Konkrete Anforderungen

- Design und Struktur
- Technik
- Content Erstellung (Videos)
- Befüllung
- Externe Tools (z.B. Youtube, Vimeo)
- Testing

Design und Struktur

- Aktivierung von Barrierefreiheisfunktionen
  - Wenn die Webseite Funktionen für die Barrierefreiheit bereithält, soll die Aktivierung dieser Funktionen für diese Nutzergruppe barrierefrei möglich sein. Zum Beispiel: Vorlesefunktion; Kontrasterhöhung; abweichende Farb-schemata; Anpassung der Schriftgröße, Schriftformatierungen; Versionen in Leichter Sprache oder Deutscher Gebärdensprache
- Ohne Bezug auf sensorische Merkmale nutzbar
  - Verweise auf Inhalte der Seite sollen nicht ausschließlich sensorische Merkmale wie Farbe, Form oder Position nutzen, sondern auch ohne bestimmte Sinneswahrnehmungen verständlich sein. Der Bezug auf die Form, Farbe oder Position von bestimmten Seiteninhalten ist für blinde Nutzende und auch Nutzende, die die Seite ohne das mitge-lieferte Stylesheet oder auf anderen Ausgabegeräten sehen, nicht brauchbar.
- Ohne Farben nutzbar
  - Über Farben vermittelte Informationen sollen auch ohne Wahrnehmung der Farbe verfügbar sein, also zusätzlich durch andere Mittel (etwa Fettung oder Einrückung) hervorgehoben sein.
- Kontraste von Texten ausreichend
  - Alle Texte der Seite sollen in allen Zuständen ausreichende Helligkeitskontraste haben (https://www.siteimprove.com/de/toolkit/color-contrast-checker/).
- Verzicht auf Schriftgrafiken
  - Grafiken sollen nicht für die Darstellung von Schriften verwendet werden. Logos sind hiervon ausgenommen
- Kontraste von Grafiken und grafischen Bedienelementen ausreichend
  - Die für die Identifizierung notwendige visuelle Information von informationstragenden Grafiken und grafischen Bedienelementen sowie deren Zuständen soll einen Kontrast von mindestens 3:1 zu angrenzenden Farben haben. In vielen Fällen, wie etwa bei einfarbigen Icons, bedeutet das einfach, dass der Kontrast zwischen der Farbe des Elements und der Hintergrundfarbe gemessen wird (https://www.siteimprove.com/de/toolkit/color-contrast-checker/).
- Alternative Zugangswege
  - Benutzer bevorzugen verschiedene Zugangswege, um zu Inhalten zu gelangen. Manche orientieren sich an hierar-chischen Navigationsmenüs, andere nutzen ein Inhaltsverzeichnis (Sitemap), noch andere ziehen eine Suchfunkti-on vor. Gerade sehbehinderte Benutzer kommen oft schneller über eine Suche zu den gewünschten Inhalten. Des-halb sollte das Angebot verschiedene Zugangswege zu den Inhalten bereitstellen.
- Keine unerwartete Kontextänderung bei Fokus

    - Unerwartete und unangekündigte Kontextänderungen bei Fokussierung einer Komponente, kann die Orientierung von Nutzenden beeinträchtigen. Zum Beispiel: das automatische Abschicken von Formularen; das Öffnen neuer Fenster...
- Keine unerwartete Kontextänderung bei Eingabe
    - Unerwartete und unangekündigte Kontextänderungen bei einer Auswahl in Formularen können die Orientierung von Nutzern beeinträchtigen. Zum Beispiel die Auswahl einer Checkbox oder ein Radio-Button ruft eine neue Seite auf; eine Auswahlliste ohne Submit-Button reagiert auf onchange.
- Konsistente Navigation
    - Eine einheitliche Navigation innerhalb des Webauftritts erleichtert das Verständnis, Gesuchtes wird leichter gefunden
- Konsistente Bezeichnung
    - Klare und durchgängig verwendete Bezeichnungen für die Navigation und sich wiederholende Funktionen erleichtern Benutzern das Verständnis der Inhalte des Angebots.
- Stelle sicher, dass die Stati Error, Warnung und Erfolg nicht nur mit Farbe visuell dargestellt werden

Technik

- Leere alt-Attribute für Layoutgrafiken
  - Eine Grafik, die keine informative Funktion hat, benötigt keinen Alternativtext. Grafiken ohne informative Funktion sind zum Beispiel Abstandshalter, Farbflächen, Muster, oder rein dekorative Fotos. Solche Grafiken sollen mit ei-nem leeren alt-Attribut (alt="") ausgezeichnet werden.
  - Thema des Prüfschritts sind auch Icons, die mittels Icon Font eingebunden sind und SVGs.
- Alternativen für Audiodateien und stumme Videos (Möglichkeit)
  - Audiodateien und stumme Videodateien, die Informationen vermitteln, müssen mit gleichwertigen Medienalter-nativen versehen werden - es sei denn, es handelt sich bei Ihnen bereits um Medienalternativen für Text.
- HTML-Strukturelemente für Überschriften
  - Überschriften müssen korrekt mit den HTML-Strukturelementen h1 bis h6 oder äquivalenten ARIA-Rollen und Attri-buten ausgezeichnet sein und die Inhalte der Seite erschließen
- Nur eine H1 pro Seite
- Keine H Level überspringen
- HTML-Strukturelemente für Listen
  - Zur Auszeichnung von Listen auf der Seite sollen HTML-Strukturelemente für Listen (ul, ol und so weiter) genutzt werden.
- HTML-Strukturelemente für Zitate
  - Zur Auszeichnung von Zitaten, die als eigenständige Textabschnitte gefasst sind, soll das dafür vorgesehene HTML-Strukturelement blockquote genutzt werden.
- Datentabellen richtig aufgebaut
  - Datentabellen sind strukturell richtig aufgebaut, Zeilen- und Spaltenüberschriften sind mit th oder den entspre-chenden ARIA-Rollen ausgezeichnet.
- Zuordnung von Tabellenzellen
  - In komplexen Datentabellen soll der Bezug von Überschriften und Inhalten (über scope oder über id und headers) definiert sein, ausdrückliche Zuordnungen von Überschriften und Inhalten in einfachen Datentabellen sollen korrekt sein.
- Kein Strukturmarkup für Layouttabellen
  - Tabellenstruktur-Mark-up soll nicht für Layouttabellen verwendet werden.
- Verwende Landmark-Elemente, um wichtige Inhaltsbereiche zu kennzeichnen
- Nutze das button Element für Buttons
- Inhalt gegliedert
  - Absätze sind mit geeigneten Strukturelementen ausgezeichnet.
  - Hervorhebungen in Texten sind mit strong oder em ausgezeichnet.
  - Text-Inhalt wird nicht per CSS eingebunden.
- Beschriftung von Formularelementen programmatisch ermittelbar (wichtig!)
  - Sichtbare Beschriftungen von Formularelementen sind vorhanden.

    - Eine sichtbare Beschriftung von Formularelementen soll vor (das heißt links neben oder über) dem zugehörigen Eingabefeld vorhanden sein. Nur die Beschriftung von Checkboxes und Radiobuttons kann (und sollte normaler-weise) rechts neben dem zugehörigen Eingabefeld angeordnet werden.
    - Wenn für die Eingabe ein bestimmtes Format verlangt wird, so sind die Anweisungen für alle Benutzer lesbar.
- Sinnvolle Reihenfolge
    - Seiteninhalte sollen unabhängig von der Darstellung in einer sinnvollen und brauchbaren Reihenfolge stehen. Was inhaltlich zusammengehört (etwa eine Überschrift und die dazugehörigen Inhalte darunter) soll nicht auseinan-dergerissen werden.
    - Mittels CSS versteckte Seiteninhalte sollen deshalb an sinnvoller Stelle im Seitenquelltext erscheinen.
    - Dynamische Inhalte, die im Ausgangszustand visuell versteckt sind, sollen auch für Screenreader verborgen sein, damit sie nicht die Lesereihenfolge stören.
- Korrekte Syntax
    - Die verwendete Markup-Sprache HTML muss korrekt eingesetzt werden. Dabei muss für jedes Element folgendes gewährleistet sein:
        - Sie besitzen vollständige Start- und Endtags
        - sie sind gemäß Spezifikation korrekt verschachtelt
        - sie enthalten keine doppelten Attribute
        - alle ihre IDs sind eindeutig, außer dort wo die Spezifikationen etwas anderes erlauben
- Name-Rolle-Wert verfügbar
    - Alle selbst gestalteten Komponenten einer Website (also Elemente oder Widgets, die nicht auf interaktiven HTML-Elementen beruhen) sind so umgesetzt, dass die semantischen Informationen (Name, Rolle, Eigenschaften) vor-handen sind. Werden nicht semantische Elemente (etwa div oder span) eingesetzt und mithilfe von JavaScript zu Bedienelmenten umfunktioniert, wird die Semantik mithilfe von WAI-ARIA bereitgestellt.
    - Die wechselnden Zustände der Bedienelemente werden nicht nur visuell über CSS und JavaScript abgebildet, son-dern auch über scriptgesteuerte Änderung der Werte der ARIA-Attribute, damit die Semantik auch bei nicht-visueller Nutzung verfügbar ist.
- Statusmeldungen programmatisch verfügbar
    - Eine Statusmeldung ist eine Nachricht, die einer Seite dynamisch hinzugefügt wird. Sie informiert Nutzenden bei-spielsweise über den Erfolg oder das Ergebnis einer Aktion, über den Fortschritt eines Prozesses oder über das Vor-kommen von Fehlern.
    - Wenn Webangebote Statusmeldungen erzeugen, sollen visuell eingeblendete Statusmeldungen mit geeigneten Rollen und Eigenschaften ausgezeichnet und programmatisch ermittelbar sein, das bedeutet die Statusmeldun-gen werden Nutzenden von assistiven Technologien (Screenreader) präsentiert, ohne dass sie den Fokus erhal-ten.Beispiele für Statusmeldungen:
        - Ware wurde im Shop dem Warenkorb hinzugefügt
        - 3 Bücher der Merkliste hinzugefügt
        - Formular erfolgreich abgeschickt (Erfolgsmeldung)

    - 5 Suchergebnisse (etwa nach Filterung der Ergebnisse)
    - 3 Fehler im Formular (bei clientseitiger Prüfung ohne Neuladen der Seite)
    - Punktestand geändert
    - Seite wird geladen (bei visueller Ladeanzeige/Fortschrittsbalken)
- Keine Beschränkung der Bildschirmausrichtung
    - Webinhalte sollen sich an die nutzergewählte Ausrichtung von Ausgabegeräten anpassen. Sie sollten sowohl im Hochformat als auch im Querformat dargestellt werden und nutzbar sein, es sei denn, eine bestimmte Ausrich-tung des Inhalts ist unerlässlich.
    - Es wird nicht verlangt, dass in beiden Ausrichtungen Inhalte und Funktionen in der gleichen Form angeboten wer-den. So können in einer anderen Orientierung Inhalte ggf. erst nach Aktivierung von Ausklappbereichen oder mittels Links auf andere Seiten angeboten werden.
- Anderssprachige Wörter und Abschnitte ausgezeichnet (wichtig!)
    - Wenn innerhalb einer Seite Wörter und Textabschnitte in einer anderen Sprache vorkommen, müssen diese mithil-fe des lang-Attributs ausgezeichnet werden.
- Eingabefelder zu Nutzerdaten vermitteln den Zweck
    - Eingabefelder, die sich auf den Nutzer selbst beziehen, sollten eine semantisch eindeutige, sprachunabhängige Bestimmung ihres Zweckes ermöglichen. Geeignet dafür ist zur Zeit das HTML autocomplete-Attribut, mit dem sich der Eingabezweck für Felder wie etwa Name, E-Mail oder Telefonnummer ebenso wie für Adress-Daten oder Kre-ditkarten-Daten definieren lässt.
- Inhalte brechen um
    - Seiten-Inhalte sollen bei einer Browserfensterbreite von 320 CSS-Pixeln (bzw. bei einer Browserfensterbreite von 1280 CSS-Pixeln und 400% Zoomvergrößerung) so umbrechen, dass alle Informationen und Funktionen verfügbar sind, ohne dass Nutzer horizontal scrollen müssen.
    - Es wird dabei nicht verlangt, dass alle Inhalte in gleicher Weise in der responsiven Ansicht verfügbar sind. So sind die sichtbaren Navigationsmenüs einer Standard-Ansicht auf dem Desktop häufig nach dem Hereinzoomen (oder bei Nutzung des Angebots auf einem Smartphone-Bildschirm) nur über eine Ausklappnavigation zugänglich. Inhal-te können auch in Ausklappbereichen oder über Links auf andere Seiten oder Ansichten verfügbar sein.
- Textabstände anpassbar
    - Die Anpassung der Zeilenhöhe auf das 1,5-fache der Textgröße, des Abstands nach Absätzen auf das 2-fache der Textgröße, von Buchstabenabständen auf das 0,12-fache der Textgröße und von Wortabständen auf das 0,16-fache der Textgröße führt nicht zu einem Verlust an Inhalten oder Funktionalitäten, zum Beispiel durch das Ab-schneiden von Inhalten in Elementen, deren Größe sich bei Einstellung größerer Textabstände und dadurch erfol-gender Textspreizungen oder Umbrüche nicht dynamisch anpasst.
    - Anmerkung: Die Anforderung verlangt nicht von Autoren, die genannten Werte bei Ihren Inhalten umzusetzen, sondern lediglich, dass von Nutzern vorgenommene Anpassungen nicht zum Abschneiden von Text oder Verlust von Funktionalitäten führt.
- Eingeblendete Inhalte bedienbar (wichtig!)

    - Zusätzliche Inhalte, die angezeigt werden, wenn Elemente den Zeiger- oder Tastaturfokus erhalten, z. B. benutzer-definierte Tooltips oder Ausklapp-Menüs, sollten drei Anforderungen erfüllen:
        - Wenn zusätzliche Inhalte durch Darüberfahren mit dem Zeiger erscheinen, können Benutzer den Zeiger über diesen Inhalt bewegen, ohne dass er verschwindet.
        - Es gibt die Möglichkeit, einen eingeblendeten Inhalt zu schließen, ohne den Fokus zu verschieben (z. B. durch Drücken der Escape-Taste oder durch Aktivieren des Elements, dessen Fokussierung den Inhalt einblendet).
        - Der Inhalt schließt nicht selbsttätig nach einer gewissen Zeitspanne.
- Ohne Maus nutzbar (wichtig!)
    - Die Webseite soll auch ohne Maus - also ausschließlich mit der Tastatur - zu benutzen sein
- Zeitbegrenzungen anpassbar (Slider?)
    - Seiteninhalte werden ohne Zeitbegrenzung angezeigt, die Zeitbegrenzung ist abschaltbar, oder sie kann verlängert werden. Dies betrifft etwa:
        - zeitbegrenzte Dialoge, welche Nutzer zu Entscheidungen auffordern
        - Online-Transaktionen mit begrenzter Session-Dauer und automatischem Ausloggen bei längerer Inaktivität
        - das automatische Neu-Laden von Seiten oder die zeitverzögerte Weiterleitung zu einer anderen Seite
- Bewegte Inhalte abschaltbar (Slider?)
    - Ablenkung durch blinkende oder sich bewegende Elemente soll vermieden werden, auf 5 Sekunden begrenzt sein oder abschaltbar sein.
    - Bewegte oder automatisch aktualisierte Inhalte, z. B. periodisch wechselnde Nachrichten-Aufmacher (Teaser) sol-len zum Lesen anhaltbar sein.
- Bereiche überspringbar (wichtig!)
    - Verschiedene Inhaltsbereiche wie Navigation, Suche oder Seiteninhalt können von Nutzern assistiver Technologien übersprungen werden. Der Seitenaufbau soll unabhängig von der Darstellung deutlich werden. Eine der folgenden Voraussetzungen soll erfüllt sein:
        - Es werden sinnvolle Bereichsüberschriften (HTML-Strukturelemente h1 bis h6) eingesetzt
        - Es sind Sprunglinks vorhanden.
        - HTML5 Elemente zur Auszeichnung von Bereichen (header, nav, main, aside, footer) erschließen den Seiten-aufbau sinnvoll.
        - WAI-ARIA document landmarks strukturieren die Seitenbereiche sinnvoll.
        - Frames und Iframes brauchen ein sinnvolles title-Attribut.
- Schlüssige Reihenfolge bei der Tastaturbedienung (wichtig!)
    - Wenn die Webseite mit der Tastatur bedient wird, soll die Reihenfolge, in der Links, Formularelemente und Objekte angesteuert werden, schlüssig und nachvollziehbar sein.
- Aktuelle Position des Fokus deutlich (wichtig!)
    - Der Tastaturfokus soll mindestens genau so deutlich hervorgehoben werden wie der Mausfokus. Bei Links, die nicht auf den Mauszeiger reagieren, soll der Tastaturfokus sich zumindest deutlich von der ausgewählten Hinter-grundfarbe abheben.

Versteckte Sprunglinks sollen bei Fokuserhalt eingeblendet werden.

- Alternativen für komplexe Zeiger-Gesten
  - Wenn Webinhalte Funktionen implementiert werden, die über pfadbasierte Zeiger-Gesten (z.B. Wischgeste, etwa zum Bewegen von Slider-Bereichen oder Löschen von Inhalten) bedient werden können, gibt es Alternativen für die Aktivierung mittels einer einfachen Zeigereingabe.
- Zeigergesten-Eingaben können abgebrochen oder widerrufen werden
  - Funktionen von Bedienelementen sollen nicht bereits durch den Down-Event eines Zeigers auf einem Bedienele-ment ausgeführt werden; falls doch, muss es eine Möglichkeit geben, die ausgelöste Funktion entweder abzubre-chen oder rückgängig zu machen.
- Sichtbare Beschriftung Teil des zugänglichen Namens (wichtig!)
  - Sichtbare Beschriftungen von Bedienelementen sollen im zugänglichen Name des Bedienelements vorkommen. Dies betrifft zum Beispiel Links, Beschriftungen von Textfeldern, Buttons oder Checkboxen.
  - Spracheingabenutzer können Bedienelemente wie Links, Tasten oder Eingabefelder aktivieren, indem sie den sichtbaren Namen sagen, auch in der Verbindung mit Befehlen (z. B. Klick "Abschicken"). Wenn die sichtbare Be-schriftung nicht in dem hinterlegten zugänglichen Namen des Bedienelements (also dem Text, der programmatisch als Beschriftung ermittelt wird) vorkommt, lässt sich das Bedienelement gegebenenfalls nicht oder nur über Um-wege mittels Spracheingabe aktivieren.
- Hauptsprache angegeben
  - Die Hauptsprache der Webseite soll angegeben werden.
  - Screenreader verwenden Wortlisten, in denen die Aussprache der Wörter festgelegt ist. Sie müssen wissen, in wel-cher Sprache ein Text verfasst ist, damit sie die richtige Wortliste verwenden und den Text korrekt aussprechen können.
- Fehlererkennung (wichtig!)
  - Wenn ein Formular Fehlermeldungen erzeugt, sollen die fehlerhaft ausgefüllten Felder identifiziert und der Fehler in Textform beschrieben werden. Nicht dynamisch mit Java Script einblenden!
- Beschriftungen von Formularelementen vorhanden (wichtig!)
  - Eine sichtbare Beschriftung von Formularelementen soll vor (das heißt links neben oder über) dem zugehörigen Eingabefeld vorhanden sein. Nur die Beschriftung von Checkboxes und Radiobuttons kann (und sollte normaler-weise) rechts neben dem zugehörigen Eingabefeld angeordnet werden.
  - Wenn für die Eingabe ein bestimmtes Format verlangt wird, so sind die Anweisungen für alle Benutzer lesbar.
- Hilfe bei Fehlern (wichtig!)
  - Wenn ein Formular Fehlermeldungen erzeugt, müssen diese verständlich sein und Hinweise geben, wie der Fehler zu korrigieren ist.
- Fehlervermeidung wird unterstützt
  - Bei wichtigen Dateneingaben (etwa bei finanziellen Transaktionen) gibt es die Möglichkeit, die Dateneingabe rück-gängig zu machen oder sie vor dem Abschicken zu überprüfen und zu korrigieren. Erfolgreiche Eingaben werden bestätigt.
- Benutzerdefinierte Einstellungen

    - Die Seite soll benutzerdefinierte Browsereinstellungen berücksichtigen. Im Einzelnen können dies folgende Punkte sein:
        - Geänderte Vorder- und Hintergrundfarben
        - Schriftarten
        - Schriftgrößen
        - Darstellung des Fokuscursors
        - Maßeinheiten
    - Die Seite kann darüber hinaus eigene Werte für diese Einstellungen anbieten. Wenn diese Einstellungen nicht ge-nutzt werden, sollen die Browsereinstellungen übernommen werden. In manchen Fällen muss die Seite neu gela-den werden, damit sich die Änderungen auswirken.
- Vermeide die Verwendung des Autofokus-Attributs
- Entferne unsichtbare fokussierbare Elemente

Content Erstellung (Videos)

- Audiodeskription für Videos
  - Für informationstragende visuelle Videoinhalte muss eine Audiodeskription bereitgestellt werden. Die Handlung von Videos kann oft auch ohne Bild recht gut verfolgt werden. Den Sprecher einer Nachrichtensendung muss man zum Beispiel nicht sehen, um zu verstehen, worum es geht. Dagegen enthalten Spielfilme und Reportagen oft Passagen, in denen wenig gesprochen wird und Inhalte über das Bild vermittelt werden. Damit ein blinder Zu-schauer den Film verfolgen kann, müssen ihm solche Passagen beschrieben werden. Hierfür wird das Verfahren der Audiodeskription eingesetzt.
- Dokumentation von Kompatibilität und Barrierefreiheit
  - Wenn ein Webangebot Dokumentation bereitstellt (und dazu gehört auch die Erklärung zur Barrierefreiheit), dann soll diese Dokumentation vorhandene zusätzliche Barrierefreiheits-Eigenschaften und -Funktionen auflisten und deren Nutzung erklären, sofern diese als Teil des Web-Angebots selbst implementiert wurden. zum Beispiel Vor-lesefunktionen, Funktionen zum Einstellen individueller Farbschemata, oder verfügbare Tastaturkurzbefehle, In-formationen zur Kompatibilität mit assistiven Technologien.
- Audiodeskription oder Volltext-Alternative für Videos
  - Für informationstragende visuelle Videoinhalte muss eine Audiodeskription oder Volltext-Alternative bereitgestellt werden (Im Gegensatz zur Audiodeskription beschränkt sich die Beschreibung des Videoteils nicht nur auf die Pau-sen im vorhandenen Dialog. Es werden vollständige Beschreibungen aller visuellen Informationen bereitgestellt, einschließlich des visuellen Kontexts, der Handlungen und Ausdrücke der Schauspieler. Darüber hinaus werden nichtsprachliche Geräusche (Lachen, Stimmen aus dem Off usw.) beschrieben, und es werden Transkripte aller Dia-loge mitgeliefert.)
- Verwenden einfache Sprache und vermeide Redewendungen, Idiome und komplizierte Metaphern.

Befüllung

- Alternativtexte für Bedienelemente
  - Grafische Bedienelemente (alle verlinkten / interaktiven Grafiken und Bilder z.B. Icons, Logos, Teaserbilder) müssen mit Alternativtexten versehen werden.
  - Die Alternativtexte für Bedienelemente sollen das Ziel des Links bezeichnen.
  - Alternativtexte für grafische Schaltflächen (Buttons) sollen die Aktion bezeichnen, die der Button auslöst.
- Alternativtexte für Grafiken und Objekte
  - Nicht verlinkte informationsorientierte Grafiken und Bilder müssen mit Alternativtexten versehen werden.
  - Die Alternativtexte ersetzen das Bild, sie sollen also (wenn möglich) dieselbe Aufgabe erfüllen wie das Bild.
  - Bei eingebundenen Multimedia-Objekten, Video- beziehungsweise Audio-Dateien oder Applets soll der Alternativ-text zumindest eine beschreibende Identifizierung des Inhalts ermöglichen.
- Für, Bilder die Text enthalten, muss der Text in der Alt-Description angegeben werden
- Sinnvolle Reihenfolge
  - Seiteninhalte sollen unabhängig von der Darstellung in einer sinnvollen und brauchbaren Reihenfolge stehen. Was inhaltlich zusammengehört (etwa eine Überschrift und die dazugehörigen Inhalte darunter) soll nicht auseinan-dergerissen werden.
  - Mittels CSS versteckte Seiteninhalte sollen deshalb an sinnvoller Stelle im Seitenquelltext erscheinen.
  - Dynamische Inhalte, die im Ausgangszustand visuell versteckt sind, sollen auch für Screenreader verborgen sein, damit sie nicht die Lesereihenfolge stören.
- Sinnvolle Dokumenttitel
  - Dokumenttitel bezeichnen den Inhalt der individuellen Seiten und das Webangebot. Sie können für die Unterschei-dung und Auswahl von Seiten genutzt werden.
  - Dokumenttitel vertreten die Seiten, zum Beispiel in Listen mit Bookmarks. Sie sind wichtig für die Navigation und Orientierung in Webangeboten. Wenn das Angebot oder der Inhalt der Seite nicht bezeichnet sind, ist die Orientie-rung beeinträchtigt.
- Aussagekräftige Linktexte
  - Ziel oder Zweck des Links sollen aus dem Linktext hervorgehen oder aus dem direkten Kontext des Links ermittel-bar sein. Falls Links nicht auf HTML-Seiten verweisen, soll der Link über das Dateiformat des Zieldokuments infor-mieren.
- Aussagekräftige Überschriften und Beschriftungen
  - Überschriften beschreiben den folgenden Inhalt, Beschriftungen sind sinngebend

Externe Tools (z.B. Youtube, Vimeo)

- Wiedergabe von Untertiteln
  - Sind Videos mit synchroner Bild- und Tonspur vorhanden und werden dynamisch zuschaltbare, textbasierte Unterti-tel (closed captions) angeboten, bietet der Videoplayer die Möglichkeit, diese ein- und auszublenden.
- Synchrone Untertitel
  - Sind Videos mit synchroner Bild- und Tonspur vorhanden, ist gewährleistet, dass der Videoplayer Untertitel und Ton synchron darstellt.
- Untertitel-Anpassungen
  - Wenn das Webangebot Videos mit Untertiteln enthält, lassen sich Untertitel anpassen, z.B. hinsichtlich der Schrift-größe.
- Gesprochene Untertitel
  - Wenn das Webangebot Videos mit einem Orginalton enthält, der nicht der Hauptsprache des Webangebots ent-spricht, sollen zuschaltbare bzw. programmatisch ermittelbare Untertitel, die in der Hauptsprache der Seite angeboten werden, als akustische Ausgabe zugänglich sein, z.B. über eine eigene Tonspur, die den Untertiteln ent-spricht, oder über eine generierte Sprachausgabe der Untertitel.

Testing

- Text auf 200% vergrößerbar
  - Text soll um bis zu 200 % vergrößert werden können, ohne dass dabei Inhalt oder Funktionalität verloren geht.
- Keine Tastaturfalle
  - Kann der Tastaturfokus auf ein Element der Seite bewegt werden, muss er auch von diesem Element wieder wegbewegt werden können. Der Inhalt darf keine Tastaturfalle erzeugen.