REACT ROUTER

Vd1

Home.js

import React from 'react';

function Home() {

return <h1>Home Page</h1>;

}

export default Home;

About.js

import React from 'react';

function About() {

return <h1>About Page</h1>;

}

export default About;

Navigation.js

import React from 'react';

import { Navbar, Nav, Container } from 'react-bootstrap';

import { Link } from 'react-router-dom';

function Navigation() {

return (

<Navbar bg="dark" variant="dark">

<Container>

<Navbar.Brand as={Link} to="/">React App</Navbar.Brand>

<Nav className="me-auto">

<Nav.Link as={Link} to="/">Home</Nav.Link>

<Nav.Link as={Link} to="/about">About</Nav.Link>

<Nav.Link as={Link} to="/posts">Posts</Nav.Link>

</Nav>

</Container>

</Navbar>

);

}

export default Navigation;

Post.js

import React from 'react';

import { Link } from 'react-router-dom';

function Post() {

return (

<div>

<h2>Post List</h2>

<ul>

<li><Link to="/post/1">Post 1</Link></li>

<li><Link to="/post/2">Post 2</Link></li>

<li><Link to="/post/3">Post 3</Link></li>

</ul>

</div>

);

}

export default Post;

PostDetail.js

import React from 'react';

import { useParams } from 'react-router-dom';

function PostDetail() {

const { id } = useParams(); // Lấy tham số ID từ URL

return (

<div>

<h2>Post Detail for Post ID: {id}</h2>

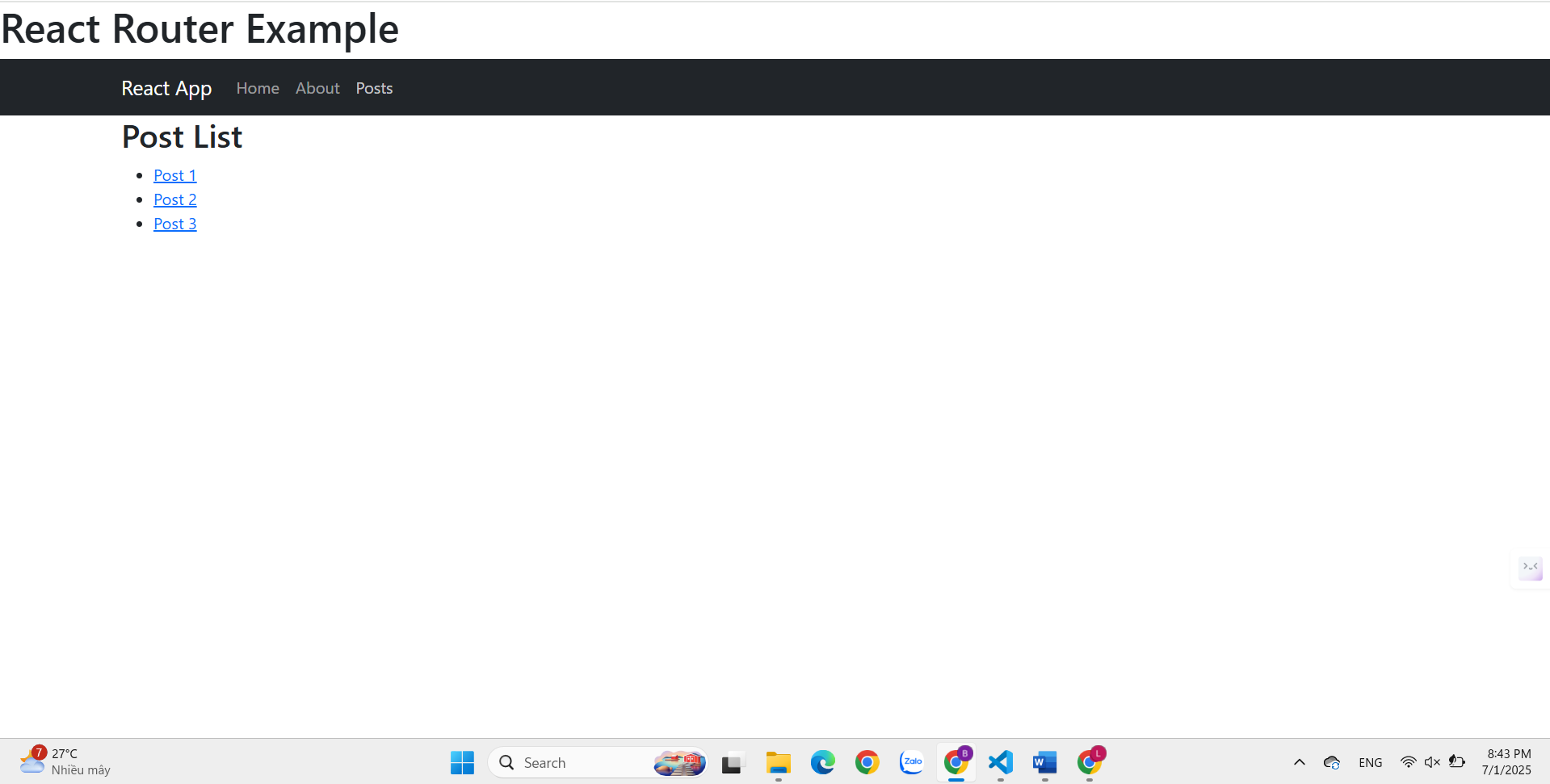
<p>This is the detailed information for post {id}.</p>

</div>

);

}

export default PostDetail;

Kết quả   


Ví Dụ 2:

Tạo file public/posts.json

[

{

"id": "1",

"title": "React Router Basics",

"content": "This post explains the basics of React Router in a React application."

},

{

"id": "2",

"title": "React Bootstrap Introduction",

"content": "This post introduces how to use React Bootstrap to create beautiful UIs."

},

{

"id": "3",

"title": "JavaScript ES6 Features",

"content": "This post discusses the new features introduced in JavaScript ES6."

}

]

Cập nhật file src/components/Post.js

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

import { Link } from 'react-router-dom';

function Post() {

const [posts, setPosts] = useState([]);

useEffect(() => {

fetch('/posts.json')

.then((response) => {

// Kiểm tra nếu phản hồi không hợp lệ (status khác 200)

if (!response.ok) {

throw new Error('Network response was not ok');

}

return response.json();

})

.then((data) => setPosts(data))

.catch((error) => {

console.error('Error loading posts:', error);

// Bạn có thể hiển thị lỗi cho người dùng nếu muốn

});

}, []);

return (

<div>

<h2>Post List</h2>

<ul>

{posts.map(post => (

<li key={post.id}>

<Link to={`/post/${post.id}`}>{post.title}</Link>

</li>

))}

</ul>

</div>

);

}

export default Post;

Cập nhật file src/components/PostDetail.js

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

import { useParams } from 'react-router-dom';

function PostDetail() {

const { id } = useParams(); // Lấy tham số ID từ URL

const [post, setPost] = useState(null);

useEffect(() => {

// Fetch dữ liệu từ file posts.json trong thư mục public

fetch('/posts.json')

.then((response) => response.json())

.then((data) => {

// Tìm bài viết theo ID

const foundPost = data.find(p => p.id === id);

setPost(foundPost);

})

.catch((error) => console.error('Error loading post:', error));

}, [id]); // Mỗi khi ID thay đổi, gọi lại API

if (!post) {

return <h2>Post not found</h2>;

}

return (

<div>

<h2>{post.title}</h2>

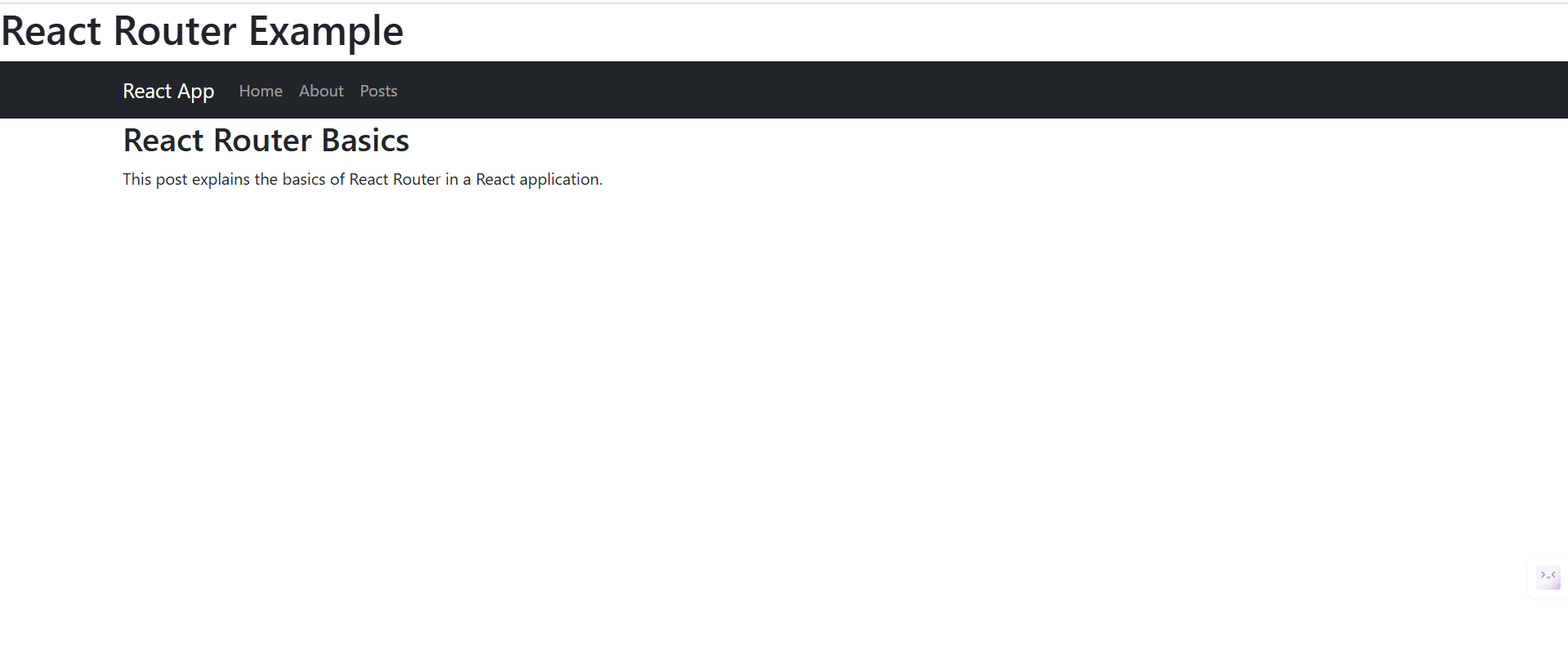
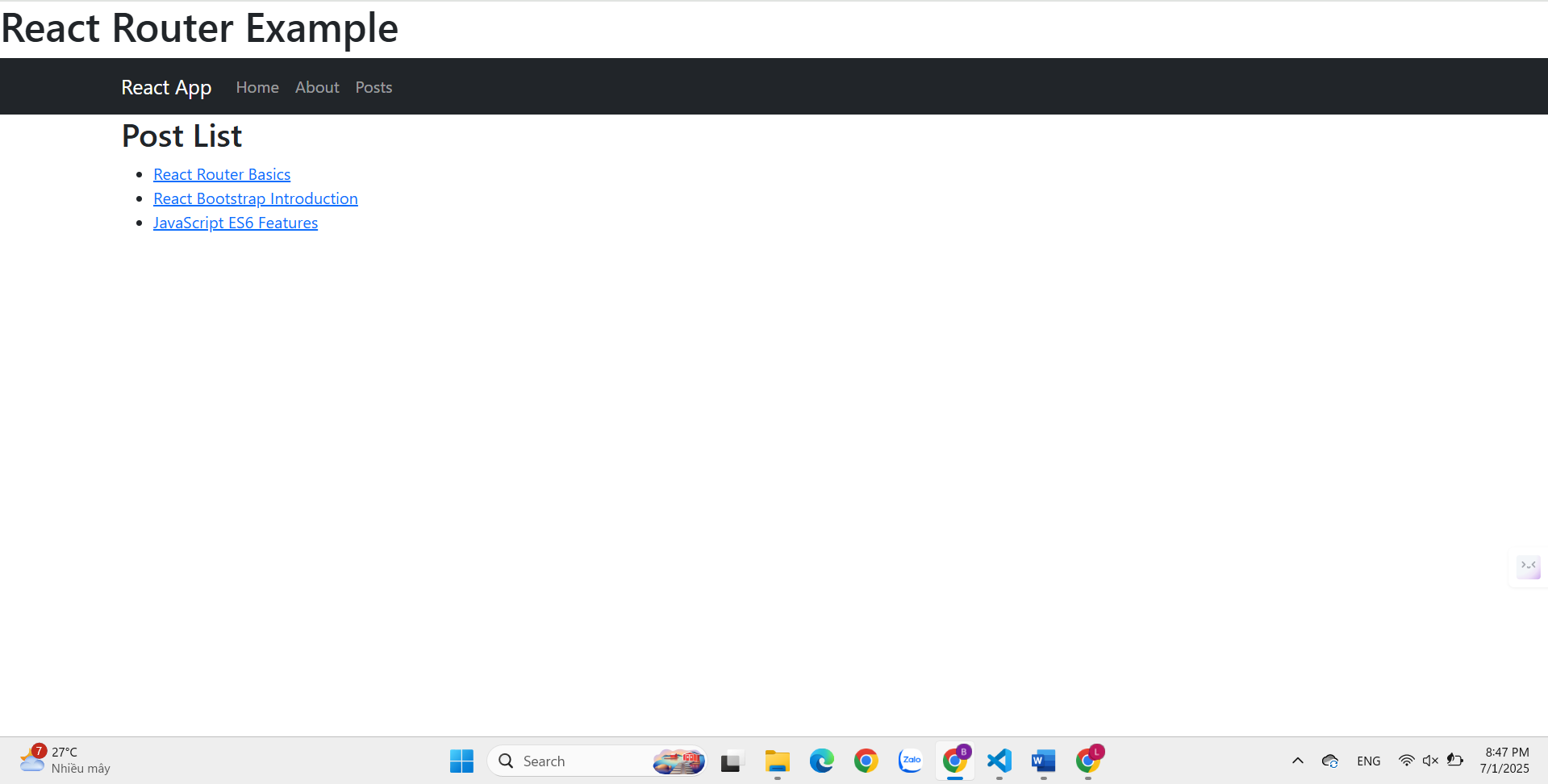
<p>{post.content}</p>

</div>

);

}

export default PostDetail;



Ví dụ 3:

Tạo file src/components/Login.js

import React, { useState } from 'react';

import { Form, Button, Alert, Container, Row, Col } from 'react-bootstrap';

import { useNavigate } from 'react-router-dom';

function Login({ setIsAuthenticated }) {

const [username, setUsername] = useState('');

const [password, setPassword] = useState('');

const [error, setError] = useState('');

const navigate = useNavigate();

const handleSubmit = (e) => {

e.preventDefault();

// Validation đơn giản

if (username === 'admin' && password === 'password') {

setIsAuthenticated(true);

localStorage.setItem('isAuthenticated', 'true');

navigate('/posts');

} else {

setError('Invalid username or password');

}

};

return (

<Container>

<Row className="justify-content-center">

<Col md={6}>

<h2>Login</h2>

{error && <Alert variant="danger">{error}</Alert>}

<Form onSubmit={handleSubmit}>

<Form.Group className="mb-3">

<Form.Label>Username</Form.Label>

<Form.Control

type="text"

value={username}

onChange={(e) => setUsername(e.target.value)}

placeholder="Enter username (admin)"

required

/>

</Form.Group>

<Form.Group className="mb-3">

<Form.Label>Password</Form.Label>

<Form.Control

type="password"

value={password}

onChange={(e) => setPassword(e.target.value)}

placeholder="Enter password (password)"

required

/>

</Form.Group>

<Button variant="primary" type="submit">

Login

</Button>

</Form>

</Col>

</Row>

</Container>

);

}

export default Login;

Src/components/ProtectedRoute.js

import React from 'react';

import { Navigate } from 'react-router-dom';

function ProtectedRoute({ children, isAuthenticated }) {

return isAuthenticated ? children : <Navigate to="/login" />;

}

export default ProtectedRoute;

Cập nhật Navigation.js để có logout

import React from 'react';

import { Navbar, Nav, Container, Button } from 'react-bootstrap';

import { Link, useNavigate } from 'react-router-dom';

function Navigation({ isAuthenticated, setIsAuthenticated }) {

const navigate = useNavigate();

const handleLogout = () => {

setIsAuthenticated(false);

localStorage.removeItem('isAuthenticated');

navigate('/');

};

return (

<Navbar bg="dark" variant="dark">

<Container>

<Navbar.Brand as={Link} to="/">React App</Navbar.Brand>

<Nav className="me-auto">

<Nav.Link as={Link} to="/">Home</Nav.Link>

<Nav.Link as={Link} to="/about">About</Nav.Link>

{isAuthenticated && <Nav.Link as={Link} to="/posts">Posts</Nav.Link>}

</Nav>

<Nav>

{!isAuthenticated ? (

<Nav.Link as={Link} to="/login">Login</Nav.Link>

) : (

<Button variant="outline-light" onClick={handleLogout}>

Logout

</Button>

)}

</Nav>

</Container>

</Navbar>

);

}

export default Navigation;

Cập nhật App.js để xử lý authentication

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

import { Routes, Route } from 'react-router-dom';

import Home from './components/Home';

import About from './components/About';

import Navigation from './components/Navigation';

import Login from './components/Login';

import ProtectedRoute from './components/ProtectedRoute';

import { Container } from 'react-bootstrap';

import Post from './components/Post';

import PostDetail from './components/PostDetail';

function App() {

const [isAuthenticated, setIsAuthenticated] = useState(false);

useEffect(() => {

const authStatus = localStorage.getItem('isAuthenticated');

if (authStatus === 'true') {

setIsAuthenticated(true);

}

}, []);

return (

<div>

<h1>React Router Example with Authentication</h1>

<Navigation isAuthenticated={isAuthenticated} setIsAuthenticated={setIsAuthenticated} />

<Container>

<Routes>

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

<Route path="/about" element={<About />} />

<Route path="/login" element={<Login setIsAuthenticated={setIsAuthenticated} />} />

<Route

path="/posts"

element={

<ProtectedRoute isAuthenticated={isAuthenticated}>

<Post />

</ProtectedRoute>

}

/>

<Route

path="/post/:id"

element={

<ProtectedRoute isAuthenticated={isAuthenticated}>

<PostDetail />

</ProtectedRoute>

}

/>

</Routes>

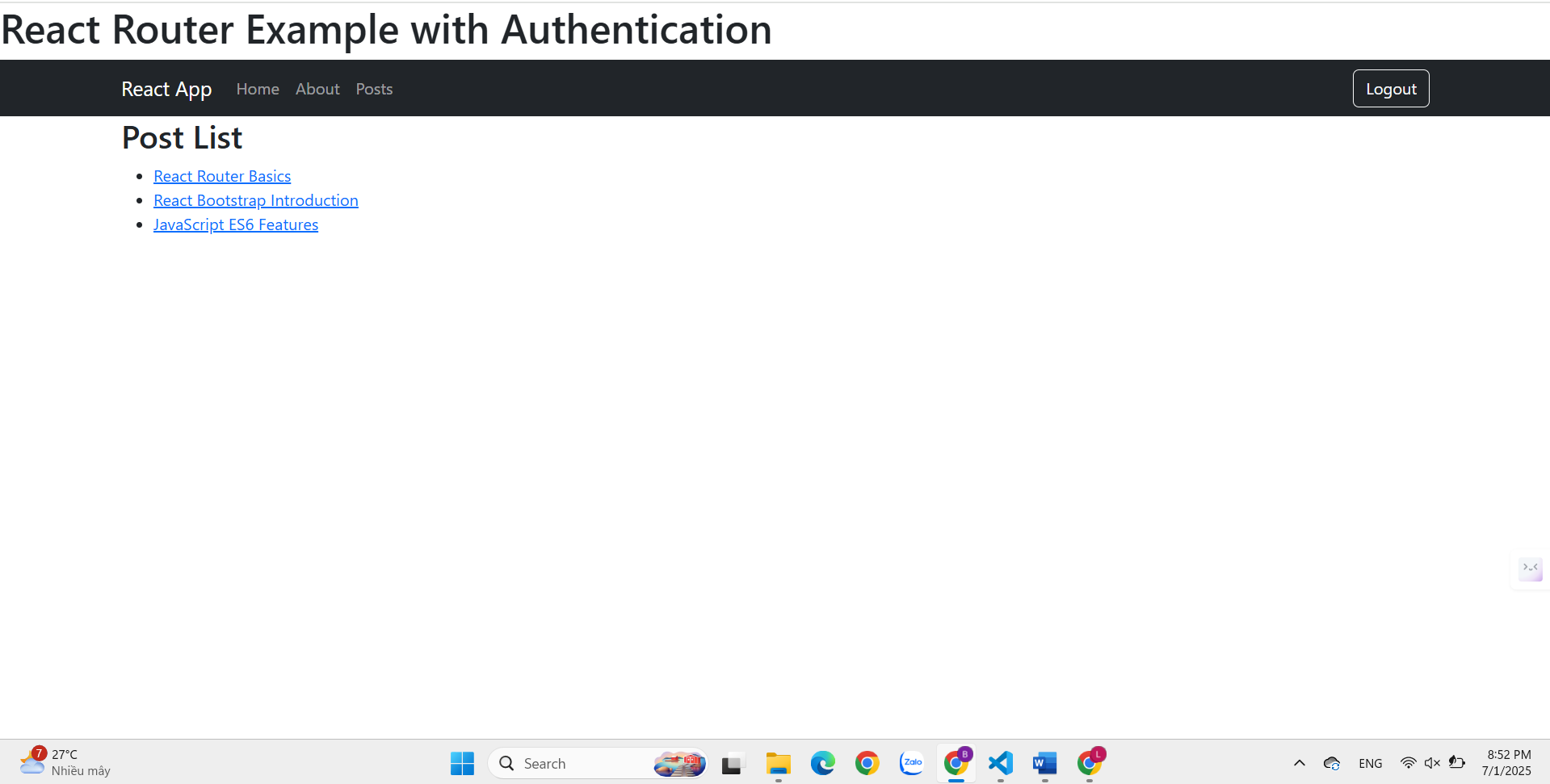
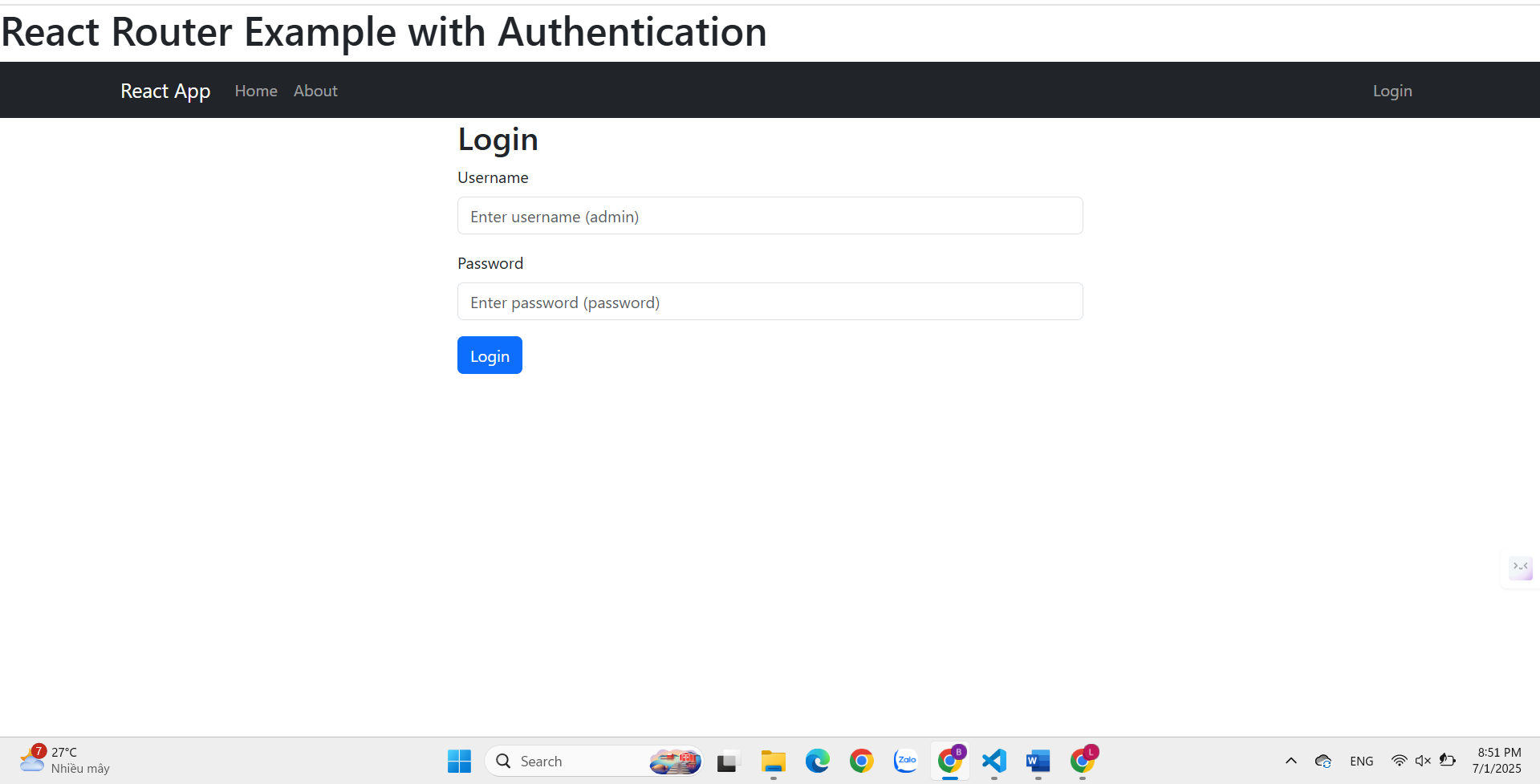
</Container>

</div>

);

}

export default App;



Ví dụ 4:

Tạo file db.json

{

"posts": [

{

"id": 1,

"title": "React Router with JSON Server",

"content": "This post explains how to use React Router with JSON Server for API simulation.",

"author": "John Doe",

"date": "2024-01-15"

},

{

"id": 2,

"title": "Advanced React Patterns",

"content": "This post discusses advanced patterns in React development including hooks and context.",

"author": "Jane Smith",

"date": "2024-01-20"

},

{

"id": 3,

"title": "State Management in React",

"content": "This post covers different approaches to state management in React applications.",

"author": "Bob Johnson",

"date": "2024-01-25"

}

],

"users": [

{

"id": 1,

"username": "admin",

"password": "password",

"email": "[admin@example.com](mailto:admin@example.com)"

}

]

}

Cập nhật package.json

{

"scripts": {

"start": "react-scripts start",

"build": "react-scripts build",

"test": "react-scripts test",

"eject": "react-scripts eject",

"json-server": "json-server --watch db.json --port 3001"

}

}

Cập nhật Post.js để sử dụng JSON Server

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

import { Link } from 'react-router-dom';

import { Card, Row, Col, Spinner } from 'react-bootstrap';

function Post() {

const [posts, setPosts] = useState([]);

const [loading, setLoading] = useState(true);

useEffect(() => {

fetch('http://localhost:3001/posts')

.then((response) => {

if (!response.ok) {

throw new Error('Network response was not ok');

}

return response.json();

})

.then((data) => {

setPosts(data);

setLoading(false);

})

.catch((error) => {

console.error('Error loading posts:', error);

setLoading(false);

});

}, []);

if (loading) {

return (

<div className="text-center">

<Spinner animation="border" role="status">

<span className="visually-hidden">Loading...</span>

</Spinner>

</div>

);

}

return (

<div>

<h2>Post List (From JSON Server)</h2>

<Row>

{posts.map(post => (

<Col md={4} key={post.id} className="mb-3">

<Card>

<Card.Body>

<Card.Title>{post.title}</Card.Title>

<Card.Text>

By: {post.author} | Date: {post.date}

</Card.Text>

<Link to={`/post/${post.id}`} className="btn btn-primary">

Read More

</Link>

</Card.Body>

</Card>

</Col>

))}

</Row>

</div>

);

}

export default Post;

Cập nhật PostDetail.js để sử dụng JSON Server

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

import { useParams, useNavigate } from 'react-router-dom';

import { Card, Button, Spinner } from 'react-bootstrap';

function PostDetail() {

const { id } = useParams();

const navigate = useNavigate();

const [post, setPost] = useState(null);

const [loading, setLoading] = useState(true);

useEffect(() => {

fetch(`http://localhost:3001/posts/${id}`)

.then((response) => {

if (!response.ok) {

throw new Error('Post not found');

}

return response.json();

})

.then((data) => {

setPost(data);

setLoading(false);

})

.catch((error) => {

console.error('Error loading post:', error);

setLoading(false);

});

}, [id]);

if (loading) {

return (

<div className="text-center">

<Spinner animation="border" role="status">

<span className="visually-hidden">Loading...</span>

</Spinner>

</div>

);

}

if (!post) {

return (

<div>

<h2>Post not found</h2>

<Button onClick={() => navigate('/posts')}>

Back to Posts

</Button>

</div>

);

}

return (

<div>

<Button

variant="secondary"

onClick={() => navigate('/posts')}

className="mb-3"

>

← Back to Posts

</Button>

<Card>

<Card.Body>

<Card.Title>{post.title}</Card.Title>

<Card.Subtitle className="mb-2 text-muted">

By: {post.author} | Date: {post.date}

</Card.Subtitle>

<Card.Text>{post.content}</Card.Text>

</Card.Body>

</Card>

</div>

);

}

export default PostDetail;