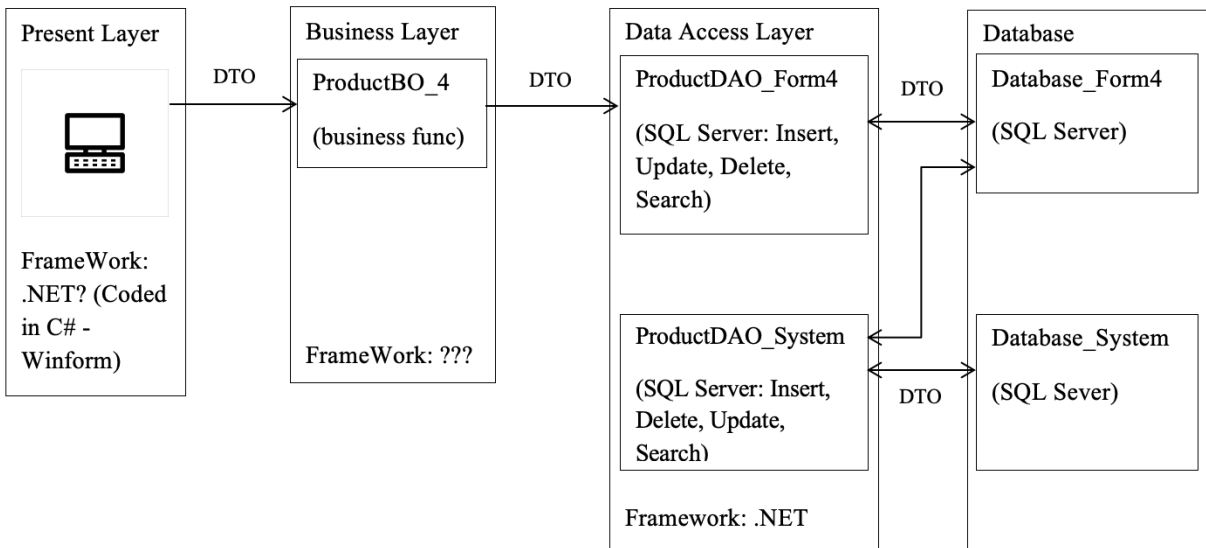
Database\_System: Database của hệ thống. Lưu toàn bộ thông tin mọi hoạt động của hệ thống

ProductDAO\_Form3: Thực hiện các thao tác thêm, sửa, xóa trên Database\_Form3

ProductBO\_3: Thực hiện các nghiệp vụ business logic trên ProductDAO\_3. Nhận thông tin và yêu cầu từ tầng Present Layer. Kiểm tra các yêu cầu đó. Nếu hợp lệ, gửi các yêu cầu về cho tầng Data Access Layer (ProductDAO\_Form3) để xử lý trên Database (Database\_Form3) hoặc truy xuất dữ liệu từ Database\_Form3

ProductDAO\_System không thực hiện thao tác gì trên Database\_Form3. Nhiệm vụ của ProductDAO\_System là truy cập Database của mỗi yêu cầu. Sau đó từ các thông tin từ Database\_FormX, cập nhật, bổ sung, xóa, sửa,... trên Database\_System. Ngược lại, nếu có xóa sửa trên Database\_System, cập nhật lại trên Database\_FormX. Mục đích là để truy xuất dữ liệu nhanh hơn.

#### Form 4 : Danh sách khách hàng



DTO: Data Transfer Object

Database\_Form4: Lưu thông tin danh sách của khách hàng

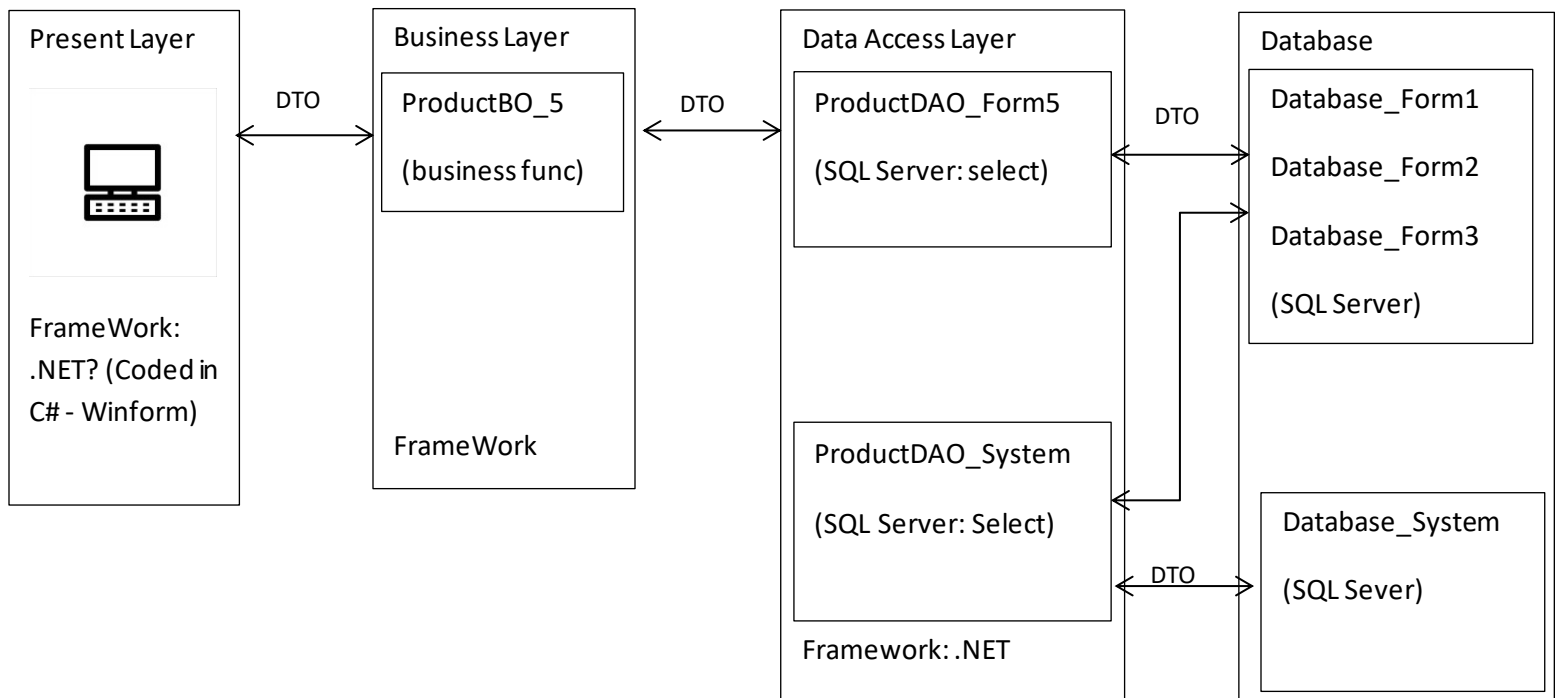
Database\_System: Database của hệ thống. Lưu toàn bộ thông tin mọi hoạt động của hệ thống

ProductDAO\_Form4: Thực hiện các thao tác thêm, sửa, xóa, tìm kiếm trên Database\_Form4

ProductDAO\_System: Thực hiện các thao tác chỉnh sửa, truy cập, tìm kiếm trên Database\_System và các Database khác

ProductBO\_4: Thực hiện các nghiệp vụ business logic trên ProductDAO\_Form4. Nhận thông tin và yêu cầu từ tầng Present Layer. Kiểm tra các yêu cầu đó. Nếu hợp lệ, gửi các yêu cầu về cho tầng Data Access Layer (ProductDAO\_Form4) để xử lý trên Database (Database\_Form4)

## Form 5.1 và 5.2: Báo cáo doanh số và báo cáo đóng mở sổ



DTO: Data Transfer Object

Database\_Form1: Lấy thông tin đăng ký của khách hàng.

Database\_Form2: Lấy thông tin gói tiết kiệm của khách hàng.

Database\_Form3: Lấy thông tin rút tiền của khách hàng.

Database\_System: Database của hệ thống. Yêu cầu toàn bộ thông tin mọi hoạt động của hệ thống

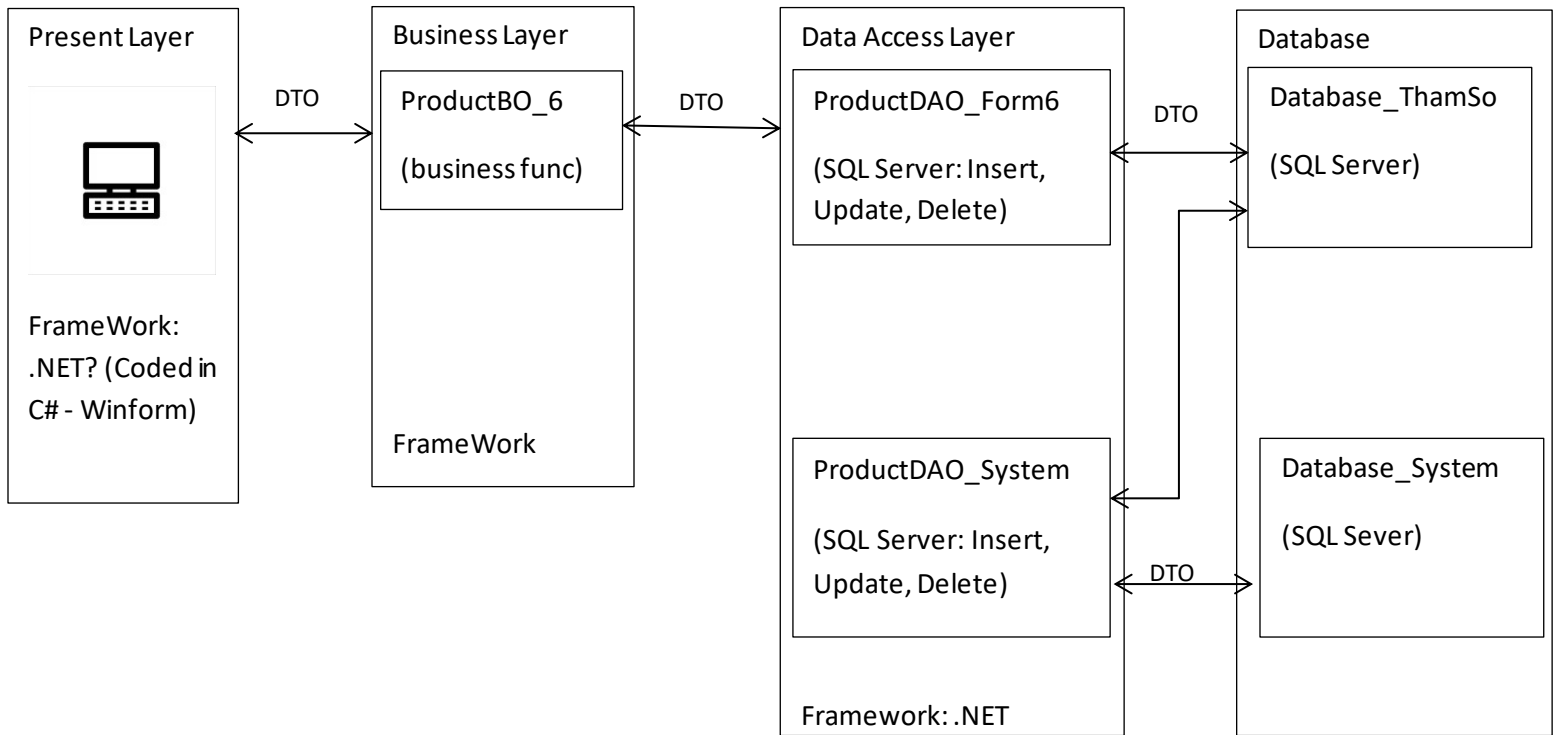
ProductDAO\_Form5: Thực hiện các thao tác search, select trên Database\_Form1, Database\_Form2, Database\_Form3.

ProductDAO\_System: Thực hiện các thao tác truy cập trên Database\_System và các Database khác.

ProductBO\_5: Thực hiện các nghiệp vụ business logic trên ProductDAO\_5. Đưa ra yêu cầu từ tầng present đến các tầng dưới và nhận thông tin trả về gồm các yêu cầu truy xuất database.



## Form 6: Thay đổi các tham số



DTO: Data Transfer Object

Database\_ThamSo: Chỉnh sửa các tham số

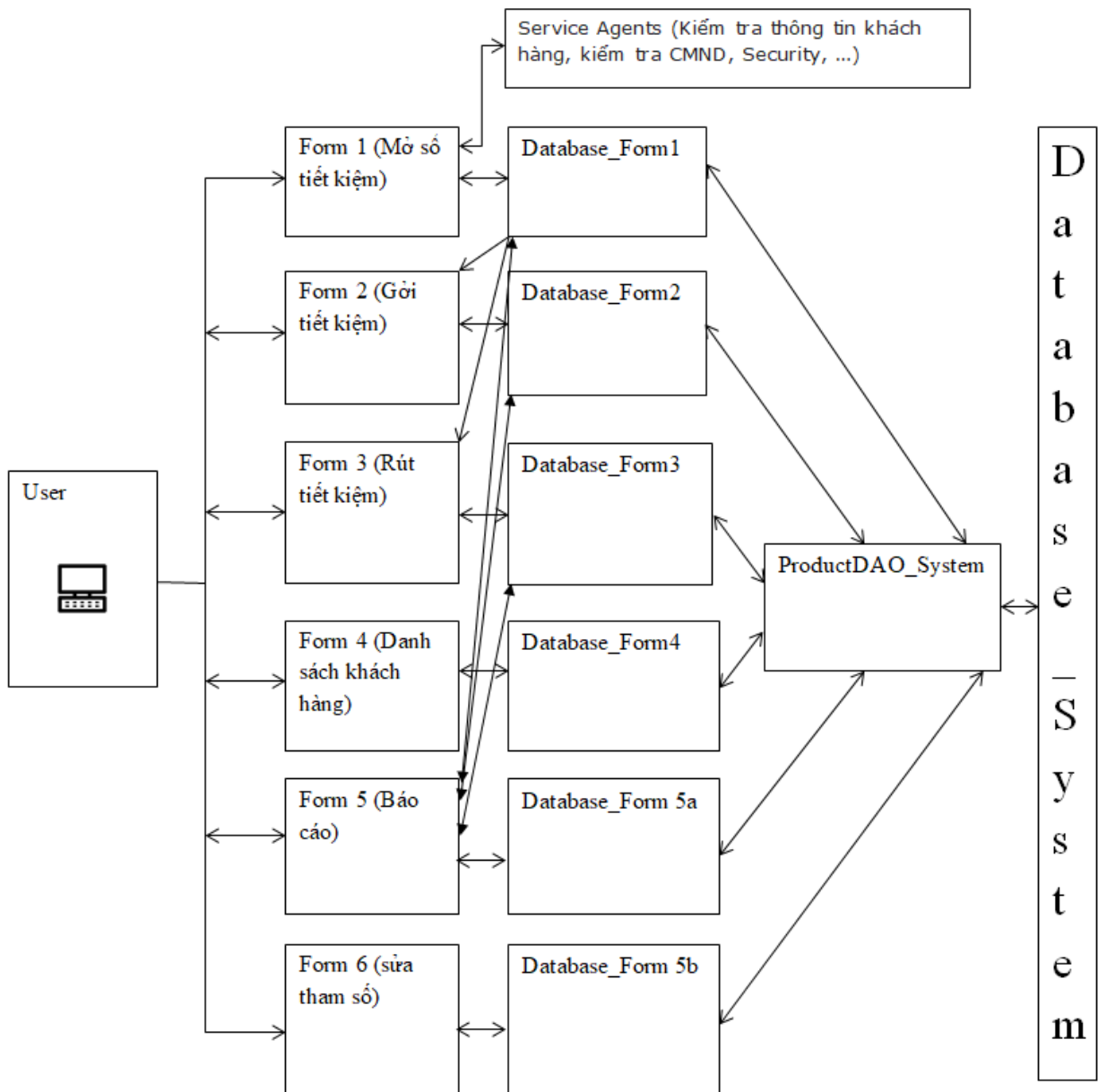
Database\_System: Database của hệ thống. Yêu cầu toàn bộ thông tin mọi hoạt động của hệ thống

ProductDAO\_Form 6: Thực hiện các thao tác Insert, Update, Delete trên Database\_ThamSo

ProductDAO\_System: Thực hiện các thao tác truy cập trên Database\_System và các Database khác.

ProductBO\_6: Thực hiện các nghiệp vụ business logic trên ProductDAO\_5. Đưa ra yêu cầu từ tầng present đến các tầng dưới và lấy thông tin từ các tầng dưới.

## 6.2. System Architecture:



## 7. Coding Convention:

### 7.1. Lập trình hướng đối tượng

- Nhóm đã chọn ngôn ngữ c# trên nền tảng .Net để viết mã, và ngôn ngữ này cũng hỗ trợ lập trình hướng đối tượng. Với hướng đối tượng, người dùng có thể dễ dàng quản lý code khi có sự thay đổi chương trình. Thay vì việc thay đổi các mã nguồn của đối tượng mà giờ đây lập trình viên chỉ cần thay đổi một số hàm thành phần. Ngoài ra lập trình hướng đối tượng còn có tính bảo mật cao và có khả năng tái sử dụng. Do đó sử dụng phương pháp lập trình hướng đối tượng giúp tăng năng suất và dễ bảo trì mã.

### 7.2. Chuẩn viết mã

#### 7.2.1. Cách đặt tên và khai báo:

- Có 3 kiểu đặt tên thông dụng nhất:
  - + Pascal Case: Chữ cái đầu tiên trong từ định danh và chữ cái đầu tiên của mỗi từ nối theo sau phải được viết hoa. Sử dụng Pascal Case để đặt tên cho một tên có từ 3 ký tự trở lên.

Vd: CodingConvention

- + Camel Case: Chữ cái đầu tiên trong từ định danh là chữ thường và chữ cái đầu tiên của mỗi từ nối theo sau phải được viết hoa.

Vd: codingConvention

- + Uppercase: Tất cả các ký tự trong từ định danh phải được viết hoa. Sử dụng quy tắc này đối với tên định danh có từ 2 ký tự trở xuống.

Vd: System.IO

- Tóm tắt cách sử dụng quy tắc:

<i>Loại</i>	<i>Kiểu đặt tên</i>	<i>Ví dụ</i>	<i>Ghi chú</i>
Tên class, enum	Pascal Case	CreateUser	Danh từ
Thuộc tính	Pascal Case	BackgroundColor	Danh từ
Phương thức	Pascal Case	GetValue()	Động từ
Sự kiện	Pascal Case	SelectedIndexChanged	Có hậu tố EventHandle
Giao diện (Interface)	Pascal Case	IButtonControl	Có tiền tố I
Tên biến	Carmel Case	firstName	Danh từ
Tham số	Carmel Case	displayTime	Danh từ
Hằng số	Uppercase	MAX_VALUE	Có gạch chân giữa các từ

- Tránh thêm tiền tố, hậu tố dư thừa.  
Vd: ~~enum BorderEnum {...}~~ -> `enum Border {...}`
- Không thêm tên lớp chứa vào tên thuộc tính.  
Vd: ~~Animal.WeightAnimal~~ -> `Animal.Weight`
- Tên biến, phương thức bool phải thể hiện được ý nghĩa nếu trả về true hoặc false. Nên sử dụng tiền tố "Is" "Can" "Has" trước tên biến, phương thức.  
Vd: ~~bool checked = true;~~ -> `bool isChecked = true;`
- Không dùng các tên giống nhau.  
Vd: `int yourAge`  
~~`int YourAge`~~
- Không tạo 2 namespace cùng tên và chỉ khác nhau ở kiểu viết hoa/thường.  
Vd: `Namespace SunAsterisk`  
~~`Namespace sunAsterisk`~~
- Không nên xây dựng 1 phương thức với các tham số có cùng tên và chỉ khác nhau kiểu chữ:  
Vd: ~~`void MyFunction(string a, string A)`~~
- Không xây dựng 1 kiểu với các tên property giống nhau và chỉ phân biệt ở kiểu chữ.  
Vd: `int Color {get, set}`  
~~`int COLOR {get, set}`~~
- Không đặt tên các phương thức có cùng tên và chỉ khác nhau ở kiểu chữ.  
Vd: `void Total ()`  
~~`void total ()`~~

### 7.2.2. Câu lệnh:

- Các câu lệnh khác nhau phải nằm trên các dòng khác nhau. Một dòng không nên nhiều quá 150 ký tự.
- Đối với câu điều kiện thì phần điều kiện nên ở chung một dòng với if, nếu nhiều điều kiện quá có thể xuống dòng. Phần code thực hiện câu điều kiện nằm ở giữa 2 dấu "{", "}". Mỗi dấu "{", "}" nằm ở 2 dòng riêng. Các câu điều kiện liên tiếp nhau cách nhau 1 dòng trống.

```

Console.WriteLine("Hello, world!");
Console.WriteLine("We are writing some code.");

if (a > b && b > c || c == a)
{
    Console.WriteLine("if condition is true, print this line.");
}

if (a < b)
{
    Console.WriteLine("if condition is true, print this line.");
}
else
{
    Console.WriteLine("Otherwise, print this line.");
}

```

### 7.2.3. Hàm và tham số

- Mỗi hàm được phải được đặt tên bằng tiếng Anh và thể hiện được rõ chức năng của hàm và các từ trong hàm được cách nhau bằng “\_”.

Vd: `int run_sum()` , không được viết thành `runsum`.

- Không được đặt số và kí tự đặc biệt vào đầu tên của mỗi hàm

Vd: `int 1run_sum()` , `int $#run_sum()`

- Không được đặt trùng tên với các phương thức có sẵn

Vd: `char sort()` , `int sum()` , `int max()`

- Mỗi hàm chỉ thực hiện một chức năng duy nhất.

- o Tham số của hàm

Tên của tham số phải thể hiện được ý nghĩa của nó, đặt tên phải liên mạch, không có khoảng trắng và các kí tự đặc biệt.

Vd: `double phan_so (int tuso, int mauso)`

Không được đặt trùng tên với các phương thức có sẵn của ngôn ngữ

Khai báo tham số phải ngắn gọn, thể hiện được mục đích của người viết.

Tham số phải khai báo ngang cung hàm với hàm, không được khai báo khác hàng.

Vd: `int sum(int a, int b, int c)`

không được viết thành

```

int sum(int a,
        int b,
        int c)

```

Khi thực hiện vòng for hoặc if else thì các phương thức trong nó phải TAB vào 1 lần.

```
Vd: for (...)  
{  
    # Phương thức 1  
    If ()  
    {  
        # Phương thức 2  
    }  
}
```

#### 7.2.4. Lớp

- Tên lớp là một danh từ hay cụm danh từ, được viết hoa chữ cái đầu tiên của mỗi từ.  
Vd: Flower, Monster, BieuMau, ...
- Hàm trong lớp làm tương tự các hàm khác.
- Các thuộc tính của lớp được khai báo ở phần đầu của lớp. Cách đặt tên, cách khai báo tương tự các biến khác.

```
Oreferences  
class Person  
{  
    string Name;  
    int age;  
  
    Oreferences  
    void showInfo()  
    {  
        Console.WriteLine("Name is ", Name);  
        Console.WriteLine("Age is ", age);  
    }  
}
```

### **III. Phần kết luận**

#### **1. Đánh giá đồ án**

##### ***1.1. Nhận xét***

- Hợp tác giữa các thành viên ổn định, phân chia công việc đồng đều
- Đa số hoàn thành các chức năng và đáp ứng yêu cầu đồ án
- Bổ sung được một tính năng khác
- Trong quá trình thực hiện đồ án, nhóm gặp một số khó khăn trong lập trình cũng như viết báo cáo. Một vài chỗ trong phần mềm chưa được hoàn thiện, chưa được thiết kế đẹp mắt.
- Có thể xuất hiện các lỗi tiềm ẩn trong phần mềm
- Phần mềm khó được phát triển rộng rãi do sự khác biệt giữa các phiên bản sql server cũng như kết nối với các cơ sở dữ liệu đó. Cần sự hiểu biết nâng cao của người dùng để tự kết nối bằng tay.

##### ***1.2. Hướng phát triển***

- Thêm nhiều chức năng hơn cho phần mềm.
- Cải thiện giao diện dễ nhìn và sử dụng.
- Phần mềm cần thêm chức năng phân quyền truy cập và quản lý. Thêm chức năng đăng nhập ứng dụng.
- Đồng bộ hóa giữa các thiết bị.
- Tự động thực hiện kết nối đến cơ sở dữ liệu mà không cần sự trợ giúp của con người.

## 2. Phân công công việc

Tên	Công việc	Tiến độ hoàn thành	Ghi chú
Nguyễn Lộc Linh 19521754	Xác định đặc tả yêu cầu phần mềm 2 chức năng	100%	
	Công cụ hỗ trợ quản lý công việc	100%	
	Thiết kế mô hình dữ liệu 2 chức năng	100%	
	Thiết kế giao diện 2 chức năng	100%	
	Thiết kế xử lý 2 chức năng	100%	
	Thiết kế phần mềm Application Architecture và System Architecture 2 chức năng	100%	
	Chuẩn viết mã: oop, statement, class	100%	
	Tổng hợp tham gia viết báo cáo	100%	
	Thực hiện lập trình ứng dụng	100%	
Trần Duy Quang 19522102	Xác định đặc tả yêu cầu phần mềm 2 chức năng	100%	
	Quy trình phát triển đồ án	100%	
	Thiết kế mô hình dữ liệu 2 chức năng	100%	
	Thiết kế giao diện 2 chức năng	100%	
	Thiết kế xử lý 2 chức năng	100%	
	Thiết kế phần mềm Application Architecture và System Architecture 2 chức năng	100%	
	Chuẩn viết mã: declarations, variables, naming conventions	100%	
	Tổng hợp tham gia viết báo cáo	100%	
Nguyễn Trí Tài 19522572	Xác định đặc tả yêu cầu phần mềm 2 chức năng	100%	
	Kế hoạch phát triển đồ án	100%	
	Thiết kế mô hình dữ liệu 2 chức năng	100%	
	Thiết kế giao diện 2 chức năng	100%	
	Thiết kế xử lý 2 chức năng	100%	
	Thiết kế phần mềm Application Architecture và System Architecture 2 chức năng	100%	
	Chuẩn viết mã: functions, parameters, loop	100%	
	Tổng hợp tham gia viết báo cáo	100%	



### **3. Tài liệu tham khảo**

[1] Tài liệu tham khảo trên [courses.uit.edu.vn](https://courses.uit.edu.vn) môn công nghệ phần mềm

[2] Slide bài giảng trên [courses.uit.edu.vn](https://courses.uit.edu.vn) môn công nghệ phần mềm

[3] Danh sách đề án tham khảo trên [courses.uit.edu.vn](https://courses.uit.edu.vn) công nghệ phần mềm

[4] C# coding convention, <https://docs.microsoft.com/en-us/dotnet/csharp/fundamentals/coding-style/coding-conventions>, (02/07/2021)

[5] Xử lý lỗi, <https://stackoverflow.com/>