

Tervezés Lépései:

1. Probléma meghatározása:

- Milyen adatok kellenek? (pl.: játékosok, csapatok, meccsek, profilok, tornák, csoportok)
- Milyen kapcsolatok léteznek? (pl.: a játékos hány gólt rúgott a meccsen, a csapat melyik csoportba lett kisorsolva, ki lett a torna győztese)

2. Entitások azonosítása: olyan objektumok, amelyek az adatbázisban szerepelni fognak. Mindegyik egyedi azonosítóval rendelkezik:

Példa:

- Játékos (azonosító: jatekosid)
- Csapat (azonosító:)
- Meccs (azonosító: meccsid)
- Profil (azonosító: profilid)

3. Attribútumok meghatározása: Profil

- o emailcim
- o felhasználonev
- o jelszo

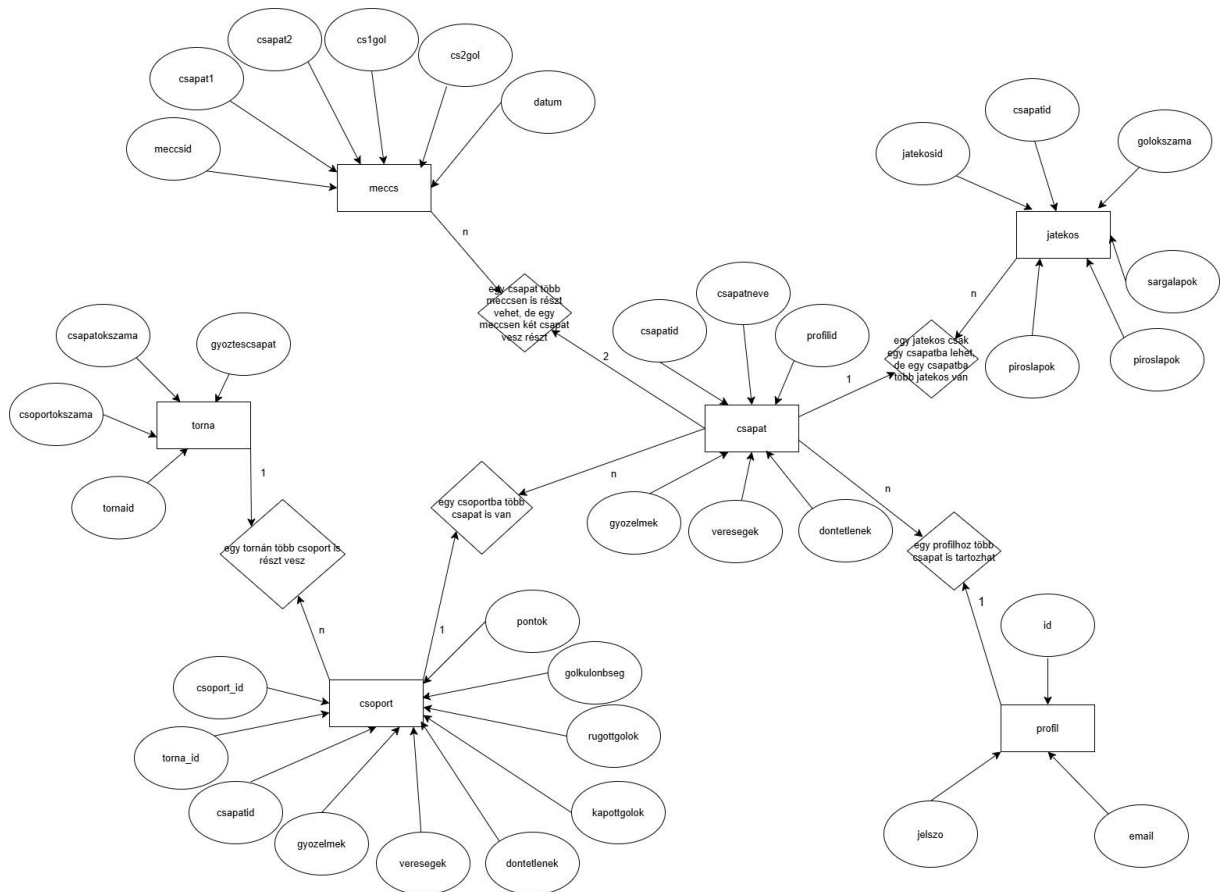
4. Kapcsolatok azonosítása: következő lépés a relációk meghatározása az entitások között.

- Egy-egy kapcsolat (1:1)
- Egy-több kapcsolat (1:N)
- Több-Több kapcsolat (N:M)

Példa:

- Profil és Meccs (1:N) : egy profil több meccset is elmenthet
- Csapat és Meccs (1:N) : egy csapat több meccsen is részt vehet
- Csapat és Játékos (1:N) : egy csapatban több játékos is van
- Játékos és Meccs (N:M) : a Csapat tábla miatt közvetett kapcsolat van közöttük

5. ER-Diagram készítése:



6. Normalizálás: célja az adatok ismétlődésének csökkentése és a tárolt adatok eredetiségének, pontosságának megőrzése. A táblák ilyenkor felosztásra kerülnek, így biztosítják a logikai adatstuktúrát.

Példa:

A Meccs táblába csak a csapat_id-t tároljuk, viszont a Csapat attribútumait külön táblába tároljuk.

7. Adatbázis sémája:

```
CREATE TABLE profil (
  id INT AUTO_INCREMENT PRIMARY KEY,
  email VARCHAR(255) NOT NULL UNIQUE,
  jelszo VARCHAR(255) NOT NULL
);
```

```
CREATE TABLE csapat (
  csapatid INT AUTO_INCREMENT PRIMARY KEY,
  profilid INT NOT NULL,
  gyozelmek INT DEFAULT 0,
  veresegek INT DEFAULT 0,
  dontetlenek INT DEFAULT 0,
  FOREIGN KEY (profilid) REFERENCES profil(id)
);
```

```
CREATE TABLE jatekos (
```

```
jatekosid INT AUTO_INCREMENT PRIMARY KEY,
csapatid INT NOT NULL,
golokszama INT DEFAULT 0,
sargalapok INT DEFAULT 0,
piroslapok INT DEFAULT 0,
FOREIGN KEY (csapatid) REFERENCES csapat(csapatid)
);
```

```
CREATE TABLE meccs (
meccsid INT AUTO_INCREMENT PRIMARY KEY,
csapat1 INT NOT NULL,
csapat2 INT NOT NULL,
cs1gol INT DEFAULT 0,
cs2gol INT DEFAULT 0,
datum DATE NOT NULL,
FOREIGN KEY (csapat1) REFERENCES csapat(csapatid),
FOREIGN KEY (csapat2) REFERENCES csapat(csapatid)
);
```

```
CREATE TABLE torna (
tornaid INT AUTO_INCREMENT PRIMARY KEY,
csoportokszama INT DEFAULT 0,
csapatokszama INT DEFAULT 0,
gyoztescsapat INT,
FOREIGN KEY (gyoztescsapat) REFERENCES csapat(csapatid)
);
```

```
CREATE TABLE csoport (
csoportid INT AUTO_INCREMENT PRIMARY KEY,
tornaid INT NOT NULL,
csapatid INT NOT NULL,
gyozelmek INT DEFAULT 0,
veresegek INT DEFAULT 0,
dontetlenek INT DEFAULT 0,
kapottgolok INT DEFAULT 0,
rugottgolok INT DEFAULT 0,
golkulonbseg INT DEFAULT 0,
pontok INT DEFAULT 0,
FOREIGN KEY (tornaid) REFERENCES torna(tornaid),
FOREIGN KEY (csapatid) REFERENCES csapat(csapatid)
);
```

8. Létrehozása és tesztelése:

```
INSERT INTO profil (email, jelszo) VALUES
('team1@example.com', 'hashedpassword1'),
('team2@example.com', 'hashedpassword2'),
('team3@example.com', 'hashedpassword3'),
('team4@example.com', 'hashedpassword4');
```

```
INSERT INTO csapat (profilid, gyozelmek, veresegek, dontetlenek, csapatneve)
VALUES
```

```
(1, 5, 2, 3, 'valami1'),
(2, 4, 4, 2, 'valami2'),
(3, 6, 3, 1, 'valami3'),
(4, 2, 5, 3, 'valami4');
```

```
INSERT INTO jatekos (csapatid, golokszama, sargalapok, piroslapok,
jatekos_neve) VALUES
```

```
(1, 10, 1, 0, 'valaki1'),
(1, 5, 2, 0, 'valaki2'),
(2, 7, 0, 1, 'valaki3'),
(2, 3, 1, 0, 'valaki4'),
(3, 12, 0, 0, 'valaki5'),
(3, 8, 1, 1, 'valaki6'),
(4, 4, 2, 0, 'valaki7'),
(4, 1, 3, 0, 'valaki8');
```

```
INSERT INTO meccs (csapat1, csapat2, cs1gol, cs2gol, datum) VALUES
```

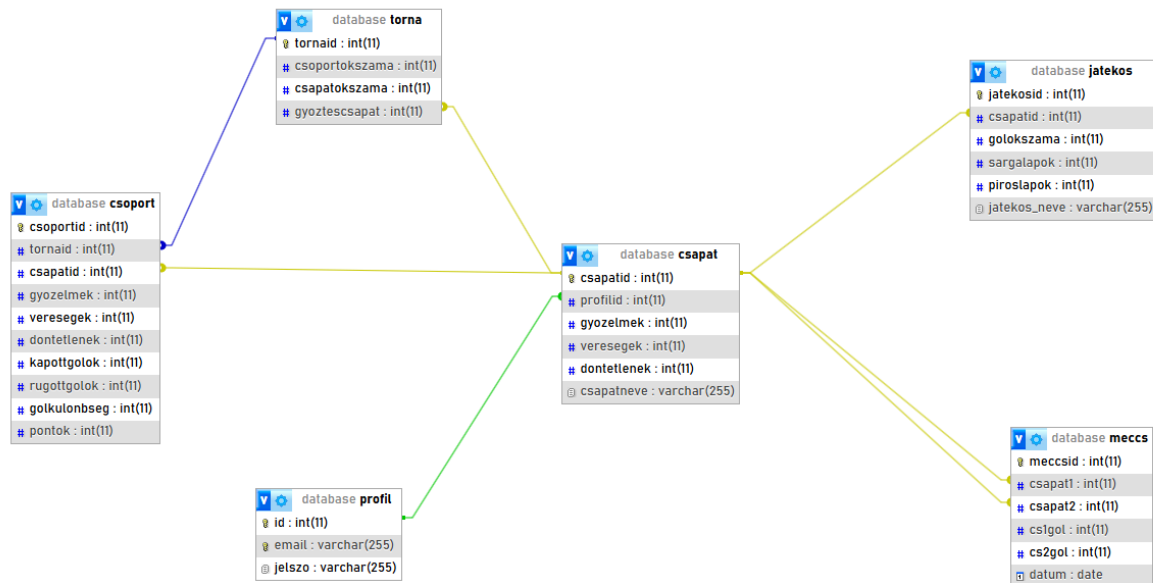
```
(1, 2, 3, 1, '2024-11-10'),
(3, 4, 2, 2, '2024-11-11'),
(1, 3, 0, 1, '2024-11-12'),
(2, 4, 1, 0, '2024-11-13');
```

```
INSERT INTO torna (csoportokszama, csapatokszama, gyoztescsapat) VALUES
```

```
(2, 4, 3),
(1, 4, NULL);
```

```
INSERT INTO csoport (tornaid, csapatid, gyozelmek, veresegek, dontetlenek,
kapottgolok, rugottgolok, golkulonbseg, pontok) VALUES
```

```
(1, 1, 2, 1, 1, 4, 6, 2, 7),
(1, 2, 1, 2, 1, 5, 4, -1, 4),
(1, 3, 3, 0, 1, 2, 7, 5, 10),
(1, 4, 0, 3, 1, 7, 3, -4, 1),
(2, 1, 1, 1, 0, 3, 4, 1, 3),
(2, 2, 1, 0, 1, 2, 3, 1, 4),
(2, 3, 0, 1, 1, 4, 2, -2, 1),
(2, 4, 0, 0, 2, 3, 3, 0, 2);
```



1. Profilok és csapataik eredményének kilistázása

SELECT p.email, c.gyozelmek, c.veresegek, c.dontetlenek FROM profil p INNER JOIN csapat c ON p.id = c.profilid;

email	gyozelmek	veresegek	dontetlenek
team1@example.com	5	2	3
team2@example.com	4	4	2
team3@example.com	6	3	1
team4@example.com	2	5	3