

EÖTVÖS LORÁND TUDOMÁNYEGYETEM
TERMÉSZETTUDOMÁNYI KAR

Középiskolások számára készített
„Multimédia a térképeken” című
weboldal modernizálása

SZAKDOLGOZAT
FÖLDTUDOMÁNYI ALAPSZAK
TÉRKÉPÉSZ ÉS GEOINFORMATIKA SPECIALIZÁCIÓ

Készítette:

Koós Katalin

Témavezető:

José Jesús Reyes Nuñez

egyetemi tanár

ELTE Térképtudományi és Geoinformatikai Intézet



Budapest, 2021

Tartalomjegyzék

1. Bevezetés.....	3
2. Weboldal célja.....	4
3. A régi weboldal hiányosságai.....	5
4. A weboldal modernizálása.....	8
4.1. A weboldal tartalmának modernizálása.....	8
4.1.1. A régi weboldal tartalmának bemutatása.....	8
4.1.2. A meglévő tartalom frissítése.....	9
4.1.3. Új tartalom hozzáadása.....	11
4.2. A weboldal informatikai modernizálása.....	14
4.2.1. Kezdeti lépések.....	14
4.2.2. A keretrendszer.....	14
4.2.3. A build rendszer.....	17
4.2.4. Új funkciók.....	18
4.2.5. Mobil kompatibilitás.....	20
4.2.6. A design átalakítása.....	22
4.2.7. Akadálymentesítés fogyatékkal élők számára.....	26
5. Összefoglalás.....	29
6. Summary.....	30
7. Irodalomjegyzék.....	31
8. Köszönetnyilvánítás.....	34
9. Nyilatkozat.....	35

1. Bevezetés

A diplomamunkám célja egy már meglévő térképészeti témájú weboldal modernizálása volt. A régi honlap még 2006-ban készült, ezért sok szempontból már elavult volt, például a webes térképszolgáltatások akkoriban még nem voltak olyan széleskörben használtak, mint manapság és maga a webes térképészet is gyerekcipőben járt. A tartalom frissítése mellett, informatikailag is korszerűsítettem a honlapot, ez többek között abban nyilvánul meg, hogy Single Page Application-ra (SPA) írtam át az eddigi egyszerű HTML oldalt.

A témaválasztásnál fontos szempont volt nálam, hogy olyan szakdolgozattal foglalkozzak, amin szívesen dolgozok és aminél a munka során szerzett tudást később is tudom hasznosítani. Mivel érdekel az informatika, ezért különösen örültem, hogy webfejlesztéssel foglalkozhatok. Ezenkívül maga a honlap témája is tetszett mivel sok érdekes példa van multimédiás térképekre, ami azért is lényeges, mivel a weboldal célközönsége a térképészet iránt érdeklődő középiskolás korúak és ezért különösen fontos, hogy a weboldal amellet, hogy új információkat szolgáltat szórakoztató is legyen.

A dolgozat szerkezeti felépítése a következő: először röviden ismertetem a weboldal célját, majd kifejtem, hogy milyen hiányosságok miatt volt szükséges és ajánlatos az újraalkotása, ezután pedig rátérek a szakdolgozat lényegi részére, a modernizálás főbb lépéseire. Ezt az utolsó részt két kisebb egységre bontottam, a tartalom bővítésére és az informatikai modernizálásra.

2. Weboldal célja

A honlap elsősorban középiskolások számára készült oktatási céllal. A témája nem része a középiskolai kerettantervnek, mivel magáról a térképészetről is csak érintőlegesen van szó földrajz órán, hanem annak a kiegészítése.[1] A térképekre rakható multimédiás tartalmakat mutatja be, a hagyományosan rajzolt képektől kezdve a digitális tartalmakig. A weboldal célja, hogy a térképek iránt érdeklődőknek szórakoztató és érdekes tartalmat biztosítson és emellett még tanítson is.

Az, hogy az oldal középiskolásoknak készült igazából csak abban nyilvánul meg, hogy az ismertető szövegek viszonylag rövidek, tömörek, hogy gyorsan át lehessen futni őket, a honlapot képek, videók színesítik, hogy felkeltsék az érdeklődést és csak annyira mélyen tárgyalják az adott témákat, ami nem igényel előzetes térképészeti tudást.

3. A régi weboldal hiányosságai

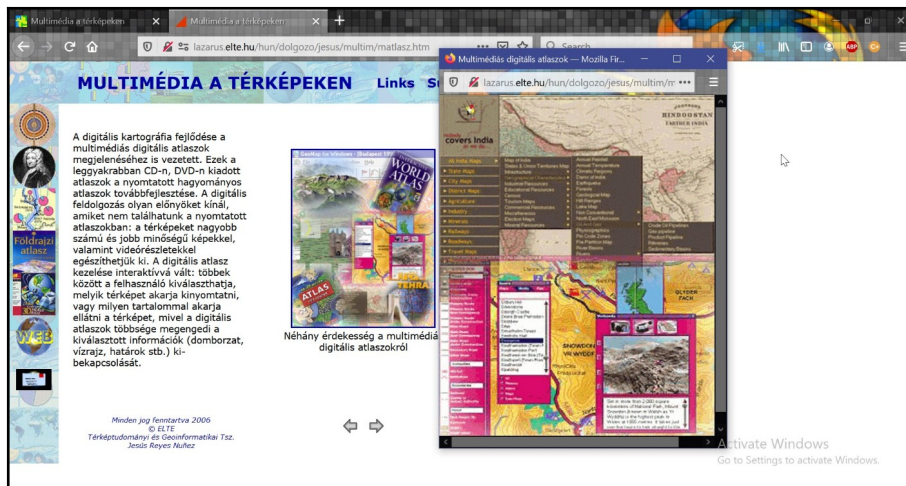
A következőkben azt szeretném kifejtetni, hogy mik voltak azok a dolgok a régi honlapon, amiket mindenképpen módosítani kellett, ha modernizálni akartuk az oldalt. Mivel a weboldal 15 éve készült az akkori technológiával, ez még a HTML5 megjelenése előtt volt, ezért a honlap már informatikailag és tartalmilag is korszerűtlen.

- Az oldal régen egy Flash animációval indult, de ez már nem működött, mivel a böngészők már nem támogatják az *Adobe Flash*-t.[2]
- Ezenkívül fix mérete van az oldalnak, vagyis nem reszponzív, ebből adódóan nem kompatibilis mobilokkal se. Nyilván amikor a honlap készült, akkor ez még kevésbé volt lényeges szempont.
- A navigáció nagyon nehézkes, mivel, ahogy az az 1. ábrán is látszik, az oldalsó menü csak képeket tartalmaz, ezért nehezen átlátható és a témák címei sincsenek kiírva a szövegek fölé.



1. ábra: A régi honlap menürendszere

- A képek általában gif-ként lettek berakva, amik, ahogy az a 2. ábrán is látszik, kattintásra egy külön ablakban nyílnak meg, ami nemcsak a vakok számára, de az átlagos látogatónak is zavaró.



2. ábra: A régi képnézegető

- A 3. ábrán látszik a hivatkozások gyűjteménye, ezek közül a legtöbb már nem létező helyre mutat és a honlap tartalma is elavult. Szó van az oldalon például az ESRI amerikai cég *Geographic Network* hálózatáról, ami azóta már megszűnt.

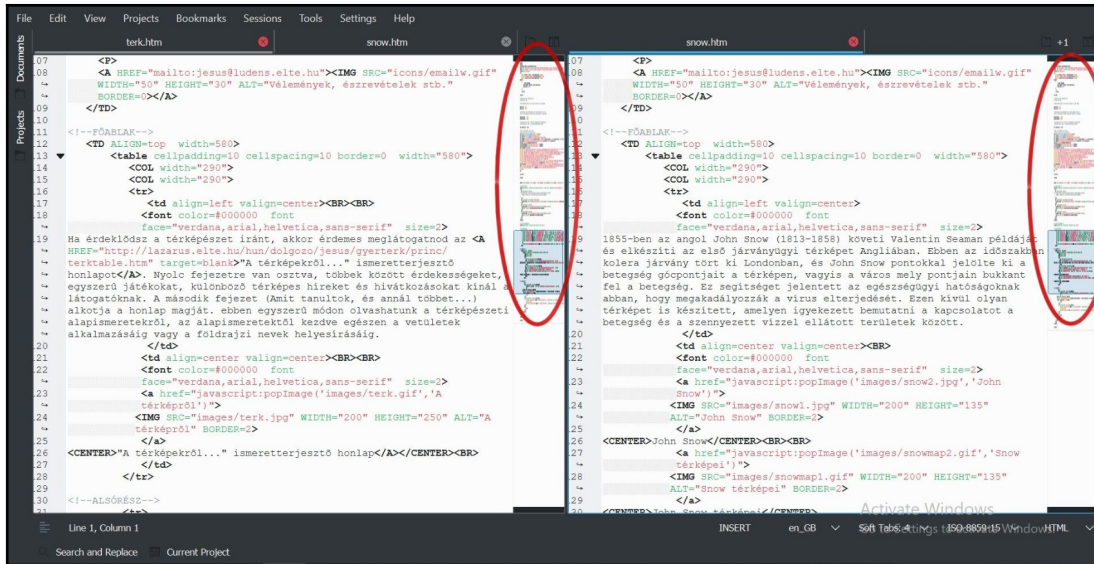


3. ábra: A régi linkgyűjtemény

Ezenkívül mivel a régi oldal sima HTML-ként van megírva, ezért minden témához tartozik egy html kiterjesztésű fájl, vagyis összesen 49 ilyen fájlból áll össze a honlap. Ahogy a 4. ábrán is látszik ezeknek a fájloknak a nagy része ugyanaz a kód, ami a fejléc, a menük és a honlap többi szerkezeti elemének a felépítését határozza meg és többnyire csak a pár mondat tartalom, illetve a képek elérési útvonala különbözik.

Ezzel azon kívül, hogy időigényes megírni, az a probléma, hogy ha később valamit módosítani akarunk, mert rájöttünk, hogy elrontottunk valamit vagy például csak meg

akarjuk változtatni a címnek a színét, akkor azt a jelen esetben 49-szer kell megtennünk. Ez olyankor a legzavaróbb, mint ami a jelen szakdolgozat célja is volt, vagyis amikor modernizálni akarjuk az oldalt. Ez itt nem jelentett problémát, mivel én a régi honlapnak csak a tartalmát használtam fel, de ha ez egy komplexebb weboldal lenne, amit nem akarnánk teljesen újrakezdeni, akkor ez jelentősen megnehezítené a munkánkat.



4. ábra: A régi oldal két html fájljának összehasonlítása

Ezeket az előbbieken felsorolt hiányosságokat próbáltam elsősorban pótolni az új weboldalon, de ezeken kívül sok más is átalakításra került. A szakdolgozat következő fejezetében magára a modernizálás menetére térek rá.

4. A weboldal modernizálása

Az oldal megújításának két fő része volt: a tartalom frissítése és az informatikai modernizálás. A következőkben ezeket fogom részletezni.

4.1. A weboldal tartalmának modernizálása

4.1.1. A régi weboldal tartalmának bemutatása

A régi weboldal hat fő fejezetből áll, ezek a következők: az adatok ábrázolása, az első tematikus térképek, mai tematikus térképek, hagyományos multimédia a térképeken, digitális multimédia a térképeken, multimédia a webkartográfiában.

Az adatok ábrázolásán belül szó van az adatvizualizációról, ez a Katalán Atlasz példáján keresztül van bemutatva, John Graunt adatgyűjtéseiről, William Playfair a grafikus megjelenítés kapcsán kitalált új módszereiről, Alexander von Humboldt munkásságáról, a diagramok népszerűsítéséről, azon belül Ottó Neurath ISOTYPE rendszeréről és Bertin grafikai elméletéről, aki a grafikának hét vizuális változóját határozta meg.

Az első tematikus térképeknél említve van Edmund Halley mágneses deklinációt ábrázoló térképe, Valentine Seaman sárgaláz térképe, William Smith Angliáról és Walesről készült földtani térképe, Humboldt Berghaus atlaszában megjelent hőmérsékletváltozást ábrázoló világtérképe, szó van a francia hatásról, ezen belül az első fekete-fehér kartogramról és az első pontszórásos térképről, és be van mutatva John Snow járványügyi térképe is.

A mai tematikus térképészetnél van egy „Tudományok a térképeken” című bevezető téma, ezután szó van a jelekkel való ábrázolásról, a felületek ábrázolásáról, az izovonalakról, a térképre rakható diagramokról, a kartogramokról és torzított kartogramokról, a pontokkal való ábrázolásról, és a mozgásvonalakról.

A „Hagyományos multimédia a térképeken” című témán belül öt alfejezet van, ezek címei: „A mai multimédia előtt ...”, „Mi a multimédia?”, „Térképek a multimédiában”, „Multimédia az atlaszokban” és „Az első animációs térképek”, amiben a Frank Capra által rendezett „Why We Fight” propaganda sorozat animált térképeiről van szó.

A „Digitális multimédia a térképeken” című téma egy általános bevezetővel kezdődik, ezután szó van a digitális multimédia kezdeteiről, itt említve van a *Canada Land Inventory* és az 1960-as évek első számítógépes animációi, ezután következik a digitális térképek alfejezet, ahol a térképkészítésre alkalmas szoftverek vannak röviden ismertetve, majd a multimédiás digitális atlaszokkal záródik ez a nagy téma.

A hatodik nagy témakör a „Multimédia a webkartográfiában” a „Térképek a weben” című bevezető fejezettel kezdődik, azután következik az „Interaktivitás a webatlaszokban”, majd pedig a *Geographic Network*, a Vakegér vaktérképes játék, a „Lázár térkép települései” című honlap és az „A térképekről ...” című ismeretterjesztő honlap bemutatása.

4.1.2. A meglévő tartalom frissítése

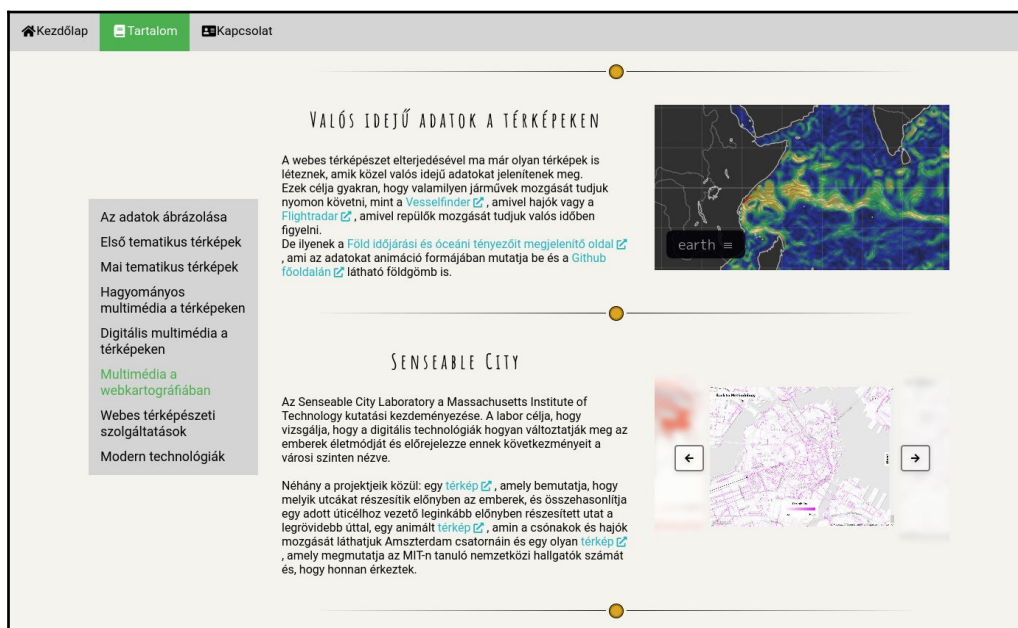
Egyes témáknál csak a megfogalmazáson változtattam, vagy a címeket írtam át általánosabbra, például a „William Playfair” címet lecseréltem a „Grafikus megjelenítés új módszerei”-re, mivel az első cím valószínűleg nem sokat mond az olvasónak, míg a másodikból lehet rá következtetni, hogy miről lesz szó. Ezenkívül kiszedtem azokat a részeket, amik már elavultak, mint például említve volt a *Freehand* szoftver, aminek még 2003-ban leálltak a fejlesztésével és az is, hogy a digitális atlaszokat leggyakrabban CD-n vagy DVD-n adják ki, ami már nyilván nem állja meg a helyét.[3] Voltak olyan már meglévő témák, ahova új hivatkozásokat raktam be, mint például az *USGS National Map Viewer*-t az „Interaktivitás a webatlaszokban” című részhez.

Olyan témák is voltak, amiket teljesen lecseréltem, például az *ESRI Geographic Network* hálózata már megszűnt, ennek a helyét az *ArcGIS Online* vette át. Ebben írtam a *ArcGIS Online* néhány alkalmazásáról és egy videót is készítettem, amiben ezekkel készített térképek vannak bemutatva. A „Lázár térkép települései” című témát is kiszedtem, mivel a honlap, amiről itt szó volt már nem elérhető.

Mind a már meglévő, mind az új témák között vannak, amik jobban és vannak, amik kevésbé kapcsolódnak magához a honlap témájához. Ettől függetlenül törekedtem arra, hogy minél szorosabban kötődjenek a témák a multimédiás térképekhez. Emiatt a meglévő tartalomból is volt, amit kihagytam, például „A térképekről ...” ismeretterjesztő

oldalt, ami eddig a webkartográfia témakörön belül volt, pedig magához a multimédiás térképekhez viszonylag kevés köze van.

A kivett témák helyett beraktam újakat, ezek az 5. ábrán is látható „Senseable City”, ami az MIT kutatási kezdeményezése, ebben néhány projektjükéről írtam és a „Valós idejű adatok a térképeken” című téma, amiben főként példákat sorolok fel, többek között a *Vesselfindert* és a *Flightradar24*-t is említtem.



5. ábra: A két új téma a webkartográfián belül

A képekből próbáltam, minél érdekesebbeket keresni és olyanokat, amik jól tükrözik a bemutatni kívánt témakört. Volt ahol videókat vagy iframe-eket raktam be képek helyett, főleg ott, ahol modernebb technológiákat mutatok be, ezzel is kicsit változatosabbá téve az oldalt. Ahol tudtam saját képeket, videót készítettem.

További módosítás volt, hogy a Linkek és a Súgó menüket megszüntettem, mivel ezekre már nem volt szükség. A hivatkozásokat ehelyett magába a szövegbe szúrtam be, mert úgy gondolom, így jobban felkeltik az érdeklődést. A súgó pedig ahogy az a 6. ábrán is látszik a honlapon való navigációt magyarázta, az új honlapon viszont ez már egyértelmű.



6. ábra: A régi weboldal Súgó menüje

4.1.3. Új tartalom hozzáadása

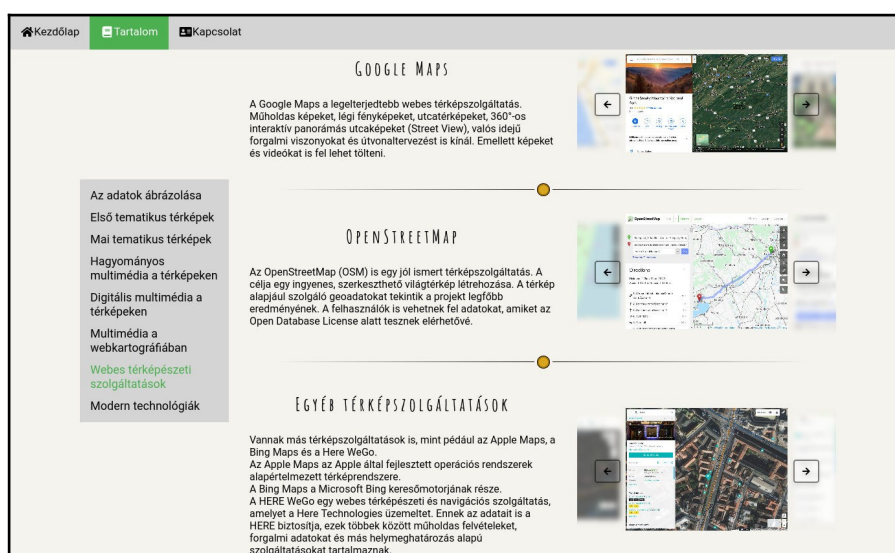
Témavezetőm kérésére a honlap tartalmát olyan új témákkal is kiegészítettem, mint a webes térképszolgáltatások vagy a kiterjesztett valóság, más néven AR (Augmented Reality) technológiák. Ezenkívül csináltam egy kezdőlap és egy kapcsolat menüpontot is.

A kezdőlapon, ahogy az a 7. ábrán is látszik, írtam egy rövid bevezetőt a honlaphoz, hogy mi a témája, kiknek és minek a keretében készült és, hogy milyen a szerkezeti felépítése. Továbbá hivatkoztam a másik két honlapra, amik a 'weboldal sorozat' részét képezik. Ezek közül a geoinformatikai témájú is ebben az évben került megújításra szakdolgozat keretében. Nálam még a régi oldalra mutat a link, mivel az új honlapok még nincsenek a végleges helyükre kirakva, de ezt még lehet később módosítani. A hivatkozások beszúrását az oldalra azért is célszerűnek tartom, mivel ezek a honlapok egy sorozat részét képezik és ilyen módon sokkal egyszerűbb őket megtalálni is.

A térképszolgáltatások témánál először egy általános bevezetőt írtam. Ezután, ahogy az a 8. ábrán is látszik, külön tárgyaltam a fontosabbakat, mint a *Google Maps*, *OpenStreetMap*, míg a kevésbé ismerteket, mint az *Apple Maps*, *Bing Maps* és a *Here WeGo* összevontam egy témába. A szövegeket a *Wikipédia* alapján írtam, de a képeket én készítettem.



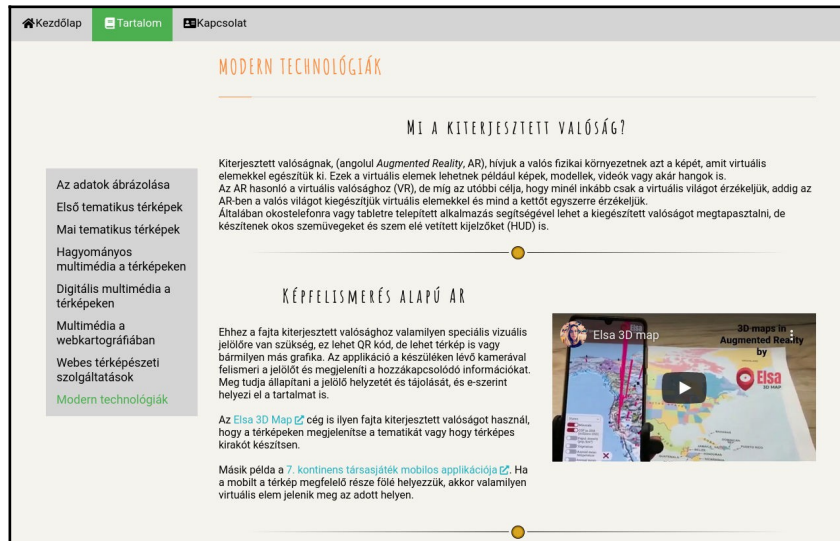
7. ábra: Az új kezdőoldal



8. ábra: Térképszolgáltatások téma

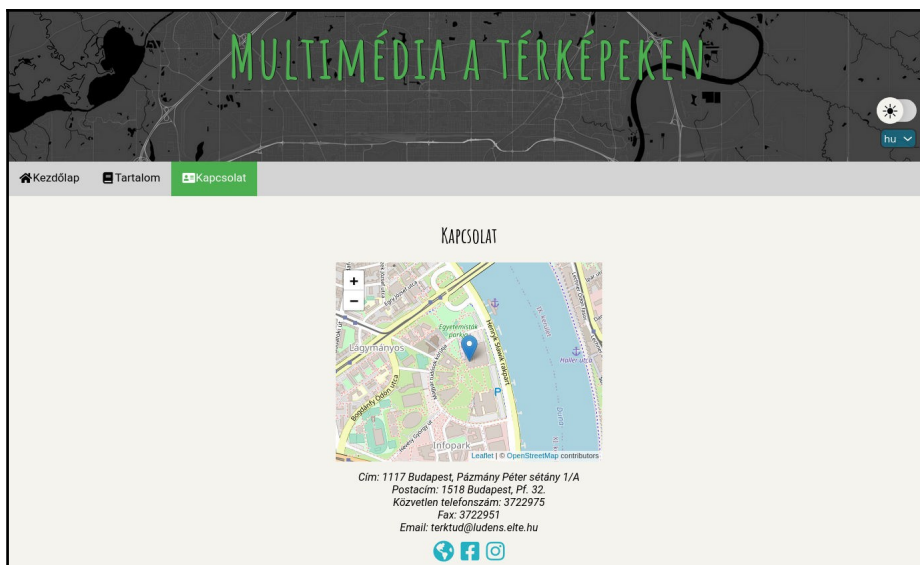
Ahogy ez a 9. ábrán is látszik, az AR technológiánál is írtam egy általános bevezetőt. Ezután kétfajta kiterjesztett valóságot mutattam be: a képfelismerésen alapulót és a helymeghatározáson alapulót. Mindkettőre írtam példákat is, az előbbire az *Elsa 3D Map*-et [4], témavezetőm javaslata alapján és a 7. *kontinens* társasjátékot, az utóbbira pedig a *Star Walk 2* alkalmazást, ami ugyan a térképekhez csak lazán kapcsolódik, de jól bemutatja a helymeghatározás alapú AR-t. Ezután egy külön témában írtam az *Augmented Reality Sandbox*-ról, ami ugyan nem térkép, de szorosan kapcsolódik a térképészethez. Ennél homokból épített formákra valós időben vetítik rá projektorral a szintvonalakat és a

magassági színezést. Ennek a nagytémának a feldolgozásához főként Nikola Yonov: „School Atlas with Augmented Reality” című publikációját vettem alapul.[5] Ezenkívül az AR Sandbox leírását a honlapján található részletes útmutató alapján írtam.[6]



9. ábra: Modern technológiák téma

A Kapcsolat oldalra, mint ahogy a 10. ábra is mutatja, egy *Leaflet*-tel készült térképet raktam be, amin az intézet helyét egy jelölő mutatja, ha erre rákattintunk, akkor a koordináták is kiírásra kerülnek. Ez alatt vannak felsorolva az intézet elérhetőségei, amiket a hivatalos honlapról vettem át és a hivatkozások az intézet új weboldalára, valamint a közösségi médiákra (*Facebookra* és *Instagramra*).



10. ábra: Az új kapcsolat oldal

4.2. A weboldal informatikai modernizálása

4.2.1. Kezdeti lépések

A weboldal informatikai megújítása kapcsán úgy döntöttem, hogy nem a régi oldalt fogom átalakítani, hanem egy újat csinálok. A weboldal készítéséhez a *Vue.js* frontend keretrendszert és *Webpack* build rendszert használtam. A kezdeti állományokat, KartoffelCheetah nevű felhasználó Vue Base nevű projektjét, a *GitLab*ról töltöttem le.[7] Ez egy egyszerű tanító projekt, ami ezeknek a programoknak a kezdeti használatát segíti. Tartalmazza már például a *Webpack* konfigurációs fájlját és a *Stylus* fájlt, amibe a stílusok egy részét írtam.

4.2.2. A keretrendszer

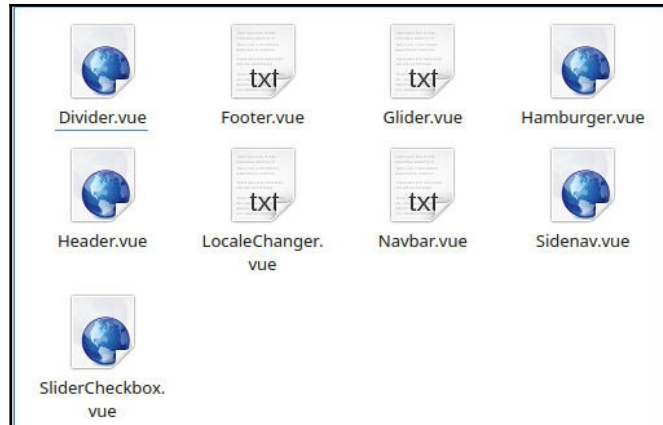
A honlap készítésénél a *Vue.js* keretrendszert alkalmaztam. Ezt Evan You szoftverfejlesztő dolgozta ki 2014-ben és ő és a csapata azóta is fejlesztik. A *Vue 3*, amit én a szakdolgozathoz használtam, 2020 szeptemberében lett kiadva.[8]

Olyan neves cégek weboldalai készültek *Vue*-val mint a *WizzAir* és a *Nintendo* néhány európai weboldala, de hogy térképészeti témájút is említsek a világ metál együtteseinek a térképe is elkészült ezzel a keretrendszerrel.[9]

Azért ez a keretrendszer mellett döntöttem, mert ez egy frontend keretrendszer, vagyis nincs úgynevezett „backendes” függősége. Így nincs abból probléma, ha például a szerveren más *PHP* vagy *Python* verzió van telepítve, mint amivel a weboldalt készítem és ezek nem kompatibilisek egymással. Emellett a frontend keretrendszerek közül ez az, ami viszonylag könnyen tanulható és a dokumentációja is részletes és átlátható.[10] A *Vue* tanulására elsősorban a dokumentációt használtam.

Előnye, hogy moduláris vagyis a weboldalt komponensekre lehet osztani. Érdekes a többször használt logikai egységek számára külön komponenseket létrehozni, mint például a header, footer vagy a navigációs menü, így ezeknek a kódját nem kell minden oldalon újra és újra leírni, ezzel is elkerülve a redundanciát vagyis az azonos kódrészletek ismétlését. Ennek az elkerülése azért lényeges, mert egyrészt növeli a kód méretét, másrészt nagyon nagy mértékben nehezíti a módosításokat, javításokat. A 11. ábrán az általam használt kilenc komponens látható. Ezek *Vue* kiterjesztésű fájlok, a felépítésükről

a szakdolgozat következő részében fogok írni. A komponenseket úgy lehet beilleszteni a kódba, mint bármilyen HTML tag-et. Például az elválasztó használatakor csak annyit kell leírnom, hogy `<Divider />` és az elválasztó kódja be lesz illesztve az adott helyre.



11. ábra: Az általam használt komponensek

A Vue keretrendszerben a Vue-router bővítmény segítségével lehet Single Page Application-t (SPA) készíteni, az új weboldal is ilyen. Az SPA lényege, hogy csak egy HTML fájlunk van és a honlap tartalma dinamikusan újíródik az új adattal, vagyis nem teljesen új HTML oldalak töltődnek be. [11]

Fontos tulajdonsága a Vue-nak, hogy reaktív vagyis, ha a Javascript változók értéke módosul, akkor újraszámolja azokat. Ennek a segítségével lehetett megcsinálni a fordító és a témaválasztó funkciót is az oldalra. Ezekről a szakdolgozat „Új funkciók” című fejezetében fogok írni.

Másik előnye ennek a keretrendszernek, hogy lehetővé teszi Javascript kód írását a HTML-be. Néhány példa erre a Vue dokumentációjából és a honlapom kódjából:

- Dupla kapcsos zárójelek közé írhatunk Javascript kódot, aminek az értéke fog megjelenni a HTML-ben. A Vue reaktivitása miatt, ha ez az érték megváltozik, akkor a DOM (Document Object Model) is frissül.

```
<span>Message: {{ msg }}</span>
```

Így írtam ki például a menüpontok nevét:

```
<router-link>{{ $t('navbar.home') }}</router-link>
```

- Míg az előző módszer csak sima szöveget írt ki, addig a "v-html" használatával HTML kódot tudunk kiírni.

```
<span v-html="rawHtml"></span>
```

Én ezt a honlap tartalmának a megjelenítésénél használtam:

```
<div v-html="$t('webcartography.interactivity')"></div>
```

- A "v-if" arra használható, hogy egy adott elemet csak akkor jelenítsünk meg, ha egy bizonyos feltétel teljesül.

```
<p v-if="seen">Now you see me</p>
```

Ezt én az oldalsó menünél használtam, ami csak akkor látható, ha a Tartalom oldalon vagyunk:

```
<Sidenav v-if="contentMenu" />
```

```
contentMenu() {
    return this.$route.name !== 'home' && this.$route.name !==
        'contact'
}
```

- A "v-on:click" vagy "@click" valamilyen elemre, például egy gombra helyezhető rá, ez egy függvényt vár argumentumként és az elemre való kattintáskor azt végrehajtja.

```
<a @click="doSomething"> ... </a>
```

Ezt több helyen is használtam, például a mobilos menüben a Tartalom menü ki- és becsukására:

```
<button @click="subMenuToggle">
```

```
  {{ $t('navbar.content') }}
```

```
</button>
```

```
subMenuToggle() {
    this.subMenu = ! this.subMenu;
}
```


4.2.3. A build rendszer

A honlap készítéséhez a *Webpack* build rendszert használtam. Ez kényelmes fejlesztői környezetet biztosít és segít a hibák korai észrevételében azzal, hogy kiírja a hibaüzeneteket. Ezenkívül a konfigurációs fájljába meg lehet adni, hogy bizonyos kiterjesztésű fájlokat milyen loaderekkel töltsön be, vagyis hogyan kezelje ezeket. A Vue fejlesztői csináltak egy saját loadert, ez a *Vue Loader*, ebben le van írva, hogy a Webpack, mit csináljon a vue kiterjesztésű fájlokkal. [12] [13]

A vue kiterjesztésű fájlok három részből épülnek fel:

- `<template>` tagok közé írjuk a HTML-t (HyperText Markup Language)
- `<style>` tagok közé a stílust, ez lehet CSS (Cascading Style Sheets) is
- `<script>` tagok közé a Javascriptet

Ennek a fájlszerkezetnek az az előnye, hogy egy fájon belül vannak az összetartozó kódok.

Egy egyszerű példa erre az általam készített honlap Footer komponensének a felépítése:

```
<template>
  <footer :class="$style.footer">
    <i>&copy; 2021 {{t('footer')}}</i>
  </footer>
</template>
<script>
  export default {
    name: 'Footer',
  }
</script>
<style lang="styl" module>
  .footer {
    padding: 20px;
    text-align:center;
    font-size: 15px;
  }
</style>
```

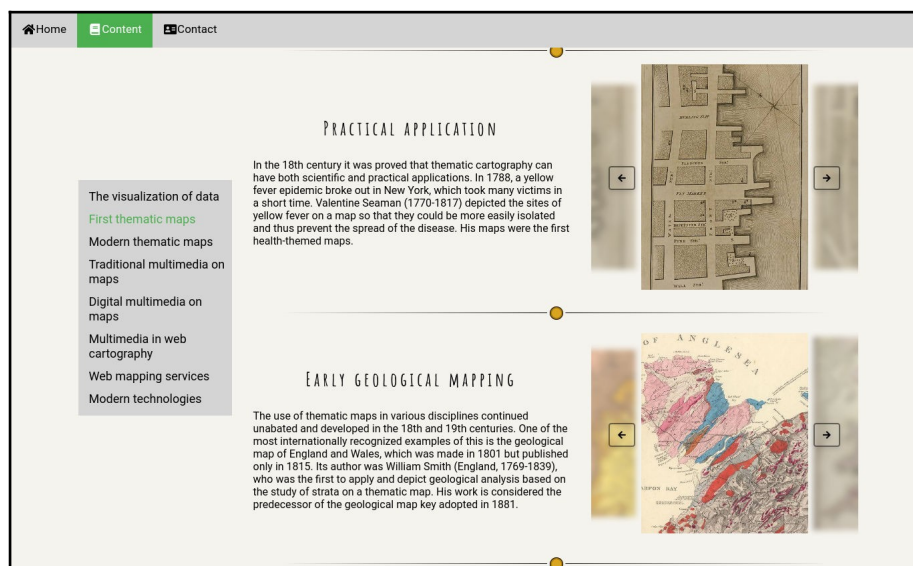
4.2.4. Új funkciók

Az általam készített weboldalra két új funkciót készítettem: a nyelvválasztót és a témaváltót. Ebben a fejezetben ezeket fogom bemutatni.

Nyelvválasztó:

Egyes egyszerű HTML-ben írt oldalakon előfordul olyan, hogy a nyelvválasztó nem csak lefordítja az oldal szövegét, hanem más tartalmat jelenít meg, esetleg maga az oldal kinézete is megváltozik. Ez amiatt fordulhat elő, mert ilyenkor a fordításoknak egy külön HTML oldalt kell írni. Mivel én Single Page Applicationt készítettem, ezért ez az új honlapon nem fordul elő.

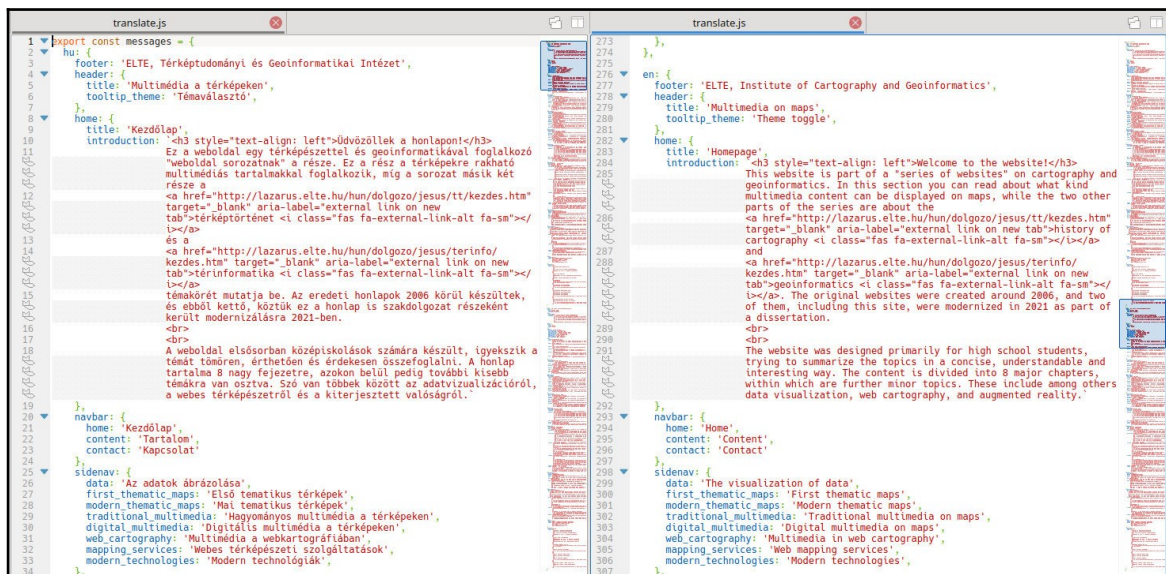
Az új weboldalon a tartalom a magyar mellett angolul is elérhető. A nyelvet az oldal jobb oldalán lévő legördülő menüből lehet kiválasztani. Az aktuálisan kiválasztott nyelvet egy változóban tároljuk. A honlapon a nyelv kiválasztásakor, a Vue reaktivitásából fakadóan, ezen változó értéke frissül és a szöveg az adott nyelven jelenik meg. A 12. ábrán látszik, hogy az angolra való fordítás során tényleg csak a szöveg változik, a honlap kinézete nem.



12. ábra: A weboldal angol nyelven

A fordításhoz a *Vuei18n* bővítményt használtam. Ennek a használatakor a lefordított szövegnek nem kell egy teljesen új HTML dokumentumot létrehozni, hanem egy Javascript fájlba írjuk mind a magyar, mind az angol nyelvű szövegeket és a HTML-be

csak változókat illesztünk be.[14] Ahogy a 13. ábrán is látszik ennek az a hátránya, hogy az egy fájlba ömlesztett szöveg kicsit nehezen kezelhető.



13. ábra: A magyar és az angol fordítás egymás mellett

Mint a keretrendszert tárgyaló fejezetnél is látható volt kapcsos zárójelek közé írva lehet Javascript kódot illeszteni a HTML-be. Ezekben belül meghívjuk a "\$t" nevű függvényt, ami kiírja az argumentumként kapott szöveghez tartozó fordítást.

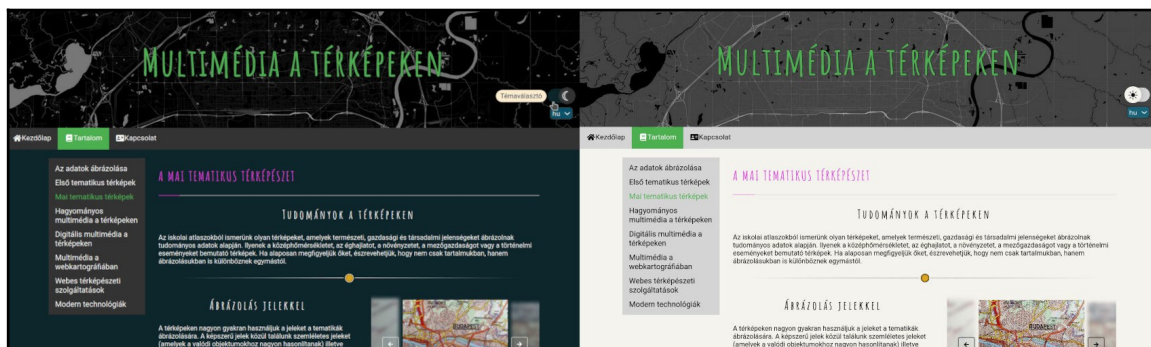
```
<div>{{ $t ('webcartography.interactivity') }}</div>
```

Témaváltó:

Egy sötét és egy világos témát csináltam az új honlaphoz. A témaváltót Musthaq Ahamad: „Dark and Light theme switcher using CSS variables and pure JavaScript” című a *Medium* weboldalon megjelent cikke alapján készítettem.[15] A 14. ábrán látható a csúszka, amivel a témák között lehet váltani, ennek a kinézetét is az épp aktuális téma határozza meg. A világos témánál látható nap és a sötét témánál látható hold egy-egy UTF-8 karakter. A 15. ábrán a honlap látható sötét, illetve világos témában.



14. ábra: A témaváltó csúszka



15. ábra: A sötét és a világos téma

A témák színeinek a tárolására négy CSS változót hoztam létre. Ezek elsősorban a honlap fő színét, a menürendszerek színét, a betűszínt és a képnézegetőben a nyilak színét határozzák meg. Ezeknek nem konkrét színt adtam meg, csak a változó nevét és attól függően, hogy melyik téma van érvényben az a változó más színt jelent. A színeken kívül a honlap háttér is változik témaváltáskor: világos témánál szürke, sötétnél fekete háttérkép van beállítva.

Az alább kódrészletben a CSS változók és értékeik láthatóak:

```
.theme-light {
  --primary-color: #f5f3ed;
  --secondary-color: #d4d4d4;
  --font-color: black;
  --arrow-color: rgba(0, 0, 0, 0.5);
}

.theme-dark {
  --primary-color: #0C262B;
  --secondary-color: #333;
  --font-color: white;
  --arrow-color: rgba(255, 255, 255, 0.5);
}
```

4.2.5. Mobil kompatibilitás

Témavezetőm kérése volt, hogy az új weboldal mobilokon is jól nézzen ki és ezt én is fontosnak tartottam, ezért az oldalt reszponzívvá alakítottam.

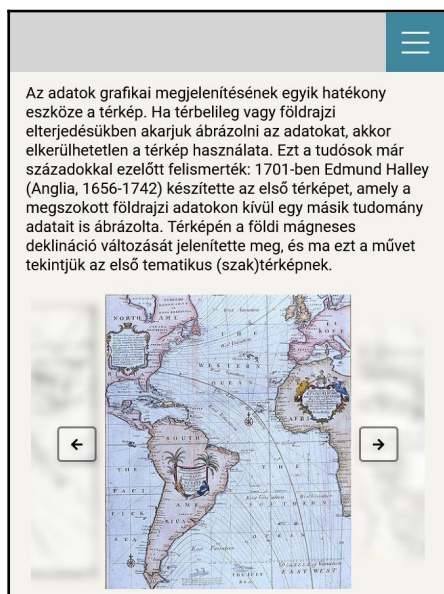
Ennek érdekében úgynevezett „media query”-ket raktam a kódba, ezek segítségével csökkentem a címek betűméretét a kisebb képernyőkön, és a képeket, videókat, iframe-eket

a szöveg alá csúsztatom egy bizonyos méret alatt. Figyeltem arra is, hogy ezeknek a mérete rezponzívan változzon.

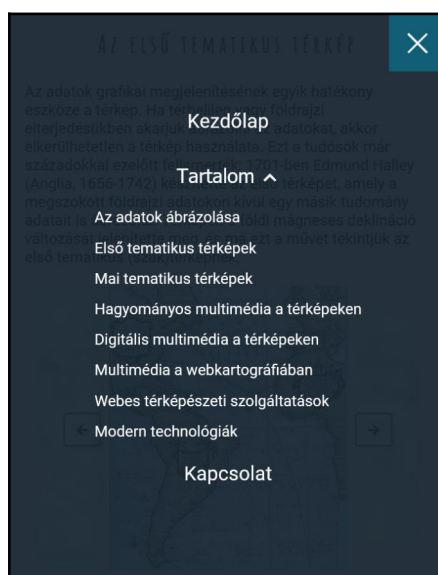
A következő kódrészletnél az látszik, hogy amikor a képernyő szélessége 1200 pixelnél nagyobb, akkor a „flex-direction” értéke „row” vagyis sor, egyébként pedig „column” vagyis oszlop. Ez határozza meg, hogy a képek a szövegek mellett vagy alatt helyezkednek el.

```
.container {  
  width: 100%;  
  display: flex;  
  @media (max-width: 1200px) {  
    flex-direction: column;  
  }  
  flex-direction: row;  
}
```

A legnagyobb változtatás a menürendszerénél történik. A kis képernyőjű eszközökön a menüket lecserélem egy hamburger típusú menüre, amire kattintva egy úgynevezett „overlay” fedi el a képernyőt, és ennek a felületén látjuk a menüpontokat. A hamburger menü vonalai animálva vannak, hogy abban az esetben, ha a menü meg van nyitva, X-et formázzanak. A 16. ábrán látható a menü zárt állapotban, a 17. ábrán pedig nyitott állapotban.



16. ábra: Hamburger menü zárt állapotban



17. ábra: Hamburger menü nyitott állapotban

4.2.6. A design átalakítása

A honlap designját vagyis kinézetét teljesen átalakítottam. A következőkben ezeket a változtatásokat fogom részletezni.

A régi weboldal szerkezete, ahogy a 18. ábrán is látszik, úgy épül fel, hogy felül egy keskeny sávban van a cím és mellette két menüpont, baloldalon vannak a fő témák menüpontjai képekként megjelenítve, alul van a lábjegyzet, a lábjegyzet mellett vannak a nyilak, amikkel előre felé és visszafelé tudunk lépkedni a tartalomban, középen pedig maga a tartalom jelenik meg.



18. ábra: A régi weboldal szerkezeti felépítése

Az új weboldal szerkezetét a 19. ábra mutatja. Itt a cím nagyobb betűmérettel egy vastagabb sávban van, ennek a sávnak háttérképet is beállítottam, itt található a téma- és a nyelvválasztó is, ez alatt van a főmenü a három menüponttal, mobilos nézetben pedig a hamburger menü, az oldal alján található a lábjegyzet, középen pedig maga a tartalom. Ha nem mobilos nézetben vagyunk és a Tartalom oldalt nézzük, akkor a bal oldalon megjelenik egy oldalsó menü is, ez tartalmazza a nagytémák neveit. A menüket úgy hoztam létre, hogy mindig látszódjanak a képernyőn, ezzel is könnyítve az oldalon való navigációt.



19. ábra: Az új weboldal szerkezeti elemei

A régi honlapon minden altéma egy külön oldalon szerepelt, ezeket témavezetőm kérésére összevontam egy oldalra. Ez azért is előnyösebb így, mivel egyrészt egy-egy alfejezet általában csak néhány mondatból álló rövid szöveget tartalmaz, de ezeket egy oldalon felsorolva már megfelelő mennyiségű tartalom van ahhoz, hogy egy átlagos méretű képernyőn is jól nézzen ki a honlap, másrészt ezzel a menürendszer is leegyszerűsödik, mivel nem kell a főtémákon belül még az altémákhoz is menüket létrehozni. Az oldal tartalmánál maximum szélességet is beállítottam, mivel ha ezt nem tettem volna meg, akkor széles monitorokon nagyon el lett volna nyújtva a tartalom.

A cím mögött látható háttérképeket az *Alvar Carto* ingyenes háttérképkészítőjével csináltam.[16] Ez telefonos háttérképkészítőként van hirdetve, de meg lehet adni a kép felbontását vagyis nagyobb monitorokra is lehet vele háttérképet készíteni. Itt több különböző területeket mutató képet is csináltam, majd ezeket kipróbáltam az általam készített weboldalon, hogy meg tudjam állapítani, melyik néz ki a legjobban. Végül a kanadai Edmontont ábrázoló mellett döntöttem. Ebből van egy fekete és egy szürke változat, ami attól függően változik, hogy sötét vagy világos téma van-e beállítva.

A betűtípusokat a *Google Fonts*-ról importálom, ami egy ingyenesen felhasználható betűtípus-gyűjtemény.[17] A honlapon két típust használok: egy könnyen olvasható,

egyszerűt a témák leírásánál, beállításoknál, menüpontoknál és egy egyedibb, ceruzás írásra hasonlító címeknél.

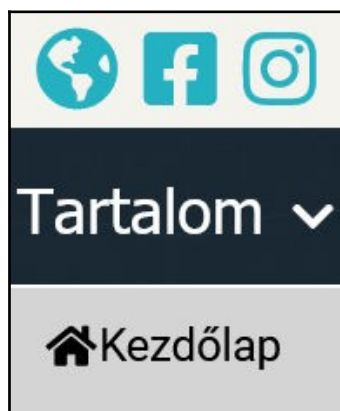
A weboldalon lehet váltani sötét és világos téma között. Mint már említettem a témaváltóról szóló fejezetben, négy szín változik témaváltáskor: ezek elősorban a háttér, a menürendszerek, a betűk és a képnézegető nyilainak a színét határozzák meg. Ezeket igyekeztem úgy megválasztani, hogy mindig jól láthatóak, olvashatóak legyenek a honlap elemei.

A témák közé elválasztókat illesztettem be, hogy ezek egymástól jobban elkülönüljenek és a honlap kinézetét is egyedibbé, érdekesebbé tegyék. Ennek a designját az *uiCookies* weboldalon felsorolt példák között találtam, ezt kismértékben át kellett alakítanom, hogy mind sötét, mind világos témánál is jól nézzen ki. Erről az oldalról származik még a nagy témák címeinek a designja is.[18] A 20. ábrán az elválasztó kinézete látható sötét illetve világos témánál.



20. ábra: Az elválasztó kinézete sötét és világos témánál

Az ikonok, amiket a menüben és a hivatkozások mellett használok és amik a közösségi médiákat jelölik a *Font Awesome* ikonkészlet oldaláról importálom.[19] Ezekre a 21. ábrán látható néhány példa.



21. ábra: Néhány példa a felhasznált ikonokra

Mivel a honlap egyes témáihoz több képet is akartam rakni, ezért a képek megjelenítéséhez a *Glide.js* slidert használok. Ezen is kellett pár módosítást végrehajtanom, hogy az általam készített honlapon is megfelelően működjön. Az új honlap képnézegetője a 22. ábrán látható. A régi weboldalon a képek váltása animálva volt, az új oldalon viszont nyilakkal tudunk közöttük lépkedni, ez azért is jobb, mert így mi döntjük el, hogy mennyi ideig akarunk egy képet nézni. A Glide-nál meg lehet adni, hogy hány kép látszódjon egyszerre a képnézegetőben, nálam ez úgy van beállítva, hogy a szomszédos képekből is látszódjon egy kis részlet. A középen elhelyezkedő aktív képet úgy emelem ki, hogy kissé megnövelem a méretét és az oldalsó képeket homályossá teszem. [20]



22. ábra: Az új honlap képnézegetője

A HTML5, ami a legújabb HTML verzió, 2008-ban jelent meg. Fő célja, hogy biztosítsa a multimédia támogatását a böngészőkben és hogy a webes alkalmazásokhoz ne legyen szükség pluginek (például *Adobe Flash*) telepítésére. Olyan új HTML elemek váltak elérhetővé ezzel, mint a `<video>`, `<audio>` vagy a `<canvas>`. Ezenkívül a HTML5 az SVG-t (Scalable Vector Graphics) is támogatja. [21] Ennek köszönhetően tudtam az új honlapra videókat és SVG kiterjesztésű ikonokat berakni. Erre a régi honlapon még nem volt lehetőség, mivel az két évvel a HTML5 megjelenése előtt készült.

4.2.7. Akadálymentesítés fogyatékkal élők számára

„*The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect.*” Tim Berners-Lee

A fenti idézet magyarra körülbelül így fordítható: „*A web ereje az egyetemességében rejlik. Elengedhetetlen szempont, hogy fogyatéktól függetlenül mindenki hozzáférhessen.*”

Azért kezdtem ezt az alfejezetet Tim Berners-Lee, a World Wide Web feltalálójának az idézetével, mivel jól összefoglalja, hogy miért különösen fontos a webes környezetben az akadálymentesítés. [22]

A webfejlesztésben akadálymentesítés alatt azt értjük, amikor a weboldalakat megpróbáljuk minél több ember számára egyenlően elérhetővé, használhatóvá tenni. Ennek kapcsán általában a fogyatékkal élőkre szoktak gondolni, de valójában az akadálymentesítés azokra is vonatkozik, akiknek lassú az internet kapcsolata vagy mobil eszközöket használnak. Számukra is biztosítani kell, hogy használni tudják a weboldalunkat. [23]

A honlap készítése során próbáltam a WCAG (Web Content Accessibility Guidelines) előírásait követni, ami egy standard a webes tartalmak akadálymentesítésére. A weboldal nincs teljesen akadálymentesítve, de törekedtem rá, hogy ahol ez nem okozott túl nagy nehézséget, ott ezeket az előírásokat minél inkább figyelembe vegyem. [24]

Az akadálymentesítés alkalmazása a honlapon:

Szemantikus HTML elemek használata: Ez azt jelenti, hogy az adott célnak megfelelő HTML elemeket használjuk. Az általam készített weboldalon erre egy jó példa a mobilos menü Tartalom menüpontja, ami a funkciója szerint egy gomb, mivel erre kattintva meg lehet jeleníteni és el lehet tüntetni az almenüt, ezért itt `<button>` HTML elemet használok, nem pedig általános `<div>` elemet. Ez azért lényeges mert így a képernyőolvasók (screen reader) is gombként azonosítják a menüt, illetve a `<button>` elem tabbal is kijelölhető, ami fontos a vakok számára, mivel ők általában a billentyűzettel navigálnak. Ezenkívül az úgynevezett „heading” típusú HTML elemek a hierarchiájuknak megfelelően vannak használva: a `<h1>` a honlap címét, a `<h2>` a fő fejezetek címeit és a `<h3>` az alfejezetek címeit jelöli.

Alternatív szövegek írása: Az „alt” attribútummal alternatív szövegeket tudunk megadni. Képeknél szokták használni, ez a szöveg akkor látszódik, ha valami okból kifolyólag nem lehet a képet megjeleníteni, például lassú internetkapcsolat miatt. Ezenkívül információt biztosít a képről azoknak, akik képernyőolvasót használnak. Az új weboldalon minden képnél megadtam egy rövid szöveget, ami tömören leírja, hogy mi látható rajta.

Cím megadása: A „title” attribútummal plusz információt adunk meg az adott HTML elemről. Ezt is felolvassák a képernyőolvasók, így ezzel is tudunk a vakok számára információt biztosítani egy adott elemről. Én például az iframe-eknél adtam meg címet.

ARIA (Accessible Rich Internet Applications) attribútumok megadása: Ezekkel plusz információt biztosíthatunk a weboldal egyes részeivel kapcsolatban a látássérült felhasználók számára. Például a Divider komponensnél megadtam a „role” attribútumnak a „separator” értéket, ez alapján a képernyőolvasók felismerik, hogy ez az elem elválasztásra szolgál:

```
<template>
  <div :class="$style.wrapper" role="separator">
    <div :class="[$style.divider, $style.transparent,
      $style.dot]"></div>
  </div>
</template>
```

Aria-label attribútum megadása: Ennek az attribútumnak szöveges értéket adunk meg, amiben leírjuk, hogy mi a célja egy adott HTML elemnek. A hivatkozásoknál mindig megadtam az aria-label értékének, hogy a linkek új tabon nyílnak meg. Ez különösen a látássérültek számára fontos, mivel például a figyelmeztetés nélkül felugró ablakok megzavarhatják őket az oldalon való navigációban. Ezenkívül megadtam aria-labelt az <input> típusú elemeknél is, például a mobilos menünél, hogy egyértelműbbé tegyem a céljukat:

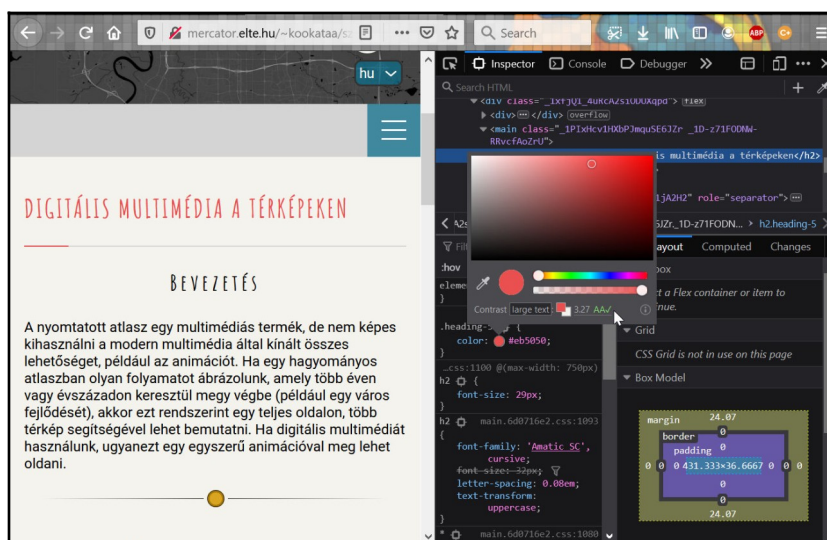
```
<a href="https://elsa3dmap.com/" target="_blank"
aria-label="external link on new tab">Elsa 3D Map</a>
```

A HTML dokumentum nyelvének megadása: Ez azért is lényeges, mert a képernyőolvasók ebből tudják, hogy milyen nyelvű a honlap és ez alapján tudják lefordítani, ha szükséges. Emiatt én is megadom a HTML dokumentum elején a „lang”

attribútum értékét és ezt az alapján változtatom, hogy a felhasználó milyen nyelvet választ ki a nyelvválasztóban.

Értelmezhető hivatkozások: Igyekeztem jól leíró hivatkozás szövegeket használni és elkerülni a túl rövid, keveset mondó hivatkozásokat. Ezt nem mindig tudtam megoldani, mert nem akartam, hogy túl hosszúak legyenek a linkek.

Színkontraszt: a Firefox böngésző fejlesztői környezetében ellenőriztem, hogy elég nagy-e a színkontraszt ahhoz, hogy mindenki számára jól olvashatóak legyenek a szövegek. A 23. ábrán látható ennek a folyamata. Mivel az új weboldalon lehet váltani sötét és világos téma között, ezért nem mindig tudtam úgy kiválasztani a színeket, hogy mindkét témánál megfelelő legyen a kontraszt, de figyeltem arra, hogy legalább az egyik témánál mindig elég nagy legyen.



23. ábra: Cím kontrasztjának ellenőrzése

Végül a weboldalt teszteltem a Firefox beépített akadálymentesítés eszközével is, ami figyelmeztet az oldal azon részeivel kapcsolatban, amelyek gondot okozhatnak a képernyőolvasóknak vagy nem megfelelő kontraszttal rendelkeznek. Ezek a figyelmeztetések alapján még elvégeztem néhány javítást.

Az új weboldal még nincs a végleges helyén, ideiglenesen az alábbi címen érhető el:

<http://mercator.elte.hu/~kookataa/szakdoga/dist/>

5. Összefoglalás

A szakdolgozatom feladata egy 15 évvel ezelőtt készült weboldalt modernizálása volt. Ennek a keretében frissítettem és kibővítettem a weboldal tartalmát, például olyan új témákról írtam, mint a webes térképszolgáltatások és a kiterjesztett valóság. Ezenkívül a honlapot átírtam Single Page Applicationre vagyis a weboldal tartalma dinamikusan újra íródik, nem pedig teljesen új oldalak töltődnek be. Az új honlapot reszponzívvá vagyis mobil kompatibilissé alakítottam, ezért a kisebb képernyőkön is jól néz ki. Emellett a designt is teljesen átalakítottam, modernebb kinézetűvé tettem. Olyan új funkciókat építettem az új oldalba, mint a téma- és a nyelvválasztó. A weboldalt részlegesen akadálymentesítettem a fogyatékkal élők számára, vagyis most már azok is tudják használni az oldalt, akik a honlap szövegének a felolvasására képernyőolvasót használnak.

A feladat végrehajtása során fontos volt számomra, hogy korszerű technológiákat alkalmazzak, ezért döntöttem amellett, hogy a Vue.js frontend keretrendszert és a Webpack build rendszert fogom használni. Ezen kívül törekedtem rá, hogy a kód minél átláthatóbb legyen, hogyha később az oldal továbbfejlesztésére lenne szükség, akkor minél könnyebb legyen értelmezni azt.

A honlap modernizálása ugyan megtörtént, de még vannak további fejlesztési lehetőségek. A legnyilvánvalóbb a tartalom további bővítése, a honlap még több képpel, videóval való kiegészítése. A képeknél egy nagyító funkció is hasznos lenne és tovább lehet fejleszteni a designt is, például logót készíteni a honlapnak. Esetleg más idegen nyelvekre is le lehet még fordítani az oldalt és az angol fordítást is lehet még pontosítani.

Azonban a honlap már most is kielégítően működik és a tőle elvártaknak teljes mértékben megfelel. Összességében a szakdolgozat feladatát elvégeztem, a weboldalt sikeresen modernizáltam.

6. Summary

The task of my dissertation was to modernize a website that was made 15 years ago. As part of this, I updated and expanded the content of the site, for example, I have written about new topics such as web mapping services and augmented reality. In addition, I rewrote the website to Single Page Application, which means that the content of the website is dynamically rewritten and not completely new pages are loaded. I made the new website responsive, i.e. mobile compatible, so it looks good on smaller screens too, and I also completely remade the design to make it look more modern. I have added new features like the theme and language selector. I have made the website partially accessible for people with disabilities, meaning that those who use a screen reader can now also use it.

In performing this task, it was important for me to use state-of-the-art technologies, so I decided to use the Vue.js frontend framework and the Webpack module bundler. In addition, I tried to make the code as readable as possible, so that if the site needed to be improved later, the code would be easy to interpret.

Although the website has been modernized, there is still room for improvement. The most obvious is adding more content or even more images and videos to the site. A magnifier would also be useful for images and the design could be even further developed, such as creating a logo for the site. The website could be translated into other foreign languages and the English translation could also be refined.

However, the site is already working satisfactorily and fully meets its expectations. Overall, I completed the task of my dissertation and successfully modernized the website.

7. Irodalomjegyzék

[1] Kerettanterv a gimnáziumok 9–12. évfolyama számára

https://www.oktatas.hu/koznevelas/kerettantervek/2020_nat/kerettanterv_gimn_9_12_evf,

Utolsó látogatás: 2021.05.09

[2] Adobe Flash Player leírása a Wikipédián

https://en.wikipedia.org/wiki/Adobe_Flash_Player#End_of_life, Utolsó látogatás:

2021.05.10

[3] Adobe FreeHand leírása a Wikipédián

https://en.wikipedia.org/wiki/Adobe_FreeHand, Utolsó látogatás: 2021.05.16

[4] Az Elsa 3D Map cég weboldala

<https://elsa3dmap.com/>, Utolsó látogatás: 2021.05.16

[5] Yonov, N. (2019): School Atlas with Augmented Reality

<https://www.researchgate.net/publication/>

[334379242_School_Atlas_with_Augmented_Reality](https://www.researchgate.net/publication/334379242_School_Atlas_with_Augmented_Reality) Utolsó látogatás: 2021.05.11.

[6] Reed, S. (2014): Augmented Reality Sandbox Facilitator's Guide

<https://arsandbox.ucdavis.edu/technical-resources-2/>, Utolsó látogatás: 2021.05.11.

[7] KartoffelCheetah felhasználó Vue Base projektje a GitLabon

<https://gitlab.com/KartoffelCheetah/vue-base/-/tree/webpack>, Utolsó látogatás: 2021.04.19

[8] Vue.js keretrendszer leírása a Wikipédián

<https://en.wikipedia.org/wiki/Vue.js>, Utolsó látogatás: 2021.04.19

[9] Vue.js-el készített weboldalak listája

<https://madewithvuejs.com/websites>, Utolsó látogatás: 2021.04.19

[10] A Vue.js keretrendszer dokumentációja

<https://v3.vuejs.org/guide/introduction.html>, Utolsó látogatás: 2021.04.19

[11] Alves, R. (2021): What is a Single Page Application(SPA)?

<https://www.outsystems.com/blog/posts/single-page-application/>, Utolsó látogatás:
2021.05.16

[12] Webpack build rendszer főoldala

<https://webpack.js.org/>, Utolsó látogatás: 2021.05.16

[13] A Vue Loader dokumentációja

<https://vue-loader.vuejs.org/#what-is-vue-loader>, Utolsó látogatás: 2021.05.16

[14] A Vuei18n bővítmény dokumentációja

<https://vue-i18n.intlify.dev/guide/>, Utolsó látogatás: 2021.04.19

[15] Ahamad, M. (2019): Dark and Light theme switcher using CSS variables and pure JavaScript

<https://medium.com/@haxzie/dark-and-light-theme-switcher-using-css-variables-and-pure-javascript-zocada-dd0059d72fa2>, Utolsó látogatás: 2021.05.15

[16] Az Alvar Carto cég főoldala

<https://alvarcarto.com>, Utolsó látogatás: 2021.05.15

[17] A Google Fonts főoldala

<https://fonts.google.com/>, Utolsó látogatás: 2021.05.16

[18] Rai, S. (2021): 25+ Creative CSS Divider Examples With Tutorials (Code Snippets) & Previews

<https://uicookies.com/css-divider/>, Utolsó látogatás: 2021.05.15

[19] A Font Awesome ikonkészlet főoldala

<https://fontawesome.com/>, Utolsó látogatás: 2021.05.16

[20] A Glide.js dokumentációja

<https://glidejs.com/docs/>, Utolsó látogatás: 2021.05.15

[21] A HTML5 leírása a Wikipédián

<https://en.wikipedia.org/wiki/HTML5>, Utolsó látogatás: 2021.05.15

[22] A World Wide Web Consortium (W3C) akadálymentesítéssel kapcsolatos leírása

<https://www.w3.org/standards/webdesign/accessibility>, Utolsó látogatás: 2021.05.16

[23] Akadálymentesítéssel kapcsolatos rész a Mozilla dokumentációjából

https://developer.mozilla.org/en-US/docs/Learn/Accessibility/What_is_accessibility,

Utolsó látogatás: 2021.05.15

[24] WCAG akadálymentesítési útmutató

<https://www.w3.org/WAI/standards-guidelines/wcag/>, Utolsó látogatás: 2021.04.19

8. Köszönetnyilvánítás

Ezúton szeretném köszönetemet nyilvánítani témavezetőmnek, José Jesús Reyes Nuñeznek segítőkézségéért és a honlap tartalmi bővítésével kapcsolatos javaslataiért.

Testvéremnek, Koós Andrásnak a programozásban való segítségnyújtásáért.

Végül, de nem utolsósorban köszönöm az egész családomnak, hogy a tanulmányaim során mindig támogattak.

9. Nyilatkozat

Nyilatkozat

Alulírott,Koós Katalin..... nyilatkozom, hogy jelen szakdolgozatom teljes egészében saját, önálló szellemi termékem. A szakdolgozatot sem részben, sem egészében semmilyen más felsőfokú oktatási vagy egyéb intézménybe nem nyújtottam be. A szakdolgozatomban felhasznált, szerzői joggal védett anyagokra vonatkozó engedély a mellékletben megtalálható.

A témavezető által benyújtásra elfogadott szakdolgozat PDF formátumban való elektronikus publikálásához a tanszéki honlapon

HOZZÁJÁRULOK

NEM JÁRULOK HOZZÁ

Budapest, 2021. május 15.

.....Koós Katalin.....
a hallgató aláírása