

ADATBÁZISÚ RENDSZEREK KÖTELEZŐ PROGRAM ETR

A CSAPATON BELÜL A MUNKA FELOSZTÁSÁNAK RÉSZLETES LEÍRÁSA

A csapat 3 emberből áll, név szerint: Német Tímea Kata, Vass Levente, és Subicz Máté István.

A munka részletes felosztása az első mérföldkőre:

Német Tímea Kata

- **Adatmodellezés és relációs adatelemzés**

Ebbe a témakörbe tartozik az E-K diagram átnézése és leképezése, a kapott formák normalizálása és az adattáblák leírása.

- **Funkciómeghatározás**

Egyed-esemény mátrixszal vagy szerep-funkció mátrixszal is megvalósítható.

- **Adatbázissal kapcsolatos követelmények:**

Összefüggő és konszolidált adattáblák (resztvesz, kurzus, tárgy, vizsga)

A feltöltött adatrekordok (resztvesz, kurzus, tárgy, vizsga)

Triggerek (trg_Kurzus_dates, trg_Kurzus_letszam)

Az alkalmazásban használt, adatbázisban tárolt eljárások és függvények

Vass Levente

- **Adatfolyam-diagramok**

Logikai és Fizikai formában is megvalósításra kerül, legalább 1. és 2. szinten.

- **Egyedmodell**

E-K diagram formájában.

- **Adatbázissal kapcsolatos követelmények:**

Összetett lekérdezések

Összefüggő és konszolidált adattáblák (Hallgato, Oktato, Adminisztrátor, Felhasznalo, Vizsgazik)

A feltöltött adatrekordok (Hallgato, Oktato, Adminisztrátor, Felhasznalo, Vizsgazik)

Triggerek (trg_Szak_unique_szakid)

Az alkalmazásban használt, adatbázisban tárolt eljárások és függvények

Subicz Máté István

- **A csapaton belül a munka felosztásának részletes leírása**

Feladatok rögzítése.

- **Követelménykatalógus**

Ez tartalmazza a feladat részletes leírását, meghatározza, hogy milyen funkciókkal tervezett a program megvalósítása.

- **Programhoz eszközök leírása**

Magában foglalja a programot, nyelvet, környezetet, és egyéb modulokat vagy dependenciákat.

- **Adatbázissal kapcsolatos követelmények:**

Összefüggő és konszolidált adattáblák (Uzenet, Szak, Terem, vizsgaztat, tart)

A feltöltött adatrekordok (Uzenet, Szak, Terem, vizsgaztat, tart)

Triggerek (trg_Resztvesz_noveles)

Az alkalmazásban használt, adatbázisban tárolt eljárások és függvények

SZÖVEGES, RÉSZLETES FELADATLEÍRÁS, KÖVETELMÉNYKATALÓGUS (MI A MEGVALÓSÍTANDÓ CÉL, MILYEN FUNKCIÓKKAL KELL RENDELKEZNI A PROGRAMNAK)

Cél az első mérföldkőre

Az első mérföldkőig a tervezésen lesz a hangsúly. Dokumentálásra kerülnek a folyamatok, Elkészülnek a diagramok, Összeáll, hogy hogyan is fog kinézni a program.

Követelménykatalógus

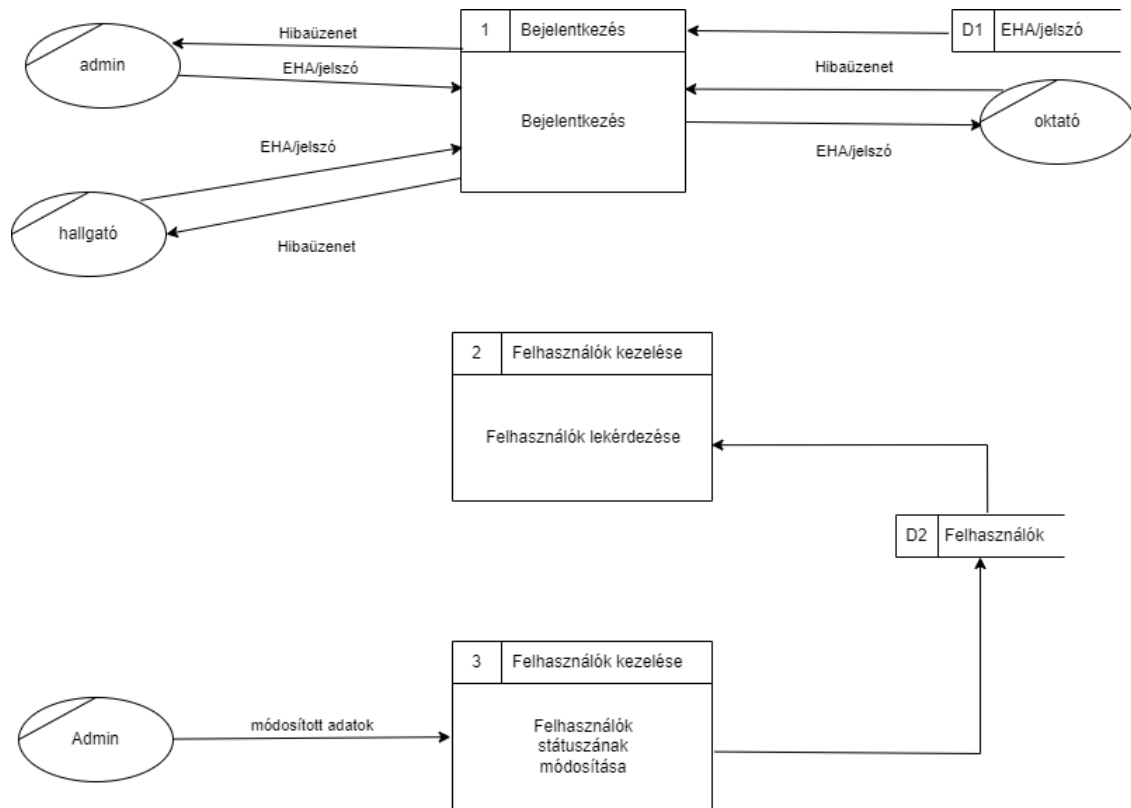
A program php-val, oracle-t használva készül.

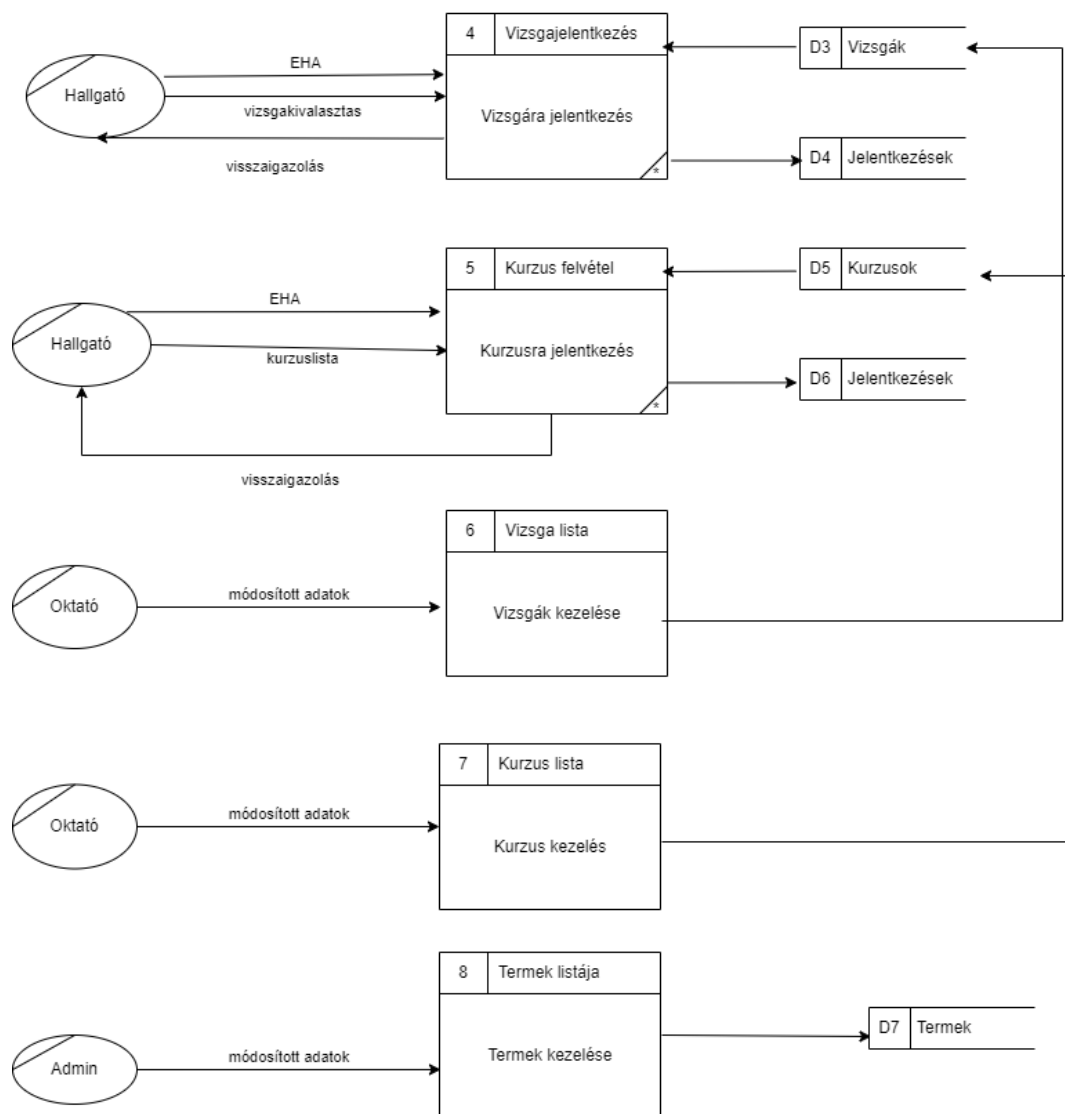
A rendszer az alábbi funkciókkal kell rendelkezzen:

1. Tanulók, Oktatók regisztrálása, bejelentkezése
2. Termek, kurzusok, vizsgák listázása
3. Kurzusra, vizsgára jelentkezés, Létrehozás
4. Saját profil megtekintése

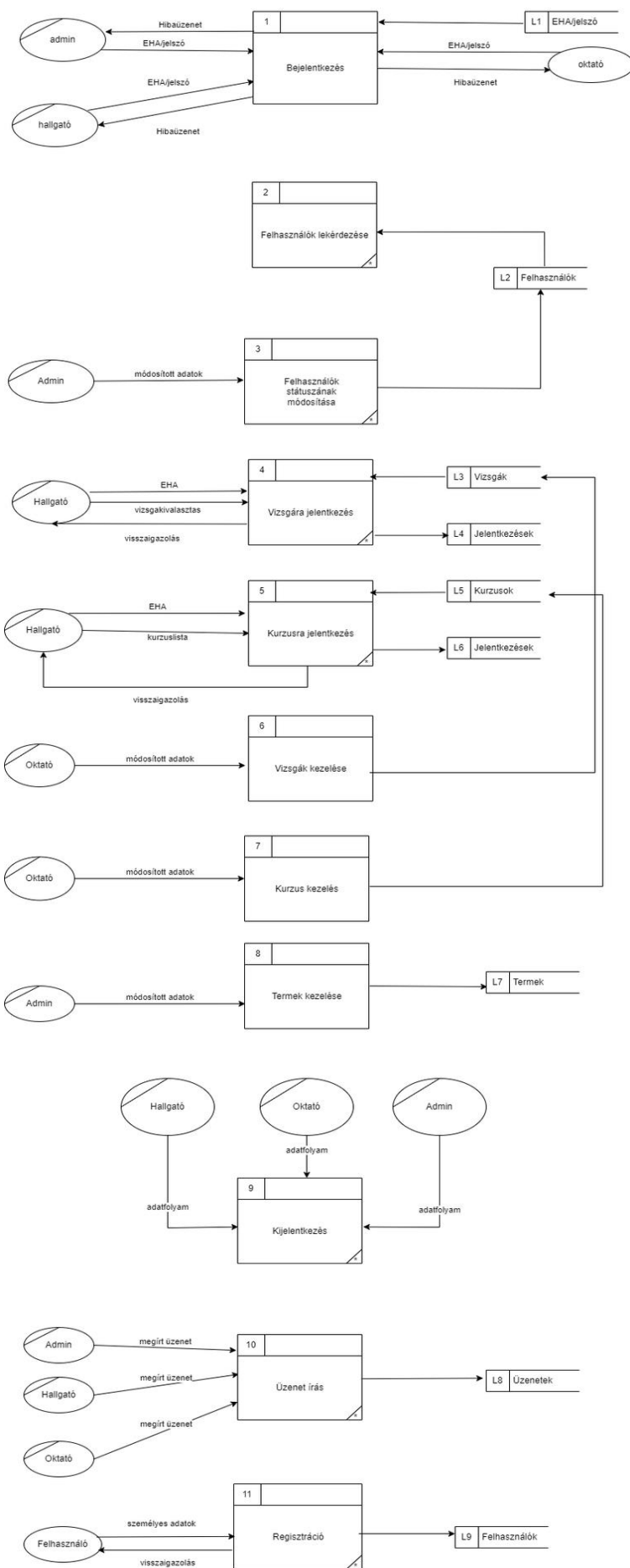
5. Fórum oldal, üzenetírás

ADATFOLYAM-DIAGRAMOK (LOGIKAI ÉS FIZIKAI ADATFOLYAM-DIAGRAM LEGALÁBB 1. ÉS 2. SZINTEN) / 2. EGYEDMODELL, E K DIAGRAM

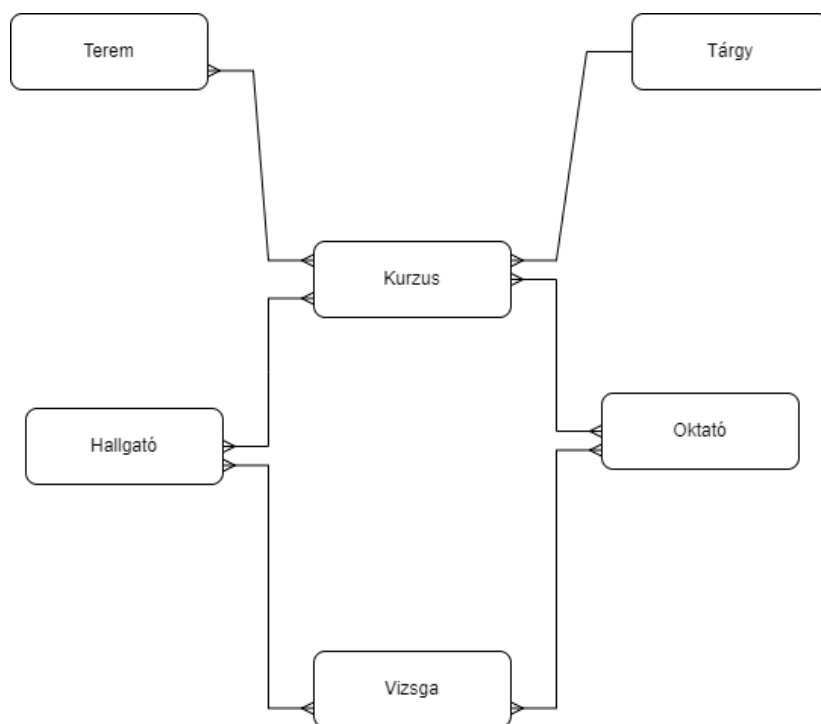




1. Fizikai adatfolyamdiagram



3. Logikai adatfolyamdiagram



4. Egyedmodell

ADATMODELLEZÉS ÉS RELÁCIÓS ADATELEMZÉS (E-K DIAGRAM, E-K DIAGRAM LEKÉPEZÉSE, NORMALIZÁLÁS, ADATTÁBLÁK LEÍRÁSA)

1NF

Felhasználó(EHA kód, Név, Jelszó)
 Hallgató(EHA kód, tanulmányi_átlag, súlyozott_átlag)
 Szak(EHA kód, szak)
 Oktató(EHA kód, beosztás)
 Adminisztrátor(EHA kód)
 Üzenet(EHA kód, tartalom, dátum)
 Kurzus(kód, név, kredit, maxfő, időpont, szemeszter, nyitott)
 Terem(név, gépes, férőhely)
 Tárgy(tárgykód, név, ajánlott_félév, óraszám,)
 Előfeltétel(tárgykód, előfeltétel)
 Vizsga(azonosító, típus, kezdet, vég, helyszín, maxfő, létszám)

Résztvesz(érdemjegy, hallgató.EHA kód, kurzus.kód)
 Tart(oktató.EHA kód, kurzus.kód)
 Tartozik(terem.név, kurzus.kód)
 Tartozik(tárgy.tárgykód, kurzus.kurzus kód)
 Vizsgázik(vizsga.azonosító, hallgató.EHA kód, érdemjegy)
 Vizsgáztat(vizsga.azonosító, oktató.EHA kód)

2NF/3NF

Felhasználó(EHA kód, Név, Jelszó)
 Hallgató(EHA kód, tanulmányi_átlag, súlyozott_átlag)
 Szak(EHA kód, szak)
 Oktató(EHA kód, beosztás)

Adminisztrátor(EHA kód)

Üzenet(EHA kód, tartalom, dátum)

Kurzus(kód, név, kredit, maxfő, időpont, szemeszter, nyitott, terem.név, tárgy.kód)

Terem(név, gépes, férőhely)

Tárgy(tárgykód, név, ajánlott_félév, óraszám,)

Előfeltétel(tárgykód, előfeltétel)

Vizsga(azonosító, típus, kezdet, vég, helyszín, maxfő, létszám)

Résztvesz(éremjegy, hallgató.EHA kód, kurzus.kód)

Tart(oktató.EHA kód, kurzus.kód)

Vizsgázik(vizsga.azonosító, hallgató.EHA kód, éremjegy)

Vizsgáztat(vizsga.azonosító, oktató.EHA kód)

FUNKCIÓMEGHATÁROZÁS EGYED-ESEMÉNY MÁTRIX SZEREP-FUNKCIÓ MATRIX

Funkciók meghatározása

Felhasználó

- **Felhasználó létrehozása**
 - Felhasználói szerepek: Adminisztrátor
 - Leírás: Az adminisztrátor jogosult új felhasználó létrehozására a rendszerben.
- **Felhasználó módosítása**
 - Felhasználói szerepek: Adminisztrátor
 - Leírás: Az adminisztrátor jogosult meglévő felhasználó adatainak módosítására a rendszerben.
- **Felhasználó törlése**
 - Felhasználói szerepek: Adminisztrátor
 - Leírás: Az adminisztrátor jogosult meglévő felhasználó törlésére a rendszerből.

Kurzus

- **Kurzus létrehozása**
 - Felhasználói szerepek: Adminisztrátor, Oktató
 - Leírás: Az adminisztrátor és az oktató jogosult új kurzus létrehozására a rendszerben.
- **Kurzus módosítása**
 - Felhasználói szerepek: Adminisztrátor, Oktató
 - Leírás: Az adminisztrátor és az oktató jogosult meglévő kurzus adatainak módosítására a rendszerben.
- **Kurzus törlése**
 - Felhasználói szerepek: Adminisztrátor, Oktató
 - Leírás: Az adminisztrátor és az oktató jogosult meglévő kurzus törlésére a rendszerből.

Tárgy

- **Tárgy létrehozása**
 - Felhasználói szerepek: Adminisztrátor, Oktató
 - Leírás: Az adminisztrátor és az oktató jogosultságokkal rendelkezik a tárgyak létrehozására az alkalmazásban. A funkció célja, hogy a felhasználó létrehozassa az új tárgyakat az alkalmazásban.
- **Tárgy módosítása**
 - Felhasználói szerepek: Adminisztrátor, Oktató
 - Leírás: Az adminisztrátor és az oktató jogosultságokkal rendelkezik a tárgyak módosítására az alkalmazásban. A funkció célja, hogy a felhasználó módosíthassa a meglévő tárgyakat az alkalmazásban.
- **Tárgy törlése**
 - Felhasználói szerepek: Adminisztrátor, Oktató
 - Leírás: Az adminisztrátor és az oktató jogosultságokkal rendelkezik a tárgyak törlésére az alkalmazásban. A funkció célja, hogy a felhasználó törölje a meglévő tárgyakat az alkalmazásban.

Vizsga

- **Vizsga létrehozása**
 - Felhasználói szerepek: Adminisztrátor, Oktató
 - Leírás: Az adminisztrátor és az oktató jogosultságokkal rendelkezik a vizsgák létrehozására az alkalmazásban. A funkció célja, hogy a felhasználó létrehozassa az új vizsgákat az alkalmazásban.
- **Vizsga módosítása**
 - Felhasználói szerepek: Adminisztrátor, Oktató
 - Leírás: Az adminisztrátor és az oktató jogosultságokkal rendelkezik a vizsgák módosítására az alkalmazásban. A funkció célja, hogy a felhasználó módosíthassa a létező vizsgákat az alkalmazásban.
- **Vizsga törlése**
 - Felhasználói szerepek: Adminisztrátor, Oktató
 - Leírás: Az adminisztrátor és az oktató jogosultságokkal rendelkezik a vizsgák törlésére az alkalmazásban. A funkció célja, hogy a felhasználó törölhesse a létező vizsgákat az alkalmazásból.

Terem

- **Terem létrehozása**
 - Felhasználói szerepek: Adminisztrátor
 - Leírás: Az adminisztrátor jogosultságokkal rendelkezik a termek létrehozására az alkalmazásban. A funkció célja, hogy az adminisztrátor létrehozassa az új termeket az alkalmazásban.
- **Terem módosítása**
 - Felhasználói szerepek: Adminisztrátor
 - Leírás: Az adminisztrátor jogosultságokkal rendelkezik a termek módosítására az alkalmazásban. A funkció célja, hogy az adminisztrátor módosíthassa a létező termeket az alkalmazásban.
- **Terem törlése**
 - Felhasználói szerepek: Adminisztrátor
 - Leírás: Az adminisztrátor jogosultságokkal rendelkezik a termek törlésére az alkalmazásban. A funkció célja, hogy az adminisztrátor törölhesse a létező termeket az alkalmazásból.

Szak

- **Szak létrehozása**
 - Felhasználói szerepek: Adminisztrátor
 - Leírás: Az adminisztrátor jogosultságokkal rendelkezik a szakok létrehozására az alkalmazásban. A funkció célja, hogy az adminisztrátor létrehozhassa az új szakokat az alkalmazásban.
- **Szak módosítása**
 - Felhasználói szerepek: Adminisztrátor
 - Leírás: Az adminisztrátor jogosultságokkal rendelkezik a szakok módosítására az alkalmazásban. A funkció célja, hogy az adminisztrátor módosíthassa a szakokat az alkalmazásban.
- **Szak törlése**
 - Felhasználói szerepek: Adminisztrátor
 - Leírás: Az adminisztrátor jogosultságokkal rendelkezik a szakok törlésére az alkalmazásban. A funkció célja, hogy az adminisztrátor törölhesse a szakokat az alkalmazásból.

Üzenet

- **Üzenet küldése**
 - Felhasználói szerepek: Adminisztrátor, Oktató, Diák
 - Leírás: Az adminisztrátor, oktató és diák jogosultságokkal rendelkezik üzenet küldésére az alkalmazásban. A funkció célja, hogy a felhasználók egymás között kommunikálhassanak az alkalmazásban.

Egyedek / Események	Bejelentkezés	saját felhasználó módosítása	Kursus létrehozása	Kursus módosítás	Kursus jelentkezik	K. Jelentkezés elgondolás	Tárgy létrehozása	Tárgy módosítás	Tárgy jelentkezik	T. Jelentkezés elgondolás	Új Oktató létrehozása	Oktató módosítása	Hallgató létrehoz	Hallgató módosít	Adminisztrátor létrehoz	Adminisztrátor módosít
Adminisztrátor	O	MT			L				L		L	MT	L	MT	L	MT
Hallgató	O	MT														
Oktató	O	MT			O				O							
Kurszus			L	MT												
Tárgy				LT	O	L	L	MT	O	L						
Vizsga																
Terem																
Szak																
Üzenet																

Egyedek / Események	Vizsga létrehozása	Vizsga módosítása	Terem kioszt	Terem módosítása	Szak létrehozása	Szak módosítása	Üzenet küldése	Előfeltétel létrehozása	Előfeltétel módosítása
Adminisztrátor							LM		
Hallgató							LM		
Oktató							LM		
Kurszus									
Tárgy									
Vizsga	L	MT							
Terem			L	MT					
Szak					L	MT			
Üzenet							LM		

L	Létrehoz															
O	Olvas															
M	módosít															
T	Törlés															

	Regisztráció	Bejelentkezés	Üzenet	Kurszus létrehoz	Kurszus jelentkezik	Terem kioszt	Tárgy létrehoz	Kurszus résztvesz	Előfeltétel	Vizsga létrehoz	Vizsga résztvesz
Oktató	X	X	X	X					X	X	X
Hallgató	X	X	X		X			X			X
Adminisztrátor	X	X	X	X		X	X		X	X	

A FORDÍTÁSHOZ/FUTTATÁSHOZSZÜKSÉGES ESZKÖZÖK (HASZNÁLT PROGRAMNYELV, VERZIÓ,KÜLSŐ FÜGGVÉNYKÖNYVTÁRAK/MODULOK, ...)

Az alkalmazás futtatásához a következő eszközök és függőségek szükségesek:

Telepített Oracle adatbázis, ehhez a java jdk 11-es verziója
PHP 8.1
Apache webserver
Oci_connect 3.0

A MEGVALÓSÍTOTT ÖSSZETETT LEKÉRDEZÉSEK SZÁMA HELYE A PROGRAMBAN

Az alábbi lekérdezés visszaadja a tárgyak nevét és a hozzá tartozó kurzusok számát (targyak.php 35.sor):

```
SELECT tárgy.targy_kod,tárgy.nev,COUNT(kurzus.kod) "Kurzusok száma" FROM kurzus,tárgy  
WHERE kurzus.targy_kod=tárgy.targy_kod GROUP BY tárgy.nev,tárgy.targy_kod
```

Az alábbi lekérdezés visszaadja azokat a kurzusokat, amelyet a bejelentkezett felhasználó vett fel. Név szerint növekvő sorrendben (kurzusaim.php 41.sor):

```
SELECT kurzus.kod,kurzus.nev FROM kurzus WHERE kurzus.kod IN (SELECT kurzus_kod FROM  
resztvesz WHERE résztvesz.hallgato_aha_kod='$username') ORDER BY kurzus.nev")
```

Az alábbi lekérdezés visszaadja azokat a vizsgákat, amelyet a bejelentkezett felhasználó vett fel. Név szerint növekvő sorrendben (vizsgaim.php 40.sor):

```
SELECT vizsga.azonosito,vizsga.tipus,vizsga.kezdet,vizsga.veg,vizsga.teremnev,vizsga.letszam FROM  
vizsga WHERE azonosito IN (SELECT azonosito FROM vizsgazik WHERE eha_kod='$username')
```

Az alábbi lekérdezés visszaadja egy adott kurzuson lévő hallgatók EHA kódját, vezetéknévét, keresztnévét (kurzushallgatoi 39.sor):

```
SELECT  
resztvesz.kurzus_kod,hallgato.aha_kod,felhasznalo.vezetek,felhasznalo.keresztnév,resztvesz.erdemjegy  
FROM hallgato,resztvesz,felhasznalo WHERE hallgato.aha_kod=resztvesz.hallgato_aha_kod AND  
hallgato.aha_kod=felhasznalo.aha_kod AND résztvesz.kurzus_kod='$kurzuskód'
```

Az alábbi lekérdezés visszaadja a hallgatók eha kód, szak id, szaknév valamint a hallgatók átlagát (hallgatoListaz.php 34.sor):

```
SELECT hallgato.aha_kod,szak.szakid,szak.szaknev,hallgato.atlag FROM hallgato,szak WHERE  
szak.szakid=hallgato.szakid
```

Az alábbi lekérdezést visszaadja azokat a kurzusokat, amelyek a bejelentkezett oktatóhoz tartoznak (kurzusListaz 34.sor):

```
SELECT  
kurzus.kod,kurzus.nev,kurzus.kredit,kurzus.oraszam,kurzus.nap,kurzus.kezdet,kurzus.veg,kurzus.teremne  
v,kurzus.targy_kod,kurzus.zart FROM kurzus WHERE kurzus.kod IN (SELECT kurzus_kod FROM tart  
WHERE tart.oktato_aha_kod='$username') ORDER BY kurzus.nev
```

Az alábbi lekérdezés vissza adja egy kiválasztott vizsgán lévő hallgatókat (vizsgahallgatoi.php 41.sor):

```
SELECT
vizsgazik.azonosito,hallgato.eha_kod,felhasznalo.vezetek,felhasznalo.keresztnev,vizsgazik.erdemjegy
FROM hallgato,vizsgazik,felhasznalo WHERE hallgato.eha_kod=vizsgazik.eha_kod AND
hallgato.eha_kod=felhasznalo.eha_kod AND vizsgazik.azonosito='$kurzuskod'
```

Az alábbi lekérdezés vissza adja azokat a vizsgákat, amin a bejelentkezett felhasználó a vizsgáztató (vizsgaListaz.php 35.sor):

```
SELECT * FROM vizsga WHERE azonosito IN (SELECT azonosito FROM vizsgaztat WHERE
vizsgaztat.eha_kod='$username')
```

TRIGGEREK

A 'kurzus' táblához egy trigger, amely ellenőrzi, hogy a kurzus kezdeti és befejező dátuma helyes-e

```
CREATE OR REPLACE TRIGGER trg_Kurzus_dates
BEFORE INSERT OR UPDATE ON kurzus
FOR EACH ROW
BEGIN
IF :NEW.kezdet >= :NEW.veg THEN
RAISE_APPLICATION_ERROR(-20001, 'A kurzus befejező dátuma nem lehet korábbi vagy egyenlő a
kezdő dátumnál.');
```

A kurzus táblához egy trigger, amely ellenőrzi, hogy a terem férőhelyénél nagyobb e a hozzáadandó kurzus létszáma, ha igen, akkor dob egy kivételt:

```
CREATE OR REPLACE TRIGGER trg_Kurzus_letszam
BEFORE INSERT OR UPDATE ON kurzus
FOR EACH ROW
DECLARE
v_ferohely terem.fero_hely%TYPE;
v_letszam NUMBER;
BEGIN
SELECT fero_hely INTO v_ferohely FROM terem WHERE teremnev = :NEW.teremnev;
SELECT COUNT(*) INTO v_letszam FROM resztvesz WHERE kurzus_kod = :NEW.kod;

IF v_letszam > v_ferohely THEN
RAISE_APPLICATION_ERROR(-20005, 'A kurzus résztvevőinek száma meghaladja a terem
kapacitását. Kérlek, válassz másik termet vagy csökkentsd a résztvevők számát.');
```

A 'Szak' táblához egy trigger, amely ellenőrzi a szakid egyediségét

```
CREATE OR REPLACE TRIGGER trg_Szak_unique_szakid
BEFORE INSERT OR UPDATE ON szak
FOR EACH ROW
DECLARE
v_count NUMBER;
BEGIN
SELECT COUNT(*) INTO v_count FROM szak WHERE szakid = :NEW.szakid;

IF v_count > 0 THEN
RAISE_APPLICATION_ERROR(-20006, 'A megadott szakid már létezik. Kérlek, adj meg egy egyedi
szakid-t.');
```

-- A vizsga jelenlegi létszámát beállító trigger

```
CREATE OR REPLACE TRIGGER trg_Resztvesz_noveles
AFTER INSERT ON vizsgazik
FOR EACH ROW
BEGIN
UPDATE vizsga
SET letszam = letszam + 1
WHERE azonosito = (SELECT azonosito FROM vizsga WHERE azonosito = :NEW.azonosito);
END;
```

ELJÁRÁSOK, FÜGGVÉNYEK

-- 1. Jelentkezteti a diákot a kurzusra

```
CREATE OR REPLACE PROCEDURE diak_jelentkezés(
  p_aha_kod IN VARCHAR2,
  p_kod IN VARCHAR2
)
AS
BEGIN
  INSERT INTO Resztvesz (hallgato_aha_kod, kurzus_kod)
  VALUES (p_aha_kod, p_kod);

  UPDATE Kurzus
  SET jelentkezok_szama = jelentkezok_szama + 1
  WHERE kod = p_kod;
END;
```

-- 2. Tárgyon résztvevők száma

```
CREATE OR REPLACE PROCEDURE targy_resztvevok_szama(  
  p_targy_kod VARCHAR2,  
  p_resztvevok_szama OUT NUMBER  
)  
IS  
BEGIN  
  SELECT COUNT(*) INTO p_resztvevok_szama  
  FROM Resztvesz r  
  JOIN Kurzus k ON r.kurzus_kod = k.kod  
  WHERE k.targy_kod = p_targy_kod;  
END;  
/
```

-- 3. Diák kurzusról való lejelentkezése

```
CREATE OR REPLACE PROCEDURE diak_lejelentkezes(  
  p_aha_kod IN VARCHAR2,  
  p_kod IN VARCHAR2  
)  
AS  
BEGIN  
  DELETE FROM Resztvesz  
  WHERE hallgato_aha_kod = p_aha_kod AND kurzus_kod = p_kod;  
  
  UPDATE Kurzus  
  SET jelentkezo_kod_szama = jelentkezo_kod_szama - 1  
  WHERE kod = p_kod;  
  
  COMMIT;  
END;  
/
```

-- 4. Vizsgaszámot számít

```
CREATE OR REPLACE PROCEDURE vizsgak_szama(  
  p_vizsgak_szama OUT NUMBER  
)  
AS  
BEGIN  
  SELECT COUNT(*) INTO p_vizsgak_szama FROM Vizsga;  
END;  
/
```

-- 5. Tárgyhoz kurzusok száma

```
CREATE OR REPLACE PROCEDURE kurzusok_szama(  
  p_targy_kod VARCHAR2,  
  p_kurzusok_szama OUT NUMBER  
)  
AS  
BEGIN  
  SELECT COUNT(*) INTO p_kurzusok_szama FROM Kurzus WHERE targy_kod = p_targy_kod;  
END;  
/
```