

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**======\*\*\*======**

****

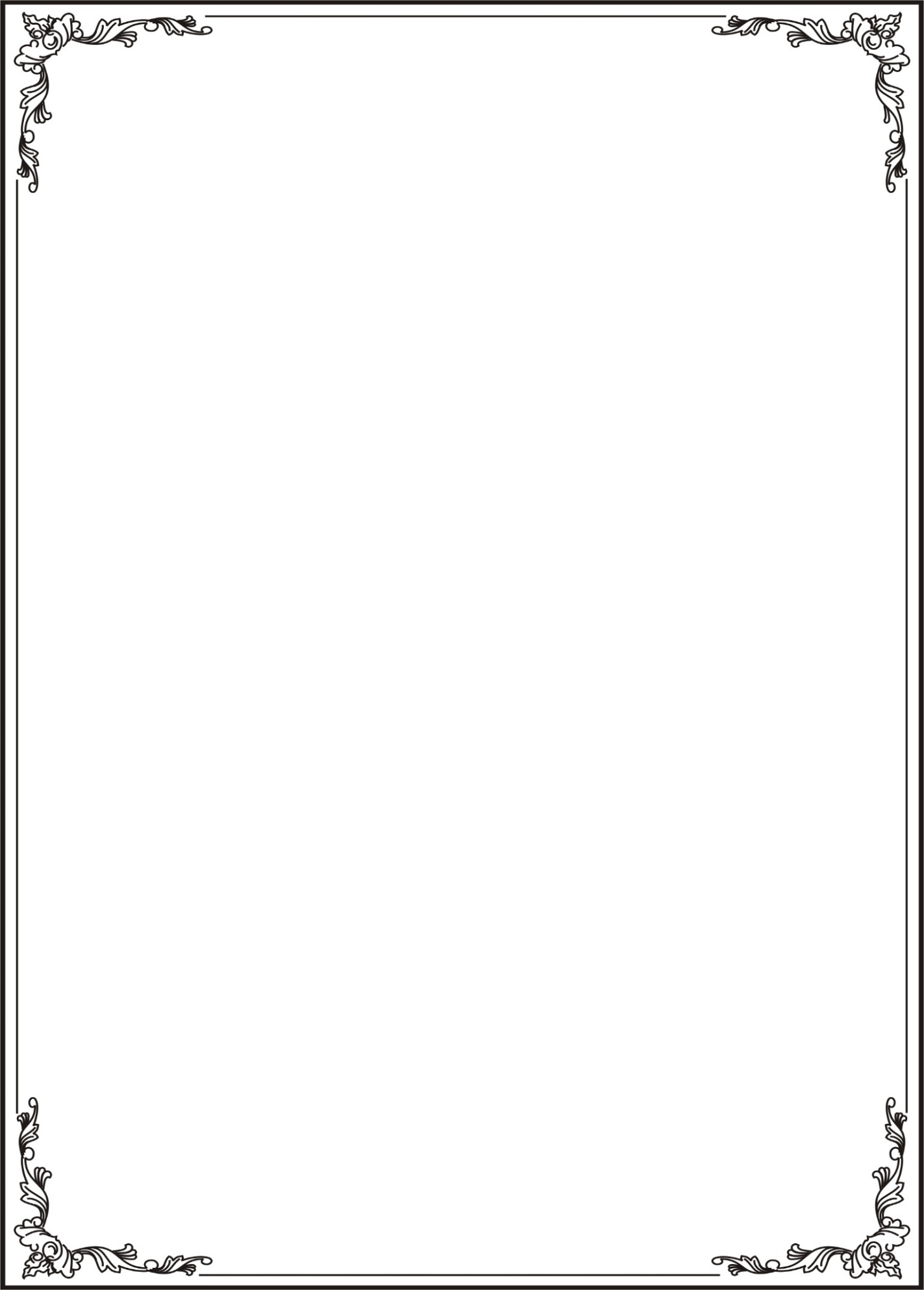
**BÁO CÁO THỰC NGHIỆM THUỘC HỌC PHẦN:**

**LẬP TRÌNH JAVA**

**XÂY DỰNG PHẦN MỀM QUẢN LÝ CƠ SỞ VẬT CHẤT TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

|  |  |  |
| --- | --- | --- |
| **GVHD** | **:** | ***TS. Hà Mạnh Đào*** |
| **Lớp**  **Khóa**  **Nhóm thực hiện** | **:**  **:**  **:** | **20241IT6019001**  **17**  **18** |

**Hà Nội – Năm 2024**



**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**======\*\*\*======**

****

**BÁO CÁO THỰC NGHIỆM THUỘC HỌC PHẦN:**

**LẬP TRÌNH JAVA**

**XÂY DỰNG PHẦN MỀM QUẢN LÝ CƠ SỞ VẬT CHẤT TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**Nhóm 18**

|  |  |  |  |
| --- | --- | --- | --- |
| **GVHD** | **:** | ***TS. Hà Mạnh Đào*** | |
| **Sinh viên** | **:** | **Nguyễn Thị Oanh**  **Bùi Văn Nghiêm**  **Trần Hồng Quân**  **Vũ Xuân Thương** | **- 2022604013 (Nhóm trưởng)**  **- 2022606281**  **- 2022605308**  **- 2022605678** |
| **Lớp - Khóa** | **:** | **20241IT6019001 - 17** | |

# LỜI CẢM ƠN

**Hà Nội – Năm 2024**

Trước tiên, chúng em xin gửi lời cảm ơn chân thành đến Ban giám hiệu Trường Đại học Công nghiệp Hà Nội, Khoa Công nghệ Thông tin, cùng toàn thể các thầy cô đã tận tình giảng dạy, truyền đạt kiến thức quý báu, tạo điều kiện để chúng em thực hiện báo cáo thực nghiệm này.

Chúng em xin đặc biệt cảm ơn thầy Hà Mạnh Đào đã hướng dẫn, hỗ trợ và góp ý nhiệt tình trong suốt quá trình nghiên cứu và xây dựng phần mềm quản lý cơ sở vật chất. Những kiến thức và kinh nghiệm mà thầy chia sẻ không chỉ giúp chúng em hoàn thiện báo cáo mà còn là hành trang quý báu trên con đường học tập và sự nghiệp sau này.

Chúng em cũng xin cảm ơn các anh/chị, bạn bè đã đồng hành, chia sẻ kinh nghiệm, cũng như hỗ trợ kỹ thuật trong quá trình thực hiện dự án. Đồng thời, ủng hộ về mặt tinh thần để chúng em có thể hoàn thành tốt nhiệm vụ học tập và nghiên cứu. Mặc dù đã nỗ lực hết mình, nhưng báo cáo thực nghiệm này khó tránh khỏi những thiếu sót. Chúng em rất mong nhận được sự góp ý từ thầy cô và các bạn để hoàn thiện hơn trong tương lai.

Chúng em xin trân trọng cảm ơn!

# MỤC LỤC

[LỜI CẢM ƠN 2](#_Toc185601910)

[DANH MỤC HÌNH ẢNH 5](#_Toc185601912)

[DANH MỤC BIỂU ĐỒ 6](#_Toc185601913)

[CHƯƠNG I: MỞ ĐẦU 7](#_Toc185601914)

[1.1. Ý tưởng nghiên cứu 7](#_Toc185601915)

[1.2. Những kiến thức và kỹ năng cần đạt được 7](#_Toc185601916)

[1.3. Yêu cầu về đầu ra sản phẩm nghiên cứu 7](#_Toc185601917)

[1.4. Ý tưởng về hoạt động của phần mềm 7](#_Toc185601918)

[1.5. Các công cụ được sử dụng để làm sản phẩm 8](#_Toc185601919)

[CHƯƠNG II: KẾT QUẢ NGHIÊN CỨU 9](#_Toc185601920)

[2.1. Giới thiệu 9](#_Toc185601921)

[2.2. Khảo sát hệ thống 10](#_Toc185601922)

[2.2.1. Khảo sát sơ bộ 10](#_Toc185601923)

[2.2.2. Tài liệu đặc tả yêu cầu 16](#_Toc185601924)

[2.3. Phân tích hệ thống 17](#_Toc185601925)

[2.3.1. Mô hình hóa chức năng hệ thống 17](#_Toc185601926)

[2.3.2. Mô hình hóa dữ liệu và cài đặt hệ thống 20](#_Toc185601927)

[2.4. Thực hiện bài toán 26](#_Toc185601928)

[2.4.1. Bùi Văn Nghiêm – Quản lý thông tin tài sản 26](#_Toc185601929)

[2.4.2. Nguyễn Thị Oanh – Quản lý mua sắm tài sản 38](#_Toc185601930)

[2.4.3. Trần Hồng Quân – Quản lý điều chuyển tài sản 44](#_Toc185601931)

[2.4.4. Vũ Xuân Thương – Quản lý sửa chữa tài sản 52](#_Toc185601932)

[CHƯƠNG III: KẾT LUẬN VÀ BÀI HỌC KINH NGHIỆM 59](#_Toc185601933)

[3.1. Nội dung đã thực hiện 59](#_Toc185601934)

[3.2. Hướng phát triển 60](#_Toc185601935)

[TÀI LIỆU THAM KHẢO 61](#_Toc185601936)

[PHỤ LỤC 62](#_Toc185601937)

# DANH MỤC HÌNH ẢNH

[Hình 2. 1: Giao diện Quản lý thông tin tài sản - Thao tác nhập tài sản mới – Bùi Văn Nghiêm 28](#_Toc185520581)

[Hình 2. 2: Giao diện Quản lý thông tin tài sản - Sau khi thêm tài sản mới – Bùi Văn Nghiêm 28](#_Toc185520582)

[Hình 2. 3: Giao diện Quản lý mua sắm tài sản – Nguyễn Thị Oanh 39](#_Toc185520583)

[Hình 2. 4: Giao diện Quản lý mua sắm tài sản - Thao tác in danh sách mua sắm – Nguyễn Thị Oanh. 40](#_Toc185520584)

[Hình 2. 5. Giao diện Quản lý điều chuyển tài sản – Trần Hồng Quân 45](#_Toc185520585)

[Hình 2. 6. Giao diện Quản lý điều chuyển tài sản - Thao tác in chi tiết điều chuyển - Trần Hồng Quân 46](#_Toc185520586)

[Hình 2. 7. Giao diện Quản lý sửa chữa tài sản – Vũ Xuân Thương 53](#_Toc185520587)

[Hình 2. 8. Giao diện Quản lý sửa chữa tài sản - Thao tác xét duyệt sửa chữa - Vũ Xuân Thương 54](#_Toc185520588)

# DANH MỤC BIỂU ĐỒ

[Biểu đồ 2. 1: Biểu đồ Use Case tổng quát – Trần Hồng Quân 20](#_Toc185520619)

[Biểu đồ 2. 2: Biểu đồ lớp tham gia use case Quản lý thông tin tài sản – Bùi Văn Nghiêm 22](#_Toc185520620)

[Biểu đồ 2. 3: Biểu đồ lớp tham gia use case Quản lý mua sắm tài sản – Nguyễn Thị Oanh 23](#_Toc185520621)

[Biểu đồ 2. 4: Biểu đồ lớp tham gia use case Quản lý điều chuyển tài sản – Trần Hồng Quân 24](#_Toc185520622)

[Biểu đồ 2. 5: Biều đồ lớp tham gia use case Quản lý sứa chữa tài sản – Vũ Xuân Thương 26](#_Toc185520623)

# CHƯƠNG I: MỞ ĐẦU

## Ý tưởng nghiên cứu

Đối với các cơ sở, tổ chức giáo dục cấp Đại học, việc ứng dụng phần mềm vào việc xử lý, giải quyết các bài toán quản lý khối lượng công việc lớn là vô cùng cần thiết. Nhận biết được hiện trạng ở phần lớn các trường Đại học, việc quản lý cơ sở vật chất vẫn còn đang thực hiện bằng phương pháp thủ công, xử lý các nghiệp vụ qua sổ sách còn nhiều bất tiện, rời rạc, tốn nhiều thời gian, công sức. Vì vậy, cần có một phần mềm giúp giải quyết, khắc phục những bất lợi trên. Chính bởi vậy, nhóm chúng em đã quyết định chọn đề tài “***Xây dựng phần mềm quản lý cơ sở vật chất Trường Đại học Công Nghiệp Hà Nội***” với mong muốn sẽ mang các kiến thức được học vận dụng và xây dựng lên phần mềm quản lý cơ sở vật chất, giúp thầy cô và cán bộ nhà trường thực hiện hiệu quả hơn trong công việc, tiết kiệm thời gian, công sức quản lý cơ sở vật chất.

## Những kiến thức và kỹ năng cần đạt được

* Kiến thức cơ bản về ngôn ngữ lập trình Java, Java Swing.
* Kiến thức cơ bản về thao tác với cơ sở dữ liệu với file.
* Kiến thức, kỹ năng đặc tả thiết kế phần mềm.
* Kỹ năng làm việc nhóm.

## Yêu cầu về đầu ra sản phẩm nghiên cứu

* Tiết kiệm thời gian, công sức về xử lý giấy tờ.
* Lưu trữ thông tin an toàn.
* Dễ dàng quản lý, cập nhật thông tin.
* Thuận tiện cho các nghiệp vụ đặc thù của công việc quản lý cơ sở vật chất.

## Ý tưởng về hoạt động của phần mềm

Phần mềm được viết ra với mục đích giúp những người tham gia vào quy trình quản lý cơ sở vật chất thực hiện các công việc nghiệp vụ liên quan (mua sắm, điều chuyển, sửa chữa,…) một cách hiệu quả và dễ dàng hơn. Phần mềm cho phép quản lý chi tiết các thông tin của tài sản đến quy trình các công việc lập các giấy tờ, biểu mẫu cần thiết để thực hiện nghiệp vụ.

## Các công cụ được sử dụng để làm sản phẩm

* Rational Rose
* Netbean 24
* Java 23

# CHƯƠNG II: KẾT QUẢ NGHIÊN CỨU

## Giới thiệu

Một phần mềm có thể đưa đến tay người sử dụng cần phải trải qua các bước:

* Phát hiện yêu cầu: là quy trình thu thập những thông tin về hệ thống được đề xuất và hệ thống đang tồn tại để xác định yêu cầu hệ thống và yêu cầu của người sử dụng.
* Khảo sát sơ bộ: xác định nhóm người dùng, viết báo cáo tổng hợp.
* Khảo sát chi tiết: tìm hiểu về hoạt động, các yêu cầu chức năng của hệ thống
* Phương pháp xây dựng phần mềm: quá trình tạo ra chi tiết của phần mềm thông qua sự kết hợp của viết mã, xác minh, kiểm thử đơn vị, kiểm thử tích hợp, gỡ lỗi.
* Mô hình hóa dữ liệu: Biểu đồ thực thể liên kết mức logic (mô tả yêu cầu dữ liệu, biểu đồ thực thể liên kết, quy tắc nghiệp vụ), mô hình hóa dữ liệu mức vật lý.
* Mô hình hóa chức năng: biểu đồ use case, mô tả chi tiết các use case.
* Thiết kế cơ sở dữ liệu: phân tích, thiết kế logic, thiết kế vật lý.

Phần mềm quản lý cơ sở vật chất trường Đại học Công Nghiệp Hà Nội được xây dựng bằng ngôn ngữ Java và sử dụng công cụ NetBeans.

NetBeans IDE là một công cụ hỗ trợ lập trình viết mã code miễn phí được cho là tốt nhất hiện nay, được sử dụng chủ yếu cho các lập trình viên phát triển Java tuy nhiên phần mềm có dung lượng khá là nặng dành cho các máy cấu hình có RAM, CPU tương đối cao để vận hành. NetBeans IDE là môi trường phát triển tích hợp và cực kỳ cần thiết cho các lập trình viên Java, công cụ này có thể hoạt động tốt với rất nhiều nền tảng hệ điều hành khác nhau như Linux, Windows, MacOS, ... là một mã nguồn mở cung cấp các tính năng cần thiết nhất nhằm tạo ra các ứng dụng web, thiết bị di động, desktop.

## Khảo sát hệ thống

### 2.2.1. Khảo sát sơ bộ

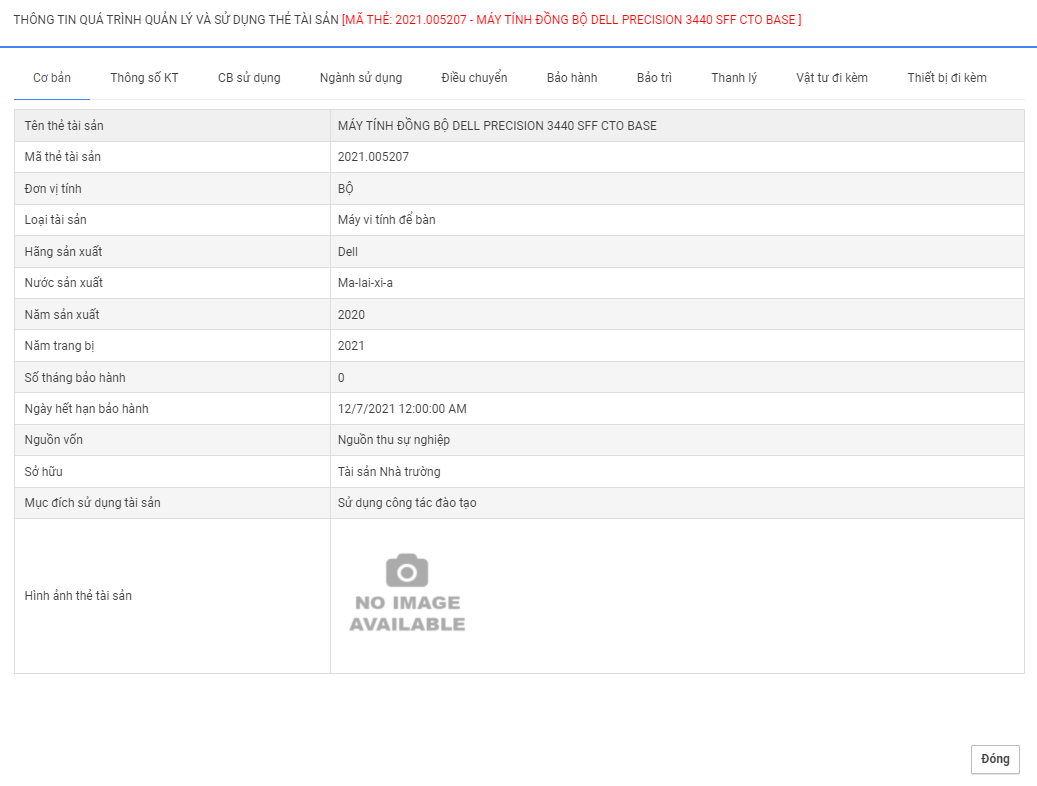
* Cấu trúc tổ chức của Trường Đại học Công Nghiệp hà Nội

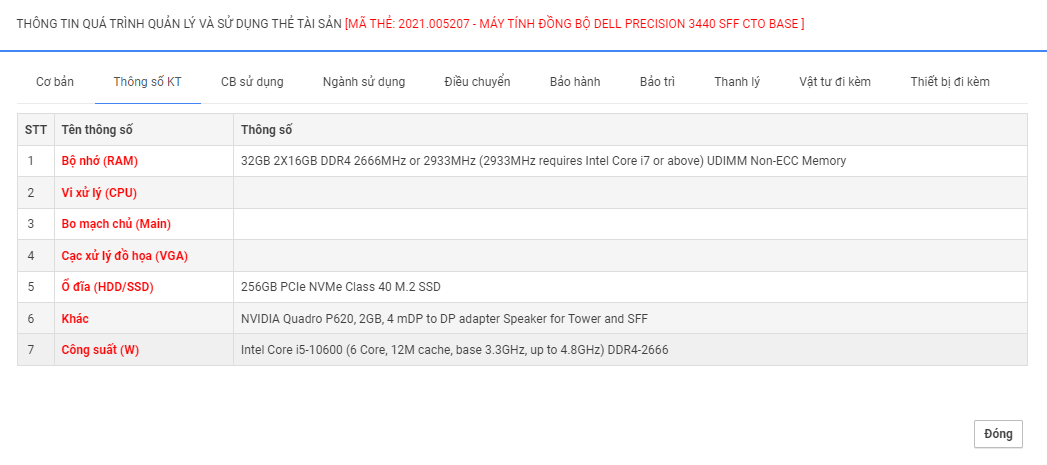
Trường Đại học Công nghiệp Hà Nội (tiếng Anh: Hanoi University of Industry, viết tắt: HaUI) là một trường đại học công lập, đa cấp, đa ngành nghề, định hướng ứng dụng và thực hành trực thuộc Bộ Công thương. Trường được nâng cấp lên đại học năm 2005 trên cơ sở Trường Cao đẳng Công nghiệp Hà Nội. Tiền thân của trường là hai trường do thực dân Pháp thành lập là Trường Chuyên nghiệp Hà Nội năm 1898, và Trường Chuyên nghiệp Hải Phòng năm 1913.

Trường hiện có hơn 1800 cán bộ, giáo viên, nhân viên, trong đó có 1451 giảng viên cơ hữu và hợp đồng dài hạn, trong đó trên 80% trình độ trên Đại học.

Hiện nay, Nhà trường có 2 cơ sở đào tạo ở Hà Nội và 1 cơ sở đào tạo ở Hà Nam với tổng diện tích gần 50 ha.

* Các vấn đề cần giải quyết từ bài toán
* Quản lý thông tin về tài sản.
* Quản lý quá trình về mua sắm tài sản.
* Quản lý thông tin điều chuyển tài sản.
* Quản lý sửa chữa tài sản.Các biểu mẫu về quản lý cơ sở vật chất thu thập được
* Biểu mẫu lưu trữ thông tin của tài sản

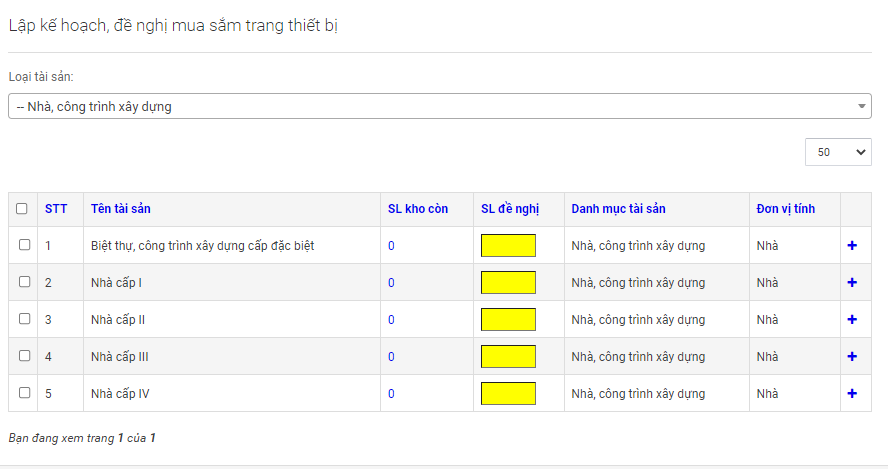




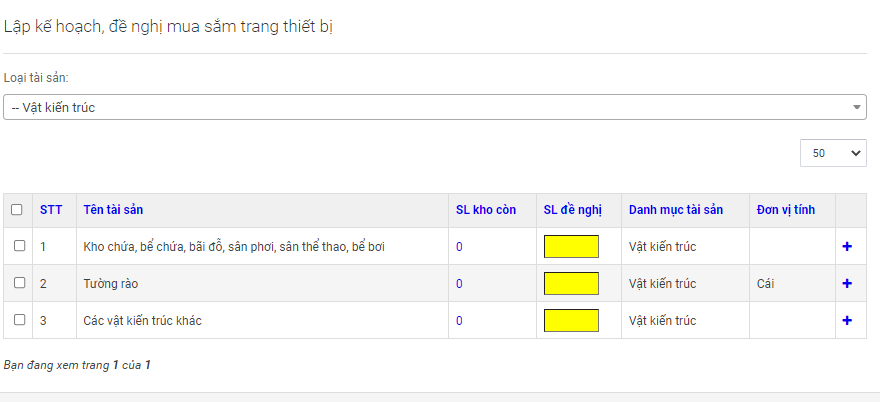


* Biểu mẫu thông tin danh mục tài sản

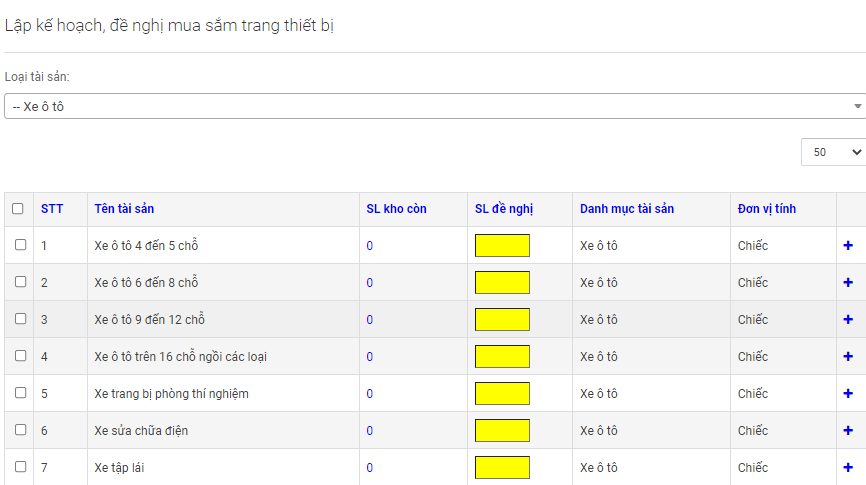
+ Nhà, công trình kiến trúc:



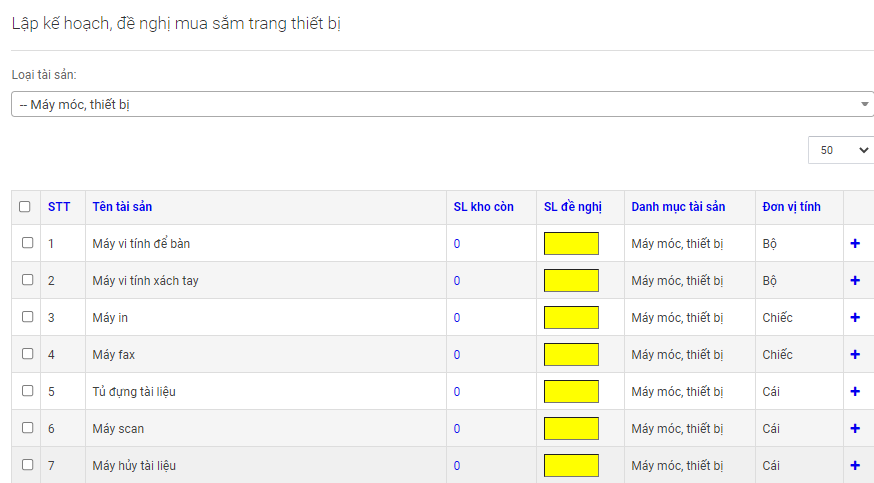
+ Vật kiến trúc:



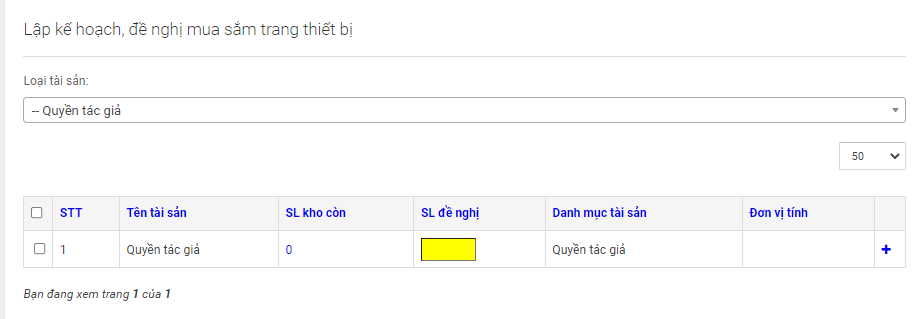
+ Xe ô tô:

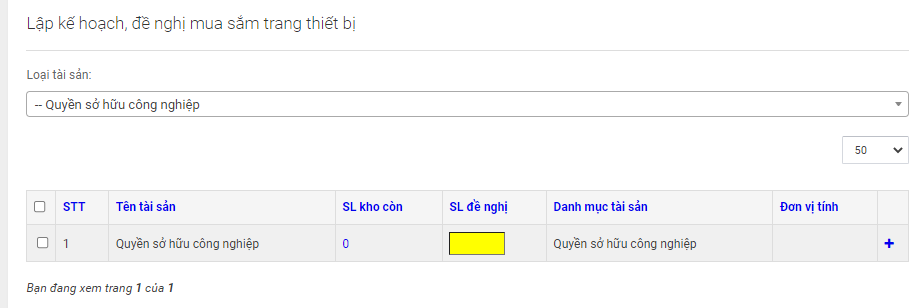


+ Máy móc, thiết bị:



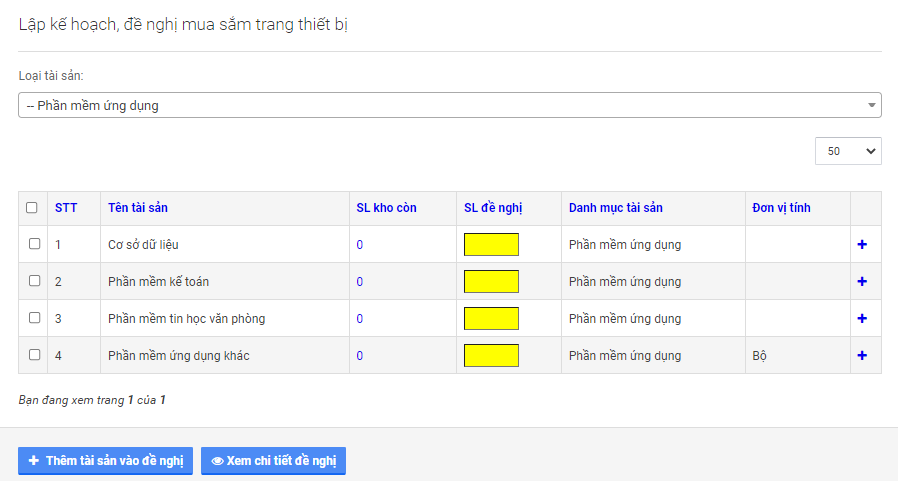
+ Các tài sản là bản quyền:



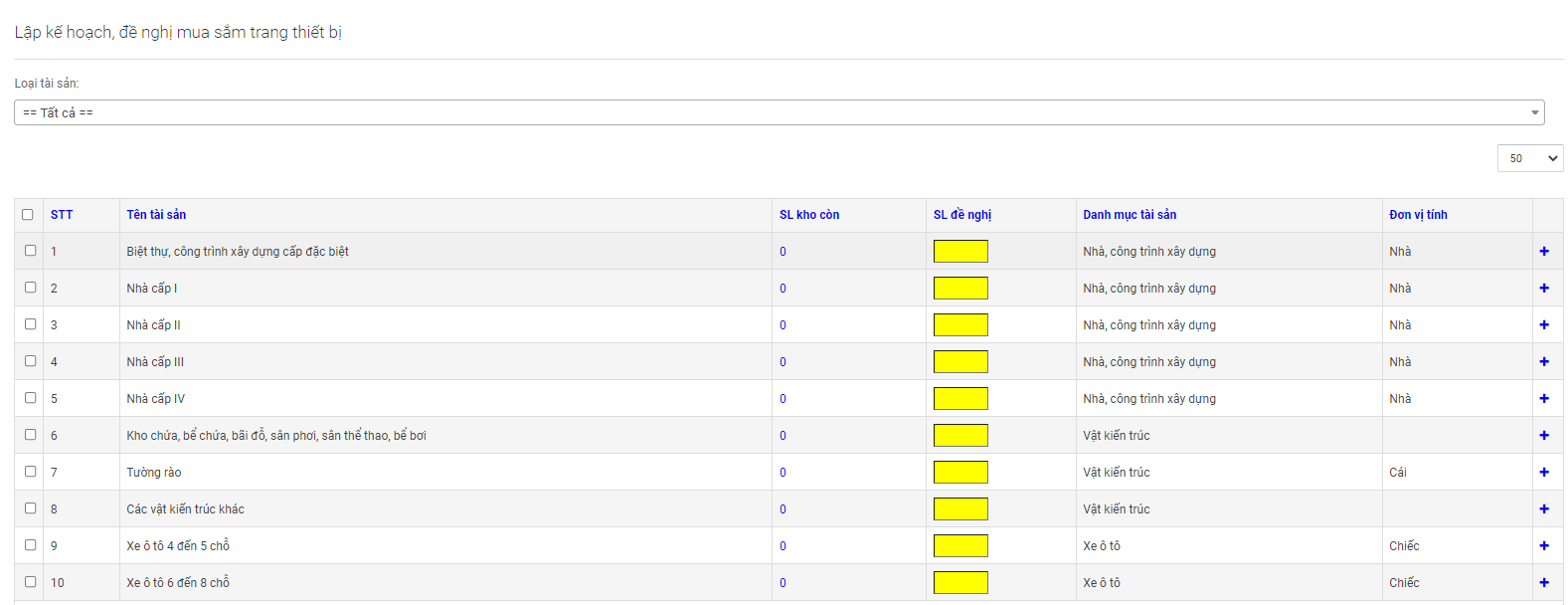




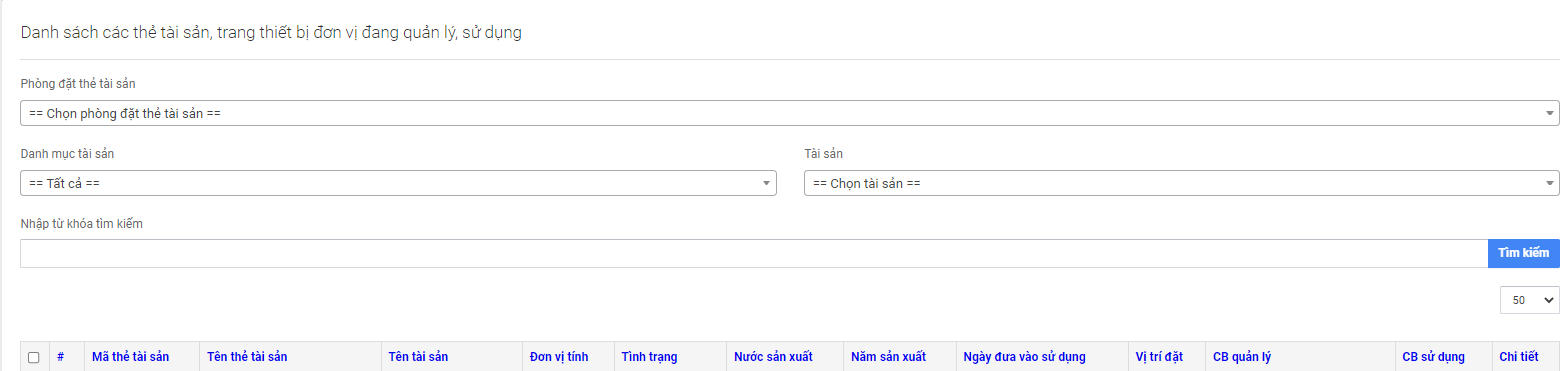
+ Phần mềm, ứng dụng:



* Biểu mẫu mua sắm tài sản



* Biểu mẫu điều chuyển tài sản



### 2.2.2. Tài liệu đặc tả yêu cầu

1. *Hoạt động của hệ thống*

Tổng quan về quy trình quản lý cơ sở vật chất tại trường:

* Nhà trường lập kế hoạch mua sắm trang thiết bị cơ sở vật chất theo năm/tháng/quý. Các khoa hay các đơn vị đặt mua thiết bị theo năm/quý/tháng theo yêu cầu dựa trên danh mục trang thiết bị, văn phòng phẩm,… Trường duyệt kế hoạch mua sắm phù hợp với tình hình tài chính. Tiến hành đấu thầu và chọn nhà thầu cung cấp, mua sắm và phân bổ theo kế hoạch thực tế khi cần xử lý các tình huống mua sắm gấp, khoa lập đề nghị gửi phòng quản trị, quản trị trình danh sách cho hiệu trưởng phê duyệt.
* Các khoa, đơn vị khi có nhu cầu hoặc được yêu cầu mua sắm tài sản và được xét duyệt sẽ mua tài sản từ nhà cung cấp được lựa chọn. Tài sản mua sẽ được kiểm tra và bàn giao thông qua phiếu giao nhận tài sản cùng hợp đồng mua. Phiếu giao nhận cùng hợp đồng mua tài sản sẽ được lưu trữ và cập nhật thông tin về tài sản mới vào hệ thống quản lý tài sản.
* Khi có nhu cầu điều chuyển tài sản thì khoa, đơn vị sẽ tổng hợp lập phiếu đưa đi xét duyệt và thực hiện bàn giao thông qua phiếu điều chuyển tài sản. Các khoa, đơn vị khi nhận tài sản được điều chuyển đến sẽ cần đánh giá lại, xác định giá trị hiện tại của tài sản đó và đưa tài sản vào sử dụng.
* Trong quá trình sử dụng nếu tài sản cần sửa chữa, thay thế thì cán bộ bảo trì, phụ trách quản lý sẽ lập phiếu sửa chữa đưa đi xét duyệt và thực hiện công tác sửa chữa.

1. *Các yêu cầu chức năng từ hoạt động*

* Quản lý thông tin tài sản:

+ Thêm thông tin tài sản mới.

+ Bảo trì thông tin tài sản.

+ Tìm kiếm tài sản.

* Quản lý mua sắm tài sản:

+ Thêm tài sản mua mới.

+ Bảo trì thông tin phiếu mua tài sản.

+ Xuất phiếu mua tài sản.

* Quản lý điều chuyển tài sản:

+ Thêm các đề nghị điều chuyển tài sản.

+ Bảo trì thông tin đề nghị điều chuyển.

+ Xuất chi tiết thông tin điều chuyển.

* Quản lý sửa chữa tài sản:

+ Tạo các đề nghị sửa chữa tài sản.

+ Bảo trì thông tin sửa chữa tài sản.

+ Phê duyệt đề nghị sửa chữa tài sản.

+ Xuất phiếu đề nghị sửa chữa tài sản.

1. *Yêu cầu phi chức năng*

* Hệ thống có khả năng bảo mật.
* Chương trình gọn nhẹ, dễ dàng sửa dụng.

## Phân tích hệ thống

### 2.3.1. Mô hình hóa chức năng hệ thống

1. *Các tác nhân (Actor) và nhiệm vụ chính*

* Quản trị viên hệ thống (Admin): Actor này có đầy đủ các chức năng, quyền hạn đổi với hệ thống như phân quyền người dùng, cập nhật, bảo trì các hạng mục trong hệ thống.
* Cán bộ quản lý (Manager-User): Actor này có thể sử dụng các chức năng xét duyệt, đánh giá các đề nghị được đưa ra.
* Cán bộ nhân viên (User): Actor này có các chức năng cơ bản của người dùng như thêm mới và bảo trì các thông tin về quy trình hoạt động.

1. *Các Use case và nhiệm vụ*

* Đăng ký:

+ Đăng ký tài khoản

* Đăng nhập:

+ Đăng nhập vào hệ thống

* Quản lý thông tin tài sản:

+ Xem danh sách tài sản

+ Bảo trì thông tin tài sản

* Quản lý mua sắm tài sản:

+ Xem danh sách mua sắm tài sản

+ Bảo trì thông tin mua sắm

+ Xuất phiếu đề nghị mua sắm

* Quản lý điều chuyển tài sản:

+ Xem danh sách điều chuyển tài sản

+ Bảo trì thông tin điều chuyển

+ Xuất phiếu chi tiết điều chuyển tài sản

* Quản lý sửa chữa tài sản

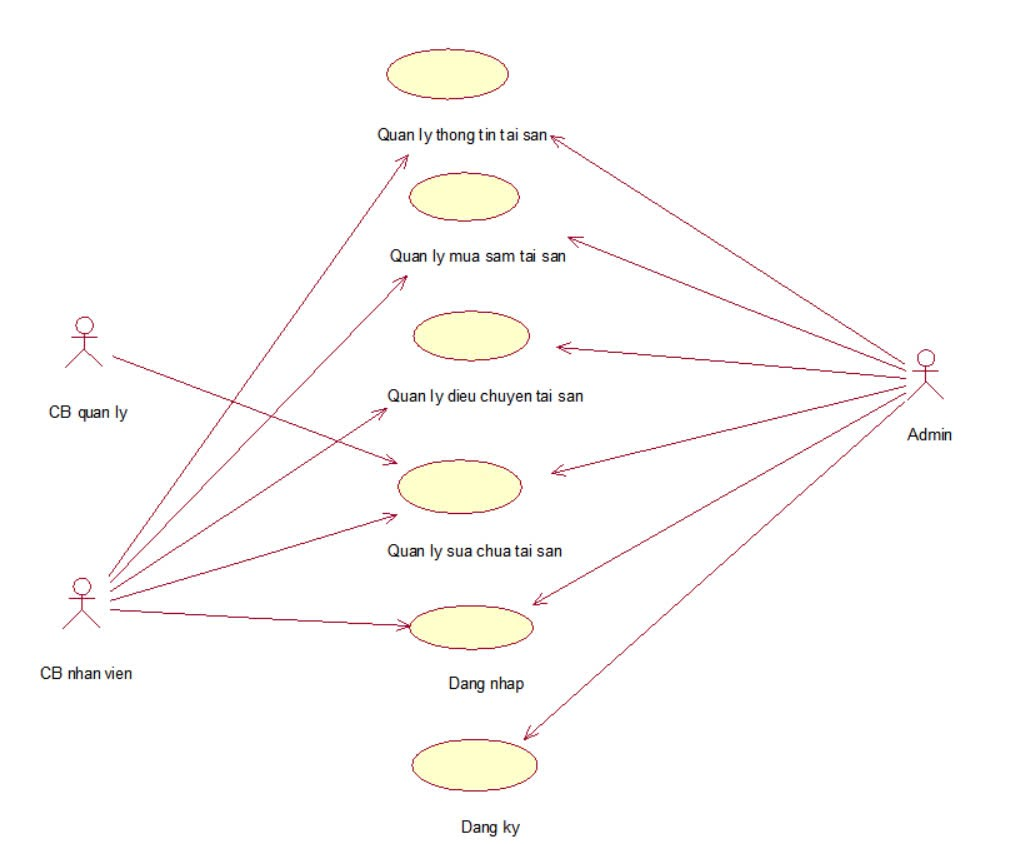
+ Xem danh sách sửa chữa tài sản

+ Bảo trì thông tin sửa chữa tài sản

+ Xét duyệt các đề nghị sửa chữa

+ Xuất danh sách đề nghị sửa chữa

1. *Biểu đồ Use Case*

**

Biểu đồ 2. 1: Biểu đồ Use Case tổng quát – Trần Hồng Quân

1. *Đặc tả các use case*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Tên use case** | **Mô tả vắn tắt** | **Chức năng** | **Ghi chú** |
| UC\_001 | Quản lý thông tin tài sản | Use case này cho phép người quản trị xem, thêm, sửa, xóa các thông tin về tài sản. | Xem thông tin tài sản.  Bảo trì thông tin tài sản. | Quản trị viên  Cán bộ nhân viên |
| UC\_002 | Quản lý mua sắm tài sản | Usecase này cho phép người dùng quản lý thông tin mua sắm tài sản của hệ thống. | Xem thông tin mua tài sản.  Bảo trì thông tin mua tài sản.  In phiếu mua tài sản. | Quản trị viên  Cán bộ nhân viên |
| UC\_003 | Quản lý điều chuyển tài sản | Use case này cho phép người dùng xem, thêm, sửa, xóa các thông tin về điều chuyển và xuất ra phiếu điều chuyển tài sản. | Xem thông tin điều chuyển.  Bảo trì thông tin điều chuyển.  Xuất phiếu điều chuyển chi tiết. | Quản trị viên  Cán bộ nhân viên |
| UC\_004 | Quản lý sửa chữa tài sản | Use case này cho phép người dùng quản lý thông tin sửa chữa tài sản của hệ thống. | Xem thông tin sửa chữa.  Bảo trì thông tin sửa chữa.  Xét duyệt thông tin sửa chữa.  Xuất phiếu đề nghị được duyệt. | Quản trị viên  Cán bộ quản lý  Cán bộ nhân viên |

### 2.3.2. Mô hình hóa dữ liệu và cài đặt hệ thống

* **Biểu đồ lớp tham gia ca sử dụng**
* **Use case “Quản lý thông tin tài sản”:**



Biểu đồ 2. 2: Biểu đồ lớp tham gia use case Quản lý thông tin tài sản – Bùi Văn Nghiêm

Use case này bắt đầu khi người dùng kích vào nút “Quản lý thông tin tài sản” trên menu quản trị. Hệ thống lấy thông tin chi tiết của các tài sản từ file chứa danh sách tài sản và hiển thị danh sách các tài sản cùng các điều khiển nhập thông tin lên màn hình.

1. Thêm tài sản:

Người dùng nhập thông tin cho tài sản vào các điều khiển và kích vào nút “Thêm”. Hệ thống sẽ tạo một tài sản mới trong danh sách, lưu lại file và hiển thị danh sách các tài sản sau khi thêm.

1. Sửa tài sản:

Người dùng kích chọn một dòng tài sản. Hệ thống sẽ lấy thông tin cũ của tài sản được chọn từ danh sách tài sản và hiển thị lên màn hình.

Người dùng nhập thông tin mới cho tài sản và kích vào nút “Sửa”. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận sửa.

Người dùng kích vào nút “Đồng ý”. Hệ thống sẽ sửa thông tin của tài sản được chọn trong danh sách tài sản, lưu lại file và hiển thị danh sách tài sản đã cập nhật.

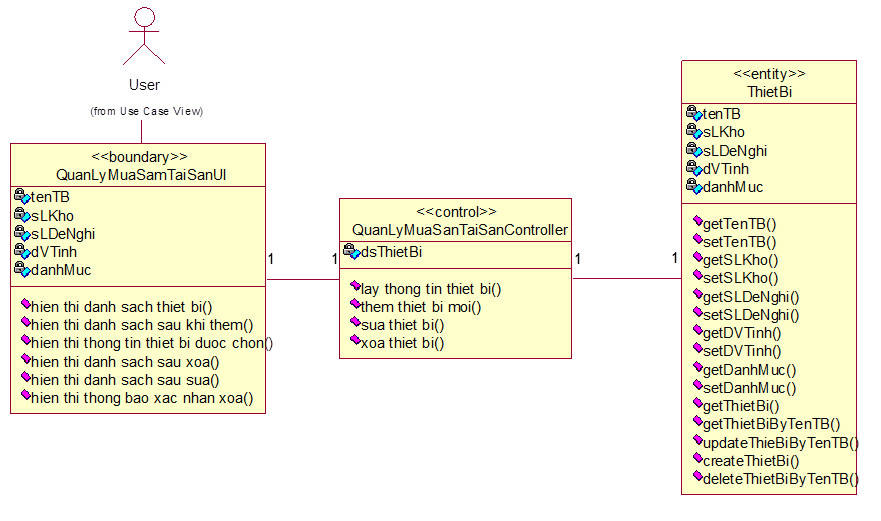
1. Xóa tài sản:

Người dùng chọn một dòng tài sản và kích vào nút “Xóa”. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa.

Người quản trị kích vào nút “Đồng ý”. Hệ thống sẽ xóa tài sản được chọn khỏi danh sách tài sản, lưu file và hiển thị danh sách các tài sản đã cập nhật.

Use case kết thúc.

* **Use case “Quản lý mua sắm tài sản”:**



Biểu đồ 2. 3: Biểu đồ lớp tham gia use case Quản lý mua sắm tài sản – Nguyễn Thị Oanh

Usecase này bắt đầu khi người dùng chọn “Quản lý mua sắm tài sản” trên mà hình chức năng chính của hệ thống. Hệ thống lấy tất cả thông tin của danh sách mua sắm thiết bị từ file và hiển thị lên màn hình.

a) Thêm thiết bị:

Người dùng nhập thông tin của một thiết bị vào form hiển thị trên màn hình.

Người dùng kích nút “Thêm” trên màn hình. Hệ thống thêm thông tin thiết bị vào danh sách mua sắm, lưu lại file và hiển thị lên màn hình.

b) Sửa thiết bị:

Người dùng chọn thiết bị cần sửa trên bảng thông tin thiết bị. Hệ thống hiển thị thông tin cũ của thiết bị vào form. Sau đó nhập thông tin cần sửa vào form.

Người dùng kích nút “Sửa” trên màn hình. Hệ thống cập nhập thông tin thiết bị vào danh sách thiết bị, lưu lại file và hiển thị lên màn hình.

c) Xoá thiết bị:

Người dùng chọn thiết bị cần xoá trên bảng thông tin thiết bị. Hệ thống hiển thị thông tin cũ của thiết bị vào form.

Người dùng kích nút “Xoá” trên màn hình. Hệ thống xóa thông tin thiết bị trong danh sách thiết bị, lưu file và hiển thị lên màn hình. Usecase kết thúc.

d) In phiếu mua sắm thiết bị:

Người dùng kích nút “In phiếu” trên màn hình. Hệ thống lấy thông tin các thiết bị của danh sách thiết bị thêm vào phiếu in và hiển thị lên mà hình.

* **Use case “Quản lý điều chuyển tài sản”:**



Biểu đồ 2. 4: Biểu đồ lớp tham gia use case Quản lý điều chuyển tài sản – Trần Hồng Quân

Use case này bắt đầu khi người dùng kích vào nút “Quản lý điều chuyển tài sản” trên menu quản trị. Hệ thống lấy thông tin chi tiết của các tài sản điều chuyển từ file tài sản điều chuyển và hiển thị danh sách các tài sản điều chuyển cùng các điều khiển nhập thông tin lên màn hình, hệ thống cũng lấy thông tin về tài sản từ file danh sách tài sản để hiện lên màn hình cho người dùng chọn tài sản thực hiện điều chuyển.

1. Thêm tài sản điều chuyển:

Người dùng chọn một tài sản từ danh sách chọn tài sản, nhập thông tin cho việc điều chuyển vào các điều khiển và kích vào nút “Thêm”. Hệ thống sẽ tạo một tài sản điều chuyển mới trong trong danh sách, lưu lại file và hiển thị danh sách các tài sản điều chuyển sau khi thêm.

1. Sửa tài sản điều chuyển:

Người dùng kích chọn một dòng tài sản điều chuyển. Hệ thống sẽ lấy thông tin cũ của tài sản điều chuyển được từ danh sách tài sản điều chuyển và hiển thị lên màn hình.

Người dùng nhập thông tin mới cho tài sản điều chuyển và kích vào nút “Sửa”. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận sửa.

Người dùng kích vào nút “Đồng ý”. Hệ thống sẽ sửa thông tin của tài sản điều chuyển được chọn trong danh sách, lưu lại file và hiển thị danh sách tài sản điều chuyển đã cập nhật.

1. Xóa tài sản điều chuyển:

Người dùng chọn một dòng tài sản điều chuyển và kích vào nút “Xóa”. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa.

Người quản trị kích vào nút “Đồng ý”. Hệ thống sẽ xóa tài sản điều chuyển được chọn khỏi danh sách, lưu lại file và hiển thị danh sách các tài sản điều chuyển đã cập nhật.

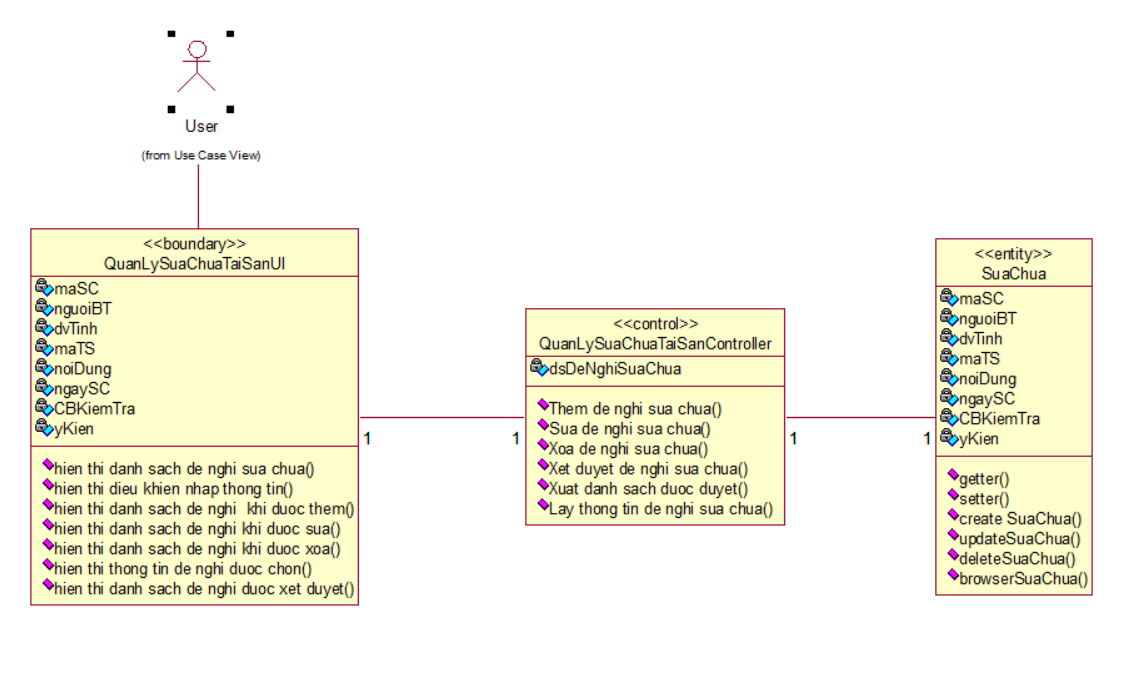
1. Xuất phiếu điều chuyển tài sản:

Người dùng chọn một dòng tài sản điều chuyển. Hệ thống sẽ hiển thị nút “Xem chi tiết”.

Người dùng nhấn chọn nút “Xem chi tiết”. Hệ thống sẽ lấy thông tin của tài sản được điều chuyển từ danh sách thông tin tài sản và thông tin của việc điều chuyển từ danh sách điều chuyển đưa vào một mẫu phiếu tạo sẵn và hiển thị lên màn hình.

Use case kết thúc.

* **Use case “Quản lý sửa chữa tài sản”:**



Biểu đồ 2. 5: Biều đồ lớp tham gia use case Quản lý sứa chữa tài sản – Vũ Xuân Thương

Usecase này bắt đầu khi người dùng chọn “Quản lý sửa chữa tài sản tài sản” trên mà hình chức năng chính của hệ thống. Hệ thống lấy tất cả thông tin từ file danh sách sửa chữa và hiển thị lên màn hình.

a) Thêm đề nghị sửa chữa

Người dùng nhập cần nhập đầy thủ thông tin của tài sản cần sửa chữa vào form hiển thị trên màn hình.

Người dùng kích nút “Thêm” trên màn hình. Hệ thống thêm thông tin thiết bị vào danh sách sửa chữa, lưu file và hiển thị lên màn hình.

b) Sửa đề nghị sửa chữa

Người dùng bấm chọn một dòng đề nghị bất kì muốn sửa đổi. Hệ thống hiển thị thông tin cũ của đề nghị vào form. Sau đó thay thông tin cần sửa vào form.

Người dùng kích nút “Sửa” trên màn hình. Hệ thống sẽ hiện lên một bảng thông báo “Bạn có muốn sửa không”. Chọn “YES”. Hệ thống cập nhập thông tin đề nghị mới vào danh sách sửa chữa, lưu file và hiển thị lên màn hình.

c) Xoá đề nghị sửa chữa

Người dùng bấm chọn một dòng đề nghị bất kì muốn xóa. Sau đó người dùng kích nút “Xóa” trên màn hình. Hệ thống sẽ hiện lên một bảng thông báo “Bạn có muốn xóa không”. Nếu muốn xóa người dùng chọn “YES”. Hệ thống sẽ xóa đề nghị sửa chữa trong danh sách, lưu file và hiển thị lên màn hình.

d) Xét duyệt đề nghị sửa chữa tài sản

Sau khi thêm các đề nghị sửa chữa tài sản. Người dùng có quyền duyệt sẽ chọn một dòng đề nghị bất kì sau đó bấm chọn trạng thái, bấm xác nhận và thêm ý kiến nếu có.Hệ thống sẽ cập nhật trạng thái trong bảng SUACHUA và hiển thị ra màn hình.

e) Xuất danh sách

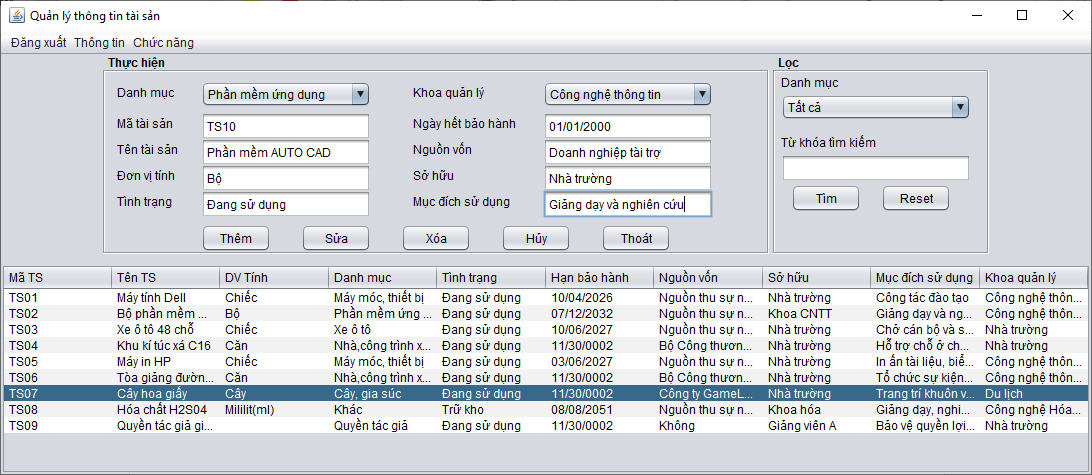
Người dùng kích nút “Xuất DS được duyệt” trên màn hình. Hệ thống lấy thông tin các đề nghị được duyệt của bảng SUACHUA thêm vào phiếu in và hiển thị lên màn hình.

Use Case kết thúc.

## Thực hiện bài toán

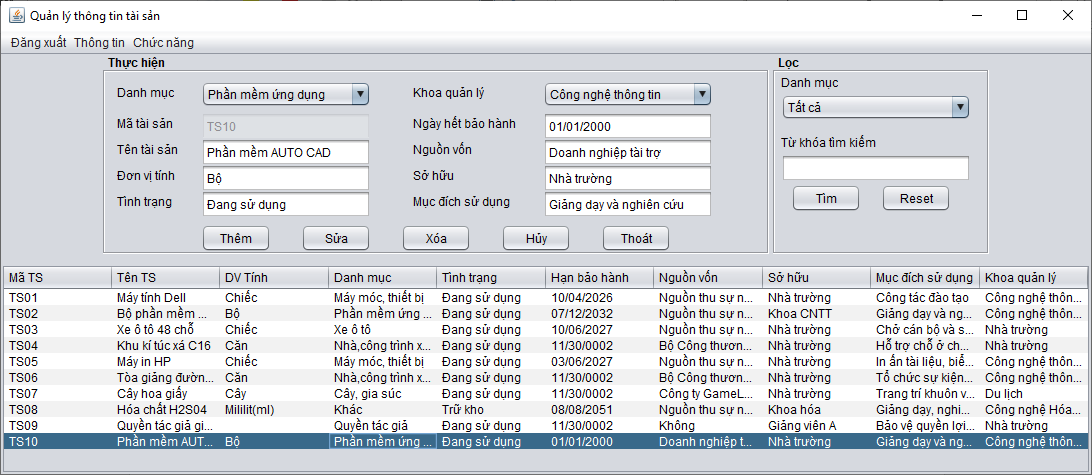
### 2.4.1. Bùi Văn Nghiêm – Quản lý thông tin tài sản

* Giao diện chức năng “Quản lý thông tin tài sản”:



Hình 2. 1: Giao diện Quản lý thông tin tài sản - Thao tác nhập tài sản mới – Bùi Văn Nghiêm

Thực hiện nhập các thông tin của tài sản từ các điều khiển trong khung “thực hiện” và nhấn nút “Thêm” để đưa tài sản mới vào danh sách.



Hình 2. 2: Giao diện Quản lý thông tin tài sản - Sau khi thêm tài sản mới – Bùi Văn Nghiêm

* Mô tả mã lệnh thực hiện chức năng “Quản lý tài sản”:
* Hướng đối tượng:

- Áp dụng được tính chất trừu tượng và đóng gói để đưa được đối tượng tài sản về lớp TaiSan lưu trữ các thông tin như sau:

public class TaiSan implements Serializable {

private String maTS;

private String tenTS;

private String dvTinh;

private String tinhTrang;

private Date ngayBH;

private String nguonVon;

private String soHuu;

private String mucDich;

private String danhMuc;

private String khoaQl;

…

}

- Áp dụng tính chất kế thừa và đa hình để tạo lên lớp TableTaiSan biểu diễn thông tin tài sản theo dạng bảng:

//Lớp TableTaiSan kế thừ lớp AbstractTableModel

public class TableTaiSan extends AbstractTableModel {

//Mảng chuỗi Name để tạo tiêu đề

private String Name[];

//Mảng class để hiển thị thông tin từ lấy từ lớp tài sản

private Class classess[] ;

//Thuộc tính dsTS để lưu thông tin về các tài sản lấy từ file dữ liệu

ArrayList<TaiSan> dsTS = new ArrayList<>();

…

//Hàm getValueAt thể hiện tính đa hình định nghĩa lại cách lấy thông tin từ lớp tài sản

@Override

public Object getValueAt(int rowIndex, int columnIndex) {

switch(columnIndex){

case 0: return dsTS.get(rowIndex).getMaTS();

case 1: return dsTS.get(rowIndex).getTenTS();

case 2: return dsTS.get(rowIndex).getDvTinh();

case 3: return dsTS.get(rowIndex).getDanhMuc();

case 4: return dsTS.get(rowIndex).getTinhTrang();

case 5: return dsTS.get(rowIndex).getNgayBH();

case 6: return dsTS.get(rowIndex).getNguonVon();

case 7: return dsTS.get(rowIndex).getSoHuu();

case 8: return dsTS.get(rowIndex).getMucDich();

case 9: return dsTS.get(rowIndex).getKhoaQl();

default : return null;

}

}

…

}

* Bắt lỗi và gom rác: sử dụng cơ chế try-catch và if-else.

- Cơ chế try-catch để bắt những ngoại lệ (exception) xảy ra do vi phạm nguyên tắc của ngôn ngữ lập trình:

//Hàm saveData có thể sẽ gặp lỗi khi lưu dữ liệu vào file lúc này sẽ bắt ngoại lệ được hệ thống ngôn ngữ báo ra.

public void saveData(){

try {

db.LuuFile(fname,dsTaiSan);

} catch (Exception ex) {

JOptionPane.showMessageDialog(null, "Lỗi lưu dữ liệu:\n" +ex,"Lỗi", JOptionPane.ERROR\_MESSAGE);

}

}

- Cơ chế if-else sử dụng để bắt ra những lỗi có thể đoán được từ thao tác của người dùng:

//Có khả năng người dùng sẽ để trống trường thông tin quan trọng là mã tài sản, lúc này sử dụng if-else để báo lại cho người dùng

private void jButtonThemActionPerformed(java.awt.event.ActionEvent evt) {

if(jMaTS.getText().trim().length()==0)

JOptionPane.showMessageDialog(null, "Mã tài sản không được để trống!", "Thông báo", JOptionPane.WARNING\_MESSAGE);

else {…}

}

* Tập hợp sử dụng: Sử dụng ArrayList để lưu trữ danh sách các tài sản, do tập hợp này dễ dàng sử dụng được các phép toán cần thiết từ lớp Collection, ngoài ra cơ chế truy cập theo chỉ số rất cần thiết để thao tác với dữ liệu kiểu bảng. Các phép toán sử dụng gồm: add(), get(), remove(), indexOf().

public class JQLTaiSan extends javax.swing.JFrame {

//dsTaiSan để lưu trữ thông tin của các tài sản

ArrayList<TaiSan> dsTaiSan = new ArrayList<>();

//dsTSWork sử dụng để thao tác với bảng khi lọc thông tin

ArrayList<TaiSan> dsTSWork = new ArrayList<>();

…

private void

tableTSMouseClicked(java.awt.event.MouseEvent evt) {

dongchon = tableTS.getSelectedRow();

if(dongchon !=-1){

tsan = dsTSWork.get(dongchon);

…

}

private void

jButtonThemActionPerformed(java.awt.event.ActionEvent evt) {

…

dsTaiSan.add(tsan);

…

}

private void jButtonXoaActionPerformed(java.awt.event.ActionEvent evt) {

…

tsan = dsTSWork.get(dongchon);

tsan = dsTaiSan.get(dsTaiSan.indexOf(tsan));

…

dsTaiSan.remove(tsan);

…

}

}

* Thao tác File: Sử dụng FileInput-OutputStream và ObjectInput-OutputStream. Đọc ghi file bằng luồng đối tượng sẽ giữ cho dữ liệu danh sách được toàn vẹn, do bản thân cả đối tượng ArrayList<TaiSan> sẽ được lưu lại mà không cần phải phân chia các đối tượng TaiSan con ở trong danh sách rồi lưu lần lượt lại như khi lưu ra file text. Và vì vậy khi đọc thông tin từ file cũng lấy thẳng cả đối tượng ArrayList<TaiSan> mà không cần phải định nghĩa cách thức lưu trữ dữ liệu đọc vào.

public void LuuFile(String fileName, Object obj)

throws Exception{

//Tạo luồng ghi file

FileOutputStream fs=new FileOutputStream(fileName);

//Tạo luồng để serial đối tượng

ObjectOutputStream os=new ObjectOutputStream(fs);

//Chuyển tải đối tượng tới đích (tập tin)

os.writeObject(obj);

//Đóng luồng

fs.close(); os.close();

}

public Object docFile(String fileName) throws Exception{

Object kq=null;

//Tạo luồng đọc file đã được serial

FileInputStream fi=new FileInputStream(fileName);

//Tạo luồng để Deserialize đối tượng

ObjectInputStream ois=new ObjectInputStream(fi);

//Tiến hành khôi phục đối tượng

kq=ois.readObject();

//Đóng luồng

fi.close();ois.close();

return kq;

}

* Mã nguồn chính của chức năng Quản lý thông tin tài sản:

- Thao tác thêm tài sản mới:

private void jButtonThemActionPerformed(java.awt.event.ActionEvent evt) {

//Kiểm tra trường bắt buộc mã tài sản có trống hay không

if(jMaTS.getText().trim().length()==0)

JOptionPane.showMessageDialog(null, "Mã tài sản không được để trống!", "Thông báo", JOptionPane.WARNING\_MESSAGE);

else {

try { //Tạo tài sản mới bằng hàm khởi tạo lấy thông tin từ các điều khiển

tsan = new

TaiSan(jMaTS.getText(),jTenTS.getText(),jDvTinh.getText(),jTinhTrang.getText(),

format.parse(jNgayBH.getText()),jNguonVon.getText(),jSoHuu.getText(),

jMucDich.getText(),jComboDM.getSelectedItem().toString(),

jComboKhoa.getSelectedItem().toString());

//Kiểm tra xem mã tài sản đã có trong danh sách chưa

if(dsTaiSan.contains(tsan))

JOptionPane.showMessageDialog(null, "Đã tồn tại mã tài sản, mời nhập lại!\n","Lỗi", JOptionPane.ERROR\_MESSAGE);

else{

//Thêm tài sản mới vào danh sách

dsTaiSan.add(tsan);

//Hiện thị lại bảng sau khi thêm

LoadTableTS(dsTaiSan);

//Đặt lại điều khiển nhập thông tin

resetTxt();

//Lưu trữ dữ liệu vào file

saveData();

}

} catch (ParseException ex) {

//Bắt lỗi chuyền định dạng dữ liệu ngày/tháng/năm

JOptionPane.showMessageDialog(null, "Hãy kiểm tra dữ liệu, định dạng ngày kiểu MM/dd/yyyy!\n" + ex,"Lỗi", JOptionPane.ERROR\_MESSAGE);

}

}

}

- Thao tác sửa tài sản:

private void jButtonSuaActionPerformed(java.awt.event.ActionEvent evt) {

//Lấy vị trí của tài sản trong bảng trên giao diện

dongchon = tableTS.getSelectedRow();

if(dongchon != -1){

int response = JOptionPane.showConfirmDialog(null, "Bạn có muốn sửa không?", "Thông báo", JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE);

if (response == JOptionPane.YES\_OPTION){

try {

//Lấy tài sản từ danh sách tạm dsTSWork làm việc với bảng

tsan = dsTSWork.get(dongchon);

//Lấy tài sản từ danh sách gốc để sửa

tsan = dsTaiSan.get(dsTaiSan.indexOf(tsan));

//Thực hiện sửa dữ liệu từ các điều khiển bằng hàm setter

tsan.setTenTS(jTenTS.getText());

tsan.setTinhTrang(jTinhTrang.getText());

tsan.setDvTinh(jDvTinh.getText());

tsan.setNgayBH(format.parse(jNgayBH.getText()));

tsan.setNguonVon(jNguonVon.getText());

tsan.setSoHuu(jSoHuu.getText());

tsan.setMucDich(jMucDich.getText());

tsan.setDanhMuc(jComboDM.getSelectedItem().toString());

tsan.setKhoaQl(jComboKhoa.getSelectedItem().toString());

//Hiển thị lại bảng sau khi sửa

LoadTableTS(dsTaiSan);

//Đặt lại các điều khiển

resetTxt();

//Lưu dữ liệu vào file

saveData();

} catch (ParseException ex) {

//Bắt lỗi chuyền định dạng dữ liệu ngày/tháng/năm

JOptionPane.showMessageDialog(null, "Hãy kiểm tra dữ liệu, định dạng ngày kiểu MM/dd/yyyy!\n" + ex,"Lỗi", JOptionPane.ERROR\_MESSAGE);

}

}

}else

//Bắt lỗi người dùng không chọn bản ghi

JOptionPane.showMessageDialog(null, "Cần chọn một bản ghi để sửa!", "Thông báo", JOptionPane.WARNING\_MESSAGE);

}

- Thao tác xóa tài sản:

private void jButtonXoaActionPerformed(java.awt.event.ActionEvent evt) {

dongchon = tableTS.getSelectedRow();

if (dongchon != -1){

tsan = dsTSWork.get(dongchon);

tsan = dsTaiSan.get(dsTaiSan.indexOf(tsan));

int response = JOptionPane.showConfirmDialog(null, "Bạn có muốn xoá không?", "Thông báo", JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE);

if (response == JOptionPane.YES\_OPTION){

try {

dsTaiSan.remove(tsan);

LoadTableTS(dsTaiSan);

resetTxt();

saveData();

} catch (Exception ex) {

JOptionPane.showMessageDialog(null, ex, "Thông báo", JOptionPane.WARNING\_MESSAGE);

}

}

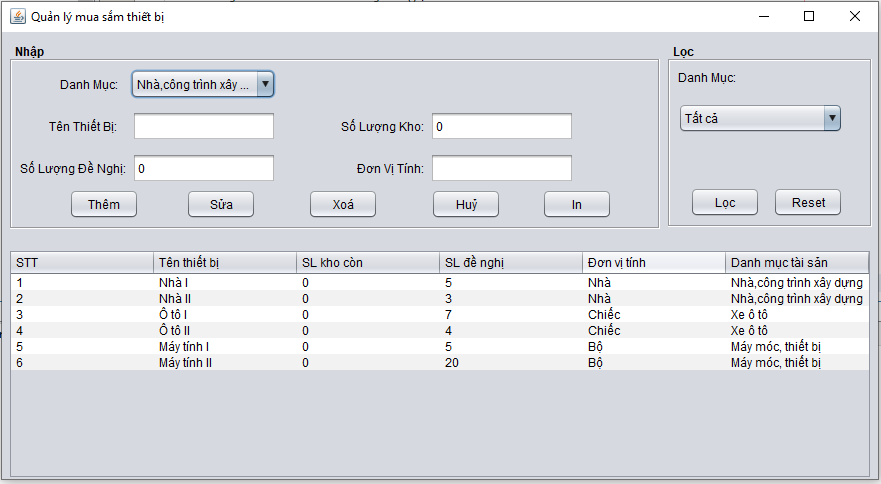
}else

JOptionPane.showMessageDialog(null, "Cần chọn một bản ghi để xóa!", "Thông báo", JOptionPane.WARNING\_MESSAGE);

}

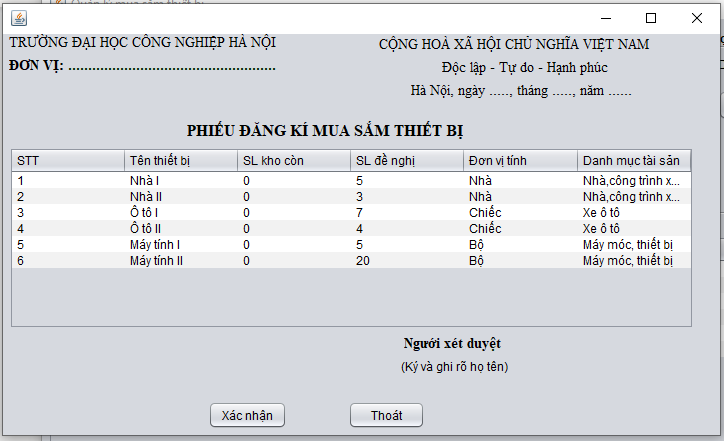
### 2.4.2. Nguyễn Thị Oanh – Quản lý mua sắm tài sản

* Giao diện chức năng “Quản lý mua sắm tài sản”:



Hình 2. 3: Giao diện Quản lý mua sắm tài sản – Nguyễn Thị Oanh

Trên khung màn hình nhập nhấn chọn nút “In” danh sách hiện có sẽ được đưa vào một biểu mẫu có sẵn và đưa lên màn hình.



Hình 2. 4: Giao diện Quản lý mua sắm tài sản - Thao tác in danh sách mua sắm – Nguyễn Thị Oanh.

* Mô tả mã lệnh thực hiện chức năng “Quản lý mua sắm tài sản”:
* Hướng đối tượng :

- Áp dụng tính chất trừu tượng và đóng gói để đưa đối tượng thiết bị về lớp ThietBi lưu trữ các thông tin như sau:

public class ThietBi implements Serializable {

private String tenTB;

private int sLKho;

private int sLDeNghi;

private String dvTinh;

private String danhMuc;

…

}

- Áp dụng tính chất kế thừa và đa hình để tạo nên lớp TableThietBi biểu diễn thông tin thiết bị theo dạng bảng:

public class TableThietBi extends AbstractTableModel{

private final String Name[] = {};

private final Class classess[] = {};

ArrayList<ThietBi> dsTB = new ArrayList<>();

public TableThietBi(ArrayList<ThietBi> tb){

…

@Override

public Object getValueAt(int rowIndex, int columnIndex){

return switch (columnIndex) {

case 0 -> rowIndex + 1;

case 1 -> dsTB.get(rowIndex).getTenTB();

case 2 -> dsTB.get(rowIndex).getsLKho();

case 3 -> dsTB.get(rowIndex).getsLDeNghi();

case 4 -> dsTB.get(rowIndex).getDvTinh();

case 5 -> dsTB.get(rowIndex).getDanhMuc();

default -> null;

};

}

…

}

* Bắt lỗi và gom rác: sử dụng cơ chế try - catch và if - else

- Bắt lỗi lưu dữ liệu:

private void saveData() {

try {

db.LuuFile(fname,dsThietBi);

} catch (Exception ex) {

JOptionPane.showMessageDialog(null, "Lỗi lưu dữ liệu:\n" +ex,"Lỗi", JOptionPane.ERROR\_MESSAGE);

}

}

- Bắt lỗi thêm thiết bị mua sắm:

private void BtnThemActionPerformed(java.awt.event.ActionEvent evt) {

if(jTenTB.getText().trim().length()==0)

JOptionPane.showMessageDialog(null, "Tên thiết bị không được để trống!", "Thông báo", JOptionPane.WARNING\_MESSAGE);

else {

…

if(dsThietBi.contains(tbi))

JOptionPane.showMessageDialog(null, "Đã tồn tại mã thiết bị, mời nhập lại!\n","Lỗi", JOptionPane.ERROR\_MESSAGE);

else{

…

}

* Tập hợp:

- Sử dụng ArrayList để lưu trữ danh sách các tài sản, do tập hợp này dễ dàng sử dụng được các phép toán cần thiết từ lớp Collection, ngoài ra cơ chế truy cập theo chỉ số rất cần thiết để thao tác với dữ liệu kiểu bảng.

- Các phép toán sử dụng: add(), remove(),get()

public class JMSThietBi extends javax.swing.JFrame {

ArrayList<ThietBi> dsThietBi = new ArrayList<>();

int dongchon = -1;

private void BtnThemActionPerformed(java.awt.event.ActionEvent evt) {

…

dsThietBi.add(tbi);

…

}

private void BtnSuaActionPerformed(java.awt.event.ActionEvent evt) {

…

tbi = dsThietBi.get(dongchon);

…

}

private void BtnXoaActionPerformed(java.awt.event.ActionEvent evt) {

…

dsThietBi.remove(tbi);

…

}

…

}

* Thao tác file :

- Đọc ghi file theo hướng đối tượng sử dụng các lớp ObjectInputStream, ObjectOutputStream, FileInputStream, FileOutputStream.

- Lưu file:

public void LuuFile(String fileName, Object obj)

throws Exception{

//Tạo luồng ghi file

FileOutputStream fs=new FileOutputStream(fileName);

//Tạo luồng để serial đối tượng

ObjectOutputStream os=new ObjectOutputStream(fs);

//Chuyển tải đối tượng tới đích (tập tin)

os.writeObject(obj);

//Đóng luồng

fs.close(); os.close();

}

- Đọc file:

public Object docFile(String fileName) throws Exception{

Object kq=null;

//Tạo luồng đọc file đã được serial

FileInputStream fi=new FileInputStream(fileName);

//Tạo luồng để Deserialize đối tượng

ObjectInputStream ois=new ObjectInputStream(fi);

//Tiến hành khôi phục đối tượng

kq=ois.readObject();

//Đóng luồng

fi.close();ois.close();

return kq;

}

* Mã nguồn chính của chức năng

- Ngoài các chức năng thêm, sửa, xoá. Hệ thống còn có chức năng In phiếu:

public class JMSThietBi extends javax.swing.JFrame {

…

private void BtnInActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

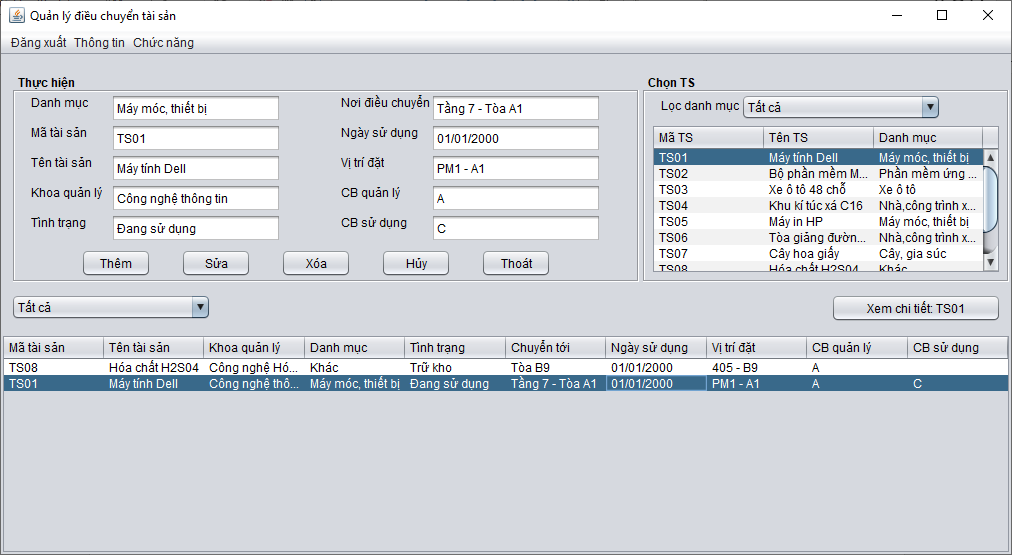
JMSThietBi\_Phieu mSTBForm = new JMSThietBi\_Phieu();

mSTBForm.setVisible(true);

}

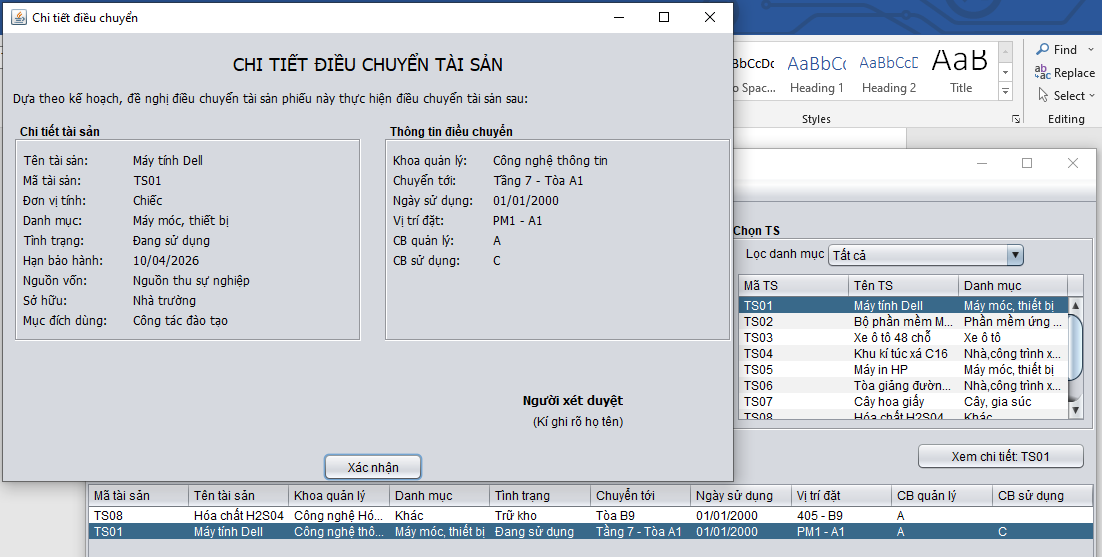
### 2.4.3. Trần Hồng Quân – Quản lý điều chuyển tài sản

* Giao diện chức năng “Quản lý điều chuyển tài sản”:



Hình 2. 5. Giao diện Quản lý điều chuyển tài sản – Trần Hồng Quân

Thực hiện thao tác xem chi tiết điều chuyển tài sản, chọn một dòng tài sản điều chuyển và bấm chọn “Xem chi tiết”, thông tin của tài sản được chọn sẽ được đưa vào biểu mẫu có sẵn và hiện lên màn hình.



Hình 2. 6. Giao diện Quản lý điều chuyển tài sản - Thao tác in chi tiết điều chuyển - Trần Hồng Quân

* Mô tả mã lệnh thực hiện chức năng “Quản lý điều chuyển tài sản”:
* Hướng đối tượng:

- Áp dụng được tính chất trừu tượng và đóng gói để đưa được đối tượng tài sản điều chuyển về lớp TSDieuChuyen lưu các thông tin như sau:

public class TSDieuChuyen implements Serializable{

private String noiDC;

private String cbQL;

private String cbSD;

private String vtDat;

private Date ngaySD;

private TaiSan tsX;

…

}

- Áp dụng tính chất kế thừa và đa hình để tạo lên lớp TableTSDieuChuyen để biểu diễn thông tin theo dạng bảng:

//Lớp TableTSDieuChuyen kế thừ lớp AbstractTableModel

public class TableTSDieuChuyen extends AbstractTableModel{

//Mảng chuỗi Name để tạo tiêu đề

private String Name[];

//Mảng class để hiển thị thông tin từ lấy từ lớp tài sản điều chuyển

private Class classess[];

//Thuộc tính dsTC để lưu thông tin về các tài sản điều chuyển lấy từ file dữ liệu

ArrayList<TSDieuChuyen> dsDC = new ArrayList<>();

…

// Hàm getValueAt thể hiện tính đa hình định nghĩa lại cách lấy thông tin từ lớp tài sản điều chuyển

@Override

public Object getValueAt(int rowIndex, int columnIndex) {

switch(columnIndex){

case 0: return dsDC.get(rowIndex).getTsX().getMaTS();

case 1: return dsDC.get(rowIndex).getTsX().getTenTS();

case 2: return dsDC.get(rowIndex).getTsX().getKhoaQl();

case 3: return dsDC.get(rowIndex).getTsX().getDanhMuc();

case 4: return dsDC.get(rowIndex).getTsX().getTinhTrang();

case 5: return dsDC.get(rowIndex).getNoiDC();

case 6: return dsDC.get(rowIndex).getNgaySD();

case 7: return dsDC.get(rowIndex).getVtDat();

case 8: return dsDC.get(rowIndex).getCbQL();

case 9: return dsDC.get(rowIndex).getCbSD();

default : return null;

}

}

}

* Bắt lỗi và gom rác: sử dụng cơ chế try-catch và if-else.

- Cơ chế try-catch để bắt những ngoại lệ (exception) xảy ra do vi phạm nguyên tắc của ngôn ngữ lập trình:

//Hàm readData có thể sẽ gặp lỗi khi đọc dữ liệu từ file lúc này sẽ bắt ngoại lệ được hệ thống ngôn ngữ báo ra.

public void readData(){

try {

dsTaiSan = (ArrayList<TaiSan>) db.docFile(fDsTS);

dsDC = (ArrayList<TSDieuChuyen>) db.docFile(fDsDC);

} catch (IOException | ClassNotFoundException ex) {

JOptionPane.showMessageDialog(null, "Có danh sách rỗng!");

} catch (Exception ex) {

JOptionPane.showMessageDialog(null, "Lỗi đọc dữ liệu:\n" +ex,"Lỗi", JOptionPane.ERROR\_MESSAGE);

}

}

- Cơ chế if-else sử dụng để bắt ra những lỗi có thể đoán được từ thao tác của người dùng:

//Có khả năng người dùng sẽ không chọn tài sản để thực hiện điều chuyển, lúc này sử dụng if-else để báo lại cho người dùng

private void jButtonThemActionPerformed(java.awt.event.ActionEvent evt) {

chonts = jTableChonTS.getSelectedRow();

if(chonts == -1)

JOptionPane.showMessageDialog(null, "Hãy chọn tài sản để điều chuyển!\n","Thông báo", JOptionPane.WARNING\_MESSAGE);

else{…}

}

* Tập hợp: Sử dụng ArrayList để lưu trữ danh sách các tài sản điều chuyển, do kích thước của ArrayList có thể thay đổi tăng hoặc giảm tùy vào số lượng phần tử trong nó, việc thêm và xóa phần tử ở bất kì vị trí nào trong mảng cũng nhanh chóng, hiệu quả.

Các phép toán sử dụng gồm: add(), get(), remove(), indexOf().

public class JQLDieuChuyen extends javax.swing.JFrame {

…

//dsDC để lưu trữ thông tin của các tài sản điều chuyển

ArrayList<TSDieuChuyen> dsDC = new ArrayList<>();

//dsDCWork sử dụng để thao tác với bảng khi lọc thông tin

ArrayList< TSDieuChuyen > dsDCWork = new ArrayList<>();

…

private void

tableTSMouseClicked(java.awt.event.MouseEvent evt) {

dongchon = tableDC.getSelectedRow();

if(dongchon !=-1){

dchuyen = dsDCWork.get(dongchon);

…

}

private void

jButtonThemActionPerformed(java.awt.event.ActionEvent evt) {

…

dsDC.add(dchuyen);

…

}

private void jButtonXoaActionPerformed(java.awt.event.ActionEvent evt) {

…

dchuyen = dsDCWork.get(dongchon);

dchuyen = dsDC.get(dsDC.indexOf(dchuyen));

…

dsDC.remove(dchuyen);

…

}

}

* Thao tác file: Sử dụng FileInput-OutputStream và ObjectInput-OutputStream. Đọc ghi file bằng luồng đối tượng sẽ giữ cho dữ liệu danh sách được toàn vẹn, do bản thân cả đối tượng ArrayList<TSDieuChuyen> sẽ được lưu lại mà không cần phải phân chia các đối tượng TaiSan con ở trong danh sách rồi lưu lần lượt lại như khi lưu ra file text. Và vì vậy khi đọc thông tin từ file cũng lấy thẳng cả đối tượng ArrayList<TSDieuChuyen> mà không cần phải định nghĩa cách thức lưu trữ dữ liệu đọc vào.

public void LuuFile(String fileName, Object obj)

throws Exception{

//Tạo luồng ghi file

FileOutputStream fs=new FileOutputStream(fileName);

//Tạo luồng để serial đối tượng

ObjectOutputStream os=new ObjectOutputStream(fs);

//Chuyển tải đối tượng tới đích (tập tin)

os.writeObject(obj);

//Đóng luồng

fs.close(); os.close();

}

public Object docFile(String fileName) throws Exception{

Object kq=null;

//Tạo luồng đọc file đã được serial

FileInputStream fi=new FileInputStream(fileName);

//Tạo luồng để Deserialize đối tượng

ObjectInputStream ois=new ObjectInputStream(fi);

//Tiến hành khôi phục đối tượng

kq=ois.readObject();

//Đóng luồng

fi.close();ois.close();

return kq;

}

* Mã nguồn của các chức năng chính trong Quản lý điều chuyển tài sản:

Chức năng “Quản lý điều chuyển tài sản” bao gồm các thao tác thêm, sửa, xóa tương tự như của chức năng “Quản lý thông tin tài sản”, và có thêm thao tác xem chi tiết thông tin điều chuyển tài sản:

private void jButtonChiTietActionPerformed(java.awt.event.ActionEvent evt) {

// Lấy ra tài sản để thực hiện xem chi tiết

dongchon = jTableDC.getSelectedRow();

dchuyen = dsDCWork.get(dongchon);

dchuyen = dsDC.get(dsDC.indexOf(dchuyen));

//Hiển thị thông tin chi tiết với mẫu phiếu có sẵn

JXemChiTiet chiTietDC = new JXemChiTiet(dchuyen);

chiTietDC.setVisible(true);

}

Mã nguồn đưa thông tin tài sản điều chuyển được chọn lên màn hình:

public class JXemChiTiet extends javax.swing.JFrame {

//Khai báo một tài sản điều chuyển để thực hiện gán thông tin

TSDieuChuyen dchuyen = new TSDieuChuyen();

…

public JXemChiTiet(TSDieuChuyen tsdc) {

initComponents();

setTitle("Chi tiết điều chuyển");

//Gán thông tin lấy từ danh sách vào đối tượng nhận

dchuyen = tsdc;

setInfoDC();

setInforTS();

}

// Thực hiện hiển thị thông tin chi tiết về điều chuyển

private void setInforTS(){

jLabelTen.setText(dchuyen.getTsX().getTenTS());

jLabelMa.setText(dchuyen.getTsX().getMaTS());

jLabelDV.setText(dchuyen.getTsX().getDvTinh());

jLabelTTrang.setText(dchuyen.getTsX().getTinhTrang());

jLabelDM.setText(dchuyen.getTsX().getDanhMuc());

jLabelBaoHanh.setText(dchuyen.getTsX().getNgayBH());

jLabelVon.setText(dchuyen.getTsX().getNguonVon());

jLabelSoHuu.setText(dchuyen.getTsX().getSoHuu());

jLabelMD.setText(dchuyen.getTsX().getMucDich());

jLabelKhoa.setText(dchuyen.getTsX().getKhoaQl());

}

private void setInfoDC(){

jLabelVTDC.setText(dchuyen.getNoiDC());

jLabelNgay.setText(dchuyen.getNgaySD());

jLabelCBQL.setText(dchuyen.getCbQL());

jLabelCBSD.setText(dchuyen.getCbSD());

jLabelVT.setText(dchuyen.getVtDat());

}

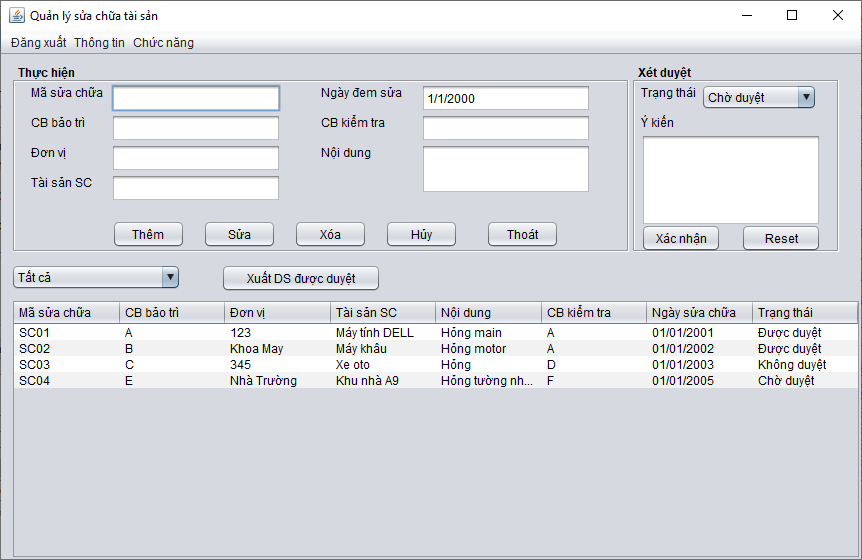
//Kết thúc thao tác hiện

…

}

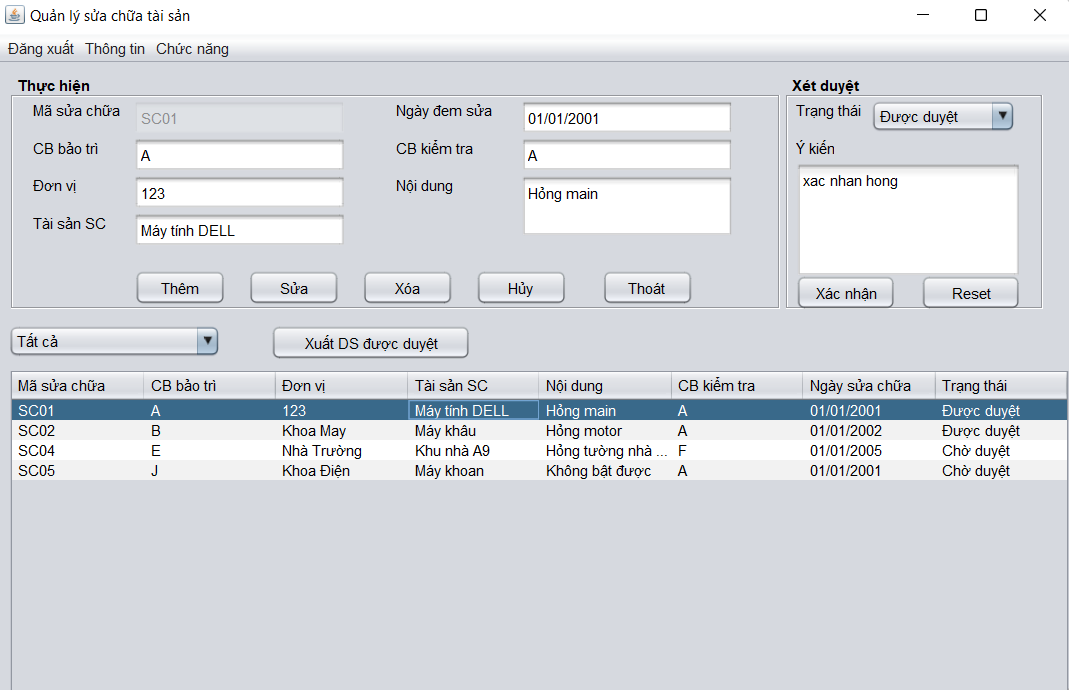
### 2.4.4. Vũ Xuân Thương – Quản lý sửa chữa tài sản

* Giao diện chức năng “Quản lý sửa chữa tài sản”:



Hình 2. 7. Giao diện Quản lý sửa chữa tài sản – Vũ Xuân Thương

Thực hiện xét duyệt đề nghị sửa chữa tài sản từ điều khiển trong khung “Xét duyệt ” và nhấn nút “Xác nhận” để chuyển đổi trạng thái đề nghị sửa chữa tài sản mới vào danh sách.



Hình 2. 8. Giao diện Quản lý sửa chữa tài sản - Thao tác xét duyệt sửa chữa - Vũ Xuân Thương

* Mô tả mã lệnh thực hiện chức năng “Quản lý sửa chữa tài sản”:
* Hướng đối tượng:

- Áp dụng được tính chất trừu tượng và đóng gói để đưa được đối tượng tài sản về lớp SuaChua lưu trữ các thông tin như sau:

public class SuaChua implements Serializable {

private String maSC;

private String nguoiBT;

private String dvTinh;

private String maTS;

private String noiDung;

private String CBKiemTra;

private Date ngaySC;

private String trangThai;

private String yKien;

…

}

- Áp dụng tính chất kế thừa và đa hình để tạo lên lớp TableSuaChua biểu diễn thông tin tài sản theo dạng bảng:

//Lớp TableSuaChua kế thừ lớp AbstractTableModel

public class TableSuaChua extends AbstractTableModel {

//Mảng chuỗi Name để tạo tiêu đề

private String Name[];

//Mảng class để hiển thị thông tin từ lấy từ lớp lưu dữ liệu

private Class classess[] ;

//Thuộc tính dsTS để lấy thông tin về các đề nghị sửa chữa từ file dữ liệu

ArrayList<SuaChua> dsTT = new ArrayList<>();

…

//Hàm getValueAt thể hiện tính đa hình định nghĩa lại cách lấy thông tin

@Override

public Object getValueAt(int rowIndex, int columnIndex) {

switch(columnIndex){

case 0: return dsTT.get(rowIndex).getMaSC();

case 1: return dsTT.get(rowIndex).getNguoiBT();

case 2: return dsTT.get(rowIndex).getDvTinh();

case 3: return dsTT.get(rowIndex).getMaTS();

case 4: return dsTT.get(rowIndex).getNoiDung();

case 5: return dsTT.get(rowIndex).getCBKiemTra();

case 6: return dsTT.get(rowIndex).getNgaySC();

case 7: return dsTT.get(rowIndex).getTrangThai();

case 8: return dsTT.get(rowIndex).getyKien();

default : return null;

}

…

}

* Bắt lỗi và gom rác: Sử dụng cơ chế try-catch và if-else.

- Cơ chế try-catch để bắt những ngoại lệ (exception) xảy ra do vi phạm nguyên tắc của ngôn ngữ lập trình:

//Hàm saveData có thể sẽ gặp lỗi khi lưu dữ liệu vào file lúc này sẽ bắt ngoại lệ được hệ thống ngôn ngữ báo ra.

public void saveData(){

try {

db.LuuFile(fname,dsSC);

} catch (Exception ex) {

JOptionPane.showMessageDialog(null, "Lỗi lưu dữ liệu:\n" +ex,"Lỗi", JOptionPane.ERROR\_MESSAGE);

}

}

- Cơ chế if-else để bắt lỗi nhập dữ liệu:

//Khi người sử dụng nhập thiếu thuộc tính mã sửa chữa. Thông báo lỗi "Mã sửa chữa không được để trống!"

if(jMaSC.getText().trim().length()==0 || jMaTS.getText().trim().length()==0)

JOptionPane.showMessageDialog(null, "Mã sửa chữa và Tài sản sửa chữa không được bỏ trống!", "Thông báo", JOptionPane.WARNING\_MESSAGE);

else{…}

* Tập hợp:

- Trong bài toán không yêu cầu đặc biệt về cách lưu trữ nên dùng ArrayList vừa đơn giản tiện lợi vừa dễ xử lý.

ArrayList<SuaChua> dsTT = new ArrayList<>();

- Các phép toán sử dụng như thêm phần tử (add) xoá phần tử (remove)

dsTTinh.add(ttinh);

dsTTinh.remove(ttinh);

* Thao tác file:

Sử dụng FileInput-OutputStream và ObjectInput-OutputStream. Đọc ghi file bằng luồng đối tượng sẽ giữ cho dữ liệu danh sách được toàn vẹn, do bản thân cả đối tượng ArrayList<SuaChua> sẽ được lưu lại mà không cần phải phân chia các đối tượng TaiSan con ở trong danh sách rồi lưu lần lượt lại như khi lưu ra file text. Và vì vậy khi đọc thông tin từ file cũng lấy thẳng cả đối tượng ArrayList<SuaChua> mà không cần phải định nghĩa cách thức lưu trữ dữ liệu đọc vào.

public void LuuFile(String fileName, Object obj)

throws Exception{

//Tạo luồng ghi file

FileOutputStream fs=new FileOutputStream(fileName);

//Tạo luồng để serial đối tượng

ObjectOutputStream os=new ObjectOutputStream(fs);

//Chuyển tải đối tượng tới đích (tập tin)

os.writeObject(obj);

//Đóng luồng

fs.close(); os.close();

}

public Object docFile(String fileName) throws Exception{

Object kq=null;

//Tạo luồng đọc file đã được serial

FileInputStream fi=new FileInputStream(fileName);

//Tạo luồng để Deserialize đối tượng

ObjectInputStream ois=new ObjectInputStream(fi);

//Tiến hành khôi phục đối tượng

kq=ois.readObject();

//Đóng luồng

fi.close();ois.close();

return kq;

}

* Mã nguồn chính của chức năng :

Các thao tác chính thêm, sửa, xoá đều tương tự như của “Quản lý thông tin tài sản”, và có thêm các thao tác:

- Thao tác xét duyệt :

private void jButtonXNActionPerformed(java.awt.event.ActionEvent evt) {

dongchon = ttinhCSVC.getSelectedRow();

if(dongchon != -1){

ttinh = dsTTinh.get(dongchon);

ttinh.setTrangThai(jComboTT.getSelectedItem().toString());

ttinh.setyKien(jTextYKien.getText());

LoadTableSC(dsTTinh);

resetTxt();

jComboTT.setSelectedIndex(0);

jTextYKien.setText("");

saveData();

}else

JOptionPane.showMessageDialog(null, "Cần chọn một bản ghi để xét duyệt!", "Thông báo", JOptionPane.WARNING\_MESSAGE);

}

- Thao tác xuất danh sách được duyệt:

private void jButtonInDSActionPerformed(java.awt.event.ActionEvent evt) {

ArrayList<SuaChua> dsDuyet = new ArrayList<>();

for(SuaChua itm : dsTTinh)

if(itm.getTrangThai().equalsIgnoreCase("Được duyệt")) dsDuyet.add(itm);

JIn\_DSDuyet inDSDForm = new JIn\_DSDuyet(dsDuyet);

inDSDForm.setVisible(true);

}

# CHƯƠNG III: KẾT LUẬN VÀ BÀI HỌC KINH NGHIỆM

## Nội dung đã thực hiện

* Những kiến thức và kỹ năng đã học được

Để có thể hoàn thành được đề tài “Xây dựng phần mềm quản lý cơ sở vật chất Trường Đại học Công nghiệp Hà Nội” chúng em đã vận dụng những kỹ năng và kiến thức từ các môn học:

* Công nghệ phần mềm: áp dụng được kiến thức phát triển một phần mềm sử dụng mô hình chế thử.
* Phân tích thiết kế hệ thống:

+ Kỹ năng phân tích nghiệp vụ của một tổ chức

+ Kỹ năng đặc tả chức năng phần mềm

+ Kỹ năng thực hiện mô hình hóa các chức năng của hệ thống

+ Kỹ năng thiết kế dữ liệu

+ Kiến thức thiết kế giao diện

* Lập trình Java:

+ Kiến thức về hướng đối tượng để thiết kế các lớp chức năng

+ Kỹ năng về thuật toán xử lý

+ Kỹ năng làm việc với các tập hợp

+ Kỹ năng làm việc với mô hình MVC của Java Swing

* Các kỹ năng mềm:

+ Quản lý thời gian dành cho đề tài

+ Kỹ năng làm việc nhóm

* Những chuẩn đầu ra đạt được

Qua quá trình hoàn thiện bài tập lớn nhóm chúng em đã đạt được các chuẩn đầu ra theo chương trình gồm:

* Chuẩn L3 – Làm việc với vai trò thành viên hoặc trưởng nhóm:

+ Trưởng nhóm – Nguyễn Thị Oanh: phân chia công việc, đôn đốc tiến độ, tổng hợp tài liệu.

+ Các thành viên: hoàn thành công việc được giao, báo cáo lại đúng tiến độ quy trình, sẵn sàng nêu ý kiến hoặc vướng mắc.

* Chuẩn L4 – Triển khai kỹ thuật tổng hợp trong lập trình Java:

Như trình bày ở mục 2.4 các thành viên đều ứng dụng được các kiến thức về hướng đối tượng, kỹ thuật bẫy lỗi – gom rác, làm việc với tập hợp, đọc ghi file và giao diện đồ họa trong từng chức năng được phân công.

## Hướng phát triển

Để chương trình có thể hoạt động hiệu quả hơn cần dữ liệu phải được lưu trữ bằng hệ quản trị cơ sở dữ liệu có tính bảo mật hơn. Cần có sự kết hợp song song giữa quản lý cán bộ và tài sản được phân công một cách chặt chẽ hơn để việc theo dõi và quản lý tài sản nhà trường được tốt hơn.

Nếu có thể đưa chương trình quản lý các thông tin về tài sản nhà trường lên trang Web để giúp cho cán bộ cùng các nhân viên có sự thuận lợi hơn để theo dõi được quá trình quản lý tài sản cho dù là đang ngồi ở nhà hay ở bất kỳ nơi nào.

# TÀI LIỆU THAM KHẢO

**TÀI LIỆU TIẾNG VIỆT**

[1] Giáo trình Lập trình HĐT với Java, Nguyễn Bá Nghiễn, Ngô Văn Bình, Vương Quốc Dũng, Đỗ Sinh Trường; NXB Thống kê, 2020.

[2] Lập trình hướng đối tượng với Java; Đoàn Văn Ban; NXB Khoa học và Kỹ thuật, Hà Nội 2006 (Tái bản).

[3] Lập trình Java nâng cao, Đoàn Văn Ban, NXB Khoa học và Kỹ thuật, Hà Nội 2006.

[4] Bộ slide bài giảng lập trình Java - Bộ môn CNPM- trường ĐHCN HN.

**TÀI LIỆU TIẾNG ANH**

[5] The Java Programming Language; Author: K. Arnold, J. Gosling; Published: Addison-Wesley, 1996, ISBN 0-201-63455-4.

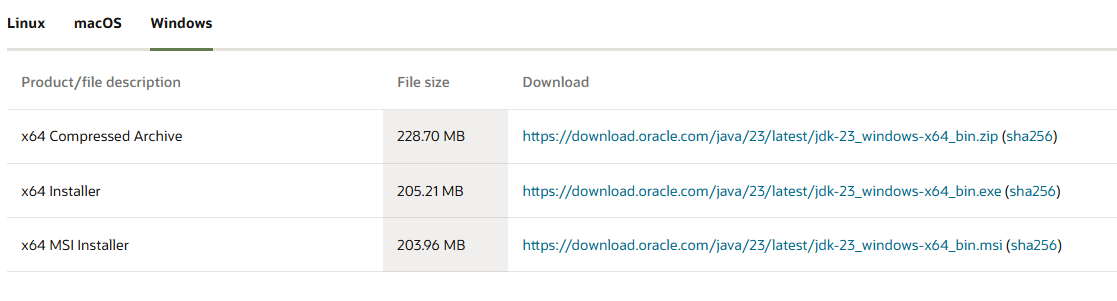
# PHỤ LỤC

**HƯỚNG DẪN CÀI ĐẶT NETBEANS VÀ MÔI TRƯỜNG JDK**

Bước 1. Cài đặt NETBEANS – Nếu máy chưa cài

Bước 2. Cài đặt môi trường JDK *(oracle.com/java/technologies/downloads)*

Bước 3. Chọn nền tảng hệ điều hành và kích vào đường dẫn Download: *x64 Installer*



Bước 4. Chạy file .exe vừa tải xuống

Bước 5. Copy đường dẫn thư mục *bin* (Đường dẫn mặc định là: *C:\Program Files\Java\jdk-23\bin*)

Bước 6. Mở hộp thoại *System Properties* *Environment* *Variables* *System Variables Path Edit* Paste đường dẫn thư mục *bin* ở trên.

Bước 7. Kiểm tra trong hộp thoại *Command Prompt* với lệnh: *java -- version*

