|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **KHOA CÔNG NGHỆ THÔNG TIN \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**      **BÁO CÁO THÍ NGHIỆM/THỰC NGHIỆM**  **LẬP TRÌNH JAVA**  **TRIỂN KHAI DỰ ÁN XÂY DỰNG PHẦN MỀM**  **QUẢN LÝ NHÂN VIÊN**   |  |  |  |  | | --- | --- | --- | --- | | GVHD: | : | Vũ Đình Minh |  | | Khóa | : | K17 |  | | Lớp | : | IT6019002 |  | | Sinh viên: | : | Lê Minh Quang:  Phạm Hoàng Thái:  Lê Quang Thắng: | 2022602763  2022602929  2022602065 | | Nhóm |  | 19 |  |   Hà Nội, 2024 |

# **MỤC LỤC**

**[LỜI MỞ ĐẦU](#_e7a0vktv1if8)** 7

**[Chương 1: Giới Thiệu](#_qjtrvy8mhj2v)** 9

[1.1 Tên đề tài](#_gpuftlfg8ci4) 9

[1.2 Lý do chọn đề tài](#_zf0i9qiajtoh) 9

[1.3 Xác định nội dung](#_u9if4gn3j7gj) 9

[1.4 Công cụ thực hiện đề tài](#_jmlun7qjdssx) 10

**[Chương 2: KẾT QUẢ NGHIÊN CỨU](#_xyzy8kybw724)** 11

[2.1. Giới thiệu](#_kp0a4vkj3669) 11

[2.1. Khảo sát hệ thống](#_1r9sv63e5nj6) 13

[2.1.1. Khảo sát sơ bộ](#_8yge1yd2enik) 13

[2.1.2. Đặc tả yêu cầu hệ thống](#_so97fgf3sncn) 13

[2.1.2.1. Yêu cầu chức năng](#_h13nhulkwqaa) 13

[2.1.2.2. Yêu cầu phi chức năng](#_24toq7jbw9is) 15

[2.2. Phân tích hệ thống](#_rtnj3p50scrf) 16

[2.2.1. Mô hình hóa chức năng hệ thống](#_tui17bhmi80o) 16

[2.2.2.1. Biểu đồ use case tổng quát](#_9ub05wm2kz73) 16

[2.2.2.2. Biểu đồ use case thứ cấp](#_33s6lv8qv1oy) 17

[2.2.2.3. Đặc tả use case](#_52mkn8674jik) 17

[2.2.2.3.1. Đăng nhập](#_44jfcp93vi) 17

[2.2.2.3.2. Tạo Tài Khoản Quản Lý Phòng Ban](#_8c77ufh6zi5j) 20

[2.2.2.3.3. Đổi Mật Khẩu Tài Khoản Quản Lý](#_yf7pyjuibkul) 23

[2.2.2.3.4. Quản lý nhân viên](#_qstr842t8e7q) 26

[2.2.2.3.5. Quản lý phòng ban](#_973rt6gjv08g) 32

[2.2.2.3.6. Tra cứu và tìm kiếm](#_59h1o7ih7qv1) 37

[2.2.2.3.7. Thao Tác Trên File Dữ Liệu](#_zcsaear68tx) 39

[2.2.2. Mô hình hóa dữ liệu.](#_rc0ssokkg527) 41

[2.2.2.1. Các yêu cầu dữ liệu](#_6s3wiz4i1hsk) 41

[2.2.2.2. Biểu đồ thực thể liên kế mức logic](#_xwojtttdiiot) 41

[2.2.2.3. Giao diện hệ thống](#_620n8tfd9tso) 41

[2.3. Thực hiện bài toán](#_mqolv3455fjk) 47

[2.3.1. Đối với người quản trị viên (Admin)](#_5ketsoj5n1zk) 47

[2.3.1.1. Tạo tài khoản cho Quản Lý Phòng Ban](#_ptmk093glzzc) 48

[2.3.1.2. Đăng Nhập](#_5a33mu3gktq) 51

[2.3.1.3. Quản lý phòng ban](#_of3473tgi4wh) 54

[2.3.1.4. Quản lý tài khoản](#_cu3m6o4g94v2) 57

[2.3.1.5. Tra cứu và tìm kiếm tài khoản (hoặc phòng ban)](#_a9l8eie7arqx)61

[2.3.1.6. Sửa thông tin của Tài Khoản (hoặc Phòng Ban)](#_dg5ll7l7tm9w) 65

[2.3.2. Đối với người quản lý ( Quản lý phòng ban)](#_nrfwobl60uxf) 67

[2.3.1.2. Đăng Nhập](#_wguvvq9hy4q7) 67

[2.3.2.2. Quản lý nhân viên của phòng ban](#_rvy8j9hssv0f) 70

[2.3.1.4. Tra cứu và tìm kiếm nhân viên](#_j8wgrxx1tel8) 74

[2.3.1.6. Sửa thông tin của Nhân Viên](#_8w2cbdv87jxj) 77

**[Chương 3: KẾT LUẬN VÀ BÀI HỌC KINH NGHIỆM](#_as25x5htivxv)** 80

[3.1. Nội dung đã thực hiện](#_w7q0m27esy68) 80

[3.1.1. Kiến thức và kỹ năng đã học được](#_iskvp41qq1yr) 80

[3.1.2. Chuẩn đầu ra đạt được](#_8cxi5zq4b6wm) 80

[3.1.3. Bài học kinh nghiệm](#_glo386xfuiva) 80

[3.2. Hướng phát triển](#_qfvksph31wek) 80

# 

# 

# 

# 

# **LỜI MỞ ĐẦU**

Trong bối cảnh phát triển mạnh mẽ của khoa học và công nghệ hiện nay, Công Nghệ Thông Tin (CNTT) đang có những bước tiến vượt bậc, làm thay đổi cách con người sống và làm việc. Một trong những xu hướng quan trọng là việc áp dụng CNTT vào nhiều lĩnh vực khác nhau nhằm hướng tới xây dựng một hành tinh thông minh. Sự kết hợp giữa CNTT và Internet đã tạo nên một mạng lưới kết nối toàn cầu, hỗ trợ đắc lực cho nhiều ngành nghề như giao thông, quân sự, y học, và đặc biệt là trong công tác quản lý, bao gồm quản lý nhân viên.

Để đáp ứng nhu cầu này, dự án này nhằm mục đích xây dựng một phần mềm quản lý nhân viên hoàn chỉnh và hiệu quả. Phần mềm này không chỉ giúp tổ chức quản lý thông tin nhân viên một cách tiện lợi, mà còn cung cấp các tính năng như quản lý phòng ban, đánh giá hiệu suất, và tương tác nội bộ, đồng thời đảm bảo tính bảo mật và tuân thủ các quy định pháp lý.

Nhờ sự hướng dẫn tận tình của thầy Vũ Đình Minh, nhóm chúng em đã nghiên cứu và áp dụng các kiến thức đã học để xây dựng chương trình quản lý nhân viên. Trong quá trình khảo sát, nhóm đã tìm hiểu kỹ về nhu cầu và khó khăn trong việc quản lý nhân viên tại các công ty và doanh nghiệp. Dựa trên những yêu cầu thu thập được, nhóm tiến hành phân tích và thiết kế hệ thống với các chức năng quản lý nhân viên. Sau đó, nhóm sử dụng ngôn ngữ lập trình Java để phát triển các chức năng đã thiết kế, đồng thời tiến hành kiểm thử để đảm bảo phần mềm hoạt động hiệu quả.

Do thời gian và kiến thức còn hạn chế, phần mềm của nhóm vẫn còn một số thiếu sót. Nhóm rất mong nhận được ý kiến đóng góp của thầy cô và các bạn để hoàn thiện phần mềm hơn.

Nhóm hy vọng rằng phần mềm quản lý nhân viên này sẽ là công cụ hữu ích cho việc quản lý nhân viên và phòng ban tốt hơn. Chúng em rất mong nhận được phản hồi để tiếp tục cải tiến và phát triển phần mềm.

# **Chương 1: Giới Thiệu**

## **1.1 Tên đề tài**

Xây dựng phần mềm quản lý nhân viên

## **1.2 Lý do chọn đề tài**

Ngành Công nghệ thông tin (CNTT) đang trải qua một sự phát triển nhanh chóng và có ảnh hưởng sâu sắc đến nhiều lĩnh vực. Trong đó có môi trường kinh doanh ngày nay, việc quản lý nhân viên là một phần không thể thiếu để đảm bảo sự hiệu quả và thành công của một tổ chức. Đối diện với sự phát triển không ngừng của công nghệ thông tin, việc sử dụng phần mềm quản lý nhân viên đã trở thành một yếu tố quan trọng, giúp tối ưu hóa quy trình quản lý, tăng cường sự linh hoạt và nâng cao năng suất lao động.

Nhận thấy được, quản lý nhân viên là một yêu cầu cấp thiết trong các tổ chức như công ty, doanh nghiệp. Nhóm 19 quyết định chọn đề tài: “***Xây dựng phần mềm quản lý nhân viên***.” Một phần mềm có thể tận dụng những lợi ích của CNTT để tối ưu hóa quá trình quản lý nhân viên trong các phòng ban. Phần mềm quản lý nhân viên giúp tạo ra sự thuận lợi và nhanh chóng trong việc quản lý. Nó giúp tổ chức tối ưu hóa quy trình làm việc và tăng cường năng suất lao động thông qua việc tự động hóa các nhiệm vụ quản lý như lập kế hoạch làm việc, đánh giá hiệu suất, và quản lý thời gian.

## **1.3 Xác định nội dung**

* Mô hình hóa bài toán cụ thể:
* Người quản trị(Admin) : Sau khi đăng nhập quản lý các tài khoản và quản lý các phòng ban.
* Quản lý tài khoản:
* Quản lý phòng ban:
* Tra cứu và tìm kiếm:
* Tương tác với File lưu trữ dữ liệu
* Người quản lý phòng ban : Sau khi đăng nhập quản lý nhân viên trong các phòng ban.
* Quản lý nhân viên

## **1.4 Công cụ thực hiện đề tài**

Cài đặt và triển khai phần mềm dựa trên công cụ viết mã nguồn như:

* NetBeans:
* Là một công cụ hỗ trợ người dùng lập trình mã nguồn mở. Đây là một môi trường phát triển tích hợp dành cho Java. Người dùng có thể sử dụng NetBeans để mở rộng hơn mã nguồn bằng cách chèn plugins cho project. Đặc biệt, NetBeans là phần mềm hoàn toàn miễn phí, với nhiều plugin hữu ích.
* Dùng để cài đặt chương trình.
* JDK 17:
* JDK 17 là một bản tham chiếu của Java 17, được ra mắt vào ngày 15 tháng 9. JDK 17 trong đó là bản phát hành hỗ trợ được Oracle hỗ trợ trong sáu tháng. Oracle triển khai JDK 17 như một điểm khởi đầu để chuyển sang JDK 17, với người dùng có thể thử nghiệm trên JDK 17 trước khi triển khai lên JDK 19. Các bản phát hành LTS được xuất bản ba năm một lần.
* JDK 17 (Java Development Kit 17) là một bản phát hành của Java Development Kit, bao gồm môi trường phát triển và các công cụ cần thiết để phát triển ứng dụng Java. Phiên bản này thường đi kèm với một số cải tiến, sửa lỗi và tính năng mới so với các phiên bản JDK trước đó, cũng như các bản cập nhật cho ngôn ngữ Java. JDK 17 có thể được sử dụng để phát triển và triển khai các ứng dụng Java.

- Case Studio 2 :

* Case Studio – Phần mềm vẽ mô hình Entity Relationship Diagrams (ERD) và Data Flow Diagrams (DFD) cũng như tạo script SQL cho nhiều cơ sở dự liệu một cách tự động. Phần mềm hỗ trợ đầy đủ cho 20 cơ sở dữ liệu như Oracle, DB2, MS SQL, Sybase, MySQL, Firebird, PostgreSQL vvv.
* Chức năng chính của CASE Studio 2 : tạo mô hình ERD từ mã script SQL, Đảo mã từ file HTML cụ thể hay file dữ liệu RTF, xuất mô hình Data Flow Diagrams thành file quản lý định đạng XML, Templater editor…

- Star UML :

* StarUML là một mô hình nền tảng, là phần mềm hỗ trợ UML (Unified Modeling Language). Nó hỗ trợ các phương pháp tiếp cận MDA (Model Driven Architecture) bằng cách hỗ trợ các khái niệm hồ sơ UML. Tức là StarUML hỗ trợ phân tích và thiết kế hệ thống một điều mà bất cứ dự án nào đều cần có. Ngoài ra dùng StarUML sẽ đảm bảo tối đa hóa năng suất và chất lượng của các dự án phần mềm của bạn. Vì nó cho phép mô hình hóa nên sẽ không phụ thuộc vào người code, ngôn ngữ code hay nền tảng sử dụng. Do mô hình hóa nên rất dễ đọc và dễ hiểu. Một điều mình thích nhất ở StarUML là khả năng tự sinh code từ những mô hình class.

# **Chương 2: KẾT QUẢ NGHIÊN CỨU**

## **2.1. Giới thiệu**

Trong bối cảnh số hóa và quản lý thông tin ngày càng quan trọng, nhóm chúng tôi quyết định lựa chọn đề tài "Xây dựng phần mềm quản lý nhân viên" nhằm tạo ra một công cụ hữu ích cho việc quản lý và tra cứu thông tin một cách hiệu quả. Sản phẩm của chúng em sẽ là một ứng dụng phần mềm chạy được, cung cấp các chức năng quản lý thông tin như thêm, xóa, sửa, tìm kiếm nhân viên , phòng ban , quản lý nhân viên và tăng khả năng tổ chức của các phòng ban.

Để phát triển phần mềm này, chúng em sẽ áp dụng mô hình phát triển phần mềm MVC, một phương pháp phát triển phần mềm hiện đại giúp tăng tính linh hoạt và khả năng phản hồi nhanh chóng đối với các yêu cầu và thay đổi mới. Quy trình bao gồm các giai đoạn chính: khởi tạo, lập kế hoạch và thiết kế, phát triển, kiểm thử tích hợp và triển khai, cùng bảo trì và hỗ trợ. Mỗi giai đoạn sẽ có các bước cụ thể nhằm đảm bảo chất lượng và tiến độ của dự án.(\*)

Mô hình Model-View-Controller (MVC) là một mẫu kiến ​​trúc phân tách một ứng dụng thành ba thành phần logic chính Model, View và Controller. Do đó viết tắt MVC. Mỗi thành phần kiến ​​trúc được xây dựng để xử lý khía cạnh phát triển cụ thể của một ứng dụng.

Nhóm sẽ sử dụng các công cụ hỗ trợ viết mã nguồn như NetBeans, GitHub, Rational Rose, JDK 17 của ngôn ngữ lập trình Java. Các công cụ này không chỉ hỗ trợ mạnh mẽ cho việc phát triển mà còn giúp tối ưu hóa quy trình làm việc nhóm, đảm bảo tính toàn vẹn và khả năng mở rộng của phần mềm.

Java là ngôn ngữ lý tưởng để xây dựng chương trình nhờ tính đa nền tảng, bảo mật cao, và quản lý bộ nhớ hiệu quả. Cộng đồng hỗ trợ lớn và khả năng mở rộng linh hoạt giúp dễ dàng thêm tính năng mới. Hiệu suất cao, tích hợp công nghệ hiện đại, và khả năng bảo trì tốt nâng cao hiệu quả phát triển. Hỗ trợ đa luồng cải thiện hiệu suất và khả năng đáp ứng, trong khi khả năng tích hợp dễ dàng với các hệ thống khác giúp trao đổi dữ liệu mượt mà. Những ưu điểm này khiến Java trở thành lựa chọn mạnh mẽ và tin cậy.

Dù Java có nhiều ưu điểm, nó cũng có nhược điểm như hiệu suất thấp hơn so với ngôn ngữ biên dịch, tiêu thụ bộ nhớ nhiều, và cú pháp phức tạp. Java khởi động chậm, thiếu kiểm soát chi tiết bộ nhớ, và phụ thuộc vào JVM. Chu kỳ cập nhật liên tục đòi hỏi lập trình viên phải theo kịp, và chi phí phát triển có thể tăng để đạt hiệu suất cao. Những yếu tố này cần được cân nhắc khi chọn Java cho chương trình quản lý thông tin nhân viên.

Với mục tiêu tạo ra một sản phẩm hoàn thiện và tiện dụng, chúng em hy vọng phần mềm quản lý nhân viên này sẽ đáp ứng tốt nhu cầu của người dùng.

## **2.1. Khảo sát hệ thống**

### ***2.1.1. Khảo sát sơ bộ***

Mục tiêu chính của xây dựng phần mềm quản lý nhân viên là nhằm tối ưu hóa các quy trình liên quan đến nhân viên hoặc phòng ban.

Về phạm vi, hệ thống cần đáp ứng các yêu cầu cơ bản như quản lý thông tin nhân viên, theo dõi quá trình làm việc. Ngoài ra, hệ thống cũng cung cấp các tính năng nâng cao như tìm kiếm/tra cứu nhân viên, tăng cường tốt hơn khả năng tổ chức và quản lý từng cá thể trong từng phòng ban. Qua đó, phần mềm quản lý nhân viên sẽ giúp các tổ chức quản lý nguồn lao động hiệu quả hơn.

### **2.1.2. Đặc tả yêu cầu hệ thống**

#### **2.1.2.1. Yêu cầu chức năng**

* Đối với ***người quản trị viên (Admin)*** :
  + **Tạo Tài Khoản Phòng Ban**: Cho phép quản trị viên tạo tài khoản quản lý phòng ban mới trong hệ thống bằng cách cung cấp thông tin tài khoản cần thiết.
  + **Đăng nhập**: Cho phép người dùng đăng nhập vào tài khoản đã tạo trước đó bằng cách cung cấp thông tin đăng nhập như tên tài khoản và mật khẩu để thực hiện các chức năng tương ứng với quyền của mình.
  + **Xem thông tin nhân viên**: cho phép bản đọc xem thông tin chi tiết về nhân viên trong phòng ban.
  + **Tìm kiếm nhân viên bằng id**: Cho phép người dùng tìm kiếm thông tin nhân viên dựa trên id của nhân viên.
  + **Tìm kiếm nhân viên bằng tên**: Cho phép người dùng tìm kiếm thông tin nhân viên dựa trên tên của nhân viên.
  + **Tìm kiếm nhân viên bằng phòng ban**: Cho phép người dùng tìm kiếm thông tin nhân viên dựa trên phòng ban của nhân viên.
  + **Chỉnh sửa phòng ban**: Cho phép người dùng khả năng chỉnh sửa thông tin phòng ban vào Model danh sách phòng ban.
  + **Thêm phòng ban**: Cho phép người dùng khả năng thêm phòng ban vào Model danh sách phòng ban.
  + **Xóa phòng ban**: Cho phép người dùng khả năng xóa phòng ban khỏi Model danh sách phòng ban.
  + **Tìm kiếm phòng ban theo id**: Cho phép người dùng tìm kiếm thông tin phòng ban dựa trên id phòng ban.
  + **Tìm kiếm phòng ban theo tên**: Cho phép người dùng tìm kiếm thông tin phòng ban dựa trên tên phòng ban.
  + **Hiển thị danh sách nhân viên**: Cho phép người dùng hiện thị danh sách nhân viên ra màn hình.
  + **Hiển thị danh sách phòng ban**: Cho phép người dùng hiện thị danh sách phòng ban ra màn hình.
  + **Lưu thông tin cập nhập**: Cho phép người dùng cập nhập lại thông tin danh sách nhân viên trong một phòng ban dựa trên hành động trước đó.
  + **Đọc file lưu trữ**: Cho phép người dùng đọc file lưu trữ thông tin danh sách nhân viên dựa trên file được chọn.
  + **Lưu file lưu trữ**: Cho phép người dùng lưu thông tin dữ liệu danh sách nhân viên vào file lưu trữ danh sách nhân viên vào file được chọn.
  + **Đóng file lưu trữ**: Cho phép người dùng đóng file lưu trữ đang được mở.
  + **Xóa form**: Tái thiết lập giá trị mặc định của trường nhập dữ liệu.
  + **Thoát chương trình**: Cho phép người dùng dừng chương trình.
  + **Đăng xuất**: Cho phép người dùng đăng xuất khỏi tài khoản hiện tại, kết thúc phiên làm việc và trở về giao diện người dùng chưa đăng nhập.
* Đối với ***người quản lý (Quản lý từng phòng ban )*** :
* **Chính sửa thông tin nhân viên**: Cung cấp cho người dùng khả năng chỉnh sửa thông tin nhân viên trong phòng ban.
* **Thêm nhân viên**: Cho phép người dùng khả năng thêm nhân viên vào danh sách nhân sự trong phòng ban.
* **Xóa nhân viên**: Cho phép người dùng khả năng xóa nhân viên ra khỏi danh sách nhân sự trong phòng ban.
* **Hiển thị danh sách nhân viên**: Cho phép người dùng hiện thị danh sách nhân viên ra màn hình.
* **Hiển thị danh sách phòng ban**: Cho phép người dùng hiện thị danh sách phòng ban ra màn hình.
* **Lưu thông tin cập nhập**: Cho phép người dùng cập nhập lại thông tin danh sách nhân viên trong một phòng ban dựa trên hành động trước đó.
* **Đọc file lưu trữ**: Cho phép người dùng đọc file lưu trữ thông tin danh sách nhân viên dựa trên file được chọn.
* **Lưu file lưu trữ**: Cho phép người dùng lưu thông tin dữ liệu danh sách nhân viên vào file lưu trữ danh sách nhân viên vào file được chọn.
* **Đóng file lưu trữ**: Cho phép người dùng đóng file lưu trữ đang được mở.
* **Xóa form**: Tái thiết lập giá trị mặc định của trường nhập dữ liệu.
* **Thoát chương trình**: Cho phép người dùng dừng chương trình.
* **Đăng xuất**: Cho phép người dùng đăng xuất khỏi tài khoản hiện tại, kết thúc phiên làm việc và trở về giao diện người dùng chưa đăng nhập.

#### **2.1.2.2. Yêu cầu phi chức năng**

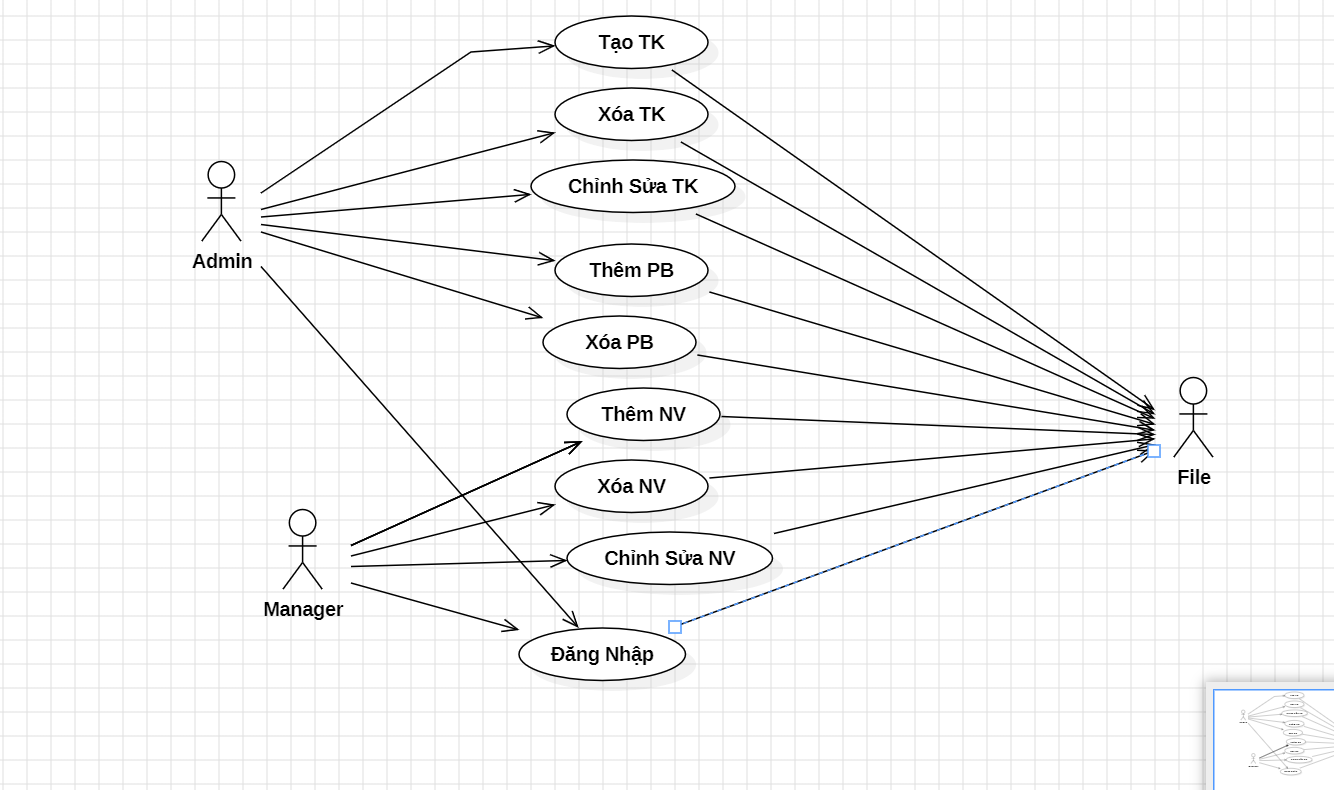
* Hệ thống xử lý nhanh gọn, chính xác và thuận tiện.
* Giao diện hài hòa, thân thiện và dễ sử dụng ( có thể thêm chức năng cần thiết sau này).
* Quy trình phát triển phần mềm phù hợp để dễ dàng bảo trì và nâng cấp.
* Có sự ràng buộc chặt chẽ về chất lượng, môi trường và chuẩn sử dụng.
* Đảm bảo về mặt thời gian, bản quyền.
* Chi phí chấp nhận được.
* Ngôn ngữ lập trình: Java - Swing.
* Lưu trữ cơ sở dữ liệu: File Text.
* Hệ điều hành: Windows

## **2.2. Phân tích hệ thống**

### **2.2.1. Mô hình hóa chức năng hệ thống**

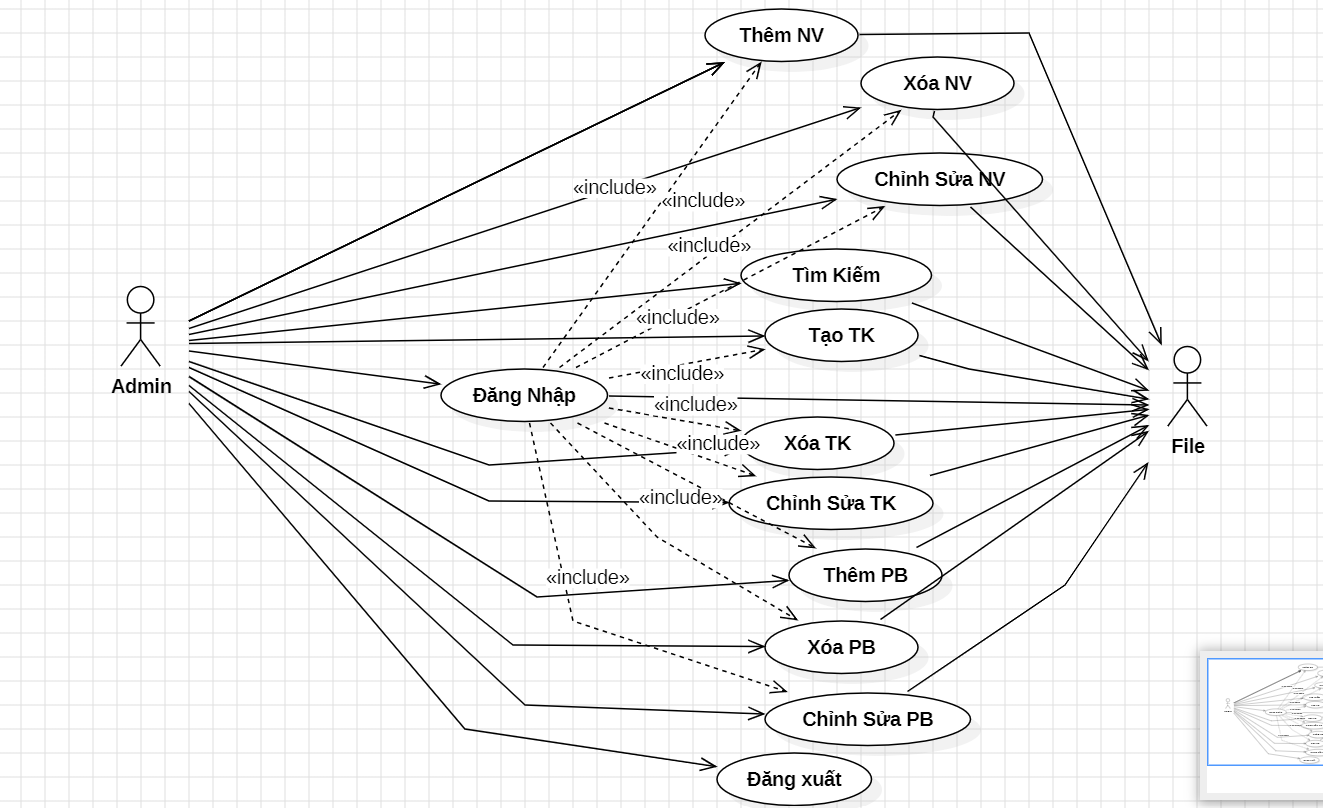
#### **2.2.2.1. Biểu đồ use case tổng quát**

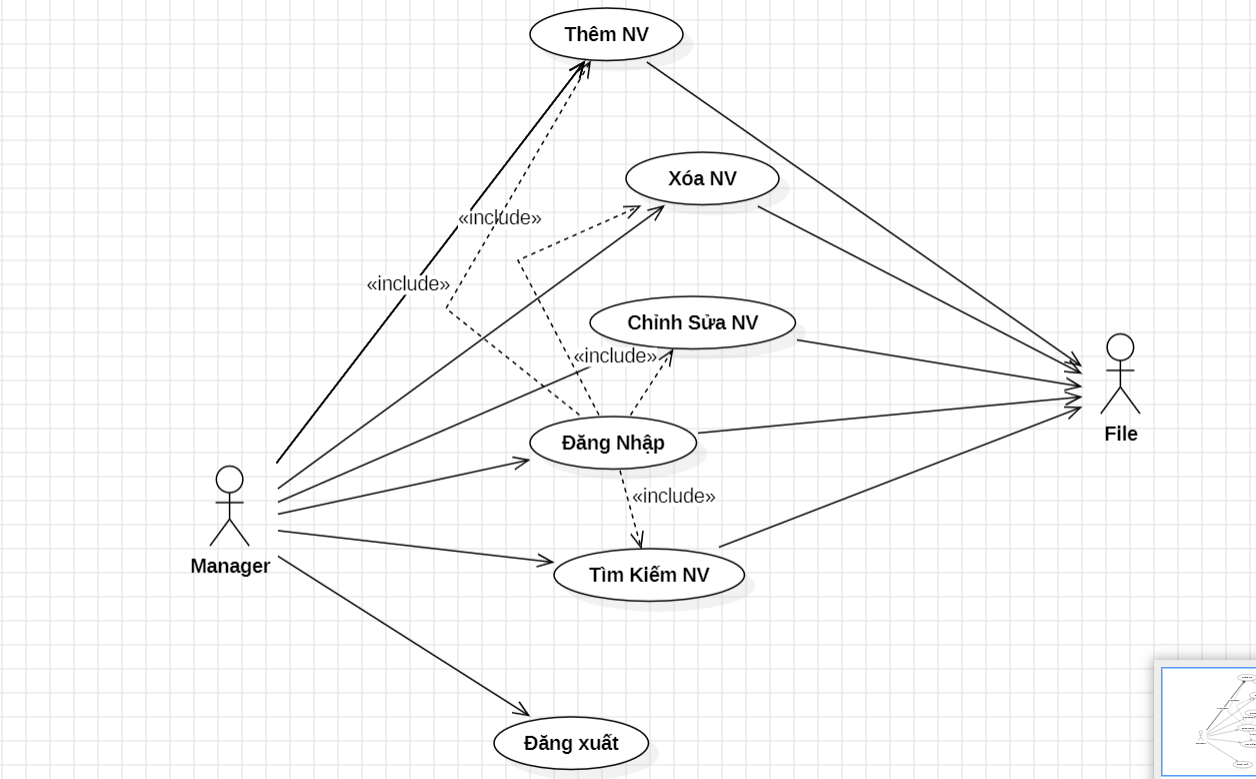
* Các tác nhân của hệ thống: Theo phần đặc tả yêu cầu người dùng, có thể xác định các actor như sau:
  + Người quản trị viên (admin) : Đảm nhận nhiệm vụ quản lý và tổ chức quản lý các phòng ban.
  + Quản lý phòng ban (manager department) : Đảm nhận nhiệm vụ quản lý và thao tác nâng cao đối với từng cá thể nhân viên tồn tại trong các phòng ban.
* Các use case chính: Dựa theo việc xác định actor ở trên, ta xác định được các use case tương ứng:
  + Đăng nhập
  + Tạo tài khoản quản lý
  + Quản lý nhân viên
  + Quản lý phòng ban
  + Tra cứu và tìm kiếm



*Hình 2.1 Biểu đồ use case tổng quát*

#### **2.2.2.2. Biểu đồ use case thứ cấp ( Biểu đồ use case phân rã )**





*Hình 2.0 Biểu đồ use case thứ cấp*

#### **2.2.2.3. Đặc tả use case**

##### **2.2.2.3.1. Đăng nhập**

a. Đăng nhập (người quản trị hoặc quản lý phòng ban )

|  |  |
| --- | --- |
| Tên use case | Đăng nhập |
| Tác nhân | Người quản lý phòng ban / quản trị |
| Mô tả | Use case này cho phép người dùng đăng nhập tài khoản vào hệ thống. Để thực hiện các chức năng quản lý. |
| Sự kiện kích hoạt chức năng | Use case bắt đầu khi người dùng bắt đầu nhập thông tin tài khoản vào trường nhập dữ liệu và nhấp nút “Đăng nhập” trên giao diện chính. Hệ thống hiển thị lên màn hình một giao diện thực hiện chức năng tương ứng với quyền của tài khoản đó. |
| Tiền điều kiện | Thông tin tài khoản người quản lý đã có trong file lưu trữ |
| Luồng sự kiện chính | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản trị, quản lý hệ thống | 1. Người quản trị/quản lý nhập đầy đủ các thông tin như tên tài khoản, mật khẩu vào các trường tương ứng. 2. Người quản trị/quản lý kích vào nút “Đăng nhập”. Hệ thống kiểm tra chi tiết người dùng trong file lưu trữ các TAI\_KHOAN và hiển thị trang chủ đầy đủ các chức năng tương ứng với quyền của người quản lý. | |  | Use case kết thúc. | |
| Luồng sự kiện thay thế | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản trị, quản lý | Tại mục 1a trong luồng sự kiện chính nếu người quản lý nhập không đâỳ đủ thông tin đã nêu trên thì hệ thống sẽ hiển thị lên màn hình thông báo lỗi “Tài khoản hoặc mật khẩu bị sai”. Use case kết thúc. | | Người quản lý, hệ thống | Tại mục 1a trong luồng sự kiện chính, nếu người dùng nhập sai tên tài khoản hay mật khẩu thì hệ thống sẽ thông báo lỗi. Người dùng có thể nhập lại và tiếp tục bước 2 hoặc chọn các chức năng khác để kết thúc use case. | | Người quản lý, hệ thống | Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với file lưu trữ dữ liệu tài khoản thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. | |
| Hậu điều kiện | Không có |
| Điểm mở rộng | Không có |

*Bảng 2.1 Use case Đăng nhập (Quản trị viên hoặc quản lý)*

##### **2.2.2.3.2. Tạo Tài Khoản Quản Lý Phòng Ban**

b. Tạo Tài Khoản Quản Lý Phòng Ban (người quản trị viên(Admin) )

|  |  |
| --- | --- |
| Tên use case | Tạo tài khoản quản lý phòng ban |
| Tác nhân | Người Quản Lý Phòng Ban |
| Mô tả | Use case này cho phép người quản trị ( Admin) đăng ký tài khoản của người quản lý phòng ban để thực hiện các chức năng quản lý nhân viên của phòng ban đó |
| Sự kiện kích hoạt chức năng | Use case bắt đầu khi người quản trị (Admin) kích vào nút “Tạo Tài Khoản” trên giao diện của tài khoản người quản trị (Admin) đã đăng nhập. Hệ thống hiển thị lên màn hình một giao diện cho phép nhập thông tin đăng ký. |
| Tiền điều kiện | Đăng nhập với tài khoản có tư cách là người quản trị(Admin) |
| Luồng sự kiện chính | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản trị(Admin)/hệ thống | 1. Người quản trị(Admin) nhập đầy đủ thông tin như tên tài khoản, mật khẩu vào các trường tương ứng 2. Người quản trị(Admin) kích vào nút “Lưu”nếu các thông tin đều đúng yêu cầu thì hệ thống sẽ hiện thông báo đăng ký tài khoản thành công 3. Sau khi Người quản trị(Admin) click vào nút “Lưu” hệ thống sẽ trở về giao diện ban đầu. | | Hệ thống | Use case kết thúc | |
| Luồng sự kiện thay thế | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản trị(Admin)/hệ thống | Tại mục 1a trong luồng sự kiện chính nếu người dùng nhập không đâỳ đủ thông tin đã nêu trên thì hệ thống sẽ hiển thị lên màn hình thông báo lỗi “Vui lòng nhập đầy đủ thông tin”. Use case kết thúc. | | Người quản trị(Admin)/hệ thống | Tại mục 1a nếu người dùng nhập sai yêu cầu mật khẩu “mật khẩu phải có ít nhất 5 ký tự” thì yêu cầu này sẽ hiện màu đỏ và không thể đăng ký được.Use case kết thúc | | Người quản trị(Admin)/hệ thống | Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với file lưu trữ dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. | |
| Hậu điều kiện | Thông tin người dùng được thêm vào file lưu trữ các TAIKHOAN |
| Điểm mở rộng | Không có |

*Bảng 2.2 Use case Tạo Tài Khoản Quản lý Phòng Ban*

##### **2.2.2.3.3. Đổi Mật Khẩu Tài Khoản Quản Lý**

c. Đổi mật khẩu (Người quản trị (Admin))

|  |  |
| --- | --- |
| Tên use case | Đổi mật khẩu |
| Tác nhân | Người quản trị (Admin) |
| Mô tả | Use case cho phép Người quản trị (Admin) thay đổi mật khẩu. |
| Sự kiện kích hoạt chức năng | Use case này bắt đầu khi Người quản trị (Admin) kích chọn chức năng “Chi Tiết” trên 1 dòng thông tin tại giao diện quản lý danh sách tài khoản tại giao diện trang chủ của admin. |
| Tiền điều kiện | Phải đăng nhập thành công tài khoản với tư cách là Người quản trị (Admin) |
| Luồng sự kiện chính | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản lý, hệ thống | Hệ thống sẽ hiển thị giao diện đổi mật khẩu gồm nhập mật khẩu cũ, nhập mật khẩu mới. Người quản trị (Admin) nhập đầy đủ các thông tin được nêu trên | | Người quản lý, hệ thống | Người quản trị (Admin) kích vào nút “Lưu”. Hệ thống hiển thị thông báo “Đổi mật khẩu thành công”. | | Hệ thống | Use case kết thúc. | |
| Luồng sự kiện thay thế | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản lý, hệ thống | Tại bước đăng nhập tạo tài khoản quản lý trong luồng sự kiện chính nếu người dùng nhấn vào nút “Huỷ”. Hệ thống sẽ hiển thị giao diện trang chủ chính. Use case kết thúc. | | Người quản lý, hệ thống | Tại bước đăng nhập trong luồng sự kiện chính nếu người quản lý không nhập đầy đủ thông tin hệ thống hiển thị lên màn hình thông báo lỗi “Vui lòng nhập đầy đủ thông tin”. Use case kết thúc.. | | Người quản lý, hệ thống | Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với file lưu trữ dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. | |
| Hậu điều kiện | Thông tin mật khẩu mới được cập nhật vào file lưu trữ dữ liệu. |
| Điểm mở rộng | Không có |

*Bảng 2.3 Use case Đổi mật khẩu (admin)*

##### **2.2.2.3.4. Quản lý nhân viên**

d. Quản lý nhân viên

|  |  |
| --- | --- |
| Tên use case | Quản lý nhân viên |
| Tác nhân | Người quản lý phòng ban |
| Mô tả | Use case này cho phép quản lý thực hiện các chức năng thêm, sửa, xóa nhân viên , xem danh sách nhân viên. |
| Sự kiện kích hoạt chức năng | Use case này bắt đầu khi Người quản lý phòng ban chọn các chức năng thêm nhân viên, sửa thông tin nhân viên, xóa nhân viên , xem nhân viên tại bảng hiển thị thông tin danh sách ”nhân viên” trên menu tại giao diện trang chủ của người quản lý. |
| Tiền điều kiện | Cần phải đăng nhập thành công tài khoản với tư cách người quản lý phòng ban |
| Luồng sự kiện chính | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản lý phòng ban, hệ thống | Người quản lý chọn “Thêm nhân viên”.   1. Hệ thống sẽ hiển thị giao diện cho phép người quản lý nhập các thông tin: mã nhân viên, tên nhân viên, ngày sinh, phòng ban, chức vụ, giới tính, địa chỉ, ngày vào làm, lương cơ bản. 2. Người quản lý nhập đầy đủ thông tin đã nêu trên. 3. Người quản lý nhấn vào nút “Lưu”. Hệ thống sẽ tự động cập nhập và hiển thị danh sách mới. Use case kết thúc. | | Người quản lý phòng ban, hệ thống | Người quản lý chọn “Xóa nhân viên”.   1. Người quản lý sẽ chọn chính xác nhân viên cần xóa dựa trên thao tác nhấp vào một dòng thông tin chi tiết nhân viên. 2. Hệ thống sẽ hiển thị thông báo “Bạn có chắc chắn muốn xóa” lên màn hình. 3. Người quản trị kích vào nút “Yes”. Hệ thống sẽ hiển thị thông báo “Xóa thành công” lên màn hình. Use case kết thúc. | | Người quản lý phòng ban, hệ thống | Người quản lý chọn “Sửa thông tin nhân viên”.   1. Hệ thống sẽ hiển thị các thông tin gồm: mã nhân viên, tên nhân viên, ngày sinh, phòng ban, chức vụ, giới tính, địa chỉ, ngày vào làm, lương cơ bản. 2. Người quản lý chọn một dòng thông tin chi tiết nhân viên. Hệ thống sẽ hiển thị lên giao diện sửa thông tin sửa gồm các thông tin: mã nhân viên, tên nhân viên, ngày sinh, phòng ban, chức vụ, giới tính, địa chỉ, ngày vào làm, lương cơ bản. 3. Người quản lý sẽ nhập những thông tin cần chỉnh sửa và kích vào nút “Lưu”. Hệ thống sẽ hiển thị lên màn hình thông báo “Sửa thông tin nhân viên thành công”. Use case kết thúc. | |
| Luồng sự kiện thay thế | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản lý phòng ban, hệ thống | Tại use case quản lý nhân viên, trong luồng sự kiện chính nếu người quản lý nhập không đủ thông tin thì hệ thống sẽ hiển thị thông báo lỗi lên màn hình. Use case kết thúc. | | Người quản lý phòng ban, hệ thống | Tại use case quản lý nhân viên trong luồng sự kiện chính, nếu người quản lý kích vào nút “Hủy” thì toàn bộ thông tin vừa nhập sẽ bị xóa. Use case kết thúc | |  |  | | Người quản lý phòng ban, hệ thống | Trước khi thực hiện use case quản lý nhân viên tại luồng sự kiện chính người quản lý có thể nhập tên nhân viên vào ô tìm kiếm và kích vào nút “Tìm kiếm”.Hệ thống sẽ hiển thị thông tin nhân viên liên quan đến từ khóa. Người dùng tiếp tục thực hiện trong luồng sự kiện chính. Use case kết thúc. | | Người quản lý phòng ban, hệ thống | Tại use case quản lý nhân viên tại luồng sự kiện chính nếu người quản lý kích vào nút “No” hoặc “Cancel”. Hệ thống sẽ không ghi nhận hành động xóa thông tin nhân viên vừa chọn. Use case kết thúc. | | Người quản lý phòng ban, hệ thống | Tại bước use case quản lý nhân viên trong luồng sự kiện chính nếu người quản lý nhấp vào nút “Hủy” thì hệ thống sẽ quay lại thông tin ban đầu của nhân viên.. Use case kết thúc. | | Người quản lý phòng ban, hệ thống | Tại bước use case quản lý nhân viên trong luồng sự kiện chính khi kích vào nút “Huỷ” hệ thống sẽ quay lại mục 3a. Use case kết thúc. | | Người quản lý phòng ban, hệ thống | Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với file lưu trữ dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. | |
| Hậu điều kiện | Thông tin được cập nhật file lưu trữ dữ liệu |
| Điểm mở rộng | Không có |

*Bảng 2.4 Use case Quản lý nhân viên*

##### **2.2.2.3.5. Quản lý phòng ban**

e. Quản lý phòng ban

|  |  |
| --- | --- |
| Tên use case | Quản lý phòng ban |
| Tác nhân | Người quản lý phòng ban |
| Mô tả | Use case này cho phép quản lý thực hiện các chức năng thêm, sửa, xóa phòng ban. |
| Sự kiện kích hoạt chức năng | Use case này bắt đầu khi người quản lý chọn các chức năng thêm phòng ban, sửa thông tin phòng ban, xóa phòng ban tại bảng hiển thị thông tin danh sách ”phòng ban” trên menu tại giao diện trang chủ của người quản lý. |
| Tiền điều kiện | Cần phải đăng nhập thành công tài khoản với tư cách người quản lý phòng ban |
| Luồng sự kiện chính | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản lý, hệ thống | Người quản lý chọn “Thêm phòng ban”.   1. Hệ thống sẽ hiển thị giao diện cho phép người quản lý nhập các thông tin: mã phòng ban, tên phòng ban, ngày thành lập. 2. Người quản lý nhập đầy đủ thông tin đã nêu trên. 3. Người quản lý nhấn vào nút “Lưu”. Hệ thống sẽ tự động cập nhập và hiển thị danh sách mới. Use case kết thúc. | | Người quản lý, hệ thống | Người quản lý chọn “Xóa phòng ban”.   1. Người quản lý sẽ chọn chính xác phòng ban cần xóa dựa trên thao tác nhấp vào một dòng thông tin chi tiết phòng ban. 2. Hệ thống sẽ hiển thị thông báo “Bạn có chắc chắn muốn xóa” lên màn hình. 3. Người quản trị kích vào nút “Yes”. Hệ thống sẽ hiển thị thông báo “Xóa thành công” lên màn hình. Use case kết thúc. | | Người quản lý, hệ thống | Người quản lý chọn “Sửa thông tin phòng ban”.   1. Hệ thống sẽ hiển thị các thông tin gồm: mã phòng ban, tên phòng ban, ngày thành lập. 2. Người quản lý chọn một dòng thông tin chi tiết phòng ban. Hệ thống sẽ hiển thị lên giao diện sửa thông tin sửa gồm các thông tin: mã phòng ban, tên phòng ban, ngày thành lập. 3. Người quản lý sẽ nhập những thông tin cần chỉnh sửa và kích vào nút “Lưu”. Hệ thống sẽ hiển thị lên màn hình thông báo “Sửa thông tin phòng ban thành công”. Use case kết thúc. | |
| Luồng sự kiện thay thế | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản lý, hệ thống | Tại use case quản lý phòng ban, trong luồng sự kiện chính nếu người quản lý nhập không đủ thông tin thì hệ thống sẽ hiển thị thông báo lỗi lên màn hình. Use case kết thúc. | | Người quản lý, hệ thống | Tại use case quản lý phòng ban trong luồng sự kiện chính, nếu người quản lý kích vào nút “Hủy” thì toàn bộ thông tin vừa nhập sẽ bị xóa. Use case kết thúc | | Người quản lý, hệ thống | Trước khi thực hiện use case quản lý phòng ban tại luồng sự kiện chính người quản lý có thể nhập tên phòng ban vào ô tìm kiếm và kích vào nút “Tìm kiếm”.Hệ thống sẽ hiển thị thông tin phòng ban liên quan đến từ khóa. Người dùng tiếp tục thực hiện trong luồng sự kiện chính. Use case kết thúc. | | Người quản lý, hệ thống | Tại use case quản lý phòng ban tại luồng sự kiện chính nếu người quản lý kích vào nút “No” hoặc “Cancel”. Hệ thống sẽ không ghi nhận hành động xóa thông tin phòng ban vừa chọn. Use case kết thúc. | | Người quản lý, hệ thống | Tại bước use case quản lý phòng ban trong luồng sự kiện chính nếu người quản lý nhấp vào nút “Hủy” thì hệ thống sẽ Huỷ thông tin ban đầu của phòng ban. Use case kết thúc. | | Người quản lý, hệ thống | Tại bước use case quản lý phòng ban trong luồng sự kiện chính khi kích vào nút “Huỷ” hệ thống sẽ quay lại mục 3a. Use case kết thúc. | | Người quản lý, hệ thống | Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với file lưu trữ dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. | |
| Hậu điều kiện | Thông tin được cập nhật file lưu trữ dữ liệu |
| Điểm mở rộng | Không có |

*Bảng 2.5 Use case Quản lý phòng ban*

##### **2.2.2.3.6. Tra cứu và tìm kiếm**

f. Tra cứu và tìm kiếm (Người quản trị (Admin) / người quản lý phòng ban)

|  |  |
| --- | --- |
| Tên use case | Tra cứu và tìm kiếm |
| Tác nhân | Người quản trị (Admin) / người quản lý phòng ban |
| Mô tả | Use case này cho phép người dùng tìm kiếm thông tin về nhân viên ,phòng ban hoặc tài khoản. |
| Sự kiện kích hoạt chức năng | Use case bắt đầu khi người quản lý nhập thông tin vào bộ lọc dữ liệu tìm kiếm để tìm kiếm đối tượng cần tìm (nhân viên hoặc phòng ban). |
| Tiền điều kiện | Cần phải đăng nhập thành công tài khoản với tư cách Người quản trị (Admin) / người quản lý phòng ban. |
| Luồng sự kiện chính | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản trị (Admin) / người quản lý phòng ban | Người quản lý kích chọn 1 trong 2 lựa chọn trên và kích vào nút “Tìm kiếm”. | | Hệ thống | Hệ thống hiển thị thông tin nhân viên hoặc phòng ban có mã hoặc tên phù hợp bắt đầu bằng từ khóa tìm kiếm hoặc ký tự phù hợp với các thuộc tính của tài khoản. | | Hệ thống | Use case kết thúc. | |
| Luồng sự kiện thay thế | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản trị (Admin) / người quản lý phòng ban | Tại bước 2 trong luồng sự kiện chính nếu người quản lý kích vào nút “hủy bỏ”. Hệ thống hiển thị lên màn hình giao diện trang chủ chính của hệ thống. Use case kết thúc. | |
| Hậu điều kiện | Không có |
| Điểm mở rộng | Không có |

*Bảng 2.6 Use case Tra cứu và tìm kiếm (Người quản trị (Admin) / người quản lý phòng ban)*

##### **2.2.2.3.7. Thao Tác Trên File Dữ Liệu**

g. Thao tác trên file dữ liệu (Người quản trị (Admin) / người quản lý phòng ban)

|  |  |
| --- | --- |
| Tên use case | Thao tác trên file dữ liệu |
| Tác nhân | Người quản lý |
| Mô tả | Use case này cho phép Người quản trị (Admin) / người quản lý phòng ban lưu, đọc, đóng và mở file dữ liệu thông tin về nhân viên hoặc phòng ban. |
| Sự kiện kích hoạt chức năng | Use case bắt đầu khi Người quản trị (Admin) / người quản lý phòng ban nhấp vào các lựa chọn thao tác trên file dữ liệu (lưu, đọc, đóng và mở file dữ liệu). |
| Tiền điều kiện | Cần phải đăng nhập thành công tài khoản với tư cách Người quản trị (Admin) / người quản lý phòng ban. |
| Luồng sự kiện chính | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản trị (Admin) / người quản lý phòng ban | Người quản lý kích chọn 1 trong các lựa chọn trên. | | Hệ thống | Hệ thống hiển thị giao diện quản lý tệp tin. | | Hệ thống | Use case kết thúc. | |
| Luồng sự kiện thay thế | |  |  | | --- | --- | | Thực hiện bởi | Hành động | | Người quản lý, hệ thống | Tại bước 2 trong luồng sự kiện chính nếu người quản lý kích vào nút “hủy bỏ”. Hệ thống hiển thị lên màn hình giao diện trang chủ chính của hệ thống. Use case kết thúc. | |
| Hậu điều kiện | Không có |
| Điểm mở rộng | Không có |

*Bảng 2.7 Use case thao tác trên file dữ liệu (Người quản lý)*

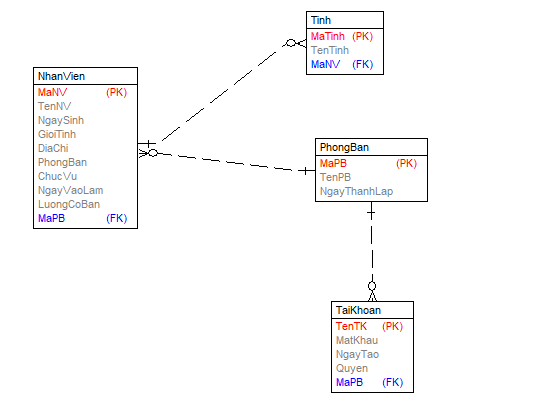
### **2.2.2. Mô hình hóa dữ liệu**.

### 

#### **2.2.2.1. Các yêu cầu dữ liệu**

* + Nhân viên gồm mã nhân viên, tên nhân viên, ngày sinh, giới tính, địa chỉ, phòng ban, chức vụ ngày vào làm, lương cơ bản. Mỗi phòng ban có thể chứa 1 hoặc nhiều nhân viên. Mỗi nhân viên có thể ở trong 0 hoặc 1 phòng ban.
  + Phòng ban gồm mã phòng ban, tên phòng ban, ngày thành lập.
  + Tỉnh gồm mã tỉnh, tên tỉnh. Mỗi nhân viên chỉ có địa chỉ thuộc về 1 và chỉ 1 tỉnh. Mỗi tỉnh có thể có 0,1 hoặc nhiều nhân viên.
  + Tài khoản gồm có tên tài khoản, mật khẩu , quyền , ngày tạo.

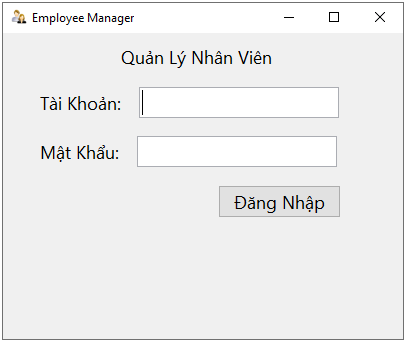
#### **2.2.2.2. Biểu đồ thực thể liên kết mức logic**



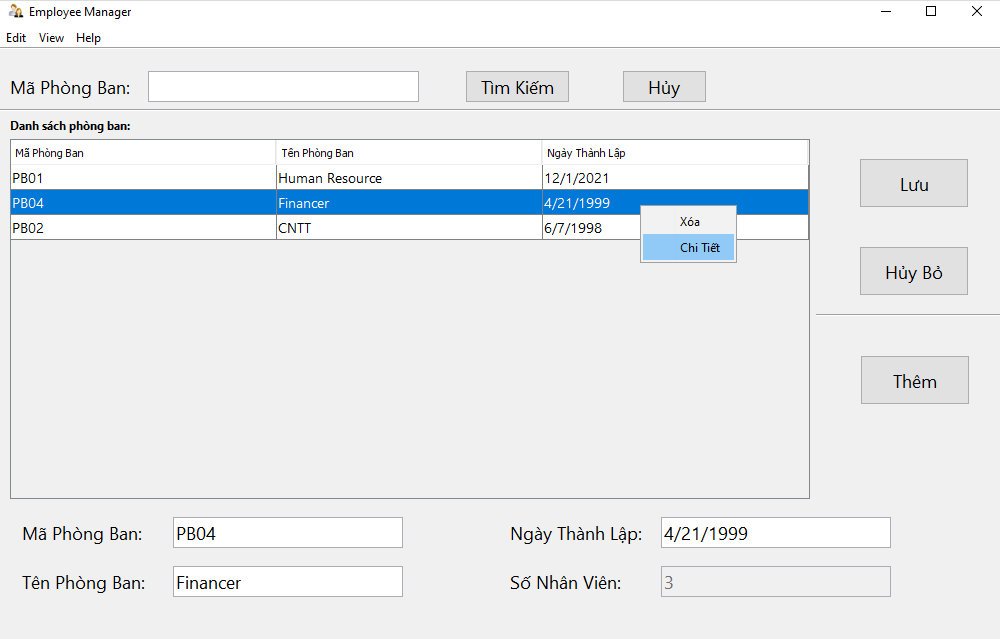
*Hình 2.0 Biểu đồ thực thể liên kết mức logic*

#### **2.2.2.3. Giao diện hệ thống**

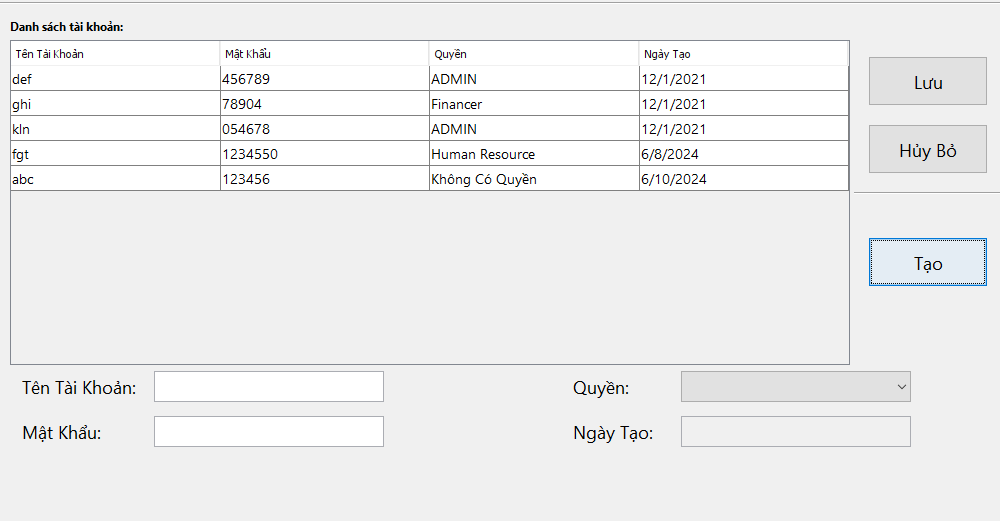
#### 



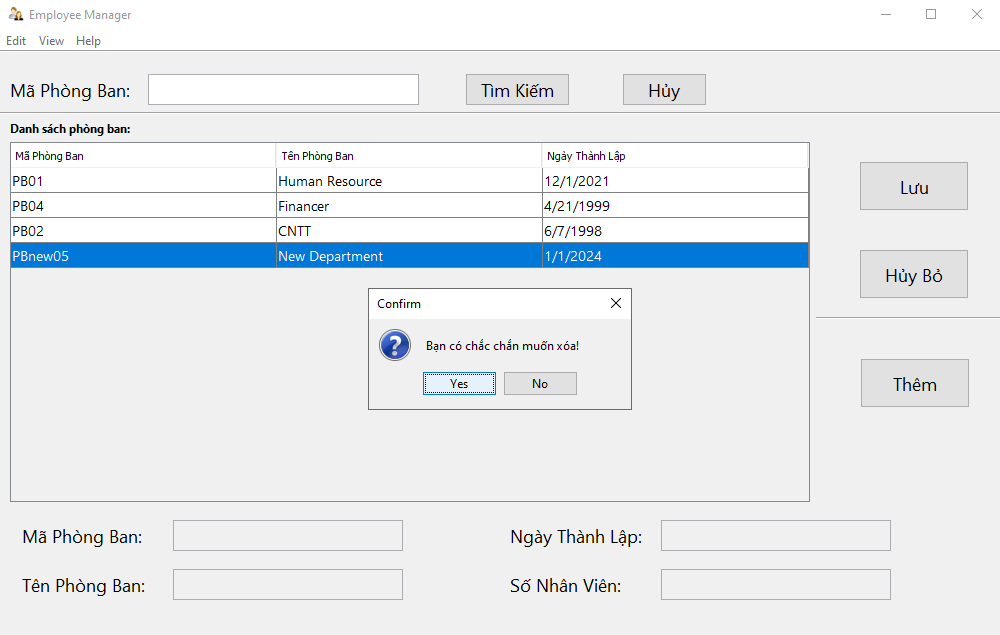
*Hình 2.1. Giao diện khi mới vào*



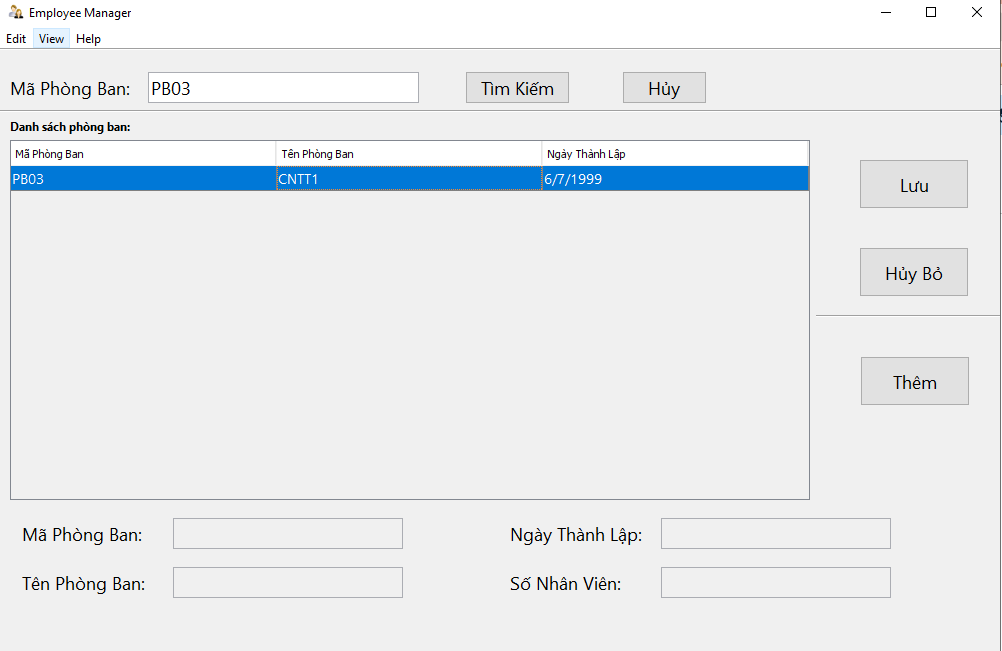
*Hình 2.2. Giao diện quản lý phòng ban(người quản trị Admin)*



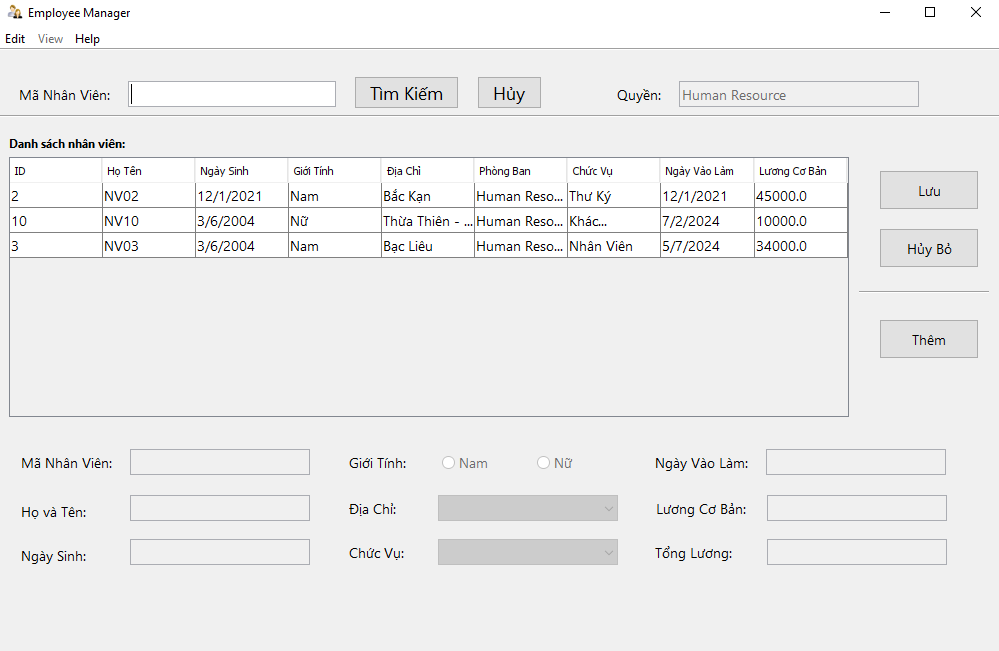
*Hinh 2.3 Giao diện Tạo Tài Khoản Cho Quản Lý Phòng Ban(người quản trị Admin)*



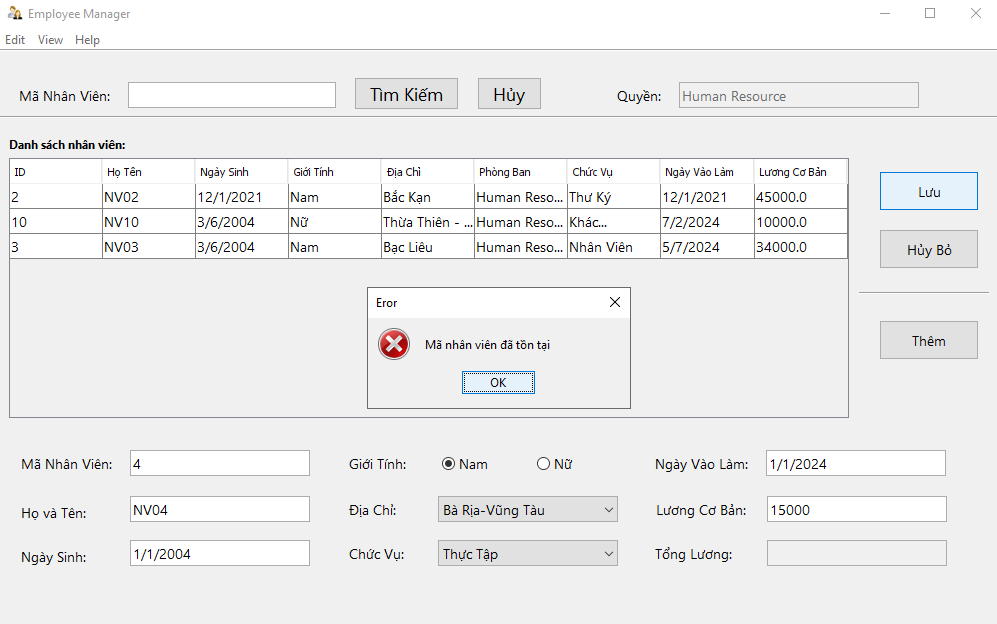
*Hình 2.4. Giao diện Xóa Phòng Ban(người quản trị Admin)*



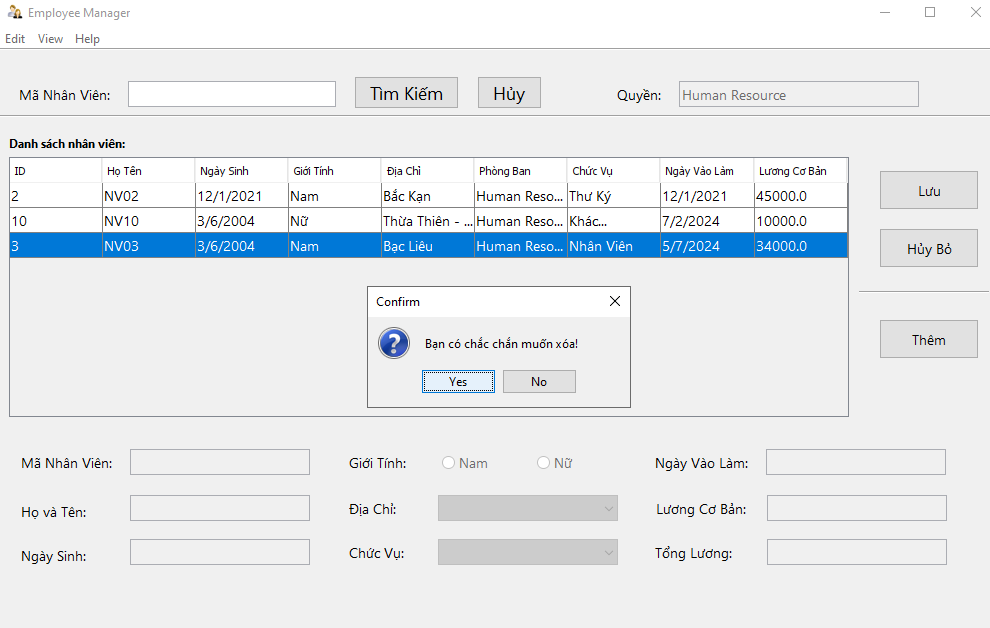
*Hình 2.5. Giao diện Tìm kiếm Phòng Ban dựa trên thuộc tính Mã Phòng Ban (người quản trị Admin)*

****

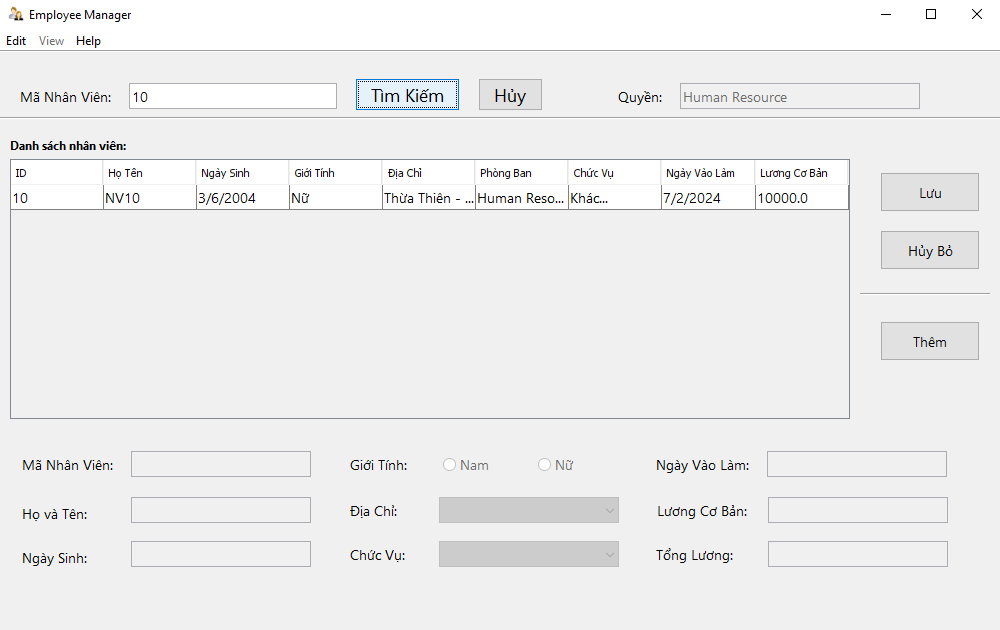
*Hình 2.6. Giao diện quản lý nhân viên của Quản Lý (Quản lý Phòng Ban đó)*



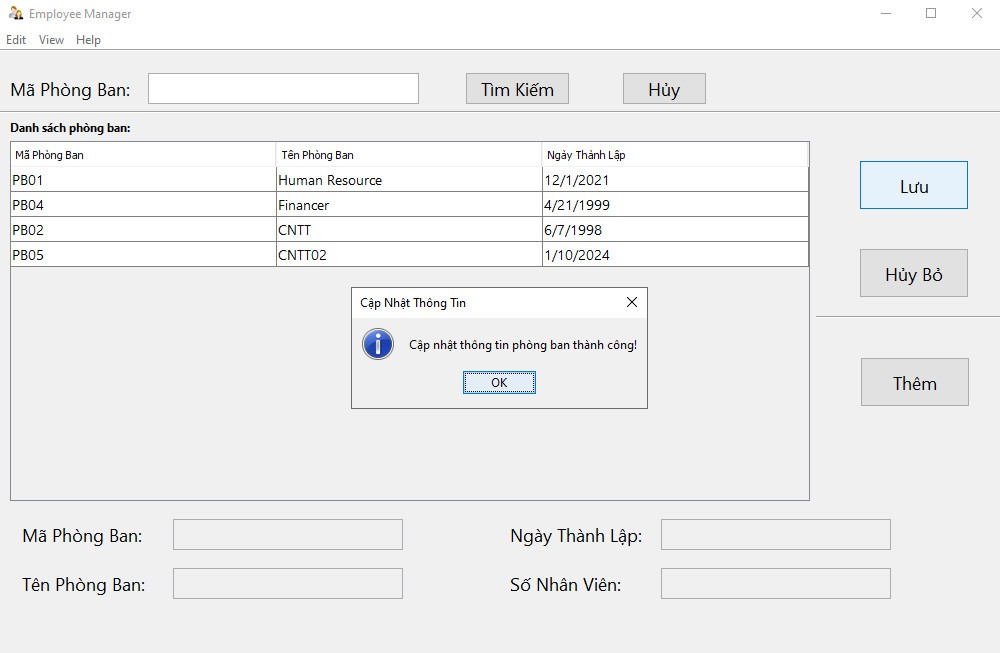
*Hình 2.7. Giao diện thêm thông tin nhân viên vào phòng ban (Quản lý Phòng Ban đó)*



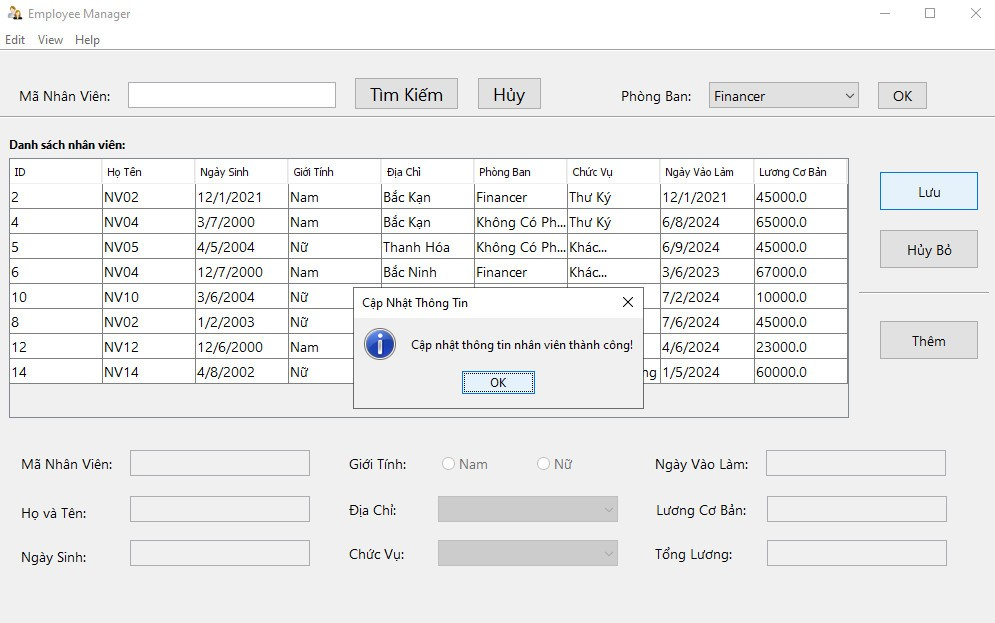
*Hình 2.8. Giao diện Xóa thông tin nhân viên ra khỏi phòng ban của Quản Lý (Quản lý Phòng Ban đó)*

****

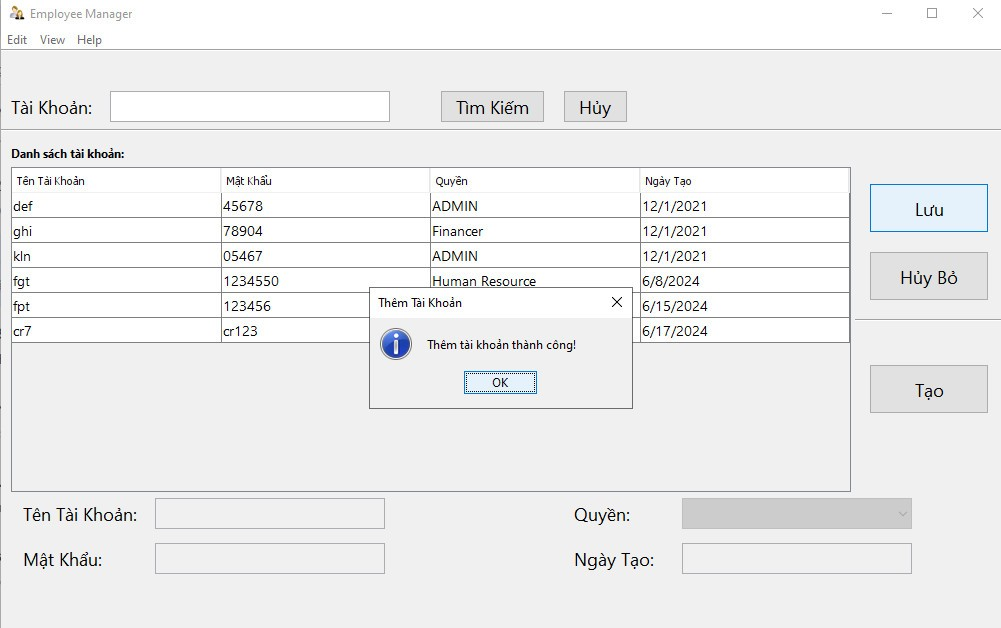
*Hình 2.9. Giao diện Tìm kiếm thông tin nhân viên dựa trên thuộc tính Mã Nhân Viên(Quản Lý của phòng ban đó)*



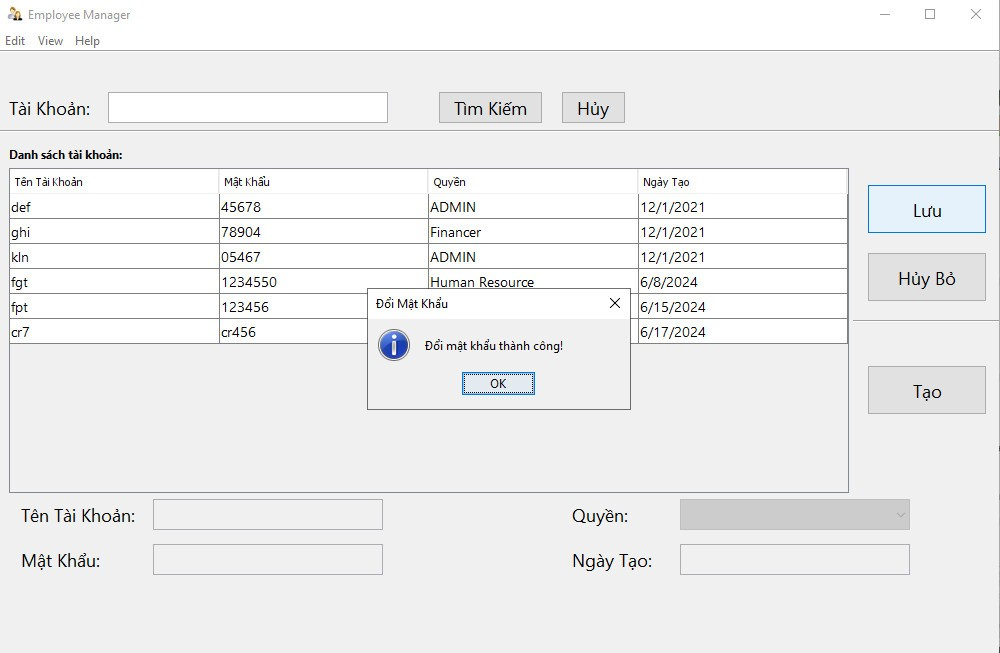
### *Hình 2.10. Giao diện cập nhật thông tin phòng ban**(admin)*



### *Hình 2.11. Giao diện cập nhật thông tin nhân viên*



### *Hình 2.12. Giao diện tạo tài khoản(admin)*



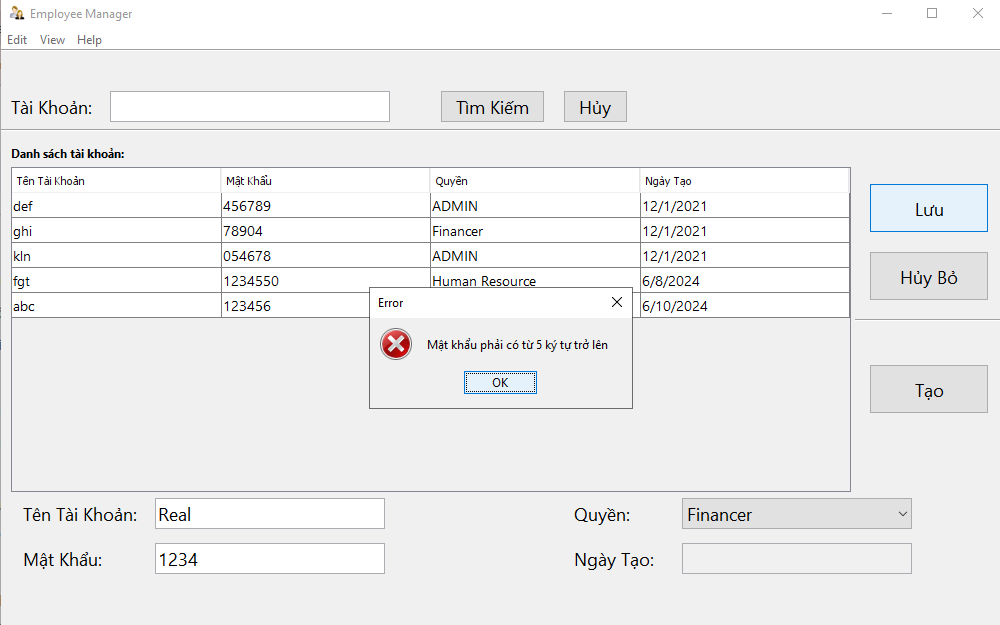
*Hình 2.13. Giao diện đổi mật khẩu (admin)*

### **2.3. Thực hiện bài toán**

#### **2.3.1. Đối với người quản trị viên (Admin)**

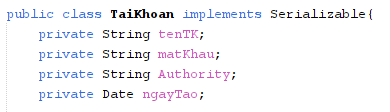
#### 

##### *2.3.1.1. Tạo tài khoản cho Quản Lý Phòng Ban*



*Hình 3.1 Giao diện tạo tài khoản cho Quản Lý ( Admin )*

* **Nội dung nhập/xuất:**Người quản trị viên (Admin) nhập đầy đủ thông tin về tài khoản Quản Lý bao gồm : Tên tài khoản , mật khẩu , chọn quyền , ngày tạo. Mật khẩu của người dùng phải có ít nhất 5 ký tự trở lên. Đăng ký thành công Quản Lý có thể dùng tên tài khoản và mật khẩu vừa đăng ký để bắt đầu quản lý nhân viên của phòng ban của tài khoản vừa tạo.
* **Hướng đối tượng:** Định nghĩa lớp TaiKhoan với các thuộc tính tên tài khoản, mật khẩu , Authority , Ngày Tạo . Lớp này có các phương thức khởi tạo, các phương thức truy cập (getter) và thiết lập (setter) cho các thuộc tính, và một phương thức toString() được ghi đè để in ra thông tin của đối tượng TaiKhoan



*Hình 3.2 Các thuộc tính của lớp TaiKhoan*

* **Bắt lỗi và gom rác:**

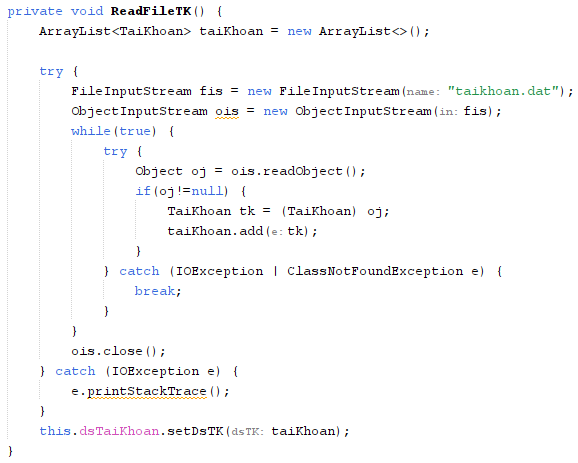
- Đoạn code sẽ kiểm tra xem người dùng đã nhập đầy đủ các thông tin yêu cầu chưa. Nếu người dùng còn để trống hệ thống hoặc nhập thông tin tài khoản mật không đúng với dữ liệu tài khoản được lưu trữ trong file sẽ hiện thông báo “Tài Khoản hoặc Mật Khẩu bị sai” bằng lệnh this.view.label\_Error.setText("Tài khoản hoặc mật khẩu bị sai")

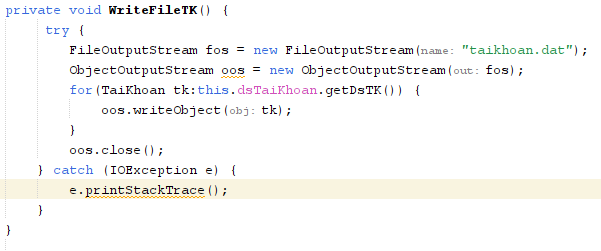
- Nếu người dùng nhập thông tin mật khẩu của tài khoản không đúng với định dạng yêu cầu sẽ hiển thị JOptionPane thông báo “mật khẩu phải có 5 ký tự trở lên ” .



*Hình 3.3 Bắt lỗi khi thực hiện tạo tài khoản*

* **Thao tác file:** Các hàm ReadFileTK() và WriteFileTK() dùng để đọc và ghi danh sách TaiKhoan.

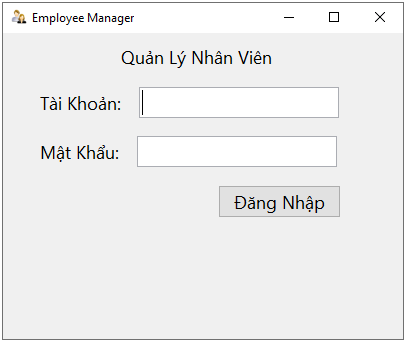
****

****

*Hình 3.4 Thao tác đọc ghi dữ liệu với file taikhoan.dat*

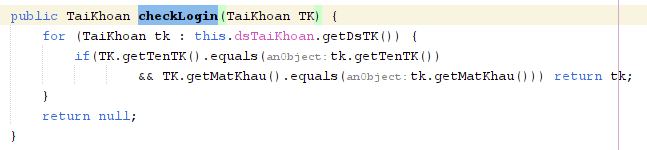
##### 

##### *2.3.1.2. Đăng Nhập*



*Hình 3.5 Giao diện Đăng nhập (Admin)*

* **Nội dung nhập/xuất:** Để đăng nhập người dùng nhập vào tên tài khoản và mật khẩu. Tên tài khoản và mật khẩu của người dùng là tên và mật khẩu mà người dùng đã đăng ký bởi người quản trị ( Admin ) trước đó. Sau khi đăng nhập thành công sẽ hiển thị lên màn hình giao diện trang chủ chính.
* **Hướng đối tượng:** Lớp TaiKhoan được sử dụng để thể hiện thông tin của người dùng. Phương thức checkLogin(TaiKhoan TK) thể hiện việc lấy thông tin tài khoản trong File lưu trữ .



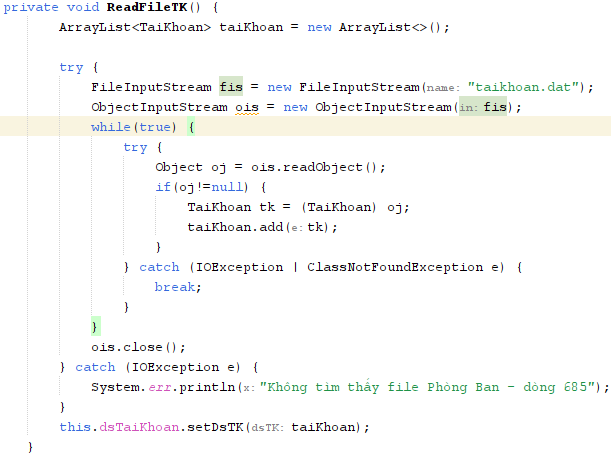
*Hình 3.6. Phương thức Đăng nhập (Người dùng)*

* **Bắt lỗi và gom rác:**
  + Nếu đăng nhập thành công một phiên làm việc mới được bắt đầu bằng cách gọi checkLogin(TK) để kiểm tra thông tin về tài khoản đăng nhập. Sau đó, một cửa sổ bảng điều khiển chính (Dashboard) được tạo và hiển thị cho người dùng.
  + Nếu đăng nhập không thành công. Hệ thống sẽ hiện thị thông báo "Tài khoản hoặc mật khẩu bị sai", “Lỗi” cho người dùng thông qua JLabel\_Error. Thông báo này có thể thông báo cho người dùng biết rằng thông tin đăng nhập không hợp lệ hoặc có lỗi xảy ra trong quá trình xác minh đăng nhập.



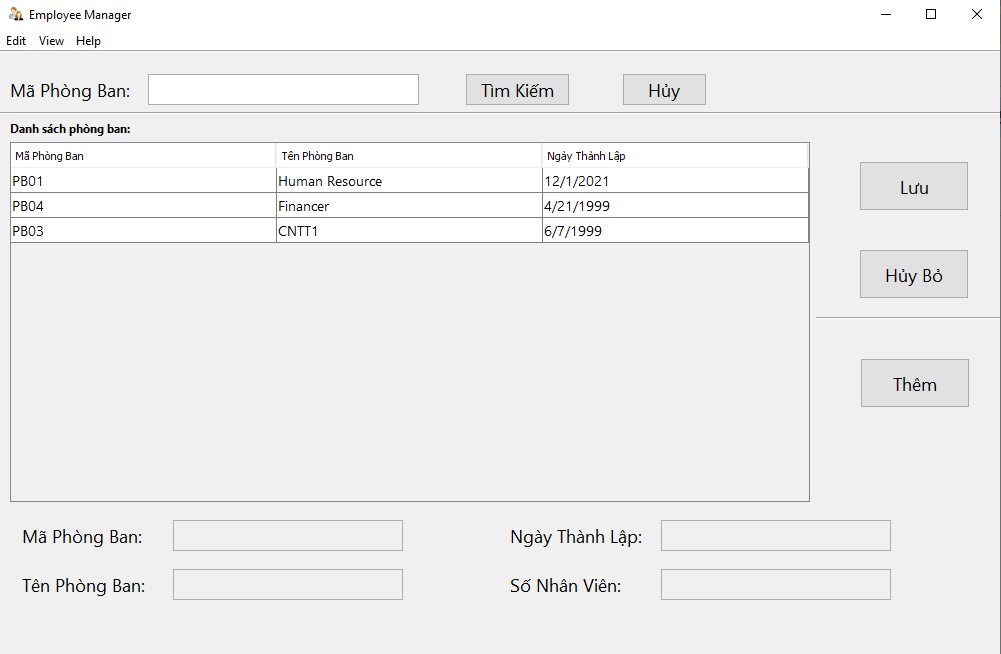
*Hình 3.7 Bắt lỗi khi thực hiện Đăng nhập*

* **Thao tác file:** Sử dụng phương thức readObject() của phương thức ReadFileTK() trong thư viện Java Core để đọc dữ liệu từ tệp lưu trữ dữ liệu tài khoản taikhoan.dat và sau đó kiểm tra thông tin đăng nhập trong danh sách các tài khoản từ tệp .



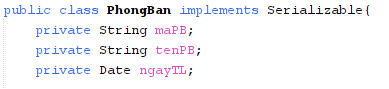
*Hình 3.8 Thao tác đọc file taikhoan.dat khi Đăng nhập*

##### *2.3.1.3. Quản lý phòng ban*



*Hình 3.9 Giao diện Quản Lý Phòng Ban*

* **Nội dung nhập/xuất:** Nhập thông tin phòng ban . Sau khi thay đổi click nút “Lưu” ngay lập tức thông tin phòng ban sẽ được cập nhật vào hệ thống. Nếu click nút “Huỷ Bỏ” thì hệ thông vẫn sẽ huỷ những thông tin đang nhập , ngăn truy cập vào trường nhập liệu và làm sạch trường nhập.
* **Hướng đối tượng:** Lớp PhongBan được sử dụng để biểu diễn thông tin của một Phòng Ban. Lớp DSPhongBan dùng để thể hiện và quản lý danh sách các phòng ban trong giao diện của tài khoản có quyền Admin.





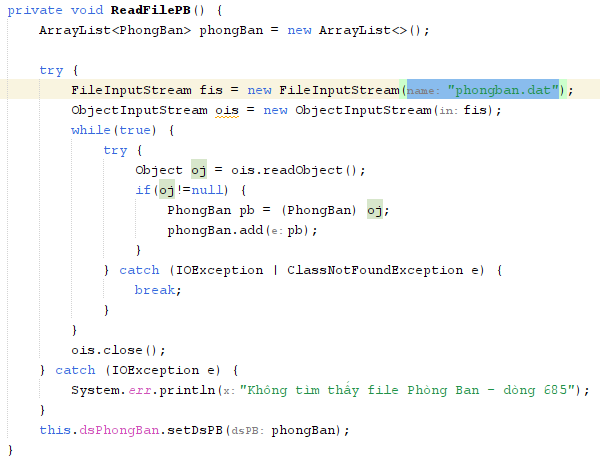
*Hình 3.10 Các thuộc tính của lớp PhongBan và lớp DSPhongBan*

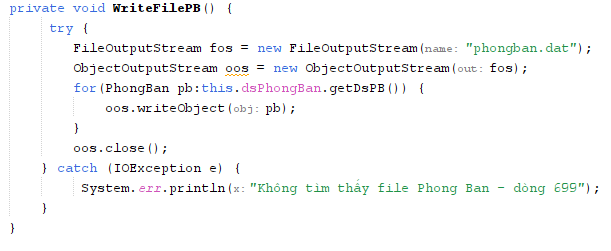
* **Bắt lỗi và gom rác:** 
  + Đầu tiên, mã kiểm tra xem có trường nào trong các trường nhập liệu (Mã Phòng Ban , Ngày Thành Lập , Tên Phòng Ban) bị bỏ trống hay không.
  + Nếu có bất kỳ trường nào bị bỏ trống, hệ thống sẽ hiển thị một hộp thoại cảnh báo (JOptionPane.showMessageDialog) với thông báo "Lỗi dữ liệu" và dừng việc thực hiện.
* Nếu tất cả các trường đều đã được nhập, mã sẽ tiến hành thêm đối tượng PhongBan mới với các giá trị thuộc tính mới từ các trường nhập liệu:Mã Phòng Ban , Ngày Thành Lập , Tên Phòng Ban.



*Hình 3.11 Bắt lỗi khi thực hiện thông tin phòng ban*

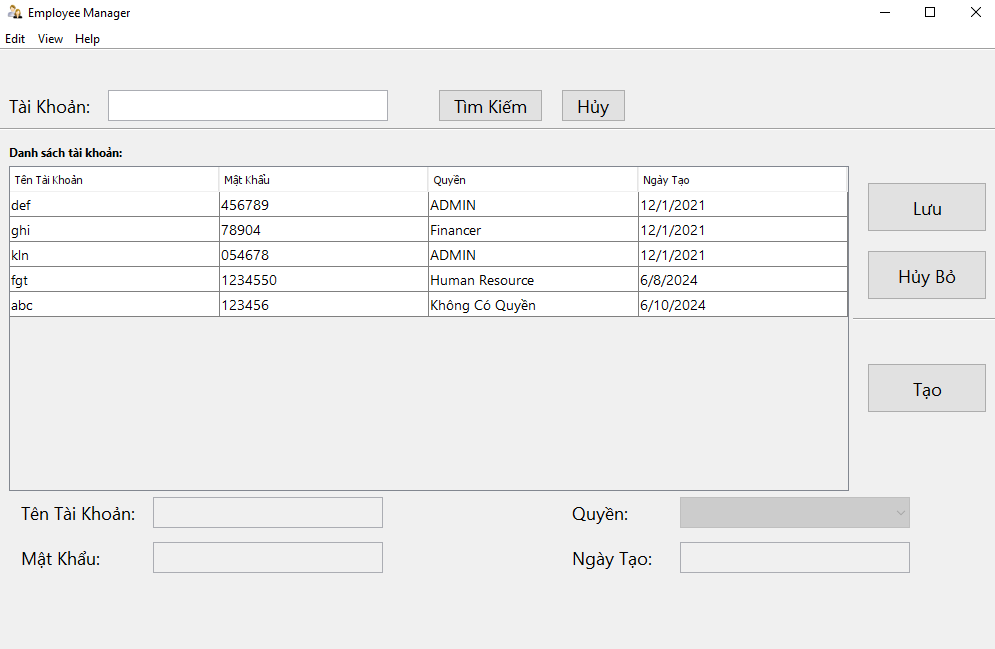
* Nếu trước khi nhấp nút “Lưu” mà biến trạng thái “status” đang lưu có giá trị là : “Thêm” . Hệ thống sẽ thực hiện thêm 1 phòng ban mới với các giá trị thuộc tính trong trường nhập liệu vào danh sách phòng ban . Khi thêm phòng ban , nếu thông tin phòng ban mới nhập có dữ liệu có sự trùng lặp , hệ thống sẽ hiển thị một hộp thoại cảnh báo (JOptionPane.showMessageDialog) với thông báo ""Mã phòng ban đã tồn tại"" và dừng việc thực hiện.
* Nếu trước khi nhấp nút “Lưu” mà biến trạng thái “status” đang lưu có giá trị là : “Chi tiết” . Hệ thống sẽ thực hiện cập nhật thông tin cho 1 phòng ban được chọn trong danh sách phòng ban . Khi cập nhật phòng ban , nếu thông tin phòng ban mới nhập có dữ liệu có sự trùng lặp , hệ thống sẽ hiển thị một hộp thoại cảnh báo (JOptionPane.showMessageDialog) với thông báo ""Mã phòng ban đã tồn tại"" và dừng việc thực hiện.
* **Thao tác file:** Đoạn code sử dụng phương thức ois.readObject() và oos.writeObject(PhongBan pb) để đọc/ghi danh sách các thông tin về phòng ban dùng file lưu trữ tên "phongban.dat".





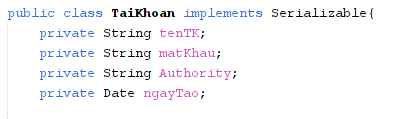
*Hình 3.12 Thao tác Đọc/Ghi dữ liệu file phongban.dat*

##### *2.3.1.4. Quản lý tài khoản*



*Hình 3.13 Giao diện Quản Lý Tài Khoản(Admin)*

* **Nội dung nhập/xuất:** Nhập thông tin tài khoản. Sau khi thay đổi click nút “Lưu” ngay lập tức thông tin tài khoản sẽ được cập nhật vào hệ thống. Nếu click nút “Huỷ Bỏ” thì hệ thông vẫn sẽ huỷ những thông tin đang nhập , ngăn truy cập vào trường nhập liệu và làm sạch trường nhập. click nút “Quay lại” thì hệ thông vẫn lưu những thông tin đã đc lưu trước đó.
* **Hướng đối tượng:** Lớp TaiKhoan được sử dụng để biểu diễn thông tin của một Tài Khoản . Lớp DSTaiKhoan dùng để thể hiện và quản lý danh sách các tài khoản trong giao diện của tài khoản có quyền Admin.





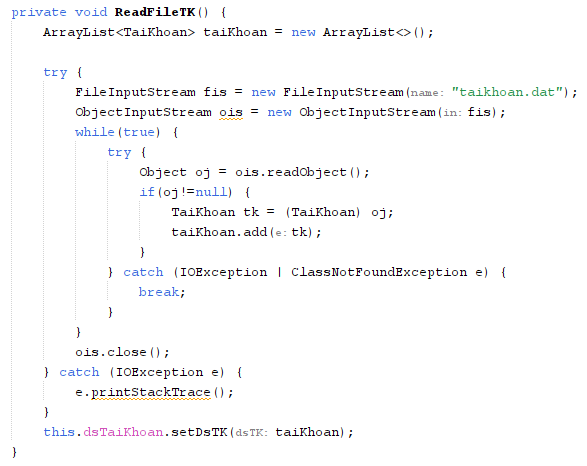
*Hình 3.14 Các thuộc tính của lớp TaiKhoan và lớp DSTaiKhoan*

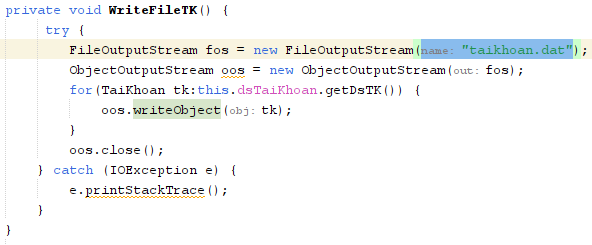
* **Bắt lỗi và gom rác:** 
  + Đầu tiên, mã kiểm tra xem có trường nào trong các trường nhập liệu (Tên Tài Khoản , Mật Khẩu , Quyền) bị bỏ trống hay không.
  + Nếu có bất kỳ trường nào bị bỏ trống, mã sẽ hiển thị một hộp thoại cảnh báo (JOptionPane.showMessageDialog) với thông báo "Lỗi dữ liệu" và dừng việc thực hiện.
* Nếu tất cả các trường đều đã được nhập, hệ thống sẽ tiến hành thêm đối tượng TaiKhoan mới với các giá trị thuộc tính mới từ các trường nhập liệu:Tên Tài Khoản , Mật Khẩu , Quyền.



*Hình 3.15 Bắt lỗi khi thực hiện thông tin tài khoản*

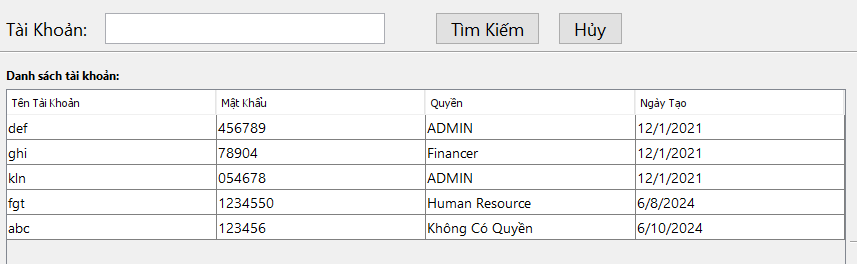
* Nếu trước khi nhấp nút “Lưu” mà biến trạng thái “status” đang lưu có giá trị là : “Tạo” . Hệ thống sẽ thực hiện thêm 1 tài khoản mới với các giá trị thuộc tính trong trường nhập liệu vào danh sách tài khoản . Khi thêm tài khoản , nếu thông tin phòng ban mới nhập có dữ liệu có sự trùng lặp , hệ thống sẽ hiển thị một hộp thoại cảnh báo (JOptionPane.showMessageDialog) với thông báo ""Tài khoản đã tồn tại"" và dừng việc thực hiện.
* matKhau.length() là một biến kiểm tra độ dài của mật khẩu. Nếu matKhau.length() là < 5 , mật khẩu mới không đáp ứng yêu cầu về độ dài .Hệ thống hiển thị thông báo lỗi “Mật khẩu phải có từ 5 ký tự trở lên” và dừng quá trình .
* Nếu trước khi nhấp nút “Lưu” mà biến trạng thái “status” đang lưu có giá trị là : “Chi tiết” . Hệ thống sẽ thực hiện cập nhật thông tin cho 1 tài khoản được chọn trong danh sách tài khoản . Khi cập nhật tài khoản , nếu thông tin tài khoản mới nhập có dữ liệu có sự trùng lặp , hệ thống sẽ hiển thị một hộp thoại cảnh báo (JOptionPane.showMessageDialog) với thông báo ""Tài khoản đã tồn tại"" và dừng việc thực hiện.
* **Thao tác file:** Đoạn code sử dụng phương thức ois.readObject() và oos.writeObject(TaiKhoan tk) để đọc/ghi danh sách các thông tin về tài khoản "taikhoan.dat".



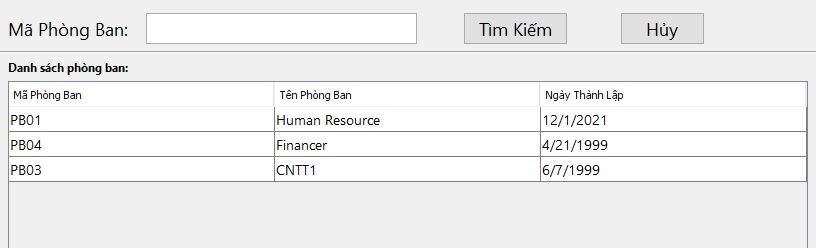


*Hình 3.16 Thao tác Đọc/Ghi dữ liệu file taikhoan.dat*

##### *2.3.1.5. Tra cứu và tìm kiếm tài khoản ( hoặc phòng ban)*

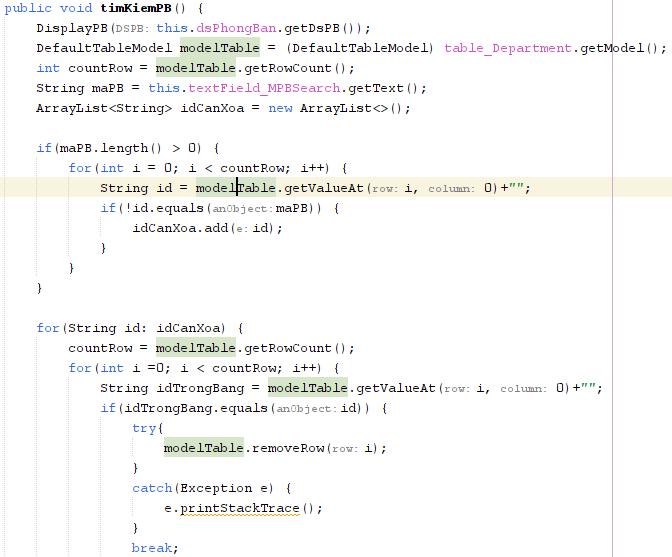


*Hình 3.17 Giao diện tìm kiếm Tài Khoản*

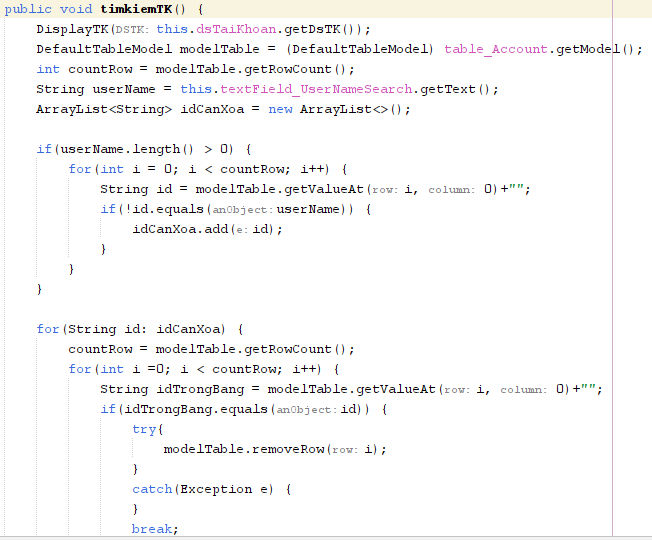


*Hình 3.18 Giao diện tìm kiếm Phòng Ban*

* **Nội dung nhập/xuất:** Người dùng nhập từ khóa cần tìm kiếm vào thanh tìm kiếm. Sau khi nhấn click nút “Tìm kiếm” hệ thống sẽ hiển thị những thông tin liên quan về từ khóa lên màn hình.
* **Hướng đối tượng:** Lớp TaiKhoan và PhongBan được sử dụng để biểu diễn thông tin về một đối tượng . Phương thức timkiemTK() và timKiemPB() dùng để thể hiện hành vi của việc tìm kiếm thông tin của tài khoản hoặc phòng ban.

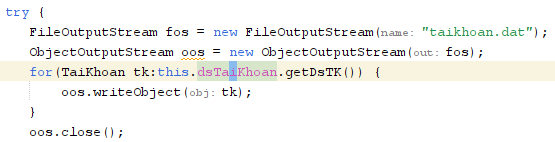


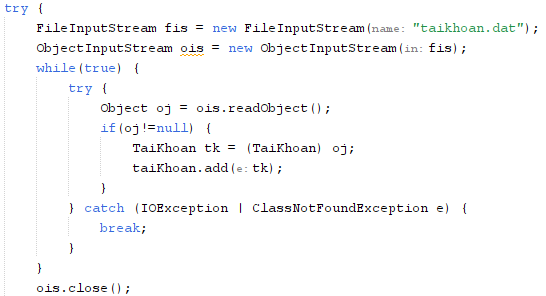
*Hình 3.19 Gọi phương thức timKiemPB() để tìm kiếm toàn phòng ban*



*Hình 3.20 Gọi phương thức timkiemTK() để tìm kiếm toàn tài khoản*

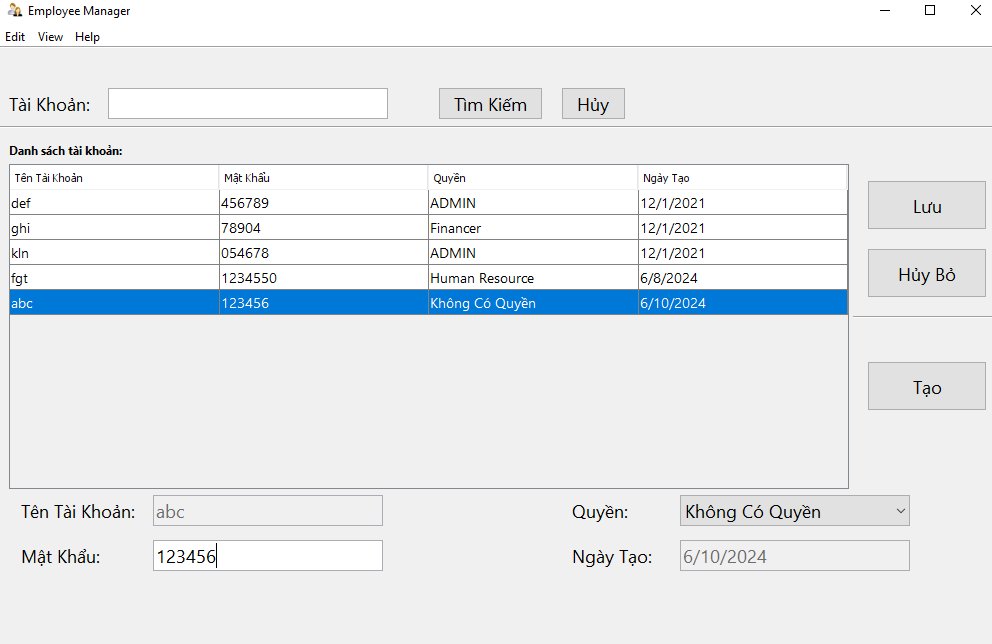
* **Bắt lỗi và gom rác:**
  + Lấy từ khóa tìm kiếm từ trường văn bản nhập nội dung tìm kiếm và loại bỏ các khoảng trắng ở đầu và cuối chuỗi bằng cách sử dụng phương thức trim().
  + Nếu người dùng không nhập gì, một hộp thoại thông báo lỗi sẽ xuất hiện để yêu cầu người dùng nhập từ khóa tìm kiếm.
  + Nếu người dùng đã nhập, hàm timkiemTK() hoặc timKiemPB() sẽ được gọi để tìm kiếm các đối tượng phù hợp với từ khóa. Kết quả trả về là một danh sách các đối tượng có liên quan.
  + Nếu danh sách kết quả rỗng, bảng hiển thị kết quả table sẽ được xóa hết các hàng hiện có.
  + Nếu danh sách kết quả không rỗng, hệ thống sẽ cập nhật bảng hiển thị với danh sách kết quả mới.
* **Thao tác file:** 
  + Đoạn mã này thực hiện thao tác đọc tệp văn bản để tìm kiếm thông tin về sách trong tệp dữ liệu**.**Sử dụng một đối tượng ObjectInputStream và FileInputStream để đọc từng đối tượng trong tệp văn bản .Sử dụng FileOutputStream và ObjectOutputStream để ghi từng đối tượng vào file lưu trữ "taikhoan.dat" hoặc "phongban.dat".





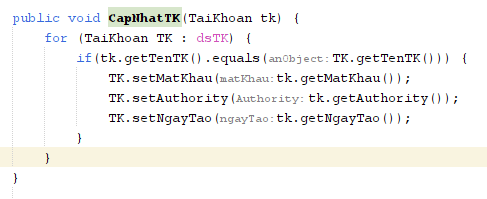
*Hình 3.21 Đọc và Ghi dữ liệu đối tượng từ taikhoan.dat*

##### *2.3.1.6. Sửa thông tin của Tài Khoản (hoặc Phòng Ban)*



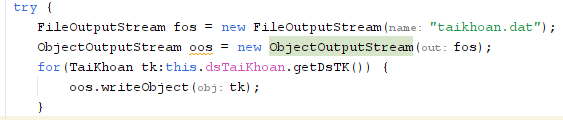
*Hình 3.22 Giao diện cập nhật thông tin tài khoản*

* **Nội dung nhập/xuất:** Người dùng nhập thông tin mới cho đối tượng cần cập nhật vào các trường nhập liệu. Sau khi nhấn click nút “Lưu” hệ thống sẽ tự động cập nhật những thông tin mới về đối tượng lên màn hình.
* **Hướng đối tượng:** Lớp DSTaiKhoan(DSPhongBan) được sử dụng để gọi đến các phương thức setter() và getter() để cập nhật thông tin của một phòng ban. Phương thức CapNhatTK(TaiKhoan tk) (CapNhatPB(String MPB, PhongBan pb)) dùng để thể hiện hành vi của việc thay đổi thông tin của Tài khoản ( hoặc phòng ban).



*Hình 3.23 Phương thức Chỉnh sửa Thông tin tài khoản*

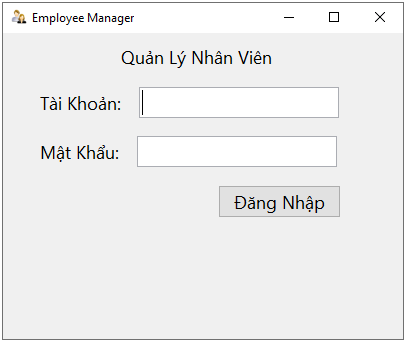
* **Thao tác file:** Đoạn code sử dụng phương thức writeObject() để ghi thông tin mới cập nhật lại của tài khoản (hoặc phòng ban) vào file chứa danh sách lưu trữ "taikhoan.dat" (hoặc “phongban.dat”).



*Hình 3.24 Thao tác Cập nhật lại dữ liệu taikhoan.dat*

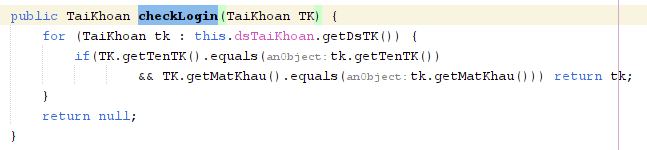
#### **2.3.2. Đối với người quản lý ( Quản lý phòng ban)**

##### *2.3.1.2. Đăng Nhập*



*Hình 3.25 Giao diện Đăng nhập ( Quản lý phòng ban)*

* **Nội dung nhập/xuất:** Để đăng nhập người dùng nhập vào tên tài khoản và mật khẩu. Tên tài khoản và mật khẩu của người dùng là tên và mật khẩu mà người dùng đã đăng ký bởi người quản trị ( Admin ) trước đó để cho người quản lý phòng ban sử dụng( Quản lý phòng ban). Sau khi đăng nhập thành công sẽ hiển thị lên màn hình giao diện trang chủ chính.
* **Hướng đối tượng:** Lớp TaiKhoan được sử dụng để thể hiện thông tin của người dùng. Phương thức checkLogin(TaiKhoan TK) thể hiện việc lấy thông tin tài khoản trong File lưu trữ .



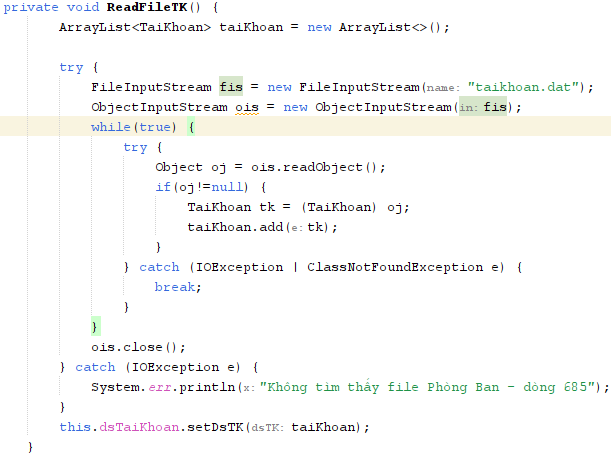
*Hình 3.26. Phương thức Đăng nhập (Người dùng)*

* **Bắt lỗi và gom rác:**
  + Nếu đăng nhập thành công một phiên làm việc mới được bắt đầu bằng cách gọi checkLogin(TK) để kiểm tra thông tin về tài khoản đăng nhập. Sau đó, một cửa sổ bảng điều khiển chính (Dashboard) được tạo và hiển thị cho người dùng.
  + Nếu đăng nhập không thành công. Hệ thống sẽ hiện thị thông báo "Tài khoản hoặc mật khẩu bị sai", “Lỗi” cho người dùng thông qua JLabel\_Error. Thông báo này có thể thông báo cho người dùng biết rằng thông tin đăng nhập không hợp lệ hoặc có lỗi xảy ra trong quá trình xác minh đăng nhập.



*Hình 3.27 Bắt lỗi khi thực hiện Đăng nhập*

* **Thao tác file:** Sử dụng phương thức readObject() của phương thức ReadFileTK() trong thư viện Java Core để đọc dữ liệu từ tệp lưu trữ dữ liệu tài khoản taikhoan.dat và sau đó kiểm tra thông tin đăng nhập trong danh sách các tài khoản từ tệp .



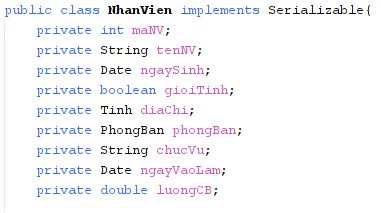
*Hình 3.28 Thao tác đọc file taikhoan.dat khi Đăng nhập*

##### *2.3.2.2. Quản lý nhân viên của phòng ban*

##### 

*Hình 3.29 Giao diện Quản Lý Nhân Viên ( Quản lý phòng ban)*

* **Nội dung nhập/xuất:** Nhập thông tin nhân viên . Sau khi thay đổi click nút “Lưu” ngay lập tức thông tin nhân viên sẽ được cập nhật vào hệ thống. Nếu click nút “Huỷ Bỏ” thì hệ thông vẫn sẽ huỷ những thông tin đang nhập , ngăn truy cập vào trường nhập liệu và làm sạch trường nhập.
* **Hướng đối tượng:** Lớp NhanVien được sử dụng để biểu diễn thông tin của một Phòng Ban. Lớp DSNhanVien dùng để thể hiện và quản lý danh sách các nhân viên trong giao diện của tài khoản có quyền người Quản Lý(Quản lý phòng ban).





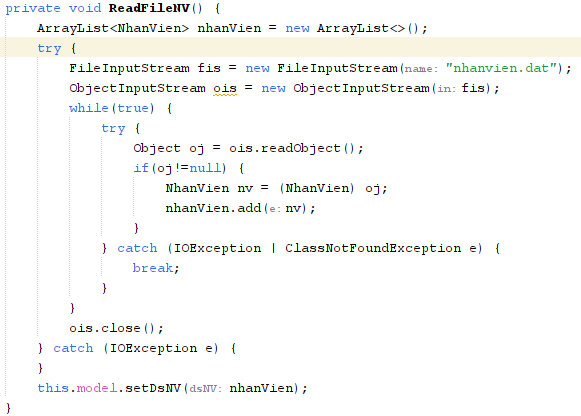
*Hình 3.30 Các thuộc tính của lớp NhanVien và lớp DSNhanVien*

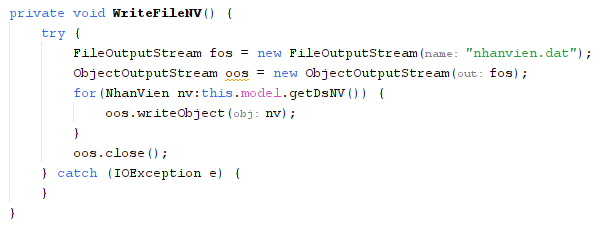
* **Bắt lỗi và gom rác:** 
  + Đầu tiên, mã kiểm tra xem có trường nào trong các trường nhập liệu (Mã Nhân Viên , Tên Nhân Viên , Ngày Sinh , Giới Tính , Địa Chỉ , Phòng Ban , Chức vụ , Ngày Vào Làm , Lương cơ bản) bị bỏ trống hay không.
  + Nếu có bất kỳ trường nào bị bỏ trống, hệ thống sẽ hiển thị một hộp thoại cảnh báo (JOptionPane.showMessageDialog) với thông báo "Lỗi dữ liệu" và dừng việc thực hiện.
* Nếu tất cả các trường đều đã được nhập, mã sẽ tiến hành thêm đối tượng NhanVien mới với các giá trị thuộc tính mới từ các trường nhập liệu:Mã Nhân Viên , Tên Nhân Viên , Ngày Sinh , Giới Tính , Địa Chỉ , Phòng Ban , Chức vụ , Ngày Vào Làm , Lương cơ bản.



*Hình 3.31 Bắt lỗi khi thực hiện thông tin nhân viên*

* Nếu trước khi nhấp nút “Lưu” mà biến trạng thái “status” đang lưu có giá trị là : “Thêm” . Hệ thống sẽ thực hiện thêm 1 nhân viên mới với các giá trị thuộc tính trong trường nhập liệu vào danh sách nhân viên . Khi thêm nhân viên , nếu thông tin nhân viên mới nhập có dữ liệu có sự trùng lặp , hệ thống sẽ hiển thị một hộp thoại cảnh báo (JOptionPane.showMessageDialog) với thông báo "Mã nhân viên đã tồn tại" và dừng việc thực hiện.
* Nếu trước khi nhấp nút “Lưu” mà biến trạng thái “status” đang lưu có giá trị là : “Chi tiết” . Hệ thống sẽ thực hiện cập nhật thông tin cho 1 nhân viên được chọn trong danh sách nhân viên . Khi cập nhật nhân viên , nếu thông tin nhân viên mới nhập có dữ liệu có sự trùng lặp , hệ thống sẽ hiển thị một hộp thoại cảnh báo (JOptionPane.showMessageDialog) với thông báo "Mã nhân viên đã tồn tại" và dừng việc thực hiện.
* **Thao tác file:** Đoạn code sử dụng phương thức ois.readObject() và oos.writeObject(NhanVien nv) để đọc/ghi danh sách các thông tin về nhân viên dùng file lưu trữ tên "nhanvien.dat".





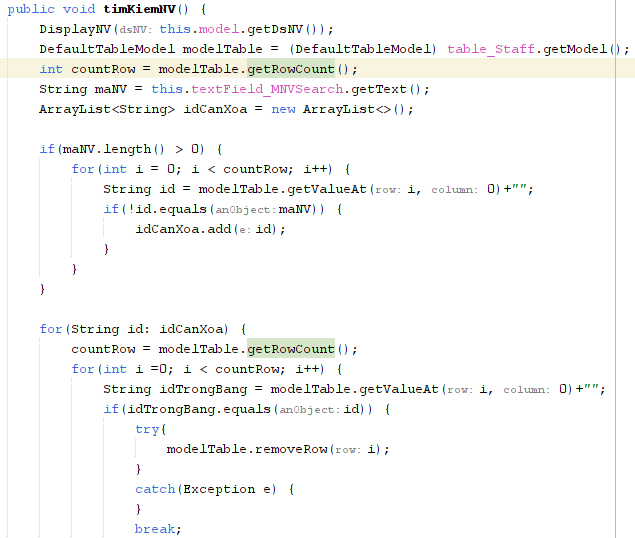
*Hình 3.32 Thao tác Đọc/Ghi dữ liệu file nhanvien.dat*

##### *2.3.1.4. Tra cứu và tìm kiếm nhân viên*



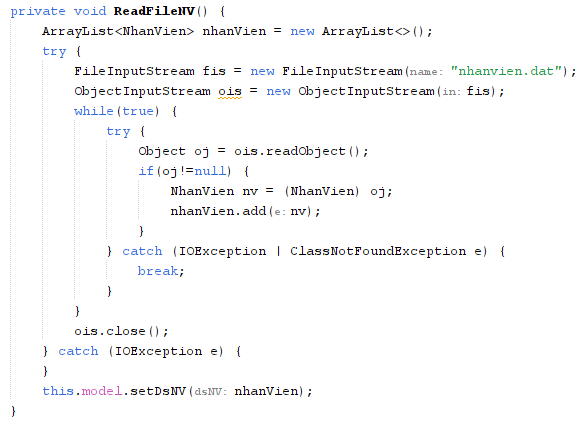
*Hình 3.33 Giao diện tìm kiếm Nhân Viên*

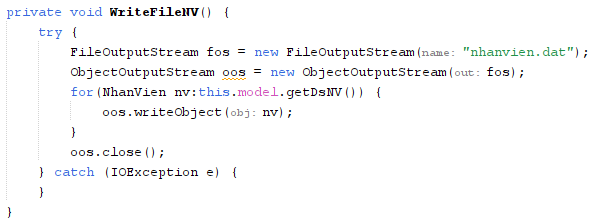
* **Nội dung nhập/xuất:** Người dùng nhập từ khóa cần tìm kiếm vào thanh tìm kiếm. Sau khi nhấn click nút “Tìm kiếm” hệ thống sẽ hiển thị những thông tin đối tượng liên quan về từ khóa lên màn hình.
* **Hướng đối tượng:** Lớp NhanVien được sử dụng để biểu diễn thông tin về một đối tượng Nhân Viên . Phương thức timkiemNV() dùng để thể hiện hành vi của việc tìm kiếm thông tin của nhân viên.



*Hình 3.34 Gọi phương thức timKiemNV() để tìm kiếm toàn nhân viên*

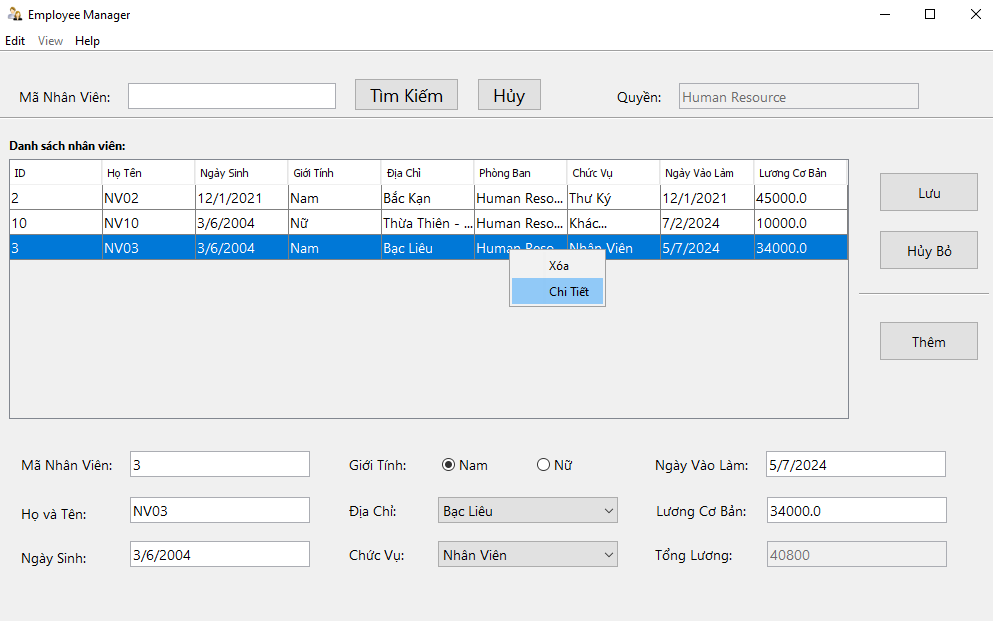
* **Bắt lỗi và gom rác:**
  + Lấy từ khóa tìm kiếm từ trường văn bản nhập nội dung tìm kiếm và loại bỏ các khoảng trắng ở đầu và cuối chuỗi bằng cách sử dụng phương thức trim().
  + Nếu người dùng không nhập gì, một hộp thoại thông báo lỗi sẽ xuất hiện để yêu cầu người dùng nhập từ khóa tìm kiếm.
  + Nếu người dùng đã nhập, hàm timkiemNV() sẽ được gọi để tìm kiếm các đối tượng phù hợp với từ khóa. Kết quả trả về là một danh sách các đối tượng có liên quan.
  + Nếu danh sách kết quả rỗng, bảng hiển thị kết quả table sẽ được xóa hết các hàng hiện có.
  + Nếu danh sách kết quả không rỗng, hệ thống sẽ cập nhật bảng hiển thị với danh sách kết quả mới.
* **Thao tác file:** 
  + Đoạn mã này thực hiện thao tác đọc tệp văn bản để tìm kiếm thông tin về sách trong tệp dữ liệu**.**Sử dụng một đối tượng ObjectInputStream và FileInputStream để đọc từng đối tượng trong tệp văn bản .Sử dụng FileOutputStream và ObjectOutputStream để ghi từng đối tượng vào file lưu trữ "nhanvien.dat".





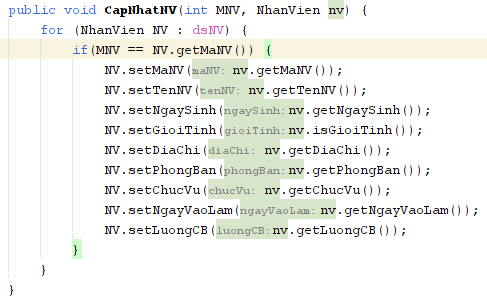
*Hình 3.35 Đọc và Ghi dữ liệu đối tượng nhân viên từ file nhanvien.dat*

##### *2.3.1.6. Sửa thông tin của Nhân Viên*



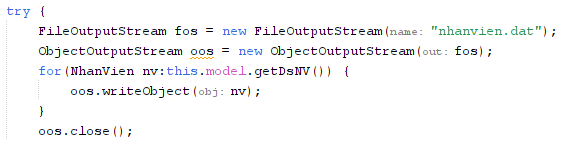
*Hình 3.36 Giao diện cập nhật thông tin nhân viên*

* **Nội dung nhập/xuất:** Người dùng nhập thông tin mới cho đối tượng cần cập nhật vào các trường nhập liệu. Sau khi nhấn click nút “Lưu” hệ thống sẽ tự động cập nhật những thông tin mới về đối tượng lên màn hình.
* **Hướng đối tượng:** Lớp DSNhanVien được sử dụng để gọi đến các phương thức setter() và getter() để cập nhật thông tin của một phòng ban. Phương thức CapNhatNV(int MNV, NhanVien nv) dùng để thể hiện hành vi của việc thay đổi thông tin của Nhân Viên.



*Hình 3.37 Phương thức Chỉnh sửa Thông tin nhân viên*

* **Thao tác file:** Đoạn code sử dụng phương thức writeObject() để ghi thông tin mới cập nhật lại của nhân viên vào file chứa danh sách lưu trữ "nhanvien.dat".



*Hình 3.38 Thao tác Cập nhật lại dữ liệu nhanvien.dat*

# 

# **Chương 3: KẾT LUẬN VÀ BÀI HỌC KINH NGHIỆM**

## **3.1. Nội dung đã thực hiện**

### **3.1.1. Kiến thức và kỹ năng đã học được**

* Hiểu và vận dụng các kỹ thuật phân tích và thiết kế hệ thống như mô hình hóa chức năng (use case), thao tác với tệp, file, mô hình hóa dữ liệu (biểu đồ thực thể liên kết).
* Thiết kế và xây dựng giao diện người dùng.
* Lập trình các chức năng chính của hệ thống như đăng nhập, quản lý nhân viên, …
* Áp dụng các kỹ năng làm việc nhóm, phân công công việc và lập kế hoạch.

### **3.1.2. Chuẩn đầu ra đạt được**

* Lập trình và triển khai được các chức năng chính của hệ thống.
* Thiết kế được giao diện hệ thống của người dùng
* Thực hành kỹ năng phân tích, thiết kế, lập trình và quản lý tiến độ.

### **3.1.3. Bài học kinh nghiệm**

* Tầm quan trọng của lên kế hoạch và phối hợp nhóm hiệu quả.
* Cách tiếp cận và giải quyết các vấn đề phát sinh trong quá trình triển khai, thực hiện bài tập lớn.

## **3.2. Hướng phát triển**

* Tính khả thi: Chủ đề về ***“Xây dựng phần mềm quản lý nhân viên”*** có tính khả thi cao vì đây là một vấn đề thực tế, cụ thể.
* Yêu cầu khi thực hiện:
  + Đề tài này yêu cầu người thực hiện phải có kiến thức về phân tích thiết kế hệ thống và một số kiến thức cơ bản về quản lý nhân viên.
  + Các kỹ năng cần có bao gồm: lập trình, thiết kế giao diện, phân tích yêu cầu, mô hình hóa chức năng,...
* Thuận lợi và khó khăn
  + Thuận lợi:
* Tăng cường hiệu suất: Phần mềm quản lý nhân viên giúp tổ chức tối ưu hóa quy trình làm việc, từ việc lập kế hoạch làm việc đến quản lý thời gian và đánh giá hiệu suất.
* Tiết kiệm thời gian và tài nguyên: Tích hợp các chức năng tự động hóa giúp giảm bớt công việc thủ công và giảm tải cho nhân viên quản lý.
* Tăng cường trải nghiệm nhân viên: Cung cấp một cổng thông tin dễ sử dụng và tiện lợi giúp nâng cao trải nghiệm làm việc của nhân viên.
* Tối ưu hóa quản lý dữ liệu: Phần mềm quản lý nhân viên giúp tổ chức quản lý thông tin nhân viên một cách hiệu quả và bảo mật.
* Tính linh hoạt và mở rộng: Có khả năng điều chỉnh và mở rộng phần mềm để phản ánh sự thay đổi trong tổ chức và môi trường kinh doanh.
  + Khó khăn:
* Chi phí và thời gian: Xây dựng một phần mềm quản lý nhân viên đòi hỏi đầu tư lớn về cả chi phí và thời gian phát triển.
* Yêu cầu kỹ thuật cao: Đòi hỏi kiến thức chuyên sâu về lập trình và phát triển phần mềm để tạo ra một hệ thống chất lượng.
* Quản lý thay đổi: Sự thay đổi quy trình làm việc và sự chuyển đổi sang một hệ thống mới có thể gây khó khăn và phản đối từ phía nhân viên.
* Bảo mật và tuân thủ quy định: Cần phải đảm bảo rằng phần mềm được phát triển và triển khai tuân thủ các quy định bảo mật và pháp lý.
* Hỗ trợ và bảo trì: Sau khi triển khai, cần phải có các biện pháp hỗ trợ và bảo trì đảm bảo rằng phần mềm luôn hoạt động một cách trơn tru và hiệu quả.

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 