

1. BEVEZETÉS

A munkahelyemen szükségessé vált, hogy lecseréljük korábban még DOS-os operációsrendszerre fejlesztett raktárprogramunkat. Manapság használt operációs rendszereken nehézkes és elég körülményes futtatni egy ilyen alapokra épített programot, ha ki is szeretnénk használni a mai kornak megfelelő új operációsrendszerekben rejlő lehetőségeket. Mivel nap mint nap én kerülök szembe eme rendszer problémájával, így tudom mi az, amire még szükségünk lehet az új raktárprogramban. Ezért is döntöttem úgy, hogy egyúttal diplomamunkámként is ezt a programot mutatom be.

1.1 Luinox Kft. bemutatása

A cég 1997-ben alakult és rozsdamentes élelmiszeripari szerelvények kereskedelmével foglalkozik. Elég nagy a termékskála még így is, hogy elég speciális anyagokról van szó. Nyitáskor még kézzel írott számlákat állítottak ki a kolléganők, ami elég sok problémához vezetett. Egy elírt számlánál új számlát kellett írni, így sok rontott számla is keletkezett. A terméktételeket soronként kellett leírni, a darabszámot összeszorozni az árral, áfát kiszámolni, a raktárosnak kartont vezetni és sorolhatnám még a sok nehézséget. Számlázó programra volt szükségünk. A megvásárolt számítógépre egy Windows 95 operációsrendszer került és egy raktárprogram, amely Pascal-ban lett fejlesztve. Mai napig ez a program teljesíti feladatát hibátlanul, de az idő eljárt felette. Cégünk növekszik, bővül és ezért több funkcióra és információra van szükségünk mind a beszerzésről, mind a kereskedésről is. Új operációsrendszerek jelentek meg új lehetőségekkel, új formátumokkal. Ezeken a rendszereken a fejlesztői környezetek is korszerűbbek, gyorsabbak, átláthatóbbak, könnyebben kezelhetőek.

2. FEJLESZTŐI KÖRNYEZET BEMUTATÁSA

Egyetemi tanulmányi éveim során ismerkedtem meg a Java programozási nyelvvel, ami a szívemhez nőtt. Ezért is került a választásom az új raktárprogramunk fejlesztésének alapjává. Fejlesztői környezetnek a NetBeans IDE 6.0 programot találtam ideálisnak, könnyen megtanulhatónak. A program grafikus fejlesztőfelületet kínál a különböző alkalmazások, Appletek vagy akár JavaBeanek elkészítéséhez, amelynek segítségével könnyebben, gyorsabban tudjuk fejleszteni saját programjainkat. Szükségem volt még egy adat-báziskezelő rendszerre. Választásom a PostgreSQL 8.2-es verziójára esett, népszerűsége és ingyenessége miatt is. A program fejlesztése során szükségessé vált még egy fejlesztői eszköz integrálása, amely a riportok és nyomtatványok megjelenítéséért felelős. Ez számomra egy teljesen ismeretlen eszköz (JasperSoft által fejlesztett iReport-nb-3.1.0) volt. Programozásának megtanulása és kiismerése, újabb idővel tolt ki a program fejlesztési idejét.

Megjegyzés: diplomamunkámban a fejezetek végén a kapcsos zárójelek közötti számok az irodalomjegyzék pontjaira való hivatkozást mutatják.

2.1 Java nyelvről röviden

2.1.1 Java története

A Sun Microsystems 1990 decemberében szórakoztató elektronikai készülékek számára fejlesztett beágyazott operációs rendszer szoftvereket (Green project). A fejlesztőcsapat tagjai voltak James Gosling (Java atyja), Patrick Naughton és Mike Sheridan.

1991-ben a C++ fordító kiterjesztésével foglalkoztak, céljuk egy olyan platformfüggetlen, objektum-orientált nyelv kidolgozása volt, amely lehetővé teszi a különféle eszközök (CD lejátszó, műholdvevő, stb.) közötti kommunikációt. Gosling felismerte a C++ alkalmatlanságát erre a feladatra, és a csapat megtervezte saját nyelvét az Oak-ot (James Gosling, a nyelv atyja nevezte így el az irodája előtt növő

tölgyfáról). Később kiderült, hogy ilyen elnevezésű nyelv már létezik, ezért végül *Java* néven vált ismertté.

Az Internet és a World Wide Web akkori rohamos terjedése miatt a Java hamarosan népszerű lett, mint biztonságos nyelv. A Microsoftot annyira aggasztotta a Java sikere, hogy elkezdte fejleszteni saját Java verzióját, amely végül a Visual J++ fejlesztői környezetben öltött testet. A Microsoft igyekezett a Java-t Windows-hoz kötni, emiatt a nyelv platformfüggetlenségét aláásó, Windows-specifikus megoldásokat valósított meg benne. Ezt a Java-t a szakmai tolvajnyelv „polluted Java”-nak, szennyezett Java-nak nevezte el. Mivel a Microsoft ezt a nyelvet is Java-nak nevezte el a SUN pert indított ellene. Az évekig tartó eljárás vége az lett, hogy a Microsoft-nak fel kellett hagynia a „polluted Java” fejlesztésével, és a Windows XP-be már a Microsoft Java VM-et sem tehette bele. A *Java* szó a Sun Microsystems védjegye. Ennélfogva engedélye nélkül más nem használhatja más által kifejlesztett termékek megjelölésére, még pl. Java-szerű stb. összetételekben sem, mert ez a védjegyjogosult jogaiba ütközik. [1]

2.1.2 Objektum orientáltság

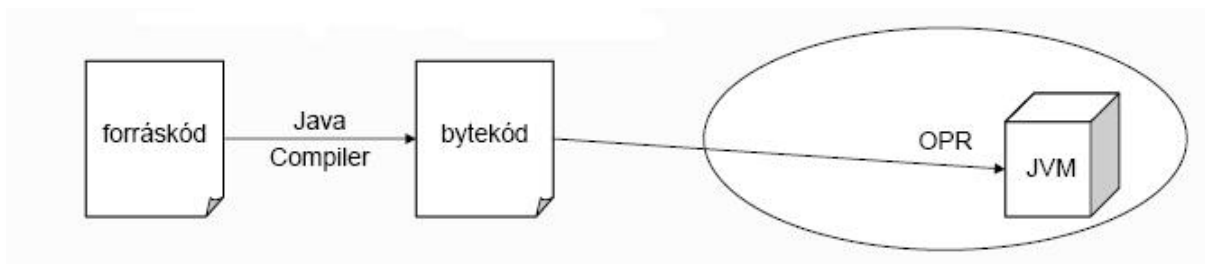
A nyelv első tulajdonsága, az objektum-orientáltság ("OO"), a programozási stílusra és a nyelv struktúrájára utal. Az OO fontos szempontja, hogy a szoftvert "dolgozók" (objektumok) alapján csoportosítja, nem az elvégzett feladatok a fő szempont. Ennek alapja, hogy az előbbi sokkal kevesebbet változik, mint az utóbbi, így az objektumok (az adatokat tartalmazó entitások) jobb alapot biztosítanak egy szoftverrendszer megtervezéséhez. A cél az volt, hogy nagy fejlesztési projekteket könnyebben lehessen kezelni, így csökken az elhibázott projektek száma.

2.1.3 Platformfüggetlenség (hordozhatóság)

Platformfüggetlenség azt jelenti, hogy a Java-ban íródott programok hasonlóan fognak futni különböző hardvereken. Ezt úgy lehet megvalósítani, hogy a Java fordítóprogram csak egy úgynevezett *Java gépi kódra* fordítja le a forráskódot, ami aztán futtatva lesz a virtuális gépben, amely lefordítja az illető hardver gépi kódjára.

Továbbá, szabványos könyvtárcsomagok léteznek, amelyek elérhetővé teszik az illető hardver sajátosságosságait (grafika, szálak és hálózat) egységes módon.

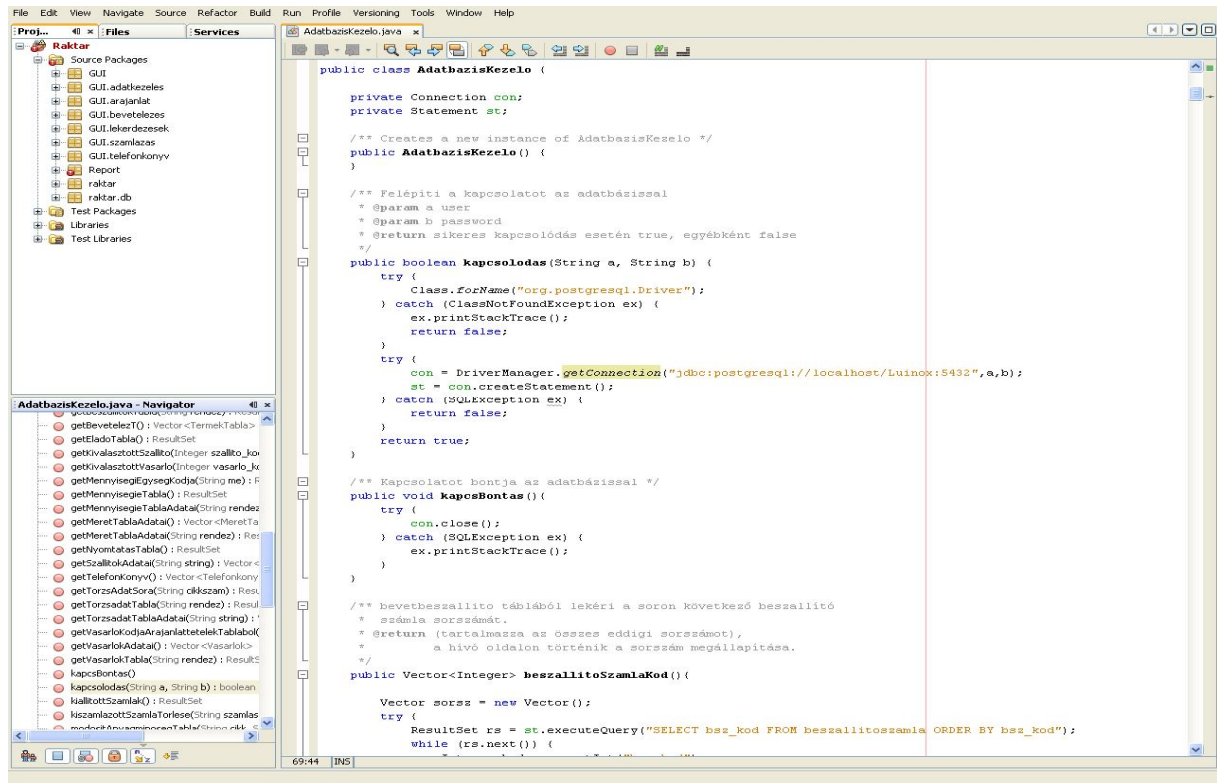
1. A programozó elkészíti a forráskódokat
 - A forrásfájlok .java kiterjesztésű fájlok
2. A forráskódot bytekóddá fordítja (javac compiler)
 - A bytekód .class kiterjesztésű bináris állomány
 - A bytekód hordozható, platformfüggetlen, kompakt
3. A bytekódot a felhasználó a Virtuális Gépen lefuttatja
 - Azaz a bytekód a JVM gépi kódja



[1]

2.1.4 NetBeans IDE fejlesztői környezet rövid bemutatása

A Sun Microsystems 2000 júniusában hozta létre a NetBeans nyílt forráskódú projectet. Olyan fejlesztői környezet, amely lehetővé teszi a számomra (programozó számára), hogy programokat írjak, fordítsak, teszteljek, hibakeresést végezzek az alkalmazásokban. Java nyelven íródott, de bármilyen más programozási nyelvet is támogat (C/C++, PHP, Ajax). A NetBeans IDE (Integrated Development Environment) szoftvert számos modullal bővíthetem (mint például a későbbiekben is bemutatott iReport-nb-3.1.0) és futtatható Linux, Windows, Mac OS X és Solaris rendszereken is. A NetBeans IDE ingyenes termék és nincsenek érvényben korlátozások a használatára vonatkozóan.



Saját tapasztalatok: egyszerűen kezelhető, szinte a programozó kezébe adja a különböző programozási eszközöket. Gyorsan tudok létrehozni vele felületeket, keretet (Frame), paneleket (Panel), amelyek az egyes újonnan megnyílt ablakokat valósítják meg. Ezekre gyors kattintgatásokkal tudok új objektumokat elhelyezni, mint például gombokat (Button), szöveges mezőket (Label), táblákat, görgető sávokat (Scroll Bar), stb. A csomagok, fájlok, módszerek, eljárások a bal oldali ablakokban teljes mértékben átláthatóak és hozzáférhetőek. Adatbázis kapcsolatot is fel tud építeni különböző adatbázisokkal. Véleményem szerint napjaink egyik legjobb programfejlesztő eszköze.

2.2 PostgreSQL adatbázis

A PostgreSQL olyan adatbázis-kezelő rendszer, amely SQL (Structured Query Language - rendszerezett lekérdezőnyelv) szabványú felületen elérhető adatbázis-kezelést tesz lehetővé. A PostgreSQL BSD licensszel rendelkező, objektum orientált adatbázis-kezelő rendszer, így a MySQL-lel ellentétben teljesen szabadon

felhasználható. A BSD licenz lényege, hogy a program szabadon felhasználható és terjeszthető. A licenz által nyújtott szabadság miatt a különféle BSD kódok nagy számban kerültek beépítésre kereskedelmi termékekbe. A PostgreSQL bármilyen felhasználásra, akár kereskedelmire is.

Egyre növekvő népszerűsége ingyenes voltán kívül annak is köszönhető, hogy minden modern Unix jellegű operációs rendszeren futtatható - ilyen, pl. az AIX, a Digital Unix, a FreeBSD, a HP-UX, és természetesen a Linux -, de van Mac OS, Windows 2000 és Windows NT alatt működő verziója is. Talán meglepő, de a PostgreSQL minőségét és képességeit illetően felveszi a versenyt a vezető piaci adatbázis-kezelő rendszerekkel is, sőt nem egy területen felül is múlja azokat.

2.2.1 Mit tud a PostgreSQL?

Minden fontosat, amit egy modern RDBMS-nek (Relation Database Management System) tudnia kell. Számos nézőpontból lehet vizsgálni a szoftvert: képességek, teljesítmény, megbízhatóság, támogatottság.

Képességek: a PostgreSQL rendelkezik a nagy, kereskedelmi DBMS-ek képességeivel, ilyenek a tranzakciók, al-lekérdezések, triggerek, nézetek, külső kulcsok, integritás és kifinomult zármechanizmusok. Van néhány képessége, ami a kereskedelmi adatbázisokból hiányzik, mint például a felhasználó által definiált típusok, öröklődés.

Teljesítmény: a PostgreSQL teljesítménye hasonlít a kereskedelmi és más nyílt adatbázis szerverekéhez. Lehet bizonyos esetekben lassabb, másokban gyorsabb.

Megbízhatóság: a PostgreSQL megbízható és stabil, akár évekig is működik leállás nélkül. Nagyméretű rendszer részeként való működésnek is megfelel. A PostgreSQL-t egy stabil fejlesztőcsoport, a Global Development Group tartja karban és egy népes közreműködői tábor nyújt segítséget a hibák felderítésében és kijavításában. Megbízhatóságát ennek is köszönheti, a legfőbb ok azonban az, hogy a PostgreSQL már viszonylag hosszú, több mint egy évtizedes múltra tekinthet vissza.

Támogatás: a PostgreSQL is rendelkezik mindenféle meghajtóval (odbc, jdbc, dbi, .net), illetve nagyon sok programozási nyelvre elérhető hozzá csatlakozási felület (C, C++, Java, PHP, Python, stb). [2]

2.2.2 PostgreSQL telepítése

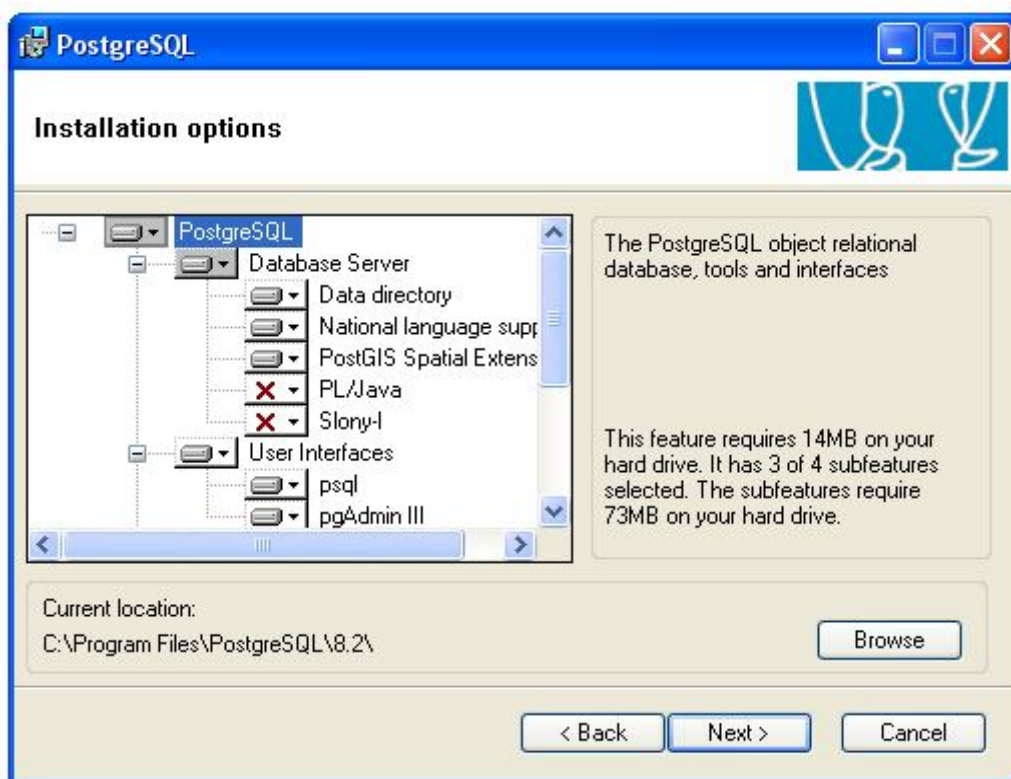
Windows-os fejlesztők közül kevesen ismerik a PostgreSQL-t. Ennek első és legfontosabb oka, hogy mostohán kezelte ezt a platformot. Csak körülményesen, külön futtatókörnyezet (Cygwin) telepítésével volt futtatható. Ez szerencsére megváltozott a 8.0 változat első bétáinak megjelenésével. Elkészültek az első natív, valóban windows-os környezetre készült telepítőcsomagok is. A PostgreSQL 8 NT alapú verziókat támogatja, ezek a Windows 2000, az XP és a Windows 2003.

A cd-n található két (postgresql-8.2.msi és postgresql-8.2-int.msi) fájl bármelyikét elindíthatjuk a telepítéshez, amely magyar nyelven még nem kommunikál.

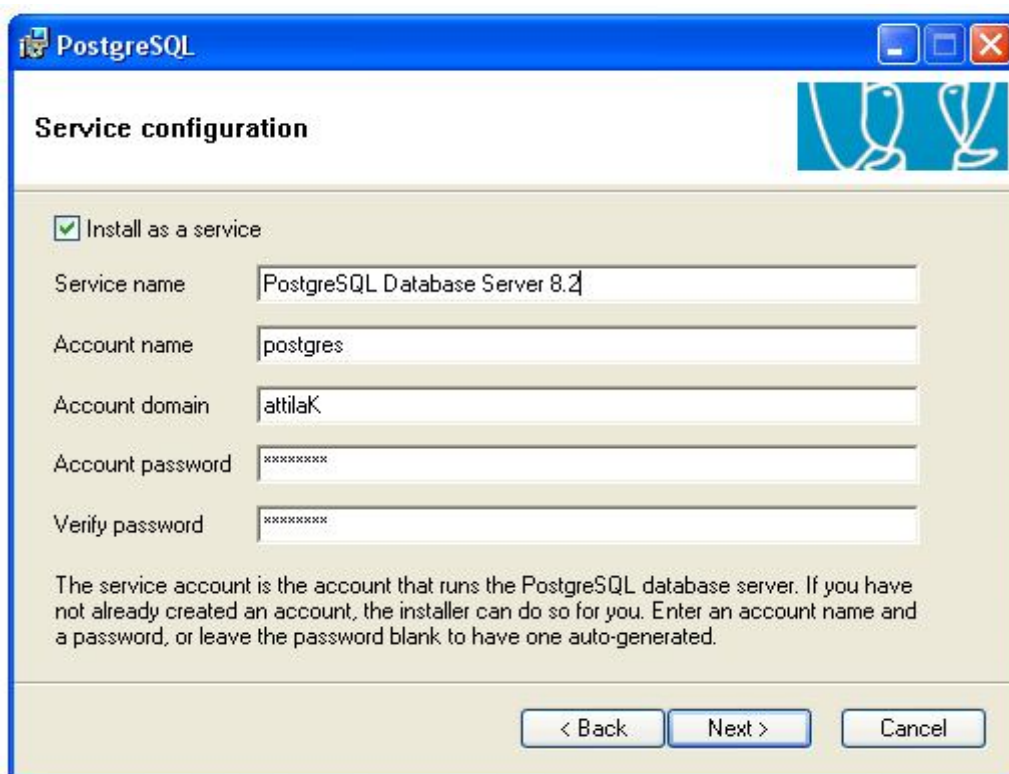


Maga a PostgreSQL 8 telepíthető mind FAT, mind NTFS meghajtókra, addig az adatbázisok csak NTFS fájlrendszeren tárolhatók, mivel lehetőséget nyújt baleset (áramkimaradás, lemezhiba, stb.) esetén az adatok valamilyen szintű helyreállítására, valamint hogy szintén biztonsági okokból a kiszolgáló nem futtatható adminisztratív jogokkal rendelkező felhasználó programjaként.

A telepítő egyebek mellett a (külön letölthető) pgAdminIII programot és a különböző meghajtó-programokat (mint ODBC, JDBC) is tartalmazza.



Kiválasztható, hogy szolgáltatásként fusson-e az adatbázis-kiszolgáló (ez ajánlott opció, valamint egyike a nyolcas verzió újdonságainak), valamint a kiszolgáló futtatására kijelölt felhasználó adatai, amelyet a telepítő létre is hozhat a rendszerben:



The image shows the 'Service configuration' window of the PostgreSQL installer. It has a blue title bar with the PostgreSQL logo and standard window controls. The main area is light beige. At the top, the title 'Service configuration' is in bold. Below it, there's a checkbox 'Install as a service' which is checked. Then, there are five input fields: 'Service name' (containing 'PostgreSQL Database Server 8.2'), 'Account name' (containing 'postgres'), 'Account domain' (containing 'attilaK'), 'Account password' (masked with 'xxxxxxx'), and 'Verify password' (masked with 'xxxxxxx'). Below these fields is a paragraph of text explaining the service account. At the bottom right, there are three buttons: '< Back', 'Next >', and 'Cancel'.

Service configuration

☒ Install as a service

Service name: PostgreSQL Database Server 8.2

Account name: postgres

Account domain: attilaK

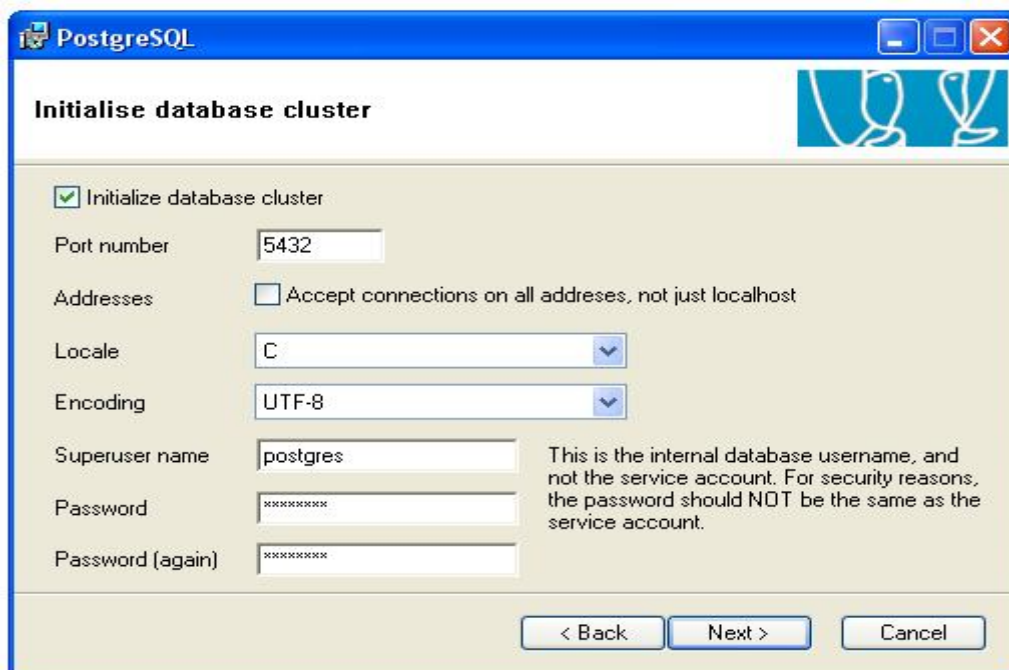
Account password: xxxxxxx

Verify password: xxxxxxx

The service account is the account that runs the PostgreSQL database server. If you have not already created an account, the installer can do so for you. Enter an account name and a password, or leave the password blank to have one auto-generated.

< Back Next > Cancel

A következő képernyőn a „szokásos” adatbázis-jellemzők állíthatók be. A kiszolgáló portja, a nyelvi beállítások, valamint az alapértelmezett felhasználó, a „superuser” neve és jelszava. A karakterkódolást át kell állítani UTF-8-ra.



The image shows the 'Initialise database cluster' window of the PostgreSQL installer. It has a blue title bar with the PostgreSQL logo and standard window controls. The main area is light beige. At the top, the title 'Initialise database cluster' is in bold. Below it, there's a checkbox 'Initialize database cluster' which is checked. Then, there are several settings: 'Port number' (5432), 'Addresses' (a checkbox 'Accept connections on all addresses, not just localhost' which is unchecked), 'Locale' (a dropdown menu showing 'C'), 'Encoding' (a dropdown menu showing 'UTF-8'), 'Superuser name' (postgres), 'Password' (masked with 'xxxxxxx'), and 'Password (again)' (masked with 'xxxxxxx'). To the right of the password fields is a paragraph of text explaining the superuser account. At the bottom right, there are three buttons: '< Back', 'Next >', and 'Cancel'.

Initialise database cluster

☒ Initialize database cluster

Port number: 5432

Addresses: ☐ Accept connections on all addresses, not just localhost

Locale: C

Encoding: UTF-8

Superuser name: postgres

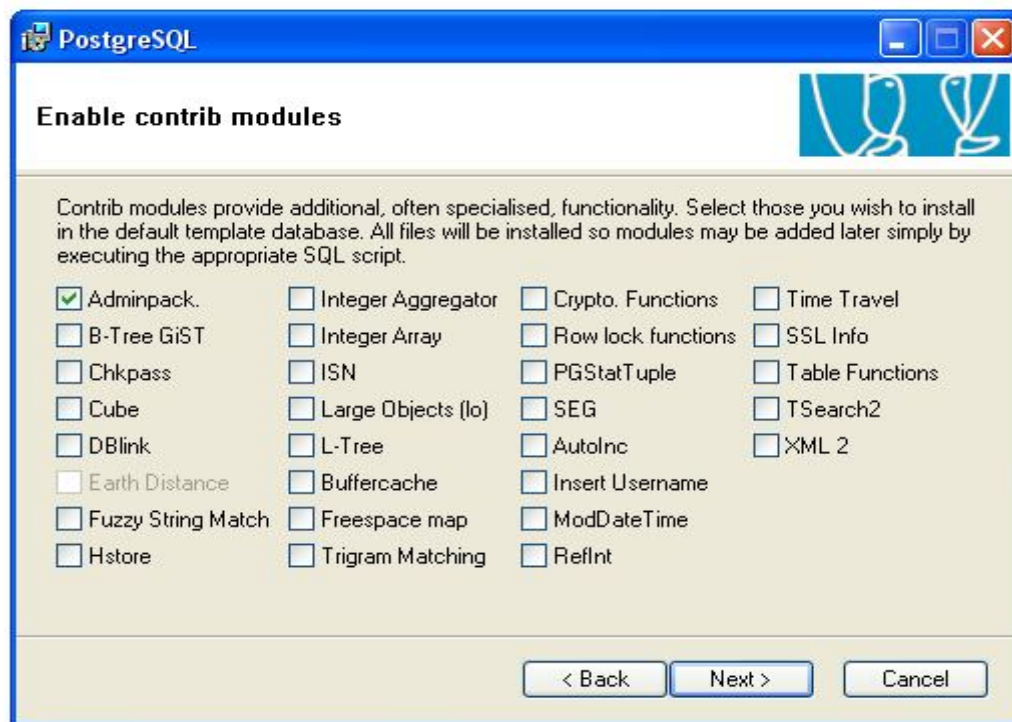
Password: xxxxxxx

Password (again): xxxxxxx

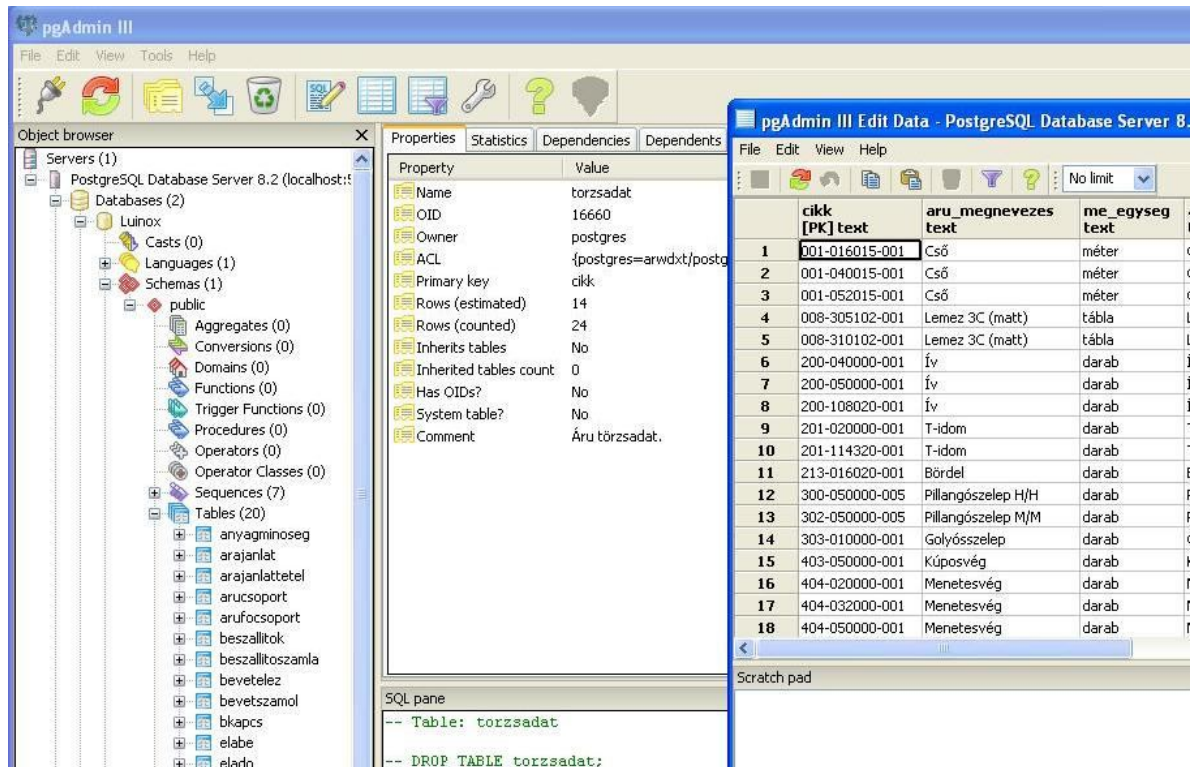
This is the internal database username, and not the service account. For security reasons, the password should NOT be the same as the service account.

< Back Next > Cancel

A telepítő által is tartalmazott PL/pgsql procedurális nyelv támogatása alapértelmezetten elérhető, de emellett számos más is bekapcsolható külön.



Telepítés végeztével még feliratkozhatunk a pgsql-announce levelezési listára, amelyen a PostgreSQL frissítésekről kapunk információkat. A Start Menü bejegyzései között indítjuk a pgAdminIII programot is, amely az egyik legátfogóbb és legkényelmesebb eszköz a PostgreSQL kiszolgáló kezeléséhez.

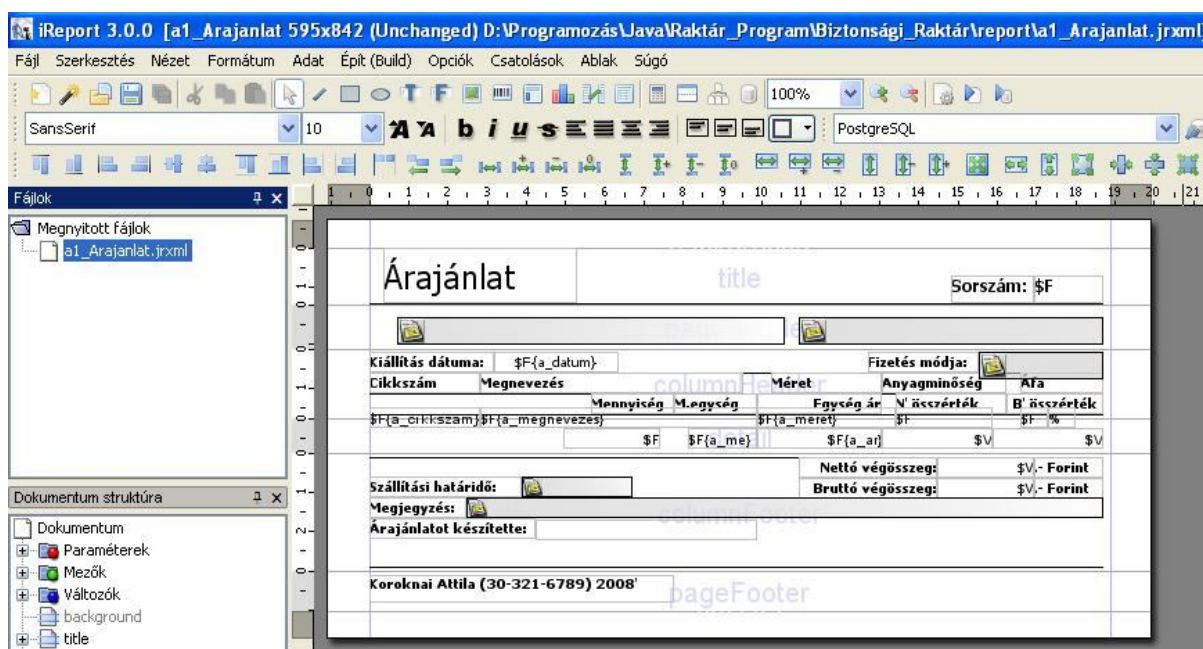


A program indítását követően jobb egérgombbal a Databases-re kattintva létre kell hozni egy új adatbázist. Névnek meg kell adni, hogy „Luinox”, tulajdonosnak pedig „postgres”. A karakterkódolást UTF8-ra kell állítani, majd a megjegyzés rovatba írható bármilyen további információ. A létrehozott új adatbázisba szintén jobb egérgombra kattintva ki kell választani a Restore Database menüpontot. A CD-ről ki kell választani a „Luinox.backup” fájlt majd az „OK” gombra kattintással az adatbázisban létrejönnek a program által használt üres táblák.

2.3 iReport (Jasper Report)

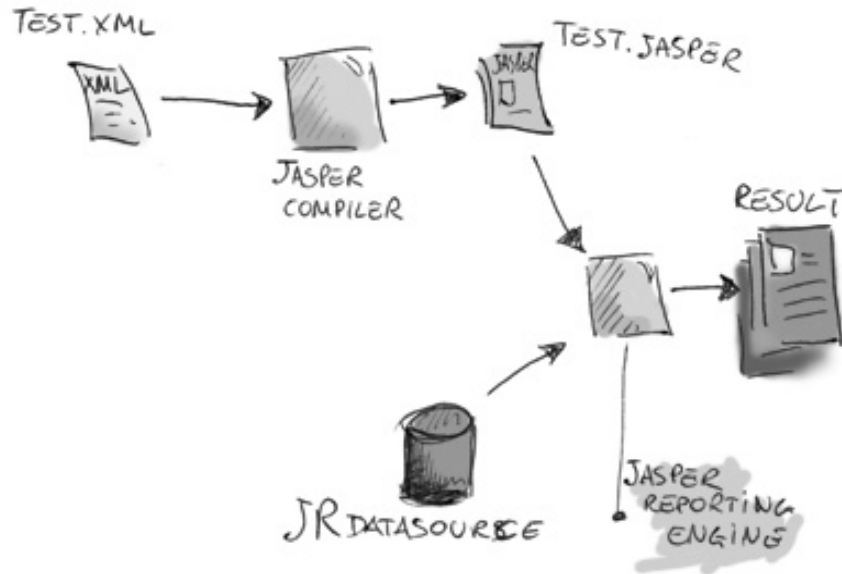
A lekérdezésekhez, számlázásokhoz szükség volt egy nyomtatványkészítő eszközre. Az iReport a JasperReport nyílt forrású grafikus riport készítő eszköze. Segítségével gyorsan készíthetünk szép riportokat. A riportokba építhető diagramok, képek, szövegdobozok, rajzok. Eddig egyedülálló alkalmazásként volt használható, de a legújabb formájában beintegrálhatóvá vált a NetBeans IDE felületre is, így a NetBeans felhasználók még egyszerűbben készíthetnek mutatós riportokat.

Használatáról röviden:



Az „Adat/Riport lekérdezés” menüpont alatt adható meg a lekérdezés (sql utasítás). Az eredményül kapott tábla oszlopai megjelennek a bal oldali „Dokumentum struktúra” ablakban a mezők között. Ahhoz, hogy az eredményül kapott tábla minden sorát megjelenítse, az oszlop vagy oszlopok nevét be kell húzni a riport detail szekciójába. Egy riport egy sql lekérdezést tud végrehajtani és eredményül kapott tábla adatait megjeleníteni. Van lehetőség alriportokat készíteni és azt beszúrni egy másikba. Így lehet több lekérdezéssel még több információt megjeleníteni. A riport készítése közben egy xml forrás generálódik. Ha nincs hiba a forrás fájlban, akkor a fordító előállít egy jasper kiterjesztésű fájlt. Ezek után történik meg az adatok beolvasása a riportba és generálódik a jelentés. A jelentés több formátumban előállítható: PDF, HTML, XML, XLS, CVS, stb. [3]





Forrástól a riportig [4]

3. FEJLESZTŐI DOKUMENTÁCIÓ

A cégnek három alkalmazottja van (tulajdonos, raktáros, adminisztrátor), ezért is nem volt szükséges a hálózat kiépítése és a program egyszerre több gépen való párhuzamos futtatása. Az adminisztrátor feladata az áruk bevételezése, számlázása, a tulajdonosnak különböző riportok készítése, vagyis Ő a program használója. Jelenleg még nincs arra igény, hogy több felhasználós rendszer legyen kiépítve, de az alapok már fel vannak építve. Mivel egy programot soha nem tekinthetünk késznek, ezért ezt a programot is lehet továbbfejleszteni és már vannak új ötletek is, melyeket szeretnék beépíteni a későbbiekben. Az alábbiakban a jelenlegi verzió felépítéséről, adatbáziskapcsolatáról, a használt táblákról fogok egy kis leírást közölni néhány java forráskóddal kiegészítve.

3.1 Adatbázis kapcsolat és a táblák felépítése

3.1.1 Kapcsolat felépítése (JDBC)

A Java programok kommunikálni tudnak a legkülönbözőbb adatbázisokkal is. A piacon rengeteg adatbázis szerver érhető el, de a Java nem tartalmazza mindegyik elérhetőségét. A Java csak egy interfész halmazt biztosít. Az adatbázisgyártónak kell megírni a tényleges kezelő osztályokat, melyek révén a párbeszéd az adatbázissal létrejön. Ezeket a megvalósításokat nevezik JDBC - Java DataBase Connectivity drivereknek. A JDBC nem más, mint egy programozói interfész (API), SQL – Structured Query Language – utasítások végrehajtására. A JDBC használatával a Java adatbázis-kezelő programjaink nemcsak platform függetlenekké, hanem még adatbázis-kezelőktől függetlenekké is válnak. [5]

Egy JDBC meghajtó-program valósítja meg a JDBC hívásokat egy adott adatbázistípushoz a DRIVER interfészt implementálva. A raktárprogramom esetében a PostgreSQL driver-ére van szükség - "org.postgresql.Driver".

Az adatbázissal való kommunikálás lépései: a DriverManager-től egy adatbázis-specifikus driver-t kérünk. A specifikus driver létrehozza a kapcsolatot az adatbázissal és egy Connection típusú objektumot ad vissza. Részlet az AdatbazisKezelo.java osztályból, hogyan is kapcsolódik az adatbázishoz. Paraméterül kapott a és b string típus a jogos felhasználó neve és kódja. A sikeres kapcsolódás esetén true értékkel tér vissza a hívóhoz.

```
public boolean kapcsolodas(String a, String b) {  
    try {  
        Class.forName("org.postgresql.Driver");  
    } catch (ClassNotFoundException ex) {  
        ex.printStackTrace();  
        return false;  
    }  
    try {  
        con = DriverManager.getConnection("jdbc:postgresql://localhost/Luinox:5432",a,b);  
        st = con.createStatement();  
    } catch (SQLException ex) {  
        return false;  
    }  
    return true;  
}
```

A Connection típusú objektum segítségével létrehozunk egy Statement objektumot, amely egy SQL kérést is tartalmaz az adatbázis felé. A Connection tulajdonképpen egy interfész, amely biztosítja a kérések eljutását az adatbázishoz és a válaszok megérkezését az adatbázistól. Az interfész metódusokat szolgáltat ahhoz, hogy információkat kapjunk az adatbázisról, bezárja az adatbázissal való kapcsolatot és lehallgatja az adatbázistól visszaérkező üzeneteket.

A Statement objektum az eredményeket egy ResultSet objektumban adja vissza, azaz lekérdezett információk egy része táblázatszerű, ezeket az SQL lekérdező utasítások eredményeit is reprezentáló eredménytábla a ResultSet osztály egy példánya fogja tartalmazni. Az alábbi példa az anyagminőség tábla minden sorával és oszlopával (a paraméterül kapott oszlopnév szerint rendezve) tér vissza a hívóhoz.

```
public ResultSet getAnyagminosegekTabla(String rendez){  
  
    ResultSet a=null;  
    try{  
        a = st.executeQuery("SELECT * FROM anyagminoseg ORDER BY "+rendez);  
    }catch (SQLException ex) {  
        Logger.getLogger(AdatbazisKezelo.class.getName()).log(Level.SEVERE, null, ex);  
    }  
    return a;  
}
```

A kapcsolat lezárása a close metódus használatával történik. A close metódus felszabadítja az adatbázis-kapcsolat által lefoglalt JDBC erőforrásokat. Ez a metódus a kapcsolatobjektum megsemmisítésekor és bizonyos fatális hibák fellépésekor automatikusan is meghívódik.

```
public void kapcsBontas(){  
    try {  
        con.close();  
    } catch (SQLException ex) {  
        ex.printStackTrace();  
    }  
}
```

3.1.2 Táblák struktúrája

A program jelenlegi verziója tizenkilenc táblával épít fel kapcsolatot. Csak a bevételezés és számlázás során hat-hat táblába kerülnek be vagy módosulnak az adatok.

- Adatkezelés során használt táblák

Új árutörzs felvétele két táblát kezel, az egyik a „törzsadat” a másik a „raktár”. A törzsadat táblába, a megfelelő mezőkbe beíródnak az új áru adatai, továbbá a raktár táblában is létrejön egy új sor, ahol az áru cikkszáma, mennyisége és annak értéke kerül tárolásra. Új törzsadat létrehozásánál a raktár táblában az új termék mennyisége és értéke automatikusan nulla lesz. Ez a tábla fogja nyilvántartani, hogy ebből a bizonyos termékből milyen mennyiség van raktáron és ennek mennyi az értéke. A raktár tábla adatai a beszerzés és értékesítés folyamán fognak változni. Ezekben a táblákban az elsődleges kulcsok a termék cikkszámai. A cikkszámok nem ismétlődhetnek, hiszen ezek fognak egy sort (terméket) azonosítani. A táblák struktúrája a következő:

Törzsadat tábla: primary_key(cikk)

Mezőnév	Típusa	Jelentése
cikk	text	Termék cikkszáma
aru_megnevezes	text	Termék neve
me_egyseg	text	Mennyiségi egysége
arucsoport	text	Melyik árucsoportoz tartozik
afa	smallint	Áfa mértéke
min_keszlet	smallint	Minimum készlet
atlag_bear	real	Átlag beszerzési ára
kisker_ear	real	Kiskereskedelmi eladási ára
torzsvasarloi_ear	real	Törzsvásárlói eladási ára
anyagminoseg	text	Anyagminősége
arufcsoport	smallint	Melyik áruőcsoportoz tartozik
aru_meret	text	Áru mérete
vtsz_szj	text	Vámtarifaszáma / SZJ száma

Raktar tábla: primary_key(r_cikk)

Mezőnév	Típusa	Jelentése
r_cikk	text	Termék cikkszáma
r_db	real	Raktári mennyiség
r_ertek	real	Nettó összértéke

További adatkezelésnél használt táblák: a vásárlók és beszállítók adatait tároló táblák, a törzsadathoz tartozó mérettábla, anyagminőség tábla, mennyiségegység tábla, árucsoport és árufőcsoport táblák. A legtöbb esetben a tábla elsődleges kulcsa serial típusú. Ha egy ilyen típus van beállítva, akkor egy új sor létrehozásánál automatikusan sorszámozza az adatbázis az adott sort. Nem kell megadni az sql utasításban ennek a mezőnek az értékét. A táblák struktúrája a következő:

Vasarlok tábla: primary_key(v_kod)

Mezőnév	Típusa	Jelentése
v_kod	serial	Vásárló kódja
v_nev	text	Vásárló neve
v_irsz	text	Irányítószáma
v_cim	text	Címe
v_telefon	text	Telefonszáma
v_fax	text	Fax száma
v_adoszam	text	Adószáma
v_vertek	real	Vásárolt érték
v_varos	text	Város
v_email	text	E-mail címe
v_kedvezmeny	smallint	Kedvezménye
v_euadoszam	text	EUs adószáma

Beszallitok tábla: primary_key(b_sorsz)

Mezőnév	Típusa	Jelentése
b_sorsz	serial	Kódja
b_nev	text	Szállító neve
b_irsz	text	Irányítószáma
b_varos	text	Város
b_cim	text	Címe
b_telefon	text	Telefonszáma
b_fax	text	Fax száma
b_email	text	E-mail címe
b_vertek	integer	Vásárolt érték

Arucsoport tábla: primary_key(aru_cs)

Mezőnév	Típusa	Jelentése
aru_cs	char. varying(30)	Csoport neve
aru_cikk	char. varying(3)	Cikkszám
aru_fcs	smallint	Főcsop. kódja

Arufocsoport tábla: primary_key(afcs_kod)

Mezőnév	Típusa	Jelentése
afcs_kod	serial	Főcs.kódja
afcs_neve	char. varying(15)	Főcs.neve

Merettábla tábla: primary_key(m_cikk)

Mezőnév	Típusa	Jelentése
m_cikk	char.varying(6)	Cikkszám
m_meret	char.varying(14)	Méret

Anyagminoseg tábla: primary_key(anyag_cikk)

Mezőnév	Típusa	Jelentése
anyag_cikk	char.varying(3)	Cikkszám
anyag_neve	char.varying(6)	Anyagminőség

Mennyisegiegység tábla: primary_key(me_kod)

Mezőnév	Típusa	Jelentése
me_kod	serial	Kódja
me_me	character	Megnevezés

Telefonkonyv tábla: primary_key(t_nev)

Mezőnév	Típusa	Jelentése
t_nev	text	Név
t_szam	text	Telefonszám

Az alábbi java kód szemlélteti, hogy az adatbáziskezelő osztály hogyan szűrja be a „vasarlo” táblába az új vásárlókat. Ez az osztály valósítja meg a vásárlót, amely többek között tartalmazza a nevét, címét, adószámát, telefonszámát, stb. Egy string típusú változóba létrehozza az sql utasítást, melynek paramétereit a „vasarlo”

objektumból olvassa ki. A sikeres új sor beszúrása esetén true értékkel tér vissza a hívó oldalra, ellenkező esetben false.

```
public boolean beszurUjVasarloTablaba(raktar.Vasarlok vasarlo){
    try{
        String sql = new String("INSERT INTO vasarlok(v_nev,v_irsz,v_cim," +
            "v_telefon,v_fax,v_adosszam,v_vertek,v_varos,v_email," +
            "v_kedvezmeny,v_euadoszam)" +
            " VALUES('"+vasarlo.getVa_nev()+"', '"+vasarlo.getVa_isz()+"', '"+vasarlo.getVa_cim()+"', '"+vasarlo.getVa_tel()+"', '"+
            +vasarlo.getVa_fax()+"', '"+vasarlo.getVa_aszam()+"', '"+
            +vasarlo.getVa_vertek()+"', '"+vasarlo.getVa_varos()+"', '"+
            +vasarlo.getVa_email()+"', '"+vasarlo.getVa_kedvezmeny()+"',
            +vasarlo.getVa_euadoszam()+"')");
        boolean uj = st.execute(sql);
        return true;
    }
    catch (SQLException ex) {
        return false;
    }
}
```

Hasonló módon történik a vásárló módosítása és törlése. Azokban a metódusokban csak az sql lekérés fog változni, a felépítése ugyanez marad. Fontos megjegyezni, hogy az itt felsorolt táblákat minden esetben reprezentálja egy osztály.

- A bevételezések során használt táblák

Módosítás történik a „torzsadat”, „raktar”, „beszallitok” táblákban, továbbá új sorokat hoz létre a „bevetelez”, „bevetszamol” és „beszallitoszamla” táblákban. Bevételez és bevételszámol táblák tárolják részletesen a beszerzéseket. A számlán szereplő anyagok tételesen kerülnek rögzítésre, ahogy a számlán szerepelnek. Minden sorhoz hozzá van rendelve a számla sorszáma és ez alapján azonosítható be, hogy melyik számlához tartozik.

Bevetelez tábla: primary_key(bsz_ekod)

Mezőnév	Típusa	Jelentése
bsz kod	integer	Számla kódja
bsz cikk	text	Termék cikkszáma
bsz bedb	real	Bevételezett mennyiség
bsz datum	date	Számla teljesítése
bsz ar	real	A termék ára
bsz ekod	serial	Sor egyéni kódja

Bevetszamol tábla: primary_key(bsz_ekod)

Mezőnév	Típusa	Jelentése
bsz cikk	text	Termék cikkszáma
bsz bedb	real	Bevételezett mennyiség
bsz kidb	real	Ebből kiszámlázott menny.
bsz datum	date	Teljesítés dátuma
bsz bear	real	Termék beszerzési ára
bsz kulonbseg	real	Bevételezett – Kiszámlázott
bsz ekod	serial	Sor egyéni kódja
bsz kod	integer	Beszállító számla kódja
bsz allapota	boolean	Tétel állapota

A bevételszámlol tábla abban különbözik a bevételeztől, hogy nyilvántartja az eladások során készletről való levételt. Mindig abból a készletből veszi el, amelyik korábban került beszerzésre. Tegyük fel, hogy egy rozsdamentes csőből egyszer érkezett 10 méter 998 Ft-ért. Következő szállításnál 12 méter érkezett, de 1001 Ft-ért. Majd az ezt követő szállításnál 6 méter 990 Ft-ért. Ekkor így néz ki a táblánk:

bsz cikk	bsz b	bsz k	bsz datum	bsz be	bsz ku	bsz ko	bsz a
001-052015-001	10	0	2008-10-10	998	10	1	TRUE

...

001-052015-001	12	0	2008-10-22	1001	12	12	TRUE
----------------	----	---	------------	------	----	----	------

...

001-052015-001	6	0	2008-11-03	990	6	33	TRUE
----------------	---	---	------------	-----	---	----	------

Vásárlás történik és ebből a termékből eladunk 12 métert 1290 Ft-ért. A program megkeresi a táblában a legkorábban bevételezett terméket és abból elvesz

annyit, amennyi még ebből a készleten szerepel. Jelen esetben az első sorból 10-et, majd a következő előfordulásnál 2-t. Mivel az első sorban mind a 10 eladásra került, ezért a `bsz_kidb` oszlopban az érték 10 lesz. A különbség 0-ra állítódik, ugyanis ez az érték a `bsz_bedb` – `bsz_kidb` adódik. Ha viszont a különbség 0, az azt jelenti, hogy ebből a bevételezett 998 Ft-os áruból már nincs raktáron. Ezért az állapot FALSE lesz, ami tudatja a programmal, hogy ezzel a sorral már nem kell számolni. A következő előfordulásnál az előbbieken leírtak alapján a `bsz_kidb` értéke 2 a különbség 10 lesz, viszont az állapot marad TRUE, mert az 1001 Ft-os csőből maradt raktáron 10 méter. A tábla még további fontos információt is tartalmaz az eladásra nézve (ár és kiszámolása), de ezt az értékesítésnél részletezem.

A számla sorszáma, teljesítése, értéke, stb. tárolására a „beszállitoszámla” tábla gondoskodik. Ennek a táblának a sorai jelennek meg bevételezett számlák menüpontban, ahol hivatkozva a kiválasztott számlára, a kódja alapján megkereshető a bevételez táblából a hozzá tartozó tételek. A tábla elsődleges kulcsa a program által automatikusan kapott számlasorszám.

Beszállitoszámla tábla: `primary_key(bsz_kod)`

Mezőnév	Típusa	Jelentése
<code>bsz_kod</code>	integer	Beszállító számla kódja
<code>szaamlaszam</code>	text	Az eredeti számlaszám
<code>beszaallito_kodja</code>	integer	Beszállítóhoz tartozó kód
<code>teljesites</code>	date	Teljesítés időpontja
<code>kelte</code>	date	Keltezés időpontja
<code>hatarido</code>	date	Fizetési határidő
<code>modja</code>	text	Fizetés módja
<code>erteke</code>	real	Számla értéke
<code>allapota</code>	text	Számla állapota

A bevételezés folyamán változik a törzsadat táblában az adott termék átlag beszerzési ára is, melyet szintén a bevetszamol táblából kap. Az előző példára visszatekinve, készleten maradt 10 méter cső az 1001 Ft-os tételből és 6 méter a 990 Ft-osból. Az értékük 15950 Ft, ezt osztja a készlettel, ami 16 méter. Átlag ár 996,88 Ft lesz. A folyamatnak itt nincs vége, a beszállítók táblában módosul a szállítócéghez tartozó eddig vásárolt érték a bevételezett számla értékével. Nyomon követhető, hogy

mennyit vásárolt a cég az aktuális beszállítótól. A készletre vett termékek mennyisége és összértéke hozzáadódnak a raktár tábla megfelelő cikkszámaihoz tartozó mezőkhöz.

- Értékesítés során használt táblák

Új tábla jelenik meg az „elabe”. Szerepe, az értékesített termékek figyelése.

Elabe tábla: primary_key(e_kod)

Mezőnév	Típusa	Jelentése
cikkszám	text	Termék cikkszám
mennyiség	real	Eladott mennyiség (részlet)
bear	real	Beszerzési ára
eladar	real	Eladási ára
o bear	real	Össz beszerzési ár
o eladar	real	Össz eladási ár
o kulonbseg	real	o eladar – o bear (árrés)
o szazalek	real	Haszon százalékban meghat.
e kod	serial	Sor egyéni kódja
szaamlaszam	text	A tételhez tartozó számlasz.
teljesites	date	Eladás teljesítése
ea ar lista	real	Eladási lista ár
kedvezmeny	integer	Kedvezmény mértéke

Nagyon részletes információval szolgál az értékesített árukról. Vegyük az előbbi példát. Az eladást követően a bevételszámlol táblából átemelt adatokat az elabé táblába helyezi. A sorok a következőképpen néznek ki:

cikkszám	menny.	bear	elada	o bear	o eladar	kul.	%
001-052015-001	10	998	1290	9980	12900	2920	23
001-052015-001	2	1001	1290	2002	2580	578	22
	e kod	szaamlaszam	teljesites	ea ar lista	kedvezm.		
	34	2008/00111	2008.11.20	1290	0		
	35	2008/00111	2008.11.20	1290	0		

Az elabé táblázatból kiolvasható, hogy az eladott 12 méter cső, hogyan lett megbontva. Számlán csak annyi fog látszani, hogy 12 méter cső 1290 Ft-os eladási áron lett értékesítve, viszont a táblázat ezt részletesebben tárolja. Első sorban kiolvasható, hogy 10 méter 998 Ft-ért érkezett és értékesítve lett 1290 Ft-ért. A második sorban van a maradék 2 méter, ami 1001 Ft-ért lett vásárolva és 1290 Ft-ért

lett értékesítve. Az össz beszerzési értékek és össz eladási értékek is beírásra kerülnek, ugyanis ebből lesz kiszámolva a különbség (árrés). Figyelemmel lehet kísérni, hogy milyen haszonnal lett értékesítve az eladott termék. Tárolásra kerül továbbá az a számlaszám, amelyen az adott tételek szerepelnek. Jelen esetben a számlaszám ugyanaz, mivel egy számlán szerepelnek. A teljesítés napja az a dátum, amikor elvitték a raktárról az árut. Ez lehet korábbi dátum is, mint a számla kiállításának ideje. Ebben az esetben szállítólevélre lett kiadva (a program nem kezeli a szállítóleveleket, így kézzel kiállított szigorú számadású tömbbe történik a kiírás). Az utolsó két mező szerepe a kedvezmény figyelése. A számla nyomtatványának készítésénél meg kell jeleníteni a lista eladási árat, majd ha arra jogosult a vásárló, akkor kedvezményben is részesíthető. A kedvezmény mértékét az utolsó mező tárolja. Természetesen, ha van kedvezmény, akkor az eladásiár mező a kedvezményes árat kapja meg értékként.

További módosított táblák: a raktár táblában csökkenti a mennyiségét és az értékét a számlán szereplő tételeknek. A vásárló táblában a vásárlóhoz tartozó vásárolt értéket növeli a számlán szereplő végösszeggel. A törzsadat táblában az értékesített tételek soraiban, újra kiszámolásra kerül a raktári készleten maradt mennyiségek átlagára. A „szamlak” tábla egy új sorral bővül.

Szamlak tábla: primary_key (szamlaszam)

Mezőnév	Típusa	Jelentése
szamlaszam	text	Számla sorszáma
v kod	integer	Vásárlót azonosító kódja
erteke	real	Számla nettó értéke
fiz mod	text	Fizetésének módja (kp,átut.)
teljesites	date	Áru kiadásának napja
kelte	date	Számla kiállításának napja
hatarido	date	Fizetési határidő
allapot	text	Állapot (élő vagy rontott)
megjegyzes	text	További megjegyzés a számlán
bruttó	integer	A számla áfával növelt értéke
bruttó szoveg	text	Végösszeg szöveggel kiírva

A számlaszám lesz a sorok azonosítója. A számlaszámnak folytonosnak kell lenni. Szigorúan követni kell a számozást. Mivel a típusa szöveges (String), ezért

egész értéké kell alakítani, és az értéket növelni eggyel. A táblázatból beolvasott számlaszám oszlop egy vektorba kerül számlaszám szerint rendezve. A vektor utolsó elemét kiemeli és szöveges típusú alakítja (legyen pl.: „2008/00001”). A string utolsó öt karakterét egész típusú alakítja, majd hozzáad egyet. A típus alakítás során a „00001”-ből 1 lesz. A nullákat a konvertálás során elhagyja. A kapott értéket megvizsgálja, hogy melyik tartományba esik, annak megfelelően újra szöveges típusú alakítja. Ezt példázza az alábbi java kód:

```
if ( !(kod.isEmpty()) ){
    String z = kod.lastElement().toString();
    int k = Integer.parseInt(z.substring(5)) + 1;
    if (k<10) szamla.setSorszam("2008/0000"+String.valueOf(k));
    if (k<100 && k>=10) szamla.setSorszam("2008/000"+String.valueOf(k));
    if (k<1000 && k>=100) szamla.setSorszam("2008/00"+String.valueOf(k));
    if (k<10000 && k>=1000) szamla.setSorszam("2008/0"+String.valueOf(k));
}
else{
    szamla.setSorszam("2008/00001");
}
```

A vásárló kódja mezőbe értelemszerűen a vásárlóhoz tartozó azonosító kód kerül. Ez a kód hivatkozik a számlán szereplő vásárlóra. A számla nyomtatványának kiállításánál a kód alapján lesz kiolvasva az adatbázisból a vásárló. Az érteke mezőbe kerül a számla nettó értéke, amely real (double) típusú, nem kerekített érték. A bruttó mezőbe a kerekített végösszeg kerül. A számla teljesítésének módja, teljesítése, keltezése, fizetés határideje kell a nyomtatvány kitöltéséhez. Az állapot mező új számla létrehozásánál automatikusan „élő”. Ha lerontásra kerül, akkor a létrejött új számla állapota „javító” az eredeti lerontott pedig „rontott” jelzőt kap. A megjegyzes mezőbe további számlával kapcsolatos megjegyzések kerülnek.

- Árajánlatkészítés során használt táblák

Az „arajanlat” tábla adatai az Árajánlatok menüpont alatt megjelenő táblázat kitöltésére lett létrehozva. A kiválasztott árajánlat sorszáma azonosítja az ajánlatot.

Arajanlat tábla: primary_key (aa_sorszama)

Mezőnév	Típusa	Jelentése
aa_sorszama	integer	Árajanlat sorszáma
aa_vasarlo	text	Árajanlatot kérő neve
aa_datum	date	Kiadás dátuma
aa_netto	real	Árajanlat áfa nélküli értéke
aa_brutto	real	Árajanlat áfával növelt ért.
aa_fizmod	text	Fizetési feltétel
aa_megjegyzes	text	Megjegyzés
aa_hatarido	text	Szállítási határidő

Az árajanlat nyomtatvány elkészítése az „arajanlattétel” tábla segítségével történik. A nyomtatványkészítő a megjelenítendő adatokat ebből a táblából olvassa ki.

Arajanlattétel tábla: primary_key() nincs

Mezőnév	Típusa	Jelentése
a_sorszam	integer	Árajanlat sorszáma
a_vasarlo	integer	Ajánlatot kérő kódja
a_datum	date	Kiadás dátuma
a_cikkszam	text	Termék cikkszáma
a_megnevezes	text	Termék neve
a_meret	text	Termék mérete
a_anyagminoseg	text	Termék anyagminősége
a_mennyiseg	real	Termék mennyisége
a_me	text	Mennyiségi egység
a_afa	smallint	Áfa mértéke
a_ar	real	Kiajánlási ára

- Beállítás során használt táblák

Beállítható az eladó adatai „elado” táblában és a számlák nyomtatásával kapcsolatos adatok a „nyomtatás” táblában.

Elado tábla: primary_key(e_cime)

Mezőnév	Típusa	Jelentése
e_neve	text	Eladó neve
e_irsz	text	Irányítószám
e_varos	text	Város
e_cime	text	Cím
e_adoszama	text	Adószám
e_bank	text	Bankszámlaszám

e telefon	text	Telefon
e fax	text	Fax
e email	text	E-mail cím
e web	text	Web cím
e euadoszam	text	Európa Unió Adószám
e iban	text	IBAN szám

Nyomtatás tábla: primary_key(ny_allapot)

Mezőnév	Típusa	Jelentése
ny_allapot	text	Nyomtatás témája
ny_peldany	smallint	Példányszám

A nyomtatás tábla első oszlopában van megadva, hogy mire vonatkozik a nyomtatás (pl.: átutalásos számla, készpénzes számla). A másik oszlop a nyomtatandó példányszámot tárolja.

3.2 Java kódrészletek bemutatása

A program írása során elég sok osztályt hoztam létre, melynek bemutatására nem lenne elég 100 oldal sem, viszont megpróbálom kiemelni azokat a kódrészleteket a forrásból, amelyek érdekesebbek (pl.: számlanyomtatvány megjelenítés). A kedvenc osztályom az „AdatbazisKezelo.java”, amely jelenlegi verziója 2275 soros. Ez az osztály építi fel és bontja le a kapcsolatot az adatbázissal, továbbá itt találhatóak meg azok a metódusok, amelyek a táblákból kiolvasnak adatokat, módosítják a táblát, törölnek a táblából sorokat, illetve újakat hoznak létre. Minden menüpont más-más érdekből akar a táblákhoz férni. Az egyik csak egy bizonyos sorra kíváncsi ugyanazon táblából, a másiknak meg a tábla minden sora kell.

Nulla készlet nulla érték:

Az alábbi metódus a számlázást követően hívódik meg, melynek szerepe, ha nulla készlet van egy termékből raktáron, akkor a hozzá tartozó értéket nullára állítja. A kerekítésekből adódóan előfordul, hogy tizedes értékek maradnak fent. Azoknak a termékeknek a cikkszámait kerülnek be egy vektorba, melyek nincsenek készleten. Ezt követően egy ciklus végigmegy ezeken a cikkeken és a hozzájuk tartozó értéket a raktártáblában kinullázza.

```

public void nullazNullaKeszletEsetenRaktarTablaban() {
    try {
        Vector <String> cikkszamok = new Vector();
        ResultSet raktar = st.executeQuery("SELECT * FROM raktar");
        while ( raktar.next() ){
            if ( raktar.getDouble("r_db") == 0 ){
                cikkszamok.add(raktar.getString("r_cikk"));
            }
        }
        raktar.close();
        for ( int i=0 ; i<cikkszamok.size() ; i++){
            String cikk = cikkszamok.elementAt(i);
            boolean a = st.execute("UPDATE raktar SET r_ertek=0 " +
                                   "WHERE r_cikk='"+cikk+"'");
        }
    } catch (SQLException ex) {
        Logger.getLogger(AdatbazisKezelo.class.getName())
            .log(Level.SEVERE, null, ex);
    }
}

```

Áfával növelt érték kerekítése:

A következő kódrészlet szintén az AdatbazisKezelo.java osztályban van létrehozva, amelynek célja, hogy a számlák végösszegét készpénzes fizetés esetén 0, 5 vagy 10 forintra kerekítse. Azért van erre szükség, mert a forgalomból ki lettek vonva az 1 és 2 forintos érmék. Csak a készpénzes fizetés esetén történik a kerekítés, átutalásos számláknál virtuálisan történik a pénzmozgás.

A metódus paraméterül megkapja az áfával növelt értéket és visszatér a kerekített egész értékkel. Az értéket stringgé alakítja, majd a tizedes pont szerint tömbösíti. A valós értéknél ez kettő darab tömböt jelent. Veszi az első tömbben lévő stringet, majd karaktertömbbé alakítja az itt lévő szöveget. Kiválasztja a tömb utolsó elemét, és szintén átkonvertálja egész értéké. Megvan a szám utolsó számjegye. Az if ágakban megtörténik a feltétel vizsgálat. Ha ez az érték 0 és 2 közzé esik, akkor a karaktertömb utolsó elemét '0'-ra állítja és konvertálja a tömböt stringgé és ezután egész értéké. Ha az érték 8 és 9, akkor csak egyszerűen a paraméterül kapott értékhez hozzáad 1-et vagy 2-t.

```

public int bruttoErtekKerekites(double ertek){

    int brutto = 0;

    String temp1 = String.valueOf(ertek);
    String temp2 = temp1.replace(".", "a");
    String[] temp3 = temp2.split("a");
    char[] c = temp3[0].toCharArray();
    String a = String.valueOf(c[c.length-1]);
    int i = Integer.parseInt(a);

    if ( i>=0 && i<=2 ){
        c[c.length-1] = '0';
        String b = String.valueOf(c);
        brutto = Integer.parseInt(b);
    }
    if ( i>=3 && i<=7 ){
        c[c.length-1] = '5';
        String b = String.valueOf(c);
        brutto = Integer.parseInt(b);
    }
    if ( i == 8 ){
        brutto = (int)ertek+2;
    }
    if ( i == 9 ){
        brutto = (int)ertek+1;
    }

    return brutto;
}

```

Táblázat adatainak a lekérése:

Ezt az eljárást a Szamlak.java osztály azért hívja meg, hogy megjelenítse a saját táblázatában a keresett számlákat. Ez igazából egy sql lekérés szintén az AdatbazisKezelo.java osztályból, melynek eredményéül egy eredmény táblát kap.

```

public ResultSet szamlakAVasarlovalEgyutt(){

    ResultSet a=null;
    try{
        a = st.executeQuery("SELECT sz.*,v.v_nev " +
            "FROM szamlak sz,vasarlok v WHERE sz.v_kod = v.v_kod");
    }catch (SQLException ex) {
        Logger.getLogger(AdatbazisKezelo.class.getName())
            .log(Level.SEVERE, null, ex);
    }
    return a;
}

```

Számlanyomtatvány készítés:

Számla készítésénél előzőleg létrejön egy számla objektum, amely tartalmazza a számlán szereplő tételeket, a vásárló adatait, a számla keltezését, teljesítését, fizetési határidejét, fizetés módját stb. Számlanyomtatvány előállításánál fontosak ezek az adatok. A nyomtatási példányszám kezdőértékként a 3-at veszi fel (általánosan használt). A következő lépésben megtörténik a kapcsolódás az adatbázishoz. Sikeres kapcsolat után egy stringbe menti a számla fizetésének módját, ami lehet átutalás bármelyik formája (pl.: 8 napos, 15 napos, azonnali átutalás, stb.) vagy készpénz. A nyomtatás nevezetű táblából kiolvassa a megfelelő sorhoz tartozó nyomtatás példányszámát. Ha a táblában a készpénzeshez 2 példány van beállítva és a számla fizetésének módja készpénz, akkor a példányszám értéke változik 2-re. Megtörténik a kapcsolat bontása az adatbázissal, ugyanis az adatok már rendelkezésre állnak.

```
int peldany_szam = 3;
if ( dbk.kapcsolodas(user, password) ){
    String all = szamla.getFiz_mod();
    ResultSet p = dbk.getNyomtatasiTabla();
    while ( p.next() ){
        String temp = p.getString("ny_allapot");
        if ( all.equals(temp) ){
            peldany_szam = p.getInt("ny_peldany");
        }
    }
}
dbk.kapcsBontas();

for ( int i=0 ; i<peldany_szam ; i++ ){

    Map parameters = new HashMap();
    parameters.put("SZAMLASZAM",szamla.getSorszam());
    parameters.put("PELDANYSZAM",peldany_szam);
    parameters.put("E_M_SZAMLA","Eredeti");
    parameters.put("V_PELDANYSZAM",i+1);
    parameters.put("MILYEN_SZAMLA","SZÁMLA");

    ReportKeszito report = new ReportKeszito();
    report.runReport("jdbc:postgresql://localhost:5432/Luinox", "postgres",
        "gportal", "D:/Programozas/Java/Raktar_program/Raktar/src/Report/" +
        "szamlai_Szamla_tetel.jrxml",parameters);
}
```

Következő lépésben egy for ciklusban előállítódnak a számlanyomtatványok, pontosan annyi, amennyi példányszámban meg kell jeleníttetni. Létrehoz egy riport (RiportKeszito) objektumot, ennek egy eljárását hívja meg (runReport()) a következő paraméterekkel. Első paraméter az adatbázis URL-je, következő az adatbázist használó neve, kódja, a riport fájl elérési útja és a riportfájlhoz tartozó paraméterek, melyeket egy HashMap-ben tárolunk.

Megjegyzés: a riportok dinamikus megjelenítéséhez elengedhetetlenek a paraméterek. Például a számla nyomtatványának alapja ugyanaz, de a rajta megjelenítendő adatok minden példányon más és más.

4. FELHASZNÁLÓI DOKUMENTÁCIÓ

Minden program fontos része a dokumentáció, hiszen nem sok haszna van egy olyan programnak, amelyet a felhasználó nem tud használni. Ebben a fejezetben bemutatásra kerül a program működése és funkcióinak használata.

Program használata

A program indítását követően megjelenik a főmenü, ahol minden fontosabb funkció külön almenüben helyezkedik el. Külön az adatkezelés, bevételezés, számlázás, lekérdezés, beállítások, árajánlatkészítés, telefonkönyv. Az adatkezelés menü pont alatt beállíthatóak a vásárlók, beszállítók, árutörzs adatai. Számlázás menüpont alatt készíthetünk új számlát, de korábbiakat is megtekinthetünk, kinyomtathatunk, vagy visszakereshetünk. Ebben a menüpontban tudunk számlát lerontani is. A bevételezés menüpontban a vásárolt termékeket tudjuk felvinni a raktárkészletbe. Továbbá, visszamenőleg kereshetünk beszállítói számlát, azt megtekinthetjük és kinyomtathatjuk. Az árajánlat menüpont alatt készíthetünk a vásárlók részére termékekről ajánlatot, ezt szintén megtekinthetjük és kinyomtathatjuk. Lekérdezések menüpont alatt különböző információkat kérhetünk

eladásokról, raktárkészletről, beszerzésekről. A telefonkönyv egy kis noteszként működtethető. Elmenthetjük, például egy fuvaros telefonszámát, akit megbízunk szállítással, ha egy vásárló kéri a vásárolt termék házhoz szállítását. Beállítások menüpont alatt, beállítható jelen verzióban az eladó adatai és a nyomtatandó példányszám átutalásos és készpénzes számlák esetén. Továbbiakban részletesen bemutatásra kerülnek a menüpontok.



4.1 Adatkezelés menüpont

Az itt található menüpontok segítségével állíthatók be vagy hozhatók létre olyan adatok (vásárlók, árutörzs), melyek a számlázásnál, bevételezésnél, árumozgásnál vagy akár az új árutörzs felvételénél a későbbiekben használni fog a felhasználó.

4.1.1 Törzsadatkezelés

Egy termék keletkezése ott kezdődik egy raktárprogramban, hogy létrehozzuk a törzsadatait. Egy új törzsadat létrehozásánál felhasználásra kerül az alábbi menüpontokban beállított alap adatok: árucsoport, árufőcsoport, méretek, anyagminőségek, mennyiségi egységek. Ezeknek az adatoknak a beállítását lentebb

lévő menüpontok alatt ismertetem. Ebben a menüpontban létrehozható, módosítható és törölhető az árutörzs.

Törzsadat Kezelő											
Cikkszám	Megnevezés	Méret	Anyagminős.	M.e	VTSZ/SZJ	Áfa	Nettó beár	Kisker ár	Törzsvásárlói ár	Áru csoport	Áru fő csop.
001-016015-001	Cső	16.0 x 1.5	1.4301	méter	7307	20	0.0	760.0	0.0	Cső	1
001-040015-001	Cső	40.0 x 1.5	1.4301	méter	7306	20	1001.12	1670.0	0.0	Cső	1
001-052015-001	Cső	52.0 x 1.5	1.4301	méter	7306	20	1399.08	2480.0	0.0	Cső	1
008-305102-001	Lemez 3C (matt)	0.5x1000x2000	1.4301	tábla	7306	20	1000.0	7800.0	8700.0	Lemez 3C (matt)	2
008-310102-001	Lemez 3C (matt)	1.0x1000x2000	1.4301	tábla	7306	20	0.0	12300.0	9999.0	Lemez 3C (matt)	2
200-040000-001	Ív	NA40	1.4301	darab	7307	20	999.0	1240.0	1000.0	Ív	3
200-050000-001	Ív	NA50	1.4301	darab	7307	20	0.0	1420.0	0.0	Ív	3
200-108020-001	Ív	108.0 x 2.0	1.4301	darab	7307	20	0.0	1303330.0	0.0	Ív	3
201-020000-001	T-idom	NA20	1.4301	darab	7307	20	0.0	1880.0	0.0	T-idom	3
201-114320-001	T-idom	114.3 x 2.0	1.4301	darab	7307	20	0.0	0.0	0.0	T-idom	3
213-016020-001	Bördel	54.0 x 2.0	1.4301	darab	7301	20	0.0	1230.0	1030.0	Bördel	9
300-050000-005	Pillangószelep H/H	NA50	1.4404	darab	8481	20	0.0	12310.0	0.0	Pillangószelep H/H	5
302-050000-005	Pillangószelep M/M	NA50	1.4404	darab	8481	20	0.0	11200.0	0.0	Pillangószelep M/M	5
303-010000-001	Golyószelep	NA10	1.4301	darab		20	0.0	56777.0	0.0	Golyószelep	5
403-050000-001	Kúposvég	NA50	1.4301	darab	7307	20	0.0	1540.0	0.0	Kúposvég	4
404-020000-001	Menetesvég	NA20	1.4301	darab	7307	20	0.0	678.0	0.0	Menetesvég	4
404-032000-001	Menetesvég	NA32	1.4301	darab	7307	20	0.0	788.0	0.0	Menetesvég	4
404-050000-001	Menetesvég	NA50	1.4301	darab	7307	20	0.0	1540.0	0.0	Menetesvég	4
405-040015-005	Clampbilincs	40.0 x 1.5	1.4404	darab	7307	20	0.0	0.0	0.0	Clampbilincs	4
703-014020-A...	Békazár	14.0 x 2.0	Farok	darab	jamesBond...	20	0.0	2340.0	0.0	Békazár	10
703-310122-002	Békazár	60 x 40 x 1.5	A2	darab		20	0.0	988.0	902.0	Békazár	10
712-305102-001	Awab Bilincs	0.5x1000x2000	1.4301	méter		20	0.0	0.0	0.0	Awab Bilincs	10
999-000000-998	123456789012345678901234567890	123456	123456	karton	1234567890	20	0.0	234567.0	0.0	Fűrészfűr Cobaltos HSS	8
999-000000-999	Téj	1 literes	2,4%	karton	111	12	0.0	200.0	120.0	Egyéb	12

Új Törzs

Módosítás

Törölés

Kiválasztott törzs: 008-310102-001 Lemez 3C (matt) 1.0x1000x2000

Rendezés:

Cikkszám

Kilép

Új törzsadat létrehozása:

Fontos, hogy legyen feltöltve az árucsoport, áru főcsoport, anyagminőségek, méretek táblázat. Ha itt nincsenek megadva adatok, akkor az új törzsadat választásánál üres listák jelennek meg és nem hozható létre új árutörzs. Ha egy termék nem rendelkezik méret és anyagminőség paraméterekkel, akkor érdemes létrehozni mind a kettőben egy „nincs” sort. Ha ezt választjuk az új törzs létrehozásánál, akkor ebben az oszlopban üres mezők lesznek.

Új Törzsadat felvitele

Árucsoport: Cső

Áru főcsoport: Szál anyag

Áru megnevezése: Cső

Méret: 13.0 x 1.0

Mennyiségi egység: méter

Beszerzési ár (nettó): 320 Forint

Eladási ár (nettó): 440 Forint

Törzsvás. ár (nettó): 400 Forint

Áfa: 20 %

Minimál készlet: 6

Anyagminőség: 1.4301

Cikkszám: 001 013010 001

VTSZ / SZJ: 7307

Mégsem

Új Törzs Felvitel

Meg kell adni elsősorban, hogy melyik árucsoporthoz tartozik a termék. Ha nincs még árucsoport megadva, akkor nem lehet tovább haladni az árutörzs létrehozásában. Továbbá, ha nincs olyan árucsoport, amelyikbe az új termék tartozik, akkor létre kell hozni. Ha sikerült kiválasztani a megfelelőt, akkor automatikusan kiírja, hogy melyik áru főcsoporthoz tartozik a termék (árucsoport menüpont alatt állítható be). Az áru megnevezéséhez a kiválasztott árucsoport neve kerül, de ez megváltoztatható. Ez az automatikus beállítás azért fontos, hogy a megnevezések rendezettek legyenek. Ha keresünk egy terméket a neve alapján és ebből több is van a táblában csak más méretezéssel, akkor jó, ha egyforma a megnevezés. Könnyen előfordulhat, hogy leütünk egy karaktert és nem találjuk a termékek között. Ebből, pedig gondunk is származhat egy rendelésnél, mert azt gondoljuk, hogy nincs és rendelünk belőle, pedig van, de valahol a sok termék között eltűnt.

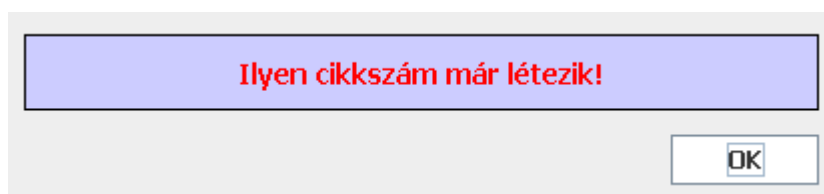
Meg kell adni továbbá a termékhez tartozó méretet. Ha nem rendelkezik ilyen paraméterrel, akkor válasszuk ki a „nincs”-et a listából, ha már korábban ezt

beállítottuk a méretek táblában. Itt is fontos az, hogy ha nincs a mérettáblában létrehozva egyetlen egy sor sem, akkor nem tudunk a továbbiakban létrehozni új törzset. Ez vonatkozik a következő lépésben megadható mennyiségi egységre is (darab, méter, karton, stb.).

Meg lehet adni továbbá a beszerzési árat, eladási árat és törzsvásárlói árat. Ennek a pontnak a kitöltése nem kötelező, hiszen bevételezésnél is módosulhat egy áru beszerzési ára, illetve eladásnál is bármikor módosítható.

Beállítható még az áfa, minimál készlet és az áruhoz tartozó vámtarifaszám. Minimál készlet értéke figyelj, hogy a termék mennyisége a raktárban megfelelő legyen. A lekérdezéseknél megtekinthető, hogy melyik termék került már ezen érték alá. Amelyik szerepel a listán, abból akár rendelhető is.

A cikkszám a beállítások során automatikusan generálódik. Három nagyobb csoportból tevődik össze. Az első az árucsoportból, a második a méretből és a harmadik az anyagminőségből. Mind a három táblában a sorokhoz rendelve van egy cikkszám, ami ide bekerül. Ez biztosítja, hogy két ugyanolyan termék ne kerüljön be a törzsbe. De azaz előnye is megvan, hogy a felhasználó nem fogja például a csövek közzé felvenni a vágókorongot. A cikkszám akár átírható is. Fontos, hogy minden terméknek egyedi, más cikkszáma legyen. Ha már létezik termék a megadott cikkszámon az nem fog bekerülni az új árutörzsbe, „Ilyen cikkszám már létezik!” hibaüzenettel megáll, ellenkező esetben „Új törzs felvétele sikeres!” üzenetet kapja.



Törzsadat módosítása:

A táblázatból ki kell választani a módosítandó árutörzset. Ha nincs kijelölve és úgy kattint a „Módosítás” gombra, hibaüzenetet kap „!Nincs kijelölve sor!” szöveggel. A kiválasztott árutörzs megjelenik a táblázat alatt. A „Módosítás” gombra

kattintva a megjelenő ablakban a megfelelő mezők kitöltve jelennek meg. Bármilyen módosítás itt hajtható végre. Átírható a megnevezés, méret, anyagminőség stb. Módosítani lehet az árucsoporton, átírható az áfa mértéke is. A „Jóváhagyás” gomb hatására megtörténik a módosítás. A „Sikeres módosítás!” üzenettel tudatja a program, hogy megtörtént a módosítás.

Törzsadat törlése:

Árutörzset a felhasználó előre is generálhat. Például szabvány szerint létezik nagyon sok átmérőjű rozsdamentes cső. A felhasználó előre létrehozza több ilyen árunak a törzsét, hogy a későbbiekben ne kelljen ezzel foglalkozni. Van olyan eset, amikor az ilyen árutörzsekre évekig nem is érkezik áru és feleslegessé válik, hogy sok törzs szerepeljen a nyilvántartásban, így azok törlésre kerülnek. Csak ezek a törzsek törölhetőek. Ha már egy árutörzsszel történt bevételezés az már nem törölhető. Azért nem törölhető, mert a későbbiekben az árumozgást követő lekérdezések az áru törzsét ebből az adathalmazból olvassák ki.

4.1.2 Vásárlók és beszállítók adatainak kezelése

A két menüpont hasonló szisztéma alapján működik. Felvehető új vásárló vagy beszállító, módosítható illetve törölhető.

Új vásárló vagy beszállító felvétele:

Értékesítés során elég gyakran érkeznek új vásárlók, ami csak jó egy kereskedelmi cégnek, mert egyre többen ismerik meg és térnek be vásárolni. Továbbá a kereskedelmi cégek több beszállító céggel tartják a kapcsolatot, viszont kutatnak is még olcsóbb beszerzési forrás után. Ezek új beszállítókat jelentenek. Adataikat ebben a menüpontban vagy a számlakészítés ill. a bevételezés menüpontban vehetjük fel az adatbázisba. A megjelenő ablakok sorait értelemszerűen ki kell tölteni. Fontos, hogy meg kell adni a vásárló nevét. Kihagyása során hibaüzenetet kap a felhasználó. A táblázatba ugyanis nem szerepelhet egy teljesen üres sor.

Megadható továbbá a vásárló vagy beszállító címe, telefonszáma, fax száma, e-mail címe, adószáma, európa unios adószáma és a kedvezménye. A kedvezmény mértéke a számlázás során minden tétel felvételénél automatikusan megjelenik. A „Jóváhagy” gomb hatására bekerül az adatbázisba, ha „Új vásárló felvétele sikeres!” üzenetet küldi vissza a képernyőre.

Vásárló vagy Beszállító adatainak módosítása:

Telephely címének változásánál vagy névváltozásnál szükséges, hogy tudja a felhasználó módosítani az adatokat. A kiválasztott sort követően megjelenik a táblázat alján a kiválasztott vásárló vagy beszállító. Csak kiválasztás után lehet módosítani, különben hibaüzenetet küld a program „Nincs kiválasztva sor!”.

Vásárló adatainak módosítása

Vásárló neve:

Irányítószám:

Város:

Cím:

Telefon:

Fax:

E-mail:

Adószám:

Kedvezmény: %

EU-s adószám:

A „Módosít” gombra kattintva a megjelenő ablakban a megfelelő mezők kitöltve jelennek meg. Átírható, kitölthető bármelyik sor tetszőlegesen. A program figyeli a beírható karakterek számát, ezért ha több karakter kerül be egy mezőbe, akkor figyelmeztető üzenetet küld. A „Jóváhagy” gomb aktiválásával megtörténik a módosítás, melyet egy üzenet kísér, „Sikeres módosítás!”.

Vásárló vagy Beszállító adatainak törlése:

A törlés csak akkor történik meg, ha ki van választva a vásárló vagy beszállító és nem történt eladás vagy bevételezés vele. Szintén ugyanaz a helyzet fordul elő, mint a törzsadat törlése esetén. A későbbi lekérdezések során a megfelelő helyre nem kerülnek adatok és ez hibát jelentene illetve üres kitöltetlen mezőket.

4.1.3 Árucsoport és árufőcsoport adatainak kezelése

A termékeket csoportosítani lehet. A rozsdamentes iparban is nagyon sokféle terméket különböztetünk meg. Ilyenek például a rozsdamentes csövek, ívek, lemezek, csavarok stb. Ezeket is lehet még tovább csoportosítani. Például a csöveket, zártszelvényeket, köracélokat besoroljuk a szálanyagok közé. Ezek lesznek a főcsoportok. Új árutörzs felvételénél kiválasztható az árucsoport, mellyel könnyebben tudunk létrehozni új törzset. Előnye, hogy egy csoportba lévő terméknek a megnevezése ugyanaz lesz, csak méretükben vagy anyagminőségükben térnek el és nem keveredik el a több száz termék között.

Új árucsoport és árufőcsoport létrehozása:

Új árucsoport felvételénél meg kell adni az árucsoport nevét. Rendelhető hozzá egy árufőcsoport azon árufőcsoportok közül, melyek már az adatbázisban szerepelnek. Ha a termék olyan főcsoportba tartozik, amelyik még nincs felvéve,

akkor azt könnyedén felvehetjük az árufőcsoport menü alatt. Továbbá, meg kell adni egy cikkszámot, ami egyben az árucsoport azonosítója és az automatikus cikkszámkészítés első három karaktere lesz. A cikkszám kiválasztásánál segítséget ad a „Foglalt cikkszámok” ikon. Erre kattintva a foglalt cikkszámok tekinthetők meg. Ezeket már nem lehet megadni egy új árucsoportnak. A „Jóváhagy” gomb hatására megtörténik az új árucsoport felvétele. Itt is visszajelzést küld a program a sikeres és sikertelen felvételtől. Új árufőcsoportnál csak a főcsoport nevét kell megadni. Ebben az esetben is az üres mezőt hibának tekinti, tehát csak kitöltve fogadja el.

Árucsoport és Árufőcsoport adatainak módosítás:

Csak kijelölt árucsoport és árufőcsoport esetén történik módosítás. A kijelölt sor adatai megjelennek a táblázat alatt. Az „árucsoport módosítása” ablakban módosítható az árucsoport esetén az árucsoport neve. Továbbá beállítható, hogy melyik főcsoporthoz soroljuk be. A hozzá tartozó cikkszám nem módosítható. Árufőcsoport módosítása esetén a főcsoport neve írható át.

Árucsoport és Árufőcsoport adatainak törlése:

A törlés teljes mértékben engedélyezett. Az innen bekerült adatokat a törzsbe a törzsadat tábla tárolja, hivatkozás ide nem mutat vissza. A megfelelő sor kiválasztása után történhet meg a törlés.

4.1.4 Méretek és anyagminőségek

Egyes termékek csak méretükben vagy anyagminőségükben térnek el. Egy rozsdamentes cső lehet akár 6 milliméter átmérőjű vagy akár 154 milliméter is. De azonos átmérőnél is lehetnek különbségek. Például egy 154 milliméter átmérőjű csőnek akár lehet a falvastagsága 1,5 vagy 2 milliméter is. Anyagminőségeknél is hasonló a helyzet. Egy azonos mérettartományban lévő cső csak abban tér el egy másiktól, hogy más anyagminőséggel gyártották. Ezeket a különböző anyagminőségeket is szabványokkal látták el.

A törzsadatban külön kerül tárolásra a termék mérete és anyagminősége. Egyrészt azért is, mert az automatikus cikkszám előállításánál a középső hat karakter a mérethez rendelt cikkszám lesz, amíg az utolsó három, pedig az anyagminőséghez rendelt cikkszám.

Új méret és anyagminőség felvétele:

Méret felvétele esetén meg kell adni egy maximum hat karakterből álló nem foglalt cikkszámot. Ha nincs hat karakter, a program nem fog hibaüzenetekkel visszatérni. Az átláthatóság és logikus cikkszám felépítés végett érdemes kihasználni a karakterek adta lehetőséget. A következő mezőben beírható a méret megnevezése. Az „OK” gombra kattintva megtörténik az új méret felvétele az adatbázisba. Visszajelzést küld a program a sikeres vagy sikertelen felvételtől egyaránt. Az anyagminőség felvétele esetén ugyanez az eljárás, csak itt maximum háromkarakterű cikkszámot kell megadni.

Módosítás és törlés:

A kijelölt méret vagy anyagminőség adatai megjelennek a táblázat alatt. Szintén csak kijelölés után lehetséges a módosítás vagy törlés. Méret és anyagminőség módosítása esetén a cikkszám nem módosítható, csak a cikkszámhoz tartozó méret írható át. A sikeres vagy sikertelen módosításról vagy törlésről visszajelzést küld a program a kijelzőre.

4.1.5 Mennyiségegységek

A vásárolt termékeket mennyiségük alapján bevételezzük. A termék jellegzetessége, hogy milyen mennyiségegységre lehet bontani. A szálanyagokat (csövek, zártszelvények, köracélok) méterben kerülnek bontásra, a lemezek kilogrammra, a szerelvények darabszámra stb. Ebben a táblázatban hozhatók létre, módosíthatók és törölhetők ezek az egységek. Például: kg, méter, tábla, darab, stb.

4.2 Bevételezés menüpont

A bevételezés menü alatt választható a „Számla bevételezése”, „Bevételezett számlák”, „Bevételezett számla törlése”. Első menüpont alatt a beérkezett árukat lehet raktárkészletre vinni. A következő menüpontban meg lehet tekinteni az eddig már bevételezett számlákat. Utolsó menüpont alatt törölni lehet egy hibás vagy rontott bevételezést.

4.2.1 Számla bevételezése

Megjelennek azok a beszállítók, akik már korábban felvételre kerültek az adatkezelés menüben vagy egy korábbi bevételezés során létre lettek hozva. Esetleg egy üres tábla, ha még eddig nem történt volna készletrevételezés. Lehetőség van ebben az ablakban is új beszállítót felvinni, ha egy új cégtől történik a vásárlás. Módosítani is lehet a beszállító adatait, ha például megváltozott volna a címe vagy egyéb adata. A beszállító kiválasztását követően annak adatai megjelennek a táblázat felett. A bevételezést is csak kiválasztott beszállítóval lehet elindítani.

Áru bevételezés								
Corarmatura Kft. 1113 Budapest Hamzsabégi út 60/F Tel.: 1 - 253 - 6583 Fax: 1 - 253 - 6582			Sorszám: 14 <input checked="" type="checkbox"/> Átutalás <input type="checkbox"/> Készpénz			Teljesítés napja: 2008.10.10. Számla kelte: 2008.10.10. Fizetési határidő: 2008.10.10. Számla száma: 2008/00123-V01		
Cikkszám	Megnevezés	Méret	Anyagminőség	Mennyiség	M.e.	Kiskerár	Érték	
001-016015-001	Cső	16.0 x 1.5	1.4301	0.0	méter	760.0	0.0	
001-040015-001	Cső	40.0 x 1.5	1.4301	12.3	méter	1670.0	12313.8	
001-052015-001	Cső	52.0 x 1.5	1.4301	72.2	méter	2480.0	101014.0	
008-305102-001	Lemez 3C (matt)	0.5x1000x2000	1.4301	10.0	tábla	7800.0	10012.0	
008-310102-001	Lemez 3C (matt)	1.0x1000x2000	1.4301	0.0	tábla	12300.0	123.0	
200-040000-001	Ív	NA40	1.4301	10.0	darab	1240.0	9990.0	
200-050000-001	Ív	NA50	1.4301	0.0	darab	1420.0	0.0	
200-108020-001	Ív	108.0 x 2.0	1.4301	0.0	darab	1303330.0	0.0	
201-020000-001	T-idom	NA20	1.4301	0.0	darab	1880.0	0.0	
201-114320-001	T-idom	114.3 x 2.0	1.4301	0.0	darab	0.0	0.0	
213-016020-001	Bördel	54.0 x 2.0	1.4301	0.0	darab	1230.0	0.0	
300-050000-005	Pillangószelep H/H	NA50	1.4404	0.0	darab	12310.0	0.0	
302-050000-005	Pillangószelep M/M	NA50	1.4404	0.0	darab	11200.0	0.0	
303-010000-001	Golyószelep	NA10	1.4301	0.0	darab	56777.0	0.0	
403-050000-001	Kúposvég	NA50	1.4301	0.0	darab	1540.0	0.0	
Új törzs Törzs módosít Tétel felvétele								
Kiválasztott tétel: 001-052015-001 Cső 52.0 x 1.5 1.4301								
Nettó összesen: 35520.0						Számlatétel Bevételezés Kilép		

A bal felső sarokban megjelenik a szállító cég fontosabb adataival. A jobb felső sarokban meg kell adni a teljesítés napját, a számla keltezését, fizetési határidejét és a számla számát. Ha a dátum mezők nincsenek kitöltve, akkor nem történik meg a bevételezés. Hibaüzenetet küld a felhasználónak „Nincs dátum megadva!” üzenettel. A számla sorszámozása automatikus. Ha nem történt még bevételezés, akkor 1-es sorszámmal látja el. Megadható még középen fent a továbbiakban a követelés módja (készpénz vagy átutalás).

Középen a táblázatban megjelennek az árutételek. Ha olyan termék érkezik, amelyiknek nincs még árutörzs, akkor felvehető új törzs itt is vagy a meglévő módosítható. Ezután bevételezhető tételenként a szállítói számla úgy, hogy ki kell választani az árutörzset a táblázatból és a „Tétel felvétele” gombra kattintani az egérrel.

200-050000-001 Ív NA50 1.4301 7307 20 %

Bevételezni: darab Nettó vételi ár: HUF / darab

A megjelenő új ablakba be kell írni a kiválasztott tétel darabszámát és nettó vételi árát, majd a „Bevételez” gombra kell kattintani. Ha valamelyik mező üres, hibaüzenetet küld a program „Üres adatmező!” üzenettel. Ezután ez a tétel bekerül a számlatételbe, ugyanígy az ezt követő többi bevételezett tétel. A bal alsó sarokban megjelenik a bevételezett tételek nettó összértéke. A „Számlatétel” gombra kattintva meg lehet nézni, hogy milyen tételek szerepelnek már a beszállítói számlán.

Bevételezett tételek							
Cikkszám	Megnevezés	Méret	Anyagminőség	Mennyiség	M.e.	Beszerz.ára	Érték
001-052015-001	Cső	52.0 x 1.5	1.4301	24.0	méter	1480.0	35520.0
200-050000-001	Ív	NA50	1.4301	100.0	darab	1003.5	100350.0
300-050000-005	Pillangószelep H/H	NA50	1.4404	2.0	darab	9988.0	19976.0
404-050000-001	Menetcső	NA50	1.4301	10.0	darab	998.8	9988.0

Kiválasztott tétel:

Nettó összértéke:

Hibás árutétel felvételét még van alkalom törölni. Továbbiakban összevonja, ha egy azonos cikkszámú tétel külön-külön kerül felvételre egyforma áron (Pl: Ív NA50-est elsőnek 40 darabot 1003,5 Ft-ért, másodjára 60 darabot szintén 1003,5 Ft-ért).

Ha minden megegyezik a „Bevételezés” gombra kattintva megtörténik az áruk készletrevétele és növeli a beszállító céghez tartozó vásárolt értéket a számla értékével.

4.2.2 Bevételezett számla megtekintése

Visszamenőleg is megtekinthetők a korábban bevételezett beszállítói számlák. A táblázatban megjelent számlák áfa nélküli összértéke megjelenik a bal alsó sarokban. Lehetőséget ad a menüpont, különböző keresésekre és csoportosításra. Megadható egy időintervallum pl: 2008.01.01. – 2008.10.10. A dátumformátumnak meg kell egyeznie a példában leírttal, ugyanis nem fogadja el a program, ha eltérő. Ez vonatkozik minden későbbiekben megadható dátumra! Továbbá, a számla akár az eredeti számlán lévő számlaszám alapján is kikereshető, ha az a bevételezés során helyesen lett megadva. A harmadik lehetőség a beszállítók alapján csoportosítás. A leugró listában csak azok a beszállítók szerepelnek, akik szállítottak már a cégnek. Az itt megadott szállító cég összes eddig bevételezett számlája megjelenik.

Bevételezett számlák

☒ Keresés szállító szerint Fantom Kft. ▼
☐ Keresés számlaszám alapján
☐ Keresés dátum szerint -

Számla kódja	Beszállító	Számla száma	
16	Fantom Kft.		2008-01-12
17	Fantom Kft.	2008/00012	2008-01-31
18	Fantom Kft.	2008/00027	2008-02-26
19	Fantom Kft.	2008/00122	2008-05-20
20	Fantom Kft.	2008/00333	2008-09-10

A táblázatban megjelenik a program által automatikusan adott egyedi azonosító a számla kódja, beszállító neve, eredeti számlán szereplő számlaszám, teljesítés

dátuma és a számla áfa nélküli értéke. Ha sikerült megtalálni a keresett számlát a „Számla megtekint” gombra kattintva megnézhető annak tartalma.

Beszállító Számla

Beszállító: Fantom Kft.
Cím: 4001 Debrecen
 Budapesti út 23.
Számlaszám: 2008/00122
Nyilvántartási 19 **Teljesítés kelte:** 2008/05/20
Fizetés módja: Átutalás **Kiállítás kelte:** 2008/05/20
Számla állapota: élő **Fizetési határidő:** 2008/06/20

Cikkszám	Megnevezés	Méret	Anyagminőség	
		Mennyiség	Egység ár	Nettó Értéke
200-040000-001	Ív	NA40	1.4301	
		20,00	677,00	13540,00
200-050000-001	Ív	NA50	1.4301	
		20,00	977,00	19540,00
403-050000-001	Kúposvég	NA50	1.4301	
		10,00	455,00	4550,00
404-050000-001	Menetesvég	NA50	1.4301	
		10,00	1266,20	12662,00
001-040015-001	Cső	40.0 x 1.5	1.4301	
		30,00	1102,40	33072,00
Számla Nettó Értéke:				83364,00
Számla bruttó értéke:				100036,80

4.2.3 Bevételezett számla törlése

Ki kell választani a törlendő beszállító számlát a táblázatból és a „Számla törlése” gombra kattintani. Sikeres törlés esetén üzenetet küld a program és leemeli a készletről a számlán szereplő tételeket. Továbbá a számla összértékével csökkenti a beszállítócégtől vásárolt értéket.

„Nincs kiválasztva számla vagy már rontott!” hibaüzenet. Egyrészt azért küldheti a program, ha a számla már egyszer le lett rontva. A raktárkészlet nem

egyezne, ha ismét le lehetne rontani. Másrészt akkor sem engedélyezi, ha a számlán szereplő tételből már került értékesítésre termék.

4.3 Számlázás menüpont

A számlázás menü alatt választható „Számlakészítés”, „Számlák” és „Számla törlése”. Első menüpont alatt készíthető el a számla, melyet a vásárlás alkalmával állít ki az eladó. Következő menüpont alatt megtekinthetők és kikereshetők a kiállított számlák. Az utolsó menüben törölhetők a rosszul kiállított számlák.

4.3.1 Számlakészítés

A vásárlás alkalmával számlát kell kiállítani a vásárolt termékről, termékekről. A megjelenő új táblázat a vásárlókat tartalmazza minden fontos adataikkal. Ha új vásárló érkezik, akkor a táblázat alatt lévő „Új vásárló” gombra kattintva lehet új vásárlót felvenni a listába. Továbbá lehetséges a vásárló adatainak módosítása ebben a menüpontban is, így nem kell kilépni és az adatkezelőbe beállítani az adatokat. A kiválasztott vásárlót követően „Számlakészítés” gomb hatására lép a Számlakészítés ablakba.

Számlakészítés							
Koroknai Attila 4027 Debrecen Bőszörményi út 22.						Számla sorszáma: 2008.00009	
Cikkszám	Megnevezés	Méret	Anyagminőség	Mennyiség	M.e.	Kiskerár	
001-016015-001	Cső	16.0 x 1.5	1.4301	0.0	méter	760.0	
001-040015-001	Cső	40.0 x 1.5	1.4301	41.3	méter	1670.0	
001-052015-001	Cső	52.0 x 1.5	1.4301	64.2	méter	2480.0	
008-305102-001	Lemez 3C (matt)	0.5x1000x2000	1.4301	5.0	tábla	7800.0	
008-310102-001	Lemez 3C (matt)	1.0x1000x2000	1.4301	0.0	tábla	12300.0	
200-040000-001	Ív	NA40	1.4301	120.0	darab	1240.0	
200-050000-001	Ív	NA50	1.4301	70.0	darab	1420.0	
200-108020-001	Ív	108.0 x 2.0	1.4301	0.0	darab	1303330.0	
201-020000-001	T-idom	NA20	1.4301	10.0	darab	1880.0	
201-114320-001	T-idom	114.3 x 2.0	1.4301	0.0	darab	0.0	
213-016020-001	Bördel	54.0 x 2.0	1.4301	0.0	darab	1230.0	
300-050000-005	Pillangószelep H/H	NA50	1.4404	0.0	darab	12310.0	
302-050000-005	Pillangószelep M/M	NA50	1.4404	0.0	darab	11200.0	
303-010000-001	Golyószelep	NA10	1.4301	0.0	darab	56777.0	
403-050000-001	Kúposvég	NA50	1.4301	20.0	darab	1540.0	
404-020000-001	Menetesvég	NA20	1.4301	0.0	darab	678.0	
404-032000-001	Menetesvég	NA32	1.4301	0.0	darab	788.0	
404-050000-001	Menetesvég	NA50	1.4301	20.0	darab	1540.0	
405-020010-004	Clampbilincs	20.0 x 1.0	1.4401	0.0	darab	0.0	
405-040015-005	Clampbilincs	40.0 x 1.5	1.4404	0.0	darab	0.0	
515-000000-001	Csiszoló szalag		1.4301	0.0	méter	0.0	

Törzsadat módosítás

Kiválasztott termék: 200-040000-001 Ív NA40 1.4301

Tételek hozzáad

Megjegyzés: [A számla a 123456 szállítólevél alapján készült kiállításra]

Számla tétel Számlázás Kilép

A felső információs sávba megjelenik a vásárló neve és címe, illetve a számla sorszáma. A sorszámozás automatikusan történik, ugyanis a szigorú számadást úgy kell biztosítani, hogy a számlát készítő program a sorszámozást kihagyás és ismétlés nélkül biztosítsa és az a felhasználó által nem módosítható.

Középen a táblázatban jelennek meg a termékek a készleten lévő mennyiségükkel együtt. Látható a termék cikkszáma, megnevezése, eladási ára is. A kiválasztott termék adatai megjelennek a táblázat alatt. Ha nincs kiválasztva sor, akkor „Tételt hozzáad” gombra hibaüzenetet küld a program.

A kiválasztott termék számlára helyezése a következő módon történik. A „Tételt hozzáad” gombra kattintva megjelenik egy új ablak, ahol megadható a vásárolt mennyiség és nettó eladási ár. Az eladási ár mező kitöltve jelenik meg, az árutörzsből olvassa ki az adatot. Az engedmény mező is automatikusan töltődik ki (a vásárlóhoz tartozó kedvezmény), de ez tetszés szerint módosítható. Megoldható az is egy számlán, hogy a tételek közül csak egy termékre ad az eladó kedvezményt (kicsit horpadt a cső). Egyik nagy előnye a programnak, hogy soronként is lehet a termékekre kedvezményt adni.

200-050000-001 Ív NA50 1.4301 7307									
Mennyiség:	<input type="text" value="20"/>	darab	Engedmény:	<input type="text" value="0"/>	%	Nettó eladási ár:	<input type="text" value="1420.0"/>	<input type="button" value="Tételt hozzáad"/>	<input type="button" value="Kilép"/>

A mennyiség megadásánál, ha többet ír be a felhasználó, mint a készleten lévő mennyiség, akkor figyelmeztető üzenetet kap, hogy „Nincs ennyi raktáron!”. A program nem engedi meg, hogy a raktárkészlet negatív értéket vegyen fel. Továbbá figyelmeztetést küld, ha olcsóbban kerül értékesítésre a beszerzési árhoz képest. Engedélyezi az eladást, de előtte figyelmezteti az eladót.

Az eladott áru felvételét követően megtekinthető a számlán szereplő tételek a „Szamlatétel” gombra kattintva. Ha a vásárló meggondolja magát és egy terméket csak nem vesz meg, akkor ebben az ablakban kitörölhető az árutétel. Információ olvasható le a számla áfa nélküli és áfával növelt értékéről a táblázat alatt.

Számlán szereplő tételek							
Cikkszám	Megnevezés	Méret	Anyagminőség	Mennyiség	M.e.	Egység ára	Áfa
200-050000-001	Ív	NA50	1.4301	20.0	darab	1420.0	20
200-050000-001	Ív	NA50	1.4301	10.0	darab	1278.0	20
404-050000-001	Menetesvég	NA50	1.4301	5.0	darab	1232.0	20
200-050000-001 Ív NA50 1.4301							

Nettó értéke: Bruttó értéke:

Az ablak alsó részén található egy megjegyzés rovat. Ide bármilyen megjegyzés írható a számlával kapcsolatban. Például: a számla az 123456 sorszámú szállítólevél alapján került kiállításra. Ez a megjegyzés is meg fog jelenni a számlán, amely a vásárlónak szintén egy információ. Ha elkészült a számla, akkor a „Számlázás” gombra kattintva a következő ablak jelenik meg.

Számla teljesítésének napja:	<input type="text" value="2008.11.16."/>
Számla készítésének napja:	<input type="text" value="2008.11.16."/>
Számla fizetésének határideje:	<input type="text" value="2008.11.16."/>
<input type="text" value="Készpénz"/>	
<input type="button" value="Mégsem"/> <input type="button" value="Ok"/>	

Kitöltve jelennek meg a dátum mezők. Az aktuális napra állítja be mind a teljesítést, számlakeltét és fizetés határidejét. A teljesítés időpontja változtatható. Egy kézzel megírt szállítólevél kiszámlázása esetén, amit a számlázás előtt egy-két nappal állítottak ki, át kell tudni írni a dátumot. A számla keletkezésének napja nem módosítható, ezt is törvény írja elő. A fizetési határidő tetszőlegesen módosítható, de a számolgatásokat elkerülve lenyitható egy lista, ahol megadható a fizetési feltétel és kiszámolja a fizetés határidejének dátumát. Készpénz esetében egyazon nap van a

határidő, mint a számla keltezése. Átutalásos számláknál megadható, hogy 8, 15, 30, 45 vagy 60 napos átutalással fizet a vásárló és ennek alapján kerül kiszámolásra a fizetési határidő, amit a program automatikusan a mezőbe ír. Az „OK” gombra kattintva elkészül a számla és a végösszeg felíródik a vásárló eddig vásárolt értékéhez.

SZÁMLA

(Eredeti 1. példány)
A számla 3 példányban készült.

Számlaszám: 2008/00009

Eladó adatai:

Luinox Kereskedelmi és Szolgáltató Kft.
4027 Debrecen Böszörményi út 22.
Telefon: 52/319-610 Fax: 52/314-241
e-mail: luinox@luinox.hu
Adószáma: 11554479-2-09
Bankszámlaszáma: 10103434-47176128-00000000
IBAN: HU40-10103434-47176128-00000000
EU adószám: EU11554479

Vevő adatai:

Koroknai Attila
4027 Debrecen
Böszörményi út 22.
Telefon: 30-321-6789
Fax:
e-mail: attila.koroknai@freemail.hu
Adószáma:
EU adószám:

Megjegyzés: A számla a 123456 szállítólevél alapján került kiállításra. **Fizetés módja:** Átutalás - 8 napos

Teljesítés napja: 2009.02.22 **Számla kelte:** 2009.02.22 **Fizetési határidő:** 2009.03.02

Cikkszám	Megnevezés			Mérete	Anyagmin.	VTSZ /SZJ	Áfa	
Mennyiség	M.e.	Egységár	Kedv.	Eladásiár	B' egységár	N' össz.é.	Áfa össz.	B' össz.é.
200-050001-001	Ív 90"			NA50 52.0x1.5		1.4301	7307000000	20
20,00	darab	1420,00	0 %	1420,00	1704,00	28400,00	5680,00	34080,00
200-050001-001	Ív 90"			NA50 52.0x1.5		1.4301	7307000000	20
10,00	darab	1420,00	10 %	1278,00	1533,60	12780,00	2556,00	15336,00
404-050000-001	Menetesvég			NA50		1.4301	7307000000	20
5,00	darab	1540,00	20 %	1232,00	1478,40	6160,00	1232,00	7392,00
P.H.						Nettó végösszeg:	47340,00	
						Áfa értéke:	9468,00	
						Bruttó végösszeg:	56808,00	
átvevő				Kerekített fizetendő bruttó végösszeg:			56808,00	
azaz ötvenhatezernyolcszáznyolc HUF								

A „Beállítások” menüpont alatt lehet megadni, hogy a készpénzes és átutalásos számlák mennyi példányban kerüljenek nyomtatásra. A számlán fel van tüntetve, hogy mennyi példány készült belőle. A fejléc tartalmazza a számla szigorú sorsszámát. Itt jelenik meg, hogy melyik példányról van szó és az „Eredeti” jelző. Fontossága, hogy egy számlát eredeti példányban csak egyszer lehet kinyomtatni. Ha újra nyomtatásra kerülne, akkor itt már csak a „Másolat” jelző szerepelhet. Fel van tüntetve az eladó és vásárló címe, adószáma (ha a vásárló cég) és fontosabb adataik.

Szerepel a számlán a keltezés, teljesítés, fizetési határidő, fizetés módja. A számla tétel tartalmazza az áru megnevezését, mennyiségét, mennyiségegységét, lista egységárat áfa nélkül, kedvezményt, kedvezményezett egységárat áfa nélkül, áfával növelt egységárat, nettó összértéket, nettó összérték áfáját és bruttó összértéket. A tételek alatt kerül összeadásra a számla nettó végösszege, az áfa mértéke és a bruttó végösszeg. A kerekített fizetendő bruttó végösszeg átutalásos számlánál a kerekítés csak egész értékre történik, készpénzes számlánál pedig 0, 5 és 10 forintra.

4.3.2 Kiállított számla megtekintése

A „Számlák” menüpontban megjelenik az eddig kiállított összes számla. A számlák dátum szerint, számlaszám szerint és vásárlók szerint kikereseshetők. A dátum szerinti keresésnél fontos a dátum formátumának helyes megadása. Az év, hónap, napot ponttal elválasztva, pl.: 2008.09.10. A számlaszám szerinti keresésnél a számla sorszámát kell a mezőben megadni. A vásárlók szerinti keresésnél a listából ki kell választani az adott vásárló nevét. A „Keresés” gomb hatására megjelennek a keresett számlák a táblázatban. A táblázatban kiválasztott számla megtekinthető vagy kinyomtatható.

4.3.3 Kiállított számla törlése

Ha egy vásárló visszahoz egy olyan terméket, ami hibás vagy cserélni akarja, akkor szükséges a számla lerontása. A táblázatban ki kell keresni azt a számlát, amelyiket le kell rontani, majd a „Számlát leront” gombra kattintva lehet a törlést aktiválni. A kiválasztott számla sorában a számla állapota „rontott” néven jelenik meg. A szigorú sorszámadást betartva a program egy új számlát hoz létre (javító számla). Az új számla a rontandó számlán szereplő összes adatot ugyanúgy tartalmazza, kivéve a végösszeget, amely negatív értékkel jelenik meg. A számla megjegyzés rovatába kerül, hogy ez melyik számla módosító számlája.

JAVÍTÓ SZÁMLA(Másolat 1. példány)
A számla 1 példányban készült.**Számlaszám: 2008/00010****Eladó adatai:**

Luinox Kereskedelmi és Szolgáltató Kft.
 4027 Debrecen Böszörményi út 22.
 Telefon: 52/319-610 Fax: 52/314-241
 e-mail: luinox@luinox.hu
 Adószáma: 11554479-2-09
 Bankszámlaszám: 10103434-47176128-00000000
 IBAN: HU40-10103434-47176128-00000000
 EU adószám: EU11554479

Vevő adatai:

Koroknai Attila
 4027 Debrecen
 Böszörményi út 22.
 Telefon: 30-321-6789
 Fax:
 e-mail: attila.koroknai@freemail.hu
 Adószáma:
 EU adószám:

Megjegyzés: A 2008/00009 javító számlája**Fizetés módja:** Átutalás - 8 napos

Teljesítés napja: 2009.02.22				Számla kelte: 2009.02.22		Fizetési határidő: 2009.02.22		
Cikkszám	Megnevezés			Mérete		Anyagmin.	VTSZ/SZJ	Áfa
Mennyiség	M.e.	Egységár	Kedv.	Eladásiár	B' egységár	N' össz.é.	Áfa össz.	B' össz.é.
200-050001-001	Ív 90"				NA50 52.0x1.5	1.4301	7307000000	20
20,00	darab	1420,00	0 %	1420,00	1704,00	28400,00	5680,00	34080,00
200-050001-001	Ív 90"				NA50 52.0x1.5	1.4301	7307000000	20
10,00	darab	1420,00	10 %	1278,00	1533,60	12780,00	2556,00	15336,00
404-050000-001	Menetesvég				NA50	1.4301	7307000000	20
5,00	darab	1540,00	20 %	1232,00	1478,40	6160,00	1232,00	7392,00
						Nettó végösszeg:	47340,00	
P.H.						Áfa értéke:	9468,00	
						Bruttó végösszeg:	56808,00	
_____ átvevő				Kerekített fizetendő bruttó végösszeg:			-56808,00	
azaz mínusz ötvenhatezernyolcszáznyolc HUF								

Csak olyan számlát lehet lerontani, melynek állapota „élő” néven szerepel a táblázatban. Ellenkező esetben hibaüzenetet küld a felhasználó felé „A számla már rontott!” üzenettel. A számla lerontását követően a számlán szereplő tételek visszakerülnek a raktárkészletre. Erről egy új bevételezett számla készül, mely a „Bevételezett számlák” menüpontban megtekinthető.

4.4 Árajánlat menüpont

Legtöbb esetben a vásárlást megelőzi egy árajánlatkérés. A vásárló több cégtől is kikéri egy termék árát. Attól veszi meg, amelyik a legkedvezőbb ajánlatot nyújtotta. A felhasználó munkáját könnyíti, hogy lehetősége van a program segítségével árajánlatot készíteni.

4.4.1 Árajánlat készítés

A vásárló kiválasztását követően (ha a táblázatban nem szerepel a vásárló neve, lehetőség van új felvételére) megjelenik egy új ablak, amely az árulistát tartalmazza. Ebből a listából kerül kiválasztásra az az árutétel/árutételek, amely vásárló által kért árajánlaton szerepelni fog. Ha olyan termékről kér ajánlatot, amely nem szerepel az árutörzsben, akkor lehetőség van ebben a menüpontban is újat létrehozni.

Ha ki lett választva a termék, akkor a „Tételt felvesz” gombra kattintva megjelenik egy új ablak. Ebben az ablakban adható meg a kiválasztott áru mennyisége és a kiajánlási ára. Az itt megadható mennyiség független az árukészleten lévőttől, ugyanis itt nem történik árumozgás. Erről csak egy árajánlat nyomtatvány készül. A „Tételt felvesz” gomb hatására felkerül az árajánlatra.

Árajánlat tétel	
Mennyiség:	10 darab
Nettó eladási ára:	1420.0
ív NA50 1.4301	<input type="button" value="Tételt felvesz"/> <input type="button" value="Kilép"/>

Az árajánlaton szereplő anyagok ellenőrizhetők a „Tételek” gombra kattintva.

Árajánlat tételek							
Cikkszám	Megnevezés	Méret	Anyagminőség	Mennyiség	M.e.	Áfa	Kiajánlott ár
200-050000-001	ív	NA50	1.4301	10.0	darab	20	1420.0
300-050000-005	Pillangószelep H/H	NA50	1.4404	2.0	darab	20	12310.0
008-310102-001	Lemez 3C (matt)	1.0x1000x2000	1.4301	1.0	tábla	20	12300.0

Kiválasztott tétel: ív NA50 1.4301

Nettó összesen:	51120.0	Bruttó összesen:	61344.0	<input type="button" value="Tétel törlése"/> <input type="button" value="Ok"/>
-----------------	---------	------------------	---------	--

A táblázat alatt jelenik meg az „Árajánlat sorszáma”, amelyre a vásárló hivatkozni tud a későbbiekben. A program a sorszámot automatikusan rendeli hozzá

az ajánlathoz. Megadható az árajánlatra vonatkozó teljesítés határideje, pl.: raktárról; 1-2 nap; stb., továbbá a fizetésének módja, pl.: készpénz, átutalás. A „Megjegyzés” rovat segítségével további információt közölhetünk a vásárló felé. Végül az „Árajánlat” gombra kattintva elkészül az árajánlat egy példányban.

Árajánlat

Sorszám: 14

Eladó adatai:

LUINOX Kereskedelmi és Szolgáltató Kft.
 4027 Debrecen
 Böszörményi út 22.
 Telefon: +(36) 52-319-610
 Fax: +(36) 52-314-241
 e-mail: luinox@luinox.hu
 Adószáma: 11554479-2-09
 Bankszámlaszám: Budapest Bank Rt: 10103434-47176128-00000000-00000000
 IBAN: IBAN
 EU Tax: EU

Vásárló adatai:

Lukács Kft.
 4060 Balmazújváros
 Arany János u.23.
 Telefon: 52-370-413
 Fax: 52-370-413
 e-mail: postmaster@lukacsft.axelero.net
 Adószáma: 12502743-2-09
 EU Tax: EU 12502743

Kiállítás dátuma: 2008. november

Fizetés módja: Azonnali átutalás

Cikkszám	Megnevezés	Méret	Anyagminőség	Áfa	
	Mennyiség	M.egység	Egység ár	N'összérték	B'összérték
200-050000-001	Ív		NA50	1.4301	20 %
	10,00	darab	1420,00	14200,00	17040,00
300-050000-005	Pillangószelep H/H		NA50	1.4404	20 %
	2,00	darab	12310,00	24620,00	29544,00
008-310102-001	Lemez 3C (matt)		1.0x1000x2000	1.4301	20 %
	1,00	tábla	12300,00	12300,00	14760,00

Szállítási határidő: Raktárról

Nettó végösszeg: 51120,00.- Forint
Bruttó végösszeg: 61344,00.- Forint

Megjegyzés: A árajánlat érvényessége 10 nap.

Árajánlatot készítette: _____

Koroknai Attila (30-321-6789) 2008'

4.4.2 Árajánlat megtekintése

Az elkészült árajánlatokat ebben a menüpontban lehet megtekinteni. Ha a vásárló a későbbiekben az árajánlatra hivatkozva megrendeli a neki kiajánlott árut, ebben a menüpontban egyszerűen visszakereshető a nyomtatvány.

4.5 Lekérdezések menüpont

Egy cég hatékony működéséhez elengedhetetlen, hogy az általa használt nyilvántartó program információkkal lássa el az áruk mozgásáról, az eladásból származó nyereségről, raktárkészletről stb. Ebben a menüpontban került beépítésre a leltárkészítés, minimum készlet figyelése is. A lekérdezések elsőnek egy „Viewer”-ben jelennek meg, ha igény merül fel, akkor a nyomtatásuk lehetséges.

Leltárkészítés: a riporton megtekinthető a raktárkészleten lévő összes termék és a raktárkészlet nettó összértéke.

Elábé (Eladott Áruk Beszerzési Értéke) készítés: a lekérdezés lényege, hogy figyeli az eladott áruknak a beszerzési és eladási értékét, majd ezekből az adatokból kiszámolja az árrést. A lekérdezést egy meghatározott időintervallumban kell megadni, amelyen belül eső számlák tételei kerülnek listázásra.

LUINOX Kereskedelmi és
4027 Debrecen
Böszörményi út 22.

ELÁBÉ (Eladott Áruk Beszerzési Értéke)

Cikkszám	Menny.	Besz. ár	Elad. ár	Össz.beár	Össz.eladár	Különbség	Haszon	Számlaszám
200-040000-001	10,00	688,00	1116,00	6880,00	11160,00	4280,00	38,35	2008/00011
200-040000-001	10,00	688,00	1240,00	6880,00	12400,00	5520,00	44,52	2008/00011
201-020000-001	1,00	1221,00	1692,00	1221,00	1692,00	471,00	27,84	2008/00011
001-040015-001	2,00	1102,40	1670,00	2204,80	3340,00	1135,20	33,99	2008/00012
001-052015-001	3,00	1480,00	2480,00	4440,00	7440,00	3000,00	40,32	2008/00012
200-040000-001	2,00	688,00	1240,00	1376,00	2480,00	1104,00	44,52	2008/00012
200-050000-001	1,00	988,00	1420,00	988,00	1420,00	432,00	30,42	2008/00012
404-050000-001	5,00	1266,20	1386,00	6331,00	6930,00	599,00	8,64	2008/00013
008-305102-001	1,00	1000,00	7020,00	1000,00	7020,00	6020,00	85,75	2008/00013
Beszerzési érték	31 321	Eladási érték:		53 882	Árrés:	22 561	42 %	

A riporton megjelenik a termék cikkszama, eladott mennyisége, ennek egységre eső beszerzési ára, az egységre eső eladási ára, valamint az össz beszerzési (mennyiség x beszerzési ár) és össz eladási (mennyiség x eladási ár) ár. A két utóbbinak veszi a különbségét, ez lesz az árrés. A hasznot százalékban is meghatározza soronként. A lekérdezés végén összesítve is megjelennek az adatok. Jelen esetben a három számlán szereplő termékek eladásából származó nyereség a cég számára 42%.

Beszerzés tételenként: a lekérdezés lehetőséget ad arra, hogy egy termék beszerzései nyomon követhetők legyenek. A táblázatban kiválasztott árut a riportban, bevételezési sorrendben rendezi. Minden sorban megadja a beszerzési dátumot, a mennyiséget és beszerzési árat. A lekérdezés lényege, hogy az adott termék összes eddigi bevételezését megjelenítse. A riport második felében megjelenik az adott áru raktári készlete. A következő lista tartalmazza azt, hogy a raktári készlet melyik beszerzésekből épül fel. Itt szintén megjelenik a beszerzés dátuma, mennyisége és a beszerzési ár.

Aktuális termék beszerzése tételesen lebontva

Kiválasztott anyag tétel			
404-050000-001	Menetesvég	NA50	1.4301

Beszerzés dátuma	Mennyiség	Egység	Beszerzési ár	Nettó összesen
szeptember 10, 2008	10,00	darab	1222,00	12220,00
november 16, 2008	5,00	darab	1266,20	6331,00
május 20, 2008	10,00	darab	1266,20	12662,00

Ebből raktári tétel: 15,00

Beszerzés dátuma	Mennyiség	Egység	Beszerzési ár	Nettó összesen
szeptember 10, 2008	10,00	darab	1222,00	12220.0
november 16, 2008	5,00	darab	1266,20	6331.0

Eladás tételenként: ebben a lekérdezésben a kiválasztott termék eladásai követhetők nyomon.

4.6 Telefonkönyv menüpont

Funkciója egyszerű, telefonszámok tárolására használható. Például nyilvántartható a dolgozók, több árufuvarozó cég, reklámügynökségek, könyvelő cég telefonszámai, stb. A telefonszámok bővíthetők, módosíthatók és törölhetők.

4.7 Beállítások

Jelen verzióban lévő beállítási lehetőségek az eladó adatainak beállítása és a nyomtatással kapcsolatos beállítások. A nyomtatásoknál megadható a készpénzes és átutalásos számlák nyomtatásának példányszáma.

5. ZÁRÓ GONDOLATOK

A dolgozat megírásánál az a cél vezérelt, hogy bemutassam a programomat az adatbázis-kezelői oldalról megközelítve. Elég nagy témakört foglalt volna be, ha a java forráskód bemutatásáról is írtam volna. A dolgozat keretei határt szabnak, hiszen olyan nagy anyagról lenne szó, melynek könyvek sokasága sem lenne elég. Írásos dokumentációt (könyvet) az adatbázis-kezelőről és az iReport programról nem nagyon lehet találni. A programok hivatalos weboldalain lehet információkhoz jutni, mind a telepítésről és használatukról. A programom a végleges verzióját még nem vette fel, ugyanis vannak új ötletek, melyeket a szakdolgozat megírása után fogok megvalósítani. Ilyen például a készletről való leemelés úgy, hogy csak egy belső nyomtatvány készül a tranzakcióról, továbbá szállítólevél kiállítása elektronikusan, illetve megrendelés visszaigazolás készítés.

A program forráskódja (java kód), a PostgreSQL 8.2 és az iReport 3.0.0 programok, mellékelve vannak a szakdolgozathoz készült cd-n.

6. KÖSZÖNETNYILVÁNÍTÁS

Ezúton szeretnék köszönetet mondani témavezetőmnek, Dr. Boda István tanár úrnak, hogy lehetőséget biztosított munkám sikeres elvégzéséhez, valamint dolgozatom megírásához nyújtott útmutató tanácsaiért.

7. IRODALOMJEGYZÉK

1. Budapest Műszaki és Gazdaságtudományi Egyetem

<http://octopus.hit.bme.hu>

2. PostgreSQL

<http://www.postgresql.org>

3. JasperSoft

<http://jasperforge.org>

4. A design tool iReport for JasperReports

<http://ireport.sourceforge.net>

5. JDBC