# **Web Page Performance Feature Documentation**

## **1. Introduction**

The Web Page Performance feature focuses on optimizing website speed, rendering efficiency, and overall user experience. By addressing loading times, responsiveness, and resource management, we enhance user satisfaction and engagement.

**Web performance is crucial for several reasons:**

User Experience: Fast-loading websites provide a better user experience. Users are more likely to stay on a site that loads quickly and responds promptly to their interactions. Slow sites, on the other hand, can lead to frustration and abandonment.

Accessibility: Good web performance ensures that content is accessible to users regardless of their device or internet connection speed. This is particularly important for users in areas with slower internet speeds or those using older devices.

Business Goals: Efficient web performance supports business objectives by reducing bounce rates and increasing user engagement. A well-performing site can lead to higher conversion rates and customer satisfaction.

Cost Efficiency: Optimizing web performance can reduce the amount of data transferred, which is beneficial for users with limited data plans and can also lower hosting costs for website owners.

## **2. Feature Overview**

The Web Page Performance feature includes the following components:

### **2.1 Format Suggestion for Images**

* Serve correctly sized images:
  + Use the src set attribute to provide different image sources based on screen resolution.
  + Optimize images for specific devices to reduce unnecessary data transfer.
* Modern image formats:
  + Consider using WebP or AVIF formats for better compression without sacrificing quality.
  + These formats improve loading speed and reduce bandwidth usage.

### **2.2 Mobile-Friendly and Responsive Pages**

* Implement responsive layouts using HTML and CSS.
* Set the viewport meta tag to control page dimensions across devices.
* Use responsive images and adapt font sizes for varying screen sizes.

### **2.3 JS/CSS Size and Page Reloading**

* Efficient JS/CSS:
  + Minimize file sizes by removing unused code and optimizing libraries.
  + Leverage minification and compression techniques.
* Page reloading strategies:
  + Refresh the page on browser resize to handle layout changes.
  + Use event listeners to trigger reloading when necessary.

## **3. How It Works**

1. **URL Input**:
   * Provide the webpage URL for analysis.
2. **Element Assessment**:
   * Evaluate loading times, responsiveness, and resource usage.
   * Generate performance metrics.

## **4. Work Distribution**

* **Timeline**: 2 Weeks
* **Backend & Algorithm**: Animesh
* **Frontend**: Vidya
* **Detailing & Designing**: Ankita

## **5. Technical Elements**

## **6. References and Resources**

* [The "why" of web performance - Learn web development | MDN (mozilla.org)](https://developer.mozilla.org/en-US/docs/Learn/Performance/why_web_performance)
  + It covers loading speed, rendering, and user experience. Slow sites lead to abandonment, while good performance goes unnoticed by users.
* [Optimizing Website Performance: Best Practices | Website Designers](https://websitedesigners.com/blog/optimizing-website-performance-best-practices-for-faster-loading-times/)
  + This shows fast loading times and mobile optimization are essential for user satisfaction and engagement.
* [MergedFile (diva-portal.org)](https://kth.diva-portal.org/smash/get/diva2:1228341/FULLTEXT01.pdf)
  + Explores techniques like caching, fewer HTTP requests, and Web Workers, emphasizing their effect on user satisfaction.
* [IJRTI (ijcrt.org)](https://ijcrt.org/papers/IJCRT2106225.pdf)
  + FACTORS AFFECTING BROWSER PERFORMANCE FOR WEBSITES