

F.-A.-Brockhaus-Gymnasium

Schuljahr: 2023/24

Facharbeit  
Grundkurs Informatik

**Barrierefreiheit im Web:  
Möglichkeiten als Nutzer und Entwickler**

Eine Arbeit zu den Möglichkeiten der Barrierefreiheit im Internet, welche sich intensiv mit der Umsetzung einer barrierefreien Website beschäftigt.

Verfasser: Louis Taube

Betreuerin: Frau Webs

Abgabetermin: 01.03.2024

# Inhaltsverzeichnis

|   |           |
|---|-----------|
| <b>1 Einleitung</b>                               | <b>1</b>  |
| <b>2 Begriffserklärung</b>                        | <b>2</b>  |
| <b>3 Möglichkeiten als Nutzer</b>                 | <b>3</b>  |
| 3.1 Screenreader                                  | 3         |
| 3.2 Vergrößerungssoftware                         | 4         |
| 3.3 Tastaturbedienung                             | 4         |
| <b>4 Möglichkeiten als Entwickler</b>             | <b>6</b>  |
| 4.1 Rechtliche Grundlagen und Normen              | 6         |
| 4.2 Barrierefreies Design                         | 7         |
| 4.3 Seitenstruktur                                | 8         |
| <b>5 Mehraufwand einer barrierefreien Website</b> | <b>10</b> |
| 5.1 Vorgehensweise und Technologien               | 10        |
| 5.2 Programmierung einer barrierefreien Website   | 11        |
| 5.3 Überprüfung der Barrierefreiheit              | 11        |
| 5.4 Bewertung des Mehraufwandes                   | 11        |
| <b>6 Fazit</b>                                    | <b>13</b> |
| <b>7 Anhang</b>                                   | <b>14</b> |
| 7.1 Anh. 1: Umfrage Entwickler                    | 14        |
| <b>8 Literaturverzeichnis</b>                     | <b>16</b> |
| <b>9 Selbstständigkeitserklärung</b>              | <b>18</b> |

# 1 Einleitung

Barrierefreiheit - ein Thema für uns alle. Ein Thema, welches in unserer Gesellschaft zurecht immer mehr Zuspruch und Platz bekommt.

Dazu digitalisiert sich unsere Gesellschaft immer weiter. Fast jeder Deutsche besitzt bereits einen Internetzugang, alleine in Deutschland. Ein Zugang zum Internet bedeutet auch Zugriff auf zahlreiche Möglichkeiten und Tools. Nutzern mit Einschränkungen sollten diese Tools nicht verwehrt bleiben. Aus diesem Grund möchte ich mich in dieser Facharbeit intensiv mit den Möglichkeiten für Nutzer und Entwickler beschäftigen, um die Barrieren in der Webnutzung zu überwinden.

Mein Ziel ist es, mit dieser Facharbeit auf die Wichtigkeit des Themas aufmerksam zu machen und herauszufinden, wie weit die Umsetzung dieses Themas bereits auf verschiedenen Webseiten erfolgt. Des Weiteren habe ich ein persönliches Interesse daran, die Webseiten, an denen ich arbeite, auch weitestgehend barrierefrei machen zu können. Ich möchte herausfinden, wie viel Mehraufwand die barrierefreie Gestaltung einer Webseite benötigt und wie man diesen Aufwand reduzieren kann. Dazu werde ich eine eigene Website programmieren, welche über Barrierefreiheit informiert und dabei bestmöglich barrierefrei sein sollte. Zudem werde ich im Theorieteil meiner Facharbeit mein Wissen über die Möglichkeiten der Barrierefreiheit von Webseiten für Nutzer und Entwickler erweitern und so sehr vertiefen, dass ich in der Lage bin, die Barrierefreiheit einer Website zu bewerten.

Um den Zeitaufwand und die Kritikpunkte von Barrierefreiheit besser beurteilen zu können, habe ich eine anonyme Umfrage<sup>1</sup> mit mehreren Informatikern durchgeführt, welche ihre Einschätzung zur Wichtigkeit von Barrierefreiheit auf Webseiten, sowie ihre eigenen Erfahrungen mitteilen konnten.

---

<sup>1</sup> Anh. 1: Umfrage Entwickler

## 2 Begriffserklärung

Spricht man von Barrierefreiheit im Internet , handelt es sich häufig nicht nur um die vereinfachte Nutzung der Websites von körperlich eingeschränkten Personen, sondern von jeglicher Art der Einschränkung. wie zum Beispiel ein hohes Alter oder ein falsches Endgerät, welche hierbei ebenfalls als Einschränkungen gelten. An diesem Punkt unterscheidet sich der Begriff Barrierefreiheit in der Webentwicklung zum Begriff der Barrierefreiheit im natürlichen Leben.

Wenn man sich die Definition bezüglich Barrierefreiheit in der Webnutzung anschaut, wird einem ziemlich schnell klar, dass es sehr viele Einschränkungen und Aspekte geben kann, die den Umgang damit für Menschen mit Einschränkungen jeglicher Art erschweren können. Keine Website kann zu 100 Prozent barrierefrei sein, das ist eine utopische Vorstellung! Darum sollte es in der Webentwicklung immer das Ziel sein, die Webseite so gut wie möglich barrierefrei zu gestalten.

Vergleich zu DUDEN; berühmte Personen ZITIEREN

## 3 Möglichkeiten als Nutzer

Für Menschen, welche das Internet täglich ohne Einschränkungen nutzen können, ist es zu einer Selbstverständlichkeit geworden, sich frei im Web bewegen zu können. Ihr einziger Kritikpunkt gilt dann eher den schlechten oder älteren Webseiten, oder dem veralteten User Interface (Benutzeroberfläche) bzw. dem Design. Was aber für diese Nutzer nur ein kleines optisches Problem bedeutet, kann für Anwender mit Einschränkungen ein echtes Hindernis sein. Diese Hindernisse werden leider nicht immer, eher sehr selten, von den Verantwortlichen entfernt. Daher müssen sich Betroffene selbst Hilfe zur Anwendung suchen. Diese Hilfe kann in der Software-, sowie in Hardware-Lösungen liegen. Einige ausgewählte IT-Lösungen möchte ich Ihnen hierzu gern vorstellen.

### 3.1 Screenreader

Eine der Software Anwendungsmöglichkeiten für Menschen mit Sehschwäche oder vollkommener Blindheit sind Screenreader. Dieser liest dem Nutzer die Inhalte einer Website laut vor. Er soll die visuellen Inhalte von Webseiten in Sprache umwandeln bzw. gibt die Inhalte so wieder, dass Menschen mit Sehbeeinträchtigung diese wahrnehmen können. Hierbei entsteht allerdings das Problem, dass sie nicht die ganze Website umwandeln sollen, sondern nur die Inhalte und Informationen, welche für den Nutzer wichtig sein sollen. Dabei sind die Screenreader auf die Hilfe von Entwicklern angewiesen. Des Weiteren muss ein Screenreader auch in der Lage sein, Inhalte zu bedienen, wie z. B. Input Felder auszufüllen oder Checkboxes zu aktivieren bzw. zu deaktivieren.

Die Screenreader sind IT-Lösungen, welche vom Nutzer auf dem digitalen Endgerät installiert werden müssen bzw. abhängig vom Betriebssystem schon vorinstalliert sind und nur noch aktiviert werden müssen. Entwickler können die Screenreader unterstützen, indem Sie auf z.B. eine klare Quelltext Struktur<sup>2</sup> achten. Etablierte Screenreader sind z. B. JAWS (Windows), VoiceOver (MacOS) oder TalkBack (Android) [1]. Dennoch sind Screenreader

---

<sup>2</sup> weitere Informationen: 4.3 Seitenstruktur

keine perfekte Lösung, da sie von der sauberen und strukturierten Arbeit des Entwicklers abhängig sind.

## 3.2 Vergrößerungssoftware

Ein weitere weit verbreitete Möglichkeit und Software-Lösung für Nutzer mit Sehschwäche ist Vergrößerungssoftware. Diese ist in vielen Browsern und Betriebssystemen schon vorinstalliert. Der Sinn solcher Software bzw. die Funktion liegt darin, kleine Texte oder Details auf dem Bildschirm zu erkennen. Die Nutzer mit Seheinschränkungen können die Inhalte der Seiten auf ihre passende gewünschte Größe skalieren.

Die Funktionen dieser IT-Lösungen variieren von Modus und Art der Vergrößerungssoftware. Beispielsweise gibt es Anwendungen, welche die gesamte Website vergrößern, oder auch nur einen bestimmten Bereich, wie mit einer Lupe skalieren. Ergänzend dazu ändern viele Vergrößerungssoftware auch noch die Farben, um somit ein besseres Kontrastverhältnis zu erzeugen [9]. Der Nachteil dieser Software-Lösungen liegt darin, dass die Software oft nur einen Teil des Bildschirms vergrößert, wobei teilweise der Kontext des Inhalts verloren gehen kann oder die Nutzer bekommen ein Popup Window nicht mit, da es außerhalb ihres Sichtbereiches aufgeht [9].

Vergrößerungssoftware ist eine weitere Möglichkeit für einen effektiveren und erleichterten Umgang auf Websites für Nutzer mit Einschränkungen. Sie sind die perfekte Ergänzung für Nutzer, welche nur eine Sehschwäche haben und nicht vollkommen erblindet sind.

## 3.3 Tastaturbedienung

Einer der wichtigsten Fragen, welche für Nutzer mit Einschränkungen beantwortet werden müssen, ist die Navigation. Wie navigiert man sich auf der Website, wenn man nichts sieht? Oder wie navigiert man sich, wenn man nur eine Hand hat? Die Antwort für diese Nutzer kann die Tastaturnavigation sein.

Die Tastaturnavigation bezieht sich auf die Möglichkeit, Webinhalte nur durch die Tastatur zu erreichen. Dies ermöglicht auch Menschen mit

motorischen Beeinträchtigungen, oder Nutzern, welche Schwierigkeiten mit dem präzisen Umgang mit einer Maus haben, eine zielführende Navigation im Internet. Diese präzise Navigation ist allerdings wieder auf die Sauberkeit des Quellcodes der Seite angewiesen. Die Navigation erfordert, dass sämtliche Funktionen und Interaktionen, welche mit der Maus erfolgen, auch über die Tastatur steuerbar sind. Dies bedeutet im Speziellen, dass Nutzer sich mit Tastenkombinationen durch die Inhalte der Websites, wie z. B. Links, Schaltflächen oder Input-Feldern, navigieren können.

Des Weiteren sind die Tastenkombinationen oft gleich, um den Nutzern das Lernen zu erleichtern. Beispielsweise kann sich mit der Tabulatortaste von Element zu Element navigiert werden und mit der Enter- bzw. der Leertaste kann bestätigt werden. Dies ist aber nur möglich, wenn Entwickler ihre Elemente alle einzeln implementieren und somit keine Doppelauswahl von einem Element möglich ist.

## 4 Möglichkeiten als Entwickler

Bei all den Möglichkeiten, welche Nutzer haben, um sich barrierefreier im Internet zu bewegen, ist zu hervorzuheben, dass die Beseitigung dieser Barrieren eigentlich die Aufgabe der Entwickler der jeweiligen Webseiten ist. Des Weiteren funktioniert ein Großteil der IT-Lösungen für die Nutzer nur als Ergänzung und auch nur dann, wenn sich die Entwickler der Webseiten an bestimmte Normen und Regeln halten, welche vorgegeben werden. Ausgewählte Möglichkeiten für die Barrierefreiheit aus der Entwicklersicht, stelle ich Ihnen im folgenden Abschnitt vor.

### 4.1 Rechtliche Grundlagen und Normen

Die Umsetzung von Barrierefreiheit im Web ist nicht nur eine ethische Verpflichtung, sondern sie ist in vielen Ländern auch rechtlich vorgeschrieben. Dies dient dazu, Entwickler zu verpflichten, wenn sie z.B. an Behördenseiten arbeiten, diese auch barrierefrei zu entwickeln und damit einen gleichberechtigten Zugang zum Internet für alle Nutzer zu schaffen.

Eine weit verbreitete Norm ist z. B. die Web Content Accessibility Guidelines (WCAG). Dies sind vom World Wide Web Consortium (W3C) verfasste anerkannte internationale Richtlinien. Sie liefern Kriterien und Anweisungen für die Umsetzung von Barrierefreiheit im Web.

Eine deutsche Richtlinie, welche sich eng an der WCAG orientiert, ist die ["Barrierefreie-Informationstechnik-Verordnung - BITV 2.0"](#). Sie regelt die EU-Richtlinie 2016/2102 im deutschen Recht. Die BITV 2.0 richtet sich an öffentliche Stellen des Bundes aber gibt im Vergleich zu der ersten Version der BITV (BITV 1.0) keine eigenen Standards, sonder verweist auf die Europäische Norm ["ETSI EN 301 549 - V3.2.1 - Accessibility requirements for ICT products and services"](#) [\[3\]-\[5\]](#).

Für öffentliche Stellen des Bundes ist diese Norm verpflichtend und wird auch überprüft, aber auch viele private Unternehmen orientieren sich an der WCAG und der BITV, um einen Maßstab für die Barrierefreiheit auf ihrer Website zu haben.



## 4.2 Barrierefreies Design

Wenn wir über barrierefreies Design reden, sprechen wir nicht davon, dass wir für Menschen mit Einschränkungen ein neues Design und eine neue UI erstellen müssen. Barrierefreies Design ist Design für alle [2]. Die Vorteile liegen nicht nur bei Nutzern mit Einschränkungen, sondern bei allen Nutzern. Beispielsweise erleichtert ein starkes Kontrastverhältnis auch Nutzern ohne Einschränkungen die Inhalte einer Website unter Blendung durch ungünstige Sonneneinstrahlung zu erkennen. Oder ein Responsive Design ist nicht ausschließlich für die Vergrößerungen auf einem Bildschirm gut, sondern kann sich auch den verschiedenen Größen der Bildschirme anpassen.

Bei der Verwendung von Farben gibt es viele Aspekte bei der Entwicklung zu beachten. Beispielsweise sollten Informationen nicht nur durch Farben übermittelt werden, damit auch Farbenblinde Nutzer diese Anwendung nutzen und bedienen können. Weiterhin sollte auf ein starkes Kontrastverhältnis geachtet werden, um auch Nutzern mit Sehschwäche eine gute Arbeit mit der Website zu ermöglichen. Bestimmte Kontrastverhältnis Werte werden z. B. in den WCAG empfohlen. Eine weitere Möglichkeit ein hohes Kontrastverhältnis anzubieten, wäre es einen Button auf der Website einzufügen, welcher die Hintergrundfarbe und die Textfarbe tauschen.

Ein große Hilfe für Nutzer mit Einschränkungen ist das Vergrößern von Inhalten mit externen Anwendungen oder internen Buttons der Website. Die Aufgabe der Entwickler liegt darin, dass die Inhalte so designt werden, dass sie mindestens bis zu 200% skaliert werden können, ohne die Funktion der Website zu verlieren. Dies gilt für die komplette Website. Um diese Skalierung zu ermöglichen, wird oft ein Responsive Design benötigt [6]. Ein Responsive Design ist ein flexibles Design, es passt sich der Bildschirmbreite und Höhe an, ohne dass der Entwickler mehrere Designs anfertigen muss [7, Abs. 4].

Bei der Entwicklung von Websites ist Regelmäßigkeit wichtig, Entwickler sollten beispielsweise für Überschriften einer Ebene immer das gleiche Layout und Design verwenden. Des Weiteren sollte Text am besten rechts- oder linksbündig positioniert sein und eine Zeile sollte nicht mehr 80 Zeichen beinhalten [6].

## 4.3 Seitenstruktur

Wenn man von Möglichkeiten für Entwickler ihre Seiten barrierefrei zu machen spricht, ist eine saubere organisierte Seitenstruktur, nicht nur im visuellen, sondern auch im Quelltext, was für einen Entwickler am einfachsten und schnellsten umzusetzen ist.

Eine große Hilfe für Screenreader für Nutzer ist eine klare und lineare Quelltext-Struktur. Gemeint ist hierbei die Anordnung der HTML-Elemente im Quellcode, diese sollten linear sein und durch das Stylen der Elemente, z. B. mit CSS oder einem CSS-Framework wie TailwindCSS<sup>3</sup>, sollte keine große Veränderung der Position und der Elemente im Vergleich zur visuellen Website vorhanden sein. Beispielsweise sollte eine Navbar, welche an den oberen Bildschirmrand geheftet ist, auch im Quellcode ganz oben stehen. Dadurch ermöglicht der Entwickler den Nutzern, mit deren Screenreadern ähnliche Websites mit dem gleichen Layout zu besuchen, da der Screenreader sich am Quellcode orientiert.

Entwickler sollten ebenso beachten (falls sie nicht englische Inhalte entwickeln) die verwendete Sprache im HTML-Element zu kennzeichnen (`<html lang="de">`), dies sollte genauso bei Zitaten in Blockquote-Elemente erfolgen [6].

HTML in seiner Urform, ohne Frameworks, bietet auch einige Möglichkeiten, um Screenreader zu unterstützen. Eine davon ist, dass der Entwickler die richtigen HTML-Elemente nutzt. Dies klingt am Anfang sehr einfach, kostet aber doch ein wenig Disziplin, da Entwickler eben nicht nur ein Span-Element (`<span>Lorem Ipsum</span>`) nutzen und das Layout mit CSS verändern kann, sondern dass für eine Überschrift die richtige Ebene genutzt werden muss. Beispielsweise Ebene 1 (`<h1>Lorem Ipsum</h1>`). Dies bezieht sich auch auf alle verschiedenen HTML-Elemente, solange der Entwickler die Möglichkeit hat, etwas über ein Element zu spezialisieren, sollte er dies auch tun. Der Aufwand lohnt sich, denn bei den richtigen Elementen erkennt auch

---

<sup>3</sup> [www.tailwindcss.com](http://www.tailwindcss.com)

der Screenreader, ob es sich um einen normalen Text handelt oder eben über eine wichtige Überschrift.

Einzelne CSS-Framework bieten genauso Möglichkeiten an, um die Screenreader zu unterstützen. Beispielsweise bietet TailwindCSS bestimmte Klassen an, um Dinge visuell von der Website auszublenden, aber dem Screenreader diese Elemente trotzdem anzuzeigen [8].

Weitere Möglichkeiten welche ein wenig zeitintensiver sind, wären Alternativinhalte. Das fängt bei einfachen Beschreibungen von wichtigen Bildern mit Hilfe des `alt=""` Attributs bei der Implementierung von Bildern an (`<img src="" alt="" />`), kann aber eben genauso etwas komplizierter werden, wenn eine Beschreibung von Diagrammen implementiert wird. Was man dabei auf keinen Fall vergessen darf, ist, dass es nicht darum geht, alles zu beschreiben, sondern nur die wichtigsten Inhalte wiederzugeben. Ein Bild, welches nur für die Gestaltung im Hintergrund liegt und keinen informativen Mehrwert hat, sollte auch nicht beschrieben werden.

## 5 Mehraufwand einer barrierefreien Website

Ein großes Argument von Web-Entwicklern gegen die Barrierefreiheit im Web ist die Zeit. Einige Entwickler sind der Meinung, dass die Zeit, die eine barrierefreie Website mehr benötigt, zu viel ist und sich nicht genügend entlohnt wird. Des Weiteren wird oft kritisiert, dass keine verbindlichen Richtlinien für Entwickler in der freien Wirtschaft existieren, was dazu führt, dass Auftraggeber oft für Barrierefreiheit nicht zahlen wollen. Aber wie groß ist der zeitliche Mehraufwand wirklich? ■ Dies möchte ich in diesem Abschnitt meiner Arbeit untersuchen. Dafür habe ich eine Website<sup>4</sup> programmiert, welche die Vorgaben meiner Facharbeit für Barrierefreiheit versucht umzusetzen und gleichzeitig noch über Barrierefreiheit informiert.

### 5.1 Vorgehensweise und Technologien

Um den Zeitaufwand meiner barrierefreien Website bestmöglich bewerten zu können, habe ich die gleichen Frameworks und Programmiersprachen verwendet wie in meinen sonstigen Projekten. Bei Frameworks handelt es sich um Grundgerüste für das Programmieren, welche bestimmte Prozesse beim Programmieren optimieren und effizienter laden [11]. Als Programmiersprache verwende ich Typescript bzw. JavaScript, wobei es sich bei Typescript um eine Weiterentwicklung JavaScript handelt. Als Web-Framework kommt SvelteKit<sup>5</sup> zum Einsatz mit dem CSS-Framework TailwindCSS als Ergänzung. In Verbindung mit TailwindCSS basiert mein Layout und Design der barrierefreien Website auf einer eigenen UI-Bibliothek<sup>6</sup>, die in Zusammenarbeit mit anderen Entwicklern während einer aktuellen Neuaufsetzung unseres gemeinsamen Projektes "Vokabel-Box"<sup>7</sup> entstanden ist, aber von mir mithilfe von TailwindCSS umgesetzt wurde bzw. wird.

Für die Entwicklung und Planung der Website entwarf ich ein Figma Design<sup>8</sup>, wobei es sich um eine Vorentwurf der Webseite handelt, welcher das

---

<sup>4</sup> **Aufrufbar unter:** [facharbeit.louist2469.de](https://facharbeit.louist2469.de)

<sup>5</sup> <https://kit.svelte.dev/>

<sup>6</sup> **Aufrufbar unter:** <https://ui.louist2469.de/>

<sup>7</sup> **Aufrufbar unter:** <https://app.vokabel-box.de/>

<sup>8</sup> **Aufrufbar unter:** [facharbeit.louist2469.de/figma-design](https://facharbeit.louist2469.de/figma-design)

Layout wiedergibt. Dieser Vorentwurf trug dazu bei, ein einheitliches Layout zu entwickeln und sich auch erste Gedanken zur Umsetzung der Barrierefreiheit zu machen. Nach der Fertigstellung des Entwurfs lag meine Aufgabe darin, diesen Entwurf mit Hilfe von TailwindCSS in meinem SvelteKit-Projekt umzusetzen<sup>9</sup>. Damit ich die Barrierefreiheit besser einschätzen konnte, lag nach der Umsetzung meine Aufgabe darin, diese zu überprüfen<sup>10</sup>, die Erkenntnisse bei der Weiterentwicklung zu berücksichtigen und das Projekt fertigzustellen. Um eine klare Dokumentation meiner Arbeit durchzuführen und gleichzeitig eine Sicherung meiner täglichen Arbeit zu haben, erstellte ich ein Github-Projekt<sup>11</sup> mit meiner Website. Bei Github handelt es sich um eine internationale Plattform mit einer großen Community, welche sich auf die Sicherung und die Verwaltung von Programmierprojekten spezialisiert hat [12]. Dieses Github-Projekt wurde dann anschließend über die Web-Deployment-Plattform Vercel<sup>12</sup> veröffentlicht und mit der zugeordneten Domain bzw. Subdomain verknüpft [13].

## 5.2 Programmierung einer barrierefreien Website

Für eine bestmögliche Umsetzung des visuellen Figma-Entwurfs, welcher auch barrierefrei ist, konzentrierte ich mich erstmals nach der Aufsetzung des Projektes um die Umsetzung der HTML-Struktur, in Ergänzung mit TailwindCSS.

Die Schwierigkeit lag für mich darin, die richtigen HTML-Elemente zu benutzen, dies hing damit zusammen, dass ich in meinen vorigen Projekten die meisten Elemente nur durch CSS nachgestylt habe und die Funktionen durch JavaScript implementiert hatte, wobei das richtige Element nebensächlich war.

Eine weitere Schwierigkeit bei der anfänglichen Umsetzung lag darin, dass im Vergleich zum Figma-Design keine abweichende Gruppe von Bildschirmgrößen bestehen, sondern dass das mein Ziel darin liegt, das Design für alle Bildschirmgrößen so umzusetzen, dass so wenig Code wie möglich benötigt wird und er bestmöglich Effizient ist. Der Vorteil von diesem Aspekt

---

<sup>9</sup> **Mehr Informationen:** 5.2 Programmierung einer barrierefreien Website

<sup>10</sup> **Mehr Informationen:** 5.3 Überprüfung der Barrierefreiheit

<sup>11</sup> **Aufrufbar unter:** <https://github.com/LouisT2469/Barrierefreiheit-im-Web>

<sup>12</sup> <https://vercel.com/>

liegt darin, dass TailwindCSS durch ihre unterschiedlichen Klassen, auch Breakpoints hat, welche man vor die anderen Designklassen setzen kann, um damit ab bestimmten Größen die Inhalte zu verändern [10]. In Verbindung mit Flex-Boxen konnte ich die verschiedenen Bildschirmgrößen gut umsetzen. Trotz alledem war dies nicht überall möglich, beispielsweise bei der Navigationsleiste am oberen Bildschirmrand.

Diese anfänglichen Schwierigkeiten ließen sich aber lösen und hielten mich nicht davon ab, schon von Anfang an einzelne Aspekte der Barrierefreiheit umzusetzen. Ein Aspekt davon war die Verwendung von beschreibenden Wortgruppen, welche ich in bestimmte Elemente mit Funktionen als Attribut zuwies, beispielsweise Elemente, die einen auf andere Seiten weiterleiten. Für diese beschreibenden Wortgruppen nutze ich das Attribut `aria-label="Lorem Ipsum"`. Des Weiteren achtete ich von Beginn an auf die richtige Verwendung der Funktion entsprechenden Elemente und auf eine klare gerenderte Quelltext Struktur.

Für eine bessere Barrierefreiheit fing ich nach dem das Grundgerüst der Website stand an, an weiteren Features der Barrierefreiheit zu arbeiten. Ich fing mit der Arbeit an zwei Buttons an, welche die Schriftgröße vergrößern und die Farben für ein besseres Kontrastverhältnis tauschen würden. Für die Umsetzung der verschiedenen Schriftgrößen nutze ich ein spezielles Feature von Svelte, nämlich das Binding-Feature. Dieses ermöglicht mir, dass ich innerhalb von HTML-Tags Variablen verwenden kann, um die Inhalte dynamisch zu machen. Um effizienter zu arbeiten und nicht die gleiche Code-Logik immer wieder verwenden zu müssen, erstelle ich einen Text-Component, welcher die benötigte Logik für die Größenveränderung von 100% zu 200% beinhaltet. Dieser Component kommt in meinem Projekt immer dann zum Einsatz, wenn ich irgendwo Text einfügen möchte. Das Feature der Farbumkehrung für ein besseres Kontrastverhältnis setzte ich mit einer Verbindung aus Svelte und TailwindCSS um, ich nutze If-Verzweigungen in den HTML-Elementen um verschiedene Farben bzw. Themes zu implementieren. Diese Themes werden dann durch deinen Button getauscht.

Nach dem erfolgreichen Testen der Website, schloss ich diesen Abschnitt der Seite und hatte Version 1.0 meines Projektes.

### 5.3 Überprüfung der Barrierefreiheit

### 5.4 Bewertung des Mehraufwandes

## 6 Fazit



# 7 Anhang

## 7.1 Anh. 1: Umfrage Entwickler

Um möglichst praxisnah auf die in meiner Facharbeit zu behandelnden Möglichkeiten zur Herstellung von barrierefreien Websites einzugehen, habe ich eine Umfrage unter Entwicklern durchgeführt. Dies ermöglichte mir einen realistischen Einblick in die aktuelle Situation zu erhalten.

### Fragen:

1. Zu welcher Altersgruppe gehören Sie?\*

- ☐ unter 25 Jahre
- ☐ 25 - 34 Jahre
- ☐ 35 - 44 Jahre
- ☐ 45 - 54 Jahre
- ☐ 55 -64 Jahre
- ☐ 65 Jahre oder älter

2. In welchem Bereich der Informatik/ IT arbeiten Sie?\*

\_\_\_\_\_

3. Inwiefern spielt das Thema Barrierefreiheit in Ihrer Entwicklungsarbeit eine Rolle?\*

0      1      2      3      4      5      6      7      8      9      10

4. Hatten Sie schon mal Kontakt mit Barrierefreiheit im Web während Ihrer Arbeit? (wenn ja, wobei?)

\_\_\_\_\_

5. Wie bedeutsam sind diese Tools und Optionen zur Barrierefreiheit im Web aus Ihrer Sicht?\*

- Responsive Design

- Starke Farbkontraste
- unterschiedliche Schriftgrößen
- Screenreaders

6. Wie wichtig ist Barrierefreiheit im Vergleich zu Funktionalität und Sicherheit auf Websites?

0      1      2      3      4      5      6      7      8      9      10

7. Welche Herausforderungen sehen Sie bei der Umsetzung von Barrierefreiheit auf Websites, und wie überwinden Sie diese?\*

---

## 8 Literaturverzeichnis

- [1] Mozilla. 2024. "What is accessibility? - Learn web development | MDN." MDN Web Docs.  
[https://developer.mozilla.org/en-US/docs/Learn/Accessibility/What\\_is\\_accessibility#people\\_with\\_visual\\_impairments](https://developer.mozilla.org/en-US/docs/Learn/Accessibility/What_is_accessibility#people_with_visual_impairments).
- [2] Probiesch, Kerstin. 2013. *Barrierefreiheit im Web*. N.p.: RKW Kompetenzzentrum.  
<https://www.rkw-kompetenzzentrum.de/fileadmin/media/Produkte/2013/Faktenblatt/20130128-Barrierefreiheit-im-Web.pdf>.
- [3] Bundesamt für Justiz. n.d. "BITV 2.0 - Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz." Gesetze im Internet. Accessed February 14, 2024.  
[https://www.gesetze-im-internet.de/bitv\\_2\\_0/BJNR184300011.html](https://www.gesetze-im-internet.de/bitv_2_0/BJNR184300011.html).
- [4] Sieger, Volker. n.d. "Die neue BITV 2.0." Bundesfachstelle Barrierefreiheit. Accessed February 14, 2024.  
[https://www.bundesfachstelle-barrierefreiheit.de/DE/Fachwissen/Informationstechnik/EU-Webseitenrichtlinie/BGG-und-BITV-2-0/Die-neue-BITV-2-0/die-neue-bitv-2-0\\_node.html#doc7b61b41e-e7fa-4086-9857-1ab9e2ac0b6ebodyText1](https://www.bundesfachstelle-barrierefreiheit.de/DE/Fachwissen/Informationstechnik/EU-Webseitenrichtlinie/BGG-und-BITV-2-0/Die-neue-BITV-2-0/die-neue-bitv-2-0_node.html#doc7b61b41e-e7fa-4086-9857-1ab9e2ac0b6ebodyText1).
- [5] ETSI. 2021. "ETSI EN 301 549 - V3.2.1 - Accessibility requirements for ICT products and services." ETSI EN 301 549 - V3.2.1 - Accessibility requirements for ICT products and services.  
[https://www.etsi.org/deliver/etsi\\_en/301500\\_301599/301549/03.02.01\\_60/en\\_301549v030201p.pdf](https://www.etsi.org/deliver/etsi_en/301500_301599/301549/03.02.01_60/en_301549v030201p.pdf).
- [6] pepper — Agentur für Webdesign, Print und Marketing. n.d. "Checkliste für barrierefreies Webdesign." barrierefreies.design. Accessed February 14, 2024.  
<https://barrierefreies.design/barrierefreiheit-interaktiv-testen/checkliste-fur-barrierefreies-webdesign#checkliste>.
- [7] Kollmann, Tobias, and Alexander Michaelis. 2015. "Responsive Webdesign." In *WiSt Heft*, 406-409. 7th ed. N.p.: WiSt - Wirtschaftswissenschaftliches Studium.  
[https://web.archive.org/web/20220223150023id\\_/https://www.beck-elibrary.de/10.15358/0340-1650-2015-7-50.pdf](https://web.archive.org/web/20220223150023id_/https://www.beck-elibrary.de/10.15358/0340-1650-2015-7-50.pdf).
- [8] TailwindCSS. n.d. "Screen Readers." Tailwind CSS. Accessed February 14, 2024. <https://tailwindcss.com/docs/screen-readers>.

- [9] ILIAS. n.d. "ILU: 4. Übersicht verschiedener assistiver Technologien." ILU. Accessed February 14, 2024. [https://ilu.th-koeln.de/goto.php?target=cat\\_75998](https://ilu.th-koeln.de/goto.php?target=cat_75998).
- [10] TailwindCSS. n.d. "Responsive Design." Tailwind CSS. Accessed February 16, 2024. <https://tailwindcss.com/docs/responsive-design>.
- [11] selfhtml. 2024. "Framework – SELFHTML-Wiki." SelfHTML. <https://wiki.selfhtml.org/wiki/Framework>.
- [12] DataScientest. (2022, October 13). GitHub: Was steckt dahinter und wie lerne ich den Umgang damit? DataScientest.com. Retrieved February 18, 2024, from <https://datascientest.com/de/github>
- [13] twigbit. (n.d.). Vercel - Einfach erklärt. Twigbit Technologies. Retrieved February 18, 2024, from <https://www.twigbit.com/glossar/vercel>

**LETZTES MAL AUFGERUFEN FEHLT!**

## 9 Selbstständigkeitserklärung

Hiermit versichere ich, dass ich die Arbeit selbständig und nur unter Verwendung der angegebenen Hilfsmittel verfasst habe.

A handwritten signature in black ink, reading "L. Taube", with a horizontal line underneath.

Louis Taube

Leipzig, 29.02.2024