|  |  |
| --- | --- |
|  | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------**  ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC  NGÀNH KHOA HỌC MÁY TÍNH  **ĐỀ TÀI: Xây dựng webste bán giày thể thao cho cửa hàng**  **MinhSport**  **CBHD:** ThS. Phạm Thế Anh  **Sinh viên:** Trần Tuấn Minh  **Mã số sinh viên:** 2019605362  Hà Nội – Năm 2024 |

**MỤC LỤC**

# DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG BIỂU

# LỜI CẢM ƠN

# LỜI NÓI ĐẦU

# CHƯƠNG 1: TỔNG QUAN

## Lý do chọn đề tài

Ngày nay nhu cầu mua sắm sản phẩm online đang phát triển mạnh mẽ. Việc xây dựng 1 website để đáp ứng nhu cầu trên là cần thiết vì:

* Sản phẩm có nhu cầu: Giày thể thao là một sản phẩm phổ biến và có nhu cầu cao trong thị trường hiện nay. Việc xây dựng một website bán giày thể thao có thể giúp đáp ứng nhu cầu của khách hàng và tăng doanh số bán hàng cho doanh nghiệp.
* Thị trường tiềm năng: Thị trường giày thể thao đang phát triển mạnh mẽ và tiềm năng với sự tăng trưởng của ngành thể thao và phong cách thời trang đường phố.
* Cạnh tranh: Mặc dù thị trường có tiềm năng nhưng cũng rất cạnh tranh. Việc xây dựng một website bán giày thể thao giúp doanh nghiệp tạo ra sự khác biệt và thu hút được khách hàng với các ưu đãi và dịch vụ tốt.
* Tiết kiệm chi phí: Việc xây dựng một website bán hàng giày thể thao giúp tiết kiệm chi phí so với việc mở cửa hàng bán lẻ truyền thống. Doanh nghiệp có thể giảm chi phí vận hành, quảng cáo, thuê mặt bằng và chi phí nhân sự….

Vì những lý do này, việc xây dựng website bán giày thể thao là một lựa chọn hợp lý cho doanh nghiệp hoạt động trong ngành này.

## Đối tượng, mục tiêu và phạm vi nghiên cứu

### Đối tượng nghiên cứu

- Ngôn ngữ: JavaScript, ReactJS Framework, NodeJS(Express) .

- Công cụ: MongoDBCompass, Postman, Chrome, Visual Studio Code.

- Cơ sở dữ liệu: MongoDB.

### Mục tiêu nghiên cứu

- Phân tích, hiểu bài toán nghiệp vụ thực tế.

- Tìm hiểu công nghệ xây dựng.

- Xây dựng được trang web đáp ứng đầy đủ các yêu cầu đặt ra.

### Phương pháp nghiên cứu

- Phạm vi không gian: công cụ Visual Studio Code, ngôn ngữ JavaScript.

- Phạm vi thời gian: 9 tuần

## Phương pháp nghiên cứu

- Lý thuyết:

Nghiên cứu và tiến hành thu thập các tài liệu có liên quan đến đề tài.

Tổng hợp các tài liệu đã thu thập và tiến hành phân tích.

Chọn lọc để báo cáo.

- Thực nghiệm:

Phân tích các trang web có chức năng tương tự.

Phân tích, tiến hành xây dựng theo phân tích và yêu cầu thực tế.

Tìm hiểu cách thức, tiêu chí kiểm thử giao diện và chức năng.

# CHƯƠNG 2: CÔNG NGHỆ SỬ DỤNG

## 2.1. Ngôn ngữ JavaScript

* *Giới thiệu chung về JavaScript*

Đây là ngôn ngữ lập trình kịch bản, chạy trên trình duyệt, chuyên dùng để xây dựng phần front-end của trang web, hiện nay JavaScript đã xuất hiện hầu như tại tất cả các trang web đang hoạt động cùng với những Frameworks của nó.

JavaScript cung cấp DOM API (bộ công cụ dùng để tương tác với thẻ HTML trên web). Với DOM API các lập trình viên có thể thực hiện được rất nhiều thao tác để xử lý logic cho người dùng, hay thực hiện các thao tác để thay đổi chỉnh sửa giao diện của website trong quá trình hoạt động.

Về tổng quan, JavaScript cũng có các thành phần giống như PHP đang có (hướng đối tượng, hướng chức năng, biến, mảng, hàm, vòng lặp, …). Để chạy được JavaScript chỉ cần khai báo thẻ script trong trang HTML và viết các đoạn lệnh của JavaScript là có thể chạy được hoặc với những script lớn, ta có thể thực hiện riêng ra một có đuôi “.js” sau đó thực hiện nhúng tệp vào trang HTML bằng đoạn mã lệnh <script src=” đường dẫn tệp cần nhúng ” /> là có thể thực hiện được.

* *Một vài cập nhật mới cho phiên bản ES6*

Mỗi phiên bản cập nhật, JavaScript lại cung cấp cho người dùng thêm nhiều tính năng mới. Ví dụ: tại các phiên bản gần đây JavaScript cung cấp tính năng module, giúp cho việc tái sử dụng code trở nên dễ dàng hơn, hay là nội suy, với tính năng này việc lấy dữ liệu từ các biến trong JS để nhúng vào một chuỗi cũng trở nên dễ dàng hơn (thay vì phải nối chuỗi).

* *JavaScript trong thực tiễn*

Trên thực tế, hiện nay các website hiện đại đang dần được thay thế và phát triển bởi các JavaScript Framework như là: ReactJS, VueJS, AngularJS với mục đích tạo ra các trang web có hiệu suất xử lý nhanh hơn nhờ tương tác với DOM ảo và nguyên lý ứng dụng đơn trang (chỉ cập nhật phần nào thay đổi).

## 2.2. ReactJS Framework

ReactJS là một thư viện JavaScript chuyên giúp các nhà phát triển xây dựng giao diện người dùng hay UI. Trong lập trình ứng dụng front-end, lập trình viên thường sẽ phải làm việc chính trên 2 thành phần sau: UI và xử lý tương tác của người dùng. UI là tập hợp những thành phần mà bạn nhìn thấy được trên bất kỳ một ứng dụng nào, ví dụ có thể kể đến bao gồm: menu, thanh tìm kiếm, những nút nhấn, card,… Giả sử bạn đang lập trình một website thương mại điện tử, sau khi người dùng chọn được sản phẩm ưng ý rồi và nhấn vào nút “Thêm vào giỏ hàng”, thì việc tiếp theo mà bạn phải làm đó là thêm sản phẩm được chọn vào giỏ hàng và hiển thị lại sản phẩm đó khi user vào xem => xử lý tương tác.

Trước khi có ReactJS, lập trình viên thường gặp rất nhiều khó khăn trong việc sử dụng “vanilla JavaScript” (JavaScript thuần) và JQuery để xây dựng UI. Điều đó đồng nghĩa với việc quá trình phát triển ứng dụng sẽ lâu hơn và xuất hiện nhiều bug, rủi ro hơn. Vì vậy vào năm 2011, Jordan Walke – một nhân viên của Facebook đã khởi tạo ReactJS với mục đích chính là cải thiện quá trình phát triển UI.

Hơn nữa, để tăng tốc quá trình phát triển và giảm thiểu những rủi ro có thể xảy ra trong khi coding, React còn cung cấp cho chúng ta khả năng Reusable Code (tái sử dụng code) bằng cách đưa ra 2 khái niệm quan trọng bao gồm:

* JSX.
* Virtual DOM.

## 2.3. Nền tảng NodeJS

NodeJs là một môi trường runtime chạy Javascript đa nền tảng và có mã nguồn mở, được sử dụng để chạy các ứng dụng web bên ngoài trình duyệt của client. Nền tảng này được phát triển bởi Ryan Dahl vào năm 2009, được xem là một giải pháp hoàn hảo cho các ứng dụng sử dụng nhiều dữ liệu nhờ vào mô hình hướng sự kiện (event-driven) không đồng bộ.

## 2.4. Cơ sở dữ liệu

## 2.5. JSON Web Token (JWT)

JSON Web Token (JWT) là một chuẩn mở (RFC 7519) định nghĩa một cách nhỏ gọn và khép kín để truyền một cách an toàn thông tin giữa các bên dưới dạng đối tượng JSON. Thông tin này có thể được xác minh và đáng tin cậy vì nó có chứa chữ ký số. JWTs có thể được ký bằng một thuật toán bí mật (với thuật toán HMAC) hoặc một public / private key sử dụng mã hoá RSA.

Thành phần chính của JWT:



Hình 2. 1 Thành phần của JWT

*Header*

Gồm hai phần chính: loại token (mặc định là JWT - Thông tin này cho biết đây là một Token JWT) và thuật toán đã dùng để mã hóa (HMAC SHA256 - HS256 hoặc RSA).

Ví dụ:

{

    "typ": "JWT",

    "alg": "HS256"

}

*Payload*

Payload sẽ chứa các thông tin đặt trong chuỗi Token như issuer, issued\_at, expiration\_time, subject, audience, …

Ví dụ:

{

    "iss": "jira:1314039",

    "iat": 1300819370,

    "exp": 1300819380,

    "qsh": "8063ff4ca1e41df7bc90c8ab6d0f6207d491cf6dad7c66ea797b4614b71922e9",

    "sub": "batman",

    "context": {

        "user": {

            "userKey": "batman",

            "username": "bwayne",

            "displayName": "Bruce Wayne"

        }

    }

}

**Public Claims** - Claims được cộng đồng công nhận và sử dụng rộng rãi.

**Private Claims** - Claims tự định nghĩa (không được trùng với Reserved Claims và Public Claims), được tạo ra để chia sẻ thông tin giữa 2 parties đã thỏa thuận và thống nhất trước đó.

*Signature*

Chữ ký Signature trong JWT là một chuỗi được mã hóa bởi header, payload cùng với một chuỗi bí mật theo nguyên tắc sau:

HMACSHA256(

  base64UrlEncode(header) + "." +

  base64UrlEncode(payload),

  secret)

Do bản thân Signature đã bao gồm cả header và payload nên Signature có thể dùng để kiểm tra tính toàn vẹn của dữ liệu khi truyền tải.

# CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## 3.2. Phân tích yêu cầu

### 3.2.1. Yêu cầu chức năng

* Đăng nhập
* Đăng ký
* Tìm kiếm
* Xem giỏ hàng
* Đánh giá sản phẩm
* Xem chi tiết sản phẩm
* Xem sản phẩm theo danh mục
* Liên hệ
* Xem sản phẩm
* Thêm vào giỏ hàng
* Đặt hàng
* Quản lý sản phẩm
* Quản lý danh mục
* Quản lý blog
* Quản lý đơn hàng

### 3.2.2. Yêu cầu phi chức năng

### Hiệu suất:

* + Hệ thống cần đảm bảo khả năng xử lý lớn và đồng thời của dữ liệu để đảm bảo trải nghiệm người dùng mượt mà và không bị gián đoạn.
  + Thời gian phản hồi của hệ thống cần được giữ ở mức thấp, đảm bảo việc truy cập và sử dụng website luôn nhanh chóng và hiệu quả.

### Bảo mật:

* + Hệ thống cần có các biện pháp bảo mật mạnh mẽ để bảo vệ thông tin cá nhân của người dùng, bao gồm mã hóa dữ liệu, kiểm soát truy cập và cơ chế xác thực..
  + Phải có cơ chế sao lưu và khôi phục dữ liệu định kỳ để đảm bảo an toàn và tin cậy cho dữ liệu.

### Tương thích:

* + Website cần phải tương thích với nhiều loại trình duyệt web phổ biến như Chrome, Firefox, Safari và Edge để đảm bảo mọi người dùng có thể truy cập và sử dụng một cách thuận tiện.
  + Giao diện của website cũng cần phải tương thích với các thiết bị di động như điện thoại thông minh và máy tính bảng.

### Dễ dàng điều hướng:

* + Giao diện người dùng cần phải được thiết kế sao cho dễ dàng điều hướng và tìm kiếm thông tin, giúp người dùng tìm được thông tin một cách nhanh chóng và thuận tiện.

### Dễ dàng sử dụng:

* + Giao diện người dùng cần phải được thiết kế đơn giản và dễ sử dụng, giúp người dùng mới dễ dàng tiếp cận và sử dụng hệ thống một cách tự nhiên.

### Bảo Mật Thông Tin:

* + Đảm bảo an toàn và bảo mật cho thông tin cá nhân của người tìm việc và nhà tuyển dụng.
  + Áp dụng các biện pháp bảo mật để ngăn chặn sự truy cập trái phép vào hệ thống và lợi dụng thông tin.

### Giao Diện Thân Thiện:

* + Xây dựng giao diện dễ sử dụng và thân thiện với người dùng, bao gồm cả trên máy tính và thiết bị di động.
  + Đảm bảo trải nghiệm người dùng mượt mà và hiệu quả.

.