

BKSZC Pogány Frigyes Szakgimnáziuma  
Szoftverfejlesztő  
54 213 05

# **Információs rendszer kis- és középvállalatok számára**

Tóth József  
konzulens

Kiss J. Gábor  
2/14. ED

Budapest, 2020

© Kiss József Gábor, 2020

© BKSzC Pogány Frigyes Szakgimnáziuma, 2020

# Tartalom

<b>1</b>	<b>FELHASZNÁLÓI DOKUMENTÁCIÓ.....</b>	<b>1</b>
1.1	A PROGRAMRÓL ÁLTALÁBAN.....	1
1.2	RENDSZERKÖVETELMÉNYEK.....	2
1.2.1	Szoftver.....	2
1.2.1.1	Windows.....	2
1.2.1.2	Mac OS X.....	2
1.2.1.3	Linux.....	2
1.2.2	Hardver.....	2
1.3	A PROGRAM TELEPÍTÉSE.....	3
1.4	A PROGRAM HASZNÁLATÁNAK A RÉSZLETES LEÍRÁSA.....	3
1.4.1	Az alkalmazás indítása.....	4
1.4.2	Belépés az adatbázisba.....	5
1.4.3	Az adatbázis használata.....	7
1.4.4	Kijelentkezés és kilépés a programból.....	10
1.4.5	A grafikus felület részei.....	11
1.4.5.1	A menü sáv.....	11
1.4.5.2	Az eszköztár.....	12
1.4.5.3	Az állapotjelző sáv.....	14
1.4.6	Fejlesztési lehetőségek.....	14
<b>2</b>	<b>FEJLESZTŐI DOKUMENTÁCIÓ.....</b>	<b>15</b>
2.1	TÉMAVÁLASZTÁS.....	15
2.2	AZ ALKALMAZOTT FEJLESZTŐI ESZKÖZÖK.....	16
2.3	AZ ADATMODELL LEÍRÁSA.....	16
2.3.1	Az adatbázis.....	16
2.3.2	A vállalat adatai tábla.....	17
2.3.3	A felhasználói csoportok tábla.....	17
2.3.4	A felhasználók adatai tábla.....	17
2.3.5	A feljegyzések tábla.....	18
2.3.6	A felhasználók.....	19
2.3.7	Az adatbázis fejlesztése.....	21
2.3.7.1	Az eszköz csoportok tábla.....	22
2.3.7.2	Az eszközök adatai tábla.....	22
2.3.7.3	Az eszköznyilvántartás tábla.....	22
2.3.7.4	Az új felhasználó csoportok.....	23
2.4	RÉSZLETES FELADATSPECIFIKÁCIÓ, ALGORITMUSOK.....	23
2.5	FORRÁSKÓD.....	25
2.6	TESZTELÉSI DOKUMENTÁCIÓ.....	25
2.7	TOVÁBBFEJLESZTÉSI LEHETŐSÉGEK.....	25
2.8	IRODALOMJEGYZÉK, FORRÁSMEGJELÖLÉS.....	25

*Információs rendszer kis- és középvállalatok számára*

## 1 Felhasználói dokumentáció

### 1.1 A programról általában

Egy feladat, egy projekt megvalósítása során a résztvevők közötti tájékoztatási rendszer jó működése biztosítja a sikert. Lényeges, hogy az adatok átadása gördülékeny, áttekinthető és nyomon követhető legyen. Ehhez biztosít kiváló lehetőségeket az elektronikus kommunikáció, mely napjainkra az információ közvetítésének alapvető módjává vált.

A kapcsolattartás sokféle lehetőségét kínálják a különböző közösségi hálózatok és levelező programok. Ezeket a megoldásokat azonban nehézkes a szűkebb vállalati környezetben alkalmazni.

A tájékoztatási rendszer hiánya negatívan befolyásolja a cégvezető (projektvezető) tevékenységét. Az emberei elbeszélnek egymás mellett, pedig egy kis félreértés is hatalmas hibákat okozhat az üzletmenetben. A munkatársak felesleges adminisztratív munkával töltik drága idejüket. Stresszesen, papírhalmokat átnyálazva keresnek adatokat, mert nincs ott az a kolléga, aki tudja hol találhatók. Egy egyszerű kérdésre is órákig vadásszák össze az adatokat az Excel táblázatokból. Nem tudja melyik munkatársa mit csinál, ki eredményes és kitől kellene megválnia. Óriási kínlódást jelent egy távozó munkatárs pótlása, mert csak az Ő fejében voltak meg fontos információk.

Az információs rendszer egy kapcsolattartási megoldást kínál, mely elősegíti egy vállalat ügymenetét. Minden munkatárs könnyen hozzájut a számára lényeges információkhoz. Mindig látszik, hogy kinek mi a feladata. Átlátható a működés. A kapcsolódás és stressz helyett tervezett és nyugodt mindenki munkája. Azonnali, naprakész információk alapján gyorsan, helyes döntéseket lehet hozni. A vezető minden munkatársáról pontosan tudja, hogy éppen mit csinál, min dolgozik.

A program alaprendszere egyszerűen módosítható és bővíthető a cég speciális igényinek figyelembevételével. Személyre szabott, a felelősséghez és munkakörhöz kapcsolódó hozzáférési jogosultságok állíthatók be. Az alapprogram adatbázisa tartalmazza a cég és a dolgozók adatait, valamint a vállalati projektekkel kapcsolatos munkájukat. A felhasználók a programban feljegyzéseik segítségével biztosítják az információ áramlását és dokumentálják tevékenységüket.

A programban elérhető lehetőségek igény szerint bővíthetők. A következő felsorolásban kiemelek néhány példát a lehetséges kiegészítő modulokra:

- munkaidő nyilvántartás
- szabadság nyilvántartás
- eszköz- és anyagnyilvántartás
- fizetések és juttatások
- útnyilvántartás (kiküldetési rendelvények)
- készletnyilvántartás
- partnerek
- kimenő és bejövő számlák nyilvántartása
- ajánlatok és szerződések nyilvántartása
- számlák nyilvántartása
- ...

Az információs rendszer programmal a résztvevők minden szinten pontos képpel rendelkeznek a projektek állásáról, folyamatosan értesülnek az új fejleményekről. Új modulok beépítésével az ügyviteli folyamatok automatizálhatók, így az időrabló feladatok néhány gombnyomással elvégezhetők lesznek.

## **1.2 Rendszerkövetelmények**

### **1.2.1 Szoftver**

#### **1.2.1.1 Windows**

Windows Vista SP2,  
Windows 7 SP1,  
Windows 8.x,  
Windows 10 (8u51 vagy újabb),  
Windows Server 2008 R2 SP1 (64-bit),  
Windows Server 2012 and 2012 R2 (64-bit)

#### **1.2.1.2 Mac OS X**

Intel-alapú Mac és Mac OS X 10.7.3 (Lion) vagy későbbi,  
Adminisztrátor jogok a telepítéshez.

#### **1.2.1.3 Linux**

- Oracle Linux 5.5+,  
Red Hat Enterprise Linux 5.5+,  
Suse Linux Enterprise Server 10 SP2,  
Ubuntu Linux 10.04 vagy újabb.

### **1.2.2 Hardver**

Processzor: Pentium 2 266 MHz vagy gyorsabb,  
RAM: min. 128 MB,  
Szabad tárhely: min. 124 MB.

### 1.3 A program telepítése

A java futtatókörnyezet telepítése alapvető feltétele az alkalmazás használatának. A program fejlesztése java 1.8 környezetben történt, így futtatásához a Java SE Development Kit 8 telepítése ajánlott. Mivel kompatibilis a legújabb Oracle java verziókkal és az Open JDK-val is, ezért szükségtelen a már telepített java környezet megváltoztatása, vagy visszaállítása alacsonyabb verzió számúra.

A z alkalmazás MySQL alapú adatbázist kezel. Az adatbázis lehet lokális elérésű és lehet távoli elérésű is. A példában lokális elérésű adatbázist használok, mely nyilvános IP cím hiányában csak a munkahelyi magánhálózat gépeiről érhető el. Az adatbázis kezelést az XAMPP<sup>1</sup> webservert-szoftvercsomag MariaDB összetevője biztosítja.

A java futtatókörnyezet és az XAMPP webservert csomag telepítése, illetve meglétük ellenőrzése után az alkalmazás a mappájában található B6kkvXxx.jar fájl elindításával futtatható. Ez egy úgynevezett portable (hordozható) program, mely telepítés nélkül, akár USB kulcsról is futtatható. Lényeges, hogy a tartalmazó mappát teljesen, mindenestől kell „hordozni”. Az alkalmazás működéséhez szükséges a lib mappa a benne található mysql-connector-java-5.1.23-bin.jar fájljal együtt.

Az adatbázis kezdeti beállítását a V01-01kkvDBcreate.sql, a V01-02kkvDBtables.sql, a V01-03kkvDBusers.sql és a V01-04kkvDBdatas.sql fájlok importálásával végezzük el. Ezek a beállítások érzékeny adatokat tartalmaznak, amik nem írhatók bele az alkalmazás kódjába. A továbbfejlesztés során szóba jöhet titkosítás az importálandó fájlknál.

### 1.4 A program használatának a részletes leírása

Az alkalmazás egy adatbázist használ, ami egy adatbáziskezelőn keresztül érhető el. Az adatbázis kezelést esetünkben az XAMPP csomag biztosítja. A helyi hálózat egyik gépén futnia kell az Apache webservertnek és a MariaDB adatbázis-

<sup>1</sup> Az XAMPP egy szabad és nyílt forrású platformfüggetlen webservert-szoftvercsomag, amelynek legfőbb alkotóelemei az Apache webservert, a MariaDB (korábban a MySQL) adatbázis-kezelő, valamint a PHP és a Perl programozási nyelvek értelmezői.

kezelőnek. Ezeket az XAMPP kontroll panelen indíthatjuk el, bekapcsolt állapotukat a panelen a nevük zöld színű kiemelése jelzi.

#### 1.4.1 Az alkalmazás indítása

Az alkalmazást a program mappájában található B6kkvXxXx.jar fájl futtatásával indítjuk. Első lépésben a program csatlakozik az adatbázishoz.



1. Ábra: Az alkalmazás csatlakozik az adatbázishoz

A csatlakozási folyamatot a menü sor alatti eszköz panelen kísérhetjük figyelemmel (1. ábra).

Amennyiben egy másik gépen működik a szerver és az adatbázis-kezelő, akkor az eszköztár jobb szélén található „felhő”-gombra kattintás után megjelenő ablakban ki kell egészíteni, távoli szerver esetén pedig meg kell adni a a szerver-gép IP-címét. Az elfogadás ([OK]-gomb) után ezen az új címen keresi az adatbázis-kezelőt a program és folytatja a csatlakozást.

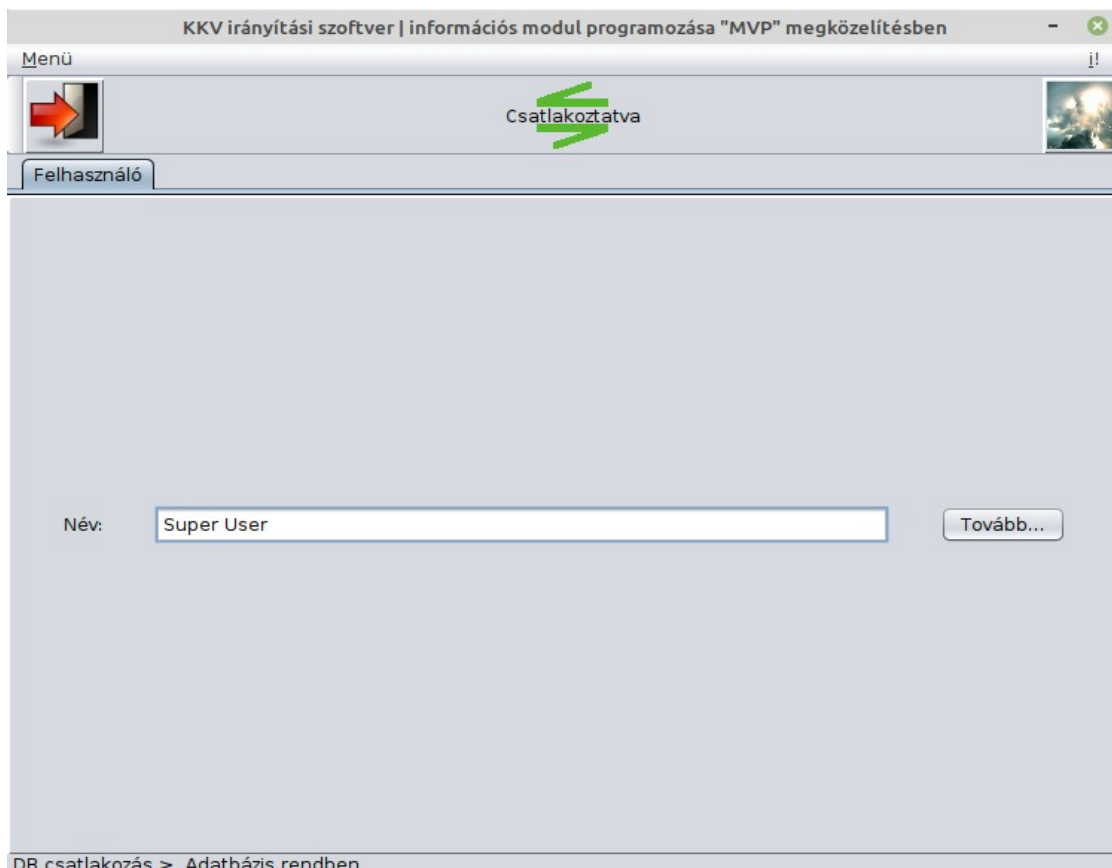


Sikeres csatlakozás esetén

- az eszköztár jobb szélén a „felhő”-gomb világosabbra vált;
- a középső ikon átvált „csatlakoztatva” állapotba;
- az alsó állapotjelző sorban pedig megjelenik az „adatbázis rendben” felirat (2. ábra).

### 1.4.2 Belépés az adatbázisba

Az adatbázisba való bejelentkezéshez első lépésben az ablak közepén levő mezőben meg kell adni a felhasználó nevet, majd a „tovább” gombra kattintva a felhasználói jelszót.

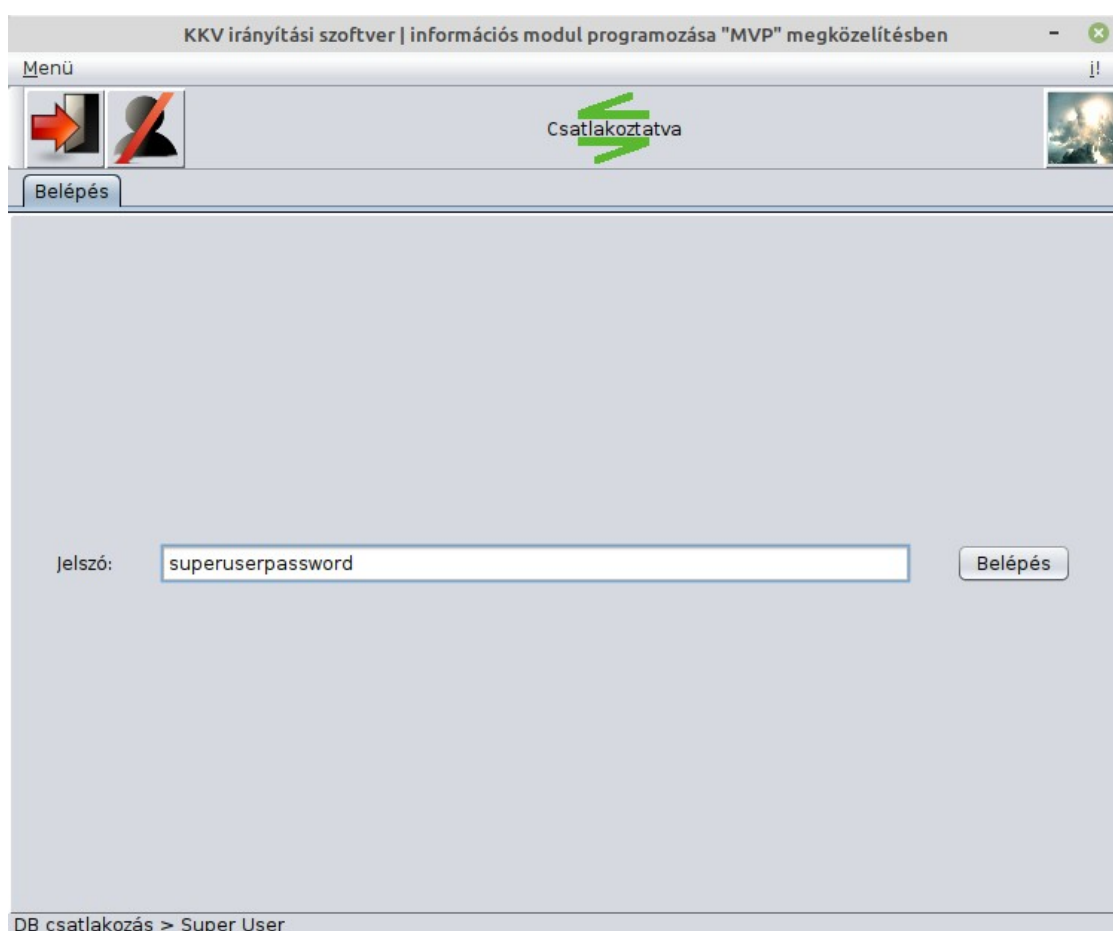


2. Ábra: Az alkalmazás csatlakozott az adatbázishoz

A bejelentkezés során az alkalmazás a felhasználói név alapján hozzárendeli a belépő felhasználóhoz a jogosultságot, ami alapján a program futása során csak az annak megfelelő elemeket jeleníti meg a munkaterületen. Ez azt jelenti a példa alkalmazásnál, hogy az adminisztrátori jogokkal rendelkező Super User az adatbázis teljes forgalmát láthatja, a csoport vezető a csoport összes feljegyzéséhez hozzáfér, míg a csoport tagjai esetünkben csak a saját maguk által létrehozott, vagy a nekik

szóló jegyzeteket láthatják. A jogosultságok és az általuk meghatározott hozzáférések megváltoztatása csak a nem kívánt szerepkör elvetésével és egy új szerepkör bevezetésével, teljes felépítésével történhet. Ebben az esetben ehhez az új jogosultsági szinthez új lekérdezések és új grafikus felületek is tartoznak, tartozhatnak.

Példánkban, amennyiben a felhasználó jogosultsággal rendelkezik az adatbázisba történő belépésre, akkor a [Tovább...] lenyomásával eljutunk a jelszó megadásához (3. ábra). Az állapot jelző sorban látható lesz a felhasználónév. Az eszköztárban a kilépés gomb mellett megjelenik a kijelentkezés gomb is, mely itt a bejelentkezés megszakításával visszavisz az előző képernyőre.

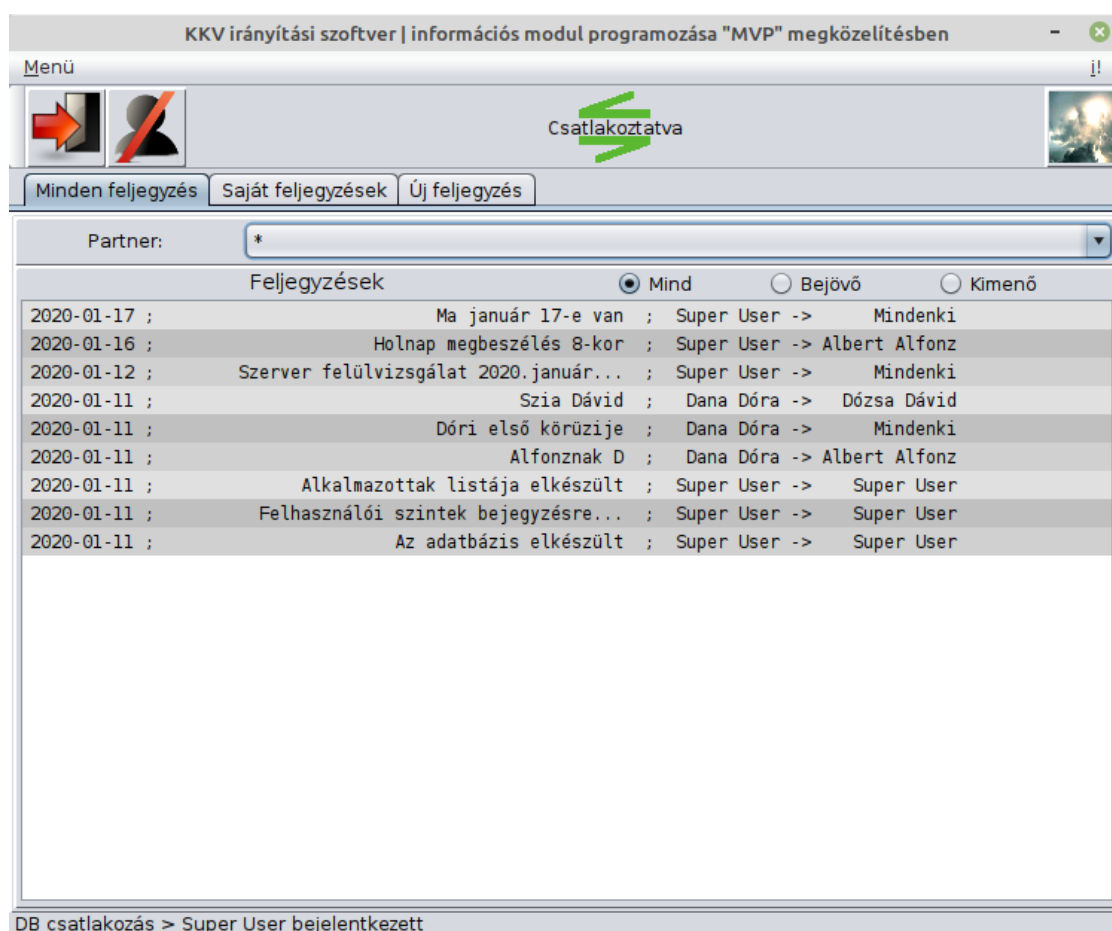


3. Ábra: A belépési jelszó megadása

Amennyiben az ablak közepén levő mezőbe a helyes jelszót adjuk meg, akkor a [Belépés] sikeresen megtörténik. A bejelentkezéshez használt felhasználónév és jelszó páros nem alkalmas az adatbázishoz történő közvetlen hozzáférésre (pl. phpMyAdmin-on keresztül), így is garantálva a rögzített adatok sérthetetlenségét.

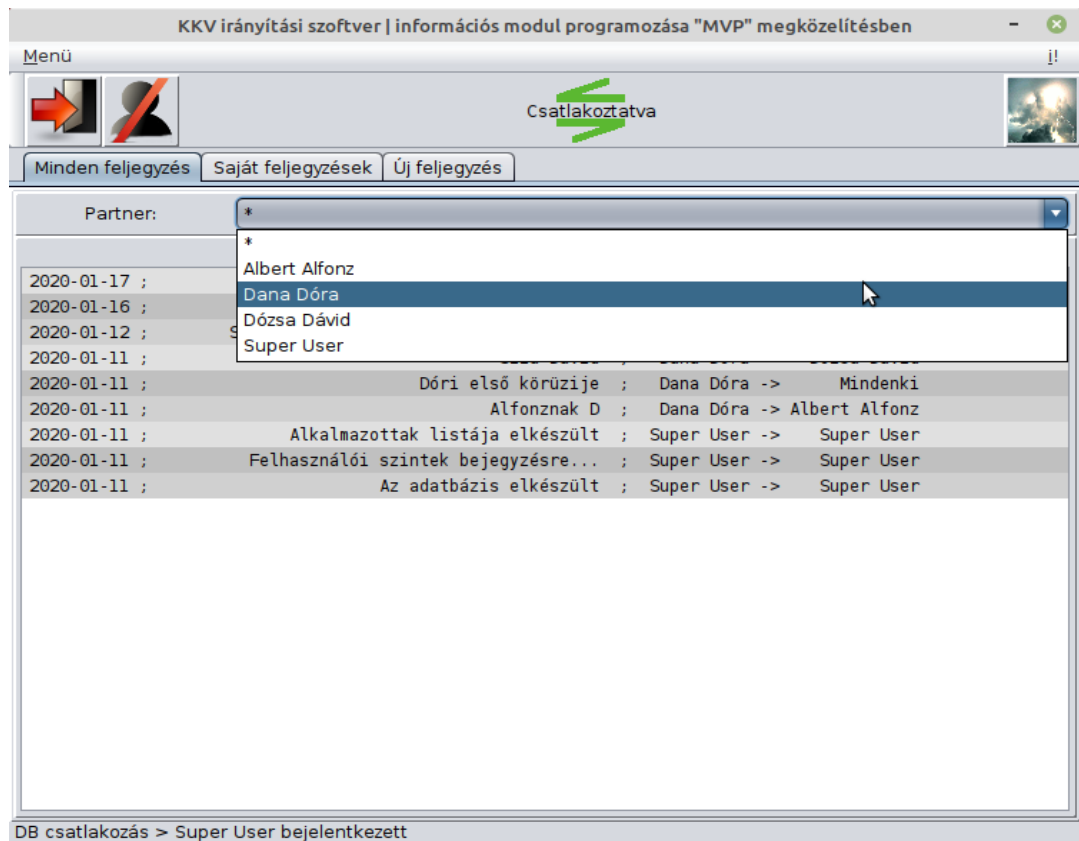
### 1.4.3 Az adatbázis használata

Tehát a [Belépés] gombra kattintva eljutunk a felhasználó jogosultságainak megfelelő munkafelülethez. Példánkban „Super User”-ként és vezetőként hozzáférhetünk a teljes információs adatbázishoz, csoporttagként pedig csak az általunk írt és a nekünk (is) címzett feljegyzésekhez. Feljegyzés írására esetünkben mindenki jogosult. Amennyiben nem vagyunk az információs adatbázissal támogatott fejlesztés részesei, akkor nincs jogosultságunk az adatbázisba történő belépésre sem.

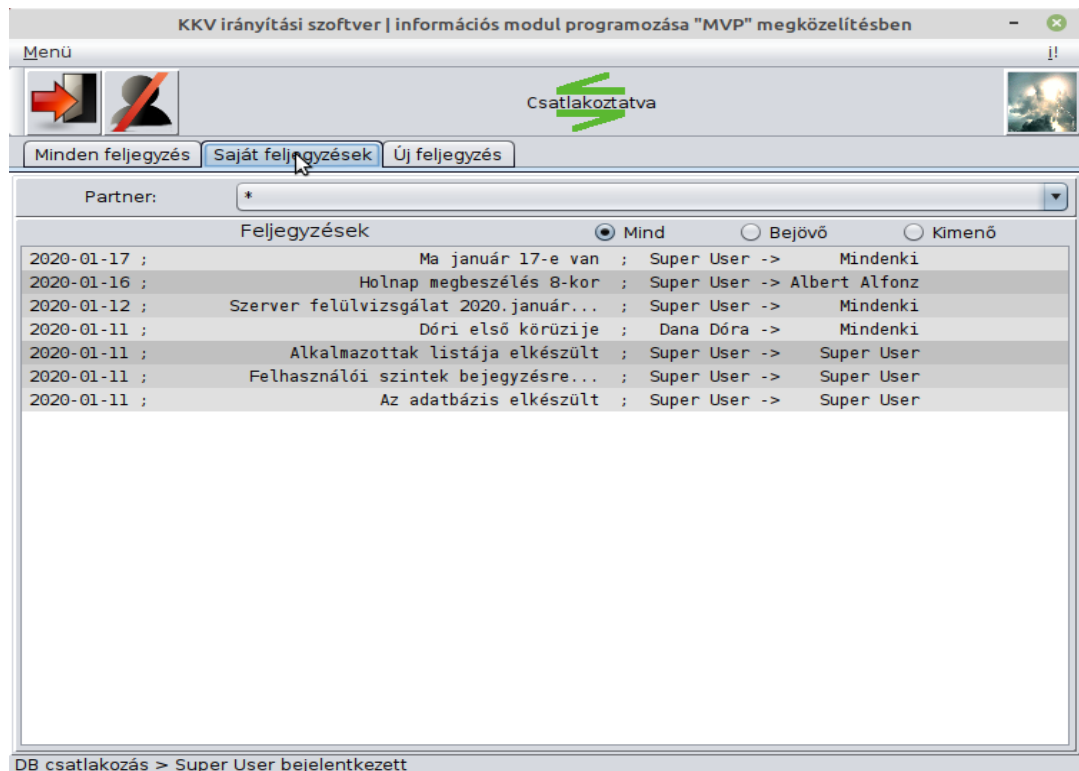


4. Ábra: Super User minden feljegyzés fül

A 4. ábrán látható a belépés utáni felhasználói felület. Az adminisztrátori és csoportvezetői jogokkal rendelkezőknek a „Minden feljegyzés” fül jelenik meg a „Saját feljegyzések” és az „Új feljegyzés” fülek fölött. A legördülő menüből kiválasztható, hogy kinek a tevékenysége érdekel (5. ábra). A \* választása esetén az összes felhasználó feljegyzése látható lesz időrendileg csökkenő sorrendben. A „Mind”, a „Bejövő” és a „Kimenő” gombokkal tovább finomíthatjuk szűrést. Egy üzenetre kattintva egy felugró ablakban megjelenik a teljes üzenet, minden adatával együtt.



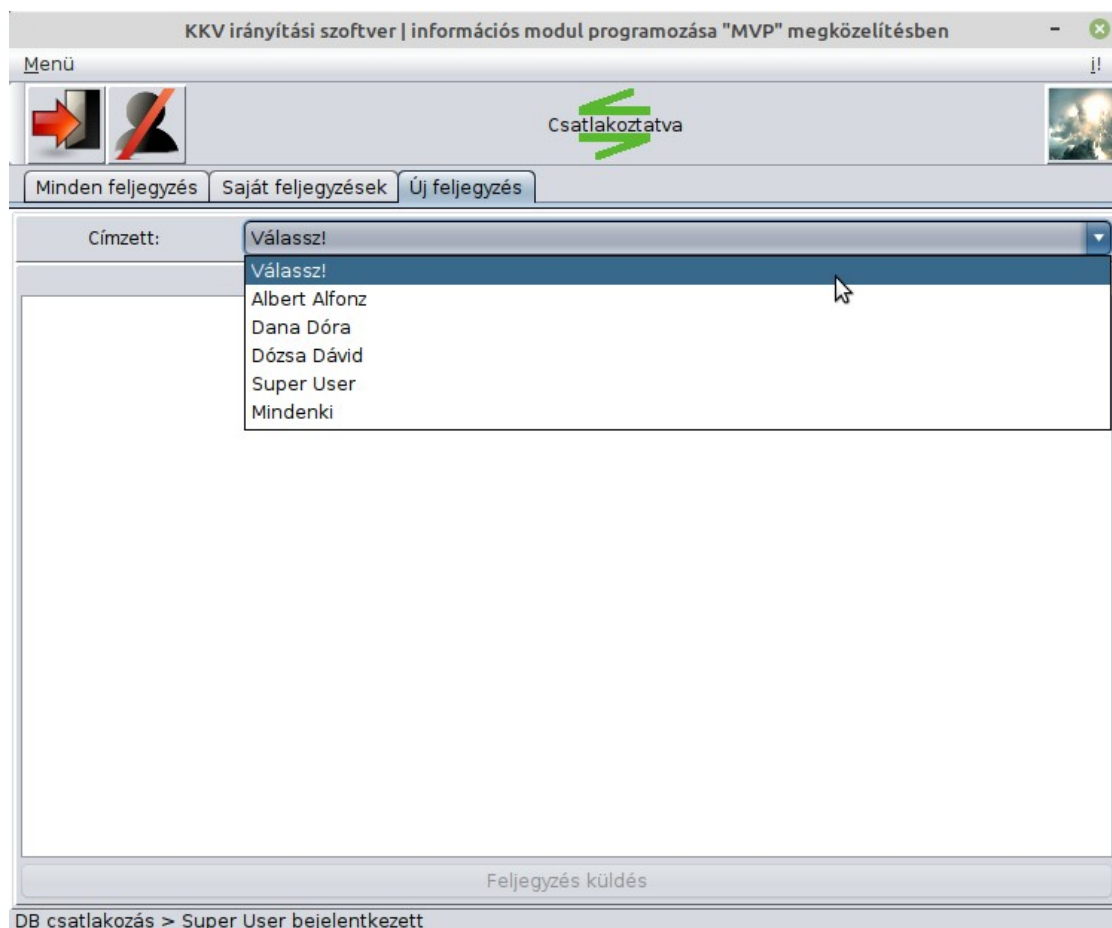
5. Ábra: Szűrés legördülő menüből név alapján



6. Ábra: Saját feljegyzések fül

A „Saját feljegyzések” fülön a felhasználó az általa írt és a neki címzett értesítéseket látja (6. ábra).

A tárolt adatoknál fontos követelmény a tárolt információ hitelessége. Ezért az elkészült feljegyzés nem módosítható és nem is törölhető. Amennyiben valamilyen okból a feljegyzés hibás adatokat tartalmazott, azokat csak egy új feljegyzés készítésével lehet módosítani. Ezzel a felelősség fontos kérdését igyekeztem figyelembe venni. A szűrés az előzőekkel megegyező módon történik. A feljegyzés teljes tartalma és adatai kattintásra itt is megjelennek.



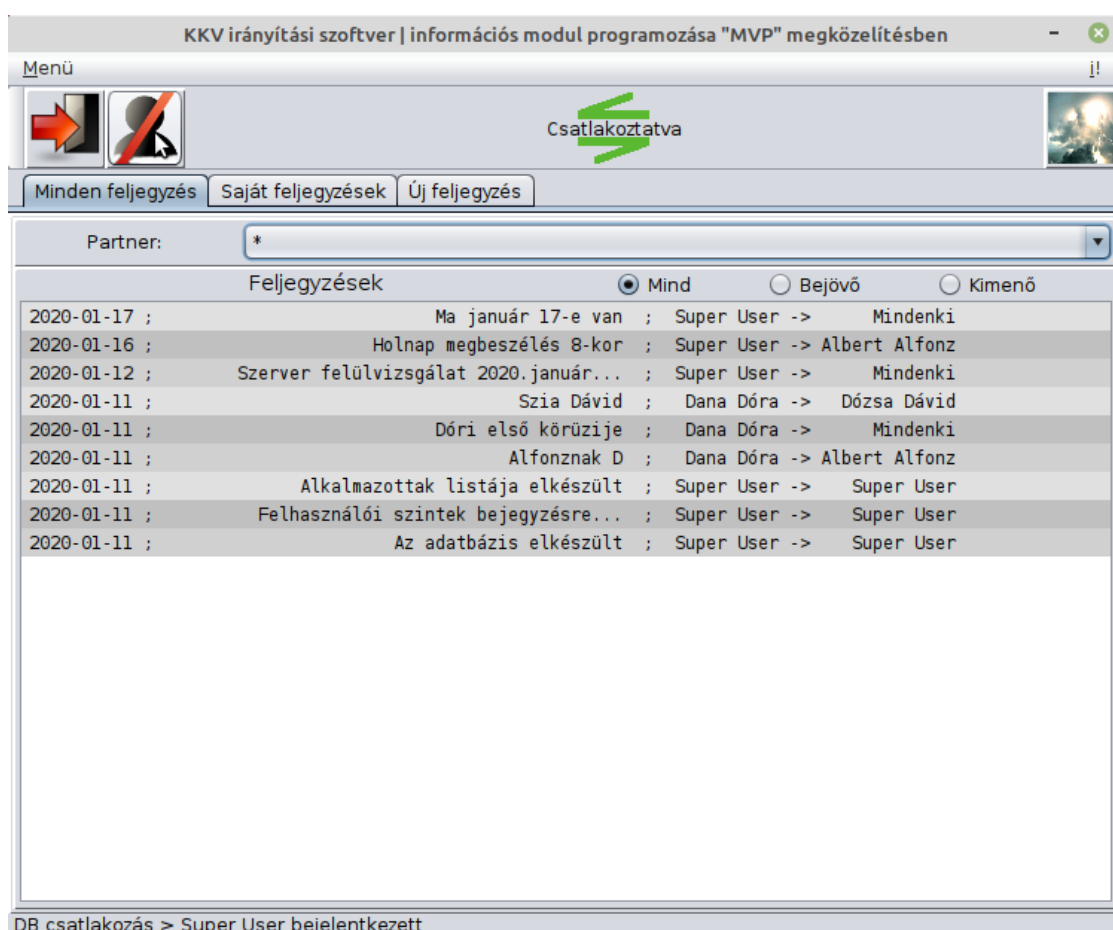
7. Ábra: Új feljegyzés írása

Az „Új feljegyzés” fülön saját feljegyzést, infó blokkot készíthetünk. Első lépésben ki kell választani azt a személyt, akivel az információt meg akarjuk osztani. Választható a „Mindenki” opció is (7. ábra). Ebben az esetben mindenki láthatja majd a kész dokumentumot. A feljegyzés gépelése közben engedélyezetté válik a „Feljegyzés küldés” gomb. A gombra kattintással visszavonhatatlanul rögzítésre kerül az adatbázisban a feljegyzés.

A feljegyzés moderálására csak a rendszergazda képes, az adatbázisba történő belépés után. Ekkor is csak a nem kívánatos esetleg sértő tartalmak semleges (pl.: xxx) karakterekkel történő feltöltése, felülírása a javasolt.

#### 1.4.4 Kijelentkezés és kilépés a programból

A munka nézetből a bejelentkezés ablakhoz a kijelentkezés gombbal, vagy a Menüben található Kijelentkezés pont aktiválásával, vagy az **Alt+x** billentyűkombinációval lehet visszatérni.



8. Ábra: A kijelentkezés és kilépés gombok az eszköztáron

A programból a kilépés gombbal, vagy a Menüben található Kilépés pontjának, vagy az eszköztár kilépés gombjának aktiválásával, és az **Alt+F4** billentyűkombinációval lehet kilépni. Ezekben az esetekben egy rövid animációval záródik be az ablak. Amennyiben az ablakot bezáró [x] gombot használjuk a kilépéshez, akkor az animáció nem jelenik meg (8. ábra).

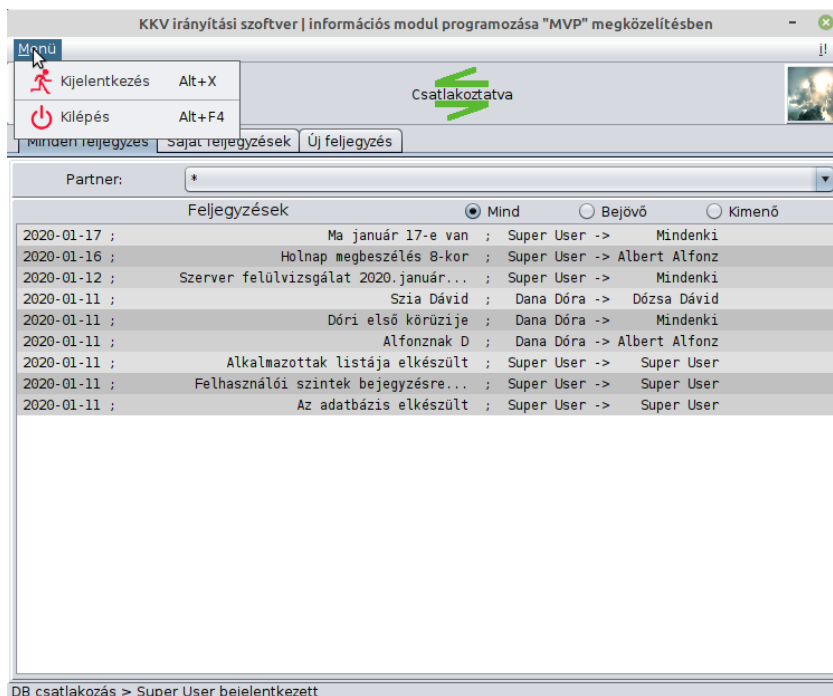
A program a beküldött adatokat azonnal tárolja, így a kilépés a heveny viselkedést okozó visszakérdezés - „Biztosan kilép a programból? (I/N)” - nélkül történik.

### 1.4.5 A grafikus felület részei

#### 1.4.5.1 A menü sáv

A menü sávon a főmenü ( **Menü** ) és az információs menü ( **i!** ) menü található. A program továbbfejlesztése során ez igények alapján további bővíthető.

A „**Menü**” pont induláskor csak a „Kilépés” lehetőségét tartalmazza. A „Kijelentkezés” almenü (9. ábra) a bejelentkezési folyamattal párhuzamosan kerül hozzáfűzésre.

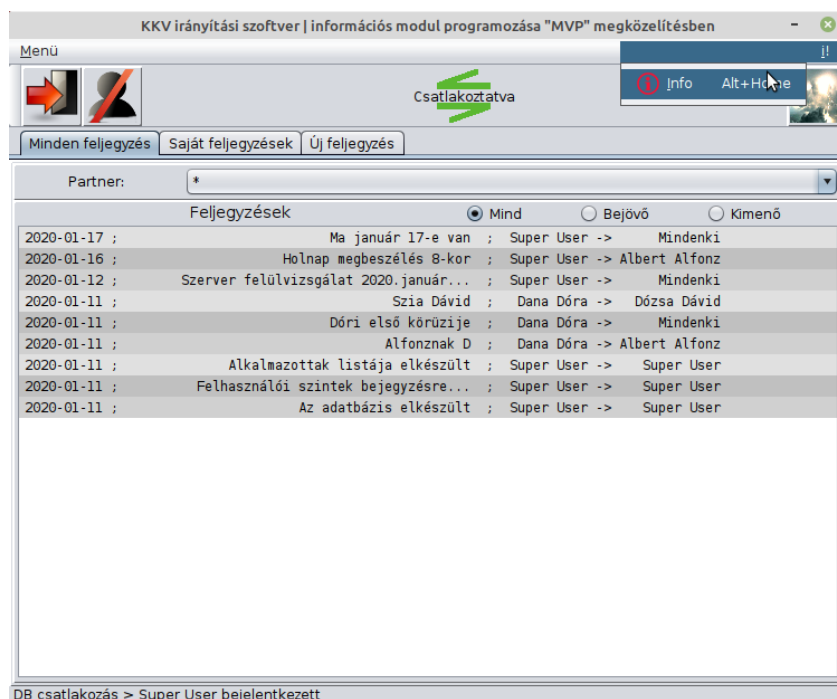


9. Ábra: A menüsáv, a főmenü

A menü sáv további menüpontokkal történő bővítése hasonlóan az almenü pontjaihoz, a programban megjelenő igény szerint dinamikusan történhet.

Például, amennyiben a „Super User” felhasználóra bízunk a felhasználók kezelését a programban, akkor megjelenítjük a „Felhasználó” menüt a menüsávon és hozzáfűzésre kerülnek a lenyíló menüben az „Új felhasználó”, a „Felhasználó törlése” és a „Felhasználó adatainak módosítása” alpontok is. Természetesen csak akkor, ha a „Super User” lép be.

Az információs menü egy alpontjában a program néhány adata jeleníthető meg. A fejlesztés során itt helyet kaphatnak a segítség, a copyright, a cégalapítás és más további alpontok is. Az információs menü aktiválható az **Alt+i** billentyű kombinációval is. Az **Alt+Home** kombináció pedig közvetlenül aktiválja a felugró információs ablakot (10. ábra).



10. Ábra: A menüsáv, az információs menü

#### 1.4.5.2 Az eszköztár

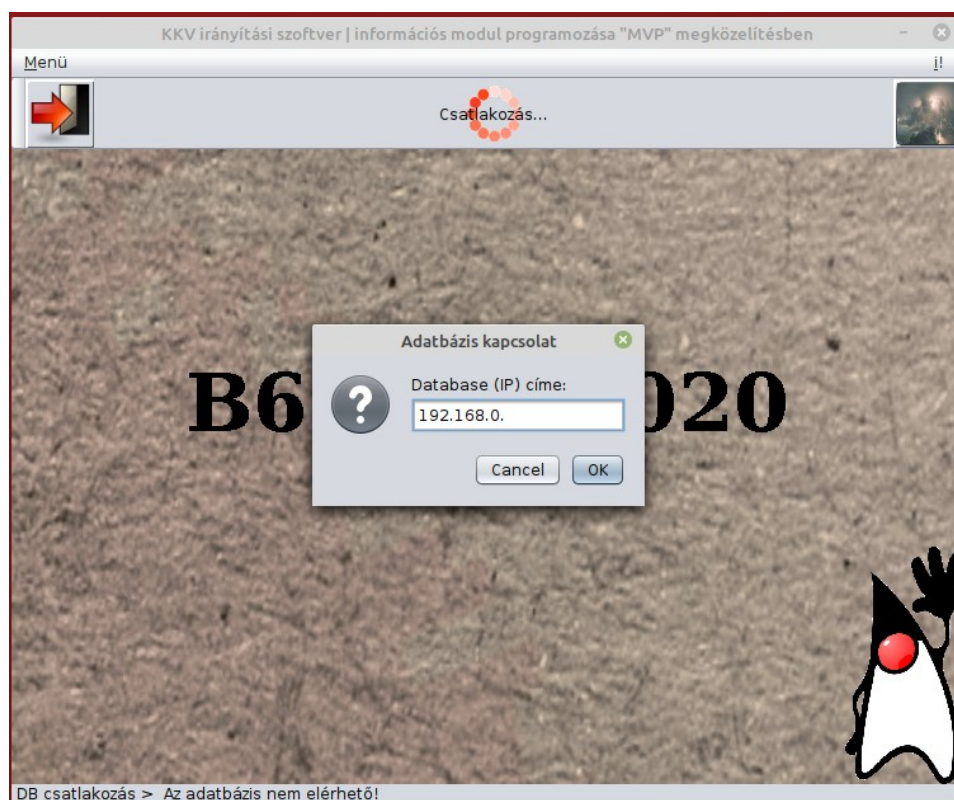
Az eszköztáron található gombok tárháza bővíthető, láthatóságuk az éppen elérhető funkciók szerint változtatható.

A „kilépés” és „kijelentkezés” gombok megjelenítése a végleges felületen indokoltnak látszik, hiszen ezt a két funkciót az átlagos felhasználó is használni fogja.

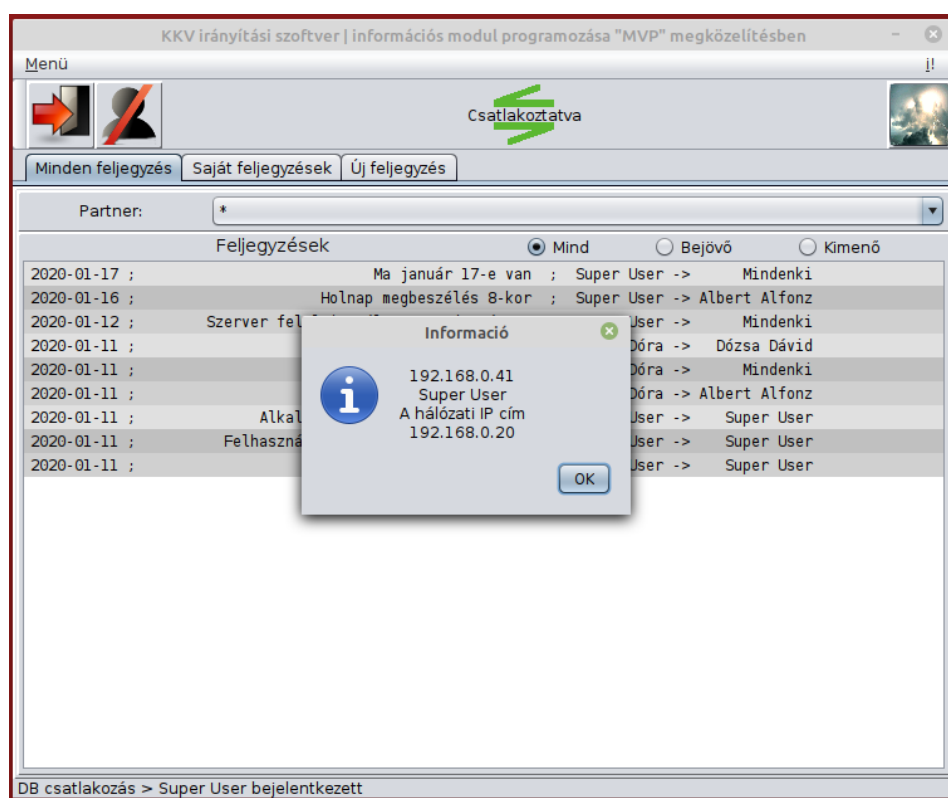
A „csatlakozás” és „felhő” gombok tesztüzemben mindenképpen hasznosak, bár az átlagos felhasználó érdeklődése nyilván nem terjed ki az általuk nyújtott információkra. Az előbbi ellenőrzi és jelzi az adatbázishoz történő csatlakozás folyamatát és a sikeres csatlakozás tényét. Az utóbbi egy sikertelen csatlakozási folyamat esetén lehetőséget ad a szerver IP-címének közvetlen megadására (11. ábra), illetve sikeres csatlakozás esetén információkkal szolgál a kapcsolatról (12. ábra).

A program jelenlegi verziójának eszköztára jelenleg tartalmazza a fent fölso-  
rolt gombokat.





11. Ábra: A szerver IP-címének megadása



12. Ábra: Kapcsolati adatok

### 1.4.5.3 Az állapotjelző sáv

Az állapotjelző sávban a folyamatban lévő tevékenységgel kapcsolatos üzeneteket jeleníti meg a program.

### 1.4.6 Fejlesztési lehetőségek

A fejlesztések a grafikus felületekbe épített új funkciók könnyebben használhatóvá tehetik az alkalmazást, az új modulok hozzáfűzésével pedig a program hasznosságát fokozzák.

Grafikus fejlesztési lehetőségekre példák:

- Az üzenetek megtekintése panelhez radiogomb hozzáfűzése a saját magunknak címzett üzenetek kiválaszthatóságára.
- A teljes üzenet megtekintése ne külön felugró ablakban, hanem harmonika szerűen nyíljon meg a fókuszba került üzenet az üzenet listában.
- Keresés lehetőségének a beépítése az üzenetek panelbe.

Modul fejlesztési lehetőségek:

- A „Super User” felhasználóként belépett felhasználó egy külön panelen adminisztrálhassa a felhasználókat. Tehát legyen joga és lehetősége új felhasználó felvételére, egy felhasználó adatainak és felhasználói csoportjának módosítására, valamint egy felhasználó törlésére.
- Eszköz nyilvántartási modul hozzáfűzése az alkalmazáshoz. Ez az adatbázisban új felhasználói csoporto(k) hozzáfűzését, új adattáblák kialakítását jelenti. Új SQL lekérdezéseket kell összeállítani az eszközök adatainak az eléréséhez és módosításához. Új panelek kialakítása szükséges, melyeken keresztül a hozzáféréssel rendelkező felhasználók kezelni tudják az adatbázisban az eszközök adatait.

A fejlesztői dokumentációban az eszköz nyilvántartási modul hozzáfűzését megvalósító lépéseket részletesebben is kifejtem.

- További modulok hozzáfűzése (Lásd a 2. oldal tetején látható listát).

## 2 Fejlesztői dokumentáció

### 2.1 Témaválasztás

Egy vállalkozás sikerességének egyik fontos feltétele, hogy működése fennakadás mentes legyen. Ezt a dolgozók közötti kommunikáció minősége nagy mértékben befolyásolja. Ezért lényeges, hogy a feladatok megszabásához, a megoldások ismertetéséhez, a közös munka végzéséhez és az eredmények értékeléséhez rendelkezzenek egy közös módszerrel. Ezt a módszert képezheti le egy számítógépes program, mely egy projekttel kapcsolatosan az összes utasítást és magyarázatot, az egyedi véleményeket, a megszerzett ismereteket és az elvégzett tevékenységeket egy helyen tárolja és elérhetővé teszi.

Egy ilyen alkalmazás biztosítja a megvalósítás nyomon követhetőségét és a befejezés utáni kiértékelést is jelentősen megkönnyíti. Az adatbázis „napra kész” állapota miatt az esetleg kieső résztvevők könnyebben helyettesíthetők.

Természetesen ez a programként fejlesztett módszer is csak akkora mértékben hatékony, amekkora mértékben alkalmazzák. A felhasználóknak maguknak is fel kell ismerniük a tájékoztatási rendszer hasznosságát, el kell fogadniuk az alkalmazását és folyamatosan használniuk is kell az általa nyújtott lehetőségeket.

A kapcsolattartás sokféle lehetőségét kínálják a különböző közösségi hálózatok és levelező programok. Ezek a megoldások azonban nem alkalmasak vállalati környezetben a feladatvégzéssel kapcsolatos információk biztonságos megosztására. Erre csak a vállalat saját informatikai környezetében működő zárt hálózati rendszer lehet alkalmas.

A felsorolt gondolatok és érvelések figyelembevételével fogalmazódott meg a kisvállalati információs rendszer gerincének az ötlete. Az információs rendszer alap változata egy levelezésszerű kapcsolattartási megoldást kínál. A felhasználók a programban feljegyzések segítségével dokumentálják tevékenységüket. Segítségével a munkatársak könnyen hozzájutnak az ismert adatokhoz és a naprakész információk alapján gyorsan és helyesen tudnak dönteni.

A programot úgy terveztem, hogy új igények felmerülése esetén egyszerűen módosítható és bővíthető legyen.

## 2.2 Az alkalmazott fejlesztői eszközök

A program fejlesztéséhez a JAVA programozási nyelvet választottam. A JAVA programok a legnépszerűbb operációs rendszereken futtathatók, így platform függetlennek mondhatók.

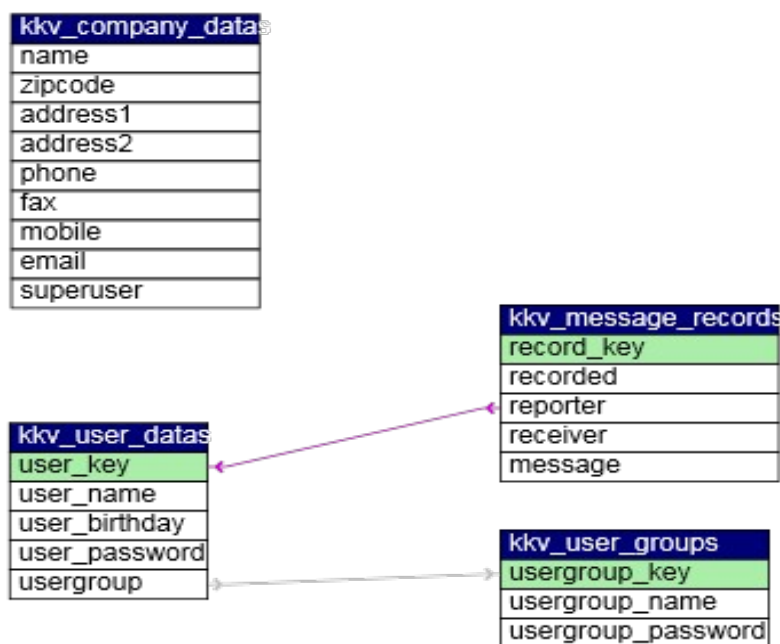
A fejlesztéshez szükséges környezetet egy Virtualboxban futó Windows 7 operációs rendszeren alakítottam ki. Ebbe telepítettem a JAVA kód fejlesztéséhez a NetBeans IDE 8.2-es keretrendszert és az adatbázis kezeléshez a szabad és nyílt forrású platformfüggetlen webservert-szoftvercsomagot, az XAMPP-t.

A képek, ikonok és ábrák készítéséhez és módosításához a GIMP képszerkesztő programot együtt használtam a rendszer saját képernyőkép készítőjével. A folyamatábrák esetén a Dia program lehetőségeit aknáztam ki. Az adatmodell képeit a phpMyAdmin adatbázis-kezelő „Designer” menüpontjából exportáltam.

## 2.3 Az adatmodell leírása

### 2.3.1 Az adatbázis

Az alkalmazás a `kkv_sample` nevű adatbázist használja az adatok tárolására. A bemutatott változatban négy tábla található (13. ábra).



13. Ábra: A `kkv_sample` adatbázis táblái

### 2.3.2 A vállalat adatai tábla

A `kkv_company_data` tábla (14. ábra) a kisvállalat adatait tartalmazza. Az oszlopok tartalma egyértelmű (név, irányítószám, cím1, cím2, telefon, fax, mobil, email), egyedül a `superuser` oszlop szorul magyarázatra. A vállalat adatait módosítani jogosult felhasználói csoport (`kkv_user_groups`) kulcsát (`usergroup_key`) adja meg.

```
/* COMPANY DATAS -----*/

/* KKV SAMPLE DATABASE 2019-2020 */
/* KKV COMPANY DATAS: */
/* NAME, ADDRESS, PHONES, MAIL */
CREATE TABLE `kkv_company_datas` (
  name VARCHAR(60) NOT NULL,
  zipcode INT NOT NULL,
  address1 VARCHAR(60) NOT NULL,
  address2 VARCHAR(60) NOT NULL,
  phone VARCHAR(20) NOT NULL,
  fax VARCHAR(20) NOT NULL,
  mobile VARCHAR(20) NOT NULL,
  email VARCHAR(60) NOT NULL,
  superuser INT DEFAULT 2
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_hungarian_ci;
```

14. Ábra: A `kkv_company_data` tábla szerkezete

### 2.3.3 A felhasználói csoportok tábla

A `kkv_user_groups` tábla (15. ábra) az alapja a felhasználók jogosultsági besorolásának. A csoport név (`usergroup_name`) egyben az adatbázis eléréséhez szükséges felhasználónév, míg a csoport jelszó (`usergroup_password`) az adatbázis eléréséhez szükséges belépési jelszó.

```
/* USEGROUPS -----*/

/* KKV SAMPLE DATABASE 2019-2020 */
/* KKV USERGROUPS: */
/* USER PRIVILEGES AND MYSQL GRANTS */
CREATE TABLE `kkv_user_groups` (
  usergroup_key INT AUTO INCREMENT,
  usergroup_name VARCHAR(40) NOT NULL,
  usergroup_password varchar(255) NOT NULL,
  CONSTRAINT pk_usergroup PRIMARY KEY (usergroup_key)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_hungarian_ci;
```

15. Ábra: A `kkv_user_groups` tábla szerkezet

### 2.3.4 A felhasználók adatai tábla

A `kkv_user_datas` tábla (16. ábra) az adatbázis elérésére jogosult felhasználók adatait tartalmazza. A név (`user_name`) és születési dátum (`user_birthday`) egyértelmű-

en meghatározza a felhasználót. A jelszó (user\_password) az alkalmazásba – és *nem* az adatbázisba - történő belépéshez szükséges. A felhasználói csoport (usergroup) pedig megadja, hogy melyik csoport tagjaként jogosult az adatbázist használni. A sorban az első felhasználó a „BroadCastUser”, a neki címzett feljegyzéseket, mint körüzeneteket, mindenki láthatja.

```
/*          USERS -----*/

/* KKV SAMPLE DATABASE 2019-2020 */
/* KKV USER DATAS */
/* NAME, BIRTHDAY, USERGROUP */
CREATE TABLE `kkv_user_datas` (
  user_key INT AUTO INCREMENT,
  user_name VARCHAR(40) NOT NULL,
  user_birthday DATE NOT NULL,
  user_password varchar(255) NOT NULL,
  usergroup INT,
  CONSTRAINT pk_user PRIMARY KEY (user_key, user_name)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_hungarian_ci;

/* SET FOREIGN KEY IN KKV USER DATAS */
ALTER TABLE `kkv_user_datas` ADD FOREIGN KEY(usergroup)
REFERENCES `kkv_user_groups` (usergroup_key);
```

16. Ábra: A kkv\_user\_datas tábla szerkezete

### 2.3.5 A feljegyzések tábla

A kkv\_message\_records tábla (17. ábra) tartalmazza az írás idejét (recorded), az író (reporter), a címzettet (reciever) és a feljegyzést (message).

```
/*          RECORDS - MESSAGES -----*/

/* KKV SAMPLE DATABASE 2019-2020 */
/* KKV RECORDS */
/* TOOL, QUANTITY */
CREATE TABLE `kkv_message_records` (
  record_key INT AUTO INCREMENT,
  recorded DATETIME DEFAULT CURRENT_TIMESTAMP,
  reporter INT NOT NULL,
  receiver INT,
  message TEXT,
  CONSTRAINT pk_comment PRIMARY KEY (record_key)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_hungarian_ci;

/* SET FOREIGN KEY IN KKV COMMENT */
ALTER TABLE `kkv_message_records` ADD FOREIGN KEY(reporter)
REFERENCES `kkv_user_datas` (user_key);
```

17. Ábra: A kkv\_message\_records tábla szerkezete

### 2.3.6 A felhasználók

Az adatbázis használatához különböző hozzáféréssel rendelkező csoportok jogosultak.

Az adatbázis biztonságát szem előtt tartva létre kell hozni egy „root” szintű adminisztrátor felhasználót, majd törölni kell a telepítés során alapértelmezésben ballított jelszó nélkül root felhasználó valamennyi példányát.

Az információs rendszer adminisztrátorát „rootschiold” -nak kereszteltem és minden hálózatról teljes hozzáférést biztosítottam neki (22. ábra).

```
GRANT ALL PRIVILEGES ON *.* TO 'rootschiold'@ '%' IDENTIFIED BY PASSWORD
'*609F49858EF550B6556A011E4E551F69383453B5' WITH GRANT OPTION;

GRANT ALL PRIVILEGES ON *.* TO 'rootschiold'@'127.0.0.1' IDENTIFIED BY PASSWORD
'*609F49858EF550B6556A011E4E551F69383453B5' WITH GRANT OPTION;

GRANT ALL PRIVILEGES ON *.* TO 'rootschiold'@'::1' IDENTIFIED BY PASSWORD
'*609F49858EF550B6556A011E4E551F69383453B5' WITH GRANT OPTION;

GRANT ALL PRIVILEGES ON *.* TO 'rootschiold'@'localhost' IDENTIFIED BY PASSWORD
'*609F49858EF550B6556A011E4E551F69383453B5' WITH GRANT OPTION;
```

18. Ábra: A "rootschiold" felhasználói csoport

Az információs rendszer adatbázis elérhetőségét a programban a „testuser” hozzáféréssel teszteltem. Ez a felhasználói tesztcsoport rendelkezik a legalacsonyabb jogosultságokkal (23. ábra), de minden hálózatról hozzáféréssel bír.

```
GRANT USAGE ON *.* TO 'testuser'@ '%' IDENTIFIED
BY PASSWORD '*3A2EB9C80F7239A4DE3933AE266DB76A7846BCB8';
GRANT SELECT (user_password, user_name, usergroup, user_key)
ON `kkv_sample`.`kkv_user_datas` TO 'testuser'@ '%';

GRANT USAGE ON *.* TO 'testuser'@'127.0.0.1' IDENTIFIED
BY PASSWORD '*3A2EB9C80F7239A4DE3933AE266DB76A7846BCB8';
GRANT SELECT (user_password, user_name, usergroup, user_key)
ON `kkv_sample`.`kkv_user_datas` TO 'testuser'@'127.0.0.1';

GRANT USAGE ON *.* TO 'testuser'@'::1' IDENTIFIED
BY PASSWORD '*3A2EB9C80F7239A4DE3933AE266DB76A7846BCB8';
GRANT SELECT (user_password, user_name, usergroup, user_key)
ON `kkv_sample`.`kkv_user_datas` TO 'testuser'@'::1';

GRANT USAGE ON *.* TO 'testuser'@'localhost' IDENTIFIED
BY PASSWORD '*3A2EB9C80F7239A4DE3933AE266DB76A7846BCB8';
GRANT SELECT (user_password, user_name, usergroup, user_key)
ON `kkv_sample`.`kkv_user_datas` TO 'testuser'@'localhost';
```

19. Ábra: A "testuser" felhasználói csoport



A következő csoport az „employee” alkalmazotti csoport, mely a feljegyzések írására és a neki szóló feljegyzések olvasására jogosult (24. ábra). Az ebbe a csoportba tartozó felhasználók csak helyi hálózatról, azaz munkahelyi gépről léphetnek be.

```
GRANT USAGE ON *.* TO 'employee'@'localhost' IDENTIFIED
  BY PASSWORD '*334AD31E3FB6986ED62F9C8D2F252E2FC475D5FA';
GRANT SELECT, INSERT
  ON `kkv_sample`.`kkv_message_records` TO 'employee'@'localhost';
```

20. Ábra: A "employee" felhasználói csoport

A ranglétra legmagasabb fokán álló „director” irányító csoport betekinthez az összes feljegyzésbe és írhat saját feljegyzéseket (25. ábra). Az ebbe a csoportba tartozó felhasználók csak helyi hálózatról, azaz munkahelyi gépről léphetnek be.

```
GRANT USAGE ON *.* TO 'director'@'localhost' IDENTIFIED
  BY PASSWORD '*1ABF1A3C7938A769F31C95E3D5CFC2FA2007598';
GRANT SELECT (user_name, user_birthday, usergroup, user_key),
  INSERT ON `kkv_sample`.`kkv_user_datas` TO 'director'@'localhost';
GRANT SELECT (reporter, message, receiver, recorded)
  ON `kkv_sample`.`kkv_message_records` TO 'director'@'localhost';
```

21. Ábra: A "director" felhasználói csoport

A rendszer majdnem adminisztrátora(i) a superuser(ek). A csoport (26. ábra) jogosult a felhasználói adatok módosítására és ezen felül rendelkezik az irányító csoport jogaival is.

```
GRANT USAGE ON *.* TO 'superuser'@'%' IDENTIFIED
  BY PASSWORD '*4FB15991C85A2EB8A3B4B2D3411AA2AD094E7229';
GRANT SELECT, INSERT ON `kkv_sample`.* TO 'superuser'@'%';
GRANT UPDATE (zipcode, address1, fax, email, phone, mobile, name, address2)
  ON `kkv_sample`.`kkv_company_datas` TO 'superuser'@'%';
GRANT UPDATE (user_password, user_birthday, usergroup, user_name)
  ON `kkv_sample`.`kkv_user_datas` TO 'superuser'@'%';
GRANT UPDATE (usergroup_name)
  ON `kkv_sample`.`kkv_user_groups` TO 'superuser'@'%';
GRANT SELECT, INSERT
  ON `kkv_sample`.`kkv_message_records` TO 'employee'@'localhost';

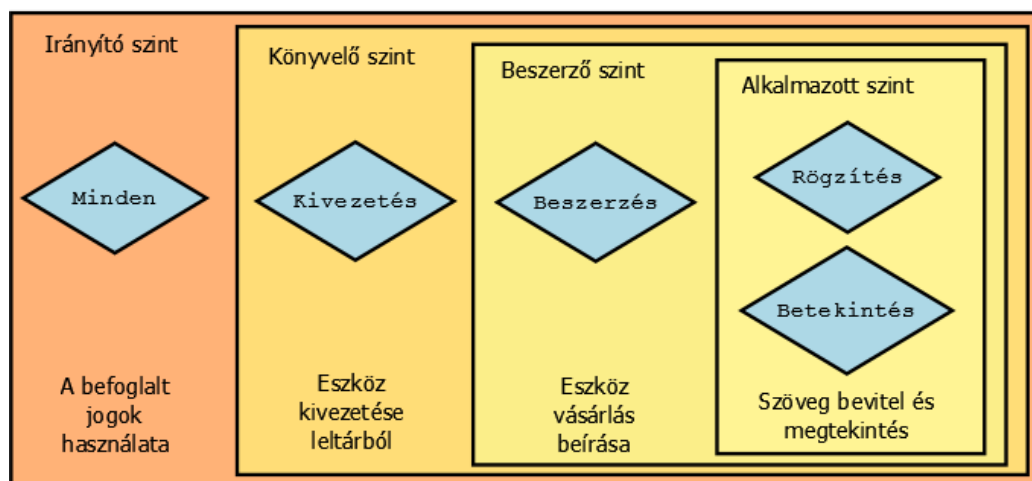
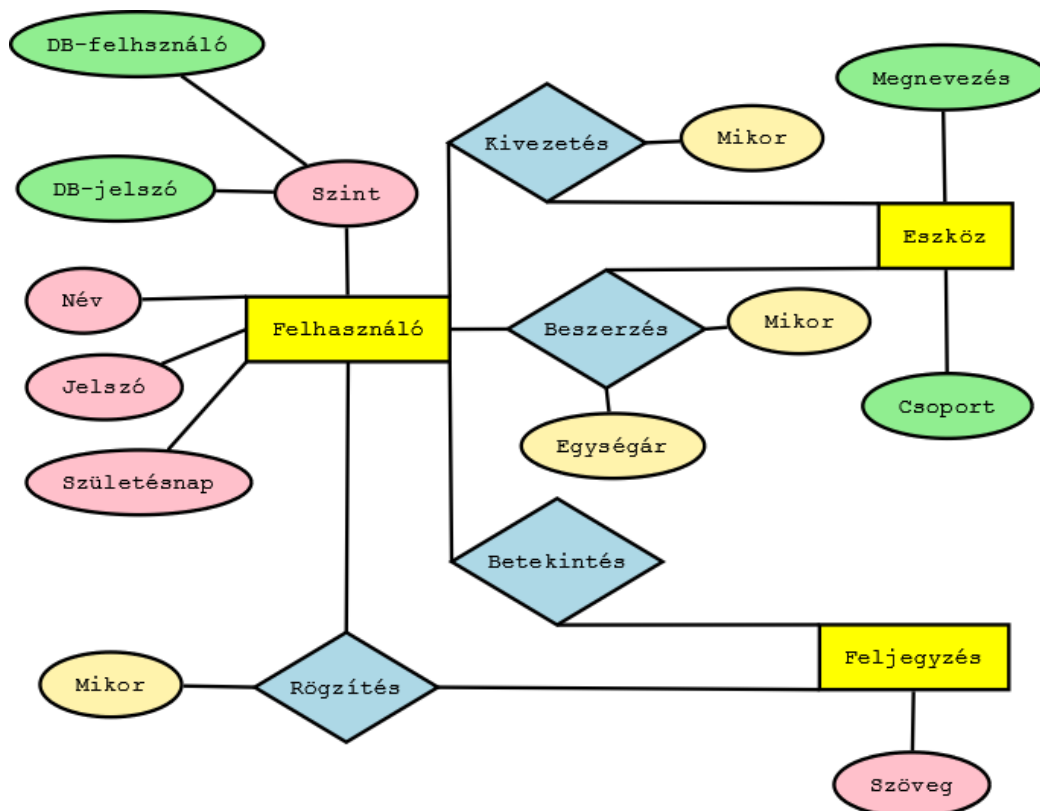
GRANT USAGE ON *.* TO 'superuser'@'localhost' IDENTIFIED
  BY PASSWORD '*4FB15991C85A2EB8A3B4B2D3411AA2AD094E7229';
GRANT SELECT, INSERT ON `kkv_sample`.* TO 'superuser'@'localhost';
GRANT UPDATE (zipcode, address1, fax, email, phone, mobile, name, address2)
  ON `kkv_sample`.`kkv_company_datas` TO 'superuser'@'localhost';
GRANT UPDATE (user_password, user_birthday, usergroup, user_name)
  ON `kkv_sample`.`kkv_user_datas` TO 'superuser'@'localhost';
GRANT UPDATE (usergroup_name)
  ON `kkv_sample`.`kkv_user_groups` TO 'superuser'@'localhost';
GRANT SELECT, INSERT
  ON `kkv_sample`.`kkv_message_records` TO 'employee'@'localhost';
```

22. Ábra: A "superuser" felhasználói csoport



### 2.3.7 Az adatbázis fejlesztése

Az eszköznyilvántartó modullal történő fejlesztése után az adatbázis szerkezetét, a kapcsolatokat és az egységeket a 21. ábrán látható E-K diagram szemlélteti. Az ábra alsó részén az egymásba ágyazott kapcsolati elemek a felhasználói szintek rangsor szerinti végrehajtási jogait mutatják az irányítói szinttel bezárólag.



23. Ábra: A bővített adatbázis E-K diagramja és a hozzáférési besorolások

A fejlesztés során tehát három táblázattal és két felhasználói csoporttal bővül az adatbázis. Az új csoportokhoz fűződő a jogosultságokat a fejlesztés során az irányító és superuser felhasználók is megkap(hat)ják.

### 2.3.7.1 Az eszköz csoportok tábla

A `kkv_tool_groups` tábla (18. ábra) az eszközökhöz rendelhető csoportneveket (`toolgroup_name`) tartalmazza.

```
/* TOOLGROUPS -----*/
/* KKV SAMPLE DATABASE 2019-2020 */
/* KKV TOOLGROUPS: */
/*
CREATE TABLE `kkv_tool_groups` (
  toolgroup_key INT AUTO_INCREMENT,
  toolgroup_name VARCHAR(255) NOT NULL,
  CONSTRAINT pk_toolgroups PRIMARY KEY (toolgroup_key)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_hungarian_ci;
```

24. Ábra: A `kkv_tool_groups` tábla szerkezete

### 2.3.7.2 Az eszközök adatai tábla

A `kkv_tool_datas` tábla (19. ábra) a nyilvántartásba vehető eszközök adatait tárolja. Az eszköz nevét, csoportját és a minimális darabszámot.

```
/* TOOLS -----*/
/* KKV SAMPLE DATABASE 2019-2020 */
/* KKV TOOL DATAS */
/* NAME, TOOLGROUP */
CREATE TABLE `kkv_tool_datas` (
  tool_key INT AUTO_INCREMENT,
  tool_name VARCHAR(255) NOT NULL,
  toolgroup INT,
  tool_min_quantity INT,
  CONSTRAINT pk_tool PRIMARY KEY (tool_key)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_hungarian_ci;
```

25. Ábra: A `kkv_tool_datas` tábla szerkezete

### 2.3.7.3 Az eszköznyilvántartás tábla

A `kkv_tool_records` tábla (20. ábra) adataiban a nyilvántartási folyamatot tükrözteti. A nyilvántartásba vétel és selejtezés dátumát, a könyvelőre mutató kulcsot, az eszközre utaló kulcsot, a beszerzési árat és egy megjegyzést foglal egy adatsorba.

```
/*                                RECORDS - TOOLS -----*/

/* KKV SAMPLE DATABASE 2019-2020      */
/* KKV RECORDS                        */
/* TOOL, QUANTITY                      */
CREATE TABLE `kkv_tool_records` (
  record_key INT AUTO INCREMENT,
  recorded DATETIME DEFAULT CURRENT_TIMESTAMP,
  rollout DATE,
  reporter INT NOT NULL,
  tool INT,
  price INT,
  notes TEXT,
  CONSTRAINT pk_comment PRIMARY KEY (record_key)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_hungarian_ci;
```

26. Ábra: A kkv\_tool\_records tábla szerkezete

### 2.3.7.4 Az új felhasználó csoportok

Az eszköznyilvántartó modul két felhasználói csoporttal fejleszti az adatbázist. Ez a két csoport a beszerzői – shopper – csoport (27. ábra) és az eszközök nyilvántartásával foglalkozó könyvelői – accountant – csoport (28. ábra).

```
CREATE USER IF NOT EXISTS 'shopper'@'localhost' IDENTIFIED BY 'shoppermysql';
GRANT SELECT
  ON `kkv_sample`.`kkv_tool_records` TO 'shopper'@'localhost';
GRANT INSERT (`recorded`, `reporter`, `tool`, `price`, `notes`)
  ON `kkv_sample`.`kkv_tool_records` TO 'shopper'@'localhost';
GRANT SELECT (`recorded`, `reporter`, `receiver`, `message`)
  ON `kkv_sample`.`kkv_message_records` TO 'shopper'@'localhost';
```

27. Ábra: A "shopper" felhasználói csoport

```
CREATE USER IF NOT EXISTS 'accountant'@'localhost' IDENTIFIED BY 'accountantmysql';
GRANT SELECT (`user_key`, `user_name`, `user_birthday`, `usergroup`), INSERT
  ON `kkv_sample`.`kkv_user_datas` TO 'accountant'@'localhost';
GRANT SELECT, INSERT
  ON `kkv_sample`.`kkv_tool_datas` TO 'accountant'@'localhost';
GRANT SELECT, INSERT
  ON `kkv_sample`.`kkv_tool_records` TO 'accountant'@'localhost';
GRANT SELECT (`recorded`, `reporter`, `receiver`, `message`)
  ON `kkv_sample`.`kkv_message_records` TO 'accountant'@'localhost';
```

28. Ábra: Az "accountant" felhasználói csoport

Ezek a csoportok az alkalmazotti csoport jogosultságain felül az eszközökhöz fűződő adatok bejegyzésére és egyes adatok módosítására is jogosultak.

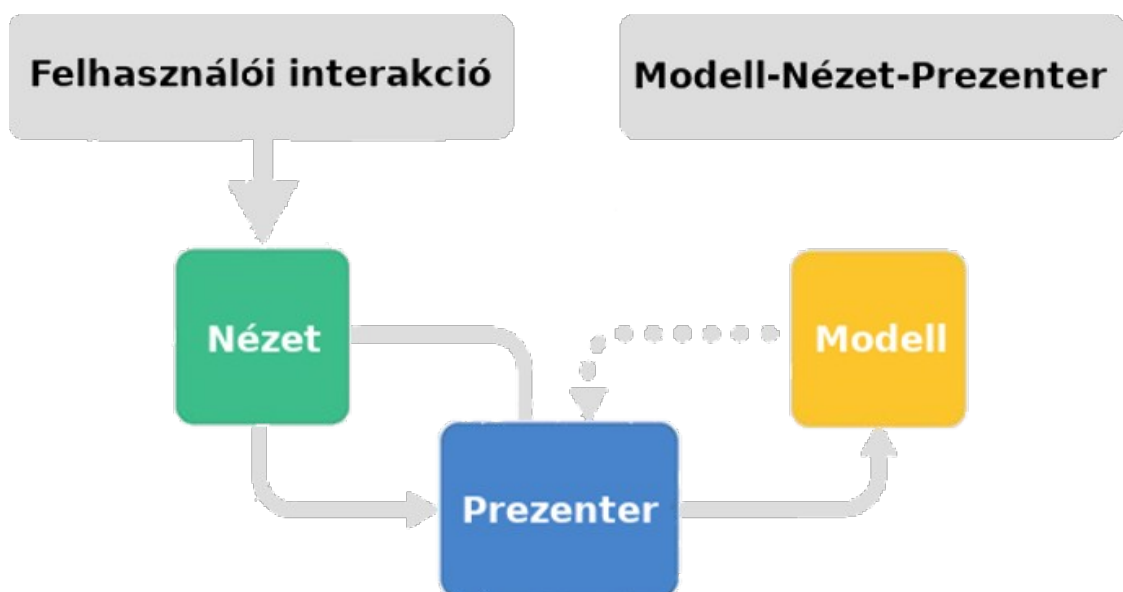
## 2.4 Részletes feladat-specifikáció, algoritmusok

A programozás során a Modell-Nézet-Prezenter szoftvertervezési mintát alkalmaztam (angolul: Modell-View-Presenter, röviden: MVP), mivel ez felelt meg leg-

jobban a feltételként támasztott modulos bővíthetőségnek. A minta rétegeinek jól körülhatárolt szerepe segíti a kód nyomkövethetőségét.

Az MVP-modell főleg olyan alkalmazásokban nyújt jelentős előnyöket – hasonlóan az MVC-hez –, ahol komplex adathalmazokon kell műveleteket végezni, és ezek eredményeit a felhasználó elé tárni. Az MVP modell három rétegre osztja az alkalmazást. Mindhárom rétegnek jól körülírható feladatai vannak, és csak a szomszéd réteggel kommunikálhatnak. Ez nagyfokú rugalmasságot ad: a két szélső komponens egymás tudta nélkül bármikor lecserélhető.<sup>2</sup>

Az MVP minta három rétege közötti kommunikációt a 29. ábra mutatja be.



29. Ábra: A Modell-Nézet-Prezenter szoftverfejlesztési minta

Az alkalmazást három fő részre osztottam. A bevezető rész ellenőrzi az adatbázis elérhetőségét. A fő programrész a felhasználó bejelentkezésével indít, majd a felhasználó azonosítása után a felhasználóhoz tartozó munkafelületet jeleníti meg és a felhasználó interakciói alapján visszajelzést nyújtva elvégzi a kívánt adatbázissal kapcsolatos műveleteket. A harmadik résszel lezárjuk a programot.

Az osztályoknál és módszereiknél a specifikációja (mit valósít meg az adott módszer, milyen paraméterei vannak, milyen értéket ad vissza).

Az algoritmizálható részek leírása valamilyen algoritmus-leíró eszközzel (struktogram, pszeudo-kód, folyamatábra).

Ajánlott terjedelem: a feladat jellegétől függően 3-4 oldal.

<sup>2</sup> [Részletek a Wikipédiából](#)

## 2.5 Forráskód

A teljes forráskódot a nyomtatott dokumentációba nem kell beletenni.

Lehet viszont a nyomtatott dokumentációban a fontosabb kódrészeket magyarázattal szerepeltetni

## 2.6 Tesztelési dokumentáció

Különböző tesztesetek részletes bemutatása:

- különböző felhasználó tevékenységek esetén hogyan reagált a program
- milyen üzeneteket kaptunk
- mi a teendő az egyes üzenetek esetében

Normál teszteset, extrém teszteset (bolondbiztosság tesztelése)

A tesztelés során kiderült hibák felsorolása

A tesztelési dokumentációból derüljön ki, hogy ismered a különböző tesztelési módszereket (pl. fekete doboz, fehér doboz módszer)

Ajánlott terjedelem: a feladat jellegétől függően 2-3 oldal.

## 2.7 Továbbfejlesztési lehetőségek

Olyan ötletek, amelyeket meg szeretnél valósítani, de nem sikerült, vagy nem fért bele az időbe.

Olyan ötletek, amelyeket még érdemes a jövőben megvalósítani

Ajánlott terjedelem: 1-2 oldal

## 2.8 Irodalomjegyzék, forrásmegjelölés

Minden olyan forrás pontos megadása, amelyet a szakdolgozatodban felhasználtál.

A forrás lehet pl.

Könyv. Meg kell adnod a következőket: szerző(k), cím, kiadó, kiadás éve.

Weboldal. Meg kell adnod a linket, az oldal címét

Elektronikus dokumentum. Meg kell adnod a szerzőt és a letöltés helyét

Ha a szakdolgozatban valamely forrásból szó szerint idézel, akkor a megfelelő szövegrészt idézőjelbe kell tenni, és lábjegyzetben meg kell jelölnöd az idézet forrását.

Ajánlott terjedelem: 1/2 -1 oldal.