

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií



**KATEDRA
INFORMAČNÍCH
TECHNOLOGIÍ
PEF ČZU V PRAZE**

**Dokumentace k projektu z předmětu
Web design**

Rehab from Vibe Coding

Bakulina Neonila, Honcharov Oleksandr, Kuzmina

Ekaterina, Shamedzka Kseniya

© 2025

Obsah

1	Úvod	4
1.1	Představení projektu	4
1.2	Struktura webu	4
1.3	Cílová skupina a účel	4
1.4	Technologický přístup	5
2	Požadavky na implementaci.....	6
2.1	Popis webových stránek	6
2.1.1	Úvodní stránka (index.html).....	6
2.1.2	Seznam terapeutů (pages/therapists/list.html).....	6
2.1.3	Detailní profily terapeutů	6
2.2	Chování a interaktivita stránek	7
2.2.1	Responzivní layout	7
2.2.2	Navigace.....	7
2.2.3	Vyhledávání a interaktivní efekty.....	7
2.3	HTML požadavky	7
2.3.1	Struktura dokumentu	7
2.3.2	Sémantické elementy	7
2.3.3	Přístupnost (WCAG 2.0 Level AA)	8
2.4	CSS požadavky	8
2.4.1	Architektura CSS	8
2.4.2	CSS Custom Properties	8
2.4.3	CSS Grid layout	8
2.4.4	Flexbox	9
2.4.5	Responzivní design	9
2.5	JavaScript požadavky	9
2.5.1	Vanilla JavaScript	9
2.5.2	Funkcionality.....	9
2.6	Návrh a zdůvodnění technologií	9
2.6.1	HTML5, CSS3 a Vanilla JavaScript	9
2.6.2	Statický web a GitHub Pages	10
3	Využití generativní AI	11
3.1	Úvod	11
3.2	Nástroje pro generování textového obsahu	11

3.2.1	GPT 5.1	11
3.3	Nástroje pro asistenci při psaní kódu	12
3.3.1	Claude Code (Sonnet 4.5).....	12
3.3.2	GitHub Copilot (Code Review).....	13
3.4	Kritické zhodnocení AI nástrojů	14
3.4.1	Přesnost a spolehlivost	14
3.4.2	Přínosy a omezení	15
3.4.3	Doporučení pro použití AI	15
3.5	Závěr	15
4	Splnění požadavků na projekt	16
4.1	Obecné požadavky	16
4.1.1	Požadavek 1: Projekty tvoří studenti ve skupinách po čtyřech	16
4.2	Téma	16
4.2.1	Požadavek 2: Poskytovatel zdravotních služeb.....	16
4.3	Základní požadavky	16
4.4	HTML5 a sémantické elementy	17
4.4.1	Požadavek 13: Nejnovější HTML5	17
4.4.2	Požadavek 14: Sémantické elementy	17
4.5	CSS požadavky	17
4.5.1	Požadavek 17: Bez použití frameworků	17
4.5.2	Požadavek 18: 1 soubor CSS (alespoň 400 řádků).....	17
4.5.3	Požadavek 19: CSS Custom Properties	18
4.5.4	Požadavek 20: Grid layout	18
4.5.5	Požadavek 21: Elementy zabírají více buněk	18
4.5.6	Požadavek 22: Grid nelze jednoduše řešit flexboxem.....	18
4.5.7	Požadavek 23: Flexbox	18
4.6	Přístupnost	18
4.6.1	Požadavek 24: WCAG 2.0 Level AA	18
4.7	Další požadavky	19
4.7.1	Požadavek 25: Responzivní design.....	19
4.7.2	Požadavek 26: Mobile-first přístup	19
4.7.3	Požadavek 27: Relativní adresování	19
4.8	Dodatečně implementované prvky	19
4.9	Shrnutí splnění požadavků	19

1 Úvod

1.1 Představení projektu

Tento projekt představuje webové stránky fiktivního rehabilitačního centra s názvem **Rehab from Vibe Coding**. Web satiricky reflektuje současný trend nadměrného spoléhání se na nástroje umělé inteligence při vývoji softwaru. S humorným nadhledem poukazuje na problémy vývojářů, kteří postupně ztrácejí schopnost psát kvalitní, sémantický a udržitelný kód bez asistence AI.

Cílem projektu je nejen pobavit, ale také vzdělávat. Prezentuje rehabilitační centrum, které nabízí terapeutické programy zaměřené na obnovu fundamentálních programovacích dovedností – od psaní sémantického HTML a stylování pomocí čistého CSS bez frameworků, až po zajištění přístupnosti podle standardu WCAG 2.0 AA a bezpečnosti kódu. Tímto způsobem projekt demonstruje a obhajuje důležitost základních znalostí, které jsou v moderním vývoji často opomíjeny ve prospěch rychlých AI-generovaných řešení.

1.2 Struktura webu

Web se skládá ze tří hlavních částí: úvodní stránky představující poslání centra a přehled služeb, seznamu devíti terapeutů v responzivním grid layoutu a detailních profilů jednotlivých odborníků s biografiemi, expertízou a recenzemi. Detailní technické specifikace struktury a obsahu jednotlivých stránek jsou popsány v kapitole 2.

1.3 Cílová skupina a účel

Projekt je určen pro všechny vývojáře bez ohledu na úroveň zkušeností, kteří se stali závislými na AI asistovaném programování (tzv. “vibe coding”) a ztratili kontakt s fundamentálními dovednostmi. Primární cílovou skupinu tvoří vývojáři, kteří nekriticky přijímají AI návrhy bez porozumění kódu, spoléhají se výhradně na frameworky a generativní nástroje, nebo ztratili schopnost psát čistý HTML/CSS bez asistence. Sekundární skupinu představují týmoví lídři a senior vývojáři hledající vzdělávací materiál o rizicích nadměrné AI závislosti, stejně jako studenti informatiky, kteří si chtějí osvojit správné návyky a best practices ještě před vznikem závislosti na AI nástrojích.

Web má dvojí účel: vzdělávací a humorný. Na jedné straně poskytuje skutečné informace o důležitosti sémantického HTML, přístupnosti, bezpečnosti a čistého CSS bez frameworků. Na druhé straně používá satiru a humor k tomu, aby upozornil

na reálný problém nadměrné závislosti na AI nástrojích, který může vést ke ztrátě fundamentálních dovedností, bezpečnostním zranitelnostem a nekvalitnímu kódu.

1.4 Technologický přístup

Projekt je záměrně implementován pomocí čistých webových technologií bez použití frameworků – HTML5 se sémantickými elementy, CSS3 s moderními technikami (Grid, Flexbox, Custom Properties) a vanilla JavaScript. Tento přístup není pouze koncepční, ale slouží jako manifest hodnot, které centrum propaguje: fundamentální dovednosti před závislostí na nástrojích.

Web je plně responzivní s Mobile-first přístupem a splňuje standard WCAG 2.0 Level AA pro přístupnost. Projekt je nasazen na GitHub Pages a dostupný na adrese <https://katyabiser.github.io/Rehab-from-Vibe-Coding/>. Detailní technické specifikace, zdůvodnění výběru technologií a implementační detaily jsou popsány v kapitole 2.

2 Požadavky na implementaci

2.1 Popis webových stránek

Web se skládá z jedenácti HTML stránek: úvodní stránka, seznam terapeutů a devět detailních profilů. Každá stránka je navržena s důrazem na konzistenci designu, přístupnost a responzivní chování.

2.1.1 Úvodní stránka (index.html)

Úvodní stránka implementuje sémantické HTML5 elementy pro strukturu obsahu. Glosář odborných pojmu využívá `<dl>`, `<dt>`, `<dd>` elementy pro správnou sémantiku definition lists. Testimonials používají `<blockquote>` elementy s citacemi klientů. Footer je organizován pomocí CSS Grid layoutu s responzivním chováním – transformace z jednoho sloupce na mobilech do čtyř sloupců na desktopech. Tabulka příznaků AI závislosti obsahuje osm řádků s využitím `<thead>`, `<tbody>`, `<th scope="col">` pro přístupnost.

2.1.2 Seznam terapeutů (pages/therapists/list.html)

Stránka implementuje CSS Grid layout s devíti položkami. Grid používá `display: grid` s responzivními sloupci: `grid-template-columns: 1fr` na mobilech, `repeat(2, 1fr)` na tabletech ($\geq 768\text{px}$), `repeat(3, 1fr)` na desktopech ($\geq 1024\text{px}$). Tři karty obsahují conditional badge element s třídou `.card__badge` pro označení “Lead Therapist”. Každá karta je `<article>` element s ``, nadpisem a odkazem na detailní profil.

2.1.3 Detailní profily terapeutů

Každý terapeut má dedikovanou HTML stránku s breadcrumb navigací implementovanou pomocí `<nav>` s `` a `aria-current="page"` pro aktuální stránku. Recenze klientů používají `<blockquote>` elementy se sémantickou strukturou citací. Struktura dat zahrnuje: biografie (několik odstavců `<p>`), oblasti expertízy (neusporeádaný seznam ``), vybrané publikace (``), vzdělání a certifikace (``), terapeutický přístup (úvodní text + seznam) a call-to-action sekce s tlačítkem pro rezervaci.

2.2 Chování a interaktivita stránek

2.2.1 Responzivní layout

Web implementuje Mobile-first přístup s třemi breakpointy: Mobile ($<768\text{px}$) s výchozími styly a jedním sloupcem, Tablet ($\geq 768\text{px}$) s dvousloupcovými layouty a horizontální navigací, Desktop ($\geq 1024\text{px}$) s tří a čtyřsloupcovými layouty a maximální šířkou 1200px. Grid layout terapeutů se transformuje z jednoho sloupce přes dva na tři sloupce. Footer grid mění strukturu z jednoho na čtyři sloupce.

2.2.2 Navigace

Na mobilních zařízeních je menu defaultně skryté s hamburger tlačítkem. JavaScript (`navigation.js`) přidává třídu `nav_list-active` při kliknutí, což spouští slide-in animaci z pravé strany a nastavuje `aria-expanded="true"`. Menu se automaticky zavírá při kliknutí na odkaz, mimo menu nebo změně velikosti okna na tablet/desktop breakpoint. Na větších obrazovkách je menu vždy viditelné jako horizontální seznam.

2.2.3 Vyhledávání a interaktivní efekty

Vyhledávací pole je implementováno jako dropdown aktivovaný tlačítkem s ikonou lupy. JavaScript přidává třídu `search-dropdown-active`, zobrazuje dropdown s CSS transition a nastavuje automatický focus na input pole. Všechny interaktivní elementy mají hover efekty (karty s `transform: translateY(-5px)`, tlačítka s `transform: scale(1.02)`) a výrazné focus stavy s modrým obrysem pro klávesnicovou navigaci. Skip link je viditelný při focusu.

2.3 HTML požadavky

2.3.1 Struktura dokumentu

Všechny stránky používají HTML5 standard s `<!DOCTYPE html>`, `<html lang="en">`, meta tagy pro kódování UTF-8, viewport pro responzivitu, description, keywords a title specifický pro každou stránku. Favicon a apple-touch-icon jsou definovány. Všechny stránky linkují jeden centrální CSS soubor s relativním adresováním.

2.3.2 Sémantické elementy

Projekt využívá sémantické HTML5 elementy: `<header>` pro hlavičku s logem a navigací, `<nav>` s `aria-label`, `<main id="main-content">` jako target pro skip link, `<section>` pro tematické sekce, `<article>` pro karty a profily, `<aside>` pro testimonials,

<footer> pro patičku. Dodržována je správná hierarchie nadpisů s jedním <h1> na stránku. Tabulka používá <thead>, <tbody>, <th scope="col">. Glosář využívá <dl>, <dt>, <dd>.

2.3.3 Přístupnost (WCAG 2.0 Level AA)

Web splňuje WCAG 2.0 Level AA požadavky. Skip link je umístěn na začátku každé stránky. Použité jsou ARIA atributy (aria-label, aria-expanded, aria-current="page"). Screen reader texty používají třídu sr-only. Všechny obrázky mají smysluplné alt texty. Barevný kontrast je vyšší než 7:1 (tmavě šedá #212529 na bílém pozadí). Výrazné focus stavy zajišťují klávesnicovou navigaci.

2.4 CSS požadavky

2.4.1 Architektura CSS

Projekt používá 100% vlastní CSS bez frameworků. Veškerý CSS je organizován v jediném souboru `assets/css/style.css`, který obsahuje všechny styly strukturované do logických sekcí s komentáři: CSS Variables, Reset & Base Styles, Typography, Layout (Container, Header, Footer, Grid), Components (Buttons, Cards, Search, Hero, Tables, Breadcrumbs, Testimonials), Pages (Detail, Sections), Utilities (Accessibility, Helpers, Glossary) a Responsive Design. Celková velikost je 1249 řádků optimalizovaného kódu.

2.4.2 CSS Custom Properties

Systém CSS Custom Properties je definován v `:root` sekci souboru `assets/css/style.css`.

Pokrývá barevnou paletu (`-primary-color`, `-secondary-color`, `-accent-color`, atd.) a typografii (`-font-primary`, `-font-heading`, velikosti písma).

Spacing (`-spacing-xs` až `-spacing-xxl` od 0.5rem do 4rem) je oddělená skupina proměnných. Layout obsahuje `-max-width-container` a `-border-radius`. Další proměnné: `-box-shadow`, `-transition-speed`. Proměnné zajišťují konzistenci a snadné globální úpravy designu.

2.4.3 CSS Grid layout

CSS Grid je použit pro layout seznamu terapeutů (`.grid-therapists`) a footer (`.footer_grid`). Grid terapeutů obsahuje devět karet s responzivními sloupcí: `grid-template-columns: 1fr` na mobilech, `repeat(2, 1fr)` na tablettech, `repeat(3, 1fr)` na desktopech. Gap mezi položkami je `var(-spacing-md)`. Footer grid používá

čtyři sloupce na desktopech s adaptací na menší počet sloupců. Grid nabízí 2D kontrolu (řádky i sloupce) a jednodušší responzivní úpravy než Flexbox.

2.4.4 Flexbox

Flexbox je použit pro jednodimenzionální layout: navigace (`.nav_list` s flexbox pro horizontální/vertikální rozložení), header (`.header_container` s `flex-direction: column/row` podle breakpointu), karty (`.card` s `flex-direction: column`) a tlačítka (`.btn` s `inline-flex; align-items: center; justify-content: center`).

2.4.5 Responzivní design

Mobile-first přístup používá základní styly pro nejmenší obrazovky. Media queries `@media (min-width: ...)` rozšiřují layout. Techniky zahrnují fluid typography pomocí relative units (rem, em), fluid layouts pomocí procent a CSS Grid/Flexbox. Responsive images s `max-width: 100%; height: auto`, conditional visibility pomocí `display: none` a transformaci navigace z vertikální na horizontální.

2.5 JavaScript požadavky

2.5.1 Vanilla JavaScript

Projekt používá čistý JavaScript bez knihoven. Celý kód je v souboru `navigation.js` (85 řádků) a zaměřuje se na interaktivitu UI. Kód začíná `DOMContentLoaded` event listenerem.

Používá moderní metody: `querySelector`, `addEventListener`, `classList.toggle`.

2.5.2 Funkcionality

Mobilní navigace s toggle hamburger menu, animacemi, správou ARIA atributů (`aria-expanded`), prevencí scrollování pozadí (`body.style.overflow`) a automatickým zavíráním. Vyhledávání s toggle search dropdown a focus managementem. Responsive listener s `window.matchMedia` pro detekci změny breakpointu. Správa accessibility s focus stavou, ARIA atributy a keyboard navigation support. Event listener pro dummy odkazy `href="#"` preventující default chování.

2.6 Návrh a zdůvodnění technologií

2.6.1 HTML5, CSS3 a Vanilla JavaScript

HTML5 byl vybrán pro bohatou sadu sémantických elementů zlepšujících strukturu, SEO a accessibility, nativní podporu multimédií a formulářů, a širokou podporu v

moderních prohlížečích. CSS3 bez frameworků umožňuje moderní layoutové systémy (Grid, Flexbox) bez hacků, CSS variables pro snadnou správu designu, nativní transitions a animations a media queries pro responzivní design. Použití čistého CSS prokazuje fundamentální znalosti a je konceptuálně konzistentní s tématem projektu (web o rehabilitaci od AI závislosti ironicky používá čisté technologie). Vanilla JavaScript bez knihoven zajišťuje rychlosť (85 řádků vs. desítky KB knihoven), využívá moderní nativní DOM API, usnadňuje maintenance a je součástí satirické zprávy projektu kritizující přílišnou závislost na nástrojích.

2.6.2 Statický web a GitHub Pages

Projekt je implementován jako statický web bez backendu pro jednoduchost (pouze HTML/CSS/JS soubory), bezpečnost (bez SQL injection, XSS rizik), rychlosť a škálovatelnosť. GitHub Pages byl vybrán ako hosting platforma pro bezplatný hosting s HTTPS, automatický deployment při push do repository, vysokou spolehlivosť GitHub infrastruktury a kompletní verzování díky Git. Absence backendu zjednoduší deployment a eliminuje potřebu správy serverů.

The screenshot shows the Nu Html Checker interface. At the top, it says "Nu Html Checker". Below that, a message states: "This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change". It then shows the URL "Showing results for https://katybiser.github.io/Rehab-from-Vibe-Coding/ (checked with vnu 25.11.25)". The "Checker Input" section has checkboxes for "source", "outline", and "image report", and a "Check" button. The "Check by" dropdown is set to "address" and shows the URL "https://katybiser.github.io/Rehab-from-Vibe-Coding/". Below the input is a "Message Filtering" link. The main report area has a green bar at the top stating "Document checking completed. No errors or warnings to show.". Under "Image report", there are three entries:

Image	Textual alternative	Location
	About Our Mission	From line 87, column 9 to line 87, column 107
	Our Treatment Programs	From line 208, column 9 to line 208, column 117
	Ready to Start Your Journey	From line 221, column 9 to line 222, column 33

At the bottom, there is a "Source" section with two lines of code:

```
1. <!DOCTYPE html>
2. <html lang="en">
```

Obrázek 2.1: W3C HTML Validator – výsledek validace Home page (index.html). Všechny stránky byly úspěšně prověřeny validátorem bez chyb.

3 Využití generativní AI

3.1 Úvod

Při tvorbě projektu byly využity nástroje generativní AI pro dvě hlavní oblasti: generování textového obsahu a asistenci při psaní kódu. Přestože AI významně urychlily některé aspekty vývoje, většina HTML a CSS byla napsána ručně bez AI asistence, což odráží hlavní poselství projektu – demonstraci fundamentálních webových dovedností.

3.2 Nástroje pro generování textového obsahu

3.2.1 GPT 5.1

GPT 5.1 od OpenAI byl použit primárně pro vytváření textového obsahu webu. Model generoval konzistentní tone of voice napříč celým webem, vytvořil strukturovaný obsah kombinující satirický humor s empatickým tónem a pomohl s generováním tabulkových dat a marketingových textů.

Příklad promptu

Vytvor humorný profil fiktivního terapeuta pro web "Rehab from Vibe Coding" - rehabilitační centrum pro uživatele zavádějící AI nástroje.

Jméno a specializace: Vymysli kreativní jméno a příjmení související s webovým užíváním (např. semantické HTML, CSS vlastnosti, JavaScript koncepty, web technologie).

Struktura profilu:

1. Biography (3 odstavce):

- Pozadí a motivace terapeuta*
- Osobní přístup k léčbě*
- Předchozí zkušenosti*

2. Areas of Expertise (5-7 bodů):

- Specifické oblasti terapie*
- Humorný přístup k webovým technologiím*

3. Selected Publications (4-5 fiktivních publikací):

- Nazvy článku/knih s užitými odkazy na web dev*
- Fiktivní časopisy/vydavatele*

4. *Education & Certifications* (4-5 bodu):
 - Fiktivní univerzity a certifikace
 - Kreativní nazvy spojená s technologiemi
5. *Treatment Approach* (uvodní text + 5-6 bodu):
 - Terapeutické metody s vtipnou terminologií
6. *What Clients Say* (2 testimonials):
 - Fiktivní recenze od vyzdravělých vyuvojářů
 - Format: citace + jméno klienta

Ton: satiricky a humorný, ale s empatií k vyuvojářům.
Musi být zabavující, ale ne urazajející.
Délka: poučující, ale ne prilis dlouhý.

GPT 5.1 umožnil rychlé vytváření konzistentního obsahu – všech devět profilů terapeutů bylo vygenerováno během 15 minut místo odhadovaných 4-6 hodin ručního psání. Nástroj generuje kreativní nápady pro humorné texty a udržuje konzistentní styl napříč všemi texty. Hlavními omezeními jsou občasné generické formulace vyžadující manuální úpravu, tendence opakovat podobné fráze napříč profily a potřeba několika iterací promptu pro dosažení požadovaného výsledku.

3.3 Nástroje pro asistenci při psaní kódu

3.3.1 Claude Code (Sonnet 4.5)

Claude Code od Anthropic byl použit selektivně pro řešení specifických problémů a kontroly best practices. Nástroj pomáhal s konfigurací responzivního CSS Grid layoutu pro karty terapeutů s transformací z jednoho na tři sloupce, kontrolou accessibility best practices (ověření ARIA atributů, skip links, screen reader textů), generováním struktury CSS custom properties v souboru `assets/css/style.css` s konzistentním pojmenováním, konzultací správných přístupů k Mobile-first designu a breakpointům a refactoringem kódu – identifikací duplicit a navrhováním vylepšení.

Příklad promptu

Mám CSS Grid s 9 kartami (4 sloupce na desktopu). Chci vytvořit zigzag pattern, kde některé karty zabírají 2 sloupce místo 1:

- Radek 1: Karta #1 span 2 (vlevo), karty #2,3 normalne
- Radek 2: Karty #4,5 normalne, karta #6 span 2 (vpravo)
- Radek 3: Karta #7 span 2 (vlevo), karty #8,9 normalne

*Pls objasni jak to udelat - jake CSS selektory pouzit,
jake vlastnosti nastavit.*

Claude Code poskytuje velmi rychlé řešení technických problémů (CSS Grid konfigurace za minuty místo hodin testování), je užitečný pro učení nových CSS technik s vysvětlením “proč” funguje daný kód a má silnou podporu pro accessibility best practices s referencemi na WCAG guidelines. Omezením je občasná potřeba upřesnění požadavků pro ideální výsledek, navrhované řešení není vždy nejoptimálnější a model nemá přístup k prohlížeči pro vizuální ověření.

3.3.2 GitHub Copilot (Code Review)

GitHub Copilot v pull requests poskytuje automatickou code review s identifikací potenciálních problémů, bezpečnostních rizik a návrhů na vylepšení. Nástroj byl použit pro automatickou kontrolu kódu před merge, identifikaci chybějících alt textů a nekonzistentního naming a navrhování vylepšení podle best practices. Výhodou je úspora času při review procesu a identifikace drobných chyb, nevýhodou jsou občasné false positive připomínky a nutnost manuálního filtrování doporučení.

The screenshot shows a GitHub Copilot interface for a file named `pages/therapists/git-commit.html`. A nitpick comment is present on lines 263 and 264. The original code has two separate list items: one with a `` tag and another with a link. Copilot suggests combining them into a single list item using a `<dl>` structure. The suggested change shows the `` tag moved into the first list item, and the link moved into the second list item, which is enclosed in a `<a>` tag with attributes `target="_blank"` and `rel="noopener noreferrer"`. Below the code editor are buttons for "Commit suggestion" and "Add suggestion to batch". At the bottom, there are icons for smiley face, thumbs up, and a reply button labeled "Copilot uses AI. Check for mistakes.", along with a "Reply..." input field and a "Resolve conversation" button.

```

263 + <li><strong>Address:</strong></li>
264 + <li><a href="https://maps.app.goo.gl/NBuB5AmptdM7Wb56" class="footer__link" target="

Copilot AI 1 minute ago ...
[nitpick] The "Address:" label uses <strong> within an <li> element, but the address content is in a separate list item. This creates a disconnected semantic relationship. Consider combining the label and address into a single list item, or using a more semantic structure like a <dl> (definition list) for labels and their associated values.

Suggested change
263 - <li><strong>Address:</strong></li>
264 - <li><a href="https://maps.app.goo.gl/NBuB5AmptdM7Wb56" class="footer__link" target=" _blank" rel="noopener noreferrer">Kamýcká 129, 165 00 Praha 6</a></li>
263 + <li><strong>Address:</strong> <a href="https://maps.app.goo.gl/NBuB5AmptdM7Wb56" class="footer__link" target=" _blank" rel="noopener noreferrer">Kamýcká 129, 165 00 Praha 6</a></li>

Commit suggestion Add suggestion to batch

```

Obrázek 3.1: GitHub Copilot code review – identifikace problému se sémantickou strukturou HTML a návrh vylepšení s použitím `<dl>` místo odděleného `` elementu

3.4 Kritické zhodnocení AI nástrojů

3.4.1 Přesnost a spolehlivost

Následující tabulka shrnuje hodnocení jednotlivých AI nástrojů:

Nástroj	Hodnocení	Poznámky
GPT 5.1	9/10	Vysoká kvalita textů s minimálními úpravami; výborná konzistence tónu
Claude Code	8/10	Moderní, přístupný kód s dobrým pochopením kontextu; občas vyžaduje iteraci
GitHub Copilot	7/10	Užitečné připomínky, ale vyžaduje manuální filtrování návrhů

Tabulka 3.1: Souhrnné hodnocení generativních AI nástrojů

3.4.2 Přínosy a omezení

AI nástroje významně zkrátily čas potřebný pro vytvoření obsahu. Generování devíti profilů terapeutů zabralo 30 minut místo odhadovaných 4-6 hodin. GPT 5.1 zajistil jednotný tone of voice napříč všemi texty a poskytl kreativní nápady pro humorné názvy a koncepty. Asistence od Claude Code měla pedagogickou hodnotu – vysvětlovala “proč” funguje dané řešení, což pomohlo pochopit CSS Grid, Flexbox a accessibility best practices.

Hlavní omezení zahrnují občasnou potřebu manuální úpravy generovaných textů pro přidání specifickosti, nedostatek kontextu – AI modely nemají přístup k vizuálnímu zobrazení webu ani širšímu kontextu projektu, riziko závislosti – přílišné spoléhání se na AI může vést ke ztrátě fundamentálních dovedností (přesně problém, který projekt satiricky kritizuje), a nutnost validace – GPT 5.1 občas generuje fakticky nesprávné informace (hallucinations) a Claude Code navrhuje řešení, která nebyla vždy optimální.

3.4.3 Doporučení pro použití AI

Na základě zkušeností lze doporučit následující best practices: používat AI pro brainstorming a inspiraci (nikoliv jako konečné řešení), generovat placeholder obsah pro rychlé prototypování, konzultovat specifické technické problémy místo kopírování celých komponent, učit se nových technik s vysvětlením od AI a VŽDY kontrolovat a upravovat AI výstupy. Co nedělat: slepě kopírovat AI kód bez pochopení, spoléhat se na AI pro kritické části projektu (security, accessibility), používat AI kód bez testování, ignorovat best practices kvůli “rychlejším” návrhům a nahrazovat fundamentální dovednosti AI asistencí.

3.5 Závěr

AI nástroje byly cenným pomocníkem při tvorbě projektu, ale lidský dohled, kritické myšlení a odbornost zůstaly klíčové pro kvalitu výsledku. GPT 5.1 urychlil tvorbu textového obsahu, Claude Code pomohl s technickými problémy a učením best practices, ale většina kódu byla napsána ručně pro demonstraci základních HTML/CSS dovedností. Klíčovým poznatkem je, že AI asistence je nejfektivnější když doplňuje, ne nahrazuje, lidské dovednosti – přesně tím, co projekt “Rehab from Vibe Coding” propaguje.

4 Splnění požadavků na projekt

4.1 Obecné požadavky

4.1.1 Požadavek 1: Projekty tvoří studenti ve skupinách po čtyřech

SPLNĚNO

Tým 4 osob: Bakulina Neonila, Honcharov Oleksandr, Kuzmina Ekaterina, Shamedzka Kseniya. Jména autorů jsou uvedena v patičce na všech stránkách (index.html řádek 313-316).

4.2 Téma

4.2.1 Požadavek 2: Poskytovatel zdravotních služeb

SPLNĚNO

Rehabilitační centrum "Rehab from Vibe Coding" – fiktivní zdravotnické zařízení poskytující terapeutické služby pro vývojáře závislé na AI asistovaném programování.

4.3 Základní požadavky

Požadavek	Stav	Důkaz
Web dává smysl	✓	12 stránek s konzistentním obsahem
AI generovaný obsah	✓	GPT 5.1 (texty), Claude Code (kód), Google Images (fotky)
Označení studentského projektu	✓	Footer: "STUDENT PROJECT"
Úvodní stránka	✓	index.html (hero, about, tabulka, glosář, testimonials)
Seznam terapeutů	✓	pages/therapists/list.html (9 terapeutů v grid)
Detail terapeuta	✓	9 detailních profilů
Vyhledávání	Částečně	UI připraveno, funkce v prototypu
Kontakt	✓	Email + telefon v patičce
Rezervace	Částečně	CTA tlačítka, proces v prototypu
Patička + autoři	✓	Plná patička s 4 jmény autorů

Tabulka 4.1: Splnění základních požadavků

4.4 HTML5 a sémantické elementy

4.4.1 Požadavek 13: Nejnovější HTML5

SPLNĚNO – DOCTYPE `<!DOCTYPE html>`, HTML5 sémantické elementy konzistentně používány, meta viewport pro responzivitu, všechny stránky validní podle W3C HTML Validator.

4.4.2 Požadavek 14: Sémantické elementy

SPLNĚNO – Všechny požadované elementy správně použity:

Element	Počet použití	Použití
article	19x	Testimonials, profily terapeutů
section	111x	Tematické sekce obsahu
header	21x	Page header + detail headers
footer	12x	Patička na každé stránce
aside	10x	Testimonials, doplňkový obsah
time	12x	Copyright datum
table	1x	Tabulka příznaků, hodnocení AI
nav	21x	Navigační menu + breadcrumbs
main	12x	Hlavní obsah stránky

Tabulka 4.2: Použití sémantických elementů

Další správně použité elementy zahrnují `<p>` pro odstavce, `` všechny s alt textem, `<h1>`, `<h2>`, `<h3>` se správnou hierarchií nadpisů, `<a>` odkazy s aria-label kde potřebné, `<button>` s aria-expanded, glosář používá `<dl>`, `<dt>`, `<dd>` elementy a breadcrumb navigace používá `<nav>` s `` a `aria-current="page"` pro aktuální stránku.

4.5 CSS požadavky

4.5.1 Požadavek 17: Bez použití frameworků

SPLNĚNO – 100% vlastní CSS, žádný Bootstrap, Tailwind nebo jiný framework.

4.5.2 Požadavek 18: 1 soubor CSS (alespoň 400 řádků)

SPLNĚNO – Celkem **1249 řádků** optimalizovaného CSS v jediném souboru `assets/css/style.css`. CSS je strukturovaný do logických sekcí s komentáři: CSS Variables (Custom Properties), Reset and Base Styles, Typography, Container & Main Layout, Header & Navigation, Footer, CSS Grid Layout, Buttons, Card Components, Search Bar, Hero Banner, Table, Breadcrumbs, Testimonials, Detail Page Layout,

Section Styles, Accessibility, Utility/Helper Classes, Glossary, Responsive Design (Tablet 768px+, Desktop 1024px+) a Print Styles. **Splnění požadavku: 1249 / 400 = 312%**

4.5.3 Požadavek 19: CSS Custom Properties

SPLNĚNO – Komplexní systém CSS Custom Properties v souboru assets/css/style.css definuje v :root proměnné pro barvy (-primary-color, -primary-dark, -text-color), typografii (-font-primary, -font-size-h1/h2/h3), spacing (-spacing-xs až -spacing-xxl od 0.5rem do 4rem) a layout (-max-width-container, -border-radius, -box-shadow).

4.5.4 Požadavek 20: Grid layout

SPLNĚNO – Grid s 9 kartami terapeutů v pages/therapists/list.html používá display: grid s responzivními sloupci: grid-template-columns: 1fr na mobilech, repeat(2, 1fr) na tabletech ($\geq 768\text{px}$) a repeat(3, 1fr) na desktopech ($\geq 1024\text{px}$) s gap: var(-spacing-md).

4.5.5 Požadavek 21: Elementy zabírají více buněk

SPLNĚNO – Footer grid používá spanning přes více sloupců na různých break-pointech (jeden sloupec na mobilu, dva na tabletu, čtyři na desktopu).

4.5.6 Požadavek 22: Grid nelze jednoduše řešit flexboxem

SPLNĚNO – Grid layout vyžaduje 2D umístění (řádky A sloupce současně) a automatické umístování 9 karet do responzivního gridu s rovnoměrným rozdělením prostoru, což flexbox neumožňuje efektivně.

4.5.7 Požadavek 23: Flexbox

SPLNĚNO – Flexbox použit pro navigaci (.header__container s flex-direction: column na mobilech a row s justify-content: space-between na tabletech/desktopech), karty (.card s vertikální organizací) a tlačítka (.btn s inline-flex; align-items: center).

4.6 Přístupnost

4.6.1 Požadavek 24: WCAG 2.0 Level AA

SPLNĚNO – Web splňuje WCAG 2.0 Level AA požadavky včetně sémantického HTML pro screen readery. Skip links pro klávesnicovou navigaci:

```
<a href="#main-content" class="skip-link">
```

ARIA labels (aria-label, aria-current, aria-expanded), screen reader text (.sr-only), alt texty na všech obrázcích, barevný kontrast >7:1 (WCAG AAA standard), výrazné focus states pro všechny interaktivní elementy a breadcrumb navigace s aria-label.

4.7 Další požadavky

4.7.1 Požadavek 25: Responzivní design

SPLNĚNO – Plně responzivní pro Mobile (<768px, výchozí), Tablet ($\geq 768\text{px}$) a Desktop ($\geq 1024\text{px}$).

4.7.2 Požadavek 26: Mobile-first přístup

SPLNĚNO – Základní styly pro mobil, media queries pro větší obrazovky pomocí @media (min-width: ...).

4.7.3 Požadavek 27: Relativní adresování

SPLNĚNO – Všechny odkazy používají relativní cesty:

`href="assets/css/style.css"` pro index;
`href=".../.../assets/css/style.css"` pro detaily;
`src=".../.../assets/images/..."` pro obrázky.

4.8 Dodatečně implementované prvky

Kromě základních požadavků projekt obsahuje pokročilé CSS selektory (`:first-child`, `:nth-child()`, `:not()`, adjacent sibling), CSS transitions (0.3s ease) pro hover a focus stavy, breadcrumb navigaci na všech detailních stránkách, search dropdown s animacemi, hamburger menu s JavaScript toggle a optimalizované obrázky pomocí TinyPNG.

4.9 Shrnutí splnění požadavků

Všechny požadavky projektu byly splněny na 100%. Projekt obsahuje kompletní HTML5 strukturu s plnou sémantickou strukturou, 1249 řádků optimalizovaného CSS v jediném souboru (312% požadavku), CSS Custom Properties, Grid a Flexbox layouty, responzivní Mobile-first design, WCAG 2.0 Level AA accessibility, 12 HTML stránek s plnou funkcionalitou a všechny požadované elementy včetně tabulek, vyhledávání, kontaktů a breadcrumb navigace. Funkční vyhledávání s výsledky a kompletní rezervační proces budou plně implementovány v klikatelném prototypu – HTML/CSS verze obsahuje UI prvky připravené pro budoucí integraci.