**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO THÍ NGHIỆM/THỰC NGHIỆM**

**LẬP TRÌNH JAVA**

**ĐỀ TÀI:**

**XÂY DỰNG PHẦN MỀM**

**QUẢN LÝ TRỰC NHẬT SINH VIÊN HAUI**

|  |  |
| --- | --- |
| **GVHD:** | Vũ Thị Dương |
| **Sinh viên:** | Hoàng Kim Cường |
|  | Nguyễn Duy Đạo |
|  | Nguyễn Trung Đạt |
|  | Phạm Trọng Nghĩa |
|  | Nguyễn Thị Bích Ngọc |
| **Nhóm:** | 16 |
| **Lớp:** | 20231IT6019001. **Khóa:** K16 |

**Hà Nội - Năm 2023**

**MỤC LỤC**

[DANH MỤC BẢNG BIỂU 3](#_Toc154403097)

[DANH MỤC HÌNH ẢNH 4](#_Toc154403098)

[PHẦN 1: MỞ ĐẦU 5](#_Toc154403099)

[1.1 Tổng quan về báo cáo 5](#_Toc154403100)

[1.2 Lý do chọn đề tài 5](#_Toc154403101)

[1.3 Các kỹ năng cần có để thực hiện đề tài: 5](#_Toc154403102)

[1.4 Mục tiêu mong muốn đạt được 6](#_Toc154403103)

[PHẦN 2: KẾT QUẢ NGHIÊN CỨU 7](#_Toc154403104)

[2.1 Giới thiệu 7](#_Toc154403105)

[2.2 Khảo sát hệ thống 8](#_Toc154403106)

[2.2.1 Khảo sát sơ bộ 8](#_Toc154403107)

[2.2.2 Tài liệu đặc tả yêu cầu 8](#_Toc154403108)

[2.3 Phân tích hệ thống 22](#_Toc154403109)

[2.3.1 Mô hình hóa chức năng hệ thống 22](#_Toc154403110)

[2.3.2 Mô hình hóa dữ liệu và giao diện hệ thống 24](#_Toc154403111)

[2.4 Thực hiện bài toán 29](#_Toc154403112)

[2.4.1 Phạm Trọng Nghĩa – Quản lý lớp học (Homepage) 29](#_Toc154403113)

[2.4.2 Nguyễn Duy Đạo – Quản lý lịch trực nhật 34](#_Toc154403114)

[2.4.3 Nguyễn Trung Đạt, Nguyễn Thị Bích Ngọc – Quản lý phân công trực nhật 39](#_Toc154403115)

[2.4.4 Hoàng Kim Cường – Xem lịch trực nhật 43](#_Toc154403116)

[2.4.5 Hoàng Kim Cường - Đăng nhập 46](#_Toc154403117)

[PHẦN 3: KẾT LUẬN VÀ BÀI HỌC KINH NGHIỆM 51](#_Toc154403118)

[3.1 Nội dung đã thực hiện 51](#_Toc154403119)

[3.2 Hướng phát triển 51](#_Toc154403120)

[TÀI LIỆU THAM KHẢO 53](#_Toc154403121)

# DANH MỤC BẢNG BIỂU

[Bảng 2.1: Mô tả chung về tài liệu 9](#_Toc154402769)

[Bảng 2.2: Mô hình hóa chức năng hệ thống 23](#_Toc154402770)

[Bảng 2.3: Mô tả Use Case 24](#_Toc154402771)

# DANH MỤC HÌNH ẢNH

[Hình 2.1: Apache NetBeans 7](#_Toc154402790)

[Hình 2.2: Giao diện phác thảo màn hình Đăng nhập 12](#_Toc154402791)

[Hình 2.3: Giao diện phác thảo màn hình Trang chủ 13](#_Toc154402792)

[Hình 2.4: Giao diện phác thảo màn hình Quản lý lịch trực nhật 14](#_Toc154402793)

[Hình 2.5: Giao diện phác thảo màn hình Phân công trực nhật 14](#_Toc154402794)

[Hình 2.6: Giao diện phác thảo màn hình Trang chủ trang sinh viên 15](#_Toc154402795)

[Hình 2.7: Giao diện trang Đăng nhập 16](#_Toc154402796)

[Hình 2.8: Giao diện trang Trang chủ 16](#_Toc154402797)

[Hình 2.9: Giao diện trang Quản lý phân công trực nhật 17](#_Toc154402798)

[Hình 2.10: Giao diện trang Sửa thông tin trực nhật 18](#_Toc154402799)

[Hình 2.11: Giao diện trang Quản lý trực nhật sinh viên 19](#_Toc154402800)

[Hình 2.12: Giao diện trang Thêm/Sửa sinh viên trực nhật 20](#_Toc154402801)

[Hình 2.13: Giao diện xác nhận xóa 20](#_Toc154402802)

[Hình 3.14: Giao diện trang Trang chủ trang sinh viên 21](#_Toc154402803)

[Hình 2.15: Biểu đồ Use Case tổng quan 23](#_Toc154402804)

[Hình 2.16: Giao diện phác thảo trang Đăng nhập 25](#_Toc154402805)

[Hình 2.17: Giao diện phác thảo trang Trang chủ 26](#_Toc154402806)

[Hình 2.18: Giao diện phác thảo trang Quản lý lịch trực nhật 27](#_Toc154402807)

[Hình 2.19: Giao diện phác thảo trang Quản lý phân công trực nhật 28](#_Toc154402808)

[Hình 2.20: Giao diện phác thảo trang Trang chủ trang sinh viên 29](#_Toc154402809)

# PHẦN 1: MỞ ĐẦU

## 1.1 Tổng quan về báo cáo

Trong môi trường đại học như HaUI, hoạt động quản lý và phân công sinh viên trực nhật là một phần quan trọng để đảm bảo sự hiệu quả trong các hoạt động học tập tại trường. Đề tài "Xây dựng phần mềm Quản lý trực nhật sinh viên HaUI" tập trung vào việc phát triển một ứng dụng chuyên nghiệp giúp tổ chức và quản lý sinh viên thực hiện các nhiệm vụ trực nhật một cách thuận tiện và hiệu quả

## 1.2 Lý do chọn đề tài

Phân công sinh viên trực nhật là một hoạt động quan trọng trong môi trường đại học, đặc biệt là đối với HaUI với lượng sinh viên đông đảo. Đề tài hướng đến việc giải quyết một vấn đề cụ thể và cấp bách trong quản lý hoạt động sinh viên.

Trong một lớp tại trường, giáo viên có thể nhanh chóng quản lý, phân công các sinh viên trực nhật một cách nhanh chóng, hiệu quả, đảm bảo được tính minh bạch, rõ ràng.

Đề tài tập trung vào việc cải thiện hiệu quả quản lý thông qua việc sử dụng công nghệ. Sự tự động hóa trong phân công và theo dõi tiến trình sẽ giúp giảm thời gian và công sức, đồng thời tăng cường độ chính xác và minh bạch. Sinh viên theo dõi được lịch trực nhật của mình một cách tiện lợi, không bị quên.

Nghiên cứu và xây dựng phần mềm này có thể đóng góp vào sự phát triển của công nghệ giáo dục, mở ra những cơ hội mới để cải thiện quản lý sinh viên và hoạt động giáo dục.

## 1.3 Các kỹ năng cần có để thực hiện đề tài:

Chương trình quản lý trực nhật sinh viên của trường đại học Công Nghiệp Hà Nội là một ứng dụng được xây dựng bằng ngôn ngữ lập trình Java, vì vậy cần có các kỹ năng sau để thực hiện đề tài này:

* Kỹ năng lập trình Java: Hiểu biết về ngôn ngữ lập trình Java, bao gồm cú pháp, lập trình hướng đối tượng, xử lý ngoại lệ, và các tính năng mới của Java.
* Phát triển giao diện người dùng (UI): Thiết kế và phát triển giao diện người dùng bằng java Swing, để tạo ra một giao diện người dùng thân thiện và dễ sử dụng
* Kiểm thử và debugging: Kỹ năng kiểm thử đơn vị (unit testing) và debugging để đảm bảo rằng mã nguồn hoạt động đúng và hiệu quả.
* Kỹ năng giao tiếp và tương tác trong hoạt động nhóm: Giao tiếp mạch lạc để truyền đạt thông tin và yêu cầu một cách rõ ràng, lắng nghe để hiểu rõ ý kiến và nhu cầu của các thành viên nhóm.
* Khả năng sử dụng công nghệ: Hiểu biết và sử dụng các công cụ và phần mềm quản lý dự án và phân công lao động.

## 1.4 Mục tiêu mong muốn đạt được

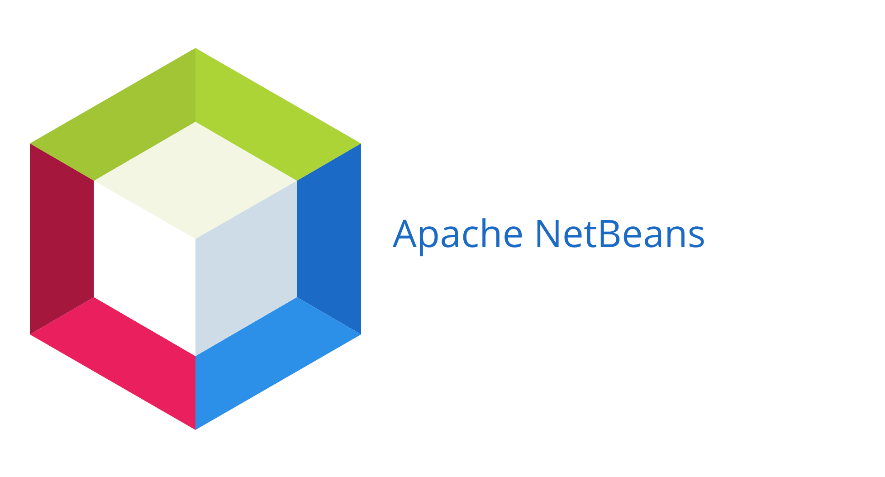
Mục tiêu mong muốn sau khi thực hiện bài tập lớn:

* Thiết kế được giao diện chương trình cơ bản.
* Vận dụng được các kỹ thuật lập trình Java để xử lý tốt các luồng sự kiện của người dùng như thêm, sửa, xóa các dữ liệu.
* Hiển thị các dữ liệu cần thiết giúp cho việc quản lý các hoạt động hợp tác của trường đại học trở nên thuận tiện, đơn giản và hiệu quả nhất.
* Ôn tập, củng cố và nâng cao kiến thức và cài đặt được các kỹ thuật lập trình căn bản của ngôn ngữ Java cách thành thạo.

# PHẦN 2: KẾT QUẢ NGHIÊN CỨU

## 2.1 Giới thiệu

Bài tập lớn nhóm em triển khai dưới đây là một hệ thống quản lý trực nhật, được xây dựng bằng ngôn ngữ lập trình Java và sử dụng thư viện đồ họa Java Swing để tạo giao diện người dùng. Mục tiêu của dự án là cung cấp một nền tảng hiệu quả để quản lý và phân công lịch làm việc cũng như là giám sát công việc trực nhật cho giáo viên và sinh viên. Nhóm đã quyết định sử dụng quy trình phát triển phần mềm Water Fall trong dự án của mình để đảm bảo tính đơn giản, dễ áp dụng trong thời gian làm việc ngắn đối với dự án nhỏ này. Lựa chọn này giúp chúng em nhanh xây dựng nên chức năng và giao diện hệ thống nhanh chóng để đưa vào kiểm thử và tối ưu hóa sản phẩm. Về công cụ, Netbeans IDE được lựa chọn để thực hiện triển khai xây dụng hệ thống này. Lý do là vì sự thuận tiện và khả năng tích hợp mạnh mẽ của Netbeans với ngôn ngữ Java, đặc biệt là trong việc phát triển ứng dụng GUI với Java Swing. Điều này giúp tăng cường hiệu suất làm việc, dễ nắm bắt để sử dụng và giảm độ phức tạp của quá trình thực hiện dự án.



Hình 2.: Apache NetBeans

## 2.2 Khảo sát hệ thống

### 2.2.1 Khảo sát sơ bộ

Cấu trúc tổ chức của đơn vị đặt hàng: Đơn vị đặt hàng là trường Đại học Công Nghiệp Hà Nội với nhiều lớp và lịch giảng dạy khác nhau. Sinh viên sẽ xem lịch trực nhật ở từng lớp mình tham gia học. Giáo viên sẽ quản lý phân công và giám sát lịch trực nhật của sinh viên trong lớp của mình.

Tổng quan các vấn đề bài toán giải quyết: Ứng dụng "Quản lý trực nhật sinh viên HaUI" giải quyết vấn đề phân công, quản lý lịch trực nhật của các sinh viên trong mỗi lớp. Hệ thống giúp sinh viên và giáo viên giám sát lịch làm việc và phân công lao động trực nhật từ đó phụ giúp thầy cô trong công tác giảng dạy, tăng hiệu quả các buổi học và nâng cao ý thức cho sinh viên.

Biểu mẫu tham khảo được thu thập từ các mục “xem thời khóa biểu”, “xem lịch giảng dạy” trên trang chủ nhà trường https://sv.haui.edu.vn/. Nhóm sẽ dựa trên các biểu mẫu này để thiết kế giao diện và tính năng cho hệ thống quản lý trực nhật.

### 2.2.2 Tài liệu đặc tả yêu cầu

#### 2.2.2.1 Giới thiệu

##### 2.2.2.1.1 Mục đích

Mục đích của tài liệu này là để cung cấp mô tả chi tiết về các yêu cầu cho hệ thống quản lý trực nhật. Tài liệu này sẽ minh họa mục đích cụ thể và thông tin chi tiết cho việc phát triển hệ thống.

##### 2.2.2.1.2 Phạm vi

Hệ thống quản lý trực nhật này được cài đặt trên trang sinh viên của nhà trường, giúp giảng viên dễ dàng thực hiện việc quản lý và phân công lao động cho sinh viên trong mỗi tiết học mình đứng lớp. Đồng thời cho phép sinh viên theo dõi lịch trực nhật trong mỗi lớp mình tham gia.

##### 2.2.2.1.3 Định nghĩa

|  |  |  |
| --- | --- | --- |
| **Thuật ngữ** | **Định nghĩa** | **Giải thích** |
| HaUI | Trường Đại học Công nghiệp Hà Nội | Tên trường |
| CSDL | Cơ sở dữ liệu | Nơi lưu trữ thông tin và cho phép truy cập |

##### 2.2.2.1.4 Tài liệu tham khảo

[1] IEEE Software Engineering Standards Committee, “IEEE Std 830- 1998, IEEE Recommended Practice for Software Requirement Specifications”, October 20, 1998.

[2] Nguyễn Bá Nghiễn, Ngô Văn Bình, Vương Quốc Dũng, Đỗ Sinh Trường (2020), *Giáo trình Lập trình hướng đối tượng với Java*, NXB Thống kê, trường Đại học Công nghiệp Hà Nội.

##### 2.2.2.1.5 Mô tả chung về tài liệu

|  |  |  |
| --- | --- | --- |
| Phần | Miêu tả | Mục đích |
| 1. Giới thiệu | Mô tả cách khái quát nhất về phạm vi và mục đích tài liệu cần đạt được | Đưa ra các công việc cần phải thực hiện của tài liệu |
| 2. Mô tả chung về sản phẩm | Khái quát về phần mềm, đưa ra mô hình cho phần mềm | Khái quát phần mềm, chức năng của phần mềm, đối tượng sử dụng và mô hình ngữ cảnh |
| 3. Các yêu cầu chi tiết đối với sản phẩm | Các yêu cầu về chức năng, giao diện, hiệu năng của phần mềm | Trình bày các yêu cầu mà phần mềm cần phải thực thi |

Bảng 2.: Mô tả chung về tài liệu

#### 2.2.2.2 Mô tả chung

##### 2.2.2.2.1 Tổng quan về sản phẩm

Hệ thống phần mềm bao gồm 2 phần: Quản lý việc phân công lao động trực nhật cho giáo viên và theo dõi lịch trực nhật cho sinh viên theo lớp học.

a, Quản lý phân công lao động trực nhật

Giảng viên nhập thông tin lớp học, thực hiện các chức năng thêm, sửa, xóa, tìm kiếm, sắp xếp lớp học theo các đề xuất có sẵn: mã lớp, tên lớp ,ngày bắt đầu. ngày kết thúc, số buổi học, số lượng sinh viên. Giảng viên có thể mở danh sách và theo dõi lịch trực nhật của lớp.

Giảng viên có quyền cập lịch trực nhật, cập nhật sinh viên trực nhật cũng như các tác vụ thêm sửa xóa.

b, Theo dõi lịch trực nhật

Sinh viên sau khi đăng nhập vào hệ thống, lựa chọn mục “Lịch trực nhật” để xem lịch trực nhật của bản thân theo tuần, theo buổi học hoặc theo lớp học.

##### 2.2.2.2.2 Chức năng của sản phẩm

a, Quản lý phân công lao động trực nhật

Cho phép giảng viên thực hiện các tác vụ sau:

* Đăng nhập dưới quyền admin.
* Truy cập vào màn hình chính hệ thống quản lý trực nhật, cho phép thêm sửa xóa tìm kiếm sắp xếp các lớp học do mình quản lý.
* Cho phép theo dõi và thêm sửa xóa lịch phân công trực nhật của sinh viên trong lớp.
* Cho phép cập nhật lại hoặc phân công phân sinh viên trực nhật cho các buổi học trong lớp.

b, Theo dõi lịch trực nhật

Cho phép sinh viên thực hiện các tác vụ sau:

* Đăng nhập dưới quyền user.
* Cho phép sinh viên theo dõi lịch trực nhật của mình trong kỳ học. Trong đó sinh viên sẽ theo dõi được tên lớp, ngày trực nhật, buổi học, các sinh viên khác trực nhật cùng mình trong buổi đó và các lưu ý của giáo viên.

##### 2.2.2.2.3 Đối với người dùng

Phần mềm được ứng dụng cho 2 đối tượng người dùng là: giáo viên và sinh viên trường Đại học Công nghiệp Hà Nội:

* Giáo viên là người trực tiếp sử dụng phần mềm để xếp lịch trực nhật cho các sinh viên trong lớp mình dạy, có quyền phân công lao động, chỉnh sửa hoặc xóa sinh viên khỏi lịch trực nhật.
* Sinh viên là người sử dụng phần mềm để theo dõi lịch trực nhật của bản thân theo tuần, theo buổi học và theo lớp học như đã được phân công.

##### 2.2.2.2.4 Ràng buộc tổng thể

- Hệ thống bị ràng buộc bới giảng viên: chỉ khi giảng viên thực hiện thêm sửa xóa phân công lao động cho các sinh viên trong lớp thì hệ thống mới được cập nhật và hiện thị lịch trực nhật cho sinh viên.

- Ràng buộc về dữ liệu đầu vào so với dữ liệu trong CSDL : cần phải nhập chính xác mã lớp / mã sinh viên để hiện thị các trang liên quan.

- Ràng buộc với thiết bị bên ngoài: máy tính dùng để chạy chương trình cần phải có excel được kết nối với netbeans để upload dữ liệu lên giao diện

##### 2.2.2.2.5 Giả thiết và sự lệ thuộc

Phần mềm được áp dụng trong giả thiết đó là hệ thống trang sinh viên https://sv.haui.edu.vn và trên trang giảng viên https://gv.haui.edu.vn , khi đó phần mềm mới được kết nối với CSDL của nhà trường. Hoặc nếu không cài đặt trên hệ thống của nhà trường mà cài đặt trên máy tính khác buộc phải có CSDL nhà trường thì mới thực hiện được các thao tác chức năng.

#### 2.2.2.3 Yêu cầu chi tiết

##### 2.2.2.3.1 Yêu cầu chức năng

A, Yêu cầu chức năng Đăng nhập – Hoàng Kim Cường

+ Giới thiệu: cho phép sinh viên đăng nhập dưới quyền user hoặc giáo viên dưới quyền admin.

+ Dữ liệu vào: Tài khoản, mật khẩu người dùng nhập vào.

+ Xử lý: Kiểm tra người dùng đã điền đầy đủ thông tin đăng nhập chưa nếu chưa sẽ hiển thị thông báo ra màn hình. Nếu người dùng đã điền đầy đủ thông tin thì sẽ kiểm tra tài khoản, mật khẩu với tài khoản mật khẩu trên hệ thống xem có tồn tại hay không, nếu tồn tại thì sẽ chuyển đến trang theo tên đăng nhập.

A screenshot of a login form

Description automatically generated

Hình 2.: Giao diện phác thảo màn hình Đăng nhập

+ Kết quả: Với tài khoản có tên đăng nhập là “admin” thì sẽ truy cập vào trang chủ bên phía admin, còn tài khoản là mã sinh viên thì sẽ truy cập vào trang xem lịch trực nhật của sinh viên đó.

B, Yêu cầu chức năng Quản lý thông tin lớp học – Phạm Trọng Nghĩa

+ Giới thiệu: Cho phép giảng viên quản lý các lớp mình dạy.

+ Dữ liệu vào: Mã lớp, tên lớp, Ngày bắt đầu, ngày kết thúc, số buổi học, danh sách lớp

+ Xử lý: Kiểm tra mã lớp có bị trùng không, kiểm tra định dạng ngày-thàng- năm của ngày, kiểm tra định dạng số buổi học, kiểm tra đường dẫn file excel. xử lý không để bị trùng sinh viên nếu sinh viên học nhiều hơn 1 lớp

+ Kết quả: Hiện thị thông tin các lớp, màn hình hiện thị gồm các nút “Thêm”, “Sửa”, “Xóa”, “Sắp xếp”, “Tìm kiếm”, cho phép giảng viên thực hiện quản lý các lớp.

A white and green document with black text

Description automatically generated

Hình 2.: Giao diện phác thảo màn hình Trang chủ

C, Yêu cầu chức năng Quản lý lịch trực nhật sinh viên – Nguyễn Duy Đạo

+ Giới thiệu: Cho phép giảng viên quản lý lịch trực nhật của lớp mình dạy.

+ Dữ liệu vào: danh sách lớp, danh sách trực nhật

+ Xử lý: Kiểm tra người dùng đã tạo các lớp học chưa. Nếu người dùng đã tạo các lớp học, khi click vào một lớp học trong danh sách lớp học , hệ thống sẽ kiểm tra mã lớp và đưa ra lịch trực nhật của các sinh viên trong lớp học đấy

+ Kết quả: Hiện thị danh sách lịch trực nhật của lớp đó, màn hình hiện thị gồm các nút “sửa”, “chia trực nhật”, “sửa chi tiết”, “xóa”, “trở về”, “tìm kiếm”. Cho phép giảng viên thực hiện quản lý lịch trực nhật của lớp, hoặc chuyển sang các tác vụ “Quản lý phân công trực nhật”.

A screenshot of a computer

Description automatically generated

Hình 2.: Giao diện phác thảo màn hình Quản lý lịch trực nhật

D, Yêu cầu chức năng Quản lý phân công trực nhật – Nguyễn Trung Đạt

+ Giới thiệu: Cho phép giảng viên phân công lao động trực nhật cho các sinh viên trong một buổi.

+ Dữ liệu vào: Danh sách trực nhật, danh sách sinh viên

+ Xử lý: Chức năng này bắt đầu khi người dùng click vào nút “Sửa chi tiết” ở trang “Quản lý lịch trực nhật”. Hệ thống sẽ kiểm tra và hiện thị danh sách sinh viên trực nhật trong buổi được chọn.

+ Kết quả: Hiện thị trang “Quản lý phân công trực nhật” với các chức năng thêm, sửa, xóa sinh viên đang trực nhật trong buổi đó.

A screenshot of a computer

Description automatically generated

Hình 2.: Giao diện phác thảo màn hình Phân công trực nhật

E, Hiện thị lịch trực nhật trên trang sinh viên – Nguyễn Thị Bích Ngọc

+ Giới thiệu: Hiển thị lịch trực nhật của sinh viên theo từng lớp sinh viên đang tham gia học tập.

+ Dữ liệu vào: thông tin lớp học sinh viên đang tham gia: mã lớp, tên lớp, số buổi học, danh sách sinh viên lớp học, lưu ý của giảng viên

+ Xử lý: Sau khi người dùng đăng nhập với tài khoản là mã sinh viên, hệ thống sẽ lấy các buổi trực nhật của sinh viên ý hiển thị ra màn hình gồm tên lớp, ngày trực nhật, những ai trực nhật, lưu ý.

+ Kết quả: khi sinh viên đăng nhập bằng tài khoản sinh viên sẽ hiển thị lịch trực nhật của sinh viên theo danh sách lớp học

A green and white card with white text

Description automatically generated

Hình 2.: Giao diện phác thảo màn hình Trang chủ trang sinh viên

#### 2.2.2.3.2 Yêu cầu giao diện ngoài

A, Giao diện người dùng

***Chức năng Đăng nhập:***



Hình 2.: Giao diện trang Đăng nhập

Màn hình hiện ra giao diện đăng nhập chứa thông tin cần điền để truy cập tài khoản. Khi đăng nhập sai, màn hình giao diện hiện thông báo và cho hiển thị màn hình nhập lại.

**Trang chủ phía Admin**

A screenshot of a computer

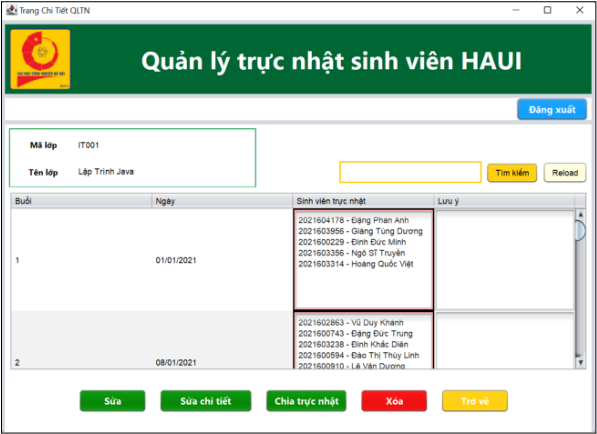
Description automatically generated

Hình 2.: Giao diện trang Trang chủ

Khi admin thực hiện đăng nhập mở phần mềm Quản lý trực nhật sinh viên HaUI, trang chủ sẽ hiển thị ra màn hình.

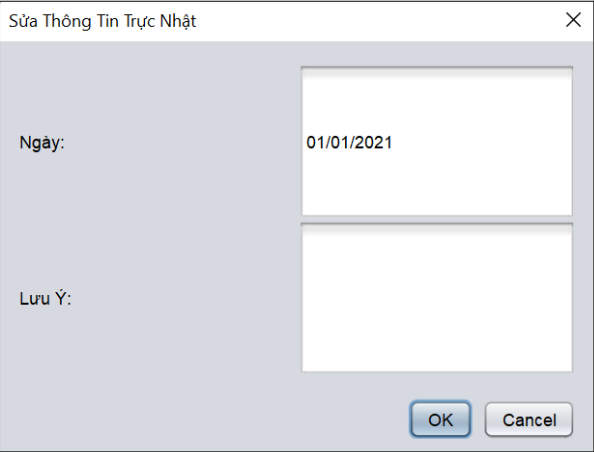
Trang chủ gồm các thành phần chức năng và Table hiển thị danh sách thông tin các lớp học. Các chức năng: Trang chủ (trở về trang Trang chủ),  Đăng xuất (Thoát khỏi trang chủ, trở về trang Đăng nhập), Thêm (thêm thông tin một lớp học), Sửa (sửa thông tin một lớp học), Xóa (xóa thông tin một lớp học), Sắp xếp (sắp xếp danh sách lớp học theo các đề xuất), Tìm kiếm (tìm kiếm lớp học theo mã lớp, tên lớp), Reload (tải lại bảng hiển thị danh sách lớp học). Trang chứa các TextBox để nhập thông tin lớp học: mã lớp, tên lớp, số buổi học, ngày bắt đầu, ngày kết thúc, danh sách lớp (kích nút “Chọn file” để chọn danh sách sinh viên lớp học).

**Giao diện quản lý phân công trực nhật**

****

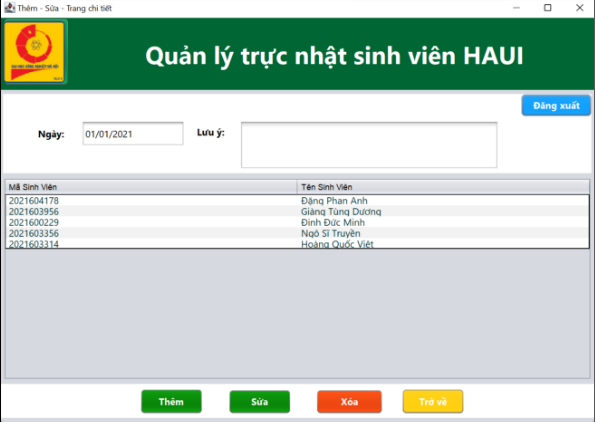
Hình 2.: Giao diện trang Quản lý phân công trực nhật

Khi admin kích đúp chuột vào một hàng trong Table hiển thị danh sách thông tin các lớp học, màn hình hiển thị ra giao diện **Quản lý phân công trực nhật** chứa các chức năng và Table hiển thị lịch phân công trực nhật**.** Các chức năng: Trang chủ (trở về trang Trang chủ), Đăng xuất (Thoát khỏi trang chủ, trở về trang Đăng nhập), Sửa (sửa thông tin một buổi trực nhật), Sửa chi tiết (sửa thông tin phân công sinh viên trực nhật trong một buổi), Xóa (xóa thông tin một buổi trực nhật), Chia trực nhật (phân chia lịch trực nhật), Tìm kiếm (tìm kiếm lớp học theo mã lớp, tên lớp), Reload (tải lại bảng hiển thị lịch phân công trực nhật),Trở về (chuyển về trang trước). Khi admin kích vào nút “Sửa”, màn hình hiển thị ra màn hình trang phụ “Sửa thông tin trực nhật”: nhập vào thông tin ngày và lưu ý cho lịch trực nhật.



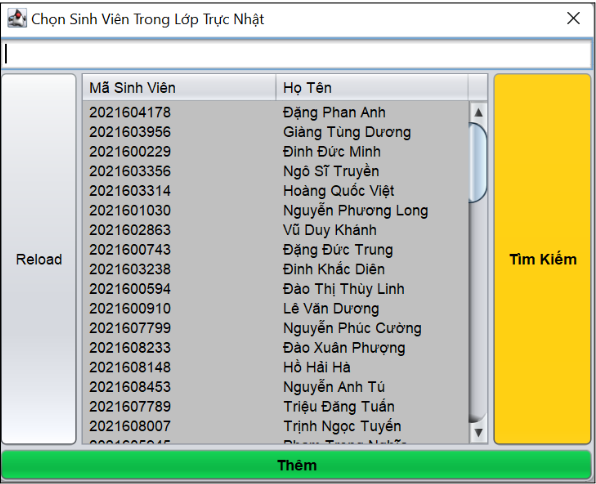
Hình 2.: Giao diện trang Sửa thông tin trực nhật

**Giao diện quản lý trực nhật sinh viên:**

****

Hình 2.: Giao diện trang Quản lý trực nhật sinh viên

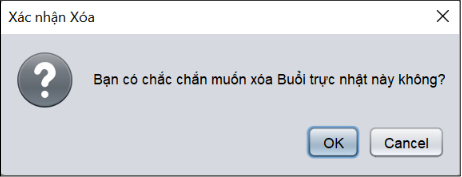
* Khi admin chọn một hàng trong Table hiển thị lịch phân công trực nhật và kích nút “Sửa chi tiết”, màn hình hiển thị trang **Quản lý lịch trực nhật sinh viên** chứa các chức năng và Table hiển thị thông tin sinh viên trực nhật của buổi học. Các chức năng: Trang chủ (trở về trang Trang chủ), Đăng xuất (Thoát khỏi trang chủ, trở về trang Đăng nhập), Thêm (thêm sinh viên trực nhật), Sửa (sửa sinh viên trực nhật), Xóa (xóa sinh viên trực nhật), Trở về (chuyển về trang trước).
* Khi admin kích một sinh viên sau đó ấn nút “Sửa” hoặc admin kích “Thêm”, màn hình hiển thị ra màn hình trang phụ “Chọn sinh viên trong lớp trực nhật”: nhập vào thông tin để tìm sinh viên và kích nút “Sửa” hoặc “Thêm” để xác nhận.



Hình 2.: Giao diện trang Thêm/Sửa sinh viên trực nhật

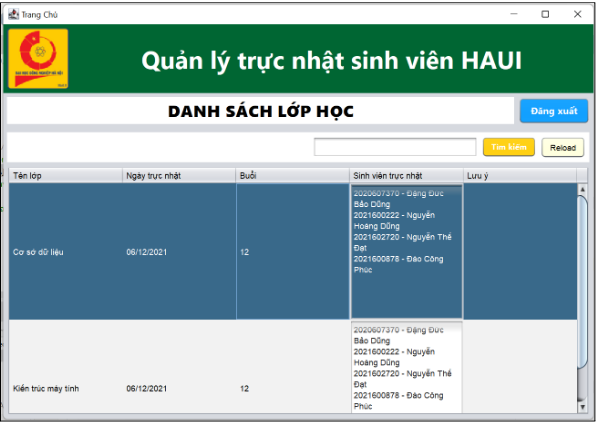
**Chức năng xóa:**

Khi admin kích nút “Xóa” để xóa một đối tượng, màn hình sẽ hiển thị màn hình phụ xác nhận xóa. Nếu người dùng nhấn “OK”, hệ thống sẽ xóa đối tượng muốn xóa, ngược lại, nếu nhấn “Cancel”, màn hình sẽ hiển thị như ban đầu.

****

Hình 2.: Giao diện xác nhận xóa

**Trang chủ phía sinh viên:**



Hình 3.: Giao diện trang Trang chủ trang sinh viên

* Khi  admin thực hiện đăng nhập mở phần mềm Quản lý trực nhật sinh viên HaUI, trang chủ sẽ hiển thị ra màn hình.
* Trang chủ gồm các thành phần chức năng và Table hiển thị danh sách thông tin các lớp học. Các chức năng: Trang chủ (trở về trang Trang chủ), Đăng xuất (Thoát khỏi trang chủ, trở về trang Đăng nhập), Tìm kiếm (tìm kiếm lịch trực nhật), Reload (tải lại bảng hiển thị danh sách lịch trực nhật). Trang chứa các TextBox để nhập thông tin tìm kiếm lịch trực nhật.

Dựa vào các phân tích đặc tả yêu cầu phần mềm và nội dung thiết kế, sử dụng cơ chế xây dựng giao diện theo kéo thả, xây dựng giao diện người dùng bằng cách kéo thả các thành phần trực tiếp lên bản thiết kế mà không cần viết mã code, kết hợp xây dựng giao diện bằng cách xây dựng mã, sử dụng Table để hiển thị dữ liệu, ta xây dựng ứng dụng như sau.

B, Giao diện phần cứng

Phần mềm quản lý trực nhật và CSDL của trường học cần được cài đặt chung trên một máy chủ, được quản lý với hệ điều hành của máy tính. Trong trường hợp cài đặt thử nghiệm trên máy tính khác, cần có mô phỏng CSDL của nhà trường.

C, Giao diện phần mềm

- Giao diện quản lý trực nhật: chỉ cho phép giảng viên sử dụng để đọc và chỉnh sửa thông tin phân cao trực nhật cho sinh viên.

- Giao diện xem lịch trực nhật: chỉ cho phép sinh viên xem lịch trực nhật của mình đã được phân công.

##### 2.2.2.3.3 Yêu cầu hiệu suất

* Thời gian đáp ứng của hệ thống khi tìm kiếm và cập nhật thông tin phân công trực nhật không nên vượt quá một ngưỡng nhất định.
* Hệ thống  cần hỗ trợ đồng thời nhiều người dùng để đảm bảo rằng sinh viên và giáo viên có thể truy cập vào cùng một thời điểm mà không gặp trở ngại.

##### 2.2.2.3.4 Ràng buộc thiết kế

Ngôn ngữ được sử dụng hoàn toàn bằng Java, giao diện của hệ thống được thiết kế bằng JavaSwing và sử dụng Netbeans IDE làm môi trường duy nhất để phát triển hệ thống. Theo yêu cầu bài toán được giao, dữ liệu toàn bộ hệ thống sẽ được đọc ghi thông qua file Excel.

##### 2.2.2.3.5 Thuộc tính

Về khả năng mở rộng: Hệ thống cần có khả năng mở rộng để đáp ứng nhu cầu ngày càng tăng về số lượng sinh viên và giáo viên tham gia.

Về  tương tác người dùng: Hỗ trợ các chức năng đăng nhập, quản lý phân công và xem lịch trực nhật cá nhân một cách dễ dàng. Cung cấp các cập nhật để thông báo sinh viên về các thay đổi liên quan đến công việc trực nhật của mình.

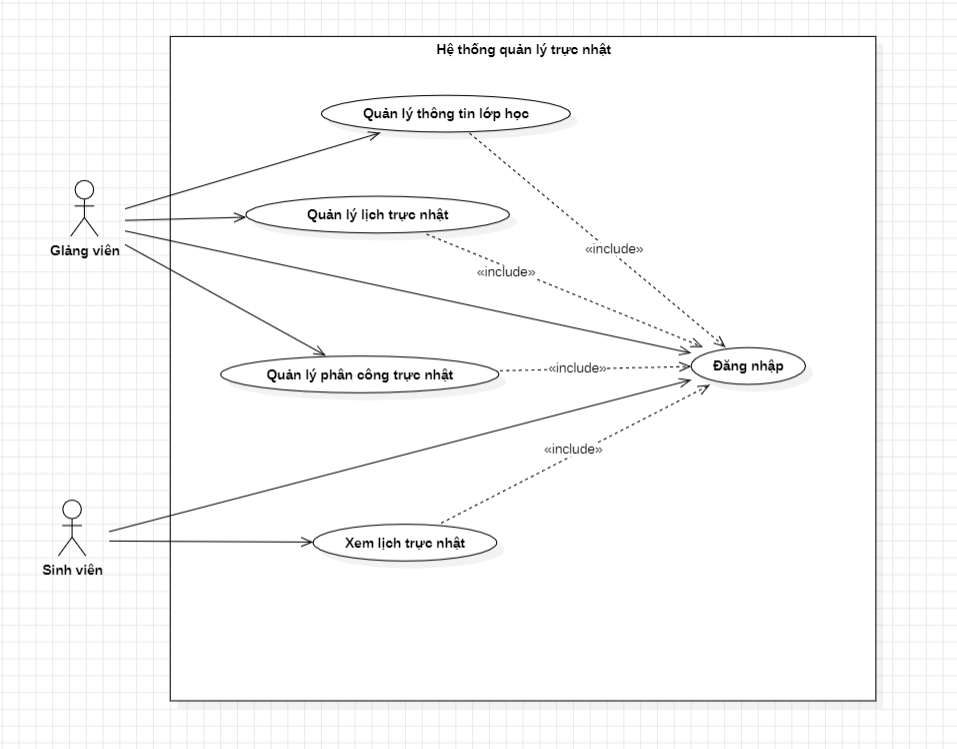
## 2.3 Phân tích hệ thống

### 2.3.1 Mô hình hóa chức năng hệ thống

|  |  |
| --- | --- |
| Tác nhân | Hoạt động |
|  |  |
| Sinh viên | 1. Đăng nhập  2. Xem lịch trực nhật |
| Giảng viên | 1. Đăng nhập  2. Quản lý thông tin các lớp dạy  3. Quản lý thông tin sinh viên  4. Quản lý phân công trực nhật |

Bảng 2.: Mô hình hóa chức năng hệ thống

Biểu đồ Use Case:



Hình 2.: Biểu đồ Use Case tổng quan

Mô tả Use Case:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Use Case | Mô tả | Chức năng | Ghi chú |
| DN01 | Đăng nhập | Chức năng đăng nhập | Cho phép người sử dụng đăng nhập vào tài khoản của mình | Giảng viên, Học sinh |
| QLTTL01 | Quản lý thông tin lớp | Quản lý thông tin các lớp giáo viên đó đứng lớp | Cho phép giảng viên theo dõi, thêm, sửa, xóa và sắp xếp danh sách lớp học mình đứng lớp. | Giảng viên |
| QLLTN01 | Quản lý lịch trực nhật | Giám sát lịch trực nhật theo tuần, theo buổi học và theo lớp học cho giảng viên | Cho phép giảng viên xem lịch giảng dạy và sinh viên nào trực nhật trong các buổi học đó | Giảng viên |
| QLPC01 | Quản lý phân công trực nhật | Phân chia lao động chi tiết cho sinh viên trong buổi học đó | Cho phép giảng viên phân công lao động chi tiết cho các sinh viên trong buổi học đó | Giảng viên |
| XLTN01 | Xem lịch trực nhật | Dành cho sinh viên theo dõi lịch trực nhật | Sinh viên theo dõi lịch trực nhật của mình theo tuần, theo buổi, theo lớp tại đây | Sinh viên |

Bảng 2.3: Mô tả Use Case

### 2.3.2 Mô hình hóa dữ liệu và giao diện hệ thống

**Chức năng Đăng nhập:**

Input: Tên tài khoản, mật khẩu

Output:

Thực thể:

* Account
* Mã sinh viên
* Mật khẩu



Hình 2.: Giao diện phác thảo trang Đăng nhập

**Chức năng Quản lý thông tin lớp:**

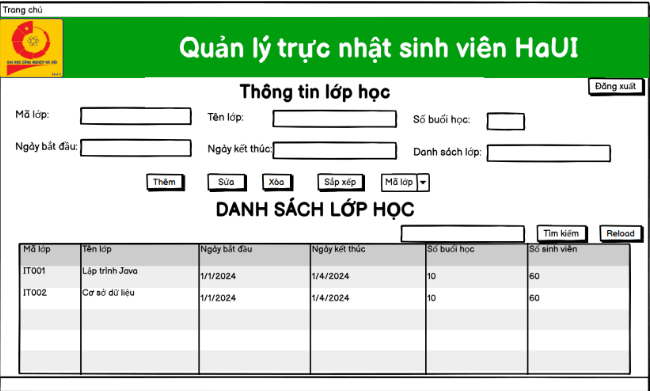
Input: Mã Lớp, Tên Lớp, Ngày Bắt Đầu, Ngày Kết Thúc, Số Buổi Học, Danh sách sinh viên (file excel).

Output: Danh sách các lớp trong hệ thống.

Thực thể:

* Lớp
* Mã lớp
* Tên lớp
* Ngày Bắt Đầu
* Ngày Kết Thúc
* Số Buổi Học
* List SinhVien

Phác thảo giao diện:



Hình 2.: Giao diện phác thảo trang Trang chủ

**Chức năng quản lý lịch trực nhật**

Input: Mã lớp, tên lớp, danh sách sinh viên, ngày trực nhật, lưu ý

Output: Danh sách phân công sinh viên trực nhật theo từng buổi học trong lớp học

Thực thể:

* Sinh viên trực nhật
* Lớp
* List sinh viên
* Ngày trực nhật
* Buổi học
* Lưu ý

Phác thảo giao diện:

A screenshot of a computer

Description automatically generated

Hình 2.: Giao diện phác thảo trang Quản lý lịch trực nhật

**Chức năng quản lý phân công trực nhật**

Input: Mã lớp, tên lớp, danh sách sinh viên trực nhật

Output: Danh sách những sinh viên được phân công lao động của lớp học

Thực thể:

* Sinh viên
* Mã sinh viên
* Tên sinh viên
* List mã lớp

Phác thảo giao diện:

A screenshot of a computer

Description automatically generated

Hình 2.: Giao diện phác thảo trang Quản lý phân công trực nhật

**Xem lịch trực nhật (cường)**

Input: Mã lớp, tên lớp, danh sách sinh viên, ngày trực nhật, lưu ý

Output: Lịch trực nhật của sinh viên theo lớp học

Thực thể:

* Sinh viên trực nhật
* Lớp
* List sinh viên
* Ngày trực nhật
* Buổi học
* Lưu ý

Phác thảo giao diện:

A green and white document with black text

Description automatically generated

Hình 2.: Giao diện phác thảo trang Trang chủ trang sinh viên

## 2.4 Thực hiện bài toán

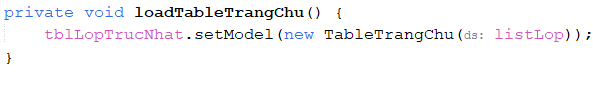
### 2.4.1 Phạm Trọng Nghĩa – Quản lý lớp học (Homepage)

Cơ chế kết hợp với Table để hiện thị dữ liệu:

+ Tạo lớp TableTrangChu kế thừa lớp AbstractTableModel:



+ Sử dụng phương thức setModel() để gán đối tượng mô hình dữ liệu AbstractTableModel cho bảng tblLopTrucNhat:



+ Hướng đối tương:

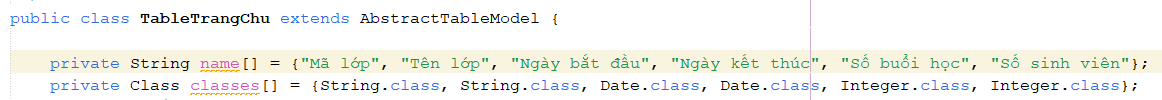
* Tính đóng gói: Xây dựng class SinhVien và Lop với các thuộc tính là private, muốn thao tác với đối tượng phải thông qua phương thức get() set() và phương thức khác của đối tượng.

A screenshot of a computer code

Description automatically generated

A computer code with black text

Description automatically generated



* Tính trừu tương: Xây dựng những thuộc tính cần thiết của Lop và SinhVien cho việc xây dựng chương trình.
* Tính đa hình trong hướng đối tượng:
* Lớp ghi đè các phương thức boolean, int equals(), getHashCode()

A screenshot of a computer program

Description automatically generated

* SinhVien ghi đè các phương thức boolean, int equals(), getHashCode.

A screenshot of a computer program

Description automatically generated

* TableTrangChu ghi đè các phương thức getRowCount(), int getValueAt(int rowIndex, int columnIndex), getColumnCount(), getColumnClass(int columnIndex):

A screenshot of a computer program

Description automatically generated

+ Bắt lỗi và gom rác:

* Cơ chế bẫy lỗi: Kiểm tra True/ False và ném ra ngoại lệ nếu có bằng cách sử dụng throws Exception.

A computer code with text

Description automatically generated

* Sau đó ta dùng try/catch để xử lý ngoại lệ, không có tạo lớp xử lý lỗi của riêng nhóm. Nếu có ngoại lệ từ thông tin nhập vào thì sẽ được thông báo ra màn hình.

A screenshot of a computer

Description automatically generated

+ Tập hợp:

* Sử dụng tập hợp ArrayList để lưu trữ danh sách doanh nghiệp. Bởi ArrayList là một phần của Collection framework và được chứa trong gói java.util. ArrayList cung cấp cho người dùng một mảng cho phép lập trình viên lưu trữ và thao tác với một lượng lớn dữ liệu.

A close-up of a text

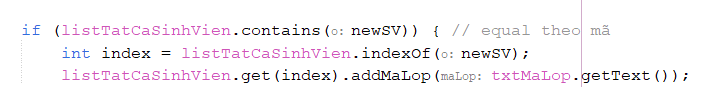
Description automatically generated

* Ở đây tập ArrayList được sử dụng do hỗ trợ nhiều phép toán như sau:
* Sử dụng phương thức add() để thêm mới đối tượng mới vào mảng:

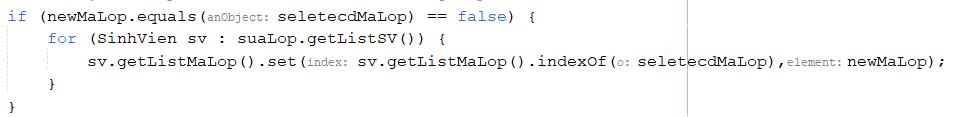
A close-up of a computer code

Description automatically generated

* Lấy ra các phần tử trong ArrayList với phương thức get()



* Lấy ra vị trí của một phần tử trong mảng với phương thức indexOf() và sửa một phần tử ở vị trí index với phép toán set()



* Phương thức removeIf(Predicate<? super E> filter để xóa một đối tượng ra khỏi danh sách dựa trên một điều kiện đã cho:

A black and blue text

Description automatically generated

* ArrayList còn hỗ trợ phương thức contains() để kiểm tra đối tượng có tồn tại trong danh sách không:

A computer screen shot of a computer code

Description automatically generated

+ Thao tác file:

Nhập danh sách lớp bằng file Excel với dữ liệu trong file Excel gồm 2 cột (mã sinh viên, họ tên):

A screenshot of a computer program

Description automatically generated

Hàm sẽ đọc từng dòng trong file excel sau đó tạo đối tượng sinh viên và thêm vào listTatCaSinhVien. Quá trình thêm vào sẽ đảm bảo ko có 2 sinh viên trùng mã.

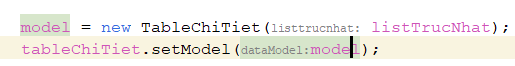
### 2.4.2 Nguyễn Duy Đạo – Quản lý lịch trực nhật

+ Cơ chế kết hợp Table để hiện thị dữ liệu:

* Tạo lớp TableChiTiet kế thừa lớp AbstractTableModel:



* Tạo một đối tượng listTrucNhat bằng cách kéo thả Table.
* Sử dụng phương thức setModel () để gán đối tượng mô hình dữ liệu AbstractTableModel cho bảng tableChiTiet:

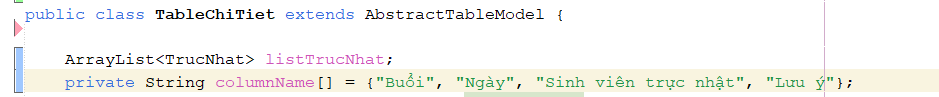


+ Hướng đối tương:

* Tính đóng gói: Xây dựng class TrucNhat và TableChiTiet:

A screen shot of a computer

Description automatically generated



* Lớp TrucNhat thể hiện tính trừu tượng: Trong thực tế , khi phân công trực nhật chỉ lấy những thuộc tính cân thiết và loại bỏ những thuộc tính không cần thiết của đối tượng này:

A close up of text

Description automatically generated

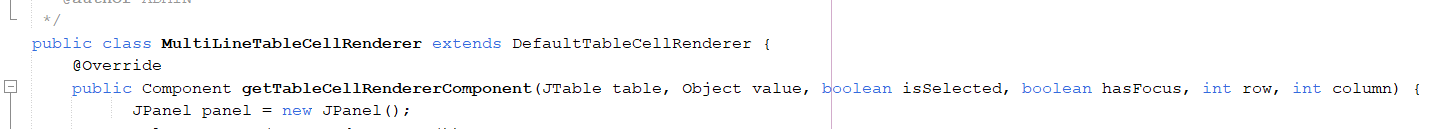
* Lớp TableChiTiet được kế thừa từ AbstractTableModel:



* Tính đa hình trong hướng đối tượng: Phương thức getTableCellRendererComponent là phương thức được ghi đè từ lớp DefaultTableCellRenderer. Trong đó, tùy chỉnh cách dữ liệu được hiển thị trong ô của bảng dựa trên kiểu của giá trị, đối tượng TrucNhat có 2 cách truyền thông tin không đối và có đối, ghi đè từ lớp cha:

A close-up of a computer code

Description automatically generated



+ Bắt lỗi và gom rác:

* Cơ chế bẫy lỗi: Kiểm tra các trường nhập ngày và ném ra ngoại lệ nếu có bằng cách sử dụng throws Exception:

A close up of text

Description automatically generated

* Sau đó ta dùng try/catch để xử lý ngoại lệ, không có tạo lớp xử lý lỗi của riêng nhóm. Nếu có ngoại lệ từ thông tin nhập vào ngày thì sẽ được thông báo ra màn hình.

A white screen with text

Description automatically generated with medium confidence

+ Tập hợp:

* Sử dụng tập hợp ArrayList để lưu trữ danh sách buổi trực nhật. Bởi ArrayList là một phần của Collection framework và được chứa trong gói java.util. ArrayList cung cấp cho người dùng một mảng động (kích thước thay đổi cho phép lập trình viên lưu trữ và thao tác với một lượng lớn dữ liệu:

A white background with black and pink text

Description automatically generated

* Ở đây tập ArrayList được sử dụng do hỗ trợ nhiều phép toán như sau:
* Sử dụng phương thức add() để thêm mới một đối tượng vào mảng:

A screenshot of a computer

Description automatically generated

* Trích xuất một phần của danh sách và trả về một đối tượng ArrayList mới, chứa các phần tử thuộc về khoảng chỉ định của danh sách gốc. với phương thức subList(int fromIndex, int toIndex):

A close-up of a white background

Description automatically generated

* Lấy ra các phần tử trong ArrayList với phương thức get (int index):

A close-up of a computer code

Description automatically generated

* Phương thức remove(int index) filter để xóa một đối tượng ra khỏi danh sách dựa trên một điều kiện đã cho:

A computer code with black text

Description automatically generated with medium confidence

* ArrayList còn hỗ trợ phương thức contains () để kiểm tra đối tượng có tồn tại trong danh sách hay không:

A close-up of a computer code

Description automatically generated

+ Tháo tác file: sử dụng lớp DataSingleTon để lưu trữ ,truyền , chia sẻ dữ liệu giữa các trang trên toàn bộ ứng dụng mà không cần tạo nhiều thể hiện hoặc ghi file để lưu trữ:

A screenshot of a computer program

Description automatically generated

### 2.4.3 Nguyễn Trung Đạt, Nguyễn Thị Bích Ngọc – Quản lý phân công trực nhật

+ Cơ chế kết hợp với table hiện thị dữ liệu:

* Tạo lớp TableSinhVienTN kế thừa lớp AbstractTableModel:

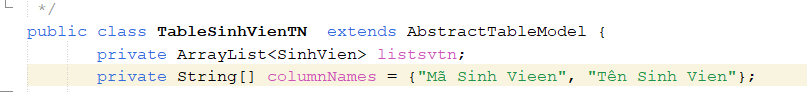


* Tạo một đối tượng bằng cách kéo thả Table
* Sử dụng phương thức setModel () để gán đối tượng mô hình dữ liệu AbstractTableModel cho bảng TableSinhVienTN:



+ Hướng đối tượng:

* Tính đóng gói: Xây dựng class TableSinhVienTN

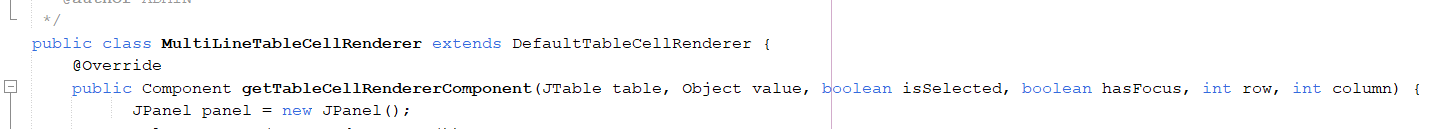


* Tính trừu tượng:  chỉ lấy những thuộc tính cần thiết cho sinh viên trực nhật trong một buổi và loại bỏ những thuộc tính không cần thiết trong thực tế.
* Lớp TableSinhVienTN được kế thừa từ AbstractTableModel:

A screenshot of a computer program

Description automatically generated

* Tính đa hình trong hướng đối tượng: Phương thức getTableCellRendererComponent là một phương thức được ghi đè từ lớp cha DefaultTableCellRenderer. Trong đó, tùy chỉnh cách dữ liệu được hiển thị trong ô của bảng dựa trên kiểu của giá trị:



+ Bắt lỗi và gom rác:

* Cơ chế bẫy lỗi: Kiểm tra sinh viên tồn tại trong buổi đó chưa và ném ra ngoại lệ nếu có bằng cách sử dụng throws Exception:

A screen shot of a computer code

Description automatically generated

* Sau đó ta dùng try/catch để xử lý ngoại lệ, không có tạo lớp xử lý lỗi của riêng nhóm. Nếu có ngoại lệ từ thêm một sinh viên trực nhật trong một buổi thì sẽ được thông báo ra màn hình:

A screenshot of a computer code

Description automatically generated

+ Tập hợp:

* Sử dụng tập hợp ArrayList để lưu trữ danh sách buổi trực nhật. Bởi ArrayList là một phần của Collection framework và được chứa trong gói java.util. ArrayList cung cấp cho người dùng một mảng động (kích thước thay đổi cho phép lập trình viên lưu trữ và thao tác với một lượng lớn dữ liệu:



A close up of a text

Description automatically generated

* Ở đây tập ArrayList được sử dụng do hỗ trợ nhiều phép toán như sau:
* Sử dụng phương thức add() để thêm mới một đối tượng vào mảng:

A close-up of a text

Description automatically generated

* Trích xuất một phần của danh sách và trả về một đối tượng ArrayList mới, chứa các phần tử thuộc về khoảng chỉ định của danh sách gốc. với phương thức subList(int fromIndex, int toIndex):

A close-up of a white background

Description automatically generated

* Lấy ra các phần tử trong ArrayList với phương thức get(int index):

A close up of text

Description automatically generated

* Phương thức remove(int index) filter để xóa một đối tượng ra khỏi danh sách dựa trên một điều kiện đã cho:

A close-up of a computer code

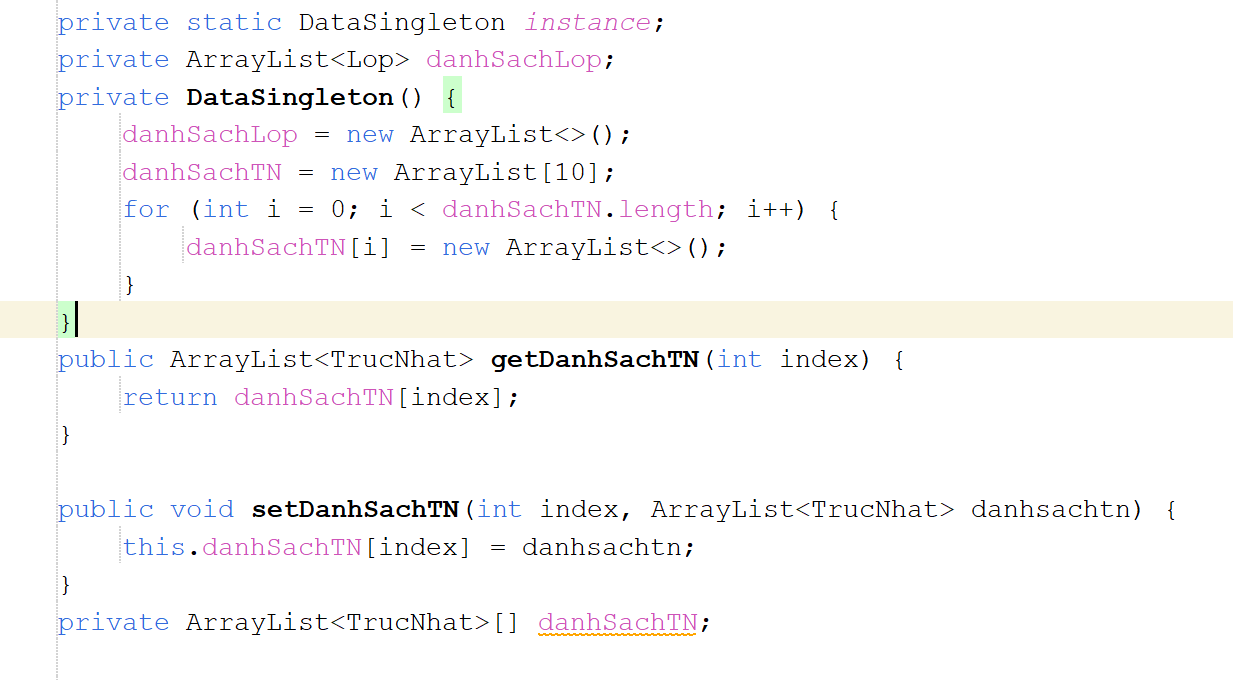
Description automatically generated

* ArrayList còn hỗ trợ phương thức contains () để kiểm tra đối tượng có tồn tại trong danh sách hay không:

A close up of a computer screen

Description automatically generated

+ **Thao tác File:** sử dụng lớp DataSingleTon để lưu trữ ,truyền , chia sẻ dữ liệu giữa các trang trên toàn bộ ứng dụng mà không cần tạo nhiều thể hiện hoặc ghi file để lưu trữ:



### 2.4.4 Hoàng Kim Cường – Xem lịch trực nhật

+ Cơ chế kết hợp với Table để hiện thị dữ liệu:

* Tạo lớp TableTrangChu1 kế thừa lớp AbstractTableModel:



* Sử dụng phương thức setModel () để gán đối tượng mô hình dữ liệu AbstractTableModel cho bảng tblLopTrucNhat:



+ Hướng đối tượng: Xây dựng class SinhVienLop

* Tính đóng gói: các thuộc tính là private, muốn thao tác với đối tược phải thông qua phương thức get() set():

A screenshot of a computer program

Description automatically generated

* Tính trừu tượng: Xây dựng những thuộc tính cần thiết cho việc xây dựng chương trình
* Tính đa hình trong hướng đối tượng: ghi đề phương thức equals()

A computer code with text

Description automatically generated

+ Bắt lỗi và gom rác:

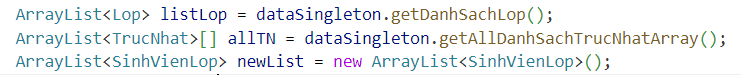
* Cơ chế bẫy lỗi: ta dùng try/catch để xử lý ngoại lệ, không có tạo lớp xử lý lỗi của riêng nhóm. Nếu có ngoại lệ từ thông tin nhập vào thì sẽ được thông báo ra màn hình:

A white screen with text

Description automatically generated

+ Tập hợp:

* Sử dụng tập hợp ArrayList để lưu trữ danh sách doanh nghiệp. Bởi ArrayList là một phần của Collection framework và được chứa trong gói java.util. ArrayList cung cấp cho người dùng một mảng động (kích thước thay đổi cho phép lập trình viên lưu trữ và thao tác với một lượng lớn dữ liệu





* Ở đây tập ArrayList được sử dụng do hỗ trợ nhiều phép toán như sau:
* Sử dụng phương thức add() để thêm mới một đối tượng vào mảng:

A close up of text

Description automatically generated

A close-up of a computer screen

Description automatically generated

* ArrayList còn hỗ trợ phương thức contains () để kiểm tra đối tượng có tồn tại trong danh sách hay không:

A screenshot of a computer code

Description automatically generated

A screenshot of a computer code

Description automatically generated

### 2.4.5 Hoàng Kim Cường - Đăng nhập

+ Hướng đối tượng:

* Tính đóng gói: Xây dựng class Account với các thuộc tính là private, muốn thao tác với đối tượng phải thông qua phương thức get() set() và phương thức khác của đối tượng:

A screenshot of a computer

Description automatically generated

* Tính trừu tượng: Xây dựng những thuộc tính cần thiết của Lop và SinhVien cho việc xây dựng chương trình.
* Tính đa hình trong hướng đối tượng: Account ghi đè các phương thức boolean, int equals(), getHashCode.

A screenshot of a computer code

Description automatically generated

+ Bắt lỗi và gom rác:

* Cơ chế bẫy lỗi: ta dùng try/catch để xử lý thông tin đăng nhập, nếu là admin thì chuyển đến trang admin nếu là người dùng chuyển đến trang người dùng.

A screen shot of a computer

Description automatically generated

+ Tập hợp:

* Sử dụng tập hợp ArrayList để lưu trữ danh sách tài khoản. Bởi ArrayList là một phần của Collection framework và được chứa trong gói java.util. ArrayList cung cấp cho người dùng một mảng động (kích thước thay đổi cho phép lập trình viên lưu trữ và thao tác với một lượng lớn dữ liệu.
* Ở đây tập ArrayList được sử dụng do hỗ trợ nhiều phép toán như sau:
* Sử dụng phương thức add() để thêm mới một đối tượng vào mảng:

A screenshot of a computer

Description automatically generated

* ArrayList còn hỗ trợ phương thức contains () để kiểm tra đối tượng có tồn tại trong danh sách hay không:

A computer screen shot of a computer code

Description automatically generated

# PHẦN 3: KẾT LUẬN VÀ BÀI HỌC KINH NGHIỆM

## 3.1 Nội dung đã thực hiện

Thông qua đề tài lần này các thành viên trong nhóm đã học được nhiều điều. Được ôn tập và rèn luyện các kỹ năng của môn lập trình Java để xử lý giao diện. Các kỹ năng mềm như làm việc nhóm, tổng hợp thông tin.

Qua quá trình thực hiện bài tập lớn, nhóm chúng em đã tích lũy được rất nhiều kiến thức trong học tập, rèn luyện các kỹ năng của môn lập trình Java để xử lý giao diện cũng như cải thiện các kỹ năng mềm cần thiết: kỹ năng làm việc nhóm, kỹ năng lãnh đạo, kỹ năng quản lý thời gian, kỹ năng thuyết trình và phản biện.

Đề tài đã xác định được các tác nhân, chức năng và đưa ra các quy trình tương ứng nhằm xây dựng một ứng dụng hiệu quả để quản lý và phân công lịch làm việc cũng như là giám sát công việc trực nhật cho giáo viên và sinh viên. Hệ Thống đáp ứng được những nhu cầu cơ bản:

Đăng nhập

* Xem lịch trực nhật
* Quản lý thông tin các lớp dạy
* Quản lý phân công trực nhật cho sinh viên
* Quản lý lịch trực nhật

Do thời gian có hạn và kiến thức còn hạn chế nên số mục còn sơ sài, chưa cụ thể và còn nhiều thiếu sót. Nhóm chúng em rất mong nhận được những đánh giá và góp ý từ thầy cô để giúp đề tài được hoàn thiện hơn.

## 3.2 Hướng phát triển

Đề tài mang tính thực tế cao, tạo kinh nghiệm cho các thành viên trong nhóm nâng cao kỹ năng của bản thân, tuy nhiên, một vài chức năng chưa hoàn thiện trọn vẹn.

Một số các chức năng sẽ phát triển:

* Nâng cấp hệ thống bảo mật thông tin người dùng và admin
* Kết nối với cơ sở dữ liệu để có thể mở rộng quy mô của dự án.
* Phát triển ứng dụng di động mang lại sự tiện lợi và linh hoạt cho người sử dụng.
* Nâng cao khả năng làm việc với các file dữ liệu.

# TÀI LIỆU THAM KHẢO

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. The Java Programming Language; Author: K. Arnold, J. Gosling; Published: Addison-Wesley, 1996, ISBN 0-201-63455-4.
3. Lập trình hướng đối tượng với Java; Đoàn Văn Ban; NXB Khoa học và Kỹ thuật, Hà Nội 2006 (Tái bản).
4. Lập trình Java nâng cao, Đoàn Văn Ban, NXB Khoa học và Kỹ thuật, Hà Nội 2006.
5. https://www.javatpoint.com/java-tutorial