

Universitatea din București  
Facultatea de Matematică și Informatică  
Departmentul de Informatică  
Master Inteligență Artificială

# **Baza de date pentru gestionarea unei librării de închiriere jocuri**

Andreea-Daniela Ene  
grupa 407

București  
February 4, 2017



# Cuprins

1	Descrierea modelului . . . . .	5
2	Constrângerile modelului . . . . .	6
3	Entități: descriere, cheie primară . . . . .	7
4	Relații: descriere, cardinalitate . . . . .	9
5	Atribute: descriere, tip, constrângeri . . . . .	12
6	Diagrama entitate-relație . . . . .	19
7	Diagrama conceptuală . . . . .	20
8	Scheme relaționale . . . . .	21
9	Normalizare . . . . .	22
10	Crearea tabelor în SQL . . . . .	25



# 1 Descrierea modelului

In acest proiect este propus un model pentru baza de date in care se poate realiza gestionarea unei librării de inchiriere jocuri.

In model, entitatile au fost organizate in patru grupuri, joc, magazin, angajat si client. Joc, grupul principal, contine diferite informatii legate de jocurile disponibile pentru inchiriere, precum tipul, categoria, producatorul, numarul de jucatori recomandati (categorie de varsta), precum si numarul de jocuri disponibile. In grupul client se regasesc informatii legate de acesa si produsele pe care el le-a inchiriat. In cel de-al treilea grup, magazin, sunt informatii despre evenimentele ce se pot desfasura in magazine, precum si departamentele din acestea. In ultimul grup, angajat, se regasesc sunt disponibile informatii despre toti angajatii companiei, pozitia pe care acestia o ocupa, precum si contractul sau nivelul salarial.

Această librărie asigura clientilor sai o gama variata de jocuri, pe care acestia le pot inchiria pentru 30 de zile, din oricare magazin din cele disponibile. Clientii pot returna jocul in oricare dintre locatiile disponibile ale librăriei, si pot participa la diferitele evenimente organizate in cadrul acestora. Fiecare client are asociat un cont in baza de date, fiind neaparat necesar sa transmita datele sale de contact, pentru eventualele intarzieri ale returnarii. Acestia pot de asemenea sa dea o nota intre 1 si 5 tuturor jocurilor pe care le-au inchiriat.

Jocurile disponibile in baza de date sunt stocate impreuna cu informatii referitoare la tipurile, categoriile, producatorii, categoriile de varsta, dar si notele oferite de clienti. Jocurile fac parte dintr-un singur tip, categorie, sau producator, insa pot exista mai multe jocuri din fiecare dintre acestea. In magazine, pot fi organizate diferite evenimente pentru clienti, in care acestia pot participa la competitii de jocuri de societate si testa jocurile lor preferate. In magazine exista diferite departamente care se ocupa de logistica, inchiriere jocuri, sau achizitie. Toti angajatii au asignat cate un contract unic, si au un pachet salarial care face parte dintr-o clasa salariala. Acestia ocupa anumite pozitii in departamentele din care fac parte.

## 2 Constrângerile modelului

Acest model respectă următoarele reguli:

- un joc poate fi de un singur tip.
- un joc poate avea un singur producător.
- un angajat are un singur contract.
- un angajat ocupă o singură poziție.
- într-un contract trebuie definit un singur salariu.
- data începerii contractului este mai mica decat data incheierii acestuia.
- valoarea salariului trebuie sa fie un numar mai mare decat 925, si este un numar intre 1000 si 9999, cu 2 zecimale.
- într-un magazin pot fi mai multe evenimente
- unui client i s-a emis cel puțin o factură.
- data emiterii unei facturi este mai mică decât data limită.
- data de returnare a jocurilor este mai mică sau egală decât data limită.
- nota dintr-un review poate fi un numar intre 1 si 5.

### 3 Entități: descriere, cheie primară

Entitățile acestui model de baze de date sunt următoarele:

- **GAME**: informații generale despre toate jocurile disponibile în magazine. Cheia primară este *game\_id*, și conține 4 chei străine, *category\_id*, *producer\_id*, *type\_id* și *players\_id*.
- **GAME\_TYPE**: în această tabelă sunt reținute informații despre tipurile de jocuri. Cheia primară este *type\_id* și nu are nicio cheie străină.
- **GAME\_CATEGORY**: o lista a tuturor categoriilor din care un joc poate face parte. Cheia primară este *category\_id* și nu conține nicio cheie străină.
- **GAME\_AGE\_GROUP**: în tabela aceasta sunt informații legate de numărul de jucători recomandați pentru fiecare joc. Cheia primară este *players\_id* și nu conține nicio cheie străină.
- **GAME\_PRODUCER**: conține toți producătorii de jocuri. Cheia primară este *players\_id* și nu conține nicio cheie străină.
- **GAME\_REVIEW**: notele, pe o scară de la 1 la 5, evaluate de către clienți pentru jocurile selecționate de către aceștia. Cheia primară este *review\_id* iar cele două chei străine sunt *game\_id* și *client\_id*.
- **CLIENT**: oferă informații generale despre fiecare client în parte, precum și adresa acestora de contact. Cheia primară este *client\_id* și nu conține nicio cheie străină.
- **CLIENT\_LOAN**: în această tabelă sunt toate facturile emise unui client, precum și datele în care jocurile au fost împrumutate sau returnate. Cheia primară este *bill\_id*, iar cea străină este *client\_id*.
- **LOAN\_PRODUCTS**: tabel asociativ în care sunt reținute jocurile incluse în facturi. Cheia primară este *loan\_products\_id* iar cele străine sunt *game\_from\_store\_id* și *client\_loan\_id*.

- **STORE:** informații legate despre magazine. Cheia primară este *store\_id* și nu conține nicio cheie străină.
- **GAME\_FROM\_STORE:** conține informații legate de numărul de jocuri din fiecare magazin în parte. Cheia primară este compusă, formată din *game\_id* și *store\_id*.
- **STORE\_EVENT:** în această tabelă sunt informații despre evenimentele din magazine. Cheia primară este *event\_id* iar cea străină este *store\_id*.
- **STORE\_DEPARTMENT:** detalii despre departamentele din magazine. Cheia primară este *department\_id* iar cea străină este *store\_id*.
- **EMPLOYEE:** descrierea angajaților din magazine. Aceștia ocupă o poziție într-un departament, și le este asociat un contract. Cheia primară este *employee\_id* iar cele străine sunt *department\_id*, *contract\_id* și *corporate\_title\_id*.
- **EMPLOYEE\_CORPORATE\_TITLE:** în această tabelă sunt pozițiile ocupate în fiecare departament în parte. Cheia primară este *corporate\_title\_id* și nu conține nicio cheie străină.
- **EMPLOYEE\_CONTRACT:** tabelă cu toate contractele asociate angajaților ce conține datele de început și încheiere ale acestora. Cheia primară este *contract\_id* iar cea străină este *salary\_level\_id*.
- **EMPLOYEE\_SALARY\_LEVEL:** informații legate de salariile angajaților, asociate tabelului contract. Cheia primară este *salary\_level\_id*.



## 4 Relații: descriere, cardinalitate

Relațiile dintre entități sunt următoarele:

- **game\_has\_game\_type**: relație ce face legătura între joc și tipul acestuia. Cardinalitatea minimă este 1:0, unui tip nu îi este neapărat asociat un joc, dar un joc trebuie să aibă un tip. Cardinalitatea maximă este m:1, un joc trebuie să fie asociat unui singur tip, însă mai multe jocuri pot fi asociate aceluiași tip.
- **game\_belongs\_to\_game\_category**: relație ce face legătura între joc și categoria acestuia. Cardinalitatea minimă este 1:1, cel puțin o categorie este asociată un joc, iar un joc trebuie să aibă o minim o categorie. Cardinalitatea maximă este m:n, un joc este asociat mai multor categorii, iar mai multe jocuri pot fi asociate aceluiași categorii.
- **game\_created\_by\_game\_producer**: relație ce face legătura între joc și producătorul acestuia. Cardinalitatea minimă cât și cea maximă este 1:1, un joc are un singur producător, iar un producător este asociat unui singur joc.
- **game\_has\_game\_review**: relație care leagă jocurile de review-uri. Cardinalitatea minimă este 0:0, un joc nu trebuie să aibă neapărat un review, iar un review nu trebuie să fie dat neapărat unui joc. Cardinalitatea maximă este 1:n, un joc poate avea cel puțin un review, iar același review nu poate fi dat mai multor jocuri de către aceeași persoană.
- **game\_is\_in\_game\_age\_group**: relație care face legătura între joc și numărul de jucători recomandat. Cardinalitatea minimă este 1:0, un joc are un număr de jucători recomandat, însă un număr de jucători nu este neapărat asociat tuturor jocurilor. Cardinalitatea maximă este 1:m, un joc are un număr de jucători, iar un număr de jucători este recomandat pentru mai multe jocuri.
- **client\_writes\_game\_review**: relație ce face legătura între clienți și review-urile pe care aceștia le-au dat jocurilor. Cardinalitatea minimă este 0:0, un client nu trebuie neapărat

să fi scris un review, iar un review nu trebuie să fi fost scris neapărat de un client. Cardinalitatea maximă este  $m:n$ , un client poate scrie mai multe review-uri, și același review poate fi scris de mai mulți clienți. (Review-ul este o notă pe o scară de la 1 la 5)

- **client\_makes\_client\_loan\_with\_loan\_products**: relație care leagă clienții de facturile cu jocurile pe care aceștia le-au împrumutat, prin intermediul tabelului asociativ `loan_products`. Cardinalitatea minimă este  $1:1$ , un client trebuie să aibă cel puțin o factură cu cel puțin un produs, iar cardinalitatea maximă este  $1:n$ , un client poate avea mai multe facturi cu mai multe produse, însă mai mulți clienți nu pot avea aceeași factură.
- **game\_is\_available\_in\_store**: relație ce leagă jocurile de stocul din magazine. Cardinalitatea minimă este  $1:1$ , trebuie să fie cel puțin un joc ca să poată exista în baza de date, iar într-un stoc este cel puțin un joc. Cardinalitatea maximă este  $1:n$ , un joc poate face parte din mai multe stocuri din diferite magazine.
- **store\_has\_game**: relație ce leagă magazinele de stocuri. Cardinalitatea minimă este  $1:1$ , într-un magazin trebuie să fie cel puțin un joc ce face parte dintr-un stoc, iar un stoc este cel puțin într-un magazin. Cardinalitatea maximă este  $1:n$ , într-un magazin pot fi mai multe jocuri ce fac parte din diferite stocuri, însă nu poate fi același stoc în mai multe magazine.
- **store\_holds\_store\_event**: relație ce face legătura între magazine și evenimentele din acestea. Cardinalitatea minimă este  $0:1$ , un magazin nu trebuie să aibă neapărat un eveniment asociat, însă un eveniment trebuie să fie planuit pentru un magazin. Cardinalitatea maximă este  $m:n$ , un eveniment poate fi asociat mai multor magazine, iar mai multe evenimente pot fi asociate aceluiași magazin.
- **store\_has\_store\_department**: relație ce leagă magazinele de departamente. Cardinalitatea minimă este  $1:0$ , un departament nu se află neapărat într-un magazin, însă un magazin trebuie să aibă cel puțin un departament. Cardinalitatea maximă este  $m:n$ , un magazin poate avea mai multe departamente, și un departament poate fi în mai multe magazine.
- **employee\_works\_in\_department**: relație ce face legătura între departament și angajați. Cardinalitatea minimă este  $1:1$ , un angajat face parte dintr-un singur departament și

intr-un departament este cel puțin un angajat. Cardinalitate maxima este 1:n, intr-un departament pot fi mai multi angajat, dar un angajat face parte dintr-un singur departament.

- **employee\_has\_employee\_corporate\_title:** relație ce face legătura între angajat și funcția acestuia. Cardinalitatea minima este 1:0, o funcție nu este neapărat deținută de un angajat, însă un angajat trebuie să aibă o funcție. Cardinalitatea maximă este m:1, un angajat trebuie să dețină o singură funcție, iar o funcție poate fi deținută de mai mulți angajați.
- **employee\_has\_contract:** relație care leagă angajații de contractele individuale de muncă. Cardinalitatea minimă cât și cea maximă este 1:1, un contract este asociat unui singur angajat, iar un angajat poate avea un singur contract.
- **contract\_has\_employee\_salary\_level:** relație care leagă contractele de salariile angajatorilor. Cardinalitatea minimă cât și cea maximă este 1:1, un contract trebuie să aibă un salariu, iar un salariu este asociat unui singur contract.

## 5 Atribute: descriere, tip, constrângeri

Atributele tabelelor sunt următoarele:

### Game

Nume	Tip	Descriere
game_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
name	VARCHAR(128)	Numele jocului
thumbnail	TEXT(256)	Link-ul catre imaginea copertei.
description	TEXT(500)	Descrierea jocului.
year_published	INT(4)	Anul in care jocul a fost publicat.
category_id	INT	Cheie externă, corespunzătoare cheii primare din tabelul category. Este specificată categoria din care face parte un joc.
producer_id	INT	Cheie externă, corespunzătoare cheii primare din tabelul producer. Este specificat producatorul unui joc.
type_id	INT	Cheie externă, corespunzătoare cheii primare din tabelul type. Este specificat tipul din care face parte un joc.
players_id	INT	Cheie externă, corespunzătoare cheii primare din tabelul players. Este specificat numărul de jucători recomandat pentru un joc.

### Game\_type

Nume	Tip	Descriere
type_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
name	VARCHAR(45)	Numele tipului din care face parte un joc
description	VARCHAR(45)	Descrierea tipului din care face parte un joc

### Game\_category

Nume	Tip	Descriere
category_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
name	VARCHAR(45)	Numele categoriei din care face parte un joc
description	VARCHAR(45)	Descrierea categoriei din care face parte un joc

### Game\_producer

Nume	Tip	Descriere
producer_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
producer_name	VARCHAR(45)	Numele producătorului unui joc
country	VARCHAR(45)	Țara de origine a producătorului

### Game\_age\_group

Nume	Tip	Descriere
players_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
age_min	INT(2)	Varsta minima recomandata pentru un joc
age_max	INT(2)	Varsta maxima recomandata pentru un joc

### Game\_review

Nume	Tip	Descriere
review_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
grade	NUMBER(1)	Nota, pe o scara de la 1 la 5, oferită unui joc de către un client.
client_id	INT	Cheie externă, corespunzătoare cheii primare din tabelul client. Este specificat clientul care a facut un review.
game_id	INT	Cheie externă, corespunzătoare cheii primare din tabelul game. Este specificat jocul caruia i s-a facut un review.

### Client

Nume	Tip	Descriere
client_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
name	VARCHAR(45)	Numele clientului
phone	INT(10)	Numarul de telefon al clientului, fara prefix
e-mail	VARCHAR(50)	Adresa de e-mail a clientului
address	TEXT(50)	Adresa de contact a clientului

### Client\_loan

Nume	Tip	Descriere
client_loan_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
start_date	DATE	Data în care a fost emisă factura
end_date	DATE	Data limită până când pot fi returnate produsele.
returned_date	DATE	Data în care au fost returnat produsele
client_id	INT	Cheie externă corespunzătoare cheii primare din tabelul client. Este specificat clientul căruia i s-a emis factura.

### Loan\_products

Nume	Tip	Descriere
bill_products_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
game_fr_store_id	INT	Cheie externă corespunzătoare cheii primare din tabelul game_from_store. Este specificat jocul care se află într-o factură
client_loan_id	INT	Cheie externă corespunzătoare cheii primare din tabelul client_loan. Este specificată factura din care face parte un joc

### Game\_from\_store

Nume	Tip	Descriere
game_id	INT	Cheie primara compusa corespunzătoare cheii primare din tabelul game. Este specificat jocul care se află într-un magazin
store_id	INT	Cheie primara compusa corespunzătoare cheii primare din tabelul store. Este specificat magazinul in care sunt disponibile anumite jocuri
is_available	BOOL	Valoare booleana ce este 1 daca produsul este disponibil, si 0 daca nu a fost inchiriat si nu este disponibil. Valoarea 1 este default.

### Store

Nume	Tip	Descriere
store_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
name	VARCHAR(64)	Numele unui magazin
address	VARCHAR(45)	Adresa in care se afla magazinul

### Store\_event

Nume	Tip	Descriere
event_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
name	VARCHAR(128)	Numele unui eveniment dintr-un magazin
description	TEXT(500)	Descrierea evenimentului
date	DATE	Data in care se desfasoara evenimentul
store_id	INT	Cheie externă corespunzătoare cheii primare din tabelul store. Este specificat magazinul in care se va desfasura evenimentul
game_id	INT	Cheie externă corespunzătoare cheii primare din tabelul game. Este specificat jocul ce va fi disponibil pentru eveniment



## Store\_department

Nume	Tip	Descriere
department_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
name	VARCHAR(128)	Numele unui departament dintr-un magazin
store_id	INT	Cheie externă corespunzătoare cheii primare din tabelul store. Este specificat magazinul din care face parte un departament

## Employee

Nume	Tip	Descriere
employee_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
name	VARCHAR(128)	Numele unui angajat dintr-un magazin
PIN	INT(10)	Codul numeric personal al unui angajat
Address	TEXT(100)	Adresa de contact al unui angajat
Phone	INT(10)	Numarul de telefon al angajatului, fara prefix
department_id	INT	Cheie externă corespunzătoare cheii primare din tabelul department. Este specificat departamentul în care lucreaza un angajat
contract_id	INT	Cheie externă corespunzătoare cheii primare din tabelul contract. Este specificat contractul individual de munca al angajatului
coporate.title	INT	Cheie externă corespunzătoare cheii primare din tabelul coporate.title. Este specificata pozitia pe care un angajat o ocupa într-un departament

### Employee\_corporate\_title

Nume	Tip	Descriere
corporate_title_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
title	VARCHAR(30)	Numele pozitiei unui angajat
description	TEXT(500)	Descrierea pozitiei unui angajat

### Employee\_contract

Nume	Tip	Descriere
contract_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
start_date	DATE	Data in care a fost intocmit contractul unui angajat si a intrat in vigoare
end_date	DATE	Data in care a fost reziliat contractul
salary_level_id	INT	Cheie externă corespunzătoare cheii primare din tabelul employee_salary_level. Este specificat nivelul salarial disponibil in contract

### Employee\_salary\_level

Nume	Tip	Descriere
salary_level_id	INT	Cheie primară. Numărătoarea se face în mod crescător, începând cu 1
value	NUMERIC(6,2)	Suma de bani exprimata in RON care este acordata unui angajat

## 6 Diagrama entitate-relație

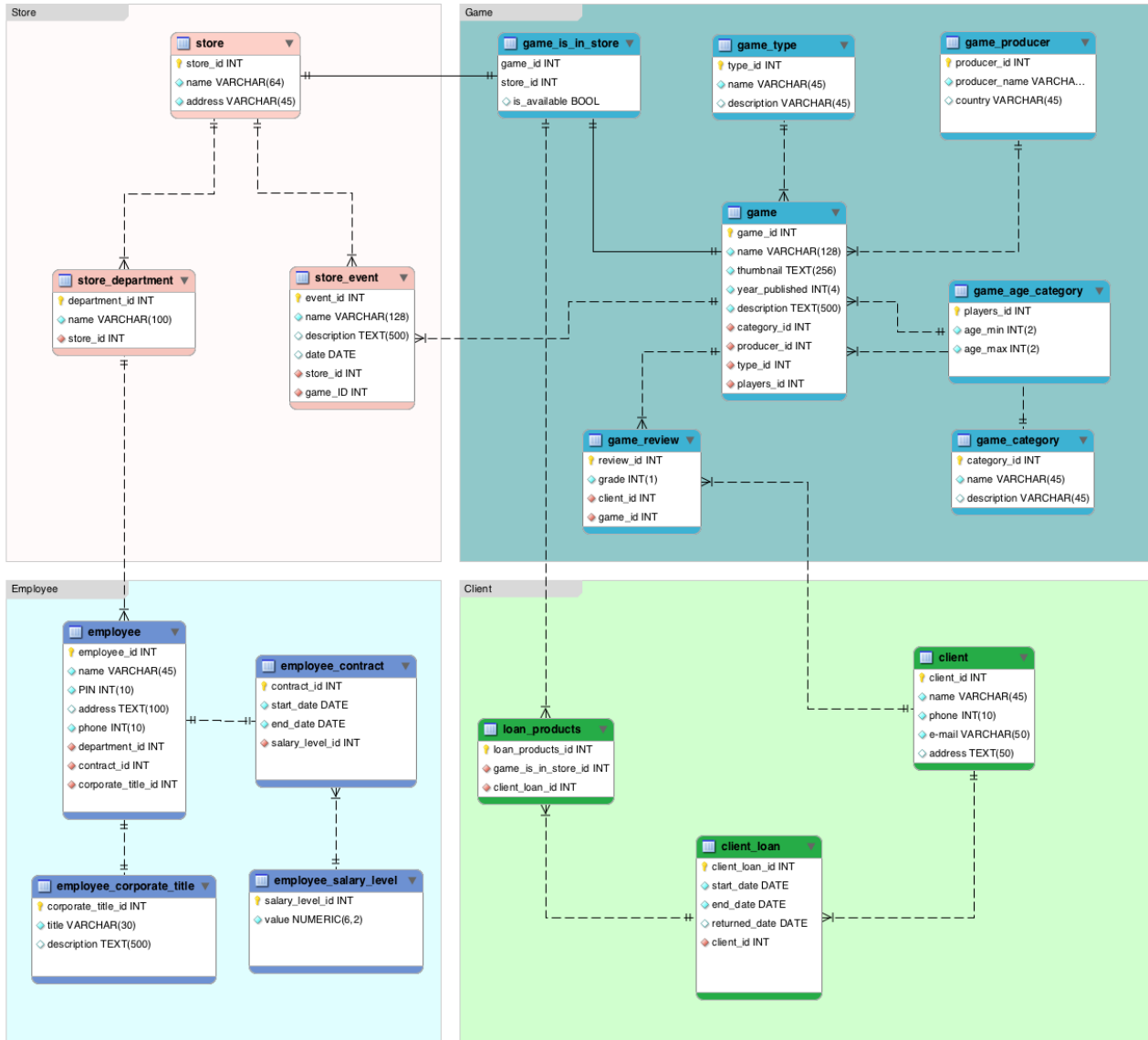


Figure 6.1: Diagrama entitate-relație.

## 7 Diagrama conceptuală

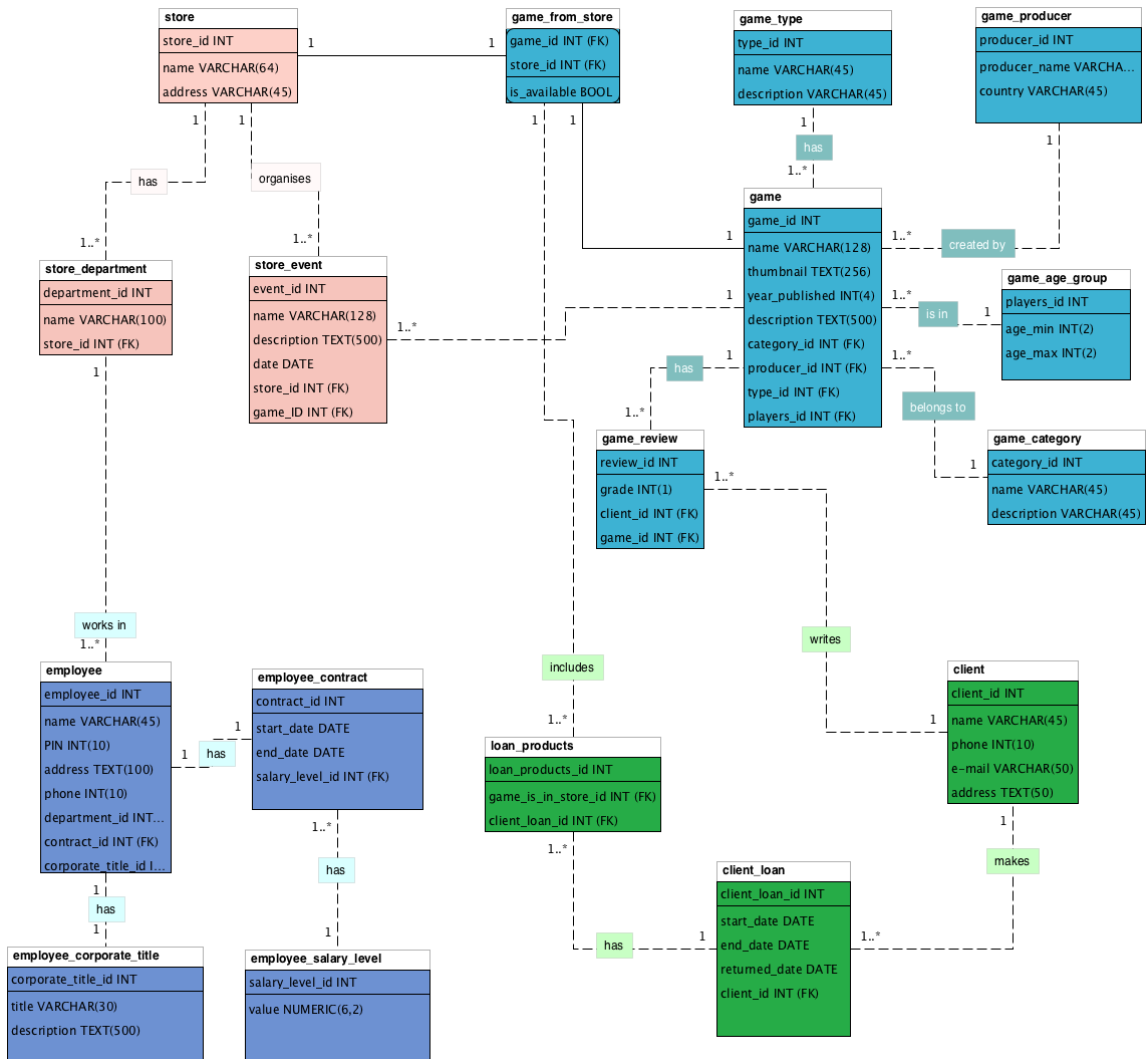


Figure 7.1: Diagrama conceptuală.

## 8 Scheme relaționale

Schemele relaționale sunt următoarele:

- **game:** #game\_id, name, thumbnail, year\_published, description, category\_id, producer\_id, type\_id, players\_id.
- **game\_type:** #type\_id, name, description.
- **game\_category:** #category\_id, name, description.
- **game\_age\_group:** #players\_id, number.
- **game\_producer:** #producer\_id, producer\_name, country.
- **game\_review:** #review\_id, grade, client\_id, game\_id.
- **client:** #client\_id, name, phone, e-mail, address.
- **client\_loan:** #client\_loan\_id, start\_date, end\_date, returned\_date, client\_id.
- **loan\_products:** #loan\_products\_id, game\_from\_store\_id, client\_loan\_id.
- **game\_from\_store:** #game\_id, store\_id, is\_available.
- **store:** #store\_id, name, address.
- **store\_event:** #event\_id, name, description, date, store\_id, game\_id.
- **store\_department:** #department\_id, name, store\_id.
- **employee:** #employee\_id, name, PIN, address, phone, department\_id, contract\_id, corporate\_title\_id.
- **employee\_corporate\_title:** #corporate\_title\_id, title, description.
- **employee\_contract:** #contract\_id, start\_date, end\_date, salary\_level\_id.
- **employee\_salary\_level:** #salary\_level\_id, value.

## 9 Normalizare

Aceasta bază de date a fost proiectată cu scopul de a reprezenta cât mai corect informațiile iar posibilitatea de a ajunge la informații eronate să se fie diminuată. Pentru a ajunge la această performanță, au fost folosite diferite tehnici de normalizare. Aceasta presupune urmarea unor reguli prin care baza de date se poate normaliza până la un anumit grad. Dacă o cerință nu este satisfăcută, relația trebuie dezcompusă în mai multe relații, care individual satisfac anumite cerințe.

- **Forma nenormalizată (UNF):** O tabelă care conține una sau mai multe grupuri repetitive. (Fig. 9.1)

**Store Department (UNF)**

Name	Employee	Corporate Title
Human Resources	Melvin, Edward, Alex	HR Partner
IT	Michael	IT Consultant
Marketing	John	Marketing Specialist

Figure 9.1: Forma nenormalizată.

- **Forma Normala 1 (FN1):** Relație în care la intersecția oricărei linii cu oricare coloană găsim un câmp care conține exact o valoare. Toate tabelele formate se afla cel puțin în forma FN1. (Fig. 9.2)

Store Department (FN1)

Name	Employee	Corporate Title
Human Resources	Melvin	HR Partner
Human Resources	Edward	HR Partner
Human Resources	Alex	HR Partner
IT	Michael	IT Consultant
Marketing	John	Marketing Specialist

Figure 9.2: Forma Normala 1.

- **Forma Normala 2 (FN2):** Dacă este în FN1 și orice atribut noncheie (care nu face parte din cheia primară) este dependent funcțional complet de cheia primară a relației. (Fig 9.3)

Store Department (FN1)

Name	Employee	Corporate Title
Human Resources	Melvin	HR Partner
Human Resources	Edward	HR Partner
Human Resources	Alex	HR Partner
IT	Michael	IT Consultant
Marketing	John	Marketing Specialist

Department (FN2)

department_id	Name
1	Human Resources
2	IT
3	Marketing

Employee

employee_id	department_id	Employee Name	Corporate Title
1	1	Melvin	HR Partner
2	1	Edward	HR Partner
3	1	Alex	HR Partner
4	2	Michael	IT Consultant
5	3	John	Marketing Specialist

Figure 9.3: Forma Normala 2.

- **Forma Normala 3 (FN3):** O relație respectă FN3 dacă respectă FN2 și toate atributele noncheie sunt dependente funcțional netranzitiv de cheia primară a relației. (Fig 9.4)

### Employee

employee_id	department_id	Employee Name	Corporate Title
1	1	Melvin	HR Partner
2	1	Edward	HR Partner
3	1	Alex	HR Partner
4	2	Michael	IT Consultant
5	3	John	Marketing Specialist

### Employee

employee_id	department_id	Corporate Title	Employee Name
1	1	1	Melvin
2	1	1	Edward
3	1	1	Alex
4	2	2	Michael
5	3	3	John

### Corporate Title(FN3)

corporate_title_id	Title
1	Human Resources
2	IT
3	Marketing

Figure 9.4: Forma Normala 3.



## 10 Crearea tabelelor în SQL

### Tabela **game**

```
CREATE TABLE "game" (  
    "game_id" INT NOT NULL,  
    "name" VARCHAR(128) NOT NULL,  
    "thumbnail" VARCHAR(256) NOT NULL,  
    "year_published" SMALLINT NOT NULL,  
    "description" VARCHAR(500) NOT NULL,  
    "category_id" INT NOT NULL,  
    "producer_id" INT NOT NULL,  
    "type_id" INT NOT NULL,  
    "players_id" INT NOT NULL,  
    PRIMARY KEY ("game_id"),  
    CONSTRAINT "category_id"  
        FOREIGN KEY ("category_id")  
        REFERENCES "category" ("category_id")  
        ENABLE,  
    CONSTRAINT "producer_id"  
        FOREIGN KEY ("producer_id")  
        REFERENCES "producer" ("producer_id")  
        ENABLE,  
    CONSTRAINT "type_id"  
        FOREIGN KEY ("type_id")  
        REFERENCES "type" ("type_id")  
        ENABLE,  
    CONSTRAINT "players_id"  
        FOREIGN KEY ("players_id")  
        REFERENCES "players" ("players_id")  
        ENABLE);
```

## Tabela **game\_type**

```
CREATE TABLE "type" (  
    "type_id" INT NOT NULL,  
    "name" VARCHAR(45) NOT NULL,  
    "description" VARCHAR(45) NOT NULL,  
    PRIMARY KEY ("type_id"),  
    CONSTRAINT "name_UNIQUE" UNIQUE ("name" ));
```

## Tabela **game\_category**

```
CREATE TABLE "game\_category" (  
    "category_id" INT NOT NULL,  
    "name" VARCHAR(45) NOT NULL,  
    "description" VARCHAR(45) NOT NULL,  
    PRIMARY KEY ("category_id"),  
    CONSTRAINT "name_UNIQUE" UNIQUE ("name" ),  
    CONSTRAINT "category_id_UNIQUE" UNIQUE ("category_id" ));
```

## Tabela **game\_producer**

```
CREATE TABLE "game\_oroducer" (  
    "producer_id" INT NOT NULL,  
    "producer_name" VARCHAR(100) NOT NULL,  
    "country" VARCHAR(45) NOT NULL,  
    PRIMARY KEY ("producer_id"),  
    CONSTRAINT "producer_name_UNIQUE" UNIQUE ("producer_name" ));
```

## Tabela **game\_age\_group**

```
CREATE TABLE "game\_age\_group" (  
    "players_id" INT NOT NULL,  
    "age\_min" NUMBER(2,0) NOT NULL,  
    "age\_max" NUMBER(2,0) NOT NULL,  
    PRIMARY KEY ("players_id"));
```

## Tabela **game\_review**

```
REATE TABLE "game\_review" (  
  "review_id" INT NOT NULL,  
  "grade" NUMBER(1) NOT NULL,  
  "client_id" INT NOT NULL,  
  "game_id" INT NOT NULL,  
  PRIMARY KEY ("review_id"));
```

## Tabela **client**

```
CREATE TABLE "client" (  
  "client_id" INT NOT NULL,  
  "name" VARCHAR(45) NOT NULL,  
  "phone" NUMBER(10) NOT NULL,  
  "e-mail" VARCHAR(20) NOT NULL,  
  "address" VARCHAR(50) NULL,  
  PRIMARY KEY ("client_id"));
```

## Tabela **client\_loan**

```
CREATE TABLE "client\_loan" (  
  "client\_loan_id" INT NOT NULL,  
  "start_date" DATE NOT NULL,  
  "end_date" DATE NOT NULL,  
  "returned_date" DATE NULL,  
  "client_id" INT NOT NULL,  
  PRIMARY KEY ("bill_id"));
```

## Tabela **loan\_products**

```
CREATE TABLE "loan_products" (  
  "loan_products_id" INT NOT NULL,  
  "game_is_in_store" INT NOT NULL,  
  "client_loan_id" INT NOT NULL,  
  PRIMARY KEY ("loan_products_id"));
```

## Tabela `game_from_store`

```
CREATE TABLE "game\_from\_store" (  
  "game_id" INT NOT NULL,  
  "store_id" INT NOT NULL,  
  "is_available" NUMBER(1) DEFAULT 1 NOT NULL,,  
  PRIMARY KEY ("game\_id", "store\_id")),
```

## Tabela `store`

```
CREATE TABLE "store" (  
  "store_id" INT NOT NULL,  
  "name" VARCHAR(64) NOT NULL,  
  "address" VARCHAR(45) NOT NULL,  
  PRIMARY KEY ("store_id"));
```

## Tabela `store_event`

```
CREATE TABLE "event" (  
  "event_id" INT NOT NULL,  
  "name" VARCHAR(128) NOT NULL,  
  "description" VARCHAR(500) NULL,  
  "date" DATE NULL,  
  "store_id" INT NOT NULL,  
  "game_id" INT NOT NULL,  
  PRIMARY KEY ("event_id");
```

## Tabela `store_department`

```
CREATE TABLE "department" (  
  "department_id" INT NOT NULL,  
  "name" VARCHAR(100) NOT NULL,  
  "store_id" INT NOT NULL,  
  PRIMARY KEY ("department_id"),  
  CONSTRAINT "department_id_UNIQUE" UNIQUE ("department_id" ));
```

## Tabela **employee**

```
CREATE TABLE "employee" (  
  "employee_id" INT NOT NULL,  
  "name" VARCHAR(45) NOT NULL,  
  "PIN" NUMERIC(10,0) NOT NULL,  
  "address" VARCHAR(100) NULL,  
  "phone" NUMBER(10) NOT NULL,  
  "department_id" INT NOT NULL,  
  "contract_id" INT NOT NULL,  
  "corporate_title_id" INT NULL,  
  PRIMARY KEY ("employee_id"),  
  CONSTRAINT "name_UNIQUE" UNIQUE ("name" ),  
  CONSTRAINT "PIN_UNIQUE" UNIQUE ("PIN" ),  
  CONSTRAINT "contract_id_UNIQUE" UNIQUE ("contract_id" ),  
  CONSTRAINT "corporate_title_id_UNIQUE" UNIQUE ("corporate_title_id" ));
```

## Tabela **employee\_corporate\_title**

```
CREATE TABLE "corporate_title" (  
  "corporate_title_id" INT NOT NULL,  
  "title" VARCHAR(30) NOT NULL,  
  "description" VARCHAR(500) NOT NULL,  
  PRIMARY KEY ("corporate_title_id"));
```

## Tabela **employee\_contract**

```
CREATE TABLE "contract" (  
  "contract_id" INT NOT NULL,  
  "start_date" DATE NOT NULL,  
  "end_date" DATE NOT NULL,  
  "salary\_level_id" INT NULL,  
  PRIMARY KEY ("contract_id"));
```

## Tabela **employee\_salary\_level**

```
CREATE TABLE "salary" (  
  "salary_level_id" INT NOT NULL,  
  "value" NUMBER(4,2) NOT NULL,  
  PRIMARY KEY ("salary_id"));
```