

Hà Nội, 2021

Giảng viên hướng dẫn: **TS. Hà Mạnh Đào**

Lớp : **KTPM2 – K13**

Nhóm 12: **Bùi Tất Trung – 2018602729**

**Trương Thị Dụ – 2018603749**

**Lê Quang Ngân – 2019608305**

**BÀI TẬP LỚN**

**MÔN: LẬP TRÌNH JAVA**

**Đề tài: *Quản lý hồ sơ học sinh***

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

------------

**MỤC LỤC**

[DANH MỤC HÌNH ẢNH 3](#_Toc72710399)

[MỞ ĐẦU 6](#_Toc72710400)

[CHƯƠNG 1: TỔNG QUAN 7](#_Toc72710401)

[1.1 Khảo sát sơ bộ 7](#_Toc72710402)

[1.2 Khảo sát chi tiết 8](#_Toc72710403)

[CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 9](#_Toc72710404)

[2.1 Các yêu cầu chức năng 9](#_Toc72710405)

[2.2 Các yêu cầu phi chức năng 9](#_Toc72710406)

[2.3 Biểu đồ use case 10](#_Toc72710407)

[2.4 Mô tả chi tiết và phân tích các use case 10](#_Toc72710408)

[CHƯƠNG 3: THỰC NGHIỆM VÀ ĐÁNH GIÁ 40](#_Toc72710409)

[3.1 Thực nghiệm chương trình 40](#_Toc72710410)

[3.2 Đánh giá 61](#_Toc72710411)

[TÀI LIỆU THAM KHẢO 63](#_Toc72710412)

# DANH MỤC HÌNH ẢNH

[Hình 2.1. Biểu đồ use case tổng quát 10](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710413)

[Hình 2.2. Biểu đồ trình tự use case “Đăng nhập” 12](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710414)

[Hình 2.3. Giao diện “Đăng nhập” 14](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710415)

[Hình 2.4. Biểu đồ trình tự use case “Quản lý người dùng” 18](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710416)

[Hình 2.5. Biểu đồ lớp phân tích use case “Quản lý người dùng” 18](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710417)

[Hình 2.6. Giao diện “Quản lý người dùng” 19](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710418)

[Hình 2.7. Biểu đồ trình tự use case “Quản lý khóa học” 23](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710419)

[Hình 2.8. Biểu đồ lớp phân tích use case “Quản lý khóa học” 23](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710420)

[Hình 2.9. Giao diện “Quản lý khóa học” 24](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710421)

[Hình 2.10. Biểu đồ trình tự use case “Quản lý kỳ học” 28](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710422)

[Hình 2.11. Biểu đồ lớp phân tích use case “Quản lý kỳ học” 28](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710423)

[Hình 2.12. Giao diện “Quản lý kỳ học” 29](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710424)

[Hình 2.13. Biểu đồ trình tự use case “Quản lý lớp học” 33](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710425)

[Hình 2.14. Biểu đồ lớp phân tích use case “Quản lý lớp học” 33](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710426)

[Hình 2.15. Giao diện “Quản lý lớp học” 34](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710427)

[Hình 2.16. Biểu đồ trình tự use case “Quản lý học sinh” 38](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710428)

[Hình 2.17. Biểu đồ lớp use case “Quản lý học sinh” 38](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710429)

[Hình 2.18. Giao diện “Quản lý học sinh” 39](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710430)

[Hình 3.1. Màn hình Form đăng nhập 40](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710431)

[Hình 3.2. Màn hình chức năng 40](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710432)

[Hình 3.3. Màn hình quản lý người dùng 41](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710433)

[Hình 3.4. Màn hình thêm người dùng 41](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710434)

[Hình 3.5. Màn hình kết quả dữ liệu sau khi cập nhật 42](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710435)

[Hình 3.6. Màn hình sửa tài khoản người dùng 42](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710436)

[Hình 3.7. Màn hình kết quả sửa tài khoản người dùng 43](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710437)

[Hình 3.8. Màn hình xóa tài khoản người dùng 43](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710438)

[Hình 3.9. Màn hình kết quả xóa tài khoản người dùng 44](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710439)

[Hình 3.10. Màn hình tìm kiếm tài khoản người dùng 44](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710440)

[Hình 3.11. Màn hình kết quả tìm kiếm tài khoản người dùng 45](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710441)

[Hình 3.12. Màn hình thêm khóa học 45](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710442)

[Hình 3.13. Màn hình kết quả thêm khóa học 46](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710443)

[Hình 3.14. Màn hình sửa thông tin khóa học 46](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710444)

[Hình 3.15. Màn hình kết quả sửa thông tin khóa học 47](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710445)

[Hình 3.16. Màn hình xóa thông tin khóa học 47](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710446)

[Hình 3.17. Màn hình kết quả xóa thông tin khóa học 48](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710447)

[Hình 3.18. Màn hình tìm kiếm thông tin khóa học 48](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710448)

[Hình 3.19. Màn hình kết quả tìm kiếm thông tin khóa học 49](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710449)

[Hình 3.20. Màn hình thêm mới kỳ học 49](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710450)

[Hình 3.21. Màn hình kết quả thêm mới kỳ học 50](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710451)

[Hình 3.22. Màn hình sửa thông tin kỳ học 50](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710452)

[Hình 3.23. Màn hình kết quả sửa thông tin kỳ học 51](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710453)

[Hình 3.24. Màn hình Xóa thông tin kỳ học 51](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710454)

[Hình 3.25. Màn hình kết quả sau khi Xóa thông tin kỳ học 52](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710455)

[Hình 3.26. Màn hình tìm kiếm thông tin kỳ học 52](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710456)

[Hình 3.27. Màn hình kết quả tìm kiếm thông tin kỳ học 53](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710457)

[Hình 3.28. Màn hình thêm mới lớp học 53](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710458)

[Hình 3.29. Màn hình kết quả thêm mơi lớp học 54](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710459)

[Hình 3.30. Màn hình sửa lớp học 54](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710460)

[Hình 3.31. Màn hình kết quả sau khi sửa lớp học 55](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710461)

[Hình 3.32. Màn hình xóa lớp học 55](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710462)

[Hình 3.33. Màn hình kết quả sau khi xóa lớp học 56](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710463)

[Hình 3.34. Màn hình tìm kiếm lớp học 56](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710464)

[Hình 3.35. Màn hình kết quả sau khi tìm kiếm lớp học 57](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710465)

[Hình 3.36. Màn hình thêm mới học sinh 57](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710466)

[Hình 3.37. Màn hình kết quả sau thêm mới học sinh 58](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710467)

[Hình 3.38. Màn hình sửa học sinh 58](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710468)

[Hình 3.39. Màn hình kết quả sau khi sửa thông tin học sinh 59](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710469)

[Hình 3.40. Màn hình Xóa học sinh 59](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710470)

[Hình 3.41. Màn hình kết quả sau khi xóa thông tin học sinh 60](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710471)

[Hình 3.42. Màn hình tìm kiếm thông tin học sinh 60](file:///D:\Study\Java\Nhom12_BaoCao.docx#_Toc72710472)

[Hình 3.43. Màn hình kết quả sau khi Tìm kiếm thông tin học sinh 61](#_Toc72710473)

# MỞ ĐẦU

Hiện nay, sự phát triển của công nghệ thông tin là một trong những lĩnh vực được phát triển hàng đầu với việc triển khai rộng rãi các ứng dụng cho các ngành nghề, tổ chức và xã hội. Cùng với đó là sự phát triển của phần mềm, phần mềm giúp tăng năng suất công việc, dễ dàng quản lý và lưu trữ thông tin.

Ví dụ như để quản lý hồ sơ học sinh thì người quản lý cần liên tục cập nhật các thông tin liên quan về học sinh như: họ và tên, tuổi, quê quán, năm sinh …., Công việc đó người quản lý rất vất vả. Với phần mềm “Quản lý hồ sơ học sinh” mà nhóm chúng em đã nghiên cứu và xây dựng trong bài tập sau đây sẽ là phần nào giúp cho người quản lý làm việc dễ dàng hơn. Vì sự hiểu biết và kinh nghiệm còn hạn chế nên chắc chắn phần mềm của chúng em không tránh khỏi những thiếu sót. Rất mong nhận được ý kiến đóng góp của thầy.

Chúng em xin chân thành cảm ơn !

**Những ưu điểm khi sử dụng phần mềm:**

* Tiết kiệm thời gian, công sức.
* Lưu trữ thông tin an toàn.
* Dễ dàng quản lý, cập nhật thông tin học sinh.

**Một số nhược điểm:**

* Dễ bị tin tặc tấn công.
* Dễ mất dữ liệu.

**Hoạt động của phần mềm:**

Phần mềm được viết ra với mục đích giúp người giáo viên quản lý được thông tin học sinh dễ dàng hơn. Giáo viên có thể thêm, sửa, xóa, cập nhật thông tin học sinh, lớp học, khóa học và kỳ học. Ngoài ra phần mềm còn giúp giáo viên tìm kiếm học sinh theo năm sinh và quê quán nhanh chóng.

**Các công cụ hỗ trợ:**

* Netbeans 8.2, MySQL, Rational rose.

**Kết quả đạt được:**

Xây dựng được các chức năng cơ bản bao gồm: thêm, sửa, xóa, cập nhật, tìm kiếm để quản lý thông tin về lớp, khóa học, kỳ học và thông tin cá nhân của mỗi học sinh.

# TỔNG QUAN

## Khảo sát sơ bộ

### Mục tiêu

Mục tiêu xây dựng phần mềm quản lý hồ sơ học sinh, giúp cho người quản lý dễ dàng nắm bắt được thông tin của học sinh đồng thời có thể thêm, sửa, xóa, cập nhật thông tin học sinh theo yêu cầu.

### Phương pháp

* Điều tra:

Các câu hỏi điều tra ví dụ như:

* Cô giáo nghĩ như nào nếu có một phần mềm làm cho việc quản lý hồ sơ học sinh đơn giản, hiệu quả cao hơn?
* Nếu có phần mềm như vậy để quản lý hồ sơ học sinh thì Thầy/Cô giáo muốn nó có những chức năng gì?
* Những chức năng nào của phần mềm Thầy/Cô giáo chú ý và muốn nó được hoàn thiện nhiều nhất có thể?

### Đối tượng khảo sát

Đối tượng phỏng vấn: là các người quản lý hồ sơ học sinh có nhu cầu có một phần mềm hỗ trợ quản lý học sinh như Thầy/Cô giáo.

Đối tượng học hỏi, quan sát: Các hệ thống khác có mục tiêu tương tự đã có trên mạng.

### Kết quả sơ bộ

Biết được nhu cầu cơ bản của người được phỏng vấn và giao diện, các chức năng cơ bản của phần mềm sẽ xây dựng.

### Các tài liệu thu được

Thu thập được tài liệu hỗ trợ việc xây dựng phần mềm.

## Khảo sát chi tiết

Hệ thống phần mềm cho phép người sử dụng thực hiện các chức năng đăng nhập vào hệ thống; thêm, sửa, xóa các thông tin, tìm kiếm quản lý hồ sơ học sinh.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Các yêu cầu chức năng

* Đăng nhập vào hệ thống
* Quản lý người dùng
* Quản lý khóa học
* Quản lý kỳ học
* Quản lý lớp học
* Quản lý học sinh

## Các yêu cầu phi chức năng

* Hiệu năng hoạt động: Hiệu năng liên quan đến tài nguyên được sử dụng trong các điều kiện nhất định.
* Tính tương thích: Thực thi các chức năng cần thiết của người dùng.
* Tính khả dụng: Dễ sử dụng, Giao diện bố cục rõ ràng, khả năng truy cập nhanh.
* Tính tin cậy: Thực hiện các chức năng nhanh chóng và chính xác.
* An toàn thông tin: Mức độ hệ thống có thể bảo vệ thông tin, dữ liệu và đảm bảo các cá nhân, hệ thống khác có thể truy cập dữ liệu trong phạm vi được cho phép.

## Biểu đồ use case

Hình 2.1. Biểu đồ use case tổng quát

## Mô tả chi tiết và phân tích các use case

### Use case Đăng Nhập.

**a) Mô tả use case**

* **Tên use case**

Đăng Nhập.

* **Mô tả chi tiết use case:**

Use case này cho phép người dùng đăng nhập vào hệ thống.

* **Luồng sự kiện:**
* Luồng cơ bản:
  + 1. Use case bắt đầu khi người dùng chạy chương trình. Hệ thống hiển thị màn hình đăng nhập.
    2. Người dùng nhập tên đăng nhập, mật khẩu và chọn nút “Đăng nhập”. Hệ thống kiểm tra thông tin đăng nhập, nếu hợp lệ cho phép người dùng truy cập vào hệ thống.

Use case kết thúc.

* Luồng rẽ nhánh:

1. Tại bước 2 luồng cơ bản, nếu người dùng nhập thông tin không hợp lệ. Hệ thống hiển thị thông báo yêu cầu người dùng nhập lại.
2. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use kết thúc.

* **Các yêu cầu đặc biệt:**

Không có.

* **Tiền điều kiện:**

Không có.

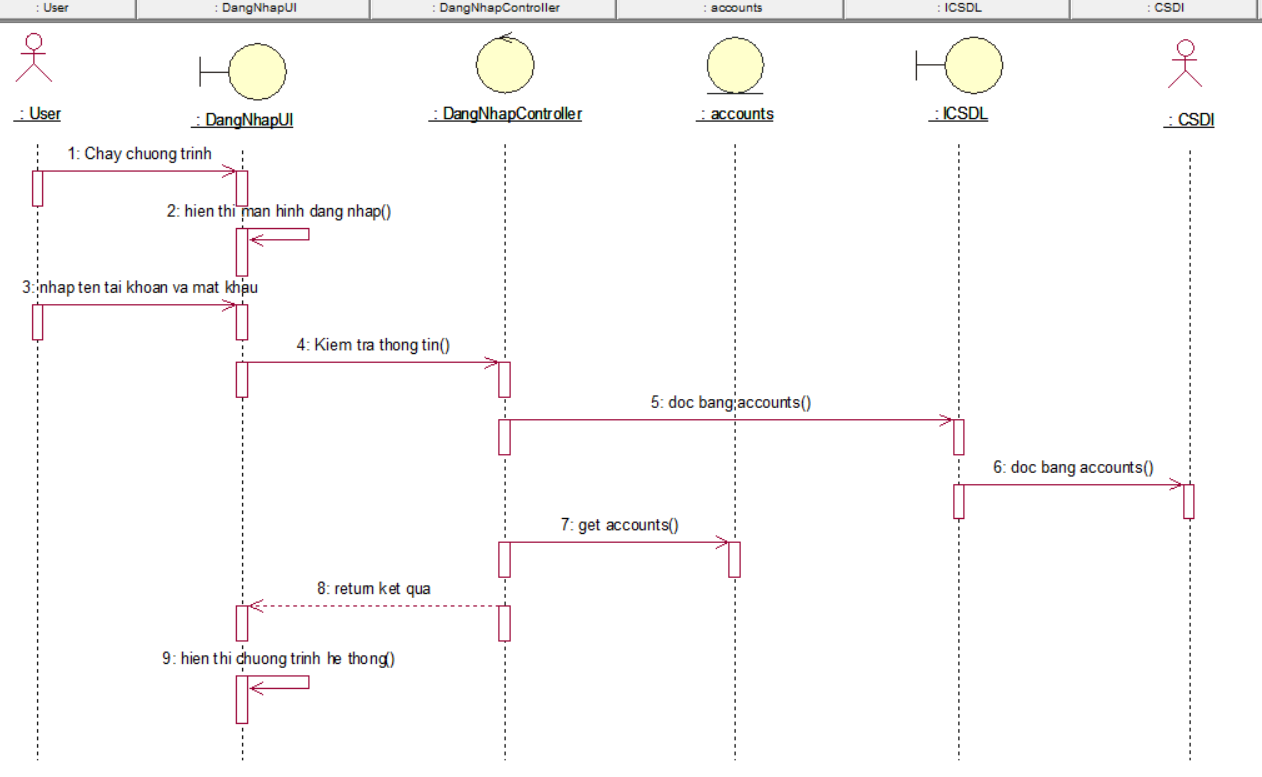
* **Hậu điều kiện:**

Không có.

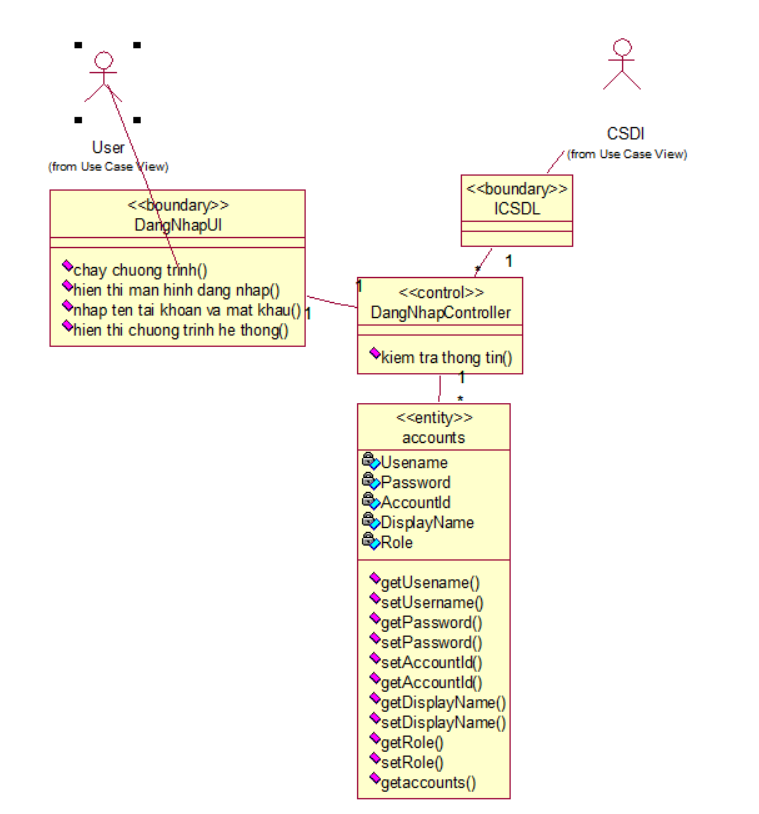
* **Điểm mở rộng:**

Không có.

**b) Phân tích use case**

* **Biểu đồ trình tự**

Hình 2.2. Biểu đồ trình tự use case “Đăng nhập”

* **Biểu đồ lớp phân tích**

**c) Giao diện**

Hình 2.3. Giao diện “Đăng nhập”

### Use case Quản lý người dùng

**a) Mô tả use case**

* **Tên use case**

Quản lý người dùng

* **Mô tả chi tiết use case:**

Use case này cho phép người quản trị xem, thêm, sửa và xóa người dùng trong bảng accounts.

* **Luồng sự kiện:**
* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng chọn vào nút “Quản lý người dùng” trên màn hình chức năng của chương trình. Hệ thống lấy thông tin chi tiết của người dùng (accountId, Username, DisplayName, Role ) từ bảng accounts trong cơ sở dữ liệu và hiển thị danh sách người dùng lên màn hình.

*2. Thêm người dùng:*

1. Người dùng chọn vào nút “thêm mới” trên màn hình quản lý sách và nhập thông tin người dùng mới và chọn nút “Thêm”. Hệ thống cập nhật thông tin sách trong bảng accounts và hiển thị màn hình thông tin danh sách sách đã cập nhật.

*3. Sửa thông tin người dùng:*

1. Người dùng chọn AccountId trên bảng hiên thị cần sửa. Hệ thống hiển thị thông tin người dùng cần sửa vào form sửa.

2. Người dùng nhập thông tin sửa và chọn nút “Sửa”. Hệ thống cập nhật thông tin người dùng trong bảng accounts và hiển thị màn hình thông tin danh sách đã cập nhật.

4. *Xóa người dùng:*

1. Người dùng chọn tài khoản cần xóa. Hệ thống hiển thị thông tin tài khoản cần xóa vào form sửa.

2. Người dùng chọn nút “Xóa” sẽ hiển thị 1 popup thông báo Xác nhận xóa,sau khi click nut “OK” Hệ thống cập nhật thông tin sách trong bảng accounts và hiển thị màn hình thông tin danh sách người dùng.

1. Tìm kiếm người dùng:

1. Người dùng nhập tên tài khoản cần tìm kiếm, bấm Enter. Hệ thống sẽ hiển thị thông tin tài khoản cần tìm lên bảng.

Use case kết thúc.

* Luồng rẽ nhánh:

1. Tại bước 2 và 3 trong luồng cơ bản nếu người dùng nhập thông tin tài khoản không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi.

2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không có.

* **Tiền điều kiện:**

Người dùng cần đăng nhập hệ thống khi có thể thực hiện use case.

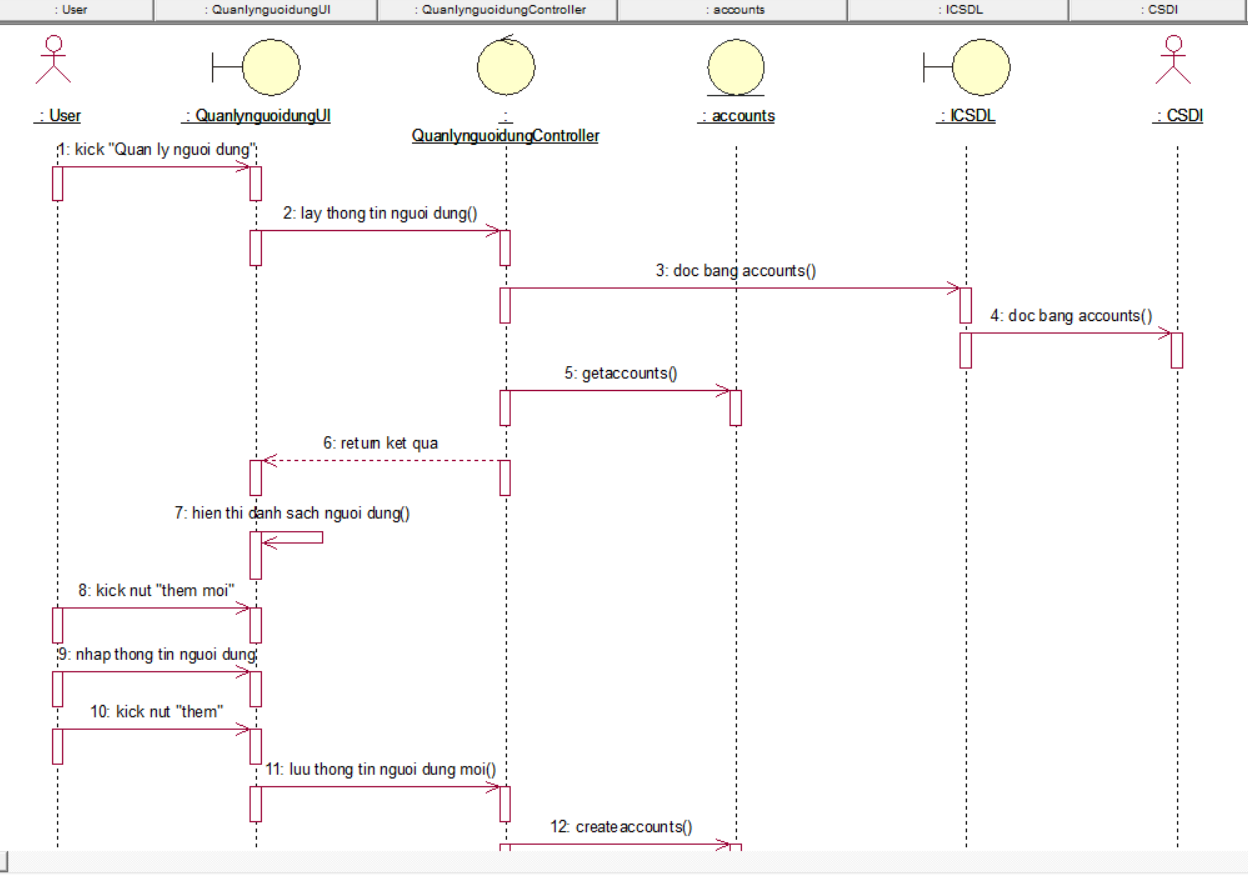
* **Hậu điều kiện:**

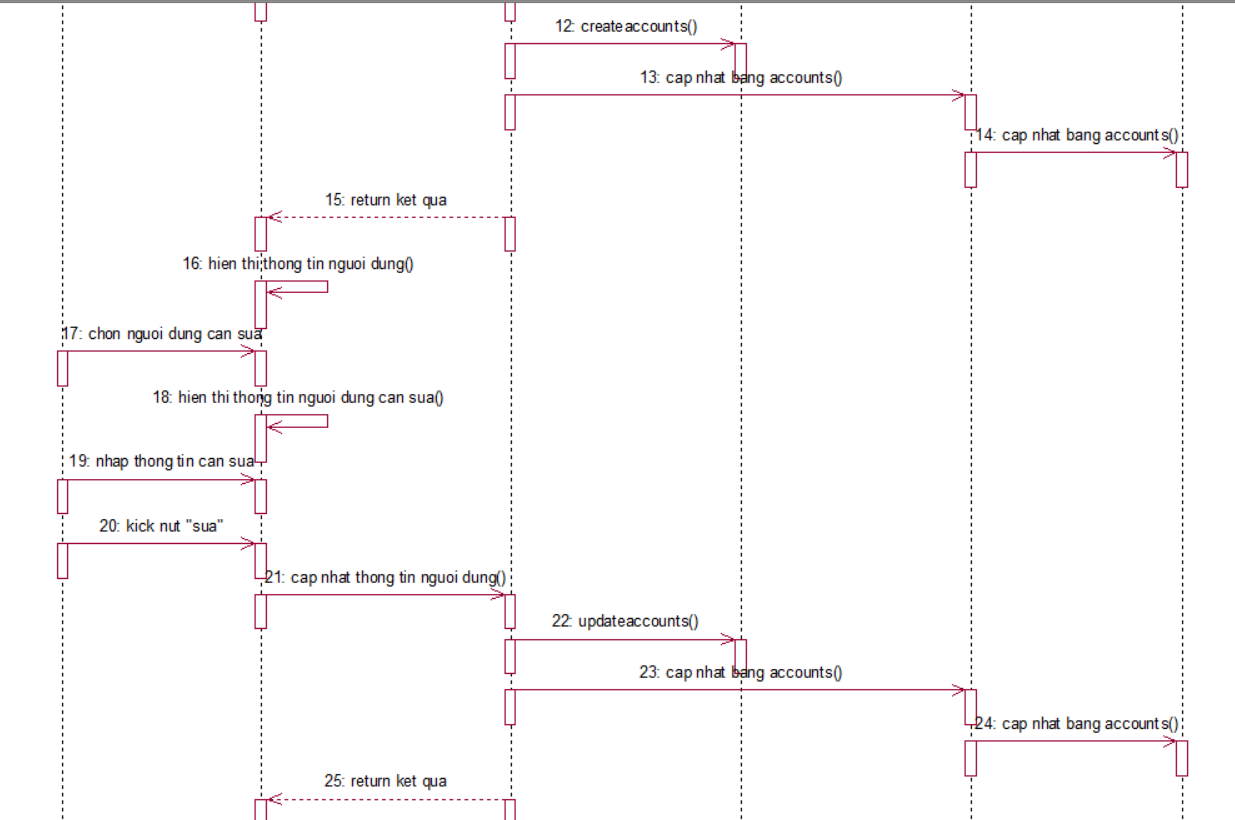
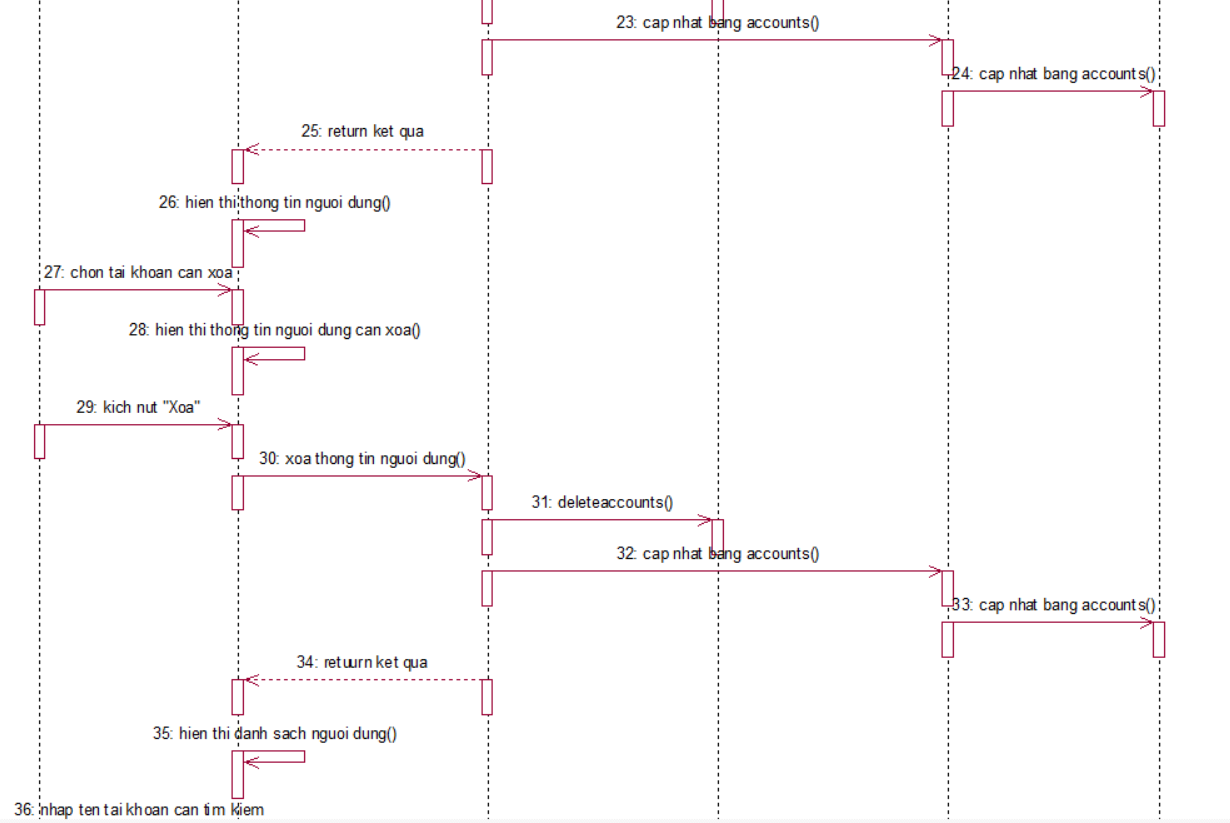
Nếu use case kết thúc thành công thì thông tin về người dùng sẽ được cập nhập trong cơ sở dữ liệu.

* **Điểm mở rộng:**

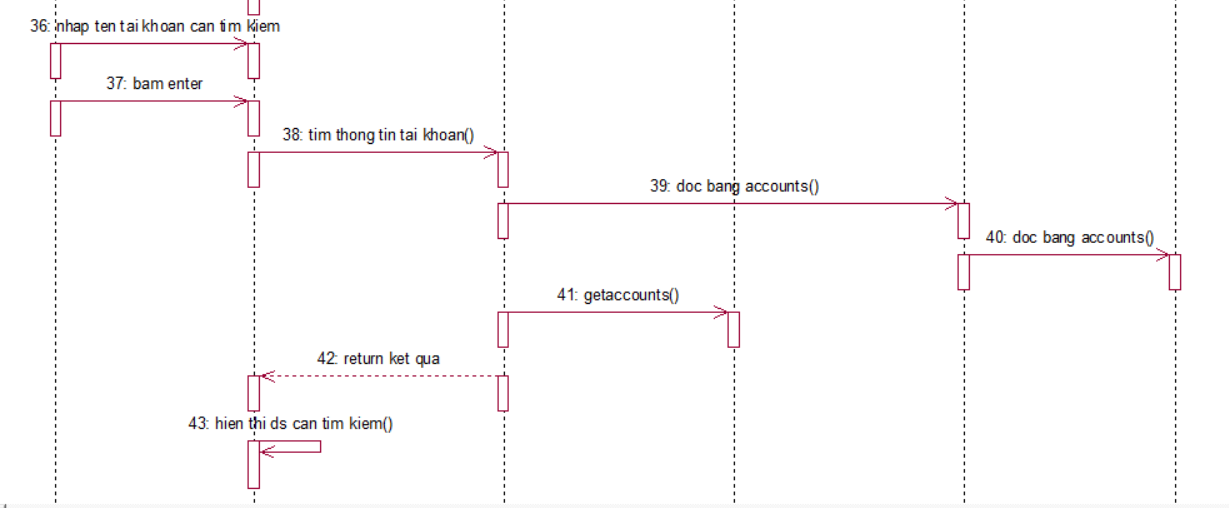
Không có.

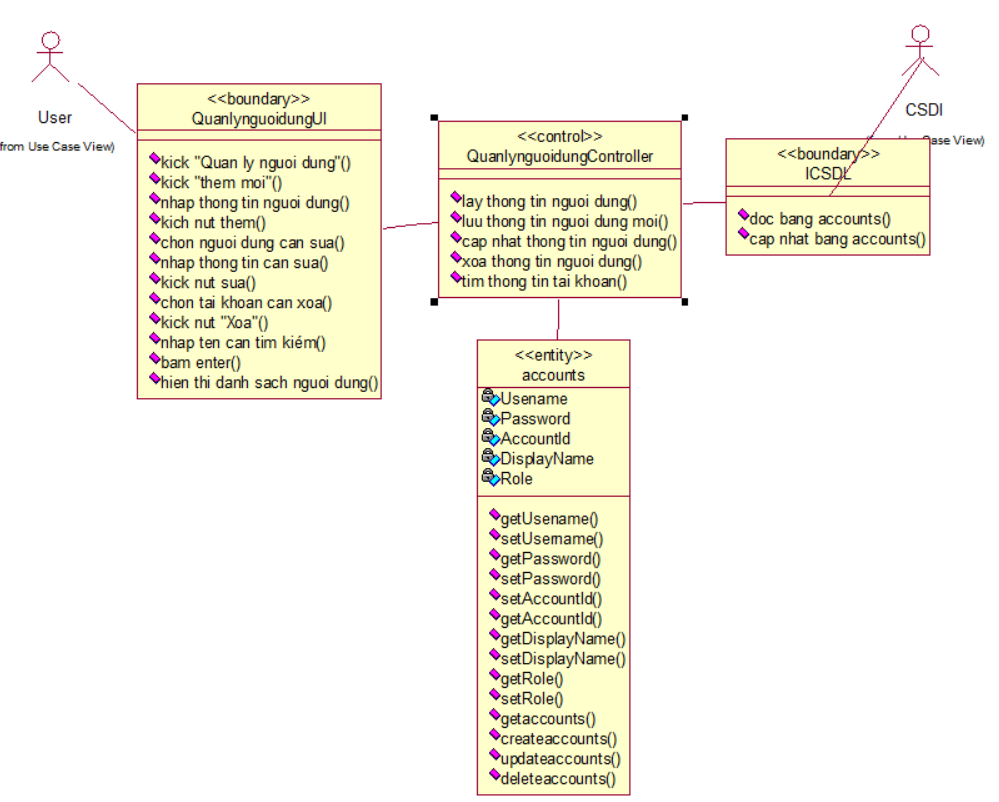
**b) Phân tích use case**

* **Biểu đồ trình tự**

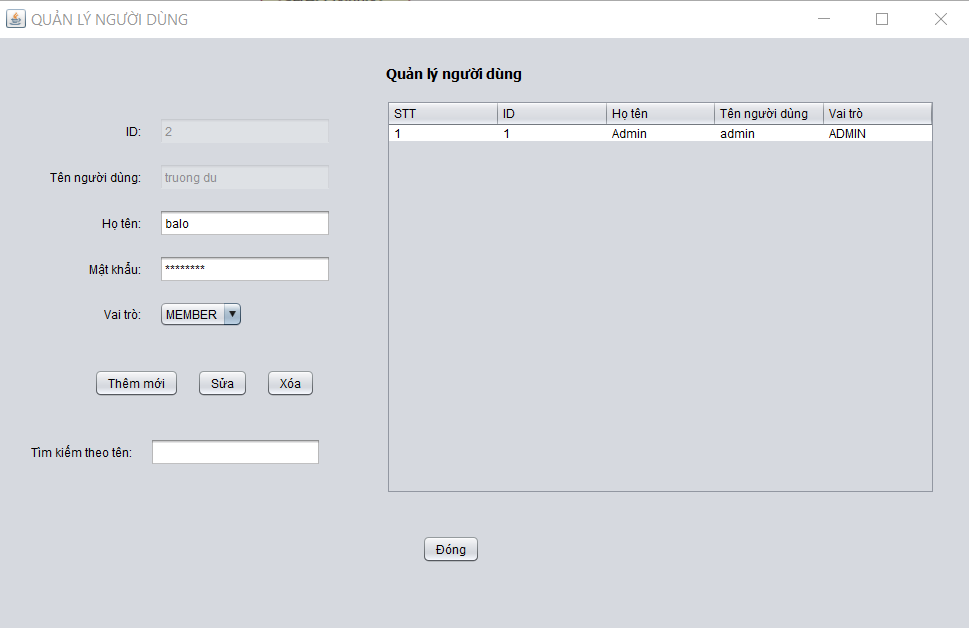


Hình 2.4. Biểu đồ trình tự use case “Quản lý người dùng”



* **Biểu đồ lớp phân tích**

Hình 2.5. Biểu đồ lớp phân tích use case “Quản lý người dùng”

**c) Giao diện**

Hình 2.6. Giao diện “Quản lý người dùng”

### Use case Quản lý khóa học

**a) Mô tả use case**

* **Tên use case**

Quản lý khóa học

* **Mô tả chi tiết use case:**

Use case này cho phép người quản trị xem, thêm, sửa và xóa khóa học trong bảng courses.

* **Luồng sự kiện:**
* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng chọn vào nút “Quản lý khóa học” trên màn hình chức năng của chương trình. Hệ thống lấy thông tin chi tiết của người dùng (CourseId, Coursename, StartTime, EndTime ) từ bảng courses trong cơ sở dữ liệu và hiển thị danh sách khóa học lên màn hình.

*2. Thêm khóa học:*

1. Người dùng chọn vào nút “thêm mới” trên màn hình quản lý khóa học và nhập thông tin khóa học mới và chọn nút “Thêm”. Hệ thống cập nhật thông tin khóa học trong bảng courses và hiển thị màn hình thông tin danh sách khóa học đã cập nhật.

*3. Sửa thông tin khóa học:*

1. Người dùng chọn khóa học trên bảng hiên thị cần sửa. Hệ thống hiển thị thông tin khóa học cần sửa vào form sửa.

2. Người dùng nhập thông tin sửa và chọn nút “Sửa”. Hệ thống cập nhật thông tin khóa học trong bảng courses và hiển thị màn hình thông tin danh sách đã cập nhật.

4. *Xóa khóa học:*

1. Người dùng chọn khóa học cần xóa. Hệ thống hiển thị thông tin khóa học cần xóa vào form sửa.

2. Người dùng chọn nút “Xóa” sẽ hiển thị 1 popup thông báo Xác nhận xóa,sau khi click nut “Yes” Hệ thống cập nhật thông tin khóa học trong bảng courses và hiển thị màn hình thông tin danh sách khóa học.

5.Tìm kiếm khóa học:

1. Người dùng nhập tên khóa học cần tìm kiếm, bấm Enter. Hệ thống sẽ hiển thị thông tin khóa học cần tìm lên bảng.

Use case kết thúc.

* Luồng rẽ nhánh:

1. Tại bước 2 và 3 trong luồng cơ bản nếu người dùng nhập thông tin khóa học không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi.

2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không có.

* **Tiền điều kiện:**

Người dùng cần đăng nhập hệ thống khi có thể thực hiện use case.

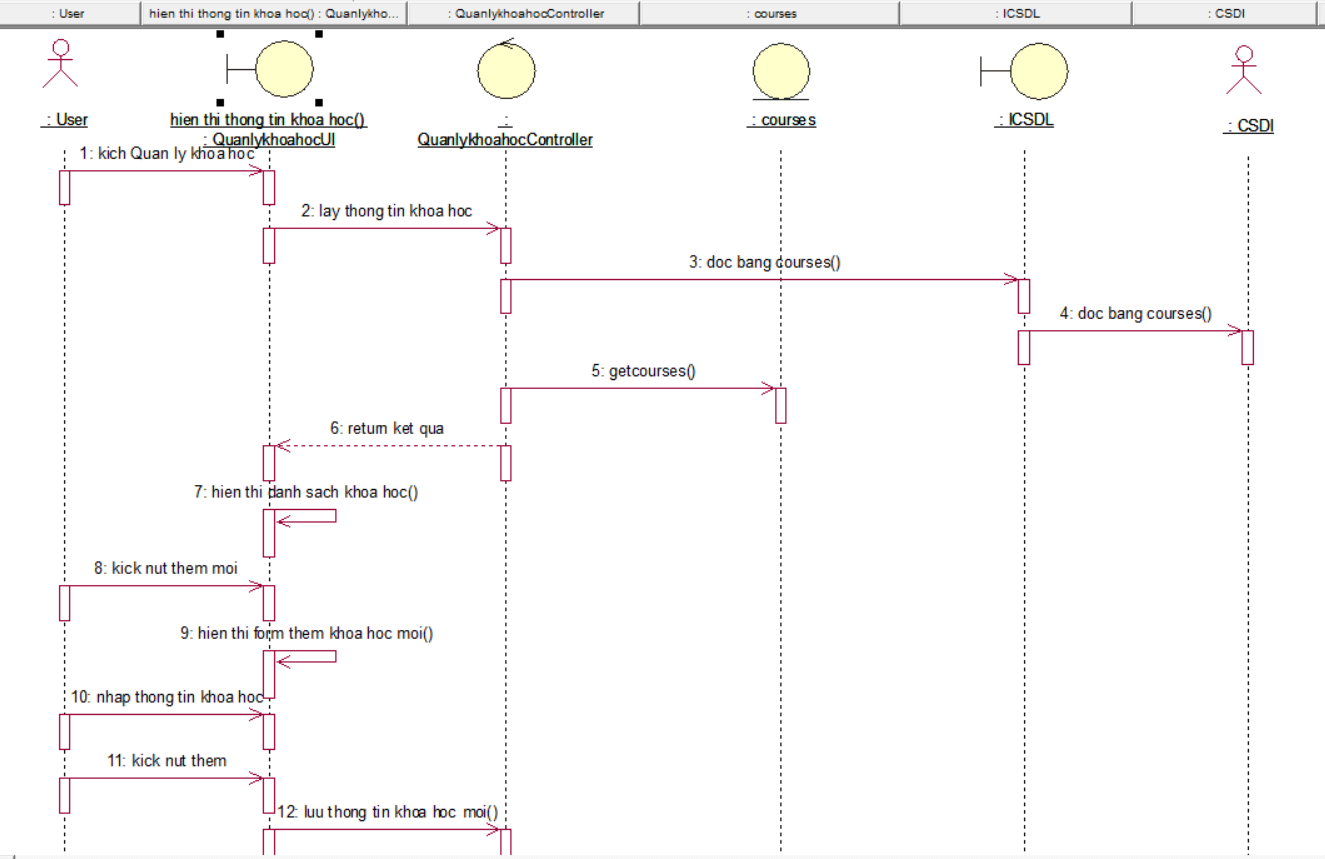
* **Hậu điều kiện:**

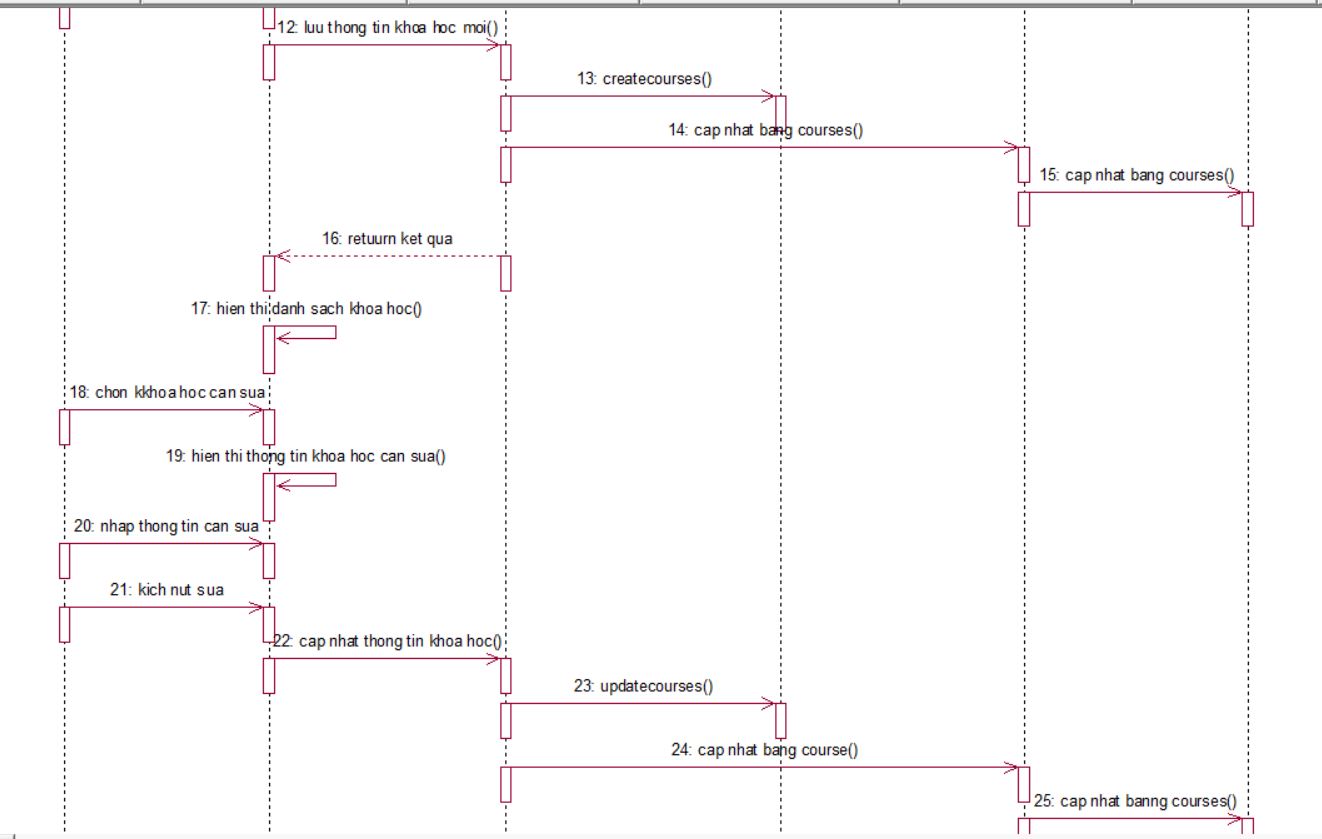
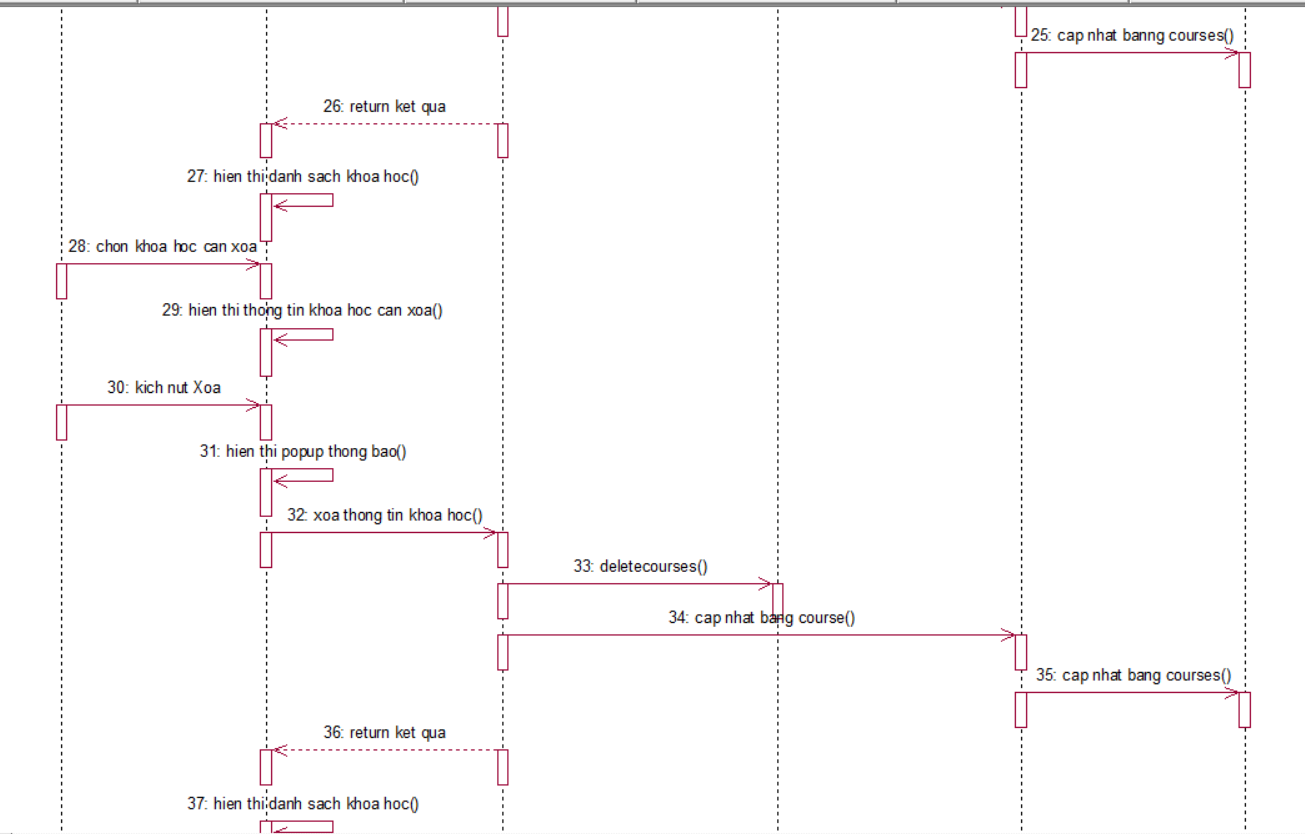
Nếu use case kết thúc thành công thì thông tin về khóa học sẽ được cập nhập trong cơ sở dữ liệu.

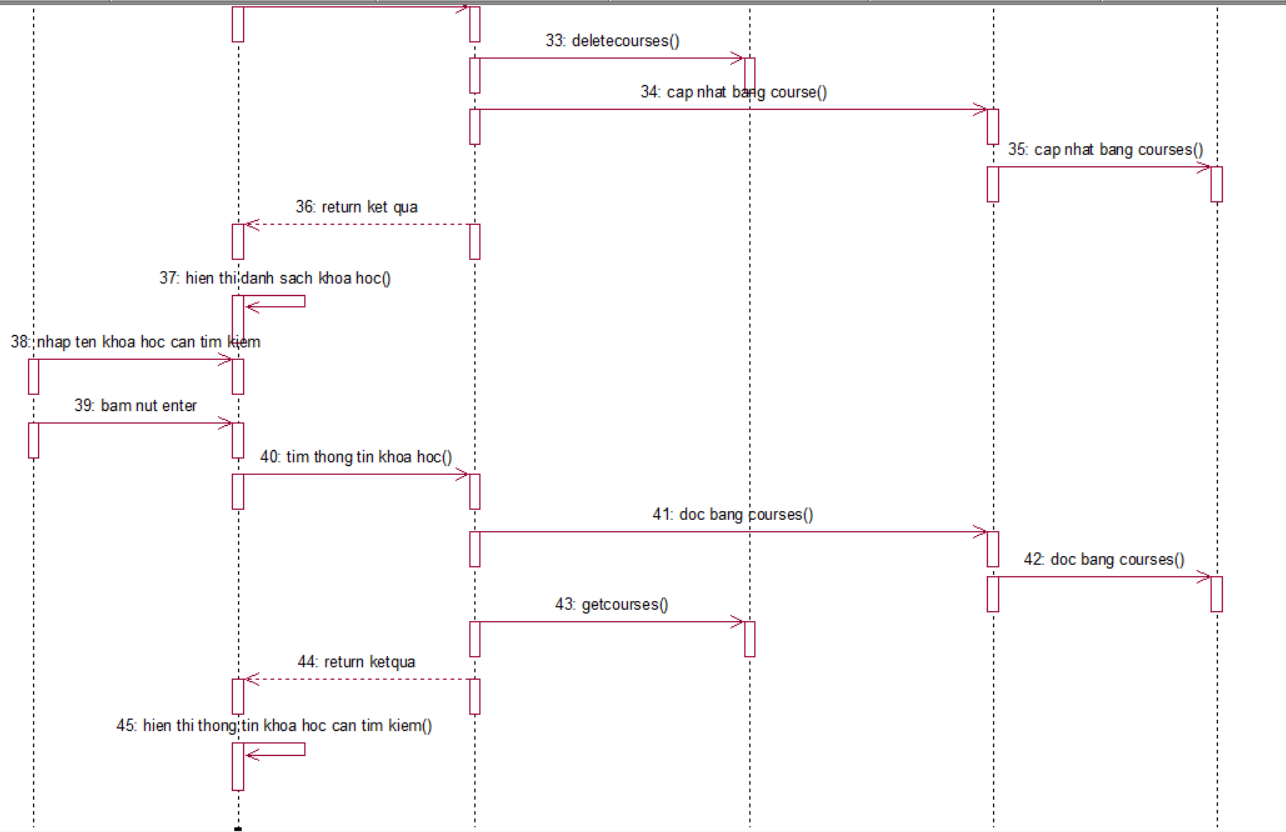
* **Điểm mở rộng:**

Không có.

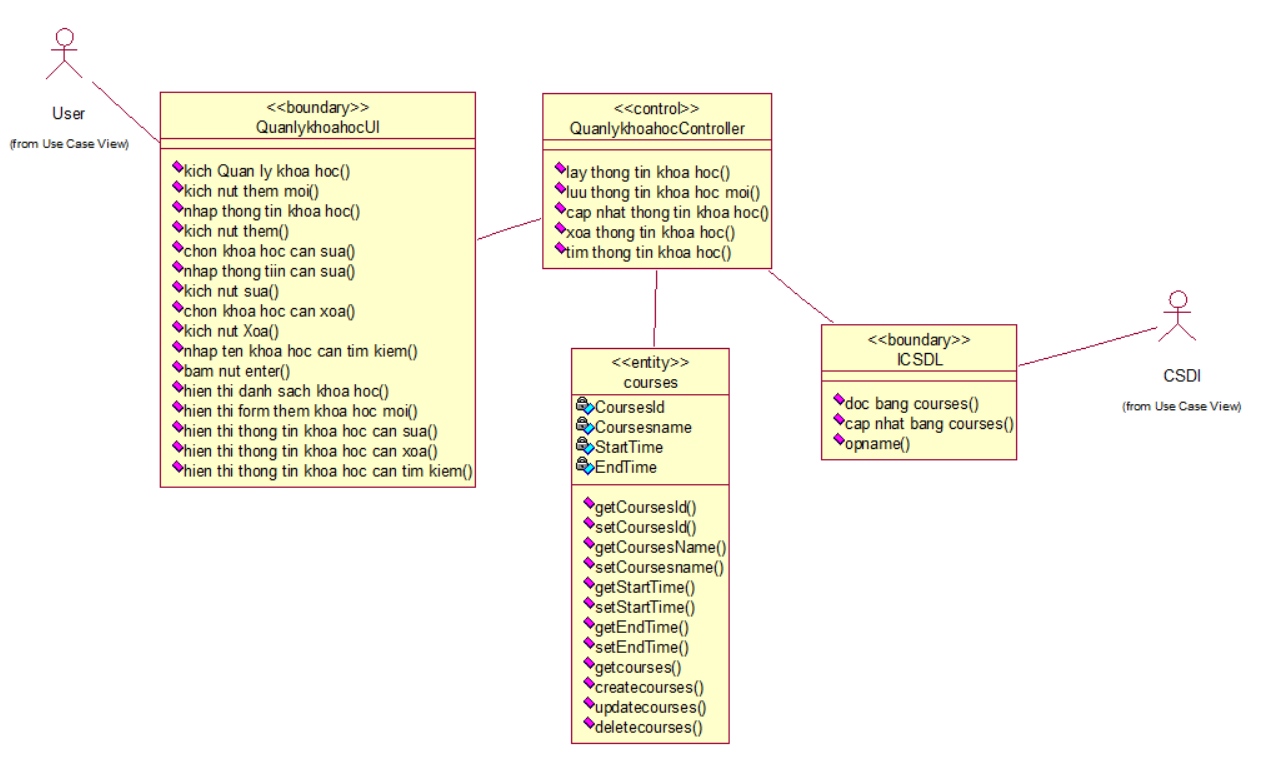
**b) Phân tích use case**

* **Biểu đồ trình tự**

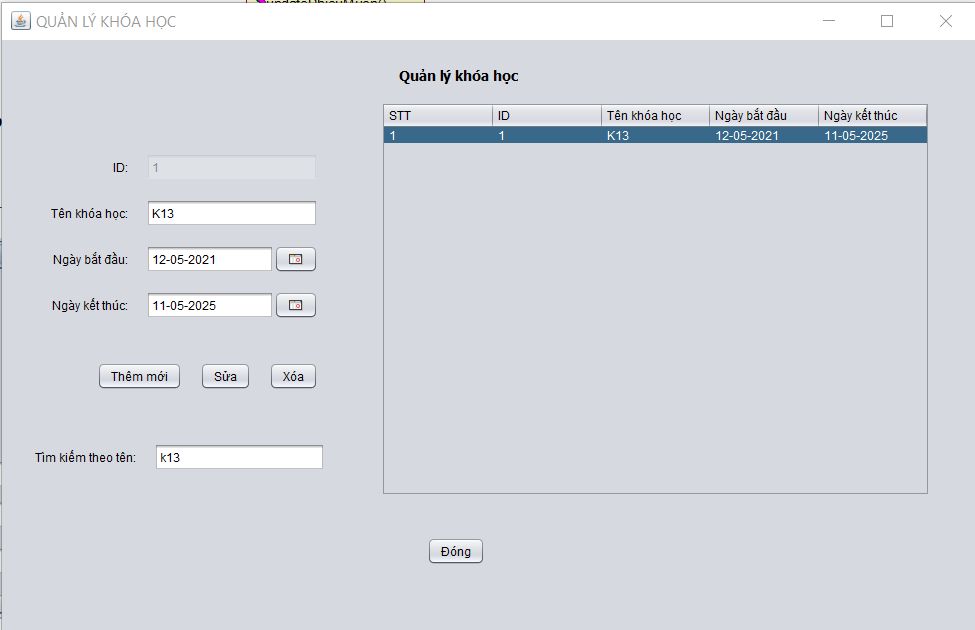




Hình 2.7. Biểu đồ trình tự use case “Quản lý khóa học”

* **Biểu đồ lớp phân tích**

Hình 2.8. Biểu đồ lớp phân tích use case “Quản lý khóa học”

**c) Giao diện**

Hình 2.9. Giao diện “Quản lý khóa học”

### Use case Quản lý kỳ học

**a) Mô tả use case**

* **Tên use case**

Quản lý kỳ học

* **Mô tả chi tiết use case:**

Use case này cho phép người dùng xem, thêm, sửa và xóa kỳ học trong bảng semesters.

* **Luồng sự kiện:**
* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng chọn vào nút “Quản lý kỳ học” trên màn hình chức năng của chương trình. Hệ thống lấy thông tin chi tiết của kỳ học (SemesterId, SemeterName, StartTime, EndTime, CoursesId ) từ bảng semesters trong cơ sở dữ liệu và hiển thị danh sách kỳ học lên màn hình.

*2. Thêm kỳ học:*

1. Người dùng chọn vào nút “thêm mới” trên màn hình quản lý ỳ học và nhập thông tin kỳ học mới và chọn nút “Thêm”. Hệ thống cập nhật thông tin kỳ học trong bảng semesters và hiển thị màn hình thông tin danh sách kỳ học đã cập nhật.

*3. Sửa thông tin kỳ học:*

1. Người dùng chọn kỳ học trên bảng hiển thị cần sửa. Hệ thống hiển thị thông tin kỳ học cần sửa vào form sửa.

2. Người dùng nhập thông tin sửa và chọn nút “Sửa”. Hệ thống cập nhật thông tin kỳ học trong bảng semesters và hiển thị màn hình thông tin danh sách đã cập nhật.

4. *Xóa kỳ học:*

1. Người dùng chọn kỳ học cần xóa. Hệ thống hiển thị thông tin kỳ học cần xóa vào form sửa.

2. Người dùng chọn nút “Xóa” sẽ hiển thị 1 popup thông báo Xác nhận xóa,sau khi click nut “Yes” Hệ thống cập nhật thông tin kỳ học trong bảng semesters và hiển thị màn hình thông tin danh sách kỳ học.

5.Tìm kiếm kỳ học:

1. Người dùng nhập tên kỳ học cần tìm kiếm, bấm Enter. Hệ thống sẽ hiển thị thông tin kỳ học cần tìm lên bảng.

Use case kết thúc.

* Luồng rẽ nhánh:

1. Tại bước 2 và 3 trong luồng cơ bản nếu người dùng nhập thông tin kỳ học không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi.

2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không có.

* **Tiền điều kiện:**

Người dùng cần đăng nhập hệ thống khi có thể thực hiện use case.

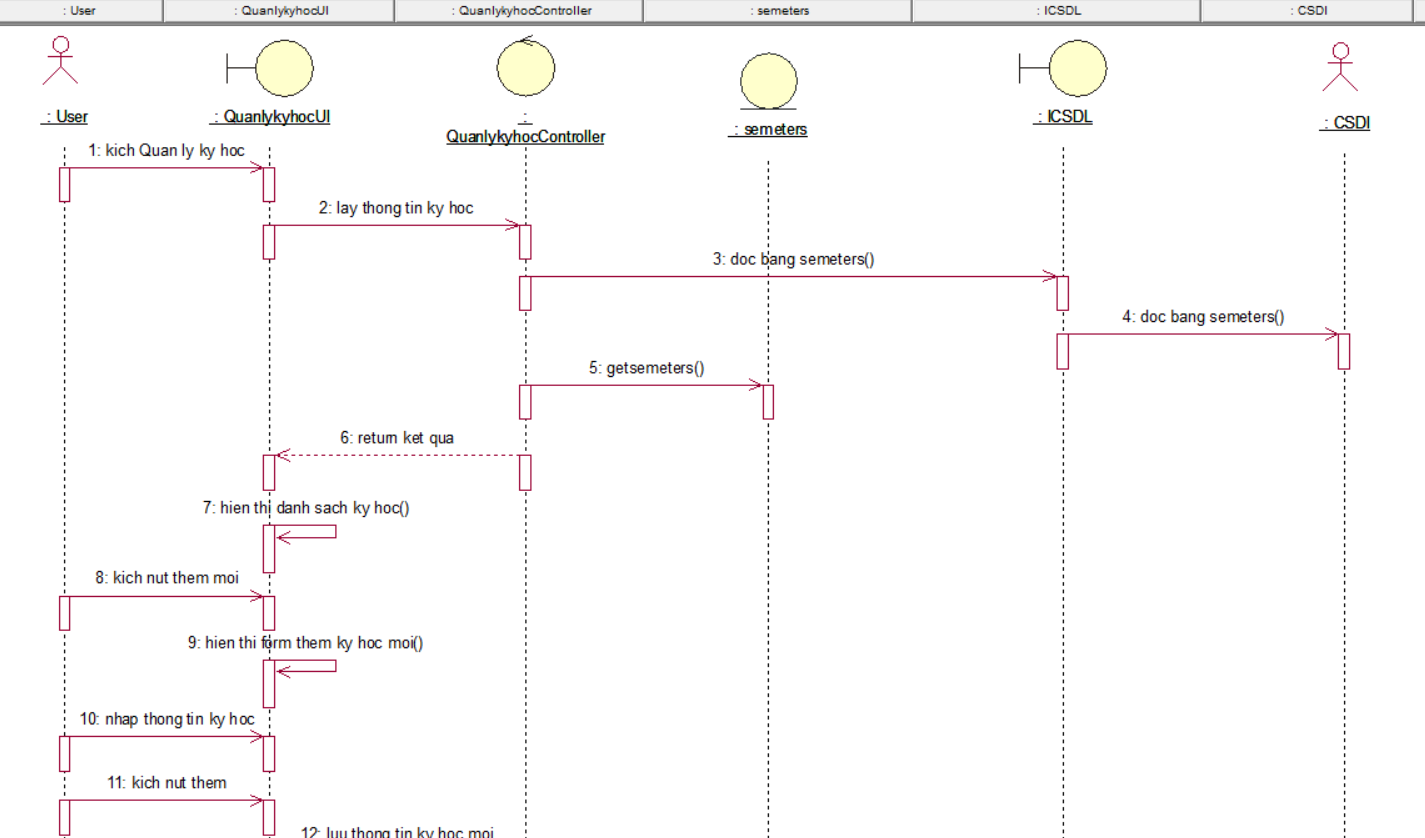
* **Hậu điều kiện:**

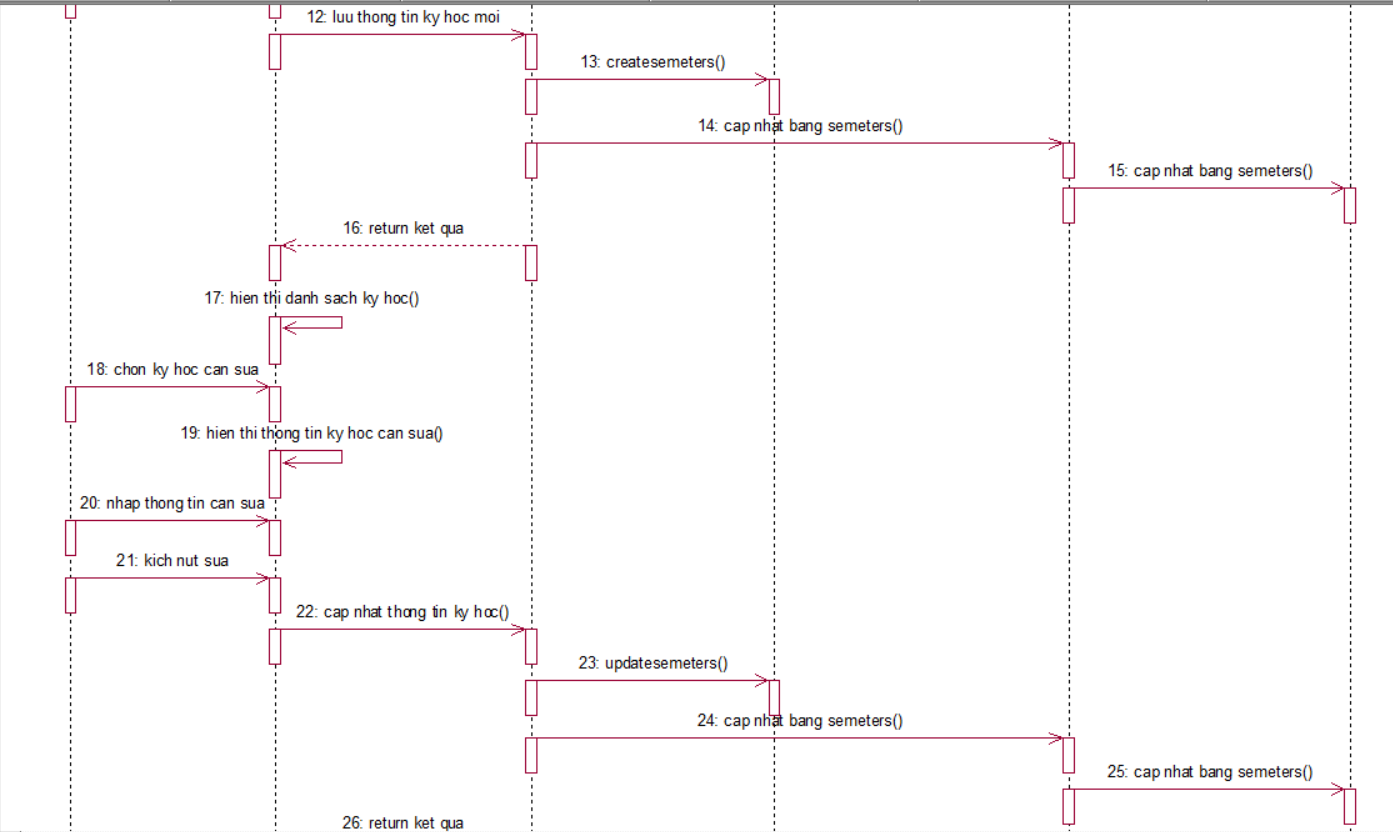
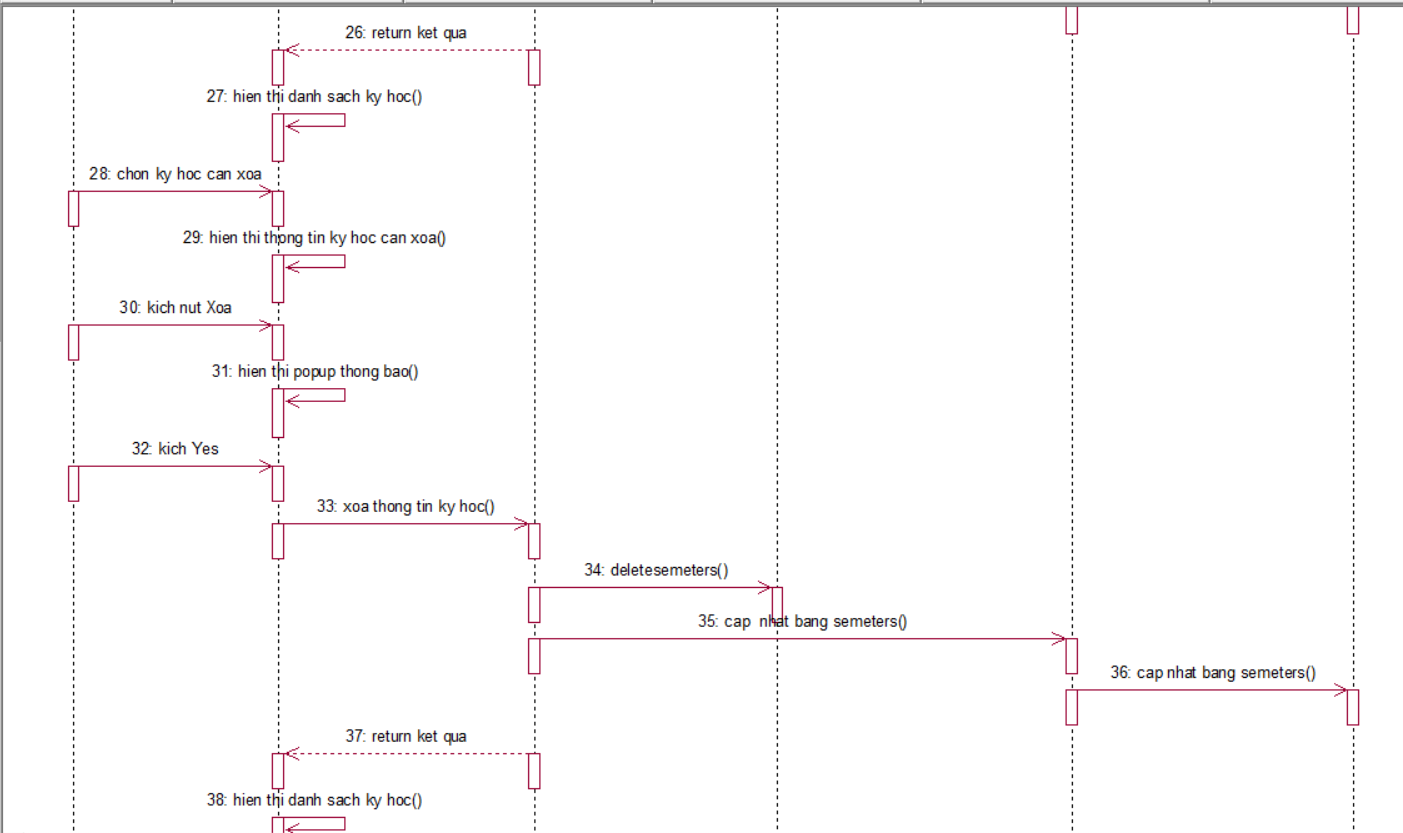
Nếu use case kết thúc thành công thì thông tin về kỳ học sẽ được cập nhập trong cơ sở dữ liệu.

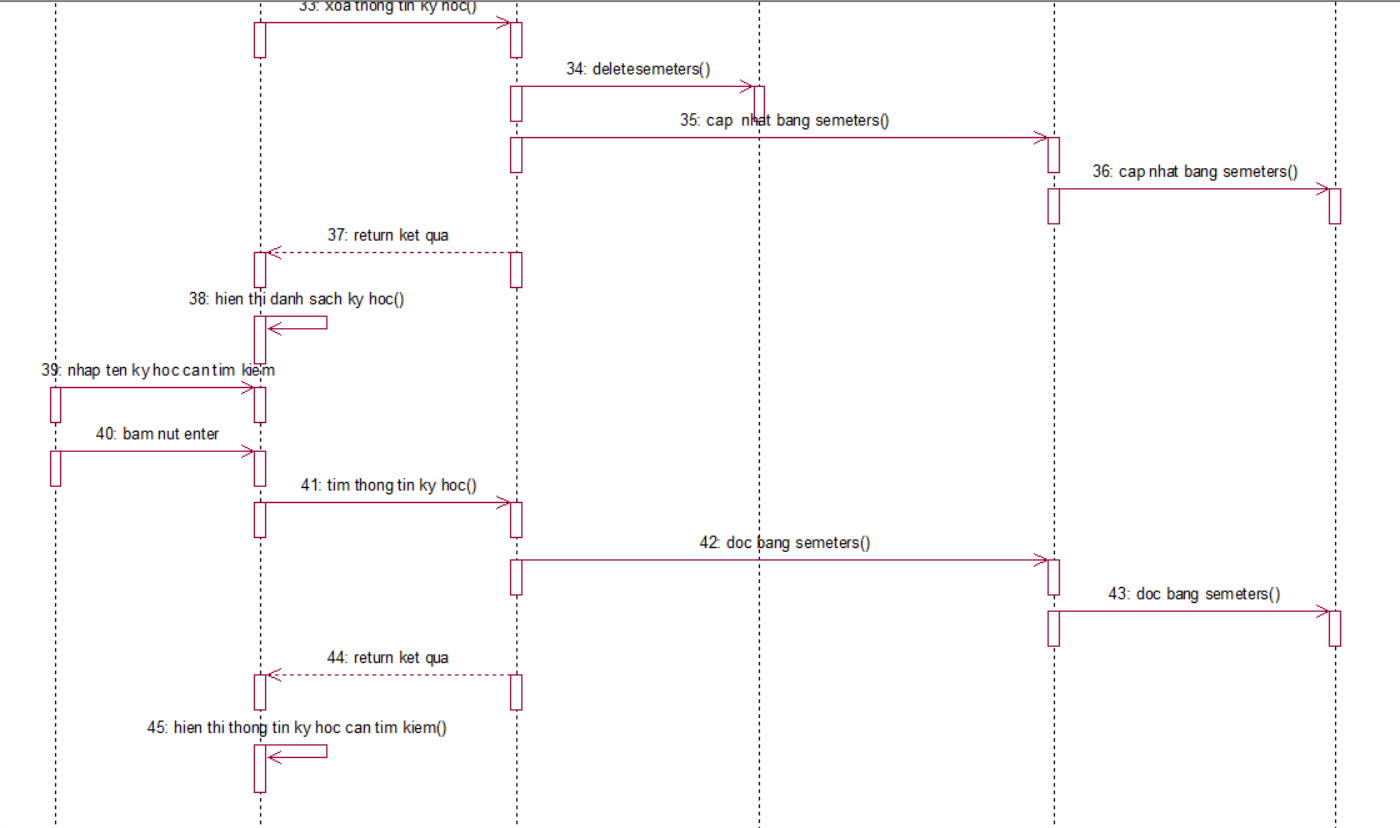
* **Điểm mở rộng:**

Không có.

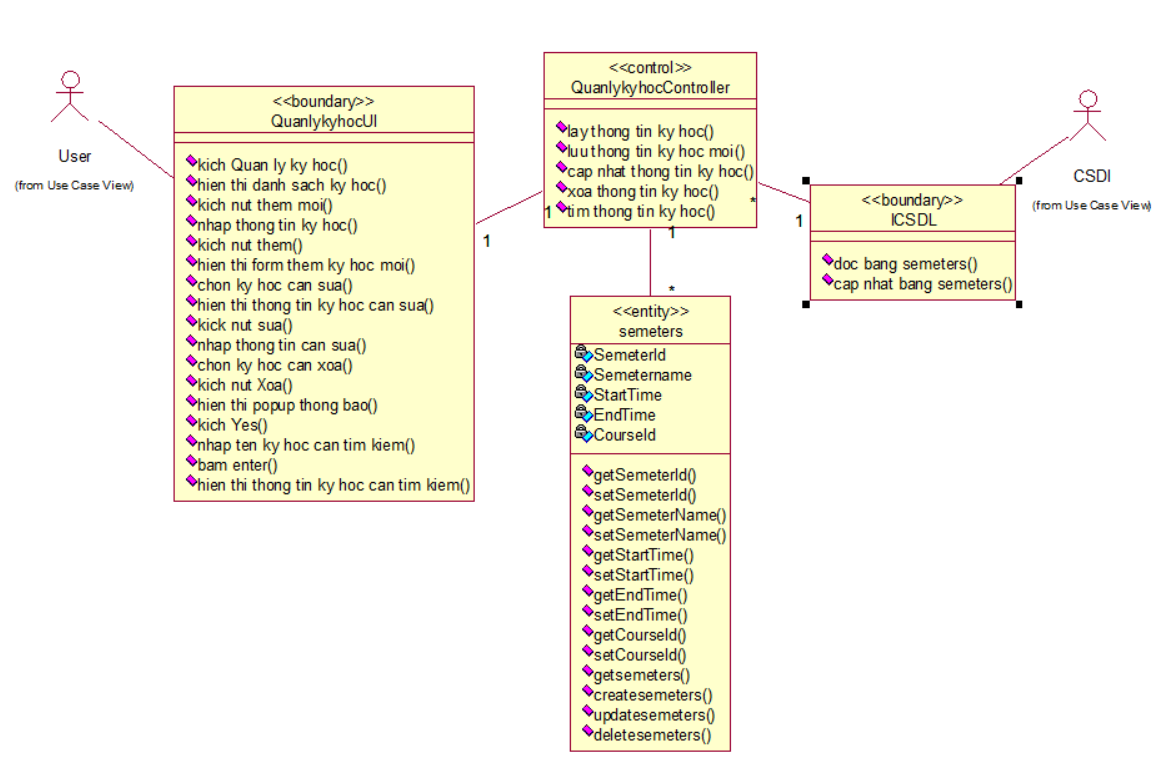
**b) Phân tích use case**

* **Biểu đồ trình tự**

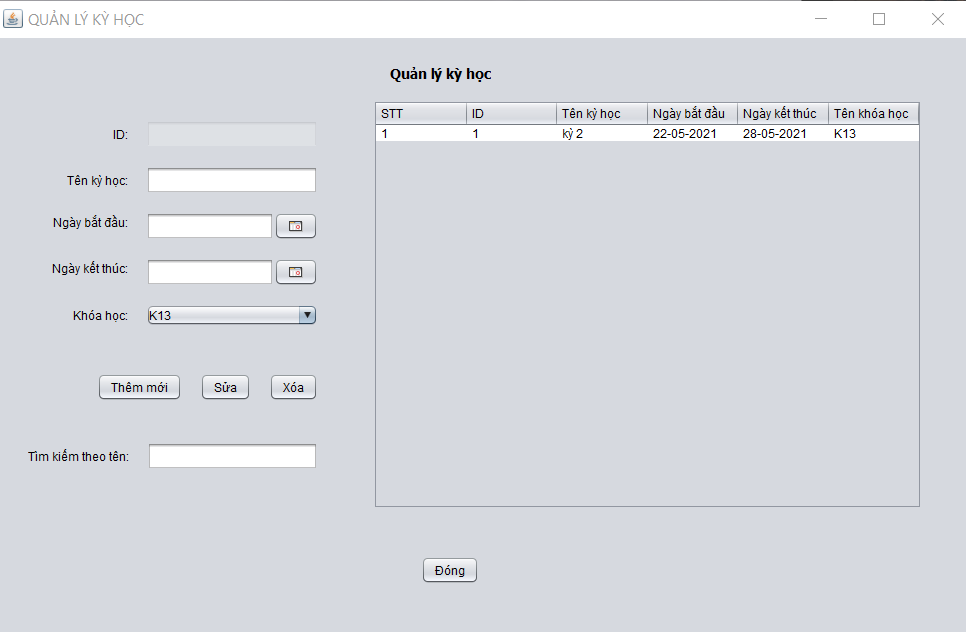




Hình 2.10. Biểu đồ trình tự use case “Quản lý kỳ học”

* **Biểu đồ lớp phân tích**

Hình 2.11. Biểu đồ lớp phân tích use case “Quản lý kỳ học”

**c) Giao diện**

Hình 2.12. Giao diện “Quản lý kỳ học”

### Use case Quản lý lớp học

**a) Mô tả use case**

* **Tên use case**

Quản lý lớp học

* **Mô tả chi tiết use case:**

Use case này cho phép người dùng xem, thêm, sửa và xóa lớp học trong bảng classes.

* **Luồng sự kiện:**
* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng chọn vào nút “Quản lý lớp học” trên màn hình chức năng của chương trình. Hệ thống lấy thông tin chi tiết của lớp học (ClassId, ClassName, StartTime, EndTime, SemeterId) từ bảng classes trong cơ sở dữ liệu và hiển thị danh sách lớp học lên màn hình.

*2. Thêm lớp học:*

1. Người dùng chọn vào nút “thêm mới” trên màn hình quản lý lớp học và nhập thông tin lớp học mới và chọn nút “Thêm”. Hệ thống cập nhật thông tin lớp học trong bảng classes và hiển thị màn hình thông tin danh sách lớp học đã cập nhật.

*3. Sửa thông tin lớp học:*

1. Người dùng chọn lớp học trên bảng hiển thị cần sửa. Hệ thống hiển thị thông tin lớp học cần sửa vào form sửa.

2. Người dùng nhập thông tin sửa và chọn nút “Sửa”. Hệ thống cập nhật thông tin lớp học trong bảng classes và hiển thị màn hình thông tin danh sách đã cập nhật.

4. *Xóa lớp học:*

1. Người dùng chọn lớp học cần xóa. Hệ thống hiển thị thông tin lớp học cần xóa vào form sửa.

2. Người dùng chọn nút “Xóa” sẽ hiển thị 1 popup thông báo Xác nhận xóa,sau khi click nut “Yes” Hệ thống cập nhật thông tin lớp học trong bảng classes và hiển thị màn hình thông tin danh sách lớp học.

5.Tìm kiếm lớp học:

1. Người dùng nhập tên lớp học cần tìm kiếm, bấm Enter. Hệ thống sẽ hiển thị thông tin lớp học cần tìm lên bảng.

Usecase kết thúc.

* Luồng rẽ nhánh:

1. Tại bước 2 và 3 trong luồng cơ bản nếu người dùng nhập thông tin lớp học không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi.

2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không có.

* **Tiền điều kiện:**

Người dùng cần đăng nhập hệ thống khi có thể thực hiện use case.

* **Hậu điều kiện:**

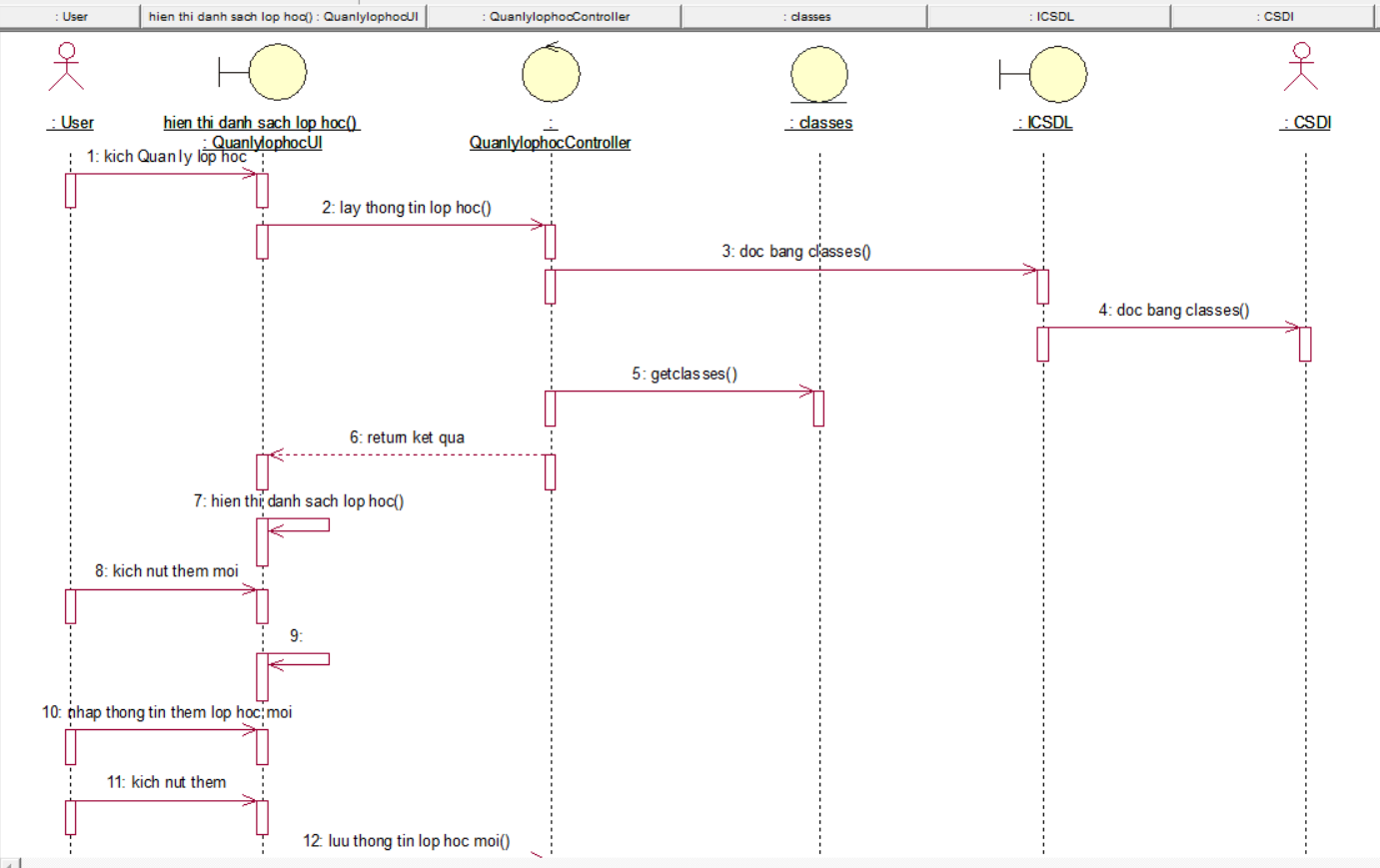
Nếu use case kết thúc thành công thì thông tin về lớp học sẽ được cập nhập trong cơ sở dữ liệu.

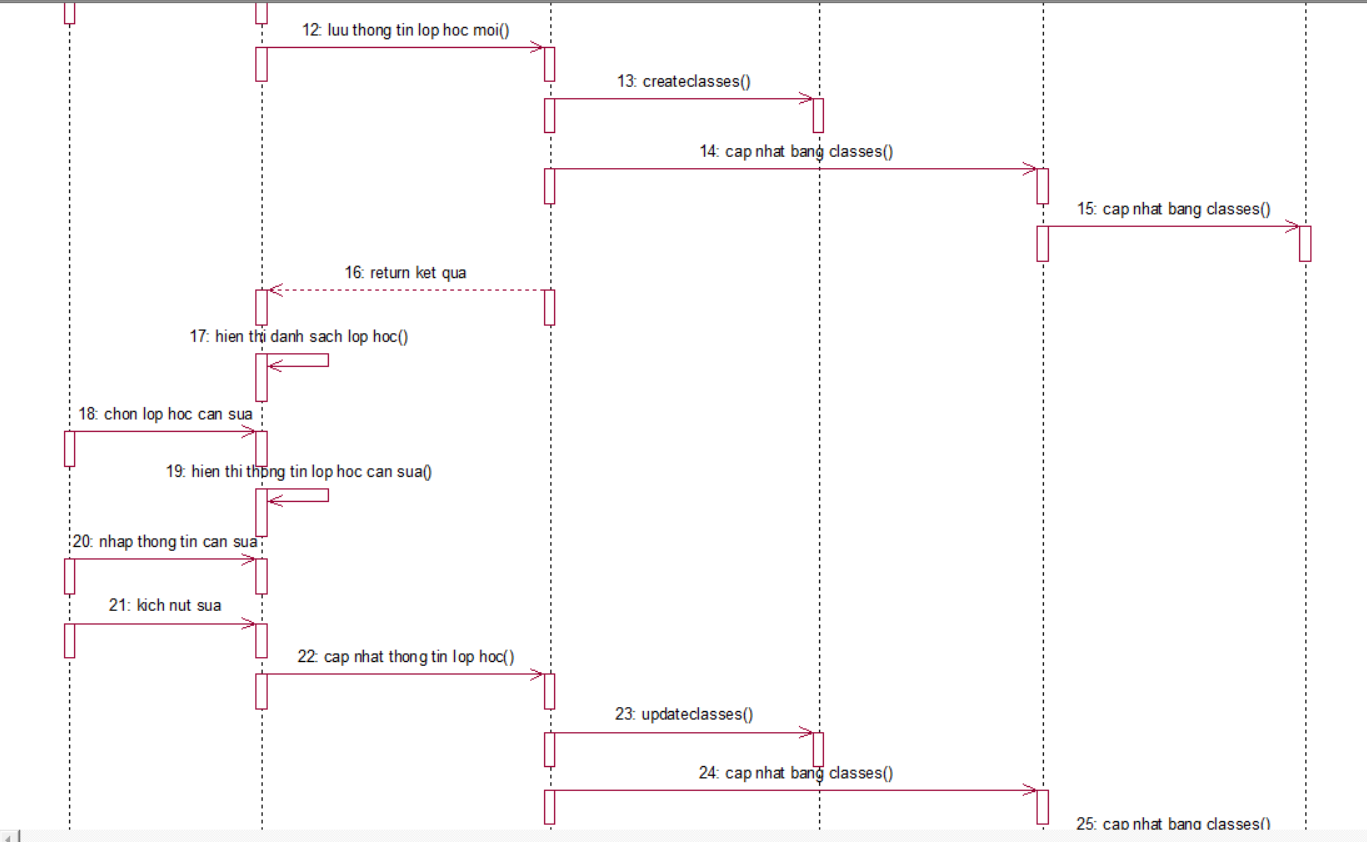
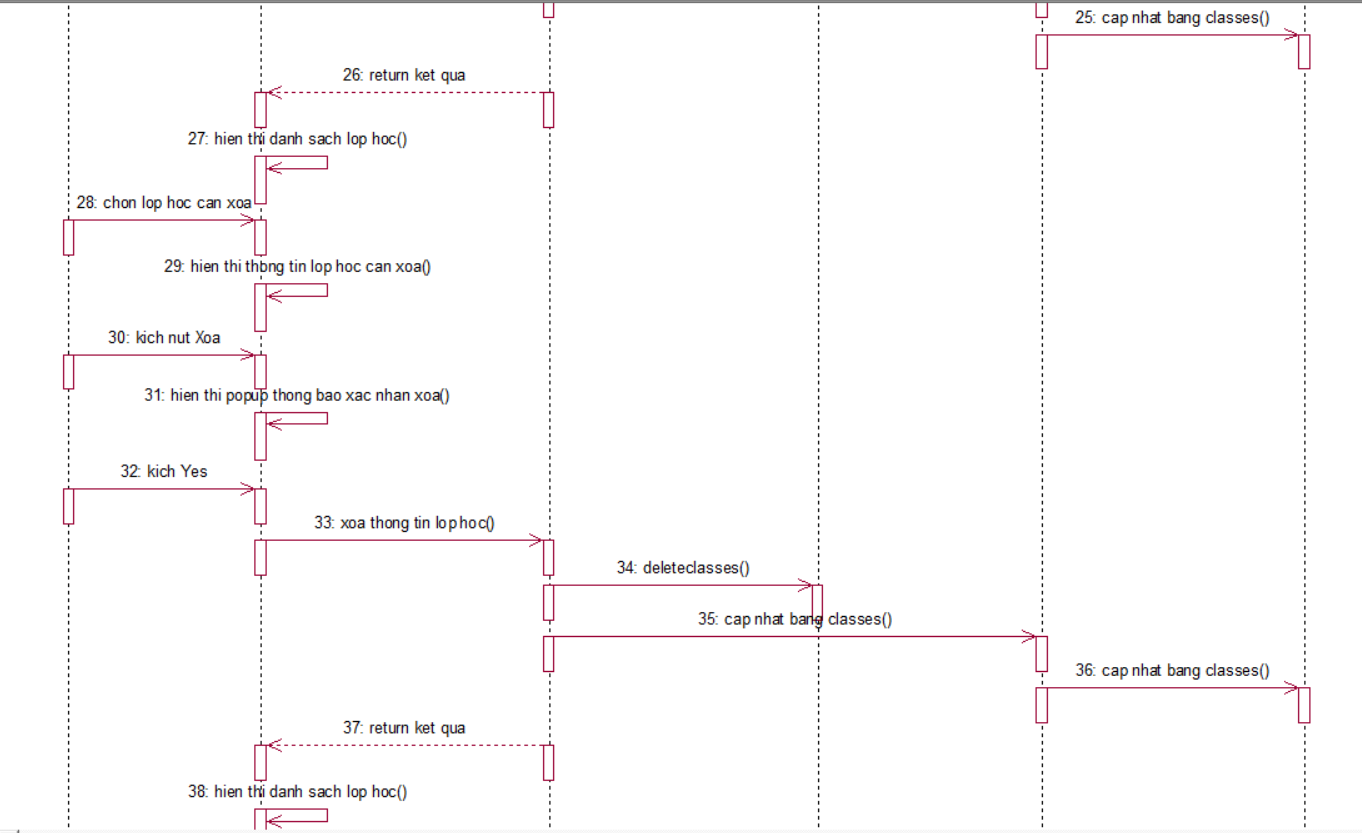
* **Điểm mở rộng:**

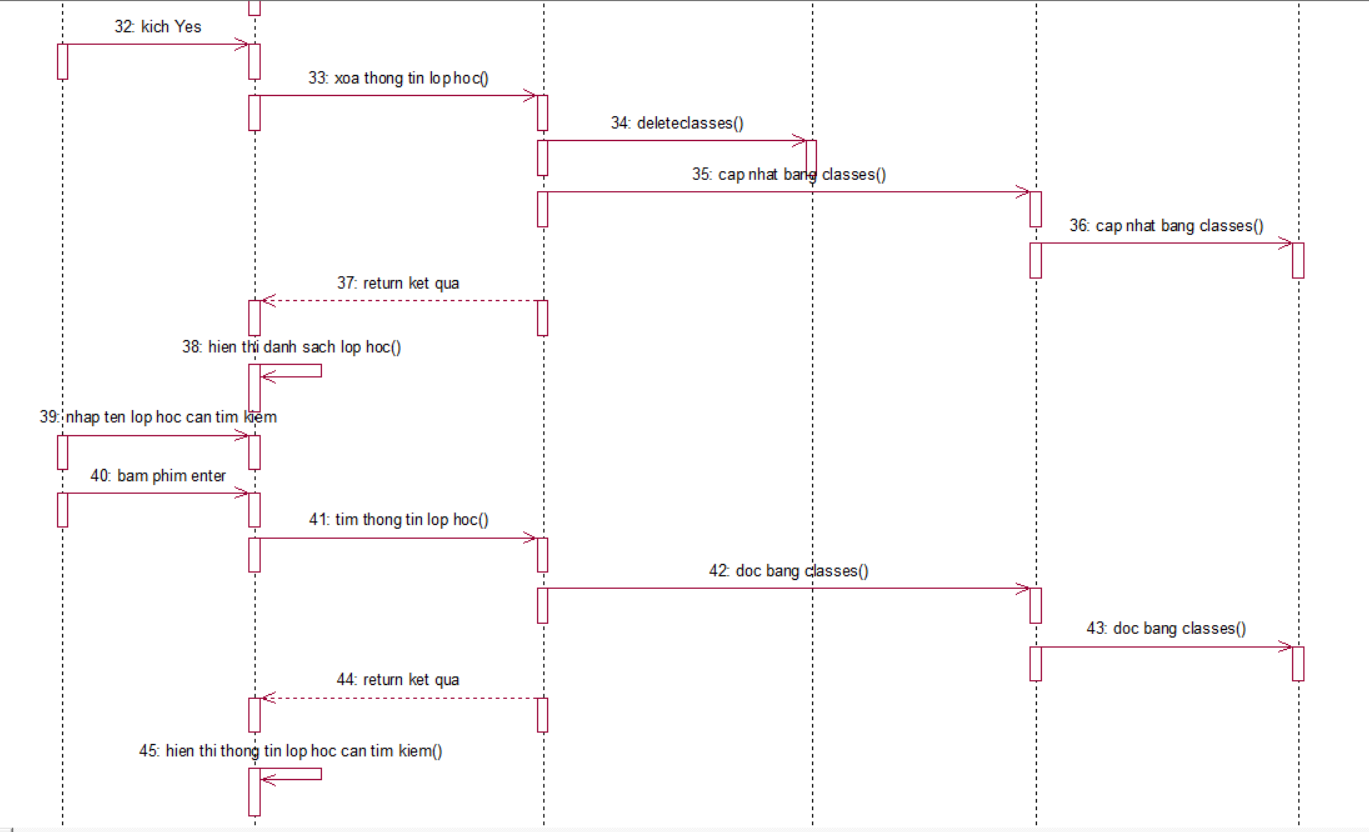
Không có.

**b) Phân tích use case**

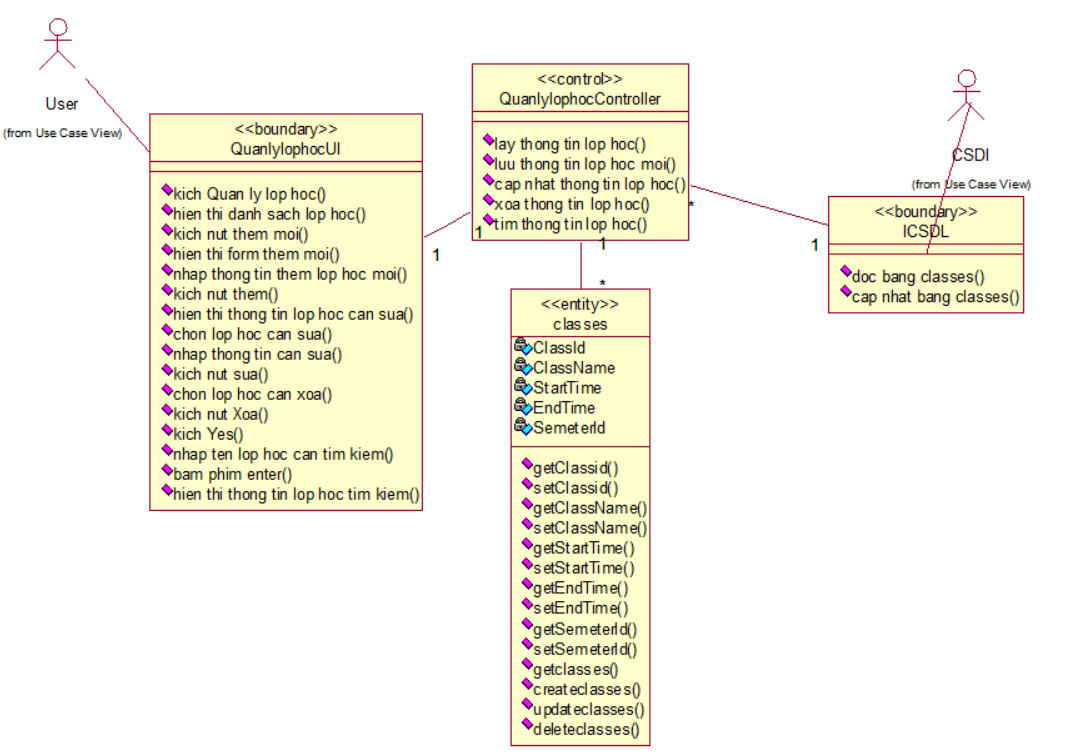
* **Biểu đồ trình tự**



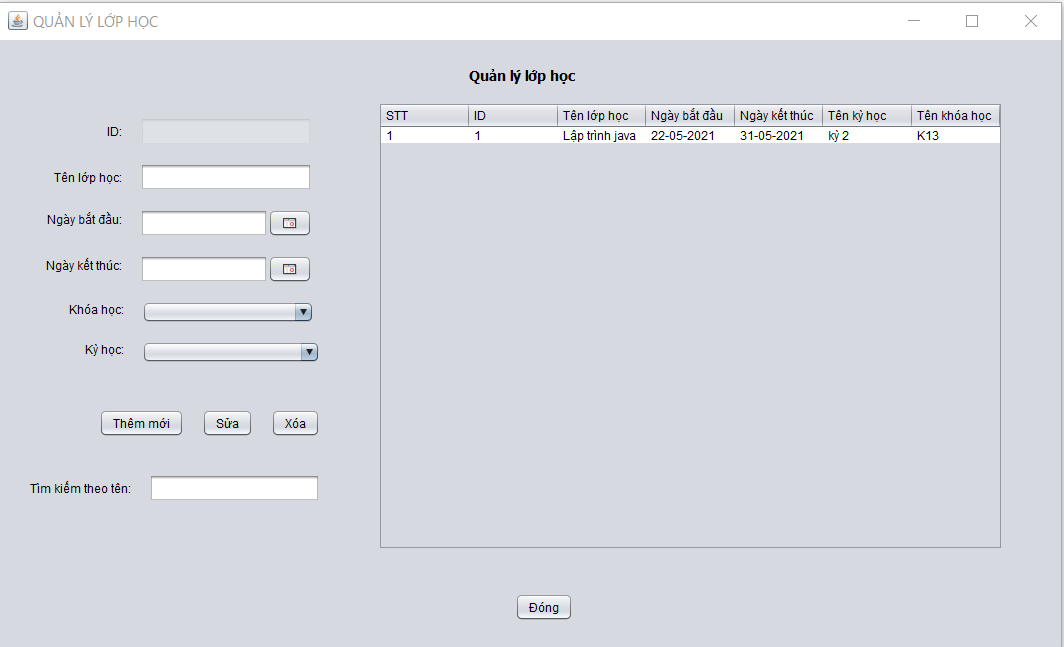




Hình 2.13. Biểu đồ trình tự use case “Quản lý lớp học”

* **Biểu đồ lớp phân tích**

Hình 2.14. Biểu đồ lớp phân tích use case “Quản lý lớp học”

**c) Giao diện**

Hình 2.15. Giao diện “Quản lý lớp học”

### Use case Quản lý học sinh

**a) Mô tả use case**

* **Tên use case**

Quản lý học sinh

* **Mô tả chi tiết use case:**

Use case này cho phép người quản trị xem, thêm, sửa và xóa học sinh trong bảng students.

* **Luồng sự kiện:**
* Luồng cơ bản:

1. Use case này bắt đầu khi người dùng chọn vào nút “Quản lý học sinh” trên màn hình chức năng của chương trình. Hệ thống lấy thông tin chi tiết của người dùng (StudentId, FullName, YearOfBirth, Address, ClassId ) từ bảng students trong cơ sở dữ liệu và hiển thị danh sách học sinh lên màn hình.

*2. Thêm học sinh:*

1. Người dùng chọn vào nút “thêm mới” trên màn hình quản lý học sinh và nhập thông tin học sinh mới và chọn nút “Thêm”. Hệ thống cập nhật thông tin học sinh trong bảng students và hiển thị màn hình thông tin danh sách học sinh đã cập nhật.

*3. Sửa thông tin học sinh:*

1. Người dùng chọn học sinh trên bảng hiển thị cần sửa. Hệ thống hiển thị thông tin học sinh cần sửa vào form sửa.

2. Người dùng nhập thông tin sửa và chọn nút “Sửa”. Hệ thống cập nhật thông tin học sinh trong bảng students và hiển thị màn hình thông tin danh sách đã cập nhật.

4. *Xóa học sinh:*

1. Người dùng chọn học sinh cần xóa. Hệ thống hiển thị thông tin học sinh cần xóa vào form sửa.

2. Người dùng chọn nút “Xóa” sẽ hiển thị 1 popup thông báo Xác nhận xóa,sau khi click nut “Yes” Hệ thống cập nhật thông tin khóa học trong bảng students và hiển thị màn hình thông tin danh sách học sinh.

5.Tìm kiếm học sinh:

1. Người dùng nhập tên học sinh cần tìm kiếm, bấm Enter. Hệ thống sẽ hiển thị thông tin học sinh cần tìm lên bảng.

Usecase kết thúc.

* Luồng rẽ nhánh:

1. Tại bước 2 và 3 trong luồng cơ bản nếu người dùng nhập thông tin học sinh không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi.

2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:**

Không có.

* **Tiền điều kiện:**

Người dùng cần đăng nhập hệ thống khi có thể thực hiện use case.

* **Hậu điều kiện:**

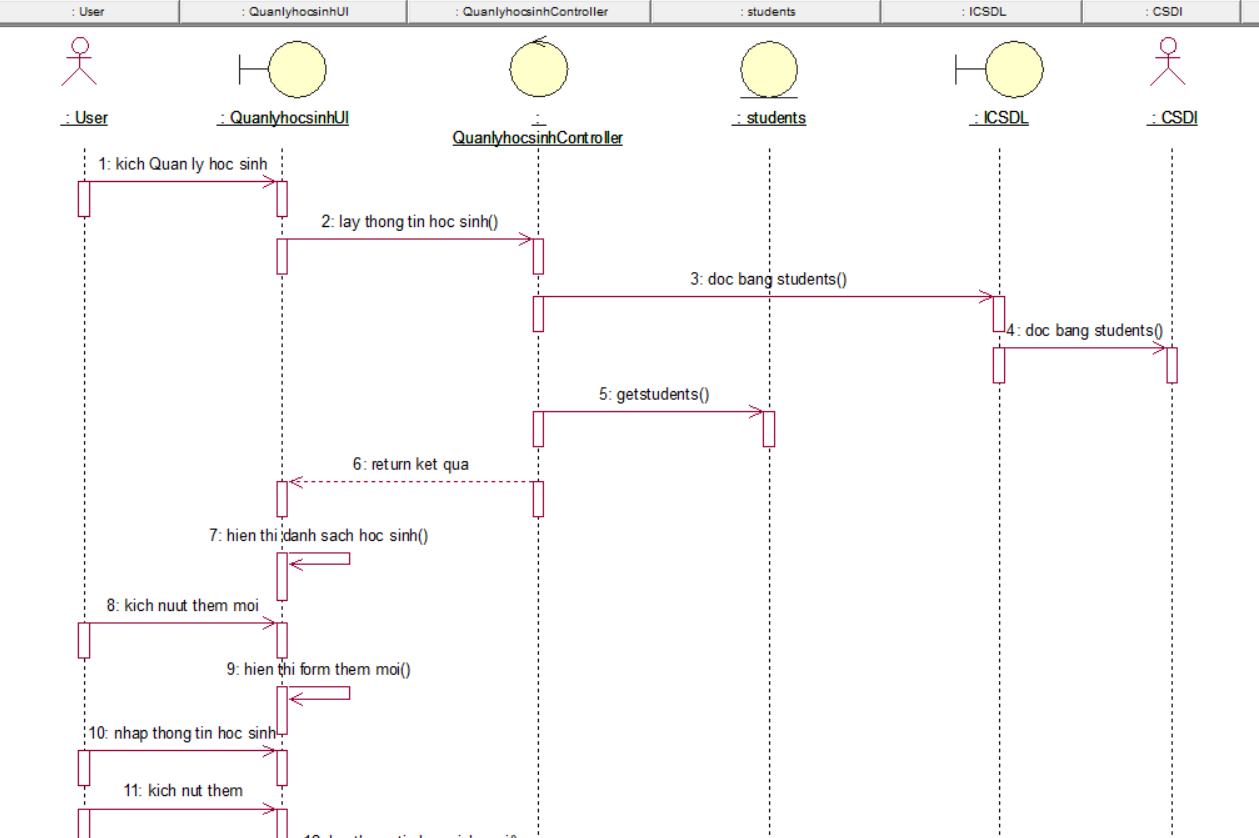
Nếu use case kết thúc thành công thì thông tin về học sinh sẽ được cập nhập trong cơ sở dữ liệu.

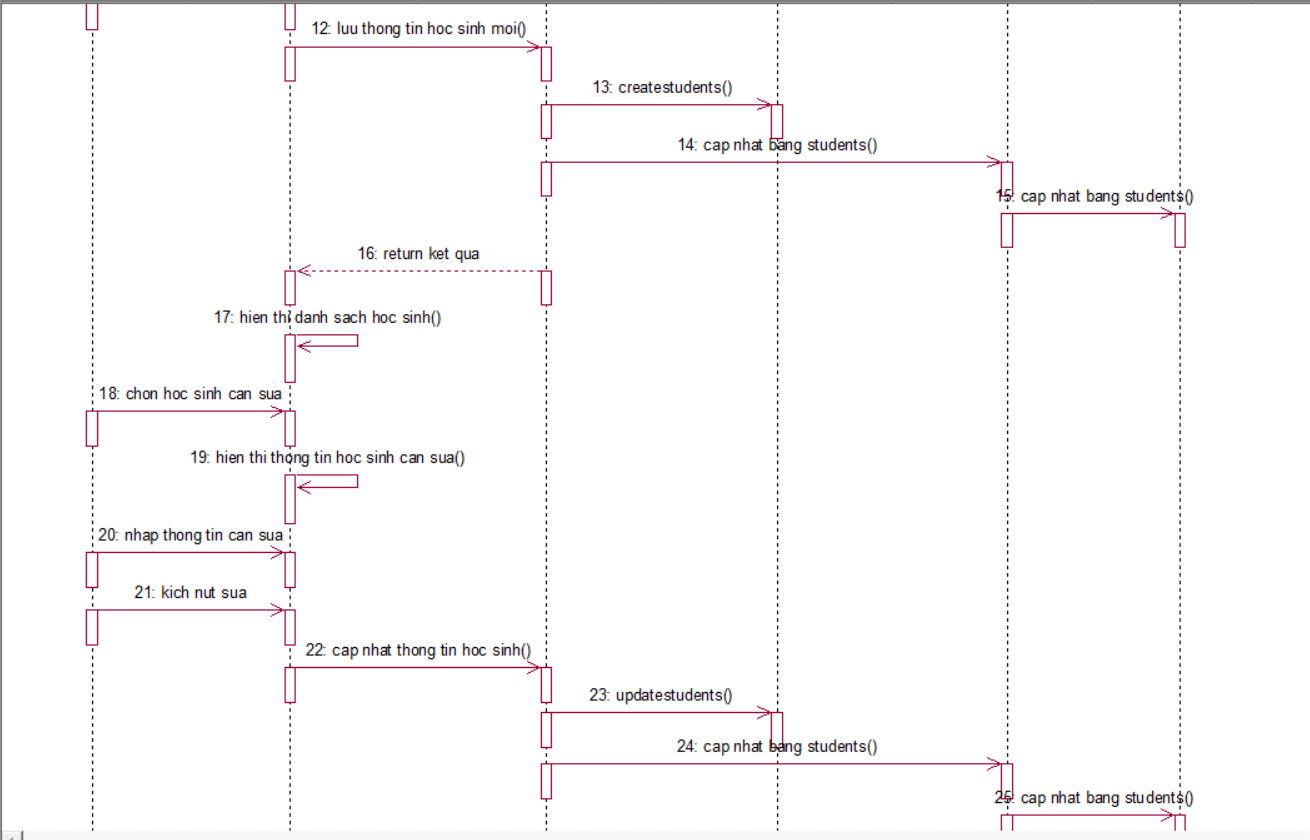
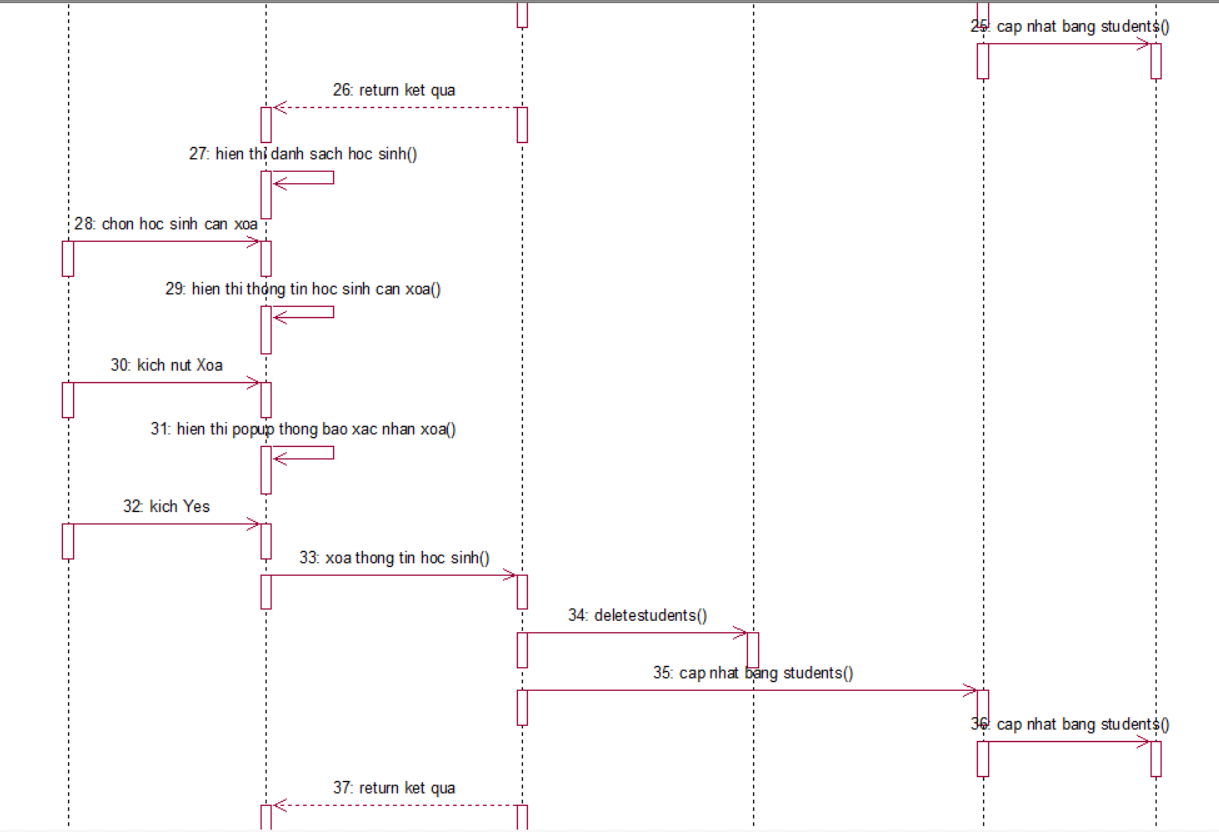
* **Điểm mở rộng:**

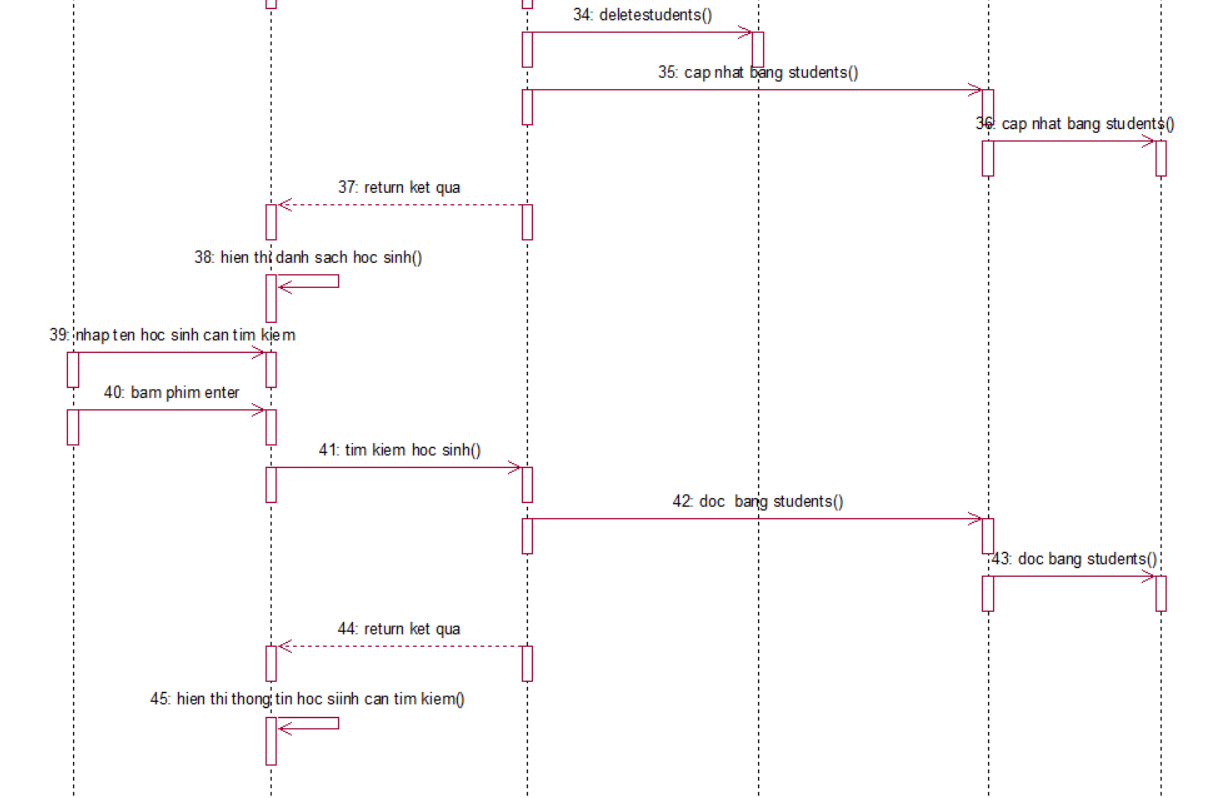
Không có.

**b) Phân tích use case**

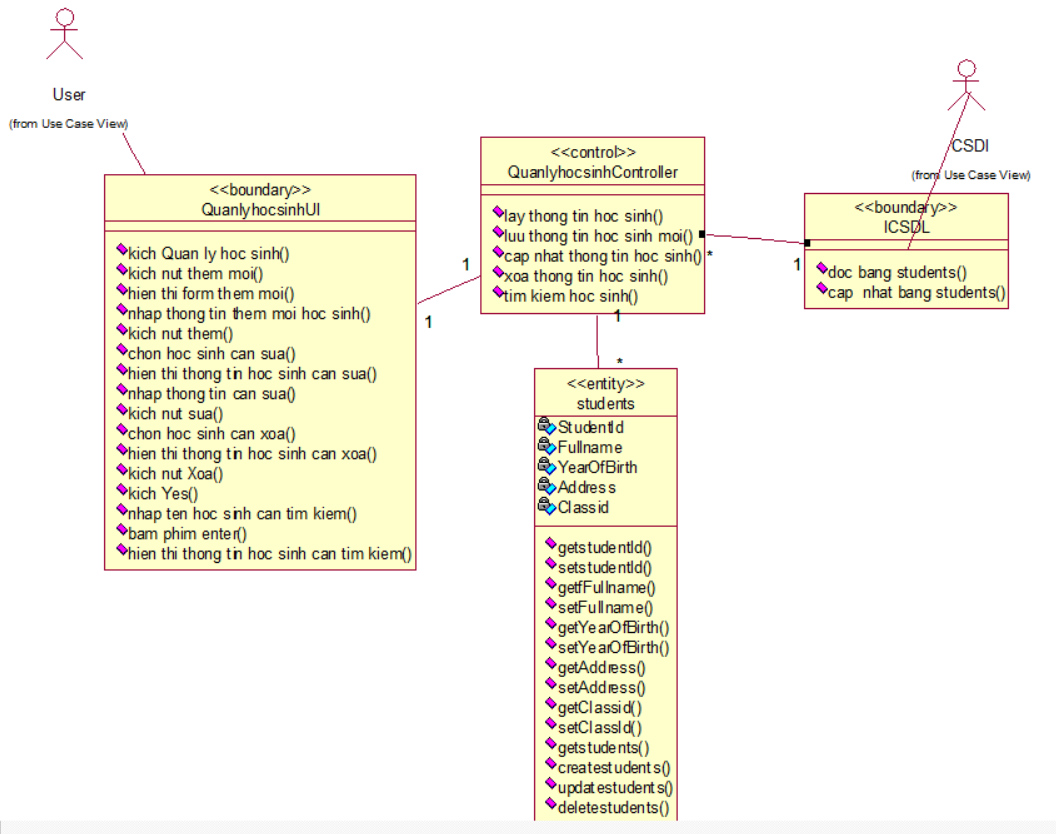
* **Biểu đồ trình tự**



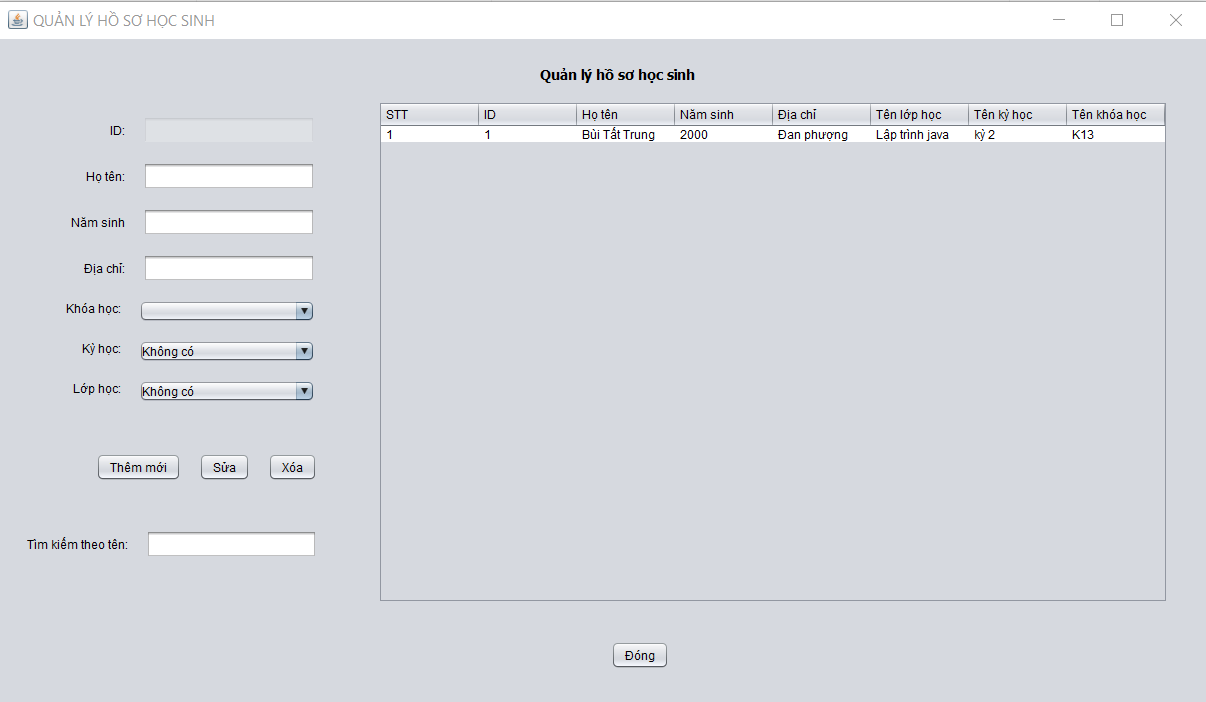




Hình 2.16. Biểu đồ trình tự use case “Quản lý học sinh”

* **Biểu đồ lớp:**

Hình 2.17. Biểu đồ lớp use case “Quản lý học sinh”

**c) Giao diện**

Hình 2.18. Giao diện “Quản lý học sinh”

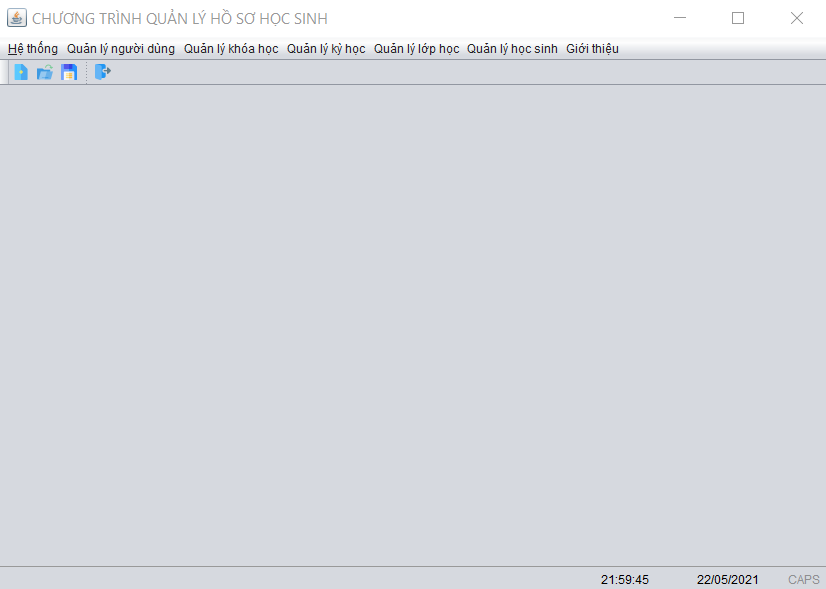
# THỰC NGHIỆM VÀ ĐÁNH GIÁ

## Thực nghiệm chương trình

### Đăng nhập

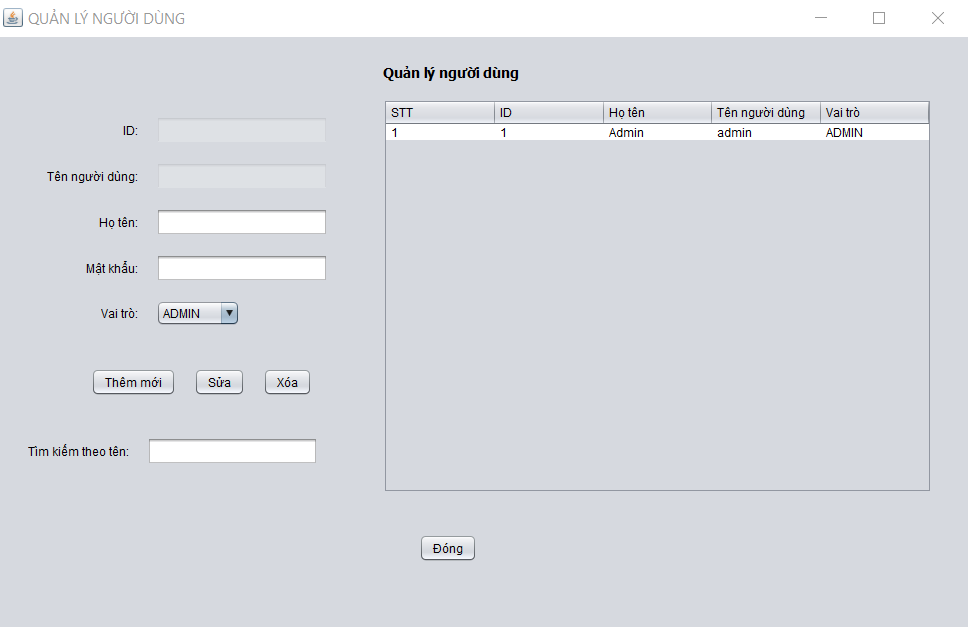
* Nhập thông tin tài khoản.

Hình 3.1. Màn hình Form đăng nhập

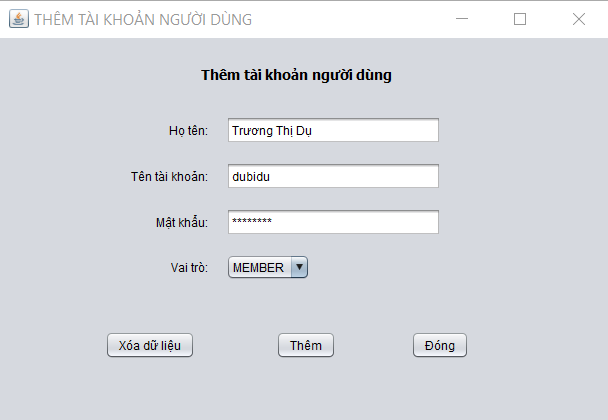
* Đăng nhập thành công vào hệ thống.

Hình 3.2. Màn hình chức năng

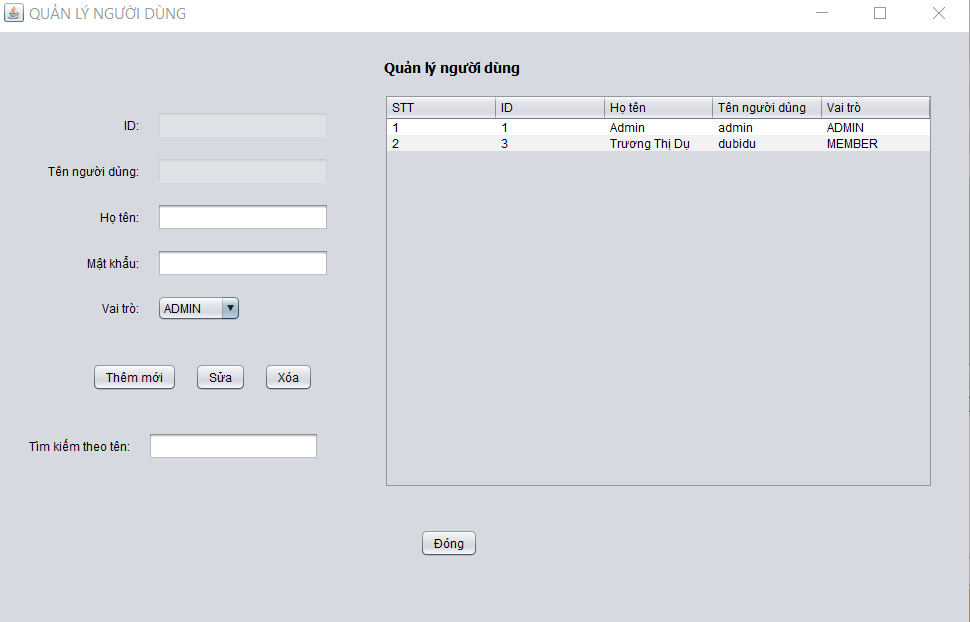
### Quản lý người dùng

* Thêm người dùng

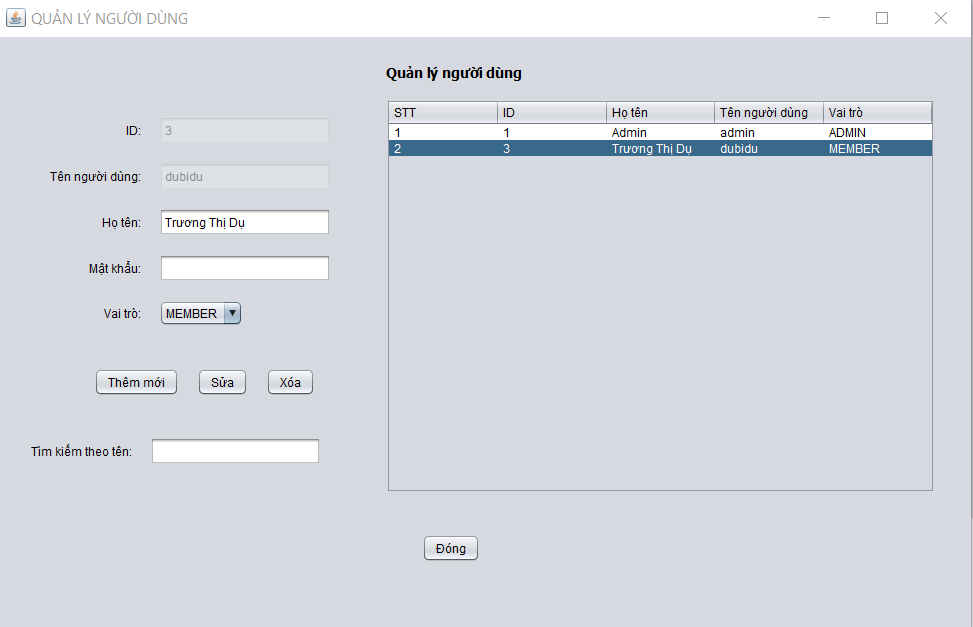
Hình 3.3. Màn hình quản lý người dùng

* Chọn “Thêm mới”, nhập thông tin người dùng mới và ấn “Thêm”

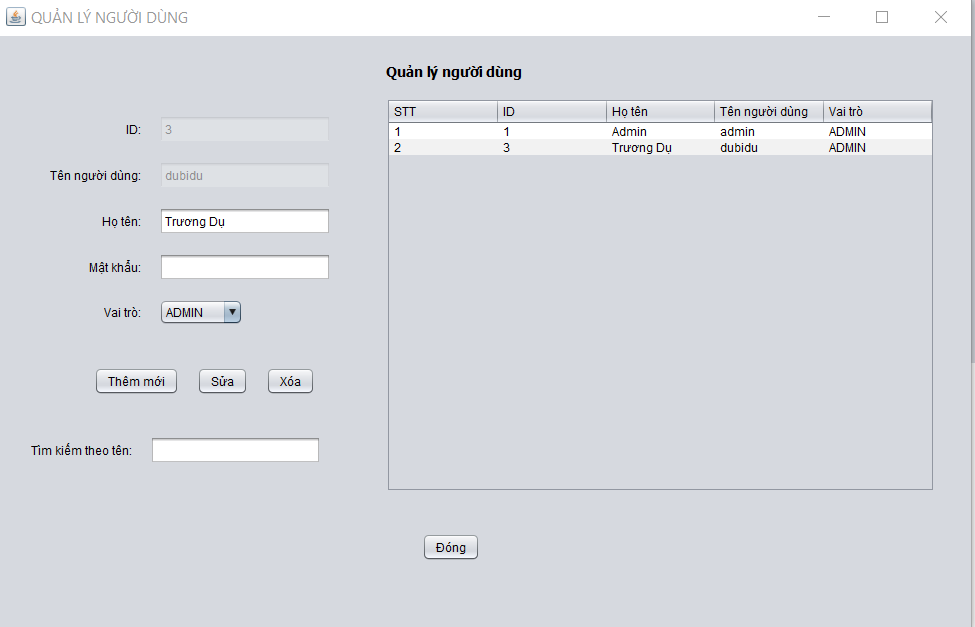
Hình 3.4. Màn hình thêm người dùng

* Kết quả:

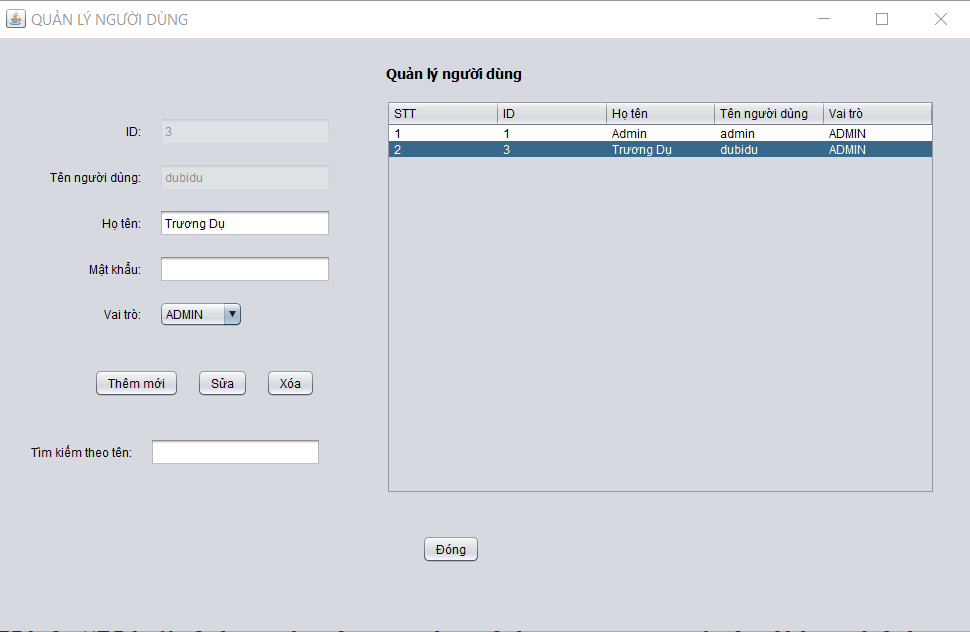
Hình 3.5. Màn hình kết quả dữ liệu sau khi cập nhật

* Sửa người dùng
* Chọn một tài khoản người dùng:

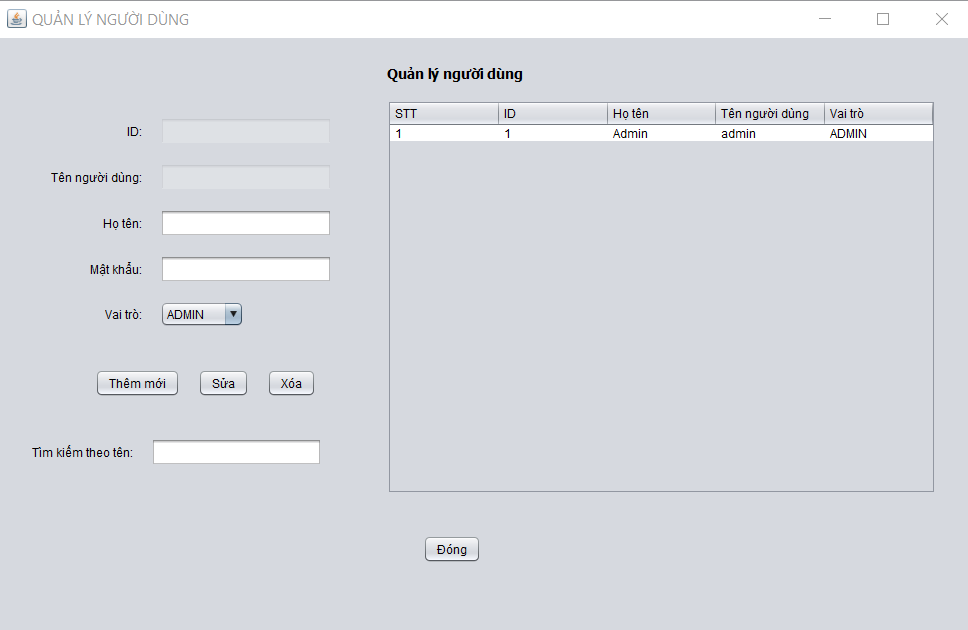
Hình 3.6. Màn hình sửa tài khoản người dùng

* Sửa thông tin người dùng, chọn “Sửa” và kết quả:

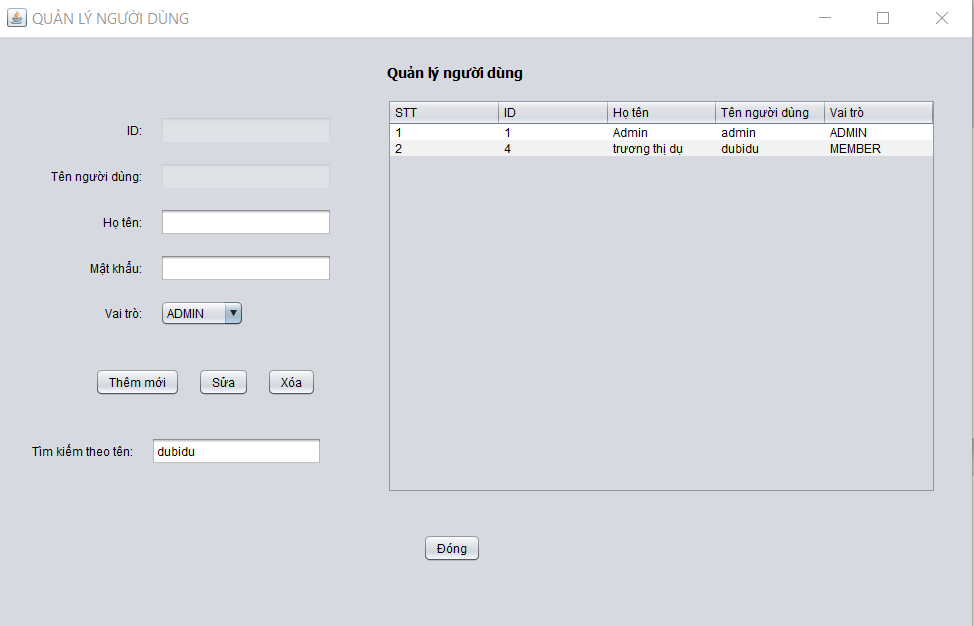
Hình 3.7. Màn hình kết quả sửa tài khoản người dùng

* Xóa tài khoản người dùng
* Chọn một tài khoản:

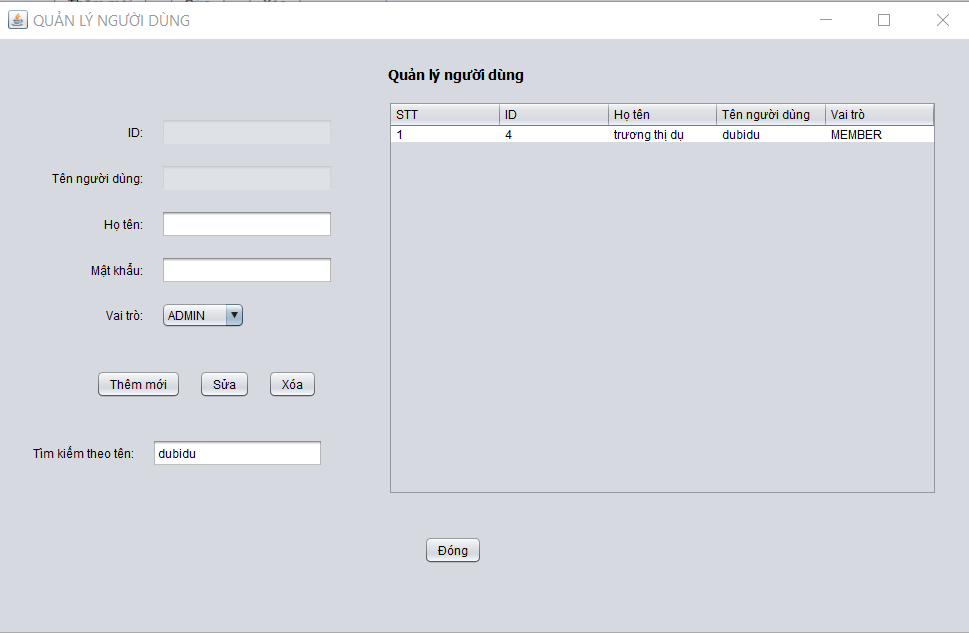
Hình 3.8. Màn hình xóa tài khoản người dùng

* Chọn “Xóa” thông tin được cập nhật trong cơ sở dữ liệu và kết quả:

Hình 3.9. Màn hình kết quả xóa tài khoản người dùng

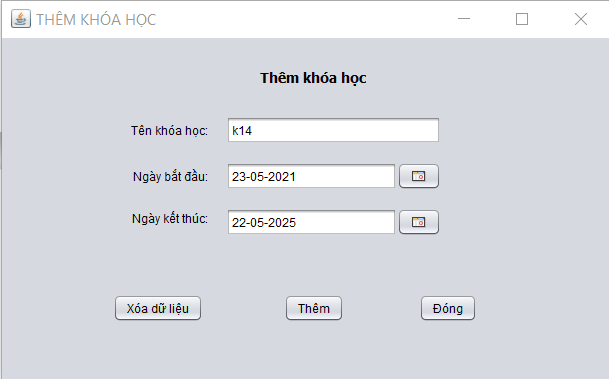
* Tìm kiếm tài khoản người dùng

Hình 3.10. Màn hình tìm kiếm tài khoản người dùng

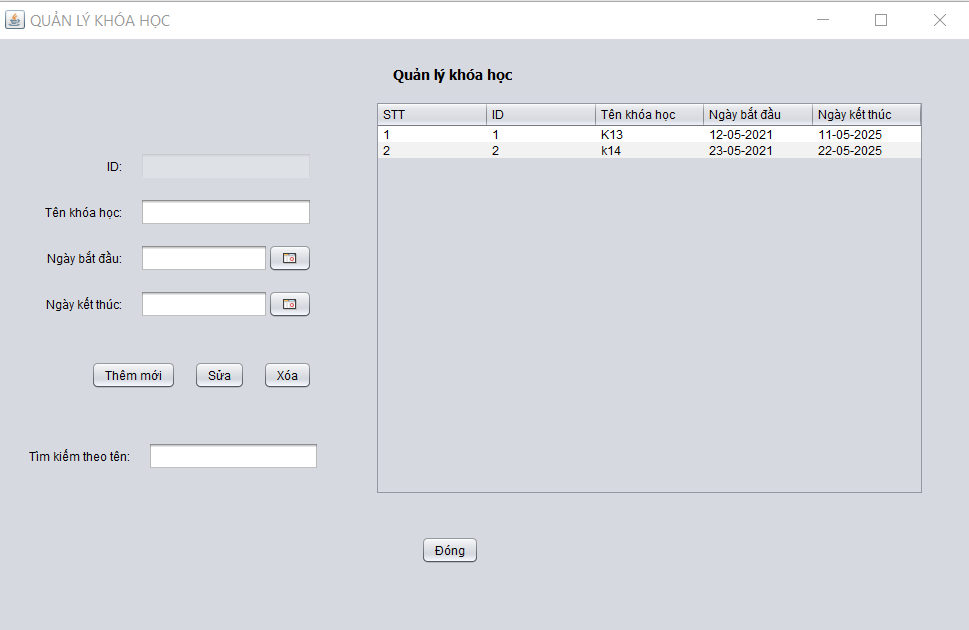
* Bấm phím enter:

Hình 3.11. Màn hình kết quả tìm kiếm tài khoản người dùng

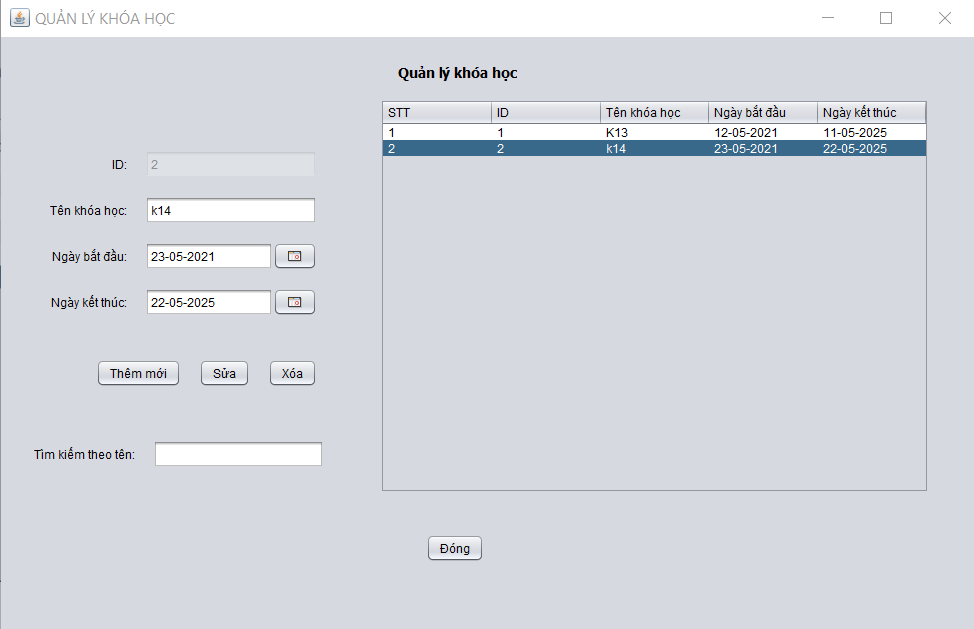
### Quản lý khóa học

* Thêm khóa học
* Chọn “thêm mới” và điền thông tin khóa học.

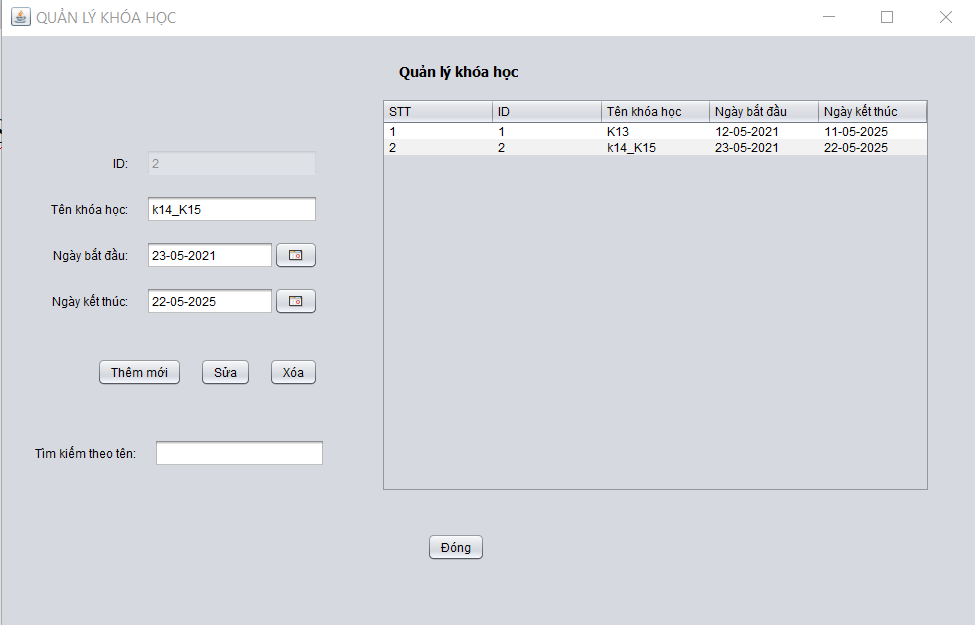
Hình 3.12. Màn hình thêm khóa học

* Chọn “Thêm” thông tin được cập nhật trong cơ sở dữ liệu và kết quả:

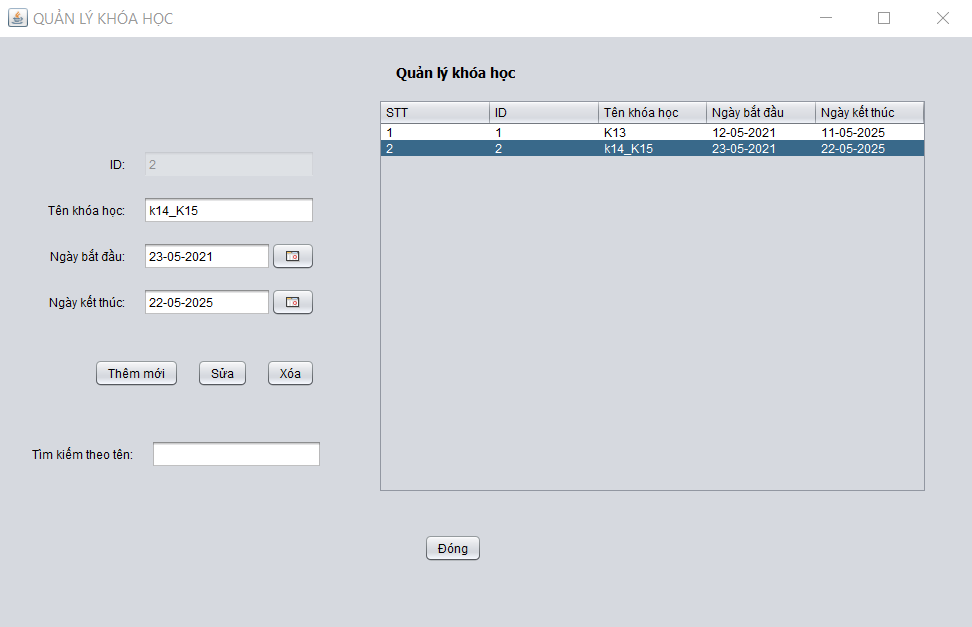
Hình 3.13. Màn hình kết quả thêm khóa học

* Sửa thông tin khóa học
* Chọn Khóa học cần sửa, sửa thông tin.

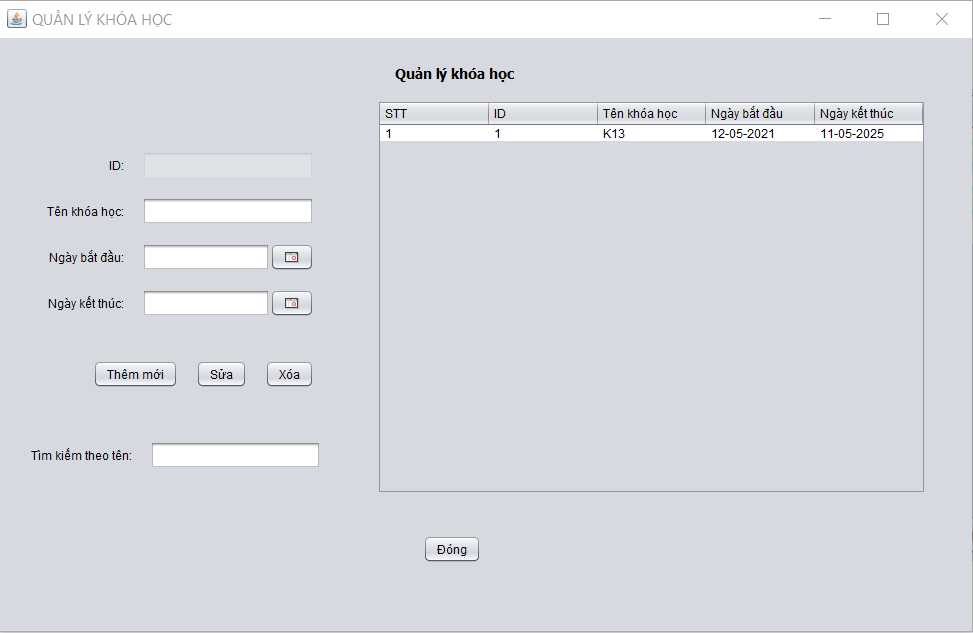
Hình 3.14. Màn hình sửa thông tin khóa học

* Chọn “Sửa” thông tin được cập nhật trong cơ sở dữ liệu và kết quả:

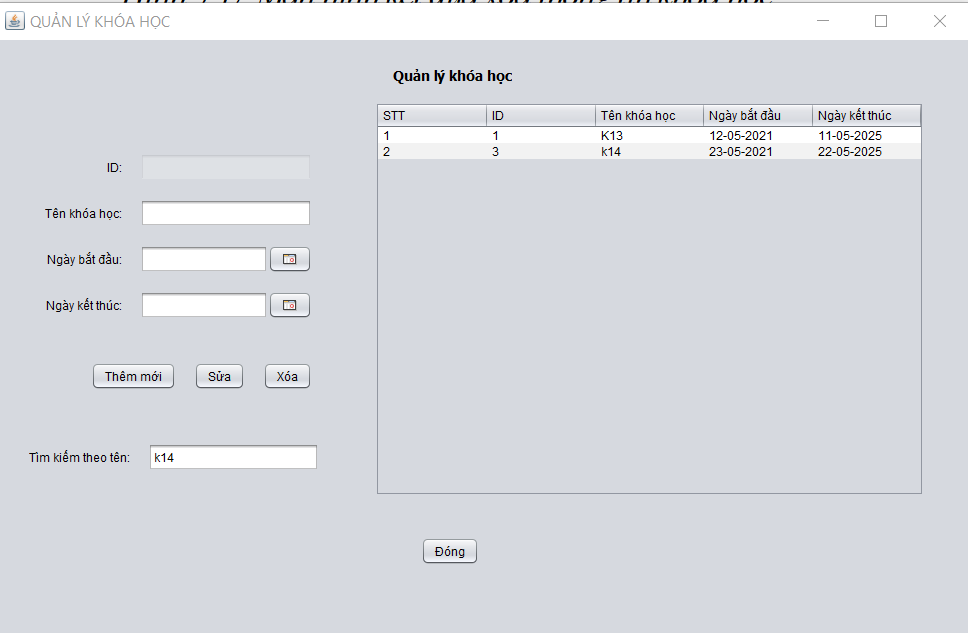
Hình 3.15. Màn hình kết quả sửa thông tin khóa học

* Xóa thông tin khóa học
* Chọn Khóa học cần xóa

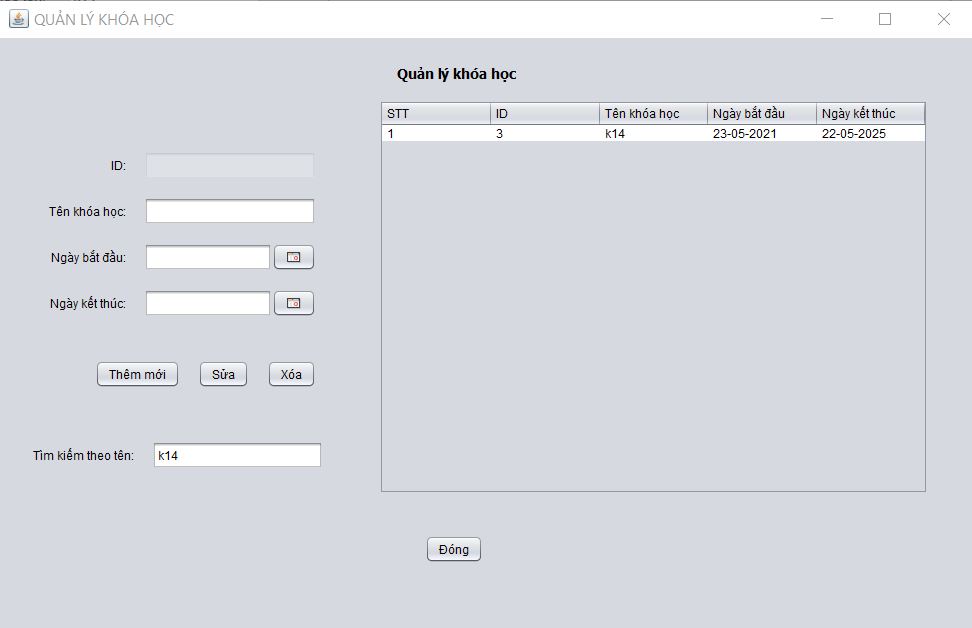
Hình 3.16. Màn hình xóa thông tin khóa học

* Chọn “Xóa” thông tin được cập nhật trong cơ sở dữ liệu và kết quả:

Hình 3.17. Màn hình kết quả xóa thông tin khóa học

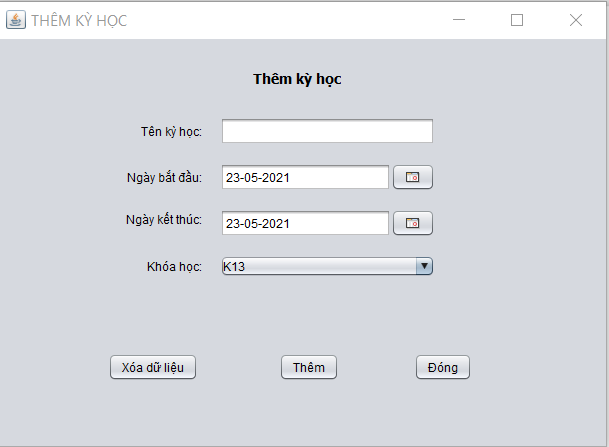
* Tìm kiếm khóa học
* Nhập tên khóa học cần tìm kiếm

Hình 3.18. Màn hình tìm kiếm thông tin khóa học

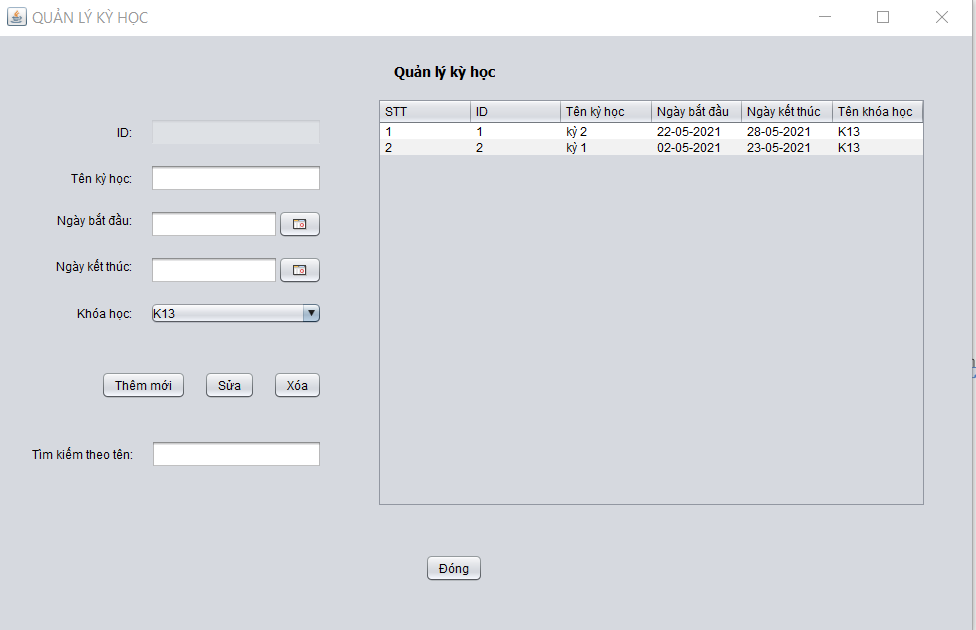
* Bấm phím enter:

Hình 3.19. Màn hình kết quả tìm kiếm thông tin khóa học

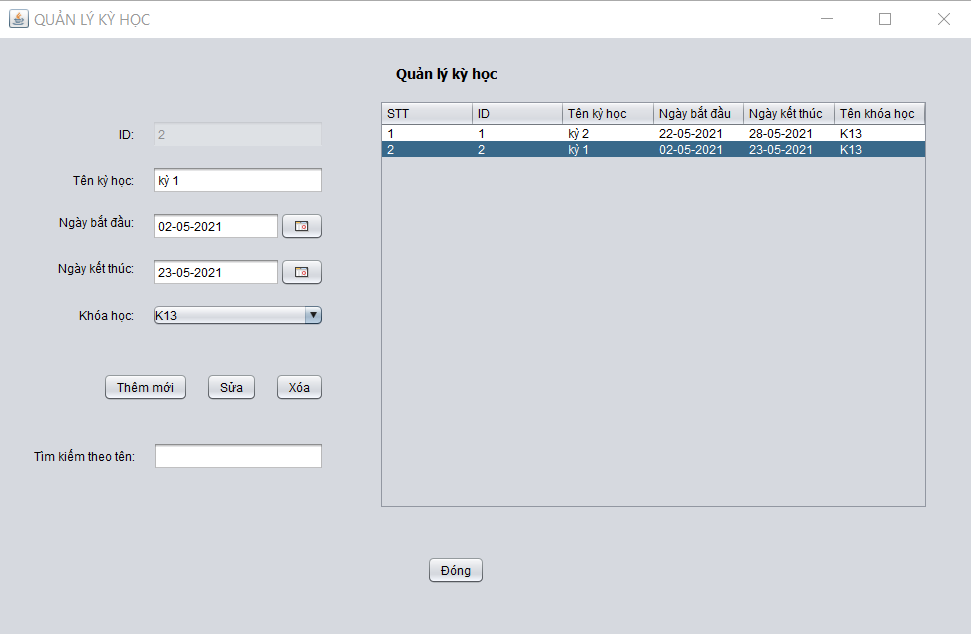
### Quản lý kỳ học

* Thêm kỳ học
* Chọn “thêm mới” và điền thông tin kỳ học.

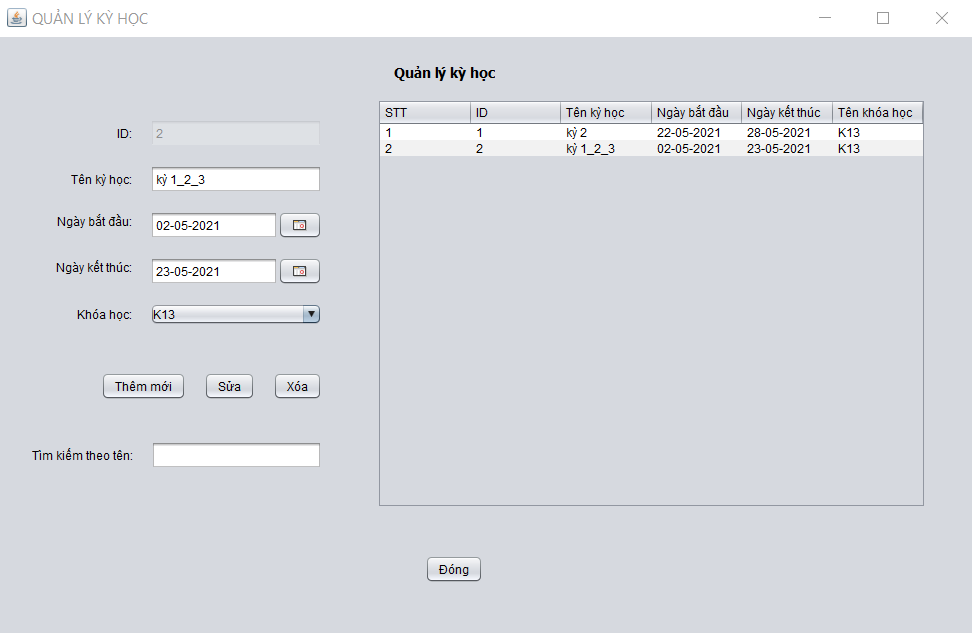
Hình 3.20. Màn hình thêm mới kỳ học

* Chọn “Thêm” thông tin được cập nhật trong cơ sở dữ liệu và kết quả:

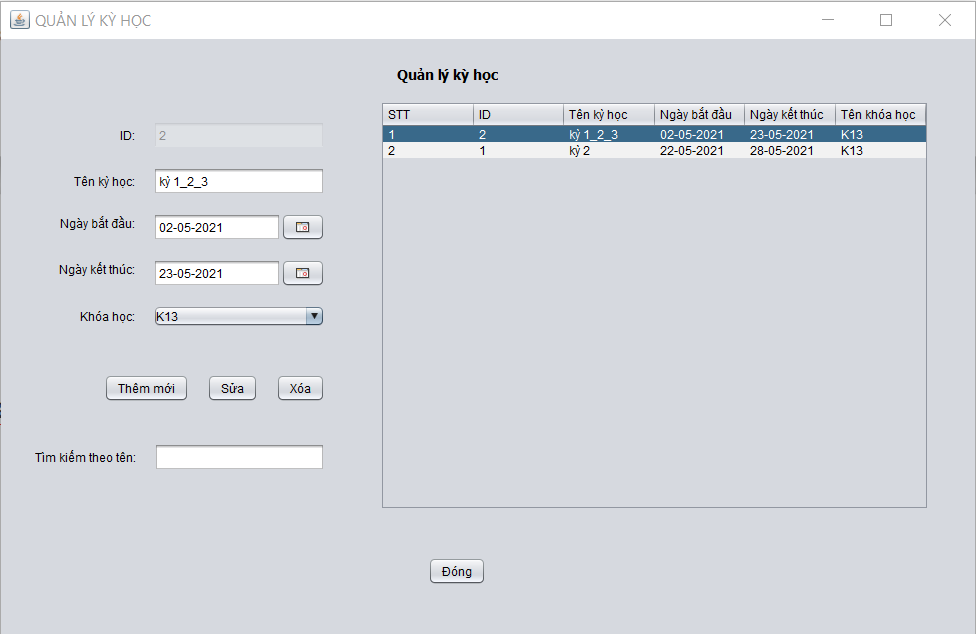
Hình 3.21. Màn hình kết quả thêm mới kỳ học

* Sửa thông tin kỳ học
* Chọn Kỳ học cần sửa, sửa thông tin.

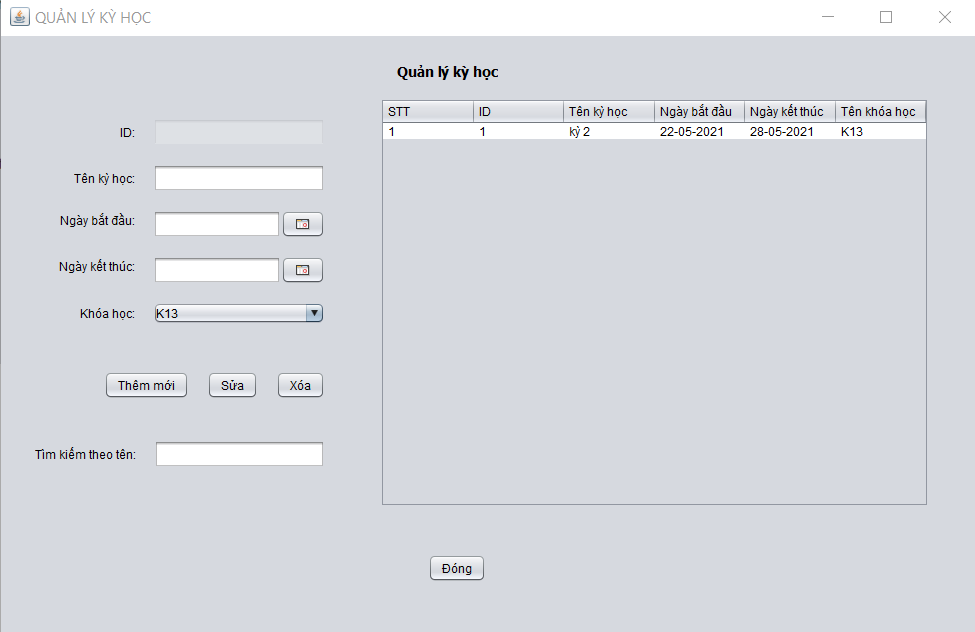
Hình 3.22. Màn hình sửa thông tin kỳ học

* Chọn “Sửa” thông tin được cập nhật trong cơ sở dữ liệu và kết quả:

Hình 3.23. Màn hình kết quả sửa thông tin kỳ học

* Xóa thông tin kỳ học
* Chọn kỳ học cần xóa:

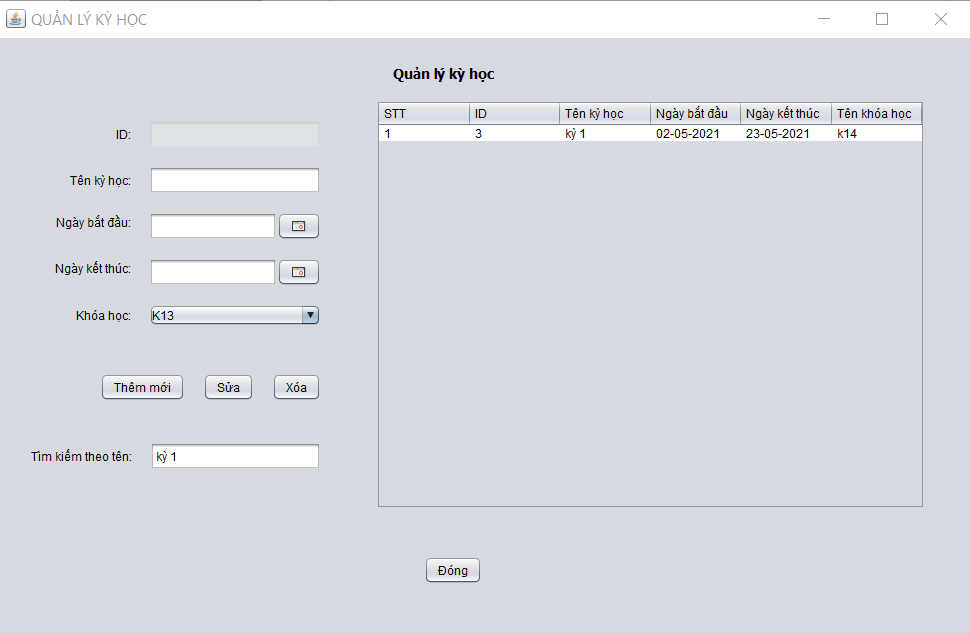
Hình 3.24. Màn hình Xóa thông tin kỳ học

* Chọn nút Xóa:

Hình 3.25. Màn hình kết quả sau khi Xóa thông tin kỳ học

* Tìm kiếm thông tin kỳ học

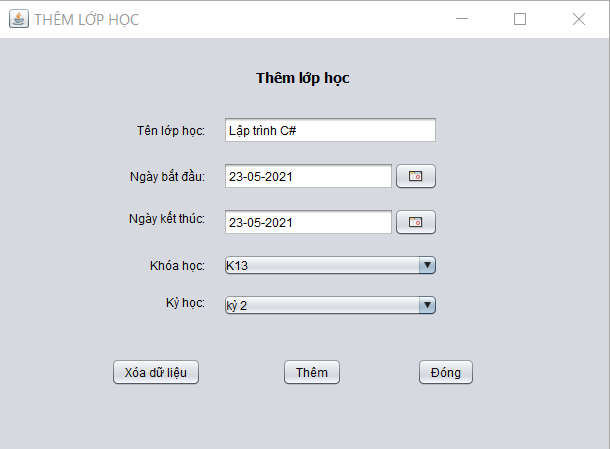
Hình 3.26. Màn hình tìm kiếm thông tin kỳ học

* Bấm phím enter:

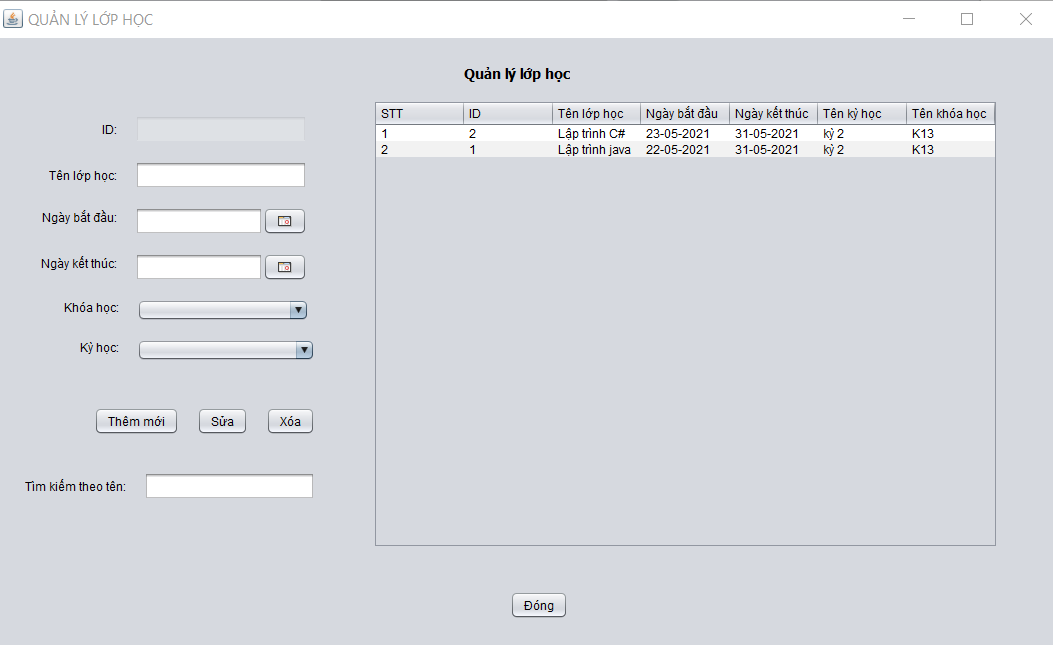
Hình 3.27. Màn hình kết quả tìm kiếm thông tin kỳ học

### Quản lý lớp học

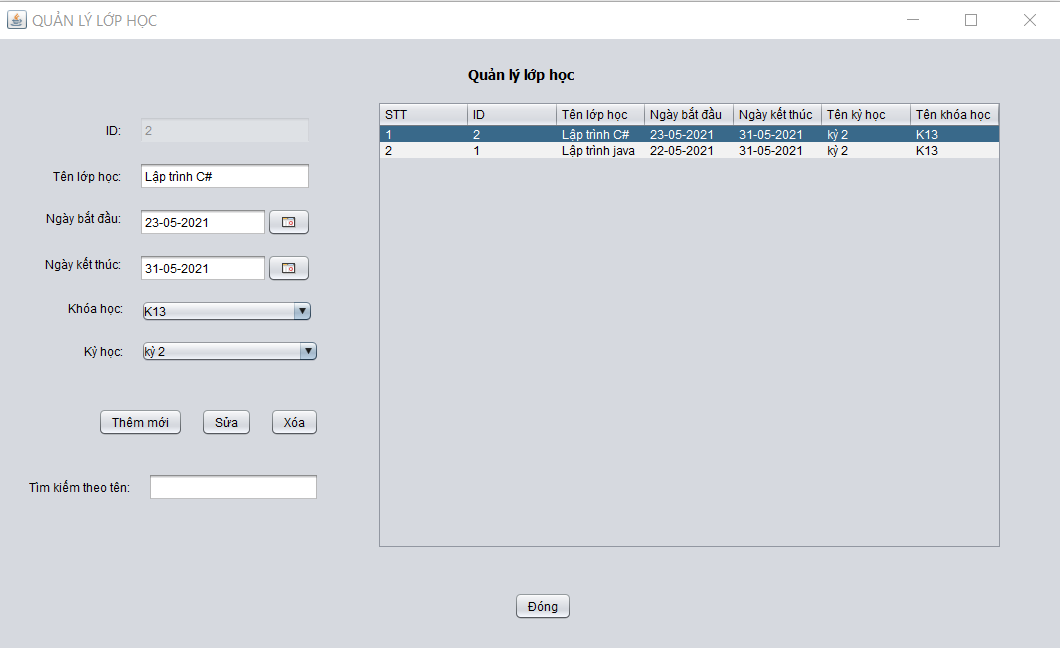
* Thêm lớp học

Chọn “thêm mới” và điền thông tin lớp học.

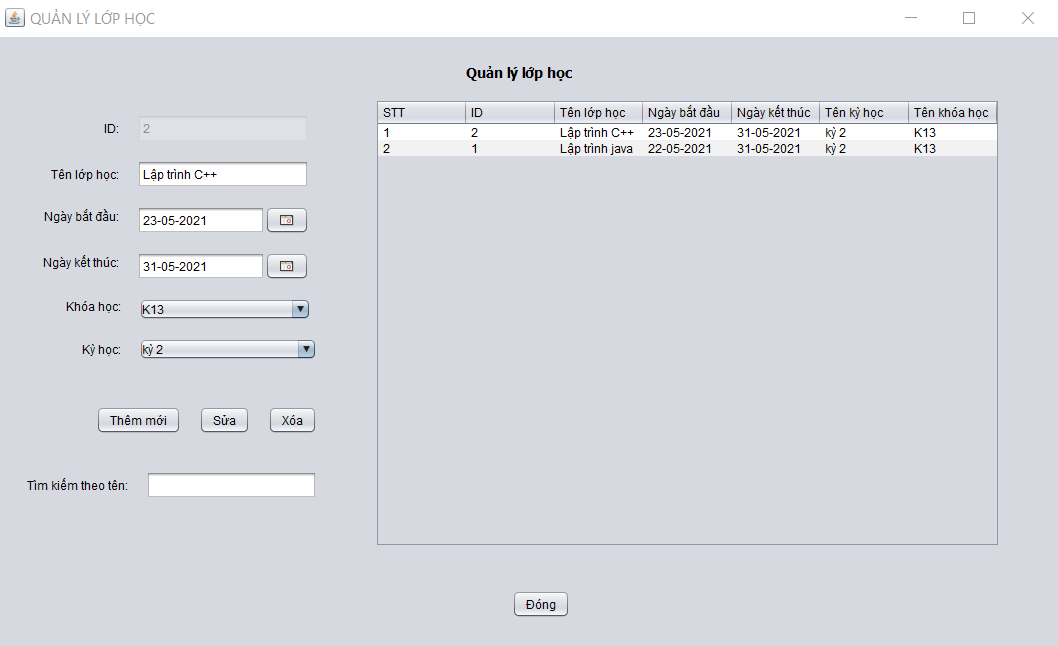
Hình 3.28. Màn hình thêm mới lớp học

* Chọn nút “thêm”

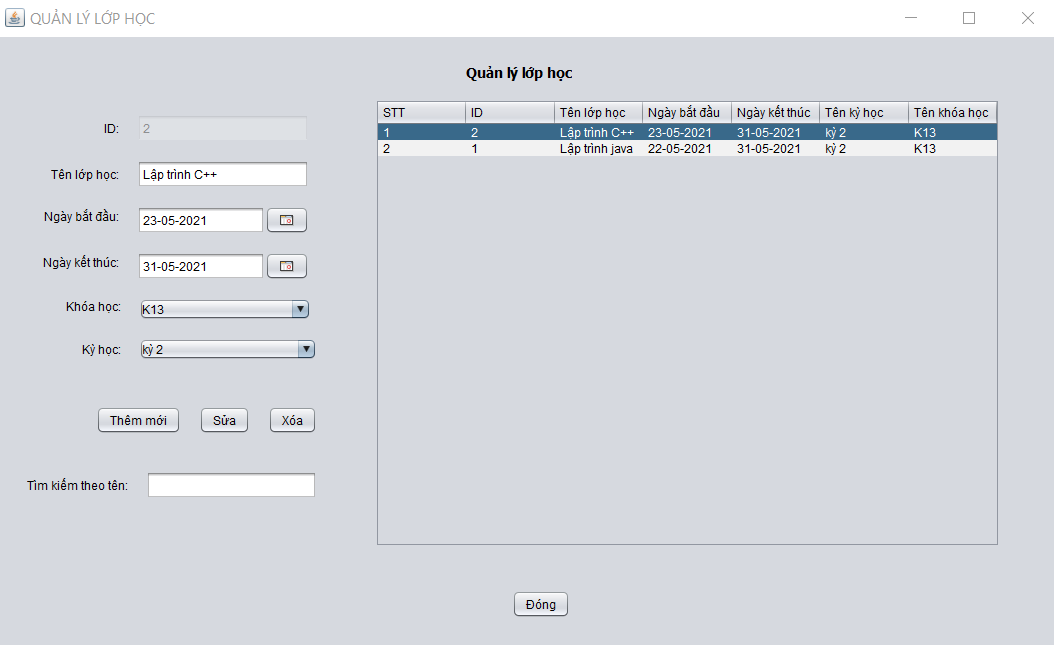
Hình 3.29. Màn hình kết quả thêm mơi lớp học

* Sửa lớp học
* Chọn lớp học cần sửa:

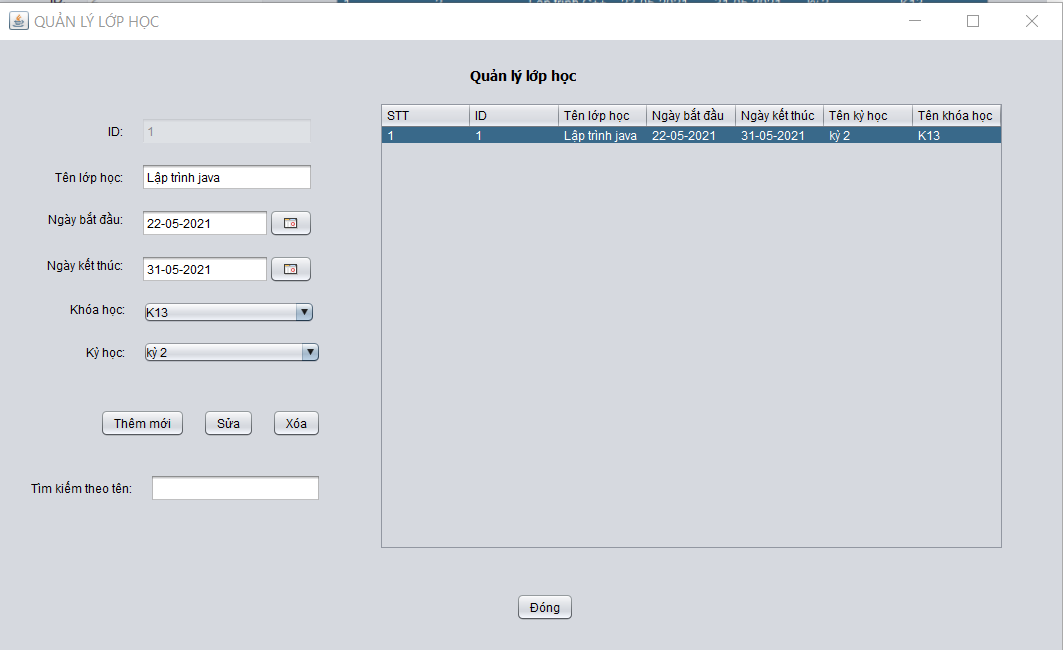
Hình 3.30. Màn hình sửa lớp học

- Nhập thông tin cần sửa và chọn nút Sửa :

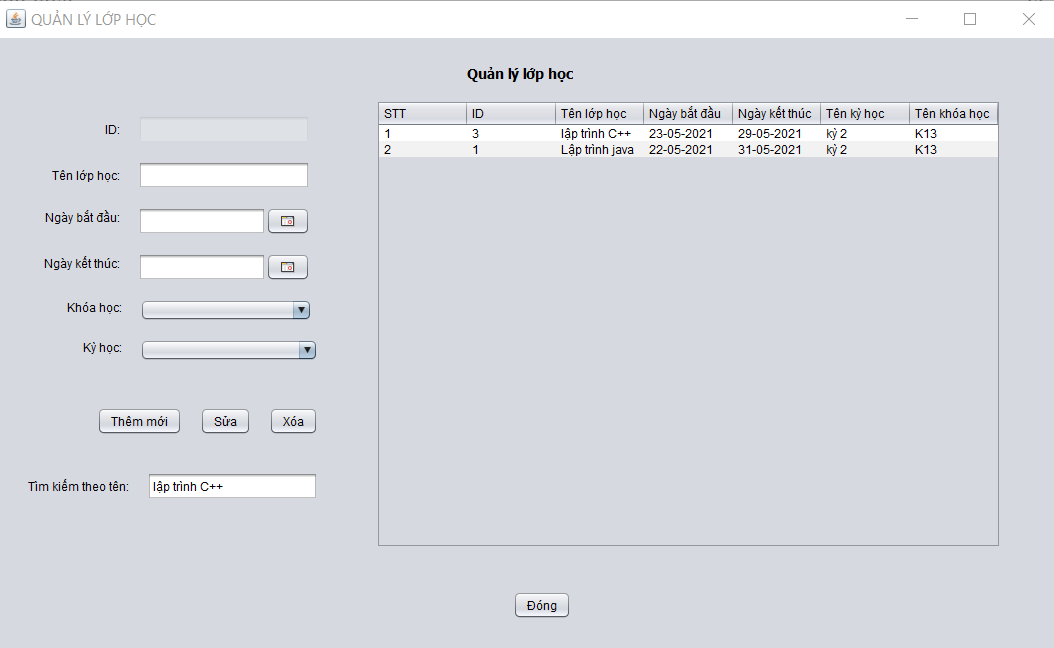
Hình 3.31. Màn hình kết quả sau khi sửa lớp học

* Xóa lớp học
* Chọn lớp học cần xóa:

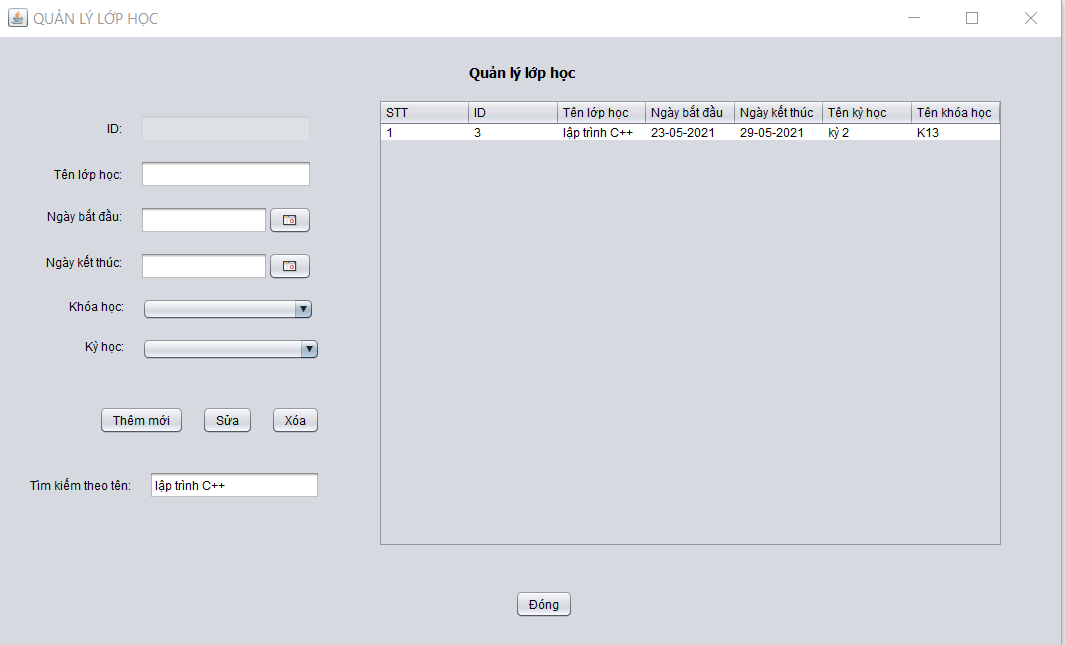
Hình 3.32. Màn hình xóa lớp học

* Chọn nút Xóa

Hình 3.33. Màn hình kết quả sau khi xóa lớp học

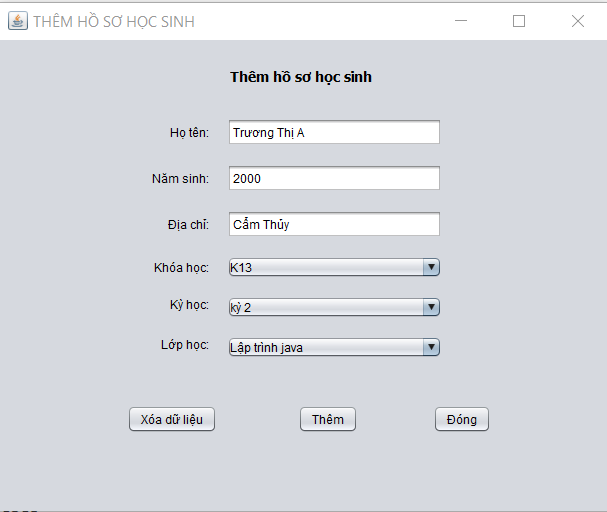
* Tìm kiếm lớp học

Hình 3.34. Màn hình tìm kiếm lớp học

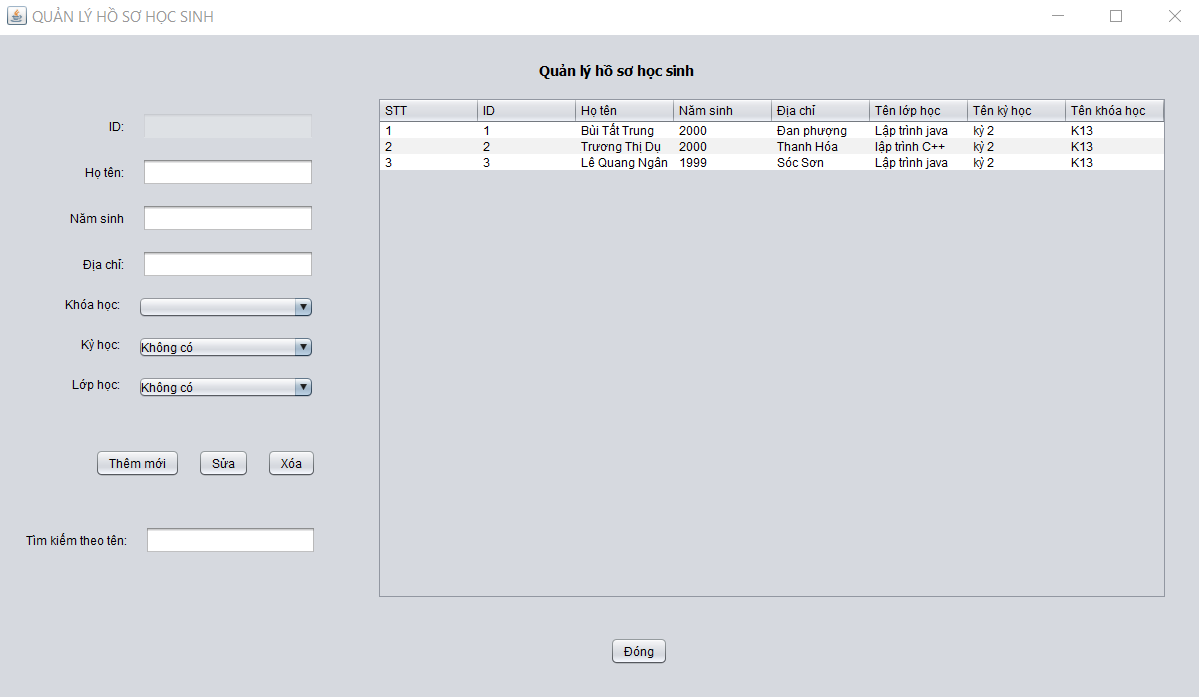
* Bấm phím enter:

Hình 3.35. Màn hình kết quả sau khi tìm kiếm lớp học

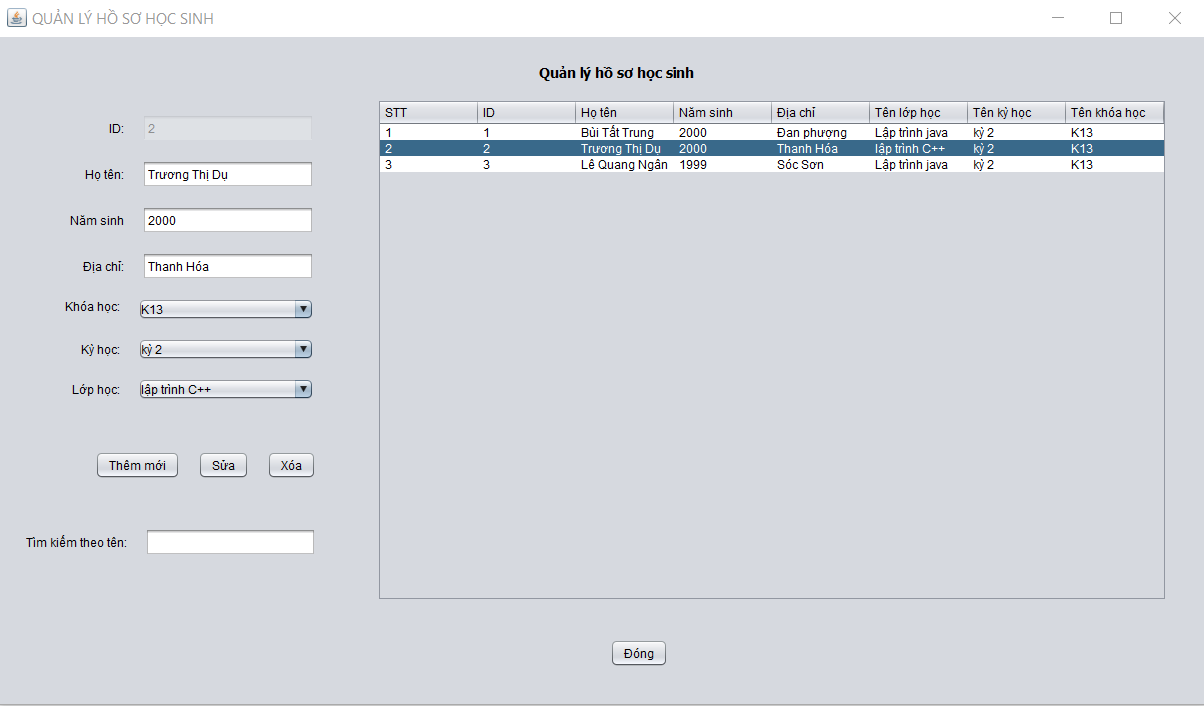
### Quản lý học sinh

* Thêm mới học sinh
* Chọn nút “Thêm mới”

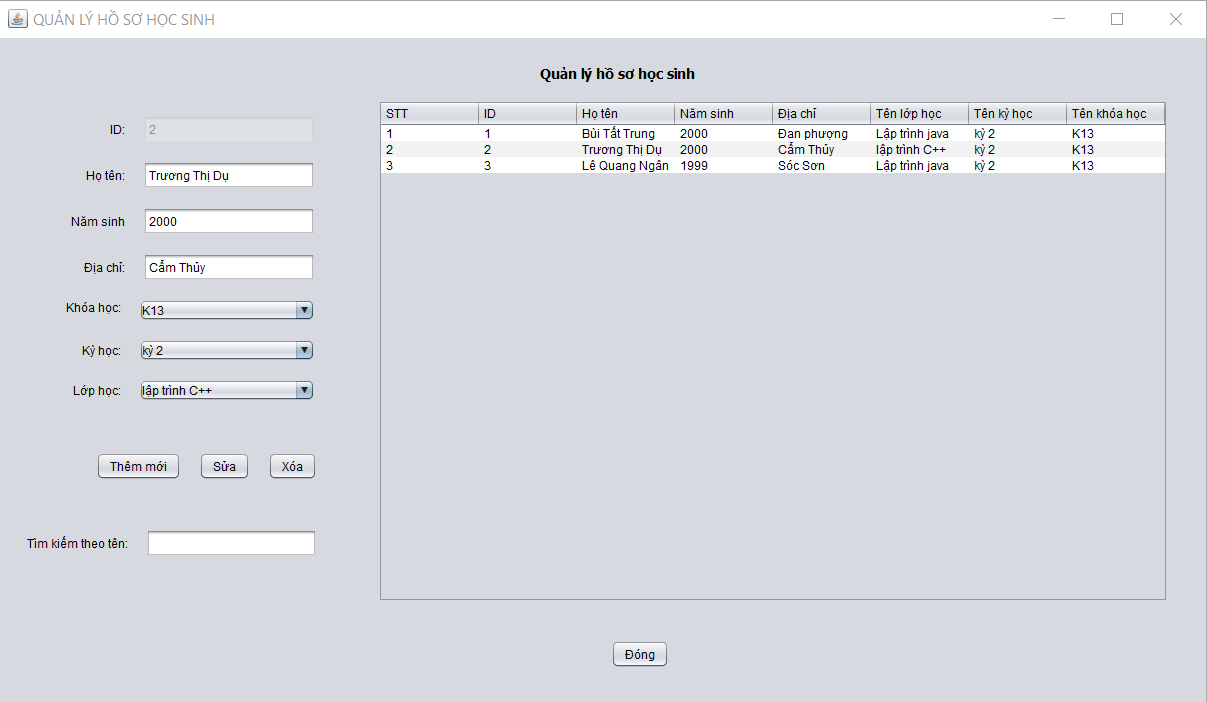
Hình 3.36. Màn hình thêm mới học sinh

* Chọn nút Thêm

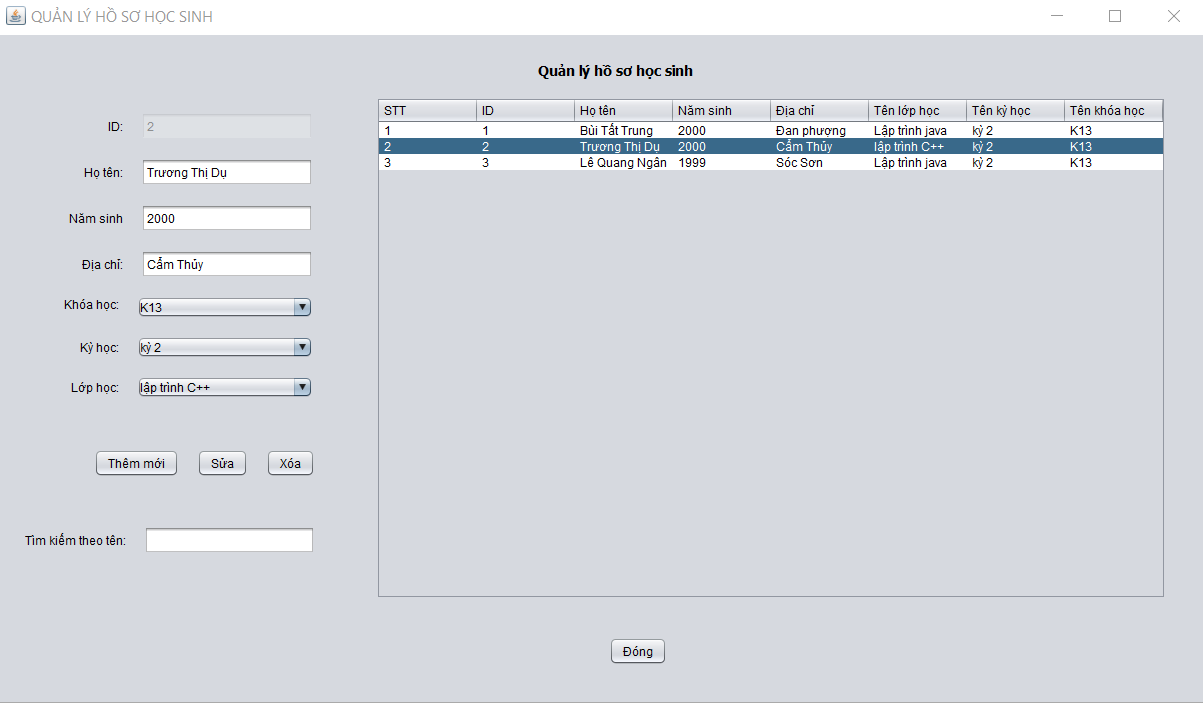
Hình 3.37. Màn hình kết quả sau thêm mới học sinh

* Sửa học sinh
* Chọn học sinh cần sửa

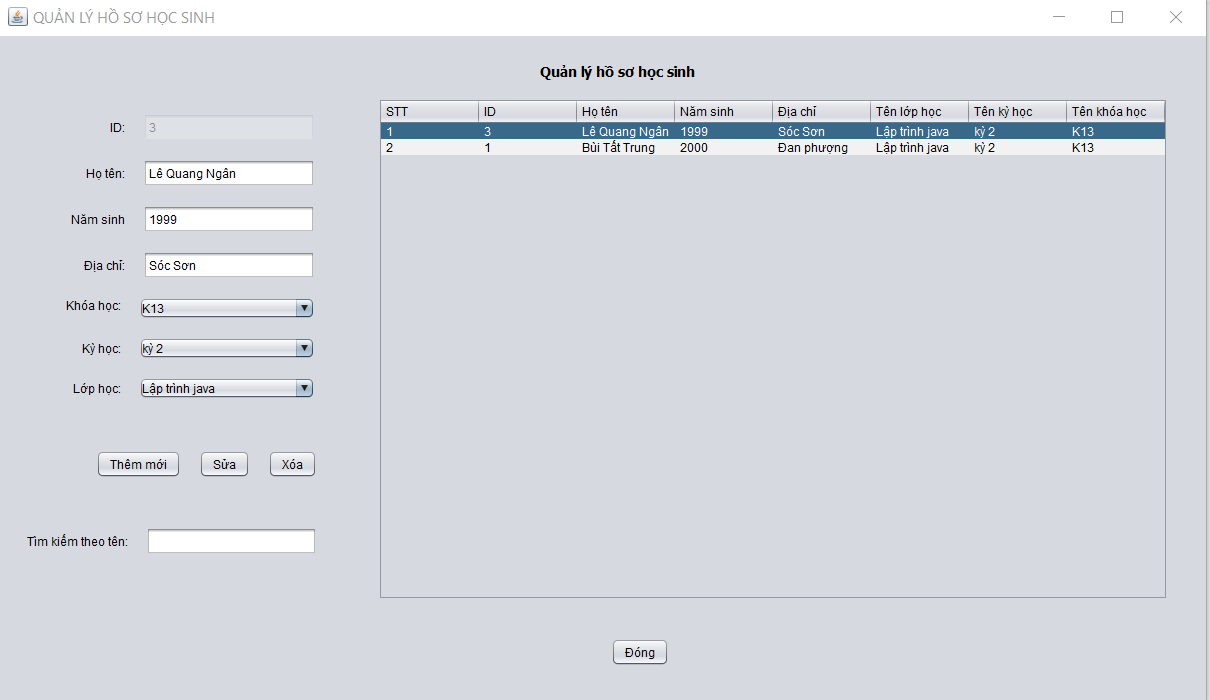
Hình 3.38. Màn hình sửa học sinh

* Chọn nút Sửa

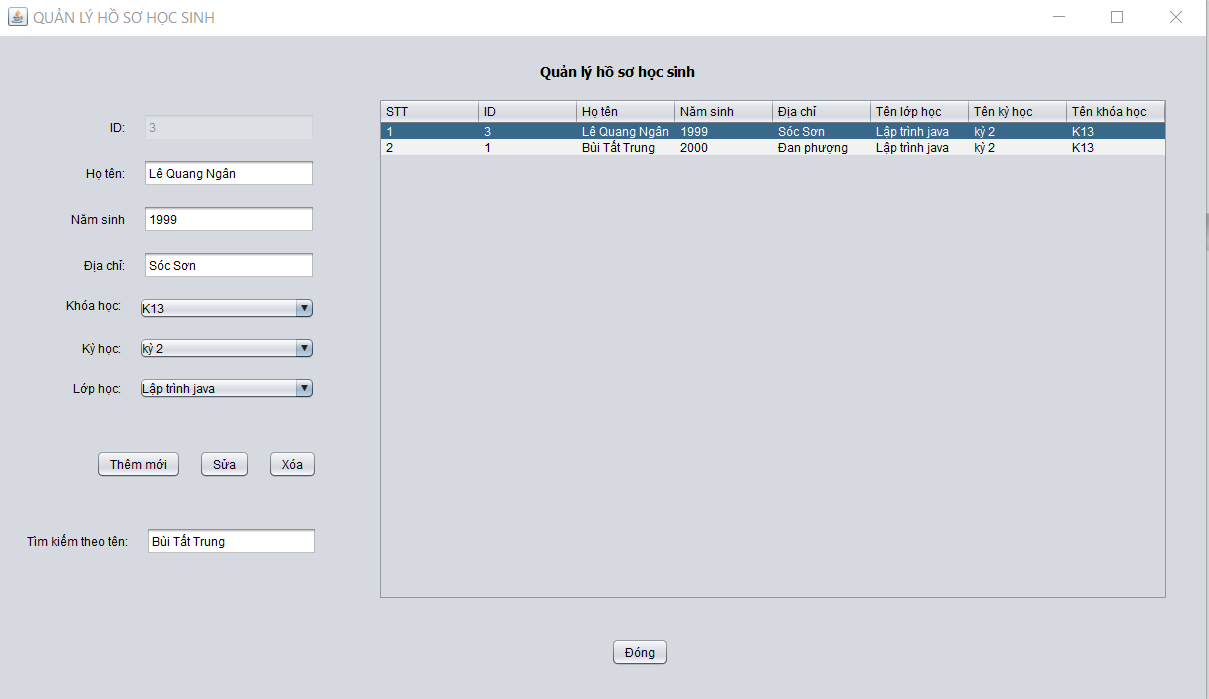
Hình 3.39. Màn hình kết quả sau khi sửa thông tin học sinh

* Xóa học sinh
* Chọn học sinh cần xóa

Hình 3.40. Màn hình Xóa học sinh

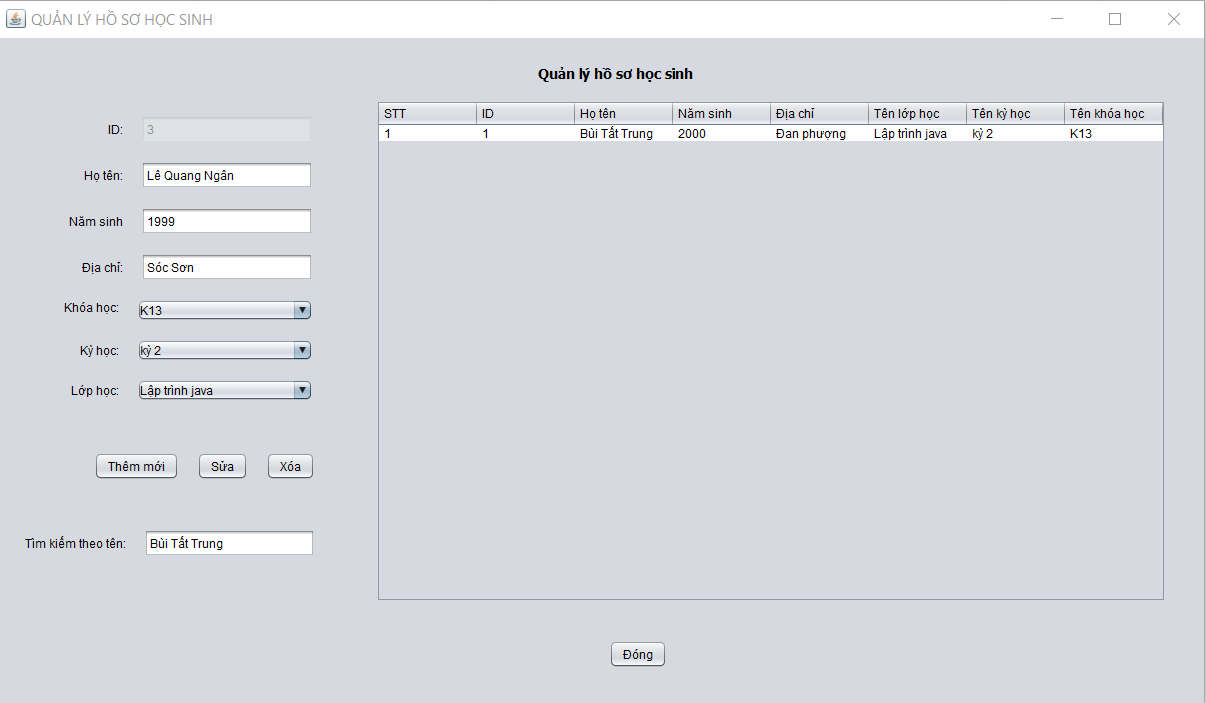
* Chọn nút Xóa

Hình 3.41. Màn hình kết quả sau khi xóa thông tin học sinh

* Tìm kiếm học sinh
* Nhập tên học sinh cần tìm kiếm

Hình 3.42. Màn hình tìm kiếm thông tin học sinh

* Bấm phím enter



Hình 3.43. Màn hình kết quả sau khi Tìm kiếm thông tin học sinh

## Đánh giá

Qua quá trình xây dựng và phát triển phần mềm, hiện nay phần mềm đã được hoàn thiện và có thể chạy bình thường. Các chức năng của phần mềm hoạt động ổn định và đúng với vai trò của nó. Tuy nhiên giao diện phần mềm còn chưa bắt mắt, chức năng chưa phong phú.

**Kết luận**

Trong quá trình xây dựng và phát triển phần mềm, nhóm đã gặp rất nhiều khó khăn trong việc xây dựng phần mềm do còn nhiều hạn chế về kiến thức hệ thống cũng như kiến thức về lĩnh vực hướng tới của phần mềm.

Xong hệ phần mềm đã đáp ứng các nhu cầu cơ bản của việc quản lý hồ sơ học sinh. Nhưng cần bổ sung thêm 1 số tính năng, chỉnh sửa giao diện sao cho đẹp mắt, tối ưu nhất.

Chương trình được kiểm tra rà soát sao cho việc xuất hiện lỗi là tối thiểu, ít nhất. Đảm bảo chương trình hoạt động một cách mượt mà nhất. Tối ưu giao diện và đơn giản hóa giúp cho việc sử dụng dễ dàng hơn.

Mục tiêu đạt được: Hoàn thiện một phần mềm quản lý thông tin học sinh với những chức năng cơ bản, cần thiết nhất thuận tiện cho người dùng. Hạn chế lỗi ở mức tối thiểu.

Giao diện tối giản hóa, dễ sử dụng.

Chưa đạt được: Giao diện cổ điển, chưa bắt mắt, ít chức năng.

Kết thúc việc xây dựng phần mềm, nhóm đã đúc kết được nhiều kinh nghiệm, học hỏi thêm được nhiều kiến thức trong việc phát triển phần mềm bằng ngôn ngữ Java, góp phần hoàn thiện kỹ năng, tuy duy trong xây dựng phần mềm và củng cố kiến thức đã học.

Nhóm xin được cảm ơn sự hỗ trợ, chỉ dẫn của Giảng viên Hà Mạnh Đàotrong quá trình xây dựng phần mềm!

# TÀI LIỆU THAM KHẢO

1. <https://stackoverflow.com/>
2. <https://www.javatpoint.com/>