



SOTER-LINE  
OKTATÁSI KÖZPONT

# Soter-Line Oktatási, Továbbképző és Szolgáltató Kft.

1073 Budapest, Erzsébet krt. 7. Tel: 887-2323; Fax: 887-2324, [www.soterline.hu](http://www.soterline.hu)

Nyilvántartásba vételi szám: E-000201/2014

## ZÁRÓDOLGOZAT

.....

Ügyfélnyilvántartó rendszer

Szoftverfejlesztő tanfolyam

OKJ száma: 54 213 05

Témavezető:

Kerecsen László.....

Készítette:

Kerecsen László.....

# Tartalomjegyzék

<b>CÉLJA A SZOFTVERNEK:</b> .....	<b>3</b>
<b>TERVEZÉS:</b> .....	<b>5</b>
ADATBÁZIS TERVEK:.....	5
FELÜLETI TERVEK:.....	6
<b>FEJLESZTÉS:</b> .....	<b>11</b>
<b>TESZTELÉSEK:</b> .....	<b>14</b>
<b>BEFEJEZÉS:</b> .....	<b>15</b>
<b>KÉPEK, ILLUSZTRÁCIÓK:</b> .....	<b>16</b>

## Célja a szoftvernek:

Amit most készíteni fogok az egy olyan szoftver ahol el lehet tárolni az ügyfelek elérhetőségeit nevet, telefonszámot, email címet. Szerintem elég sok cég küszködik azzal a problémával hogy nagy ügyféltörzsszel kell tartani a kapcsolatot és a telefonban lévő elérhetőségek már átláthatatlanok. Általában a felét ritkán használják a negyedét pedig szinte évente egyszer kétszer. Na ezek az elérhetőségek szoktak törlésre kerülni mikor már teljes a káosz a telefonkönyvünkben. De mielőtt ezt megteesszük általában rá vagyunk kötelezve egy mentésre. Vagy manuálisan felírjuk egy noteszba, vagy Excel táblázatba ami mind a kettő elég átláthatatlan. Vagy valami más bonyolult szoftverbe begépeljük. Ez ismerős mindenkinek kikeresni a partnert, új kapcsolat begépelni a nevet a telefonszámot ez macerás. És ha még gépelés közben el is rontottam a számot soha többet nem tudom felhívni vagy csak nagy utánajárással.

Utánajártam az interneten természetesen találtam hasonló programot és sokkal komplexebb programokat is, nagy része inkább hasonlított egy CRM rendszerre, de ennek a szoftveremnek nem ez a célja. Ez a szoftvernek inkább a kis és mikro vállalatok és egyéni vállalkozások lenne a célközönsége. Ami még az egyedisége, hogy nem találtam olyan rendszert amely támogatja azt a technológiát amit én szeretnék készíteni, használni amit lentebb ki is fejtek bővebben. Persze nem tartom kizártnak, de kutatásaim alatt nem találtam. Itt nincs elgépelési lehetőség, ha már hívtam az a telefonszámot, akkor csak mentenem kell a fontos adatok módosítása nélkül. Szinte mindent a telefon, a számítógép és a szoftver old meg helyettem.

Erre a problémára találtam ki, hogy létezik egy szabvány a névjegyeknél, amit minden telefon ismer, ámbár eléggé kihasználatlan ez a vCard. Ezt a szabványt felhasználva nincs más teendő csak lementeni a telefonomról a számokat neveket esetleg email címeket ebbe a (.vcf) kiterjesztésű fájlba csatlakoztatni a mobilt a PC-hez és már mehet is az adatok importálása a szoftverbe. Egy két telefonszámnál lehet nincs jelentősége időben, de 10-15 esetleg 20 telefonszámnál már félórát is meg lehet spórolni. De ugyanígy exportálni is lehet a programból.

Magából a szoftverből lehet keresni, szűrni Üzletnévre, cégnévre, vagy esetleg ügyvezető nevére, megbízott fő kapcsolatot tartóra, akinek ha több üzlete, cége van mindet kiadja a program. És még továbbá lekérdezéseket is lehet benne készíteni.

Maga a szoftver Windows alkalmazás lenne, és egy adatbázisba mentené le az adatokat.

# Tervezés:

## Adatbázis tervek:

Maga a rendszer elképzelésem szerint viszonylag sok összefüggő nem nagy terjedelmű adathalmazból fog állni. Így mindenféleképpen az adatok tárolásához valami adatbázisra van szükség. A választásom MySQL adatbázis szerverre esett. A legfőbb szempont az adatok biztonsága volt. Lehetett volna bármely más szerver, de ez volt az ami az igényeket kielégíti és könnyen elérhető akár éles akár teszt vagy fejlesztési környezetben. A cél az is volt hogy nem feltétlenül lokális adatbázist használjak hanem egy távoli szerveren lévő. Ez lehet egy web tárhely is akár.

Az adatbázis összesen hét táblából áll aminek nagy része összefügg egymással. A struktúrájában két különálló táblát lehet észrevenni az egyik a „users” tábla ami a bejelentkezési adatok tárolására szolgál. Úgy mint egy egyedi azonosító, felhasználónév, egy hozzá tartozó jelszó ami md5 titkosítással van eltárolva.

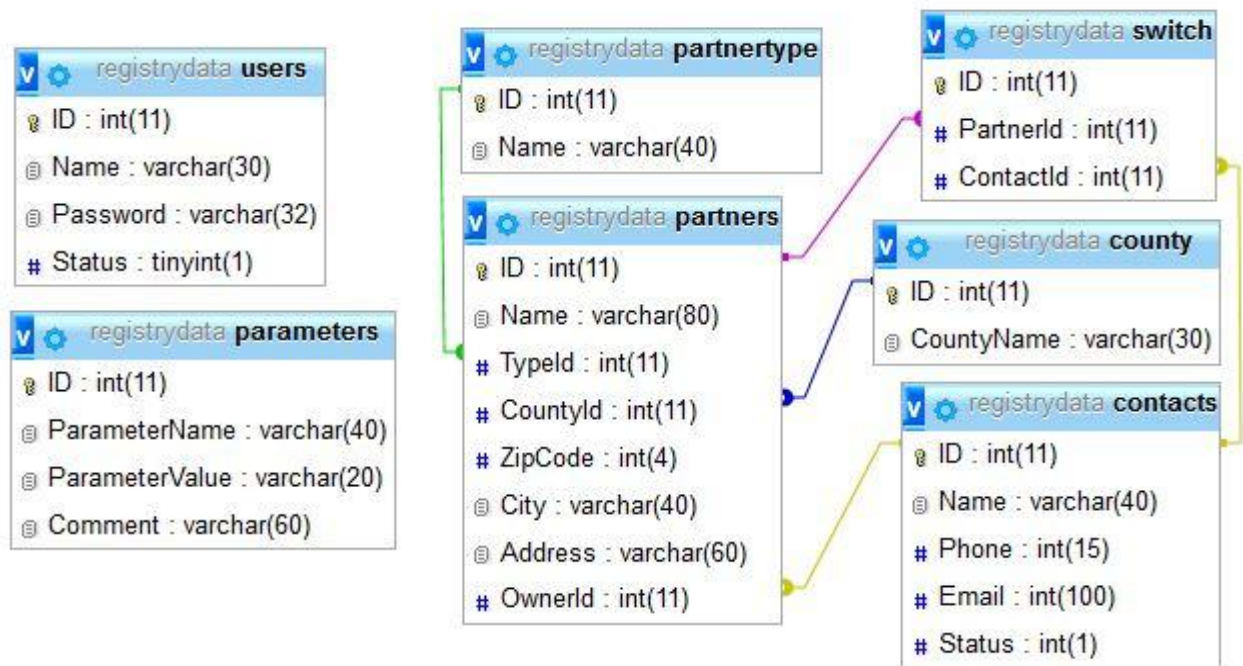
A másik külön álló tábla a „parameters” ami a lehetővé teszi az egyedi programbeállításokat. Itt is van egy egyedi azonosító, a paraméter neve, paraméter értéke, és van egy magyarázat.

Az összes többi tábla valamilyen másik táblával vagy tábláktól függ. Vegyük őket sorra. Az első a „county” itt Magyarország megyéi vannak eltárolva. Van még egy hasonló tábla a „partnertype” ami a partnertípusokat lehet megadni. Van még a két fő tábla a „partners” ami a partner cégek ügyfelek nevét, típusát, pontos címét, tulajdonosát, és megyét tudom eltárolni.

A másik fő tábla a „contacts” ahol konkrétan az ügyfeleket, személyek nevét, telefonszámát, és email – címét és státuszát tudom eltárolni.

Végére pedig a „switch” tábla maradt, aminek az a szerepe, hogy összerendelje a partnereket az ügyfelekkel.

Maga az adatbázis struktúra a következő képen látható.



## Felületi tervek:

Maga a felület egy Windows alkalmazás. A felület leírása WPF Window-ban történik. Ez a felület több szempontból is hasznos. Fejlesztői szempontból áttekinthető, jobban értelmezhető, teljesen dinamikus. XAML kódban lehet megírni a felületet. Felhasználói szempontból szebb modernebb külsőt lehet készíteni mint a régi Windows Form-os alkalmazásokban. Bár a szoftver nem egy teljes képernyős alkalmazás, nem igényli mert nincs rajta olyan sok input és output eszköz. De bizonyos ablakok méretének módosítása engedélyezett és a szoftver megőrzi eredeti kinézetét és alkalmazkodik a felhasználó által beállított új méretre, tehát egyszerűen ügyel a reszponzivitásra.

A program indításakor a „MainWindow” hívódik meg, itt helyezkedik el a menüsor. Alatta a gyorsmenü funkció található, ez a rész két gombot kapott ahonnan kényelmesen el tudom indítani a két legfontosabb részét a programnak, a partnereket és az elérhetőségeket. A gyorsmenü alatt pedig kiírja a program hogy melyik felhasználó jelentkezett be.

*A felület terv az alábbi képen látható*

UI design of a login window. The window has a title bar with a close button (X). Below the title bar is a menu bar with four 'Menü' items. The main area contains two green buttons labeled 'Gyors gomb'. At the bottom left, it says 'Bejelentkezett: Felhasználó'.

Mivel a program bejelentkezést kér ezért nem feltétlenül az előző képen látható felületet látja meg először a felhasználó hanem a „Login” ablakot. Itt kettő szöveg jelzi hogy a mellette lévő beviteli mezőbe mit is kell írni. A felső egy felhasználónevet kér, az alsó pedig egy jelszót. Mindezek alatt pedig egy gomb helyezkedik el ami a Bejelentkezés feliratot kapta.

*A felület terv az alábbi képen látható*

UI design of a login window. It features two input fields. The first is labeled 'Felhasználónév:' and the second is labeled 'Jelszó:'. Below the input fields is a blue button labeled 'Bejelentkezés'.

A következő felülete ami bemutatásra kerül az a” PartnersWindow” nevet kapta. Ennek a felületnek az egyedisége hogy nem csak egy funkciót tölt be egyszerre . Ez a felület hívódik meg ha a MainWindow-n akár a partnerekre akár az elérhetőségekre kattintunk. Természetesen a megfelelő adatokat hozza, de ezt majd később részletesen kifejtem a következő fejezetben. És van még egy funkciója ennek

az ablaknak, amikor majd kapcsolatokat szeretnénk hozzáadni a partnereinkhez ugyan ez az ablak lett felhasználva.

Az ablak kinézete pedig a következő. Itt már a cím is változik attól függően milyen szerepet tölt be az ablak. A nagy részét egy lista szerűen elrendezett felsorolás alkotja ahol ki tudom választani az elemeket. Ez alatt található egy beviteli mező ami a keresésre szolgál ezt jelzi a mellette lévő szöveg is. A jobb alsó sarokban található még két gomb, ami az egyik új partner felvételére a másik pedig a kiválasztott partner törlésére szolgál.

*A felület terv az alábbi képen látható*

The diagram shows a rectangular window with an orange border. The top-right corner has a pink button with a black 'X'. The main content area is white. At the bottom left, the word 'Keresés' is written in bold black text next to a white rectangular input field. To the right of the input field are two stacked green rectangular buttons.

Az előbb tárgyalt felületnél ha bármelyik elemen kiválasztjuk a megfelelő szerkesztési ablakok jönnek fel vagy akár az Új partner gomb megnyomására.



Először nézzük ahol a személyeket tudom felvinni. Ez a ContactFormWindow névre hallgat. Itt négy darab input mezőt lehet látni. Az elsőbe a személy neve megy a másodikba a telefonszáma itt csak számbevitel a megengedett utána az email cím és a legvégén pedig egy ComboBox- ban tudom kiválasztani a személy státuszát milyen titulusban dolgozik. Ezek alatt található két gomb a jobb oldali a mentés a másik pedig a Névjegy mentése. Ezzel a gombbal tudom a személy adatait olyan formába kimenteni amit bizonyos telefonok vagy akár az Outlook is tud értelmezni.

*A felület terv az alábbi képen látható*

The diagram illustrates the ContactFormWindow interface. It features a title bar with a close button (X). The main area contains four labeled input fields: 'Név:' (Name), 'Telefonszám:' (Phone Number), 'Emailcím:' (Email Address), and 'Státusz:' (Status). The 'Státusz:' field is a dropdown menu. Below these fields are two buttons: 'Névjegy mentés' (Save Name Card) and 'Mentés' (Save).

A másik űrlap ablak az a partnerek cégek felvételére, módosítására szolgál. Itt az ablak felső részén viszonylag sok beviteli mező található ezt mindet nem tárgyalnánk végig mindegyik mellé oda van írva mire való milyen adatot kell neki eltárolni. Az alsó részén ennek az ablaknak is található egy lista ez tárolja azokat az embereket akiket kapcsolatba lehet hozni ezzel a céggel. A lista alatt a bal oldalon található egy + és is – gomb ezzel tudom módosítani a listámat. Rajtuk kívül található még a nyomtatási kép és a mentés gomb amik magukért beszélnek szerintem. A nyomtatási viszonylag sok variációm van mit tehetek ezzel.

A felület terv az alábbi képen látható

X

Név:	<input type="text"/>	Megye:	<input type="text"/>
Partnertípus:	<input type="text"/>	Irsz:	<input type="text"/>
Tulajdonosneve	<input type="text"/>	Város	<input type="text"/>
		Cím:	<input type="text"/>

+

-

Nyomtatási kép

Mentés

## Fejlesztés:

A fejlesztés 0. lépése előtt már jó előre el kezdtem információt gyűjteni az interneten a megvalósítás legkritikusabb pontjairól. Amikor megtaláltam a lehetséges válaszokat erre, ezeket elmentve belevágtam a projekt elkészítésébe.

A fejlesztés 0. lépése a projekt struktúra elkészítése. Ez természetesen egy modern fejlesztői környezetben történt meg egy sablon által új projektet generálva elindult a munka. Ehhez a lépéshez még ide tartozik, hogy magát a projektnek előkészítettem egy Git repository-t és az érintetlen sablont fel is töltöttem a saját tárhelyemre.

Mindezek után nekiálltam az adatbázist megtervezni, először papír alapon hogy milyen relációkra és azon belül milyen attribútumokra lesz szükségem. Utána el is készítettem magát az adatbázis struktúrát. Ez a része könnyen ment hisz a jelenlegi munkámban eltöltött évek során szinte nap mint nap adatbázisokkal foglalkozom. Ezért is kezdtem a lényegi munkát ezzel, mert úgy gondolom, hogy egy jó szoftver alapja egy igényesen jól megtervezett adatbázis.

A következő lépésem az volt, hogy magához a projekthez hozzáadtam azokat a csomagokat amire szükségem lesz a fejlesztés során.

Ezt követően létrehoztam a projekt és az adatbázis közötti kapcsolatot. Ezt a hidat Entity Framework-el oldottam meg. Itt sem volt túl sok probléma, miután ezek az apró hibákat javítottam az Entity Framework gyönyörűen le is generálta nekem az adatbázis modelljeimet a projektbe.

A 3. lépés elkezdtem dolgozni a felületekre, mivel MVVM struktúrát használtam a projektben ezzel párhuzamosan létrehoztam a View Modelleket is. A felületek túl sok biznisz logikát nem tartalmaznak csak amennyi feltétlenül szükséges hozzá. Leginkább eseménykezeléseket tartalmaznak és az erre várt reakciók vannak elhelyezve. Elsőnek a fő ablakon kezdtem el dolgozni. Itt nem sok biznisz logika kellett a megvalósításhoz. Ennek az ablaknak a célja, hogy elnavigálja a felhasználót oda amit éppen szeretne csinálni. Egy menüt és a benne lévő elemeket tartalmazza és egy pár gyors gombot, első időkben még a bejelentkezés megvalósításáért sem felelt. Másodikként azt a felületet készítettem el ami több funkciót is betölt a program élete során. Ez az a felület amihez egyszerre több view modell is rendelve van attól függően

mit is csinál a felhasználó éppen. feleslegesnek gondoltam két vagy akár 3 szinte egyforma felületet létrehozni hisz csak az adatok megjelenítésére szolgál azt hogy milyen adatot jelenít meg majd a view modell megadja neki. Mindez után elkészültek az űrlapok is. Itt az egyikkel sok probléma nem volt hisz három adat megjelenítésére és fogadására szolgál. A másik kicsit problémásabb volt maga a partner űrlap. Leginkább az adatkötések miatt és a pontos adat megjelenítése miatt mert itt egyszerre több relációval vagyis több táblával kell neki egyszerre dolgozni. Itt jöttem rá hogy nem lesz elég csak adatbázis modellekkel és view modellekkel dolgoznom hanem kell egy köztes réteg. Itt hoztam létre mellé egy új projektet egy class liblay-t vagyis egy dll fájlt, hogy ez tartalmazza majd az adatbázis modelleket. Leginkább a módosítások azonnali frissítése miatt volt ez hasznos, de sok más is leegyszerűsödött tőle. Így már csak szinte a biznisz logikát az események pontos lekezelését az adatok validálását kellett tökéletesíteni. De ezzel a köztes modell segítségével már semmi gond nem volt.

Következett a névjegy fájl vagy is a Vcard feldolgozása itt sem volt semmi probléma. Mielőtt nekiálltam ezt elkészíteni körülnéztem és láttam, hogy van olyan csomag ami ezt tudja olvasni, egy percre nem gondolkodtam rajta úgy döntöttem hogy megírom inkább én saját magam és tudom úgy alakítani ezt a részét ahogy nekem kényelmes és végül is ennek a szoftvernek ez akar lenni az egyedisége, nem illik más munkájával dicsekedni.

Mikor ezzel a művelettel is készen voltam következett az egyik legkritikusabb része a szoftver elkészítésénél mert elterveztem, hogy nyomtatási képet is készít majd a program. Ehhez a Report tervezőt használtam fel. Igen csak hogy ennek van egy saját ReportViewer nevezetű felület eleme amit sajnos csak Windows Form-on lehet alkalmazni, és ráadásul az adatkötéseket sem lehet kivitelezni ezért itt meg kellett szakítanom az mvvm struktúrát és változók hivatkozásával beállítani a megfelelő értékeket. Az alap felület ugyan úgy Wpf Window maradt csak kapott egy Form Host-ot és erre került a Report megjelenítő elme.

A nyomtatási kép elkészültével az adatok különböző lekérdezéseivel kezdtem foglalkozni. Itt kellett hozzá a View Modellen kívül egy plusz segéd osztály amivel lényegesen leegyszerűsödött a fa struktúra megjelenítése. Itt is minden lekérdezésnél egy ablakot használok, de két külön pici View Modellt ami ugyanabba a .cs fájlba került.

Mivel már majdnem elkészült az összes felület már csak a felhasználókkal kapcsolatos ablakok nem készültek el. Tehát ezek következtek. Elsőként a Login vagyis a bejelentkezés kivitelezése készült el, itt újra elő került kicsit jobban a Main Window mivelhogy a program futtatásakor ez hívódik meg elsőként, de még mielőtt betöltődne az adatbázishoz fordul egy kérdéssel mert úgy lett tervezve a szoftver hogy a bejelentkezés nem kötelező vagyis paraméterezhető adatbázis szinten, de Felhasználó mindenféleképpen kell. Ha a beállítás azt az értéket adja vissza, hogy nem kell bejelentkezni automatikusan az alap Adminisztrátor felhasználó töltődik be amit a fő ablak alján látni is lehet. Másik esetben mikor kötelező a bejelentkezés még betöltés előtt meghívódik a Login felület és addig nem enged tovább míg nincs olyan felhasználó és jelszó páros ami valós. Itt is meg kellett szakítanom az mvvm struktúrát mert a C# kimondja hogy a jelszót direktben memóriában tárolni veszélyes ezért csak változó szinten érhető el a beviteli mező értéke és ezt tiszteletben is tartva azonnal egy hash-elt formátumba mentem már a memóriába és így dolgozom vele tovább. Másodikként a felhasználók űrlapja készült el, mivel a login felületnél már kész lett így sok mindent fel lehetett használni innen és pillanatok alatt elkészült.

## Tesztelések:

A szoftver fejlesztése közben folyamatosan volt tesztelve a különböző részekységek. A működése, az elvart étékek megvizsgálásával több munkaóra eltelt, hogy ezek valóban jól működjenek, de ezek mind manuális tesztek voltak.

Természetesen miután végeztem a kivitelezéssel nekiálltam saját teszteket írni a programhoz mert manapság egy szoftver fejlesztés elképzelhetetlen, hogy úgy legyen megírva, hogy nem legyen mögötte teszt. Erre azért is van szükség, mert később bármi módosítás történik a kódba a tesztek lefuttatásával rögtön lehet látni, hogy milyen kihatása van a szoftver más részeire. Igaz ez időigényes munka, de a későbbiek során rengeteg órát meg lehet vele spórolni.

Maga a szoftver egy éles testen is részt vett. Ezt úgy értem, hogy megkértem egy ismerősömet tesztelje le a működést, mindezt úgy hogy szinte semmi lényeges információt nem adtam át neki. Ezt azért találtam szükségesnek, hogy egy kívülálló is próbálja le, egy olyan ember aki pártatlan.

Mindezen tesztek eredményével elégedet vagyok hisz sikeresen zárultak és úgy érzem a célját elérte.

## Befejezés:

A záró dolgozatomat sikeresen elkészítettem, egy olyan szoftvert amely más külső hardver és egy régóta meglévő, de kevésbé használt szabvány felhasználva szeretném ezt a technológiát népszerűsíteni, mert jól működő és kényelmes felhasználást biztosít. Amit a fejlesztésben így utólag belátva másképp csinálnék az a tesztek írása. Mivel úgy gondoltam ez nem egy túl bonyolult és több helyen összefüggő szoftver így a tesztek írását a végére hagyhatom nem pedig a különböző egységek után megírva őket. Néhol bizony könnyebb helyzetben lettem volna ha rögtön tudtam volna az előző részt is tesztelni.

Végezetül szerintem sikerült egy olyan Ügyfélnyilvántartó rendszert készíteni, ami alkalmas nagy partnertörzsszel rendelkező kis és középvállalkozások ügyfeleinek nyilvántartására. Későbbiek folyamán lehet a szoftvert bővíteni, vagy akár igények szerint új modulokat hozzá vagy rá építeni.

## Képek, illusztrációk:

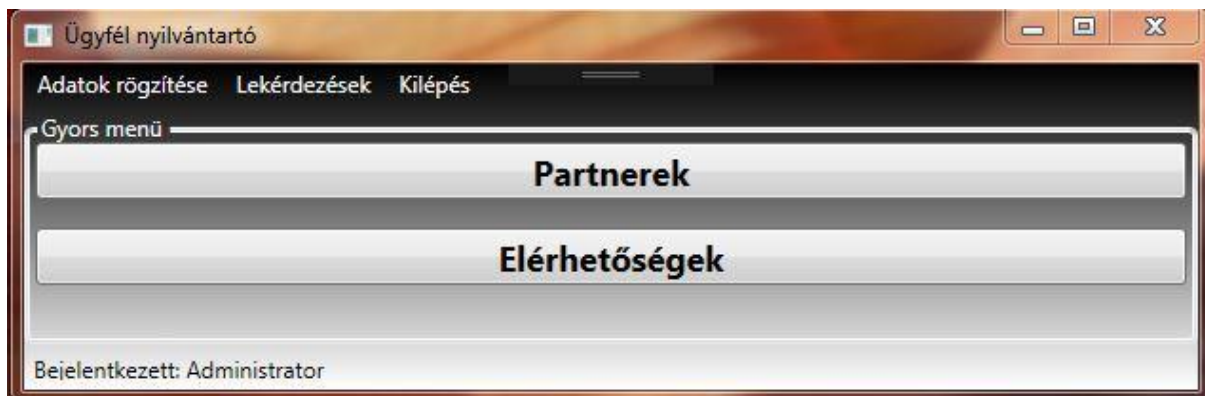
### Bejelentkezési felület.

*Ezzel találkozik először a felhasználó, ha kötelező a bejelentkezés.*



### A program fő, elsődleges ablaka.

*Ez az ablak hívódik meg program induláskor.*





## Listák megjelenítése.

*Ez az ablak többször van felhasználva a program során.*

The image shows two side-by-side application windows. The left window is titled 'Partnerek' and displays a list of companies with their county and city. The right window is titled 'Elérhetőségek' and displays a list of individuals with their phone and email addresses. Both windows have a search bar and buttons for adding and deleting records.

Név	megye	város
Alma kft	Békés megye	Bcsaba
BlaBlaCom Kft.	Pest megye	Debrecen
Havas Imre ev.	Pest megye	Kerepes
SDC Bt	Heves megye	Hatvan
Teszt Kft	Egyéb	Budapest

Név	tel.	email
Alap Béla	063011555888	alap.bela@examle.com
Alsó Andrea	063-840-28777	a.andi@example.com
Bab Béla	067-194-91111	bb@example.com
Bab Endre	0678 849 51 60	bab.endre@gmail.com
Gábor Bácsi	068011111999	info@gabibacsi.hu
Gárdonyi Géza	06701231212	geza@gardonyi.info
Kerepesi Sándor	067 147 91180	kerepesi.s@example.com
Sás Pali		

## Kapcsolatok űrlapja.

*Itt lehet felvinni, szerkeszteni a személyek adatait.*

The image shows a form window titled 'Kapcsolattartó űrlap'. It contains fields for Name, Phone number, Email address, and Status. There are buttons for 'Névjegy mentése' and 'Mentés'.

Név: Alap Béla

Telefonszám: 063011555888

Email cím: alap.bela@examle.com

Sátusz: Tulajdonos

Névjegy mentése      Mentés

## Partnerek űrlapja.

*Itt lehet a partnerek adatait megadni és a kapcsolatokat mellérendelni.*

**Partner űrlap**

Név:  Megye:

Partner típus:  Irsz.:  Város:

Tulajdonos neve:  Cím:

Név: Gábor Bácsi Tel: 068011111999 Email: info@gabibacsi.hu

+ - Nyomtatási kép Mentés

## Partner űrlap nyomtatási képe.

**Nyomtatási kép**

1 of 1 100%

**Partner űrlap** 2017. október 7

Név: Havas Imre ev.  
Város: Kerepes  
Cím: Vas krt. 75  
Tulajdonos neve: Sás Pali

**Kapcsolatok**

Név	Telefonszám	Email
Teszt Géza	06309999988	teszt.geza@example.com
Sás Pali	708889011	pali@sas.hu
Gábor Bácsi	068011111999	info@gabibacsi.hu
Bab Béla	067-194-91111	bb@example.com

Ügyfényilvántartó rendszer