


# Možnosti WEB Accessibility v moderních prohlížečích

Bc. Adam Strojil

---

Diplomová práce  
2023

 Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky

---

Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky  
Ústav informatiky a umělé inteligence

Akademický rok: 2022/2023

# ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Bc. Adam Strojil**  
Osobní číslo: **A20626**  
Studijní program: **N0613A140022 Informační technologie**  
Specializace: **Softwarové inženýrství**  
Forma studia: **Kombinovaná**  
Téma práce: **Možnosti WEB Accessibility v moderních prohlížečích**  
Téma práce anglicky: **WEB Accessibility options in modern browsers**

## Zásady pro vypracování

1. Definujte cílovou skupinu, pro kterou je práce určena, popište nejčastější druhy postižení uživatelů webu.
2. Představte WAI (Web Accessibility Initiative) a WCAG (Web Content Accessibility Guidelines), porovnejte jednotlivé hlavní verze, úrovně shody (conformance levels), popište jednotlivá kritéria přístupnosti.
3. Představte pravidla přístupnosti v České republice a porovnejte je se zahraničím.
4. Uvedte atributy přístupnosti wai-aria v HTML5, připojte ukázky implementace.
5. Vyzkoušejte a popište uživatelské nástroje přístupnosti a nástroje používané při vývoji přístupného software a testování přístupnosti software.
6. Na praktických příkladech demonstруйте rozdíly mezi nepřístupnými webovými prvky a předvedte, jak je lze zpřístupnit.

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. WAI-ARIA Overview [online]. 2020. Dostupné z: [<https://www.w3.org/WAI/standards-guidelines/aria/>](<https://www.w3.org/WAI/standards-guidelines/aria/>)
2. Web Content Accessibility Guidelines 1.0 [online]. W3C 1999. Dostupné z: [<https://www.w3.org/TR/WAI-WEBCONTENT/>](<https://www.w3.org/TR/WAI-WEBCONTENT/>)
3. Web Content Accessibility Guidelines (WCAG) 2.0 [online]. W3C 2008. Dostupné z: [<https://www.w3.org/TR/WCAG20/>](<https://www.w3.org/TR/WCAG20/>)
4. Přístupnost.cz Otevřete svůj web všem! [online]. Dostupné z: [<http://www.pristupnost.cz/>](<http://www.pristupnost.cz/>)
5. Přístupnost internetových stránek a mobilních aplikací [online]. Ministerstvo vnitra České republiky 2019. Dostupné z: [<https://www.mvcr.cz/clanek/pristupnost-internetovych-stranek-a-mobilnich-aplikaci.aspx>](<https://www.mvcr.cz/clanek/pristupnost-internetovych-stranek-a-mobilnich-aplikaci.aspx>)
6. Digital Education Strategies, The Chang School. GAY, Greg & Karasyov, Igor (November 29, 2019) – Web Accessibility for Developers
7. Harper, Simon & Yesilada, Yeliz. (2008). Web Accessibility and Guidelines. 10.1007/978-1-84800-050-6\_6.

Vedoucí diplomové práce:

**Ing. David Malaník, Ph.D.**

Ústav informatiky a umělé inteligence

Datum zadání diplomové práce: **2. prosince 2022**

Termín odevzdání diplomové práce: **26. května 2023**



**doc. Ing. Jiří Vojtěšek, Ph.D. v.r.**  
děkan

**prof. Mgr. Roman Jašek, Ph.D., DBA v.r.**  
ředitel ústavu

Ve Zlíně dne 7. prosince 2022

## **Prohlašuji, že**

- beru na vědomí, že odevzdáním diplomové práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomové práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

## **Prohlašuji,**

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

Bc. Adam Strojil, v. r.

## **ABSTRAKT**

Tato diplomová práce se zabývá tématem přístupnosti webových stránek pro uživatele s různými druhy postižení. Práce definuje cílovou skupinu uživatelů, popisuje nejčastější druhy postižení a představuje WAI a WCAG. V práci jsou porovnány jednotlivé verze WCAG, úrovně shody, zásady, pokyny a popsána kritéria přístupnosti. Dále práce porovnává pravidla přístupnosti z hlediska legislativy v Česku se zahraničím. Uvádí atributy přístupnosti wai-aria a sémantické značky v HTML5 s ukázkami jejich použití. V praktické části se práce zaměřuje na uživatelské nástroje a nástroje používané při vývoji a testování přístupnosti. Na konkrétních příkladech a zkušebním auditu webové stránky demonstruje rozdíly mezi nepřístupnými a přístupnými webovými prvky a ukazuje, jak lze nepřístupné prvky zpřístupnit. Cílem práce je poskytnout ucelený přehled o webové přístupnosti a přispět k větší informovanosti a povědomí o této problematice v oblasti webového vývoje.

Klíčová slova: web, přístupnost, html, a11y, webdesign, WCAG, ARIA, postižení, frontend, UI, UX

## **ABSTRACT**

This diploma thesis deals with the topics of web accessibility for users with various types of disabilities. The thesis defines the target group of users, describes the most common types of disabilities and explains WAI and WCAG. The thesis describes versions of WCAG, conformance levels, principles, guidelines and success criteria. Furthermore, the thesis compares accessibility rules in Czechia and abroad. It lists the wai-aria accessibility attributes and semantic tags in HTML5 with examples of their use. In the second part, the thesis focuses on tools used in the development and testing of accessibility. Utilizing practical examples and a test audit, it demonstrates the differences between inaccessible and accessible web elements and shows how elements can be improved. The goal of the thesis is to provide a comprehensive overview of web accessibility and to contribute to greater awareness of this issue in the field of web development.

Keywords: web, accessibility, html, a11y, webdesign, WCAG, ARIA, disability, frontend, UI, UX

V první řadě bych chtěl vyjádřit své díky společnostem CN Group CZ, Interflex a také jmenovitě Joschimu Kuphalovi, s jejichž spoluprací jsem získal většinu svých dosavadních znalostí v oblasti webové přístupnosti. Dále chci poděkovat vedoucímu této práce panu Ing. Davidovi Malaníkovi, Ph.D. za cenné rady a připomínky v průběhu zpracování práce.

Zároveň děkuji i své rodině za její podporu během psaní této práce a celého mého studia, jak ve Zlíně, tak i v zahraničí.

## OBSAH

ÚVOD .....	10
<b>I TEORETICKÁ ČÁST .....</b>	<b>10</b>
<b>1 NEJČASTĚJŠÍ DRUHY POSTIŽENÍ UŽIVATELŮ .....</b>	<b>12</b>
1.1 NEVIDOMOST, BARVOSLEPOST .....	12
1.1.1 Barvoslepost a webová přístupnost .....	12
1.2 SLUCHOVĚ POSTIŽENÍ UŽIVATELÉ.....	15
1.3 OMEZENÍ POHYBU .....	16
1.4 KOGNITIVNÍ OMEZENÍ.....	16
1.5 DOČASNÉ NEBO SITUAČNÍ POSTIŽENÍ.....	16
1.6 UŽIVATELÉ S ALTERNATIVNÍM SOFTWAREM A HARDWAREM .....	17
<b>2 VÝHODY PŘÍSTUPNÉHO WEBU.....</b>	<b>18</b>
<b>3 WAI A WCAG .....</b>	<b>19</b>
3.1 WEB ACCESSIBILITY INITIATIVE (WAI).....	19
3.2 WEB CONTENT ACCESSIBILITY GUIDELINES (WCAG) .....	19
3.2.1 WCAG 1.0.....	21
3.2.2 WCAG 2.0.....	21
3.2.3 WCAG 2.1.....	21
3.2.4 WCAG 2.2.....	21
3.3 LAYERS OF GUIDANCE.....	22
3.3.1 Zásady .....	22
3.3.2 Pokyny.....	22
3.3.3 Kritéria přístupnosti .....	22
3.3.4 Dostatečné a doporučené techniky .....	23
3.4 ÚROVNĚ SHODY.....	23
3.4.1 Úroveň A.....	23
3.4.2 Úroveň AA .....	26
3.4.3 Úroveň AAA.....	28
<b>4 WEBOVÁ PŘÍSTUPNOST V ČESKÉ REPUBLICCE.....</b>	<b>31</b>
4.1 LEGISLATIVA .....	31
4.1.1 Prohlášení o přístupnosti.....	31
4.2 POROVNÁNÍ SE ZAHRANIČÍM .....	32
<b>5 SÉMANTICKÉ ZNAČKY V HTML5 .....</b>	<b>33</b>
<b>6 WAI-ARIA V HTML5.....</b>	<b>35</b>

6.1	1. PRAVIDLO ARIA .....	35
6.2	ČLENĚNÍ ATRIBUTŮ .....	36
6.2.1	Atributy widgetů .....	36
6.2.2	Atributy živé oblasti (live-region) .....	37
6.2.3	Atributy Drag-and-drop .....	38
6.2.4	Atributy vztahu .....	38
<b>7</b>	<b>FOCUS MANAGEMENT</b> .....	<b>39</b>
7.1	TABINDEX .....	39
7.2	TOK FOKUSU (FOCUS FLOW) .....	40
7.3	FOKUS PAST (FOCUS TRAP) .....	40
<b>II</b>	<b>PROJEKTOVÁ ČÁST</b> .....	<b>41</b>
<b>8</b>	<b>NÁSTROJE PRO VÝVOJ A TESTOVÁNÍ PŘÍSTUPNOSTI</b> .....	<b>42</b>
8.1	ČTEČKY OBRAZU (SCREEN READERS) .....	42
8.2	LIGHTHOUSE .....	44
8.3	CHROME DEVTOOLS .....	45
8.4	PRÁCE S BARVAMI .....	47
8.4.1	Měření kontrastu .....	47
8.4.2	Simulace barvosleposti .....	49
8.5	HTML VALIDÁTORY .....	50
8.6	BOOKMARKLETY .....	51
8.7	ZKUŠEBNÍ AUDIT .....	52
8.7.1	HTML Validátor .....	52
8.7.2	Lighthouse a DevTools .....	53
8.7.3	Manuální testování .....	57
<b>9</b>	<b>PROCES ZPŘÍSTUPNĚNÍ WEBOVÝCH KOMPONENT</b> .....	<b>60</b>
9.1	UKÁZKA 1 .....	60
9.1.1	Počáteční kód .....	61
9.1.2	Identifikace problémů .....	61
9.1.3	Zpřístupnění komponenty .....	63
9.1.4	Refaktorovaný kód .....	64
9.2	UKÁZKA 2 .....	66
9.2.1	Počáteční kód .....	67
9.2.2	Identifikace problémů .....	67
9.2.3	Zpřístupnění komponenty .....	69
9.2.4	Refaktorovaný kód .....	71



ZÁVĚR .....	72
SEZNAM POUŽITÉ LITERATURY .....	73
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK .....	75
SEZNAM OBRÁZKŮ .....	76
SEZNAM PŘÍLOH .....	78

## ÚVOD

Přístupnost webu je zásadní téma, které v posledních letech získává velkou pozornost. S rostoucím počtem uživatelů se zdravotním postižením používajících web (ať už z důvodu prostého růstu populace, či nedávných krizových událostí, jakými jsou například ozbrojený konflikt na Ukrajině, nebo zemětřesení v Turecku) se stává zásadní zajistit, aby byly webové stránky přístupné všem. Tato diplomová práce se zaměřuje právě na přístupnost webu pro uživatele s různými typy postižení, ať už zdravotními, dočasnými, či situačními.

Cílovou skupinou této diplomové práce jsou především weboví vývojáři, designéři UI/UX, a softwaroví testéři, ale věřím, že může být zajímavá i pro běžné uživatele webu. Smyslem práce je poskytnout lepší pochopení přístupnosti webu a toho, jak vytvářet přístupné webové aplikace pro uživatele se zdravotním postižením. Tito uživatelé čelí různým výzvám při jejich používání, a je odpovědností webových vývojářů zajistit, aby byly webové stránky přístupné všem, bez ohledu na jejich schopnosti. V rámci webů některých subjektů, kterými jsou například stát či vysoké školy, je navíc webová přístupnost vyžadována zákonem.

V teoretické části poskytuje práce ucelený přehled o tématu, od popisu nejběžnějších typů postižení, představení WAI, WCAG, ARIA a sémantických značek HTML5, až po představení legislativy upravující webovou přístupnost v České republice a zahraničí.

V praktické části práce jsou prezentovány uživatelské nástroje, které lze využít při vývoji a testování přístupných webových aplikací. Na názorných příkladech a zkušebním auditu domovské stránky utb.cz práce demonstruje rozdíly mezi nepřístupnými a přístupnými prvky webu a ukazuje, jakými technikami a postupy lze špatně přístupné prvky odhalit, a následně vylepšit tak, aby odpovídaly pravidlům přístupnosti.

# I. TEORETICKÁ ČÁST

## 1 Nejčastější druhy postižení uživatelů

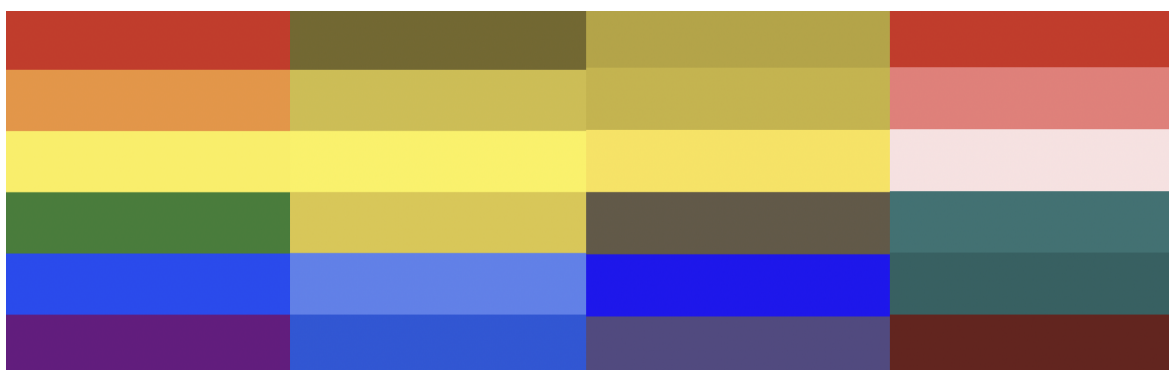
### 1.1 Nevidomost, barvoslepost

Podle Světové zdravotnické organizace (WHO) má přibližně 253 milionů lidí na celém světě středně těžké až těžké poškození zraku a 36 milionů lidí je nevidomých. To jsou pro představu přibližně 3 % světové populace.[1]

I nevidomí lidé však mohou poměrně snadno používat web (za pomoci klávesnice, čtečky obrazu, či další AT (asistenční technologie), dodrží-li jeho vývojáři alespoň základní zásady přístupnosti.

Dále se předpokládá, že 1 z 12 mužů (8 %) a 1 z 200 žen je barvoslepých (přibližně 4,5 % světové populace). V důsledku čehož je na světě více než 350 milionů barvoslepých lidí. Ve většině případů se jedná o poruchu vnímání zelené a červené barvy, ostatní formy jsou vzácnější.[2]

- **Protanopie** - postižený nevnímá červenou barvu. Částečná porucha se nazývá protanomalie
- **Deuteranopie** - postižený nevnímá zelenou barvu. Částečná porucha je označována jako deuteranomalie.
- **Tritanopie** - postižený nevnímá modrou barvu. Částečná porucha nese označení tritanomalie.

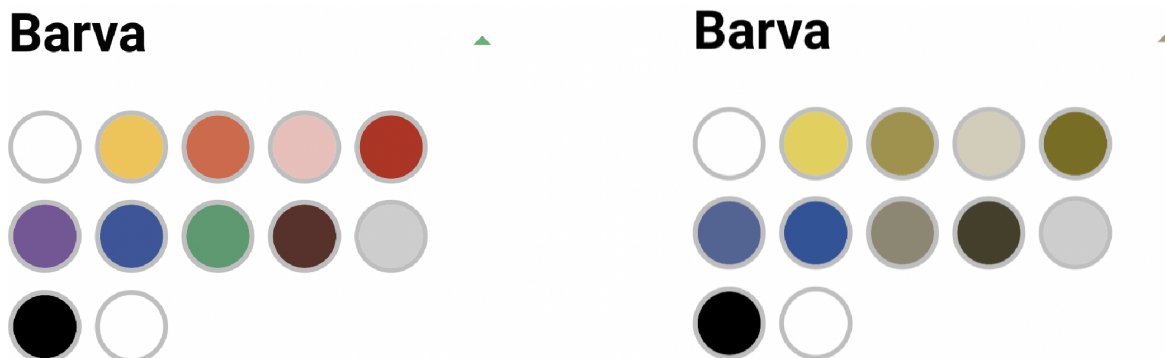


Obr. 1.1 Normální barvocit, protanopie, deuteranopie a tritanopie

#### 1.1.1 Barvoslepost a webová přístupnost

V rámci WACG je s barvoslepostí nejvíce spojeno kritérium úspěchu 1.4.1: Use of Color (Použití barvy), které říká, že *"Barva se nepoužívá jako jediný vizuální prostředek pro přenos informací, označení akce, vyvolání reakce nebo rozlišení vizuálního prvku."*[3]

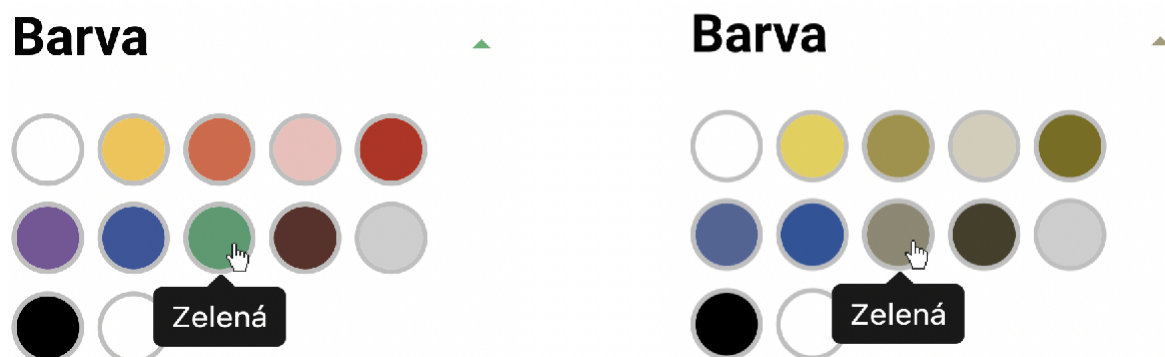
To v praxi znamená, že u webové komponenty, která slouží například v e-shopu k výběru barvy trika, nelze použít pouze onu barvu samotnou. Jako příklad nepřístupného řešení lze uvést tuto jednoduchou komponentu pro výběr barvy (Obr. 1.2, v levo):



Obr. 1.2 Výběr barvy - normální barvocit vs. deuteranopie

Pokud by návštěvník e-shopu trpěl například typem barvosleposti deuteranopie je z pravé části Obr. 1.2 zřejmé, že by pro něj bylo obtížné správně zvolit požadovanou barvu. Pro člověka, který je nevidomý by to bylo nemožné.

Aby byla tato komponenta přístupná, musí být informace o barvě sdělena také jiným způsobem, nejlépe textem. Ten by mohl být přidán buď přímo vedle barvy (Obr. 1.4), nebo s použitím vyskakovacího popisku (tooltip), zobrazovaného po najetí ukazatele myši (Obr. 1.3). V tomto případě je ale nutno myslet i na uživatele, kteří myš používat nemohou, a buď poskytnout další mechanismus zobrazení tohoto popisku (například při fokusu dané barvy). Rovněž je vhodné ještě popisek a barvu provázat pomocí atributů `aria-label` nebo `aria-labelledby`.



Obr. 1.3 Přístupná komponenta - normální barvocit vs. deuteranopie, verze 1

Druhá verze na (Obr. 1.3) je z hlediska přístupnosti vhodnější z důvodu okamžitě viditelného textového popisku (není třeba použití ukazatele, ani klávesnice pro jeho zobrazení) a jeho automatické dostupnosti pro AT, bez dodatečného použití aria atributů. To v praxi znamená, že například čtečka obrazovky pro vybranou možnost (barvu) uživateli přečte její text, a komponentu lze tedy použít i při úplné nevidomosti. Na druhou stranu ale tato verze s sebou nese vizuální designové změny, které nemusí být praktické z jiného úhlu pohledu.



Obr. 1.4 Přístupná komponenta - normální barvocit vs. deuteranopie, verze 2

Dalším velmi častým případem, kdy dochází k porušení SC 1.4.1, jsou hypertextové odkazy, označeny pouze barvou. I zde je vhodné poskytnout uživateli znatelnější rozlišení od obyčejného textu, než pouze jinou barvu. Demonstrováno na (Obr. 1.5, 1.6, 1.7 a 1.8), kdy je k označení odkazu použito v kombinaci s barvou také podtržení textu.

## Trička

Naše trička jsou kvalitní a navíc Vám při splnění **podmínek** garantujeme dopravu zdarma!

Obr. 1.5 Původní odkaz v textu, normální

## Trička

Naše trička jsou kvalitní a navíc Vám při splnění **podmínek** garantujeme dopravu zdarma!

Obr. 1.6 Původní odkaz v textu, deuteranopie

# Trička

Naše trička jsou kvalitní a navíc Vám při splnění [podmínek](#) garantujeme dopravu zdarma!

Obr. 1.7 Přístupný odkaz v textu, normální

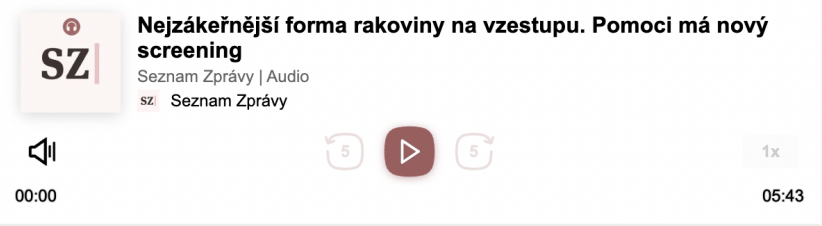
# Trička

Naše trička jsou kvalitní a navíc Vám při splnění [podmínek](#) garantujeme dopravu zdarma!

Obr. 1.8 Přístupný odkaz v textu, deuteranopie

## 1.2 Sluchově postižení uživatelé

Uživatelé, kteří mají úplně nebo částečně omezenou schopnost slyšet, většinou nemají při práci s webem velké potíže. Při navrhování webu je ale třeba vzít v úvahu i tato omezení. To platí zejména v případě, kdy web obsahuje zvukový obsah. V takovém případě je třeba uživatelům poskytnout možnost obsah konzumovat také v alternativní podobě. To může zahrnovat použití titulků, či textových popisů pro video či audio záznamy. Takový popis je rovněž vhodné psát strukturovaně, jednoduchým jazykem, a bez použití cizích slov.[4]



**Nejzákeřnější forma rakoviny na vzestupu. Pomoci má nový screening**  
Seznam Zprávy | Audio  
sz. Seznam Zprávy

00:00 05:43

Článek si také můžete poslechnout v audioverzi.

Karcinom slinivky břišní patří podle dat mezi nejzákeřnější formy onkologického onemocnění. Dlouho probíhá prakticky bez příznaků a lékaři ho tak obvykle objeví až ve chvíli, kdy už je na úspěšnou léčbu pozdě. Až 95 procent pacientů umírá do pěti let od diagnózy.

Obr. 1.9 Textový přepis audio rozhovoru na webu Seznam Zprávy

### 1.3 Omezení pohybu

Omezení pohybu se vztahuje nejčastěji na uživatele, kteří mají vinou fyzického postižení stíženou (nebo nemožnou) schopnost práce s webem, který vyžaduje použití myši, či jiného ukazovacího zařízení. Příčinou těchto postižení mohou být diagnózy jako je třes, omezené ovládání jemné motoriky nebo omezený dosah. Do této kategorie také spadají lidé s absencí horních končetin důsledkem amputace, či vrozené vady.

### 1.4 Kognitivní omezení

Označení kognitivní omezení zastřešuje vícero typů poruch. Patří sem převážně potíže s porozuměním, zpracováním a uchováním informací. Některé příklady kognitivních poruch, se kterými se uživatelé webu mohou potýkat, zahrnují:

- Poruchy paměti
- Porucha pozornosti s hyperaktivitou (ADHD)
- Dyslexie nebo jiné potíže se čtením
- Porucha autistického spektra (ASD)
- Traumatické poranění mozku (TBI)
- Demence nebo Alzheimerova choroba
- Poruchy učení, jako je dyskalkulie (potíže s matematikou)

### 1.5 Dočasné nebo situační postižení

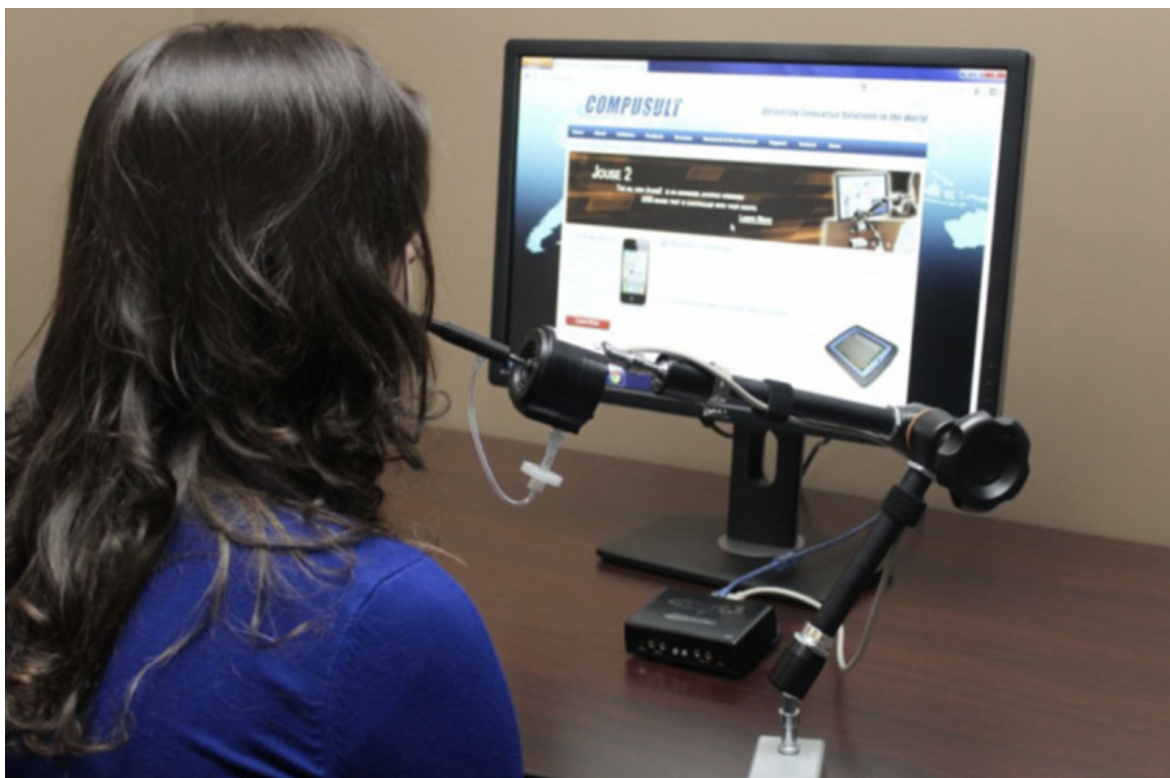
Do této kategorie jsou zařazeny dočasné komplikace, se kterými se mohou relativně snadno potkat i jinak zdraví jedinci. Příkladem dočasného postižení by mohla být zlomená ruka, zápěstí či prst, což dokáže znatelně omezit schopnost uživatele pracovat s myší. Dalším příkladem je dočasné omezení vidění či sluchu v důsledku nemoci nebo nehody.

Druhou podkategorií jsou situační postižení. Ze všech postižení jsou nejméně závažné, ale na druhou stranu pravděpodobně nejčastější. Jedná se o omezení vyplývající z aktuálního (nepříznivého) prostředí či situace, ve které se uživatel nachází. Typickými zástupci této kategorie jsou používání telefonu nebo počítače v rušném prostředí, což téměř znemožňuje uživateli konzumovat zvukový obsah webu, nebo jejich používání na přímém slunci. V takové situaci téměř nelze rozeznat obsah, který nemá dostatečný kontrast, nebo dostatečnou velikost (u písma).



## 1.6 Uživatelé s alternativním softwarem a hardwarem

Existuje celá škála zařízení, které mohou uživatelé pro přístup k webu využít, a každé z nich může mít jiné rozhraní. Je třeba to mít na paměti a zajistit, že důležité informace budou pro uživatele dostupné bez ohledu na to, jaké zařízení pro zobrazení webu používají. Může sem patřit responzivní design webů, který se stará o to aby byl zobrazen korektně jak na velké obrazovce počítače, tak na malém displeji telefonu. Do této kategorie také spadají uživatelé využívající alternativní, specifická zobrazovací zařízení nebo asistenční technologie, např. čtečky obrazovky.[4]



Obr. 1.10 Jouse 3 Joystick ovládaný ústy od Compusult [5]

## 2 Výhody přístupného webu

Weby které splňují požadavky přístupnosti nejsou benefiční pouze pro uživatele s postiženími. Nabízí se zde analogie z reálného světa ohledně designu chodníků. Aby je mohli bez problémů využívat i lidé na vozíčku, byla přidána místa, která nemají patník, ale nájezdovou rampu. Vlastnost, která je určena primárně pro vozíčkáře je ale benefiční pro celou řadu dalších uživatelů, jako jsou cyklisté nebo lidé s kočárky. Stejně tak tomu je i v případě webové přístupnosti. Jako příklad lze zmínit situaci, kdy je pro obrázek přidán alternativní textový popis, sloužící primárně pro předání informace v něm obsažené nevidomým lidem. Současně mohou ale tento popis využít také internetové vyhledávače k indexaci obsahu, kdy může poté popis sloužit k vyhledání obrázku. Obdobně je takový popis přínosný pro uživatele s pomalým připojením k internetu, kteří se mohou rozhodnout obrázky záměrně nezobrazovat. Důvodů proč usilovat o dosažení webové přístupnosti je několik:[6]

- **Vylepšená optimalizace pro vyhledávače (SEO)** - Zákazníci budou moci snáze web objevit, protože jej vyhledávače mohou lépe indexovat.
- **Lepší použitelnost** - Zákazníci budou při používání webu spokojenější, a tím pádem mohou v případě e-shopů více utrácet a vracet se.
- **Snížení nákladů na webové stránky** - Vývoj proti standardu snižuje počet chyb a problémy interoperability, snížení nákladů na vývoj a problémy s integrací s jinými systémy.
- **Lidé s postižením mají kupní sílu** - Pokud daný e-shop není přístupný, tato skupina je nucena přejít ke konkurenci, která bere webovou přístupnost v potaz.
- **Podpora uživatelů s pomalým připojením** - Pokud se web načítá příliš dlouho, uživatelé půjdou jinam.
- **Společenská odpovědnost** - Zákazníci ocení, když uvidí, že se web snaží zlepšit svět a myslí na lidi s postižením jako na plnoprávné občany.
- **Podpora stárnoucí populace** - Starší lidé také ocení webovou přístupnost a budou pro nákupy preferovat weby, se kterými se jim snadno pracuje.
- **Snížené právní riziko** - Pokud je znemožněn rovný přístup k informacím a obsahu, mohou se objevit žaloby za diskriminaci.

Téměř každá změna zavedená do webového obsahu, byť určena primárně lidem se zdravotním postižením, lepší přístup a použitelnost také všem ostatním.[6]

### 3 WAI a WCAG

#### 3.1 Web Accessibility Initiative (WAI)

Web Accessibility Initiative (WAI) je iniciativou W3C (World Wide Web Consortium), jež si vzalo za závazek vést web k jeho plnému potenciálu, což zahrnuje mimo jiné i podporu vysokého stupně použitelnosti pro lidi s postižením.[7]

WAI rozvíjí svou práci prostřednictvím procesu založeného na konsensu W3C, který zahrnuje různé zainteresované strany v oblasti přístupnosti webu. Těmito stranami jsou průmysl, organizace pro osoby se zdravotním postižením, vláda, organizace zabývající se výzkumem přístupnosti a další.[7]

WAI ve spolupráci s organizacemi po celém světě usiluje o dostupnost webu prostřednictvím těchto primárních aktivit:

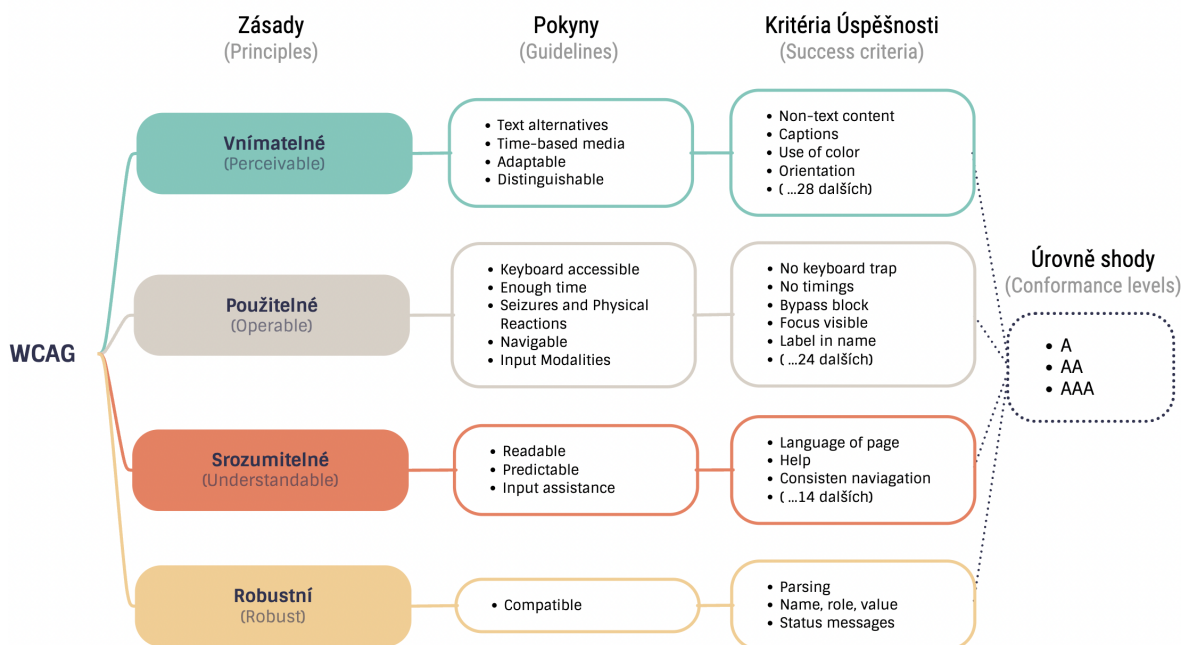
- zajištění toho, že standardy W3C podporují přístupnost
- vývoj pokynů pro přístupnost pro webový obsah a aplikace, prohlížeče a nástroje pro tvorbu obsahu
- vývoj zdrojů pro zlepšení procesů a nástrojů hodnocení dostupnosti webu
- podpora vzdělávání a informování o dostupnosti webu
- koordinace s výzkumem a vývojem, které mohou ovlivnit budoucí dostupnost webu
- podpora harmonizovaného mezinárodního přijímání norem pro přístupnost webu

WAI je členěna do pěti pracovních skupin z nichž za zmínku určitě stojí Accessibility Guidelines Working Group (AG WG), která má na starosti vývoj specifikací pro podporu implementace přístupných webových technologií, a také skupina Accessible Rich Internet Applications (ARIA), která se stará o vývoj technologií, které zlepšují přístupnost webového obsahu.[7]

#### 3.2 Web Content Accessibility Guidelines (WCAG)

WCAG lze přeložit jako "Pokyny pro přístupnost webového obsahu". Ty jsou souborem mezinárodně uznávaných pokynů a standardů pro zpřístupnění webového obsahu lidem s postižením. Pokyny poskytují komplexní soubor kritérií pro hodnocení přístupnosti webového obsahu a jsou navrženy tak, aby byly technologicky neutrální, tj. vztahují se na všechny typy webového obsahu bez ohledu na to, jak byl vytvořen. V době psaní této práce je aktuální verze 2.1 publikovaná v roce 2018, a většina obsahu této práce se tak vztahuje právě k této verzi.

Standardy WCAG mají 13 pokynů, které jsou uspořádány podle 4 principů: vnímatelné, použitelné (funkční), srozumitelné a stabilní. Pro každý pokyn existují testovatelná kritéria úspěšnosti. Kritéria úspěšnosti jsou na třech úrovních: A, AA a AAA. Kritéria úspěšnosti určují „soulad“ s WCAG. To znamená, že obsah lze požadovat za přístupný podle WCAG (na dané úrovni), v momentě, kdy jsou splněna kritéria úspěšnosti (dané úrovně).[9]



Obr. 3.1 Struktura WCAG

WCAG jsou primárně určeny pro:[9]

- Vývojáře webového obsahu (autoři, návrháři stránek, ...)
- Vývojáře nástrojů pro tvorbu webu
- Vývojáře nástrojů pro hodnocení přístupnosti webu
- Ostatní, kteří chtějí nebo potřebují standard pro přístupnost webu, včetně mobilní přístupnosti

Jen málo návrhářů a autorů ví, nebo rozumí tomu, jak postižení lidé přistupují na web, a co je třeba udělat pro vytvoření přístupné stránky. Některé studie navíc ukazují, že použití pokynů podléhá výkladu - dva návrháři aplikující stejnou sadu pokynů na stejnou sadu stránek mohou generovat různé výsledky.[10]

### 3.2.1 WCAG 1.0

I přes to, že se nejedná o první zmínku pravidel přístupnosti v online prostředí (před vydáním v roce 1999 již existovala například americká "*Section 508 accessibility standards*"), WCAG 1.0 byl prvním souborem mezinárodně uznávaných směrnic, které se konkrétně zaměřovaly na přístupnost webového obsahu. Jeho cílem bylo poskytnout společný soubor principů a pokynů, které by bylo možné použít k vyhodnocení a zlepšení dostupnosti webového obsahu pro osoby se zdravotním postižením.[11][12]

### 3.2.2 WCAG 2.0

Verze 2.0 byla vydána 11. prosince 2008, tj. 9 let po verzi předchozí. V této verzi bylo představeno 12 pokynů, které se dále člení na testovatelná kritéria úspěšnosti rozdělená do tří úrovní (A, AA, AAA). V této verzi bylo také poprvé představeno členění pokynů do čtyř kategorií, a sice Perceivable (vnímatelné), Operable (funkční, použitelné), Understandable (srozumitelné), a Robust (stabilní, robustní). WCAG 2.0 rovněž klade důraz na to být technologicky neutrální (pokyny nejsou vázány na konkrétní technologii použitou k tvorbě či prezentaci webu), oproti verzi 1.0, která se více soustředila na obsah založen na HTML. V této verzi byla také vzána v potaz internacionalizace, zatímco předchozí verze se soustředila primárně na uživatele používající angličtinu.[13]

### 3.2.3 WCAG 2.1

WCAG 2.1 bylo vydáno 5. června 2018 a obsahuje 17 nových kritérií úspěšnosti určených pro řešení nových technologií a kontextů použití, které se objevily od vydání WCAG 2.0. Těchto 17 nových kritérií je rozloženo do čtyř nových pokynů (1.3.4, 1.3.5, 2.5.3, 2.5.4) a dvou stávajících pokynů (1.4, 2.2), které byly rozšířeny. Z předchozí verze WCAG 2.0 žádné stávající pokyny odstraněny nebyly.[14]

### 3.2.4 WCAG 2.2

V době psaní této práce se verze 2.2 stále nacházela ve stádiu návrhu. Je zpětně kompatibilní s předešlými verzemi (web splňující WCAG 2.2 splňuje i verzi 2.1). Oproti verzi 2.1 obsahuje 9 nových kritérií úspěšnosti a jedno (4.1.1 Parsing) odebírá. Někteřími z nových požadavků jsou minimální velikost cíle (target) pro ukazatele, viditelnost ukazatele zvýraznění (focus), přístupná autentizace, či automatické předvyplnění polí již jednou zadanými informacemi. Obsahuje také několik změn, z nichž nejvýraznější je povýšení kritéria "2.4.7 Focus Visible" z úrovně AA na úroveň A. Vydání této verze se předpokládá na rok 2023.[15]

### 3.3 Layers of Guidance

WCAG používají jednotlivci i organizace, designéři, vývojáři, nákupčí, učitelé, studenti a další. Aby bylo možné uspokojit různé potřeby těchto subjektů, WCAG poskytuje několik úrovní pokynů včetně obecných zásad, obecných pokynů, testovatelných kritérií úspěchu, bohaté sbírky poradenských technik, zdokumentovaných běžných selhání s příklady a odkazy na zdroje s kódem.[16]

#### 3.3.1 Zásady

Všechny pokyny a kritéria úspěšnosti jsou založeny na čtyřech základních principech přístupnosti. V případě, že některý z těchto principů není respektován, uživatelé s postižením nebudou schopni daný web používat.

- **Vnímatelné (Perceivable)** - Informace a komponenty uživatelského rozhraní musí být uživatelům prezentovatelné způsobem, který mohou vnímat. To znamená, že uživatelé musí být schopni obsah vnímat alespoň některým z jejich smyslů.
- **Ovladatelné (Operable)** - Komponenty uživatelského rozhraní a navigace musí být funkční. To znamená, že po uživateli nemůže být vyžadována akce, kterou není schopen provést.
- **Srozumitelné (Understandable)** - Informace a ovládání uživatelského rozhraní musí být srozumitelné, a uživatelé jej musí být schopni pochopit.
- **Stabilní (Robust)** - Obsah musí být dostatečně robustní, aby mohl být spolehlivě interpretován širokou škálou uživatelských agentů, včetně asistenčních technologií, a to i při rostoucím technologickém pokroku v budoucnu.

#### 3.3.2 Pokyny

Zásady dále zastřešují pokyny. Ve verzi WCAG 2.1 je jich celkem 13. Těchto 13 pokynů poskytuje základní cíle, kterých by měli autoři dosáhnout, aby byl obsah webu přístupný uživatelům s různým postižením. Pokyny nejsou testovatelné, ale poskytují obecný rámec a celkové cíle, které pomáhají autorům porozumět kritériím úspěchu a implementovat řešení.[16]

#### 3.3.3 Kritéria přístupnosti

Každý pokyn má několik kritérií úspěšnosti (success criteria), což jsou specifická prohlášení, která nastiňují, co musí být provedeno, aby byl obsah s daným pokynem v souladu.

Přesný počet kritérií úspěšnosti pro každý pokyn se liší, ve verzi WCAG 2.1 lze najít celkem 81 kritérií úspěšnosti označených úrovní shody (A, AA, AAA - více v kapitole Úrovně shody).[16]

### 3.3.4 Dostatečné a doporučené techniky

Pro každý z pokynů a kritérií úspěchu je zdokumentována (ve WCAG 2.0) široká škála technik. Tyto techniky jsou informativní a spadají do dvou kategorií: ty, které jsou dostatečné pro splnění kritérií úspěšnosti, a ty, které jdou nad rámec toho, co jednotlivá kritéria úspěšnosti vyžadují. Techniky autorům umožňují se lépe orientovat v pokynech a procesu zpřístupnění obsahu. Obsahují také ukázky nejčastějších selhání (failures) jednotlivých kritérií.[16]

## 3.4 Úrovně shody

WCAG definuje celkem tři úrovně shody. Jsou označeny jako A, AA a AAA. Jedním slovem by se jednotlivé úrovně shody daly popsat jako minimální (minimal), přijatelná (acceptable) a optimální (optimal). Každá z úrovní obsahuje sadu kritérií, které je nutno dodržet.

### 3.4.1 Úroveň A

Úroveň A je základní, tj. minimální, úrovní přístupnosti, která by měla být dodržena za každých okolností. Webové stránky, které nedosahují této úrovně velmi pravděpodobně obsahují bariéry, které v mnoha případech dokážou lidem s postižením kompletně znemožnit jejich používání. Dosažení této úrovně obvykle nevyžaduje velké úsilí ani výrazné změny v designu, struktuře, či vzhledu aplikace. Na této úrovni se nachází (ve verzi WCAG 2.1) 30 základních kritérií.

#### *Kritéria na úrovni A*

- **Non-text Content** - Veškerý obsah prezentován uživateli, který není ve formě textu, má textovou alternativu, která slouží stejnému účelu, až na několik výjimek jako například captcha, obsah v testu, nebo čistě dekorativní obsah.
- **Audio-only and Video-only (Prerecorded)** - Pro předem nahraný zvukový nebo video obsah existuje alternativa se stejným významem.
- **Captions (Prerecorded)** - Pro předem nahraný zvukový obsah jsou dostupné titulky.

- **Audio Description or Media Alternative (Prerecorded)** - Pro předem nahraný video obsah existuje zvukový nebo textový popis jeho obsahu.
- **Info and Relationships** - Informace, struktura a vztahy odvozené při prezentaci mohou být určeny také programově, nebo jsou dostupné v textu.
- **Meaningful Sequence** - Když pořadí, ve kterém je obsah prezentován, ovlivňuje jeho význam, lze toto pořadí určit rovněž programově.
- **Sensory Characteristics** - Pokyny pro pochopení a manipulaci s obsahem se nespolehají pouze na senzorní vlastnosti komponent jako je tvar, barva, velikost, vizuální umístění, orientace nebo zvuk.
- **Use of Color** - Barva není použita jako jediný způsob přenosu informací, označení akce, vyvolání reakce nebo rozlišení vizuálního prvku.
- **Audio Control** - Pro zvuk hrající automaticky déle než tři vteřiny existuje mechanismus k jeho zastavení, nebo nezávislému ovládní jeho hlasitosti.
- **Keyboard** - Veškerá funkcionalita obsahu (až na výjimky) lze být obsluhovaná pouze s pomocí klávesnice.
- **No Keyboard Trap** - Pokud lze s pomocí klávesnice přesunout fokus dovnitř komponenty, existuje způsob, jak jej pouze s pomocí klávesnice přesunout i ven.
- **Character Key Shortcuts** - Pokud je implementována klávesová zkratka bez pomoci netisknutelných kláves (ctrl, alt), je třeba poskytnout alespoň jeden z následujících mechanismů: deaktivace zkratky, přemapování, aktivace pouze při fokusu.
- **Timing Adjustable** - Pokud nespadá pod některou z výjimek, je třeba, aby časový limit obsahu bylo možno deaktivovat, nastavit, nebo prodloužit.
- **Pause, Stop, Hide** - Pro pohybující se, blikající, rolovací nebo automaticky aktualizovaný obsah, kdy tato funkcionalita není nezbytná, existuje mechanismus umožňující její pozastavení, zastavení, nebo skrytí.
- **Three Flashes or Below Threshold** - Obsah webu neblíká rychleji než třikrát za vteřinu.
- **Bypass Blocks** - Existuje mechanismus pro přeskočení bloků obsahu, které se opakují na více webových stránkách (navigace, hlavička, ...),
- **Page Titled** - Webové stránky mají popisný a srozumitelný titulek.



- **Focus Order** - Pokud lze webovou stránkou procházet sekvenčně, fokusovatelné komponenty obdrží fokus v pořadí, které zachovává význam a funkčnost obsahu.
- **Link Purpose (In Context)** - Smysl odkazu lze určit již z textu obsahu, nebo kontextu ve kterém se nachází.
- **Pointer Gestures** - Veškerý obsah, který k ovládní používá gesta z více bodů nebo cest, lze ovládat pomocí jediného ukazatele bez gesta založeného na cestě, nebo na více bodech, pokud to není nezbytné.
- **Pointer Cancellation** - Akce není provedena při stisknutí tlačítka ukazatele (down-event), ale při jeho uvolnění (up-event), nebo je poskytnut mechanismus zrušení či vrácení akce.
- **Label in Name** - Jméno UI komponenty s popisem je obsaženo v tomto popisku (ideálně na jeho začátku).
- **Motion Actuation** - Akce, které lze kontrolovat pohybem zařízení nebo pohybem uživatele, lze také ovládat komponentami uživatelského rozhraní. Reakci na pohyb lze deaktivovat, aby se zabránilo náhodné aktivaci.
- **Language of Page** - Výchozí jazyk obsahu lze určit programově.
- **On Focus** - Při obdržení fokusu komponentou není realizována změna kontextu (fokus, obsah mění význam stránky, ...).
- **On Input** - Změna nastavení jakékoli komponenty uživatelského rozhraní automaticky nezpůsobí změnu kontextu, pokud uživatel nebyl o chování informován před použitím komponenty.
- **Error Identification** - Pokud je automaticky zjištěna chyba v uživatelském vstupu, je identifikována chybná položka, a chyba je uživateli popsána (v textu).
- **Labels or Instructions** - V případě, kdy je po uživateli vyžadován vstup, mu jsou poskytnuty instrukce nebo popis.
- **Parsing** - Značkový jazyk (např. HTML) použitý pro definici obsahu je validní.
- **Name, Role, Value** - Pro všechny komponenty uživatelského rozhraní, lze název (name) a roli (role) určit programově. Stav, vlastnosti a hodnoty, které může nastavit uživatel, lze nastavit programově, a oznámení o změnách těchto položek je k dispozici uživatelským agentům, včetně asistenčních technologií. Určeno pro vývojáře vlastních komponent, HTML5 komponenty toto kritérium splňují ve výchozím stavu.

### 3.4.2 Úroveň AA

Úroveň AA obsahuje veškeré pokyny z předešlé úrovně A, a přidává 20 nových pokynů. Považuje se za nejvíce žádoucí úroveň. Dosažení AA úrovně znamená, že je web použitelný a pochopitelný pro většinu běžných uživatelů i uživatelů s postiženími, přičemž zástupci obou skupin jsou na webu schopni dosáhnout stejné funkcionality.

Většina organizací se snaží dosáhnout této úrovně jelikož je v mnoha státech legislativou považována za standard webové přístupnosti. Často je také vhodným kompromisem mezi úsilím (prostředky) jež je třeba vynaložit k jejímu dosažení, a mírou přínosu pro uživatele.

#### *Kritéria na úrovni AA*

- **Captions (Live)** - Pro veškerý zvukový obsah přenášený živě jsou poskytnuty titulky.
- **Audio Description (Prerecorded)** - Pro veškerý již existující video obsah existuje zvukový popis.
- **Orientation** - Prezentace obsahu není omezena pouze "na výšku" nebo "na šířku", pokud to není nezbytné.
- **Identify Input Purpose** - Účel vstupních polí musí být možno určit programově. Lze využít HTML atributů jako *type* a *autocomplete*.
- **Contrast (Minimum)** - Texty a obrázky textu mají kontrastní poměr minimálně 4,5:1, s výjimkou velkého textu (ten může mít 3:1), dekorativního textu a textu, který je částí loga.
- **Resize text** - S výjimkou titulků a obrázků textu lze velikost textu zvětšit až o 200 %, bez ztráty obsahu nebo funkčnosti.
- **Images of Text** - Textové informace jsou zobrazeny v textové podobě namísto obrázků textu, není-li to nezbytné.
- **Reflow** - Obsah lze prezentovat bez nutnosti posouvání (scrolling) ve dvou rozměrech. Například při zvětšení textu text přeteče na nový řádek, namísto zobrazení horizontálního posuvníku.
- **Non-text Contrast** - Komponenty uživatelského rozhraní a grafické objekty nesoucí význam mají kontrastní poměr vůči okolí minimálně 3:1.

- **Text Spacing** - Žádný obsah ani funkcionalita nejsou ztraceny při změně řádkování na alespoň 1,5 násobek velikosti písma, mezery za odstavci alespoň na dvojnásobek, mezera mezi písmeny na alespoň 0,12 násobek, mezery mezi slovy na alespoň 0,16 násobek.
- **Content on Hover or Focus** - Obsah zobrazující se při překrytí ukazatelem (hover), lze skrýt bez pohybu ukazatele (např. stiskem ESC). Obsah zůstane viditelný i při přesunu ukazatele ze spouštěcího elementu (trigger) na dodatečný obsah, a dodatečný obsah zůstane viditelný dokud jej uživatel nezavře.
- **Multiple Ways** - Existuje více než jeden způsob nalezení a přesunu na webovou stránku, která je částí sady webových stránek s výjimkou stránek, které jsou součástí procesu.
- **Headings and Labels** - Nadpisy a titulky dostatečně popisují téma nebo obsah.
- **Focus Visible** - Uživatelské rozhraní má mód, kdy je při navigaci s pomocí klávesnice viditelný ukazatel fokusu.
- **Language of Parts** - Jazyk částí obsahu lze programově určit. Existují výjimky jako technické termíny, jména.
- **Consistent Navigation** - Navigační mechanismy, které se opakují na více webových stránkách v rámci sady webových stránek, se pokaždé vyskytují ve stejném pořadí, pokud jej uživatel nezmění.
- **Consistent Identification** - Komponenty uživatelského rozhraní, které mají stejnou funkcionalitu v rámci sady webových stránek, jsou napříč webem identifikovány (popsány) konzistentně.
- **Error Suggestion** - Pokud je automaticky detekována chyba v uživatelském vstupu a jsou dostupné návrhy na její opravu, jsou návrhy poskytnuty uživateli, pokud to neohrožuje bezpečnost nebo účel obsahu.
- **Error Prevention (Legal, Financial, Data)** - U webových stránek, které uživateli způsobují právní závazky, provádějí finanční transakce, upravují nebo mažou uživatelská data, nebo odesílají odpovědi uživatelského testu, jsou takové operace buď reverzibilní, nebo kontrolovány (a je poskytnuta možnost opravy chyb), nebo je lze před odesláním zkontrolovat, opravit a potvrdit.
- **Status Messages** - Stavové zprávy lze určit programově (pomocí role nebo atributů) tak, aby mohly být prezentovány uživateli pomocí asistenčních technologií, aniž by musely být fokusovány.

### 3.4.3 Úroveň AAA

Úroveň AAA je aktuálně nejvyšší a nejpřísnější úrovní přístupnosti dle WCAG. Obsahuje veškeré pokyny z předešlých úrovní A a AA, a navíc ve verzi WCAG 2.1 přidává 28 dalších pokynů.

Často se jedná o další zpřísnění stávajících kritérií jako třeba minimálního kontrastu textu, alternativy zvukového obsahu ve znakovém jazyce, úplný zákaz používání obrázků textu, a další. . .

Splnění této úrovně poskytuje nejlepší zážitek pro nejvyšší počet uživatelů, a umožňuje tak snadnou práci s webem. Ačkoliv by bylo ideální, aby většina webů této úrovně dosahovala, není to vždy možné, jak rovněž informovala i samotná W3C

*"Nedoporučuje se, aby byla úroveň shody AAA vyžadována jako obecná zásada pro celé weby, protože u některého obsahu není možné splnit všechna kritéria úspěšnosti úrovně AAA."*[14]

O splnění této úrovně je ale vhodné usilovat například v případě, kdy je web již konkrétně cílen na uživatele s postižením, či starší generace.

#### *Kritéria na úrovni AAA*

- **Sign Language (Prerecorded)** - Pro veškerý zvukový obsah je poskytnuta alternativa ve znakové řeči.
- **Extended Audio Description (Prerecorded)** - Pro obsah, který nemá dostatečné mezery umožňující přidání dodatečného zvukového popisu je obsah pro tyto účely rozšířen.
- **Media Alternative (Prerecorded)** - Pro veškerý již existující video obsah existuje rozšířený textový popis.
- **Audio-only (Live)** - Pro obsah, který je přenášen živě, existuje textová alternativa.
- **Identify Purpose** - U obsahu definovaného značkovacím jazykem lze programově určit účel UI komponent, ikon a oblastí.
- **Contrast (Enhanced)** - Kontrast textu a obrázků textu je alespoň 7:1, nebo 4.5:1 u velkého textu. Nevztahuje se na dekorativní text a na loga.
- **Low or No Background Audio** - Zvukový obsah obsahující primárně řeč (s výjimkou hudby a CAPTCHA) neobsahuje zvuk na pozadí. Pokud jej obsahuje lze vypnout, nebo je alespoň o 20dB tišší, než řeč.

- **Visual Presentation** - Pro text existuje mechanismus pro změnu barvy popředí a pozadí, šířka není větší než 80 znaků, text není současně zarovnán k levému i pravému okraji (justified) a velikost textu lze změnit bez AT až o 200 %.
- **Images of Text (No Exception)** - Obrázky textu nejsou použity, s výjimkou pro loga a dekorace.
- **Keyboard (No Exception)** - Veškerá funkcionalita (bez výjimek) lze být ovládnána pouze klávesnicí.
- **No Timing** - Události a aktivity nemají časový limit.
- **Interruptions** - Přerušování může uživatel odložit nebo skrýt/ztlumit (mute), s výjimkou přerušování týkajících se nouzových situací.
- **Re-authenticating** - Po vypršení platnosti ověřené relace může uživatel po opětovném ověření pokračovat v činnosti bez ztráty předtím zadaných dat.
- **Timeouts** - Uživatel je upozorněn na možnost ztráty dat v případě neaktivity kratší než 20 hodin.
- **Three Flashes** - Webové stránky neobsahují nic, co by blikalo více než třikrát za sekundu.
- **Animation from Interactions** - Animace spuštěná interakcí lze být deaktivována, není-li nezbytná pro funkcionalitu.
- **Location** - Informace o aktuální lokaci uživatele na webu je mu poskytnuta.
- **Link Purpose (Link Only)** - Účel odkazu lze zjistit pouze z textu odkazu.
- **Section Headings** - Části stránky jsou označeny nadpisy.
- **Target Size** - Velikost interaktivních elementů pro ukazatele je až na výjimky alespoň 44 x 44 pixelů.
- **Concurrent Input Mechanisms** - Uživatel může měnit způsob zadávání v jeho průběhu (např. z dotykové obrazovky na hardwarovou klávesnici).
- **Unusual Words** - Existuje mechanismus pro vysvětlení významu neobvyklých slov.
- **Abbreviations** - Existuje mechanismus pro zobrazení celého názvu zkratk.
- **Reading Level** - V případě, kdy obsah vyžaduje vyšší než středoškolskou úroveň čtení, je poskytnut zjednodušený alternativní obsah.

- 
- **Pronunciation** - Existuje mechanismus pro určení výslovnosti slov, pokud by jinak byl jejich význam v obsahu nejednoznačný.
  - **Change on Request** - Změny kontextu jsou prováděny pouze uživatelem, nebo je lze deaktivovat.
  - **Help** - Je dostupná nápověda ohledně aktuálně prováděné aktivity.
  - **Error Prevention (All)** - U webových stránek, které vyžadují odesílání zadaných dat jsou takové operace buď reverzibilní, nebo kontrolovány (a je poskytnuta možnost opravy chyb), nebo je lze před odesláním zkontrolovat, opravit a potvrdit.

## 4 Webová přístupnost v České republice

### 4.1 Legislativa

V České republice je webová přístupnost popsána zejména Zákonem č. 99/2019 Sb., o přístupnosti internetových stránek a mobilních aplikací. Tento zákon byl později novelizován a dále upraven zákonem 81/2006 sb.[17]

*"Povinný subjekt zajistí, aby jím spravované internetové stránky a mobilní aplikace byly přístupné, a to tím, že je pro jejich uživatele, zejména pro osoby se zdravotním postižením, učiní vnímatelnými, ovladatelnými, srozumitelnými a stabilními." [18]*

Při jeho zkoumání si lze všimnout určité podobnosti s WCAG, kdy jsou požadavky na přístupnost popsány víceméně totožně, jako kategorie pokynů v WCAG, tj. vnímatelný, ovladatelný, srozumitelný, stabilní. . .

#### 4.1.1 Prohlášení o přístupnosti

Prohlášení o přístupnosti je dokument, ve kterém subjekt zveřejňuje informace o přístupnosti svých webových stránek a služeb. Tento dokument obsahuje informace o dodržování příslušných zákonů a směrnic v oblasti webové přístupnosti (sdělení, v jakém rozsahu internetová stránka splňuje zákonné požadavky), a o technických řešeních použitých pro zlepšení přístupnosti. Dále obsahuje informace o možnostech komunikace v případě nabytí podezření o nesplnění zákonných požadavků (nebo nabytí podezření, že uvedený důvod jejich nesplnění není dostatečný), a poskytnutí zpětné vazby pro uživatele s postižením. V případě, že z nějakého důvodu nelze určité části webu zpřístupnit, je zde tato informace uvedena s vysvětlením a s případným náhradním řešením. Osvědčeným postupem je umístění odkazu na tento dokument v patičce stránky.

## Prohlášení o přístupnosti

Univerzita Tomáše Bati ve Zlíně se zavazuje k zpřístupnění svých webových stránek a mobilních aplikací v souladu se zákonem č. 99/2019 Sb., o přístupnosti webových stránek a mobilních aplikací a o změně zákona č. 365/2000 Sb., o informačních systémech veřejné správy a o změně některých dalších zákonů, ve znění pozdějších předpisů.

Toto prohlášení o přístupnosti se vztahuje na obsah internetových stránek Univerzity Tomáše Bati ve Zlíně, jejich fakult a součástí.

### Stav souladu

Uvedené webové stránky jsou plně v souladu požadavky stanovenými zákonem č. 99/2019 Sb. a mezinárodním standardem přístupnosti **WCAG 2.1**. Webové stránky jsou aktualizovány a dále vyvíjeny za dodržování postupů a zásad přístupnosti tak, aby se s jejich obsahem mohli seznámit i uživatelé se specifickými potřebami.

Obr. 4.1 Část prohlášení o přístupnosti na utb.cz

## 4.2 Porovnání se zahraničím

Evropská unie (EU) reguluje přístupnost webu prostřednictvím normy *EN 301 549 V1.1.2 (2015-04)* a směrnice o přístupnosti webu (WAD) *2016/2102 (Directive (EU) 2016/2102 of the European Parliament and of the Council of 26 October 2016 on the accessibility of the websites and mobile applications of public sector bodies)*. V této verzi jsou sice zmínky o WCAG 2.0, ale až novější verze *EN 301 549 V2.1.2 (2018-08)* se přímo odkazuje na WCAG 2.1 AA.

WAD vyžaduje, aby orgány veřejného sektoru zajistily, že jejich webové stránky a mobilní aplikace budou přístupné lidem se zdravotním postižením, včetně těch se zrakovým, sluchovým, motorickým a kognitivním postižením. Cílem směrnice je odstranit překážky a zajistit, aby měl každý rovný přístup k online informacím a službám. Členské státy EU jsou odpovědné za transpozici WAD do svých národních zákonů, a za jeho prosazování.

Například v sousedním Německu je webová přístupnost regulována právními předpisy *BITV (Barrierefreie Informationstechnik-Verordnung)* a *BGG (Bundesbehindertengleichstellungsgesetz)*, které vycházejí z WCAG a dále je rozšiřují. BITV je součástí německého práva a stanovuje požadavky na přístupnost webových stránek a mobilních aplikací pro veřejnou správu. BGG je zákon, který se v Německu zabývá otázkami rovných příležitostí pro osoby se zdravotním postižením. Oba tyto předpisy se snaží zajistit, aby měli všichni stejný přístup k online informacím a službám. Vztahují se na všechny subjekty veřejného sektoru.

V mnoha dalších státech po celém světě je jako právní norma pro webovou přístupnost považováno splnění WCAG 2.0 na úrovni AA. Mezi takové státy patří například Kanada, Japonsko, Spojené království, Austrálie, nebo Indie.[19]

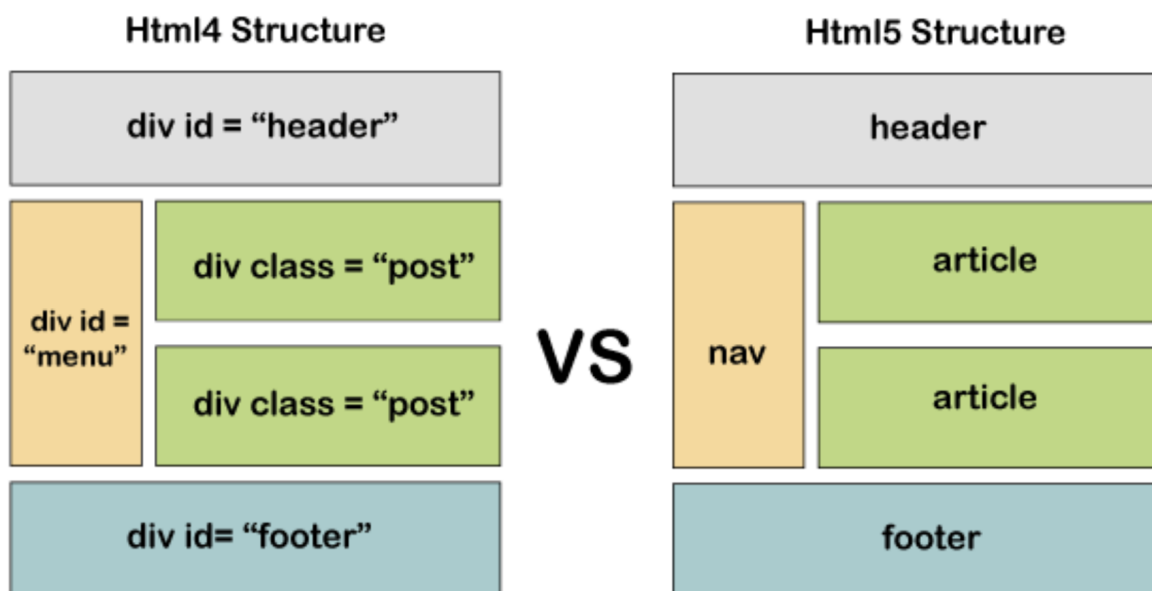


## 5 Sémantické značky v HTML5

Sémantická značka je taková, která v sobě nese informaci o obsahu či struktuře webové stránky, a usnadňuje tak vyhledávačům a asistenční technologii pochopit účel a obsah dané stránky.

Dnes existuje okolo 110 validních HTML značek. V nejnovější verzi HTML5 bylo přidáno 32 nových značek, z nichž je většina přínosem právě pro webovou přístupnost.[20] Nejčastěji používané značky `<div>` a `<span>` nelze považovat jako sémantické, protože pouze označují část textu, nebo obecné seskupení elementů, ale dále již neříkají nic o jejich roli v rámci stránky. Vývojáři webových stránek by se tedy měli snažit tyto obecné značky nahradit sémantickými vždy, když je to možné. Například podle kontextu lze značku `<span>` často nahradit značkami `<p>`, `<label>`, `<h1>`, apod... Obdobně lze často `<div>` nahradit vhodnějšími značkami jako `<header>`, `<article>`, `<nav>`, a dalšími...

Pro správné fungování asistenčních technologií je sémantické značky obzvláště důležité používat při tvorbě základní struktury stránky.



Obr. 5.1 Sémantické značky v HTML5 [21]

Jako jedny z nejdůležitějších (v abecedním pořadí) sémantických značek lze zmínit:

- **<article>** - Samostatný obsah, jako například příspěvek na blogu.
- **<aside>** - Obsah mimo hlavní obsah stránky, často se používá pro postranní sloupec.
- **<details>** - Dodatečné podrobnosti, které uživatel může zobrazit nebo skrýt.
- **<em>** - Zdůrazňuje text, který se obvykle čte s nápadnějším tónem. Pro tento účel se dnes často nesprávně používá zastaralý element **<i>**.
- **<figcaption>** - Popisek pro prvek **<figure>**.
- **<figure>** - Obsah jako jsou ilustrace, fotografie, seznamy kódu a další.
- **<footer>** - Obsah zápatí dokumentu nebo sekce.
- **<header>** - Obsah hlavičky dokumentu nebo sekce.
- **<main>** - Hlavní obsah dokumentu, v rámci jedné stránky by se měl vyskytnout pouze jednou.
- **<mark>** - Označený/zvýrazněný text.
- **<nav>** - Sekce odkazů pro navigaci.
- **<p>** - Definuje odstavec textu.
- **<section>** - Sekce dokumentu, jako jsou kapitoly, nebo jiné sekce dokumentu.
- **<strong>** - Zdůrazňuje text, který je důležitý, nebo který má být výraznější. Pro tento účel se dnes často nesprávně používá zastaralý element **<b>**.
- **<summary>** - Viditelný nadpis pro prvek **<details>**, na který lze kliknout pro zobrazení nebo skrytí dodatečného obsahu.

Použití některých z výše zmíněných značek je demonstrováno na konkrétních ukázkách v praktické části práce.

## 6 Wai-aria v HTML5

WAI-ARIA (*Web Accessibility Initiative - Accessible Rich Internet Applications*) je technická specifikace od World Wide Web Consortium (W3C). Nabízí způsob jak poskytnout asistenčním technologiím, jako jsou čtečky obrazovky, dodatečné sémantické informace o prvcích uživatelského rozhraní (HTML elementech, značkách) na webové stránce, což lidem s postižením dále usnadňuje porozumění obsahu a interakci s ním.[8]

WAI-ARIA byla ve verzi 1.0 poprvé zveřejněna 20. března roku 2014. V prosinci roku 2017 se dočkala vylepšení v podobě WAI-ARIA 1.1, a v současnosti již probíhají práce na nejnovější verzi 1.2.

WAI-ARIA rovněž podporuje dynamické aktualizace a pokročilé uživatelské interakce, jako jsou oblasti typu drag-and-drop a živé oblasti (live region, část stránky která se může změnit i bez jejího kompletního obnovení, např. nová zpráva v chatu).

Je nutno podotknout, že ARIA pouze upravuje strom přístupnosti a mění způsob, jakým asistenční technologie prezentují obsah uživatelům. ARIA **nemění** nic na funkci nebo chování daného prvku. [22]

### 6.1 1. Pravidlo ARIA

Odborníci na webovou přístupnost a ARIA se s nadsázkou shodují, že prvním pravidlem při práci s ARIA, je vůbec ARIA atributy nepoužívat. To v praxi znamená, že při vytváření přístupného webu by mělo být vždy v první řadě usilováno o využití nativních elementů a technologií, které jsou přístupné již ve výchozím stavu "out-of-the-box", tzn. nevyžadují žádné dodatečné úpravy ze strany vývojáře.

Pomocí ARIA atributů lze poskytovat dodatečné informace, pokud situace neumožňuje tyto informace poskytnout nativně, například použitím sémantických značek.

To lze předvést na následující ukázce.

```
<span
  role="button"
  tabindex="0"
  onclick="login()"
  onkeypress="if (event.key === 'Enter') login()"
  style="cursor: pointer;"
>
  Log in
</span>
```

V prvním příkladu výše byla k vytvoření komponenty tlačítka použita značka `<span>`. Atribut `role` musí být nastaven na `"button"`, aby bylo asistenčním technologiím oznámeno, že by měl být tento prvek považován za tlačítko. Atribut `tabindex` je třeba nastavit na

0, aby byl prvek fokusovatelný. Atribut *onkeypress* slouží k specifikaci funkce, která má být vykonána, když je stisknuta klávesa Enter, zatímco má tlačítko fokus, jako by tomu bylo u normálního tlačítka. Navíc je pomocí CSS třeba nastavit vlastnost *cursor* na *"pointer"*, aby bylo zřejmé, že na prvek lze kliknout.

```
<button onclick="login()">Přihlasit</button>
```

V druhém případě je k vytvoření totožného tlačítka použito nativního HTML elementu *<button>*, který již má v sobě veškerou tuto funkcionalitu zabudovanou interně. Co se týče přístupnosti jsou ve výsledku obě tlačítka v pořádku, ale je zřejmé, že pro korektní řešení, kdy je tlačítko vybudováno s pomocí *<span>* elementu bylo třeba vynaložit daleko víc úsilí a výsledný kód je obsáhlejší, a tím pádem méně čitelný a více náchylný k chybám.

## 6.2 Členění atributů

V době psaní práce je definováno celkem 53 ARIA atributů, které se člení do následujících čtyř kategorií.[22]

### 6.2.1 Atributy widgetů

Atributy widgetů jsou specifické pro běžné prvky uživatelského rozhraní, které přijímají uživatelský vstup a zpracovávají uživatelské akce (tlačítka, vstupní pole, dialogové okna).[23]

Ve verzi WAI-ARIA 1.1 je jich 24, a mezi nejzajímavější z nich patří:

- **aria-hidden** - Indikuje, zda by měl být prvek skrytý před AT, například obsah opakovaný na stránce, nebo grafický obsah čistě pro dekorativní účely.
- **aria-label** - Poskytuje asistenčním technologiím popis pro prvek, typicky vstupní pole, v případě, kdy jej nelze poskytnout standardním způsobem. (například když vstupní pole nemá viditelný popis, třeba u vyhledávacího filtru).

```
<input type="text" placeholder="Maly princ" aria-label="Nazev knihy"/>
```

- **aria-required** - Indikuje, že je vyžadováno vyplnění prvku před odesláním formuláře.
- **aria-expanded** - Indikuje současný stav kolapsibilního (collapsible) prvku, jako je například rozevírací menu nebo "skládací text" (accordion).
- **aria-selected** - Indikuje, že prvek je aktuálně vybrán, například záložka v seznamu záložek.

- **aria-errormessage** - Identifikuje prvek, který obsahuje chybovou zprávu související s jiným prvkem formuláře.
- **aria-disabled** - Indikuje, že prvek je nefunkční, což ho činí neinteraktivním. Například u vstupních komponent "na míru", které nepoužívají nativní značky.

### 6.2.2 Atributy živé oblasti (live-region)

Tyto atributy lze použít na jakýkoli prvek. Účelem těchto atributů je naznačit, že mohou nastat změny obsahu, aniž by byl prvek fokusován (focused), či došlo k obnovení stránky, a poskytnout asistenčním technologiím informace o tom, jak s těmito aktualizacemi obsahu naložit. Některé role určují výchozí hodnotu pro atribut *aria-live* specifický pro danou roli. Příkladem živé oblasti je sekce tickeru, která uvádí aktualizované kurzy akcií. Tato kategorie obsahuje čtyři atributy.[23]

- **aria-atomic** - Indikuje, že aktualizace živé oblasti prvku by měly být uživateli prezentovány jako celek, nikoliv postupně.
- **aria-busy** - Indikuje, že prvek se nachází ve stavu aktivity a zpracovává úkol. Například pokud dochází ke změně výsledku matematické rovnice v důsledku změny vstupních parametrů, ale výpočet je časově náročnější a nový výsledek není ještě známý.

```
<button aria-busy="false" onclick="loadContent()">
  Obnovit obsah
</button>
```

Po kliknutí na tlačítko lze při zpracovávání požadavku pomocí JavaScriptu nastavit hodnotu *aria-busy="true"* a při dokončení požadavku opět zpátky na *aria-busy="false"*.

- **aria-live** - Určuje typ živé oblasti, jako například polite nebo assertive, což indikuje úroveň důležitosti aktualizací v živé oblasti a způsob jejich následné prezentace uživateli.
- **aria-relevant** - Indikuje typ změn obsahu, které by měly vyvolat aktualizaci živé oblasti. jako například přidání (addition), odstranění (removal), změny textu (text), nebo všechny změny (all).

### 6.2.3 Atributy Drag-and-drop

Tato část uvádí atributy, které označují informace o prvcích rozhraní přetažení, jako jsou přetahovatelné prvky a jejich cíle přetažení. Informace o cíli přetažení budou vykresleny vizuálně autorem a poskytnuty asistenčním technologiím prostřednictvím alternativního způsobu. Jsou pouze dva.[23]

- **aria-dropeffect** - Indikuje typ operace přetažení a puštění, která je na prvku možná, jako například kopírování (copy), přesun (move) nebo odkaz (link). Ve verzi WAI-ARIA 1.1 je označen jako zastaralý (deprecated) a očekává se, že bude v příštích verzích nahrazen.
- **aria-grabbed** - Indikuje, zda je prvek "uchycen", a je připraven k přetažení a puštění. Ve verzi WAI-ARIA 1.1 je označen jako zastaralý (deprecated) a očekává se, že rovněž bude v příštích verzích nahrazen.

### 6.2.4 Atributy vztahu

V této kategorii jsou atributy, které indikují vztahy a asociace mezi elementy, které jinak nejsou patrné ze struktury dokumentu.[23] Spadá sem 16 atributů a mezi nejzajímavější patří níže uvedené.

- **aria-activedescendant** - Obsahuje identifikátor aktivního potomka, který je aktuálně používán jako aktivní prvek v rámci rodičovského prvku.
- **aria-controls** - Určuje identifikátory prvků, které jsou ovládány aktuálním prvkem.
- **aria-labelledby** - Obsahuje identifikátor prvku, který popisuje aktuální prvek. Typicky vstupní pole, jehož popisem není klasický `<label>`.
- **aria-rowcount** - Určuje celkový počet řádků v tabulce nebo seznamu.

## 7 Focus management

Anglický pojem "focus" je do češtiny poněkud těžce přeložitelný, ale rozumí se jím prosté vybrání, zvýraznění či fokus elementu webové stránky. Nepsaným standardem je použití dostatečně širokého modrého ohraničení pro aktuálně zaměřený interaktivní element. Díky tomu dokáže uživatel snadno rozeznat, který element je na stránce zrovna aktivní, což je užitečné i pro všechny běžné uživatele.

WCAG Kritérium úspěchu 2.4.7 Focus Visible (Úroveň AA) vyžaduje, aby bylo toto ohraničení viditelné pro elementy jako tlačítka, vstupní pole, odkazy a ostatní interaktivní prvky sloužící k manipulaci s obsahem. Ke stylování tohoto ohraničení se nejčastěji využívá CSS atributu *outline*.

Vstupní pole bez "fokusu":

Obr. 7.1 Nepřístupné vstupní pole

Vstupní pole s "fokusem":

Obr. 7.2 Přístupné vstupní pole

### 7.1 Tabindex

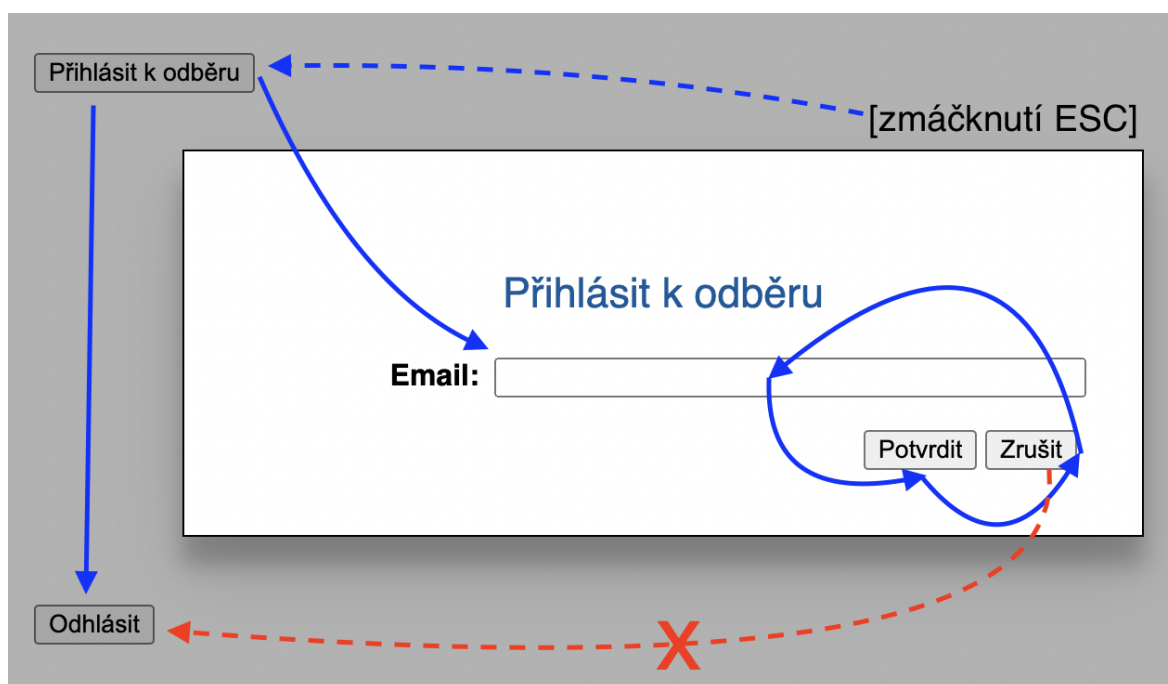
Index tabulátoru (*tabindex*) je atribut který lze přiřadit HTML elementům, a který označuje pořadí, ve kterém budou prvky fokusovány, když uživatel naviguje pomocí klávesy tab. Prvky s hodnotou indexu tabulátoru 0 jsou zahrnuty do standardního pořadí, a prvky s kladnou hodnotou jsou navigovány v pořadí určeném jejich *tabindex* hodnotou (*tabindex* 1 bude zvýrazněn první, následuje prvek s *tabindex* 2, atd.). Prvky, kterým je nastavena záporná hodnota, jsou z pořadí zcela vyjmuty (tj. vůbec je nelze při navigaci tabulátorem zvýraznit). Tyto prvky je možné zvýraznit pouze programově, s pomocí JavaScriptu.

## 7.2 Tok fokusu (focus flow)

Termín "*focus flow*" odkazuje na pořadí ve kterém jsou prvky fokusovány při navigaci pomocí klávesnice nebo asistenční technologie, jako je čtečka obrazovky. Tok fokusu je určen pořadím prvků ve zdrojovém kódu a strukturou dokumentu HTML, přičemž by fokus měl obecně plynout shora dolů a zleva doprava. V případě, kdy prvky, které jsou ve zdrojovém kódu na jiných místech jsou zobrazovány po sobě, by mělo být s pomocí *tabindex* atributu docíleno toho, že bude zachován standardní tok fokusu. I zde ovšem platí 1. pravidlo webové přístupnosti, nepoužívat atributy webové přístupnosti, a místo použití *tabindexu* se snažit o docílení přístupnosti nativním způsobem, v tomto případě tedy seskupením prvků ve správném pořadí již ve zdrojovém kódu.

## 7.3 Fokus past (focus trap)

Tímto pojmem je označován stav, kdy se z nějakého důvodu dostane uživatel během navigace stránkou pomocí tabulátoru do smyčky, kterou není schopen opustit, i když se na stránce reálně nachází více interaktivních elementů. Při vývoji webu je třeba dbát na to, aby takové stavy nebyly dosažitelné a focus flow testovat. V některých případech je ale toto chování žádoucí. Může to být typicky v situaci, kdy se uživateli otevře jednoduché dialogové okno vyzývající k potvrzení či zrušení aktuální operace. V takovém okamžiku není žádoucí aby fokus opustil toto okno a umožnil manipulaci se stránkou překrytou tímto oknem, ale je třeba aby byl "uvězněn" pouze v novém okně dokud jej uživatel nezavře (typicky pomocí tlačítka nebo zmáčknutím ESC).



Obr. 7.3 Uzamčení fokusu v dialogovém okně [25]



## II. PROJEKTOVÁ ČÁST

## 8 Nástroje pro vývoj a testování přístupnosti

### 8.1 Čtečky obrazu (Screen readers)

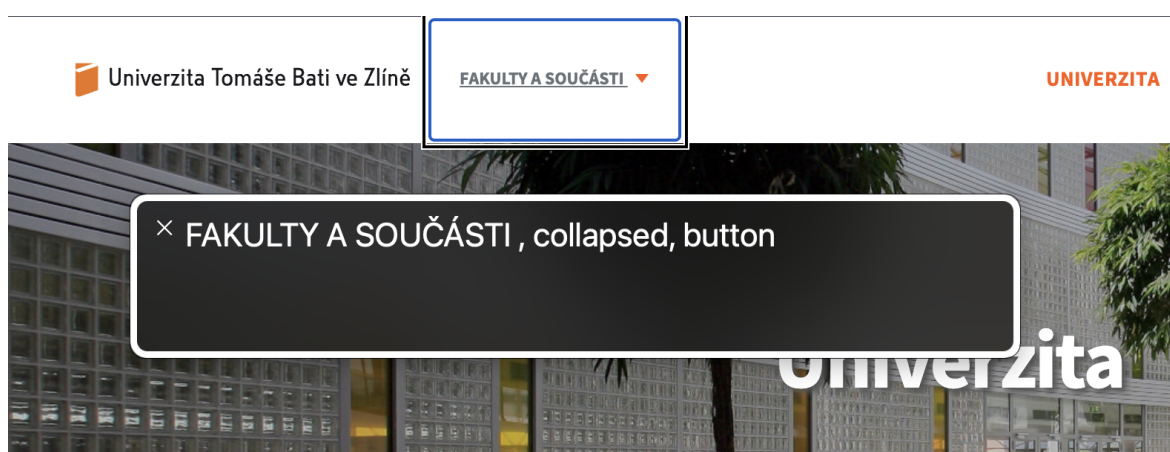
Čtečka obrazovky je softwarová aplikace, která převádí skryté (alternativní) popisky a text zobrazený na obrazovce počítače do řeči nebo výstupu v Braillově písmě (Obr. 8.1). Čtečky obrazovky umožňují uživatelům se zrakovým postižením používat počítač a přistupovat k informacím na něm nebo na jiných digitálních zařízeních. Konkrétní implementace se mohou lišit, ale typicky softwarové čtečky obrazu nabízejí dva módy, kdy jeden slouží pro rychlou navigaci stránkou a čte především nejdůležitější a interaktivní prvky, a druhý, který čte úplně všechno.



Obr. 8.1 Braille displej Brailliant BI 40X [25]

Některé příklady široce používaných čteček obrazovky zahrnují:

- **VoiceOver** je čtečka obrazovky zabudovaná do operačních systémů Apple Mac OS X a iOS. Poskytuje intuitivní rozhraní pro navigaci a přístup k informacím na počítačích Mac a iOS zařízeních. Pro její používání není třeba samostatné licence a lze ji tedy považovat za "bezplatnou". Voiceover nabízí širokou škálu nastavení, ve výchozím módu zobrazuje čtený text v okně s vysokým kontrastem a současně jej uživateli čte s pomocí TTS (Text to speech - převod textu na hlas) technologie (Obr. 8.2).
- **JAWS (Job Access With Speech)** vyvinutý společností Freedom Scientific, je jednou z nejoblíbenějších čteček obrazovky, která nabízí komplexní sadu funkcí pro Windows i webové aplikace. Tato čtečka je ale placená, a tak nemusí být vhodná pro všechny uživatele.
- **NVDA (Non-Visual Desktop Access)** je bezplatná (funguje na principu dárčovství) a open source čtečka obrazovky, která funguje na operačním systému Windows. Poskytuje podporu pro širokou škálu aplikací, včetně webových prohlížečů a kancelářského softwaru. Na oficiálním webu <https://www.nvaccess.org/> jsou k zakoupení dostupné také výukové materiály pro používání NVDA, například pro práci s kancelářskou sadou Microsoft Office.
- **TalkBack** vyvinutý společností Google, je čtečka obrazovky, která je součástí operačního systému Android a je nejčastěji používána na mobilních zařízeních.



Obr. 8.2 Ukázka výstupu VoiceOver

## 8.2 Lighthouse

Lighthouse je open-source nástroj zabudovaný do webového prohlížeče Google Chrome, který pomáhá auditovat a zlepšovat kvalitu webových stránek. Lighthouse kontroluje výkon, úroveň SEO (Search Engine Optimization), respektování osvědčených postupů, kompatibilitu s PWA (Progressive Web Applications), mobilní zobrazení, a především přístupnost. V případě přístupnosti, Lighthouse kontroluje širokou škálu vlastností, některými z nichž jsou:

- *alt* text u obrázků
- Správné použití sémantických HTML elementů
- Dostatečný kontrast mezi textem a pozadím
- Správné zacházení s fokusem
- ARIA atributy a jejich použití
- Přístupnost při použití klávesnice

Zatímco je Lighthouse bezpochyby užitečným nástrojem pro tvorbu a audit webových stránek i sám Google zdůrazňuje, že ne všechny příležitosti ke zlepšení přístupnosti lze odhalit pomocí automatizovaných testů, a je vhodné na něj navázat i testováním manuálním.[24]

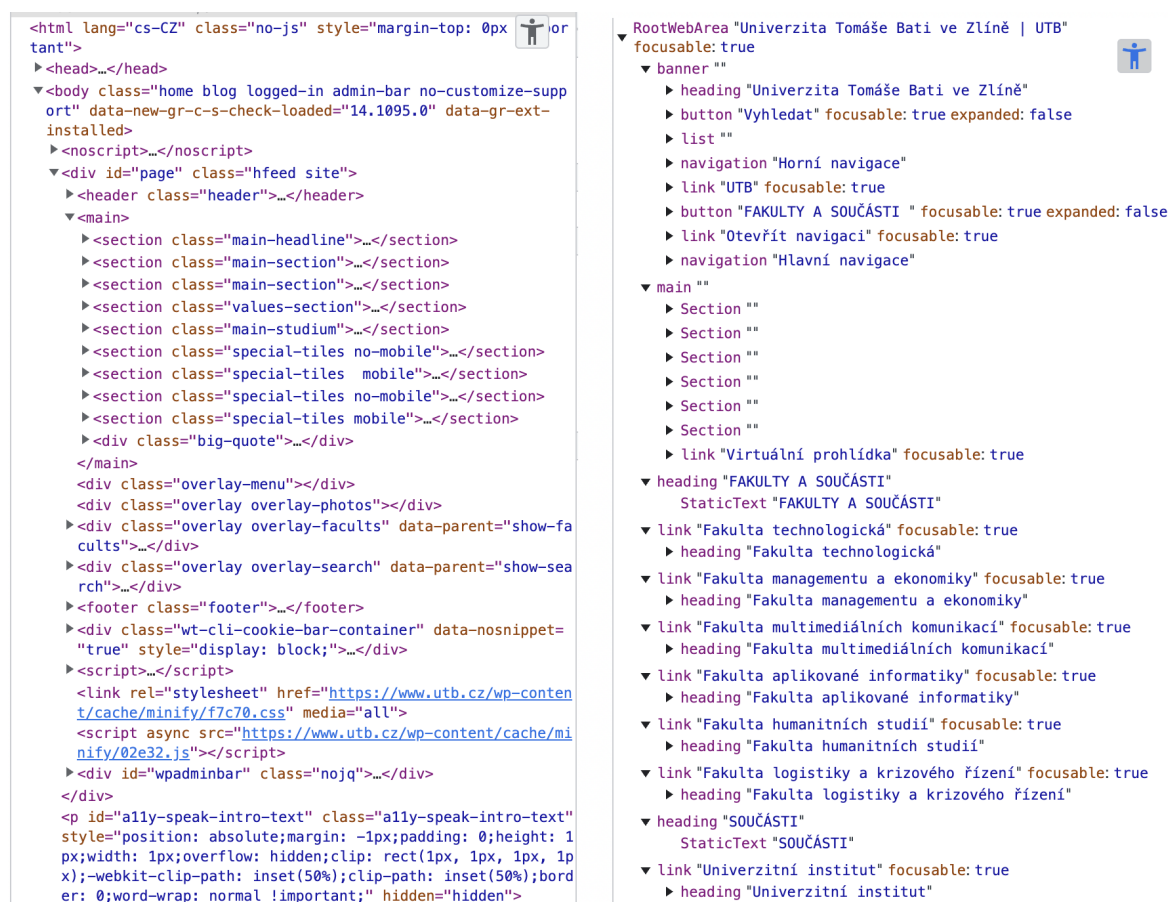
Za pomocí manuálních testů je doporučeno zaměřit se například na to, že:

- Ovládací prvky na míru mají role ARIA a přiřazené štítky (labels)
- Fokus není omylem uvězněn v jedné oblasti a jeho plynutí je intuitivní
- Obsah mimo obrazovku je skrytý před asistenční technologií
- Jsou použity sémantické prvky HTML5 ke zlepšení navigace (Více v kapitole Sémantické značky v HTML5)
- Vizuální pořadí komponent na stránce odpovídá jejich pořadí DOM-u (Document Object Model)

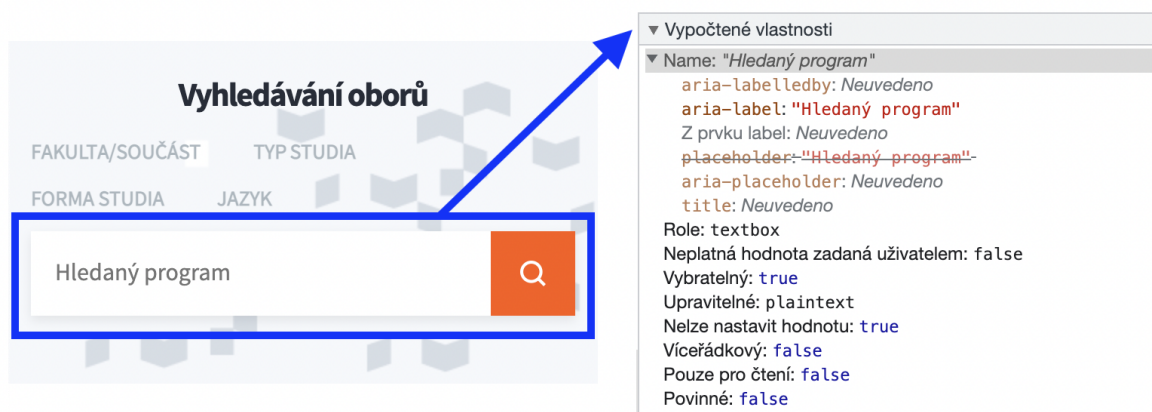
Využití tohoto nástroje pro kontrolu přístupnosti v praxi je demonstrováno v pozdější kapitole "Zkušební audit".

### 8.3 Chrome DevTools

Mimo nástroj Lighthouse Google ve svém prohlížeči nabízí i další prostředky, které lze využít pro práci s přístupností. Jedním z nich je panel „Přístupnost“ v nástroji DevTools, který poskytuje především informace o stromu přístupnosti (accessibility tree) stránky (Obr. 8.3). Strom přístupnosti představuje strukturu a obsah stránky, včetně informací o rolích, stavech a vlastnostech prvků, které jsou pro přístupnost významné. Pomocí panelu Přístupnost mohou vývojáři tento strom zkontrolovat a zjistit, jak odpovídá vizuálnímu rozvržení stránky, což usnadňuje identifikaci a opravu problémů s přístupností. Na panelu Přístupnost lze také najít informace o použitých komponentách jako je jejich role, název, popis, aria-atributy, stav každého prvku a také klávesové zkratky a posluchače událostí, které jsou k prvku připojeny. (Obr. 8.4) To je například užitečné pro kontrolu, zda všechna vstupní pole mají svůj název odvozený od jejich popisku, nebo ARIA atributů aria-label či aria-labelledby.

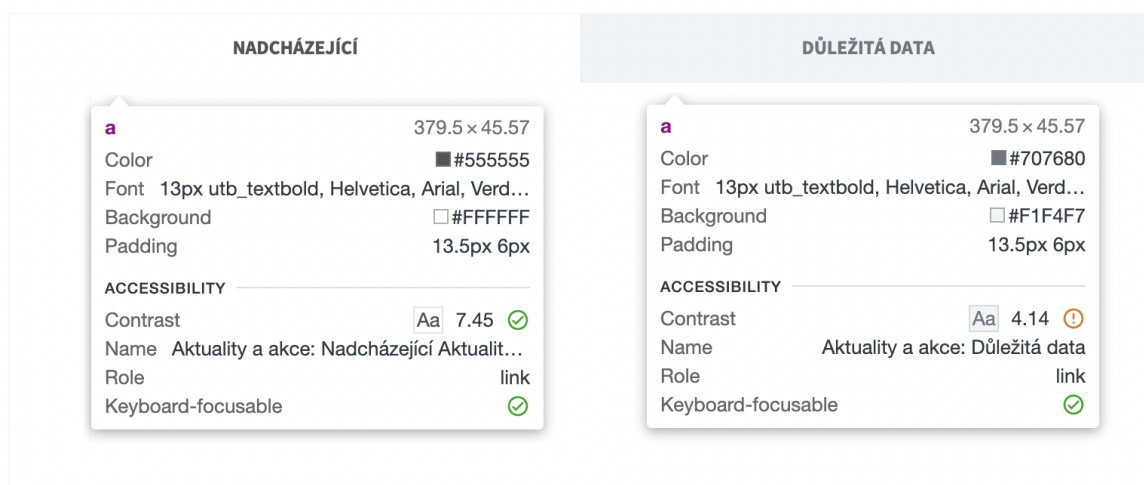


Obr. 8.3 DOM strom vs. A11Y strom



Obr. 8.4 Vlastnosti prvku v panelu přístupnosti

I s pomocí akce "prozkoumat" lze získat stručné informace o přístupnosti aktuálně vybraného prvku. Tato funkce je užitečná pro rychlou kontrolu minimálního kontrastu barev, rolí, existenci popisků komponent (name) a fokusovatelnost pomocí klávesnice (Obr. 8.5).

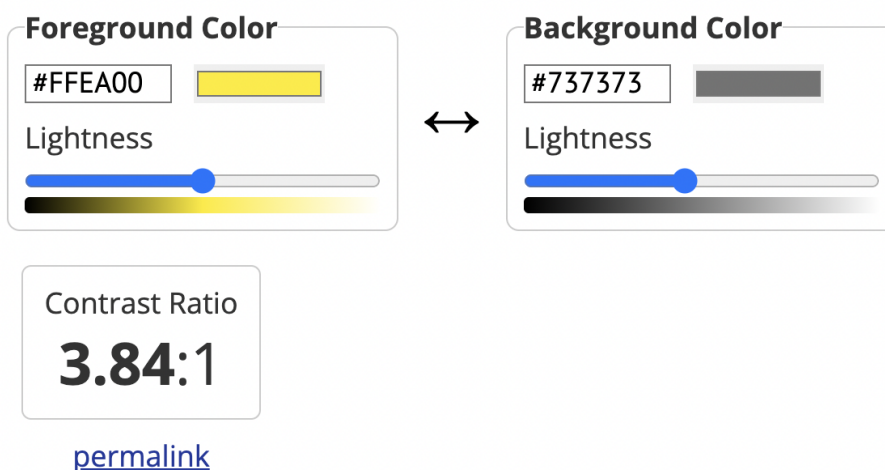


Obr. 8.5 Odhalení nedostatečného kontrastu s pomocí DevTools

## 8.4 Práce s barvami

### 8.4.1 Měření kontrastu

Pro měření kontrastu barev existuje nepřehledné množství nástrojů. V předchozí kapitole již byla zmíněna ke kontrole kontrastu možnost využití Chrome DevTools. Lze ale také najít jiné online nástroje, které nabízejí více funkcí. Například web [webaim.org](http://webaim.org) umožňuje porovnání a výpočet kontrastu mezi barvou textu a barvou pozadí (Obr. 8.6), nebo třeba vzájemné porovnání barev textu, odkazů a pozadí (Obr. 8.7). Pro všechna tato porovnání se rovněž zobrazuje, zda daný kontrast splňuje relevantní WCAG kritéria, a nabízí další informace k tématu. Kromě online nástrojů pro testování přístupnosti web také obsahuje články, blog, a další výukové materiály (v angličtině).



### Normal Text

WCAG AA: **Fail**  
WCAG AAA: **Fail**

The five boxing wizards jump quickly.

### Large Text

WCAG AA: **Pass**  
WCAG AAA: **Fail**

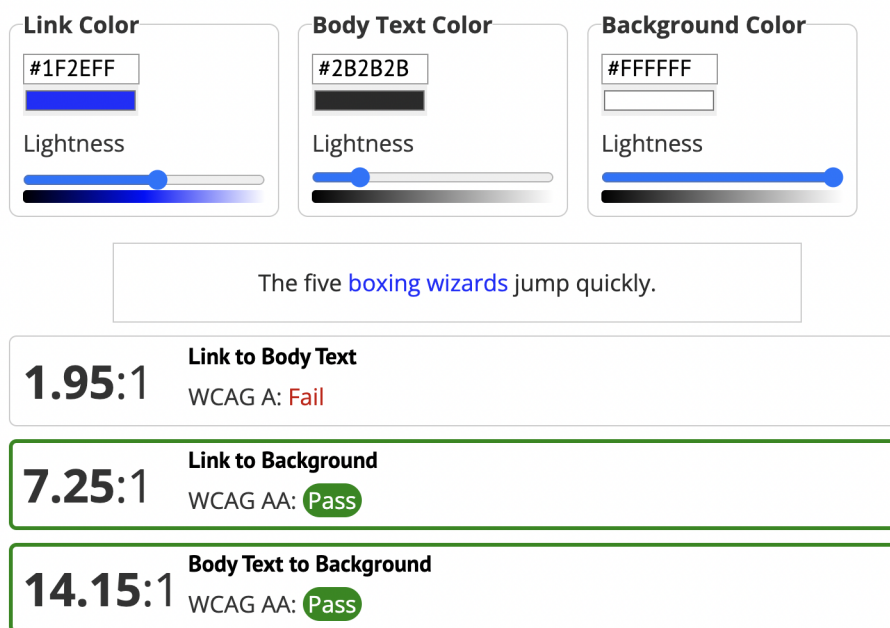
The five boxing wizards jump quickly.

### Graphical Objects and User Interface Components

WCAG AA: **Pass**

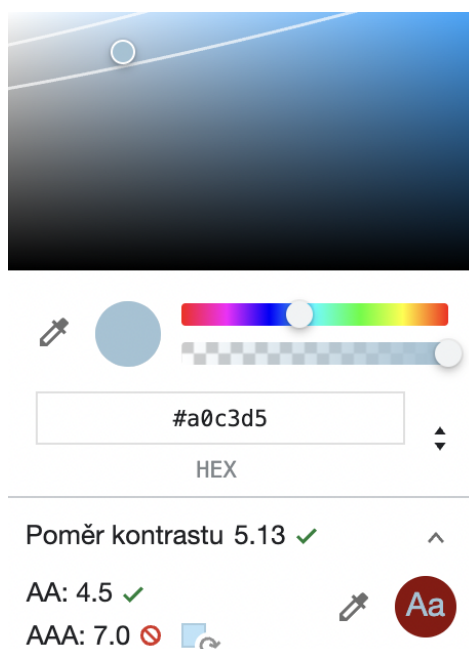
Text Input ✓

Obr. 8.6 Kontrast text vs. pozadí



Obr. 8.7 Kontrast odkaz vs. text vs. pozadí

Užitečná je i funkce výběru barvy v nástroji Google Chrome DevTools (Obr. 8.8). Ta automaticky porovnává barvu textu s pozadím a nabízí interaktivní výběr z oblastí splňující kritéria WCAG.



Obr. 8.8 Výběr kontrastních barev s pomocí DevTools

Z desktopových aplikací lze poté zmínit třeba Color contrast analyzer pro Windows, či Color Contrast pro zařízení s MacOS.



### 8.4.2 Simulace barvosleposti

Pro představu, jak web můžou vnímat uživatelé trpící barvoslepostí existuje užitečné rozšíření prohlížeče Google Chrome s názvem *Let's get color blind*. Toto rozšíření bylo použito pro generování obrázků v kapitole Barvoslepost a webová přístupnost.



Obr. 8.9 Let's get color blind - normální barvy



Obr. 8.10 Let's get color blind - deuteranopie



Obr. 8.11 Let's get color blind - Tritanopie

## 8.5 HTML validátory

HTML validátor je nástroj používaný ke kontrole syntaxe a struktury kódu. Kontroluje se, že kód splňuje standardy stanovené W3C. Validátor kontroluje chyby jako jsou chybějící značky, nesprávné vnoření prvků a další problémy, které by mohly způsobit problémy se zobrazením a funkčností webových stránek. Validní kód zlepšuje přístupnost webu, a rovněž napomáhá k lepšímu SEO. Existuje několik online validátorů HTML dostupných zdarma. Jedním z nich je například *W3C HTML Validator* dostupný na adrese <https://validator.w3.org/>. Nabízí možnost validace kódu zadáním URL adresy webu, ale i vložením samotného kódu přímo ve formě souboru nebo textu.



Checker Input

Show  source  outline  image report 

Check by   css

```
<span aria-labeledby="id">
  <div>Tento zápis není validní!</div>
  <a href="#">odkaz
</span>
```

Obr. 8.12 Nevalidní kód připravený k ověření



- Error** Attribute `aria-labeledby` not allowed on element `span` at this point.  
From line 1, column 1; to line 1, column 26  
`<span aria-labeledby="id"><di`  
Attributes for element `span`:  
[Global attributes](#)
- Error** Element `div` not allowed as child of element `span` in this context.  
(Suppressing further errors from this subtree.)  
From line 2, column 3; to line 2, column 7  
`y="id"><div>Tento`
- Error** Attribute `href` not allowed on element `a` at this point.  
From line 3, column 1; to line 3, column 12  
`ní!</div><a href="#">odkaz`
- Error** End tag `span` seen, but there were open elements.  
From line 4, column 1; to line 4, column 7  
`"#>odkaz</span>`
- Error** Unclosed element `a`.  
From line 3, column 1; to line 3, column 12  
`ní!</div><a href="#">odkaz`

Obr. 8.13 Výstup validátoru - výčet chyb

Na příkladu z Obr. 8.13 lze vidět, že s pomocí validátoru bylo odhaleno nevalidní vnořování elementů (blokový `<div>` nelze být vnořen uvnitř inline `<span>`), v atributech `href` a `aria-labelledby` byly překlepy, a rovněž chybí uzavírací značka `</a>`.

## 8.6 Bookmarklety

Bookmarklety jsou malé JavaScriptové programy, které lze uložit jako záložky ve webovém prohlížeči. Po otevření záložky se program spustí v kontextu aktuální webové stránky a může provádět různé akce, jako jsou změna vzhledu stránky a přidání nebo úprava obsahu. Bookmarklety představují pohodlný způsob rozšíření funkcí prohlížeče, aniž by bylo třeba cokoli instalovat (na rozdíl od samostatných rozšíření (extensions) prohlížeče). Vzhledem k tomu, že bookmarklety jsou pouze kódy JavaScriptu uloženy v záložce, jsou platformě nezávislé. Lze je použít v jakémkoli prohlížeči, který podporuje JavaScript.

V rámci webové přístupnosti je lze použít například pro přidání alternativního textu k obrázkům, zvýraznění odkazů, zvětšení velikosti textu, zlepšení barevného kontrastu, přidání navigace pomocí klávesnice, zvýraznění struktury stránky, a mnoho dalších. . .

### Test Content

[PaulJAdam.com](#) `tabindex="2"`

`tabindex="0"`

[Deque](#) `tabindex="0"`

[Google](#) `tabindex="-1"`

`X` `tabindex="0"`

Lorem ipsum  
dolor sit  
amet,  
consectetuer  
adipiscing

`tabindex="0"`  
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

`tabindex="0"`  
Lorem ipsum

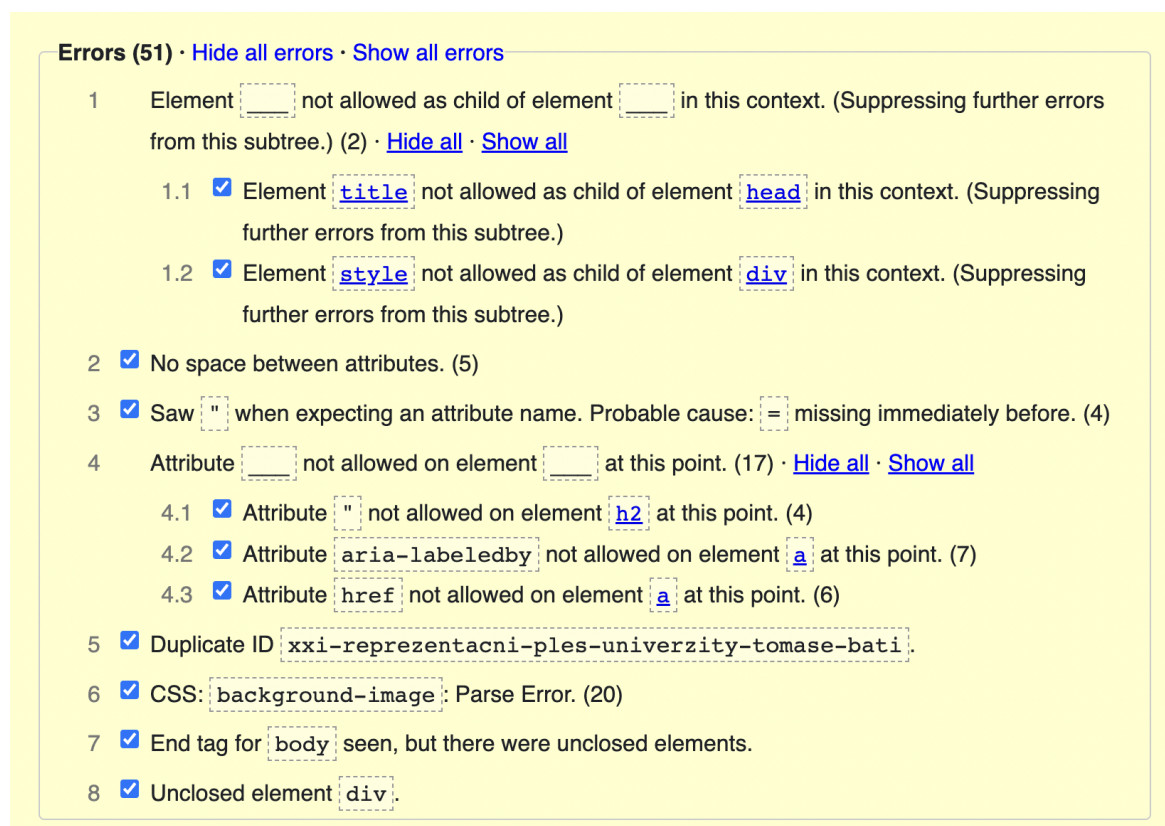
Obr. 8.14 Bookmarklet z pauljadam.com pro kontrolu hodnoty tabindex [26]

## 8.7 Zkušební audit

V této kapitole je na domovské stránce webu `utb.cz` demonstrováno, jakým způsobem lze provést základní audit a otestovat webovou přístupnost. Je využito dříve zmíněných nástrojů, konkrétně HTML Validátoru `validator.w3.org`, Google Lighthouse, Google Devtools a manuálního testování.

### 8.7.1 HTML Validátor

V tomto případě lze u online W3 validátoru využít možnosti validace stránky podle URL adresy, a není tak třeba kopírovat celý její zdrojový kód. Jak lze vidět na výpisu níže (Obr. 8.13), už v tomto kroku bylo odhaleno několik nedostatků. Nejčastěji se jedná o drobné chyby a překlepy, kdy například přebývá znak uvozovky `"`, chybí uzavírací závorky `)"`, chybí mezery mezi atributy, nebo je špatně napsáno `aria-labelledby` (chybí jedno písmeno `'l'`). Dalšími odhalenými nedostatky jsou duplicitní id elementu (v rámci jedné stránky by se nikdy nemělo opakovat stejné id) a chybějící uzavírací tag `</div>`. Většina z těchto nedostatků je v rozporu s kritériem úspěšnosti 4.1.1: Parsing (Úroveň A).



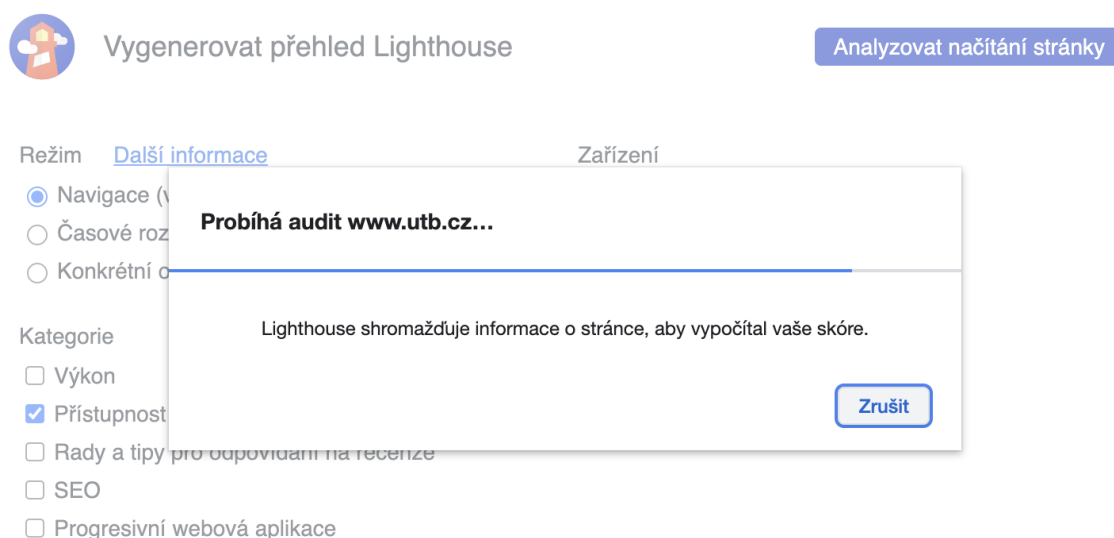
The image shows a screenshot of an HTML validator's error report. At the top, it says "Errors (51) · Hide all errors · Show all errors". The errors are listed in a numbered list:

- 1 Element `<div>` not allowed as child of element `<div>` in this context. (Suppressing further errors from this subtree.) (2) · [Hide all](#) · [Show all](#)
  - 1.1  Element `<title>` not allowed as child of element `<head>` in this context. (Suppressing further errors from this subtree.)
  - 1.2  Element `<style>` not allowed as child of element `<div>` in this context. (Suppressing further errors from this subtree.)
- 2  No space between attributes. (5)
- 3  Saw `"` when expecting an attribute name. Probable cause: `=` missing immediately before. (4)
- 4 Attribute `<div>` not allowed on element `<div>` at this point. (17) · [Hide all](#) · [Show all](#)
  - 4.1  Attribute `<div>` not allowed on element `<h2>` at this point. (4)
  - 4.2  Attribute `<aria-labelledby>` not allowed on element `<a>` at this point. (7)
  - 4.3  Attribute `<href>` not allowed on element `<a>` at this point. (6)
- 5  Duplicate ID `<xxi-reprezentacni-ples-univerzity-tomase-bati>`.
- 6  CSS: `<background-image>`: Parse Error. (20)
- 7  End tag for `<body>` seen, but there were unclosed elements.
- 8  Unclosed element `<div>`.

Obr. 8.15 Výstup HTML validátoru pro `utb.cz`

### 8.7.2 Lighthouse a DevTools

V dalším kroku auditu lze vyzkoušet práci s nástrojem Lighthouse. Zde je třeba nejprve otevřít požadovanou stránku v prohlížeči Google Chrome, poté otevřít panel Vývojářských nástrojů (DevTools), a v horní liště vybrat kartu Lighthouse. Po nastavení předvoleb nástroje (v tomto případě byla vybrána pouze možnost "Přístupnost"), a kliknutí na tlačítko "Analyzovat načítání stránky" se automaticky spustí audit (Obr. 8.17).



Obr. 8.16 Běžící audit Lighthouse

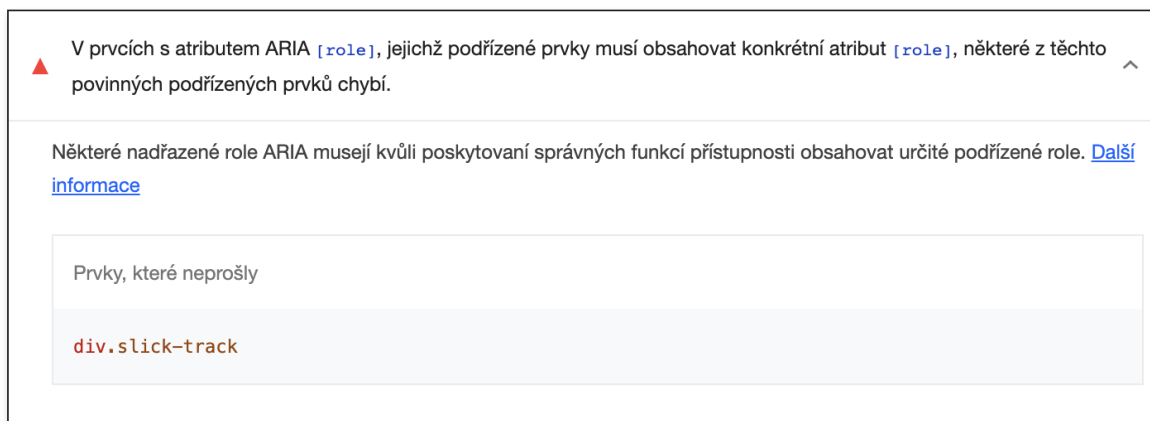
Za několik vteřin (až desítek vteřin, v závislosti na velikosti stránky) je zobrazeno hodnocení (Obr. 8.17) a list objevených pochybení.



Obr. 8.17 Hodnocení stránky

V tomto případě vyšlo hodnocení na 86 bodů (ze 100) a automatizované testy upozorňují na čtyři příležitosti ke zlepšení (Obr. 8.18, 8.21, 8.22 a 8.24).

První příležitostí je upozornění, že podřízené prvky některých prvků s aria atributy role musí rovněž obsahovat specifickou roli. (Obr. 8.18). Při následné inspekci kódu lze odhalit, že se v tomto případě jedná o roli "list", která vyžaduje aby její potomci měli roli "listitem", ale v kódu se aktuálně nachází role "none" (Obr. 8.19).



Obr. 8.18 Chybějící správná aria role

```
▼ <div role="list" class="slick-track" style="opacity: 1; width: 4914px; transform: translate3d(-702px, 0px, 0px);">
  ::before
  ▶ <div class="slick-slide slick-cloned" role="none" data-slick-index="-2" id
    aria-hidden="true" tabindex="-1" style="width: 351px;">...</div>
  ▶ <div class="slick-slide slick-cloned" role="none" data-slick-index="-1" id
    aria-hidden="true" tabindex="-1" style="width: 351px;">...</div>
```

Obr. 8.19 Chybějící správná aria role, kód

Druhou příležitostí je upozornění na překlepy v názvech aria atributů. Zde se jedná o problém zjištěný již při validaci kódu v předchozí kapitole - překlep u atributu aria-labelledby, kde chybí písmeno l (Obr. 8.21, 8.23).

```
▼ <a href="https://www.utb.cz/kalendar/brusleni-pro-studenty-a-zamestnance/"
  class="pane-calendar-item" aria-labelledby="brusleni-pro-studenty-a-zamestn
  ance">
```

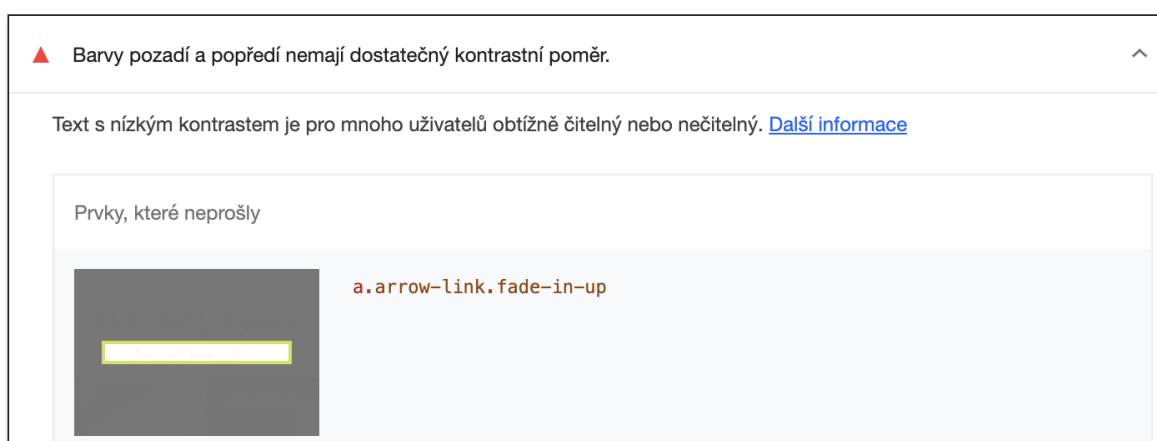
Obr. 8.20 Aria atribut s překlepem, kód



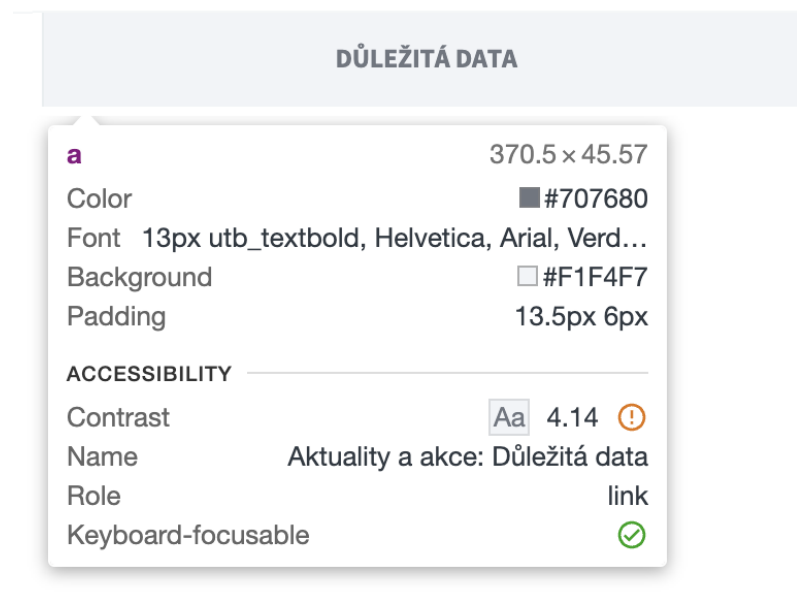
Obr. 8.21 Aria atribut s překlepem

Třetí příležitostí je upozornění na nedostatečný kontrast barev hned u několika komponent (Obr. 8.22). S pomocí funkce DevTools "prozkoumat" lze tuto skutečnost dále ověřit pro každý prvek zvlášť (accessibility - contrast - ikona vykřičníku, demonstrováno na Obr. 8.23). Zde dochází k porušení kritéria úspěchu 1.4.3 Contrast (Minimum) (Úroveň AA).

#### KONTRAST



Obr. 8.22 Nízký kontrast textu



Obr. 8.23 Nízký kontrast textu, DevTools

Poslední, čtvrtou příležitostí je upozornění na kladnou hodnotu atributu *tabindex* (Obr. 8.24, 8.25). Za normálních okolností není vhodné takové hodnoty používat, protože to vede ke změně pořadí fokusu prvků, což může být pro uživatele matoucí, a současně vést k porušení SC 2.4.3: Focus Order (Úroveň A). V tomto případě se ale jedná o výjimku, a element naopak slouží k usnadnění navigace, kdy umožňuje přeskočení bloku obsahu a přímý přechod k navigační liště (splnění SC 2.4.1: Bypass Blocks (Úroveň A)).

## NAVIGACE



Obr. 8.24 Skip link

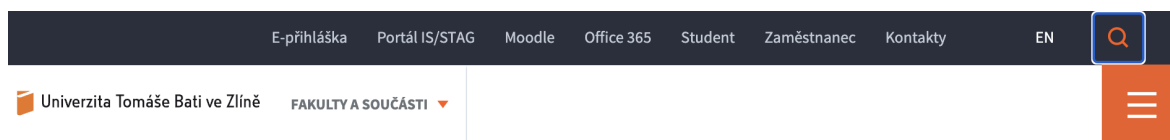
```
<a class="screen-reader-shortcut" href="#wp-toolbar" tabindex="1">Přejít k navigační liště</a>
```

Obr. 8.25 Skip link, kód

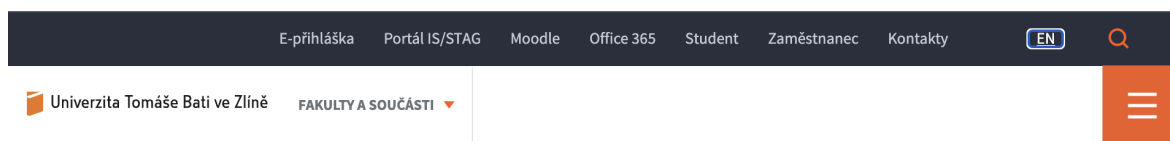


### 8.7.3 Manuální testování

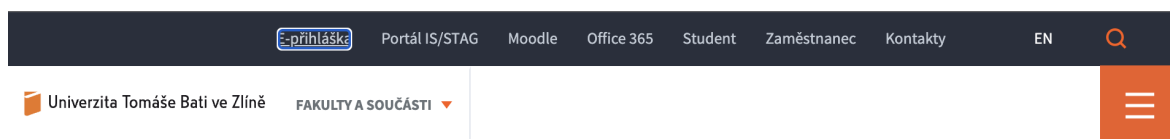
Jedním z hlavních bodů, které je vhodné testovat manuálně, je správný tok a viditelnost fokusu při navigaci klávesnicí. Zde bylo objeveno poněkud zvláštní pořadí prvků v navigační liště, kdy je nejdříve uživatel směřován úplně vpravo na vyhledávání, potom doleva na výběr jazyku, a v dalším kroku je přeskočen zbytek lišty, a fokus se objeví úplně vlevo na odkazu E-přihláška. Následně již pokračuje doprava jak by se očekávalo. Zde by se dalo argumentovat, že toto pořadí může být v rozporu s SC 2.4.3: Focus Order (Úroveň A). Toto chování je zachyceno a popsáno na obrázcích 8.26, 8.27, 8.28 a 8.29 níže.



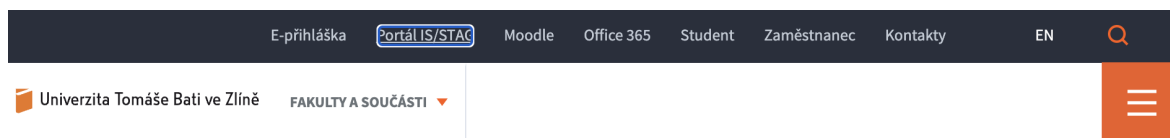
Obr. 8.26 Fokus po prvním stisku TAB



Obr. 8.27 Fokus po druhém stisku TAB

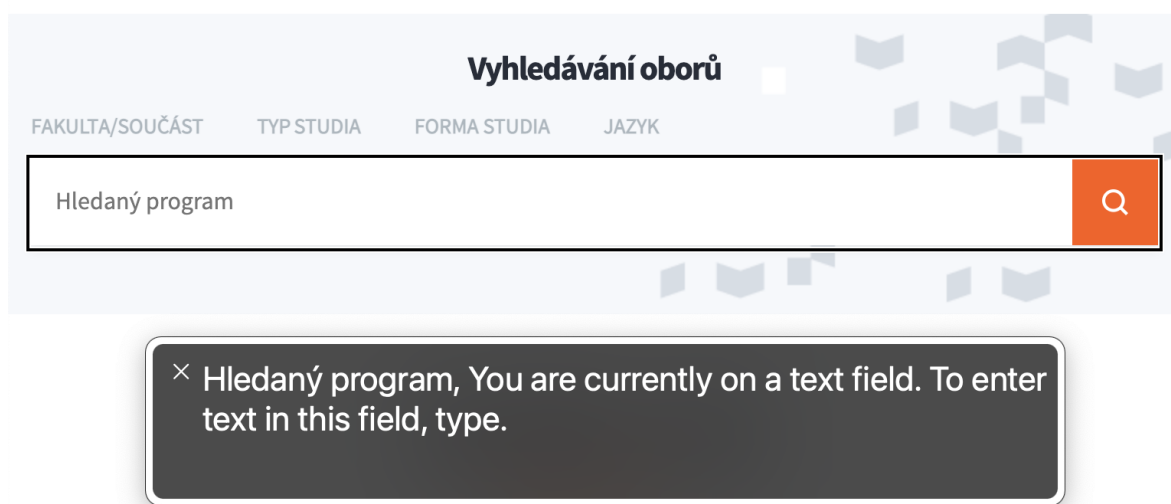


Obr. 8.28 Fokus po třetím stisku TAB

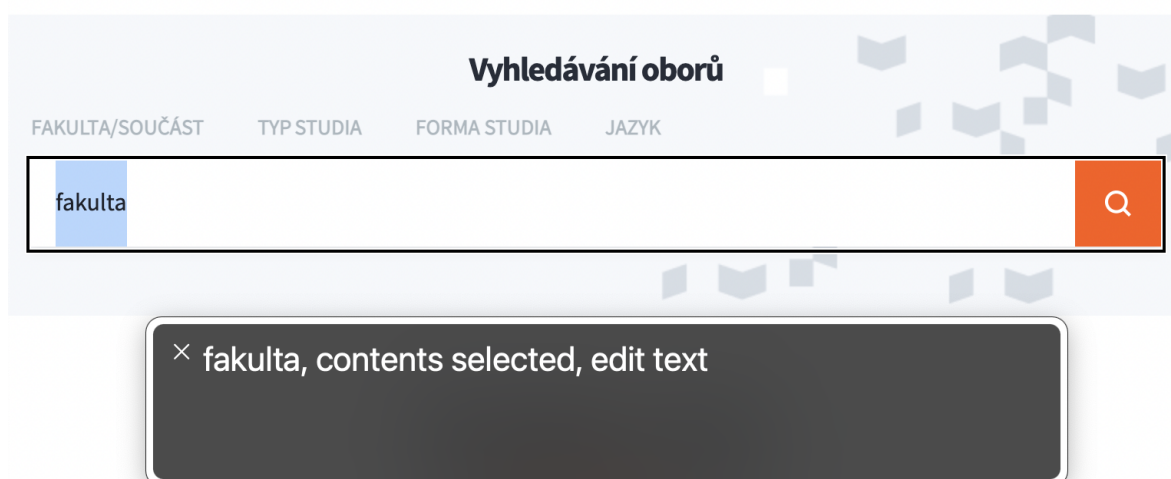


Obr. 8.29 Fokus po čtvrtém stisku TAB

Další příležitostí k vylepšení přístupnosti je nepoužívání placeholderu coby názvu vstupního pole. Zde částečně dochází k porušení SC 2.4.3: Role, Value (Úroveň A), protože ne veškerá AT vnímá placeholder jako název vstupních polí. I v případě, kdy tomu tak je, text placeholderu zmizí, jakmile pole obsahuje nějaký uživatelský text. V tento moment potom informace o názvu (a účelu) pole není dostupná. Toto chování je demonstrováno na obrázcích 8.30 a 8.31, kdy čtečka obrazovky VoiceOver ve výchozím stavu oznámí název pole správně odvozený z hodnoty placeholderu (Obr. 8.30), ale v momentě, kdy je do pole zadán nějaký text, už tento název uživateli oznámen není (Obr. 8.31).



Obr. 8.30 Vyhledávací pole, VoiceOver



Obr. 8.31 Vyhledávací pole, VoiceOver, obsahující text a placeholder

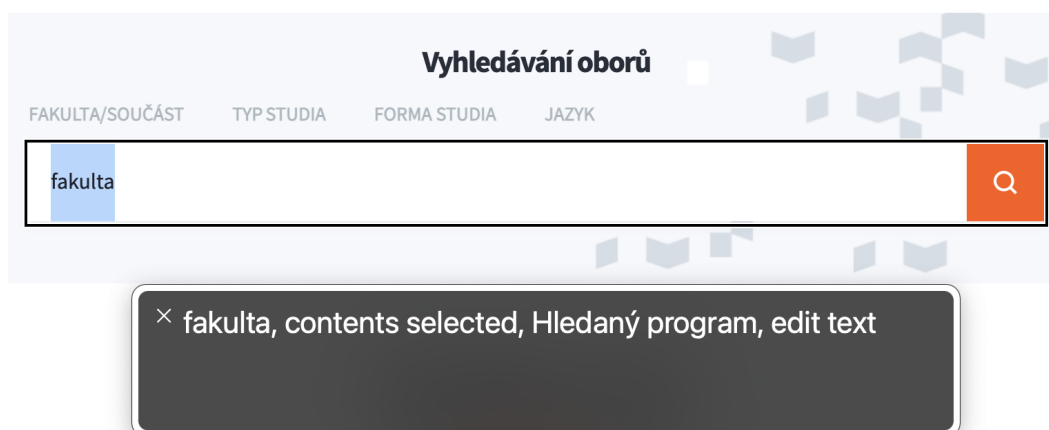
Nejlepším řešením by bylo využití nativního elementu `<label>`, a jeho propojení s polem pomocí atributů `for` a `id`. Předpokládá se, že v tomto případě se jedná o kom-

ponentu na míru, a toto řešení není dostupné. Nabízí se tak několik dalších možností zachování přístupnosti za použití aria atributů. Jednou z nich by mohlo být užití atributu *aria-labelledby*, kterému by byla přiřazena hodnota atributu *id* z elementu nadpisu komponenty "Vyhledávání oborů". V tomto případě by pak VoiceOver oznamoval při vybrání pole právě tento text.

Druhou variantou (demonstrovanou níže) je využití atributu *aria-label* na elementu vstupního pole. V tomto případě není třeba přidávat dalším prvkům *id*, ani žádné jejich propojení, ale stačí pouze poskytnout skrytý text (popisek) daného pole právě jako hodnotu atributu *aria-label*. Ta následně pro AT přepíše jméno prvku původně stanovené podle placeholderu (Obr.8.32). V tomto případě je vstupní pole uživateli správně oznámeno i v případě, že do něj již zadal nějaký text a skryl placeholder (Obr.8.33).

▼ Vypočtené vlastnosti	▼ Vypočtené vlastnosti
▼ Name: "Hledaný program" aria-labelledby: Neuvedeno aria-label: Neuvedeno Z prvku label: Neuvedeno placeholder: "Hledaný program" aria-placeholder: Neuvedeno title: Neuvedeno Role: textbox Neplatná hodnota zadaná uživatelem: false Vybratelný: true Upravitelné: plaintext Nelze nastavit hodnotu: true Víceřádkový: false Pouze pro čtení: false Povinné: false	▼ Name: "Hledaný program" aria-labelledby: Neuvedeno aria-label: "Hledaný program" Z prvku label: Neuvedeno placeholder: "Hledaný program" aria-placeholder: Neuvedeno title: Neuvedeno Role: textbox Neplatná hodnota zadaná uživatelem: false Vybratelný: true Upravitelné: plaintext Nelze nastavit hodnotu: true Víceřádkový: false Pouze pro čtení: false Povinné: false
placeholder="Hledaný program"	aria-label="Hledaný program": placeholder="Hledaný program"

Obr. 8.32 Odvození jména elementu podle aria-label



Obr. 8.33 Vyhledávací pole, VoiceOver, obsahující text, placeholder a aria-label

## 9 Proces zpřístupnění webových komponent

### 9.1 Ukázka 1

V prvním příkladu je předvedena ukázka zjednodušeného osobního portfolia. Při zobrazení HTML souboru se zdrojovým kódem v prohlížeči se výsledná stránka vizuálně podobá následujícímu obrázku (Obr. 9.1).

Diplomová práce ukázka - nepřístupná verze

#### Adam Strojil - portfolio



Jsem *webový vývojář* a zaměřuju se na **a11y**. Jsem absolvent bakalářského studia na univerzitě Tomáše Bati ve Zlíně. Studoval jsem obor Softwarové inženýrství.

#### Dovednosti

- HTML
- CSS
- Javascript
- React

#### Jazyky

Čeština C2 - nativní úroveň  
Angličtina C1 - pokročilý  
Turečtina B1 - mírně pokročilý

Můj kontakt [zde](#).

 Univerzita Tomáše Bati  
Fakulta aplikované informatiky

Obr. 9.1 Ukázka 1 - nepřístupné portfolio

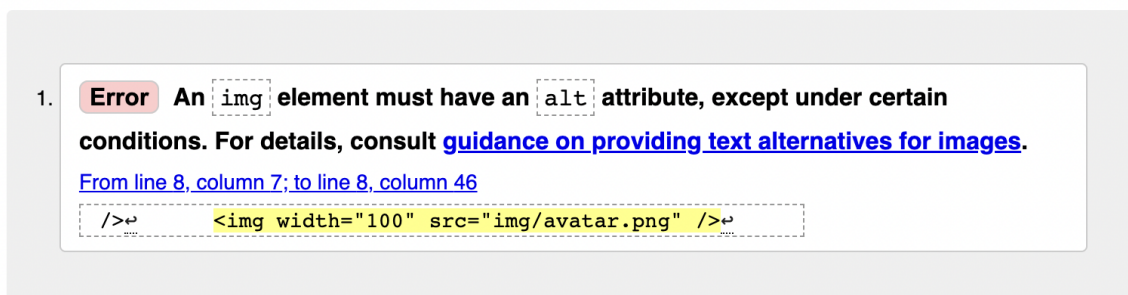
### 9.1.1 Počáteční kód

```
<div class="header">
  <br />Diplomová práce ukázka - nepřístupná verze<br /><br />
</div>
<div class="main">
  <h2>Adam Strojil - portfolio</h2>
  <br />
  
  <br />
  <div class="p">
    Jsem <i>webový vývojář</i> a zaměřuju se na <b>a1ly</b>.
    Jsem absolvent bakalářského studia na univerzitě Tomáše Bati ve
    Zlíně.<br />
    Studoval jsem obor Softwarové inženýrství.
  </div>
  <h2>Dovednosti</h2>
  - HTML<br />
  - CSS<br />
  - Javascript<br />
  - React<br /><br />
  <h2>Jazyky</h2>
  Čeština C2 - nativní úroveň<br />
  Angličtina C1 - pokročilý<br />
  Turečtina B1 - mírně pokročilý<br />
</div>
<div class="footer">
  <br />
  <div>Můj kontakt <a href="mailto:a_strojil@utb.cz">zde</a>.</div>
  <br />
  <a href="https://www.utb.cz">
    
  </a>
</div>
```

Obr. 9.2 Ukázka 1 - nepřístupné portfolio, kód

### 9.1.2 Identifikace problémů

Jedním z nejsnadnějších kroků ke zjištění potencionálních problémů je ověřit, zda-li je kód v první řadě validní. K tomuto účelu existuje celá řada nástrojů, zde je opět použit online validátor Markup Validation Service od W3C, dostupný na adrese <https://validator.w3.org/>. Po vyfiltrování výsledků od nerelevantních varování a doporučení zbyl pouze jeden problém - obrázek nemá nastaven žádný alternativní text.



Obr. 9.3 Výstup validátoru

Chybějící alternativní text je závažný problém, ale není zdaleka jediným. Bohužel automatizované testy nedokáží odhalit veškeré nedostatky, a tak je vždy na místě provést i manuální testování. V této ukázce lze totiž takto najít spoustu dalších méně či více závažných problémů.

Při inspekci zdrojového kódu lze odhalit i další nedostatky jako:

- **Nedodržení struktury nadpisů** - Je použit `<h2>` bez předchozího `<h1>`.
- **Zneužívání značky `<br>`** - Oato značka slouží k zalomení řádku, nikoliv pro vytvoření vertikálního odsazení.
- **Používání značek bez sémantického významu** - Jsou naprosto ignorovány sémantické HTML5 značky jako `<header>`, `<footer>`, `<main>`, které zde zastupují generické značky `<div>` rozlišené pomocí CSS tříd.
- **Používání HTML atributů pro stylování** - Zde se jedná o atributy `width` a `height` u značky `<img>`. Z hlediska AT je vhodnější pro tyto účely použít CSS.
- **Odkazy bez popisných názvů** - Pokud je odkaz čten samostatně lze jen těžko určit jeho cíl z popisku "zde".
- **Seznamy nevyužívající značky `<ul>`, `<ol>`, či `<dl>`** - V obsahu se nachází několik seznamů, které jsou nesprávně formátovány pomocí odrážek a značky `<br>`.
- **Používání zastaralých sémantických značek** - Existují moderní a vhodnější alternativy pro značky `<i>` a `<b>`.
- **Alternativní text obrázku není popisný** - Druhý obrázek sice obsahuje `alt` text a nebyl tak odhalen automatickými testy, ale jeho obsah je nesmyslný.

### 9.1.3 Zpřístupnění komponenty

Namísto použití obecné značky `<div>` je pro záhlaví stránky vhodnější zvolit sémantickou značku `<header>`, která uživateli či AT napoví, že se jedná právě o záhlaví, a také z této části udělá tzv. *landmark*, který může sloužit k rychlejší navigaci stránkou. Dále lze uvnitř headeru odstranit značky `<br>`, které jsou zde nesprávně použity pro vytvoření vertikálního odsazení. Toho lze lépe docílit pomocí nastavení požadovaných hodnot atributů `margin` či `padding` pro `<header>`.

Obdobně jako byla použita značka `<header>` pro záhlavní, je také doporučeno použít značku `<main>` pro hlavní obsah stránky. Značka `<main>` by měla být použita právě jednou na každé stránce. Dalším prohřeškem je nesprávné použití `<h2>` tagu. Ten je sice sémantický a říká, že se jedná o nadpis druhé úrovně, ale jednotlivé úrovně nelze přeskakovat. Na stránce se nikde nenachází nadpis úrovně 1, a tak se jím stane tento. V případě, kdy výchozí stylování nadpisů neodpovídá požadavkům (nadpisy vyšší úrovně mají větší písmo a odsazení), lze toto chování přepsat s použitím CSS. Na několik místech se opět nesprávně nachází značka nového řádku použitá ke stylování. Bude odstraněna, a efektu odsazení bude dosaženo s pomocí CSS.

Další na řadě je již zmíněný problém s obrázky, které postrádají alternativní text. I zde je náprava snadná, stačí pouze tento text dodat pomocí HTML atributu `alt`. Další drobností je použití HTML atributu `width`. Ačkoliv se nejedná o závažný problém, je lepší pro tyto vlastnosti používat CSS. Může se stát, že takové atributy budou matoucí pro asistenční technologie, například čtečky obrazovky.

Pro následující popis je lepší využít sémantickou značku `<p>` (odstavec), namísto `<div>`. Uvnitř textu se nachází dvě zvýrazněné fráze, jedna ve stylu italic a druhá tučně. V obou případech jsou použity nesémantické a zastaralé HTML značky, které sice vizuálně odvedou stejnou práci jako jejich nástupci v HTML5, ale budou ignorovány asistenčními technologiemi. Z důvodu zachování zdůraznění částí textu i uživatelům, kteří jej nemůžou přímo vidět, jsou tyto značky nahrazeny za `<em>` namísto `<i>` a `<strong>` namísto `<b>`.

U podnadpisu "Dovednosti" (a později i "Jazyky") je nyní použito značky `<h2>`. Výčet dovedností, který následuje, lze vhodněji vyjádřit jako list `<ul>` a jeho položky `<li>`. Podobně lze také vylepšit výčet jazyků, s malým rozdílem, že zde bude použit list definicí s pomocí značek `<dl>`, `<dt>` a `<dd>`.

V záhlaví stránky je vhodné nahradit `<div>` sémantickým `<footer>`. Používání těchto elementů je důležité pro správné navigování napříč stránkami při použití AT. Posledním vylepšením v rámci této ukázky je poskytnutí smysluplných popisků pro hypertextové odkazy. Zde se nabízí několik možností, jak tohoto dosáhnout. Nejlepší z nich je použití dostatečně popisného textu uvnitř odkazu. Lze tedy změnit text od-

kazu ze "zde" na například "a\_strojil@utb.cz" nebo "pošlete mi email". Může se ale stát, že tento přístup aplikovat nelze, jako třeba na dalších řádcích, kdy je jako odkaz použit obrázek. V takovém případě asistenční technologie použije jako název pro odkaz alternativní text daného obrázku. Alternativním řešením by rovněž bylo použití aria-label atributu na <a> značce, pomocí kterého lze AT říct, pod jakým jménem má daný element prezentovat.

#### 9.1.4 Refaktorovaný kód

```
<header>Diplomová práce ukázka - přístupná verze</header>
<main>
  <h1 class="h2">Adam Strojil - portfolio</h1>
  
  <p>
    Jsem <em>webový vývojář</em> a zaměřuju se na
    <strong>aally</strong> (accessibility). Jsem absolvent bakalářského
    studia na univerzitě Tomáše Bati ve Zlíně.<br />
    Studoval jsem obor Softwarové inženýrství.
  </p>
  <h2>Dovednosti</h2>
  <ul>
    <li>HTML</li>
    <li>CSS</li>
    <li>Javascript</li>
    <li>React</li>
  </ul>
  <h2>Jazyky</h2>
  <dl>
    <div>
      <dt>Čeština C2</dt>
      <dd>nativní úroveň</dd>
    </div>
    <div>
      <dt>Angličtina C1</dt>
      <dd>středně pokročilý</dd>
    </div>
    <div>
      <dt>Turečtina B1</dt>
      <dd>mírně pokročilý</dd>
    </div>
  </dl>
</main>
<footer>
  <div style="margin: 16px 0">
    Můj kontakt <a href="mailto:a_strojil@utb.cz">a_strojil@utb.cz</a>.
  </div>
  <a href="https://www.utb.cz" aria-label="Univerzita Tomáše Bati">
    
  </a>
</footer>
```

Obr. 9.4 Ukázka 1 - přístupné portfolio, kód



Diplomová práce ukázka - přístupná verze

## Adam Strojil - portfolio



Jsem *webový vývojář* a zaměřuju se na **a11y** (accessibility). Jsem absolvent bakalářského studia na univerzitě Tomáše Bati ve Zlíně. Studoval jsem obor Softwarové inženýrství.

### Dovednosti

- HTML
- CSS
- Javascript
- React

### Jazyky

**Čeština C2**   nativní úroveň  
**Angličtina C1**   středně pokročilý  
**Turečtina B1**   mírně pokročilý

Můj kontakt [a\\_strojil@utb.cz](mailto:a_strojil@utb.cz).

 Univerzita Tomáše Bati  
Fakulta aplikované informatiky

Obr. 9.5 Ukázka 1 - přístupné portfolio

## 9.2 Ukázka 2

Druhá ukázka se zabývá příkladem jednoduchého webového formuláře, a demonstruje nejčastější pochybení právě v této oblasti. V tomto případě si lze představit, že se jedná o formulář určený pro podání přihlášky uchazeče o zaměstnání či členství v klubu programátorů. V první verzi vypadá v prohlížeči formulář při fokusování druhého pole jako na Obr. 9.6.

# Přihláška

## Základní údaje

Jméno:

Email:

## Ostatní

Počet let zkušeností s vývojem:

[Zpět](#)

[Odeslat →](#)

Obr. 9.6 Ukázka 2 - nepřístupný formulář

Ukázka obsahuje nadpis a jeden formulář rozdělený do dvou sekcí - "Základní údaje" a "Ostatní". Celkem obsahuje dvě povinná pole a jedno volitelné. Dále obsahuje odkaz pro návrat na předchozí stránku a tlačítko sloužící k odeslání vyplněných dat.

### 9.2.1 Počáteční kód

Druhá ukázka (Obr. 9.6) využívá níže prezentovaný kód (Obr. 9.7). Vizualní styl komponent je kvůli přehlednosti řešen pomocí CSS v samostatném souboru. Protože se jedná pouze o demonstraci konceptu, a většina stylů s přístupností této ukázky nesouvisí, je tento kód pro účely práce vynechán. Změny v přístupnosti ovlivněny pomocí CSS budou označeny přímo v textu.

```
<h1>Přihláška</h1>
<form action="#">
  <div>
    <h3>Základní údaje</h3>
    <div>
      <span>Jméno:</span>
      <input required id="name" />
    </div>
    <div>
      <p>Email:</p>
      <input required id="email" />
    </div>
  </div>
  <h3>ostatní</h3>
  <div>
    <span>Počet let zkušeností s vývojem:</span>
    <input id="years" />
  </div>
  <div>
    <a href="#">Zpět</a>
    <button>odeslat →</button>
  </div>
</form>
```

Obr. 9.7 Ukázka 2 - nepřístupný formulář, kód

### 9.2.2 Identifikace problémů

Pokud je výše zmíněný kód opět testován ve validátoru nejsou tentokrát nalezeny žádné chyby, a lze jej tedy z hlediska HTML považovat za validní. Z hlediska webové přístupnosti ale obsahuje velké množství prohřešků, z nichž většina není na první pohled patrná. Lze je ale odhalit manuální inspekcí kódu. Mezi nejzávažnější problémy v tomto případě patří:

- **Vstupní pole (input) nemají přiřazeny popisky (label)** - Vizualně se zdá být vše v pořádku, ale na úrovni kódu nejsou popisky a pole nijak provázány. Navíc ani nejsou využity sémantické elementy `<label>` kdy na místo nich jsou jen generické `<span>`, či špatně použitý element `<p>`.
- **Přeskakování úrovně nadpisů** - "Přihláška" je nadpis úrovně 1 a "Základní údaje" nadpis úrovně 3. Tímto řešením sice lze jednoduše dosáhnout požadova-

ného vizuálního stylu (menší velikost písma), ale není sémanticky správné. Vizuální stylování by mělo být řešeno za pomoci CSS.

- **Seskupování polí pomocí ne-sémantických elementů** - Pro vytvoření skupin "Základní údaje" a "Ostatní" jsou využity nadpisy. Vhodnější volbou by byly sémantické elementy `<fieldset>` a `<legend>`.
- **Vstupní pole nemají definován typ** - I pro pole, která očekávají jiný formát dat než jen prostý text (email, číslo) je tedy použit výchozí typ "text".
- **Modrá barva není dostatečně kontrastní** - Při jejím ověřování pomocí Google DevTools (popsáno v teoretické části) lze zjistit, že kontrast dosahuje hodnoty 3.97. Pro splnění urovně AA je ale požadován kontrast nejméně 4.5 a pro AAA je to dokonce 7.0.
- **Povinná pole nejsou označena** - Chybí jakékoliv vizuální označení povinných polí. Uživatel tak neví která pole opravdu musí vyplnit, dokud se nepokusí formulář odeslat, a nedostane se do chybového stavu.
- **Odkaz je označen pouze barvou** - Pro označení funkcionality dle WCAG nestačí pouze použít jinou barvu. Je třeba poskytnout další vizuální odlišení, například podtržení.
- **Fokus není viditelný** - I přesto, že je fokusováno druhé pole formuláře (Email), ukazatel fokusu není viditelný.

Mezi méně závažné prohřešky a doporučení patří:

- **Tlačítko nemá specifikován typ** - Použije se sice výchozí typ `submit`, což je v tomto případě v pořádku, ale v jiných případech to může vést k neočekávaným chybám. Je vhodné vždy typ specifikovat.
- **Chybí placeholder text** - Je vhodné poskytnout ukázkou požadovaného formátu dat pomocí placeholderu. Samotný placeholder ale nikdy nesmí sloužit jako jediný popis pole. I na tento text se vztahuje kritérium minimálního kontrastu barev.
- **Chybí jméno formuláře** - Formulář může být pojmenován s pomocí propojení existujícího nadpisu přes `aria-labelledby` atribut.
- **Označení povinných polí pro AT** - Povinná pole lze dále označit s pomocí `aria-required`.

- **Zarovnání a pořadí tlačítek** - Doporučuje se tlačítka zarovnat v levo (pro jazyky kde je směr čtení z leva do prava) a seřadit tak, že na začátku jsou ta, která obsluhují primární akci. Takový design pomůže v orientaci například uživatelům, kteří používají nástroj Lupa pro zvětšení výřezu obrazovky.
- **Chybí specifikace obsahu polí** - S pomocí atributu *autocomplete* lze prohlížeči usnadnit automatické (před)vyplňování polí.
- **Nadbytečné užívání `<div>` kontejnerů** - Popisek a pole jsou obaleny v elementu `<div>`, který slouží pouze pro vizuální zalomení řádku. Stejněho vzhledu lze docílit pomocí CSS stylů. Je vhodné usilovat o co nejstručnější a nejjednodušší strukturu DOM-u a dosáhnout tak přehlednějšího HTML kódu.
- **Správné jednotky pro velikost textu** - Je vhodné se ujistit, že v CSS jsou použity pro definici velikosti textu responzivní jednotky jako *em* či *rem*, namísto pixelů. To umožní AT a prohlížečům správné škálování obsahu.

### 9.2.3 Zpřístupnění komponenty

Níže jsou definovány kroky, které lze podniknout pro zlepšení přístupnosti, čitelnosti a udržitelnosti kódu formuláře z ukázky 2 (Obr. 9.7). Jejich pořadí odpovídá výskytu v kódu odshora dolů.

- Propojení `<h1>` a `<form>` s pomocí *id* a *aria-labelledby*.
- Použití elementů `<fieldset>` a `<legend>` pro seskupování částí formuláře.
- Použití elementů `<label>`, a jejich správné propojení se vstupním polem pomocí atributů *for* a *id* (demonstrováno u pole "Jméno"), nebo za pomoci vnořování (demonstrováno u pole "Email").
- Viditelné označení povinných polí s pomocí slova "povinné".
- Specifikování typu polí i tlačítek s pomocí atributy *type*.
- Poskytnutí *placeholder* textu s ukázkami vstupních dat.
- Specifikace účelu polí v atributu *autocomplete*.
- Přidání aria atributu *aria-required="true"* pro povinná pole.
- Změna modré barvy na tmavší, více kontrastní odstín (pomocí CSS, atributy *color* a *background-color*).
- Změna pořadí a zarovnání tlačítek (pomocí CSS).

- Odkaz je označen podtržením (pomocí CSS, atribut *text-decoration*).
- Fokus polí není skrytý (pomocí CSS, atribut *outline*).

U třetího pole (Počet let zkušeností. . .) je navíc demonstrována možnost propojení vstupního pole a popisku za pomoci aria atributu *aria-labelledby*, které rovněž splňuje požadavky přístupnosti, ale protože aria nemění funkcionalitu prvků a pouze poskytuje další informace pro AT ztrácí uživatel při tomto řešení nativní funkcionalitu přesunu fokusu na pole při kliknutí na popisek, kterou většina prohlížečů nabízí. Tento přístup by tedy měl být použit až jako poslední možnost.

Konečný formulář po všech opravách poté vypadá jako na (Obr. 9.8) níže. I zde má opět fokus pole Email.

## Přihláška

### Základní údaje

Jméno (povinné):

Email (povinné):

### Ostatní

Počet let zkušeností s vývojem:

[Zpět](#)

Obr. 9.8 Ukázka 2 - přístupný formulář

### 9.2.4 Refaktorovaný kód

Na obrázku níže (Obr. 9.9) lze vidět původní kód z Obr. 9.7 po provedení všech zmíněných úprav.

```
<h1 id="form-title">Přihláška</h1>
<form aria-labelledby="form-title" action="#">
  <fieldset>
    <legend>Základní údaje</legend>
    <label for="name">Jméno (povinné): </label>
    <input
      required
      id="name"
      type="text"
      placeholder="Adam Strojil"
      autocomplete="name"
      aria-required="true"
    />
  <div>
    <label>
      Email (povinné):
      <input
        required
        id="email"
        type="email"
        placeholder="a_strojil@utb.cz"
        autocomplete="email"
        aria-required="true"
      />
    </label>
  </div>
</fieldset>
<fieldset>
  <legend>Ostatní</legend>
  <label id="yearsLabel">Počet let zkušeností s vývojem: </label>
  <input
    id="years"
    type="number"
    placeholder="3"
    aria-labelledby="yearsLabel"
  />
</fieldset>
<button type="submit">Odeslat →</button>
<a href="#">Zpět</a>
</form>
```

Obr. 9.9 Ukázka 2 - přístupný formulář, kód

## ZÁVĚR

Napříč prací bylo představeno značné množství pravidel, pokynů, požadavků, nástrojů, technik a ukázek, demonstrujících význam webové přístupnosti a následné vytvoření (či vylepšení) webových komponent, které tato doporučení berou v potaz a jsou přístupné.

Přístupnost webu je uznávána celosvětově a mnoho zemí přijalo zákony a předpisy, které vyžadují, aby webové stránky splňovaly specifické standardy přístupnosti, nejčastěji WCAG 2.1 úrovně AA.

Kromě rostoucího počtu uživatelů webu se zdravotním postižením, se stala webová přístupnost důležitým tématem i kvůli šíření nových technologií a zařízení s různými velikostmi obrazovky, rozlišením a metodami zadávání vstupních dat. Tato rozmanitost zařízení může vytvářet pro uživatele překážky, a ti mohou mít s interakcí s webovým obsahem problémy.

Dalším faktorem, který zdůrazňuje důležitost přístupnosti webu, je ekonomický dopad, který mohou mít nedostupné webové aplikace na podniky. Nepřístupné webové stránky mohou vést ke ztrátě příjmů, zvýšeným nákladům na zákaznickou podporu a negativní pověsti značky. Přístupnost webu proto není jen morální a zákonná povinnost, ale svým způsobem také obchodní benefit.

Závěrem lze říci, že implementace přístupných webů je zásadním krokem k vytvoření inkluzivního online prostředí, které všem uživatelům umožní snadný přístup k informacím a službám, bez ohledu na jejich fyzické, smyslové nebo kognitivní schopnosti, nebo na hardware, který používají. Pro webové vývojáře, designéry a testery je proto nezbytné, aby při vytváření a údržbě webových aplikací pamatovali i na tuto skupinu uživatelů a vyvíjeli své aplikace tak, aby splňovaly alespoň základní pravidla webové přístupnosti.



## SEZNAM POUŽITÉ LITERATURY

- [1] *Global Estimates On Visual Impairment* [online]. [cit. 2022-02-05]. Dostupné z WWW: <https://www.emro.who.int/control-and-preventions-of-blindness-and-deafness/announcements/global-estimates-on-visual-impairment.html>.
- [2] *Colorblind People Population Live Counter* [online]. [cit. 2022-02-05]. Dostupné z WWW: <https://www.colorblindguide.com/post/colorblind-people-population-live-counter>.
- [3] *Use Of Color* [online]. [cit. 2022-02-05]. Dostupné z WWW: <https://www.w3.org/WAI/WCAG21/Understanding/use-of-color.html>.
- [4] *Hendikepovaní uživatelé* [online]. [cit. 2022-02-05]. Dostupné z WWW: <http://www.pristupnost.cz/o-pristupnosti/hendikepovani-uzivatele/>.
- [5] *Jouse 3 Joystick* [online]. Adaptive Specialties [cit. 2022-02-05]. Dostupné z WWW: <https://www.adaptivespecialties.com/jouse-3-mouth-controlled-usb-joystick-by-compusult.aspx>.
- [6] *Digital Education Strategies, The Chang School. GAY, Greg & KARASYOV, Igor. Web Accessibility for Developers.* November 29, 2019.
- [7] *WAI About.* [online]. [cit. 2022-02-17]. Dostupné z WWW: <https://www.w3.org/WAI/about/>.
- [8] *WAI-ARIA Overview.* [online]. [cit. 2022-02-17]. Dostupné z WWW: <https://www.w3.org/WAI/standards-guidelines/aria/>.
- [9] *WCAG intro.* [online]. [cit. 2022-02-17]. Dostupné z WWW: <https://www.w3.org/WAI/standards-guidelines/wcag/#intro>.
- [10] *HARPER, Simon & YESILADA, Yeliz. Web Accessibility: A Foundation for Research. Web Accessibility and Guidelines.* 2008. ISBN: 978-1-84800-049-0
- [11] *Laws And Policies* [online]. [cit. 2022-02-24]. Dostupné z WWW: <https://www.section508.gov/manage/laws-and-policies/>.
- [12] *Web Content Accessibility Guidelines 1.0.* [online]. W3C 1999. [cit. 2022-02-24]. Dostupné z WWW: <https://www.w3.org/TR/WAI-WEBCONTENT/>.
- [13] *Web Content Accessibility Guidelines (WCAG) 2.0.* [online]. W3C 2008. [cit. 2022-02-24]. Dostupné z WWW: <https://www.w3.org/TR/WCAG20/>.

- [14] *Web Content Accessibility Guidelines (WCAG) 2.1*. [online]. W3C 2008. [cit. 2022-02-24]. Dostupné z WWW: <https://www.w3.org/TR/WCAG21>.
- [15] *Web Content Accessibility Guidelines (WCAG) 2.2*. [online]. W3C 2008. [cit. 2022-02-24]. Dostupné z WWW: <https://www.w3.org/TR/WCAG22>.
- [16] *Layers of guidance* [online]. W3C 2008. [cit. 2022-02-24]. Dostupné z WWW: <https://www.w3.org/TR/WCAG21/#wcag-2-layers-of-guidance>.
- [17] *Přístupnost internetových stránek a mobilních aplikací*. [online]. Ministerstvo vnitra České republiky 2019. [cit. 2022-02-27]. Dostupné z WWW: <https://www.mvcr.cz/clanek/pristupnost-internetovych-stranek-a-mobilnich-aplikaci.aspx>.
- [18] *Zákon č. 99/2019 Sb. Zákon o přístupnosti internetových stránek a mobilních aplikací*. In: *Zákony pro lidi* [online]. AION CS, s.r.o. 2010-2023 [cit. 2022-02-27]. Dostupné z WWW: <https://www.zakonyprolidi.cz/cs/2019-99>.
- [19] *What are wcag A, AA and AAA* [online]. USERWAY [cit. 2022-02-27]. Dostupné z WWW: <https://userway.org/blog/what-are-wcag-2-0-a-aa-and-aaa/>.
- [20] *HTML5 New Elements* [online]. W3Schools 2023. [cit. 2022-02-28]. Dostupné z WWW: <https://www.w3schools.blog/html5-new-elements>.
- [21] *HTML vs HTML5* [online]. Javatpoint [cit. 2022-02-28]. Dostupné z WWW: <https://www.javatpoint.com/html-vs-html5>.
- [22] *ARIA Attributes* [online]. Mozilla 2023. [cit. 2022-02-28]. Dostupné z WWW: <https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Attributes>.
- [23] *ARIA 1.1* [online]. W3C 2008. [cit. 2022-03-01]. Dostupné z WWW: <https://www.w3.org/TR/wai-aria-1.1/>.
- [24] *Lighthouse - Accessibility* [online]. Google 2023 [cit. 2022-03-01]. Dostupné z WWW: <https://developer.chrome.com/docs/lighthouse/accessibility/>.
- [25] *Brailliant bi 40x braille display* [online]. [cit. 2022-03-01]. Dostupné z WWW: <https://store.humanware.com/hus/brailiant-bi-40x-braille-display.html>.
- [26] *JavaScript Bookmarklets for Accessibility Testing* [online]. [cit. 2022-03-01]. Dostupné z WWW: <https://pauljadam.com/bookmarklets/index.html>.

**SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK**

AG	Accessibility guidelines
ARIA	Accessible Rich Internet Applications
AT	Assistive technology
A11Y	Accessibility
BITV	Barrierefreie Informationstechnik-Verordnung
BGG	Bundesbehindertengleichstellungsgesetz
CSS	Cascading Style Sheet
DOM	Document Object Model
EU	European Union
HTML	Hyper Text Markup Language
JS	JavaScript
PWA	Progressive Web Application
SC	Success criterion
SEO	Search Engine Optimization
UI	User Interface
URL	Uniform Resource Locator
UX	User Experience
W3C	World Wide Web Consortium
WAD	Web Accesibility Directive
WAI	Web Accessibility Initiative
WCAG	Web Content Accessibility Guidelines

## SEZNAM OBRÁZKŮ

1.1	Normální barvocit, protanopie, deuteranopie a tritanopie . . . . .	12
1.2	Výběr barvy - normální barvocit vs. deuteranopie . . . . .	13
1.3	Přístupná komponenta - normální barvocit vs. deuteranopie, verze 1	13
1.4	Přístupná komponenta - normální barvocit vs. deuteranopie, verze 2	14
1.5	Původní odkaz v textu, normální . . . . .	14
1.6	Původní odkaz v textu, deuteranopie . . . . .	14
1.7	Přístupný odkaz v textu, normální . . . . .	15
1.8	Přístupný odkaz v textu, deuteranopie . . . . .	15
1.9	Textový přepis audio rozhovoru na webu Seznam Zprávy . . . . .	15
1.10	Jouse 3 Joystick ovládaný ústy od Compusult [5] . . . . .	17
3.1	Struktura WCAG . . . . .	20
4.1	Část prohlášení o přístupnosti na utb.cz . . . . .	31
5.1	Sémantické značky v HTML5 [21] . . . . .	33
7.1	Nepřístupné vstupní pole . . . . .	39
7.2	Přístupné vstupní pole . . . . .	39
7.3	Uzamčení fokusu v dialogovém okně [25] . . . . .	40
8.1	Braille displej Brailliant BI 40X [25] . . . . .	42
8.2	Ukázka výstupu VoiceOver . . . . .	43
8.3	DOM strom vs. A11Y strom . . . . .	45
8.4	Vlastnosti prvku v panelu přístupnosti . . . . .	46
8.5	Odhalení nedostatečného kontrastu s pomocí DevTools . . . . .	46
8.6	Kontrast text vs. pozadí . . . . .	47
8.7	Kontrast odkaz vs. text vs. pozadí . . . . .	48
8.8	Výběr kontrastních barev s pomocí DevTools . . . . .	48
8.9	Let's get color blind - normální barvy . . . . .	49
8.10	Let's get color blind - deuteranopie . . . . .	49
8.11	Let's get color blind - Tritanopie . . . . .	49
8.12	Nevalidní kód připravený k ověření . . . . .	50
8.13	Výstup validátoru - výčet chyb . . . . .	50
8.14	Bookmarklet z pauljadam.com pro kontrolu hodnoty tabindex [26] .	51
8.15	Výstup HTML validátoru pro utb.cz . . . . .	52
8.16	Běžící audit Lightouse . . . . .	53
8.17	Hodnocení stránky . . . . .	53
8.18	Chybějící správná aria role . . . . .	54
8.19	Chybějící správná aria role, kód . . . . .	54
8.20	Aria atribut s překlepem, kód . . . . .	54

---

8.21	Aria atribut s překlepem . . . . .	55
8.22	Nízký kontrast textu . . . . .	55
8.23	Nízký kontrast textu, DevTools . . . . .	56
8.24	Skip link . . . . .	56
8.25	Skip link, kód . . . . .	56
8.26	Fokus po prvním stisku TAB . . . . .	57
8.27	Fokus po druhém stisku TAB . . . . .	57
8.28	Fokus po třetím stisku TAB . . . . .	57
8.29	Fokus po čtvrtém stisku TAB . . . . .	57
8.30	Vyhledávací pole, VoiceOver . . . . .	58
8.31	Vyhledávací pole, VoiceOver, obsahující text a placeholder . . . . .	58
8.32	Odvození jména elementu podle aria-label . . . . .	59
8.33	Vyhledávací pole, VoiceOver, obsahující text, placeholder a aria-label . . . . .	59
9.1	Ukázka 1 - nepřístupné portfolio . . . . .	60
9.2	Ukázka 1 - nepřístupné portfolio, kód . . . . .	61
9.3	Výstup validátoru . . . . .	62
9.4	Ukázka 1 - přístupné portfolio, kód . . . . .	64
9.5	Ukázka 1 - přístupné portfolio . . . . .	65
9.6	Ukázka 2 - nepřístupný formulář . . . . .	66
9.7	Ukázka 2 - nepřístupný formulář, kód . . . . .	67
9.8	Ukázka 2 - přístupný formulář . . . . .	70
9.9	Ukázka 2 - přístupný formulář, kód . . . . .	71

## SEZNAM PŘÍLOH

P I. CD s elektronickou verzí práce

## **PŘÍLOHA P I. CD S ELEKTRONICKOU VERZÍ PRÁCE**

Tato příloha obsahuje elektronickou PDF/A verzi práce.