

MINISTERUL EDUCAȚIEI ȘI CERCETĂRII ȘTIINȚIFICE
UNIVERSITATEA PETROL – GAZE DIN PLOIEȘTI
FACULTATEA LITERE ȘI ȘTIINȚE
DEPARTAMENTUL INFORMATICĂ, TEHNOLOGIA INFORMAȚIEI,
MATEMATICĂ ȘI FIZICĂ
SPECIALIZAREA INFORMATICĂ
CURSURI DE ZI

PROIECT LA DISCIPLINA BAZE DE DATE

TEMA: Mersul trenurilor

Titular disciplină:
Prof. univ. dr. Monica Vlădoiu

Student:
Andrei Alin Niță

PLOIEȘTI
2021

“Cei care visează ziua sunt conștienți de multe lucruri ce le scapă celor care visează doar noaptea.” - Edgar Allen Poe

Link bază de date: github.com/andreinitadev/mersul_trenurilor/

1. Prezentare conceptuală

Într-o lume în continuă dezvoltare, adaptarea la noile cerințe ale utilizatorului este esențială pentru o bună calitate a serviciului. Acest fapt este valabil și în domeniul transportului feroviar. Soluția pentru a rămâne adaptat este o bază de date.

Această bază de date reprezintă o aplicație de gestiune a informațiilor despre mersul trenurilor.

Utilitatea ei constă în viteza mare de prelucrare eficientă a datelor, comparativ cu organizarea clasică a dosarelor fizice pentru gestionarea datelor.

Cerințe utilizator:

- ce trenuri sunt administrate de o companie feroviară
- ce rute parcurg trenurile
- ce specificații are un tren
- cu ce stații de tren lucrează companiile feroviare
- ce case de bilete se află într-o stație de tren
- ce rute sunt disponibile la o casă de bilete

De asemenea, baza de date oferă posibilitatea de a verifica dacă o rută are schimbări între stații, dacă are facilități precum vagon de dormit/vagon bar. Mai putem verifica câți angajați are o casă bilete, câte reclamații și câte posturi disponibile pentru angajare. Putem verifica dacă într-o stație de tren există parcare, toaletă, magazin și dacă există, de ce tip sunt acestea.

2. Prezentare tehnică

Mulțimi entitate:

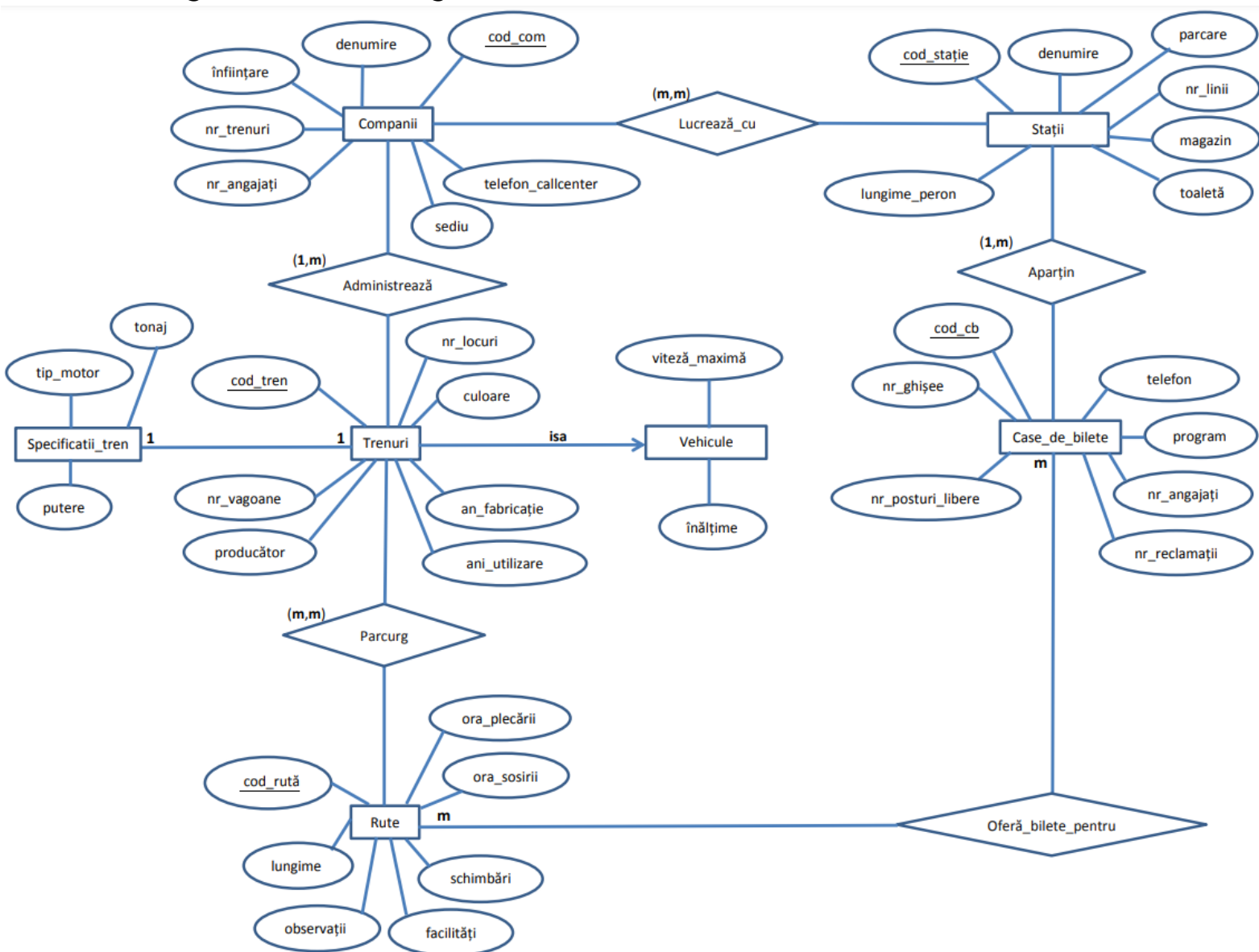
- Companii
- Stații
- Case_de_bilete
- Rute
- Trenuri
- Specificații_tren (incompletă)
- Vehicule (incompletă)

Mulțimi legătură:

- Administrează (1,m)
- Parcurg (m,m)
- Lucrează_cu (m,m)
- Aparțin (1,m)
- Oferă_bilete_pentru (m,m)

Relație 1-1 între Trenuri și Specificații_tren
Ierarhie *isa* între Trenuri și Vehicule

Diagramă Entitate-Legătură:



`cod_com` este cheie primară în *Companii* pentru că fiecare companie feroviară deține un cod unic, format din 1-3 litere, după care o putem identifica.

Exemple: R, IR, TFC

`cod_tren` este cheie primară în *Trenuri* deoarece fiecare tren are un cod unic de identificare, de obicei format din 3-5 cifre.

Exemple: 139, 2327, 15051

`cod_ruta` este cheie primară în *Rute* generată automat sau introdusă manual pentru a identifica unic o rută.

`cod_statie` este cheie primară în *Statii* generată automat sau introdusă manual pentru a identifica unic o stație.

`cod_cb` este cheie primară în *Case_de_bilete* generată automat sau introdusă manual pentru a identifica unic o casă de bilete.

Trenuri 1-1 Specificatii_tren, pentru că specificațiile extind detaliile unui tren.

Trenuri isa Vehicule, deoarece un tren este un caz particular de vehicul.

Mai **multe** trenuri *parcurs* mai **multe** rute.

Relația “*parcurs*” este reprezentată în baza de date relațională ca o relație separată.

O companie *administrează* mai **multe** trenuri.

Relația “*administrează*” este reprezentată în baza de date relațională ca o relație separată.

O companie *lucrează cu* mai **multe** stații.

Relația “*lucrează cu*” este reprezentată în baza de date relațională ca o relație separată.

Mai **multe** case de bilete *aparțin* unei stații.

Relația “*aparțin*” este reprezentată în baza de date relațională ca o relație separată.

Mai **multe** case de bilete *oferă bilete pentru* mai **multe** rute.

Relația “*oferă bilete pentru*” este reprezentată în baza de date relațională ca o relație separată.

Secvența de instrucțiuni MySQL pentru crearea bazei de date relaționale:

```
CREATE TABLE companii (  
    cod_com VARCHAR(3),  
    denumire VARCHAR(30) DEFAULT NULL,  
    infiintare VARCHAR(30) DEFAULT NULL,  
    nr_trenuri INT DEFAULT NULL,  
    nr_angajati INT DEFAULT NULL,  
    sediu VARCHAR(30) DEFAULT NULL,  
    telefon_callcenter VARCHAR(13) DEFAULT NULL,  
    PRIMARY KEY (cod_com)  
);  
CREATE TABLE statii (  
    cod_statie INT AUTO_INCREMENT,  
    denumire VARCHAR(30) DEFAULT NULL,  
    lungime_peron VARCHAR(30) DEFAULT NULL,  
    parcare VARCHAR(30) DEFAULT NULL,  
    nr_linii INT DEFAULT NULL,  
    magazin VARCHAR(30) DEFAULT NULL,  
    toaleta VARCHAR(30) DEFAULT NULL,  
    PRIMARY KEY (cod_statie)  
);  
CREATE TABLE case_de_bilete (  
    cod_cb INT AUTO_INCREMENT,  
    cod_com VARCHAR(3) NOT NULL,  
    telefon VARCHAR(13) DEFAULT NULL,  
    program VARCHAR(255) DEFAULT NULL,  
    nr_ghisee INT DEFAULT NULL,  
    nr_angajati INT DEFAULT NULL,  
    nr_reclamatii INT DEFAULT NULL,  
    nr_posturi_libere INT DEFAULT NULL,  
    PRIMARY KEY (cod_cb),  
    FOREIGN KEY (cod_com) REFERENCES companii(cod_com) ON DELETE CASCADE  
    ON UPDATE CASCADE  
);  
CREATE TABLE trenuri (  
    cod_tren INT,  
    cod_com VARCHAR(3) NOT NULL,  
    nr_locuri INT DEFAULT NULL,  
    nr_vagoane INT DEFAULT NULL,  
    producator VARCHAR(30) DEFAULT NULL,  
    culoare VARCHAR(30) DEFAULT NULL,  
    an_fabricatie INT DEFAULT NULL,  
    ani_utilizare INT DEFAULT NULL,  
    PRIMARY KEY (cod_tren),  
    FOREIGN KEY (cod_com) REFERENCES companii(cod_com) ON DELETE CASCADE  
    ON UPDATE CASCADE  
);
```

```
CREATE TABLE vehicule (  
    cod_tren INT,  
    viteza_maxima VARCHAR(30) NOT NULL,  
    inaltime VARCHAR(30) NOT NULL,  
    PRIMARY KEY (cod_tren),  
    FOREIGN KEY (cod_tren) REFERENCES trenuri(cod_tren) ON DELETE CASCADE  
    ON UPDATE CASCADE  
);  
CREATE TABLE specificatii_tren (  
    cod_tren INT,  
    tonaj VARCHAR(30) NOT NULL,  
    tip_motor VARCHAR(30) NOT NULL,  
    putere VARCHAR(30) NOT NULL,  
    PRIMARY KEY (cod_tren),  
    FOREIGN KEY (cod_tren) REFERENCES trenuri(cod_tren) ON DELETE CASCADE  
    ON UPDATE CASCADE  
);  
CREATE TABLE rute (  
    cod_ruta INT AUTO_INCREMENT,  
    statie_plecare INT NOT NULL,  
    statie_sosire INT NOT NULL,  
    ora_plecarii TIME DEFAULT NULL,  
    ora_sosirii TIME DEFAULT NULL,  
    schimbari INT DEFAULT NULL,  
    facilitati VARCHAR(255) DEFAULT NULL,  
    observatii VARCHAR(255) DEFAULT NULL,  
    lungime VARCHAR(30) DEFAULT NULL,  
    PRIMARY KEY (cod_ruta),  
    FOREIGN KEY (statie_plecare) REFERENCES statii(cod_statie) ON DELETE  
    CASCADE ON UPDATE CASCADE,  
    FOREIGN KEY (statie_sosire) REFERENCES statii(cod_statie) ON DELETE  
    CASCADE ON UPDATE CASCADE  
);  
CREATE TABLE lucreaza_cu (  
    cod_com VARCHAR(3),  
    cod_statie INT,  
    PRIMARY KEY (cod_com, cod_statie),  
    FOREIGN KEY (cod_com) REFERENCES companii(cod_com) ON DELETE CASCADE  
    ON UPDATE CASCADE,  
    FOREIGN KEY (cod_statie) REFERENCES statii(cod_statie) ON DELETE CAS  
    CADE ON UPDATE CASCADE  
);
```

```
CREATE TABLE apartin (  
    cod_statie INT NOT NULL,  
    cod_cb INT,  
    PRIMARY KEY (cod_cb),  
    FOREIGN KEY (cod_cb) REFERENCES case_de_bilete(cod_cb) ON DELETE CAS  
CADE ON UPDATE CASCADE,  
    FOREIGN KEY (cod_statie) REFERENCES statii(cod_statie) ON DELETE CAS  
CADE ON UPDATE CASCADE  
);  
CREATE TABLE ofera_bilete_pentru (  
    cod_ruta INT,  
    cod_cb INT,  
    PRIMARY KEY (cod_ruta, cod_cb),  
    FOREIGN KEY (cod_ruta) REFERENCES rute(cod_ruta) ON DELETE CASCADE O  
N UPDATE CASCADE,  
    FOREIGN KEY (cod_cb) REFERENCES case_de_bilete(cod_cb) ON DELETE CAS  
CADE ON UPDATE CASCADE  
);  
CREATE TABLE administreaza (  
    cod_com VARCHAR(3) NOT NULL,  
    cod_tren INT,  
    PRIMARY KEY (cod_tren),  
    FOREIGN KEY (cod_tren) REFERENCES trenuri(cod_tren) ON DELETE CASCAD  
E ON UPDATE CASCADE,  
    FOREIGN KEY (cod_com) REFERENCES companii(cod_com) ON DELETE CASCADE  
ON UPDATE CASCADE  
);  
CREATE TABLE parcurg (  
    cod_tren INT,  
    cod_ruta INT,  
    PRIMARY KEY (cod_tren, cod_ruta),  
    FOREIGN KEY (cod_ruta) REFERENCES rute(cod_ruta) ON DELETE CASCADE O  
N UPDATE CASCADE,  
    FOREIGN KEY (cod_tren) REFERENCES trenuri(cod_tren) ON DELETE CASCAD  
E ON UPDATE CASCADE  
);
```

Operații algebră relațională

Reuniune

Administreaza U Lucreaza_cu

```
SELECT *  
FROM administreaza  
UNION  
SELECT *  
FROM lucreaza_cu;
```

cod_com	cod_tren
----------------	-----------------

IR	15055
----	-------

IR	15056
----	-------

IR	15060
----	-------

TFC	2327
-----	------

TFC	2329
-----	------

CFR	1
-----	---

TFC	18
-----	----

Diferență

Rute - Parcurg

```
SELECT cod_ruta
FROM rute
LEFT JOIN parcurg USING (cod_ruta)
WHERE parcurg.cod_ruta IS NULL;
```

cod_ruta

1
4
5
3
19
11
18
8
20
12
17
15
16
2
7
6

Selecție

σ cod_com = "IR" and nr_vagoane > 5 (Trenuri)

```
SELECT *
FROM trenuri
WHERE cod_com = "IR"
AND nr_vagoane > 5;
```

cod_tren	cod_com	nr_locuri	nr_vagoane	producator	culoare	an_fabricatie	ani_utilizare
15053	IR	282	6	Alstom	NULL	1990	13
15054	IR	282	6	Alstom	NULL	1990	11
15055	IR	282	6	Alstom	NULL	1990	20

Proiecție

 Π cod com, cod tren, nr vagoane (Trenuri)

```
SELECT cod_com,  
       cod_tren,  
       nr_vagoane  
FROM trenuri;
```

cod_com	cod_tren	nr_vagoane
H	139	4
TFC	2327	3
TFC	2328	5
TFC	2329	5
TFC	2330	6
IRN	15051	6
IRN	15052	6
IR	15053	6
IR	15054	6
IR	15055	6
IR	15056	5
IR	15057	5
IR	15058	5
IR	15059	5
IR	15060	5
R	15202	3
R	15203	3
R	15204	3
R	15205	3
R	15206	3

Produs cartezian

Administreaza X Lucreaza_cu

```
SELECT *  
FROM administreaza  
CROSS JOIN lucreaza_cu;
```

cod_com	cod_tren	cod_com	cod_statie
IR	15055	CFR	1
IR	15055	TFC	18
IR	15056	CFR	1
IR	15056	TFC	18
IR	15060	CFR	1
IR	15060	TFC	18
TFC	2327	CFR	1
TFC	2327	TFC	18
TFC	2329	CFR	1
TFC	2329	TFC	18

3. Direcții de dezvoltare

- Adăugare mulțime entitate Întârzieri
- Creare UI pentru cumpărare bilete online