

DẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC - KỸ THUẬT MÁY TÍNH



BÁO CÁO BÀI TẬP LỚN
LẬP TRÌNH WEB(CO3050)

Đề tài

WEBSITE BÁN ĐIỆN THOẠI

Giảng viên hướng dẫn:	Nguyễn Hữu Hiếu	
Sinh viên:	Lê Nguyễn Huyền Thoại	2012122
	Trần Tuấn Anh	2010878
	Trương Huy Thái	2012036
	Võ Minh Trí	1915667
Email liên hệ:	thoai.le9102002@hcmut.edu.vn	

TP. HỒ CHÍ MINH, THÁNG 6/2022



Mục lục

1 Giới thiệu	2
2 Cơ sở lý thuyết	2
2.1 Thư viện được sử dụng trong trang Web	2
2.1.1 Front End	2
2.1.2 Back end	3
2.2 Các lỗ hổng bảo mật trong ứng dụng Web	4
2.3 SEO	6
2.4 Mô hình Model View Controller (MVC)	7
3 Thiết kế ứng dụng	8
3.1 Thiết kế cơ sở dữ liệu	8
3.1.1 Các yêu cầu dữ liệu của ứng dụng	8
3.1.2 Hiện thực database	8
3.2 Cấu trúc mã nguồn	10
3.2.1 Front-end	10
3.2.2 Back-end	12
4 Hiện thực	20
4.1 Đăng nhập, đăng ký	20
4.2 Giao diện chính	21
4.3 Danh sách sản phẩm và lọc sản phẩm	22
4.4 Chi tiết sản phẩm	23
4.5 Trang tin tức	25
4.6 Trang cá nhân của khách hàng	26
4.7 Trang cá nhân của admin	27
4.8 Thêm, chỉnh sửa sản phẩm	29
4.9 Thêm, chỉnh sửa bài đăng	30
4.10 Giỏ hàng	31
4.11 Giới thiệu	31
5 Cách thức cài đặt ứng dụng, môi trường chạy ứng dụng	32
5.1 Thông tin phiên bản	32
5.2 Link tải source code	32
5.3 Các bước cài đặt	32
6 Nhiệm vụ vai trò của thành viên trong nhóm	36
Tài liệu tham khảo	37



1 Giới thiệu

Hơn 20 năm kể từ ngày chiếc di động thông minh đầu tiên được bán ra, Việt Nam đã trải qua một giai đoạn bùng nổ thật sự. Nielsen cho biết, tỷ lệ sở hữu smartphone so với điện thoại phổ thông (feature phone) đã cán mốc 84%, tăng gấp 3 lần so với nửa thập niên trước. Ngay cả vùng nông thôn, trong 89% dân cư dùng điện thoại di động, thì có đến 68% sử dụng smartphone.

Cụ thể dân số Việt Nam là hơn 90 triệu người thì có đến 39,8 triệu người sử dụng internet (chiếm 44%), 28 triệu người sở hữu tài khoản mạng xã hội (khoảng 31%), 128.3 triệu người có kết nối mạng di động (tương đương 141%) và số người sử dụng tài khoản xã hội trên điện thoại là 24 triệu (chiếm 26%). Rõ ràng, mobile đang là mảnh đất màu mỡ mà bất kỳ doanh nghiệp nào cũng không nên bỏ qua.

Vì những lí do trên nhóm quyết định xây dựng một trang Web bán điện thoại có thương hiệu là BARD qua đó khách hàng có thể xem thông tin và truy cập trực tiếp để mua bán sản phẩm trong một thời gian tương đối ngắn với những tính năng dễ dàng thao tác.

2 Cơ sở lý thuyết

2.1 Thư viện được sử dụng trong trang Web

2.1.1 Front End

- react

– Ưu điểm:

- * 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.
- * Có nhiều công cụ phát triển
- * Làm việc với vấn đề test giao diện: Nó cực kì 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.
 - * React khá nặng nề so với các framework khác, có kích thước tương đương với Angular (Khoảng 35kb so với 39kb của Angular). Trong khi đó Angular là một framework hoàn chỉnh
 - * Khó tiếp cận cho người mới học Web
- clsx: là một package như classnames nhưng vì có ít chức năng hơn cho nên có dung lượng nhỏ đáng kể (223Byte), đây sẽ là sự lựa chọn thích hợp cho bạn nếu như chỉ cần một số chức năng cần thiết mà không muốn làm tăng kích thước của dự án.
 - react-router-dom: React-Router cho phép ta định tuyến "luồng dữ liệu" (data flow) trong ứng dụng của bạn một cách rõ ràng. Nó tương đương với sự khảng định, nếu có URL này, nó sẽ tương đương với Route này, và giao diện tương ứng.
 - axios: là một HTTP client được viết dựa trên Promises được dùng để hỗ trợ cho việc xây dựng các ứng dụng API từ đơn giản đến phức tạp và có thể được sử dụng cả ở trình duyệt hay Node.js.
 - @fortawesome: là một trong những thư viện icon phổ biến và được các Frontend Developer sử dụng một cách rộng rãi. Đây là một thư viện khá lớn với rất nhiều icon miễn phí cũng như trả phí đáp ứng đầy đủ các nhu cầu của người dùng.
 - @editorjs: là một thư viện giống như một trình soạn thảo văn bản, cho phép người dùng ghi lại nội dung, chèn ảnh, ... một cách dễ dàng nhất.
 - react-bootstrap: Là một trong những thư viện React lâu đời nhất, React-Bootstrap đã phát triển và phát triển cùng với React, khiến nó trở thành một lựa chọn tuyệt vời làm nền tảng giao diện người dùng.
 - react-cookie: Là một thư viện cho phép thêm, sửa, xóa cookie khi viết phần mềm bằng thư viện reactJS

2.1.2 Back end

Bên phía back-end sử dụng thư viện MySQLi. MySQLi (hay MySQL Improved) extension là một phần mở rộng (còn được gọi là thư viện hay mô-đun) được phát triển dành cho ngôn ngữ PHP giúp lập trình viên có thể dễ dàng kết nối tới cơ sở dữ liệu MySQL. Thông thường thì MySQL extension sẽ được cài đặt mặc định khi bạn cài PHP trên máy tính.



Hình 1: Ví dụ về sử dụng MySQLi

```
class Model {
    protected $servername = "localhost";
    protected $username = "root";
    protected $password = "tiachop1";
    protected $db_name = "phoneshop";
    protected $conn;

    public function __construct()
    {
        $this->conn = mysqli_connect($this->servername, $this->username, $this->password, $this->db_name);
        // Check connection
        if (!$this->conn) {
            echo "null";
            die("Connection failed: " . mysqli_connect_error());
        }
    }

    public function dbConnection() {
        $conn = mysqli_connect($this->servername, $this->username, $this->password, $this->db_name);
        // Check connection
        if (!$conn) {
            echo "null";
            die("Connection failed: " . mysqli_connect_error());
        }
    }
}
```

2.2 Các lỗ hổng bảo mật trong ứng dụng Web

- Unvalidated Parameters

- Khái niệm: Unvalidated Parameters là lỗi kẻ tấn công có thể dễ dàng thay đổi bất kỳ phần nào của yêu cầu HTTP trước khi gửi URL, Cookies, form fields ...
- Cách xử lý: Dánh giá mã (kiểm tra biến so với danh sách các giá trị được phép), sử dụng Application firewalls.

- Broken Access Control

- Khái niệm: Broken Access Control là lỗi phổ biến về việc phân quyền trong hệ thống. Các lỗi phân quyền thường do thiếu đi các bộ phát hiện lỗi tự động hoặc cách thức kiểm thử hoặc các hàm kiểm thử chưa hiệu quả khiến cho hệ thống bị rò rỉ
- Cách xử lý: Giới hạn quyền truy cập API và controller để giảm thiểu thiệt hại, thực hiện các cơ chế kiểm soát quyền truy cập và thực hiện nó trên toàn ứng dụng, cài đặt các rule ở Model để quản lý các thao tác với database.

- Broken Account and Session Management

- Khái niệm: Broken Account and Session Management cho phép hacker có thể truy cập trái phép tài nguyên nội bộ (admin page, inside, control page ...) để thực hiện hành vi nâng quyền quản trị, tấn công vào các dạng như session... để đánh cắp mật khẩu, session token hoặc tận dụng các lỗi khác nhằm giả mạo người dùng.



– Cách xử lý: Triển khai cơ chế xác thực 2 lớp nhằm chống lại các cuộc tấn công tự động như Brute Force, kiểm tra độ an toàn của Password, không cho phép người dùng đặt những mật khẩu quá đơn giản chẳng hạn chỉ toàn số, hoặc toàn chữ. Bạn cũng có thể kiểm tra mật khẩu người dùng đặt trong top 10000 passwords tệ nhất.

- Cross-Site Scripting (XSS)

- Khái niệm : hacker chèn những đoạn script độc hại vào website và sẽ được thực thi ở phía người dùng.
 - Cách xử lý: Sử dụng frameworks mà tự động phát hiện XSS như (Ruby on Rails, ReactJS, Laravel)

- Command injection

- Khái niệm: Command injection là lỗi Cho phép kẻ tấn công chuyển tiếp mã độc hại trong các form hoặc URL.
 - Cách xử lý: Mã hoá các dữ liệu đầu vào, tránh các lời gọi hệ thống(system calls), chạy với các đặc quyền hạn chế.

- Sensitive Data Exposure

- Khái niệm: Sensitive Data Exposure là việc các dữ liệu nhạy cảm được lưu trữ không an toàn có thể gây ra những ảnh hưởng to lớn cho hệ thống máy chủ, cũng như cho khách hàng vd như tài khoản tín dụng, thẻ ngân hàng, thông tin mật...
 - Cách xử lý: Sử dụng HTTPS có chứng chỉ phù hợp và PFS (Perfect Forward Secrecy). Không nhận bất cứ thông tin gì trên các kết nối không phải là HTTPS. Có cờ an toàn trên cookie, nếu có dữ liệu nhạy cảm thực sự cần, lưu trữ mã hóa nó và đảm bảo rằng tất cả các mật khẩu được sử dụng hàm Hash để bảo vệ.

- Missing Function Level Access Control

- Khái niệm: là những sai sót trong hạn chế truy cập dẫn đến Hacker có thể giả mạo URL để truy cập vào những trang thông tin ẩn.
 - Cách xử lý: Ở phía máy chủ, phải luôn được phân quyền một cách triệt để từ khâu thiết kế. Không có ngoại lệ – mọi lỗ hổng sẽ dẫn đến đủ các vấn đề nghiêm trọng



2.3 SEO

Search Engine Optimization (tối ưu hóa công cụ tìm kiếm) là một quy trình nâng cao thứ hạng của website trên các công cụ tìm kiếm giúp người dùng có thể tìm thấy trang web dễ dàng hơn trên bảng kết quả tìm kiếm

- On-Page Optimization: quy tắc đầu tiên của On-page optimization là suy nghĩ về mục tiêu người dùng đang tìm kiếm và đảm bảo các từ khóa phù hợp có trên trang Web.
 - Meta Tags: Các thẻ meta là các thẻ dữ liệu chính thức cho mỗi trang web được tìm thấy giữa các thẻ head mở và đóng trong mã HTML. Meta tags giúp các công cụ tìm kiếm thông tin có liên quan mô tả nội dung của trang, điều này giúp các công cụ tìm kiếm quyết định xem trang web của bạn có phải là một danh sách thích hợp để phản hồi cho một truy vấn tìm kiếm cụ thể hay không.
 - Header Tags: là một yếu tố SEO quan trọng vì chúng truyền đạt cho công cụ tìm kiếm nội dung trang web của bạn. Công cụ tìm kiếm coi bản sao trong Header Tags của trang Web quan trọng phần còn lại.
 - Contents: là những gì trình thu thập thông tin của công cụ tìm kiếm cần để liên kết trang của bạn với một tập hợp các từ khóa hoặc cụm từ khóa.
 - Site Map: là một trang xml trên trang web của bạn kết hợp danh sách các liên kết trong mỗi trang đơn lẻ trên trang web của bạn. Nó cũng có thể trình bày chi tiết thứ bậc của các trang trên trang web của bạn.
- Page Address: Việc triển khai các từ khóa trong việc đặt tên địa chỉ trang sẽ giúp các công cụ tìm kiếm có manh mối về nội dung trang Web.
- Subfolder(Subdirectory): thường được Google coi là miền riêng biệt và thực hiện gấp đôi SEO giúp việc kết quả tìm kiếm có khả năng tìm thấy hơn.
- Off-page Optimization: SEO Off-page là tất cả về việc xây dựng uy tín trực tuyến cho sự tin cậy và danh tiếng cho trang web của bạn. Nó được xác định bởi số lượng, chất lượng và mức độ liên quan của các liên kết đến trang web của bạn để thiết lập cơ quan quản lý SEO của bạn và cuối cùng ảnh hưởng đến xếp hạng kết quả tìm kiếm.
 - Link building: là việc xây dựng các backlink cho website
 - Social media: Các tín hiệu trên mạng xã hội là tín hiệu của công cụ tìm kiếm như các yếu tố xếp hạng tự nhiên
 - Public Relations: tăng cường hiển thị trực tuyến và thường được công nhận và lập chỉ mục bởi tất cả các công cụ tìm kiếm lớn.

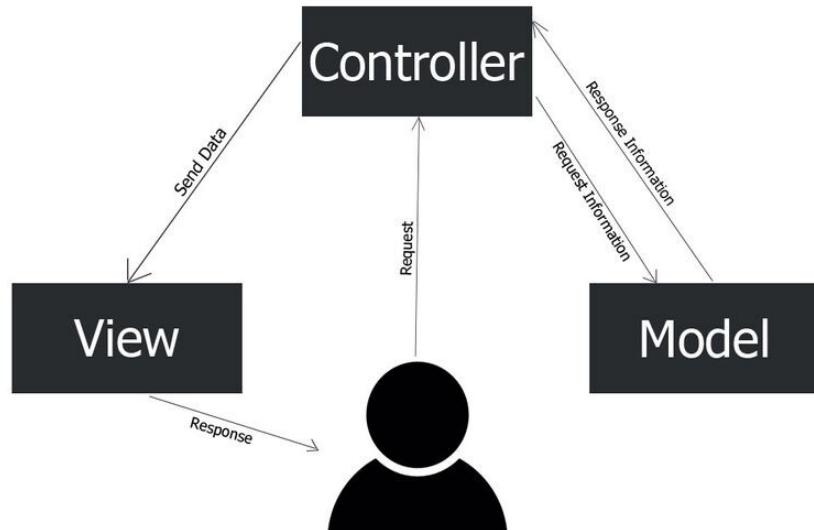


2.4 Mô hình Model View Controller (MVC)

- Model View Controller là mô hình mẫu được sử dụng nhiều nhất cho các ứng dụng Web trên thế giới ngày nay.
 - Model: có nhiệm vụ quản lý dữ liệu trang Web và là cầu nối giữa 2 thành phần bên dưới là View và Controller.
 - View: hiển thị dữ liệu được cung cấp bởi Model ở một định dạng cụ thể.
 - Controller: điều khiển Model và View để xử lý công việc.

Hình 2: Mô hình MVC

Model-View-Controller



- **Ưu điểm:** MVC không sử dụng viewstate nên khá tiết kiệm diện tích băng thông. Khi sử dụng, người dùng có thể sử dụng ứng dụng trên web cần tương tác gửi và nhận dữ liệu một cách liên tục. Do đó, việc giảm băng thông giúp cho website hoạt động tốt và ổn định hơn.
- **Nhược điểm:** với các dự án nhỏ, mô hình MVC có thể gây cồng kềnh, tốn thời gian trong quá trình phát triển cũng như thời gian trung chuyển dữ liệu.



3 Thiết kế ứng dụng

3.1 Thiết kế cơ sở dữ liệu

3.1.1 Các yêu cầu dữ liệu của ứng dụng

Ứng dụng yêu cầu bên phía database phải:

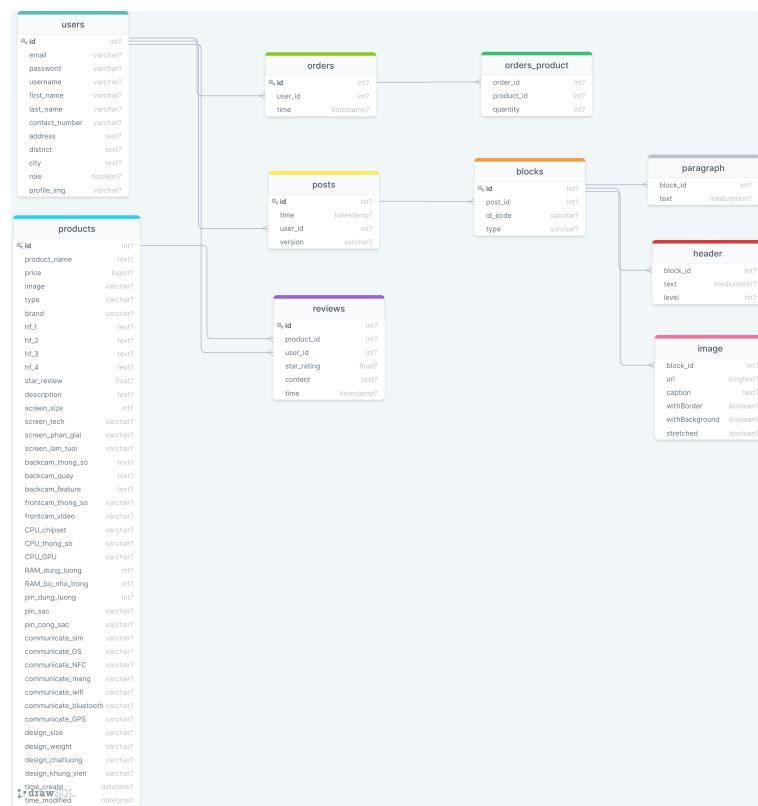
- Lưu trữ được các thông tin chi tiết về từng mặt hàng trong cửa hàng.
- Lưu trữ thông tin của người dùng, đồng thời cũng lưu thêm các lịch sử mua hàng của họ trên website.
- Lưu trữ dữ liệu về những bình luận, đánh giá của khách hàng đối với sản phẩm.
- Lưu trữ một hệ thống các bài viết để promote cho cửa hàng.

3.1.2 Hiện thực database

Trong project này, nhóm sử dụng hệ quản trị cơ sở dữ liệu là MySQL.

Chi tiết về cơ sở dữ liệu:

Hình 3: Thiết kế database





Chi tiết các bảng dữ liệu:

- Products: Đây là bảng lưu trữ thông tin các sản phẩm trong cửa hàng
 - Có khóa chính là trường id.
 - Trường product_name và price là tên và giá cả của sản phẩm.
 - Trường star_review là số sao người dùng đánh giá cho sản phẩm.
 - Trường image là trường lưu trữ liên kết đến hình ảnh của sản phẩm.
 - Các trường còn lại là các chi tiết thêm của sản phẩm được hiển thị trong trang chi tiết sản phẩm của cửa hàng.
- Users: Đây là bảng lưu trữ thông tin người dùng của cửa hàng.
 - Có khóa chính là trường id.
 - Hai trường email và user_name là email và tên đăng nhập của người dùng, và cả hai đều là duy nhất, không giống nhau giữa các users.
 - Trường password là mật khẩu của người dùng, và được lưu trong database sau khi đã được mã hóa bởi md5.
 - Trường profile_img là trường lưu trữ liên kết đến hình ảnh đại diện của người dùng.
 - Các trường còn lại là các thông tin thêm về người dùng .
- Orders: Đây là bảng lưu trữ thông tin lịch sử mua hàng của khách hàng.
 - Có khóa chính là trường id.
 - Có khóa ngoại là trường user_id tham chiếu đến khóa chính "id" của bảng Users.
 - Trường time là thời gian người dùng thực hiện order đó.
- Orders_Product: Đây là dạng thực thể yếu của bảng Orders, sử dụng để lưu thông tin của các sản phẩm tương ứng với một order.
 - Có khóa ngoại là trường order_id tham chiếu đến khóa chính "id" của bảng Orders.
 - Có khóa ngoại là trường product_id tham chiếu đến khóa chính "id" của bảng Products.
 - Trường quantity là số lượng của sản phẩm mà khách hàng đặt mua.
- Reviews: Đây là bảng lưu trữ các đánh giá của người dùng cho các sản phẩm trong cửa hàng.
 - Có khóa chính là trường id.
 - Có khóa ngoại là trường user_id tham chiếu đến khóa chính "id" của bảng Users.



- Có khóa ngoại là trường product_id tham chiếu đến khóa chính "id" của bảng Products.
- Trường star_rating là số sao người dùng đánh giá trong review.
- Trường data là nội dung của review.
- Posts: Đây là bảng lưu trữ các bài viết của người dùng trên website.
 - Có khóa chính là trường id.
 - Có khóa ngoại là trường user_id tham chiếu đến khóa chính "id" của bảng Users.
 - Trường time là thời gian đăng lên bài viết đó.
 - Trường version chỉ version hiện tại của bài viết.
- Blocks: Đây là bảng lưu trữ các thành phần chi tiết(block) của một bài viết.
 - Có khóa chính là trường id.
 - Có khóa ngoại là trường post_id tham chiếu đến khóa chính "id" của bảng Posts.
 - Trường id_code là mã code của 1 block biểu diễn bằng ký tự.
 - Trường type chỉ loại dữ liệu của block(đoạn văn, hình ảnh, ...)
- Cùng với các bảng con lưu thông tin chi tiết của từng loại Block(Header, Paragraph, Image).

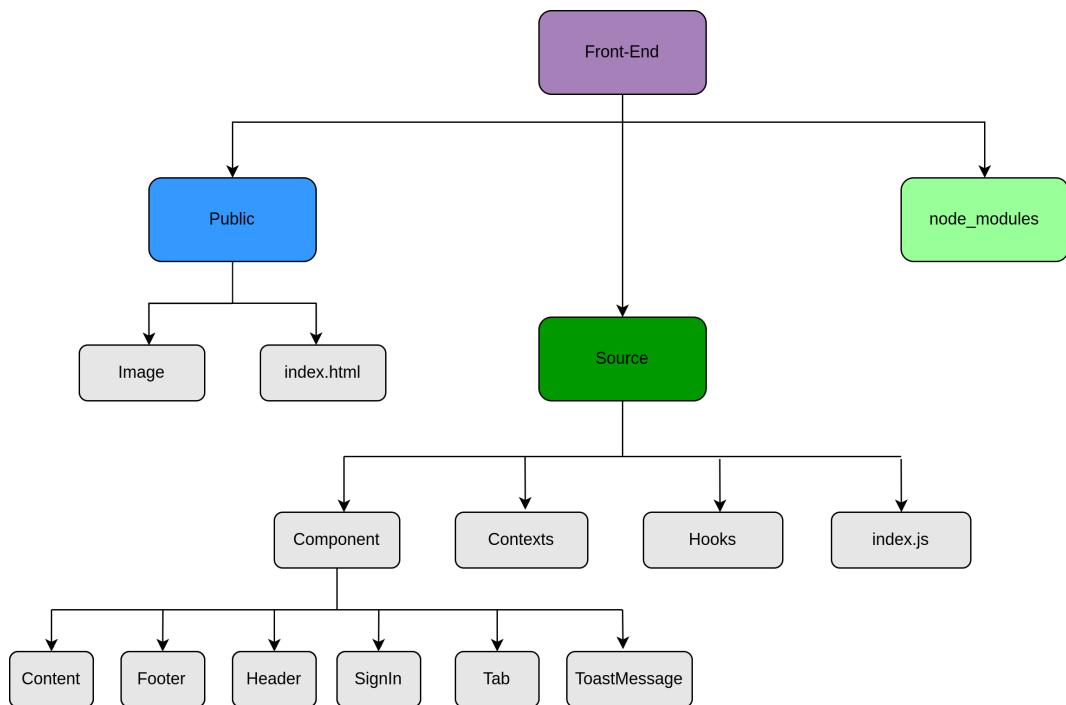
3.2 Cấu trúc mã nguồn

3.2.1 Front-end

Nhóm sử dụng ReactJS để xây dựng Front-end.



Hình 4: Cấu trúc mã nguồn



- Public: Folder "Public" chứa file index.html, các hình ảnh dùng cho trang Web và file chạy ứng dụng.
- Components: chứa các file xây dựng nền giao diện tương tác với người dùng bao gồm:
 - Header: chứa giao diện và các tương tác đến phần header và navigation.
 - Footer: chứa giao diện của phần footer.
 - SignIn: chứa giao diện và các tương tác của trang đăng nhập hoặc đăng ký thành viên.
 - Content: chứa giao diện của các nội dung chính trong trang web như trang home, danh sách sản phẩm, thông tin chi tiết, ...
 - Card: chứa giao diện của các thẻ sản phẩm hoặc thẻ tin tức.
 - Tab: chứa giao diện của thành phần các tab nội dung như thông tin người dùng, cấu hình chi tiết của sản phẩm,
 - ToastMessage: chứa giao diện của thông báo dạng thẻ xuất hiện khi người dùng đặt hàng, hay admin thêm xóa chỉnh sửa sản phẩm hoặc bài đăng.
- Contexts: chứa file UserContext.js. Trong UserContext.js nhóm lưu các dữ liệu được sử dụng xuyên suốt giữa các trang trong trang web cùng với đó là các tính



năng như đăng nhập, đăng ký, chỉnh sửa thông tin tài khoản, thêm sản phẩm vào giỏ hàng, thanh toán,....

- Hooks: gồm file userFetchData.js sử dụng để lấy dữ liệu về cho cơ sở dữ liệu.
- index.js: index.js là file chính chứa mọi component của trang web. Tại đây các component được định tuyến dữ liệu (nếu có URL này nó sẽ tương đương với Route và giao diện tương ứng).

Hình 5: file index.js

```
import App from './App'
import SignIn from './Components/SignIn/SignIn'
import SignUp from './Components/SignIn/SignUp'
import Home from './Components/Content/Home/Home';
import News from './Components/Content/News/News'
import NewDetail from './Components/Content/NewDetail/NewDetail'
import ProductDetail from './Components/Content/ProductDetail/ProductDetail'
import ProductList from './Components/Content/ProductList/ProductList'
import AddProduct from './Components/Content/AddProduct/AddProduct'
import EditProduct from './Components/Content/EditProduct/EditProduct'
import Account from './Components/Content/Account/Account'
import Cart from './Components/Content/Cart/Cart'
import About from './Components/Content/About/About'
import AddNew from './Components/Content/AddNew/AddNew'
import NotFound from './Components/Content/NotFound/NotFound';
import { UserProvider } from "./Contexts/UserContext";
import EditNew from './Components/Content/EditNew/EditNew';

import './index.css'
import 'animate.css'

const container = document.getElementById('root')
const root = ReactDOM.createRoot(container)

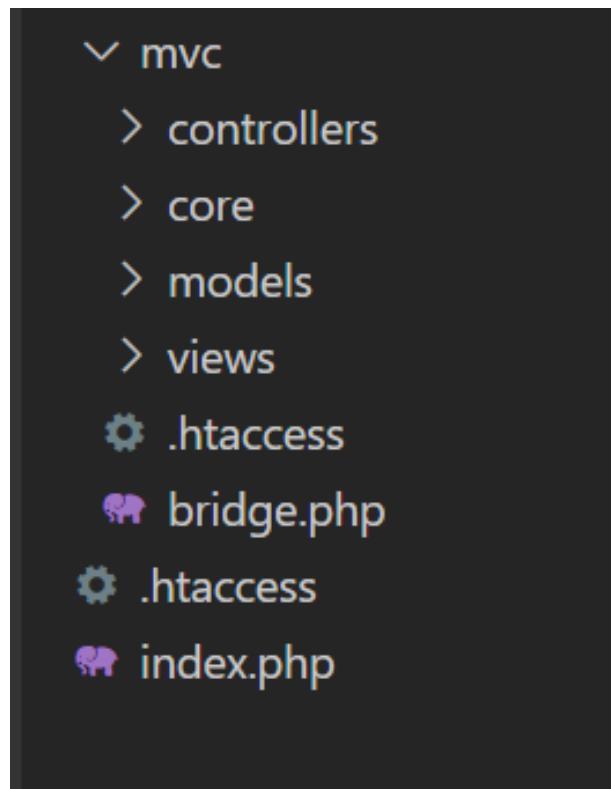
root.render(
  <React.StrictMode>
    <UserProvider>
      <BrowserRouter>
        <Routes>
          <Route path="/" element={<App />}>
            <Route index element={<Home />}></Route>
            <Route path="account/:id" element={<Account />}></Route>
            <Route path="products">
              <Route index element={<ProductList />}></Route>
              <Route path="brand" element={<ProductList />}></Route>
              <Route path="brand/:id" element={<ProductDetail />}></Route>
              <Route path="add" element={<AddProduct />}></Route>
              <Route path="edit/:id" element={<EditProduct />}></Route>
            </Route>
            <Route path="news">
              <Route index element={<News />}></Route>
              <Route path=":id" element={<NewDetail />}></Route>
              <Route path="add:userid" element={<AddNew />}></Route>
              <Route path="edit/:id" element={<EditNew />}></Route>
            </Route>
            <Route path="cart" element={<Cart />}></Route>
            <Route path="about" element={<About />}></Route>
            <Route path="*" element={<NotFound />}></Route>
          </Routes>
        </BrowserRouter>
      </UserProvider>
    </React.StrictMode>
)
```

3.2.2 Back-end

Nhóm xây dựng back-end dựa trên mô hình MVC (Model - View - Controller)



Hình 6: Mô hình MVC



- Core: Đây là thành phần cốt lõi, liên kết các thành phần trong model lại với nhau. Server sẽ bắt đầu từ file index.php, và sau đó bridge.php sẽ được gọi, và từ đây sẽ chạy hàm main() trong file main.php ở trong core.
Hàm main được sử dụng để cắt chuỗi URL người dùng nhập vào, và dùng chuỗi cắt đó để gọi hàm execute() và các tham số trên mỗi lớp controller tương ứng



Hình 7: Hàm main trong core

```
9
10
11 function main() {
12     $arr = urlProcess();
13
14     if (isset($arr[0]) && file_exists("./mvc/controllers/".$arr[0].".php")) {
15         $controller_name = $arr[0];
16
17         require_once "./mvc/controllers/".$controller_name.".php";
18         $controller = new $controller_name();
19
20         try {
21             $controller->execute($arr);
22         } catch (Exception $e) {
23             require_once "./mvc/views/ErrorView.php";
24             $err = new ErrorView($e->getMessage());
25             $err->respond();
26         }
27     } else {
28         require_once "./mvc/views/ErrorView.php";
29         $err = new ErrorView("file not exist");
30         $err->respond();
31     }
32 }
33
34 }
```

Ngoài ra trong core còn có lớp Model, là lớp định nghĩa và kết nối với cơ sở dữ liệu trong MySQL

Hình 8: Lớp model trong core

```
class Model {
protected $servername = "localhost";
protected $username = "root";
protected $password = "tiachop1";
protected $db_name = "phoneshop";
protected $conn;

public function __construct()
{
    $this->conn = mysqli_connect($this->servername, $this->username, $this->password, $this->db_name);
    // Check connection
    if (!$this->conn) {
        echo "null";
        die("Connection failed: " . mysqli_connect_error());
    }
}

public function dbConnection()
{
    $conn = mysqli_connect($this->servername, $this->username, $this->password, $this->db_name);
    // Check connection
    if (!$conn) {
        echo "null";
        die("Connection failed: " . mysqli_connect_error());
    }
}
```

- Controller: Được sử dụng để điều hướng server và gọi các hàm trong phần model tương ứng

Để gọi được đến các controller, người dùng phải nhập URL dưới dạng localhost/[servername]/[Controller]/[Arg1]/[Arg2]... với Controller là tên của các controller, Arg là các tham số người dùng truyền vào (nếu có)



Hình 9: Ví dụ về lớp Order trong controller với hàm execute()

```
13     $this->model = new OrderModel();
14     $this->view = new OrderView();
15 }
16
17 function execute($arr) {
18     if (isset($arr[1])) {
19         if ($arr[1]=="create") {
20             echo "create";
21             $data = json_decode(file_get_contents("php://input"));
22
23             $user_id = $data->user_id;
24             $product_list = $data->product_list;
25             $this->view->createRespond($this->model->create($user_id, $product_list));
26         }
27         else if ($arr[1]=="read") {
28             //echo "read";
29             //echo $arr[2];
30             //echo is_integer($arr[2]);
31             //echo ((int)$arr[2])>0;
32             if (isset($arr[2]) && is_numeric($arr[2]) && ((int)$arr[2]>0) {
33                 //echo "right para";
34                 $result = $this->model->read((int)$arr[2]);
35                 $this->view->readRespond($result);
36             }
37         }
38     }
39     else throw new Exception("wrong user id");
40 }
```

Trong controller có 5 lớp Order (Order.php), Post(Post.php), Product(Product.php), Review(Review.php) và User(User.php), bên trong mỗi lớp controller sẽ có các đối tượng Model và View được tạo tương ứng.

– Lớp Order: có 2 phương thức chính

- * create: localhost/[servername]/Order/create - sử dụng để gọi hàm tạo order.
- * read: localhost/[servername]/Order/read/[user_id] - sử dụng để gọi hàm lấy thông tin order.

– Lớp Post: có 4 phương thức chính

- * create: localhost/[servername]/Post/create - sử dụng để gọi hàm thêm bài viết.
- * read: localhost/[servername]/Post/read - sử dụng để gọi hàm lấy thông tin danh sách các bài viết, hoặc lấy thông tin của bài viết duy nhất bằng cách thêm "/[post_id]" vào đường dẫn.
- * update: localhost/[servername]/Post/update - sử dụng để gọi hàm cập nhật bài viết
- * delete: localhost/[servername]/Post/delete/[post_id] - sử dụng để xóa bài viết trên bảng Post

– Lớp Product: có 6 phương thức chính

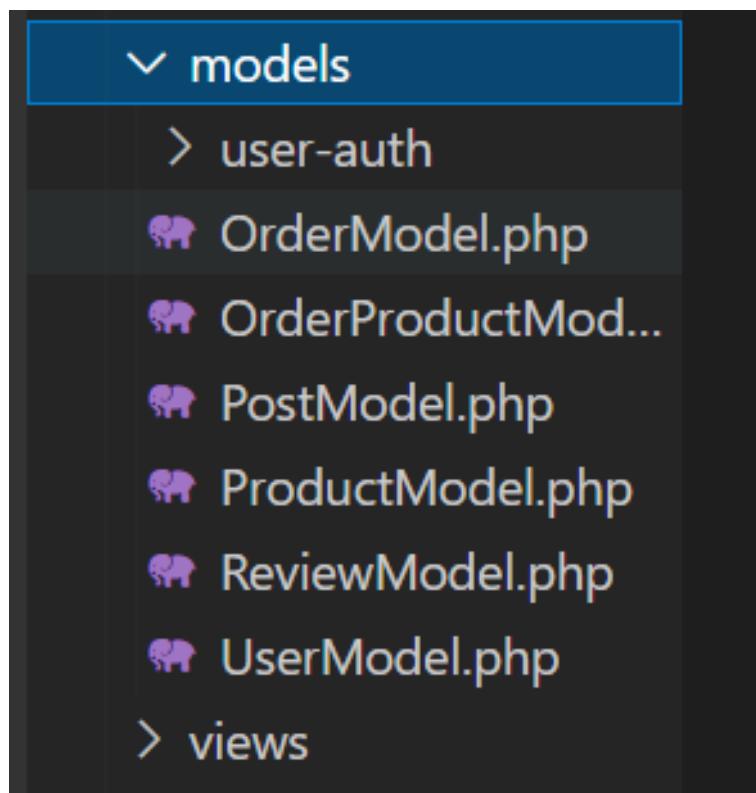
- * 4 phương thức đầu là create, read, update, delete tương tự với 2 lớp trên
- * searchProduct: localhost/[servername]/Product/searchProduct - sử dụng để gọi hàm tìm kiếm sản phẩm theo tên bên lớp ProductModel/
- * filter: localhost/[servername]/Product/filter - sử dụng để gọi hàm lọc các sản phẩm theo loại, thương hiệu, giá cả,... bên lớp ProductModel

– Lớp Review: có 4 phương thức chính



- * create: localhost/[servername]/Post/create - sử dụng để thêm bình luận về sản phẩm.
- * read: localhost/[servername]/Post/read/[product_id] - sử dụng để gọi hàm lấy thông tin danh sách các bình luận về sản phẩm.
- * update: localhost/[servername]/Post/update - sử dụng để gọi hàm cập nhật bình luận
- * delete: localhost/[servername]/Post/delete/[review_id] - sử dụng để xóa bình luận về sản phẩm
- Lớp User: có 5 phương thức chính
 - * 4 phương thức đầu là create, read, update, delete tương tự như trên
 - * login: localhost/[servername]/User/login - dùng để gọi hàm xác thực người dùng bên UserModel
- Model: Đây là thành phần giao tiếp với database, bao gồm các lớp và phương thức sẽ được sử dụng bởi các controller

Hình 10: Các lớp trong model



Mỗi lớp trong model sẽ có các phương thức xử lý các tác vụ trên mỗi bảng tương



ứng trong database

Ngoài ra trong model còn có một lớp Auth (Authentication) để có thể xác thực người dùng và mã hóa thông tin người dùng gửi từ server thành các token để tăng tính bảo mật

Hình 11: Token người dùng nhận được sau khi đăng nhập thành công

The screenshot shows a Postman interface with a POST request to `http://localhost/phoneshop_backend_mvc/User/login`. The Body tab is selected, showing the following JSON payload:

```
1: {
2:   "username": "haicau2",
3:   "password": "tiachop1"
4: }
```

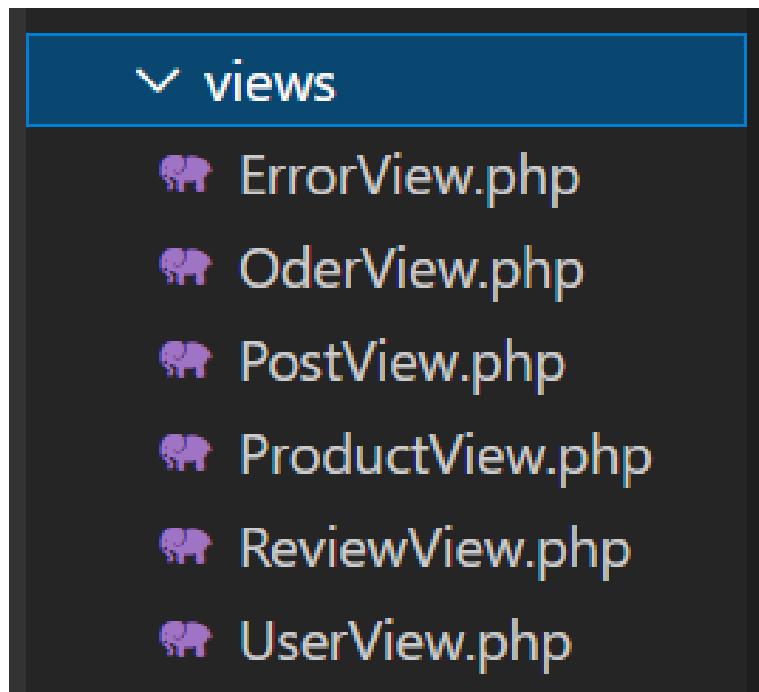
The response tab shows a successful 200 OK status with a response time of 58 ms and a response size of 877 B. The response body is:

```
1: {
2:   "message": "successful",
3:   "token": "dwV1qQslwlx5xn1LxR9UKq6ELI7aWNtXm111mMx9BqWUNCjf5YNtXYyiC5kGYcStb+4BnKZTZY060eSY3RBH9y蒲uvRbzaJBnBn40JFUXShy5/gsqu0u+S3xGh0YSfegtje1pePVXh70d3I/tc2sGn01sfFI1TQPh2gcxgMhx8deD0Iw0hsjtwrZX4DCcu6OheyAQRj/AR9qeWqyMz14DIVMmz5GP6E6odeZMXDiuWpknrJ71W6r4tqXo75kX/n8s6UPJuFp5JUHiCpx854F4+QKKb2P29QqauSg39ExaIy4Zo="
```

- View: View được các controller gọi để respond lại các request từ phía client



Hình 12: Các lớp trong view



Vì bên phía front-end đã sử dụng react nên phần view bên server không trả về nguyên một trang html đầy đủ, mà chỉ trả về các message và data dưới dạng json, sau đó mới được react sử dụng bên phía front-end



Hình 13: Ví dụ về lớp ProductView

```
class ProductView {
    public function readRespond($result) {
        echo json_encode($result);
    }

    public function createRespond($success) {
        if ($success)
            echo json_encode(array("message" => "success"));
        else
            echo json_encode(array("message" => "error"));
    }

    public function updateRespond($success) {
        if ($success)
            echo json_encode(array("message" => "success"));
        else
            echo json_encode(array("message" => "error"));
    }

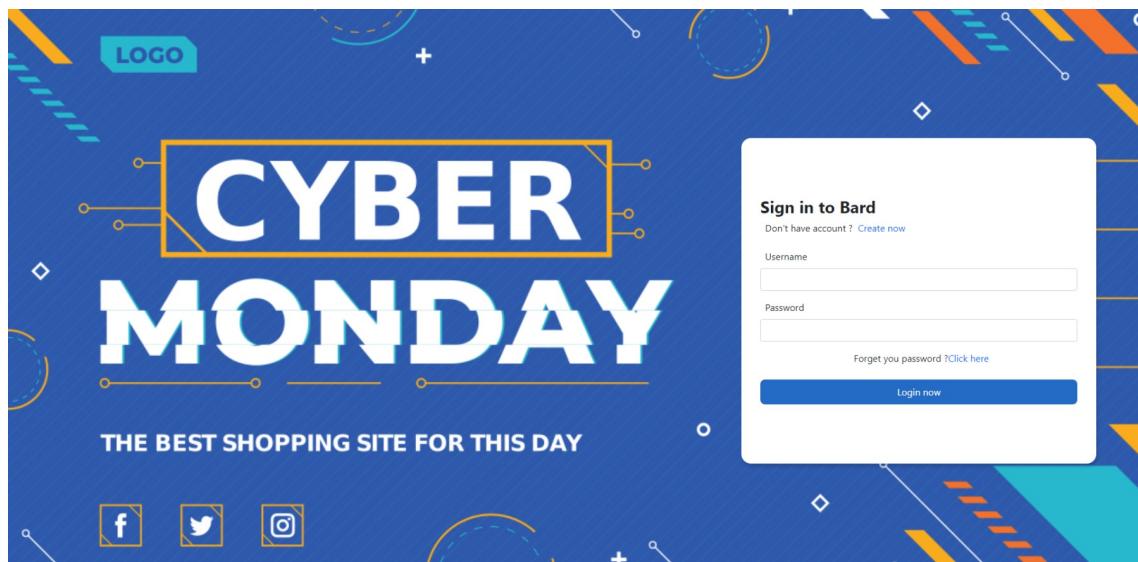
    public function deleteRespond($success) {
        if ($success)
            echo json_encode(array("message" => "success"));
        else
            echo json_encode(array("message" => "error"));
    }
}
```



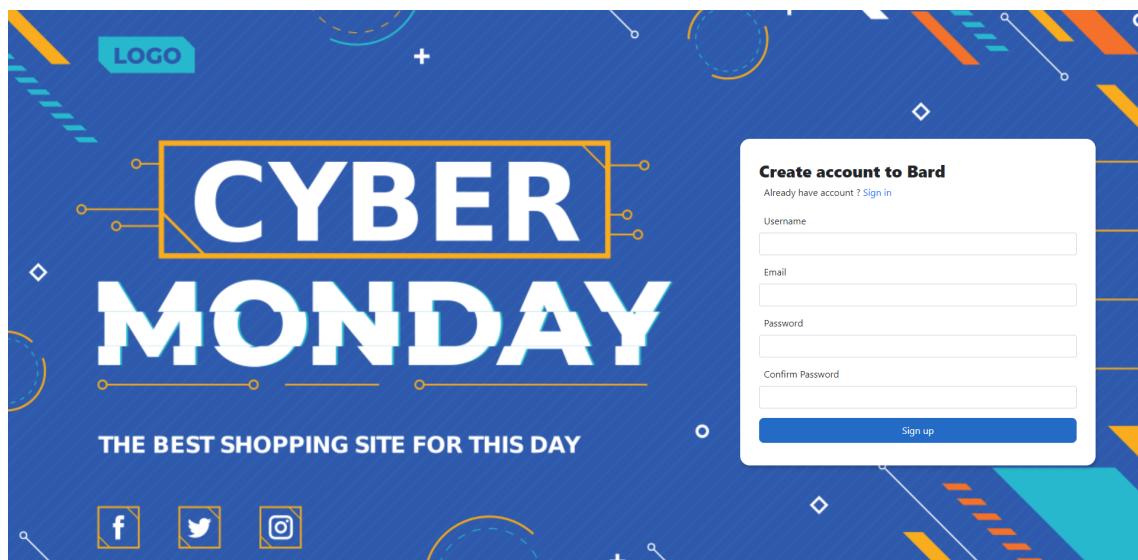
4 Hiện thực

4.1 Đăng nhập, đăng ký

Hình 14: Đăng nhập



Hình 15: Đăng ký





4.2 Giao diện chính

Hình 16: Giao diện chính

The screenshot shows a mobile phone e-commerce website. At the top, there's a navigation bar with a menu icon, the logo 'BARD', and 'Sign in' / 'Sign up' buttons. Below the navigation is a large banner for 'CYBER MONDAY' with text 'THE CYBEROFFERS ARE HERE!' and sections for 'SPECIAL OFFERS', 'THOUSANDS OF ITEMS', 'THE BEST QUALITY', and 'PURCHASING ADVICE'. To the right of the banner are social media links for Facebook, Twitter, Instagram, and YouTube. Below the banner is a section titled 'New arrival' featuring four smartphone models:

Smartphone	Model	Price
Smartphone	Samsung Galaxy A32	\$ 800
Smartphone	Samsung Galaxy S20 FE	\$ 1000
Smartphone	Samsung Galaxy S21 FE	\$ 1150
Smartphone	Samsung Galaxy S22 Ultra	\$ 1299

Hình 17: Giao diện chính

The screenshot shows a mobile phone e-commerce website. At the top, there's a navigation bar with a menu icon, the logo 'BARD', and 'Sign in' / 'Sign up' buttons. Below the navigation is a large banner for a promotion with text 'Ưu Đãi Khủng', 'Giảm Giá Lên Tận 7.000.000Đ', and 'THU CŨ ĐỔI MỚI TIẾT KIỆM' with a 'MUA NGAY' button. To the right of the banner is another promotional graphic for 'BACK TO SCHOOL SALE' with 'UP TO 50% OFF'. Below these banners is a section titled 'Trending' featuring four smartphone models:

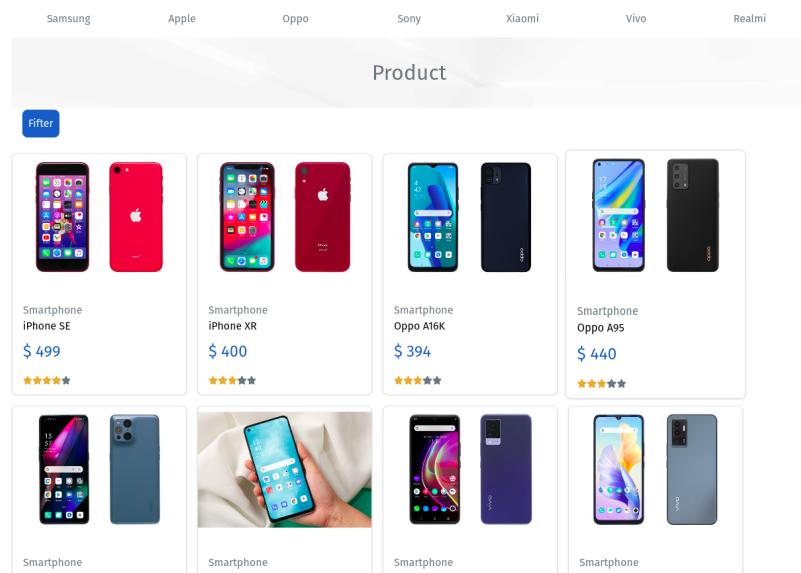
Smartphone	Model	Price
Smartphone	Vivo V21	\$ 399
Smartphone	Vivo V23e	\$ 450
Smartphone	Vivo X70 Pro	\$ 634
Smartphone	Vivo Y72	\$ 310

Below the trending section is a 'News' section with a 'See more' button.

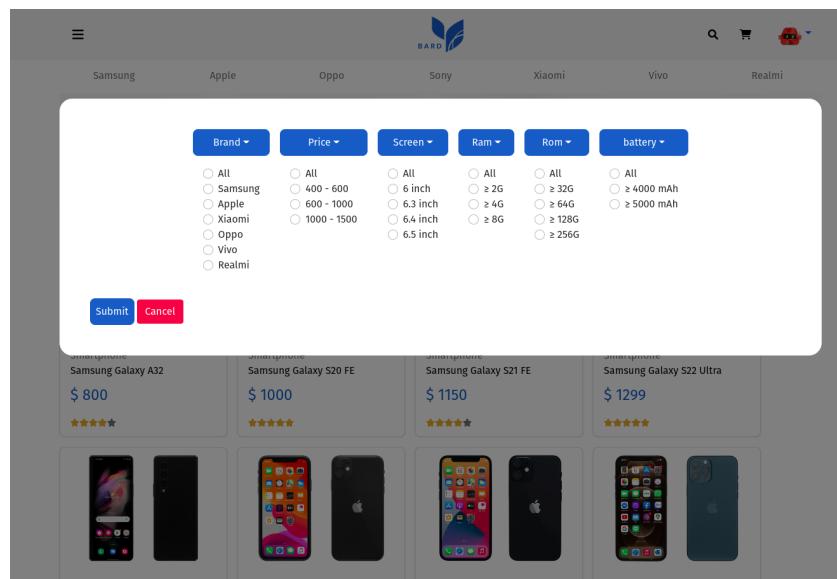


4.3 Danh sách sản phẩm và lọc sản phẩm

Hình 18: Danh sách sản phẩm

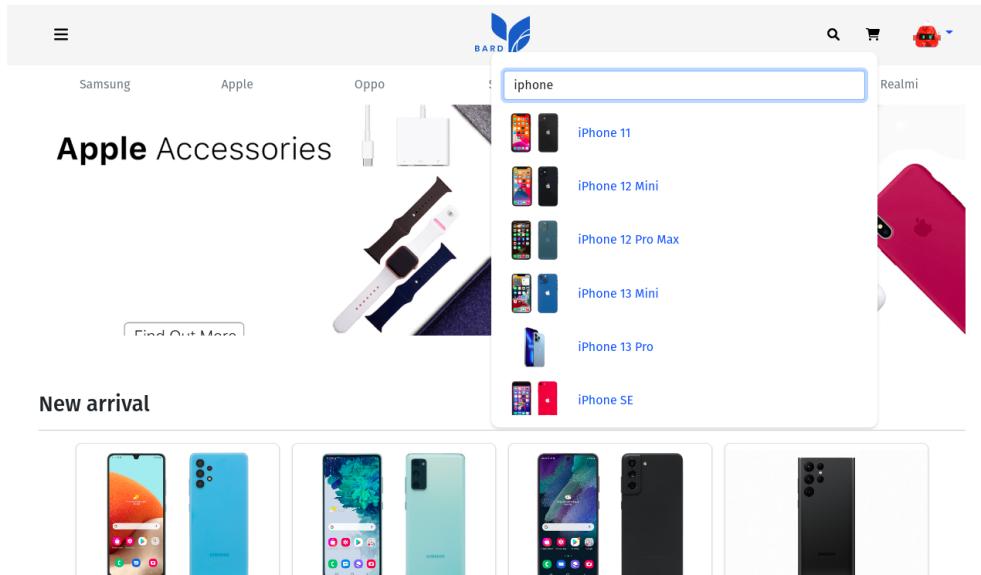


Hình 19: Lọc sản phẩm





Hình 20: Tìm kiếm sản phẩm



4.4 Chi tiết sản phẩm

Hình 21: Thông tin sản phẩm chi tiết

Samsung Galaxy S20 FE

★★★★★

Highlight feature

- Không gian giải trí bất tận
- Hiệu năng mạnh mẽ
- Bừng sáng mọi khung hình
- Sử dụng liên tục trong cả ngày dài

Price: \$ 1000

- 1 +

Buy Now Add to cart

Description Tech Reviews

Samsung S20 FE là chiếc điện thoại sắp được ra mắt từ Samsung, với chữ FE đằng sau tên gọi của máy có nghĩa là Fan Edition. Đây là dòng điện thoại ra mắt như để gửi lời tri ân đến các fan trung thành đã và đang sử dụng các sản phẩm của Samsung. Với số lượng sản phẩm được giới hạn và chỉ mở bán trong...

Show more



Hình 22: Cấu hình sản phẩm

Màn hình		Ram & Bộ nhớ trong	
Kích thước màn hình	6	RAM	6GB
Công nghệ màn hình	Super AMOLED	Bộ nhớ trong	128GB
Độ phân giải	1080 x 2400 pixels (FullHD+)	Pin	
Tần số quét	90Hz	Dung lượng	5000mAh
Camera sau		Công nghệ sạc	Sạc nhanh 15W
Thông số	Camera góc rộng: 64 MP, f/1.8, 26mm, PDAF; Camera góc siêu rộng: 8 MP, f/2.2, 123, 1/4.0, 1.12µm; Camera macro: 5 MP, f/2.4; Cảm biến chiều sâu: 5 MP, f/2.4	Cổng sạc	USB Type-C
Quay video	Giao tiếp và kết nối		
Tính năng	Góc rộng (Wide); Góc siêu rộng (Ultrawide); HDR; Lấy nét theo pha (PDAF); Siêu cận (Macro); Toàn cảnh (Panorama); Xóa phông	Thẻ sim	2 SIM (Nano-SIM)
Camera trước		Hệ điều hành	Android 11, One UI 3.0
Thông số	20 MP, f/2.2	Công nghệ NFC	Có
Video	20 MP, f/2.2	Hỗ trợ mạng	4G
CPU		WiFi	Wi-Fi 802.11 a/b/g/n/ac, dual-band, Wi-Fi Direct, hotspot
		Bluetooth	5.0, A2DP, LE
		GPS	Có, hỗ trợ A-GPS, GLONASS

Hình 23: Thêm bình luận nếu người dùng đã có tài khoản thành viên

thoailc

★★★★★

Sản phẩm đẹp, chất lượng cap

Rate Sản phẩm đẹp, chất lượng cap



4.5 Trang tin tức

Hình 24: Tổng hợp tin tức

The screenshot shows a news feed with three articles. Each article has a small thumbnail image of a smartphone with the word 'TINH' on its screen. The first article is titled 'iOS 16 có thật sự tốn pin điện thoại không? Và cách hạn chế hao pin iOS 16 mà bạn nên biết' with a timestamp of 'Ngày đăng: 2022-06-14 21:09:04'. The second article is titled 'Bật mí cách tạo avatar 3D trên Instagram cực đơn giản và thú vị' with a timestamp of 'Ngày đăng: 2022-06-14 21:08:18'. The third article is titled 'WWDC 2022: Apple mang đến những gì? Có đáng để chờ đợi?' with a timestamp of 'Ngày đăng: 2022-06-14 20:55:01'. At the bottom of the feed, there is a blue button labeled 'Show more'.

Hình 25: Cụ thể một tin tức

iOS 16 có thật sự tốn pin điện thoại không? Và cách hạn chế hao pin iOS 16 mà bạn nên biết

iOS 16 là hệ điều hành mới nhất của **Apple** và được nhiều người dùng bàn luận cũng như đón nhận vì nhiều tính năng mới được nâng cấp và cải tiến. Vậy liệu sự hao pin iOS 16 có thực sự xuất hiện, cùng tìm hiểu với Sforum ngay bên dưới nhé.

Tham khảo ngay những cách tiết kiệm pin trên iPhone ngay bây giờ.





4.6 Trang cá nhân của khách hàng

Hình 26: Thông tin cá nhân

The screenshot shows a user profile page with the following fields:

- First Name: [empty input]
- Last Name: [empty input]
- Email: thaihuy@gmail.com
- Contacts Number: [empty input]
- Address: [empty input]
- District: [empty input]
- City: [empty input]

A blue "Save" button is located at the bottom left of the form area.

Hình 27: Đơn đặt hàng

The screenshot shows an order placement page with the following details:

Item	Quantity	Price
iPhone 12 Mini	1	709
iPhone 13 Mini	1	750



4.7 Trang cá nhân của admin

Hình 28: Thông tin cá nhân

The screenshot shows a user profile form. At the top, there are links for Samsung, Apple, Oppo, Sony, Xiaomi, Vivo, and Realme. Below that, tabs for Profile, Products, Post, and Users are present, with Profile being the active tab. On the left is a circular profile picture placeholder with a red robot icon. The form fields include:

- First Name: [empty input]
- Last Name: [empty input]
- Email: thoaile@gmail.com
- Contacts Number: [empty input]
- Address: [empty input]
- District: [empty input]
- City: [empty input]

A blue "Save" button is located at the bottom left of the form.

Hình 29: Quản lý sản phẩm

The screenshot shows a list of products. At the top, tabs for Profile, Products, Post, and Users are shown, with Products being the active tab. A blue "Add product" button is visible. The table below lists three products:

ID	Product name	Picture	Created	Modified	Action
6	Samsung Galaxy A32		2022-06-12 17:05:08	2022-06-12 17:05:08	
7	Samsung Galaxy S20 FE		2022-06-12 17:14:43	2022-06-12 17:14:43	
8	Samsung Galaxy S21 FE		2022-06-12 17:25:32	2022-06-12 17:25:32	



Hình 30: Quản lý bài đăng

The screenshot shows a table with four columns: ID, Post title, Created, and Action. There is one row of data with ID 1, Post title 'WWDC 2022: Apple mang đến những gì? Có đáng để chờ đợi?', Created '2022-06-14 20:55:01', and Action buttons (trash and edit).

ID	Post title	Created	Action
1	WWDC 2022: Apple mang đến những gì? Có đáng để chờ đợi?	2022-06-14 20:55:01	

Hình 31: Quản lý tài khoản người dùng

The screenshot shows a table with six columns: ID, Avatar, Username, Email, Mod, and Action. There are three rows of data with IDs 2, 3, and 4, each having a unique avatar and email address. The 'Mod' column contains icons for a crown (yellow), person (green), and person with a crown (blue). The 'Action' column contains edit and delete icons.

ID	Avatar	Username	Email	Mod	Action
2		thoaile	thoaile@gmail.com		
3		tungkhung	tung@gmail.com		
4		huythai	thaihuy@gmail.com		

Free shipping
Orders 150K or more

Free return
Within 30 days

Get 20% off
For new member

We support
24/7 amazing service



4.8 Thêm, chỉnh sửa sản phẩm

Hình 32: Thêm sản phẩm mới

The screenshot shows a web-based form for adding a new product. At the top, there are tabs for different brands: Samsung, Apple, Oppo, Sony, Xiaomi, Vivo, and Realmi. Below these tabs are several input fields for general product information: Product Name, Price, Type, Brand, and a search bar labeled 'Type to search'. To the right of these fields is a section titled 'HIGHLIGHT FEATURE' containing four input fields for 'Highlight feature 1' through 'Highlight feature 4'. Further down the page are sections for 'SCREEN' (with fields for Size, Resolution, Refresh rate, and a search bar) and 'BACK CAMERA' (with fields for Tech, Video, and Feature). A 'GENERAL' section is also present on the left side of the page.

Hình 33: Thêm sản phẩm mới

This screenshot shows a detailed product addition form for a Samsung Galaxy A32. The 'GENERAL' section contains fields for Product Name (Samsung Galaxy A32), Price (800), Review Point (4), Description (a note about the phone's features), Type (Smartphone), Brand (Samsung), and a search bar. The 'SCREEN' section includes fields for Size (6), Resolution (1080 x 2400 pixels), Refresh rate (90Hz), and a Tech search bar. The 'BACK CAMERA' section includes fields for Tech, Video, and Feature, with specific camera details listed. The 'HIGHLIGHT FEATURE' section lists four features: 'Màn hình siêu mượt, hình ảnh sắc nét', 'Thiết kế tối giản, thời thượng', 'Thời lượng pin ấn tượng', and 'Camera góc rộng: 64 MP, f/1.8, 26mm, PDAF; Camera góc siêu rộng: 8 MP, f/2.2, 123, 1/4.0, 1.12µm'.

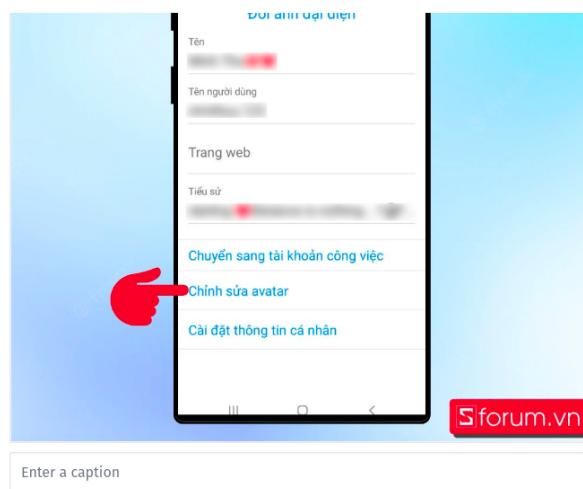


4.9 Thêm, chỉnh sửa bài đăng

Hình 34: Thêm bài đăng mới



Hình 35: Chỉnh sửa bài đăng



Tạm kết

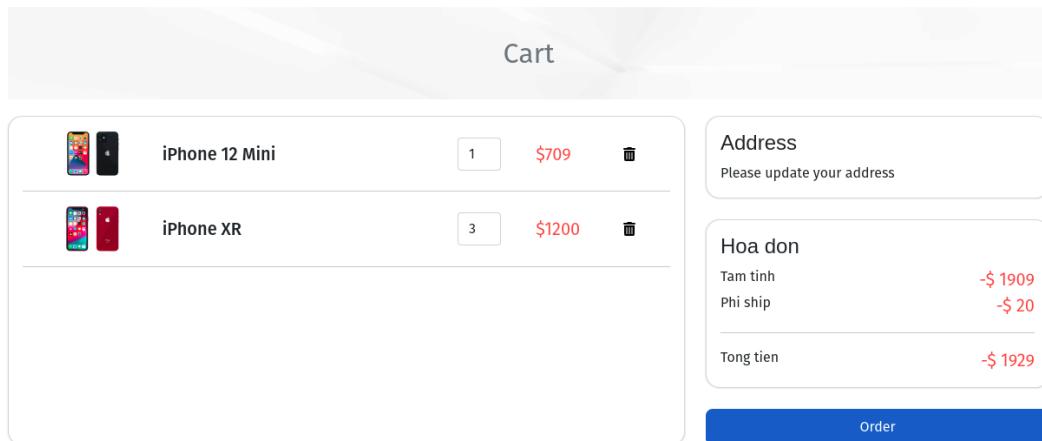
- + :: Như vậy Sforum đã giúp bạn biết được cách tạo avatar 3D trên Instagram cực đơn giản và độc đáo. Chúc các bạn tạo ra được những avatar thật sinh động và đáng yêu.

Submit



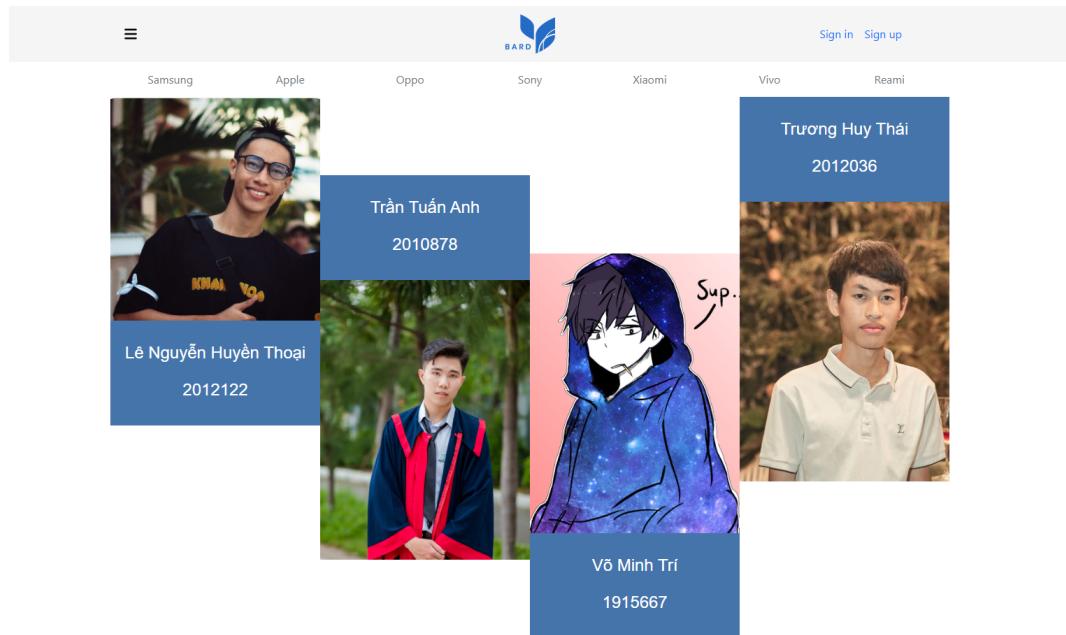
4.10 Giỏ hàng

Hình 36: Giỏ hàng



4.11 Giới thiệu

Hình 37: Thông tin nhóm





5 Cách thức cài đặt ứng dụng, môi trường chạy ứng dụng

5.1 Thông tin phiên bản

- React: V18.0
- NodeJS: v17.5.0
- PHP: 7.4.27 trở lên
- phpMyAdmin: 5.1.1

5.2 Link tải source code

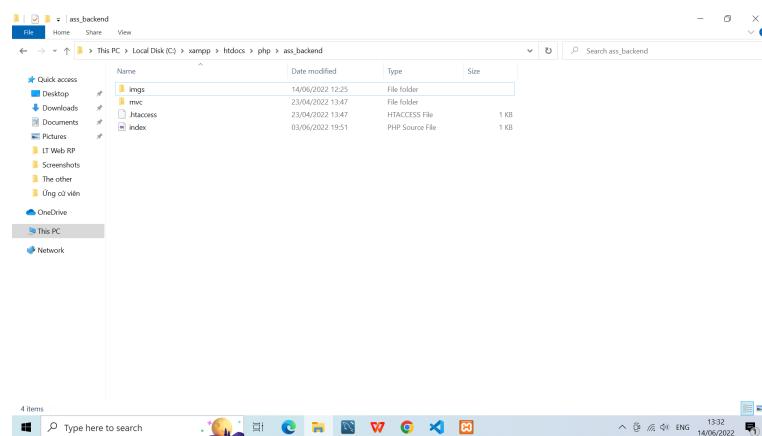
- Link github front-end: https://github.com/thoaiKun/WebAss_Reactjs
- Link github back-end: https://github.com/trtuananh/phonestop_backend_mvc

5.3 Các bước cài đặt

Phía front-end của ứng dụng sử dụng framework reactjs, vì vậy để có thể cài đặt ứng dụng cần phải cài đặt Nodejs trước. Sau khi đã cài đặt xong Node, thực hiện theo các bước sau để có thể cài đặt ứng dụng:

- Bước 1: Tải xuống các thư mục mã nguồn của front-end và back-end theo các link github ở trên
- Bước 2: Đổi tên lại thư mục mã nguồn back-end từ phonestop_backend_mvc thành ass_backend, sau đó vào thư mục htdocs trong xampp, tạo một thư mục mới tên là "php", sau đó bỏ mã nguồn vào thư mục php

Hình 38: Đường dẫn đúng sau khi đã tải xuống code





Khá là quan trọng việc tên và vị trí các thư mục đúng như mô tả ở trên, nếu không thì phía front-end không thể gọi tới các api ở localhost được

- Bước 3: Vào file mã nguồn của phía back-end theo đường dẫn ass_backend/mvc/core/Model.php, sau đó thay đổi các thông số server của hệ QTCSQL đúng như máy của mình.

Hình 39: Nơi chứa các tùy chỉnh về database

```
File Edit Selection View Go Run Terminal Help
EXPLORER ... Model.php M x
ASS_BACKEND mvc > core > Model.php
  1 <?php
  2
  3
  4     protected $servername = "localhost";
  5     protected $username = "root";
  6     protected $password = "tisshop1";
  7     protected $db_name = "phoneshop";
  8     protected $conn;
  9
 10    public function __construct()
 11    {
 12        $this->conn = mysqli_connect($this->servername, $this->username, $this->password, $this->db_name);
 13        // Check connection
 14        if ($this->conn) {
 15            echo "Connected";
 16        } else {
 17            die("Connection failed: " . mysqli_connect_error());
 18        }
 19    }
 20
 21    public function dbConnection()
 22    {
 23        $conn = mysqli_connect($this->servername, $this->username, $this->password, $this->db_name);
 24        // Check connection
 25        if ($conn) {
 26            echo "Connected";
 27        } else {
 28            die("Connection failed: " . mysqli_connect_error());
 29        }
 30        return $conn;
 31    }
}
In 7 Col 38 Spaces:4 UTF-8 CR LF IP R D
Type here to search
13:37 14/06/2022
```

- Bước 4: Vào server của MySQL đã chọn ở bước trên và chạy file MySQL script nằm ở đường dẫn ass_backend/mvc/core/phoneshop_init.sql. Việc này sẽ tạo ra schema và thêm vào các dữ liệu khởi tạo của ứng dụng.

Hình 40: File phoneshop_init.sql

```
File Edit Selection View Go Run Terminal Help
EXPLORER ... phoneshop_init.sql X
ASS_BACKEND mvc > core > phoneshop_init.sql
  1 SET GLOBAL max_allowed_packet=10485760;
  2
  3 drop schema if exists phoneshop;
  4 create schema phoneshop;
  5 use phoneshop;
  6
  7 create table users (
  8     id int primary key auto_increment,
  9     email varchar(30),
 10    password varchar(100),
 11    username varchar(20),
 12    first_name varchar(30),
 13    last_name varchar(30),
 14    contact_number varchar(30),
 15    address text,
 16    district text,
 17    city text,
 18    note blob,
 19    profile_img varchar(200)
 20 );
 21
 22 create table products (
 23     id int primary key auto_increment,
 24     product_name text,
 25     price bigint,
 26     image varchar(200),
 27     type varchar(100),
 28     brand varchar(100),
 29     hf_1 text,
 30     hf_2 text,
 31     hf_3 text,
 32 );
}
In 10 Col 29 Spaces:4 UTF-8 CR LF IP R D
Type here to search
13:34 14/06/2022
```

- Bước 5: Về phía front-end, chúng ta cần phải cài đặt các module cần thiết của react. Vào terminal của thư mục WebAss_Reactjs, sau đó chạy lệnh npm install.



Hình 41: Terminal sau khi đã chạy lệnh

```
File Edit Selection View Go Run Terminal Help
EXPLORER ... UserContext.js ...
src > Contexts > # UserContext.js > #0 UserProvider
  1 import React from 'react';
  2 import { Context } from 'react';
  3 import { CookiesProvider, useCookies } from 'react-cookie';
  4
  5 const UserContext = React.createContext();
  6
  7 export const UserProvider = ({ children }) => {
  8   const [userCookies, setuserCookies, removeUserCookies] = useCookies('user');
  9   const [login, setLogin] = React.useState(userCookies?.login ? userCookies.login : false);
 10   const [token, setToken] = React.useState(userCookies?.token ? userCookies.token : '');
 11   const [cart, setCart] = React.useState(userCookies?.cart ? userCookies.cart : []);
 12
 13   return (
 14     <UserContext.Provider value={{ userCookies, setuserCookies, removeUserCookies, login, setLogin, token, setToken, cart, setCart }}>{children}</UserContext.Provider>
 15   );
 16 }
 17
 18 export default UserProvider;
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

```
Windows PowerShell
Copyright © Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

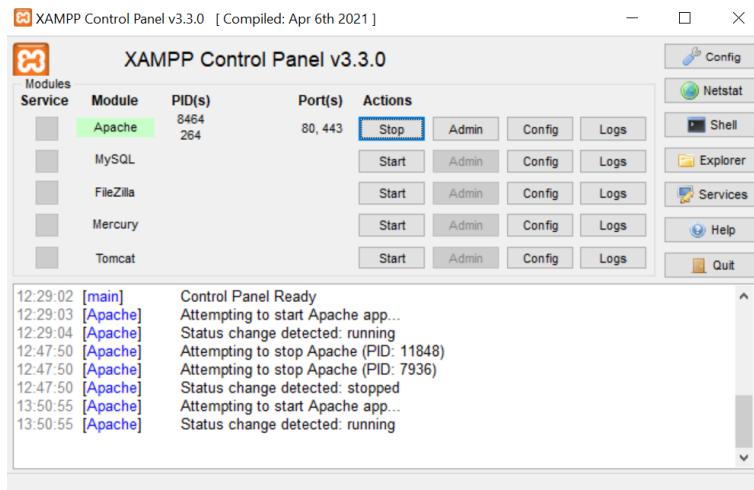
PS C:\Users\ADMIN\Desktop\Frontend\WebAss_Reactjs> npm install
npm WARN deprecated source-map-resolve@0.6.0: See https://github.com/lydell/source-map-resolve#deprecated
npm WARN deprecated source-map-resolve@0.5.3: This module is no longer supported. Upgrade to v1.0.0
npm http fetch error: https://registry.npmjs.org/-/resolve/-/resolve-2.0.0-next.3.tgz 29984ms (cache miss)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Type here to search

- Bước 6: Bật control panel của xampp lên và khởi chạy Apache

Hình 42: Xampp control panel



- Bước 7: Lúc này phía server đã chuẩn bị xong, bây giờ ta quay lại terminal của thư mục WebAss_Reactjs và chạy lệnh npm start. Sau đó chỉ còn lại việc trải nghiệm ứng dụng.



Hình 43: Terminal sau khi chạy thành công npm start

```
File Edit Selection View Go Run Terminal Help
NewDetail.js - WebAss_Reactjs - Visual Studio Code

EXPLORER ... node_modules
> Components > Content > NewDetail > NewDetail.js
src > Components > Content > NewDetail > NewDetail.js >
1  import { useState } from 'react'
2  import { useParams } from 'react-router'
3  import useFetchData from '../../../../../Hooks/useFetchData'
4  import EditorJS from '@editorjs/editorjs'
5  import List from '@editorjs/list'
6  import Image from '@editorjs/image'
7  import SimpleImage from '@editorjs/simple-image'
8  import Header from '@editorjs/header'
9  import './NewDetail.css'
10
11 const NewDetail = () => {
  PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPITER
  Compiled successfully!
  You can now view assignment in the browser.
  Local: http://localhost:3000
  On Your Network: http://192.168.1.10:3000
  Note that the development build is not optimized.
  To create a production build, use npm run build.
  assets by path static/media/... 1.86 MB
  static/media/TA_79ced655040d26480b0.jpg 243 KiB [emitted] [immutable] [from: public/img/member/TA.jpg] (auxiliary name: main)
  static/media/banner4_800c8891fc8676ef9105.jpg 178 KiB [emitted] [immutable] [From: public/img/banner/banner4.jpg] (auxiliary name: main)
  e: member/assignment/assignment3ande.jpg 144 KiB [emitted] [immutable] [from: public/img/member/assignment3ande.jpg] (auxiliary name: main)
  n) asset static/media/ThoiTai_800f79678d48ea3ande.jpg 128 KiB [emitted] [immutable] [from: public/img/member/ThoiTai.jpg] (auxiliary name: main)
  asset static/media/ThoiTai_4b84c4b2eac4270e960b.jpg 98 KiB [emitted] [immutable] [From: public/img/member/ThoiTai.jpg] (auxiliary name: main)
  in)
  In 5, Col 34 Spaces: 4 UFT-8 CRLF JavaScript AR Q
  node + v x
  Search: Search
  Timeline
  In 5, Col 34 Spaces: 4 UFT-8 CRLF JavaScript AR Q
  1355
  ENG
  14/06/2022
```

- Lưu ý: Nếu như gặp lỗi như hình dưới sau khi chạy lệnh npm start, thì quay lại terminal, chạy lệnh npm i –save @editorjs/list, sau đó thực hiện lại bước 7

Hình 44: Lỗi có thể xảy ra vì npm install cài đặt không đủ module

```
File Edit Selection View Go Run Terminal Help
My Bach Khoa | Trang chủ | CO3050_0 | Báo cáo VN | thoikuniv | BardShop | npm-audit | Module m... | +
localhost:3000
Compiled with problems:
ERROR in ./src/Components/Content/AddNew/AddNew.js 12:0:34
Module not found: Error: Can't resolve '@editorjs/list' in
'C:\Users\ADMIN\Desktop\Frontend\WebAss_Reactjs\src\Components\Content\AddNew'

ERROR in ./src/Components/Content/EditNew/EditNew.js 12:0:34
Module not found: Error: Can't resolve '@editorjs/list' in
'C:\Users\ADMIN\Desktop\Frontend\WebAss_Reactjs\src\Components\Content>EditNew'

ERROR in ./src/Components/Content/NewDetail/NewDetail.js 10:0:34
Module not found: Error: Can't resolve '@editorjs/list' in
'C:\Users\ADMIN\Desktop\Frontend\WebAss_Reactjs\src\Components\Content>NewDetail'

Type here to search
12:34 1355
ENG
14/06/2022
```



6 Nhiệm vụ vai trò của thành viên trong nhóm

Tên	MSSV	Phân công	Tỷ lệ đóng góp
Lê Nguyễn Huyền Thoại	2012122	Thiết kế giao diện Làm front-end	100%
Trương Huy Thái	2012036	Làm front-end Viết báo cáo	100%
Trần Tuấn Anh	2010878	Thiết kế cơ sở dữ liệu Làm backend	100%
Võ Minh Trí	1915667	Thiết kế cơ sở dữ liệu Làm backend	100%



Tài liệu

- [1] *Slide bài giảng*, Nguyễn Hữu Hiếu
- [2] *ReactJS - Ưu điểm và nhược điểm*, Hoàng Trọng Huy
- [3] *ReactJS document*, Facebook
- [4] *EditorJS document*, EditorJS
- [5] *How to create React JS + PHP Login Registration System?*, w3jar.com
- [6] *LẬP TRÌNH PHP THEO MÔ HÌNH MVC [BUỔI 1]*, Trung tâm đào tạo công nghệ Khoa Phạm