

Pixel Oktatási Kft.
Szoftverfejlesztő képzés
54 213 05

SZAKDOLGOZAT

TeamAd Csoportadminisztrációs webalkalmazás



Szabó Péter
Témavezető

Süle Szabolcs Péter
Szerző

Budapest, 2021.

Tartalomjegyzék

Tartalomjegyzék	2
Felhasználói dokumentáció	3
A program általános specifikációja	3
Rendszerkövetelmények.....	4
A program telepítése	4
A program használatának részletes leírása	4
Fejlesztői dokumentáció.....	15
Témaválasztás indoklása	15
Alkalmazott fejlesztői eszközök.....	16
Adatmodell leírása.....	17
Szerveroldali rendszerkövetelmények.....	22
Webalkalmazás beállítása szerveroldalon	23
Folyamatok algoritmusai és forráskód-részletei.....	24
Tesztelési dokumentáció	27
Továbbfejlesztési lehetőségek.....	30
Felhasznált irodalmak és források	31
Ábra- és képjegyzék	32
Képernyőképek.....	32
Ábrák	32
Forráskódok.....	32

Felhasználói dokumentáció

A program általános specifikációja

A TeamAd csoportadminisztrációs webalkalmazás olyan személyek számára készült, akik elsősorban egyénileg végzik tevékenységüket, mely egyszerre több személy számára történő szolgáltatásnyújtásból áll. A rendszer megkönnyíti például oktatók, előadók, tanárok, edzők és egyéb csoport-, illetve kurzusvezetők munkájukból eredő adminisztrációs és kommunikációs feladatainak ellátását.

A webalkalmazás lehetőséget biztosít a felhasználó által szervezett és tartott kurzusok és tanfolyamok, illetve a résztvevők adatainak rögzítésére. A kurzusok kapcsán szabadon beállítható annak bármilyen heti időbeosztás, kezdődátum és alkalomszám. Ezen adatok alapján a rendszer kiszámolja az egyes alkalmak időpontjait külön-külön. Ha vannak beállítva szünetek, illetve ünnepnapok, akkor az említett ütemezés ezeket automatikusan figyelembe fogja venni. Ha később kerül rögzítésre egy szünet-időszak, akkor a már ütemezett kurzusok is újraütemezhetők. A könnyebb átláthatóság és kezelhetőség érdekében az alkalmazás a heti beosztást – a kurzusok és a szünetek időpontjaiból – interaktív grafikus felületen is megjeleníti, ahol egyértelműen látható, hogy melyik napon, milyen időpontban, mely események várhatók.

A kurzusok és résztvevők adatainak tárolása és rendszerezése mellett az alkalmazás a kommunikációt is támogatja. Szükség esetén könnyedén küldhetünk e-mail üzeneteket egy személynek, vagy akár egy csoport összes résztvevőjének. Amennyiben egy konkrét ügyfél számára küldenénk üzenetet, a résztvevők valós idejű, részletes és gyors keresőjével pillanatok alatt megtalálhatjuk a címzettet.

Az alkalmazás használata közben az adatok biztonsága miatt sem kell aggódnia a felhasználónak, ugyanis az elvárt, korszerű felhasználókezelés is biztosítva van. Csak a helyes felhasználónévvel és jelszóval lehet belépni az adminisztrációs felületre. Természetesen a jelszó megváltoztatható a jelenlegi jelszó ismeretében. Szükség esetén pedig – például asszisztensek, segítők számára – több felhasználó is rögzíthető.

Az alkalmazás funkcióinak tervezésekor fő szempont volt az optimális és gyors működés, illetve a könnyen kezelhető és interaktív megjelenés. Általánosságban az ügyfél bármit is szeretne elvégezni, azt könnyen és gyorsan teheti meg a rendszer bármelyik felületén.

A webalkalmazás a modern kor követelményeinek megfelelően az asztali számítógépek mellett használható táblagépeken és okostelefonokon egyaránt. A felület mindig igazodik az adott képernyő méreteihez és a tartalmakat optimálisan helyezi el.

Rendszerkövetelmények

A webalkalmazás megjelenítése bármelyik böngészőben lehetséges. Felhasználói szempontból tehát a használatához egy okostelefon, táblagép vagy asztali számítógép szükséges, stabil internethozzáféréssel. A várható hálózati adatforgalom alacsony, így nyugodtan üzemeltethető mobilinternet mellett is.

A program telepítése

A webalkalmazás felhasználói oldalról nem igényel telepítést, annak beüzemelése fejlesztői vagy rendszerüzemeltetői feladat, melyet egyszer szükséges elvégezni. Az ehhez szükséges információk és utasítások rendelkezésre állnak a fejlesztői dokumentációban. Ha megtörtént a beüzemelés, a webalkalmazást valamely böngészőben a szerver címének megadásával bármikor lehet használni.

A program használatának részletes leírása

Felhasználókezelés

Miután a fejlesztő vagy a rendszerüzemeltető beállította a kívánt szerveren a webalkalmazást, a felhasználó rendelkezésére fogja bocsátani a weboldal címét és a számára létrehozott felhasználónevet és jelszót.

Bejelentkezés

Az alkalmazás címét a böngészőben megnyitva a bejelentkezési űrlap^(1. képernyőkép) lesz látható. Ezen két beviteli mező található, egy a felhasználónévnek, egy pedig a jelszónak. A helyes adatok megadása után a bejelentkezés gombra kell kattintani. Ha az adatokat helytelenül adjuk meg, egy hibaüzenet jelenik meg „Hibás belépési adatok!” szöveggel. Ilyenkor az űrlapot újra ki kell tölteni és el kell küldeni, figyelve a megadott adatok helyességére. Amennyiben a

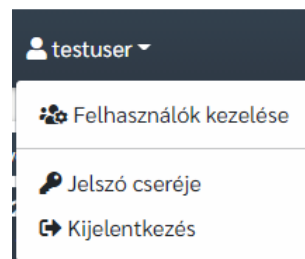


1. képernyőkép - Bejelentkezési űrlap

mezők helyesen kerülnek kitöltésre, a felhasználó a főoldalra kerül átirányításra, ahol a felhasználóneve látható lesz a navigációs sáv utolsó elemében.

Jelszó cseréje

A jelszó cseréjéhez a navigációs sávon található felhasználónévre szükséges kattintani. Ekkor megjelenik a felhasználókezeléssel kapcsolatos legördülő menüsor.^(2. képernyőkép) Ebben a „Jelszó cseréje” opcióra kattintva nyílik meg a jelszócsere felület.



2. képernyőkép - Legördülő menüsor

A felületen egy űrlap található 3 beviteli mezővel. A jelszó megváltoztatásához a régi jelszót helyesen, majd az új jelszót kétszer szükséges megadni. Az űrlap valós időben jelzi, hogy a megadott értékek helyesek-e, illetve, ha nem, akkor mi a hiba pontos oka. Ha minden adat helyesen beírásra került, akkor az űrlapot a „Jelszó cseréje” feliratú gombbal lehet elküldeni. Amennyiben a régi jelszó helytelenül lett megadva, az alábbi hibaüzenet fog megjelenni: „A jelszó cseréje nem sikerült! Kérlek, ellenőrizd a régi jelszó helyességét!” Ha a jelszóváltoztatás végre lett hajtva, akkor pedig a „A jelszó cseréje sikeres volt!” értesítés kerül kiírásra.

Felhasználók létrehozása és törlése

Az előző folyamat leírásakor említett felhasználókezeléssel kapcsolatos menüsorban a „Felhasználók kezelése” opcióra kattintáskor a meglévő felhasználók adminisztrációs felülete nyílik meg.^(3. képernyőkép) Itt lehetőség van új felhasználó létrehozására, illetve a további felhasználók adatainak megtekintésére és törlésére.

A screenshot of the 'Felhasználók kezelése' (User Management) interface. The top header shows the 'TeamAd' logo and navigation links for 'Kurzusok', 'Résztevők', 'Szűnetek', and 'testuser'. The main title is 'Felhasználók kezelése'. Below it is a section for 'Új felhasználó' (New User) with four input fields: 'Felhasználónév:' (Username), 'e-mail cím:' (Email), 'Jelszó:' (Password), and 'Jelszó:' (Confirm Password). Each field has a red border and a red error message below it: 'Rövidebb, mint 5 karakter' (Shorter than 5 characters), 'Helytelen formátum' (Invalid format), 'Rövidebb, mint 8 karakter!' (Shorter than 8 characters!), and 'Nincs megadva' (Not provided). A '+ Létrehozás' (Create) button is at the bottom of this section. Below the form is a section for 'Rögzített felhasználók' (Fixed Users) with a table showing user details. The table has columns for 'Id', 'Felhasználónév', and 'e-mail'. It contains one row with '1', 'suleszabi', and 'suleszabi@gmail.com'. A red minus icon is at the bottom right of the table.

3. képernyőkép - Felhasználók adminisztrációs felülete

Új felhasználó létrehozásához az „Új felhasználó” szürke sávra szükséges kattintani, melynek hatására megjelenik az ehhez szükséges űrlap 4 beviteli mezővel: Felhasználónév, e-mail cím, jelszó és jelszó megerősítése. Az űrlap csak akkor küldhető el, ha minden adat a követelményeknek megfelelően lett megadva. Ebben segítenek a beviteli mezők alatt található iránymutató üzenetek, melyek valós időben, karakterleütésenként frissülnek. Ha minden adat helyesen került megadásra és az űrlapot elküldjük, az új felhasználó létrehozásra kerül.

Egy meglévő felhasználó törléséhez a „Rögzített felhasználók” fejlécű dobozban található táblázatban a hozzá tartozó sor végén található piros szimbólumra kell kattintani. A törlés előtt az alkalmazás megerősítést fog kérni, melyet jóváhagyva az adott fiók megszűnik. Természetesen ebben a listában az aktuálisan használt felhasználói fiók nem kerül megjelenítésre.

Kijelentkezés

A felhasználó kijelentkezhet az alkalmazásból. Ehhez a felhasználókezeléssel kapcsolatos legördülő menüsor utolsó „Kijelentkezés” opciójára kell kattintani. Kijelentkezés után az alkalmazás felületei – kivéve a bejelentkezés – nem hozzáférhetők. Az alkalmazás újbóli használatához újra be kell jelentkezni.

Szünet-időszakok kezelése

A szünetek a kurzusok ütemezésekor figyelembe lesznek véve, azaz ha egy alkalom ütközne egy szünettel, akkor az kihagyásra kerül, és a kurzus annnyival tovább fog tartani.

Új szünet rögzítése és törlése

A navigációs sáv „Szünetek” menüpontjára kattintva megjelenik a szünetek adminisztrációs felülete. Itt lehetőség van a már rögzített szünetek áttekintésére és törlésére, illetve újabb szünet-időszakok rögzítésére.

Új időszak létrehozásához az „Új szünet rögzítése” fejléccel ellátott dobozban található űrlapot^(4. képernyőkép) kell kitölteni, és elküldeni a „Létrehozás” gombra kattintva. Itt meg kell adni a szünet megnevezését, illetve kezdő és befejező időpontját. Mind a két időpontot két részletben kell megadni, egy dátummal és egy óra-

Új szünet rögzítése

Új szünet neve:

Szünet kezdete: Szünet vége:

éééé. hh. nn. éééé. hh. nn.

--:-- --:--

+ Létrehozás

4. képernyőkép - Új szünet rögzítését szolgáló űrlap

perc időponttal. Az adatok megadásakor figyelni kell arra, hogy a kezdő időpontnak kisebbnek kell lennie, mint a befejező időpontnak. Helytelen kitöltés esetén a rögzítés sikertelen lesz.

Ha egy szünet-időszak visszavonása, azaz törlése szükséges, akkor ez a „Rögzített szünetek” című doboz listájában^(5. képernyőkép) megtehető, az érintett szünet sorának végén

Rögzített szünetek			
Név	Kezdete	Vége	
Év végi szünet	2021-12-20 00:00:00	2022-01-02 23:59:59	
Mindenszentek	2021-10-30 00:00:00	2021-11-01 23:59:59	
Ünnepnap 56-os forradalom	2021-10-23 00:00:00	2021-10-23 23:59:59	

5. képernyőkép - Rögzített szünetek listája

található piros színű gombra kattintva. A törlés előtt az alkalmazás felugró ablakban a felhasználó megerősítését fogja kérni, hogy a véletlen kattintás miatti adatmódosítás elkerülhető legyen.

Kurzusok kezelése

Új kurzus rögzítése

A navigációs sáv „Kurzusok” menüpontjára kattintva az oldal megnyitja a kurzusok kezelésére szolgáló felületet.^(6. képernyőkép) Itt az „Új kurzus” szürke sávra kattintva kinyílik egy űrlap egy beviteli mezővel. Ezt kitöltve és elküldve létre lehet hozni egy újabb kurzust. A név megadásakor fontos követelmény, hogy az minimum 4, maximum 40 karakter hosszú legyen. Amennyiben ez a feltétel nem teljesül, az űrlap nem elküldhető.

Egy újonnan létrehozott kurzusnak még nincs kezdő és befejező időpontja. Ehhez előbb be kell állítani annak beosztását, és ez alapján lehet ütemezni az időpontjait. Ezek után a kezdő időpont az első alkalomból, a befejező időpont az utolsó alkalomból kerül megjelenítésre.

+ Új kurzus

Kurzus neve:

Rövidebb, mint 9 karakter

+ Létrehozás

Rögzített kurzusok

Id	Név	Alkalmak	Kezdés	Befejezés
3	React Tanfolyam	8 alk.	2021-10-02 09:00 Szombat	2021-12-04 16:00 Szombat
2	JavaScript tanfolyam	12 alk.	2021-09-29 09:00 Szerda	2021-11-17 12:15 Szerda
1	Webszerkesztés tanfolyam	13 alk.	2021-09-27 17:00 Hétfő	2021-11-17 20:15 Szerda

6. képernyőkép - Új kurzus létrehozására és a rögzített kurzusok áttekintésére szolgáló dobozok

Meglévő kurzus törlése

Az előzőleg említett kurzusok kezelésére szolgáló felületen, a „Rögzített kurzusok” fejlécű dobozban áttekinthetők a meglévő kurzusok főbb adatai. Ezen lista minden sorának végén található 3 gomb. A pirosra kattintva az adott kurzus törölhető. Törlés előtt az alkalmazás megerősítést fog kérni a felhasználatól.

Kurzus nevének módosítása

A „Rögzített kurzusok” fejlécű dobozban a kék színű „Részletek” gombra kattintva egy új felületre kerül a felhasználó, ahol az adott kurzushoz köthető összes adatot lehet áttekinteni, illetve módosítani. Ezen felület „Alapadatok” című dobozában^(7. képernyőkép) látható a kurzus neve és alkalmainak száma, illetve lehetőség van az előbbi módosítására is, a beviteli mező helyes kitöltésével és a „Módosítás” gombra való kattintással. A módosítás csak akkor hajtható végre, ha az új név megfelel a követelményeknek.

Alapadatok

Név: Webszerkesztés tanfolyam

Alkalmak: 13 alk.

Kurzus nevének módosítása:

Rendben

Módosítás

Beosztás

Kezdés	Befejezés
Csütörtök 16:00	Csütörtök 19:15
Hétfő 17:00:00	Hétfő 20:15:00
Szerda 17:00:00	Szerda 20:15:00

7. képernyőkép - Kurzus alapadatainak és a beosztásainak kezelését szolgáló dobozok

Kurzus beosztásának beállítása

Az alkalmak időpontjainak kiszámításához minden kurzusnak be kell állítani a heti beosztását. Erre és a meglévő időszakok áttekintésére a „Beosztás” fejléccel rendelkező dobozban^(7. képernyőkép) van lehetőség. A kezdő és befejező időpontokat megadva és a plusz gombra kattintva, egy hétre bármennyi különálló időszak felvehető beosztásként. Mind a két időpont két külön adattal adatható meg: a hét valamely napjával és a pontos idővel, melyet óra:perc pontossággal szükséges kitölteni. Fontos szempont, hogy egy újabb beosztás kezdő időpontjának korábbinak kell lennie, mint a befejezőnek, és az új időszak nem ütközhet a meglévőkkel. Amennyiben ezen feltételek valamelyike nem teljesül, a rögzítési művelet sikertelen lesz, melyről a felület üzenetben értesíti a felhasználót.

A meglévő beosztások természetesen törölhetők is, az adott sor végén található piros színű gombra kattintva. A szokott módon az alkalmazás megerősítést fog kérni a törléshez.

Kurzus alkalmainak ütemezése

Amennyiben egy kurzus számára be lett állítva a beosztása, az alkalmak ütemezésére lehetőség nyílik. Ehhez az „Ütemezés” fejléccel rendelkező dobozban^(8. képernyőkép) található űrlapot kell kitölteni és végrehajtani. A folyamathoz két adatra van szükség: az adott kurzus kezdő dátumára és a tervezett alkalmainak számára. Ha ez a két adat megadásra került, az „Ütemezés” gombra kattintva elindul az alkalmak időpontjainak kiszámítása. Ez a folyamat figyelembe veszi a már rögzített szünet-

Ütemezés		Alkalmak																																																																																															
Ütemezés kezdő dátuma: <input type="text" value="2021. 09. 27."/>																																																																																																	
Alkalmak száma: <input type="text" value="13"/>																																																																																																	
<input type="button" value="▶ Újraütemezés"/>																																																																																																	
		<table border="1"> <thead> <tr> <th colspan="3">Kezdés</th> <th colspan="3">Befejezés</th> </tr> <tr> <th>Dátum</th> <th>Nap</th> <th>Idő</th> <th>Dátum</th> <th>Nap</th> <th>Idő</th> </tr> </thead> <tbody> <tr><td>2021-09-27</td><td>Hétfő</td><td>17:00</td><td>2021-09-27</td><td>Hétfő</td><td>20:15</td></tr> <tr><td>2021-09-29</td><td>Szerda</td><td>17:00</td><td>2021-09-29</td><td>Szerda</td><td>20:15</td></tr> <tr><td>2021-10-06</td><td>Szerda</td><td>17:00</td><td>2021-10-06</td><td>Szerda</td><td>20:15</td></tr> <tr><td>2021-10-11</td><td>Hétfő</td><td>17:00</td><td>2021-10-11</td><td>Hétfő</td><td>20:15</td></tr> <tr><td>2021-10-13</td><td>Szerda</td><td>17:00</td><td>2021-10-13</td><td>Szerda</td><td>20:15</td></tr> <tr><td>2021-10-18</td><td>Hétfő</td><td>17:00</td><td>2021-10-18</td><td>Hétfő</td><td>20:15</td></tr> <tr><td>2021-10-20</td><td>Szerda</td><td>17:00</td><td>2021-10-20</td><td>Szerda</td><td>20:15</td></tr> <tr><td>2021-10-25</td><td>Hétfő</td><td>17:00</td><td>2021-10-25</td><td>Hétfő</td><td>20:15</td></tr> <tr><td>2021-10-27</td><td>Szerda</td><td>17:00</td><td>2021-10-27</td><td>Szerda</td><td>20:15</td></tr> <tr><td>2021-11-03</td><td>Szerda</td><td>17:00</td><td>2021-11-03</td><td>Szerda</td><td>20:15</td></tr> <tr><td>2021-11-10</td><td>Szerda</td><td>17:00</td><td>2021-11-10</td><td>Szerda</td><td>20:15</td></tr> <tr><td>2021-11-15</td><td>Hétfő</td><td>17:00</td><td>2021-11-15</td><td>Hétfő</td><td>20:15</td></tr> <tr><td>2021-11-17</td><td>Szerda</td><td>17:00</td><td>2021-11-17</td><td>Szerda</td><td>20:15</td></tr> </tbody> </table>						Kezdés			Befejezés			Dátum	Nap	Idő	Dátum	Nap	Idő	2021-09-27	Hétfő	17:00	2021-09-27	Hétfő	20:15	2021-09-29	Szerda	17:00	2021-09-29	Szerda	20:15	2021-10-06	Szerda	17:00	2021-10-06	Szerda	20:15	2021-10-11	Hétfő	17:00	2021-10-11	Hétfő	20:15	2021-10-13	Szerda	17:00	2021-10-13	Szerda	20:15	2021-10-18	Hétfő	17:00	2021-10-18	Hétfő	20:15	2021-10-20	Szerda	17:00	2021-10-20	Szerda	20:15	2021-10-25	Hétfő	17:00	2021-10-25	Hétfő	20:15	2021-10-27	Szerda	17:00	2021-10-27	Szerda	20:15	2021-11-03	Szerda	17:00	2021-11-03	Szerda	20:15	2021-11-10	Szerda	17:00	2021-11-10	Szerda	20:15	2021-11-15	Hétfő	17:00	2021-11-15	Hétfő	20:15	2021-11-17	Szerda	17:00	2021-11-17	Szerda	20:15
Kezdés			Befejezés																																																																																														
Dátum	Nap	Idő	Dátum	Nap	Idő																																																																																												
2021-09-27	Hétfő	17:00	2021-09-27	Hétfő	20:15																																																																																												
2021-09-29	Szerda	17:00	2021-09-29	Szerda	20:15																																																																																												
2021-10-06	Szerda	17:00	2021-10-06	Szerda	20:15																																																																																												
2021-10-11	Hétfő	17:00	2021-10-11	Hétfő	20:15																																																																																												
2021-10-13	Szerda	17:00	2021-10-13	Szerda	20:15																																																																																												
2021-10-18	Hétfő	17:00	2021-10-18	Hétfő	20:15																																																																																												
2021-10-20	Szerda	17:00	2021-10-20	Szerda	20:15																																																																																												
2021-10-25	Hétfő	17:00	2021-10-25	Hétfő	20:15																																																																																												
2021-10-27	Szerda	17:00	2021-10-27	Szerda	20:15																																																																																												
2021-11-03	Szerda	17:00	2021-11-03	Szerda	20:15																																																																																												
2021-11-10	Szerda	17:00	2021-11-10	Szerda	20:15																																																																																												
2021-11-15	Hétfő	17:00	2021-11-15	Hétfő	20:15																																																																																												
2021-11-17	Szerda	17:00	2021-11-17	Szerda	20:15																																																																																												

8. képernyőkép - Kurzus ütemezését és alkalmainak megjelenítését szolgáló dobozok

időszakokat, azaz nem fog létrejönni olyan kurzusalkalom, mely ütközne valamelyik szünettel. Az ilyen időpont „átugrásra” kerül.

Amennyiben valamelyik alkalom egy másik kurzus már beütemezett időpontjával ütközne, a folyamat sikertelen lesz, és nem lesznek létrehozva új időpontok. Erről a hibáról a felület üzenetben értesíti a felhasználót. Ezen probléma megoldása lehet a beosztás és/vagy a kezdőidőpont átállítása, majd az ütemezés újraindítása.

Ha egy új szünet-időszak kerül beállításra és az ütközik egy meglévő kurzussal, akkor az adott kurzust újra lehet ütemezni, amely után az érintett alkalmak kimaradnak, és ezek számával a kurzus befejező időpontja eltolódik. Amennyiben egy kurzusnak már vannak ütemezett alkalmi, az ütemezés ugyanúgy végrehajtható az „Ütemezés” doboz használatával, viszont ilyenkor a végrehajtó gomb neve már „Újraütemezés” lesz.

A kurzus alkalmainak adatai az „Alkalmak” dobozban^(8. képernyőkép) kerülnek megjelenítésre listaként. Minden alkalom kezdő és befejező időpontja látható külön sorban.

Kurzus résztvevőinek névsora

Az adott kurzusra besorolt résztvevők adatait a „Névsor” dobozban lehet áttekinteni, melynek fejlécében a létszám is megjelenik. A résztvevőket ki is lehet sorolni a kurzusról az adott sor végén látható piros színű gombbal. A kisoroláshoz is megerősítést fog kérni az alkalmazás.

Résztvevők kezelése

Új résztvevő rögzítése

A navigációs sáv „Résztvevők” opciójára kattintva az alkalmazás résztvevők kezeléséért felelős felülete nyílik meg. Ennek az „Új résztvevő” című dobozában^(9. képernyőkép) található egy űrlap, mely segítségével a felhasználó újabb klienseinek adatait rögzítheti. Ehhez négy adatot szükséges megadni: A résztvevő teljes nevét, e-mail címét, születési helyét és idejét. A megadott adatoknak külön-külön, eltérő követelményeknek

9. képernyőkép - Új résztvevő rögzítését szolgáló doboz

kell megfelelnie. Például a teljes névnek minimum 6, maximum 40 karakter hosszúnak kell lennie, az e-mail címnek helyes formátummal kell rendelkezni és nem lehet hosszabb 50 karakternél. A követelmények teljesüléséről az adott beviteli mező alatt kap visszajelzést a felhasználó, mely karakterleütésenként frissül. Egészen addig, amíg az összes mező nem lesz helyes, a rögzítés végrehajtásához szükséges „Létrehozás” gomb inaktív marad.

Rögzített résztvevők kezelése

A „Résztevők listája” fejléccel rendelkező dobozban^(10. képernyőkép) listázásra kerülnek a már rögzített felhasználók adatai, illetve lehetőség van keresésre is. Utóbbihoz az oszlopok fejlécei alatt található mezőket kell használni. Minden mező az alatta lévő oszlop adataiban szűr. A keresés minden leütés után, valós időben történik. Egyszerre több oszlopban is adható meg keresési kifejezés, azaz összetett szűrés is végrehajtható.

Résztevők listája				
ID	Teljes név	e-mail cím	Születési hely	Születési idő
<input type="text" value="Keresés ID-ra"/>	<input type="text" value="Barna"/>	<input type="text" value="Keresés e-mail címre"/>	<input type="text" value="Keresés születési helyre"/>	<input type="text" value="Keresés születési időre"/>
13	Barna Mária	barna.maria@dzsmail.com	Marcali	1969-06-17
2	Barna Krisztina	barna.krisztina@dzsmail.com	Sajószentpéter	1997-06-20

10. képernyőkép - Résztevők áttekintését és keresését szolgáló doboz

Résztevő törlése

Az adott résztvevő törölhető a hozzá tartozó sor végén található piros színű gombbal. A törlés végrehajtásához az alkalmazás a felhasználó megerősítését kéri.

Résztevő csoportba sorolása

Az adott résztvevő csoportba sorolásához a sorának végén található zöld színű gombra kell kattintani. Ilyenkor egy felugró ablak^(11. képernyőkép) jelenik meg a képernyőn, melyben megjelenítésre kerül az adott tanuló neve, illetve azon kurzusok listája, melyekben még nincs benne a résztvevő. Minden sor végén található egy-egy zöld gomb,

Csoportba sorolás			
Barna Krisztina			
ID	Név	Kezdés	Befejezés
3	React Tanfolyam	2021-10-02	2021-12-04
2	JavaScript tanfolyam	2021-09-29	2021-11-17
Bezárás			

11. képernyőkép - Résztevő csoportba sorolását szolgáló felugró ablak

mely segítségével kiválasztható, hogy a résztvevő melyik kurzusba kerüljön besorolásra. A kiválasztás után a felugró ablak eltűnik.

Résztvevő adatainak módosítása

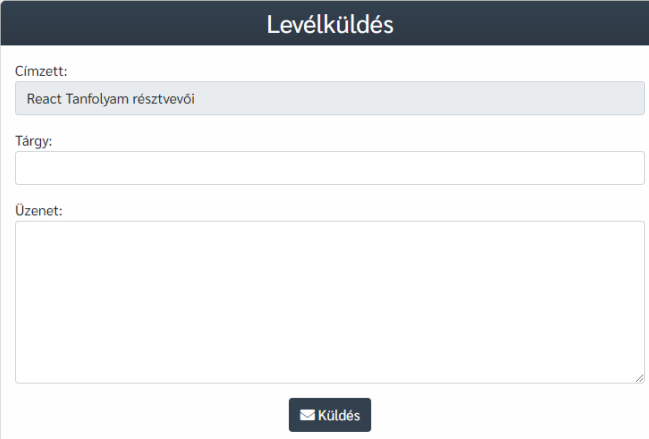
A rögzített résztvevők adatai természetesen szerkeszthetők. Ehhez az adott személy sorának végén található kék színű gombra kell kattintani. Ezután a felhasználó a résztvevő adatainak szerkesztését szolgáló felületre kerül. Itt található egy űrlap, melynek megjelenése és használata szinte teljesen megegyezik az új felhasználó létrehozását végzővel. A különbség csupán annyi, hogy ez az űrlap alapértelmezetten kitöltésre kerül a résztvevő aktuális adataival. A módosítani kívánt mező átírása után a „Szerkesztés” gombra kattintva a változás rögzítésre kerül. Amennyiben valamely új adat nem felel meg a követelményeknek, a végrehajtó gomb inaktívvá válik.

Levélküldés a résztvevők számára

Levélküldés indítása

Az alkalmazáson belül két helyről lehet levelet küldeni, annak függvényében, hogy a címzett egy konkrét személy, vagy egy kurzus összes résztvevője. Egy címzett esetén a „Résztvevők” menüpont „Résztvevők listája” dobozon belül^(10. képernyőkép), az adott személy e-mail címe előtti sárga borítékra kattintva kerül a felhasználó átirányításra a levélküldő felületre. Ha a címzett egy kurzus összes résztvevője, akkor a „Kurzusok” menüpont „Rögzített kurzusok” dobozában^(6. képernyőkép), az adott csoport sorának végén található, szintén sárga színű borítékra szükséges kattintani.

A kiindulási ponttól függetlenül a levélküldő felület megjelenése és működése megegyező.^(12. képernyőkép) Két mezőt szükséges kitölteni: a levél tárgyat és az üzenet szövegét. A felületen található egy harmadik mező a címzett számára. Ezt az alkalmazás automatikusan kitölti, annak függvényében, hogy a levél egy résztvevőnek vagy egy csoportnak van címezve, így ez nem is módosítható.



12. képernyőkép - Levélküldő felület

Ha a levél adatai megadásra kerültek, azt el lehet küldeni a „Küldés” gombbal. Amíg tart a folyamat, az összes beviteli mező inaktívvá válik. Ha a művelet véget ért, az alkalmazás üzenetben tájékoztatja a felhasználót.

Főoldal – Heti beosztás

Heti beosztás interaktív, grafikus megjelenítése

Bejelentkezés után közvetlenül, vagy a navigációs sávon a „TeamAd” névre kattintva az alkalmazás főoldalára vagy másnéven nyitóoldalára kerül a felhasználó. Annak ellenére, hogy ez az alkalmazás kezdő felülete, mégis utoljára kerül bemutatásra. Ennek oka, hogy az itt megjelenő tartalmak erősen függenek a többi felület működésétől és használatától.

← Előző hét								Következő hét →
Idő	2021-9-27	2021-9-28	2021-9-29	2021-9-30	2021-10-1	2021-10-2	2021-10-3	
óó:pp	Hétfő	Kedd	Szerda	Csütörtök	Péntek	Szombat	Vasárnap	
00:00								
02:00								
04:00								
06:00								
08:00								
10:00			JavaScript tanfolyam 09:00 - 12:15			React Tanfolyam 09:00 - 16:00	Utazás Debrecenbe 10:00-tól	
12:00								
14:00								
16:00								
18:00	Webszerkesztés tanfolyam 17:00 - 20:15		Webszerkesztés tanfolyam 17:00 - 20:15					
20:00								
22:00								

13. képernyőkép - Heti beosztás grafikus megjelenítése a főoldalon

A főoldalon alapértelmezetten az aktuális hét beosztása kerül megjelenítésre grafikus eszközök segítségével.^(13. képernyőkép) Minden nap számára található egy-egy külön hasáb, melynek fejlécében látható a dátum és a nap neve. Az első oszlop a napi időbeosztást segíti azzal, hogy jelzi a páros órák függőleges elhelyezkedését. Az adott héten zajló kurzusok és szünetek a kezdőidőpontjuknak megfelelő pozícióban jelennek meg, az időbeli hosszukkal arányos magassággal. A kurzusok színe sárga, míg a szüneteké kék. Egy-egy esemény dobozában látható annak megnevezése és napon belüli időintervalluma. Amennyiben egy esemény korábbi napon kezdődött és az adott napon csak a befejező időpontja esedékes, csak a befejezés időpontja jelenik meg. Természetesen ez fordítva is igaz, ha egy alkalomnak csak a kezdőidőpontja esik egy dátumra, akkor csak a kezdés ideje kerül elhelyezésre.

Az eseményekre a felhasználó rákattinthat. Amennyiben ez egy kurzus dobozán történik, akkor annak adatainak áttekintését és módosítását szolgáló felületre kerül átirányításra. Ha viszont szünetre kattint, akkor a szünetek kezelését biztosító oldalra jut.

Ha a felhasználó további hetek beosztásaira is kíváncsi, lehetősége van előre- és visszafelé is léptetni a heteket a „Következő hét” és az „Előző hét” gombokkal. Az aktuális hétre is vissza lehet ugrani a navigációs sávon található „TeamAd” névre kattintással.

Fejlesztői dokumentáció

Témaválasztás indoklása

Jómagam gyakorlati oktatóként tevékenykedem a felnőttképzésben. Több kollégámmal egyeztetve felmerült az igény egy webalkalmazásra, ami a kurzusokkal kapcsolatos adminisztrációs feladatokat segítené, illetve könnyítené.

Rövid piackutatás után megállapítottam, hogy jelenleg nehezen található erre a konkretizált igényre kész alkalmazás. Persze az oktatási intézmények általában szoktak rendelkezni belső vállalatirányítási rendszerrel, de ezek túl összetettek, több oktatóra és tanfolyamra, illetve jelentősebb mennyiségű tanulóra vannak kifejlesztve. Az is igaz, hogy egy ilyen programnak, ha csak kimondottan oktatók számára készül, akkor a célcsoportja egy elég szűk réteg, tehát nem biztos, hogy megéri lefejleszteni. Véleményem szerint viszont a célcsoportot könnyedén ki lehet bővíteni, hiszen ha nem kizárólag „oktatók” számára készül, hanem általánosságban véve minden olyan egyéni szolgáltatónak, aki a szolgáltatását csoportosan adja át az ügyfeleinek, akkor máris sokkal több személy kerül az emberek azon halmazába, akiknek hasznára válhat egy ilyen webalkalmazás. Ha jobban belegondolunk, sok ilyen csoportos szolgáltatót fel tudunk sorolni, mint például a már említett oktatók, előadók és tanárok, vagy egyéb területeken akár edzők, motivációs trénerek, motoros oktatók stb.

Saját magamból kiindulva, elsősorban a leggyakrabban felmerülő problémákra, feladatokra szerettem volna megoldást nyújtani ezzel a webalkalmazással. Egyrészt szükséges a kurzusok és a résztvevők adatainak biztonságos tárolása és rendszerezése, illetve könnyű elérése. Az időpontok rendezési feladata is körülményes lehet egyszerű eszközök segítségével, mint például naptár és táblázatkezelő program. Tehát célként fogalmazódott meg egy másik funkció is, mely a kurzusokat automatikusan ütemezi úgy, hogy közben elkerülhető az időpontütközés más elfoglaltságokkal. Az ütemezés kapcsán segítség a felhasználónak az is, hogy ha az aktuális időpontjait könnyedén átláthatja, így elkészítésre került egy grafikus heti beosztásmegjelenítő felület is. A felsoroltakon kívül pedig kiemelt cél volt a kapcsolattartás megkönnyítése is. Így megvalósításra került a levélküldés funkciója is, melynek címzettjei lehetnek konkrét résztvevők, vagy komplett csoportok is.

Alkalmazott fejlesztői eszközök

Webalkalmazás fejlesztése során használt eszközök

A webalkalmazás fejlesztése során az alábbi program-, parancs- és leírónyelvek kerültek felhasználásra:

- Kliensoldal – Frontend:
 - Tartalmak leírása:
 - HTML5
 - Megjelenés beállítása, formázás:
 - CSS3
 - Bootstrap 5
 - Interakciók kezelése, tartalmak manipulálása, aszinkronikus kérések kezelése:
 - JavaScript
- Szerveroldal – Backend:
 - Adatbázis parancsnyelv:
 - MySQL
 - Szerveroldali folyamatok kezelése:
 - PHP (OOP)

A fejlesztés során az alábbi szoftverek és segédprogramok használata volt szükséges:

- Kódszerkesztés:
 - Visual Studio Code (Microsoft)
- Web- és adatbáziskiszolgáló:
 - XAMPP szoftvercsomag
 - Apache HTTP webservert
 - MariaDB (MySQL) adatbázis-kiszolgáló
 - phpMyAdmin adatbáziskezelő alkalmazás
 - PHP interpreter (értelmező)
- Webalkalmazás tesztelése:
 - Google Chrome
 - Microsoft Edge
- Logó tervezés, színek kiválasztása:
 - Adobe Photoshop CC

A webalkalmazás a levélküldéshez a PHPMailer nevű, mások által fejlesztett és szabadon felhasználható PHP osztályt és annak segédosztályait használja.

Dokumentáció készítése során használt eszközök

A dokumentáció készítéséhez az alábbi szoftverek kerültek felhasználásra:

- Dokumentáció írása és formázása:
 - Microsoft Office Word 2019
- Prezentáció elkészítése:
 - Microsoft Office PowerPoint 2019
- Képernyőképek készítése és szerkesztése:
 - Microsoft Windows 10 Pro
 - Google Chrome
 - Microsoft Paint
- Ábrák rajzolása, készítése
 - draw.io

Adatmodell leírása

A webalkalmazás által használt adatok tárolása MariaDb vagy MySQL relációs adatbázisokban történik. Ez lehetőséget ad az adatok védelméhez, ugyanis a kiszolgálóhoz való hozzáféréshez ismerni kell a felhasználónevet és a hozzá tartozó jelszót. Másik előnye, hogy ha megfelelően kerülnek konfigurálásra a táblaindexek, a táblakapcsolatok és a mezőindexek, akkor a műveletek nagy mennyiségű adatokkal is elfogadhatóan rövid idő alatt lefutnak.

A program 2 külön adatbázist használ. Az autentikációhoz szükséges felhasználói adatok elkülönítésre kerültek a résztvevők és a kurzusok, illetve a hozzájuk kapcsolódó adatoktól. Ennek egyik oka, hogy csökkenjen a biztonsági kockázat, a másik, hogy a két adathalmaz egymástól teljesen független, nincs közöttük semmilyen kapcsolat.

Felhasználói adatok adatbázisa

A felhasználói adatok tárolása egy táblában történik.

Az „id” mező a tábla indexe és elsődleges kulcsa. Típusa egész szám, ami automatikusan növekszik, így garantálva, hogy az érték

minden rekordnál egyedi legyen. Mivel az alkalmazás alapvetően egy egyéni szolgáltató számára készült, így nem valószínű, hogy sok felhasználói fiók lesz tárolva, így a maximális hossz 2 karakter lehet.

user		
Mezőnév	Típus és hossz	Tulajdonságok
id	Int(2)	PK, AI, NN
username	Varchar(10)	NN
hash_salt	Varchar(10)	NN
hash_pwd	Varchar(128)	NN
email	Varchar(50)	NN

A felhasználók táblájából jól látszik, hogy az alkalmazás jelszókezelése és tárolása korszerű, megfelel a mai elvárásoknak. A SHA512 titkosító algoritmus használata előtt a jelszavak „sózása” is végbemegy.

Kurzusok és résztvevők adatbázisa

A kurzusok és a résztvevők, illetve az ezekhez kapcsolódó egyéb adatok tárolásához hat táblára van szükség. Az indexek esetében a mezőhosszok megállapításához egy átlagos egyéni szolgáltató 5 éves adatmennyisége lett figyelembe véve. Ez várhatóan 100 csoport, 5000 résztvevő, 1000 beosztás, 8000 kurzusalkalom, illetve 150 szünet-időszak.

Résztvevők táblája

Az „id” mező a tábla indexe és elsődleges kulcsa, mely ebben a táblában is automatikusan növekszik. A tábla minden mezőjének kitöltése kötelező. Egy résztvevő esetében eltárolásra kerül annak teljes neve, e-mail címe, születési helye és ideje.

participant		
Mezőnév	Típus és hossz	Tulajdonságok
id	Int(4)	PK, AI, NN
full_name	Varchar(40)	NN
email	Varchar(50)	NN
birthplace	Varchar(20)	NN
birthdate	date	NN

Kurzusok táblája

Az index, illetve elsődleges kulcs ebben a táblában is az „id” mezőben kerül tárolásra. Egy kurzus esetében a megnevezése kerül eltárolásra.

course		
Mezőnév	Típus és hossz	Tulajdonságok
id	Int(3)	PK, AI, NN
name	Varchar(40)	NN

Résztvevők és kurzusok kapcsoló táblája

A résztvevők és a kurzusok között „több a többhöz” kapcsolat áll fent, hiszen egy kurzuson többen is részt vehetnek, és egy személy

part_course		
Mezőnév	Típus és hossz	Tulajdonságok
id	Int(5)	PK, AI, NN
participant	Int(4)	FK, NN
course	Int(3)	FK, NN

több kurzusra is jelentkezhet. Ennek a táblának ez a tény ad létjogosultságot, ugyanis ezt a fajta kapcsolatot csak egy kiegészítő kapcsolótábla segítségével lehet megvalósítani úgy, hogy ne keletkezzen adatismétlés, azaz redundancia.

Beosztások táblája

Egy kurzusnak egy vagy több heti beosztás-időszaka is lehet. Egy ilyen beosztás esetében el kell tárolni az időszak kezdő, illetve befejező időpontjait. Egy-egy ilyen időpont két-két adatból áll. Az egyik

schedule		
Mezőnév	Típus és hossz	Tulajdonságok
id	Int(3)	PK, AI, NN
course	Int(3)	FK, NN
start_day	Int(1)	NN
start_time	Time	NN
end_day	Int(1)	NN
end_time	Time	NN

a nap száma, mely 0 és 6 között mozoghat, a másik pedig az adott napon belüli időpont. Az előbbieket a „start_day” és az „end_day”, míg utóbbiak a „start_time” és az „end_time” mezőkben kerülnek tárolásra. Természetesen azt is rögzíteni kell, hogy egy beosztás melyik kurzushoz tartozik. Ez a „course” mezőben tárolt idegenkulccsal történik.

Alkalmak táblája

A kurzusok esetében a konkrét időpontok adatait is el kell tárolni, mely a beosztás alapján történő ütemezés során történik. Minden alkalom esetében idegenkulccsal

occasion		
Mezőnév	Típus és hossz	Tulajdonságok
id	Int(4)	PK, AI, NN
course	Int(3)	FK, NN
start	Datetime	NN
end	Datetime	NN

rögzítésre kerül a „course” mezőben, hogy melyik kurzushoz tartozik. Azt, hogy mikor kezdődik, illetve fejeződik be, a „start” és „end” mezőkből lehet megtudni.

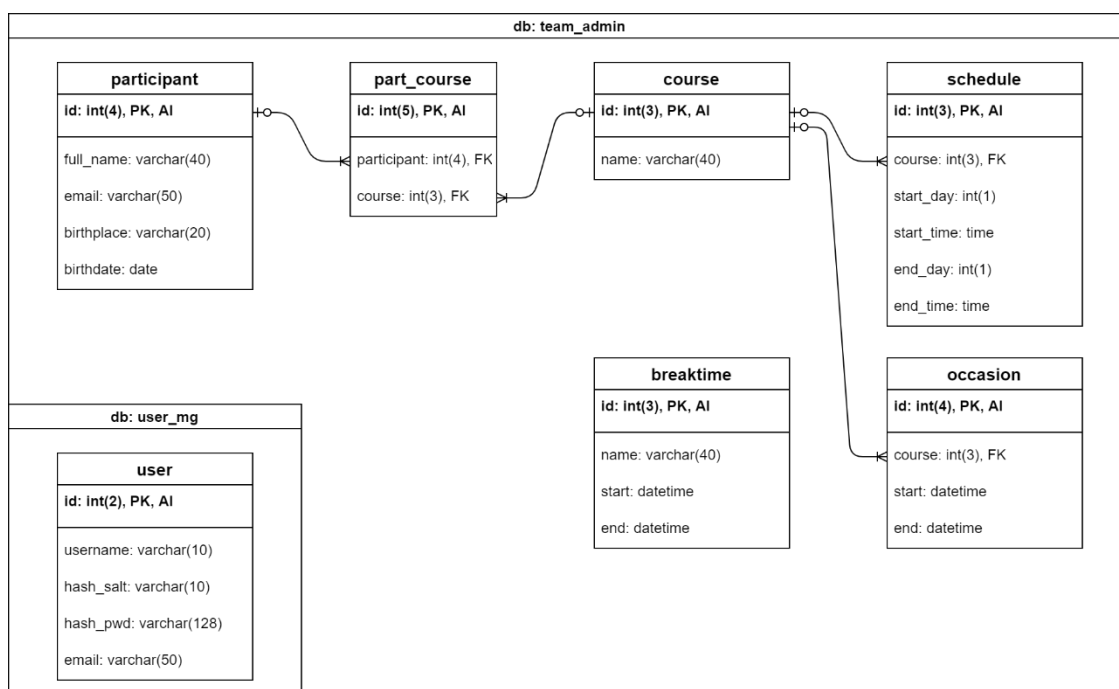
Szünetek táblája

A szünet-időszakok nagyon hasonlóak a kurzusok alkalmaihoz, hiszen ezek is kezdő, illetve befejező időpontokból állnak. Különbség viszont, hogy ebben a táblában nincs

breaktime		
Mezőnév	Típus és hossz	Tulajdonságok
id	Int(3)	PK, AI, NN
name	Varchar(40)	NN
start	Datetime	NN
end	Datetime	NN

idegenkulcs, illetve a szünet-időszakok számára megnevezés is tárolásra kerül a „name” mezőben.

A webalkalmazás adatbázisainak szerkezete



1. ábra - Adatbázisok szerkezete

Táblakapcsolatok és megkötések

A szerkezeti ábrán jól látszanak a táblák közötti kapcsolatok. A résztvevők és a kurzusok között a már korábban említett „több a többhöz” kapcsolat látható, mely a „part_course” kapcsolótáblával valósul meg. Ezen kívül még két kapcsolat van az adatbázisban. Az egyik a kurzusok és a beosztások, a másik pedig a kurzusok és az alkalmak között található. Mind a kettő „egy a többhöz” kapcsolat, hiszen egy kurzushoz több beosztás vagy alkalom tartozhat, de egy konkrét beosztáshoz vagy időponthoz csak egy kurzus.

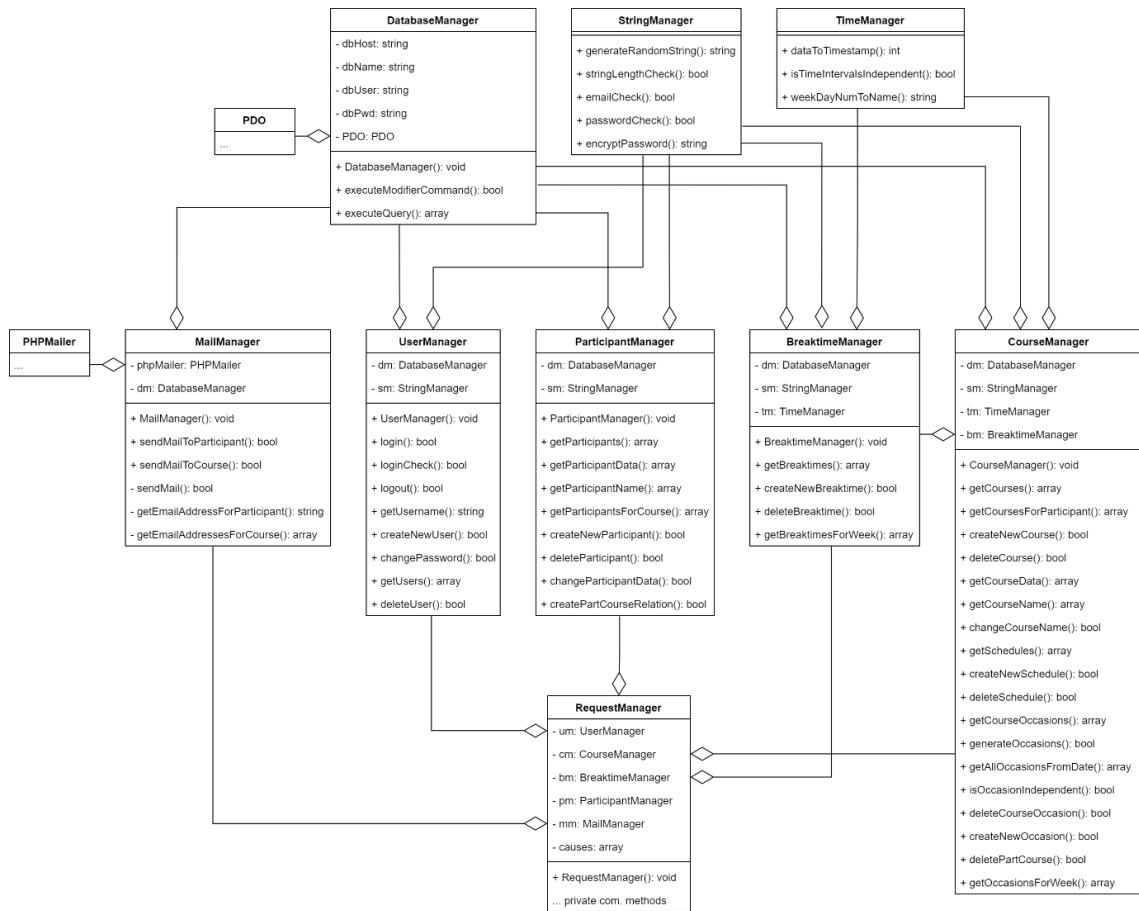
Az adatbázisban az idegen kulcsok és a rájuk vonatkozó megkötések be lettek állítva. Utóbbi minden esetben „CASCADE”, tehát egy résztvevő vagy kurzus törlésekor az egymással összekapcsoló rekord is törlésre kerül, illetve kurzus esetében a hozzá tartozó beosztások és időpontok is eltávolításra kerülnek.

Szerveroldali osztályok

A szerveroldali folyamatok megvalósítása PHP nyelven történt, az objektum orientált tervezési szemlélet szerint. A működéshez 9 saját osztály lett létrehozva. A szerver a kliens számára csak adatokat szolgál JSON formátumban, melyeket aszinkron kérésekre küld el válaszul. Az egyes folyamatok pedig POST kérések hatására kerülnek elindításra. A klienssel közvetlen kapcsolatban csak egy RequestManager osztályú objektum áll. Az ő feladata a kérések fogadása és megválaszolása. A fennmaradó 8 osztály feladatai a következők:

- DatabaseManager
 - Adatbázis utasítások végrehajtása
- StringManager
 - Karakterláncokkal kapcsolatos utasítások (pl.: hossz és formátum ellenőrzései)
- TimeManager
 - Időadatokkal összefüggő műveletek (pl.: intervallumütközések vizsgálata, időbélyeg konvertálása stb.)
- UserManager
 - Felhasználókezeléssel kapcsolatos műveletek (pl.: bejelentkezés)
- BreaktimeManager
 - Szünet-időszakok kezelése
- CourseManager
 - Kurzusok kezelése
- ParticipantManager
 - Résztevők kezelése
- MailManager
 - Levélküldés kezelése PHPMailer objektum segítségével

UML osztálydiagram



2. ábra - UML osztálydiagram

Szerveroldali rendszerkövetelmények

A webalkalmazást szerveroldalon szükséges konfigurálni. Ehhez egy HTTP webszerverre és MySQL vagy MariaDB adatbázis kiszolgálóra van szükség. A felhasználók adatbázisának várható maximális mérete 48KB, az adminisztrációs adatbázisé pedig 9040KB, ami 8,83MB.

A webszerveren az alkalmazás forrásfájljai 413KB tárhelyet fognak elfoglalni. Ezen méret nem fog változni, ugyanis a program működés közben nem fog létrehozni vagy eltárolni újabb állományokat.

Webalkalmazás beállítása szerveroldalon

A szükséges adatbázisok exportfájljai és a forráskódok megtalálhatóak a mellékelt adathordozón.

Az alkalmazás adatbázisai többféle módon is rendelkezésre állnak az „Adatbázisok” mappában. A két adatbázis egyben, próbaadatokkal feltöltve az „alldatabases.sql” fájlból importálható. Amennyiben külön-külön, csak a táblákra van szükség, a felhasználók adatbázisának tartalma a „user-database-table.sql”, míg az adminisztrációs adatbázisé az „administration-database-tables.sql” fájlból importálható. Amennyiben nincs szükség a próbaadatokra, utóbbi rendelkezésre áll üresen is a „administration-database-tables-empty.sql” fájlban.

Az „Alkalmazás” mappában találhatóak az alkalmazás forrásfájljai, tehát ez a mappa a webalkalmazás gyökérkönyvtára. Ennek tartalmát a kívánt webszerver publikus könyvtárába kell másolni. A helyes működéshez be kell állítani a levélküldő szerver és a két adatbázis adatait. Ezeket a webalkalmazás szerveroldalon PHP konstansokból használja. A konstansok definiálása a gyökérkönyvtár „php” mappájában a „consts.php” fájlban történik. Ez alapvetően helyi kiszolgálóra van konfigurálva, illetve a levélküldés a fejlesztő által bérelt levélküldő szerveren keresztül történik. A saját használathoz a konstansok értékeit át kell állítani. Ebben az alábbi táblázat tud segítséget nyújtani:

Konstans név	Jelentés
DB_USER_MG_HOST	Felhasználók adatbázisához tartozó kiszolgáló címe
DB_USER_MG_USER	Felhasználók adatbázisához tartozó felhasználónév
DB_USER_MG_PWD	Felhasználók adatbázisához tartozó jelszó
DB_USER_MG_NAME	Felhasználók adatbázisához tartozó megnevezés
DB_TEAM_ADMIN_HOST	Adminisztrációs adatbázishoz tartozó kiszolgáló címe
DB_TEAM_ADMIN_USER	Adminisztrációs adatbázishoz tartozó felhasználónév
DB_TEAM_ADMIN_PWD	Adminisztrációs adatbázishoz tartozó jelszó
DB_TEAM_ADMIN_NAME	Adminisztrációs adatbázishoz tartozó megnevezés
MAIL_HOST	Levelező SMTP szerver címe
MAIL_PORT	Levelező szerver által használt port
MAIL_USERNAME	Levelező szerverhez tartozó felhasználónév
MAIL_PASSWORD	Levelező szerverhez tartozó jelszó
MAIL_FROM_EMAIL	Levélküldés során használt feladó e-mail címe
MAIL_FROM_NAME	Levélküldés során használt feladó megnevezése

Folyamatok algoritmusai és forráskód-részletei

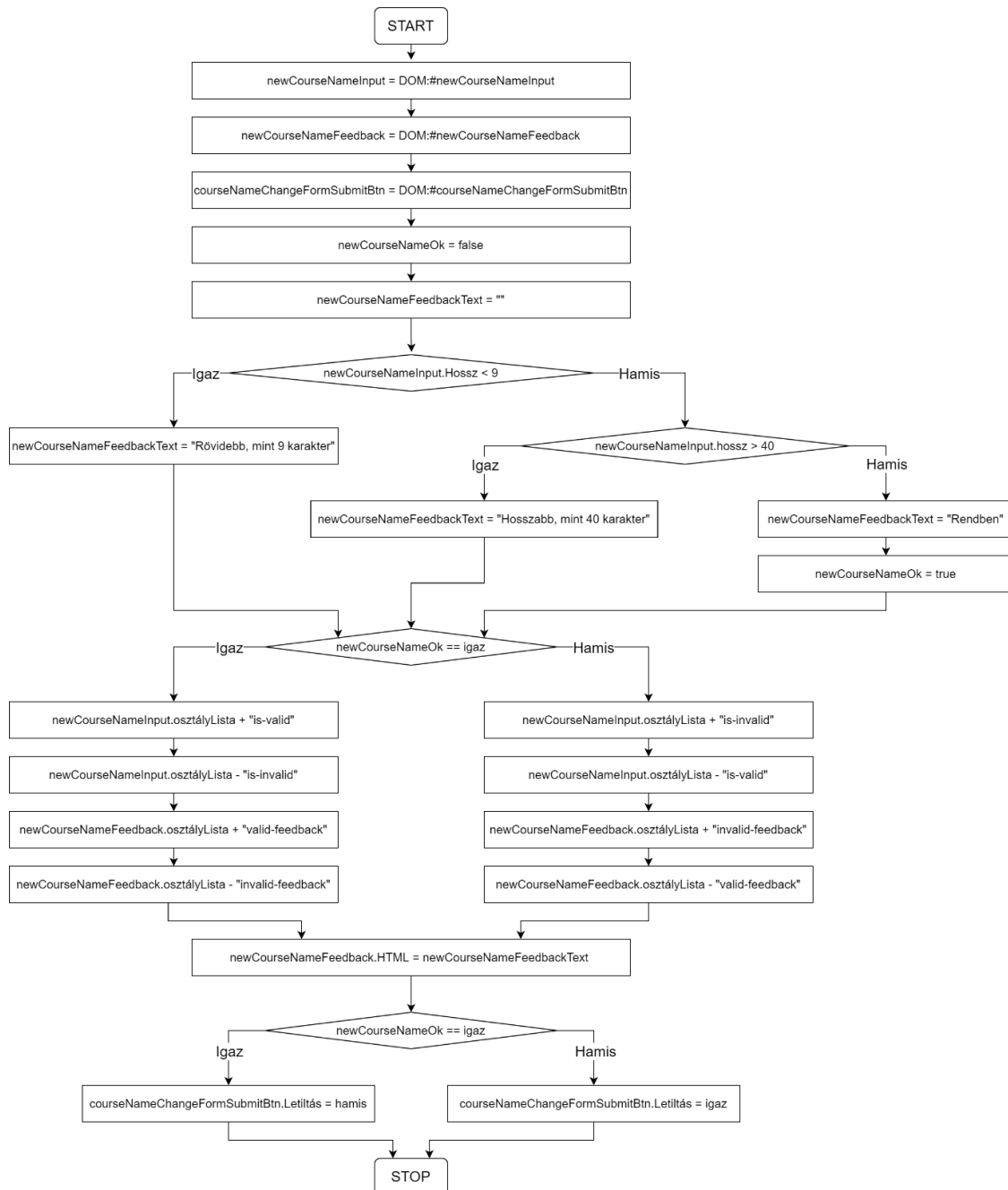
A webalkalmazás jelentős mennyiségű függvényt vagy metódust tartalmaz, ugyanis a csoportok, résztvevők és felhasználók adminisztrálása sok esetben komplex folyamatokat igényel. A megtalálható űrlapok és interakciók kiértékelése két szinten történik: az egyik a felhasználót segítő kliensoldali validáció, a másik pedig a helyes adatok tárolását garantáló szerveroldali ellenőrzés. Előbbi biztonsági szempontból jelentéktelen, hiszen a kliens oldalon működő kódok és tartalmak manipulálhatók, ezért is szükséges minden módosítás, illetve adatküldés előtt a körülményeket szerveroldalon kiértékelni, és csak ezek megfelelésége esetén végrehajtani az adott folyamatot. A következőkben az említett folyamatok közül egy-egy kerül bemutatásra.

Kurzus nevének módosítását szolgáló űrlap kliensoldali validációja

Egy adott kurzus nevének több feltételnek is meg kell felelnie. A módosítást szolgáló űrlap egymezős, de hasonló elven működik a webalkalmazás többi űrlapjának jelentős része is, így ezen validáció logikája átültethető a többire is. Egyik fő cél volt, hogy a felhasználó a lehető leggyorsabban és legkönnyebben tudjon helyes adatot megadni. Így a validációs folyamat az űrlap beviteli mezőjében történő billentyűleütésként újra és újra lefut. Ha az adatok megfelelőek, az űrlap küldés gombja aktív lesz, ellenkező esetben pedig letiltásra kerül. A felhasználó valós idejű információt kap a beviteli mező alatt található visszajelző szövegből, illetve ezen elem és a mező színeiből. A hibát piros színnel és x-szel, míg a helyes adatot zöld színnel és pipával jelzi a felület. A színezés beállítása Bootstrap osztályok segítségével történik.

A kliensoldali validáció folyamata

Az űrlap küldés gombjának azonosítója „courseNameChangeFormSubmitBtn”, a beviteli mezőnek „newCourseNameInput” és a visszajelző szövegnek pedig „newCourseNameFeedback”. Az ellenőrző JavaScript függvény neve: validateCourseNameChangeForm().



3. ábra - Kliensoldali űrlapvalidáció folyamata

Kliensoldali validáció folyamatának forráskódja

```

const validateCourseNameChangeForm = () => {
  const newCourseNameInput = document.getElementById('newCourseNameInput');
  const newCourseNameFeedback = document.getElementById('newCourseNameFeedback');
  const courseNameChangeFormSubmitBtn =
    document.getElementById('courseNameChangeFormSubmitBtn');

  let newCourseNameOk = false;
  let newCourseNameFeedbackText = '';
  if(newCourseNameInput.value.length < 9) {
    newCourseNameFeedbackText = 'Rövidebb, mint 9 karakter';
  } else if(newCourseNameInput.value.length > 40) {
    newCourseNameFeedbackText = 'Hosszabb, mint 40 karakter';
  } else {
    newCourseNameFeedbackText = 'Rendben';
    newCourseNameOk = true;
  }
  if(newCourseNameOk) {
    newCourseNameInput.classList.add('is-valid');
    newCourseNameInput.classList.remove('is-invalid');
    newCourseNameFeedback.classList.add('valid-feedback');
    newCourseNameFeedback.classList.remove('invalid-feedback');
  } else {
    newCourseNameInput.classList.add('is-invalid');
    newCourseNameInput.classList.remove('is-valid');
    newCourseNameFeedback.classList.add('invalid-feedback');
    newCourseNameFeedback.classList.remove('valid-feedback');
  }
  newCourseNameFeedback.innerHTML = newCourseNameFeedbackText;

  if(newCourseNameOk) {
    courseNameChangeFormSubmitBtn.disabled = false;
  } else {
    courseNameChangeFormSubmitBtn.disabled = true;
  }
}

```

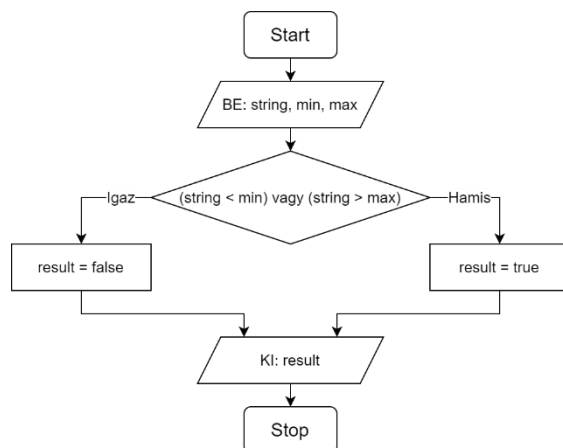
1. forráskód - Kliensoldali űrlapvalidáció JavaScript forráskód részlete

Karakterláncok hosszellenőrzése szerveroldalon

Korábban már említésre került, hogy a kliensoldali validációk csak a felhasználói élményt javítják. Az adatok szerveroldalon mindig kiértékelésre kerülnek, mielőtt használatba vagy eltárolásra kerülnének. A szerveroldali folyamatok szkriptjei PHP nyelven OOP szemlélet szerint kerültek megírásra. Mivel az alkalmazásban több űrlap található, és ezekről javarészt karakterláncok érkeznek szerveroldalra, ezért a különböző karakterlánc ellenőrzések és műveletek számára készült egy külön objektum-osztály „StringManager” néven. Ennek egyik leggyakrabban igénybe vett metódusa a karakterláncok hosszáért felelős „stringLengthCheck” nevű tagfüggvény. Adattáblákba rögzítése előtt fontos, hogy a tárolni kívánt adat megfelel-e az adott mező hossz tulajdonságának.

Karakterlanc hosszellenőrzésének szerveroldali folyamata

Az ellenőrizendő karakterlanc bemeneti paraméterének neve „string”, az elfogadható legkisebb egész karakterszáma „min”, a legnagyobb pedig „max”. A folyamat annak függvényében, hogy a bemeneti karakterlanc értékének hossza megfelelő, logikai értékkel tér vissza.



4. ábra - Karakterlanc hosszellenőrzésének folyamata

Ezen metódus bármilyen más folyamatban hasznos lehet, ahol egy karakterlanc hosszára követelmények vonatkoznak.

Karakterlanc hosszellenőrző folyamatának forráskódja

```

public function stringLengthCheck(string $string, int $min, int $max) {
    if(strlen($string) < $min || strlen($string) > $max) {
        $result = false;
    } else {
        $result = true;
    }
    return $result;
}
    
```

2. forráskód - Karakterlanc hosszellenőrzésének PHP forráskód részlete

Tesztelési dokumentáció

Egy alkalmazás használata során felmerülhetnek fejlesztői és felhasználói hibák egyaránt. Előbbire jó példa, ha a program valamely funkciója nem tud lefutni vagy helytelen eredményt produkál. A felhasználó pedig akkor tud hibát véteni, ha a tőle várt adatokra vonatkozó valamely követelmény nem teljesül, melyre az egyszerű példa egy helytelen jelszó vagy felhasználónév. Magát a programot az esetleges véletlen vagy szándékos felhasználói hibákra fel kell készíteni, ennek elmaradása vagy hiányos megvalósítása a fejlesztői hibák közé sorolható.

A fejlesztés során és végén a hibák feltárásának állandó módszere a tesztelés, melynek módszereit két csoportba sorolhatjuk: feketedobozos és fehérdobozos. Feketedobozos tesztelés során a tesztelő magát a programot vizsgálja használat közben, annak felületén keresztül, tehát ez egyfajta felhasználó jellegű folyamat. Fehérdobozos tesztelés esetén rendelkezésre áll az alkalmazás forráskódja is, melynek egyes strukturális elemeire fókuszál az ellenőrzés.

A webalkalmazás minden folyamatát szükséges volt tesztelni próbafelhasználók segítségével, hogy kiderüljön, a program funkciói helyesen működnek-e, és minden hibás kliensoldali viselkedést megfelelően kezel-e. A következőkben ezen esetekből egy-egy kerül bemutatásra.

Új résztvevő rögzítése

Új résztvevő rögzítésekor a felhasználónak 4 adatot kell helyesen megadnia az azokra vonatkozó követelmények betartásával: az új résztvevő teljes nevét, e-mail címét, születési helyét és idejét. Amennyiben valamelyik adat helytelenül kerül kitöltésre, az űrlap elküldésére szolgáló „Létrehozás” gomb letiltásra kerül, illetve az adott mező alatti visszajelző szöveg is pirosan jelzi a probléma pontos okát^(14. képernyőkép).

14. képernyőkép - Új résztvevő létrehozására szolgáló űrlap hibás kitöltés esetén

Természetesen a kliensoldali tartalmak manipulálhatók. Például az elküldésre szolgáló button elem disabled attribútuma felhasználói oldalon törölhető, vagy az e-mail címhez tartozó input elem típusa átírható. Tehát kliensoldalon kikényszeríthető, hogy hibás adatok kerüljenek továbbítása a szerveroldalra. A webalkalmazás a fejlesztése során az ilyen jellegű manipulációkra fel lett készítve. Ha hibás adatok kerülnek küldésre egy módosított űrlap segítségével, a folyamat nem fog lefutni, mert az adatok ezen az oldalon is ellenőrzésre kerülnek, és amennyiben hiba merül fel, a folyamat leáll, hibaüzenetet küldve vissza a kliens számára^(15. képernyőkép).

15. képernyőkép - Manipulált űrlappal küldött hibás adatokra kapott hibaüzenet

Új kurzusbeosztás rögzítése

A felhasználónak ahhoz, hogy egy kurzus alkalmainak ütemezését elindíthassa, be kell állítani annak heti beosztását. Egy csoporthoz bármennyi beosztásintervallum felvehető. Arra kell viszont figyelni, hogy azok nem ütközhetnek, azaz nem lehetnek egymással átfedésben. Ennek

ellenőrzése szerveroldalon történik. Ha a felhasználó olyan intervallumot próbál rögzíteni, ami már ütközik egy másikkal, akkor a szerver nem hajtja végre a rögzítést, hanem válaszként hibaüzenetet küld vissza a kliensoldal kérésére^(16. képernyőkép).

Akkor is ugyanez történik, ha a felhasználó véletlenül későbbi kezdőidőpontot állít be, mint a befejezés időpontja.

Tesztelés során felmerült hiba és javítása

Az alkalmazás tesztelésének során felmerült egy hibajelenség a grafikus heti beosztás működésével kapcsolatban. A hetek léptetésével egyes napok esetében egymás mellé két ugyanolyan dátum került megjelenítésre, és így a naptár elcsúszott egy nappal. Ennek oka az volt, hogy betöltéskor a kezdőnap az adott hét hétfőjének 0 óra 0 perce volt. A dátumokkal való műveleteket Unix formátumú egész szám időbélyegekkel végzi az alkalmazás, tehát amikor előre- vagy visszaléptet a felhasználó, akkor 1 hétnek megfelelő egész szám kerül hozzáadásra vagy kivonásra az aktuális hétfő időpontjából. A megjelenítés előtt viszont lokális dátumformátumra kerülnek átkonvertálásra a napok dátumai. A duplázódó dátumok pedig az óraátállítások következtében jöttek létre. Mivel a hét kezdődátumát 0 óra 0 perces időponttal kezelte az alkalmazás, visszaállításkor a következő nap is ugyanazon dátum 23 óra 0 perce lett. A probléma megoldása egyszerű volt, a kezdőidőpont 0 órától 4 órára lett módosítva, így akár előre, akár vissza kerül állításra a lokális óra, dátumbéli elcsúszás már nem fog bekövetkezni.

Beosztás			
Kezdés		Befejezés	
Hétfő ▾	14:00 ⌚	Hétfő ▾	17:15 ⌚
<div style="background-color: #f8d7da; padding: 10px; text-align: center;"> <p>Az új időpontot nem sikerült rögzíteni!</p> <p>A kezdésnek korábbinak kell lennie, mint a befejezésnek.</p> <p>Az új időpont nem ütközhet a meglévőkkel.</p> </div>			
Hétfő	17:00:00	Hétfő	20:15:00
Szerda	17:00:00	Szerda	20:15:00

16. képernyőkép - Hibaüzenet, egymással átfedésben lévő beosztások rögzítésekor

Továbbfejlesztési lehetőségek

Felhasználói üzemeltetés

A webalkalmazás jelenleg egy szolgáltató adatait tudja kezelni, a felhasználói fiókok mind ugyanazokra az adatokra látnak rá. Ebből következik, hogy az esetleges új felhasználóknak rendelkezniük kell saját szerverrel, illetve tárhellyel, melyre külön kell konfigurálni a programot. Tehát jelenleg az alkalmazás forráskóddal együtt lenne értékesíthető. Érdemes lenne viszont átalakítani úgy, hogy az alkalmazás csak a fejlesztő szerverén üzemeljen és a felhasználók egymástól függetlenül kezelhessék az adataikat, illetve ezek védve is legyenek egymástól. Így az üzleti modell átalakítható lenne egyfajta egyszerű előfizetési szolgáltatássá, ahol az új ügyfélnek annyi lenne a dolga, hogy regisztrál és előfizet, majd használja a webalkalmazást. Ennek megvalósításához szükség lenne az adatbázis szerkezet módosítására, hiszen kezelni kellene azt is, hogy egy-egy résztvevő vagy kurzus mely felhasználóhoz tartozik. Ezen változtatás természetesen a webalkalmazás folyamataira is hatással lenne. Például időpontütközés vizsgálatának során csak az adott felhasználó időpontjait lenne szükséges vizsgálni. Egy ilyen előfizetési rendszer pedig igényt formálna arra, hogy az alkalmazáson belül lehessen fizetni, viszont ezáltal már a felhasználókkal való szerződéskötés is a webalkalmazáson belül történne, mely miatt el kellene helyezni az oldalon az általános szerződési feltételeket és az adatkezelési tájékoztatót.

Fejlettebb levélcímzett kezelés

Jelenleg a webalkalmazásban levelet küldeni egy résztvevőnek vagy egy csoport tagjainak lehet. Ezt hasznos lenne oly módon továbbfejleszteni, hogy a címzettlista személyre szabható legyen. Ezt egy lebegő listával lehetne megvalósítani, ahonnan bármelyik aloldalról indítható lehetne a levélküldés. Erre a listára tetszőleges számú résztvevő és csoport is felvehető lenne, akár együttesen is.

Felhasznált irodalmak és források

Könyvek

Nagy Gusztáv: Web programozás alapismeretek – Ad Librum Kiadó, Budapest, 2011

George Schlossnagle: PHP fejlesztés felsőfokon – Kiskapu Kft., Budapest, 2004

Online dokumentációk

W3Schools – <https://www.w3schools.com/>

Algoritmusleírások – http://progalap.elte.hu/downloads/seged/eTananyag/lecke4_lap1.html

Webalkalmazás modulok

Bootstrap 5 – <https://getbootstrap.com/docs/5.0/about/license/>

PHPMailer – <https://github.com/PHPMailer/PHPMailer/blob/master/LICENSE>

Ábra- és képjegyzék

Képernyőképek

1. KÉPERNYŐKÉP - BEJELENTKEZÉSI ŰRLAP	4
2. KÉPERNYŐKÉP - LEGÖRDÜLŐ MENÜSOR	5
3. KÉPERNYŐKÉP - FELHASZNÁLÓK ADMINISZTRÁCIÓS FELÜLETE.....	5
4. KÉPERNYŐKÉP - ÚJ SZÜNET RÖGZÍTÉSÉT SZOLGÁLÓ ŰRLAP.....	7
5. KÉPERNYŐKÉP - RÖGZÍTETT SZÜNETEK LISTÁJA	7
6. KÉPERNYŐKÉP - ÚJ KURZUS LÉTREHOZÁSÁRA ÉS A RÖGZÍTETT KURZUSOK ÁTTEKINTÉSÉRE SZOLGÁLÓ DOBOZOK.....	8
7. KÉPERNYŐKÉP - KURZUS ALAPADATAINAK ÉS A BEOSZTÁSAINAK KEZELÉSÉT SZOLGÁLÓ DOBOZOK	8
8. KÉPERNYŐKÉP - KURZUS ÜTEMEZÉSÉT ÉS ALKALMAINAK MEGJELENÍTÉSÉT SZOLGÁLÓ DOBOZOK	9
9. KÉPERNYŐKÉP - ÚJ RÉSZTVEVŐ RÖGZÍTÉSÉT SZOLGÁLÓ DOBOZ.....	10
10. KÉPERNYŐKÉP - RÉSZTVEVŐK ÁTTEKINTÉSÉT ÉS KERESÉSÉT SZOLGÁLÓ DOBOZ.....	11
11. KÉPERNYŐKÉP - RÉSZTVEVŐ CSOPORTBA SOROLÁSÁT SZOLGÁLÓ FELUGRÓ ABLAK.....	11
12. KÉPERNYŐKÉP - LEVÉLKÜLDŐ FELÜLET	12
13. KÉPERNYŐKÉP - HETI BEOSZTÁS GRAFIKUS MEGJELENÍTÉSE A FŐOLDALON	13
14. KÉPERNYŐKÉP - ÚJ RÉSZTVEVŐ LÉTREHOZÁSÁRA SZOLGÁLÓ ŰRLAP HIBÁS KITÖLTÉS ESETÉN	28
15. KÉPERNYŐKÉP - MANIPULÁLT ŰRLAPPAL KÜLDÖTT HIBÁS ADATOKRA KAPOTT HIBAÜZENET.....	28
16. KÉPERNYŐKÉP - HIBAÜZENET, EGYMÁSSAL ÁTFEDÉSBEN LÉVŐ BEOSZTÁSOK RÖGZÍTÉSEKOR.....	29

Ábrák

1. ÁBRA - ADATBÁZISOK SZERKEZETE.....	20
2. ÁBRA - UML OSZTÁLYDIAGRAM.....	22
3. ÁBRA - KLIENSOLDALI ŰRLAPVALIDÁCIÓ FOLYAMATA	25
4. ÁBRA - KARAKTERLÁNC HOSSZELLENŐRZÉSÉNEK FOLYAMATA	27

Forráskódok

1. FORRÁSKÓD - KLIENSOLDALI ŰRLAPVALIDÁCIÓ JAVAScript FORRÁSKÓD RÉSZLETE	26
2. FORRÁSKÓD - KARAKTERLÁNC HOSSZELLENŐRZÉSÉNEK PHP FORRÁSKÓD RÉSZLETE	27