

Adatbázis alapú rendszerek

2023-2024/2

IB152L-8

Piktüre

Készítette:

Tóth Martin

Szabó Gergő

Kovács Tamás

Munka felosztása

1. mérföldkő:
Dokumentáció: mindenki.
2. mérföldkő:
Adatbázis implementálása: mindenki
3. mérföldkő:
Kovács Tamás: minden más
Tóth Martin: Adatbázis feltöltése
Szabó Gergő: Admin felület
4. mérföldkő:
Tóth Martin: Trigger-ek elkészítése.
Kovács Tamás: Tárolt funkciók.
Szabó Gergő: Auto increment trigger-ek
5. mérföldkő
Kovács Tamás: Minden.
Szabó Gergő: dokumentáció aktualizálása / javítása
Tóth Merlin: Debug.

Értékelési mód:

Kovács Tamás: elérhető max
Szabó G. Tóth M.: Maradék/2 személyenként.

Feladat szöveges leírása

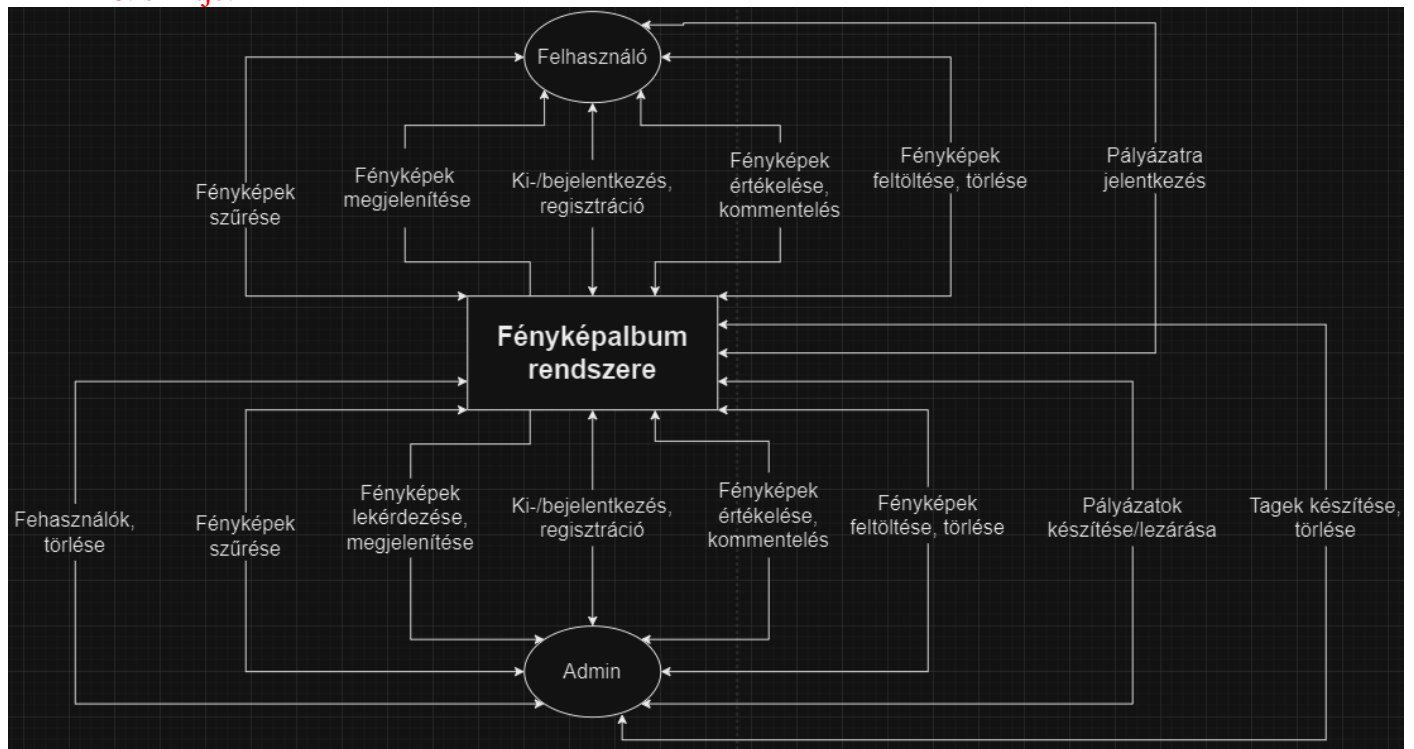
Egy fényképeket kezelő weboldal. A felhasználók feltölthetik képeiket kategóriákkal és helymegjelöléssel. Egymás képei értékelhetőek, hozzászólhatóak. Kategóriák, hely szerint sorba rendezhetőek. Fénykép pályázatok indíthatóak, ezekre képek küldhetőek, szavazással eldönthetőek.

Követelménykatalógus

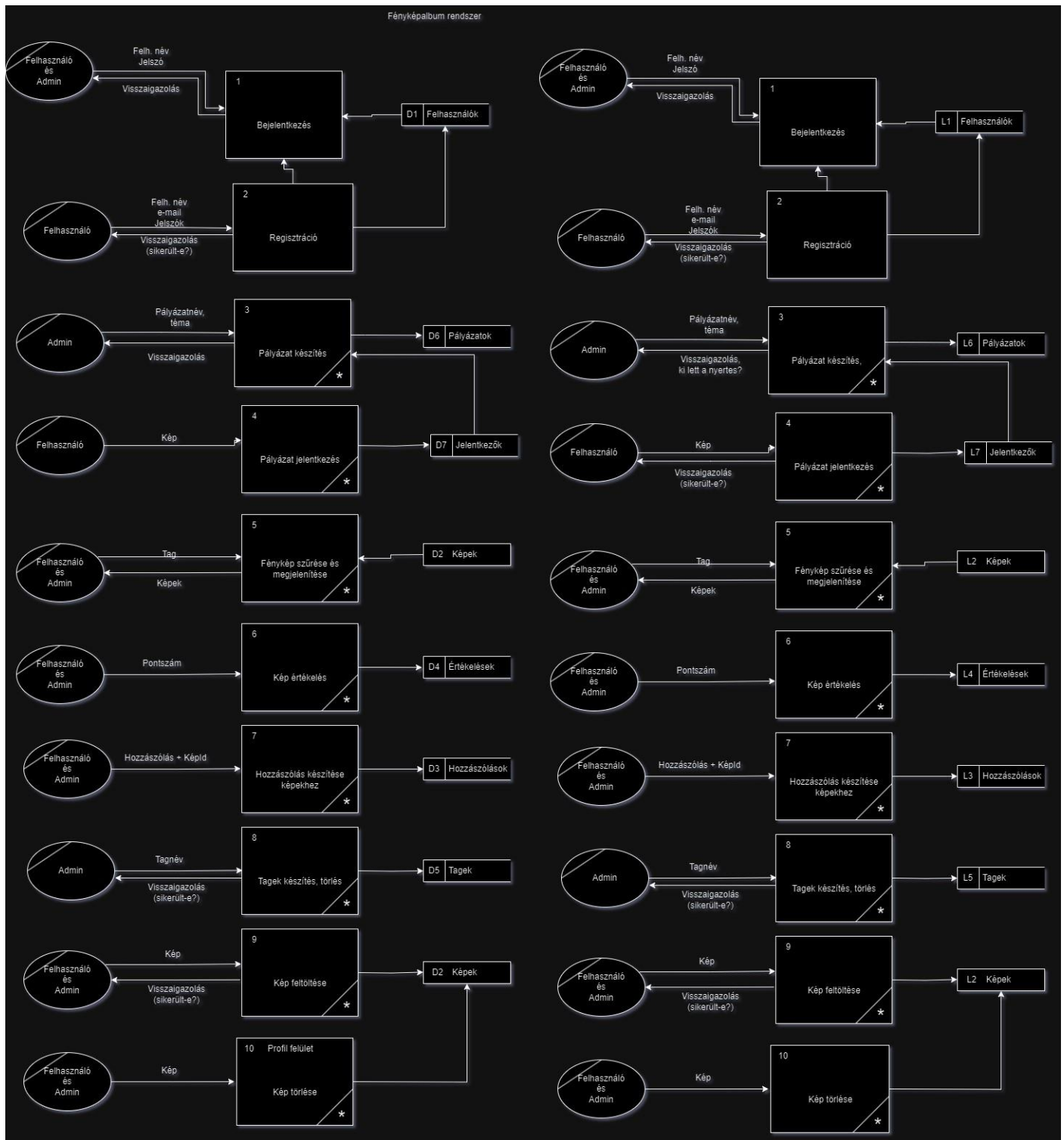
- Felhasználók kezelése
- Képek feltöltése
- Kategóriák
- Adott kategóriában hány kép van?
- Legtöbb képpel rendelkező felhasználók
- Képek értékelése
- A kategóriák legjobb képei
- Országok, megyék, városok felvitele
- Képhez megjelölni, hogy hol készült
- Felhasználóknál a település megjelölése
- Településenként hány fénykép készült
- Városok „arcai” a feltöltött képek alapján
- Feltöltési statisztikák
- Fórumok, megjegyzések fényképekhez, válaszolási lehetőséggel
- Felhasználói értékelés, pontozás a feltöltött képekre kapott pontszámok, és az adott pontszámok, hozzászólások alapján
- Fotópályázatok kiírása, szavazás, nyertes kihirdetése, ...

Adatfolyam diagram (DFD):

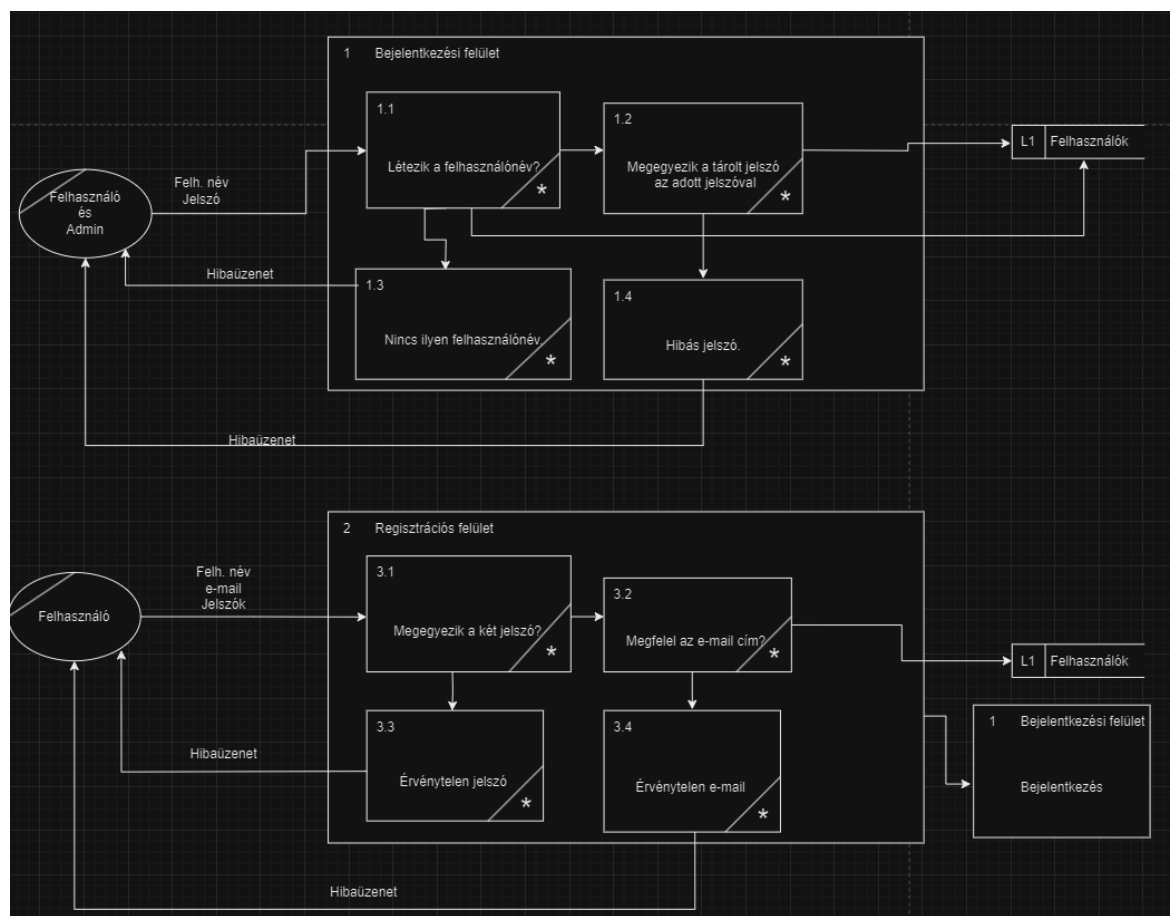
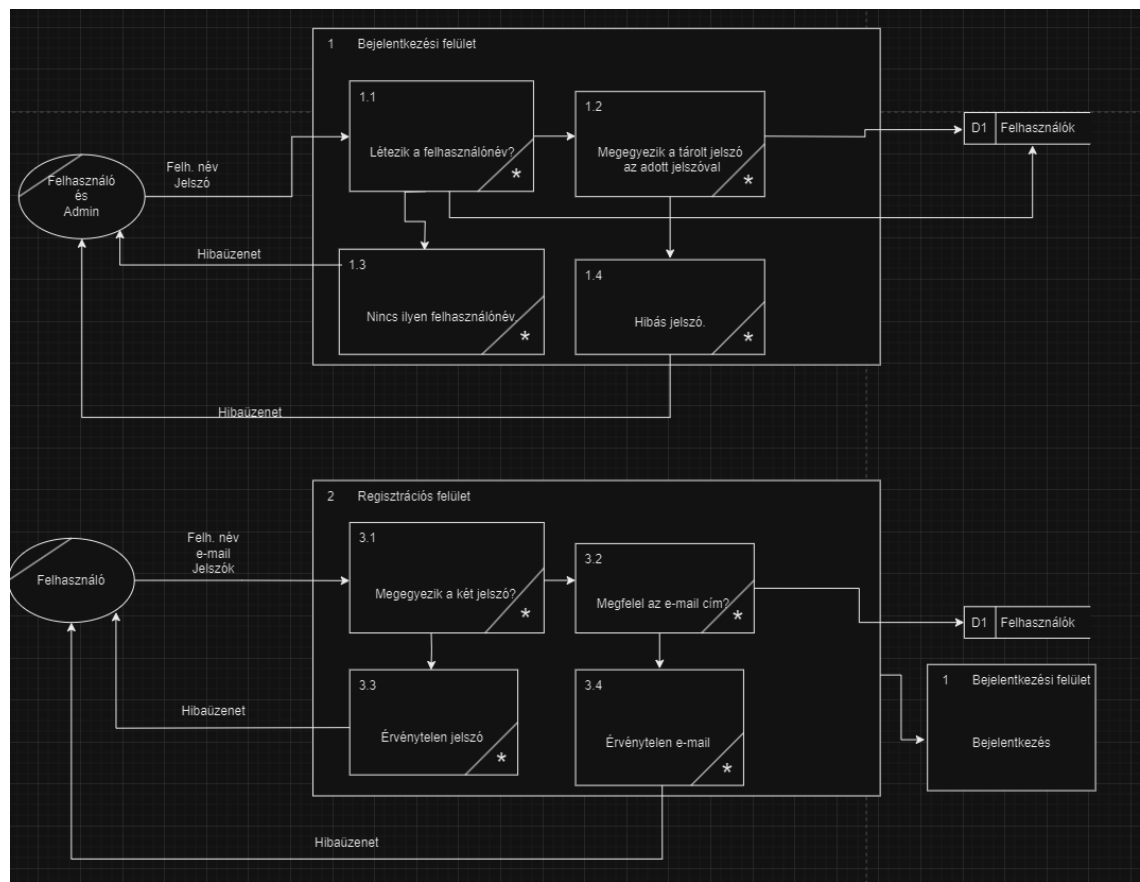
A DFD 0. szintje:



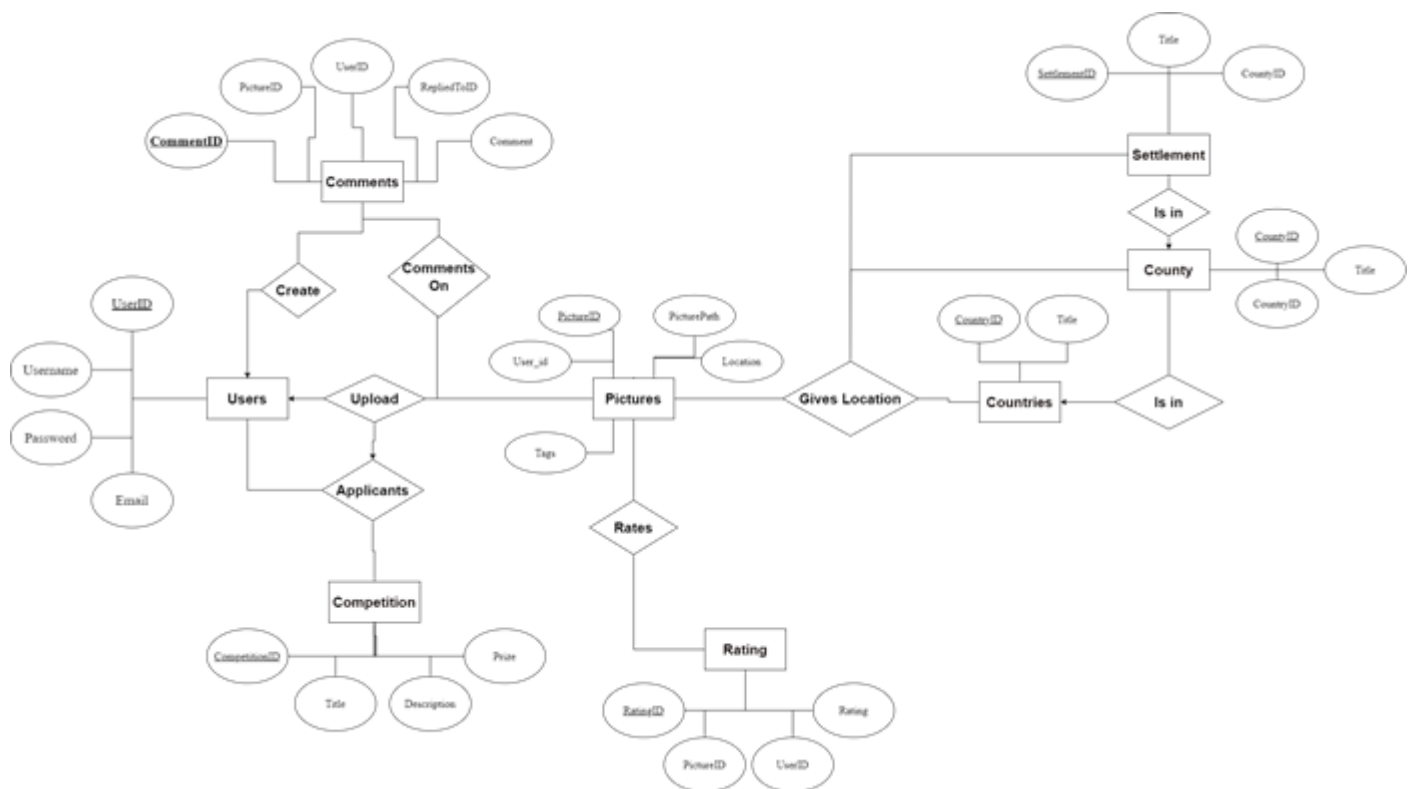
A DFD 1. szintje:



A DFD 2. szintje:



EK-diagram:



Az EK-diagram leképezése relációsémákra:

Users (UserID, Username, Password, Email)
Pictures (PictureID, User_ID, Tag, PicturePath, Location)
Comments (CommentID, PictureID, UserID, RepliedToID, Comment)
Rating (RatingID, PictureID, UserID, Star)
Locations (LocationID, CountryID, CountyID, SettlementID)
Competition(CompetitionID, Title, Description, Prize)
Applicants (UserID, ApplicationID, PictureID)

A relációsémák normalizálása:

Normalizálás „táblázatos” formában.

Normálforma	Leírás
1NF	Teljesült mert leképezés után nem maradt se összetett se többértékű attribútum.
2NF	Teljesül mert, minden kulcs egyelemű.
3NF	Teljesül mert mindegyik másodlagos attribútum közvetlenül függ a kulcstól.

Táblák leírása:

Users

Név	Típus	Leírás
UserID*	INT	az felhasználók azonosítása id alapján
Username	VARCHAR	felhasználók nevei (lehet ugyanaz)
Password	VARCHAR	felhasználó jelszava titkosítva
Email	VARCHAR	a felhasználó email-címe (nincs 2 azonos)

Pictures

Név	Típus	Leírás
PictureID*	INT	a képek azonosítása id alapján
User_id	INT	kép és felhasználó összekapcsolása
Tags	VARCHAR	kép alá tartozó tag-k összefűzve
PicturePath	VARCHAR	a kép elérési útvonala
Location	VARCHAR	ország, megye, település összefűzve

Comments

Név	Típus	Leírás
CommentID*	INT	a hozzászólás azonosítása id alapján
PictureID	INT	kép és hozzászólás összekapcsolása
UserID	INT	kép és felhasználó összekapcsolása
RepliedToID	INT	melyik hozzászólásra válaszolt
Comment	TEXT	hozzászólás

Rating

Név	Típus	Leírás
RatingID*	INT	az értékelés azonosítása id alapján
PictureID	INT	kép és értékelés összekapcsolása
UserID	INT	értékelés és felhasználó összekapcsolása
Rating	INT	értékelés 1-5

Countries

Név	Típus	Leírás
CountryID*	INT	a országok azonosítása id alapján
Title	VARCHAR	ország pl. Németország

Counties

Név	Típus	Leírás
CountyID*	INT	a megyék azonosítása id alapján
Title	VARCHAR	megye pl. Pest
CountryID	INT	melyik országhoz tartozik a megye

Settlement

Név	Típus	Leírás
SettlementID*	INT	a települések azonosítása id alapján
Title	VARCHAR	település pl. Pécs
CountyID	INT	melyik megyéhez tartozik a település

Competition

Név	Típus	Leírás
CompetitionID*	INT	a verseny azonosítása id alapján
Title	VARCHAR	a verseny neve
Description	TEXT	a verseny leírása
Prize	INT	nyeremény

Applicants

Név	Típus	Leírás
UserID*	INT	a pályázó azonosítása id alapján
PictureID	INT	melyik képpel pályázott
CompetitionID	INT	melyik versenyre pályázott

Szerep-funkció mátrix:

Pályázat leültása	x				
Pályázat lezárása	..	x	x				
Pályázat írása	..	x	x				
Hozzászólás törlése	x				
Hozzászólás	..	x	x				
Kép értékelése	..	x	x				
Más képének törlése	x	Felhasználónév vál.	..	x	
Kép törlése	..	x	x	Jelszó változtatás	..	x	
Kép feltöltése	..	x	x	Kép csoportosítása	..	x	
Kép megtekintése	..	x	x	Tag törlése	..	x	
Bejelentkezés	..	x	x	Felhasználó törlése	..	x	
Regisztráció	x	x	x	Tag hozzáadás	..	x	
	Látogató	Felhasználó	Admin		Látogató	Felhasználó	Admin

Nem számláló Trigger:

```
create or replace TRIGGER NaploUpload
AFTER INSERT
ON PICTURES
FOR EACH ROW
BEGIN
    INSERT INTO UPLOAD VALUES (:NEW.USERID, SYSDATE, :NEW.PICTUREID);
END;
```

Tárolt funkció:

```
create or replace FUNCTION NewRating (
    p_id IN number,
    u_id IN number,
    rat IN number)
RETURN BOOLEAN
AS
    v_rat NUMBER := 0;
BEGIN
    SELECT COUNT(*) INTO v_rat FROM RATING WHERE USERID = u_id AND PICTUREID = p_id;
    IF v_rat = 0 THEN
        INSERT INTO RATING VALUES (p_id, u_id, rat);
        RETURN TRUE;
    END IF;
    UPDATE RATING SET STAR = rat WHERE USERID = u_id AND PICTUREID = p_id;
    RETURN FALSE;
END NewRating;
```

```
create or replace FUNCTION NewRating (
    p_id IN number,
    u_id IN number,
    rat IN number)
RETURN BOOLEAN
AS
    v_rat NUMBER := 0;
BEGIN
    SELECT COUNT(*) INTO v_rat FROM RATING WHERE USERID = u_id AND PICTUREID = p_id;
    IF v_rat = 0 THEN
        INSERT INTO RATING VALUES (p_id, u_id, rat);
        RETURN TRUE;
    END IF;
    UPDATE RATING SET STAR = rat WHERE USERID = u_id AND PICTUREID = p_id;
    RETURN FALSE;
END NewRating;
```

Egyéb trigger:

```
CREATE OR REPLACE TRIGGER NaploUploadRat
AFTER INSERT
ON RATING
FOR EACH ROW
BEGIN
    INSERT INTO UPLOAD VALUES (:NEW.USERID, SYSDATE, 'Kép értékelés: ' || :NEW.PICTUREID);
END
;
```

```

create or replace FUNCTION "LOGIN" (
    try_email IN VARCHAR2,
    try_password IN VARCHAR2)
RETURN NUMBER
AS
    id NUMBER;
BEGIN
    SELECT MAX(USERID) INTO id FROM USERS WHERE EMAIL = try_email AND PASSWORD = try_password;
    RETURN id;
END login;

```

```

create or replace FUNCTION NewPic (
    c_name IN VARCHAR2,
    co_name IN VARCHAR2,
    s_name IN VARCHAR2)
RETURN NUMBER
AS
    c_db NUMBER := 0;
    co_db NUMBER := 0;
    s_db NUMBER := 0;
    c_new NUMBER := 0;
    co_new NUMBER := 0;
    s_new NUMBER := 0;
    l_id NUMBER := 0;
BEGIN
    SELECT COUNT() INTO c_db FROM COUNTRY WHERE NAME = c_name;
    IF c_db = 0 THEN
        INSERT INTO COUNTRY (NAME) VALUES (c_name);
    END IF;
    SELECT MAX(ID) INTO c_new FROM COUNTRY WHERE NAME = c_name;

    SELECT COUNT() INTO co_db FROM COUNTY WHERE NAME = co_name;
    IF co_db = 0 THEN
        INSERT INTO COUNTY (COUNTRYID, NAME) VALUES (c_new, co_name);
    END IF;
    SELECT MAX(ID) INTO co_new FROM COUNTY WHERE NAME = co_name;

    SELECT COUNT(*) INTO s_db FROM SETTLEMENT WHERE NAME = s_name;
    IF s_db = 0 THEN
        INSERT INTO SETTLEMENT (COUNTRYID, NAME) VALUES (co_new, s_name);
    END IF;
    SELECT MAX(ID) INTO s_new FROM SETTLEMENT WHERE NAME = s_name;

    IF s_db = 0 OR co_db = 0 OR c_db = 0 THEN
        INSERT INTO LOCATIONS (COUNTRYID, COUNTYID, SETTLEMENTID) VALUES (c_new, co_new, s_new);
    END IF;
    SELECT MAX(LOCATIONID) INTO l_id FROM LOCATIONS WHERE COUNTRYID = c_new AND COUNTYID = co_new AND SETTLEMENTID = s_new;
    RETURN l_id;
END NewPic;

```