Cindy Pham

Kane

CIT 313

12/1/24

Portfolio React Website

My project is a portfolio website built using React. It showcases various projects and showcases the skills and technologies I have mastered. The site is designed with a focus on responsiveness, performance, and reusability.

Key Features

- **React Components**: The site is built using React components that break down the interface into reusable sections for better maintainability and organization.
- **Responsiveness**: The design of the website adapts to various screen sizes.
- **Performance**: Careful attention has been given to the performance of the application by using performant components and managing props

Components

The application is structured around multiple React components that make it dynamic and reusable. The key components are:

1. **Header Component**: Displays the site's title and navigation menu. It is used across all pages.
2. **Navigation Bar Component:** Allows routing to all pages and includes hamburger menu.
3. **Footer Component**: A static footer that appears at the bottom of each page.

4. **Project Cards Component**: This component is used to display individual project information. It helps in organizing project details into a card style layout
5. **Carousel Component**: Displays featured images of projects in a slideshow format and adds dynamic visual to the homepage.
6. **Accordion Component**: Used to display information like skills, technologies, and development tools.
7.

Responsiveness

The website is fully responsive, ensuring that the layout adjusts gracefully across various devices. Key techniques used for responsiveness include:

- **CSS Flexbox and Grid**
- **Media Queries**
- **Tailwind CSS**

Performance

The performance of the website has been optimized by focusing on rendering and reducing unnecessary re-renders. Some of the strategies include:

- **Component Reusability**.
- **Efficient Use of Props**

React Hooks

The two most commonly used hooks in this project are:

1. **useState**: This hook is used to manage the component state. It helps in tracking dynamic data like the state of modal windows or the list of blog posts.

2. **useEffect**: This hook is used to perform side effects in the application, such as fetching data from an external API or loading JSON files. It runs after the component renders, making it ideal for tasks like data fetching or updating the DOM.

Conclusion

This project shows how useful using React is in building modular, responsive, and performant web applications. Using React components, responsive design principles, and performance optimization techniques, the website provides a smooth user experience. The effective use of React hooks like useState and useEffect helps in managing state and side effects, ensuring that the application runs efficiently. Overall, this portfolio website demonstrates my skills and my ability to create using React.