TRAINING TR-102 DAY 6 REPORT

1. NON-FUNCTIONAL REQUIREMENTS (NFR) FOR RDF:

Definition: NFRs define how a system performs a function. They are not about what the system does but how it does it. These include performance, usability, reliability, etc.

Examples for RDF:

Scalability: The RDF system should handle increasing amounts of data efficiently.

Performance: Queries on RDF datasets should be processed quickly.

Usability: The system should be user-friendly for developers and end-users.

2. FUNCTIONAL REQUIREMENTS (FR) FOR RDF:

Definition: FRs describe specific behaviors or functions of a system. They outline what the system should do.

Examples for RDF:

Data Storage: Ability to store RDF triples.

Data Retrieval: Capability to run SPARQL queries to retrieve data.

Data Integration: Integrate data from various sources.

3. PAGE SPEED ENHANCEMENT:

Lighthouse:

Tool Description: Lighthouse is an open-source tool used to improve the quality of web pages. It audits performance, accessibility, best practices, and SEO.

Usage: Running Lighthouse audits to get insights on page speed and recommendations for improvement.

PageSpeed Insights:

Tool Description: PageSpeed Insights analyzes the content of a web page, then generates suggestions to make that page faster.

Usage: Using the tool to identify issues and optimize web page speed.

4. CREATING A MINIFIED VERSION OF A PAGE:

Definition: Minification is the process of removing unnecessary characters from code without changing its functionality, enhancing load times.

Tools Used: Examples include UglifyJS for JavaScript, CSSNano for CSS, and HTMLMinifier for HTML.

Benefits: Reduced file sizes lead to faster load times and improved performance.

5. NBDA Screen Reader:

Description: Screen readers are assistive technology software that help visually impaired users by reading out the content on the screen.

Usage: Understanding how to make web pages accessible by ensuring compatibility with screen readers like NVDA (NonVisual Desktop Access).

6. HOSTING A WEBPAGE USING NETLIFY:

Description: Netlify is a platform for automating modern web projects, providing continuous deployment from Git repositories.

Steps Involved:

Deploying a Site: Connecting a GitHub repository and deploying the site.

Continuous Deployment: Changes pushed to the repository automatically trigger new builds and deployments.

Domain Management: Setting up custom domains and HTTPS.

CONCLUSION:

On Day 6, we deepened our understanding of non-functional and functional requirements for RDF systems, focusing on performance and usability aspects. We explored tools like Lighthouse and PageSpeed Insights to enhance web page speed, learning how to create minified versions of web pages to improve load times. Accessibility was emphasized through the use of screen readers like NBDA. Finally, we gained practical experience in hosting web pages using Netlify, leveraging its powerful deployment and domain management features.