



# Easy Web Check

Ein vereinfachtes Prüfverfahren mit  
kostenlosen Werkzeugen

Andreas Burkard & Gottfried Zimmermann

Kompetenzzentrum Digitale Barrierefreiheit

Hochschule der Medien, Stuttgart

# Agenda

---

- Was ist der „Easy Web Check“?
- Prüfmethode
- Werkzeugliste
- Checkliste Live-Demo
- Vor- und Nachteile
- Abdeckung der Kriterien
- Einblicke in „Easy Web Check: Eine vereinfachte Prüfmethode zur Barrierefreiheit im Web“
- Fragen und Antworten

# Was ist der „Easy Web Check“?



- **Vereinfachte Prüfmethode**
  - Fokus auf einfach zu findende Barrieren („low hanging fruits“)
  - Keine technischen Kenntnisse in Webdesign oder –programmierung erforderlich
- Die meisten (nicht alle) Kriterien nach BITV 2.0 werden überprüft [1]
  - 52 von 94 verpflichtende Kriterien werden berücksichtigt
  - Manche Kriterien werden nur unvollständig geprüft (6 von 94)
- **Halbautomatische Prüfung**
  - Automatische Tools (Desktop-Anwendungen, Browser-Erweiterungen)
  - Zusätzliche manuelle Sichtprüfungen

# Prüfmethode (1)

---

Checkliste enthält leicht verständliche Anweisungen:

## 1. Werkzeuge aus Werkzeugliste herunterladen

- Browsererweiterungen und Desktop Applikationen
- Alle Tools sind gratis

## 2. Automatische Prüfung

- Etwas weniger als ein Drittel der Barrieren werden automatisch durch Werkzeuge gefunden
- Checkliste / Mini-Workshop bietet einfache Erklärungen zu gefundenen Barrieren

# Prüfmethode (2)

---

## 3. Toolunterstützte Prüfung

- Weitere Barrieren werden durch unterstützende Tools gefunden
- Checkliste / Mini-Workshop erklärt, wie man die Tools einsetzen soll

## 4. Manuelle geleitete Prüfung

- Checkliste erklärt in einfachen Worten, ohne Programmierkenntnisse, wie man ohne Tools Barrieren findet.

# Werkzeugliste (1)

- "Siteimprove Accessibility Checker" Browser Extension  
([Google Chrome](#) | [Firefox](#))
- Bookmarklet "Inhalte gegliedert"  
aus der [BIK BITV Werkzeugliste Bookmarklets](#)
- "Lists Bookmarklet" aus der [Paul J. Adam's Bookmarklet-Liste](#)
- "Tables Bookmarklet" aus der [Paul J. Adam's Bookmarklet-Liste](#)
- "taba11y" ([für Google Chrome](#))
- "Images Bookmarklet" aus [Paul J. Adam's Bookmarklet-Liste](#)
- "Lang Bookmarklet" aus der Tabelle in der [Paul J. Adam's Bookmarklet-Liste](#)

# Werkzeugliste (2)



- Browserextension "Web Developer" ([Google Chrome](#) | [Firefox](#))
- WCAG Color contrast checker ([Google Chrome](#) | [Firefox](#))
- [Colour Contrast Analyser](#) von TPGi
- Bookmarklet "Textabstände"  
aus der [BIK BITV-Test Werkzeugliste Bookmarklets](#)
- Browser Extension "Landmarks" ([Google Chrome](#) | [Firefox](#))
- "Autocomplete Attribute Favlet" Bookmarklet  
von [Level Access Web Labs](#)

# Checkliste Live-Demo

# Vor- und Nachteile



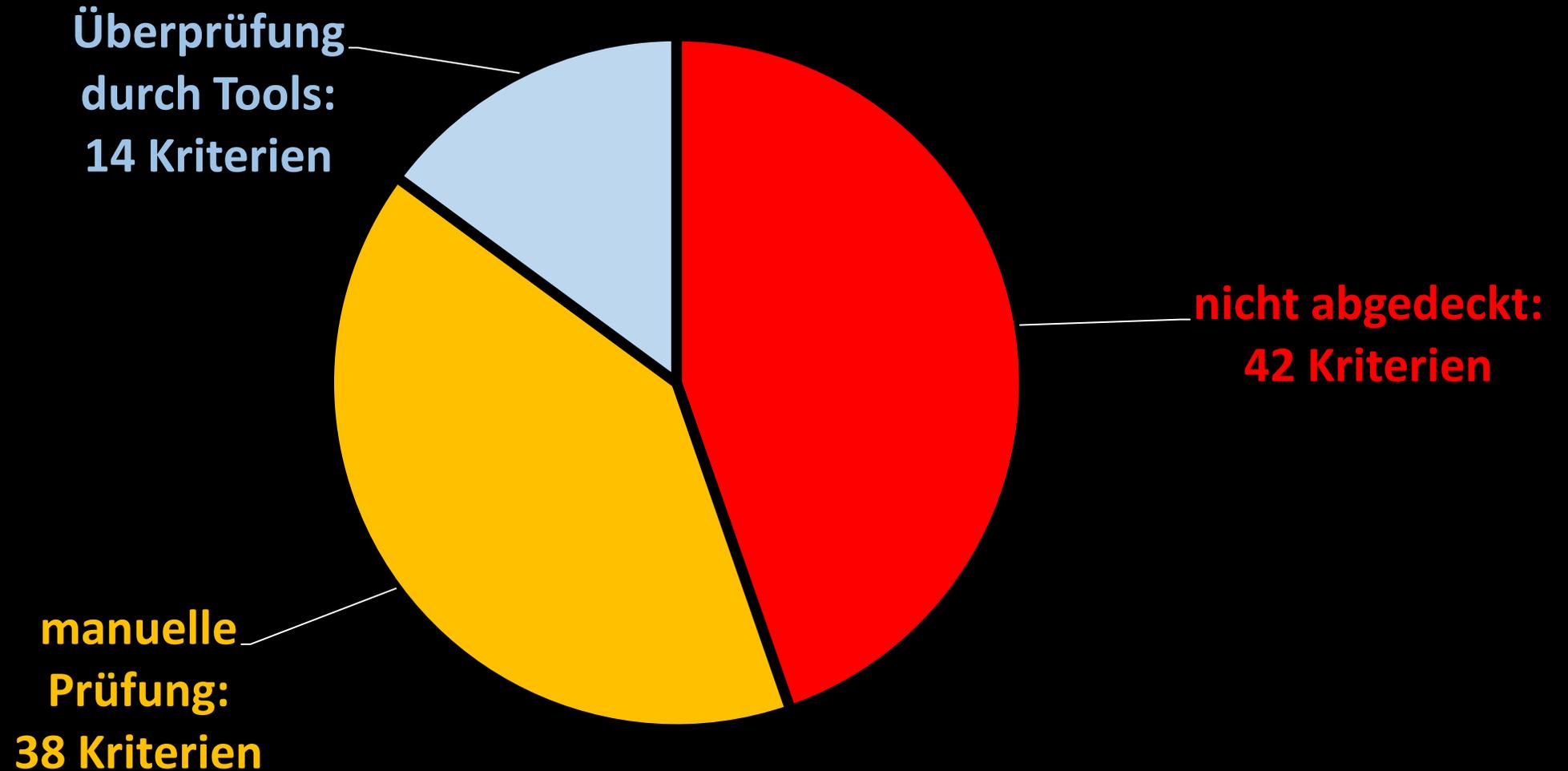
## Vorteile

- Schnell & effizient Barrieren finden
- Ohne Programmierkenntnisse
- Die Checkpoints sind einfach erklärt und durchführbar
- Anschauliche Beispiele (im Mini-Workshop)
- Teilweise konkrete Lösungsvorschläge

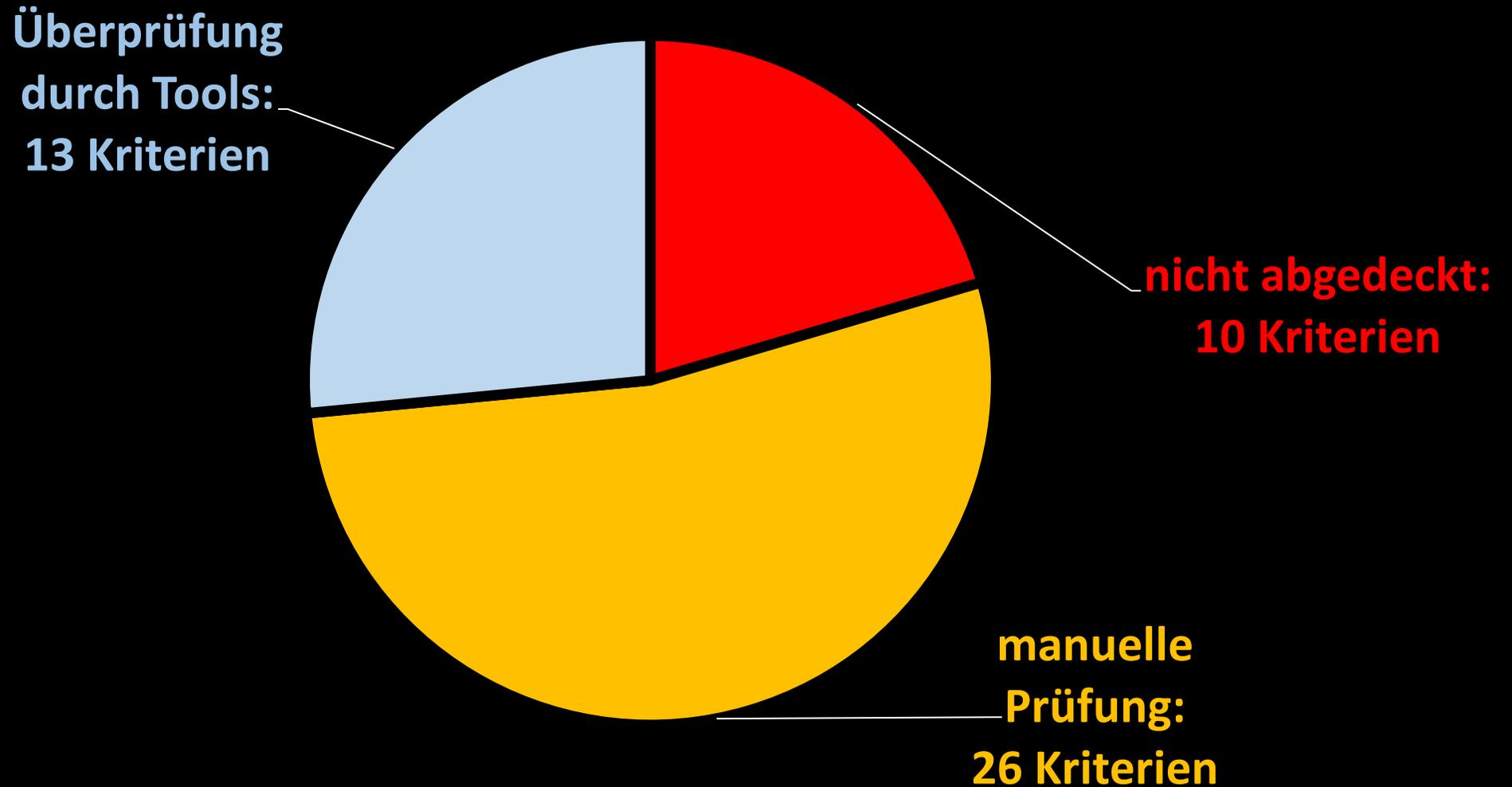
## Nachteile

- Es werden nicht alle Barrieren gefunden
- Kein Ersatz für Expertenprüfung
- Manche Kriterien werden nur teilweise überprüft
- Nicht alle Sonderfälle werden abgedeckt

# Kriterien-Abdeckung Easy Web Check BITV 2.0



# Kriterien-Abdeckung Easy Web Check WCAG 2.1



# Automatische Prüfung

Beispiele aus dem Mini-Workshop: „Easy Web Check: Eine vereinfachte Prüfmethode zur Barrierefreiheit im Web“

# Siteimprove Accessibility Checker



- "Siteimprove Accessibility Checker" Browser Extension ([Chrome](#) | [Firefox](#)) ist ein Tool zur automatischen Überprüfung auf Barrierefreiheit. Nach der Installation soll man sich anmelden, drücken Sie stattdessen "Skip form".

**E-mail**

Yes, I agree to receive emails about Siteimprove products and services. You may unsubscribe at any time. (Optional)

I have read and accept the [Privacy Policy](#).

Erweiterung: (Siteimprove Accessibility Che... — □ ×

[Matrimony, Marriage, Free Matrimonial Sites, Match Making](#)

**Siteimprove**  
Accessibility Checker

Issues Explorer

Issues 11

[Autocomplete does not work as intended](#) → 1

**AA** 1.3.5 Identify Input Purpose

[Link without a text alternative](#) →

**A** 2.4.4 Link Purpose (In Context) 12

**AAA** 2.4.9 Link Purpose (Link Only)

**A** 4.1.2 Name, Role, Value

# Alternativtexte für Bilder

Bilder haben jeweils überhaupt einen Alternativtext (dieser darf auch leer sein, er darf nur nicht komplett fehlen).

**Fehlerbeispiel:** Das Bild hat keinen Alternativtext.

→ Screenreader-Nutzenden wird der Inhalt des Bildes nicht vorgelesen, stattdessen etwas wie: „Bild mit fehlender Bildbeschreibung“.

Image without a text alternative →

**A** 1.1.1 Non-text Content

5



# Alternativtexte für (Bild-)Links

(Bild-)Links oder grafische Bedienelemente haben jeweils entweder eine Beschriftung oder einen Alternativtext

**Fehlerbeispiel:** Der Bild-Link hat keinen Alternativtext.

→ Screenreader-Nutzenden wird das Ziel des Links nicht vorgelesen.

Link without a text alternative →

**A** 2.4.4 Link Purpose (In Context)

**AAA** 2.4.9 Link Purpose (Link Only)

**A** 4.1.2 Name, Role, Value

3



# Programmatisch verbundene Beschriftung von Formular-Elementen



Alle Formularelemente haben programmatisch korrekt verknüpfte Beschriftungen.

**Fehlerbeispiel:** Die Beschriftung „Kommentare“ ist nicht programmatisch mit dem Eingabefeld verknüpft.

→ Screenreader-Nutzenden wird nicht vorgelesen, wofür das Eingabefeld da ist.

Form field is not labeled →

**A** 4.1.2 Name, Role, Value

1

Abgabekommentare

Kommentare

X

Speichern

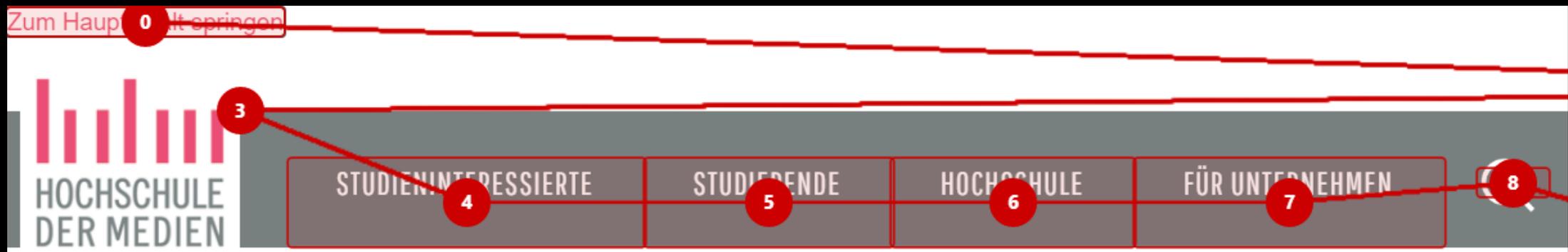
# Toolunterstützte manuelle Prüfung

Beispiele aus dem Mini-Workshop: „Easy Web Check: Eine vereinfachte Prüfmethode zur Barrierefreiheit im Web“

# taba11y Browser Extension



Die Browser Extension „taba11y“ (für Google Chrome) [1] ermöglicht es, die Tab-Reihenfolge mit der Tastatur anzuzeigen. Gleichzeitig sieht man auch, welche Elemente **nicht** mit der Tastatur fokussiert werden können und somit **nicht** tastaturbedienbar sind.



[1] <https://chrome.google.com/webstore/detail/tab11y/aocppmckdocdjkphmofnklcjhdidgmga>

# Fokussierbare Bedienelemente



Alle Bedienelemente sind mit der Tastatur fokussierbar (haben in „tab11y“ eine Markierung und Nummer).

**Fehlerbeispiel:** Die Akkordeon-Buttons sind nicht Teil der Tab-Reihenfolge und damit nicht fokussierbar und nicht mit der Tastatur bedienbar.

→ Bei vielen Einschränkungen und Behinderungen ist es nicht möglich, die Maus zu bedienen, die betroffenen Menschen sind auf Tastaturbedienung angewiesen.

Was ist das praktische Studiensekretariat (PS)?

Kann man während des Studiums ins Ausland gehen? Und wann geht das am besten?

**ADRESSE**

Hochschule der Medien  
Nobelstraße 10  
70569 Stuttgart  
0711 8903-10

**QUICKLINKS**

Intranet

Öffnungszeiten

19

21

22

# Sinnvolle Tab-Reihenfolge bei der Tastaturbedienung

Die Reihenfolge der Tab-Navigation (mit der Tastatur) ist sinnvoll.

**Fehlerbeispiel:** Während am Anfang die Reihenfolge noch konsistent ist mit: 9, 10, 13, wird ein Element, das sich direkt daneben befindet erst an 25ter Stelle fokussiert.



# Manuelle geleitete Prüfung

Beispiele aus dem Mini-Workshop: „Easy Web Check: Eine vereinfachte Prüfmethode zur Barrierefreiheit im Web“

# Dokumenttitel



Der Titel des Tabs (Dokumenttitel) enthält sowohl den Namen des Webauftritts als auch den der Unterseite, auf der man sich befindet.

**Fehlerbeispiel:** Der Dokumenttitel, welcher der Beschriftung des Tabs einer Webseite im Browser oben entspricht, ist in diesem Fall nur „Help“ und gibt damit die Unterseite des Webauftritts wieder, aber gibt fälschlicherweise keine Auskunft darüber, auf welcher Webseite man sich befindet.

**Lösungsvorschlag:** Korrekt wäre z. B.: „jeevansathi – Help“ und sollte in das `<title>`-Element, welches innerhalb des `<head>`-Elements zu finden ist, eingetragen werden.



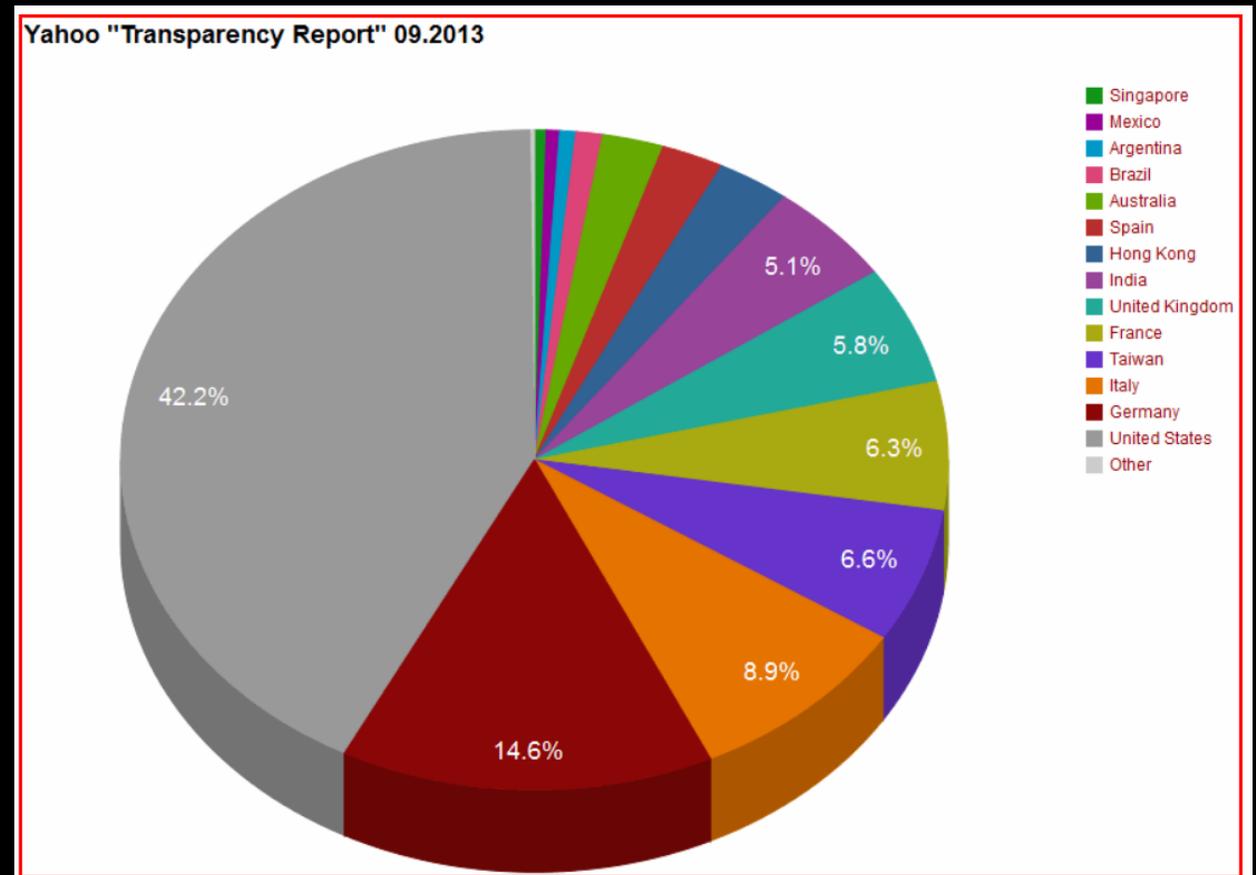
# Diagramme und Abbildungen ohne Farbwahrnehmung benutzbar

Diagramme oder Abbildungen sind auch ohne Farbwahrnehmung sinnvoll lesbar.

**Fehlerbeispiel:** Das Diagramm ist nur benutzbar, wenn man Farben unterscheiden kann.

**Lösungsvorschläge** wären zusätzlich:

- Muster einzusetzen
- Die Datensätze mit Strichen oder Pfeilen zu verbinden
- Das Diagramm innerhalb beschriften





Vielen Dank für Ihre  
Aufmerksamkeit!

Noch Fragen?

# Danke



Andreas Burkard

Prof. Dr. Gottfried Zimmermann

[barrierefreiheit@hdm-stuttgart.de](mailto:barrierefreiheit@hdm-stuttgart.de)

Kompetenzzentrum Digitale Barrierefreiheit

Hochschule der Medien, Stuttgart

Bestellen Sie unseren Newsletter:

[barrierefreiheit.hdm-stuttgart.de/newsletter](http://barrierefreiheit.hdm-stuttgart.de/newsletter)



**Alle Rechte vorbehalten**

# Nicht abgedeckte Kriterien (1)



- Alle Kriterien aus EN Kap. 6: IKT mit Zweiwege-Sprachkommunikation
- Alle Kriterien aus EN Kap. 11.8: Autorenwerkzeuge
- 5.2 Aktivierung von Barrierefreiheitsfunktionen
- 5.3 Biometrie
- 5.4 Erhaltung von Barrierefreiheitsinformationen bei Konvertierung
- 7.1.3 Erhaltung von Untertiteln
- 7.2.1 Wiedergabe von Audiodeskription
- 7.2.2 Synchrone Audiodeskription

# Nicht abgedeckte Kriterien (2)

---

- 9.1.3.2 Bedeutungsvolle Reihenfolge
- 9.1.3.3 Ohne Bezug auf sensorische Merkmale nutzbar
- 9.1.4.13 Eingblendete Inhalte bedienbar
- 9.2.1.4 Tastatur-Kurzbefehle abschaltbar oder anpassbar
- 9.3.2.1 Keine unerwartete Kontextänderung bei Fokus
- 9.3.2.2 Keine unerwartete Kontextänderung bei Eingabe
- 9.3.2.4 Konsistente Bezeichnung

# Nicht abgedeckte Kriterien (3)



- 9.3.3.4 Fehlervermeidung wird unterstützt (rechtlich, finanziell, Benutzerdaten)
- 9.4.1.2 Name, Rolle, Wert verfügbar
- 9.4.1.3 Statusmeldungen programmatisch verfügbar
- 9.6 WCAG Konformitätskriterien
- 11.7 Benutzerdefinierte Einstellungen
- 12.1.1 Dokumentation von Kompatibilität und Barrierefreiheit
- 12.2.2 Technischer Support