

**Trường Đại Học Công Nghiệp Hà Nội**

**Khoa công nghệ thông tin**

-----🙞🙜🕮🙞🙜-----

****

**BÁO CÁO THỰC NGHIỆM**

**BỘ MÔN: LẬP TRÌNH JAVA**

|  |  |
| --- | --- |
| **GVHD:**  **Sinh viên:** | ***TS. Hà Mạnh Đào***  **Nguyễn Việt Tiến**  **Trần Thị Tuyết**  **Hoàng Văn Tuyến** |
| **Nhóm:**  **Lớp:** | **23**  **20222IT6019005 Khóa: 15** |

**quản trị TUYỂN SINH TRƯỜNG ĐHCNHN**

Hà Nội, 2023

**MỤC LỤC**

[LỜI MỞ ĐẦU 3](#_Toc26101)

[PHẦN 1: MỞ ĐẦU 4](#_Toc6616)

[Tính cấp thiết của đề tài 4](#_Toc17932)

[Tổng quan đề tài nghiên cứu 4](#_Toc13643)

[PHẦN 2: TỔNG QUAN VỀ ĐỀ TÀI NGHIÊN CỨU 6](#_Toc27050)

[2.1 Giới thiệu 6](#_Toc27944)

[2.2 Khảo sát hệ thống 6](#_Toc19475)

[2.3 Tài liệu đặc tả yêu cầu 8](#_Toc20883)

[2.3.1 Biểu đồ Logic 8](#_Toc32509)

[2.3.2 Biểu đồ use case 10](#_Toc3821)

[2.3.2.1 Use case Tra Cứu Thông Tin Tuyển Sinh 10](#_Toc24775)

[2.3.2.2 Use case Đăng Ký Hồ Sơ 11](#_Toc18347)

[2.3.2.3 Use case Đăng Nhập( Admin) người quản trị 12](#_Toc6485)

[2.3.2.4 Use case quản trị Hồ Sơ 12](#_Toc32325)

[2.3.2.5 Use case quản trị Tài Khoản 14](#_Toc15072)

[2.3.2.6 Use case nhập điểm hồ sơ 16](#_Toc5951)

[2.3.2.7 Use case quản trị Khối Dự Thi 17](#_Toc3646)

[2.3.2.8 Use case thí sinh trúng tuyển 17](#_Toc31962)

[2.3.2.9 Use case Thí sinh vi phạm 18](#_Toc3692)

[2.4 Biểu đồ lớp phân tích 19](#_Toc30962)

[2.4.1 Đăng Ký Hồ Sơ 19](#_Toc30046)

[2.4.2 Đăng Nhập 21](#_Toc16138)

[2.4.3 Quản lý Tài Khoản 21](#_Toc3174)

[2.4.4 Quản lý hồ sơ 22](#_Toc12195)

[2.4.5 Quản lý điểm đầu vào 23](#_Toc17580)

[2.4.6 Quản lý thí sinh vi phạm 24](#_Toc26438)

[2.4.7 Thông báo trúng tuyển 25](#_Toc139)

[2.4.8 Tra cứu thông tin tuyển sinh 26](#_Toc32495)

[2.5 Demo chương trình 27](#_Toc31964)

[2.5.1 Màn hình đăng nhập 27](#_Toc20664)

[2.5.2 Màn hình quản lý sau khi đăng nhập admin 27](#_Toc361)

[2.5.3 Quản lý hồ sơ thí sinh 28](#_Toc27383)

[2.5.4 Quản lý tài khoản 28](#_Toc31261)

[2.5.5 Nhập điểm hồ sơ 29](#_Toc27945)

[2.5.6 Quản lý khối dự thi 29](#_Toc26297)

[2.5.7 Thí sinh trúng tuyển 30](#_Toc9571)

[2.5.8 Thí sinh vi phạm 30](#_Toc15623)

[2.5.9 Đăng ký hồ sơ 31](#_Toc14881)

[2.5.10 Tra cứu thông tin xét tuyển 31](#_Toc31214)

[PHẦN 3: TỔNG KẾT 32](#_Toc19367)

[3.1 Các kết quả đạt được 32](#_Toc16335)

[3.2 Kết luận 33](#_Toc4432)

[TÀI LIỆU THAM KHẢO 34](#_Toc5390)

**LỜI MỞ ĐẦU**

Kể từ những năm 80 của thế kỷ trước, thế giới chính thức bước vào nền văn minh mới, nền văn minh thông tin. Trong nền văn minh này công nghệ thông tin có vai trò đặc biệt ảnh hưởng trực tiếp đến sự phát triển của nền kinh tế. Hiện nay, Việt Nam là nước đang phát triển có tốc độ tăng trưởng cao nhất khu vực và thứ 2 Châu Á, trong những năm qua công nghệ thông tin ở nước ta có những bước phát triển tột bực. Đó là nhờ sự quan tâm của Đảng và Nhà Nước trong lĩnh vữ công nghệ thông tin.

Trong xu hướng tin học toàn cấu hóa và chính sách tin học hóa trong quản trị của nhà nước. Các trường đại học đóng vai trò là một nơi hội tụ những sinh viên mang tầng lớp tri thức đóng góp rất lớn cho tương lai cả nước. và hàng năm, các trường đại học tuyển sinh hơn 2000 sinh viên hệ chính quy… Cho nên nhu cầu tin học hóa trong quản trị là vấn đề đặc biệt cần thiết và quan trọng.

Xuất phát từ nhu cầu thực tế của quá trình tuyển sinh, chúng em xin chọn đề tài nghiên cứu đó là “Xây dựng phần mềm quản trị tuyển sinh” với mục đích hiểu rõ hơn về công tác tuyển sinh và mong muốn đóng góp một phần trong quá trình xây dựng nền văn minh tin học.

**PHẦN 1: MỞ ĐẦU**

**Tính cấp thiết của đề tài**

Bài toán quản trị tuyển sinh vốn luôn là vấn đề nan giải với các trường Đại Học. Trong xu thế phát triển và hội nhập, mục tiêu tin học hóa công tác quản trị luôn được đặt ra bức thiết với các trường trong đó có trường Đại Học Công Nghiệp.

Như hầu hết các trường đại học, mô hình đào tạo hiện nay là đa lĩnh vực và, đa địa điểm, đa cấp và có nhiều phương thức đào tạo khác nhau. Thách thứ lớn nhất đối với các trường đại học hiện nay đó là công tác quản trị tuyển sinh là phải theo kịp sự phát triển của trường khi quy mô tuyển sinh ngày càng tang, còn nguồn lực cán bộ, giảng viên và cơ cấu phòng học hầu như không thay đổi.Công nghệ thông tin giờ đây đã trở thành một phần không thể thiếu với xã hội như hiện nay. Mọi công việc, nghề nghiệp đều cần đến công nghệ thông tin, đặc biết không thể không nhắc đến đó là lĩnh vực quản trị.

**Tổng quan đề tài nghiên cứu**

Nhận thấy nhu cầu trên, việc xây dựng được phần mềm quản trị tuyển sinh cho trường đại học là rất cần thiết. Vì vậy chúng em đã chọn đề tài “Xây dựng phần mềm quản trị tuyển sinh” với mục đích nghiên cứu và xây dựng phần mềm quản trị tuyển sinh để có thể đáp ứng nhu cầu quản trị tuyển sinh cho trường cũng như người quản trị có thể tiết kiệm công sức, thời gian , dễ dàng và tiện lợi hơn trong việc quản lí, vận hành bộ máy tuyển sinh.

Quản trị tuyển sinh là công việc nhằm quản trị thông tin về thí sinh (cụ thể là hồ sơ dự thi và kết quả thi của thí sinh) một cách tổng thể từ lúc nộp hồ sơ đăng ký dự thi cho tới khi trúng tuyển bao gồm tất cả các thông tin về thí sinh như: Họ tên, địa chỉ, số báo danh, số phòng, điểm thi,… của từng thí sinh.

Phần mềm quản trị tuyển sinh tại chức giúp các nhà quản trị (hay phòng đào tạo) trong công việc tổng hợp và đánh giá số lượng thí sinh đăng ký dự thi và chất lượng thí sinh dự thi và trường năm nay so với các năm trước.

quản trị tuyển sinh được thiết kế dựa trên các nguyên tắc cơ bản sau:

* Dựa vào quy chế tuyển sinh do Bộ giáo dục ban hành.
* Dựa vào chỉ tiêu tuyển sinh của trường.
* Dựa vào kinh nghiệm quản trị đào tạo tuyển sinh của trường từ trước đến nay.
* Tìm ra mô hình quản trị đơn giản, dễ hiểu mà chính xác, tốn ít thời gian mà vẫn đảm bảo tính bí mật của các bộ phận đánh phách, ghép phách, chấm thi và trộn bài thi…

**PHẦN 2: TỔNG QUAN VỀ ĐỀ TÀI NGHIÊN CỨU**

**2.1 Giới thiệu**

- Tên bài toán sẽ xây dựng: Xây dựng phần mềm quản trị tuyển sinh.

**-** Hình thức sản phẩm: sản phẩn ứng dụng, chương trình chạy trên nền windows để giải quyết bài toán liên quan đến quản trị tuyển sinh.

**-** Kết quả đạt được: Cài đặt và triển khai phần mềm dựa trên công cụ hỗ trợ viết mã nguồn là Apache Netbeans IDE 13.

Phần mềm Apache Netbeans IDE 13 là phần mềm miễn phí, dễ sử dụng và hỗ trợ nhiều tính năng. Apache Netbeans IDE 13 là một môi trường phát triển tích hợp dùng cho lập trình máy tính. Nó chứa một không gian làm việc cơ sở và một hệ thống plug-in để mở rộng để tùy chỉnh môi trường. Netbeans được viết chủ yếu từ Java và lập trình chủ yếu cho lập trình ứng dụng Java, nhưng cũng có thể dùng để lập trình các ngôn ngữ khác như PHP, asd, C/C++, Hỗ trợ chính thức cho Ruby đã bị loại bỏ từ phiên bản 7.0, …

**2.2 Khảo sát hệ thống**

**2.2.1 Khảo sát sơ bộ**

Đối tượng: Trường đại học Công nghiệp Hà Nội.

Địa chỉ: Số 298 Đường Cầu Diễn, Minh Khai, Bắc Từ Liêm, Hà Nội

Khi một thí sinh nộp hồ sơ thì thông tin sẽ được lưu lại bới bộ phận quản trị hồ sơ. Vì số lượng thí sinh đăng ký thi vào đại học rất lớn nên dẫn đến việc sai sót thông tin có thể xảy ra rất cao, việc làm khá thủ công dẫn đến lực lượng cán bộ nhân sự tương đối lớn của trường đồng thời cũng tốn rất nhiều thời gian và công sức của bộ phận quản thí khi cần truy xuất thông tin của một thí sinh nào đó.

Hệ thống quản lí sinh viên ở trường còn vấp phải sự bất cập giữa các phòng và khoa, nhờ việc sửa đổi thông tin về sinh viên ở khoa, nhưng ở các phòng khác vẫn giữ nguyên thông tin cũ chưa được cập nhật. Hằng năm những phiếu xếp loại rèn luyện và điểm đánh giá rèn luyện của sinh viên lại chất đầy phòng khoa, mất thời gian lưu trữ và tìm kiếm. Muốn tìm thông tin nào đó của sinh viên phải mất rất nhiều thời gian.

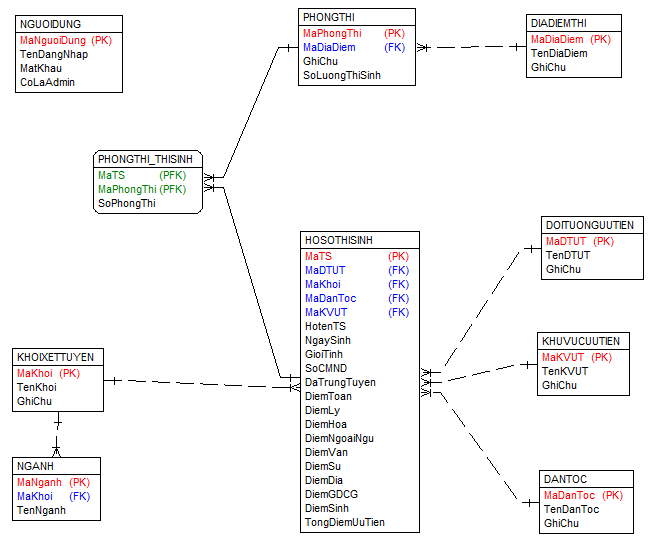
Yêu cầu đặt ra là chương trình phải khắc phục được những hạn chế nêu trên, việc chỉnh sửa thông tin phải được đồng nhất. Quản lí sinh viên cần phải có đầy đủ thông tin của sinh viên theo học ngành nào, khoá nào, khoa nào, đang theo học hình thức nào, quản lí điểm đánh giá rèn luyện và xếp loại rèn luyện của sinh viên.

Để kết quả của quá trình khảo sát được chính xác và khách quan, chúng tôi quyết định bằng phương pháp phỏng vấn trực tiếp bộ phận quản lí thông qua phiếu phỏng vấn dưới đây:

|  |  |
| --- | --- |
| Kế hoạch phỏng vấn | |
| Người phỏng vấn:  Hoàng Văn Tuyến | Người được phỏng vấn:  Đặng Thị Nhung |
| Địa chỉ: Số 298 Đường Cầu Diễn, Minh Khai, Bắc Từ Liêm, Hà Nội | Thời gian hẹn: 25/03/2023  Thời gian bắt đầu: 8h  Thời gian kết thúc: 9h |
| Cần thu nhập thông tin về:  Hồ sơ sinh viên  Điểm thi của sinh viên | Hiểu rõ về quản lí sinh viên và môn học |
| Nội dung:  Giới thiệu tổng quan về dự án  Câu 1:  Nhà trường cần một hệ thống quản lí sinh viên như thế nào?  Câu 2:  Hệ thống cần có những gì ngoài các tính năng cơ bản như nhập thông tin, điểm sinh viên?  Câu 3:  Hệ thống có cần đáp ứng lượng truy cập khi có nhiều người cùng truy cập vào giờ cao điểm không ?  Tổng hợp các nội dung chính ý kiến của người được hỏi  Kết thúc | Dự kiến thời gian;  10 phút  Nhà trường cần một hệ thống dễ nhìn, dễ sử dụng giúp trường có thể quản trị sinh viên dễ dàng hơn  10 phút  Hệ thống có những tính năng xem hồ sơ sinh viên, dân tộc, điểm, đối tượng xét tuyển, ngành học,.. những thông tin để trường nắm bắt được sinh viên tốt hơn  10 phút  Có. Hệ thống cần đáp ứng được yêu cầu khi có nhiều người truy cập vào cùng lúc  10 phút  20 phút   * Dự kiên tổng cộng 60 phút |

**2.3 Tài liệu đặc tả yêu cầu**

**2.3.1 Biểu đồ Logic**



*Hình 2.1 Biểu đồ ERD*

Quy tắc nghiệp vụ:

* Một địa điểm thi có nhiều phòng thi – Một phòng thi chỉ có thể có 1 địa điểm thi.
* Một phòng thi có nhiều thí sinh – Một thí sinh thì thi nhiều phòng thi.
* Một đối tượng ưu tiên có nhiều hồ sơ thí sinh - một hồ sơ thí sinh thì chỉ là 1 đối tượng ưu tiên.
* Một khu vực ưu tiên có nhiều hồ sơ thí sinh – một hồ sơ thí sinh chỉ có 1 khu vực ưu tiên.
* Một dân tộc có nhiều hồ sơ thí sinh – Một hồ sơ thí sinh chỉ có 1 dân tộc.
* Một khối xét tuyển có nhiều ngành – một ngành có 1 khối xét tuyển.

Một khối xét tuyển có nhiều hồ sơ thí sinh – một hồ sơ thí sinh có 1 khối xét tuyển.

**2.3.2 Biểu đồ use case**



*Hình 2.2 Biểu đồ use case*

**2.3.2.1 Use case Tra Cứu Thông Tin Tuyển Sinh**

1. Mô tả: Usecase này cho phép thí sinh xem lại mọi thông tin chi tiết về chính mình.
2. Luồng sự kiện

2.1 Luồng cơ bản

1. Usecase bắt đầu khi người dùng click vào nút “Đăng ký” trên màn hình người dùng. Hệ thống sẽ cho thí sinh nhập thông tin về chính thí sinh đó gồm họ tên, ngày sinh, số CMND, mã dân tộc, đối tượng ưu tiên, khu vực ưu tiên, giới tính. Usecase kết thúc.

2.2 Luồng rẽ nhánh

1. Tại bất kỳ bước nào trong luồng cơ bản nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc.
2. Các yêu cầu đặc biệt: Không có
3. Tiền điều kiện: Không có
4. Hậu điều kiện: Không có
5. Điểm mở rộng: Không có.

**2.3.2.2 Use case Đăng Ký Hồ Sơ**

1. Mô tả: Usecase này cho phép thí sinh xem lại mọi thông tin chi tiết về chính mình.
2. Luồng sự kiện

2.1 Luồng cơ bản

1. Usecase bắt đầu khi người dùng click vào nút “tra cứu” trên màn hình người dùng. Hệ thống sẽ lấy thông tin về chính thí sinh đó gồm mã thí sinh, họ tên thí sinh, ngày sinh, giới tính, số CMND, điểm toán, lý hóa, ngoại ngữ, văn, sử, địa, GDCD, Sinh, và tổng điểm… từ bảng HOSOTHISINH sau đó hiển thị lên màn hình và usecase kết thúc.

2.2 Luồng rẽ nhánh

1. Tại bất kỳ bước nào trong luồng cơ bản nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc.
2. Các yêu cầu đặc biệt: Không có
3. Tiền điều kiện: Không có
4. Hậu điều kiện: Không có
5. Điểm mở rộng: Không có.

**2.3.2.3 Use case Đăng Nhập( Admin) người quản trị**

1. Mô tả: Use case này cho phép người quản trị đăng nhập vào hệ thống phần mềm tuyển sinh.
2. Luồng sự kiện
   1. Luồng cơ bản
3. Usecase này bắt đầu khi người dùng click vào nút “Đăng nhập” trên màn hình người quản trị. Hệ thống sẽ hiển thị màn hình đăng nhập.
4. Người quản trị nhập email và kiểm tra email, mật khẩu nếu đúng thì cho phép đăng nhập vào hệ thống quản trị tuyển sinh. Use case kết thúc.
   1. Luồng rẽ nhánh
5. Tại bước 2 trong luồng cơ bản khi người quản trị nhập email hoặc mật khẩu sai thì hệ thống thông báo lỗi và quay lại bước 2 trong luồng cơ bản.
6. Tại bất kỳ bước nào trong luồng cơ bản nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc.
7. Các yêu cầu đặc biệt: Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.
8. Tiền điều kiện: Người quản trị cần đăng nhập với cai trò quản trị hệ thống trước khi có thể thực hiện use case.
9. Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu.
10. Điểm mở rộng: Không có.

**2.3.2.4 Use case quản trị Hồ Sơ**

1. Mô tả: Use case này cho phép người quản trị xem, thêm, sửa, xóa hồ sơ thí sinh trong bảng HOSOTHISINH.
2. Luồng sự kiện
   1. Luồng cơ bản
3. Use case này bắt đầu khi người quản trị click vào nút “quản trị hồ sơ thí sinh” trên menu quản trị. Hệ thống lấy ra danh sách hồ sơ thí sinh bao gồm mã thí sinh, họ tên, ngày sinh, địa chỉ, số CMND, mã dân tộc, dân tộc, mã dân tộc ưu tiên, mã khu vực ưu tiên, giới tính. Từ bảng HOSOTHISINH trong cơ sở dũ liệu và hiển thị danh sách các hồ sơ lên màn hình
4. Xóa hồ sơ sinh viên

- Người quản trị click vào nút “xóa” trên một hồ sơ bất kỳ trên giao diện quản trị hồ sơ thí sinh. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa.

- Người quản trị click nút “đồng ý” Hệ thống sẽ xóa hồ sơ thí sinh được chọn khỏi bảng HOSOTHISINH. Hệ thống hiển thị danh sách hồ sơ thí sinh đã được cập nhật.

1. Thêm hồ sơ sinh viên

- Người quản trị nhập thông tin thí sinh cần thêm bao gồm mã thí sinh, họ tên, ngày sinh, địa chỉ, số CMND, mã dân tộc, dân tộc, mã dân tộc ưu tiên, mã khu vực ưu tiên, giới tính vào bảng HOSOTHISINH trong cơ sở dữ liệu. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận thêm.

- Người quản trị ấn vào nút “đồng ý” hệ thống sẽ thêm thí sinh đó vào bảng HOSOTHISINH.

1. Sửa hồ sơ sinh viên

- Người quản trị nhấn vào hồ sơ thí sinh cần sửa trên giao diện. Quản trị sẽ sửa thông tin cần sửa đã được hiển thị trên màn hình gồm có Họ và tên, ngày sinh, số CMND, mã dân tộc, mã dân tộc ưu tiên, mã khu vực ưu tiên, giới tính,… từ bảng HOSOTHISINH. Hệ thống sẽ hiển thị một màn hình để xác nhận sửa.

- Người quản trị ấn vào nút “đồng ý” hệ thống sẽ cập nhật lại hồ sơ đó trong bảng HOSOTHISINH. Use case kết thúc.

* 1. Luồng rẽ nhánh

1. Tại bước 2 ý b) trong luồng cơ bản, nếu người quản trị click vào nút
2. thí sinh.
3. Tại bước 2 ý c) trong luồng cơ bản. Nếu quản trị click vào nút “hủy bỏ” hệ thống sẽ hủy bỏ thao tác thêm thí sinh vào bảng HOSOTHISINH.
4. Tại bước 2 ý d) trong luồng cơ bản. Nếu người quản trị nhấn vào nút “hủy bỏ” hệ thống sẽ hủy bỏ thao tác và trả lại giao diện quản trị hồ sơ thí sinh.
5. Tại bất kỳ bước nào trong luồng cơ bản nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc.
6. Các yêu cầu đặc biệt: Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.
7. Tiền điều kiện: Người quản trị cần đăng nhập với cai trò quản trị hệ thống trước khi có thể thực hiện use case.
8. Hậu điều kiện: Nếu use case kết thcus thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu.
9. Điểm mở rộng: Không có.

**2.3.2.5 Use case quản trị Tài Khoản**

1. Mô tả: Use case này cho phép người quản trị xem, thêm, sửa, xóa tài khoản thí sinh và người thuộc bộ phận quản trị.
2. Luồng sự kiện
   1. Luồng cơ bản
3. Use case này bắt đầu khi người quản trị click vào nút “quản trị tài khoản” trên menu quản trị. Hệ thống lấy ra danh sách hồ sơ thí sinh bao gồm tên đăng nhập, mật khẩu, ID từ bảng TAIKHOAN trong cơ sở dũ liệu và hiển thị danh sách các tài khoản trên màn hình. Hệ thống lấy ra danh sách hồ sơ thí sinh bao gồm tên đăng nhập, mật khẩu, ID từ bảng TAIKHOAN trong cơ sở dũ liệu và hiển thị danh sách các tài khoản trên màn hình.
4. Xóa tài khoản

- Người quản trị nhấn vào tài khoản có trên bảng TAIKHOAN đã được hiển thị trên màn hình. Hệ thống hiển thị lên màn hình xác nhận có đồng ý xóa không.

- Người quản trị nhấn vào nút “xác nhận” tài khoản sẽ được xóa trên cơ sở dữ liệu được lấy từ bảng TAIKHOAN.

1. Thêm tài khoản cho người quản trị

- Người quản trị điền thông tin gồm có Tên đăng nhập, mật khẩu và ID vào bảng TAIKHOAN và nhấn vào nút “thêm tài khoản quản trị”, hệ thống hiển thị lên màn hình tài khoản đã được thêm vào bảng TAIKHOAN vào hộp thoại đã có trước đó.

1. Thêm tài khoản cho người dùng

- Người quản trị điền thông tin gồm có Tên đăng nhập, mật khẩu và ID vào bảng TAIKHOAN rồi nhấn vào nút “thêm tài khoản người dùng”, hệ thống hiển thị lên màn hình tài khoản đã được thêm vào bảng TAIKHOAN vào hộp thoại đã có trước đó

1. Sửa tài khoản

- Người quản trị nhấn vào tài khoản cần sửa trên hộp thoại có sẵn đã được hiển thị trên màn hình. Hệ thống hiển thị lên màn hình form thông tin sửa mật khẩu

- Người quản trị nhấn vào nút sửa. Hệ thống cập nhật lại cơ sở dữ liệu trong bảng TAIKHOAN

* 1. Luồng rẽ nhánh

1. Tại bước bước 2 ý b) trong luồng cơ bản, nếu người quản trị nhấn vào nút “hủy bỏ” hệ thống sẽ hủy bỏ thao tác và trả lạ giao diện quản trị tài khoản.
2. Tại bước 1 ý e) trong luồng cơ bản, nếu người quản trị sửa cả tên đăng nhập hoặc ID mà không phải là mật khẩu. Thì thông báo lỗi
3. Tại bất kỳ bước nào trong luồng cơ bản nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc.
4. Các yêu cầu đặc biệt: Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.
5. Tiền điều kiện: Người quản trị cần đăng nhập với cai trò quản trị hệ thống trước khi có thể thực hiện use case.
6. Hậu điều kiện: Người use case kết thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu.
7. Điểm mở rộng: Không có.

**2.3.2.6 Use case nhập điểm hồ sơ**

1. Mô tả: Use case này cho phép người quản trị nhập điểm thi vào hồ sơ thí sinh.
2. Luồng sự kiện
   1. Luồng cơ bản
3. Use case này bắt đầu khi người quản trị nhấn vào nút “Nhập điểm hồ sơ” trên menu quản trị. Hệ thống sẽ hiển thị form nhập gồm mã thí sinh, điểm toán, điểm lý, điểm hóa, điểm sinh, điểm văn, điểm anh, điểm sử, điểm địa, điểm GDCD được lấy ra từ bảng HOSOTHISINH.
4. Nhập điểm

- Khi quản trị đã nhập điểm vào form và nhấn vào nút nhập điểm, hệ thống sẽ lưu điểm đã nhập vào bảng HOSOTHISINH.

1. Tạo lại

- Khi quản trị nhấn vào nút “tạo lại”, form thông tin đã nhập điểm sẽ xóa toàn bộ vào quay lại form trống để nhập lại form mới. Use case kết thúc.

* 1. Luồng rẽ nhánh

1. Tại bước b của luồng cơ bản, nếu nhập mã thí sinh bị sai sẽ không thể thêm điểm vào cơ sở dữ liệu. Hiển thị thông báo lỗi.
2. Tại bất kỳ bước nào trong luồng cơ bản nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc.
3. Các yêu cầu đặc biệt: Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.
4. Tiền điều kiện: Người quản trị cần đăng nhập với cai trò quản trị hệ thống trước khi có thể thực hiện use case.
5. Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu.
6. Điểm mở rộng: Không có

**2.3.2.7 Use case quản trị Khối Dự Thi**

1. Mô tả: Use case này cho phép quản trị sửa khối dự thi.
2. Luồng sự kiện
   1. Luồng cơ bản
3. Người quản trị ấn nhập thông tin vào form có trên màn hình gồm Mã thí sinh và mã khối và ấn vào nút “sửa” hệ thống sẽ sữa mã khối và khối dự thi.
4. Tạo lại

- Người quản trị ấn vào nút “tạo lại”, hệ thống sẽ quay trở lại dữ liệu ban đầu đã có sẵn trong cơ sở dữ liệu. Use case kết thúc.

* 1. Luồng rẽ nhánh

1. Tại bước a) ở luồng cơ bản nếu nhập mã thí sinh sai hoặc mã khối sai. Hiển thị thông báo lỗi.
2. Tại bất kỳ bước nào trong luồng cơ bản nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc.
3. Các yêu cầu đặc biệt: Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.
4. Tiền điều kiện: Người quản trị cần đăng nhập với cai trò quản trị hệ thống trước khi có thể thực hiện use case.
5. Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu.
6. Điểm mở rộng: Không có.

**2.3.2.8 Use case thí sinh trúng tuyển**

1. Mô tả: Use case cho phép xem thí sinh đã trúng tuyển.
2. Luồng sự kiện
   1. Luồng cơ bản
3. Use case bắt đầu khi người quản trị nhấn vào nút “thí sinh trúng tuyển” được hiển thị trên menu quản trị. Hệ thống sẽ hiển thị form thí sinh trúng tuyển trên màn hình gồm mã thí sinh, họ tên, trạng thái và khối xét tuyển.
4. Cập nhật hồ sơ trúng tuyển

- Quản trị nhấn vào một thí sinh được hiển thị trên hộp thoại. Điền khối xét tuyển phù hợp với dữ liệu và nhấn “cập nhật sách trúng tuyển”. Hệ thống sẽ cập nhật lại thí sinh đó và lưu vào bảng HOSOTHISINH.

1. Hủy điều kiện xét tuyển

- Quản trị nhấn vào một thí sinh được hiển thị trên hộp thoại. và nhấn nút “hủy điều kiện xét tuyển”. Hệ thống hiện hộp thoại xác nhận hủy điều kiện xét tuyển.

- Quản trị nhấn nút “đồng ý”, thí sinh đó sẽ hủy điều kiện xét tuyển và xóa khỏi bảng HOSOTHISINH.

* 1. Luồng rẽ nhánh

1. Tại bước 2 ý c) nếu nhấn nút “hủy bỏ” thì hệ thống sẽ quay trở lại dữ liệu ban đầu đã có sẵn trong cơ sở dữ liệu.
2. Tại bất kỳ bước nào trong luồng cơ bản nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc.
3. Các yêu cầu đặc biệt: Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.
4. Tiền điều kiện: Người quản trị cần đăng nhập với cai trò quản trị hệ thống trước khi có thể thực hiện use case.
5. Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu.
6. Điểm mở rộng: Không có.

**2.3.2.9 Use case Thí sinh vi phạm**

1. Mô tả Use case cho phép xem và nhập thí sinh đã vi phạm.
2. Luồng sự kiện
   1. Luồng cơ bản
3. Use case bắt đầu khi người quản trị nhất vào nút “thí sinh vi phạm”, hệ thống sẽ hiển thị form gồm mã thí tinh, họ tên, số CMND, trang thái đươc lấy từ bảng HOSOTHISINH.
4. Thêm thí sinh vi phạm

- Quản trị nhập mã thí sinh vào form đã được hiển thị trên màn hình. Sau đó nhấn nút “thêm”. Hệ thống sẽ cập nhật thí sinh đó trong cơ sở dữ liệu và hiển thị ra màn hình.

1. Tạo lại

- Người quản trị ấn vào nút “tạo lại”, hệ thống sẽ quay trở lại dữ liệu ban đầu đã có sẵn trong cơ sở dữ liệu.Use case kết thúc.

* 1. Luồng rẽ nhánh

1. Tại bất kỳ bước nào trong luồng cơ bản nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc.
2. Các yêu cầu đặc biệt: Use case này chỉ cho phép một số vai trò như người quản trị, người chủ hệ thống thực hiện.
3. Tiền điều kiện:Người quản trị cần đăng nhập với cai trò quản trị hệ thống trước khi có thể thực hiện use case.
4. Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về tài khoản sẽ được cập nhật trong cơ sở dữ liệu.
5. Điểm mở rộng: Không có.

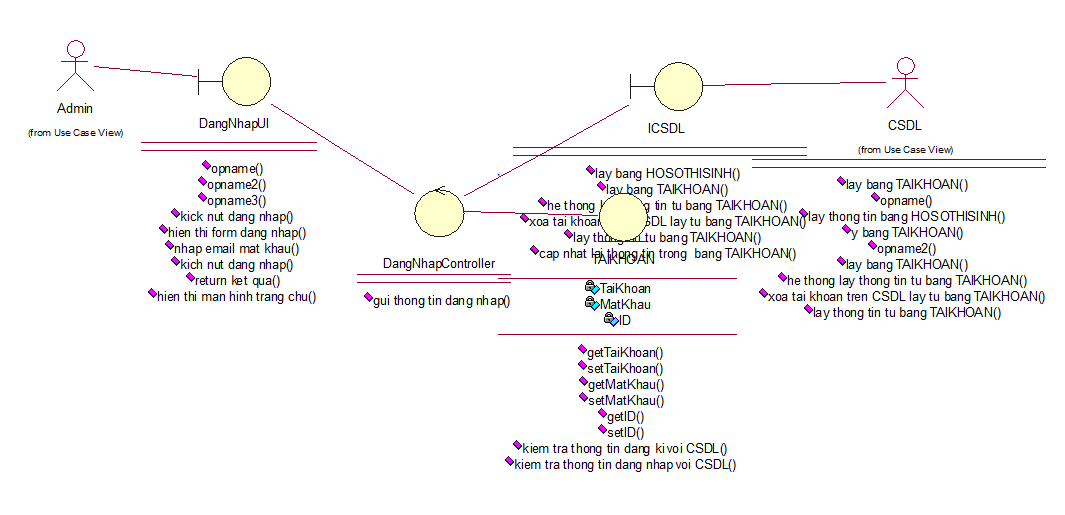
**2.4 Biểu đồ lớp phân tích**

**2.4.1 Đăng Ký Hồ Sơ**



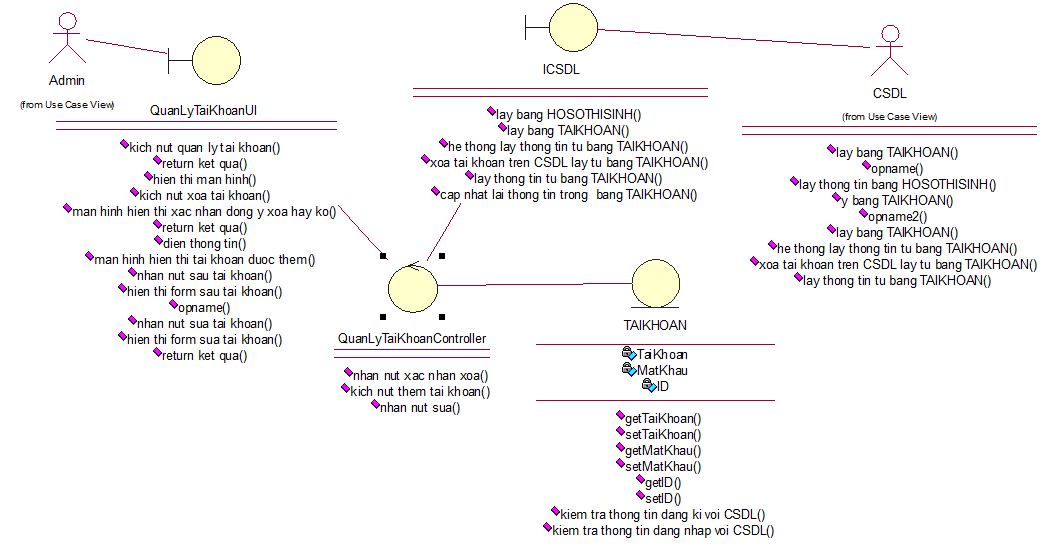
*Hình 2.3 Biểu đồ phân tích use case Đăng Ký Hồ Sơ*

**2.4.2 Đăng Nhập**



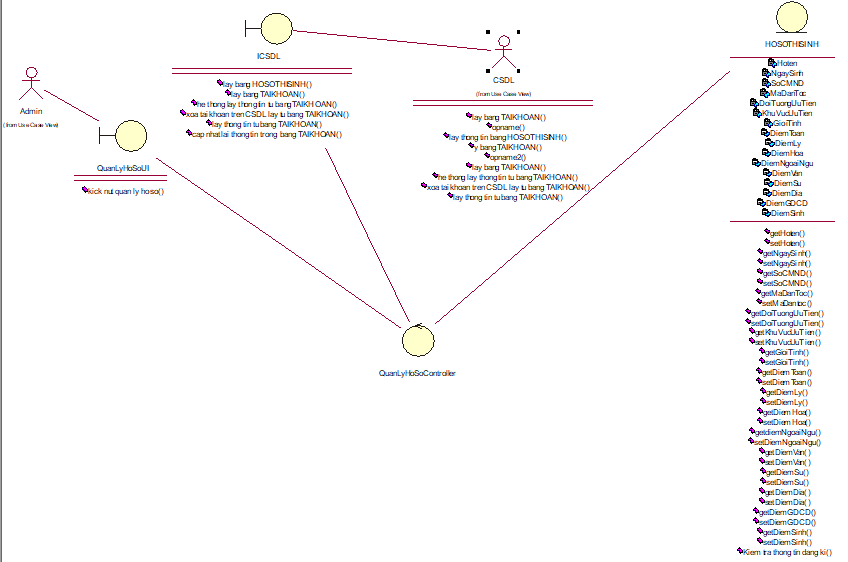
*Hình 2.4 Biểu đồ phân tích use case Đăng Nhập*

**2.4.3 Quản lý Tài Khoản**



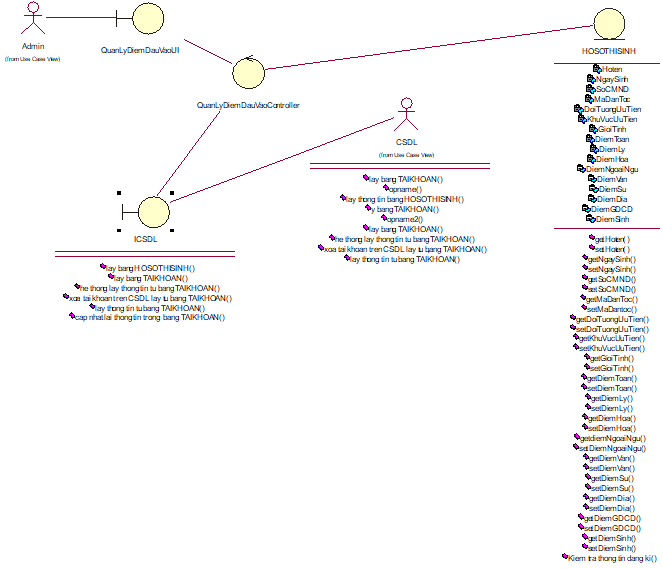
*Hình 2.5 Biểu đồ phân tích use case Quản Lý Tài Khoản*

**2.4.4 Quản lý hồ sơ**



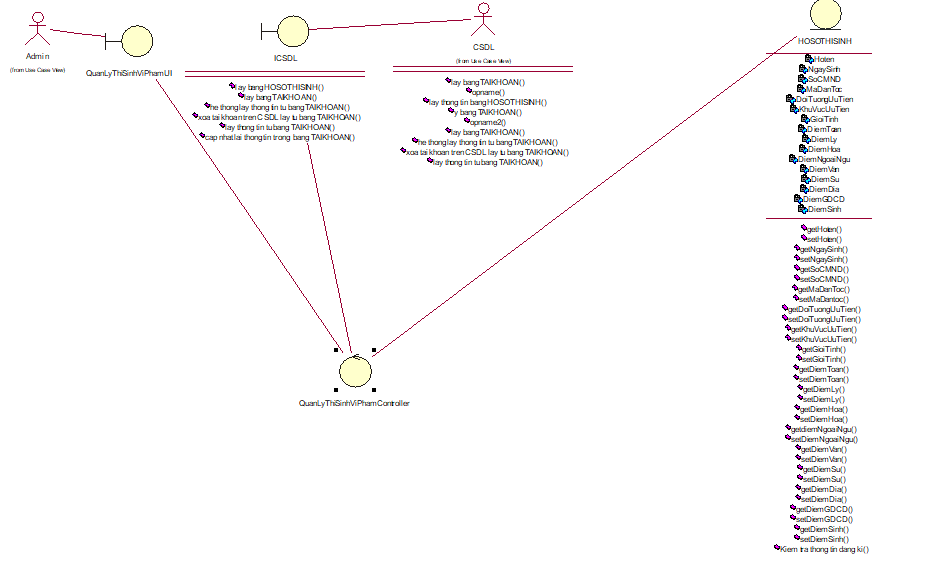
*Hình 2.6 Biểu đồ phân tích use case Quản Lý Hồ Sơ*

**2.4.5 Quản lý điểm đầu vào**



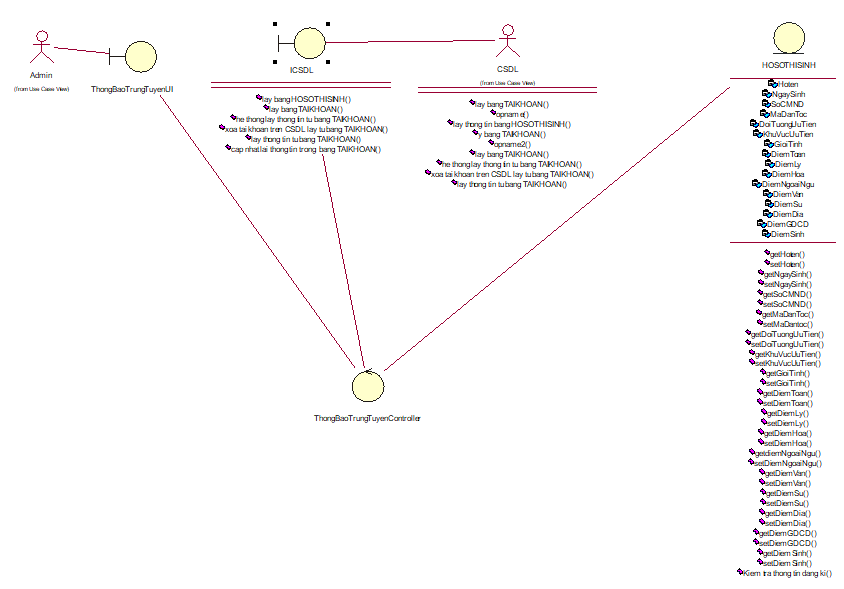
*Hình 2.7 Biểu đồ phân tích use case Quản Lý Điểm Đầu Vào*

**2.4.6 Quản lý thí sinh vi phạm**



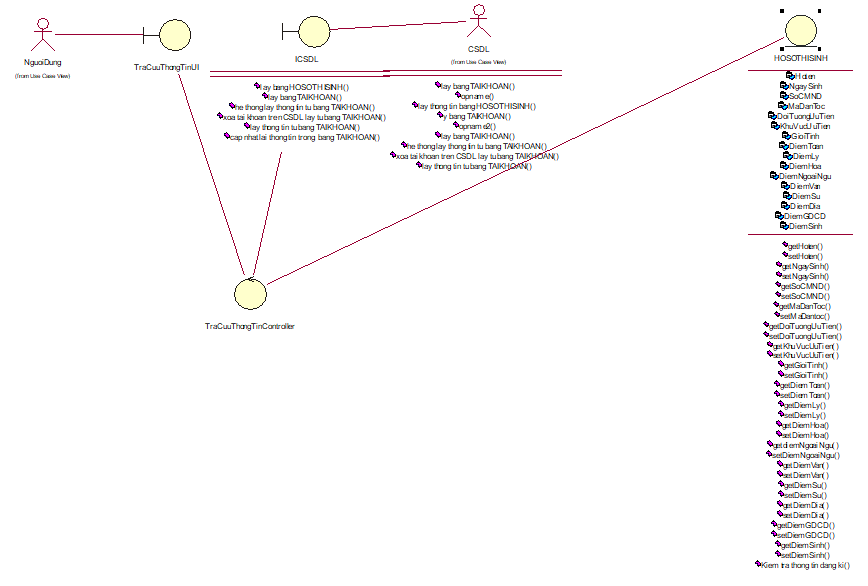
*Hình 2.8 Biểu đồ phân tích use case Quản Lý Thí Sinh Vi Phạm*

**2.4.7 Thông báo trúng tuyển**



*Hình 2.9 Biểu đồ phân tích use case Thong Báo Trúng Tuyển*

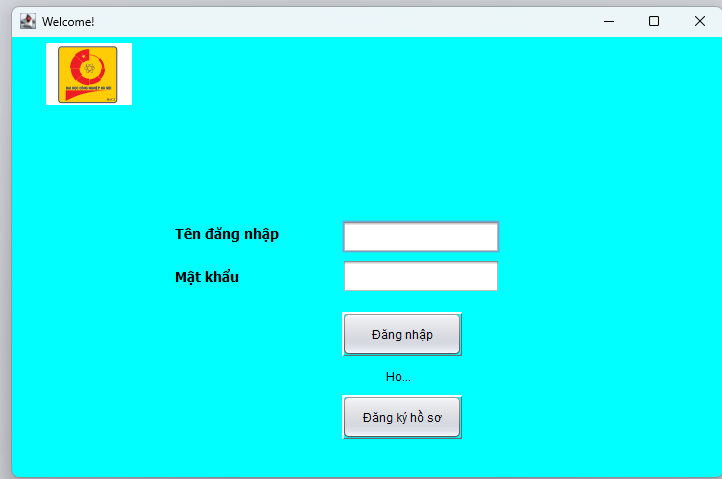
**2.4.8 Tra cứu thông tin tuyển sinh**



*Hình 2.10 Biểu đồ phân tích use case Tra Cứu Thông Tin Tuyển Sinh*

**2.5 Demo chương trình**

### 2.5.1 Màn hình đăng nhập



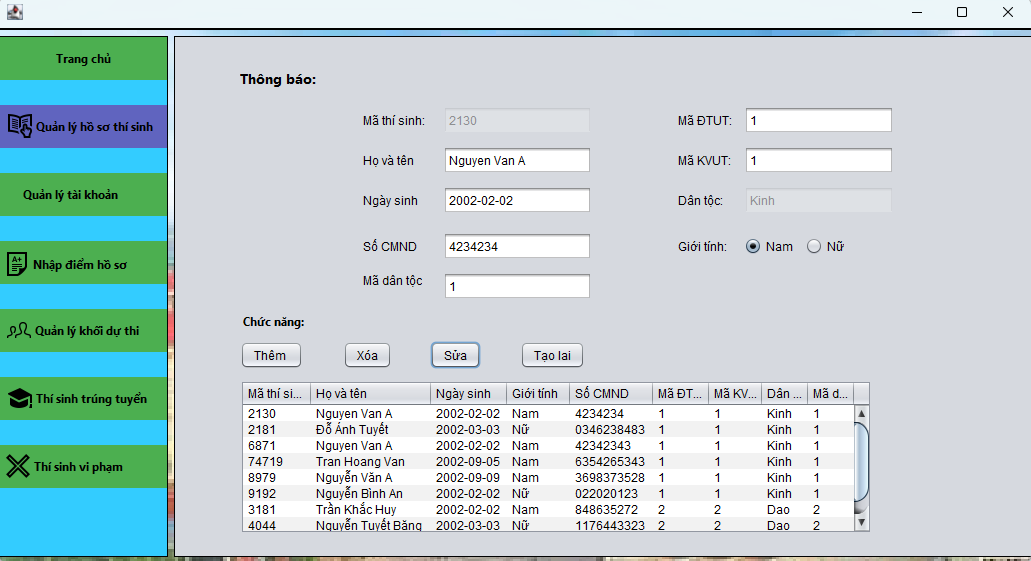
*Hình 2.11 Màn hình Đăng Nhập*

### 2.5.2 Màn hình quản lý sau khi đăng nhập admin



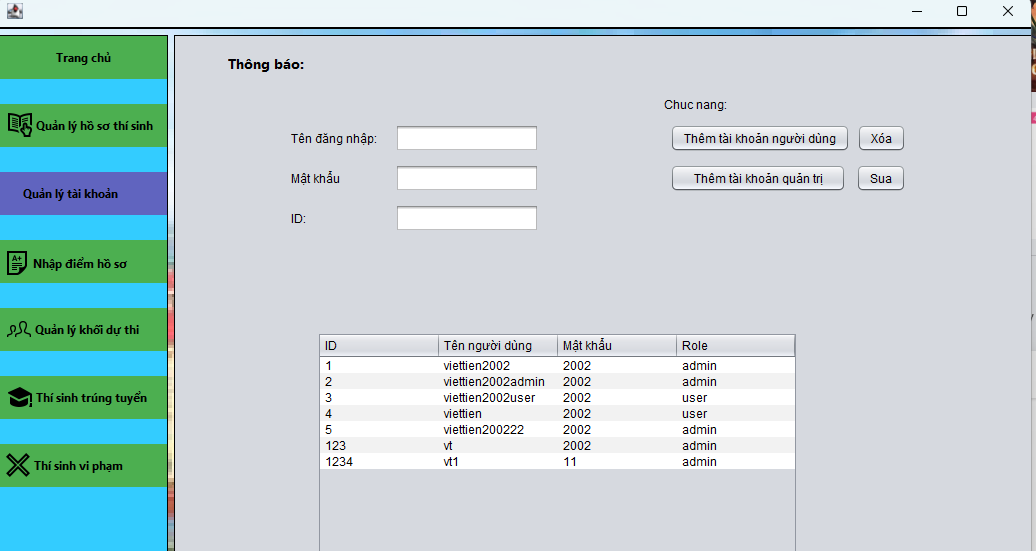
*Hình 2.12 Màn hình quản lý sau khi đăng nhập Admin*

### 2.5.3 Quản lý hồ sơ thí sinh



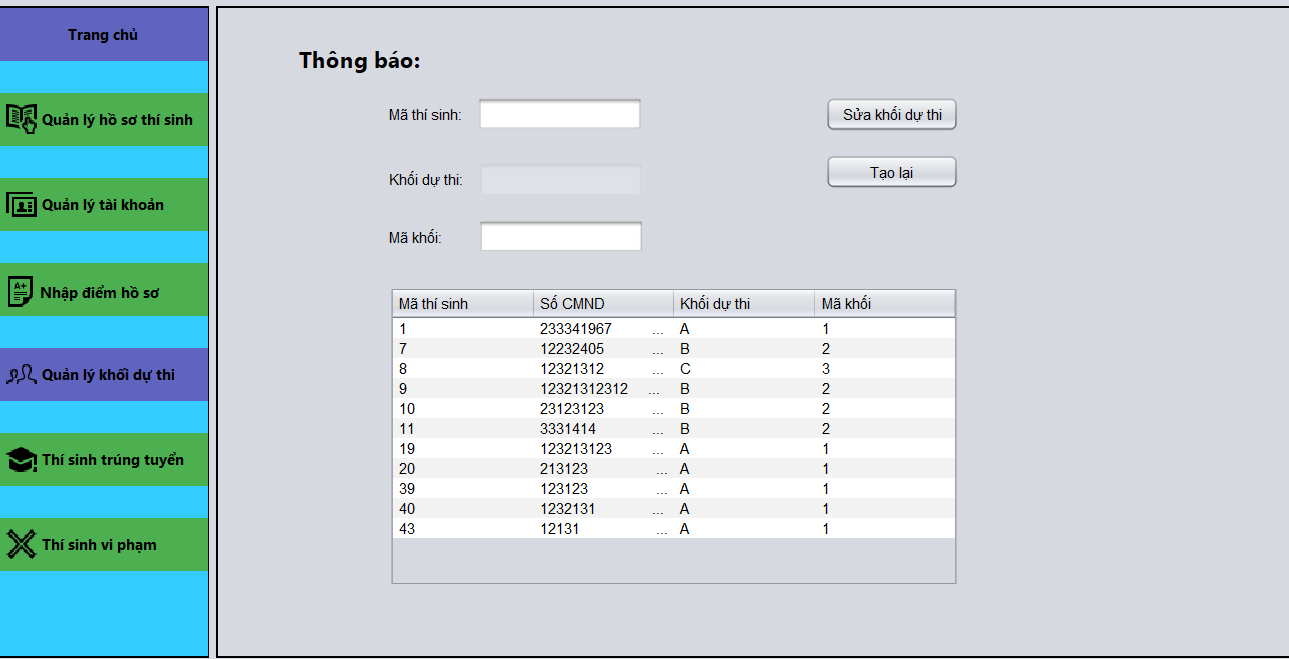
*Hình 2.13 Màn hình Quản Lý Hồ Sơ Thí Sinh*

### 2.5.4 Quản lý tài khoản

****

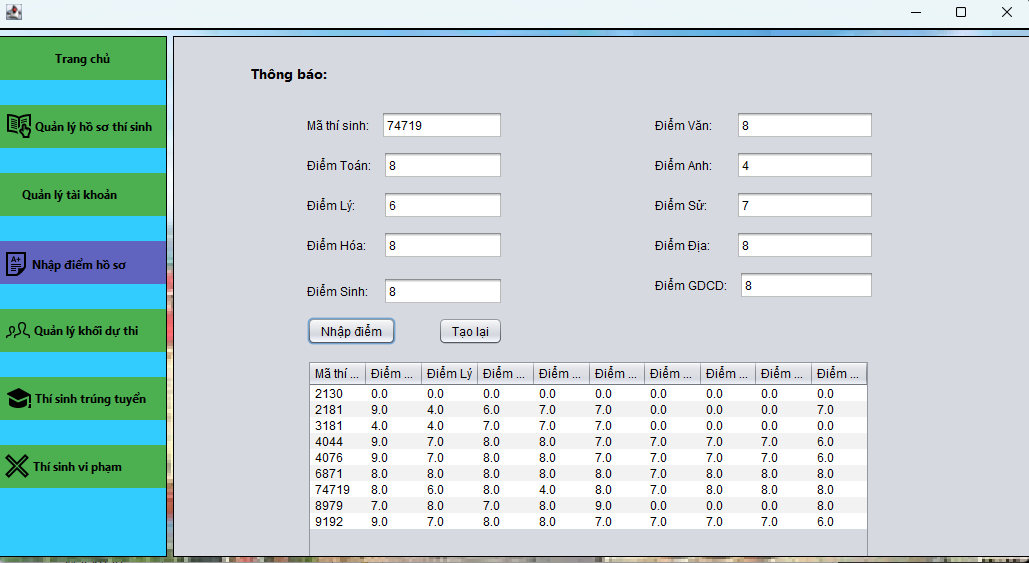
*Hình 2.14 Màn hình Quản Lý Tài Khoán*

### 2.5.5 Quản lý khối dự thi



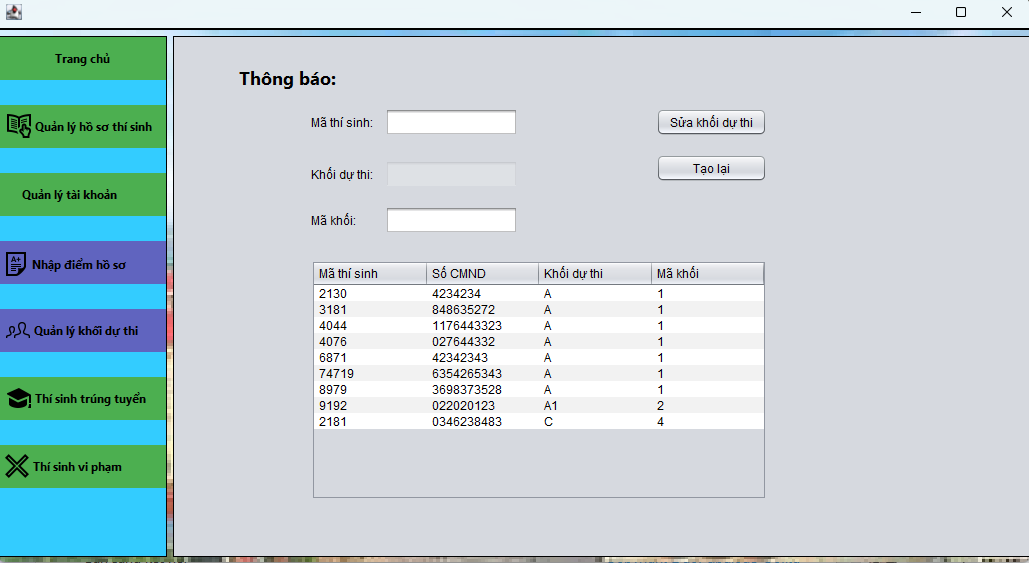
*Hình 2.16 Màn hình Quản Lý Khối Dự Thi*

### 2.5.6 Nhập điểm hồ sơ



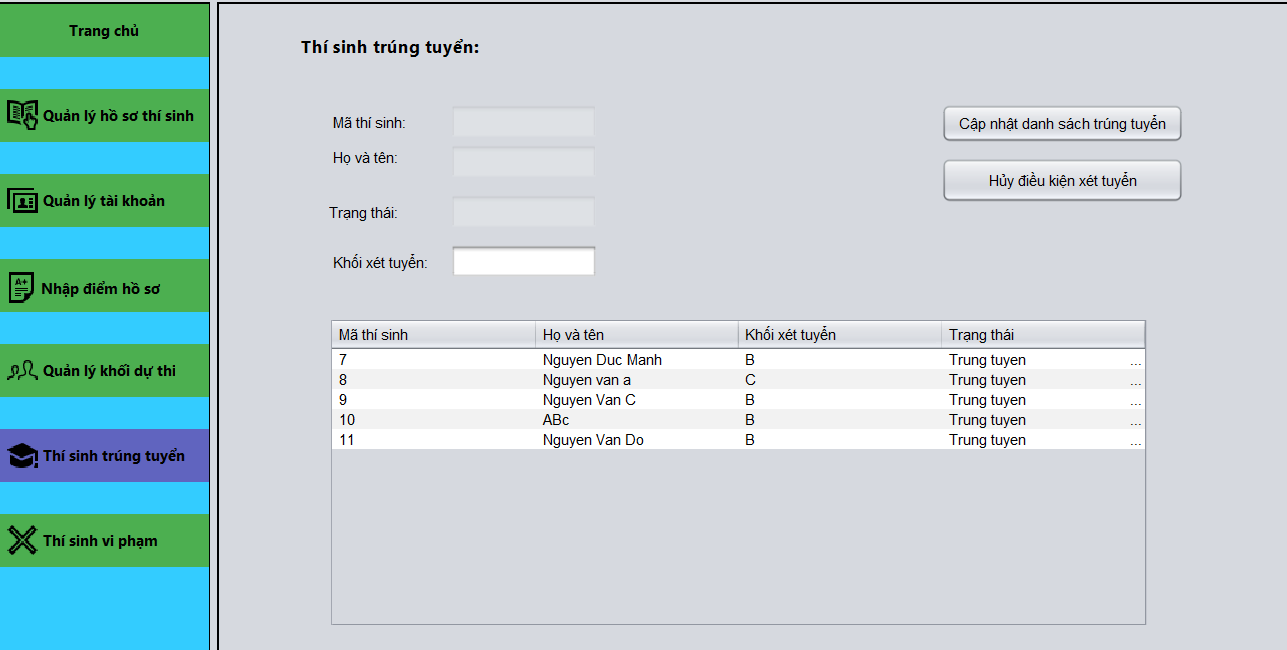
*Hình 2.17: Nhập điểm hồ sơ*

### 2.5.7 Thí sinh trúng tuyển

**

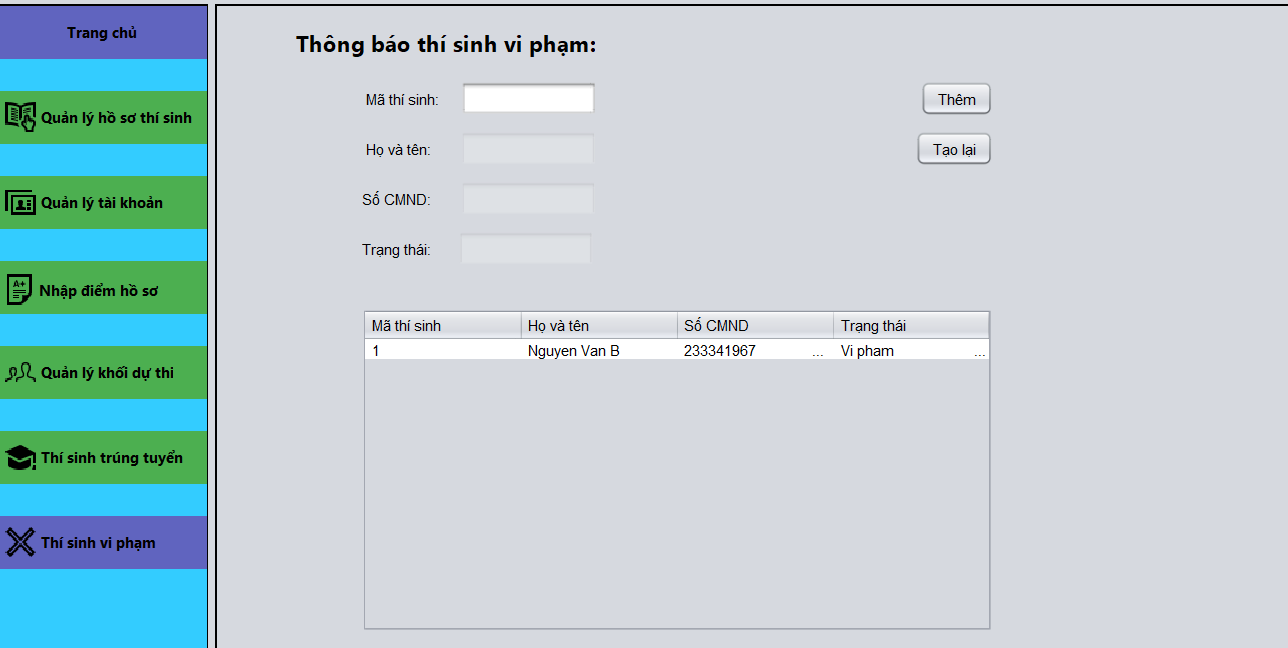
*Hình 2,18: Quản lý khối dự thi*

### 2.5.8 Thí sinh trúng tuyển



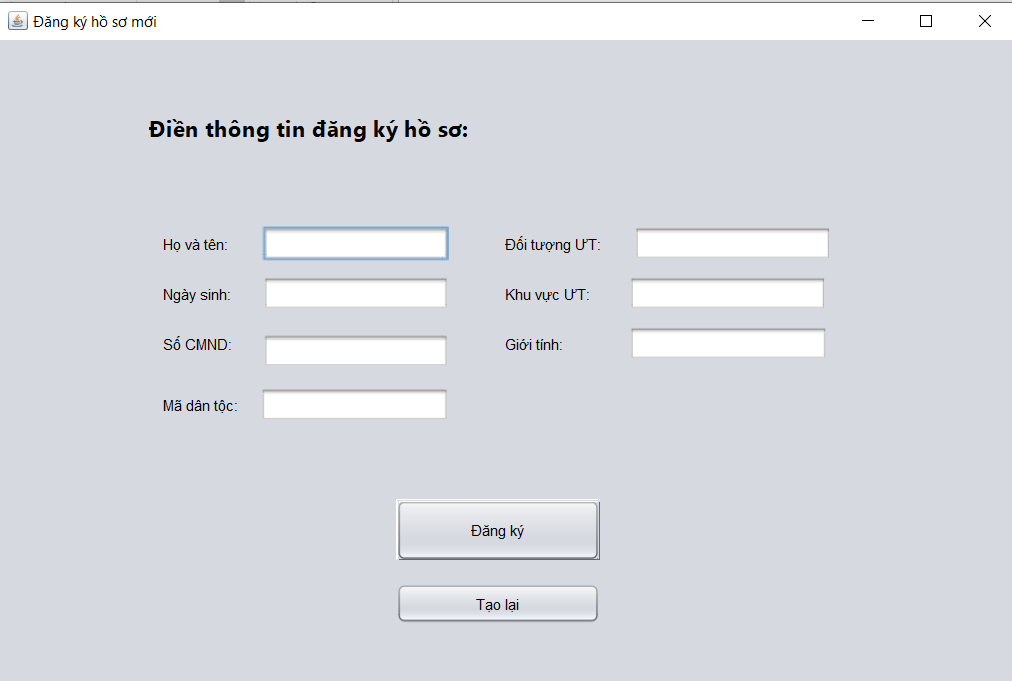
*Hình 2.19 Màn hình Thí Sinh Trúng Tuyển*

### 2.5.9 Thí sinh vi phạm



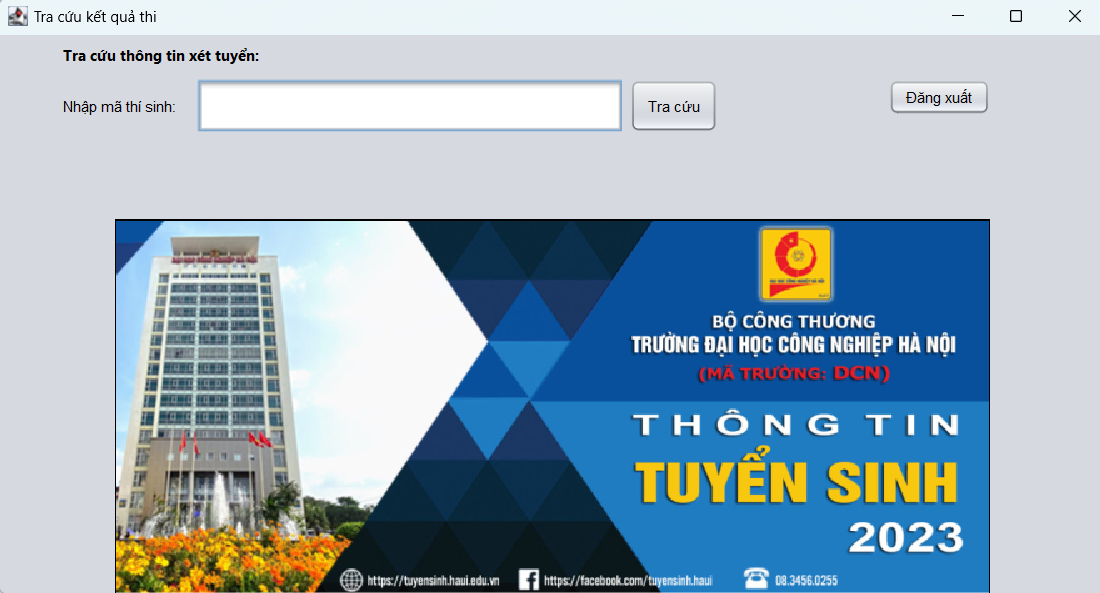
*Hình 2.20 Màn hình Thí Sinh Vi Phạm*

### 2.5.10 Đăng ký hồ sơ



*Hình 2.21 Màn hình Đăng Ký Hồ Sơ*

### 2.5.11 Tra cứu thông tin xét tuyển



*Hình 2.20 Màn hình Tra Cứu Thông Tin Xét Tuyển*

**PHẦN 3: TỔNG KẾT**

## 3.1 Các kết quả đạt được

- Các kiến thức sơ bộ:

* **Hướng đối tượng**  Trong Java, mọi thứ đều là một Object. Java có thể dễ dàng mở rộng và bảo trì vì nó được xây dựng dựa trên mô hình Object.
* **Nền tảng độc lập**  Không giống nhiều ngôn ngữ lập trình khác bao gồm cả C và C ++, khi Java được biên dịch, nó không được biên dịch thành ngôn ngữ máy nền tảng cụ thể, thay vào mã byte - nền tảng độc lập. Mã byte này được thông dịch bởi máy ảo (JVM) trên nền tảng nào đó mà nó đang chạy.
* **Đơn giản**  Java được thiết kế để dễ học. Nếu bạn hiểu khái niệm cơ bản về OOP Java, sẽ rất dễ để trở thành master về java.
* **Bảo mật**  Với tính năng an toàn của Java, nó cho phép phát triển các hệ thống không có virut, giả mạo. Các kỹ thuật xác thực dựa trên mã hoá khóa công khai.
* **Kiến trúc trung lập** - Trình biên dịch Java tạo ra định dạng tệp đối tượng kiến trúc trung lập, làm cho mã biên dịch được thực thi trên nhiều bộ vi xử lý, với sự hiện diện của hệ điều hành Java.
* **Portable**  Là kiến trúc tập trung và không có khía cạnh thực hiện phụ thuộc của đặc tả này làm cho Java khả chuyển. Trình biên dịch trong Java được viết bằng ANSI C, đó là một tập con POSIX.
* **Mạnh mẽ**  Java làm nỗ lực để loại trừ các tình huống dễ bị lỗi bằng cách kiểm tra lỗi tại thời gian biên dịch và kiểm tra lỗi tại runtime.
* **Đa luồng**  Với tính năng đa luồng của Java có thể viết các chương trình có thể thực hiện nhiều tác vụ đồng thời. Tính năng thiết kế này cho phép các nhà phát triển xây dựng các ứng dụng tương tác có thể chạy trơn tru hơn.
* **Thông dịch**  Mã byte Java được dịch trực tiếp tới các máy tính gốc và không được lưu trữ ở bất cứ đâu.
* **Hiệu năng cao**  Với việc sử dụng trình biên dịch Just-In-Time, Java cho phép thực hiện hiệu năng cao.
* **Phân tán**  Java được thiết kế cho môi trường phân tán của Internet.
* **Năng động**  Java là năng động hơn C hoặc C++ vì nó được thiết kế để thích nghi với môi trường đang phát triển. Các chương trình Java có thể mang một lượng lớn thông tin tại runtime mà có thể được sử dụng để xác minh và giải quyết các truy cập vào các đối tượng tại runtime.

- Hoàn thiện chương trình với các chức năng được thiết kế, đảm bảo khả năng vận hành các thao tác theo từng chức năng đã có. Bắt lỗi, soát các lỗi thực thi ở mức cơ bản.

## 3.2 Kết luận

Trong quá trình thực hiện bài tập lớn, chúng em đã cố gắng hết sức để tìm hiểu và học hỏi nhưng vì khả năng còn giới hạn không tránh khỏi những sai sót, nên có thể chưa giải quyết được tất cả những vấn đề đặt ra. Chúng em rất mong nhận được sự thông cảm của quý thầy cô và các bạn. Chúng em xin chân thành cảm ơn!

**TÀI LIỆU THAM KHẢO**

<https://www3.ntu.edu.sg/home/ehchua/programming/java/J4a_GUI.html>

<https://www.javatpoint.com/java-tutorial>

https://www.javatpoint.com/java-swing

<https://v1study.com/java-cach-su-dung-password-field.html#gsc.tab=0>

<https://www3.ntu.edu.sg/home/ehchua/programming/java/J4a_GUI_2.html>

<https://www3.ntu.edu.sg/home/ehchua/programming/java/J4a_GUI_2.html>