

Gymnázium, Praha 6, Arabská 14

Předmět Programování



ROČNÍKOVÝ PROJEKT

Vliv implementace prvků přístupnosti do webu a
knihovna s nadstandardními funkcemi

Autor: Filip Beneš

Ročník: IV.E 2023/24

Škola: Gymnázium, Praha 6, Arabská 14

Konzultant: Mgr. Jan Lána

2. dubna 2024 Praha

PROHLÁŠENÍ

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené. Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

V Praze dne 2. dubna 2024 Filip Beneš

ANOTACE

Tato ročníková práce se zaměřuje na implementaci prvků přístupnosti do webových aplikací v jazyce Javascript a frameworku VueJS, s primárním cílem vytvořit knihovnu s nadstandardními funkcemi, včetně Text to Speech a Speech to Text. Cílem je usnadnit nevidomým uživatelům pohyb a interakci na webových stránkách, přičemž analýza technologických možností bude klíčovým prvkem.

Výsledná knihovna představuje inovativní nástroj pro vývojáře, kteří chtějí vylepšit přístupnost svých webových projektů. Zohledněním specifických potřeb nevidomých uživatelů a využitím moderních technologií přispěje k rozvoji inkluzivního designu na internetu.

KLÍČOVÁ SLOVA

Lidé se zrakovým postižením, knihovna, TTS (text to speech), STT (speech to text), FS (Focus speaker)

ANNOTATION

This Project focuses on the implementation of accessibility features in web applications in Javascript and the VueJS framework, with the primary goal of creating a library with superior features, including Text to Speech and Speech to Text. The goal is to make it easier for blind users to navigate and interact on web pages, with analysis of technological options being a key element.

The resulting library is an innovative tool for developers who want to improve the accessibility of their web projects. By taking into account the specific needs of blind users and using modern technologies, it will contribute to the development of inclusive design on the Internet.

KEYWORDS

People with visual impairment, library, TTS (text to speech), STT (speech to text), FS (Focus speaker)

OBSAH

1. ZADÁNÍ.....	5
2. ÚVOD.....	6
3. DEFINOVANÉ STANDARDY PODLE W3C WAI	7
3.1 WCAG – WEB CONTENT ACCESSIBILITY GUIDELINES.....	7
3.2 ATAG – AUTHORIZING TOOL ACCESSIBILITY GUIDELINES	7
3.3 UAAG – USER AGENT ACCESSIBILITY GUIDELINES	8
4. UŽIVATELSKÁ ČÁST	9
4.1 ŘEČNÍK	9
4.2 HLASOVÝ NAVIGÁTOR	10
4.3 SPEECH FOCUS	10
5. ADMINISTRÁTORSKÁ ČÁST	12
5.1 INSTALACE KNIHOVNY	12
5.2 NASTAVENÍ A SPUŠTĚNÍ KNIHOVNY	12
5.3 DALŠÍ FUNKCE A MOŽNOSTI KNIHOVNY	14
6. KNIHOVNA	15
6.1 ZALOŽENÍ KNIHOVNY	15
6.2 DEMONSTRAČNÍ WEB.....	17
6.3 IMPLEMENTACE NA REÁLNÝ PROJEKT	18
6.4 SPOLUPRÁCE SE ŠKOLOU PRO ZRAKOVĚ POSTIŽENÉ	18
7. POUŽITÉ TECHNOLOGIE	19
7.1 VUE.JS	19
7.2 SCSS	19
8. ZÁVĚR.....	20
9. POUŽITÉ ZDROJE	21
10. SEZNAM OBRÁZKŮ	21

1. ZADÁNÍ

Téma projektu: Vliv implementace prvků přístupnosti do webu a knihovna s nadstandardními funkcemi

Jméno a příjmení: Filip Beneš

Popis projektu: Tato ročníková práce se zaměřuje na implementaci prvků přístupnosti do webu, správného plánování UI/UX a jejich následného vlivu na spektrum uživatelů webu – tedy uživatele, kteří prvky potřebují a uživatele, kteří je mohou využívat nepodmínečně. Skládá se tedy ze dvou komplementárních částí:

- **Implementace prvků přístupnosti**
tyto prvky jsou většinou standardem W3C WAI. Patří mezi ně například alternativní popisky obrázků (alt), ovládání pomocí klávesnice, přepis audia nebo barevné módy. Cílem je spojit se s nějakou organizací sdružující například zrakově postižené a využít jejich dalších podnětů na základě zpětné vazby.
- **Knihovna s nadstandardními funkcemi**
další funkce, které nepatří mezi ty standardní, mohou být obzvláště přínosné i pro běžné uživatele a zároveň je možné je implementovat obecně – nezávisle na webu, je cílem seskupit do knihovny. Tato knihovna by pak tyto funkce poskytovala s návodem na jejich použití. Mezi tyto funkce patří například TTS (Text To Speech) nebo ovládání pomocí hlasu.

Platforma: Javascript, VueJS, HTML, SCSS, CSS

2. ÚVOD

Rychlý technologický pokrok v oblasti webových aplikací s sebou přináší nejen nové možnosti, ale také zvýšenou odpovědnost za zajištění přístupnosti pro všechny uživatele bez výjimek. Tato ročníková práce se zabývá analýzou vlivu implementace prvků přístupnosti do webových aplikací, s důrazem na použití programovacího jazyka JavaScript. Zároveň zkoumá význam a přínosy knihovny s nadstandardními funkcemi pro celkovou uživatelskou zkušenost.

Moderní vývoj webových aplikací si klade za cíl nejen estetickou a funkční dokonalost, ale také inkluzivitu, která zajišťuje přístupnost pro co nejširší spektrum uživatelů. Z tohoto důvodu se mimo jiné stává důležitým faktorem také zapojení uživatelů s různými typy omezení, kterými jsou například zraková nebo motorická postižení.

Práce se zaměřuje na analýzu a implementaci prvků přístupnosti do webových aplikací, a to jak těch standardních definovaných standardem W3C WAI, tak i nadstandardních funkcí, které mohou přinést významné vylepšení uživatelské zkušenosti. Cílem je vytvořit prostředí, které bude přístupné pro co nejvíce uživatelů a zároveň poskytne nadstandardní funkce pro vylepšení uživatelské interakce.

Práce se bude zabývat konkrétními technologiemi a metodami pro implementaci těchto prvků a jejich vlivem na uživatelskou zkušenost. Důraz bude kladen nejen na samotné technické řešení, ale také na optimalizaci a testování, aby byla zajištěna maximální efektivita a použitelnost v praxi.

3. DEFINOVANÉ STANDARDY PODLE W3C WAI

Ve snaze o zajištění přístupnosti webových technologií pro všechny uživatele, včetně těch s různými typy omezení, vyvinulo World Wide Web Consortium (W3C¹) iniciativu nazvanou Web Accessibility Initiative (WAI²). Tato kapitola se zaměřuje na standardy a doporučení definované touto iniciativou s cílem pomoci vylepšit přístupnost webových aplikací.

3.1 WCAG – WEB CONTENT ACCESSIBILITY GUIDELINES

Nejdůležitějším dokumentem v rámci W3C WAI jsou Web Content Accessibility Guidelines (WCAG³). Tyto směrnice jsou jako příručka pro tvůrce webů, aplikací, dokumentů a dalších online obsahů, aby byly dostupné pro všechny uživatele, bez ohledu na jejich schopnosti nebo omezení. WCAG se zaměřuje na čtyři hlavní principy:

Perceivable (Vnímatelnost): Webový obsah by měl být vnímatelný smysly – zrak, sluch, dotek atd.

Operable (Ovladatelnost): Uživatelé by měli být schopni ovládat obsah pomocí různých vstupních zařízení, jako jsou klávesnice, myši, hlasové ovládání atd.

Understandable (Srozumitelnost): Obsah by měl být jasný a srozumitelný pro všechny uživatele, bez ohledu na jejich úroveň schopností.

Robust (Robustnost): Obsah by měl být vytvořen tak, aby fungoval spolehlivě na různých zařízeních a s různými technologiemi.

WCAG poskytuje konkrétní pokyny a techniky, jak dosáhnout těchto principů. Mezi tyto techniky patří například správné použití alternativních textů pro obrázky, logické struktury stránek pro usnadnění navigace nebo dostatečný kontrast barev pro zlepšení čitelnosti.

3.2 ATAG – AUTHORIZING TOOL ACCESSIBILITY GUIDELINES

Dalším důležitým dokumentem je Authoring Tool Accessibility Guidelines (ATAG⁴). Tyto směrnice se zaměřují na nástroje pro tvorbu webového obsahu, jako jsou redaktoři, správci obsahu (CMS – Content Management System) a další. Cílem ATAG je zajistit, aby tyto nástroje byly samy o sobě přístupné a uživatelsky přívětivé pro tvůrce obsahu, kteří vytvářejí webové stránky a aplikace.

¹ W3C – <https://www.w3.org/>

² WAI – <https://www.w3.org/WAI/>

³ WCAG – <https://www.w3.org/TR/WCAG21/>

⁴ ATAG – <https://www.w3.org/WAI/standards-guidelines/atag/>

ATAG stanovuje tři hlavní oblasti:

Přístupnost editorů: Nástroje pro tvorbu obsahu by měly umožňovat tvorbu přístupného obsahu bez ohledu na schopnosti tvůrce obsahu. To zahrnuje například možnost přidávat alternativní texty k obrázkům, kontroly na validní HTML⁵ strukturu nebo možnost snadného vytváření přístupných formulářů.

Přístupnost výstupního obsahu: Nástroje by měly generovat webový obsah, který je sám o sobě přístupný a splňuje požadavky WCAG.

Podpora tvorby přístupného obsahu: Nástroje by měly poskytovat uživatelům informace a nástroje, které jim pomohou vytvářet přístupný obsah.

3.3 UAAG – USER AGENT ACCESSIBILITY GUIDELINES

User Agent Accessibility Guidelines (UAAG⁶) se věnují uživatelským agentům, což jsou software, které umožňují uživatelům prohlížet, číst a interagovat s webovým obsahem, jako jsou prohlížeče, plug-iny a další rozšíření. UAAG stanovuje doporučení pro vývojáře těchto nástrojů, aby bylo zajištěno, že budou přístupné a uživatelsky přívětivé pro uživatele s různými potřebami. UAAG se zaměřuje na:

Přístupnost uživatelského rozhraní: Uživatelské rozhraní prohlížečů a dalších nástrojů by mělo být navrženo tak, aby bylo snadno použitelné pro uživatele s různými schopnostmi a potřebami.

Podpora přístupnosti webového obsahu: Nástroje by měly umožňovat uživatelům snadno přistupovat k různým typům webového obsahu a zlepšovat jeho přístupnost.

Nabízení alternativních způsobů interakce: UAAG doporučuje, aby nástroje poskytovaly alternativní způsoby interakce pro uživatele s různými schopnostmi, jako jsou hlasové ovládání, klávesové zkratky nebo možnost personalizace uživatelského rozhraní.

Všechny tyto standardy a doporučení společně tvoří důležitý rámec pro vytváření a hodnocení přístupnosti webových aplikací. Jejich dodržování je klíčové pro zajištění toho, aby byl internetový obsah dostupný pro všechny uživatele, bez ohledu na jejich schopnosti nebo omezení.

⁵ HTML – Hypertext Markup Language (<https://html.spec.whatwg.org/multipage/>)

⁶ UAAG – <https://www.w3.org/WAI/standards-guidelines/uaag/>

4. UŽIVATELSKÁ ČÁST

Projekt klade důraz především na uživatele a jsou tudíž alfou a omegou celé práce, proto je uživatelská část promyšlena do posledního detailu.

4.1 ŘEČNÍK

Text to Speech (TTS⁷) představuje klíčový prvek vylepšující uživatelskou stránku webových aplikací, a to zvláště pro uživatele s omezením vizuálního vnímání. Tato funkcionality transformuje textový obsah na zvukový výstup, což má významné důsledky pro dostupnost informací. Z hlediska uživatelské stránky se jedná o revoluční nástroj, který překračuje bariéry pro osoby se zrakovým postižením, ale může být prospěšný i pro ty, kteří preferují poslech před čtením.

Pro nevidomé a slabozraké uživatele má Text to Speech klíčový význam při přístupu k obsahu webových stránek. Bez ohledu na to, zda jde o běžný text na blogu, komentáře pod příspěvky, nebo komplexní informace na odborném webu, TTS umožňuje těmto uživatelům snadný a plnohodnotný přístup k obsahu bez nutnosti spoléhat se výhradně na vizuální vnímání. Tímto způsobem se zvyšuje dostupnost informací a vytváří se inkluzivnější online prostředí.

Po uživatelské stránce se Řečník spouští pomocí klávesnice, konkrétněji je spuštění defaultně nastavené na klávesu „2“. Po stisknutí se spustí oznamovací hlas, který uživateli oznámí, že je Řečník spuštěný. Následuje proces prvního předřikání. Řečník lze ovládat pomocí šipek na klávesnici. Šipka doprava přeskočí na další element a šipka zpět se vrátí na předchozí element.

Funkcionalitu lze uzavřít dvěma způsoby. Buď stisknutím klávesy „espace“, nebo klávesovou zkratkou, kterou se funkcionality spouští.

Kromě ovládání přes klávesnici lze Řečníka také ovládat graficky pomocí implementované komponenty. Po kliknutí na tlačítko spuštění Řečníka se zobrazí grafický ovládací prvek, který umožňuje uživatelům spravovat Řečníka. Na této komponentě lze pozastavit Řečníka, přeskočit na další odstavec textu pomocí šipek, nastavit rychlost čtení, a samozřejmě také tlačítko pro vypnutí Řečníka.

Celkově lze Řečník považovat za revoluční nástroj, který přispívá k vytváření přístupného a inkluzivního online prostředí. Jeho implementace do naší aplikace zlepšuje uživatelskou zkušenost a umožňuje širšímu spektru uživatelů pohodlnější a efektivnější interakci s obsahem webových stránek.

⁷ TTS – https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API

4.2 HLASOVÝ NAVIGÁTOR

Další klíčovou součástí uživatelské části aplikace je Hlasový Navigátor. Tato funkce je navržena s kladením důrazu na interaktivní a efektivní průvodce webovými stránkami pomocí hlasových pokynů, což zvyšuje přístupnost pro uživatele s omezením motoriky nebo vizuálního vnímání.

Hlasový navigátor využívá funkci Speech to Text (SpeechRecognition⁸) v JavaScriptu umožňuje převod mluvené řeči na text. To poskytuje možnost interakce s webovou aplikací nebo stránkou pomocí hlasových příkazů, což může být užitečné pro uživatele s omezením nebo pro vytvoření hlasově aktivovaných ovládacích prvků. SpeechRecognition je standardní součástí Web Speech API, které je podporováno ve většině moderních prohlížečů, včetně Chrome.

Hlasový Navigátor lze snadno aktivovat pomocí klávesové zkratky, což umožňuje uživatelům přizpůsobit si své preference. Defaultně je hlasový navigátor spustitelný pomocí klávesy „3“.

Tato funkce umožňuje uživatelům navigaci na webových stránkách pomocí hlasových pokynů. Například uživatel může říci "Přejít na domovskou stránku" a navigátor automaticky provede přechod na domovskou stránku webové aplikace. Podobně lze použít hlasové příkazy pro navigaci mezi odkazy, vyplnění formulářů nebo pro přístup k různým funkcím aplikace.

Hlasový Navigátor je navržen tak, aby byl snadno ovladatelný a intuitivní pro uživatele. Po získání hlasového vstupu se provede analýza příkazu a provede se odpovídající akce v souladu s funkcionalitou aplikace. Pro uživatele je takto poskytnuta alternativa k tradičnímu ovládání pomocí klávesnice nebo myši, což zvyšuje dostupnost a uživatelskou pohodlnost pro širší spektrum uživatelů.

4.3 SPEECH FOCUS

Speech Focus představuje klíčovou funkcionalitu aplikace, která zajišťuje orientaci na webových stránkách pomocí focusAble elementů. Tyto elementy jsou definovány tak, že mohou být aktivovány klávesou „TAB“. V HTML jsou některé elementy přednastaveny jako focusAble, ale pro ty, které tuto možnost implicitně nemají, nebo pro ty, které by neměly být zobrazeny, lze využít atribut „tabindex“. Tímto atributem lze nastavit prioritu zobrazení pro daný element. Pokud je hodnota atributu nastavena na „0“, znamená to, že element nebude aktivován pomocí klávesy „TAB“. Další hodnoty atributu určují pořadí prioritního zobrazení elementu (např. první, druhý, až poslední).

Speech Focus může být aktivován uživatelem stisknutím klávesy „1“, a následně je uživateli potvrzeno zapnutí této funkce. Ovládání probíhá výhradně pomocí klávesy „TAB“, což umožňuje postupné proklikávání mezi jednotlivými focusAble elementy. Funkci lze vypnout

⁸ SpeechRecognition – <https://developer.mozilla.org/en-US/docs/Web/API/SpeechRecognition>

dvěma způsoby: opětovným stiskem klávesy „1“ nebo klávesou „ESCAPE“ (která zároveň vypíná celou knihovnu a všechny její funkce).

Speech Focus analyzuje obsah na focus elementu a hledá texty k přečtení, přičemž nejdříve zjišťuje o jaký typ elementu se jedná. Toto slouží k tomu, aby poté vybral z pole pořadí priorit vyhledávání možných textů k přečtení. Následně postupně podle priorit zkouší, zda se tam daný text nachází a pokud ano přečte ho a zastaví se. Například pro elementy typu input, kde není přímo dostupný textContent, může algoritmus začít hledáním u popisků (labels) a pokračovat například u titulků. Naopak u odkazů (link) je pole prioritizováno jinak, přičemž textContent je na prvním místě, následovaný titulkem atd.

```
switch (focusEl.nodeName) {
  case "A":
  case "BUTTON":
  case "SUMMARY":
    newField = fields[0];
    break;
  case "INPUT":
    newField = fields[1];
    break;
  default:
    newField = fields[1];
    break;
};

const fields = [
  ["textContent", "title", "aria-label", "alt", "name"],
  ["labels", "title", "placeholder", "aria-label", "alt"],
  ["aria-label", "textContent", "labels", "title", "placeholder", "alt"]
];
```

Obrázek 1: Obrázek kódu, pole priorit a switch s elementy

5. ADMINISTRÁTORSKÁ ČÁST

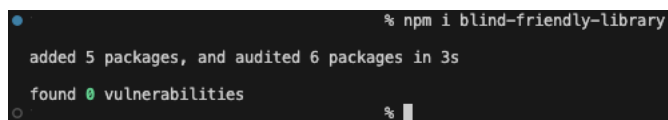
V této kapitole se zaměříme na administrátorskou část aplikace, kde budou podrobněji rozebrány dvě klíčové podkapitoly, a to konkrétně Instalace Knihovny a Nastavení a Spuštění Knihovny. Tyto sekce poskytnou uživatelům a správcům potřebné informace k úspěšnému nasazení, konfiguraci a správě funkcí knihovny.

5.1 INSTALACE KNIHOVNY

Knihovna je dostupná na platformě npm⁹, což umožňuje jednoduché stažení do vašeho projektu využívajícího node.js. Postup instalace je snadný – v konzoli ve vašem projektovém adresáři jednoduše zadejte následující příkaz:

```
`npm install blind-friendly-library`10
```

Po provedení tohoto příkazu bude knihovna stažena a automaticky umístěna do složky s dalšími knihovnami, konkrétně do adresáře (node_modules).



Obrázek 2: Obrázek kódu z konzole

5.2 NASTAVENÍ A SPUŠTĚNÍ KNIHOVNY

Všechny kroky od instalace, přes nastavení, až po vysvětlení jednotlivých funkcí jsou detailně popsány v dokumentaci, kterou můžete najít v souboru *README.md*¹¹ v projektu knihovny nebo na webových stránkách knihovny (<https://www.npmjs.com/package/blind-friendly-library>). Po úspěšné instalaci je potřeba knihovnu importovat a spustit v souboru *main.js* projektu, ve kterém je knihovna využívána.

Konkrétněji importováním souboru *speaker.js* z knihovny. Poté, pro spuštění knihovny, je potřeba zavolat funkci *setBFL()* a nastavit její základní parametry pro hlasového navigátora a celou knihovnu (volume, rate, pitch, lang).

⁹ NPM – <https://www.npmjs.com/>

¹⁰ BFL – <https://www.npmjs.com/package/blind-friendly-library>

¹¹ Základní příručka projektu pro ostatní vývojáře

```
import { setBFL, setKeyBFL } from '../lib/index';

const myKey = {
  kscFocus: 'event.key === "0"',
  kscSpeaker: 'event.key === "9"',
  kscVoiceControl: 'event.key === "8"',
  kscManual: 'event.key === "7"',
};

setBFL(1, 1, 1.2, "en");
setKeyBFL(myKey);
```

Obrázek 3: Obrázek kódu, importování knihovny v main.js a nastavení nových klávesových zkratk

Po těchto krocích bude například funkce SpeechFocus automaticky aktivní. Ostatní funkce vyžadují přidání vlastních atributů k elementům na webu. Pro funkci řečníka je třeba na tyto elementy přidat custom atribut, konkrétně *data-speaker*. Řečník poté, jakmile je aktivován, bude číst všechny elementy, které mají tento atribut. Autor má následně možnost mezi těmito elementy přepínat pomocí šipek. Existuje i druhá možnost, jak uživateli jednoduše umožnit přístup k řečníkovi, a to pomocí vložení komponenty z knihovny. Tato komponenta funguje poté sama od sebe a uživatel v ní má možnosti, jak si přizpůsobit a ovládat řečníka sám.

Funkce ovládání hlasem nabízí možnost nastavení podobně jako u řečníka, což přináší zcela novou úroveň interakce s webovými stránkami. K tomu, aby byly určité prvky ovladatelné hlasově, stačí přidat atribut *data-el-text*. Tento atribut obsahuje text, který je potřeba vyslovit, aby se vyvolala daná akce. Díky podpoře vícejazyčných webů umožňuje knihovna přidat do jednoho prvku různé verze textu, například českou verzi pomocí atributu *data-el-text-cs*. Stačí tedy přidat zkratku jazyka na konec atributu (-cs pro češtinu), a uživatelé se mohou snadno orientovat ve svém rodném jazyce.

Jednou z dalších funkcí je možnost přidání atributu *data-el-action*, který definuje konkrétní akci, která se má provést po rozpoznání hlasovým navigátorem. Standardní akcí je "click", což simuluje kliknutí myši na daný prvek, ideální například pro odkazy. Přidání nových příkazů do hlasového navigátoru je snadné díky funkci, kterou lze importovat z knihovny. Stačí definovat nový příkaz jako textový řetězec a určit akci, která má následovat. Tímto se otevírá široká škála možností pro přizpůsobení interakce s webovými stránkami prostřednictvím hlasového ovládání.

```
addCommand('new commad', () => { console.log('This is a new command!'); });
```


Obrázek 4: Obrázek kódu, importování metody na klasickém webu, díky které lze přidat nový příkaz

5.3 DALŠÍ FUNKCE A MOŽNOSTI KNIHOVNY

Mezi další užitečné funkce, které knihovna nabízí, patří hlavně funkce k vylepšení a přizpůsobení si knihovny a řečníka podle sebe. Administrátor má možnost si importovat z knihovny několik funkcí včetně: Zapnutí řečníka, přepnutí na jiný odstavec, zastavení řečníka, atd... Jde celkově o funkce, které umožňují používání řečníka a uživatelé mají možnost k němu vytvořit svoje vlastní grafické rozhraní.

Knihovna také nabízí předvytvořené grafické rozhraní, které stačí importovat do vlastního projektu jako komponentu. Je možné si poté tuto komponentu nastylovat podle vlastního designu. Komponenta obsahuje základní funkce pro řečníka.

V projektu je k správě překladů řečníka využívána knihovna *i18n-js*¹². Tato knihovna umožňuje snadnou integraci textů z *.json*¹³ souboru příslušného jazyka. Přidání nových jazykových verzí je jednoduché – stačí vytvořit nový *.json* soubor s překlady a aplikace automaticky načte a použije tento nový soubor pro daný jazyk. Díky této struktuře je možné aktualizovat překlady bez zásahu do kódu aplikace, což usnadňuje údržbu a správu textů v různých jazycích. Je důležité mít primární verzi textů v angličtině, aby uživatelé měli k dispozici alespoň anglickou verzi textů, pokud není k dispozici překlad do jejich jazyka. To minimalizuje možné problémy s chybějícími nebo nekompletními texty a vede k lepší uživatelské zkušenosti. Celkově je používání knihovny *i18n-js* efektivním způsobem, jak spravovat a lokalizovat texty v aplikaci a minimalizovat chyby spojené s nedostupnými překlady.



```
"globalSpeech": {
  "textNotFound": "Nenachází se zde žádný text k přečtení",
  "ttsEnd": "Řečník byl vypnut",
  "ttsStart": "Řečník byl zapnut"
},
"globalSpeech": {
  "textNotFound": "There is no text to read",
  "ttsEnd": "The speaker has been turned off",
  "ttsStart": "The speaker was turned on"
},
```

Obrázek 5 úryvek z *.json* souboru z překladů (vlevo český překlad a vpravo anglický překlad)

¹²*i18n-js* knihovna – <https://www.npmjs.com/package/i18n-js>

¹³ JavaScript Object Notation

6. KNIHOVNA

V této kapitole se hlouběji ponoříme do samotné struktury a organizace knihovny. Kapitola 6 poskytne uživatelům důkladné informace o vzniku, publikaci a správě knihovny npm, aby ji mohl poté kdokoliv vytvořit.

6.1 ZALOŽENÍ KNIHOVNY

Knihovna je založena a publikována na npm (Node.js Package Manager). Npm slouží jako správce balíčků pro JavaScript a je výchozím správcem balíčků pro Node.js¹⁴. Disponuje online databází knihoven, ke které lze přistupovat jak interně, tak programově. Na webových stránkách npm lze nalézt veřejné knihovny vytvořené komunitou.

Pro úspěšné publikování nové knihovny je nutné řádně nastavit soubor *package.json* ve vašem projektu. Tento soubor musí obsahovat následující informace:

- *Name* (jméno projektu)
- *Description* (podrobnější vysvětlení projektu)
- *Version* (aktuální verze knihovny, začíná se od 0.0.0)
- *Repository* (odkaz na git repositář, kde je knihovna uložena. Obvykle na GitHubu¹⁵)
- *Keywords* (slova, podle kterých bude vyhledávána knihovna na internetu)
- *Author* (jméno autora)
- *License* (licence, pod kterou je chráněna)
- *Files* (specifikuje, které přesně soubory má obsahovat knihovna)

```
{
  "name": "blind-friendly-library",
  "version": "1.0.68",
  "description": "BFL - Blind friendly library",
  "main": "src/lib/index.js",
  "repository": {
    "type": "git",
    "url": "git+https://github.com/FilipekBenes/BlindFriendlyWeb.git"
  },
  "keywords": [
    "Blind",
    "JS",
    "TTS",
    "VoiceControl",
    "SpeechFocus",
    "STT"
  ],
  "author": "Filip Beneš",
  "license": "MIT",
  "bugs": {
    "url": "https://github.com/FilipekBenes/BlindFriendlyWeb/issues"
  },
  "homepage": "https://github.com/FilipekBenes/BlindFriendlyWeb",
  "files": [
    "src/lib/*"
  ]
}
```

Obrázek 6: Obrázek kódu, setup v *package.json*

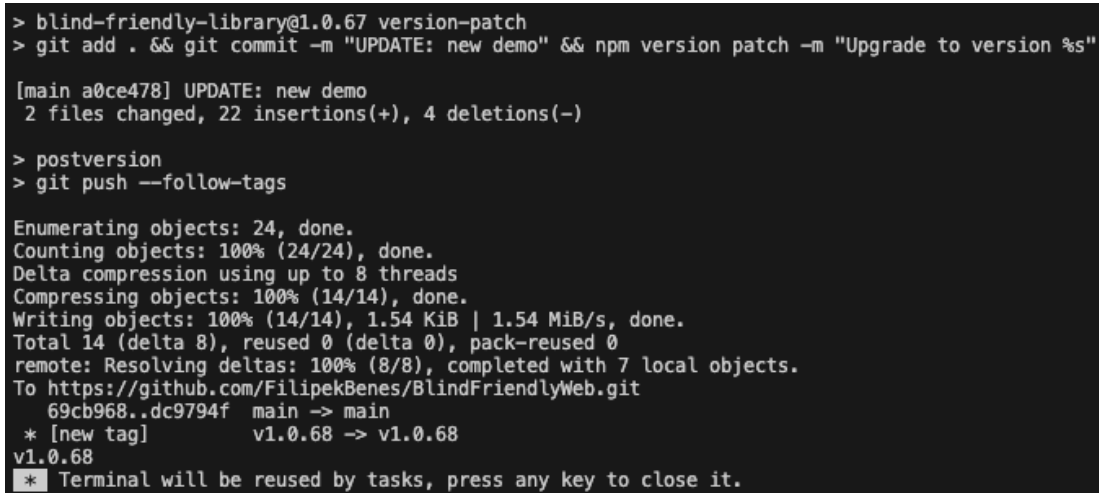
¹⁴ NodeJS – <https://nodejs.org/en>

¹⁵ Github – <https://github.com/>

Po nastavení *package.json* musí autor své nové knihovny vytvořit účet na *npmjs.com*¹⁶ a přihlásit se. Následně se v konzoli přihlásí do svého účtu pomocí příkazu ``npm login`` a zadá své uživatelské jméno a heslo. Pokud bylo přihlášení úspěšné, zobrazí se tato zpráva:

Logged in as <uživatelské-jméno> on https://registry.npmjs.org/.

Následně stačí jednoduše spustit příkaz ``npm publish`` v konzoli pro publikování nové knihovny. Pokud vše proběhne v pořádku, objeví se následující zpráva:



```
> blind-friendly-library@1.0.67 version-patch
> git add . && git commit -m "UPDATE: new demo" && npm version patch -m "Upgrade to version %s"

[main a0ce478] UPDATE: new demo
 2 files changed, 22 insertions(+), 4 deletions(-)

> postversion
> git push --follow-tags

Enumerating objects: 24, done.
Counting objects: 100% (24/24), done.
Delta compression using up to 8 threads
Compressing objects: 100% (14/14), done.
Writing objects: 100% (14/14), 1.54 KiB | 1.54 MiB/s, done.
Total 14 (delta 8), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (8/8), completed with 7 local objects.
To https://github.com/FilipekBenes/BlindFriendlyWeb.git
 69cb968..dc9794f  main -> main
 * [new tag]         v1.0.68 -> v1.0.68
v1.0.68
* Terminal will be reused by tasks, press any key to close it.
```

Obrázek 7: Obrázek z konzole po publikování nové verze

Tím je knihovna úspěšně publikována a dostupná pro další vývojáře přes npm.

¹⁶ <https://www.npmjs.com/signup>

6.2 DEMONSTRAČNÍ WEB

Součástí této knihovny je i demonstrační web¹⁷, který slouží k vysvětlení a ukázce funkcí, které knihovna nabízí, a zároveň ukazuje, jak ji správně používat. V dnešní době je takový demonstrační web nezbytným prvkem pro prezentaci software a jeho schopností. Demonstrace na živém webu umožňuje uživatelům interaktivně prozkoumat knihovnu a lépe si představit, jak by mohla jejich práci usnadnit.

Hlavní stránka demonstračního webu slouží jako úvodní bod, kde jsou poskytnuty základní informace o projektu. Tato stránka je klíčová pro uživatele, kteří se seznamují s knihovnou poprvé, a poskytuje jim základní informace pro úspěšné začlenění knihovny do svého projektu. Stránka obsahuje:

- Popis projektu: Stručný přehled toho, co knihovna umožňuje a jakým způsobem může být užitečná pro uživatele.
- Autor: Informace o tvůrci knihovny a jeho kontaktní údaje pro případné dotazy či podporu.
- Návod k instalaci: Podrobný návod, jak knihovnu nainstalovat do svého projektu.
- Tutoriál: Praktické instrukce k použití knihovny, včetně příkladů a doporučených postupů.
- Spuštění: Návod k základnímu spuštění knihovny a prvotnímu nastavení.

Druhou částí demonstračního webu je sekce „Demo“, která je zaměřena na předvedení konkrétních funkcí a schopností knihovny. V této části jsou uvedeny:

- Seznam funkcí: Detailní popis jednotlivých funkcí, které knihovna poskytuje, spolu s ukázkami kódu a výsledným efektem.
- Interaktivní prostředí: Možnost interakce s knihovnou prostřednictvím demo stránky, kde uživatel může vyzkoušet funkce a experimentovat s různými parametry.
- Cvičná data: Předpřipravená cvičná data a elementy, na kterých je možné okamžitě vyzkoušet funkcionality knihovny.

Cílem této sekce je umožnit uživatelům reálně prozkoumat, jak knihovna funguje a jak může být integrována do jejich vlastních projektů. Interaktivní prostředí a připravená cvičná data jsou klíčové pro získání konkrétních zkušeností s použitím knihovny. Po prohlédnutí tohoto demonstračního webu by měl uživatel mít jasnou představu o možnostech a výhodách použití knihovny a být připraven ji začlenit do svého vlastního softwarového projektu.

¹⁷ BFL Demo – <https://bfl-t6ps.onrender.com/>

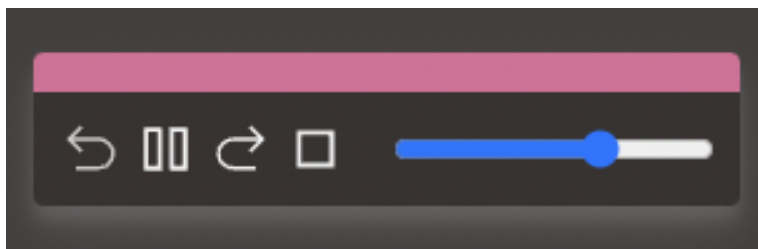
6.3 IMPLEMENTACE NA REÁLNÝ PROJEKT

Knihovna je využívána na inovativním projektu DEH¹⁸ (Digitální učebnicová platforma). Jde o projekt, který se zaměřuje na vzdělání a lze v něm jednoduše vytvářet své vlastní učebnice. Konkrétně na tomto projektu je tato knihovna velmi užitečná, jelikož díky ní je projekt DEH přístupný i pro uživatele se zrakovým postižením.

Uživatelé si mohou nechat předčítat celé články. To je velmi užitečné nejen pro uživatele se zrakovým postižením, ale i pro uživatele, kteří chtějí články poslouchat jenom jako podcast a nemusí je tedy číst. Tito uživatelé určitě také ocení grafické rozhraní pro řečníka. Konkrétně na DEH lze spustit řečník pomocí tlačítka, a poté se zobrazí posunovatelný element, na kterém jsou základní funkce řečníka. Mezi tyto funkce patří například: pozastavení, posunutí na předchozí odstavec, posunutí na další odstavec, ukončení řečníka, nebo také nastavení rychlosti mluvení řečníka.

Hlasový navigátor v tomto systému pomáhá orientaci na webu. Slouží jako pomocník při přecházení mezi stránkami, které se nachází v navigaci v hlavičce webu. Pomocí klávesnice uživatel zapne hlasového navigátora a poté zadá příkaz, který chce, aby se provedl.

Vyzkoušet si tuto knihovnu na již používaném projektu je možné na stránce biobrejn.cz¹⁹. Jedná se o jednu z již plně funkčních instancí Digitální webové platformy, která byla vytvořena pro pana Mgr. Šimona Hrozinku.



Obrázek 8: Obrázek z grafického rozhraní Řečníka na webové aplikaci Biobrejn

6.4 SPOLUPRÁCE SE ŠKOLOU PRO ZRAKOVĚ POSTIŽENÉ

V tuto chvíli je snaha navázat spolupráci se školou pro zrakově postižené v Praze. Osloveno bylo několik škol s možností vyzkoušení si této knihovny na projektu biobrejn.cz. Zatím byla odezva od školy *Gymnázium pro zrakově postižené a Střední odborná škola pro zrakově postižené v Praze 5*²⁰. Do budoucna se dá předpokládat navázání další spolupráce, která pomůže ve vývoji projektu a zejména prokáže jeho praktické využití.

¹⁸ DEH web – <https://ekdyson-dev.github.io/DEH-web/>

¹⁹ Biobrejn web – <https://biobrejn.cz/>

²⁰ GoPraha web – <https://www.goapraha.cz/kontakty/>

7. POUŽITÉ TECHNOLOGIE

Při výběru technologií pro tento projekt jsem pečlivě zvažoval mé stávající znalosti a specifické potřeby aplikace. Pro vývoj frontendu k demo webu jsem se rozhodl použít moderní JS framework Vue.js, který nabízí vynikající výkon, flexibilitu a silnou podporu komunity vytvářející vlastních komponent a rozšíření.

7.1 VUE.JS

Při výběru mezi Vue.js a React.js pro demo web jsem zvolil Vue.js kvůli jeho menší velikosti, vyššímu výkonu a snadné přizpůsobitelnosti. Vue.js je navržen pro tvorbu uživatelských rozhraní a poskytuje deklarativní programovací model založený na komponentách, což usnadňuje efektivní vývoj. Tato volba mi umožňuje pracovat s moderními technologiemi a dosáhnout vysoké úrovně výkonu a flexibility aplikace. Díky šablonové syntaxi Vue.js mohu snadno definovat výstup HTML na základě stavu JavaScriptu a automatické sledování změn v JS umožňuje reaktivní aktualizaci DOM, což je klíčové pro dynamické a responzivní uživatelské rozhraní mého demo webu.

7.2 SCSS

Jedná se o preprocesor, pro jednodušší práci s CSS. SCSS je zkratka pro Sassy Cascading Style Sheets a jde de facto o pokročilejší variantu CSS. Kromě přípony souborů .scss se od standardního CSS liší i větším množstvím funkcí, které přispívají k více spolehlivému a DRY²¹ CSS.

Dále rozšiřuje CSS o řadu funkcí, jako jsou například mixiny, funkce, vnoření tříd apod. Mixiny jsou vhodné pro vytknutí stejných částí stylů do jednoho „bloku“, který pak stačí referovat na potřebných místech pomocí „@include“. Lze jim také vkládat argumenty, které pak mohou být použity ve stylech mixinu. Podobné mixinům jsou funkce, ale ty nevrací blok stylů, nýbrž mohou vrátit hodnotu použitelnou ve stylu. Funkcím lze také vkládat argumenty. Výhodou také je, že všechny CSS verze jsou kompatibilní s SCSS. Pro optimalizaci kompilace je možné v názvu souboru použít prefix „_“ a tím z něj vytvořit tzv. modul. Ten se nejdříve vloží do souborů, které ho importují, a až tyto soubory se zkompilují. Tuto funkci využíváme například pro soubor s mixiny.

²¹ DRY – Don't Repeat Yourself

8. ZÁVĚR

Tato ročníková práce se zaměřila na implementaci prvků přístupnosti do webových aplikací a na vytvoření knihovny s nadstandardními funkcemi pro usnadnění uživatelské interakce. Práce se zabývala analýzou relevantních standardů a doporučení W3C WAI, včetně WCAG, ATAG a UAAG, a implementovala tyto standardy do webové aplikace.

Knihovna s nadstandardními funkcemi zahrnuje řečníka pro předčítání textu na webových stránkách, hlasový navigátor pro navigaci pomocí hlasových příkazů a Speech Focus pro usnadnění navigace pomocí klávesnice a focusable elementů. Tato knihovna je dostupná na platformě npm a je doplněna o demonstrační web, který slouží k prezentaci funkcí a návodům k použití.

Výsledky práce naznačují, že implementace prvků přístupnosti a nadstandardních funkcí může výrazně zlepšit uživatelskou zkušenost pro široké spektrum uživatelů, včetně těch se zrakovým, sluchovým nebo motorickým postižením.

Knihovna s nadstandardními funkcemi představuje inovativní nástroj pro vývojáře, kteří chtějí vylepšit přístupnost svých webových projektů. Poskytuje sadu funkcí, které lze snadno integrovat do existujících aplikací a usnadnit tak jejich používání pro všechny uživatele.

Další kroky by mohly zahrnovat další rozvoj knihovny o další funkce a rozšíření kompatibility s různými platformami, propagaci knihovny mezi vývojáři a designéry a spolupráci s organizacemi sdružujícími osoby se specifickými potřebami pro sbírání zpětné vazby a další zlepšování funkcí.

Celkově lze konstatovat, že implementace prvků přístupnosti a nadstandardních funkcí do webových aplikací je důležitým krokem směrem k vytváření inkluzivního online prostředí. Tato práce ukazuje, že moderní technologie a standardy mohou významně přispět ke zlepšení přístupnosti webu a usnadnit jeho používání napříč uživateli. Budoucí práce a spolupráce mohou dále rozvíjet tyto koncepty a přinést ještě větší benefity celé on-line komunitě.

9. POUŽITÉ ZDROJE

1. You, Evan. Vue.js. [Online] 11 2023. <https://vuejs.org/>.
2. StackOverflow. *stackoverflow*. [Online] 11 2023. <https://stackoverflow.com/>.
3. mozilla.org. SpeechSynthesis. [Online] 10 2023. <https://developer.mozilla.org/en-US/docs/Web/API/SpeechSynthesis>.
4. —. SpeechRecognition. [Online] 10 2023. <https://developer.mozilla.org/en-US/docs/Web/API/SpeechRecognition>.
5. —. Using_custom_elements. [Online] 1 2024. https://developer.mozilla.org/en-US/docs/Web/API/Web_components/Using_custom_elements.
6. Semah, Benjamin. How to Create and Publish an NPM Package – a Step-by-Step Guide. [Online] 1. 2 2023. <https://www.freecodecamp.org/news/how-to-create-and-publish-your-first-npm-package/>.
7. fnando. i18n-js. [Online] 12 2023. <https://www.npmjs.com/package/i18n-js>.
8. W3C®. About W3C web standards. [Online] 12 2023. <https://www.w3.org/>.

10. SEZNAM OBRÁZKŮ

Obrázek 1: Obrázek kódu, pole priorit a switch s elementy	11
Obrázek 2: Obrázek kódu z konzole	12
Obrázek 3: Obrázek kódu, importování knihovny v main.js a nastavení nových klávesových zkratk.....	13
Obrázek 4: Obrázek kódu, importování metody na klasickém webu, díky které lze přidat nový příkaz	13
Obrázek 5 úryvek z .json souboru z překladů (vlevo český překlad a vpravo anglický překlad)	14
Obrázek 6: Obrázek kódu, setup v package.json	15
Obrázek 7: Obrázek z konzole po publikování nové verze	16
Obrázek 8: Obrázek z grafického rozhraní Řečníka na webové aplikaci Biobrejn	18