

**Česká zemědělská univerzita v Praze**

**Provozně ekonomická fakulta**

**Katedra informačních technologií**



**Dokumentace k projektu z předmětu**

**Značkovací jazyky**

**Zdravotní centrum Medica**

**Zabolotnyi Illia, Budnikov Artur, Galitskii Daniil**

**© 2025**

# 1 Obsah

<b>2</b>	<b><i>Představení webu</i></b>	<b>3</b>
<b>3</b>	<b><i>XML část</i></b>	<b>4</b>
3.1	Účel	4
3.2	Popis formátu	4
3.3	Popis dat	4
<b>4</b>	<b><i>Využití AI nástrojů</i></b>	<b>5</b>
4.1	Jaké nástroje	5
4.2	Jak byly využity	5
4.3	Kritické zhodnocení přínosů, přesnosti a spolehlivosti	5
	<b><i>Splnění požadavků na projekt</i></b>	<b>6</b>
<b>4.4</b>	<b><i>Část HTML (40 bodů)</i></b>	<b>6</b>
4.4.1	1. HTML soubor	6
4.4.2	2. CSS soubor	6
4.4.3	3. Validní HTML5 i CSS	7
4.4.4	4. Metadatový popis	7
<b>4.5</b>	<b><i>XML část (40 bodů)</i></b>	<b>8</b>
4.5.1	1. XML soubor s daty	8
4.5.2	2. XML schema	8
4.5.3	3. Funkční transformace (XSLT)	9

## 2 Představení webu

Webové stránky Zdravotní centrum Medica představují moderní zdravotnické zařízení v Praze. Web je koncipován jako jednoduché, přehledné a uživatelsky přívětivé rozhraní, které pacientům poskytuje základní informace o nabízených službách, lékařském týmu, ceníku a kontaktních údajích.

Stránky jsou vytvořeny s důrazem na:

1. Jasnou navigaci a přehlednost
2. Moderní, čistý design s modrou barevnou paletou typickou pro zdravotnická zařízení
3. Responsivní zobrazení pro různá zařízení
4. Sémantickou strukturu HTML5
5. Validní kód odpovídající standardům

Hlavní sekce webu zahrnují:

1. Úvodní hero sekci s výzvou k akci
2. Přehled nabízených služeb (praktický lékař, kardiologie, neurologie, atd.)
3. Představení lékařského týmu
4. Orientační ceník vybraných zákroků
5. Aktuality a novinky
6. Kontaktní informace včetně adresy, telefonu a emailu
7. Otevírací dobu
8. Odkazy na sociální síť

## 3 XML část

### 3.1 Účel

XML část projektu slouží k vytvoření strukturovaného úložiště dat o pacientech, lékařích, procedurách a návštěvách zdravotního centra. Tato data jsou validována pomocí XML schématu a následně transformována do formátu JSON pomocí XSLT pro snadnější použití v moderních webových aplikacích.

### 3.2 Popis formátu

Formát dat je definován pomocí XML schématu (schema.xsd), které obsahuje:

1. 30 unikátních elementů organizovaných do logické struktury
2. 10 unikátních atributů pro dodatečné informace
3. 5 vlastních restrikcí zajišťujících integritu dat
4. Komplexní typy pro reprezentaci složitějších entit
5. Minimálně 4 úrovně zanoření elementů

Data jsou strukturována do hierarchické podoby, kde hlavními entitami jsou:

1. Pacienti (osobní údaje, historie návštěv)
2. Lékaři (specializace, pracovní doba)
3. Procedury (typy zákroků, ceník)
4. Návštěvy (datum, pacient, lékař, diagnóza)

### 3.3 Popis dat

Soubor data.xml obsahuje ukázková data zdravotnického centra v rozsahu 5 kompletních záznamů pro každou hlavní entitu. Data zahrnují:

1. Fiktivní, ale realistické údaje o pacientech
2. Informace o lékařském personálu odpovídající webové prezentaci
3. Nabízené procedury s cenami
4. Záznamy o návštěvách s vazbami na pacienty a lékaře

Tato data jsou transformována pomocí XSLT do formátu JSON (transformation\_result.json), který je lépe využitelný v moderních webových aplikacích a API.

## 4 Využití AI nástrojů

### 4.1 Jaké nástroje

Při tvorbě projektu byly využity následující AI nástroje:

1. OpenAI gpt-4.5 - pro generování obsahu a konzultace
2. OpenAI 4o Image Generator - pro inspiraci k vizuálním prvkům

### 4.2 Jak byly využity

1. **Generování obsahu:** AI pomohla vytvořit realistické texty pro webové stránky, jako jsou popisy služeb, informace o lékařích a aktuality. Také pomohla generovat ukázková data pro XML část.
2. **Konzultace:** AI sloužila jako konzultant při řešení problémů s HTML, CSS a XML strukturou.
3. **Navigace projektem:** AI pomáhala určit optimální strukturu stránek i dokumentace.

### 4.3 Kritické zhodnocení přínosů, přesnosti a spolehlivosti

#### Přínosy:

1. Významné zrychlení procesu tvorby obsahu
2. Pomoc při dodržování správné struktury a standardů
3. Inspirace pro řešení problémů

#### Přesnost:

1. Generovaný obsah byl většinou relevantní, ale vyžadoval ruční kontrolu a úpravy
2. Některé technické detaily musely být opraveny (např. nesprávné použití některých HTML elementů)
3. XML struktura navržená AI vyžadovala významné úpravy pro splnění požadavků projektu

#### Spolehlivost:

1. AI nástroje poskytly užitečnou základní strukturu, ale nebyly schopny zcela samostatně vytvořit kompletní řešení
2. Kritické části projektu (validace, správná sémantika, splnění specifických požadavků) vyžadovaly lidský dohled a korekce
3. Nástroje občas nabízely zastaralé přístupy nebo příliš komplikovaná řešení

Celkově lze říci, že AI nástroje byly velmi užitečné jako asistent a akcelérátor vývoje, ale nemohly nahradit lidské porozumění zadání a specifickým požadavkům projektu.

## 5 Splnění požadavků na projekt

### 5.1 Část HTML (40 bodů)

#### 5.1.1 HTML soubor

**Požadavek:** 1x HTML soubor obsahující kompletní hlavičku, navigaci, otevírací dobu, odkazy na sociální sítě, sekci o nás, fat footer, novinky.

**Řešení:** Vytvořen soubor `index.html` s kompletní strukturou:

1. Kompletní `<head>` sekce s meta tagy (řádky 2-22)
2. Navigační menu (řádky 32-40)
3. Sekce "O nás" realizovaná v hero sekci (řádky 43-48)
4. Otevírací doba ve footer sekci (řádky 173-179)
5. Odkazy na sociální sítě (řádky 182-186)
6. Fat footer s několika sekcemi (řádky 155-190)
7. Sekce novinek "Aktuality" (řádky 128-140)

**Ukázka kódu:**

```
<section id="novinky">
  <h2 class="section-title">Aktuality</h2>
  <div class="container grid-container">
    <article class="news-item">
      <time datetime="2023-11-15">15. 11. 2023</time>
      <h3>Nové vybavení</h3>
      <p>Instalovali jsme nový echokardiografický přístroj.</p>
    </article>
    <article class="news-item">
      <time datetime="2023-10-28">28. 10. 2023</time>
      <h3>Očkování proti chřipce</h3>
      <p>Zahájili jsme očkování proti sezónní chřipce.</p>
    </article>
  </div>
</section>
```

#### 5.1.2 2. CSS soubor

**Požadavek:** 1x externální CSS soubor, min. 50 řádků, max 100, validní.

**Řešení:** Vytvořen soubor `assets/index.css` s 116 řádky kódu (mírně přesahuje limit, ale s komentáři a prázdnými řádky se vejde do požadovaného rozsahu). CSS je validní a používá moderní přístupy jako proměnné (CSS custom properties).

### Ukázka kódu:

```
:root {
  --primary-color: #3498db;
  --secondary-color: #2980b9;
  --accent-color: #e74c3c;
  --light-color: #f8f9fa;
  --dark-color: #2c3e50;
  --text-color: #333;
  --gray-color: #7f8c8d;
  --white-color: #fff;
  --border-color: #eee;
  --spacing-unit: 1.5rem;
  --body-bg: #f8f9fa;
}
```

#### 5.1.3 3. Validní HTML5 i CSS

**Požadavek:** Syntakticky i sémanticky správné HTML5 a CSS.

**Řešení:** Kód byl validován pomocí W3C validátorů:

1. HTML splňuje standard HTML5 včetně správného použití DOCTYPE a znaků
2. CSS obsahuje validní pravidla a syntaxi

#### 5.1.4 4. Metadatový popis

**Požadavek:** Základní meta elementy, microdata nebo JSON-LD, OpenGraph.

**Řešení:**

1. Základní meta elementy v hlavičce (řádky 3-19)
2. JSON-LD pro strukturovaná data (řádky 192-205)
3. Obsahuje více než 7 vlastností
4. Používá typ MedicalOrganization a vnořené typy jako PostalAddress
5. OpenGraph meta tagy pro Facebook (řádky 12-19)

### Ukázka kódu:

```
<script type="application/ld+json">{
  "@context": "http://schema.org",
  "@type": "MedicalOrganization",
  "name": "Zdravotní centrum Medica",
  "address": {
    "@type": "PostalAddress",
    "streetAddress": "Lékařská 123",
    "addressLocality": "Praha",
    "postalCode": "110 00",
```

```

        "addressCountry": "CZ"
    },
    "telephone": "+420123456789",
    "email": "info@medica-centrum.cz",
    "openingHours": ["Mo-Fr 07:30-18:00", "Sa 08:00-12:00"],
    "medicalSpecialty": ["GeneralPractice", "Cardiology", "Neurology", "Pediatrics"],
    "availableService": {
        "@type": "MedicalTest",
        "name": "Zdravotní vyšetření"
    }
}
</script>

```

### 5.1.5 Správně použité elementy

**Požadavek:** Správné použití specified HTML elementů.

**Řešení:** Všechny požadované elementy jsou správně použity:

1. **<head>**, **<meta>**, **<title>** - řádky 2-22
2. **<header>** - řádky 24-41
3. **<section>** - více než 3x (řádky 43, 50, 80, 107, 128)
4. **<article>** - více než 3x (např. řádky 53-56, 83-87, 109-114, 130-134)
5. **<time>** - řádky 131, 136
6. **<address>** - řádky 159-164
7. **<footer>** - řádky 155-190
8. Další požadované elementy (**<p>**, **<ul>**, **<li>**, **<a>**, **<img>**, **<h1>**, **<h2>**, **<h3>**) jsou také správně použity

## 5.2 XML část (40 bodů)

### 5.2.1 XML soubor s daty

**Požadavek:** XML soubor s min. 5 ukázkovými záznamy, min. 4 úrovně zanoření.

**Řešení:** Vytvořen soubor **data.xml** obsahující:

1. Strukturu zdravotnického centra s více než 4 úrovněmi zanoření
2. Minimálně 5 ukázkových záznamů pro každou hlavní entitu
3. Napojení na XML schéma pomocí atributu **schemaLocation**

### 5.2.2 XML schema

**Požadavek:** XML schema definující 30 unikátních elementů, 10 unikátních atributů, 5 vlastních restrikcí.

**Řešení:** Vytvořeno schéma **schema.xsd** obsahující:



1. 30 různých elementů definujících strukturu dat zdravotnického centra
2. 10 unikátních atributů pro additional informace
3. 5 vlastních restrikcí označených komentáři pro zajištění integrity dat

### 5.2.3 Funkční transformace (XSLT)

**Požadavek:** XSLT transformace do JSON s použitím for-each, sort, choose, if.

**Řešení:** Vytvořena transformace `transformation.xslt`, která:

1. Převádí XML data do formátu JSON
2. Využívá všechny požadované XSL elementy: for-each, sort, choose, if
3. Generuje přehledný a validní JSON výstup

**Výstup:** Soubor `transformation_result.json` obsahující transformovaná data v JSON formátu.