



EÖTVÖS LORÁND TUDOMÁNYEGYETEM

INFORMATIKAI KAR

PROGRAMOZÁSI NYELVEK ÉS FORDÍTÓPROGRAMOK

TANSZÉK

Beadandó kezelő rendszer megvalósítása

Témavezető:

Poór Artúr

egyetemi tanársegéd

Szerző:

Csiki Erik Gergely

programtervező informatikus BSc

Budapest, 2021

Az eredeti szakdolgozati / diplomamunka témabejelentő helye.

Tartalomjegyzék

1. Bevezetés	3
2. Felhasználói dokumentáció	4
2.1. Bejelentkezés és nyelvválasztás	4
2.1.1. Bejelentkezés	5
2.1.2. Nyelvválasztás	6
2.2. Szerepkörök	6
2.2.1. Rendszergazda	6
2.2.2. Tárgyfelelős	10
2.2.3. Gyakorlatvezető	12
2.2.4. Hallgató	16
2.2.5. Mindenki számára elérhető oldalak	18
3. Fejlesztői dokumentáció	21
3.1. Keretrendszer és az alkalmazás felépítése	21
3.1.1. Keretrendszer	21
3.1.2. Az alkalmazás felépítése	21
3.2. Logolás	23
3.3. Adatbázis	24
3.3.1. Technológiák	24
3.3.2. Az adatbázis táblái	25
3.4. Model réteg	27
3.5. Vezérlő réteg	27
3.6. Nézet réteg	27
4. Tesztelés	28

5. Köszönetnyilvánítás	29
Irodalomjegyzék	30

1. fejezet

Bevezetés

A szakdolgozatom témája egy beadandó kezelő rendszer megvalósítása webes alkalmazásként. Az elkészült rendszer az Assignment Supervisor System nevet kapta (röviden ASS), a továbbiakban így hivatkozom rá.

Az alkalmazás célja, hogy segítse az egyetemi munkát, mind a hallgatók, mind az oktatók részére. Lehetőségünk van tantárgyak létrehozására, amelyekhez létrehozhatunk csoportokat. A hallgatók ezekbe a csoportba tudnak jelentkezni, amit a gyakorlatvezető hagyhat jóvá. A hallgatók számára lehetőségünk van határidővel ellátott feladatokat kiírni. A feladatkiírásánál támogatott a `Markdown` és a `LATEX` kifejezések. A beküldött megoldásokat a gyakorlatvezetők értékelhetik. Az alkalmazás támogatja a külföldi hallgatók munkáját, számukra elérhető az alkalmazás angol nyelvű változatban.

2. fejezet

Felhasználói dokumentáció

2.1. Bejelentkezés és nyelvválasztás

Az oldalra érkezve a kezdőoldalt láthatjuk, ahol egy üdvözlő üzenet fogad minket. Majd lehetőségünk nyílik bejelentkezni a rendszerbe, vagy a rendszer által támogatott lokalizációt tudjuk kiválasztani (2.1 ábra). ¹

Köszöntjük!

Az oldalt az INF-es (Pandorás) azonosítóval tudjuk használni.

Bejelentkezés

Magyar

English

2.1. ábra. Főoldal

¹Erre majd még a bejelentkezés után is lehetőségünk nyílik lásd később, 2.2.5: „Mindenki számára elérhető oldalak”

2.1.1. Bejelentkezés

A rendszerbe bejelentkezni az INF-es felhasználónkkal tudunk. Ha a bejelentkezés sikertelen volt, azt a rendszer hibaüzenetekkel jelzi a számunkra (2.2 ábra). Amennyiben a bejelentkezés sikeres volt, a *szerepkörnek* megfelelő kezdőoldalon találjuk magunkat (2.3 ábra).

Köszöntjük!

Az oldalt az INF-es (Pandorás) azonosítóval tudjuk használni.

A INF-es felhasználónév megadása kötelező!


A INF-es jelszó megadása kötelező!

Bejelentkezés

Magyar

English




2.2. ábra. Bejelentkezési hiba

 Főoldal

 Teszt Elek / student Csoport jelentkezés Kijelentkezés

Feladatok

▲ Gyak #1

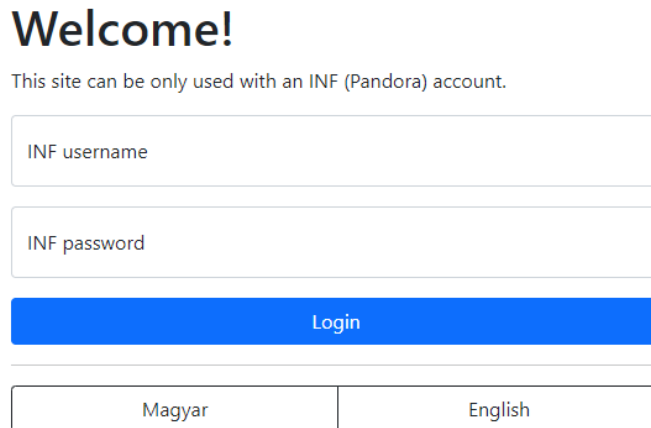
Feladat	Határidő	Értékelés	
1. Házi feladat	2021.05.30. 01:00		 Feladat megtekintése
Elso	2021.05.07. 00:00		 Feladat megtekintése
2. Házi feladat	2021.05.30. 01:00		 Feladat megtekintése

◀ 1 ▶
1-1 a(z) 1 elemből

2.3. ábra. Sikeres bejelentkezés

2.1.2. Nyelvválasztás

A rendszer kilistázza a támogatott lokalizációkat (jelenleg magyar és angol). Alapértelmezett beállítás a magyar. Ezt felültudjuk írni, ha valamelyik gombra rákattintunk. (2.4 ábra)



The screenshot shows a login interface. At the top, it says 'Welcome!' followed by the text 'This site can be only used with an INF (Pandora) account.' Below this are two input fields: 'INF username' and 'INF password'. A blue 'Login' button is positioned below the password field. At the bottom, there is a horizontal bar with two buttons: 'Magyar' and 'English'.

2.4. ábra. Nyelvváltás

2.2. Szerepkörök

A felhasználók négy csoportba tartozhatnak:

- Rendszergazda
- Tárgyfelelős
- Gyakorlatvezető
- Hallgató

Egy felhasználó tartozhat több szerepkörbe. Ha egy felhasználó több szerepkörbe is tartozik, akkor a felület menüsorán megjelenik egy "Szerepkör váltás" lenyitható menü, ahol a felhasználóhoz rendelt szerepköröket találjuk, a kiválasztott linkre kattintva a csoporthoz tartozó kezdőoldalra navigáljuk magunkat. A felhasználóhoz a szerepköröket a felhasználó létrehozásakor is megadhatjuk, valamint a létrehozást követően tudjuk módosítani.

2.2.1. Rendszergazda

A rendszergazda a következő funkciókat érheti el:

- Tantárgy létrehozása, módosítása, törlése, tárgyi információk megtekintése

- Felhasználó létrehozása, módosítása, a felhasználók adatainak a megtekintése
- Ha rendszergazdaként jelentkezünk be az alábbi két táblázat fogad minket a kezdőoldalon (2.5 ábra).

Főoldal	Teszt Elek / admin	Tárgy létrehozás	Felhasználó létrehozás	Kijelentkezés
---------	--------------------	------------------	------------------------	---------------

Tantárgyak információi

Keresés...		
Tantárgy neve	Tárgyfelelős(ök)	
Funkcionális programozás EA+GY	Bozó István (bozo_i), Poór Artúr (poora)	Szerkesztés Törölés
Funkcionális nyelvek EA+GY	Poór Artúr (poora)	Szerkesztés Törölés

1-2 a(z) 2 elemből

(a) Tantárgyak táblázata

Felhasználók listája

Keresés...				
Név	Neptun kód	Email	Szerepkörök	
Teszt Elek	student	student@inf.elte.hu	Hallgató	Szerkesztés
Poór Artúr	poora	poora@inf.elte.hu	Tárgyfelelős, Gyakorlatvezető	Szerkesztés
Csikó Erik Gergely	csikie	csikie@inf.elte.hu	Gyakorlatvezető	Szerkesztés
Bajkó János Róbert	bajkoj	bajkoj@inf.elte.hu	Gyakorlatvezető	Szerkesztés
Bozó István	bozo_i	bozo_i@inf.elte.hu	Tárgyfelelős	Szerkesztés

1-5 a(z) 5 elemből

(b) Felhasználók táblázata

2.5. ábra. Rendszergazdai szerepkör kezdőoldala

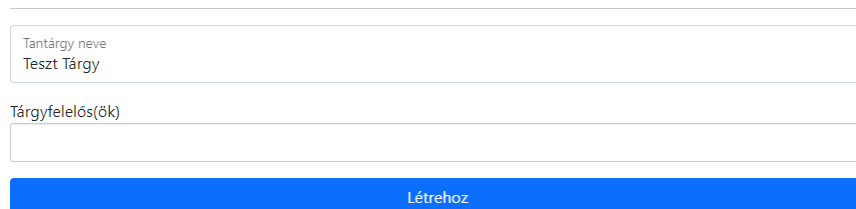
Az első táblázatban a rendszerben létrehozott tantárgyak és a hozzájuk tartozó információk olvashatóak le. A táblázatban az egyes tantárgyakhoz tartozó adatok módosíthatóak, illetve az egész tárgyat lehet törölni. A módosítás során validálásra

kerül, hogy a módosított név létezik-e már a rendszerben, ha igen, akkor ezt a rendszer jelzi számunkra. A második táblázatban a rendszerben létrehozott felhasználókat és a hozzájuk tartozó információkat láthatjuk. A rendszergazda a felhasználók adatait és szerepköreit tudja módosítani.

Tantárgy létrehozása

A „Tárgy létrehozás” linkre kattintva az alkalmazás átnavigál minket egy űrlapra, ahol az új tantárgy szükséges adatait tudjuk kitölteni (2.6(a) ábra). Ha az adatok validálása és feldolgozása sikeres, akkor visszanavigálódunk a kezdőoldalra. Az esetleges validálási hibákat a rendszer jelzi számunkra (2.6(b) ábra).

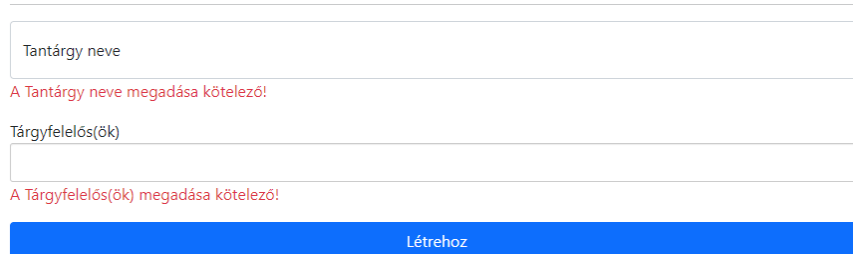
Tantárgy létrehozás



The screenshot shows a web form titled "Tantárgy létrehozás". It has two input fields: "Tantárgy neve" with the value "Teszt Tárgy" and "Tárgyfelelős(ök)". Below the fields is a blue button labeled "Létrehoz".

(a) Űrlap

Tantárgy létrehozás



The screenshot shows the same form as in (a), but with validation errors. The "Tantárgy neve" field has a red error message: "A Tantárgy neve megadása kötelező!". The "Tárgyfelelős(ök)" field has a red error message: "A Tárgyfelelős(ök) megadása kötelező!". The blue "Létrehoz" button is still visible at the bottom.

(b) Adatok validálása

2.6. ábra. Tantárgy létrehozás

Felhasználó létrehozása

Felhasználót létrehozni a „Felhasználó létrehozás” linkre kattintva tudjuk megtenni, ami továbbnavigál minket egy űrlapra, ahol az új felhasználónak az adatait tudjuk megadni (2.7(a) ábra). Ha az adatok validálása sikeres, akkor a felhasználó

elkészült és visszavigázódunk a kezdőoldalra, ha nem volt sikeres, akkor a rendszer ezt hibaüzenetekkel jelzi nekünk (2.7(b) ábra).

Felhasználó létrehozás

The screenshot shows a web form titled "Felhasználó létrehozás". It contains six input fields stacked vertically: "Neptun kód", "Név", "Email", "Jelszó", "Jelszó megerősítése", and "Szerepkör(ök)". Below these fields is a blue button labeled "Létrehoz".

(a) Űrlap

Felhasználó létrehozás

The screenshot shows the same "Felhasználó létrehozás" form as in (a), but with red error messages below each input field: "A Neptun kód megadása kötelező!", "A Név megadása kötelező!", "A Email megadása kötelező!", "A Jelszó megadása kötelező!", "A Jelszó megerősítése megadása kötelező!", and "A Szerepkör(ök) megadása kötelező!". The blue "Létrehoz" button is still at the bottom.

(b) Adatok validálása

2.7. ábra. Felhasználó létrehozás

2.2.2. Tárgyfelelős

Ha tárgyfelelősként jelentkezünk be a rendszerbe, akkor az alábbi kezdőoldal fogad minket (2.8 ábra). A kezdőoldalon egy táblázat található, amiben látjuk azokat a tantárgyakat, és a tantárgyakhoz tartozó csoportokat, amelyeknek a felelősei vagyunk. A tárgyfelelős a következő funkciókat használhatja:

- Csoport létrehozása egy tantárgyhoz
- Csoport módosítása

Tantárgyak

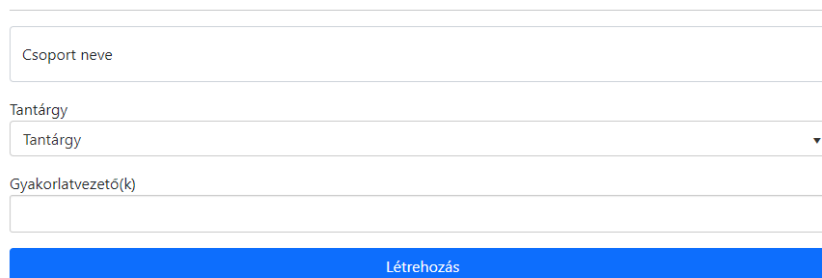
Keresés...								
Tantárgy								
<div> <div>Funkcionális programozás EA+GY</div> <table> <tr> <th>Csoport</th> <th>Gyakorlatvezető(k)</th> <th></th> </tr> <tr> <td>Gyak #1</td> <td>Bajkó János Róbert (bajkoj), Csiki Erik Gergely (csikie)</td> <td>Szerkesztés</td> </tr> </table> </div>			Csoport	Gyakorlatvezető(k)		Gyak #1	Bajkó János Róbert (bajkoj), Csiki Erik Gergely (csikie)	Szerkesztés
Csoport	Gyakorlatvezető(k)							
Gyak #1	Bajkó János Róbert (bajkoj), Csiki Erik Gergely (csikie)	Szerkesztés						
<div> <div>Funkcionális nyelvek EA+GY</div> <table> <tr> <th>Csoport</th> <th>Gyakorlatvezető(k)</th> <th></th> </tr> <tr> <td>Gyak #2</td> <td>Csiki Erik Gergely (csikie)</td> <td>Szerkesztés</td> </tr> </table> </div>			Csoport	Gyakorlatvezető(k)		Gyak #2	Csiki Erik Gergely (csikie)	Szerkesztés
Csoport	Gyakorlatvezető(k)							
Gyak #2	Csiki Erik Gergely (csikie)	Szerkesztés						

2.8. ábra. Tárgyfelelős kezdőoldala

Csoport létrehozása

A menüsoron a „Csoport létrehozás” linkre kattintva a rendszer átirányít minket egy űrlapra, ahol létre tudunk hozni egy csoportot (2.9 ábra). Az adatokat a rendszer validálja, és az esetleges hibákat jelzi számunkra.

Csoport létrehozás



Csoport neve

Tantárgy

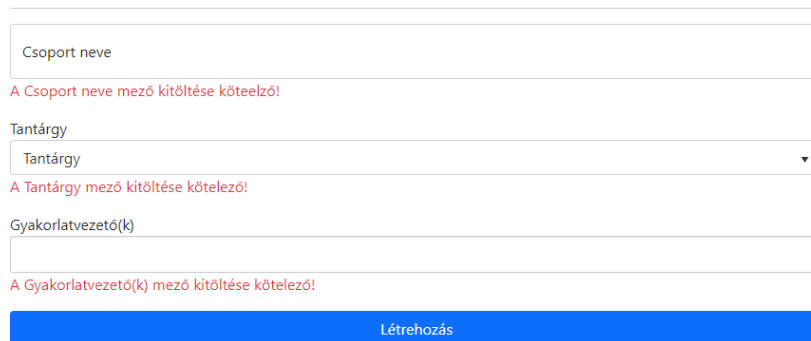
Tantárgy

Gyakorlatvezető(k)

Létrehozás

(a) Űrlap

Csoport létrehozás



Csoport neve

A Csoport neve mező kitöltése kötelező!

Tantárgy

Tantárgy

A Tantárgy mező kitöltése kötelező!

Gyakorlatvezető(k)

A Gyakorlatvezető(k) mező kitöltése kötelező!

Létrehozás

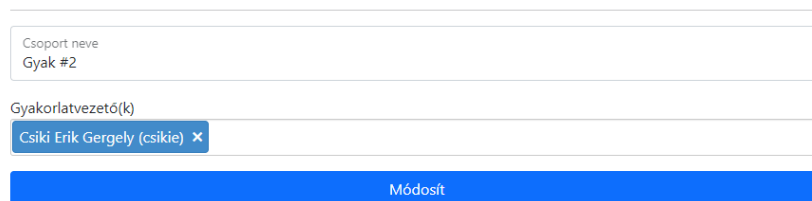
(b) Adatok validálása

2.9. ábra. Csoport létrehozás

Csoport módosítása

Csoportokat szerkeszteni a kezdőoldalon található táblázat segítségével tudunk. A táblázatban lenyitható minden kilistázott tantárgy. Itt találjuk a tantárgyakhoz már létrehozott csoportokat. Minden tantárgy mellett találunk egy „Szerkesztés” gombot, melyre kattintva elérhetővé válik a csoport szerkesztése. Az adatok validálásra kerülnek, az esetleges hibákat a rendszer jelzi számunkra (2.10 ábra).

Csoport szerkesztése



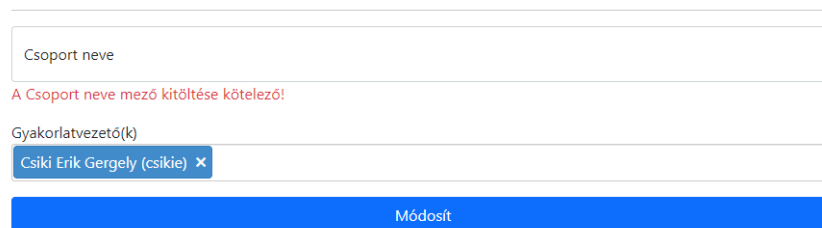
Csoport neve
Gyak #2

Gyakorlatvezető(k)
Csiki Erik Gergely (csikie) ✕

Módosít

(a) Űrlap

Csoport szerkesztése



Csoport neve

A Csoport neve mező kitöltése kötelező!

Gyakorlatvezető(k)
Csiki Erik Gergely (csikie) ✕

Módosít

(b) Adatok validálása

2.10. ábra. Csoport módosítása

2.2.3. Gyakorlatvezető

Gyakorlatvezetőként bejelentkezve a rendszerbe a 2.11 ábrán látható kezdőoldal fogad minket. Az oldalon az „Értékelendő beadandók” cím alatt, a hozzánk rendelt csoportok hallgatóit láthatjuk egy-egy táblázatban, ahol láthatjuk, hogy egy hallgató az adott feladatra adott-e be megoldást. A „Hallgatói várólista” cím alatt szintén egy táblázatot találunk (2.11(b) ábra), ahol tantárgyanként csoportosítva a következő információkat olvashatjuk le:

- Hallgató neve
- Hallgató neptun kódja
- Csoport neve

Értékelendő beadandók

Funkcionális programozás EA+GY - Gyak #1

Hallgató neve

Teszt Elek (student)

1. Házi feladat

Funkcionális nyelvek EA+GY - Gyak #2

Hallgató neve

1. Házi feladat

(a) Értékelendő beadandók

Hallgatói várólista

Keresés...			
Hallgató neve	Neptun kód	Csoport	
<div>Funkcionális nyelvek EA+GY</div>			
Teszt Elek	student	Gyak #2	<div>Elutasít</div> <div>✓ Elfogad</div>
<div>1-1 a(z) 1 elemből</div>			

(b) Hallgatói várólista

2.11. ábra. Gyakorlatvezető kezdőoldala

A gyakorlatvezető az alábbi funkciókat éri el:

- Feladat kiírása
- Jelentkezések bírálata
- Beadott munka értékelése

Feladat kiírása

Feladatot kiírni a „Feladat létrehozása” menüpont alatt tudunk, ahol egy űrlapot találunk (2.12). Ahhoz hogy létrehozzunk egy feladatot, a következő információkat szükséges megadnunk:

- A feladat neve
- Kezdés és befejezés dátuma
- Feladat leírása

- Mely csoporthoz legyen létrehozva²

Feladat létrehozása

Feladat neve

Kezdés dátuma
2021.05.07 00:00

Befejezés dátuma
2021.05.07 00:00

Leírás

Ennek a szövegnek markdown formátumban kell lennie. Íme egy rövid áttekintés:

- Ez egy lista egyik eleme, *dólt betűvel*.
- Ez pedig egy másik elem, **félkövérrel**. Ügyeljünk arra, hogy a szöveg többi része igazítva maradjon.

Néha előre megformázott szöveget akarunk írni:

```
~~~~~
előre megformázott
szöveg
~~~~~
```

Habár `szavakat` bármikor írhatunk a backtick (```) szimbólum segítségével.

Csoport(ok)

Létrehozás

Előnézet

Törlés

2.12. ábra. Feladat létrehozás

A rendszer támogatja, hogy a feladatnak leírása ne csak egyszerű szöveg legyen. A szövegdobozban megadhatunk **Markdown** és **L^AT_EX** kifejezéseket is. Mielőtt létrehoznánk a feladatot, meg tudjuk tekinteni, hogy a hallgató milyen formában fogja látni a kiírva a feladatot. Így le tudjuk ellenőrizni kényelmesen a feladat leírását, valamint azt is tudjuk ellenőrizni, hogy a **Markdown** és **L^AT_EX** kifejezéseinket helyesen írtuk-e meg. Ha végeztünk, a „Létrehozás” gombbal tudjuk elküldeni a rendszernek az adatokat. Az adatokat a rendszer leellenőrzi, az esetleges hibákat jelzi számunkra (2.13).

²A lenyíló kiválasztó menüben lehetőségünk van több csoportot is kiválasztani

Feladat létrehozása

Feladat neve

A Feladat neve megadása kötelező!

Kezdés dátuma

2021.05.07 00:00

A Kezdés dátuma megadása kötelező!

Befejezés dátuma

2021.05.07 00:00

A Befejezés dátuma megadása kötelező!

Leírás

A Leírás megadása kötelező!

Csoport(ok)

A Csoport(ok) megadása kötelező!

2.13. ábra. Adatok validálása

Jelentkezések bírálata

Egy hallgatónak a csoportba való jelentkezését a „Hallgatói várólista” táblázatában tudjuk megtenni (2.11(b) ábra), az utolsó oszlopban lévő gombok segítségével. Miután elvégeztük a bírálatot, a táblázatból törlődik a jelentkezett hallgató, és az oldal frissítése után, a megfelelő táblázatban látjuk, hogy a hallgatót a rendszer felvette a jelentkezett csoportba.

Beadott munka értékelése

Egy feladatra beadott megoldás értékeléséhez a kívánt feladat oszlopában kattintsunk a *szürke négyzetre*. Ilyenkor a rendszer átirányít minket az értékelő felületre (2.14). A rendszer a felületre a „Megoldás(ok)” alatt felsorolja a hallgatónak az összes beadott megoldását a beküldés ideje szerint csökkenően rendezve. Elég egy megoldást értékelnünk, de értékelhetjük az összeset is. Viszont a rendszer a legutolsó értékelést veszi számításba. Ezt fogja a hallgató is látni. A beadott megoldások között a kívánt sorra kattintva tudjuk kiválasztani, hogy melyik megoldást szeretnénk változtatni. Ilyenkor a „Beadott megoldás” alatt látjuk, mi a hallgatónak a beadott megoldása. Az értékelésünket az oldal alján található űrlapon tudjuk megtenni. Az

értékelés után a rendszer a kezdőoldalra navigál minket. Az értékelt feladatnál a *szürke négyzet egy körben lévő pipára* cserélődik.

Feladat értékelése

Információk

Hallgató neve	Teszt Elek (student)
Feladat neve	2. Házi feladat
Kezdés dátuma	2021. 05. 07. 0:00:00
Befejezés dátuma	2021. 05. 30. 1:00:00

Beadott megoldás

Ez az én megoldásom.

Megoldás(ok)

Beadási idő: 2021. 05. 07. 16:30:28

Értékelés

Mentés

2.14. ábra. Feladat értékelése

2.2.4. Hallgató

Hallgatóként bejelentkezve a 2.15 ábrán látható kezdőoldal fogad minket. A „Feladatok” cím alatti táblázatban a hallgató számára listázásra kerül az összes olyan csoportja, ahova elfogadták a jelentkezését. A táblázatban csoportokra lebontva jelennek meg a hallgató számára a kiírt feladatok. A táblázatban egy feladatról a következő információkat láthatjuk: neve, határideje, kapott értékelés.

Feladatok

Keresés...

Csoport

Gyak #1

Feladat	Határidő	Értékelés	
1. Házi feladat	2021.05.30. 01:00		Feladat megtekintése
Elso	2021.05.07. 00:00		Feladat megtekintése
2. Házi feladat	2021.05.30. 01:00		Feladat megtekintése

1-1 a(z) 1 elemből

2.15. ábra. Hallgató kezdőoldala

A hallgató az alábbi funkciókat használhatja:

- Csoportba jelentkezés
- Megoldás beadása

Csoportba jelentkezés

Csoportba jelentkezni a „Csoport jelentkezés” menüpontra kattintva tudunk. A rendszer egy űrlapot biztosít számunkra (2.16 ábra), ahol listázásra kerülnek a rendszerben található csoportok, amelyekre még nem jelentkezünk. A rendszer lehetőséget biztosít számunkra, hogy akár egyszerre több csoportra is leadjuk a jelentkezésünket. Jelentkezésünket a „Jelentkezés” gombbal tudjuk továbbítani a rendszer számára. Az űrlap validálásra kerül, hogy üresen ne tudjuk beküldeni azt. Sikeres jelentkezés esetén a rendszer a kezdőoldalra navigál minket.

Csoport jelentkezés

Csoport

Csoport(ok)

Jelentkezés

2.16. ábra. Csoportba jelentkezése

Megoldás beadása

Megoldás beküldéséhez válasszuk ki a kívánt feladatot, amire megoldást szeretnénk beküldeni, majd kattintsunk a „Feladat megtekintése” gombra. Ekkor a rendszer egy új ablakban megnyitja a feladatot (2.17 ábra). Az oldalon a következő információkat látjuk:

- Határidő visszaszámláló
- Feladat neve, leírása
- Beküldött megoldások
- Űrlap a megoldás beküldéséhez

Határidő: 22 nap 8:12:53

2. Házi feladat

Ennek a szövegnek markdown formátumban kell lennie. Íme egy rövid áttekintés:

- Ez egy lista egyik eleme, *dolt betuvel*.
- Ez pedig egy másik elem, **félkövérrel**. Ügyeljünk arra, hogy a szöveg többi része igazítva maradjon.

Néha előre megformázott szöveget akarunk írni:

```
elore megformázott
szöveg
```

Habár *szavakat* bármikor írhatunk a backtick (`) szimbólum segítségével.

Bizonyosan kódrészletet is szeretnénk mutatni:

```
main :: IO ()
main = putStrLn "I'm highlighted"
```

Számos nyelv támogatott, többek között *ada, agda, c, cpp, erlang, java, lex, python* és *yacc*.

Képletek és egyenletek pont úgy működnek, mint *LaTeX*-ben. Használható szövegközi és kiemelt mód a matematikai tartalom szép szedésére. A szövegközi módnál, például $\sum_{j=0}^n (a + bj)$ esetében, két dollár szimbólum közé írjuk a képletet vagy egyenletet. A nyitó dollár szimbólum után és a záró előtt nincs szóköz. A kiemelt mód nagyobb hangsúlyt ad:

$$\sum_{j=0}^n (a + bj) = a + (a + b) + \dots + (a + nb)$$

A támogatott *LaTeX* makrók [ezen az oldalon](#) vannak felsorolva.

Emellett még linkek is [illeszthetők](#) a szövegbe.

Beküldési idő: 2021. 05. 07. 16:30:28

✓

2.17. ábra. Feladat megtekintése

2.2.5. Mindenki számára elérhető oldalak

A rendszerben jelenleg három olyan oldal található, amelyet minden szerepkörben elérhetünk. Az egyik oldal a felhasználónk adatainak megtekintésére szolgál. Ezt a funkciót a menüsoron a nevünkre kattintva tudjuk elérni. Ezen a felületen (2.18 ábra) a következőket tekinthetjük meg a „Személyes adatok” cím alatt: név, neptun kód, e-mail cím és a felhasználónkhoz rendelt szerepkörök. Ezen felületen továbbá be

tudjuk állítani, hogy a rendszer milyen lokalizációval működjön (magyar és angol). Ezt a megfelelő gombra kattintva tudjuk változtatni.



2.18. ábra. Profil oldal

A másik két oldal az esetleges nem várt hibákról tájékoztat minket. Ezen hibák két kategóriába oszthatók: jogosulatlan kérés a rendszer felé, egyéb nem várt hiba. Jogosulatlan kérés akkor lép fel, ha megpróbálunk a szerepkörünkhöz nem tartozó funkciót elérni a rendszerben. Például: csak hallgatói szerepkörrel rendelkező felhasználóval vagyunk bejelentkezve a rendszerbe és a webcím végén a „Student”-et lecseréljük „Admin”-ra. Ezzel olyan kérést indítunk a rendszernek, hogy navigáljon minket a rendszergazdai szerepkörhöz tartozó kezdőoldalra, amihez nincs jogosultságunk, ezért a rendszer megtadja a hozzáférést a kért oldalhoz. Ilyenkor a rendszer az alábbi 2.19 ábrán látható oldalra navigál minket, ahonnan lehetőségünk van visszatérni a szerepkörünkhöz tartozó kezdőoldalra.

Hozzáférés megtagadva.

A viharba, nincs hozzáféréseid ehhez a kéréshez! 😞

[Vissza a kezdőoldalra](#)

2.19. ábra. Jogosulatlan kérés

Egyéb nem várt hiba lehet például, hogy a rendszer nem tud csatlakozni a hozzá tartozó adatbázishoz. Ilyenkor az alábbi 2.20 ábrán látható oldalra navigál minket.

Error.

Hiba lépett fel a folyamat során! Kérlek beszélj a rendszergazdával!

2.20. ábra. Egyéb nem várt hiba

3. fejezet

Fejlesztői dokumentáció

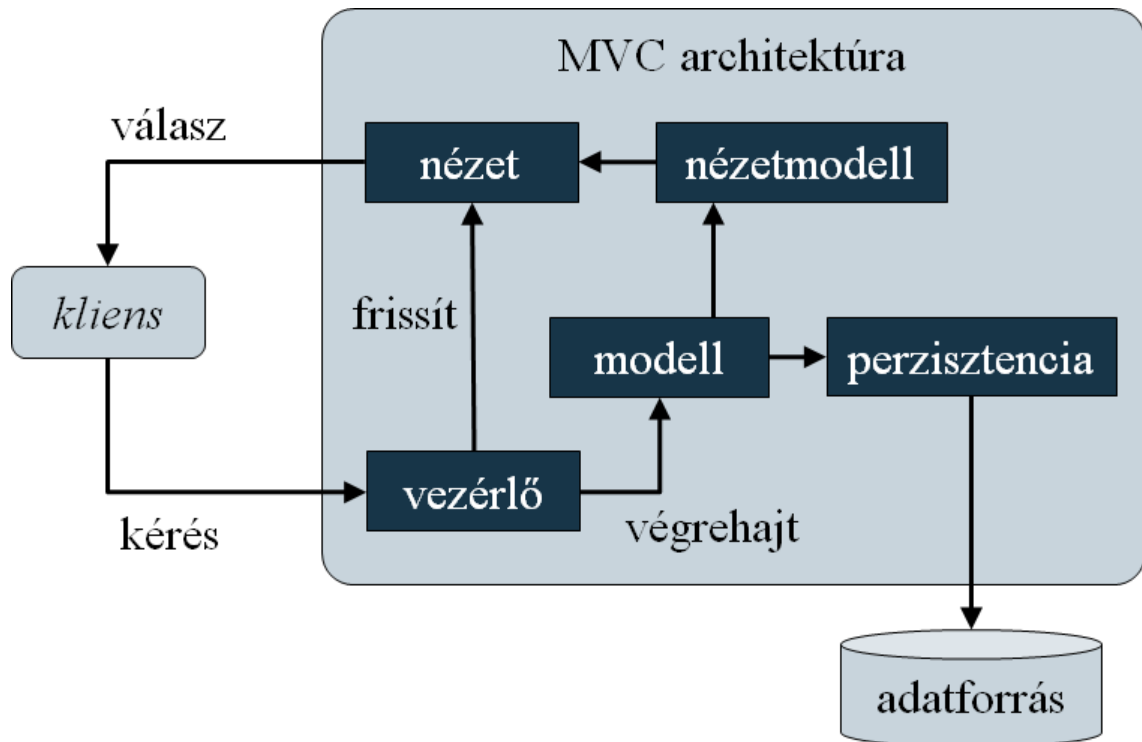
3.1. Keretrendszer és az alkalmazás felépítése

3.1.1. Keretrendszer

Az alkalmazás ASP.NET core 3.1 keretrendszerben készült [1], ami egy nyílt forráskódú, webes alkalmazások készítésére szolgáló programkönyvtár, melyet a *Microsoft* fejleszt. A keretrendszer lehetővé teszi, hogy az alkalmazás több platformon is tudjon futni (*Linux*, *macOS* és *Windows*). Továbbá az alkalmazás rendelkezik autentikációval és autorizációval.

3.1.2. Az alkalmazás felépítése

Az alkalmazás az *MVC* architektúrára épül (3.1 ábra)[2]. Tehát három rétegre bontható a felépítése, Modell-Nézet-Vezérlő. A Modell réteg tartalmazza az üzleti logikát, amely az adatokat kezeli és kapcsolatban van az adatbázissal. A nézet réteg (angolul *View*) felelős a megjelenítésért. A vezérlő réteg (angolul *Controller*) fogadja a kliens a kéréseit és válaszol a kérésekre. Az *MVC* architektúra fő előnye, hogy jól elkülöníthetők a rétegek, így a nézet független marad a modelltől. Ezáltal, ha szükséges könnyedén le tudjuk cserélni az egész alkalmazás nézetét, vagy fordítva újra implementálhatjuk a modell réteg működését, anélkül hogy ez a nézeten bármi gondot okozna.



3.1. ábra. A Model-View-Controller architektúra

Az alkalmazásban a könnyebb és egyszerűbb fejleszthetőség miatt, a *Model* réteget több komponensre bontjuk. Így az alábbi komponensekből áll össze a *Model* réteg:

```

ASS
├── ASS.BLL/
│   ├── Interfaces/
│   └── Services/
├── ASS.DAL/
│   ├── Models/
│   ├── ASSContext.cs
│   └── DbInitializer.cs
├── ASS.WEB/
│   ├── Models/
│   │   ├── DTOs/
│   │   └── ViewModels
  
```


ASS.BLL: az üzleti logikai réteget megvalósító komponens (angolul *Business Logic Layer*).

ASS.DAL: az adatelérési réteget megvalósító komponens (angolul *Data Access Layer*).

ASS.WEB.Models: ebben a komponensben tároljuk az adatok bevitelére és az adatok megjelenítésére szolgáló osztályokat.

ASSContext.cs: Az adatbázist leíró osztály.

DbInitializer.cs: Az adatbázist létrehozó statikus osztály.

3.2. Logolás

Az alkalmazás fájl szintű logolást tartalmaz, amit a *Serilog.Extensions.Logging.File* nyílt forráskódú programkönyvtár használatával valósítjuk meg [3]. Az alkalmazás automatikusan logolja a futás közbeni eseményeket és az esetleges kivételeket. Természetesen támogatott a saját logüzenetek létrehozása is. A logolás beállításait az *appsettings.json* (3.2 ábra) fájlban tudjuk személyre szabni. Az alábbi négy értéket szabályuk személyre az alkalmazáshoz:

- **PathFormat:** itt tudjuk megadni az alkalmazás logfájljainak a mentési helyét, és egy sablont a fájlok nevére. A *{Date}* paraméter helyére az aktuális dátum kerül beillesztésre (pl.: 20210513). Ha az elérési útban található mappa nem létezik azt a programkönyvtár automatikusan létrehozza a számunkra.
- **OutputTemplate:** itt adható meg a log üzenetek sablonja, hogy hogyan néznek ki a logolt üzenetek³. Az alkalmazás a következő sablont használja a logokra: *[Időbélyeg] - [Log szintje] - [Üzenet] Új sor [Kivétel (ha van)]*.
- **LogLevel:** itt állíthatjuk be, hogy milyen minimum szintű események kerüljenek logolásra [4]. A jelenlegi beállítással az alkalmazás minden legalább *Information* szinttel rendelkező eseményt logol.

³Az alábbi linken részletes leírást olvashatunk az *OutputTemplate*-ben használható paramétekről.

```
1 ...
2 "Logging": {
3   "PathFormat": "../Logs/log-{Date}.log",
4   "OutputTemplate": "[{Timestamp:yyyy.MM.dd HH:mm:ss}] - [{Level:u
      ↪   }] - {Message}{NewLine}{Exception}",
5   "LogLevel": {
6     "Default": "Debug",
7     "Microsoft": "Information"
8   }
9 },
10 ...
```

3.1. forráskód. Logolás beállításai

3.3. Adatbázis

3.3.1. Technológiák

Az alkalmazáshoz szükséges telepítünk egy *MySQL Community Server*-re, ajánlott a 8.0.25-ös verzió.⁴ Az autentikáció és autorizáció megvalósításához a Microsoft által készített *Microsoft.AspNetCore.Identity.EntityFrameworkCore* nyílt forráskódú programkönyvtárat használja rendszer. A programkönyvtár tartalmaz meglévő adatbázis táblákat melyeknek a tartalma és működése elolvasható a Microsoft hivatalos honlapján [5]. A programkönyvtár gondoskodik a jelszavak biztonságos tárolásáról, melyet időfüggő szózással és a jelszó hashelésével valósít meg.

Az adatbázis *code first* módszerrel van megvalósítva, tehát nem az adatbázis szerveren *SQL* kódot futattva hozzuk létre az adatbázis táblákat, hanem modell osztályokkal definiáljuk az adatbázis táblákat [6]. Ezen modelleket az *ASS.DAL.Models* névtérben tároljuk.

Az adatelérést az *Entity Framework Core ORM*⁵ keretrendszer biztosítja [7]. Így az alkalmazás forráskódjában nincsenek beégetett *SQL* kódok. Ezek helyett a

⁴Az alkalmazás működik régebbi verzióval is. Viszont az alkalmazás nincs felkészítve az esetleges verziók közötti különbségekre.

⁵Jelentése: objektum-relációs leképezés, egy technika az adatok konvertálására nem kompatibilis típusos rendszerek és objektumorientált programozási nyelvek között.

*CRUD*⁶ műveleteket a *.NET* nyújtotta és az *Entity Framework Core* által is támogatott *LINQ* (Language Integrated Queries) metódus hívásokkal valósul meg [8]. Továbbá a keretrendszer védelmet biztosít az *SQL Injection* támadások ellen [9], ugyanis a műveletek a *C#* és *LINQ* metódusokból kerülnek előállításra paraméterezetten.

Az adatbázis elérését az alkalmazás konfigurációs fájljában (*appsettings.json*) tudjuk megadni illetve módosítani.

```
1 ...  
2 "ConnectionStrings": {  
3   "DefaultConnection": "server=localhost;database=ASS;uid=username;  
   ↪ password=fooBarraBoof "  
4 },  
5 ...
```

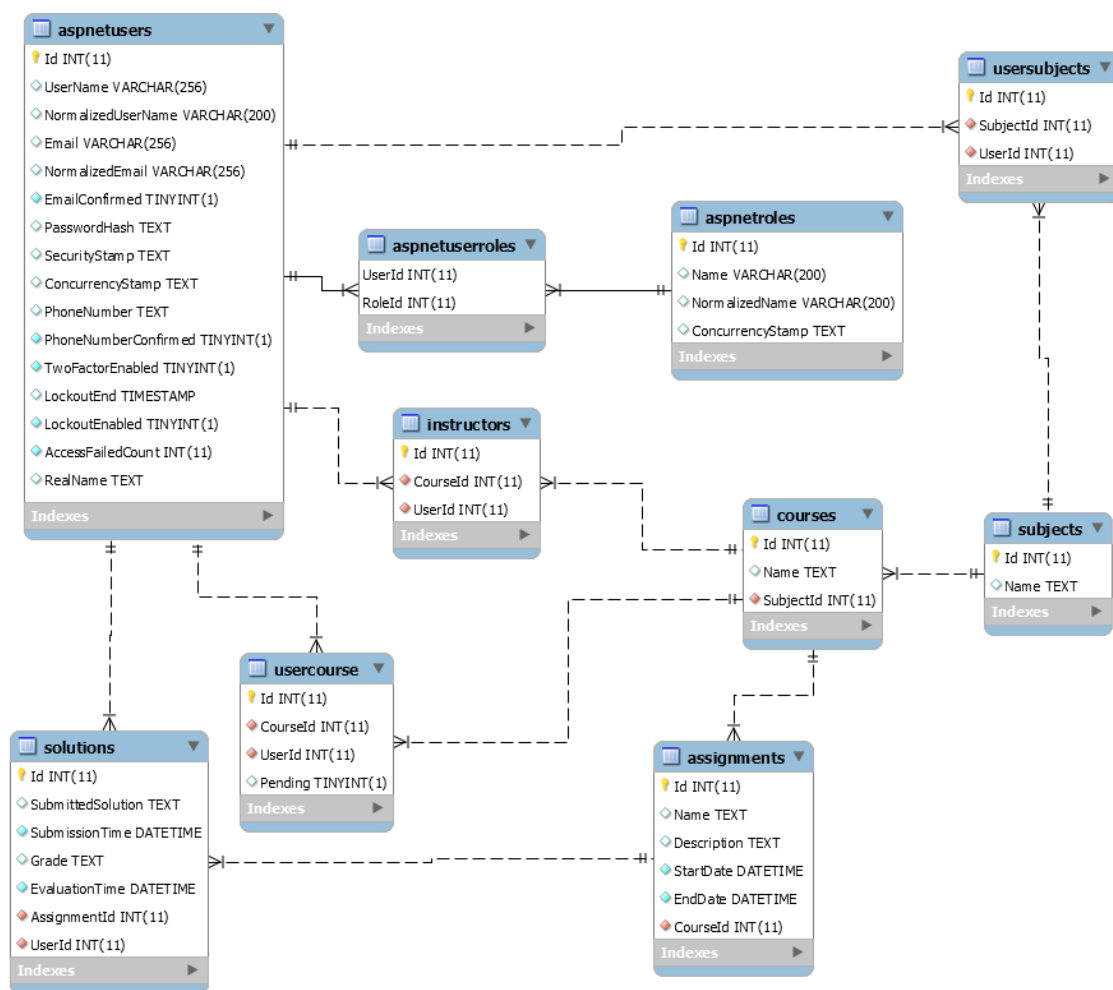
3.2. forráskód. Adatbázis elérése

3.3.2. Az adatbázis táblái

Az alkalmazás adatbázis diagramját a 3.2 ábrán tekinthetjük meg. A *Microsoft.AspNetCore.Identity.EntityFrameworkCore* keretrendszer által létrehozott táblákból csak azon a táblák és mezők kerülnek részletezésre, melyeket a rendszer aktívan használ. ⁷

⁶Create,Read,Update,Delete műveleteknek a rövidítése

⁷A táblák amik nem kerülnek részletezésre a Microsoft hivatalos honlapján meg lehet tekinteni.



3.2. ábra. Az adatbázis táblái

aspnetusers

A *Microsoft.AspNetCore.Identity.EntityFrameworkCore* programkönyvtár által automatikusan létrehozott tábla. A felhasználók adatait tárolja.

Mező neve	Típus	Leírás
Id	int	Elsődleges kulcs
UserName	string	Felhasználónév (neptun kód)
Email	string	Felhasználó e-mail címe
RealName	string	Felhasználó neve
PasswordHash	string	Felhasználó hashelt jelszava

3.1. táblázat. Adatbázis: felhasználók táblája

3.4. Model réteg

3.5. Vezérlő réteg

3.6. Nézet réteg

4. fejezet

Tesztelés

5. fejezet

Köszönetnyilvánítás

Irodalomjegyzék

- [1] Microsoft. *Introduction to ASP.NET Core*. URL: <https://docs.microsoft.com/en-us/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-3.1>.
- [2] Cserép Máté. *Webfejlesztés MVC architektúrában (ASP.NET Core)*. URL: https://mcserep.web.elte.hu/data/education/2019-2020-2_WAF/elte_waf_ea02.pdf.
- [3] Serilog. *Serilog.Extensions.Logging.File*. URL: <https://github.com/serilog/serilog-extensions-logging-file>.
- [4] Nicholas Blumhardt. *Logging “levels” in a structured world*. URL: <https://nblumhardt.com/2014/03/logging-levels-in-a-structured-world/>.
- [5] Microsoft. *Introduction to Identity on ASP.NET Core*. URL: <https://docs.microsoft.com/en-us/aspnet/core/security/authentication/identity?view=aspnetcore-3.1&tabs=visual-studio>.
- [6] Entity Framework Tutorial. *What is Code-First?* URL: <https://www.entityframeworktutorial.net/code-first/what-is-code-first.aspx>.
- [7] Microsoft. *Entity Framework Core*. URL: <https://docs.microsoft.com/en-us/ef/core/>.
- [8] Microsoft. *Language Integrated Query (LINQ)*. URL: <https://docs.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/linq/>.
- [9] OWASP. *SQL Injection*. URL: https://owasp.org/www-community/attacks/SQL_Injection.