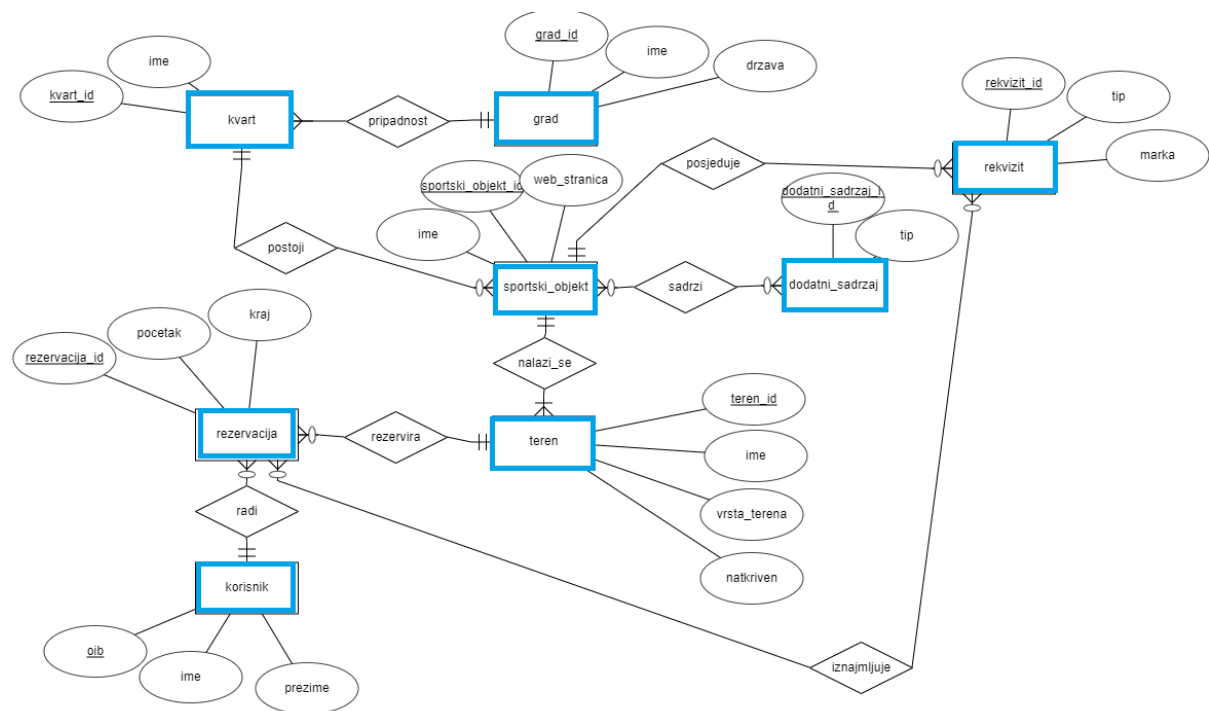


## ER dijagram



### Entiteti

korisnik – osoba koja želi rezervirati teren

rezervacija – informacije o rezervaciji

teren – teren koji se želi rezervirati

sportski\_objekt – mjesto u kojem se nalaze različiti sportski tereni

dodatni\_sadržaj – informacije o dodatnim sadržajima koje sportski objekt može imati (kafić, tuševi, teretana i sl.)

rekvizit – sportski rekviziti (nogometne lopte, košarkaške lopte, reketi, mreže za stolni tenis i sl.) koji se mogu iznajmiti u sportskom objektu

kvart – kvart u kojem se sportski objekt nalazi

grad – grad u kojem se sportski objekt nalazi

## Relacije

**radi** – određeni korisnik može napraviti nula ili više rezervacija, određena rezervacija može biti napravljena od jednog i samo jednog korisnika

**rezervira** – određenom rezervacijom se može rezervirati jedan i samo jedan teren, nad određenim terenom se može napraviti nula ili više rezervacija

**iznajmljuje** – određenom rezervacijom se iznajmljuje nula ili više rekvizita, određeni rekvizit može biti iznajmljen u nula ili više rezervacija

**nalazi\_se** – u određenom sportskom objektu se može nalaziti jedan ili više terena, određeni teren se nalazi u jednom i samo jednom sportskom objektu

**sadrzi** – određeni sportski objekt može sadržavati nula ili više dodatnih sadržaja, dodatni sadržaj može biti sadržan u nula ili više sportskih objekata

**posjeduje** – određeni sportski objekt posjeduje nula ili više rekvizita, određeni rekvizit može biti posjedovan od jednog i samo jednog sportskog objekta

**ima** – određeni grad može imati jedan ili više sportskih objekata, određeni sportski objekt može pripadati jednom i samo jednom gradu

**pripadnost** – određeni kvart pripada jednom i samo jednom gradu, gradu može pripadati više kvartova

**postoji** – u jednom kvartu postoji nula ili više sportskih objekata, jedan sportski objekt postoji u samo jednom kvartu

## Kod za stvaranje tablica

```
CREATE TABLE system.grad
(
    grad_id INT NOT NULL,
    ime VARCHAR(60) NOT NULL,
    drzava VARCHAR(60) NOT NULL,
    PRIMARY KEY (grad_id)
);

CREATE TABLE system.korisnik
(
    oib VARCHAR(60) NOT NULL,
    ime VARCHAR(60) NOT NULL,
    prezime VARCHAR(60) NOT NULL,
    PRIMARY KEY (oib)
);

CREATE TABLE system.dodatni_sadrzaj
(
    dodatni_sadrzaj_id INT NOT NULL,
    tip VARCHAR(60) NOT NULL,
```

```

    PRIMARY KEY (dodatni_sadrzaj_id)
);

CREATE TABLE system.kvart
(
    kvart_id INT NOT NULL,
    ime VARCHAR(60) NOT NULL,
    grad_id INT NOT NULL,
    PRIMARY KEY (kvart_id),
    FOREIGN KEY (grad_id) REFERENCES system.grad(grad_id)
);

CREATE TABLE system.sportski_objekt
(
    sportski_objekt_id INT NOT NULL,
    ime VARCHAR(60) NOT NULL,
    web_stranica VARCHAR(60) NOT NULL,
    kvart_id INT NOT NULL,
    PRIMARY KEY (sportski_objekt_id),
    FOREIGN KEY (kvart_id) REFERENCES system.kvart(kvart_id)
);

CREATE TABLE system.teren
(
    teren_id INT NOT NULL,
    vrsta_terena VARCHAR(60) NOT NULL,
    natkriven INT NOT NULL,
    ime VARCHAR(60) NOT NULL,
    sportski_objekt_id INT NOT NULL,
    PRIMARY KEY (teren_id),
    FOREIGN KEY (sportski_objekt_id) REFERENCES
system.sportski_objekt(sportski_objekt_id)
);

CREATE TABLE system.rezervacija
(
    pocetak DATE NOT NULL,
    kraj DATE NOT NULL,
    rezervacija_id INT NOT NULL,
    teren_id INT NOT NULL,
    oib VARCHAR(11) NOT NULL,
    PRIMARY KEY (rezervacija_id),
    FOREIGN KEY (teren_id) REFERENCES system.teren(teren_id),
    FOREIGN KEY (oib) REFERENCES system.korisnik(oib)
);

CREATE TABLE system.rekvizit
(
    rekvizit_id INT NOT NULL,
    tip VARCHAR(60) NOT NULL,
    marka VARCHAR(60) NOT NULL,
    sportski_objekt_id INT NOT NULL,
    PRIMARY KEY (rekvizit_id),
    FOREIGN KEY (sportski_objekt_id) REFERENCES
system.sportski_objekt(sportski_objekt_id)
);

```

```

CREATE TABLE system.rezervacija_rekvizit
(
    rezervacija_id INT NOT NULL,
    rekvizit_id INT NOT NULL,
    PRIMARY KEY (rezervacija_id, rekvizit_id),
    FOREIGN KEY (rezervacija_id) REFERENCES
system.rezervacija(rezervacija_id),
    FOREIGN KEY (rekvizit_id) REFERENCES system.rekvizit(rekvizit_id)
);

CREATE TABLE system.objekt_sadrzaj
(
    sportski_objekt_id INT NOT NULL,
    dodatni_sadrzaj_id INT NOT NULL,
    PRIMARY KEY (sportski_objekt_id, dodatni_sadrzaj_id),
    FOREIGN KEY (sportski_objekt_id) REFERENCES
system.sportski_objekt(sportski_objekt_id),
    FOREIGN KEY (dodatni_sadrzaj_id) REFERENCES
system.dodatni_sadrzaj(dodatni_sadrzaj_id)
);

```

## Funkcije i triggeri

```

CREATE OR REPLACE TRIGGER insertTrigger
BEFORE INSERT ON system.rezervacija
FOR EACH ROW
DECLARE
    threeDaysException EXCEPTION;
    someNum INTEGER;
BEGIN
    SELECT COUNT(*) INTO someNum
    FROM KORISNIK
    JOIN REZERVACIJA ON KORISNIK.OIB = REZERVACIJA.OIB
    WHERE SUBSTR(CURRENT_DATE,0,9) = SUBSTR(REZERVACIJA.pocetak,0,9)
    AND :new.OIB = KORISNIK.OIB;
    IF (someNum>2) THEN
        RAISE threeDaysException;
    END IF;
EXCEPTION
    WHEN threeDaysException THEN
        RAISE_APPLICATION_ERROR(-20005, 'Only 3 reservations per day per
user allowed');
END;
/

```

```

CREATE OR REPLACE TRIGGER insertTrigger2
BEFORE INSERT ON system.rezervacija
FOR EACH ROW
DECLARE
    timeAlreadyReserverd EXCEPTION;
    someNum INTEGER;
BEGIN
    SELECT COUNT(*) INTO someNum

```

```

FROM REZERVACIJA
WHERE :new.TEREN_ID = TEREN_ID AND
      :new.POCETAK = rezervacija.pocetak;
IF (someNum>0) THEN
    RAISE timeAlreadyReserved;
END IF;
EXCEPTION
    WHEN timeAlreadyReserved THEN
        RAISE_APPLICATION_ERROR(-20005, 'Time slot is already
reserved');
END;
/

CREATE OR REPLACE TRIGGER insertTrigger3
BEFORE INSERT ON system.rezervacija
FOR EACH ROW
DECLARE
    alreadyPlayingElsewhere EXCEPTION;
    someNum INTEGER;
BEGIN
    SELECT COUNT(*) INTO someNum
    FROM REZERVACIJA
    WHERE :new.POCETAK = rezervacija.pocetak
    AND :new.OIB = rezervacija.oib;
    IF (someNum>0) THEN
        RAISE alreadyPlayingElsewhere;
    END IF;
EXCEPTION
    WHEN alreadyPlayingElsewhere THEN
        RAISE_APPLICATION_ERROR(-20005, 'You are already plaing at
another field');
END;
/

CREATE OR REPLACE FUNCTION get_rand_start_time
RETURN VARCHAR2
IS
    rand_start_time integer;
    rand_end_time integer;
    date_only varchar2(30);
    full_start_date varchar2(30);
    full_end_date varchar2(30);
BEGIN
    rand_start_time := dbms_random.value(10,20);
    rand_end_time := rand_start_time + 1;
    date_only:= SUBSTR(CURRENT_DATE,0,9);
    date_only:= CONCAT(date_only,' ');
    full_start_date:= CONCAT(date_only,rand_start_time);
    full_start_date:= CONCAT(full_start_date,':00');
    RETURN full_start_date;
END get_rand_start_time;

declare
    ukupno_terena integer;
    rand_teren integer;
    res_id integer;

```

```

        rand_start_time date;
        rand_end_time date;
begin
    SELECT COUNT(*) INTO ukupno_terena
    FROM teren;

    SELECT COUNT(*) INTO res_id
    FROM rezervacija;

    rand_teren := dbms_random.value(1,ukupno_terena);
    rand_start_time:= TO_DATE(get_rand_start_time(),'dd-MON-yy
hh24:mi');
    rand_end_time:= TO_DATE(rand_start_time+1/24,'dd-MON-yy
hh24:mi');

    INSERT INTO
system.REZERVACIJA (REZERVACIJA_ID,POCETAK,KRAJ,TEREN_ID,OIB) VALUES
(
    res_id+1,
    rand_start_time,
    rand_end_time,
    rand_teren,
    '0311'
);
end;
/

```

## Postavljanje okruženja

```
git clone https://github.com/defilippomattia/uprpod
```

```
cd uprpod/
```

```
Start Docker
```

```
Start SQLDeveloper
```

```
docker-compose up -d --force-recreate
```

```
Paste and Run functions_and_triggers.sql in SQLDeveloper
```

```
alter session set nls_date_format = 'dd-MON-yy hh24:mi'
```

## Početni zahtjevi

**„Korisnici moraju vidjeti sve dostupne terene u njihovoj blizini (po kvartovima) te vidjeti postoji li slobodan termin.”**

```
SELECT TEREN.VRSTA_TERENA, TEREN.NATKRIVEN, SPORTSKI_OBJEKT.IME SPORTSKI_OBJEKT_IME,
KVART.IME KVART_IME
FROM TEREN
JOIN SPORTSKI_OBJEKT ON teren.sportski_objekt_id = sportski_objekt.sportski_objekt_id
JOIN KVART ON sportski_objekt.kvart_id = kvart.kvart_id
```

```
SELECT pocetak, kraj, teren.vrsta_terena, teren.ime TEREN_IME, kvart.ime KVART_IME FROM
REZERVACIJA
JOIN TEREN ON teren.teren_id = rezervacija.teren_id
JOIN SPORTSKI_OBJEKT ON sportski_objekt.sportski_objekt_id = teren.sportski_objekt_id
JOIN KVART ON kvart.kvart_id = sportski_objekt.kvart_id
WHERE CURRENT_DATE < pocetak
```

**“Rezervacija je moguća ako postoji slobodan termin. Jedna osoba može rezervirati jedan termin u određenom trenutku (ne može se dogoditi da osoba rezervira 2 termina u isto vrijeme na različitim terenima).”**

```
SELECT * FROM REZERVACIJA
```

Testiranje dodavanjem istog termina:

```
INSERT INTO system.REZERVACIJA(REZERVACIJA_ID,POCETAK,KRAJ,TEREN_ID,OIB) VALUES (
'9',
TO_DATE('27-NOV-22 14:00', 'dd-MON-yy hh24:mi'),
TO_DATE('27-NOV-22 15:00', 'dd-MON-yy hh24:mi'),
'4',
'0515');
```

Testiranje rezerviranja istog termina u različitim terenima

```
INSERT INTO system.REZERVACIJA(REZERVACIJA_ID,POCETAK,KRAJ,TEREN_ID,OIB) VALUES (
'9',
TO_DATE('27-NOV-22 14:00', 'dd-MON-yy hh24:mi'),
TO_DATE('27-NOV-22 15:00', 'dd-MON-yy hh24:mi'),
'1',
'0515');
```

**“Korisnik može vidjeti i povijest svojih rezervacija.”**

```
SELECT korisnik.oib,pocetak, kraj, teren.vrsta_terena, teren.ime TEREN_IME, kvart.ime KVART_IME
FROM REZERVACIJA
JOIN TEREN ON teren.teren_id = rezervacija.teren_id
JOIN SPORTSKI_OBJEKT ON sportski_objekt.sportski_objekt_id = teren.sportski_objekt_id
JOIN KVART ON kvart.kvart_id = sportski_objekt.kvart_id
JOIN KORISNIK ON korisnik.oib = rezervacija.oib
WHERE pocetak < CURRENT_DATE
```

**“Treba postojati i mogućnost rezerviranja termina na način “I’m feeling lucky” gdje će se rezervirati nasumični (slobodni) termin u nasumičnom kvartu za nasumični sport.”**

```
declare
    ukupno_terena integer;
    rand_teren integer;
    res_id integer;
    rand_start_time date;
    rand_end_time date;
begin
    SELECT COUNT(*) INTO ukupno_terena
    FROM teren;

    SELECT COUNT(*) INTO res_id
    FROM rezervacija;

    rand_teren := dbms_random.value(1,ukupno_terena);
    rand_start_time:= TO_DATE(get_rand_start_time(),'dd-MON-yy
hh24:mi');
    rand_end_time:= TO_DATE(rand_start_time+1/24,'dd-MON-yy
hh24:mi');

    INSERT INTO
system.REZERVACIJA (REZERVACIJA_ID,POCETAK,KRAJ,TEREN_ID,OIB) VALUES
(
    res_id+1,
    rand_start_time,
    rand_end_time,
    rand_teren,
    '0311'
);
end;
/
```

**“Za svaki objekt treba postojati i više informacija o dostupnim sadržajima (kafić u sklopu kompleksa, tuševi, je li moguće iznajmljivanje reketa, sportskih lopti i sl.).”**

```
SELECT SPORTSKI_OBJEKT.ime, dodatni_sadrzaj.tip FROM SPORTSKI_OBJEKT JOIN OBJEKT_SADRZAJ
ON sportski_objekt.sportski_objekt_id = objekt_sadrzaj.sportski_objekt_id
JOIN DODATNI_SADRZAJ ON dodatni_sadrzaj.dodatni_sadrzaj_id =
objekt_sadrzaj.dodatni_sadrzaj_id
```



## Zahtjevi programera

**“Želim dobiti popis svih rezervacija za određenog korisnika u određenom vremenskom razdoblju.”**

```
SELECT SPORTSKI_OBJEKT.IME SPORTSKI_OBJEKT_IME, TEREN.VRSTA_TERENA, TEREN.TEREN_ID,
REZERVACIJA.POCETAK, REZERVACIJA.KRAJ FROM REZERVACIJA
JOIN KORISNIK ON REZERVACIJA.OIB = KORISNIK.OIB
JOIN TEREN ON REZERVACIJA.TEREN_ID = TEREN.TEREN_ID
JOIN SPORTSKI_OBJEKT ON teren.sportski_objekt_id = sportski_objekt.sportski_objekt_id
WHERE REZERVACIJA.POCETAK BETWEEN
TO_DATE('09-NOV-22 14:00', 'dd-MON-yy hh24:mi') AND
TO_DATE('29-DEC-22 14:00', 'dd-MON-yy hh24:mi')
AND KORISNIK.OIB IN ('0515');
```

**“Moguće je napraviti maksimalno 3 rezervacije u danu po osobi.”**

Testiranje:

```
INSERT INTO system.REZERVACIJA(REZERVACIJA_ID, POCETAK, KRAJ, TEREN_ID, OIB) VALUES (
'9',
TO_DATE('15-DEC-22 19:00', 'dd-MON-yy hh24:mi'),
TO_DATE('15-DEC-22 20:00', 'dd-MON-yy hh24:mi'),
'4',
'0515');
```

**Želim dobiti popis svih rezervacija za određeni kvart.**

```
SELECT KVART.IME KVART_IME, SPORTSKI_OBJEKT.IME SPORTSKI_OBJEKT_IME,
TEREN.VRSTA_TERENA, TEREN.TEREN_ID, REZERVACIJA.POCETAK, REZERVACIJA.KRAJ FROM
REZERVACIJA
INNER JOIN TEREN
ON REZERVACIJA.TEREN_ID = TEREN.TEREN_ID
INNER JOIN SPORTSKI_OBJEKT
ON SPORTSKI_OBJEKT.SPORTSKI_OBJEKT_ID = TEREN.SPORTSKI_OBJEKT_ID
INNER JOIN KVART
ON SPORTSKI_OBJEKT.KVART_ID = KVART.KVART_ID
WHERE KVART.IME IN ('Trnje')
```

**Želim dobiti popis svih objekata u kojima je moguće iznajmiti određeni rekvizit i u kojoj je količini trenutno taj rekvizit raspoloživ.**

```
SELECT TIP, SPORTSKI_OBJEKT.IME, COUNT(*) UKUPNO_REKVIZITA FROM REKVIZIT
LEFT OUTER JOIN SPORTSKI_OBJEKT
ON rekvizit.sportski_objekt_id = SPORTSKI_OBJEKT.SPORTSKI_OBJEKT_ID
GROUP BY TIP, SPORTSKI_OBJEKT.IME
```