

Lucrare Individuală nr. 1

Disciplina: Baze de Date

Efectuat: Maxim Pricop

Grupa: P-2423

Verificat: Natalia Gairunova

Teorie

Sarcina: Fișiere și baze de date. Arhitectura BD. Arhitectura ANSI SPARC.

Fișiere și Baze de Date

Organizarea informației într-un sistem informatic începe adesea cu **fișiere**: colecții de înregistrări (linii) stocate pe disc, fiecare înregistrare având câmpuri într-un anumit format, de exemplu fișiere text, foi de calcul excel, etc. Fișierele sunt adecvate pentru aplicații simple sau pentru stocare locală, dar prezintă limitări când datele trebuie partajate, validate sau integrate între mai multe aplicații: redundanță, inconsistență, lipsa unui mecanism standard de lucru și dificultăți în menținerea integrității.

Din aceste nevoi rezultă ideea că **informația organizată** (date structurate, relații între entități, reguli de validare) trebuie gestionată de un sistem proiectat special, adică o bază de date. Scopul unei BD este să păstreze datele într-o formă logică, stabilă și reutilizabilă pentru mai multe aplicații și utilizatori, oferind reguli de structură (schemă) și mecanisme care asigură corectitudinea și accesibilitatea.

Arhitectura BD

Baza de date este o colecție organizată de informații însoțită de o descriere formală a structurii sale. Această descriere (numită schemă) definește entitățile (de exemplu tabele în modelul relațional), atributele lor și regulile generale. Conținutul efectiv al bazei într-un moment dat se numește instanță. În contextul lecțiilor, noi lucrăm în principal cu modelul relațional, unde datele sunt exprimate ca relații (tabele) compuse din atribute (coloane) și tupluri (rânduri).

Arhitectura ANSI/SPARC

Când vorbim despre arhitectura unei BD ne referim la modul în care sunt separate preocupările între descrierea logică a datelor și modul efectiv de stocare. În practică, această arhitectură include: definiții și reguli la nivel conceptual, perspective specifice utilizatorilor, și modalități concrete de păstrare a datelor fizice. Scopul separării este să permită schimbări locale (de ex. reorganizarea stocării) fără a afecta aplicațiile și, invers, modificări logice controlate fără a expune complexitatea fizică utilizatorilor.

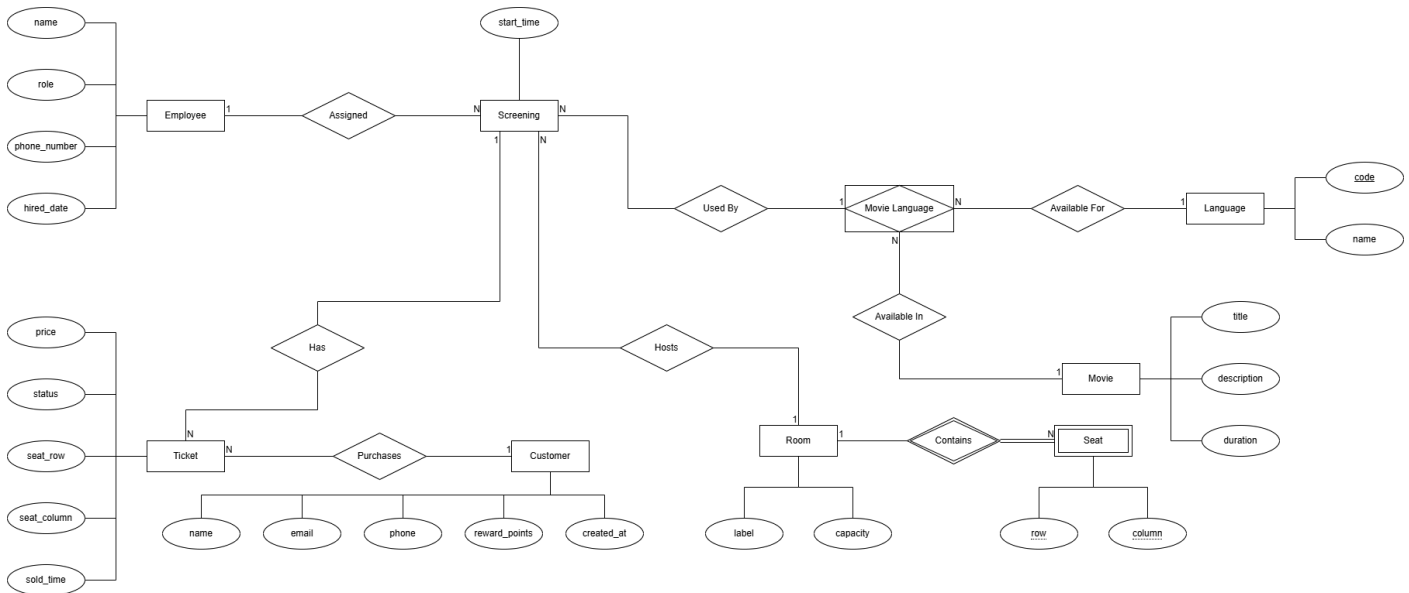
Modelul standard care formalizează această separare este ANSI/SPARC (arhitectura în trei niveluri):

1. Nivel extern - setul de vederi specifice utilizatorilor sau aplicațiilor. Fiecare utilizator vede doar o parte din date, relevantă lui.
2. Nivel conceptual - schema globală a bazei de date: entități, relații, atribute și constrângeri. Definește modelul logic al informației.
3. Nivel intern (fizic) - descrierea modului de stocare a datelor: structuri de fișiere, organizare a înregistrărilor, și alte detalii care țin de implementare.

Practică

Sarcina: Cineplex Mall.

Diagramă



Descriere

Entități și Atribute

Room (Sală)

- Ce reprezintă: Sala fizică în care se proiectează filmul.
- Atribute afișate: *label* (numele sălii), *capacity* (numărul de locuri)

Seat (Loc) - Entitate slabă

- Ce reprezintă: Un scaun fizic dintr-o sală.
- Atribute afișate: *row* (rândul), *column* (coloana)

Movie (Film)

- Ce reprezintă: Informațiile despre un film.
- Atribute afișate: *title* (titlul filmului), *duration* (durata în minute), *description* (descriere scurtă a filmului)

Language (Limbă)

- Ce reprezintă: Toate limbile disponibile în care filmele pot fi vizionate în cinema.
- Atribute afișate: *code* (codul ISO a limbii, ex: EN, RO, RU), *name* (numele complet a limbii)

Movie Language (Limba Filmului) - Entitate Asociativă

- Ce reprezintă: Perechea dintre un film și o limbă, adică o versiune a unui singur film într-o anumită limbă.
- Motivație: Asigură că proiecțiile folosesc doar limbile disponibile pentru filme și că nu pot exista proiecții cu limbi invalide pentru filmul afișat.

Screening (Proiecția unui film)

- Ce reprezintă: O afișare programată a unei versiuni a unui film (film + limbă) într-o sală la o oră fixă.
- Atribute afișate: *start_time* (timpul de început)

Customer (Client)

- Ce reprezintă: Un utilizator/client înregistrat online.
- Atribute afișate: *name* (numele), *email*, *phone* (numărul de telefon), *reward_points* (puncte bonusuri ce pot fi utilizate pentru reduceri), *created_at* (data înregistrării contului)

Employee (Angajat)

- Ce reprezintă: Angajații cinematografului.
- Atribute afișate: *name* (numele), *role* (rolul angajatului), *phone* (numărul de telefon), *hired_date* (data angajării)

Ticket (Bilet)

- Ce reprezintă: Rezervare / Bilet emis pentru o singură proiecție a unui film.
- Atribute afișate: *price* (prețul biletului), *status* (cumpărat/rezervat/anulat), *seat_row* (rândul scaunului rezervat de bilet), *seat_column* (coloana scaunului rezervat de bilet), *sold_time* (timpul când a fost vândut)

Relații

Room (1) — contains — (N) Seat

O cameră conține mai multe scaune.

Movie (1) — Available In — (N) Movie Language

Un film poate avea mai multe versiuni în diferite limbi.

Language (1) — Available For — (N) Movie Language

O limbă poate fi folosită pentru mai multe filme.

Movie Language (1) — Used By — (N) Screening

O versiune a unui film (limbă + film) poate fi proiectată de mai multe ori în timpuri diferite.

Room (1) — Hosts — (N) Screening

O sală găzduiește mai multe proiecții cu diferite filme la ore diferite.

Employee (1) — Assigned For — (N) Screening

Un angajat e responsabil pentru mai multe proiecții.

Screening (1) — Has — (N) Ticket

O proiecție are mai multe bilete ce.

Customer (1) — Purchases (N) Ticket

Un client poate cumpăra mai multe bilete pentru diferite proiecții.

Concluzie

Această lucrare pune în evidență ideea centrală că gestionarea eficientă a informației necesită trecerea de la simple colecții de date către o structură logică bine definită, însoțită de reguli care să garanteze corectitudinea și coerența datelor. Pornind de la analiza teoretică a fișierelor ca suport fizic și continuând cu conceptele fundamentale ale bazelor de date, am arătat că modelarea clară a entităților și relațiilor (prin diagrama ER realizată pentru domeniul cinema) și declararea constrângerilor necesare sunt elementele care transformă datele în informație utilizabilă.

Rezultatele obținute sunt în primul rând conceptuale: o schemă coerentă și justificată pentru un sistem de gestionare a activității unui cinema și clarificarea modului în care nivelurile descrise de modelul ANSI/SPARC leagă această schemă de stocarea fizică. Modelarea ER a permis identificarea atributelor esențiale, precum și justificarea relațiilor (de exemplu, de ce o proiecție e legată cu sălile și filme, sau de ce un bilet este legat de o proiecție și de un spectator). Aceste concluzii susțin teza lucrării: o bună proiectare logică este fundamentul oricărui sistem informațional fiabil și scalabil.

Lucrarea are, totuși, limitări clare: abordarea a rămas la nivel teoretic și de proiectare conceptuală, nu a inclus implementarea într-un SGBD, testarea cu date reale, simulări de încărcare sau scenarii de concurență (de exemplu rezervări simultane). De aceea, recomandările practice vizează pașii următori: implementarea schemei într-un SGBD pentru validare, crearea unui set de date de test pentru verificarea constrângerilor și a integrității, definirea de view-uri relevante pentru roluri (casier, management, client) și investigarea gestionării tranzacțiilor pentru operațiuni critice (rezervări/vânzări bilete). Din punct de vedere al proiectării, merită analizate dependențele funcționale pentru a verifica normalizarea și a reduce redundanțele.

Concluzia principală a acestei lucrări este că investiția în proiectarea conceptuală (ER, scheme și constrângeri) plătește prin încredere și stabilitate în sistemele care folosesc datele, un aspect esențial pentru aplicații reale precum gestionarea unui cinema. Pe termen lung, o astfel de abordare facilitează implementări robuste, extinderi funcționale și întreținere mai ușoară, oferind o bază solidă pentru continuarea studiului și pentru dezvoltări ulterioare.

Webografie

1. https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model
2. <https://app.diagrams.net>
3. <https://code.visualstudio.com>