

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**  
**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**



**MÔN HỌC: ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO ĐỀ TÀI**  
**XÂY DỰNG GIAO DIỆN WEBSITE BÁN SÁCH**

**GVHD: TS. Trần Nhật Quang**

**SVTH**

**MSSV**

**Nguyễn Vinh Khang**

**20110131**

**Lưu Đặc Vũ**

**20110044**

**Tp. Hồ Chí Minh, tháng 12 năm 2022**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**  
**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**



**MÔN HỌC: ĐỒ ÁN CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO ĐỀ TÀI**  
**XÂY DỰNG GIAO DIỆN WEBSITE BÁN SÁCH**

**GVHD: TS. Trần Nhật Quang**

**SVTH**

**MSSV**

**Nguyễn Vinh Khang**

**20110131**

**Lưu Đặc Vũ**

**20110044**

**Tp. Hồ Chí Minh, tháng 12 năm 2022**

## MỤC LỤC

<b>Danh mục hình ảnh .....</b>	<b>4</b>
<b>Danh mục các bảng .....</b>	<b>5</b>
<b>NỘI DUNG.....</b>	<b>7</b>
<b>I. Đặc tả.....</b>	<b>7</b>
<b>II. Phân công công việc .....</b>	<b>7</b>
<b>III. Thiết kế .....</b>	<b>7</b>
1. Thuật toán .....	7
2. Thiết kế lớp.....	9
3. Thiết kế cơ sở dữ liệu.....	19
4. Thiết kế giao diện .....	21
<b>IV. Cài đặt và kiểm thử.....</b>	<b>24</b>
<b>V. Kết luận.....</b>	<b>25</b>

## **Danh mục hình ảnh**

Hình 1. Mô hình MVC .....	7
Hình 2. Diagram mô tả các bảng.....	20

## Danh mục các bảng

Bảng 1. Bảng phân công công việc .....	7
Bảng 2. Bảng mô tả các lớp A.....	9
Bảng 3. Bảng mô tả các lớp B.....	10
Bảng 4. Bảng mô tả các phương thức của lớp app.js .....	11
Bảng 5. Bảng mô tả các phương thức của lớp appController.js.....	12
Bảng 6. Bảng mô tả các phương thức của lớp product.ejs .....	16
Bảng 7. Bảng mô tả các phương thức của lớp style.js .....	16
Bảng 8. Bảng mô tả các Table trong cơ sở dữ liệu. ....	19
Bảng 9. Bảng mô tả các các Fields trong mySessions .....	19
Bảng 10. Bảng mô tả các các Fields trong users .....	20
Bảng 11. Bảng mô tả các các Fields trong Books .....	20
Bảng 12. Thiết kế giao diện.....	21
Bảng 13. Cài đặt kiểm thử.....	27

## NỘI DUNG

### I. Đặc tả

Sản phẩm của đề án là hệ thống website bán sách. Website được sử dụng để trưng bày sản phẩm và thực hiện giao dịch với khách hàng.

Công cụ xây dựng website: website được viết trên Visual Studio Code, sử dụng html, css, javascript, nodejs, express và mongodb.

Dữ liệu, thông tin đầu vào là các yêu cầu, thao tác của người dùng hệ thống website.

Giao diện của website bao gồm giao diện chính trưng bày sản phẩm và lọc sản phẩm theo nhà xuất bản, trang chi tiết mô tả sản phẩm, trang giỏ hàng để lưu các sản phẩm được thêm vào và để thực hiện giao dịch thì khách hàng sẽ phải có một tài khoản, nếu chưa có thì có thể đăng ký và nó sẽ được lưu vào database để quản lý.

### II. Phân công công việc

Bảng 1. Bảng phân công công việc

TT	Tên sinh viên	Mô tả công việc	Đóng góp
1	Nguyễn Vinh Khang	Xây dựng giao diện chính, trang sản phẩm và đổ dữ liệu lên trang web	100%
2	Lưu Đặc Vũ	Xây dựng database, trang đăng ký/đăng nhập và giỏ hàng	100%

### III. Thiết kế

#### 1. Thuật toán

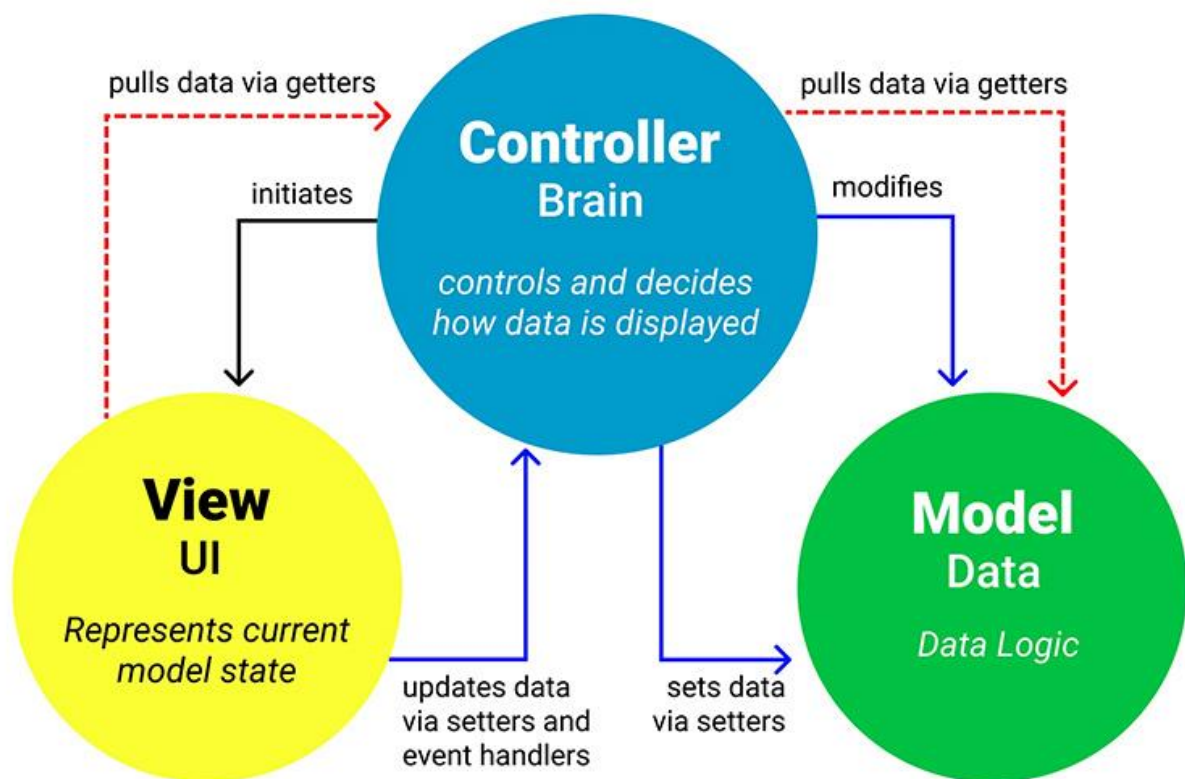
Nhóm sử dụng mô hình MVC( Model – View – Controller). MVC là mô hình thiết kế sử dụng trong kỹ thuật phần mềm. Mô hình source code thành 3 phần, tương ứng mỗi từ. Mỗi từ tương ứng với một hoạt động tách biệt trong một mô hình:

- Controller nhận yêu cầu từ người dùng và sử dụng Model và View để xử lý và

trả kết quả cho người dùng.

- View là giao diện của hệ thống tương tác trực tiếp với người dùng.
- Model là lớp chứa thông tin đối tượng (Dữ liệu), tương tác với Database. Chịu trách nhiệm chính trong mô hình hóa đối tượng.

## MVC Architecture Pattern



Hình 1. Mô hình MVC

**Model (M):**

Là bộ phận có chức năng lưu trữ toàn bộ dữ liệu của ứng dụng. Bộ phận này là một cầu nối giữa 2 thành phần bên dưới là View và Controller. Model thể hiện dưới hình thức là một cơ sở dữ liệu hoặc có khi chỉ đơn giản là một file XML bình thường. Model thể hiện rõ các thao tác với cơ sở dữ liệu như cho phép xem, truy xuất, xử lý dữ liệu,...

**View (V):**

Đây là phần giao diện (theme) dành cho người sử dụng. Nơi mà người dùng có thể lấy được thông tin dữ liệu của MVC thông qua các thao tác truy vấn như tìm kiếm hoặc sử dụng thông qua các website.

Thông thường, các ứng dụng web sử dụng MVC View như một phần của hệ thống, nơi các thành phần HTML được tạo ra. Bên cạnh đó, View cũng có chức năng ghi nhận hoạt động của người dùng để tương tác với Controller. Tuy nhiên, View không có mối quan hệ trực tiếp với Controller, cũng không được lấy dữ liệu từ Controller mà chỉ hiển thị yêu cầu chuyển cho Controller mà thôi.

**Controller (C):**

Bộ phận có nhiệm vụ xử lý các yêu cầu người dùng đưa đến thông qua view. Từ đó đưa ra dữ liệu phù hợp với người dùng. Bên cạnh đó, Controller còn có chức năng kết nối với model.



## 2. Thiết kế lớp

A. Sinh viên phụ trách: Nguyễn Vinh Khang

Bảng 2. Bảng mô tả các lớp do sinh viên thực hiện

TT	Tên lớp	Mục đích
1	app.js	Thực thi chạy chương trình và server
2	appController.js	Tập hợp xử lý các yêu cầu của người dùng và kết nối với model
3	header.ejs	Giao diện của header được dùng trong trang web
4	style.css	Áp dụng thiết kế vào giao diện trang web
5	style.js	Thư viện các hàm javascript được dùng trong trang giỏ hàng và đơn hàng
6	collection.ejs	Giao diện sách được trưng bày cùng với công cụ lọc sách theo nhà xuất bản
7	landing.ejs	Giao diện chính của trang web được hiển thị khi khách hàng chưa đăng nhập
8	dashboard.ejs	Giao diện chính của trang web được hiển thị khi khách hàng đã đăng nhập

9	product.ejs	Giao diện chi tiết sản phẩm được hiển thị khi khách hàng chọn vào một cuốn sách bất kỳ
10	master.ejs	View engine chịu trách nhiệm tạo html cho giao diện người dùng
11	Book.js	Model book để làm việc với cơ sở dữ liệu trong việc load thông tin sách lên trang

B. Sinh viên phụ trách: Lưu Đặc Vũ

Bảng 3. Bảng mô tả các lớp do sinh viên

TT	Tên lớp	Mục đích
1	db.js	Khởi tạo kết nối và kiểm tra kết nối đến cơ sở dữ liệu MongoDB
2	is-auth.js	Kiểm tra việc đăng nhập của người dùng có hợp lệ hay không
3	User.js	Model user để làm việc với cơ sở dữ liệu trong đăng ký và đăng nhập
4	style.css	Áp dụng thiết kế vào giao diện trang web
5	style.js	Thư viện các hàm javascript được dùng trong trang giỏ hàng và đơn hàng
6	footer.ejs	Giao diện của footer được dùng trong trang web
7	cart.ejs	Giao diện giỏ hàng và thanh toán được sử dụng trong trang web
8	order.ejs	Giao diện đơn hàng khi khách hàng đặt hàng thành công

9	login.ejs	Giao diện đăng nhập xuất hiện khi khách hàng muốn thực hiện giao dịch
10	register.ejs	Giao diện đăng ký tài khoản nếu khách hàng chưa có tài khoản

**Chi tiết các lớp được sử dụng trong đề tài.**

**Sinh viên phụ trách: Nguyễn Vinh Khang**

Bảng 4. Bảng mô tả các phương thức của lớp app.js

TT	Phương thức	Mục đích	Tên file, số thứ tự dòng chứa khai báo
1	<b>app.set("view engine", "ejs")</b> Input: Không có Output: Không có	Thiết lập view engine theo ejs	app.js(22)
2	<b>app.set('layout', 'master')</b> Input: Không có Output: Không có	Xác định master.ejs là layout của view engine	app.js(24)
3	<b>app.use(express.static(__dirname + '/public'))</b> Input: Không có Output: Không có	Sử dụng đường path dẫn tới public	app.js(37)
4	<b>app.get("/", appController.landing page)</b> Input: Không có Output: Không có	Nhận đường dẫn tới trang đầu tiên	app.js(41)
5	<b>app.get("/login", appController.login_get)</b> Input: Không có Output: Không có	Nhận đường dẫn tới trang đăng nhập	app.js(44)
6	<b>app.post("/login", appController.login_post)</b> Input: Không có Output: Không có	Nhận đường dẫn tới trang đăng nhập	app.js(45)

7	<b>app.get("/register", appController.register_get)</b> Input: Không có Output: Không có	Chuyển tới trang đăng kí	app.js(48)
8	<b>app.post("/register", appController.register_post)</b> Input: Không có Output: Không có	Lấy thông tin đăng kí thêm vào mongodb	app.js(49)
9	<b>app.get("/dashboard", isAuth, appController.dashboard_get)</b> Input: Không có Output: Không có	Chuyển tới trang người dùng đăng nhập	app.js(52)
10	<b>app.get("/collections", appController.collections_get)</b> Input: Không có Output: Không có	Chuyển tới trang danh mục sách	app.js(56)
11	<b>app.get("/collections/khoagiao", appController.collections_getKG)</b> Input: Không có Output: Không có	Chuyển tới trang sách thể loại khoa giáo	app.js(57)
12	<b>app.get("/collections/truyen", appController.collections_getTT)</b> Input: Không có Output: Không có	Chuyển tới trang sách thể loại truyện tranh	app.js(58)
13	<b>app.get("/collections/vanhoc", appController.collections_getVH)</b> Input: Không có Output: Không có	Chuyển tới trang sách thể loại văn học	app.js(59)
14	<b>app.get("/collections/kinhte", appController.collections_getKT)</b> Input: Không có Output: Không có	Chuyển tới trang sách thể loại kinh tế	app.js(60)
15	<b>app.get("/collections/nxbhn", appController.collections_gethn)</b> Input: Không có Output: Không có	Chuyển tới trang sách của nhà xuất bản Hà Nội	app.js(61)
16	<b>app.get("/collections/nxbkd", appController.collections_getkd)</b> Input: Không có Output: Không có	Chuyển tới trang sách của nhà xuất bản Kim Đồng	app.js(62)

17	<b>app.get("/collections/nxbtn", appController.collections_gettn)</b> Input: Không có Output: Không có	Chuyển tới trang sách của nhà xuất bản Thanh Niên	app.js(63)
18	<b>app.get("/collections/nxbtr", appController.collections_gettr)</b> Input: Không có Output: Không có	Chuyển tới trang sách của nhà xuất bản Trẻ	app.js(64)
19	<b>app.get("/collections/product", appController.product_get)</b> Input: Không có Output: Không có	Chuyển tới trang thông tin chi tiết sách	app.js(67)
20	<b>app.get("/cart", appController.cart_get)</b> Input: Không có Output: Không có	Chuyển tới trang thông tin giỏ hàng	app.js(70)
21	<b>app.get("/order", appController.order_get)</b> Input: Không có Output: Không có	Chuyển tới trang đơn hàng	app.js(73)
22	<b>app.post("/logout", appController.logout_post)</b> Input: Không có Output: Không có	Đăng xuất khỏi tài khoản hiện tại	app.js(75)
23	<b>app.listen(3000, console.log("App Running on <a href="http://localhost:3000">http://localhost:3000</a>"))</b> Input: Không có Output: Không có	Chạy server local host	app.js(77)

Bảng 5. Bảng mô tả các phương thức của lớp appController.js

TT	Phương thức	Mục đích	Tên file, số thứ tự dòng chứa khai báo
1	<b>exports.landing_page = (req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện chính của web (chưa đăng nhập)	appController.js(7)
2	<b>exports.login_get = (req, res) =&gt; {}</b> Input: Không có Output: Không có	Tạo và reset biến error trên session rồi load trang login	appController.js(16)

3	<b>exports.login_post = async (req, res) =&gt; {}</b> Input: Email, mật khẩu Output: Đăng nhập thành công hoặc không hợp lệ Pseudo code: Nếu user chưa tồn tại thì phải tạo tài khoản Nếu user tồn tại nhưng mật khẩu không đúng thì được yêu cầu nhập lại mật khẩu Nếu tất cả đều hợp lệ thì sẽ được phép truy cập vào trang web	Xử lý quá trình đăng nhập của khách hàng	appController.js(22)
4	<b>exports.register_get = (req, res) =&gt; {}</b> Input: Không có Output: Không có	Tạo và reset biến error trên session rồi load trang register	appController.js(43)
5	<b>exports.register_post = async (req, res) =&gt; {}</b> Input: tên tài khoản, email, mật khẩu Output: Tạo tài khoản thành công hoặc không hợp lệ Pseudo code: Nếu user đã tồn tại thì thông báo đã tồn tại Nếu user chưa tồn tại thì hash mật khẩu và đưa vào database rồi chuyển sang trang login	Xử lý quá trình đăng ký tài khoản của khách hàng	appController.js(49)
6	<b>exports.dashboard_get = (req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện chính của web (đã đăng nhập)	appController.js(71)
7	<b>exports.logout_post = async (req, res) =&gt; {}</b> Input: Không có Output: Không có Pseudo code: Xóa session đăng nhập của user và chuyển hướng về	Xử lý quá trình đăng xuất tài khoản của khách hàng	appController.js(82)

	trang đăng nhập		
8	<b>exports.collections_get=</b> <b>(req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện trưng bày sản phẩm có công cụ lọc	appController.js(90)
9	<b>exports.collections_getKG</b> <b>= (req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện trưng bày sản phẩm lọc theo Khoa giáo	appController.js(98)
10	<b>exports.collections_getVH</b> <b>= (req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện trưng bày sản phẩm lọc theo Văn học	appController.js(106)
11	<b>exports.collections_getTT</b> <b>= (req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện trưng bày sản phẩm lọc theo Truyện tranh	appController.js(114)
12	<b>exports.collections_getKT</b> <b>= (req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện trưng bày sản phẩm lọc theo Kinh tế	appController.js(122)
13	<b>exports.collections_getkd=</b> <b>(req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện trưng bày sản phẩm lọc theo NXB Kim Đồng	appController.js(130)
14	<b>exports.collections_gettr=</b> <b>(req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện trưng bày sản phẩm lọc theo NXB Trẻ	appController.js(138)
15	<b>exports.collections_gethn=</b> <b>(req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện trưng bày sản phẩm lọc theo NXB Hà Nội	appController.js(146)
16	<b>exports.collections_gettn=</b> <b>(req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện trưng bày sản phẩm lọc theo NXB Thanh niên	appController.js(154)
17	<b>exports.product_get=</b> <b>(req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giao diện chi tiết sản phẩm	appController.js(165)

18	<b>exports.cart_get= (req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang giỏ hàng	appController.js(180)
19	<b>exports.order_get= (req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang đơn hàng sau khi khách hàng thực hiện thanh toán	appController.js(185)
20	<b>exports.order_post = async (req, res) =&gt; {}</b> Input: Không có Output: Không có	Load trang đơn hàng sau khi khách hàng thực hiện thanh toán	appController.js(188)

Bảng 6. Bảng mô tả các phương thức của lớp product.ejs

TT	Phương thức	Mục đích	Tên file, số thứ tự dòng chứa khai báo
1	<b>handlePlus = function () {}</b>	Xử lý giá trị tăng của số lượng sản phẩm	MenuController.cs (165)
2	<b>handleSub = function () {}</b>	Xử lý giá trị giảm của số lượng sản phẩm	MenuController.cs (170)

**Sinh viên phụ trách: Lưu Đặc Vũ**

Bảng 7. Bảng mô tả các phương thức của lớp style.js

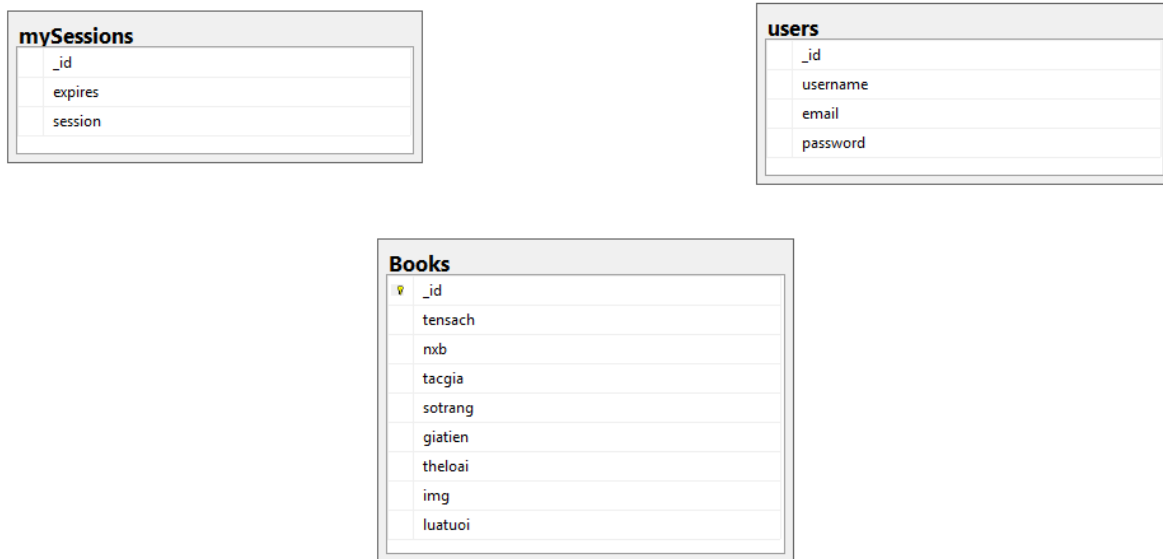
TT	Phương thức	Mục đích	Tên file, số thứ tự dòng chứa khai báo
1	<b>document.getElementById('showcart').style.display = 'none'</b> Input: thẻ có id “showcart” Output: Không có	Ẩn giỏ hàng khi vừa vào trang sản phẩm	style.js (2)
	<b>function themvaogiohang()</b> Input: Không có Output: Không có Pseudo code:	Xử lý đưa dữ liệu của sách	style.js(6)



2	Kiểm tra trong giỏ hàng đã có sản phẩm hay chưa Nếu đã có thì tăng số lượng Nếu chưa có thì thêm mới Lưu vào sessionStorage	vào giỏ hàng	
3	<b>function showcountsp()</b> Input: Không có Output: Không có	Xử lý hiển thị số lượng sản phẩm trong mục giỏ hàng	style.js(35)
4	<b>function showcart()</b> Input: Không có Output: Không có Pseudo code: Nếu đang hiện khi bấm vào sẽ ẩn và ngược lại	Xử lý hiển thị giỏ hàng trong trang sản phẩm	style.js(42)
5	<b>function showmycart()</b> Input: Không có Output: Không có Pseudo code: For (từng sản phẩm trong mảng giỏ hàng) Thực hiện in ra thông tin của từng sản phẩm	Xử lý hiển thị dữ liệu trong giỏ hàng trong trang sản phẩm	style.js(52)
6	<b>function xoasp(x)</b> Input: sản phẩm cần xóa Output: Không có Pseudo code: Xóa thẻ thông tin chứa sản phẩm Xóa sản phẩm trong mảng Xóa thông tin sản phẩm được lưu trên session	Xử lý xóa dữ liệu của một sản phẩm ra khỏi giỏ hàng	style.js(83)
7	<b>function thanhtoan()</b> Input: Không có Output: Không có Pseudo code: Tạo mảng thông tin người nhận lấy dữ liệu từ các trường dữ liệu trên trang web	Xử lý thông tin của khách hàng khi thực hiện thanh toán sản phẩm	style.js(105)

	Lưu mảng vào session Chuyển sang trang đơn hàng		
8	<b>function showthongtinnguoinhan()</b> Input: Không có Output: Không có Pseudo code: Lấy thông tin khách hàng từ session Thực hiện đưa thông tin của khách hàng lên trang đơn hàng	Hiện thị thông tin của khách hàng lên trang đơn hàng được lấy từ session storage	style.js(119)
9	<b>function showmycartDH()</b> Input: Không có Output: Không có Pseudo code: For (từng sản phẩm trong mảng giỏ hàng) Thực hiện in ra thông tin của từng sản phẩm	Xử lý hiện thị dữ liệu các sản phẩm trong trang đơn hàng	style.js(142)

### 3. Thiết kế cơ sở dữ liệu



Hình 2. Diagram mô tả các bảng

Bảng 8. Bảng mô tả các Table trong cơ sở dữ liệu mongodb.

TT	Tên bảng	Mục đích
1	mySessions	Lưu trữ session của khách hàng đăng nhập vào web
2	users	Lưu trữ thông tin tài khoản của khách hàng
3	Books	Lưu trữ dữ liệu của sách

#### Bảng mô tả các Fields trong 1 Table

Bảng 9. Bảng mô tả các các Fields trong mySessions.

TT	Tên trường	Kiểu dữ liệu	Mục đích
1	id	objectid	Là id riêng được MongoDB cấp cho từng session
2	expires	date	Thời điểm hết hạn của session
3	session	object	Chứa các thông tin của session

Bảng 10. Bảng mô tả các các Fields trong users.

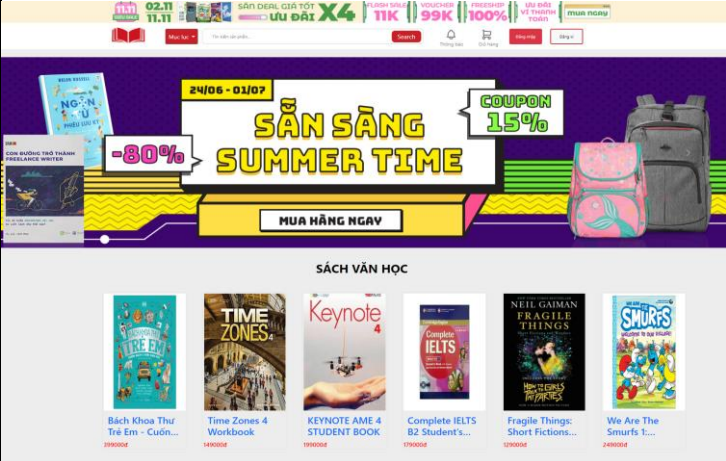
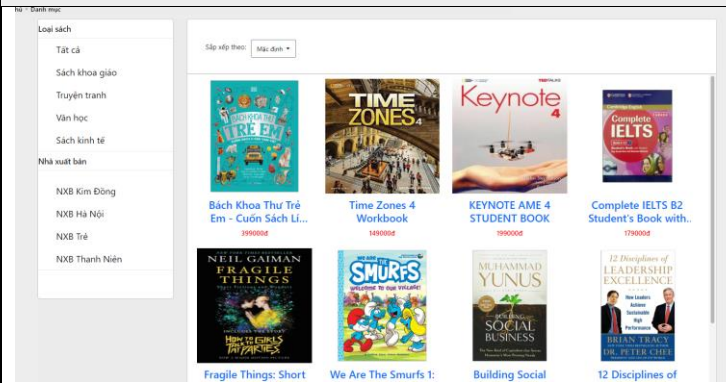
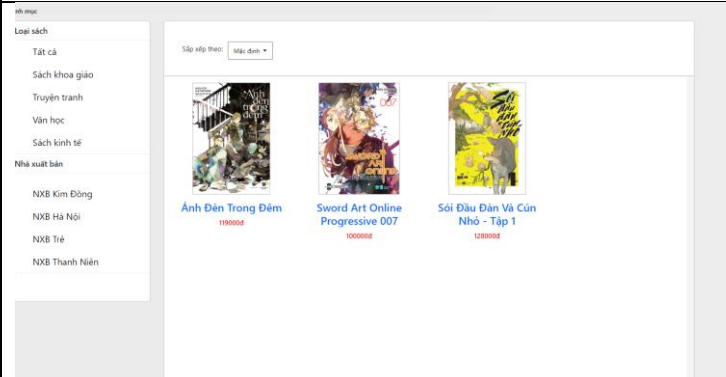
TT	Tên trường	Kiểu dữ liệu	Mục đích
1	id	objectid	Là id riêng được MongoDB cấp cho từng user
2	username	string	Chứa tên tài khoản của khách hàng
3	email	string	Chứa email mà khách hàng dùng để đăng ký tài khoản
3	password	string	Chứa mật khẩu của tài khoản khách hàng đã được hash

Bảng 11. Bảng mô tả các các Fields trong Books.

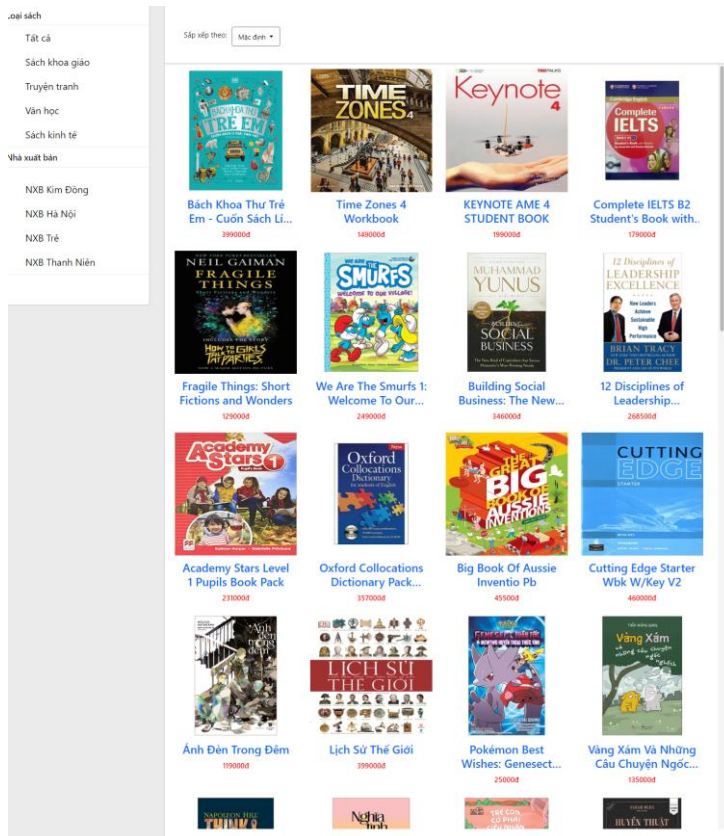
TT	Tên trường	Kiểu dữ liệu	Mục đích
1	id	objectid	Là id riêng được MongoDB cấp cho từng sách
2	tensach	string	Chứa tên của sách
3	nxb	string	Chứa tên nhà xuất bản của sách
4	tacgia	string	Chứa tên tác giả đã viết sách
5	sotrang	int	Chứa số lượng trang của sách
6	giatien	int	Chứa giá tiền của sách
7	theloai	string	Chứa thể loại của sách
8	img	string	Chứa đường dẫn đến hình ảnh của sách
9	luatuoi	string	Chứa loại đối tượng mà sách nhắm đến

#### 4. Thiết kế giao diện

Bảng 12. Thiết kế giao diện

TT	Màn hình/Cửa sổ/Dialog	Mục đích	Giải thích
1	<p>Nhà sách K&amp;V (giao diện chính)</p>  <p>1</p>  	<p>Hiện thị giao diện chính cho khách hàng sử dụng</p>	<p>Giao diện chính được xây dựng với các banner và các danh mục sản phẩm cùng với một số thông tin khuyến mãi và liên hệ giúp người dùng dễ dàng nắm bắt.</p>

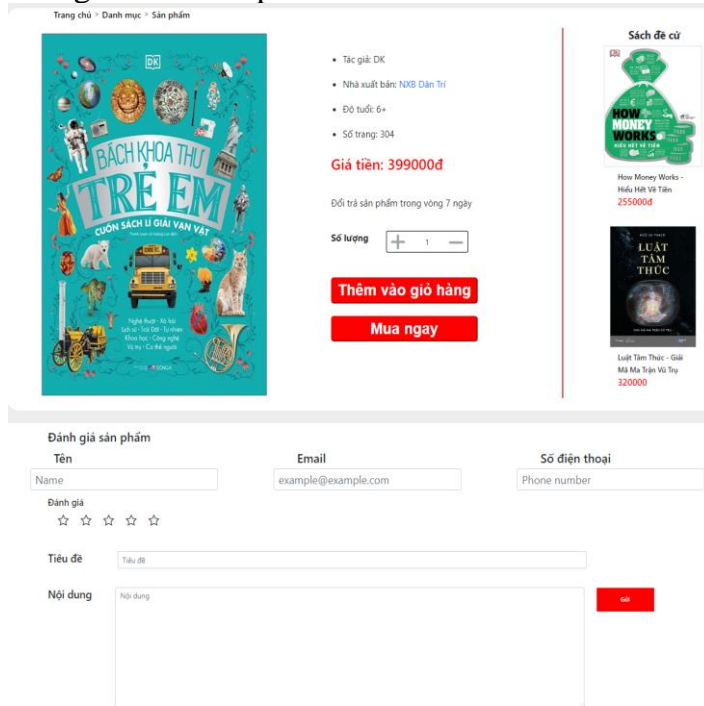
## Trang sản phẩm



Cung cấp thông tin về các sản phẩm cho người dùng.

Cung cấp tên giá và hình ảnh về sản phẩm. Cùng với bộ lọc sản phẩm theo loại giúp dễ dàng tìm kiếm sản phẩm.

## Trang chi tiết sản phẩm



Hiện thị thông tin chi tiết sản phẩm

Mỗi sản phẩm đều chứa một lượng thông tin nhất định cho khách hàng

4	<p><b>Đăng kí</b></p> 	Tạo tài khoản cho khách hàng	Định danh khách hàng
6	<p><b>Đăng nhập</b></p> 	Dùng để đăng nhập tài khoản của khách hàng	Kiểm soát việc truy cập trang
7	<p><b>Giỏ hàng và thanh toán</b></p> 	Dùng để chứa sản phẩm của người chọn mua và tính ra số tiền khách hàng phải trả	Tính toán số tiền và thông báo cho khách hàng đã mua những gì, hết bao nhiêu

#### IV. Cài đặt và kiểm thử

Bảng 13. Cài đặt kiểm thử

TT	Tình huống	Mục đích	Giải thích
1	<p>Tình huống 1:</p> <p>Dữ liệu vào:</p> <p>Truy cập vào trang giao diện chính của hệ thống, bấm chọn sản phẩm rồi bấm thêm vào giỏ hàng khi chưa đăng nhập</p> <p>Kết quả dự kiến:</p> <p>Được yêu cầu đăng nhập và sẽ được chuyển hướng tới trang đăng nhập</p>	Kiểm thử chức năng đăng nhập của trang web	Mỗi khách hàng khi muốn đặt hàng thì phải có một tài khoản trong hệ thống để xử lý đơn hàng
2	<p>Tình huống 2:</p> <p>Dữ liệu vào:</p> <p>Đăng ký tài khoản trên web và tiến hành đặt hàng như tình huống 1</p> <p>Kết quả dự kiến:</p> <p>Thành công đặt hàng và có thể xem lại đơn hàng tại trang đơn hàng</p>	Kiểm thử chức năng đăng ký tài khoản cũng như đặt hàng của trang web	Khi khách hàng đã có một tài khoản thì có thể tiến hành đặt hàng và sẽ được trả về một đơn hàng khi hoàn tất



## **V. Kết luận**

### **- Kết quả đạt được:**

Về cơ bản nhóm tự nhận xét chương trình tạo ra đã giải quyết được một phần những yêu cầu đề ra. Nhóm đã sử dụng node.js, bootstrap và các kiến thức về html, css, javascript trong chương trình học để xây dựng nên giao diện website hoàn thành được những yêu cầu cơ bản của đề tài.

### **- Định hướng phát triển:**

- + Tối ưu hóa code để thời gian chạy tốt hơn.
- + Thiết kế lại giao diện.
- + Phát triển thêm tính năng cho trang web như bình luận, đánh giá sản phẩm,...
- + Cấu trúc file dễ hiểu, dễ nhìn hơn.
- + Xây dựng hệ thống database rõ ràng hơn để lưu trữ tối ưu các dữ liệu của trang web

## NGUỒN THAM KHẢO

**Bootstrap:** <https://getbootstrap.com/>

**Giao diện:** <https://www.fahasa.com/>

**Ejs :** <https://ejs.co/>