# MỤC LỤC

[MỤC LỤC 1](#_Toc8165162)

[CHƯƠNG 1 : GIỚI THIỆU 3](#_Toc8165163)

[1.1 Tổng quan 3](#_Toc8165164)

[1.2 Mục tiêu đề tài 3](#_Toc8165165)

[1.3 Phạm vi đề tài 3](#_Toc8165166)

[1.4 Mô tả yêu cầu chức năng 3](#_Toc8165167)

[CHƯƠNG 2 : CƠ SỞ LÝ THUYẾT 4](#_Toc8165168)

[2.1 Java 4](#_Toc8165169)

[2.2 Tomcat Server 5](#_Toc8165170)

[2.3 Apache Maven 5](#_Toc8165171)

[CHƯƠNG 3 : PHÂN TÍCH 7](#_Toc8165172)

[3.1 Mô hình Use case tổng quát 7](#_Toc8165173)

[3.2 Đặc tả Use case 8](#_Toc8165174)

[3.3 Activity Diagram 12](#_Toc8165175)

[3.4 Mô hình CSDL 15](#_Toc8165176)

[3.5 Mô hình lớp Entity 16](#_Toc8165177)

[CHƯƠNG 4 : THIẾT KẾ VÀ HIỆN THỰC 17](#_Toc8165178)

[4.1 Giao diện 17](#_Toc8165179)

[4.2 Một vài hình ảnh giao diện 17](#_Toc8165180)

[CHƯƠNG 5 : KẾT LUẬN 18](#_Toc8165181)

[5.1 Kết quả đạt được 18](#_Toc8165182)

[5.2 Hạn chế của đồ án 18](#_Toc8165183)

[5.3 Hướng phát triển 18](#_Toc8165184)

# : GIỚI THIỆU

## Tổng quan

* Đề tài xây dựng hệ thống website bán máy in, máy scan được thiết kế với mục đích thay thế hệ thống quản lí thủ công đã lỗi thời của các cửa hàng. Cụ thể thì khi đưa vào làm việc, hệ thống website sẽ tổng hợp các nghiệp vụ quản lí sản phẩm, danh mục,... Giúp cho công tác quản lí được tập trung, giảm thiểu nhân lực vận hành và đạt kết quả cao.

## Mục tiêu đề tài

* Thực hiện được việc quản lí sản phẩm, tài khoản.
* Sản phẩm được lưu trữ với thông tin rõ ràng.
* Nhận biết được thông tin sản phẩm nhanh chóng, đầy đủ.
* Giao diện đơn giản, dễ nhìn, dễ thao tác.
* Người sử dụng là Admin, khách hàng, nhân viên.
* Đảm bảo việc bảo mật thông tin khách hàng.

## Phạm vi đề tài

* Xuất phát từ tình hình thực tiễn, với phạm vi hướng tới là các cửa hàng bán máy in, máy scan có quy mô vừa và nhỏ ,trang web của chúng tôi sẽ hỗ trợ tốt cho Admin trong công tác quản lý hoạt động của cửa hàng.

## Mô tả yêu cầu chức năng

* Khách hàng:
  + Xem thông tin sản phẩm.
  + Thêm sản phẩm vào giỏ hàng.
  + Quản lí, thao tác với giỏ hàng.
  + Đăng kí tài khoản.
  + Đăng nhập.
* Admin:
  + Quản lí sản phẩm.
  + Quản lí tài khoản.

# : CƠ SỞ LÝ THUYẾT

## Java

* Java là một ngôn ngữ lập trình và nền tảng tính toán được phân phối lần đầu tiên bởi Sun Microsystems vào năm 1995. Rất nhiều ứng dụng, trang web đều được viết bằng Java. Java nhanh, bảo mật và đáng tin cậy.
* Java là ngôn ngữ lập trình hướng đối tượng (OOP).
* Về tốc độ:
  + Trước đây, Java chạy chậm hơn những ngôn ngữ dịch thẳng ra mã máy như C và C++, nhưng sau này nhờ công nghệ “biên dịch tại chỗ” – Just in time compilation, khoảng cách này đã được thu hẹp, và trong một số trường hợp đặc biệt Java có thể chạy nhanh hơn.
  + Java chạy nhanh hơn ngôn ngữ thông dịch như Python, Perl, PHP gấp nhiều lần.
* Về quản lí bộ nhớ:
  + Trong Java, hiện tượng rò rỉ bộ nhớ hầu như không xảy ra do bộ nhớ được quản lý bởi Java Virtual Machine (JVM) bằng cách tự động “dọn dẹp rác”. Người lập trình không phải quan tâm đến việc cấp phát và xóa bộ nhớ như C, C++. Tuy nhiên khi sử dụng những tài nguyên mạng, file IO, database (nằm ngoài kiểm soát của JVM) mà người lập trình không đóng các kết nối thì rò rỉ dữ liệu vẫn có thể xảy ra.
* Về cú pháp:
  + Cú pháp Java được vay mượn nhiều từ C & C++ nhưng có cú pháp hướng đối tượng đơn giản hơn và ít tính năng xử lý cấp thấp hơn. Do đó việc viết một chương trình bằng Java dễ hơn, đơn giản hơn, đỡ tốn công sửa lỗi hơn.
* Các đặc trưng của Java:
  + Hướng đối tượng.
  + Đơn giản.
  + Độc lập phần cứng và hệ điều hành.
  + Mạnh mẽ.
  + Bảo mật.
  + Phân tán.
  + Đa luồng.

## Tomcat Server

* **Tomcat**là một ứng dụng chủ (Application Server), là một phần mềm mã nguồn mở được cung cấp bởi **Apache. Tomcat**nổi tiếng vì gọn nhẹ nó thường được sử dụng trong quá trình phát triển một ứng Web trên nền tảng **Java Servlet**.
* Bạn cũng có thể sử dụng **Tomcat**để triển khai các ứng dụng **Java Web**trên thực tế và hoàn toàn có thể tin tưởng vào sự ổn định của nó.
* Tomcat không nên được hiểu nhầm với các máy chủ HTTP Apache - cái mà dùng để thực thi các câu lệnh viết bằng ngôn ngữ C trên máy chủ HTTP; có 2 máy chủ web được kết nối với nhau. Apache Tomcat cung cấp các công cụ cho việc cấu hình và quản lý, nhưng cũng có thể được cấu hình bởi việc soạn thảo các file cấu hình viết bằng XML.
* Tomcat phiên bản 4.x xuất hiện bao gồm:
  + Catalina: Catalina chính là bộ servlet container của Tomcat. Catalina thực hiện các chi tiết kỹ thuật của Sun Microsystems đối với servlet và các trang JavaServer (JSP). Người đã xây dựng lên Catalina là Craig McClanahan.
  + Coyote: Coyote là bộ phận kết nối HTTP của Tomcat, có cung cấp giao thức HTTP 1.1 cho các máy chủ web hoặc các ứng dụng khác. Coyote nghe ngóng các kết nối đến nó trên cổng TCP được định sẵn trên máy chủ và sau đó trả lời các yêu cầu đến Tomcat để thực thi các yêu cầu và gửi lại trả lời cho máy trạm đã yêu cầu.
  + Jasper: Jasper là công cụ JSP của Tomcat. Tomcat 5.x sử dụng Jasper 2, chính là một thực thi các trang JSP được chỉ rõ của Sun Microsystems. Jasper phân tích các file JSP để biên dịch chúng trong code Java như là các servlets (có thể được điều khiển bởi Catalina). Tại thời điểm thực thi, Jasper có khả năng tự động dò ra các file JSP và biên dịch chúng.

## Apache Maven

* Maven là công cụ quản lý và thiết lập tự động 1 dự án phần mềm. Chủ yếu dùng cho các lập trình viên java, nhưng nó cũng có thể được dùng để xây dựng và quản lý các dự án dùng C#, Ruby, Scala hay ngôn ngữ khác. Maven phục vụ mục đích tương tự như Apache Ant, nhưng nó dựa trên khái niệm khác và cách hoạt động khác. Maven được quản lý bởi Apache Software Foundation, nó là một phần của dự án Jakarta Project.
* Maven dùng khái niệm Project Object Model (POM) để mô tả việc build project, các thành phần phụ thuộc và các module. Nó định nghĩa trước các target cho việc khai báo task, trình biên dịch, đóng gói và thứ tự hoạt động để mọi việc diễn ra tốt nhất.
* Maven tải các thư viện, plug-in từ 1 hoặc nhiều repositories. Maven cung cấp việc hỗ trợ tải file từ repository trung tâm của Maven 2 và các repositories khác. Cũng có thể upload các sản phẩm lên repository sau khi đã build xong.
* Maven được xây dựng dựa trên kiến trúc plugin, cho phép sử dụng bất kỳ ứng dụng điều khiển nào thông qua đầu vào tiêu chuẩn. Về mặt lý thuyết, điều này cho phép bất kỳ ai cũng có thể viết các plugin để giao tiếp với các build tool (trình biên dịch, unit test tool, etc) cho bất kỳ ngôn ngữ nào khác. Nhưng trong thực tế, hỗ trợ và sử dụng cho các ngôn ngữ khác đã bị thu hẹp. Hiện tại thì plugin cho .Net đã có và vẫn được duy trì, và một plugin cho [C](https://vi.wikipedia.org/wiki/C_(programming_language))/[C++](https://vi.wikipedia.org/wiki/C%2B%2B) cũng được duy trì cho Maven.

# : PHÂN TÍCH

## Mô hình Use case tổng quát

## Đặc tả Use case

|  |  |
| --- | --- |
| **Use case:** UC001-Đăng nhập | |
| Mục đích: | Đăng nhập thành công vào trang web |
| Mô tả: | Mô tả các bước đăng nhập của các tác nhân vào phần mềm |
| Tác nhân: | Admin, khách hàng |
| Điều kiện trước: | Tác nhân đã đăng kí tài khoản trước đó |
| Điều kiện sau: | Đăng nhập thành công vào trang web |
| Luồng sự kiện chính (Basic flows) | 1. Sau khi truy cập trang web, tác nhân có thể đăng nhập vào tài khoản bằng cách nhấn vào nút “Đăng nhập” 2. Xuất hiện form nhập tài khoản và mật khẩu 3. Tác nhân nhập vào tài khoản và mật khẩu, nhấn “Đăng nhập” 4. Hệ thống sẽ kiểm tra dữ liệu nhập 5. Đăng nhập thành công |
| Luồng sự kiện phụ | 5.1 Không thành công, quay lại bước 3 |
| Luồng sự kiện ngoại lệ (Exception Flows): |  |

|  |  |
| --- | --- |
| **Use case:** UC002-Đăng kí | |
| Mục đích: | Đăng kí tài khoản để thực hiện thanh toán việc mua hàng |
| Mô tả: | Mô tả các bước đăng kí tài khoản của khách hàng |
| Tác nhân: | Khách hàng |
| Điều kiện trước: | Khách hàng truy cập thành công vào trang web |
| Điều kiện sau: | Đăng kí thành công tài khoản |
| Luồng sự kiện chính (Basic flows) | 1. Sau khi truy cập trang web, khách hàng nhấn vào nút “Đăng kí” 2. Xuất hiện form đăng kí tài khoản 3. Khách hàng nhập vào tài khoản, mật khẩu và các thông tin cá nhân cần thiết, nhấn nút “Đăng kí” 4. Hệ thống sẽ kiểm tra dữ liệu nhập 5. Đăng kí thành công tài khoản |
| Luồng sự kiện phụ | 5.1 Không thành công, quay lại bước 3 |
| Luồng sự kiện ngoại lệ (Exception Flows): |  |

|  |  |
| --- | --- |
| **Use case:** UC003-Thêm sản phẩm vào giỏ hàng, tăng giảm số lượng sản phẩm | |
| Mục đích: | Chọn những sản phẩm mà khách hàng muốn mua |
| Mô tả: | Mô tả các bước thêm sản phẩm vào giỏ hàng của khách hàng |
| Tác nhân: | Khách hàng |
| Điều kiện trước: | Khách hàng truy cập thành công vào trang web |
| Điều kiện sau: | Thêm thành công sản phẩm vào giỏ hàng |
| Luồng sự kiện chính (Basic flows) | 1. Khách hàng nhấn vào sản phẩm mà mình muốn mua 2. Xuất hiện chi tiết thông tin và giá của sản phẩm đó 3. Khách hàng nhấn “Thêm vào giỏ hàng” nếu muốn mua 4. Nếu sản phẩm chưa có trong giỏ hàng, nó sẽ xuất hiện với số lượng là 1 5. Khách hàng có thể tăng hoặc giảm số lượng theo ý muốn của mình |
| Luồng sự kiện phụ |  |
| Luồng sự kiện ngoại lệ (Exception Flows): |  |

|  |  |
| --- | --- |
| **Use case:** UC004-Xem danh sách sản phẩm | |
| Mục đích: | Khách hàng xem các sản phẩm hiện có tại cửa hàng |
| Mô tả: | Mô tả các bước xem các sản phẩm của khách hàng |
| Tác nhân: | Khách hàng |
| Điều kiện trước: | Khách hàng truy cập thành công vào trang web |
| Điều kiện sau: | Xem được các sản phẩm có tại cửa hàng |
| Luồng sự kiện chính (Basic flows) | 1. Khách hàng có thể chọn danh mục sản phẩm để phân loại ra theo nhu cầu của mình, tìm kiếm dễ dàng hơn (máy in, máy scan) 2. Hệ thống sẽ xuất ra các sản phẩm thuộc danh mục mà khách hàng đã chọn |
| Luồng sự kiện phụ |  |
| Luồng sự kiện ngoại lệ (Exception Flows): |  |

|  |  |
| --- | --- |
| **Use case:** UC005-Thanh toán | |
| Mục đích: | Khách hàng thanh toán sau khi mua hàng xong |
| Mô tả: | Mô tả các bước thanh toán của khách hàng |
| Tác nhân: | Khách hàng |
| Điều kiện trước: | Khách hàng đã đăng nhập tài khoản và giỏ hàng có sản phẩm |
| Điều kiện sau: | Thanh toán thành công |
| Luồng sự kiện chính (Basic flows) | 1. Sau khi lựa chọn các sản phẩm theo nhu cầu và cho vào giỏ hàng, khách hàng nhấn nút “Thanh toán” để tiến hành thanh toán tiền 2. Thanh toán thành công |
| Luồng sự kiện phụ | 2.1 Nếu chưa đăng nhập sẽ chuyển sang trang đăng nhập |
| Luồng sự kiện ngoại lệ (Exception Flows): |  |

|  |  |
| --- | --- |
| **Use case:** UC006-Thêm sản phẩm | |
| Mục đích: | Thêm sản phẩm mới vào trang web |
| Mô tả: | Mô tả các bước thêm sản phẩm mới của Admin |
| Tác nhân: | Admin |
| Điều kiện trước: | Admin đã đăng nhập vào trang web với tài khoản admin |
| Điều kiện sau: | Thêm thành công sản phẩm mới |
| Luồng sự kiện chính (Basic flows) | 1. Sau khi đăng nhập với tài khoản admin, trang quản lí sản phẩm sẽ xuất hiện 2. Admin nhập các thông tin về sản phẩm mới vào form bên dưới, nhấn “Lưu” 3. Hệ thống kiểm tra dữ liệu nhập 4. Thêm thành công sản phẩm mới |
| Luồng sự kiện phụ | 4.1 Không thành công, quay lại bước 2 |
| Luồng sự kiện ngoại lệ (Exception Flows): |  |

|  |  |
| --- | --- |
| **Use case:** UC007-Sửa sản phẩm | |
| Mục đích: | Sửa thông tin sản phẩm tại cửa hàng |
| Mô tả: | Mô tả các bước sửa sản phẩmcủa Admin |
| Tác nhân: | Admin |
| Điều kiện trước: | Admin đã đăng nhập vào trang web với tài khoản admin |
| Điều kiện sau: | Sửa thành công sản phẩm |
| Luồng sự kiện chính (Basic flows) | 1. Sau khi đăng nhập với tài khoản admin, trang quản lí sản phẩm sẽ xuất hiện 2. Admin nhấn nút “Sửa” tại sản phẩm muốn sửa 3. Trang thông tin của sản phẩm sẽ xuất hiện 4. Admin chỉnh sửa lại các thông tin cần thiết, nhấn “Lưu” 5. Hệ thống kiểm tra dữ liệu nhập 6. Sửa thành công thông tin sản phẩm đó |
| Luồng sự kiện phụ | 6.1 Không thành công, quay lại bước 4 |
| Luồng sự kiện ngoại lệ (Exception Flows): |  |

|  |  |
| --- | --- |
| **Use case:** UC008-Xóa sản phẩm | |
| Mục đích: | Xóa sản phẩm tại cửa hàng |
| Mô tả: | Mô tả các bước xóa sản phẩm của Admin |
| Tác nhân: | Admin |
| Điều kiện trước: | Admin đã đăng nhập vào trang web với tài khoản admin |
| Điều kiện sau: | Xóa thành công sản phẩm |
| Luồng sự kiện chính (Basic flows) | 1. Sau khi đăng nhập với tài khoản admin, trang quản lí sản phẩm sẽ xuất hiện 2. Admin nhấn nút “Xóa” tại sản phẩm muốn xóa 3. Hiển thị nhắc nhở có chắc chắn xóa hay không 4. Admin nhấn nút “Xóa” 5. Xóa thành công sản phẩm đó |
| Luồng sự kiện phụ |  |
| Luồng sự kiện ngoại lệ (Exception Flows): |  |

## Activity Diagram

* Activity Đăng nhập
* Activity Đăng kí
* Activity Quản lí giỏ hàng (thêm, tăng giảm số lượng)
* Activity Xem danh sách sản phẩm
* Activity Thanh toán
* Activity Thêm sản phẩm
* Activity Sửa sản phẩm
* Activity Xóa sản phẩm

## Mô hình CSDL

## Mô hình lớp Entity

# : THIẾT KẾ VÀ HIỆN THỰC

## Giao diện

* Danh sách các giao diện:
  + Trang chủ.
  + Trang đăng nhập.
  + Trang đăng kí.
  + Trang thông tin chi tiết sản phẩm.
  + Trang giỏ hàng.
  + Trang quản lí của Admin.

## Một vài hình ảnh giao diện

# : KẾT LUẬN

## Kết quả đạt được

* Xem danh sách sản phẩm.
* Thực hiện giỏ hàng Session.
* Đăng kí tài khoản.
* Thêm sản phẩm vào CSDL.

## Hạn chế của đồ án

* Lưu thông tin đơn hàng xuống CSDL.
* Xác nhận đơn hàng của Admin.
* Tìm sản phẩm tương đối

## Hướng phát triển

* Hoàn thiện phần giỏ hàng.
* Hoàn thiện các chức năng của Admin.
* Và quan trọng nhất là xác nhận đơn hàng.