

BÁO CÁO ĐỒ ÁN
HỌC PHẦN: Lập Trình Ứng Dụng Với Java***Chủ Đề: Website Quản Lý Ngân Hàng***

Giảng viên hướng dẫn: Th.s Nguyễn Hữu Trung

Sinh viên thực hiện:

MSSV:

Lớp:

- | | | |
|--------------------------------|------------------|----------|
| 1. Lâm Đức Huy | 2011062146 | 20DTHB1. |
| 2. Trần Đình Thái Hoà | 2011144071 | 20DTHB1. |
| 3. Nguyễn Phạm Thành Nam | 2011064708 | 20DTHB2. |
| 4. Trần Vũ Ngọc Hiếu | 2011063175 | 20DTHB1. |

TP. Hồ Chí Minh, 2023

LỜI CẢM ƠN

Chúng em xin gửi lời cảm ơn chân thành và sự tri ân sâu sắc đối với các thầy cô của trường Đại học Công Nghệ TP.HCM, đặc biệt là các thầy cô khoa Công Nghệ Thông Tin của trường đã tận tình giảng dạy và trang bị cho em những kiến thức nền tảng trong suốt quá trình học tập tại khoa để có thêm nhiều kinh nghiệm cho tương lai. Và em cũng xin chân thành cảm ơn thầy Nguyễn Hữu Trung đã nhiệt tình hướng dẫn hướng dẫn chỉ bảo em trong quá trình thực hiện đề tài. Trong quá trình học tập, cũng như là trong quá trình làm bài báo cáo đồ án, nhóm em khó tránh khỏi sai sót, rất mong Thầy bỏ qua. Đồng thời do trình độ lý luận cũng như kinh nghiệm thực tiễn còn hạn chế nên bài báo cáo không thể tránh khỏi những thiếu sót, em rất mong nhận được ý kiến đóng góp từ Thầy, để em học thêm được nhiều kinh nghiệm và sẽ hoàn thành tốt hơn ở bài báo cáo môn học sắp tới.

Chúng em xin chân thành cảm ơn!

LỜI MỞ ĐẦU

Ngành ngân hàng đóng vai trò quan trọng trong sự phát triển và ổn định kinh tế của một quốc gia. Với sự phát triển của công nghệ thông tin, việc áp dụng các giải pháp công nghệ vào quản lý ngân hàng đã trở thành một xu hướng không thể thiếu. Trong đề án này, chúng em đã tiến hành phát triển một web quản lý ngân hàng dựa trên kiến trúc Layered Architecture.

Chúng em đã sử dụng các công nghệ và công cụ như Java 8, Spring MVC, Spring Boot, Thymeleaf, AJAX, Spring Security, Poi-Ooxml và Spring Mail để xây dựng ứng dụng này đồng thời đặt mục tiêu tạo ra một ứng dụng hiệu quả và an toàn, mà còn đề cao tính ứng dụng thực tế và sự tiện lợi cho người dùng. Chúng tôi tin rằng ứng dụng web quản lý ngân hàng sẽ cung cấp một công cụ mạnh mẽ cho các tổ chức ngân hàng để nâng cao hiệu suất quản lý và cung cấp trải nghiệm tốt hơn cho khách hàng.

MỤC LỤC

BẢNG PHÂN CÔNG NHIỆM VỤ	Error! Bookmark not defined.
LỜI MỞ ĐẦU	Error! Bookmark not defined.
LỜI CẢM ƠN	Error! Bookmark not defined.
DANH MỤC HÌNH ẢNH	5
DANH MỤC BẢNG	6
MỤC LỤC	4
CHƯƠNG 1: GIỚI THIỆU TỔNG QUÁT & KHẢO SÁT	Error! Bookmark not defined.
1.1. Giới thiệu	Error! Bookmark not defined.
1.2. Đặt vấn đề	Error! Bookmark not defined.
1.3. Xác định yêu cầu	Error! Bookmark not defined.
1.4. Cấu trúc	Error! Bookmark not defined.
CHƯƠNG 2: CƠ CỞ LÝ THUYẾT	Error! Bookmark not defined.
2.1. Công cụ và ngôn ngữ lập trình sử dụng.....	Error! Bookmark not defined.
2.2. Thiết kế hệ thống.....	Error! Bookmark not defined.
CHƯƠNG 3: MÔ HÌNH HÓA NGHỆ VỤ	Error! Bookmark not defined.
3.1. Thiết kế cơ sở dữ liệu (Database)	Error! Bookmark not defined.
3.2. Thiết kế bảng cơ sở dữ liệu	Error! Bookmark not defined.
2.3. Thiết kế Usecase chi tiết	Error! Bookmark not defined.
3.3. Thiết kế mô hình Sequence	Error! Bookmark not defined.
CHƯƠNG 3: KẾT QUẢ THỬ NGHIỆM	Error! Bookmark not defined.
6.1. Phương pháp kiểm thử.....	Error! Bookmark not defined.
6.2. Kiểm thử	Error! Bookmark not defined.
CHƯƠNG 4: GIAO DIỆN PHẦN MỀM	Error! Bookmark not defined.

5.1. Giao diện quản trị.....	Error! Bookmark not defined.
5.2. Giao diện người dùng.....	Error! Bookmark not defined.
CHƯƠNG 5: KẾT LUẬN VÀ KIẾN NGHỊ.....	Error! Bookmark not defined.
7.1. Kết luận	Error! Bookmark not defined.
7.2. Kiến nghị	Error! Bookmark not defined.
Tài liệu tham khảo.....	Error! Bookmark not defined.

DANH MỤC HÌNH ẢNH

Hình 1 Mô hình hóa biểu đồ quan hệ (class).....	Error! Bookmark not defined.
Hình 2 Mô hình hóa biểu đồ chức năng (use case)	Error! Bookmark not defined.
Hình 3 Use case đăng nhập	Error! Bookmark not defined.
Hình 4 Use case quản lý khách hàng.....	Error! Bookmark not defined.
Hình 5 Use case quản lý nhân viên	Error! Bookmark not defined.
Hình 6 Biểu đồ Activity hoạt động đăng nhập.....	Error! Bookmark not defined.
Hình 7 Biểu đồ activity hoạt động thêm nhân viên.....	Error! Bookmark not defined.
Hình 8 Biểu đồ activity hoạt động sửa nhân viên	Error! Bookmark not defined.
Hình 9 Biểu đồ activity hoạt động xóa nhân viên	Error! Bookmark not defined.
Hình 10 Biểu đồ Sequence hoạt động đăng nhập.....	Error! Bookmark not defined.
Hình 11 Biểu đồ Sequence hoạt động thêm nhân viên..	Error! Bookmark not defined.
Hình 12 Biểu đồ Sequence hoạt động sửa nhân viên	Error! Bookmark not defined.
Hình 13 Biểu đồ Sequence hoạt động xóa nhân viên....	Error! Bookmark not defined.
Hình 14 Sequence Quản lý xem thống kê	Error! Bookmark not defined.
Hình 15 Biểu đồ activity tạo giao dịch.....	Error! Bookmark not defined.
Hình 16 Biểu đồ activity tạo tài khoản gửi tiền.....	Error! Bookmark not defined.
Hình 17 Biểu đồ activity quản lý hạn tài khoản	Error! Bookmark not defined.
Hình 18 Biểu đồ activity xem lương	Error! Bookmark not defined.
Hình 19 Activity xem số liệu thống kê.....	Error! Bookmark not defined.
Hình 20 Trang chủ.....	Error! Bookmark not defined.
Hình 21 Giao diện đăng nhập.....	Error! Bookmark not defined.
Hình 22 Giao diện chính.....	Error! Bookmark not defined.

Hình 23 Giao diện quản lý nhân viên	Error! Bookmark not defined.
Hình 24 Giao diện quản lý khách hàng	Error! Bookmark not defined.
Hình 25 Giao diện quản lý lượt giao dịch	Error! Bookmark not defined.
Hình 26 Giao diện xem chi tiết lượt giao dịch	Error! Bookmark not defined.
Hình 27 Giao diện quản lý hạn tài khoản	Error! Bookmark not defined.
Hình 28 Giao diện số liệu thống kê	Error! Bookmark not defined.
Hình 29 Giao diện quản lý lương nhân viên	Error! Bookmark not defined.
Hình 30 Giao diện xem hồ sơ cá nhân.....	Error! Bookmark not defined.
Hình 31 Giao diện xem lương bản thân	Error! Bookmark not defined.

DANH MỤC BẢNG

Bảng 1. Bảng phân công nhiệm vụ	Error! Bookmark not defined.
Bảng 2. Bảng kiểm thử	Error! Bookmark not defined.

CHƯƠNG I: TỔNG QUAN

1.1. Giới thiệu đề tài

Ngày nay, thương mại điện tử đã trở thành một phần quan trọng trong hoạt động kinh doanh, cung cấp một phương tiện hiệu quả để mua, bán và trao đổi thông tin qua Internet. Việc tận dụng công nghệ và sử dụng website là một cách tiếp cận tiện lợi để công ty, doanh nghiệp giới thiệu sản phẩm, cung cấp thông tin và tạo cơ hội cho khách hàng lựa chọn. Trong ngành ngân hàng, việc xây dựng một hệ thống web quản lý ngân hàng đáp ứng các yêu cầu và nhu cầu của khách hàng là rất cần thiết đó là lý do cho sự ra đời của trang web **HuHuBank**.

1.2. Tóm tắt lý thuyết và nghiên cứu trước đây

Trước khi triển khai hệ thống web quản lý ngân hàng, cần có sự nắm vững về các lý thuyết và nghiên cứu đã được thực hiện trong lĩnh vực này. Các tài liệu tham khảo và công trình nghiên cứu liên quan đến quản lý ngân hàng và các hệ thống tương tự có thể được sử dụng để xác định cơ sở lý thuyết và khung tương đối cho đề tài sẽ được liệt kê trong mục tài liệu tham khảo.

Trong lĩnh vực quản lý ngân hàng, đã có nhiều nghiên cứu và ứng dụng công nghệ thông tin nhằm cải thiện quá trình giao dịch và quản lý ngân hàng. Các công trình nghiên cứu đã nghiên cứu về hệ thống quản lý tài khoản, chuyển tiền, thanh toán, quản lý vay nợ và nhiều khía cạnh khác của ngành ngân hàng. Nghiên cứu trước đây đã đề xuất các phương pháp và công nghệ để xây dựng hệ thống quản lý ngân hàng hiệu quả. Ví dụ, một số công trình đã áp dụng kiến trúc Layered Architecture để phân chia hệ thống thành các lớp logic riêng biệt, từ đó tăng tính tổ chức và dễ bảo trì của hệ thống. Các công nghệ và ngôn ngữ lập trình như Java 8, Spring MVC, Spring Boot, Thymeleaf, AJAX, Spring Security, Poi-Ooxml và Spring Mail đã được sử dụng rộng rãi trong việc phát triển các ứng dụng web và hệ thống quản lý ngân hàng.

Thông qua việc tổng hợp và phân tích các nghiên cứu trước đây, chúng ta sẽ có một cơ sở lý thuyết vững chắc và tầm nhìn tổng quan về lĩnh vực quản lý ngân hàng cùng với các công nghệ liên quan. Điều này sẽ đóng vai trò quan trọng trong việc xây dựng và triển khai thành công hệ thống web quản lý ngân hàng theo kiến trúc Layered Architecture sử dụng các công nghệ và công cụ tương ứng.

1.3. Nhiệm vụ đồ án

Nhiệm vụ của đồ án là xây dựng một hệ thống web quản lý ngân hàng hiện đại, đáp ứng các yêu cầu và nhu cầu của khách hàng. Đề tài được đề xuất nhằm giải quyết các vấn đề và cung cấp giải pháp cho việc quản lý ngân hàng thông qua một giao diện trực tuyến. Mục tiêu của đồ án là tạo ra một hệ thống ổn định, bảo mật và dễ sử dụng, cung cấp các chức năng quản lý ngân hàng cơ bản như quản lý tài khoản, chuyển tiền, thanh toán, v.v.

1.4. Cấu trúc đồ án

Đồ án sẽ được tổ chức thành các chương như sau:

Chương 1: Tổng quan

Phần 1.1: Giới thiệu đề tài

Phần 1.2: Tóm tắt lý thuyết và nghiên cứu trước đây

Phần 1.3: Nhiệm vụ đồ án

Phần 1.4: Cấu trúc đồ án

Chương 2: Cơ sở lý thuyết

Phần 2.1: Giới thiệu về kiến trúc Layered Architecture

Phần 2.1.1: Định nghĩa và mô tả kiến trúc Layered Architecture

Phần 2.1.2: Các lớp và chức năng trong kiến trúc Layered Architecture

Phần 2.1.3: Ưu điểm và hạn chế của kiến trúc Layered Architecture

Phần 2.2: Giới thiệu về công nghệ Java 8

Phần 2.3: Giới thiệu về Spring MVC, Spring Boot và Thymeleaf

Phần 2.4: Giới thiệu về AJAX

Phần 2.5: Giới thiệu về Spring Security

Phần 2.6: Giới thiệu về Poi-Ooxml

Phần 2.7: Giới thiệu về Spring Mail

Chương 3: Kết quả thực nghiệm

Phần 3.1: Mô hình hóa nghiệp vụ

Phần 3.2: Giao diện ứng dụng

Chương 4: Kết luận và kiến nghị

Phần 4.1: Kết luận chung

Phần 4.2: Đóng góp, đề xuất mới và kiến nghị

Phần 4.3: Tài liệu tham khảo

CHƯƠNG II: CƠ SỞ LÝ THUYẾT

2.1. Giới thiệu về kiến trúc Layered Architecture

Kiến trúc Layered Architecture là một mô hình phân lớp trong phát triển phần mềm, trong đó hệ thống được chia thành các lớp logic khác nhau. Mỗi lớp có nhiệm vụ cụ thể và tương tác với các lớp khác theo một quy tắc nhất định. Kiến trúc này giúp tách biệt các phần của hệ thống, làm cho mã nguồn dễ bảo trì, mở rộng và tái sử dụng.

Trong hệ thống web quản lý ngân hàng, kiến trúc Layered Architecture được áp dụng để tạo ra một cấu trúc rõ ràng và dễ quản lý. Các tầng lớp được xây dựng theo logic chức năng và có thể phát triển độc lập, giúp tăng tính linh hoạt và khả năng mở rộng của hệ thống. Kiến trúc này cũng cho phép phân chia công việc giữa các nhóm phát triển và tăng hiệu suất làm việc.

2.1.1. Định nghĩa và mô tả kiến trúc Layered Architecture

Kiến trúc Layered Architecture là một kiến trúc phần mềm phổ biến trong phát triển ứng dụng web. Nó giúp tổ chức và phân lớp các thành phần của hệ thống theo cách thức có cấu trúc và dễ bảo trì.

Layered Architecture được xây dựng dựa trên nguyên tắc chia tách và phân cấp logic của các thành phần hệ thống thành từng tầng (layer) riêng biệt. Mỗi tầng có trách nhiệm và chức năng riêng, và các tầng này tương tác với nhau thông qua giao diện đơn giản.

2.1.2. Các lớp và chức năng trong kiến trúc Layered Architecture

Mô hình Layered Architecture thường được phân thành các tầng chính sau:

Presentation Layer (Tầng trình bày): Đây là tầng giao diện người dùng, chịu trách nhiệm hiển thị thông tin và tương tác với người dùng. Nó cung cấp các giao diện đồ

họa và các thành phần tương tác như các form nhập liệu, nút bấm và menu. Trong ứng dụng web, tầng này thường sử dụng các công nghệ như HTML, CSS, JavaScript và Thymeleaf để hiển thị dữ liệu và tương tác với người dùng.

Business Layer (Tầng logic kinh doanh): Tầng này chứa các quy tắc và logic xử lý nghiệp vụ của hệ thống. Nó xử lý các yêu cầu từ tầng trình bày, thực hiện các hoạt động kinh doanh, và gửi yêu cầu tương tác với tầng dữ liệu. Trong kiến trúc Layered Architecture, tầng này thường được triển khai bằng các thành phần Java như Spring MVC hoặc Spring Boot, và nó có thể gọi các dịch vụ ngoài qua các API.

Data Layer (Tầng dữ liệu): Tầng này là nơi lưu trữ và truy xuất dữ liệu từ các nguồn dữ liệu như cơ sở dữ liệu, API ngoại vi, hoặc các tệp tin. Nó cung cấp các phương thức truy xuất dữ liệu và thực hiện các thao tác tương tác với dữ liệu. Trong ứng dụng web, tầng này thường được triển khai bằng các công nghệ như Hibernate, JPA hoặc JDBC để thao tác với cơ sở dữ liệu.

Các tầng trong kiến trúc Layered Architecture được tổ chức theo thứ tự từ trên xuống dưới, và mỗi tầng chỉ nắm bắt trách nhiệm của mình mà không can thiệp vào các tầng khác. Sự phân tách rõ ràng giữa các tầng giúp tăng tính rõ ràng, dễ bảo trì và phát triển của hệ thống.

2.1.3. Ưu điểm và hạn chế của kiến trúc Layered Architecture

Lợi ích của việc sử dụng kiến trúc Layered Architecture trong hệ thống web quản lý ngân hàng là:

Tính linh hoạt và dễ mở rộng: Các tầng riêng biệt cho phép dễ dàng thay thế hoặc mở rộng một phần của hệ thống mà không ảnh hưởng đến các tầng khác.

Tái sử dụng mã nguồn: Các tầng riêng biệt giúp tách biệt logic kinh doanh và giao diện người dùng, giúp tái sử dụng mã nguồn và giảm thiểu việc viết lại mã.

Dễ bảo trì: Các tầng riêng biệt và phân chia rõ ràng giúp dễ dàng xác định và sửa chữa lỗi trong hệ thống.

Phân tách trách nhiệm: Mỗi tầng có trách nhiệm cụ thể, giúp đơn giản hóa việc phân công công việc và quản lý dự án.

Tuy nhiên, việc áp dụng kiến trúc Layered Architecture cũng có những hạn chế và điểm yếu như:

Khả năng mở rộng hạn chế: Kiến trúc Layered Architecture có thể gặp khó khăn trong việc mở rộng khi số lượng tầng và sự phức tạp của hệ thống tăng lên.

Hiệu suất: Do việc tương tác giữa các tầng thông qua giao diện, có thể gây ra một số hiệu suất chậm khi cần truyền dữ liệu qua nhiều tầng.

2.2. Giới thiệu về công nghệ Java 8

Java 8 là một phiên bản quan trọng và mang tính đột phá của ngôn ngữ lập trình Java. Được phát hành vào năm 2014 bởi Oracle Corporation, Java 8 mang đến nhiều cải tiến và tính năng mới mạnh mẽ, giúp tăng cường hiệu suất và khả năng phát triển của ứng dụng.

Một số tính năng nổi bật của Java 8 bao gồm:

Lambda Expressions: Lambda expressions là một tính năng quan trọng trong Java 8, cho phép viết mã ngắn gọn và linh hoạt hơn. Nó giúp đơn giản hóa việc sử dụng các hàm ẩn danh và xử lý sự kiện. Lambda expressions giúp code trở nên ngắn gọn, dễ đọc và dễ hiểu hơn, đồng thời cung cấp khả năng xử lý các tác vụ song song một cách hiệu quả.

Stream API: Stream API là một phần quan trọng của Java 8, cung cấp một cách linh hoạt và hiệu quả để xử lý dữ liệu trong Java. Stream API cho phép thực hiện các phép biến đổi, lọc và tổng hợp trên dữ liệu trong một chuỗi liên tiếp các bước. Nó giúp giảm thiểu việc sử dụng vòng lặp truyền thống và tăng cường khả năng xử lý và hiệu suất của ứng dụng.

Default Methods: Default methods là một tính năng mới trong Java 8, cho phép viết các phương thức mặc định trong các giao diện. Trước đây, giao diện chỉ có thể chứa

các khai báo phương thức, nhưng không thể chứa mã thực thi. Với default methods, ta có thể thêm các phương thức đã được triển khai sẵn vào các giao diện mà không cần thay đổi các lớp đã tồn tại triển khai giao diện đó. Điều này giúp hỗ trợ việc mở rộng và nâng cấp các giao diện mà không làm hỏng các lớp đã tồn tại.

Date and Time API: Java 8 giới thiệu một API mới để làm việc với ngày và thời gian, gọi là Date and Time API. API này giải quyết nhiều vấn đề liên quan đến xử lý thời gian như định dạng, tính toán và so sánh. Nó giúp đơn giản hóa việc xử lý các tác vụ liên quan đến ngày và thời gian trong ứng dụng Java.

Parallel Streams: Java 8 cung cấp khả năng xử lý song song thông qua Parallel Streams. Parallel Streams cho phép ta chia tách các tác vụ xử lý dữ liệu thành các công việc nhỏ và thực hiện chúng song song trên nhiều luồng. Điều này giúp tăng tốc độ xử lý dữ liệu và cải thiện hiệu suất của ứng dụng.

2.3. Giới thiệu về Spring MVC, Spring Boot và Thymeleaf

Spring MVC (Model-View-Controller) là một framework phát triển ứng dụng web được xây dựng trên nền tảng Spring Framework. Nó cung cấp một cách tiếp cận theo mô hình MVC để xây dựng các ứng dụng web chất lượng cao và dễ bảo trì. Spring MVC tách biệt các phần khác nhau của ứng dụng, bao gồm mô hình dữ liệu (Model), giao diện người dùng (View) và xử lý logic (Controller).

Mô hình MVC trong Spring MVC cho phép phân chia rõ ràng và cấu trúc hóa ứng dụng web. Model đại diện cho dữ liệu và logic xử lý, View đại diện cho giao diện người dùng và Controller đảm nhận vai trò điều khiển và xử lý các yêu cầu từ người dùng. Spring MVC cung cấp các thành phần và tính năng như DispatcherServlet, Controllers, Resolvers, Validators, và ViewResolver để hỗ trợ việc xây dựng ứng dụng web linh hoạt và mạnh mẽ.

Spring Boot là một dự án trong hệ sinh thái Spring được phát triển nhằm đơn giản hóa việc xây dựng ứng dụng Java và giảm bớt công việc cấu hình. Nó tập trung vào việc cung cấp một cách tiếp cận "convention-over-configuration" (quy ước hơn là cấu hình) để phát triển ứng dụng nhanh chóng và dễ dàng.

Spring Boot tự động cấu hình một số thành phần cần thiết như Dependency Injection, Configuration và Auto-configuration. Nó cũng hỗ trợ việc nhúng các container như Tomcat hoặc Jetty để triển khai ứng dụng web một cách đơn giản. Một điểm đáng chú ý khác của Spring Boot là nó tích hợp sẵn Spring Actuator, cho phép quản lý và giám sát ứng dụng một cách dễ dàng.

Thymeleaf là một template engine cho phép tích hợp mã Java và HTML một cách mạnh mẽ. Nó được sử dụng trong các ứng dụng web để tạo ra giao diện người dùng động và linh hoạt. Thymeleaf cung cấp các tính năng như binding dữ liệu, lặp lại, điều kiện, và biểu thức trong các template.

Một điểm mạnh của Thymeleaf là nó cho phép hiển thị dữ liệu trực tiếp từ các đối tượng Java vào các template HTML mà không cần phải tạo ra các mã trung gian. Điều này giúp việc phát triển giao diện trở nên dễ dàng và hiệu quả. Thymeleaf cũng hỗ trợ việc tạo ra các form và xử lý sự kiện trên giao diện người dùng.

2.4. Giới thiệu về AJAX

AJAX (Asynchronous JavaScript and XML) là một kỹ thuật phát triển ứng dụng web cho phép trao đổi dữ liệu giữa trình duyệt và máy chủ một cách bất đồng bộ mà không cần tải lại toàn bộ trang. AJAX sử dụng JavaScript để gửi và nhận dữ liệu từ máy chủ và cập nhật phần giao diện người dùng một cách linh hoạt và mượt mà.

Trong kiến trúc Layered Architecture, AJAX thường được sử dụng để tạo ra các trang web động, nơi người dùng có thể tương tác với ứng dụng mà không cần tải lại trang. Thay vì gửi yêu cầu trang hoàn chỉnh, AJAX cho phép gửi các yêu cầu nhỏ hơn đến máy chủ và nhận kết quả trả về dưới dạng dữ liệu JSON hoặc XML. Sau đó, JavaScript sẽ sử dụng dữ liệu này để cập nhật phần giao diện người dùng mà không làm gián đoạn trải nghiệm người dùng. AJAX cung cấp một số lợi ích quan trọng. Đầu tiên, nó tạo ra trải nghiệm người dùng tốt hơn với việc cập nhật nhanh chóng và mượt mà của giao diện. Thứ hai, nó giảm tải cho máy chủ và mạng bằng cách giảm lượng dữ

liệu truyền đi. Thứ ba, nó cho phép tái sử dụng lại các thành phần giao diện và logic xử lý trên nhiều trang.

2.5. Giới thiệu về Spring Security

Spring Security là một framework mạnh mẽ và phổ biến trong việc xác thực và kiểm soát truy cập trong ứng dụng web. Nó cung cấp các tính năng bảo mật và quản lý danh tính để bảo vệ các tài nguyên và dữ liệu quan trọng trong hệ thống. Spring Security giúp xác thực người dùng bằng cách xác minh thông tin đăng nhập, như tên người dùng và mật khẩu. Nó hỗ trợ nhiều phương pháp xác thực, bao gồm xác thực cơ bản (basic authentication), xác thực dựa trên biểu mẫu (form-based authentication), xác thực qua dịch vụ bên ngoài (external service authentication) như OAuth, và nhiều phương pháp xác thực khác.

Ngoài việc xác thực, Spring Security cung cấp cơ chế kiểm soát truy cập linh hoạt. Bằng cách sử dụng các quy tắc và quyền (authorities), bạn có thể quản lý và kiểm soát quyền truy cập đối với các tài nguyên và chức năng trong hệ thống. Điều này giúp đảm bảo rằng chỉ những người dùng được phép có quyền truy cập vào các phần quan trọng của ứng dụng. Spring Security cũng tích hợp tốt với các công nghệ và tiêu chuẩn bảo mật phổ biến như OAuth, JWT (JSON Web Token), và LDAP (Lightweight Directory Access Protocol). Điều này cho phép bạn triển khai các phương thức bảo mật tiên tiến trong ứng dụng của mình.

2.6. Giới thiệu về Poi-Ooxml

POI-OOXML là một thư viện mã nguồn mở trong dự án Apache POI, được sử dụng để làm việc với tệp tin Office Open XML (OOXML). OOXML là định dạng tệp tin chuẩn được sử dụng bởi các ứng dụng văn phòng như Microsoft Excel, Word và PowerPoint. POI-OOXML cung cấp các công cụ và API để đọc, tạo và chỉnh sửa các tệp tin OOXML. Điều này cho phép bạn tương tác với các tệp tin Excel (.xlsx), Word (.docx) và PowerPoint (.pptx) một cách dễ dàng và linh hoạt. Với POI-OOXML, bạn có thể đọc dữ liệu từ các tệp tin Excel và Word và thực hiện các thao tác xử lý như đọc và ghi dữ liệu, định dạng văn bản, tạo bảng tính, thêm hình ảnh và biểu đồ, và nhiều hoạt động khác. Bạn cũng có thể tạo các tệp tin Excel và Word mới từ đầu hoặc chỉnh sửa

các tệp tin đã tồn tại. POI-OOXML cung cấp một API mạnh mẽ và linh hoạt, cho phép bạn tương tác với các yếu tố cấu trúc của tệp tin OOXML như sheets, rows, cells, paragraphs, styles, và nhiều yếu tố khác. Bạn có thể truy cập và thay đổi dữ liệu, thuộc tính, và định dạng của các yếu tố này để tạo ra các tệp tin OOXML tùy chỉnh và phức tạp.

POI-OOXML là một công cụ quan trọng trong phát triển ứng dụng quản lý ngân hàng vì nó cho phép bạn làm việc với các tệp tin Excel và Word, rất phổ biến trong lĩnh vực ngân hàng. Bạn có thể sử dụng POI-OOXML để tạo và tùy chỉnh báo cáo, biểu đồ, và các tài liệu văn bản khác trong hệ thống của mình.

2.7. Giới thiệu về Spring Mail

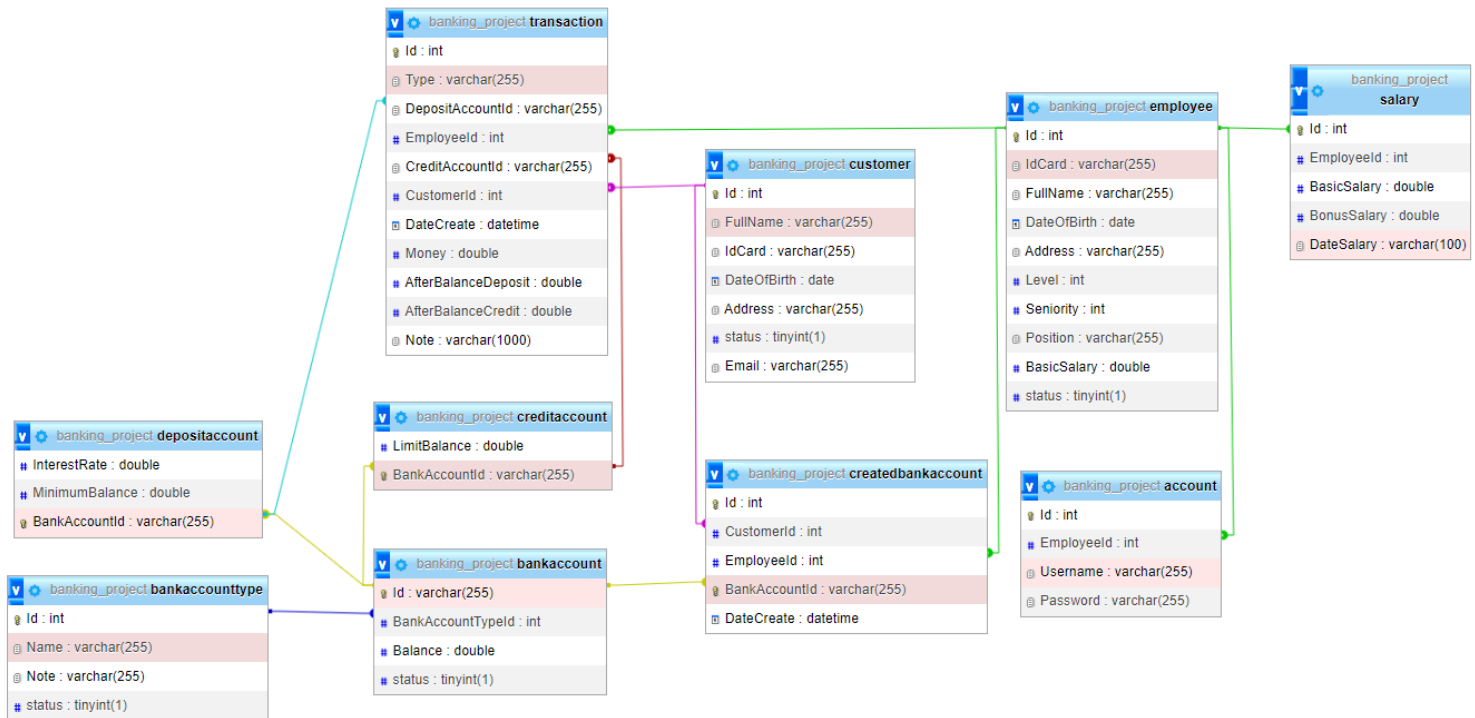
Spring Mail là một module trong Spring Framework cung cấp các tính năng và API để gửi và nhận email trong ứng dụng Java. Nó tương tác với các giao thức email như SMTP, POP3 và IMAP để thực hiện các hoạt động liên quan đến email. Spring Mail giúp đơn giản hóa việc gửi và nhận email trong ứng dụng. Nó cung cấp một cách tiếp cận linh hoạt và mạnh mẽ để tạo và gửi các email đơn giản hoặc phức tạp. Bạn có thể dễ dàng cấu hình các thông số như địa chỉ email nguồn, địa chỉ email đích, tiêu đề, nội dung và tệp đính kèm. Spring Mail cung cấp tích hợp với Spring Framework, cho phép bạn tận dụng các tính năng khác của Spring như dependency injection, quản lý transaction, và kiểm tra đơn vị. Bạn có thể sử dụng các bean và cấu hình Spring để quản lý các thông tin cấu hình của email như các địa chỉ email, cài đặt giao thức, và thông tin xác thực. Với Spring Mail, bạn có thể thực hiện các hoạt động như gửi email thông qua giao thức SMTP, nhận email qua giao thức POP3 hoặc IMAP, xóa email, đánh dấu email là đã đọc, và nhiều hoạt động khác. Nó cung cấp các lớp và API để tương tác với các thư mục, hộp thư và email trong một máy chủ email.

Spring Mail hỗ trợ cả việc gửi email đồng bộ và bất đồng bộ. Bạn có thể sử dụng nó để gửi email trực tiếp trong quá trình thực thi hoặc thông qua các tiến trình bất đồng bộ như message queue hoặc thread riêng. Sử dụng Spring Mail, bạn có thể tích hợp chức năng gửi và nhận email trong ứng dụng web quản lý ngân hàng của mình.

CHƯƠNG III: KẾT QUẢ THỰC NGHIỆM

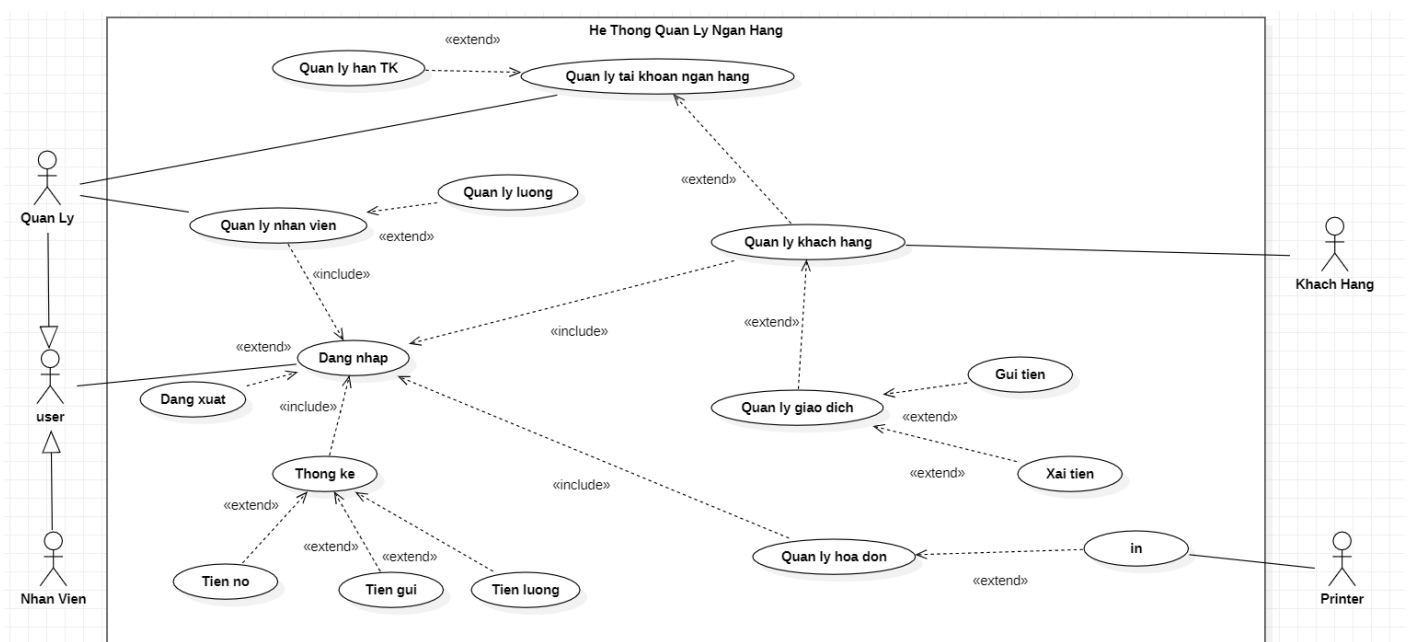
3.1. Mô hình hóa nghiệp vụ

3.1.1. Mô hình hóa biểu đồ quan hệ (class)



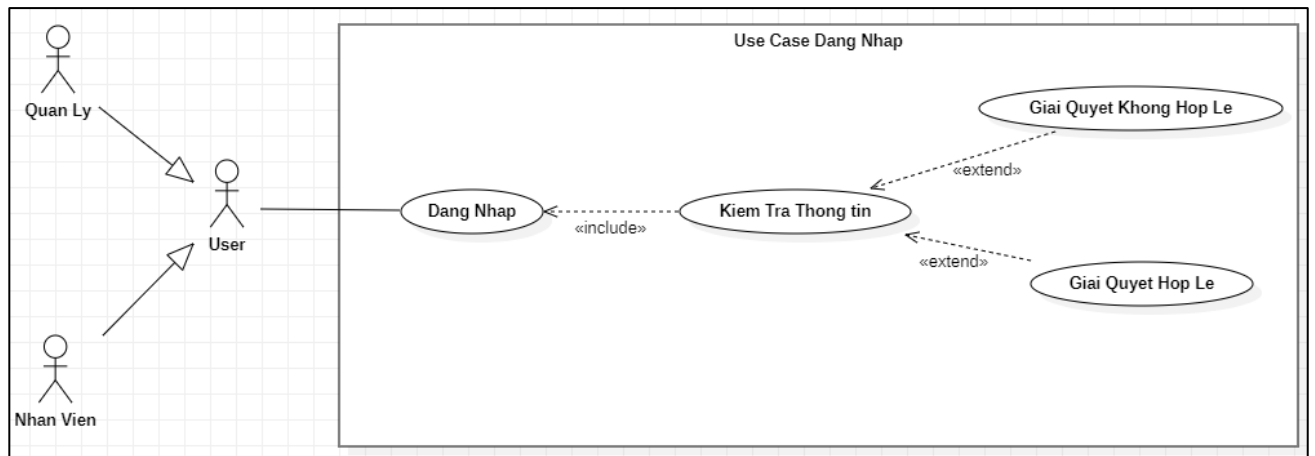
Class mô hình hóa biểu đồ quan hệ, thuộc tính

3.1.2. Mô hình hóa biểu đồ chức năng (use case)



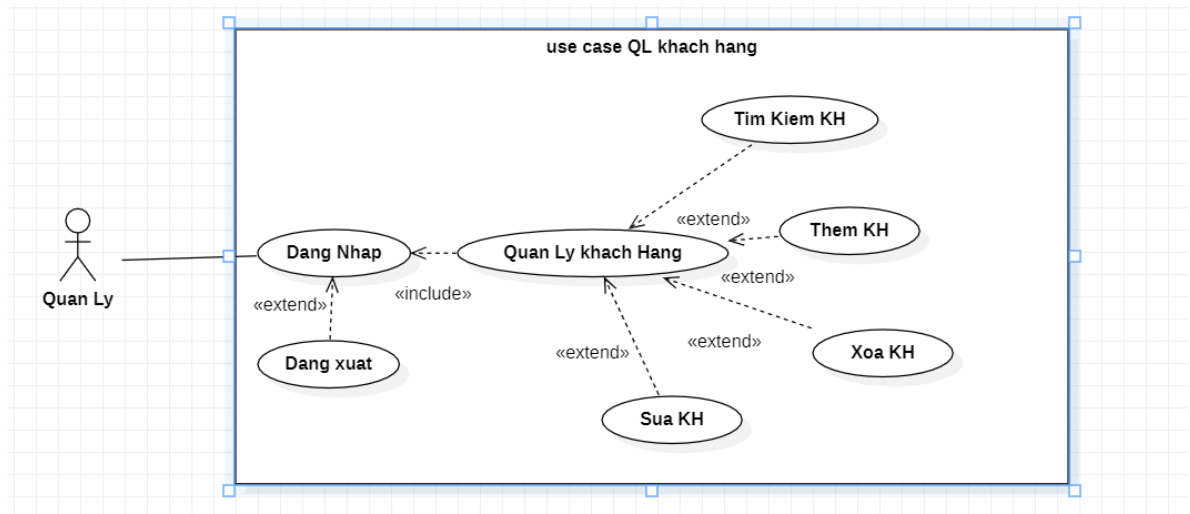
Use case tổng quát

3.1.3. Use case đăng nhập



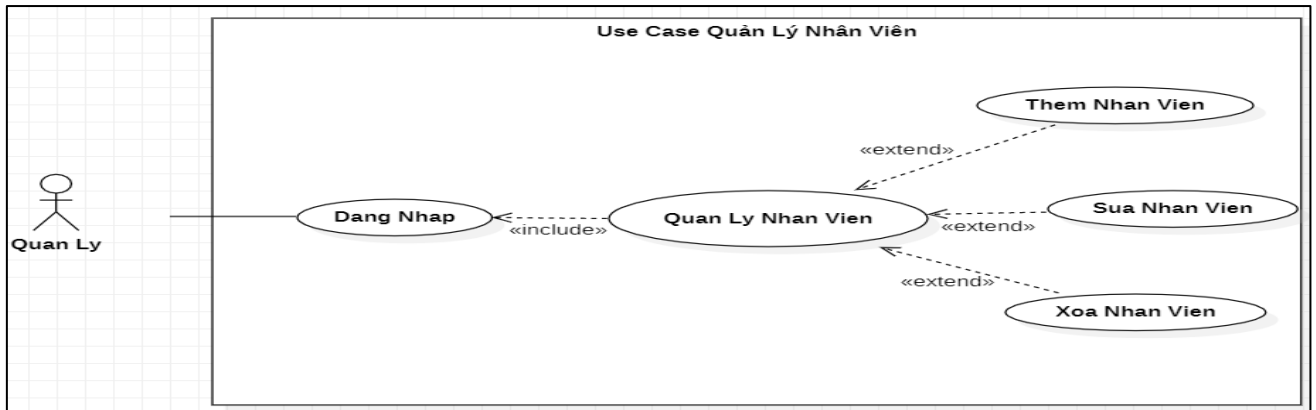
Use case đăng nhập

3.1.4. Use case quản lý khách hàng



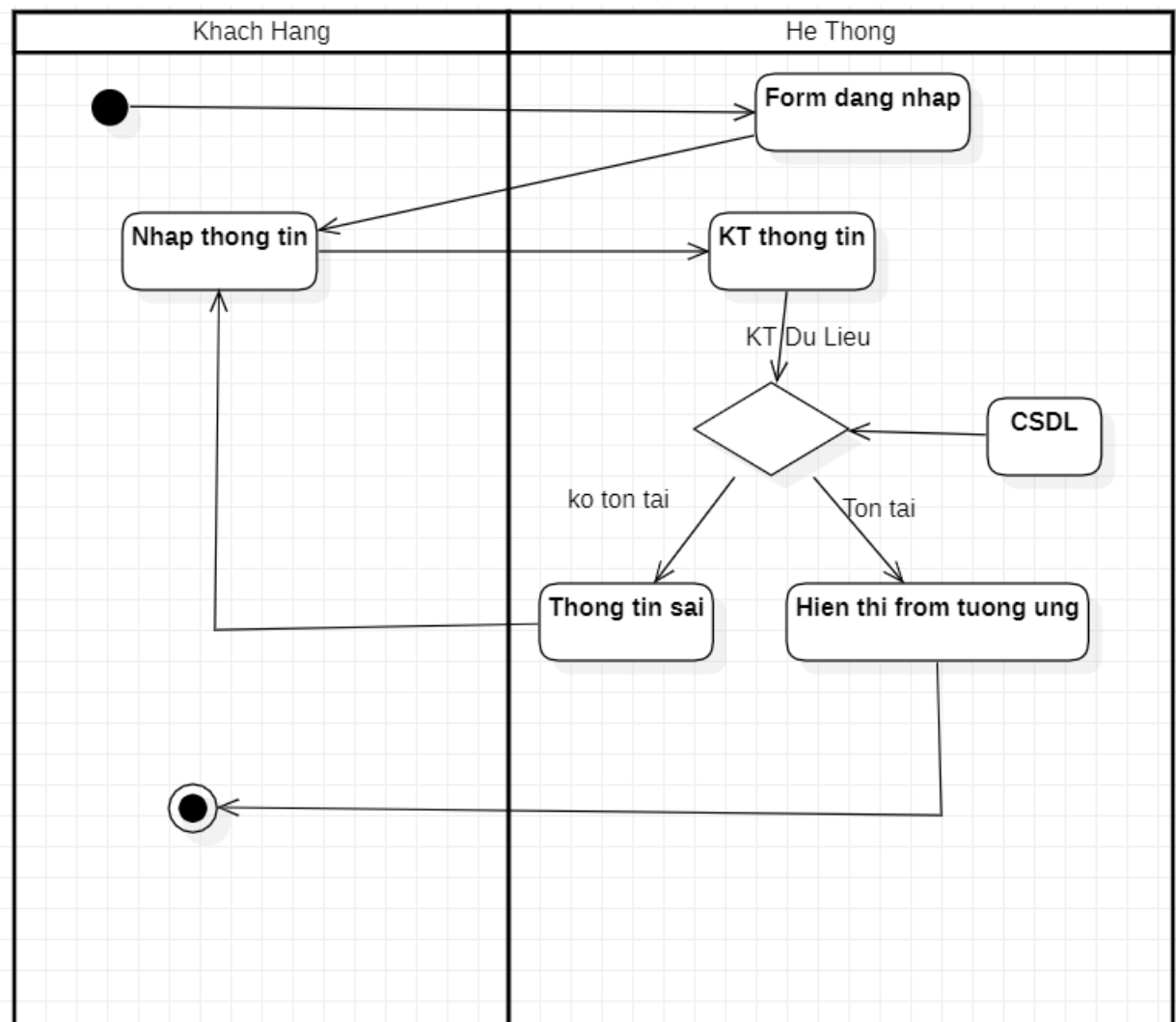
Use case quản lý khách hàng

3.1.5. Use case quản lý nhân viên

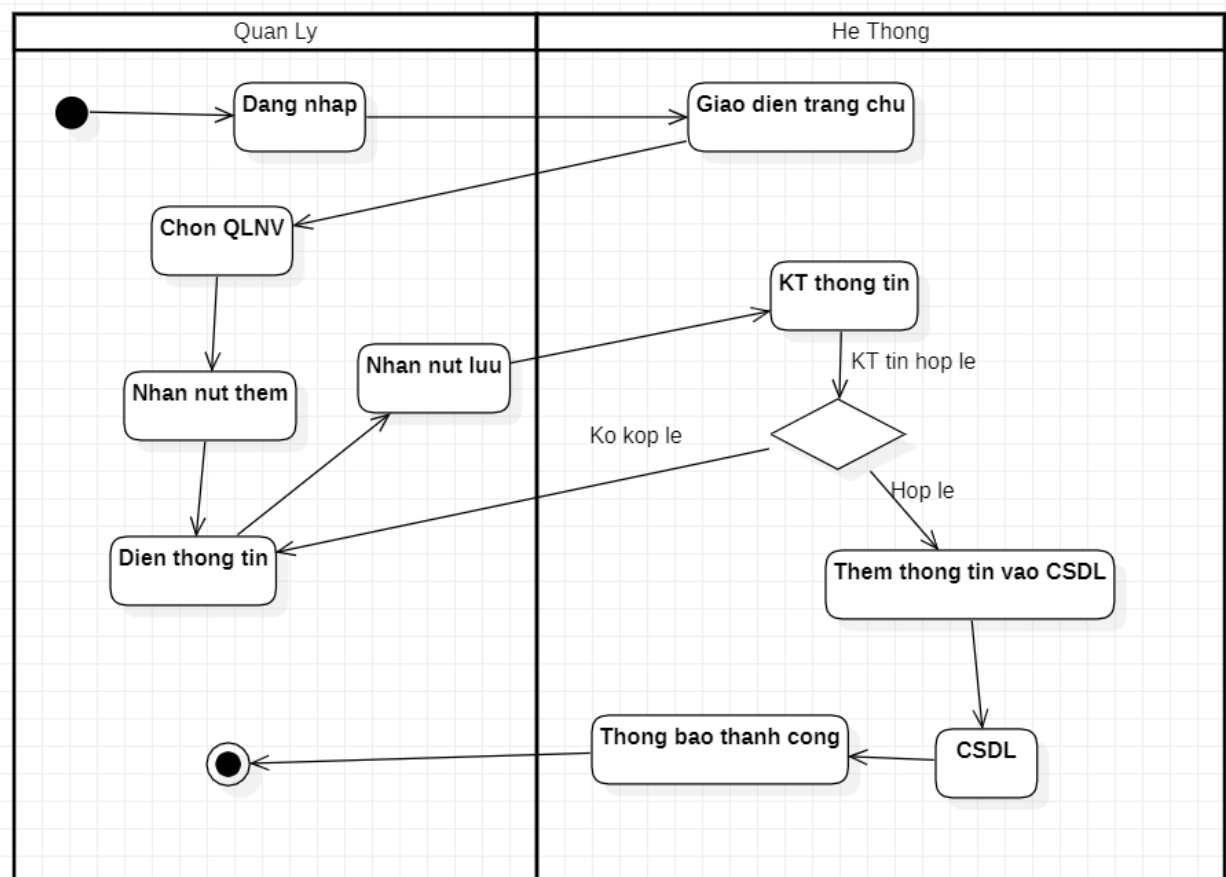


Use case quản lý nhân viên

3.1.6. Biểu đồ Activity hoạt động đăng nhập

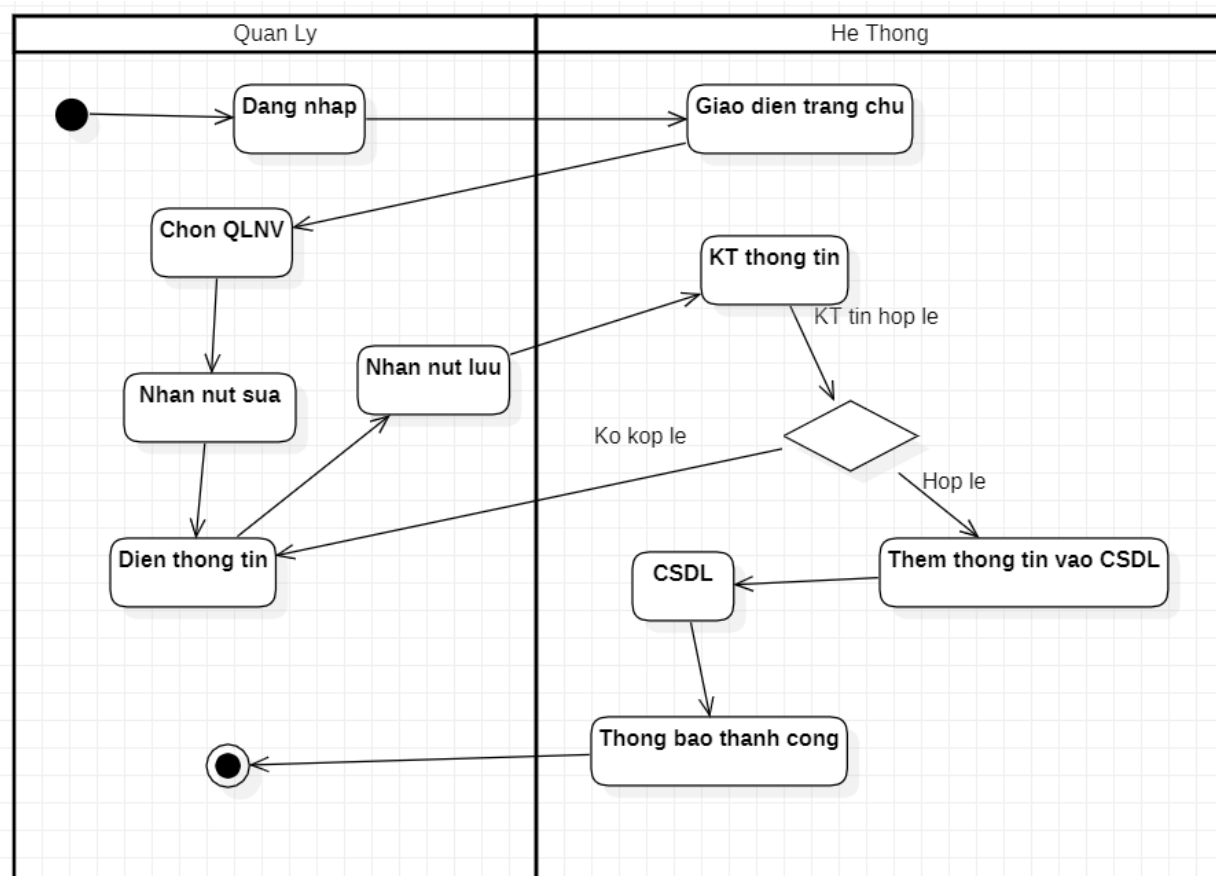


3.1.7. Biểu đồ activity hoạt động thêm nhân viên



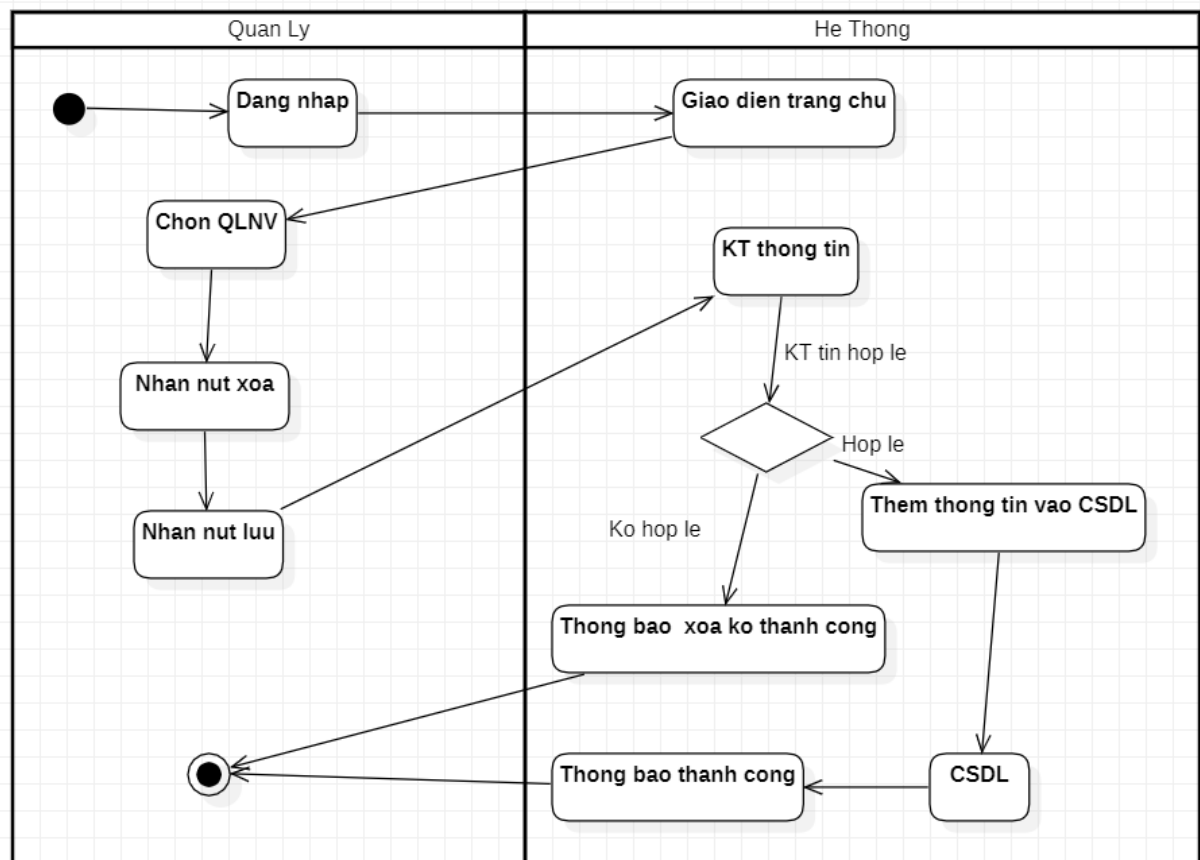
Activity hoạt động thêm nhân viên

3.1.8. Biểu đồ activity hoạt động sửa nhân viên



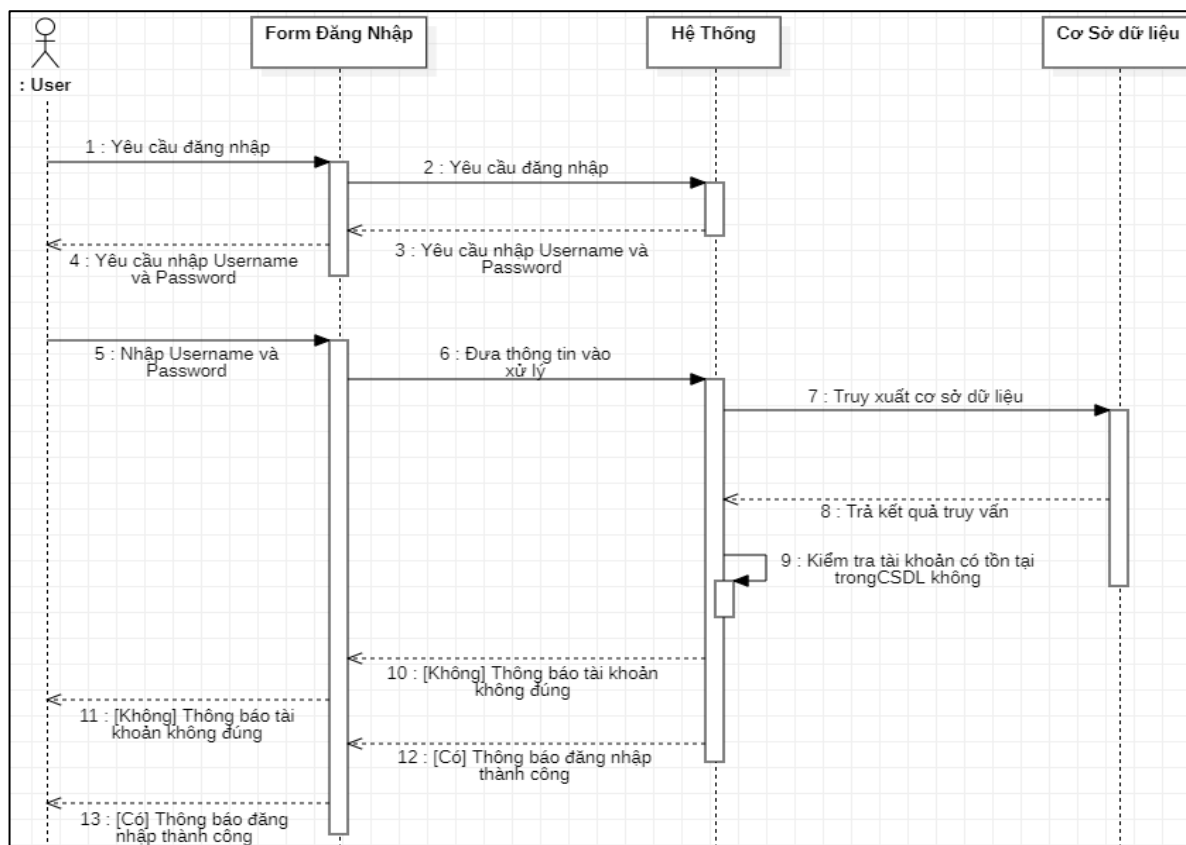
Activity hoạt động sửa nhân viên

3.1.9. Biểu đồ activity hoạt động xóa nhân viên



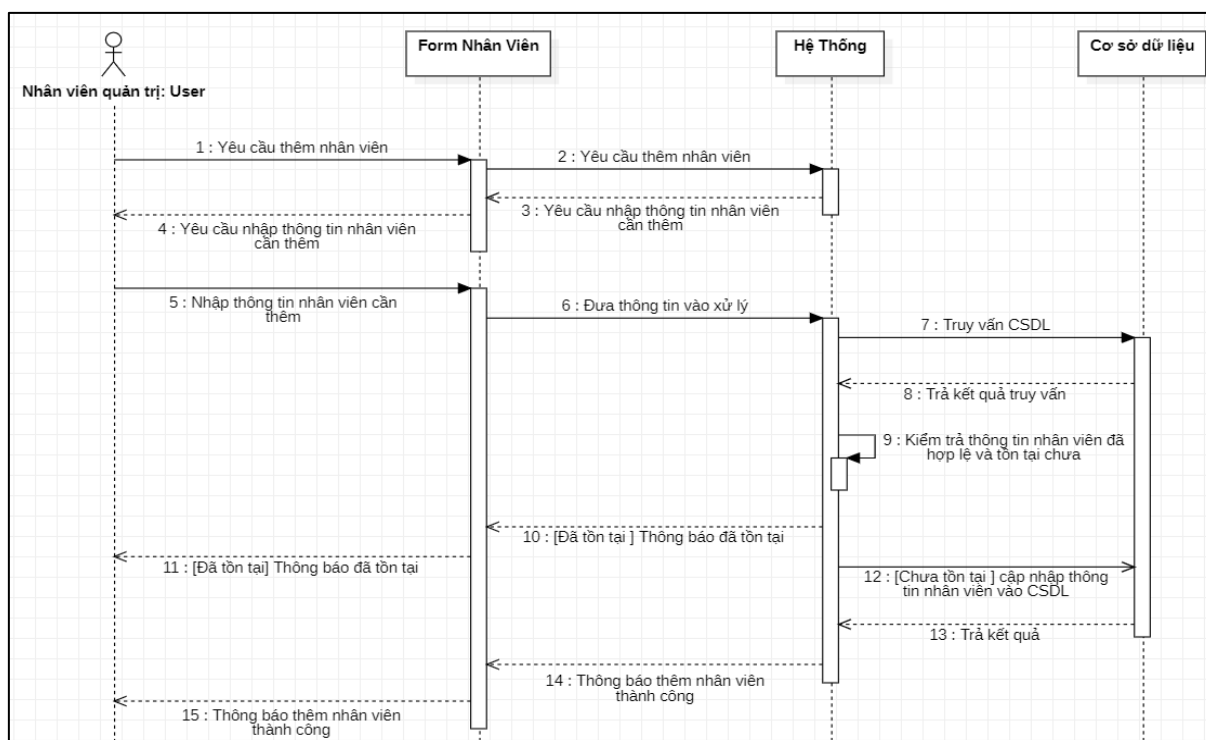
Activity hoạt động xóa nhân viên

3.1.10. Biểu đồ Sequence hoạt động đăng nhập



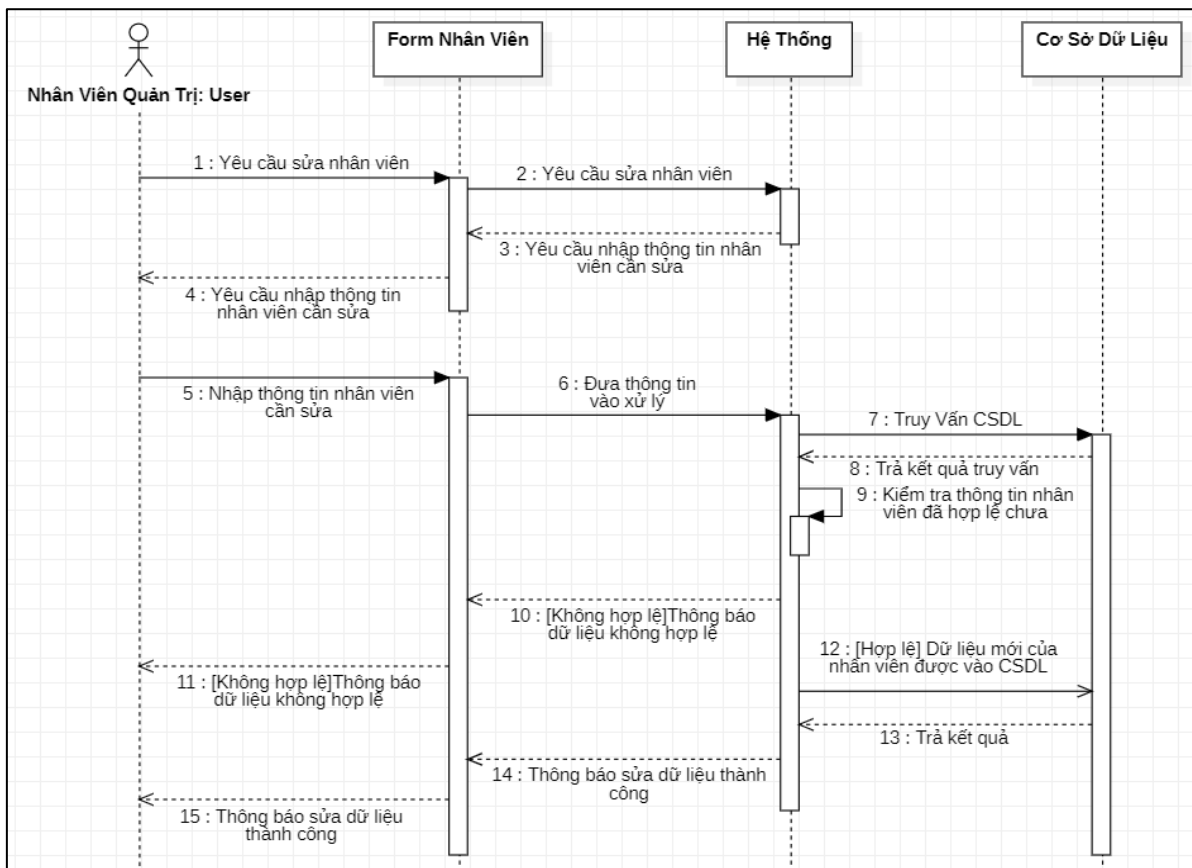
Sequence đăng nhập

3.1.11. Biểu đồ Sequence hoạt động thêm nhân viên



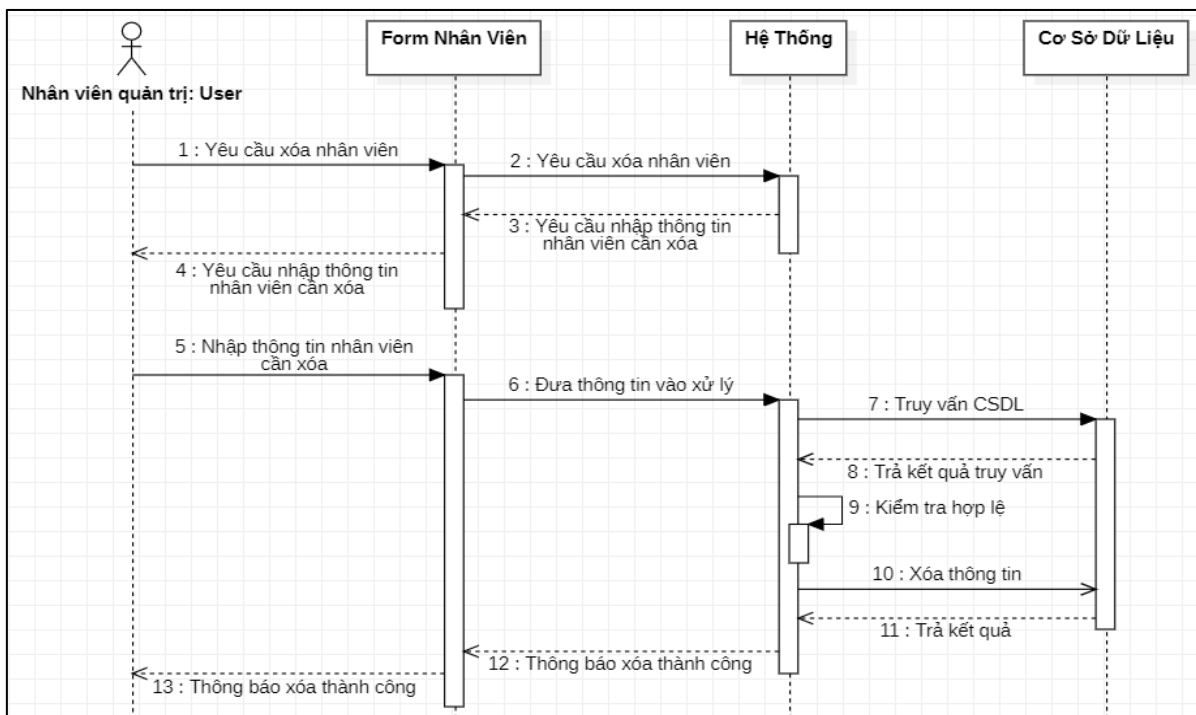
Sequence thêm nhân viên

3.1.12. Biểu đồ Sequence hoạt động sửa nhân viên



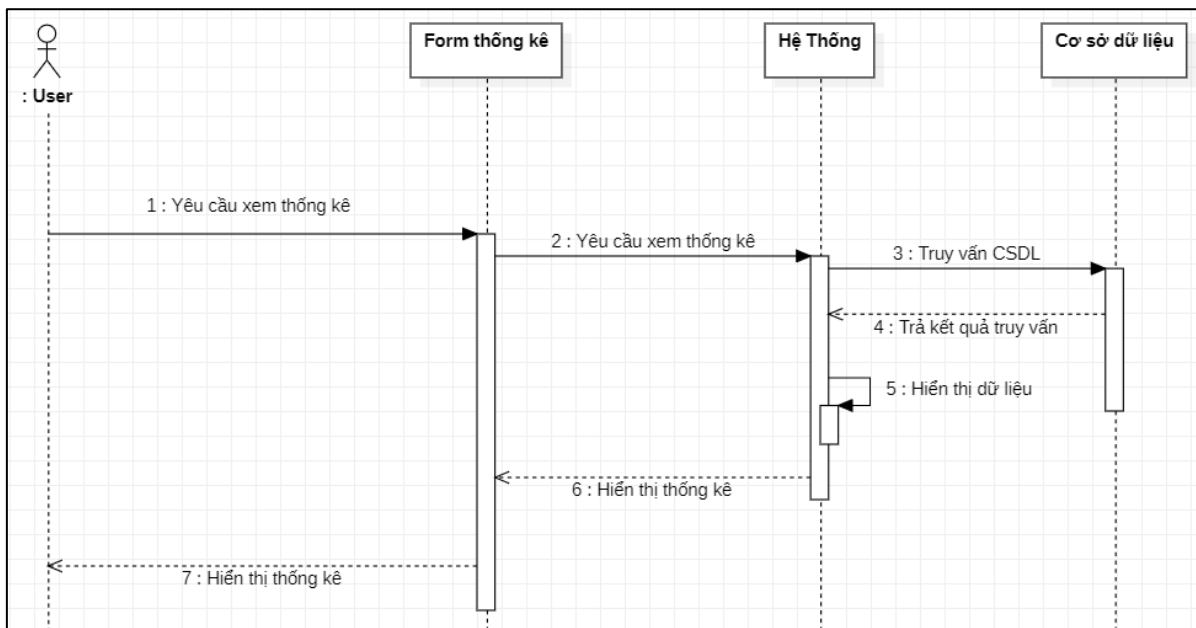
Sequence sửa nhân viên

3.1.13. Biểu đồ Sequence hoạt động xóa nhân viên



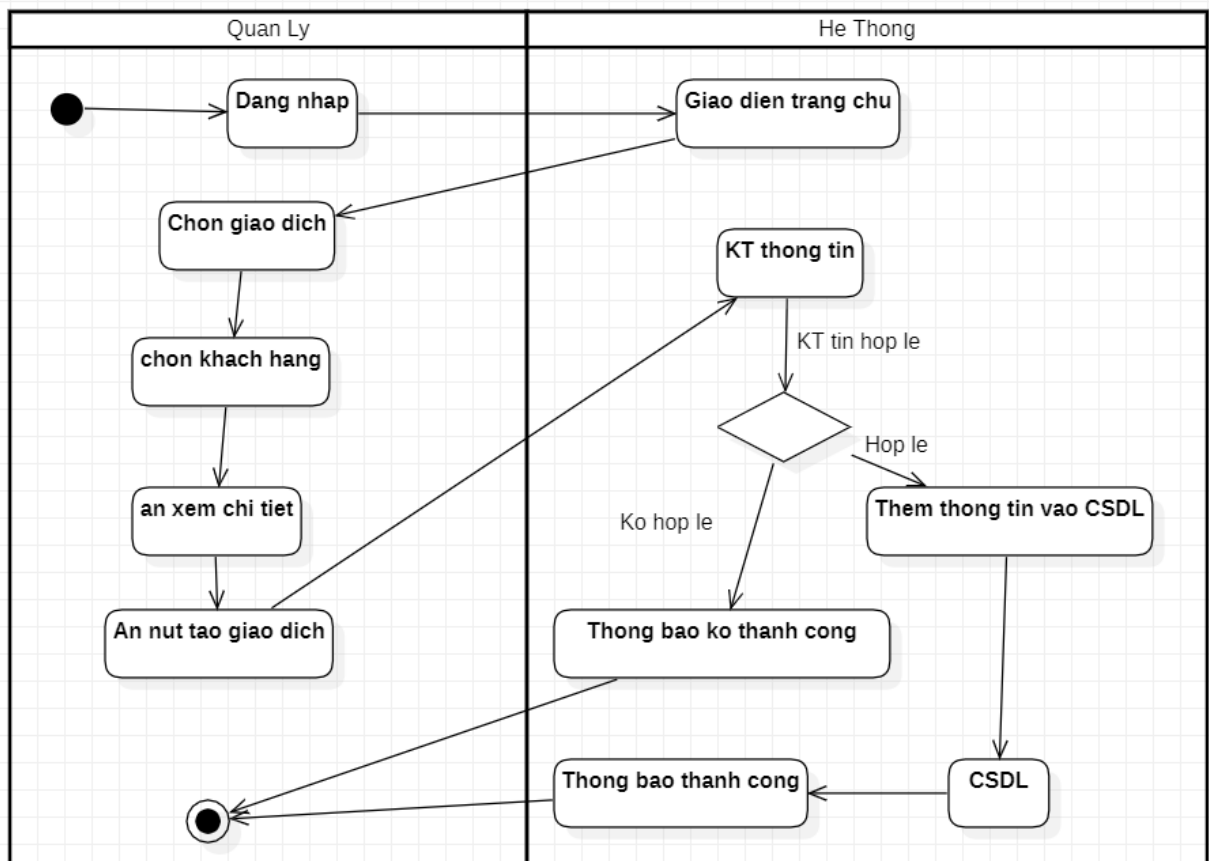
Sequence xóa nhân viên

3.1.14. Sequence Quản lý xem thống kê



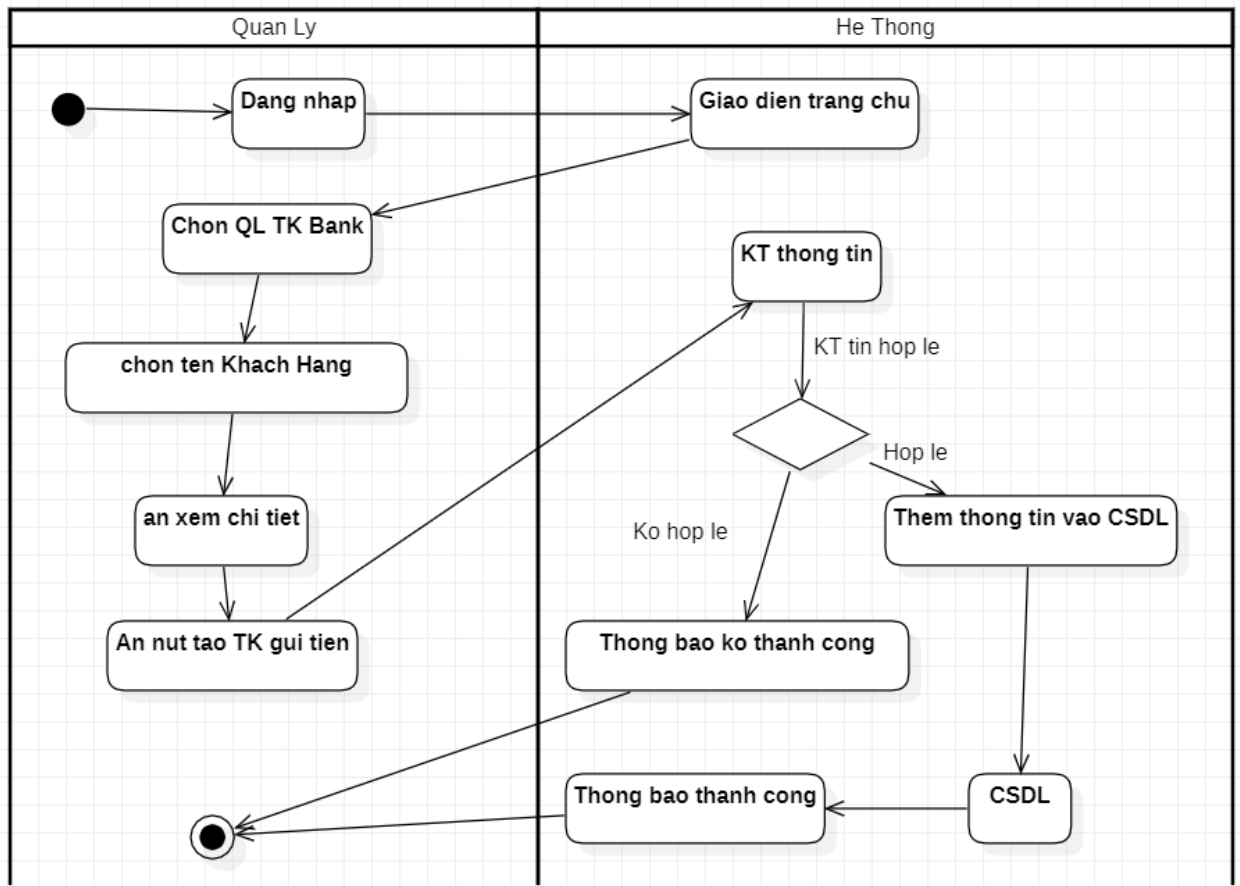
Sequence sơ đồ xem thống kê

3.1.15. Biểu đồ activity tạo giao dịch



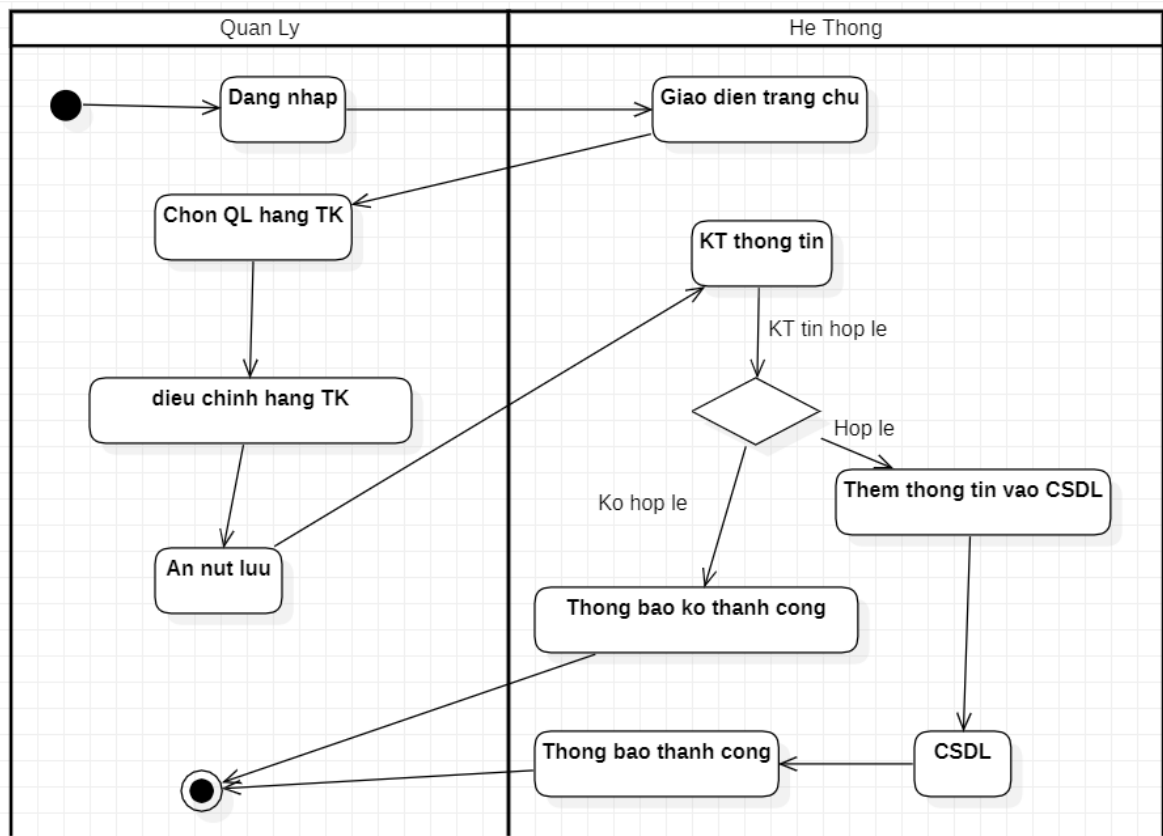
Activity tạo giao dịch

3.1.16. Biểu đồ activity tạo tài khoản gửi tiền



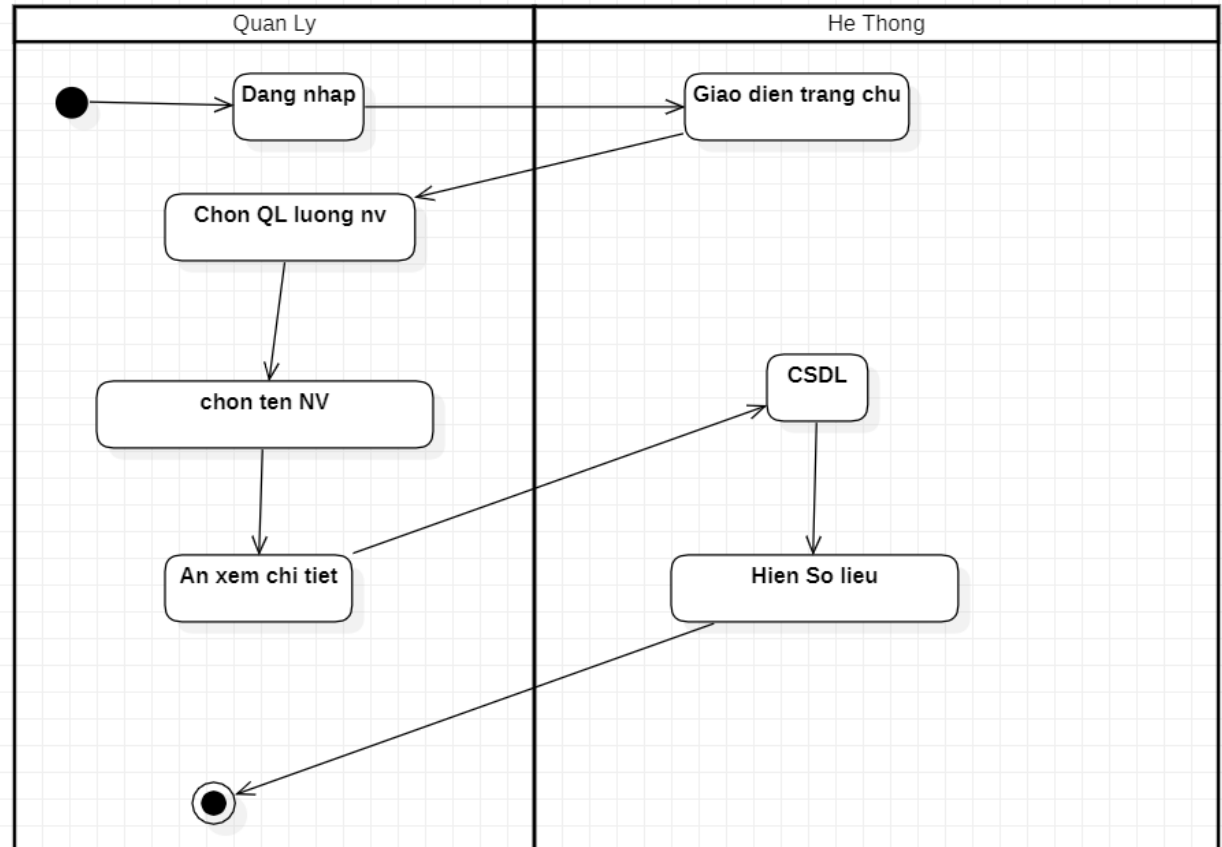
Activity tạo TK gửi tiền

3.1.17. Biểu đồ activity quản lý hạn tài khoản



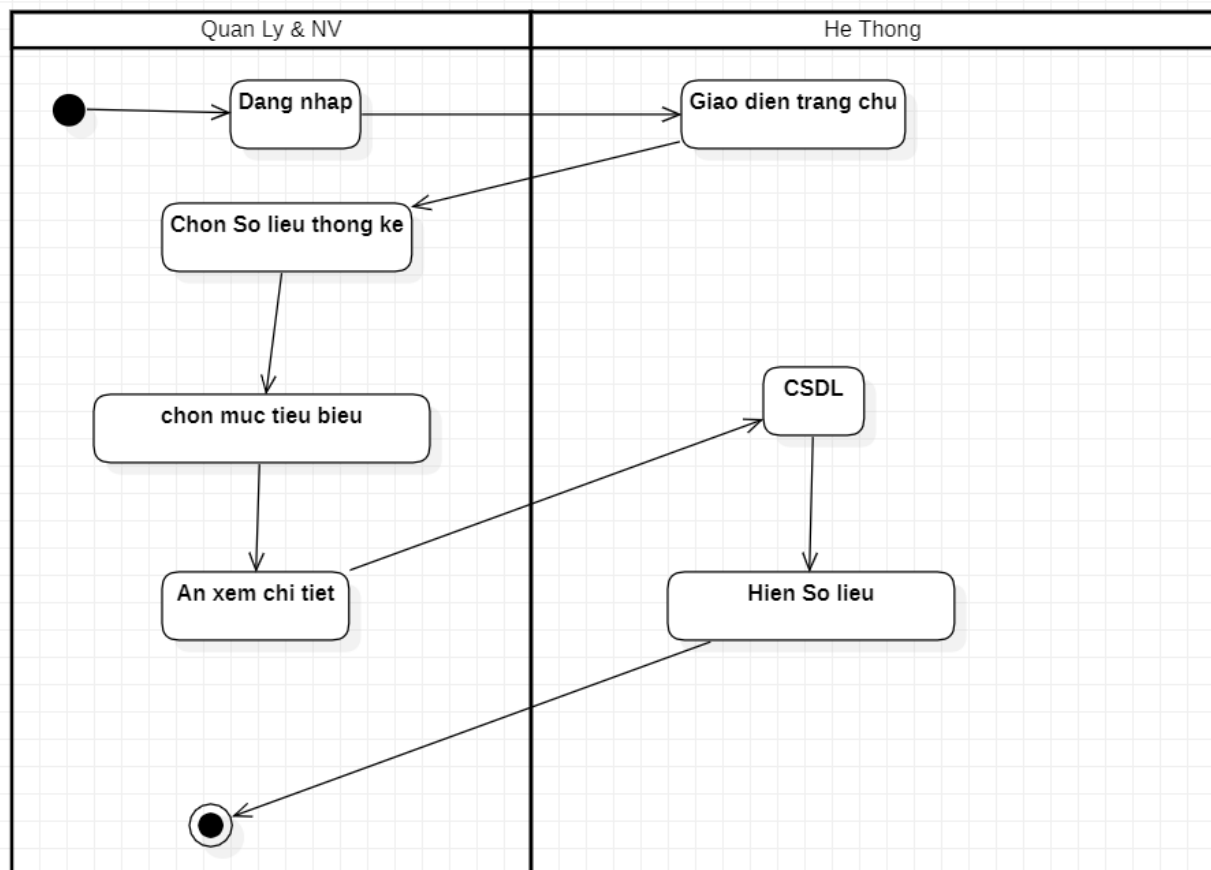
Acivty quản lý hạng tài khoản

3.1.18. Biểu đồ activity xem lương



Activity xem lương

3.1.19. Activity xem số liệu thống kê

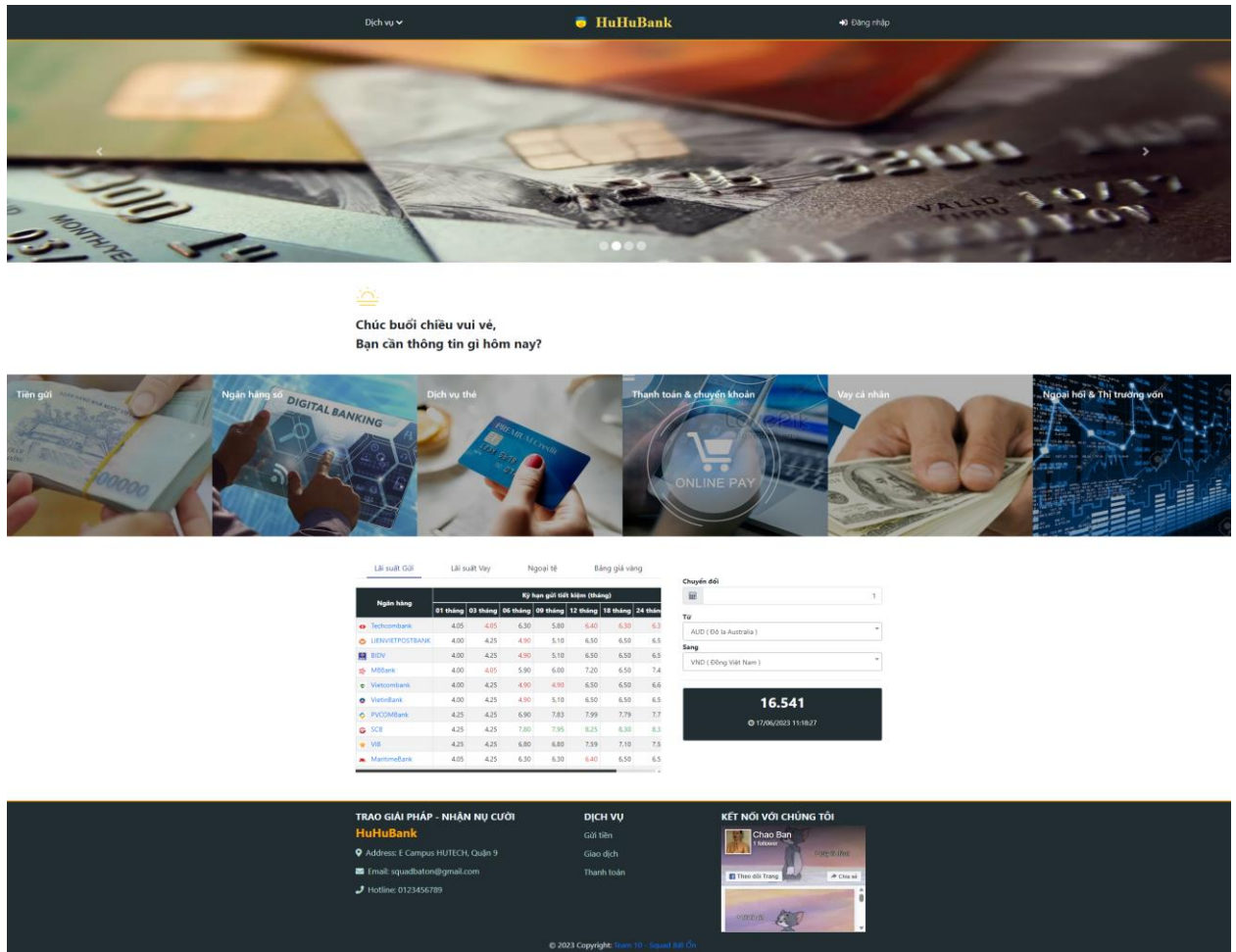


Activity xem số liệu thống kê

CHƯƠNG IV: GIAO DIỆN PHẨM MỀM

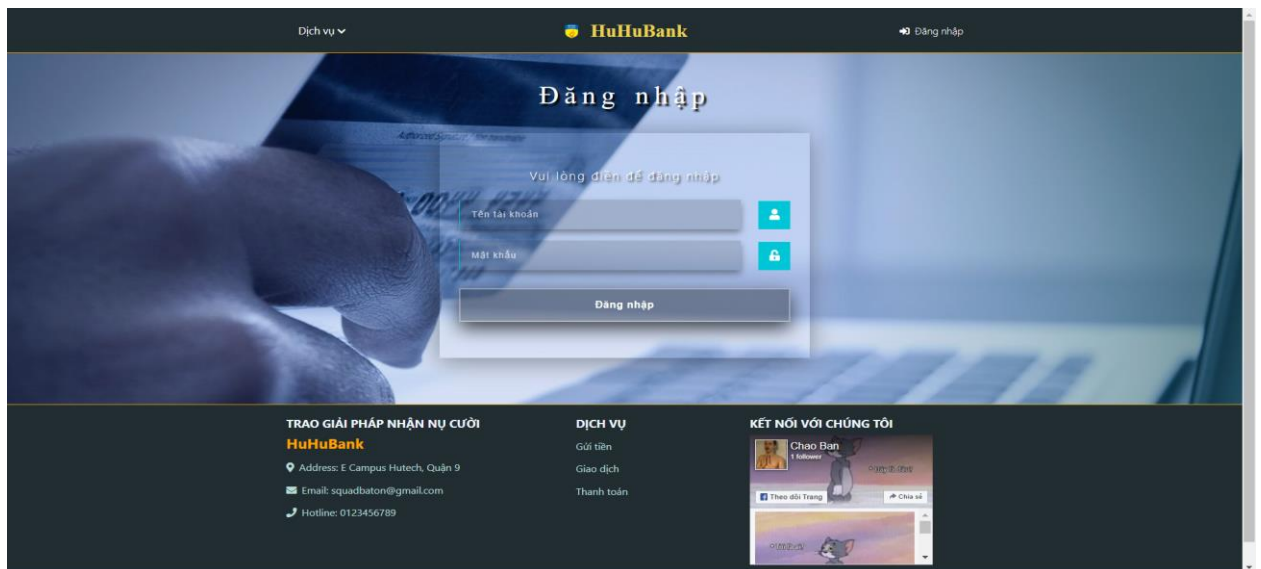
3.2. Giao diện ứng dụng

3.2.1. Trang chủ



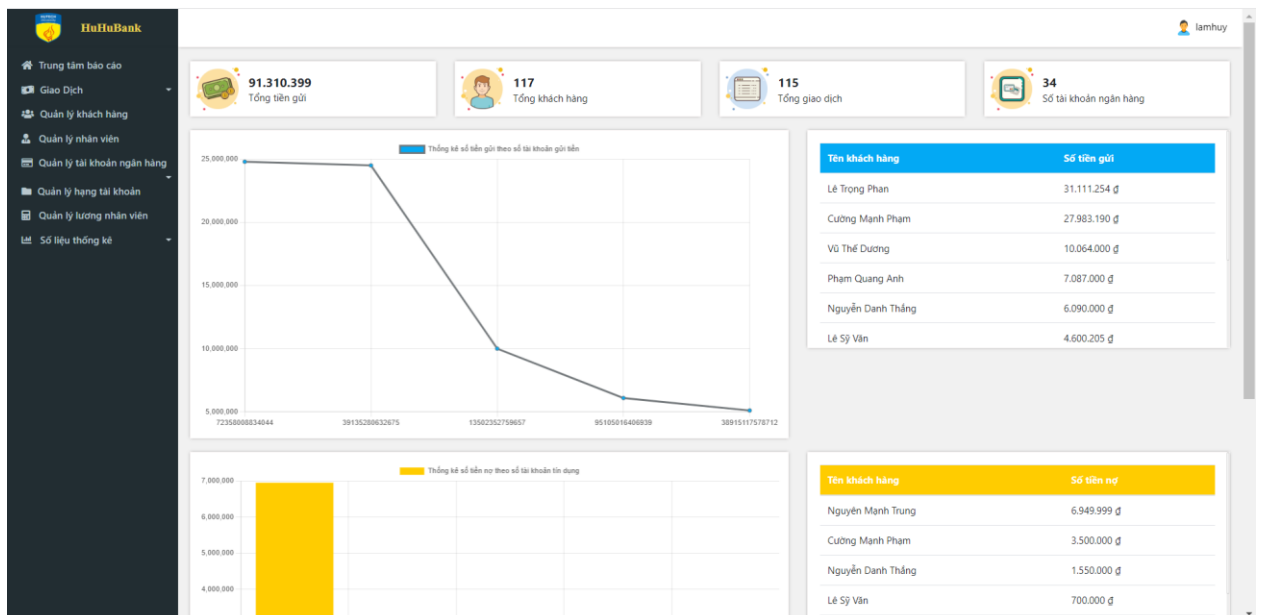
Giao diện trang chủ

3.2.2. Giao diện đăng nhập



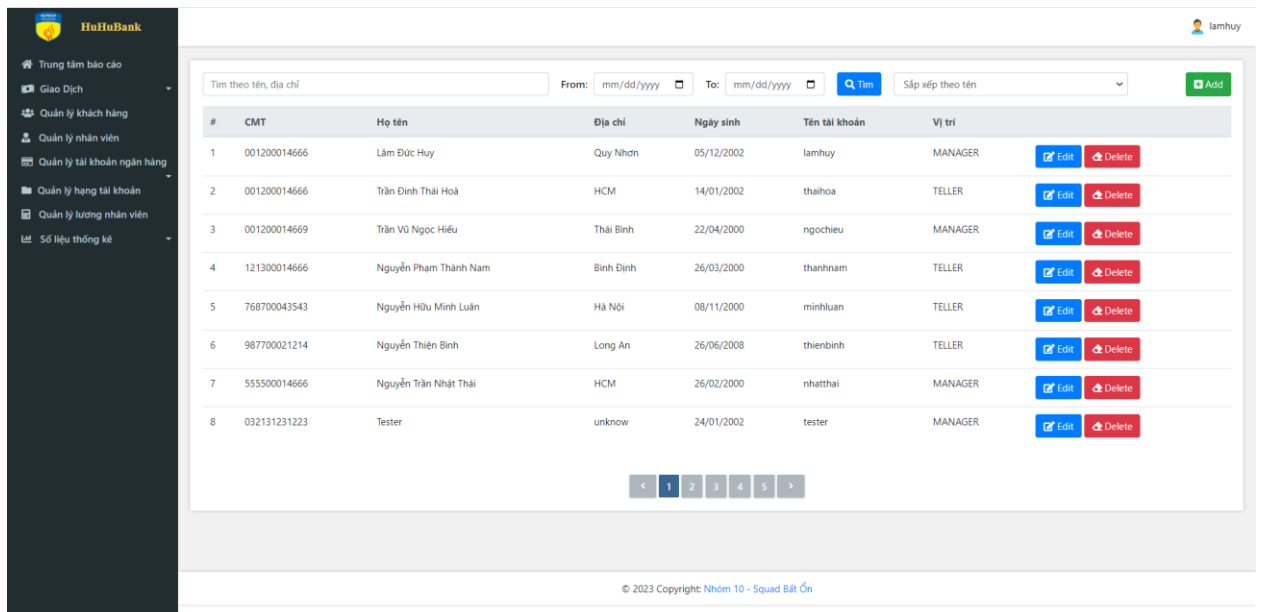
Giao diện đăng nhập

3.2.3. Giao diện chính



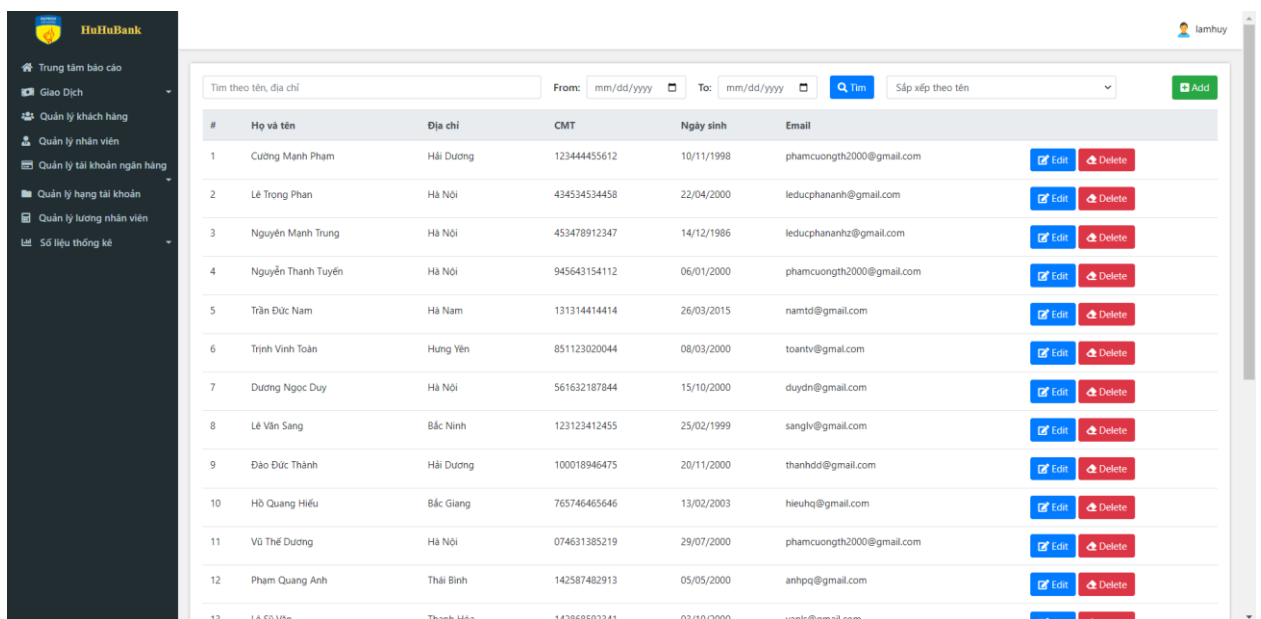
Giao diện chính

3.2.4. Giao diện quản lý nhân viên



Giao diện quản lý nhân viên

3.2.5. Giao diện quản lý khách hàng



Giao diện quản lý khách hàng

3.2.6. Giao diện quản lý lượt giao dịch

<div> <div> <div>HuHuBank</div> </div> <div> <div>Trung tâm báo cáo</div> <div>Giao Dịch</div> <div> <div>Giao dịch</div> <div>Gửi tiền</div> <div>Thanh toán trực tiếp</div> <div>Thanh toán từ tài khoản gửi tiền</div> </div> <div> <div>Quản lý khách hàng</div> <div>Quản lý nhân viên</div> <div>Quản lý tài khoản ngân hàng</div> <div>Quản lý hạn tài khoản</div> <div>Quản lý lương nhân viên</div> <div>Số liệu thống kê</div> </div> </div> </div>																																																																																																																													
<div> <div> <div> <div>Tìm theo tên, địa chỉ</div> <div> <div>From: mm/dd/yyyy</div> <div>To: mm/dd/yyyy</div> <div>Q.Tìm</div> </div> <div>Sắp xếp theo tên</div> </div> <table> <tr> <th>#</th><th>Họ và tên</th><th>Địa chỉ</th><th>CMT</th><th>Email</th><th>Ngày sinh</th><th></th></tr> <tr> <td>1</td><td>Cường Mạnh Phạm</td><td>Hải Dương</td><td>123444455612</td><td>phamcuongth2000@gmail.com</td><td>10/11/1998</td><td>Xem chi tiết</td></tr> <tr> <td>2</td><td>Lê Trọng Phan</td><td>Hà Nội</td><td>434534534458</td><td>leducphananh@gmail.com</td><td>22/04/2000</td><td>Xem chi tiết</td></tr> <tr> <td>3</td><td>Nguyễn Mạnh Trung</td><td>Hà Nội</td><td>453478912347</td><td>leducphananh@gmail.com</td><td>14/12/1986</td><td>Xem chi tiết</td></tr> <tr> <td>4</td><td>Nguyễn Thanh Tuyền</td><td>Hà Nội</td><td>945643154112</td><td>phamcuongth2000@gmail.com</td><td>06/01/2000</td><td>Xem chi tiết</td></tr> <tr> <td>5</td><td>Trần Đức Nam</td><td>Hà Nam</td><td>131314414414</td><td>namtd@gmail.com</td><td>26/03/2015</td><td>Xem chi tiết</td></tr> <tr> <td>6</td><td>Trịnh Vinh Toàn</td><td>Hưng Yên</td><td>851123020044</td><td>toantv@gmail.com</td><td>08/03/2000</td><td>Xem chi tiết</td></tr> <tr> <td>7</td><td>Dương Ngọc Duy</td><td>Hà Nội</td><td>561632187844</td><td>duydn@gmail.com</td><td>15/10/2000</td><td>Xem chi tiết</td></tr> <tr> <td>8</td><td>Lê Văn Sang</td><td>Bắc Ninh</td><td>123123412455</td><td>sanglv@gmail.com</td><td>25/02/1999</td><td>Xem chi tiết</td></tr> <tr> <td>9</td><td>Đào Đức Thành</td><td>Hải Dương</td><td>100018946475</td><td>thanhdd@gmail.com</td><td>20/11/2000</td><td>Xem chi tiết</td></tr> <tr> <td>10</td><td>Hồ Quang Hiếu</td><td>Bắc Giang</td><td>765746465646</td><td>hieuhq@gmail.com</td><td>13/02/2003</td><td>Xem chi tiết</td></tr> <tr> <td>11</td><td>Vũ Thế Dương</td><td>Hà Nội</td><td>074631385219</td><td>phamcuongth2000@gmail.com</td><td>29/07/2000</td><td>Xem chi tiết</td></tr> <tr> <td>12</td><td>Phạm Quang Anh</td><td>Thái Bình</td><td>142587482913</td><td>anhpq@gmail.com</td><td>05/05/2000</td><td>Xem chi tiết</td></tr> <tr> <td>13</td><td>Lê Sỹ Văn</td><td>Thanh Hóa</td><td>142868592341</td><td>vanls@gmail.com</td><td>03/10/2000</td><td>Xem chi tiết</td></tr> <tr> <td>14</td><td>Nguyễn Danh Thắng</td><td>Thanh Hóa</td><td>142958423111</td><td>thangnd@gmail.com</td><td>12/09/2000</td><td>Xem chi tiết</td></tr> <tr> <td>15</td><td>Nguyễn Văn Thử</td><td>Ninh Bình</td><td>140786429131</td><td>thunv@gmail.com</td><td>20/09/2000</td><td>Xem chi tiết</td></tr> <tr> <td>16</td><td>Đặng Đức Long</td><td>Hải Dương</td><td>142864239305</td><td>longdd@gmail.com</td><td>30/01/2000</td><td>Xem chi tiết</td></tr> </table> </div> </div>							#	Họ và tên	Địa chỉ	CMT	Email	Ngày sinh		1	Cường Mạnh Phạm	Hải Dương	123444455612	phamcuongth2000@gmail.com	10/11/1998	Xem chi tiết	2	Lê Trọng Phan	Hà Nội	434534534458	leducphananh@gmail.com	22/04/2000	Xem chi tiết	3	Nguyễn Mạnh Trung	Hà Nội	453478912347	leducphananh@gmail.com	14/12/1986	Xem chi tiết	4	Nguyễn Thanh Tuyền	Hà Nội	945643154112	phamcuongth2000@gmail.com	06/01/2000	Xem chi tiết	5	Trần Đức Nam	Hà Nam	131314414414	namtd@gmail.com	26/03/2015	Xem chi tiết	6	Trịnh Vinh Toàn	Hưng Yên	851123020044	toantv@gmail.com	08/03/2000	Xem chi tiết	7	Dương Ngọc Duy	Hà Nội	561632187844	duydn@gmail.com	15/10/2000	Xem chi tiết	8	Lê Văn Sang	Bắc Ninh	123123412455	sanglv@gmail.com	25/02/1999	Xem chi tiết	9	Đào Đức Thành	Hải Dương	100018946475	thanhdd@gmail.com	20/11/2000	Xem chi tiết	10	Hồ Quang Hiếu	Bắc Giang	765746465646	hieuhq@gmail.com	13/02/2003	Xem chi tiết	11	Vũ Thế Dương	Hà Nội	074631385219	phamcuongth2000@gmail.com	29/07/2000	Xem chi tiết	12	Phạm Quang Anh	Thái Bình	142587482913	anhpq@gmail.com	05/05/2000	Xem chi tiết	13	Lê Sỹ Văn	Thanh Hóa	142868592341	vanls@gmail.com	03/10/2000	Xem chi tiết	14	Nguyễn Danh Thắng	Thanh Hóa	142958423111	thangnd@gmail.com	12/09/2000	Xem chi tiết	15	Nguyễn Văn Thử	Ninh Bình	140786429131	thunv@gmail.com	20/09/2000	Xem chi tiết	16	Đặng Đức Long	Hải Dương	142864239305	longdd@gmail.com	30/01/2000	Xem chi tiết
#	Họ và tên	Địa chỉ	CMT	Email	Ngày sinh																																																																																																																								
1	Cường Mạnh Phạm	Hải Dương	123444455612	phamcuongth2000@gmail.com	10/11/1998	Xem chi tiết																																																																																																																							
2	Lê Trọng Phan	Hà Nội	434534534458	leducphananh@gmail.com	22/04/2000	Xem chi tiết																																																																																																																							
3	Nguyễn Mạnh Trung	Hà Nội	453478912347	leducphananh@gmail.com	14/12/1986	Xem chi tiết																																																																																																																							
4	Nguyễn Thanh Tuyền	Hà Nội	945643154112	phamcuongth2000@gmail.com	06/01/2000	Xem chi tiết																																																																																																																							
5	Trần Đức Nam	Hà Nam	131314414414	namtd@gmail.com	26/03/2015	Xem chi tiết																																																																																																																							
6	Trịnh Vinh Toàn	Hưng Yên	851123020044	toantv@gmail.com	08/03/2000	Xem chi tiết																																																																																																																							
7	Dương Ngọc Duy	Hà Nội	561632187844	duydn@gmail.com	15/10/2000	Xem chi tiết																																																																																																																							
8	Lê Văn Sang	Bắc Ninh	123123412455	sanglv@gmail.com	25/02/1999	Xem chi tiết																																																																																																																							
9	Đào Đức Thành	Hải Dương	100018946475	thanhdd@gmail.com	20/11/2000	Xem chi tiết																																																																																																																							
10	Hồ Quang Hiếu	Bắc Giang	765746465646	hieuhq@gmail.com	13/02/2003	Xem chi tiết																																																																																																																							
11	Vũ Thế Dương	Hà Nội	074631385219	phamcuongth2000@gmail.com	29/07/2000	Xem chi tiết																																																																																																																							
12	Phạm Quang Anh	Thái Bình	142587482913	anhpq@gmail.com	05/05/2000	Xem chi tiết																																																																																																																							
13	Lê Sỹ Văn	Thanh Hóa	142868592341	vanls@gmail.com	03/10/2000	Xem chi tiết																																																																																																																							
14	Nguyễn Danh Thắng	Thanh Hóa	142958423111	thangnd@gmail.com	12/09/2000	Xem chi tiết																																																																																																																							
15	Nguyễn Văn Thử	Ninh Bình	140786429131	thunv@gmail.com	20/09/2000	Xem chi tiết																																																																																																																							
16	Đặng Đức Long	Hải Dương	142864239305	longdd@gmail.com	30/01/2000	Xem chi tiết																																																																																																																							

Giao diện quản lý lượt giao dịch

3.2.7. Giao diện xem chi tiết lượt giao dịch

</

Giao diện xem chi tiết lượt giao dịch

3.2.8. Giao diện quản lý hạn tài khoản

- Trung tâm báo cáo
- Giao Dịch
- Quản lý khách hàng
- Quản lý nhân viên
- Quản lý tài khoản ngân hàng
- Quản lý hạn tài khoản
- Quản lý lương nhân viên
- Số liệu thống kê

Search...

Add

#	Tên	Ghi chú		
1	Silver	bạc	Edit	Delete
2	Gold	vàng	Edit	Delete
3	Platinum	bạc kim	Edit	Delete
4	Diamond	kim cương	Edit	Delete
5	Master	Chủ tịch	Edit	Delete
6	Bronze	Đồng	Edit	Delete

© 2023 Copyright: Nhóm 10 - Squad Bất Ổn

Giao diện quản lý hạn tài khoản

3.2.9. Giao diện số liệu thống kê

- Trung tâm báo cáo
- Giao Dịch
- Quản lý khách hàng
- Quản lý nhân viên
- Quản lý tài khoản ngân hàng
- Quản lý hạn tài khoản
- Quản lý lương nhân viên
- Số liệu thống kê
 - Thống kê 10 khách hàng với số tiền gửi nhiều nhất
 - Thống kê 10 khách hàng với số tiền nợ nhiều nhất
 - Thống kê 10 tài khoản tín dụng có số nợ nhiều nhất
 - Thống kê 10 tài khoản gửi tiền có số dư nhiều nhất
 - Thống kê khách hàng với số tiền giao dịch theo các loại giao dịch trong khoảng thời gian
 - Hiện thị các giao dịch theo khách hàng và xuất dữ liệu ra Excel
 - Tìm khách hàng và liệt kê giao dịch của số tài khoản đó

Search...

#	Họ và tên	Địa chỉ	CMT	Ngày sinh	Email	Số tiền gửi
1	Lê Trọng Phan	Hà Nội	434534534458	22/04/2000	leducphananh@gmail.com	31.111.254 đ
2	Cường Mạnh Phạm	Hải Dương	123444455612	10/11/1998	phamcuongth2000@gmail.com	27.983.190 đ
3	Vũ Thế Dương	Hà Nội	074631385219	29/07/2000	phamcuongth2000@gmail.com	10.064.000 đ
4	Phạm Quang Anh	Thái Bình	142587482913	05/05/2000	anhpq@gmail.com	7.087.000 đ
5	Nguyễn Danh Thắng	Thanh Hóa	142958423111	12/09/2000	thangnd@gmail.com	6.090.000 đ
6	Lê Sỹ Văn	Thanh Hóa	142868592341	03/10/2000	vanls@gmail.com	4.600.205 đ
7	Đặng Đức Long	Hải Dương	142864239305	30/01/2000	longdd@gmail.com	2.464.700 đ
8	Nguyễn Văn Thử	Ninh Bình	140786429131	20/09/2000	thunv@gmail.com	1.471.750 đ
9	Nguyễn Mạnh Trung	Hà Nội	453478912347	14/12/1986	leducphananhz@gmail.com	256.000 đ
10	Hồ Quang Hiếu	Bắc Giang	765746465646	13/02/2003	hieuhq@gmail.com	102.300 đ

© 2023 Copyright: Nhóm 10 - Squad Bất Ổn

Giao diện số liệu thống kê

3.2.10 Giao diện quản lý lương nhân viên

HallaBank

Trung tâm bảo cáo

Giao Dịch

Quản lý khách hàng

Quản lý nhân viên

Quản lý tài khoản ngân hàng

Quản lý hàng tài khoản

Quản lý lương nhân viên

Số liệu thống kê

Tháng: 6 Năm: 2023 Xem lương

Export Excel

#	CMT	Họ tên	Địa chỉ	Ngày sinh	Vị trí	Tổng lương	
1	001200014666	Lâm Đức Huy	Quy Nhơn	05/12/2002	MANAGER	9.000.000 đ	Xem chi tiết lương
2	001200014666	Trần Đình Thái Hoà	HCM	14/01/2002	MANAGER	6.000.000 đ	Xem chi tiết lương
3	001200014669	Trần Vũ Ngọc Hiếu	Thái Bình	22/04/2000	MANAGER	9.000.000 đ	Xem chi tiết lương
4	121300014666	Nguyễn Phạm Thành Nam	Bình Định	26/03/2000	TELLER	6.000.000 đ	Xem chi tiết lương
5	768700043543	Nguyễn Hữu Minh Luân	Hà Nội	08/11/2000	TELLER	6.000.000 đ	Xem chi tiết lương
6	987700021214	Nguyễn Thiện Bình	Long An	26/06/2008	TELLER	6.000.000 đ	Xem chi tiết lương
7	555500014666	Nguyễn Trần Nhật Thái	HCM	26/02/2000	TELLER	9.000.000 đ	Xem chi tiết lương

1

2

3

4

5

6

7

© 2023 Copyright: Nhóm 10 - Squad B&T Ôn

Giao diện quản lý lương – và xuất file excel

3.2.11 Giao diện xem hồ sơ cá nhân

HallaBank

Trung tâm bảo cáo

Giao Dịch

Quản lý khách hàng

Quản lý nhân viên

Quản lý tài khoản ngân hàng

Quản lý hàng tài khoản

Quản lý lương nhân viên

Số liệu thống kê

Họ và tên	Lâm Đức Huy
Vị trí	MANAGER
Ngày sinh	05/12/2002
CMT	001200014666
Địa chỉ	Quy Nhơn
Thâm niên	10
Bậc nghề	12

Giao diện xem hồ sơ cá nhân

3.2.12 Giao diện xem lương bản thân

- Trung tâm báo cáo
- Giao Dịch
- Quản lý khách hàng
- Quản lý nhân viên
- Quản lý tài khoản ngân hàng
- Quản lý hàng tài khoản
- Quản lý lương nhân viên
- Sổ liệu thống kê

Họ và tên

Lâm Đức Huy

Vị trí

MANAGER

Thâm niên

10

Bậc nghề

12

From:

Nhập theo định dạng MM/yyyy

To:

Nhập theo định dạng MM/yyyy

Q.Tìm

Sắp xếp theo lương

#	Lương cơ bản	Lương thưởng	Lương tháng
1	9.000.000 đ	3.424.600 đ	12/2021
2	9.000.000 đ	0 đ	06/2023

<

1

2

3

4

5

>

Giao diện xem lương bản thân

CHƯƠNG V: KẾT LUẬN VÀ KIẾN NGHỊ

4.1. Kết luận

Dựa trên quá trình nghiên cứu và triển khai, chúng ta đã thành công xây dựng một hệ thống web quản lý ngân hàng với các chức năng quan trọng như quản lý tài khoản ngân hàng, giao dịch và thanh toán, quản lý khách hàng, quản lý vay và cho vay, quản lý báo cáo và thống kê. Hệ thống được xây dựng dựa trên kiến trúc Layered Architecture, sử dụng công nghệ Java 8, Spring MVC, Spring Boot, Thymeleaf, AJAX, Spring Security, Poi-Ooxml và Spring Mail.

Kết quả đạt được từ việc triển khai hệ thống là đáng khích lệ. Hệ thống đáp ứng được các yêu cầu chức năng của một dịch vụ quản lý ngân hàng hiện đại, mang lại tiện ích và trải nghiệm tốt cho người dùng. Việc sử dụng các công nghệ và frameworks như Java 8, Spring và Thymeleaf đã giúp tăng tính bảo mật, hiệu suất và khả năng mở rộng của hệ thống.

4.2. Kiến nghị

Dựa trên quá trình triển khai và trải nghiệm của hệ thống, chúng ta có thể đưa ra một số kiến nghị và định hướng phát triển trong tương lai:

Tăng cường bảo mật: Hệ thống quản lý ngân hàng đòi hỏi mức độ bảo mật cao. Chúng ta có thể nghiên cứu và áp dụng các biện pháp bảo mật bổ sung như mã hóa dữ liệu, xác thực hai lớp và kiểm tra thường xuyên để đảm bảo an toàn cho thông tin ngân hàng.

Mở rộng chức năng: Để đáp ứng nhu cầu ngày càng đa dạng của người dùng, chúng ta có thể mở rộng chức năng của hệ thống bằng cách thêm tính năng như chuyển khoản quốc tế, tạo lập tài khoản tiết kiệm, quản lý danh sách người thụ hưởng, và hỗ trợ thanh toán bằng các phương thức thanh toán mới.

Tối ưu hóa hiệu suất: Để đảm bảo hệ thống hoạt động mượt mà và đáp ứng được tải cao, chúng ta có thể tối ưu hóa hiệu suất bằng cách sử dụng bộ nhớ cache, tối ưu hóa truy vấn cơ sở dữ liệu, và tăng cường khả năng xử lý đồng thời.

Cải thiện giao diện người dùng: Giao diện người dùng đóng vai trò quan trọng trong trải nghiệm của người dùng. Chúng ta có thể nâng cấp giao diện người dùng bằng cách sử dụng các thư viện và công nghệ giao diện tiên tiến, tạo ra trải nghiệm tương tác hấp dẫn và thân thiện.

Kiểm thử và bảo trì: Quá trình kiểm thử và bảo trì đóng vai trò quan trọng trong việc duy trì và cải thiện chất lượng của hệ thống. Chúng ta nên thực hiện kiểm thử tự động, kiểm thử đơn vị và kiểm thử hệ thống để phát hiện và khắc phục các lỗi và vấn đề một cách nhanh chóng.

Những kiến nghị và định hướng phát triển trên sẽ giúp nâng cao chất lượng và khả năng phục vụ của hệ thống quản lý ngân hàng. Chúng ta cần tiếp tục nghiên cứu, cải tiến và theo dõi các xu hướng công nghệ mới để đáp ứng nhu cầu ngày càng cao của khách hàng và thị trường.

4.3. Tài liệu tham khảo

- [DyvakYA/bank: :moneybag: Web project \(online banking\) for banking system \(honest.bank\). Spring, Hibernate, Lombok - in use. \(github.com\)](#)
- [Hệ thống quản lý ngân hàng trong Java - Codefly.Vn](#)
- [Share Code - Quản lý ATM - Quản lý tài khoản ngân hàng \(gokisoft.com\)](#)
- [Java-SO: Hệ thống quản lý tài khoản ngân hàng | V1Study.com - YouTube](#)
- [Java - Bank Account Management \(w3resource.com\)](#)
- [Banking Application in Java - Javatpoint](#)

