Day 6: Functional and Non-Functional Requirements, Web Optimization, and Performance Tools

Contents Covered:

Functional and Non-Functional Requirements:

- **Functional Requirements:** Specific behaviors and functions that a system must perform. Examples include user authentication, data processing, and transaction handling.

- **Non-Functional Requirements:** Attributes that define the quality and performance of the system. These include:

    - **Performance:** Speed and responsiveness of the web application.
    - **Reliability:** Consistency in performance and availability.
    - **Security:** Protection against threats and vulnerabilities.
    - **Usability:** Ease of use and user experience.

Web Page Optimization:

- Discussed the importance of optimizing web pages according to standards to enhance performance and improve indexing on Google.
- Techniques for improving code to meet these standards, including reducing file sizes, enhancing load times, and ensuring accessibility.

Tools for Web Optimization and Performance:

- **Google PageSpeed Insights :** A tool for analyzing and improving web page performance.
- **Lighthouse:** An open-source, automated tool providing audits for performance, accessibility, progressive web apps, SEO, and best practices.
- **GTmetrix:** Provides insights and recommendations for improving site speed and performance.
- **Minify CSS/JS:** Techniques and tools to reduce the size of CSS and JavaScript files for faster loading times.
- **Image Optimization:** Using formats like JPG or WebP for efficient image loading.
- **NVDA Screen Reader:** A tool for testing and ensuring web accessibility for visually impaired users.
- **Content Delivery Network (CDN):** Distributes content across multiple servers to improve load times and reliability.

Tasks:

Task 1: Analyzing Page Speeds

Used tools like Google PageSpeed Insights and GTmetrix to evaluate the performance of web pages created in previous tasks. Identified areas for improvement and implemented recommendations.

Task 2: Minifying CSS

Applied minification techniques to CSS files to reduce their size and improve page load times.

Task 3: Studying NVDA Screen Reader

Explored the NVDA screen reader to understand how it aids accessibility and tested web pages for compatibility and usability for visually impaired users.

Tools Suggested/Used:

Google PageSpeed Insights:

- Used to analyze web page performance and provide actionable insights for optimization.

GTmetrix:

- Provided detailed reports on page speed and recommendations for performance improvements.

Minify CSS/JS:

- Tools and techniques used to reduce the size of CSS and JavaScript files.

NVDA Screen Reader:

- Tested web pages for accessibility, ensuring compatibility with screen reader technology.

Summary:

Day 6 focused on understanding the critical aspects of functional and non-functional requirements and their impact on web performance and user experience. Emphasized the importance of optimizing web pages according to standards to enhance performance and improve indexing on Google. Explored various tools and techniques for web optimization, including Google PageSpeed Insights, GTmetrix, minification of CSS/JS, and the use of NVDA screen reader for accessibility testing. Practical tasks involved analyzing page speeds, minifying CSS files, and studying the NVDA screen reader, equipping essential skills for creating efficient, accessible, and high-performing web applications.