

LỜI CẢM ƠN

Trên thực tế, không có sự thành công nào, không cần đến sự giúp đỡ từ người khác dù ít hay nhiều. Để đồ án được thành công như ngày hôm nay, trước hết tôi xin gửi lời cảm ơn sâu sắc đến quý thầy (cô) bộ môn Công nghệ Thông tin đã truyền đạt những kiến thức, kinh nghiệm quý báu cho tôi trong suốt quá trình học tập vừa qua.

Đặc biệt, tôi xin bày tỏ tình cảm và lòng biết ơn đến thầy Phạm Minh Đương, người đã tận tình hướng dẫn, giúp đỡ tôi trong việc học tập cũng như trong thời gian thực hiện đồ án cơ sở ngành.

Trong quá trình hoàn thành đề tài, với điều kiện thời gian cũng như kinh nghiệm còn hạn hẹp, tôi không thể tránh khỏi những sai sót. Vì vậy, tôi rất mong nhận được những ý kiến đóng góp của quý thầy (cô) để em có thể khắc phục những thiếu sót còn lại. Đây chắc chắn sẽ là những kiến thức đáng quý và cũng là hành trang để tôi có thể vững bước sau này.

Một lần nữa tôi xin chân thành cảm ơn.

MỤC LỤC

DANH MỤC HÌNH ẢNH – BẢNG BIỂU	6
TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH.....	8
MỞ ĐẦU	9
1. Lý do chọn đề tài	9
2. Mục tiêu của đề tài	9
3. Nội dung nghiên cứu	9
4. Phương pháp nghiên cứu.....	9
5. Đối tượng và phạm vi nghiên cứu.....	10
CHƯƠNG 1. TỔNG QUAN	11
CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT	12
2.1. Giới thiệu về Bootstrap	12
2.1.1 Khái niệm Bootstrap	12
2.1.2. Lịch sử của Bootstrap	12
2.2. Mô hình MVC	12
2.2.1. Khái niệm mô hình MVC	12
2.2.2. Thành phần trong mô hình MVC.....	13
2.2.3. Luồng xử lý mô hình MVC	14
2.3. Hệ quản trị cơ sở dữ liệu MySQL	14
2.3.1. Khái niệm hệ quản trị cơ sở dữ liệu MySQL	14
2.3.2. Đặc điểm nổi bật của MySQL	15
2.4. Giới thiệu về ReactJS	15
2.4.1 Khái niệm.....	15
2.4.2 Thành phần cơ bản của ReactJS (components)	15
2.4.3 Ưu và nhược điểm của ReactJS	17
2.5. Nghiệp vụ liên quan đến đề tài.....	18

2.5.1. Khái niệm bán hàng online	18
2.5.2. Giới thiệu về dịch vụ đồ ăn vặt.....	18
2.6. Các công trình nghiên cứu liên quan	19
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU	20
3.1 Mô tả tổng quan về website	20
3.2 Các chức năng của website	20
3.3 Sơ đồ cấu trúc của website.....	21
3.4 Các bước xây dựng website bằng ReactJS.....	22
3.4.1 Chuẩn bị cài đặt	22
3.4.1.1 Cài đặt một số thư viện cần thiết cho dự án	24
3.4.1.2 Cài đặt NodeJS	26
3.4.1.3 Xây dựng Router đường dẫn tới các trang	27
3.4.1.4 Xây dựng Component cho slideshow ở trang chủ	27
3.4.2 Thiết kế giao diện	28
3.4.2.1 Xây dựng giao diện trang chủ.....	28
3.4.2.2 Xây dựng giao diện sản phẩm	32
3.4.2.3 Xây dựng giao diện trang bài viết.....	32
3.4.2.4 Xây dựng giao diện trang đăng nhập.....	33
3.4.2.5 Xây dựng giao diện trang đăng ký.....	34
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU	36
CHƯƠNG 5: HƯỚNG PHÁT TRIỂN.....	42
5.1 Kết luận	42
5.2 Hướng phát triển	42
TÀI LIỆU THAM KHẢO.....	43

DANH MỤC HÌNH ẢNH – BẢNG BIỂU

Hình 2-1. Mô hình MVC	13
Hình 2-2. Luồng xử lý mô hình MVC.....	14
Hình 3-1. Sơ đồ cấu trúc website	21
Hình 3-2. Cài đặt ReactJS	22
Hình 3-3. Cấu trúc thư mục của ReactJS	22
Hình 3-4. Cài đặt Yarn	23
Hình 3-5. Chạy project với localhost 3000	23
Hình 3-6. Cấu trúc thư mục dự án	23
Hình 3-7. Cài đặt React-Router-DOM	24
Hình 3-8. Cài đặt thành công React-Router	24
Hình 3-9. Cài đặt React-icons	24
Hình 3-10. Cài đặt React Slick	25
Hình 3-11. Cài đặt Styled-Component	25
Hình 3-12. Cài đặt React Redux	25
Hình 3-13. Cài đặt Redux Toolkit	26
Hình 3-14. Cài đặt Ant Design	26
Hình 3-15. Cài đặt NodeJS	27
Hình 3-16. Xây dựng đường dẫn cho từng trang.....	27
Hình 3-17. Xây dựng Component cho slideshow ở trang chủ	28
Hình 3-18. Xây dựng phần Header.....	28
Hình 3-19. Xây dựng phần Footer.....	29
Hình 3-20. Xây dựng phần Slider.....	30
Hình 3-21. Xây dựng trang HomePage	31
Hình 3-22. Xây dựng giao diện sản phẩm.....	32
Hình 3-23. Xây dựng giao diện trang bài viết.....	33

Hình 3-24. Xây dựng giao diện trang đăng nhập	33
Hình 3-25. Xây dựng giao diện trang đăng ký	34
Hình 3-26. Xây dựng giao diện trang sản phẩm chi tiết.....	35
Hình 4-1. Giao diện trang chủ	37
Hình 4-2. Giao diện sản phẩm.....	38
Hình 4-3. Giao diện sản phẩm chi tiết.....	39
Hình 4-4. Giao diện trang liên hệ	39
Hình 4-5. Giao diện trang bài viết.....	40
Hình 4-6. Giao diện trang đăng nhập	41
Hình 4-7. Giao diện trang đăng ký	41

TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH

Tổng quan đề tài “Tìm hiểu ReactJS và ứng dụng xây dựng website bán đồ ăn vặt” đầu tiên là tìm hiểu nghiên cứu khái niệm và cách hoạt động của ReactJS. Khái niệm thành phần cơ bản ReactJS như: components, JSX, props, state, virtual DOM, lifecycle methods...ưu điểm và nhược điểm của ReactJS.

Tìm hiểu khái niệm và cách sử dụng của những thư viện và ngôn ngữ có sử dụng trong đề tài như: HTML, Javascript, CSS, Bootstrap, Ant Design, React-Router-DOM, React Slick, React-icons...

Thiết kế mô hình MVC (Model-View-Controller) mẫu kiến trúc phần mềm để tạo lập giao diện người dùng trên máy tính. Tìm hiểu từng thành phần của mô hình như: Model, View, Controller và luồng xử lý của từng thành phần trên.

Nghiên cứu những nghiệp vụ liên quan đến đề tài như: khái niệm về bán hàng online, cách thức bán hàng online sau đó xây dựng nên thành phần dự án cho đề tài.

Tìm kiếm những tài liệu, sách vở, báo chí, kênh youtube và các công trình nghiên cứu liên quan đến đề tài. Và cuối cùng vận dụng tổng hợp những kiến thức trên thực hiện đề tài “Tìm hiểu ReactJS và ứng dụng xây dựng website bán đồ ăn vặt”.

Website được thiết kế có các trang như: trang chủ, bài viết, danh mục sản phẩm, thông tin liên hệ, đăng nhập và đăng ký. Trong phần danh mục được chia làm ba danh mục như: thức ăn nhẹ, đồ uống và trái cây để khách hàng có thể dễ dàng lựa chọn. Website có giao diện đẹp mắt, bố cục rõ ràng, màu sắc hài hòa và thân thiện với người dùng.

MỞ ĐẦU

1. Lý do chọn đề tài

Ngày nay, việc xây dựng ứng dụng website kết hợp với thương mại điện tử cũng đã trở thành một xu hướng trong thời đại công nghệ như hiện nay.

Framework là một công cụ phát triển phần mềm được sử dụng để xây dựng ứng dụng, website hoặc các hệ thống phức tạp. Nó cung cấp một cấu trúc và tập hợp các thư viện, công cụ, quy tắc phát triển giúp cho việc xây dựng phần mềm trở nên dễ dàng và hiệu quả hơn, trong số đó không thể nào không kể đến ReactJS. ReactJS là một thư viện Javascript mã nguồn mở thiết kế bởi Facebook cho việc phát triển ứng dụng web, được giới công nghệ ra mắt lần đầu vào năm 2013.

Từ đó đến nay, ReactJS đã trở thành một trong những công nghệ phát triển web phổ biến nhất trên thế giới bởi những tính năng linh hoạt và đơn giản với hơn 1,300 developer và hơn 94,000 trang web sử dụng framework này.

Bên cạnh đó, nhu cầu ăn uống của con người chưa bao giờ giảm, nhất là đối với học sinh, sinh viên, dân công sở, văn phòng. Tận dụng điều đó, tôi đã ứng dụng ReactJS và việc xây dựng website bán đồ ăn vặt, nhằm mục đích áp dụng công nghệ mới vào thiết kế website bán hàng trực tuyến bên cạnh đó cũng đề cập đến những ưu điểm trên vào trong đề tài.

2. Mục tiêu của đề tài

Tìm hiểu lý thuyết về ReactJS.

Xác định cấu trúc, thành phần chính trong ReactJS.

Xây dựng một website bán đồ ăn vặt có sử dụng ReactJS.

Tích hợp một vài tính năng như đặt hàng, thêm sản phẩm.

3. Nội dung nghiên cứu

Tìm hiểu khái niệm, thành phần cơ bản và chức năng của ReactJS.

Các tính năng và cách ứng dụng ReactJS vào thiết kế website bán đồ ăn vặt.

Tìm hiểu nghiên cứu và xây dựng website bán đồ ăn website bán đồ ăn vặt tích hợp thêm sản phẩm vào giỏ hàng.

4. Phương pháp nghiên cứu

Nghiên cứu lý thuyết: Tìm hiểu và phân tích các bài báo, tài liệu, sách vở và những trang internet liên quan đến đề tài từ đó chọn lọc thông tin để viết báo cáo.

Nghiên cứu thực nghiệm: Xây dựng website bán đồ ăn vặt minh họa cho vấn đề nghiên cứu.

5. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu

ReactJS.

Website bán đồ ăn vặt.

Bootstrap và các Component khác.

Phạm vi nghiên cứu

Tìm hiểu khái quát về ReactJS, nắm được các định nghĩa, khái niệm, cách render, import trong React và các thuộc tính chủ chốt như prop, state...

Với đề tài thiết kế website bán đồ ăn vặt, khách hàng có thể thấy giao diện dễ dàng sử dụng, đẹp mắt, bố cục rõ ràng, tối ưu hóa trang web để tăng tốc độ tải trang.

CHƯƠNG 1. TỔNG QUAN

ReactJS là một thư viện Javascript phổ biến, phát triển tại Facebook được sử dụng để xây dựng giao diện người dùng động và tương tác trên website. Nó giúp tạo ra các ứng dụng web đáp ứng, linh hoạt và dễ bảo trì.

Trong ReactJS thường dùng Javascript để thiết kế bố cục cho trang web, nhưng nhược điểm là cấu trúc khá khó. Thay vào đó sử dụng JSX và nhúng các đoạn HTML vào JavaScript, ta thấy cú pháp dễ hiểu hơn và JSX cũng có thể tối ưu code khi biên soạn. Vừa dễ cho người lập trình mà vừa tiện cho việc biên dịch.

Với ReactJS, chúng ta có thể tận dụng các thành phần tái sử dụng để xây dựng giao diện linh hoạt và dễ dàng mở rộng. Đồng thời, nó cũng hỗ trợ việc quản lý trạng thái ứng dụng một cách hiệu quả, giúp tăng tính ổn định và hiệu suất của trang web.

Trong đề tài này, sẽ tìm hiểu về ReactJS và ứng dụng vào thiết kế một trang web bán đồ ăn vặt. Website bán đồ ăn vặt sẽ cung cấp đầy đủ thông tin của sản phẩm như giá cả, thông tin, hình ảnh. Mọi thông tin về sản phẩm ăn vặt sẽ được đăng tải đầy đủ cùng với các hình ảnh giúp khách hàng nắm được rõ thông tin một cách đầy đủ, chính xác và rõ ràng.

Ngoài những thông tin trên website bán đồ ăn vặt cũng tích hợp thêm một vài tính năng giới hạn ở việc khách hàng lựa chọn sản phẩm thêm vào giỏ hàng, và tiến hành điền thông tin khách hàng để gửi hóa đơn đến admin. Về phía admin sẽ tiếp nhận đơn hàng, thực hiện giao hàng đồng thời có thể quản lí, thêm, xóa sản phẩm và chỉnh sửa giao diện website bán hàng của mình.

Tóm lại, đề tài này giúp chúng ta hiểu cách được hoạt động, các thành phần của ReactJS, một số khái niệm liên quan đến đề tài, và ứng dụng chúng vào việc xây dựng thiết kế website bán đồ ăn vặt.

CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT

2.1. Giới thiệu về Bootstrap

2.1.1 Khái niệm Bootstrap

Bootstrap được phát triển bởi hai kỹ sư công nghệ đó là Mark Otto và Jacob Thornton khi cả hai đang làm việc tại Twitter. Bootstrap là một framework front-end phổ biến được sử dụng để xây dựng giao diện web. Nó cung cấp một tập hợp các công cụ và tài nguyên giúp tạo ra các trang web đáng tin cậy, linh hoạt và tương thích trên nhiều thiết bị khác nhau. Bootstrap sử dụng HTML, CSS và JavaScript để xây dựng các thành phần và bố cục của trang web một cách nhanh chóng và dễ dàng.

2.1.2. Lịch sử của Bootstrap

Phiên bản thứ 1: ra mắt lần đầu tiên vào ngày 19/8/2011 trên mạng xã hội GitHub và có tên là Twitter Blueprint.

Phiên bản thứ 2: phát hành ngày 31/1/2012, với nhiều cải tiến đáng kể liên quan đến thiết kế, tính năng và hiệu suất.

Phiên bản thứ 3: phát hành ngày 19/8/2013, phiên bản này tập trung vào việc hỗ trợ thiết kế web responsive, tức là các trang web có thể tự điều chỉnh kích thước theo kích cỡ màn hình của thiết bị.

Phiên bản thứ 4: được công bố vào tháng 10/2014, phiên bản này đã chuyển từ việc sử dụng Less sang Sass để biên dịch mã CSS có độ linh hoạt cao hơn.

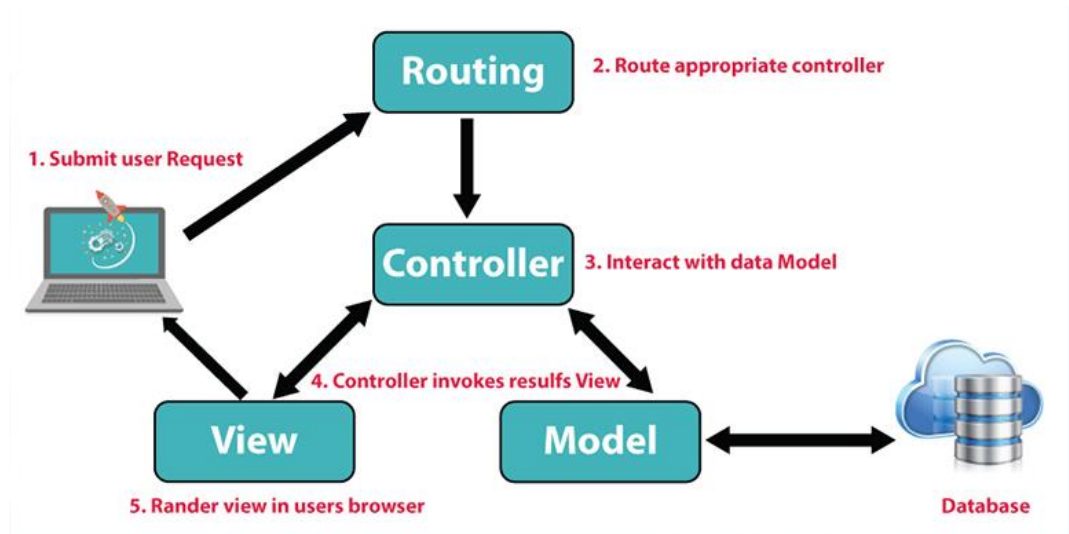
Phiên bản thứ 5: phát hành ngày 5/5/2020, phiên bản này loại bỏ sự phụ thuộc vào jQuery, và sử dụng các tính năng mới của JavaScript như ES6 và Fetch API. Ngoài ra, cũng cập nhật các icon, màu sắc, font chữ và các thành phần giao diện khác.

2.2. Mô hình MVC

2.2.1. Khái niệm mô hình MVC

MVC là viết tắt của cụm từ “Model-View-Controller”, đây là mô hình thiết kế được sử dụng trong kỹ thuật phần mềm. MVC là một mẫu kiến trúc phần mềm

để tạo lập giao diện người dùng trên máy tính. Mô hình chia thành ba phần được kết nối với nhau và mỗi thành phần đều có một nhiệm vụ riêng của nó và độc lập với các thành phần khác.



Hình 2-1. Mô hình MVC

2.2.2. Thành phần trong mô hình MVC

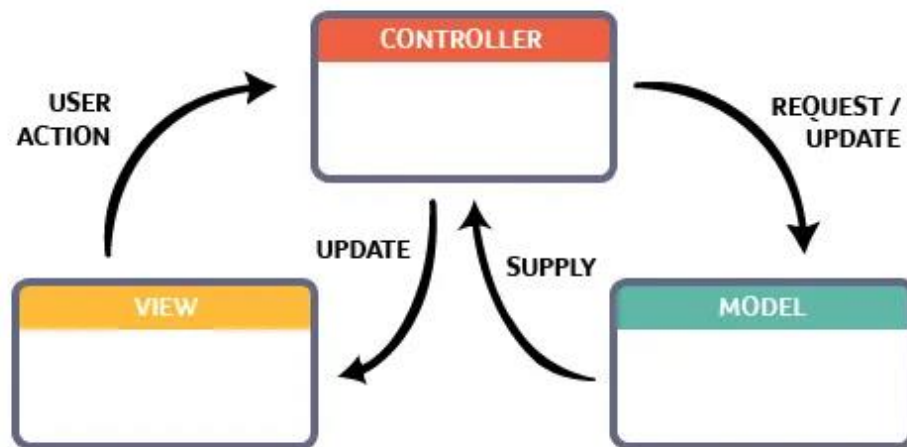
Model (M): chứa một cấu trúc dữ liệu có chức năng lưu trữ toàn bộ các thông tin dữ liệu của một ứng dụng. Thành phần này là một cầu nối giữa hai thành phần bên dưới là View và Controller. Model thể hiện rõ các thao tác với cơ sở dữ liệu như cho phép xem, truy xuất, xử lý dữ liệu,...

View (V): là các thành phần dùng để thể hiện giao diện người dùng (UI). Đảm nhận việc hiển thị, trả về thông tin, dữ liệu cho end-user. Ví dụ: nút “delete” được tạo bởi View khi người dùng nhấn vào nút đó sẽ có một hành động trong Controller.

Controller (C): giữ nhiệm vụ tiếp nhận yêu cầu từ người dùng, sau đó gửi lệnh đến Model để lấy hay cập nhật dữ liệu, rồi truyền lệnh đến View để cập nhật giao diện. Đây còn là nơi quản lý sự trao đổi dữ liệu và nguyên tắc nghề nghiệp trong các thao tác liên quan đến mô hình. Controller giữ vai trò trung gian giữa Model và View.

2.2.3. Luồng xử lý mô hình MVC

1. Khi người dùng tương tác với View (ví dụ: nhấn vào một nút), giao diện sẽ gửi thông tin đến Controller.
2. Controller nhận thông tin từ View và xử lý yêu cầu. Sau đó nó sẽ gọi phương thức trong Model để lấy hoặc cập nhật dữ liệu.
3. Model là nơi xử lý và lưu trữ dữ liệu. Nó sẽ lấy thông tin từ cơ sở dữ liệu (nếu có) hoặc thực hiện các thao tác xử lý dữ liệu. Tiếp theo, nó trả về kết quả cho Controller.
4. Controller nhận kết quả từ Model và quyết định các hiển thị dữ liệu trên View. View sẽ cập nhật giao diện để hiển thị thông tin mới nhận được từ Model.



Hình 2-2. Luồng xử lý mô hình MVC

2.3. Hệ quản trị cơ sở dữ liệu MySQL

2.3.1. Khái niệm hệ quản trị cơ sở dữ liệu MySQL

MySQL là hệ quản trị cơ sở dữ liệu mã nguồn mở miễn phí nằm trong nhóm LAMP (Linux – Apache -MySQL – PHP) và Microsoft (Windows, IIS, SQL Server, ASP/ASP.NET), vì MySQL được tích hợp sử dụng chung với apache, PHP nên nó phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Chúng được sắp xếp rõ ràng và phân lớp ngăn nắp những thông tin mình lưu trữ, vì vậy có thể truy cập vào dữ liệu một cách rất nhanh chóng và thuận tiện.

Các website nổi tiếng như Facebook, Google, Twitter, Youtube... đều đang sử dụng MySQL để lưu trữ thông tin [1].

2.3.2. Đặc điểm nổi bật của MySQL

Tốc độ truy vấn và tốc độ phản hồi dữ liệu tương đối nhanh chóng.

MySQL có thể xử lý một lượng lớn dữ liệu và có khả năng được phát triển và mở rộng dễ dàng.

Sử dụng miễn phí cho hầu hết các việc sử dụng trong một tổ chức.

Kết nối và bảo mật, MySQL được kết nối các mạng đầu đủ, các cơ sở dữ liệu có thể truy vấn bất cứ nơi nào trên Internet và có thể chia sẻ với bất kì ai, bất kì nơi nào. Nhưng MySQL trang bị các kỹ thuật mạnh, chỉ có người sử dụng đã được xác nhận mới truy cập được vào cơ sở dữ liệu.

Quản lý dễ dàng.

2.4. Giới thiệu về ReactJS

2.4.1 Khái niệm

ReactJS là một thư viện Javascript mã nguồn mở được sử dụng để xây dựng giao diện người dùng cho các website. ReactJS được phát triển bởi các kỹ sư phần mềm của Facebook và được ra mắt lần đầu năm 2013. Đồng thời, đang được các công ty nổi tiếng sử dụng để phát triển sản phẩm của họ như Netflix, Uber, Paypal, Intstagram... ReactJS sử dụng cơ chế “Virtual DOM” để cải thiện hiệu suất và tốc độ của ứng dụng. Nó cũng hỗ trợ tái sử dụng các thành phần, giúp tăng tính modular và dễ bảo trì của mã nguồn.

2.4.2 Thành phần cơ bản của ReactJS (components)

Component: là các khối xây dựng cơ bản của ứng dụng, được sử dụng để tạo ra giao diện người dùng, bao gồm HTML, CSS và JavaScript. Mỗi component là một đơn vị độc lập có thể được sử dụng lại trong nhiều vị trí khác nhau của ứng dụng.

Ví dụ, một component đơn giản trong ReactJS:

```
import React from 'react';

function Hello(props) {

    return <div>Hello {props.name}!</div>;

}

export default Hello;
```

JSX: là một phần mở rộng của JavaScript, cho phép người phát triển ReactJS viết mã HTML tương tự như cách họ viết mã HTML trong các tệp riêng biệt. JSX giúp tăng tính tường minh của mã và giúp cho việc phát triển ứng dụng ReactJS trở nên dễ dàng hơn.

Ví dụ, viết mã HTML trong JavaScript như sau:

```
const element = <h1>Hello, world!</h1>;
```

Props: viết tắt của từ properties là một cách truyền dữ liệu từ các component cha (parent component) sang component con (child component). Props là các đối số (arguments) được truyền vào cho một component và có thể được sử dụng bên trong component đó.

Ví dụ:

```
<Hello name="Thu" age={20} />
```

Trong ví dụ trên, component “Hello” được truyền hai props là “name” và “age”. Props “name” có giá trị là “Thu” và props “age” có giá trị là 20.

State: là một đối tượng JavaScript chứa các thông tin có thể thay đổi trong một component. Khi state thay đổi, ReactJS sẽ tự động cập nhật giao diện người dùng.

Ví dụ, một component Counter sử dụng state để theo dõi và hiển thị giá trị của một biến đếm:

```
import React, { useState } from 'react';

function Counter() {

  const [count, setCount] = useState(0);

  const increment = () => {

    setCount(count + 1);

  };

  return (

    <div>

      <p>Count: {count}</p>

      <button onClick={increment}>Increment</button>

    </div>

  );

}
```

Virtual DOM: viết tắt của từ Virtual Document Object Model giúp tối ưu hiệu suất và tăng tốc độ render trong ứng dụng ReactJS. Thay vì cập nhật trực tiếp các phần tử trên trang web mỗi khi có sự thay đổi, React chỉ cập nhật những phần tử thực sự cần thiết, giúp giảm thiểu số lần truy cập và thay đổi trực tiếp vào DOM.

Lifecycle methods: là phương thức vòng đời, các phương thức này được gọi tự động tại các giai đoạn khác nhau trong quá trình “vòng đời” của một component. Các phương thức này cho phép thực hiện các hành động như khởi tạo, cập nhật và hủy bỏ một component.

2.4.3 Ưu và nhược điểm của ReactJS

Ưu điểm

Hiệu suất cao và tốc độ nhanh, ReactJS sử dụng Virtual DOM để tối ưu hóa hiệu suất và tăng tốc độ của ứng dụng.

Phát triển các thành phần độc lập, dễ dàng tái sử dụng và kết hợp các thành phần để tạo nên nhiều ứng dụng phức tạp.

Giúp cho việc quản lý và kiểm soát các thành phần trở nên dễ dàng, bảo trì và cập nhật ứng dụng trở nên thuận tiện.

Nhược điểm

Việc học và sử dụng ReactJS khá phức tạp cho người mới bắt đầu.

ReactJS không cung cấp một giải pháp chính thức để quản lý trạng thái ứng dụng.

Ứng dụng ReactJS có thể trở nên chậm do cách ReactJS hoạt động và cách nó cập nhật lại giao diện người dùng.

Để sử dụng ReactJS hiệu quả, người phát triển cần phải hiểu rõ về JavaScript và các khái niệm liên quan như JSX, components, props, state...

2.5. Nghiệp vụ liên quan đến đề tài

2.5.1. Khái niệm bán hàng online

Bán hàng online là hoạt động buôn bán, trao đổi hàng hóa, dịch vụ trên không gian mạng Internet. Nếu bán hàng truyền thống người bán cần có cửa hàng vật lý thì bán hàng online lại không cần phải có một cửa hàng hiện hữu. Nhà bán hàng cũng không cần quản lý quá nhiều nhân sự để vận hành cửa hàng online.

Các hình thức bán hàng online:

Bán hàng qua mạng xã hội như Facebook, Tiktok, Zalo, Youtube...

Bán hàng qua sàn thương mại điện tử như Shoppe, Lazada, Tiki...

Bán hàng qua website như vatgia.com, enbac.com...

2.5.2. Giới thiệu về dịch vụ đồ ăn vặt

Dịch vụ đồ ăn vặt là các hoạt động kinh doanh cung cấp đồ ăn nhẹ, phổ biến ở thị trường ẩm thực hiện nay. Những dịch vụ này thường tập trung vào việc cung cấp các loại thức ăn đựng trong hộp hoặc đóng gói sẵn, tiện lợi để khách hàng có thể dễ dàng mang đi hoặc ăn tại chỗ. Các món ăn vặt thường gồm đồ chiên, trà sữa, bánh tráng, kimbap, trái cây cắt sẵn, cơm trộn...

Dịch vụ đồ ăn vặt có thể hoạt động thông qua cửa hàng cố định, xe đẩy hoặc qua những website và ứng dụng di động để đặt hàng online và giao hàng tận nơi.

Khách hàng thường xuyên lựa chọn đồ ăn vặt để giải quyết nhanh chóng nhu cầu ăn uống hằng ngày, vì vậy dịch vụ đồ ăn vặt là loại hình kinh doanh phổ biến và lợi nhuận.

2.6. Các công trình nghiên cứu liên quan

“React Native: A framework for building native apps using React” là một công trình của Facebook giới thiệu React Native, một framework cho phép xây dựng ứng dụng di động sử dụng JavaScript và React. React Native cho phép các nhà phát triển xây dựng ứng dụng di động với khả năng tái sử dụng mã nguồn giữa các nền tảng khác nhau, bao gồm iOS và Android. Công trình này giới thiệu các khái niệm cơ bản về React Native, cách sử dụng nó để xây dựng ứng dụng di động và cung cấp một số lời khuyên và kinh nghiệm trong quá trình phát triển. Công trình này đã đóng góp đáng kể vào sự phát triển của ReactJS và giúp các nhà phát triển xây dựng các ứng dụng di động hiệu quả hơn.

"React: A JavaScript library for building user interfaces" là một bài báo của Jordan Walke, người sáng lập React. Công trình nghiên cứu này giải thích cách React hoạt động, cung cấp một cái nhìn tổng quan về lý thuyết và thiết kế của nó. Bài báo đã đóng góp quan trọng vào sự phát triển và phổ biến của React trong cộng đồng phát triển web và đã trở thành nền tảng cho việc xây dựng các ứng dụng web hiệu quả và dễ bảo trì.

"ReactJS Best Practices for 2021" là một bài viết của Brad Westfall, một chuyên gia về React và tác giả của nhiều cuốn sách và bài viết về React. Bài viết này giới thiệu các quy tắc và thực hành tốt nhất khi sử dụng ReactJS trong năm 2021. Nội dung của bài viết tập trung vào các chủ đề như cấu trúc component, quản lý trạng thái, xử lý sự kiện, tối ưu hóa hiệu suất và kiểm thử. Bài viết này cung cấp cho các nhà phát triển React các lời khuyên và kinh nghiệm để xây dựng các ứng dụng React hiệu quả và dễ bảo trì. Bài viết này đã được đánh giá cao trong cộng đồng phát triển web và được coi là một tài liệu tham khảo hữu ích cho các nhà phát triển React.

CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

3.1 Mô tả tổng quan về website

Đề tài “Tìm hiểu ReactJS và ứng dụng xây dựng website bán đồ ăn vặt” là một giao diện website chuyên về những món ăn vặt. Được thiết kế mang phong cách tươi trẻ, màu sắc hài hòa, cùng với bố cục rõ ràng. Website này bao gồm trang chủ, sản phẩm, bài viết, liên hệ, đăng ký và đăng nhập:

Trang chủ: có danh mục sản phẩm được phân loại theo từng danh mục và phần banner được xây dựng slideshow.

Sản phẩm: thông tin chia tiết sản phẩm, thức ăn vặt, thức ăn đêm, đồ uống, trái cây, giao diện lọc sản phẩm theo giá tiền, số sao đánh giá và danh mục sản phẩm.

Bài viết: những bài viết giới thiệu về đồ ăn vặt, và về những món ăn phù hợp với sức khỏe.

Liên hệ: địa chỉ email, số điện thoại, facebook, địa chỉ...

Đăng nhập: sử dụng email và mật khẩu để đăng nhập.

Đăng kí: sử dụng email, mật khẩu và nhập lại mật khẩu để đăng kí.

3.2 Các chức năng của website

Yêu cầu xem được các thông tin về sản phẩm, thông tin chi tiết về sản phẩm như tính năng, mẫu mã của sản phẩm các loại đồ ăn vặt, nơi xuất xứ, ngày sản xuất, thông tin khuyến mãi của sản phẩm...

Trang web có phần giới thiệu về công ty và thông tin địa chỉ liên hệ...

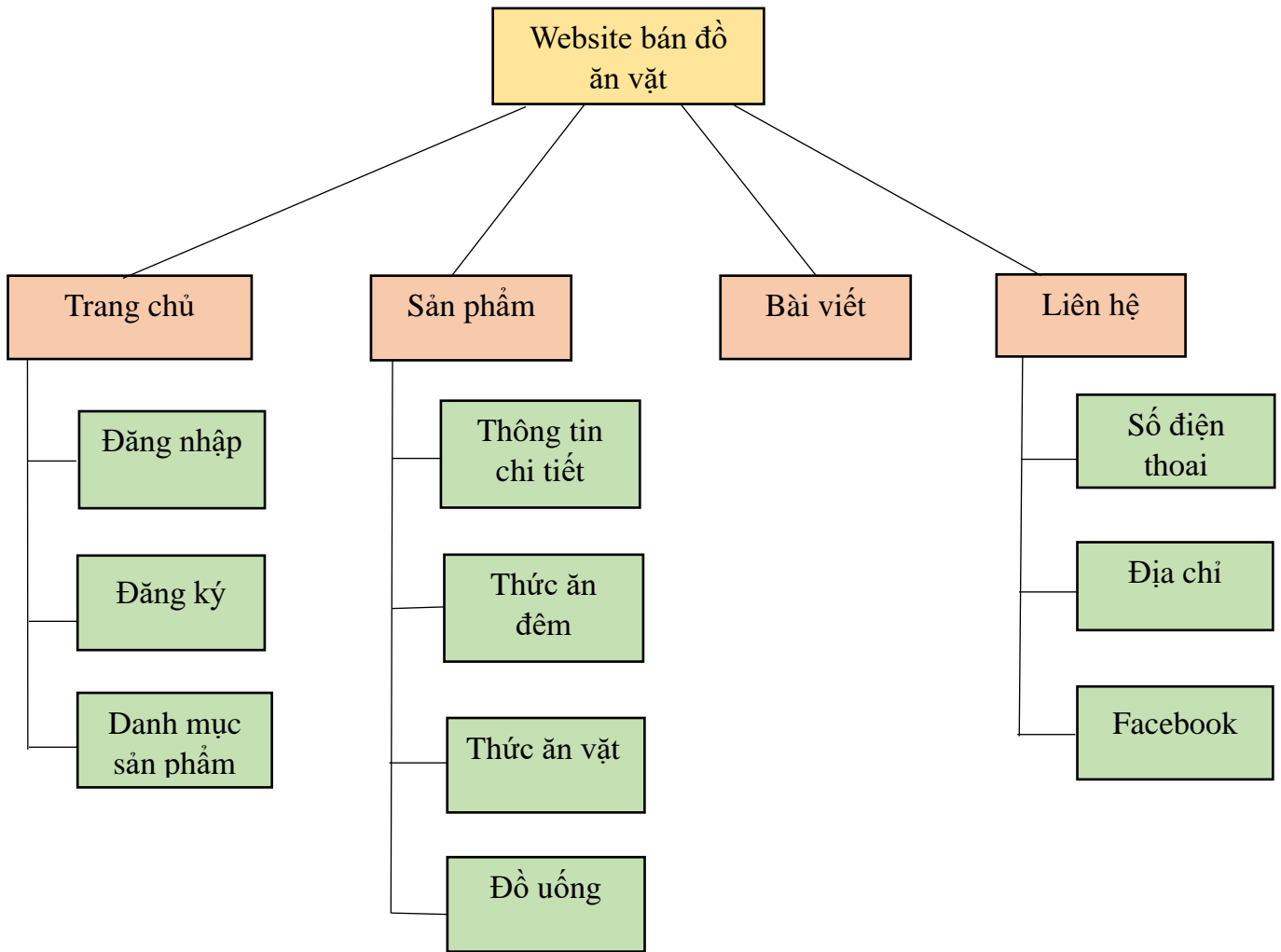
Trang web có phần tin tức về thị trường, thông tin khuyến mãi... giúp khách hàng có đầy đủ thông tin, có thêm kiến thức nhằm giúp người mua lựa chọn được sản phẩm ưng ý nhất.

Hiện thị danh sách các mặt hàng theo từng danh mục sản phẩm để khách hàng xem và chọn mua.

Màu sắc hài hòa, bố cục rõ ràng, thân thiện, không gây bất tiện cho khách hàng.

Yêu cầu phải đăng nhập hoặc đăng ký khi khách hàng muốn mua hàng.

3.3 Sơ đồ cấu trúc của website



Hình 3-1. Sơ đồ cấu trúc website

Mô tả: Website bán đồ ăn vặt được thiết kế với 4 trang chính: trang chủ, sản phẩm, bài viết, liên hệ.

Trang chủ gồm: trang đăng nhập, đăng ký và có danh mục sản phẩm, ở phần danh mục sản phẩm được chia ra làm 3 phần là sản phẩm nổi bật, sản phẩm mới, đồ uống và trái cây.

Trang sản phẩm gồm: tất cả các loại sản phẩm đồ ăn vặt và thông tin chi tiết từng sản phẩm.

Trang bài viết thể hiện những gì mới sắp ra mắt của website, cách thức ăn uống để có sức khỏe tốt cuối cùng là thể hiện sự uy tín của sản phẩm ở website.

Trang liên hệ thể hiện địa chỉ của công ty, số điện thoại liên hệ, facebook hoặc khách hàng có thể góp ý trực tiếp trong trang này.

3.4 Các bước xây dựng website bằng ReactJS

3.4.1 Chuẩn bị cài đặt

3.4.1.1 Cài đặt ReactJS

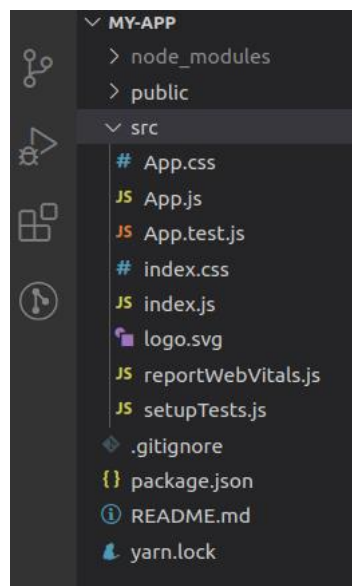
Tích hợp ReactJS vào website

Chạy lệnh `npx create-react-app ghienanvat_reactjs` để tải ReactJS về trong thư mục có tên “ghienanvat_reactjs”

```
luuth@TThu MINGW64 /d/DO_AN_CO_NGANH_2023/csn-da21tta-luuthikimthu-bandoanvat-re  
actjs (master)  
$ npx create-react-app ghienanvat_reactjs
```

Hình 3-2. Cài đặt ReactJS

Cấu trúc thư mục của ReactJS



Hình 3-3. Cấu trúc thư mục của ReactJS

Cài đặt Yarn cho dự án

Chạy lệnh `npm i yarn` để cài đặt yarn, trong project này sử dụng yarn để thực thi dự án

```
luuth@TThu MINGW64 /d/DO_AN_CO_NGANH_2023/csn-da2itta-luuthikimthu-bandoanvat-reactjs/ghienanvat_reactjs (master)
$ npm i yarn

added 1 package, and audited 1652 packages in 11s

259 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (2 moderate, 6 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

Hình 3-4. Cài đặt Yarn

Chạy ReactJS với localhost 3000

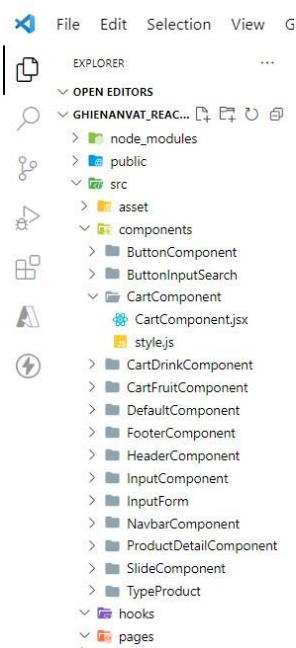
Nhập lệnh yarn start để chạy project với localhost 3000



Hình 3-5. Chạy project với localhost 3000

Tạo component cho dự án

Mỗi project gồm nhiều component, component được tạo ra để thực hiện một chức năng nào đó. Ví dụ như FooterComponent, HeaderComponent, CartComponent, ButtonComponent... Mỗi component là một chi tiết quan trọng:



Hình 3-6. Cấu trúc thư mục dự án

Cài đặt React-Router-DOM cho dự án

Chạy lệnh **npm i react-router-dom@6** như hình để cài đặt **React- Router**

```
luuth@TThu MINGW64 /d/DO_AN_CO_NGANH_2023/csn-da21tta-luuthikimthu-bandoanvat-reactjs/ghienanvat_reactjs (master)
$ npm i react-router-dom@6

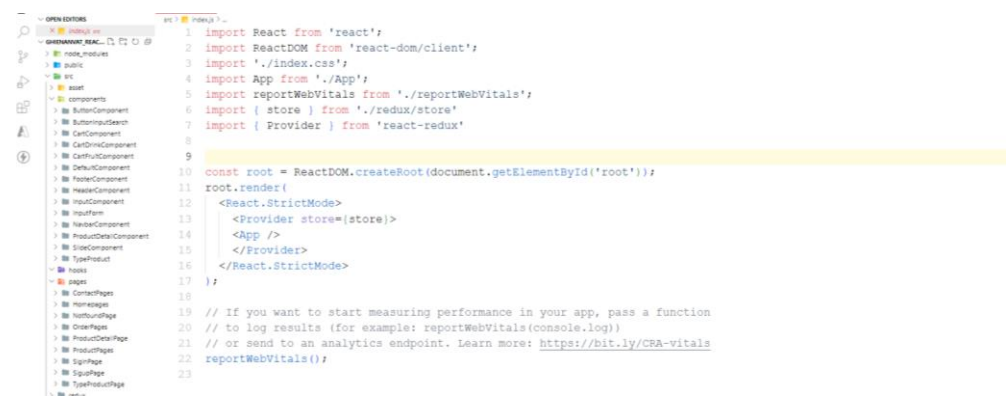
added 3 packages, and audited 1644 packages in 16s

259 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (2 moderate, 6 high)
```

Hình 3-7. Cài đặt React-Router-DOM

Cài đặt thành công React- Router



```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';
import { store } from './redux/store';
import { Provider } from 'react-redux';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <Provider store={store}>
      <App />
    </Provider>
  </React.StrictMode>
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

Hình3-8. Cài đặt thành công React-Router

3.1.1.2 Cài đặt một số thư viện cần thiết cho dự án

React-icons

Chạy lệnh **npm i react-icons** để cài đặt thư viện **react-icons**

```
PS D:\DO_AN_CO_NGANH_2023\csn-da21tta-luuthikimthu-bandoanvat-reactjs\ghienanvat_reactjs> npm i react-icons

up to date, audited 1651 packages in 6s

259 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (2 moderate, 6 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

Hình 3-9. Cài đặt React-icons

React Slick

Chạy lệnh **npm i react-slick** để cài đặt thư viện, react slick là một thư viện carousel được sử dụng để tạo các slideshow trên trang web

```
luuth@TThu MINGW64 /d/DO_AN_CO_NGANH_2023/csn-da21tta-luuthikimthu-bandoanvat-reactjs/ghienanvat_reactjs (master)
$ npm i react-slick

added 2 packages, and audited 1646 packages in 11s

259 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (2 moderate, 6 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

Hình 3-10. Cài đặt React Slick

Styled-Components

Chạy lệnh **npm i styled-components** để cài đặt thư viện. Styled-Components là một thư viện giúp tổ chức và quản lý code CSS một cách dễ dàng trong các dự án React

```
luuth@TThu MINGW64 /d/DO_AN_CO_NGANH_2023/csn-da21tta-luuthikimthu-bandoanvat-reactjs/ghienanvat_reactjs (master)
$ npm i styled-components

added 12 packages, and audited 1641 packages in 13s

259 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (2 moderate, 6 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

Hình 3-11. Cài đặt Styled-Component

React Redux

Chạy lệnh **npm i react-redux** để cài đặt thư viện

```
luuth@TThu MINGW64 /d/DO_AN_CO_NGANH_2023/csn-da21tta-luuthikimthu-bandoanvat-reactjs/ghienanvat_reactjs (master)
$ npm i react-redux

added 3 packages, and audited 1629 packages in 12s

256 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (2 moderate, 6 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

Hình 3-12. Cài đặt React Redux

Redux Toolkit

Chạy lệnh **npm i @reduxjs/toolkit** để cài đặt thư viện

```
luuth@TThu MINGW64 /d/DO_AN_CO_NGANH_2023/csn-da21tta-luuthikimthu-bandoanvat-reactjs/ghienanvat_reactjs (master)
$ npm install @reduxjs/toolkit

added 5 packages, and audited 1626 packages in 10s

256 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (2 moderate, 6 high)

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

Hình 3-13. Cài đặt Redux Toolkit

Ant Design

Chạy lệnh **npm i antd** để cài đặt thư viện, Ant Design là một hệ thống thiết kế (Design System) chuyên nghiệp mà các doanh nghiệp có thể dùng ngay lập tức. Nó cung cấp các component và demo chất lượng cao để xây dựng các UI có thể tương tác. Bạn cũng có thể điều chỉnh các component này để hợp nhu cầu thiết kế của mình.

```
luuth@TThu MINGW64 /d/DO_AN_CO_NGANH_2023/csn-da21tta-luuthikimthu-bandoanvat-reactjs/ghienanvat_reactjs (master)
$ npm install antd

added 66 packages, and audited 1621 packages in 20s

255 packages are looking for funding
  run `npm fund` for details

8 vulnerabilities (2 moderate, 6 high)

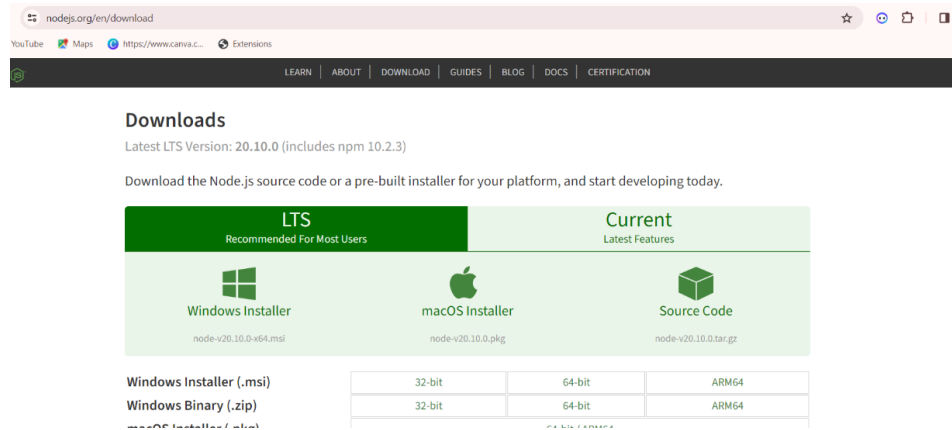
To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.
```

Hình 3-14. Cài đặt Ant Design

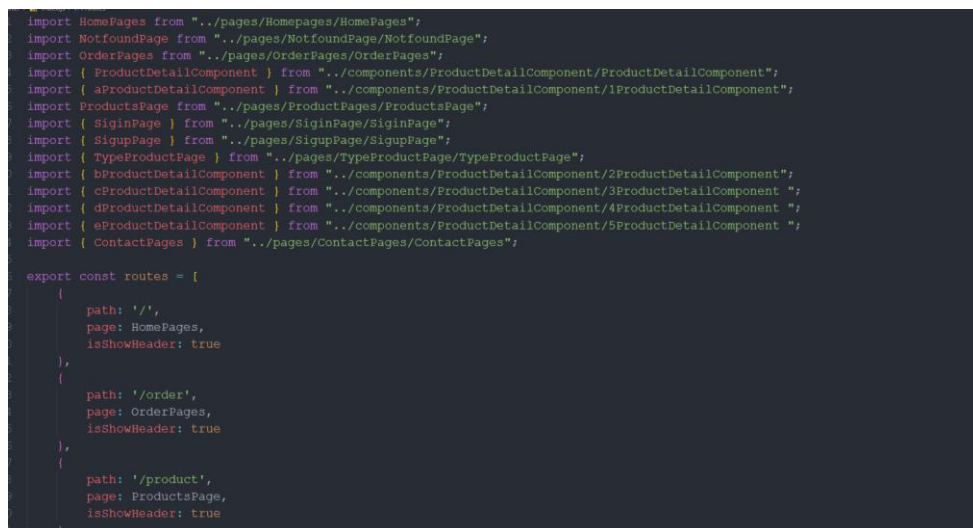
3.1.1.3 Cài đặt NodeJS

Vào <https://nodejs.org> tải bản Nodejs mới nhất, rồi chạy file mới tải cài đặt vào máy



Hình 3-15. Cài đặt NodeJS

3.1.1.4 Xây dựng Router đường dẫn tới các trang



Hình 3-16. Xây dựng đường dẫn cho từng trang

3.1.1.5 Xây dựng Component cho slideshow ở trang chủ



Hình 3-17. Xây dựng Component cho slideshow ở trang chủ

3.4.2 Thiết kế giao diện

3.4.2.1 Xây dựng giao diện trang chủ

Xây dựng phần Header: tạo file HeaderComponent.jsx

Import {Col} from 'antd' để chia header ra 3 phần bằng cách sử dụng thẻ <Col> trong thư viện antd và sau đó viết code như hình để tạo logo của trang web, thanh tìm kiếm, phần đăng nhập/đăng ký và giỏ hàng

```
1 import { Badge, Col } from 'antd';
2 import React from 'react'
3 import { WrapperHeader, WrapperTextHeader, WrapperHeaderAccount, WrapperTextHeaderSmall } from './style';
4 import {
5   UserAddOutlined,
6   BellOutlined,
7   FacebookOutlined,
8   InstagramOutlined,
9   ShoppingCartOutlined,
10 } from '@ant-design/icons';
11 import ButtonInputSearch from './ButtonInputSearch/ButtonInputSearch';
12 import { useNavigate } from 'react-router-dom';
13
14
15 const HeaderComponent = () => {
16   const navigate = useNavigate()
17   const handelNevigatelogin = () =>{
18     navigate('./sign-in')
19   }
20   const handelHomePage = () =>{
21     navigate('./')
22   }
23   return (
24     <div>
25       <WrapperHeader>
26         <Col span={6}>
27           <WrapperTextHeader style={{color: '#000', borderRadius: '5px', }} onClick={handelHomePage}>
28             <span style={{color: '#1c1c1c', background: '#F2C53D'}}>ANVAT</span> <span style={{background: '#09A603', color: '#F2C53D'}}>
29             </WrapperTextHeader>
30         </Col>
31         <Col span={12}>
32           <ButtonInputSearch
33             bordered={false}
34             size='small'
35             placeholder='Bạn đang tìm gì?'
36             textButton='Tìm kiếm'
37           />
38         </Col>
39         <Col span={6} style={{display: 'flex', gap: '20px', alignItems: 'center'}}>
40           <WrapperHeaderAccount>
41             <UserAddOutlined style={{fontSize: '25px', color: '#272727'}} />
42             <div onClick={handelNevigatelogin} style={{cursor: 'pointer'}}>
43               <WrapperTextHeaderSmall>Đăng nhập/ Đăng ký</WrapperTextHeaderSmall>
44             </div>
45             <WrapperTextHeaderSmall>Tài khoản</WrapperTextHeaderSmall>
46             <BellOutlined style={{marginRight: '5px'}} />
47             <FacebookOutlined style={{marginLeft: '5px', marginRight: '5px'}} />
48             <InstagramOutlined />
49           </div>
50         </Col>
51       </WrapperHeader>
52       <div>
53         <WrapperHeaderAccount>
54           <div>
55             <Badge count='4' size='small'>
56               <ShoppingCartOutlined style={{fontSize: '20px', color: '#272727'}} />
57             </Badge>
58             <WrapperTextHeaderSmall>Giỏ hàng</WrapperTextHeaderSmall>
59           </div>
60         </Col>
61       </WrapperHeader>
62     </div>
63   )
64 }
65 export default HeaderComponent;
```

Hình 3-18. Xây dựng phần Header

Xây dựng phần Footer: tạo file FooterComponent.jsx

Viết code như hình:

```
import React from 'react'
import { Link } from 'react-router-dom';
import { AiOutlineFacebook } from 'react-icons/ai';
import { AiOutlineInstagram } from 'react-icons/ai';
import { AiTwitterOutline } from 'react-icons/ai';
import { AiOutlineTiktok } from 'react-icons/ai';
import { Col, Row } from 'antd';
import { WrapperIconFooter } from './style';

const FooterComponent = () => {
  return (
    <div className="footer" style={{ background: '#F0F0F0'}}>
      <div className="container" >
        <Row>
          <Col span={6} style={{paddingLeft: '30px'}}>
            <div className="footer_about">
              <h3><span style={{color: '#1c1c1c', background: '#F2C53D'}}>ANVAT</span><span style={{background: '#09A603', color: '#fff', borderRadius: '5px'}}>
                <div>
                  <li>Địa chỉ: Sơn Thông, Phường 7, Trà Vinh</li>
                  <li>Phone: 0387-521-506</li>
                  <li>Email: luthikimthu2002@gmail.com</li>
                </div>
              </div>
            </Col>
            <Col span={12}>
              <div className="footer_widget">
                <h3><span style={{paddingLeft: '200px'}}>Cửa hàng/hỗ</span>
                <div style={{display: 'flex', padding: '20px'}}>
                  <ul>
                    <li><Link to="/"></li>
                    <li>Liên hệ</li>
                    <li><Link to="/"></li>
                    <li>Thông tin về chúng tôi</li>
                    <li><Link to="/"></li>
                    <li>Sản phẩm kinh doanh</li>
                    <li><Link to="/"></li>
                    <li>Thông tin tài khoản</li>
                    <li><Link to="/"></li>
                    <li>Giới thiệu</li>
                    <li><Link to="/"></li>
                    <li>Danh sách ưu thích</li>
                    <li><Link to="/"></li>
                  </ul>
                </div>
              </div>
            </Col>
            <Col span={6}>
              <div className="footer_widget">
                <h3>Chuyển mail & ưu đãi</h3>
                <p>Đăng kí nhận thông tin tại đây </p>
                <form action="#">
                  <div className="input_group">
                    <input type="text" placeholder="Nhập email"/>
                    <button type="submit" className="btn_submit">
                      Đăng ký
                    </button>
                  </div>
                </form>
              </div>
            </Col>
          </Row>
        </div>
      </div>
    </div>
  );
};

export default FooterComponent;
```

Hình 3-19. Xây dựng phần Footer

Xây dựng phần Slider

Tạo SliderComponent.jsx:

import Slider from 'react-slick' để sử dụng mẫu slide có sẵn trong thư viện React slick

```
import { Image } from 'antd';
import React from 'react';
import Slider from 'react-slick';

const SlideComponent = ({arrImages}) => {
  const settings = {
    dots: true,
    infinite: true,
    speed: 500,
    slidesToShow: 1,
    slidesToScroll: 1,
    autoplay: true,
    autoplaySpeed: 1000
  };

  return (
    <Slider {...settings}>
      {arrImages.map((image) => {
        return (
          <Image src={image} alt='slider' preview={false} width='100%' height='600px'>
        </Image>
        )
      })}
    </Slider>
  )
}
```

Hình 3-20. Xây dựng phần Slider

Xây dựng trang HomePage:

Tạo file HomePage.jsx để import tất cả các file Header, Footer và Slider đã tạo vào trong file này và viết import những hình ảnh sản phẩm để tạo thêm phần sản phẩm nổi bật, đồ uống, trái cây.

```
import React, { useState } from 'react';
/*import TypeProduct from '../components/TypeProduct/TypeProduct';*/
import { WrapperButtonMore, WrapperProducts, WrapperTitle } from './style';
import SlideComponent from '../components/SlideComponent/SlideComponent';
import slider1 from '../asset/images/slider_2.jpg';
import slider2 from '../asset/images/1.jpg';
import slider3 from '../asset/images/slider_5.jpg';
import slider4 from '../asset/images/tra-i-cay-giau-canxi.jpg';
import item1 from '../asset/images/item1.jpg';
import item2 from '../asset/images/featitem6.jpg';
import item3 from '../asset/images/item3.jpg';
import item4 from '../asset/images/item4.jpg';
import item5 from '../asset/images/drink4.jpg';
import item6 from '../asset/images/item12.jpg';
import Process1 from '../asset/images/say-kho-hoa-quả-2.jpg';
import Process2 from '../asset/images/traicaysaydeo (1).jpg';
import Process3 from '../asset/images/fresh-orange-juice-glass-dark-background.jpg';
import Carousel from 'react-multi-carousel';
import 'react-multi-carousel/lib/styles.css';
import CartComponent from '../components/CartComponent/CartComponent';
import { Button } from 'antd';
import CartDrinkComponent from '../components/CartDrinkComponent/CartDrinkComponent';
import CartFruitComponent from '../components/CartFruitComponent/CartFruitComponent';
import FooterComponent from '../components/FooterComponent/FooterComponent';
import { Link, useNavigate } from 'react-router-dom';
import { AiOutlineMenu } from 'react-icons/ai';
import './style.scss';

const HomePage = () => {
  const navigate = useNavigate();
  const handelHomePage = () => {
    navigate('/')
  }
  const handelContact = () => {
    navigate('/contact')
  }
  const handelProduct = () => {

```

```
const sliderItems = [
  {
    img:item1,
    name: "Mì cay"
  },
  {
    img: item2,
    name: "Dầu tẩy"
  },
  {
    img: item3,
    name: "Bánh kẹo"
  },
  {
    img: item4,
    name: "Bánh ướt"
  },
  {
    img: item5,
    name: "Trà trái cây"
  },
  {
    img: item6,
    name: "Pizza"
  }
];

// const arr = [ 'Trang chủ', 'Sản phẩm', 'Bài viết', 'Liên hệ' ]
return (
  <div>
    <div style={{display: 'flex',background:' #efefef'}}>
```

```
return (
  <div>
    <div style={{display: 'flex',background:' #efefef'}}>
      <div className="hero_categories">
        <div className="hero_categories_all" onClick={() => setShowCategories(listShowCategories)}>
          <OutlineMenu>
            Danh sách sản phẩm
          </div>
          <ul className={isShowCategories ? "" : "hidden"}>
            <li>
              <Link to =[""]>
                Thực ăn nhẹ
              </Link>
            </li>
            <li onClick={handelDrink}>
              <Link to =[""]>
                Đồ uống
              </Link>
            </li>
            <li onClick={handelJruit}>
              <Link to =[""]>
                Trái cây
              </Link>
            </li>
          </ul>
        </div>
      </div>
      <div className="list_menu" style={{padding: '0 300px', display: 'flex',paddingTop: '0px', background:' #efefef'}}>
        <ul>
          <li onClick={ handelHomePage}>
            TRANG CHỦ
          </li>
          <li onClick={handelProduct}>
            SẢN PHẨM
          </li>
          <li onClick={handelAbout}>
            BÀI VIẾT
          </li>
          <li onClick={handelContact}>
```

```
</div>
<div className=" col px-5 text-center">
  <div className="mb-3">
    <img src={Process2} className="w-50 h-50" alt="" />
  </div>
  <h5>Trái cây sấy dẻo</h5>
  <p>
    Có thể ăn hoặc tặng vào những ngày lễ tết
  </p>
</div>
<div className=" col px-5 text-center">
  <div className="mb-3">
    <img src={Process3} className="w-50 h-50" alt="" />
  </div>
  <h5>Nước ép rau củ quả</h5>
  <p>
    Phù hợp với những người không thể ăn trực tiếp rau củ quả
  </p>
</div>
</div>

  </*Categories Begin*/>
  <div className="container container_categories_slide">
    <Carousel responsive={responsive} className="categories_slide">
      {sliderItems.map((item, key) =>{
        <div className="categories_slide_items"
          style = {(backgroundImage: `url(${item.img})`)}
          key = {key}
        >
          <p> {item.name} </p>
        </div>
      )
    }
    </Carousel>
  </div>
  </*Categories end*/>
```

Hình 3-21. Xây dựng trang HomePage

3.4.2.2 Xây dựng giao diện sản phẩm

Tạo file CartComponent.jsx và viết đoạn code như hình:

```
import React from 'react'
import { StyleNameProduct, WrapperCartStyle, WrapperDiscountText, WrapperPriceText, WrapperReportText, WrapperStyleTextSell } from './style'
import {
  StarFilled
} from '@ant-design/icons';
import { formatter } from '../formatter';
import { WrapperButtonCart } from '../CartComponent/style';
import { IoCartOutline } from 'react-icons/io5';
import { WrapperProducts } from '../pages/Homepages/style';

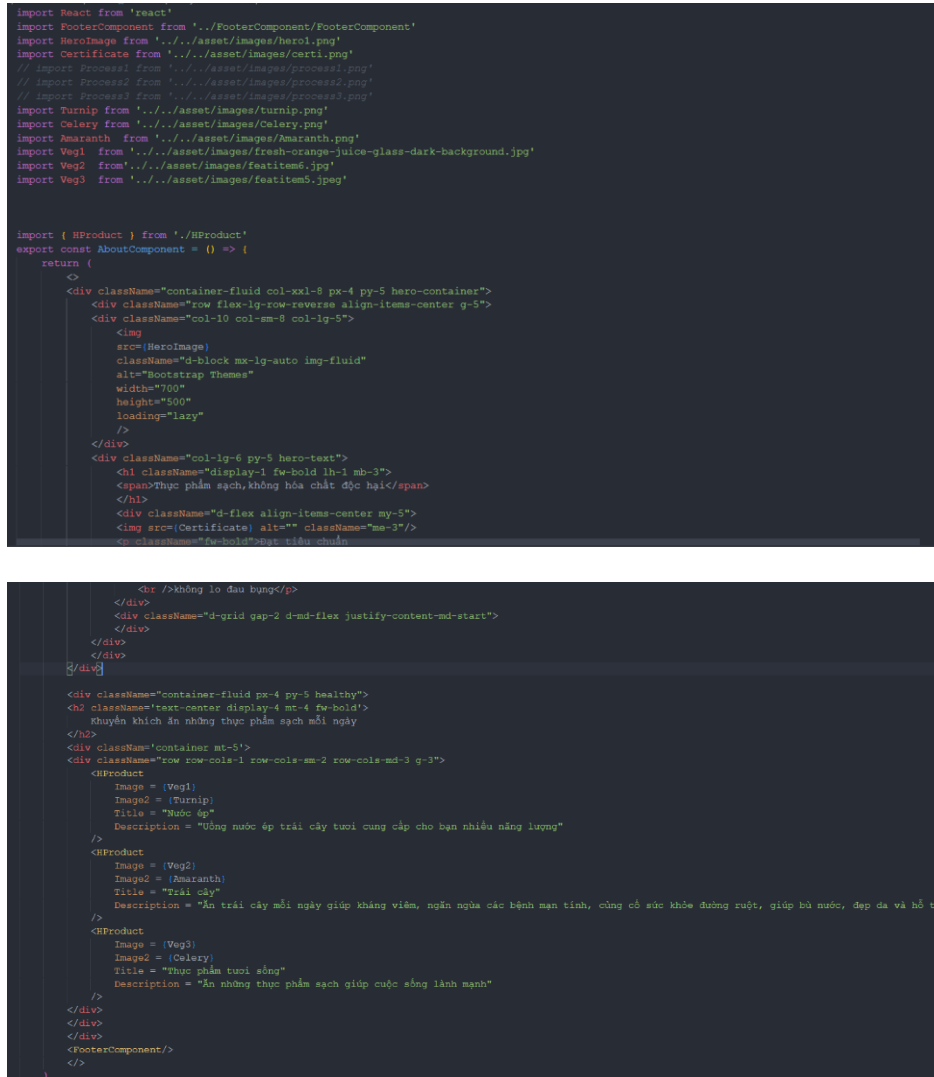
const CartDrinkComponent = () => {
  const product = [
    {
      "id": 1,
      "name": "Bà tui trần châu đường đen",
      "price": 17000,
      "buy": "Bà bán 9%",
      "sale": "7%",
      "thumbnail": require("../asset/images/drink1.jpg")
    },
    {
      "id": 2,
      "name": "Trà chanh dây",
      "buy": "Bà bán 24%",
      "price": 23000,
      "thumbnail": require("../asset/images/drink_chanh_day.jpg")
    },
    {
      "id": 3,
      "name": "Nước cam",
      "buy": "Bà bán 12%",
      "price": 15000,
      "thumbnail": require("../asset/images/drink_nuoc_cam.jpg")
    },
    {
      "id": 4,
      "name": "Bánh tổ xoài",
    }
  ]

  return (
    <WrapperProducts>
      {product.map((product) => (
        <div style={{ gap: '15px', marginTop: '10px', flexWrap: 'wrap', justifyContent: 'center', padding: '12px'}}>
          <WrapperCartStyle
            hoverable
            headStyle={{width: '200px', height: '200px'}}
            style={{width: 200, borderRadius: '5px'}}
            bodyStyle={{padding: '10px'}}
            cover=<img alt="example" src={product.thumbnail} />
          >
            <StyleNameProduct style={{fontSize: '18px', fontWeight: '600', fontFamily: 'serif', textAlign: 'center'}}>{product.name}</StyleNameProduct>
            <WrapperReportText >
              <span style={{marginRight: '4px'}}>
                <StarFilled style={{fontSize: '12px', color: 'yellow'}} />
                <WrapperStyleTextSell style={{paddingLeft: '10px'}}>{product.buy}</WrapperStyleTextSell>
              </span>
            </WrapperReportText >
            <WrapperPriceText >
              <span style={{marginRight: '8px'}}>{formatter(product.price)}</span>
            </WrapperPriceText>
            <WrapperDiscountText>
              {product.sale}
            </WrapperDiscountText>
            <WrapperButtonCart href="#" style={{textDecoration: 'none', justifyContent: 'center', display: 'flex', fontFamily: 'serif', fontWeight: '600', border: '1px solid black', padding: '5px 10px'}}>
              Thêm vào
            </WrapperButtonCart>
          </div>
        </WrapperCartStyle>
      ))}
    </WrapperProducts>
  )
}
```

Hình 3-22. Xây dựng giao diện sản phẩm

3.4.2.3 Xây dựng giao diện trang bài viết

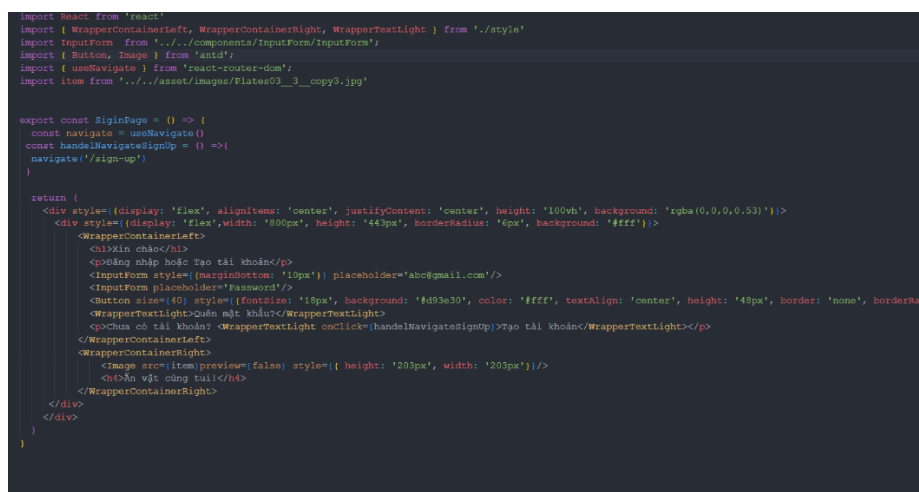
Tạo file AboutComponent.jsx và viết đoạn code sau:



Hình 3-23. Xây dựng giao diện trang bài viết

3.4.2.4 Xây dựng giao diện trang đăng nhập

Tạo file SignInPage.jsx và viết code như hình



Hình 3-24. Xây dựng giao diện trang đăng nhập

3.4.2.5 Xây dựng giao diện trang đăng ký

Tạo file SignupPage.jsx và viết code như hình

```
import React from 'react'
import { WrapperContainerLeft, WrapperContainerRight, WrapperTextLight } from '../SignInPage/style'
import InputForm from '../components/InputForm/InputForm'
import { Button, Image } from 'antd'
import itemBackground from '../asset/images/2.jpeg'

export const SignupPage = () => {
  return (
    <div style={{display: 'flex', alignItems: 'center', justifyContent: 'center', height: '100vh', background: 'rgba(0,0,0,0.53)'}}>
      <div style={{display: 'flex', width: '800px', height: '440px', borderRadius: '6px', background: '#fff'}}>
        <WrapperContainerLeft>
          <h1>Xin chào</h1>
          <p>Đăng nhập hoặc Tạo tài khoản</p>
          <InputForm style={{marginBottom: '10px'}} placeholder='abc@gmail.com' />
          <InputForm placeholder='Password' style={{marginBottom: '10px'}} />
          <InputForm placeholder='Confirm Password' />
          <Button size={40} style={{fontSize: '18px', background: '#d93e30', color: '#fff', textAlign: 'center', height: '40px', border: 'none', borderRadius: '4px'}}>Đăng ký</Button>
          <WrapperTextLight>Quên mật khẩu</WrapperTextLight>
          <p>Bạn đã có tài khoản? <WrapperTextLight>Đăng ký</WrapperTextLight></p>
        </WrapperContainerLeft>
        <WrapperContainerRight>
          <Image src={itemBackground} preview={false} style={{height: '203px', width: '203px'}} />
          <h1>Xin chào</h1>
          <h2>Xin chào</h2>
        </WrapperContainerRight>
      </div>
    </div>
  )
}
```

Hình 3-25. Xây dựng giao diện trang đăng ký

3.4.2.5 Xây dựng giao diện trang sản phẩm chi tiết

Tạo file ProductDetailComponent.jsx và viết code như hình

```
import { Col, Row, Image, Button } from 'antd'
import React from 'react'
import ImageProduct from '../asset/images/Item1.jpg'
import { WrapperStyleNameProduct, WrapperAddressProduct, WrapperInputNumber, WrapperPriceProduct, WrapperPriceTextProduct, WrapperQualityProduct, WrapperStyleTextProduct, WrapperStyleTextSell } from '../components/WrapperStyle'
import { StarFilled, PlusOutlined, MinusOutlined } from '@ant-design/icons'
import { formatter } from '../format'

export const ProductDetailComponent = () => {
  const onChange = () => {}
  return (
    <div>
      <Row style={{padding: '16px', background: '#fff', borderRadius: '4px'}}>
        <Col span={10} style={{borderRight: '1px solid #e6e6e6', paddingRight: '8px'}}>
          <Image src={ImageProduct} alt='image product' preview={false} />
        </Col>
        <Col span={14} style={{paddingLeft: '20px'}}>
          <WrapperStyleNameProduct>Mi cay Hàn Quốc 7 cấp độ</WrapperStyleNameProduct>
          <div>
            <StarFilled style={{fontSize: '12px', color: 'yellow'}} />
            <StarFilled style={{fontSize: '12px', color: 'yellow'}} />
            <StarFilled style={{fontSize: '12px', color: 'yellow'}} />
            <StarFilled style={{fontSize: '12px', color: 'yellow'}} />
            <StarFilled style={{fontSize: '12px', color: 'yellow'}} />
          </div>
          <WrapperStyleTextProduct>Đã bán 2310</WrapperStyleTextProduct>
          <WrapperPriceProduct>
            <WrapperPriceTextProduct>{formatter(45000)}</WrapperPriceTextProduct>
          </WrapperPriceProduct>
          <div style={{fontSize: '20px', fontWeight: '500'}}>
            <p>Vị chua chua, cay cay trong những món ăn khiến ai đã dùng qua đều phải  
nhưng nhớ mỗi khi nhắc đến. Đó là các món ăn đặc trưng đến từ xứ sở kim chi,  
đất nước Hàn Quốc xinh đẹp. Với vị ngon của tôm mi cay đông lạnh tươi ngon, nổi lâu hương kim chi khiến bạn như được đưa đến thiên đường ẩm  
vực còn chờ chờ gì nữa, hãy mau mau đến với Seoul Đồng An nơi vị giác được đánh thức và khơi dậy.</p>
          </div>
        </Col>
      </Row>
    </div>
  )
}
```

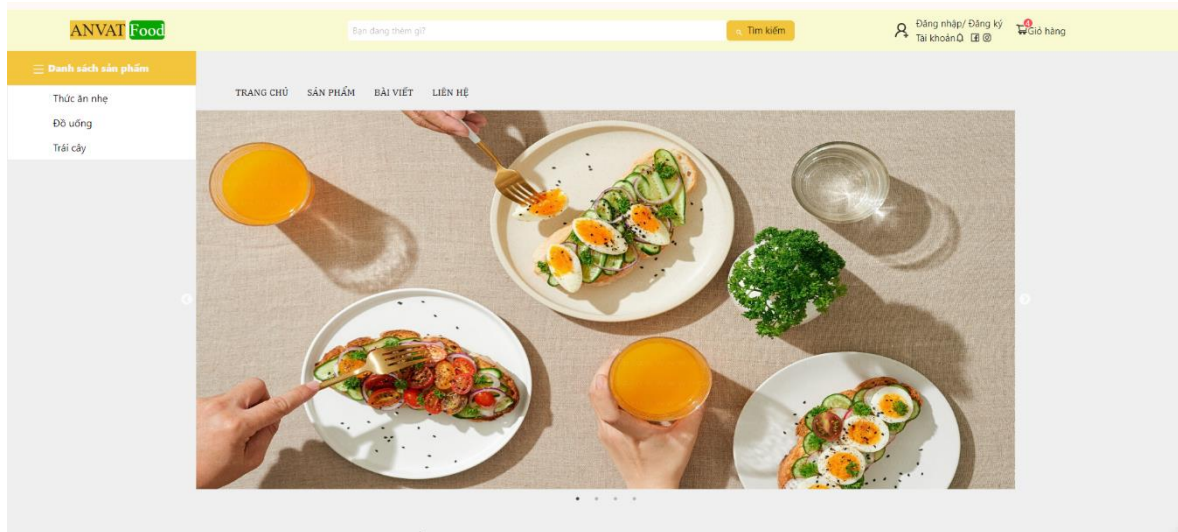


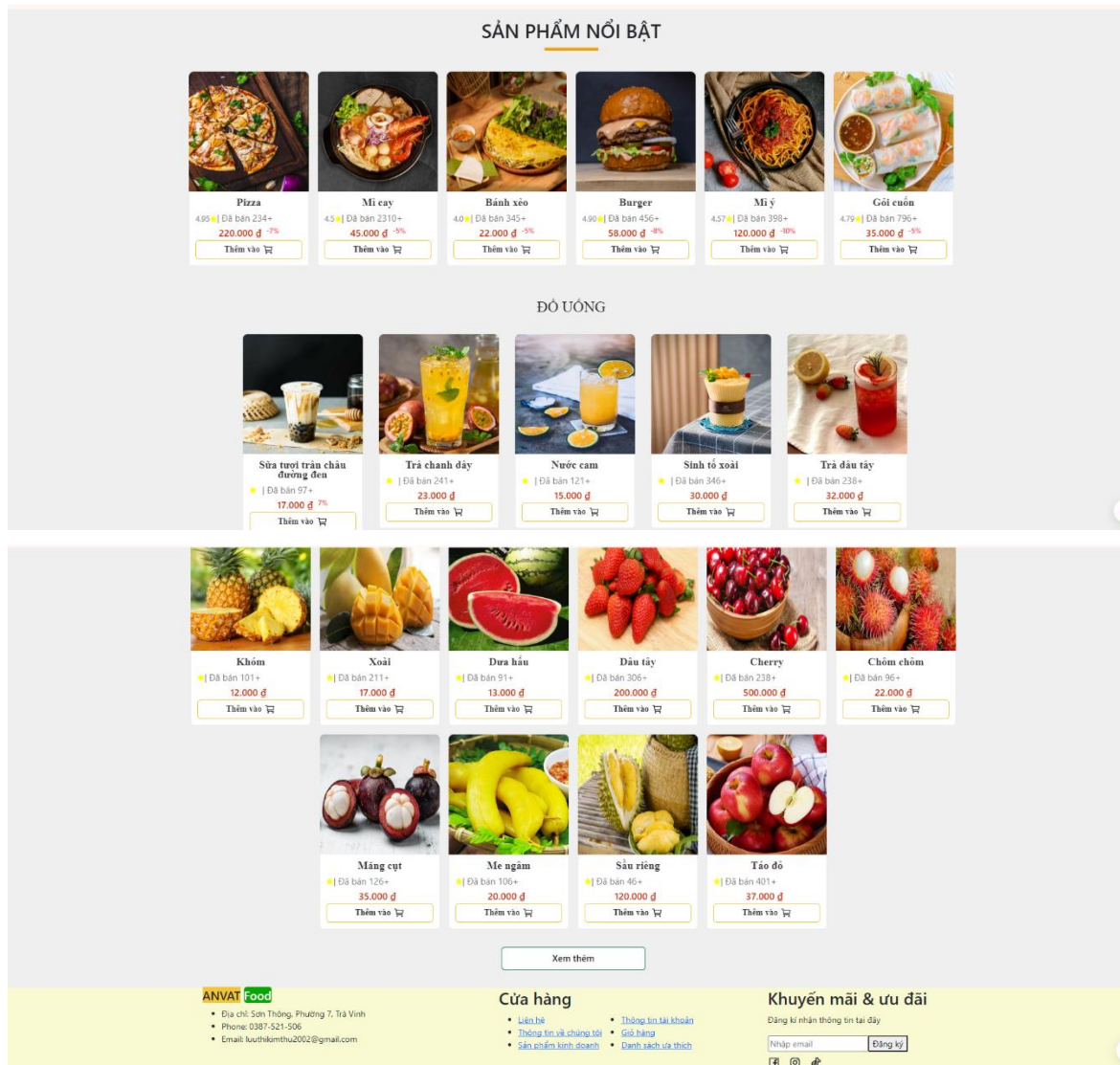
```
39     </p>
40   </div>
41   <WrapperAddressProduct>
42     <span>Giao dân </span>
43     <span className='address'>Đường Sơn Thông, xóm 1, phường 7, Thành phố Trà Vinh</span> -
44     <span className='change-address'>Đổi địa chỉ </span>
45   </WrapperAddressProduct>
46   <div style={({margin: '20px 0 40px',padding: '10px 0', borderBottom: '1px solid #e5e5e5', borderTop: '1px solid #e5e5e5'})}>
47     <div style={({marginBottom: '10px', fontSize: '18px'})}>Số lượng</div>
48     <WrapperQualityProduct>
49       <Button style={({border: 'none'})}>
50         <PlusOutlined style={({fontSize: '18px', color: '#f2b705'})} />
51       </Button>
52       <WrapperInputNumber defaultValue={3} onChange={onChange} />
53       <Button style={({border: 'none'})}>
54         <MinusOutlined style={({fontSize: '18px', color: '#f2b705'})}/>
55       </Button>
56     </WrapperQualityProduct>
57   </div>
58   <div>
59     <Button size={40} style={({fontSize: '18px', background: '#078c03', color: '#fff', textAlign: 'center', height: '48px', width: '220px', border: '1px solid #078c03'})}>Thêm vào giỏ hàng</Button>
60   </div>
61 </Col>
62 </Row>
63 </>
64 </>
65 )
66 }
67
```

Hình 3-26. Xây dựng giao diện trang sản phẩm chi tiết

CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

Giao diện trang chủ

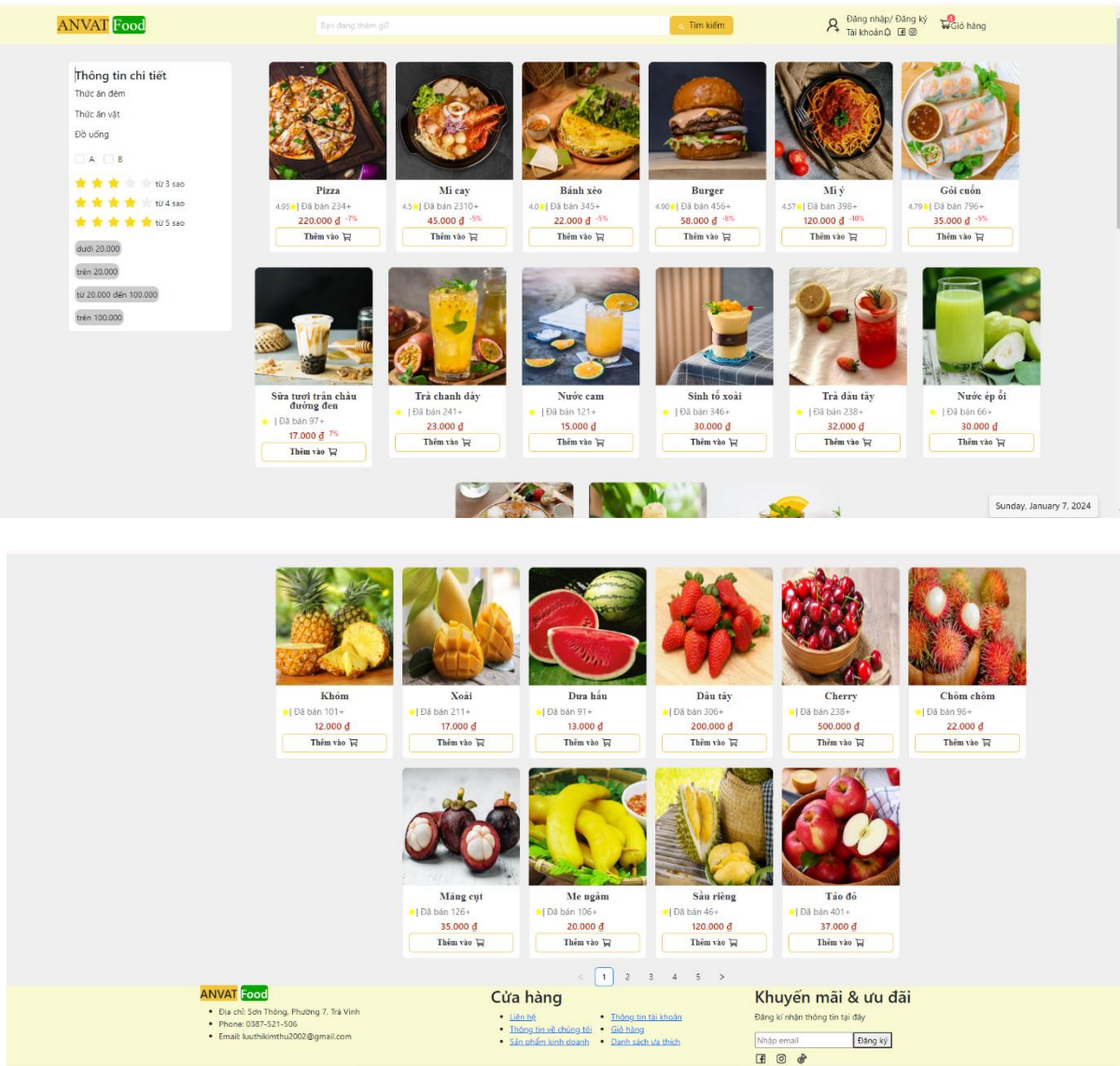




Hình 4-1. Giao diện trang chủ

Mô tả giao diện trang chủ: ở trang này thể hiện giao diện tổng quan về sản phẩm đồ ăn vặt, thanh tìm kiếm, các loại sản phẩm mới, sản phẩm nổi bật...

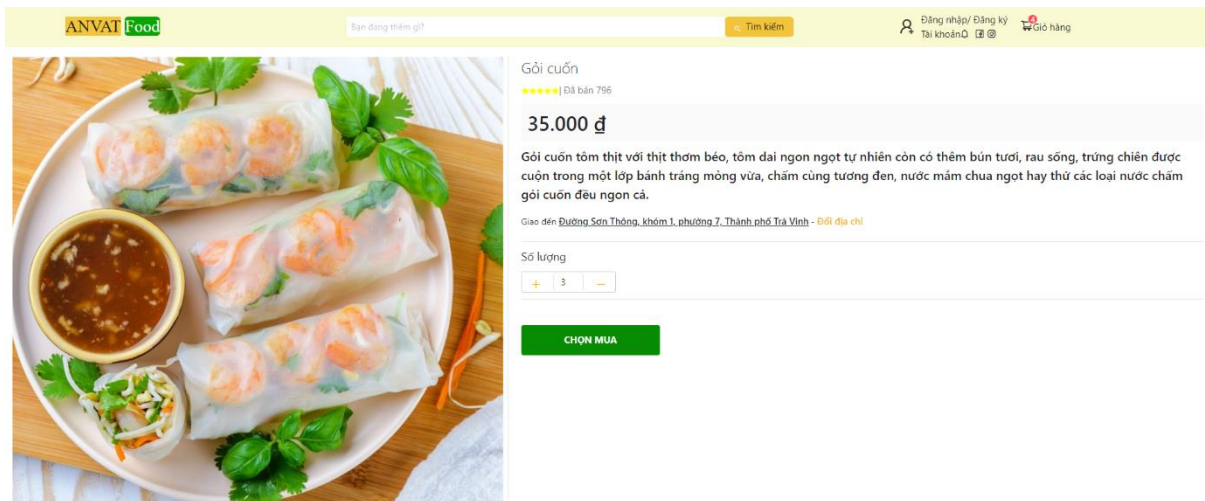
Giao diện trang sản phẩm



Hình 4-2. Giao diện sản phẩm

Mô tả giao diện trang sản phẩm: gồm nhiều loại sản phẩm như trái cây, nước uống, thức ăn nhanh, thông tin chi tiết về số sao đánh giá sản phẩm, giá tiền, loại thức ăn...

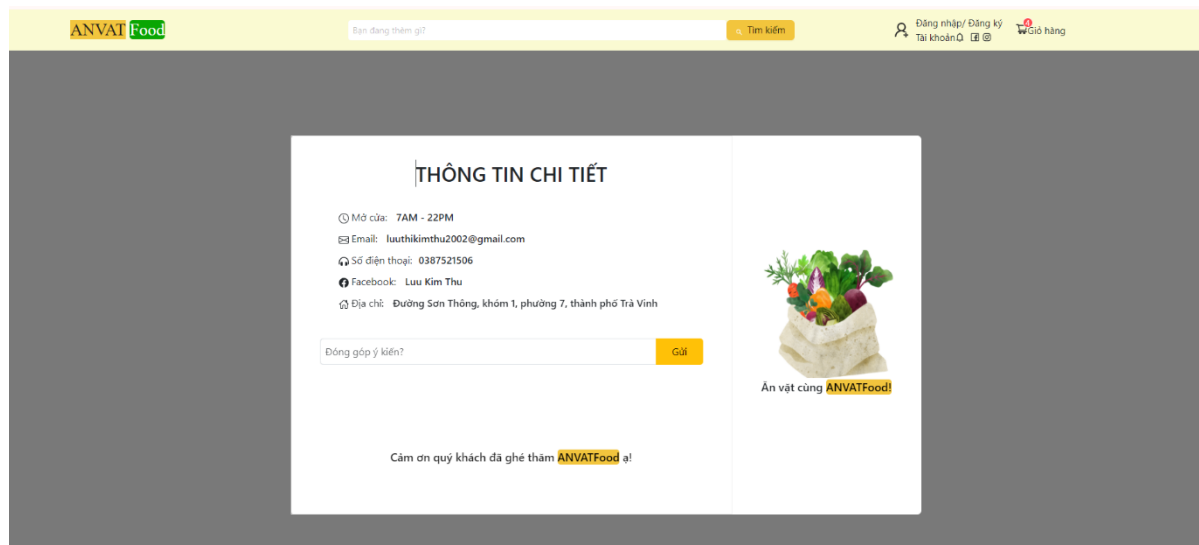
Giao diện sản phẩm chi tiết



Hình 4-3. Giao diện sản phẩm chi tiết

Mô tả giao diện trang sản phẩm chi tiết: giao diện này thể hiện tên sản phẩm, giá tiền, sao đánh giá, hình sản phẩm, địa chỉ của nơi bán, giới thiệu tổng quan về món ăn và có thể chọn mua

Giao diện trang liên hệ



Hình 4-4. Giao diện trang liên hệ

Mô tả giao diện trang liên hệ: giao diện này thể hiện những thông tin để khách hàng có thể liên hệ như email, số điện thoại, facebook, địa chỉ, giờ hoạt động và khách hàng có thể đóng góp ý kiến ngay trên form này.

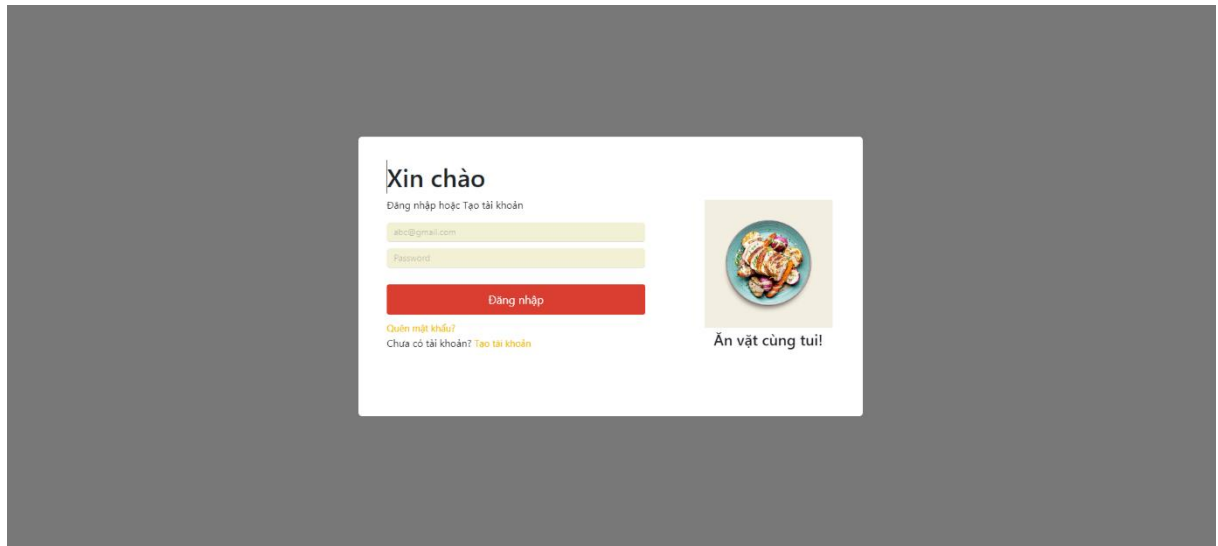
Giao diện trang bài viết



Hình 4-5. Giao diện trang bài viết

Mô tả trang bài viết: giao diện này thể hiện uy tín của các sản phẩm trên trang web, mô tả đúng nguồn gốc, xuất xứ, chất lượng và những món ăn tốt cho sức khỏe.

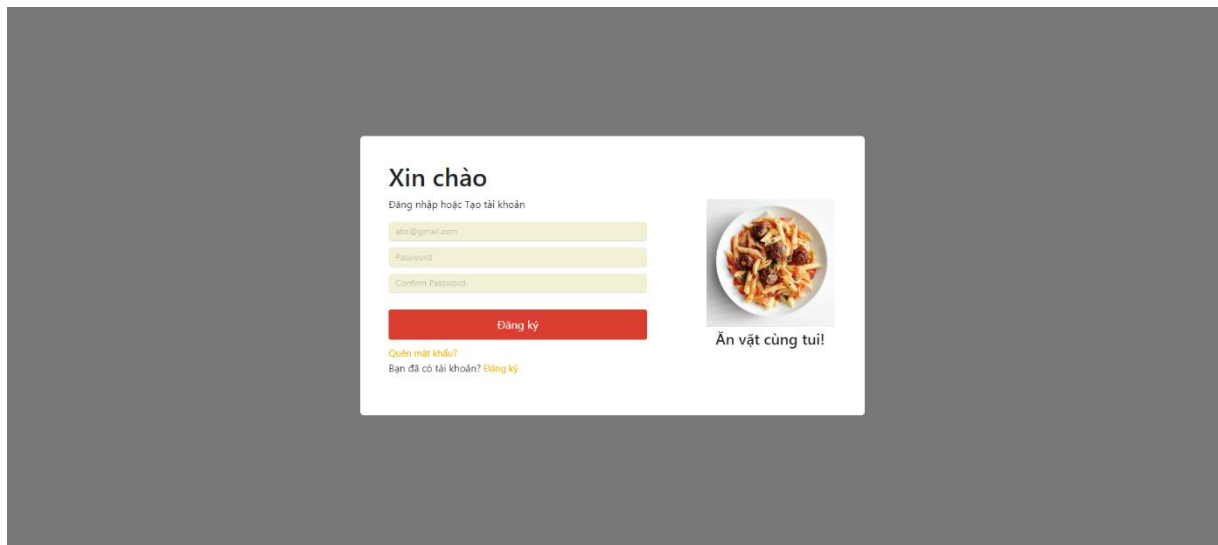
Giao diện trang đăng nhập



Hình 4-6. Giao diện trang đăng nhập

Mô tả trang đăng nhập: khách hàng có tài khoản email để đăng nhập và mua hàng

Giao diện trang đăng ký



Hình 4-7. Giao diện trang đăng ký

Mô tả trang đăng ký: khách hàng lần đầu tiên vào website thì có thể đăng ký tài khoản, sau đó mua hàng.

CHƯƠNG 5: HƯỚNG PHÁT TRIỂN

5.1 Kết luận

Về kiến thức:

- Sử dụng thành thạo ngôn ngữ Javascript để lập trình và làm việc với ReactJS Framework.
- Hiểu và vận dụng được cách sử dụng các State, React-Router-Dom, components, JSX...
- Biết cách sử dụng CSS để thiết kế bố cục giao diện đẹp mắt và hài hòa.

Về ứng dụng:

Ứng dụng thành công ReactJS thiết kế giao diện website bán đồ ăn vặt đáp ứng các nhu cầu về: màu sắc, danh sách sản phẩm, phân loại sản phẩm, trang chủ, đăng nhập, đăng ký.

Về kỹ năng:

- Cải thiện được tư duy phân tích thiết kế hệ thống.
- Kỹ năng đọc hiểu tài liệu chuyên ngành.

5.2 Hướng phát triển

Do thời gian thực hiện đề tài có hạn và kiến thức về ReactJS còn hạn chế, nên nếu có thời gian tôi sẽ xây dựng thêm một số chức năng sau:

Website tạo ra còn đơn giản cần cung cấp những tính năng và thông tin phong phú hơn.

Quá trình phân tích thiết kế hệ thống chưa hợp lý.

Tìm kiếm sản phẩm theo tên và theo giá tiền.

Thêm sản phẩm vào giỏ hàng.

Nâng cấp giao diện để dễ sử dụng.

Có thể đặt hàng online và thanh toán qua ví điện tử.

TÀI LIỆU THAM KHẢO

- [1] Đồ án cơ sở ngành Tìm hiểu Laravel Framework và ứng dụng của Thạch Minh Lực
giáo viên hướng dẫn Phạm Minh Dương
- [2] <https://vn.got-it.ai/blog/mysql-la-gi-tim-hieu-he-quan-tri-co-so-du-lieu-mysql>
- [3] <https://www.decosoftware.com/react-native/>
- [4] <https://legacy.reactjs.org/>
- [5] <https://www.tatvasoft.com/blog/reactjs-best-practices/>
- [6] <https://200lab.io/blog/reactjs-la-gi/>
- [7] <https://ant.design/>
- [8] <https://aptech.fpt.edu.vn/reactjs-la-gi.html>
- [9] <https://stringee.com/vi/blog/post/reactJS-la-gi>
- [10] <https://chat.openai.com/>