

Szegedi Tudományegyetem
Informatikai Tanszékcsoporth

Bevetésirányítás a térinformatikában,
avagy az Sx_Bevetésirányító ismertetése

Diplomamunka

Készítette:

Szikszai Sándor

programtervező matematikus

szakos hallgató

képfeldolgozás szakirány

Témavezető:

Dr. Katona Endre

egyetemi docens

Szeged

2005

Feladatkiírás

A feladat a mentőszervezetek (tűzoltóság, illetve a mentőszolgálat) valós igényeinek megfelelő programrendszer elkészítése. A hallgató felveszi a kapcsolatot a mentőszervezetekkel, majd az elvégzendő feladatról követelmény specifikációt készít, amelyet a témavezetővel egyeztet.

A hallgató az alábbi részfeladatokat kell, hogy elvégezze:

1. A rendszerhez szükséges térképi adatok beszerzése vagy előállítása, lehetőleg vektoros formátumban.
2. A rendszerhez szükséges leíró adatok beszerzése vagy előállítása.
3. A feladathoz SSADM modell kidolgozása.
4. A feladat egyed-kapcsolat modelljének, relációs modelljének, adatfolyam diagramjának kidolgozása.
5. A rendszer programozása Java vagy C++ környezetben, illetve az adatbázisrendszer létrehozása.
6. A rendszer felhasználói és fejlesztői dokumentációjának elkészítése.

Tartalmi összefoglaló

- **Téma megnevezése:**

A mentőszervezetek munkáját segítő bevetésirányító program tervezése és megvalósítása.

- **Megadott feladat megfogalmazása:**

A bevetésirányító rendszer bemutatása, a program lehetőségei a mentőszolgálat, és a tűzoltóság esetén. A térképrendszer bemutatása. A GPS jármű park, a szolgálatvezénylési és egyéb menedzselési programrészek megvalósítása, felhasznált struktúrák ismertetése.

- **Megoldási mód:**

A cél olyan bevetésirányító program elkészítése volt, amely nemcsak adattáblákat kezel, hanem ezeket a táblákat összekapcsolja egy térképrendszerrel, illetve keresőrendszerrel. Fontos a könnyű kezelhetőség, és az átláthatóság, amit a speciálisan erre a rendszerre kifejlesztett színes, és ikonos **táblamegjelenítő** biztosít. A program fontos tulajdonsága az egységesített bevetésirányítás szellemében elkészített **hasonló sémák** alkalmazása, ami lehetővé teszi a program más bevetésirányítási rendszerekre való egyszerű implementálhatóságát. (A programban olyan sémákat alkalmaztam, amely mindkét mentőszervezet esetében felhasználhatók. Pl.: betegfelvétel – helyszín állapot felvétel).

- **Alkalmazott eszközök, módszerek:**

A rendszer Borland C++ Builder fejlesztői környezet alatt íródott. A DBE adatbázis motorhoz kapcsolódik, és Paradox táblákat használ. DXF, illetve BMP térképet kezel. A program vektoros térkép-megjelenítője négyesfában tárolt térképet jelenít meg. A program betegfelvételi, illetve helyszínállapot felvételi része XML fában van tárolva, és a GPS adatszolgáltatás TCP/IP alapú hálózaton valósul meg.

- **Elért eredmények:**

Sikerült számítógépes megoldást találni a mentőszolgálatok bevetésirányítási feladatainak megvalósítására. Ráadásul a programban a vektoros és raszteres térképek együttes kezelése is megvalósult, sőt még egy ál-GPS szerver járműparkját is képes volt megjeleníteni. A felsorolt eszközökön túl a programban nagyon sok apróság is megtalálható, amely nemcsak megkönnyíti, de fel is gyorsítja a bevetésirányító munkáját.

- **Kulcsszavak:**

Bevetésirányító program, térinformatika, vektoros és raszteres térképek összehangolása, GPS járműpark, mentős és tűzoltós szolgálatvezénylés, betegfelvétel, és tünet felvétel, Borland C++ Builder.

Tartalomjegyzék


| | | |
|-------|---|----|
| 1 | Bevezető..... | 5 |
| 2 | A rendszer felépítése..... | 6 |
| 2.1 | Az elképzelt rendszer részei | 6 |
| 2.2 | A megvalósított rendszer leírása..... | 7 |
| 3 | A bevetésirányító-kliens ismertetése | 9 |
| 3.1 | A főmenü ismertetése | 10 |
| 3.2 | Az akció indítása..... | 10 |
| 3.2.1 | Az eset felvételi lap..... | 12 |
| 3.2.2 | A járművek oldal | 16 |
| 3.2.3 | Az Akció / Térkép oldal..... | 24 |
| 3.3 | Menedzsmentek | 25 |
| 3.3.1 | A szolgálatvezénylési tábla..... | 26 |
| 3.3.2 | A helyszín menedzsment | 28 |
| 3.4 | A térkép oldal..... | 29 |
| 4 | A felhasznált struktúrák leírása..... | 34 |
| 4.1 | A program fejlesztésének életútja..... | 34 |
| 4.2 | A térkép technikai ismertetője | 35 |
| 4.3 | A GPS rendszer – GPS szerver-kliens | 36 |
| 4.4 | A négyes fa (quad tree) [1] | 38 |
| 4.5 | A DXF adattípus [1,2] | 40 |
| 4.6 | A helyszín állapota, illetve a betegfelvétel XML beolvasója | 43 |
| 4.7 | A program adattábláinak kapcsolata..... | 44 |
| 4.8 | A program adattábláinak leírása | 46 |
| 4.9 | A program adatfolyam diagramjai (DFD) | 50 |
| 5 | Továbbfejlesztési lehetőségek | 52 |
| 5.1 | A garázkliens támogatás..... | 52 |
| 5.2 | Járműkliens támogatás | 52 |
| 5.3 | Többgépes támogatás..... | 52 |
| 5.4 | Webes fejlesztés..... | 52 |
| 6 | Irodalomjegyzék | 53 |
| 7 | Köszönetnyilvánítás | 55 |

Bevezető

Napjaink rohanó világában egyre több ember él a Földön. Logikusnak tűnik, hogy ezzel a növekedéssel egyenes arányban a segítő-szervezetek is növekedni fognak. A nagyobb, és több szervezet összehangolása komoly feladat. Gondoljunk csak bele, mekkora munka lehet a mentőszolgálat, tűzoltóság, rendőrség, katasztrófavédelem munkájának összehangolása. A napi riasztások hada, a sok jármű személyzetének beosztása, a járművek hollétének nyilvántartása, a bevetés helyszínéhez a leggyorsabb út megtalálása. Ezekre a feladatokra próbáltam számítógépes megoldást találni.

A megvalósított program a mentőszervezetek bevetésirányítási feladatait szeretné gyorsítani. Továbbá szeretném megmutatni, **hogyan kapcsolható össze a bevetésirányítás a térinformatikával, illetve miként lehet a térinformatikai ismereteket mentési munkákban kamatoztatni.** A munka feladata tulajdonképpen egy erős bevetésirányítási adatbázisrendszer térképrendszerrel való összekapcsolása.

A dolgozat a két mentőszervezet, a mentőszolgálat és a tűzoltóság munkáját segítő programot ismerteti, ami képes mindkét üzemmódban dolgozni. Mivel a dolgozatban vannak olyan részek, amelyek csak a mentők, illetve csak a tűzoltók esetén fordulnak

elő, ezek a részek baloldalon a következő jelöléssel vannak ellátva: .

A mentőszervezetek közül bevetésirányítási feladatokat ellátó program a Fővárosi Tűzoltóparancsnokság birtokában megtalálható, de sajnos abban nem található meg a térképtámogatás. Ezt a rendszert csak az egységek bevetésirányítására tervezték.

A mentőszolgálatnál a számítógépes bevetésirányítás nincs megvalósítva. A mentőszolgálat irányító-központjaiban mágnes táblán, vagy papíron követik figyelemmel a mentőautók mozgását. Viszont fejlesztések előbb-utóbb itt is várhatók.

A bevetésirányítás folyamatában az irányító parancsnok, az a vezető, aki az ügyeleket irányítja, akinek jól kell gazdálkodnia azzal az erővel, eszközzel, ami rendelkezésre áll. Ezt az erőt akkor tudja igazából hatékonyan és eredményesen mozgatni, ha a helyét is pontosan meg tudja állapítani az eseménynek, illetve fel tudja mérni, hogy a rendelkezésre álló erőből mennyi szükséges a sikeres akcióhoz, és ezt a kettőt össze tudja hangolni.

A diplomamunkában megvalósított program segítségével percről-percre nyomon lehet követni az eseményeket, illetve azt, hogy laktanyában (vagy mentőállomáson) mely jármű milyen felszereléssel áll készenlétben. Sőt a bevetés során tüzeset, vagy baleset bekövetkezésekor még az adott egységek riasztását is felajánlja.

A program nagyon sok hasznos újonságot, ötletet tartalmaz, amelyek ténylegesen segítik a mentőszolgálatok munkáját, és akár más hasonló fejlesztéseknél is felhasználhatók.

2 A rendszer felépítése

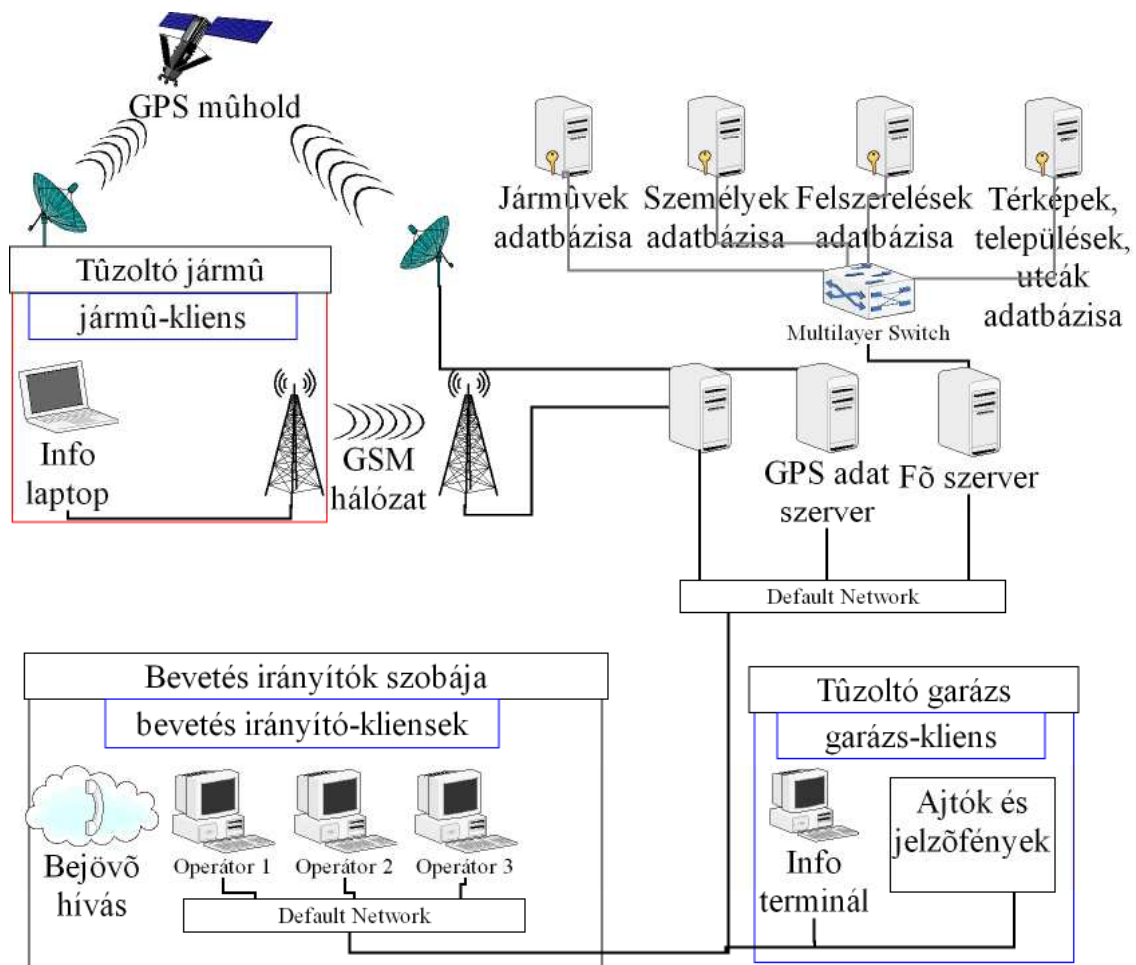
A rendszerleírás 2 részre osztható. Az első rész az elképzelt rendszer, amit pl. későbbi fejlesztések során meg lehetne valósítani, és a második rész a tényleges megvalósítás.

2.1 Az elképzelt rendszer részei

- **1. fő szerver:** Erre a szerverre kapcsolódnak a hírközpontok, ahonnan elérhetik a főbb adatbázisokat. Főbb adatbázis a programban a **személyzeti-**, a **jármű-**, a **szolgálatvezénylési-**, a **napló-**, és az **utcaadatbázis**, melyek általában több adattáblából állnak. A **térképfájlok** is a fő szerverről érhetők el, és a **GPS adatok** is ide futnak be (persze ez lehet másik szerveren is). Ezek az adatbázisok nagyobb rendszer esetén külön szervereken tárolhatók, mivel ezeknek nem kell feltétlenül egy helyiségben lenni. Például a személyadatbázis gépét a személyzeti osztályon lehet tartani, mert a személyzet beosztása, felvétele is itt történik. Amennyiben sok hírközpont lesz, a fő szerver elé több kiszolgáló szervert érdemes telepíteni.
- **2. bevetésirányító-kliensek:** E kliens mellől lehet irányítani a bevetéseket. A kliensek a hírközpontban helyezkednek el. A kliens a fő szerverre kapcsolódva elérheti a térképeket, a főszerver adatbázisait, továbbá a GPS jármű koordinátákat. A hírközpontokhoz lehetne jogosultságokat rendelni, így a bevetésirányító hatásköre személyre szabható lenne. Ezáltal kialakítható lenne **több különböző prioritású hírközpont** is.

- **3. jármű-kliens:** a járművekből elérhető az aktuális riasztás helyszínének útvonalterve, a tűzcsapokkal feltüntetve, az objektum-mentési, oltási terve, vagy éppen az objektum alaprajza, illetve lehetőség nyílik a járművel való kapcsolattartásra is.
- **4. garázs-kliens:** vezérli a kapunyitó áramköröket, a riasztó lámpákat.

Az alábbi képen az elképzelt rendszer felépítése látható:



2.2 A megvalósított rendszer leírása

Az elképzelt rendszer megvalósítása nem fért volna bele a diplomamunka kereteibe, ezért csak a bevetésirányító kliensprogramja lett kidolgozva. Itt a kliens nem a főszerver adatbázisaihoz kapcsolódik, hanem a gépen található helyi adatbázisokhoz. Míg az

elképzelt modell esetén a kliens a főszerverhez kapcsolódnak. Ebben a modellben a bevetésirányítást egy személy végzi.

Az alábbi ábrán a megvalósított rendszer felépítése látható:

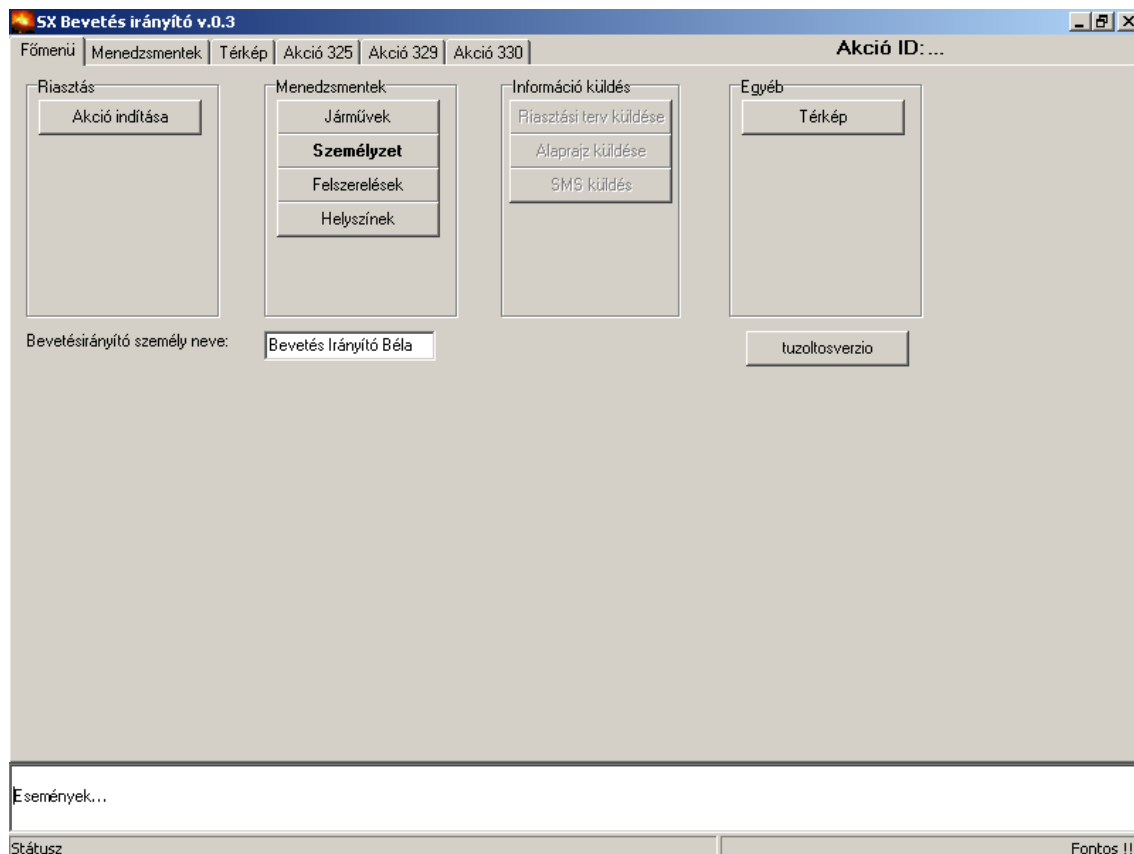


Fontos: A program a mentőszolgálat, valamint a tűzoltóság részére lett fejlesztve, így az mindkét működési formában alkalmazható. A működési mód beállítását a tényleges használat előtt kell megtenni! Az alapértelmezett működési mód a mentős, átkapcsolni a **tűzoltós verzió** gomb segítségével lehet.

Az elkövetkező fejezetben a bevetésirányító-kliens felhasználói dokumentációja tekinthető meg.

3 A bevetésirányító-kliens ismertetése

A program elindítása után a következő képet látjuk:



A képernyőn a főmenü jelenik meg, ahonnan az összes fontosabb menüpontot elérhető.

A program felülete 3 részből áll:

- felső fülek
- fő munkaterület
- eseménysor + státuszsor

A program felső részén a **felső fülek** helyezkednek el ezzel a legfőbb műveletek választhatóak ki, továbbá itt jelennek meg az indított akciók is új fülek formájában. Ez alatt található a **fő munkaterület**, ami a program lényegi része, az események nagy része itt zajlik. Az **eseménysoron** a kiadott műveleteket nézhetjük vissza, ezt külön segíti a **színes kiírás**, így a színek segítségével az események megkülönböztethetők.

3.1 A főmenü ismertetése

A főmenü részei:

- akció indítása
- menedzsmentek
- térkép

Az **akció indítása** gomb segítségével tudunk indítani új riasztást, vagy felvenni utólagos bejelentést. A **menedzsmentek** menüpont alatt a különféle erőforrások szervizelését végezhetjük el, itt adhatunk hozzá új egységet, módosíthatunk a paraméterein, törölhetünk közülük. Megkülönböztetünk jármű-, személy-, szolgálatvezénylési-, felszerelés-, helyszín-, tűzcsap-menedzsmentet is.

Most pedig nézzük meg miként lehet bevetéseket indítani.

3.2 Az akció indítása

Ahhoz, hogy megértsük, hogyan épül fel a riasztás, meg kell ismernünk a hírközpontok működését.

A hírközpontok működése

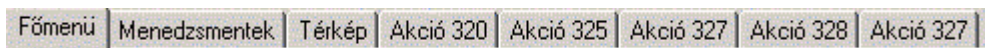
- A hírközpontba befut a hívás. A bejelentő valószínűleg valamilyen segítséget szeretne kapni. Lelkiállapotától függően elmondja, hogy mi történt, mi a baj, hol történt, van-e veszélyben élet, életek. A hírközpontos feladata mérlegeli a problémát, a bejelentő lelkiállapotát is figyelembe véve. Majd felveszi a bejelentő adatait az esetleges ismételt kapcsolatfelvétel miatt. Az adatokat valamilyen adatbázisba célszerű felvenni (papír nyomtatvány, v. számítógépes adatbázis), ugyanis minden riasztást naplózni kell! Ha a bejelentő a sérült, akkor utasításokkal kell ellátni, illetve meg kell nyugtatni, és megelőzni hogy bármilyen „butaságot” tegyen.
- Közben el kell indítania a riasztást. Felmérnie a helyzetet, hogy milyen eszközök szükségesek az akció sikeréhez. Az eszközökön itt járműveket és felszereléseket kell érteni. A járművek helyes megválasztása nagyon fontos. Egy rosszul megválasztott konfiguráció, az akció sikerét befolyásolhatja. A túl kevés lehet, hogy nem lesz elég, bár ez esetben mozgósíthatóak újabb egységek is, de kérdéses hogy ezek időben érkeznek-e meg a helyszínre. A túl sok viszont, az erőforrások pazarlásának tekinthető.

- Végül a járműveket el kell indítani, esetleg újabb járműveket kell bevonni az akcióba.

Tehát a feladat: irányítani és menedzselni a bevetést. Most pedig nézzük meg azt, hogy a program milyen lehetőségeket kínál fel.

Kattintsunk az **akció indítása** gombra.

Ekkor egy új felső fül jelenik meg. Egy adott riasztásra a felső fül kiválasztásával lehet hivatkozni:



A program 3 részoldalt biztosít az akció kezelésére.

A 3 rész második fülsor segítségével választható. Ezen oldalak a következők:

1. esetfelvételi lap – itt történik az adatok felvétele
2. járművek lap – itt a járművek indítását végezhetjük
3. akció / térkép lap – itt pedig az akciót figyelhetjük.

Az **akció indítása menüpont** aktiválásakor a képernyőn az **esetfelvételi lap** jelenik meg.


3.2.1 Az eset felvételi lap

Az oldal részei:


- a. helyszín választása
 - o térkép menü
 - o helyszín állapotának felvétele (tűzoltók esetén)
- b. a bejelentő adatainak felvétele + akció leírása
- c. a betegek felvétele (mentők esetén)

Ezt a lapot a mentőszolgálat papír formában használja. Az oldal felső részén **a bejelentés dátuma**, az **esetfelvételi lap sorszáma**, illetve **a bevetést irányító személy neve** jelenik meg. Nézzük meg az alábbi részeket:

a. Helyszín választása

Az oldal felső részén kiválaszthatunk egy, vagy több helyszínt. Több helyszín kiválasztása betegszállítás, vagy több helyszínen történő riasztás (pl. viharok) esetén indokolt. A helyszínekhez sorszám is tartozik, ami a meglátogatásuk sorrendjét adja meg. A helyszín kiválasztása a térkép menüben történik, ami a „**Térkép**”  gombra kattintással érhető el.

A térkép menü ismertetése a 3.4. fejezetben történik.

A helyszín felvétele után az esetfelvételi lapra visszajutni a „Vissza”  gomb segítségével lehet.



Helyszín állapotfelvétel (tűzoltók esetén)

A program a **tűzoltós verzió** bekapcsolása esetén lehetőséget biztosít a helyszínek állapotadatainak felvételére. A helyszín ebben az esetben baleseti, vagy tüzeseti hely, ahová különböző típusú járműveket kell eljuttatni. Nagy segítség lenne, ha egy program az akcióban résztvevő járműtípusokra ajánlást adna. Ehhez feltétlenül szükséges tudni, hogy mi történt. Ez besorolható valamilyen **esetcsoportha**. Ilyen esetcsoport lehet, pl.: közlekedési baleset, vízi baleset, lakóházi tüzeset, stb.

Az **esetcsoportha jellemző kérdéseket** a bevetésirányító teszi fel. A kérdésre adott válaszokból, meg tudja határozni a riasztás fokát. Ilyen kérdések lehetnek, pl.: lakóházi tüzeset esetében: - milyen anyag ég, - mennyire ég, - mióta ég, - mi okozta.

Az esemény körülményeit rögzíteni kell! Ennek rögzítése az **esetcsoportha-válaszok** bejelölésével történhet. Az előző példánál esetcsoport válaszok lehetnek a következők:

- a tűz jellegére utaló válaszok, pl.: - ég; - lángol; - robban; - parázslík,
- a tűz terjedésére vonatkozó válaszok, pl.: - gyorsan terjed,
- a tűz oka, pl.: - kukatűz; - konyhai olajtűz; - elektromos készülék, kábeltűz.

A helyszínek állapota menü a helyszínre kétszeri kattintással érhető el:



A baloldalon az **esetcsoporthok**, a jobb oldal felső részén a bejelentőtől kérdezhető **esetcsoportha jellemző kérdések**, és a jobb oldal alsó részén az **esetcsoportha-válaszok** találhatók meg. Bármely válasz bejelölése esetén az esetcsoportja oldalán piros karika jelenik meg. Fontos megjegyezni, hogy az egyes válaszok mellé automatikusan rendelhető egy adott járműtípus!

Megjegyzés: A betegszállítás részmodulban is ugyanez a módszer van alkalmazva.

A helyszín állapotadatait a „**Váltás az esetfelvételi lapra**” gombbal menthetjük.

Fontos: a helyszín **nem mentett** állapotadatai elvesznek, másik helyszín kiválasztásakor!

b. A bejelentő adatainak felvétele

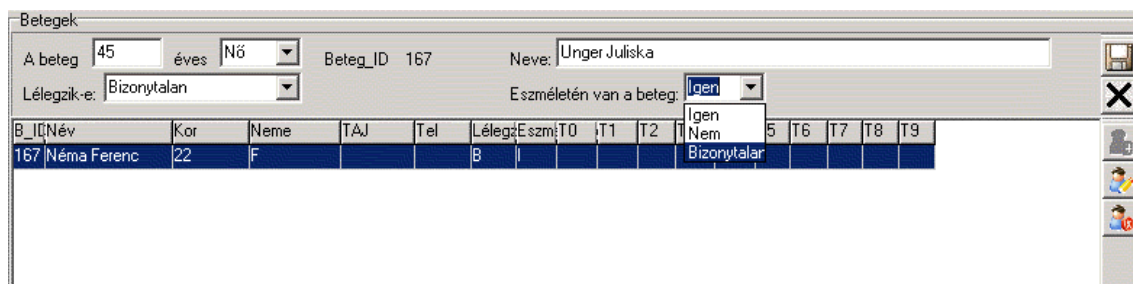
Az oldal második része a **bejelentő adatainak** rögzítésére, illetve az akció leírására, megjegyzések felvitelére szolgál.

Fontos: A bevitt adatokat menteni kell a **mentés** gombbal  !






c. A beteg felvétele (mentős verzió)

A **beteg felvételi lap** csak mentős üzemmódban érhető el, tűzoltók esetén ilyen felvételi lapra nincs szükség.



A betegfelvételi lap segítségével vehetők fel a betegek adatai és tünetei, betegszállítás, illetve mentés esetén. Míg a beteg főbb adatait ezen az oldalon, tüneteit a beteg nevére 2-szeri kattintással vehetjük fel:

| | |
|---|---|
|  Melkasi fájdalom / Szívpanaszok | Légzése változott-e? (szapora, kapkodó, felületes, akadozó) |
|  Fejfájás / Agyrész katasztrófa | Tudati állapota változott-e? (nyugtalan, aluszékony, zavart) |
|  Has / hátfájás | Külleme változott-e? (sápadt, veritékes, elékült, kipirult) |
| | Gyógyszert vett-e be? (saját gyógyszer, élvezeti szer, más) |
| | Tudati állapota változott-e? (nyugtalan, aluszékony, zavart) |
| | Volt-e hasonló panasz miatt kivizsgálása / kórházi kezelése? |
| | <input type="checkbox"/> 01A2 Tudati állapotváltozás / ájulás 'A' |
| | <input type="checkbox"/> 01A3 Sápadt, veritékes, cyanoticus 'A' |
| | <input type="checkbox"/> 01B1 Anamnesisben igazolt szívbetegség 'B' |
| | <input checked="" type="checkbox"/> 01B2 Élezeti szer, kábítószer túlzott fogyasztása 'B' |
| | <input type="checkbox"/> 01B3 35 évnél idősebb ffi., 40 évnél idősebb nő 'B' |
| | <input type="checkbox"/> 01C1 Légzéssel összefüggő fájdalom a fentiek nélkül 'C' |

A tűzoltó helyszín állapothoz hasonlóan, a **tünetcsoportok** a bal oldali ablakban, a **tünetcsoport** aktuális **kérdései** a jobb felső ablakban jelennek meg. Ezeket a kérdéseket kell feltenni a bejelentő személynek, majd a jobb alsó ablakban a **válaszlehetőségeket** bejelölni.

A válasz után egy betű található idézőjelek között, ami a riasztás fokát mutatja. Ez 'A', 'B' vagy 'C' lehet. A válasz bejelölésével a jobb oldali csoportok színe megváltozik. A riasztás foka szerint piros, narancs vagy sárga lesz. A bejelölés hatására rendre roham, eset-, vagy mentőkocsit ajánl fel a program. Az oldal tetején a pedig a válaszok alapján a magasabb riasztás színe jelenik meg. (A fenti példában a narancs, a piros, és a zöld közül a piros fog megjelenni.)

Fontos: a beteg tüneteit „**A váltás az esetfelvételi lapra**” gombbal lehet menteni!

Másik betegre 2-t kattintva annak tünete meg fognak jelenni.

A mentős **betegfelvétel** és a tűzoltós verzió **helyszínek állapot felvétele** nagyon hasonlít egymásra. A különbség az, hogy a **betegfelvétel** esetén a válaszok egy kocsitípust jelölnek meg, **helyszín állapot felvétel** esetén több kocsitípus is kijelölődhet egyszerre.

A későbbi fejlesztés során lehetőség nyílhat további mezők felvételére is.

Miután minden adatot felvettünk, a következő oldalon az akcióban résztvevő járműveket választhatjuk ki.

3.2.2 A járművek oldal

SX Bevetés irányító v.0.3

Főmenü Menedzsmentek Térkép Akció 325 Akció 329 Akció 330 Akció ID: 325

Ételi lap Járművek Akció/Térkép

Jármű lista ☐ csak ilyen típust! Műszak választó: a Csak az akció járműveit mutasd!

| ID | Rendszám | Feltöltési | Státusz | Körzet/UT | Jármű típus | Létszám | Maxhely |
|----|----------|------------|--------------|-----------|----------------------|---------|---------|
| 1 | tuz-001 | | startra kész | 1/1 | 1-es fecskendő | 6 | 6 |
| 4 | tuz-004 | | startra kész | 2/1 | 1-es fecskendő | 1 | 6 |
| 14 | tuz-014 | | startra kész | 2/1 | 1-es fecskendő | 1 | 6 |
| 2 | tuz-002 | | startra kész | 1/2 | 2-es fecskendő | 1 | 4 |
| 11 | tuz-011 | | startra kész | 1/2 | 2-es fecskendő | 1 | 4 |
| 6 | tuz-006 | | startra kész | 2/4 | partalék gépjármű f. | 1 | 6 |
| 7 | tuz-007 | | startra kész | 3/5 | magasból mentő j. | 1 | 6 |
| 21 | men-002 | | akcióban | 1/20 | roham kocs | 4 | 6 |

☒ Felszerelés

| ID | Név | Mennyiség | Készlet |
|----|------------|-----------|---------|
| 4 | injekciósű | 2 | 12 |
| 1 | kotszer | 1 | 20 |
| 6 | vödör | 1 | 23 |

☒ Személyzet

| ID | HelyszNév | Beosztás | Képesítések |
|----|----------------|-----------------|-------------|
| 6 | 1 Hajdu Péter | küszlő százados | |
| 94 | 2 Izmos Gáspár | küszlő | |
| 97 | 3 Lehel Ottó | küszlő | |
| 95 | 4 Léha Tihomér | küszlő | |
| 98 | 5 Lejtő Endre | küszlő százados | |
| 92 | 6 Okos Ernő | küszlő | |

Opciók:

| | |
|-----------------|-----------------|
| Mentés indítása | Betegszállítás |
| Orvosi ügyelet | Gyakorlat |
| Szerviz | Felkészítésre |
| Kiérkezés | Felsz.használat |
| megsz/Visszaind | |
| Visszaérkezés | Bel |

Körzet: ☐ Csak ezt mutasd!

| ID | Körzet |
|----|----------|
| 1 | Belváros |
| 2 | Újszeged |
| 3 | Gyálarét |
| 4 | Dorozsma |
| 5 | Tápe |

Az oldal részei:

- járműlista
- személyzetlista
- felszereléslista
- körzetlista
- opciók
- járműtípus szelektáló
- váltás az akció járművekre
- műszak választó
- riasztási szintjelző

A **járműlista** az összes járművet megjeleníti, amin **legalább 1 személy** tartózkodik. Ennek oka: a személyzet nélküli kocsi nem osztható be bevetésre.

A **jármű adatai** közül annak azonosítóját, rendszámát, készenléti állapotát, státuszát, körzetét és típusát, típusának ikonját, valamint nevét jeleníti meg. A **jármű listájából** megtudható, hogy az adott jármű működéséhez **hány** személy kell (Maxhely), illetve az


adott műszakban **hány személy van** rá beosztva (Létszám). A műszak mezőről bővebben a **Menedzsmentek -> Szolgálatvezénylési tábla** fejezetben lesz szó.

A **jármű állapotok** különböző színekkel vannak jelölve. Ennek nagy jelentősége van, mivel így a járművek sokkal jobban megkülönböztethetők, ezért gyorsabban kiválaszthatók és bevetethetők. Ezek az állapotok következők:

0. garázsban - startra kész
1. garázsban - nincs feltöltve
2. akcióban
3. akció vége, visszaút
4. betegszállítás
5. orvosi ügyelet
6. rendezvény
7. gyakorlat
8. szerviz
9. garázsban - feltöltése folyamatban



A mentőszolgálat bevetései után szükség van a mentőjármű készleteinek utánpótlására, illetve annak takarítására. Ezen állapotok nyilvántartása a **jármű készenléti állapot** mezőben történik. A járműveknek 2 készenléti állapota van:

- feltöltött állapot
- nincs feltöltve, vagy készletfogyóban (jelölés: )

A készenléti állapot használható tűzoltóság esetén is. Pl.: a tartálykocsi feltöltöttségének jelzésére.

A **körzet listával** egy adott körzet járművei jeleníthetők meg. Ez nagyon hasznos opció, ha bevetés közben a helyszínhez legközelebbi állomás járműveit kell elindítani.

A járműtípus-lista

Járművek: ☐ csak ilyen típust! Műszak választó: a

| | ID | Rendszám | Feltő | Státusz |
|---|----|----------|-------|---------------|
| <input type="checkbox"/> roham kocsi | 1 | tuz-001 | | startra kész! |
| <input type="checkbox"/> eset kocsi | 2 | tuz-002 | | startra kész! |
| <input type="checkbox"/> mentő kocsi | 11 | tuz-011 | | startra kész! |
| <input type="checkbox"/> szállító kocsi | 21 | men-002 | | startra kész! |
| <input type="checkbox"/> MOK szakorvosi k. | | | | |
| <input type="checkbox"/> ONE orvos nélküli k. | | | | |

A **járműtípus-lista** megjeleníthető a „**Járművek**” címkére kattintásával. A **járműtípus** szerinti megjelenítése a „**Csak ilyen típust!**” jelölőnégyzet bekapcsolásával érhető el.

A **járműlistájából** egy járművet kiválasztva, megmutatja a **jármű személyzetét**, illetve **felszerelését**. A személyzetlista megjeleníti a személy nevét, beosztását, illetve a **képesítéseit** is. A személyzetlista külön érdekessége a **helyszám**.



A helyszám jelentősége

A helyszámnak a tűzoltóságnál, főleg a szolgálatbeosztás végrehajtásánál van jelentősége, mivel egy adott helyszámhoz más-más feladat tartozik. **Járműtípusok** szerint külön jelentése van mindegyik helyszámnak.

Tűzoltóság esetén az alábbi járművek választhatók:



Ezek az ikonok sorrendjében a következők:

- 1-es fecskendő
- 2-es fecskendő
- 3-as fecskendő
- tartalékfecskendő
- magasból mentő gépjármű
- műszaki kocsi
- vízszállító
- halonnal oltó gépjármű
- porral oltó gépjármű
- tűzoltó hajó

A következő táblázatban az egyes helyek jelentését ábrázoltam:

| helyszám | 1-es fecskendő | 2-es fecskendő | 3-as fecskendő |
|----------|------------------|-----------------|----------------|
| 0 | parancsnok | parancsnok | parancsnok |
| 1 | 1.fecsk. 1.ember | 1.fecsk 1.ember | |
| 2 | 1.fecsk. 2.ember | 1.fecsk 2.ember | |
| 3 | 2.fecsk 1.ember | | |
| 4 | 2.fecsk 2.ember | | |
| 5 | sofőr | sofőr | sofőr |


Fontos megemlíteni azt is, hogy még a fecskendő tartó személyek helye sem cserélhető fel. Az első személy feladata a cső irányítása. Ennek a személynek kell szembenézni a tűzzel, ezért ide jóval tapasztaltabb és bátrabb tűzoltót osztanak be. A második személy feladata a cső tartása, erre a helyre újonc, vagy nem annyira tapasztalt személyt, esetleg veterán tűzoltót osztanak. A jármű akkor számít 1 fecskendősnak, ha megtalálható rajta az 1. és a 2. személy, és pontosan a 1.2. helyekre vannak beosztva. Az 1-es fecskendőre 2 fecskendőtartó személyzet, a 2-es fecskendőre 1 fecskendőtartó személyzet, a 3-as fecskendőre 0 fecskendőtartó szükséges. Persze szükség esetén a 2-es és 3-as fecskendőn is lehet 2 fecskendőtartó személyzet.

Képesítések

Mind a mentős, mind a tűzoltós helyek elfoglalásához a beosztott személyeknek különböző **képesítésekkel** kell rendelkezniük.




Képesítések, és feltételek tűzoltók esetén:

a. tűzoltó (jele: )


feltétel:

- érettségi,
- tűzoltó alkalmassági vizsga;

b. sofőr (jele: )

feltétel:

- tűzoltó,
- érvényes gépkocsi vezetői engedély,
- szivattyúkezelési engedély (az oltásnál a sofőr kezeli a jármű szivattyú berendezését),

c. parancsnok (jele: )

feltétel:

- tűzoltó
- szolgálatvezénylési engedély
- parancsnoki vizsga

d. darukezelő (jele: )

feltétel:

- tűzoltó
- darukezelői jogosítvány

e. ...

A jármű csak akkor vethető be, ha a szükséges személyzettel rendelkezik. Ennek fontossága a jármű más városba, körzetbe való riasztása esetén nyilvánul meg. Ezért a körzetben lévő főparancsnoknak nem kell tudnia a körzetbe beriasztott járművek személyzeti listáját – mivel feltételezhető, hogy azokon teljes személyzet található, így ő már a jármű típusáról meg tudja mondani, hogy milyen személyzet van rajta.



A mentők esetében nincs megkötés a helyszámok jelentésére. Viszont saját meggyőződésből megkötöttem az utolsó helyszámot, ide ebben az esetben is a sofőrnek kell kerülni.


A mentők esetében is **a járművek típusa** meghatározó. Ebből tudni lehet, hogy a járművön milyen minimális személyzet van.

Itt a következő járműtípusok lehetnek:




1. rohamkocsi
2. eset kocsi
3. mentőkocsi
4. szállító
5. MOK szakorvosi kocsi
6. ONE orvos nélküli kocsi


Képesítések, és megkötések mentők esetén:


a. orvos (jele: )

Megkötés: - nem lehet sofőr

b. mentőtiszt (jele: )

c. szakápoló (jele: )

d. ápoló (jele: )

e. sofőr (jele: )

Megkötés: -nem lehet orvos

A járműtípusok esetén, a jármű minimális személyzete:

1. rohamkocsi: a személyzet 1 tagja mindig orvos
2. eset kocsi: 1 személy legalább mentőtiszt vagy szakápoló
3. mentőkocsi: 1 személy ápoló vagy szakápoló
4. szállítókocsi: orvosi személyzet nem szükséges

A személyek beosztása a **Menedzsmentek** -> **Szolgálatvezénylési tábla** segítségével történik.



Riasztási szint (tűzoltók esetén)

Ezen nézetekben (tűzoltós üzemmód esetén) a program tűzoltós riasztási szint megjelenítésére is képes. A járművek számától függően a következő szintek lehetségesek:

5 - ös szintű készültség: 5 db 1-es fecskendő és 4 db 2-es fecskendő

4 - es szintű készültség: 4 db 1-es fecskendő és 3 db 2-es fecskendő

3 - as szintű készültség: 3 db 1-es fecskendő és 2 db 2-es fecskendő

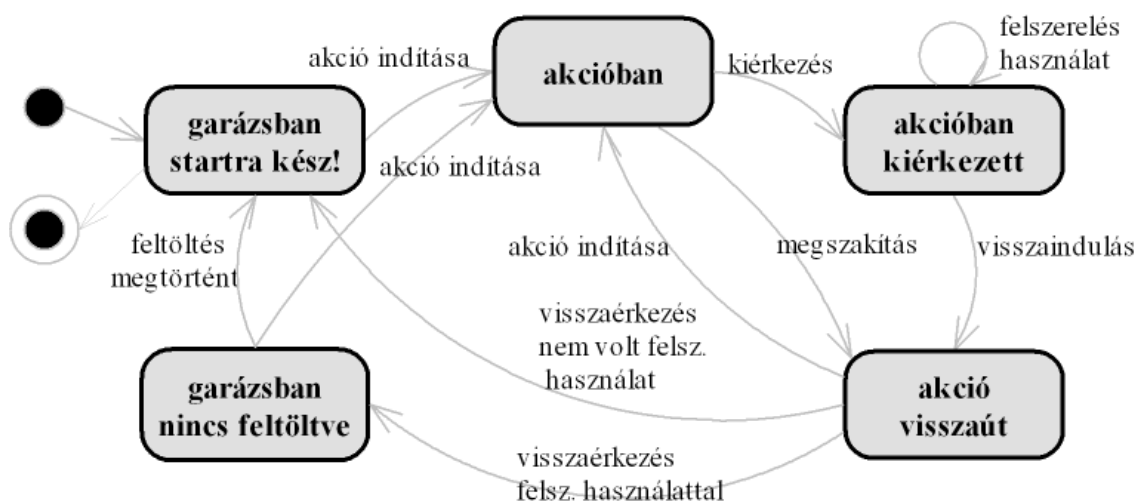
2 - es szintű készültség: 2 db 1-es fecskendő és 1 db 2-es fecskendő

1 - es szintű készültség: 1 db 1-es fecskendő vagy 1 db 2-es fecskendő

Beszélhetünk a különleges gépjárművekről is, ilyen, pl.: a műszaki mentő, daru, porral-oltó gépjármű. Ekkor a készültség kap még egy **K** jelölést is, ami a bevetésben résztvevő különleges gépjárművek jelenlétét jelöli. Az ilyen járműveket „különleges szereknek” nevezik.

A járművek indítási lehetőségei

A járművek a következő életcikluson mennek végig:



A program megkülönböztet 6 **akció indítási típust**:

- akció (mentés)
- betegszállítás
- orvosi ügyelet, tűzoltók esetén rendezvény
- gyakorlat
- szerviz
- feltöltésre küldés

Minden indítási típus az opciók menüből érhető el.

Akció közben lehetőség nyílik **parancsok** kiadására. Ezek a következők:

- jármű indítása (akció, betegszállítás, gyakorlat, orvosi ügyelet, stb...)
- jármű kiérkezés / jármű út megszakítás (visszahívás) időpontjának rögzítése
- jármű állomásra visszaérkezés időpontjának rögzítése
- felszerelés használat

e. térkép műveletek

- egységek kiválasztása (járművek, tűzcsapok, objektumok)
- távolságok meghatározása

f. bevetés befejezése/megszakítása

További (tervezett) funkciók: (Mivel a rendszer jármű-kliense nem lett megvalósítva, így ezek a funkciók sem lettek a programba építve.)

g. jármű-klienssel kapcsolatos műveletek:

- kliens üzenetküldés,
- útvonalterv küldése,
- tűzcsaptérkép küldése,
- riasztási, oltási terv küldése,
- objektum alaprajz küldése.

Parancskiadás csak két esetben lehetséges:

- „**Csak az akció járműveit mutasd!**” jelölőnégyzet bejelölésével, vagy az
- „**Akcio / Térkép**” oldalra lapozással.

Fontos: Az akció parancsok más esetben nem érhetők el!

Ezekben a nézetekben csak a bevetés járművei láthatók.

3.2.3 Az Akció / Térkép oldal

The screenshot shows the 'SX Bevetés irányító v.0.3' application. The top navigation bar includes 'Főmenü', 'Menedzsmentek', 'Térkép', 'Akció 325', and 'Akció 330'. The 'Akció ID: 325' is displayed on the right. Below the navigation bar, there are tabs for 'Esetfelvételi lap', 'Járművek', and 'Akció/Térkép'. The main map area shows a street grid with several red car icons representing vehicles, each with a label like 'J_ID = 7 30 : 287'. A table at the bottom lists vehicle details. On the right, there's a sidebar with buttons for 'Akció részletei' and 'Befejezés'.

| ID | Rendszer | Felki | Státusz | Körzet | Jármű | Indulás | Kiérkezés | Visszaérkezés | Megs. |
|----|----------|--------------|---------|--------|-------|-----------------------|----------------------|----------------------|-------|
| 1 | tuz-001 | startra kész | 1/1 | | | 12005.04.08. 19:26:57 | | | |
| 9 | tuz-009 | akcióban | 1/1 | | | 12005.04.16. 14:45:59 | | | |
| 12 | tuz-012 | startra kész | 1/1 | | | 12005.04.16. 14:46:36 | | 2005.04.16. 14:46:37 | |
| 13 | tuz-013 | startra kész | 1/1 | | | 12005.04.15. 22:06:59 | 2005.04.15. 22:07:17 | 2005.04.15. 22:07:18 | |

Az oldal részei:

- a. térkép megjelenítő
- b. opciók
- c. az akció járművei

A térkép megjelenítő

A GPS adatszolgáltatás révén az akciójárművek koordinátái és sebesség adatai leolvashatók az alsó táblázatról, vagy figyelemmel kísérhetők a térképen. Ezért lehetőség nyílik az akció helyszínéhez legközelebb lévő jármű riasztására.

A riasztási parancsok ezen az oldalon is kiadhatók. A térkép funkcióit „A térkép” fejezetben ismertetem.

3.3 *Menedzsmentek*

Menedzsmenten egy adott egységtípus mozgásainak nyomon követését, illetve annak számítógépes naplózását értjük. Ide tartozik:

- a járművek felszerelésének beosztása,
- felszerelések törlése, új felszerelések regisztrálása, felszerelés adatok módosítása,
- adott járműre kerülő személyek beosztása, regisztrálása
- az utca, tűzcsap, helyszín adattáblák adatainak karbantartására.

A felsorolás utolsó pontja nem feltétlenül a bevetésirányító központból történik.

Erre több fajta elképzelés is született. Pl.: a személy menedzsment – portaszolgálat, objektum menedzsment – ügyfélszolgálat, tűzcsap menedzsment – vízművek, utca menedzsment – önkormányzat v. Informatikai osztály.

A menedzsmentek közül az alábbiak kerülnek ismertetésre:

1. szolgálatvezénylési tábla
2. helyszín menedzsment

A többi menedzsment működése hasonló a fenti menedzsmentekhez, ezért ezek ismertetésére nem térek ki.

3.3.1 A szolgálatvezénylési tábla

SX Bevetés irányító v.0.3

Főmenü Menedzsmentek Térkép Akció 325 Akció 329 Akció 330 Akció ID: 330

Járművek Szolgálatvezénylési tábla Felszerelések Helyszínek

| ID | Név | Beosztás | 1.személy | 2.személy | 3.személy | 4.személy | 5.személy | Sofőr | Tel | Cím |
|----|-------------------|----------|-----------|-----------|-----------|-----------|-----------|-------|------------|----------------------------|
| 1 | Szakszai Sándor | | | | | | | | 321423-232 | Szeged, , 1. |
| 2 | Suweges Gyurka2 | | | | | | | | 564564564 | Szeged, , 2. |
| 3 | Nagy Gyozo3 | | 55 | | | | | | 5667234432 | Szeged, , 15. |
| 4 | Kis Erno | | 5 | | | | | | 3243242332 | Szeged, , 4. |
| 5 | Dancs Brigi | | 2 | 1 | | | | | 7363286439 | Szeged, , 5. |
| 6 | Hajdu Péter | | | | 1 | | | | 9875623238 | Szeged, , 5. |
| 7 | Kiss Csaba | | | | | 1 | | | 4875237843 | Szeged, , 7. |
| 8 | Kovács Márton | | | | | | 1 | | 4397764397 | Csongrád, , 8. |
| 9 | Németh OKOS Gábor | | | | | | | | 4311257671 | Csongrád, , 9. |
| 10 | Emuka | | | | | | | 1 | 2342322342 | |
| 11 | Keserő Béla | | | | | 1 | 1 | | 3476764239 | Csongrád, , 11. |
| 12 | Keserő Eszter | | 1 | | | | | | 9652398727 | Csongrád, Budapesti út, 5. |
| 13 | Barta Gergő | | | | | | | | 1114554156 | Hódmezővásárhely, , 13. |

Műszak: a c,

| Járművek | 1.személy | 2.személy | 3.személy | 4.személy | 5.személy | Sofőr |
|----------|-----------------|---------------|-----------------|----------------|--------------|-------|
| 1/1 | 94 Izmos Gáspár | 97 Lehel Ottó | 95 Léha Tihamér | 98 Lejtő Endre | 92 Okos Ernő | |
| 1/1 | | | | | | |
| 9/1 | | | | | | |
| 12/1 | | | | | | |
| 13/1 | | | | | | |

Események...
[2005.04.15. 22:02:17] 330 .akció : J_ID:2rsz:tuz-002st:startra készl ;1/2 2-es fecskendő létszam:1 maxhely:4 felszhaszn:False jármű elindult a helyszínrre .

Státusz Fontos !!!

Az oldal részei:

- személylista
- szolgálatvezénylési tábla
- műszak választó

A személylista

A **személylistában** az összes személy adata megtalálható. A program nyilvántartja a személy azonosítóját, nevét, beosztását, képesítéseit, telefonszámot, címét. Tűzoltók esetén még azt is, hogy a személy melyik posztra hányszor volt beosztva. (Ennek szolgálatbeosztásnál lesz szerepe). A személy lakcímét a térkép oldalon választhatjuk ki a cím mezőre kattintással. A cím választás hasonló módon történik az esetfelvétel helyszín kiválasztásával. (Ebben az esetben a helyszín címét menteni nem szükséges.)

A szolgálatvezénylési tábla

A szolgálatvezénylési tábla bal oldalán a járművek főbb adatai láthatók. Ez a mező a **jármű mező**, ami 2 sorból áll. Az 1. sorban a jármű körzet és a jármű típus / - jellel elválasztva, illetve a **járműfeltétel-karaktersora**.

Ennek mérete mentő esetén 5 karakter, és tűzoltók esetén 3 karakter.

Mentők esetén az 5 karakter jelentése:

1 – sofőr jelenléte; 2 – orvosok száma; 3 – mentőtisztek száma; 4 – szakápolók száma;
5 – ápolók száma

| | | | | | | | |
|------------|-------|-------------------|--|--|--|--|----------------|
| 2/20 | S0100 | 20 Hollósi Botond | | | | | 42 Magnum |
| 24 men-005 | | | | | | | |
| 2/21 | S0100 | 100 Intel Pál | | | | | 43 Bud Spencer |
| 25 men-006 | | | | | | | |

Pl.: S0020 – a járművön van sofőr és két szakápoló.

Tűzoltók esetén a 3 karakter jelentése:

1 – Sofőr jelenléte; 2 – Parancsnok jelentése; 3 – Fecskendők száma

| | | | | | | | |
|------------|-----|---------------|----------------|-----------------|------------------|----------------|----------------|
| 1/1 | SP2 | 5 Dancs Brigi | 97 Lehel Ottó | 95 Léha Tihamér | 12 Keserű Eszter | 98 Lejtő Endre | 92 Okos Ernő |
| 13 tuz-013 | | | | | | | |
| 1/2 | SP1 | 6 Hajdu Péter | 96 Csöves Géza | 94 Izmos Gáspár | | | 93 Eszes Péter |
| 2 tuz-002 | | | | | | | |

Pl.: S_1 – a járművön van sofőr és 1 fecskendőtartó személyzet, de nincs parancsnok.

A jármű mező után a jármű hely-, székszámai következnek. A helyszámra személyt beosztani, mozgatni a személylistából a személyt áthúzával lehet. Törlés a DEL billentyű lenyomásával történik.

A szolgálatvezénylési tábla műveletei mindig az aktuális műszakra vonatkoznak. Legördülő menü segítségével a **műszak** megváltoztatható. A műszak betűjelét beírva az adott műszak szolgálatvezénylési táblája aktiválódik.

A feltétel-kalkuláló

A napi szolgálatbeosztás megkönnyítése érdekében, a program el lett látva a helyszámok figyelését végző feltétel-kalkuláló programrészrel.

A programrész azt figyeli, hogy az adott jármű megfelelő személyzettel van-e ellátva. Ezeket a paramétereket a képzettségek mezőből olvassa ki. A jármű e tulajdonságait a jármű mező háttérszínének beállításával éri el.

Mentők esetében figyelmeztet, ha:

- a járművön nincs, vagy nem megfelelő képzettségű orvos van (a jelölés színe: piros)
- a járművön nincs sofőr (a jelölés színe: szürke)

Tűzoltók esetében:

- a járművön nincs parancsnok képzettségű személy (a jelölés színe: piros)
- a járművön nincs sofőr (a jelölés színe: szürke)
- a járművön nincsen elegendő számú fecskendőtartó személyzet (a jelölés színe: sárga)

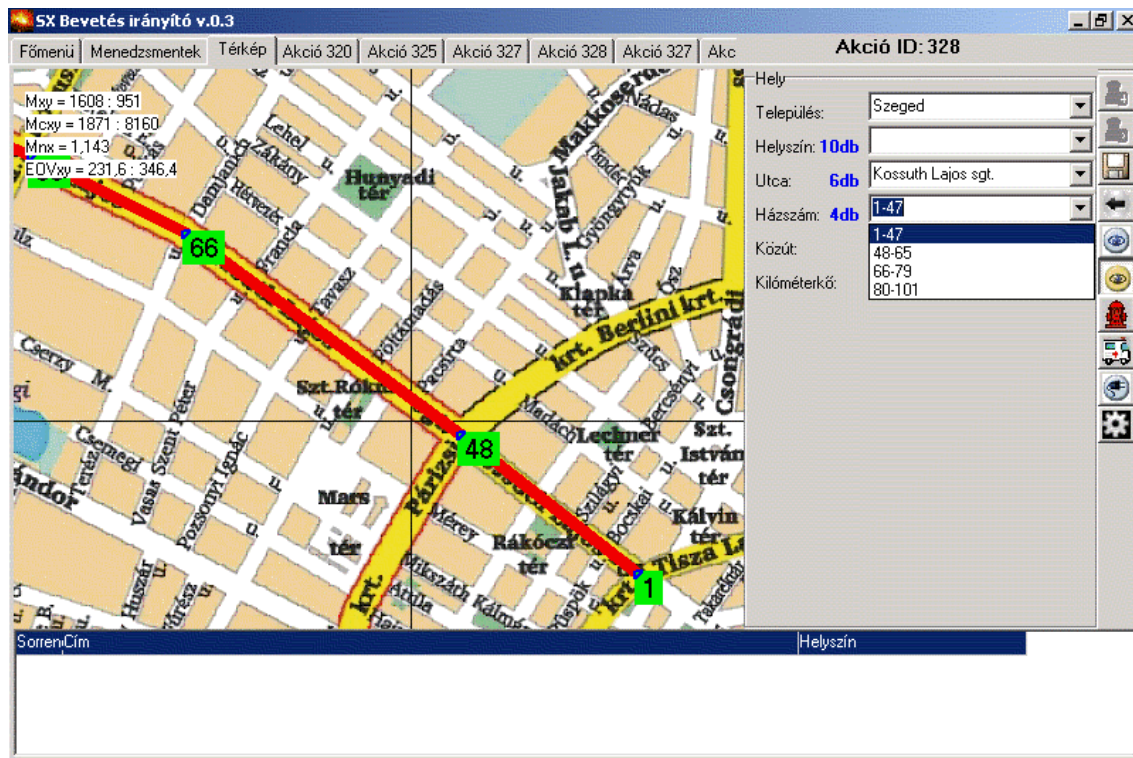
Megjegyzés: 1-es fecskendő esetén 2, 2-es fecskendő esetén 1, 3-as fecskendő esetén 0 csőtartó személyzet szükséges.

Megjegyzés: Ha a járművön egyszerre többféle hiány lép fel, akkor a magasabb prioritású színe fog látszani.

3.3.2 A helyszín menedzsment

Ezen az oldalon vehetünk fel új helyszíneket, illetve módosíthatjuk adataikat. Meg lehet adni a helyszín nevét, címét, típusát, illetve fontos megjegyzéseket is fűzhetünk a helyszínhez. A helyszín címének kiválasztása a térkép oldalon történik, amit a címre kattintással érhetünk el. A **helyszíntípus** mező meghatározza az objektum típusát. Ez lehet például: oktatási intézmény, bank mezőgazdasági telep, irodaház, kórház. Ezeket össze lehetne kapcsolni a helyszínállapot felvételi résszel, így egyes beállítások már a helyszín felvételekor kiválasztódnának.

3.4 A térkép oldal

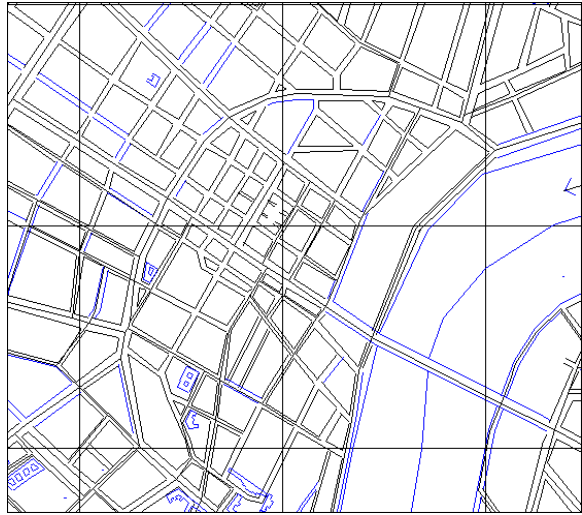
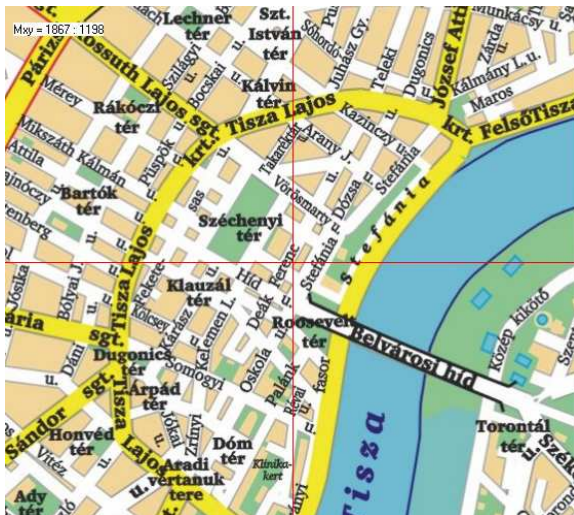




Az oldal részei:

- térkép megjelenítő
- kereső
- opciók

A térkép megjelenítő

A program térkép megjelenítője mozgatható, skálázható. A térkép több adattáblával is össze van kapcsolva. Kapcsolat van a város, utca, hátszám, helyszín, tűzcsap adattáblával. Sőt még a GPS járműkliensek helyzete is megjelenik a térképen. Az összes helykeresési művelet ezen oldalon történik. Erre az oldalra ugrás általában a cím mezőre, vagy a „Térkép” ➡ gombra kattintással érhető el. Az eredeti oldalra pedig a „Vissza” ⬅ gombbal juthatunk.



A térkép megjelenítő két fajta térképpel rendelkezik. Az egyik típus a **raszteres térkép**, ami a város BMP fájlja egy részletének megjelenítésén alapul. A raszteres térkép megjeleníthető a  gomb lenyomásával. A másik típus a **vektoros térkép**, itt a térképadatok a város DXF fájlban vannak eltárolva. A DXF térkép a  gomb lenyomásakor jelenik meg. A két térkép együttes kezelése is lehetséges.

A kereső

A **kereső** egy nagyon hatékony utcakereső rendszer, amely képes a város helyszíneit, tűzcsapjait, illetve az utcák házszámait megkeresni. A keresőt az alábbi módon használjuk:

A település kiválasztása után a többi mező aktualizálódik. Ekkor az utca és helyszín mező előtt megjelenik a **találatok száma**. A település helyszínei, és az utcái ezután elérhetők az adott lenyíló menük használatával. Az utca bevitele után, annak vonala rögtön megjelenik a térképen, sőt az utca házszámai is aktualizálódnak a lenyíló menüben, ezért nem létező helyszínt nem tudunk bevenni. Ez az opció nagyon hasznos, mert megelőzi „fals riasztások” kiadását. Példa: Kalmár József u. -> 1-26 27-33 34-56.


Helyszín bevitel esetén az adott helyszín jelenik meg a térképen.




A program el van látva „**betűszűkítő**” opcióval, ami minden egyes bevitt betű után szűkíti az eredményt. Pl.: kal -> Kalmár József u.

Amennyiben közúton történt baleset, a közút nevét az utca mezőbe kell bevinni, majd a kilométerkő számát a házszám mezőbe írhatjuk.

Tipp: A helyszín mezővel ismert helyszíneket lehet kikeresni. Pl.: Szeged Pláza, Szegedi Önkormányzat, Károlyi Mihály Kollégium, SZTE Gazdasági épület, Posta.



A helyszín kiválasztása után az utca és a házszám mezők automatikusan kitöltődnek.

Fontos: Esetfelvétel → Helyszín felvétele esetén több helyszín is felvehető, ezért a **mentés** gombot  feltétlenül meg kell nyomni!

Új helyszín felvétele a , helyszín megváltoztatása a , helyszín törlésére a  gombbal történik. A felvett helyszínek között az alsó lista segítségével válthatunk, ezek ilyenkor ismételten megjelennek a térképen.



A tűzcsapok megjelenítése

A tűzcsapok megjelenítése a  gombbal történik. Ilyekor a kép alján tűzcsaplista jelenik meg. Erre kattintva az aktuális tűzcsap kék kijelölő négyzetet kap. A tűzcsapok típusuk szerint megkülönböztethetők a térképen: 

Ezek jelentése: -föld feletti; - föld feletti rossz; - föld alatti; - föld alatti rossz tűzcsap.

A GPS jármű megjelenítő

SX Bevetés irányító v.0.3 Folkestra - Fehér 3:29

Főmenü | Menedzsmentek | Térkép | Akció 325 | Akció 330 **Akció ID: 325**

Esetfelvételi lap | Járművek | Akció/Térkép

| LID | Rendszám | Felállítás | Státusz | Körzet | Jármű | Indulás | Kiérkezés | Visszaérkezés | Megj. |
|-----|----------|--------------|---------|--------|-------|----------------------|----------------------|----------------------|-------|
| 1 | tuz-001 | startra kész | 1/1 | | | 2005.04.08. 19:26:57 | | | |
| 9 | tuz-009 | akcióban | 1/1 | | | 2005.04.16. 14:45:59 | | | |
| 12 | tuz-012 | startra kész | 1/1 | | | 2005.04.16. 14:46:36 | | 2005.04.16. 14:46:37 | |
| 13 | tuz-013 | startra kész | 1/1 | | | 2005.04.15. 22:06:59 | 2005.04.15. 22:07:17 | 2005.04.15. 22:07:18 | |

A GPS járművek helyzete a gombbal jeleníthető meg. A megjelenítés feltétele a GPS szerverre való kapcsolódás, amit a gombbal lehet megtenni.


A térképen a jármű ikonja, azonosító száma, pozíciója van feltüntetve. A többi adat a térkép alján lévő listában található.

Az opciók közül a beállítások gombra kattintva a beállítások menü segédprogramjai hívhatók elő.

A beállítások menü segédprogramjai


1. Utcák felvétele menü (jele:)



Ez a segédprogram új utcavonalak felvételére alkalmas. A térképen 2 pont kijelölésével vehető fel az új utcarész (a kijelölés a képen a kék vonal). Az utca és a város mezőket ki kell tölteni, majd megadni az utcaszakasz kezdő és végső házszámát. A program érdekessége, hogy az utcaszámozást automatikusan növeli, ezért az utcaszakasz végső házszámát kell csak beírni. Egy-egy szakasz a mentés gombbal  vehető fel. Az új utcaszakasz az utca adatbázisba kerül.



Fontos: Kijelölés közben a SHIFT billentyűt le kell nyomni!

2. EOVS koordináta beállítás menü (jele:)

A vektoros DXF térkép EOVS koordinátarendszerben van megadva. A DXF egyes pontjai össze kell kötni a raszteres térkép pontjaival. A program pixel és EOVS koordináták között konvertálást végez. A két térkép szinkronizálását itt végezhetjük el. Ehhez a BMP térkép valamely két pontjához tartozó DXF - beli megfelelőjét kell megadni. A két képpontot a SHIFT billentyű lenyomásával lehet bevinni. A mentés gombbal  rögzíthető a két koordinátarendszer egymáshoz való viszonya.

3. GPS tesztautók felvétele menü (jele:)

Az elképzelt rendszerben a program igazi GPS szerverhez kapcsolódik. Az elképzelt rendszer mintájára egy ál-GPS szerver lett elkészítve, ami hasonlóan jármű

koordinátákat küld, csak itt a járművek pályadatait nem GPS műhold szolgáltatja, hanem egy-egy szöveges állomány. Ilyen pálya előállítására képes ez a programrész. Lehetőség van a tesztautók mozgásának felvételére (jele: ), illetve annak lejátszására (jele: )

A felvétel gomb lenyomását követően 5 másodperc áll rendelkezésre a kezdőkoordinátára való beállításra, és csak ezután kezdődik meg a felvétel. Ezt a felvétel gomb villogása is jelzi. A felvétel leállítását a felvétलगomb újbóli megnyomásával tehetjük meg. A felvétel végén a jármű pályája az események ablakban megtekinthető. Az itt felvett pálya a GPS szerver járműfájljai közé beilleszthető.

4 A felhasznált struktúrák leírása

4.1 A program fejlesztésének életútja

Az Sx_Bevetésirányító első verzióit JAVA-ban írtam, ahol MYSQL adatbázist használtam az adattáblák kezelésére. JAVA-ban való fejlesztés mellett a JAVA platform függetlensége, az egyszerű adatbázis kezelése, illetve a nagyszerű Eclipse fejlesztőkörnyezet használata miatt döntöttem. A vektoros térképező rendszer teljes egésze kezdetektől fogva megtalálható volt a programban. Később a program fejlesztése Borland C++ Builder 5 - ös verziójában folytatódott. Ennek oka: a Builder nagyon egyszerű felületkezelése, mivel nagyon speciális táblázatmegjelenítési formát szerettem volna létrehozni (színes, ikonokkal teli, könnyen paraméterezhető, hatékony adatbázis kapcsolat az adattábla és megjelenítés között). A raszteres térképkezelés is csak ebben a verzióban található meg. A C++ fejlesztés másik oka; a program gyorsítása volt.

| | JAVA-s rendszer : | Borland C++ Builder-es rendszer : |
|---------------------|-------------------|-----------------------------------|
| Adatbázis kezelés: | JDBC - MYSQL | Borland BDE |
| Térkép kezelés: | DXF | BMP, DXF |
| Felület kezelés: | Java Swing | Borland VCL könyvtárak |
| Fejlesztőkörnyezet: | Eclipse | Borland C++ Builder 5 |
| Hálózati kapcsolat: | TCP/IP | TCP/IP |

4.2 A térkép technikai ismertetője

A térkép-megjelenítő két fajta térképpel rendelkezik. Az egyik megoldás a **raszteres képtárolás**, aminek a kezelése a város BMP fájl egy részletének megjelenítésén alapul. Itt a térkép pontjai pixelekkal vannak leírva, sajnos ebben az esetben a térképet a memóriában kell tárolni, ami nagyon sok helyet igényel, és skálázás esetén a kép minősége egyre rosszabb lesz. (Ez persze javítható interpoláció használatával.)

A bitkép memória igénye csökkenthető, ha csak a megjelenített részt tároljuk a memóriában. A többi részt csak akkor olvassuk be a merevlemezről, ha szükséges, ekkor viszont a betöltéseknél várni kell.

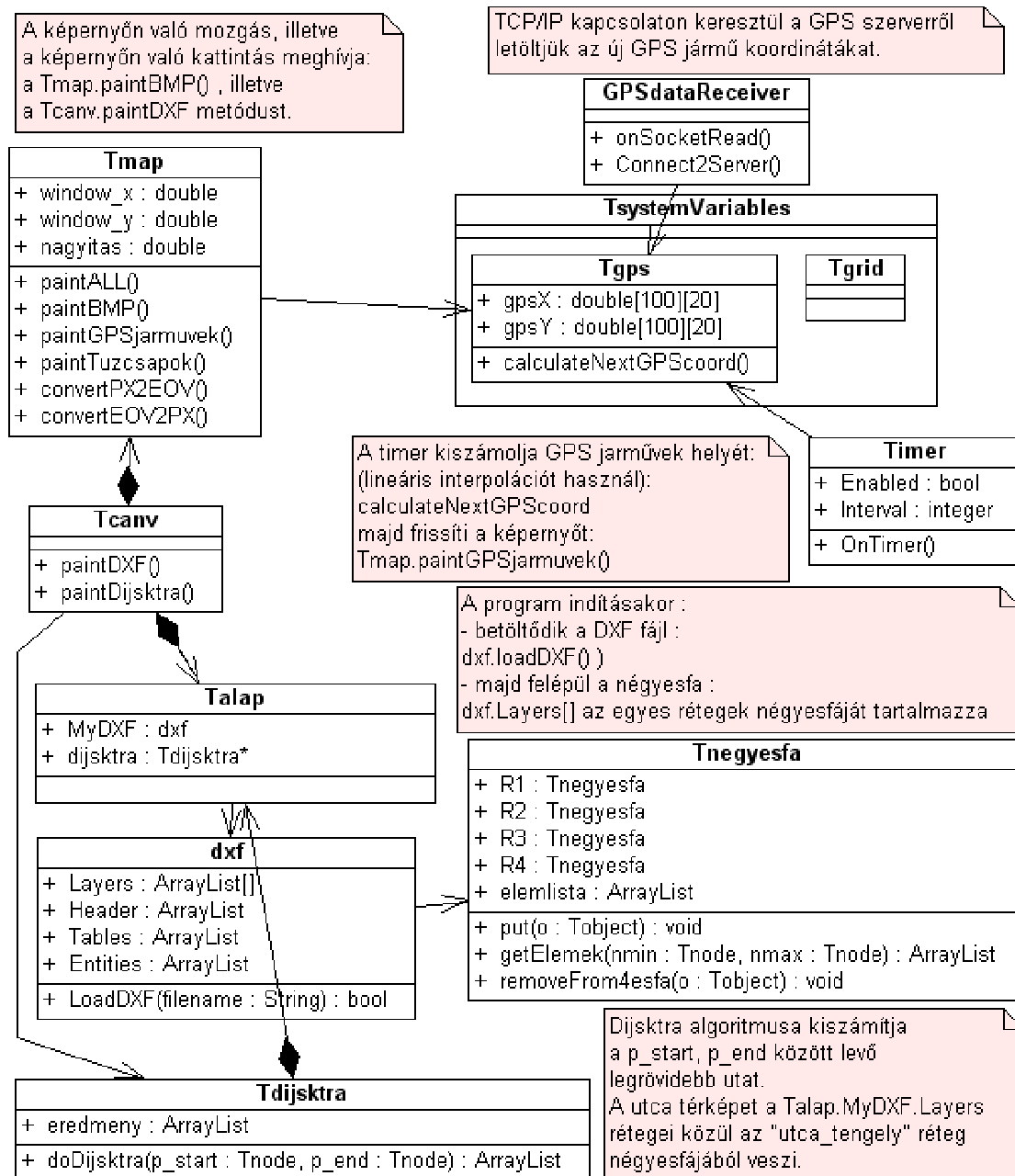
A másik megoldás a **vektoros képtárolás**. Itt a város térképadatai a DXF fájlban vannak tárolva. A DXF-ben a térképadatok pontok, vonalak, poligonok formájában vannak leírva. A térképadatok koordinátái EOVS koordinátarendszerben vannak felsorolva. A vektoros térkép jóval kisebb, mint a bitképes, viszont ennek az adatainak meghatározása jóval számításigényesebb, mivel minden egyes mozgás, skálázás esetén a pontok helyét újra kell számolni. A DXF megjelenítése gyorsítható vektoros koordináták alapján létrehozott memóriában tárolt bitkép használatával. Sajnos ebben az esetben is nagy lesz a memória felhasználás. A vektoros megjelenítés gyorsítására kiváló módszer a **négyesfa** alkalmazása. Ebben az esetben a számításigény lerövidül, mivel itt a kép csak egy bizonyos szeletét kell megjeleníteni. A megjelenítés során a megjelenítendő rész négyesfáját kell bejárni és kirajzolni. A négyesfáról bővebben a „Négyesfa” fejezetben írok.

A DXF térkép rétegekből áll. Egy rétege az utcák tengelyvonalát tartalmazza, aminek segítségével a program képes Dijkstra útkereső algoritmus használatával a térkép két pontja között a legrövidebb utat meghatározni.

A két térkép megjelenítésekor kiszámítjuk a megjelenítendő rész kezdő koordinátáit, majd meghívjuk a képernyő kirajzolását végző metódusokat. A BMP-DXF megjelenítésnél pixel - EOVS koordináta konverziót kell végrehajtani.

A két térkép-réteg felett a GPS járműveket jelenítjük meg. Meghatározott időben frissítjük a járművek helyzetét, ezt a feladatot egy időzítő végzi. A GPS megjelenítésről a „GPS rendszer” fejezetben tesztek említést.

Most pedig nézzük meg, a megjelenítés modulok felépítését és kapcsolatát:



(Az ábrán beszédesebb nevek vannak, és nem mindenhol a programbeli megnevezés, így könnyebb megérteni a térkép-megjelenítő működését.)

4.3 A GPS rendszer – GPS szerver-kliens

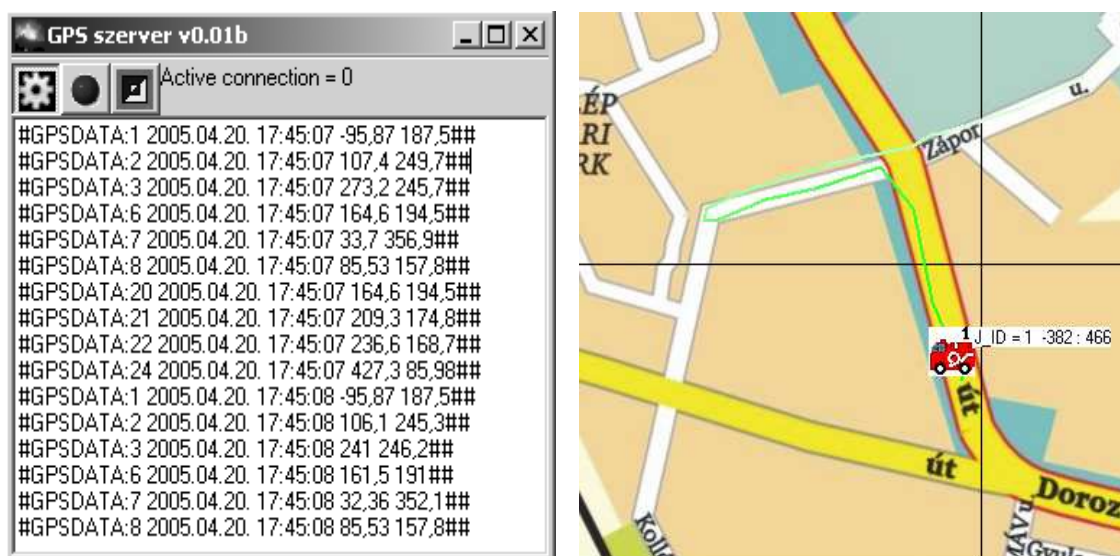
A GPS rendszer ismertetése

A GPS rendszer a Föld körüli pályára állított NAVSTAR (NAVigation Satellite Timing And Ranging) műholdakon alapszik. A holdakat eredetileg katonai célokra alkalmazták, de később polgári felhasználók is igénybe vehették szolgáltatásaikat. A Föld körüli

pályán 20000 km távolságban keringő műholdak állandóan egy speciális kódot sugároznak ki, amely az adott hold azonosítóját tartalmazza. A G.P.S. vevő veszi a műhold jelét, és a vett információ alapján meghatározza a műhold távolságát, ilyen távolságokból pedig meghatározható a vevő földrajzi helyzete.

A kapcsolat megvalósítása

Sajnos nem volt lehetőségem igazi GPS vevőt kipróbálni, ezért az [1], illetve az Interneten lévő anyagokból kerestem választ a kérdéseimre. Elkészítettem saját GPS szerveremet, ami képes GPS adatokat szolgáltatni. A program egy egyszerű szerver, ami a rá kapcsolódó klienseknek küld járműkoordinátákat, amit több szöveges állományból olvas ki (gpsdata.txt). A fájl felépítése nagyon egyszerű, soronként vannak felsorolva az X, és Y koordináták. A szerver koordinátákon kívül az elküldés idejét is rögzíti. Az első képen a szerver program, míg a második képen a kliens GPS megjelenítője látható:



A kliens bejelentkezés után folyamatosan kapja a járművek EOVS koordináta adatait. Szeretném felhívni a figyelmet arra, hogy a példa DXF térkép EOVS koordinátái fiktív adatokat tartalmaznak! A program természetesen képes valós adatokkal is dolgozni. A szerver másodpercenként küld friss adatokat. Viszont a kliens járműpark helyzete 0.1 másodpercenként frissül. A köztes állapotokat interpolációval számítjuk ki. A kliens GPS adatainak tárolása sor adattípust használtam fel. Amennyiben a megjelenítés késne a sorba történő írás szünetel. A járművek nyomvonala megjeleníthető a járművek táblába való kattintással (lásd a 2. képen a vonal). Duplakattintás esetén a térkép középpontjában jelenik meg a jármű.

A hálózati kapcsolat megvalósítására két módszert lehet alkalmazni:

TCP protokoll alkalmazása

Csak a kapcsolat felépítése után indítható az adatszolgáltatás. A kapcsolat felépítése relatív sok időt vesz igénybe, ezért azt nem célszerű bontani minden egyes csomag elküldése után. A TCP protokoll alkalmazásával megbízható adatcsatornát kapunk, és a csomagok elvesztése esetén kapunk értesítést.

UDP protokoll alkalmazása

Ebben az esetben a kapcsolatot nem kell felépíteni, az adatcsomagok közvetlen szórással küldhetők. Viszont nincs garantálva, hogy megkapjuk a csomagot, de akkor sem történik nagy baj, ha véletlenül egy-két GPS jármű koordináta nem érkezik meg, a program ugyanis a jármű koordinátákat interpolációval számítja.

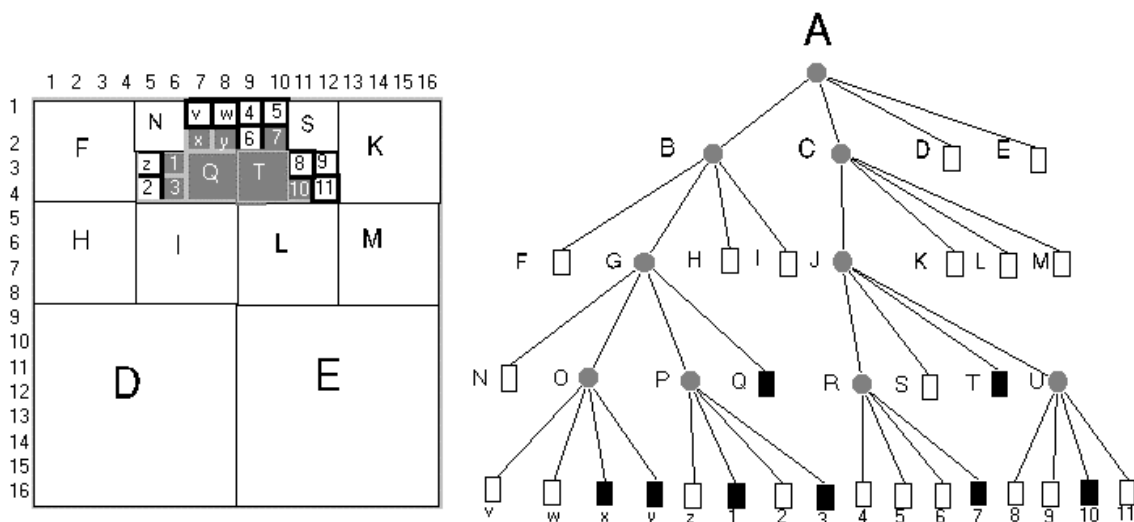
Habár a 2. módszert célszerűbb alkalmazni, a programban az 1. módszert alkalmaztam biztonsági szempontok miatt. Illetve WAN hálózatok esetén az UDP-s kapcsolat nem mindig biztosítható.

4.4 A négyes fa (quad tree) [1]

Algoritmus:

A képet négy részre osztjuk a vízszintes és függőleges felezővonalak mentén, majd ezeket a negyedeket is tovább negyedeljük. Így egy fastruktúra keletkezik, ami leírja a teljes képet, a negyedek pedig a kép egyes szegmenseit alkotják.

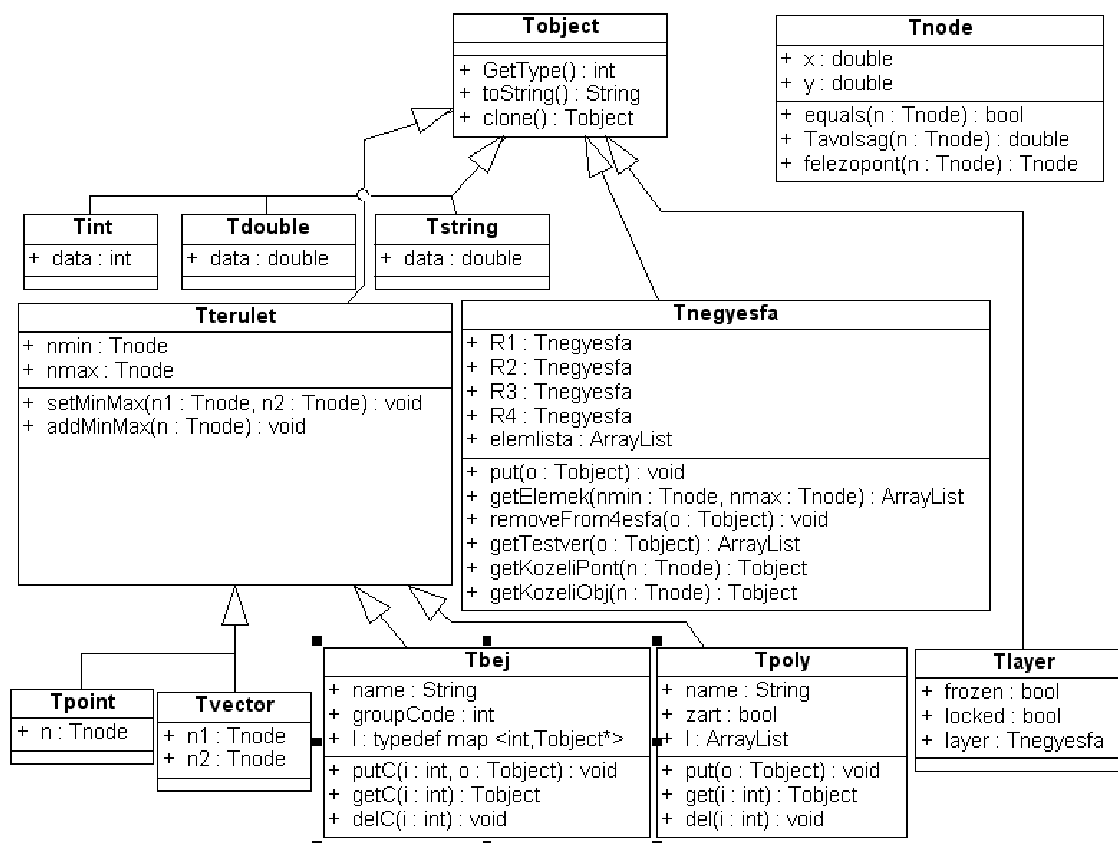
Minden rajzelemet a négyesfa **pontosan egy** szögpontjához rendeljük, és abba a szögpontba kerül, amelyhez tartozó szegmensbe az őt befoglaló téglalap teljes egészében belefér, de annak egyik negyedébe sem fér már bele.



Adatstruktúra:

```
class Tnegyesfa {
    Tnegyesfa *P1
    Tnegyesfa *P2
    Tnegyesfa *P3
    Tnegyesfa *P4
    ArrayList elemlista;
};
```

A négyesfa elemlistája Tobject típusú elemeket tartalmaz. A Tobject osztályból sokféle osztály származtatható. A GetType() metódus a benne levő osztály típusát határozza meg. A clone() metódus másolatot készít az osztályról. A toString() pedig szöveges információkat ad az osztály adatairól. Öröklődés segítségével az alábbi osztályok helyezhetők el a négyesfa elemlistájába:



Különleges figyelmet érdemel a Tbej (Tbejegyzés osztály) ami DXF egyedeinek felépítését próbálja megformálni. (Lásd: **DXF felépítése** fejezet) A Tbej egy olyan osztály, amiben egy map adatszerkezet található. Egy csoportkódhoz hozzárendel egy bármilyen Tobject-ből származó osztályt. (Megjegyzés: a DXF ENTITIES egyedeihez csoportkódok segítségével kapcsolódnak a változók. Lásd: „DXF felépítése”)

A Tbej felépítése:

```
class Tbej : public Tterulet {
public:
    String name;                //bejegyzes neve
    int groupCode;              //DXF fajl 1 bejegyzesenek csoportkodja
    typedef map <int,Tobject*> l; //albejegyzesek listaja

//lekerdez csoportkod alapján
Tobject *getC(int gc);
//beszur egy megadott
//csoportkoddal egy Objektumot

    void putC(int gc, Tobject *o);
    void delC(int gc);
    virtual String toString();
    virtual int GetType();
    virtual Tobject clone();
};
```

Példa:

LINE egyed esetén:

| csoportkód | jelentés |
|------------|-------------------------|
| 10 | a vonal X1 koordinátája |
| 11 | a vonal X2 koordinátája |
| 20 | a vonal Y1 koordinátája |
| 21 | a vonal Y2 koordinátája |
| 5 | egyed azonosítója |
| 6 | vonaltípus |
| 7 | betűtípus |
| 8 | réteg (layer) neve |

| LINE | |
|-------------|-----------------------------|
| name="LINE" | |
| groupCode=0 | |
| | groupCode = 10 |
| | double data = 14.567; |
| | groupCode = 20 |
| | double data = 64.231; |
| | groupCode = 20 |
| | double data = 64.231; |
| | groupCode = 21 |
| | double data = 25.233; |
| | groupCode = 5 |
| | String data = "345H"; |
| | groupCode = 6 |
| | String data = "CONTINUOUS"; |
| | groupCode = 7 |
| | String data = "STANDARD"; |
| | groupCode = 8 |
| | String data = "UTCA_LAYER"; |

4.5 A DXF adattípus [1,2]

A DXF az AutoDesk cég által specifikált vektoros adatcsere formátum (DXF = Drawing eXchange File format). A programban az AC1009 verziójú DXF fájl beolvasása van megvalósítva.

Az érték definiálás 2 sorból áll:

- csoportkód
- érték

Fontosabb csoportkódok:

- 0: szöveg: szekció, egyed azonosítására
- 1: szöveg: elsődleges szöveg mező
- 2: szöveg: szekciónév, blokknév, attribútum címke, stb.
- **5: szöveg: az egyed azonosítója (hexadecimális számjegyek)**
- 6: szöveg: vonaltípus név (pl. CONTINUOUS)
- 7: szöveg: betűtípus
- **8: szöveg: layer (réteg) neve**
- 9: változónév azonosító (header-ben használatos)
- 10, 11, ..., 18: X koordináta
- 20, 21, ..., 28: Y koordináta
- 30, 31, ..., 38: Z koordináta
- 40, 41, ..., 48: lebegőpontos érték

A DXF fájl, az alábbi szekciókból áll:

- HEADER: változók beállítása
- TABLES: vonaltípus, réteg, stb.
- BLOCKS: blokk definíciók
- ENTITIES: rajzelemek

Ezen kívül még egyéb szekciók is léteznek, de azokat nem lettek használva.

A Header szekció:

\$ACADVER 1: AutoCAD verziószám, pl: AC1009
\$EXTMIN 10,20,30: rajzterjedelem bal alsó sarokpontja
\$EXTMAX 10,20,30: rajzterjedelem jobb felső sarokpontja
\$LIMMIN 10,20: rajzhatár bal alsó sarokpontja
\$LIMMAX 10,20: rajzhatár jobb felső sarokpontja
\$TEXTSIZE 40: alapértelmezett szövegmagasság

A Tables szekció:

LAYER: 2(layer neve), 70(flag-ek: a layer fagyott, lock-olt, stb...), 6(vonal típus)

Az Entities szekció:

LINE 10, 20, 30 (kezdőpont), 11, 21, 31 (végpont)

CIRCLE 10, 20, 30 (középpont), 40 (sugár)

TEXT 10, 20, 30 (beillesztési pont), 40 (magasság), 1 (szöveg)

POLYLINE (töröttvonal kezdete) 70(flag: zárt vonal)

VERTEX (töréspont) 10, 20, 30

SEQUEND (polyline vége)

DXF fájl felépítése:

| | |
|----------|--------------------------|
| 0 | Header szekció kezdete |
| SECTION | |
| 2 | |
| HEADER | |
| ... | |
| 0 | |
| ENDSEC | Header szekció vége |
| 0 | Tables szekció kezdete |
| SECTION | |
| 2 | |
| TABLES | |
| ... | |
| 0 | |
| ENDSEC | Tables szekció vége |
| 0 | Blocks szekció kezdete |
| SECTION | |
| 2 | |
| BLOCKS | |
| ... | |
| 0 | |
| ENDSEC | Blocks szekció vége |
| 0 | Entities szekció kezdete |
| SECTION | |
| 2 | |
| ENTITIES | |
| ... | |
| 0 | |
| ENDSEC | Entities szekció kezdete |
| 0 | |
| EOF | Fájl vége |

4.6 A helyszín állapotfelvétel, illetve a betegfelvétel XML beolvasója

Mindkét programrész egy XML fájl alapján építi fel a faszerkezetét. Mindkét esetben 2 mélységű fát használ, melynek első szintjén az esetcsoportok, míg a 2. szinten az adott esetcsoport kérdései, illetve válaszai vannak

Az xml-fa felépítése:

`<m></m>` - ebben a blokkban a betegfelvételi fa,

`<t></t>` - ebben pedig a tűzoltóság helyszín állapot felvételéhez szükséges fa található.

A blokkokon belül helyezkednek el az esetcsoport blokkok:

```
<panasz name="esetcsoport 1"> ... </panasz>
<panasz name="esetcsoport 2"> ... </panasz>
...
```

A panasz blokkon belül vannak az esetcsoport kérdései:

```
<kerdes name="kerdes 1"/>
<kerdes name="kerdes 2"/>
<kerdes name="kerdes 3"/>
...
```

Illetve az esetcsoport válaszai:

```
<valasz name="csoportkód1 válasznév1 'riasztási szint1'" akcio="akciókód1"/>
<valasz name="csoportkód2 válasznév2 'riasztási szint2'" akcio="akciókód2"/>
<valasz name="csoportkód3 válasznév3 'riasztási szint3'" akcio="akciókód3"/>
...
```

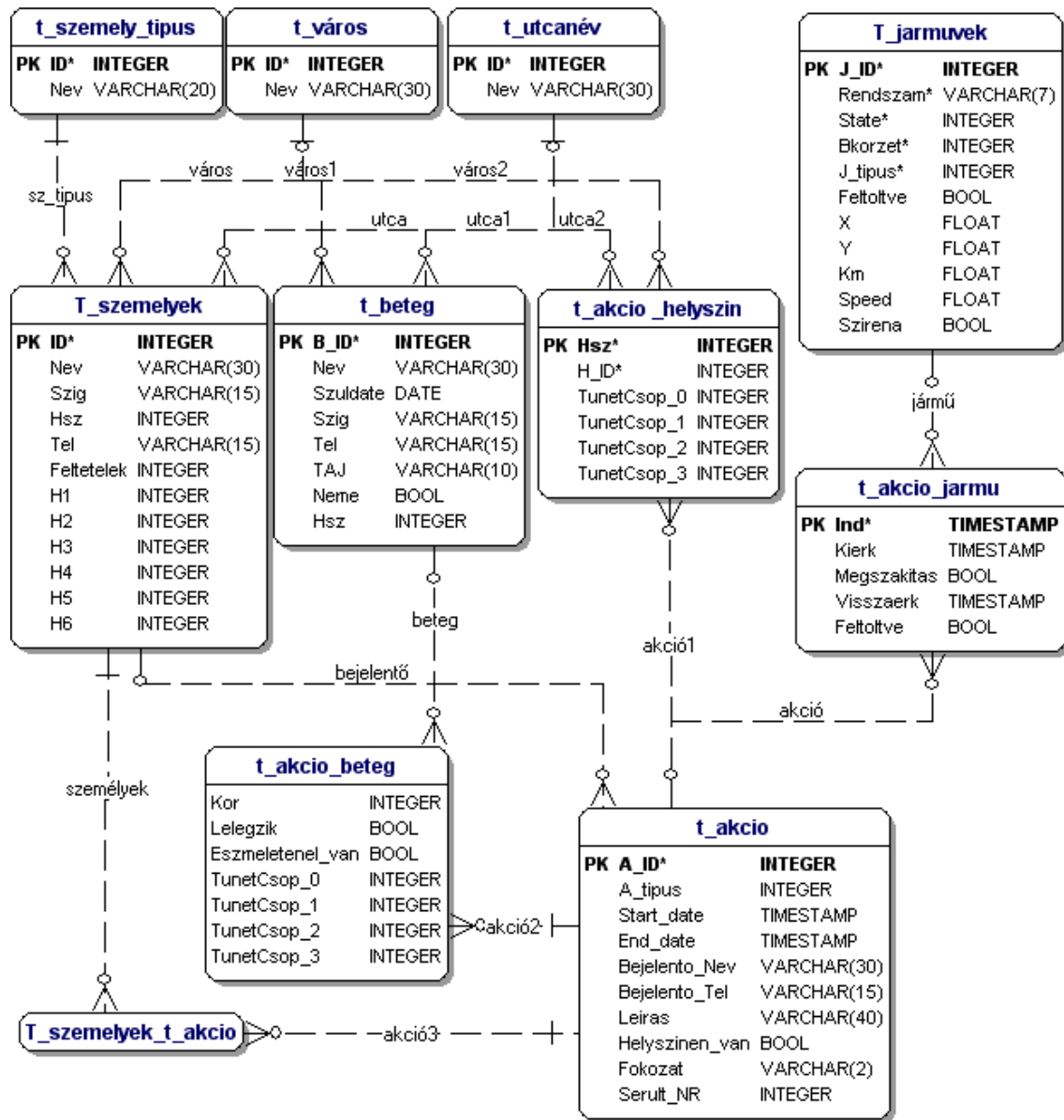
A válaszok esetén megadható a riasztási szint és az akciókód is. Továbbfejlesztési lehetőségként az akciókódokat makrókkal lehetne összekapcsolni. Így a program tünet, vagy helyszín kiválasztás estén automatikusan elvégezne különböző műveleteket.

Példa az XML fájlra:

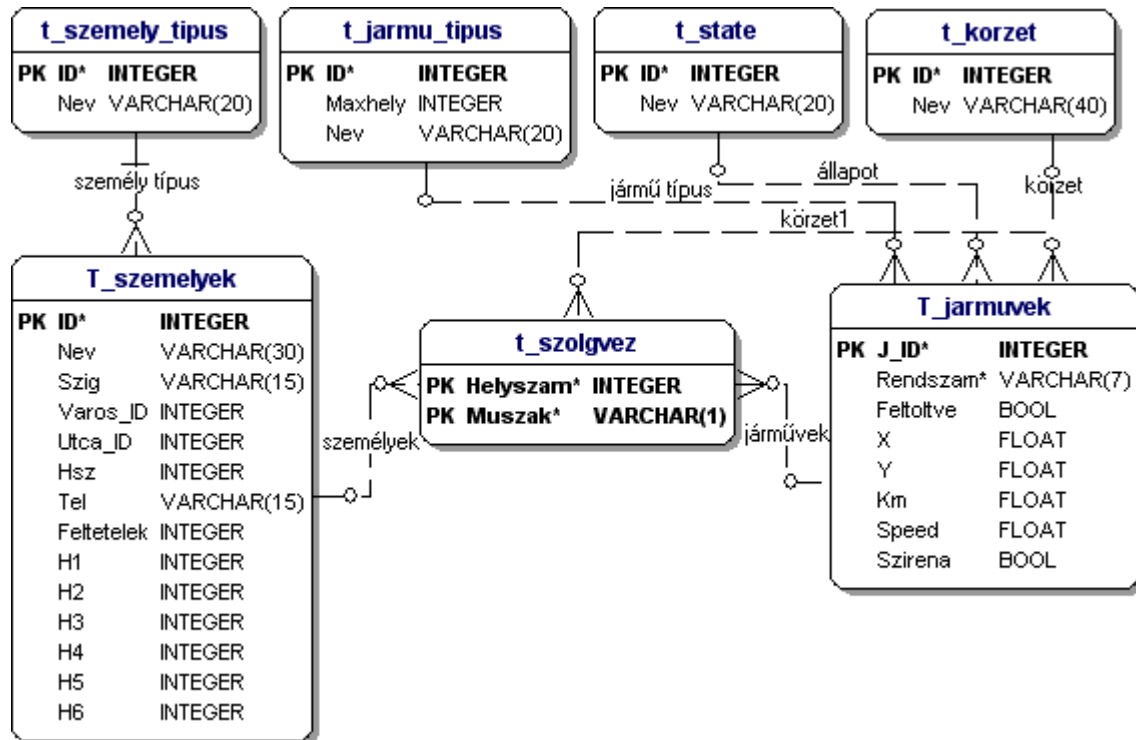
```
<m>
  <panasz name="Melkasi fájdalom / Szívpanaszok">
    <kerdes name="Légzése változott-e? (szapora, kapkodó, felületes, akadozó)"/>
    <kerdes name="Tudati állapota változott-e? (nyugtalan, aluszékony,zavart)"/>
    <valasz name="01A1 Súlyos légzési elégtelenség 'A'" akcio="01A1"/>
    <valasz name="01A2 Tudati állapotváltozás / ájulás 'A'" akcio="01A2"/>
  </panasz>
</m>
```

4.7 A program adattábláinak kapcsolata

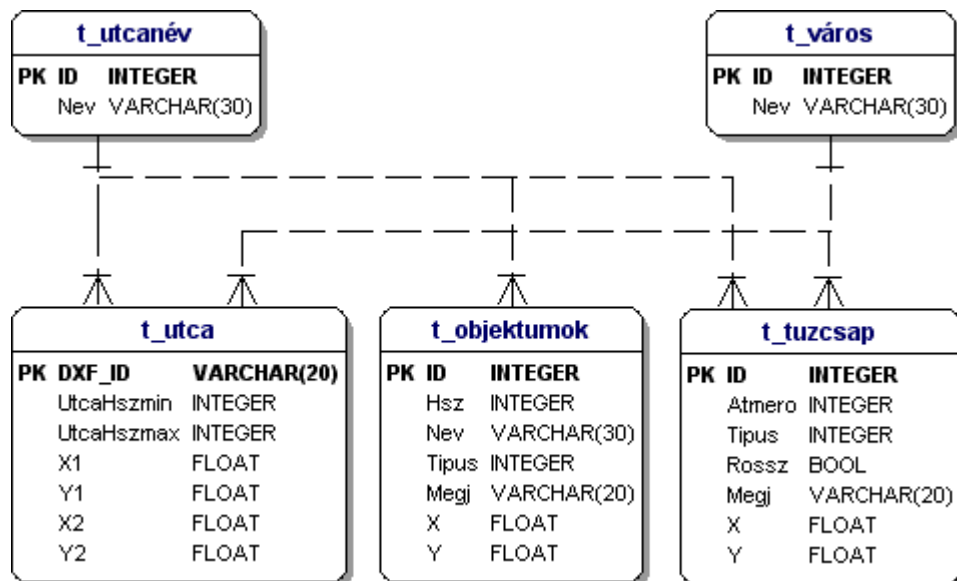
Az akció adattáblák kapcsolata



A szolgálatvezénylés tábláinak kapcsolata



Utca, helyszín adattáblák kapcsolata



4.8 A program adattábláinak leírása

Megjegyzés: - kiemelt betű = elsődleges kulcs; - dőlt betű = külső kulcs

Jármű tábla - t_jarmuvek

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|---------------------|-----|-----|-----------------------------------|---------------------------------|
| J_ID | o | | integer, auto increment, not null | |
| Rendszam | | | varchar (7) | |
| State | | o | integer, not null | a jármű állapota |
| J_tipus | | o | integer, not null | a jármű típuskódja |
| Bazis_korzet | | o | integer, not null | a jármű állomásának a körzete |
| X | | | float | GPS koordináta X |
| Y | | | float | GPS koordináta Y |
| Km | | | float | megtett út |
| Szirena | | | boolean | sziréna bekapcsolva |
| Speed | | | float | átlagsebesség |
| Feltoltve | | | boolean | a jármű feltöltöttségi állapota |

Azért nem a rendszám, az elsődleges kulcs, mert az összes kulcs a programban egész, ezt könnyebb volt használni.

Szolgálatvezénylési tábla - t_szolgvez

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------------|-----|-----|-------------------|--|
| J_ID | o | o | integer, not null | a jármű azonosítója |
| Korzet | | o | integer | a jármű aktuális körzete |
| Helyszam | o | | integer, not null | a járművön ülő személy hely(szék)száma |
| Sz_ID | o | o | integer | a helyszámot elfoglaló személy azonosítója |
| muszak | o | | varchar(1) | a műszak karakterkódja, pl.: A,B,C,... |

Jármű típus tábla - t_jarmu_tip

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------|-----|-----|-----------------------------------|----------------------------------|
| ID | o | | integer, auto increment, not null | |
| Maxhely | | | integer, not null | a jármű típus maximális létszáma |
| Nev | | | varchar (15) | a jármű típus neve |

Jármű körzet tábla - t_korzet

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------|-----|-----|-----------------------------------|---------------|
| ID | o | | integer, auto increment, not null | |
| Nev | | | varchar (15) | a körzet neve |

Jármű állapot tábla - t_state

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------|-----|-----|-----------------------------------|----------------------|
| ID | o | | integer, auto increment, not null | |
| Nev | | | varchar (15) | a jármű állapot neve |

Személyek tábla - t_szemelyek

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------------|-----|-----|-----------------------------------|---------------------------------------|
| ID | o | | integer, auto increment, not null | |
| Nev | | | varchar (30) | |
| Szig | | | varchar (15) | személyi igazolvány száma |
| <i>Varos_ID</i> | | o | integer, not null | város azonosító |
| <i>Utca_ID</i> | | o | integer, not null | utcanév azonosító |
| Hsz | | | integer, not null | házsza |
| TAJ | | | varchar (15) | |
| Tel | | | varchar (20) | |
| <i>Sz_tpus</i> | | o | integer, not null | orvosi beosztás |
| Feltetelek | | | integer | jogositványok, képesítések bitvektora |
| H1 | | | integer | statisztika mező: |
| H2 | | | integer | hányszor volt erre a helyre beosztva |
| H3 | | | integer | ... |
| H4 | | | integer | ... |
| H5 | | | integer | ... |
| H6 | | | integer | ... |

Személy típus tábla - t_szemely_tip

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------|-----|-----|-----------------------------------|-----------|
| ID | o | | integer, auto increment, not null | |
| Nev | | | varchar (30) | |

Akció csoport tábla - t_a_csop

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|--------------|-----|-----|-------------------|---|
| <i>A_ID</i> | o | o | integer, not null | az akció azonosító kódja |
| <i>Sz_ID</i> | o | o | integer, not null | az akcióban résztvevő személy az. kódja |

Betegek, sérültek tábla - t_betegek

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------------|-----|-----|-----------------------------------|---------------------------|
| ID | o | | integer, auto increment, not null | beteg azonosító |
| Nev | | | varchar (30) | |
| Szig | | | varchar (15) | személyi igazolvány száma |
| <i>Varos_ID</i> | | o | integer, not null | város azonosító |
| <i>Utca_ID</i> | | o | integer, not null | utcanév azonosító |
| Hsz | | | integer, not null | házsza |
| TAJ | | | varchar (15) | |
| Tel | | | varchar (20) | |
| Neme | | | Bool | |

Betegek állapot tábla - t_a_betegek

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|------------------|-----|-----|-------------------|-------------------------------|
| A_ID | o | o | integer, not null | akció azonosító |
| B_ID | o | o | integer, not null | beteg azonosító |
| Kor | | | integer | a beteg kora |
| Lelegzik | | | bool | lélegzik-e? |
| Eszmeletenel_van | | | bool | eszméleténél van? |
| TunetCsop0 | | | integer | tünetetcsoportok: |
| ... | | | ... | egy csoport összes válaszának |
| ... | | | ... | jelölőnégyzete |
| TunetCsop29 | | | integer | ... |

Helyszín akció tábla - t_a_helyszin

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------------|-----|-----|-------------------|---|
| A_ID | o | o | integer, not null | akció azonosító |
| Varos_ID | o | o | integer, not null | város azonosító |
| Utca_ID | o | o | integer, not null | utca azonosító |
| Hsz | o | | integer, not null | házzsám |
| H_ID | | | integer | helyszín azonosító (ha ismert helyszín) |
| Tunetcsop0 | | | integer | helyszín állapot csoportok: |
| .. | | | .. | egy csoport összes válaszának - |
| .. | | | .. | jelölőnégyzete |
| Tunetcsop30 | | | integer | ... |

Felszerelések tábla - t_felszek

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------------|-----|-----|-----------------------------------|---------------------------|
| felsz_ID | o | | integer, auto increment, not null | a felszerelés azonosítója |
| felsz_nev | | | varchar (30) | a felszerelés neve |

Felszerelés csoport tábla - t_felsz_csop

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------------|-----|-----|-------------------|-----------------------|
| A_ID | o | o | integer, not null | akció azonosító |
| felsz_ID | o | o | integer, not null | felszerelés azonosító |
| menny | | | float | meglevő mennyiség |

Akcio tábla - t_akcio

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-------------------|-----|-----|-----------------------------------|---|
| A_ID | o | | integer, auto increment, not null | akció azonosító |
| A_tipus | | | integer | akció típusa |
| Start_date | | | timestamp | akció indítási ideje |
| End_date | | | timestamp | akció befejezés ideje |
| Felvevo_ID | | o | integer, not null | bevetésirányító azonosítója |
| Bejelento_nev | | | varchar (30) | a bejelentő személy neve |
| Bejelento_tel | | | varchar (15) | |
| Leiras | | | varchar (40) | akció leírása |
| Helyszinen_van | | | bool | a bejelentő személy a helyszínen van-e? |
| Fokozat | | | varchar (2) | riasztási fokozat (1-5-ig + K jelölés) |
| SerultNR | | | integer | sérültek száma |

Akcio járművek tábla - t_a_jatmu

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-------------|-----|-----|---------------------|-----------------------------|
| A_ID | o | o | integer, not null | akció azonosító |
| J_ID | o | o | integer, not null | jármű azonosító |
| Ind | o | | timestamp, not null | a jármű indítási ideje |
| Kierk | | | timestamp | a jármű kiérkezés ideje |
| Visszaerk | | | timestamp | a jármű visszaérkezés ideje |
| Feltolte | | | bool | a jármű feltöltési állapota |

Utcanev tábla - t_utcanev

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------|-----|-----|-----------------------------------|------------|
| ID | o | | integer, auto increment, not null | |
| nev | | | varchar (20) | utca nevek |

Városnév tábla - t_varos

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------|-----|-----|-----------------------------------|-------------|
| ID | o | | integer, auto increment, not null | |
| nev | | | varchar (20) | város nevek |

Utca tábla - t_utca

| mező név: | PK: | FK: | jellemzők: | jelentés: |
|-----------------|-----|-----|-----------------------------------|---------------------------|
| DXF_ID | o | | integer, auto increment, not null | |
| Varos_ID | | o | integer, not null | város azonosító |
| Utca_ID | | o | integer, not null | utca azonosító |
| UtcaHszMin | | | integer | utcaszakasz kezdő házsza |
| UtcaHszMax | | | integer | utcaszakasz vég házsza |
| X1 | | | float | utca X kezdő koordinátája |
| Y1 | | | float | utca Y kezdő koordinátája |
| X2 | | | float | utca X vég koordinátája |
| Y2 | | | float | utca Y vég koordinátája |

Helyszínek tábla - t_objektumok

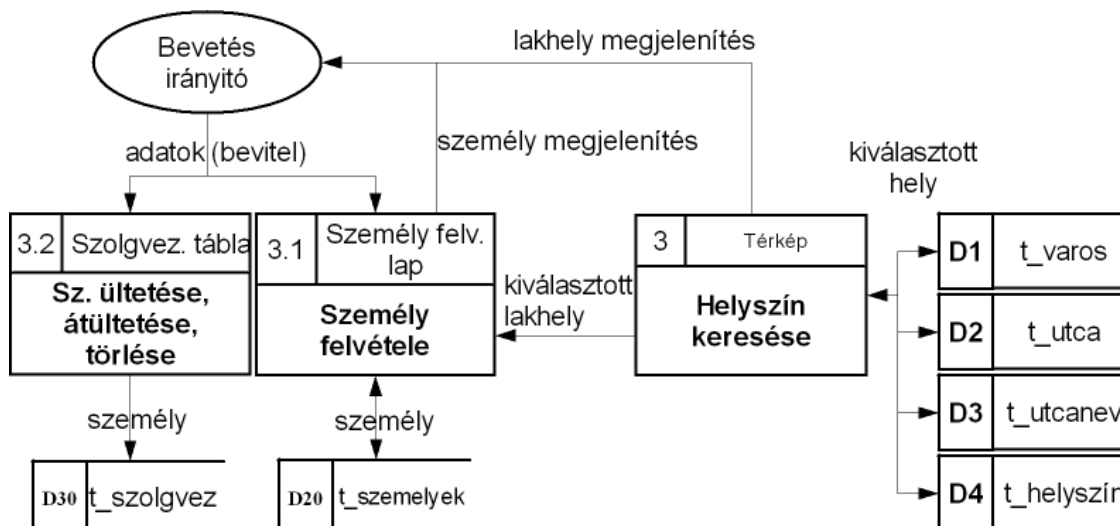
| mező név: | PK: | FK: | jellemzők: | jelentés: |
|--------------------------|-----|-----|-----------------------------------|--|
| ID | o | | integer, auto increment, not null | |
| Varos_ID | | o | integer, not null | város azonosító |
| Utca_ID | | o | integer, not null | utca azonosító |
| Hsz | | | integer | házsám |
| Nev | | | varchar (30) | |
| Típus | | | integer, not null | Helyszín típusa (okt.intézmény, iroda, stb.) |
| Megj | | | varchar (20) | |
| X | | | float | helyszín X koordinátája |
| Y | | | float | helyszín Y koordinátája |

Tűzcsapok tábla - t_tuzcsapok

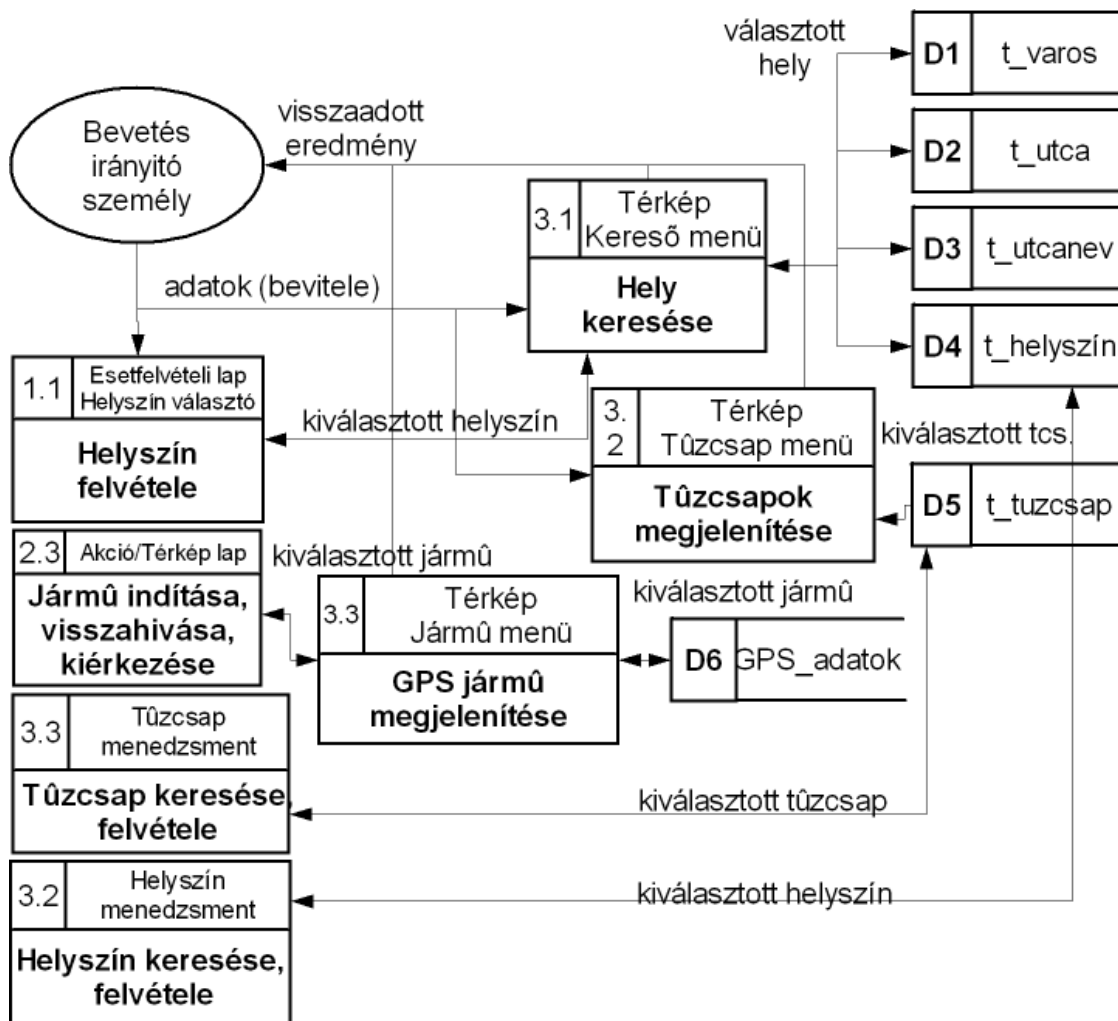
| mező név: | PK: | FK: | jellemzők: | jelentés: |
|--------------------------|-----|-----|-----------------------------------|---|
| J_ID | o | | integer, auto increment, not null | |
| Varos_ID | | o | integer, not null | város azonosító |
| Utca_ID | | o | integer, not null | utca azonosító |
| Atmero | | | integer | házsám |
| Típus | | | integer, not null | tcs. típusa (föld alatti, föld feletti) |
| Rossz | | | bool | hibátlan-e a tűzcsap? |
| Megj | | | varchar (20) | |
| X | | | float | tűzcsap X koordinátája |
| Y | | | float | tűzcsap Y koordinátája |

4.9 A program adatfolyam diagramjai (DFD)

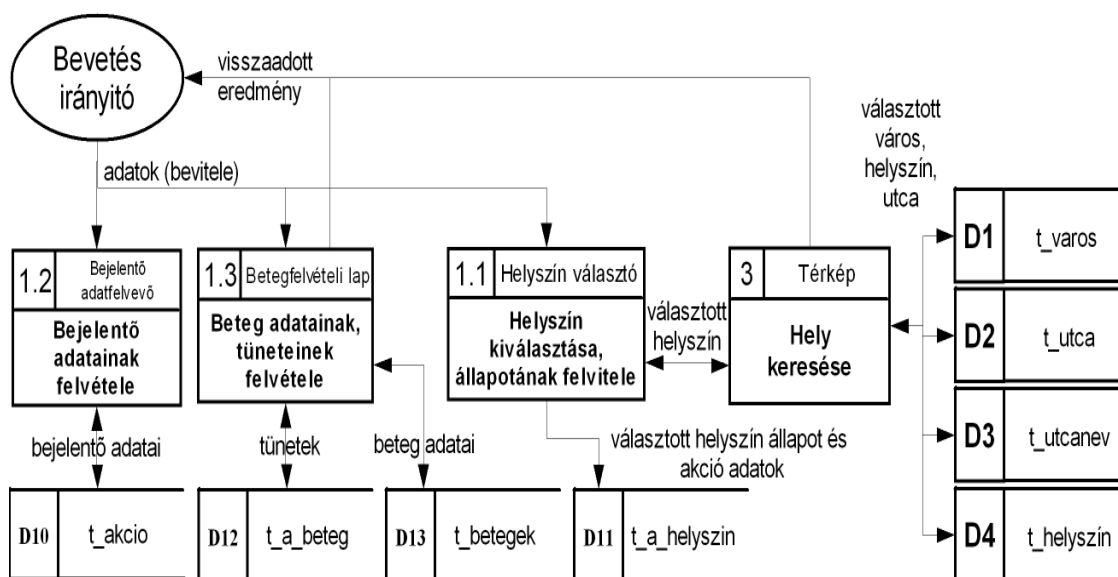
A szolgálatvezénylés DFD-je



Utca, helyszín, tűzcsap DFD



Az esetfelvétel DFD



5 Továbbfejlesztési lehetőségek

5.1 A garázskliens támogatás

A program felkészíthető lenne a járműpark kapuinak nyitására, illetve a riasztás fényjelzéseinek kezelésére. A kapunyitó áramkört a párhuzamos port segítségével lehetne vezérelni. A kapunyitás a jármű akcióindításával történne meg ezzel is gyorsítva a járművek indítását. Az utcában lévő forgalmi lámpák a forgalmat zárnák le addig, amíg a jármű kifut az állomásról.

5.2 Járműkliens támogatás

Sokkal nagyobb költségű fejlesztés lenne a járművek GPS meghatározó rendszerrel való ellátása, ami tulajdonképpen információkat küldene a bevetésirányító központba. A programban ez a támogatás megvalósult. Egy erre alkalmas GPS szerverrel a program képes lenne kommunikálni. Ennél nagyobb fejlesztés is lehetséges, amikor nemcsak koordináta információkat fogadhatunk, hanem a jármű is képes lenne információkat küldeni. Ilyen lehetőség például: a mentőjármű még az úton küldhetne információkat a kórháznak, a sérült személyről.

5.3 Többgépes támogatás

A programot ki lehetne terjeszteni többgépes rendszerre, ahol a bevetésirányítók külön terminálok segítségével irányítanák bevetéseiket. Sőt a bevetésirányító-kliensekhez akár prioritási szinteket rendelhetnénk.

Másik fejlesztési lehetőség az adatbázisok külön szerverekre való szétbontása, így ezeknek nem kell feltétlenül egy helyiségben lenni. Például a személyadatbázis gépét a személyzeti osztályon lehet tartani, mert a személyzet munkabeosztása, felvétele itt történik. Több hírközpont esetén, a fő szerver elé több kiszolgáló szervert érdemes telepíteni.

5.4 Webes fejlesztés

Hasznos fejlesztési lehetőség a c++ nyelven írt kliensek alkalmazása helyett, webes böngészőn futó programok alkalmazása. Ekkor a szerver állítaná elő a megjelenítendő felületet. Az adatbázisok kezelése ebben az esetben is a szerver feladata. A GPS jármű térkép frissítését pedig az oldal újratöltéséért felelős javascript végezné. Nem lehetetlen megoldás a rendszer PHP-ben való megírása. Az adatbázisok pedig MYSQL vagy

PostgreSQL adatbázis-szerverre való kapcsolódással lehetnének elérhetőek. Egy másik komoly érv a fejlesztés mellett, hogy napjainkban a PHP – MySQL páros kezelésére sok ingyenes szerver ad támogatást.

6 Irodalomjegyzék

Felhasznált anyagok:

- [1] Dr. Katona Endre: Térinformatika (előadási jegyzet), SZTE 2005, [47-48,112-114]
<http://www.inf.u-szeged.hu/~katona/gis.pdf>
- [2] AutoCad: DXF Reference, Autocad 2005, [4-10,13,44,61-107]
<http://www.autodesk.com/techpubs/autocad/acad2000/dxf/index.htm>
- [3] McGill University: Picture Representation Using Quad Trees, McGill Univ. 1999,
<http://www.cs.ubc.ca/~pcarbo/cs251/welcome.html>
- [4] www.magyarorszag.hu: A segélyszolgálatok egységes bevetés-irányítása(cikk) 2004,
<http://www.magyarorszag.hu/hirek/kozelet/bevetesiranyitas20040217.html>
- [5] Rumpler István: Vitorlázórepülő versenykiértékelő program GNSS (Global Navigation Satellite System) segítségével (diplomamunka), SZTE 1999
- [6] Kincses László: SSADM strukturált rendszerelmezési és tervezési módszer, MTA Információtechnológiai Alapítvány 1993
<http://www.itb.hu/ajanlasok/a4/a4.doc>
- [7] Szegedi Önkormányzat: szeged.jpg - Szeged térkép (információs katalógusból)
<http://www.szegedvaros.hu/index2.php?oldal=letoltesek>

Felhasznált programok:

Borland C++ Builder 5.0 - programozási nyelv – www.borland.com
JAVA - programozási nyelv – www.java.sun.com
Eclipse - JAVA fejlesztő környezet - www.eclipse.org
MySQL - adatbázis – www.mysql.com
SmartDraw 7.0 - ábratervező program – www.smartdraw.com
Metamill 4.0 - ábratervező program – www.metamill.com
DeZign for Databases 3.4.0 - ábratervező program – www.datanamic.com

Nyilatkozat

Alulírott Szikszai Sándor programtervező matematikus szakos hallgató, kijelentem, hogy a dolgozatomat a Szegedi Tudományegyetem, Informatikai Tanszékcsoport Képfeldolgozás és Számítógépes Grafika Tanszékén készítettem, a programtervező matematikus diploma megszerzése érdekében. Kijelentem, hogy a dolgozat saját munkám eredménye, és csak a hivatkozott forrásokat (szakirodalom, eszközök, stb.) használtam fel.

Tudomásul veszem, hogy a diplomamunkámat a Szegedi Tudományegyetem könyvtárában, a kölcsönözhető könyvek között helyezik el.

Aláírás:.....

Dátum:.....

7 Köszönetnyilvánítás

Köszönetet szeretnék mondani témavezetőmnek **Dr. Katona Endrének**.

Szeretném megköszönni az OMSZ informatikai osztály vezetőjének **Kikindai Bélának** a mentőszolgálattal kapcsolatos észrevételeit, és a mentési információkat.

Dr. Cziva Oszkár tűzoltó ezredesnek köszönöm, hogy lehetőséget kaptam a Fővárosi Tűzoltó Parancsnokság munkájának megismeréséhez, valamint szeretném megköszönni **Kun Zoltán** főhadnagynak a tűzoltóság működésével, munkájával kapcsolatos információkat, illetve a pontos meghatározásokat.