Pannon Egyetem

Műszaki Informatikai Kar

Programtervező informatikus felsőoktatási szakképzés

ZÁRÓDOLGOZAT

SharePoint

Máté Ferenc

Csontos Balázs

2022

**Hallgatói nyilatkozat**

Alulírott **Máté Ferenc** hallgató kijelentem, hogy a záródolgozatot a Pannon Egyetem Műszaki Informatikai Karán készítettem felsőfokú fejlesztő programtervező informatikusasszisztens végzettség megszerzése érdekében.

Kijelentem, hogy a dolgozatban lévő érdemi rész saját munkám eredménye, az érdemi részen kívül csak a hivatkozott forrásokat (szakirodalom, eszközök stb.) használtam fel.

Tudomásul veszem, hogy a dolgozatban foglalt eredményeket a Pannon Egyetem saját céljaira szabadon felhasználhatja.

Veszprém, 2021. december 7.

aláírás

**Témavezetői/ Szakfelelősi\* nyilatkozat**

Alulírott <témavezető/szakfelelős neve> témavezető/szakfelelős\* kijelentem, hogy a záródolgozatot <**Máté Ferenc**> a Pannon Egyetem Műszaki Informatikai Karán készítette felsőfokú fejlesztő programtervező informatikus-asszisztens végzettség megszerzése érdekében.

Kijelentem, hogy a dolgozat védésre bocsátását engedélyezem.

Veszprém, 2021. december 7.

aláírás

\*: A záródolgozatukat a szakmai gyakorlat során végzett projektmunka, fejlesztési feladat témából elkészítő hallgatók dolgozata a szakfelelős nyilatkozatával adható be. A Projekt labor és tervezés tárgyat 2020/21/1 félévben vagy azt követően teljesítő hallgatók záródolgozatának témája a tárgy keretében végzett egyéni fejlesztés feladat(rész). A Záródolgozatuk beadásának feltétele a kitöltött, aláírt témavezetői nyilatkozat.

**Köszönetnyilvánítás**

**Tartalomjegyzék**

[Szöveges specifikáció 2](#_Toc58520749)

[Aktorok és használati esetek 3](#_Toc58520750)

[Use case diagram 4](#_Toc58520751)

[Use case táblázatok 5](#_Toc58520752)

[Adatbázis felépítése 8](#_Toc58520753)

[Pattern és felépítése 11](#_Toc58520754)

[MVC patterns felosztás 13](#_Toc58520755)

[elvégzet munka leírása 14](#_Toc58520756)

[Model 14](#_Toc58520757)

[View 19](#_Toc58520758)

[Controller 2](#_Toc58520759)3

[Források 26](#_Toc58520756)

[Tartalomjegyzék 27](#_Toc58520756)

**Az ötlet**

A feladatom egy fájlfeltöltő oldal elkészítése volt. Manapság a fájlfeltöltő oldalaknak sok fajtája létezik és terjedt el az interneten. Beszélhetünk itt különböző Drive-okról, pl.: Google Drive, iCloud stb. Ezeknek a fő célja a fájlok tárolása a felhasználó által szeparálva. Felhasználónak lehetősége van megosztani, de lényegében csak egy privát feltöltő oldalként funkcionálhat. Én ezen felül szerettem volna kerekedni és egy bizonyos célt szerettem volna adni az oldalnak. Végül a koronavírusban megélt tapasztalatok adtak ötletet a téma kiválasztásához.   
A koronavírus okozta távoktatás nagyba megnehezítette a csapatmunkát. A közös munka legnagyobb nehézsége az volt, hogy nem láttuk a másik munkáját. Különféle fájlmegosztó drive-ok segítségével osztottuk meg egymással a munkát. Egy olyan weboldalt szerettem volna létrehozni, ami segítségével sokkal könnyebé válik a csapatban dolgozás, akár tanár(okk)al együtt is. Az elkészült fájlok könnyű feltöltése, kategóriába sorolása után minden felhasználó képes látni a feltöltött fájlokat, illetve le tudja őket tölteni. Esetlegesen a tanároknak is sokkal egyszerűbb lenne a kiadott beadandó feladatokat összegyűjteni, feladatokat kiírni minta vagy követelmény dokumentumokkal.   
Természetesen az oldalra sok másféle témát rá lehetne húzni viszont én a tanulási folyamat, illetve a csapatmunka, több emberrel való dolgozás folyamatát szerettem volna megkönnyíteni. Ötletet főként a kari Moodle rendszerből merítettem.

**Szöveges specifikáció**

**A Weboldal célja**

A weboldal célja a felhasználó által elkészített dokumentumok, fájlok tárolása, fel és letöltése, megosztása.

**Admin felület**

A weboldalra az adminnak van jogosultsága felhasználót regisztrálni, ennek szintje lehet oktató, hallgató. A felhasználóknak új jelszót az admin generálhat és a neptunban megadott egyetemi e-mail címre küldheti ki. A felhasználónak a fájl feltöltésekor mindenképpen ki kell választania a fájl kategóriáját. Az adminnak lehetősége van a feltöltött fájlok kategóriáját létrehozni és törölni. Az admin ezeken felül törölheti és módosíthatja a felhasználókat, illetve a fájlokat. Az admin ellenőrizheti a log oldalon az elmúlt műveleteket, melyeket a felhasználók végeztek. pl: bejelentkezés, fájl feltöltése, vagy módisítás.

**felhasználó felület**

Hallgatókat és az oktatókat az admin regisztrálja neptunkód és név alapján. Bejelentkezés után a weboldal listázza a fájlokat. A mappák és fájlok között lehetőség van kulcsszavak (neptunkód, cím, téma,felhasználó) szerint keresni. A feltöltött módosíthatóak és letölthetők. A feltöltött fájl formátuma lehet mp3, mp4, jpg, png, pdf, docx. A feltöltött dokumentumot a hallgató kategória szerint módosíthatja, törölheti és letölthetik. Az oktatónak lehetősége van még kategóriákat kezelni. új kategóriát képes létrehozni illetve módosítani és törölni is tud egy létezőt.

**Aktorok és használati esetek**

**Admin**

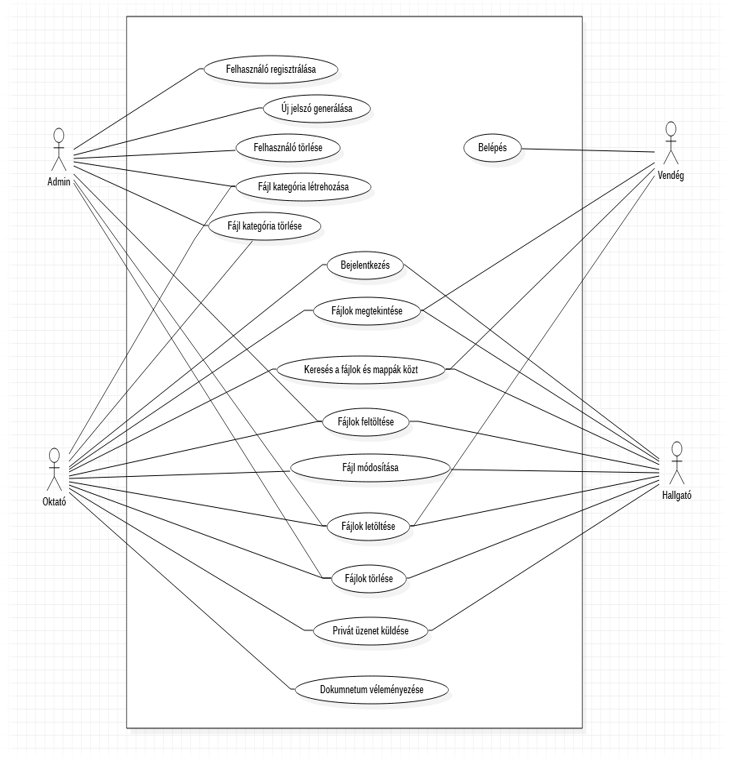
* Felhasználó regisztrálása
* Új jelszó generálása és küldése
* Fájl kategóriák létrehozása
* Fájl kategóriák törlése
* Felhasználó törlése
* Fájlok feltöltése
* Fájlok törlése
* Fájlok letöltése

**Oktató**

* Bejelentkezés
* Keresés a fájlok közt
* Fájlok letöltése
* Fájlok feltöltése
* Fájl kategóriák létrehozása
* Fájl kategóriák törlése
* Fájl módosítása
* Fájlok törlése

**Hallgató**

* Bejelentkezés
* Keresés a fájlok közt
* Fájlok letöltése
* Fájlok feltöltése
* Fájlok módosítása
* Fájlok törlése

**Use case diagram**

[Ide írja az idézet forrását.]

**Use case táblázatok**

|  |  |
| --- | --- |
| Név | Bejelentkezés |
| Követelmény | Regisztrált hallgató vagy oktató fiók |
| Cél | A hallgató vagy oktató be tudjon lépni |
| Előfeltétel | A hallgató vagy oktató legyen regisztrálva |
| Sikeres lefutás | Be tud lépni a felhasználói felületre |
| Sikertelen lefutás | Belépés megtagadása, hibaüzenet |
| Elsődleges aktor | Hallgató, Oktató, Admin |
| Másodlagos aktor |  |
| Kiváltó esemény | A hallgató vagy oktató be akar lépni |
| Fő lépések | |  |  | | --- | --- | | Lépés | Tevékenység | | 1 | Az aktor kezdeményezi a bejelentkezést | | 2 | A rendszer megjeleníti a bejelentkezési felületet | | 3 | Az aktor megadja az adatokat | | 4 | A rendszer ellenőrzi, hogy az adatok megfelelőek e | | 5 | A rendszer elérhetővé teszi a felhasználói felületet | | 6 |  | |
| Kiegészítések | |  |  | | --- | --- | | Lépés | Elágazó tevékenység | |  | A rendszer hibaüzenetet küld, ha az adatok nem megfelelőek | |  | Elfelejtett jelszó esetén az admin újat generál és elküldi a megadott e-mail címre | |  |  | |

|  |  |
| --- | --- |
| Név | Fájl feltöltése |
| Követelmény | Fájl adatainak ismerete |
| Cél | Új fájl létrehozása |
| Előfeltétel | Új fájl még nem szerepel a listában |
| Sikeres lefutás | Új fájl megjelenik a listában |
| Sikertelen lefutás | Új fájl nem jelenik meg a listában |
| Elsődleges aktor | Hallgató, Oktató |
| Másodlagos aktor | Admin |
| Kiváltó esemény | A felhasználó új fájlt szeretne feltölteni |
| Fő lépések | |  |  | | --- | --- | | Lépés | Tevékenység | | 1 | Az aktor fájlt szeretne feltölteni | | 2 | A rendszer kéri a kategóriaválasztást | | 3 | Az aktor kiválasztja a kategóriát | | 4 | A rendszer tárolja a fájl adatait | | 5 | Rendszer feltölti a fájlt | | 6 | A rendszer megjeleníti a fájlt | | 7 |  | | 8 |  | |
| Kiegészítések | |  |  | | --- | --- | | Lépés | Elágazó tevékenység | |  | Aktor hiányosan töltötte ki az űrlapot | |  |  | |  |  | |

|  |  |
| --- | --- |
| Név | Fájl törlése |
| Követelmény | A törlendő fájl kiválasztása |
| Cél | A már nem aktuális fájl törlése |
| Előfeltétel | Legyen egy fájl a rendszerben |
| Sikeres lefutás | A fájl törlése |
| Sikertelen lefutás | A fájl törlése sikertelen |
| Elsődleges aktor | Hallgató, Oktató |
| Másodlagos aktor | Admin |
| Kiváltó esemény | Az aktor listázza a fájlokat |
| Fő lépések | |  |  | | --- | --- | | Lépés | Tevékenység | | 1 | Az aktor listázza a fájlokat | | 2 | A rendszer listázza a fájlokat | | 3 | Az aktor kiválasztja a törlendő fájlt | | 4 | Az aktor törli a fájlt | | 5 | A rendszer törli a fájlt | | 6 |  | | 7 |  | | 8 |  | |
| Kiegészítések | |  |  | | --- | --- | | Lépés | Elágazó tevékenység | |  | Az aktor nem megfelelően kezeli a törlési panelt | |  |  | |  |  | |

Ezek csak a legfontosabb események

**Adatbázis felépítése**

Az adatbázis a phpMyAdmin 4.9.7 –es verziójában fut és utf8\_hungarian\_ci illesztést használ.

**Adattáblák:**

**Users** (ID, Username, Password, Admin)

ID Felhasználó azonosítója (szám), elsődleges kulcs, auto increment

Username Felhasználó neve, általában neptun kódja (szöveg [8])

Password Felhasználó jelszava (szöveg [32]), MD5 titkosítás

Admin Felhasználó admin-e? (szám)

**Files** (ID, Filename, Link, OwnerID, CategoryID, Public)

ID Fájl azonosítója (szám), elsődleges kulcs, auto increment

Filename Fájl neve (szöveg [255])

Link Fájl elérési útvonala (szöveg [255])

OwnerID Tulajdonos felhasználó azonosítója (szám)

CategoryID Kategória azonosítója (szám)

Public Az állomány publikus-e? (boolean)

**Categorys** (ID, CatName, Description)

ID Kategória azonosítója (szám), elsődleges kulcs, auto increment

CatName Kategória neve (szöveg [255])

Description Kategória leírása (szöveg[65535])

**Log** (ID, Date, Log)

ID Változás azonosítója (szám), elsődleges kulcs, auto increment

Date Változás időpontja (dátum)

Log Változás leírása (szöveg[65535])

**Triggerek**

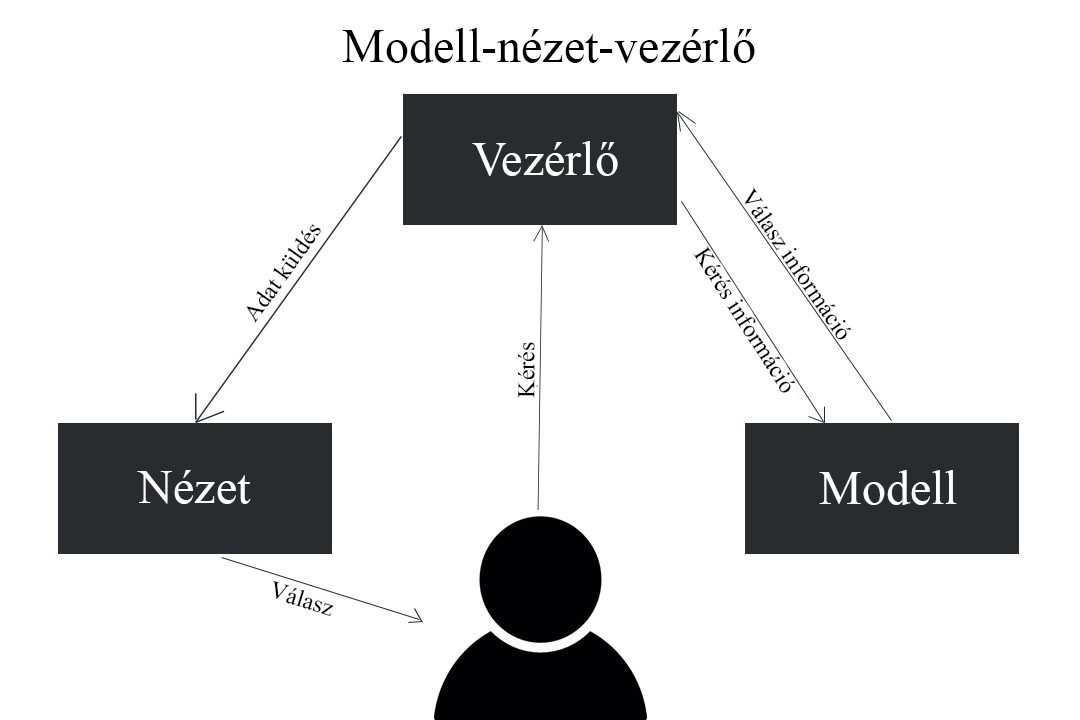
A log tábla kivételével minden tábla tartalmaz egy triggert ami rögzíti az összes változást a Log táblába.

A Categorys tábla tartalmaz egy triggert, amely rekord törlésére aktiválódig.  
A trigger lefutásakor a törölt kategóriában lévő fájlok kategóriáját módosítja alapértelmezettre. Az alapértelmezett kategória nem törölhető az adattáblából.

**Pattern és felépítése**

**Modell-nézet-vezérlő (MNV)**

A modell-nézet-vezérlő (MNV) (angolul model-view-controller) a szoftvertervezésben használatos programtervezési minta. Összetett, sok adatot a felhasználó elé táró számítógépes alkalmazásokban gyakori fejlesztői kívánalom az adathoz (modell) és a felhasználói felülethez (nézet) tartozó dolgok szétválasztása, hogy a felhasználói felület ne befolyásolja az adatkezelést, és az adatok átszervezhetők legyenek a felhasználói felület változtatása nélkül. A modell-nézet-vezérlő ezt úgy éri el, hogy elkülöníti az adatok elérését és az üzleti logikát az adatok megjelenítésétől és a felhasználói interakciótól egy közbülső összetevő, a vezérlő bevezetésével



**Modell**

Az alkalmazás által kezelt információk tartomány-specifikus ábrázolása. A tartománylogika jelentést ad a puszta adatnak (pl. kiszámolja, hogy a mai nap a felhasználó születésnapja-e, vagy az összeget, adókat és szállítási költségeket egy vásárlói kosár elemeihez).

Sok alkalmazás használ állandó tároló eljárásokat (mint mondjuk egy adatbázis) adatok tárolásához. Az MNV nem említi külön az adatelérési réteget, mert ezt beleérti a modellbe.

**Nézet**

Megjeleníti a modellt egy megfelelő alakban, mely alkalmas a felhasználói interakcióra, jellemzően egy felhasználói felületi elem képében. Különböző célokra különböző nézetek létezhetnek ugyanahhoz a modellhez.

**Vezérlő**

Az eseményeket, jellemzően felhasználói műveleteket dolgozza fel és válaszol rájuk, illetve a modellben történő változásokat is kiválthat.

**Előnyök**

* Egyidejű fejlesztés – Több fejlesztő tud egyszerre külön a modellen, vezérlőn és a nézeteken dolgozni.
* Magas szintű összetartás – MNV segítségével az összetartozó funkciók egy vezérlőben csoportosíthatóak. Egy bizonyos modell nézetei is csoportosíthatóak.
* Függetlenség – MNV mintában az elemek alapvetően nagy részben függetlenek egymástól
* Könnyen változtatható – Mivel a felelősségek szét vannak választva a jövőbeli fejlesztések könnyebbek lesznek
* Több nézet egy modellhez – Modelleknek több nézetük is lehet
* Tesztelhetőség - mivel a felelősségek tisztán szét vannak választva, a külön elemek könnyebben tesztelhetőek egymástól függetlenül

**MVC patterns felosztása**

Az MNV felépítésnek köszönhetően a felosztás nagyon egyszerű.

* Megjelenítéshez szükséges kódrészleteket (**Nézet**)
* Adatok és adatbázis kezelését. (**Model)**
* Ezek közti kapcsolatot és a felhasználó kéréseinek feldolgozáshoz tartozó kódrész (**Vezérlő)**

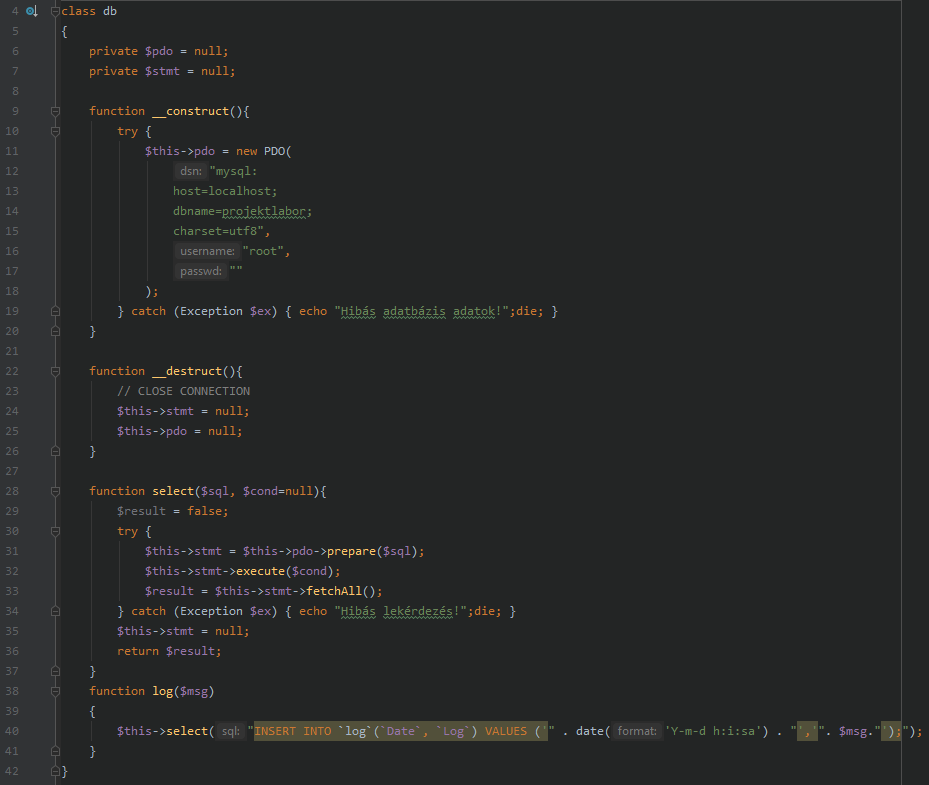
**elvégzet munka leírása**

## **Model**

A weboldal felépítésében a model felel az adatbázis kapcsolatért és annak kezeléséért.

**model/db.php**

Létrehozza az adatbázis és a weboldal közti kapcsolatot, valamint 2 alap függvényt.



**select($sql, $cond=null)**

Egy SQL lekérdezést futtat le és visszatér annak eredményével.

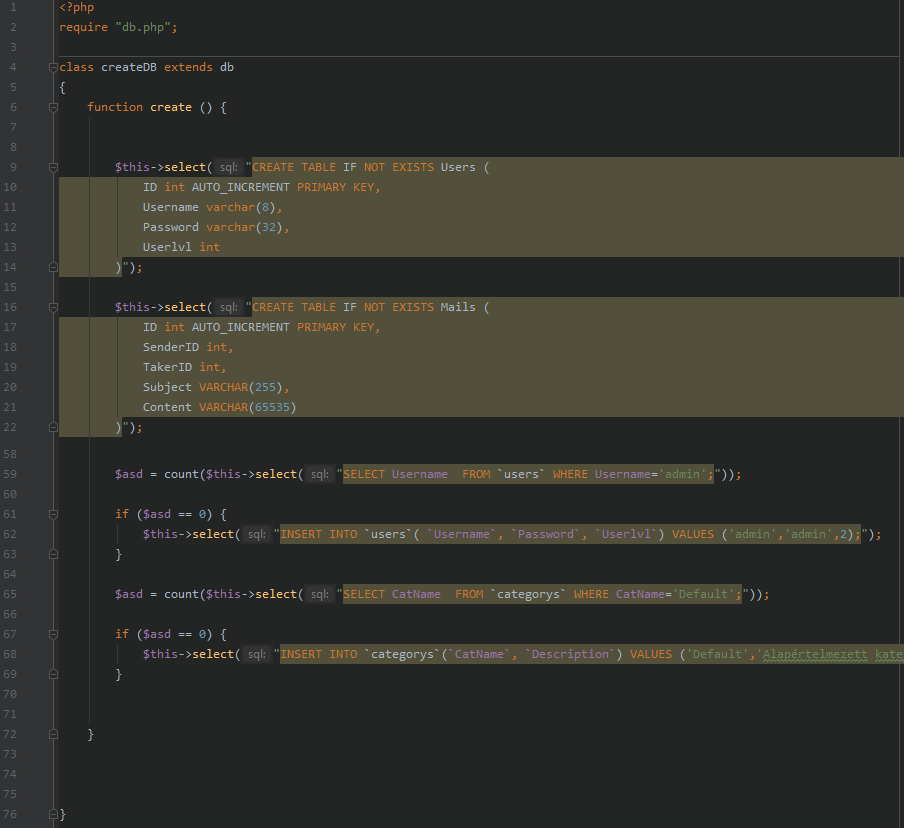
Amennyiben hibás lekérdezést hajtunk végre egy hibaüzentet ír ki.

**log($msg)**

A jelenleg dátummal és idővel egy log bejegyzést hoz létre.

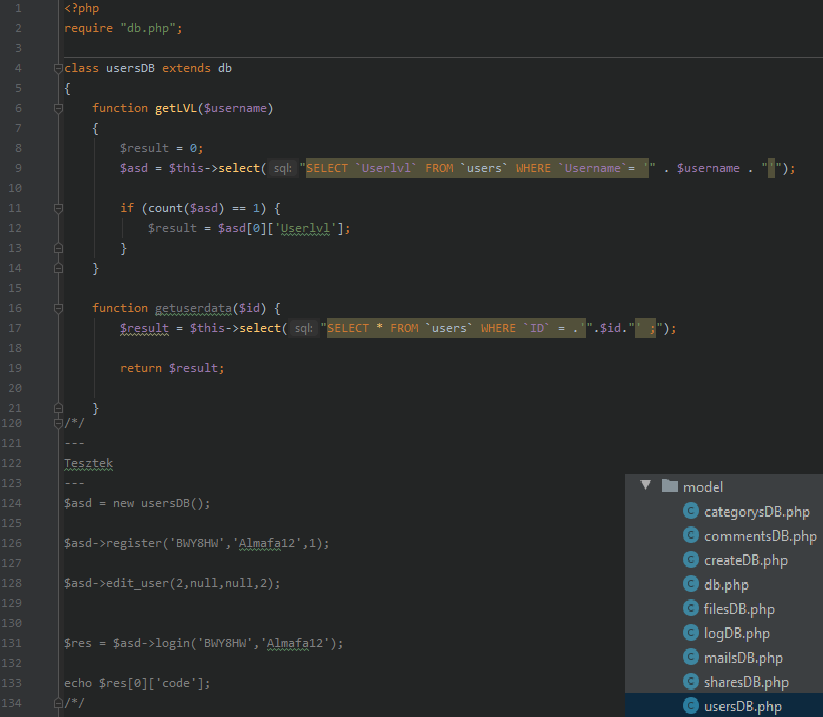
**model/createdb.php**

A weboldal első indításakor létrehozza az adattáblákat és az alapértelmezett admin felhasználót és kategóriát.



**Táblákhoz tartozó függvények**

Minden adattáblához tartoznak a kezeléshez szükséges függvények.



A fájlok végén kommentelve találhatóak a tesztek.

**Összes függvény**

**usersdb.php**

**getLVL($username)**Visszatér a felhasználó szintjével.

**getuserdata($id)**Visszatér a felhasználó összes adatával.

**getID($username)**Visszatér az adott felhasználónévhez tartozó ID-val.

**login($username,$pw)**Visszatér a bejelentkezési adatok eredményével, ha nem talált akkor a 'code' értéke 404, ha talál akkor 200.

**listall()**Visszatér az összes felhasználóval.

**register($username,$pw,$userlvl=0)**Felhasználó regisztrálása, a függvény 1-el tér vissza ha a regisztráció sikeres volt, 0-val ha nem.

**delete\_user($id)**Törli a Users táblából az ID-val megegyező felhasználót. Alapértelmezett felhasználó nem törölhető!

**edit\_user($id,$username=null,$pw=null,$userlvl=null)**Felhasználó összes adata szerkeszthető.

**filesdb.php**

**uploadfile($filename, $link, $ownerID, $categoryID, $public)**Fájl feltöltésekor az adatbázisba tárolja az adatait.

**deletefile($ID)**Fájl adatainak törlése az adattáblából.

**editfile($ID, $filename=null, $link=null, $ownerID=null, $categoryID=null, $public=null)**Fájl adatainak szerkesztése.

**getfiledata($ID)**Fájl adatainak lekérdezése.

**listallfiles()**Összes fájl listázása.

**listfiles($categoryID)**Megadott kategóriában lévő fájlok listázása.

**categorysdb.php**

**createcategory($catName,$description)**Kategória létrehozása.

**deletecategory($ID)**Kategória törlése, default kategória nem törölhető!

**editcategory($ID,$catName=null,$description=null)**Kategória adatainak szerkesztése.

**sharesdb.php**

**createShare($fileID,$sharedID)**Fájl megosztás létrehozása.

**deleteShare($fileID,$sharedID)**Fájl megosztás törlése.

**commentsdb.php**

**createComment($fileID,$commenterID,$comment)**Komment létrehozása egy fájlnál.

**deleteComment($ID)**Komment törlése.

**mailsdb.php**

**sendmail($senderID,$takerID,$subject,$content)**Üzenet küldése másik felhasználónak.

**listmails($ID)**A felhasználóhoz tartozó üzenetek listázása.

**logdb.php**

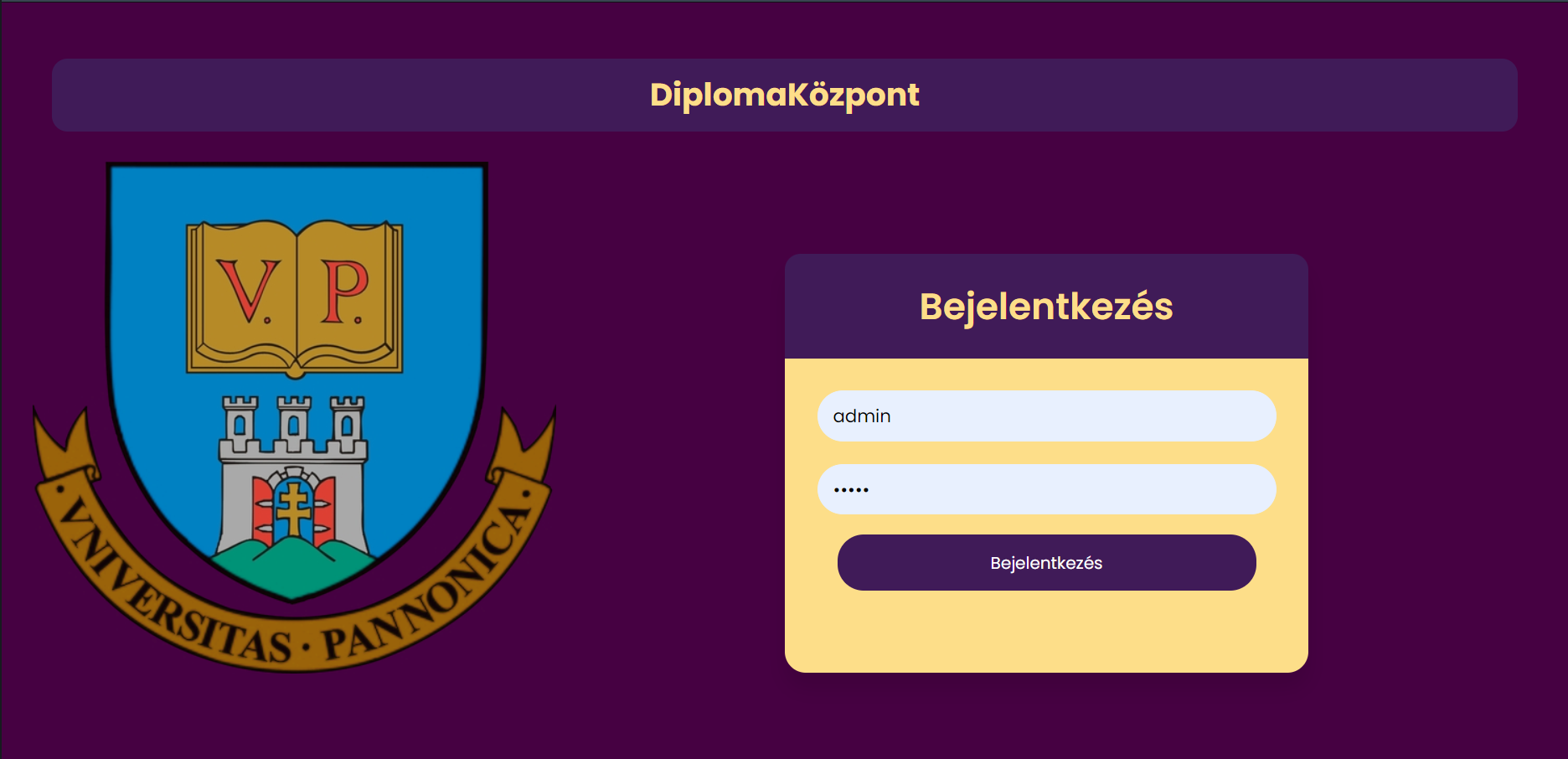
**listlog()**Kilistázza a log tábla tartalmát.

## **View**

Megjeleníti a modellt egy megfelelő alakban, mely alkalmas a felhasználói interakcióra, jellemzően egy felhasználói felületi elem képében.

**Felületek:**

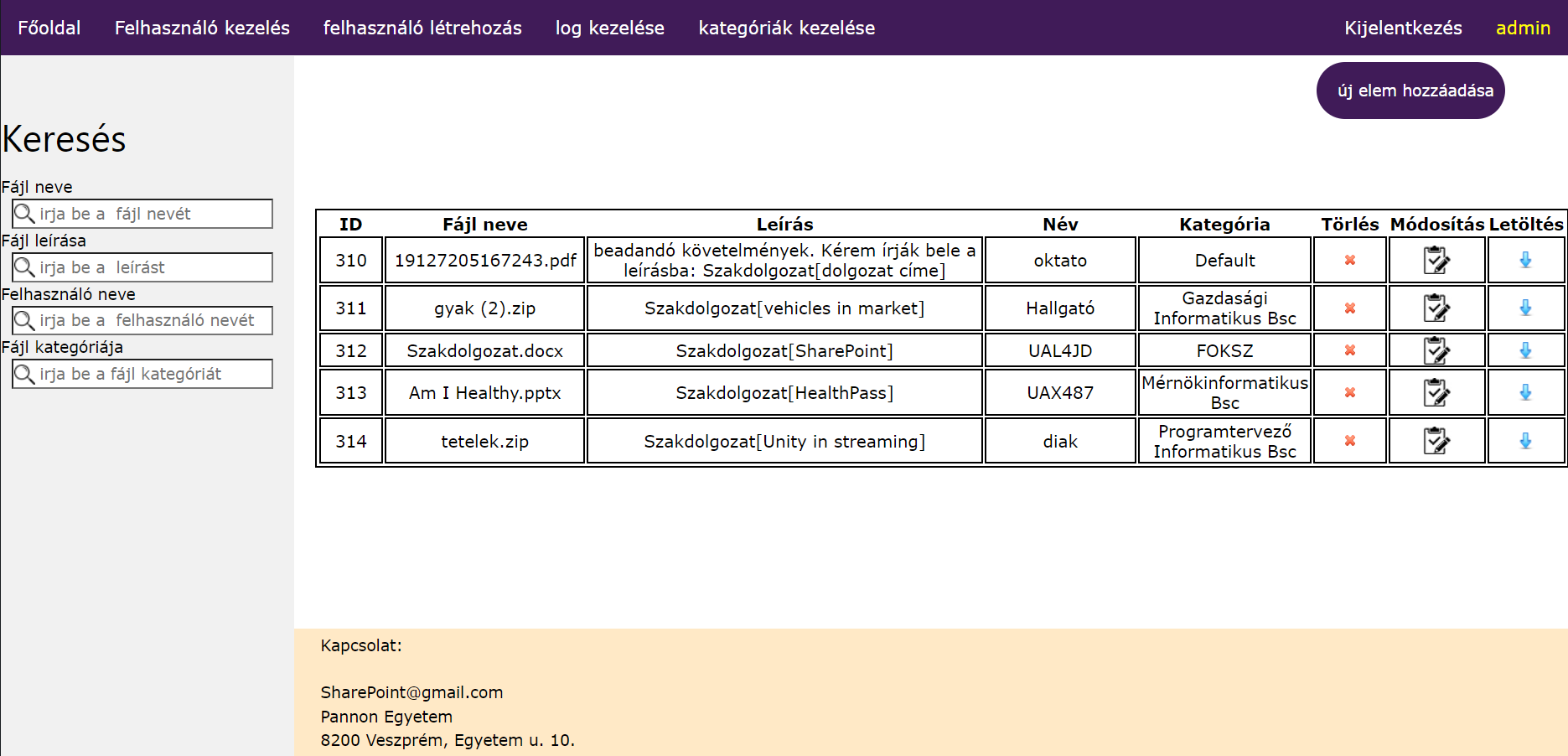
*Bejelentkező felület:*

**

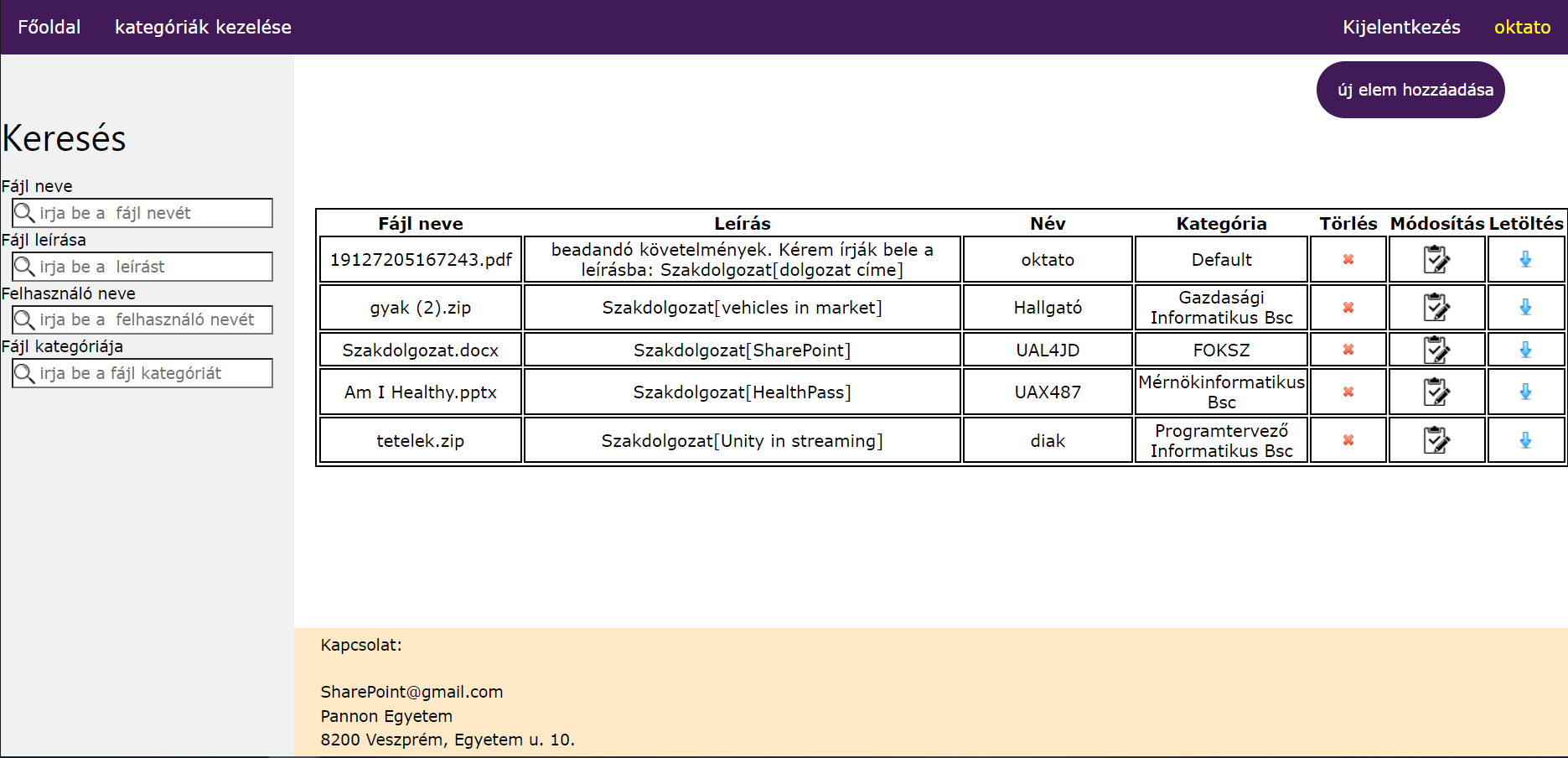
* A felhasználónak meg kell adnia a neptunkódját, és a jelszavát amennyiben be akar jelentkezni a kívánt oldalra.
* A belépés után a felhasználót a fő felületre irányítja át

**Fő felület:**

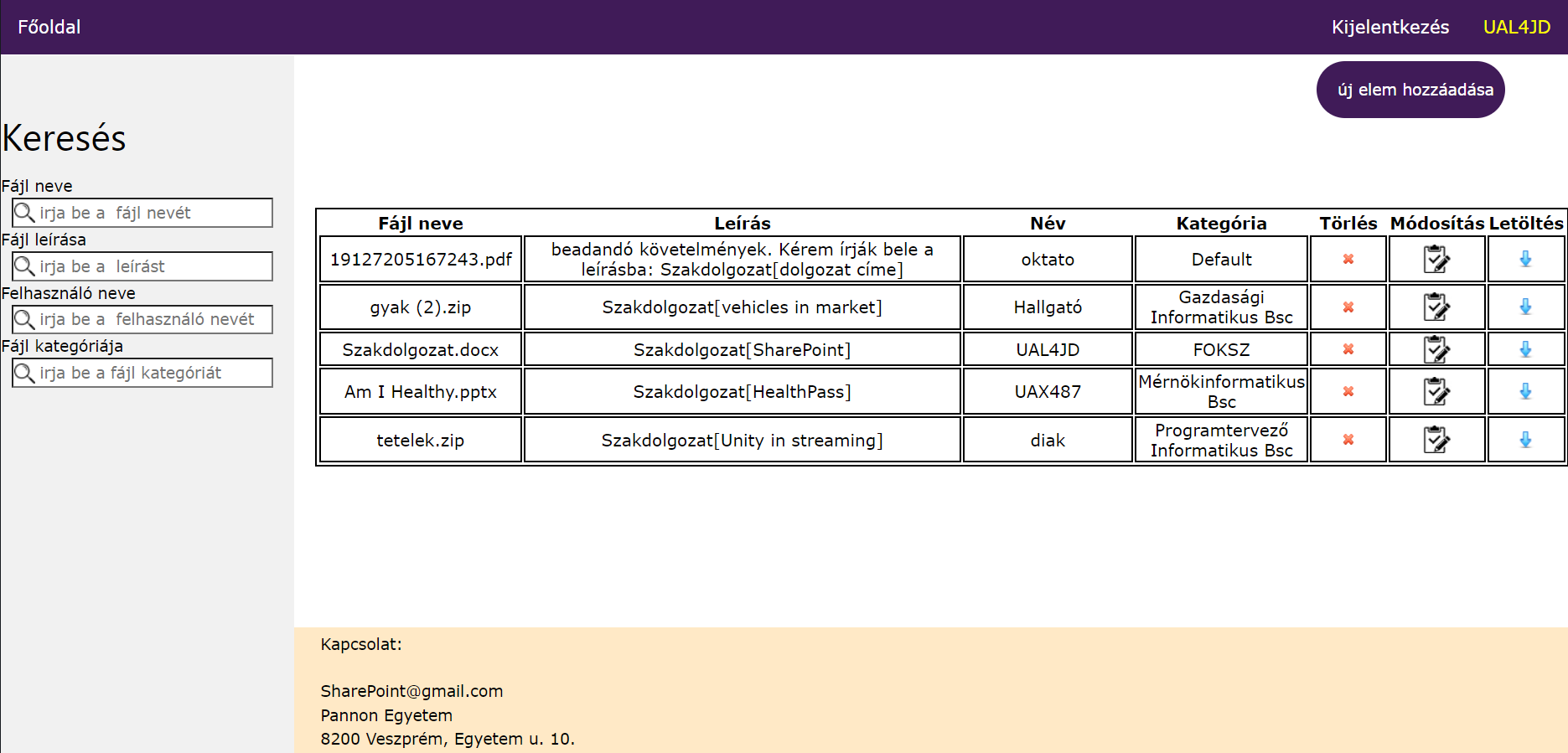
Bejelentkezni admin, oktató, illetve hallgatóként lehetséges. A fő felületen mindegyik szintű felhasználónak különféle jogosultságai vannak.

****Admin bejelentkezés:

Oktató bejelentkezés:



Hallgató bejelentkezés:



A fő felület 4 részre különíthető el:

1. A fő részre, vagyis a táblázatra
2. header, vagyis a fejléc
3. sidebar, vagyis az oldalsáv
4. footer, vagyis a lábléc

A felhasználó egy táblázatot lát ezen a felületen, ahol a felhasználók által feltöltött adatokat lehet megtekinteni. Saját fájlt feltölteni az Új elem hozzáadása gombra kattintva lehet. A feltöltött fájlt módosítani és törölni is lehetőség van.

A fájlok feltöltésénél, egy aloldal nyílik meg ami kitöltése után lehet a kívánt elemet feltölteni. Itt lehet megadni leírást és kategóriát a feltölteni kívánt fájl számára.

A footer részben információt lehet találni az oldalról.

A sidebar rész segítségével az oldal tartalmából lehet szürni, keresni lehet az elemeket különböző szempontok alapján.

A header részben pedig az oldalról különboző aloldalakra lehet eljutni. A jobb oldalán található kijelentkezés gomb, és melette a jelenleg bejelentkezett felhasználó neve található.

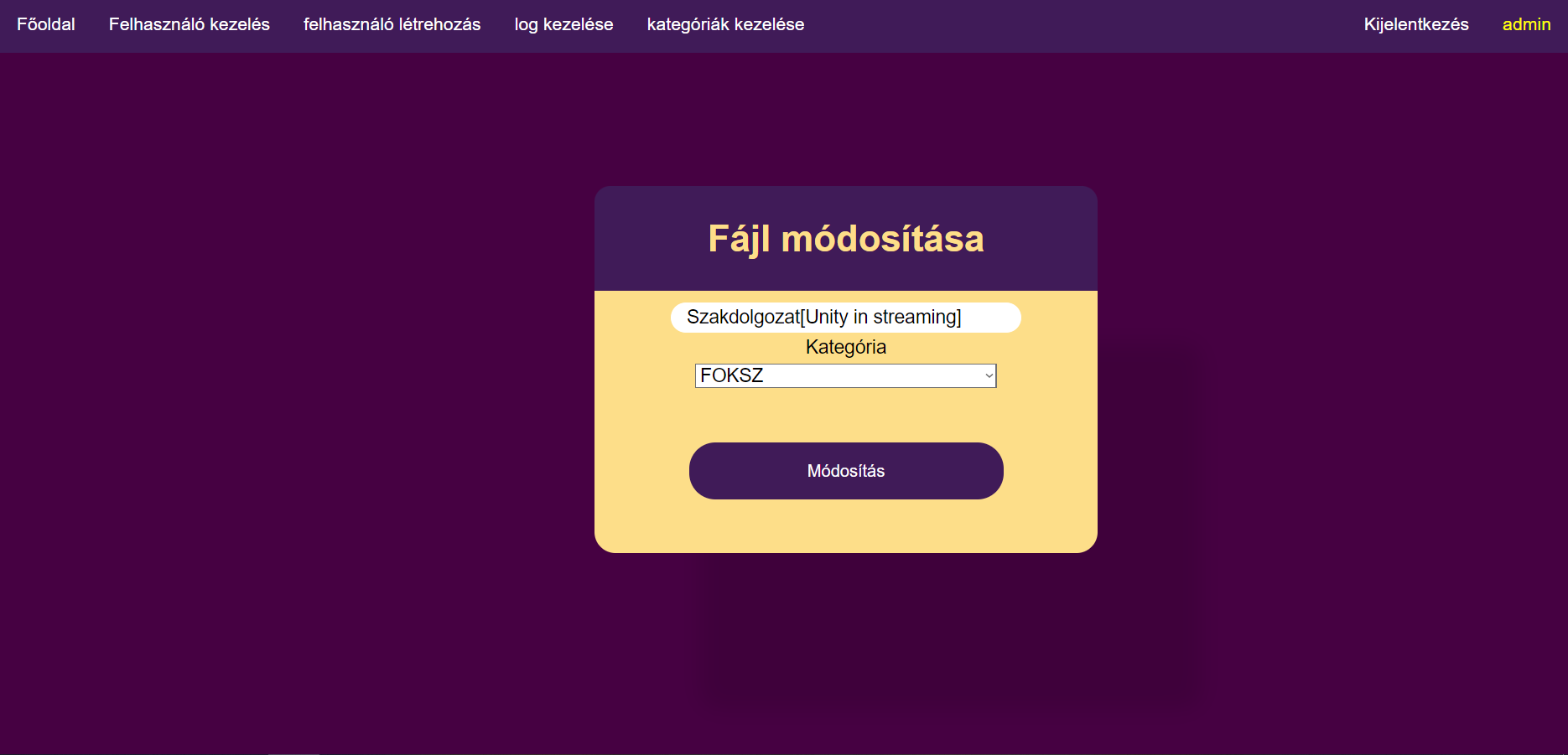
A hallgatók csak ezt a felületet látják bejelentkezés után.

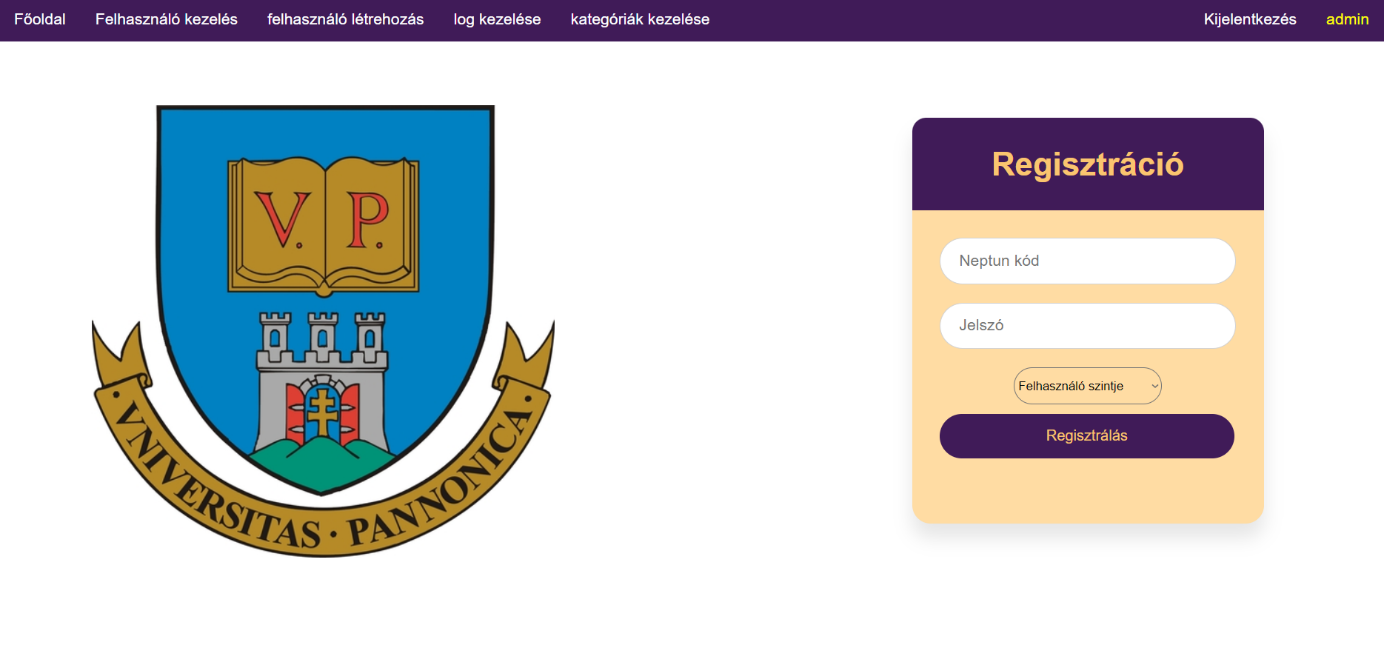
**Az aloldalak**

Új fájl feltöltése:



fájl módosítása:



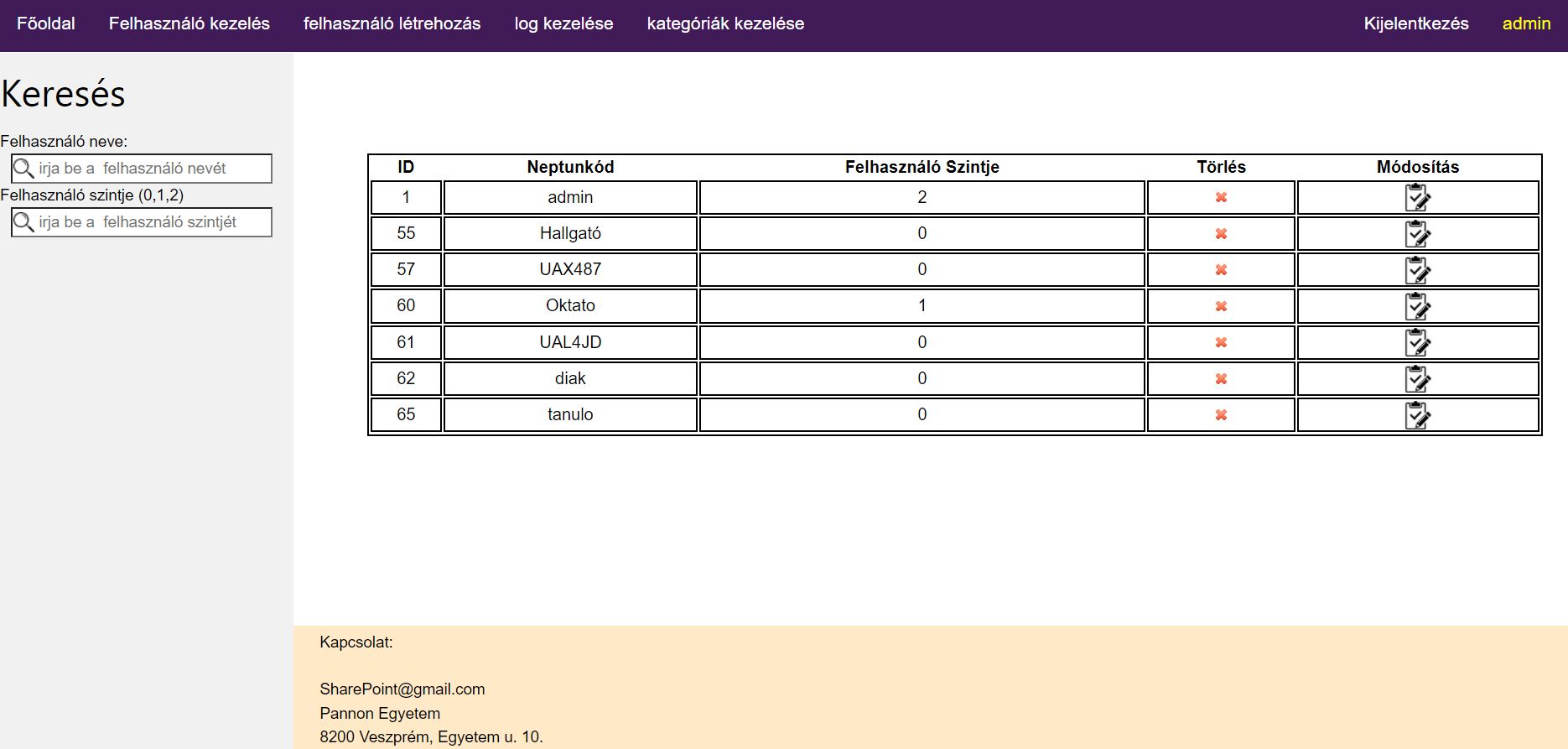
Felhasználó létrehozása:

A felhasználó regisztrálásának egyetlen módja. Tehát felhasználót regisztrálni csak az admin képes. A regisztrációhoz egy Neptunkódot, egy jelszót és egy felhasználói szintet kell megadni.

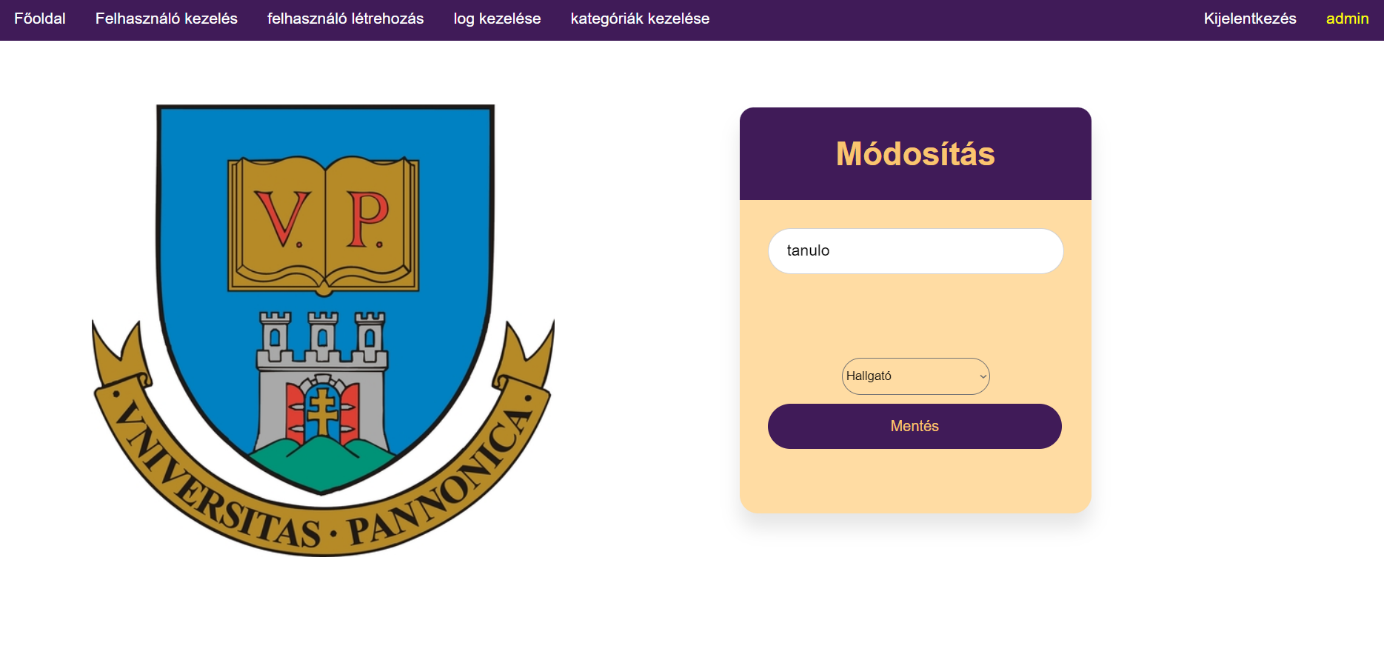
A felhasználói szint lehet:

* Admin
* Oktató
* Hallgató

Felhasználó kezelése:

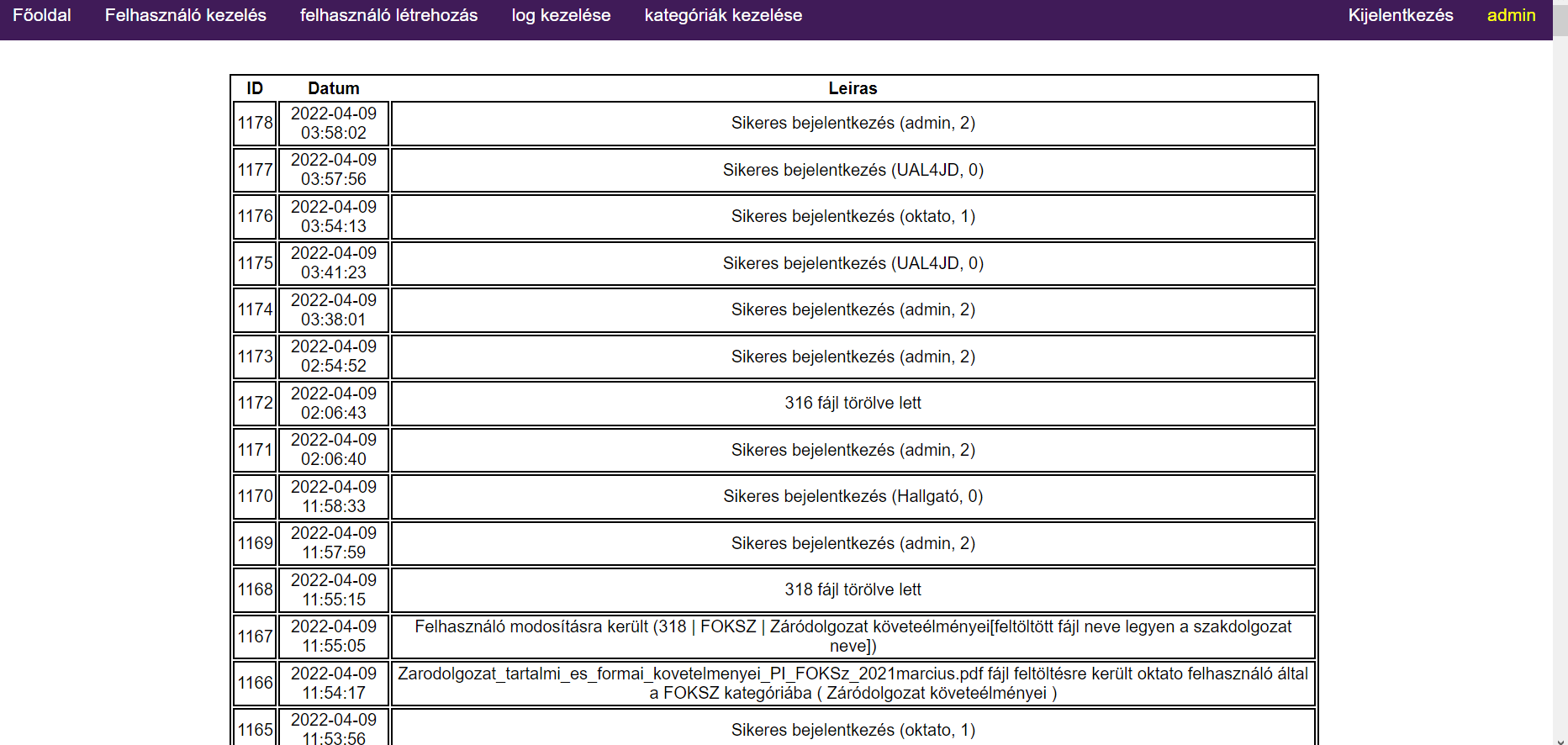
Ezen az oldalon a regisztrált felhasználók neveit és adatait láthatjuk. Lehetőség van a felhasználót törölni, illetve módosítani. Az oldalsó sidebar segítségével tudunk keresni a táblázat elemeire különböző szempontok alapján.

A felhasználó módosítása

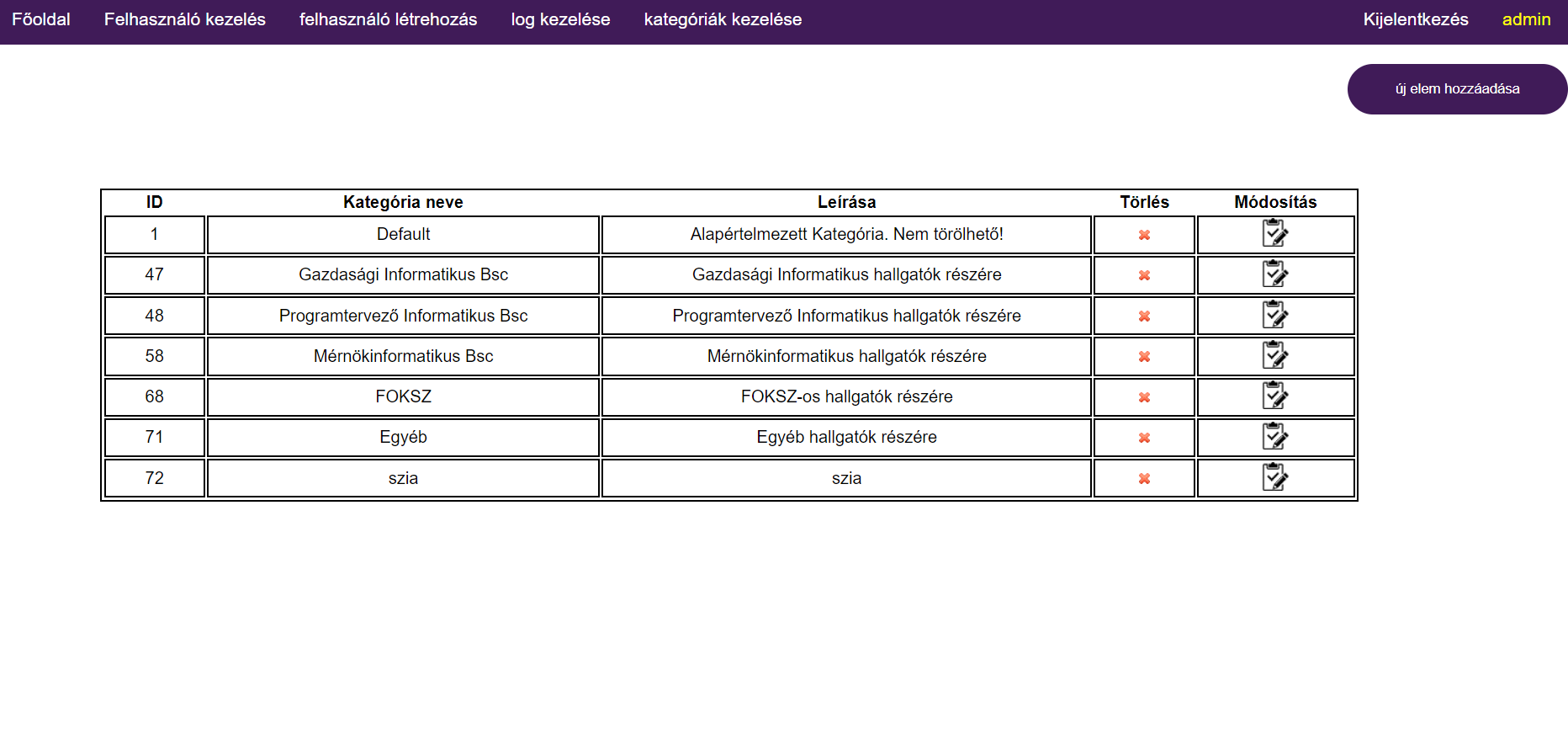


A log kezelése

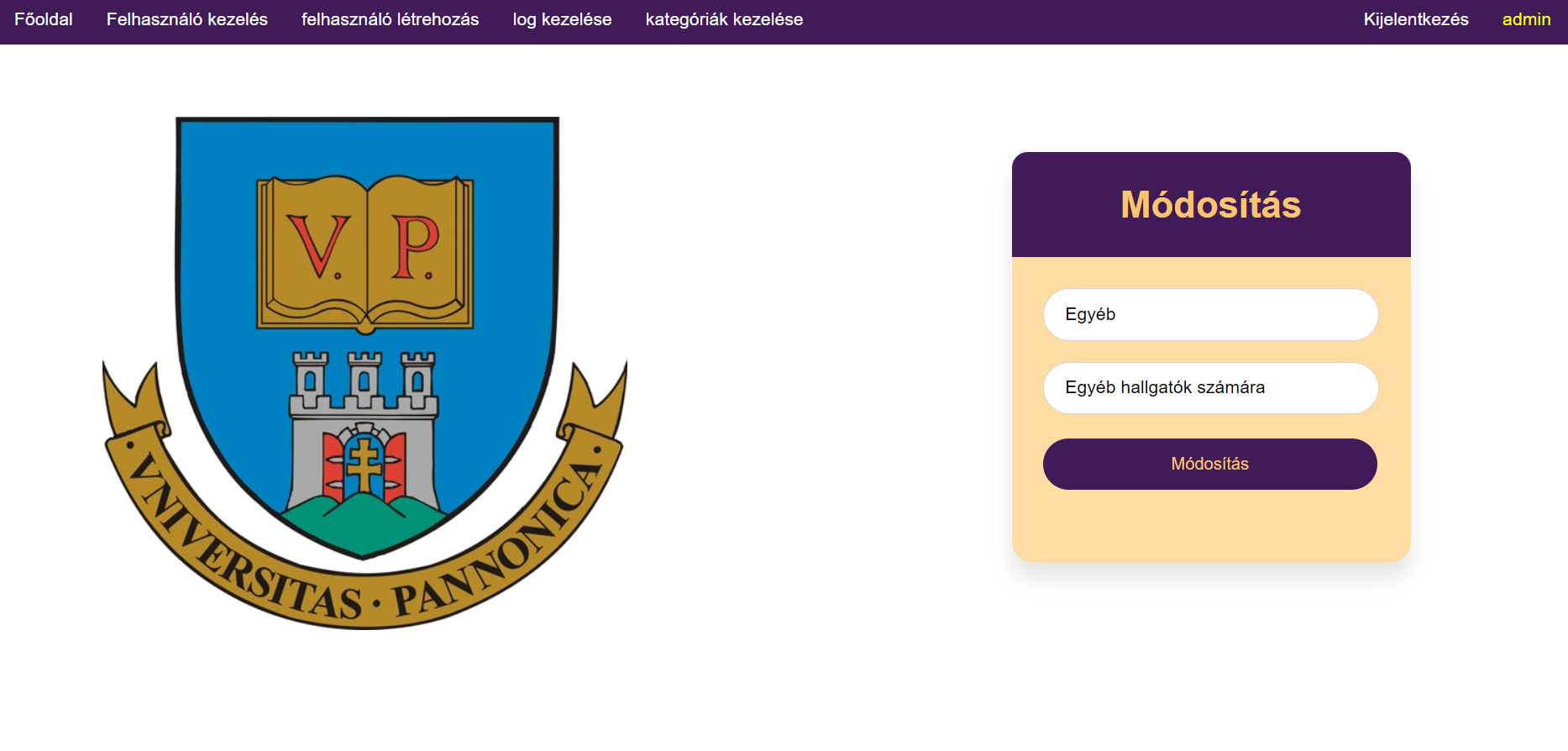
A log kezelése egy egyszerű listázást mutat. Kilistázza a fájlokkal, kategóriákkal történő műveleteket (létrehozás, módosítás, törlés), illetve a bejelentkezéseket, valamint megmutatja ezek idejét, és azt, hogy sikeres vagy nem volt sikeres. Ezt az oldalt csak az admin érheti el.



Kategória kezelése

Ezen oldalon a kategóriák listája található. A felhasználó a kategóriát a dokumentum feltöltésnél köteles megadni. Ezen az oldalon azokat a kategóriákat lehetőség van módosítani, törölni, vagy újat létrehozni. Ezt az oldalt az oktató és az admin is elérheti.

Kategória módosítása:



## **Cotroller**

Összeköti a view és a model működését.

**Bejelentkezés:**



**A kód működése:**

* A függvény ellenőrzi, a neptun kódot és a jelszót.
* Ha a neptun kód és a jelszó szerepel az adatbázisban, a függvény átirányítja a felhasználót a fő oldalra.
* Ha a neptun kód vagy a jelszó nem megfelelő, a függvény kiírja a „Hibás felhasználó név vagy jelszó” szöveget és visszairányítja a bejelentkező felületre.

**Kijelentkezés:**

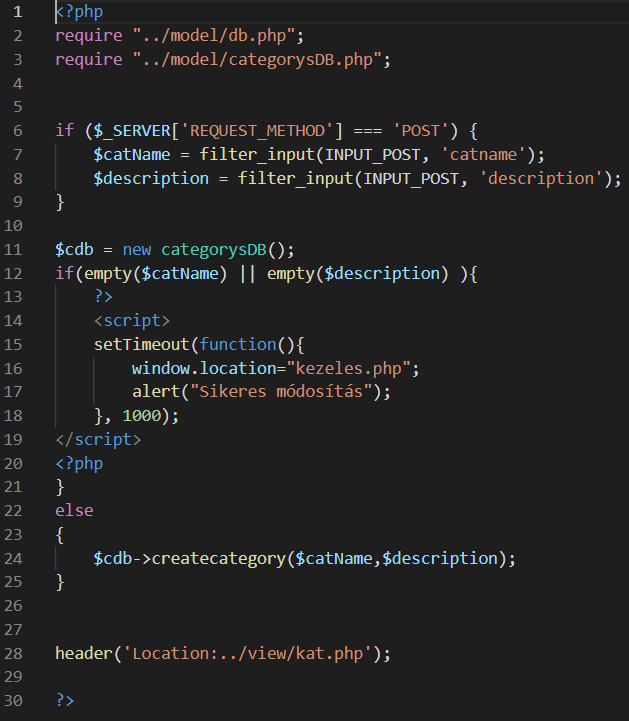
A kód működése:

* A függvény átirányítja a felhasználót a bejelentkező felületre.

**Kategória hozzáadása:**

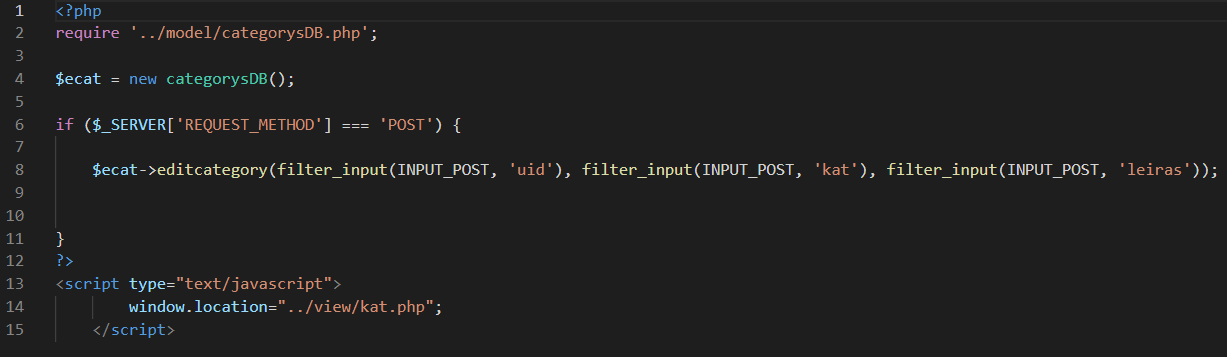
Ha egy új kategóriát szeretnénk hozzáadni, a *kategória kezelése* fülön belül válasszuk *az új elem hozzáadása* gombot

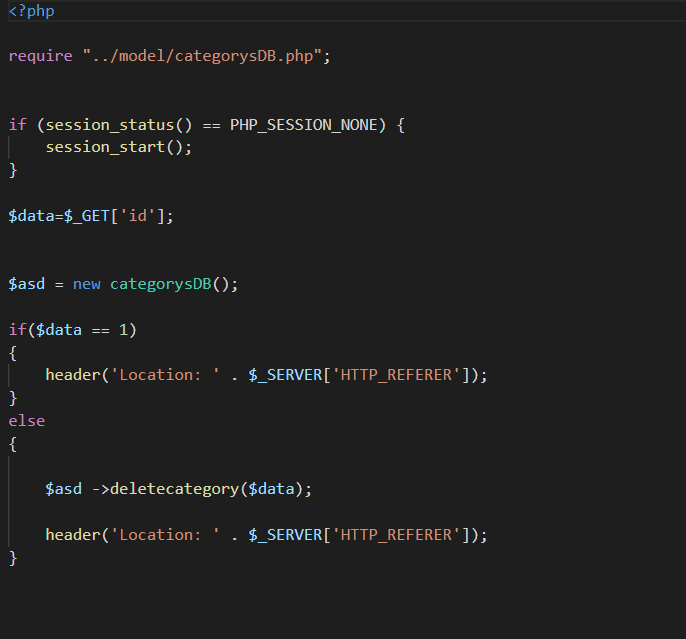
Ekkor a felugró ablakban lévő mezők kitöltésével és a feltöltés gomb megnyomásával tudunk új kategóriát létrehozni. Ezután a kezelőfelületen azonnal megjelenik a létrehozott kategória. Ezekután módosítani, illetve törölni tudjuk ezen kategóriát a táblázat jobb szélén található törlés, illetve módosítás címszó alatt lévő Ikon segítségével. A kategóriák kezeléséhez admin vagy oktató felhasználói szint szükséges!

**A kategória hozzáadása kódszinten:**

A model által definiált *creatcategory* függvény használatával történik**Kategória módosítása:**

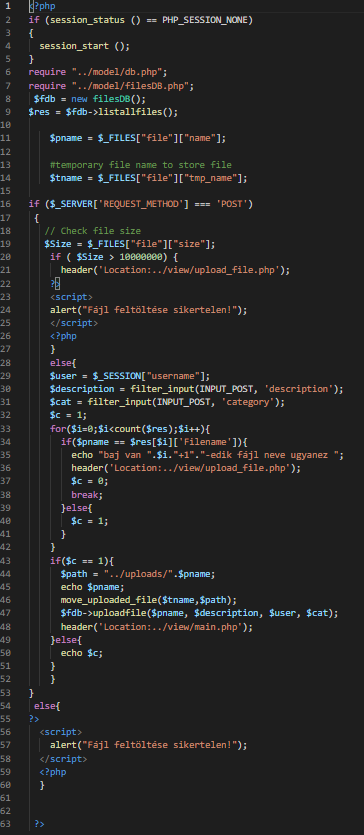
A módosítás ikonra kattintva tudjuk módosítani a már létrehozott kategóriáinkat.

A model által definiált *editcategory* függvény használatával történik

****

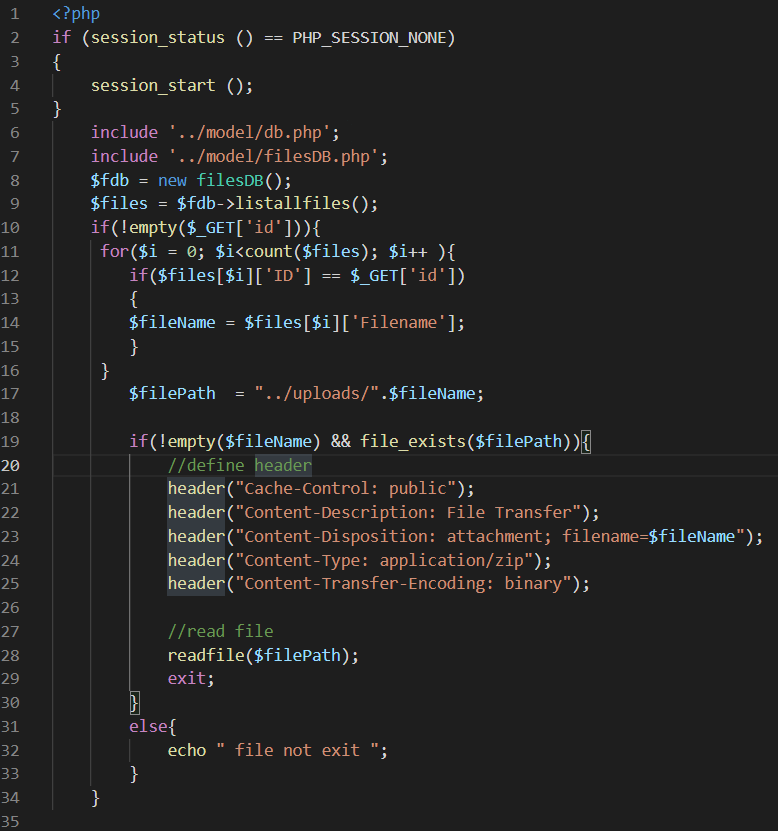
**Kategória törlése**

A model által definiált *deletecategory* függvény használatával történik

**A fájl feltöltése**

Az oldal fő célja az, hogy lehetővé tegye fájlok feltöltését felhasználó által. Egy már regisztrált felhasználó bejelentkezés után a már meglévő fájlok listáját látja. A fájl feltöltése gombra kattintva egy új oldal nyílik meg ahol a felhasználó a megfelelő adatok kitöltésével és a feltölteni kívánt fájl kiválasztásával képes lesz egy új fájlt hozzáadni a már létező fájlok listájához, amennyiben nincs ugyanolyan néven található fájl már a listában. Egy fájl feltöltésekor a feltölthető fájl mérete is maximalizálva van. Így a felhasználó túl nagy méretű fájlokat nem képes feltölteni az oldalra. A feltölthető fájl maximális mérete 10Mb. A rendszer ellenőrzi a már meglévő listában található fájlokat illetve azok neveit, hogy a fájl feltöltésekor ugyanolyan nevű fájl ne kerüljön feltöltésre. A rendszer amennyiben mindent rendben talált, egy meglévő mappába áthelyezi a feltöltött dokumentumokat. A feltöltött dokumentumokat innentől kezdve lehetőség van letölteni, illetve a megadott adatait módosítani.

Fájl letöltése:



**Fejlesztési javaslat**

Ahogy azt már egyszer említettem a fájlfeltöltő oldalak ellégéé elterjedtek manapság, és éppen ezért rengeteg minden téma felé el lehetne terelni a weboldalt. Érdemes lenne egy szabadon regisztrálható lehetőséget is beépíteni a kódban, ha esetleg nem iskolai keretek között kellene használni. Ha szabadon bárki be regisztrálhat akkor sokkal nyíltabb rendszer lesz az oldalból, aminek a felhasználók mennyiségéhez kell igazodnia. Célszerű lenne egy mappa szerkezetet beépíteni, hogy a felhasználók, mint például a felhőtárhelyek esetében, képesek legyenek a saját elemeiket mappába rendezni és egy saját kis könyvtárat létrehozni. Erre akar a Kategóriák is alkalmasak lennének. ha iskolai keretek között használnánk akkor egy tanárhoz több kategóriát hozzá lehet rendelni, ami mondjuk az általa tanított tantárgyakat jelenti, és a hallgatók csak a saját tantárgyaikat látják, de több tanárral képesek kapcsolatban lépni. Ezen felül egy chat funkció is hasznosnak bizonyulna, akár a fájlok alatt lehetőség van kommenteket, hozzászólásokat írni az adott témához, illetve a felhasználók egymással is képesen lennének beszélgetni, valamilyen csapatmunka céljából.

**Források**

MVN: <https://hu.wikipedia.org/wiki/Modell-nézet-vezérlő>

<https://www.w3schools.com>