

# Exemplu diagrame proiect -- FOTBAL

## 1. Descrierea modelului real, a utilității acestuia și a regulilor de funcționare.

O aplicație pentru campionatele de fotbal din Romania.

## 4. Descrierea relațiilor, incluzând precizarea cardinalității acestora.

--puteți da explicații mai sumare în proiecte ca în exemplul cu hotel de animale. Aici sunt exemple pentru modul în care se completează cardinalitățile.

--cardinalitățile maxime sunt indicate cu simbolurile M, N, 1

--cardinalitățile minime sunt indicate în paranteză cu simbolurile 0 și 1.

--simbolurile din diagrama conceptuală:



maxim - M    minim - 0    maxim - 1    minim - 0    etc.

-- PUTEȚI COMPLETA DOAR CARDINALITĂȚI MAXIME ÎN DIAGRAMA

CONCEPTUALĂ. Obs: cardinalitățile minime sunt utile pentru plasa cheia externă în relații one-to-one sau pentru a stabili dacă o cheie externă trebuie să fie NOT NULL, ca în exemplele de mai jos.

- echipa – stadion **one-to-one**

Câte stadioane *poate* avea o echipa – 1

Câte stadioane *trebuie* (este obligatoriu) să aibă o echipa – 0

De câte echipe *poate* (maxim) fi administrat un stadion – 1

De câte echipe *trebuie* (minim) să fie administrat un stadion - 1

În diagrama conceptuală relația devine **cheie externă** în tabelul STADION (în tabelul unde avem mai puține linii, i.e în tabelul cu cardinalitate minimă 0, nu toate echipele au stadion).

- etapa - meci **one-to-many**

Câte meciuri *poate* (maxim) avea o etapă -- M

Câte meciuri *trebuie* (minim) să aibă o etapă -- 1.

De câte etape *poate* aparține un meci – 1

De câte etape *trebuie* să aparțină un meci – 0 poate fi meci amical,

În diagrama conceptuală relația devine **cheie externă** în tabelul MECI (în tabelul corespunzător cardinalității many). Cheia externă etapa\_id poate fi necompletată (NULL) în cazul meciurilor amicale.

- meci – arbitru **many-to-many**

Câți arbitri *pot* arbitra un meci -- M.

Câți arbitri *trebuie* să arbitreze un meci, arbitrul principal -- 1

La câte meciuri poate participa un arbitru – M.

La câte meciuri trebuie să participe un arbitru – 1.

În diagrama conceptuală relația devine **tabel asociativ**. Cheia primară a tabelului asociativ este compusă din cele două chei externe id\_arbitru, id\_meci. Se poate adăuga în tabelul asociativ coloana tip: principal, asistent.

- Joaca **relație de tip 3**.

Desi un jucator este înregistrat **în prezent** la un club, în etape precedente ar fi putut fi înregistrat la alt club. Pentru a păstra ca istoric componența echipelor este nevoie de o relație de tip 3: Jucătorul ... a jucat în meciul ... pentru echipa ....

În diagrama conceptuală relația de tip 3 se implementează ca **tabel asociativ** cu cheie primară artificială (cod auto- incrementat) și trei chei externe.

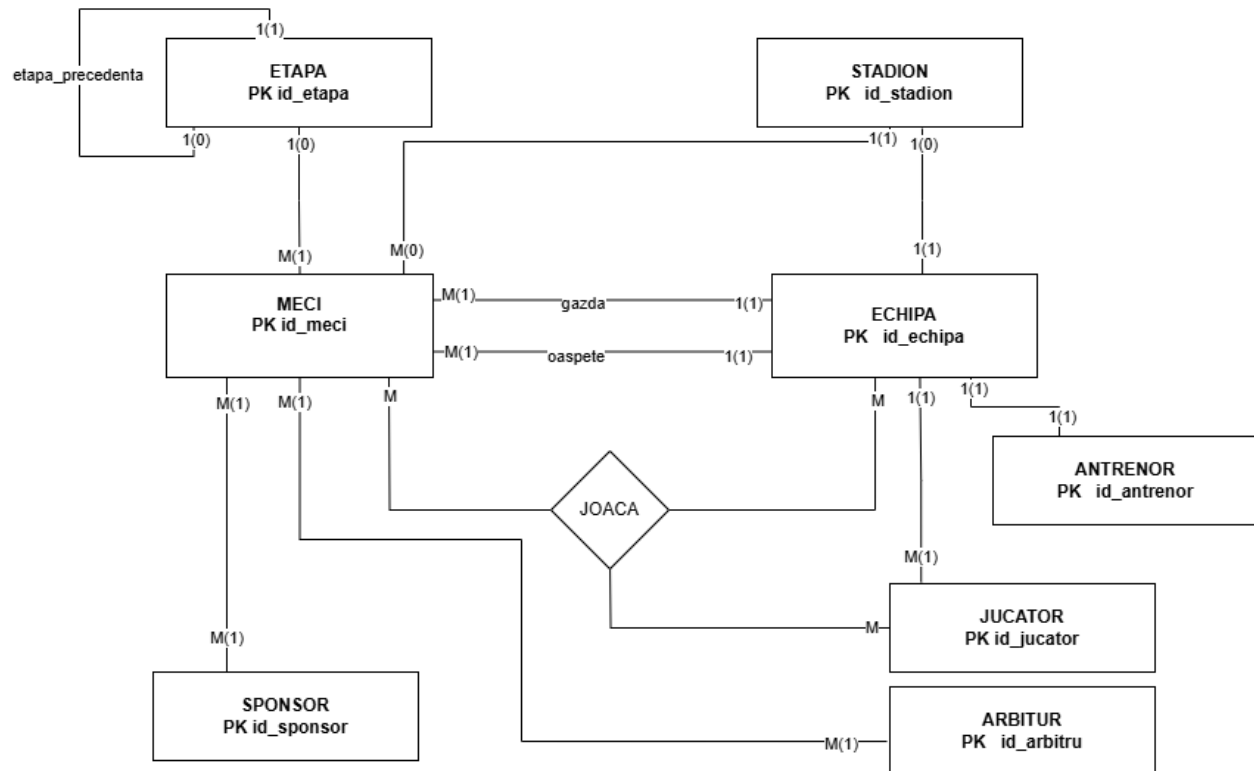
- O **etapa** este precedata de alta **etapa**. (exemplu etapa retur 2023 - etapa tur 2024 ).

**relație recursiva**. one-to-one

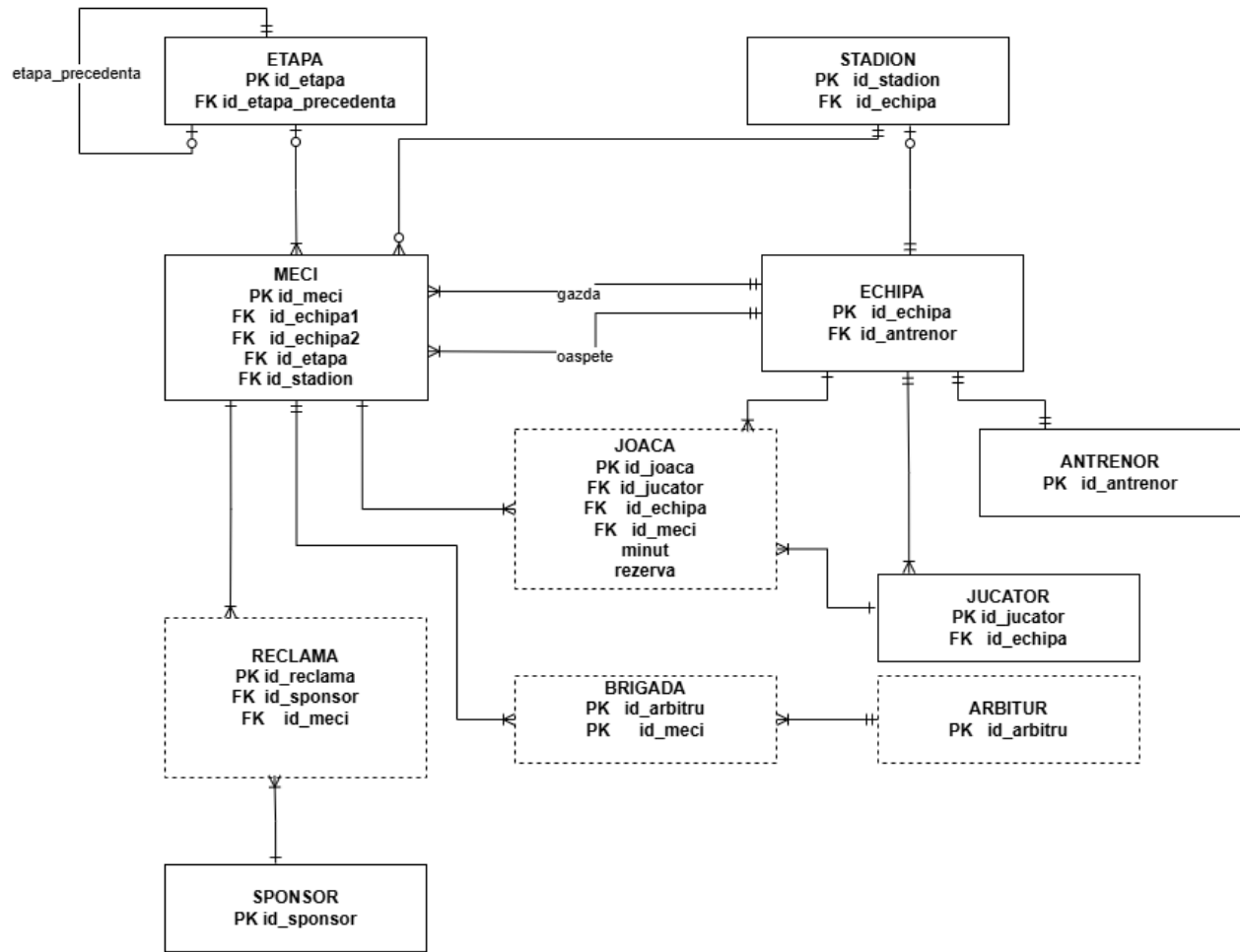
cate etape *pot* preceda o etapa – 1. cate etape *trebuie (este obligatoriu)* sa preceadă o etapa – 0

Relația se transformă în **cheie externă** în tabelul etapa care va face referire către cheia primară a tabelului etapa.

## 6. Realizarea diagramei entitate-relație corespunzătoare descrierii de la punctele 3-5.



7. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6. Diagrama conceptuală obținută trebuie să conțină minimum 7 tabele (fără considerarea subentităților), dintre care cel puțin un tabel asociativ.



### BONUS:

Exemple de interogări (detalii în laboratoarele următoare):

- Lista meciurilor cu denumirile cluburilor:

```
SELECT id_meci, e1.denumire echipa1, e2.denumire echipa1,
FROM meci m JOIN echipa e1 ON e1. echipa_id = m. echipa_id1
JOIN echipa e2 ON e2. echipa_id = m. echipa_id2;
```

- Lista meciurilor, incluzand meciurile amicale pentru care nu este asignata etapa.

```
SELECT m.id_meci, e.denumire
```

```
FROM etapa e RIGHT JOIN meci m  
ON m.id_etapa = e.id_etapa
```