



M Ű E G Y E T E M 1 7 8 2

**Budapesti Műszaki és Gazdaságtudományi Egyetem
Közlekedésmérnöki és Járműmérnöki Kar
Közlekedésüzemi és Közlekedésgazdasági Tanszék**

Közúti kapcsolatok kezelésének integrálása az E-travel multimodális útvonaltervezőbe

TDK dolgozat

.....

Katona Géza
L018ZH

2013

Absztrakt: Gyakori kérdések egy utazás során a mikor?, mennyiért?, milyen módon? A társadalom, a technológia fejlődésével a mobilitás iránti igény nő, a mobilitás kultúra pedig változik. Előtérbe kerülnek a zöld technológiák, kiegészítő szolgáltatások, esélyegyenlőségi kérdések. Az interneten elérhető információ mennyiségének növekedésével, illetve a közösségi oldalak fejlődésével együtt a közlekedési rendszerek használói is magasabb igényeket támasztanak a szolgáltatókkal, szolgáltatásokkal szemben, továbbá az információ igényük is nő.

Ezen igények kielégítését tűztem ki az E-travel fejlesztésével, mely rendszerrel már másfél éve foglalkozom. A jelenlegi fázisban a közúti hálózat és a tömegközlekedési hálózat összekapcsolását fogom elvégezni. Emellett még a térképes megjelenítést fogom fejleszteni, melynél a rendelkezésre álló térképi adatbázisokra való illesztést fogom megoldani.

Kulcsszavak: Multimodális útvonaltervező, General Transit Feed Specification GTFS

Konzulens: Lénárt Balázs, tanársegéd

Tartalomjegyzék

1	BEVEZETÉS	4
2	A PROBLÉMAKÖR BEMUTATÁSA.....	6
2.1	PÉLDA	6
2.2	A TAPASZTALAT	6
2.3	VÍZIÓ	7
3	A KIINDULÓ RENDSZER BEMUTATÁSA	7
4	A BEVITT ADATOK BEMUTATÁSA.....	13
4.1	ADATOK GENERÁLÁSA.....	13
5	A KIINDULÓ RENDSZER FELHASZNÁLÓI FELÜLETE	16
5.1	KERESÉSI EREDMÉNYEK	17
5.1.1	<i>Egyszerű példa.....</i>	<i>18</i>
5.1.2	<i>Összetett városi példa.....</i>	<i>19</i>
5.1.3	<i>Összetett nemzetközi példa.....</i>	<i>20</i>
5.1.4	<i>A Bécsi hálózat példája</i>	<i>23</i>
5.1.5	<i>Légi utazás példája</i>	<i>24</i>
6	A JELENLEGI RENDSZERÁLLAPOT BEMUTATÁSA.....	26
6.1	GEOKÓD RENDELÉSE A CÍMEKHEZ [5].....	26
6.2	KÖZTES MEGÁLLÓK KILISTÁZÁSA	29
6.3	EREDMÉNYEK BEMUTATÁSA	31
6.4	TOVÁBBFEJLESZTÉSI LEHETŐSÉGEK	37
7	ÖSSZEFOGLALÁS	39
	IRODALOMJEGYZÉK.....	41
	ÁBRAJEGYZÉK.....	42

1 Bevezetés

A gazdaság és a közlekedés kapcsolatának kutatása során kimutatták, hogy a gazdaság fejlődésével együtt a közlekedési igények is növekednek. Napjainkra és várhatóan a jövőben is egyre növekedő helyváltoztatási igényeket kell a közlekedési rendszernek kiszolgálnia. A technikai fejlődés, az energiahordozók szűkös forrása, a területigény problémája a közeljövőben sokkal jobban előtérbe fog kerülni. Egyre több a tudatos vásárló, aki számára vásárlása előtt nagyon fontos a rendelkezésre álló információk mennyisége és minősége, hiszen döntésében nem csak az ár játszik szerepet. Ebből látható, hogy a problémakör nagyon összetett, emiatt pedig rengeteg szereplő lehet érintett a kérdésben. Ebből következően az áru- illetve a személyszállítási feladatok megszervezése, lebonyolítása igen komplex. A felhasználók részéről pedig az utazási igények a gazdasági fejlődéssel arányosan nőnek.

Az Európai Unió 2001-ben kiadott Fehér Könyve, az Európai közlekedéspolitika 2010-ig: itt az idő dönteni c. kiadványa [1] összefoglalja mindazon fő fejlesztési irányelvet, amelyet ennek az évtizednek az elejére meg kellett volna valósítani – de a projektek nagy része késésben van. A könyv legnagyobb része infrastrukturális beruházásokkal foglalkozik, egy kisebb fejezet szól az utasok jogairól. A fehér könyv csak érintőlegesen foglalkozik azzal, hogy egy piaci versenyhelyzetben a felhasználó (utas, megrendelő) a legfontosabb. Ezeket a szereplőket kell a középpontba állítani, hiszen az ő döntésük határozza meg a tényleges helyváltoztatásokat.

A felhasználóknak eddig kevés beleszólása volt például az egyes tömegközlekedési vállalatok árainak alakulásába, a különböző szolgáltatások létébe. (A fehér könyv kimondja, hogy mindenki annyit fizessen a szolgáltatásért, amennyibe az tényleg kerül.) A legtöbb felhasználónak elég kevés ismerete van az egyes piaci szereplőkről, azok számáról, a helyváltoztatással együtt járó, egyéb externális költségekről és hatásairól. A piac rendkívül torz, ennek oka a különböző országok különböző jogszabályi háttere, az állami vállalatok monopolhelyzete (a legtöbb országban), a változtatások érvényre juttatásának nehézsége (rengeteg munkaadó és munkavállaló). Egy autóval nem rendelkező utas kiszolgáltató, a legtöbb esetben nincs csak egy lehetősége (Budapest, BKV), illetve ha esetleg létezik más megoldás, nem tud róla.

A dolgozat motivációja a nemzetközi utazások megtervezésének megoldatlansága. Nap mint nap tapasztalható, hogy akár az országon belül is nehéz megtervezni, hogy milyen közlekedési eszközzel utazzunk, és mennyibe fog ez kerülni, azonban az ország-

határt átlépő utazásnál, főleg ha több országot is érint az út, a kedvezmény rendszerek, várható költségek összehasonlíthatatlanná válnak, így gyakorlatilag megtalálhatatlan az utas számára a kívánt költség vagy idő optimum, csak a szerencse kérdése, hogy milyen áron utazhat.

Ebben a dolgozatban a korábbi „Integrált személyközlekedési igénykezelő és irányító rendszer” című TDK-mat és az ezt követő „Multimodális útvonaltervező fejlesztése” című diplomamunkámat folytatom. Az ezekben a dolgozatokban elért eredményekre építettem a jelen dolgozatot, az ott elkészített rendszert fejlesztettem, bővítettem új funkciókkal.

2 A problémakör bemutatása

2.1 Példa

El akarunk jutni Budapestről Prágába. Ezt megtehetjük autóval, vonattal, repülőgéppel és autóbusszal. Az autós útvonaltervezés a legkönnyebb, rengeteg internetes kereső létezik, melyek üzemanyagköltséggel is tudnak számolni. De a legtöbb ilyen rendszer nem tudja kezelni az egyes útdíjakat (így akár meg is duplázódhat egy utazás költsége). Az egyes pályaszakaszok állapotáról sem adnak visszajelzést (baleset, hóvihár, karbantartás), az útközben lévő pihenőhelyek száma, elhelyezkedése, azok szolgáltatása is csak több tízperces utánajárással deríthető ki. Repülőgéppel való tervezéskor a megfelelő portálok segítségével azonnali minőségi, szolgáltatási információkhoz juthatunk, a jegyvásárlás gyors, megbízható. De a legtöbb kereső csak az alapárát adja meg, az esetleges plusz költségek (súlyhatár fölötti poggyász, nem internetes check-in, stb., de a repülőtéri illetéket már kötelező az árba bekalkulálni) nem jelennek meg, csak a különböző légitársaságok honlapjain, így a keresés időigénye, és a költségek jelentősen megnőhetnek. Vasúti utazáskor a keresőrendszer rögtön megadja az egyes alternatívákat, minőségi paramétereket, kiegészítő szolgáltatások listáját, de egy bonyolultabb nemzetközi utazás esetén a jegyfoglalás már nem lehetséges vagy csak korlátozott. Autóbusszal történő utazás esetén a legrosszabb a helyzet. Jelenleg több társaság közlekedtet járatokat (Eurolines-Volánbusz, Orangeways, Student Agency, stb.), ám nem létezik olyan internetes felület, amelyen egyesítve megjelenének, pedig legtöbb esetben áraik igen alacsonyok, fedélzeti szolgáltatásaik messze meghaladják a vasútét vagy légiközlekedését. (A Carpooling rendszerek elterjedtsége igen alacsony, ezért ebben a példában nem vettem figyelembe, mint lehetséges opciót.)

2.2 A tapasztalat

Az utazás megtervezése az összes lehetőség és kedvezmény figyelembe vételével időigényes, és nehezen kivitelezhető. Országhatárokon átlépő utazásnál pedig gyakorlatilag lehetetlen. Ennek oka, hogy az információk elkülönülten állnak csak rendelkezésre, és az átszállási kapcsolatokat ritkán veszik csak figyelembe, de ilyenkor is csak néhány

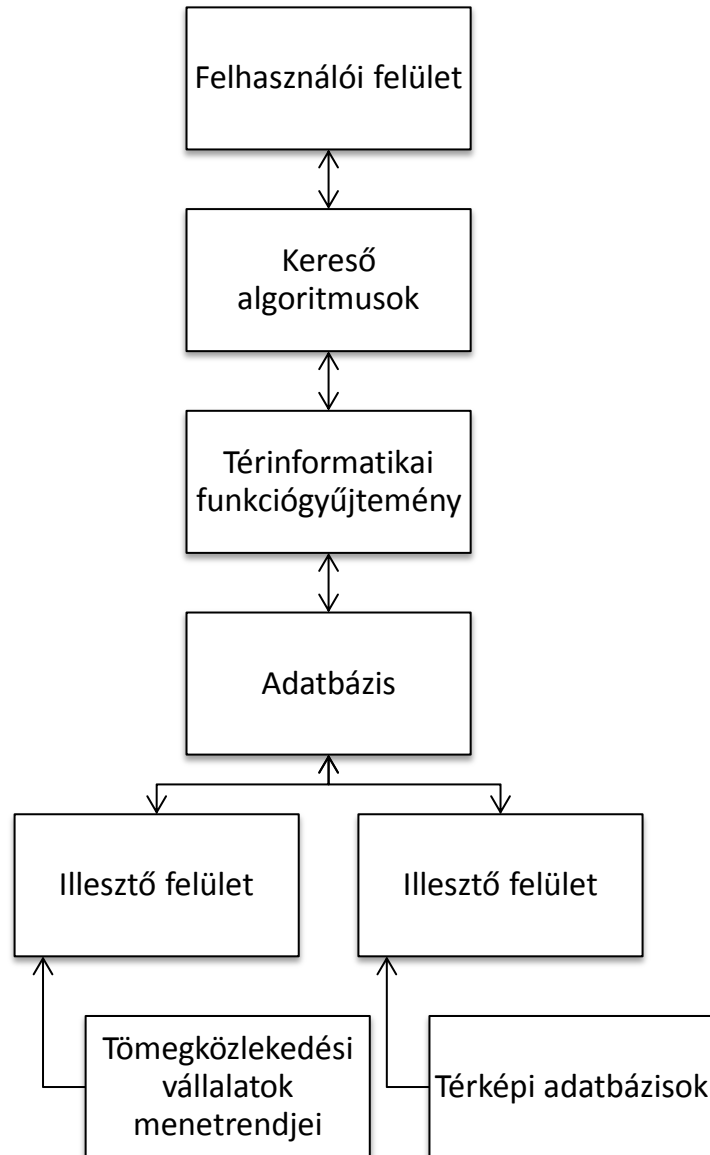
közlekedési mód között (Deutsche Bahn rendszere). A jelenlegi lehetőségeket figyelembe véve nem áll rendelkezésre egy olyan információs rendszer, mely az utasnak segítséget nyújtana, hogy két tetszőleges cím között, melyek nem csak másik városban, hanem akár másik országban is lehetnének, támogatást nyújtana az utazása megtervezéséhez. Egy ilyen rendszertől elvárható, hogy a távolságtól függően a kedvezmények, akciók figyelembe vételével a teljes közösségi szolgáltatási hálózat lehetőségeit figyelembe véve a lehető legjobb (leggyorsabb, legolcsóbb) eljutást megtervezze, a tényleges költség összetevőit megjelenítse. Ma Magyarországon két város közötti utazás megtervezése is nehézségekbe ütközik, ha vasutat, közutat és városi közlekedést is igénybe kíván venni az utazó. Jelenleg ezt három különböző felületen keresztül lehet megtenni, azonban e felületek az átszállási lehetőségekről, kapcsolódó járatokról csak minimális információt nyújtanak. Egy nemzetközi utazásnál pedig a nyelvi problémák, a szabályozási és tarifa rendszer ismeretének hiánya nehezíti tovább a tervezés folyamatát.

2.3 Vízió

A távlati cél egy olyan szolgáltatás létrehozása, mely a különböző utazási alternatívák között összehasonlítási lehetőséget teremt. Ezalatt azt értem, hogy a felhasználó számára a rendszer felajánlja mind az egyéni, mind a közösségi (ill. ezek kombinációja) lehetőségeket, melyek alapján a felhasználó egyszerűen, gyorsan a saját igényeit kielégítő döntést tud hozni. A rendszerben nagy hangsúlyt szeretnék helyezni a felhasználók magas szintű tájékoztatására, visszajelzéseiknek kezelésére, megjelenítésére. Továbbá lehetőséget szeretnék teremteni arra, hogy a keresések és jegyvásárlások eredményei elemezhetővé váljanak. Ezen statisztikai adatok alapjául szolgálhatnak az infrastruktúra fejlesztéseknek és menetrend tervezéseknek.

3 A kiinduló rendszer bemutatása

A kutatásaim alapján egy 5 szintű modell körvonalazódott a számomra. Ezt a modellt a 1. ábra mutatja be.



1. ábra Elvi rendszermodell

A tömegközlekedési vállalatok több formátumban tárolják, illetve szolgáltatják a menetrendi és foglalási adataikat. Ezek közül 3 fontos rendszert érdemes kiemelni. Az első ilyen a legrégebben működő rendszer a Sabre, mely a légitársaságok járatinformációit és foglalási adatait tartalmazza. Ezt a rendszert az American Airlines fejlesztette ki az 1950-es években, hogy automatizálhatóvá váljanak a jegyeladások. Mára a világ minden tájáról csatlakoznak a különböző utazási ügynökségek és légitársaságok, így a teljes légi piacot magába foglalja a rendszer. A második fontos az adatbázis a Német HaCon cég által üzemeltetett Hafas vasúti adatbázis [4], mely az európai vasutak menetrendi adatait tartalmazza. A Német vasúttársaság menetrendi keresője támaszkodik leginkább erre a rendszerre, ezáltal az Unió egész területére tud menetrendi ajánlatokat

adni, azonban foglalás nem lehetséges a rendszeren keresztül. A harmadik és egyben a legfiatalabb szereplő ezen a területen a Google cég által fejlesztett Google térkép szolgáltatás, mely a helyi tömegközlekedésre fókuszál. A Google ehhez kifejlesztett egy nyílt szabványt, melynek segítségével bármelyik tömegközlekedési vállalat szolgáltathat információt a menetrendjéről.

A szolgáltatás jó működéséhez szükségesek még térképi adatok is, melyek segítségével tetszőleges címek között is lehetővé válik a keresés, illetve ezáltal megjeleníthetővé válnak az eredmények. Ezen adatok hozzáférhetőek OSM formátumban, mely egy XML alapú leírása a közlekedési térképeknek.

Az első és a második szint között szükséges egy köztes lépés, mely a különböző forrásokból származó adatbázisokat a megfelelő egységes formátumba konvertálja, így használhatóvá válnak a rendszer számára.

A második szinten egy olyan programra van szükség, mely egy célzott adatbázis kezelő alkalmazás, ami képes igen nagyméretű adatbázisokat is kezelni. Erre a célra egy nyílt forráskódú alkalmazást fogok alkalmazni, a PostgreSQL nevű programot. Azért választottam ezt a programot, mert amellet, hogy nyílt forráskódú, olyan kiterjesztésekkel rendelkezik, melyek az adatbázis gyors kereshetőségében segítséget nyújthatnak, illetve több kereső algoritmust is integrálnak.

A harmadik szinten egy olyan alkalmazásra van szükség, mely lehetővé teszi, hogy egy rengeteg adatot tartalmazó adatbázis adatai gyorsan hozzáférhetőek legyenek. Mindez olyankor számít igazán, amikor folyamatos internetes kapcsolaton keresztül kéri le a felhasználók az adatokat. Erre a célra a PostgreSQL programcsomag PostGIS kiterjesztését alkalmaztam, mely lehetőséget ad nagyszámú függvény és operátor mező kezelésre (pl.: terület, kerület, hossz számítás, topológiai vizsgálatok, stb.), emellett GIST típusú térbeli indexek is alkalmazhatók, melyek lehetővé teszi a térképi adatok alapján való gyors keresést.

A negyedik szint célja, hogy az eddigi adatbázisokat felhasználva útvonal ajánlatokat tudjon adni a felhasználók számára két tetszőleges pont között. Erre több útvonal-kereső algoritmus is rendelkezésre áll, például a heurisztikus Dijkstra, az A-star és sok egyéb más lehetőség is. A PostgreSQL-nek a PgRouting kiterjesztése ezt a feladatot látja el, a legtöbb ilyen algoritmust tartalmazza a rendszer, így csak a használat szabhatárt a kereséseknek.

És végezetül az ötödik szint, mely a felhasználói felület. Olyan szempontból ez az egyik legfontosabb, hogy a tényleges felhasználók ezen a felületen keresztül fogják

majd elérni az adatokat. Fontos szem előtt tartani, hogy még ha a legjobb ajánlatokat képes adni a rendszer, ha mindezt rosszul közöljük a felhasználóval, akkor a rendszer minden lehetősége ellenére bukásra van ítélve. Ezért itt igen lényeges lesz, hogy egy közérthető, letisztult koncepciójú felület kerüljön kialakításra, mely mindenki számára világos és egyértelmű.

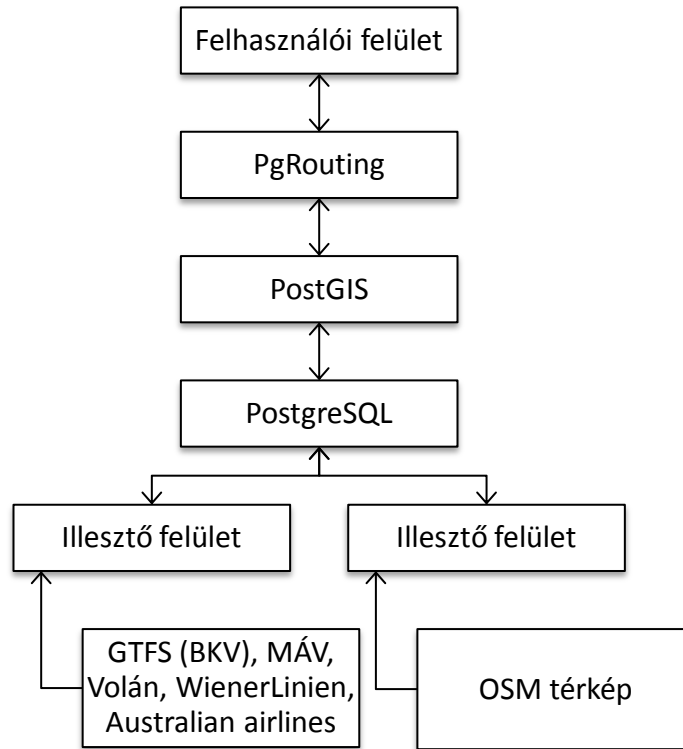
A TDK keretén belül a budapesti tömegközlekedés adatainak egy részét, a Budapest és Bécs közötti vasúti, autóbuszos és légi kapcsolatokat és a bécsi tömegközlekedés bizonyos részeit használtam fel a hálózat felépítéséhez. A 1. táblázatban található a viszonylatok listája. A kiválasztásnál szempont volt, hogy a városi közlekedés és a távolsági közlekedés kapcsolata legyen a hangsúlyos.

M1	Mexikói út M / Vörösmarty tér
M2	Őrs vezér tere M+H / Déli pályaudvar M
M3	Kőbánya-Kispest M / Újpest-központ M
4	ÚjBuda-központ / Széll Kálmán tér M
6	Móricz Zsigmond körtér / Széll Kálmán tér M
7	Bosnyák tér / Albertfalva vasútállomás
173	Bornemissza tér vá. / Újpalota, Nyírpalota út
200E	Határ út M / Liszt Ferenc Airport 2
Railjet	Budapest-Keleti / Wien Westbahnhof
912/942	Budapest-Keleti / Sopron
2821	Sopron / Wien Meidling
130	Budapest-Keleti / Bratislava Hlavná Stanica
2513	Bratislava Hlavná Stanica / Wien Hauptbahnhof
U1	Reumannplatz / Leopoldau
U2	Karlsplatz / Aspernstraße
U3	Ottakring / Simmering
U4	Hütteldorf / Heiligenstadt
U6	Siebenhirten / Floridsdorf
CAT	Bahnhof Wien Mitte / Flughafen Wien
S7	Rennweg / Flughafen Wien
Volán	Budapest, Népliget aut. pu. / Wien, VIB, U3 Station Erdberg
OS 714	BUD Nemzetközi repülőtér / Flughafen Wien

1. táblázat A viszonylatok listája

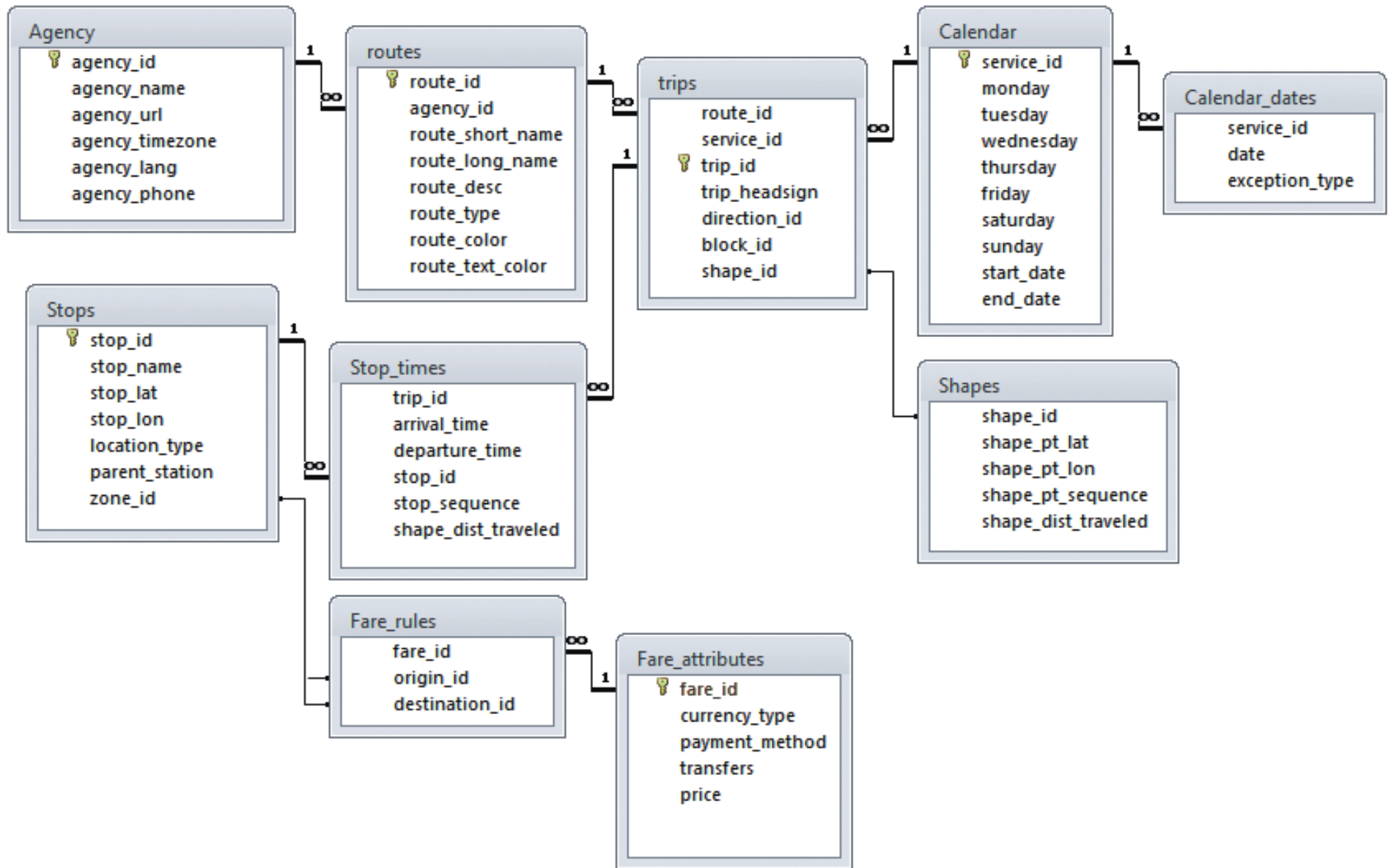
A feldolgozás a PostgreSQL adatbázis kezelő segítségével történt, melybe a Budapesti Közlekedési Központ, a MÁV, a Volánbusz, a Wiener Linien, az Australian Airlines és a Google által publikált adatait használtam fel. A rendszerbe a PgRouting által alkalmazott keresési algoritmusokkal fogom a kereséseket elvégezni.

Összefoglalva az eddigieket, a rendszer modellje a következő (2. ábra):



2. ábra Rendszermodell

A rendszer számára a GTFS specifikációja szerint 3. ábra által mutatott táblaszerkezetet készítettem el. Ezen az ábrán látható a teljes elméleti szerkezet, melynek részletezése megtalálható az 1-es számú mellékletben.



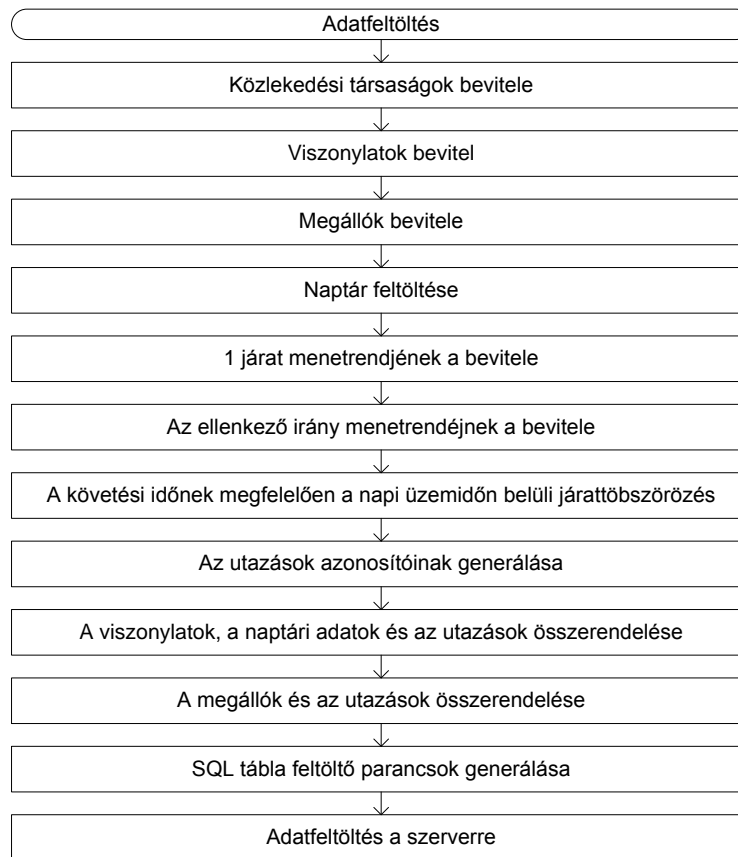
3. ábra Táblaszerkezet

4 A bevitt adatok bemutatása

4.1 Adatok generálása

Mivel az adatok különböző szolgáltatóktól származnak és ezek nem állnak rendelkezésre importálható formátumban, ezért az adatokat kézzel kellett bevinni. Erre kifejlesztettem egy generáló algoritmust, mely a megadott adatok alapján SQL parancs sorozatot készít, amelyekkel lekérdezések keretében soronként lehet feltölteni a táblákat. Ezt úgy valósítottam meg, hogy a járatok menetrendjének megfelelően felvettem egy járatot, majd egy adott követési időknél megfelelően megtöbbszöröztem. A TDK keretén belül az éjszakai közlekedéssel nem foglalkoztam, ezért csak a nappali üzemnek megfelelő adatokat vittem be a rendszerbe. Fontos, hogy azokat a járatokat, melyek éjjel után is járnak, azokat úgy definiáltam, hogy a 0, 1, 2... óra helyett 24, 25, 26... órát adtam meg.

Az algoritmus tetszőleges, táblázatosan adott adatokból képes a generálásra, azonban a teszt jelleg miatt szükség van az emberi beavatkozásra, mert a teljesen automatizált adatbevitel az adott helyzetben aránytalanul nagy erőbefektetéssel lenne csak megvalósítható. Az adatbevitel a következőképpen valósul meg (4. ábra):

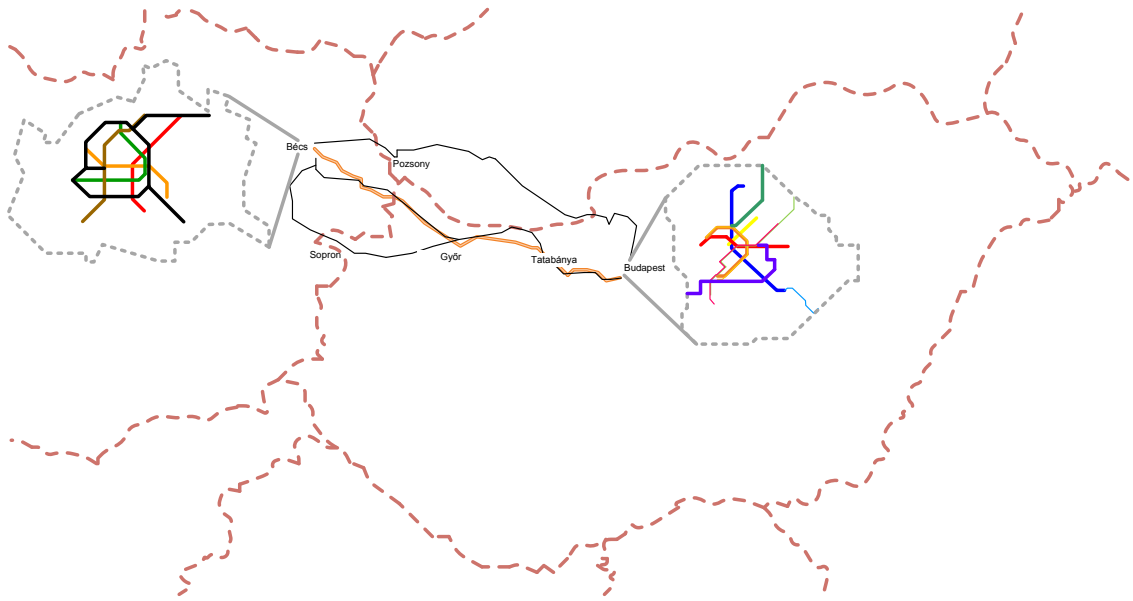


4. ábra Az adatbeviteli algoritmus folyamatábrája

Az egyszerűség kedvéért az egész üzemidő alatt azonos követési időket választottam, így a csúcsidőn kívül és a csúcsidei, illetve a különböző munkanapok és ünnepnapok illetve időszakos menetrendek nincsenek megkülönböztetve. Mivel egy teszt adatbázisról van szó, ezért ez nem okoz gondot, és nincs hatással a működésre sem, ezért megengedhető ez az elhanyagolás.

Ennek megfelelően a metrókhoz egységesen 2 perces követési időt választottam és 19 órás üzemidőt, a villamosokhoz 3 perceset és 18,5 órásat, a buszokhoz 5 illetve 10 perceset és ismét 19 órásat, a vonatokhoz 1 és 2 órás követési időt, a távolsági buszhoz és a repülőhöz pedig a járatok alacsony száma miatt a menetrendnek megfelelőt.

Az adatbevitel eredményeképpen egy Budapestet, Pozsonyt, Gyórt, Sopront és Bécset tartalmazó tömegközlekedési hálózat jött létre (5. ábra).

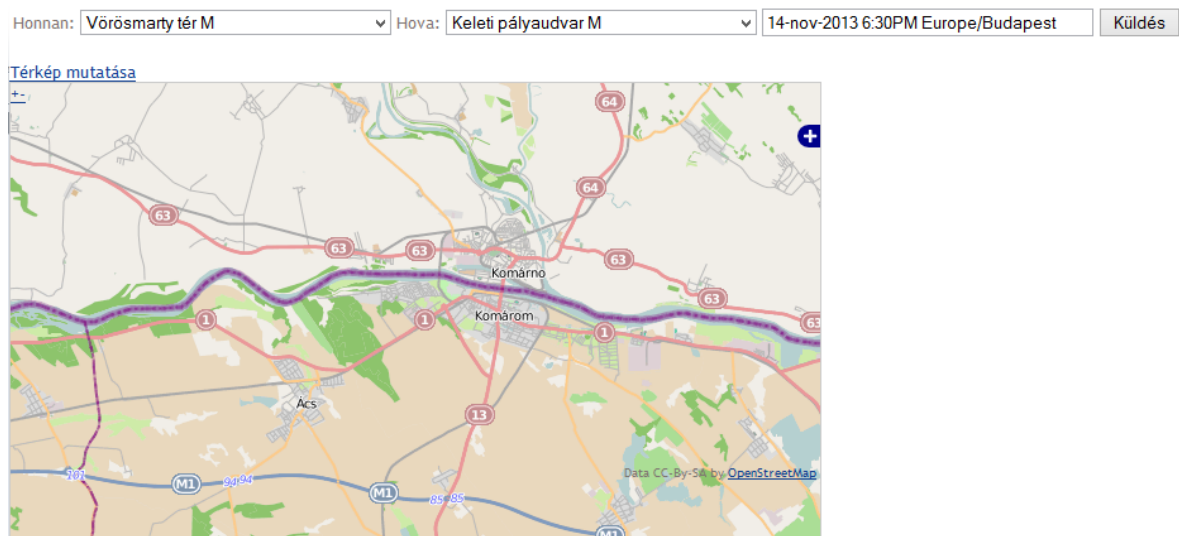


5. ábra Az adatbevitelnek megfelelő közlekedési hálózat

5 A kiinduló rendszer felhasználói felülete

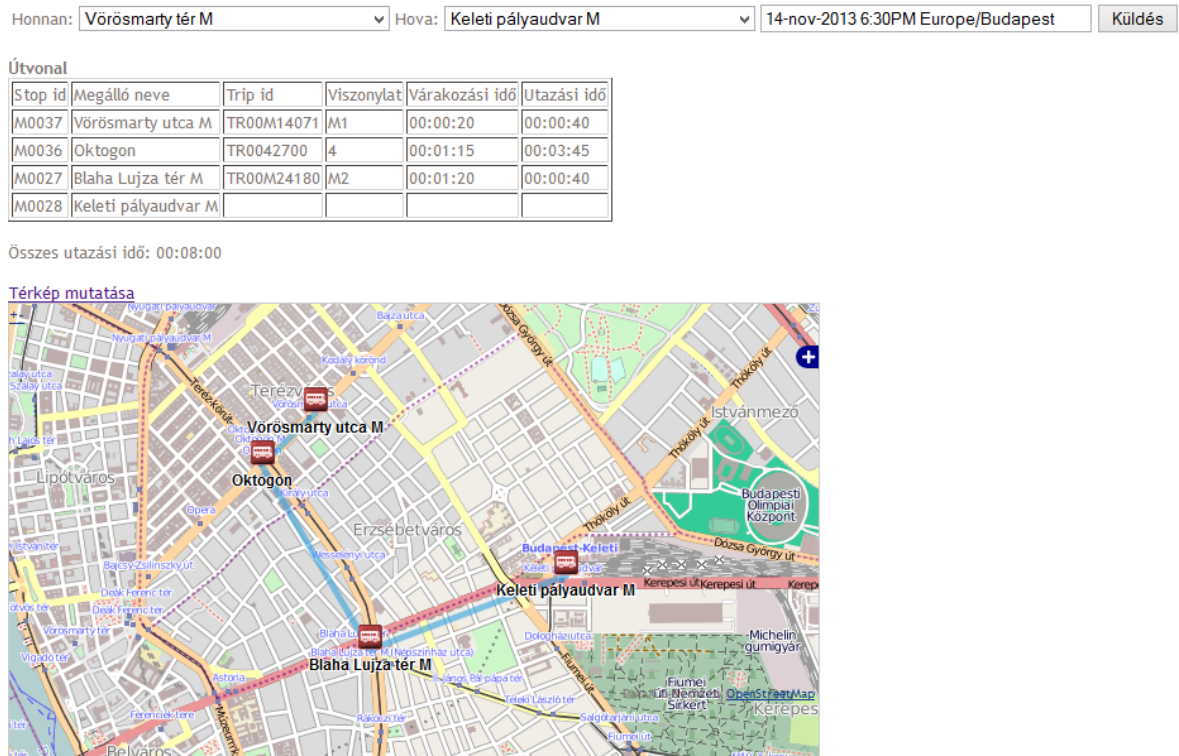
A teszteléshez elkészült egy teszt felhasználói felület, melynek a legfontosabb célja a működés ellenőrzése, ezért a felület egyszerűségére és funkcionalitására törekedtem. Azonban ez nem esik messze a vízióban megfogalmazott céloktól, mert egy jó felületnek egyszerűnek és átláthatónak kell lennie, de emellett az egyediség is igen fontos, mert főleg a bevezetésnél, de a későbbiekben is egy működő rendszernél rettentően lényeges, hogy elkülönüljön a többi rendszertől. Fontos, hogy egy önálló „márka” illetve látványvilág jellemezze az oldalt, mert ezzel lesz karaktere, így lehet üzenetet is kapcsolni mellé, illetve egy-egy hirdetésnél ezeknek az egyedi jeleknek a hangsúlyozásával építhető ki a célközönségben egy olyan behatás, melyhez az adott szimbólumrendszert kapcsolni lehet.

Az oldal PHP nyelven készült, mely JavaScript elemeket is tartalmaz. Az oldalon legördülő listából voltak kiválaszthatóak az állomások és egy beviteli mező segítségével volt megadható az utazás időpontja. A futatás eredménye egy lista volt, mely tartalmazta a kezdő és végmegállókat, az átszállási pontokat, a választandó járatot, az utazás és a várakozás időpontját.



6. ábra A kiinduló felhasználói felület

Az oldal képét a 6. ábra mutatja. Példaképpen megmutatható az 1-es metró Vörösmarty térértől való eljutás a Keleti pályaudvarhoz.



7. ábra A kiinduló rendszer adatbevitelének 1. példája

A 7. ábra mutatja a lekérdezés eredményét. Abban a verzióban azok a csomópontok voltak felsorolva, melyeknél mód- vagy eszközváltás történt, tehát a kezdő-, a vég- és az átszállási pontok. Megjelenítésre került még az utazások és a várakozások időtartama, illetve a teljes utazási idő.

A térkép azon verziójában az eredményben szereplő megállók vannak megjelenítve, amelyeket egyenes vonalak kötöttek össze.

5.1 Keresési eredmények

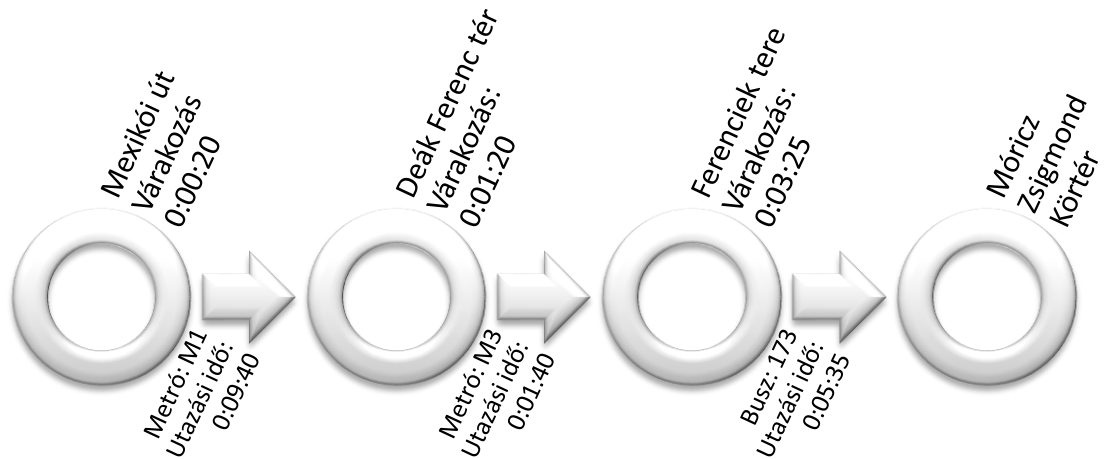
A következőkben a különböző szempontok szerint fogom bemutatni a keresési lehetőségeket, és az erre a rendszer által szolgáltatott válaszokat. Itt a legfontosabb cél az volt, hogy szimuláljam a valós kereséseket, illetve megvizsgáljam, hogy az algoritmus párhuzamos utak esetén milyen eredményeket szolgáltat, kedvezőbb, vagy azonos eljuttatás megjelenése esetén mely útvonalat preferálja az algoritmus.

5.1.1 Egyszerű példa

Kiindulópont: Mexikói út

Célpont: Móricz Zsigmond körtér

Az első példában egy budapesti keresést mutatok be, mellyel az alapvető működést szeretném demonstrálni. Ezen kiinduló és célpontokkal az alábbi eredmények születtek (8. ábra):



8. ábra A kiinduló rendszer futási eredményének útvonala (1. példa)

A térképes megjelenítés pedig (9. ábra):



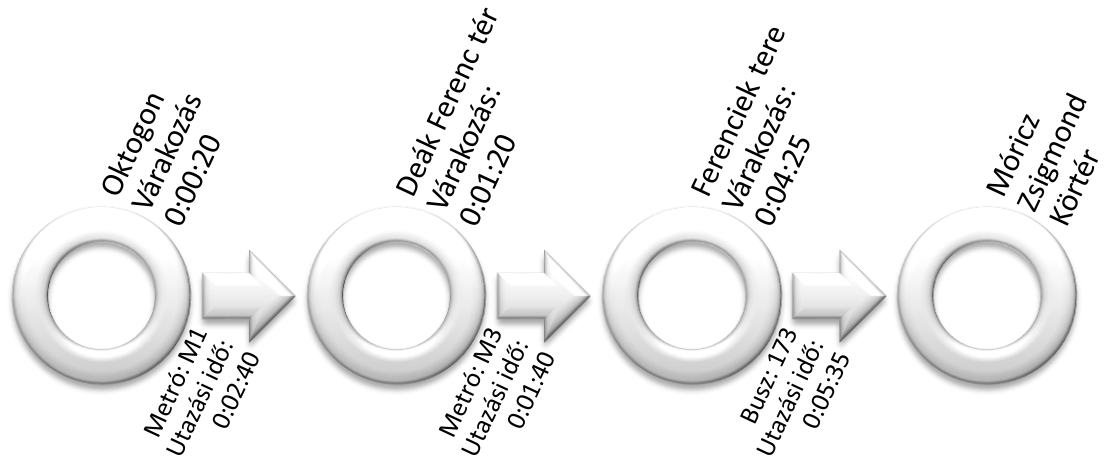
9. ábra A kiinduló rendszer futási eredményének térképe (1. példa)

5.1.2 Összetett városi példa

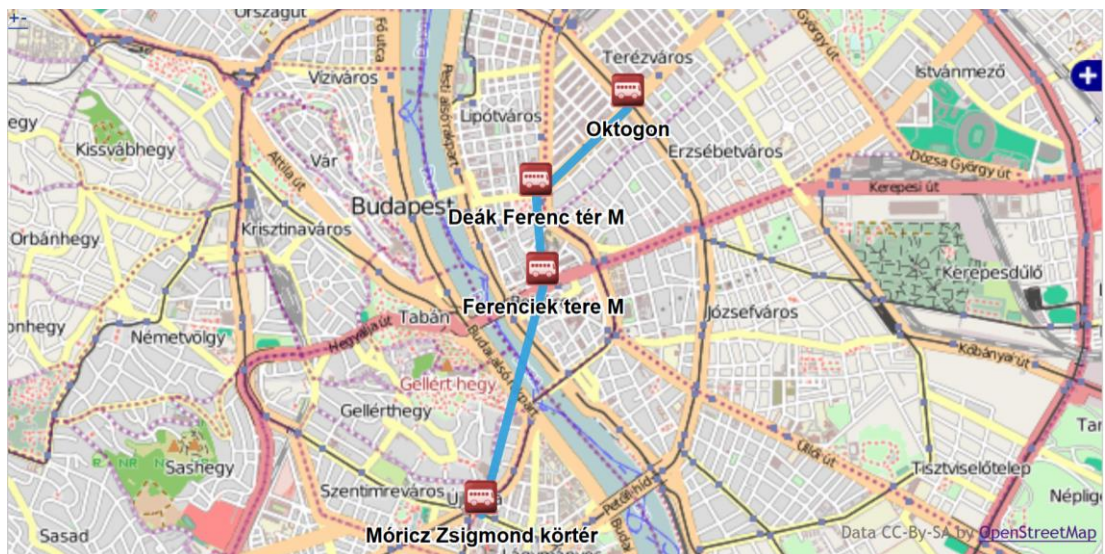
Kiindulópont: Oktogon

Célpont: Móricz Zsigmond körtér

A következő példában az Oktogontól való eljutási lehetőségeket fogom vizsgálni. A jelen hálózaton két kézenfekvő eljutási lehetőség áll rendelkezésre, az egyik az 1-es metró a Deák Ferenc térig, ahol át kell szállni a 3-as metróra, mellyel a Ferenciek Teréig kell utazni és ott lehet átszállni a 173-as buszra, mely elszállítja az utast a Móricz Zsigmond Körtérig. A másik verzió szerint a 6-os villamosra kell felszállni, és azzal lehet eljutni a Móricz Zsigmond Körtérre. Az első verzió futási eredményei a következők (10. ábra):

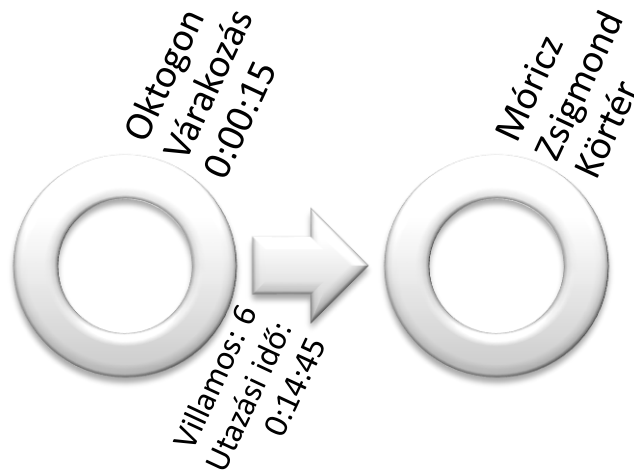


10. ábra A kiinduló rendszer futási eredményének útvonala (2a. példa)

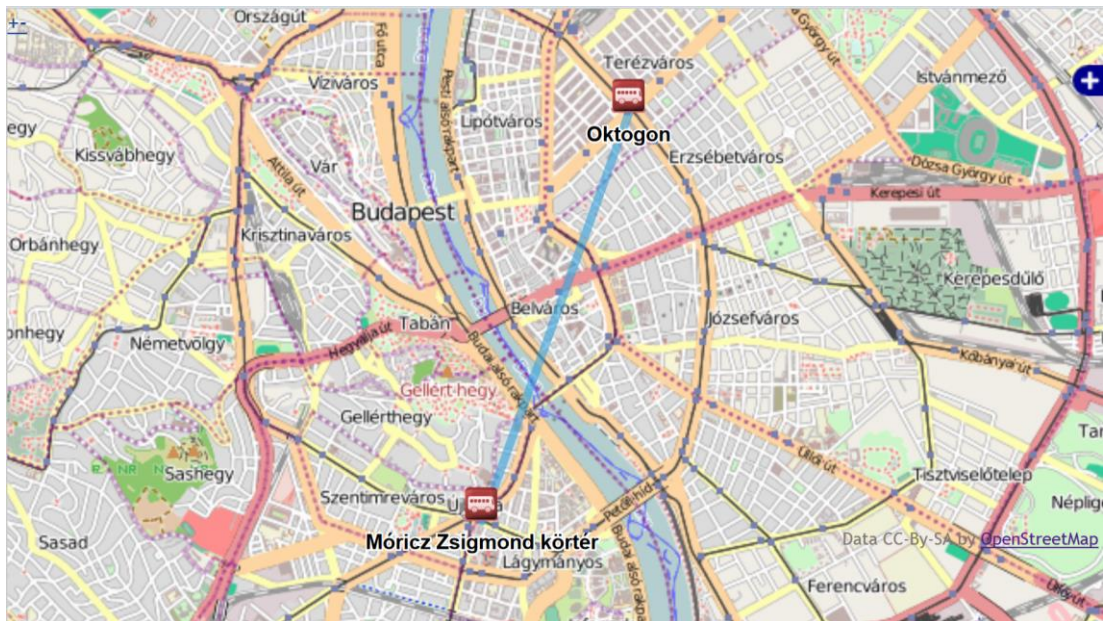


11. ábra A kiinduló rendszer futási eredményének térképe (2a. példa)

A második verzió eredményeit pedig 12. ábra mutatja.



12. ábra A kiinduló rendszer futási eredményének útvonala (2b. példa)



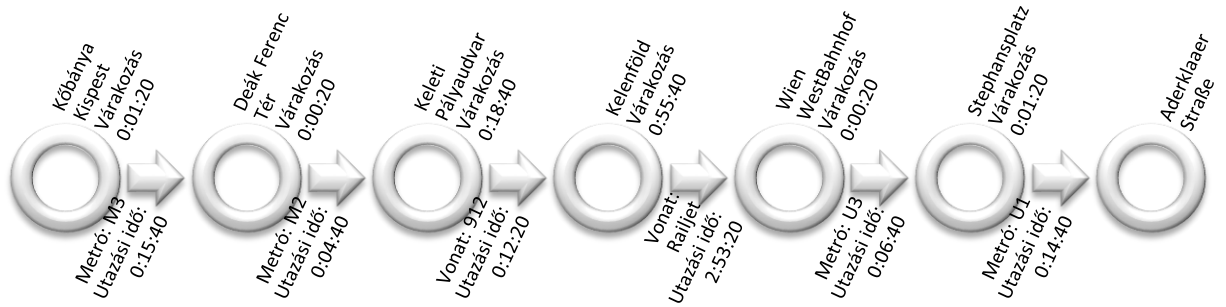
13. ábra A kiinduló rendszer futási eredményének térképe (2b. példa)

5.1.3 Összetett nemzetközi példa

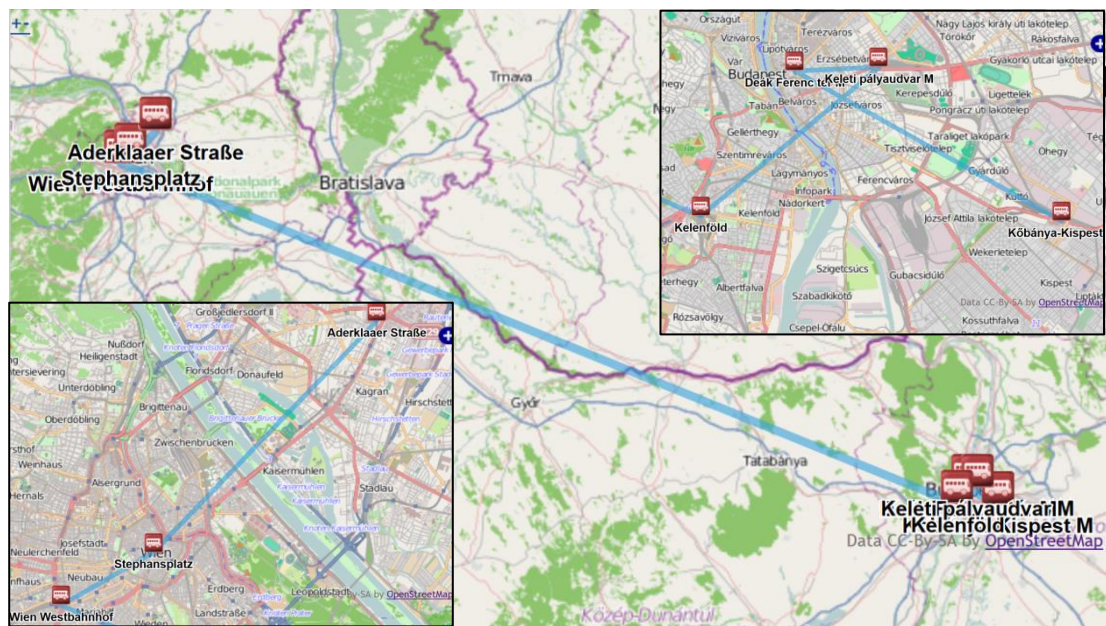
Kiindulópont: Kőbánya Kispest

Célpont: Aderklaaer Straße

A következő példa egy nemzetközi utazás lesz, a Kőbánya Kispestről és a Bécsi U1-es metróvonal Aderklaaer Straße között. Ennél az utazásnál három lehetőség van, az algoritmus az indulási időpontnak megfelelően szolgáltat megoldást. Az első lehetőség a vonatos utazás (14. ábra):



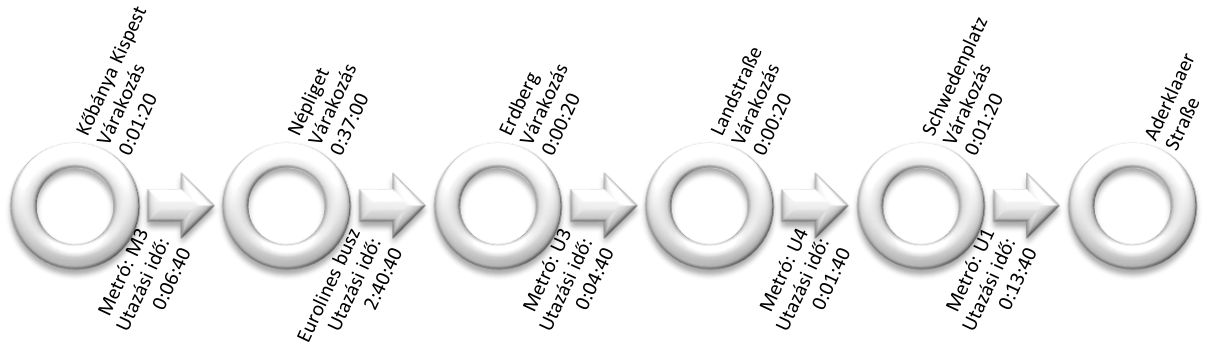
14. ábra A kiinduló rendszer futási eredményének útvonala (3a. példa)



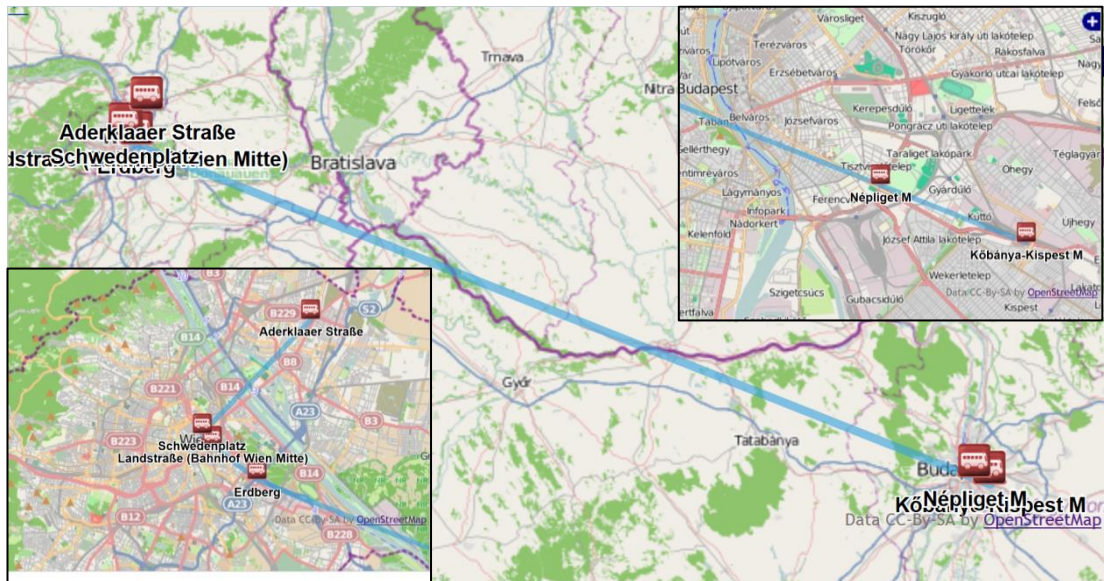
15. ábra A kiinduló rendszer futási eredményének térképe példa 3a

Az eredmények értékelése közben egy érdekességre hívnám fel a figyelmet, a Keleti pályaudvar és a Kelenföldi pályaudvar között a Railjet helyett egy személyvonattal való utazást ajánl fel az algoritmus. Ennek valószínűleg az az oka, hogy a „trip-ek” között az ABC sorrendben hamarabb van a 9-es, mint az R, így kerülhetett a személyvonat az útvonalajánlásba.

A második megoldás a buszos út. Itt a Volánbusz népligeti autóbusz-pályaudvaráról induló Eurolines busszal adott az eljutás lehetősége (16. ábra).

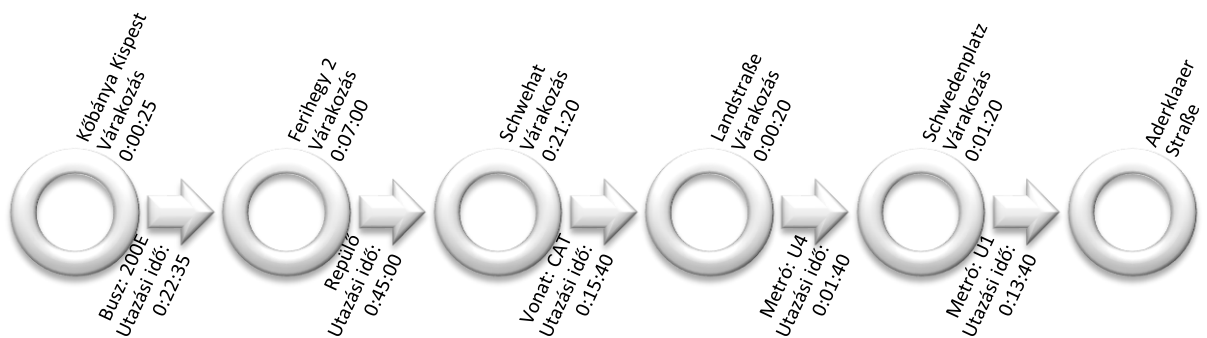


16. ábra A kiinduló rendszer futási eredményének útvonala (3b. példa)

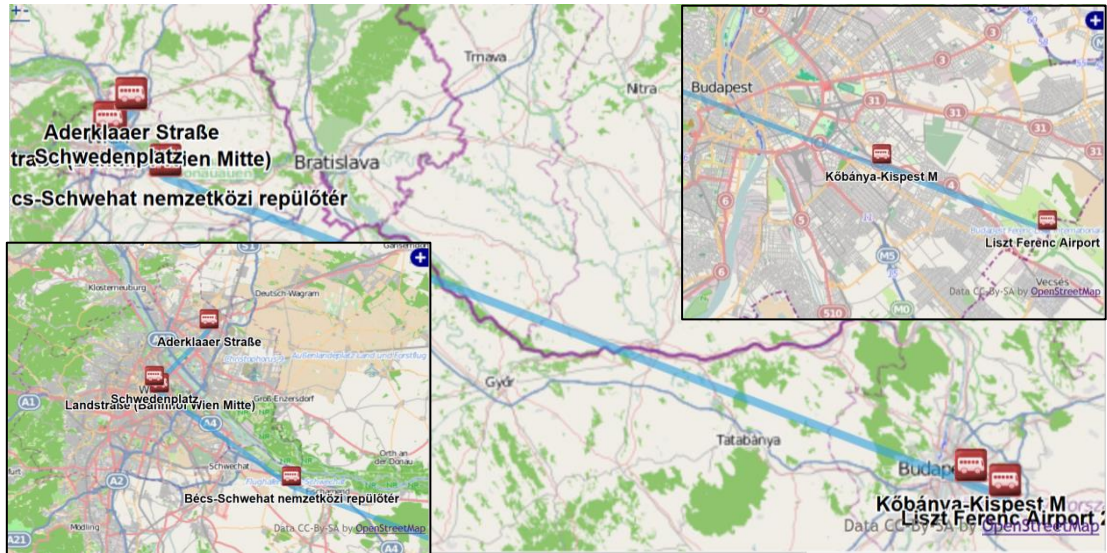


17. ábra A kiinduló rendszer futási eredményének térképe (3b. példa)

A harmadik lehetőség pedig a repülővel való eljutás (18. ábra).



18. ábra A kiinduló rendszer futási eredményének útvonala (3c. példa)



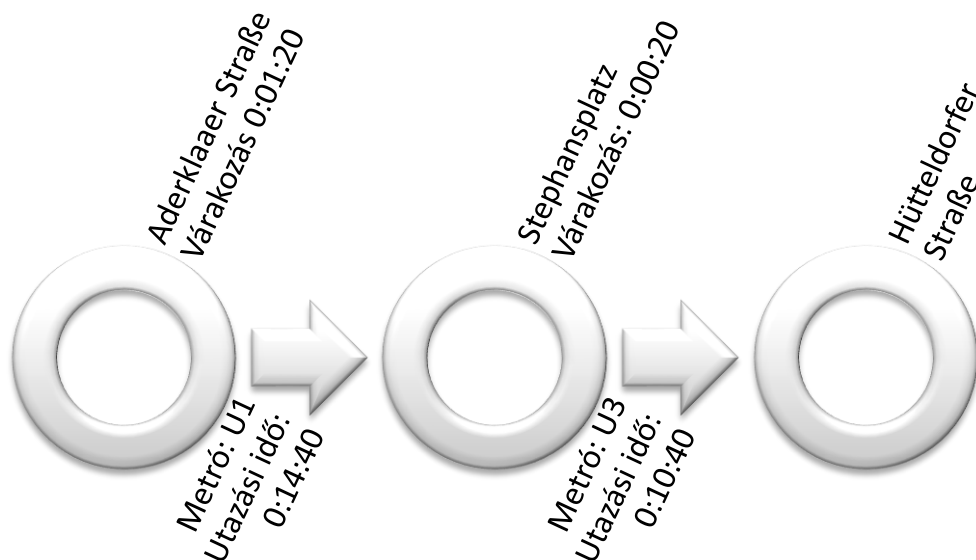
19. ábra A kiinduló rendszer futási eredményének térképe (3c. példa)

5.1.4 A Bécsi hálózat példája

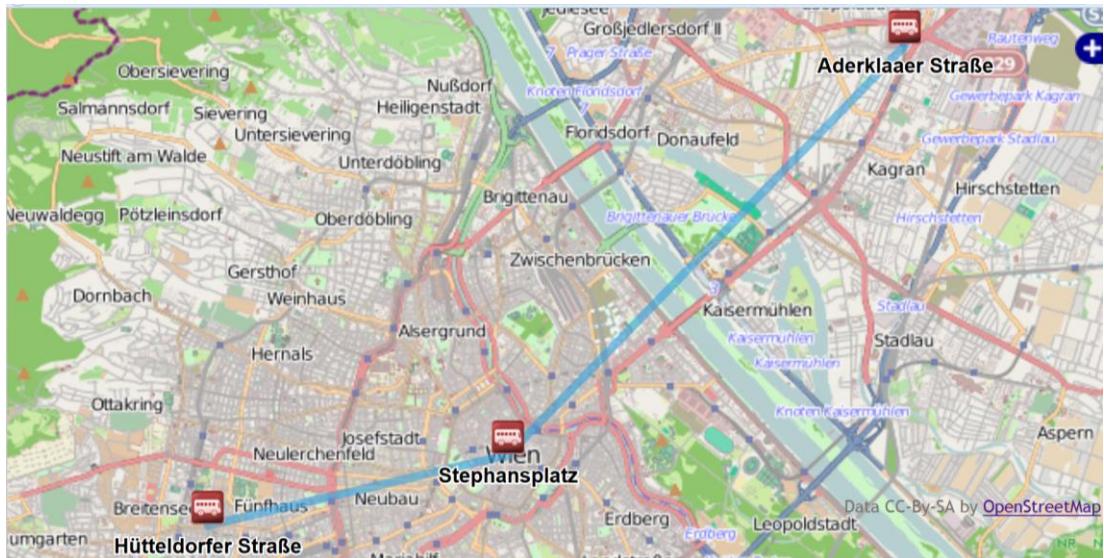
Kiindulópont: Aderklaaer Straße

Célpont: Hütteldorfer Straße

A következő utazás egy Bécsi példa lesz, az U1-es metróvonal Aderklaaer Straße megállója és az U3-as metró Hütteldorfer Straße megállója között.



20. ábra A kiinduló rendszer futási eredményének útvonala (4. példa)



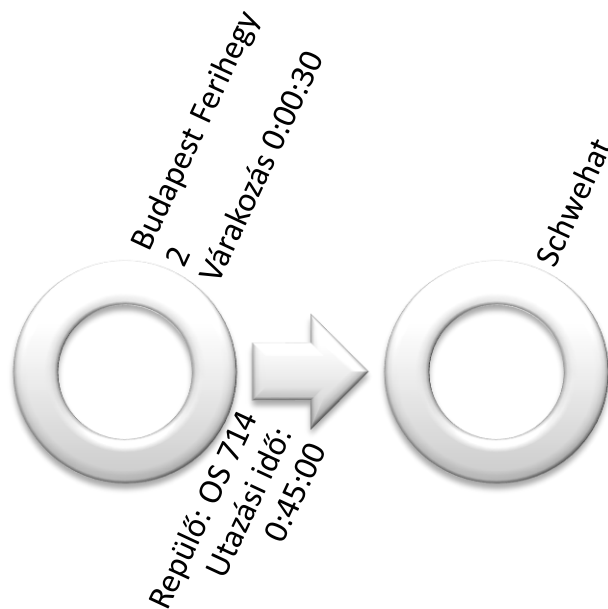
21. ábra A kiinduló rendszer futási eredményének térképe (4. példa)

5.1.5 Légi utazás példája

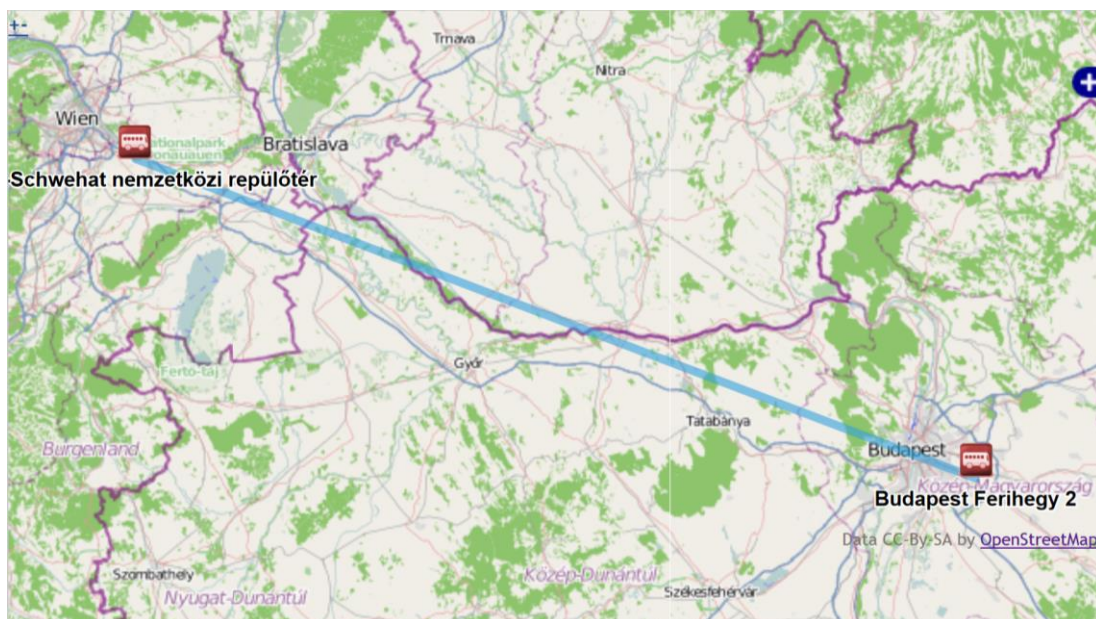
Kiindulópont: Budapest Liszt Ferenc Nemzetközi repülőtér (Ferihegy)

Célpont: Wien-Schwechat nemzetközi repülőtér

Az utolsó példa pedig a Budapesti és Bécsi repterek közötti eljutást mutatja be.



22. ábra A kiinduló rendszer futási eredményének útvonala (5. példa)



23. ábra A kiinduló rendszer futási eredményének térképe (5. példa)

6 A jelenlegi rendszerállapot bemutatása

Az eddigi eredményeket alapul véve a jelenlegi fázisban a szabad szavas címkere-
sést illesztettem a rendszerbe. Ennek segítségével két tetszőlegesen megadott cím közöt-
ti eljutás adható meg. Emellett pedig kilistázásra kerültek az átszálló pontok közötti
köztes megállók is, melynek segítségével pontosabban rajzolhatók fel az útvonalak.

A feladatot két részletre kellett bontani. Elsősorban a megadott címekhez kellett
geokód-ot rendelni, ezzel azonosítva a helyszínt. Másodsorban pedig ki kellett listázni a
köztes megállókat.

6.1 Geokód rendelése a címekhez [5]

Ehhez egy nyílt forrású oldalt használtam. A működéséhez a kiindulási vagy érke-
zési címként megadandó címet adott struktúrába kell konvertálni. Legyen a keresett
cím: 135, Pilkington Avenue, Castle Vale, Birmingham, West Midlands, Anglia. Ebben
az esetben a struktúra a következő:

Kötelező előtag: „*http://nominatim.openstreetmap.org/search?q=*”

Cím: „*135+pilkington+avenue,+birmingham*”

Kötelező utótag: „*&format=xml&polygon=1&addressdetails=1*”

Összefűzve:

```
http://nominatim.openstreetmap.org/search?q=135+pil  
kington+avenue,+birmingham&format=xml&polygon=1&add  
ressdetails=1
```

Az oldal lehetőséget ad további kétféle bevitelre, ezek a következőképpen néznek
ki ugyanennél a példánál maradva:

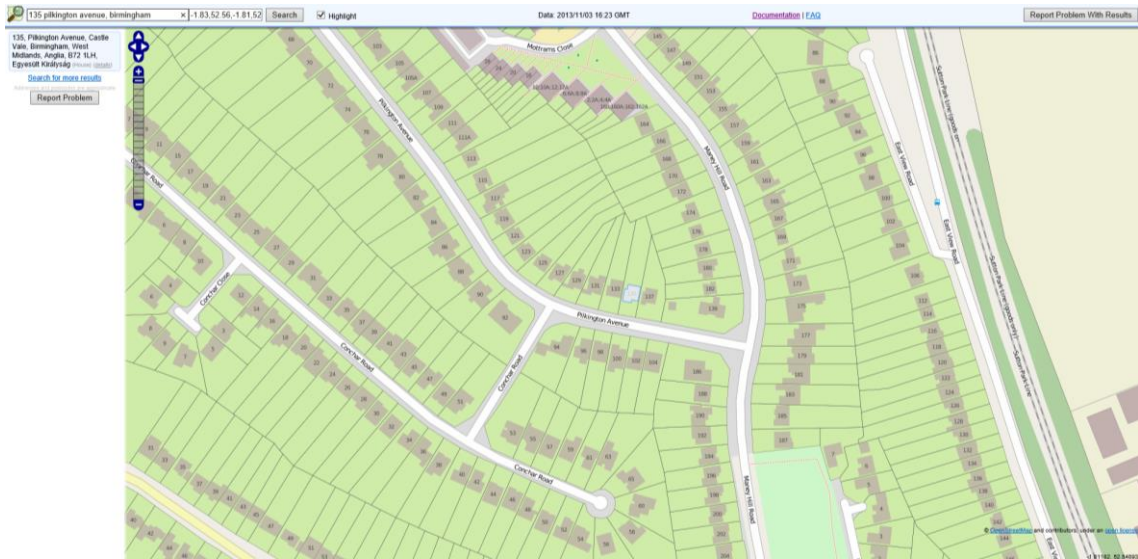
```
http://nominatim.openstreetmap.org/search/135%20pil  
kington%20avenue,%20birmingham?format=xml&polygon=1  
&addressdetails=1
```

```
http://nominatim.openstreetmap.org/search/gb/birmin  
gham/pilkington%20avenue/135?format=xml&polygon=1&a  
ddressdetails=1
```

Az eredmények tekintetében 3 lehetőség közül lehet választani, melyek:

- html

Ebben az esetben a következő kép tárul elénk (24. ábra):



24. ábra Geokód keresés eredménye html esetben

Látható, hogy ebben az esetben a lekérdezés egy grafikai megjelenítést ad válaszképpen.

- xml

Xml válasz esetén a következő kódot szolgáltatja eredményül:

```
<?xml version="1.0" encoding="UTF-8"?>
<searchresults mo-
re_url="http://nominatim.openstreetmap.org/search?f
ormat=xml&exclude_place_ids=62311100&accept-
language=hu-
HU&polygon=1&addressdetails=1&q=135+pilkington+aven
ue%2C+birmingham" exclude_place_ids="62311100"
polygon="true" querystring="135 pilkington avenue,
birmingham" attribution="Data © OpenStreetMap
contributors, ODbL 1.0.
http://www.openstreetmap.org/copyright"
timestamp="Sun, 03 Nov 13 16:25:07 +0000"><place
importance="0.701" type="house" class="place" disp-
lay_name="135, Pilkington Avenue, Castle Vale, Bir-
mingham, West Midlands, Anglia, B72 1LH, Egyesült
Királyság" lon="-1.81642903005944" lat="52.5487977"
```

```

polygonpoints="["-1.816513","52.5487566"],["-
1.8164913","52.548824"],["-
1.8164685","52.5488213"],["-
1.8164599","52.5488481"],["-
1.8163464","52.5488346"],["-
1.8163717","52.5487561"],["-
1.816429","52.5487629"],["-
1.816434","52.5487473"],["-
1.816513","52.5487566"]]"
boundingbox="52.5487442016602,52.5488510131836,-
1.81651306152344,-1.81634628772736" place_rank="30"
osm_id="90394480" osm_type="way"
place_id="62311100"><house_number>135</house_number
><road>Pilkington Avenue</road><suburb>Castle
Vale</suburb><city>Birmingham</city><county>West
Midlands</county><state_district>West
Midlands</state_district><state>Anglia</state><post
code>B72 1LH</postcode><country>Egyesült Király-
ság</country><country_code>gb</country_code></place
></searchresults>

```

– json

Ebben az esetben pedig egy szöveges fájlt készít az oldal, melynek a következő a tartalma:

```

[{"place_id":"62311100","licence":"Data \u00a9
OpenStreetMap contributors, ODbL 1.0.
http://www.openstreetmap.org/copyright","osm_typ
e":"way","osm_id":"90394480","boundingbox":["52.548
7442016602","52.5488510131836","-
1.81651306152344","-
1.81634628772736"],"polygonpoints":["-
1.816513","52.5487566"],["-
1.8164913","52.548824"],["-
1.8164685","52.5488213"],["-

```

```

1.8164599", "52.5488481"], ["-
1.8163464", "52.5488346"], ["-
1.8163717", "52.5487561"], ["-
1.816429", "52.5487629"], ["-
1.816434", "52.5487473"], ["-
1.816513", "52.5487566"]], "lat": "52.5487977", "lon": "-
-1.81642903005944", "display_name": "135, Pilkington
Avenue, Castle Vale, Birmingham, West Midlands,
England, B72 1LH, United
Kingdom", "class": "place", "type": "house", "importance
": 0.801, "address": {"house_number": "135", "road": "Pil
kington Avenue", "suburb": "Castle
Vale", "city": "Birmingham", "county": "West
Midlands", "state_district": "West
Midlands", "state": "England", "postcode": "B72
1LH", "country": "United
Kingdom", "country_code": "gb"}}]

```

A geokód kereséshez az xml formátumot választottam, mert php környezetben ez a formátum a legegyszerűbben kezelhető.

A keresést mind a kezdő mind az érkezési cím esetén le kell futtatni, ekkor rendelkezésre állnak a kezdő és végpontok geokód-jai. Ezt követően meg kell vizsgálni, hogy ehhez a geokód-hoz melyik megálló esik a legközelebb. Ezt a megállók valamint a kezdő- és végpontok geokód-jainak az egymáshoz képesti távolságainak a minimalizálásával lehet kiválasztani. Erre pedig már lefuttatható az útvonalkeresés.

6.2 Köztes megállók kilistázása

Ezt a lépést több allépésre bontottam a kezelhetőség miatt. Ezt figyelembe véve elkészítettem egy SQL lekérdezést, mely adott „trip” azonosítóhoz megadott kezdő és végpontok között kilistázza a megállókat. A lekérdezés a következőképpen néz ki, ha a keresett „trip” azonosító „TR0043100”, a kiinduló- „M0022” a végső megálló pedig „M0052”:

```

Select stop_id
From stop_times
where trip_id = 'TR0043100'
and (stop_sequence between
    (Select stop_sequence
    From stop_times
    where trip_id = 'TR0043100'
    and stop_id = 'M0022')
and
    (Select stop_sequence
    From stop_times
    where trip_id = 'TR0043100'
    and stop_id = 'M0052'))
ORDER BY stop_sequence ASC;

```

A lekérdezés eredménye pedig a következő lista (2. táblázat):

Stop_id
"M0022"
"M0044"
"M0045"
"M0046"
"M0047"
"M0048"
"M0013"
"M0036"
"M0051"
"M0052"

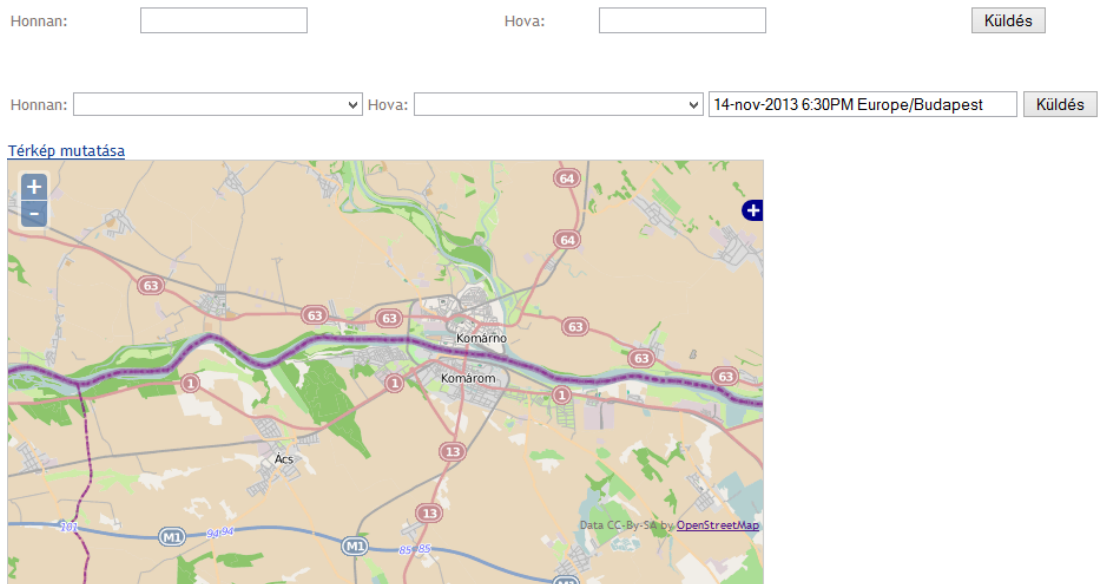
2. táblázat Köztes megállók listája

A listában szerepel minden köztes megálló a paramétereknek megfelelően. Ezt a lekérdezést kvázi függvényként alkalmazva bármennyiszer meg lehet hívni, így minden

egy-átszállásnál felbontva az utazást, viszonylatonként összerakható a megállólista. Ezt a listát felhasználva pedig kirajzolható az útvonal.

6.3 Eredmények bemutatása

Jelen verzióban a korábbi felület a címek beírását lehetővé tevő mezőkkel bővült. Továbbra is a tesztelés volt az elsődleges szempont, így a felhasználói felület is csak a szükséges mértékben bővült. Ezt a felületet mutatja a 25. ábra



25. ábra A jelenlegi felhasználói felület

Útvonal ajánlathoz első lépésben meg kell adni a kiindulási és érkezési címet, melyet a következőképpen kell bevinni: *Házszám, Utca neve, Közterület fajtája, Város neve, Irányítószám.* (26. ábra)

Honnan: Hova:

Honnan: Hova:

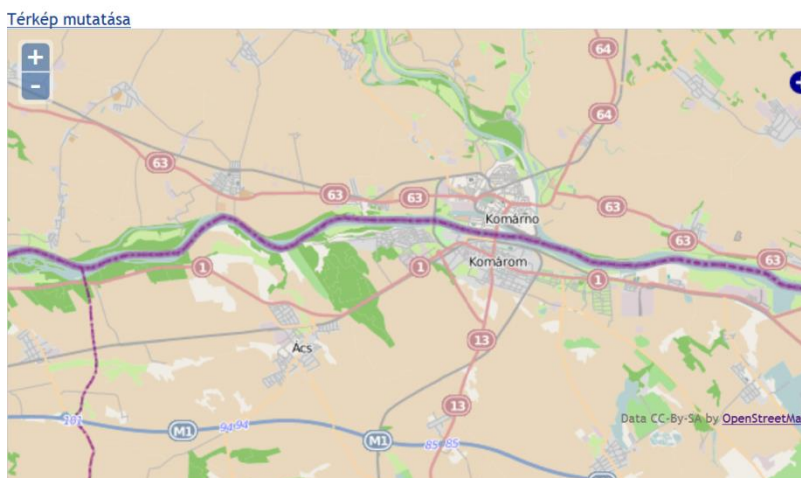


26. ábra Címek megadása

A küldés gombot megnyomva megjelennek a legközelebbi megállók (27. ábra)

Honnan: Hova:

Honnan: Hova:



27. ábra Legközelebbi megállók megjelenítése

Az időpont megadását követően pedig a következő eredményt kaptam (28. ábra)

Honnan: Hova:

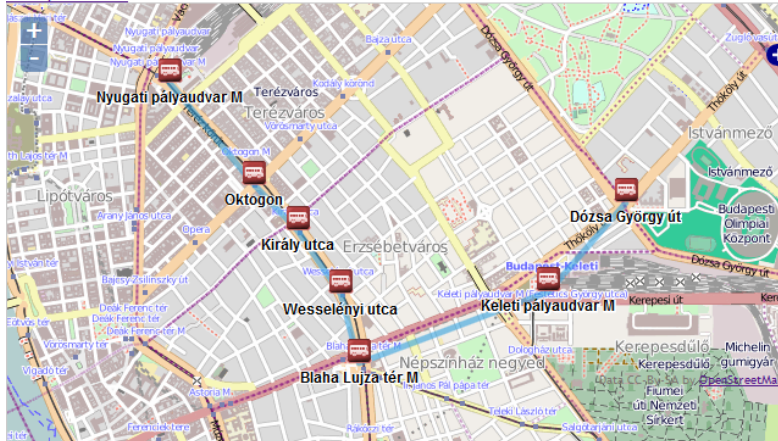
Honnan: Hova:

Útvonal

	Stop id	Megálló neve	Trip id	Viszonylat	Várakozási idő	Utazási idő
Átszállás	M0072	Dózsa György út	TR001731640	173	00:04:25	00:02:35
	M0072	Dózsa György út	TR001731640	173		
	M0028	Keleti pályaudvar M	TR001731640	173		
Átszállás	M0028	Keleti pályaudvar M	TR00M24201	M2	00:01:20	00:00:40
	M0028	Keleti pályaudvar M	TR00M24201	M2		
	M0027	Blaha Lujza tér M	TR00M24201	M2		
Átszállás	M0027	Blaha Lujza tér M	TR0062721	6	00:01:15	00:01:45
	M0027	Blaha Lujza tér M	TR0062721	6		
	M0052	Wesselényi utca	TR0062721	6		
	M0051	Király utca	TR0062721	6		
Átszállás	M0051	Király utca	TR0042721	4	00:00:15	00:02:45
	M0051	Király utca	TR0042721	4		
	M0036	Oktogon	TR0042721	4		
	M0013	Nyugati pályaudvar M	TR0042721	4		
Átszállás	M0013	Nyugati pályaudvar M				

Összes utazási idő: 00:15:00

Térkép mutatása



28. ábra Keresési eredmények megjelenítése

Látható, hogy a keresési eredmények megfelelően illeszkednek a hálózatra, az összes köztes megálló szerepel a térképen és a felsoroláson is. Ez alapján ismételt vizsgálat alá vonhatóak a kiinduló rendszer példái. Ebben az esetben el lehet tekinteni az egyszerűbb példáktól, mert az összetett példa több egyszerű esetből áll, így csak az 5.1.3-es fejezetben lévő eset mélyebb vizsgálatára szorítkozom.

Legyenek a következő címek a keresendő helyek:

Kiindulási cím: 1191 Bp. Vak Bottyán utca 77.

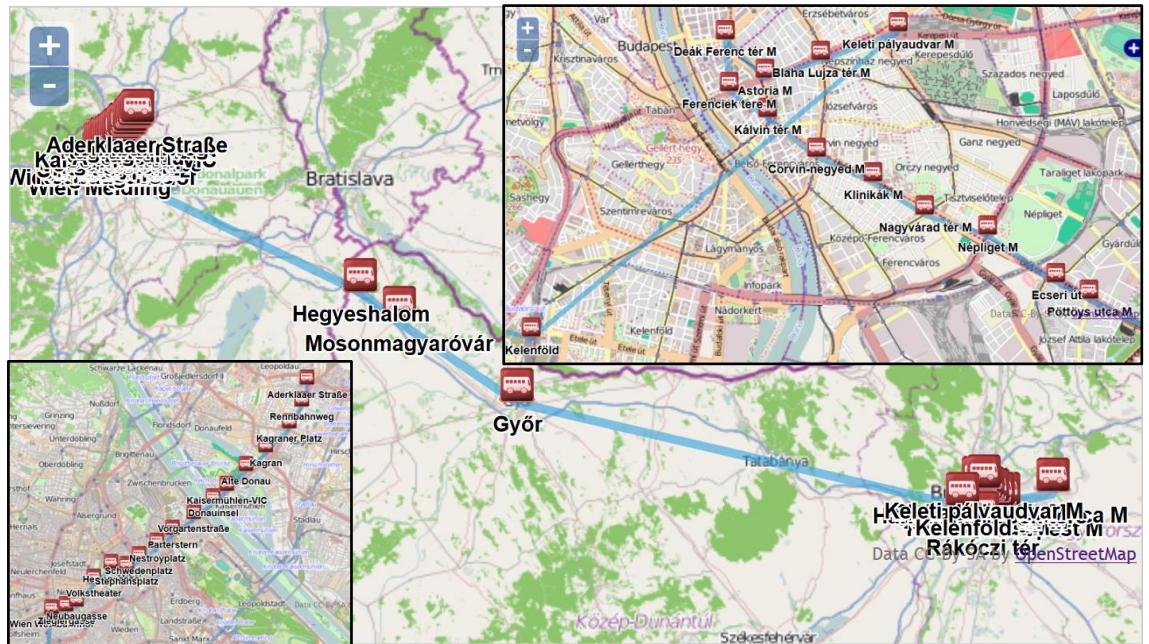
Érkezési cím: Kürschnergasse 4, 1210, Wien

Az útvonal az első verziónál (vasúton való utazás) a következő (3. táblázat).

	Megálló neve	Viszonylat	Várakozási idő	Utazási idő
Kiinduló állomás	Kőbánya-Kispest M	M3	0:01:20	0:15:40
	Határ út M	M3		
	Pöttöys utca M	M3		
	Ecséri út	M3		
	Népliget M	M3		
	Nagyvárad tér M	M3		
	Klinikák M	M3		
	Corvin-negyed M	M3		
	Kálvin tér M	M3		
	Ferenciek tere M	M3		
Átszállás	Deák Ferenc tér M	M2	0:00:20	0:04:40
	Astoria M	M2		
	Blaha Lujza tér M	M2		
Átszállás	Keleti pályaudvar M	912/942	0:18:40	0:12:20
Átszállás	Kelenföld	Railjet	0:55:40	2:53:20
	Győr	Railjet		
	Mosonmagyaróvár	Railjet		
	Hegyeshalom	Railjet		
	Wien Meidling	Railjet		
Átszállás	Wien Westbahnhof	U3	0:02:20	0:06:40
	Zieglergasse	U3		
	Neubaugasse	U3		
	Volkstheater	U3		
	Herrengasse	U3		
Átszállás	Stephansplatz	U1	0:01:20	0:14:40
	Schwedenplatz	U1		
	Nestroyplatz	U1		
	Parterstern	U1		
	Vorgartenstraße	U1		
	Donauinsel	U1		
	Kaisermühlen-VIC	U1		
	Alte Donau	U1		
	Kagran	U1		
	Kagraner Platz	U1		
	Rennbahnweg	U1		
Végállomás	Aderklaaer Straße			

3. táblázat Útvonal eredmények a. verzió

A térképes megjelenítés pedig a következőképpen néz ki (29. ábra).



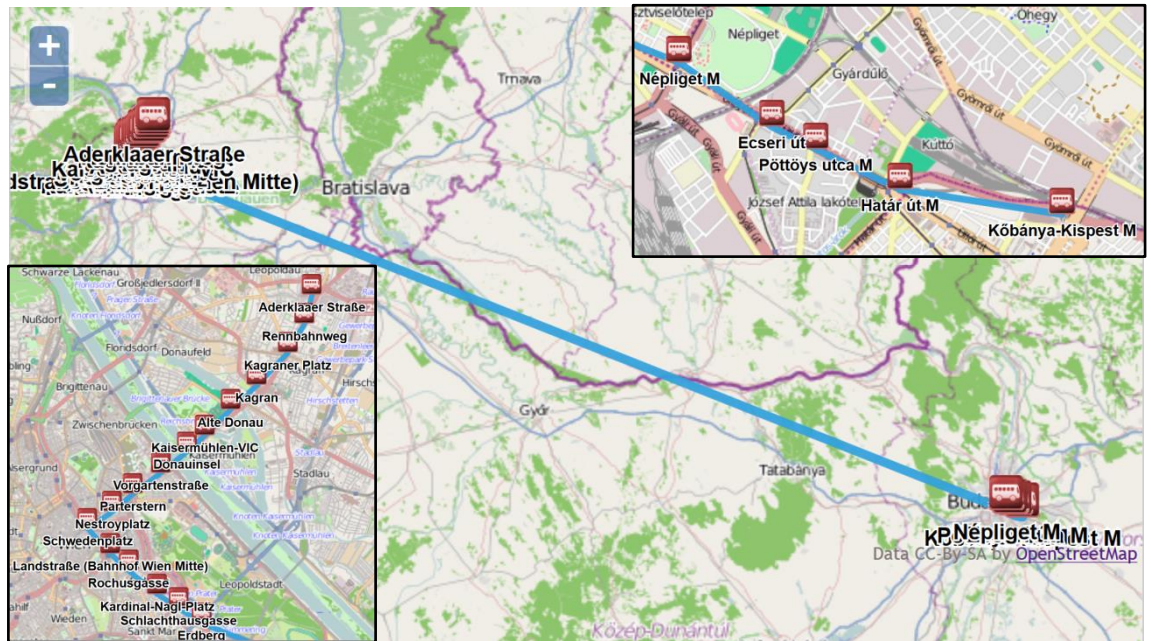
29. ábra Útvonal grafikus megjelenítése a. verzió

A második esetben, amikor a buszos utazást ajánlja Budapest és Bécs között a rendszer, akkor a következőképpen néz ki az útvonal (4. táblázat)

	Megálló neve	Viszonylat	Várakozási idő	Utazási idő
Kiinduló állomás	Kőbánya-Kispest M	M3	0:01:20	0:06:40
	Határ út M	M3		
	Pöttöys utca M	M3		
	Ecseri út	M3		
Átszállás	Népliget M	U6	0:27:00	2:40:00
Átszállás	Erdberg	U3	0:00:20	0:04:40
	Schlachthausgasse	U3		
	Kardinal-Nagl-Platz	U3		
	Rochusgasse	U3		
Átszállás	Landstraße (Bahnhof Wien Mitte)	U4	0:00:20	0:01:40
Átszállás	Schwedenplatz	U1	0:01:20	0:13:40
	Nestroyplatz	U1		
	Parterstern	U1		
	Vorgartenstraße	U1		
	Donauinsel	U1		
	Kaisermühlen-VIC	U1		
	Alte Donau	U1		
	Kagran	U1		
	Kagraner Platz	U1		
	Rennbahnweg	U1		
Végállomás	Aderklaaer Straße			

4. táblázat Útvonal eredmények b. verzió

A térképes megjelenítés pedig a következőképpen néz ki (30. ábra).



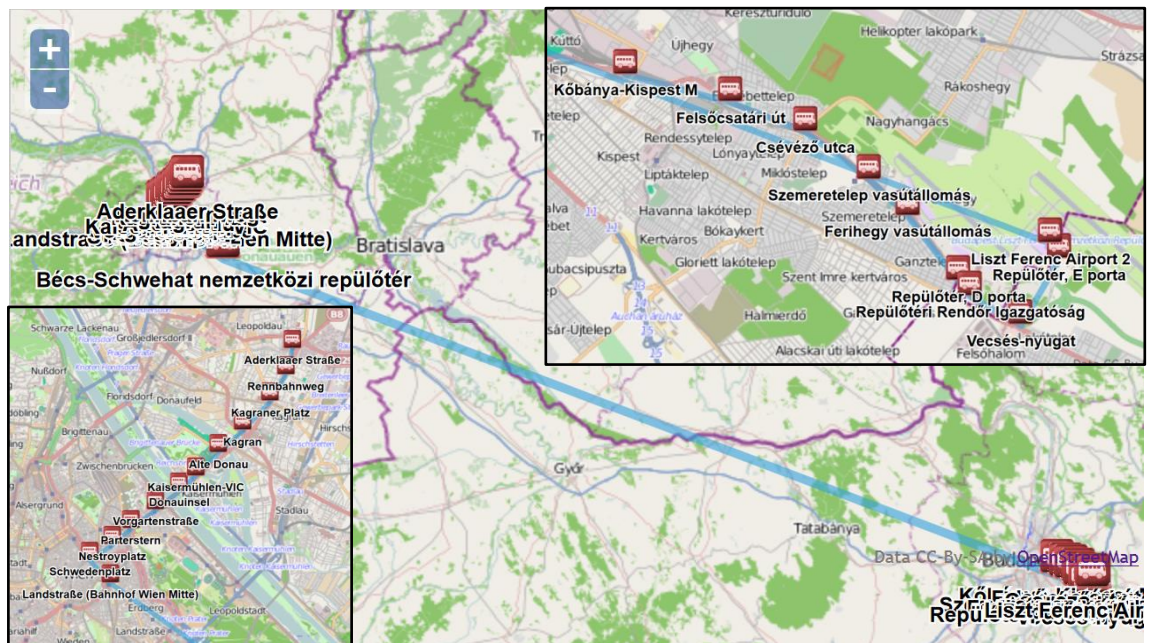
30. ábra Útvonal grafikus megjelenítése b. verzió

Végül pedig a repülővel történő utazási ajánlat a következő (5. táblázat):

	Megálló neve	Viszonylat	Várakozási idő	Utazási idő
Kiinduló állomás	Kőbánya-Kispest M	200E	0:00:25	0:22:35
	Felsőcsatári út	200E		
	Csévész utca	200E		
	Szemeretelep vasútállomás	200E		
	Ferihegy vasútállomás	200E		
	Repülőtér, D porta	200E		
	Repülőtéri Rendőr Igazgatóság	200E		
	Vecsés-nyugat	200E		
	Repülőtér, E porta	200E		
Átszállás	Liszt Ferenc Airport 2	OS_714	0:07:00	0:45:00
Átszállás	Bécs-Schwechat nemzetközi repülőtér	CAT	0:21:20	0:15:40
Átszállás	Landstraße (Bahnhof Wien Mitte)	U4	0:00:20	0:01:40
Átszállás	Schwedenplatz	U1	0:01:20	0:13:40
	Nestroyplatz	U1		
	Parterstern	U1		
	Vorgartenstraße	U1		
	Donauinsel	U1		
	Kaisermühlen-VIC	U1		
	Alte Donau	U1		
	Kagran	U1		
	Kagraner Platz	U1		
	Rennbahnweg	U1		
Végállomás	Aderklaaer Straße			

5. táblázat Útvonal eredmények c. verzió

A térképes megjelenítés pedig a következőképpen néz ki (31. ábra).



31. ábra Útvonal grafikus megjelenítése c. verzió

Látható, hogy a bővítmény megfelelően működik, és továbbra is a különböző indulási időpontoktól függ az ajánlott útvonal. Emellett az összes köztes állomást is képes a rendszer visszaadni, így kirajzolhatóak a részletesebb útvonalak. Ez kis távolságnál kielégítő közelítést jelent a valós útvonalnak, azonban nagy távolságnál továbbra is messze esik a ténylegestől.

6.4 Továbbfejlesztési lehetőségek

Értékelve az eredményeket elmondható, hogy sikerült beépíteni a szabad szavas címkeresést, így a legközelebbi megállót megtalálva képes az algoritmus eredményt szolgáltatni. Emellett a köztes megállókat is ki tudja listázni illetve megjeleníteni a térképen a teljes tömegközlekedési útvonalat. Azonban arra fel kell hívnom a figyelmet, hogy jelenleg még nem képes a terepakadályokat kezelni, azaz valamilyen fizikai akadály miatt az átjutás lehetetlen, vagy időigényes, illetve adott indulási időponttól függően nem feltétlenül a legközelebbi megálló adhatja a valamilyen szempontból legjobb eljutási lehetőséget. Továbbra is felmerül, hogy kezelni kell az átszállások kérdéskörét. Itt több megoldandó probléma merülhet fel, egyrészt a peronok illetve megállóhelyek fizikai elhelyezkedése, és a szükséges gyaloglási idő. Itt is megtörténhet, hogy fizikailag közel vannak egymáshoz, de valamilyen fizikai akadály miatt mégis időigényes az átju-

tás. Emellett előfordulhatnak eltérések, késések a menetrendben, melyeket kezelnie kell az algoritmusnak. Például egy hazai vasúti utazásnál a Balaton környékén, ahol rendszeresek a késések, ott sokkal több átszállási időt kell hagyni, mint egy németországi utazásnál. Azonban a túlbiztosítás sem célszerű, mert az utasok nem szeretnek átszállni és főleg várakozni, így egy megfelelő optimumot kell találni. A gyaloglási sebesség is egy olyan kérdéskör, melyet kezelni kell, mert nem mindegy ebben az esetben, hogy egy idősebb, vagy valamilyen oknál fogva lassabban vagy nehezebben mozgó utasról van szó, vagy egy egészséges fürge emberről. Itt még a hátrányos helyzetű utasok is egy érdekes kérdéskör lehet, mert nem biztos, hogy minden helyszínen biztosítva van számukra az átszállási lehetőség, erre példa a budapesti metró, ahol kerekesszékekkel igen nehézkesen oldható meg az átszállás.

A következő fejlesztés szint lehet a közúthálózattal való összekapcsolás. Itt nem egyszerűen a rá- és elgyaloglást kell érteni, hanem azt, hogy sok esetben autóval, taxival, vagy biciklivel mennek az emberek a legközelebbi busz-, vasútállomáshoz vagy egy reptérre. Ilyenkor figyelembe kell venni, hogy milyen parkolási lehetőségek vannak a környéken, van-e hosszútávú, vagy őrzött parkoló, kerékpár tároló, vagy lehet-e a kerékpárral utazni az adott járművön.

A hálózat bővítésével pedig a hatványozottan növekvő adattömeg kezelésére kell megoldást találni, mert egy több országot lefedő keresésnél, a szolgáltatók sokfélesége illetve az eljutási lehetőségek nagy száma miatt igen nagy mátrixok keletkezhetnek, melyekben a keresés igen időigényes. Ezen kérdéskör megoldása létfontosságú egy hatékony algoritmus kialakítására.

Újabb szintlépés lehet nem csak az útvonal megadása, hanem az utazás költségének kiszámítása, mellyel összehasonlíthatóvá válnak a közösségi közlekedés eljutási lehetőségei és az egyéni közlekedés lehetőségei. Itt a legfontosabb megoldandó probléma az adatok beszerzése mellett, hogy rengeteg árképzési megoldás létezik. Az árképzési megoldások között két szélső helyzet létezik, az egyik véglét az 1 utazásra jogosító jegy, ahol nem függ sem a távolság, sem az idő, a másik véglét pedig hogy a távolsággal, vagy az idővel vagy mindkettővel arányos a díjszabás. Emellett léteznek a bérletek, amelyek egy teljesen külön kategóriát alkotnak. Ezeknél kezelni kell a gyűjtő jegy típusúakat és a havi vagy éves bérleteket is. Továbbá a különböző kedvezményeket, akciókat is figyelembe kell venni. Ebben a kérdéskörben is hatalmas sokszínűséggel lehet találkozni, és nem csak a különböző szolgáltatók alkalmaznak teljesen eltérő kedvezményeket, hanem a különböző országok is, melyek teljesen eltérő feltételekkel biztosít-

ják azokat. Ebben a kérdéskörben még a közúti költségeket is meg kell jeleníteni, mert nem hagyható figyelmen kívül a megfelelő összehasonlításhoz, hogy az utazásnak nem csak benzin, hanem egyéb költségei is vannak, mint például az autópálya díjak, komp díjak, behajtási díjak, stb., melyek ritkán jelennek meg a különböző algoritmusokban, programokban.

Végezetül pedig a foglalások megoldása az a kérdéskör, melyet egy ilyen rendszernek tartalmaznia kell, ahhoz, hogy a betölteni kívánt célt teljesítse. Itt az online adatok mellett a jegyek eljuttatásának a kérdésköre az igazán izgalmas, mivel sok szolgáltató még nem rendelkezik elektronikus jeggyel, így azt valamilyen formában el kell juttatni a vevőhöz. Természetesen az online rendszerekhez való kapcsolódás és jegyfoglalásnál felmerülő probléma, hogy azonos, vagy adott időben az azonos helyre történő foglalás elkerülhető legyen.

Az eddigiek mellett még további támogató, illetve utazási élményfokozó bővítés kapcsolódhat egy ilyen rendszerhez, mint például az utas folyamatos végigkísérése az autós GPS-hez hasonlóan, mely tájékoztatja, hogy éppen hol van, mennyi a várható odaérési idő, mikor kell jeleznie a leszállást, hogyan kell eljutnia az átszállásnál a másik peronra vagy megállóba, mely lépcsőt használja, stb.

Áttekintve a lehetőségeket, a következő lista készíthető a fejlesztési lehetőségekről:

- átszállási lehetőségek pontosabb kezelése (akadályok, idő, hátrányos helyzetűek);
- az egyéni és a közösségi közlekedés kombinációja;
- nagytávolságú utazások kezelése (nagy adatmennyiség, Big Data);
- utazási költségek számítása;
- helyfoglalás a járatokra.

7 Összefoglalás

Összefoglalva az eddigieket, a korábbi munkáim során felépítettem egy olyan környezetet, mely a valós hálózatok működése alapján létrehoz egy virtuális hálózatot, ezen kereséseket végez, és útvonal ajánlatokat ad. A Google GTFS szabványa alapján elkészítettem egy algoritmust, mely a kézzel bevitt adatok alapján legenerálja az SQL táblák feltöltéséhez szükséges parancsokat, így ezen leegyszerűsített hálózathoz tartozó

több százezer adat felvitele is egyszerűen megtörténhet. A feltöltés eredményeképpen létrejött táblázaton pedig ezek után kereséseket hajtottam végre. A hálózat úgy lett kialakítva, hogy elegendően sok kapcsolat legyen ahhoz, hogy az algoritmusok működését speciális esetekben is vizsgálni lehessen.

Ezen eredményekre támaszkodva a jelen verzióban további funkciókat kapcsoltam a rendszerhez. Legfontosabb bővítmény, hogy most már tetszőleges címmel lehet keresni. Itt a lényeges, hogy rendelkezésre kell állnia egy geokód adatbázisnak, amely összekapcsolja a címeket a geokód-okkal. Fontos elvárás az adatokkal kapcsolatban, hogy a lefedni kívánt területen tartalmazza az összes címet.

További fejlesztés volt, hogy a köztes megállók is kilistázásra kerültek, majd felhasználásra is kerültek a térképi megjelenítésben. Erre az útvonal kirajzolásának finomítása miatt volt szükség, így sokkal közelebb került az ábrázolás a valós közlekedési hálózathoz.

Végezetül pedig bemutattam azokat az irányokat, melyek a továbbfejlesztési követelményeket meghatározzák. A fejlesztések során csak szakaszosan érdemes előrehaladni, így biztosítható, hogy egy újabb szolgáltatási szint elérése nem rontja le az addigi eredményeket, így csak a lépésről lépésre való építkezés vezethet jó eredményre.

Azonban az látható az eredmények alapján, hogy a jelenlegi algoritmusoknak komoly korlátai vannak, melyeket a továbbiakban a fejlesztésükkel, tökéletesítésükkel ki kell küszöbölni, hogy az átszállások megfelelően legyenek kezelhetők, illetve az utazási költségek tekintetében is megfelelő eredményeket szolgáltatassanak.

Irodalomjegyzék

- [1] **Az Európai Közösségek Bizottsága:** *Fehér könyv, Európai közlekedéspolitikai 2010-ig: itt az idő dönteni*, 2001. szeptember 12, Brüsszel, COM(2001)370
- [2] **Wikipédia:** *Google Térkép szócikk*,
http://hu.wikipedia.org/wiki/Google_T%C3%A9rk%C3%A9p
- [3] **Google Inc:** *GTFS specification*, 2011. szeptember 6.
http://code.google.com/intl/hu-HU/transit/spec/transit_feed_specification.html
- [4] **HaCon Ingenieurgesellschaft mbH:** *HAFAS Fahrplaninformationssystem*, 2013
http://www.hacon.de/unternehmen/presse/downloads/hafas_broschuere_download_d.pdf
- [5] **Nominatim:** *Nominatim, utoljára módosítva 2013.09.15*
<http://wiki.openstreetmap.org/wiki/Nominatim>

Ábrajegyzék

1. ÁBRA ELVI RENDSZERMODELL	8
1. TÁBLÁZAT A VISZONYLATOK LISTÁJA	10
2. ÁBRA RENDSZERMODELL	11
3. ÁBRA TÁBLASZERKEZET	12
4. ÁBRA AZ ADATBEVITELI ALGORITMUS FOLYAMATÁBRÁJA	14
5. ÁBRA AZ ADATBEVITELNEK MEGFELELŐ KÖZLEKEDÉSI HÁLÓZAT	15
6. ÁBRA A KIINDULÓ FELHASZNÁLÓI FELÜLET	16
7. ÁBRA A KIINDULÓ RENDSZER ADATBEVITELÉNEK 1. PÉLDÁJA	17
8. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK ÚTVONALA (1. PÉLDA)	18
9. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK TÉRKÉPE (1. PÉLDA)	18
10. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK ÚTVONALA (2A. PÉLDA)	19
11. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK TÉRKÉPE (2A. PÉLDA)	19
12. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK ÚTVONALA (2B. PÉLDA)	20
13. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK TÉRKÉPE (2B. PÉLDA)	20
14. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK ÚTVONALA (3A. PÉLDA)	21
15. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK TÉRKÉPE PÉLDA 3A	21
16. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK ÚTVONALA (3B. PÉLDA)	22
17. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK TÉRKÉPE (3B. PÉLDA)	22
18. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK ÚTVONALA (3C. PÉLDA)	22
19. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK TÉRKÉPE (3C. PÉLDA)	23
20. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK ÚTVONALA (4. PÉLDA)	23
21. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK TÉRKÉPE (4. PÉLDA)	24
22. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK ÚTVONALA (5. PÉLDA)	24
23. ÁBRA A KIINDULÓ RENDSZER FUTÁSI EREDMÉNYÉNEK TÉRKÉPE (5. PÉLDA)	25
24. ÁBRA GEOKÓD KERESÉS EREDMÉNYE HTML ESETBEN	27
2. TÁBLÁZAT KÖZTES MEGÁLLÓK LISTÁJA	30
25. ÁBRA A JELENLEGI FELHASZNÁLÓI FELÜLET	31
26. ÁBRA CÍMEK MEGADÁSA	32
27. ÁBRA LEGKÖZELEBBI MEGÁLLÓK MEGJELENÍTÉSE	32
28. ÁBRA KERESÉSI EREDMÉNYEK MEGJELENÍTÉSE	33
3. TÁBLÁZAT ÚTVONAL EREDMÉNYEK A. VERZIÓ	34
29. ÁBRA ÚTVONAL GRAFIKUS MEGJELENÍTÉSE A. VERZIÓ	35
4. TÁBLÁZAT ÚTVONAL EREDMÉNYEK B. VERZIÓ	35
30. ÁBRA ÚTVONAL GRAFIKUS MEGJELENÍTÉSE B. VERZIÓ	36
5. TÁBLÁZAT ÚTVONAL EREDMÉNYEK C. VERZIÓ	36
31. ÁBRA ÚTVONAL GRAFIKUS MEGJELENÍTÉSE C. VERZIÓ	37

Melléklet 1

GTFS szabvány bemutatása

A szabvány kialakulása

A Google cég 2005. február 8-án indította el ingyenes internetese térképszolgáltatását. Ebben a rendszerben térképek és műholdas felvételek segítségével lehet megtekinteni a föld felszínét. A rendszer először csak megjeleníteni tudta a helyszíneket, de hamar megjelent a keresés funkció is, melyet nem sokkal később az útvonalajánlás követte. Magyarország 2006. április 25-én került bele a rendszerbe, és 2007 májusa óta kereshetőek az utcanevek. 2011. júniusában pedig a budapesti közösségi közlekedés adatai a BKK jóvoltából felkerültek a rendszerbe.[2]

Nem sokkal a Google térkép indulása után felmerült, hogy ne csak közúton lehessen megtervezni az eljutást, hanem tömegközlekedéssel is, ezért már 2005 decemberében a Google labor oldalán megjelent ez a lehetőség is, az Oregon állambeli Portland városának tömegközlekedésére alapulva. A szűk két éves tesztüzem végén 2007 októberében építették bele véglegesen a Google térkép szolgáltatásba.[2]

A tervezés és a hosszú próbaüzem alatt egy olyan nyílt forráskódú kvázi szabványként kezelhető adatbázis szerkezetet hoztak létre, melynek a célja, hogy egy tetszőleges tömegközlekedési cég összes viszonylatát és a díjszabását rögzíteni lehessen. Fontos szempont volt, hogy bármilyen típusú jármű felvihető legyen a rendszerbe. A szabvány a General Transit Feed Specification nevet kapta, melyet a GTFS mozaikszóval rövidítenek, jelentése Általános Tömegközlekedési Adatforrások Leírása.[2]

A GTFS részletes leírása [3]

Az adatbázis oszlopai (attribútumai) a kitöltés szempontjából 3 félék lehetnek:

- Kötelező mező: Minden rekordban kötelező kitölteni az ilyen tulajdonságú mezőket. Néhány ilyen mező esetében megengedhető az üres szöveggel va-

ló kitöltés is, melyet a 2 macskaköröm közötti szünettel lehet megadni. Fontos, hogy a 0 nem jelent üres szöveget.

- Nem kötelező mező: Ezen mezők kitöltése nem kötelező, de ha már egy rekordba is adunk neki értéket, akkor az összesben kell. Ebben az esetben is megengedett az üres szöveg, az előzőekben kifejtettek szerint.
- Egyedi adat: Ez a mező olyan értékeket tartalmaz, mely különbözik a többi értéktől az oszlopban. Például ha egy viszonylathoz az 1A azonosítót rendeljük, akkor semelyik másik viszonylat nem használhatja ezt. Azonban ez az azonosító hozzárendelhető egy helyhez, mert az egy másik típusú egyed.

Táblák

Az adatbázisban 12 táblát határoztak meg, melyek a következők:

- agency (jelentése: vállalat): Ez egy kötelezően kitöltendő tábla. Tartalma a különböző tömegközlekedési vállalatok meghatározott információi.
- stops (jelentése: megállók): Ez egy kötelezően kitöltendő tábla. Ebben tárolhatóak a megállók, ahol utas csere történik.
- routes (jelentése: viszonylat): Ez is egy kötelezően kitöltendő tábla. Itt tárolhatóak a különböző információk az adott viszonylatokról. A viszonylat egy olyan csoportja a különböző utaknak, melyek az utas számára szolgáltatásként jeleníthetők meg.
- trips (jelentése utazások): Kötelező tábla. Ez a tábla az összes utazás és azok viszonylatainak a listája. Az utazás az kettő vagy több megálló sorozata, meghatározott időben.
- stop_times (jelentése megállóhelyi idők): Kötelező tábla. Ez a tábla tartalmazza a járművek összes érkezési és indulási időadatát az összes utazásra vonatkozóan.
- calendar (jelentése naptár): Kötelező tábla. Ebben a táblában vannak meghatározva a különböző viszonylatazonosítókhoz tartozó közlekedési rendek. Itt vannak leírva az üzemidő kezdetének és végének az időpontjai, illetve hogy a hét melyik napján üzemel.

- calendar_dates (jelentése naptári dátumok): Nem kötelező tábla. Itt lehet megadni a calendar táblában rögzítettekhez tartozó kivételeket. Abban az esetben, ha ez a tábla tartalmaz minden viszonylatot a dátumokkal együtt, akkor használható a calendar tábla helyett.
- fare_attributes (jelentése: díjszabási tulajdonságok): Nem kötelező tábla. Itt lehet megadni a díjszabási információkat a közlekedési vállalat különböző viszonylataihoz.
- fare_rules (jelentése: díjszabási szabályok): Nem kötelező tábla. Itt adhatóak meg a közlekedési vállalat viszonylataihoz tartozó díjazási szabályok.
- shapes (jelentése: alakzatok): Nem kötelező tábla. Ebben a táblában határozhatók meg, hogy hogyan jelenjenek meg a viszonylatok.
- frequencies (jelentése: gyakoriságok): Nem kötelező tábla. Itt adhatóak meg a követési időközök a viszonylatokhoz.
- transfers (jelentése: átszállások): Nem kötelező tábla. Ebben a táblában határozhatók meg az átszálláshoz szükséges szabályok.

GTFS szabvány tábláinak meződefiníciója

1.) *Agency (Ügynökség) tábla*

Mező neve	Részletek
agency_id	Nem kötelező. Az agency_id mező az egyedi azonosítója a közlekedési vállalatnak. Ez egy elsődleges kulcs. Ez a mező csak abban az esetben nem kötelező, ha csak egy vállalatot tartalmaz az adatbázis.
agency_name	Kötelező. Ez a mező tartalmazza a teljes nevét a közlekedési vállalatnak. A Google Térkép ezt a nevet fogja megjeleníteni.
agency_url	Kötelező. Ebben a mezőben kell megadni a vállalat honlapjának a címét. Ennek tartalmaznia kell a teljes címet, tehát a http:// vagy https:// nyitó karaktersort és minden egyéb speciális karaktert.
agency_timezone	Kötelező. Itt kell megadni, hogy az adott vállalat melyik időzónában szolgáltat. Az időzónának soha nem tartalmaznak szünetet, csak alulvonást. A megadásnál az ISO 3166-1 alpha-2 és az ISO 6709 előírásai az irányadóak. Amennyiben több szolgáltató is meg van adva a fájlban, úgy mindegyiknek egy időzónába kell esniük.
agency_lang	Nem kötelező. Ebben a mezőben az ISO 639-1 szerinti kétbetűs kód szerepel, mellyel a vállalat elsődleges nevét kell megadni. A

	megadás során a kis- és nagy betűk nem számítanak. Ez határozza meg a nagybetű és más nyelvspecifikus tulajdonságokat a jelentésre nézve.
agency_phone	Nem kötelező. Itt kell megadni a vállalat telefonos elérhetőségét. Ez egy szöveg típusú mező, mely a vállalat szolgáltatási helyén szokásos formátumban jeleníti meg a telefonszámot. A mező elválasztó karaktereket és tárcsázható karaktereket is tartalmazhat, de semmi más magyarázó karaktert nem.
agency_fare_url	Nem kötelező. Ebben a mezőben kell megadni azt a honlapcímet, ahol az utasok jegyeket vagy egyéb szolgáltatásokat vásárolhatnak online. A honlapcímnak itt is tartalmaznia kell a teljes címet, tehát a http:// vagy https:// nyitó karaktersort és minden egyéb speciális karaktert.

2.) *Stops (Megállók) tábla*

Mező neve	Részletek
stop_id	Kötelező. A stop_id egy konstans, mely egyértelműen definiálja a megállót vagy az állomást.
stop_code	Nem kötelező. Ez a mező egy rövid szöveget vagy számot tartalmaz, mely az utasoknak azonosítja a megállót. A megállókódokat gyakran alkalmazzák mobiltelefon alapú közlekedési információs rendszerekben, vagy a kinyomatott menetrendekben, hogy egyszerűbbé tegyék az utasoknak a menetrend olvasást, vagy a valós idejű érkezési idők megjelenítését. Ezt a mezőt csak a megálló kódok tárolására használjuk, egyéb azonosító kódokat a stop_id-ben tároljunk. A mezőt üresen kell hagyni, ha a megálló nem rendelkezik kóddal.
stop_name	Kötelező. Ebben a mezőben a megálló vagy az állomás neve tárolható. Lehetőleg olyan nevet kell használni, melyet a helyi lakosok és a turisták is megértenek.
stop_desc	Nem kötelező. A stop_desc mezőben a megálló leírását lehet eltárolni. Lehetőleg csak hasznos információkat tároljunk ebben a mezőben, és ne egyszerűen csak a megálló nevét írjuk be még egyszer.
stop_lat	Kötelező. Ebben a mezőben kell megadni a megálló vagy állomás szélességi koordinátáját. Csak WGS 84-nek megfelelő érték adható meg.
stop_lon	Kötelező. Ebben a mezőben kell megadni a megálló vagy állomás hosszúsági koordinátáját. Csak WGS 84-nek megfelelő érték adható meg -180-tól 180-ig.
zone_id	Nem kötelező. Ez a mező határozza meg, hogy a megálló melyik díjszabási zónába tartozik a díjszabási azonosító segítségével. A mező kötelezően kitöltendő, ha díjszabási információt szeretnénk megadni a fare_rules tábla felhasználásával. Ha a megálló azonosító megfelel a megállónak, akkor ez az azonosító figyelmen kívül marad.

stop_url	Nem kötelező. Ebben a mezőben a konkrétan a megállóra mutató hivatkozás tárolandó. Ennek különböznie kell az agency_url-től és a route_url-től. A honlapcímnak itt is tartalmaznia kell a teljes címet, tehát a http:// vagy https:// nyitó karaktersort és minden egyéb speciális karaktert.
location_type	Nem kötelező. Ez a mező határozza meg, hogy egy egyszerű megálló, vagy egy egész állomást jelent a megálló azonosító. Amennyiben nincs megadva, vagy üres értékű, akkor egyszerű megállóként kezeli a rendszer. A következő értékek adhatók meg: <ul style="list-style-type: none"> – 0 vagy üres: Megálló. Olyan helyek, ahol egyszerű utas csere történik. – 1: Állomás. Egy olyan épület vagy terület, ahol több megálló is található.
parent_station	Nem kötelező. Olyan megállók adhatók meg itt, melyek fizikailag egy állomáson belül találhatóak, ez a mező rendeli hozzá a megfelelő állomást. Amennyiben ez a mező kitöltésre kerül, akkor a jelen táblának is tartalmaznia kell legalább egy olyan sort, ahol a location_type értéke 1.

3.) *Routes (Viszonylatok) tábla*

Mező neve	Részletek
route_id	Kötelező. Az itt tárolt elemek egyértelműen azonosítják a viszonylatot. Az itt tárolt értékek egyedi azonosítók.
agency_id	Nem kötelező. Ebben a mezőben adható meg a viszonylathoz tartozó közlekedési vállalat. Ez a mező kapcsolatban van az agency táblával. Abban az esetben használandó ez a mező, amennyiben több vállalat viszonylatai is regisztrálva vannak.
route_short_name	Kötelező. Itt tároljuk a rövid nevét a viszonylatnak. Általában egy rövid, elvont azonosító, pl.: „32”, „100X” vagy „Zöld” ami alapján az utasok azonosíthatják a járatot, azonban ez nem ad semmiféle információt a viszonylat szolgáltatási területéről. Amennyiben ilyen nincs, akkor a hosszú viszonylat nevet kell használni, ezt pedig üres szöveggel kell feltölteni.
route_long_name	Kötelező. Ebben a mezőben tároljuk a viszonylat hosszú nevét. Ez a név általában sokkal több információt ad a rövid névhez képest. Általában a megállót vagy a célállomást is tartalmazza. Amennyiben ilyen nincs, akkor a rövid viszonylat nevet kell használni, ezt pedig üres szöveggel kell feltölteni.
route_desc	Nem kötelező. Ez a mező a viszonylat leírások tárolására szolgál. Lehetőleg hasznos információkkal töltsük fel ezt a mezőt, és ne csak a viszonylat nevét írjuk ide ismét. Például: A vonatok Inwood-207 St, Manhattan és Far Rockaway-Mott Avenue, Queens között minden időszakban közlekednek. Hajnali 6 és éjfél között is, a plusz A vonatok pedig Inwood-207 és Lefferts

	Boulevard között közlekednek (a vonatok jellemzően Lefferts Blvd és Far Rockaway között közlekednek)
route_type	Kötelező. Ebben a mezőben adható meg a viszonylaton üzemelő szolgáltatás típusa. A következő értékek adhatók meg: <ul style="list-style-type: none"> – 0 – Villamos, Közúti villamos, Kisvasút. Minden olyan városi vasút, mely az utca szintjén üzemel a város területén belül. – 1 – Földalatti, Metró. Minden olyan vasúti közlekedési eszköz mely a föld alatt üzemel a város területén belül. – 2 – Vasút. Nagytávolságú, vagy Inter City vonatokra. – 3 – Busz. Rövid és nagytávolságú buszjáratokra. – 4 – Komp. Rövid és nagytávolságú hajós szolgáltatás esetén. – 5 – Sikló. Olyan utcaszintű járművek esetén, amikor a vonókábel a jármű alatt található. – 6 – Gondola, Libegő. Olyan esetekben használatos, amikor a felvonó vezeték a levegőben található, és a kabinok ezen a kábelén függenek. – 7 – Fogaskerekű. Minden olyan vasúti rendszer, mely meredek emelkedőn való felkapaszkodást valósít meg.
route_url	Nem kötelező. Ebben a mezőben a konkrétan a viszonylatra mutató hivatkozás tárolandó. Ennek különböznie kell az agency_url-től. A honlapcímnak itt is tartalmaznia kell a teljes címet, tehát a http:// vagy https:// nyitó karaktersort és minden egyéb speciális karaktert.
route_color	Nem kötelező. A rendszer automatikusan hozzárendel minden viszonylathoz egy színt, de ebben a mezőben ez külön megadható. A színnek kötelezően 6 karakter hosszú hexadecimális számnak kell lennie, pl.: 00FFFF. Amennyiben nincs megadott érték, az alapértelmezett a fehér (FFFFFF).
route_text_color	Nem kötelező. Ebben a mezőben lehet megadni, hogy a viszonylat színe előtti szöveg milyen színnel jelenjen meg. A színnek kötelezően 6 karakter hosszú hexadecimális számnak kell lennie, pl.: FFD700. Amennyiben nincs megadott érték, az alapértelmezett a fekete (000000).

4.) *Trips (Utazások) tábla*

Mező neve	Részletek
route_id	Kötelező. Ebben a mezőben a routes táblára utaló viszonylatazonosítók találhatóak.
service_id	Kötelező. Az itt található bejegyzések egyértelműen azonosítják a megadott dátumokat, amikor a szolgáltatás elérhető egy vagy több viszonylaton. Ez az érték hivatkozik a calendar vagy a calendar_dates táblára.
trip_id	Kötelező. Az itt található értékek egyértelműen azonosítják az utazásokat.
trip_headsign	Nem kötelező. Az itt megadott érték jelenik meg az utasnak, mint

	célállomás. Abban az esetben használatos, amennyiben különbséget kívánunk tenni az egyazon útvonalrészletek közlekedő járatok között. Amennyiben a fejléc változik az utazás közben, akkor a stops táblában a stop_headsign értékével felülírható.
trip_short_name	Nem kötelező. Ebben a mezőben megadott szöveg fog mejelelni a menetrendben jelzőtáblákon, hogy azonosítsa az utazást az utas számára. Például, hogy a vonat számokat azonosítsa az ingázó utasok számára. Amennyiben az utasok általában nem használnak ilyen meghatározásokat, akkor hagyja üresen ezt a mezőt. Amennyiben a mező rendelkezik értékkel, egyértelműen azonosítja az utazást az adott napi közlekedési rendben, ezért ne használjunk azonos elnevezéseket a célállomással vagy a korlátozott/gyors járatokkal.
direction_id	Nem kötelező. Ez a mező egy bináris értéket tartalmaz, mely meghatározza az utazás irányát egy utazás során. A mező célja, hogy megkülönböztessük az azonos kétirányú viszonylatok irányát. Ezt a mezőt ne használjuk irányításra, ez megadja a lehetőséget, hogy megkülönböztessük az irányokat a menetrendek közreadásánál. Az irányokhoz a trip_headsign mezőben lehet nevet adni. <ul style="list-style-type: none"> - 0 – az egyik irányba utazás (pl.: kifelé utazás) - 1 – a másik irányba utazás (pl.: befele utazás)
block_id	Nem kötelező. Ez a mező azonosítja azt a blokkot, ahova az utazás tartozik. Egy blokk két vagy több egymást követő utazást tartalmaz azonos járművel, amivel az utas úgy tud egy másik utazásra áttérni, hogy nem hagyja el a járművet. Ennek az azonosítónak kettő vagy több utazásra kell hivatkozni a táblában.
shape_id	Nem kötelező. Ebben a mezőben adhatóak meg azok az azonosítók, melyek meghatározzák az utazás alakját. Ennek az értéknek hivatkozni kell a shapes táblára. A shapes tábla lehetőséget ad, hogy meghatározzuk, hogy egy vonalnak milyen módon kell mejelelnie a térképen az utazás megjelenítésére.

5.) *Stop_times (Megállóhelyi idők) tábla*

Mező neve	Részletek
trip_id	Kötelező. A trip_id tartalmazza az utazás azonosítóját. Ez az érték hivatkozik a trips táblára.
arrival_time	Kötelező. Az adott állomáshoz tartalmazó érkezési időket tartalmazza a mező. Az időt dél mínusz 12 órához viszonyítva adjuk meg (normál esetben éjfél, de a téli és a nyári időszámításnál nem) a szolgáltatás kezdetének a napján. Olyan esetekben, amikor a szolgáltatás tovább tart mint éjfél, akkor a következő formátumban kell megadni az időt (24:00:00 azaz óó:pp:mm) úgy, hogy a 24 órához adjuk hozzá a helyi aktuális időt. Amennyiben az érkezési és indulási idők azonos értékűek, akkor az indulási időnek is ezt az időt kell megadni. Minden esetben meg kell adni az első és az utolsó megálló indulá-

	<p>si idejét. Amennyiben a megálló nem egy konkrét időponthoz kötött, akkor üres szöveget kell adni mind az érkezési és indulási idő mezőnek. Amennyiben nincs érték a mezőben, akkor időben legközelebbi időponttal kezeljük a menetrendben. Annak érdekében, hogy az útvonalszámítás pontos legyen, kérjük adja meg az összes indulási és érkezési időt. Ne becsülje ezeket az értékeket. Az időt a következő formátumban kell megadni: óó:pp:mm (az ó:pp:mm is elfogadható, amennyiben az óra 0 és 9 között van). Az időt semmi esetben se tagolja szünettel. A következő táblázatban példák láthatók az időpontok megadására ebbe a mezőbe:</p> <table border="1" data-bbox="550 593 1204 795"> <thead> <tr> <th>Időpont</th> <th>arrival time mező értéke</th> </tr> </thead> <tbody> <tr> <td>de. 08:10:00</td> <td>08:10:00 vagy 8:10:00</td> </tr> <tr> <td>du. 01:05:00</td> <td>13:05:00</td> </tr> <tr> <td>du. 07:40:00</td> <td>19:40:00</td> </tr> <tr> <td>de. 01:55:00</td> <td>25:55:00</td> </tr> </tbody> </table> <p>Megjegyzés: Azon utazásoknak, melyek több napon keresztül zajlanak, nagyobb az értékük mint 24 óra. Például ha egy utazás du. 10:30:00-kor kezdődik és a következő nap de. 2:15:00-kor fejeződik be, akkor a következőképp adjuk meg őket: 22:30:00 és 26:15:00. Amennyiben így adjuk meg az adatokat: 22:30:00 és 2:15:00 akkor nem fogunk azonos eredményt kapni.</p>	Időpont	arrival time mező értéke	de. 08:10:00	08:10:00 vagy 8:10:00	du. 01:05:00	13:05:00	du. 07:40:00	19:40:00	de. 01:55:00	25:55:00
Időpont	arrival time mező értéke										
de. 08:10:00	08:10:00 vagy 8:10:00										
du. 01:05:00	13:05:00										
du. 07:40:00	19:40:00										
de. 01:55:00	25:55:00										
departure_time	<p>Kötelező. Az adott állomáshoz tartalmazó indulási időket tartalmazza a mező. Az időt dél mínusz 12 órához viszonyítva adjuk meg (normál esetben éjfél, de a téli és a nyári időszámításnál nem) a szolgáltatás kezdetének a napján. Olyan esetekben, amikor a szolgáltatás tovább tart mint éjfél, akkor a következő formátumban kell megadni az időt (24:00:00 azaz óó:pp:mm) úgy, hogy a 24 órához adjuk hozzá a helyi aktuális időt. Amennyiben az érkezési és indulási idők azonos értékűek, akkor az indulási időnek is ezt az időt kell megadni.</p> <p>Minden esetben meg kell adni az első és az utolsó megálló érkezési idejét. Amennyiben a megálló nem egy konkrét időponthoz kötött, akkor üres szöveget kell adni mind az érkezési és indulási idő mezőnek. Amennyiben nincs érték a mezőben, akkor időben legközelebbi időponttal kezeljük a menetrendben. Annak érdekében, hogy az útvonalszámítás pontos legyen, kérjük adja meg az összes indulási és érkezési időt. Ne becsülje ezeket az értékeket. Az időt a következő formátumban kell megadni: óó:pp:mm (az ó:pp:mm is elfogadható, amennyiben az óra 0 és 9 között van). Az időt semmi esetben se tagolja szünettel. A következő táblázatban példák láthatók az időpontok megadására ebbe a mezőbe:</p> <table border="1" data-bbox="550 1747 1252 1948"> <thead> <tr> <th>Időpont</th> <th>departure time mező értéke</th> </tr> </thead> <tbody> <tr> <td>de. 08:10:00</td> <td>08:10:00 vagy 8:10:00</td> </tr> <tr> <td>du. 01:05:00</td> <td>13:05:00</td> </tr> <tr> <td>du. 07:40:00</td> <td>19:40:00</td> </tr> <tr> <td>de. 01:55:00</td> <td>25:55:00</td> </tr> </tbody> </table> <p>Megjegyzés: Azon utazásoknak, melyek több napon keresztül</p>	Időpont	departure time mező értéke	de. 08:10:00	08:10:00 vagy 8:10:00	du. 01:05:00	13:05:00	du. 07:40:00	19:40:00	de. 01:55:00	25:55:00
Időpont	departure time mező értéke										
de. 08:10:00	08:10:00 vagy 8:10:00										
du. 01:05:00	13:05:00										
du. 07:40:00	19:40:00										
de. 01:55:00	25:55:00										

	<p>zajlanak, nagyobb az értékük mint 24 óra. Például ha egy utazás du. 10:30:00-kor kezdődik és a következő nap de. 2:15:00-kor fejeződik be, akkor a következőképp adjuk meg őket: 22:30:00 és 26:15:00. Amennyiben így adjuk meg az adatokat: 22:30:00 és 2:15:00 akkor nem fogunk azonos eredményt kapni.</p>
stop_id	<p>Kötelező. A megállók azonosítóit tartalmazza a mező, melyek egyértelműen azonosítják a megállókat. Több viszonylat is használhat egy megállót. A stop_id a stops táblára hivatkozik. Amennyiben a location_type mezőnek van értéke a stops táblában, akkor azon megállónak, melyekre ebben a táblában hivatkozunk 0 értékűnek kell lennie.</p> <p>Amennyiben lehetséges, a megálló azonosítóknak nem szabad változniuk a frissítések során. Tehát ha egy adatbázisban egy „A” megállónak az azonosítója „1”, akkor a következő frissítés esetén is ennek kell lennie. Amennyiben a megállóhoz nem tartozik idő adat, akkor az arrival_time és departure_time-nak is üres értéket kell tartalmazniuk.</p>
stop_sequence	<p>Kötelező. Ebben a mezőben lévő értékek segítségével tehetők sorrendbe a megállók egy utazás során. Az itt lévő értékeknek olyan nem negatív valós számoknak kell lenniük, melyek az utazás során szigorúan monoton növekednek.</p> <p>Például, az első megállónak az utazás során az itt tárolt értéke 1, a másodiknak 23, a harmadiknak 40 és így tovább.</p>
stop_headsign	<p>Nem kötelező. Ebben a mezőben egy olyan szöveget tárolunk, mely az utasok számára a utazás irányát azonosítja. Abban az esetben használjuk, amikor a trips táblában lévő trip_headsign értéke helyett mást kívánunk megjeleníteni, azaz járat célpontja az utazás közben változik. Amennyiben az érték nem változik, akkor a trip_headsign-ben kell megadni a végcél.</p>
pickup_type	<p>Nem kötelező. Ez a mező határozza meg, hogy az utasok milyen módon szállhatnak fel a járműre, azaz része a normál menetrendnek, vagy a felszállás nem lehetséges. A mező lehetőséget ad a közlekedési vállalatnak, hogy megadja, hogy az utasoknak milyen módon kell értesíteni a vállalatot vagy a jármű vezetőjét a felszállási szándékukról az adott megállóban.</p> <p>A következő értékek lehetnek a mezőben:</p> <ul style="list-style-type: none"> – 0 – Normál menetrendek megfelelő felszállóhely. – 1 – Felszállás nem lehetséges. – 2 – A közlekedési vállalatot telefonon értesíteni kell a felszálló utasokról. – 3 – A járművezetőt kell értesíteni a felszállási szándékról. <p>0 érték az alapértelmezett.</p>
drop_off_type	<p>Nem kötelező. Ez a mező határozza meg, hogy az utasok milyen módon szállhatnak le a járműről, azaz része a normál menetrendnek, vagy a leszállás nem lehetséges. A mező lehetőséget ad a közlekedési vállalatnak, hogy megadja, hogy az utasoknak milyen módon kell értesíteni a vállalatot vagy a jármű vezetőjét a leszállási szándékukról az adott megállóban.</p>

	<p>A következő értékek lehetnek a mezőben:</p> <ul style="list-style-type: none"> – 0 – Normál menetrendek megfelelő leszállóhely. – 1 – Leszállás nem lehetséges. – 2 – A közlekedési vállalatot telefonon értesíteni kell a leszálló utasokról. – 3 – A járművezetőt kell értesíteni a leszállási szándékról. <p>0 érték az alapértelmezett.</p>
shape_dist_traveled	<p>Nem kötelező. Amennyiben a mező rendelkezik értékkel, akkor megadja az első alakponttól mért távolságot. Az itt tárolt érték a valós utazási távolság az adott viszonylaton méterben vagy „lábban” megadva. Például ha a busz 5,25 km-ert megy a kiindulási alaktól a megállóig, akkor itt „5.25”-öt kell megadni értéként. Az itt található információ meghatározza az utazástervezőnek, hogy milyen hosszban kell megjeleníteni egy utazás során ezt a távot. Az értékeknek a stop_sequence-hez hasonlóan szigorúan monoton növekedniük kell, tehát adhatók meg a visszaút távolság adatai.</p> <p>A távolságokat a shapes táblának megfelelő mértékegységben kell megadni.</p>

6.) *Calendar (Naptár) tábla*

Mező neve	Részletek
service_id	<p>Kötelező. Azonosító mező, mely egyértelműen azonosítja azt a közlekedési rendet, amikor a szolgáltatás elérhető legalább napi egy járattal. Minden azonosító csak egyszer szerepelhet a táblában. Az itt található értékek egyedi azonosítók, melyekre a trips tábla hivatkozik.</p>
monday	<p>Kötelező. A Monday mező egy bináris értéket tartalmaz, mely azonosítja, hogy a szolgáltatás elérhető-e a menetrend érvényessége alatt az adott napon.</p> <ul style="list-style-type: none"> – Az „1” érték jelenti azt, hogy az összes hétfőn elérhető a szolgáltatás a menetrend érvényessége alatt. Az érvényességet a start_date és end_date mezőkben kell megadni. – A „0” érték jelenti azt, hogy egyik hétfőn sem elérhető a szolgáltatás a menetrend érvényessége alatt. <p>Megjegyzés: Lehetőség van kivételek megadására, például ünnepek, vagy munkaszüneti napok esetén a calendar_dates táblában.</p>
tuesday	<p>Kötelező. A Tuesday mező egy bináris értéket tartalmaz, mely azonosítja, hogy a szolgáltatás elérhető-e a menetrend érvényessége alatt az adott napon.</p> <ul style="list-style-type: none"> – Az „1” érték jelenti azt, hogy az összes kedden elérhető a szolgáltatás a menetrend érvényessége alatt. Az érvényességet a start_date és end_date mezőkben kell megadni. – A „0” érték jelenti azt, hogy egyik kedden sem elérhető a szolgáltatás a menetrend érvényessége alatt. <p>Megjegyzés: Lehetőség van kivételek megadására, például ünnepek</p>

	napok, vagy munkaszüneti napok esetén a <code>calendar_dates</code> táblában.
wednesday	<p>Kötelező. A <code>Wednesday</code> mező egy bináris értéket tartalmaz, mely azonosítja, hogy a szolgáltatás elérhető-e a menetrend érvényessége alatt az adott napon.</p> <ul style="list-style-type: none"> – Az „1” érték jelenti azt, hogy az összes szerdán elérhető a szolgáltatás a menetrend érvényessége alatt. Az érvényességet a <code>start_date</code> és <code>end_date</code> mezőkben kell megadni. – A „0” érték jelenti azt, hogy egyik szerdán sem elérhető a szolgáltatás a menetrend érvényessége alatt. <p>Megjegyzés: Lehetőség van kivételek megadására, például ünnepnapok, vagy munkaszüneti napok esetén a <code>calendar_dates</code> táblában.</p>
thursday	<p>Kötelező. A <code>Thursday</code> mező egy bináris értéket tartalmaz, mely azonosítja, hogy a szolgáltatás elérhető-e a menetrend érvényessége alatt az adott napon.</p> <ul style="list-style-type: none"> – Az „1” érték jelenti azt, hogy az összes csütörtökön elérhető a szolgáltatás a menetrend érvényessége alatt. Az érvényességet a <code>start_date</code> és <code>end_date</code> mezőkben kell megadni. – A „0” érték jelenti azt, hogy egyik csütörtökön sem elérhető a szolgáltatás a menetrend érvényessége alatt. <p>Megjegyzés: Lehetőség van kivételek megadására, például ünnepnapok, vagy munkaszüneti napok esetén a <code>calendar_dates</code> táblában.</p>
friday	<p>Kötelező. A <code>Friday</code> mező egy bináris értéket tartalmaz, mely azonosítja, hogy a szolgáltatás elérhető-e a menetrend érvényessége alatt az adott napon.</p> <ul style="list-style-type: none"> – Az „1” érték jelenti azt, hogy az összes pénteken elérhető a szolgáltatás a menetrend érvényessége alatt. Az érvényességet a <code>start_date</code> és <code>end_date</code> mezőkben kell megadni. – A „0” érték jelenti azt, hogy egyik pénteken sem elérhető a szolgáltatás a menetrend érvényessége alatt. <p>Megjegyzés: Lehetőség van kivételek megadására, például ünnepnapok, vagy munkaszüneti napok esetén a <code>calendar_dates</code> táblában.</p>
saturday	<p>Kötelező. A <code>Saturday</code> mező egy bináris értéket tartalmaz, mely azonosítja, hogy a szolgáltatás elérhető-e a menetrend érvényessége alatt az adott napon.</p> <ul style="list-style-type: none"> – Az „1” érték jelenti azt, hogy az összes szombaton elérhető a szolgáltatás a menetrend érvényessége alatt. Az érvényességet a <code>start_date</code> és <code>end_date</code> mezőkben kell megadni. – A „0” érték jelenti azt, hogy egyik szombaton sem elérhető a szolgáltatás a menetrend érvényessége alatt. <p>Megjegyzés: Lehetőség van kivételek megadására, például ünnepnapok, vagy munkaszüneti napok esetén a <code>calendar_dates</code> táblában.</p>
sunday	<p>Kötelező. A <code>Sunday</code> mező egy bináris értéket tartalmaz, mely azonosítja, hogy a szolgáltatás elérhető-e a menetrend érvényessége alatt az adott napon.</p> <ul style="list-style-type: none"> – Az „1” érték jelenti azt, hogy az összes vasárnap elérhető a szolgáltatás a menetrend érvényessége alatt. Az érvényességet a <code>start_date</code> és <code>end_date</code> mezőkben kell megadni.

	<p>– A „0” érték jelenti azt, hogy egyik vasárnap sem elérhető a szolgáltatás a menetrend érvényessége alatt.</p> <p>Megjegyzés: Lehetőség van kivételek megadására, például ünnepek, vagy munkaszüneti napok esetén a <code>calendar_dates</code> táblában.</p>
<code>start_date</code>	<p>Kötelező. Ez a mező tartalmazza az érvényességi idő kezdetét.</p> <p>A dátumot a következő formátumban kell megadni: <code>éééé.hh.nn</code>.</p>
<code>end_date</code>	<p>Kötelező. Ez a mező tartalmazza az érvényességi idő végét.</p> <p>A dátumot a következő formátumban kell megadni: <code>éééé.hh.nn</code>.</p>

7.) Calendar_dates (Naptári dátumok) tábla

Ez egy nem kötelező tábla. Itt lehetőség van adott dátumok szerint egyenként megadni, hogy az adott szolgáltatás elérhető-e az adott napon. Ez két módon tehető meg:

- Ajánlott: A `calendar_dates` táblát a `Calendar` táblával együtt használva itt megadhatóak azok a kivételek, amikor az alap közlekedési rendtől eltérő menetrenddel közlekednek a járművek. Amennyiben a közlekedési rendben csak néhány kivétel található (pl. ünnepek, tanítási időszak), ez egy jó megoldás.
- Választható megoldás: A `Calendar` táblát egyáltalán nem töltjük fel, és helyette az összes menetrendet ebben a táblában adjuk meg. Olyan esetekben lehet hasznos ez a megoldás, amikor a menetrend a hónap nagy részében eltérő, vagy amikor a hét napjaitól eltérően akarjuk megadni.

Mező neve	Részletek
<code>service_id</code>	<p>Kötelező. Azonosító mező, mely egyedileg azonosítja a közlekedési rendet, amikor a szolgáltatás elérhető legalább napi egy járat-tal. Minden (<code>service_id</code>, <code>date</code>) páros csak egyszer szerepelhet a táblában. Amennyiben a <code>service_id</code> mindkét táblában (<code>Calendar</code>, <code>Calendar_date</code>) szerepel, akkor az itt található értékek mindig felülírják az ott található értékeket. Az itt található értékekre a <code>trips</code> tábla hivatkozik.</p>
<code>date</code>	<p>Kötelező. Ez a mező tartalmazza azon dátumokat, amikor a szolgáltatás az általános közlekedési rendtől eltérően működik. Az <code>exception_type</code> mező alkalmazható arra, hogy megadjuk azt, hogy elérhető-e az adott napon a szolgáltatás.</p> <p>A dátumot a következő formátumban kell megadni: <code>éééé.hh.nn</code>.</p>
<code>exception_type</code>	<p>Kötelező. Ez a mező határozza meg, hogy a szolgáltatás elérhető-e az adott napon.</p> <ul style="list-style-type: none"> – A mező értéke 1, amennyiben az adott naphoz hozzá kívánjuk adni a szolgáltatást. – A mező értéke 2, amennyiben el kívánjuk távolítani a szol-

	<p>gáltatást az adott naptól.</p> <p>Például, ha egy járat csak ünnepnapokon üzemel, vagy csak az adott napon nem jár. Különböző szolgáltatási azonosító (service_id) szükséges a normál és az ünnepi közlekedési rend esetén. Praktikusan ebben a táblában célszerű megadni az ünnepi menetrend eltéréseit a normál menetrendhez képest.</p>
--	---

8.) *Fare_attributes (díjszabási tulajdonságok) tábla*

Mező neve	Részletek
fare_id	Kötelező. Ez a mező egyedi azonosítót tartalmaz a díjszabási kategóriák számára. A mező egyedi azonosító.
price	Kötelező. Az itt található érték a currency_type mezőben definiált pénznemben tartalmazza az árat.
currency_type	Kötelező. Ebben a mezőben adható meg az árat pénzneme. A pénznemeket az ISO 4217-es szabványnak megfelelően betűkóddal kell megadni.
payment_method	Kötelező. Az itt megadott értékek azonosítják, hogy az utazásért mikor kell fizetni. A következő értékek szerepelhetnek a mezőben <ul style="list-style-type: none"> – 0 – A jármű fedélzetén kell fizetni. – 1 – A járműre való felszálláskor kell fizetni.
transfers	Kötelező. Ebben a mezőben határozható meg, hogy hány átszállás engedélyezett ezért az árért. <ul style="list-style-type: none"> – 0 – Az átszállás nem engedélyezett – 1 – Egy átszállás engedélyezett. – 2 – Két átszállás engedélyezett – üres érték – Amennyiben a mezőnek üres értéke van, akkor nincs korlátozva az átszállások száma.
transfer_duration	Nem kötelező. Itt adható meg másodperben mérve, hogy milyen hosszú idő áll rendelkezésre az átszállókra mielőtt a jegy érvényességi ideje lejár. Amennyiben a transfers mezőben 0 érték szerepel, akkor az itt szereplő érték azt az érvényességi időt jelenti, ameddig az átszállás nem engedélyezett. Abban az esetben, amikor nem kívánjuk megadni a jegy érvényességi idejét, akkor a mezőt hagyjuk üresen, vagy üres értéket adjunk neki akkor, amikor a transfers mező értéke 0.

9.) *Fare_rules (Díjszabási szabályok) tábla*

Ez a tábla nem kötelező. Ebben a táblában adható meg, hogy a fare_attributes táblában lévő értékeket hogyan alkalmazza a rendszer az útvonalajánlat során. A legtöbb díjrendszer a következő szabályok valamilyen kombinációján alapszik:

- Az ár az induló vagy a célállomástól függ.
- Az ár az utazás során érintett zónáktól függ.
- Az ár az utazás során igénybevett járatoktól függ.

A következő linken található néhány példa a díjszabás rögzítésére a fare_rules és a fare_attributes táblában:

<http://code.google.com/p/googletransitdatafeed/wiki/FareExamples>

Mező neve	Részletek
fare_id	Kötelező. Ez a mező egyedi azonosítót tartalmaz a díjszabási kategóriák számára. A mező a fare_attributes táblával van kapcsolatban.
route_id	Nem kötelező. A mező összekapcsolja a díjszabás azonosítót a viszonylattal. A viszonylat azonosító a routes táblára hivatkozik. Amennyiben több viszonylatnak is azonos díjszabása van, szabályok adhatóak meg a fare_rules táblában az összes viszonylatra. Példának okáért, ha a „b” díjszabási kategória érvényes a „TSW” és „TSE” viszonylatra is, a fare_rules táblának tartalmaznia kell a következő sorokat: b, TSW b, TSE
origin_id	Nem kötelező. A mező összekapcsolja a díjszabás azonosítót a kiindulási zone_id mezővel. A zóna azonosító a stops táblára hivatkozik. Amennyiben több kiindulási pont is azonos díjszabási zónába tartozik, szabályok adhatóak meg a fare_rules táblában az összes kiindulási pontra. Példának okáért, ha a „b” díjszabási kategória érvényes az összes 2-es vagy 8-as zónából induló utazásra, a fare_rules táblának tartalmaznia kell a következő sorokat: b, , 2 b, , 8
destination_id	Nem kötelező. A mező összekapcsolja a díjszabás azonosítót az érkezési zone_id mezővel. A zóna azonosító a stops táblára hivatkozik. Amennyiben több érkezési pont is azonos díjszabási zónába tartozik, szabályok adhatóak meg a fare_rules táblában az összes érkezési pontra. Példának okáért, ha az origin_ID és a destination_ID együtt használhatók a „b” díjszabási kategóriára és ez érvényes az összes 3-as és 4-es illetve a 3-as és az 5-ös zóna közötti utazásra, a fare_rules táblának tartalmaznia kell a következő sorokat: b, , 3, 4

	b, , 3, 5
contains_id	<p>Nem kötelező. A mező összekapcsolja a díjszabás azonosítót a zone_id mezővel, hivatkozva a stops táblára. Ezután a díjszabási azonosítót összerendelhető az összes olyan zónával, melyet a mező tartalmaz az útitervben.</p> <p>Például a „c” díjszabási kategória össze van kapcsolva a GRT viszonylattal, mely az 5-ös, 6-os és 7-es zónákat is érinti, ekkor a fare_rules táblának tartalmaznia kell a következő sorokat:</p> <p>C, GRT, , , 5 C, GRT, , , 6 C, GRT, , , 7</p> <p>Amiatt, mert a mezőnek igazodnia kell a díjszabáshoz, az olyan útvonalak, melyek keresztül haladnak 5-ös és 6-os zónán, de a 7-esen nem, azoknak nem lehet „c” díjszabási kategóriájuk. További példák a következő linken érhetőek el: (http://code.google.com/p/googletransitdatafeed/wiki/FareExamples)</p>

10.) *Shapes (alakzatok) tábla*

Mező neve	Részletek
shape_id	Kötelező. Ez a mező az alakzatok egyedi azonosítóját tartalmazza.
shape_pt_lat	<p>Kötelező. Ebben a mezőben kell összerendelni az alakzatok szélességi koordinátáját az alakzat azonosítójával. Csak WGS 84-nek megfelelő érték adható meg.</p> <p>Például, ha a „A_shp” alakzat három ponttal adott, akkor a táblának a következő sorai lesznek:</p> <p>A_shp, 37.61956, -122.48161, 0 A_shp, 37.64430, -122.41070, 6 A_shp, 37.65863, -122.30839, 11</p>
shape_pt_lon	<p>Kötelező. Ebben a mezőben kell összerendelni az alakzatok hosszúsági koordinátáját az alakzat azonosítójával. Csak WGS 84-nek megfelelő érték adható meg -180-tól 180-ig.</p> <p>Például, ha a „A_shp” alakzat három ponttal adott, akkor a táblának a következő sorai lesznek:</p> <p>A_shp, 37.61956, -122.48161, 0 A_shp, 37.64430, -122.41070, 6 A_shp, 37.65863, -122.30839, 11</p>
shape_pt_sequence	<p>Kötelező. Ez a mező határozza meg az alakzatok összerendelt szélességi és hosszúsági koordináták sorrendjét. Az itt lévő értékeknek olyan nem negatív valós számoknak kell lenniük, melyek az utazás során szigorúan monoton növekednek.</p> <p>Például, ha a „A_shp” alakzat három ponttal adott, akkor a táblának a következő sorai lesznek:</p> <p>A_shp, 37.61956, -122.48161, 0 A_shp, 37.64430, -122.41070, 6 A_shp, 37.65863, -122.30839, 11</p>
shape_dist_traveled	Nem kötelező. Amennyiben a mező kitöltésre kerül, akkor az itt található értékek a kezdőponttól számított távolságot jelentik. Az

	<p>itt lévő értékek azt a valós távolságot mutatják, melyet az utazás során megteszünk mérföldben vagy kilométerben. Az itt található értékek lehetővé teszik, hogy az útvonaltervező meghatározza, hogy hány alakzatot rajzoljon ki az utazás megjelenítése során. Az itt található értékek a shape_pt_sequence mezőhöz hasonlóan szigorúan monoton növekedőknek kell lenniük, azaz nem tartalmazhat visszafordulást.</p> <p>Az itt alkalmazott értékek mértékegységei a stop_times táblában megadottakkal egyeznek meg.</p> <p>Például, ha egy busz keresztülhalad az alábbiakban három ponttal megadott „A_shp” alakzaton, a mezőben megadottak a következők szerint néznének ki kilométerben:</p> <p>A_shp, 37.61956, -122.48161, 0, 0 A_shp, 37.64430, -122.41070, 6, 6.8310 A_shp, 37.65863, -122.30839, 11, 15.8765</p>
--	--

11.) *Frequencies (Gyakoriságok) tábla*

Ez a tábla nem kötelező. A tábla célja, hogy az olyan menetrendet lehessen rögzíteni, melyeknek nincsen előre meghatározott fix menetrendje. Amennyiben egy útvonal ebben a táblában adott, akkor az útvonal ajánló figyelmen kívül hagyja az arrival_time és departure_time mezőbeli értékeket a stop_times táblában. Ehelyett a stop_times tábla a megállások gyakoriságát és a megállók közötti időt tartalmazza.

Mező neve	Részletek
trip_id	Kötelező. A mező egy olyan azonosítót tartalmaz, mely azonosítja azokat az utazásokat, melyekre a szolgáltatási gyakoriságot alkalmazzuk. Az utazás azonosítók a trips táblára hivatkoznak.
start_time	Kötelező. Ez a mező határozza meg azt az időpontot, amikor az adott gyakorisággal megadott szolgáltatás indul. Az időt dél mínusz 12 órához viszonyítva adjuk meg (normál esetben éjfél, de a téli és a nyári időszámításnál nem) a szolgáltatás kezdetének a napján. Olyan esetekben, amikor a szolgáltatás tovább tart mint éjfél, akkor a következő formátumban kell megadni az időt (24:00:00 azaz óó:pp:mm) úgy, hogy a 24 órához adjuk hozzá a helyi aktuális időt (pl.: 25:35:00).
end_time	Kötelező. Ez a mező határozza meg azt az időpontot, amikor az adott gyakorisággal megadott szolgáltatás más gyakoriságúra vált (vagy befejeződik) az útvonal első megállójánál. Az időt dél mínusz 12 órához viszonyítva adjuk meg (normál esetben éjfél, de a téli és a nyári időszámításnál nem) a szolgáltatás kezdetének a napján. Olyan esetekben, amikor a szolgáltatás tovább tart mint éjfél, akkor a következő formátumban kell megadni az időt (24:00:00 azaz óó:pp:mm) úgy, hogy a 24 órához adjuk hozzá a helyi aktuális időt (pl.: 25:35:00).
headway_secs	Kötelező. A mezőben két jármű megállóba érkezése közötti eltelt

	<p>időt a <code>start_time</code> és <code>end_time</code> mezővel meghatározott intervallumon belül. Az értékeket másodpercben kell megadni.</p> <p>Azok az időtartamok, melyeket a mezőben határozunk meg, lehetőleg ne fedjék egymást, mert nehéz meghatározni, hogy két haladás közül melyik érvényesül. Azonban a haladási időtartam akár meg is egyezhet egy másik befejező időponttal, példának okáért:</p> <p>A, 05:00:00, 07:00:00, 600 B, 07:00:00, 12:00:00, 1200</p>
<code>exact_times</code>	<p>Nem kötelező. Ebben a mezőben határozható meg, ha a gyakoriság alapú utazásnak hogyan kell pontosan meghatározni a menetrendjét a haladási információkból. A következő értékeket tartalmazhatja a mező:</p> <ul style="list-style-type: none"> - 0 vagy üres érték – A gyakoriság alapú út menetrendje nem pontosan meghatározható. Ez az alapértelmezés. - 1 – A menetrend pontosan meghatározható. A sorok menetrendjét a következőképpen határozza meg: $trip_start_time = start_time + x \cdot headway_secs$ minden x-re, ahol x értékei 0, 1, 2, ... lehetnek, és a $trip_start_time < end_time$. <p>Minden olyan rekordba, melynek <code>trip_id</code>-ja azonos, csak ugyanolyan érték állhat ebben a mezőben is. Amennyiben a mező értéke 1, és a <code>start_time</code> és <code>end_time</code> értékei azonosak, akkor a menetrend nem lesz elkészítve. Amennyiben a mező értéke 1, akkor fontos, hogy nagyobbá kell választani, mint az utolsó kívánt útvonal indulási ideje, de értéke nem lehet nagyobb, mint az indulási idő plusz a <code>headway_secs</code>.</p>

12.) Transfers (átszállások) tábla

A tábla kitöltése nem kötelező. Az útvonaltervező alap esetben a megállók relatív távolsága alapján választ az átszállási lehetőségek közül. Azon átszállási kapcsolatok esetén, amikor nem egyértelmű az átszállás, vagy adott választási lehetőséget célszerű megadni, ebben az esetben ebben a táblában adhatók meg szabályok az átszállási kapcsolatokhoz.

Mező neve	Részletek
<code>from_stop_id</code>	Kötelező. A mező annak a megállónak az azonosítóját tartalmazza, ami az adott átszállási kapcsolatnak a kezdőpontja. A megálló azonosítók a <code>stops</code> táblára hivatkoznak. Amennyiben a megálló azonosítóhoz tartozó állomás több megállót tartalmaz, akkor az itt megadott szabályok az összesre vonatkozni fognak.
<code>to_stop_id</code>	Kötelező. A mező annak a megállónak az azonosítóját tartalmazza, ami az adott átszállási kapcsolatnak a végpontja. A megálló azonosítók a <code>stops</code> táblára hivatkoznak. Amennyiben a megálló azonosítóhoz tartozó állomás több megállót tartalmaz, akkor az itt megadott szabályok az összesre vonatkozni fognak.

transfer_type	<p>Kötelező. Ebben a mezőben az összeköttetés típusa adható meg a megállópárok között (from_stop_id, to_stop_id). A következő értékeket tartalmazhatja a mező:</p> <ul style="list-style-type: none"> – 0 vagy üres érték – Ez az ajánlott átszállási kapcsolat. – 1 – Ez egy időzített átszállási kapcsolat. Az induló jármű az érkező járműre várakozik addig, amíg a érkező jármű utasai átszállnak az induló járműre. – 2 - Ebben az esetben adott minimum idő áll rendelkezésre az átszálláshoz. Az átszállási idő a min_transfer_time mezőben adható meg. – 3 – Nem lehetséges az átszállás a viszonylatok között.
min_transfer_time	<p>Nem kötelező. Amennyiben egy adott idő legalább szükséges az átszálláshoz a két megálló között (transfer_type=2) az itt megadott érték határozza meg ezt az időt, azaz legalább ennyi időnek kell rendelkezésre állnia az érkező jármű és az induló jármű érkezési és indulási ideje között. Az időt egy átlagos utasra kell megadni olyan tartalék idővel együtt, mely a késések esetén is biztosítja az átszállást.</p> <p>Az időt másodpercben kell megadni nem negatív egész számként.</p>

13.) *Feed_info (Adatbázis információ) tábla*

Az adatbázis információs tábla egy nem kötelező tábla. Az itt megadott információk elsősorban megáról az adatbázisról közölnek adatokat. A GTFS-ben jelenleg az agency tábla áll rendelkezésre közlekedési vállalatokról való adatközlésre. Azonban az adatközlő néhány esetben különböző szervezet a közlekedési vállalatoktól (pl. regionális szövetségek). Továbbá néhány mező inkább adatbázis leíró jellegű, mintsem a közlekedési vállalat további leírása a célja.

Mező neve	Részletek
feed_publisher_name	Kötelező. Ez a mező tartalmazza az adatközlő teljes nevét (ez meg is egyezhet a közlekedési vállalat nevével, ami az agency táblában van megadva). A GTFS alapú alkalmazások ezt a nevet jelenítik meg az adatbázis részletezésében.
feed_publisher_url	Kötelező. Ez a mező tartalmazza az adatközlő honlapjának az elérhetőségét (ez meg is egyezhet a közlekedési vállalat honlapjának a címével, ami az agency táblában van megadva). Ennek tartalmaznia kell a teljes címet, tehát a http:// vagy https:// nyitó karaktersort és minden egyéb speciális karaktert.
feed_lang	Kötelező. Ez a mező az adatbázis alapértelmezett nyelvét tartalmazza az IETF BCP 47 nyelvi kódoknak megfelelően. Ez az információ segítséget nyújt a GTFS felhasználóinak, hogy meghatározzák a nagybetűs szabályokat és az egyéb nyelvi beállításokat az adatbázis felhasználásakor. Az IETF BCP 47 leírását a következő linkeken lehet megtalálni:

	<p>http://www.rfc-editor.org/rfc/bcp/bcp47.txt és http://www.w3.org/International/articles/language-tags/.</p>
<p>feed_start_date feed_end_date</p>	<p>Nem kötelező. Az adatbázisnak teljeskörű és megbízható menetrendi adatokkal kell rendelkeznie az érvényesség kezdetétől feed_start_date-tól a feed_end_date-ig. Mindkét dátumot éééé.hh.nn formátumban kell megadni a calendar táblához hasonlóan, vagy üresen kell hagyni a mezőt, amennyiben nem elérhetőek ezek az adatok. Az adatszolgáltatókat ösztönzi, hogy ezen a megadott időtartamon kívül is adjon információt a jövőbeli szolgáltatásról, az adatbázis felhasználói szem előtt tarthatják, hogy ez nem egy végleges állapot. Amennyiben az itt megadott dátumok egy kiterjesztett időtartamot határoznak meg a calendar vagy a calendar_dates táblában meghatározotthoz képest, akkor az adatbázis határozottan állítja, hogy nincs szolgáltatás az adott időtartamon belül, ha ez az időtartam nem tartalmazza az aktuális naptári dátumokat.</p>
<p>feed_version</p>	<p>Nem kötelező. Az adatközlő egy szöveges meghatározást fűzhet a GTFS adatbázishoz. A GTFS-t felhasználó alkalmazások ezt az információt jelenítik meg az adatközlőknek annak eldöntésére, hogy az általuk eszközölt utolsó változtatások beépültek-e az alkalmazásokba.</p>

Melléklet 2

Viszonylat	Végállomások	Adatforrás
M1	Mexikói út M / Vörösmarty tér	http://www.bkv.hu/metro/M1_.html
M2	Őrs vezér tere M+H / Déli pályaudvar M	http://www.bkv.hu/metro/M2_.html
M3	Kőbánya-Kispest M / Újpest-központ M	http://www.bkv.hu/metro/M3_.html
4	Újbuda-központ / Széll Kálmán tér M	http://www.bkv.hu/villamos/4.html
6	Móricz Zsigmond körtér / Széll Kálmán tér M	http://www.bkv.hu/villamos/6.html
7	Bosnyák tér / Albertfalva vasútállomás	http://www.bkv.hu/busz/7.html
173	Bornemissza tér vá. / Újpalota, Nyírpalota út	http://www.bkv.hu/busz/173.html
200E	Határ út M / Liszt Ferenc Airport 2	http://www.bkv.hu/busz/200E.html
Railjet	Budapest-Keleti / Wien Westbahnhof	http://www.mav-start.hu/railjet/
912/942	Budapest-Keleti / Sopron	http://elvira.mav-start.hu/
2821	Sopron / Wien Meidling	http://elvira.mav-start.hu/
130	Budapest-Keleti / Bratislava Hlavná Stanica	http://elvira.mav-start.hu/
2513	Bratislava Hlavná Stanica / Wien Hauptbahnhof	http://elvira.mav-start.hu/
U1	Reumannplatz / Leopoldau	http://www.wienerlinien.at/media/download/2013/Linie_U1_89383.pdf
U2	Karlsplatz / Aspernstraße	http://www.wienerlinien.at/media/download/2012/Linie_U2_68801.pdf
U3	Ottakring / Simmering	http://www.wienerlinien.at/media/download/2012/Linie_U3_68803.pdf
U4	Hütteldorf / Heiligenstadt	http://www.wienerlinien.at/media/download/2012/Linie_U4_68802.pdf
U6	Siebenhirten / Floridsdorf	http://www.wienerlinien.at/media/download/2012/Linie_U6_88272.pdf
CAT	Bahnhof Wien Mitte / Flughafen Wien	http://www.cityairporttrain.com/Default.aspx
S7	Rennweg / Flughafen Wien	http://www.oebb.at/
Volán	BUDAPEST, NÉPLIGET AUT. PU. / WIEN, VIB, U3 STATION ERDBERG	http://www.volanbusz.hu/
OS 714	BUD Nemzetközi repülőtér / Flughafen Wien	http://www.austrian.com/