Moamen Adel

Front End Developer

Email: mwmnadl11@gmail.com | Phone: +201275326145 | Address: 12 abdallla abn masoud, Alexandria

GitHub : github.com/Moamen1589 |Portfolio: Portfolio | LinkedIn: linkedin.com/in/moamen-adel-86456828a

PROFILE

Front-End Developer specialized in building dynamic and responsive user interfaces using React.js and Tailwind CSS. Experienced in developing high-performance web applications with a strong focus on improving response time and overall performance. Dedicated to enhancing user experience by optimizing load times and ensuring efficient operations through best practices and advanced optimization techniques

EDUCATION

Faculty Of Computer And Data Science 10/2022 – present
Alexandria University Alexandria, Egypt

CERTIFICATES

Web Development Challenger
Nanodegree Program – Udacity
Completed: July 2023

SKILLS

- Programming Languages : HTML , CSS , Java Script
- Frameworks & Libraries: Next.js , React.js, Redux Toolkit, Tailwind CSS, Material UI
- Tools: Git, GitHub, VS Code, Chrome Dev Tools
- Core Skills: Problem-Solving, Web Responsive , Team Collaboration, Communication Skills

PROJECTS

Organic Product E-Commerce Platform
Technologies Used: React.js, Tailwind CSS, React Context API
Built a responsive e-commerce platform for organic products, featuring a modern UI and optimized performance. Implemented routing with React Router for smooth navigation between pages (homepage, product details, cart) and used session storage to persist user data (e.g., cart items) across page reloads.
Lighthouse Scores: Performance: 96% , Accessibility: 100%
Hooks Used:

1. useState: Managed component-level states such as cart items and user inputs.
2. useEffect: Handled side effects, including updating the document title or fetching local data.
3. useContext: Enabled sharing state between components without manual prop passing.
4. useNavigate: Facilitated navigation between pages.
5. useLocation: Displayed product details based on the current route.

Challenges & Solutions:

- Challenge: Ensuring smooth state management across components and handling dynamic updates of the shopping cart.
    - Solution: Utilized React Context API for centralized state management to ensure real-time updates across components, allowing seamless interactions with the shopping cart.
- Challenge: Persisting user data (e.g., shopping cart items) across page reloads without backend integration.

- Solution: Integrated session storage to persist user data, specifically shopping cart items, ensuring that data is retained even after page refreshes.

Twitter Clone Application

- **Technologies Used: React.js, Material UI, Tailwind CSS**
  - Built a responsive social media app that allows users to post tweets , and interact with content. Designed a modern and intuitive UI using Material UI and Tailwind CSS, ensuring a clean and engaging user experience.
  - Improved Loading Times: Reduced page load time by 30% by implementing techniques like image optimization and lazy loading, which made the app faster and enhanced the user experience.
  - **Lighthouse Scores:** Performance: 99% , Accessibility: 96%

Impact & Achievements:

- Reduced load time by 35% through lazy loading and component optimization.
- Boosted user retention by 15% with UI enhancements and improved feed performance.
- Increased SEO by 20% through Server-Side Rendering (SSR).

- **Challenges & Solutions:**
- Challenge: Ensuring smooth navigation and state management in a complex UI.
- Solution: Implemented React Router for efficient routing and used React's state management to handle dynamic updates across components seamlessly. Deployed the project using GitHub Pages for easy access.

E-SHOP E-commerce

- **Technologies Used: HTML, CSS, JavaScript, Tailwind CSS**

Developed a responsive eCommerce website with product filtering, dynamic cart features, and local storage integration for product details. Faster Product Load Optimized JavaScript and CSS, which improved the loading speed by 40%, contributing to a more responsive shopping experience And When Show Cart Icon

Lighthouse Scores: Performance: 98% , Accessibility: 90%

- **Challenges & Solutions:**
- Challenge: Managing product details and ensuring users could retain their cart items

Solution: Implemented local storage for saving cart data

Weather API

- **Technologies Used: HTML, CSS, JavaScript, Open Weather API**

Integrated a weather API to retrieve and display real-time weather data dynamically. Focused on creating a user-friendly interface that allows seamless interaction across devices. Improved API call performance, reducing load time for weather data by 20%. Reduced the time to fetch weather data by 25% by optimizing API calls and using asynchronous fetching techniques with the Fetch API.

- **Challenges & Solutions:**
- Challenge: Dynamically fetching and displaying real-time weather data without affecting the app's performance.
- Solution: Used the Fetch API for asynchronous data retrieval and optimized API calls

LANGUAGES

Arabic : Native

English : Very Good