

MMWP2024 - Übungsserie 10 - Optimierung einer Webseite (Barrierefreiheit, SEO, Ladezeit, Aktualisierung auf HTML5/CSS3, ES-9)

Lehrveranstaltung Multimediale Webprogrammierung
Wintersemester 2024/25

Link zum Kurs: <https://bildungsportal.sachsen.de/opal/auth/RepositoryEntry/46557921282>

Zum Vertiefen der Kenntnisse: https://developer.mozilla.org/en-US/docs/Learn_web_development/Extensions/Performance und <https://developer.mozilla.org/en-US/blog/optimize-web-performance/>

1. Analysieren und Identifizieren den momentanen Status der Webseite

1.1 Verwenden Sie Tools wie [Google-PageSpeed](#), um die Leistung der Webseite zu prüfen.

1.2 Bestimmen Sie die Zielgruppe, für die diese Webseite eingesetzt werden soll.

1.3 Bewerten Sie eingesetzte Medien und ob diese auf die Zielgruppe abgestimmt sind.

2. Optimieren Sie die Webseite für Suchmaschinen

2.1 Fügen Sie [relevante Meta-Tags](#) hinzu oder verbessern Sie sie, einschließlich des Titels, der Beschreibung und der Schlüsselwörter. Stellen Sie sicher, dass diese Tags den Inhalt der Seite genau wiedergeben und wichtige Schlüsselwörter enthalten.

2.2 Verwenden Sie [semantische HTML-Elemente](#) wie `<article>`, `<section>`, `<header>` und `<footer>` zur sinnvollen Strukturierung des Inhalts.

2.3 Stellen Sie sicher, dass die Website über eine aktuelle [Sitemap](#) und eine ordnungsgemäß konfigurierte [Robots.txt](#)-Datei verfügt, um die Crawler von Suchmaschinen optimal zu leiten.

Angenommen, der Kunde möchte keine Indexierung von AI-Services. Konfigurieren Sie die Robots.txt Datei dementsprechend.

3. Verbessern Sie die Ladezeit der Webseite

3.1 Reduzieren Sie CSS-, JavaScript- und HTML-Dateien, um die Dateigröße zu verringern, ohne die Funktionalität zu beeinträchtigen. Nutzen Sie hierfür NodeJS Bibliotheken ([cssso](#) oder [Minify](#)) oder Dienste wie [Minifier](#).

3.2 Verwenden Sie Tools wie [Paint.Net](#), [Gimp](#) oder <https://tinypng.com/>, um Bilder zu komprimieren, ohne die Qualität zu beeinträchtigen. Benutzen Sie [moderne Bilddateiformate](#), um eingesetzte Bilder weiter zu optimieren.

3.3 Implementieren Sie "[Lazy Loading](#)" für Bilder und Videos, um den CRP (critical render path) zu optimieren.

3.4 Entfernen Sie nicht benutzte Javascript- und CSS-Bibliotheken und verwenden sie asynchrones Laden von Bibliotheken, um die Seite möglichst schnell zu rendern

4. Testen und validieren Sie die optimierte Webseite.

4.1 Verwenden Sie Validatoren wie den [W3C HTML Validator](#), um sicherzustellen, dass der Code den Standards entspricht.

4.2 Führen Sie Cross-Browser-Tests durch, um die Kompatibilität mit verschiedenen Browsern und Geräten sicherzustellen.

4.3 Verwenden Sie Tools wie [Google PageSpeed Insights](#) oder [GTmetrix](#), um die Leistung der Webseite zu testen und Vorschläge für weitere Verbesserungen zu erhalten.

4.4 Führen Sie eine erneute Überprüfung der Barrierefreiheit durch z.B. mit [AccessibilityChecker](#), um sicherzustellen, dass alle Probleme behoben wurden und die Webseite den Barrierefreiheitsstandards entspricht.

Selbststudienzeit/Weitere Schritte:

Nutzen Sie die Tutorials für eine weitere Vertiefung: <https://developers.google.com/search/docs/fundamentals/seo-starter-guide?hl=de> und <https://developers.google.com/search/docs/appearance?hl=de>

Testseite wurde mit Vorlage von <https://bootstrapmade.com/> erstellt.