

Adatbázis alapú rendszerek

2023-2024/2

IB152L-6

csütörtök 16:00-18:00

Murica

Elektronikus tanulmányi rendszer

Készítette:

Szobonya Dávid

Cakó Péter

Tandi Áron

Munka felosztása

- Logikai adatfolyam-diagram – Cakó Péter
- Fizikai adatfolyam-diagram – Szobonya Dávid
- Egyedmodell – Tandi Áron
- Egyed-kapcsolat diagram – Tandi Áron
- Egyed-kapcsolat diagram leképezése relációs sémákká – Tandi Áron
- Funkcionális függések felírása és relációs sémák normalizálása 3NF-ig – Tandi Áron
- Egyed-esemény matrix – Cakó Péter
- Szerep-funkció matrix – Szobonya Dávid, Cakó Péter
- Funkcióleírás – Cakó Péter
- Képernyőtervek – Szobonya Dávid
- Menütervek – Szobonya Dávid

Értékelési mód:

Csapat

A csoportok az alábbi két értékelési módból választhatnak:

- A csoport tagjai közösen dolgoznak, minden tag ugyanannyi pontot kap. (Csapat)
- A csoport tagjai felosztják a munkát, értékelés egyénenként. (Egyéni)

Feladat szöveges leírása

A Murica egy elektronikus tanulmányi rendszer, ami követi az egyes szakokat, azokhoz tartozó tárgyakat, kurzusokat, vizsgákat, a hallgatókat és azok teljesítményét. Külön felületet biztosít a hallgatóknak, oktatóknak és adminisztrátoroknak a tanév zökkenőmentes lezajlásához.

Követelménykatalógus

Funkcionális követelmények

Tárolandó adatok:

- **Szak**
 - Megnevezés: rövid szöveg
 - Típus: bsc/msc
 - Előrelátott tartam: hány félév
- **Felhasználó**
 - Kód: 6 karakteres kód, betűket vagy számokat tartalmaz, kis-/nagybetű nem számít
 - Név
 - Jelszó
 - E-mail
 - Születési idő

Egy felhasználó lehet hallgató is. Ebben a szerepben a következő plusz adatokat kapja meg:

- Szak
- Kezdés éve

Hallgatóként felvehet kurzusokat és amennyiben az előadás, az azokhoz tartozó vizsgákat. A felvett kurzusoknak eltároljuk az érdemjegyet is.

Egy felhasználó lehet oktató is, ha van olyan kurzus ahol oktatóként van megjelölve. Ilyenkor létrehozhat vizsgákat, jegyeket írhat be az adott kurzust felvett hallgatóknak.

Egy felhasználó lehet admin is, ekkor hozhat létre tárgyakat, kurzusokat és rendelhet hozzájuk oktatókat, szakokat, tantervet és szerkesztheti azokat. Ő vehet fel további felhasználókat is, valamint módosíthatja az adataikat.

- **Üzenet**

- Címzett
- Tárgy
- Tartalom
- Idő

- **Tárgy**

- Kód
- Név
- Mennyi kreditet ér
- Gyakorlat vagy előadás

Egy tárgy több kurzust foglal össze. Egy tárgyhoz tartozó kurzust csak akkor lehet felvenni.

- **Kurzus**

- Sorszám → Ebből jön ki a kódja: [Tárgy kód]-[Kurzus sorszám]
- Melyik tárgyhoz tartozik
- Terem
- Időpont
- Férőhely
- Melyik félévhez tartozik

- **Terem**

- Kód
- Férőhely

- **Vizsga**

- Melyik tárgyhoz tartozik
- Melyik teremben lesz
- Időpont
- Vizsgáztató(k)
- Férőhely

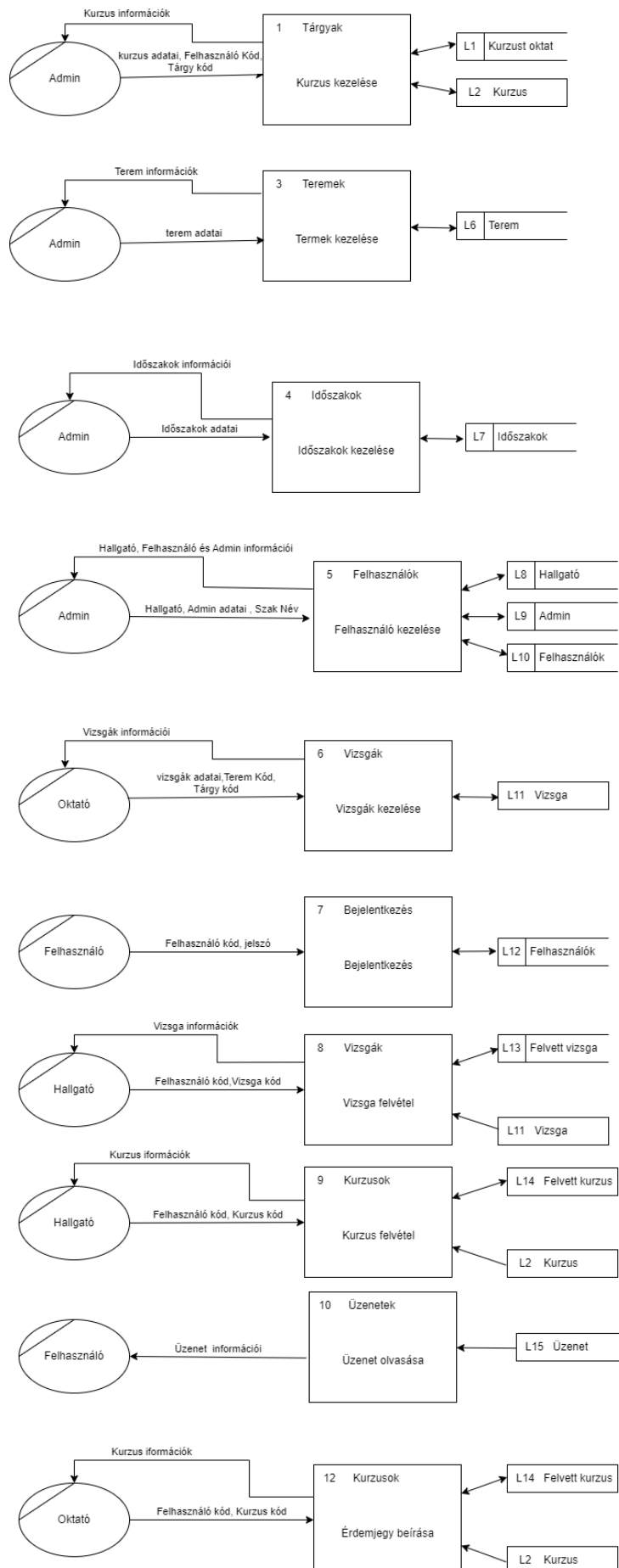
Nem funkcionális követelmények

- A jelszavak biztonságosan vannak eltárolva
- Input mezők kezelése biztonságos

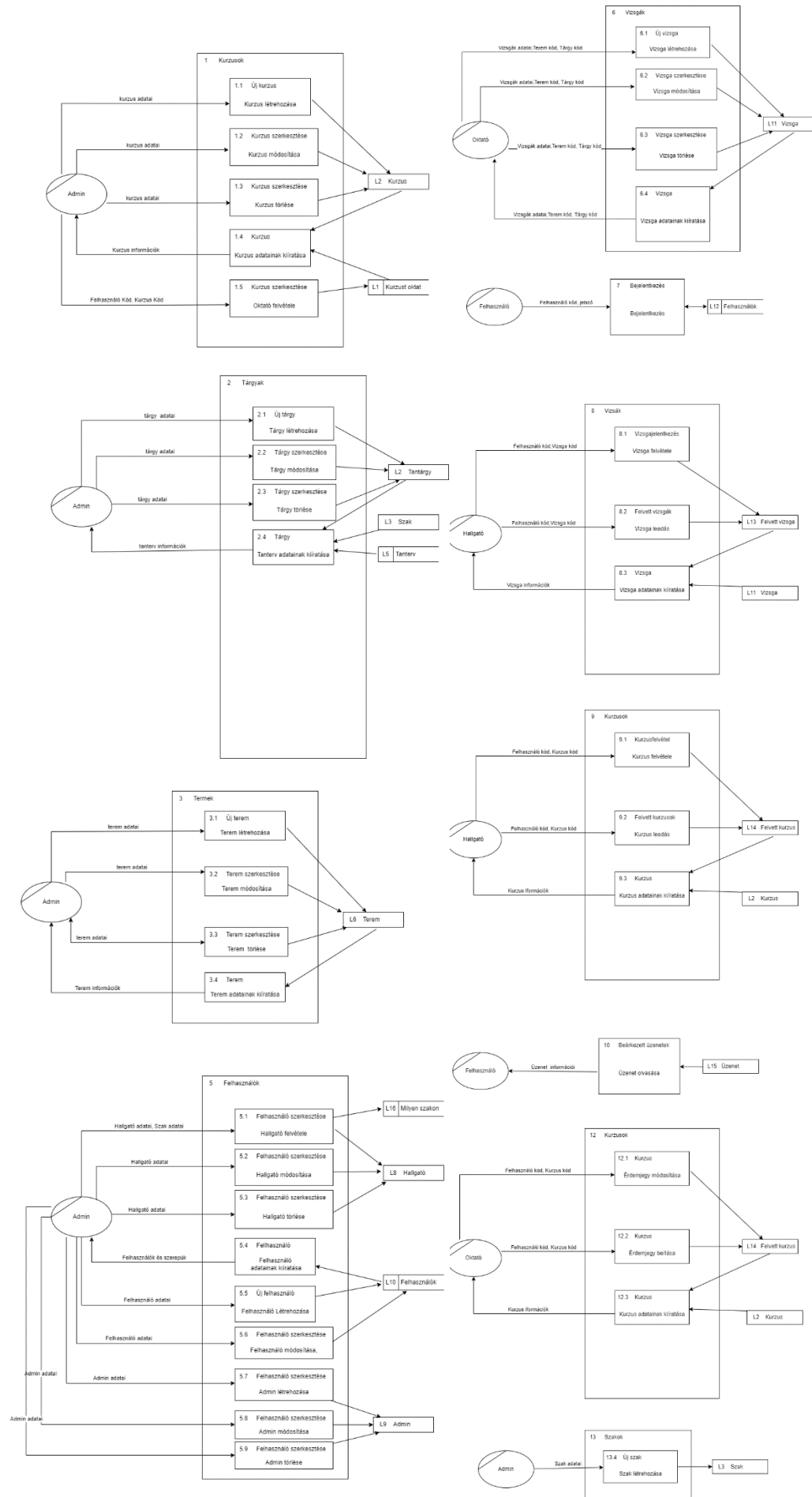
Adatfolyam diagram (DFD):

Fizikai

A DFD 1. szintje:

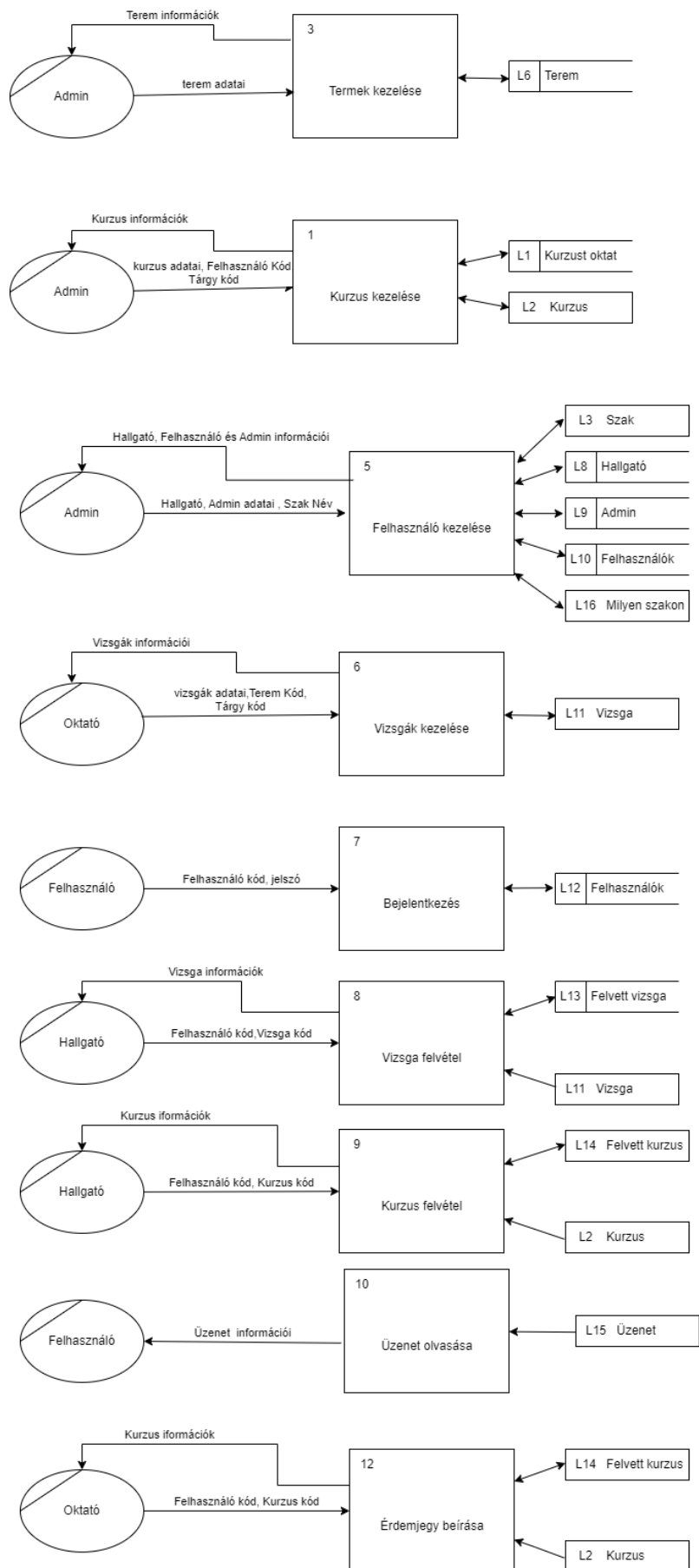


A DFD 2. szintje:

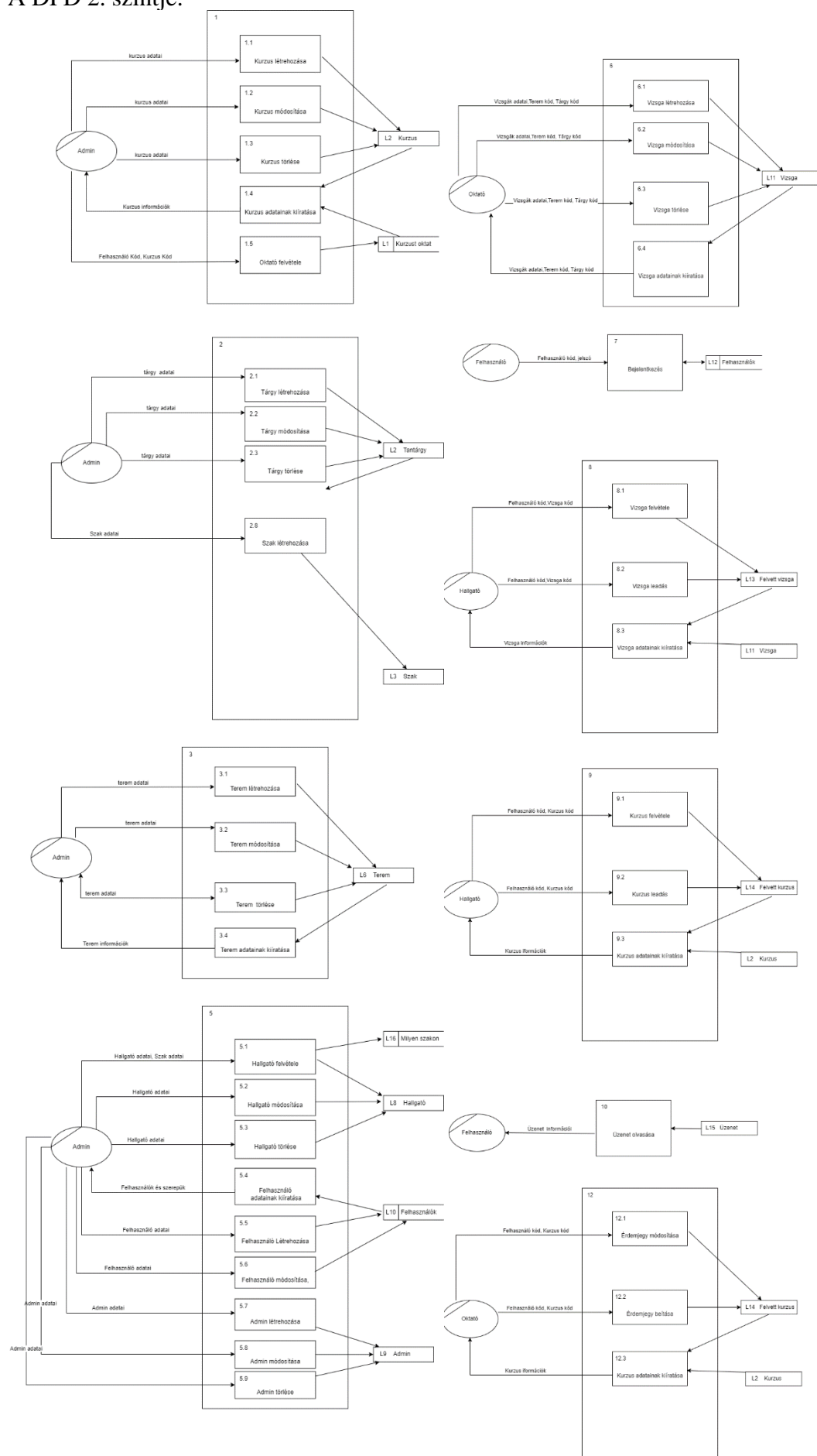


Logikai

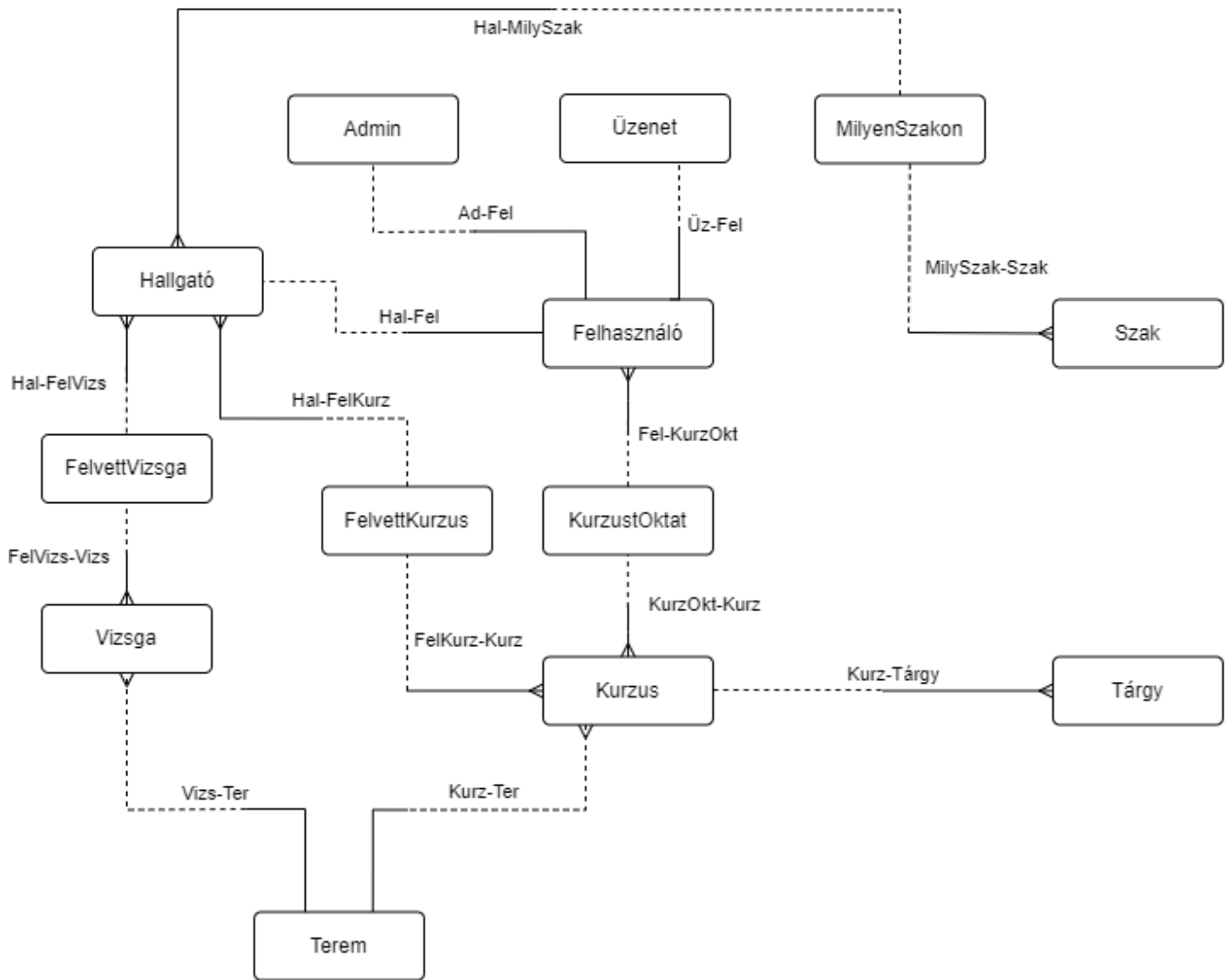
A DFD 1. szintje:



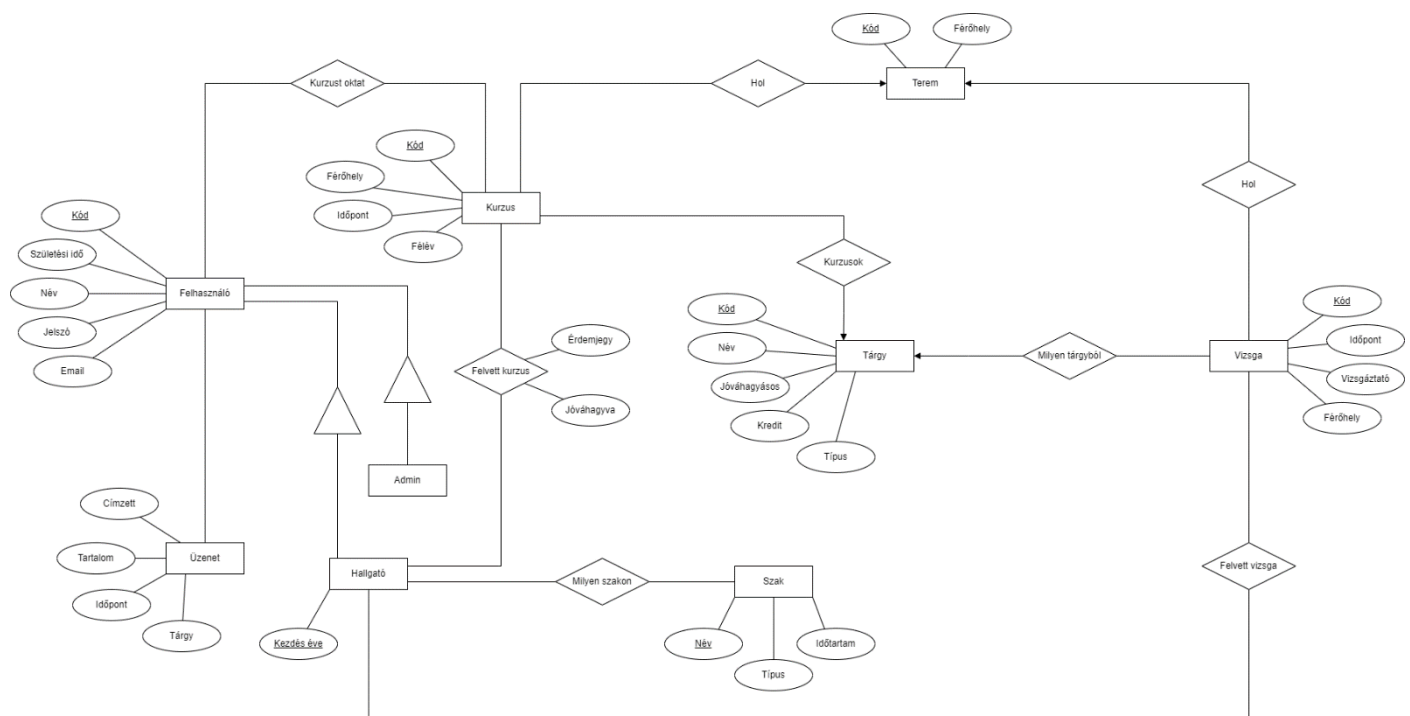
A DFD 2. szintje:



Egyedmodell:



EKT-diagram:



Relációs adatelemzés

Felhasználó(felhasználóKód, születésiIdő, név, jelszó, email)

Admin(felhasználóKód)

Hallgató(felhasználóKód, Szak.név, Szak.Típus, kezdésÉve)

Üzenet(időpont, tartalom, tárgy, felhasználóKód)

Szak(sakNév, sakTípus, időtartam)

Tárgy(tárgyKód, név, kredit, típus, jóváhagyás)

Terem(teremKód, férőhely)

Kurzus(kurzusKód, tárgyKód, férőhely, időpont, félév, TeremKód)

FelvettKurzus(Hallgató.felhasználóKód, Hallgató.SzakNév, Hallgató.SzakTípus, KurzusKód, Kurzus.tárgyKód, érdemjegy, jóváhagyva)

KurzustOktat(FelhasználóKód, KurzusKód, tárgyKód)

Vizsga(vizsgaKód, TárgyKód, kezdésiIdőpont, fejezésIdőpont, vizsgáztató, TeremKód)

FelvettVizsga(Hallgató.felhasználóKód, Hallgató.SzakNév, Hallgató.SzakTípus, VizsgaKód, Vizsga.tárgyKód)

Minden attribútum atomi, nincs a sémában sem többértékű, sem pedig összetett attribútum, így a séma 1NF-ben van.

A FelvettVizsga, KurzustOktat, nem rendelkeznek másodlagos attribútumokkal, illetve amelyek kizárólag egy kulccsal rendelkeznek, azok triviálisak. A séma 2NF-ben, illetve 3 NF-ben van.

1 NF, 2 NF, 3NF

Felhasználó

felhasználóKód

születésiIdő

név

jelszó

Hallgató

*felhasználóKód

*sakNév

*sakTípus

kezdésÉve

Admin

*felhasználóKód

Üzenet

időpont

tartalom

tárgy

*felhasználóKód

Szak

sakNév

sakTípus

időtartam

1 NF, 2 NF, 3NF

Terem

teremKód

férőhely

Kurzus

kurzusKód

*tárgyKód

férőhely

időpont

félév

*teremKód

KurzustOktat

*felhasználóKód

*kurzusKód

*kurzus.tárgyKód

Vizsga

vizsgaKód

*tárgyKód

kezdésiIdőpont

fejezésIdőpont

vizsgáztató

*teremKód

Tárgy
tárgyKód
név
kredit
típus
jóváhagyás

FelvettVizsga
*Hallgató.felhasználóKód
*Hallgató.szakNév
*Hallgató.szakTípus
* vizsgaKód

FelvettKurzus
*Hallgató.felhasználóKód
*Hallgató.szakNév
*Hallgató.szakTípus
*kurzusKód
érdemjegy
jóváhagyva

Függőségek:

{felhasználóKód} → {születésiIdő, név, jelszó, email}
{üzenetKód} → {tartalom, időpont, tárgy, feladóKód, címzettKód}
{szakNév} → {típus, időtartam}
{tárgyKód} → {név, jóváhagyásos, kredit, típus, előfeltételkód}
{szakNév, tárgyKód} → {típus, ajánlottFélév}
{teremKód} → {férőhely}
{kurzusKód} → {férőhely, időpont, félév, teremKód}
{vizsgaKód} → {időpont, vizsgáztató, tárgyKód, teremKód}
{Hallgató.felhasználóKód, kezdésÉve, kurzusKód} → {érdemjegy, jóváhagyva}

Táblák leírása:

Felhasználó		
Név	Típus	Leírás
<u>felhasználóKód</u>	CHAR(6)	a Felhasználó azonosítója
Név	VARCHAR2(50)	a Felhasználó neve
Email	VARCHAR2(320)	a Felhasználó email címe
Password	VARCHAR2(255)	a Felhasználó jelszava, titkosítva
SzületésiIdő	DATE	a Felhasználó születési dátuma

Hallgató		
Név	Típus	Leírás
* <u>felhasználóKód</u>	CHAR(6)	a Hallgató azonosítója
* <u>szakNév</u>	VARCHAR2(50)	a Szak neve, egyben azonosítója
* <u>szakTípus</u>	VARCHAR2(10)	a Szak típusa (Bsc/Mester/..)
kezdésÉve	CHAR(6)	a Hallgató kezdésének éve

Admin		
Név	Típus	Leírás
* <u>felhasználóKód</u>	CHAR(6)	az Admin azonosítója

Üzenet		
Név	Típus	Leírás
<u>időpont</u>	DATE	az Üzenet időpontja, egyben azonosítója
tartalom	CLOB	az Üzenet tartalma
tárgy	VARCHAR2(256)	az Üzenet tárgya
* <u>felhasználóKód</u>	CHAR(6)	a címzett azonosítója

Szak		
Név	Típus	Leírás
<u>név</u>	VARCHAR2(50)	a Szak neve, egyben azonosítója
<u>típus</u>	VARCHAR2(10)	a Szak típusa (Bsc/Mester/..)
időtartam	Int(5)	a Szak időtartalma félévekben

Tárgy		
Név	Típus	Leírás
<u>tárgyKód</u>	CHAR(6)	a Tárgy azonosítója
név	VARCHAR2(50)	a Tárgy neve
jóváhagyásos	NUMBER(1)	a Tárgy jóváhagyásos-e
kredit	INTEGER	a Tárgyért járó kreditek száma
típus	VARCHAR2(20)	a Tárgy típusa (Matekos/Informatikus/...)

Terem		
Név	Típus	Leírás
<u>teremKód</u>	VARCHAR2(20)	a Terem azonosítója
férőhely	NUMBER(3)	a Terem kapacitása

Kurzus		
Név	Típus	Leírás
* <u>tárgyKód</u>	CHAR(6)	a Kurzus tárgyának azonosítója
<u>kurzusKód</u>	CHAR(6)	a Kurzus azonosítója
férőhely	NUMBER(3)	a Kurzus férőhelye
időpont	CHAR(13)	a Kurzus időpontja
félév	CHAR(9)	a Kurzus melyik félévben zajlik
*teremKód	VARCHAR2(20)	a Terem azonsítója, ahol a kurzus történik

KurzusOktat		
Név	Típus	Leírás
* <u>felhasználóKód</u>	CHAR(6)	a Felhasználó azonosítója, aki az oktató
* <u>kurzusKód</u>	CHAR(6)	az oktatott Kurzus azonosítója
* <u>kurzus.tárgyKód</u>	CHAR(6)	az oktatott Kurzus tárgyának azonosítója

Vizsga		
Név	Típus	Leírás
* <u>tárgyKód</u>	CHAR(6)	a Vizsga tárgyának azonosítója
<u>vizsgaKód</u>	CHAR(6)	a Vizsga azonosítója
kezdésIdőpont	DATE	a Vizsga kezdetének időpontja
fejezésIdőpont	DATE	a Vizsga fejezésének időpontja
vizsgáztató	CHAR(6)	a vizsgáztató azonosítója
*teremKód	VARCHAR2(20)	a Terem azonosítója, ahol a vizsga történik

FelvettVizsga		
Név	Típus	Leírás
* <u>Hallgató.felhasználóKód</u>	CHAR(6)	a Hallgató azonosítója
* <u>Hallgató.szakNév</u>	VARCHAR2(50)	a Hallgató szakjának neve
* <u>Hallgató.szakTípus</u>	VARCHAR2(10)	a Hallgató szakjának típusa
* <u>vizsgaKód</u>	CHAR(6)	a felvett Vizsga azonosítója
* <u>vizsga.tárgyKód</u>	CHAR(6)	a felvett Vizsga tárgyának azonosítója

FelvettKurszus		
Név	Típus	Leírás
* <u>Hallgató.felhasználóKód</u>	CHAR(6)	a Hallgató azonosítója
* <u>Hallgató.szakNév</u>	VARCHAR2(50)	a Hallgató szakjának neve
* <u>Hallgató.szakTípus</u>	VARCHAR2(10)	a Hallgató szakjának típusa
* <u>kurszusKód</u>	CHAR(6)	a Felvett Kurszus azonosítója
* <u>kurszus.tárgyKód</u>	CHAR(6)	a Felvett Kurszus tárgyának azonosítója
érdemjegy	INT	a Felvett Kurszus érdemjegye
jóváhagyva	NUMBER(1)	a felvétel jóvá lett-e hagyva

Szerep-funkció mátrix:

Admin	Oktató	Hallgató	
X	X	X	Üzenet elolvasása
		X	Felvett kurzusok megtekintése
		X	Kurzus felvétel
		X	Kurzus leadás
X	X	X	Személyes adatok megtekintése
		X	Felvett vizsgák megtekintése
		X	Vizsgajelentkezés
		X	Vizsga leadás
X			Új tárgy
X			Tárgy szerkesztése
X			Új kurzus
X			Kurzus szerkesztése
X			Előfeltétel szerkesztése
X			Új szak
X			Szak szerkesztése
X			Új felhasználó
X			Felhasználó szerkesztése
X			Új terem
X			Terem módosítása
X			Terem törlése
	X		Hallgató jelentkezésének jóváhagyása
	X		Hallgató értékelése
	X		Vizsga hozzáadása
	X		Vizsga törlése

Egyed-esemény mátrix:

Események L(létrehozás) M(módosítást) T(törlés) Egyedek	Új felhasználó felvitele	Felhasználó adatainak módosítása	Saját adatok megtekintése	Új tárgy bevitele	Tárgy adatainak módosítása	Új kurzus bevitele	Kurzus törlése	Kurzus adatainak módosítása	Új terem bevitele	Új szak bevitele	Új üzenet küldése	Üzenet törlődik	Üzenet érkezése	Tárgy törlődik	Adminná válás	Halgatóvá válás
Felhasználó	L	M	O													
Hallgató																L M T
Tárgy				L	M									T		
Kurzus						L	T	M								
Vizsga																
Admin															L M T	
Üzenet											L	T	O			
Szak			O							L						
Terem									L							

Funkció megadása

Funkciónév	Vizsga létrehozása
Funkció azonosító	6.1
Típus	Online,felhasználói,általános
Felhasználói szerepek	Jogosult: Oktató
Funkció leírás	Új vizsgák kihirdetése az adatbázisba. Egy űrlapon meg kell adni a vizsga időpontját (dátum kiválasztóval), vizsgáztató nevét és a vizsgaterem helyszínét. Tantárgyhoz kapcsolódik a vizsga így a meghirdetett vizsga csak azoknál a Hallgatóknál lesz meghirdetve akik a tárgyhoz tartoznak.
Hibakezelés	Ha nincs megadva az űrlapon valamelyik kötelező adat, akkor erről üzenetablak jelenik meg. Ha a megadott dátum korábbi, mint az aktuális rendszerdátum , akkor “Túl korai dátum” hibaüzenet jelenik meg. Ha a kijelölt vizsgáztató már elfogadott korábban másik vizsga megtartására ugyanazon időpontra, „A vizsgáztató foglalt az adott időpontra” hibaüzenetet kap a felhasználó.
AFD-eljárások	6.1
Események	Szorgalmi- vagy a vizsga-időszakban
Esemény gyakoriság	Félévente de akkor sok
I/O leírások	
I/O szerkezetek	
Követelménykatalógusra hivatkozás	
Tömegszerűség	Hetente ahány kurzus van
Kapcsolódó funkciók	
Lekérdezések	
Lekérdezés gyakorisága	Felhasználói igénytől függ
Közös feldolgozás	
Dialógusnevek	
Szolgáltatási szint követelményei	
Leírás	
Célérték	6 mp
Tartomány	2-12 mp

Funkciónév	Tárgy létrehozás
Funkció azonosító	2.1
Típus	Online,adminisztrátori,általános
Felhasználói szerepek	Jogosult: Admin
Funkció leírás	Új tantárgy létrehozása az adatbázisba. Egy űrlapon meg kell adni a tantárgy kódját, nevét, jóváhagyásos-e, mennyi kreditet ér és a típusát. A tantárgyhoz egy előfeltétel is megadható, mely részletei egy összetett attribútumokban kerülnek tárolásra.
Hibakezelés	Ha nincs megadva az űrlapon valamelyik kötelező adat, akkor erről üzenetablak jelenik meg.
AFD-eljárások	2.1
Események	Bármikor
Esemény gyakoriság	Évente
I/O leírások	
I/O szerkezetek	
Követelménykatalógusra hivatkozás	
Tömegszerűség	1-2
Kapcsolódó funkciók	
Lekérdezések	
Lekérdezés gyakorisága	Felhasználói igénytől függ
Közös feldolgozás	
Dialógusnevek	
Szolgáltatási szint követelményei	
Leírás	
Célérték	5 mp
Tartomány	2-10 mp

Képernyőtervek

- Bejelentkezés



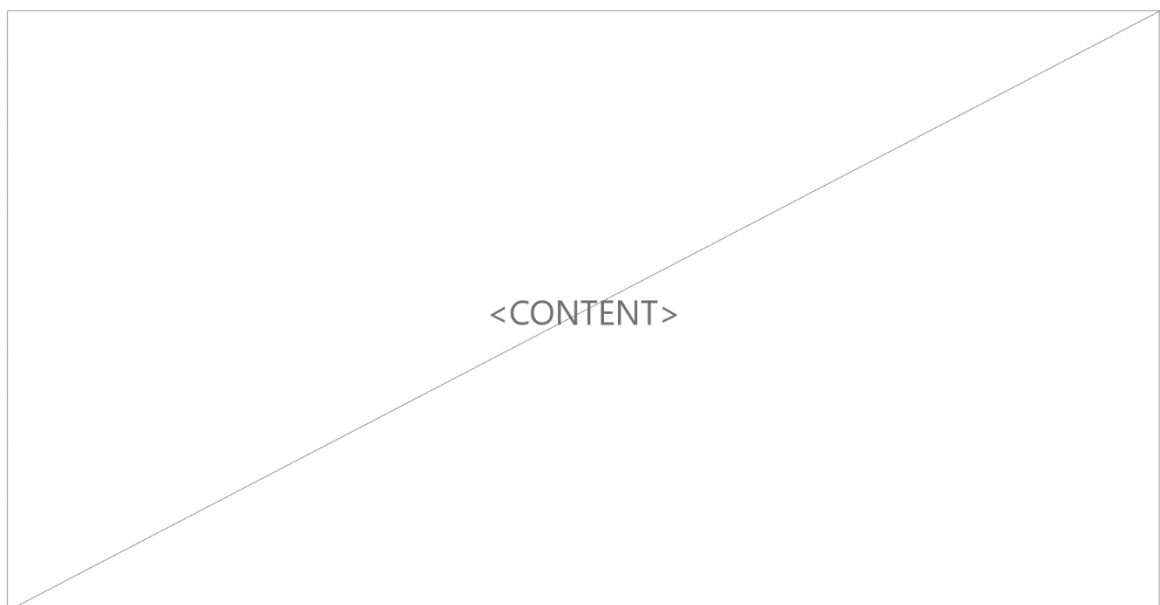
Murica kód

Jelszó

Bejelentkezés

- Menüsor [Oktató]

<LOGO>	Kurzusok	Vizsgák		Oktatói nézet ◀	Kijelentkezés
--------	----------	---------	--	-----------------	---------------



• Menüsor [Hallgató]

<LOGO>

Kurzusok

Vizsgák

{Szak neve}{kezdési félév} ◀

Kijelentkezés

Felvett kurzusok

Felvett vizsgák

Kurzusfelvétel

Vizsgajelentkezés

<CONTENT>

• Menüsor [Admin]

<LOGO>

Tárgyak

Szakok

Felhasználók

Termek

Kijelentkezés

<CONTENT>

• **Kurzusok [Oktató]**

Kurzusok

▶ {Tárgynév} {Tárgykód} {Gyakorlat/Előadás} {Kredit} {Jóváhagyásos-e}

▼ {Tárgynév} {Tárgykód} {Gyakorlat/Előadás} {Kredit} {Jóváhagyásos-e}

▶ {Kurzuskód} {Terem} {Időpont} {Jelentkezők}/{Férőhely} {oktatók}

▼ {Kurzuskód} {Terem} {Időpont} {Jelentkezők}/{Férőhely} {oktatók}

Hallgatók

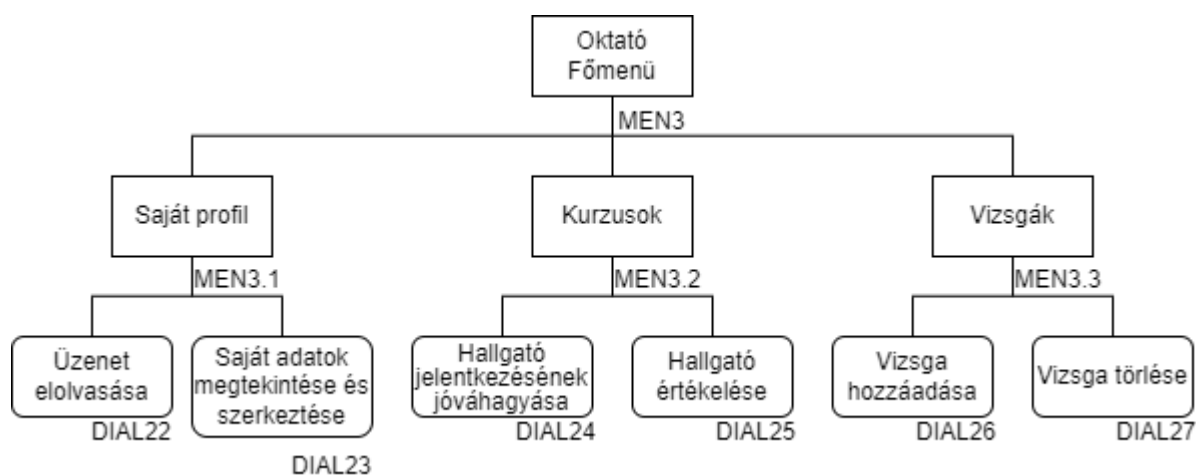
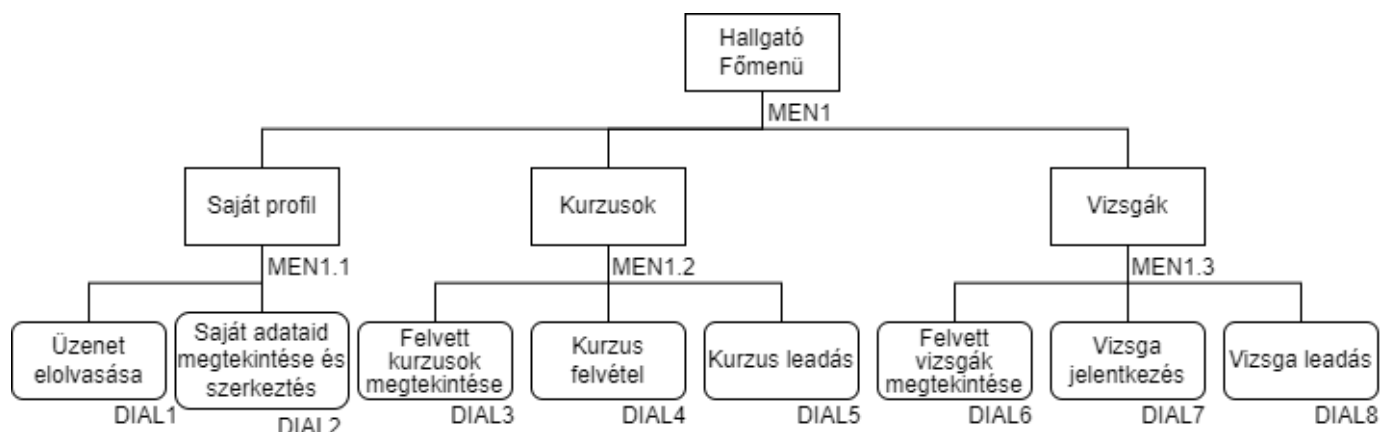
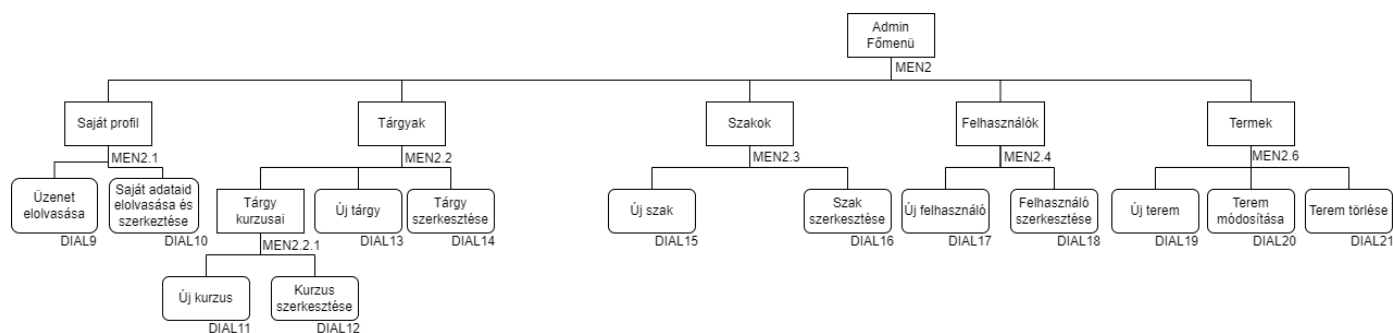
Jóváhagyás

Érdemjegy

{Murica kód}	{Név}	{Szak}	{Félév}	<input checked="" type="checkbox"/>	Jó (4) ▼
{Murica kód}	{Név}	{Szak}	{Félév}	<input checked="" type="checkbox"/>	Jó (4) ▼
{Murica kód}	{Név}	{Szak}	{Félév}	<input type="checkbox"/>	Nincs értékelve ▼
{Murica kód}	{Név}	{Szak}	{Félév}	<input checked="" type="checkbox"/>	Jó (4) ▼

▶ {Tárgynév} {Tárgykód} {Gyakorlat/Előadás} {Kredit} {Jóváhagyásos-e}

Menütervek



Triggerek:

CREATE

OR

```
replace TRIGGER approval_notification after UPDATE OF APPROVED
on murica_taken_course FOR each row DECLARE anew NUMBER; aOld
      number; uOld      char(6); sOld      char(6); text cl
ob; BEGIN
  anew := :NEW.approved;
  aold := :OLD.approved;
  sold := :OLD.subject_id;
  uold := :OLD.user_id;
  if anew = 1
  AND
  aold != anew THEN
    text := 'The registration for the subject with code ' || sold ||
    ' has been approved.';
    insert INTO murica_message VALUES
      (
        CURRENT_TIMESTAMP,
        text,
        'Status change',
        uold
      );
  end IF; IF anew = 0
  AND
  aold != anew then
    text := 'The registration for the subject with code ' || sold ||
    ' has not been approved.'; INSERT INTO murica_message VALUES
      (
        CURRENT_TIMESTAMP,
        text,
        'Status change',
        uold
      ); END IF; END;
```

Ez a trigger egy üzenetet küld a Hallgatónak, a tanár elfogadta e a jelentkezését a kurzusra.

```

CREATE
OR
replace TRIGGER grade_notification after UPDATE OF GRADE
on murica_taken_course FOR each row DECLARE gnew NUMBER; gOld
        number; uOld        char(6); sOld        char(6); text cl
ob; BEGIN
    gnew := :NEW.grade;
    gold := :OLD.grade;
    uold := :OLD.user_id;
    sold := :OLD.subject_id;
    if gnew IS NOT NULL
    AND
    gold != gnew THEN
        text := 'A grade of ' || gnew || ' has been entered for the sub
ject with code ' || sold || '.';
        insert INTO murica_message VALUES
            (
                CURRENT_TIMESTAMP,
                text,
                'Grade entry',
                uold
            );
    endIF; END;

```

Ez a trigger egy üzenetet küld a Hallgatónak, amikor a tanár be ír egy jegyet egy adott kurzusra.

PLSQL FUNKCIÓ

```
CREATE OR replace FUNCTION Calculate_ki(student_id CHAR,
                                         programname CHAR,
                                         programtype CHAR)

RETURN NUMBER
IS
    total_credits      NUMBER;
    total_grade_points NUMBER;
    average_grade      NUMBER;
BEGIN
    SELECT SUM(credit)
    INTO   total_credits
    FROM   murica_taken_course TC
          join murica_subject S
            ON TC.subject_id = S.id
    WHERE  TC.user_id = student_id
          AND TC.programme_name = programname
          AND TC.programme_type = programtype;

    SELECT SUM(credit * TC.grade)
    INTO   total_grade_points
    FROM   murica_taken_course TC
          join murica_subject S
            ON TC.subject_id = S.id
    WHERE  TC.user_id = student_id
          AND TC.programme_name = programname
          AND TC.programme_type = programtype;

    IF total_credits IS NULL
        OR total_credits = 0 THEN
        average_grade := NULL;
    ELSE
        average_grade := total_grade_points / total_credits;
    END IF;

    RETURN average_grade;
EXCEPTION
    WHEN no_data_found THEN
        RETURN NULL;
END;
```

Ez a funkció kiszámítja a Hallgató kredit index átlagát.

Implementáció itt található:

murica_bl_impl/Dao/OracleStudentDao.php


```

CREATE OR replace FUNCTION Calculate_kki(student_id CHAR,
                                         programname CHAR,
                                         programtype CHAR)
RETURN NUMBER
IS
    total_grade_points NUMBER;
    average_grade       NUMBER;
    total_credits       NUMBER;
BEGIN
    SELECT SUM(credit)
    INTO    total_credits
    FROM    murica_taken_course TC
           join murica_subject S
             ON TC.subject_id = S.id
    WHERE   TC.user_id = student_id
           AND TC.programme_name = programname
           AND TC.programme_type = programtype;

    SELECT SUM(credit * TC.grade)
    INTO    total_grade_points
    FROM    murica_taken_course TC
           join murica_subject S
             ON TC.subject_id = S.id
    WHERE   TC.user_id = student_id
           AND TC.programme_name = programname
           AND TC.programme_type = programtype;

    IF total_credits IS NULL
        OR total_credits = 0 THEN
        average_grade := NULL;
    ELSE
        average_grade := total_grade_points / 30;
    END IF;

    RETURN average_grade;
EXCEPTION
    WHEN no_data_found THEN
        RETURN NULL;
END;

```

Ez a funkció kiszámítja a Hallgató korrigált kredit index átlagát.

Implementáció itt található:

murica_bl_impl/Dao/OracleStudentDao.php

Megvalósítási szoftverkörnyezet

Felhasználni tervezett technológiák: Php, JQuery