

# SUBUS

---

Hallgató

FEJES SZILVESZTER

Mentor

DR. PINTÉR RÓBERT

Szabadka, 2015

## Tartalomjegyzék

1 Előszó.....	3
2 Probléma .....	4
3 Célok .....	4
4 Megvalósítás.....	5
4.1 Fejlesztőkörnyezet.....	5
4.2 Felhasználói felület.....	6
4.2.1 Menü .....	7
4.2.2 Marker .....	7
4.2.3 Marker rendszer kialakítása .....	8
4.2.4 Útvonal .....	9
4.2.5 Oldalnézet .....	10
5 Adatbázis .....	11
5.1 Érkezési idő kiszámítás .....	11
6 Jövőbeli célok .....	12
7 Összegzés.....	13
8 Felhasznált irodalom .....	14

## 1 Előszó

Szabadka város jól kidolgozott tömegközlekedése az utóbbi időben modern fejlesztéseken esett át. Az egyik ilyen fejlesztés, az információs táblák kihelyezése a nagyobb buszmegállókra, amelyen látható az adott busz várható érkezési ideje.

A projekt célja, hogy a meglévő információt, azaz a buszok érkezési idejét, ne korlátozzuk a buszmegállókra elhelyezett táblákra, hanem okostelefonon keresztül is folyamatosan értesülhessünk a buszok hollétéről, amikor épp nem a buszmegállóban tartózkodunk, hanem kávézóban, étteremben, iskolában, munkahelyen, stb. Ezzel az applikációval lerövidíthetjük a buszmegállókra töltött várakozási időt, vagy a járat lekéséseket is kiküszöbölhetjük. Rendkívül hasznos, a mindennapi tömegközlekedést igénybevevő embereknek.

A kész applikációnak felettebb egyszerű lesz a kezelése, azaz *user friendly* (felhasználóbarát). Ez nagyon szükséges ahhoz, hogy minden korosztály könnyedén jusson hozzá, a szükséges információkhoz, pl: menetrend, érkezési idő, útvonal stb.

Fontosnak tartom továbbá megemlíteni, hogy az alkalmazás részben készült el. Jelenleg is folyamatban van a fejlesztése. Rengeteg munkát igényel ez a kis egyszerű alkalmazás, és hogy minden a lehető leggyorsabban menjen, rendkívül sok módosítással és tervezéssel jár.

## 2 Probléma

A fennálló probléma, az az, hogy a busz érkezési idejét, csak akkor láthatjuk, ha a buszmegállóban tartózkodunk.

Mindamellet, nagyon sok az az idő, amit egy-egy buszmegállóban eltöltünk a buszra várva. Ezt az időt más hasznosabb dolgokra is elhasználhatnánk. A mai világban igen megszokott a rohanó életmód, azonban akinek nincs személygépkocsija, és más egyéb járműje, rá van kényszerülve a tömegközlekedésre. Ha naponta is csak 10 percet álldogálunk egyhelyben a buszmegállón, akkor ez egy hét alatt 70percre jön ki, egy hónapra kivetítve már 280perc, azaz 4óra és 40perc, amit azzal töltünk, hogy várakozunk.

## 3 Célok

A cél az, hogy egy mobil applikáció segítségével, interneten keresztül kapnánk az információt, hogy a busz mikor ér az adott buszmegállóra. Ezzel a rendkívül hasznos adattal, lényegesen lerövidítenénk a várakozási időt.

Rendkívül fontos kiemelni a MOBIL szót. Ma már mindenkinek ott pihen a gyorsan kézbe fogható mini számítógép, és így már bárhol és bármikor hozzájuthatunk a szükséges információkhoz.

## 4 Megvalósítás

### 4.1 Fejlesztőkörnyezet

Az applikáció elkészítésére szolgáló fejlesztőkörnyezet, az Android Studio.

Az Android Studio, egy a Google által fejlesztett és rendkívül támogatott környezet, mobil applikációk számára. Könnyedén hozhatunk létre, akár a kész sablonokból kiválasztott mintára applikációkat.

Az Android Studio felkínálja azt a lehetőséget, hogy használjuk a Google maps könyvtárat, ezzel implementálva a Google térképeket az applikációba. A projekt kifejezetten nagy része a Google mapsra támaszkodik, ezzel a lehetőséggel, a térképet nézve, könnyedén megtalálhatjuk a számunkra fontos buszmegállót.

Az emulátorral ellátott Studio-val, egyszerűen letesztelhetjük az appot, több eszközön is, többféle *hardware* beállítással is. Így jelenleg az API17-el dolgozunk. Ez a *Jelly Bean* egyik verziója, és szinte a legelterjedtebb.

Az Android Studio mellett, szó esett még az Eclipse fejlesztőkörnyezetről, viszont mivel ezt már nem támogatja a Google, gyorsan elvetettük.

Így is akadtak gondjaink. A folyamatos frissítések, összezavarják a programot és az appot is. Folyamatosan napra készen kell állnia a fejlesztőkörnyezetnek, mert hibásan fog működni. Igen zavaró, ha megoldottunk egy problémát, másnapra jön egy frissítés, és máris nem jó az egész. Ezzel sok gondba ütköztünk, de maradtunk ennél a fejlesztőkörnyezetnél.

## 4.2 Felhasználói felület

A legfőbb szempont az egyszerűség. Bár az egyszerű dizájn mögött, igen komoly a mechanizmus, amely az egészet működteti.

Az applikáció elindítását követően, egy indítási felület jelenik meg, ezután tárul elénk a Google maps integráció.

Azonban megjelenik egy ablak is, amelyben kiválaszthatjuk a kívánt busz járatot, pl: 1, 6, 3.

Hogy minél egyszerűbb legyen a buszmegálló keresése, Szabadka térképét vettük alapul.

A térképen lesznek elhelyezve a buszmegállók jelölésére használt markerok, és az őket összekötő útvonal.

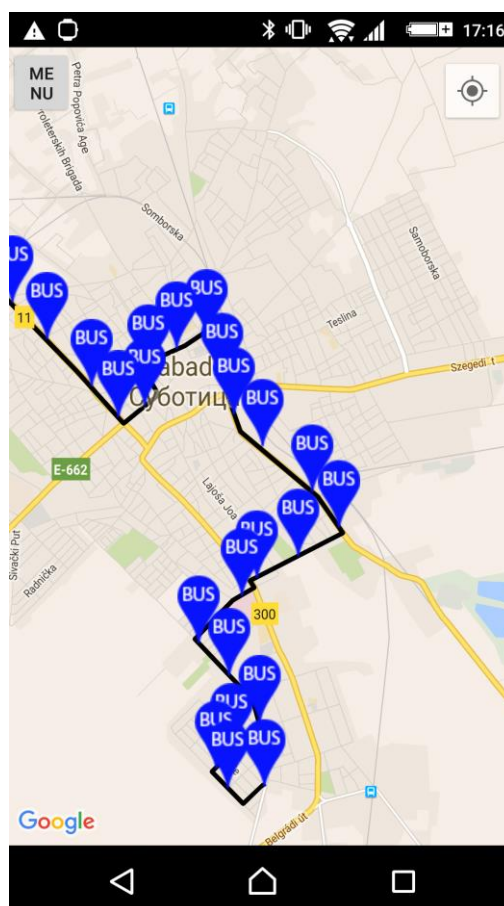
Egy-egy markert megérintve előugrik a buszmegálló neve, és hogy melyik busz mikor érkezik arra a buszmegállóra. Tehát a kezelése rettentően egyszerű.

A képernyő jobb felső sarkában, egy icon jelzi, hogy aktív-e a telefonunkon a GPS. Ha aktív (ekkor kék színű lesz), az icon-t megérintve, a térképen megjelenik a saját pozíciónk.

Bal felső sarokban pedig a Menü gomb található

Amely eddig megvalósítára került:

- buszmegállók
- útvonal
- buszmegállók nevei
- saját pozíció gomb
- menü gomb
- képernyő álló nézet
- képernyő fekvő nézet
- térkép felülnézet (alap)
- térkép oldalnézet



1. ábra: Az alkalmazás kinézete

## 4.2.1 Menü

Az applikáció menü részében, amely iconja a bal felső sarokban kap helyet, egy felület gördül le, a képernyő bal oldalától indulva, és nem egészen a jobb oldaláig. Tehát nem takarja el teljesen az alatta lévő térképet, csak rágördül.

A menü részben helyet kap a:

- *Find bus line* – a kívánt busz járatának kiválasztása
- *Find place* – a kívánt hely kiválasztása (ez egy külön funkció, melyen az applikáció elkészülése után foglalkozunk, de a későbbiekben még lesz róla szó)
- *Schedule* – menetrend
- *Settings* – különböző beállítások, például orientáció.
- *Help* – leírás az applikáció használatáról

## 4.2.2 Marker

A Google maps, markerokat használ bizonyos helyek megjelölésére. Mivel ez egy jól bevált megjelenítő eszköz, mi is ezt használtuk fel. Azonban a magunk formájára alakítottuk kinézetét.

A markeroknak átadjuk a koordinátákat, és a megfelelő helyen odakerül a térképre.

Továbbá a markerokat elláttuk eseményekkel, hogy mi történjen, ha rányomunk egy markerra.

1. Középre pozicionálja saját magát
2. Egy szövegdozban megjelenik a marker neve, azaz a buszmegálló neve
3. Egy szövegdozban megjelenik a buszok érkezési ideje

Egy érintéssel tehát több dolog is elénk tárul.



2.1 ábra (eredeti marker)



2.2 ábra (általunk szerkesztett marker)

### 4.2.3 Marker rendszer kialakítása

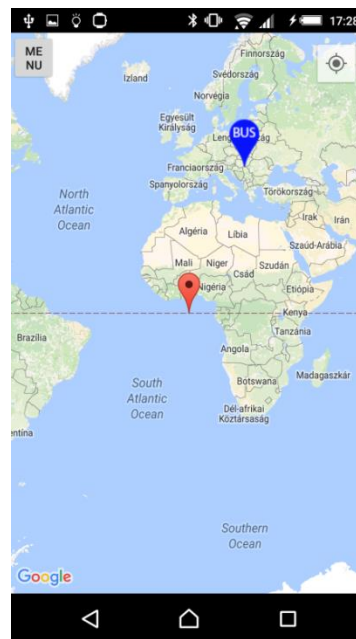
A teljesen alap beállítás a 3.1-es ábrán látható, leszámítva a saját pozíció illetve a menü gombot.

A 3.2-es ábrán már beraktuk a saját markerünket.

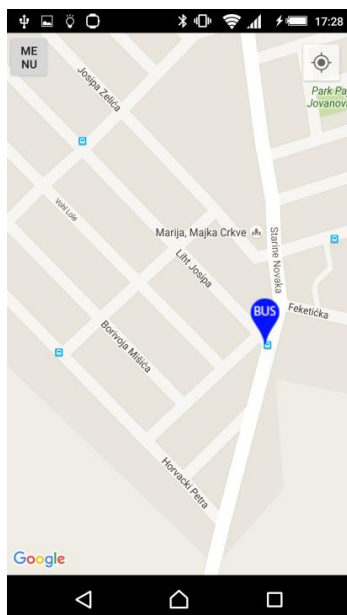
Majd még egy markert beillesztettünk amely a 3.4-es ábrán látható, és ily módon az összes többit is, mely végül kirajzolta nekünk a buszmegállók pozícióját.



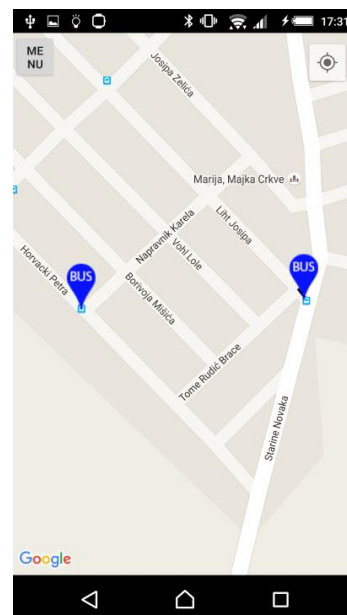
3.1 ábra alap Google maps



3.2 ábra beillesztett saját marker



3.3 ábra beillesztett saját marker



3.4 ábra 2db saját marker közelebbről



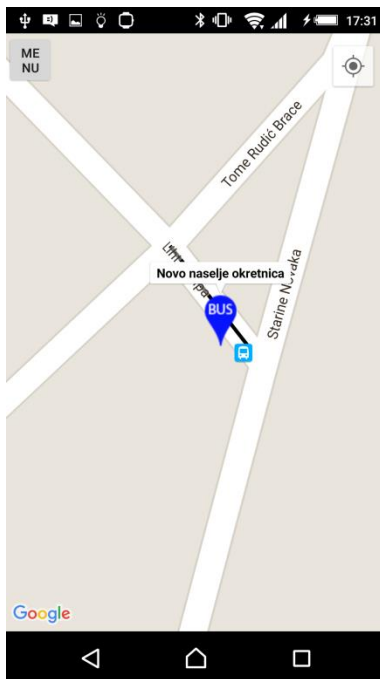
## 4.2.4 Útvonal

A Google mapsban ha megadunk két koordinátát, akkor kirajzolja a két pont közötti legrövidebb útvonalat.

Mi azonban nem használhattuk ezt a megoldást, mivel a buszok nem a legrövidebb úton haladnak, hanem az olyan pontokon, ahol a legtöbb ember gyűlik össze a legnagyobb eséllyel.

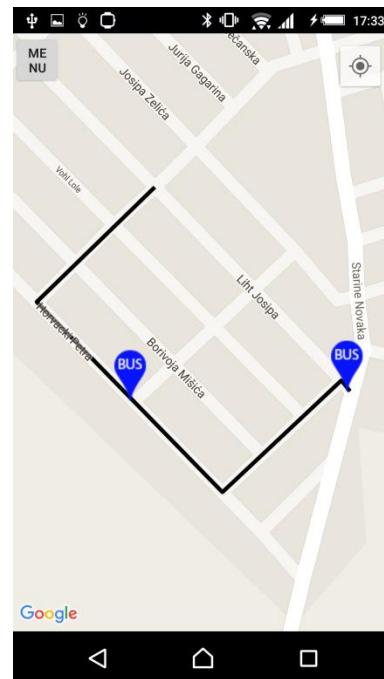
Tehát saját útvonal kirajzolást kell alkalmaznunk. Erre lehetőségünk adódik, ha két koordinátát összekötünk egy egyenes vonallal (4.1 ábra). Ebből kiindulva pontról pontra rajzoljuk ki az útvonalakat (4.2 ábra).

Ezzel a módszerrel megszabadulunk az internet kötelezettségétől, mivel maga az applikáció tárolja a koordinátákat.



4.1 ábra

saját útvonal kirajzolás: egyenes vonal

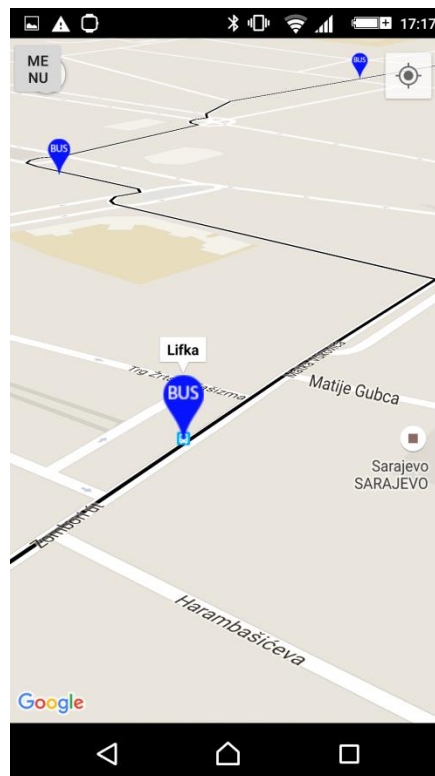


4.2 ábra

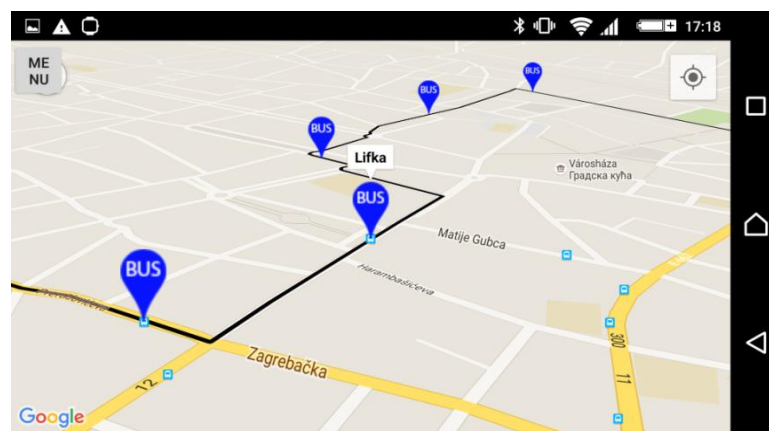
saját útvonal kirajzolása több utcán

## 4.2.5 Oldalnézet

Opcionálisnak tartjuk bemutatni az oldalnézetet, mivel ez lehetővé teszi “távolra látást”, és sokkal szebb megjelenítésre is használható amint az, az 5.1-es és 5.2-es ábrán látható.



5.1 ábra “távolra látást” lehetősége



5.2 ábra “távolra látást” lehetősége közelebbről

## 5 Adatbázis

A Subotica Trans, egyes buszmegállóin elhelyezett táblák, melyeken látható a busz érkezési ideje, egy SQL adatbázisból kapja a szükséges adatokat. A projekt lényegének megvalósításához, nekünk is szükségesek ezek az adatok. Az ő szerverük, feldolgozza azokat az adatokat melyeket a buszoktól kap, egyik ilyen a GPS adatok (és más egyéb), amivel már tudja számolni a várható érkezési időt.

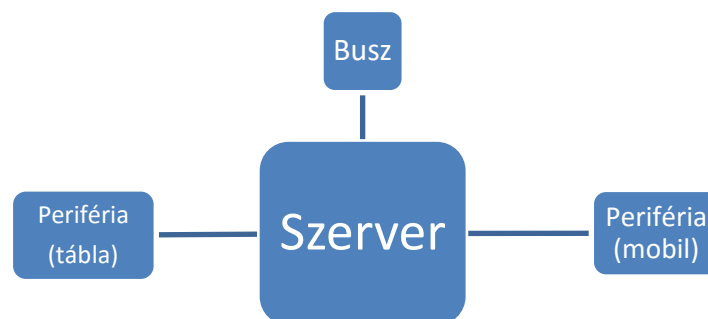
Így felkerestük az adatbázis adminisztrátorukat, aki hozzáférést biztosított az adatokhoz. Ezeket az adatokat csakis az applikációhoz használhatjuk fel, és azon belül is csak a szükségeset. Semmiféle olyan adat nem kerül rossz kezekbe amelyhez titkosított hozzáférés tartozik. Habár mi kaptunk hozzáférést az adatbázishoz, az applikációt szigorúan lekorlátozzuk azokra az adatokra, amelyekhez hozzáférhet.

Hogy a megfelelő marker a megfelelő időpontot kapja, hatalmas “papírmunka” szükséges, és nagy odafigyelés. Ráadásul ismerni kell a PHP nyelvet is, amellyel kommunikálunk a szerverrel.

Így a jelenlegi JAVA nyelvhez, hozzátársul a PHP.

### 5.1 Érkezési idő kiszámítás

Busz-Szerver(adatbázis)-Periféria kapcsolati ábra



6. ábra Sutrans információs rendszer struktúrája

Minden busz el van látva egy GPS és egy GSM modullal, melyek segítségével kommunikál a szerverrel (adatbázissal). A szerver az adott pozíciót, sebességet és más egyéb adatot összeveti a buszmegállónak a pozíciójával, és egy algoritmus segítségével kiszámolja a lehetséges érkezési időt.

Majd ezt az időt egy adatbázisba rendezi, és elküldi a buszmegállón lévő táblának, a mi esetünkben pedig a telefonunknak.

## 6 Jövőbeli célok

A projekt még fejlesztés alatt van, de így úgymond, menet közben felmerültek a következő lehetőségek integrálása is:

1. Az eddigi módszerre alapozva, létrehozni az összes többi markert, és a hozzájuk tartozó útvonalat.
2. A menü részt aktiválni, és lehetővé tenni az útvonalválasztásokat
3. Menetrend betöltése és tárolása
4. Kidolgozni a kommunikációt és a biztonságot az adatbázissal
5. Főbb kinézet javítások, pl: más színű útvonal, stb.
6. *Find place* kidolgozása.
7. A *Find place*, egy lehetőség lenne az applikációban, amely turistáknak, illetve itt élő embereknek, akik nem ismerik annyira a busz útvonalakat, és nem tudják, hogy hogyan lehetne eljutni egyik pontból a másikba, a tömegközlekedést használva.

Következzen egy példa az app alkalmazását illetően:

Palicson sétálunk a tó körül, és el szeretnénk jutni a kelebiai határátkelőre.

Ekkor megnyitjuk az applikációt, és bekapcsolni a saját pozíció lehetőséget.

A térképen megjelenik a saját jelenlegi helyzetünk. Majd a menüben a „*Find place*” részben, kiválasztjuk a „*Kelebia border*” helyet. Ekkor egy olyan algoritmus fut le háttérben, amely meg tudja nekünk mutatni, hogy először a 6-os járatszámú buszra kell felszállni, mely legközelebb ekkor és ekkor indul. Majd a Lifka buszmegállón átszállunk az 1-es járatszámú buszra, ami már eljuttat minket a kelebiai határátkelőre.

Hogyan történik mindez? (fejlesztés alatt álló algoritmus!)

1. A saját pozíciónkat összeveti a hozzánk legközelebb álló buszmegállóval, ami egy bizonyos útvonalhoz tartozik.
2. A cél helyszínt összeveti a hozzá legközelebb álló buszmegállóval, ami egy bizonyos útvonalhoz tartozik.
3. Összeveti a két útvonal találkozási pontját, mivel majd ott fog megtörténni az átszállás.

Ha nincs átszállás, akkor egyszerűen mutat egy lehetséges buszjáratot.

## 7 Összegzés

Az applikáció fő részei tehát:

1. -Google maps
2. -Markerok
3. -Útvonalak
4. -Adatbázis kommunikáció

Az applikáció félkész állapotban van. A hátralévő lépések, már papíron vannak, és jelenleg azok kidolgozásán fáradozunk. Mivel itt nemcsak egy témakörhöz kell érteni, jelentősen megnöveli a fejlesztésre szánt időt, például:

- adatbázis lekérdezés optimalizálása, mert nem szeretnénk leterhelni a meglévő rendszert
- biztonsági szempontok megfelelő alkalmazása, mert mi csak használjuk a Sutrans cég rendszerét
- az alkalmazás megfelelő megjelenítése, különböző paraméterekkel rendelkező, android operációs rendszert futtató készülékeken.

Ami a projekt jövőjét illeti, számítunk arra, hogy Szabadkán és külvárosaiban élő emberek, akik eddig is használták a Subotica Trans tömegközlekedési vállalat szolgáltatását, le fogják tölteni és napi szinten fogják használni a mi mobil alkalmazásunkat.

## 8 Felhasznált irodalom

1. Mike James: Android Adventures - Getting Started With Android Studio

<http://www.i-programmer.info/programming/android/5887-android-adventures-getting-started-with-android-studio.html?start=3>

2. Google Maps Android API

<https://developers.google.com/maps/documentation/android-api/utility/setup>

3. Android Studio Google Maps

<http://stackoverflow.com/search?q=android+studio+google+maps>

4. Ekler P., Fehér M., Forstner B., Kelényi I.:

Android-alapú szoftverfejlesztés - Az Android rendszer programozásának bemutatása,  
SZAK KIADÓ KFT., 2012