

EÖTVÖS LORÁND TUDOMÁNYEGYETEM
TERMÉSZETTUDOMÁNYI KAR

Az V. kerület idegenforgalmi célú térinformatikai adatbázisa

SZAKDOLGOZAT
FÖLDTUDOMÁNYI ALAPSZAK

Készítette:

Gyenes Tamás

térképész és geoinformatikus szakirányú hallgató

Témavezető:

Mészáros János

egyetemi tanársegéd

ELTE Térképtudományi és Geoinformatikai Tanszék



Budapest, 2015

Tartalomjegyzék

1. Bevezetés	3
2. Az V. kerület	4
2.1. Bemutatása.....	4
2.2. Történelmi áttekintése.....	5
3. Az alaptérkép.....	8
3.1. Forrás és tartalom.....	8
3.2. Személyre szabás	10
4. Terepi felmérés módszerei és menete.....	13
4.1. A használt applikáció és eszköz.....	13
4.2. Saját projektek	14
4.2.1. Éttermek	15
4.2.2. Szállodák	15
4.2.3. Látnivalók.....	17
4.3. Használt módszer	17
4.4. Felmérés menete	17
4.5. Térképi megjelenítés	18
5. Webes megjelenítés	20
5.1. Leaflet	20
5.2. Előkészítés és létrehozás.....	20
5.3. Módosítások.....	22
5.4. Publikálás.....	28
6. Összefoglalás	29
7. Hivatkozások	30
8. Ábrajegyzék.....	32
9. Köszönetnyilvánítás	33
10. Mellékletek	34
11. Nyilatkozat	35

1. Bevezetés

Dolgozatom célja Budapest V. kerületének rövid bemutatása, illetve a városrészben található idegenforgalmi célú egységeinek terepi felmérése egy új telefonos applikáció használatával, és a felmért objektumok számára egy adatbázis létrehozása, mely az elsődleges információkat tartalmazza. Ezen felül az adatbázis térképi megjelenítése egy erre a célra átalakított alaptérképen, amelyet egy geoinformatikai szoftver segítségével szerkesztettem át. Az elkészült térkép és a mögötte álló adatbázis megfelelő képet ad a kerületben elhelyezkedő éttermekről, szállodákról és látnivalókról. Annak érdekében, hogy ezt ne csak az adott szoftverrel rendelkezők tudják megtekinteni, létrehoztam egy webes megjelenítést, ezáltal mindenki számára látható.

Az én lakhelyemről, XVIII. kerület, idegenforgalmi célú adatbázist építeni nem lett volna célszerű, így a választás az V. kerületre esett. Több indokkal is lehet magyarázni a területválasztásom okát, elsősorban ez a városrész megannyi turisztikai helyet és látványosságot kínál minden idelátogatónak korosztálytól függetlenül és a városlakóknak is kedvelt szórakozási helye, másrészt személyes kötődése is van, mivel ide jártam gimnáziumba, megismerve ezáltal a kerület utcáit.

Kisebb kutatás után találtam korábbi szakdolgozatot, amely tematikája hasonló, de mind más területre készültek, illetve más módszerrel és szoftverrel építették fel az adatbázist és a térképi megjelenítést.

2. Az V. kerület

A következőkben röviden összefoglalom a kerület adottságait, történelmét és az eseményeket, amelyek miatt kormányzati és turisztikai központtá vált.

2.1. Bemutatása

Budapest V. kerületéről, a kerületi önkormányzat által megállapított összefoglaló néven: Belváros-Lipótváros, kijelenthetjük, hogy a főváros szíve. A térképre pillantva olyan ez a kerület, mint egy kerékagy, amiből küllőként indulnak ki a város évszázados jelentőségű főútjai, sugárútjai (<http://www.budapestcity.org>).

Fekvését tekintve a pesti oldal központi kerülete. Északon a Szent István körút a XIII. kerülettől, keleten a Bajcsy-Zsilinszky út a VI. kerülettől, a Károly körút a VII. kerülettől, a Múzeum körút a VIII. kerülettől, valamint délen a Vámház körút a IX. kerülettől választja el. Nyugatról a Duna folyó, illetve ezáltal a XI. kerület, az I. kerület és a II. kerület határolja.

A kerületet két városrész alkotja, Belváros és Lipótváros. Belvárosnak Római birodalmi kötődése is van, ugyanis valaha a Contra-Aquincum nevű római erőd állt a mai Március 15. tér helyén, amelynek néhány feltárt falmaradványa ma is látható (Nagy, 1934). Az északabbra fekvő Lipótvárost a Deák Ferenc utca választja el Pest történelmi városmagjától.



1. ábra: A kerület városrészei

Sok ágazat központja, ugyanis számos tudományos, politikai, művészeti és üzleti intézménynek biztosít helyet. Ez az a városrész, melyet megemlítve sokan asszociálnak a Váci utcára, a Kossuth Lajos térre, a Szent István-bazilikára és a Parlamentre; illetve gondolnak, mint az egykori forradalmak kiinduló helyére.

Nagyságát vizsgálva mind területileg, mind népessége szerint a harmadik legkisebb kerület a huszonhárom közül; területe csupán 2,59 km², a népessége pedig a 2014. január 1-jei adatok alapján 26048 fő (Waffenschmidt, 2014). Természetesen mást mutatnának a számok, ha nem csak az állandó lakosokat vennénk figyelembe, hanem a jelenlévő lakosságot is, ugyanis a kerület nap mint nap nagy forgalmat bonyolít le, beleértve a turistákat, a vásárlókat és az irodákban dolgozó embereket, illetve a zöld területre vágyókat, mivel a kerület körülbelül 146 000 m²-nyi gondozott parkkal rendelkezik (<http://www.budapestcity.org>). Az elmúlt pár évben megannyi felújítás történt a városrészben, a nagyobbak közé tartozik a Kossuth Lajos tér, az Erzsébet tér, a Ferenciek tere, a Kálvin tér és ezt a két teret összekötő útszakasz, beleértve az Egyetem tér átépítése.

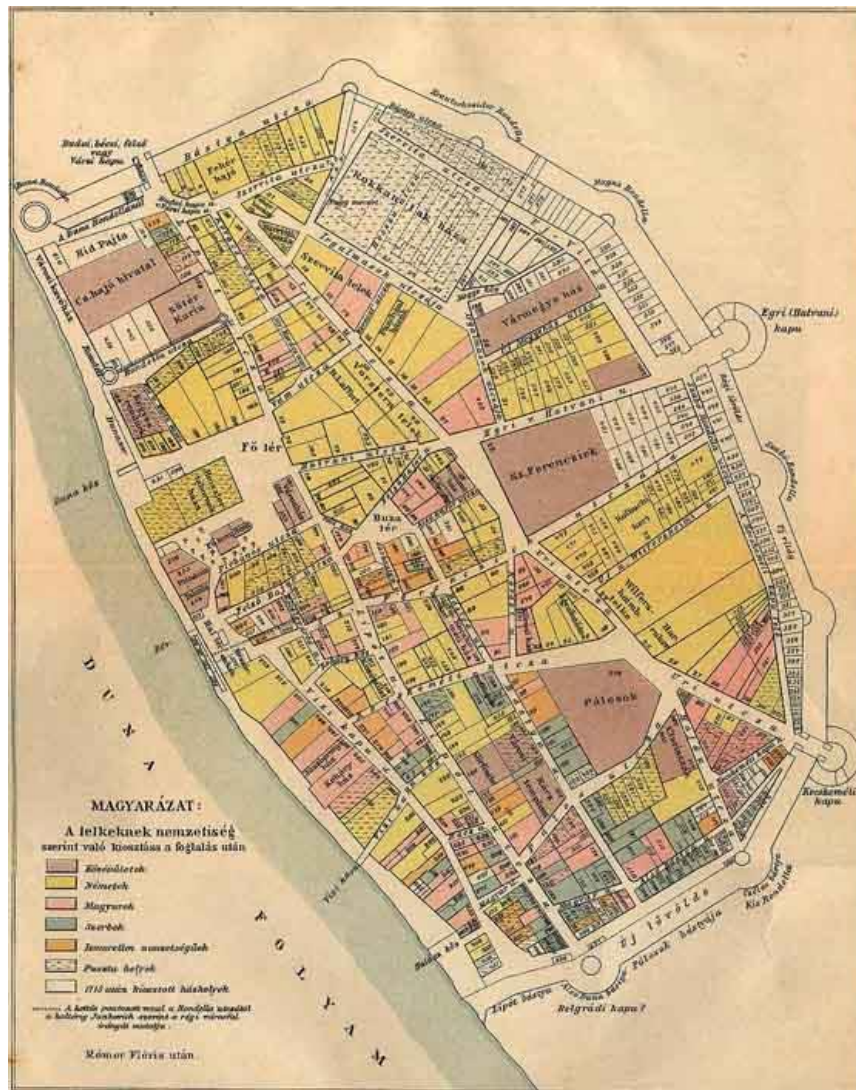
A közösségi közlekedését tekintve igen sokoldalú, számos módon megközelíthető a városrész. A Dunán négy hídon át is kötődik a kerület a túlparti részekhez, melyeken villamos- és autóbuszvonalak segítik az átjutást. A Deák Ferenc tér a főváros tömegközlekedése szempontjából nagyon jelentős. Megállója van a téren az M1-es, az M2-es és az M3-as metrónak, így a város földalatti vasúthálózatának legjelentősebb csomópontja. A vonalak állomásait a Deák téri átszálló komplexum köti össze, továbbá a felszíni tömegközlekedésben is jelentős szerepe van.

A városrész mai építészeti képének kialakulása a XIX. század elején kezdődött el. A kerület népességgyarapodása jobb helykihasználást igényelt, amit zárt sorba, klasszicista stílusban épített bérházakkal oldottak meg, melyeket olykor bérpalotáknak is neveztek (Bibó, 1972).

2.2. Történelmi áttekintése

A Belváros, illetve a régi Pest története egészen a XI. századig visszanyúlik, ezekben az időkben keletkezett, és vált kereskedőtelepüléssé. A Pest városnevet először 1148-ban használták egy oklevélben. A XIII. századra már városfal ölelte körül, melynek egykori vonalán húzódik a mai Kiskörút; ezen a falon három kapun lehetett átjutni: a Bécsi kapun a mai Váci utcában, a Hatvani kapun az Astoria aluljárónál és a Kecskeméti kapun a Kálvin tér Kecskemét utca torkolatánál. Ahol a fal vonala megtört, ott félkörös tornyok

helyezkedtek el; maga a fal körülbelül hat méter magas és nem egészen két méter széles volt (Gárdonyi, 1938).



2. ábra: Pest szabad királyi városa 1758-ban (A Pallas Nagy Lexikona, 1893)

A tatárjárást Pest városa sem úszta meg, a tatárok 1241-ben felégették a várost. A tatárjárást nehezen kiheverő város népessége lassan gyarapodott és csak a XV. században nyerte vissza a bíróválasztási jogát; ebben a században érte el középkori fénykorát Mátyás király uralkodása alatt és vált szabad királyi várossá (Uo.).

A török hódoltság alatt (1541-1686) Pest jelentősége sokat csökkent és a kereskedelmi szerepe is megsínylette a megszállást. A Habsburgok által vezetett felszabadítás során a város nagy része elpusztult és a harcoknak a lakosság nagy része is áldozatul esett (Uo.)

A XVIII. század elején a németek mellett szerb és görög telepések érkeztek a romos városba, és a tehetős kereskedőknek köszönhetően a város hamar talpra állt és az ország

kereskedelmi központjává vált. A század végére a városfal által körülölelt terület már kevés volt a pesti lakosoknak és azon túl is terjeszkedtek, ennek következményeként jött létre Teréz-, József-, Ferenc- és Lipótváros, utóbbi II. Lipót koronázásának emlékére kapta a nevét. Mivel a középkori falak csak gátolták a fejlődést, ezért megkezdődtek a lebontási munkálatok, ám nyomai még felelhetőek a mai Kiskörút belső vonalán fekvő háztömbök között, illetve a Belváros utcaszerkezete őrzi a középkori útvonalakat. Egykor a fal külső oldalán, mielőtt újra benépesedtek volna, több kisebb település helyezkedett el, például: Újbécs. Ezek a falvak a török időkben teljesen elpusztultak (Nagy, 1959).

A reform kor jó hatással volt a városra, ugyanis ezalatt az irodalmi-szellemi élet központjává vált, köszönhetően a Széchenyi István neve által fémjelzett új nemzeti mozgalomnak. Ennek keretein belül kezdődtek meg a Széchenyi lánchíd építésének folyamatai (Pallas Nagy Lexikona, 1893).

Az 1838-as pesti árvíz csaknem teljesen elöntötte, vízborítottsága néhol a két métert is elérte, okozva ezzel súlyos pusztítást. Az árvíz szintjét és tetőzésének dátumát számos emléktábla jelzi több kerületi épületen (Rajna, 1979).

Pest, Buda és Óbuda egyesítésével 1873-ban született meg Budapest és ezután kezdődött meg az a nagyszabású városrendezési program, mely a Belváros mai arculatát határozta meg. Ekkor épült meg a Szabadság és az Erzsébet híd, és alakult ki a Váci utca és a Duna-korzó is (Schiffer, 1998).

Lipótváros mai képének kialakulásának fontos állomása volt 1897, az Újépület lebontása, ahová az 1848-49-es forradalom és szabadságharc bukása után tömegével zárták a magyar hazafiakat és 1849.október 6-án az épület falánál végezték ki Batthyány Lajos gróft. A helyén jött létre a Szabadság tér és ekkor alakult ki az Országház építésével a Kossuth Lajos tér is. A terület az ország politikai-közigazgatási központjává vált, az ide települt ipart kitelepítették (Nagy, 2004).

Az egyesített várost eredetileg tíz kerületre osztották, melyek közül a Belváros a IV., míg Lipótváros az V. sorszámot kapta meg. 1950-ben Lipótváros egyik részét, Újlipótvárost, a XIII. kerülethez csatolták, a Belvárost pedig az V. kerülethez és így alakultak ki mai határai (Gergely, 2002).

A II. világháború ostromának nyomai rengeteg kerületi épületen megtalálhatóak, illetve számos épület is megsemmisült.

3. Az alaptérkép

Ebben a fejezetben bemutatom az alaptérképem forrását és tartalmát, illetve hogyan alakítottam át a téma igényeinek megfelelően.

3.1. Forrás és tartalom

Térképem alapjául a Geofabrik (<http://www.geofabrik.de>) honlapjáról ingyenesen letölthető vektoros állományt használtam fel. Ez az állomány az OpenStreetMap adatai alapján készült, mely egy szabadon szerkeszthető és felhasználható térkép bárki számára (<http://www.openstreetmap.hu>). A geofabrik oldalán az összes kontinensre és ezen belül az összes országra tölthetünk le adatokat.

A sikeres letöltés után megállapíthatjuk, hogy több réteget is tartalmaz, mint például: épületek, vasúthálózat, úthálózat, vízhálózat, zöld területek; ezek közül van, ami csak részben feltöltött, ha az épület réteget jobban megvizsgáljuk, láthatjuk, hogy sok helyen hiányos.

A rétegeket egy ESRI shape formátumban tárolja, mely az Esri (Environmental System Research Institute) vállalat saját szoftverének (ArcGis) a formátuma, ami egyben a legelterjedtebb vektoros formátum a térinformatikai szoftverek körében. A shape fájl geometriai adatokat tárol topológiai információk nélkül, valamint leíró adatokat, amelyek mind egy-egy geometriai elemhez tartoznak. Egy ilyen formátum több fájlból tevődik össze, jelen esetünkben ötből és a fájl típusára annak kiterjesztése utal.

- Ezek a következők:
- .shp - az elemek koordinátái
ez a tényleges shape fájl
 - .shx - a geometriai rekordok indexei
ez az indexfájl
 - .dbf - az attribútum táblázat
ez a dBASE fájl
 - .prj - a vetületi leírás
 - .cpg - a karakterkódolás
- ez a három alapfájl,
ezek minden esetben
szükségesek

A rétegek alapfelületét tekintve a WGS84 (World Geodetic System 1984) nevű geodéziai dátum. A letöltött vektoros állományban található rétegek közül kettő volt, amit a térképem kialakításához használtam, az út- és vízhálózat réteg. Ezeknek az adattábláit tekintve a vízhálózatnak négy darab mezője van:

- osm azonosító (osm_id)
- név (name)
- típus (type)
- szélesség (width)

A szélesség igen kevés helyen van megadva és a legtöbb elemnél null érték szerepel. A típus mezőt megtekintve láthatjuk, hogy minden szerepel benne, ami a vízhez kötődik, mint például:

- csatorna (canal)
- gát (dam)
- árok (ditch)
- hajódokk/kikötő (dock)
- zsilipkapu (lock_gate)
- folyó (river)
- folyam (stream)

Az úthálózatnak nyolc darab mezője van:

- osm azonosító (osm_id)
- név (name)
- referencia szám (ref)
- típus (type)
- egyirányúság (oneway)
- híd-e (bridge)
- alagút-e (tunnel)
- megengedett sebesség (maxspeed)

Az út jellege alapján a következő úttípusokat különítették el:

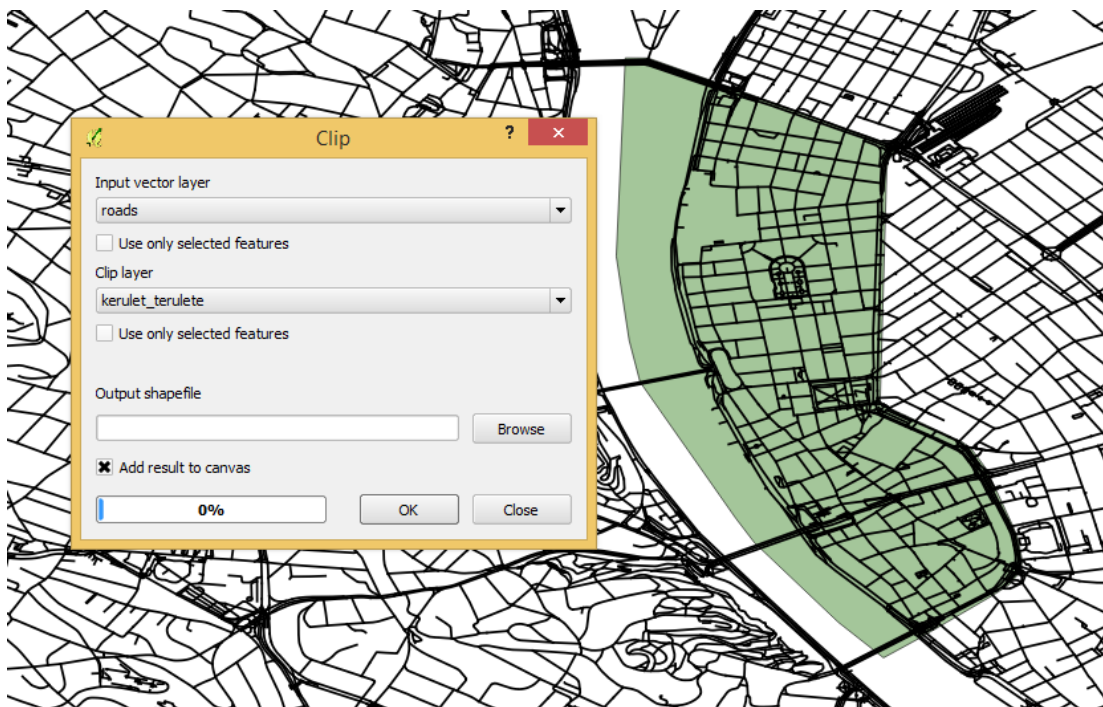
- | | | |
|---|------------------------------------|---|
| • építkezés alatt lévő út
(construction) | • gyalogút (footway) | • főút (primary) |
| • kerékpárút (cycleway) | • ösvény (path) | • földút (bridleway) |
| • lakossági út (residential) | • peron (platform) | • buszmegálló (bus_stop) |
| • lakóövezet (living_street) | • járda (pedestrian) | • pihenő övezet (rest_area) |
| • autópálya (motorway) | • parkolási zóna
(service) | • két nyomvonalú út (track) |
| • autópálya összekötő
(motorway_link) | • lépcső (steps) | • versenypálya (raceway) |
| • másodrangú út (secondary) | • harmadrangú út
(tertiary) | • tervezett út (proposed) |
| • járda (pedestrian) | • autóút (trunk) | • fordulókör (turning_circle) |
| • másodrangú út összekötő
(secondary_link) | • autóút összekötő
(trunk_link) | • nem osztályozott utak
(unclassified) |
| • főút összekötő
(primary_link) | • folyosó (corridor) | • harmadrangú út összekötő
(tertiary_link) |

3.2. Személyre szabás

A térképem kialakításához használt két réteget a QGIS nevű szoftver 2.6.1-es verziójával nyitottam meg.

Első lépésként a karakterkódolást állítottam be UTF-8-ra (8-bit Unicode Transformation Format), mely egy változó hosszúságú, veszteségmentes karakterkódolási eljárás, ezáltal a magyar abc-ben szereplő összes betű rendesen megjelenik az adattáblázatokban.

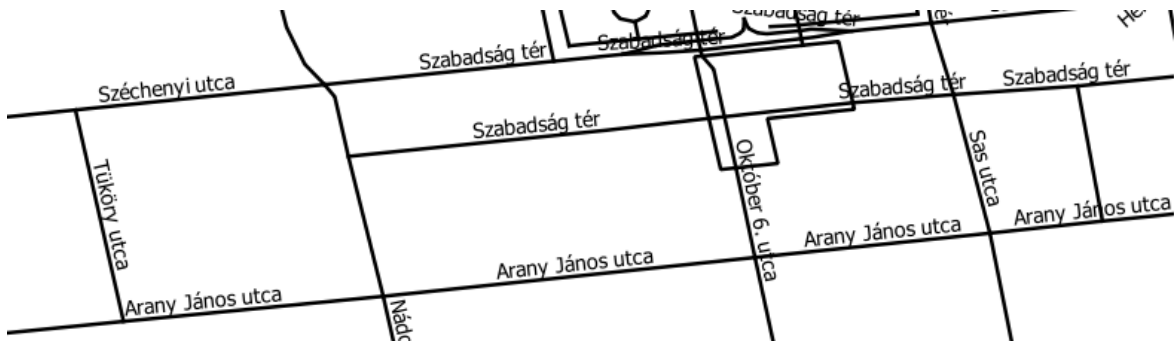
Eredetileg a két választott réteg Magyarország teljes területére kiterjedt és nekem csak az V. kerületre eső részre volt szükségem, ezért létrehoztam egy poligon típusú vektoros réteget, amely a kerület határáig terjedt ki. Ezt követően egy geometriai eszközzel (clip) az út- és a vízhálózat réteget az előbb létrehozott poligon területére korlátoztam, megkapva így a kerület utcáit és a Duna Szabadság híd és Margit híd közé eső szakaszát. A rétegek neve eredetileg angolul volt megadva, de a későbbi lépések során áttértem magyarra.



3. ábra: Levágás a "clip" parancs segítségével

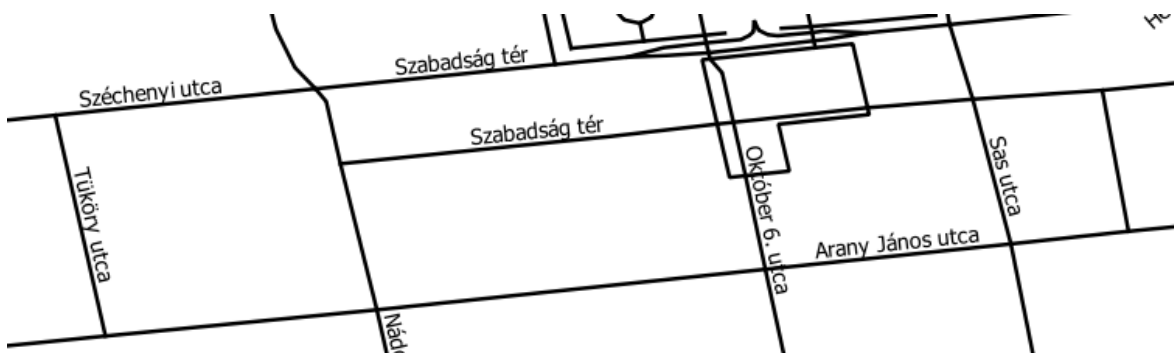
A művelet után a rétegek adattábláinak típus mezője is nagy részben megváltozott, ugyanis eltűntek azok az út és vízhálózat típusok, melyek a kerület területén nem találhatóak meg. Ezután is még sok felesleges úttípusom volt, mint például: lépcső, kerékpárút; ezért ezeket töröltem. Annak érdekében, hogy még letisztultabb legyen a

térképem, jobban megvizsgáltam a szükséges úttípusok részeit és töröltem a szükségtelen részeket, például: metróaluljárók. A rétegek adattábláinak felesleges mezőit is töröltem, így mindkettő rétegnek csak név és típus mezője lett.



4. ábra: Topológia javítás előtt

Egy utca/út több apróbb vonalrészből tevődött össze, ezért az azonos utcát/utat jelölőket összevontam, ezzel is csökkentve az attribútum táblázat méretét és javítva a térkép topológiáját.











5. ábra: Topológia javítás után

A rétegek vetületét az adatok alakhűbb szemlélése érdekében átranzformáltam EOVB-ba (Egységes Országos Vetület), ezt a transzformációt a használt szoftver segítségével könnyedén véghez tudtam vinni.

A vízhálózat megjelenítési kategorizálása nem vett sok időt igénybe, ugyanis csak egy típus van, a folyó. Az úthálózat esetében négy kategóriát különítettem el, melyeket különböző színnel és vastagsággal ábrázoltam, majd ezek szimbólum szintjét úgy állítottam be, hogy a másodrangú út legyen legfelül, ezt követi a harmadrangú út, majd a lakossági út és a legvégén a gyalogút. A kategóriák beállításánál tudjuk megadni, hogy milyen név jelenjen meg a jelmagyarázatban, a 'Value' mező mutatja az adattáblában felvett értéket, a 'Legend' mező pedig a jelmagyarázatét. Eredetileg a két mező

megegyezett, de a jelmagyarázatot átírtam magyar nyelvűre. Mértékegységként mindenhol millimétert használtam, így a későbbiekben, ha igény van rá könnyebb kialakítani a nyomtatási képet. A választott mértékegység miatt, a vonalvastagságok minden méretarányban ugyanakkorának látszódnak, emiatt a rakparti útszakasz és a Duna partvonala között annál nagyobb térköz alakul ki, minél nagyobb méretarányban nézzük. A nagy térköz elkerülése végett létrehoztam a Dunának egy poligon réteget, így rögzítve a szélességét és esztétikusabbá téve a térképet, illetve érzékelhetőbbé vált a Duna közelsége nagy méretarányban is.

Symbol	Value	Legend	Symbol	Value	Legend
	river	folyó		secondary	másodrangú út
				tertiary	harmadrangú út
				residential	lakossági út
				footway	gyalogút

6. ábra: Megjelenési kategóriák

Végezetül a névrajzot készítettem el, melyet a rétegek tulajdonságánál állítottam be, az elemek nevét tartalmazó mezőt írtam ki automatikusan, az utak esetében 10 pontos betűmagyságot és fekete, Arial betűtípust használtam. A Dunát ábrázoló poligonhoz nem tüntettem fel névrajzot, mert nem követte volna a Duna vonalát, ezért ezt a réteget legalulra helyeztem a rétegkezelőben, és ugyanolyan szint adtam neki, mint a vonalas vízhálózatnak, amelynek azonos módon, mint az utaknak elkészítettem a névrajzát, csak 30 pontos betűmagyságot és dőlt, sötétkék, Times New Roman betűtípust használva.

4. Terepi felmérés módszerei és menete

A következő fejezetben bemutatom a felméréshez használt applikációt és a felmérés menetét, továbbá a felmért pontok megjelenítését a korábban bemutatott alaptérképemen.

4.1. A használt applikáció és eszköz

A GIS Cloud és az MDC-hez hasonló rendszer az Esri által kiadott ArcGIS nevű applikáció, de ennek segítségével akár a felmérő eszközön is végrehajthatunk térképi műveleteket. Ezen kívül még pár hasonló van, de a legtöbb térinformatikai kötődésű telefonos applikáció csak helymeghatározást és útvonaltervezést tesz lehetővé, ami az én esetemben kevés lett volna.

A terület felmérését a megszokottól eltérő módon csináltam, a web alapú GIS Cloud nevű szoftverrel, mely mindenki számára szabadon elérhető az interneten (<http://www.giscloud.com>), bárki regisztrálhat és használhatja ezt bármilyen telepítés nélkül. A GIS Cloud segítségével térképeket hozhatunk létre és szerkeszthetjük, elemezhetjük azokat számos térinformatikai eszközzel, ugyanis a rétegkezelés mellett, a legalapvetőbb vektoros és adatbázis kezelő eszközök megtalálhatóak a szerkesztőfelületén. A honlapjukon rengeteg különböző tematikájú térképet megtalálhatunk és használhatjuk azokat, de ha esetleg nem találnánk a számunka megfelelőt, akkor saját térképet is importálhatunk a szerkesztőfelületre, ezen felül .csv (Comma-Separated Values) és .xls (MS Excel) kiterjesztésű fájlokat is adhatunk hozzá, amelyeket geokódolhatunk a program segítségével. Exportálni is lehet réteget és térképet egyaránt .shp (Shapefile), .mif (Mapinfo file), .kml (Keyhole Markup Language), .csv (Comma-Separated Values) vektoros formátumú kiterjesztésekben, illetve .png (Portable Network Graphics) és .tif/geotiff (Geospatial Tagged Image Format) raszteres formátumú kiterjesztésekben. Az ingyenes regisztrációval 100 Mb-os tárhely áll rendelkezésünkre vagy 10 000 mért pont tárolása. Amennyiben többet szeretnénk, lehetőségünk van prémium csomagot vásárolni, amely még további funkciókat és nagyobb tárhelyet tartalmaz.

A GIS Cloud honlapján tudunk létrehozni saját projekteket is, ez annyit tesz, hogy készíthetünk attribútum táblázatot saját igényeink szerint. Megadhatjuk az egyes mezők neveit és hogy milyen típusúak legyenek. Az utóbbira számos lehetőséget biztosít egy legördülő lista, például vannak az alapvető típusok, mint a szöveg és szám, de ezeken felül lehet dátum, fénykép és hangfelvétel is, vagy létrehozhatunk egy saját listát, amelyek közül ki lehet választani a mező értékét. A mezők számát és sorrendjét kedvünk szerint

módosíthatjuk, és adhatunk nekik leírást is. A saját projektjeink meg is jelennek a szerkesztőfelületen található 'Map' almenüpontban.

Type	Name			
Text	szöveg	Details	Delete	
Number	szám	Details	Dependencies	Delete
Date and time	dátum és idő	Details	Dependencies	Delete
Photos	fénykép	Details	Dependencies	Delete

7. ábra: Saját projekt létrehozása

A GIS Cloud másik fontos része az MDC (Mobile Data Collector). Ez egy ingyenesen letölthető telefonos applikáció, mind Android, mind IOS operációs rendszerre elérhető. Az MDC segítségével tudunk pontot hozzáadni a fentebb említett projektekhez, amely az attribútum táblázaton kívül, a pont földrajzi helyét is tárolja. A pontok helyzetét három féle módon adhatjuk meg telefonnal/tablettel: internet hozzáféréssel, ez történhet Wi-Fi vagy mobilinternet segítségével, ennek a pontossága átlagosan a gyakorlatban 20-30 méter, lehet GPS-szel (Global Positioning System), ha a telefon alkalmas rá, ez pontosabb, mint az internetes módszer, átlagosan 8-10 méteres eltérések jelentkezhetnek, végül pedig elhelyezhetjük saját magunk is a pontunkat a térképen. A hálózati hozzáféréssel tudjuk megszabni az adatfeltöltés idejét, ha ez a funkció be van kapcsolva, akkor az eszköz a felmért adatokat valós időben feltölti a webes felületre, ha pedig ki van kapcsolva, az eszköz tárolja az adatokat, ameddig manuálisan be nem kapcsoljuk és szinkronizáljuk a felmért adatokat a projekttel. Ez lehetővé teszi, hogy internet-hozzáférés nélkül is tudjunk felméréseket végezni.

Felmérésemhez egy Quilive 4,5" típusú okostelefont használtam, amelyen az Android operációs rendszer 4.2.2 verziója fut.

4.2. Saját projektek

Összesen három projektet, azaz adattáblát hoztam létre a különböző idegenforgalmi célt szolgáló egységeknek. Ezek a következők: éttermek, szállodák és látnivalók.

4.2.1. Éttermek

Ebben a táblázatban kaptak helyet az éttermek, gyorséttermek, cukrászdák, fagyfaltozók, kávézók, teázók, italozók és a szórakozóhelyek is, ezekből 346 darabot mértem fel. Az italozókat nem akartam további kategóriákra bontani, mint például: kocsmá, pub, borozó vagy bár. Az egyes vendéglátói egységekről a táblázat a következőket tartalmazza:

- név
- típus
- irányítószám
- közterület név
- házsám
- nyitva tartás
- leírás

A leírás egy pár mondatos rövid bemutatás a helyről. Ez a bemutatósöveg az esetek többségében a hely milyenségére és a konyha jellegére utal, továbbá az esetleges specialitásokat említettem meg ebben a mezőben. Ezeknél a leírásoknál elsősorban a helyek weblapját használtam, de sok esetben találtam szórólapot a bejáratoknál, melyekben megtalálható volt a szükséges információ. Az olyan éttermeket, amelyek szállodákban találhatóak és csak a szálloda vendégei vehetik igénybe a szolgáltatásait, nem képezik részét az éttermek táblázatának.



4.2.2. Szállodák

A szállodák nevű táblázatba minden olyan intézményt belevettem, amely az otthonuktól átmeneti időre távol levőknek ellenszolgáltatásért szállást ad, amely többnyire ellátást és kiszolgálást nyújt. Ezekből 45 darabot mértem fel. Az attribútum táblázat tartalma egy szállodáról:


- név
- minősítés
- irányítószám
- közterület név
- házsám
- leírás

A leírás a szállodát és különlegességeit mutatja be pár mondatban. Ezeket az információkat legtöbbször a szálloda vagy online szállásfoglaló honlapokról (pl.

<http://www.booking.com>) honlapjáról szereztem be. Magyarországon a szálloda minősítése ("hány csillagos") egy összetett szempontrendszer szerint történik, mely a következőkre vonatkozik: szobák nagyságára, berendezésére és felszereltségére, a vizesblokk mennyiségére, a személyzet öltözékére, a kötelező szolgáltatásokra, fakultatív szolgáltatásokra.

Szállodák

GPS Pinpoint

 Show on map

Accuracy: 36m

Név

Minosites

Ir. szám

Kozterulet

Send Queue Map Settings

8. ábra: A szállodák táblázatának egy része a felmérés során az MDC felületén.

4.2.3. Látnivalók

Ide tartoznak a kerületben található nagyobb szobrok, szökőkutak, templomok, illetve a Parlament épületét is itt tüntettem fel. Ezekből 46 darabot mértem fel az adatbázisomba. Az adattáblázat tartalma egy objektumról:

- név
- típus
- irányítószám
- közterület név
- házszám
- leírás

Az utcákban lévő szobrok és szökőkutak esetében a hozzájuk legközelebb lévő épület házszámát adtam meg, ezáltal könnyebben megállapítható, hogy az utca mely részén található.

Mindhárom táblázatnál ékezetek nélkül írtam a mezőneveket, mert az exportálás során az ékezeteket nem tudta megfelelően értelmezni a program, ezzel a problémámmal a GIS Cloud-ot is megkerestem, de ők se tudtak ebben a segítségemre lenni. Az egyes rövidítésekre pedig azért volt szükség, mert a shape fájl fejlécmezője maximum tíz karakter hosszú lehet.

4.3. Használt módszer

A terepi felmérés során, a már korábban említett GPS módszert használtam. A legtöbb helymeghatározási rendszerrel felszerelt telefonban csak egy egyszerűbb GPS-vevőt és kisebb kapacitású processzort találunk, ezért vannak beépített segédprogramok, mint például az A-GPS (Assisted-GPS), ez a helymeghatározáshoz szükséges adatokat mobilszolgáltatón vagy Wi-Fi hálózaton keresztül kapja meg. A másik ehhez hasonló az EPO (Extended Prediction Orbit), mely segítségével az almanachhoz hasonló, 30 napig érvényes adatokat tudunk letölteni, ezáltal kevesebb idő kell a helymeghatározáshoz (<http://www.gtop-tech.com>).

4.4. Felmérés menete

Miután a telefon megállapította a helyzetemet és a pontosság eltérése is 10 méter alá csökkent, elkezdtem bejárni a kerület utcáit és felmérni az idegenforgalmi kötődésű objektumokat. A felmérés során egy térkép segítségével haladtam, hogy véletlenül se

maradjon ki egyetlen utca sem, ez a folyamat több napot is igénybe tud venni. Amikor egy felmérni kívánt ponthoz érkeztem, kiválasztottam a megfelelő projektet, amibe az besorolható, és a 'send' gomb segítségével hozzá tudtam adni a felmért pontokhoz. Ezeket az adatokat a telefon tárolta, mivel ki volt kapcsolva a hálózati hozzáférés. Az esetek többségében az egyes pontokhoz tartozó leíró adatokat utólag, számítógép használatával vittem be, mert a terepen nagyon sok időt vett volna igénybe, illetve az objektum honlapja nélkül még nehezebb lett volna.

A felmérés során sokszor vettem fel pontot keskeny utcában, melyet oldalról magas bérházak határoltak. Az ilyen esetekben a mérés az átlagosnál nagyobb hibával terhelt, ugyanis nem megfelelő az égboltra való rálátás, illetve a falakról visszavert jelek is akadályozzák a pontos helymeghatározást.

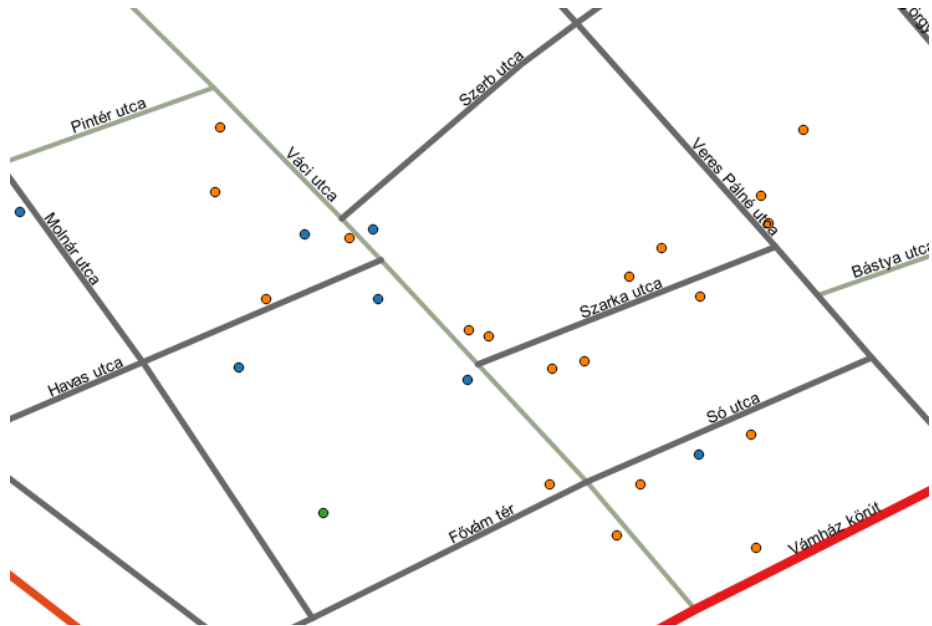
4.5. Térképi megjelenítés

A pontok felvétele után a GIS Cloud honlapjáról a három projektet shape fájlként exportáltam majd mindegyiket megnyitottam a QGIS-ben, a korábbi alaptérképet tartalmazó projektbe. Első lépésként áttranszformáltam EOVI vetületbe az új rétegeket és a karakterkódolásokat UTF-8-ra állítottam. A felmérés során az eredeti mezőkhöz az MDC három mezőt automatikusan hozzáad, melyek tartalmazzák az elem létrehozásának és módosításának dátumát, illetve annak a személynek a regisztrációs kódját aki a felmérést végezte, ez a három mező számomra felesleges volt, ezért töröltem.

Rétegszerkezetet tekintve fontos volt, hogy az éttermek, szállodák és látnivalók az utak rétege felett helyezkedjenek el, ha esetleg nagyobb méretarányban nézzük akkor se legyenek takarásban. Az egymáshoz képesti viszony nem volt lényeges, az előbb leírt sorrendben helyeztem el őket.

A különböző rétegek pontjait különböző színkitöltéssel rendelkező körökkel adtam meg, az éttermeké narancssárga, a szállodáké kék, a látnivalóké pedig zöld. Egy rétegen belül nem akartam további megjelenítési kategóriákat létrehozni, mert a sok típus miatt nagyon zsúfolt és színes lett volna a térkép.

Az exportált rétegeim pontjai sok esetben nem a megfelelő helyen helyezkedtek el a mérési hibák miatt, ezt a csomópont szerkesztő eszközzel korrigáltam, úgy elhelyezve azokat, hogy a valódi helyükön legyenek.



9. ábra: Mérési hibák

A megfelelő pozíció meghatározásához a Google Earth nevű programban kerestem rá az egyes objektumok címére. Ezt követően a hiányzó adatokat pótoltam az egyes rétegek attribútum táblázataiban.



10. ábra: Mérési hibák javítása után

5. Webes megjelenítés

A következőkben bemutatom a webes megjelenítés létrehozását és annak utólagos módosításának folyamatait.

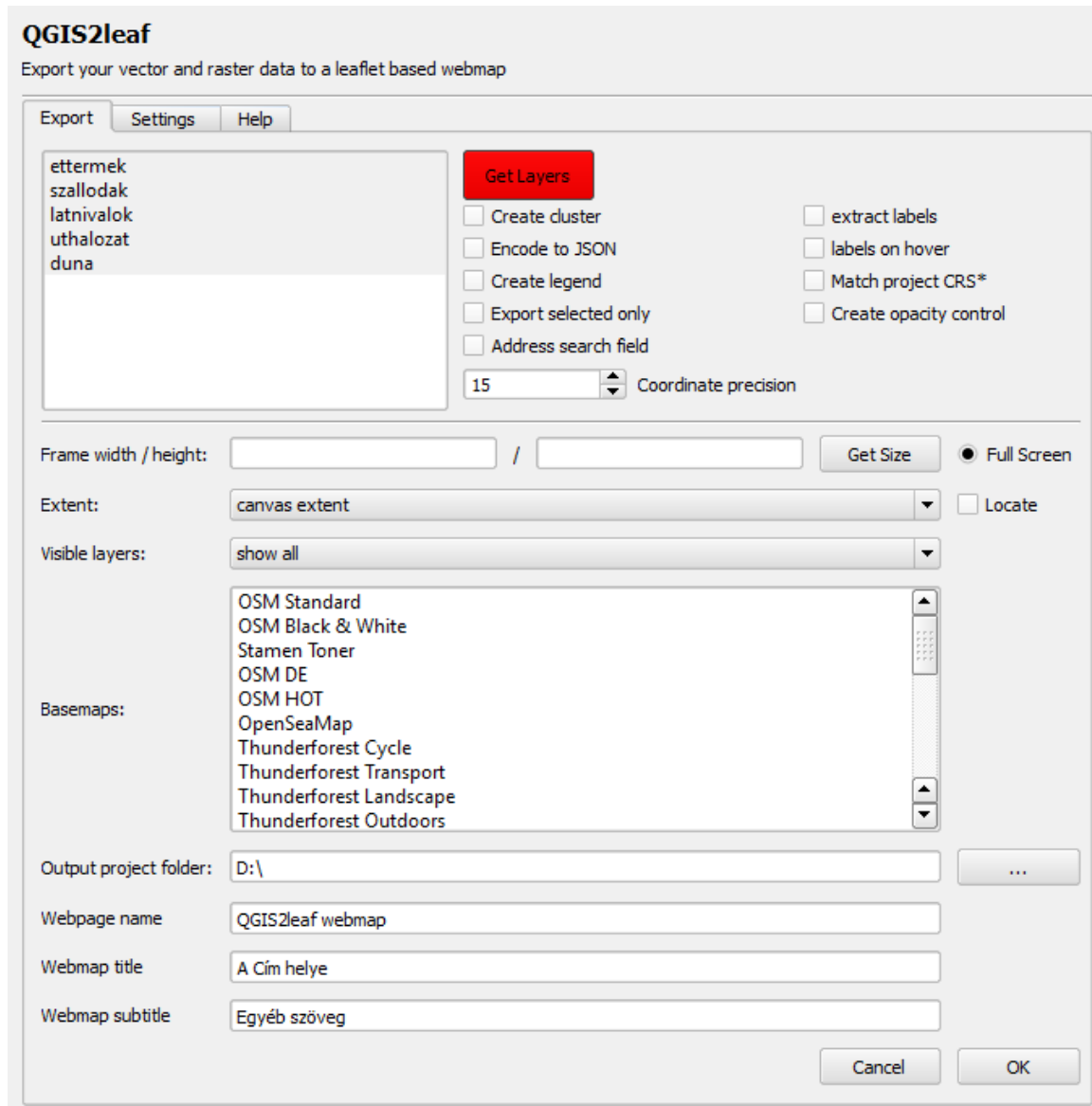
5.1. Leaflet

A Leaflet egy nyílt forráskódú JavaScript könyvtár, mobilbarát interaktív térképek számára. A JavaScript egy olyan objektum-orientált parancsnyelv, amely segítségével létrehozhatunk interaktív weboldalakat. Adatsere formátumként JSON-t (JavaScript Object Notation) használ, melynek szöveges leírása van (<http://www.json.org>). A Leaflet honlapján (<http://leafletjs.com>) az API (Application Programming Interface) fülre kattintva található meg az alkalmazásprogramozási felületet, mely egy program azon eljárásainak/szolgáltatásainak és azok használatának dokumentációja, amelyet más programok felhasználhatnak. Egy nyilvános API segítségével lehetséges egy programrendszer szolgáltatásait használni anélkül, hogy annak belső működését ismerni kellene.

5.2. Előkészítés és létrehozás

A későbbi képmegjelenítés érdekében az éttermek, szállodák és látnivalók attribútum táblázatában létrehoztam egy új mezőt "fenykep" néven, ide az adott idegenforgalmi egységet ábrázoló fénykép nevét írtam be a kiterjesztésével együtt, ha esetleg egy ponthoz több képet is hozzárendeltem, akkor a képneveket vesszővel választottam el szóköz nélkül. Ezeket a képeket az internetről töltöttem le, általában a hely honlapjáról. Nem minden ponthoz rendeltem képet, úgy gondoltam az egyszerűbb utcai falatozókról és hasonlókról nem szükséges kép.

A QGIS-en belül telepíthetjük a QGIS2leaf nevű bővítményt, mely használatával az aktuális vektoros és raszteres formátumunkból egy webes térképet készíthetünk. A bővítmény elindításakor be tudjuk állítani, hogy mely rétegeinket használja fel, mekkora méretben jelenítse meg, tudunk neki címet adni és megadni a létrehozott fájl helyét, illetve háttértérképet is beállíthatunk. Vannak további beállítások, mint például a jelmagyarázat létrehozása, de ezek a bővítmény kezdetlegessége miatt nem nagyon működnek. Célszerű a rétegek nevét ékezetek nélkül írni, különben a későbbi programozásban nehezebb dolgunk lesz.



11. ábra: A QGIS2leaf bővítmény panel

A képen látható rétegeket importáltam be a piros 'Get Layers' gomb segítségével és állítottam be teljes képernyőre a megjelenítést. Az 'ok' gombra kattintva a kijelölt helyen létrehoz egy mappát export és exportálás dátuma (pl. export20150426) néven. Ebben a mappában található index néven egy HTML (Hyper Text Markup Language) fájl, mely tartalmazza azokat a szimbólumokat, amelyek a megjelenítő programnak leírják, hogyan is kell megjeleníteni, illetve feldolgozni az adott állomány tartalmát. A HTML fájl mellett további mappák találhatóak:

- data - az egyes rétegek elemeit és azok adatait tárolja
- css - a stílusleírást tárolja
- js - az adatcsere-formátum leírását tárolja
- pictures - a képeket tárolja

A pictures nevű mappában létrehoztam három további a tematikus rétegek képei számára.

Egy böngésző segítségével megnyitottam az előbb létrehozott index.html fájlt, és első ránézésre is szembetűnő hiányosságok vehetők észre. Az úthálózat kategóriáinak a szimbólum szint beállítását nem őrizte meg az exportálás során, hanem az attribútum táblázat szerinti sorrendben következtek az egyes útszakaszok, ezáltal sok helyen az alacsonyabb szintű út fedte a magasabb szintű utat. A másik nagyobb hiányosság az utcanevek kiírása volt, a bővítmény a probléma áthidalására létrehozott egy JavaScript-es metódust, mely által az út bármely részére rákattintva felugrik egy ablak és kiírja az adott elem adatait. Ez a felugró ablak minden egyes rétegnél szerepelt.



12. ábra: A Gyűrű Pub felugró ablaka

Továbbá létrehozott egy rétegkezelő ablakot, ahol a meghívott rétegeket lehet ki-be kapcsolni.

5.3. Módosítások

Az előzőekben leírt hibák javítására és a további átalakításokra a Notepad++ forráskód szerkesztőt használtam, ebben nyitottam meg az index.html fájlt. A programozási munkálatokhoz a korábban már említett LeafLet API-t, illetve a GitHub honlapját (<https://github.com>) használtam, ahol a nyílt forráskódú rendszerek funkcionalitásait és leírásait lehet megtekinteni.

Elsősorban az utak megfelelő szimbólum szintjét oldottam meg a következő módon:

```
var propTypes = ["footway", "residential", "tertiary", "secondary"],
exp_uthalozat.features.sort(function(feature1, feature2) {
    var idx1= propTypes.indexOf(feature1.properties.type);
    var idx2= propTypes.indexOf(feature2.properties.type);
    return idx1-idx2;
});
```

Először megadtam a sorrendet, ezáltal az első kategória elemei kerültek a megjelenítés aljára, majd a JavaScript tömb objektum 'sort' metódusát hívtam meg, amelynek a paramétere két paraméteres függvény, ahol a rendezési algoritmus ezt a függvényt hívja meg a tömb két elemének összehasonlítására. A függvény visszatérési értéke negatív ha $idx1 < idx2$, nulla ha $idx1 = idx2$ és pozitív ha $idx1 > idx2$.

Az utak név kiírását nem lehetett úgy megoldani, mint ahogy a QGIS-ben is szerepel (automatikus címkékkel). A könnyebb és gyorsabb tájékozódás érdekében olyanra alakítottam, hogy nem kell rákattintani a közterület név kiírásához, elég csak ráhúzni a kurzort és megjelenik egy felugró ablakban.

```
<link rel="stylesheet" href="js/libs/leaflet.label/
leaflet.label.css" /><script
<script src="js/libs/leaflet.label/leaflet.label.js"></script>

var exp_uthalozatJSON= new L.geoJson(exp_uthalozat, {
    onEachFeature: function(feature, layer) {
        layer.bindLabel( feature.properties.name
            , {noHide:true, opacity:0.8, direction:"auto"});
    },
    style: doStyleuthalozat
});
```

Az első sorban a CSS (Cascading Style Sheets) bővítmény betöltését csináltam meg, mely egy stílusleíró nyelv, a strukturált dokumentumok megjelenítését írja le. Ezt követően a HTML bővítmény betöltése, ami a struktúrát írja le. A 'bindLabel' metódussal egy adott ponthoz címkét lehet rendelni. Az 'onEachFeature' minden elemhez meghívja a megadott függvényt, ezáltal minden egyes útszakaszhoz címke tartozik, mely az egyes elemek nevét tartalmazza, az utak típusát nem tartottam fontosnak megjeleníteni. Az 'opacity' segítségével az átlátszóságot lehet változtatni a címkének, ezt 80%-ra állítottam be, hogy ne takarjon ki adott esetben pontokat.



13. ábra: Létrehozott címke

Az egyes mappákban elhelyezett képek megjelenítésének létrehozása:

```
<link rel="stylesheet" href="js/libs/lightbox/css/lightbox.css" />
<script src="js/libs/lightbox/js/lightbox.min.js"></script>

function genImgLinks(feature, folder) {
    var fenykep = '';
    if( feature.properties['fenykep'] != null ) {
        var nev    = String(feature.properties['nev']);
        var aFiles = String(feature.properties['fenykep']).split(',');
        for(f=0; f<aFiles.length;f++) {
            if( aFiles[f].trim().length > 0 ) {
                fenykep+= '<a href="pictures/'+folder+'/' +
aFiles[f].trim()
                        + '" data-lightbox="' + nev +'" '
                        + 'data-title="'+nev
                        + '">'+(f+1)+' </a>';
            }
        }
    }
    return fenykep;
}
```

A képek megjelenítéséhez a Lightbox bővítményt töltöttem be, mely használatához a jQuery JavaScript könyvtár szükséges. Az éttermek, szállodák és látnivalók rétegeimhez akartam képmegjelenítést megvalósítani, így létrehoztam egy közös függvényt, amely mind a három helyen meghívódik, így elég csak egyszer leírni. A közös függvény neve 'genImgLinks'. A felugró ablakba generáltam a fényképek megjelenítésére szolgáló HTML linkeket. A kép megjelenítését a Lightbox végzi. Ha egy elem nem tartalmaz fényképet, azaz a 'fenykep' nevű mezője üres, akkor üres szöveget ír ki. Előfordult olyan eset is, amikor egy elemhez több képet rendeltem hozzá, ebben az esetben a 'split' függvénnyel a vesszők mentén feldaraboltam. A felugró ablakban a képek sorszáma jelenik meg szóközzel elválasztva, ezekre rákattintva nyílik meg a kép. A sorszámokhoz egyet hozzáadtam, hogy ne nullával kezdődjön.

Az index.html fájlt megnyitottam a Firefox Developer Edition böngészőben, mely kifejezetten web fejlesztés céljára készült. Itt a PASTA. gyorséterem felugró ablakában szereplő képmegjelenítést vizsgáltam meg az első kép esetében:

```
<a href="pictures/ettermek/pasta_1.png" data-lightbox="PASTA." data-title="PASTA.">1 </a>1 </a>
```

Ebből is látszik, hogy a függvénymeghívás rendesen működik, mivel a hivatkozási helyekre, mint például 'folder' vagy 'nev' a megfelelő mappanevet és képnevet helyezte be. A megnyíló kép bal alsó sarkában a felugró ablakhoz tartozó pont neve jelenik meg, illetve több kép esetén a képek sorszámát és hogy mennyi kép van összesen. A kép szélére kattintva vagy jobb/bal gomb megnyomásával tudjuk megnézni a vele szomszédos képet.

A felugró ablakban megjelenő mezőnevek és a képernyő jobb felső sarkában lévő rétegkezelőben lévő rétegek nevei a QGIS-ből öröklődtek át, amelyek nem tartalmaztak ékezetet és egyes mezőnevek rövidítve voltak.

Egy mező hivatkozása:

```
+ '</td></tr><tr><th scope="row">Házszám</th><td>'
+Autolinker.link(feature.properties['hazsam']==null?'':String
(feature.properties['hazsam']))
```

Ebben a szövegrészletben írtam át, hogy a felugró ablakban a mezőnevek ékezzel és nagy kezdőbetűvel szerepeljenek. Aláhúzással jelöltem amit ártam. Ezt a műveletet minden mezőnévnél és rétegnél megcsináltam.

A rétegkezelő:

```
L.control.layers({}, { "Éttermek" : exp_ettermekJSON
, "Szállodák" : exp_szallodakJSON
, "Látnivalók" : exp_latnivalokJSON
}
, {collapsed:true}).addTo(map);
```

Ártam a megfelelő nevekre, és eltávolítottam az úthálózat és a Dunát ábrázoló rétegeket, hogy csak ennek a három rétegnek tudjuk a láthatóságát változtatni. Olykor a rétegkezelő nagyobb méretarányban kitakart egyes pontokat, ezért a 'collapsed:true' felhasználásával létrehoztam egy kis ikont, amelyre a kurzort helyezve megjelenik a rétegkezelő, ezáltal csak akkor van szem előtt, ha tényleg szükségünk van rá.

Szépséghibának tartottam, hogy a különböző nyitva tartási időpontok egy sorban szerepelnek, és sokszor egy időintervallum közben volt a sortörés. A QGIS-ben az egyes időpontokat vesszővel választottam el egymástól, és ezek mentén akartam sortörést készíteni. Ezt a 'break' paranccsal oldottam meg. Több száz elemnél időigényes minden vessző helyére beírni a '
' címkét, ezért egy metódust hoztam létre:

```
replace(/,/g, '<br>')
```

Ezt a módszert a nyitva tartás mezőt meghívó parancs végére illesztettem. Ennek segítségével minden vessző lecserélődik a fent megadott címkére, a 'g' betű a globálisra utal, e nélkül csak az első vesszőt cserélnénk le.

A 'css' mappában az own_style.css fájlban belül létrehoztam egy stílusleírást, mely által az egyes elemek felugró ablakaiban a 'nev' mezőhöz tartozó adat más betűtípussal jelenik meg, ezzel is hangsúlyozva az egyik legfontosabb adatot a táblázatban. Erre a stílusleírásra az egyes rétegek elemeinek nevét tartalmazó mező meghívásánál hivatkoztam rá.

A név stílusleírása:

```
.featurename {
    font-family: sans-serif;
    font-size: 18px;
    font-style: bold;
    color: #000000;
}
```

Hivatkozás rá:

```
'<table><tr><th scope="row">Név</th><td class="featurename">'
```

A szállodák minősítését egyszerű szorzásjellel (*) adtam meg a QGIS-ben, de ez a felugró ablakban nem jelent meg szépen és olvashatóan, nehezen lehetett kivenni a csillagok tényleges számát. Az előzőekben leírtakhoz hasonló módon jártam el, ugyanúgy létrehoztam egy stílusleírást a minősítés megjelenítésére. Ebben a leírásban megadtam a méretét pixelben és az arany egyik árnyalatának a színkódját. A szorzásjelet pedig a szebb megjelenítés érdekében rendes csillagra cseréltem:

```
replace(/\/g, "\u2605")
```

A szorzásjel elé \ jelet kellett tenni, ugyanis bővített kifejezésnek számít és a perjel segítségével tudunk rá rendes karakterként hivatkozni. A csillagjelenek a karakterkódját adtam meg, hogy bármelyik szövegszerkesztő tudja értelmezni karakterkódolástól függetlenül.

A jelmagyarázat esetében a keretnek definiáltam egy stílusleírást, hány pixel méretű téglalapba kerüljön, illetve az átlátszóság mértékét. A HTML fájlban deklaráltam az éttermek, szállodák és látnivalók jelmagyarázatban lévő megjelenését.

```
var featureTypes =
[{"type": "Étterem", "color": '#ff7f00', "radius": 4.0}
, {"type": "Szálloda", "color": '#93aad5', "radius": 4.0}
, {"type": "Látnivaló", "color": '#33a02c', "radius": 4.0}];
var legend = L.control({position: 'bottomright'});
```

```

legend.onadd = function (map) {
    var div = L.DomUtil.create('div','info legend')
    for(f=0; f<featureTypes.length; f++) {
        div.innerHTML += '<i style="background:' +
            featureTypes[f].color + '"></i> ' +
            featureTypes[f].type + '<br>';
    }
    return div;
};

```

A deklaráció után pedig a Leaflet 'control' metódusát hívtam meg a jelmagyarázat létrehozásához.



14. ábra: A végleges címke egy szálloda esetében és a jelmagyarázat

Utolsó lépésként egy minimális és maximális nagyítási szintet adtam meg, elkerülve azt, ha esetleg túlságosan belenagyítanánk vagy kicsinyítenénk a térképen.

```

var map = L.map('map', { fullscreenControl: true
    , zoomControl:true
    , zoom: 31
    , minZoom: 30
    , maxZoom: 36
} )

```

A 'zoom' értéke a kezdő nagyítási szintet jelzi amikor megnyitjuk a térképet.

Az egér görgőjén kívül a bal felső sarokban található '+' és '-' jelű gombokra kattintva is lehet a nagyítás mértékét állítani.

5.4. Publikálás

Annak érdekében, hogy az elkészült térkép mindenki számára elérhető legyen, szükségem volt egy webserverre, mely elérhetővé teszi a helyileg tárolt weblapokat a HTTP (HyperText Transfer Protocol) protokollon keresztül. Az alapvető funkciója, hogy HTTP kéréseket fogad a hálózatról, és HTTP válaszokat küld vissza, melyek az esetek többségében egy HTML dokumentum. A tanszék webservéréhez kaptam jogosultságot, ezáltal az oda feltöltött térképemet az adatbázissal együtt bárki elérheti a mercator.elte.hu/~gyenesetamas/szakdolgozat címen.

6. Összefoglalás

Szakedolgozatom célja az V. kerület éttermeinek, szállodáinak és látnivalóinak terepi felmérése, ezek térinformatikai adatbázisának felépítése és térképi megjelenítése volt.

Dolgozatom első részében áttekintést adtam a kerület mai állapotáról és történelmének fontosabb állomásairól, majd részleteztem az alaptérképem átszerkesztésének folyamatait. Ezek után bemutattam egy telefonos applikációt, mellyel az egyes idegenforgalmi objektumokat mértem fel, amiket később a térképen jelenítettem meg. Végezetül a webes megjelenítést alakítottam ki, hogy mindenki számára elérhető legyen.

A felmérés során egy számomra új applikációval (GIS Cloud) ismerkedtem meg, melynek számos előnyét fedeztem fel, mint például, hogy saját magunk hozhatunk létre több különböző projektet az attribútum táblázatával együtt, továbbá megkönnyítette a feladatomat és időt is spórolt számomra, hogy lehetőség volt shape fájlként exportálni a felmért pontokat. Fontos pozitívum továbbá, hogy a terepi felmérés során az MDC-nek nem szükséges folyamatos internet hozzáférés, hanem tárolni tudja a felmért adatokat, ezáltal olyan terepen is végezhetünk méréseket, ahol nincsen Wi-Fi vagy mobilinternet lefedettség.

A webes megjelenítés létrehozása előtt úgy gondoltam, hogy a QGIS2leaf nevű bővítmény telepítése után egyből kész is a webes megjelenítés, ami ugyanúgy fog kinézni, mint a QGIS-ban létrehozott térképem, de a létrehozott webtérképen számos hiányosságot és hibát fedeztem fel. A bővítménynek újabb verziói hétről-hétre jelennek meg, de a szakdolgozatom elkészültéig nem jelent meg olyan verzió, mely tökéletesen reprezentálta volna a QGIS-ban létrehozott projektemet. Az előbb említett problémák áthidalása végett, szükséges volt jobban megismernem ezt a JavaScript könyvtárat, a Leaflet-et. A Leaflet és a GitHub oldala sokban hozzájárult a hibák kijavításában, ezeken az oldalakon számos leírás és példa a segítségemre volt. Ezeken az oldalakon sok olyan funkcióról olvastam, mely az én térképemnél nem, de egy más tematikájúnál szükséges lehet.

Úgy vélem, hogy a kitűzött célokat sikeresen teljesítettem, annak ellenére, hogy a webes megjelenítés létrehozásánál nem várt akadályba ütköztem, ami megoldása sok időt igényelt.

7. Hivatkozások

AGPS (Assisted GPS), http://www.gtoptech.com/en/software/Software_Services_07.html, utolsó hozzáférés: 2015.05.12.

Bibó István: Klasszicista épületegyüttes a Belvárosban, Budapest, 1972, in: Tanulmányok Budapest múltjából (19), pp. 221-230

Budapest V. kerülete, <http://www.budapestcity.org/04-kerulet/05/index-hu.htm>, utolsó hozzáférés: 2015.05.12.

Gárdonyi Albert: Pest városa a középkor alkonyán, Budapest, 1938, in: Tanulmányok Budapest múltjából (6), pp. 1-15

Gergely Gábor: Budapest kerületeinek közigazgatási változásai, Budapest, 2002, in: Tanulmányok Budapest múltjából (30), pp. 337-351

Geofabrik, <http://www.geofabrik.de>, utolsó hozzáférés: 2015.05.13.

Leaflet API reference, <http://leafletjs.com/reference.html>, utolsó hozzáférés: 2015.05.13.

Nagy Ágnes: Városnegyedek életciklusa: lakáelosztás - társadalmi rétegződés, Budapest, 2004, in: Tanulmányok Budapest múltjából (32), pp. 179-215

Nagy Lajos: Pest város eredete. (Előzetes jelentés az eskütéri ásatásokról), Budapest, 1934, in: Tanulmányok Budapest múltjából (3), pp. 7-24

Nagy Lajos: Rácok Budán és Pesten, 1686-1703, Budapest, 1959, in: Tanulmányok Budapest múltjából (13), pp. 80-103

Pallas Nagy Lexikona, Pallas Irodalmi és Nyomdai Rt., Budapest, 1893

Rajna György: Árvíztáblák Budapesten, Budapest, 1979, in: Tanulmányok Budapest múltjából (21), pp. 361-370

Schiffer János: Néhány szó Budapest egyesítéséről, Budapest, 1998, in: Tanulmányok Budapest múltjából (27), pp. 9-11

Step by Step Guide for GIS Cloud Applications,
http://www.giscloud.com/docs/GIS_Cloud_User_Manual.pdf?8f6417, utolsó hozzáférés: 2015.05.01.

Waffenschmidt Jánosné (szerk.): Magyarország közigazgatási helynévkönyve 2014. január 1., Központi Statisztikai hivatal, Budapest, 2014, ISSN 1217-2952

8. Ábrajegyzék

1. ábra: A kerület városrészei	4
2. ábra: Pest szabad királyi városa 1758-ban (A Pallas Nagy Lexikona, 1893).....	6
3. ábra: Levágás a "clip" parancs segítségével	10
4. ábra: Topológia javítás előtt	11
5. ábra: Topológia javítás után	11
6. ábra: Megjelenési kategóriák.....	12
7. ábra: Saját projekt létrehozása	14
8. ábra: A szállodák táblázatának egy része a felmérés során az MDC felületén.	16
9. ábra: Mérési hibák	19
10. ábra: Mérési hibák javítása után	19
11. ábra: A QGIS2leaf bővítmény panel	21
12. ábra: A Gyűrű Pub felugró ablaka	22
13. ábra: Létrehozott címke	24
14. ábra: A végleges címke egy szálloda esetében és a jelmagyarázat	27

9. Köszönetnyilvánítás

Jelen dolgozat nem jöhetett volna létre a körülöttem lévő emberek segítségével nélkül. Hálás köszönettel tartozom mindenk előtt Mészáros János konzulensemnek, aki idejét és fáradságát nem kímélve mindig a rendelkezésemre állt, ha éppen elakadtam valamiben.

10. Mellékletek

A DVD mellékleten megtalálható fájlok:

- jelen szakdolgozat pdf formátumban: gyenese_tamas_szakdolgozat_2015.pdf
- a létrehozott adatbázis: gyenese_tamas_szakdolgozat_2015_qgis
- Leaflet állományok: gyenese_tamas_szakdolgozat_2015_webes

11. Nyilatkozat

Alulírott, Gyenese Tamás 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, 2015. május 15.

.....
a hallgató aláírása