DAY 7 REPORT
19 June, 2024

1. Non-Functional Requirements (NFR) for RDF

Definition: Non-functional requirements (NFRs) define how a system performs its functions. They focus on aspects like performance, usability, and reliability, rather than what the system does.

Examples for RDF:

- **Scalability:** The RDF system should be capable of efficiently handling increasing volumes of data.
- **Performance:** Queries run on RDF datasets should be processed quickly to ensure efficient data retrieval.
- **Usability:** The system should be easy to use for both developers and end-users.

2. Functional Requirements (FR) for RDF

Definition: Functional requirements (FRs) describe specific tasks or functions a system must perform, outlining what the system should do.

Examples for RDF:

- **Data Storage:** The system should be able to store RDF triples effectively.
- **Data Retrieval:** It should support running SPARQL queries to retrieve relevant data from the dataset.
- **Data Integration:** The system should facilitate the integration of data from multiple sources.

3. Page Speed Enhancement

- **Lighthouse:**
  - **Tool Description:** Lighthouse is an open-source tool used for auditing web pages in terms of performance, accessibility, SEO, and best practices.
  - **Usage:** It provides insights and recommendations for improving web page speed and overall quality.

- **PageSpeed Insights:**
  - **Tool Description:** This tool analyzes the content of web pages and provides suggestions to optimize page speed.
  - **Usage:** It helps identify issues that may slow down the page and offers optimization tips for better performance.

4. Creating a Minified Version of a Page

- **Definition:** Minification is the process of removing unnecessary characters (like whitespace and comments) from code without altering its functionality, thereby improving page load times.
- **Tools Used:**
  - **JavaScript:** UglifyJS
  - **CSS:** CSSNano
  - **HTML:** HTMLMinifier
- **Benefits:** Minifying code results in smaller file sizes, faster load times, and improved page performance.

5. NBDA Screen Reader

- **Description:** Screen readers are assistive technologies designed to help visually impaired users by reading out the content on a web page.
- **Usage:** Ensuring compatibility with screen readers like NVDA (NonVisual Desktop Access) is essential for creating accessible web pages.

6. Hosting a Webpage Using Netlify

- **Description:** Netlify is a platform designed for automating modern web projects, offering continuous deployment from Git repositories.
- **Steps Involved:**
  - **Deploying a Site:** Connect a GitHub repository and deploy the site via Netlify.
  - **Continuous Deployment:** Any changes pushed to the repository automatically trigger new builds and site deployments.
  - **Domain Management:** Configure custom domains and enable HTTPS for secure browsing.

Conclusion

On Day 7, we explored both non-functional and functional requirements of RDF systems, focusing on performance and usability. We learned to enhance web page speed using tools like Lighthouse and PageSpeed Insights and applied minification techniques to improve load times. We also delved into web accessibility using screen readers such as NVDA. Lastly, we gained practical experience in deploying and managing web pages through Netlify, utilizing its deployment automation and domain management features.