

EÖTVÖS LORÁND TUDOMÁNYEGYETEM
TERMÉSZETTUDOMÁNYI KAR

Településnevek adatbázisa összehasonlító névrajzi keresőhöz

SZAKDOLGOZAT
FÖLDTUDOMÁNYI ALAPSZAK
TÉRKÉPÉSZ ÉS GEOINFORMATIKA SPECIALIZÁCIÓ

Készítette:
Krizsán Helga

Témavezető:
Dr. Ungvári Zsuzsanna
adjunktus
ELTE Térképtudományi és Geoinformatikai Intézet



Budapest, 2022

Tartalomjegyzék

1.	Bevezetés	3
2.	Előzmények	4
2.1.	A korábbi honlap felépítése	4
3.	Vízrajzinév-kereső adatbázisának bővítése.....	6
3.1.	Adattáblázatok létrehozása	6
3.2.	Előkészületek	6
3.3.	Adattáblázatok elkészítése	7
3.4.	A táblázatok beillesztése a phpMyAdmin adatbázisba.....	9
4.	SHAPE fájl	10
4.1.	SHAPE fájl kiválasztása	10
4.2.	SHAPE fájl kezelése	10
4.2.1.	Városok SHAPE fájl szerkesztése	10
4.	A keresési lehetőségek fejlesztése	12
5.1.	A már meglévő keresések fejlesztése	12
5.2.	A keresési lehetőségek kiegészítése a városokkal	12
6.	SHAPE fájlok beillesztése a weboldalba	17
6.1.	KML fájl elkészítése	17
6.2.	A térképi megjelenítés	17
6.3.	Nagyítás kiválasztott elemre.....	19
7.	A honlap módosításai.....	20
7.1.	Évszámok megjelenítése a glóbusz menüknél.....	20
7.2.	Városnévre keresés módosítása	20
7.3.	Városnevek sorrendje	20
8.	Új keresések és funkciók hozzáadása a honlaphoz	22
8.1.	Az első új keresés.....	22
8.1.1.	Az első új keresési funkció programozása	23
8.2.	A második új keresés	24
8.2.1.	A második új keresési funkció programozása.....	26
8.3.	Új funkció programozása	26
9.	Érdekességek.....	29
10.	A végleges weboldal elérhetősége	30
11.	Összefoglalás.....	31
12.	Hivatkozások	32
13.	Köszönetnyilvánítás.....	33

1. Bevezetés

A dolgozatom a tanszéken már korábban megírt diplomamunkák folytatása és bővítése. Egy weboldal továbbfejlesztése, adatbázisának és funkcióinak kibővítése. Az első Szabó Virág 2010-es diplomamunkája [1] egy névgyűjtemény harminc glóbuszról, melyek Excel táblázatokba rendezve tartalmazták a tengerek, tengeráramlások és fokok neveit. Pár évvel később Beszkid Arthur 2017-es szakdolgozata [2] keretében hozott létre egy interaktív webes keresőfelületet ehhez az adatbázishoz. A honlap címe *Vízrajzinév-kereső* volt. A honlapon követni lehet harminc glóbuszon a tengernevek változásait az 1840-es évektől egészen napjainkig. A honlapot később Rapcsán Klaudia kiegészítette a fokok és áramlások nevének vizsgálási lehetőségével a korábban elkészült Excel táblázat alapján. Kuris Zoltán 2021-es dolgozatában [3] a névadatbázis (tavak, folyók) bővítésével, keresőmotorjának fejlesztésével és további keresési folyamatok létrehozásával foglalkozott. Az én dolgozatom pedig ennek az adatbázisnak és honlapnak a továbbfejlesztése és kiegészítése városok névadatbázisával.

2. Előzmények

2.1. A korábbi honlap felépítése

A honlap felépítése viszonylag egyszerű, négy nagy részből áll. A fejlécből, ami az oldal címét tartalmazza, továbbá jobb oldalt négy sort: a projektről szóló rövid bemutató linkjét, a tanszéki honlapra mutató linket, a tanszék koordinátáit tartalmazó Google Maps linket, valamint a honlap készítőit és idejét tünteti fel. A második rész a menüsáv, amely alatt két további rész különíthető el. A bal oldalon jelenik meg a keresés eredményét mutató táblázat, jobb oldalon pedig a kiválasztott földgömb 3D-s modellje. Itt jelennek meg a kiválasztott tenger, fok, tengeráramlás, tó vagy folyó KML-ek is.

Az oldal menüsávja több részre van felosztva. Az első sorban található keresősávba tudunk a létrehozott adatbázisban keresni. Amikor a keresődobozban szövegrészetet írunk be, mellette a legördülő menüben ki tudjuk választani, hogy mit szeretnénk keresni. Keresni lehet név, illetve gömb azonosító alapján. A Gömb azonosítók nevű lenyíló ablakban meg tudjuk nézni a különböző glóbuszok adatait, ha pedig rákattintunk valamelyik gömbre, akkor a honlap egy új lapot megnyitva átirányít a kiválasztott gömb Virtuális Glóbuszok Múzeumában [4] lévő webes felületére, ahol további adatokat lehet megtudni a glóbuszról. A 2007-ben Márton Mátyás és Gede Mátyás által létrehozott virtuális tárlatban bemutatásra kerülnek az említett 30 glóbusz mellett további magyar vonatkozású föld- és éggömbök is. A menüsáv jobb oldalán található öt kapcsolóval lehet a különböző KML rétegek között váltani.

Ha a keresési mezőbe beírunk egy nevet és elindítjuk a keresést, akkor a keresés eredményeként megjelenik egy táblázat. A táblázatban szerepeltek a keresett név régi neve, mai neve, magyar és angol névváltozatokkal, valamint a gömbhöz tartozó információk egy része és egy rétegre váltás gomb. A Régi név oszlopban lévő elemekre rákattintva az adott elemre közelít rá a földgömbön és megjelenik egy felugró ablak az elem mai magyar és angol nevét megjelenítve. Ha pedig rákattintottunk a gömbhöz tartozó információkra, akkor a honlap itt is átirányít a Virtuális Glóbuszok Múzeuma (VGM) weboldalára.

A második sorban található keresősávok segítségével meg lehet vizsgálni, hogy egy adott gömbön a vizsgált rétegek közül (ezek: tenger, fok, áramlás, tó, folyó) egyet kiválasztva, milyen nevek fordulnak elő. Ehhez tartozik két darab legördülőmenü, amelyek közül az elsőnél ki lehet választani a vizsgálni kívánt glóbuszt. A második legördülőmenüben pedig a keresni kívánt rétegcsoporthoz. A keresés eredményeként pedig megkapjuk az adott gömbön előforduló földrajzi nevek régi, mai magyar és angol nevét (*1. ábra*).

A harmadik sorban látható legördülő menük segítségével meg lehet vizsgálni a mai elnevezés alapján rákeresve, hogy egy adott földrajzi névnek, milyen régi megnevezésű névváltozatai fordulnak elő a különböző gömbökön. A kereséshez szintén két darab legördülőmenü tartozik. Az első esetben ki lehet választani a kívánt keresési réteget (tenger, fok, áramlás, tó, folyó). Amikor kiválasztottuk a keresési réteget, akkor a második legördülőmenü automatikusan feltöltődik a kiválasztott rétegen előforduló földrajzi elemek mai megnevezéseivel. Amikor pedig kiválasztjuk a keresni kívánt mai névváltozatot, eredménynek megkapjuk azt a táblázatot, amiben az adott név összes előforduló névváltozatát meg lehet vizsgálni, valamint azt, hogy az adott régi névváltozat mely glóbuszon fordul elő (2.ábra). Ezeket a legördülő menüket egészítem ki a továbbiakban, hogy a városokra is tudjunk keresni az adott 30 glóbuszon.

1.ábra: Rétegekre keresés glóbuszok alapján

2.ábra: Rétegekre való keresés

3. Vízrajzinév-kereső adatbázisának bővítése

A dolgozatom egyik fő témája a weboldal adatbázisának bővítése volt további adattáblázatokkal. A legfontosabb dolog először az volt, hogy a témavezetőmmel együtt eldöntsem, milyen neveket fogok táblázatba gyűjteni. A honlapról a domborzat és a városnevek még hiányoznak. Véleményem szerint a glóbuszokon megjelenő városokra keresés az hasznos dolog, főleg azok számára, akik érdeklődnek a glóbuszok iránt. Régebben mindig szerettem volna, ha létezik egy olyan honlap, ahol könnyedén meg lehet találni a városokat bármelyik glóbuszon, valamint látni, hogy változtak a városok nevei az évszázadok/évtizedek során. Ezért választottam a városokkal való bővítést. Ezekután elkezdtem a glóbuszokon lévő városneveket táblázatokba gyűjteni.

3.1. Adattáblázatok létrehozása

A városneveket két külön Excel táblázatba kezdtem el kigyűjteni. Az egyik táblázatban a glóbuszokon előforduló névváltozatokat, a másikba pedig a városnevek mai magyar és angol névváltozatait gyűjtöttem ki. Mind a két táblázatban rögzítettem a kontinenseket és külön jeleztem a fővárosokat is.

3.2. Előkészületek

Legelső lépés a már korábban említett Virtuális Glóbuszok Múzeuma (VGM) weboldaláról letölteni annak a 30 glóbusznak a KML fájljait, melyeket egyesével át kell néznom a gyűjtéshez. Haladási szabályokat hoztam létre magamnak, amelyek segítettek abban, hogy minden gömböt és kontinenst átnézzek és ne maradjanak ki nevek a felsorolásból. Az első szabály a gömbök átnézésének sorrendje volt. Először azokkal a gömbökkel kezdtem, amelyek a korban legközelebb állnak hozzánk, így a régebbi gömbökön később már sokkal jobban tudtam tájékozódni és egyszerűbb volt a régi neveket beazonosítani. A gyűjtést éppen ezért az 1950-es évek utáni gömbökkel kezdtem, azok közül is a 32-es ID-vel rendelkező, 1978-as glóbussszal, hiszen ez volt a legrészletesebb. A következő nagy csoport az 1901-1938 közötti gömbök voltak, ezek között volt négy darab ugyanolyan névrajzzal rendelkező és egy, csupán néhány nevet tartalmazó glóbusz. Az utolsó nagy csoport az 1867-1897 közötti gömbök voltak, itt szintén előfordultak ugyanolyan névrajzzal rendelkező glóbuszok, melyek megkönnyítették a munkámat. A városok kontinenshez tartoznak ezért rögzítettem minden névhez, hogy melyik kontinensen találhatóak az adott nevek. Emiatt a következő haladási szabály a kontinenssorrend volt: Európa, Ázsia, Afrika, Ausztrália, Észak-Amerika, Közép-

Amerika és Dél-Amerika. A legutolsó szabály pedig az országok sorrendje volt, amely minimálisan változott néhány glóbuszon a városnevek sűrűsége alapján. Természetesen a gyűjtés közben több országtérképet és atlaszt [5][6] kellett használnom, valamint a fővárosok listáját is pontosítanom kellett. Nagy segítség volt a Nations Online Project weboldala [7], ahol kontinens- és országtérképek találhatóak angol nyelven fővárosokkal és nagyobb városokkal együtt.

3.3. Adattáblázatok elkészítése

A városokhoz tartozó két táblázatba tehát a korábban említett gömb- és kontinenssorrendben haladtam. A két táblázat közül elsőként a „varos” elnevezésű táblázatot készítettem el. A táblázatban hat oszlopba tagolva írtam fel a gömbökön előforduló névváltozatokat. Az oszlopok megnevezései: „varos_id”, „gomb_id”, „varos_varos_id”, „leolvasott_varosnev”, „varos_kontinens”, „fovaros”. A „varos_id” oszlopba írtam minden egyes gömbön előforduló névváltozatnak az egyedi ID azonosítóját. A „gomb_id” oszlopba van, hogy az adott név melyik gömbön található meg. A „varos_varos_id” a földrajzi névként előforduló városnevek ID azonosítóját írtam. A „leolvasott_varosnev” oszlopba a földgömbökön olvasható névváltozatokat írtam le. A „varos_kontinens”-nél rögzítettem, hogy a város melyik kontinensen található. Az utolsó „fovaros” oszlopba pedig jeleztem „1”-es számmal ha a város főváros, „0”-val ha nem (3. ábra).

	A	B	C	D	E	F
1	varos_id	gomb_id	varos_varos_id	leolvasott_varosnev	varos_kontinens	fovaros
2	1	32	1	Reykjavík	Európa	1
3	2	32	2	Seydhisfjörður	Európa	0
4	3	32	3	Belfast	Európa	0
5	4	32	4	Limerick	Európa	0
6	5	32	5	Cork	Európa	0
7	6	32	6	Dublin	Európa	1
8	7	32	7	Dundee	Európa	0
9	8	32	8	Aberdeen	Európa	0
10	9	32	9	Glasgow	Európa	0
11	10	32	10	Edinburgh	Európa	0
12	11	32	11	Liverp.	Európa	0
13	12	32	12	Hull.	Európa	0
14	13	32	13	Manch.	Európa	0
15	14	32	14	Birm.	Európa	0
16	15	32	15	London	Európa	1
17	16	32	16	South	Európa	0
18	17	32	17	Plymouth	Európa	0
19	18	32	18	Cherb.	Európa	0
20	19	32	19	Brest	Európa	0
21	20	32	20	Nantes	Európa	0

3. ábra: „varos” Excel táblázat

Ezzel párhuzamosan készítettem a másik „varos_mai_nevek” elnevezésű táblázatot. Itt öt oszlopba a „varos_varos_id”, „mai_varos_nevek”, „angol_varos_nevek”, „kontinens” és „fovaros” nevű oszlopokba lettek kigyűjtve a városok mai nevei. A „varos_varos_id” oszlopban a földrajzi névként előforduló városnevek ID azonosítója szerepel. A második oszlopba az egyes városnevek mai magyar megnevezései, a harmadikba a városok mai angol megnevezései találhatóak. A negyedik oszlopban a kontinenseket rögzítettem, az utolsó oszlopba pedig a fentebb említett jelölésekkel a fővárosok szerepelnek (4. ábra).

	A	B	C	D	E
1	varos_varos_id	mai_varos_nevek	angol_varos_nevek	kontinens	fovaros
2	1	Reykjavík	Reykjavík	Európa	1
3	2	Seydhisfjörður	Seydhisfjörður	Európa	0
4	3	Belfast	Belfast	Európa	0
5	4	Limerick	Limerick	Európa	0
6	5	Cork	Cork	Európa	0
7	6	Dublin	Dublin	Európa	1
8	7	Dundee	Dundee	Európa	0
9	8	Aberdeen	Aberdeen	Európa	0
10	9	Glasgow	Glasgow	Európa	0
11	10	Edinburgh	Edinburgh	Európa	0
12	11	Liverpool	Liverpool	Európa	0
13	12	Kingston upon Hull	Kingston upon Hull	Európa	0
14	13	Manchester	Manchester	Európa	0
15	14	Birmingham	Birmingham	Európa	0
16	15	London	London	Európa	1
17	16	Southampton	Southampton	Európa	0
18	17	Plymouth	Plymouth	Európa	0
19	18	Cherbourg	Cherbourg	Európa	0
20	19	Brest	Brest	Európa	0
21	20	Nantes	Nantes	Európa	0

4. ábra: „varos_mai_nevek” Excel táblázat

Egyes nevek keresését megnehezítette néhány gömb olvashatósága. Szerencsére az ugyanolyan névrajzzal rendelkező gömböknél volt mindig olyan glóbusz, amely jobb minőségű és jobb állapotú, tehát olvashatóbb volt. A 94-es ID-vel rendelkező 1938-as glóbusz a színhasználat miatt volt nehezebben olvasható, viszont hasonlított a 32-es ID-vel rendelkező gömbre, ami könnyebbé tette a nevek kigyűjtését. A 124-es ID-s 1930-as glóbusznál a domborzat ábrázolása nehezítette az olvashatóságot több helyen is. Néhány gömbön pár folt vagy kopott rész nehezítette az olvashatóságot, de nem voltak annyira zavaróak. A 89-es ID-vel rendelkező Nagy Károly-féle glóbusz városai nem kerültek be az adatbázisba, mert

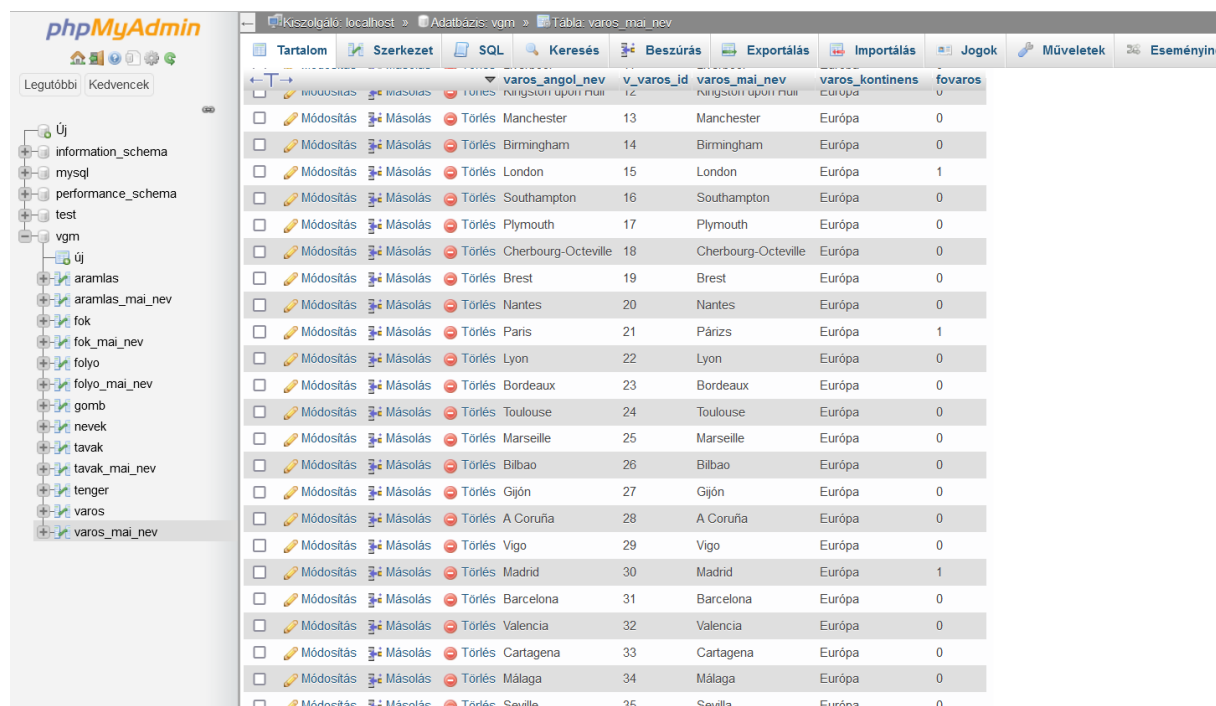
rendkívüli részletessége miatt nehezen olvasható, az átlagosnál legalább kétszer annyi nevet tartalmaz, valamint helyenként nehezen azonosíthatók a nevek mai megfelelői. A táblázatok véglegesítése után a „varos” táblázatban 13891 darab, a „varos_mai_nevek” táblázatban 2209 darab különböző város nevét gyűjtöttem ki.

3.4. A táblázatok beillesztése a phpMyAdmin adatbázisba

A phpMyAdmin egy nyílt forrású eszköz, amit PHP-ban írtak a MySQL menedzselésére egy webes kezelőfelületen keresztül. Segítségével adatbázisokat, táblákat, rekordokat tudunk kezelni létrehozni, módosítani, törölni és lekérdezni (Wikipedia, phpMyAdmin) egy grafikus felületen. A MySQL az egyik legelterjedtebb adatbázis-kezelő rendszer. Az SQL, azaz Structured Query Language pedig relációsadatbázis-kezelők szabványos lekérdezési nyelve.

A meglévő XLS formátumú táblázatokat a phpMyAdmin adatbázisba importálás előtt át kell alakítani CSV formátummá. Az átalakításnál karakterkódolást is figyelembe kellett venni.

A nevek között előfordult jó pár karakter melyet a magyar nyelvben nem használunk. Ilyen karakterek fordultak elő például legtöbbször a városok neveiben: Ålesund, Tromsø, Łódź, Đà Nẵng, São Paulo. A témavezetőmmel gyanítottuk, hogy ezek a nevek problémát okozhatnak az adatok feltöltése közben, végül többszöri próbálkozás után Excelből való kimentés során az „Unicode-os szöveg” formátum működött hibamentesen és gond nélkül be lehetett importálni a phpMyAdmin adatbázisába a két táblázatot (5. ábra).



varos_angol_nev	v_varos_id	varos_mai_nev	varos_kontinens	fovaros
Manchester	13	Manchester	Európa	0
Birmingham	14	Birmingham	Európa	0
London	15	London	Európa	1
Southampton	16	Southampton	Európa	0
Plymouth	17	Plymouth	Európa	0
Cherbourg-Octeville	18	Cherbourg-Octeville	Európa	0
Brest	19	Brest	Európa	0
Nantes	20	Nantes	Európa	0
Paris	21	Párizs	Európa	1
Lyon	22	Lyon	Európa	0
Bordeaux	23	Bordeaux	Európa	0
Toulouse	24	Toulouse	Európa	0
Marseille	25	Marseille	Európa	0
Bilbao	26	Bilbao	Európa	0
Gijón	27	Gijón	Európa	0
A Coruña	28	A Coruña	Európa	0
Vigo	29	Vigo	Európa	0
Madrid	30	Madrid	Európa	1
Barcelona	31	Barcelona	Európa	0
Valencia	32	Valencia	Európa	0
Cartagena	33	Cartagena	Európa	0
Málaga	34	Málaga	Európa	0
Sevilla	35	Sevilla	Európa	0

5. ábra: Az adatbázisba importált „varos_mai_nev” adattáblázat

4. *SHAPE* fájl

4.1. *SHAPE* fájl kiválasztása

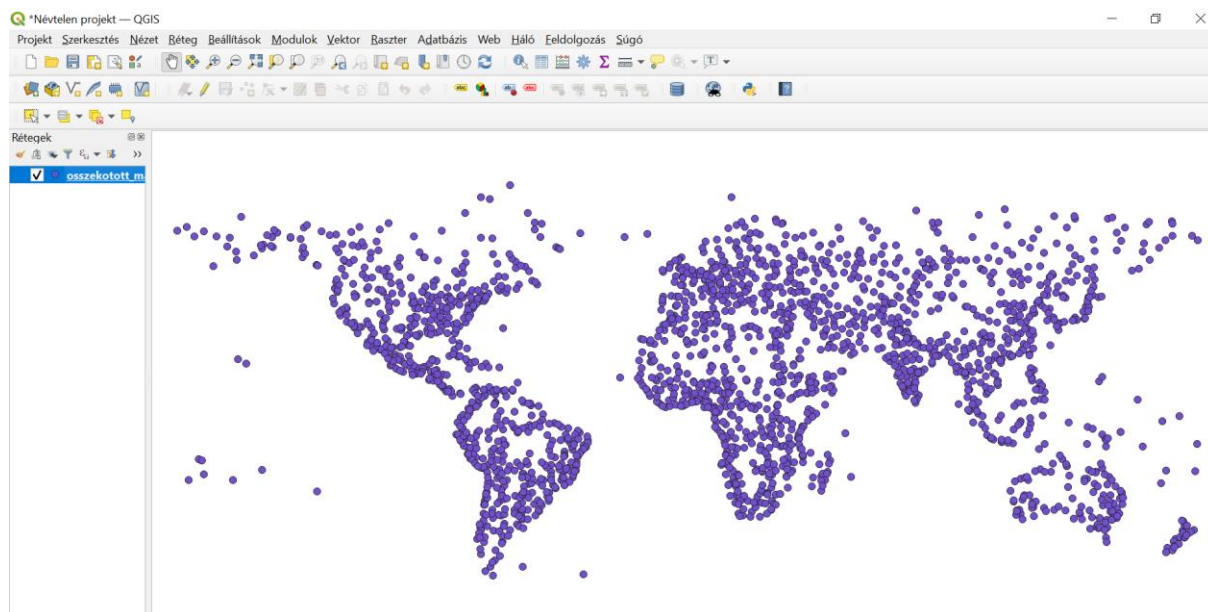
A weboldal fejlesztése során felhasznált Shape fájl és a hozzájuk tartozó adatokat tartalmazó adattáblázatokat a témavezetőm javaslatára a naturalearthdata.com oldalról töltöttem le. Ezen a honlapon lehetett választani különböző méretarányhoz tartozó adatok közül, 10, 50 és 110 milliós méretarányból. A városoknál a 10 milliós méretarányú adatokkal dolgoztam [8], ez az adatbázis tartalmazza a táblázatomban lévő elemek jelentős részét. A hiányzó elemekkel pedig külön foglalkoztam, ezekről később lesz még szó.

4.2. *SHAPE* fájl kezelése

A Shapefile főként a geoinformatikai rendszerekben elterjedt vektoros formátum, amely háromféle geometria alakzatot tartalmazhat. Ez a három alakzat a pont, vonal és a poligon. A Shape fájlokat számos szoftverben megnyithatjuk. A Shape fájlok szerkesztéséhez egy ingyenes asztali térinformatikai szoftvert, a QGIS-t használtam, mivel a szerkesztéseket és a szükséges átalakításokat egyszerűbben el lehetett végezni vele.

4.2.1. *Városok SHAPE fájl szerkesztése*

A városok pontszerű elemekként voltak megjelenítve, a korábban említett 10 milliós adatbázisban. A QGIS-ben kiválogattam a korábban elkészített adattáblázatban szereplő elemeket. Az angol nevek segítségével kapcsoltam össze az adattáblázatokat, a táblázat „mai_angol_nev” oszlopával az adatbázis „name_en” oszlopát, ezáltal könnyebben megtudtam állapítani melyek azok a városok, amelyek nem szerepelnek a letöltött adatbázis felsorolásában. Ezeket a nem ábrázolt pontokat külön Excel táblázatba rendeztem így könnyítve a munkámat, majd egyenként hozzáadtam a pontokat az adatbázishoz. Ez körülbelül 400 pont hozzáadását jelentette. Az adatok között szereplő felesleges elemek kitöröltem az adatbázisból. Kisebb nehézségeket okoztak az azonos nevű városok, azonban az adatbázis tartalmazta a városhoz tartozó országokat, megyéket és a maximum lakosságot is melyek alapján könnyebb volt megtalálni melyik a megfelelő város az adott ID-hoz (6.ábra).



6. ábra: QGIS-ben létrehozott városok SHAPE fájl

4. A keresési lehetőségek fejlesztése

5.1. A már meglévő keresések fejlesztése

Miután a meglévő honlap szerkezetét, a keresőmotort és az adatbázis felépítését áttanulmányoztam, a létező keresési lehetőségek működését kellett megértenem melyek a HTML (HyperText Markup Language) [9] leíró nyelven lettek megírva az *index.php* fájlban. Ezen kívül az keresésekhez létre lett hozva a *tipuskeres.php* fájl, amely a menüsáv második sorával, a *mainevfeltoltes.php* és *mainevkiiras.php* fájlok pedig a harmadik sor legördülő menüjével foglalkozik. Ezek a kódsorok PHP-ben [10] vannak megírva a lekérdezéshez. Az oldalon aszinkron technikával hívjuk meg a függvényeket több helyen is. Ehhez a JavaScriptet és a PHP-t együttesen alkalmazzuk. Az AJAX az **A**synchronous **J**avascript **A**nd **X**ml rövidítése. Ezzel a technikával úgy tudjuk a honlapon lévő adatainkat frissíteni, hogy nem kell a teljes weboldalt újra betölteni (szinkron mód), hanem aszinkron módon, egy esemény bekövetkeztére (pl. megnyomunk egy gombot, vagy leütünk egy billentyű stb.) töltődik be vagy frissül a tartalom. Ehhez kombináltan használtam a JavaScriptet és a PHP-t. Működését egy, a honlapomon előforduló példával illusztrálnám. Amikor a szövegmezőbe beírok egy településnevet, és rányomok a keress gombra, akkor a kattintással meghívok egy JS függvényt, amely elküld egy XHR (XMLHttpRequest-et) kérést. Ez a háttérben meghívja a PHP szkriptet, amellyel lefut a keresés az adatbázisban és visszatér az eredmény (ennek lépései: kapcsolódik az adatbázishoz, keres az adatbázisban, majd a keresési eredményeket feltölti egy HTML táblázatba). Az eredményt pedig betöltjük ebben az esetben a weboldal egy div elemébe (ahol az eredménytáblát szeretnénk megjeleníteni).

5.2. A keresési lehetőségek kiegészítése a városokkal

Következő lépés az újonnan feltöltött táblázatoknak a lekérdezésekbe és keresőbe történő beillesztése volt. Ezeket a parancsokat a témavezetőm segítségével illesztettem be a már meglévő PHP fájlokba. Első lépésként az *index.php* fájlban található „mezo”, „tipus” és „tipus2” legördülőmenüket bővítettük ki a Város választási lehetőséggel.

```
<select name="mezo" id="mezo">
  <option value="n_nevek">Tengernév</option>
  ...
  <option value="varosnevek">Városnév</option>
</select>
```

Ezután a vizuális megjelenítésben szerepet foglaló „kmlgomb” nevű elemnél egészítettük ki a választási lehetőségeket a Város rádiógombok hozzáadásával (7. ábra).

```
<br><input type="radio" id="varoskml" name="vector"
onclick="varosKml()"> Városok</input>
```

Ezt követte a már korábban meglévő keresés PHP fájljaiban végbemenő lekérdezések kiegészítése a városokkal.

Először a „tipus” -hoz tartozó *tipuskeres.php* fájlba tettük bele a keresést. Az első menünél **ki lehet választani a vizsgálni kívánt glóbuszt**, második legördülőmenüben pedig a keresni kívánt rétegcsoporthoz lehet beállítani (8. ábra). **A keresés eredményeként az általunk kiválasztott gömbön előforduló földrajzi nevek régi, mai magyar és angol nevét kapjuk meg (9. ábra).**

Az új keresési elemeket elseif elágazással adtuk hozzá a korábbi kereséshez.

```
elseif($lista=="varos"){ }
```

Majd megírtuk a lekérdezéseket az új réteghez, melyek során kikeressük egy adott név régi változatát, mai magyar és angol változatát.

```
$eredmeny_varos=mysqli_query($db,"select
varosnevek,varos_mai_nev,varos_angol_nev from varos left join
varos_mai_nev on varos.varos_varos_id=varos_mai_nev.v_varos_id
where gomb_id={$szoveg}");
```

A lekérdezést követően létrehoztuk a táblázat oszlopait.

```
if (mysqli_num_rows($eredmeny_varos)>0) {
    print "<table><tr class='elsosor'>
        <th>Régi név</th>
        <th>Mai név</th>
        <th>Angol név</th>
    </tr>\n";
    while ($sor=mysqli_fetch_assoc($eredmeny_varos))
    print "<tr>
```

A táblázat létrehozását követően feltöltöttük a táblázat oszlopait a lekérdezés eredményével.

Ha pedig a lekérdezés során nincs egyezés, kiírja hogy „Nincs találat.”

```
        <td>".$sor['varos_mai_nev']. "</td>
        <td>".$sor['varos_angol_nev']. "</td>
    </tr>\n";
    print "</table>";
}
else
    print "<b>Nincs találat!</b>";
}
```

Vízrajzinév-kereső

Írjon be egy vízrajzi nevet, válasszon névcsoporthat!

Városnév

Glóbusz

Névcsoporthat

Névcsoporthat

Jelenlegi földrajzi név

Városnév

Tengernév

Foknév

Áramlásnév

Tónév

Folyónév

Városnév

Gömb azonosítók

7. ábra: Városnévre való keresés

Vízrajzinév-kereső

Írjon be egy vízrajzi nevet, válasszon névcsoporthat!

Tengernév

Glóbusz

Névcsoporthat

Névcsoporthat

Tenger

Fok

Áramlás

Tó

Folyó

Város

Gömb azonosítók

8. ábra: Városra való keresés glóbuszok alapján

Írjon be egy vízrajzi nevet, válasszon névcsoporthat!

Városnév

ID=20, Csák Péter, Kóvári József, 16 cm

Város

Névcsoporthat

Jelenlegi földrajzi név

Gömb azonosítók

Válasszon egy földgömböt és egy névcsoporthat, vizsgálja meg a földrajzi nevek változásait, válasszon névcsoporthat!

Vizsgálja meg a földrajzi nevek változásait, válasszon névcsoporthat!

A glóbusz textúrájának betöltése

Régi név	Mai név	Angol név
Reykjavík	Reykjavík	Reykjavík
Dublin	Dublin	Dublin
London	London	London
Párizs	Párizs	Paris
Madrid	Madrid	Madrid
Barcelona	Barcelona	Barcelona
Lisszabon	Lisszabon	Lisbon
Róma	Róma	Rome
Bern	Bern	Bern
Bonn	Bonn	Bonn
Brüsszel	Brüsszel	Brussels
Hága	Hága	The Hague

9. ábra: Glóbusz alapú városnév keresés eredménytáblázata

Ezt követte a „tipus2” legördülőmenühöz tartozó *mainevfeltoltes.php* fájlban található keresésnek a városokkal való kiegészítése.

Ennél a lekérdezésnél a mai elnevezés alapján rákereshetünk a város nevére és megkapjuk, hogy az adott földrajzi névnek, milyen régi névváltozatai fordulnak elő a különböző glóbuszokon. A kereséshez szintén két darab legördülőmenü tartozik. Az első menüben ki lehet választani a kívánt keresési réteget (10. ábra). Amikor kiválasztottuk a keresési réteget, a második legördülőmenü automatikusan feltöltődik a kiválasztott rétegen előforduló földrajzi elemek mai megnevezéseivel (11. ábra). Amikor pedig kiválasztjuk a keresni kívánt mai névváltozatot, az eredményként kapott táblázatban az adott név összes előforduló névváltozatát meg lehet találni, valamint azt, hogy a régi névváltozat mely glóbuszokon fordul elő (12. ábra).

Ez a lekérdezés szintén elseif elágazással van megoldva.

```
elseif($lista=="varos"){ }
```

A lekérdezés megtörténik, melyben rákeresünk az egyrétegen előforduló nevekre.

```
$eredmeny_varos=mysqli_query($db,"SELECT * FROM varos_mai_nev  
order by varos_mai_nev");
```

Az eredményeket pedig kiíratjuk a másik legördülő menübe.

```
print "<option  
value=".$sor['v_varos_id']. ">".$sor['varos_mai_nev']. "</option  
>";
```

Végül a „minta” névre keresztelt kereséshez tartozó *kereso.php* fájlban is megírtam a két új réteghez tartozó lekérdezéseket.

```
$eredmeny_varos=mysqli_query($db,"SELECT * FROM  
'varos_mai_nev', varos, gomb where varos_varos_id=v_varos_id  
and g_gomb_id=gomb_id and v_varos_id={$szoveg}");
```

A táblázatot kiegészítettem a kiíró kódrészletekkel. Miután az összes keresésben résztvevő PHP fájlt megírtuk a város rétegre vonatkozó kódokkal, ellenőriztem, hogy minden rendben működik-e. Szerencsére minden keresés gyorsan és hibamentesen futott le.

Írjon be egy vízrajzi nevet, válasszon névcsoportot! Városnév Gömb azonosítók

Glóbusz Város Válasszon egy földgömböt és egy névcsoportot, vizsgálja

Névcsoport Jelenlegi földrajzi név Vizsgálja meg a földrajzi nevek változásait, válasszon né

Névcsoport

- Tenger
- Fok
- Áramlás
- Tó
- Folyó
- Város

10. ábra: Város rétegre való keresés

Írjon be egy vízrajzi nevet, válasszon névcsoportot! Városnév Gömb azonosítók

Glóbusz Város Válasszon egy földgömböt és egy névcsoportot, vizsgálja

Város A Coruña Vizsgálja meg a földrajzi nevek változásait, válasszon né

A Coruña

Aba

Ábádán

Abakán

Abalessa

Abéché

Abeokuta

Aberdeen

Abidjan

Abomey

Abu Hamad

Abu Nujaym

Abu-Dzabi

11. ábra: Városnevekre való keresés

Írjon be egy vízrajzi nevet, válasszon névcsoportot! Városnév Gömb azonosítók

Glóbusz Város Válasszon egy földgömböt és egy névcsoportot, vizsgálja

Város Alexandria Vizsgálja meg a földrajzi nevek változásait, válasszon né

Alexandria régi névváltozatai

Régi névváltozat	Glóbusz	Réteg váltás
El-Iskandariya (Alexandria)	VEB, 1978, 33 cm	☐
El-Iskandariya (Alexandria)	Kartográfiai Vállalat, 1965, 25 cm	☐
Alexandria	Kartográfia Vállalat, 1962, 40 cm	☐
Al Iskandariyah	Márton Mátyás, Kövári József, Mészáros Piroska, Tóvizi Kálmánné, Nagy László, 1987, 25 cm	☐
Alexandria	Kartográfia Vállalat, 1983, 25 cm	☐
Alexandria	Kartográfia Vállalat, 1983, 25 cm	☐
El-Iskandariya (Alexandria)	Cartographia, 1965, 25 cm	☐

12. ábra: Városnevek változására alapú keresés eredmény táblázata

6. *SHAPE* fájlok beillesztése a weboldalba

6.1. *KML* fájl elkészítése

A KML fájl elkészítéséhez is a QGIS szoftvert használtam. A Shape fájlból exportálni kell a vektor réteget. Az „Datasource Options” résznél tudtam megadni a „Name” (ide kerül a mai magyar név) és „Description” (ide kerül az angol név) elemeket, amelyekre szükségem volt, mert a „Description” mező alapján van összekötve az adatbázis és a virtuális Cesium glóbuszon megjelenő KML fájl. A kész „*telepulesek.kml*” fájlt a Google Earth programba nyitottam meg, hogy megnézzem minden rendesen működik-e. Ekkor nem találtam hibát a működésükben (13. ábra).

6.2. A térképi megjelenítés

A keresési lehetőségek kiegészítése után a weboldal 3D-s megjelenítőjét kezdtük el bővíteni. Itt a kódolásnak a KML fájlokban található adatok behívására vonatkozó részét továbbá a térképre való nagyítás részét kellett kiegészíteni. Ezeknek a megírását az *index.php* fájlban található Cesium-ra vonatkozó kódsorok városokkal való kiegészítésével oldottuk meg. A városokra vonatkozó kódsor megírásakor először az objektum helyét kellett megadni, ami felett az egér éppen van.

```
var mouseOverV;  
elemekV={};
```

Ezekután egy „*varosKml()*” nevű funkció következik, amellyel eltávolítjuk a térképen jelenlévő más rétegekről származó nem szükséges elemeket.

```
function varosKml() {  
    kmlt.entities.removeAll();  
    kmla.entities.removeAll();  
    kmlto.entities.removeAll();  
    kmlfo.entities.removeAll();  
    kmlf.entities.removeAll();  
}
```

Majd az üres térképre behívjuk a „*telepulesek.kml*”-ben szereplő pontszerű objektumokat.

```
viewer.dataSources.add(kmlv);  
kmlv.load('kml/telepulesek.kml?' + Math.random()).then(function(  
) {  
    var entsV=kmlv.entities.values;  
    for (i=0;i<entsV.length;i++) {  
        if (entsV[i].billboard) {  
            entsV[i].billboard.image="ylw-blank.png";  
            entsV[i].billboard.width=80;  
            entsV[i].billboard.height=80;  
            entsV[i].label.show=false;  
        }  
    }  
})
```

```

        var k=entsV[i].position._value;
        hely=[]
        ujK=[]
hely=Cesium.Cartographic.fromCartesian(k);
ujK=Cesium.Cartesian3.fromRadians(hely.longitude,hely.latitude
,40000);

        entsV[i].position=ujK;
        elemekV[entsV[i].name]=entsV[i];
    }
}
});

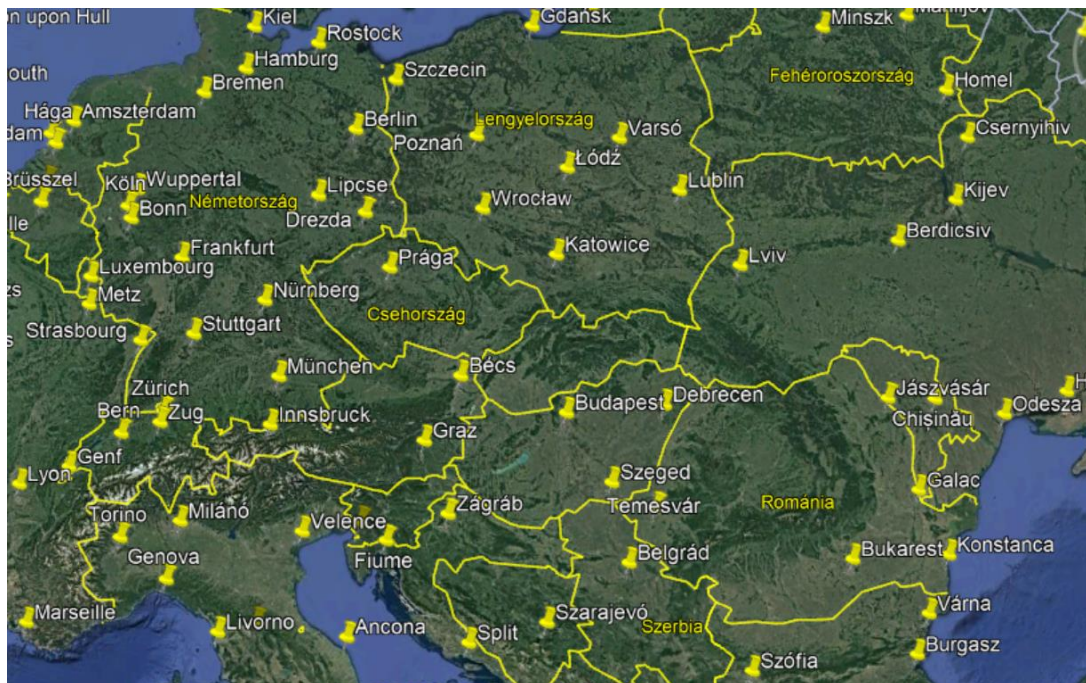
```

Ezt követően megírjuk, hogy a megjelenő objektumokat milyen formába szeretnénk megjeleníteni, valamint, hogy az egérrel milyen módon teremtsünk kapcsolatot (14. ábra).

```

var handlerV = new
Cesium.ScreenSpaceEventHandler(viewer.scene.canvas);
    handlerV.setInputAction(function(movement) {
        var objV =
viewer.scene.pick(movement.endPosition);
        if (Cesium.defined(objV)) {
            mouseOverV=objV.id;
        } else
            mouseOverV=null;
    }, Cesium.ScreenSpaceEventType.MOUSE_MOVE);

```



13. ábra: „telepulesek.kml” Google Earth-be megjelenítve



14. ábra: Városok pontszerű megjelenítése a weboldalon

6.3. Nagyítás kiválasztott elemre

A térképre való nagyítás kódolása volt az utolsó dolog, amit az index.php fájlban található Cesium-ba bővíteni kellett kódsorokkal. A nagyítás kódolásának megírásához létre kellett hozni egy funkciót a városok esetében is.

```
function zoomToVaros(nev) {
    viewer.flyTo(elemekV[nev]);
    viewer.selectedEntity=elemekV[nev];
}
```

7. A honlap módosításai

7.1. Évszámok megjelenítése a glóbusz menüknél

A már meglévő honlapon apróbb módosításokat is végrehajtottam. Az egyik ilyen a glóbuszok adataiban megjelenő évszámok voltak. Mindegyik glóbuszhoz kapcsolódó legördülő menüben beleírtam az adott gömbhöz kapcsolódó évszámot. Az index.php fájlban tudtam ezeket a módosításokat végrehajtani.

```
<!-- Gömbök listája-->
<select id="gomboc">
<option>Glóbusz</option>
    <option value="4">ID=4, Kartográfiai Vállalat, 1965, 25
cm</option>
    <option value="14">ID=14, Kogutowicz Manó, 1905, 25.5
cm</option>
```

7.2. Városnévre keresés módosítása

Eddig, ha a keresőbe beírtunk egy részletet, akkor minden olyan találatot kiadott, amelyben megtalálható volt az általunk beírt részlet. Mindez a '%{\$szoveg}%' kódrészlet miatt történik mert a % megengedi, hogy a beírt részlet előtt és után lehessenek még karakterek. Viszont mindez a városok esetében nagyon zavaró, ezért a kereso.php fájlban a beírt részlet előtti % jelet kivettem. Így a keresés könnyebben tud megvalósulni.

```
$eredmeny_varos=mysqli_query($db,"select * from varos left
join varos_mai_nev on v_varos_id = varos_varos_id left join
gomb on gomb_id=g_gomb_id where {$lista} like '{$szoveg}%')";
```

7.3. Városnevek sorrendje

Glóbuszok alapján való keresésnél alaphelyzetben olyan sorrendbe jelenítette meg az eredménytáblázat a városneveket, ahogyan az Excel-táblázatban felvettem őket (15.ábra). Keresés könnyítése érdekében a városneveket ABC-sorrendbe rendeztem a régi nevek alapján. Ez egy apróbb módosítást igényelt a tipuskeres.php fájlban az 'order by' paranccsal kiegészítve a lekérdezést.

```
$eredmeny_varos=mysqli_query($db,"select
varosnevek,varos_mai_nev,varos_angol_nev from varos left join
varos_mai_nev on varos.varos_varos_id=varos_mai_nev.v_varos_id
where gomb_id={$szoveg} order by varosnevek");
```

Írjon be egy vízrajzi nevet, válasszon névcsoportot!

Tengernév

Q

Gömb azonosítók

ID=18, Kogutowitz M

Város

Válasszon egy földgömböt és egy névcsoportot, vizsgálja meg a rajta t

Tenger

Ádeni-öböl

Vizsgálja meg a földrajzi nevek változásait, válasszon névcsoportot, és

ID18, Kogutowitz Manó, 25.5 cm

A glóbusz textúrájának betöltése

Régi név	Mai név	Angol név
Abese	Abéché	Abéché
Abomé	Abomey	Abomey
Abusehr	Busehr	Bushehr
Adelaide	Adelaide	Adelaide
Aden	Áden	Aden
Adis Abeba	Addisz-Abeba	Addis Ababa
Agadem	Agadem	Agadem
Agadesz	Agadéz	Agadez
Agra	Agra	Agra
Ahmadabad	Ahmedábád	Ahmedabad
Ain Szalah	In Salah (Ain Salah)	In Salah
Ajgun	Heihe	Heihe

15.ábra: Városneveket ABC-sorrendbe adja vissza az eredménytáblázat

8. Új keresések és funkciók hozzáadása a honlaphoz

Az előzőekben elvégzett módosítások után, a témavezetőm javaslatára új keresési funkciókat is létrehoztam a honlapon. Ezeket is aszinkron függvényhívással valósítottam meg, az előzőekhez hasonlóan. Az első új keresés lehetővé teszi, hogy egy glóbuszon a városnév réteg kiválasztásával kontinensek alapján listázzuk ki a városneveket. Egy másik lekérdezés megvalósítása is felmerült, mellyel rá lehet keresni mai elnevezés alapján, hogy az adott névnek, milyen régi névváltozatai fordulnak elő a gömbökön kontinensekre leszűkítve. Ezekhez a keresésekhez létre kellett hoznom a *varosszukit.php* és a *maivarosszukit.php* fájlt melyek tartalmazzák PHP-ben a lekérdezéshez tartozó kódsorokat. Valamint konzultációk során felmerült a fővárosok valamilyen formában történő megjelenése is.

8.1. Az első új keresés

A két új keresés közül az első keresés segít abban, hogy egy adott gömbön a város réteget kiválasztva milyen nevek fordulnak elő kontinensenként. Miután kiválasztottuk a gömböt és a város réteget, az új legördülőmenü lehetővé teszi, hogy keressünk az összes kontinensen és a hét kontinensen külön-külön (16.ábra). A keresés eredményeként pedig megkapjuk az adott gömbön előforduló városnevek régi, mai magyar és angol nevét az előbb kiválasztott kontinensen (17.ábra).

Régi név	Angol név	
Aba	Aba	
Abadan	Abadan	
Abakan	Abakan	
Abéché	Abéché	
Aberdeen	Aberdeen	
Abidjan	Abidjan	
Abu Hamed	Abu Hamad	
Abu-Dhabi	Abu Dhabi	
Acapulco	Acapulco	
Accra	Accra	
Achorage	Anchorage	
Acsinszk	Achinsk	
Adana (Seyhan)	Adana	
Addis Abeba	Addis Ababa	
Adelaide	Adelaide	
Aden	Aden	

16.ábra: Glóbusz alapú keresés összes kontinens alapján

ID32, VEB, 33 cm

Régi név	Mai név	Angol név
Adelaide	Adelaide	Adelaide
Aitape	Aitape	Aitape
Albany	Albany	Albany
Alice Springs	Alice Springs	Alice Springs
Apia	Apia	Apia
Auckland	Auckland	Auckland
Augusta	Augusta	Augusta
Ballarat	Ballarat	Ballarat
Bendigo	Bendigo	Bendigo
Birdum	Birdum	Birdum
Bourke	Bourke	Bourke
Bowen	Bowen	Bowen
Brisbane	Brisbane	Brisbane
Broken Hill	Broken Hill	Broken Hill
Broome	Broome	Broome
Bunbury	Bunbury	Bunbury

17.ábra: Glóbusz alapú városnevek keresése Ausztráliára leszűkítve

8.1.1. Az első új keresési funkció programozása

Az első lépésként az `index.php` fájlban hoztam létre a keresőhöz tartozó „kontinens1” legördülőmenüre vonatkozó kódsorokat és feltöltöttem a választási lehetőségekkel.

```
<select id="kontinens1" disabled="true"
onchange="varosSzukit();" >
  <option value="osszes">Összes kontinensről</option>
  <option value="Európa">Európa</option>
  <option value="Ázsia">Ázsia</option>
  <option value="Afrika">Afrika</option>
  <option value="Ausztrália">Ausztrália</option>
  <option value="Észak-Amerika">Észak-Amerika</option>
  <option value="Közép-Amerika">Közép-Amerika</option>
  <option value="Dél-Amerika">Dél-Amerika</option>
```

Ezután megírtam a „`varosSzukit()`” függvényt amely segítségével össze tudtam kötni a legördülőmenüt és a `varosszukit.php` fájlt. Ezáltal zajlik le a `varosszukit.php` fájlban az adatok lekérdezése.

```
function varosSzukit() {
  if (window.XMLHttpRequest) {
    xhttp=new XMLHttpRequest();

  }

  xhttp.onreadystatechange = function() {
```

```

        if (this.readyState == 4 && this.status == 200) {
            var valasz=document.getElementById("result");
            valasz.innerHTML = xhttp.responseText;
        }
    };

    var g = document.getElementById('gomboc').value;
    var k=document.getElementById('kontinens1').value;
    xhttp.open("GET", "varosszukit.php?gomb=" + g +
"&kontinens=" + k, true);
    xhttp.send();
}

```

A *varosszukit.php* fájlban a lekérdezéshez tartozó kódokat megírtam PHP programozási nyelven. Miután kapcsolódtunk az adatbázis-szerverhez végrehajtjuk a lekérdezést.

```

$eredmeny_varos=mysqli_query($db,"select
varosnevek,varos_mai_nev,varos_angol_nev from varos left join
varos_mai_nev on varos.varos_varos_id=varos_mai_nev.v_varos_id
where gomb_id={$gomb} and varos.varos_kontinens like
'{$kontinens}%' order by varosnevek");

```

A lekérdezést követően kiíratjuk őket egy táblázatba a korábban említett módon, hogy melyek a keresett gömbön található városnevek régi, mai magyar és angol névváltozatai kontinensekre bontva.

8.2. A második új keresés

A lekérdezésnél a mai elnevezés alapján rákereshetünk a város nevére és megkapjuk, hogy az adott földrajzi névnek, milyen régi névváltozatai fordulnak elő a különböző glóbuszokon. A meglévő legördülőmenükhöz egy harmadik funkciót illesztettem be, amellyel kontinensekre lehet leszűkíteni a keresést. A város réteg és a kontinens kiválasztása után a harmadik menüben a választott kontinens alapján töltődik fel a menü a városnevekkel (18., 19. ábra). Miután mind a három legördülő menüből kiválasztottuk a vizsgálni kívánt nevet az eredménytáblázatban a gömbökön előforduló régi névváltozat és a glóbusz adatai jelennek meg (20. ábra).

Írjon be egy vízrajzi nevet, válasszon névcsoportot! Tengernév Gömb azonosítók

Glóbusz Város Összes kontinens Válasszon egy földgömböt, egy kontinen

Város Összes kontinens Jelenlegi földrajzi név Vizsgálja meg a földrajzi nevek változása

Összes kontinens
Európa
Ázsia
Afrika
Ausztrália
Észak-Amerika
Közép-Amerika
Dél-Amerika

18.ábra: Város réteg és kontinens alapú keresés

Írjon be egy vízrajzi nevet, válasszon névcsoportot! Tengernév Gömb azonosítók

Glóbusz Város Összes kontinens Válasszon egy földgömböt, egy kontinen

Város Észak-Amerika Albany Vizsgálja meg a földrajzi nevek változása

Albany
Albuquerque
Alert
Amadjuak
Anchorage
Astoria
Atlanta
Augusta
Austin
Baker Lake
Baltimore
Barkerville
Bathurst Inlet
Baton Rouge

19.ábra: Város réteg és kontinens alapján feltöltődött városnév legördülőmenü

Írjon be egy vízrajzi nevet, válasszon névcsoportot! Tengernév Gömb azonosítók

Glóbusz Város Összes kontinens Válasszon egy földgömböt, egy kontinenst és egy névcsoportot, vizsgálja

Város Észak-Amerika Channel-Port aux Bas Vizsgálja meg a földrajzi nevek változásait, válasszon névcsoportot, konti

Channel-Port aux Basques régi névváltozatai		
Régi névváltozat	Glóbusz	Réteg váltás
Port-aux-Basques	VEB, 1978, 33 cm	☐
Port-aux-Basques	Kogutowicz Manó, 1910, 51 cm	☒
Port au Basque	Kogutowicz, 1938, 18 cm	☐

20.ábra: Kontinens alapú városnév keresés eredménytáblázata

8.2.1. A második új keresési funkció programozása

A második keresési funkció hasonlít az előző kereséshez. Viszont itt a keresésnél az első menüben való választás alapján töltődik fel a második és a harmadik menü. Az *index.php* fájlban létrehoztam a „kontinens2” nevű legördülőmenüt és feltöltöttem a választási lehetőségekkel.

```
<select id="kontinens2" disabled="true"
onchange="mainevSzukit();">
    <option value="osszes">Összes kontinens</option>
    <option value="Európa">Európa</option>
    <option value="Ázsia">Ázsia</option>
    <option value="Afrika">Afrika</option>
    <option value="Ausztrália">Ausztrália</option>
    <option value="Észak-Amerika">Észak-Amerika</option>
    <option value="Közép-Amerika">Közép-Amerika</option>
    <option value="Dél-Amerika">Dél-Amerika</option>
</select>
```

Megírtam a „*maivarosSzukit()*” funkciót mellyel a legördülőmenü és a *maivarosszukit.php* fájl össze lesz kötve.

```
function mainevSzukit() {
    if (window.XMLHttpRequest)
        xhttp=new XMLHttpRequest();

    xhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
            var valasz=document.getElementById("nevg");
            valasz.innerHTML = xhttp.responseText;
        }
    };
    var k=document.getElementById('kontinens2').value;

    xhttp.open("GET", "maivaroszukit.php?kontinens=" + k, true);
    xhttp.send();
}
```

A *maivarosszukit.php* fájlban megírtam a lekérdezést.

```
$eredmeny_varos=mysqli_query($db,"SELECT * FROM varos_mai_nev
where varos_kontinens like '{$kontinens}%' order by
varos_mai_nev");
```

A lekérdezést követően a lekérdezés eredményeit kiíratjuk egy táblázatba.

8.3. Új funkció programozása

Egy új funkciót is hozzáadtam a honlaphoz, amely a fővárosokat nagybetűs írásmóddal különíti el a többi várostól az eredménytáblázatban (21.ábra). Ehhez a funkcióhoz SQL

lekérdezést [11] használtam, melyet először a *kereso.php* fájl meglévő lekérdezésébe illesztettem be. Majd a *tipuskeres.php* fájl lekérdezésébe is beillesztettem ezt a nagybetűs megjelenítést a fővárosoknál (22.ábra).

```
select varos_mai_nev.*,gomb.*,varos_id, gomb_id,
varos_varos_id, CASE WHEN varos.fovaros=1 THEN
UPPER(varosnevek) ELSE varosnevek END as varosnevek from varos
left join varos_mai_nev on v_varos_id = varos_varos_id left
join gomb on gomb_id=g_gomb_id where {$lista} like
'{$szoveg}%')
```

Valamint a *mainevkiiras.php* fájl lekérdezésébe is sikerült beilleszteni a főváros jelölését (23.ábra).

```
$eredmeny_varos=mysqli_query($db,"SELECT *, CASE WHEN
varos.fovaros=1 THEN UPPER (varosnevek) ELSE varosnevek END as
varosnevek FROM `varos_mai_nev`, varos, gomb where
varos_varos_id=v_varos_id and g_gomb_id=gomb_id and
v_varos_id={$szoveg}")
```

Budapest

Városnév

Q

Gömb azonosítók

Glóbusz

Névcsoport

Összes kontinens

Válasszon egy földgömböt, egy kontinenst és egy névcsoportot

Névcsoport

Összes kontinens

Vizsgálja meg a földrajzi nevek változásait, válasszon névcsoportot

Keresett szöveg: Budapest

Régi név	Mai név	Angol név	Gömb adatai (készítő, kiadás éve, átmérő)	Réteg váltás
BUDAPEST	Budapest	Budapest	VEB, 1978, 33cm	☐
BUDAPEST	Budapest	Budapest	Kartográfiai Vállalat, 1965, 25cm	☐
BUDAPEST	Budapest	Budapest	Csák Péter, Kövári József, 1987, 16cm	☐
BUDAPEST	Budapest	Budapest	Kővári József, 1981, 16cm	☐
BUDAPEST	Budapest	Budapest	Márton Mátyás, 1985, 10cm	☐
BUDAPEST	Budapest	Budapest	Kartográfia Vállalat, 1962, 40cm	☐
BUDAPEST	Budapest	Budapest	Márton Mátyás, Kövári József, Mészáros Piroska, Tóvizi Kálmánné, Nagy László , 1987, 25cm	☐

21.ábra: Főváros megjelenése városnévre keresés alapján

Írjon be egy vízrajzi nevet, válasszon névcsoportot!

ID=15, ???, 1911, 5. Válasszon egy földgömböt, egy kontinenst és egy névcsoport

ID15, ???, 5.8 cm

Régi név	Mai név	Angol név
BERLIN	Berlin	Berlin
BUDAPEST	Budapest	Budapest
Gibraltar	Gibraltár	Gibraltar
LONDON	London	London
MADRID	Madrid	Madrid
PARIS	Párizs	Paris
ROMA	Róma	Rome
STOCKHOLM	Stockholm	Stockholm
Szt. Peterburg	Szentpétervár	Saint Petersburg

22.ábra: Glóbusz alapú keresésben a fővárosok

Írjon be egy vízrajzi nevet vagy városnevet, válasszon

Válasszon egy földgömböt, egy kontinenst és egy névcsoport

Vizsgálja meg a földrajzi nevek változásait, válasszon névcsoportot!

régi névváltozatai

Régi névváltozat	Glóbusz	Réteg váltás
BANJUL	VEB, 1978, 33 cm	☐
BATHURST	Kartográfiai Vállalat, 1965, 25 cm	☐
BANJUL	Csák Péter, Kövári József, 1987, 16 cm	☐
BANJUL	Márton Mátyás, 1985, 10 cm	☐
BANJUL	Márton Mátyás, Kövári József, Mészáros Piroska, Tóvizi Kálmánné, Nagy László, 1987, 25 cm	☐
BANJUL	Kartográfia Vállalat, 1983, 25 cm	☐
BANJUL	Kartográfia Vállalat, 1983, 25 cm	☐
BATHURST	Cartographia, 1965, 25 cm	☐
BATHURST	Kogutowicz Manó, 1910, 51 cm	☐

23.ábra: Kontinens alapú főváros keresés megjelenítése

9. Érdekességek

A városok gyűjtése közben néhány érdekességgel is találkoztam. Előfordult olyan város, amely katonai légierőbázisként funkcionált régen, ma pedig már csak romok találhatóak a helyükön: ilyen város a Mongóliában található Tamsagbulag. Vagy elhagyatott bányavárost jelöltek például az Ausztráliába lévő Nannine városa. Ezek ma már szellemvárosok, viszont megjelenítettem őket az adatbázisba, hiszen maradványok mai napig megtalálhatóak. Egyiptomban Szudán határától nem messze található Koroszkó nevű települést, amely 1890-es évekig karavánok kiindulópontja volt, az Asszuáni-gát megépítésekor a Nasszer-tó elárasztotta a várost. Valamint régebbi gömbökön összesen 38 db olyan városnév található, amit nem sikerült beazonosítanom, ezek a nevek benne maradtak az adattáblázatban viszont az adatbázisba nem jelenítettem meg őket. Ilyenek voltak például Dél-Amerikában az őserdőben jelölt települések nevei vagy Argentína déli részén jelölt Cuor nevű település, valamint Afrikában a különböző törzsek települései a Szaharában.

Külön érdekesség volt néhány gömbön megjelenő rövid jelmagyarázat. A települések lakosságszáma alapján történő különböző jelölések kapnak külön magyarázatot. A 94-es ID-vel rendelkező 1938-as glóbuszon három külön jelölés található, amely a lakosok számára utal. A 4-es és a 81-es 1965-ös glóbuszok megegyeznek, ezeken 5 különböző jelölést használtak a városokhoz, valamint, külön jelölve vannak a sarkvidéki kutatóállomások, a fővárosok ki vannak emelve nagybetűvel és aláhúzással. Az 50-es és az 53-as ID-vel rendelkező 1983-as glóbuszokon szintén 5 különböző jelölést használnak, sarkvidéki kutatóállomásokat külön jelölik. A különbség a nevek jelölésében található önálló állam fővárosát félkövér, nagybetűvel és aláhúzással jelölik, a függő területek székhelyét pedig aláhúzással. A 89-es ID-vel rendelkező Nagy Károly féle 1840-es glóbusz városai nem kerültek bele az adatbázisba, mert a gömb több ezres mennyiségben tartalmazott névjegyzet, sűrűsége miatt nehezen olvashatóak és értelmezhetőek a megírások.

10. A végleges weboldal elérhetősége

A végleges weboldalt a <http://terkeptar.elte.hu/vgm/vizrajz> oldalon tettem elérhetővé, a korábbi változatot frissítettük az újabbal.

11. Összefoglalás

A dolgozatom elkészítése folyamán egy korábban elkészített weboldal továbbfejlesztésével foglalkoztam. Munkám fő célja, a Vízrajzi-névkereső nevű weboldal kontinenseken található városnevekkel való kibővítése, keresési lehetőségek kiegészítése és módosítása volt.

A dolgozatom első részében részletesen bemutattam a fejlesztéseim előtti weboldalt, valamint a városokhoz készült adattáblázatok elkészítését. Majd bemutatásra került a táblázatokból létrejött adatbázis felépítése és az adattáblák között a lekérdezések során fellépő kapcsolat szerkezeti felépítése.

Dolgozatom második része a honlap technikai fejlesztésével foglalkozik, ahol részleteztem, milyen megoldásokkal lett bővítve a honlap. A weboldal továbbfejlesztése során használnom kellett különböző térinformatikai szoftvereket. Valamint eltérő programozási nyelvek segítségével kellett újfajta kereséseket létrehoznom, melyekkel bővítettem a weboldalt.

A fejlesztéseim végeredményeként létrejött bővített weboldalon nyomon lehet követni a tengerek, fokok, áramlások, tavak és folyók nevei mellett a városok megnevezéseinek időbeli változását a magyar nyelvű glóbuszokon.

12. Hivatkozások

- [1] Szabó Virág 2010: Tengeri vetületek térképi magyar névanyagának fejlődéstörténete vizsgálata a XIX. és XX. századi földgömbök alapján. Diplomamunka, ELTE, Budapest, 2010.
- [2] Beszkid Arthur, 2017: Webes összehasonlító névrajzi kereső kialakítása glóbuszokhoz. Diplomamunka, ELTE, Budapest, 2017.
- [3] Kuris Zoltán, 2021: Webes összehasonlító névrajzi kereső fejlesztése glóbuszokhoz. Diplomamunka, ELTE, Budapest, 2021.
- [4] Virtuális Glóbuszok Múzeuma: <http://terkeptar.elte.hu/vgm/2/?lang=hu>
- [5] Mozaik Kiadó, 2015: képes Földrajzi atlasz középiskolásoknak. Mozaik Kiadó, Szeged, 2015.
- [6] Carthographia Tankönyvkiadó, 2015: középiskolai Történelmi atlasz. Carthographia Tankönyvkiadó Kft, Budapest, 2015
- [7] Nations Online Project térképei: <https://www.nationsonline.org/oneworld/map/index.htm>
- [8] Natural Earth 10 milliós adatok: <https://www.naturalearthdata.com/downloads/10m-cultural-vectors/>
- [9] W3Schools HTML: <https://www.w3schools.com/html/>
- [10] W3Schools PHP: <https://www.w3schools.com/php/>
- [11] W3Schools SQL: https://www.w3schools.com/sql/sql_case.asp

13. Köszönetnyilvánítás

Ezúton is szeretnék köszönetet mondani témavezetőmnek, Ungvári Zsuzsannának, aki a weboldal fejlesztése során hasznos tanácsokkal látott el és rengeteg segítséget nyújtott a szakdolgozatom elkészítéséhez.

Nyilatkozat

Alulírott, Krizsán Helga 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, 2022. május 15.

.....
Krizsán Helga
a hallgató aláírása