

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN

ỨNG DỤNG CÔNG NGHỆ SPRING MVC
THIẾT KẾ WEBSITE QUẢN LÝ CỬA HÀNG TIỆN LỢI

Giảng viên hướng dẫn: ThS. Mai Anh Thor

Nhóm sinh viên thực hiện

Nguyễn Quốc Huy: 21110189

Đào Hoàng Đăng: 21110163

Nguyễn Hoàng Lực: 21110838

Thành phố Hồ chí Minh, tháng 11 năm 2023

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN

ỨNG DỤNG CÔNG NGHỆ SPRING MVC THIẾT KẾ WEBSITE QUẢN LÝ CỬA HÀNG TIỆN LỢI

Giảng viên hướng dẫn: ThS. Mai Anh Thơ

Nhóm sinh viên thực hiện

Nguyễn Quốc Huy: 21110189

Đào Hoàng Đăng: 21110163

Nguyễn Hoàng Lực: 21110838

Thành phố Hồ chí Minh, tháng 11 năm 2023

DANH SÁCH THÀNH VIÊN THAM GIA

HỌC KỲ I, NĂM HỌC: 2023 – 2024

Tên đề tài: QUẢN LÝ CỬA HÀNG TIỆN LỢI

STT	Họ và tên	MSSV	Tỉ lệ hoàn thành
1	Nguyễn Quốc Huy	21110189	100%
2	Đào Hoàng Đăng	21110163	100%
3	Nguyễn Hoàng Lực	21110838	100%

Ghi chú:

- Tỷ lệ % = 100% (Mức độ phần trăm của từng sinh viên tham gia).
 - Trưởng nhóm: Nguyễn Quốc Huy SĐT: 0842955519
-

MỤC LỤC

LỜI CẢM ƠN.....	1
LỜI NÓI ĐẦU.....	2
CHƯƠNG 1: MÔ TẢ ĐỀ TÀI.....	3
1. Mục tiêu.....	3
2. Lợi ích người dùng	3
2.1. Đối với Quản lý	3
2.2. Đối với Nhân Viên.....	4
3. Use Case diagram	5
4. Bảng mô tả Use Case.....	6
5. Bảng phân công công việc.....	10
CHƯƠNG 2: THIẾT KẾ ĐỀ TÀI.....	11
1. Miêu tả quá trình	11
1.1. Quản lý	11
1.2. Nhân viên.....	11
2. Các chức năng	11
2.1. Chức năng Đăng Nhập và Phân Quyền	11
2.2. Các chức năng của Quản lý	16
2.2.1. Quản lý Nhân Viên	16
2.2.2. Quản lý Khách Hàng	20
2.2.3. Quản lý Sản Phẩm.....	22
2.2.4. Quản lý Phân Loại Sản Phẩm	26
2.2.5. Quản lý Hóa Đơn Nhập Hàng.....	30
2.2.6. Quản lý Hóa Đơn Thanh Toán	32
2.3. Các chức năng của Nhân Viên	35
2.3.1. Quản lý Thanh Toán và xuất Hóa Đơn.....	35
2.3.2. Quản lý Sản Phẩm.....	40
2.3.3. Quản lý Danh Sách Khách Hàng	42
2.3.4. Quản lý Nhập Hàng	46
CHƯƠNG 3: KẾT LUẬN.....	49
1. Đánh giá của sinh viên	49
2. Ưu điểm và nhược điểm	49

3. Hướng phát triển.....	49
TÀI LIỆU THAM KHẢO	50

LỜI CẢM ƠN

Nhóm chúng em xin chân thành bày tỏ lòng biết ơn sâu sắc đến Cô Mai Anh Thơ, người đã đồng hành cùng chúng em trong suốt quá trình thực hiện đề tài. Chúng em muốn dành lời cảm ơn tới Cô vì những đề xuất quý báu và sự hướng dẫn tận tâm, giúp nhóm chúng em hiểu rõ hơn về những vấn đề phức tạp trong quá trình làm đề tài này.

Đề tài của chúng em đã thuận lợi hoàn thành nhờ có sự hỗ trợ từ Cô Mai Anh Thơ. Mặc dù thời gian làm dự án chỉ kéo dài mười lăm tuần, nhưng với lượng kiến thức mới và thời gian hạn chế, không tránh khỏi việc xuất hiện một số sai sót. Em rất mong nhận được sự phản hồi chân thành từ Cô để có thể nâng cao kiến thức bị hạn chế của bản thân cũng như rút kinh nghiệm làm tiền đề cho những môn học sau này. Xin chân thành cảm ơn cô

LỜI NÓI ĐẦU

Trong thời kỳ cách mạng công nghiệp 4.0, sự lan tỏa của công nghệ thông tin đã mở ra nhiều cơ hội mới và thách thức đối với nhiều lĩnh vực, trong đó có lĩnh vực quản lý và kinh doanh. Bối cảnh này đặt ra yêu cầu cao về sự linh hoạt, tính hiệu quả và khả năng đáp ứng nhanh chóng của các hệ thống thông tin. Chính vì vậy, đề tài “ỨNG DỤNG CÔNG NGHỆ SPRING MVC THIẾT KẾ WEBSITE QUẢN LÝ CỬA HÀNG TIỆN LỢI” được nhóm chúng em lựa chọn không chỉ để đáp ứng nhu cầu thực tế của việc quản lý cửa hàng mà còn nhằm khám phá và áp dụng những tiến bộ công nghệ mới nhất.

Bài báo cáo này sẽ giới thiệu quá trình nghiên cứu, phân tích và triển khai dự án, với sự tập trung vào việc ứng dụng Spring MVC, một trong những framework phổ biến trong phát triển ứng dụng web Java. Em sẽ trình bày cách em đã xây dựng và tối ưu hóa hệ thống quản lý cửa hàng tiện lợi, đồng thời đề cập đến những thách thức mà em đã đối mặt và cách giải quyết.

Em hy vọng rằng thông qua bài báo cáo này, cô sẽ có cái nhìn rõ ràng và chi tiết về quá trình xây dựng website, đồng thời em mong nhận được sự quan tâm và phản hồi tích cực từ cô.

CHƯƠNG 1: MÔ TẢ ĐỀ TÀI

1. Mục tiêu

Mục tiêu chính của đề tài này là xây dựng một hệ thống quản lý cửa hàng tiện lợi sử dụng công nghệ Spring MVC kết hợp với việc xây dựng một giao diện người dùng thân thiện, dễ sử dụng, và thẩm mỹ để tạo trải nghiệm tốt nhất cho người quản lý cửa hàng và nhân viên của họ. Bên cạnh đó việc phát triển chức năng quản lý dữ liệu một cách hiệu quả, bao gồm quản lý sản phẩm, nhân viên, đơn hàng, và các thông tin liên quan, giúp cải thiện quá trình quản lý cửa hàng.

Thêm vào đó phát triển tính năng thống kê chi tiêu và báo cáo doanh thu, giúp người quản lý dễ dàng theo dõi hiệu suất kinh doanh và đưa ra quyết định thông minh dựa trên hóa đơn nhập và bán hàng của cửa hàng. Tính năng này sẽ giúp cải thiện khả năng quản lý tài chính của cửa hàng, tăng tính minh bạch và giảm rủi ro liên quan đến chi tiêu và doanh thu. Đồng thời, nó cũng cung cấp công cụ hữu ích để đưa ra quyết định chiến lược và tối ưu hóa hiệu suất kinh doanh.

2. Lợi ích người dùng

2.1. Đối với Quản lý

- **Thống Kê Tổng Quan:** Quản lý có thể dễ dàng theo dõi các chỉ số chính như doanh thu, chi tiêu, và lợi nhuận thông qua các bảng danh sách đơn hàng và danh sách nhập hàng. Điều này giúp quản lý đưa ra quyết định chiến lược và điều chỉnh kế hoạch kinh doanh.
- **Theo Dõi Quản Lý Nhân Viên:** Quản lý có khả năng thêm, đánh giá hiệu suất của nhân viên thông qua việc theo dõi doanh số bán hàng, đơn hàng xử lý, và các chỉ số khác. Điều này giúp tối ưu hóa phân công công việc và đào tạo nhân viên hiệu quả hơn.
- **Quản Lý Sản Phẩm:** Quản lý có thể dễ dàng thêm mới sản phẩm vào hệ thống với các thông tin chi tiết như tên sản phẩm, loại sản phẩm, giá và số

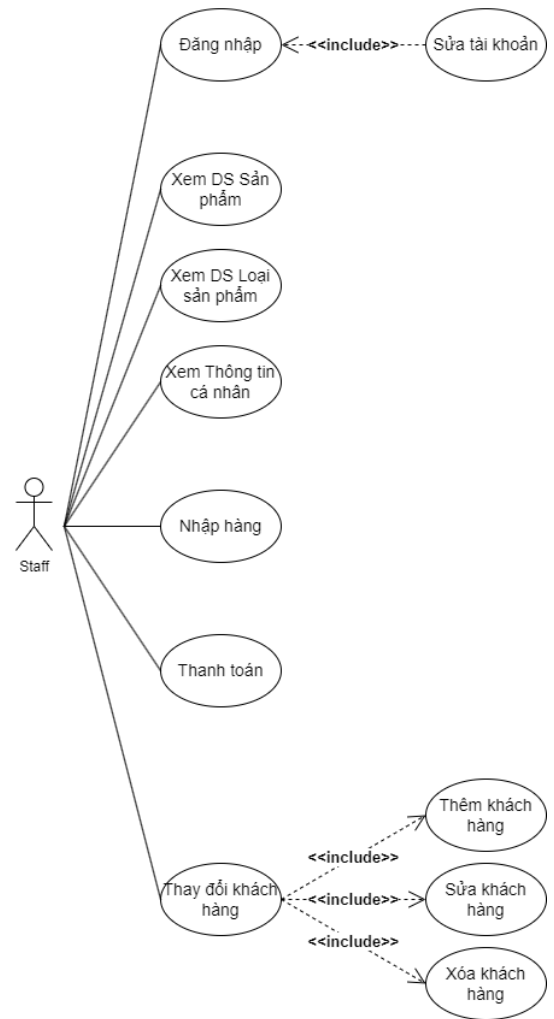
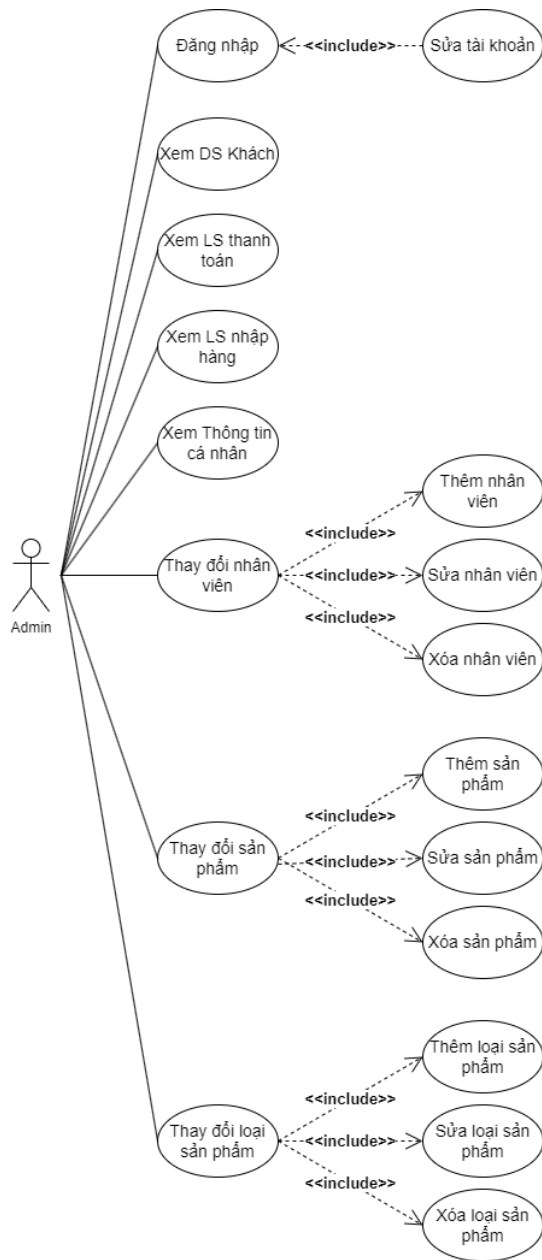
lượng tồn kho. Cập nhật thông tin sản phẩm, điều chỉnh giá cả hoặc thay đổi loại sản phẩm.

- Quản Lý Nhập Hàng: Liên kết với các nhà cung cấp, theo dõi tình trạng đặt hàng và nhập hàng để đảm bảo nguồn cung ổn định.
- Quản Lý Từ Xa: Quản lý có thể truy cập hệ thống từ xa để kiểm soát và theo dõi hoạt động cửa hàng mọi lúc, mọi nơi, đảm bảo tính minh bạch và an toàn thông tin.

2.2. Đối với Nhân Viên

- Quản Lý Đơn Hàng: Nhân viên có thể tạo, xem và quản lý hóa đơn bán hàng và nhập hàng một cách dễ dàng. Thông tin chi tiết về đơn hàng, số lượng, tổng tiền sẽ giúp nhân viên nhanh chóng đáp ứng nhu cầu của khách hàng.
- Quản lý Nhập Hàng: Hệ thống tự động cập nhật số lượng của sản phẩm sau mỗi giao dịch và nhập hàng giúp nhân viên theo dõi số lượng hàng tồn kho, nhập hàng khi sản phẩm đã hết để đảm bảo sẵn có sản phẩm để phục vụ khách hàng.
- Quản lý Dữ Liệu Cá Nhân: Nhân viên có thể điều chỉnh thông tin cá nhân của mình và cập nhật lại nếu như có sai sót về thông tin cá nhân của bản thân.
- Quản lý Phân Loại Sản Phẩm: Nhân viên có thể xem sản phẩm theo danh mục, thương hiệu, hay các thuộc tính khác để dễ dàng tìm kiếm.
- Quản Lý Thông Tin Khách Hàng: Nhân viên có thể dễ dàng thêm mới thông tin khách hàng vào hệ thống, bao gồm mã khách hàng, tên, số điện thoại.

3. Use Case diagram



4. Bảng mô tả Use Case

Table 1 - Use case [Đăng nhập] description

Use case name	Đăng nhập
Description	Cho phép người dùng đăng nhập vào hệ thống
Actor	Admin & Staff
Preconditions	Nhấp vào button “ <i>Đăng nhập</i> ”, điền tên đăng nhập và mật khẩu, sau đó nhấp vào button “ <i>Đăng nhập</i> ”

Table 2 - Use case [Sửa tài khoản] description

Use case name	Sửa tài khoản
Description	Cho phép người dùng sửa tên đăng nhập và mật khẩu
Actor	Admin & Staff
Preconditions	Nhấp vào button có hiển thị là mã nhân viên, điền tên đăng nhập và mật khẩu mới, sau đó nhấp vào button “ <i>Chỉnh sửa</i> ”

Table 3 - Use case [Xem thông tin cá nhân] description

Use case name	Xem Thông tin cá nhân
Description	Cho phép người dùng xem thông tin cá nhân
Actor	Admin & Staff
Preconditions	Nhấp vào button có hiển thị là mã nhân viên

Table 4 - Use case [Xem danh sách Khách hàng] description

Use case name	Xem DS Khách
Description	Cho phép người dùng xem danh sách khách hàng
Actor	Admin
Preconditions	Nhấp vào “ <i>Khách hàng</i> ”

Table 5 - Use case [Xem LS nhập hàng] description

Use case name	Xem LS nhập hàng
Description	Cho phép người dùng xem lịch sử nhập hàng
Actor	Admin
Preconditions	Nhấp vào “ <i>Lịch Sử Nhập hàng</i> ”

Table 6 - Use case [Xem LS thanh toán] description

Use case name	Xem LS thanh toán
Description	Cho phép người dùng xem lịch sử thanh toán
Actor	Admin
Preconditions	Nhấp vào “ <i>Lịch Sử Thanh toán</i> ”

Table 7 - Use case [Thêm nhân viên] description

Use case name	Thêm nhân viên
Description	Cho phép người dùng thêm thông tin của nhân viên vào bảng
Actor	Admin
Preconditions	Nhấp vào “Nhân viên”, sau đó nhấp vào button “Thêm”, điền thông tin nhân viên và nhấp button “Lưu”

Table 8 - Use case [Sửa nhân viên] description

Use case name	Sửa nhân viên
Description	Cho phép người dùng chỉnh sửa thông tin của nhân viên
Actor	Admin
Preconditions	Nhấp vào “Nhân viên”, sau đó nhấp vào button “Chỉnh sửa” ở dòng nhân viên cần chỉnh sửa, điền thông tin nhân viên và nhấp button “Cập nhật”

Table 9 - Use case [Xóa nhân viên] description

Use case name	Xóa nhân viên
Description	Cho phép người dùng xóa thông tin của nhân viên khỏi bảng
Actor	Admin
Preconditions	Nhấp vào “Nhân viên”, sau đó nhấp vào button “Xóa” ở dòng nhân viên cần xóa

Table 10 - Use case [Thêm sản phẩm] description

Use case name	Thêm sản phẩm
Description	Cho phép người dùng thêm thông tin của sản phẩm vào bảng
Actor	Admin
Preconditions	Nhấp vào “Sản phẩm”, sau đó nhấp vào button “Thêm”, điền thông tin sản phẩm và nhấp button “Lưu”

Table 11 - Use case [Sửa sản phẩm] description

Use case name	Sửa sản phẩm
Description	Cho phép người dùng chỉnh sửa thông tin của sản phẩm
Actor	Admin
Preconditions	Nhấp vào “Sản phẩm”, sau đó nhấp vào button “Chỉnh sửa” ở dòng sản phẩm cần chỉnh sửa, điền thông tin sản phẩm và nhấp button “Cập nhật”

Table 12 - Use case [Xóa sản phẩm] description

Use case name	Xóa sản phẩm
Description	Cho phép người dùng xóa thông tin của sản phẩm khỏi bảng
Actor	Admin
Preconditions	Nhấp vào “Sản phẩm”, sau đó nhấp vào button “Xóa” ở dòng sản phẩm cần xóa

Table 13 - Use case [Thêm loại sản phẩm] description

Use case name	Thêm loại sản phẩm
Description	Cho phép người dùng thêm thông tin của loại sản phẩm vào bảng
Actor	Admin
Preconditions	Nhấp vào “Loại sản phẩm”, sau đó nhấp vào button “Thêm”, điền thông tin loại sản phẩm và nhấp button “Lưu”

Table 14 - Use case [Sửa loại sản phẩm] description

Use case name	Sửa loại sản phẩm
Description	Cho phép người dùng chỉnh sửa thông tin của loại sản phẩm
Actor	Admin
Preconditions	Nhấp vào “Loại sản phẩm”, sau đó nhấp vào button “Chỉnh sửa” ở dòng loại sản phẩm cần chỉnh sửa, điền thông tin loại sản phẩm và nhấp button “Cập nhật”

Table 15 - Use case [Xóa loại sản phẩm] description

Use case name	Xóa loại sản phẩm
Description	Cho phép người dùng xóa thông tin của loại sản phẩm khỏi bảng
Actor	Admin
Preconditions	Nhấp vào “Loại sản phẩm”, sau đó nhấp vào button “Xóa” ở dòng loại sản phẩm cần xóa

Table 16 - Use case [Xem danh sách Sản phẩm] description

Use case name	Xem DS Sản phẩm
Description	Cho phép người dùng xem danh sách sản phẩm
Actor	Staff
Preconditions	Nhấp vào “Sản phẩm”

Table 17 - Use case [Xem danh sách Loại sản phẩm] description

Use case name	Xem DS Loại sản phẩm
Description	Cho phép người dùng xem danh sách loại sản phẩm
Actor	Staff
Preconditions	Nhấp vào “Loại sản phẩm”

Table 18 - Use case [Nhập hàng] description

Use case name	Nhập hàng
Description	Cho phép người dùng thực hiện nhập hàng
Actor	Staff
Preconditions	Nhấp vào “ <i>Nhập hàng</i> ”, chọn sản phẩm và nhập số lượng cần nhập, sau đó nhấp vào button “ <i>Nhập hàng</i> ”

Table 19 - Use case [Thanh toán] description

Use case name	Thanh toán
Description	Cho phép người dùng thực hiện chọn sản phẩm cần thanh toán và xuất hóa đơn
Actor	Staff
Preconditions	Nhấp vào “ <i>Thanh toán</i> ”, chọn sản phẩm cần thanh toán bằng cách nhấp vào button “ <i>Mua</i> ” ở sản phẩm đó, sau đó chọn số lượng. Cuối cùng là nhấp vào button “ <i>Thanh toán</i> ” và “ <i>Xuất hóa đơn</i> ”

Table 20 - Use case [Thêm khách hàng] description

Use case name	Thêm khách hàng
Description	Cho phép người dùng thêm thông tin của khách hàng vào bảng
Actor	Staff
Preconditions	Nhấp vào “ <i>Khách hàng</i> ”, sau đó nhấp vào button “ <i>Thêm</i> ”, điền thông tin khách hàng và nhấp button “ <i>Lưu</i> ”

Table 21 - Use case [Sửa khách hàng] description

Use case name	Sửa khách hàng
Description	Cho phép người dùng chỉnh sửa thông tin khách hàng
Actor	Staff
Preconditions	Nhấp vào “ <i>Khách hàng</i> ”, sau đó nhấp vào button “ <i>Chỉnh sửa</i> ” ở dòng khách hàng cần chỉnh sửa, điền thông tin khách hàng và nhấp button “ <i>Cập nhật</i> ”

Table 22 - Use case [Xóa khách hàng] description

Use case name	Xóa khách hàng
Description	Cho phép người dùng xóa thông tin của khách hàng khỏi bảng
Actor	Staff
Preconditions	Nhấp vào “ <i>Khách hàng</i> ”, sau đó nhấp vào button “ <i>Xóa</i> ” ở dòng khách hàng cần xóa

5. Bảng phân công công việc

Họ và Tên	Nhiệm vụ	Mức độ hoàn thành
Nguyễn Quốc Huy	Chức năng mua hàng thanh toán, word,	100%
Đào Hoàng Đăng	Thiết kế giao diện người dùng, Cơ sở dữ liệu	100%
Nguyễn Hoàng Lực	Chức năng CRUD thực thể, Cơ sở dữ liệu	100%

CHƯƠNG 2: THIẾT KẾ ĐỀ TÀI

1. Miêu tả quá trình

1.1. Quản lý

Quản lý sau khi đăng nhập và xác thực, sử dụng dashboard của phần Admin để theo dõi tổng quan về nhân viên, khách hàng, loại sản phẩm, sản phẩm, nhập hàng, thanh toán. Từ đó, họ quản lý danh sách nhân viên, khách hàng, sản phẩm, lịch sử nhập hàng, lịch sử thanh toán. Quản lý có thể thực hiện các thao tác cơ bản như bằng cách thêm mới, chỉnh sửa thông tin và cập nhật các danh sách theo yêu cầu.

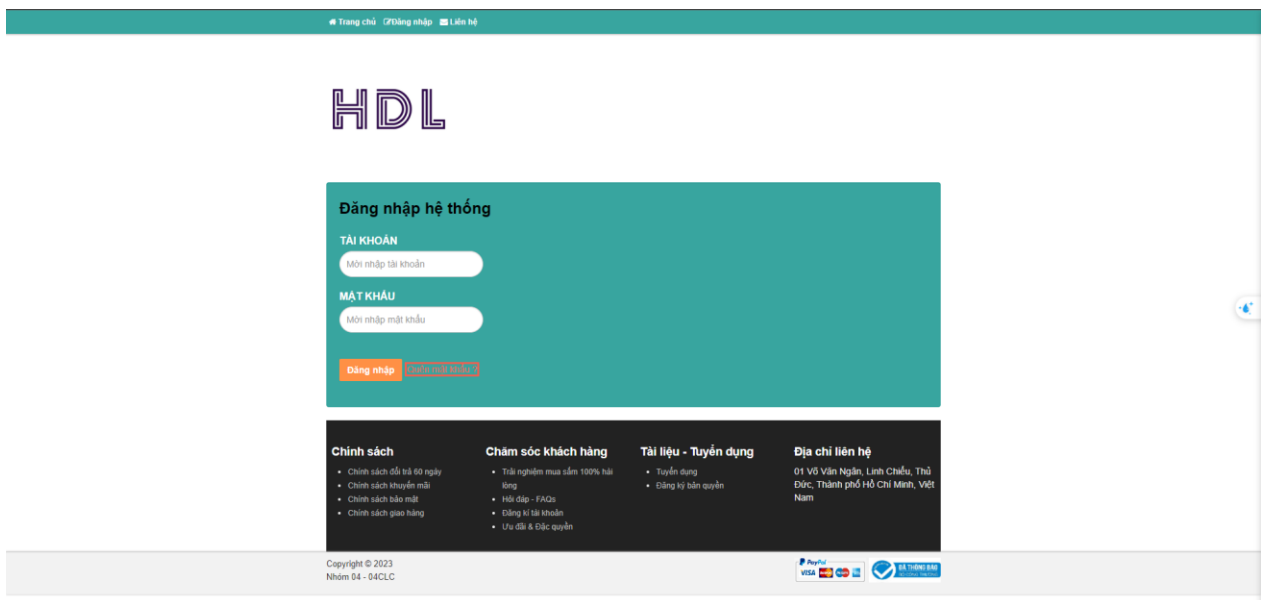
1.2. Nhân viên

Quản lý sau khi đăng nhập và xác thực, sử dụng dashboard của phần User để theo dõi tổng quan về thanh toán, sản phẩm, khách hàng và nhập hàng. Nhân viên có thể xem tổng quan về hồ sơ của bản thân và thay đổi thông tin đăng nhập của mình. Nhân viên thực hiện công việc hàng ngày của mình, bao gồm quản lý sản phẩm, xử lý đơn hàng và hóa đơn và thêm mới khách hàng khi cần thiết

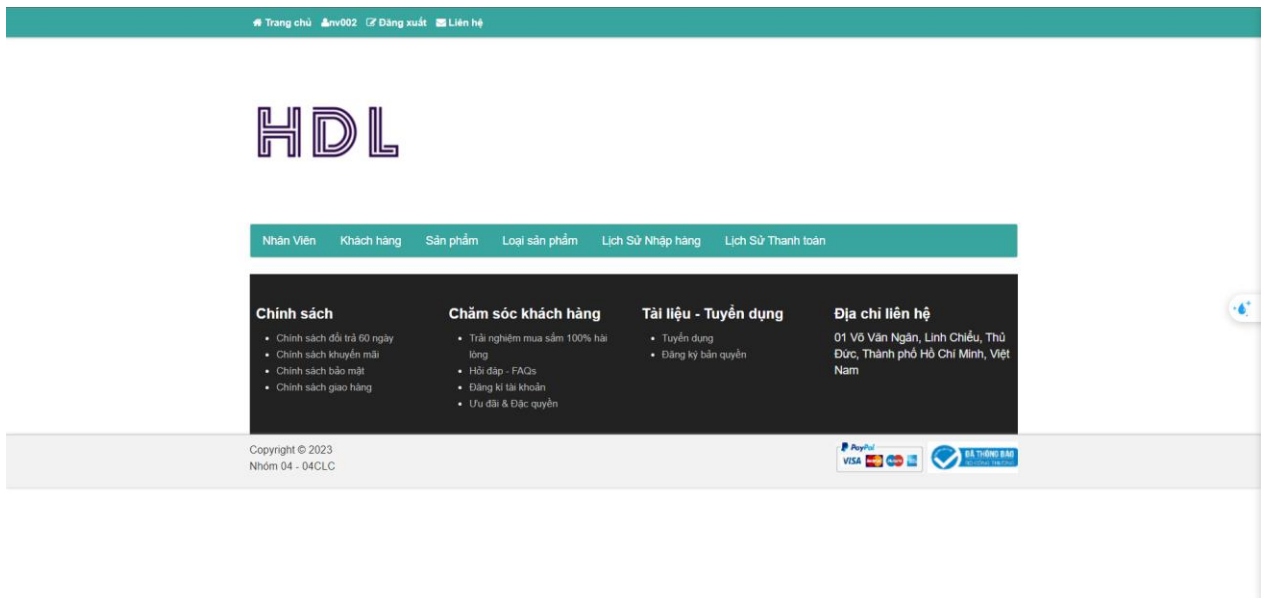
2. Các chức năng

2.1. Chức năng Đăng Nhập và Phân Quyền

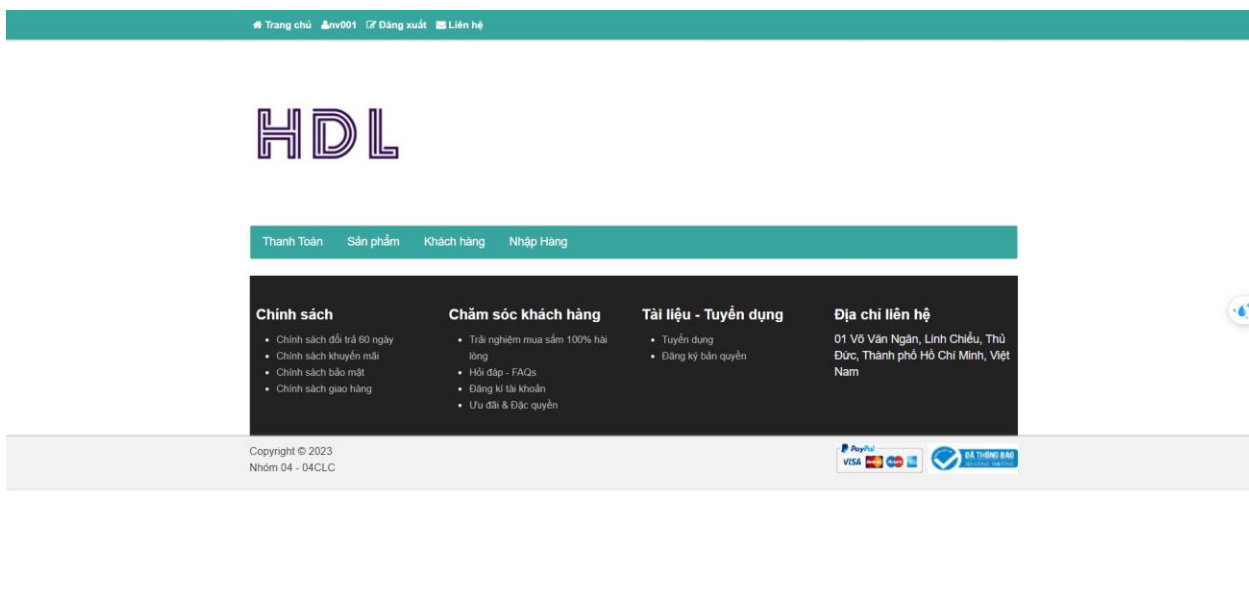
Người dùng được chào đón bởi trang đăng nhập với các trường nhập thông tin tài khoản và mật khẩu. Tiếp theo là nhập thông tin tài khoản và mật khẩu sau đó ấn nút “Đăng nhập” để truy cập vào hệ thống



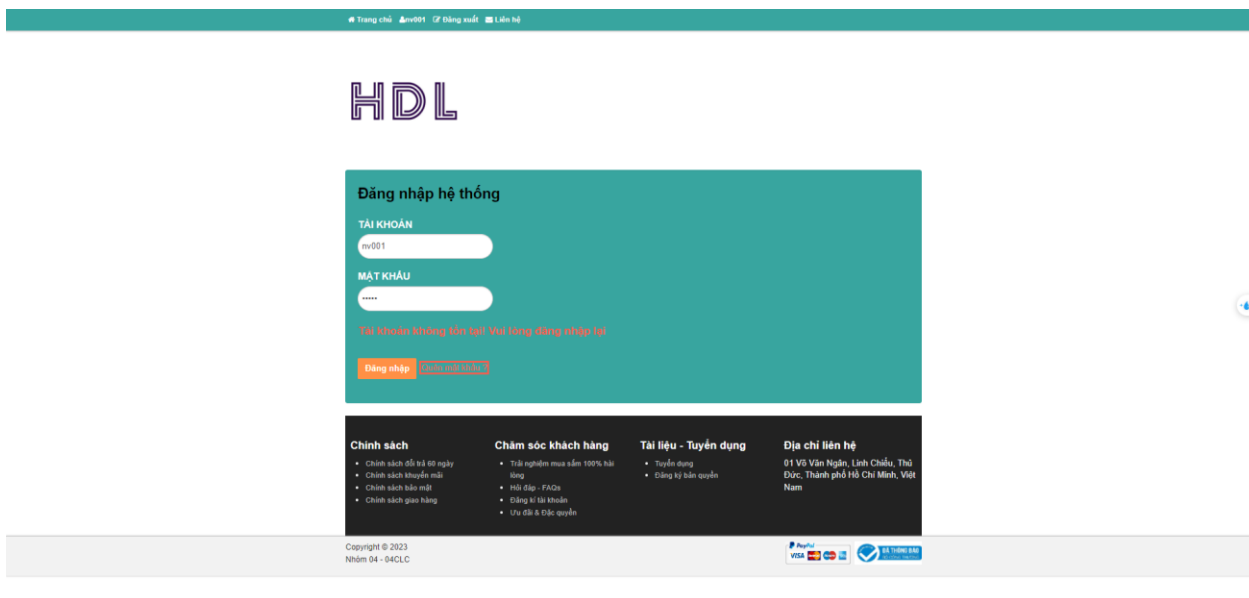
Quản lý (Admin) khi đăng nhập vào hệ thống sẽ được cấp phép truy cập vào toàn bộ các chức năng quản lý, bao gồm quản lý nhân viên, khách hàng, sản phẩm, phân loại, nhập hàng và thanh toán. Họ có khả năng thêm, sửa, xóa thông tin, cũng như quản lý tài khoản người dùng khác và gán quyền hạn cho họ.



Nhân viên (User) sau khi đăng nhập sẽ chỉ có quyền truy cập vào những chức năng liên quan đến việc xem và tương tác với sản phẩm, đặt hàng, thông tin khách hàng, nhập hàng và cập nhật thông tin cá nhân. Họ không có quyền truy cập vào các chức năng quản trị và chỉ nhìn thấy thông tin mà họ được phép xem.



Hiện thị thông báo lỗi nếu thông tin đăng nhập không chính xác.



File .jsp chức năng Đăng nhập

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@include file="/WEB-INF/views/layouts/user/taglib.jsp"%>
<head>
<title>Đăng nhập</title>
</head>
<body>
    <div class="well">
        <h5>Đăng nhập hệ thống</h5>
        <br>

        <div class="frmLogin">
            <form:form action="dang-nhap" method="POST"
modelAttribute="user">
                <div class="control-group">
                    <label class="control-label" for="inputEmail"
                        style="font-weight: bold;">TÀI
KHOẢN</label>

                    <div class="controls">
                        <form:input type="text" class="span3"
                            style="border-radius : 50px; height
: 30px;"
                            placeholder="Mời nhập tài khoản"
path="tenDangNhap" />
                    </div>
                </div>
                <div class="control-group">
                    <label class="control-label"
for="inputPassword"
                        style="font-weight: bold;">MẬT KHẨU
</label>

                    <div class="controls">
                        <form:input type="password" class="span3"
                            style="border-radius : 50px; height
: 30px;"
                            placeholder="Mời nhập mật khẩu"
path="matKhau" />
                    </div>
                </div>
                <c:if test="${not empty statusLogin}">
                    <h3 style="color: #F55E4F">${statusLogin}</h3>
                </c:if>
                <br />
                <div class="control-group">
                    <div class="controls">
                        <button type="submit"
class="defaultBtn">Đăng nhập</button>
                        <a href="<c:url value="/user/quen-mat-
khau"/>"
                            style="color: #F55E4F; border-
style: solid;">Quên mật khẩu ?</a>
                    </div>
                </div>
            </form:form>
        </div>
    </div>
</body>
```

Controller chức năng Đăng Nhập

```
@RequestMapping(value = "/user/dang-nhap", method = RequestMethod.POST)
public String login(@ModelAttribute("user") TaiKhoanDto user, Model model,
    HttpServletRequest request) {
    String taiKhoan = user.gettenDangNhap();
    String matKhau = user.getmatKhau();

    List<TaiKhoanDto> dsTaiKhoanNhanVien = taiKhoanDao.getAllNhanVienTaiKhoan();
    Optional<TaiKhoanDto> matchedUser = dsTaiKhoanNhanVien.stream()
        .filter(taiKhoanDto -> Objects.equals(taiKhoan,
            taiKhoanDto.gettenDangNhap())
            && Objects.equals(matKhau,
            taiKhoanDto.getmatKhau()))
        .findFirst();

    if (matchedUser.isPresent()) {
        TaiKhoanDto loginInfo = matchedUser.get(); // Sử dụng matchedUser
        HttpSession session = request.getSession();
        session.setAttribute("LoginInfo", loginInfo);
        session.setAttribute("statusLogin", null);

        session.setAttribute("maNhanVien", loginInfo.getmaNhanVien());
        Serializable maNhanVien = loginInfo.getmaNhanVien();
        manv = loginInfo.getmaNhanVien();
        NhanVienDto nhanVienDangNhap = nhanvienDao.getById(maNhanVien);
        String chucVu = nhanVienDangNhap.getchucVu();

        if ("Staff".equals(chucVu)) {
            return "redirect:/user/";
        } else {
            return "redirect:/admin/";
        }
    } else {
        HttpSession session = request.getSession();
        session.setAttribute("statusLogin", "Tài khoản không tồn tại! Vui lòng
        đăng nhập lại");
        return "user/account/signin";
    }
}

@RequestMapping(value="/dang-xuat",method= RequestMethod.GET)
public String Login(HttpSession session, HttpServletRequest request) {
    session.removeAttribute("LoginInfo");
    return "redirect:/user/dang-nhap";
}

@RequestMapping(value = "/editInfo/{maNhanVien}", method = RequestMethod.GET)
public String editInfo(@PathVariable("maNhanVien") String maNhanVien, Model model) {
    TaiKhoanDto taiKhoan = accountService.GetInfoByID(maNhanVien);
    model.addAttribute("taiKhoan", taiKhoan);
    return "user/account/editinfo";
}

@RequestMapping(value= "/update-info", method = RequestMethod.POST)
public String update(@RequestParam("maNhanVien") String maNhanVien,
    @RequestParam("tenDangNhap") String tenDangNhap,
    @RequestParam("matKhau") String matKhau) {
    TaiKhoanDto taiKhoanDto = taiKhoanDao.getById(maNhanVien);

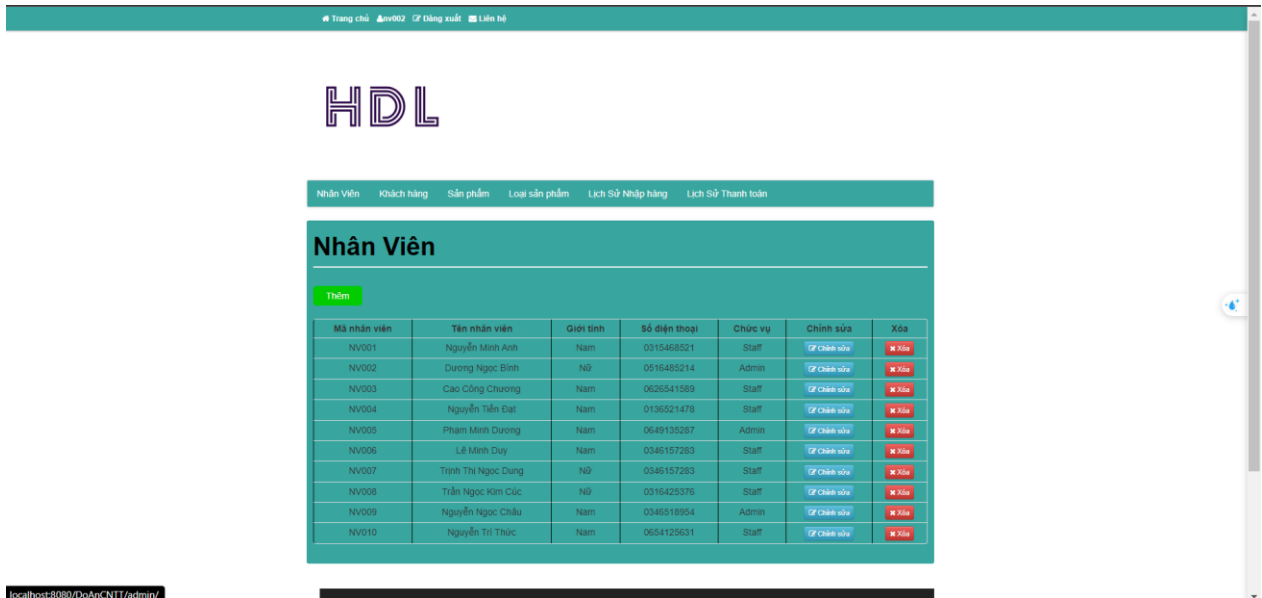
    taiKhoanDto.settenDangNhap(tenDangNhap);
    taiKhoanDto.setmatKhau(matKhau);
    taiKhoanDao.update(taiKhoanDto);
    return "redirect:/thong-tin";
}

@RequestMapping(value = "/user/quen-mat-khau", method = RequestMethod.GET)
public String forgotPass() {
    return "user/account/forgotpassword";
}
```

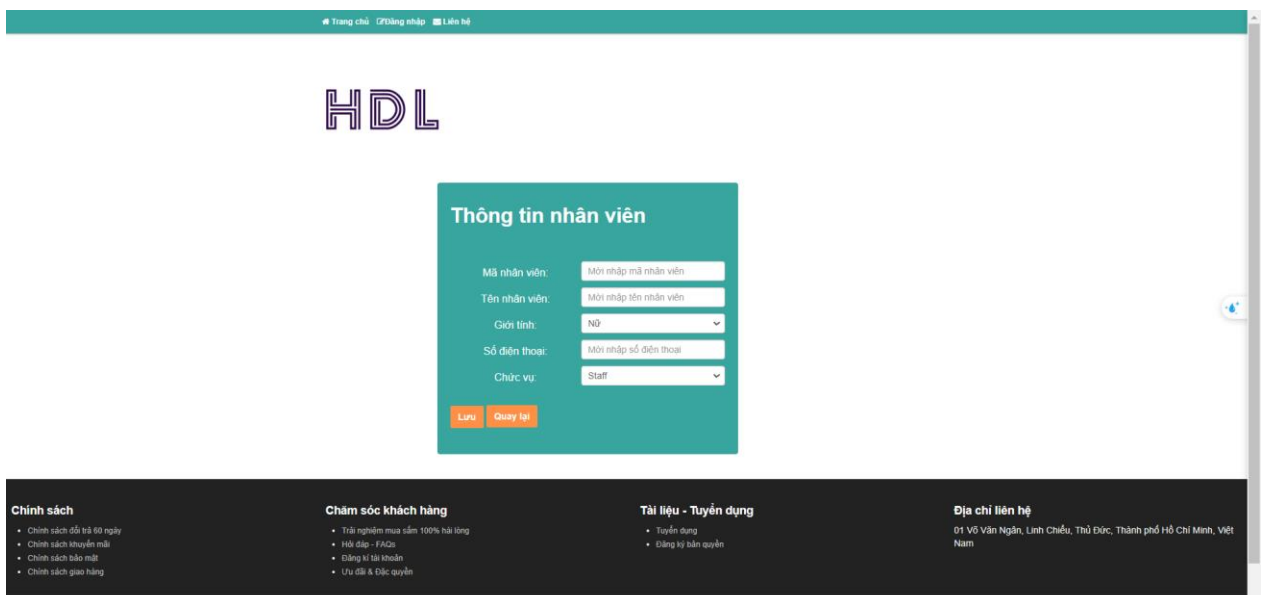
2.2. Các chức năng của Quản lý

2.2.1. Quản lý Nhân Viên

Quản lý có thể xem danh sách toàn bộ nhân viên cũng như việc thêm và chỉnh sửa cũng như xóa thông tin nhân viên.



Nếu muốn thêm nhân viên mới thì ấn vào nút “Thêm” và điền các thông tin vào các trường và ấn nút “Lưu” nếu muốn thoát ấn nút “Quay lại”



Nếu muốn chỉnh sửa thông tin nhân viên thì ấn vào nút “Chỉnh sửa” và điền các thông tin vào các trường và ấn nút “Cập nhật” nếu muốn thoát ấn nút “Quay lại”

The screenshot shows a web browser window with the URL `localhost:8080/DoAnCNTT/admin/`. The page has a teal header with links for "Trang chủ", "Đăng nhập", and "Liên hệ". The main content area features the "HDL" logo and a central form titled "Sửa thông tin nhân viên". The form contains the following fields:

- Mã nhân viên: NV001
- Tên nhân viên: Nguyễn Minh Anh
- Giới tính: Nam (dropdown menu)
- Số điện thoại: 0315468521
- Chức vụ: Staff

At the bottom of the form are two buttons: "Cập nhật" (orange) and "Quay lại" (orange). The footer is dark grey and contains four columns of links:

- Chính sách**
 - Chính sách đổi trả 60 ngày
 - Chính sách chuyển đổi
 - Chính sách bảo mật
 - Chính sách giao hàng
- Chăm sóc khách hàng**
 - Trải nghiệm mua sắm 100% hài lòng
 - Hỏi đáp - FAQs
 - Đăng ký tài khoản
 - Ưu đãi & Đặc quyền
- Tài liệu - Tuyển dụng**
 - Tuyển dụng
 - Đăng ký bản quyền
- Địa chỉ liên hệ**
 - 01 Võ Văn Ngân, Linh Chiểu, Thủ Đức, Thành phố Hồ Chí Minh, Việt Nam

Nếu muốn xóa nhân viên thì ấn vào nút “Xóa”

File jsp Quản lý Nhân Viên

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@include file="/WEB-INF/views/layouts/user/taglib.jsp"%>
<head>
<title>Nhân Viên</title>
</head>
<body>
    <div class="row">
        <div class="navbar">

            <div class="well well-small">
                <h1>Nhân Viên</h1>
                <hr class="soften" />
                <br> <a href="<c:url value="/them-nhan-vien"/>" class="addBtn">Thêm</a>
                <br> <br>

                <table class="table table-bordered table-condensed">
                    <thead>
                        <tr>
                            <th>Mã nhân viên</th>
                            <th>Tên nhân viên</th>
                            <th>Giới tính</th>
                            <th>Số điện thoại</th>
                            <th>Chức vụ</th>
                            <th>Chỉnh sửa</th>
                            <th>Xóa</th>
                        </tr>
                    </thead>
                    <tbody>
                        <c:forEach var="item" items="{ nhanvien }">
                            <tr>
                                <td>${ item.maNhanVien }</td>
                                <td>${ item.tenNhanVien }</td>
                                <td><c:if test="{ item.gioiTinh == 1 }">
                                    Nam
                                </c:if> <c:if test="{ item.gioiTinh == 0 }">
                                    Nữ
                                </c:if></td>
                                <td>${ item.sdt }</td>
                                <td>${ item.chucVu }</td>
                                <td><a
                                    href="<c:url
                                        value='/editemployee/${item.maNhanVien}' />"
                                    class="btn btn-mini btn-info
                                        class="icon-edit"></span>
                                        </a></td>
                                <td><a
                                    href="<c:url
                                        value='/deleteEmployee/${item.maNhanVien}' />"
                                    class="btn btn-mini btn-danger
                                        class="icon-remove"></span>
                                        </a></td>
                            </tr>
                        </c:forEach>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</body>
```

Controller Quản lý Nhân Viên

```
@Controller
public class NhanVienController extends BaseController{
    @Autowired
    NhanVienServiceImpl nhanvienService = new NhanVienServiceImpl();

    @Autowired
    NhanVienDao nhanvienDao;

    @RequestMapping(value = "/admin/nhan-vien")
    public ModelAndView Index() {
        _mvShare.addObject("nhanvien", _homeService.GetDataEmployees());
        _mvShare.setViewName("admin/employees/employee");
        return _mvShare;
    }

    @RequestMapping(value="them-nhan-vien",method= RequestMethod.GET)
    public ModelAndView AddCreateEmployee() {
        _mvShare.setViewName("admin/employees/addemployee");
        _mvShare.addObject("nhanvien", new NhanVien());
        _mvShare.addObject("getnhanvien", _homeService.GetDataEmployees());
        return _mvShare;
    }

    @RequestMapping(value="them-nhan-vien",method= RequestMethod.POST)
    public String CreateEmployee(HttpServletRequest request,
    @ModelAttribute("nhanvien") NhanVien nhanvien) {
        HttpSession session = request.getSession();
        int count = nhanvienService.AddEmployee(nhanvien);
        if(count > 0) {
            session.setAttribute("status", "Thêm thành công!");
        }
        else {
            session.setAttribute("status", "Thêm thất bại!");
        }
        return "redirect:" +request.getHeader("Referer");
    }

    @RequestMapping(value = "/editemployee/{maNhanVien}", method =
    RequestMethod.GET)
    public String editEmployee(@PathVariable("maNhanVien") String maNhanVien, Model
    model) {
        NhanVienDto employee = nhanvienService.GetEmployeeById(maNhanVien);
        model.addAttribute("nhanVien", employee);
        return "admin/employees/editemployee";
    }

    @RequestMapping(value= "/update-employee", method = RequestMethod.POST)
    public String update(@RequestParam("maNhanVien") String maNhanVien,
        @RequestParam("tenNhanVien") String tenNhanVien,
        @RequestParam("gioiTinh") int gioiTinh,
        @RequestParam("sdt") String sdt) {
        NhanVienDto nhanvienDto = nhanvienDao.getById(maNhanVien);
        if (nhanvienDto == null) {
            return "redirect:/error-page";
        }
        nhanvienDto.settenNhanVien(tenNhanVien);
        nhanvienDto.setgioiTinh(gioiTinh);
        nhanvienDto.setsdt(sdt);
        nhanvienDao.update(nhanvienDto);
        return "redirect:/admin/nhan-vien";
    }
}
```


2.2.2. Quản lý Khách Hàng

Quản lý có thể xem danh sách toàn bộ khách hàng cũng như thông tin liên lạc của họ

[Trang chủ](#) [Đăng xuất](#) [Liên hệ](#)

HDL

[Nhấn Viên](#) [Khách hàng](#) [Sản phẩm](#) [Loại sản phẩm](#) [Lịch Sử Nhập hàng](#) [Lịch Sử Thanh toán](#)

Khách hàng

Mã khách hàng	Tên khách hàng	Số điện thoại
KH001	Nguyễn Minh Kha	0541256145
KH002	Đỗ Kim Ngọc	094613527
KH003	Nguyễn Hoàng Lục	0825850092
KH004	Nguyễn Quốc Huy	0842955519

Chính sách

- Chính sách đổi trả 60 ngày
- Chính sách khuyến mãi
- Chính sách bảo mật
- Chính sách giao hàng

Chăm sóc khách hàng

- Trải nghiệm mua sắm 100% hài lòng
- Hỏi đáp - FAQs
- Đăng kí tài khoản
- Ưu đãi & Đặc quyền






Tài liệu - Tuyển dụng

- Tuyển dụng
- Đăng kí bản quyền

Địa chỉ liên hệ

01 Võ Văn Ngân, Linh Chiểu, Thủ Đức, Thành phố Hồ Chí Minh, Việt Nam

Copyright © 2023
Nhóm 04 - 04CLC



File jsp Quản lý Khách Hàng

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@include file="/WEB-INF/views/layouts/user/taglib.jsp"%>
<head>
<title>Khách hàng</title>
</head>
<body>
    <div class="row">
        <div class="navbar">

            <div class="well well-small">
                <h1>Khách hàng</h1>
                <hr class="soften" />
                <br>

                <table class="table table-bordered table-condensed">
                    <thead>
                        <tr>
                            <th>Mã khách hàng</th>
                            <th>Tên khách hàng</th>
                            <th>Số điện thoại</th>
                        </tr>
                    </thead>
                    <tbody>

                        <c:forEach var="item" items="{ khachhang }">
                            <tr>
                                <td>${ item.maKhachHang }</td>
                                <td>${ item.tenKhachHang }</td>
                                <td>${ item.sdt }</td>
                            </tr>
                        </c:forEach>
                    </tbody>
                </table>
            </div>
        </div>
    </body>
```

Controller Quản lý Khách Hàng

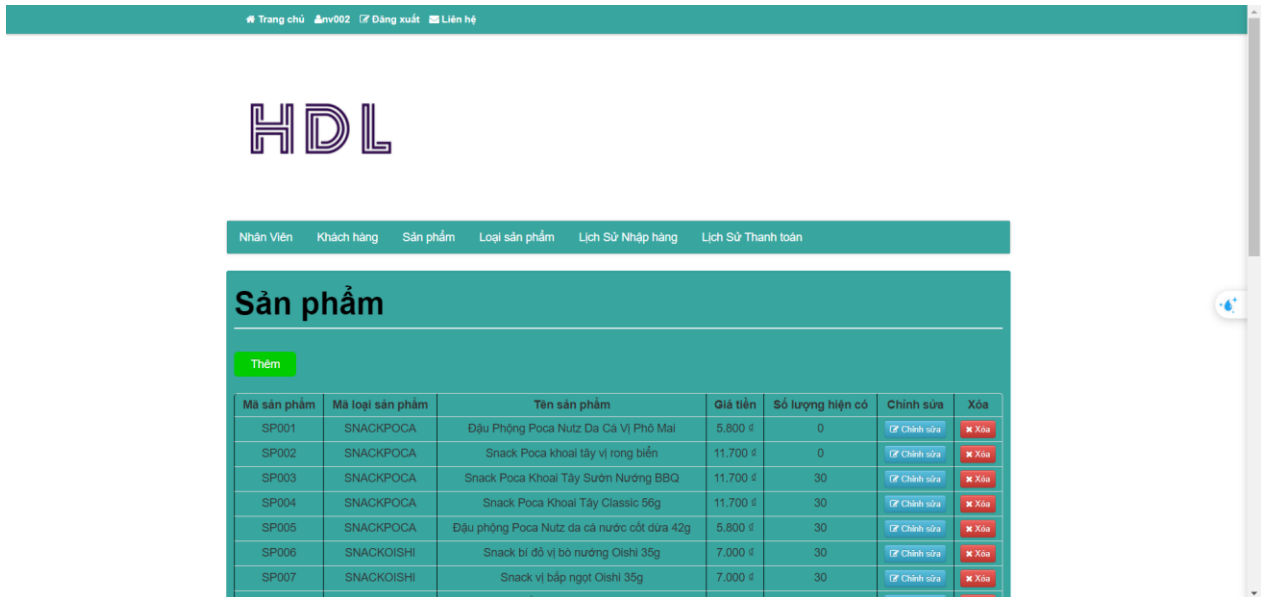
```
@Autowired
KhachHangServiceImpl khachhangService = new KhachHangServiceImpl();

@Autowired
KhachHangDao khachhangDao;

@RequestMapping(value="them-khach-hang",method= RequestMethod.GET)
public ModelAndView AddCreateCustomer() {
    _mvShare.setViewName("user/customers/addcustomer");
    _mvShare.addObject("khachhang", new KhachHang());
    return _mvShare;
}
```

2.2.3. Quản lý Sản Phẩm

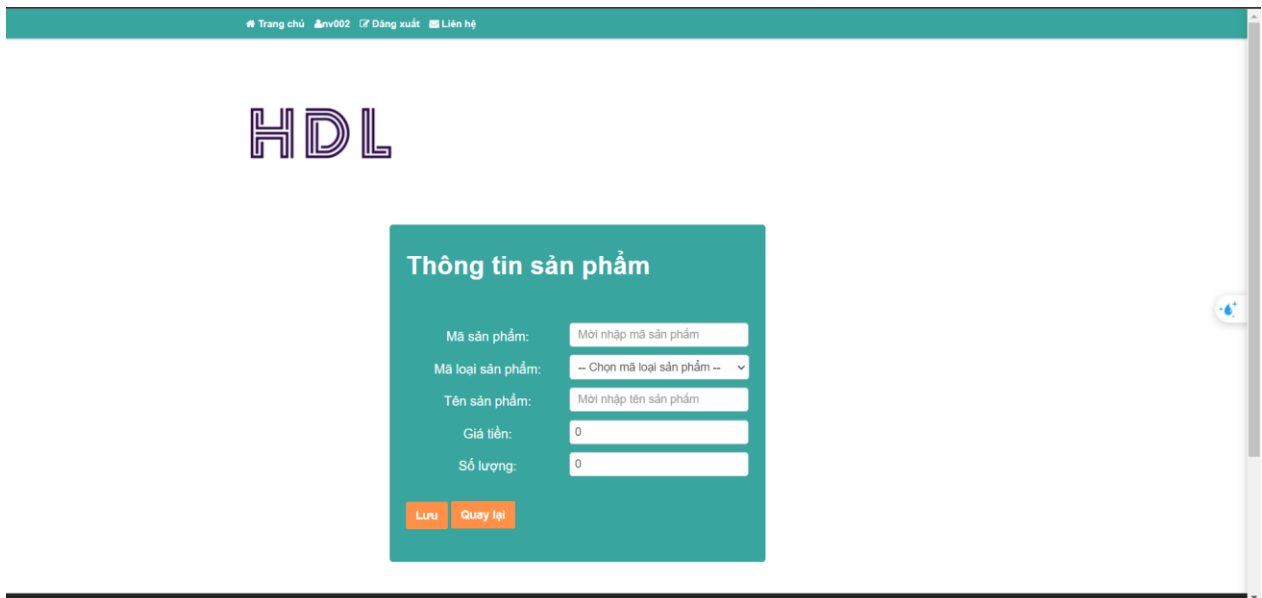
Quản lý có thể xem danh sách toàn bộ sản phẩm cũng như việc thêm và chỉnh sửa cũng như xóa thông tin sản phẩm



The screenshot shows the HDL product management interface. At the top, there is a navigation bar with links: Trang chủ, nv002, Đăng xuất, and Liên hệ. Below the navigation bar is the HDL logo. A horizontal menu contains links: Nhân Viên, Khách hàng, Sản phẩm, Loại sản phẩm, Lịch Sử Nhập hàng, and Lịch Sử Thanh toán. The main section is titled "Sản phẩm" and features a green "Thêm" button. Below this is a table listing products with columns for Mã sản phẩm, Mã loại sản phẩm, Tên sản phẩm, Giá tiền, Số lượng hiện có, Chỉnh sửa, and Xóa.

Mã sản phẩm	Mã loại sản phẩm	Tên sản phẩm	Giá tiền	Số lượng hiện có	Chỉnh sửa	Xóa
SP001	SNACKPOCA	Đậu Phộng Poca Nutz Da Cà Vị Phô Mai	5.800 đ	0	Chỉnh sửa	Xóa
SP002	SNACKPOCA	Snack Poca khoai tây vị rong biển	11.700 đ	0	Chỉnh sửa	Xóa
SP003	SNACKPOCA	Snack Poca Khoai Tây Sườn Nướng BBQ	11.700 đ	30	Chỉnh sửa	Xóa
SP004	SNACKPOCA	Snack Poca Khoai Tây Classic 56g	11.700 đ	30	Chỉnh sửa	Xóa
SP005	SNACKPOCA	Đậu phộng Poca Nutz da cá nước cốt dừa 42g	5.800 đ	30	Chỉnh sửa	Xóa
SP006	SNACKOISHI	Snack bí đỏ vị bơ nướng Olshi 35g	7.000 đ	30	Chỉnh sửa	Xóa
SP007	SNACKOISHI	Snack vị bắp ngọt Olshi 35g	7.000 đ	30	Chỉnh sửa	Xóa

Nếu muốn thêm nhân viên mới thì ấn vào nút “Thêm” và điền các thông tin vào các trường và ấn nút “Lưu” nếu muốn thoát ấn nút “Quay lại”



The screenshot shows the HDL product management interface with the "Thông tin sản phẩm" form. The form contains fields for Mã sản phẩm, Mã loại sản phẩm, Tên sản phẩm, Giá tiền, and Số lượng, each with a corresponding input field. At the bottom of the form are two buttons: "Lưu" and "Quay lại".

Thông tin sản phẩm	
Mã sản phẩm:	<input type="text" value="Mời nhập mã sản phẩm"/>
Mã loại sản phẩm:	<input type="text" value="-- Chọn mã loại sản phẩm --"/>
Tên sản phẩm:	<input type="text" value="Mời nhập tên sản phẩm"/>
Giá tiền:	<input type="text" value="0"/>
Số lượng:	<input type="text" value="0"/>
<input type="button" value="Lưu"/> <input type="button" value="Quay lại"/>	

Nếu muốn chỉnh sửa thông tin nhân viên thì ấn vào nút “Chỉnh sửa” và điền các thông tin vào các trường và ấn nút “Cập nhật” nếu muốn thoát ấn nút “Quay lại”

The screenshot shows a web browser window with the HDL logo at the top. Below the logo is a teal-colored form titled "Sửa thông tin sản phẩm" (Edit product information). The form contains the following fields:

- Mã sản phẩm: SP001
- Mã loại sản phẩm: SNACKPOCA
- Tên sản phẩm: Đậu Phộng Poca Nutz Đa Cà Vê
- Giá tiền: 5800

At the bottom of the form are two buttons: "Cập Nhật" (Update) and "Quay lại" (Go back). The browser's address bar shows the URL "http://localhost:8080/". The footer of the website contains the text "Chăm sóc khách hàng", "Tài liệu - Tuyển dụng", and "Địa chỉ liên hệ".

Nếu muốn xóa sản phẩm ấn nút “Xóa”

File jsp Quản lý Sản Phẩm

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@include file="/WEB-INF/views/layouts/user/taglib.jsp"%>
<head>
<title>Sản phẩm</title>
</head>
<body>
    <div class="row">

        <div class="well well-small">
            <h1>Sản phẩm</h1>
            <hr class="soften" />
            <br> <a href="<c:url value="/them-san-pham"/>"
class="addBtn">Thêm</a>
            <br> <br>

            <table class="table table-bordered table-condensed">
                <thead>
                    <tr>
                        <th>Mã sản phẩm</th>
                        <th>Mã loại sản phẩm</th>
                        <th>Tên sản phẩm</th>
                        <th>Giá tiền</th>
                        <th>Số lượng hiện có</th>
                        <th>Chỉnh sửa</th>
                        <th>Xóa</th>
                    </tr>
                </thead>
                <tbody>

                    <c:forEach var="item" items="${ sanpham }">
                        <tr>
                            <td>${ item.maSanPham }</td>
                            <td>${ item.maLoaiSanPham }</td>
                            <td>${ item.tenSanPham }</td>
                            <td>${ fmt:formatNumber type="number"
                                value="${ item.giaTien }"
                                /> đ</td>
                            <td>${ item.soLuongHienCo }</td>
                            <td><a
                                href="<c:url
                                value='/editProduct/${item.maSanPham}' />"
                                type="button"> <span
                                    class="btn btn-mini btn-info"
                                    class="icon-edit"></span>
                                Chỉnh sửa
                                </a></td>
                            <td><a
                                href="<c:url
                                value='/deleteProduct/${item.maSanPham}' />"
                                type="button"> <span
                                    class="btn btn-mini btn-danger"
                                    class="icon-
                                    remove"></span> Xóa
                                </a></td>
                        </tr>
                    </c:forEach>
                </tbody>
            </table>
        </div>
    </div>
</body>
```

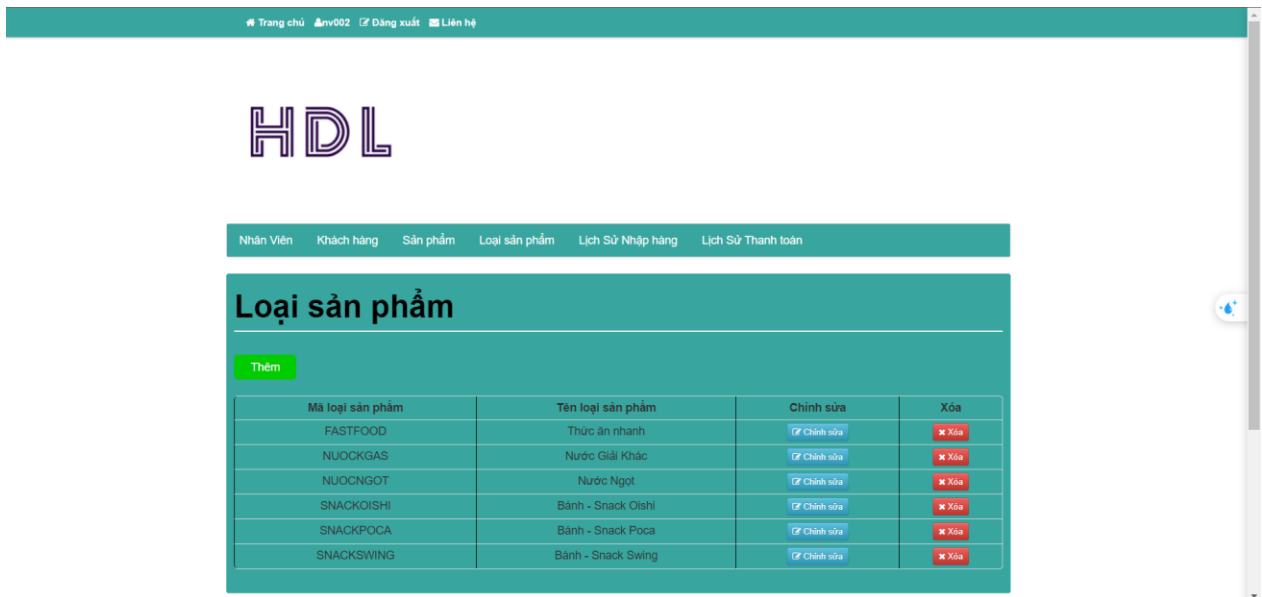
Controller Quản lý Sản Phẩm

```
@Controller
public class SanPhamController extends BaseController {
    @Autowired
    SanPhamServiceImpl sanphamService = new SanPhamServiceImpl();
    @Autowired
    LoaiSanPhamServiceImpl loaisanphamService = new LoaiSanPhamServiceImpl();
    @Autowired
    SanPhamDao sanphamDao;

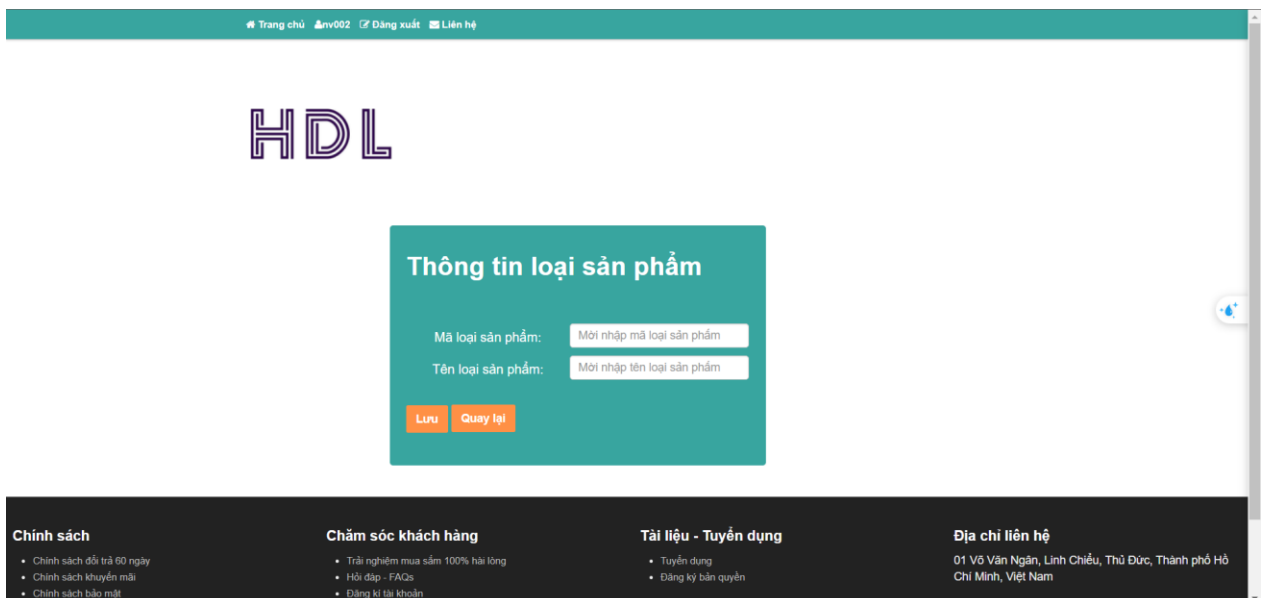
    @RequestMapping(value = "/admin/san-pham")
    public ModelAndView AdminIndex() {
        _mvShare.addObject("sanpham", _homeService.GetDataProducts());
        _mvShare.setViewName("admin/products/product");
        return _mvShare;
    }
    @RequestMapping(value = "/user/xem-san-pham")
    public ModelAndView UserIndex() {
        _mvShare.addObject("sanpham", _homeService.GetDataProducts());
        _mvShare.setViewName("user/products/product");
        return _mvShare;
    }
    @RequestMapping(value = "them-san-pham", method = RequestMethod.GET)
    public ModelAndView AddProduct() {
        _mvShare.setViewName("admin/products/addproduct");
        _mvShare.addObject("sanpham", new SanPham());
        _mvShare.addObject("loaisanpham", _homeService.GetDataLoaiSanPham());
        return _mvShare;
    }
    @RequestMapping(value="them-san-pham",method= RequestMethod.POST)
    public String CreateProduct(HttpServletRequest request, @ModelAttribute("loaisanpham")
    SanPham sanpham) {
        HttpSession session = request.getSession();
        int count = sanphamService.AddProduct(sanpham);
        if(count > 0) {
            session.setAttribute("status", "Thêm thành công!");
        }
        else {
            session.setAttribute("status", "Thêm thất bại!");
        }
        return "redirect:" + request.getHeader("Referer");
    }
    @RequestMapping(value = "/editProduct/{maSanPham}", method = RequestMethod.GET)
    public String editProduct(@PathVariable("maSanPham") String maSanPham, Model model) {
        SanPhamDto product = sanphamService.GetProductByID(maSanPham);
        List<LoaiSanPhamDto> producttype = loaisanphamService.GetDataLoaiSanPham();
        model.addAttribute("sanPham", product);
        model.addAttribute("loaisanpham", producttype);
        return "admin/products/editProduct";
    }
    @RequestMapping(value= "/update-product", method = RequestMethod.POST)
    public String update(@RequestParam("maSanPham") String maSanPham,
        @RequestParam("maLoaiSanPham") String maLoaiSanPham,
        @RequestParam("tenSanPham") String tenSanPham,
        @RequestParam("giaTien") int giaTien, SanPhamDto sp) {
        SanPhamDto sanPhamDto = sanphamDao.getById(maSanPham);
        if (sanPhamDto == null) {
            return "redirect:/error-page";
        }
        sanPhamDto.setmaLoaiSanPham(maLoaiSanPham);
        sanPhamDto.settenSanPham(tenSanPham);
        sanPhamDto.setgiaTien(giaTien);
        sanphamDao.update(sanPhamDto);
        return "redirect:/admin/san-pham";
    }
}
```

2.2.4. Quản lý Phân Loại Sản Phẩm

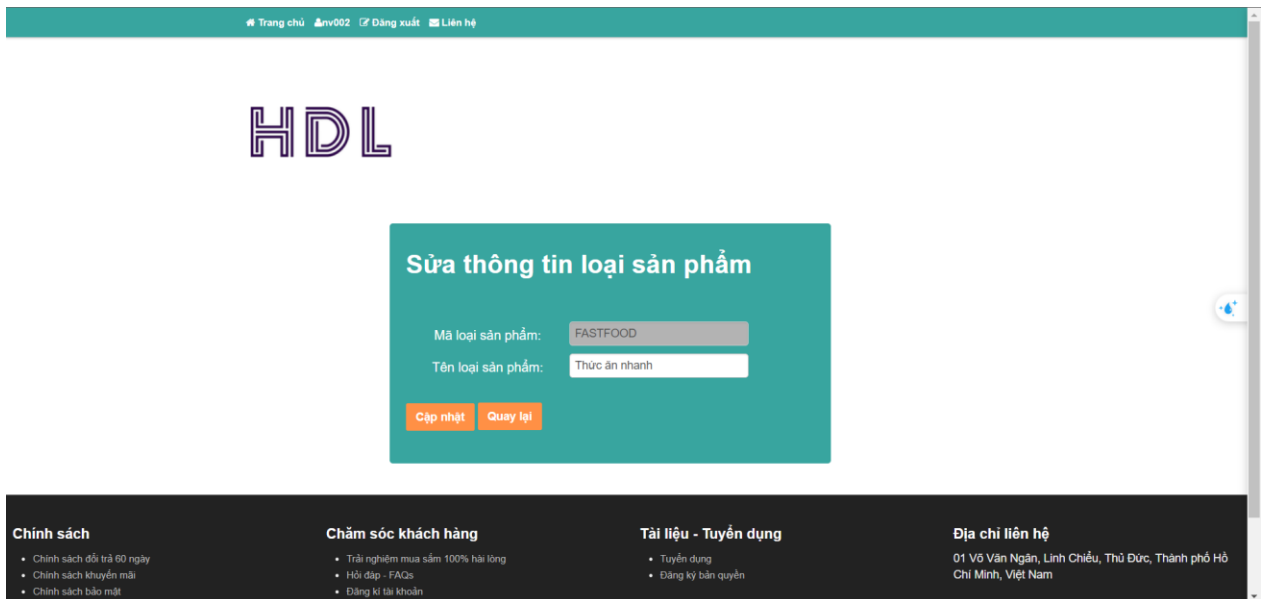
Quản lý có thể xem danh sách toàn bộ phân loại sản phẩm cũng như việc thêm và chỉnh sửa cũng như xóa thông tin sản phẩm



Nếu muốn thêm loại sản phẩm mới thì ấn vào nút “Thêm” và điền các thông tin vào các trường và ấn nút “Lưu” nếu muốn thoát ấn nút “Quay lại”



Nếu muốn chỉnh sửa thông tin loại sản phẩm thì ấn vào nút “Chỉnh sửa” và điền các thông tin vào các trường và ấn nút “Cập nhật” nếu muốn thoát ấn nút “Quay lại”



The screenshot shows a web browser window with the HDL logo at the top. Below the logo is a teal-colored form titled "Sửa thông tin loại sản phẩm". The form contains two input fields: "Mã loại sản phẩm:" with the value "FASTFOOD" and "Tên loại sản phẩm:" with the value "Thức ăn nhanh". At the bottom of the form are two buttons: "Cập nhật" (Update) and "Quay lại" (Back). The footer of the website is dark gray and contains four columns of links: "Chính sách" (Policy), "Chăm sóc khách hàng" (Customer Care), "Tài liệu - Tuyển dụng" (Documents - Recruitment), and "Địa chỉ liên hệ" (Contact Address).

Trang chủ nv002 Đăng xuất Liên hệ

HDL

Sửa thông tin loại sản phẩm

Mã loại sản phẩm: FASTFOOD

Tên loại sản phẩm: Thức ăn nhanh

Cập nhật Quay lại

Chính sách

- Chính sách đổi trả 60 ngày
- Chính sách khuyến mãi
- Chính sách bảo mật

Chăm sóc khách hàng

- Trải nghiệm mua sắm 100% hài lòng
- Hỏi đáp - FAQs
- Đăng kí tài khoản

Tài liệu - Tuyển dụng

- Tuyển dụng
- Đăng ký bản quyền

Địa chỉ liên hệ

01 Võ Văn Ngân, Linh Chiểu, Thủ Đức, Thành phố Hồ Chí Minh, Việt Nam

Nếu muốn xóa sản phẩm ấn nút “Xóa”

File jsp Quản lý Phân Loại Sản Phẩm

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@include file="/WEB-INF/views/layouts/user/taglib.jsp"%>
<head>
<title>Loại Sản phẩm</title>
</head>
<body>
    <div class="row">
        <div class="navbar">

            <div class="well well-small">
                <h1>Loại sản phẩm</h1>
                <hr class="soften" />
                <br> <a href="<c:url value="/them-loai-san-pham"/>"
                    class="addBtn">Thêm</a> <br> <br>

                <table class="table table-bordered table-condensed">
                    <thead>
                        <tr>
                            <th>Mã loại sản phẩm</th>
                            <th>Tên loại sản phẩm</th>
                            <th>Chỉnh sửa</th>
                            <th>Xóa</th>
                        </tr>
                    </thead>
                    <tbody>

                        <c:forEach var="item" items="${ loaisanpham }">
                            <tr>
                                <td>${ item.maLoaiSanPham }</td>
                                <td>${ item.tenLoaiSanPham }</td>
                                <td><a
                                    href="<c:url
value='/editProductType/${item.maLoaiSanPham}']/>"
                                    class="btn btn-mini btn-info"
type="button"> <span
                                    class="icon-
edit"></span> Chỉnh sửa
                                </a></td>
                                <td><a
                                    href="<c:url
value='/deleteProductType/${item.maLoaiSanPham}']/>"
                                    class="btn btn-mini btn-
danger" type="button"> <span
                                    class="icon-
remove"></span> Xóa
                                </a></td>
                            </tr>
                        </c:forEach>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</body>
```

Controller Quản lý Phân Loại Sản Phẩm

```
@Controller
public class LoaiSanPhamController extends BaseController{
    @Autowired
    LoaiSanPhamServiceImpl loaisanphamService = new LoaiSanPhamServiceImpl();

    @Autowired
    LoaiSanPhamDao loaisanphamDao;

    @RequestMapping(value = "/admin/loai-san-pham")
    public ModelAndView Index() {
        _mvShare.addObject("loaisanpham", _homeService.GetDataLoaiSanPham());
        _mvShare.setViewName("admin/products/loaiSanPham");
        return _mvShare;
    }

    @RequestMapping(value="them-loai-san-pham",method= RequestMethod.GET)
    public ModelAndView AddProductType() {
        _mvShare.setViewName("admin/products/addproducttype");
        _mvShare.addObject("loaisanpham", new LoaiSanPham());
        return _mvShare;
    }

    @RequestMapping(value="them-loai-san-pham",method= RequestMethod.POST)
    public String CreateProductType(HttpServletRequest request,
    @ModelAttribute("loaisanpham") LoaiSanPham loaisanpham) {
        HttpSession session = request.getSession();
        int count = loaisanphamService.AddProductType(loaisanpham);
        if(count > 0) {
            session.setAttribute("status", "Thêm thành công!");
        }
        else {
            session.setAttribute("status", "Thêm thất bại!");
        }
        return "redirect:" +request.getHeader("Referer");
    }

    @RequestMapping(value = "/editProductType/{maLoaiSanPham}", method =
    RequestMethod.GET)
    public String editProductType(@PathVariable("maLoaiSanPham") String
    maLoaiSanPham, Model model) {
        LoaiSanPhamDto producttype =
        loaisanphamService.GetProductTypeById(maLoaiSanPham);
        model.addAttribute("loaiSanPham", producttype);
        return "admin/products/editproducttype";
    }

    @RequestMapping(value= "/update-producttype", method = RequestMethod.POST)
    public String update(@RequestParam("maLoaiSanPham") String maLoaiSanPham,
        @RequestParam("tenLoaiSanPham") String tenLoaiSanPham) {
        LoaiSanPhamDto loaiSanPhamDto = loaisanphamDao.getById(maLoaiSanPham);
        if (loaiSanPhamDto == null) {
            return "redirect:/error-page";
        }
        loaiSanPhamDto.setTenLoaiSanPham(tenLoaiSanPham);
        loaisanphamDao.update(loaiSanPhamDto);
        return "redirect:/admin/loai-san-pham";
    }
}
```

2.2.5. Quản lý Hóa Đơn Nhập Hàng

Quản lý có thể xem danh sách toàn bộ hóa đơn nhập hàng để dễ dàng kiểm soát số lượng mỗi sản phẩm cũng như quản lý chi tiêu cho cửa hàng một cách hợp lý

Trang chủ nv002 Đăng xuất Liên hệ

HDL

Nhân ViênKhách hàngSản phẩmLoại sản phẩmLịch Sử Nhập hàngLịch Sử Thanh toán

Lịch Sử Nhập Hàng

Mã Nhân Viên	Mã Sản Phẩm	Ngày Nhập Hàng	Số Lượng
NV001	SP001	2023-11-25	30
NV001	SP002	2023-11-25	30
NV001	SP003	2023-11-25	30
NV001	SP004	2023-11-25	30
NV001	SP005	2023-11-25	30
NV001	SP006	2023-11-25	30
NV001	SP007	2023-11-25	30
NV001	SP008	2023-11-25	30
NV001	SP009	2023-11-25	30

File jsp Quản lý Hóa Đơn Nhập Hàng

```
<%@ page language="java" contentType="text/html; charset=utf-8"
    pageEncoding="utf-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Lịch Sử Nhập Hàng</title>
<style>

table, h1{

    width: 80%;
    margin: 0 auto;
    border-collapse: collapse;
    margin-top: 10px;
}

th, td {
    border: 1px solid #ddd;
    padding: 8px;
    text-align: left;
}

th {
    background-color: #38A59F;
}
</style>
</head>
<body>
<div class="navbar">
<h1>Lịch Sử Nhập Hàng</h1>
<table id="myTable">
    <thead>
        <tr>
            <th>Mã Nhân Viên</th>
            <th>Mã Sản Phẩm</th>
            <th>Ngày Nhập Hàng</th>
            <th>Số Lượng</th>
        </tr>
    </thead>
    <tbody>
        <c:forEach var="lichSuNhapHang" items="${lichSuNhapHangList}">
            <tr>
                <td>${lichSuNhapHang.maNhanVien}</td>
                <td>${lichSuNhapHang.maSanPham}</td>
                <td>${lichSuNhapHang.ngayNhapHang}</td>
                <td>${lichSuNhapHang.soLuong}</td>
            </tr>
        </c:forEach>
    </tbody>
</table>
</body>
</html>
```

Controller Quản lý Hóa Đơn Nhập Hàng

```
@Controller
@RequestMapping(value = "/admin")
public class HomeAdminController extends BaseController {
    @Autowired
    NhapHangDao nhaphangdao;
    @Autowired
    DonHangDao donhangdao;
    @RequestMapping(value = "/lich-su-nhap-hang")
    public ModelAndView lsunhaphang() {
        _mvShare.addObject("lichSuNhapHangList", nhaphangdao.GetDataNhapHang());
        _mvShare.setViewName("admin/his_nhaphang");
        return _mvShare;
    }
}
```

2.2.6. Quản lý Hóa Đơn Thanh Toán

Quản lý có thể xem danh sách toàn bộ hóa đơn thanh toán để dễ dàng kiểm soát doanh thu của cửa hàng cũng như mặt hàng nào được mua nhiều nhất để có thể tìm ra những chiến lược phát triển sau này

Trang chủ nv002 Đăng xuất Liên hệ

HDL

Nhân ViênKhách hàngSản phẩmLoại sản phẩmLịch Sử Nhập hàngLịch Sử Thanh toán

Lịch Sử Thanh Toán

Tổng Cộng: 676700.00 đ

Mã Đơn Hàng	Ngày Đặt Hàng	Tổng Cộng	Nhân Viên Cash
1	2023-12-03	17500	NV001
10	2023-12-03	5800	NV001
11	2023-12-03	5800	NV001
12	2023-12-03	52500	NV001
13	2023-12-03	70100	NV001
14	2023-12-03	52500	NV001
15	2023-12-03	35000	NV001

File jsp Quản lý Hóa Đơn Thanh Toán

```
<%@ page language="java" contentType="text/html; charset=utf-8"
    pageEncoding="utf-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Lịch Sử Nhập Hàng</title>
<script>
    function calculateTotalRevenue() {
        var tongCong = 0;
        var table = document.getElementById("myTable");
        var rows = table.getElementsByTagName("tbody")[0].getElementsByTagName("tr");

        for (var i = 0; i < rows.length; i++) {
            var cell = rows[i].getElementsByTagName("td")[2];
            var value = parseFloat(cell.innerText || cell.textContent);

            if (!isNaN(value)) {
                tongCong += value;
            }
        }

        return tongCong;
    }

    function displayTotalRevenue() {
        var tongCong = calculateTotalRevenue();
        var totalRevenueElement = document.getElementById("totalRevenue");
        var totalAmountElement = document.getElementById("totalAmount");

        if (totalAmountElement) {
            totalAmountElement.innerHTML = tongCong.toFixed(2) + " đ";
        }

        if (totalRevenueElement) {
            totalRevenueElement.style.display = "block";
        }
    }

    window.onload = displayTotalRevenue;
</script>
</head>
<body>
    <div class="navbar">
        <h1>Lịch Sử Thanh Toán</h1>
        <!-- Đoạn mã hiển thị tổng cộng -->
        <h2 id="totalRevenue" style = "color:red">Tổng Cộng: <span id="totalAmount"></span></h2>
        <table id="myTable">
            <thead>
                <tr>
                    <th>Mã Đơn Hàng</th>
                    <th>Ngày Đặt Hàng</th>
                    <th>Tổng Cộng</th>
                    <th>Nhân Viên Cash</th>
                </tr>
            </thead>
            <tbody>
                <c:forEach var="lichSuNhapHang" items="${lichSuNhapHangList}">
                    <tr>
                        <td>${lichSuNhapHang.maDonHang}</td>
                        <td>${lichSuNhapHang.ngayDatHang}</td>
                        <td>${lichSuNhapHang.tongCong}</td>
                        <td>${lichSuNhapHang.maNhanVien}</td>
                    </tr>
                </c:forEach>
            </tbody>
        </table>
    </body>
</html>
```

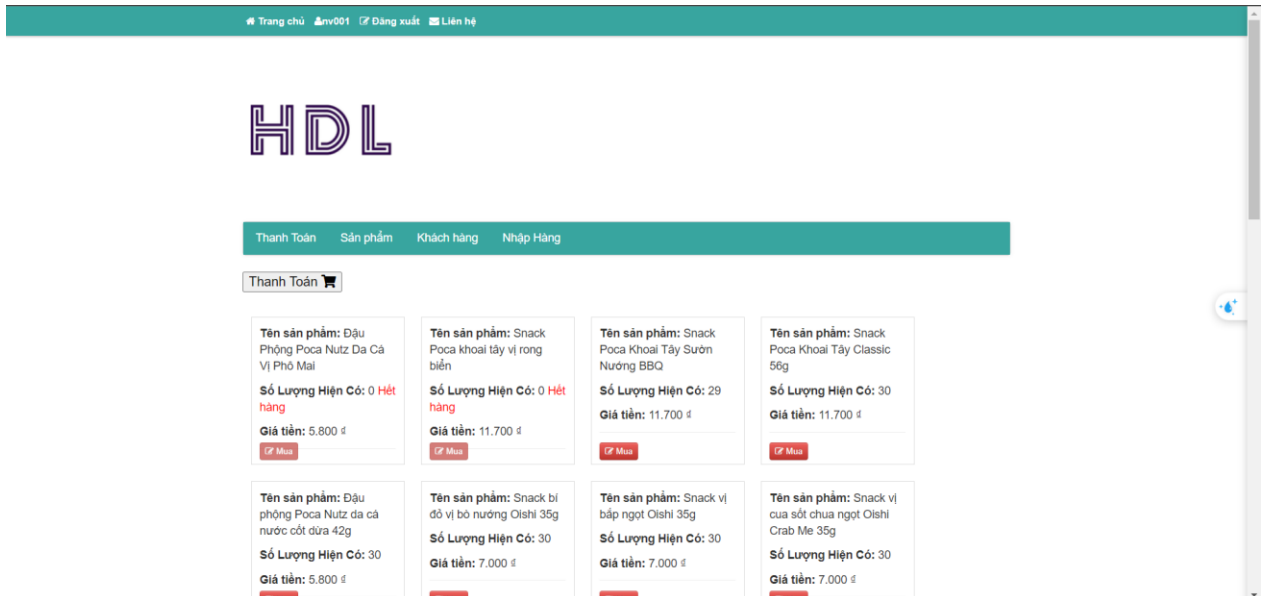
Controller Quản lý Hóa Đơn Thanh Toán

```
@Controller
@RequestMapping(value = "/admin")
public class HomeAdminController extends BaseController {
    @Autowired
    NhapHangDao nhaphangdao;
    @Autowired
    DonHangDao donhangdao;
    @RequestMapping(value = "/")
    public String homeAdmin() {
        return "admin/index";
    }
    @RequestMapping(value = "/lich-su-nhap-hang")
    public ModelAndView lsunhaphang() {
        _mvShare.addObject("lichSuNhapHangList",
            nhaphangdao.GetDataNhapHang());
        _mvShare.setViewName("admin/his_nhaphang");
        return _mvShare;
    }
    @RequestMapping(value = "/lich-su-thanh-toan")
    public ModelAndView lsuhoadon() {
        _mvShare.addObject("lichSuNhapHangList",
            donhangdao.GetDataDonHang());
        _mvShare.setViewName("admin/his_thanhtoan");
        return _mvShare;
    }
}
```

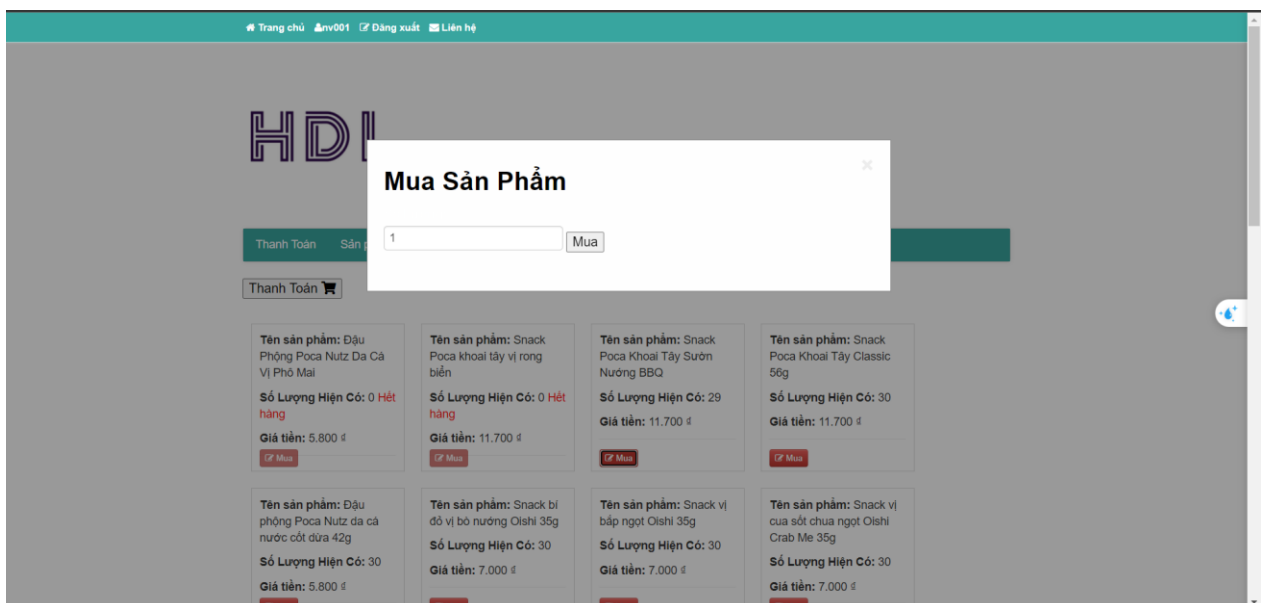
2.3. Các chức năng của Nhân Viên

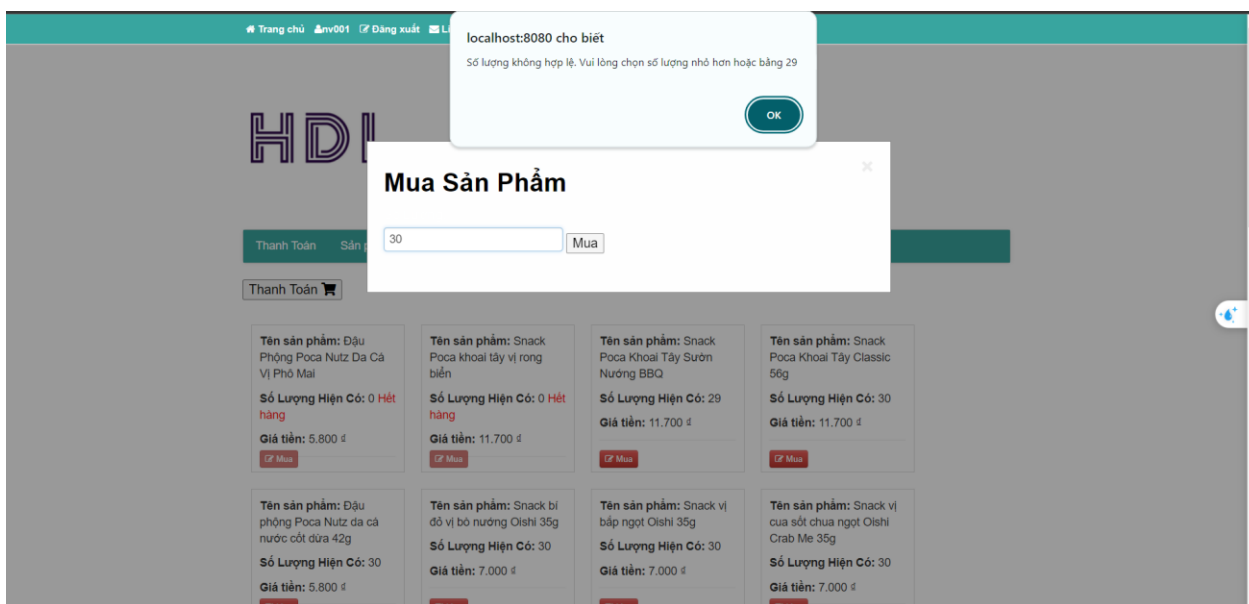
2.3.1. Quản lý Thanh Toán và xuất Hóa Đơn

Nhân Viên có thể xem toàn bộ danh sách sản phẩm bao gồm: Tên sản phẩm, số lượng hiện có và giá tiền. Mỗi khi số lượng hiện có là 0 nhân viên sẽ được thông báo hết hàng để có thể nhập hàng để đáp ứng nhu cầu của khách hàng

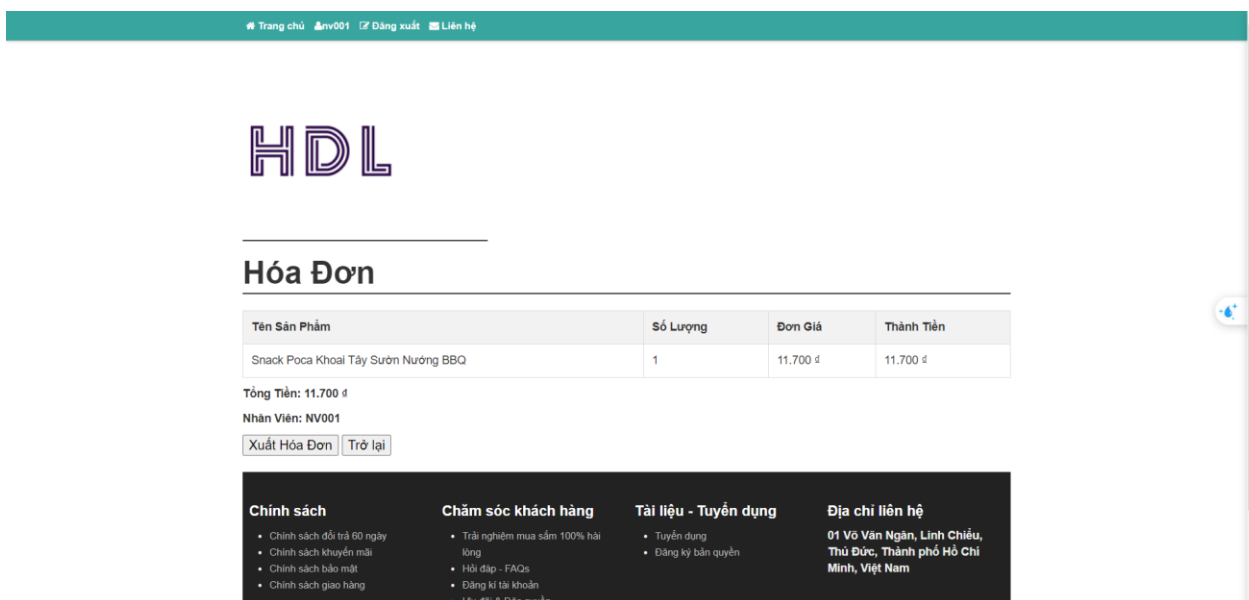


Khi thanh toán Nhân Viên ấn vào nút “Mua” ở sản phẩm tương ứng và nhập số lượng sản phẩm. Số lượng chỉ có thể nhập từ 1 và nếu số lượng mua nhiều hơn số lượng có sẵn của sản phẩm thì sẽ báo lỗi





Sau khi đủ số lượng sản phẩm thì ấn vào nút “Thanh Toán” để kiểm tra lại hóa đơn và tổng tiền



Nếu muốn thêm sản phẩm thì ấn nút “Trở lại” và thao tác như trên. Nếu sản phẩm đã đủ thì ấn nút “Xuất hóa đơn”

File jsp Quản lý Thanh Toán và xuất Hóa Đơn

```
<%@ page language="java" contentType="text/html; charset=utf-8"
    pageEncoding="utf-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>

<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<link rel="stylesheet"
    href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">
<title>Thanh Toán</title>
</head>
<body>
    <div class="navbar">
        <div class="navbar-inner">

            <form action="xuat-hoa-don" method="POST">
                <button type="submit">
                    Thanh Toán
                    <c:choose>
                        <c:when test="${soLuongDaMua != null}">
                            <span id="soLuongDaMua" style="color:red"> <i
class="fas fa-shopping-cart"></i>
                                ${soLuongDaMua}</span>
                        </c:when>
                        <c:otherwise>
                            <i class="fas fa-shopping-cart"></i>
                        </c:otherwise>
                    </c:choose>
                </button>
            </form>

            <!-- Danh sách sản phẩm -->
            <div class="product-list">
                <c:forEach var="item" items="${sanpham}">
                    <div class="product-item">
                        <button class="btn btn-mini btn-danger edit-cart" type="button"
                            onclick="openModal('${item.maSanPham}', '${item.tenSanPham}', '${item.soLuongHienCo}',
                                '${item.giaTien}')">
                            <span class="icon-edit"></span> Mua
                        <p>
                            <strong>Tên sản phẩm:</strong> ${item.tenSanPham}
                        </p>
                        <p>
                            <strong>Số Lượng Hiện Có:</strong> ${item.soLuongHienCo}
                            ${item.soLuongHienCo eq 0 ? '<span style="color:red;">Hết
hàng</span>' : ''}
                        </p>
                        <p>
                            <strong>Giá tiền:</strong>
                            <fmt:formatNumber type="number" groupingUsed="true"
                                value="${item.giaTien}" />
                        </p>
                        <hr class="soften" />
                    </div>
                </c:forEach>
            </div>

            <!-- Modal form mua sản phẩm -->
            <div id="myModal" class="modal">
                <div class="modal-content">
                    <span class="close" onclick="closeModal()">&times;</span>
                </div>
            </div>
        </div>
    </body>
</html>
```

```

<h2 >Mua Sản Phẩm</h2>
<form id="muaForm" action="thanh-toan" method="POST"
                                onsubmit="return
submitForm()" ">
    <input type="hidden" id="maSanPhamInput" name="maSanPhamInput">
    <input type="hidden" id="giaTienInput" name="giaTienInput">
    <input type="hidden" id="soLuongHienCo" name="soLuongHienCo"
                                value="0">
<label for="soLuongInput">Số Lượng:</label> <input
                                type="number"
id="soLuongInput" name="soLuongInput" min="1"
                                value="1"
onchange="updateSoLuong(this.value)">
    <button type="submit">Mua</button>
</form>
</div>
</div>
<script>
    function updateSoLuong(newValue) {
        var soLuongInput = document.getElementById('soLuongInput');
        soLuongInput.value = newValue;
        console.log("Số Lượng:", soLuongInput.value);
    }
    function openModal(maSanPham, tenSanPham, soLuong, giaTien) {
        if (soLuong === 0) {
            alert('Sản phẩm đã hết hàng.');
```

bằng '

```

            return;
        }

        var soLuongInput = document.getElementById('soLuongInput');
        var maSanPhamInput = document.getElementById('maSanPhamInput');
        var giaTienInput = document.getElementById('giaTienInput');
        var soLuongHienCoInput = document.getElementById('soLuongHienCo');

        soLuongInput.value = 1;
        updateSoLuong(soLuongInput.value);

        maSanPhamInput.value = maSanPham;
        giaTienInput.value = giaTien;
        soLuongHienCoInput.value = soLuong;

        console.log("Ma San Pham:", maSanPhamInput.value);
        console.log("Gia Tien:", giaTienInput.value);

        document.getElementById('myModal').style.display = 'block';
    }
    function closeModal() {
        document.getElementById('myModal').style.display = 'none';
    }
    function submitForm() {
        var soLuongInput = document.getElementById('soLuongInput');
        var soLuongHienCo = parseInt(document
            .getElementById('soLuongHienCo').value);

        if (parseInt(soLuongInput.value) > soLuongHienCo) {
            alert('Số lượng không hợp lệ. Vui lòng chọn số lượng nhỏ hơn hoặc
                                + soLuongHienCo);
            return false;
        } else {
            closeModal();
            return true;
        }
    }
</script>
</body>
</html>

```

Controller Quản lý Thanh Toán và xuất Hóa Đơn

```
@Controller
@RequestMapping(value = "/user")
public class HomeUserController {

    @Autowired
    SanPhamDao sanphamDao;

    @Autowired
    DonHangDao donhangDao;

    @Autowired
    ChiTietDonHangDao chitietdonhangDao;

    @Autowired
    KhachHangDao khachhangDao;

    @RequestMapping(value = "/")
    public String homeUser() {
        return "user/index";
    }

    @RequestMapping(value = "/thanh-toan", method = RequestMethod.GET)
    public String thanhToan() {
        return "user/thanhtoan";
    }

    @ModelAttribute("sanpham")
    public List<SanPhamDto> getAllSanPham() {
        return sanphamDao.getAll();
    }

    @ModelAttribute("customerList")
    public List<KhachHangDto> getAllKhachHang() {
        return khachhangDao.getAll();
    }

    @RequestMapping(value = "/thanh-toan", method = RequestMethod.POST)
    public String muaHang(HttpServletRequest request, @RequestParam("maSanPhamInput") String maSanPham,
        @RequestParam("soLuongInput") int soLuong, @RequestParam("giaTienInput") int giaTien,
        Model model) {
        ChiTietDonHangDto chitietdonhang = new ChiTietDonHangDto();
        chitietdonhang.setmaSanPham(maSanPham);
        chitietdonhang.setGiaTien(giaTien);
        chitietdonhang.setsoLuong(soLuong);
        chitietdonhangDao.insert(chitietdonhang);
        int soLuongDaMua = chitietdonhangDao.soLuongDaMua();
        model.addAttribute("soLuongDaMua", soLuongDaMua);
        return "user/thanhtoan";
    }

    @RequestMapping(value = "/xuat-hoa-don", method = RequestMethod.POST)
    public String hoaDon(Model model) {

        List<ChiTietDonHangDto> donHang = chitietdonhangDao.getAll();
        int soLuong = chitietdonhangDao.tongTien();
        model.addAttribute("chiTietDonHangList", donHang);
        model.addAttribute("tongTien", soLuong);
        return "user/hoadon";
    }

    @RequestMapping(value = "/xuat-hoa-don", method = RequestMethod.GET)
    public String xuấtHoaDon(Model model, HttpSession session, @RequestParam("tongTien") int tongTien,
        @RequestParam("tenNhanVien") String tenNhanVien) {
        DonHangDto donHang = new DonHangDto();
        donHang.settongCong(tongTien);
        donHang.setmaNhanVien(tenNhanVien);
        donhangDao.insert(donHang);
        List<ChiTietDonHangDto> donHang1 = chitietdonhangDao.getAll();
        for (ChiTietDonHangDto chiTietDonHang : donHang1) {
            int soLuongHienCo = chiTietDonHang.getsoLuong();
            String maSanPham = chiTietDonHang.getmaSanPham();
            SanPhamDto sanPhamDto = new SanPhamDto();
            sanPhamDto.setmaSanPham(maSanPham);
            sanPhamDto = sanphamDao.getById(sanPhamDto.getmaSanPham());
            sanPhamDto.setsoLuongHienCo(sanPhamDto.getsoLuongHienCo() - soLuongHienCo);
            sanphamDao.updateSoLuong(sanPhamDto);
        }
        chitietdonhangDao.deleteAll();
        return "redirect:thanh-toan";
    }
}
```

2.3.2. Quản lý Sản Phẩm

Nhân viên có thể xem danh sách các sản phẩm cũng như số lượng để có thể nhập hàng kịp thời phục vụ nhu cầu khách hàng

Trang chủ nv001 Đăng xuất Liên hệ

HDL

Thanh Toán Sản phẩm Khách hàng Nhập Hàng

Sản phẩm

Mã sản phẩm	Mã loại sản phẩm	Tên sản phẩm	Giá tiền	Số lượng hiện có
SP001	SNACKPOCA	Đậu Phộng Poca Nutz Da Cá Vịt Phở Mai	5800	0
SP002	SNACKPOCA	Snack Poca khoai tây vị rong biển	11700	0
SP003	SNACKPOCA	Snack Poca Khoai Tây Sườn Nướng BBQ	11700	29
SP004	SNACKPOCA	Snack Poca Khoai Tây Classic 56g	11700	30
SP005	SNACKPOCA	Đậu phộng Poca Nutz da cá nước cốt dừa 42g	5800	30
SP006	SNACKOISHI	Snack bí đỏ vị bò nướng Oishi 35g	7000	30
SP007	SNACKOISHI	Snack vị bắp ngọt Oishi 35g	7000	30
SP008	SNACKOISHI	Snack vị cua sốt chua ngọt Oishi Crab Me 35g	7000	30
SP009	SNACKOISHI	Snack vị cà chua Oishi Tomati 35g	7000	30
SP010	SNACKOISHI	Snack vị dưa chuột Oishi dưa chuột 35g	7000	30

File jsp Quản lý Sản phẩm

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@include file="/WEB-INF/views/layouts/user/taglib.jsp"%>
<head>
<title>SẢN PHẨM</title>
</head>
<body>
    <div class="row">
        <div class="navbar">
            <div class="well well-small">
                <h1>SẢN PHẨM</h1>
                <hr class="soften" />
                <br>

                <table class="table table-bordered table-condensed">
                    <thead>
                        <tr>
                            <th>Mã sản phẩm</th>
                            <th>Mã loại sản phẩm</th>
                            <th>Tên sản phẩm</th>
                            <th>Giá tiền</th>
                            <th>Số lượng hiện có</th>
                        </tr>
                    </thead>
                    <tbody>

                        <c:forEach var="item" items="${ sanpham }">
                            <tr>
                                <td>${ item.maSanPham }</td>
                                <td>${ item.maLoaiSanPham }</td>
                                <td>${ item.tenSanPham }</td>
                                <td>${ item.giaTien }</td>
                                <td>${ item.soLuongHienCo }</td>
                            </tr>
                        </c:forEach>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</body>
```

Controller Quản lý Sản phẩm

```
@Controller
public class SanPhamController extends BaseController {

    @Autowired
    SanPhamServiceImpl sanphamService = new SanPhamServiceImpl();

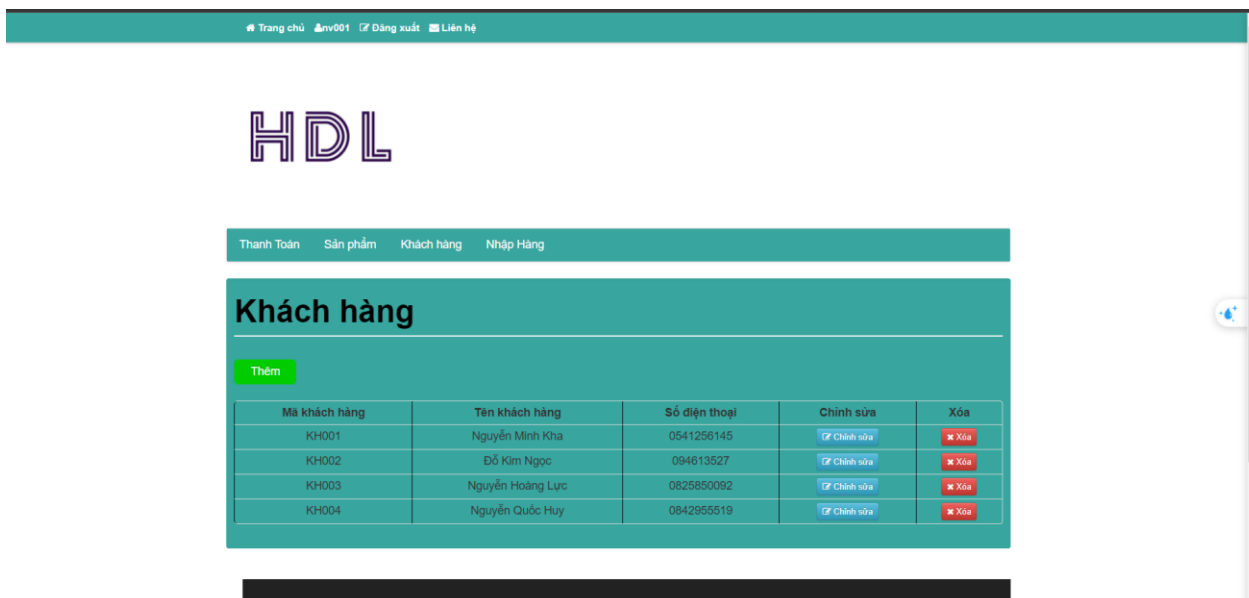
    @Autowired
    LoaiSanPhamServiceImpl loaisanphamService = new
    LoaiSanPhamServiceImpl();

    @Autowired
    SanPhamDao sanphamDao;

    @RequestMapping(value = "/user/xem-san-pham")
    public ModelAndView UserIndex() {
        _mvShare.addObject("sanpham", _homeService.GetDataProducts());
        _mvShare.setViewName("user/products/product");
        return _mvShare;
    }
}
```

2.3.3. Quản lý Danh Sách Khách Hàng

Nhân Viên có thể xem danh sách toàn bộ Khách Hàng cũng như việc thêm và chỉnh sửa cũng như xóa thông tin Khách Hàng



Mã khách hàng	Tên khách hàng	Số điện thoại	Chỉnh sửa	Xóa
KH001	Nguyễn Minh Kha	0541256145	Chỉnh sửa	Xóa
KH002	Đỗ Kim Ngọc	094613527	Chỉnh sửa	Xóa
KH003	Nguyễn Hoàng Lực	0825850092	Chỉnh sửa	Xóa
KH004	Nguyễn Quốc Huy	0842955519	Chỉnh sửa	Xóa

Nếu muốn thêm nhân viên mới thì ấn vào nút “Thêm” và điền các thông tin vào các trường và ấn nút “Lưu” nếu muốn thoát ấn nút “Quay lại”

The screenshot shows a web browser window with the HDL logo at the top. Below the logo is a teal-colored form titled "Thông tin khách hàng". The form contains three input fields: "Mã khách hàng:" with a placeholder "Mời nhập mã khách hàng", "Tên khách hàng:" with a placeholder "Mời nhập tên khách hàng", and "Số điện thoại:" with a placeholder "Mời nhập số điện thoại". At the bottom of the form are two buttons: "Lưu" (Save) and "Quay lại" (Back). The footer of the website is dark grey and contains four sections: "Chính sách" (Policy) with a link to "Chính sách đổi trả 60 ngày", "Chăm sóc khách hàng" (Customer Care) with a link to "Trải nghiệm mua sắm 100% hài lòng", "Tài liệu - Tuyển dụng" (Documents - Recruitment) with a link to "Tuyển dụng", and "Địa chỉ liên hệ" (Contact Address) with the address "01 Võ Văn Ngân, Linh Chiểu, Thủ Đức, Thành phố Hồ Chí Minh".

Nếu muốn chỉnh sửa thông tin Khách Hàng thì ấn vào nút “Chỉnh sửa” và điền các thông tin vào các trường và ấn nút “Cập nhật” nếu muốn thoát ấn nút “Quay lại”

Nếu muốn xóa sản phẩm ấn nút “Xóa”

File jsp Quản lý Danh sách Khách Hàng

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@include file="/WEB-INF/views/layouts/user/taglib.jsp"%>
<head>
<title>Khách hàng</title>
</head>
<body>
    <div class="row">
        <div class="navbar">

            <div class="well well-small">
                <h1>Khách hàng</h1>
                <hr class="soften" />
                <br> <a href="<c:url value="/them-khach-hang"/>"
                    class="addBtn">Thêm</a> <br>
                <br>

                <table class="table table-bordered table-condensed">
                    <thead>
                        <tr>
                            <th>Mã khách hàng</th>
                            <th>Tên khách hàng</th>
                            <th>Số điện thoại</th>
                            <th>Chỉnh sửa</th>
                            <th>Xóa</th>
                        </tr>
                    </thead>
                    <tbody>

                        <c:forEach var="item" items="${ khachhang }">
                            <tr>
                                <td>${ item.maKhachHang }</td>
                                <td>${ item.tenKhachHang }</td>
                                <td>${ item.sdt }</td>
                                <td><a
                                    href="<c:url
value='/editCustomer/${item.maKhachHang}' />"
                                    class="btn btn-mini btn-info"
type="button"> <span
                                    class="icon-
edit"></span> Chỉnh sửa
                                </a></td>
                                <td><a
                                    href="<c:url
value='/deleteCustomer/${item.maKhachHang}' />"
                                    class="btn btn-mini btn-
danger" type="button"> <span
                                    class="icon-
remove"></span> Xóa
                                </a></td>
                            </tr>
                        </c:forEach>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</body>
```

Controller Quản lý Danh sách Khách Hàng

```
@Controller
public class KhachHangController extends BaseController{
    @Autowired
    KhachHangServiceImpl khachhangService = new KhachHangServiceImpl();

    @Autowired
    KhachHangDao khachhangDao;

    @RequestMapping(value = "/user/khach-hang")
    public ModelAndView UserIndex() {
        _mvShare.addObject("khachhang", _homeService.GetDataCustomers());
        _mvShare.setViewName("user/customers/customer");
        return _mvShare;
    }

    @RequestMapping(value="them-khach-hang",method= RequestMethod.GET)
    public ModelAndView AddCreateCustomer() {
        _mvShare.setViewName("user/customers/addcustomer");
        _mvShare.addObject("khachhang", new KhachHang());
        return _mvShare;
    }

    @RequestMapping(value="them-khach-hang",method= RequestMethod.POST)
    @ModelAttribute("khachhang") KhachHang khachhang) {
        HttpSession session = request.getSession();
        int count = khachhangService.AddCustomer(khachhang);
        if(count > 0) {
            session.setAttribute("status", "Thêm thành công!");
        }
        else {
            session.setAttribute("status", "Thêm thất bại!");
        }
        return "redirect:" +request.getHeader("Referer");
    }

    @RequestMapping(value = "/editCustomer/{maKhachHang}", method =
    RequestMethod.GET)
    public String editCustomer(@PathVariable("maKhachHang") String
    maKhachHang, Model model) {
        KhachHangDto customer =
        khachhangService.GetCustomerById(maKhachHang);
        model.addAttribute("khachHang", customer);
        return "user/customers/editcustomer";
    }

    @RequestMapping(value= "/update-customer", method = RequestMethod.POST)
    public String update(@RequestParam("maKhachHang") String maKhachHang,
        @RequestParam("tenKhachHang") String tenKhachHang,
        @RequestParam("sdt") String sdt, KhachHangDto kh) {
        KhachHangDto khachhangDto = khachhangDao.getById(maKhachHang);
        if (khachhangDto == null) {
            return "redirect:/error-page";
        }
        khachhangDto.settenKhachHang(tenKhachHang);
        khachhangDto.setsdt(sdt);
        khachhangDao.update(khachhangDto);
        return "redirect:/user/khach-hang";
    }
}
```

2.3.4. Quản lý Nhập Hàng

Nhân Viên có nhập thêm số lượng của sản phẩm khi hết hàng. Khi nhập hàng thành công sẽ tự động cập nhật lại số lượng hiện có của sản phẩm đó. Đầu tiên chọn tên sản phẩm tương ứng sau đó nhập số lượng muốn nhập hàng. Nếu muốn nhập hàng thì ấn “Nhập hàng” ngược lại thì ấn “Quay lại”

The screenshot displays the HDL website interface. At the top, a teal navigation bar contains links for 'Trang chủ', 'nv001', 'Đăng xuất', and 'Liên hệ'. Below this, the 'HDL' logo is prominently displayed. The main content area features a teal box titled 'Nhập Hàng'. Inside this box, there is a 'Chọn Sản Phẩm:' dropdown menu currently showing 'Đậu Phộng Poca Nutz Đa Cà'. Below the dropdown is a text input field for 'Số Lượng:'. At the bottom of the teal box are two orange buttons: 'Nhập hàng' and 'Quay lại'. The footer is a dark grey bar divided into four sections: 'Chính sách' (with links to return policy, promotion policy, and warranty policy), 'Chăm sóc khách hàng' (with links to return policy, FAQs, and account management), 'Tài liệu - Tuyển dụng' (with links to recruitment and job application), and 'Địa chỉ liên hệ' (providing the address: 01 Võ Văn Ngân, Linh Chiểu, Thủ Đức, Thành phố Hồ Chí Minh, Việt Nam).

File jsp Quản lý Nhập Hàng

```
<%@ page language="java" contentType="text/html; charset=utf-8"
    pageEncoding="utf-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt"%>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Nhập Hàng</title>
</head>
<body>
    <div class="row">
        <div id="sidebar"></div>
        <div class="span12">
            <div class="span2">&nbsp;</div>
            <div class="span6">
                <div class="well">
                    <h1>Nhập Hàng</h1>
                    <br> <br>
                    <form
                        action="{pageContext.request.contextPath}/xuly-nhap-hang"
                        method="POST">
                        <input type="hidden" name="maNhanVien"
                            value="{maNhanVien}">
                        <div class="form-row">
                            <label for="sanPham">Chọn Sản
                                Phẩm:</label>
                                <div>
                                    <select id="sanPham"
                                        name="sanPham">
                                        <c:forEach
                                            var="sanPhamItem" items="{danhSachSanPham}">
                                            <option
                                                value="{sanPhamItem.maSanPham}">{sanPhamItem.tenSanPham}</option>
                                        </c:forEach>
                                    </select>
                                </div>
                                <div class="form-row">
                                    <label for="soLuong">Số Lượng:</label>
                                    <div>
                                        <input type="number" id="soLuong"
                                            name="soLuong" min="1"
                                            required>
                                    </div>
                                </div>
                                <br>
                                <button type="submit" class="defaultBtn">Nhập
                                    hàng</button>
                                <a href="{c:url value='/user/'}"
                                    class="defaultBtn">Quay
                                    lại</a>
                                </form>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</body>
</html>
```

Controller Quản lý Nhập hàng

```
@Controller
public classNhapHangController extends BaseController{

    @Autowired
    private SanPhamServiceImpl sanPhamService;
    @Autowired
    SanPhamDao sanphamDao;

    @Autowired
    NhapHangDao nhaphangDao;

    @GetMapping("/user/nhap-hang")
    public ModelAndView hienThiTrangNhapHang(Model model, HttpSession
session) {
        // Lấy mã nhân viên từ session
        _mvShare.addObject("danhSachSanPham", sanPhamService.GetAll());
        _mvShare.setViewName("user/nhaphang");
        return _mvShare;
    }

    @RequestMapping(value= "/xuly-nhap-hang", method = RequestMethod.POST)
    public String xuLyNhapHang(@RequestParam("maNhanVien") String maNhanVien,
                                @RequestParam("sanPham") String
maSanPham,
                                @RequestParam("soLuong") int soLuong) {

        SanPhamDto sanPhamDto = new SanPhamDto();
        sanPhamDto.setmaSanPham(maSanPham);
        sanPhamDto = sanphamDao.getById(sanPhamDto.getmaSanPham());
        sanPhamDto.setsoLuongHienCo(sanPhamDto.getsoLuongHienCo() + soLuong);

        sanphamDao.updateSoLuong(sanPhamDto);

        NhapHang nhaphang = new NhapHang();
        nhaphang.setmaNhanVien(maNhanVien);
        nhaphang.setmaSanPham(maSanPham);
        nhaphang.setsoLuong(soLuong);

        nhaphangDao.themNhapHang(nhaphang);

        return "redirect:user/xem-san-pham";
    }
}
```

CHƯƠNG 3: KẾT LUẬN

1. Đánh giá của sinh viên

- Đạt hầu hết các yêu cầu đề ra của một website quản lý cơ bản
- Phân quyền rõ ràng giữa Staff và Admin
- Giao diện thân thiện dễ thao tác cho người dùng
- Code trình bày dễ hiểu
- Phân lớp rõ ràng giúp bảo mật dữ liệu của người dùng

2. Ưu điểm và nhược điểm

- Ưu điểm
 - + Giao diện thân thiện người dùng
 - + Phân quyền rõ ràng
 - + Clean code giúp người dùng có thể tái tạo code
- Nhược điểm
 - + Chưa cho phép người dùng chèn ảnh vào database
 - + Chưa thực hiện được thao tác quản lý chi tiết doanh thu, hóa đơn
 - + Chưa có tính năng tìm kiếm

3. Hướng phát triển

- Xây dựng tính năng tìm kiếm sản phẩm và hóa đơn
- Cho phép người dùng chèn hình ảnh vào sản phẩm
- Mã hóa mật khẩu trong Database

TÀI LIỆU THAM KHẢO

1. . <https://cuongquach.com/tai-lieu-java-spring-mvc-nhat-nghe-pdf.html>
2. <https://thuvienso.hcmute.edu.vn/tailieuvn/doc/pro-spring-mvc-with-web-flow-1367788.html>