

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO
THỰC TẬP CƠ SỞ
LẬP TRÌNH VÀ XÂY DỰNG GIAO DIỆN WEBSITE
BẰNG REACTJS

Giảng viên hướng dẫn: Nguyễn Duy Phương

Sinh viên thực hiện: Vũ Hoàng Anh

Mã sinh viên: B20DCCN080

Lớp: D20CQCN08-B

SĐT: 0385994124

Hà Nội, 2023

LỜI MỞ ĐẦU

Thực hiện nhiệm vụ học tập của nhà trường trong đợt thực tập cơ sở, được sự đồng ý của thầy ***Nguyễn Duy Phương*** em đã làm một project nhỏ dựa trên kiến thức mà mình đã học. Trong quá trình thực hiện em đã được trau dồi thêm kiến thức về frontend nói chung và reactjs nói riêng.

Được sự tận tình giúp đỡ của bạn bè và nỗ lực trong học tập, tìm tòi học hỏi, em đã hoàn thành báo cáo thực tập cơ sở của mình.

Báo cáo này gồm các phần chính sau:

Phần I: Tổng quan về những kiến thức cần học

Phần II: Tổng quan về đề tài

Trong quá trình thực hiện báo cáo, tuy đã cố gắng nhưng em vẫn còn những hạn chế về thời gian tìm hiểu, kiến thức cũng như là kinh nghiệm và vẫn còn nhiều sai sót. Em rất mong được nhận những ý kiến đóng góp và nhận xét của thầy để em có thể hoàn thiện hơn.

LỜI CẢM ƠN

Để hoàn thành được bài thuyết trình này em xin gửi lời cảm ơn tới ban lãnh đạo Học viện, Khoa đào tạo đã thiết lập khung chương trình có môn học “Thực tập cơ sở” để em có thể vận dụng những kiến thức học được trên giảng đường và những kiến thức ngoài áp dụng vào công việc và được tìm hiểu rõ về chuyên ngành mình chọn sắp tới.

Em cũng xin gửi lời cảm ơn đến những người bạn của em, luôn sẵn sàng giúp đỡ, giải đáp thắc mắc khi em gặp rắc rối trong công việc của mình.

Em xin chân thành cảm ơn!

Hà Nội, ngày 15 tháng 05 năm 2023

MỤC LỤC

LỜI MỞ ĐẦU	2
LỜI CẢM ƠN	3
MỤC LỤC	4
DANH MỤC CÁC TỪ VIẾT TẮT	6
PHẦN I: TỔNG QUAN VỀ NHỮNG KIẾN THỨC CẦN HỌC	7
CHƯƠNG I: LỘ TRÌNH HỌC	7
CHƯƠNG II: TỔNG QUAN VỀ HTML – CSS - JS – SCSS –BOOTSTRAP	8
CHƯƠNG III: TỔNG QUAN VỀ REACTJS	8
1. Tổng quan về ReactJS	8
2. Một số khái niệm kiến thức khi nghiên cứu reactJS.....	9
3. Kết luận về ReactJS	10
PHẦN 2: TỔNG QUAN VỀ ĐỀ TÀI.....	12
CHƯƠNG I: TỔNG QUAN ĐỀ TÀI	12
1. Giới thiệu	12
2. Mục tiêu đề tài.....	12
CHƯƠNG II: Về website BOMSTORE	13
1. Quá trình khởi tạo	13
2. Giao diện website bán hàng	16
2.1. Giao diện trang chủ	16
2.2. Giao diện trang sản phẩm theo dạng lưới.....	18
2.3. Giao diện detail.....	19
2.4. Giao diện login & register	19
2.5. Giao diện trang contact	20
2.6. Giao diện trang blog	20
2.7. Giao diện trang about	22
PHẦN 3: TỔNG KẾT.....	23
CHƯƠNG I: Tài liệu tham khảo	23
1. Themeforest.net.....	23
2. React.dev.....	23
3. React Router	23
4. Stackoverflow	23

5. Codesandbox.io	23
--------------------------------	-----------

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Viết đầy đủ	Giải thích nghĩa
MVC	Model-View-Controller	Đây là mô hình thiết kế được sử dụng trong kỹ thuật phần mềm
HTML	HyperText Markup Language	Ngôn ngữ Đánh dấu Siêu văn bản
JS	Javascript	Sử dụng để tạo trang web tương tác
JSX	Javascript + XML	Đây là phần mở rộng của cú pháp js cung cấp cách cấu trúc kết xuất thành phần bằng cách sử dụng cú pháp tương tự như HTML
UI	User Interface	Giao diện người dùng
UX	User Experience	Trải nghiệm người dùng

PHẦN I: TỔNG QUAN VỀ NHỮNG KIẾN THỨC CẦN HỌC

CHƯƠNG I: LỘ TRÌNH HỌC

Quá trình học tập trong vòng gần 3 tháng để giúp hoàn thiện chủ đề “Thiết kế giao diện website bán hàng sử dụng reactJS”

1. HTML / BEM / SEO (2 tuần đầu)
 - HTML cơ bản
 - HTML thông dụng
 - Đặt tên chuẩn BEM
 - Viết HTML chuẩn SEO
 - Thực hành
2. CSS và CSS3 (2 tuần đầu)
 - Cơ bản
 - Flexbox, grid
 - CSS nâng cao
 - Thực hành
3. SCSS (3 tuần đầu)
 - SASS/SCSS là gì?
 - So sánh và tìm hiểu ưu nhược điểm
 - Các tính năng của SCSS
 - Thực hành
4. Responsive (3 tuần đầu)
 - Tổng quan
 - Media queries, grid
 - Bootstrap
 - Thực hành
5. Js (3 tuần đầu)
 - Cơ bản
 - Các tính năng của js
 - Thực hành
6. Framework/ library JS (2 tháng)
 - ReactJS
 - * React cơ bản, React DOM
 - * JSX, Components, Props
 - * Create React App
 - * Hooks
 - * CSS, SCSS module
 - * React Router v6
 - * ReactSlick

CHƯƠNG II: TỔNG QUAN VỀ HTML – CSS - JS – SCSS –BOOTSTRAP

Ngôn ngữ đánh dấu siêu văn bản (HTML) và Ngôn ngữ định kiểu theo tầng (CSS) là hai ngôn ngữ lập trình khác mà các nhà phát triển sử dụng trong việc phát triển frontend. HTML là khối dựng cơ bản của hầu hết các trang web. Tất cả các đoạn văn, phần mục, hình ảnh, tiêu đề và văn bản đều được viết bằng HTML. Nội dung sẽ xuất hiện trên trang web theo thứ tự được viết bằng HTML.

CSS là một ngôn ngữ của các quy tắc định kiểu mà sử dụng để định kiểu cho nội dung của HTML, có thể sử dụng CSS để thiết kế các phần tử của trang web như màu nền, phông chữ, cột và đường viền.

HTML được biết đến là một ngôn ngữ dùng để mô tả các trang web ở trạng thái tĩnh. Không những vậy HTML còn đem đến cho người dùng những trải nghiệm về tốc độ truy cập web nhanh hơn, tốt hơn, tài nguyên phong phú hơn.

Javascript (JS) là một ngôn ngữ lập trình động phía client phổ biến được sử dụng để tạo ra các ứng dụng web tương tác động. Với js, ta có thể thực hiện các thao tác, tương tác và thay đổi nội dung trang web mà không cần tải lại trang. Đây là một trong những ngôn ngữ quan trọng và phổ biến nhất trong lĩnh vực phát triển web.

SCSS là một chương trình tiền xử lý CSS (*CSS preprocessor*). Nó giúp bạn viết CSS theo cách của một ngôn ngữ lập trình, có cấu trúc rõ ràng, rành mạch, dễ phát triển và bảo trì code hơn. Ngoài ra nó có rất nhiều các thư viện hỗ trợ kèm theo giúp bạn viết code CSS một cách dễ dàng vào đơn giản hơn.

BOOTSTRAP là một framework front-end phổ biến được sử dụng để xây dựng giao diện web đẹp và đáp ứng. Nó được tạo ra nhằm giúp nhà phát triển web dễ dàng và nhanh chóng xây dựng các trang web chuyên nghiệp, hiện đại và tương thích với nhiều thiết bị khác nhau. Bootstrap mang đến một bộ công cụ linh hoạt và mạnh mẽ, cung cấp một loạt các thành phần, lớp CSS và các tiện ích Js để tạo ra một giao diện đồng nhất, hấp dẫn và dễ dùng. Một điểm mạnh của bootstrap là khả năng tùy chỉnh cao. Bạn có thể lựa chọn các thành phần, kiểu dáng và màu sắc khác nhau để phù hợp ý tưởng thiết kế của mình.

CHƯƠNG III: TỔNG QUAN VỀ REACTJS

1. Tổng quan về ReactJS

React.js là một thư viện Javascript đang nổi lên trong những năm gần đây với xu hướng Single Page Application. Trong khi những framework khác đang cố gắng hướng đến mô hình MVC hoàn thiện thì React nổi bật với sự đơn giản và dễ dàng phối hợp với những thư viện Javascript khác. Nếu như AngularJS là một framework cho phép nhúng code js trong code HTML thông qua các attribute như ng-model, ng-repeat... thì với react là một library cho phép nhúng code HTML trong code js nhờ vào JSX, ta có thể dễ dàng lồng các đoạn HTML vào trong JS. Tích hợp giữa js và HTML vào JSX làm cho các component dễ hiểu và dễ thực hiện chức năng của nó hơn.

React là một thư viện UI được FaceBook phát triển để hỗ trợ việc xây dựng

những thành phần components UI có tính tương tác cao, có trạng thái và có thể sử dụng lại được. React đã được sử dụng tại Facebook trong production, và instagram được viết hoàn toàn trên React.

Có thể nói ReactJS là một công cụ tuyệt vời giúp doanh nghiệp vượt qua đối thủ khi cạnh tranh về mảng thiết kế các ứng dụng web.

- Ứng dụng được tạo ra bởi ReactJS với cấu trúc UI tốt hơn giúp nâng cao và tối ưu hóa UX. Các doanh nghiệp có thể dễ dàng tăng tương tác của người dùng, tỉ lệ click và tỉ lệ chuyển đổi (conversion) trên trang web của mình.
- Hiệu suất kinh doanh khi có dùng ứng dụng của ReactJS cũng tốt hơn so với khi dùng các framework khác. Nó sẽ hạn chế cập nhật của DOM, từ đó tăng tốc độ truy cập ứng dụng và cải tiến UX tốt hơn.
- ReactJS cũng được thiết kế để giúp cải thiện số trang render tổng từ phía server. Đồng thời sẽ giúp tối ưu các nút để render phía client-side. Khả năng tùy chỉnh công cụ tốt, thậm chí làm giảm ngân sách bảo trì hệ thống, giúp ReactJS hiệu quả cao hơn.

Sau một khoảng thời gian tìm hiểu và học tập, do mong muốn phát triển kỹ năng, khả năng sáng tạo của bản thân và công dụng cao của ReactJS nên em đã quyết định lựa chọn chủ đề thực tập cơ sở là “Xây dựng giao diện website bằng ReactJS”. Cũng như mong muốn tìm cho mình một hướng đi về sau cho chuyên ngành và công việc mình chọn lựa.

2. Một số khái niệm kiến thức khi nghiên cứu reactJS

2.1. React DOM

React DOM là một phần của React, nó được sử dụng để tương tác với DOM (document object model) trong ứng dụng web. React DOM cung cấp các phương thức để hiển thị và cập nhật các thành phần react lên trình duyệt

2.2. JSX

JSX là một cú pháp mở rộng cho js, là kết hợp của js và XML. Nó chuyển đổi cú pháp dạng gần như XML về thành js, giúp chúng ta có thể code ReactJS bằng cú pháp XML thay vì phải dùng js.

Thay vì phải truy xuất một thẻ HTML bất kì thông qua các thuộc tính như id, class,... thì ta có thể thoải mái sử dụng các thẻ HTML trong code Js.

Ví dụ:

```
const name = 'JoshPerez';  
const element = <h1>Hello,{name}</h1>;  
ReactDOM.render(  
  Element,  
  document.getElementById('root')  );
```

2.3. *Component*

Component là nơi render ra các mã HTML thuần, một điều quan trọng khác là nó cho phép ta chia nhỏ đoạn code của UI thành những phần độc lập với nhau, mục đích để tiện cho việc quản lý và tái sử dụng. Vậy làm cách nào để có thể gọi đến một Component? Câu trả lời là trong React tên của Component sẽ được dùng để làm tên thẻ và cách sử dụng tương tự các thẻ tag trong HTML.

2.4. *Props*

Props là viết tắt của Properties, ta có thể hiểu nó như là những thuộc tính của một Component. Nếu coi Component như là đối tượng trong lập trình OOP thì Props giống như là những miêu tả về tính chất, đặc trưng của 1 đối tượng. Để định nghĩa một Props thì ta khai báo chúng ở bên trong dấu `{}` của Component.

2.5. *State*

State được dùng để biểu diễn trạng thái của Component. Tức là giá trị của state có thể thay đổi được, trái ngược hoàn toàn với Props là một giá trị bất biến.

2.6. *Hooks*

Hooks cho phép chúng ta sử dụng các tính năng React khác nhau từ các component. Chúng ta có thể sử dụng hooks có sẵn hoặc kết hợp chúng để tạo ra hooks của riêng.

2.7. *React Router v6*

React Router là một thư viện điều hướng phổ biến được sử dụng trong ứng dụng react để quản lý định tuyến và định hướng giữa các trang.

“Các ứng dụng single-page” trở nên khá rầm rộ ngày nay. Đây là những trang web mà chỉ thực hiện khởi tạo một lần, và khi người dùng ấn vào một đường dẫn hay nút ấn thì Javascript trên trang web sẽ thực thi mà trang web không cần phải reload lại. Việc quản lý địa chỉ trên thanh địa chỉ được thực hiện bởi router.

3. Kết luận về ReactJS

Ưu điểm:

- + ReactJS cực kì hiệu quả: Reactjs tạo ra cho chính nó DOM ảo – nơi mà các component thực sự tồn tại trên đó. Điều này sẽ giúp cải thiện hiệu suất rất nhiều. ReactJS cũng tính toán những thay đổi nào cần cập nhật lên DOM và chỉ thực hiện chúng. Điều này giúp reactJS tránh những thao tác cần trên DOM mà nhiều chi phí.
- + ReactJS giúp việc viết các đoạn code JS dễ dàng hơn: Nó dùng cú pháp đặc biệt

là JSX cho phép ta trộn lẫn giữa code HTML và JS. Ta có thể thêm vào các đoạn HTML vào trong các hàm render mà không cần phải nối chuỗi. Đây là đặc tính thú vị của ReactJS. Nó sẽ chuyển đổi các đoạn HTML thành các hàm khởi tạo đối tượng HTML bằng bộ biến đổi JSX.

- + React có nhiều công cụ phát triển

- + Render tầng server: Một trong những vấn đề với các ứng dụng đơn trang là tối ưu SEO và thời gian tải trang. Nếu tất cả việc xây dựng và hiển thị trang đều thực hiện ở client, thì người dùng sẽ phải chờ cho trang được khởi tạo và hiển thị lên. Điều này thực tế là chậm. Hoặc nếu giả sử người dùng vô hiệu hóa Javascript thì sao? ReactJS là một thư viện component, nó có thể vừa render ở ngoài trình duyệt sử dụng DOM và cũng có thể render bằng các chuỗi HTML mà server trả về.

- + Làm việc với vấn đề test giao diện: Nó cực kỳ dễ để viết các test case giao diện vì virtual DOM được cài đặt hoàn toàn bằng JS.

- + Hiệu năng cao đối với các ứng dụng có dữ liệu thay đổi liên tục, dễ dàng cho bảo trì và sửa lỗi.

Nhược điểm:

ReactJS chỉ phục vụ cho tầng View. React chỉ là View Library nó không phải là một MVC framework như những framework khác. Đây chỉ là thư viện của Facebook giúp render ra phần view. Vì thế React sẽ không có phần Model và Controller, mà phải kết hợp với các thư viện khác. React cũng sẽ không có 2-way binding hay là Ajax. Tích hợp ReactJS vào các framework MVC truyền thống yêu cầu cần phải cấu hình lại.

PHẦN 2: TỔNG QUAN VỀ ĐỀ TÀI

CHƯƠNG I: TỔNG QUAN ĐỀ TÀI

1. Giới thiệu

Trên thế giới hiện nay, công nghệ thông tin là một ngành vô cùng phát triển. Chúng đang được áp dụng và phát triển rộng rãi đặc biệt là những lĩnh vực kinh doanh, buôn bán.

Hiện nay, nhu cầu buôn bán, trao đổi hàng hóa ngày càng tăng và hình thức mua và bán cũng ngày càng được cải thiện. Ở trên môi trường người mua và người bán không cần trực tiếp gặp nhau nhưng vẫn có thể thực hiện được giao dịch thì đòi hỏi phải có một môi trường ảo hỗ trợ và những website thương mại điện tử chính là môi trường tốt nhất. Chính vì vậy mà em đã chọn nghiên cứu, tìm hiểu về một thư viện thiết kế phổ biến nhất hiện nay là reactJS và đồng thời xây dựng một website bán hàng. Đây là một hệ thống lớn, và cần một chặng đường dài nên trong quá trình thực hiện đề tài này chắc chắn em sẽ mắc phải những thiếu sót nên rất mong nhận được sự góp ý của thầy, để em có cơ hội sửa sai, phát triển trong thực tế và từng bước hoàn thiện mình.

2. Mục tiêu đề tài

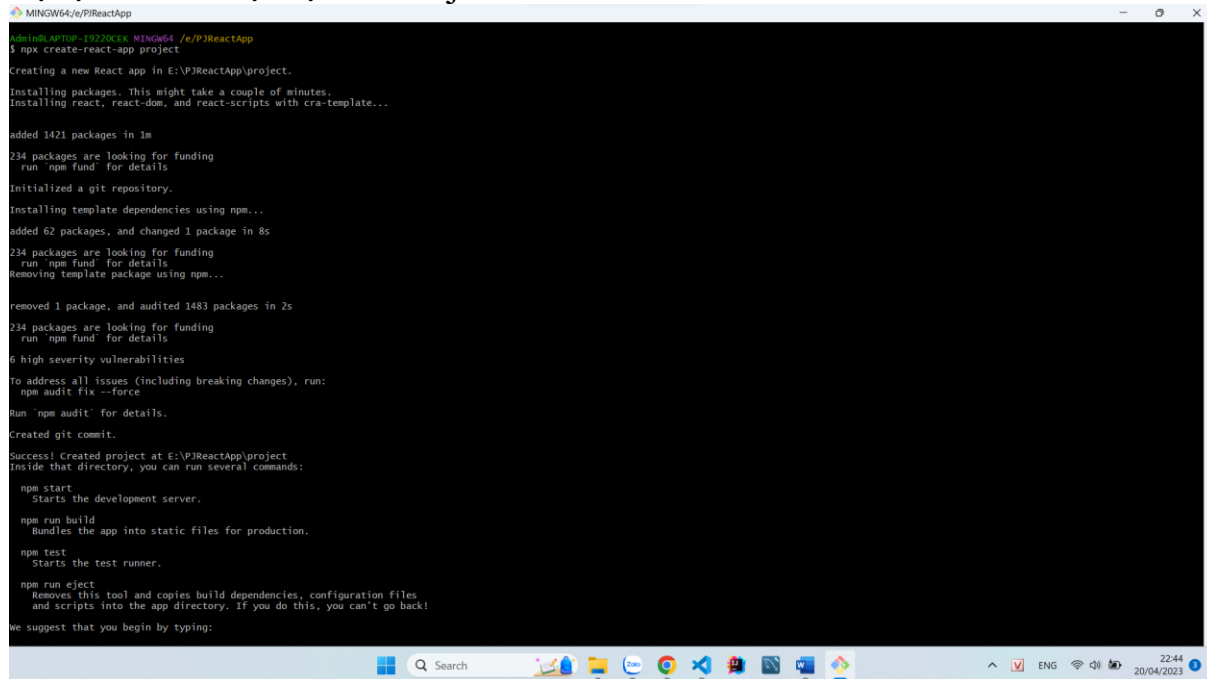
Bước đầu lập trình và xây dựng giao diện, chức năng cơ bản của website bán hàng bằng reactjs tích hợp sử dụng bootstrap. Website có khả năng tương thích, hiển thị được trên tất cả các thiết bị hiện đại và có thể nâng cấp trong tương lai.

Website được hiển thị một cách sinh động, đẹp mắt mà không quá phức tạp màu mè để cho tất cả mọi người đều sử dụng được.

CHƯƠNG II: Về website BOMSTORE

1. Quá trình khởi tạo

Sử dụng cú pháp “`npx create-react-app bomstore`” : đây là bộ thư viện có sẵn hỗ trợ tạo nhanh một dự án reactjs



```
Admin@LAPTOP-192Z0C1X MINGW64 /e/PJReactApp
$ npx create-react-app project
Creating a new React app in E:\PJReactApp\project.
Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...

added 1421 packages in 1m
234 packages are looking for funding
  run `npm fund` for details
Initialized a git repository.
Installing template dependencies using npm...
added 62 packages, and changed 1 package in 8s
234 packages are looking for funding
  run `npm fund` for details
Removing template package using npm...

removed 1 package, and audited 1483 packages in 2s
234 packages are looking for funding
  run `npm fund` for details
6 high severity vulnerabilities
To address all issues (including breaking changes), run:
  npm audit fix --force
Run `npm audit` for details.
Created git commit.
Success! Created project at E:\PJReactApp\project
Inside that directory, you can run several commands:

  npm start
    Starts the development server.

  npm run build
    Bundles the app into static files for production.

  npm test
    Starts the test runner.

  npm run eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:
```

Bên cạnh đó em còn tải một số thư viện hỗ trợ việc code như : **bootstrap** , **bootstrap-icon**, **react-router-dom**...v..v Tất cả thư viện đã cài đặt được lưu trong “**package.json**”

Sau khi cài đặt xong các thư viện cần dùng, sẽ phân chia bóc tách website thành nhiều phần và tạo các component.

Tiếp theo em sẽ defaultLayout để định dạng website của mình bao gồm 3 phần: header , outlet, footer. Khi làm vậy thì ta sẽ giữ được phần header và footer duy trì sử dụng chung ở tất cả các trang và nội dung chỉ thay đổi trong outlet.

```
import React from "react";
import {Outlet} from "react-router-dom";
import Header from "../components/header/header";
import Footer from "../components/footer/footer";
import "../assets/css/layout.css";
const Layout = () => {
```

```

return (
  <>
    <div className="page">
      <div className="container">
        <div className="outer-row">
          <div className="col-md-12">
            <Header />
            <Outlet />
            <Footer />
          </div>
        </div>
      </div>
    </div>
  </div>
</>
);
};
export default Layout;

```

Cấu hình router

```

import React from "react";
import "./assets/css/app.css";
import { BrowserRouter, Routes, Route } from 'react-router-dom';
import Layout from "./layout/defaultLayout/layout";
import Home from "./pages/home/home";
import About from "./pages/about/about";
import Contact from "./pages/contact/contact";
import Login_register from './pages/login_register/login_register';
import Blog from './pages/blog/blog';
function App() {
  return(
    <>

      <BrowserRouter>
        <Routes>
          <Route path="/" element={<Layout/>}>
            <Route index element = {<Home/>} />
            <Route path="/Login_register" element={<Login_register/>} />
            <Route path="/Contact" element={<Contact/>} />
            <Route path="/About" element = {<About />} />

```

```

    <Route path="/Blog" element = {<Blog />} />

  </Route>

</Routes>
</BrowserRouter>
</>
);
}
export default App;

```

Sử dụng `React.StrictMode` để kiểm tra lỗi khi phát triển website

```

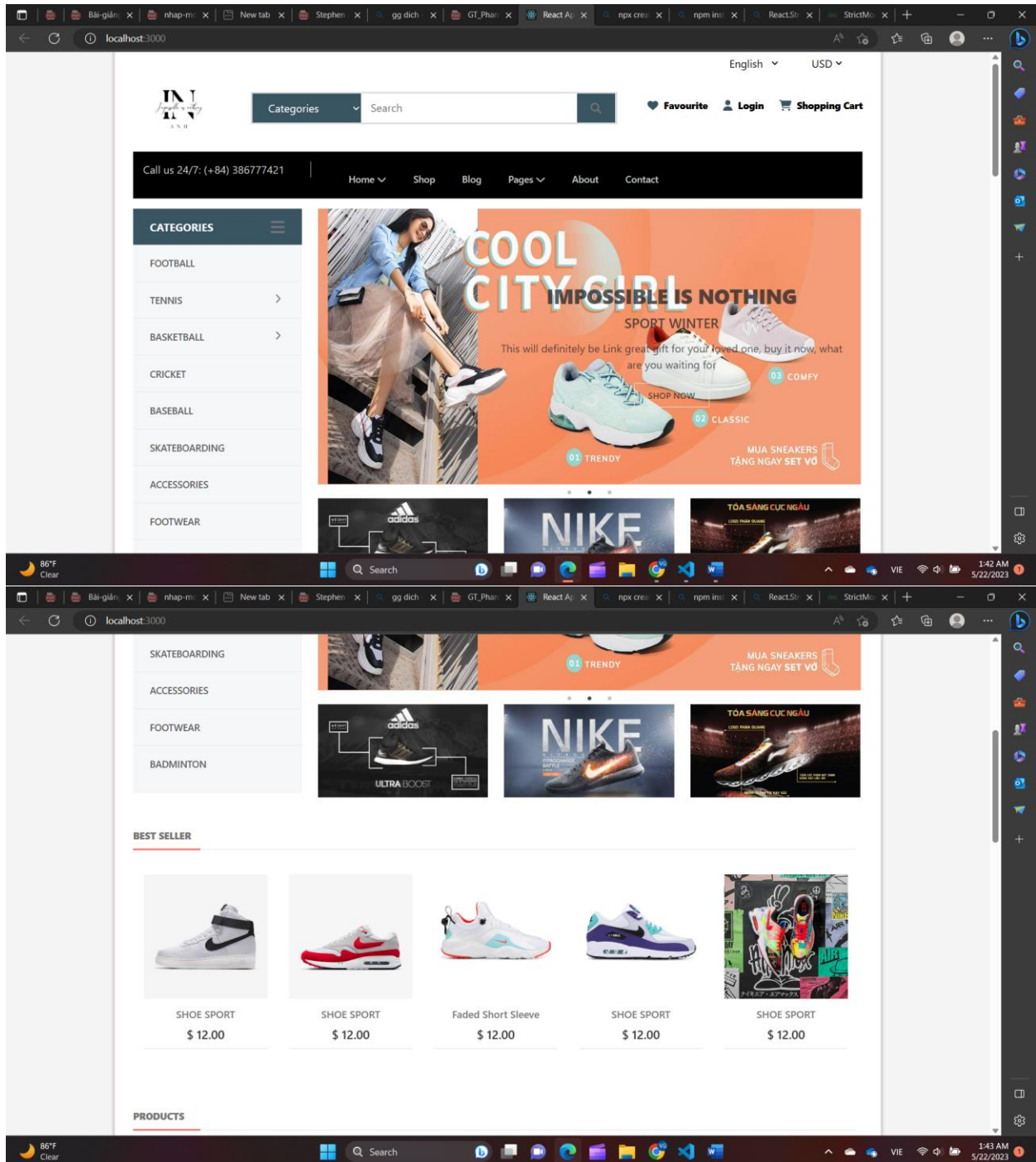
import React from "react";
import ReactDOM from "react-dom";
import App from "./App";
import 'bootstrap/dist/css/bootstrap.min.css';

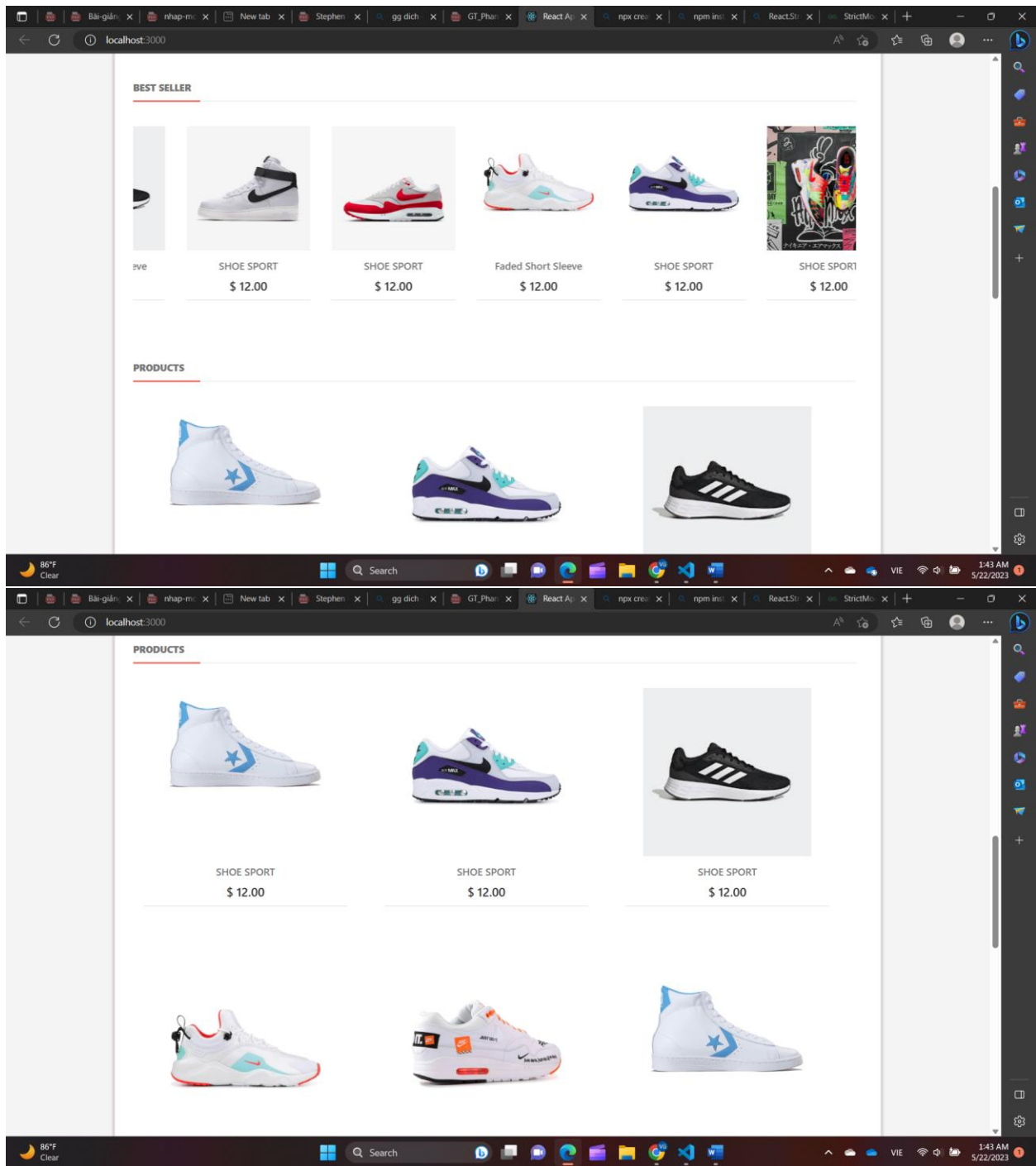
ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById('root')
);

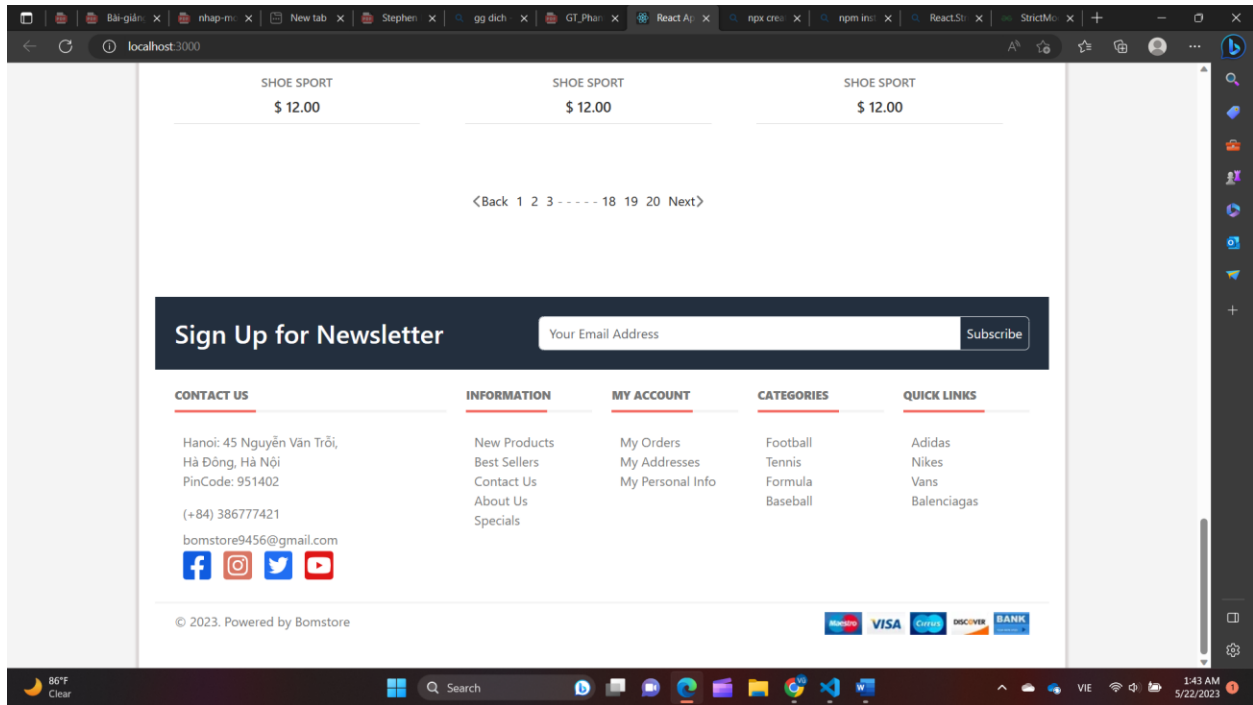
```

2. Giao diện website bán hàng

2.1. Giao diện trang chủ

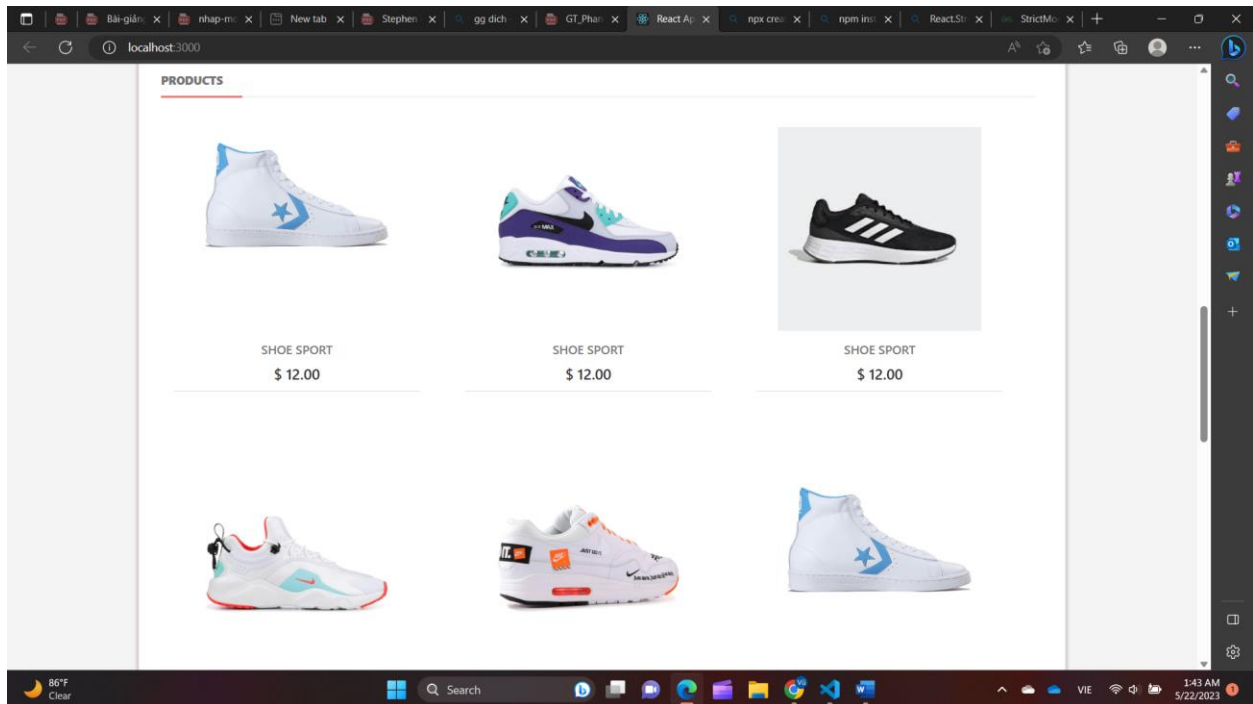






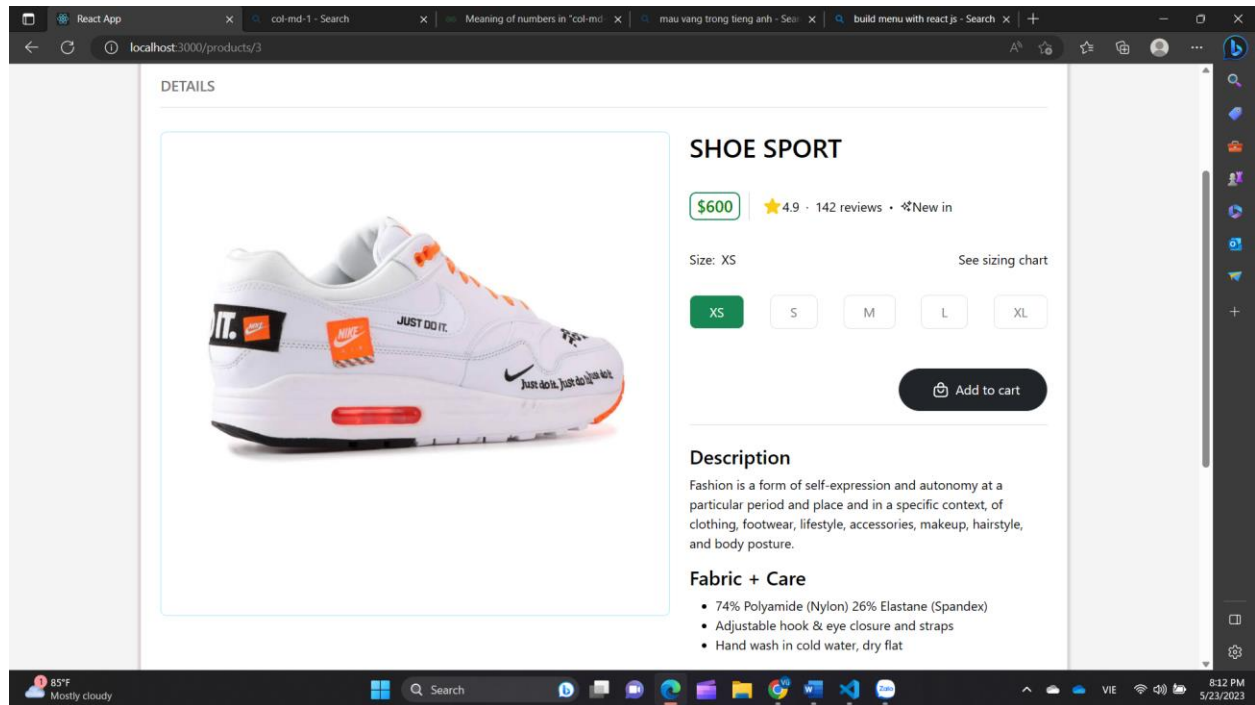
2.2. Giao diện trang sản phẩm theo dạng lưới

Tất cả sản phẩm sẽ được lưu trong file productList.js , sau đó sẽ được đổ ra product.js thông qua key là 'id'.

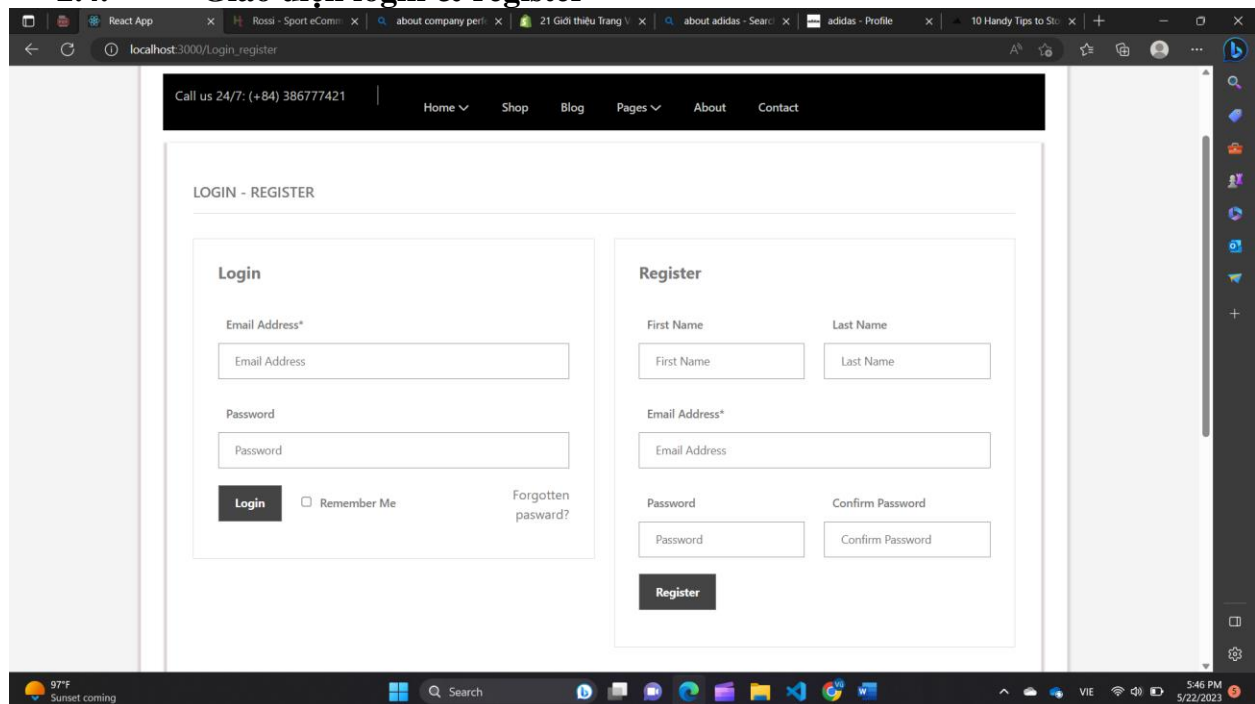


2.3. Giao diện detail

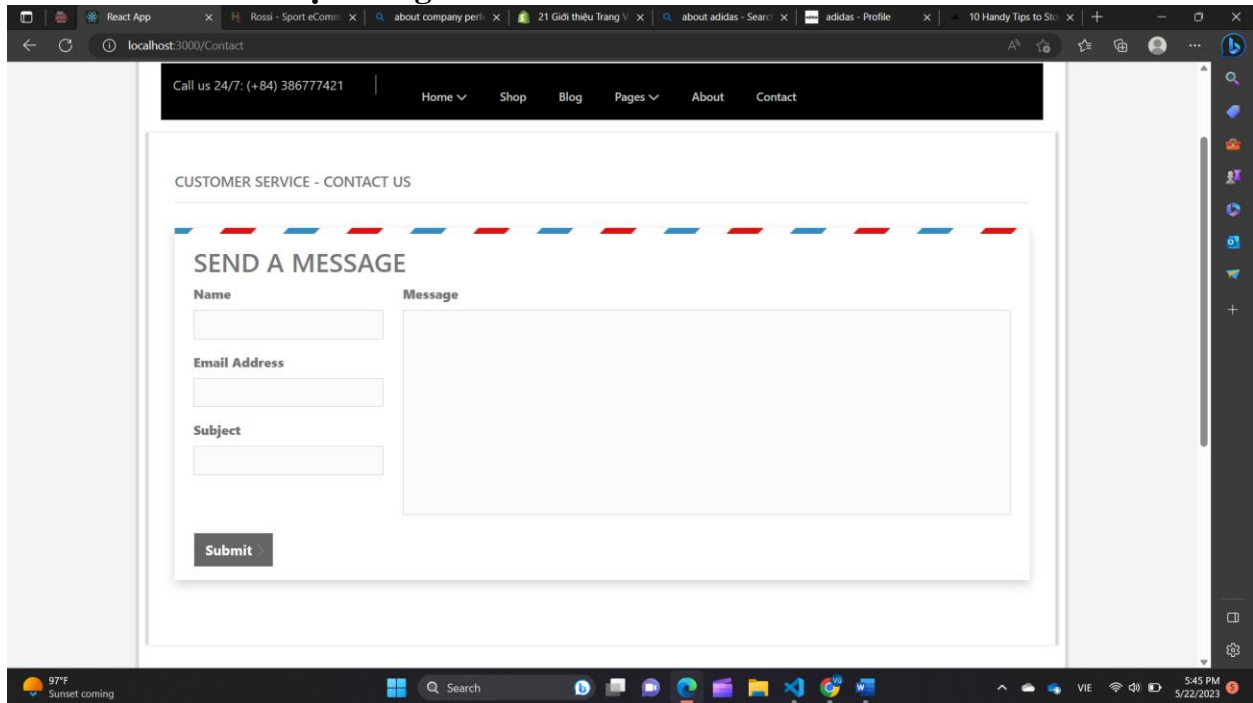
Dựa vào khóa chính của product để lấy ra detail các sản phẩm



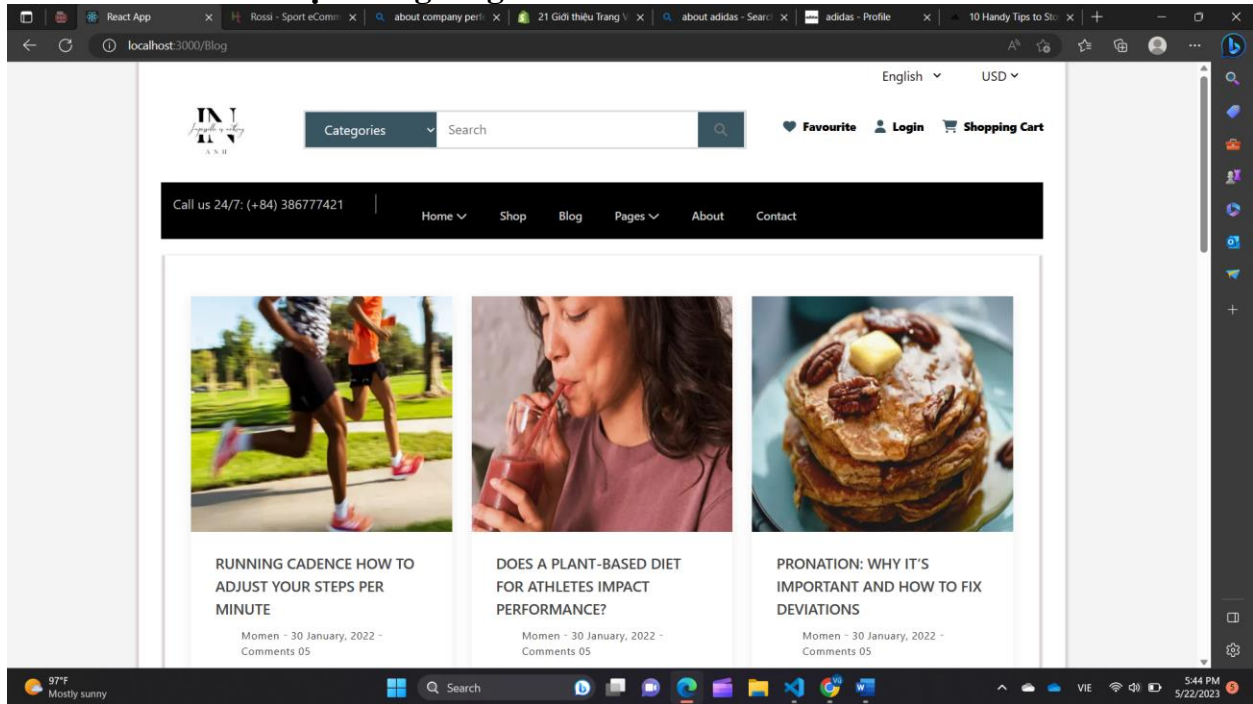
2.4. Giao diện login & register

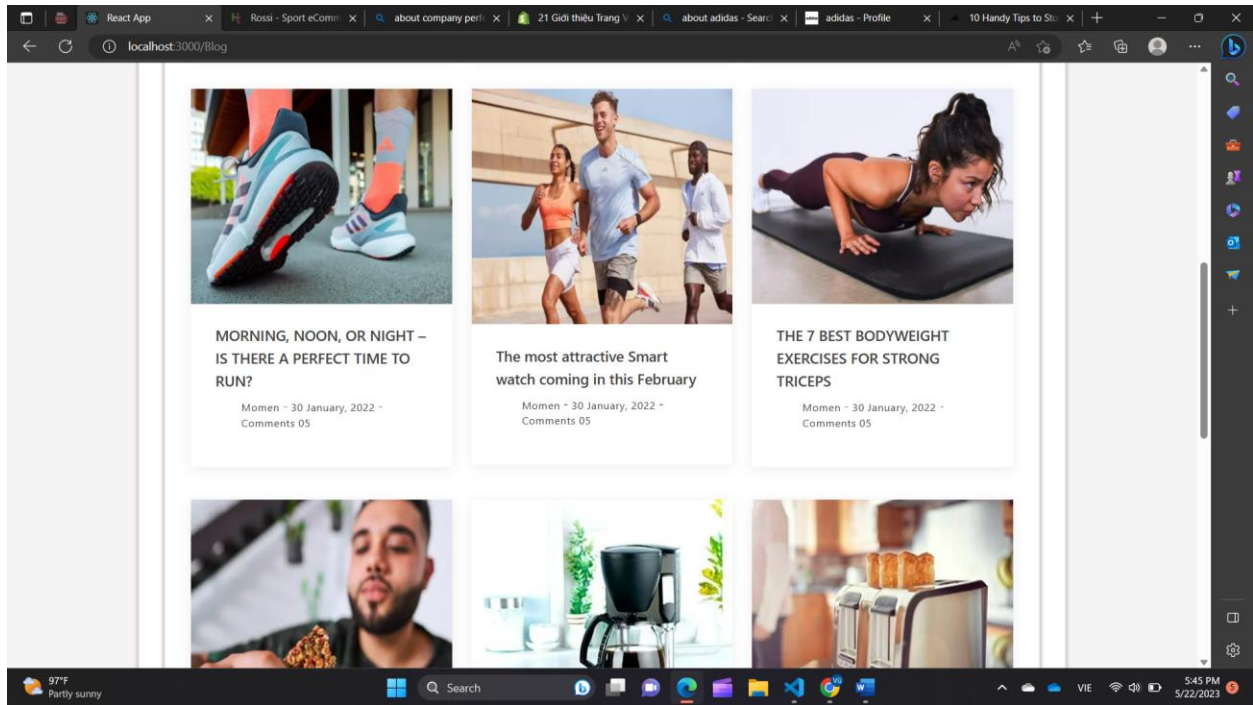


2.5. Giao diện trang contact

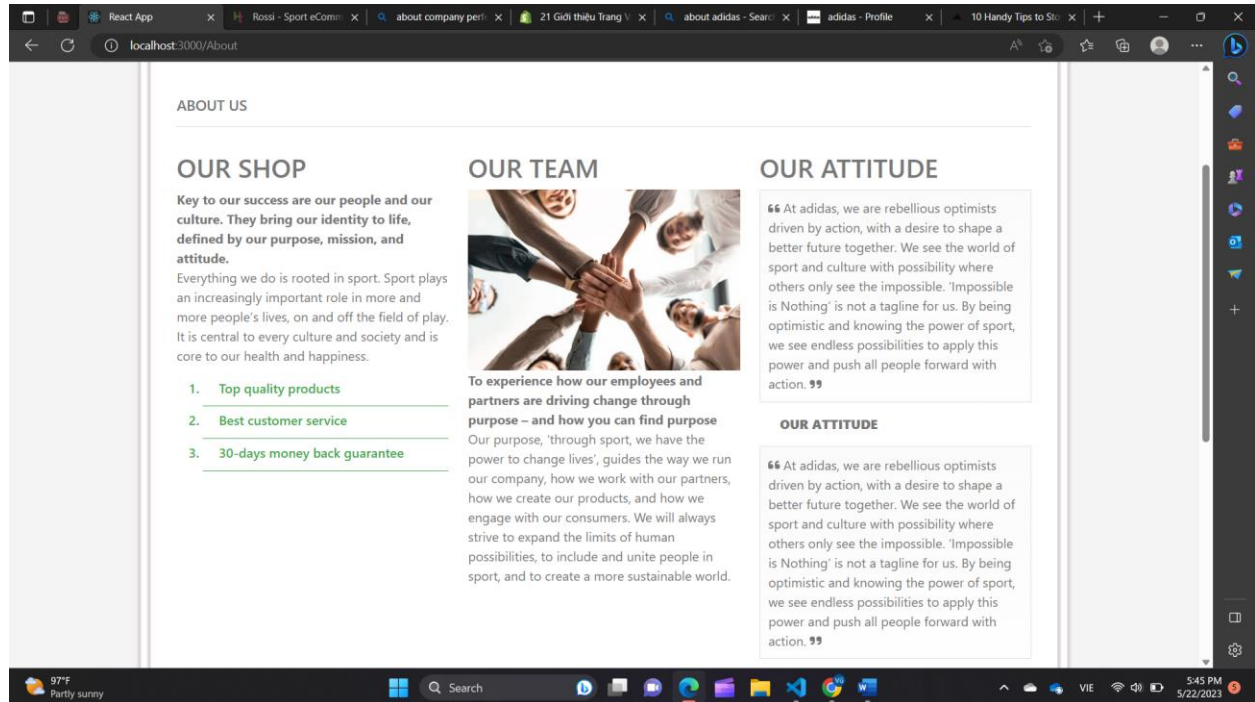


2.6. Giao diện trang blog





2.7. Giao diện trang about



PHẦN 3: TỔNG KẾT

CHƯƠNG I: Tài liệu tham khảo

1. Themeforest.net

Tham khảo cách mẫu thiết kế

<https://themeforest.net>

2. React.dev

<https://react.dev/learn>

3. React Router

<https://reactrouter.com/en/main>

4. Stackoverflow

<https://stackoverflow.com>

5. Codesandbox.io

<https://codesandbox.io>