

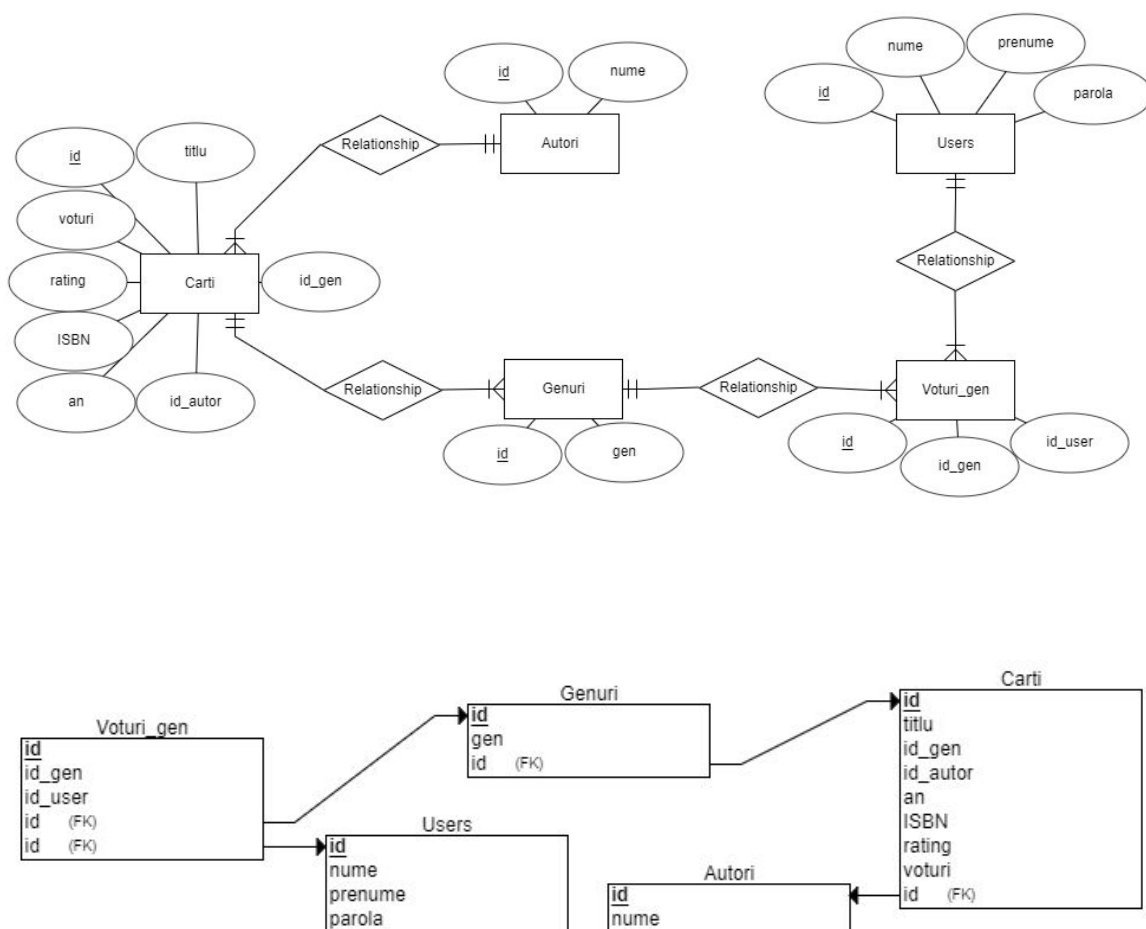
# BookMate

## Despre aplicație

O platformă unde utilizatorii pot căuta, descoperi și descărca cărți. Fiecare carte va avea un autor și un gen principal în care se încadrează, iar fiecare gen va avea un anumit număr de voturi primite de la diverși utilizatori.

Utilizatorii pot oferi un rating cărților și pot primi recomandări pe baza unui algoritm care analizează preferințele și istoricul utilizatorului, cât și utilizatori cu gusturi similare.

## EntityRelationship Schema



Diagramă realizată utilizând <https://erdplus.com/>.

```

CREATE TABLE carli (
    id INT NOT NULL PRIMARY KEY,
    titu VARCHAR(128) NOT NULL,
    id_gen INT NOT NULL,
    id_autor INT NOT NULL,
    an INT NOT NULL,
    isbn INT NOT NULL,
    rating FLOAT NOT NULL,
    voturi INT NOT NULL
)

CREATE TABLE genuri (
    id INT NOT NULL PRIMARY KEY,
    gen VARCHAR(128) NOT NULL
)

CREATE TABLE voturi_gen (
    id INT NOT NULL PRIMARY KEY,
    id_gen INT NOT NULL,
    id_user INT NOT NULL
)

CREATE TABLE autori (
    id INT NOT NULL PRIMARY KEY,
    nume VARCHAR(128) NOT NULL
)

CREATE TABLE users (
    id INT NOT NULL PRIMARY KEY,
    nume VARCHAR(128) NOT NULL,
    prenume VARCHAR(128) NOT NULL,
    parola VARCHAR(128) NOT NULL
)

--cod inspirat din scriptul initial de creare a tabelelor pentru studii
SET SERVEROUTPUT ON;
DECLARE
    TYPE varr IS VARRAY(1000) OF varchar(255);

    lista_nume_varr :=
varr('Ababei','Acasandrei','Adascaltei','Almanasi','Agafiei','Aghape','Akiane','Alexandrescu','Alexandru','Alesae','Alexei','Amarghiotei','Ambrosio','Andonesi','Andreai','Andrian','Andrio','Andronic','Andros','Anghelina','Anila','Antochi','Antoniei','Apetrei','Apostoi','Arhip','Arhive','Atenei','Avraie','Asafiei','Asafiei','Augurendu','Avadanei','Avrami','Babele','Babi','Babici','Babin','Babici','Babu','Babescu','Bacsi','Baigan','Bejan','Bejancu','Belcoveanu','Belciuganu','Benchea','Blari','Brinaru','Bivri','Bisu','Boici','Bolhar','Bolintean','Botari','Botocanu','Botocatu','Bradeti','Brașoiu','Butacu','Buzoi','Călcănuț','Calinescu','Capasneanu','Caraiman','Carbunari','Carpi','Catarii','Catinari','Catanoli','Cazacu','Cazamir','Cebeanu','Celarian','Cemescu','Chelarie','Chelmici','Chelmus','Chibot','Chicoș','Chilabot','Chilei','Chiriaci','Chirilă','Chisot','Chitici','Chimilevski','Compoșoi','Coșbanu','Ciobanaru','Ciochiu','Ciofă','Ciomei','Ciolei','Ciucanu','Clainici','Clim','Cobzac','Coca','Copcalari','Copcaru','Conducare','Coroi','Corduneanu','Corfu','Coreanu','Cordesco','Cosenu','Cosnita','Costari','Covataru','Cozma','Cozmiciu','Craciunas','Crainiceanu','Creanga','Crețu','Cristea','Croacescu','Cumpana','Curca','Cușmaliuc','Damian','Damoș','Danielu','Danilei','Daniela','Darie','Dascalescu','Dascalu','Dașconu','Dimă','Dimache','Diru','Dobos','Doehlie','Doehliu','Dogaru','Dogaru','Domanița','Domeneau','Dragan','Dragomir','Dragomir','Dragomirescu','Dușacu','Dudaș','Dumee','Edi','Eduard','Eusebiu','Fedelei','Fereastrănu','Filibu','Filimon','Filip','Florinel','Folveler','Frumosu','Fruntza','Gălbănuș','Gavrilă','Gavriliu','Gavrilovic','Gherase','Gherca','Ghergu','Ghemari','Ghibridie','Giosanu','Gitlan','Giurgiu','Goldanu','Golden','Grana','Grigori','Grigori','Gross','Groșanu','Gura','Haba','Hamblu','Harion','Narga','Herde','Hencovski','Hociung','Hodoreanu','Hosticu','Huma','Hudani','Hutomi','Iacob','Iacobuta','Ianco','Ichimi','Iliescu','Ilie','Isauresti','Itensei','Jones','Jonita','Ionache','Ionache-Iroiu','Jordan','Josai','Jou','Jimini','Jvascu','Jac','U','Jlatareșu','Ulta','Joldescu','Jaruvic','Larion','Latet','Lati','Lucia','Leleu','Leon','Leonte','Leucutu','Levișteanu','Luca','Lucaci','Lucasi','Luncasu','Lungeanu','Lungi','Lupsacu','Lupu','Macariu','Maconeschi','Malfei','Magariu','Margaleagu','Maralache','Manole','Marco','Marinoi','Martinas','Marton','Mataca','Matcovici','Malei','Maltes','Matrasu','Maximu','Mezurescu','Mazilu','Mazu','Melnicu-puiu','Micu','Mihaela','Mihai','Mihală','Mihalcescu','Mihalcea','Mihalcea','Mihalcescu','Mihalci','Minea','Miner','Miruși','Miruș','Mitani','Moșu','Monni-sbyaneu','Morarescu','Morosanu','Moscu','Motrescu','Motro','Munteanu','Murarasu','Musca','Musculescu','Nastacu','Nechitu','Neghină','Neghini','Negrușen','Negrutu','Nemtoș','Netedu','Nica','Nicu','Oana','Olariu','Olariu','Olaru','Oru','Oprișcu','Opres','Ostafe','Otrocot','Palhiovici','Pantiru','Panfircu','Paparus','Pascani','Patalski','Patrasi','Perșani','Petru','Pelcu','Pia','Pietile','Pișui','Platon','Pogariu','Podaru','Poenariu','Pojar','Popa','Popescu','Popovici','Popubala','Postolache','Predoiu','Prise','Prasa','Procop','Prodan','Prai','Rautu','Rechenen','Razvan','Reut','Roscani','Rita','Robu','Roman','Romanescu','Rosca','Rusu','Samson','Sandu','Sarduteache','Save','Savescu','Schindler','Scorianu','Scuti','Slarghiu','Silici','Simingradu','Simioni','Simionescu','Simonești','Simon','Ștancu','Stegheș','Solici','Spanhel','Spidonu','Stari','Staravache','Stefan','Stefanita','Stingaci','Stuflicu','Șoalan','Șoica','Șolovu','Șolniceanu','Șolnicu','Strainu','Știmbu','Suhanu','Tabasca','Talia','Tanasa','Tecliu','Teodororescu','Teșu','Thea','Timofte','Tincu','Tipescu','Toader','Tofan','Tomă','Tonci','Tondici','Tudosae','Tudosar','Tuduș','Tulu','Turcu','Ulinici','Urghianu','Urquiansu','Ursache','Uscaci','Uscaci','Ușuș','Ușuș','Varlan','Văntur','Vaslache','Vasilu','Ventaniciu','Vicol','Vidru','Vinatoru','Viad','Voades','Vrabie','Vulpescu','Zamostanu','Zazuleac');

    lista_prenume_fete_varr :=
varr('Adina','Alexandra','Alina','Ana','Anca','Anda','Andrea','Andreea','Antonio','Bianca','Camelia','Claudia','Codrina','Cristina','Daniela','Daria','Delia','Denisa','Diana','Ecaterina','Elena','Eleonora','Elisa','Emi','Emanuel','Emma','Gabriela','Georgiana','Ileana','Iona','Ioana','Ioanda','Irina','Julia','Juliana','Laura','Lara','Loredana','Madalina','Malina','Manuela','Maria','Mihaela','Mirela','Monica','Oana','Paula','Petruta','Raluca','Sabina','Sanzana','Simina','Simona','Stefana','Stefania','Tamara','Teodora','Theodora','Vasilica','Xena');

    lista_prenume_baili_varr :=
varr('Adrian','Alex','Alexandru','Aliu','Andreas','Andre','Aurelian','Beniamin','Bogdan','Camil','Catalin','Cezar','Ciprian','Clausu','Codrin','Constantin','Corneliu','Cosmin','Costel','Cristian','Damian','Dan','Daniel','Danuș','Darius','Denise','Dimitrie','Dorian','Dorin','Dragos','Dumitru','Eduard','Elvie','Emil','Ervin','Eugen','Eusebiu','Fabian','Filip','Florian','Florin','Gabriel','George','Gheorghe','Giani','Gulio','Iaroslav','Ilie','Ioan','Ion','Iosef','Iosif','Irinei','Iulian','Justin','Laurențiu','Liviu','Lucian','Marian','Markus','Matei','Mihai','Mihail','Nicolae','Nicu','Nicosu','Octavian','Ovidiu','Paul','Petru','Radu','Rares','Razvan','Richard','Robert','Roland','Romanescu','Sabin','Samuel','Sebastian','Sergiu','Shivu','Stelian','Teodor','Teofil','Theodor','Tudor','Vadin','Valentin','Valeriu','Vassile','Victor','Viad','Vladimir','Vladuți');

    lista_știri_varr := varr ('Măltreșu'):O mie noua sute optzeci și patru,Batrându si marea;Portretul lui Dorian Gray,'Don Quixote de la Mancha','Fernando almaguer','Lola','Micu priu','Puterea prezentului','Inteligenta emotională','Pe cîmble disperate','Topical capricornular';Străinu;Procesul';'Achimsitul';Aninții de pribegie;'Harry Potter');

    lista_genuri_varr := varr ('autobiografie','romantă','nouvele','ticdine','satira','dezvoltare','filozofie','fantazie');

    lista_autori_varr := varr ('Mircea Eliade','George Orwell','Ernest Hemingway','Miguel de Cervantes','Vladimir Nabokov','Antoine de Saint-Exupery','Eckhart Tolle','Daniel Goleman','Emil Cioran','Henry Miller','Albert Camus','Paulo Coelho','Neagu Djvara','J.K. Rowling');

    v_nume VARCHAR(255);
    v_prenume VARCHAR(255);
    v_prenume1 VARCHAR(255);
    v_prenume2 VARCHAR(255);
    v_id VARCHAR(255);
    v_id_gen INT;
    v_id_autor INT;
    v_an NUMBER(4, 0);
    v_isbn NUMBER(13, 0);
    v_rating FLOAT;
    v_voturi INT;
    v_id_user INT;

BEGIN
    DBMS_OUTPUT.PUT_LINE(insertare a 100.000 useri);
    FOR v_i IN 1..100000 LOOP
        v_nume := lista_nume(TRUNC(DBMS_RANDOM.VALUE(0,lista_nume.count))+1);
        IF (DBMS_RANDOM.VALUE(0,100)/50) THEN
            v_prenume1 := lista_prenume_fete(TRUNC(DBMS_RANDOM.VALUE(0,lista_prenume_fete.count))+1);
            LOOP
                v_prenume2 := lista_prenume_fete(TRUNC(DBMS_RANDOM.VALUE(0,lista_prenume_fete.count))+1);
            exit when v_prenume1<>v_prenume2;
            END LOOP;
            ELSE
                v_prenume1 := lista_prenume_baili(TRUNC(DBMS_RANDOM.VALUE(0,lista_prenume_baili.count))+1);
                LOOP
                    v_prenume2 := lista_prenume_baili(TRUNC(DBMS_RANDOM.VALUE(0,lista_prenume_baili.count))+1);
                exit when v_prenume1<>v_prenume2;
                END LOOP;
            END IF;
            IF (DBMS_RANDOM.VALUE(0,100)/60) THEN

```

```

IF LENGTH(v_prenume1 || ' ' || v_prenume2) <= 20 THEN
    v_prenume := v_prenume1 || ' ' || v_prenume2;
END IF;
else
    v_prenume:=v_prenume1;
END IF;
INSERT INTO users VALUES (v_i, v_nume, v_prenume, 'hash');
END LOOP;

DBMS_OUTPUT.PUT_LINE('Inserarea a 100.000 carti:');
FOR v_i IN 1..100000 LOOP
    v_titlu := lista_titlu(TRUNC(DBMS_RANDOM.VALUE(0,lista_titlu.count))+1);
    v_id_gen := TRUNC(DBMS_RANDOM.VALUE(0,lista_genuri.count))+1;
    v_id_autor := TRUNC(DBMS_RANDOM.VALUE(0,lista_autori.count))+1;
    v_an := TRUNC(DBMS_RANDOM.VALUE(1200, 2019));
    v_isbn := TRUNC(DBMS_RANDOM.VALUE(1000000000000, 999999999999));
    v_rating := DBMS_RANDOM.VALUE(1, 5);
    v_voturi := TRUNC(DBMS_RANDOM.VALUE(500, 10000));
    INSERT INTO carti VALUES (v_i, v_titlu, v_id_gen, v_id_autor, v_an, v_isbn, v_rating, v_voturi);
END LOOP;

DBMS_OUTPUT.PUT_LINE('Simularea a 1.000.000 voturi de callback:');
FOR v_i IN 1..1000000 LOOP
    v_titlu := lista_titlu(TRUNC(DBMS_RANDOM.VALUE(0,lista_titlu.count))+1);
    v_id_gen := TRUNC(DBMS_RANDOM.VALUE(0,lista_genuri.count))+1;
    v_id_user := TRUNC(DBMS_RANDOM.VALUE(1,100000))+1;
    INSERT INTO voturi_gen VALUES (v_i, v_id_gen, v_id_user);
END LOOP;

DBMS_OUTPUT.PUT_LINE('Inserarea genurilor:');
FOR v_i IN 1..8 LOOP
    INSERT INTO genuri VALUES (v_i, lista_genuri(v_i));
END LOOP;

DBMS_OUTPUT.PUT_LINE('Inserarea autorilor:');
FOR v_i IN 1..14 LOOP
    INSERT INTO autori VALUES (v_i, lista_autori(v_i));
END LOOP;

END;

```

## Descrierea structurilor de date

- tabela ‘Cărți’ - conține informațiile asociate tuturor cărților
  - id - cheie de indexare unică în tabel
  - titlu - titlul cărții sub formă de string - varchar2
  - id\_autor - cheie străină către tabela ‘Autori’
  - id\_gen - cheie străină către tabela ‘Genuri’
  - an - anul apariției - int
  - ISBN - șir unic de referință - poate fi folosit ca și hash pentru numele fișierului - va identifica în mod unic pdf-ul cărții în sistemul de fișiere - int
  - rating - media voturilor acordate cărții respective - float
  - voturi - numărul de voturi acordate, pentru a putea actualiza corect matematic rating-ul - int
- tabela ‘Autori’
  - id - cheie de indexare unică în tabel (referențiată în tabela ‘Cărți’)
  - nume - numele autorului - varchar2
- tabela ‘Genuri’
  - id - cheie de indexare unică în tabel (referențiată în tabela ‘Cărți’ și în tabela ‘Voturi\_gen’)
  - gen - genul abordat în conținutul cărții - varchar2
- tabela ‘Users’
  - id - cheie de indexare unică în tabel (referențiată în tabela ‘Voturi\_gen’)
  - nume - numele utilizatorului - varchar2
  - prenume - prenumele utilizatorului - varchar2
  - parola - în acest câmp va fi stocat doar un hash al parolei, care va împiedica pe oricine (din sistem sau din afara lui) să poată decripta parola utilizatorului - verificarea făcându-se doar comparând output-ul funcției de hash cu intrarea din această tabelă, atunci când utilizatorul se autentifică.

- tabela 'Voturi\_gen'
  - id - cheie de indexare unică în tabel
  - id\_gen - cheie străină către tabela 'Genuri'
  - id\_user - cheie străină către tabela 'Users'

### **Descrierea funcțiilor si procedurilor**

- GetBrothers (user\_id u) - va interoga și corela datele din tabele pentru a afla care utilizatori au preferințele cele mai asemănătoare cu userul dat
- GetTopBooks (id\_gen g) - va întoarce lista cu cele mai bune cărți în ordinea rating-ului pentru un anumit gen
- VisitBook (id\_carte c, user\_id u) - funcție de callback care se va apela mereu când un utilizator interacționează cu o anumită carte, pentru a contoriza interesul său față de attributele ei
- GetBookLink (id\_carte c) - va calcula un hash pe ISBN-ul cărții pentru a întoarce calea de pe disk către pdf-ul asociat
- CheckPassword (user\_id u, varchar2 password) - va întoarce true sau false dacă parola primită ca parametru (în plaintext) corespunde hash-ului parolei utilizatorului dat ca parametru
- GetRecommendations (user\_id u) - va întoarce o listă cu noi cărți (pe care utilizatorul nu le-a mai întâlnit până acum), folosind și prima funcție din acest paragraf - GetBrothers. Pentru fiecare carte se poate calcula un coeficient de interes - bazat pe numărul de voturi, valoarea rating-ului, valoarea rating-ului raportat la numărul de voturi, popularitatea genului ei printre cei mai activi utilizatori, numărul total de vizualizări raportat la anul apariției, experiența autorului ei, rating-ul altor cărți scrise de același autor, numărul de accesări într-un interval setat de timp, numărul de apariții în istoricul ultimelor 100 cărți vizitate global.