

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

# BÁO CÁO BÀI TẬP LỚN

TRẦN PHÚC MẠNH LINH

Linh.tpm200352@sis.hust.edu.vn

Giảng viên: TS. Trịnh Tuấn Đạt

Bộ môn: Thiết kế và xây dựng phần mềm

Lớp: 143801

Nhóm: 6

HÀ NỘI, 12/2023

## MỤC LỤC

<b>CHƯƠNG 1. PHÂN TÍCH YÊU CẦU (BÀI 2).....</b>	<b>1</b>
1.1    Biểu đồ use case .....	1
1.1.1    Biểu đồ use case tổng quan .....	1
1.1.2    Biểu đồ use case phân rã “Xem thông tin chấm công” .....	3
1.1.3    Chỉnh sửa thông tin chấm công.....	4
1.2    Đặc tả Use case .....	5
1.2.1    Use case “Xem báo cáo chấm công cho đơn vị nhân viên văn phòng”	5
1.3    Từ điển thuật ngữ .....	8
1.3.1    Comment .....	8
1.3.2    Click .....	9
1.3.3    Import.....	9
1.3.4    Export .....	9
1.4    Đặc tả phụ trợ .....	9
1.4.1    Hiệu năng .....	9
1.4.2    Độ tin cậy .....	9
1.4.3    An toàn, bảo mật .....	9
1.4.4    Bảo trì .....	9
1.4.5    Khả năng sử dụng .....	10
1.4.6    Khả năng mở rộng.....	10
<b>CHƯƠNG 2. PHÂN TÍCH USE CASE (BÀI 3).....</b>	<b>11</b>
2.1    Thiết kế biểu đồ tương tác.....	11
2.1.1    Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”	11
2.2    Thiết kế biểu đồ lớp .....	18
2.2.1    Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”	18
<b>CHƯƠNG 3. THIẾT KẾ GIAO DIỆN VÀ SUBSYSTEM (BÀI 4) .....</b>	<b>19</b>
3.1    Sơ đồ chuyển đổi màn hình.....	19
3.1.1    Sơ đồ chuyển đổi màn hình của UC Xem báo cáo chấm công cho đơn vị nhân viên văn phòng .....	19

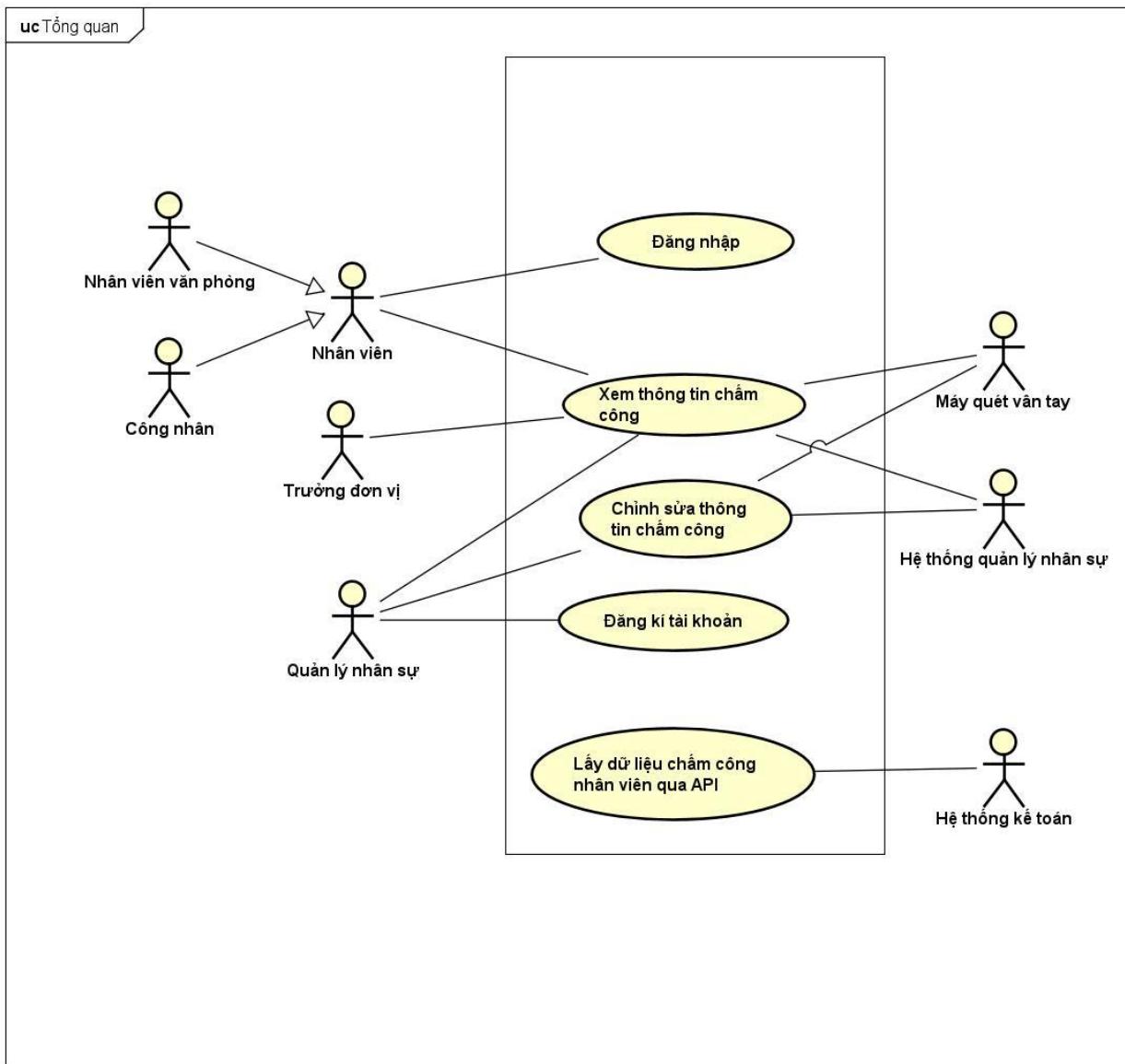
3.2	Đặc tả màn hình .....	20
3.2.1	Đặc tả màn hình OfficerDepartmentAttendanceReportView ...	20
3.2.2	Đặc tả màn hình EmployeeAttendanceView .....	22
3.2.3	Đặc tả màn hình AttendanceReportExportPanel .....	23
3.3	Biểu đồ trình tự cho Subsystem .....	24
3.4	Biểu đồ lớp cho Subsystem.....	26
<b>CHƯƠNG 4. THIẾT KẾ CHI TIẾT LỚP (BÀI 5)</b>		<b>27</b>
4.1	Thiết kế biểu đồ trình tự mức thiết kế.....	27
4.1.1	Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”	27
4.2	Biểu đồ lớp mức thiết kế .....	34
4.2.1	Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”	34
<b>CHƯƠNG 5. CẢI THIỆN BIỂU ĐỒ TRÌNH TỰ VÀ LỚP MỨC THIẾT KẾ (BÀI 6)</b>		<b>38</b>
5.1	Cách chỉnh sửa và ý nghĩa mang lại.....	38
5.1.1	Áp dụng mẫu thiết kế Singleton.....	38
5.1.2	Áp dụng mẫu thiết kế Factory .....	38
5.1.3	Áp dụng nguyên tắc Cohesion and Coupling .....	38
5.1.4	Áp dụng nguyên lý SOLID .....	38
5.2	Thiết kế biểu đồ trình tự mức thiết kế.....	40
5.2.1	Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”	40
5.3	Biểu đồ lớp mức thiết kế .....	46
5.3.1	Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”	47
<b>CHƯƠNG 6. LẬP TRÌNH VÀ KIỂM THỬ ĐƠN VỊ (BÀI 7)</b>		<b>51</b>
6.1	Thiết kế test case cho module: UserService.....	51
6.1.1	Mô tả module .....	51
6.1.2	Thiết kế test case cho phương thức authenticate của UserService	51
6.1.3	Kỹ thuật sử dụng kiểm thử hộp đen và phân vùng tương đương.	51
6.1.4	Kỹ thuật sử dụng: kiểm thử hộp trắng và bảng quyết định.....	51
6.2	Thiết kế kiểm thử use case .....	53

6.2.1	Test case Xem chấm báo cáo công đơn vị .....	54
6.2.2	Test case Xem chấm báo cáo công đơn vị vào tháng khác.....	54
6.2.3	Test case Xuất báo cáo chấm công thành file báo cáo thành công	
	55	
6.2.4	Test case Xuất báo cáo chấm công thành file báo cáo với vị trí sai	
	56	
6.2.5	Test case Tìm kiếm nhân viên bằng mã nhân viên .....	57
6.2.6	Test case Xem báo cáo công chi tiết nhân viên .....	58
6.2.7	Test case Xem báo cáo chấm công chi tiết nhân viên vào tháng khác	
	59	

# CHƯƠNG 1. PHÂN TÍCH YÊU CẦU (BÀI 2)

## 1.1 Biểu đồ use case

### 1.1.1 Biểu đồ use case tổng quan



Giải thích về các tác nhân:

- **Nhân viên:** Người dùng cấp thấp nhất, chỉ có thể xem được các thông tin chấm công liên quan đến mình.
- **Nhân viên văn phòng:** Kế thừa nhân viên, chỉ xem được thông tin chấm công dạng nhân viên văn phòng.
- **Công nhân:** Kế thừa nhân viên, chỉ xem được thông tin chấm công dạng công nhân.
- **Trưởng đơn vị:** Người dùng cấp cao hơn nhân viên, có thể xem được thông tin chấm công của các thành viên trong đơn vị của mình quản lý.
- **Quản lý nhân sự:** Người dùng cấp cao nhất, có thể xem được thông tin chấm công của toàn bộ nhân viên công ty, có quyền chỉnh sửa thông tin chấm công, xem báo cáo, import/export dữ liệu chấm công.

- **Máy quét vân tay:** Hệ thống bên ngoài có nhiệm vụ cung cấp API cho hệ thống lấy dữ liệu chấm công
- **Hệ thống quản lý nhân sự:** Hệ thống bên ngoài có nhiệm vụ cung cấp API cho hệ thống chấm công lấy thông tin của nhân viên.
- **Hệ thống kế toán:** Hệ thống bên ngoài có khả năng lấy thông tin chấm công từ hệ thống chấm công để tính lương của nhân viên cho từng tháng.

Giải thích về các use case:

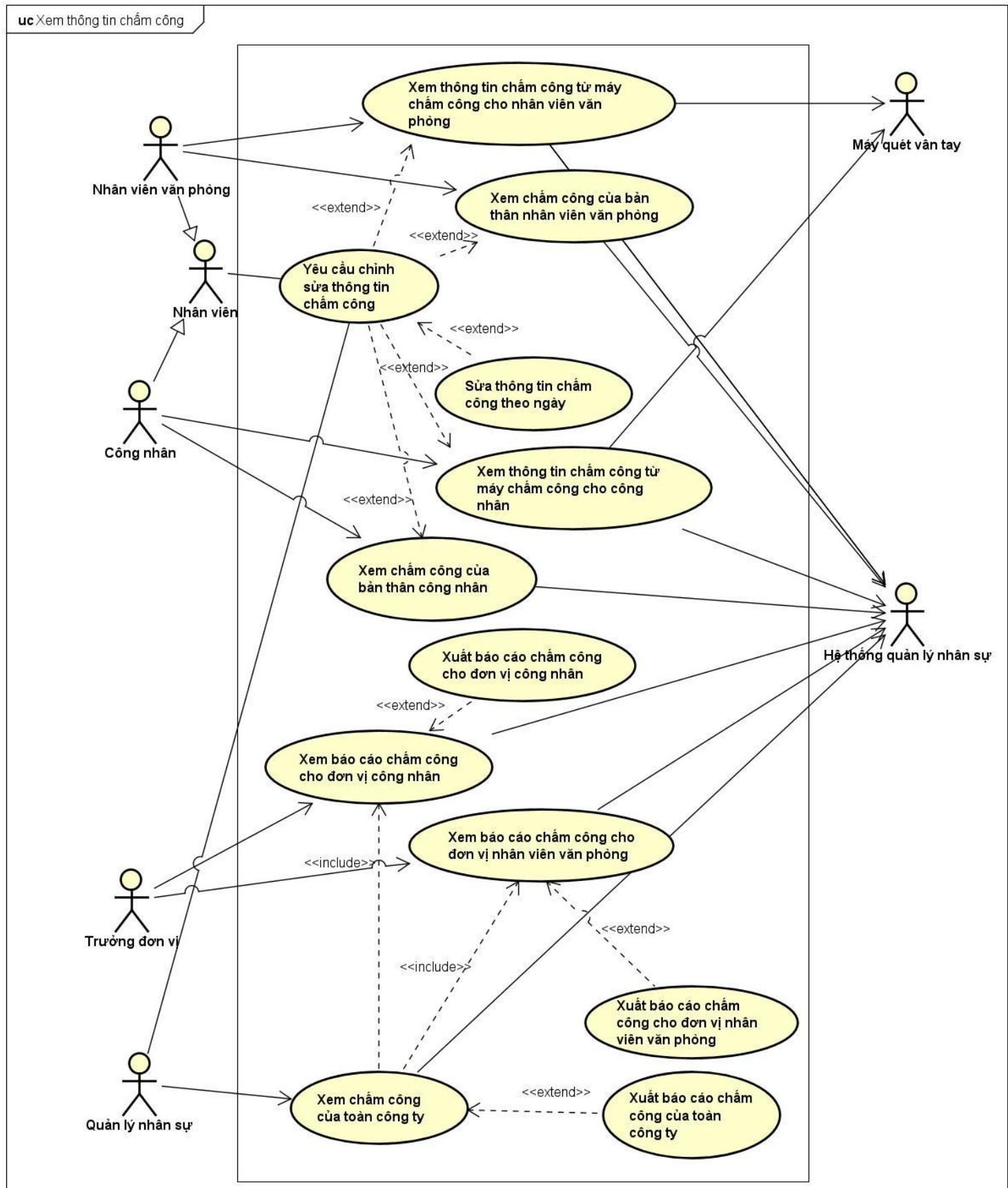
Composite use case:

- **Xem thông tin chấm công:** Use case tổng quát của các use case thực hiện các hành vi nghiệp vụ lên máy chấm công vân tay.
- **Chỉnh sửa thông tin chấm công:** Use case tổng quát của các use case thực hiện các hành vi nghiệp vụ lên

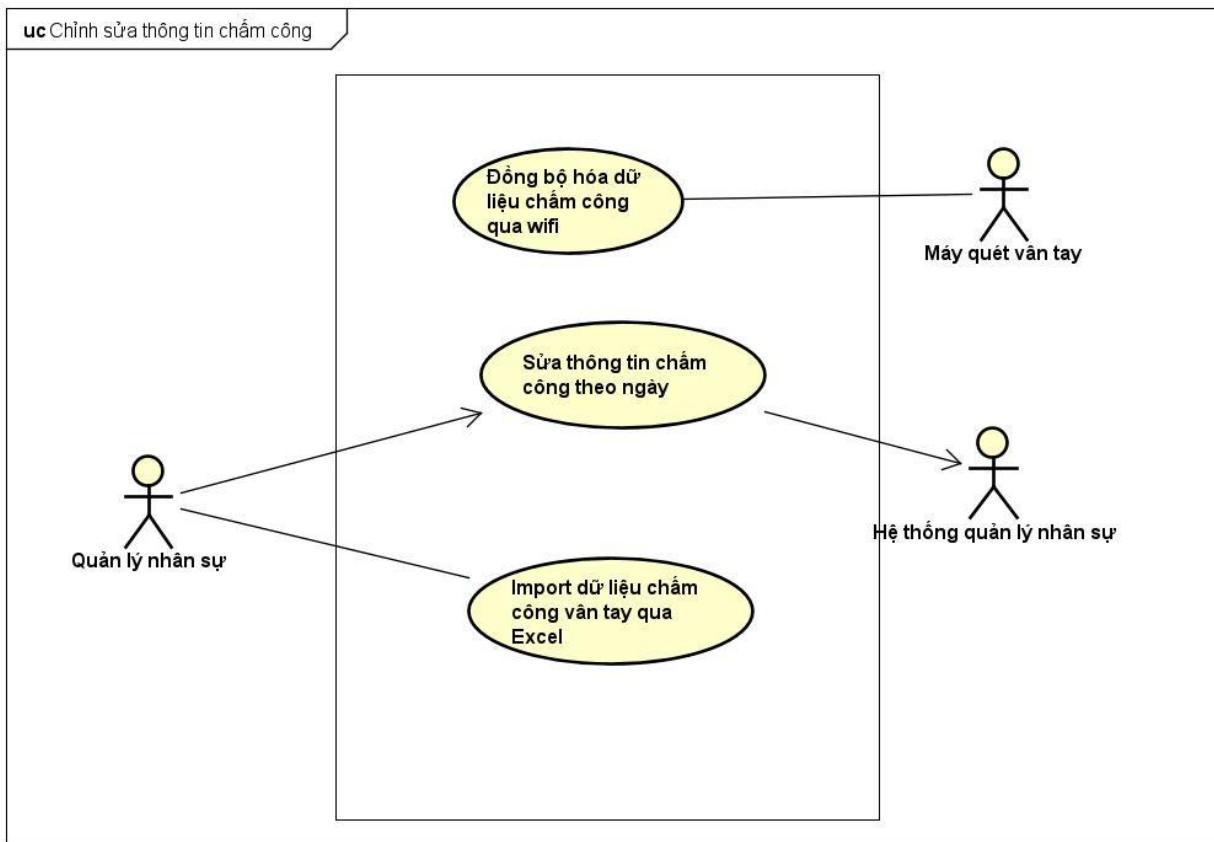
Actual use case:

- **Cung cấp dữ liệu chấm công nhân viên qua API**
- **Đăng ký tài khoản**
- **Đăng nhập**
- **Lấy dữ liệu chấm công nhân viên qua API**

### 1.1.2 Biểu đồ use case phân rã “Xem thông tin chấm công”



### 1.1.3 Chỉnh sửa thông tin chấm công



## 1.2 Đặc tả Use case

### 1.2.1 Use case “Xem báo cáo chấm công cho đơn vị nhân viên văn phòng”

<b>Mã Use case</b>	UC001	<b>Tên Use case</b>	Xem báo cáo chấm công cho đơn vị nhân viên văn phòng
<b>Tác nhân</b>	Trưởng đơn vị, Hệ thống quản lý nhân sự		
<b>Tiền điều kiện</b>	Đã đăng nhập vào hệ thống		
<b>Luồng sự kiện chính (Thành công)</b>	STT	Thực hiện bởi	Hành động
	1.	Trưởng đơn vị	Chọn chức năng Xem báo cáo chấm công
	2.	Hệ thống	Gọi đến Hệ thống quản lý nhân sự để lấy thông tin về đơn vị trưởng đơn vị quản lý
	3.	Hệ thống quản lý nhân sự	Trả về thông tin của đơn vị trưởng đơn vị quản lý
	4.	Hệ thống	Gọi đến Hệ thống quản lý nhân sự để xem danh sách các nhân viên trong đơn vị cũng như thông tin về các nhân viên đó
	5.	Hệ thống quản lý nhân sự	Trả về danh sách các nhân viên trong đơn vị cũng như thông tin về các nhân viên đó
	6.	Hệ thống	Lấy các bản ghi chấm công tháng này của các thành viên trong đơn vị đó từ CSDL
	7.	Hệ thống	Tổng hợp các bản ghi chấm công thành các bản ghi báo cáo chấm công của từng thành viên trong đơn vị theo tháng.
	8.	Hệ thống	Bổ sung thêm thông tin họ tên cho các bản ghi báo cáo chấm công dựa trên thông tin nhân viên lấy được từ Hệ thống quản lý nhân sự
	9.	Hệ thống	Tổng hợp số giờ đi muộn, về sớm, ... của tất cả thành viên trong đơn vị
<b>Luồng sự kiện thay thế</b>	10.	Hệ thống	Hiển thị các báo cáo chấm công tháng này đến cho người dùng.
	STT	Thực hiện bởi	Hành động
	10a.	Trưởng đơn vị	Chọn xuất báo cáo ra CSV hoặc Excel
	11a.	Hệ thống	Hiển thị bảng tùy chọn xuất báo cáo
	12a.	Trưởng đơn vị	Chọn định dạng file mong muốn: CSV hoặc Excel
	13a.	Trưởng đơn vị	Chọn vị trí lưu file mong muốn
	14a.	Trưởng đơn vị	Xác nhận xuất file
	15a.	Hệ thống	Kiểm tra xem Vị trí lưu file có tồn tại hay không
	16a.	Hệ thống	Lưu file báo cáo và về giao diện Xem báo cáo chấm công
	16a1.	Hệ thống	Thông báo lỗi: Vị trí lưu file không tồn tại Quay lại bước 15b
	10b.	Trưởng đơn vị	Chọn tháng khác để xem (tương tự với quý, năm)
	11b.	Hệ thống	Lấy các bản ghi chấm công tháng được chọn của các thành viên trong đơn vị đó từ CSDL

	12b.	Hệ thống	Bổ sung thêm thông tin họ tên cho các bản ghi báo cáo chấm công
	13b.	Hệ thống	Tổng hợp số giờ đi muộn, về sớm, ... của tất cả thành viên trong đơn vị
	14b.	Hệ thống	Hiển thị các báo cáo chấm công tháng được chọn đến cho người dùng.
	10c.	Trưởng đơn vị	Chọn nhân viên muốn xem chi tiết
	11c.	Hệ thống	Lấy các bản ghi chấm công của nhân viên đó từ CSDL
	12c.	Hệ thống	Bổ sung thêm thông tin họ tên cho các bản ghi báo cáo chấm công
	13c.	Hệ thống	Tổng hợp số giờ đi muộn, về sớm, ... của nhân viên đó
	14c.	Hệ thống	Hiển thị các báo cáo chấm công của nhân viên đó đến cho người dùng.
	10c1.	Trưởng đơn vị	Chọn tháng muốn xem chi tiết
	10c2.	Hệ thống	Lấy các bản ghi chấm công của nhân viên đó từ CSDL trong tháng muốn xem chi tiết
	10c3.	Hệ thống	Bổ sung thêm thông tin họ tên cho các bản ghi báo cáo chấm công
	10c4.	Hệ thống	Tổng hợp số giờ đi muộn, về sớm, ... của nhân viên đó
	10c5.	Hệ thống	Hiển thị các báo cáo chấm công của nhân viên đó trong tháng mong muốn đến cho người dùng.
	10d.	Trưởng đơn vị	Nhập vào ô tìm kiếm nhân viên bằng mã nhân viên
	11d.	Trưởng đơn vị	Xác nhận tìm kiếm
	12d.	Hệ thống	Gọi đến hệ thống quản lý nhân sự lấy thông tin của các nhân viên văn phòng ở trong đơn vị mình quản lý mà có mã nhân viên bao hàm chuỗi kí tự tìm kiếm
	13d.	Hệ thống quản lý nhân sự	Trả về thông tin nhân viên theo yêu cầu
	14d.	Hệ thống	Lọc ra danh sách nhân viên thỏa mãn yêu cầu dựa trên thông tin nhận được và trên.
	15d.	Hệ thống	Lấy các bản ghi chấm công tháng này của những nhân viên thỏa mãn yêu cầu tìm kiếm trong đơn vị đó từ CSDL
	16d.	Hệ thống	Bổ sung thêm thông tin họ tên cho các bản ghi báo cáo chấm công dựa trên thông tin nhân viên lấy được từ Hệ thống quản lý nhân sự
	17d.	Hệ thống	Tổng hợp số giờ đi muộn, về sớm, số buổi làm,... của những nhân viên thỏa mãn yêu cầu tìm kiếm trong đơn vị
	18d.	Hệ thống	Hiển thị các bản ghi báo cáo chấm công của những nhân viên thỏa mãn yêu cầu tìm kiếm
<b>Hậu điều kiện</b>	Không		

\* Dữ liệu của từng đơn vị gồm các trường dữ liệu sau:

STT	Trường dữ liệu	Mô tả	Ví dụ
1.	Mã đơn vị	Mã để định danh từng đơn vị	RD
2.	Tên đơn vị		Phòng R&D

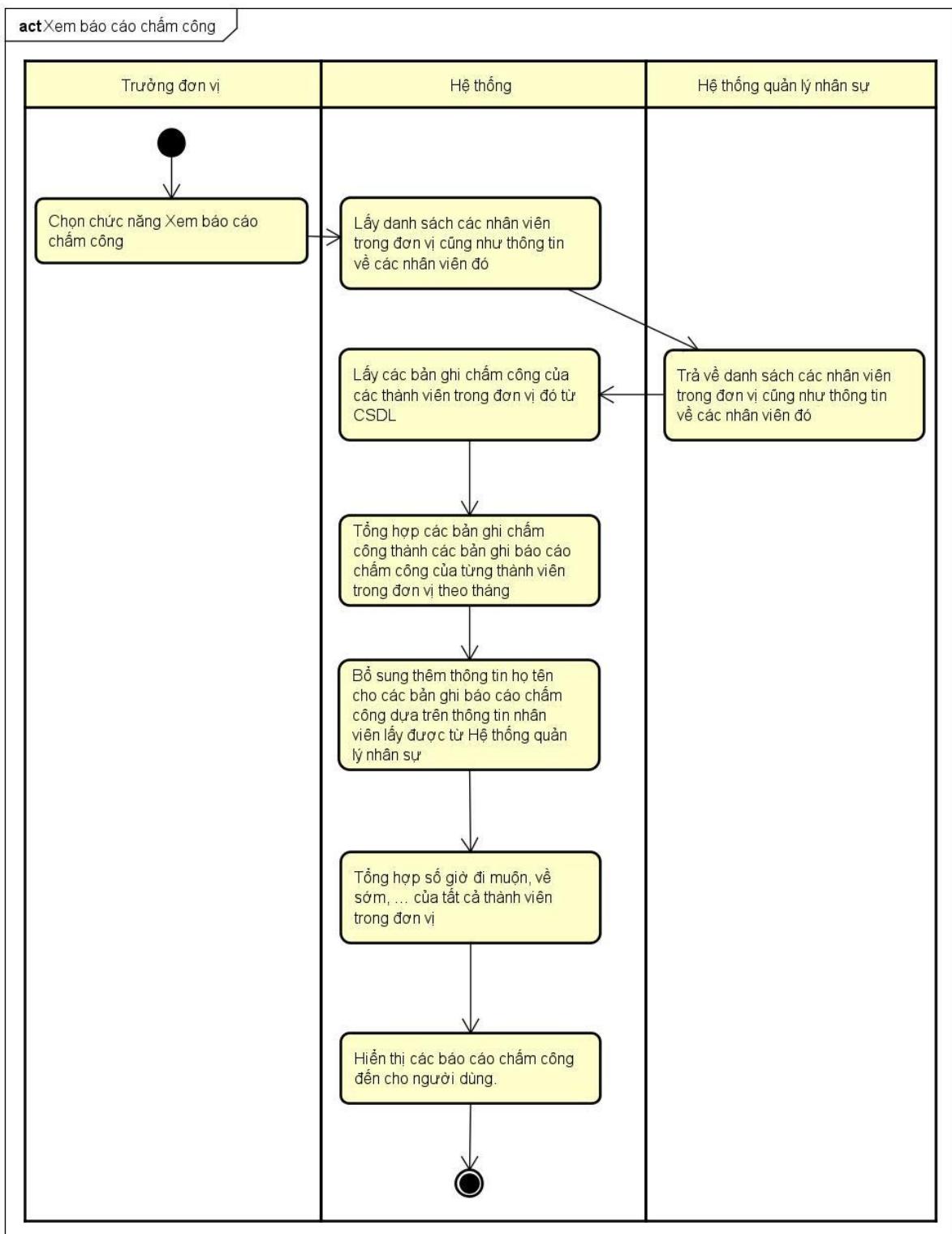
\* Dữ liệu của từng bản ghi báo cáo chấm công nhân viên văn phòng gồm các trường dữ liệu sau:

STT	Trường dữ liệu	Mô tả	Ví dụ
1.	Họ tên		Nguyễn An Lành
2.	Mã nhân viên	Mã để định danh từng nhân viên	NV-2501
3.	Đơn vị	Đơn vị mà nhân viên ấy nằm trong đó	Phòng R&D
4.	Tháng		06/2023
5.	Tổng số buổi đi làm gồm cả sáng và chiều	Tổng số buổi đi làm gồm cả sáng và chiều	48
6.	Tổng số giờ đi muộn/về sớm		2

\* Thông tin để xuất file báo cáo gồm các trường dữ liệu sau:

STT	Trường dữ liệu	Mô tả	Bắt buộc?	Điều kiện hợp lệ	Ví dụ
1.	Định dạng file	CSV hoặc Excel	Có		CSV
2.	Vị trí lưu file	Một chuỗi ký tự chỉ đến một địa chỉ thư mục nào đó	Có	Thư mục phải tồn tại trong hệ thống	C:/ThuMucMuonLuu

\* Biểu đồ hoạt động:



### 1.3 Từ điển thuật ngữ

Sau đây là một số thuật ngữ được sử dụng trong tài liệu:

#### 1.3.1 Comment

- Thêm miêu tả cho một cái gì đó, cụ thể trong tài liệu này là thêm miêu tả cho mã nguồn

### **1.3.2 Click**

- Hành động bấm vào nút bên trái chuột 2 lần liên tiếp trong một khoảng thời gian rất ngắn.

### **1.3.3 Import**

- Hành động nhập dữ liệu bằng file.

### **1.3.4 Export**

- Hành động xuất dữ liệu ra một file.

## **1.4 Đặc tả phụ trợ**

### **1.4.1 Hiệu năng**

- Trải nghiệm sử dụng mượt mà, ổn định.
- Thời gian phản hồi của ứng dụng dưới 1s.
- Cho phép 100 người dùng truy cập đồng thời.
- Cập nhật thông tin chấm công từ máy chấm công vân tay đến hệ thống chấm công với độ trễ ít hơn 1s

### **1.4.2 Độ tin cậy**

- Đảm bảo nhận diện thành công ít nhất 98% lần thử vân tay.
- Dữ liệu được backup thường xuyên đảm bảo không bị mất mát (1 lần/tuần).
- Có công cụ phục hồi dữ liệu từ cơ sở dữ liệu dự phòng nếu cơ sở dữ liệu chính gặp sự cố.
- Cho phép hệ thống hoạt động liên tục 24 giờ/ngày, 7 ngày/tuần, tăng khả năng chịu lỗi bằng việc duy trì hệ thống chấm công trên 2 node.
- Không quá 1 lỗi trên 800 dòng mã.

### **1.4.3 An toàn, bảo mật**

- Sử dụng chuẩn băm SHA-256 để mã hoá dữ liệu nhạy cảm như đặc trưng vân tay, mật khẩu của người dùng.
- Phân quyền người dùng để những người không liên quan có thể truy cập đến dữ liệu nhạy cảm.
- Yêu cầu xác thực người dùng (đăng nhập) trước khi thực hiện các hành động truy cập vào tài nguyên của hệ thống để tránh dữ liệu bị rò rỉ ra bên ngoài.
- Hệ thống được triển khai trên server cục bộ và được ngắt kết nối với internet để đảm bảo an toàn cho dữ liệu nhân viên (Hệ thống chỉ truy cập được bằng wifi của công ty).

### **1.4.4 Bảo trì**

- Phần mềm được tách ra thành các module cho từng tính năng để thuận tiện cho quá trình bảo trì.
- Phần mềm sẽ được duy trì trong khoảng 3 năm.
- Tất cả các phương thức trong mã nguồn đều phải được comment cũng như có tài liệu về cấu trúc hệ thống, cấu trúc mã nguồn.

#### **1.4.5 Khả năng sử dụng**

- Giao diện Material thân thiện kèm tài liệu sử dụng, hầu như không cần đào tạo để sử dụng thành thạo.
- Giao diện người dùng có thể chạy được trên các trình duyệt phổ biến, truy cập chỉ bằng 1 click.

#### **1.4.6 Khả năng mở rộng**

- Ứng dụng phải được thiết kế sao cho thuận tiện cho việc mở rộng quy mô về tính năng cũng như số lượng người dùng, số lượng dữ liệu liên tục mở rộng.

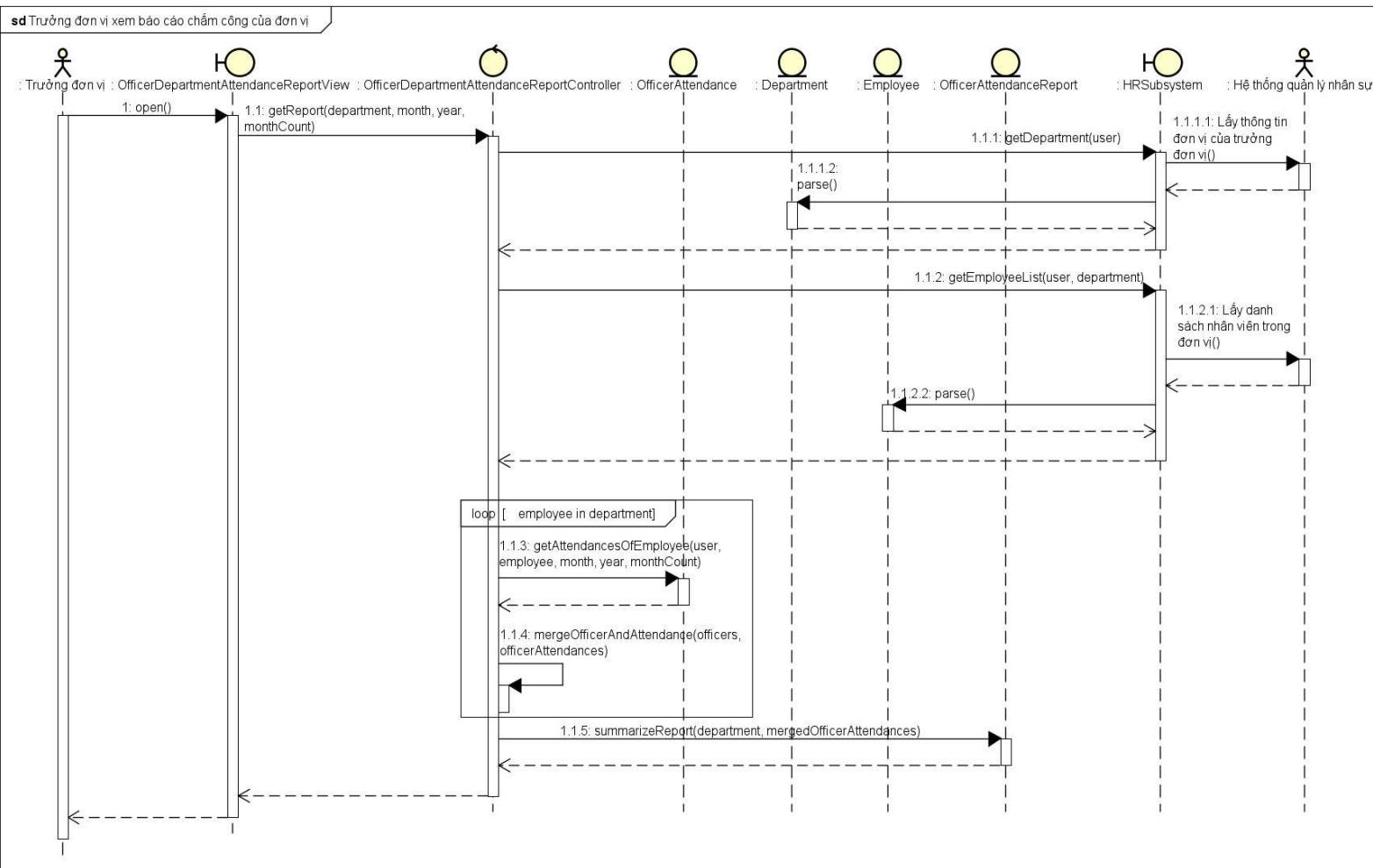
## CHƯƠNG 2. PHÂN TÍCH USE CASE (BÀI 3)

### 2.1 Thiết kế biểu đồ tương tác

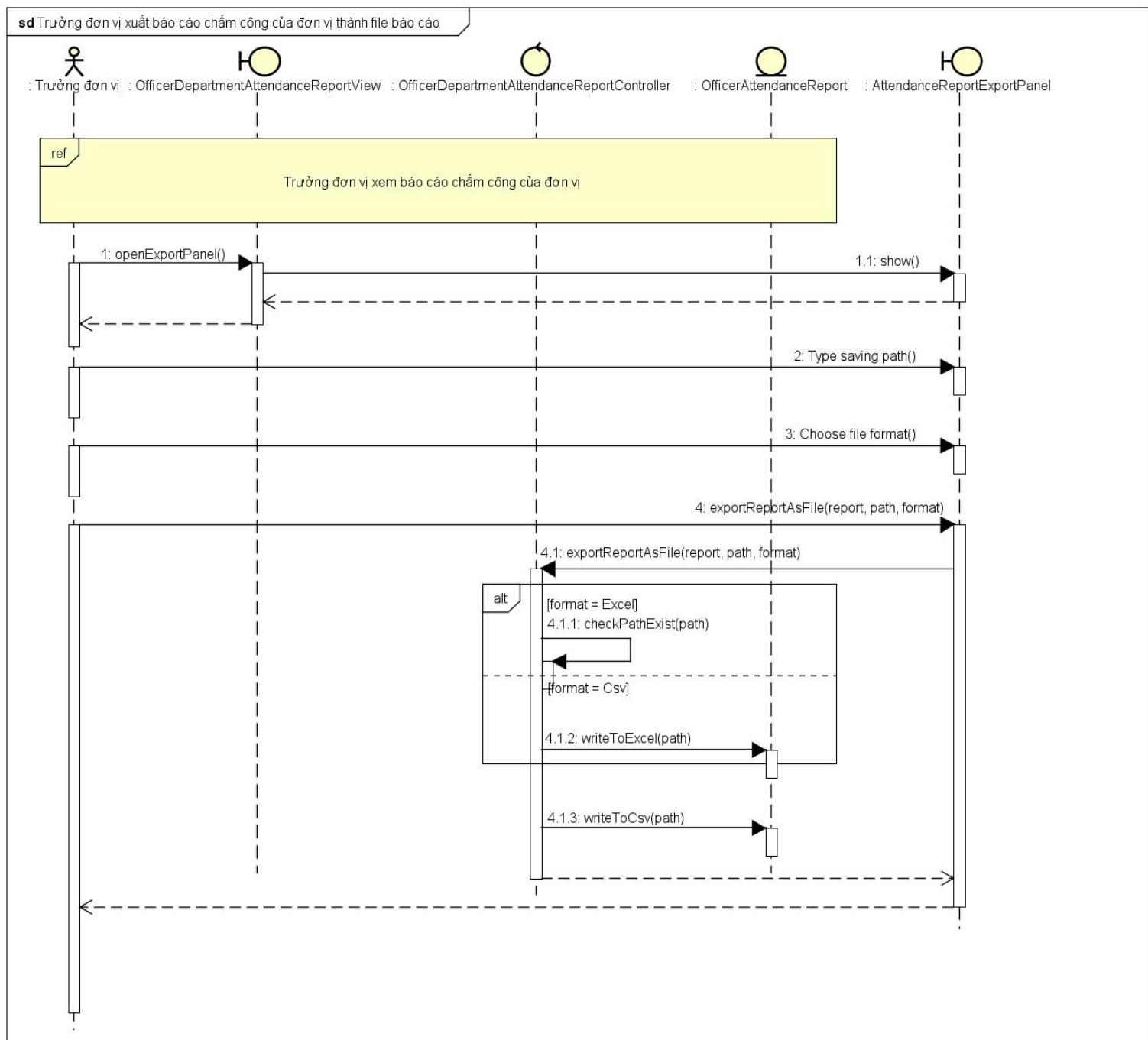
#### 2.1.1 Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”

##### 2.1.1.1. Biểu đồ trình tự (mức phân tích)

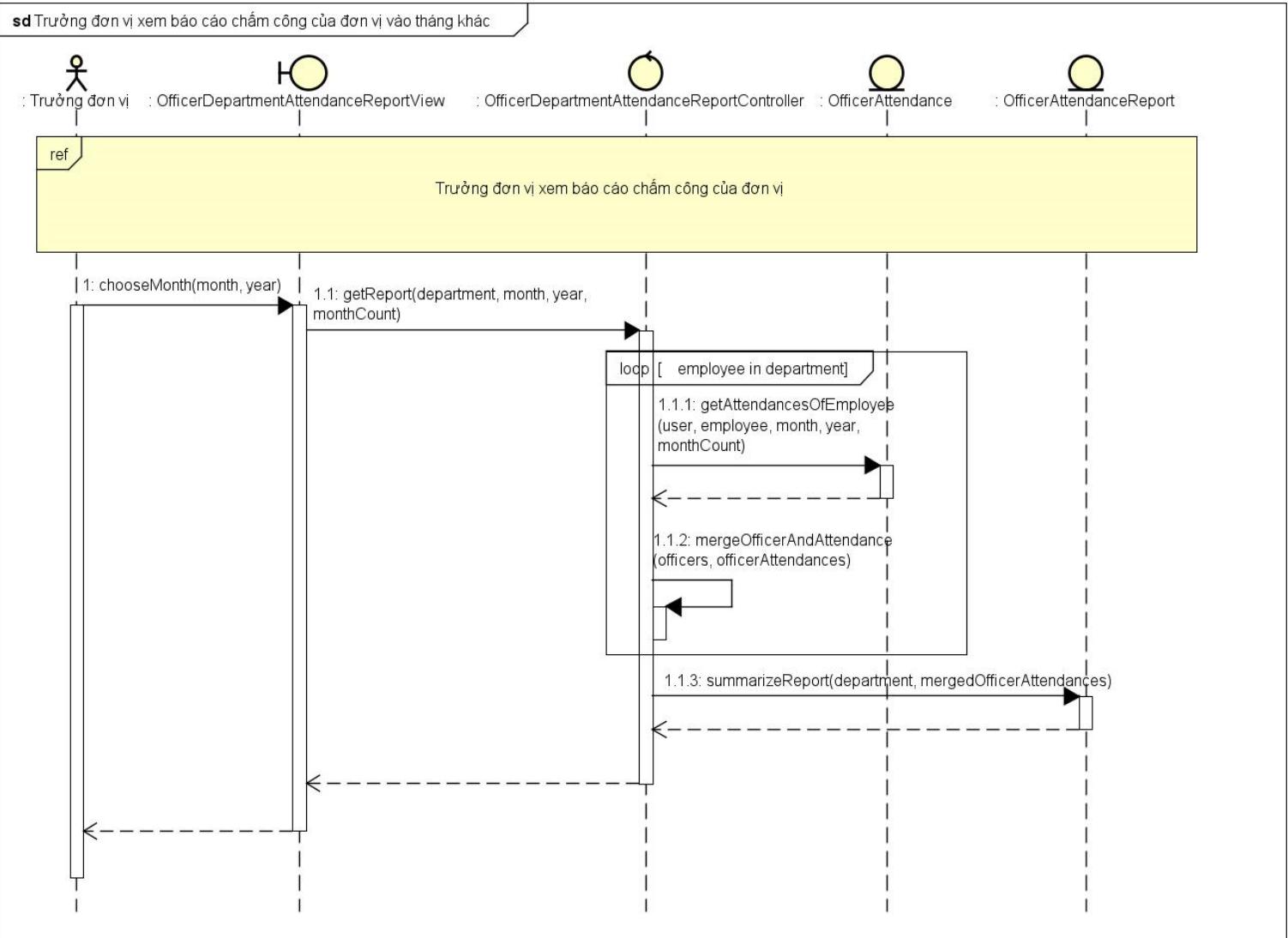
\* Main Scenario: Trưởng đơn vị xem báo cáo chấm công của đơn vị



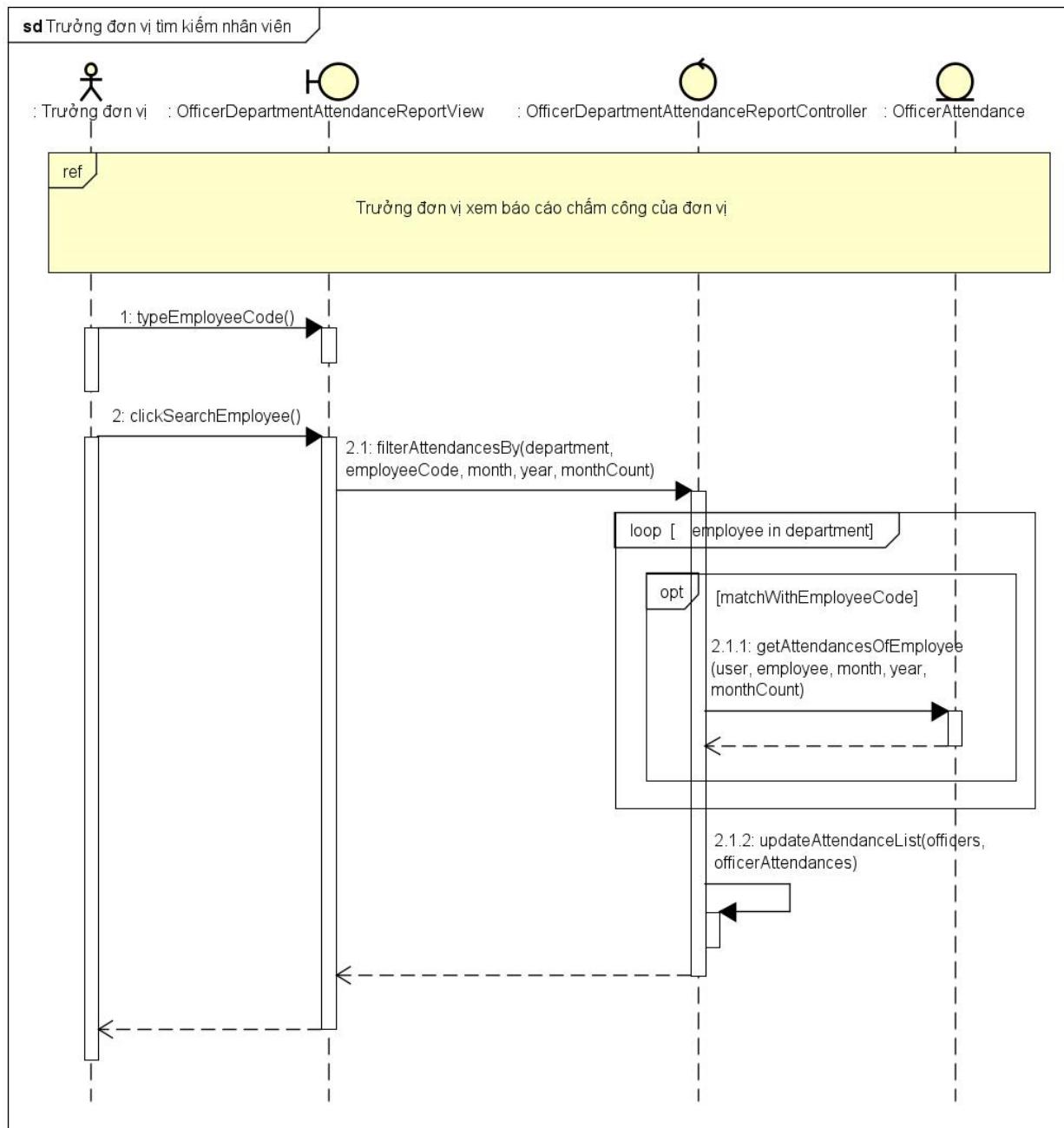
\* Scenario 2: Trưởng đơn vị xuất báo cáo chấm công của đơn vị thành file báo cáo



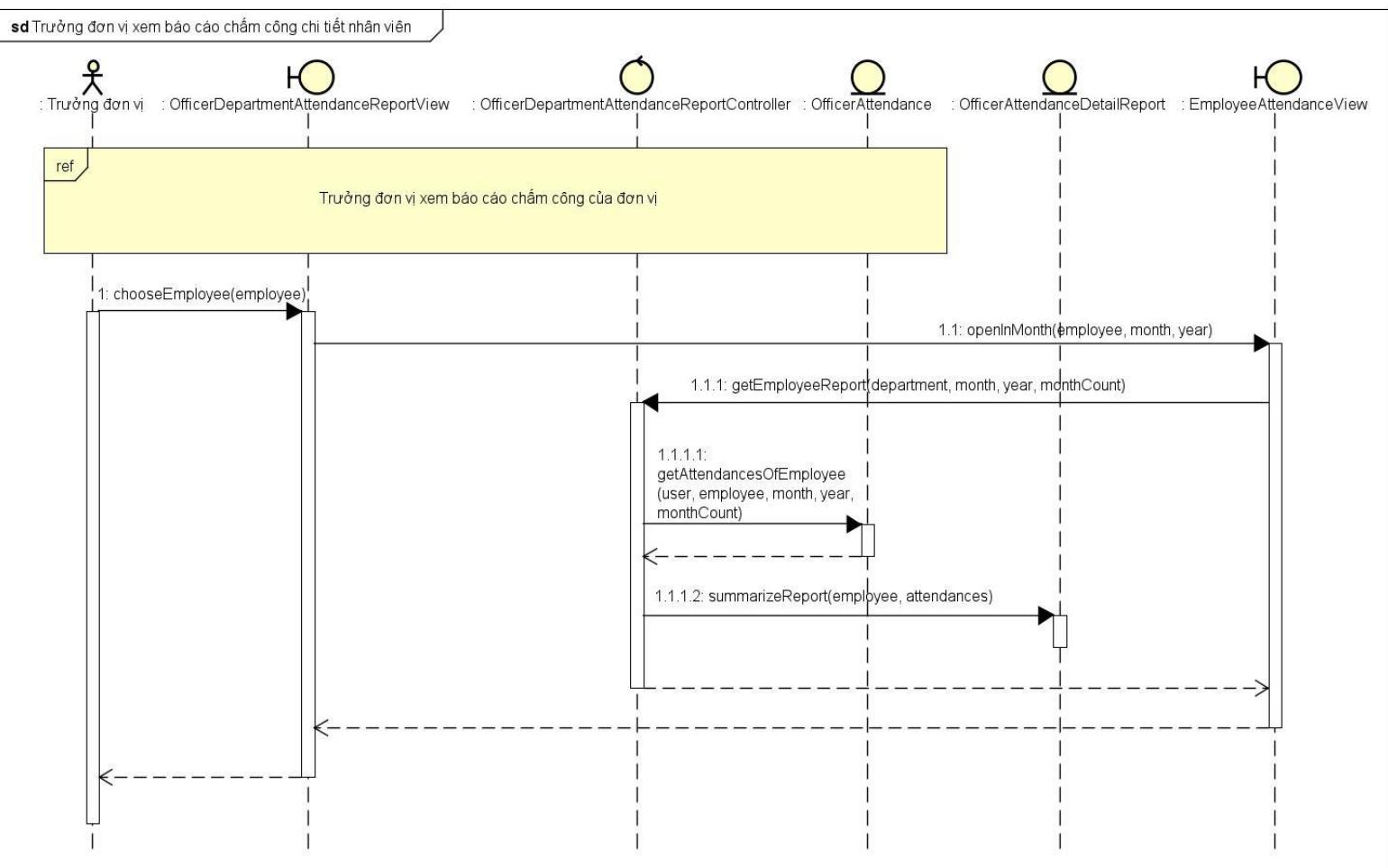
\* Scenario 3: Trưởng đơn vị xem báo cáo chấm công của đơn vị vào tháng khác (tương tự với quý, năm)



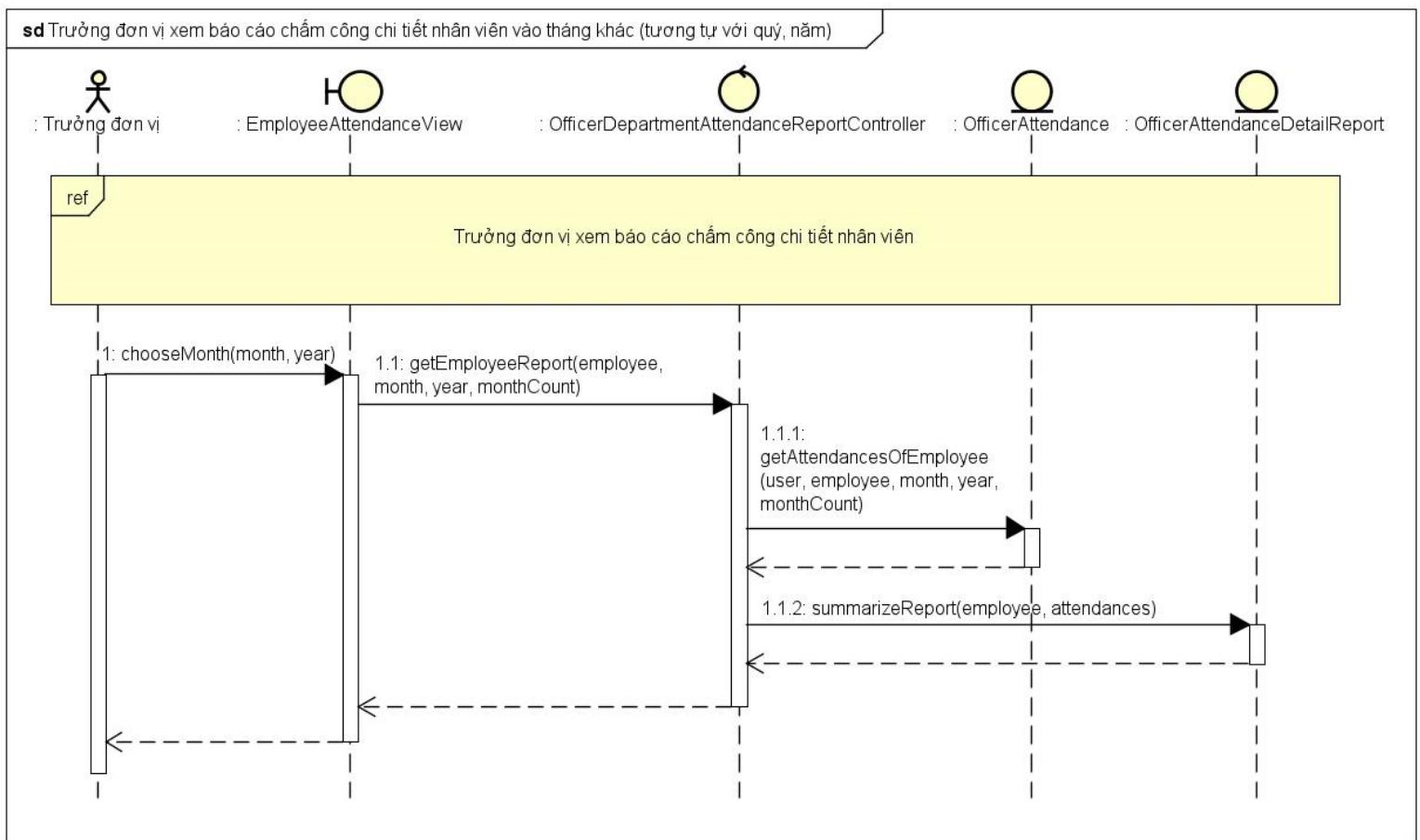
\* Scenario 4: Trưởng đơn vị tìm kiếm nhân viên



\* Scenario 5: Trưởng đơn vị xem báo cáo chấm công chi tiết nhân viên

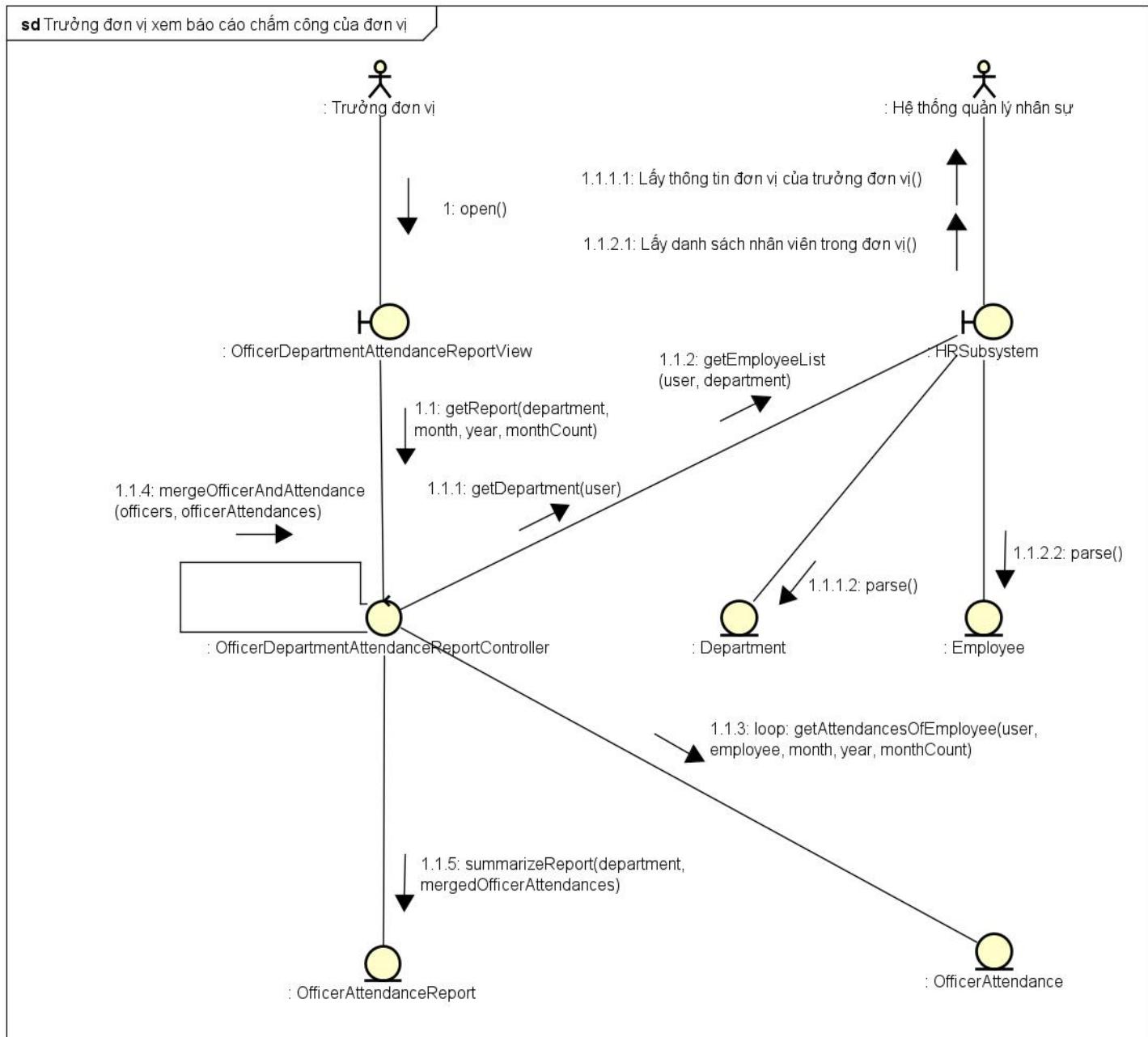


\* Scenario 6: Trưởng đơn vị xem báo cáo chấm công chi tiết nhân viên vào tháng khác (tương tự với quý, năm)



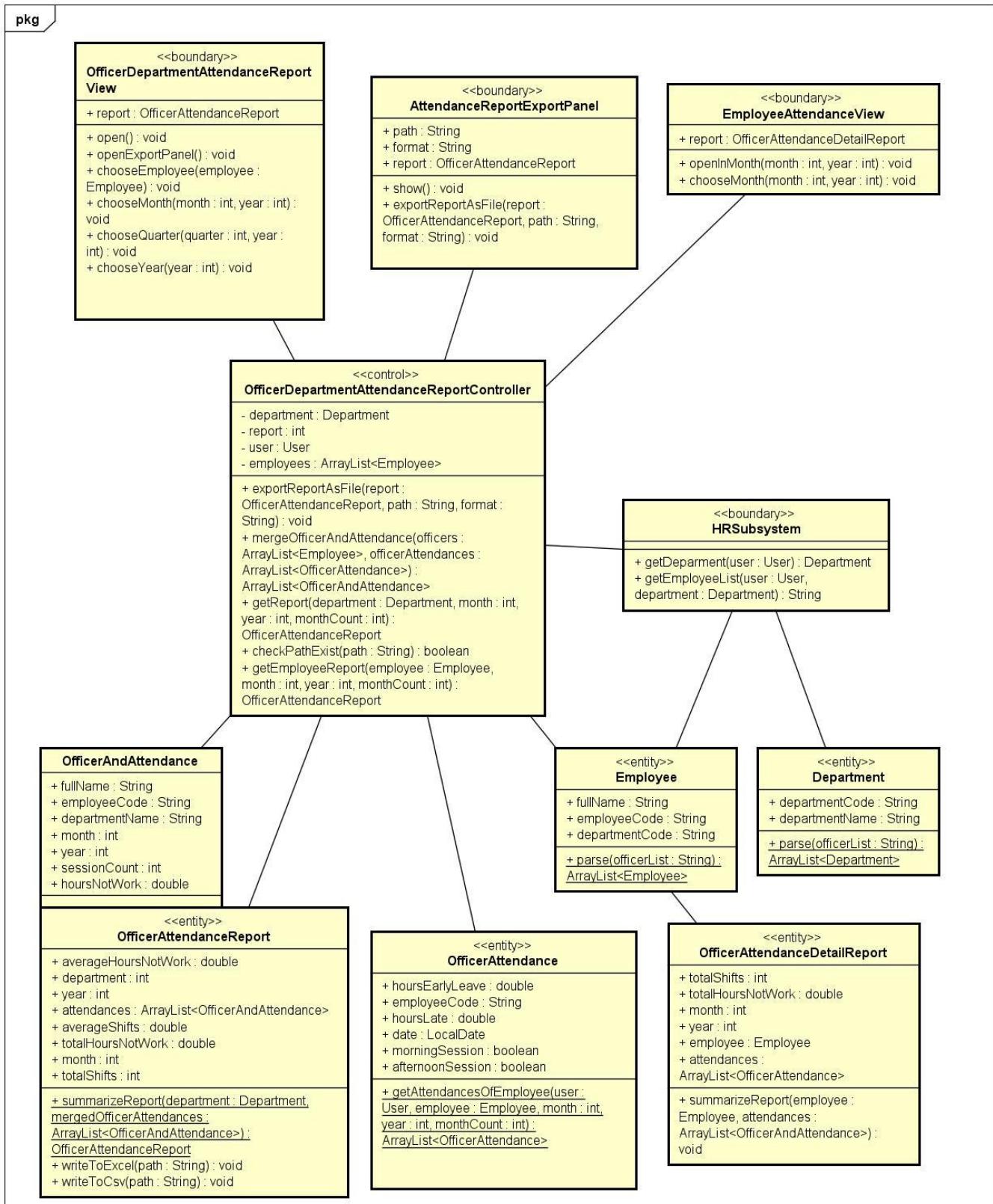
### 2.1.1.2. Biểu đồ giao tiếp biểu diễn tương đương biểu đồ trình tự

\* Main Scenario: Xem báo cáo chấm công - Trưởng đơn vị xem báo cáo chấm công của đơn vị nhân viên văn phòng



## 2.2 Thiết kế biểu đồ lớp

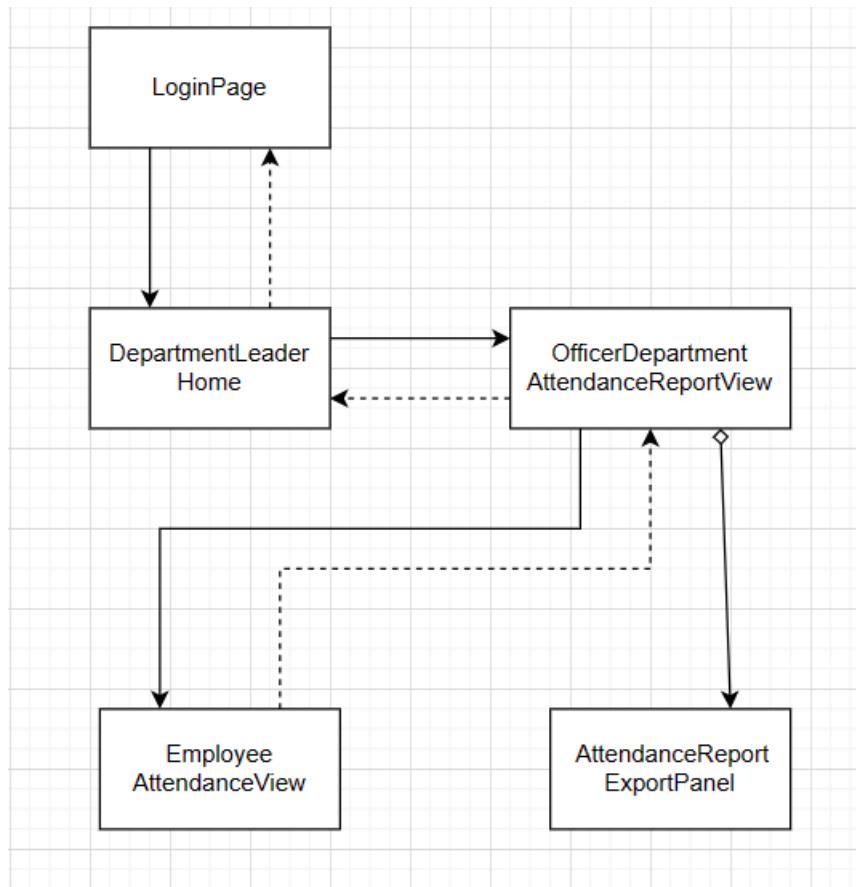
### 2.2.1 Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”



## CHƯƠNG 3. THIẾT KẾ GIAO DIỆN VÀ SUBSYSTEM (BÀI 4)

### 3.1 Sơ đồ chuyển đổi màn hình

#### 3.1.1 Sơ đồ chuyển đổi màn hình của UC Xem báo cáo chấm công cho đơn vị nhân viên văn phòng



## 3.2 Đặc tả màn hình

### 3.2.1 Đặc tả màn hình OfficerDepartmentAttendanceReportView



Nguyễn Văn Khoa  
DM-203520  
Phòng R&D  
Đăng xuất

Trang chủ

Xem báo cáo chấm công

### Xem báo cáo chấm công của đơn vị

Tháng 8/2023
Xuất báo cáo chấm công

50

Tổng giờ  
Đi muộn/  
Về sớm

30

Tổng số  
buổi đi làm

5

Giờ đi muộn/  
Về sớm  
Trung bình  
Trên người

3

Số buổi đi làm  
trung bình trên  
người

Chấm công của từng nhân viên
 Tìm kiếm

Họ tên	Mã nhân viên	Đơn vị	Tổng số buổi đi làm	Tổng số giờ đi muộn/về sớm	
Nguyễn Văn A	OF-15	R&D	30	2	<button>Xem chi tiết</button>
Nguyễn Văn A	OF-15	R&D	30	2	<button>Xem chi tiết</button>

200 bản ghi Trang hiện tại: 15/20 Trang trước Trang sau Đi đến Trang

\* Đặc tả điều khiển

Điều khiển	Hành vi	Chức năng
Hình đại diện	Initial	Hiển thị ảnh đại diện của người dùng
Text Tên người dùng	Initial	Hiển thị tên người dùng
Text Mã nhân viên	Initial	Hiển thị mã nhân viên của người dùng
Text Tên phòng ban	Initial	Hiển thị tên phòng ban của người dùng
Button Đăng xuất	Click	Đăng xuất người dùng khỏi hệ thống
Button Trang chủ	Click	Điều hướng về Trang chủ của Trưởng đơn vị
Datepicker Chọn Tháng	Click	Mở ra báo cáo chấm công đơn vị của tháng chỉ định
Button Xuất báo cáo chấm công	Click	
Text Tổng số giờ đi muộn/về sớm	Initial	Hiển thị Tổng số giờ đi muộn/về sớm của báo cáo
Text Tổng số buổi đi làm	Initial	Hiển thị Tổng ca làm của báo cáo
Text Giờ Đi muộn/Về sớm Trung bình	Initial	Hiển thị Giờ Đi muộn/ Về sớm Trung bình của báo cáo

trên người		
Text Số buổi đi làm Trung bình trên người	Initial	Hiển thị Ca làm Trung bình của báo cáo
Textfield Mã nhân viên	Type	Chứa thông tin mã nhân viên cần tìm kiếm trong báo cáo
Button Tìm kiếm	Click	Tìm kiếm hàng có mã nhân viên trong Textfield Mã nhân viên và hiển thị cho người dùng
Table Chấm công của từng nhân viên	Initial	Hiển thị thông tin chấm công của từng nhân viên trong báo cáo
Button Xem chi tiết	Click	Điều hướng đến trang Hiển thị thông tin chấm công chi tiết của nhân viên
Text Tổng số bản ghi	Initial	Hiển thị tổng số dòng của bảng
Text Trang hiện tại	Initial	Hiển thị vị trí trang hiện tại trên tổng số trang của Table Chấm công của từng nhân viên
Button Trang trước	Click	Hiển thị trang trước của Table Chấm công của từng nhân viên
Button Trang sau	Click	Hiển thị trang sau của Table Chấm công của từng nhân viên
Textfield Trang	Type	Trang mong muốn hiển thị trên Table Chấm công của từng nhân viên
Button Đi đến	Click	Hiển thị trang mong muốn trên Table Chấm công của từng nhân viên

\* Đặc tả dữ liệu trên từng dòng của bảng chấm công của từng nhân viên

Tên trường dữ liệu	Kiểu dữ liệu	Kiểu cách
Họ tên	String	Căn trái
Mã nhân viên	String	Căn giữa
Đơn vị	String	Căn trái
Tổng số buổi đi làm	Int	Căn phải
Tổng số giờ đi muộn/về sớm	Double	Căn phải

### 3.2.2 Đặc tả màn hình EmployeeAttendanceView



Nguyễn Văn Khoa  
DM-203520  
Phòng R&D  
[Đăng xuất](#)

[Trang chủ](#)  
[Xem báo cáo chấm công](#)

Quay lại [Xem báo cáo chấm công chi tiết của nhân viên Nguyễn Văn A \[OF-15\]](#)

Tháng 8/2023

**Thông tin tổng hợp**

50  
Tổng giờ  
Đi muộn/  
Về sớm

30  
Tổng số  
buổi đi  
làm

**Chấm công theo ngày**

Officer	Ngày	Thứ 2 1/2/2021	Thứ 3 2/2/2021	Thứ 4 3/2/2021	...
OF-15	Sáng (morningSession)	có	có	có	
	Chiều (afternoonSession)	không	có	có	
	Đi muộn (hoursLate)	0.25	0	0	
	Về sớm (hoursEarlyLeave)	0.1	0	0	

Tuần hiện tại: 1/4 [Tuần trước](#) [Tuần sau](#) [Đi đến](#) [Tuần](#)

\* Đặc tả điều khiển

Điều khiển	Hành vi	Chức năng
Hình đại diện	Initial	Hiển thị ảnh đại diện của người dùng
Text Tên người dùng	Initial	Hiển thị tên người dùng
Text Mã nhân viên	Initial	Hiển thị mã nhân viên của người dùng
Text Tên phòng ban	Initial	Hiển thị tên phòng ban của người dùng
Button Đăng xuất	Click	Đăng xuất người dùng khỏi hệ thống
Button Trang chủ	Click	Điều hướng về Trang chủ của Trưởng đơn vị
Button Xem báo cáo chấm công	Click	Quay về màn hình OfficerDepartmentAttendanceReportView
Text Xem báo cáo chấm công chi tiết của nhân viên	Initial	Hiển thị tiêu đề của màn hình + tên nhân viên + mã nhân viên
Datepicker Chọn Tháng	Click	Mở ra báo cáo chấm công chi tiết của nhân viên đó trong tháng chỉ định
Text Tổng số giờ đi muộn/về sớm	Initial	Hiển thị Tổng số giờ đi muộn/về sớm nhân viên đó trong tháng

Text Tổng số buổi đi làm	Initial	Hiển thị Tổng ca làm của nhân viên đó trong tháng
Table Chấm công theo ngày	Initial	Hiển thị thông tin chấm công theo từng ngày, từng ca của nhân viên trong tháng
Text Tuần hiện tại	Initial	Hiển thị vị trí số thứ tự tuần trong tháng đang chứa các ngày trong bảng, trên tổng số tuần mà tháng đó có
Button Tuần trước	Click	Hiển thị các ngày trong tuần trước của Table Chấm công theo ngày
Button Tuần sau	Click	Hiển thị các ngày trong tuần sau của Table Chấm công theo ngày
Textfield Trang	Type	Tuần mong muốn hiển thị trên Table Chấm công theo ngày
Button Đi đến	Click	Hiển thị tuần mong muốn trên Table Chấm công theo ngày

### 3.2.3 Đặc tả màn hình AttendanceReportExportPanel

Họ tên	Mã nhân viên	Đơn vị	Tháng	Tổng số buổi	Tổng số giờ đi muộn/về sớm	
Nguyễn Văn A	OF-15	R&D	6	30	2	Xem chi tiết
Nguyễn Văn A	OF-15	R&D	6	30	2	Xem chi tiết

200 bản ghi      Trang hiện tại: 15/20      Trang trước      Trang sau      Đi đến      Trang

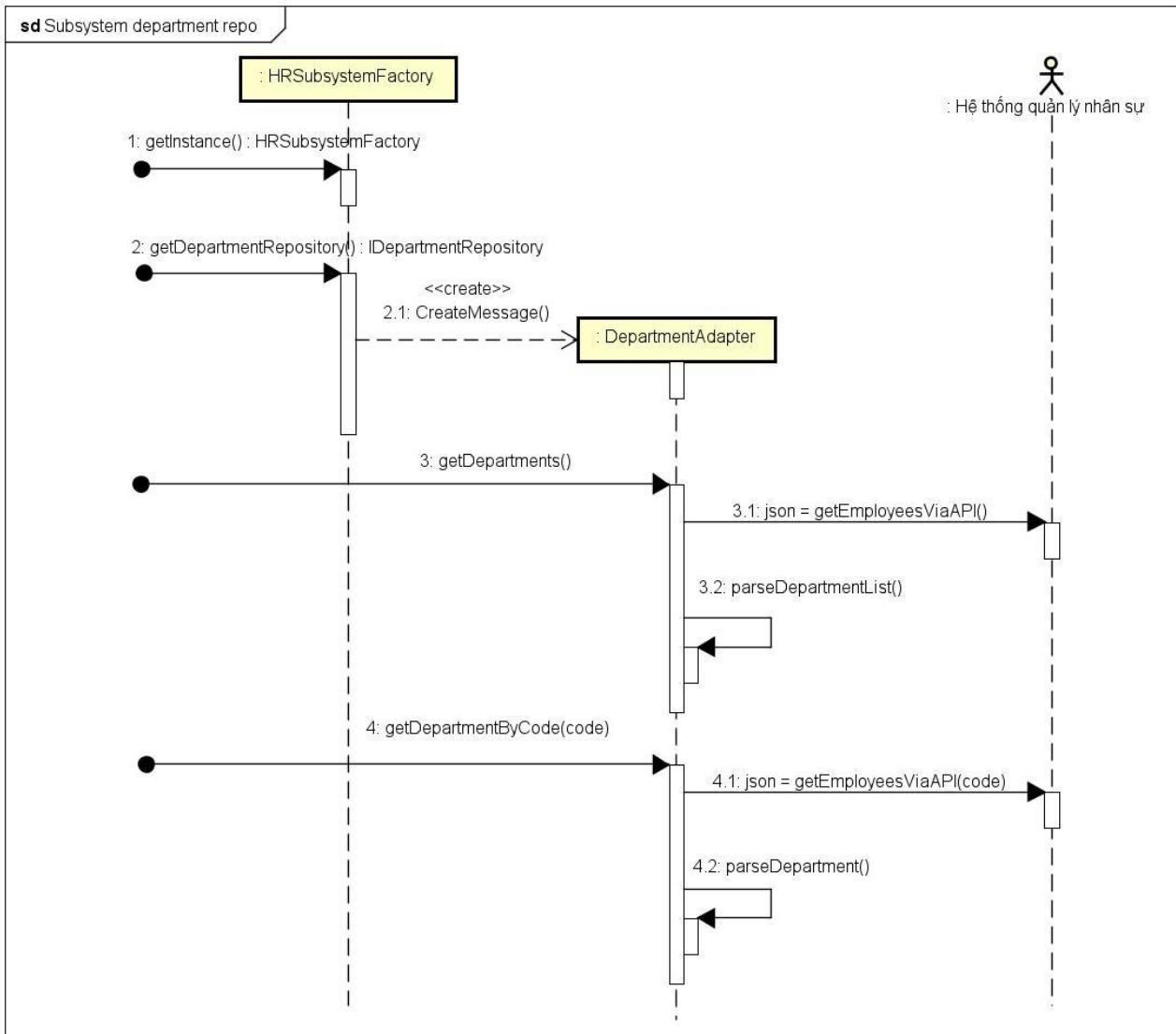
\* Đặc tả điều khiển

Điều khiển	Hành vi	Chức năng
Combobox Định dạng	Click	Chọn định dạng file báo cáo
Textfield Vị trí lưu	Type	Nhập đường dẫn đến nơi muốn lưu file, cũng

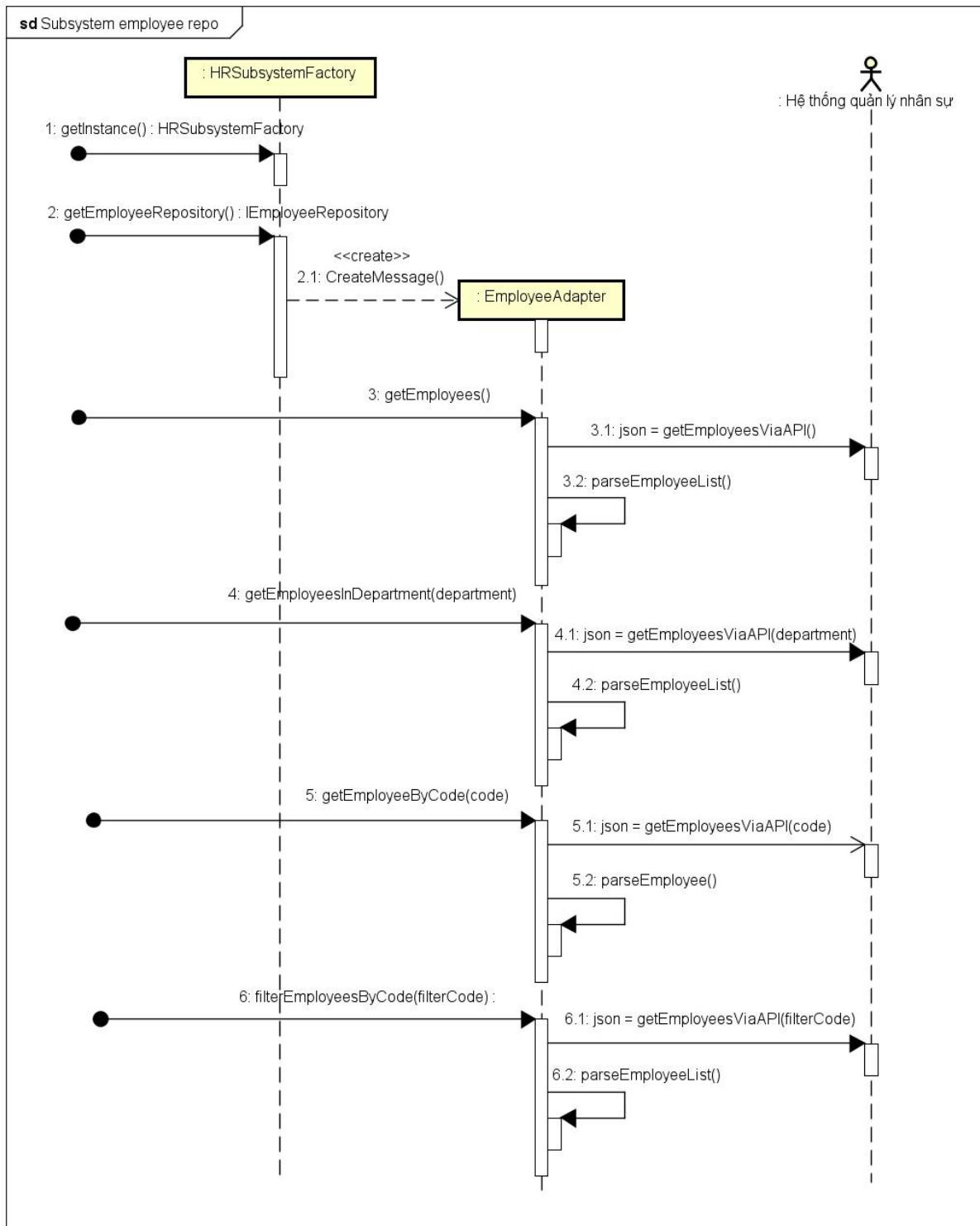
		như tùy chọn tên file
Button Chọn vị trí lưu	Click	Gọi hộp thoại hệ thống để chọn vị trí lưu, và đặt đường dẫn vào Textfield Vị trí lưu
Button Xuất	Click	Thực hiện quá trình xuất file báo cáo và lưu vào vị trí chỉ định với định dạng chỉ định

### 3.3 Biểu đồ trình tự cho Subsystem

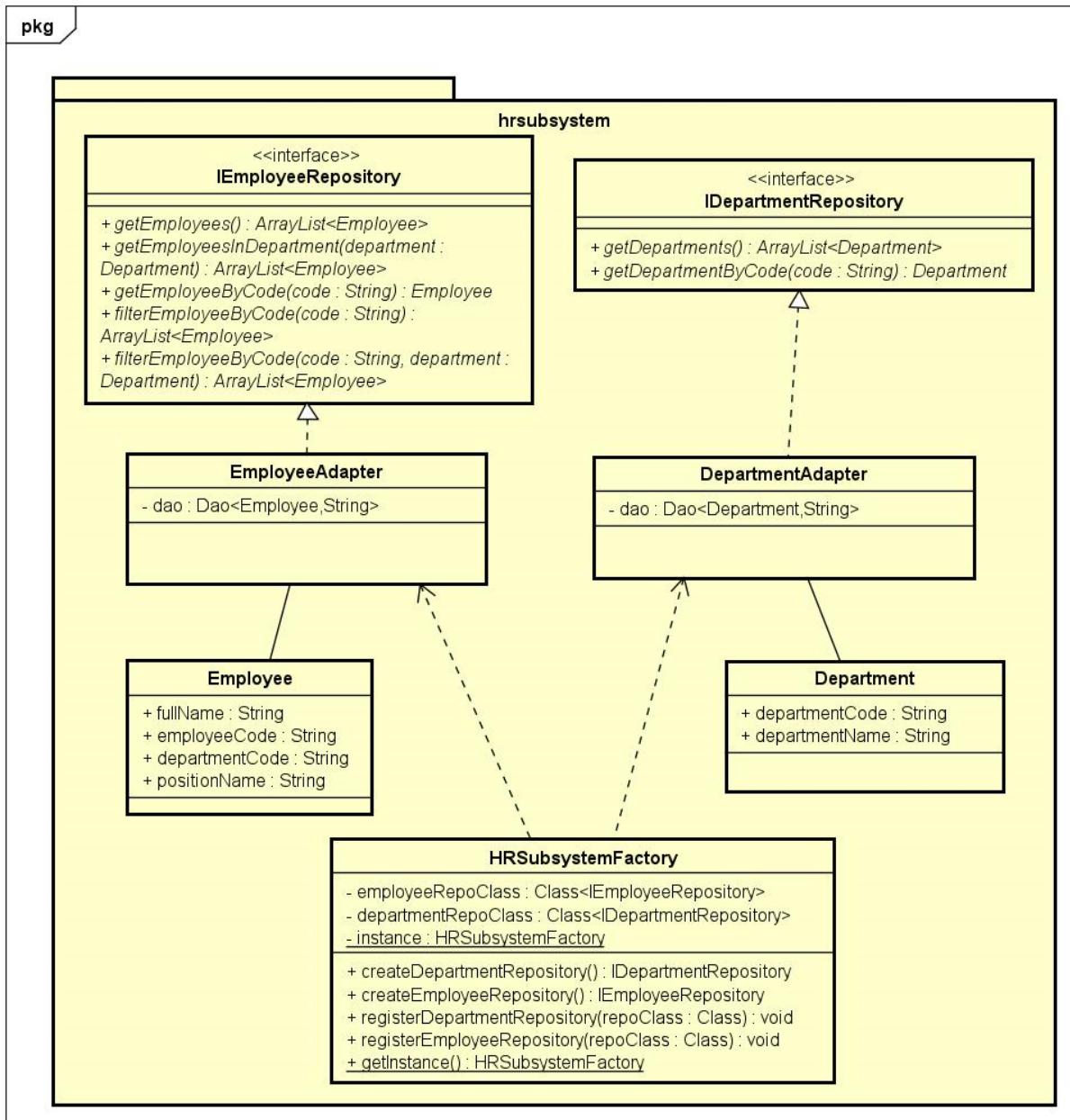
\*Biểu đồ trình tự liên quan đến DepartmentAdapter



\*Biểu đồ trình tự liên quan đến EmployeeAdapter



### 3.4 Biểu đồ lớp cho Subsystem

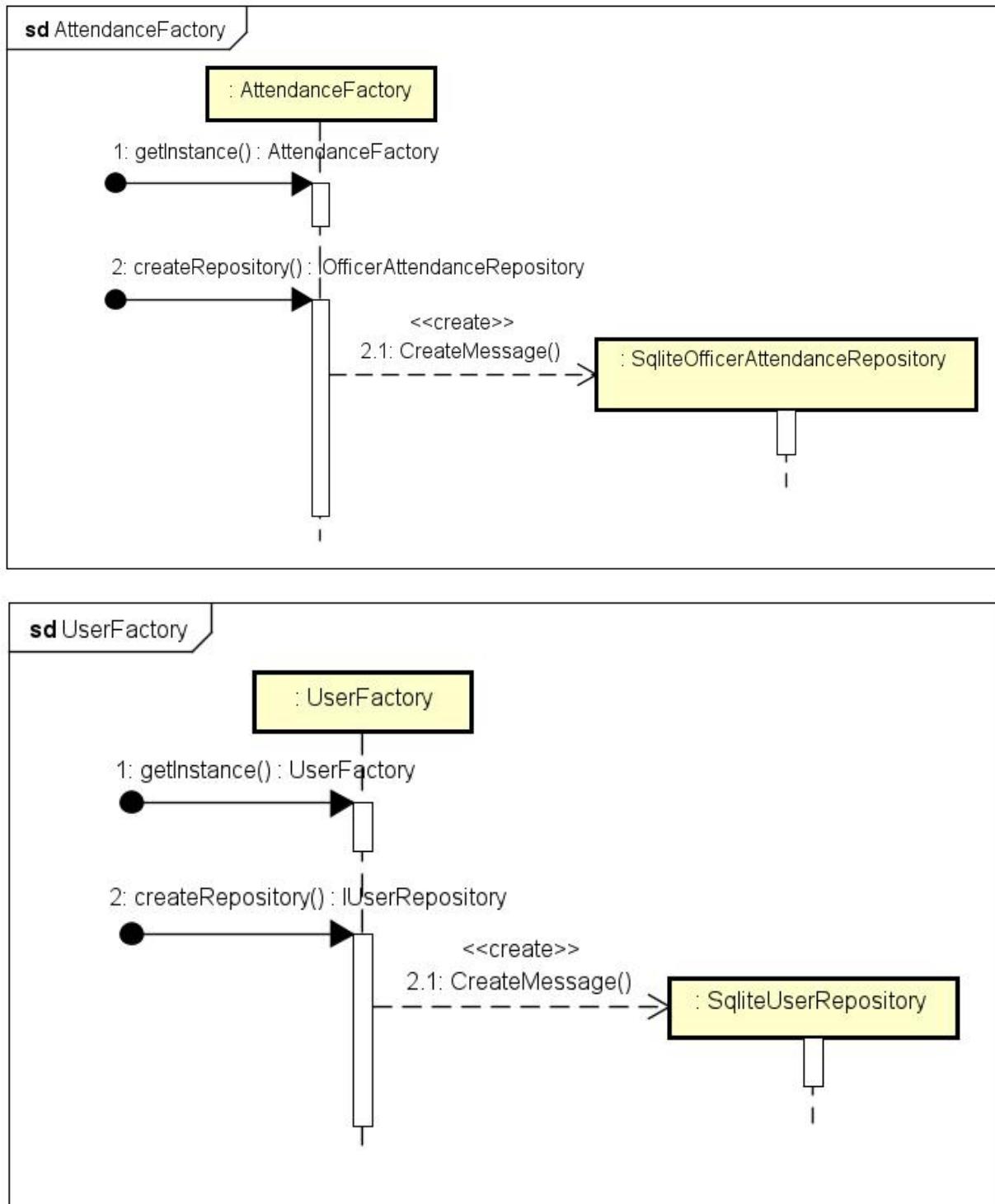


## CHƯƠNG 4. THIẾT KẾ CHI TIẾT LỚP (BÀI 5)

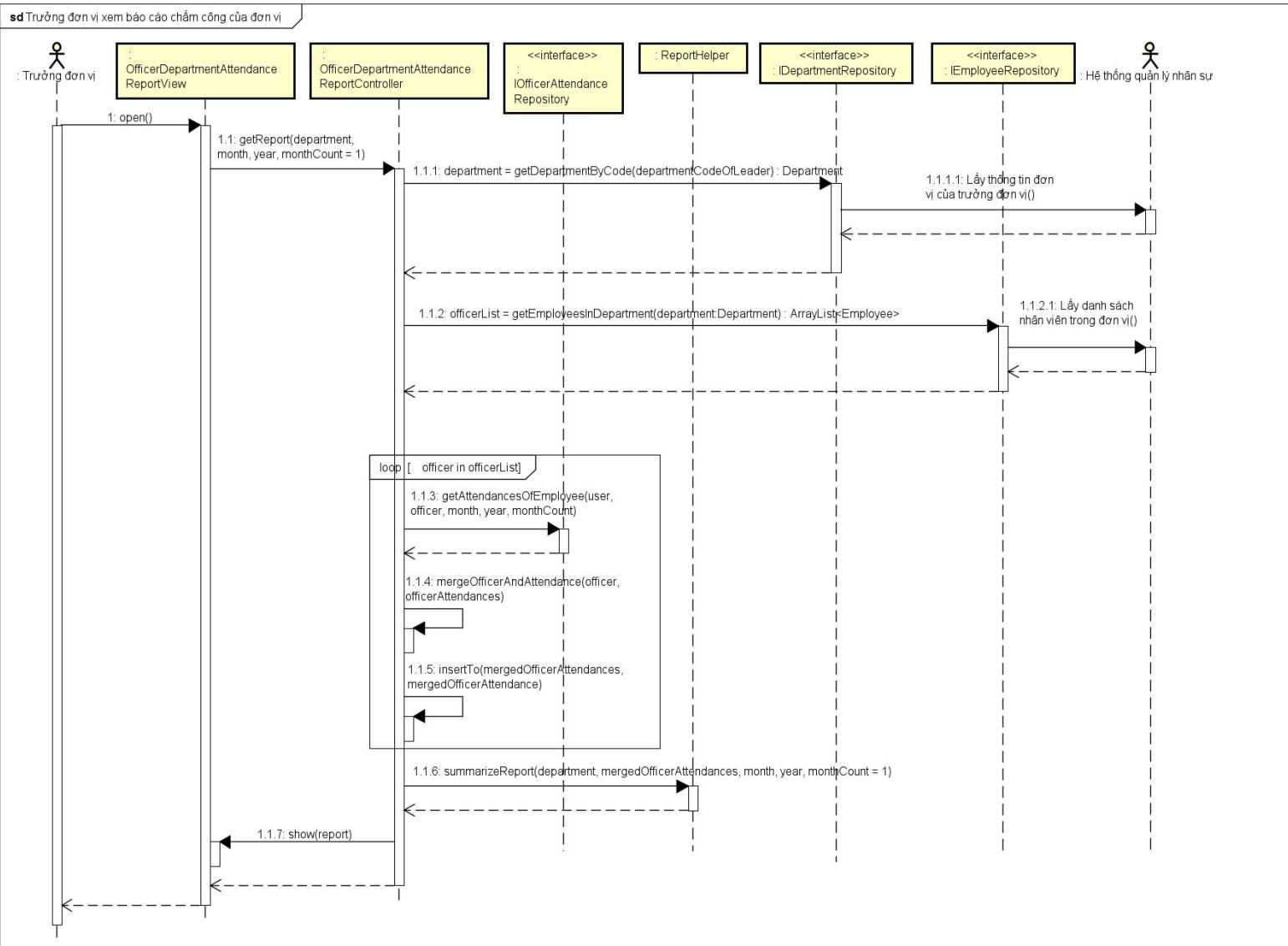
### 4.1 Thiết kế biểu đồ trình tự mức thiết kế

#### 4.1.1 Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”

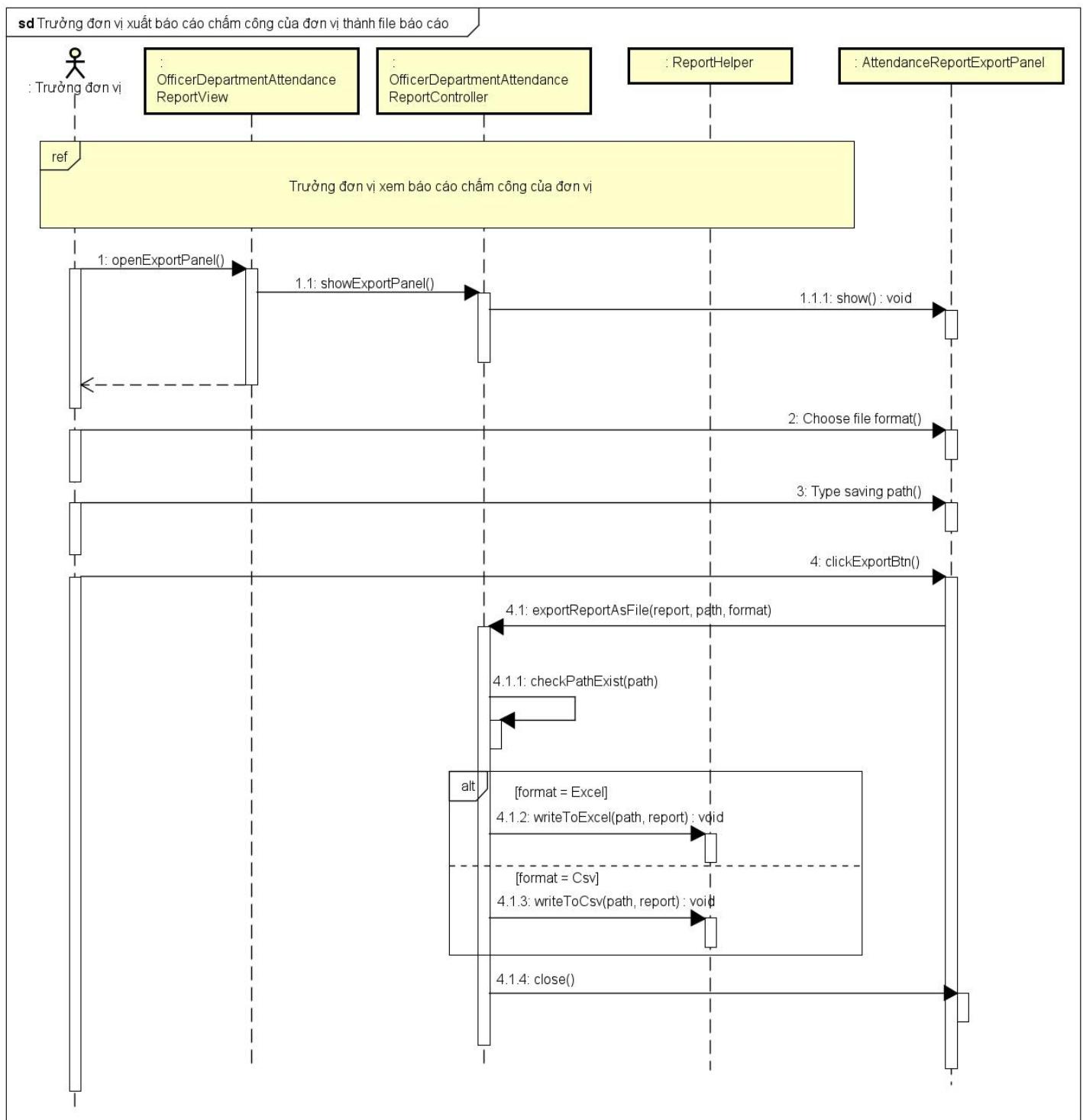
\* Tạo ra các đối tượng IUserRepository, IOfficerAttendanceRepository



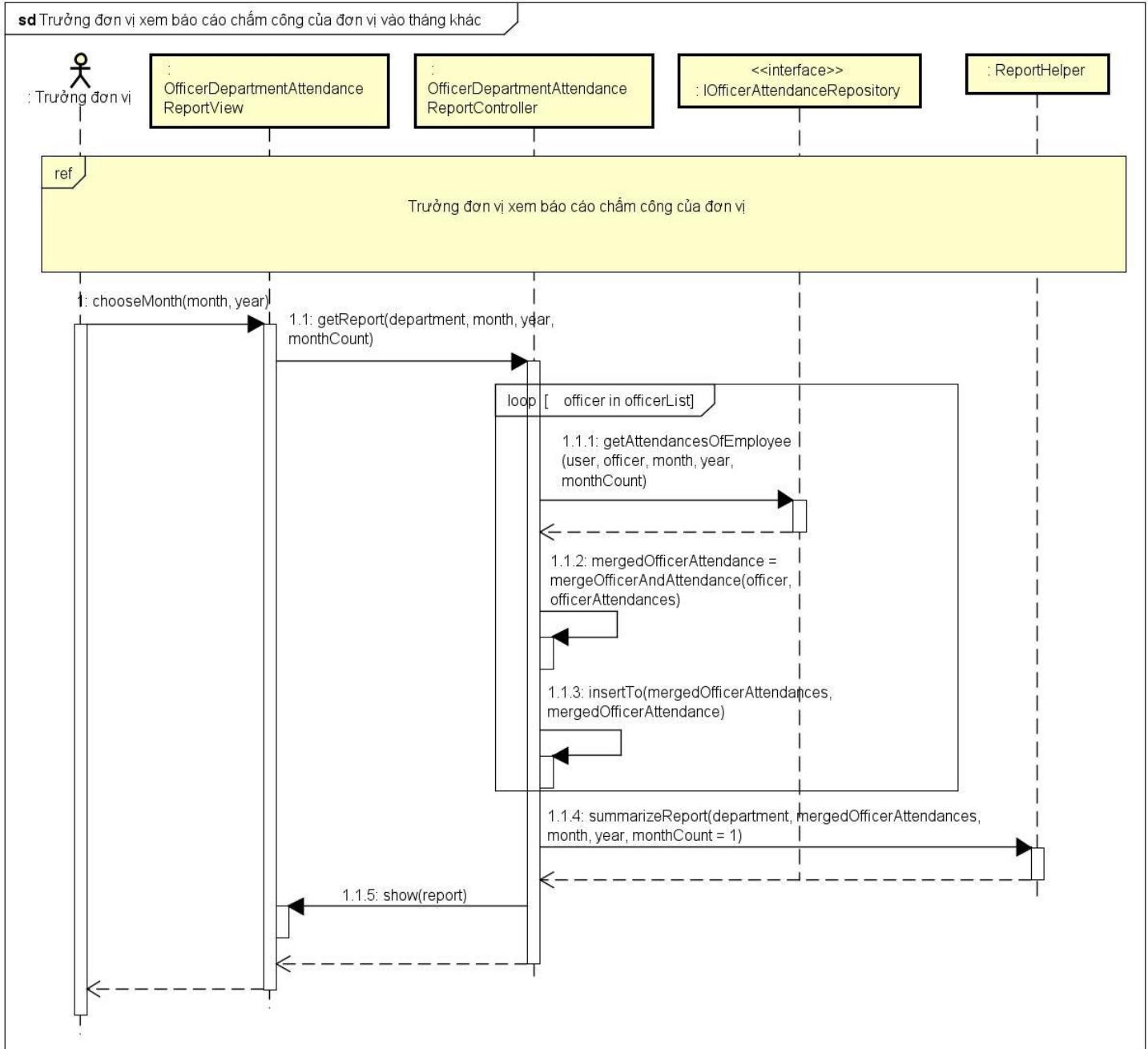
\* Main Scenario: Trưởng đơn vị xem báo cáo chấm công của đơn vị



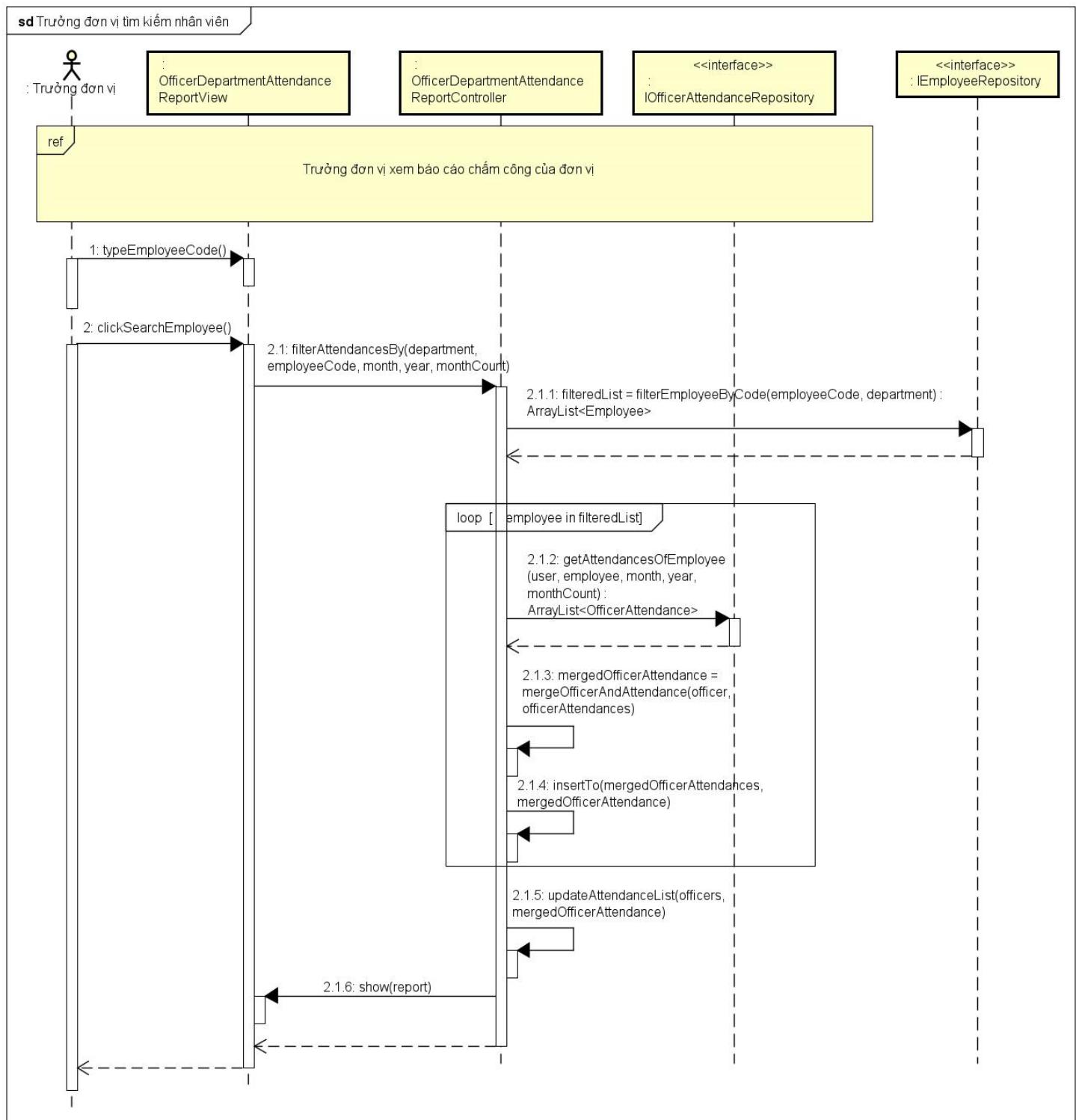
\* Scenario 2: Trưởng đơn vị xuất báo cáo chấm công của đơn vị thành file báo cáo



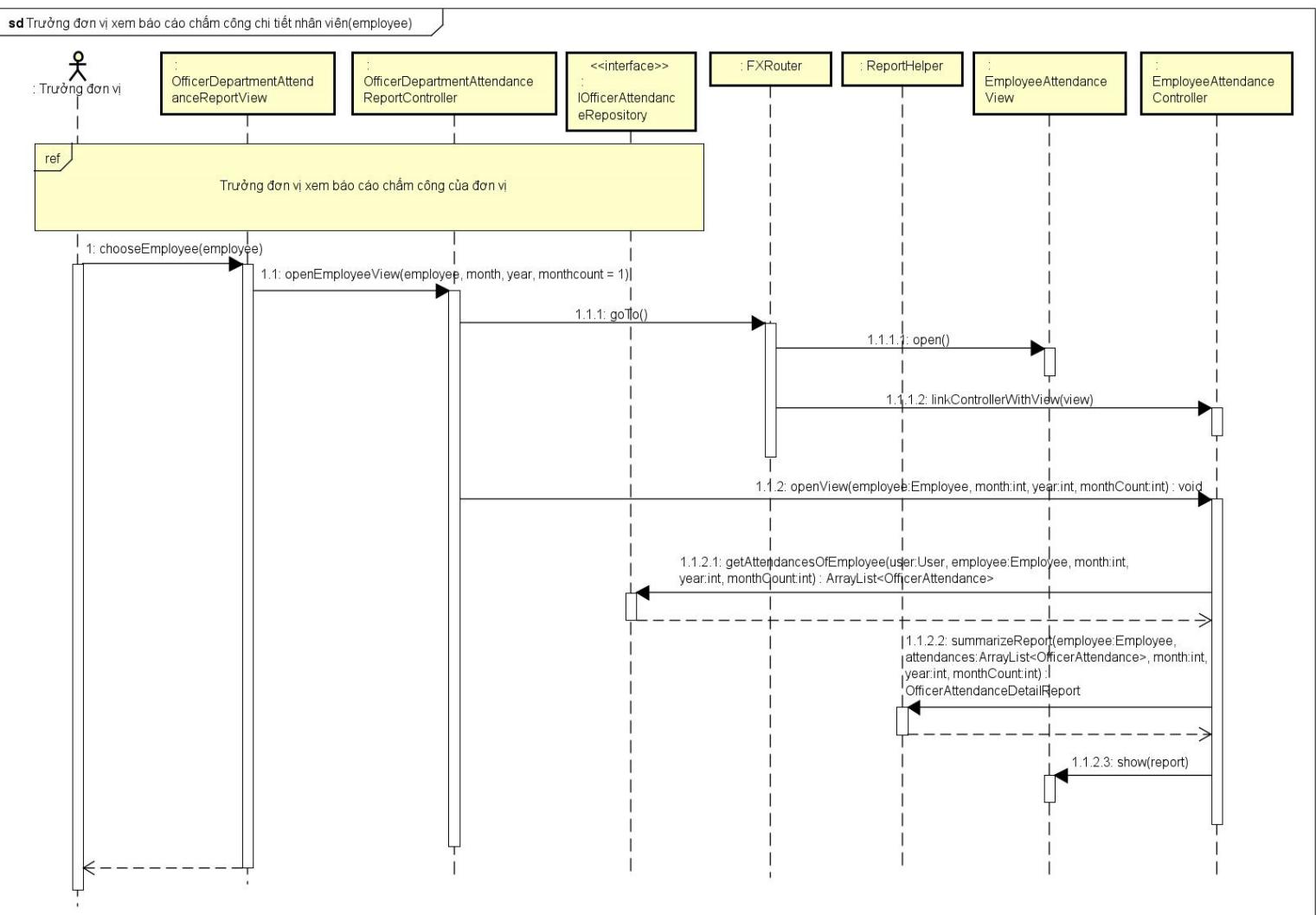
\* Scenario 3: Trưởng đơn vị xem báo cáo chấm công của đơn vị vào tháng khác (tương tự với quý, năm)



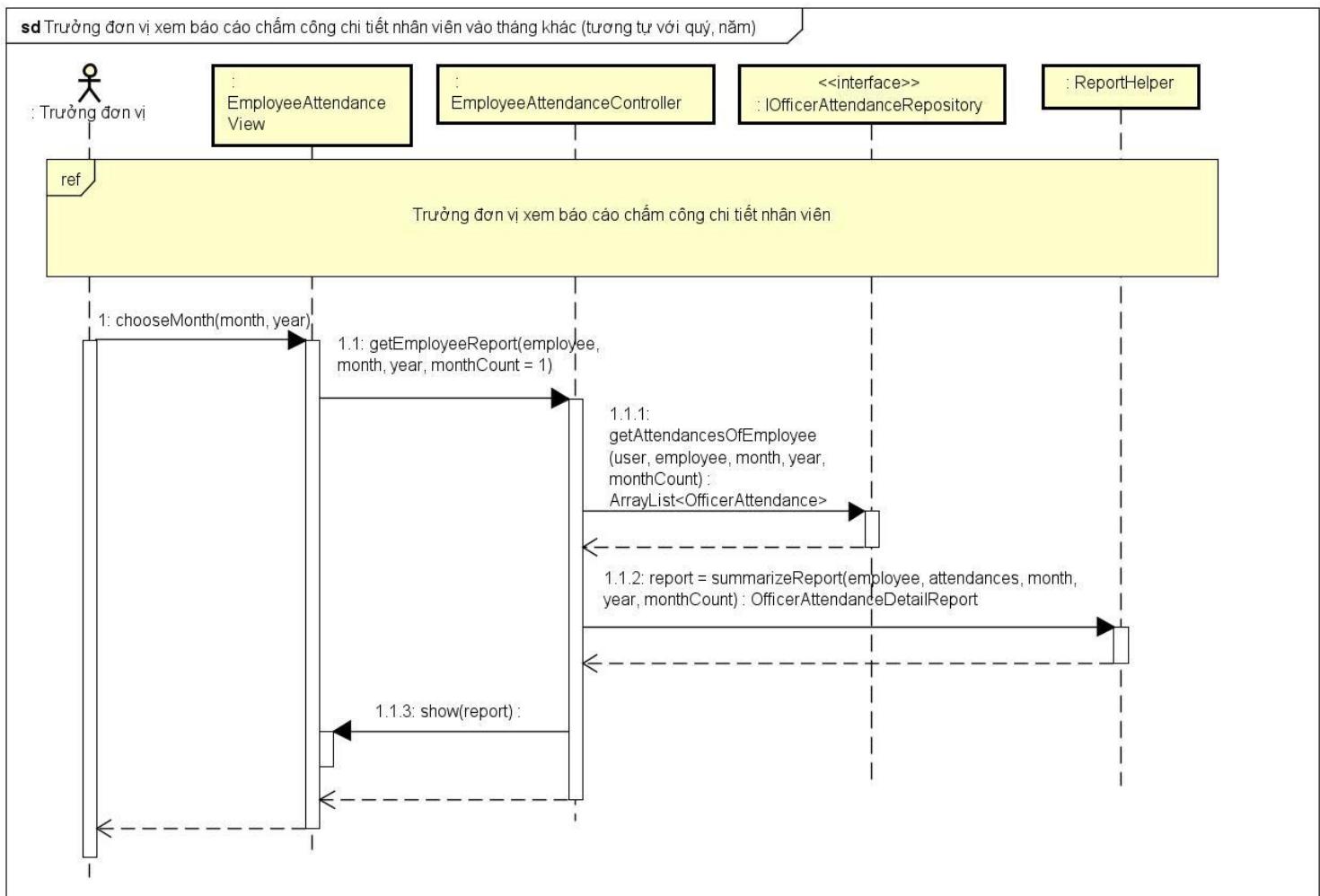
## \* Scenario 4: Trưởng đơn vị tìm kiếm nhân viên



\* Scenario 5: Trưởng đơn vị xem báo cáo chấm công chi tiết nhân viên

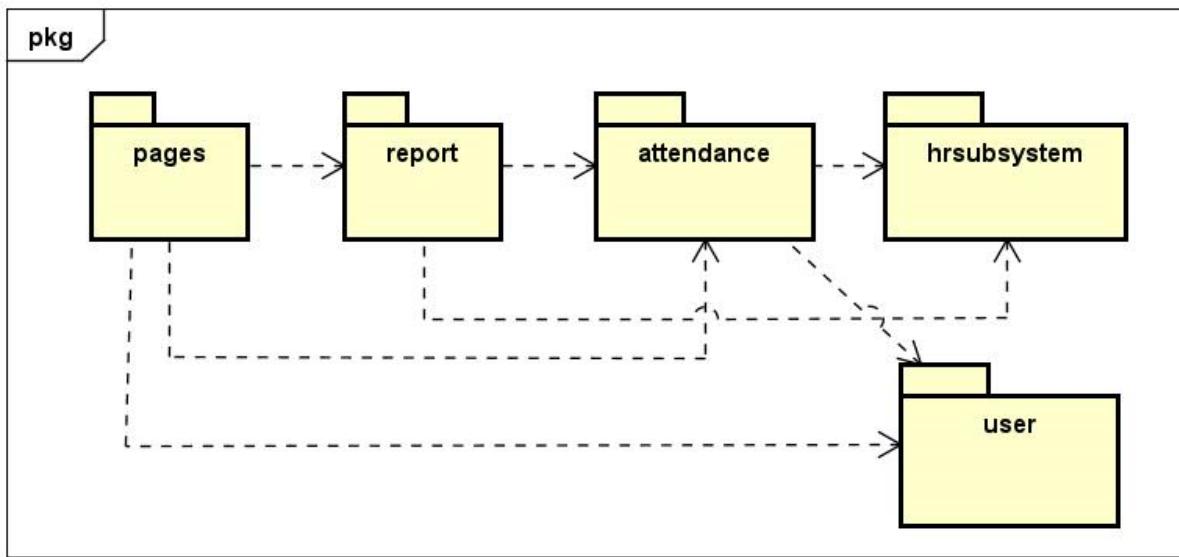


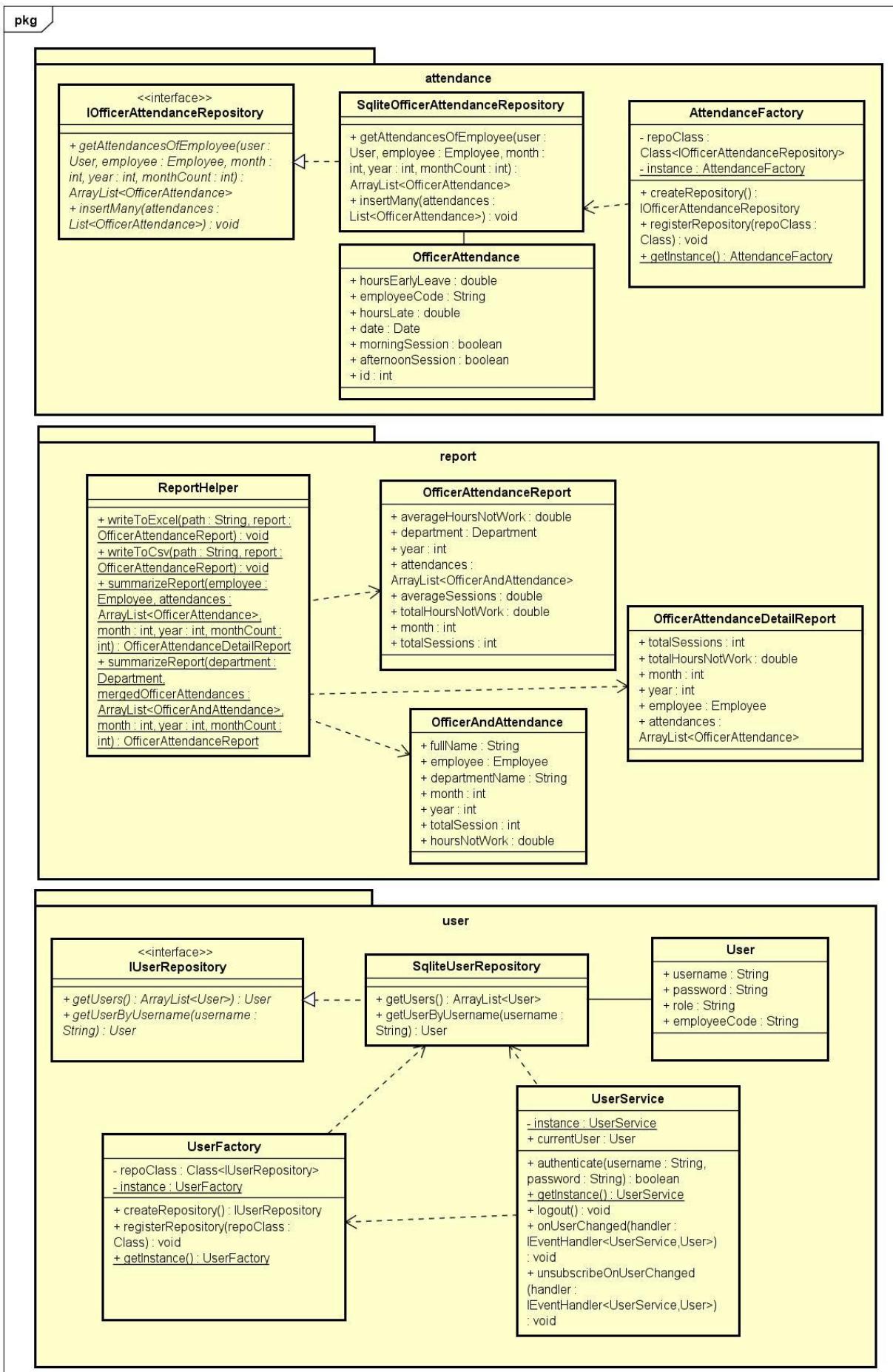
\* Scenario 6: Trưởng đơn vị xem báo cáo chấm công chi tiết nhân viên vào tháng khác (tương tự với quý, năm)

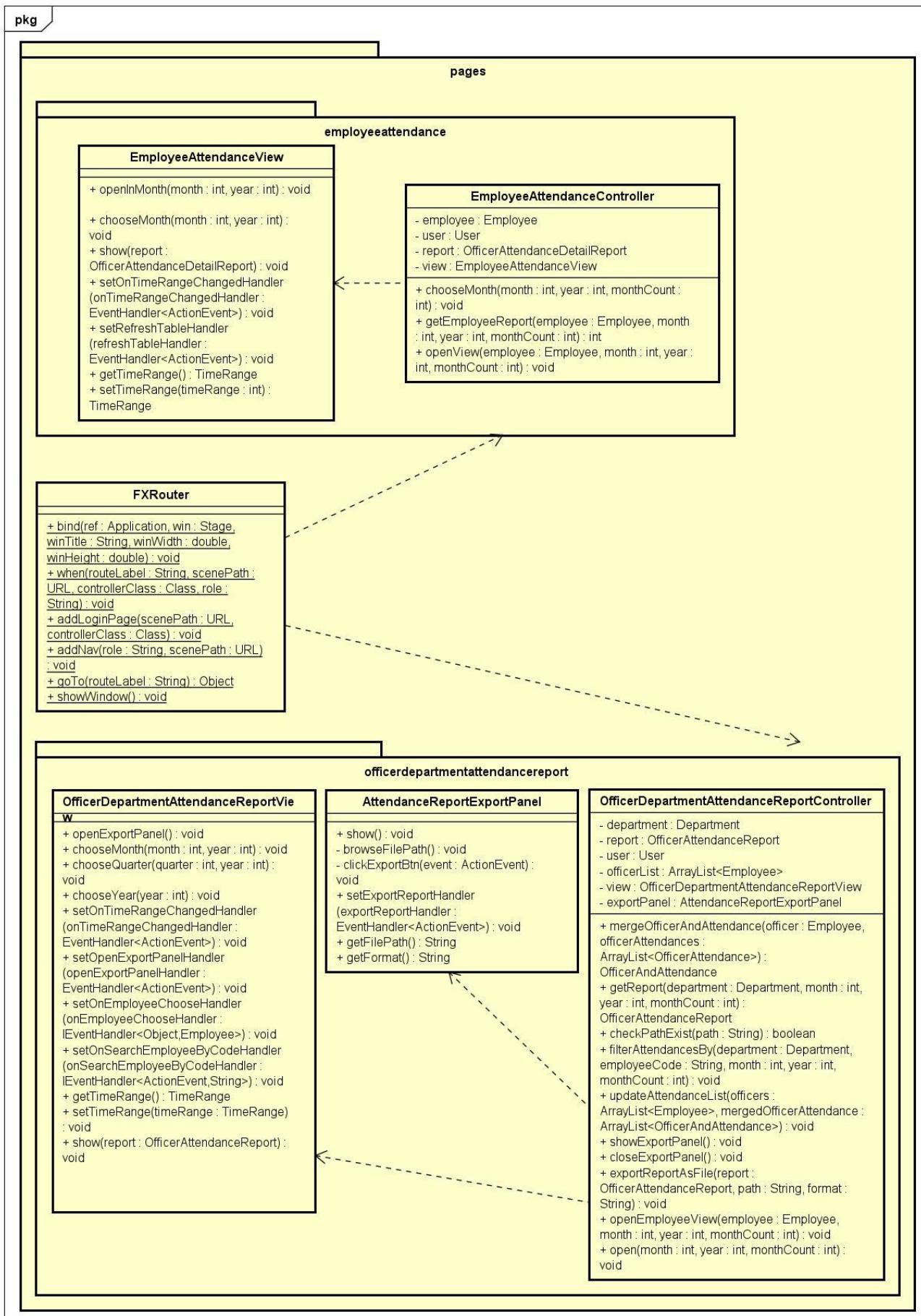


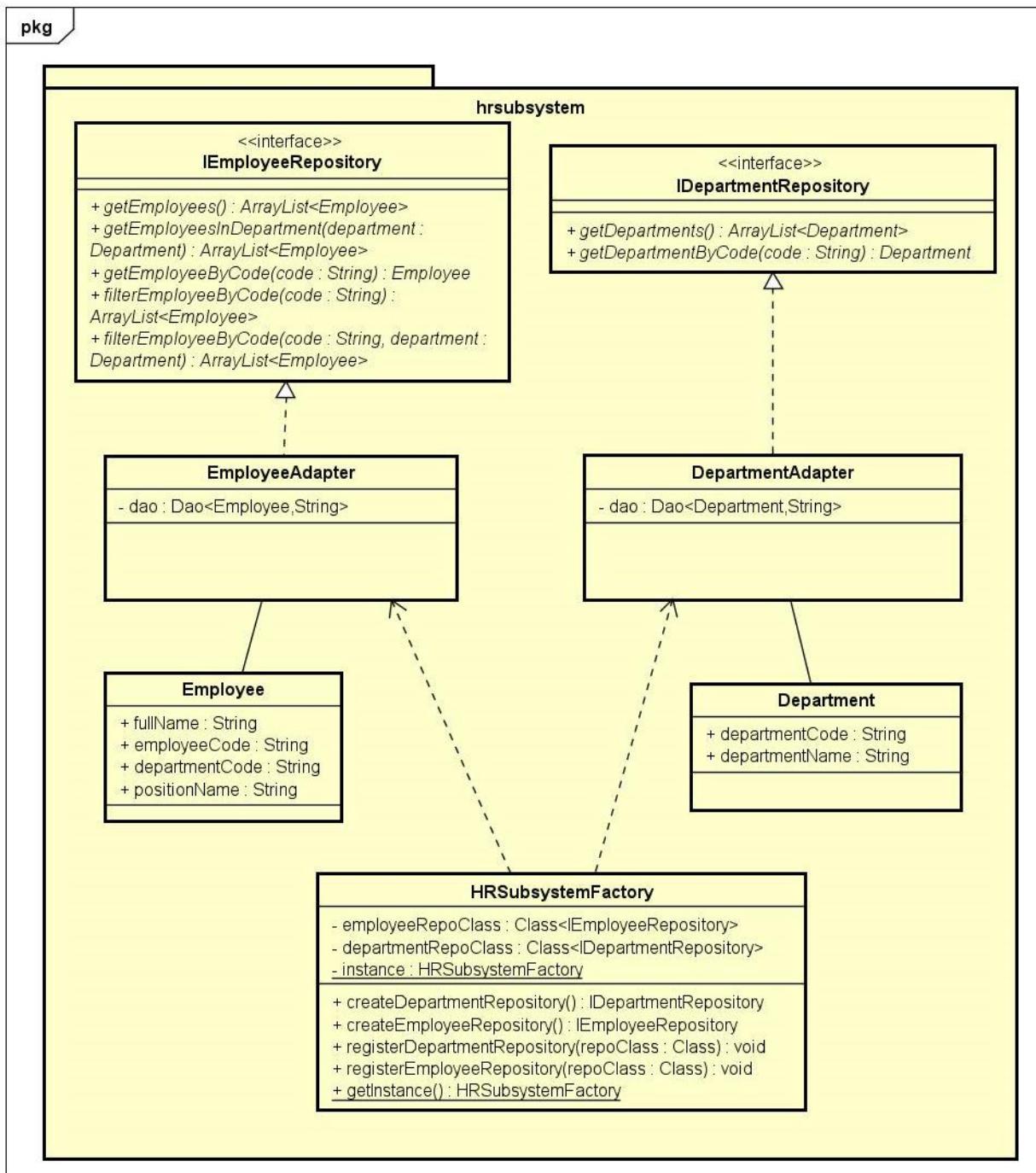
## 4.2 Biểu đồ lớp mức thiết kế

### 4.2.1 Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”









## CHƯƠNG 5. CẢI THIỆN BIẾU ĐỒ TRÌNH TỰ VÀ LỚP MỨC THIẾT KẾ (BÀI 6)

### 5.1 Cách chỉnh sửa và ý nghĩa mang lại

#### 5.1.1 Áp dụng mẫu thiết kế Singleton

Mẫu thiết kế Singleton là một trong những mẫu thiết kế phổ biến trong lập trình, được sử dụng trong những lớp **UserService**, **UserFactory**, **AttendanceFactory**, **HRSubsystemFactory** để đảm bảo rằng:

- Một lớp chỉ có duy nhất một thể hiện.
- Có thể dễ dàng truy cập toàn cục đối tượng.
- Không cần phải quan tâm cách khởi tạo đối tượng.

#### 5.1.2 Áp dụng mẫu thiết kế Factory

Mẫu thiết kế Singleton là một trong những mẫu thiết kế phổ biến trong lập trình, được sử dụng trong những lớp **IEmployeeRepository**, **IDepartmentRepository**, **IOfficerAttendanceRepository**, **IUserRepository** với các lớp Factory gồm **UserFactory**, **AttendanceFactory**, **HRSubsystemFactory** được nhằm mục đích:

- **Tạo đối tượng mà không cần biết chi tiết cụ thể:** Mẫu Factory giúp ẩn đi logic cụ thể của việc tạo đối tượng. Khi sử dụng một phương thức tạo đối tượng từ Factory, bạn không cần biết chi tiết cách đối tượng đó được khởi tạo hoặc được xử lý bên trong.
- **Đảm bảo tuân thủ nguyên tắc "Open/Closed":** Mẫu Factory giúp đảm bảo rằng khi thêm một triển khai mới của interface mà không cần phải sửa đổi mã nguồn hiện tại. Thay vào đó, chỉ cần thêm một lớp mới vào Factory.
- **Tăng sự linh hoạt của hệ thống:** Factory có thể được cấu hình để tạo ra các đối tượng thuộc các lớp con khác nhau dựa trên điều kiện nào đó. Điều này tăng sự linh hoạt và tái sử dụng mã nguồn.
- **Giảm sự phụ thuộc với đối tượng cụ thể:** Khi sử dụng một Factory để tạo đối tượng thì không cần phải biết chi tiết cụ thể cách khởi tạo của đối tượng đó. Điều này giảm sự phụ thuộc và giúp giữ cho mã nguồn linh hoạt hơn.

#### 5.1.3 Áp dụng nguyên tắc Cohesion and Coupling

Các lớp được phân lẻ thành các gói **pages**, **user**, **report**, **attendance**, **hrsubsystem** với mục đích tăng tính kết dính cho các lớp trong một gói, và giảm sự phụ thuộc giữa các gói với nhau. Những lớp có cùng một nhiệm vụ, cùng một nghiệp vụ thì được gom nhóm lại một gói, và mỗi gói lại có một nhiệm vụ khác nhau.

Cohesion giúp tăng tính đóng gói (encapsulation) và giảm sự phụ thuộc giữa các module. Nó làm cho mã nguồn dễ đọc, dễ bảo trì, và dễ kiểm thử. Cohesion cao cũng giúp tái sử dụng mã nguồn dễ dàng hơn.

Coupling giúp tăng tính linh hoạt và tái sử dụng của hệ thống. Khi các thành phần độc lập và không phụ thuộc quá mức lớn vào nhau, sự thay đổi trong một thành phần không ảnh hưởng quá lớn đến các thành phần khác. Điều này làm cho hệ thống linh hoạt và dễ mở rộng, bảo trì.

#### 5.1.4 Áp dụng nguyên lý SOLID

- *S - Một class chỉ nên giữ một trách nhiệm duy nhất*

Tất cả các lớp đều chỉ thực hiện một trách nhiệm duy nhất, như các lớp Model **OfficerAttendance**, **OfficerAndAttendance**, **User**, **Employee**, **Department** chỉ chứa dữ liệu về đối tượng, trong khi các lớp **EmployeeAdapter**, **DepartmentAdapter**, **SqliteOfficerAttendanceRepository**, **SqliteUserRepository** phụ trách việc thao tác lưu trữ đối tượng trong database hoặc tương tác với hệ thống bên ngoài để lấy đối tượng, ...

- *O - Có thể thoải mái mở rộng 1 module, nhưng hạn chế sửa đổi bên trong module đó*  
Phần lớn các hàm đều được thiết kế theo dạng subscribe, unsubscribe để tránh việc thay đổi logic hàm, như các hàm **createRepository** và **registerRepository** trong lớp **UserFactory**, **AttendanceFactory**

- *L - Trong một chương trình, các object của class con có thể thay thế class cha mà không làm thay đổi tính đúng đắn của chương trình*

Không có lớp nào sử dụng kế thừa nên xem như hợp nguyên tắc.

- *I - Thay vì dùng 1 interface lớn, ta nên tách thành nhiều interface nhỏ, với nhiều mục đích cụ thể*

Subsystem được tách thành **IEmployeeRepository**, **IDepartmentRepository** nhằm để thực hiện truy vấn với từng kiểu dữ liệu lần lượt là **Employee**, **IDepartment** thay vì dồn vào 1 lớp duy nhất.

- *D - Các module cấp cao không nên phụ thuộc vào các module cấp thấp. Cả 2 nên phụ thuộc vào abstraction. Interface (abstraction) không nên phụ thuộc vào chi tiết, mà ngược lại. (Các class giao tiếp với nhau thông qua interface, không phải thông qua implementation.)*

Thay vì phải để mã phụ thuộc vào **EmployeeAdapter**, **DepartmentAdapter**, **SqliteOfficerAttendanceRepository**, **SqliteUserRepository** thì cho những đoạn mã phụ thuộc thay thế lần lượt vào **IEmployeeRepository**, **IDepartmentRepository**, **IOfficerAttendanceRepository**, **IUserRepository** sẽ mang lại những điều sau:

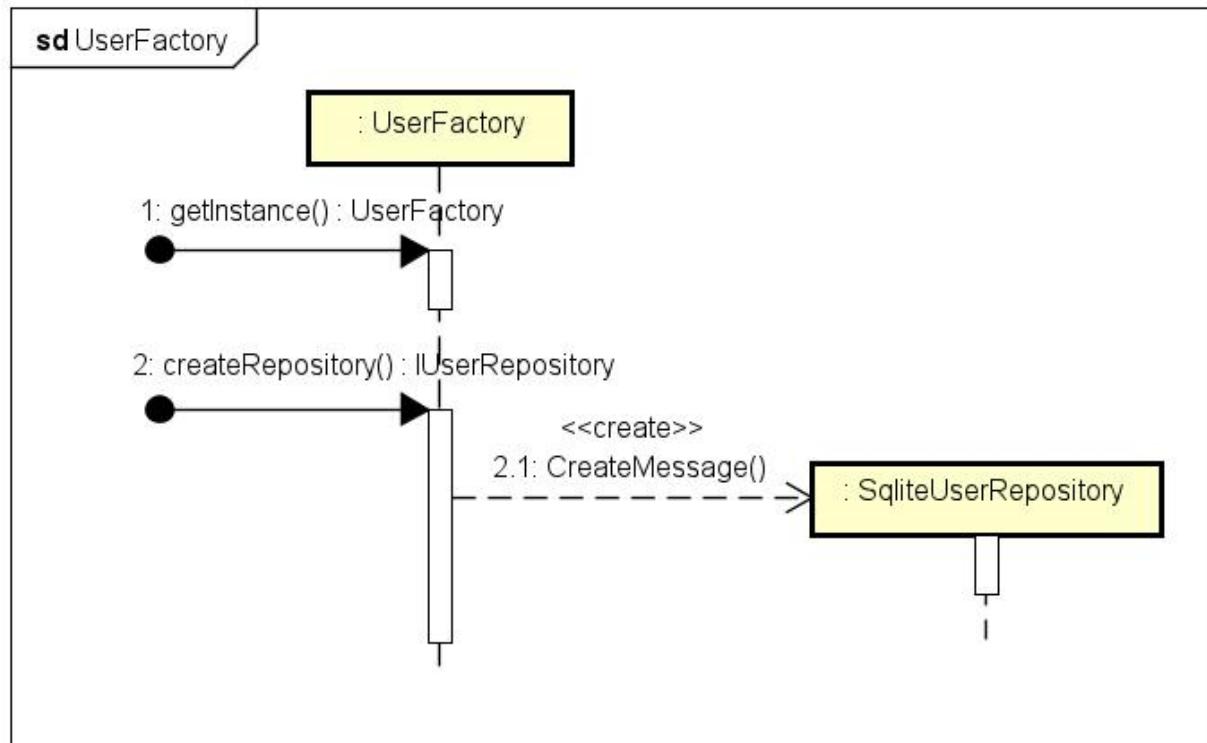
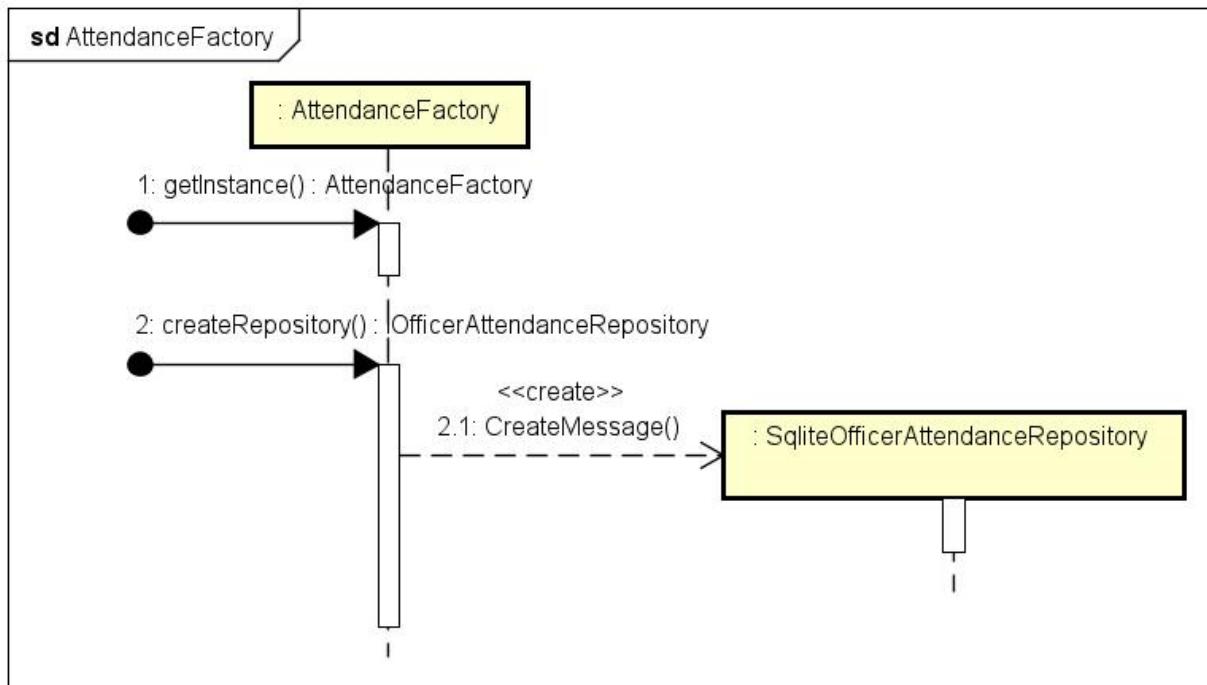
- **Định hình Cấu trúc Hệ thống:** Nguyên tắc Dependency Convention giúp định hình cấu trúc của hệ thống bằng cách quy ước về cách các thành phần phụ thuộc lẫn nhau. Điều này tạo ra một cấu trúc tự nhiên và dễ hiểu, giúp người đọc mã nguồn dễ dàng theo dõi và hiểu cấu trúc tổng thể của hệ thống.
- **Giảm Độ phức tạp:** Nguyên tắc Dependency Convention giúp giảm độ phức tạp của hệ thống bằng cách hạn chế sự phụ thuộc giữa các thành phần (thay vào đó là phụ thuộc vào interface). Điều này giúp giảm nguy cơ xung đột và khó khăn trong quá trình phát triển và bảo trì.
- **Dễ Bảo trì và Nâng cấp:** Nguyên tắc Dependency Convention giúp việc bảo trì và nâng cấp trở nên dễ dàng hơn. Các thay đổi có thể được thực hiện mà không làm ảnh hưởng đến toàn bộ hệ thống, chỉ bằng cách thay đổi implement này bằng implement khác của interface, và người phát triển có thể tập trung vào các thành phần cụ thể mà họ đang làm việc.
- **Tăng Tính tái sử dụng:** Dependency Convention thúc đẩy việc sử dụng lại mã nguồn bằng cách tạo ra các thành phần độc lập, có thể được tái sử dụng trong các phần khác nhau của hệ thống hoặc trong các dự án khác.

- **Tăng Tính Mô đun hóa:** Nguyên tắc này hỗ trợ tính mô đun hóa của hệ thống, trong đó mỗi mô đun (hoặc thành phần) có trách nhiệm và chức năng cụ thể. Điều này giúp tạo ra các mô đun độc lập, dễ kiểm thử và duy trì.

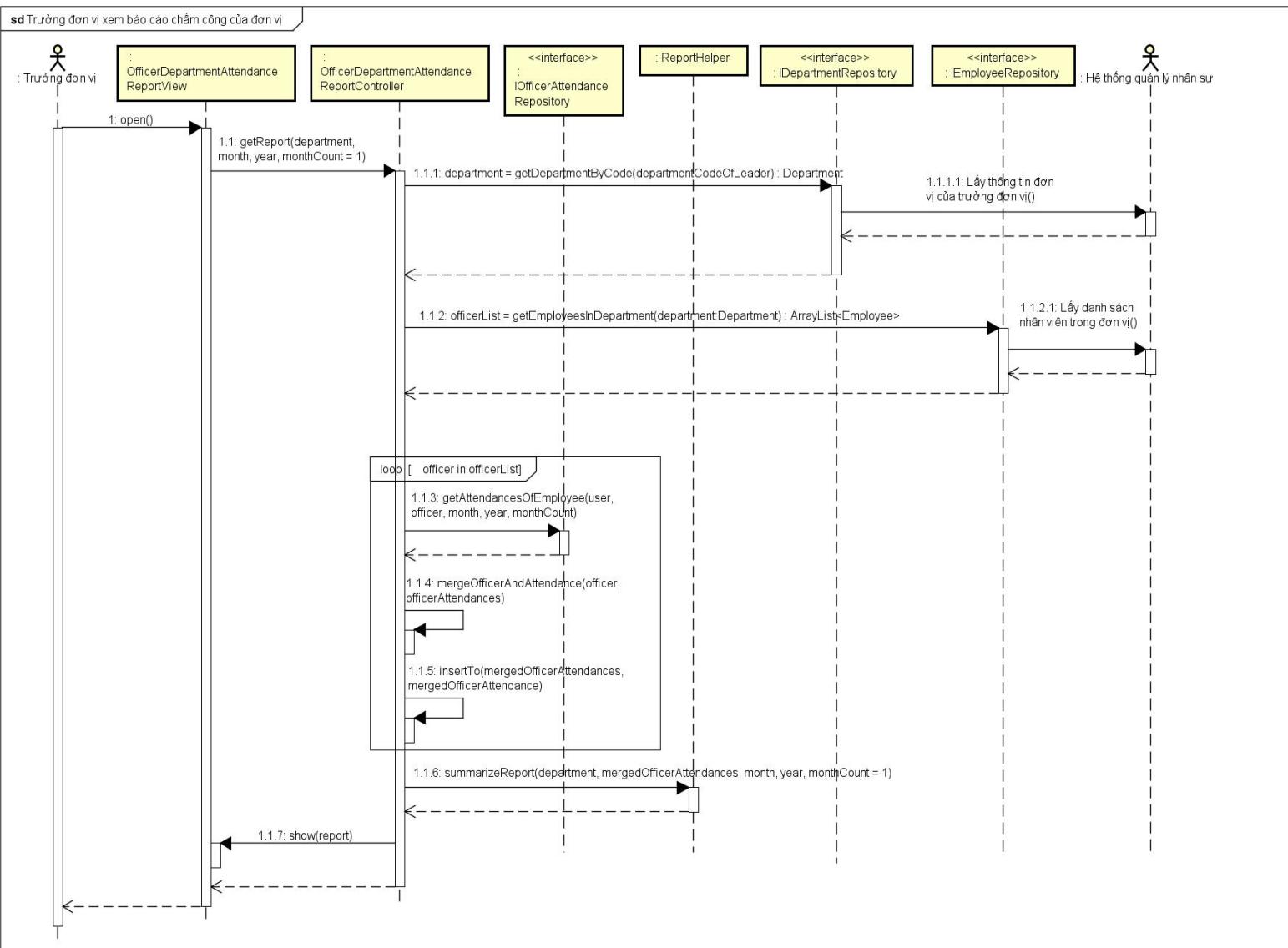
## 5.2 Thiết kế biểu đồ trình tự mức thiết kế

### 5.2.1 Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”

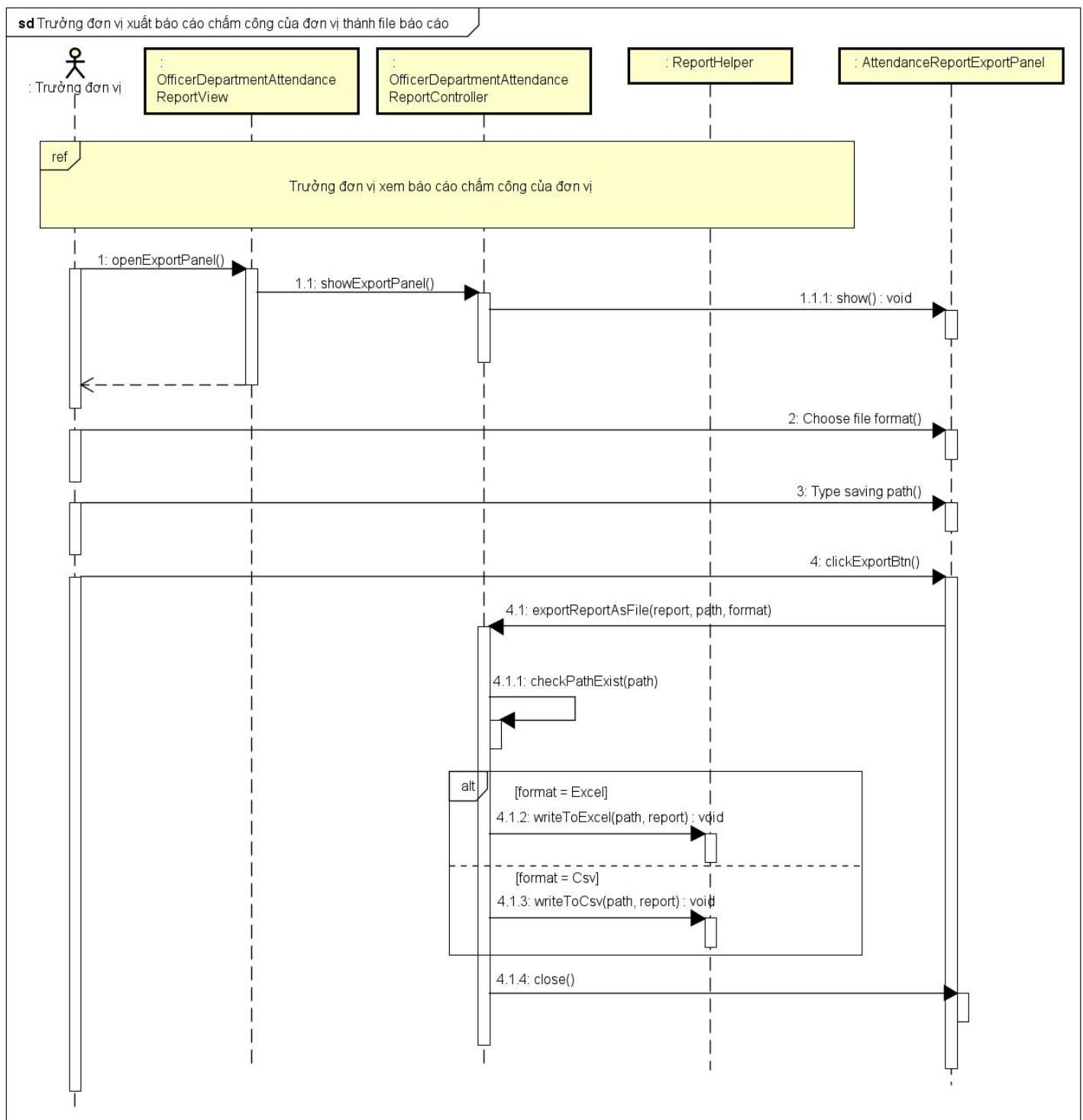
\* Tạo ra các đối tượng I UserRepository, IOfficerAttendanceRepository



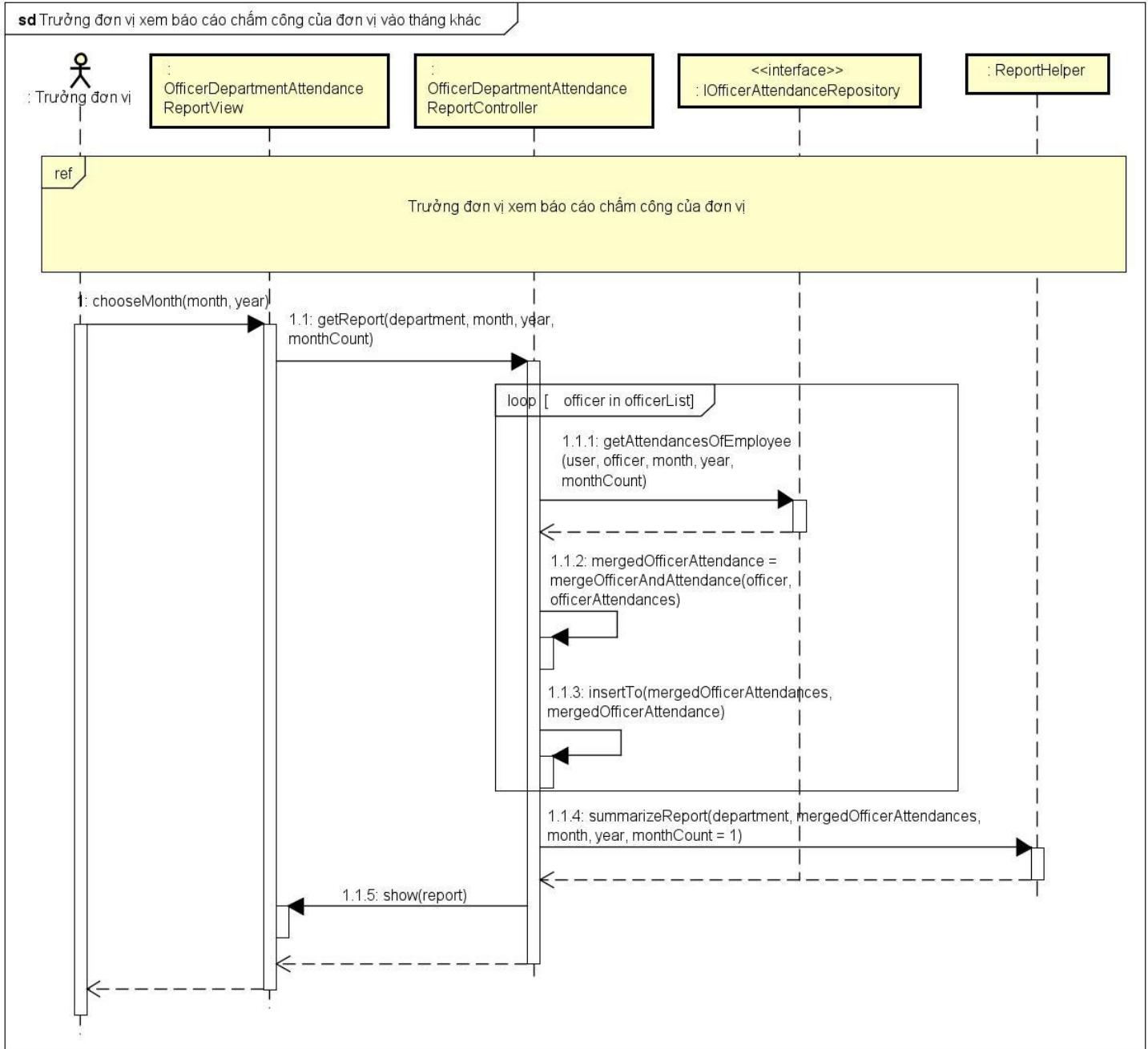
\* Main Scenario: Trưởng đơn vị xem báo cáo chấm công của đơn vị



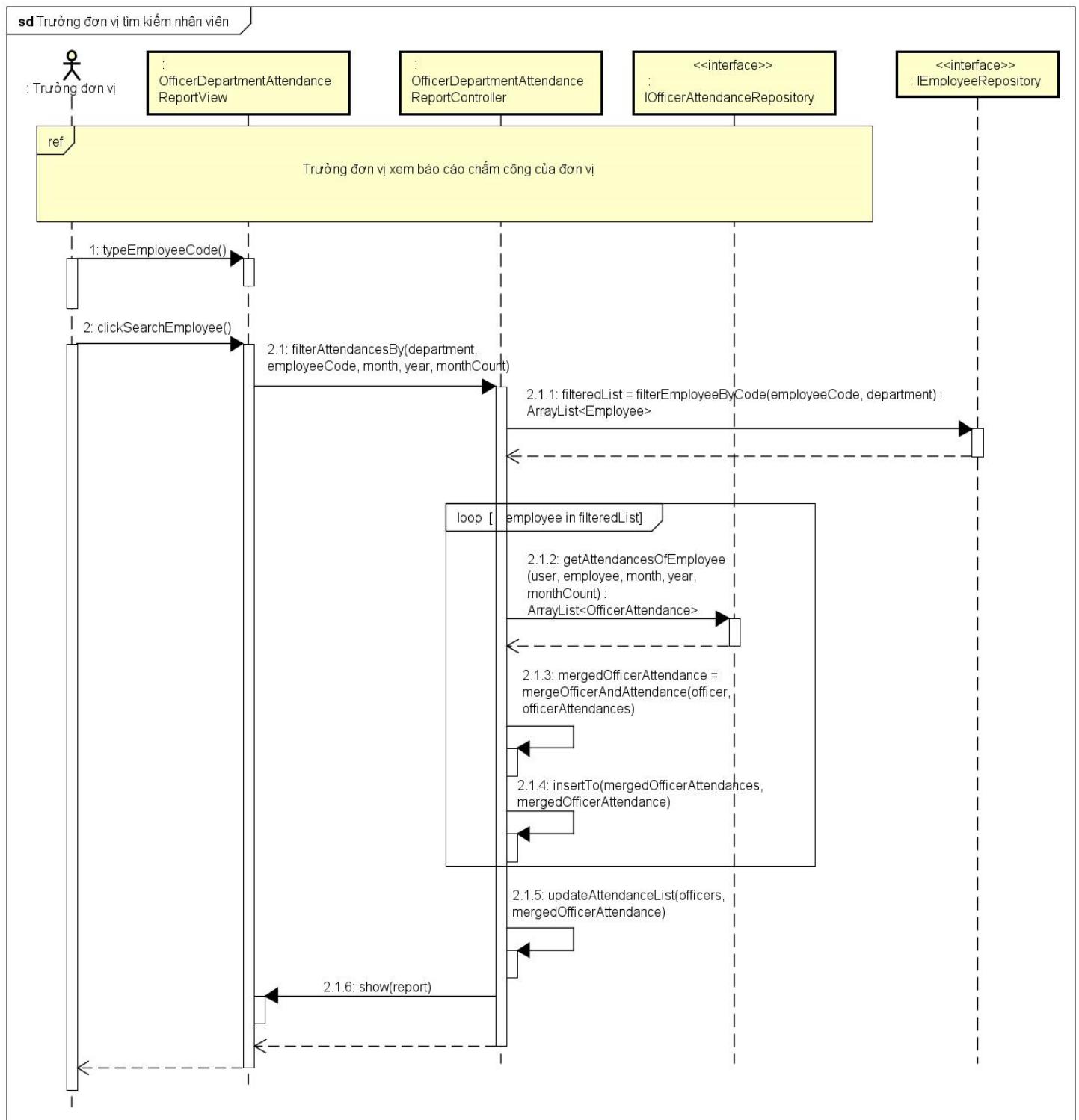
\* Scenario 2: Trưởng đơn vị xuất báo cáo chấm công của đơn vị thành file báo cáo



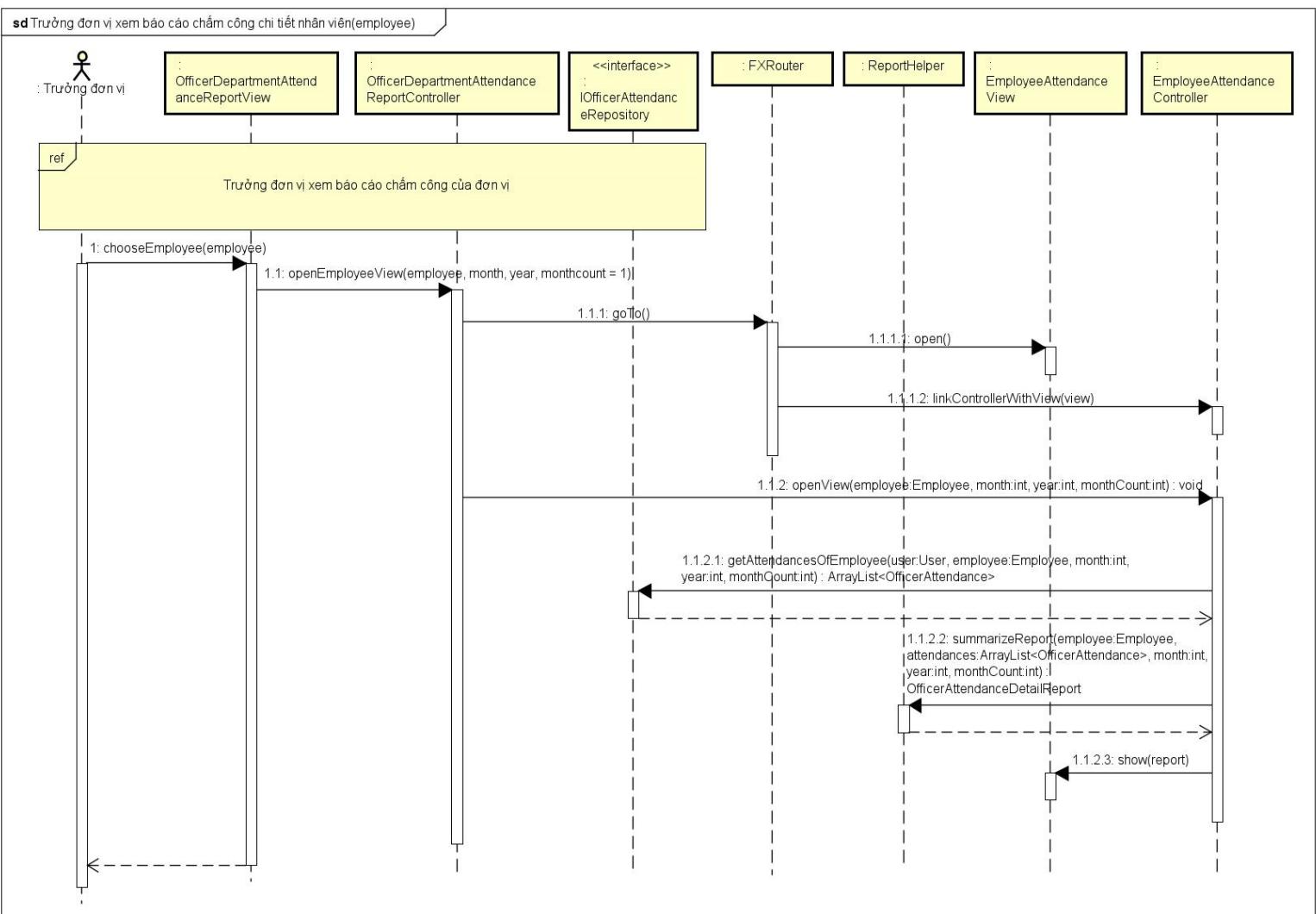
\* Scenario 3: Trưởng đơn vị xem báo cáo chấm công của đơn vị vào tháng khác (tương tự với quý, năm)



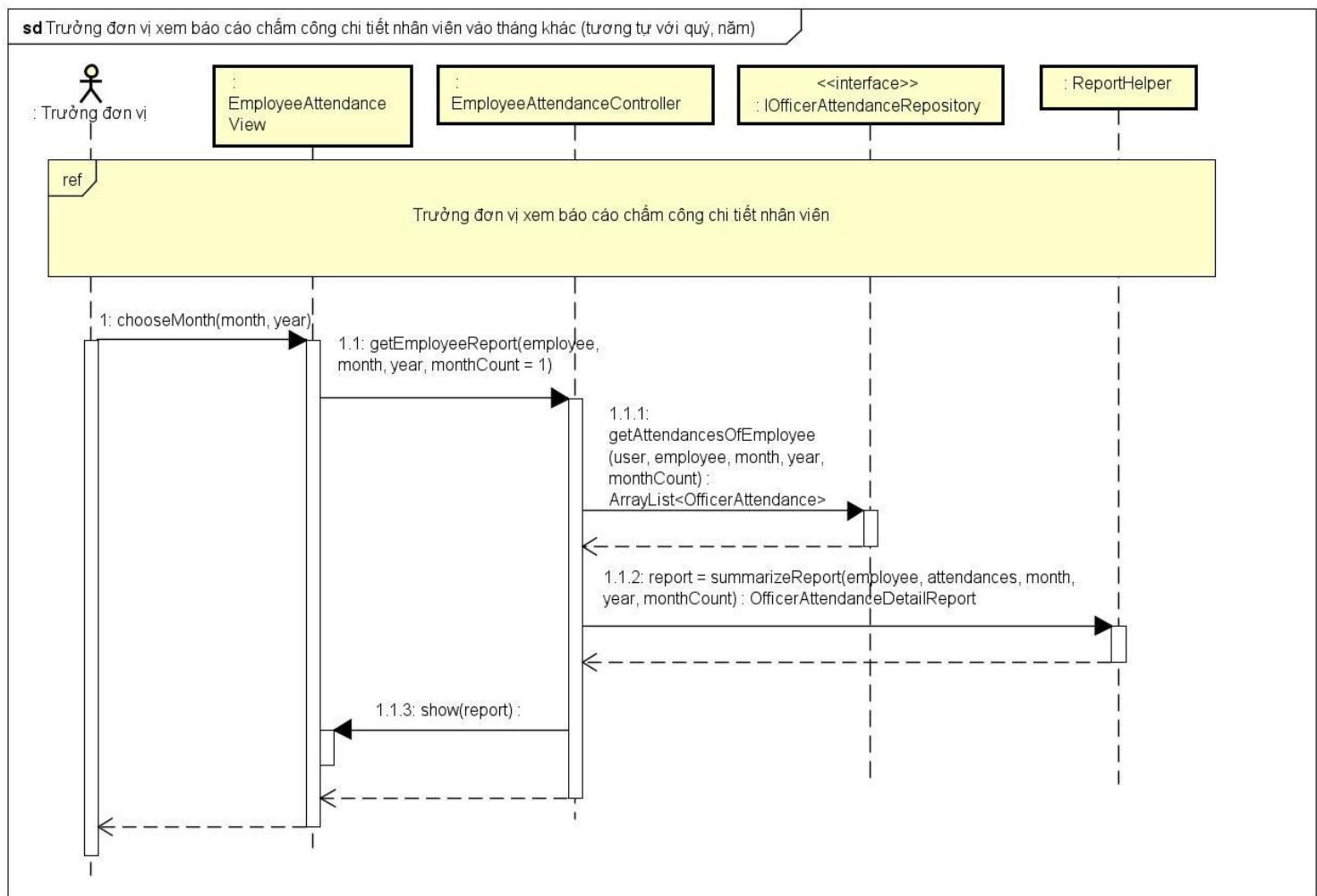
## \* Scenario 4: Trưởng đơn vị tìm kiếm nhân viên



\* Scenario 5: Trưởng đơn vị xem báo cáo chấm công chi tiết nhân viên



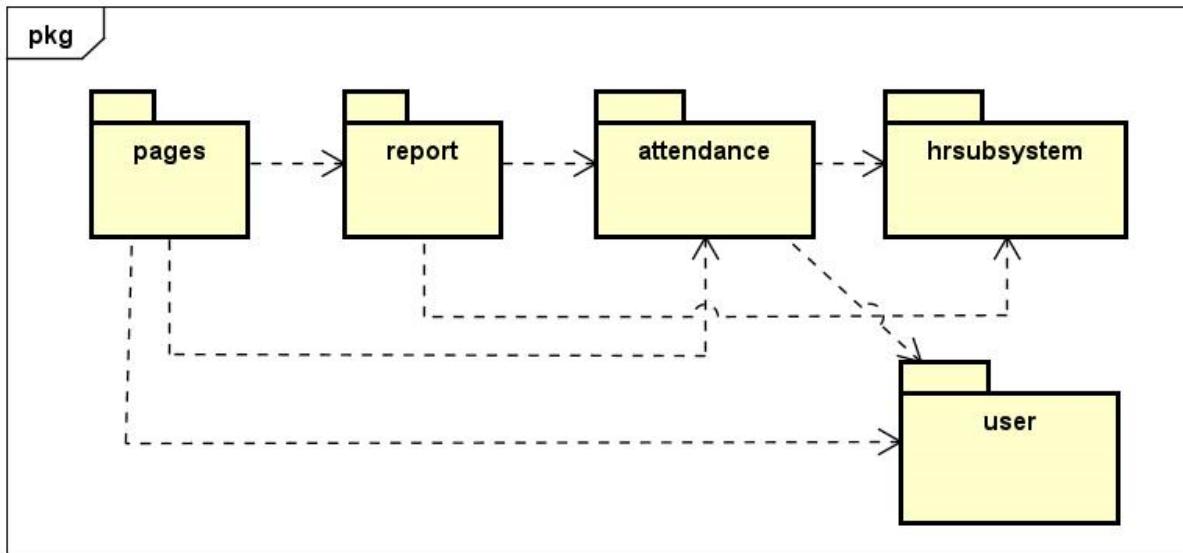
\* Scenario 6: Trưởng đơn vị xem báo cáo chấm công chi tiết nhân viên vào tháng khác (tương tự)

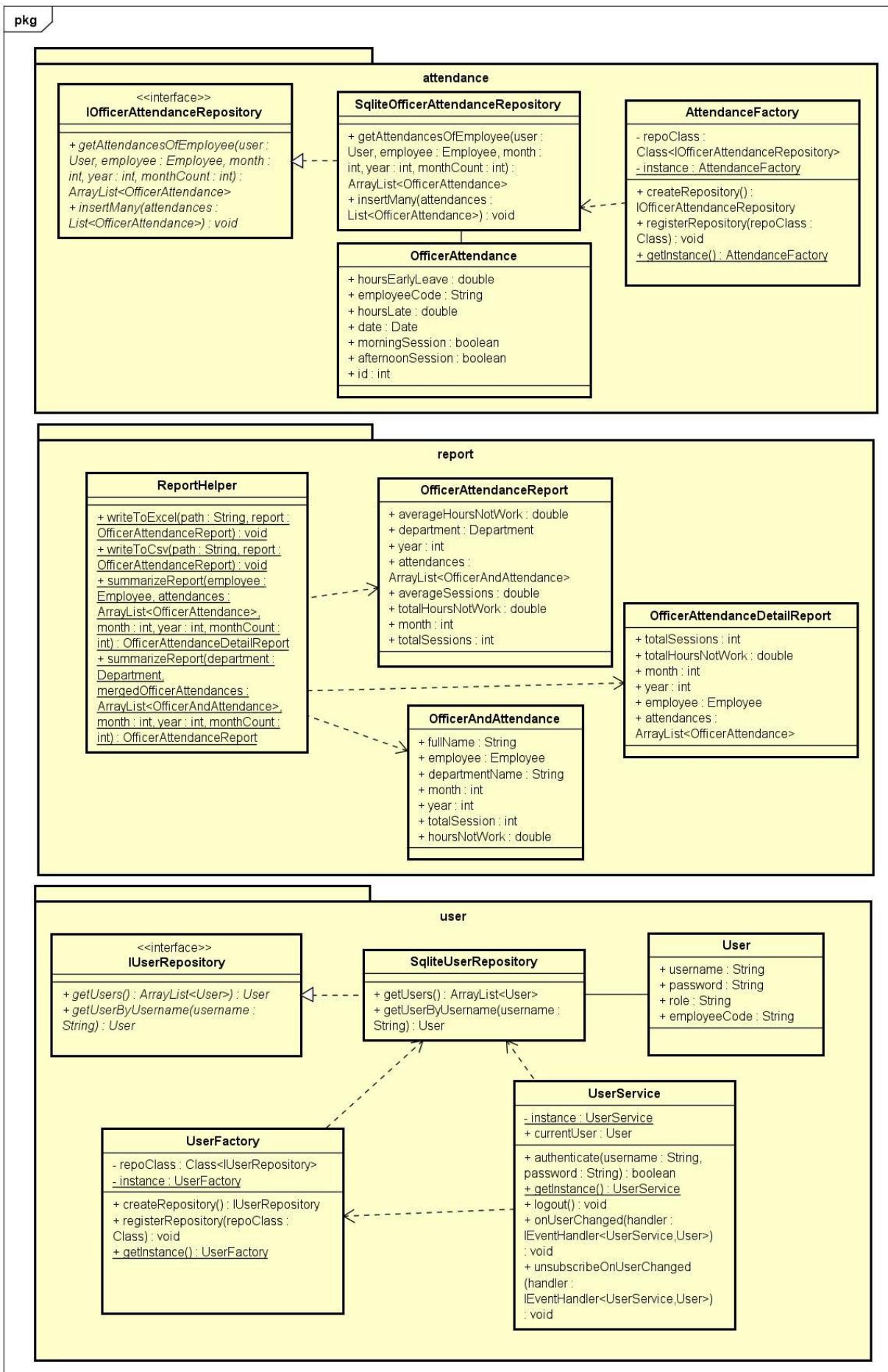


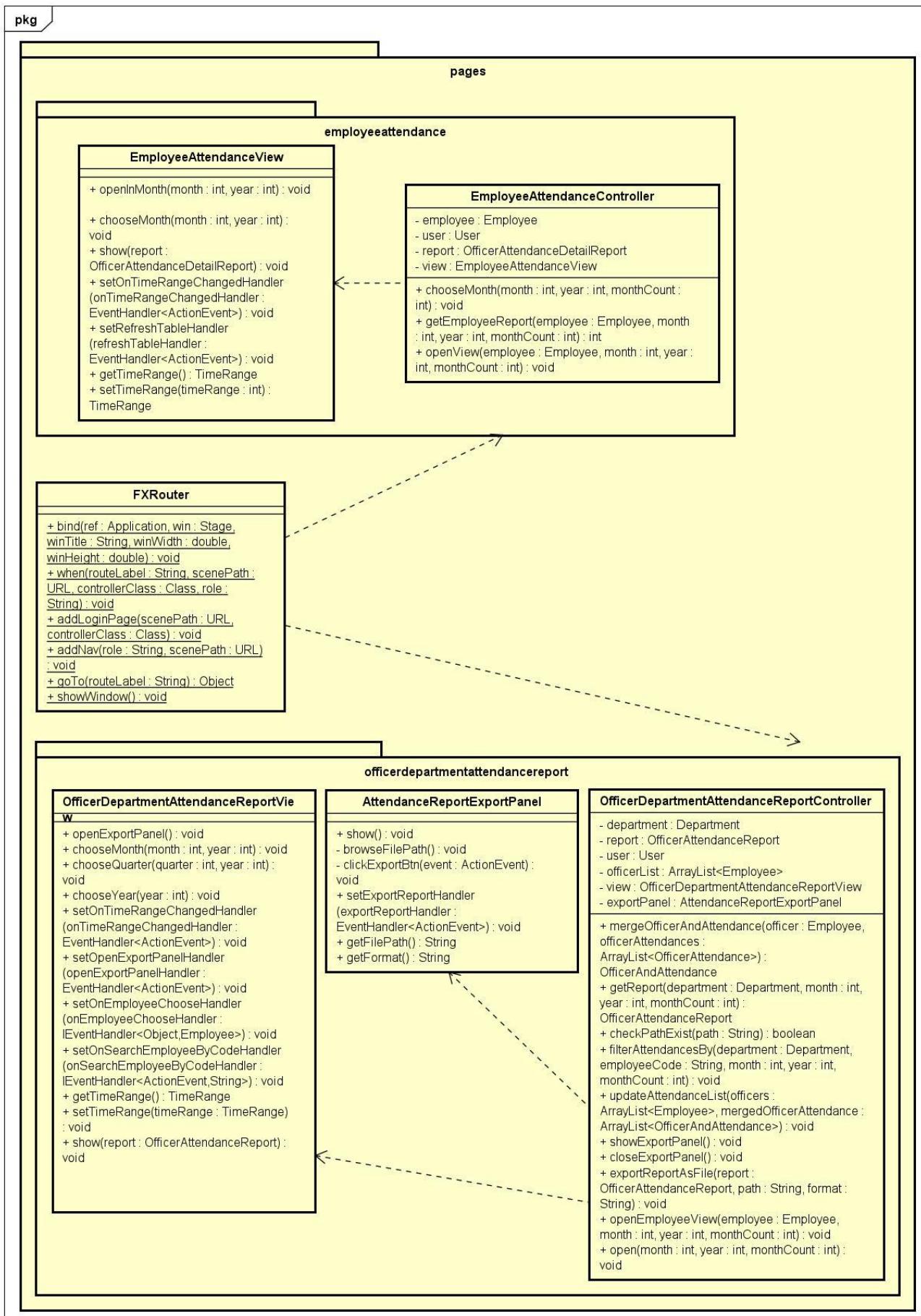
với quý, năm)

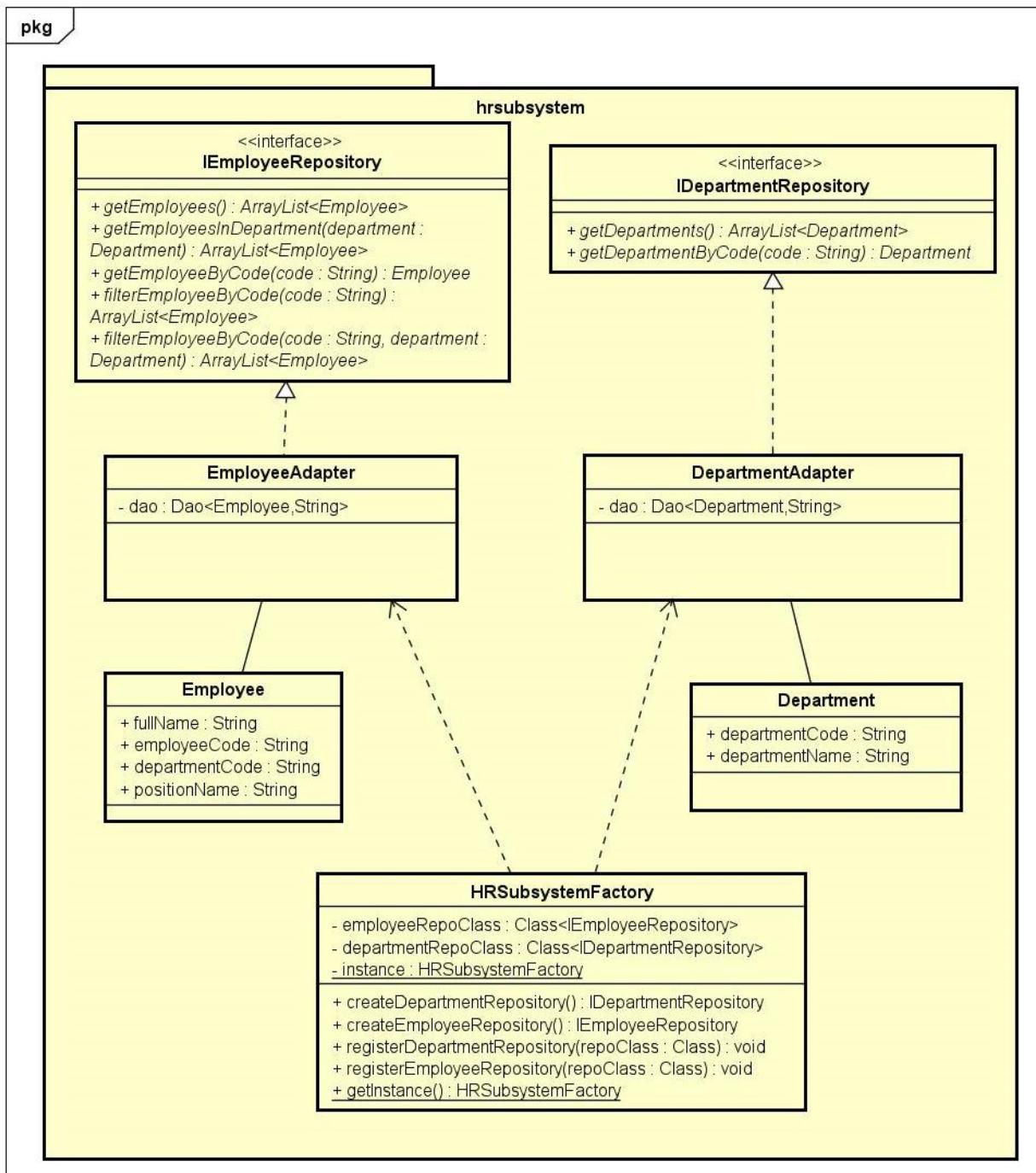
### 5.3 Biểu đồ lớp mức thiết kế

### 5.3.1 Use case “Xem báo cáo chấm công của đơn vị nhân viên văn phòng”









## CHƯƠNG 6. LẬP TRÌNH VÀ KIỂM THỬ ĐƠN VỊ (BÀI 7)

### 6.1 Thiết kế test case cho module: UserService

#### 6.1.1 Mô tả module

Module chứa các hàm xử lý hoạt động của User như đăng nhập, đăng xuất, lấy thông tin người dùng hiện tại. Lớp này phụ thuộc vào UserFactory và IUserRepository, 2 lớp trên cung cấp khả năng truy cập cơ sở dữ liệu, giúp Userservice nắm được thông tin người dùng.

#### 6.1.2 Thiết kế test case cho phương thức authenticate của UserService

Tên phương thức: authenticate

Kiểu trả về: boolean

Danh sách tham số: tên đăng nhập (username) và mật khẩu (password)

Chi tiết: Hàm nhận vào tên đăng nhập và mật khẩu, đồng thời xác nhận xem trong cơ sở dữ liệu có tài khoản nào có tên đăng nhập và mật khẩu khớp không, trả về true nếu có, trả về false nếu không.

#### 6.1.3 Kỹ thuật sử dụng kiểm thử hộp đen và phân vùng tương đương.

Kiểm thử hộp đen: Không cần biết logic bên trong code, đầu vào hợp lệ và đầu vào không hợp lệ.

Phân vùng tương đương cho phép phân chia tập hợp các điều kiện kiểm tra thành một phân vùng nên được coi là giống nhau. Để thiết kế được các test case cho phương thức này, đầu tiên xác định các vùng tương đương của đầu vào, ở đây là tên đăng nhập và mật khẩu.

Do lớp UserService phụ thuộc vào IUserRepository để biết được có những tài khoản tương ứng với mật khẩu nào đang có trong cơ sở dữ liệu, nên ta sẽ dựng một implement (FakeUserRepository) của IUserRepository chỉ chứa mỗi tài khoản có tên đăng nhập là “user” và mật khẩu là “1234” để phục vụ cho việc kiểm thử trở nên dễ dàng.

Với FakeUserRepository ở trên, ta được vùng tương đương cho kết quả là true:

- username=”user” và password=”1234”

Phân vùng tương đương cho kết quả là false:

- username khác ”user”
- password khác ”1234”

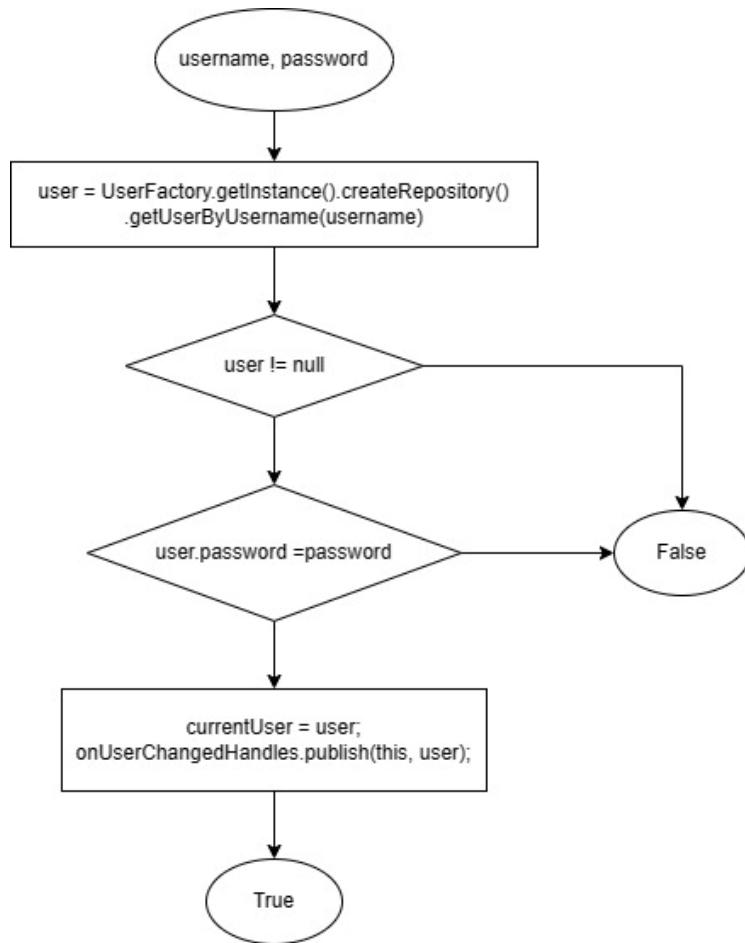
Tên class kiểm thử: UserServiceTest. Các phương thức kiểm thử trong class:

1. authenticate\_TrueResult: test trường hợp hàm authenticate trả về true  
Kiểm thử với đầu vào username=”user” và password=”1234”
2. authenticate\_FalseResult: test trường hợp hàm authenticate trả về false  
Kiểm thử với đầu vào username=” admin” và password=” 123456”  
Kiểm thử với đầu vào username=” user” và password=”123456”  
Kiểm thử với đầu vào username=” hello” và password=” 1234”

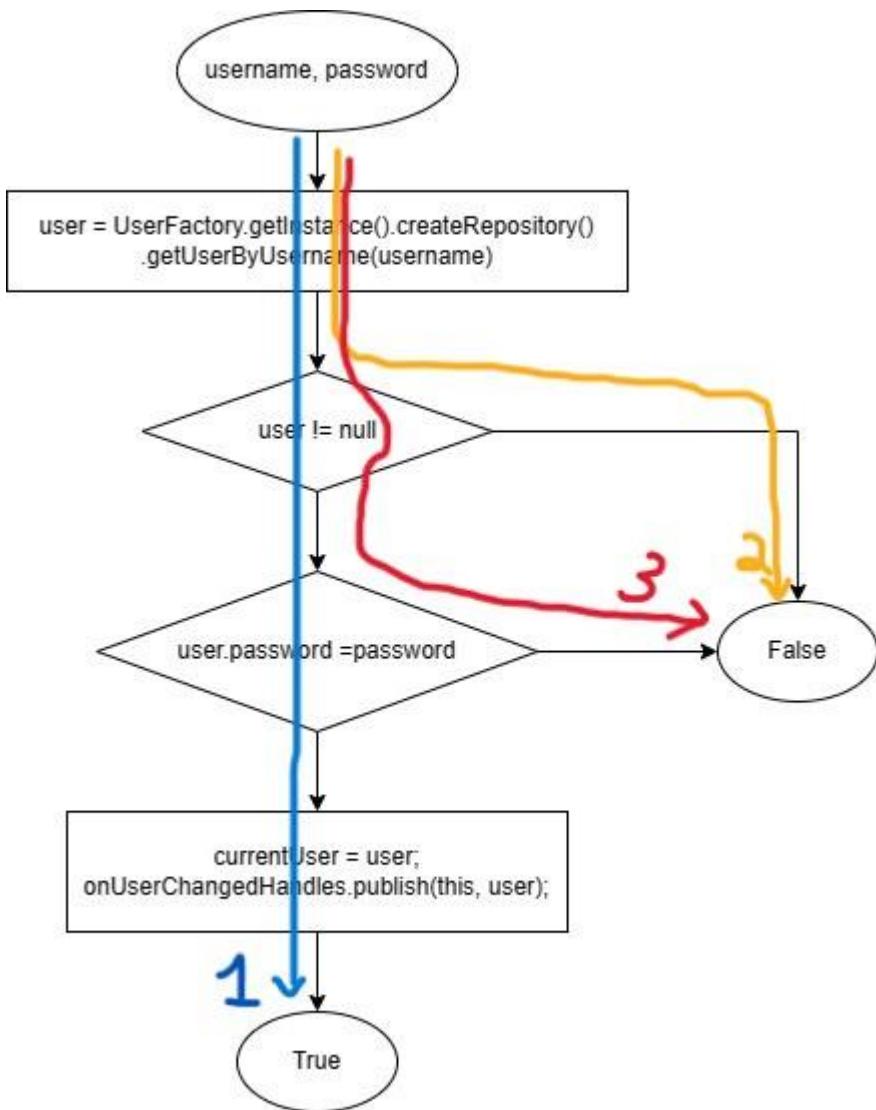
#### 6.1.4 Kỹ thuật sử dụng: kiểm thử hộp trắng và bảng quyết định.

Kiểm thử hộp trắng (độ đo c1): Tất cả các nhánh trong phương thức/module đều được thực thi.

Sử dụng Control Flow test:



Ta nhận thấy ứng dụng có 3 luồng như sau:



Ta xây dựng test case kiểm thử hộp trống:

- Luồng số 1: authenticate\_UserNotNull\_PasswordSame  
Kiểm thử với đầu vào username="user" và password="1234"  
Lúc này user != null cũng như user.password = password -> true
- Luồng số 2: authenticate\_UserNull  
Kiểm thử với đầu vào username="admin" và password="123456"  
Lúc này user = null -> false
- Luồng số 3: authenticate\_UserNotNull\_PasswordNotSame  
Kiểm thử với đầu vào username="user" và password="123456"  
Lúc này user != null cũng như user.password != password -> false

## 6.2 Thiết kế kiểm thử use case

### 6.2.1 Test case Xem chấm báo cáo công đơn vị

<b>Test case #:</b> 1.1	<b>Tên testcase:</b> Xem chấm báo cáo công đơn vị
<b>Hệ thống:</b> Hệ thống quản lý chấm công	<b>Hệ thống con:</b> Hệ thống quản lý nhân sự
<b>Thiết kế bởi:</b> Trần Phúc Mạnh Linh	<b>Ngày thiết kế:</b> 20/12/2023
<b>Thực thi bởi:</b> Hệ thống quản lý chấm công	<b>Ngày thực thi:</b> 24/12/2023
<b>Mô tả ngắn gọn:</b> Kiểm tra dữ liệu hiện lên trên trang xem báo cáo chấm công của đơn vị có chính xác, quá trình hiển thị có suôn sẻ	

#### Tiền điều kiện:

Đã đăng nhập vào hệ thống dưới vai trò trưởng phòng

Bước	Hành động	Phản hồi mong muốn từ hệ thống	Pass/Fail	Comment
1	Người dùng click nút Xem báo cáo chấm công	Hệ thống hiển thị báo cáo chấm công của đơn vị	Pass	

#### Trạng thái của hệ thống sau khi chạy hàm:

Hệ thống hiển thị thông tin tổng hợp gồm tổng số buổi làm, tổng số giờ đi muộn về sớm, số giờ đi muộn về sớm trung bình, số buổi đi làm trung bình cũng như danh sách tổng số buổi làm, tổng số giờ đi muộn về sớm của từng nhân viên trong khoảng thời gian xác định.

### 6.2.2 Test case Xem chấm báo cáo công đơn vị vào tháng khác

<b>Test case #:</b> 1.2	<b>Tên testcase:</b> Xem chấm báo cáo công đơn vị vào tháng khác
<b>Hệ thống:</b> Hệ thống quản lý chấm công	
<b>Thiết kế bởi:</b> Trần Phúc Mạnh Linh	<b>Ngày thiết kế:</b> 20/12/2023
<b>Thực thi bởi:</b> Hệ thống quản lý chấm công	<b>Ngày thực thi:</b> 24/12/2023
<b>Mô tả ngắn gọn:</b> Kiểm tra dữ liệu hiện lên trên trang xem báo cáo chấm công sau khi chọn tháng khác của đơn vị có chính xác,	

quá trình hiển thị có suôn sẻ	
-------------------------------	--

### Tiền điều kiện:

Đã đăng nhập vào hệ thống dưới vai trò trưởng phòng

Đã vào được trang Xem báo cáo chấm công

Bước	Hành động	Phản hồi mong muốn từ hệ thống	Pass/Fail	Comment
1	Người dùng click nút Chọn khoảng thời gian (Tháng/Quý/Năm)	Hệ thống hiển thị báo cáo trên khoảng thời gian được chọn	Pass	
2	Người dùng click nút chọn tháng	Hệ thống hiển thị báo cáo trên khoảng thời gian được chọn	Pass	
3	Người dùng click nút chọn quý	Hệ thống hiển thị báo cáo trên khoảng thời gian được chọn	Pass	
4	Người dùng click nút chọn năm	Hệ thống hiển thị báo cáo trên khoảng thời gian được chọn	Pass	

### Trạng thái của hệ thống sau khi chạy hàm:

Hệ thống hiển thị thông tin tổng hợp gồm tổng số buổi làm, tổng số giờ đi muộn về sớm, số giờ đi muộn về sớm trung bình, số buổi đi làm trung bình cũng như danh sách tổng số buổi làm, tổng số giờ đi muộn về sớm của từng nhân viên trong khoảng thời gian được chọn.

#### 6.2.3 Test case Xuất báo cáo chấm công thành file báo cáo thành công

<b>Test case #:</b> 1.3	<b>Tên testcase:</b> Xuất báo cáo chấm công thành file báo cáo
<b>Hệ thống:</b> Hệ thống quản lý chấm công	
<b>Thiết kế bởi:</b> Trần Phúc Mạnh Linh	<b>Ngày thiết kế:</b> 20/12/2023
<b>Thực thi bởi:</b> Hệ thống quản lý chấm công	<b>Ngày thực thi:</b> 24/12/2023
<b>Mô tả ngắn gọn:</b> Kiểm tra xem hệ thống xuất báo cáo thành công không	

**Tiền điều kiện:**

Đã đăng nhập vào hệ thống dưới vai trò trưởng phòng

Đã vào được trang Xem báo cáo chấm công

Bước	Hành động	Phản hồi mong muốn từ hệ thống	Pass/Fail	Comment
1	Người dùng click nút Xuất báo cáo	Hệ thống hiển thị giao diện xuất báo cáo	Pass	
2	Người dùng chọn định dạng	Hệ thống hiển thị giá trị định dạng mong muốn	Pass	
3	Người dùng nhập dữ liệu vào Vị trí lưu	Hệ thống hiển thị những gì người dùng nhập	Pass	
4	Người dùng nhấn Chọn vị trí	Hệ thống mở hộp thoại chọn vị trí lưu	Pass	
5	Người dùng nhấn nút Xuất	Báo cáo được xuất ra vị trí mong muốn và giao diện xuất báo cáo được đóng lại	Fail	

**Trạng thái của hệ thống sau khi chạy hàm:**

Hệ thống xuất báo cáo đến vị trí mong muốn.

**6.2.4 Test case Xuất báo cáo chấm công thành file báo cáo với vị trí sai**

<b>Test case #:</b> 1.3	<b>Tên testcase:</b> Xuất báo cáo chấm công thành file báo cáo
<b>Hệ thống:</b> Hệ thống quản lý chấm công	
<b>Thiết kế bởi:</b> Trần Phúc Mạnh Linh	<b>Ngày thiết kế:</b> 20/12/2023
<b>Thực thi bởi:</b> Hệ thống quản lý chấm công	<b>Ngày thực thi:</b> 24/12/2023
<b>Mô tả ngắn gọn:</b> Kiểm tra xem hệ thống có cảnh báo người dùng nhập sai vị trí không	

**Tiền điều kiện:**

Đã đăng nhập vào hệ thống dưới vai trò trưởng phòng

Đã vào được trang Xem báo cáo chấm công

Bước	Hành động	Phản hồi mong muốn từ hệ thống	Pass/Fail	Comment
1	Người dùng click nút Xuất báo cáo	Hệ thống hiển thị giao diện xuất báo cáo	Pass	
2	Người dùng chọn định dạng	Hệ thống hiển thị giá trị định dạng mong muốn	Pass	
3	Người dùng nhập dữ liệu sai vào Vị trí lưu	Hệ thống hiển thị những gì người dùng nhập	Pass	
4	Người dùng nhấn nút Xuất	Thông báo người dùng nhập sai vị trí	Pass	

**Trạng thái của hệ thống sau khi chạy hàm:**

Hệ thống xuất báo cáo đến vị trí mong muốn.

**6.2.5 Test case Tìm kiếm nhân viên bằng mã nhân viên**

<b>Test case #:</b> 1.4	<b>Tên testcase:</b> Tìm kiếm nhân viên bằng mã nhân viên
<b>Hệ thống:</b> Hệ thống quản lý chấm công	<b>Hệ thống con:</b> Hệ thống quản lý nhân sự
<b>Thiết kế bởi:</b> Trần Phúc Mạnh Linh	<b>Ngày thiết kế:</b> 20/12/2023
<b>Thực thi bởi:</b> Hệ thống quản lý chấm công	<b>Ngày thực thi:</b> 24/12/2023
<b>Mô tả ngắn gọn:</b> Kiểm tra khả năng lọc và hiển thị mã nhân viên	

**Tiền điều kiện:**

Đã đăng nhập vào hệ thống dưới vai trò trưởng phòng

Đã vào được trang Xem báo cáo chấm công

Bước	Hành động	Phản hồi mong muốn từ	Pass/Fail	Comment

		hệ thống		
1	Người dùng nhập dữ liệu vào Textfield Mã nhân viên	Hệ thống hiển thị những gì người dùng nhập	Pass	
2	Người dùng click nút Tìm kiếm	Hệ thống hiển thị danh sách nhân viên mong muốn.	Pass	
3	Người dùng chọn nút Hủy tìm kiếm	Hệ thống hiển thị toàn bộ nhân viên trong đơn vị.	Pass	

#### Trạng thái của hệ thống sau khi chạy hàm:

Hệ thống hiển thị danh sách nhân viên mong muốn tìm kiếm.

#### 6.2.6 Test case Xem báo cáo công chi tiết nhân viên

<b>Test case #:</b> 1.5	<b>Tên testcase:</b> Xem báo cáo chấm công chi tiết nhân viên
<b>Hệ thống:</b> Hệ thống quản lý chấm công	
<b>Thiết kế bởi:</b> Trần Phúc Mạnh Linh	<b>Ngày thiết kế:</b> 20/12/2023
<b>Thực thi bởi:</b> Hệ thống quản lý chấm công	<b>Ngày thực thi:</b> 24/12/2023
<b>Mô tả ngắn gọn:</b> Kiểm tra việc chuyển trang sang Xem chi tiết nhân viên có thành công không, trang có hiển thị dữ liệu không	

#### Tiền điều kiện:

Đã đăng nhập vào hệ thống dưới vai trò trưởng phòng

Đã vào được trang Xem báo cáo chấm công

Bước	Hành động	Phản hồi mong muốn từ hệ thống	Pass/Fail	Comment
1	Người dùng click nút Xem chi tiết	Hệ thống hiển thị giao diện Xem chi tiết nhân viên, trang hiển thị dữ liệu chính xác	Pass	

#### Trạng thái của hệ thống sau khi chạy hàm:

Hệ thống hiển thị giao diện Xem chi tiết nhân viên với thông tin tổng hợp gồm tổng số buổi làm, tổng số giờ đi muộn về sớm của nhân viên cũng như danh sách chấm công theo ngày.

#### 6.2.7 Test case Xem báo cáo chấm công chi tiết nhân viên vào tháng khác

<b>Test case #:</b> 1.6	<b>Tên testcase:</b> Xem chấm báo cáo công chi tiết nhân viên vào tháng khác
<b>Hệ thống:</b> Hệ thống quản lý chấm công	
<b>Thiết kế bởi:</b> Trần Phúc Mạnh Linh	<b>Ngày thiết kế:</b> 20/12/2023
<b>Thực thi bởi:</b> Hệ thống quản lý chấm công	<b>Ngày thực thi:</b> 24/12/2023
<b>Mô tả ngắn gọn:</b> Kiểm tra dữ liệu hiện lên trên trang xem chi tiết nhân viên có chính xác, quá trình hiển thị có suôn sẻ	

#### Tiền điều kiện:

Đã đăng nhập vào hệ thống dưới vai trò trưởng phòng

Đã vào được trang Xem chi tiết nhân viên

Bước	Hành động	Phản hồi mong muốn từ hệ thống	Pass/Fail	Comment
1	Người dùng click nút Chọn khoảng thời gian (Tháng/Quý/Năm)	Hệ thống hiển thị báo cáo trên khoảng thời gian được chọn	Pass	
2	Người dùng click nút chọn tháng	Hệ thống hiển thị báo cáo trên khoảng thời gian được chọn	Pass	
3	Người dùng click nút chọn quý	Hệ thống hiển thị báo cáo trên khoảng thời gian được chọn	Pass	
4	Người dùng click nút chọn năm	Hệ thống hiển thị báo cáo trên khoảng thời gian được chọn	Pass	

#### Trạng thái của hệ thống sau khi chạy hàm:

Hệ thống hiển thị thông tin tổng hợp gồm tổng số buổi làm, tổng số giờ đi muộn về sớm của nhân viên cũng như danh sách chấm công theo ngày.