**BÁO CÁO ASSIGNMENT FINAL**

**NHẬP MÔN KỸ THUẬT PHẦN MỀM**

**Giảng viên hướng dẫn: Thầy Lại Minh Đăng**

# **Y1. Phân tích yêu cầu phần mềm - Hệ thống quản lý giảng viên**

### **1. Mô tả bài toán**

Trong các trường đại học, việc quản lý giảng viên là nhiệm vụ quan trọng của ban giám hiệu và phòng đào tạo. Tuy nhiên, hiện nay nhiều trường vẫn quản lý theo cách thủ công qua Excel, email, hoặc hồ sơ giấy. Điều này gây ra:

* Khó khăn trong việc lưu trữ, cập nhật hồ sơ giảng viên (thông tin cá nhân, học vị, khoa, môn phụ trách).
* Phân công lịch giảng dạy thủ công, dễ trùng ca, mất cân đối về khối lượng giảng dạy giữa các giảng viên.
* Khó theo dõi tiến độ giảng dạy, hoạt động tham gia sự kiện, nghiên cứu, hoặc đóng góp cho trường.
* Công tác đánh giá giảng viên chủ yếu dựa vào báo cáo nội bộ, thiếu minh bạch, tốn nhiều thời gian tổng hợp.
* Phản hồi từ sinh viên thường thu qua khảo sát giấy hoặc form online riêng lẻ, dữ liệu phân tán, không đồng bộ.

**Hệ thống quản lý giảng viên** được xây dựng nhằm giải quyết các vấn đề trên, với mục tiêu:

* Lưu trữ tập trung hồ sơ giảng viên, đảm bảo thông tin cập nhật và chính xác.
* Hỗ trợ nhà trường trong việc phân công, quản lý và theo dõi lịch giảng dạy.
* Quản lý hoạt động chuyên môn và đóng góp khác của giảng viên (giảng dạy, tham gia sự kiện, nghiên cứu khoa học…).
* Thu thập và xử lý phản hồi sinh viên định kỳ (mỗi môn 1 lần), phục vụ công tác đánh giá chất lượng giảng dạy.
* Cung cấp báo cáo, thống kê cho ban giám hiệu và phòng đào tạo để đưa ra quyết định quản lý.

### 

### **2. Các bên liên quan**

1. **Ban giám hiệu**
   * Quan tâm đến chất lượng giảng dạy và sự phát triển đội ngũ giảng viên.
   * Cần báo cáo, thống kê để làm cơ sở ra quyết định, xét thưởng hoặc khen ngợi.
2. **Phòng đào tạo / Khoa**
   * Nhập và quản lý thông tin giảng viên.
   * Phân công, theo dõi lịch dạy và hoạt động chuyên môn.
   * Thu thập, tổng hợp phản hồi sinh viên để đánh giá khách quan hơn.
3. **Giảng viên**
   * Truy cập hệ thống để xem lịch dạy, sự kiện được phân công.
   * Cập nhật một số thông tin cá nhân (liên hệ, địa chỉ, email).
   * Theo dõi hoạt động giảng dạy, xem kết quả đánh giá tổng hợp từ sinh viên.
4. **Quản trị hệ thống (IT Admin)**
   * Quản lý tài khoản, phân quyền sử dụng.
   * Đảm bảo an toàn dữ liệu, sao lưu, và vận hành hệ thống ổn định.

### **3. Yêu cầu chức năng (Functional Requirements)**

Hệ thống cần cung cấp các chức năng chính sau:

### **Quản lý giảng viên**

* Thêm mới, cập nhật, xóa, tìm kiếm hồ sơ giảng viên.
* Quản lý các thông tin cơ bản (họ tên, mã giảng viên, liên hệ, trình độ, đơn vị công tác).

### **Quản lý môn học & lớp học phần**

* Quản lý danh sách môn học (thêm, sửa, xóa).
* Quản lý lớp học phần (tạo, cập nhật thông tin lớp, sĩ số, học kỳ).
* Phân công giảng dạy (gán giảng viên vào môn học/lớp học phần).

### **Quản lý thời khóa biểu & giờ giảng**

* Tự động tổng hợp và hiển thị thời khóa biểu theo tuần/học kỳ.
* Quản lý giờ giảng (ghi nhận số tiết, thống kê giờ giảng theo khoa/giảng viên).

### **Quản lý tài khoản & phân quyền**

* Đăng nhập/Đăng xuất an toàn (xác thực người dùng).
* Quản lý tài khoản giảng viên, phòng đào tạo, Ban giám hiệu, IT Admin.
* Phân quyền theo vai trò: **Ban giám hiệu**, **Phòng đào tạo/Khoa**, **Giảng viên**, **IT Admin**.
* Sao lưu và phục hồi dữ liệu (dành cho IT Admin).

### **Báo cáo – Thống kê**

* + Thống kê giờ giảng theo khoa, học kỳ hoặc cá nhân.
  + Thống kê số lượng giảng viên theo học vị/học hàm.
  + Xuất báo cáo tổng hợp ra **Excel/PDF** phục vụ Ban giám hiệu ra quyết định.

**4. Yêu cầu phi chức năng (Non-Functional Requirements)**

Hệ thống cần đảm bảo các đặc tính chất lượng sau:

### **Hiệu năng**

* Thời gian phản hồi khi tìm kiếm giảng viên hoặc hiển thị thời khóa biểu **< 3 giây**.
* Xử lý tối thiểu 500 yêu cầu đồng thời mà không giảm tốc độ đáng kể.

### **Bảo mật**

* Phân quyền rõ ràng: mỗi nhóm người dùng chỉ được truy cập chức năng thuộc quyền hạn.
* Mã hóa mật khẩu, bảo vệ dữ liệu nhạy cảm (thông tin cá nhân giảng viên).
* Ghi nhật ký truy cập (log) cho các hoạt động quan trọng (thêm, xóa, phân công).

### **Tính khả dụng**

* Hệ thống hoạt động 24/7 với thời gian gián đoạn (downtime) < **2%/năm**.
* Hỗ trợ truy cập trên mạng nội bộ và Internet.

### **Tính mở rộng**

* Cho phép bổ sung thêm giảng viên, khoa hoặc lớp học phần khi quy mô trường tăng.
* Hỗ trợ tích hợp với các hệ thống khác (ví dụ: hệ thống quản lý sinh viên) trong tương lai.

### **Tính dễ sử dụng**

* Giao diện trực quan, hỗ trợ tiếng Việt đầy đủ.
* Hoạt động tốt trên **máy tính, laptop** và tối ưu cho **tablet** để giảng viên dễ truy cập.

### **Tuân thủ**

* Đáp ứng các quy định bảo vệ dữ liệu cá nhân theo luật pháp Việt Nam.

## **5. Kỹ thuật 1: Phỏng vấn (Interview)**

**Mục tiêu**

* Hiểu bối cảnh, quy tắc nghiệp vụ, luồng công việc, các khó khăn và ngoại lệ.
* Nắm yêu cầu về báo cáo, hiệu năng, bảo mật, phân quyền để phục vụ thiết kế hệ thống.

**Đối tượng phỏng vấn (đóng vai)**

* **Ban Giám Hiệu**: cần báo cáo, thống kê, thông tin phục vụ ra quyết định.
* **Phòng Đào Tạo/Khoa**: trực tiếp quản lý hồ sơ giảng viên, phân công giảng dạy, sắp xếp thời khóa biểu.
* **Giảng Viên**: sử dụng hệ thống để xem lịch dạy, cập nhật thông tin cá nhân.
* **IT Admin**: quản lý tài khoản, phân quyền, sao lưu và bảo mật hệ thống.

**Câu hỏi mẫu**

1. Anh/chị hiện đang quản lý thông tin giảng viên và lịch giảng dạy bằng cách nào?
2. Quy trình phân công lịch dạy gồm những bước nào? Khâu nào thường gặp lỗi hoặc trùng ca?
3. Khi có thay đổi lịch đột xuất, hiện tại xử lý ra sao?
4. Thông tin nào là bắt buộc trong hồ sơ giảng viên?
5. Ban Giám Hiệu cần những loại báo cáo, thống kê nào? Tần suất xuất báo cáo ra sao?
6. Với IT Admin: cơ chế sao lưu và phục hồi dữ liệu hiện nay có gặp khó khăn gì?
7. Quy trình cấp/thu hồi tài khoản người dùng hiện được thực hiện như thế nào?
8. Tiêu chí hiệu năng hoặc chất lượng nào (ví dụ tốc độ tra cứu <3 giây) được mong đợi?
9. Có quy định pháp lý hoặc chuẩn bảo mật nào bắt buộc hệ thống phải tuân thủ?
10. Nếu chỉ chọn 3 việc hệ thống cần làm thật tốt, đó sẽ là gì?

**Đầu ra**

* Biên bản phỏng vấn từng nhóm actor
* User Stories sơ bộ
* Acceptance Criteria nháp cho từng chức năng chính

## **6. Kỹ thuật 2: Khảo sát (Survey/Questionnaire)**

**Mục tiêu**

* Lượng hóa tần suất sử dụng, mức độ khó khăn, ưu tiên tính năng, mức kỳ vọng về tốc độ và bảo mật.
* Xác định các nhu cầu chung, phục vụ xếp hạng ưu tiên phát triển.

**Thiết kế khảo sát**

* Số câu: 8–12 câu, thời gian trả lời <5 phút.
* Hình thức: Google Form/Microsoft Form.
* Dạng câu hỏi: Likert 1–5, chọn lựa nhiều phương án, 1–2 câu mở.
* Mỗi câu đo 1 khái niệm rõ ràng.

**Câu hỏi mẫu**

1. Vai trò của bạn? (Ban Giám Hiệu / Phòng Đào Tạo / Giảng Viên / IT Admin)
2. Bạn tra cứu hồ sơ giảng viên bao nhiêu lần mỗi tuần? (0 / 1–2 / 3–5 / >5)
3. Mức khó khi kiểm tra trùng lịch giảng dạy? (1 = rất dễ, 5 = rất khó)
4. Tính năng bạn ưu tiên nhất (chọn tối đa 3): Hồ sơ giảng viên / Trùng lịch / Báo cáo / Quản lý tài khoản / Phân quyền
5. Mức quan trọng của việc tìm kiếm và hiển thị kết quả trong <3 giây? (1–5)
6. Độ quan trọng của việc sao lưu dữ liệu an toàn? (1–5)
7. Mức mong đợi về thời gian phục hồi hệ thống khi gặp sự cố (ví dụ <2 giờ)? (1–5)
8. Nếu chỉ cải tiến 1 việc trong quy trình quản lý hiện tại, đó là gì và vì sao? (câu mở)

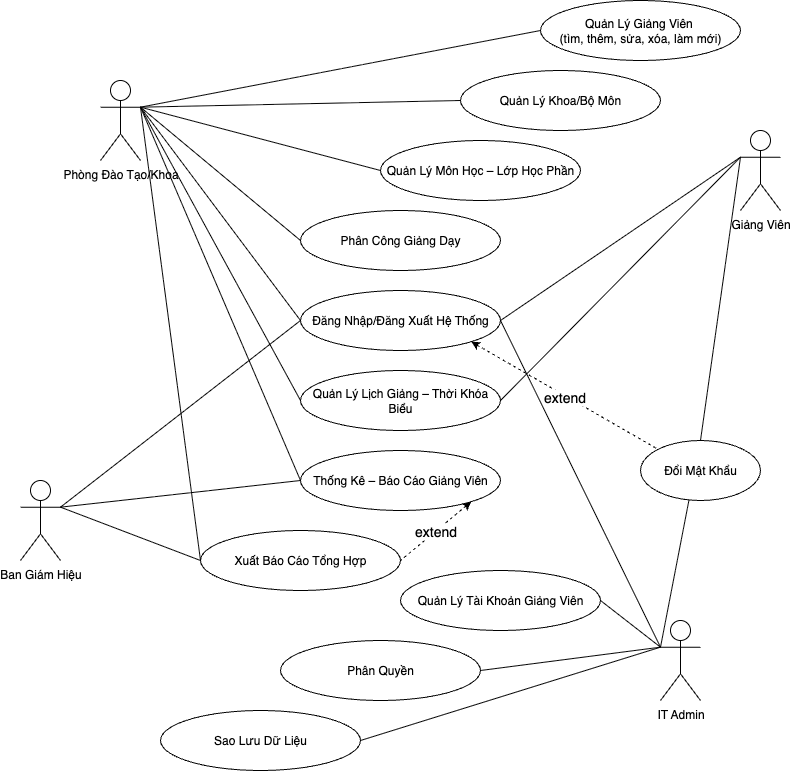
**Phân tích dữ liệu**

* Tính trung bình điểm Likert cho từng yếu tố.
* Xếp hạng ưu tiên các tính năng theo tần suất + mức ảnh hưởng cao.
* Xác định **Top 3 vấn đề cốt lõi** để đưa vào giai đoạn phát triển đầu tiên.

**Đầu ra**

* Link khảo sát đã triển khai
* Ảnh biểu đồ kết quả (Likert trung bình, biểu đồ cột/tỷ lệ)
* Bảng xếp hạng ưu tiên các tính năng

# **Y2. Đặc tả & mô hình hóa Use Case**

****

### **Bộ Đặc Tả Use Case Chi Tiết (UC1 - UC12)**

#### **UC1: Quản lý giảng viên**

* **Actor chính:** Phòng Đào Tạo/Khoa
* **Mục tiêu:** Lưu trữ, chỉnh sửa và cập nhật hồ sơ giảng viên một cách tập trung và chính xác.
* **Tiền điều kiện:** Người dùng đã đăng nhập với quyền của Phòng Đào Tạo/Khoa.
* **Hậu điều kiện thành công:** Thông tin giảng viên được cập nhật thành công trong cơ sở dữ liệu.

**Luồng sự kiện chính (Thêm mới):**

1. Người dùng chọn chức năng "Quản lý giảng viên" từ menu.
2. Hệ thống hiển thị danh sách giảng viên hiện có cùng các tùy chọn: "Thêm mới", "Sửa", "Xóa", "Tìm kiếm".
3. Người dùng chọn "Thêm mới".
4. Hệ thống hiển thị form nhập thông tin giảng viên.
5. Người dùng nhập đầy đủ và hợp lệ các thông tin (Mã giảng viên, họ tên, khoa, email...).
6. Người dùng nhấn "Lưu".
7. Hệ thống kiểm tra dữ liệu và xác nhận Mã giảng viên chưa tồn tại.
8. Hệ thống lưu thông tin giảng viên mới vào cơ sở dữ liệu.
9. Hệ thống hiển thị thông báo "Thêm mới giảng viên thành công!" và cập nhật lại danh sách.

**Luồng rẽ nhánh & Ngoại lệ:**

* **E1: Mã giảng viên bị trùng:**
  + **Xảy ra tại bước 7.** Hệ thống phát hiện Mã giảng viên đã tồn tại.
  + Hệ thống hiển thị lỗi: "Lỗi: Mã giảng viên đã tồn tại. Vui lòng nhập mã khác." và quay lại bước 5.
* **E2: Dữ liệu bắt buộc bị bỏ trống:**
  + **Xảy ra tại bước 7.** Hệ thống phát hiện trường bắt buộc (ví dụ: Họ tên) bị bỏ trống.
  + Hệ thống hiển thị lỗi: "Lỗi: Vui lòng điền đầy đủ các trường bắt buộc." và quay lại bước 5.
* **E3: Xóa giảng viên đã được phân công giảng dạy:**
  + Khi người dùng thực hiện thao tác "Xóa".
  + Hệ thống kiểm tra và phát hiện giảng viên đang có lịch dạy trong học kỳ.
  + Hệ thống hiển thị cảnh báo: "Không thể xóa giảng viên này vì đã được phân công giảng dạy." Thao tác bị hủy.

#### **UC2: Phân công giảng dạy**

* **Actor chính:** Phòng Đào Tạo/Khoa
* **Actor phụ:** Giảng viên (người xem kết quả)
* **Mục tiêu:** Gán môn học, lớp, thời gian, phòng học cho giảng viên và kiểm tra trùng lịch.
* **Tiền điều kiện:** Đã có dữ liệu giảng viên, môn học, lớp học, phòng học. Người dùng đã đăng nhập với quyền phù hợp.
* **Hậu điều kiện thành công:** Lịch giảng dạy được lưu và giảng viên có thể xem được.

**Luồng sự kiện chính:**

1. Người dùng chọn chức năng "Phân công giảng dạy".
2. Người dùng chọn học kỳ và lớp học phần cần phân công.
3. Người dùng chọn một Giảng viên, nhập thời gian (Thứ, Tiết) và chọn một Phòng học.
4. Người dùng nhấn "Lưu phân công".
5. Hệ thống kiểm tra tính hợp lệ và đầy đủ của dữ liệu.
6. Hệ thống kiểm tra lịch của Giảng viên và Phòng học để đảm bảo không bị trùng.
7. Hệ thống lưu thông tin phân công vào CSDL.
8. Hệ thống hiển thị thông báo "Phân công thành công!".

**Luồng rẽ nhánh & Ngoại lệ:**

* **E1: Giảng viên bị trùng lịch:**
  + **Xảy ra tại bước 6.** Hệ thống phát hiện giảng viên đã có một lớp khác vào cùng thời điểm.
  + Hệ thống hiển thị lỗi: "Lỗi: Giảng viên [Tên] đã bị trùng lịch." và quay lại bước 3.
* **E2: Phòng học bị trùng lịch:**
  + **Xảy ra tại bước 6.** Hệ thống phát hiện phòng học đã được sử dụng vào cùng thời điểm.
  + Hệ thống hiển thị lỗi: "Lỗi: Phòng học [Tên phòng] đã bị trùng lịch." và quay lại bước 3.

#### **UC3: Quản lý môn học**

* **Actor chính:** Phòng Đào Tạo/Khoa
* **Mục tiêu:** Quản lý danh mục môn học (tạo, sửa, xóa, tra cứu).
* **Tiền điều kiện:** Người dùng đã đăng nhập.

**Luồng sự kiện chính (Thêm mới):**

1. Người dùng chọn "Quản lý môn học".
2. Người dùng chọn "Thêm môn học mới".
3. Người dùng nhập Mã môn học, Tên môn học, Số tín chỉ.
4. Người dùng nhấn "Lưu".
5. Hệ thống kiểm tra mã môn học chưa tồn tại.
6. Hệ thống lưu môn học mới vào CSDL và báo thành công.

**Luồng rẽ nhánh & Ngoại lệ:**

* **E1: Mã môn học đã tồn tại:**
  + **Xảy ra tại bước 5.** Hệ thống báo lỗi: "Lỗi: Mã môn học này đã tồn tại." và quay lại bước 3.
* **E2: Xóa môn học đang được sử dụng:**
  + Khi người dùng thực hiện thao tác "Xóa".
  + Hệ thống phát hiện môn học đã được gán cho lớp học phần.
  + Hệ thống hiển thị cảnh báo: "Không thể xóa môn học này vì đang được sử dụng." Thao tác bị hủy.

#### **UC4: Quản lý lớp học**

* **Actor chính:** Phòng Đào Tạo/Khoa
* **Mục tiêu:** Tạo, cập nhật và xóa thông tin lớp học.
* **Tiền điều kiện:** Người dùng đã đăng nhập.

**Luồng sự kiện chính (Tạo lớp):**

1. Người dùng chọn "Quản lý lớp học".
2. Người dùng chọn "Tạo lớp học mới".
3. Người dùng nhập thông tin: Mã lớp, Tên lớp, Sĩ số, Khoa quản lý, Học kỳ.
4. Người dùng nhấn "Lưu".
5. Hệ thống kiểm tra Mã lớp chưa tồn tại.
6. Hệ thống lưu thông tin lớp học vào CSDL và báo thành công.

**Luồng rẽ nhánh & Ngoại lệ:**

* **E1: Mã lớp đã tồn tại:**
  + **Xảy ra tại bước 5.** Hệ thống báo lỗi: "Lỗi: Mã lớp này đã tồn tại." và quay lại bước 3.
* **E2: Dữ liệu sĩ số không hợp lệ:**
  + **Xảy ra tại bước 5.** Người dùng nhập sĩ số là chữ hoặc số âm.
  + Hệ thống báo lỗi: "Sĩ số phải là một số nguyên dương." và quay lại bước 3.

#### **UC5: Quản lý phòng học**

* **Actor chính:** Phòng Đào Tạo/Khoa
* **Mục tiêu:** Quản lý danh sách phòng học (thêm, sửa, xóa, tình trạng).
* **Tiền điều kiện:** Người dùng đã đăng nhập.

**Luồng sự kiện chính (Thêm phòng học):**

1. Người dùng chọn "Quản lý phòng học".
2. Người dùng chọn "Thêm phòng học mới".
3. Người dùng nhập thông tin: Mã phòng, Sức chứa, Trạng thái.
4. Người dùng nhấn "Lưu".
5. Hệ thống kiểm tra Mã phòng chưa tồn tại.
6. Hệ thống lưu phòng học mới vào CSDL và báo thành công.

**Luồng rẽ nhánh & Ngoại lệ:**

* **E1: Mã phòng đã tồn tại:**
  + **Xảy ra tại bước 5.** Hệ thống báo lỗi: "Lỗi: Mã phòng này đã tồn tại." và quay lại bước 3.
* **E2: Xóa phòng học đang được sử dụng:**
  + Khi người dùng thực hiện thao tác "Xóa".
  + Hệ thống phát hiện phòng học đang có trong một lịch dạy.
  + Hệ thống hiển thị cảnh báo: "Không thể xóa phòng học này vì đang được sử dụng." Thao tác bị hủy.

#### **UC6: Xem lịch giảng dạy**

* **Actor chính:** Giảng viên
* **Mục tiêu:** Giảng viên tra cứu lịch giảng dạy của cá nhân mình.
* **Tiền điều kiện:** Giảng viên đã đăng nhập và lịch đã được phân công.
* **Hậu điều kiện thành công:** Lịch giảng dạy được hiển thị chính xác.

**Luồng sự kiện chính:**

1. Giảng viên chọn "Xem lịch giảng dạy".
2. Hệ thống mặc định hiển thị lịch của tuần hiện tại.
3. Giảng viên có thể chọn xem theo tuần khác hoặc toàn bộ học kỳ.
4. Hệ thống truy xuất và hiển thị lịch dạy tương ứng (Tên môn, Lớp, Phòng, Thời gian).

**Luồng rẽ nhánh & Ngoại lệ:**

* **A1: Giảng viên không có lịch dạy:**
  + **Xảy ra tại bước 4.** Hệ thống không tìm thấy lịch dạy nào trong khoảng thời gian đã chọn.
  + Hệ thống hiển thị thông báo: "Bạn không có lịch giảng dạy trong thời gian này."

#### **UC7: Cập nhật thông tin cá nhân**

* **Actor chính:** Giảng viên
* **Mục tiêu:** Cho phép giảng viên chỉnh sửa thông tin liên hệ cá nhân.
* **Tiền điều kiện:** Giảng viên đã đăng nhập.

**Luồng sự kiện chính:**

1. Giảng viên chọn "Thông tin cá nhân".
2. Hệ thống hiển thị thông tin hiện tại.
3. Giảng viên chỉnh sửa Email, Số điện thoại.
4. Giảng viên nhấn "Lưu thay đổi".
5. Hệ thống kiểm tra tính hợp lệ của dữ liệu mới (định dạng email, sđt).
6. Hệ thống cập nhật thông tin và báo thành công.

**Luồng rẽ nhánh & Ngoại lệ:**

* **E1: Email không đúng định dạng:**
  + **Xảy ra tại bước 5.** Hệ thống báo lỗi: "Định dạng email không hợp lệ." và quay lại bước 3.
* **E2: Số điện thoại không hợp lệ:**
  + **Xảy ra tại bước 5.** Hệ thống báo lỗi: "Số điện thoại chỉ được chứa ký tự số." và quay lại bước 3.

#### **UC8: Xem báo cáo giảng viên**

* **Actor chính:** Ban Giám Hiệu
* **Mục tiêu:** Xem báo cáo về giảng viên (số lượng, trình độ, lịch giảng dạy).
* **Tiền điều kiện:** Đã đăng nhập với quyền Ban Giám Hiệu.

**Luồng sự kiện chính:**

1. Người dùng chọn "Báo cáo – Thống kê".
2. Người dùng chọn loại báo cáo và các bộ lọc (nếu có).
3. Người dùng nhấn "Xem báo cáo".
4. Hệ thống xử lý, hiển thị báo cáo và cho phép xuất file PDF/Excel.

**Luồng rẽ nhánh & Ngoại lệ:**

* **A1: Không có dữ liệu để tạo báo cáo:**
  + **Xảy ra tại bước 4.** Hệ thống không tìm thấy dữ liệu phù hợp.
  + Hệ thống hiển thị thông báo: "Không có dữ liệu phù hợp để tạo báo cáo."

#### **UC9: Quản lý thông tin giảng viên (dành cho BGH)**

* **Actor chính:** Ban Giám Hiệu
* **Mục tiêu:** Cho phép Ban Giám Hiệu xem và chỉnh sửa các thông tin chuyên môn của giảng viên.
* **Tiền điều kiện:** Đã đăng nhập với quyền Ban Giám Hiệu.

**Luồng sự kiện chính:**

1. Người dùng truy cập danh sách giảng viên.
2. Người dùng tìm và chọn một giảng viên cụ thể.
3. Hệ thống hiển thị hồ sơ chi tiết của giảng viên.
4. Người dùng chỉnh sửa các thông tin được phép (ví dụ: học vị, phụ cấp, ghi chú đánh giá).
5. Người dùng nhấn "Lưu".
6. Hệ thống cập nhật thông tin và báo thành công.

**Luồng rẽ nhánh & Ngoại lệ:**

* **E1: Dữ liệu nhập không hợp lệ:**
  + **Xảy ra tại bước 6.** Ví dụ: nhập chữ vào ô phụ cấp (số).
  + Hệ thống báo lỗi và yêu cầu nhập lại.

#### **UC10: Quản lý tài khoản hệ thống**

* **Actor chính:** IT Admin
* **Mục tiêu:** Quản lý tài khoản người dùng (tạo, sửa, khóa, xóa).
* **Tiền điều kiện:** IT Admin đã đăng nhập.

**Luồng sự kiện chính (Tạo tài khoản):**

1. Admin chọn "Quản lý tài khoản".
2. Admin chọn "Tạo tài khoản mới".
3. Admin nhập Tên đăng nhập, Mật khẩu mặc định, Email, và chọn vai trò.
4. Admin nhấn "Lưu".
5. Hệ thống kiểm tra Tên đăng nhập và Email chưa được sử dụng.
6. Hệ thống tạo tài khoản mới và báo thành công.

**Luồng rẽ nhánh & Ngoại lệ:**

* **E1: Tên đăng nhập đã tồn tại:**
  + **Xảy ra tại bước 5.** Hệ thống báo lỗi: "Tên đăng nhập đã được sử dụng." và quay lại bước 3.
* **E2: Khóa tài khoản Admin cuối cùng:**
  + Khi Admin khóa tài khoản.
  + Hệ thống kiểm tra nếu đây là tài khoản IT Admin duy nhất còn hoạt động.
  + Hệ thống báo lỗi: "Không thể khóa tài khoản quản trị viên cuối cùng." Thao tác bị hủy.

#### **UC11: Quản lý phân quyền**

* **Actor chính:** IT Admin
* **Mục tiêu:** Thiết lập quyền truy cập chi tiết cho từng vai trò (Role).
* **Tiền điều kiện:** IT Admin đã đăng nhập.

**Luồng sự kiện chính:**

1. Admin chọn "Phân quyền".
2. Hệ thống hiển thị danh sách các vai trò (BGH, Giảng viên, Phòng Đào Tạo).
3. Admin chọn một vai trò để chỉnh sửa.
4. Hệ thống hiển thị danh sách các quyền (VD: "Thêm giảng viên", "Xóa giảng viên", "Xem báo cáo").
5. Admin tick/bỏ tick các quyền.
6. Admin nhấn "Lưu thay đổi".
7. Hệ thống cập nhật ma trận phân quyền và báo thành công.

**Luồng rẽ nhánh & Ngoại lệ:**

* **E1: Cố gắng bỏ quyền quản trị của vai trò IT Admin:**
  + **Xảy ra tại bước 6.**
  + Hệ thống phát hiện người dùng đang cố gắng bỏ các quyền quản trị cốt lõi khỏi chính vai trò IT Admin.
  + Hệ thống báo lỗi: "Không thể bỏ các quyền quản trị cốt lõi khỏi vai trò Admin." Thao tác bị hủy.

#### **UC12: Đổi mật khẩu**

* **Actor chính:** Tất cả người dùng
* **Mục tiêu:** Cho phép người dùng thay đổi mật khẩu để tăng tính bảo mật.
* **Tiền điều kiện:** Người dùng đã đăng nhập.

**Luồng sự kiện chính:**

1. Người dùng chọn "Đổi mật khẩu".
2. Người dùng nhập Mật khẩu cũ, Mật khẩu mới, và Xác nhận mật khẩu mới.
3. Người dùng nhấn "Lưu thay đổi".
4. Hệ thống kiểm tra Mật khẩu cũ chính xác.
5. Hệ thống kiểm tra Mật khẩu mới và Mật khẩu xác nhận trùng khớp.
6. Hệ thống cập nhật mật khẩu mới và báo thành công.

**Luồng rẽ nhánh & Ngoại lệ:**

* **E1: Mật khẩu cũ không chính xác:**
  + **Xảy ra tại bước 4.** Hệ thống báo lỗi: "Mật khẩu cũ không đúng." và quay lại bước 2.
* **E2: Mật khẩu xác nhận không khớp:**
  + **Xảy ra tại bước 5.** Hệ thống báo lỗi: "Mật khẩu xác nhận không khớp." và quay lại bước 2.
* **E3: Mật khẩu mới không đủ mạnh (nếu có quy định):**
  + **Xảy ra tại bước 5.** Hệ thống báo lỗi: "Mật khẩu mới quá yếu." và quay lại bước 2.

# **Y3. Thiết kế kiến trúc hệ thống**

**1. Mô hình kiến trúc được chọn: MVC (Model – View – Controller)**

**Lý do lựa chọn:**

* Hệ thống có giao diện người dùng (web/app) và nhiều nhóm người dùng khác nhau như **Ban Giám Hiệu, Phòng Đào Tạo, Giảng viên, Quản trị viên**.
* **MVC** giúp tách biệt rõ ràng giữa:
* **View:** giao diện người dùng.
* **Controller:** xử lý và điều phối yêu cầu.
* **Model:** dữ liệu và nghiệp vụ.
* Việc phân tách này giúp **dễ mở rộng, bảo trì, kiểm thử và tái sử dụng** mã nguồn.
* Mô hình phù hợp với các ứng dụng web hiện đại phát triển bằng **Spring Boot** hoặc **.NET MVC**.

**2. Mô tả các thành phần kiến trúc**

| **Thành Phần** | **Vai trò** | **Mô tả chi tiết** |
| --- | --- | --- |
| **Model** | **Dữ liệu & Nghiệp vụ** | Chứa các lớp đại diện cho đối tượng trong hệ thống như: GiangVien  , Khoa  , LichDay  , PhanCong  , PhanHoi  , TaiKhoan  Thực hiện nghiệp vụ và giao tiếp với cơ sở dữ liệu. |
| **View** | **Giao diện người dùng** | Hiển thị thông tin và nhận thao tác từ người dùng.  Ví dụ:  trang “Quản lý hồ sơ giảng viên”  , trang “Phân công lịch dạy”  , trang “Xem báo cáo thống kê”. |
| **Controller** | **Điều phối & Xử lý yêu cầu** | Tiếp nhận yêu cầu từ người dùng, gọi đến Model hoặc Service để xử lý, sau đó trả kết quả về View.  Ví dụ:  GiangVienController  , LichDayController  , BaoCaoController |
| **Database Layer** | **Lưu trữ dữ liệu** | Cơ sở dữ liệu quan hệ (MySQL/PostgreSQL) chứa các bảng: GiangVien  , Khoa  , LichDay  , PhanHoi  , TaiKhoan  , VaiTro |
| **Service Layer**  **(tuỳ chọn)** | **Tách logic nghiệp vụ** | Đặt giữa Controller và Model, dùng trong hệ thống có nghiệp vụ phức tạp nhằm giảm tải Controller và tăng khả năng tái sử dụng. |

**3. Sơ đồ kiến trúc tổng quan**

**Mô tả luồng hoạt động:**

1. Người dùng (BGH, Giảng viên, Phòng Đào Tạo, Admin) truy cập hệ thống qua giao diện web.
2. Yêu cầu được gửi đến **Controller** tương ứng.
3. **Controller** gọi **Service** hoặc **Model** để xử lý nghiệp vụ và truy xuất dữ liệu từ **Database**.
4. Kết quả được trả về **View** để hiển thị cho người dùng.

**Sơ đồ tổng quan :**

[Người dùng]

│

▼

[View (Giao diện Web)]

│

▼

[Controller] ───> [Service Layer] ───> [Model] ───> [Database]

│ ▲

└─────────────── Nhận dữ liệu hiển thị ─┘

**Ví dụ:**khi **Giảng viên đăng nhập**, yêu cầu được gửi đến LoginController, sau đó gọi TaiKhoanService để kiểm tra thông tin trong Database. Kết quả được gửi lại cho LoginView hiển thị thông báo đăng nhập thành công hoặc thất bại.

**4. Công nghệ đề xuất**

| **Thành phần** | **Công nghệ sử dụng** |
| --- | --- |
| **Frontend** | HTML, CSS, JavaScript, hoặc framework như Vue.js / React |
| **Backend** | Spring Boot (Java) theo mô hình MVC |
| **Database** | MySQL hoặc PostgreSQL |
| **Triển khai** | Server nội bộ hoặc nền tảng Cloud (Heroku, Render, AWS) |

**5. Ưu điểm của mô hình MVC**

* Tách biệt rõ ràng giữa các tầng: dễ dàng **phát triển song song** và **bảo trì**.
* **Tăng khả năng mở rộng** khi thêm tính năng mới.
* **Dễ kiểm thử** và tái sử dụng mã nguồn.
* Phù hợp cho các **ứng dụng web hiện đại** và **đa người dùng**.

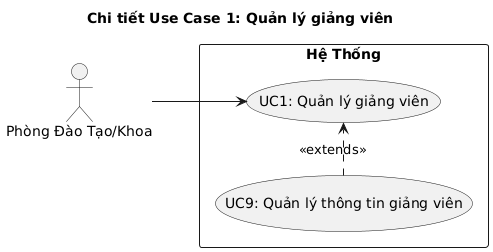
# 

# **Y4. Thiết kế UML**

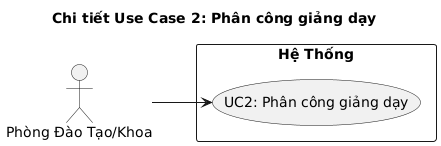
### **Use Case Diagram:**

#### UC1: Quản lý giảng viên

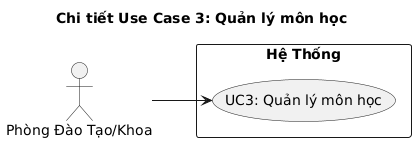
#### 



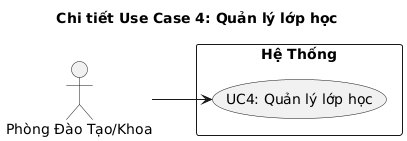
#### UC2: Phân công giảng dạy



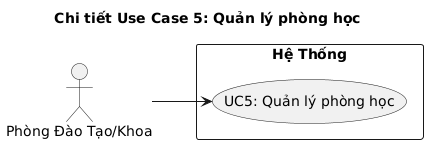
#### UC3: Quản lý môn học



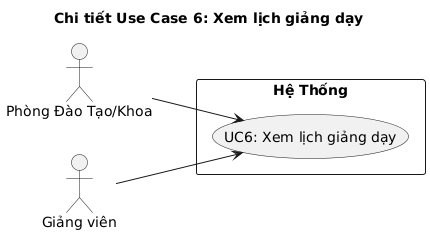
#### UC4: Quản lý lớp học



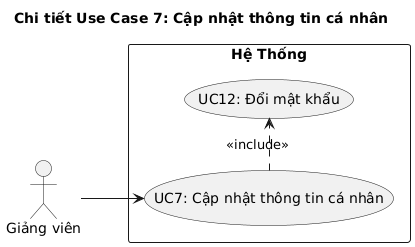
#### UC5: Quản lý phòng học



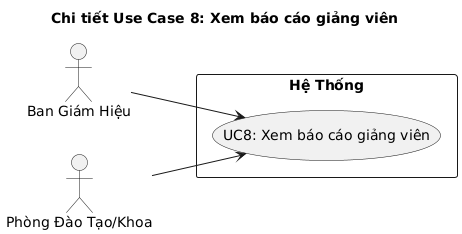
#### UC6: Xem lịch giảng dạy



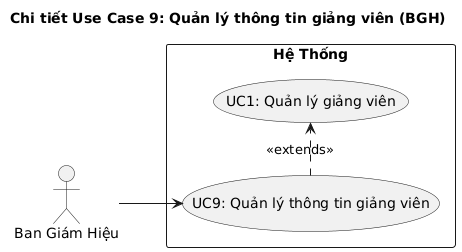
#### UC7: Cập nhật thông tin cá nhân



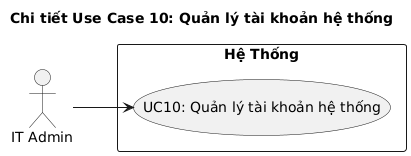
#### UC8: Xem báo cáo giảng viên



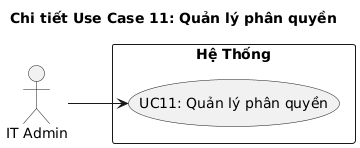
#### UC9: Quản lý thông tin giảng viên (xem/chỉnh sửa)



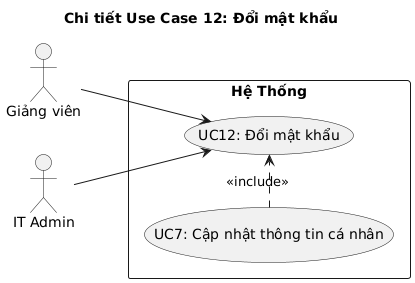
#### UC10: Quản lý tài khoản hệ thống



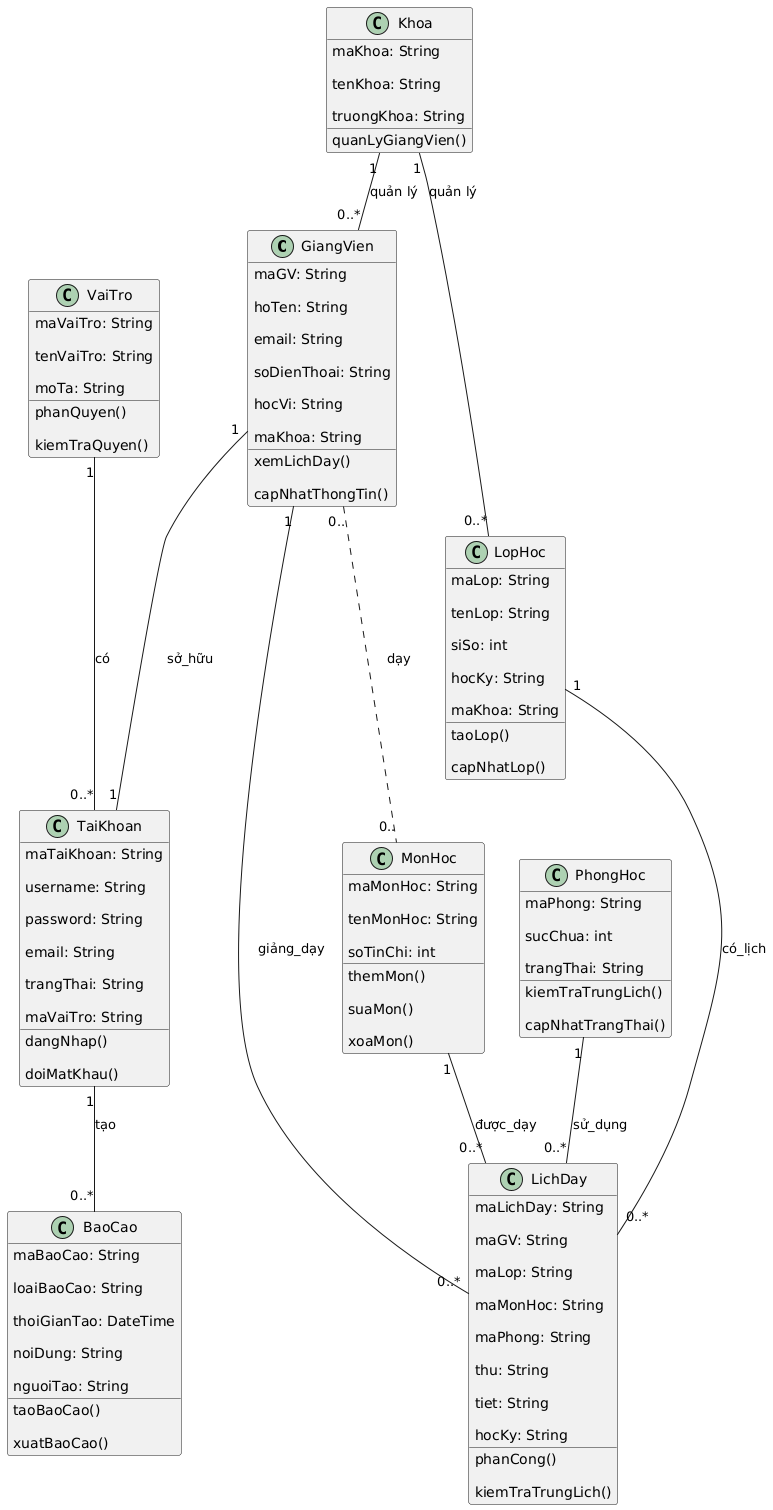
#### UC11: Quản lý phân quyền



#### UC12: Đổi mật khẩu

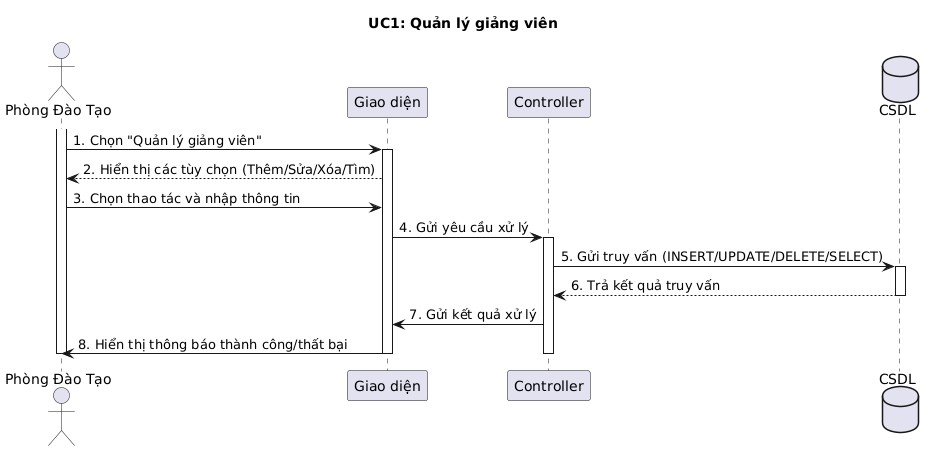


### **Class & Object Diagram:**

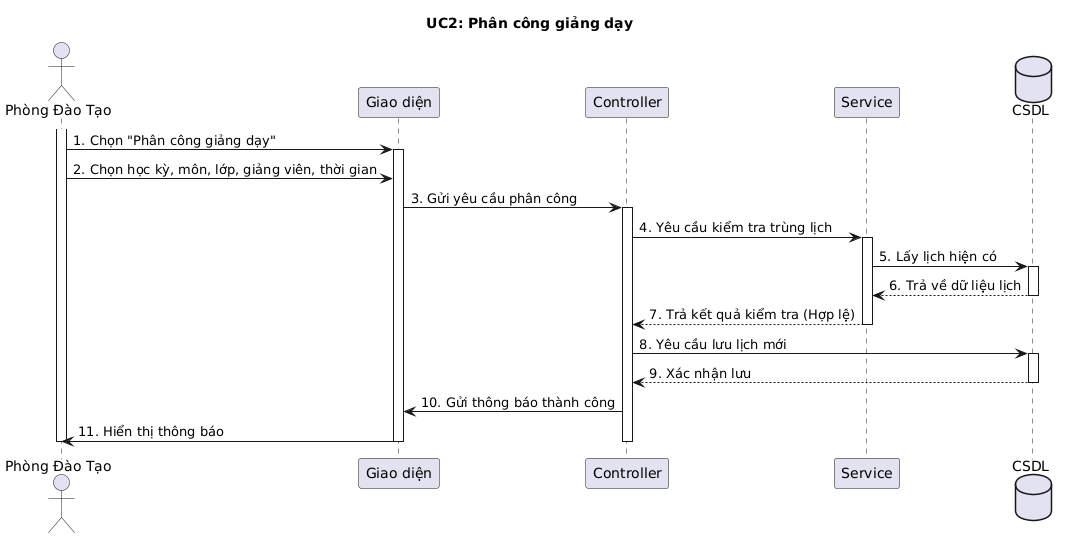


### **Sequence Diagram:**

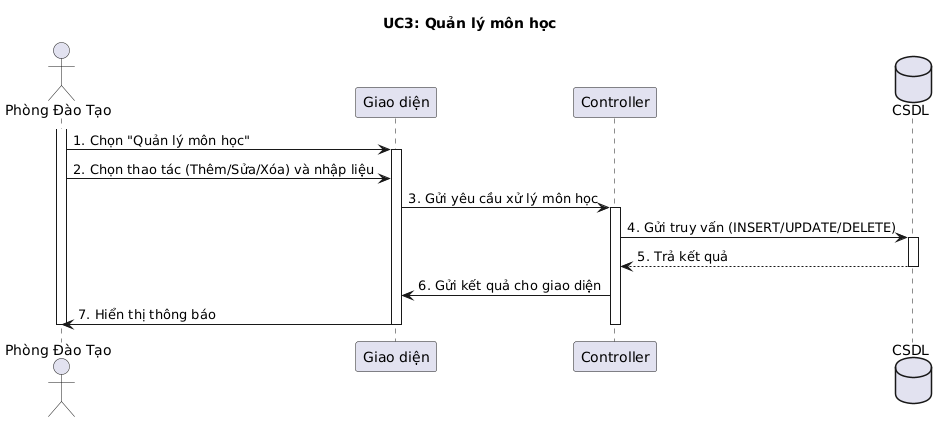
#### UC1: Quản lý giảng viên



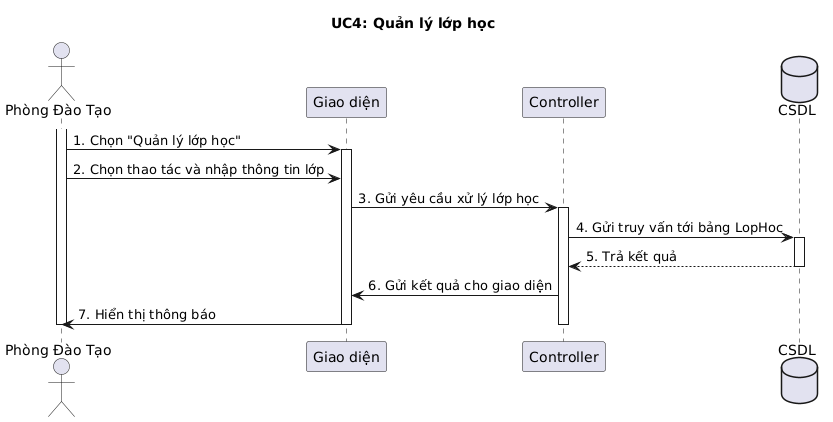
#### UC2: Phân công giảng dạy



#### UC3: Quản lý môn học



#### UC4: Quản lý lớp học



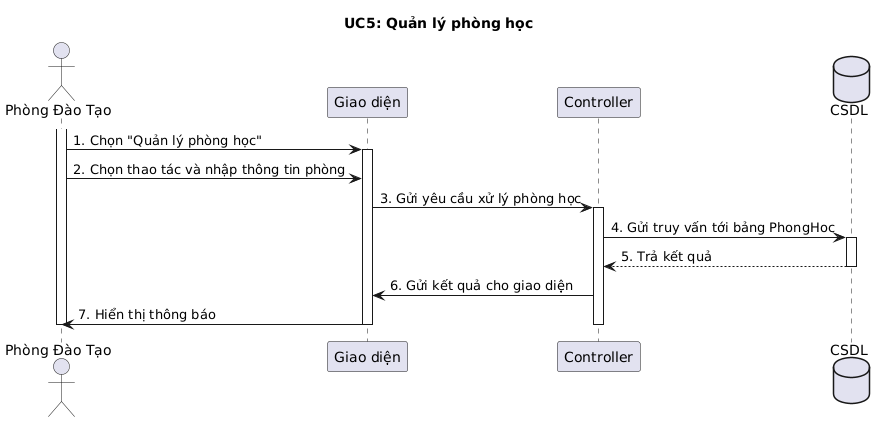
#### 

#### 

#### 

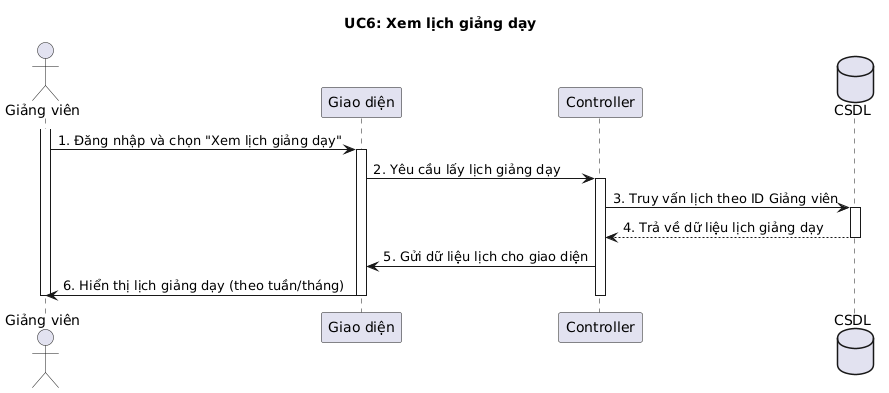
#### 

#### UC5: Quản lý phòng học



#### 

#### UC6: Xem lịch giảng dạy



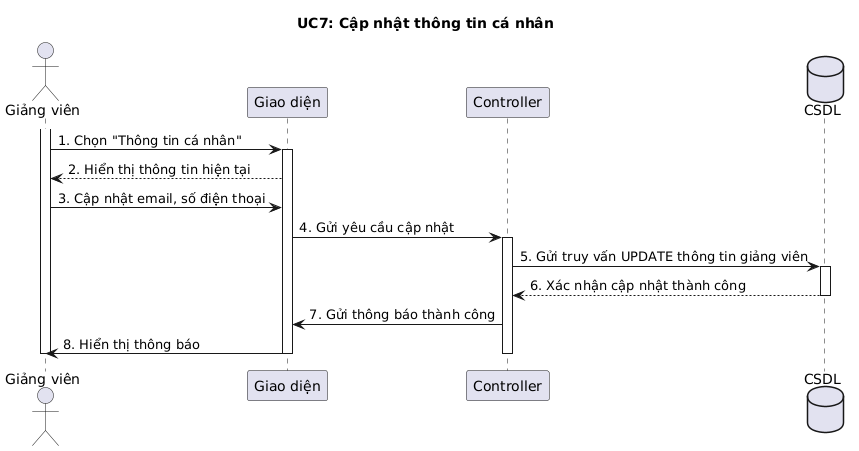
#### 

#### 

#### 

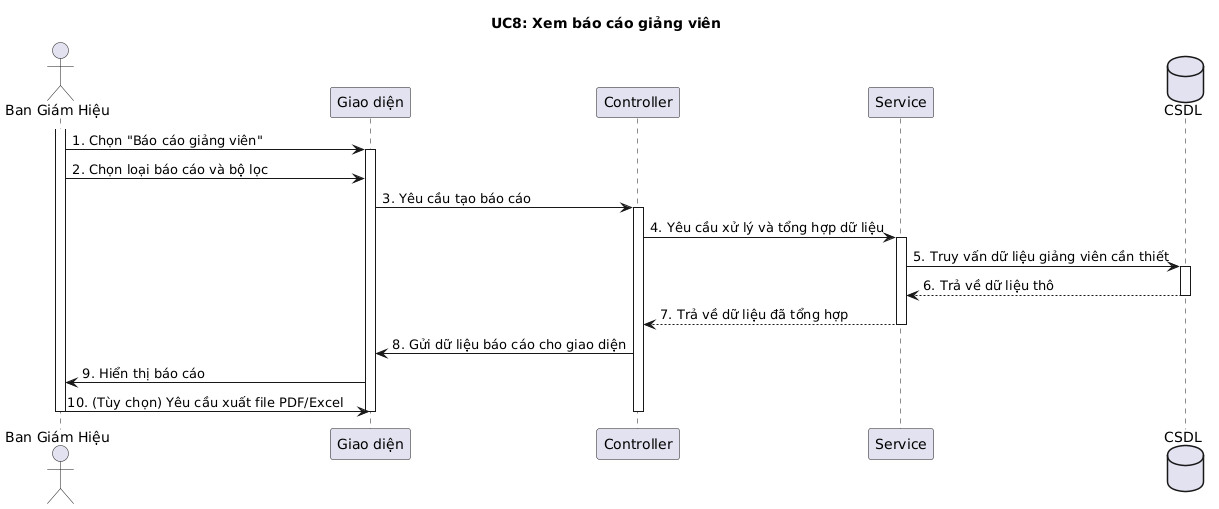
#### 

#### UC7: Cập nhật thông tin cá nhân



#### 

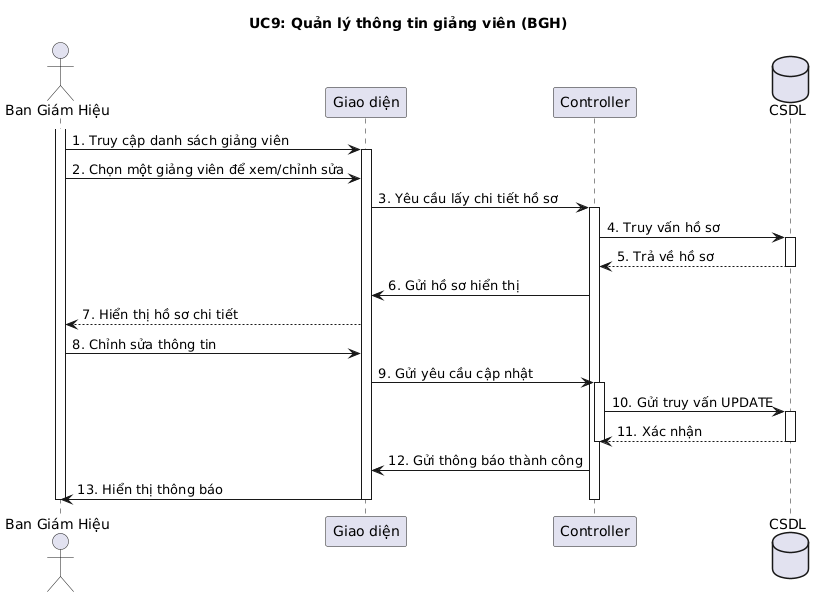
#### UC8: Xem báo cáo giảng viên



#### 

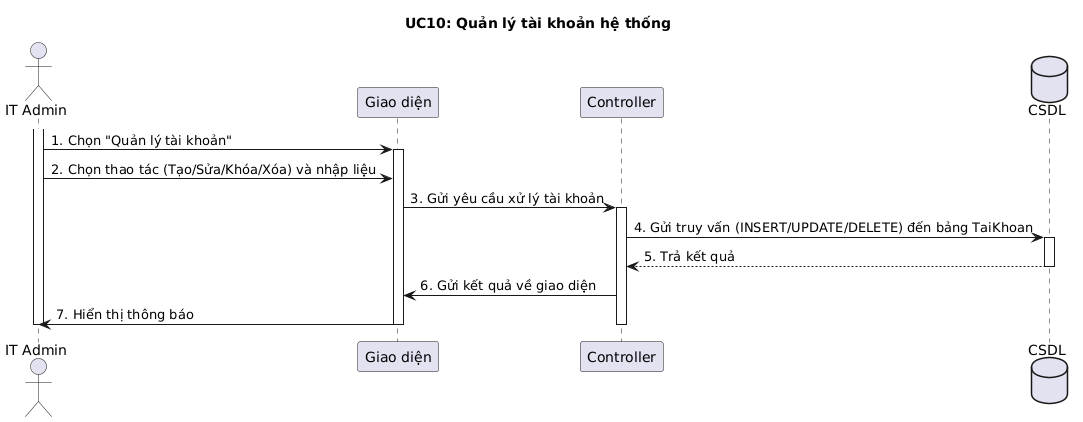
#### 

#### UC9: Quản lý thông tin giảng viên (xem/chỉnh sửa)

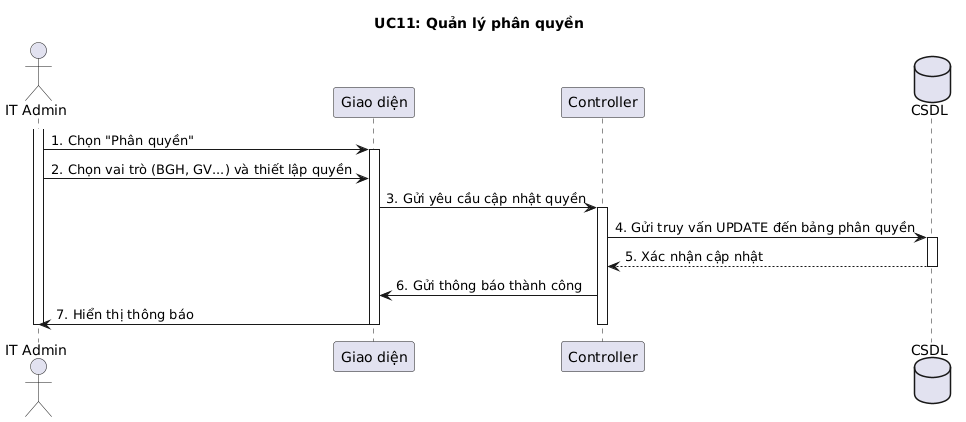


#### 

#### UC10: Quản lý tài khoản hệ thống

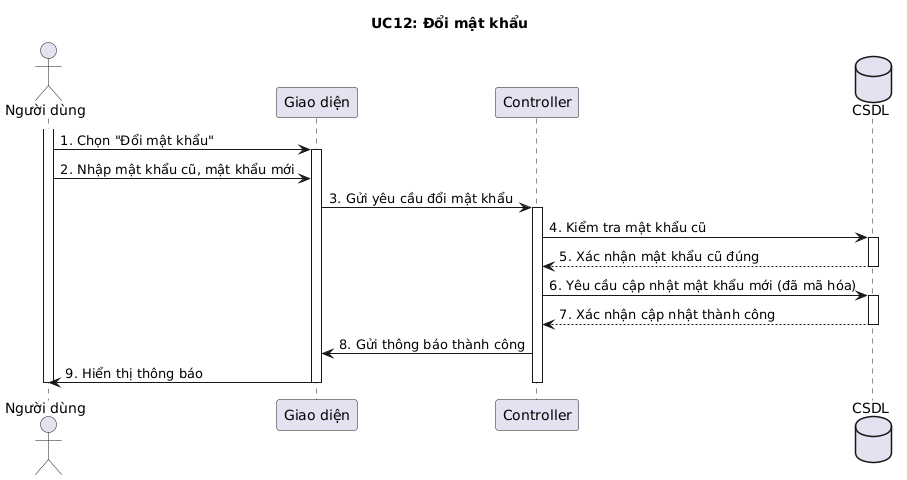


#### UC11: Quản lý phân quyền



#### 

#### UC12: Đổi mật khẩu

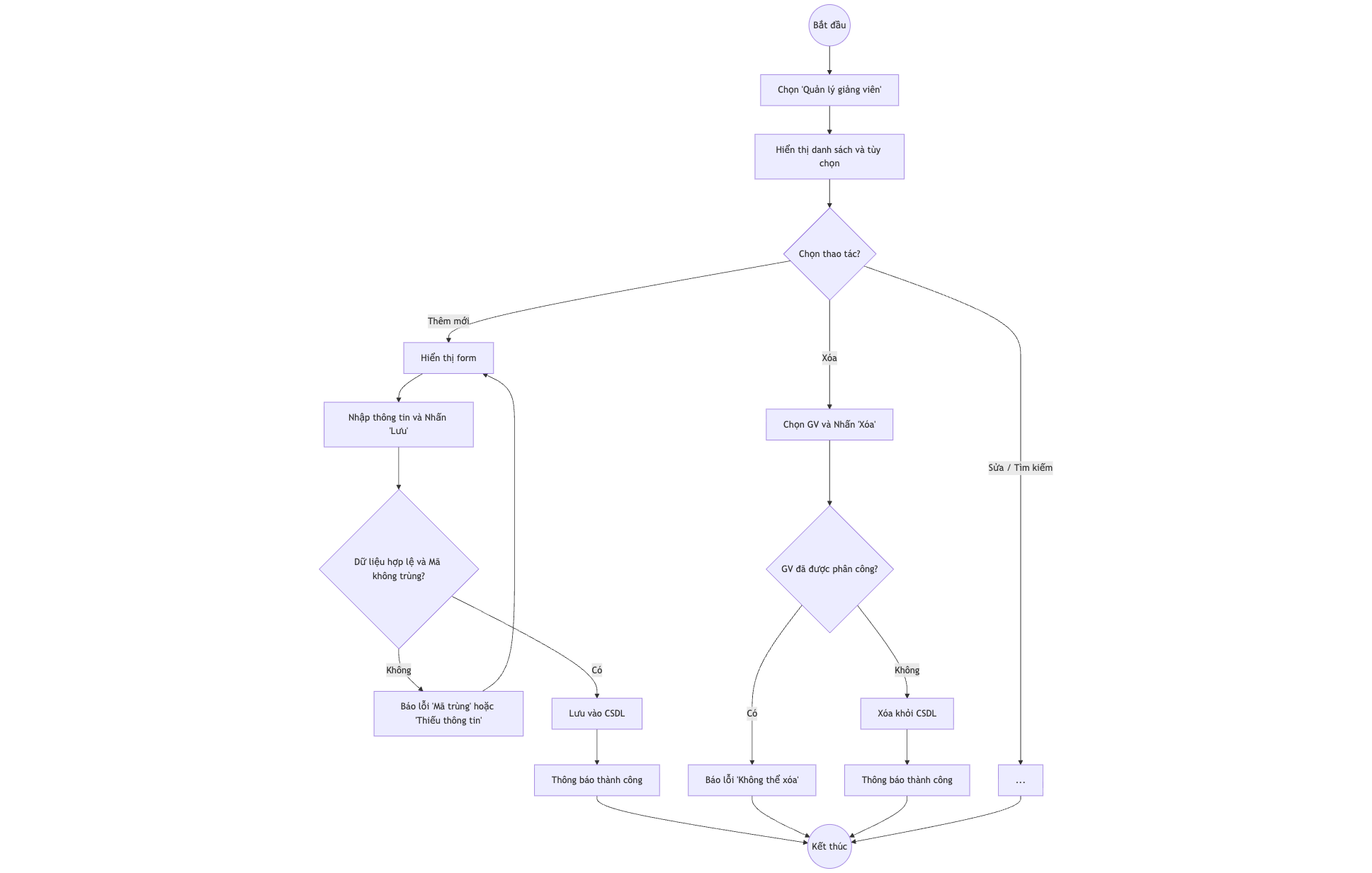


### 

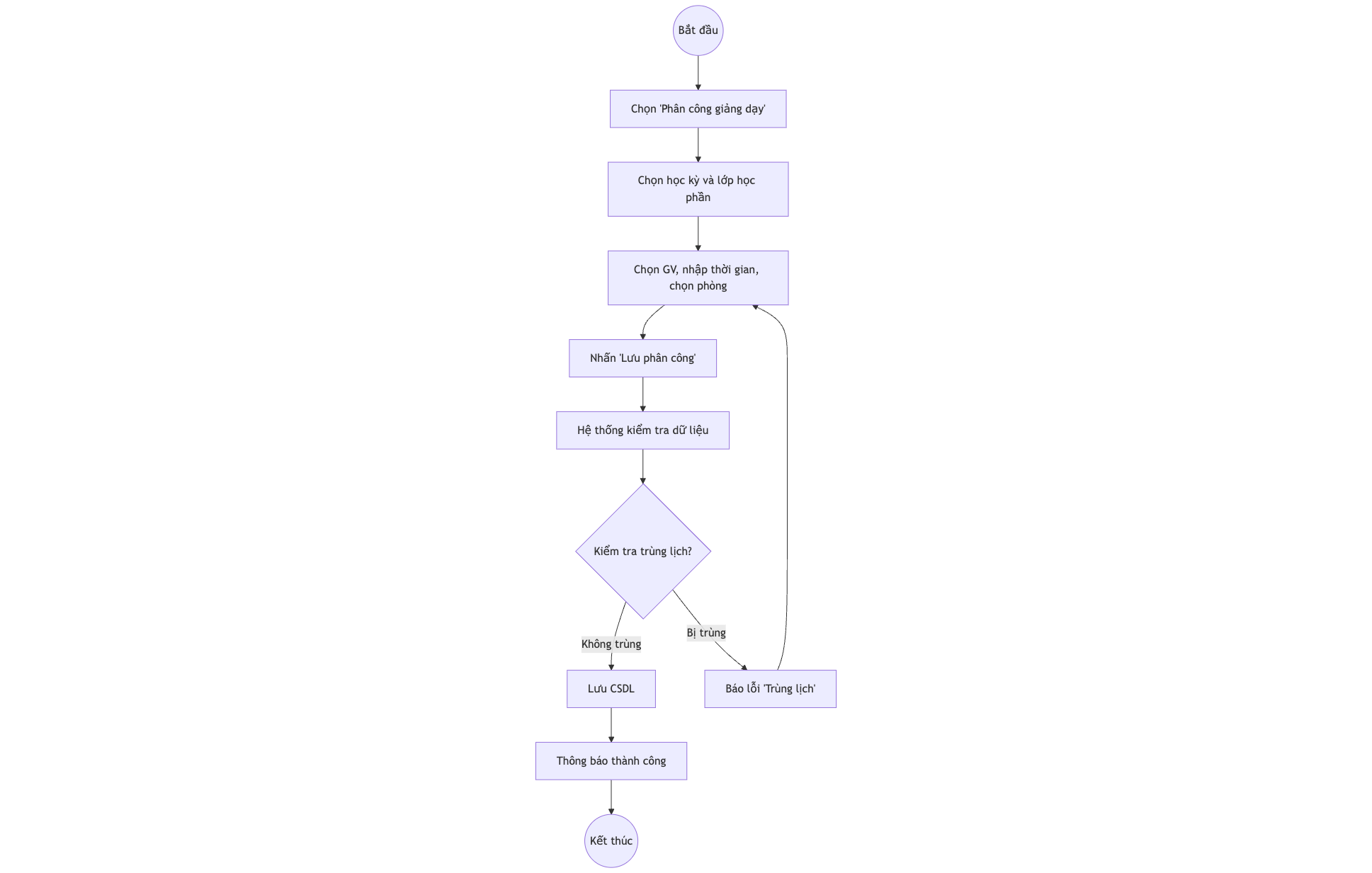
### 

### **Activity Diagram:**

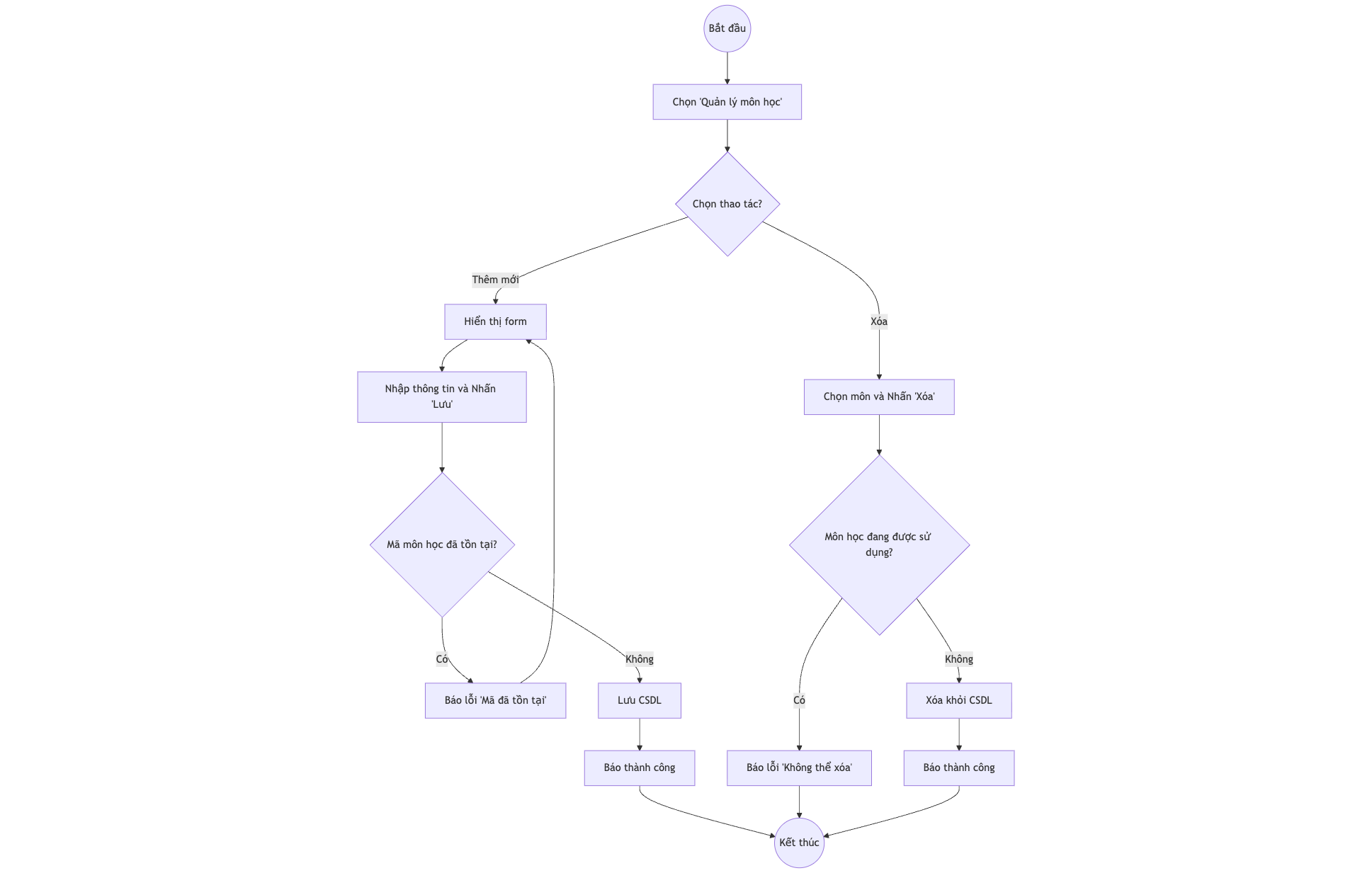
#### UC1: Quản lý giảng viên



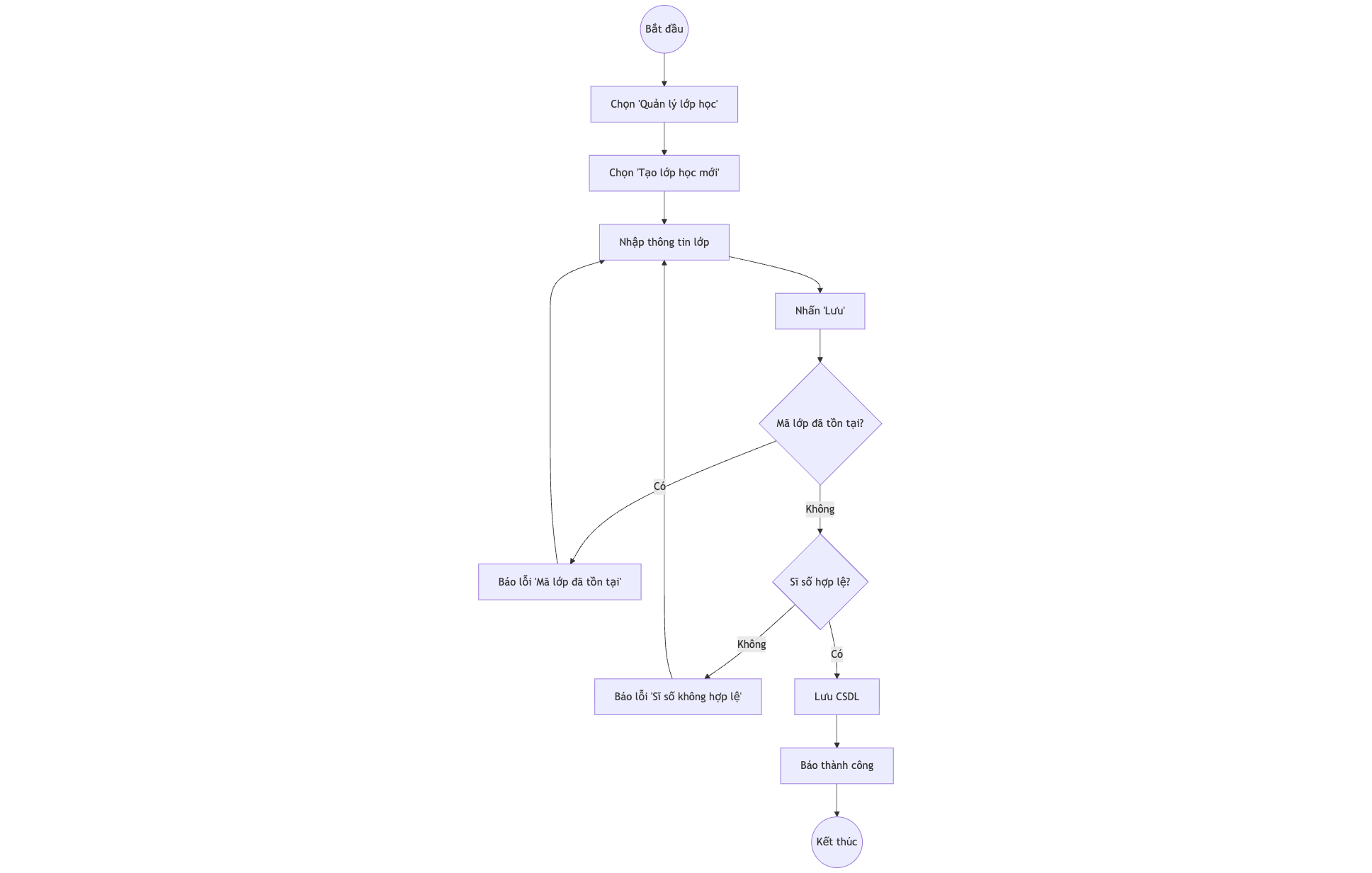
#### UC2: Phân công giảng dạy



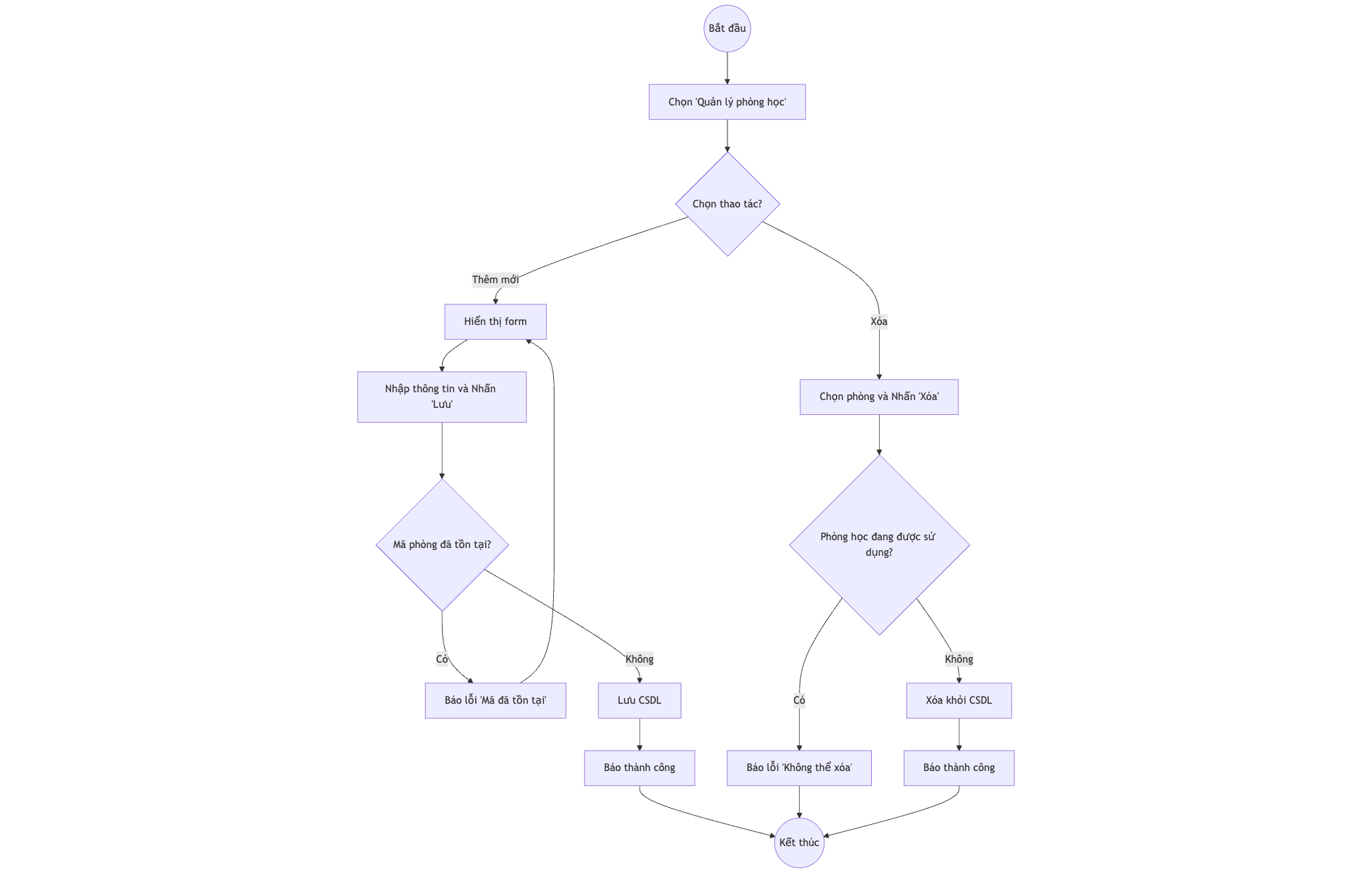
#### UC3: Quản lý môn học



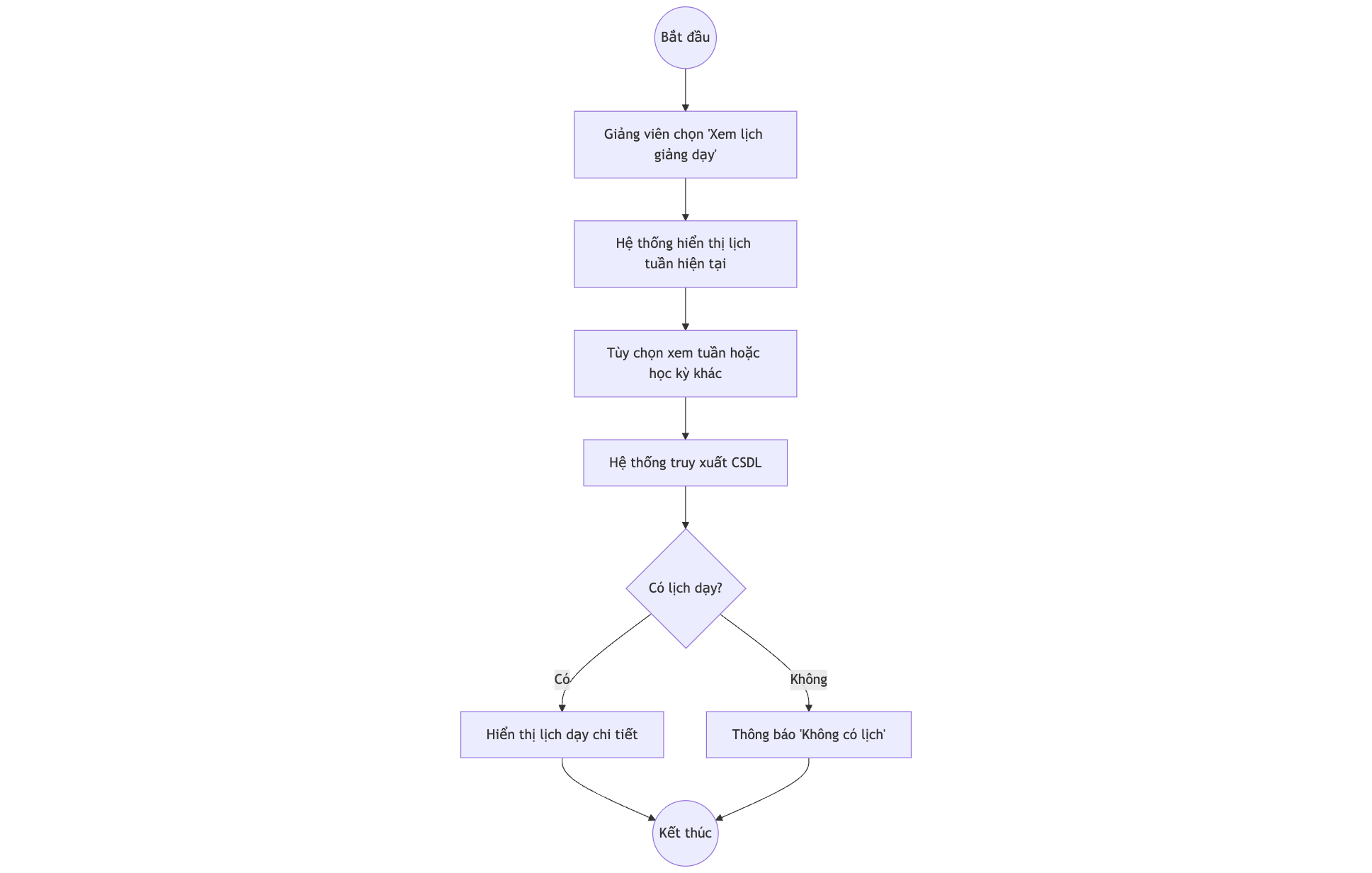
#### UC4: Quản lý lớp học



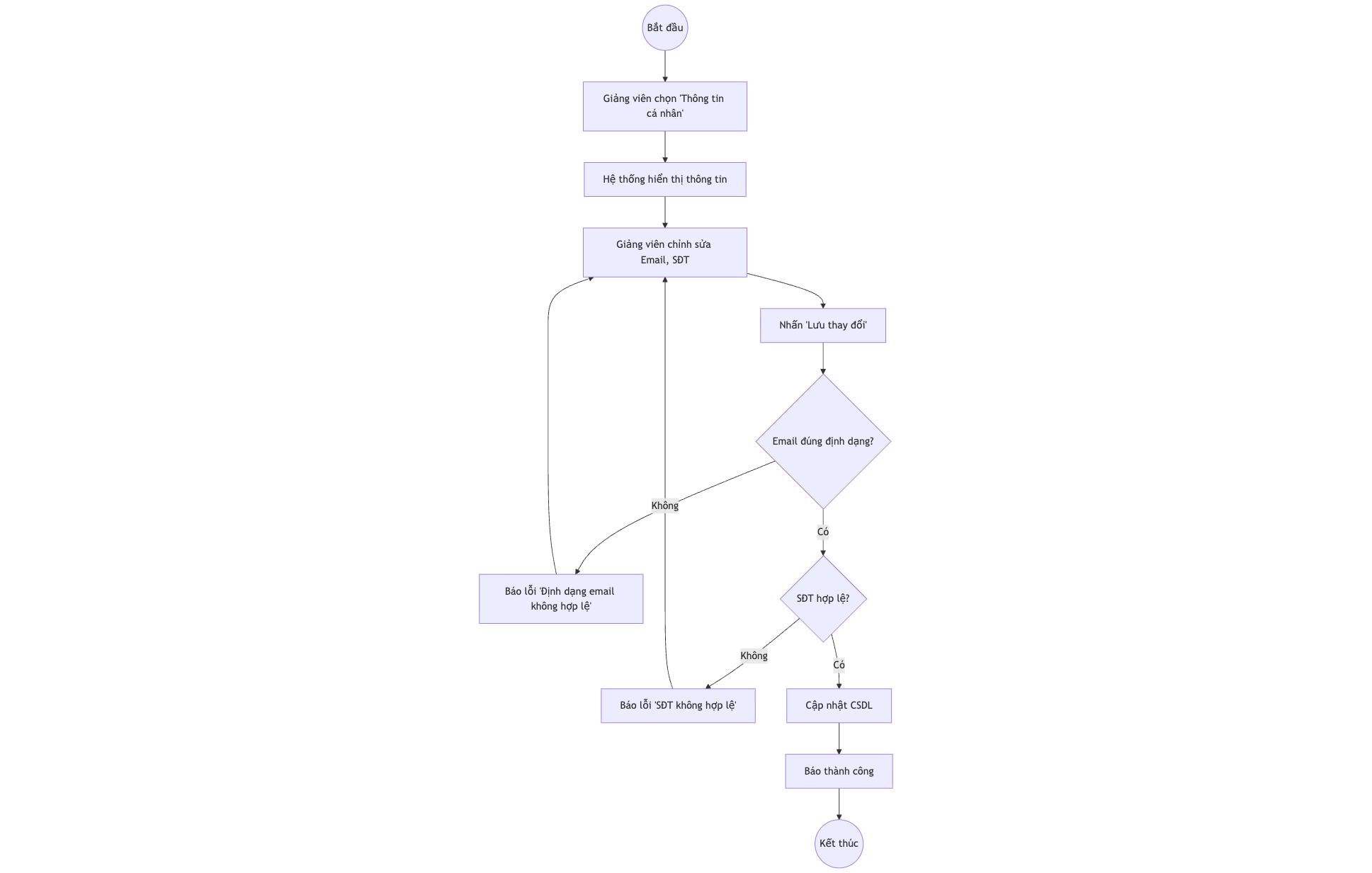
#### UC5: Quản lý phòng học



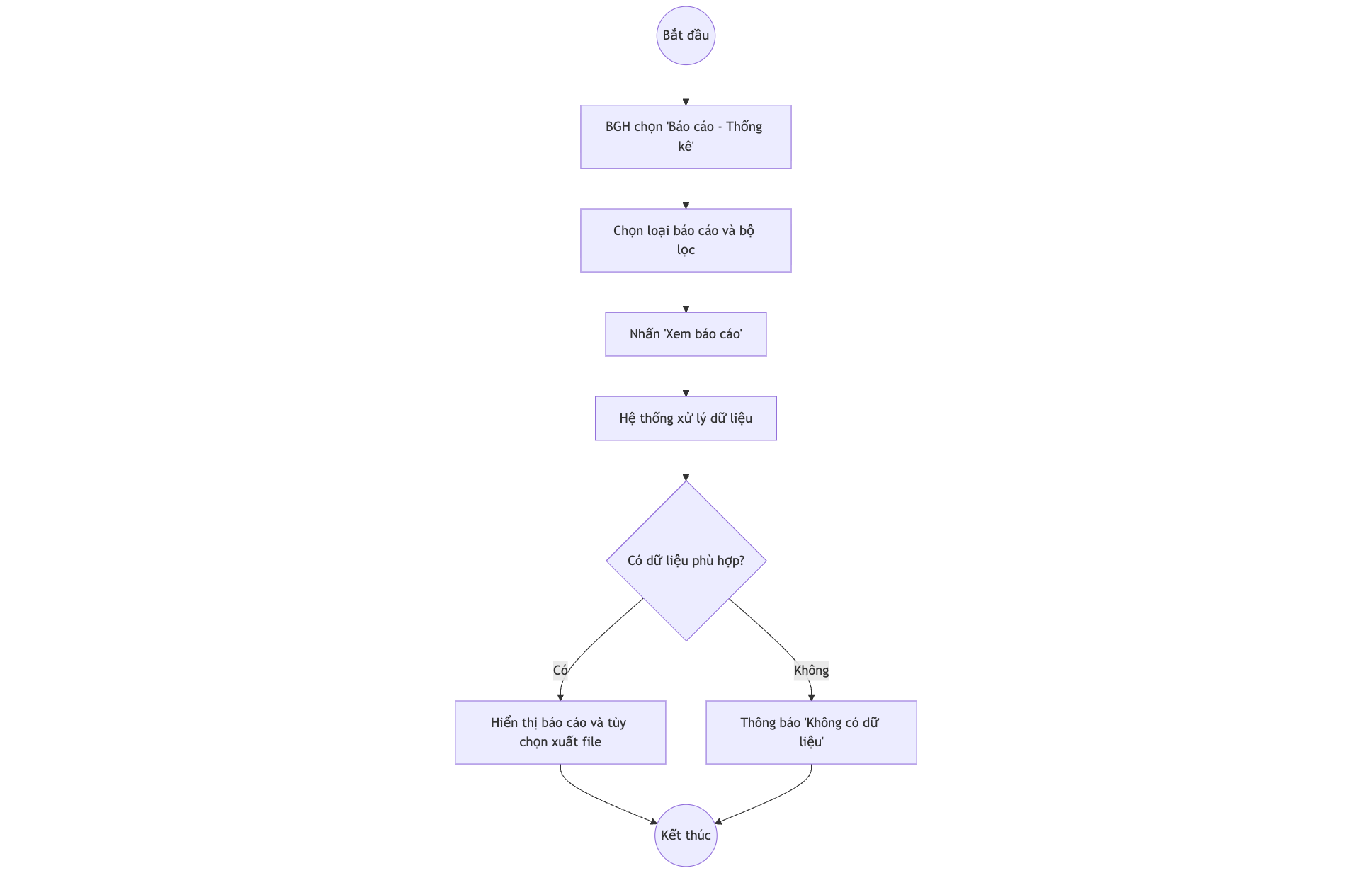
#### UC6: Xem lịch giảng dạy



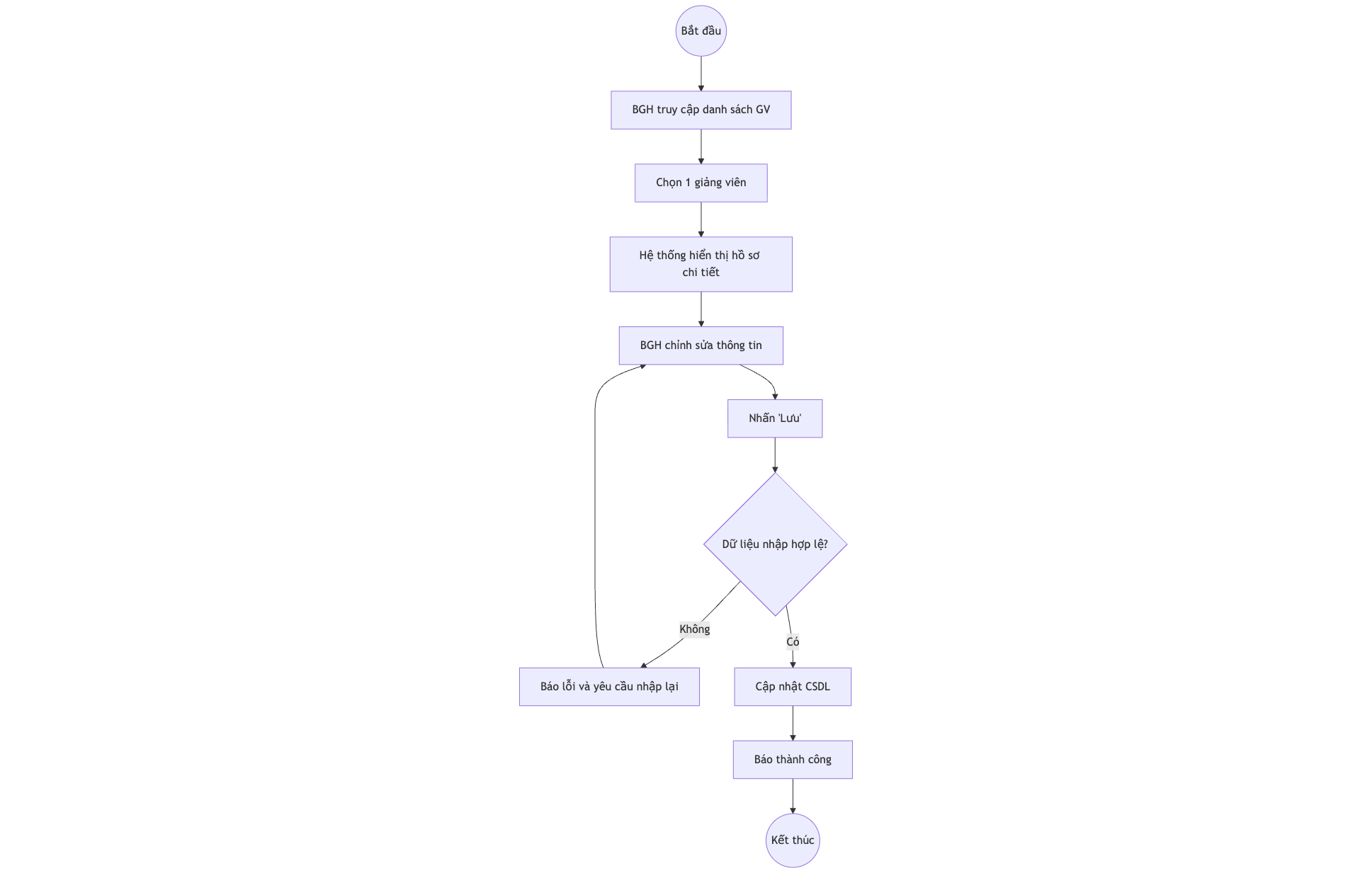
#### UC7: Cập nhật thông tin cá nhân



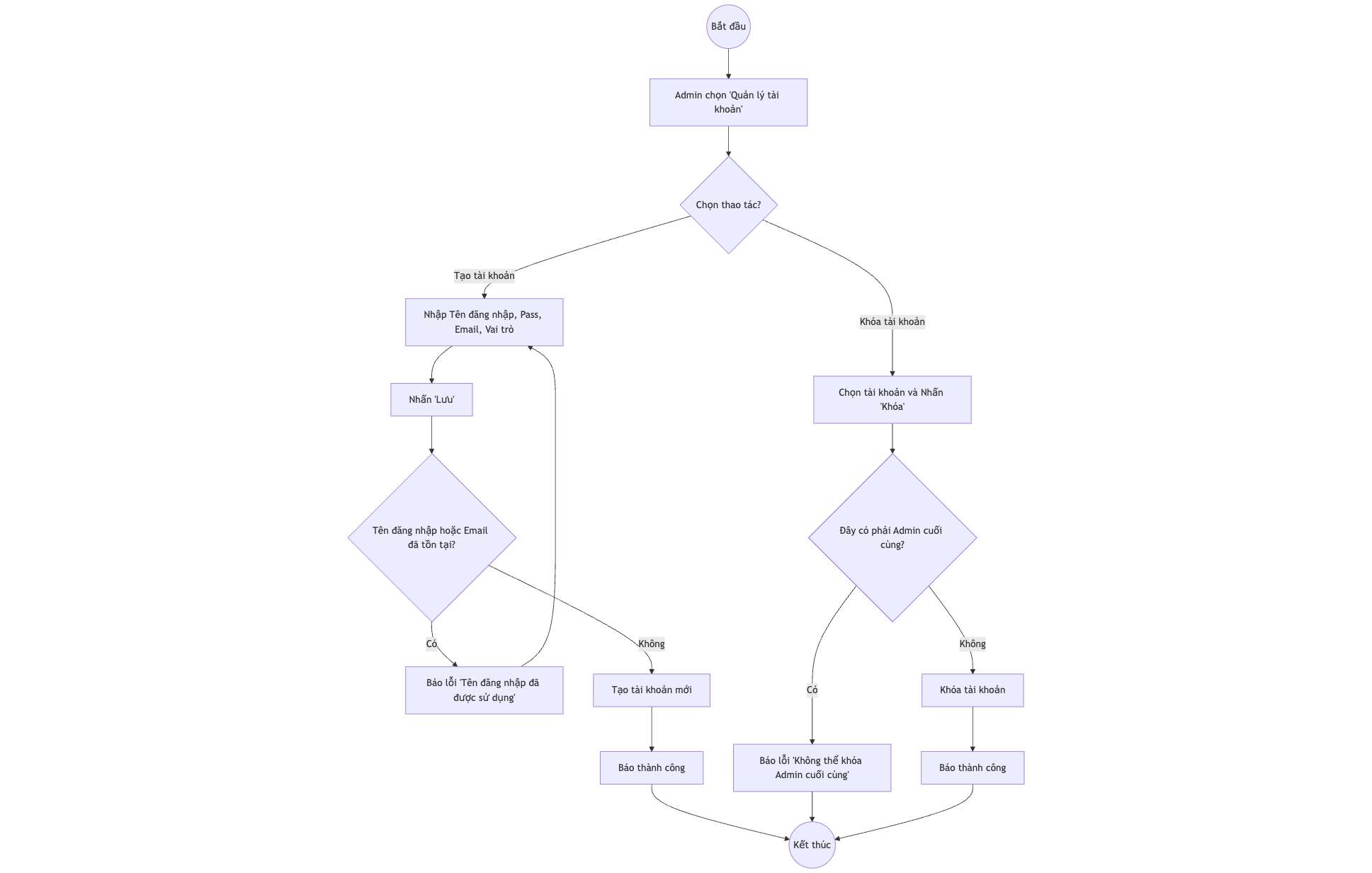
#### UC8: Xem báo cáo giảng viên



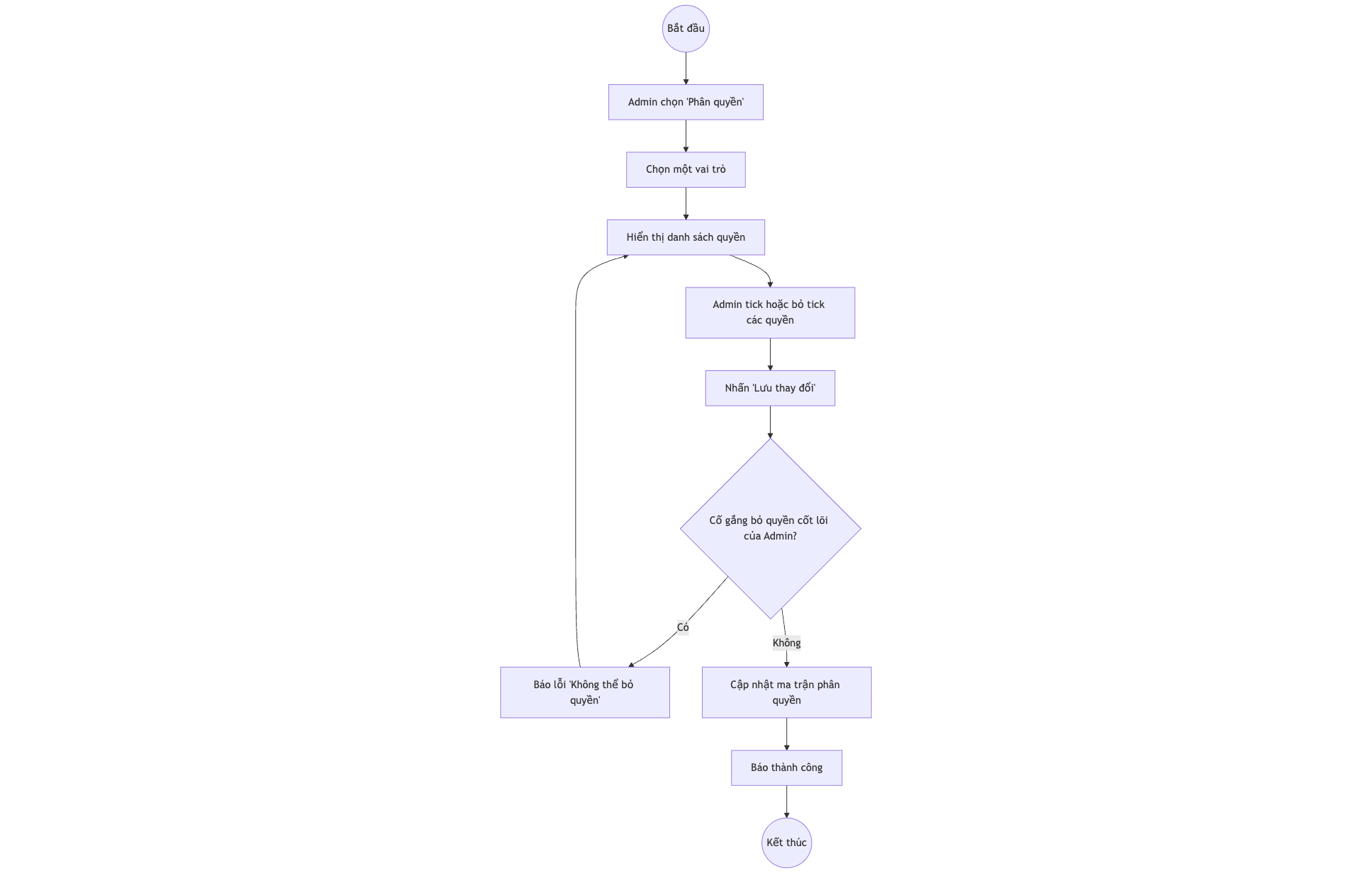
#### UC9: Quản lý thông tin giảng viên (xem/chỉnh sửa)



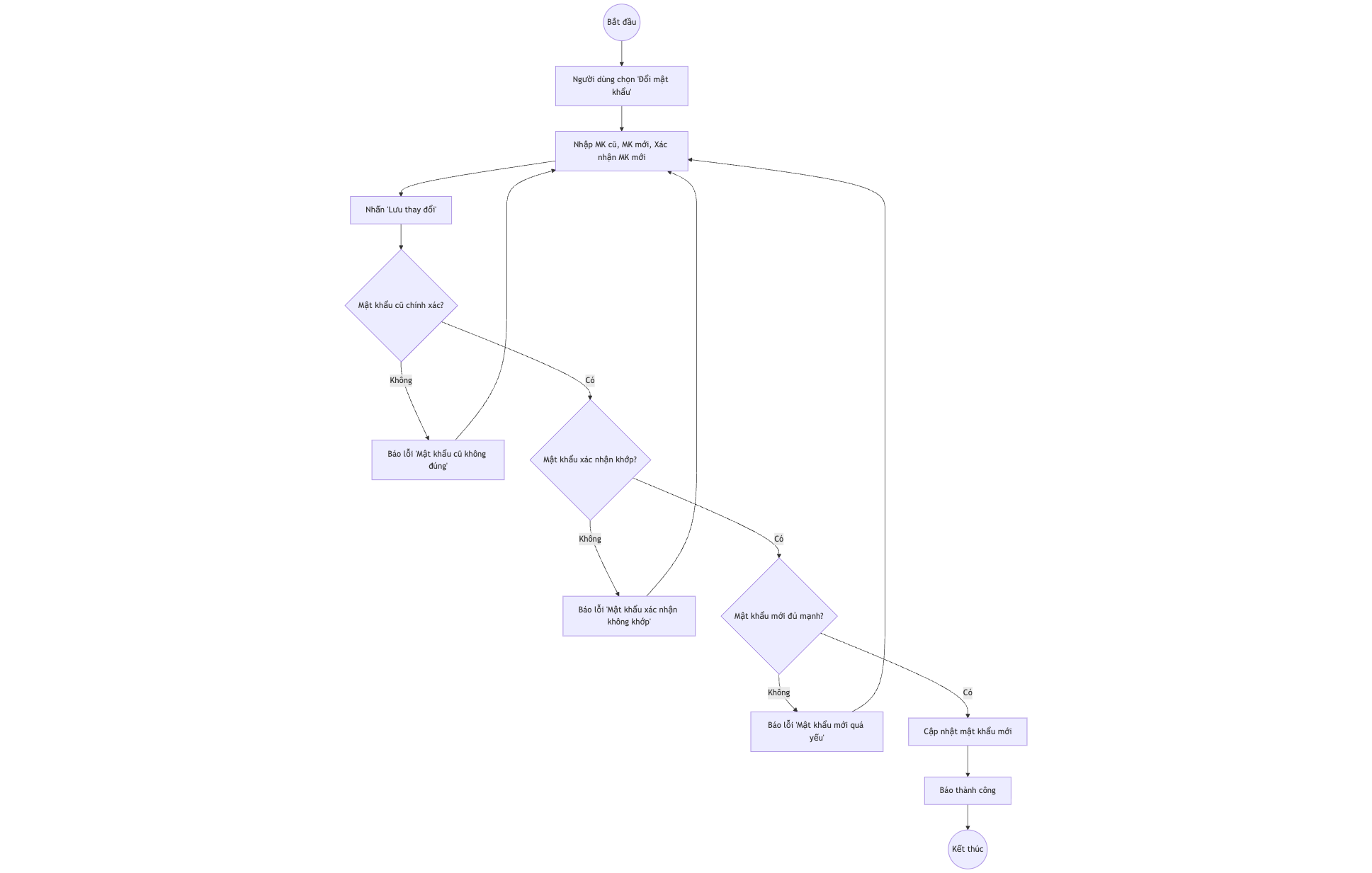
#### UC10: Quản lý tài khoản hệ thống



#### UC11: Quản lý phân quyền



#### UC12: Đổi mật khẩu



# **Y5. Kiểm thử phần mềm**

# • Viết ít nhất 3 test case (có input, output mong đợi, kết quả)

# • Lập kế hoạch kiểm thử cơ bản (unit test, kiểm thử chức năng)

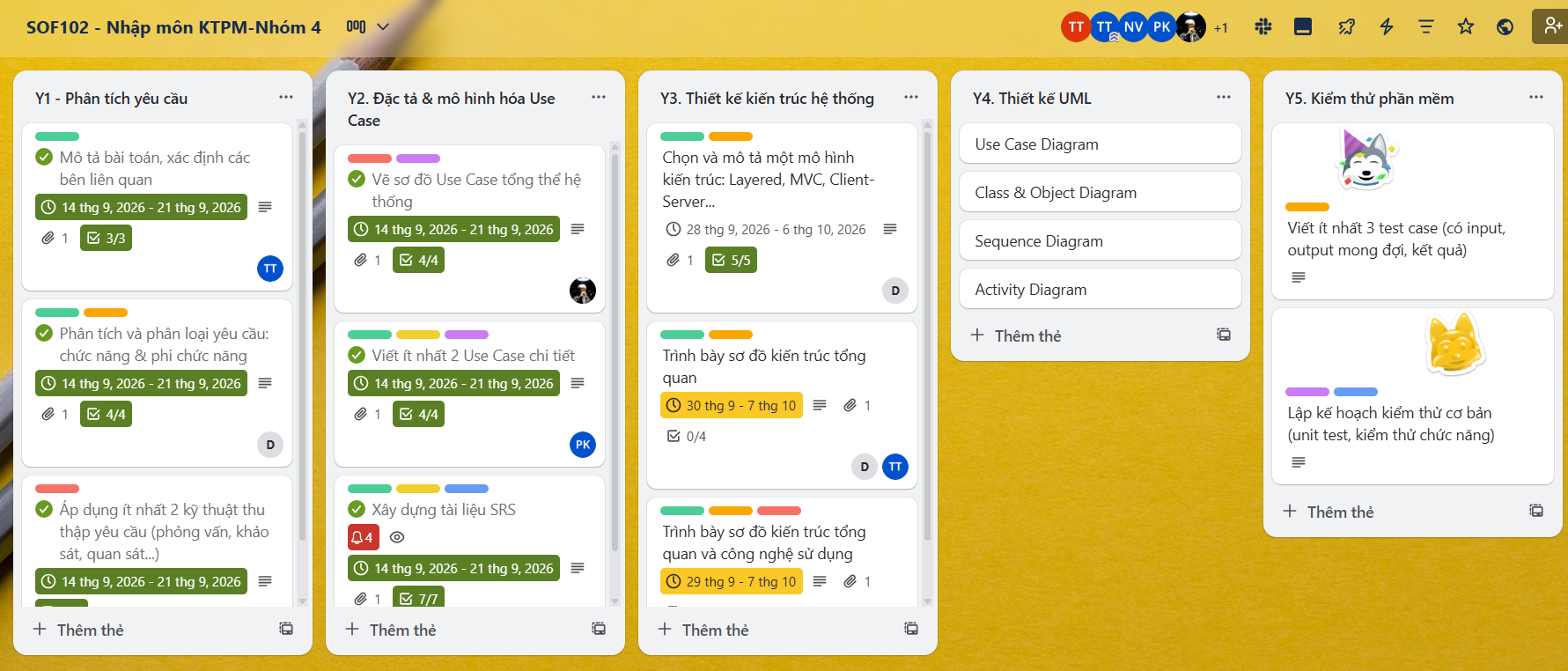
[Y5\_TestCases\_HeThongQuanLyGiangVien.xlsx](https://docs.google.com/spreadsheets/d/1mAvOy5olj9yu5-9H_SRv2nf06naE_g1m/edit?gid=646002941#gid=646002941)

# 

# **Y6. Quản lý dự án với Trello và GitHub**

1. Tạo bảng Trello quản lý công việc theo các sprint

[https://trello.com/invite/b/68e37ee32d804ba9dd9ba5fc/ATTIa0594db4580b92c91904ac284d47e8cc546BBB75/sof102-nhập-mon-ktpm-nhom-4](https://trello.com/invite/b/68e37ee32d804ba9dd9ba5fc/ATTIa0594db4580b92c91904ac284d47e8cc546BBB75/sof102-nh%E1%BA%ADp-mon-ktpm-nhom-4)



1. Sử dụng GitHub để quản lý mã nguồn hoặc tài liệu

<https://github.com/Khoa180806/Assignmment_NhapMonKyThuatPhanMem>