Kolegij:

Baze podataka I.

Projekt:

Sustav za upravljanje knjižnicom

Autor:

Maja Kovačević

JMBAG 0009074625

# Uvod

## Cilj mog modula

U sklopu timskog projekta "Sustav za upravljanje knjižnicom", moj zadatak je bio osmisliti i implementirati modul koji pokriva evidenciju događaja u knjižnici, povezanost događaja s određenim knjigama, te vođenje inventara knjiga. Kroz ove tri povezane tablice prikazan je dodatni sadržaj i funkcionalnost koju jedna suvremena knjižnica može nuditi korisnicima.

Modul omogućuje:

* Centralizirano čuvanje podataka o događajima knjižnice.
* Praćenje izlaganja knjiga na događajima.
* Evidentiranje fizičkih primjeraka knjiga (inventar) i statistički uvidi u stanje inventara.

## Tehnologije

* MySQL – relacijski sustav za upravljanje bazom podataka.
* MySQL Workbench – modeliranje sheme (ER-dijagram) i izvršavanje skripti.
* Visual Studio Code + proširenje SQL Tools – pisanje koda.
* Git + GitHub – timska kontrola inačica i kontinuirana integracija.
* Microsoft Word – pisanje dokumentacije.

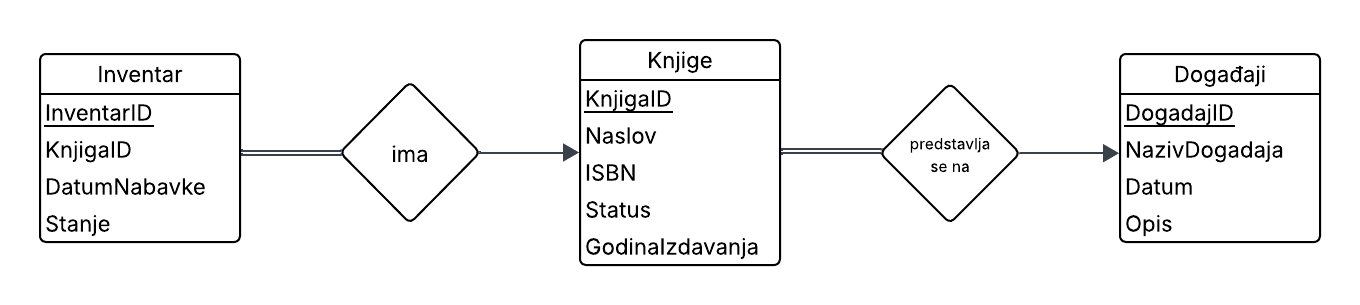
## Struktura dokumenta

U nastavku prvo slijedi konceptualni ER-dijagram te logička shema baze podataka.

Zatim detaljan opis tablica i atributa, uključujući tablice *Događaji, KnjigaDogađaji* i *Inventar*.

Središnji dio sadržava pregled izrađenih SQL upita i pogleda s objašnjenjima istih te na kraju zaključak.

# ER dijagram



**Relacijski model:**

Knjige (KnjigaID, Naslov, ISBN, Status, GodinaIzdavanja)

Inventar (InventarID, KnjigaID, DatumNabavke, Stanje)

Dogadaji (DogadajID, NazivDogadaja, Datum, Opis)

KnjigaDogadaji (KnjigaID, DogadajID)

# Detaljan opis tablica, atributa i domena s komentarima

**Tablica Dogadaji**

|  |  |  |
| --- | --- | --- |
| Atribut | Domena/Ograničenja | Komentar |
| DogadajID | INT PK, AUTO\_INCREMENT, NOT NULL | ID događaja |
| NazivDogadaja | VARCHAR(100) NOT NULL | Naziv događaja |
| Datum | DATE NOT NULL | Datum održavanja događaja |
| Opis | TEXT | Kratki opis događaja i dodatne informacije |

**Tablica KnjigaDogadaji**

|  |  |  |
| --- | --- | --- |
| Atribut | Domena/Ograničenja | Komentar |
| KnjigaID | INT FK NOT NULL | Referenca na knjigu koja se izlaže na događaju/je dio događaja |
| DogadajID | INT FK NOT NULL | Referenca na događaj na kojem je knjiga/knjige predstavljena |

**Tablica Inventar**

|  |  |  |
| --- | --- | --- |
| Atribut | Domena/Ograničenja | Komentar |
| InventarID | INT PK, AUTO\_INCREMENT, NOT NULL | ID inventarskog zapisa |
| KnjigaID | INT FK NOT NULL | Referenca na knjigu |
| DatumNabavke | DATE NOT NULL | Datum nabavke primjerka |
| Stanje | VARCHAR(50) | Stanje primjerka |

# Pogledi

Knjige dodane u posljednjih godinu dana

Prikazuje sve stavke iz knjižničnog inventara koje su nabavljene u posljednjih 12 mjeseci. Koristan je za praćenje novih unosa u fond knjižnice, godišnje izvještaje o nabavi, identifikaciju novih knjiga i opreme za promotivne aktivnosti.

CREATE VIEW dodano\_u\_zadnjih\_god\_dana AS

SELECT \*

FROM inventar

WHERE DatumNabavke >= CURDATE() - INTERVAL 1 YEAR;

SELECT \*

FROM dodano\_u\_zadnjih\_god\_dana

WHERE Stanje = 'Oštećeno';

**Detalji:**

* CREATE VIEW dodano\_u\_zadnjih\_god\_dana AS – kreira pogled naziva dodano\_u\_zadnjih\_god\_dana
* SELECT \* FROM Inventar – odabire sve atribute iz tablice Inventar, uključujući InventarID, KnjigaID, DatumNabavke i Stanje
* WHERE DatumNabavke >= CURDATE() - INTERVAL 1 YEAR – filtrira stavke koje su nabavljene unutar zadnjih godinu dana od današnjeg datuma (CURDATE())
* WHERE Stanje = 'Oštećeno' – filtrira samo oštećene stavke među novim primjercima

Prikaz stanja knjiga, neovisno o naslovima

Pogled knjige\_stanje prikazuje koliko se fizičkih primjeraka knjiga nalazi u svakom stanju, poput *Novo, Oštećeno, Izgubljeno,* itd. Ovaj pogled je koristan za brzu provjeru stanja fonda knjižnice, prepoznavanje postoji li potreba za obnovom inventara te izvještavanje o stanju imovine knjižnice.

CREATE VIEW knjige\_stanje AS

SELECT

i.Stanje,

COUNT(\*) AS BrojKnjiga

FROM Inventar i

GROUP BY i.Stanje;

**Detalji:**

* CREATE VIEW knjige\_stanje AS - kreira pogled s nazivom knjige\_stanje
* SELECT i.Stanje, COUNT(\*) AS BrojKnjiga:
  + Stanje – opis stanja primjerka (Novo, Oštećeno);
  + BrojKnjiga – broj primjeraka u tom stanju
* FROM Inventar i - izvlači podatak iz tablice Inventar
* GROUP BY i.Stanje - grupira rezultate po stanju kako bi dobili broj za svaku kategoriju

## Naslovi knjiga grupirani prema stanju

Detaljan pregled naslova svih knjiga prema njihovom stanju i datumu nabavke. Daje mogućnost identifikacije stanja po naslovima, korisno za obnovu fonda.

CREATE VIEW naslovi\_po\_stanju AS

SELECT

k.Naslov,

i.Stanje,

i.DatumNabavke

FROM Inventar i

JOIN Knjige k ON i.KnjigaID = k.KnjigaID

ORDER BY i.Stanje, i.DatumNabavke;

SELECT \* FROM naslovi\_po\_stanju

WHERE Stanje = 'Oštećeno'

ORDER BY DatumNabavke DESC;

**Detalji:**

* CREATE VIEW naslovi\_po\_stanju AS – kreira novi pogled naslovi\_po\_stanju
* SELECT k.Naslov, i.Stanje, i.DatumNabavke:
  + Naslov - naziv knjige; Stanje - fizičko stanje knjige (Novo, Oštećeno, Izgubljeno);
  + DatumNabavke - kada je primjerak kupljen/nabavljen
* FROM Inventar JOIN Knjige – spaja tablice Inventar (fizičke jedinice) i Knjige (naslovi)
* ORDER BY i.Stanje, i.DatumNabavke – sortiranje po stanju i datumu

Prikazuje naslova knjiga i pripadajućih događaja

Prikazuje koji su naslovi uključeni u koje događaje. Korisno za pregled književnih večeri, radionica itd.

CREATE VIEW dogadaji\_s\_naslovima AS

SELECT

d.NazivDogadaja,

d.Datum,

k.Naslov AS NaslovKnjige

FROM KnjigaDogadaji kd

JOIN Dogadaji d ON kd.DogadajID = d.DogadajID

JOIN Knjige k ON kd.KnjigaID = k.KnjigaID

ORDER BY d.Datum;

SELECT \* FROM dogadaji\_s\_naslovima

ORDER BY Datum DESC;

SELECT \* FROM dogadaji\_s\_naslovima

WHERE NazivDogadaja = 'Noć knjige';

**Detalji:**

* CREATE VIEW dogadaji\_s\_naslovima AS – kreira pogled naziva dogadaji\_s\_naslovima
* SELECT d.NazivDogadaja, d.Datum, k.Naslov AS NaslovKnjige:
  + NazivDogadaja – npr. “Noć knjige”
  + Datum – kada se događaj odvija
  + NaslovKnjige: knjiga koja se koristi
* FROM KnjigaDogadaji JOIN Dogadaji JOIN Knjige - povezuje veznu tablicu s događajima i naslovima.
* ORDER BY d.Datum – rezultati sortirani kronološki
* ORDER BY Datum DESC – sortira rezultate tako da su najnoviji događaji prikazani prvi
* WHERE NazivDogadaja = 'Noć knjige' – filtrira rezultate tako da prikazuje samo naslove knjiga koji su bili dio događaja "Noć knjige"

## Prikazuje broj knjiga uključenih u događaj

Prikazuje koliko je knjiga povezano s pojedinim događajem u knjižnici. Korisno pri procjeni veličine i opsega događaja.

CREATE VIEW broj\_knjiga\_po\_dogadaju AS

SELECT

d.NazivDogadaja,

d.Datum,

COUNT(kd.KnjigaID) AS BrojKnjiga

FROM Dogadaji d

LEFT JOIN KnjigaDogadaji kd ON d.DogadajID = kd.DogadajID

GROUP BY d.DogadajID, d.NazivDogadaja, d.Datum;

SELECT \* FROM broj\_knjiga\_po\_dogadaju

ORDER BY Datum DESC, BrojKnjiga DESC

LIMIT 10;

SELECT \* FROM broj\_knjiga\_po\_dogadaju

WHERE BrojKnjiga > 1;

**Detalji:**

* CREATE VIEW broj\_knjiga\_po\_dogadaju AS – kreira pogled
* SELECT d.NazivDogadaja, d.Datum, COUNT(kd.KnjigaID) AS BrojKnjiga:
  + NazivDogadaja i Datum – osnovne info o događaju
  + BrojKnjiga – ukupan broj povezanih knjiga
* LEFT JOIN – za prikaz događaja koji još nemaju dodijeljene knjige
* Koristi se kako bi se prikazali i događaji koji nemaju još dodijeljene knjige.
* GROUP BY – za grupiranje po jedinstvenim događajima
* SELECT \* FROM broj\_knjiga\_po\_dogadaju – dohvaća sve retke iz pogleda broj\_knjiga\_po\_dogadaju
* ORDER BY Datum DESC, BrojKnjiga DESC – rezultati su sortirani po datumu silazno/descending, tako da su najnoviji događaji prvi
  + Ako više događaja ima isti datum, onda se dodatno sortiraju po broju knjiga (više knjiga → više prioriteta)
* LIMIT 10 – ograničava prikaz na samo 10 najnovijih i najvećih događaja
* WHERE BrojKnjiga > 1 – prikazuje samo one događaje u kojima je uključeno više od jedne knjige

# Zaključak

Kroz rad na ovom modulu usvojila sam konkretna znanja o kreiranju i povezivanju tablica u MySQL-u, korištenju stranih ključeva, i pisanju kompleksnijih SQL upita. Dodatno, rad s CSV datotekama i testiranjem putem komandne linije omogućio mi je dublje razumijevanje rada s bazom izvan grafičkog sučelja. Moj doprinos u dokumentaciji, strukturiranju baze i izradi pogleda direktno je podržao cjelokupnu funkcionalnost sustava.