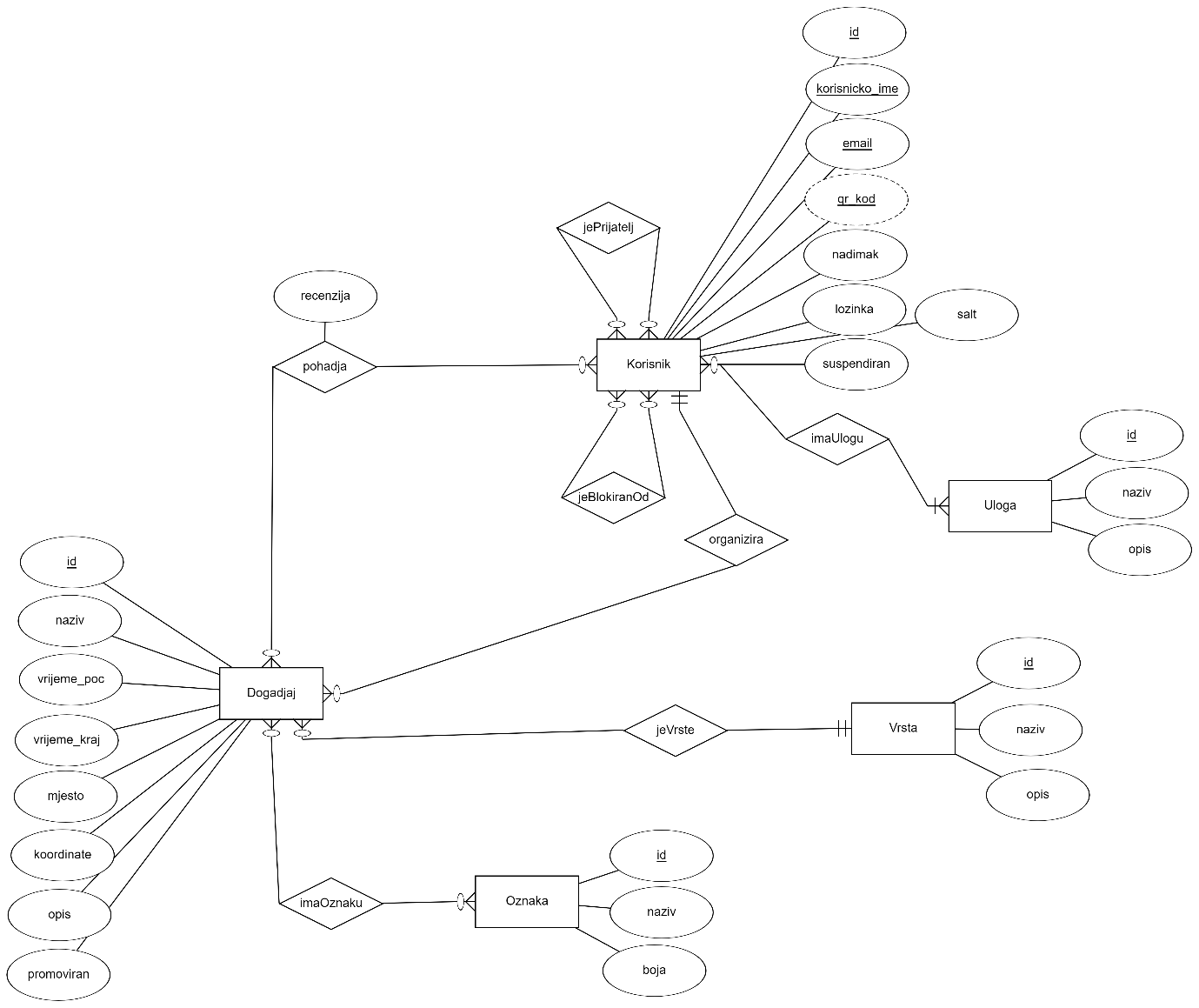
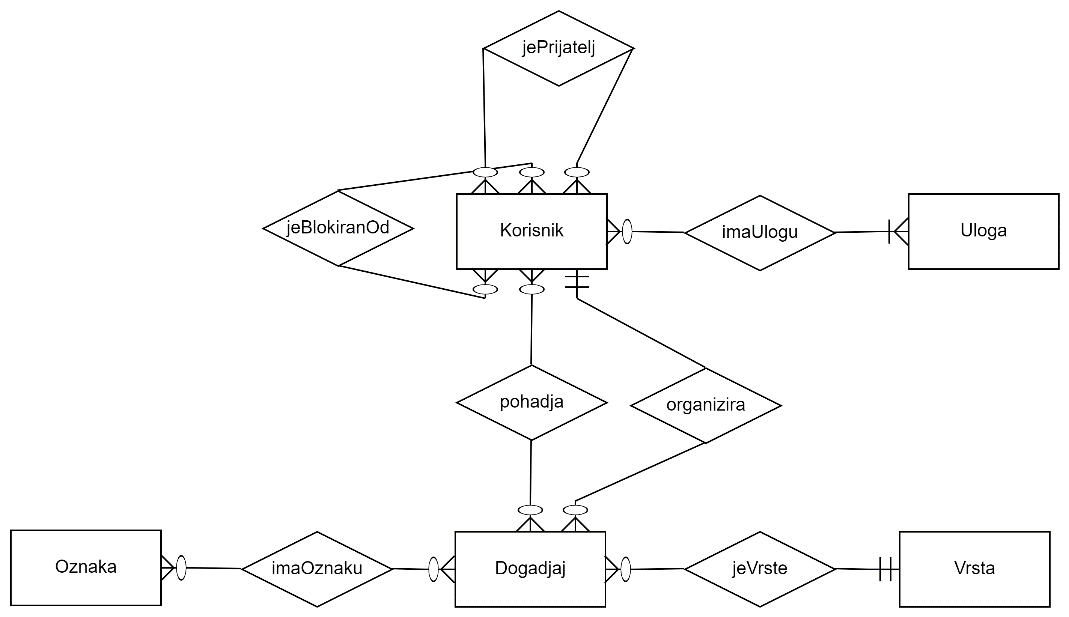
EVENTKO – BAZA PODATAKA

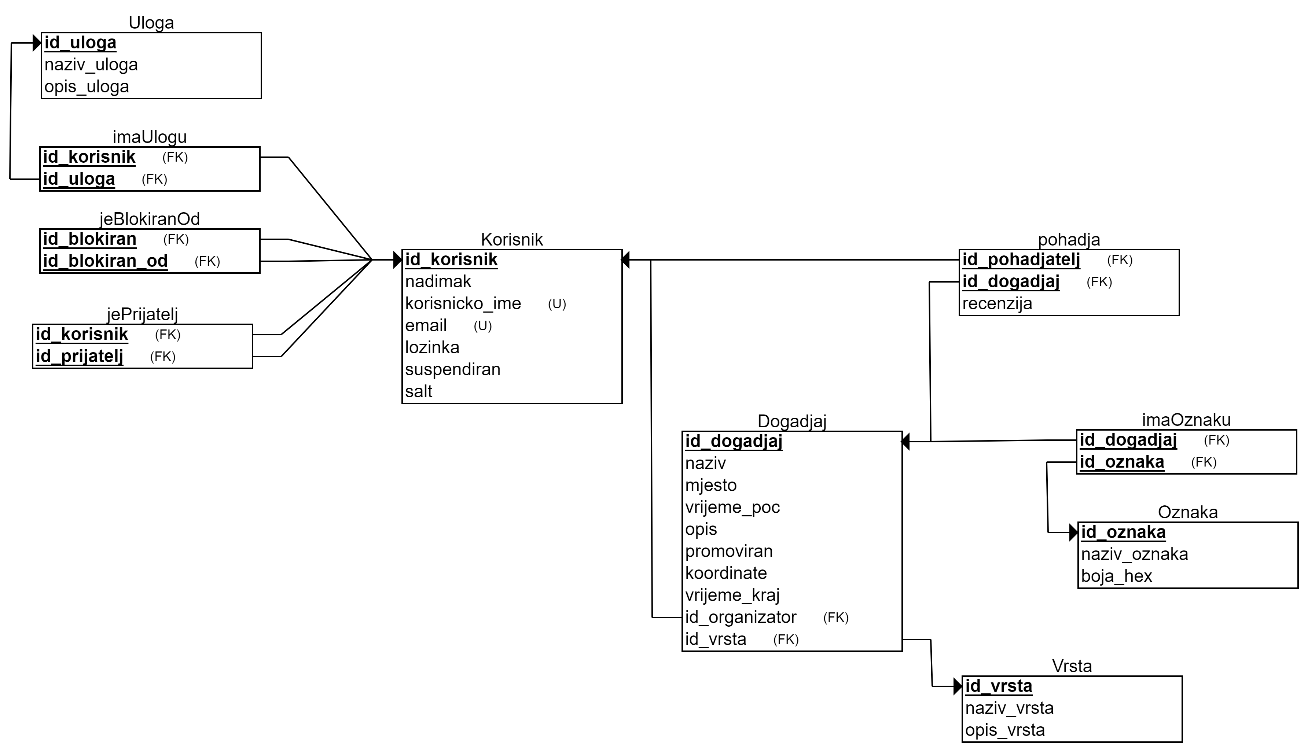
# ER dijagram



# ER dijagram bez atributa



# Relacijska shema



# Važni atributi

## Korisnik

### Korisničko ime

Korisničko ime je jedinstveni identifikator korisničkog računa. Mora imati barem 2 znaka, može imati najviše 25 znakova. Može sadržavati samo znakove engleske abecede, brojeve od 0 do 9 i znak podvlake „\_“ (*Unicode* simbol 0x1F).

### Nadimak

Nadimak je naziv korisničkog računa koji će se u većini slučajeva prikazivati na stranici. Ne mora biti jedinstveno. Ako nije odabrano biti će postavljeno na vrijednost korisničkog imena. Sadrži najmanje 2, a najviše 25 znakova. Ne smije počinjati niti završavati sa znakom razmaka.

### Email

Račun elektroničke pošte vezan uz korisnički račun. Mora bit jedinstven i u ispravnom formatu (npr. *ime@domena.hr*).

### Lozinka

Lozinka je hashirana (pretvorena algoritmom raspršivanja) tekstualna vrijednost proizvoljnog oblika. Tekstualna vrijednost koja se hashira mora sadržavati barem 4 znaka.

## Oznaka

### Boja\_hex

Boja kojom će se prikazivati na stranici oznaka za događaje. Mora biti u heksadekadskom prikazu boje (npr. #fa34e0).

## Pohadja

### Recenzija

Korisnikova recenzija na događaj za koji je prijavio svoj dolazak. Može biti 0 (nije ocijenio), -1 (negativna ocjena) ili 1 (pozitivna ocjena).

# SQL opis baze

DROP TABLE IF EXISTS Korisnik CASCADE;

DROP TABLE IF EXISTS Uloga CASCADE;

DROP TABLE IF EXISTS imaUlogu;

DROP TABLE IF EXISTS jeBlokiranOd;

DROP TABLE IF EXISTS jePrijatelj;

DROP TABLE IF EXISTS Dogadjaj CASCADE;;

DROP TABLE IF EXISTS pohadja;

DROP TABLE IF EXISTS imaOznaku;

DROP TABLE IF EXISTS Oznaka;

DROP TABLE IF EXISTS Vrsta;

DROP TABLE IF EXISTS SPRING\_SESSION CASCADE;

DROP TABLE IF EXISTS SPRING\_SESSION\_ATTRIBUTES;

DROP SEQUENCE IF EXISTS hibernate\_sequence;

CREATE TABLE Korisnik

(

id\_korisnik BIGINT NOT NULL,

nadimak VARCHAR(25) NOT NULL,

korisnicko\_ime VARCHAR(25) NOT NULL,

email VARCHAR(255) NOT NULL,

salt BYTEA NOT NULL,

lozinka BYTEA NOT NULL,

suspendiran BOOLEAN NOT NULL,

PRIMARY KEY (id\_korisnik),

UNIQUE (korisnicko\_ime),

UNIQUE (email),

CHECK (email LIKE '%\_@\_\_%.\_\_%'), --Provjera formata emaila

CHECK (nadimak NOT LIKE ' \*' AND nadimak NOT LIKE '\* '), --Provjera da nadimak ne pocinje ili zavrsava s razmakom

CHECK (korisnicko\_ime ~ '^[A-Za-z0-9\_]+$') --Provjera da korisnicko ime ima samo alfanumeričke simbole

);

CREATE TABLE Uloga

(

id\_uloga BIGINT NOT NULL,

naziv\_uloga VARCHAR(255) NOT NULL,

opis\_uloga VARCHAR(255) NOT NULL,

PRIMARY KEY (id\_uloga)

);

CREATE TABLE Vrsta

(

id\_vrsta INT NOT NULL,

naziv\_vrsta VARCHAR(255) NOT NULL,

opis\_vrsta VARCHAR(255) NOT NULL,

PRIMARY KEY (id\_vrsta)

);

CREATE TABLE Oznaka

(

id\_oznaka INT NOT NULL,

naziv\_oznaka VARCHAR(255) NOT NULL,

boja\_hex CHAR(7) NOT NULL,

PRIMARY KEY (id\_oznaka),

CHECK (boja\_hex IS NULL OR boja\_hex ~\* '^#[a-f0-9]{6}$') --Provjera formata hex boje

);

CREATE TABLE imaUlogu

(

id\_korisnik BIGINT NOT NULL,

id\_uloga BIGINT NOT NULL,

PRIMARY KEY (id\_korisnik, id\_uloga),

FOREIGN KEY (id\_korisnik) REFERENCES Korisnik(id\_korisnik),

FOREIGN KEY (id\_uloga) REFERENCES Uloga(id\_uloga)

);

CREATE TABLE jePrijatelj

(

id\_korisnik BIGINT NOT NULL,

id\_prijatelj BIGINT NOT NULL,

PRIMARY KEY (id\_korisnik, id\_prijatelj),

FOREIGN KEY (id\_korisnik) REFERENCES Korisnik(id\_korisnik),

FOREIGN KEY (id\_prijatelj) REFERENCES Korisnik(id\_korisnik)

);

CREATE TABLE jeBlokiranOd

(

id\_blokiran BIGINT NOT NULL,

id\_blokiran\_od BIGINT NOT NULL,

PRIMARY KEY (id\_blokiran, id\_blokiran\_od),

FOREIGN KEY (id\_blokiran) REFERENCES Korisnik(id\_korisnik),

FOREIGN KEY (id\_blokiran\_od) REFERENCES Korisnik(id\_korisnik)

);

CREATE TABLE Dogadjaj

(

id\_dogadjaj BIGINT NOT NULL,

naziv VARCHAR(255) NOT NULL,

mjesto VARCHAR(255) NOT NULL,

vrijeme\_poc TIMESTAMP NOT NULL,

vrijeme\_kraj TIMESTAMP NOT NULL,

opis VARCHAR(255) NOT NULL,

promoviran BOOLEAN NOT NULL,

koordinate VARCHAR(255) NOT NULL, --smisliti kako će se pohranjivati koordinate

id\_organizator BIGINT NOT NULL,

id\_vrsta INT NOT NULL,

PRIMARY KEY (id\_dogadjaj),

FOREIGN KEY (id\_organizator) REFERENCES Korisnik(id\_korisnik),

FOREIGN KEY (id\_vrsta) REFERENCES Vrsta(id\_vrsta),

CHECK (vrijeme\_poc < vrijeme\_kraj)

);

CREATE TABLE pohadja

(

recenzija SMALLINT NOT NULL,

id\_pohadjatelj BIGINT NOT NULL,

id\_dogadjaj BIGINT NOT NULL,

PRIMARY KEY (id\_pohadjatelj, id\_dogadjaj),

FOREIGN KEY (id\_pohadjatelj) REFERENCES Korisnik(id\_korisnik),

FOREIGN KEY (id\_dogadjaj) REFERENCES Dogadjaj(id\_dogadjaj),

CHECK ((recenzija = 1) OR (recenzija = -1) OR (recenzija = 0))

);

CREATE TABLE imaOznaku

(

id\_dogadjaj BIGINT NOT NULL,

id\_oznaka INT NOT NULL,

PRIMARY KEY (id\_dogadjaj, id\_oznaka),

FOREIGN KEY (id\_dogadjaj) REFERENCES Dogadjaj(id\_dogadjaj),

FOREIGN KEY (id\_oznaka) REFERENCES Oznaka(id\_oznaka)

);

INSERT INTO vrsta VALUES (1, 'Obveza', 'Događaj koji se odnosi samo na korisnika koji ga je stvorio');

INSERT INTO vrsta VALUES (2, 'Privatni događaj', 'Događaj koji se odnosi na korisnika koji ga je stvorio i ljude koje je pozvao');

INSERT INTO vrsta VALUES (3, 'Javni događaj', 'Događaj na koji se mogu prijaviti svi korisnici platforme, osim onih koje je korisnik, koji je događaj stvorio, blokirao');

INSERT INTO uloga VALUES (1, 'Korisnik', 'Običan korisnik platforme');

INSERT INTO uloga VALUES (2, 'Premium korisnik', 'Korisnik koji ima mogućnost promocije svojih javnih događaja');

INSERT INTO uloga VALUES (3, 'Moderator', 'Korisnik koji ima mogućnost suspendiranja ostalih korisnika, brisanja javnih događaja i uređivanja oznaka na javnim događajima');

INSERT INTO uloga VALUES (4, 'Administrator', 'Korisnik koji ima mogućnost promocije korisnika na moderatore i brisanja korisničkih računa');

INSERT INTO oznaka VALUES (1, 'Kava', '#6f4e37');

INSERT INTO oznaka VALUES (2, 'Piva', '#f28e1c');

--Treba dodat još oznaka...

--Spring session

CREATE TABLE SPRING\_SESSION (

PRIMARY\_ID CHAR(36) NOT NULL,

SESSION\_ID CHAR(36) NOT NULL,

CREATION\_TIME BIGINT NOT NULL,

LAST\_ACCESS\_TIME BIGINT NOT NULL,

MAX\_INACTIVE\_INTERVAL INT NOT NULL,

EXPIRY\_TIME BIGINT NOT NULL,

PRINCIPAL\_NAME VARCHAR(100),

CONSTRAINT SPRING\_SESSION\_PK PRIMARY KEY (PRIMARY\_ID)

);

CREATE UNIQUE INDEX SPRING\_SESSION\_IX1 ON SPRING\_SESSION (SESSION\_ID);

CREATE INDEX SPRING\_SESSION\_IX2 ON SPRING\_SESSION (EXPIRY\_TIME);

CREATE INDEX SPRING\_SESSION\_IX3 ON SPRING\_SESSION (PRINCIPAL\_NAME);

CREATE TABLE SPRING\_SESSION\_ATTRIBUTES (

SESSION\_PRIMARY\_ID CHAR(36) NOT NULL,

ATTRIBUTE\_NAME VARCHAR(200) NOT NULL,

ATTRIBUTE\_BYTES BYTEA NOT NULL,

CONSTRAINT SPRING\_SESSION\_ATTRIBUTES\_PK PRIMARY KEY (SESSION\_PRIMARY\_ID, ATTRIBUTE\_NAME),

CONSTRAINT SPRING\_SESSION\_ATTRIBUTES\_FK FOREIGN KEY (SESSION\_PRIMARY\_ID) REFERENCES SPRING\_SESSION(PRIMARY\_ID) ON DELETE CASCADE

);

CREATE SEQUENCE hibernate\_sequence MINVALUE 2;

--Testni podaci

INSERT into Korisnik VALUES (1, 'Admin', 'admin', 'admin.adminic@fer.hr',CONVERT\_TO('sol', 'UTF8'), '1234', FALSE);

INSERT into imaulogu VALUES (1, 1);

INSERT into imaulogu VALUES (1, 2);

INSERT into imaulogu VALUES (1, 3);

INSERT into imaulogu VALUES (1, 4);

INSERT INTO dogadjaj VALUES(1, 'Testni dogadjaj', 'FER', NOW(), NOW() + '23 HOURS'::INTERVAL, 'opis dogadjaja', FALSE, '0, 0', 1, 1);

INSERT INTO pohadja VALUES (1, 1, 1);

INSERT INTO imaoznaku VALUES (1, 1);