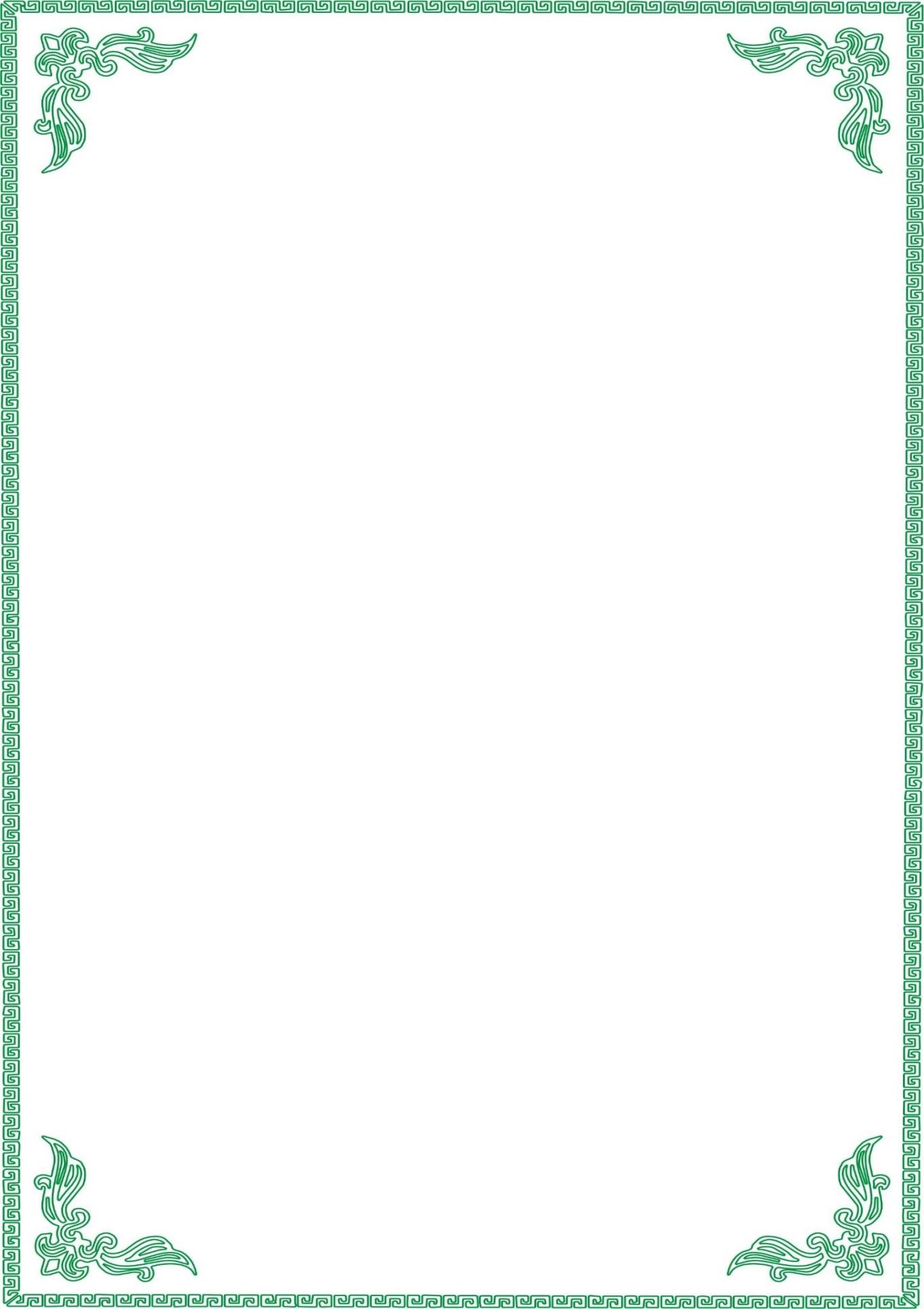
1



**KHOA CÔNG NGHỆ THÔNG TIN BỘ MÔN LẬP TRÌNH WEB**

ĐỒ ÁN CUỐI KỲ

**\*\*\***

WEBSITE BÁN ĐỒ CHƠI

**GVHD: Trương Thị Khánh Dịp NHÓM: 10**

|  |  |  |
| --- | --- | --- |
| **STT** | **Họ Và Tên** | **Mã Số Sinh Viên** |
| **1** | **Nguyễn Thị Thùy Linh** | **21133051** |
| **2** | **Trần Phan Quốc** | **21133108** |
| **3** | **Trần Thị Ngọc** **Trang** | **21133109** |

**LỜI CẢM ƠN**

Bài báo cáo này là sản phẩm của một quá trình học tập và làm việc nhóm. Để có thể hoàn thành bài báo cáo, chúng em đã nhận được rất nhiều sự hỗ trợ từ cô và các bạn. Do đó nhóm chúng em xin được phép gửi lời cảm ơn chân thành và sâu sắc nhất đến:

1. *Cô Trương Thị Khánh Dịp – giảng viên bộ môn Lập trình Web, khoa công nghệ thông tin đại học Sư Phạm Kỹ Thuật thành phố Hồ Chí Minh. Nhóm xin cảm ơn sự quan tâm và giúp đỡ tận tình của cô trong suốt quá trình giảng dạy. Cảm ơn cô đã luôn giải đáp những thắc mắc cũng như đưa ra những nhận xét, góp ý giúp nhóm thực hiện cải thiện chất lượng công việc của nhóm.*
2. *Thư viện trường đại học Sư Phạm Kỹ Thuật thành phố Hồ Chí Minh, nơi cung cấp môi trường học tập, nghiên cứu và làm việc nhóm để chúng em có thể hoàn thành tốt báo cáo của nhóm mình.*
3. *Các bạn học cùng lớp đã có những nhận xét, đóng góp về mặt kiến thức lẫn tinh thần cho nhóm.*

Đề tài và bài báo cáo được chúng em thực hiện trong khoảng thời gian ngắn, với những hạn chế khác về mặt kiến thức, kỹ thuật và kinh nghiệm trong việc thực hiện báo cáo. Do đó, trong quá trình làm đề tài có những thiếu sót là điều không thể tránh khỏi. Chúng em rất mong nhận được những ý kiến đóng góp quý báu của cô để kiến thức của chúng em được hoàn thiện hơn và làm tốt hơn nữa trong những lần sau. Chúng em xin chân thành cảm ơn!

*Trân trọng,*

**Nhóm 10**

**NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN**

**Họ và tên sinh viên thực hiện:**

Nguyễn Thị Thùy Linh - 21133051 Trần Phan Quốc - 21133108

Trần Thị Ngọc Trang - 21133109

**Chuyên ngành:** Kỹ thuật dữ liệu (Data Engineering)

**Môn học:** Lập Trình Web

**Nhận xét**

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

………………………………………………………………………………………

Thành phố Hồ Chí Minh, / /2023 Giảng viên hướng dẫn

(Tên và chữ ký)

Trương Thị Khánh Dịp

2

**MỤC LỤC**

[PHẦN MỞ ĐẦU 1](#_Toc154178442)

[1.1 Lý do chọn đề tài 1](#_Toc154178443)

[1.2 Mục tiêu và phạm vi 2](#_Toc154178444)

[1.2.1 Mục tiêu chính 2](#_Toc154178445)

[1.2.2 Phạm vi đồ án 2](#_Toc154178446)

[1.3 Đối tượng sử dụng 2](#_Toc154178447)

[PHẦN NỘI DUNG 4](#_Toc154178448)

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 4](#_Toc154178449)

[1.1. Lý thuyết 4](#_Toc154178450)

[1.1.1. Java Servlet 4](#_Toc154178451)

[1.1.2. AJAX (Asynchronous Javascript and XML) 5](#_Toc154178452)

[1.1.3. JDBC (Java Database Connectivity) 5](#_Toc154178453)

[1.1.4. Mô hình MVC (Model-View-Controller) 6](#_Toc154178454)

[1.1.5. Bootstrap và jQuery 6](#_Toc154178455)

[1.1.6. jsp (Java Server Page) 6](#_Toc154178456)

[1.1.7. Ngrok 7](#_Toc154178457)

[1.2. Thực hành 7](#_Toc154178458)

[CHƯƠNG 2. KHẢO SÁT HIỆN TRẠNG VÀ MÔ HÌNH HÓA YÊU CẦU 8](#_Toc154178459)

[2.1 KHẢO SÁT HIỆN TRẠNG 8](#_Toc154178460)

[2.2 XÁC ĐỊNH YÊU CẦU 8](#_Toc154178461)

[2.2.1 Yêu cầu chức năng 8](#_Toc154178462)

[2.2.2 Yêu cầu phi chức năng 9](#_Toc154178463)

[2.3 MÔ HÌNH HÓA YÊU CẦU 9](#_Toc154178464)

[2.3.1 Lược đồ UseCase 9](#_Toc154178465)

[2.3.2 Mô hình hóa yêu cầu 10](#_Toc154178466)

[CHƯƠNG 3. THIẾT KẾ PHẦN MỀM 21](#_Toc154178467)

[3.1 DATABASE DIAGRAM 21](#_Toc154178468)

[3.2 THIẾT KẾ CƠ SỞ DỮ LIỆU 21](#_Toc154178469)

[3.2.1 Grade 21](#_Toc154178470)

[3.2.2 Gunpla 21](#_Toc154178471)

[3.2.3 Order 22](#_Toc154178472)

[3.2.4 OrderDetail 23](#_Toc154178473)

[3.2.5 Role 23](#_Toc154178474)

[3.2.6 Status 23](#_Toc154178475)

[3.2.7 Store 24](#_Toc154178476)

[3.2.8 Users 24](#_Toc154178477)

[3.3 THIẾT KẾ GIAO DIỆN 26](#_Toc154178478)

[3.3.1 Form đăng ký (Register) 26](#_Toc154178479)

[3.3.2 Form đăng nhập (Login) 26](#_Toc154178480)

[3.3.3 Trang chủ (Home) 27](#_Toc154178481)

[3.3.4 Trang sản phẩm (Toy) 28](#_Toc154178482)

[3.3.5 Trang chi tiết sản phẩm (View Detail) 30](#_Toc154178483)

[3.3.6 Trang thông tin cá nhân (User Info) 31](#_Toc154178484)

[3.3.7 Trang giỏ hàng (Cart) 31](#_Toc154178485)

[3.3.8 Trang thanh toán (Order) 32](#_Toc154178486)

[3.3.9 Trang quản lý cửa hàng (My Store) 33](#_Toc154178487)

[3.3.10 Trang quản lý đơn hàng (Orders Management) 33](#_Toc154178488)

[3.3.11 Trang xem chi tiết đơn hàng (Details Order) 34](#_Toc154178489)

[3.3.12 Trang nạp tiền vào ví (Add Funds) 34](#_Toc154178490)

[3.3.13 Trang quản lý sản phẩm (My Toy) 35](#_Toc154178491)

[3.3.14 Trang thống kê (Statistic) 35](#_Toc154178492)

[3.3.15 Trang quản lý cửa hàng (Stores Management) 36](#_Toc154178493)

[3.3.16 Trang quản lý tài khoản (Users Management) 36](#_Toc154178494)

[3.3.17 Trang quản lý loại sản phẩm (Grade Management) 37](#_Toc154178495)

[3.3.18 Trang giới thiệu 37](#_Toc154178496)

[3.3.19 Trang chính sách thanh toán 38](#_Toc154178497)

[3.3.20 Trang chính sách giao hàng 38](#_Toc154178498)

[3.3.21 Trang chính sách đổi trả 39](#_Toc154178499)

[3.3.22 Trang chính sách bảo mật 39](#_Toc154178500)

[3.3.23 Trang điều khoản dịch vụ 40](#_Toc154178501)

[CHƯƠNG 4. KIỂM THỬ VÀ HƯỚNG PHÁT TRIỂN 41](#_Toc154178502)

[4.1 KIỂM THỬ 41](#_Toc154178503)

[4.2 HƯỚNG PHÁT TRIỂN 43](#_Toc154178504)

[TÀI LIỆU THAM KHẢO 44](#_Toc154178505)

# PHẦN MỞ ĐẦU

## Lý do chọn đề tài

Trong thời đại công nghệ hiện đại, sự phổ biến của thương mại điện tử và mua sắm trực tuyến đã mở ra những cơ hội mới và thách thức đối với nhiều doanh nghiệp. Để đáp ứng nhu cầu ngày càng tăng về trải nghiệm mua sắm thuận tiện và đa dạng, quyết định thực hiện đề tài về việc phát triển một website bán đồ chơi của cửa hàng ToyStore đã được đưa ra.

Việc xây dựng một nền tảng trực tuyến chính thức cho ToyStore không chỉ là một bước đi quan trọng để thích ứng với sự biến đổi của thị trường mà còn là cơ hội để tạo ra một trải nghiệm mua sắm và tiện ích cho khách hàng, đặc biệt là các bậc phụ huynh. Trang web không chỉ đơn giản là một cửa hàng trực tuyến, mà còn là không gian tương tác giữa khách hàng và sản phẩm.

Một trong những lợi ích quan trọng của việc triển khai website là khả năng mở rộng thị trường tiêu thụ và tiếp cận đến đối tượng khách hàng mới. Khách hàng không chỉ thuận lợi trong việc chọn lựa sản phẩm mà còn có cơ hội hiểu rõ hơn về nguồn gốc, chất lượng và giá trị của từng món đồ chơi để đưa đến cho trẻ em những niềm vui nho nhỏ.

Bằng cách này, việc quảng bá thương hiệu và tăng cường lòng tin từ phía khách hàng trở nên hiệu quả hơn. Mỗi sản phẩm trên trang web không chỉ là một món đồ chơi giải trí, mà còn là câu chuyện về nghệ thuật và sự tận tâm của cửa hàng đối với mỗi sản phẩm bán ra.

Hơn nữa, website còn là một công cụ quản lý kinh doanh hiệu quả, giúp doanh nghiệp theo dõi tồn kho, đánh giá hiệu suất bán hàng, và thu thập phản hồi từ khách hàng. Nhờ đó, ToyStore có cơ hội điều chỉnh chiến lược kinh doanh và nhanh chóng đáp ứng các xu hướng thị trường mới.

Vì vậy, việc thực hiện đề tài này không chỉ là một bước quan trọng để thích ứng với sự phát triển của thị trường mà còn là cơ hội để cửa hàng bán đồ chơi trẻ em gặt hái những thành công mới trong việc tạo ra trải nghiệm mua sắm trực tuyến đẳng cấp và thăng hoa.

## Mục tiêu và phạm vi

### **1.2.1 Mục tiêu chính**

Mục tiêu của dự án là xây dựng một nền tảng trực tuyến bán hàng với đầy đủ các chức năng, thuận tiện và dễ sử dụng cho người yêu thích mua sắm tại nhà.

* + - * Hiểu biết hình thức kinh doanh thương mại điện tử.
      * Hiểu biết về các giải pháp xây dựng một website thương mại điện tử.
      * Xây dựng một website mang lại thuận tiện cho người dùng, tiết kiệm thời gian khi có nhu cầu mua một sản phẩm.
      * Xây dựng Website giúp cho nhân viên quản lý thông tin một cách trực quan, thuận tiện.

### **1.2.2 Phạm vi đồ án**

Phạm vi của đồ án nằm trong giới hạn môn lập trình web.

* + - * Ngôn ngữ lập trình được sử dụng:

+ JavaScript

+ HTML

+ CSS

+ SQL

* + - * Phần mềm được sử dụng:

+ Eclipse (Tomcat 9.0)

+ SQL Server

+ Ngrok

* + - * Mô hình MVC

## Đối tượng sử dụng

Đối tượng sử dụng của đồ án web bán đồ chơi được chia thành bốn nhóm chính: Khách hàng tiềm năng (potential customers), Khách hàng (customer), Người bán (Vendor) và Quản trị viên (Admin). Dưới đây là mô tả chi tiết về các đối tượng này:

\* **Khách hàng tiềm năng :**

+ Đăng ký tài khoản.

+ Truy cập trang chủ và xem danh sách các sản phẩm đồ chơi.

+ Tìm kiếm và xem thông tin chi tiết về sản phẩm đó.

+ Xem thông tin của trang website, điều khoản và các chính sách.

**\* Khách hàng (User):**

+ Đăng nhập và đăng xuất tài khoản.

+ Nạp tiền vào tài khoản mua sắm.

+ Cập nhật thông tin cá nhân.

+ Truy cập trang chủ và xem danh sách các sản phẩm đồ chơi.

+ Tìm kiếm và xem thông tin chi tiết về sản phẩm đó.

+ Cập nhật giỏ hàng : thêm, xóa sản phẩm.

+ Mua sản phẩm

+ Xem thông tin của trang website, điều khoản và các chính sách.

**\* Người bán (User):**

+ Đăng nhập và đăng xuất tài khoản.

+ Nạp tiền vào tài khoản.

+ Xem cửa hàng cá nhân

+ Cập nhật thông tin cá nhân, thông tin cửa hàng.

+ Truy cập trang chủ và xem danh sách các sản phẩm đồ chơi.

+ Tìm kiếm và xem thông tin chi tiết về sản phẩm đó.

+ Quản lý sản phẩm: thêm, xóa, chỉnh sửa.

+ Quản lý hóa đơn

+ Xem thông tin của trang website, điều khoản và các chính sách.

+ Dừng hoạt động

**\* Admin (Quản trị viên):**

+ Đăng nhập và đăng xuất tài khoản.

+ Nạp tiền vào tài khoản.

+ Cập nhật thông tin cá nhân.

+ Xem doanh thu theo tháng.

+ Xem số tài khoản người dùng.

+ Xem và phê duyệt cửa hàng.

+ Thăng quyền cho user.

+ Quản lý loại sản phẩm.

+ Xem thông tin của trang website, điều khoản và các chính sách.

Qua đó, mô hình đối tượng sử dụng này giúp phân biệt rõ ràng các quyền của mỗi đối tượng, đồng thời đảm bảo rằng mọi người dùng có trải nghiệm tốt nhất trên trang mua sắm.

# PHẦN NỘI DUNG

# CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

## 1.1. Lý thuyết

### **1.1.1. Java Servlet**

Servlet có thể được mô tả bằng nhiều cách, tùy thuộc vào ngữ cảnh:

* Servlet là một công nghệ được sử dụng để tạo ra ứng dụng web.
* Servlet là một API cung cấp các interface và lớp bao gồm các tài liệu.
* Servlet là một thành phần web được triển khai trên máy chủ để tạo ra trang web động.

Có nhiều interface và các lớp trong API servlet như Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponse, …

Servlet thực hiện các nhiệm vụ chính sau đây:

Đọc dữ liệu rõ ràng do khách hàng (trình duyệt) gửi. Điều này bao gồm một mẫu HTML trên một trang Web hoặc nó cũng có thể đến từ một applet hoặc một chương trình khách hàng HTTP tùy chỉnh.

Đọc dữ liệu yêu cầu HTTP ẩn được gửi bởi khách hàng (trình duyệt). Điều này bao gồm các cookie, loại phương tiện truyền thông và các chương trình nén mà trình duyệt hiểu được, v.v.

Xử lý dữ liệu và tạo ra các kết quả. Quá trình này có thể yêu cầu nói chuyện với một cơ sở dữ liệu, thực hiện một cuộc gọi RMI hoặc CORBA, gọi một dịch vụ Web, hoặc tính trực tiếp phản hồi.

Gửi dữ liệu rõ ràng (tức là tài liệu) tới khách hàng (trình duyệt). Tài liệu này có thể được gửi bằng nhiều định dạng, bao gồm văn bản (HTML hoặc XML), nhị phân (hình ảnh GIF), Excel, v.v ...

Gửi phản hồi HTTP ẩn cho khách hàng (trình duyệt). Điều này bao gồm nói với trình duyệt hoặc các trình khách khác loại tài liệu đang được trả về (ví dụ, HTML), thiết lập cookie và các tham số bộ nhớ đệm, và các tác vụ khác.

Java Servlets là các lớp Java chạy bởi một máy chủ web có một trình thông dịch hỗ trợ đặc tả Java Servlet.

Servlets có thể được tạo ra bằng cách sử dụng các gói javax.servlet và javax.servlet.http , đây là một phần chuẩn của phiên bản Enterprise của Java, một phiên bản mở rộng của thư viện lớp Java hỗ trợ các dự án phát triển quy mô lớn.

Các lớp này thực hiện các đặc tả Java Servlet và JSP.

Các servlet Java đã được tạo ra và biên dịch giống như các lớp Java khác. Sau khi cài đặt gói servlet và thêm chúng vào Classpath của máy tính, bạn có thể biên dịch các servlet bằng trình dịch Java của JDK hoặc bất kỳ trình biên dịch hiện tại nào khác.

### **1.1.2. AJAX (Asynchronous Javascript and XML)**

AJAX là chữ viết tắt của cụm từ Asynchronous Javascript and XML. AJAX là phương thức trao đổi dữ liệu với máy chủ và cập nhật một hay nhiều phần của trang web, hoàn toàn không reload lại toàn bộ trang.

Ajax được viết bằng Javascript chạy trên client, tức là mỗi browser sẽ chạy độc lập hoàn toàn không ảnh hưởng lẫn nhau. Về mặt kỹ thuật, nó đề cập đến việc sử dụng các đối tượng XmlHttpRequest để tương tác với một máy chủ web thông qua Javascript.

AJAX là một trong những công cụ giúp chúng ta đem lại cho người dùng trải nghiệm tốt hơn. Khi cần một thay đổi nhỏ thì sẽ không cần load lại cả trang web, làm trang web phải tải lại nhiều thứ không cần thiết.

Những lợi ích mà AJAX mang lại:

AJAX được sử dụng để thực hiện một callback. Được dùng để thực hiện việc truy xuất dữ liệu hoặc lưu trữ dữ liệu mà không cần phải reload lại toàn bộ trang web. Với những server nhỏ thì việc này cũng tiết kiệm được băng thông cho chúng ta hơn.

Cần gì thì chỉ gửi dữ liệu phần đó, load lại 1 phần nhỏ để cập nhật thông tin chứ không load cả trang. Bằng cách này thì có thể giảm thiểu được tốc độ tải trang giúp người dùng có trải nghiệm tốt hơn.

### **1.1.3. JDBC (Java Database Connectivity)**

Thuật ngữ JDBC là viết tắt của Java Database Connectivity. JDBC là một thông số kỹ thuật từ Sun microsystems. Java JDBC là một java API được sử dụng để kết nối và thực hiện truy vấn với cơ sở dữ liệu. JDBC API sử dụng trình điều khiển jdbc để kết nối với cơ sở dữ liệu.

Sử dụng JDBC, chúng ta có thể viết các chương trình cần thiết để truy cập cơ sở dữ liệu. JDBC và trình điều khiển cơ sở dữ liệu có khả năng truy cập cơ sở dữ liệu và bảng tính. API JDBC cũng hữu ích trong việc truy cập dữ liệu doanh nghiệp được lưu trữ trong cơ sở dữ liệu quan hệ (RDB)

### **1.1.4. Mô hình MVC (Model-View-Controller)**

Mô hình Model-View-Controller (MVC) là một mẫu kiến ​​trúc phân tách một ứng dụng thành ba thành phần logic chính Model, View và Controller. Do đó viết tắt MVC. Mỗi thành phần kiến ​​trúc được xây dựng để xử lý khía cạnh phát triển cụ thể của một ứng dụng. MVC tách lớp logic nghiệp vụ và lớp hiển thị ra riêng biệt. Ngày nay, kiến ​​trúc MVC đã trở nên phổ biến để thiết kế các ứng dụng web cũng như ứng dụng di động.

MVC quan trọng bao gồm:

* Model: Nó bao gồm tất cả dữ liệu và logic liên quan của nó.
* View: Trình bày dữ liệu cho người dùng hoặc xử lý tương tác của người dùng.

- Controller: Là phần quan trọng nhất trong mô hình, nó liên kết phần Model và View.

### **1.1.5. Bootstrap và jQuery**

*Bootstrap* là một bộ sưu tập miễn phí của các *mã nguồn mở* và công cụ dùng để tạo ra một mẫu website hoàn chỉnh. Với các thuộc tính về giao diện được quy định sẵn như kích thước, màu sắc, độ cao, độ rộng…, các designer có thể sáng tạo nhiều sản phẩm mới mẻ nhưng vẫn tiết kiệm thời gian khi làm việc với framework này trong quá trình *thiết kế giao diện website*.

jQuery không phải là một ngôn ngữ lập trình riêng biệt mà hoạt động liên kết với JavaScript. Với jQuery, bạn có thể làm được nhiều việc hơn mà lại tốn ít công sức hơn. jQuery cung cấp các API giúp việc duyệt tài liệu HTML, hoạt ảnh, xử lý sự kiện và thao tác AJAX đơn giản hơn. jQuery hoạt động tốt trên nhiều loại trình duyệt khác nhau. Một trong những đối thủ nặng ký của jQuery đó là JS Framework.

### **1.1.6. jsp (Java Server Page)**

JSP là từ viết tắt của Java Server Pages. Đây là một công nghệ phía máy chủ giúp các lập trình viên tạo ra các trang web bằng Java.

Đây là một tính năng được phát triển bởi Sun microsystems để cải tiến cho Servlets, che đậy tất cả các lỗ hổng của Servlets.

Nếu bạn chưa biết, Servlets được sử dụng để chứa cả code logic và code giao diện. Nhưng trong JSP, hai thành phần này được tách ra. (Tách biệt logic nghiệp vụ và phần trình bày để dễ quản lý)

Trong JSP, chúng ta định nghĩa thẻ bằng "<% %>".

Sử dụng JSP, chúng ta có thể triển khai API Java theo mặc định vào chương trình web.

Và không chỉ API Java, JSP còn hoạt động với các thẻ HTML và XML, điều mà trước đây không thể thực hiện được.

Tương tự, JDBC cũng có thể sử dụng được nếu chúng ta sử dụng JSP.

Do đó, khả năng truy vấn động của một cơ sở dữ liệu là có thể.

Ngoài ra, JSP cung cấp nhiều tính năng hơn như:

* Truy vấn cơ sở dữ liệu và lấy thông tin.
* Chia sẻ thông tin giữa hai trang.
* Kiểm soát truy cập.

### **1.1.7. Ngrok**

Ngrok là công cụ tạo đường hầm (tunnel) giữa localhost của bạn và internet. Giúp các thiết bị có kết nối internet có thể truy cập được localhost của bạn thông qua custom domain của ngrok

Ví dụ: ngrokdoamain => localhost:8080.

Ngrok giúp:

* Chạy demo dự án cho khách hàng xem từ chính máy của bạn mà không cần deploy lên server.
* Thêm cách test responsive trên smart phone một cách dễ dàng thông qua URL mà ngrok cung cấp.
* Hỗ trợ http, https, tcp.
* Hỗ trợ IP whitelist.

Sử dụng Ngrok cho phép nhà phát triển cung cấp truy cập internet công cộng đến ứng dụng đang chạy trên máy cục bộ của họ, điều này tạo điều kiện thuận lợi cho việc chia sẻ, kiểm thử và phát triển sản phẩm. Ngrok cung cấp một cách an toàn để thử nghiệm các webhook và các giao diện lập trình ứng dụng (APIs), giúp tái tạo môi trường sản xuất và theo dõi lỗi hiệu quả. Hơn nữa, nó dễ sử dụng, linh hoạt và không đòi hỏi quy trình triển khai phức tạp, vì vậy, khi có nhu cầu public url trong các ứng dụng web cụ bộ, thì ngrok là một trong những lựa chọn.

## 1.2. Thực hành

- Thiết kế trang web bằng ngôn ngữ HTML, CSS, Javascript, Bootstrap, jQuery.

- Giao tiếp với cơ sở dữ liệu bằng JDBC.

- Thực hiện đồ án theo mô hình MVC.

- Viết code các chức năng của trang web bằng Java Servlet, Ajax

# CHƯƠNG 2. KHẢO SÁT HIỆN TRẠNG VÀ MÔ HÌNH HÓA YÊU CẦU

## 2.1 KHẢO SÁT HIỆN TRẠNG

Qua khảo sát thực tế việc bán hàng thông qua một trang web được quản lý như sau:

* Quản lý khách hàng:

Mỗi khách hàng được quản lý các thông tin sau đây: Họ tên, địa chỉ, điện thoại, email, tên đăng nhập, mật khẩu. Ngoài ra khách hàng là người bán sản phẩm thì được phân quyền rõ ràng.

* Quản lý sản phẩm:

Mỗi mặt hàng được quản lý những thông tin: Tên đặt hàng, giá, số lượng, hình ảnh, mô tả.

* Quản lý doanh thu:

Sau mỗi lần khách hàng thanh toán thì thì cập nhật doanh thu cho quyền người quản lý.

* Quản lý cửa hàng:

Phê duyệt cho cửa hàng muốn bán sản phẩm tại trang web.

* Quá trình đặt hàng của khách hàng:

Khách hàng xem và lựa chọn món đồ chơi muốn mua. Sau khi lựa chọn xong, sẽ tiến hành lập đơn hàng của khách. Sau khi tiếp nhận yêu cầu trên, tiến hành làm hóa đơn và thanh toán tiền.

## 2.2 XÁC ĐỊNH YÊU CẦU

### **2.2.1 Yêu cầu chức năng**

* Xem danh sách sản phẩm: website cung cấp các thông tin chi tiết về sản phẩm như tên, loại, mô tả chi tiết về mặt hàng cho người dùng.
* Nạp tiền vào tài khoản: website cung cấp phương thức thanh toán online cho khách hàng để tiện cho việc mua sắm bằng cách nạp tiền vào tài khoản cá nhân thông qua số thẻ ngân hàng.
* Mua hàng: người dùng sau khi lựa chọn các món đồ chơi muốn mua, bỏ vào giỏ hàng. Sau khi thực hiện các thao tác thêm, xóa,... thì hệ thống xử lý giao dịch mua hàng.
* Thanh toán: khách hàng khi đã nhấn mua hàng sẽ được chuyển đến trang thanh toán và điền các thông tin cần thiết cho việc nhận hàng và trả tiền thì hoàn tất việc thanh toán.
* Quản lý doanh thu: quản lý là người có quyền xem doanh thu khi khách hàng hoàn tất việc thanh toán đơn hàng của mình.
* Quản lý người dùng: quản lý là người có quyền xem số lượng khách hàng và người dùng (người bán hàng) trên website cửa hàng.
* Quản lý cửa hàng: quản lý là người có quyền cấp phép cho cửa hàng đó còn hoạt động hay không, nếu không thì các sản phẩm của cửa hàng đó không thể bán ra được.
* Quản lý loại sản phẩm: quản lý là người có quyền cập nhật, xóa, thêm, sửa các loại sản phẩm hiện có trên website cũng như ra mắt dòng sản phẩm mới.
* Quản lý quyền người dùng: quản lý là người có quyền thăng cấp cho người dùng một thứ bậc trong phân quyền của hệ thống cửa hàng.

### **2.2.2 Yêu cầu phi chức năng**

* Hiệu suất: Hệ thống cần xử lý nhiều giao dịch cùng lúc để đáp ứng nhu cầu đặt đặt hàng trong các ngày giảm giá.
* Bảo mật: Thông tin cá nhân và thông tin thanh toán của người dùng cần được mã hóa và bảo mật.
* Khả năng mở rộng: Hệ thống cần có khả năng mở rộng để hỗ trợ thêm rạp chiếu phim, cải thiện hiệu suất và tính năng mới trong tương lai.
* Đáng tin cậy và ổn định: Đảm bảo hệ thống hoạt động ổn định, đáng tin cậy, ít gặp lỗi và có khả năng khôi phục sau sự cố.

## 2.3 MÔ HÌNH HÓA YÊU CẦU

### **2.3.1 Lược đồ UseCase**



*Hình 1. Sơ đồ UseCase giữa người dùng và người quản lý*

### **2.3.2 Mô hình hóa yêu cầu**

#### **2.3.2.1 Mô hình hóa website phía khách hàng tiềm năng và khách hàng**

|  |  |
| --- | --- |
| **Use case: Register** | |
| **Actor** | Guest |
| **Short Description:** | Đăng ký thành viên để mua hàng |
| **Pre-Condition:** | Không có |
| **Post-Condition:** | Đăng ký thành công tài khoản member |

|  |  |
| --- | --- |
| **Main Flow:** | 1. Chọn mục Register 2. Điền thông tin (username, password, tên, số điện thoại, địa chỉ) 3. Xác nhận |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | 3.a. Thông tin điền vào không hợp lệ hoặc đã tồn tại username, hiện thông báo yêu cầu nhập lại phần bị sai, quay lại bước 2 |

|  |  |
| --- | --- |
| **Use case: Login** | |
| **Actor** | User, Admin |
| **Short Description:** | Truy cập tài khoản |
| **Pre-Condition:** | Register |
| **Post-Condition:** | Đăng nhập thành công |
| **Main Flow:** | 1. Chọn mục Login 2. Nhập tên đăng nhập, mật khẩu đăng nhập 3. Chọn type của account là “User” 4. Nhấn nút Submit 5. Hiển thị giao diện của User |
| **Alternate Flow(s):** | - Login vào Admin  3.b. Chọn type là “admin”  5.b. Hiển thị giao diện của Admin |
| **Exception Flow(s):** | 4.c. Tên tài khoản, mật khẩu hoặc type account sai thông báo lỗi -> quay lại bước 2 |

|  |  |
| --- | --- |
| **Use case: Logout** | |
| **Actor** | User, Admin |
| **Short Description:** | Đăng xuất tài khoản |
| **Pre-Condition:** | Login |
| **Post-Condition:** | Đăng xuất thành công |
| **Main Flow:** | 1.Chọn mục Logout |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | Không có |

|  |  |
| --- | --- |
| **Use case: Search Toy** | |
| **Actor** | Guest, User |
| **Short Description:** | Tìm kiếm Toy theo từ khóa |
| **Pre-Condition:** | Không có |
| **Post-Condition:** | Hiển thị kết quả tìm kiếm |
| **Main Flow:** | 1. Truy cập vào website 2. Nhập từ khóa vào thanh tìm kiếm 3. Nhấn enter hoặc nút Search |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | 3.a. Nếu không có kết quả thì sẽ hiển thị "Not found" |

|  |  |
| --- | --- |
| **Use case: View Toy** | |
| **Actor** | Guest, User |
| **Short Description:** | Hiển thị thông tin chi tiết về Toy |

|  |  |
| --- | --- |
| **Pre-Condition:** | Không có |
| **Post-Condition:** | Hiển thị trang chi tiết Toy |
| **Main Flow:** | 1. Chọn Toy đang hiển thị trên màn hình 2. Chuyển đến trang chi tiết sản phẩm |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | Không có |

|  |  |
| --- | --- |
| **Use case: Account Management** | |
| **Actor** | User, Admin |
| **Short Description:** | Cập nhật thông tin cá nhân của tài khoản |
| **Pre-Condition:** | Login |
| **Post-Condition:** | Cập nhật thành công các thông tin mà người dùng muốn  thay đổi |
| **Main Flow:** | 1. Vào mục thông tin cá nhân có icon hình người 2. Thay đổi lại các thông tin cần cập nhật 3. Nhấn nút submit 4. Hệ thống hiển thị lại thông tin trên trang đúng với thông tin mới cập nhật |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | 3.a Thông tin điền vào không hợp lệ, hiện thông báo  yêu cầu nhập lại phần bị sai -> Quay lại bước 2 |

|  |  |
| --- | --- |
| **Use case: Add Funds** | |
| **Actor** | User |
| **Short Description:** | Nạp tiền vào ví mua hàng |
| **Pre-Condition:** | Login |

|  |  |
| --- | --- |
| **Post-Condition:** | Thêm số tiền cần nạp vào ví |
| **Main Flow:** | 1. Chọn nút Balance: [số tiền] ở trang thông tin của người dùng 2. Nhập số thẻ, CVC và amount 3. Nhấn submit 4. Quay trở lại giao diện thông tin của người dùng |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | 3.a. Thông tin điền vào không hợp lệ, hiện thông báo yêu cầu  nhập lại phần bị sai -> quay lại bước 2 |

|  |  |
| --- | --- |
| **Use case: Cart Management** | |
| **Actor** | User |
| **Short Description:** | Thêm item trong giỏ hàng |
| **Pre-Condition:** | Login |
| **Post-Condition:** | Giỏ hàng được cập nhật |
| **Main Flow:** | A – Add to cart  A1. Nhấp vào thẻ Toy trên web A2. Dẫn tới trang chi tiết Toy A3. Nhấn vào nút Add to cart A4. Hiển thị giỏ hàng  B- Adjust quantity  B1. Nhấn nút giỏ hàng  B2. Nhấn dấu ‘+’ tăng số lượng  B3. Nhấn dấu ‘-’ giảm số lượng, nếu số lượng = 1 thì remove khỏi giỏ hàng  C – Remove from cart C1. Nhấn nút giỏ hàng  C2. Nhấn nút remove ở bên phải ngoài cùng của item trong order  C3. Load lại giỏ hàng |

|  |  |
| --- | --- |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | Không có |

|  |  |
| --- | --- |
| **Use case: Purchase** | |
| **Actor** | User |
| **Short Description:** | Thanh toán giỏ hàng |
| **Pre-Condition:** | Login |
| **Post-Condition:** | Cập nhật trạng thái order thành đã thanh toán |
| **Main Flow:** | 1. Nhấn nút giỏ hàng 2. Xác nhận các món hàng 3. Nhấn nút Purchase 4. Chuyển đến trang thanh toán 5. Điền đầy đủ thông tin 6. Nhấn Order 7. Trừ thành tiền vào ví của customer 8. Thông báo thanh toán thành công 9. Làm trống giỏ hàng |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | 3.a. Nếu số dư ví không đủ thì báo không đủ số dư -> quay lại bước 2. |

#### **2.3.2.2 Mô hình hóa website phía người bán và quản trị viên**

|  |  |
| --- | --- |
| **Use case: Store Management** | |
| **Actor** | Admin |
| **Short Description:** | Quản lý thông tin cửa hàng |

|  |  |
| --- | --- |
| **Pre-Condition:** | Login |
| **Post-Condition:** | Chỉnh sửa thành công thông tin cửa hàng |
| **Main Flow:** | * Với Admin   A1. Chọn Store Management ở trang quản lý của Admin A2. Hiển thị danh sách các Store  A3. Chọn Store chỉnh sửa A4. Nhấn nút Approve Store  A5. Load lại thông tin các Store   * Với User   U1. Chọn My Store  U2. Hiển thị thông tin Store U3. Chọn chỉnh sửa thông tin  U4. Chỉnh sửa lại các thông tin cần thay đổi U5. Nhấn Update  U6. Hiển thị thông báo cập nhật thành công |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | U5.a. Thông tin nhập không hợp lệ -> hiển thị thông báo nhập lại -> quay lại bước V4 |

|  |  |
| --- | --- |
| **Use case: Statistics** | |
| **Actor** | Admin |
| **Short Description:** | Thống kê số liệu |
| **Pre-Condition:** | Login |
| **Post-Condition:** | Hiển thị các số liệu |
| **Main Flow:** | 1. Chọn Statistics ở trang quản lý của Admin 2. Hiển thị giao diện số liệu |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | Không có |

|  |  |
| --- | --- |
| **Use case: User Management** | |
| **Actor** | Admin |
| **Short Description:** | Quản lý tài khoản user |
| **Pre-Condition:** | Login |
| **Post-Condition:** | Thăng quyền cho tài khoản User lên Admin |
| **Main Flow:** | 1. Chọn User management ở trang quản lý của Admin 2. Chọn user cần thăng chức 3. Nhấn nút Promote |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | Không có |

|  |  |
| --- | --- |
| **Use case: Grade Management** | |
| **Actor** | Admin |
| **Short Description:** | Quản lý Grade |
| **Pre-Condition:** | Login |
| **Post-Condition:** | Chỉnh sửa thành công Grade |
| **Main Flow:** | 1. Chọn Grade Management ở trang quản lý của Admin 2. Hiển thị giao diện thông tin Grade 3. Nhập thông tin Grade mới 4. Nhấn nút Add new Grade 5. Load lại thông tin các Grade |
| **Alternate Flow(s):** | * 1. Chỉnh sửa thông tin Grade      1. Chọn grade cần chỉnh sửa từ bảng bên dưới      2. Nhập lại thông tin Grade |

|  |  |
| --- | --- |
|  | * + 1. Nhấn nút Save     2. Load lại thông tin các Grade   1. Xóa Grade      1. Chọn Grade cần xóa      2. Nhấn nút xóa      3. Load lại thông tin các Grade |
| **Exception Flow(s):** | Không có |

|  |  |
| --- | --- |
| **Use case: Toy Management** | |
| **Actor** | User |
| **Short Description:** | Quản lý Toy |
| **Pre-Condition:** | Login, User có Store |
| **Post-Condition:** | Thêm/xóa thông tin Toy thành công |
| **Main Flow:** | 1. Chọn My Toy trong giao diện My Store 2. Hiển thị danh sách Toy của cửa hàng 3. Chọn Toy cần chỉnh sửa thông tin 4. Nhập thông tin cần chỉnh sửa 5. Nhấn nút Save 6. Hiển thị thông báo chỉnh sửa thành công |
| **Alternate Flow(s):** | * 1. Xóa Toy      1. Chọn Toy cần xóa      2. Nhấn nút remove      3. Hiển thị thông báo xóa thành công   2. Thêm Toy      1. Chọn thêm Toy      2. Nhập thông tin Toy      3. Nhấn nút Add      4. Hiển thị thông báo thêm thành công |
| **Exception Flow(s):** | Không có |

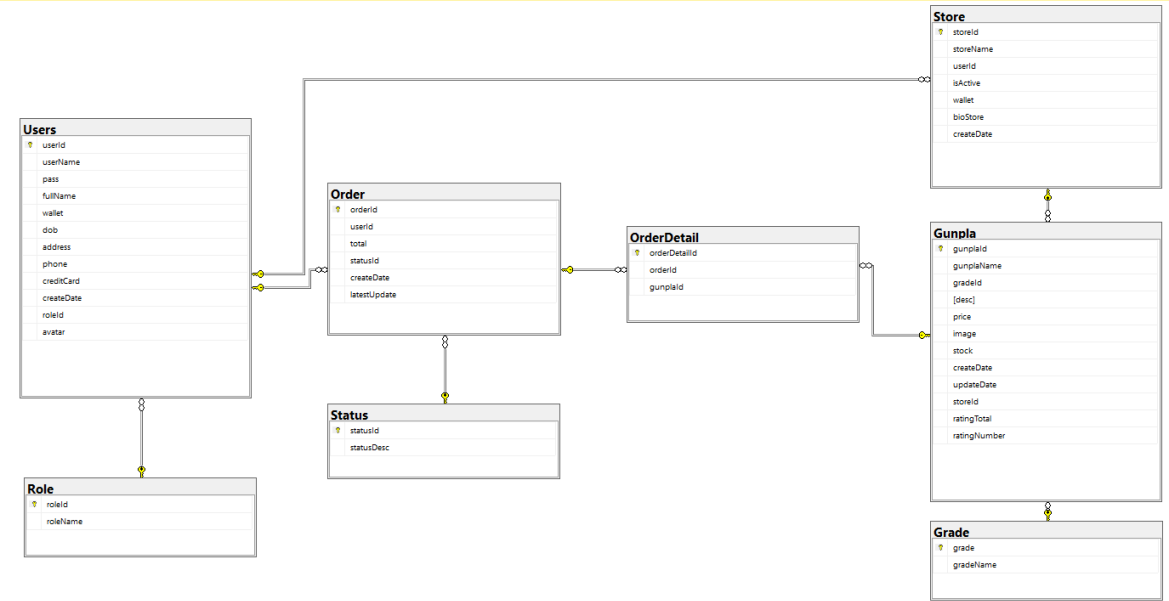
|  |  |
| --- | --- |
| **Use case: View MyStore** | |
| **Actor** | User |
| **Short Description:** | Hiển thị thông tin chi tiết về MyStore |
| **Pre-Condition:** | Login, có Store |
| **Post-Condition:** | Hiển thị trang chi tiết MyStore |
| **Main Flow:** | 1. Chọn MyStore đang hiển thị trên thanh của user 2. Chuyển đến trang chi tiết MyStore |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | Không có |

|  |  |
| --- | --- |
| **Use case: View Store Orders** | |
| **Actor** | User |
| **Short Description:** | Xem các thông tin đơn giao hàng |
| **Pre-Condition:** | Login, User có Store |
| **Post-Condition:** | Hiện tất cả đơn hàng |
| **Main Flow:** | 1. Chọn My Store 2. Chọn Store Orders cần hiển thị thông tin 3. Hiển thị thông tin đơn hàng |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | Không có |

|  |  |
| --- | --- |
| **Use case: View Detail** | |
| **Actor** | User |
| **Short Description:** | Xem thông tin chi tiết đơn giao hàng |
| **Pre-Condition:** | Login, User có Store |
| **Post-Condition:** | Hiện chi tiết đơn hàng |
| **Main Flow:** | 1. Chọn My Store 2. Chọn Store Orders cần hiển thị thông tin 3. Chọn View để xem chi tiết 4. Hiển thị thông tin chi tiết đơn hàng |
| **Alternate Flow(s):** | Không có |
| **Exception Flow(s):** | Không có |

# CHƯƠNG 3. THIẾT KẾ PHẦN MỀM

## 3.1 DATABASE DIAGRAM



*Hình 2. Mô hình quan hệ được ánh xạ từ ERD*

## 3.2 THIẾT KẾ CƠ SỞ DỮ LIỆU

### **3.2.1 Grade**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Mô tả** | **Kiểu dữ liệu** |
| 1 | grade | Id của grade  + Primary-key | String |
| 2 | gradeName | Mô tả của grade  + Required | String |

### **3.2.2 Gunpla**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Mô tả** | **Kiểu dữ liệu** |
| 1 | gunplaId | Id của đồ chơi  + Primary-key  +Auto generate | ObjectId |
| 2 | gunplaName | Tên của sản đồ chơi  +Required | String |

|  |  |  |  |
| --- | --- | --- | --- |
| 3 | gradeId | Id của grade  +Required  +Ref : Grade | String |
| 4 | desc | Mô tả chi tiết sản phẩm | String |
| 5 | price | Giá | Int |
| 6 | image | Link ảnh | String |
| 7 | stock | Số lượng trong kho | Int |
| 8 | createDate | Ngày thêm vào | Date |
| 9 | updateDate | Ngày cập nhật gần nhất | Date |
| 10 | storeId | Id store  +Required  +Ref : Store | ObjectId |
| 11 | ratingTotal | Tổng rating | Int |
| 12 | ratingNumber | Số lượng rating | Int |

### **3.2.3 Order**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Mô tả** | **Kiểu dữ liệu** |
| 1 | orderId | Id của order  + Primary-key  +Auto generate | ObjectId |
| 2 | userId | Id user  +Required  +Ref : User | ObjectId |
| 3 | total | Tổng tiền hóa đơn | Int |
| 4 | statusId | Id status  +Required | ObjectId |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | +Ref : Status |  |
| 5 | createDate | Ngày tạo  +Required | Date |
| 6 | latestUpdate | Ngày cập nhật gần nhất | Date |

### **3.2.4 OrderDetail**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Mô tả** | **Kiểu dữ liệu** |
| 1 | orderDetailId | Id của order detail  + Primary-key  + Auto generate | ObjectId |
| 2 | orderId | Id của order  + Required  + ref : Order | ObjectId |
| 3 | gunplaId | Id của gunpla  + Required  + ref : Gunpla | ObjectId |

### **3.2.5 Role**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Mô tả** | **Kiểu dữ liệu** |
| 1 | roleId | Id của role  + Primary-key  + Auto generate | ObjectId |
| 2 | roleName | Tên chức vụ  + Required | String |

### **3.2.6 Status**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Mô tả** | **Kiểu dữ liệu** |
| 1 | statusId | Id của status  + Primary-key  + Auto generate | ObjectId |
| 2 | statusDesc | Mô tả chi tiết trạng thái đơn hàng | String |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | + Required |  |

### **3.2.7 Store**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Mô tả** | **Kiểu dữ liệu** |
| 1 | storeId | Id của store  + Primary-key  + Auto generate | ObjectId |
| 2 | storeName | Tên cửa hàng  + Required | String |
| 3 | userId | Id của order  + Required  + ref : User | ObjectId |
| 4 | isActive | Tình trạng hoạt động  + Required | Boolean |
| 5 | wallet | Ví  +Default = 0 | Int |
| 6 | bioStore | Mô tả cửa hàng | String |
| 7 | createDate | Ngày tạo cửa hàng  + Required | Date |

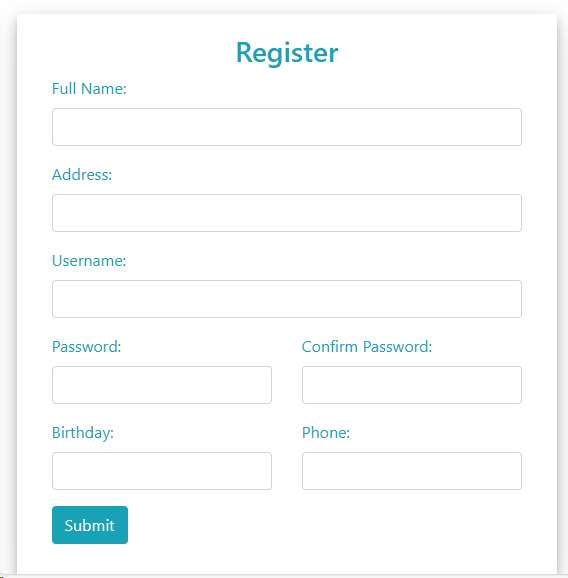
### **3.2.8 Users**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên field** | **Mô tả** | **Kiểu dữ liệu** |
| 1 | userId | Id của users  + Primary-key  + Auto generate | ObjectId |
| 2 | userName | Tên đăng nhập  + Required | String |
| 3 | pass | Mật khẩu  + Required | String |
| 4 | fullName | Tên đầy đủ của user | String |
| 5 | wallet | Ví  +Default = 0 | Int |

|  |  |  |  |
| --- | --- | --- | --- |
| 6 | dob | Ngày sinh  + Required | Date |
| 7 | address | Địa chỉ | String |
| 8 | phone | Số điện thoại | String |
| 9 | creditCard | Số thẻ | String |
| 10 | createDate | Ngày tạo tài khoản  + Required | Date |
| 11 | roleId | Id của order  + Required  + ref : Role | ObjectId |
| 12 | avatar | Link ảnh | String |

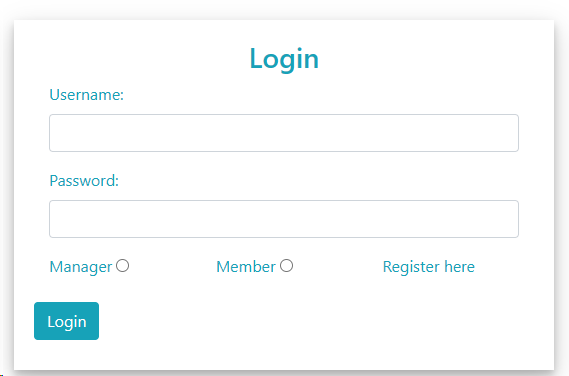
## 3.3 THIẾT KẾ GIAO DIỆN

### **3.3.1 Form đăng ký (Register)**



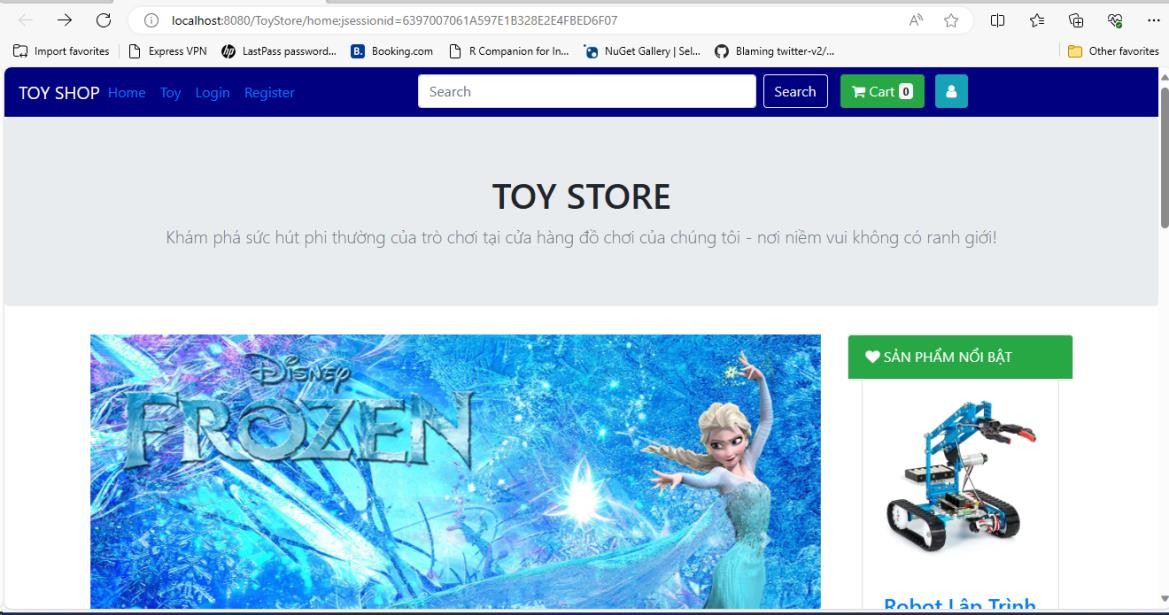
*Hình 3. Giao diện đăng ký - Register*

### **3.3.2 Form đăng nhập (Login)**

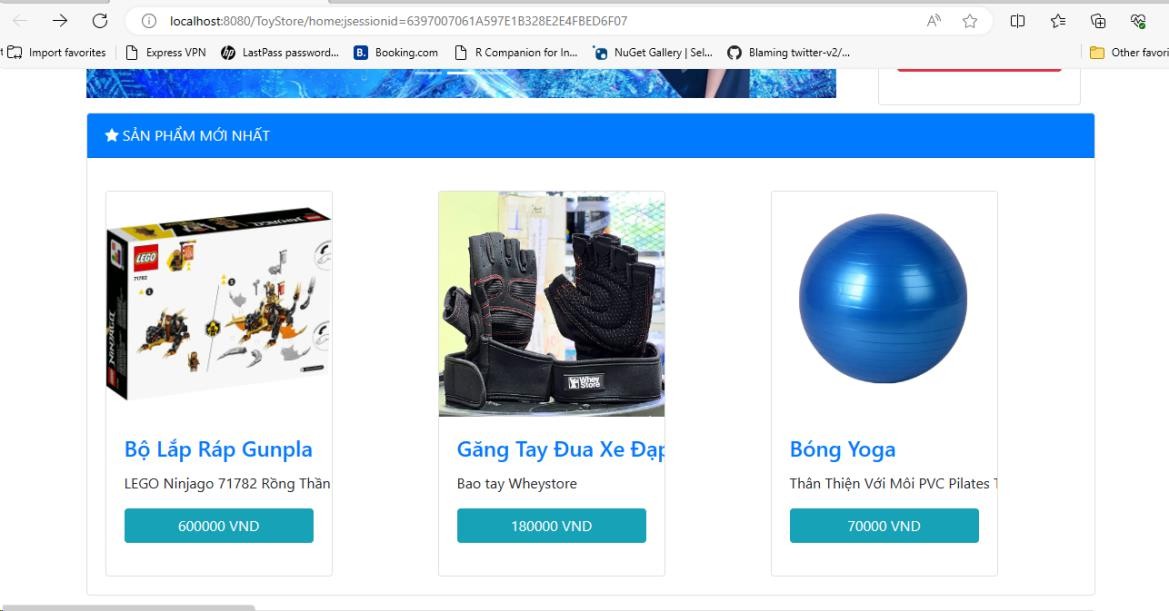


*Hình 4. Giao diện đăng nhập - login*

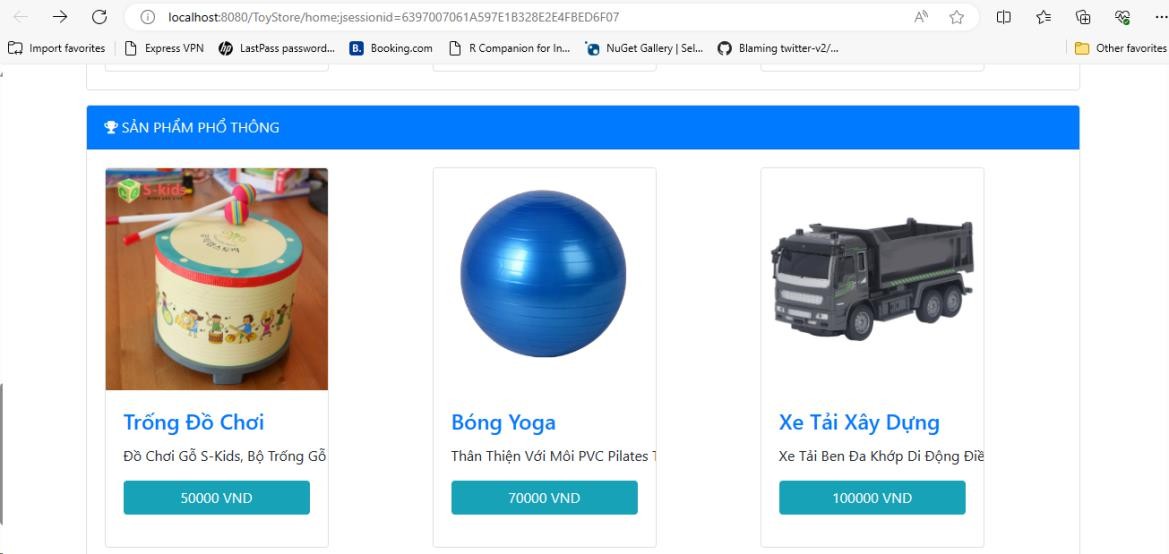
### **3.3.3 Trang chủ (Home)**



*Hình 5. Giao diện Trang chủ - Home(1)*



*Hình 6. Giao diện Trang chủ - Home(2)*

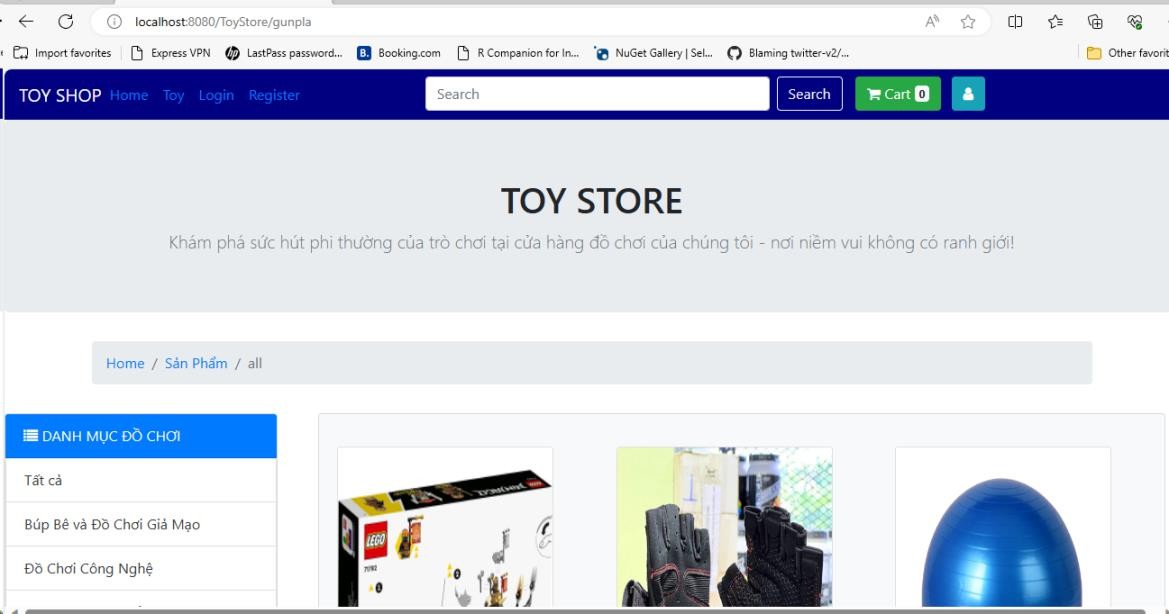


*Hình 7. Giao diện Trang chủ - Home(3)*

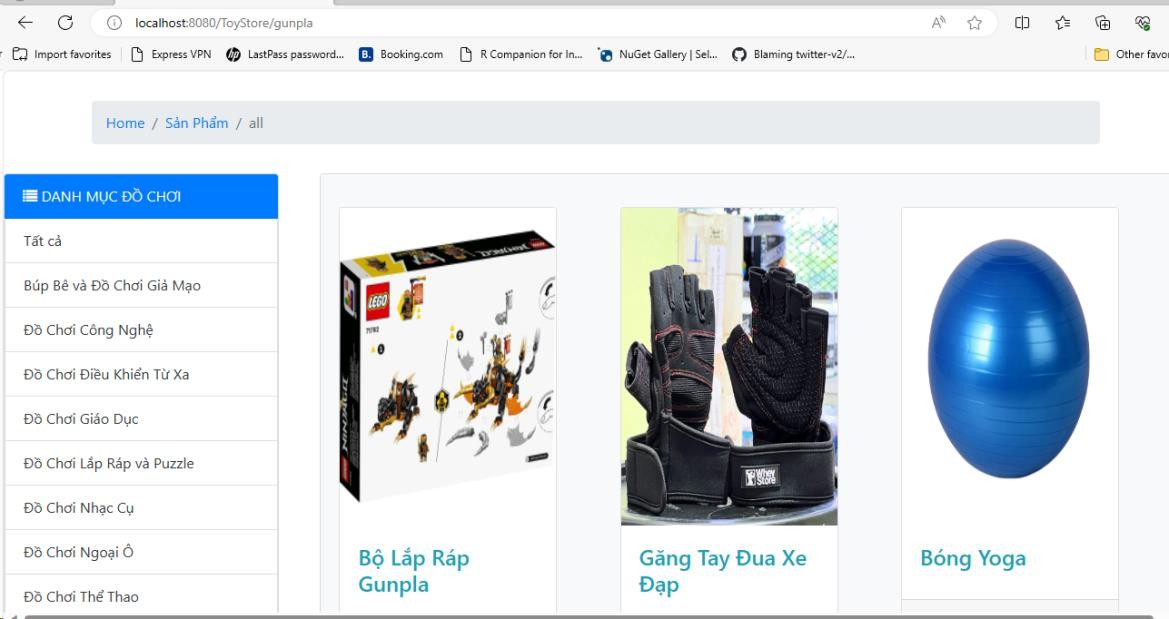


*Hình 8. Giao diện Trang chủ - Home(4)*

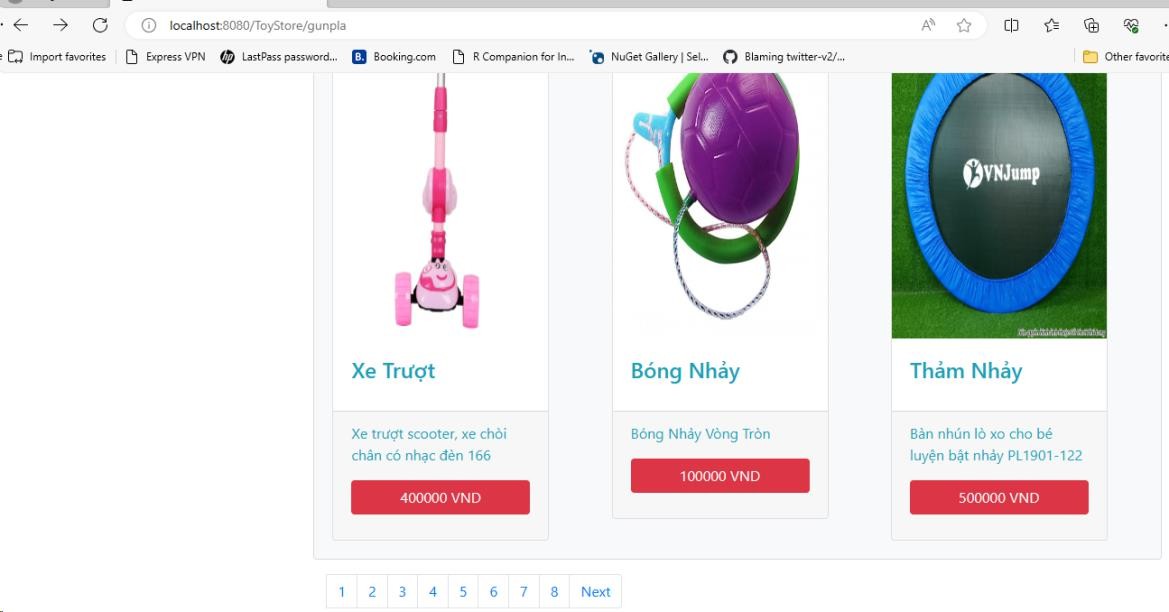
### **3.3.4 Trang sản phẩm (Toy)**



*Hình 9. Giao diện sản phẩm - Toy(1)*

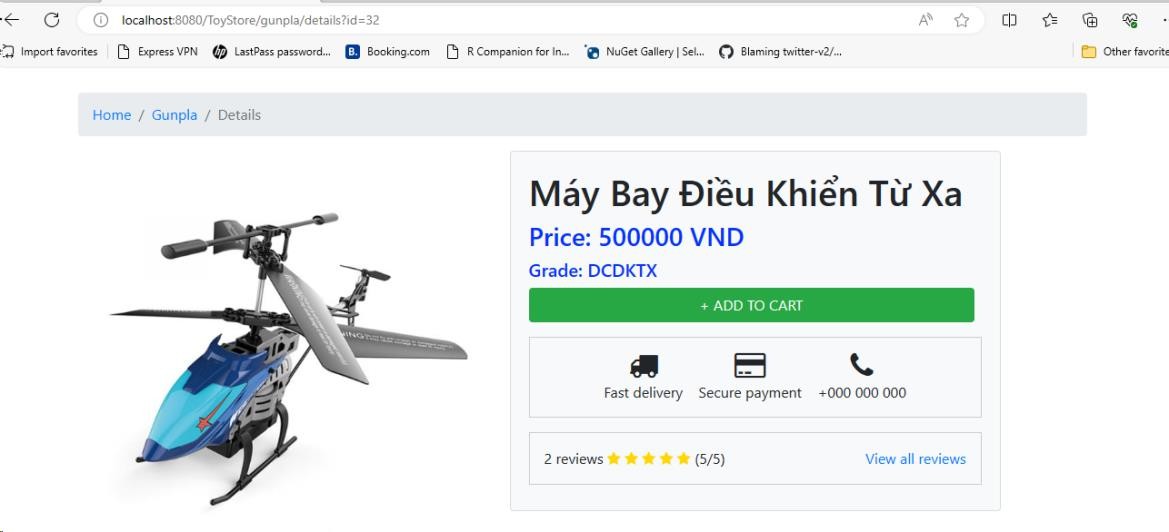


*Hình 10. Giao diện sản phẩm - Toy(2)*

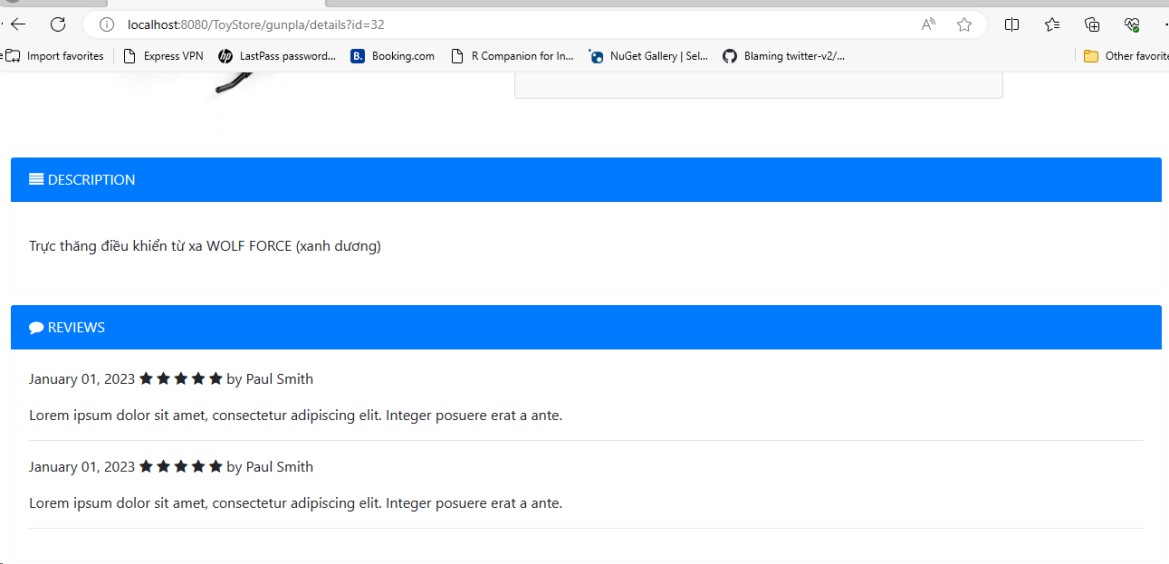


*Hình 11. Giao diện sản phẩm - Toy(3)*

### **3.3.5 Trang chi tiết sản phẩm (View Detail)**

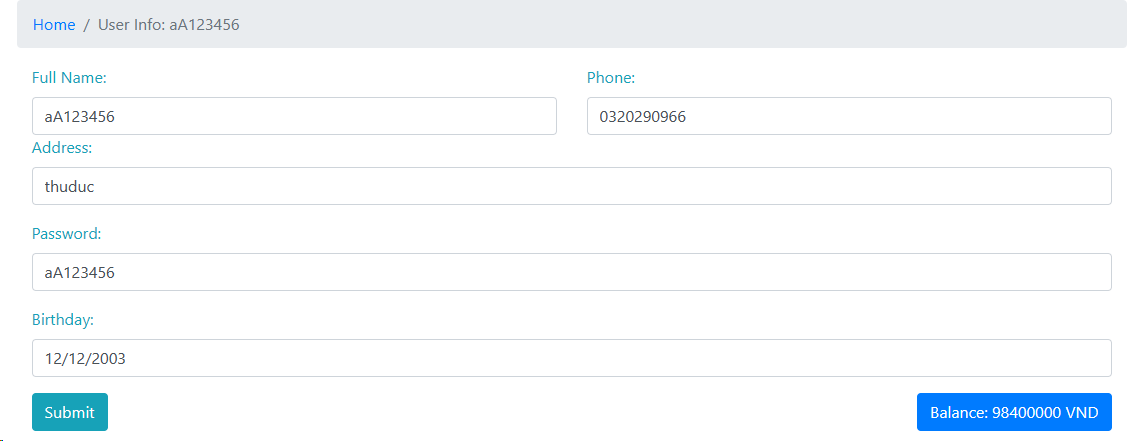


*Hình 12. Giao diện chi tiết sản phẩm - View Detail (1)*



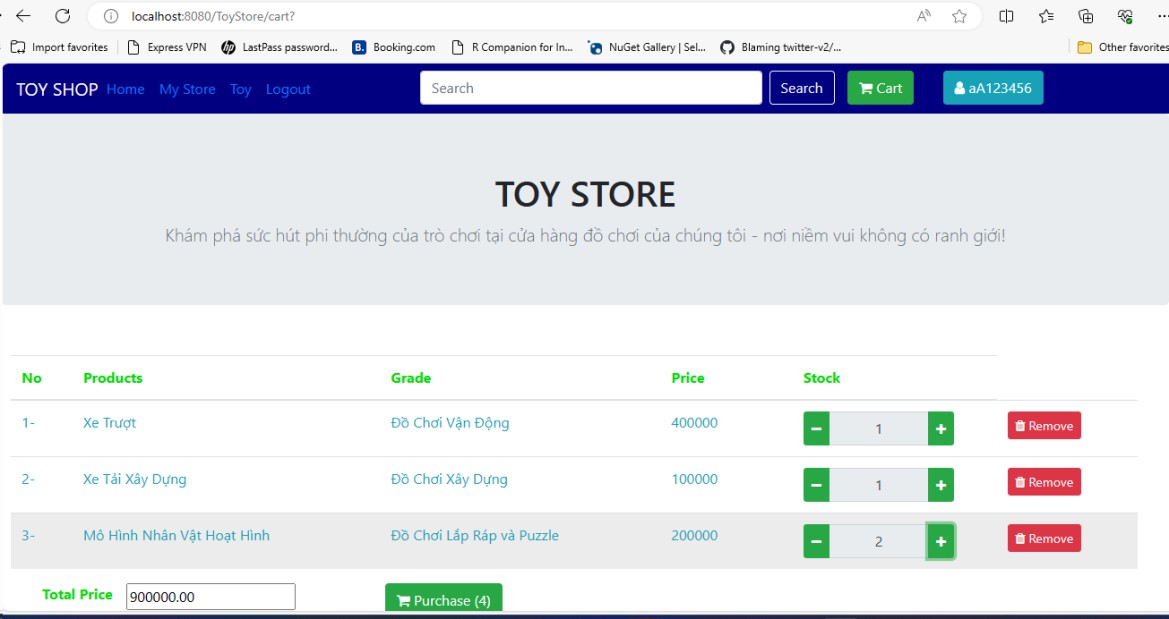
*Hình 13. Giao diện chi tiết sản phẩm - View Detail (2)*

### **3.3.6 Trang thông tin cá nhân (User Info)**



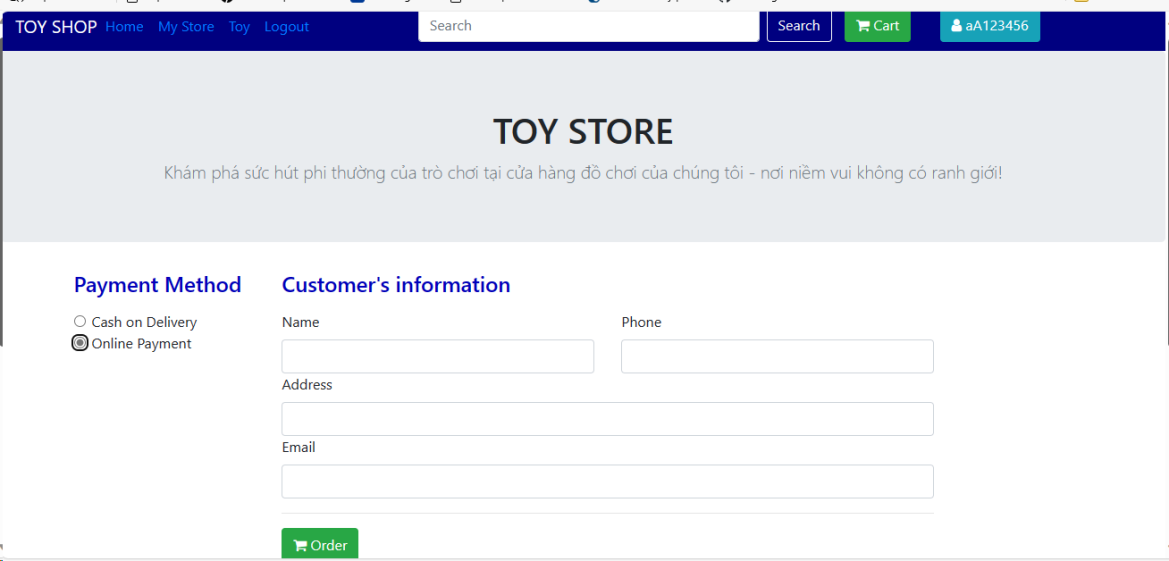
*Hình 14. Giao diện thông tin cá nhân (User Info)*

### **3.3.7 Trang giỏ hàng (Cart)**

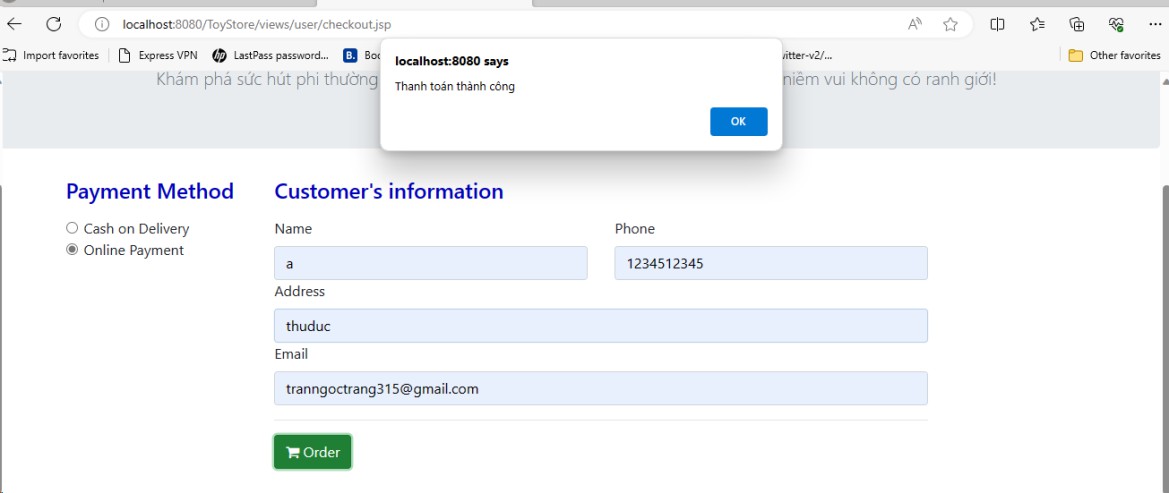


*Hình 15. Giao diện giỏ hàng (Cart)*

### **3.3.8 Trang thanh toán (Order)**

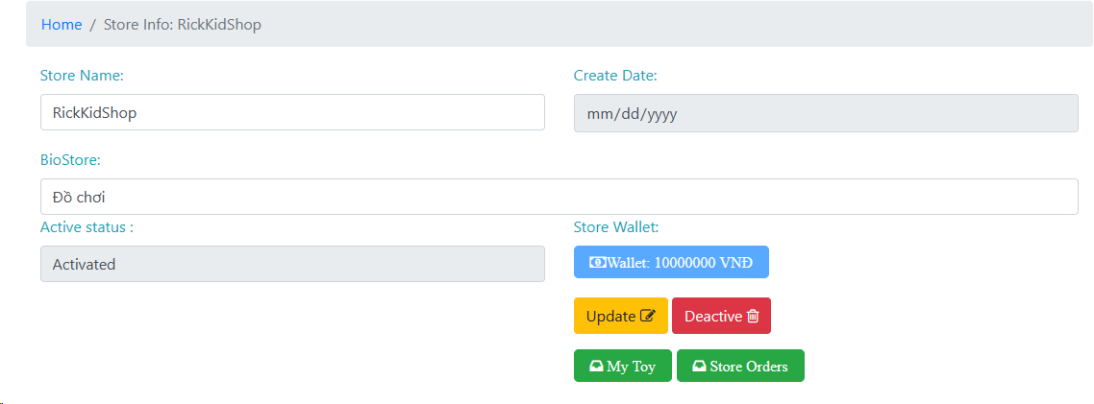


*Hình 16. Giao diện thanh toán (Order)*



*Hình 17. Giao diện thanh toán (Order) thành công*

### **3.3.9 Trang quản lý cửa hàng (My Store)**



*Hình 18. Giao diện quản lý cửa hàng (My Store)*

### **3.3.10 Trang quản lý đơn hàng (Orders Management)**

A screenshot of a computer

Description automatically generated

*Hình 19. Giao diện quản lý đơn hàng (Orders Managemnet)*

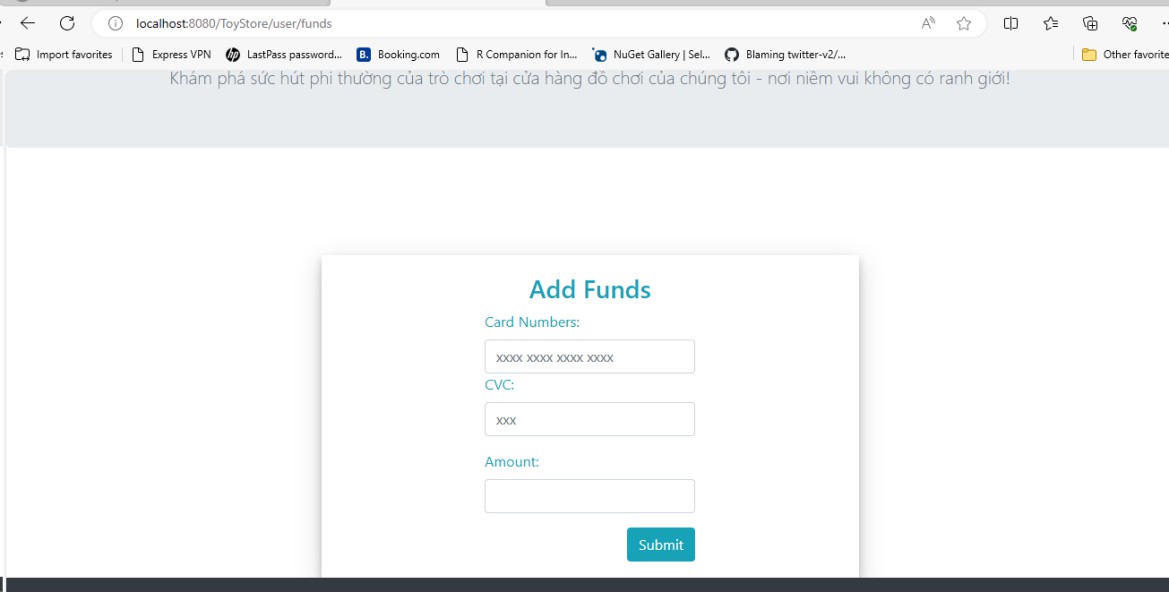
### **3.3.11 Trang xem chi tiết đơn hàng (Details Order)**

A screenshot of a computer

Description automatically generated

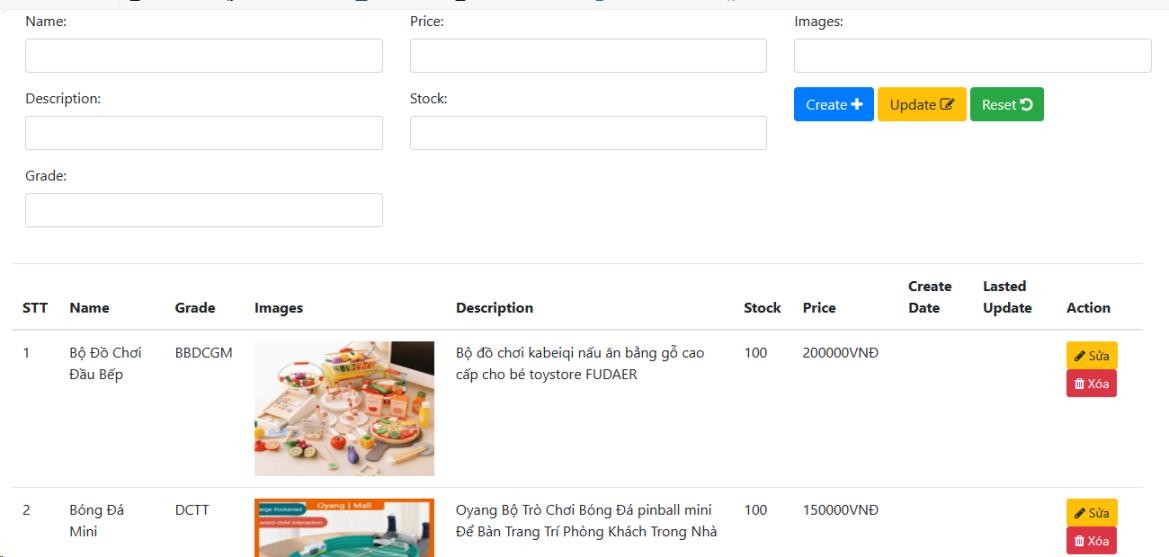
*Hình 20. Giao diện xem chi tiết đơn hàng (Details Order)*

### **3.3.12 Trang nạp tiền vào ví (Add Funds)**



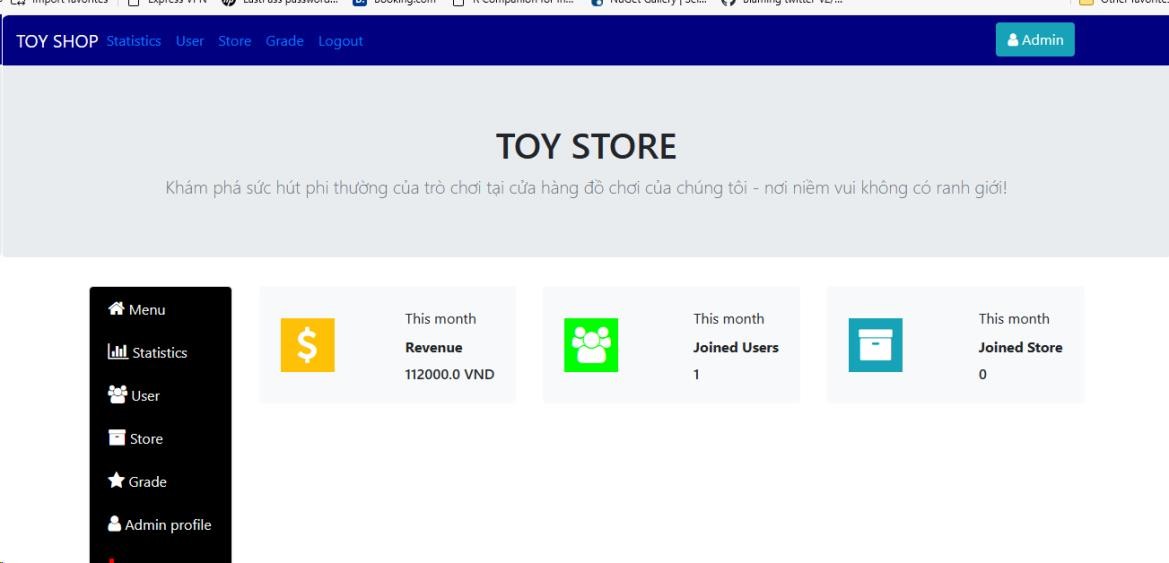
*Hình 21. Giao diện nạp tiền vào ví*

### **3.3.13 Trang quản lý sản phẩm (My Toy)**



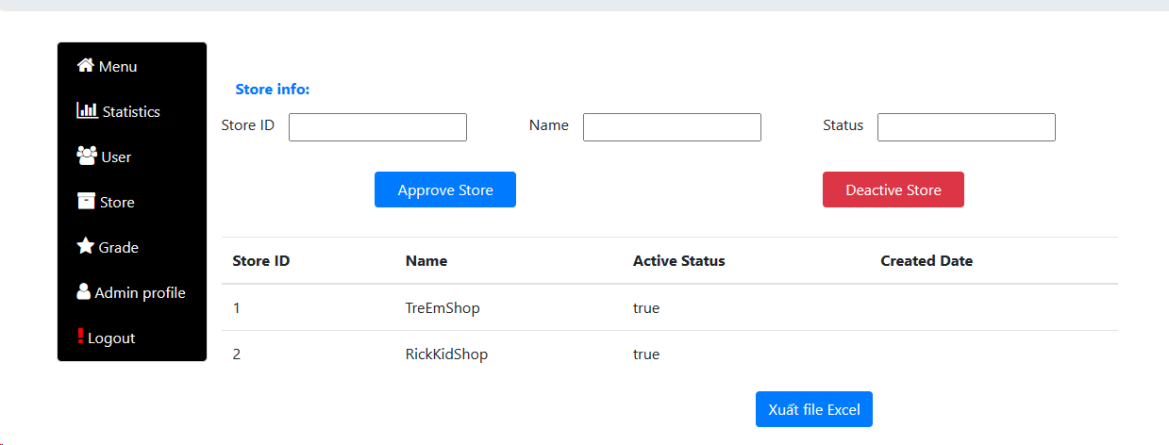
*Hình 22. Giao diện quản lý sản phẩm (My Toy)*

### **3.3.14 Trang thống kê (Statistic)**



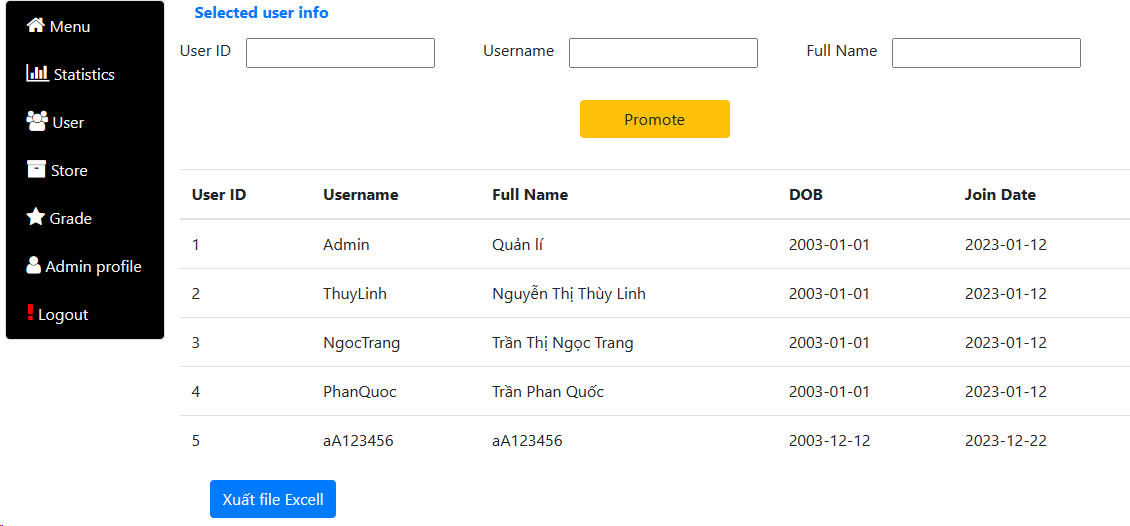
*Hình 23. Giao diện thống kê (Statistic)*

### **3.3.15 Trang quản lý cửa hàng (Stores Management)**



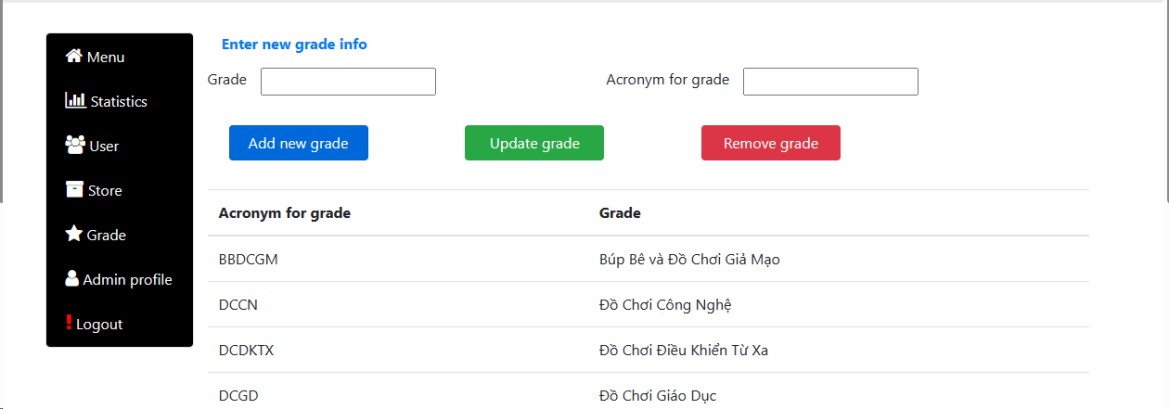
*Hình 24. Giao diện quản lý cửa hàng (Stores Management)*

### **3.3.16 Trang quản lý tài khoản (Users Management)**



*Hình 25. Giao diện quản lý tài khoản (Users Management)*

### **3.3.17 Trang quản lý loại sản phẩm (Grade Management)**



*Hình 26. Giao diện lý loại sản phẩm (Grade Management)*

### **3.3.18 Trang giới thiệu**



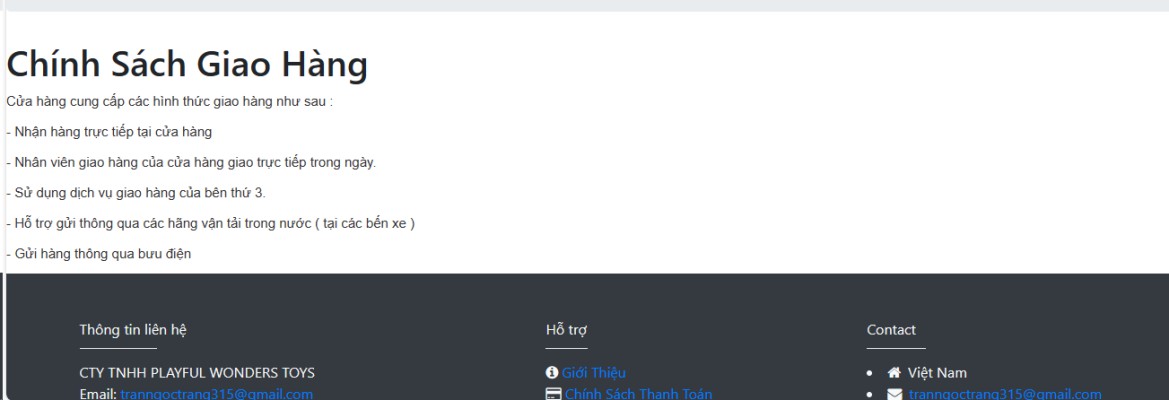
*Hình 27. Giao diện giới thiệu*

### **3.3.19 Trang chính sách thanh toán**



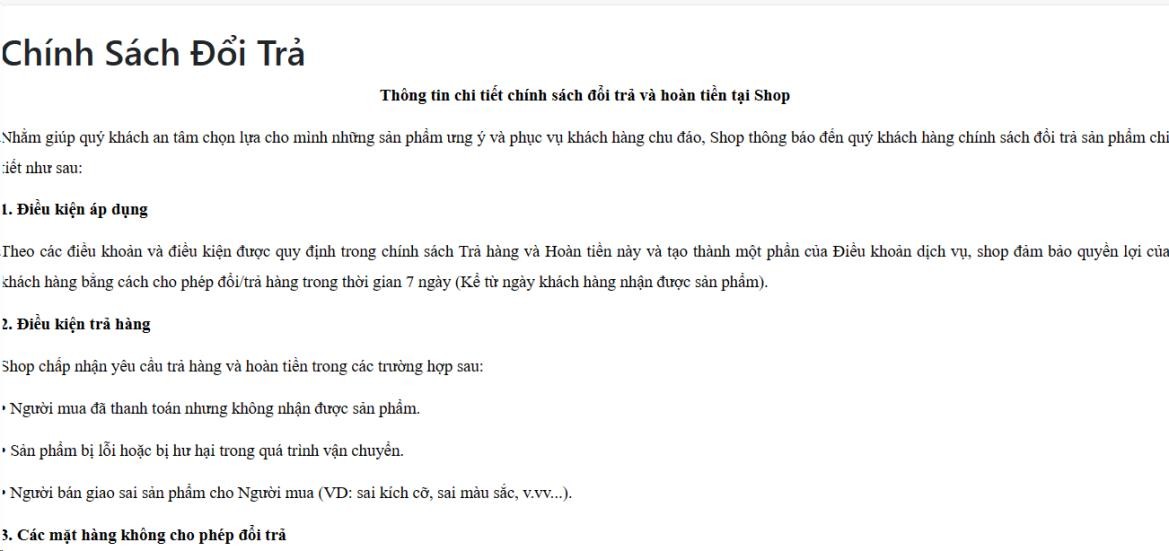
*Hình 28. Giao diện chính sách thanh toán*

### **3.3.20 Trang chính sách giao hàng**



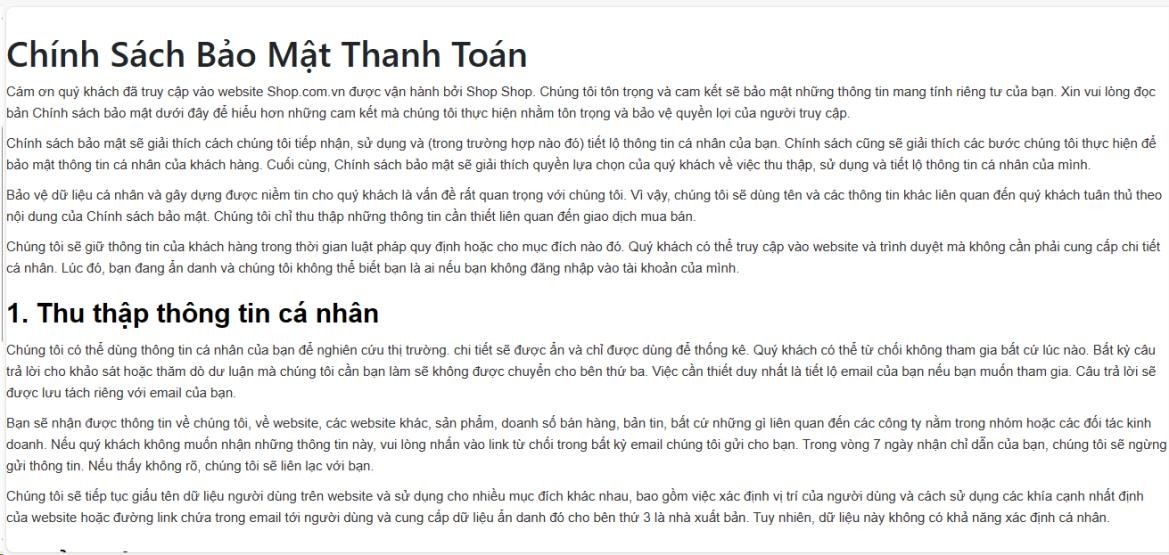
*Hình 29. Giao diện chính sách giao hàng*

### **3.3.21 Trang chính sách đổi trả**



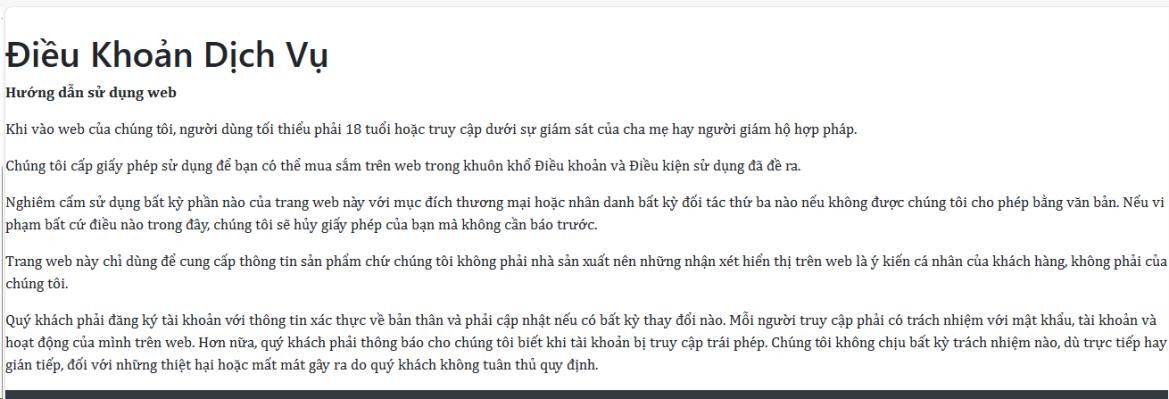
*Hình 30. Giao diện chính sách đổi trả*

### **3.3.22 Trang chính sách bảo mật**



*Hình 31. Giao diện chính sách bảo mật*

### **3.3.23 Trang điều khoản dịch vụ**



*Hình 32. Giao diện điều khoản dịch vụ*

# CHƯƠNG 4. KIỂM THỬ VÀ HƯỚNG PHÁT TRIỂN

## 4.1 KIỂM THỬ

* + - Sử dụng phần mềm Ngrok ( là công cụ tạo đường hầm (tunnel) giữa localhost của bạn và internet. Giúp người khác mạng có thể truy cập được localhost của bạn thông qua custom domain của ngrok.)

Link truy cập project : *https://eft-moral-snail.ngrok-free.app/ToyStore*

* + - Kiểm thử:

|  |  |  |  |
| --- | --- | --- | --- |
| **Mô tả** | **Các bước thực hiện** | **Kết quả mong muốn** | **Kết quả test** |
| Kiểm tra giao diện | | | |
| Kiểm tra tổng thể giao diện màn hình | Kiểm tra bố cục, phông chữ, chính tả, màu chữ | Các label, textbox, button,... có độ dài rộng thích hợp, không xô lệch. Form bố trí hợp lý và dễ sử dụng | Đạt |
| Kiểm tra thứ tự di chuyển trỏ trên màn hình | Nhấn Tab liên tục | Con trỏ di chuyển lần lượt theo thứ tự | Đạt |

|  |  |  |  |
| --- | --- | --- | --- |
| Kiểm tra thực hiện chức năng chính của màn hình | Nhấn phím Enter | Nếu chuột không focus vào nút thì thực hiện chức năng chính.  Nếu đang focus vào button thì sẽ thực hiện chức năng của button đó. | Đạt |
| Kiểm tra chọn một giá trị trong danh sách | Chọn một giá trị trong danh sách. Kiểm tra dữ liệu hiện trên Form | Hiển thị giá trị được chọn trên Form | Đạt |
| Kiểm tra chức năng của hệ thống | | | |
| Kiểm tra chức năng đăng nhập | Trên giao diện:  - Nhập thông tin tài khoản  -Nhấn nút đăng nhập | Đăng nhập thành công | Đạt |
| Kiểm tra chức năng đăng ký | Trên giao diện:   * Nhập thông tin đăng ký tài khoản * Nhấn nút đăng ký | Đăng ký thành công và vào bên trong | Đạt |

|  |  |  |  |
| --- | --- | --- | --- |
| Kiểm tra chức năng quản lý (Tài khoản, Cửa hàng, Loại sản phẩm) | Trên giao diện:  - Nhấn cập nhật thêm, xóa, sửa | Cập nhật thành công  Dữ liệu sau khi cập nhật hiển thị | Đạt |
| Kiểm tra chức năng thống kê | Trên giao diện: hiện đầy đủ mục thống kê | Hiện số liệu thống kê doanh thu, số cửa hàng và số tài khoản người dùng | Đạt |

## 4.2 HƯỚNG PHÁT TRIỂN

Qua quá trình thực hiện dự án, chúng em đã nhận ra dự án này không chỉ là một bài học sâu sắc về môn học mà còn là một cơ hội để chúng em áp dụng lý thuyết vào thực tế. Chúng em đã cố gắng bao quát toàn bộ các chủ đề mà chúng em đã được học trong môn học vào dự án này. Dự án này đã giúp chúng em nắm bắt được quy trình phát triển phần mềm và cách làm việc nhóm hiệu quả.Trong quá trình thực hiện bài báo cáo khó tránh khỏi việc mắc sai sót, đó là những hạn chế mà chúng em sẽ tiếp tục học hỏi, thực hành và hoàn thiện trong tương lai.

Về những điểm cần cải tiến trong tương lai:

* Thiết kế lại giao diện đẹp mắt và tối ưu hơn
* Phát triển giao diện phản hồi để người dùng có thể truy cập từ mọi thiết bị
* Hệ thống tìm kiếm thông minh sẽ giúp người dùng nhanh chóng tìm thấy sản phẩm theo sở thích cá nhân
* Đề xuất nội dung mới dựa trên lịch sử sử dụng
* Nâng cao hiệu suất xử lí
* Tăng cường khả năng bảo mật

# TÀI LIỆU THAM KHẢO

1. Kiến thức từ môn “Lập trình Web”
2. Các kiến thức từ môn học “Hệ quản trị cơ sở dữ liệu” và “Cơ sở dữ liệu”.
3. Các kiến thức từ môn học “Công nghệ phần mềm”
4. Thầy Nguyễn Hữu Trung. Lập trình Java Web với Spring Boot, Truy câp từ: [https://www.youtube.com/playlist?list=PL0yFty\_aTOFO\_zvII92KV36Rw22vRqYn](https://www.youtube.com/playlist?list=PL0yFty_aTOFO_zvII92KV36Rw22vRqYnV) [V](https://www.youtube.com/playlist?list=PL0yFty_aTOFO_zvII92KV36Rw22vRqYnV)
5. Giáo Trình của Cô Mai Anh Thơ , môn ‘Lập trinh Web’
6. Ngrok. "Secure Introspectable Tunnels to localhost." Truy cập từ: <https://ngrok.com/>
7. Joel Murach và Michael Urban, Murachs Java Servlets and JSP 3rd Edition (2014)