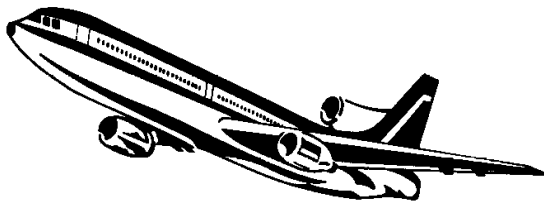


**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
ELEKTROTEHNIČKI FAKULTET**



Stručni studij



Aerodrom

**Projektni zadatak iz kolegija
Baze podataka**

Ime Prezime. Axxxx

Osijek, 2011.

Sadržaj

1.	OPIS PROJEKTOG ZADATKA.....	1
2.	ER DIJAGRAM.....	2
3.	RELACIJSKI MODEL.....	3
4.	OPIS ODABRANOG RJEŠENJA.....	4
5.	SQL NAREDBE ZA KREIRANJE BAZE PODATAKA.....	5
6.	PUNJENJE BAZE PODACIMA.....	7
7.	UPITI KOJI ĆE BITI NAJČEŠĆE KORIŠTENI.....	8
8.	ZAKLJUČAK.....	10
9.	REFERENCE.....	11

Marenić

1. OPIS PROJEKTOG ZADATKA

“Aerodrom”

Opis:

Potrebno je osmisliti i realizirati bazu podataka koja može učinkovito poslužiti za evidenciju prometa na aerodromu. U fazi analize, utvrđeno je da korisnik želi imati ažurnu evidenciju letova (oznaka, tip aviona, kompanija, polazište, odredište, datum i sat polaska, datum i sat dolaska, udaljenost, cijena leta...), kompanija (naziv, sjedište, ...), rezervacija (tko, datum polaska, polazište, destinacija), putnika, aviona (oznaka, model, broj sjedala, dolet, ...) te mogućih polazišta i destinacija (naziv). Osnovni zahtjevi korisnika su slijedeći:

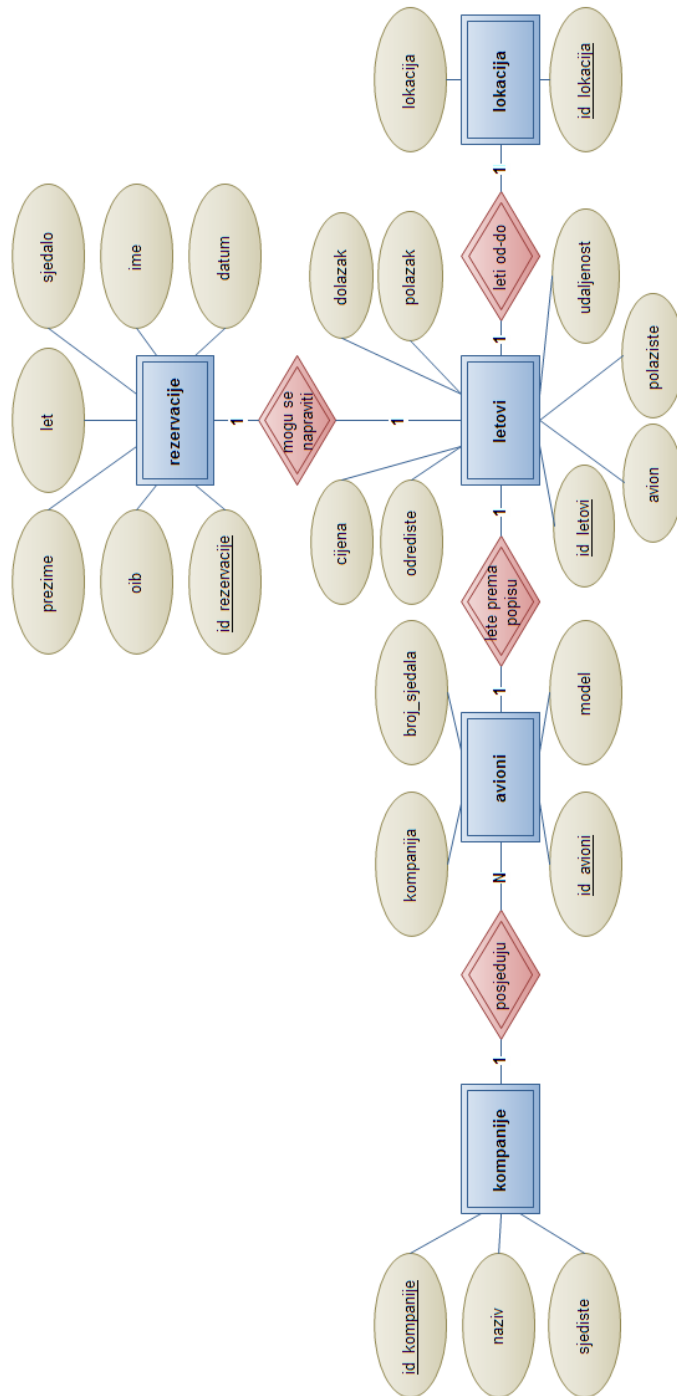
- omogućiti uvid u letove (prema odredištu, destinaciji, polasku, kompaniji, tipu aviona, ...)
- pregled budućih rezervacija
- izračun ukupne cijene za sve letove određene osobe unutar zadanog vremenskog perioda
- pregled slobodnih mjesta prema oznaci leta

Baza podataka treba biti u što većoj mjeri normalizirana.

Zadaci:

1. Napraviti ER dijagram (entity-relationship diagram) dijagram sa svim bitnim elementima.
2. Napraviti relacijski model
3. Dati kraći tekstualni opis odabranog rješenja uz osvrt na eventualne specifičnosti, pretpostavke ili ograničenja u modelu.
4. Napraviti SQL naredbe za kreiranje baze podataka koja odgovara relacijskom modelu.
5. Napraviti SQL naredbe kojima se baza puni podacima za potrebe testiranja.
6. Napraviti primjer SQL upita za koje se očekuje da će biti najčešće upotrebljavani od strane korisnika baze podataka (prema zahtjevima u opisu zadatka), opisati ih riječima (koja je svrha upita?) i dati konkretni primjer rezultata kakav se dobije takvim upitom.

2. ER DIJAGRAM



Slika 1. ER dijagram baze Aerodrom

3. RELACIJSKI MODEL

avioni				
Field	Type	Null	Default	
<u>id_avioni</u>	int(15)	No		
model	varchar(50)	No		
kompanija	int(15)	No		kompanije -> id_kompanije
broj_sjedala	int(11)	No		

kompanije				
Field	Type	Null	Default	
<u>id_kompanije</u>	int(15)	No		
naziv	varchar(100)	No		
sjediste	varchar(150)	No		

letovi				
Field	Type	Null	Default	
<u>id_letovi</u>	int(11)	No		
avion	int(15)	No		avioni -> id_avioni
polaziste	int(15)	No		lokacije -> id_lokacija
odrediste	int(15)	No		lokacije -> id_lokacija
polazak	datetime	No		
dolazak	datetime	No		
udaljenost	float(10,2)	No		
cijena	float(10,2)	No		

lokacije				
Field	Type	Null	Default	
<u>id_lokacija</u>	int(15)	No		
lokacija	varchar(50)	No		

rezervacije				
Field	Type	Null	Default	
<u>id_rezervacije</u>	int(11)	No		
datum	date	No		
ime	varchar(50)	No		
prezime	varchar(75)	No		
oib	char(11)	No		
let	int(15)	No		letovi -> id_letovi
sjedalo	varchar(5)	No		

Slika 2. Relacijski model baze Aerodrom

4. OPIS ODABRANOG RJEŠENJA

Kao rješenje projektnog zadatka kreirana je baza sa 5 tablica. Rješenje bi trebalo zadovoljavati sve zahtjeve navedene u projektnom zadatku.

Tablica „avioni“

Ova tablica sadrži podatke o tipu aviona, broju sjedala te kompaniji kojoj avion pripada. Kao primarni ključ postavljen je id_avioni, to je ujedno i strani ključ za tablicu letovi. Tablica je ograničena s jednim stranim ključem iz tablice „kompanije“ (id_kompanije).

Tablica „letovi“

Najveća tablica. Sadrži podatke o rutama aviona te njihovim cijenama i udaljenosti između njih. Primarni ključ je id_letovi koji je samopovećavajući te je tipa integer. Ujedno je strani ključ za tablicu rezervacije. Ova tablica je ograničena sa više stranih ključeva iz tablica „lokacije“ i „avioni“ (id_lokacija, id_avioni).

Tablica „rezervacije“

Tablica sadrži podatke o rezerviranim letovima za putnike. Kao primarni ključ postavljen je id_rezervacije. Tablica je ograničena sa stranim ključem id_letovi.

Tablica „kompanije“

Tablica sadrži podatke kao što je naziv kompanije i sjedište. Primarni ključ je id_kompanije, on je ujedno i strani ključ za tablicu „avioni“.

Tablica „lokacije“

Mala tablica koja sadrži podatak o lokaciji polazišta ili odredišta. Primarni ključ je id_lokacija a ujedno je i strani ključ za tablicu „letovi“.

5. SQL NAREDBE ZA KREIRANJE BAZE PODATAKA

Kreiranje baze podataka:

```
CREATE DATABASE `aerodrom` DEFAULT CHARACTER SET latin1  
COLLATE latin1_swedish_ci;  
USE `aerodrom`;
```

Kreiranje tablice „avioni“

```
CREATE TABLE IF NOT EXISTS `avioni` (  
  `id_avioni` int(15) NOT NULL AUTO_INCREMENT,  
  `model` varchar(50) NOT NULL,  
  `kompanija` int(15) NOT NULL,  
  `broj_sjedala` int(11) NOT NULL,  
  PRIMARY KEY (`id_avioni`),  
  KEY `kompanija` (`kompanija`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=68 ;
```

Kreiranje tablice „kompanije“

```
CREATE TABLE IF NOT EXISTS `kompanije` (  
  `id_kompanije` int(15) NOT NULL AUTO_INCREMENT,  
  `naziv` varchar(100) NOT NULL,  
  `sjediste` varchar(150) NOT NULL,  
  PRIMARY KEY (`id_kompanije`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=5 ;
```

Kreiranje tablice „letovi“

```
CREATE TABLE IF NOT EXISTS `letovi` (  
  `id_letovi` int(11) NOT NULL AUTO_INCREMENT,  
  `avion` int(15) NOT NULL,  
  `polaziste` int(15) NOT NULL,  
  `odrediste` int(15) NOT NULL,  
  `polazak` datetime NOT NULL,  
  `dolazak` datetime NOT NULL,  
  `udaljenost` float(10,2) NOT NULL,  
  `cijena` float(10,2) NOT NULL,  
  PRIMARY KEY (`id_letovi`),  
  KEY `avion` (`avion`),  
  KEY `polaziste` (`polaziste`),  
  KEY `odrediste` (`odrediste`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=6547 ;
```

Kreiranje tablice „lokacije“

```
CREATE TABLE IF NOT EXISTS `lokacije` (  
  `id_lokacija` int(15) NOT NULL AUTO_INCREMENT,  
  `lokacija` varchar(50) NOT NULL,  
  PRIMARY KEY (`id_lokacija`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=556 ;
```

Kreiranje tablice „rezervacije“

```
CREATE TABLE IF NOT EXISTS `rezervacije` (  
  `id_rezervacije` int(11) NOT NULL AUTO_INCREMENT,  
  `datum` date NOT NULL,  
  `ime` varchar(50) NOT NULL,  
  `prezime` varchar(75) NOT NULL,  
  `oib` char(11) NOT NULL,  
  `let` int(15) NOT NULL,  
  `sjedalo` varchar(5) NOT NULL,  
  PRIMARY KEY (`id_rezervacije`),  
  KEY `let` (`let`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=55556 ;
```

Postavljanje ograničenja (FOREIGN KEY)

```
ALTER TABLE `avioni`  
  ADD CONSTRAINT `avioni_ibfk_1` FOREIGN KEY (`kompanija`)  
  REFERENCES `kompanije` (`id_kompanije`) ON DELETE CASCADE ON  
UPDATE CASCADE;
```

Ograničenje na tablici „avioni“.

```
ALTER TABLE `letovi`  
  ADD CONSTRAINT `letovi_ibfk_3` FOREIGN KEY (`odrediste`)  
  REFERENCES `lokacije` (`id_lokacija`) ON DELETE CASCADE ON  
UPDATE CASCADE,  
  ADD CONSTRAINT `letovi_ibfk_1` FOREIGN KEY (`avion`)  
  REFERENCES `avioni` (`id_avioni`) ON DELETE CASCADE ON UPDATE  
CASCADE,  
  ADD CONSTRAINT `letovi_ibfk_2` FOREIGN KEY (`polaziste`)  
  REFERENCES `lokacije` (`id_lokacija`) ON DELETE CASCADE ON  
UPDATE CASCADE;
```

Ograničenje na tablici „letovi“. Vidi se da ima tri strana ključa.

```
ALTER TABLE `rezervacije`  
  ADD CONSTRAINT `rezervacije_ibfk_1` FOREIGN KEY (`let`)  
  REFERENCES `letovi` (`id_letovi`) ON DELETE CASCADE ON UPDATE  
CASCADE;
```

Ograničenje na tablici „rezervacije“. Strani ključ je id_letovi.

6. PUNJENJE BAZE PODACIMA

Unos podataka u tablicu „avioni“

```
INSERT INTO `aerodrom`.`avioni` (`id_avioni`, `model`,  
`kompanija`, `broj_sjedala`  
)  
VALUES (NULL , 'Cesna 55', '6', '4');
```

Unos podataka u tablicu „kompanije“

```
INSERT INTO `aerodrom`.`kompanije` (`id_kompanije`, `naziv`,  
`sjediste`)  
VALUES (NULL , 'Lufthansa', 'Cologne');
```

Unos podataka u tablicu „letovi“

```
INSERT INTO `aerodrom`.`letovi` (`id_letovi`, `avion`,  
`polaziste`, `odrediste`, `polazak`, `dolazak`, `udaljenost`,  
`cijena`  
)  
VALUES (  
NULL , '68', '556', '555', '2011-01-31 15:57:51', '2011-01-31  
21:25:00', '1150.65', '956.00'  
);
```

Unos podataka u tablicu „lokacije“

```
INSERT INTO `aerodrom`.`lokacije` (`id_lokacija`, `lokacija`  
)  
VALUES (NULL , 'Berlin');
```

Unos podataka u tablicu „rezervacije“

```
INSERT INTO `aerodrom`.`rezervacije` (`id_rezervacije`,  
`datum`, `ime`, `prezime`, `oib`, `let`, `sjedalo`  
)  
VALUES (  
NULL, '2011-02-01', 'Ivan', 'Boris', '45892638984', '6547', '2'  
);
```

7. UPITI KOJI ĆE BITI NAJČEŠĆE KORIŠTENI

Ispis podataka iz tablice „avioni“:

```
SELECT * FROM avioni;
```

Upit služi za prikaz svih podataka u tablici avioni. Ovim upitom ne dobijemo dobro ispisane podatke koji se kriju iza stranog ključa pa ćemo koristiti slijedeći uvjet:

```
SELECT `avioni`.`model` AS `model aviona`, `kompanije`.`naziv`  
AS `ime kompanije`, `avioni`.`broj_sjedala`,  
`kompanije`.`sjediste` AS `sjediste`  
FROM `kompanije`, `avioni`  
WHERE `avioni`.`kompanija`=`kompanije`.`id_kompanije` AND  
`avioni`.`kompanija`=`kompanije`.`id_kompanije`
```

Ovim upitom dobijemo podatke prikazane kao da se nalaze u jednoj tablici. Naredbom AS mijenjamo naziv stupca pri prikazu tako da bude jasnije koji se uvjet poziva. Pomoću naredbe WHERE postavili smo uvjet da se pridruže pripadajući podaci preko stranog ključa.

Ispis podataka iz tablice „letovi“:

```
SELECT * FROM letovi;
```

Upitom dobijemo prikazane sve podatke iz tablice posudba. Naravno tim podacima nisu pridružene vrijednosti koje su preko stranog ključa povezane na druge tablice. Zato koristimo slijedeće naredbe:

```
SELECT letovi.id_letovi, model AS avion, lokacije.lokacija AS  
polaziste, (SELECT lokacija FROM lokacije WHERE  
letovi.odrediste = lokacije.id_lokacija ) AS odrediste,  
polazak, dolazak, udaljenost AS 'udaljenost [km]', cijena AS  
'cijena [kn]'  
FROM letovi  
JOIN (avioni, lokacije) ON (letovi.avion = avioni.id_avioni  
AND letovi.polaziste = lokacije.id_lokacija);
```

Ovim upitom dobijemo lijepo prikazane podatke o svim letovima. Naredbom JOIN pridružili smo podatke iz drugih tablica koje su povezane na tablicu letovi pomoću stranog ključa.

Ispis podatka iz tablice „rezervacije“:

```
SELECT datum AS 'datum rezervacije', ime, prezime, oib, let, sjedalo FROM rezervacije;
```

Ovim upitom uzimaju se podaci iz tablice „rezervacije“. Nije korišten nikakvi uvjet niti naredba JOIN jer se ubiti uzima samo id leta a po potrebi može se urediti da prikazuje i ostale podatke koji su pridruženi toj tablici.

Ispis podataka iz tablice „lokacije“:

```
SELECT * FROM lokacije;
```

Upit služi za prikaz svih podataka u tablici „lokacije“.

Ispis podataka iz tablice „kompanije“:

```
SELECT * FROM kompanije;
```

Upit služi za prikaz svih podataka u tablici „kompanije“.

Izmjena podatka u tablici „letovi“:

```
UPDATE `aerodrom`.`letovi` SET `dolazak` = '2010-01-06 07:22:00' WHERE `letovi`.`id_letovi` =3333 LIMIT 1 ;
```

Brisanje podatka u tablici „rezervacije“:

```
DELETE FROM `aerodrom`.`rezervacije` WHERE `rezervacije`.`id_rezervacije` = 44444 LIMIT1
```

Gornji upit briše jedan red podataka iz tablice „rezervacije“ koji ima id = 44444.

8. ZAKLJUČAK

Izradom zadatka upoznali smo se sa osnovama baza podataka. Uvidjeli smo kako u praksi radi mysql baza (XAMPP). Naučili smo koje su osnovne naredbe SQL jezika kao što su SELECT, INSERT, UPDATE, JOIN, AS, DELETE, AND. Također smo vidjeli razliku u tipu baze. S tipom InnoDB omogućeno nam je kreiranje stranih ključeva i ograničenja na bazu.

XAMPP samo koristili jer je vrlo mali i jednostavan program sa svim potrebnim dodacima. Za crtanje ER digrama korišten je program Edraw UML Diagram 5.1, a za ispitivanje uvjeta korišten je program MySQL Query Browser. Njime je ispitivano da li su uvjeti pri ispisu podataka zadovoljeni.

9. REFERENCE

- MySQL GUI Tools, <http://dev.mysql.com/downloads/gui-tools/5.0.html>, program za ispitivanje valjanosti mysql koda
- W3schools, <http://www.w3schools.com/sql/default.asp>, tutoriali za SQL
- MySQL, <http://dev.mysql.com/doc/refman/5.0/en/tutorial.html>, službeni tutorial za zadnju verziju mysqla