TR-102

MASTERING THE SEMANTIC WEB

DAY-8

❖ Non-functional requirements in Semantic Web

Non-functional requirements (NFRs) describe how a system performs a task, as opposed to functional requirements, which describe what the system does. In the context of the Semantic Web, non-functional requirements are crucial for ensuring the system is usable, efficient, and reliable. Here are some key non-functional requirements for the Semantic Web:

➢ Scalability

- **Description:** The ability of the system to handle growing amounts of data and increasing numbers of users and queries.

- **Considerations:** The system should be able to expand in terms of storage, processing power, and network capacity without significant performance degradation.

➢ Performance

- **Description**: The speed and efficiency with which the system processes queries and returns results.
- **Considerations**: Includes query response time, data retrieval speed, and overall system throughput. Optimization techniques like indexing and caching can help enhance performance.

➢ Interoperability

- **Description**: The ability of the system to work seamlessly with other systems, applications, and data formats.
- **Considerations**: Support for standard protocols (e.g., HTTP, SPARQL) and data formats (e.g., RDF, OWL) to ensure compatibility with diverse systems and datasets.

➢ Usability

- **Description**: The ease with which users can interact with the system.
- **Considerations**: Intuitive user interfaces, clear documentation, and effective error handling to improve the user experience.

➢ Security

Name: Kanan Kaura   **Class:** D2 CSE C-2   **URN:** 2203845

- **Description**: Protecting the system and its data from unauthorized access, breaches, and other security threats.
- **Considerations**: Authentication and authorization mechanisms, encryption of data in transit and at rest, and regular security audits.

➢ **Reliability**

- **Description**: The ability of the system to function correctly and consistently over time.
- **Considerations**: Fault tolerance, redundancy, and regular maintenance to ensure high availability and minimal downtime.

➢ **Maintainability**

- **Description**: The ease with which the system can be updated, improved, and corrected.
- **Considerations**: Modular architecture, clear code documentation, and adherence to coding standards to facilitate easy maintenance and upgrades.

➢ **Extensibility**

- **Description**: The ability of the system to accommodate new features and functionalities without major changes.
- **Considerations**: Designing the system with a flexible architecture that allows for easy integration of new components and technologies.

➢ **Data Quality**

- **Description**: Ensuring the accuracy, completeness, consistency, and timeliness of data.
- **Considerations**: Implementing validation mechanisms, data cleaning processes, and provenance tracking to maintain high data quality.

➢ **Compliance**

- **Description**: Adherence to relevant standards, regulations, and best practices.
- **Considerations**: Compliance with standards such as W3C recommendations for the Semantic Web, and legal and regulatory requirements related to data privacy and security.

Non-functional requirements are essential for the successful implementation and operation of Semantic Web technologies. They ensure that the system not only meets its functional objectives but also performs efficiently, is user-friendly, secure, and can adapt to future needs. Addressing these non-functional requirements helps in building robust, scalable, and sustainable Semantic Web applications.

❖ Tools for Website Performance Optimization

➢ Google PageSpeed Insights

Google PageSpeed Insights (PSI) is a free tool provided by Google that analyzes the content of a web page and offers suggestions to make that page faster. It provides a score for both mobile and desktop versions of a website.

▪ Features:

a. **Performance Score**: A numerical score ranging from 0 to 100 that indicates the performance of a webpage.
b. **Field Data**: Real-world user data collected via the Chrome User Experience Report.
c. **Lab Data**: Simulated performance data using a fixed set of devices and network conditions.
d. **Opportunities**: Suggestions for improving page performance along with estimated savings in load time.
e. **Diagnostics**: Additional information about how a page adheres to best practices.

▪ Usage:

a. Go to the Google PageSpeed Insights website.
b. Enter the URL of the webpage you want to analyze.
c. Review the performance scores and suggestions for improvement.

➢ Lighthouse

Lighthouse is an open-source, automated tool for improving the quality of web pages. It can be run as a Chrome extension, in the Chrome DevTools, or from the command line.

▪ Features:

a. **Performance**: Measures page load speed and provides actionable recommendations.
b. **Accessibility**: Checks how accessible a site is for users with disabilities.
c. **Best Practices**: Audits for best practices in web development.
d. **SEO**: Evaluates how well the page is optimized for search engines.
e. **PWA (Progressive Web App)**: Assesses the reliability, speed, and engagement aspects of progressive web apps.

▪ Usage:

a. **Chrome DevTools**: Open DevTools (F12 or right-click and select "Inspect"), go to the "Lighthouse" tab, and run an audit.
b. **Chrome Extension**: Install the Lighthouse Chrome extension and use it to audit a page.

c. **Command Line**: Use Node.js to run Lighthouse via command line for automated workflows.

➢ GTmetrix

GTmetrix is a tool that provides insights into the performance of web pages, along with detailed recommendations for improvements. It combines data from Google Lighthouse and Web Vitals to offer a comprehensive analysis.

▪ Features:

a. **Performance Scores**: Includes scores for performance, structure, and Web Vitals.
b. **Waterfall Chart**: Visual representation of how elements load on the page.
c. **Video Playback**: Records a video of your page load for visual analysis.
d. **Historical Data**: Track performance changes over time.
e. **Test Locations**: Choose from multiple test locations around the world to see how your site performs for different users.

▪ Usage:

a. Go to the GTmetrix website.
b. Enter the URL of the webpage you want to analyze.
c. Select the test location and browser options if needed.
d. Review the detailed report and recommendations.

➢ Comparing the Tools

▪ **Google PageSpeed Insights**: Best for quick, accessible performance metrics and optimization tips, integrated with Google's ecosystem.
▪ **Lighthouse**: Comprehensive auditing tool with a broad scope, including performance, accessibility, best practices, SEO, and PWA support. Ideal for developers needing detailed insights and automation capabilities.
▪ **GTmetrix**: Combines multiple metrics and provides detailed visualizations like waterfall charts and video playback. Useful for tracking historical performance and comparing across different geographic locations.

Using these tools can help you identify and fix performance bottlenecks, improve user experience, and optimize your site for better search engine rankings. Each tool has its strengths, and often they are best used in conjunction to get a complete picture of your website's performance and optimization opportunities.

❖ NVDA (NonVisual Desktop Access)

NVDA (NonVisual Desktop Access) is a free, open-source screen reader for Microsoft Windows. It enables blind and visually impaired users to access and interact with the Windows operating system and various applications through synthesized speech and braille displays.

➤ **Key Features of NVDA**

- ▪ **Screen Reading**:

  a. Provides real-time feedback of on-screen text, menus, dialog boxes, and other interface elements.
  b. Supports a variety of languages and synthesized voices.

- ▪ **Braille Display Support**:

  a. Compatible with over 50 different braille displays, allowing users to read text in braille.
  b. Supports both USB and Bluetooth connections.

- ▪ **Web Browsing**:

  a. Works seamlessly with web browsers like Google Chrome, Mozilla Firefox, and Microsoft Edge.
  b. Provides accessibility to web content, including text, images, links, forms, and other interactive elements.

- ▪ **Application Support**:

  a. Compatible with a wide range of applications, including Microsoft Office, email clients, and instant messaging programs.
  b. Custom scripts and add-ons are available to enhance compatibility with specific applications.

- ▪ **Speech Output**:

  a. NVDA uses speech synthesis to read text aloud, allowing users to navigate and interact with their computers.
  b. Users can choose from a variety of speech synthesizers and voices.

➤ **Benefits of Using NVDA**

- ▪ **Cost**: NVDA is completely free and open-source, making it accessible to anyone regardless of financial resources.
- ▪ **Regular Updates**: Frequent updates ensure that NVDA stays current with new software and hardware developments.
- ▪ **Customization**: Users can customize NVDA to suit their specific needs through add-ons and configuration options.
- ▪ **Independence**: Empowers blind and visually impaired individuals to use computers independently for work, education, and personal activities.

➤ **Getting Started with NVDA**

- ▪ **Installation**:

a. Download the NVDA installer from the official website.
b. Run the installer and follow the on-screen instructions to complete the installation.

- **Basic Usage**:

a. Launch NVDA by double-clicking the NVDA icon or pressing the configured keyboard shortcut.
b. Use keyboard shortcuts to navigate and interact with your computer. Common shortcuts include:
    a. NVDA+N: Open the NVDA menu.
    b. NVDA+T: Read the title of the current window.
    c. NVDA+F12: Announce the current time.

- **Customization**:

a. Access the NVDA settings by pressing NVDA+N and selecting "Preferences."
b. Customize speech settings, braille display options, keyboard shortcuts, and other preferences.