

Tervezés Lépései:

1. Probléma meghatározása:

- Milyen adatokra van szükség? (profilok, játékosok, statisztika a játékosokról, mérkőzések eredményei, csapatok adatai, bajnokságok és azoknak a csoportjai)
- Milyen entitások léteznek? (felhasználó, csapat, csoport, játékos, statisztika, mérkőzés, bajnokság)
- Milyen relációk vannak az entitások között? (egy játékosnak több statisztikát is lehet készíteni, egy mérkőzés alapján több játékosról is lehet statisztikát készíteni, egy mérkőzéshez két csapat tartozik, viszont egy csapat több mérkőzésen is részt vehet, egy csapatba több játékos is van, egy edző több csapatot is edzhet, egy csoportba több csapat is van, egy bajnokságban több csoport van, egy felhasználó több bajnokságot is rendezhet)

2. Entitások azonosítása:

- felhasználó (id)
- csapat (csapat_id)
- csoport (csoport_id)
- játékos (jatekosi_id)
- mérkőzés (merkozes_id)
- bajnokság (bajnoksag_id)
- statisztika (statisztika_id)

3. Attribútumok meghatározása:

felhasznalo tábla

- id: Egyedi azonosító minden felhasználóhoz. Elsődleges kulcs, automatikusan növekvő.
- nev: A felhasználó neve (például teljes név vagy becenév).
- email: A felhasználó e-mail címe. Egyedi érték, hogy ne lehessen két felhasználónak ugyanaz az e-mail címe.
- jelszo: A felhasználó jelszava.
- iskola: Az iskola neve, amihez a felhasználó tartozik (nem kötelező kitölteni).

bajnoksag tábla

- bajnoksag_id: Egyedi azonosító minden bajnoksághoz. Elsődleges kulcs, automatikusan növekvő.
- nev: A bajnokság neve (pl. "Falcsik 2024").
- kezdet_datum: A bajnokság kezdő dátuma.
- vege_datum: A bajnokság befejező dátuma.

- szervezo_id: A bajnokság szervezőjének azonosítója.

csapat tábla

- csapat_id: Egyedi azonosító minden csapathoz. Elsődleges kulcs, automatikusan növekvő.
- nev: A csapat neve (pl. "Felrúglak Elesel").
- iskola: Az iskola neve, amihez a csapat tartozik (nem kötelező kitölteni).
- edzo_id: A csapat edzőjének azonosítója.

csoport tábla

- csoport_id: Egyedi azonosító minden csoporthoz. Elsődleges kulcs, automatikusan növekvő.
- bajnoksag_id: Annak a bajnokságnak az azonosítója, amihez a csoport tartozik.
- nev: A csoport neve (pl. "A csoport", "B csoport").
- csapat_id: Annak a csapatnak az azonosítója, ami a csoport tagja.

jatekos tábla

- jatekos_id: Egyedi azonosító minden játékoshoz. Elsődleges kulcs, automatikusan növekvő.
- nev: A játékos neve.
- kor: A játékos életkora.
- csapat_id: Annak a csapatnak az azonosítója, amihez a játékos tartozik.

merkozes tábla

- merkozes_id: Egyedi azonosító minden mérkőzéshez. Elsődleges kulcs, automatikusan növekvő.
- csapat1_id: Az első csapat azonosítója.
- csapat2_id: A második csapat azonosítója.
- datum: A mérkőzés dátuma.
- helyszin: A mérkőzés helyszíne.
- eredmeny_csapat1: Az első csapat által szerzett gólok száma.
- eredmeny_csapat2: A második csapat által szerzett gólok száma.

statisztika tábla

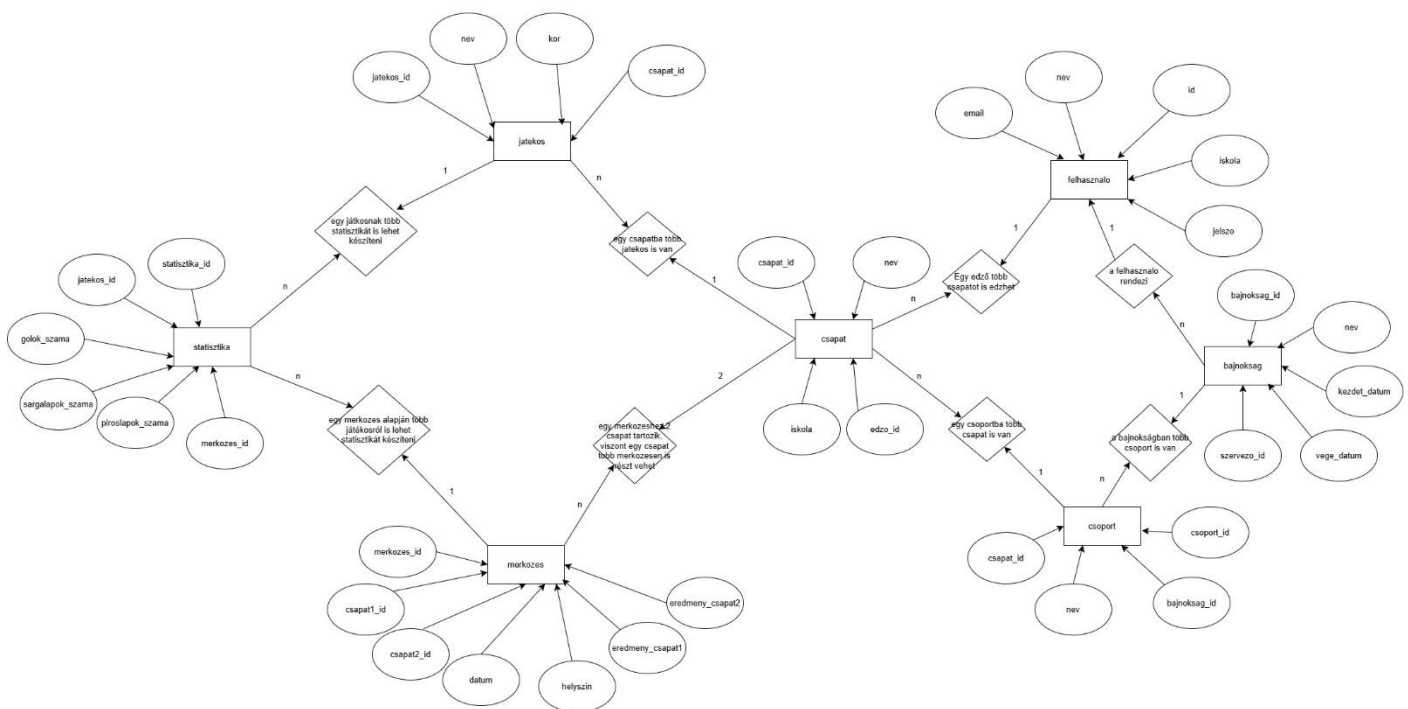
- statisztika_id: Egyedi azonosító minden statisztikai adathoz. Elsődleges kulcs, automatikusan növekvő.
- jatekos_id: A játékos azonosítója.
- golo_k_szama: A játékos által szerzett gólok száma az adott mérkőzésen.
- sargalapok_szama: A játékos által kapott sárga lapok száma az adott mérkőzésen.

- piroslapok_szama: A játékos által kapott piros lapok száma az adott mérkőzésen.
- merkozes_id: Annak a mérkőzésnek az azonosítója, amiről a statisztika készül.

4. Kapcsolatok azonosítása:

- játékos és statisztika (1:N): egy játékosnak több statisztikát is lehet készíteni
- merkozes és statisztika (1:N): egy merkozes alapján több játékosról is lehet statisztikát készíteni
- merkozes és csapat (N:2): egy merkozeshez két csapat tartozik, viszont egy csapat több mérkőzésen is részt vehet
- játékos és csapat (N:1): egy csapatba több játékos is van
- felhasználó és csapat (1:N): egy edző több csapatot is edzhet
- csoport és csapat (1:N): egy csoportba több csapat is van
- bajnoksag és csoport (1:N): egy bajnokságban több csoport van
- felhasználó és bajnoksag (1:N): egy felhasználó több bajnokságot is rendezhet

5. ER-Diagram készítése:



6. **Normalizálás:** Nincsenek ismétlődő adatok, minden mezőben pontos érték szerepel, minden táblának saját egyedi azonosítója van.

7. **Adatbázis sémája:** Az ER-diagram után, kell egy séma leírás, ami tartalmazza az entitások és kapcsolatok részletes definícióját, valamint az attribútumok típusait és a kulcsokat.

Példa:

```

CREATE TABLE felhasznalo (
    id INT AUTO_INCREMENT PRIMARY KEY,

```

```
nev VARCHAR(255) NOT NULL,  
email VARCHAR(255) NOT NULL UNIQUE,  
jelszo VARCHAR(255) NOT NULL,  
iskola VARCHAR(255)  
);
```

```
CREATE TABLE bajnoksag (  
    bajnoksag_id INT AUTO_INCREMENT PRIMARY KEY,  
    nev VARCHAR(255) NOT NULL,  
    kezdet_datum DATE NOT NULL,  
    vege_datum DATE NOT NULL,  
    szervezo_id INT,  
    FOREIGN KEY (szervezo_id) REFERENCES felhasznalo(id)  
);
```

```
CREATE TABLE csapat (  
    csapat_id INT AUTO_INCREMENT PRIMARY KEY,  
    nev VARCHAR(255) NOT NULL,  
    iskola VARCHAR(255),  
    edzo_id INT,  
    FOREIGN KEY (edzo_id) REFERENCES felhasznalo(id)  
);
```

```
CREATE TABLE csoport (  
    csoport_id INT AUTO_INCREMENT PRIMARY KEY,  
    bajnoksag_id INT NOT NULL,  
    nev VARCHAR(255) NOT NULL,  
    csapat_id INT NOT NULL,  
    FOREIGN KEY (bajnoksag_id) REFERENCES bajnoksag(bajnoksag_id),  
    FOREIGN KEY (csapat_id) REFERENCES csapat(csapat_id)  
);
```

```
CREATE TABLE jatekos (  
    jatekos_id INT AUTO_INCREMENT PRIMARY KEY,  
    nev VARCHAR(255) NOT NULL,
```

```
kor INT,  
csapat_id INT NOT NULL,  
FOREIGN KEY (csapat_id) REFERENCES csapat(csapat_id)  
);
```

```
CREATE TABLE merkozes (  
    merkozes_id INT AUTO_INCREMENT PRIMARY KEY,  
    csapat1_id INT NOT NULL,  
    csapat2_id INT NOT NULL,  
    datum DATE NOT NULL,  
    helyszin VARCHAR(255),  
    eredmeny_csapat1 INT,  
    eredmeny_csapat2 INT,  
    FOREIGN KEY (csapat1_id) REFERENCES csapat(csapat_id),  
    FOREIGN KEY (csapat2_id) REFERENCES csapat(csapat_id)  
);
```

```
CREATE TABLE statisztika (  
    statisztika_id INT AUTO_INCREMENT PRIMARY KEY,  
    jatekos_id INT NOT NULL,  
    golok_szama INT DEFAULT 0,  
    sargalapok_szama INT DEFAULT 0,  
    piroslapok_szama INT DEFAULT 0,  
    merkozes_id INT NOT NULL,  
    FOREIGN KEY (jatekos_id) REFERENCES jatekos(jatekos_id),  
    FOREIGN KEY (merkozes_id) REFERENCES merkozes(merkozes_id)  
);
```

8. Létrehozása és tesztelése: a tervezett séma alapján létrehozuk az adatbázist. Miután elkészült, fontos tesztelnünk a felépítését és a lekérdezéseket, hogy minden megfelelően működik.

1. Listázd a bajnokságok neveit és a hozzájuk tartozó szervezők nevét!

```
SELECT b.nev AS bajnoksag_nev, f.nev AS szervezo_nev  
FROM bajnoksag b INNER JOIN felhasznalo f ON b.szervezo_id = f.id;
```

bajnoksag_nev	szervezo_nev
Iskolai Foci Kupa 2024	Kovács Péter
Diák Liga	Nagy Anna

2. Listázd ki a csapatok neveit és edzőik nevét!

```
SELECT c.nev AS csapat_nev, f.nev AS edzo_nev
```

```
FROM csapat c INNER JOIN felhasznalo f ON c.edzo_id = f.id;
```

csapat_nev	edzo_nev
Futó Gepárdok	Kovács Péter
Villám Tigrisek	Nagy Anna
Száguldó Sastollak	Tóth Márk

3. Listázd a csoportokban szereplő csapatokat a bajnokság nevével együtt!

```
SELECT cs.nev AS csoport_nev, b.nev AS bajnoksag_nev, c.nev AS csapat_nev
```

```
FROM csoport cs INNER JOIN bajnoksag b ON cs.bajnoksag_id = b.bajnoksag_id
```

```
INNER JOIN csapat c ON cs.csapat_id = c.csapat_id;
```

csoport_nev	bajnoksag_nev	csapat_nev
A csoport	Iskolai Foci Kupa 2024	Futó Gepárdok
A csoport	Iskolai Foci Kupa 2024	Villám Tigrisek
B csoport	Diák Liga	Száguldó Sastollak

4. Listázd ki azoknak a játékosoknak a nevét, akiknek van legalább 2 gólja egy mérkőzésen!

```
SELECT j.nev AS jatekos_nev, s.golok_szama, m.datum AS merkozes_datum
```

```
FROM statisztika s INNER JOIN jatekos j ON s.jatekos_id = j.jatekos_id
```

```
INNER JOIN merkozes m ON s.merkozes_id = m.merkozes_id
```

```
WHERE s.golok_szama >=2;
```

jatekos_nev	golok_szama	merkozes_datum
Szabó Dávid	2	2024-03-05

5. Listázd ki a mérkőzések eredményeit és a csapatok neveit!

```
SELECT m.datum, cs1.nev AS csapat1, cs2.nev AS csapat2, m.eredmeny_csapat1,
m.eredmeny_csapat2
```

FROM merkozes m INNER JOIN csapat cs1 ON m.csapat1_id = cs1.csapat_id

INNER JOIN csapat cs2 ON m.csapat2_id = cs2.csapat_id;

datum	csapat1	csapat2	eredmeny_csapat1	eredmeny_csapat2
2024-03-05	Futó Gepárdok	Villám Tigrisek	3	2
2024-03-10	Villám Tigrisek	Száguldó Sastollak	1	1

6. Számold meg, hány gólt szereztek a játékosok csapatonként!

SELECT c.nev AS csapat_nev, SUM(s.golok_szama) AS osszes_gol

FROM csapat c INNER JOIN jatekos j ON c.csapat_id = j.csapat_id

INNER JOIN statisztika s ON j.jatekos_id = s.jatekos_id

GROUP BY c.nev;

csapat_nev	osszes_gol
Futó Gepárdok	2
Száguldó Sastollak	1
Villám Tigrisek	1

7. Listázd ki azokat a bajnokságokat, amelyekben legalább 2 különböző csapat szerepel!

SELECT b.nev AS bajnoksag_nev, COUNT(DISTINCT c.csapat_id) AS csapatok_szama

FROM bajnoksag b INNER JOIN csoport cso ON b.bajnoksag_id = cso.bajnoksag_id

INNER JOIN csapat c ON cso.csapat_id = c.csapat_id

GROUP BY b.nev

HAVING csapatok_szama >= 2;

bajnoksag_nev	csapatok_szama
Iskolai Foci Kupa 2024	2

