

Zielgröße

Aktive Elemente benötigen eine mindestgröße oder ausreichende Abstände zu benachbarten aktiven Elementen

Autor: Jan Hellbusch
Quelle: <https://www.hellbusch.de/zielgroesse/>

Bereits in den Web Content Accessibility Guidelines (WCAG) 2.1 wurden Anforderungen für die Zielgröße von Bestandteilen der Benutzerschnittstelle auf 44px × 44px festgelegt. Das neue Erfolgskriterium 2.5.8 in den WCAG 2.2 legt eine Mindestgröße für die Zielgröße mit 24px × 24px fest. Es erlaubt auch kleinere Zielgrößen, wenn die Bestandteile der Benutzerschnittstelle ausreichend Abstand zu benachbarten Bestandteilen haben.

Anforderungen (WCAG 2.2)

Success Criterion 2.5.8 Target Size (Minimum)

(Level AA)

The size of the target for pointer inputs is at least 24 by 24 CSS pixels, except where:

- Spacing: Undersized targets (those less than 24 by 24 CSS pixels) are positioned so that if a 24 CSS pixel diameter circle is centered on the bounding box of each, the circles do not intersect another target or the circle for another undersized target;
- Equivalent: The function can be achieved through a different control on the same page that meets this criterion;
- Inline: The target is in a sentence or its size is otherwise constrained by the line-height of non-target text;
- User agent control: The size of the target is determined by the user agent and is not modified by the author;
- Essential: A particular presentation of the target is essential or is legally required for the information being conveyed.

Note: Targets that allow for values to be selected spatially based on position within the target are considered one target for the purpose of the success criterion. Examples include sliders, color pickers displaying a gradient of colors, or editable areas where you position the cursor.

Note: For inline targets the line-height should be interpreted as perpendicular to the flow of text. For example, in a language displayed vertically, the line-height would be horizontal.

Success Criterion 2.5.5 Target Size (Enhanced)

2

(Level AAA)

The size of the target for pointer inputs is at least 44 by 44 CSS pixels except when:

Equivalent

The target is available through an equivalent link or control on the same page that is at least 44 by 44 CSS pixels;

Inline

The target is in a sentence or block of text;

User Agent Control

The size of the target is determined by the user agent and is not modified by the author;

Essential

A particular presentation of the target is essential to the information being conveyed.

Übersetzungen

Erfolgskriterium 2.5.8 Zielgröße (Minimum)

(Stufe AA)

Die Größe des Ziels für Zeigereingaben beträgt mindestens 24 mal 24 CSS-Pixel, außer wo:

- Abstand: Zu kleine Ziele (mit weniger als 24 mal 24 CSS-Pixel) werden so positioniert, dass, wenn ein Kreis mit einem Durchmesser von 24 CSS-Pixel auf dem Begrenzungsrahmen jedes Ziels zentriert wird, die Kreise kein anderes Ziel oder den Kreis für ein anderes zu kleines Ziel überschneiden;
- Äquivalent: Die Funktion kann durch ein anderes Steuerelement auf derselben Seite erreicht werden, das dieses Kriterium erfüllt;
- Inline: Das Ziel befindet sich in einem Satz, oder seine Größe wird anderweitig durch den Zeilenabstand von Nicht-Ziel-Text eingeschränkt;
- Steuerung durch den Benutzeragenten: Die Größe des Ziels wird vom Benutzeragenten bestimmt und vom Autor nicht verändert;
- Unentbehrlich: Eine bestimmte Präsentation des Ziels ist unentbehrlich oder gesetzlich vorgeschrieben für die zu vermittelnde Information.

Hinweis: Ziele, die die räumliche Auswahl von Werten anhand der Position innerhalb des Ziels ermöglichen, werden für die Zwecke des Erfolgskriteriums als ein Ziel betrachtet. Beispiele schließen Schieberegler, Farbwähler, die einen Farbverlauf anzeigen, oder bearbeitbare Flächen, in denen der Cursor positioniert wird, ein.

Hinweis: Für Inline-Ziele sollte der Zeilenabstand als senkrecht zum Textfluss ausgelegt werden. Bei einer vertikal dargestellten Sprache ist der Zeilenabstand zum Beispiel horizontal.

Erfolgskriterium 2.5.5 Zielgröße (erweitert)

(Stufe AAA)

Die Größe des Ziels für Zeigereingaben beträgt mindestens 44 mal 44 CSS-Pixel. Dabei gelten folgende Ausnahmen:

Äquivalent

Das Ziel ist über einen äquivalenten Link oder ein Steuerelement auf derselben Seite verfügbar, das mindestens 44 mal 44 CSS-Pixel groß ist;

Inline

Das Ziel befindet sich in einem Satz oder Textblock;

Steuerung durch den Benutzeragenten

Die Größe des Ziels wird vom Benutzeragenten bestimmt und vom Autor nicht verändert;

Unentbehrlich

Eine bestimmte Darstellung des Ziels ist unentbehrlich für die zu vermittelnde Information.

Erläuterungen

Die Zielgröße ist die Fläche eines Bestandteils der Benutzerschnittstelle, mit der Nutzerinnen mit einem Zeigegerät interagieren können. Diese Fläche muss ausreichend groß sein, damit sie von allen zuverlässig getroffen werden kann.

Nutzende mit Bewegungsstörungen können Schwierigkeiten haben, kleine Ziele mit einem Zeigegerät oder ihrem Finger anzuvisieren. Wenn Bestandteile der Benutzerschnittstelle zu klein dimensioniert und zu nah beieinander positioniert werden, besteht die Gefahr, einen benachbarten Bestandteil versehentlich zu aktivieren. Die Zielgrößen sollten auch deswegen größer sein, weil die Präzision eines Fingers (auf einem Touch-Screen) grober ist als mit einem Mauszeiger.

Sie sollten generell zu kleine Bestandteile der Benutzerschnittstelle vermeiden, was insbesondere dann gilt, wenn:

- der Bestandteil häufig genutzt werden muss,
- die durch den Bestandteil ausgelöste Funktionalität schwer oder nicht rückgängig gemacht werden kann,
- der Bestandteil schwer erreichbar ist, weil z.B. vorher gescrollt werden muss, oder
- der Bestandteil Teil einer Aufgabe ist.

Der Einsatz von Abständen muss im Zusammenhang mit Zielgrößen sorgfältig gestaltet werden. Einerseits haben vor allem Geräte mit einem Touchscreen Algorithmen, um bei Berührung den nächstgelegenen Bestandteil der Benutzerschnittstelle zu ermitteln. Ein zu geringer Abstand zwischen Bestandteilen der Benutzerschnittstelle kann aber immer noch bedeuten, dass ein anvisierter Bestandteil nicht getroffen wird. Andererseits können größere Abstände zwischen Bestandteilen dazu führen, dass die visuelle Beziehung zwischen einzelnen Bestandteilen verloren geht.

Die WCAG 2.2 bieten zwei Erfolgskriterien zur Mindestgröße von Bestandteilen der Benutzerschnittstelle:

1. Auf Konformitätsstufe AA (Erfolgskriterium 2.5.8) muss die Zielgröße mindestens 24 CSS Pixel breit und 24 CSS Pixel hoch sein. Die Bestandteile dürfen auch kleiner sein, wenn der Abstand zu benachbarten Bestandteilen groß genug ist.
2. Auf Konformitätsstufe AAA (Erfolgskriterium 2.5.5) soll die Zielgröße mindestens 44 CSS Pixel breit und 44 CSS Pixel hoch sein.

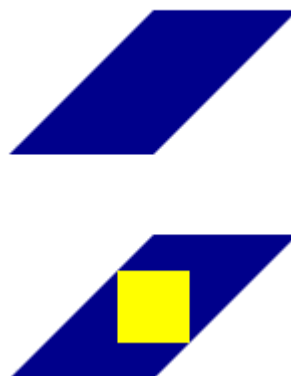
Erfolgskriterium 2.5.5 war bereits Teil von WCAG 2.1 und wird im Folgenden nicht behandelt.

Mindestgrößen

Die Mindestgröße für einen Bestandteil der Benutzerschnittstelle ist ein 24px breites Quadrat. Drei Bestandteile können ohne Abstand zueinander relativ kompakt dargestellt werden:



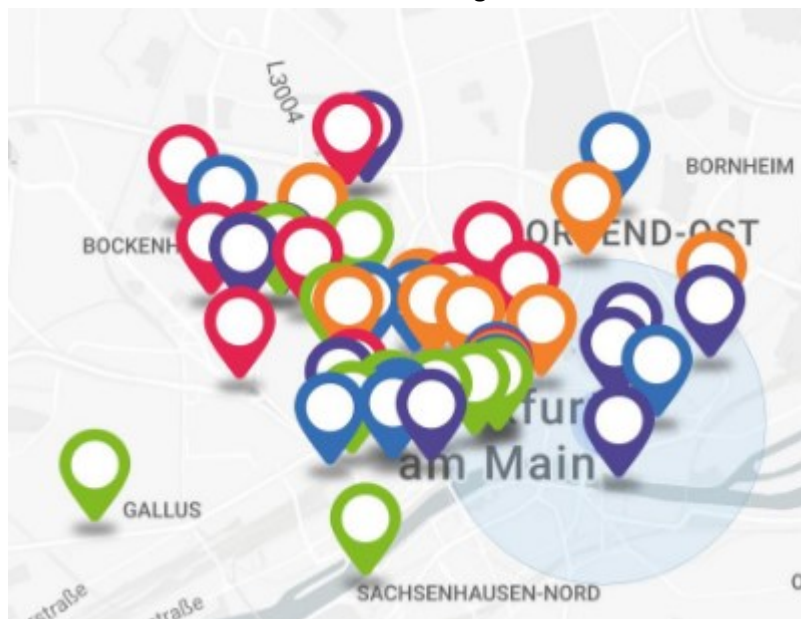
Wenn Bestandteile der Benutzerschnittstelle anders gestaltet werden, dann muss der Begrenzungsrahmen (die quadratische Zielfläche) vollständig innerhalb des Bestandteils passen. Die erläuternden Dokumente zu den WCAG 2.2 beschreiben weiter, dass die Kanten des Quadrats vertikal bzw. horizontal sein müssen. Ob ein Parallelogramm oder ein Kreis, eine 24pxx24px große Zielgröße muss vollständig enthalten sein:



Das Erfolgskriterium 2.5.8 weist auf mehrere Ausnahmen für Mindestzielgrößen hin.

5

- Beispielsweise könnte es auf einer Webseite zwei verschiedene Bestandteile der Benutzerschnittstelle mit gleicher Funktionalität geben. Der eine mit ausreichender Zielgröße, der andere mit zu geringer Zielgröße. In diesem Fall ist die zu geringe Zielgröße erlaubt, weil die Bedienbarkeit durch den alternativen Bestandteil gegeben ist.
- Manchmal ist eine bestimmte Darstellung essentiell, und dann ist das Erfolgskriterium nicht anwendbar. In einer Karte mit mehreren (kleinen) Schaltflächen zur Anzeige von Detailinformationen wird es nicht möglich sein, zusätzliche Abstände zu berücksichtigen. Das ist vor allem der Fall, wenn mehrere Pins übereinander liegen und zuerst in die Karte hineingezoomt werden muss. Eine Vergrößerung der Zielgröße dürfte die eigentliche Problematik, nämlich das versehentliche Aktivieren einer nicht anvisierten Schaltfläche, nicht lösen, weil die Schaltflächen immer übereinander liegen könnten.



Zielgröße für Links

Links müssen nur dann die Mindestzielgröße erreichen, wenn sie nicht Teil eines sonstigen Textes (Satz oder länger) sind. Der Grund dafür ist, dass die Höhe von Links im Fließtext sonst in vielen Fällen größer als die Zeilenhöhe bzw. Schriftgröße sein müsste.

Wenn Sie also in einem Absatz oder einer Liste Sätze mit Links haben, dürfen die Höhe und Breite der Links 24px unterschreiten. Anders sieht das aus bei Links in einer Linkliste. Wenn einzelne Wörter verlinkt werden, muss die Zielgröße der Links die Mindesthöhe erreichen:

```
<h2>Weitere Informationen zum Thema</h2>
<ul>
<li><a href="#">Zielgröße 24px</a></li>
<li><a href="#">Zielgröße 44px</a></li>
</ul>
```

Damit die Links die erforderliche Höhe erhalten, können Sie die Zeilenhöhe der Links erhöhen:

```
a{
line-height:24px;
display:inline-block;
}
```

Zielgröße für Formularelemente

Wenn Sie Bestandteile der Benutzerschnittstelle im Web anschauen, werden Sie schnell feststellen, dass viele Bestandteile die minimale Zielgröße nicht erreichen. Beispielsweise ist ein Kontrollfeld in Chrome standardmäßig 13px × 13px groß. Mit einer verknüpften Beschriftung wird die Zielgröße die erforderliche Mindestbreite erhalten, aber nicht unbedingt die erforderliche Mindesthöhe:

```
<p><label><input type="checkbox">Beschriftung</label></p>
```

Hinweis: Durch die Verknüpfung der Beschriftung kann das Kontrollfeld durch Klicken oder Tippen auf die Beschriftung aktiviert oder deaktiviert werden.

Ein Kontrollfeld mit einer Größe von 24px × 24px oder gar 44px × 44px wird in den WCAG 2.2 aber auch nicht verlangt. Vielmehr gilt:

1. Wenn Sie ein Bestandteil der Benutzerschnittstelle visuell ersetzen oder die Größe des standardmäßigen Erscheinungsbilds des Bestandteils verändern,
2. muss eine anklickbare Fläche von mindestens 24px × 24px bzw. soll eine anklickbare Fläche von 44px × 44px erreicht werden.

Formularelemente können unterschiedlich gut mit CSS gestaltet werden. Die Größenangaben für die Zielgröße müssen dann berücksichtigt werden, wenn Sie die Größe des Bestandteils verändern. Natürlich können Sie unabhängig davon versuchen, die Zielgröße von allen Formularelementen anzupassen, aber es wird Grenzen geben.

Eingabefelder und Schaltflächen haben vollständige Unterstützung für jede Eigenschaft des CSS Box Model. Die Größe der Steuerelemente können Sie immer verändern. Hierzu zählen die Eigenschaften width, height, padding, margin und border. Die folgenden Formularelemente können alle mit CSS vergrößert werden:

input,

textarea,

select,

button,

output {

width: 10rem;

padding: 0;

7

```
margin: 0;
box-sizing: border-box;
}
```

Es kommen auch andere CSS-Eigenschaften in Frage. Ein Kontrollfeld mit einer verknüpften Beschriftung kann z.B. durch die Zeilenhöhe eine ausreichende Zielgröße erhalten:

```
<p><label><input type="checkbox"> Beschriftung</label></p>
<style>
label {
display:inline-block;
line-height:44px;
}
</style>
```

Hinweis: Für dieses Beispiel ist die Veränderung der Zielgröße nicht erforderlich, weil der Browser die Größe des Kontrollfelds bestimmt.

In der Praxis wird es aber nicht alleine um die Zielgröße gehen. Die Formularelemente selbst übernehmen ein plattformeigenes Erscheinungsbild und können beispielsweise auf Apple-Systemen anders aussehen als auf Windows-Systemen. Während Eingabefelder und Schaltflächen meist mit den Eigenschaften des CSS box model sehr leicht angepasst werden können, gibt es einige Formularelemente, deren Erscheinungsbild praktisch ausgetauscht werden muss, um nach Belieben gestaltet werden zu können. Wenn Sie das Erscheinungsbild eines Formularelements durch Grafiken oder CSS ersetzen, dann müssen Sie sicherstellen, dass die Zielgrößen ausreichend groß sind.

Eine nützliche CSS-Eigenschaft für die Umgestaltung von Formularelementen ist `appearance`. Diese Eigenschaft wird eingesetzt, um plattformspezifische Stileigenschaften von einem Element zu entfernen. Das Element bleibt auf der Webseite zugänglich, ist aber nicht sichtbar. Beispielsweise kann das Erscheinungsbild eines Kontrollfelds visuell unterdrückt werden. Das können Sie tun, wenn Sie das Kontrollfeld mit anderen CSS-Eigenschaften anders gestalten wollen. Das folgende Kontrollfeld wollen wir nun umgestalten:

```
<p><label><input type="checkbox" name="nachtisch" value="himbeere">Ich möchte Himbeeren zum Nachtisch</label></p>
```

Das Kontrollfeld wird visuell durch ein selbst gestaltetes graues Quadrat ersetzt. Dazu kommt ein grüner Haken „✓“, den wir erst für alle Nutzenden verstecken und nur zugänglich machen, wenn die Checkbox aktiviert wird. Sie können auch ein anderes passendes Zeichen oder eine Grafik einsetzen:

```
input[type="checkbox"] {
appearance: none;
position: relative;
width: 1.1em;
```

8

```
height: 1.1em;
margin: 0 5px 0 0;
border: 1px solid gray;
color: green;
}
input[type="checkbox"]::before {
content: "✓";
position: absolute;
font-size: 1.1em;
top: -0.3em;
visibility: hidden;
}
input[type="checkbox"]:checked::before {
visibility: visible;
}
```

Im vorstehenden Beispiel ist das Kontrollfeld größer als die standardmäßige Darstellung eines Kontrollfelds, aber die Höhe des Kontrollfelds beträgt keine 24px. Vielmehr geht es in Erfolgskriterium 2.5.8 um die anklickbare Fläche, die hier mit der `line-height`-Eigenschaft bestimmt wird.

Während Kontrollfelder oder Auswahlschalter mit der `appearance`-Eigenschaft ein neues Erscheinungsbild mit einfachen Mitteln bekommen können, gilt das nicht für alle Formularelemente. Das Erscheinungsbild mancher Bestandteile wie der Anfasser eines Schiebereglers oder die Inkrementierungsfunktionen in einem Drehschalter stellen Herausforderungen dar, die nicht alleine mit CSS gelöst werden können, weil die Elemente im Shadow DOM stehen. In solchen Fällen können die Bestandteile der Benutzerschnittstelle mit JavaScript und ARIA nachgebaut werden, um ein neues Erscheinungsbild zu erhalten. Wenn Sie das Erscheinungsbild eines Formularelements ersetzen, dann müssen Sie sicherstellen, dass die Zielgrößen ausreichend groß sind.

Beispiele für schwierig zu gestaltende Formularelemente sind:

- Das `option`-Element (für `select`- und `datalist`-Elemente) können Sie nicht wirklich gestalten. Die Größe der Elemente ist per CSS nicht beeinflussbar, sondern nur die Schriftgröße.
- Das einzige Problem mit Datei-Uploads ist die Schaltfläche, um den Datei-Manager zu öffnen. Die Größe oder Farbe der Schaltfläche kann nicht geändert werden. Sie können das Problem

umgehen, wenn die Schaltfläche unsichtbar gestaltet wird (z.B. eine Größe von 1px × 1px) und eine Beschriftung mit dem label-Element erhält; damit können Sie das Erscheinungsbild verändern und die Zielgröße erhöhen.

- Drehfelder (`<input type="number">`) generieren eine Inkrementierungsfunktion (ein Pfeil nach oben und ein Pfeil nach unten). Diese beiden Steuerelemente können nicht browserübergreifend gestaltet werden.
- Schieberegler sind schwierig zu gestalten. Der Anfasser kann wie bei Kontrollfelder mit `appearance:none`; ausgeblendet und durch andere CSS-Eigenschaften größer dargestellt werden.
- Datums- und Zeitfelder erzeugen weitere Elemente, insbesondere Drehfelder und eine Schaltfläche für einen Date-Picker. Sie können die Drehfelder mit CSS gestalten, aber die Inkrementierungsfunktion der Drehfelder können Sie nicht gut gestalten. Auch an den Date-Picker kommen Sie mit CSS nicht ran.

Für diese Formularelemente ist der Nachbau mit JavaScript und ARIA sinnvoll, wenn die Darstellung einschließlich der Zielgröße verändert werden soll.

Das Erfolgskriterium beschreibt weitere Ausnahmen. In manchen Situationen ist die Unterschreitung der minimalen Zielgrößen unentbehrlich. Diese Situationen setzen praktisch eine Feinmotorik für die Bedienung voraus. Bestandteile dieser Art werden im Sinne der Erfolgskriterien als ein Ziel verstanden. Zu solchen Bestandteilen zählen:

- Bestandteile, die die Einstellung eines Werts auf einer Fläche bereitstellen, etwa ein Farbwähler.
- Schieberegler, die eine (praktisch stufenlose) Einstellung eines Wertes erlauben oder
- Eingabefelder, in denen der Cursor per Zeigegerät an einer bestimmten Stelle positioniert werden kann.

Die Anforderungen an Zielgrößen können für diese Situationen nicht umgesetzt werden. Die Auswahl bzw. Positionierung innerhalb des Bestandteils muss über andere Techniken möglich sein, etwa über Schaltflächen oder Pfeiltasten auf einer Bildschirmtastatur.

Abstände bei zu kleinen Zielgrößen




Abstände zwischen Bestandteilen der Benutzerschnittstelle können helfen, Fehler bei der Interaktion zu vermeiden. Insbesondere wenn Bestandteile die minimale Zielgröße von 24px × 24px unterschreiten, müssen Sie sogar Abstände berücksichtigen, damit die Bestandteile fehlerfrei bedient werden können.

Bestandteile der Benutzerschnittstelle dürfen eine Höhe oder Breite von weniger als 24px haben, wenn der Abstand zu benachbarten Bestandteilen ausreichend groß ist. Die Ausnahme wird mit überlappenden Kreisen beschrieben. Wenn Sie auf jeden Bestandteil, der die minimale Zielgröße unterschreitet, einen Kreis mit einem Durchmesser von 24px zentrieren, dürfen diese Kreise einander oder einen benachbarten, ausreichend großen Bestandteil nicht überlappen.

Hinweis: Die Ausnahme erlaubt winzig kleine Zielgrößen, die sogar überhaupt keine Abmessungen haben (0px×0px). Es handelt sich um eine bekannte Lücke in den WCAG 2.2, die nicht beseitigt

werden kann. Wenn die Ausnahme eine neue Zielgröße definieren würde, zum Beispiel 10px × 10px, würde sie faktisch zur neuen minimalen Zielgröße werden.

In der nachfolgenden Grafik werden verschiedene Quadrate mit unterschiedlichen Größen und Abständen dargestellt:

	Mindestgröße
	Kleiner als Mindestgröße, aber mit Abständen (konform)
	Kleiner als Mindestgröße und ohne ausreichende Abstände (nicht konform)

Die erste Reihe zeigt Bestandteile der Benutzerschnittstelle mit 24px × 24px ohne Abstände zwischen den Bestandteilen der Benutzerschnittstelle. Diese Zielgrößen erfüllen Erfolgskriterium 2.5.8 auf Konformitätsstufe AA.

Die zweite Reihe zeigt Bestandteile der Benutzerschnittstelle mit 20px × 20px mit einem Abstand von 4px zwischen den Bestandteilen der Benutzerschnittstelle. Diese Zielgrößen erfüllen Erfolgskriterium 2.5.8 auf Konformitätsstufe AA.

Die letzte Reihe zeigt Bestandteile der Benutzerschnittstelle mit 20px × 20px ohne Abstände zwischen den Bestandteilen der Benutzerschnittstelle. Diese Zielgrößen sind nicht WCAG-konform.

Nach Erfolgskriterium 2.5.8 ist bei dieser Ausnahmeregelung wie folgt vorzugehen:

1. Ermitteln Sie den Begrenzungsrahmen für jeden Bestandteil der Benutzerschnittstelle, der weniger als 24px hoch und/oder weniger als 24px breit ist. Bestimmen Sie den Mittelpunkt des Rechtecks.
2. Für jeden Bestandteil der Benutzerschnittstelle mit zu geringer Zielgröße legen Sie ein Kreis mit einem Durchmesser von 24px so, dass der Mittelpunkt des Kreises mit dem Mittelpunkt des Rechtecks übereinstimmt.
3. Das Erfolgskriterium 2.5.8 ist erfüllt, wenn es keine Schnittmengen zwischen den Kreisen oder zwischen einem Kreis und einem anderen Bestandteil der Benutzerschnittstelle gibt.

Die folgenden Beispiele zeigen Bestandteile der Benutzerschnittstelle mit zu kleiner Zielgröße. Für jeden Bestandteil wird ein Kreis mit 24px über den Begrenzungsrahmen gelegt:

- Die Vorgehensweise ist bei rechteckigen Bestandteilen leicht nachzuvollziehen, denn die Mittelpunkte von Begrenzungsrahmen und Kreis sind gleich. Im folgenden Menü sind die vertikalen Abstände nicht ausreichend, weil die einzelnen Menüeinträge nur 16px hoch sind. Die

zentrierten Kreise auf den Menüeinträgen haben Schnittmengen:

11



- Wenn das Ziel nicht rechteckig ist, z.B. weil es abgeschnitten wird, runde Ecken oder eine komplexere Form hat, dann kann bei der Feststellung des Begrenzungsrahmens der Mittelpunkt an unerwarteter Stelle liegen. Beispielsweise kann bei konkaven Formen der Mittelpunkt außerhalb des sichtbaren Ziels liegen. Der Abstand von 24px zu allen benachbarten Bestandteilen der Benutzerschnittstelle muss vom Mittelpunkt des Ziels berechnet werden (ein Beispiel finden Sie auf der Seite des W3C zu Erfolgskriterium 2.5.8, siehe unten).

Bezogen auf Abstände bei zu kleinen Zielgrößen werden Nutzerinnen mit verschiedenen Behinderungen unterschiedliche Bedürfnisse haben. Menschen mit Gesichtsfeldausfällen werden mit kleinen Zielgrößen vielleicht gut klar kommen während bei Vorliegen einer anderen Sehbehinderung eine Vergrößerung der sichtbaren Ziele besser ist. Es ist vor allem für die Bedienung auf Touchscreens vorteilhaft, wenn die Zielgröße groß genug ist, auch wenn das sichtbare Ziel selbst unterhalb der minimalen Zielgröße bleibt. Optional können Webseiten Optionen in Form eines Style-Switchers anbieten, der die Vergrößerung der Zielgrößen bzw. der sichtbaren Ziele erlaubt.

Weiterführende Links

- Erläuterungen zu Erfolgskriterium 2.5.8 beim W3C:
<https://www.w3.org/WAI/WCAG22/Understanding/target-size-minimum.html>
- Neun neue Kriterien, eine Übersicht der Änderungen in den WCAG 2.2 gegenüber WCAG 2.1:
<https://www.hellbusch.de/neun-neue-kriterien/>

Hellbusch Accessibility Consulting

12

- Tests und Gutachten zur Konformität Ihrer Webseiten, Apps und Software zu den Barrierefreiheitsrichtlinien.
- Überarbeitung Ihrer nicht barrierefreien PDF-Dokumente und -Formulare in PDF/UA-Qualität.
- Schulungen für die barrierefreie Gestaltung von Webseiten und PDF-Dokumenten.



Schauen Sie vorbei auf <https://www.barrierefreies-webdesign.de>.

Nehmen Sie Kontakt auf: jan@hellbusch.de oder +49 (163) 3369925