**Proiect Baze de Date**

Aplicatie pentru monitorizarea sezonului 2023-2024 din liga de baschet NBA

**Cuprins**

[**1.** **Descrierea Cerintelor** 1](#_Toc155915687)

[**2.** **Structura Bazei de Date** 3](#_Toc155915688)

[2.1. Diagrama Entitate – Asociere (optional) 3](#_Toc155915689)

[2.2. Schema Relationala 4](#_Toc155915690)

[2.3. Tipuri de Date si Constrangeri de Integritate 5](#_Toc155915691)

[2.3.1. Echipe 5](#_Toc155915692)

[2.3.2. AntrenoriEchipe 5](#_Toc155915693)

[2.3.3. Antrenori 5](#_Toc155915694)

[2.3.4. Arene 5](#_Toc155915695)

[2.3.5. Meciuri 6](#_Toc155915696)

[2.3.6. Jucatori 6](#_Toc155915697)

[**3.** **Functionarea aplicatiei** 7](#_Toc155915698)

[**4.** **Interogari** 7](#_Toc155915699)

[4.1. Interogari simple 7](#_Toc155915700)

[4.2. Interogari complexe 8](#_Toc155915701)

# **Descrierea Cerintelor**

Liga NBA are mai multe echipe si mai multe meciuri. Pentru fiecare echipa se retin datele: nume echipa, an infiintare, oras, antrenor principal, scor si manager. Pentru fiecare meci se retin datele: acasa, deplasare, data, scor, arena si castigator.

O echipa are mai multi antrenori, mai multi jucatori si o arena. Pentru fiecare antrenor se retin datele: nume antrenor, nationalitate si numar echipe. Pentru fiecare jucator se retin datele: nume, echipa, nr jersey, pozitie, inaltime, greutate, an debut, PPG (Points per Game), RPG (Rebounds per Game) si APG (Assists per Game). Pentru fiecare arena se retin datele: nume arena, dimensiune, an infiintare si adresa.

O echipa are unul sau mai multi antrenori si un antrenor apartine uneia sau mai multor echipe. O echipa are un singur antrenor principal si un antrenor poate fi antrenor principal doar pentru o echipa.

O echipa detine o singura arena si o arena este detinuta de una sau mai multe echipe.

Intr-o echipa sunt 5 sau mai multi jucatori si un jucator apartine unei singure echipe.

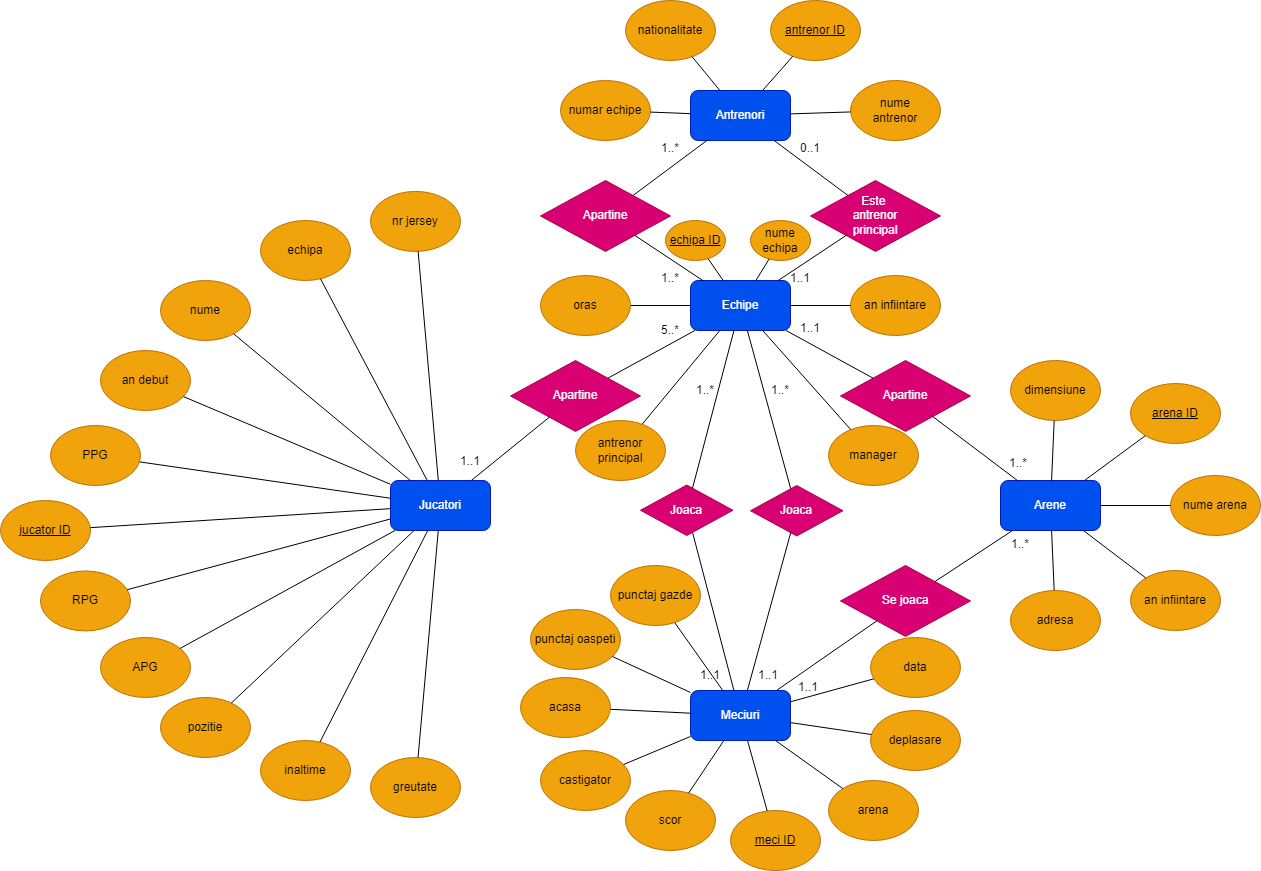
Pe o arena se joaca unul sau mai multe meciuri si un meci se joaca pe o arena.

O echipa joaca mai multe meciuri si un meci este jucat de o echipa care joaca acasa si o echipa care joaca in deplasare.

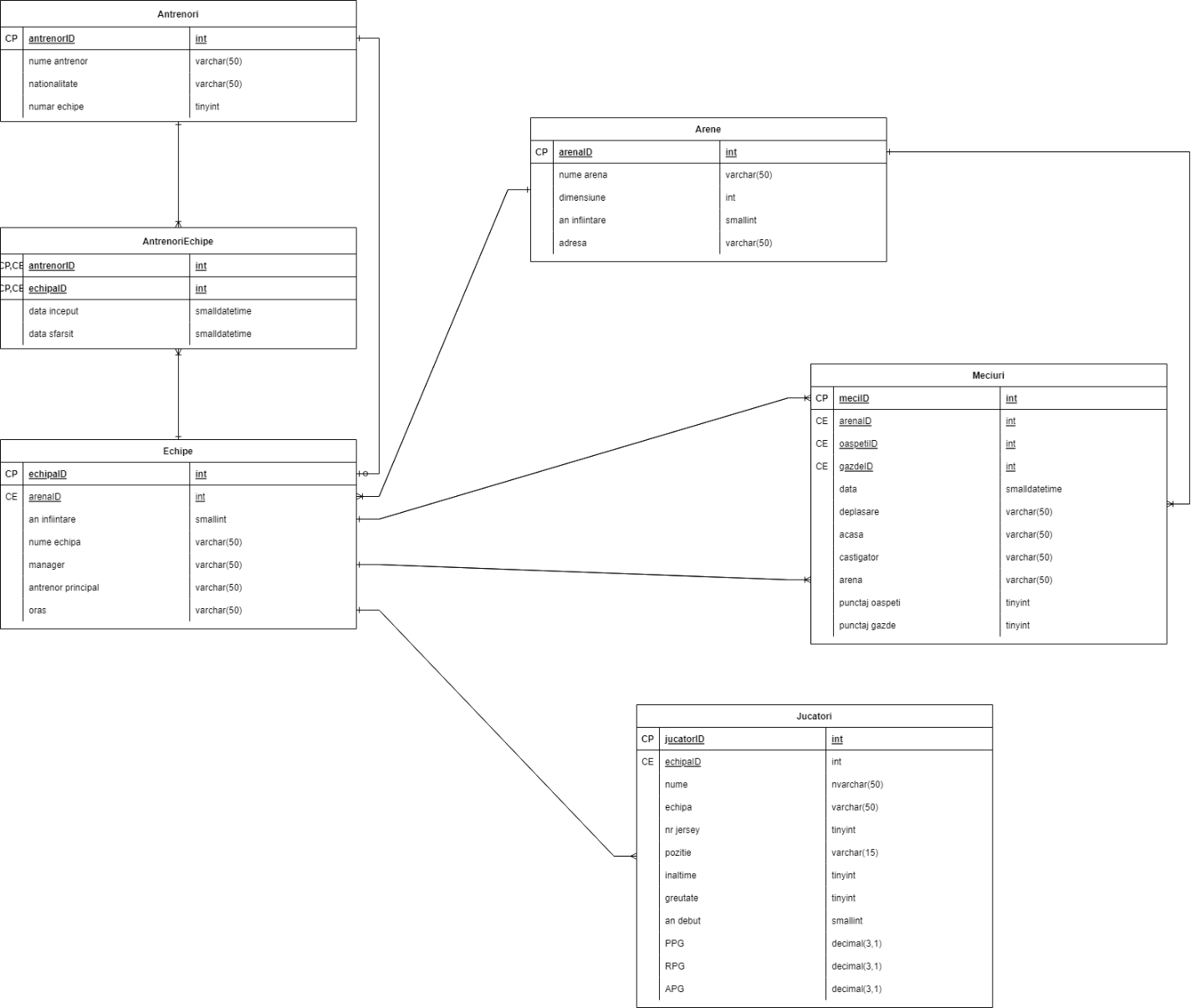
Un jucator joaca pentru o singura echipa.

# **Structura Bazei de Date**

## Diagrama Entitate – Asociere (optional)



## Schema Relationala



## Tipuri de Date si Constrangeri de Integritate

### Echipe

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Denumire** | **Tip Date** | **Not Null** | **Unique** | **Primary Key** | **Check** | **Default** | **Foreign Key** |
| **echipaID** | int | x | x | x |  |  |  |
| **arenaID** | int | x | x |  |  |  | x |
| **anInfiintare** | smallint | x |  |  |  |  |  |
| **numeEchipa** | varchar(50) | x |  |  |  |  |  |
| **manager** | varchar(50) | x |  |  |  |  |  |
| **antrenorPrincipal** | varchar(50) | x |  |  |  |  |  |
| **oras** | varchar(50) | x |  |  |  |  |  |

### AntrenoriEchipe

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Denumire** | **Tip Date** | **Not Null** | **Unique** | **Primary Key** | **Check** | **Default** | **Foreign Key** |
| **antrenorID** | int | x | x | x |  |  | x |
| **echipaID** | int | x | x | x |  |  | x |
| **dataInceput** | smalldatetime | x |  |  |  |  |  |
| **dataSfarsit** | smalldatetime | x | x |  |  |  |  |

### Antrenori

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Denumire** | **Tip Date** | **Not Null** | **Unique** | **Primary Key** | **Check** | **Default** | **Foreign Key** |
| **antrenorID** | int | x | x | x |  |  |  |
| **numeAntrenor** | varchar(50) | x |  |  |  |  |  |
| **nationalitate** | varchar(50) | x |  |  |  |  |  |
| **numarEchipe** | tinyint | x |  |  |  |  |  |

### Arene

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Denumire** | **Tip Date** | **Not Null** | **Unique** | **Primary Key** | **Check** | **Default** | **Foreign Key** |
| **arenaID** | int | x | x | x |  |  |  |
| **numeArena** | varchar(50) | x |  |  |  |  |  |
| **dimensiune** | int | x |  |  |  |  |  |
| **anInfiintare** | smallint | x |  |  |  |  |  |
| **adresa** | varchar(50) | x |  |  |  |  |  |

### Meciuri

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Denumire** | **Tip Date** | **Not Null** | **Unique** | **Primary Key** | **Check** | **Default** | **Foreign Key** |
| **meciID** | int | x | x | x |  |  |  |
| **arenaID** | int | x |  |  |  |  | x |
| **oaspetiID** | int | x |  |  |  |  | x |
| **gazdeID** | int | x |  |  |  |  | x |
| **data** | smalldatetime | x |  |  |  |  |  |
| **deplasare** | varchar(50) | x |  |  |  |  |  |
| **acasa** | varchar(50) | x |  |  |  |  |  |
| **castigator** | varchar(50) | x |  |  |  |  |  |
| **arena** | varchar(50) | x |  |  |  |  |  |
| **punctajOaspeti** | tinyint | x |  |  |  |  |  |
| **punctajGazde** | tinyint | x |  |  |  |  |  |

### Jucatori

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Denumire** | **Tip Date** | **Not Null** | **Unique** | **Primary Key** | **Check** | **Default** | **Foreign Key** |
| **jucatorID** | int | x | x | x |  |  |  |
| **echipaID** | int | x |  |  |  |  | x |
| **nume** | nvarchar(50) | x |  |  |  |  |  |
| **echipa** | varchar(50) | x |  |  |  |  |  |
| **nrJersey** | tinyint | x |  |  |  |  |  |
| **pozitie** | varchar(15) | x |  |  |  |  |  |
| **inaltime** | tinyint | x |  |  |  |  |  |
| **greutate** | tinyint | x |  |  |  |  |  |
| **anDebut** | smallint | x |  |  |  |  |  |
| **PPG** | decimal(3,1) | x |  |  |  |  |  |
| **RPG** | decimal(3,1) | x |  |  |  |  |  |
| **APG** | decimal(3,1) | x |  |  |  |  |  |

# **Functionarea aplicatiei**

Am creat o pagina web pentru interfata grafica a bazei de date utilizand React, ExpressJS si NodeJS. React a fost folosit pentru pagina web, ExpressJS si NodeJS pentru crearea server-ului ce se conecta la baza de date si sustinea endpoint-urile pentru a face request-uri de tip GET, POST, PUT si DELETE.

# **Interogari**

## Interogari simple

* + 1. Selecteaza toti jucatorii de la o anumita echipa:

SELECT Jucatori.nume

FROM Jucatori

INNER JOIN Echipe

ON Jucatori.echipaID = Echipe.echipaID

WHERE Echipe.numeEchipa = @teamName;

* + 1. Afiseaza pentru fiecare echipa numarul de meciuri jucate:

SELECT Echipe.numeEchipa, COUNT(Meciuri.meciID) AS NumarMeciuri

FROM Echipe

INNER JOIN Meciuri

ON Echipe.echipaID = Meciuri.oaspetiID

OR Echipe.echipaID = Meciuri.gazdeID

GROUP BY Echipe.numeEchipa;

* + 1. Afisarea echipelor ce au jucatori cu un numar de puncte per meci peste 20 si ai caror pozitie de joc este „forward-guard” cu numarul de pe spate „0”:

SELECT DISTINCT Echipe.numeEchipa

FROM Jucatori

JOIN Echipe

ON Jucatori.echipaID = Echipe.echipaID

WHERE Jucatori.PPG > 20;

* + 1. Afisarea tuturor datelor pentru meciurile in care a jucat un jucator specificat prin numarul sau pe tricou:

SELECT convert(varchar(10), Meciuri.data, 120) AS Data, Jucatori.nume, Jucatori.nrJersey

FROM Meciuri

JOIN Echipe

ON Meciuri.oaspetiID = Echipe.echipaID

OR Meciuri.gazdeID = Echipe.echipaID

JOIN Jucatori

ON Jucatori.echipaID = Echipa.echipaID

WHERE Jucatori.nrJersey = @jerseyNumber;

* + 1. Afisarea tuturor meciurilor impreuna cu data meciului, numele si adresa arenei, ordonate descrescator dupa punctajul total:

SELECT Meciuri.data, Arene.numeArena, Arene.adresa, (Meciuri.punctajGazde + Meciuri.punctajOaspeti) AS ScorTotal

FROM Meciuri

INNER JOIN Arene

ON Meciuri.arenaID = Arene.arenaID

ORDER BY ScorTotal DESC;

* + 1. Afisarea antrenorilor care au antrenat in arena „TD Garden”:

SELECT DISTINCT Antrenori.numeAntrenor

FROM Antrenori

JOIN AntrenoriEchipe

ON Antrenori.antrenorID = AntrenoriEchipe.antrenorID

JOIN Echipe

ON AntrenoriEchipe.echipaID = Echipe.echipaID

JOIN Meciuri

ON Echipe.echipaID = Meciuri.gazdeID

OR Echipe.echipaID = Meciuri.oaspetiID

JOIN Arene

ON Meciuri.arenaID = Arene.arenaID

WHERE Arene.numeArena = ‘TD Garden’;

## Interogari complexe

* + 1. Sa se afiseze pentru o anumita arena numele echipei si punctajul acesteia care a avut cel mai mic punctaj dintre toate meciurile jucate in arena respectiva drept gazda:

SELECT E.numeEchipa, M.punctajGazde

FROM Meciuri M

JOIN Echipe E ON M.gazdeID = E.echipaID OR M.oaspetiID = E.echipaID

WHERE M.arenaID = (SELECT arenaID FROM Arene WHERE numeArena = @arenaName)

AND M.punctajGazde = (

SELECT MIN(punctajGazde)

FROM Meciuri

WHERE arenaID = (SELECT arenaID FROM Arene WHERE numeArena = @arenaName)

);

* + 1. Afisarea echipelor si orasul lor fara meciuri castigate:

SELECT DISTINCT Echipe.numeEchipa, Echipe.oras

FROM Echipe

LEFT OUTER JOIN Meciuri

ON Echipe.echipaID = Meciuri.gazdeID OR Echipe.echipaID = Meciuri.oaspetiID

WHERE Echipe.numeEchipa NOT IN (Meciuri.castigator);

* + 1. Afisarea jucatorilor care au marcat peste media punctelor per meci in comparatie cu media echipei lor:

SELECT J.nume

FROM Jucatori J

WHERE J.PPG > (

SELECT AVG(J2.PPG)

FROM Jucatori J2

WHERE J2.echipaID = J.echipaID

);

* + 1. Afisarea echipelor care nu au jucat intr-o anumita arena drept gazde:

SELECT E.numeEchipa

FROM Echipe E

WHERE E.echipaID NOT IN (

SELECT M.gazdeID

FROM Meciuri M

WHERE M.arenaID = (SELECT A.arenaID FROM Arene A WHERE A.numeArena = @newarena)

);