

# **Záródolgozat**

**Juhász Attila**



**OKJ 54 213 05**

**— Szoftverfejlesztő —**

**2/14/F**

**Kecskeméti Szakképzési Centrum**

**Kandó Kálmán Technikum**

**Kecskemét**

**2021**

**FIGYELEM!**

A program a fejlesztő tulajdona, amit a nemzetközi szerzői jogok védenek.  
A program használati engedélyét kizárólagosan a verseny idejére biztosítom.  
Továbbadása, forráskódjának visszafejtése a szerző engedélye nélkül tilos.

# **Számla iktatási rendszer**

# Tartalomjegyzék

<b>Előszó .....</b>	<b>4</b>
<b>Fejlesztői környezet .....</b>	<b>5</b>
<b>Rendszerkövetelmények .....</b>	<b>5</b>
<b>Alkalmazott fejlesztői eszközök, programnyelvek .....</b>	<b>5</b>
<b>Használt technológiák .....</b>	<b>5</b>
<b>Megvalósítás .....</b>	<b>9</b>
<b>Az adatmodell leírása .....</b>	<b>9</b>
<b>A program felépítése .....</b>	<b>11</b>
<b>Fontosabb algoritmusok .....</b>	<b>14</b>
<b>Felhasználói dokumentáció .....</b>	<b>16</b>
<b>Tesztelési dokumentáció.....</b>	<b>21</b>
<b>Továbbfejlesztési lehetőségek .....</b>	<b>22</b>
<b>Források .....</b>	<b>23</b>
<b>A program elérhetősége .....</b>	<b>23</b>

# Előszó

A program egy, a munkahelyemen a valóságban létező problémát igyekszik orvosolni. Jelenleg a számlák igazolása teljesen manuálisan történik, ami azt jelenti, hogy a papír alapú bizonylatokat – amik vagy postán érkeztek, vagy pedig elektronikus számlák kinyomtatott példányai - utazzatunk a város különböző részén lévő szervezeti egységek között, a pénzügyi munkacsoporttól a számla igazolására illetékes vezető beosztású személyig. Mindez felesleges időbe, energiába és pénzbe kerül. A programom segítségével ez a folyamat kiváltható és teljesen elektronikus úton végezhető.

A program megvalósítja, hogy a gazdasági társaság nevére kiállított postán beérkező papír alapú, vagy pedig a valamilyen internetes csatornán keresztül érkező elektronikus számlákat iktatja, majd pedig az iktatott számlák teljesülését, és ezzel a kifizethetőségét egy arra jogosult személy leigazolhatja.

Első lépésként tárolni kell a számlák „képét”: a már eleve .pdf formátumban érkező elektronikus számlákat, és a papír alapú számlák szkennelésével nyert .pdf file-okat menteni kell egy nevesített mappába.

A következő lépés a számlák iktatása: a számlák adatai - számlaszám, dátumok, nettó érték, áfa érték, bruttó érték stb. - kézi adatrögzítés során bekerülnek egy adatbázisba, és szintén idekerül még a korábban letárolt .pdf file elérési útja is.

Ezután következhet a számlák igazolása: egy megfelelő jogosultsággal rendelkező személy, aki bír azzal az információval, hogy azok a gazdasági események, amelyekről a számlák készültek, valóban megvalósultak-e, lekérdezi az adatbázisból az iktatott, és még igazolatlan számlákat, majd pedig leigazolja, vagy pedig visszautasítja őket.

Végül az illetékes felhasználók lekérdezhetik a számlákat státuszuk szerint, és elvégezhetik velük a szükséges folyamatot; vagyis gondoskodnak az igazolt számlák kifizetéséről, a visszautasított számlákat pedig visszaküldik a számlát kiállító partnernek.

# Fejlesztői környezet

## Rendszerkövetelmények

A program fejlesztése Intel(R) Core(TM) i7-8550U típusú, 1.80GHz-es x64-alapú processzorral és 8,00 GB RAM-mal rendelkező számítógépen történt, 64 bites Windows 10 Pro operációs rendszer alatt; másfajta konfigurációval nincs tapasztalat.

## Alkalmazott fejlesztői eszközök, programnyelvek

- Xampp webserverver
- Visual Studio Code
- Github
- PHP
- HTML
- CSS
- Bootstrap keretrendszer
- JavaScript

## Használt technológiák

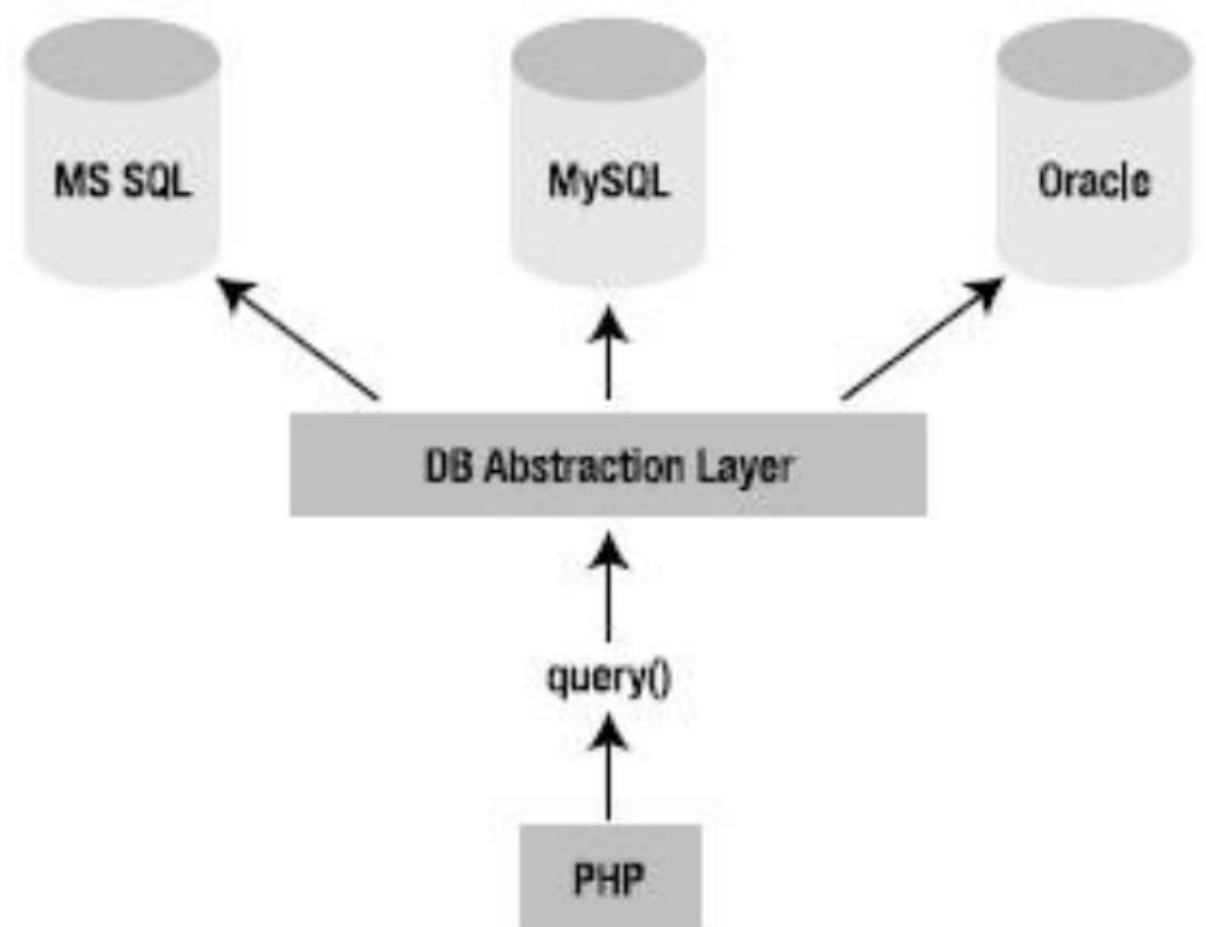
1. Adatbázis absztrakciós réteggel csatlakozunk az adatbázishoz, ami objektum orientált adatbázis kapcsolódás, így bármilyen típusú is az adatbázis, ugyanazzal a kóddal elérhető:

```

config.php
1  <?php
2  $host = 'localhost';
3  $db_name = 'iktatasi_rendszer';
4  $db_username = 'root';
5  $db_password = '';
6  $options = [PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION, PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
7  PDO::ATTR_EMULATE_PREPARES => 0];
8  try{
9      $db = new PDO("mysql:host=$host;dbname=$db_name; charset=utf8",$db_username,$db_password,$options);
10 }catch(PDOException $e) {
11     echo "!!! Az adatbázis kapcsolódás sikertelen !!!";
12     exit;
13 }
14 ?>

```

Az adatbázis absztrakciós réteg segítségével a kommunikáció központilag menedzselve valósulhat meg. Erre is létezik többféle megoldás, de talán a legelterjedtebb a PDO (PHP Data Objects).



# PDO FŐBB UTASÍTÁSAI

- `$pdo = new PDO("adatbázis_kapcsolat");`: kapcsolódás az adatbázishoz
- `$pdo = null`: kapcsolat bontása
- `$stmt = $pdo->query("sql");`: lekérdezések (`SELECT`) futtatása
- `$db = $pdo->exec("sql");`: nem lekérdezések esetén

## 2. Munkamenet kezelés:

PHP-ban számos függvény és nyelvi elem segíti a munkamenet-kezelést. A munkamenet adatait a PHP a `$_SESSION` szuperglobális asszociatív tömbben tárolja. A szkript kezdetekor ebbe tölti be, a szkript végén pedig ennek tartalmát menti ki (alapértelmezetten fájlba).

Hasznos függvények:

- `session_start()`: a munkamenet indítását jelezzük. Minden olyan szkript elején el kell helyeznünk, ahol a munkamenet adataira van szükségünk. A parancs kiadása után a PHP ellenőrzi a sütitben vagy URL-ben érkező munkamenet-azonosítót, és ha van ilyen, akkor a hozzá tartozó adatokat betölti a `$_SESSION` tömbbe. Ha nincs azonosító, akkor újat hoz létre.

- `session_destroy()`: a munkamenet megszüntetésére szolgál.

Következzen egy példa a `$_SESSION` tömb használatára. Az alábbi részlet a felhasználó bejelentkezését vezérlő `login.php` programból való. Sikeres autentikáció esetén eltároljuk (1. kép) a `$_SESSION` tömbben a felhasználó nevét (`$_SESSION['user']`) és jogosultságát (`$_SESSION['jog']`), mely meghatározza, hogy a felhasználó a program mely menüpontjait használhatja (2. kép).

```

if(isset($_POST['belepes'])) {
    extract($_POST);
    $sql="select felhasznalok.nev as 'felhasznalo',jelszo,jogosultsagok.nev as 'jogosultsag' FROM felhasznalok,jogosultsagok
    WHERE felhasznalok.jogosultsagok_az=jogosultsagok.az and felhasznalok.nev='{ $azonosito}' and jelszo='{ $password}'";
    $stmt=$db->query($sql);
    if($stmt->rowCount()==1){
        $row=$stmt->fetch();
        extract($row);
        $_SESSION['user']=$felhasznalo;
        $_SESSION['jog']=$jogosultsag;
        unset($_SESSION['msg']);
    }else{
        $_SESSION['msg']="Hibás azonosító/jelszó páros!";
        unset($_SESSION['user']);
    }
    header('Location:index.php');
}

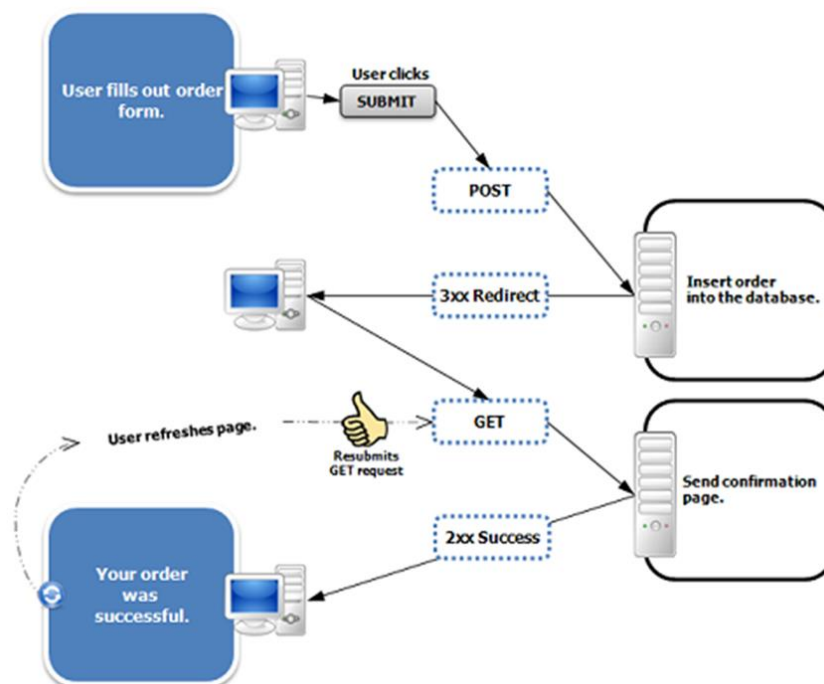
```

```

if(isset($_GET["p"]) && isset($_SESSION["jog"])){
    $oldal = $_GET["p"];
    switch($oldal){
        case "iktatas.php" :
            if ($_SESSION["jog"]=="iktatas" || $_SESSION["jog"]=="admin")
                include $oldal;
            else include("fooldal.php");
            break;
        case "felhasznalok.php" :
            if ($_SESSION["jog"]=="admin")
                include $oldal;
            else include("fooldal.php");
            break;
    }
}

```

3. PRG programozási technika (post-redirect-get): ezzel akadályozzuk meg az ugyanazon űrlap adatok ismételt elküldését.



4.

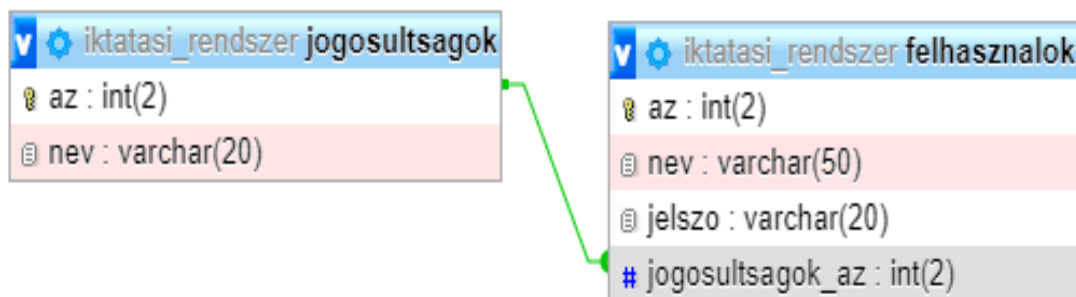


# Megvalósítás

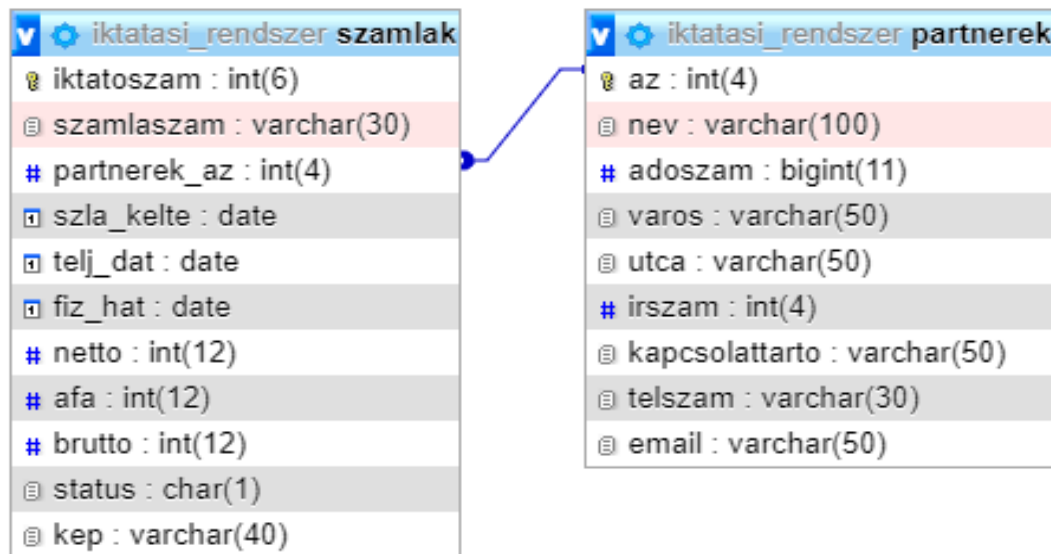
## Az adatmodell leírása

A felhasználók tárolásához az alábbi 2 adattáblára van szükség. A „felhasznalok” tábla tartalmaz egy „az” nevű azonosítószámot, ami egy futó sorszám, az egyedi azonosító, a program használatának szempontjából nincs jelentősége; egy „nev” nevű mezőt, ami a felhasználó nevét tárolja; egy „jelszo” mezőt, amibe a jelszó kerül, és egy „jogosultsagok\_az” nevű mezőt, ami a felhasználóhoz tartozó jogosultság azonosítóját tárolja; megteremti a „felhasznalok” és a „jogosultsagok” táblák közötti kapcsolatot, és vezérli, hogy az egyes menüpontokhoz melyik felhasználóknak van hozzáférése.

A „jogosultsagok” tábla 2 mezőt tartalmaz: az „az” mező a jogosultság sorszáma, az egyedi azonosító; a „nev” mező pedig a jogosultság megnevezését tartalmazza.



A „partnerek” tábla tartalmazza a beszállító cégek adatait. Az „az” mező egy futó sorszám, egyben az egyedi azonosító, és a „számlák” táblával a kapcsolatot megteremtő mező is. A „nev” mező a partner nevét, az „adoszam” az adószámát, a „varos”, „utca” és „irszam” – irányítószám - mezőkben a partner címét tároljuk; a „kapcsolattarto” mezőben annak a nevét, aki a partner céget képviseli; a „telszam” mezőben a telefonszámát, az „email” mezőben pedig az e-mail címét.



A „szamlak” tábla tartalmazza az iktatott számlák adatait. Az „iktatoszam” a tábla egyedi azonosítója. A „partnerek\_az” a számla kiállítójának azonosítója az adatbázisban; emellett ez a mező teremti meg a kapcsolatot a „partnerek” táblával. A következő mezők a számláról kapnak értéket: „szla\_kelte” a számla kiállításának dátuma; a „telj\_dat” a teljesítés dátuma; a „fiz\_hat” a fizetési határidő; a „netto”, „afa” és „bruttó” mezők pedig a számlán található érték adatokat tárolja. A „status” mező a számla „állapotát” tartalmazza, ami lehet: „N” iktatott, „I” igazolt és „V” visszautasított. A „kep” mező a számlához tartozó pdf file nevét tartalmazza.

## A program felépítése

Az oldal betöltődésekor az index.php nevű program indul el, ami önmagában vezérli az egész program működését. Itt történnek meg az alapvető beállítások, ez felelős a menüpontok megjelenítéséért, tartalmazza azt a jogosultság rendszert, amely engedi, vagy tiltja egy-egy menüpont használatát, és vezérli az egyes menüpontok mögött futó php programok betöltését. Az index.php betöltés után a login.php-ra irányít át, ami a felhasználó bejelentkezését végzi. A felhasználónévhez tartozó jogosultság egészen a kijelentkezésig meghatározza, hogy a felhasználó mely funkciók használatára jogosult. Ennek megtörténte után a felhasználó már „válogathat” a menüpontok közül.

A menüpontok kiválasztásakor történeteket a következő programrészek irányítják (tételezzük fel, hogy a felhasználó a „Szamlák iktatása” menüpontot választotta):

Ez maga a menüpont, ami valójában egy hivatkozás, ami egy lista elemeként jelenik meg a képernyőn. Kattintáskor az index.php hívódik meg úgy, hogy a „p” változó az iktatas.php értéket kapja meg.

```
<li class="nav-item">
+ <a class="nav-link text-white" href="index.php?p=iktatas.php" >Szamlák iktatása</a>
</li>
```

A \$\_GET tömb „p” eleme (\$\_GET["p"]) - aminek értéke átadódik a \$oldal változónak -, tartalmazza annak a programnak a nevét, amit a felhasználó a kiválasztott menüponttal aktiválni szeretne (esetünkben ez az iktatas.php). A \$\_SESSION tömb „jog” nevű eleme (\$\_SESSION["jog"]) tárolja a felhasználó jogosultságát. Egy feltételes elágazás vizsgálja, hogy az egyes, a \$oldal változó által meghatározott php programok futtatthatók-e (példánkban az iktatas.php akkor futtatható, ha a bejelentkezett felhasználó jogosultsága „iktatas”, vagy pedig „admin”). Ha a felhasználónak nincs megfelelő jogosultsága, akkor a fooldal.php töltődik be.

```

if(isset($_GET["p"]) && isset($_SESSION["jog"])){
    $oldal = $_GET["p"];
    switch($oldal){
        case "iktatas.php" :
            if ($_SESSION["jog"]=="iktatas" || $_SESSION["jog"]=="admin")
                include $oldal;
            else include("fooldal.php");
            break;
    }
}

```

A „rendszer” több nagyobb egységre osztható fel a hozzá tartozó php programok alapján. A login.php-ről már fentebb esett szó. A felhasználok.php – mely a felhasználók kezelését teszi lehetővé -, a partnerek.php – ahol a beszállító cégek adatait tarthatjuk nyilván - és az iktatas.php – minek segítségével rögzíthetjük a számlákat az adatbázisba – hasonló elven működnek, mindhárom a CRUD programozási technikát alkalmazza (az iktatas.php-t láthatjuk).

```

<?php
ob_start();

require_once 'config.php';
if(isset($_GET['editId']))
    include "iktatas\\edit.php";
else if(isset($_GET['deleteId']))
    include "iktatas\\delete.php";
else if(isset($_GET['insert']))
    include "iktatas\\insert.php";
else
    include "iktatas\\iktatasView.php";
?>

```

Mindhárom programhoz tartozik saját edit, delete, insert és ...View „alprogram”, melyek természetesen témánként külön mappákban tárolódnak. Az egyes „alprogramok” attól függően hívódnak meg, hogy a \$\_GET tömb milyen értéket tartalmaz. Ha a felhasználó egy Módosít „gombra” kattint, akkor az editId nevű változó értéket kap – ez az érték annak a rekordnak az egyedi azonosítója, amit az adatbázisban módosítani kell -, és ezzel együtt elindul a megfelelő edit.php program. Amikor egy Törlés „gombon” történik kattintás, akkor a deleteId megkapja a törlendő rekord egyedi azonosítóját, és ezzel a paraméterrel hívódik meg a megfelelő delete.php.

```
$tbl.="<td class=' btn btn-outline-primary m-1'><a class='text-warning'
| href='index.php?p=iktatas.php&editId=$iktatoszam'>Módosítás</a></td>";
$tbl.="<td class=' btn btn-outline-primary m-1'><a class='text-danger'
| href='index.php?p=iktatas.php&deleteId=$iktatoszam'>Törlés</a></td></tr>";
```

Új elem felvitele „gombon” történt klikkelés esetén az insert.php indul el, paraméter nélkül.

```
<div class="btn btn-sm btn-outline-light m-1 p-1 rounded"><a href="index.php?p=iktatas.php&insert=insert.php">
+ <b>Új számla rögzítése</b></a></div>
```

Egyéb esetben a ...View.php hívódik meg, ami kiírja a képernyőre táblázatos formában a megfelelő adattábla tartalmát.

Az igazolas.php – mely az iktatott számlák igazolását valósítja meg -, ettől eltérően működik. A menü futtatásakor képernyőre kerül az összes olyan számla, aminek a státusza iktatott. Ez azt jelenti, hogy az adatbázisban a „status” mező értéke 'N'. A felhasználó két lehetőség közül választhat, és ehhez két „gomb” áll rendelkezésére minden számlánál, ami két külön php programot hív meg: az egyik az „Igazolás” – igazol.php -, mely hatására a „status” értéke 'I' lesz, a másik pedig a „Visszautasítás” – visszautasit.php -, minek hatására a „status” mező értéke 'V'-re változik.

Azokat a számlákat, amik az előbbi igazolási folyamaton átesetek, két külön menüpontban láthatjuk, ezek az: „Igazolt számlák” – igazolt.php – és a „Visszautasított számlák” – visszautasitott.php.

## Fontosabb algoritmusok

A következő algoritmus, programrészlet több helyen is előfordul a programban. Feladata egy, az adatbázisból lekérdezett eredményhalmaz táblázatos megjelenítése. Az alábbi részlet a partnerekView.php programból származik.

A program elején deklarált \$tbl változó tartalmazza majd magát a táblázatot, annak soraival, oszlopaival, celláival és adataival. Az \$sql nevű változóban tároljuk le azt az sql parancssort, amivel lekérdezhethetjük a kívánt eredményhalmazt az adatbázisból – ezesetben a partnerek tábla minden rekordját. A lekérdezett eredményhalmaz a \$stmt változóban tárolódik, ennek tartalmát olvassuk végig egy while ciklus segítségével. A ciklus minden egyes futásakor a fetch() függvény segítségével bekerül a \$row változóba az \$stmt-ben tárolt adathalmaz éppen következő sora. Ennek a sornak a tartalmát „bontjuk ki” az extract utasítással azért, hogy az egyes adatokra az adatbázisbeli nevükkel tudjunk direkt módon hivatkozni. Következik a táblázat megalkotása: a \$tbl változóhoz hozzáadjuk a táblázat következő sorát, vagyis a <tr> és <td> html tagekkel létrehozuk a következő sort és annak celláit, a cellákba pedig betöltjük az adatbázis megfelelő mezőinek értékét a nevükre, mint változókra hivatkozva. Végül minden táblázatsor végére kerül egy Módosítás és egy Törlés nevű, bootstrap-pel formázott gomb.

```
<?php
$tbl="";
$sql="SELECT az,nev,adoszam,varos,utca,irszam,kapcsolattarto,telszam,email FROM partnerek";
$stmt=$db->query($sql);
while($row=$stmt->fetch())
{
    extract($row);
    $tbl.="<tr><td>{$az}</td><td>{$nev}</td><td>{$adoszam}</td><td>{$varos}</td><td>{$utca}</td>
    <td>{$irszam}</td> <td>{$kapcsolattarto}</td><td>{$telszam}</td><td>{$email}</td>";
    $tbl.="<td class=' btn btn-outline-primary m-1'><a class='text-warning'
    href='index.php?p=partnerek.php&editId=$az'>Módosítás</a></td>";
    $tbl.="<td class=' btn btn-outline-primary m-1'><a class='text-danger'
    href='index.php?p=partnerek.php&deleteId=$az&deleteNev=$nev'>Törlés</a></td></tr>";
}
?>
```

A következőkben láthatjuk azt a html kódot, ami „megrajzolja” az előbbi táblázat fejlécét, majd pedig megjeleníti a táblázat sorait, vagyis a táblázat <tbody> tagébe betölti annak a \$tbl változónak a tartalmát, amiben fentebb már futásidőben létrehoztuk és letároltuk az egyes sorokat.

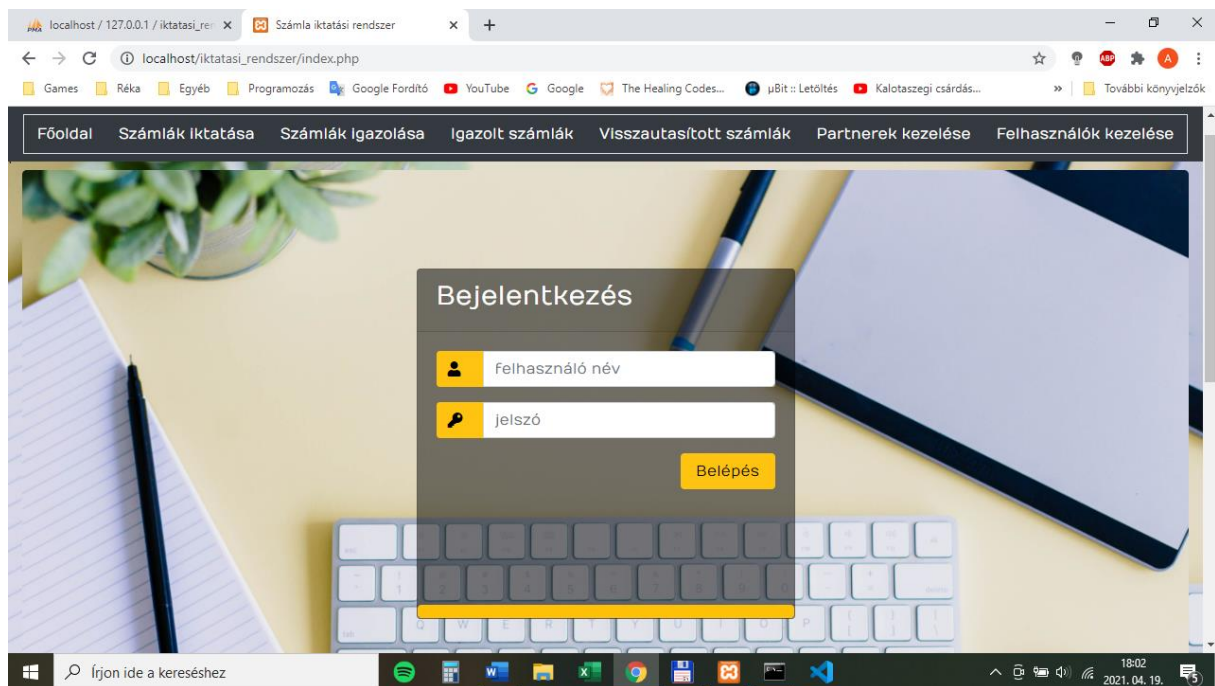
```

<table class="table table-striped table-fixed table-bordered table-sm" >
  <thead class="thead-dark">
    <tr>
      <th scope="col">Pkód</th>
      <th scope="col">Név</th>
      <th scope="col">Adószám</th>
      <th scope="col">Város</th>
      <th scope="col">Utca</th>
      <th scope="col">Irsz</th>
      <th scope="col">Kapcsolattartó</th>
      <th scope="col">Telefonszám</th>
      <th scope="col">E-mail</th>
      <th scope="col">&nbsp;</th>
    </tr>
  </thead>
  <tbody ><?=$tbl?></tbody>
</table>

```

## Felhasználói dokumentáció

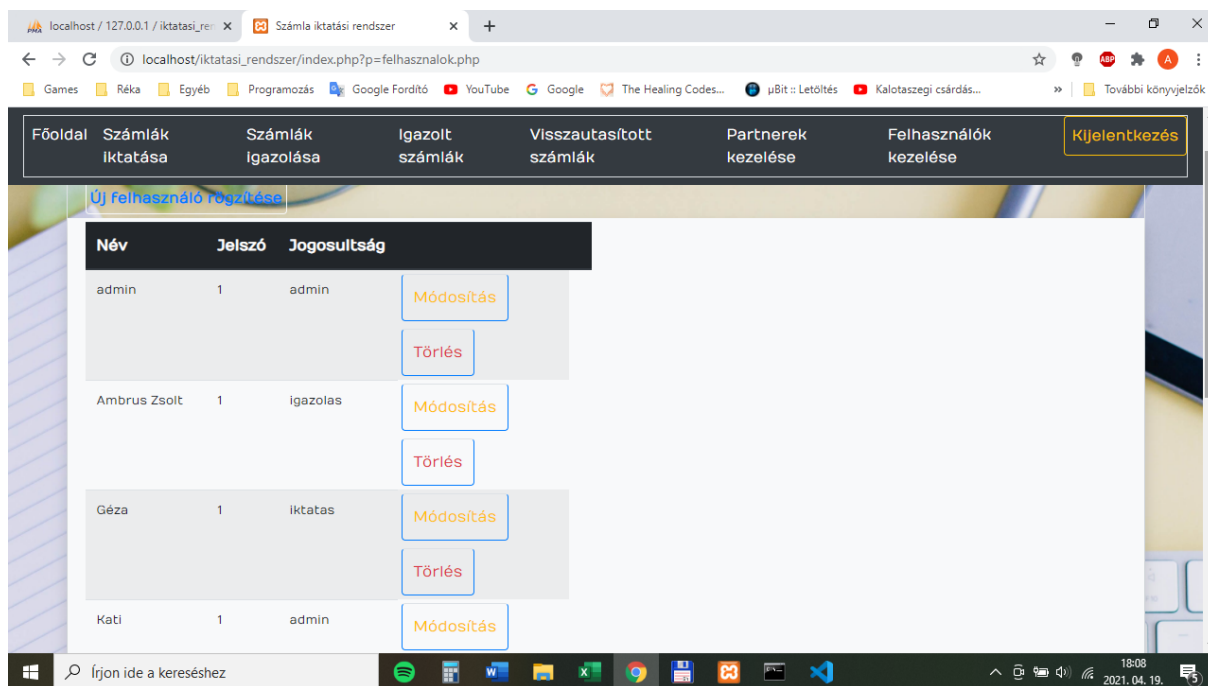
A program indítása után a bejelentkezési képernyő fogadja a felhasználót, itt kell megadnia a felhasználónevét és a jelszavát. Minden felhasználónévhez hozzá van rendelve egy jogosultság, ami meghatározza, hogy a bejelentkezés után melyik menüpontok használhatók.



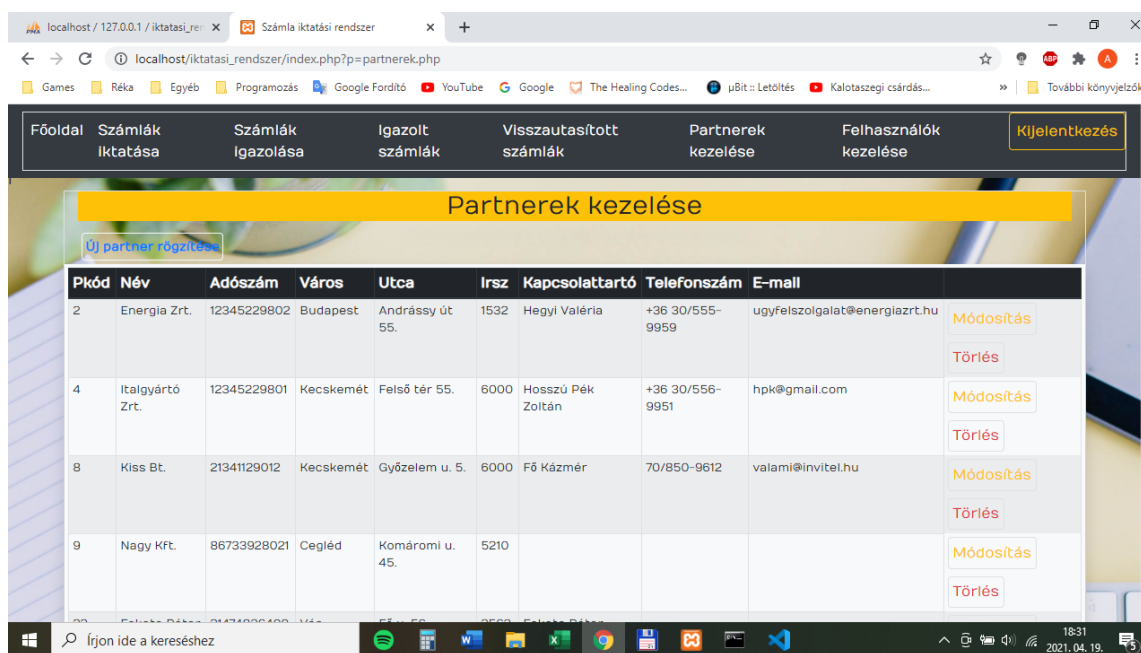
Bejelentkezés után jelenik meg a Kijelentkezés gomb, ezzel lehet a programból kilépni. Hatására a felhasználó visszakerül a bejelentkezési ablakra.

A Felhasználók kezelése menüpont csakis admin jogosultsággal használható. Itt láthatjuk a rendszerben rögzített felhasználóneveket a hozzájuk tartozó jelszóval és jogosultsággal. A Módosít gomb segítségével lehet egy felhasználó adatait megváltoztatni, a Törlés gombbal törölni, az Új felhasználó rögzítése gombbal pedig egy új felhasználó adatait berögzíteni.





A Partnerek kezelése menüpontban lehetséges a szállító partnerek adatait feltölteni. Belépéskor láthatjuk a létező partnerek adatait. Lehetőség van a Módosításra, Törlésre – kivéve, ha a partner azonosítójával már rögzítettünk számlát, mert ebben az esetben a partner nem törölhető -, és tudunk új partnert is rögzíteni. Rögzítéskor a partner azonosítóját a program automatikusan osztja ki. Rögzítés vagy Módosítás esetén a program ellenőrzi, hogy a megadott adószám létezik-e már az adatbázisban; ha igen, akkor figyelmeztet, és nem engedi a felvitelt.



A Számlák iktatása menüpontba való belépéskor az összes, még nem igazolt és nem visszautasított számlát láthatjuk, ezeket van lehetőség módosítani, vagy törölni a megfelelő gombokkal - az igazolt, vagy visszautasított számlák nem törölhetők és tartalmuk sem megváltoztatható; ezeket két külön menüpont alatt tudjuk megnézni.

Az iktatószám a számlák egyedi azonosítója, ezt a program osztja automatikusan, értéke nem megváltoztatható. Új számla felvitelekor, vagy pedig meglévő módosításakor a partner nevét egy legördülő listából választhatjuk ki. A PDF mezőbe egy pdf file neve kerül be, amit a felhasználónak kell tállózással kiválasztania. A listában lévő file nevére kattintva ez a pdf file egy új böngésző ablakban nyílik meg. Az Igazolt mező értéke iktatáskor automatikusan 'N' lesz, ez jelenti azt, hogy a számla nincs igazolva. Az összes többi mezőt a számláról kell feltölteni.

Ikt.	Számlaszám	Név	Szla. kelte	Telj. dát.	Fiz. hat.	Nettó	Áfa	Bruttó	Igazolt	PDF
2	SZ2021-00235	Energia Zrt.	2021-04-01	2021-04-01	2021-04-22	100	27	127	N	Imagináció.pdf
18	k-23/2021	Fekete Péter	2021-12-31	2021-12-31	2021-12-31	1	2	3	N	IPARI ELEKTRONIKA.pdf
19	k1-34	Italgártó Zrt.	2021-12-31	2021-12-31	2021-12-31	5	6	7	N	IPARI ELEKTRONIKA.pdf
24	2021-45/6	Kiss Bt.	2021-03-30	2021-04-14	2021-05-01	1	2	6	N	IPARI ELEKTRONIKA.pdf
25	r-45	Italgártó Zrt.	2021-12-31	2021-12-31	2021-12-31	2	3	8	N	IPARI ELEKTRONIKA.pdf
26	www	Energia Zrt.	2021-12-31	2021-12-31	2021-12-31	1	7	7	N	
27	qeqr	Fekete Péter	2021-12-31	2021-12-31	2021-12-31	-3	0	0	N	
28	eew	Energia Zrt.	2021-12-31	2021-12-31	2021-12-31	-1	0	0	N	

A számlák igazolása menüpont alatt az összes olyan számlát láthatjuk, amelyeknek a státusza 'N', vagyis nem igazolt. A menüpont használatához külön jogosultság szükséges. A számla igazolója a PDF mező tartalmára kattintva előhívhatja és megnézheti a számla képét, és eldöntheti, hogy a számla tartalma megvalósult-e; ha igen, akkor az Igazolás gombbal leigazolja – ezzel a számla státusza 'I'-re változik -, vagy pedig valamilyen kifogása van a számla ellen, akkor a Visszautasítás gombbal elutasítja a bizonylatot, aminek a státusza ezzel 'V'-re változik.

Számla iktatási rendszer

localhost/iktatasi\_rendszer/index.php?p=igazolas.php

Games Réka Egyéb Programozás Google Fordító YouTube Google The Healing Codes... μBit :: Letöltés Kalotaszegi csárdás... További könyvtárak

Főoldal Számlák iktatása Számlák igazolása Igazolt számlák Visszaautasított számlák Partnerek kezelése Felhasználók kezelése **Kijelentkezés**

### Számlák igazolása

Ikt.	Számlaszám	Név	Szla. kelte	Telj. dat.	Fiz. hat.	Nettó	Áfa	Bruttó	Igazolt	PDF	
2	SZ2021-00235	Energia Zrt.	2021-04-01	2021-04-01	2021-04-22	100	27	127	N	Imagináció.pdf	Igazolás Visszaautasítás
18	k-23/2021	Fekete Péter	2021-12-31	2021-12-31	2021-12-31	1	2	3	N	IPARI ELEKTRONIKA.pdf	Igazolás Visszaautasítás
19	k1-34	Italgártó Zrt.	2021-12-31	2021-12-31	2021-12-31	5	6	7	N	IPARI ELEKTRONIKA.pdf	Igazolás Visszaautasítás
24	2021-45/6	Kiss Bt.	2021-03-30	2021-04-14	2021-05-01	1	2	6	N	IPARI ELEKTRONIKA.pdf	Igazolás Visszaautasítás
25	r-45	Italgártó	2021-12-31	2021-12-31	2021-12-31	2	3	8	N	IPARI	Igazolás

Írjon ide a kereséshez

21:01 2021. 04. 20.

Az igazolt számlák menüpont egy lekérdezés, ami az 'I', igazolt státuszú számlákat hozza fel egy listára. Az itt látható számlák kerülhetnek kifizetésre.

Számla iktatási rendszer

localhost/iktatasi\_rendszer/index.php?p=igazolt.php

Games Réka Egyéb Programozás Google Fordító YouTube Google The Healing Codes... μBit :: Letöltés Kalotaszegi csárdás... További könyvtárak

Főoldal Számlák iktatása Számlák igazolása Igazolt számlák Visszaautasított számlák Partnerek kezelése Felhasználók kezelése **Kijelentkezés**

### Igazolt számlák

Ikt.	Számlaszám	Név	Szla. kelte	Telj. dat.	Fiz. hat.	Nettó	Áfa	Bruttó	Igazolt	PDF
1	SZ2021-00233	Energia Zrt.	2021-04-01	2021-04-01	2021-04-22	100	27	127	I	Inpiro.pdf
4	SZ2021-00250	Energia Zrt.	2021-04-01	2021-04-01	2021-04-22	100	27	127	I	Inpiro.pdf

Írjon ide a kereséshez

21:06 2021. 04. 20.

A Visszaautasított számlák menüpont szintén egy lekérdezés, itt láthatók azok a bizonylatok, melyeket a számlaigazoló visszaautasított. Ezeket a számlákat vissza kell küldeni a partnernek, aki kiállította.

Számla iktatási rendszer

localhost/iktatasi\_rendszer/index.php?p=visszautasitott.php

Games Réka Egyéb Programozás Google Fordító YouTube Google The Healing Codes... µBit :: Letöltés Kalotaszegi csárdás... További könyvtárak

Főoldal Számlák iktatása Számlák igazolása Igazolt számlák Visszaútasított számlák Partnerek kezelése Felhasználók kezelése **Kijelentkezés**

### Visszaútasított számlák

Ikt.	Számlaszám	Név	Szla. kelte	Telj. dát.	Fiz. hat.	Nettó	Áfa	Bruttó	Igazolt	PDF
3	2021/00144	Italgártó Zrt.	2021-03-16	2021-03-10	2021-04-01	-10	-3	13	V	<a href="#">Sárecz.pdf</a>
23	2021-5566	Fekete Péter	2021-03-29	2021-03-31	2021-05-06	1	2	3	V	<a href="#">Sárecz.pdf</a>

Írjon ide a kereséshez

21:08 2021. 04. 20.

## Tesztelési dokumentáció

Az alábbiakban az adatbázis kapcsolódásért felelős config.php sikeres tesztelését láthatjuk egy egységteszt segítségével, ami a PHPUnit keretrendszerrel készült.

```
app > config.php
1  <?php
2  $host = 'localhost';
3  $db_name = 'iktatasi_rendszer';
4  $db_username = 'root';
5  $db_password = '';
6  $options = [PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION, PDO::ATTR_DEFAULT_FETCH_MODE =>
7  PDO::FETCH_ASSOC, PDO::ATTR_EMULATE_PREPARES => 0];
8  try{
9      $db = new PDO("mysql:host=$host;dbname=$db_name; charset=utf8",$db_username,$db_password,$options);
10 }catch(PDOException $e) {
11     echo "!!! Az adatbázis kapcsolódás sikertelen !!!";
12     exit;
13 }
14
15 ?>
```

```
tests > configTest.php
1  <?php
2  class ConfigTests extends \PHPUnit\Framework\TestCase{
3      function testConnectionOk(){
4          include "app\config.php";
5          $this->assertEquals(is_object($db),true);
6      }
7  }
8  ?>
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/powershell>

PS C:\xampp\htdocs\iktatasi\_rendszer> ./vendor/bin/phpunit  
PHPUnit 7.0.0 by Sebastian Bergmann and contributors.

Runtime:        PHP 7.2.33  
Configuration: C:\xampp\htdocs\iktatasi\_rendszer\phpUnit.xml

.

1 / 1 (100%)

Time: 171 ms, Memory: 4.00 MB

OK (1 test, 1 assertion)

PS C:\xampp\htdocs\iktatasi\_rendszer>

# Továbbfejlesztési lehetőségek

1. A program bővítése nyomtatás funkcióval azok számára, akik papíron szeretik az adatokat elemezni. Ezt az Igazolt számlák és a Visszautasított számlák menüpontoknál tudom elképzelni.
2. Az igazolt számlákról készülhetne adatexport egy .csv file-ba, ami importálható lenne egy pénzügyi, számviteli rendszerbe további felhasználásra.
3. A visszautasított számlákról automatikus e-mail küldése az érintett partnernek, egy előre meghatározott sablon e-mail szövegtörzssel.

# Források

- <https://www.php.net/>
- <https://www.w3schools.com/>
- <https://github.com/>
- <http://webprogramozas.inf.elte.hu/webprog/ea/02/?print-pdf#/>
- <https://mek.oszk.hu/09700/09713/>

## A program elérhetősége

A forráskód helye: [https://github.com/moonieus/iktatasi\\_rendszer.git](https://github.com/moonieus/iktatasi_rendszer.git)

A program a következő URL-ről indítható: <http://moonieus.nhely.hu>