

# VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE TEHNOLOGIJE



**ITS** INFORMATION  
TECHNOLOGY  
SCHOOL

VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA IT

POWERED BY  COMTRADE | **LINKgroup**

## Završni rad

# Projektovanje i realizacija TourneyPro aplikacije za organizovanje raznih turnira

Mentor:

dr.Mladen Radaković

Student:

Luka  
Urošević  
561/18

Beograd,

Septembar, 2024

1. REZIME .....	3
2. KLJUČNE REČI .....	3
3. UVOD .....	1
4. PRIMENJENE TEHNOLOGIJE ZA RAZVOJ "TOURNEYPRO" APLIKACIJE .....	1
5. SPECIFIKACIJA KORISNIČKIH ZAHTEVA .....	2
5.1 VERBALNI OPIS .....	2
5.2 SLUČAJEVI KORIŠĆENJA .....	2
5.1. OPIS SLUČAJEVA KORIŠĆENJA.....	4
6. FAZA ANALIZE .....	12
6.1. SISTEMSKI DIJAGRAMI SEKVENCI ZA SK .....	12
6.2. KONCEPTUALNI(DOMENSKI) MODEL .....	34
6.3. RELACIONI MODEL .....	34
7. FAZA PROJEKTOVANJA .....	35
7.1. DIJAGRAMI SEKVENCI ZA SISTEMSKE OPERACIJE.....	35
7.2. PROJEKTOVANJE SKLADIŠTA PODATAKA .....	44
7.3. PROJEKTOVANJE KORISNIČKOG INTERFEJSA .....	46
8. ZAKLJUČAK .....	55
9. LITERATURA .....	55
9.1. Fizička literatura .....	55
9.2. Internet literatura .....	55

## **1. REZIME**

U završnom radu je projektovana i realizovana aplikacija "TourneyPro" čija će namena biti olakšano organizovanje bilo koje vrste turnira, koji koriste sistem "Single-elimination bracket". Aplikacija će pojednostaviti upravljanje učesnicima, žrebom, mečevima i nagradama. U uvodnom delu će biti opisana svrha aplikacije kao i njene karakteristike, zatim i ukratko objašnjene tehnologije koje su korišćene u kodiranju aplikacije. Najznačajniji i najveći deo rada predstavlja razrada gde je prikazan proces projektovanja same aplikacije kroz tri faze: specifikacija korisničkih zahteva, faza analize i faza projektovanja. Na kraju rada se nalazi zaključak koji sadrži osvrt na celokupan rad, kao i mogućnosti budućih unapređenja.

Sama aplikacija će biti izrađena u Microsoft Visual Studio okruženju. Prilikom izrade backend dela biće korišćen C# programski jezik, za grafički prikaz user interface-a biće korišćen podsistem Visual Studia, Windows Presentation foundation(WPF). Za modelovanje i skladištenje podataka aplikacija će koristiti Microsoft-ov SQL Server Managment Studio(SSMS) koji je baziran na korišćenju SQL-a.

## **2. KLJUČNE REČI**

TourneyPro, desktop aplikacija, SQL, takmičar, organizator, VisualStudio, Larmanova metoda, baza podataka, WPF...

### 3. UVOD

"TourneyPro" je desktop aplikacija namenjena organizatorima "Single-Elimination Bracket" turnira. Namenjena je za realnu upotrebu i ima ulogu u olakšavanju same organizacije. Pri prvom pristupu aplikaciji zahtevano je od korisnika da se registruje a kasnije i da se uloguje pri svakom korišćenju. Aplikacija ima tri vrste korisnika(role): administrator, organizator, takmičar. Organizator je korisnik koji ima ulogu organizovanja samog turnira. Od funkcionalnosti poseduje pregled turnira, kreiranje novog turnira, brisanje turnira, dodavanje, izmenu i brisanje takmičara, unos rezultata, dodavanje I brisanje nagrada, kao i kreiranje žreba. Organizator može da koristi ove funkcionalnosti samo za turnire koje je on organizovao. Učesnik je korisnik koji se takmiči na nekom od turnira. On može da pristupi aplikaciji i omogućen mu je uvid u sve turnire na kojima se takmiči. Takmičar može da pregleda žreb i nagrade. Administrator je uloga sistema koja, pre svega, ima sve funkcionalnosti kao organizator, s tim da admin može pristupiti svim turnirima. Takođe, administrator može da upravlja svim korisnicima. On ima funkcionalnosti brisanja i izmene korisnika. U to spada izmena role i email-a.

U završnom radu će detaljno biti opisan proces projektovanja i implementacije "TourneyPro" aplikacije primenom Larmanove metode razvoja softvera. Za razvoj aplikacije će se koristiti sledece tehnologije: Microsoft Visual Studio IDE, programski jezik C#, Server Management Studio(SSMS), SQL i Windows Presentation Foundation(WPF).

### 4. PRIMENJENE TEHNOLOGIJE ZA RAZVOJ "TOURNEYPRO" APLIKACIJE

**Microsoft Visual Studio** je integrisano razvojno okruženje(IDE) koje se koristi za razvoj softverskih aplikacija. Pruža alate za pisanje, debugovanje i testiranje koda. U Visual Studiu možete pisati kod, praviti interfejs, upravljati bazama podataka i testirati aplikacije. Ima podršku za više programskih jezika, uključujući i C#. Ima prednost integracije sa drugim Microsoft alatima, kao i podršku za razvoj desktop, mobilnih i web aplikacija.

**C#** je višenamenski, objektno-orijentisani programski jezik koji je razvio Microsoft kao deo .NET platforme. Koristi se za razvoj raznovrsnih aplikacija, uključujući web aplikacije, desktop aplikacije, igre, mobilne aplikacije, I cloud-based servise. Sintaksa samog jezika je veoma jednostavna.

**Server Management Studio(SSMS)** je alat za upravljanje relacionim bazama podataka, razvijen od strane Microsoft-a. Omogućava administraciju, konfiguraciju, upravljanje I razvoj SQL Server baze podataka. Koristi se za pisanje I izvršavanje SQL upita, upravljanje bazama podataka, konfiguraciju servera, itd... Poseduje user-friendly interfejs I mogućnost integracije sa Visual Studi-om.

**SQL** iliti Structured Query Language je standardizovani jezik za upravljanje I manipulaciju relacionim bazama podataka. Koristi se za kreiranje, čitanje, ažuriranje I brisanje podataka u bazi podataka. Takođe se koristi za kreiranje struktura baze kao što su tabele. Omogućava efikasan rad sa velikim količinama podataka. Podržan je od strane većine Sistema za upravljanje bazama podataka(DBMS).

**WPF** je napredna tehnologija za razvoj aplikacija, razvijena od strane Microsofta. Koristi se za kreiranje modernih I interaktivnih korisničkih interfejsa za Windows. WPF se ističe po svojoj

sposobnosti korišćenja XAML-a(eXtensible Application Markup Language) za deklarativno definisanje izgleda aplikacija, čime omogućuje razdvajanje dizajna i business logike.

## **5. SPECIFIKACIJA KORISNIČKIH ZAHTEVA**

### **5.1 VERBALNI OPIS**

“TourneyPro” je desktop aplikacija namenjena organizatorima turnira za pomoć u samoj organizaciji. Koristi je tri vrste korisnika: organizator, takmičar i admin. Pri pristupu aplikaciji potrebno je da se korisnik uloguje ili registruje ukoliko već nije registrovan.

Nakon što se uloguje, organizator može da pristupi turnirima koje je već organizovao ili da kreira novi. Pri kreiranju novog turnira, organizator mora da unese određene informacije vezano za samo takmičenje (naziv takmičenja, broj učesnika, datum početka, da li želi da upravlja nagradama, i ako želi potrebno je uneti iznos kotizacije). Takođe, organizator ima mogućnost da obriše ili izmeni informacije već kreiranog turnira. Kada pristupi ili kreira turnir ima mogućnost da dodaje, briše ili izmeni takmičare. Pri dodavanju takmičara organizator može da izabere već kreiranog takmičara ili da kreira novog, pri čemu unosi Ime i Email. Nakon unošenja potrebnog broja takmičara za odigravanje turnira, sistem će dozvoliti organizatoru da kreira žreb, nasumičnim izborom protivnika. Organizator može da resetuje žreb u bilo kom trenutku. Za svaki par će biti kreiran meč. Nakon odigravanja meča organizator turnira ima mogućnost unosa pobednika samog meča, nakon čega će sistem proslediti pobednika u sledeći krug takmičenja (ako ga ima, u suprotnom će proglasiti pobednika). Takođe, organizator ima pristup funkcionalnostima koje omogućuju upravljanje nagradama. Sistem omogućuje različite opcije pri kreiranju nagrada, automatsko i manuelno. Organizator može da unese željeni broj nagrada i izabere automatsko kreiranje nagrada nakon čega će sistem da kreira nagrade za zadati broj mesta po unapred definisanom algoritmu. Druga opcija za kreiranje nagrada, manuelno kreiranje, omogućuje organizatoru da manuelno upiše nagrade na osnovu iznosa nagrade ili procenta od ukupnog nagradnog fonda.

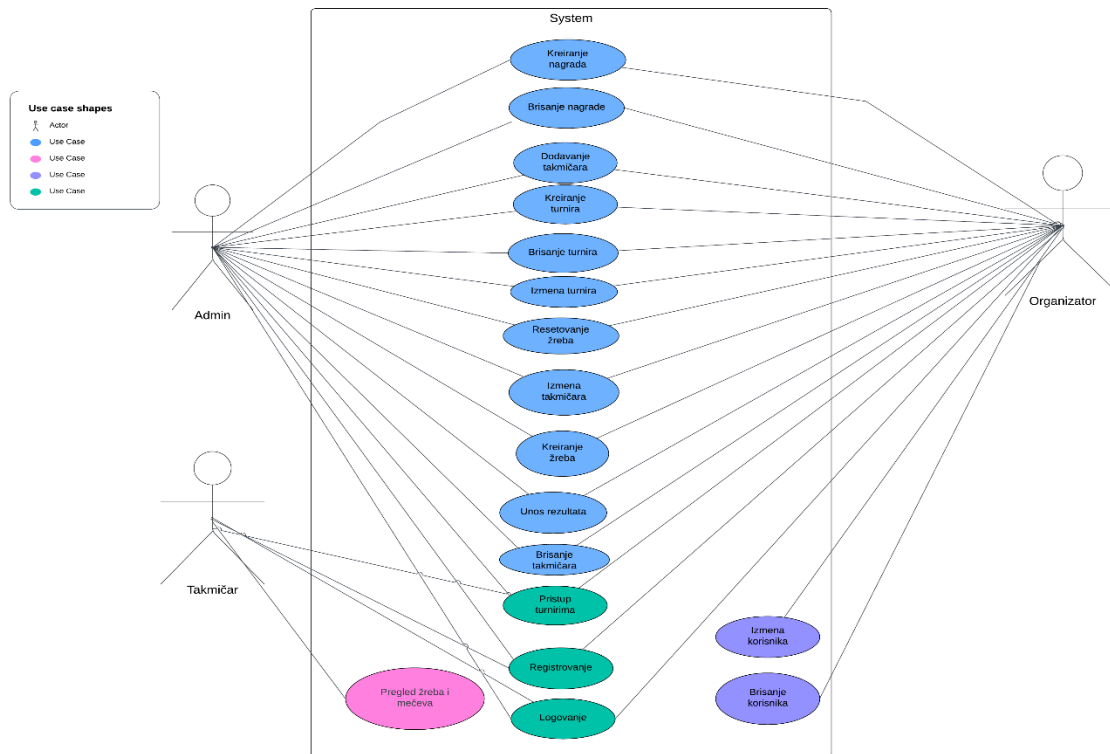
Takmičar, nakon što se uloguje, može da pristupi svim turnirima na kojima učestvuje i ima uvid u žreb, mečeve i nagrade.

Poslednja rola, admin, nakon što se uloguje ima sve mogućnosti kao i rola organizator, sa tim da, admin može da pristupi svim kreiranim turnirima. Takođe, admin ima mogućnost upravljanja svim korisnicima. On može da obriše korisnika, promeni korisnički email ili rolu.

### **5.2 SLUČAJEVI KORIŠĆENJA**

Na osnovu verbalnog modela uočeni su sledeći slučajevi korišćenja:

1. Registrovanje korisnika
2. Logovanje korisnika
3. Pristup turnirima
4. Kreiranje novog turnira
5. Brisanje turnira
6. Izmena turnira
7. Dodavanje takmičara
8. Brisanje takmičara
9. Izmena takmičara
10. Kreiranje žreba
11. Resetovanje žreba
12. Unos rezultata
13. Pregled žreba i mečeva
14. Brisanje korisnika
15. Izmena korisnika
16. Kreiranje nagrade
17. Brisanje nagrade



**Slika 5.2.1 Use-case dijagram svih slučajeva korišćenja**

## 5.1.OPIS SLUČAJEVA KORIŠĆENJA

### SK1: Registrovanje korisnika

Naziv: Registrovanje korisnika

Aktor: Korisnici

Učesnici: Korisnici i sistem

Preduslovi: Sistem je uključen i prikazana je stranica za registraciju

Osnovni scenario:

1. Korisnik unosi podatke za registraciju (APUSO)
2. Korskrik poziva sistem da registruje korisnika(APSO)
3. Sistem proverava da li su uneti podaci validni(SO)
4. Sistem registruje korisnika na sistem i prebacuje ga na prozor za logovanje(SO)

Aleternativni scenario:

3.1 Sistem nakon neuspješne validacije podataka izbacuje grešku sa porukom zašto je došlo do neuspješne registracije(IA)

### **SK2: Logovanje korisnika**

Naziv: Logovanje korisnika

Aktor: Korisnici

Učesnici: Korisnici i sistem

Preduslovi: Sistem je uključen i prikazana je stranica za logovanje

Osnovni scenario:

5. Korisnik unosi podatke za logovanje (APUSO)
6. Korisnik poziva sistem da ga uloguje na aplikaciju (APSO)
7. Sistem loguje korisnika i prebacuje ga na početni prozor aplikacije (SO)

Alternativni scenario:

3.1 Sistem ne može da pronađe korisnika u bazi podataka i uloguje ga. Sistem prikazuje poruku o tome (IA)

### **SK3: Selektovanje turnira**

Naziv: Selektovanje turnira

Aktor: Korisnici

Učesnici: Aktor i sistem

Preduslovi: Sistem je uključen, aktor je ulogovan i sistem prikazuje dropdown sa turnirima

Osnovni scenario:

1. Aktor selektuje željeni turnir (APUSO)
2. Aktor poziva sistem da pristupi selektovanom turniru (APSO)
3. Sistem otvara prozor selektovanog turnira (SO)

Alternativna scenaria:

3.1 Sistem ne može da pristupi selektovanom turniru i prikazuje grešku(IA)



**SK4: Kreiranje novog turnira**

Naziv: Kreiranje turnira

Aktor: Organizator ili admin

Učesnici: Aktor i sistem

Preduslovi: Sistem je uključen, aktor je ulogovan sa rolom organizator ili admin i prikazana je forma za kreiranje turnira

Osnovni scenario:

1. Aktor unosi potrebne informacije o turniru (APUSO)
2. Aktor poziva sistem da kreira novi turnir (APSO)
3. Sistem izvršava kreiranje novog turnira i otvara novi prozor za kreirani turnir(SO)

**SK5: Brisanje turnira**

Naziv: Brisanje turnira

Aktor: Organizator ili admin

Učesnici: Aktor i sistem

Preduslovi: Sistem je uključen, aktor je ulogovan kao organizator ili admin i sistem prikazuje odgovarajuće kreirane turnire iz baze podataka

Osnovni scenario:

1. Aktor bira željeni turnir(APUSO)
2. Akor poziva sistem da obriše turnir (APSO)
3. Sistem briše turnir i prikazuje poruku o uspešno obrisanom turniru(IA)

Alternativna scenarija:

- 3.1 Sistem ne uspeva da pronađe selektovani turnir u bazi podataka i prikazuje grešku(IA)

**SK6: Izmena turnira**

Naziv: Izmena turnira

Aktor: Organizator ili admin

Učesnici: Aktor i sistem

Preduslovi: Sistem je uključen, aktor je ulogovan kao organizator ili admin i sistem prikazuje prozor za odgovarajući turnir

Osnovni scenario:

1. Aktor poziva sistem da prikaže formu za izmenu turnira (APSO)
2. Sistem prikazuje formu (SO)
3. Aktor unosi odgovarajuće izmene (APUSO)
4. Aktor poziva sistem da sačuva izmene (APSO)
5. Sistem čuva ismene i vraća aktora na prozor turnira (SO)

Alternativna scenario:

5.1 Sistem ne može da izmeni turnir i prikazuje poruku o grešci (IA)

### **SK7: Dodavanje takmičara**

Naziv: Dodavanje takmičara

Aktor: Organizator ili admin

Učesnici: Aktor i sistem

Preduslovi: Sistem je uključen, aktor je ulogovan kao organizator ili admin i sistem prikazuje prozor odgovarajućeg turnira

Osnovni scenario:

1. Aktor poziva sistem da doda takmičara (APSO)
2. Sistem otvara novi prozor za dodavanje takmičara (SO)
3. Aktor unosi potrebne podatke za unos takmičara (APUSO)
4. Aktor poziva sistem da doda takmičara (APSO)
5. Sistem dodaje novog takmičara i vraća aktora na prethodni prozor (SO)

Alternativna scenarija:

- 3.1 Aktor selektuje već kreiranog takmičara iz baze podataka (APUSO)
- 4.1 Aktor poziva sistem da doda selektovanog takmičara (APSO)

### **SK8: Brisanje takmičara**

Naziv: Brisanje takmičara

Aktor: Organizator ili admin

Učesnici: Aktor i sistem

Preduslovi: Sistem je uključen, aktor je ulogovan kao organizator ili admin i sistem prikazuje prozor odgovarajućeg turnira i već dodate takmičare

Osnovni scenario:

1. Aktor selektuje takmičara kog želi da obriše(APUSO)
2. Akor poziva sistem da obriše takmičara (APSO)
3. Sistem briše selektovanog takmičara(SO)

Alternativna scenarija:

- 3.1 Sistem prijavljuje grešku da nije selektovan ni jedan takmičar(IA)

### **SK9: Izmena takmičara**

Naziv: Izmena takmičara

Aktor: Organizator ili admin

Učesnici: Aktor i sistem

Preduslovi: Sistem je uključen, aktor je ulogovan kao organizator ili admin i sistem I korisnik je odabrao takmičara kog želi da obrise

Osnovni scenario:

1. Aktor selektuje takmičara kog želi da izmeni(APUSO)
2. Akor poziva sistem da izmeni takmičara (APSO)
3. Sistem prikazuje novi prozor za izmenu selektovanog takmičara(SO)
4. Aktor unosi ismene takmičara(APUSO)
5. Aktor poziva sistem da izvrši unete ismene(APSO)
6. Aktor izvršava ismene I vraća korisnika na prethodni prozor(SO)

### **SK10: Kreiranje žreba**

Naziv: Kreiranje žreba

Aktor: Organizator ili admin

Učesnici: Aktor i sistem

Preduslovi: Sistem je uključen, aktor je ulogovan kao organizator ili admin i sistem prikazuje prozor odgovarajućeg turnira

Osnovni scenario:

1. Aktor poziva sistem da kreira žreb(APSO)
2. Sistem izvršava proveru da li su svi uslovi ispunjeni(SO)
3. Sistem kreira I prikazuje žreb(SO)

Alternativna scenarija:

- 3.1 Sistem prijavljuje grešku da nisu svi uslovi ispunjeni(IA)

#### **SK11: Resetovanje žreba**

Naziv: Resetovanje žreba

Aktor: Organizator ili admin

Učesnici: Aktor I sistem

Preduslovi: Sistem je uključen, aktor je ulogovan kao organizator ili admin. Sistem prikazuje prozor odgovarajućeg turnira I žreb je prethodno kreiran

Osnovni scenario:

1. Korisnik poziva sistem da resetuje žreb (APSO)
2. Sistem izvršava resetovanje žreba i prikazuje novi, prazan žreb (SO)

#### **SK12: Unos rezultata**

Naziv: Unos rezultata

Aktor: Organizator ili admin

Učesnici: Korisnik i sistem

Preduslovi: Sistem je uključen, korisnik je ulogovan kao organizator ili admin. Sistem prikazuje prozor odgovarajućeg turnira i postoji prehodno kreiran žreb

Osnovni scenario:

1. Korisnik poziva sistem da proglasi pobednika odgovarajućeg meča(APSO)
2. Sistem izvršava proglašavanje pobednika meča i šalje ga u sledeći krug takmičenja(SO)

Alternativni scenario:

2.1 Sistem izvršava proglašavanje pobednika meča i proglašava ga pobednik takmičenja jer ne postoji sledeći krug(SO)

**SK13: Pregled žreba i mečeva**

Naziv: Pregled žreba i mečeva

Aktor: Takmičar

Učesnici: Takmičar i sistem

Preduslovi: Sistem je uključen, korisnik sa rolom takmičar je ulogovan i sistem prikazuje prozor odgovarajućeg turnira

Osnovni scenario:

1. Takmičar poziva sistem da prikaze žreb i mečeve turnira(APSO)
2. Sistem prikazuje žreb i mečeve (SO)

Alternativni scenario:

2.1 Sistem prikazuje poruku da žreb i mečevi još uvek nisu određeni(IA)

**SK14: Brisanje korisnika**

Naziv: Brisanje korisnika

Aktor: Admin

Učesnici: Admin i sistem

Preduslovi: Sistem je uključen, korisnik sa rolom admin je ulogovan i sistem prikazuje prozor za upravljanje korisnicima

Osnovni scenario:

1. Admin selektuje korisnika kog želi da obrise(APUSO)
2. Admin poziva sistem da obriše selektovanog korisnika(APSO)
3. Sistem briše selektovanog korisnika(SO)
4. Sistem prikazuje poruku o uspešno obrisanom korisniku(IA)

Alternativni scenario:

3.1 Sistem prikazuje poruku da je došlo do greške pri brisanju selektovanog korisnika(IA)

**SK15: Izmena korisnika**

Naziv: korisnika

Aktor: Admin

Učesnici: Admin i sistem

Preduslovi: Sistem je uključen, korisnik sa rolom admin je ulogovan i sistem prikazuje prozor za upravljanje korisnicima

Osnovni scenario:

1. Admin selektuje korisnika kog želi da izmeni(APUSO)
2. Admin poziva sistem da izmeni selektovanog korisnika(APSO)
3. Sistem prikazuje prozor za imenu korisnika(SO)
4. Admin unosi ismene (APUSO)
5. Admin poziva sistem da sačuva ismene(APUSO)
6. Sistem izvršava ismene(SO)

**SK16: Kreiranje nagrade**

Naziv: Kreiranje nagrade

Aktor: Admin ili organizator

Učesnici: Korisnik I sistem

Preduslovi: Sistem je uključen, korisnik sa rolom admin ili organizator je ulogovan i sistem prikazuje prozor za upravljanje nagradama

Osnovni scenario:

1. Korisnik unosi potrebne informacije za kreiranje nagrade(APUSO)
2. Korisnik poziva sistem da kreira novu nagradu(APSO)
3. Sistem kreira nagradu(SO)

Alternativni scenario:

- 3.1 Sistem prikazuje grešku da nisu unete potrebne informacije jedan korisnik(IA)

**SK15: Brisanje nagrade**

Naziv: Brisanje nagrade

Aktor: Admin ili organizator

Učesnici: Korisnik I sistem

Preduslovi: Sistem je uključen, korisnik sa rolom admin ili organizator je ulogovan i sistem prikazuje prozor za upravljanje nagradama

Osnovni scenario:

1. Korisnik selektuje nagradu koju želi da obiriše(APUSO)
2. Korisnik poziva sistem da obriše selektovanu nagradu(APSO)
6. Sistem briše nagradu(SO)

Alternativni scenario:

- 3.1 Sistem prikazuje grešku da nije selektovan nijedna nagrada(IA)

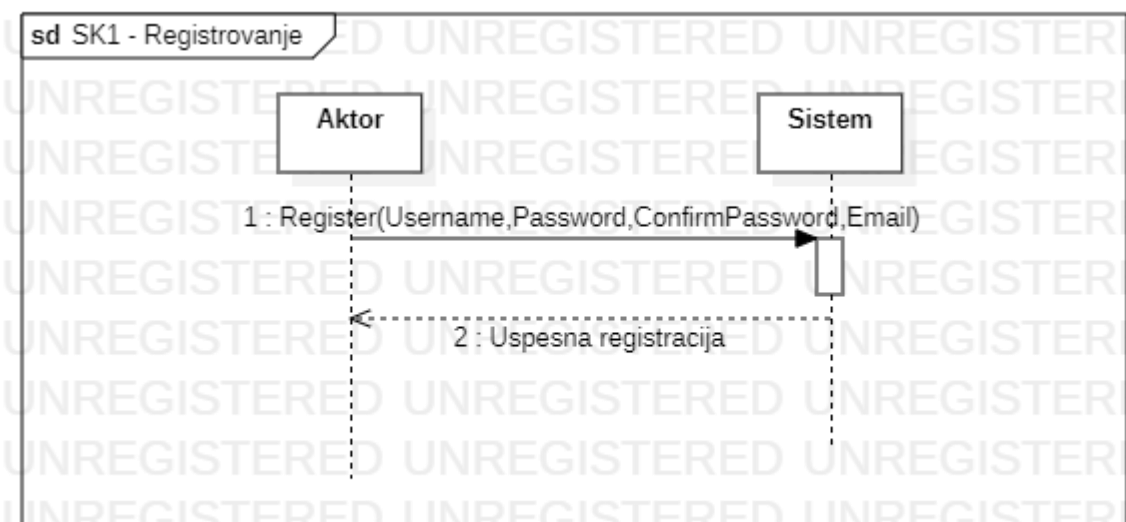
## 6. FAZA ANALIZE

### 6.1.SISTEMSKI DIJAGRAMI SEKVENCI ZA SK

SK1: Registrovanje korisnika

Osnovni scenario:

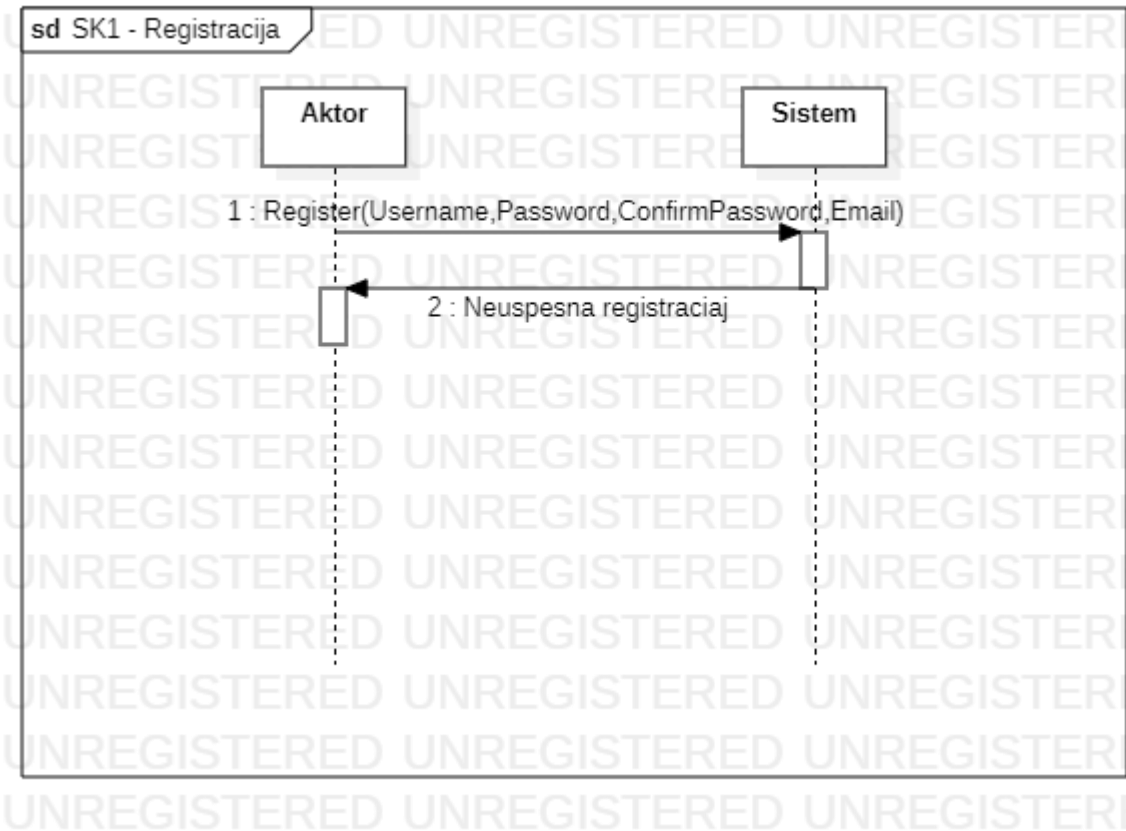
1. Aktor poziva sistem da izvrši regsitrovanje na osnovu unetih podataka (APSO)
2. Sistem regsitruje korisnika i prebacuje ga na prozor za logovanje (SO)



**Slika 6.1.1 DSSK1 - Registracija na aplikaciju**

Alternativna scenarija:

2.1 Sistem ne može da registruje korisnika i prikazuje poruku o tome (IA)



**Slika 6.1.2 DSSK1 - alternativni scenario – Neuspešna registracija**

Uvedene su sistemske operacije:

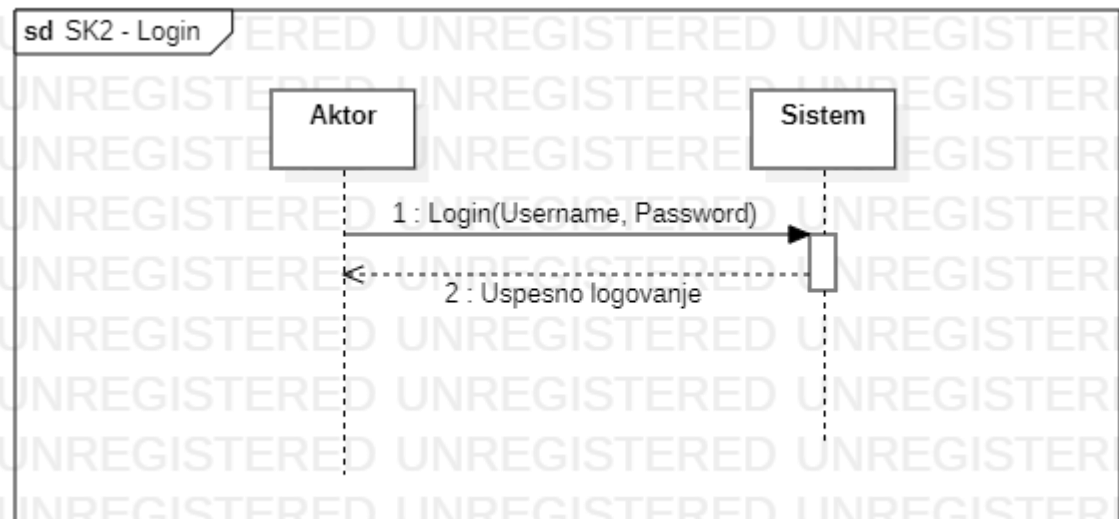
1. Register (Username, Password, ConfirmPassword, Email)

#### **SK2: Logovanje na aplikaciju**

Osnovni scenario:

1. Aktor poziva sistem da uloguje korisnika na osnovu unetih parametara (APSO)
2. Sistem loguje korisnika i prebacuje ga na početni prozor aplikacije (SO)

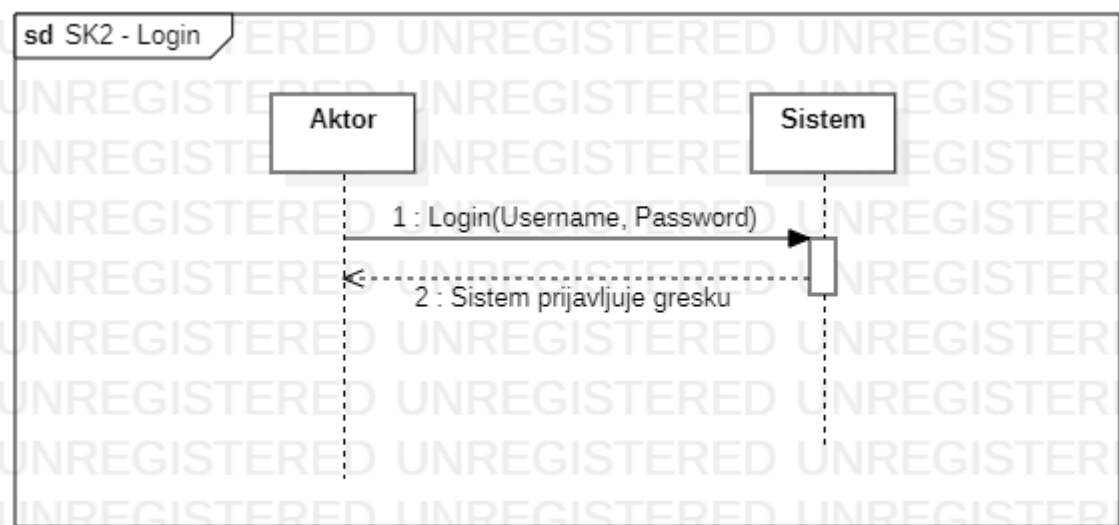




**Slika 6.2.1 DSSK2 – Logovanje korisnika**

Alternativna scenarija:

2.1 Sistem ne može da pronađe korisnika u bazi podataka i uloguje ga. Sistem prikazuje poruku o tome (IA)



**Slika 6.2.2 DDSK2 - alternativni scenario – Sistem ne može da izvrši logovanje korisnika I prikazuje grešku**

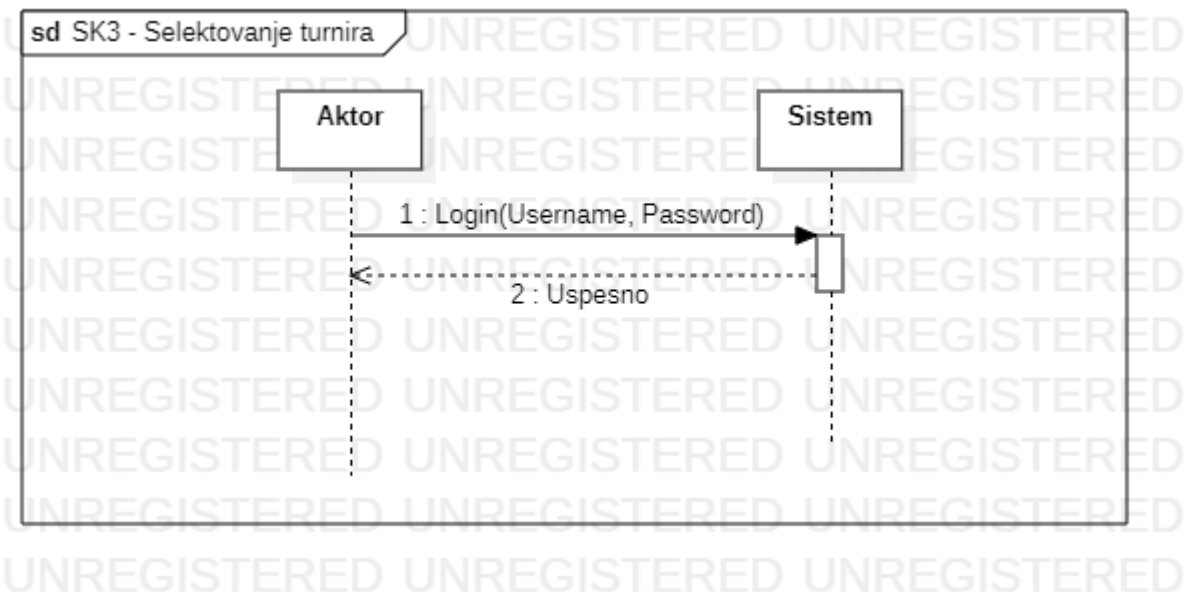
Uvedene su sistemske operacije:

1. Login(Username,Password)

**SK3: Selektovanje turnira**

Osnovni scenario:

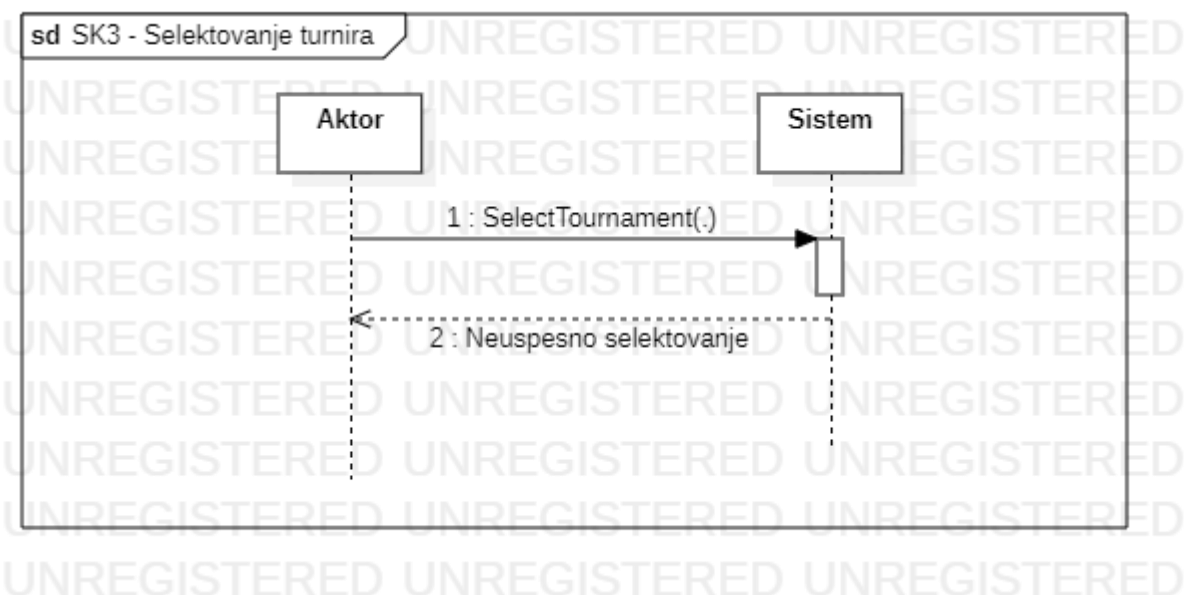
1. Aktor selektuje odgovarajući turnir i poziva sistem da otvori prozor odgovarajućeg turnira (APSO)
2. Sistem otvara novi prozor odgovarajućeg turnira(SO)



**Slika 6.3.1 DDSK3 - Selektovanje turnira**

Alternativna scenarija:

- 2.1 Sistem nije u mogućnosti da pronađe turnir i prikazuje poruku o tome (IA)



**Slika 6.3.2 DDSK3 - alternativni scenario - Sistem nije u mogućnosti da pronađe turnir**

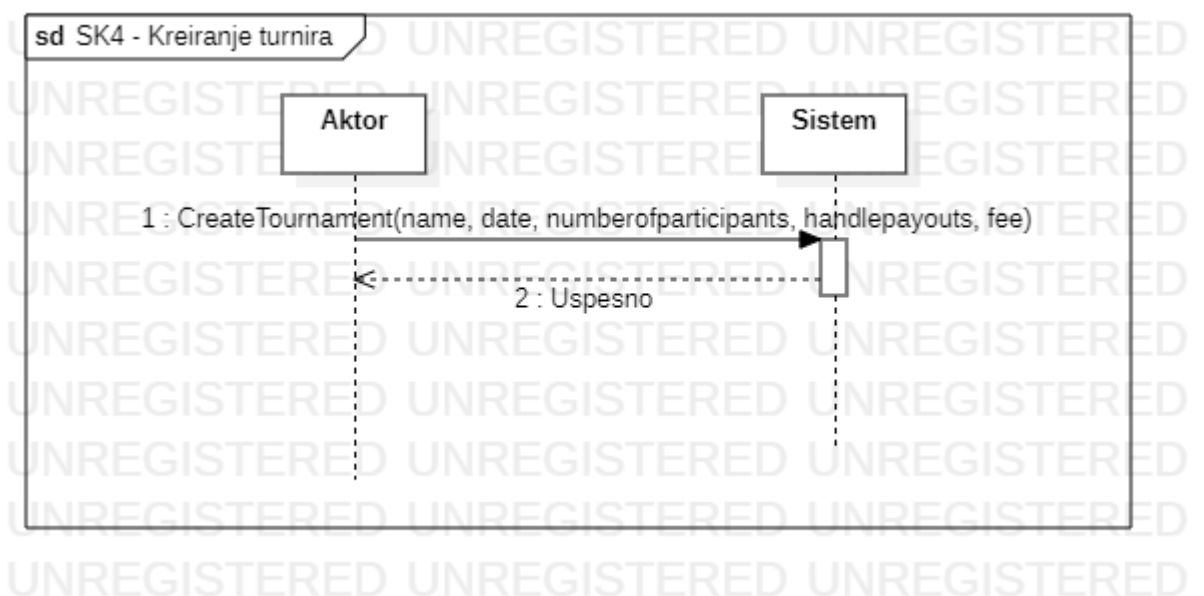
Uvedene su sistemske operacije:

- ### 1. SelectTournament (Id)

## SK4: Kreiranje novog turnira

Osnovni scenario:

1. Aktor poziva sistem da kreira turnir na osnovu unetih paramtera(APSO)
2. Sistem izvršava kreiranje novog turnira i otvara novi prozor za kreirani turnir (SO)



### Slika 6.4.1 DDSK4 – Kreiranje turnira

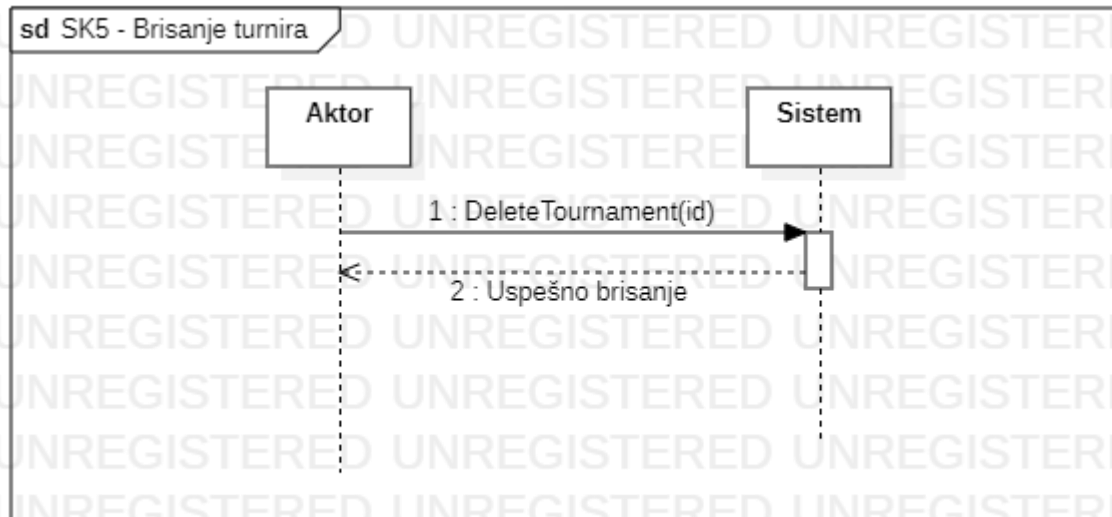
Uvedene su sistemske operacije:

1. CreateTournament (Name, date, NumOfParcitiipants, ManagePayouts, Fee)

### SK5: Brisanje turnira

Osnovni scenario:

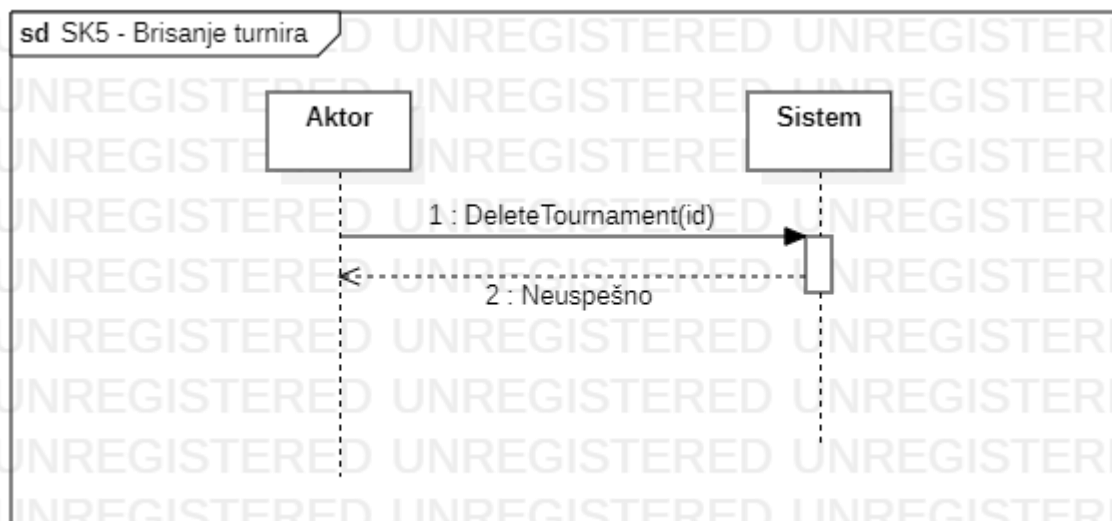
1. Aktor poziva sistem da obriše selektovani turnir(APSO)
2. Sistem uspešno briše turnir(SO)



**Slika 6.5.1 DDSK5 - Brisanje turnira**

Alternativni scenariji:

2.1 Sistem nije u mogućnosti da pronađe selektovani turnir u bazi podataka I prikazuje grešku(IA)



**Slika 6.5.2 DDSK5 -alternativni scenario - Neuspešno brisanje turnira**

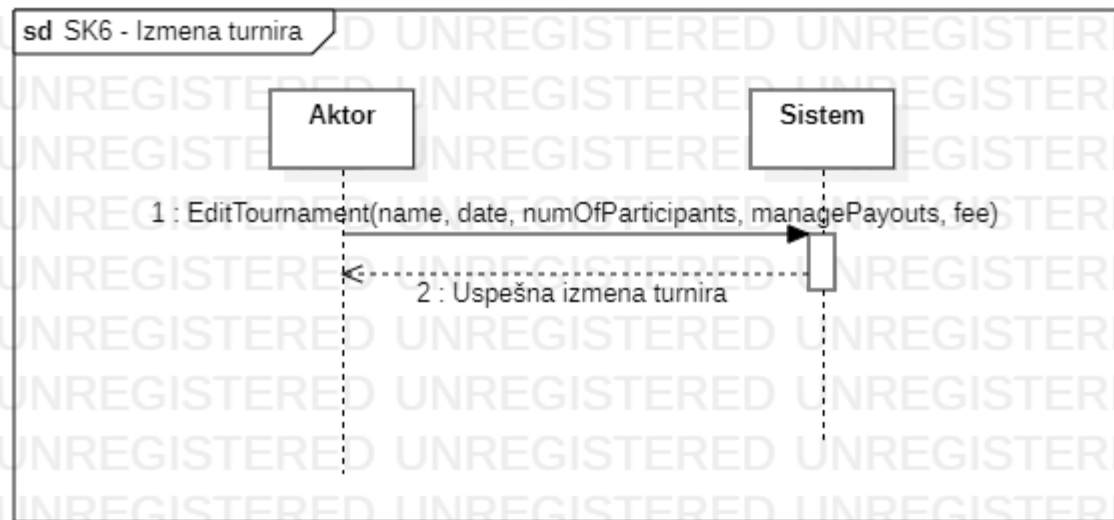
Uvedene su sistemske operacije:

1. DeleteTournament(TournamentID)

#### **SK6: Izmena turnira**

Osnovni scenario:

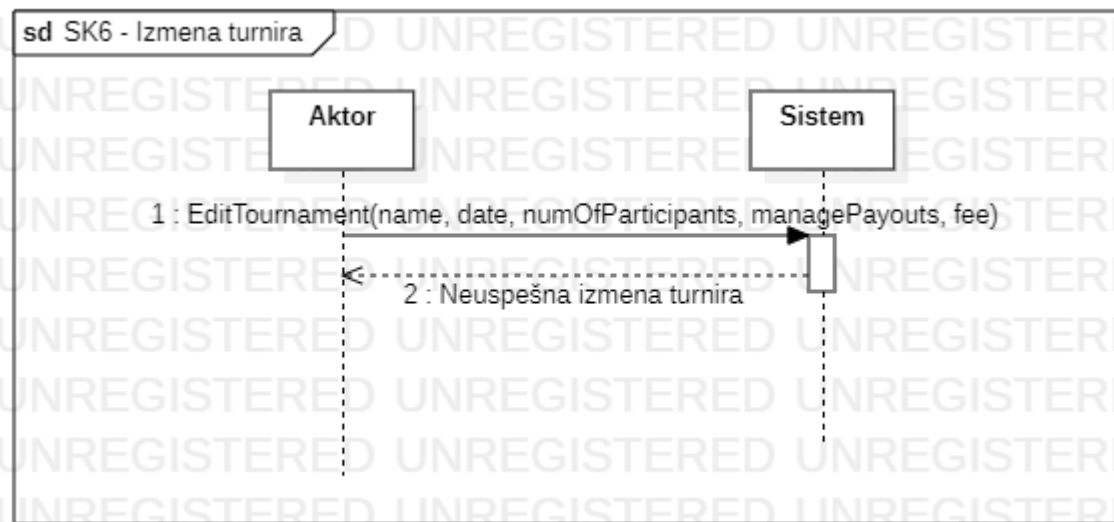
1. Aktor poziva sistem da sačuva izmene za selektovani turnir(APSO)
2. Sistem izvršava izmene za selektovani turnir(SO)



**Slika 6.6.1 DDSK6 – Izmjena turnira**

Alternativni scenariji:

- 2.1 Sistem nije u mogućnosti da izmeni turnir i prikazuje poruku o grešci(IA)



**Slika 6.6.2 DDSK6 -alternativni scenario – Neuspešna izmena turnira**

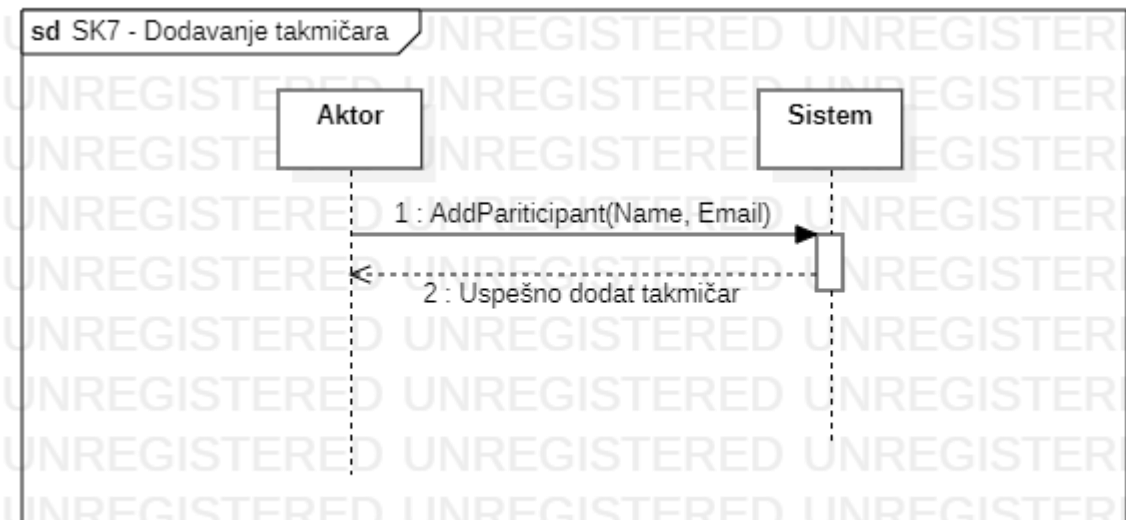
Uvedene su sistemske operacije:

1. EditTournament(Name, Date, NumOfParticipants, ManagePayouts, Fee)

**SK7: Dodavanje takmičara**

Osnovni scenario:

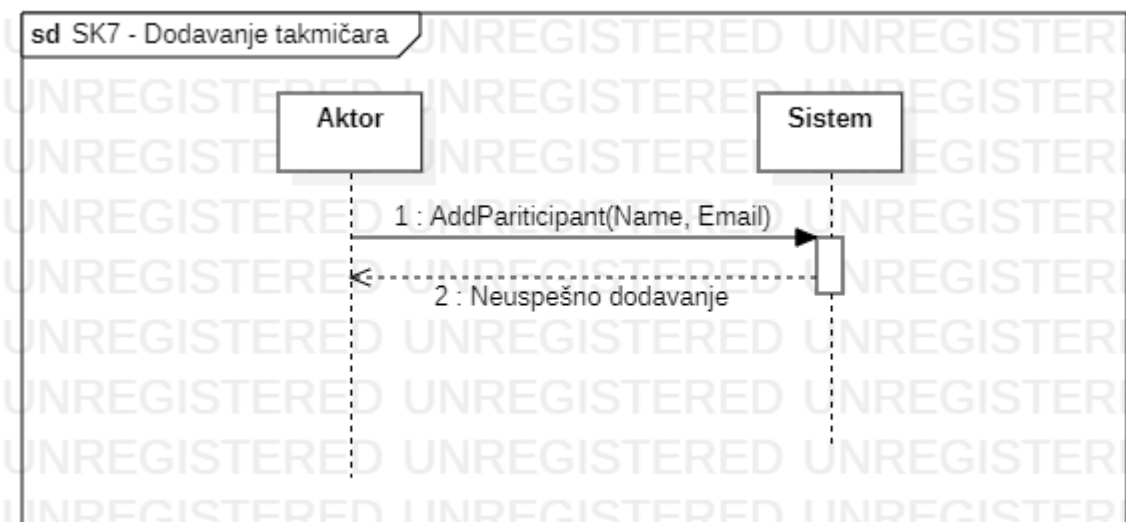
1. Aktor poziva sistem da doda takmičara na osnovu unetih parametara (APSO)
2. Sistem dodaje novog takmičara (SO)



**Slika 6.7.1 DDSK7 - Dodavanje novog takmičara**

Alternativni scenariji:

- 2.1 Sistem nije u mogućnosti da doda novog takmičara i prikazuje poruku o grešci (IA)



**Slika 6.7.2 DDSK7 - alternativni scenario – Neuspešno dodavanje novog takmičara**

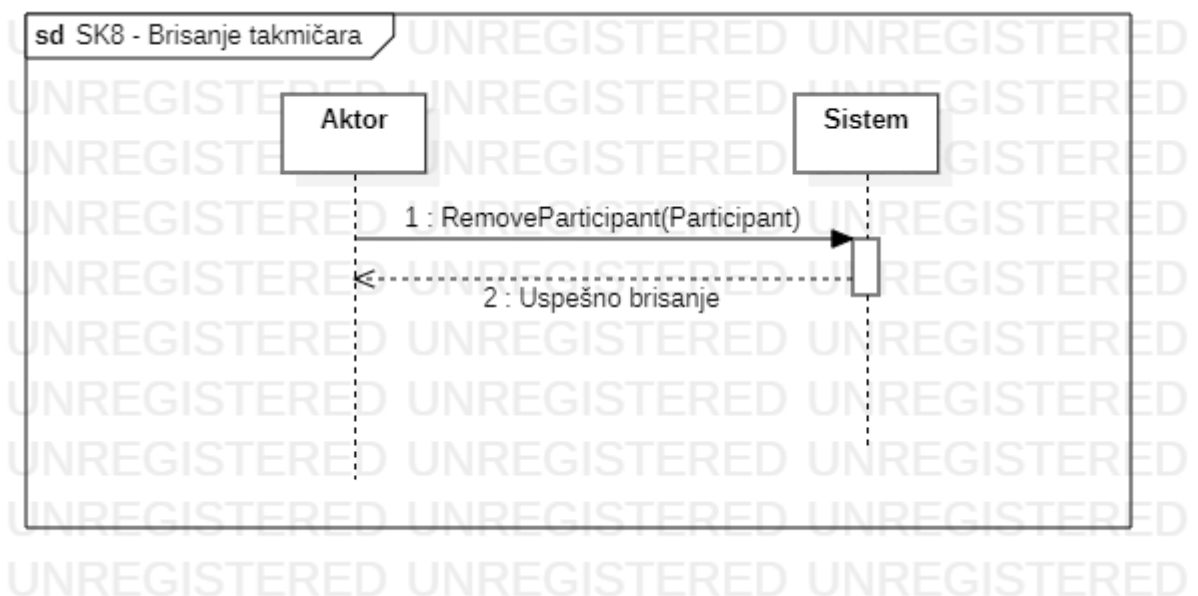
Uvedene su sistemske operacije:

1. AddParticipant(Name, JMBG)

### **SK8: Brisanje takmičara**

Osnovni scenario:

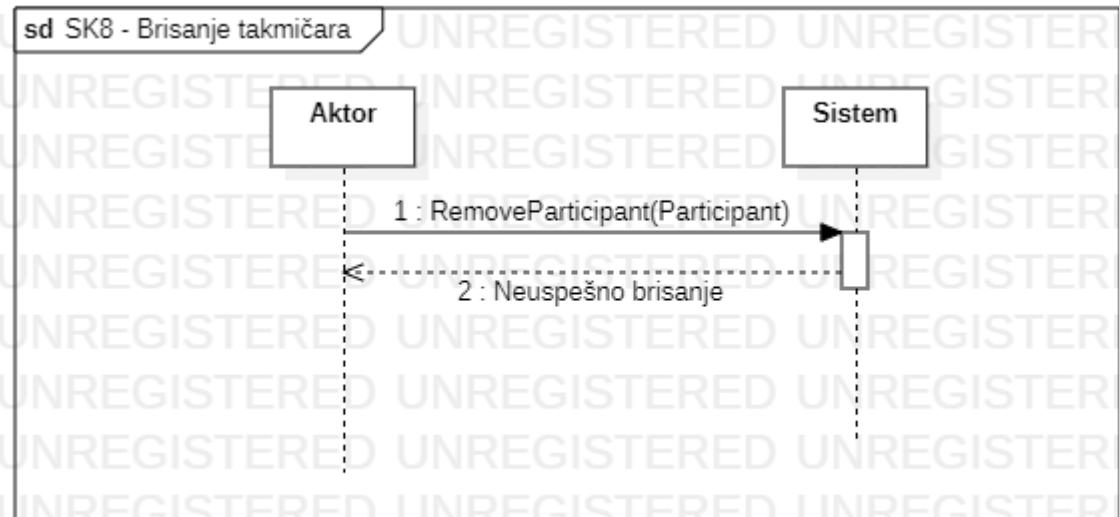
1. Aktor poziva sistem da obriše selektovanog takmičara iz takmičenja (APSO)
2. Sistem briše selektovanog takmičara (SO)



**Slika 6.8.1 DDSK8 – Brisanje takmičara**

Alternativni scenariji:

- 2.1 Sistem prijavljuje grešku da nije selektovan ni jedan takmičar(IA)



**Slika 6.8.1 DDSK9 - alteranativni scenario – Neuspešno brisanje takmičara**

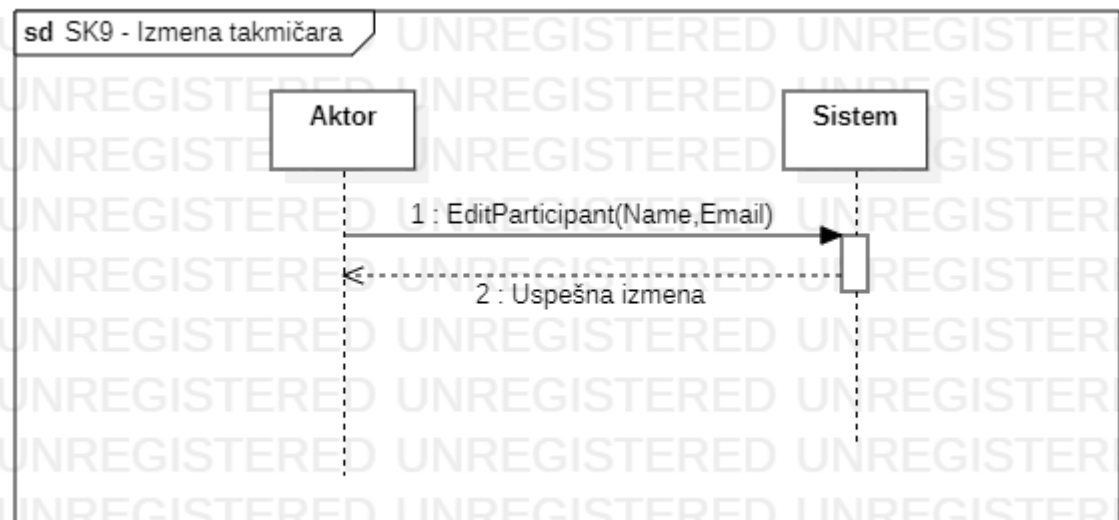
Uvedene su sistemske operacije:

1. RemoveParticipant(ParticipantID)

#### **SK9: Izmena takmičara**

Osnovni scenario:

1. Aktor poziva da sačuva unete izmene za takmičara (APSO)
2. Sistem čuva izmene za takmičara (SO)

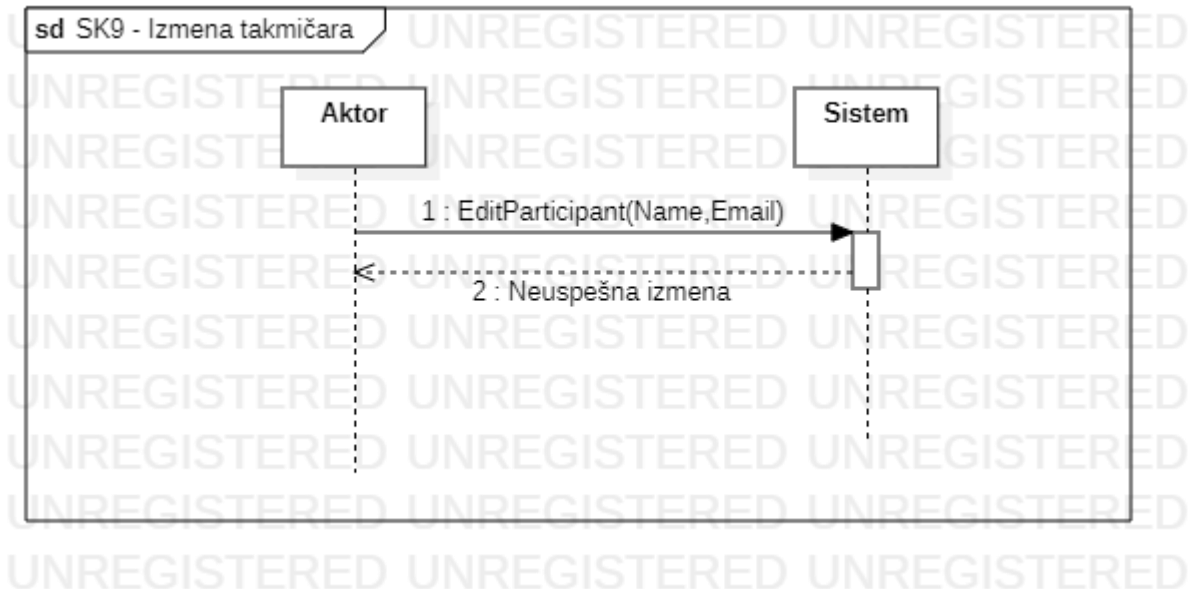


**Slika 6.9.1 DDSK9 – Izmena takmičara**



Alternativni scenariji:

2.1 Sistem prijavljuje grešku da nije moguće izvršiti izmene(IA)



**Slika 6.9.2 DDSK9 - alteranativni scenario – Neuspešna izmena takmičara**

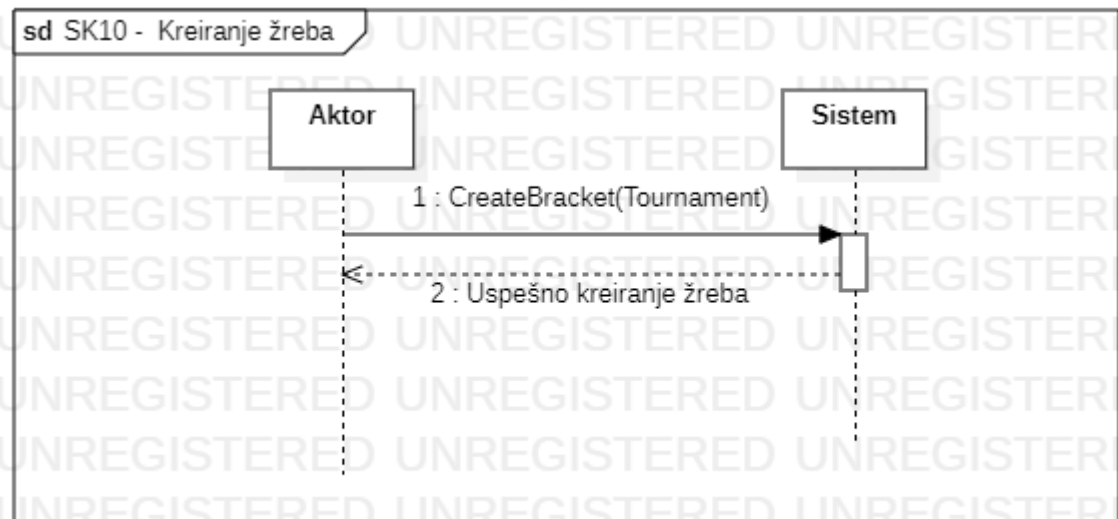
Uvedene su sistemske operacije:

1. EditParticipant(ParticipantID, Name, Email)

#### **SK10: Kreiranje žreba**

Osnovni scenario:

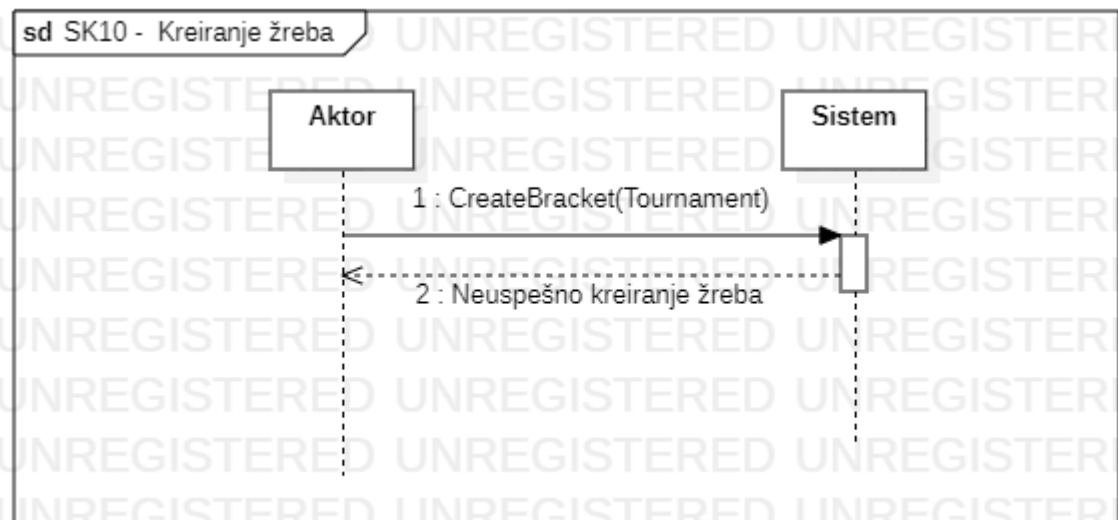
1. Aktor poziva da kreira žreb (APSO)
2. Sistem kreira i prikazuje žreb (SO)



**Slika 6.10.1 DDSK10 – Kreiranje žreba**

Alternativni scenariji:

2.1 Sistem prijavljuje grešku da nisu svi uslovi ispunjeni(IA)



**Slika 6.10.2 DDSK10 - alteranativni scenario – Neuspešno kreiranje žreba**

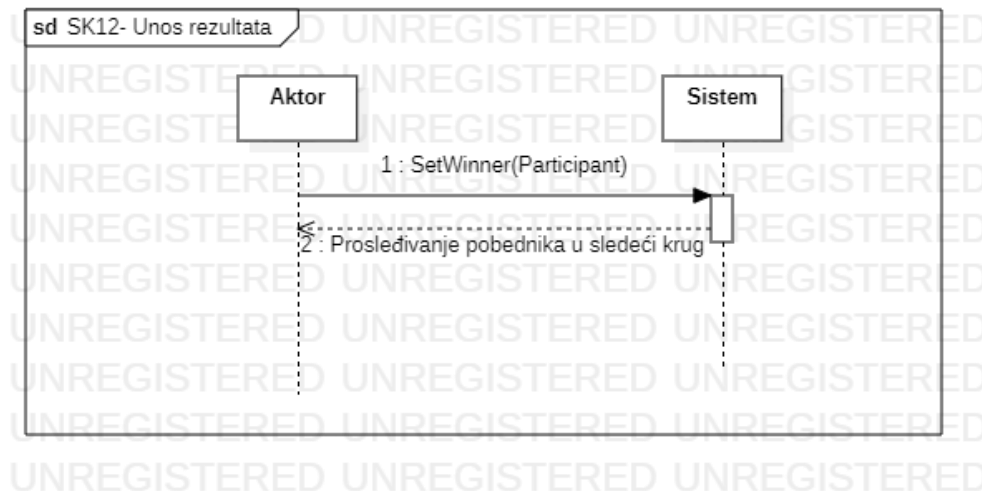
Uvedene su sistemske operacije:

1. CreateBracket(TournamentID)

**SK11: Unos rezultata**

Osnovni scenario:

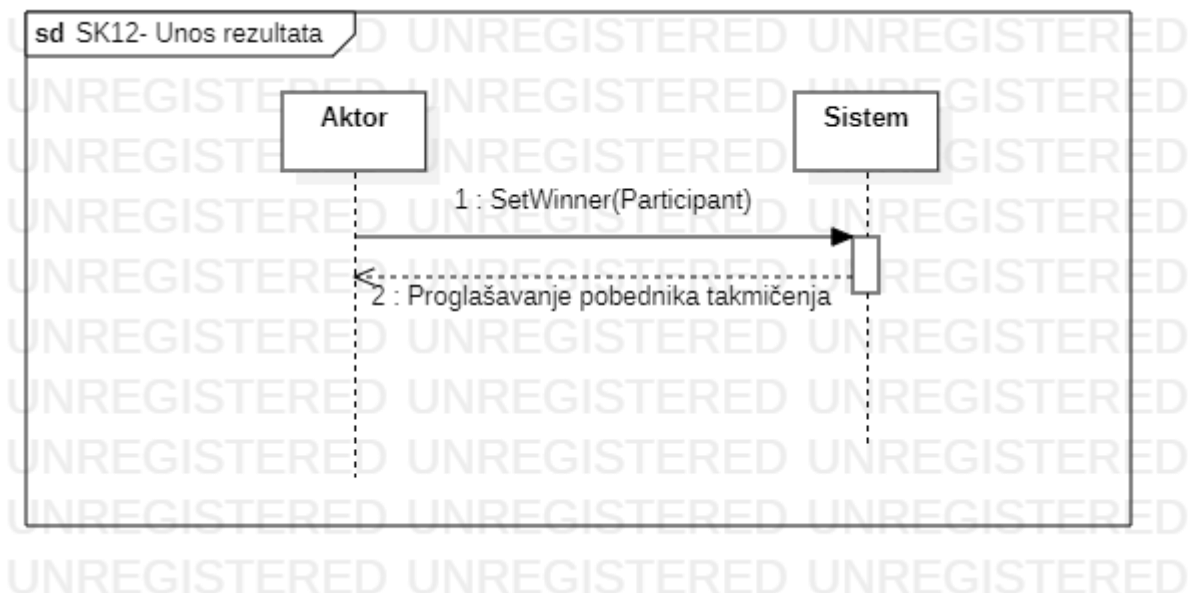
1. Aktor poziva sistem da izvrši proglašenje porednika meča unosom rezultata (APSO)
2. Sistem izvršava proglašavanje porednika meča I prosleđuje ga u sledeći krug takmičenja (SO)



**Slika 6.11.1 DDSK11 – Unos rezultata**

Alternativni scenariji:

- 2.1 Sistem Sistem izvršava proglašavanje porednika meča i proglašava ga porednik takmičenja jer ne postoji sledeći krug(SO)



**Slika 6.11.2 DDSK11 - alteranativni scenario – Unos rezultata(Proglašavanje porednika turnira)**

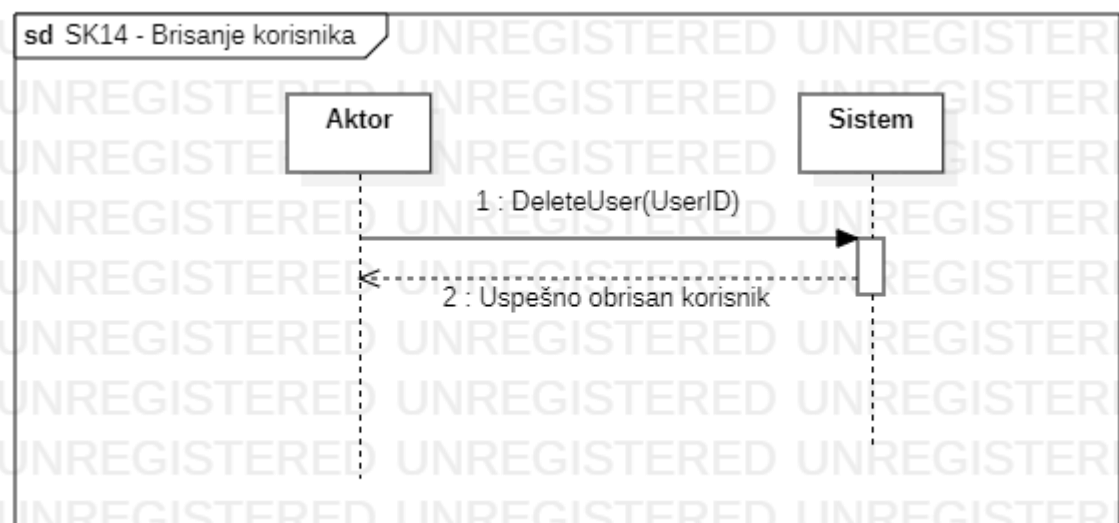
Uvedene su sistemske operacije:

1. SetWinner(Participant)

### **SK12: Brisanje korisnika**

Osnovni scenario:

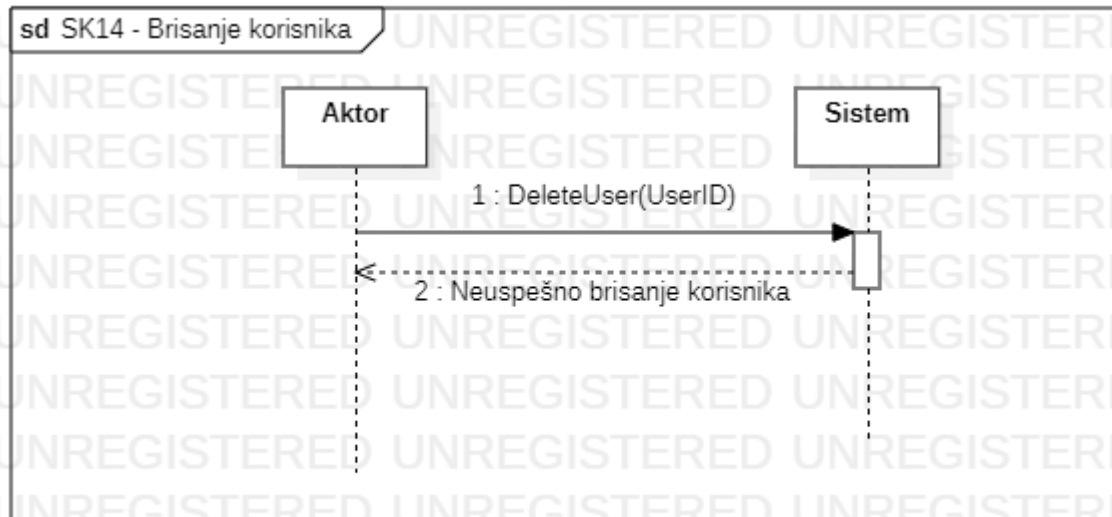
1. Aktor poziva sistem da obriše selektovanog korisnika (APSO)
2. Sistem briše selektovanog korisnika (SO)



**Slika 6.12.1 DDSK12 – Brisanje korisnika**

Alternativni scenariji:

- 2.1 Sistem prijavljuje grešku da nije moguće izvršiti brisanje korisnika (IA)



**Slika 6.12.2 DDSK12 - alteranativni scenario – Neuspešna izmena takmičara**

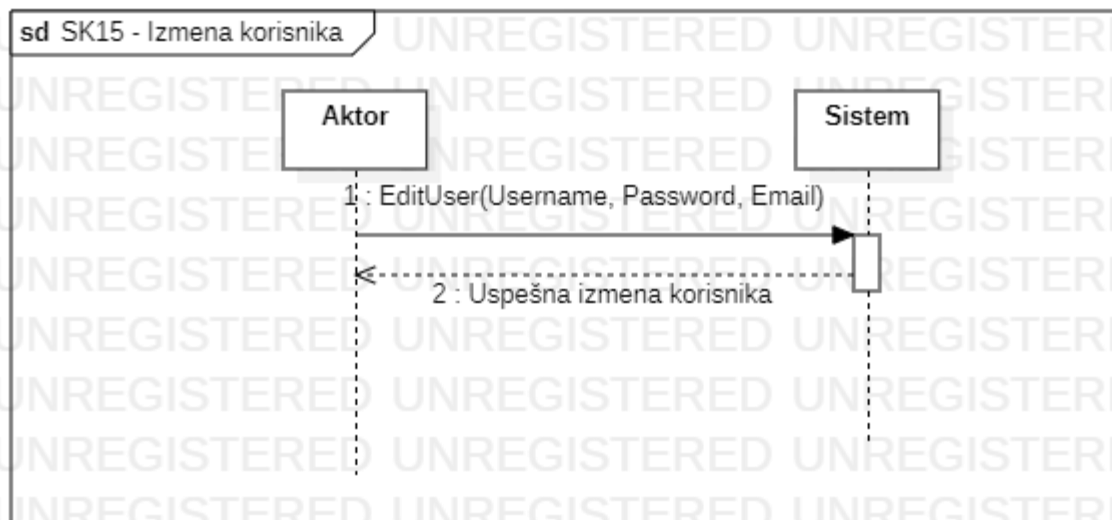
Uvedene su sistemske operacije:

1. DeleteUser(UserID)

#### **SK13: Izmjena korisnika**

Osnovni scenario:

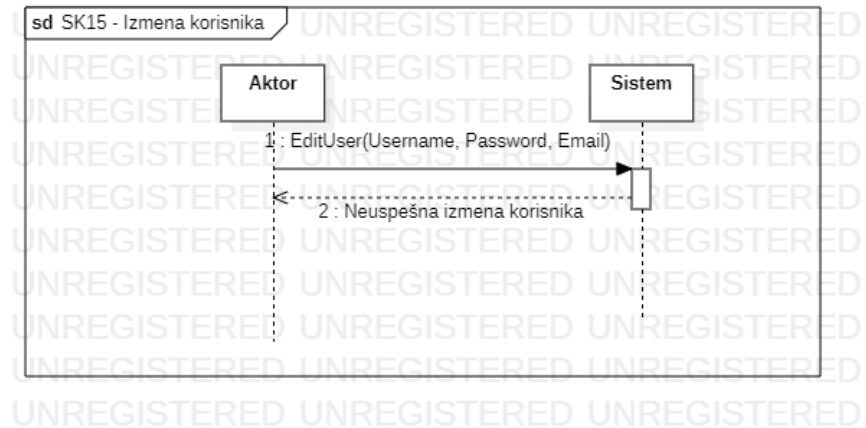
1. Aktor poziva sistem da sačuva izmene za selektovanog korisnika (APSO)
2. Sistem čuva izmene za korisnika (SO)



**Slika 6.13.1 DDSK13 – Izmjena korisnika**

Alternativni scenariji:

#### 2.1 Sistem prijavljuje grešku da nije moguće izvršiti izmene(IA)



**Slika 6.13.2 DDSK13 - alteranativni scenario – Neuspešna izmena korisnika**

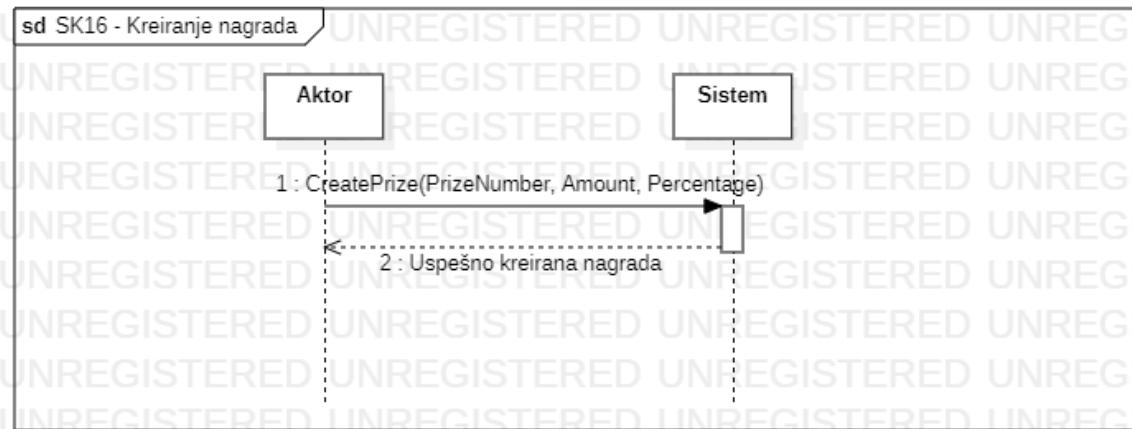
Uvedene su sistemske operacije:

1. EditUser(Email, Role)

#### **SK14: Kreiranje nagrade**

Osnovni scenario:

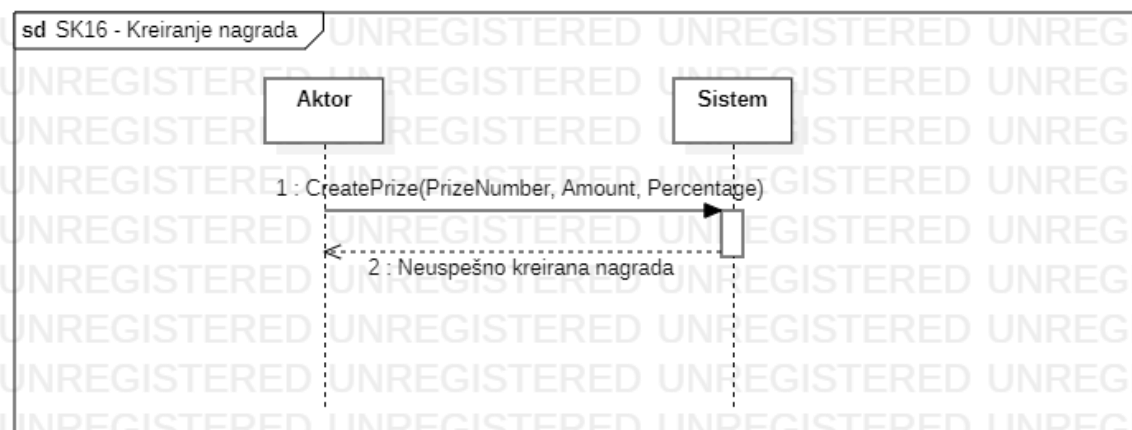
1. Korisnik unosi potrebne informacije za kreiranje nagrade(APUSO)
2. Korisnik poziva sistem da kreira novu nagradu(PSO)
3. Sistem kreira nagradu(SO)



**Slika 6.14.1 DDSK14 – Kreiranje nagrade**

Alternativni scenariji:

3.1 Sistem prijavljuje grešku da nisu unete potrebne informacije(IA)



**Slika 6.14.2 DDSK14 - alteranativni scenario – Neuspešno kreiranje nagrade**

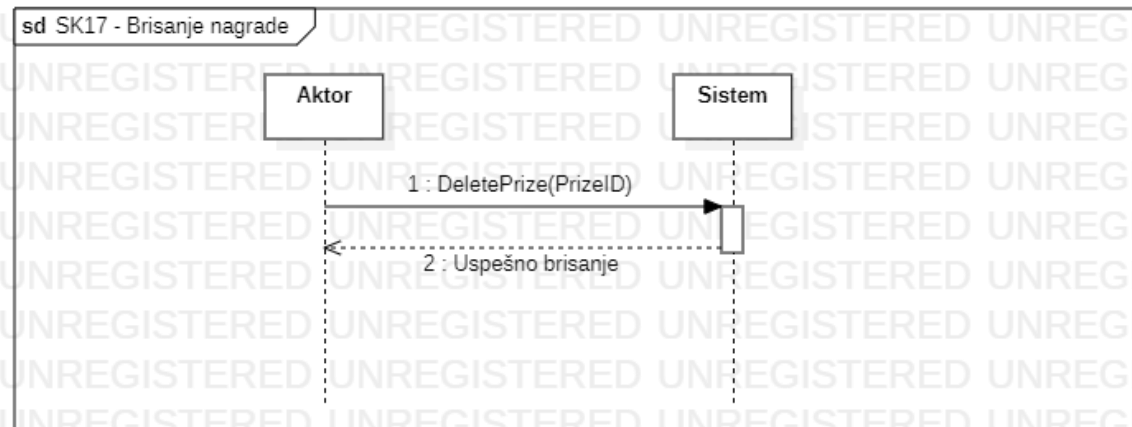
Uvedene su sistemske operacije:

1. CreatePrize(PrizeNumber, Amount, Percentage)

**SK15: Brisanje nagrade**

Osnovni scenario:

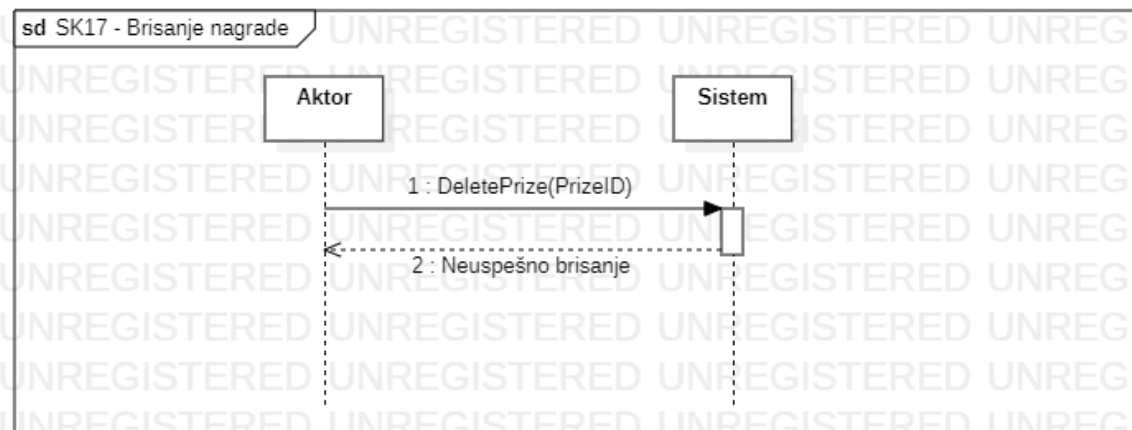
1. Aktor poziva sistem da obriše selektovanu nagradu(APSO)
2. Sistem briše selektovanu nagradu (SO)



**Slika 6.15.1 DDSK15 – Brisanje nagrade**

Alternativni scenariji:

- 3.1 Sistem prijavljuje grešku da nije selektovana ni jedna nagrada(IA)



**Slika 6.15.2 DDSK15 - alteranativni scenario – Neuspešno brisanje nagrade**

Uvedene su sistemske operacije:

1. DeletePrize(PrizeID)



## 6.2 DEFINISANJE UGOVORA O SISTEMSKIM OPERACIJAMA

Za svaku od uočenih sistemskih operacija prave se ugovori (contracts). Ugovori opisuju ponašanje sistemske operacije, tako što opisuje šta operacija radi, ali ne i kako. Jedan ugovor vezan je za jednu sistemsku operaciju.

Uočene su sledeće sistemske operacije koje treba projektovati :

1. Register(Username, Password, ConfirmPassword, Email)
2. Login(Username, Password)
3. SelectTournament(Id)
4. CreateTournament (Name, date, NumOfParcitiipants, ManagePayouts, Fee)
5. DeleteTournament(TournamentID)
6. EditTournament(TournamentID, Name, date, NumOfParcitiipants, ManagePayouts, Fee)
7. AddParticipant(Name, Email)
8. RemoveParticipant(ParticipantID)
9. EditParticipant(ParticipantID, Name, Email)
10. CreateBracket(TournamentID)
11. SetWinner(Participant)
12. DeleteUser(UserID)
13. EditUser(Role, Email)
14. CreatePrize(PrizeNumber, Amount, Percentage)
15. DeletePrize(PrizeID)

### Ugovor UG1: Registracija

Operacija: Register(Username, Password, Confirmpassword, Email)

Veza sa SK: SK1

Preduslovi: Aplikacija je pokrenuta. Korisnik nije ulogovan.

Postuslovi: Novi korisnik je kreiran i podaci su sačuvani u bazi podataka. Korisnik može da se uloguje sa svojim kredencijalima.

**Ugovor UG2: Logovanje**

Operacija: Login(Uusername, Password)

Veza sa SK: SK2

Preduslovi: Aplikacija je pokrenuta. Korisnik nije ulogovan.

Postuslovi: Pročitani su podaci iz baze i korisnik koji se prijavljuje postoji u istoj. Korisnik je uspešno ulogovan.

**Ugovor UG3: Odabir turnira**

Operacija: SelectTournament(Id)

Veza sa SK: SK3

Preduslovi: Aplikacija je pokrenuta. Korisnik je ulogovan.

Postuslovi: Turnir je izabran, podaci o turniru su učitani iz baze i otvoren je novi prozor sa informacijama samog turnira.

**Ugovor UG4: Kreiranje turnira**

Operacija: CreateTournament(Name, Date, NumOfParticipants, ManagePayouts, Fee)

Veza sa SK: SK4

Preduslovi: Aplikacija je pokrenuta. Korisnik je ulogovan kao organizator ili admin.

Postuslovi: Novi turnir je kreiran i podaci sačuvani u bazi podataka.

**Ugovor UG5: Brisanje turnira**

Operacija: DeleteTournament(TournamentID)

Veza sa SK: SK5

Preduslovi: Aplikacija je pokrenuta. Korisnik je ulogovan kao organizator ili admin.

Postuslovi: Turnir je obrisao iz baze podataka

**Ugovor UG6: Izmena turnira**

Operacija: EditTournament(TournamentID, Name, date, NumOfParticipants, ManagePayouts  
Fee)

Veza sa SK: SK6

Preduslovi: Aplikacija je pokrenuta. Korisnik je ulogovan kao organizator ili admin.

Postuslovi: Podaci o turniru su izmenjeni i sačuvani u bazi podataka.

**Ugovor UG7: Dodavanje učesnika**

Operacija: AddParticipant(Name, Email)

Veza sa SK: SK7

Preduslovi: Aplikacija je pokrenuta. Korisnik je logovan kao organizator ili admin.

Postuslovi: Novi učesnik je dodat i podaci su sačuvani u bazi podataka.

**Ugovor UG8: Brisanje Učesnika**

Operacija: RemoveParticiapnt(ParticipantID)

Veza sa SK: SK8

Preduslovi: Aplikacija je pokrenuta. Korisnik je logovan kao organizator ili admin.

Postuslovi: Učesnik je obrisani iz baze podataka.

**Ugovor UG9: Izmena Učesnika**

Operacija: EditiParticiapnt(ParticipantID, Name, Email)

Veza sa SK: SK9

Preduslovi: Aplikacija je pokrenuta. Korisnik je ulogovan kao organizator ili admin.

Postuslovi: Podaci o učesniku su izmenjeni i sačuvani u bazi podataka.

**Ugovor UG10: Kreiranje žreba**

Operacija: CreateBracket(TournamentID)

Veza sa SK: SK10

Preduslovi: Aplikacija je pokrenuta. Korisnik je ulogovan kao organizator ili admin. Ispunjen je predodređeni broj takmičara

Postuslovi: Žreb je kreiran i sačuvan u bazi podataka.

**Ugovor UG11: Proglašavanje pobednika**

Operacija: SetWinner(Participant)

Veza sa SK: SK11

Preduslovi: Aplikacija je pokrenuta. Korisnik je ulogovan kao organizator ili admin. Žreb je kreiran.

Postuslovi: Pobjednik je postavljen i podaci su sačuvani u bazi podataka.

**Ugovor UG12: Brisanje korisnika**

Operacija: DeleteUser(UserID)

Veza sa SK: SK12

Preduslovi: Aplikacija je pokrenuta. Korisnik je ulogovan kao admin.

Postuslovi: Korisnik je obrisani iz baze podataka.

**Ugovor UG13: Izmena korisnika**

Operacija: EditUser(Email, Role)

Veza sa SK: SK13

Preduslovi: Aplikacija je pokrenuta. Korisnik je ulogovan.

Postuslovi: Podaci o učesniku su izmenjeni i sačuvani u bazi podataka.

**Ugovor UG14: Kreiranje nagrade**

Operacija: CreatePrize(PrizeNumber, Amount, Percentage)

Veza sa SK: SK14

Preduslovi: Aplikacija je pokrenuta. Korisnik sa rolom admin ili organizator je ulogovan. Aplikacija prikazuje prozor za upravljanje nagradama.

Postuslovi: Podaci o nagradi su sačuvani u bazi podataka.

**Ugovor UG15: Brisanje nagrade**

Operacija: DeletePrize(PrizeID)

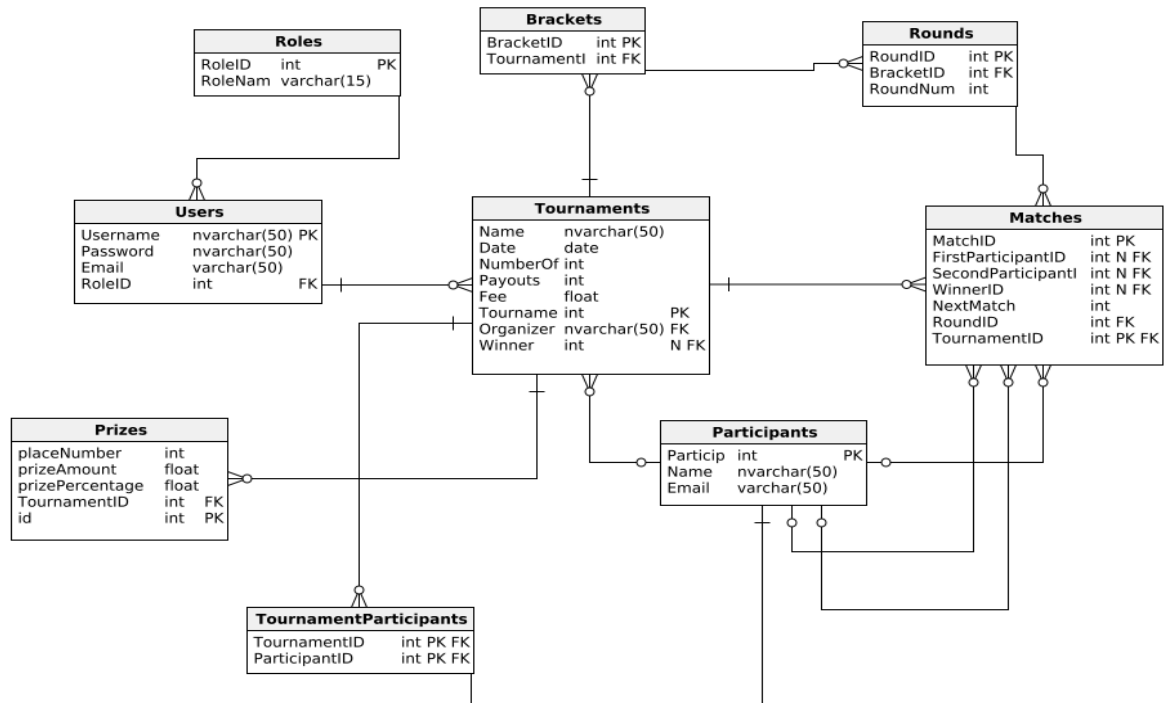
Veza sa SK: SK15

Preduslovi: Aplikacija je pokrenuta. Korisnik sa rolom admin ili organizator je ulogovan. Aplikacija prikazuje prozor za upravljanje nagradama.

Postuslovi: Podaci o nagradi su obrisani iz baze podataka.

## 6.2.KONCEPTUALNI(DOMENSKI) MODEL

Konceptualni model nastaje na osnovu podataka iz funkcionalnog zahteva I slučajeve korišćenja.



Slika 7.3.1. Konceptualni model

## 6.3.RELACIONI MODEL

Na osnovu konceptualnog modela pravi se relacioni model koji predstavlja osnovu za projektovanje relacione baze podataka.

**Roles** (ID, RoleName);

**Users** (Username, Password, Email, RoleID);

**Participants** (ParticipantID, Name, Email);

**Prizes**(ID,PlaceNumber,PrizeAmount,PrizePercentage);

**TournamentParticipants** (TournamentID, ParticipantID)

**Matches** (MatchID, FirstParticipantID, SecondParticipantID, WinnerID, NextMatch, RoundID, TournamentID);

**Rounds** (RoundID, BracketID, RoundNum);

**Brackets** (BracketID, TournamentID);

**Tournaments**(TournamentID,Name,Date,NumberOfParticipants,Payouts,Fee,Organizer,Winner);

## 7. FAZA PROJEKTOVANJA

### 7.1. DIJAGRAMI SEKVENCI ZA SISTEMSKE OPERACIJE

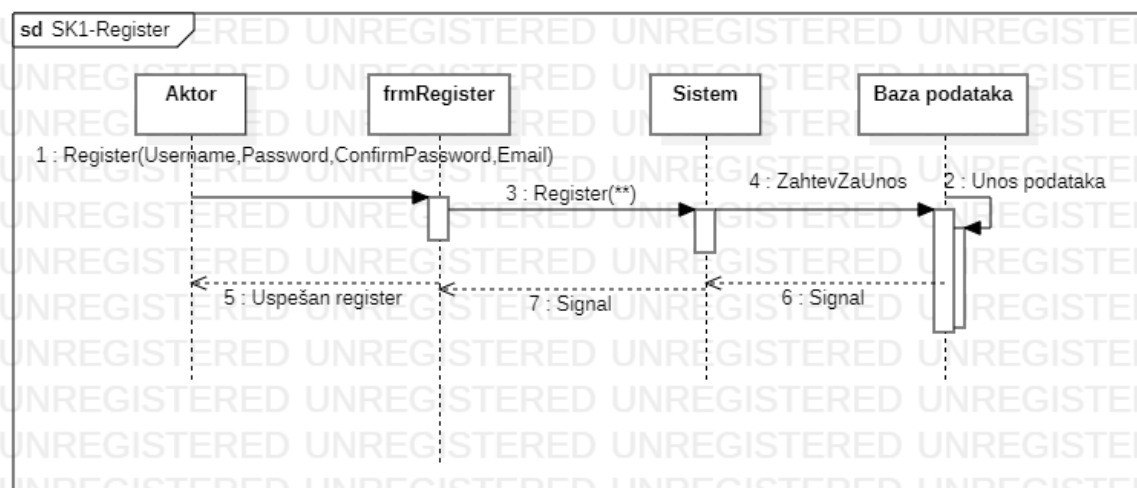
#### Ugovor UG1: Registrovanje

Operacija: Register(Username, Password, ConfirmPassword, Email)

Veza sa SK: SK1

Preduslovi: Aplikacija je pokrenuta

Postuslovi: Uneti podaci o korisniku su sačuvani u bazi podataka



**Slika 7.1.1 Dijagram sekvenci UG1 – Registrovanje**

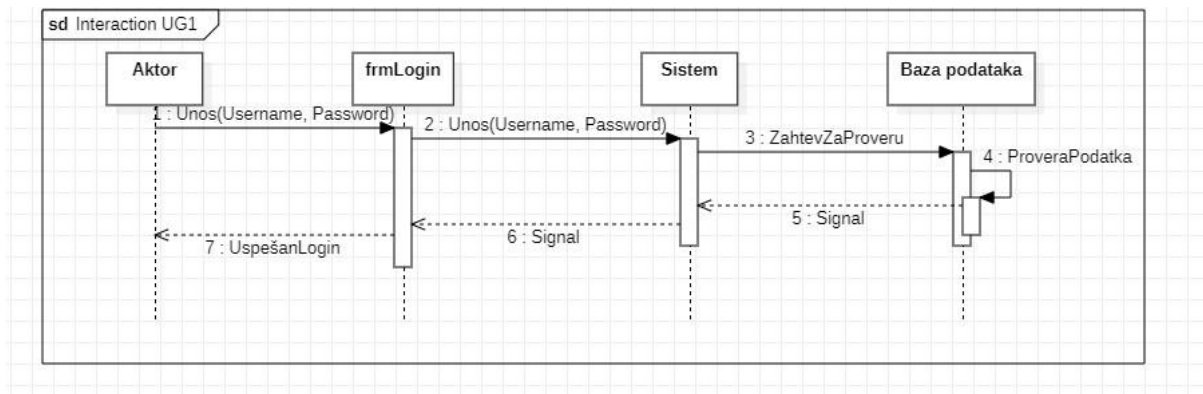
#### Ugovor UG2: Logovanje

Operacija: Login(Username, Password)

Veza sa SK: SK2

Preduslovi: Aplikacija je pokrenuta

Postuslovi: Pročitani su podaci iz baze i korisnik koji se prijavljuje postoji u istoj



**Slika 7.2.1 Dijagram sekvenci UG2 – Logovanje**

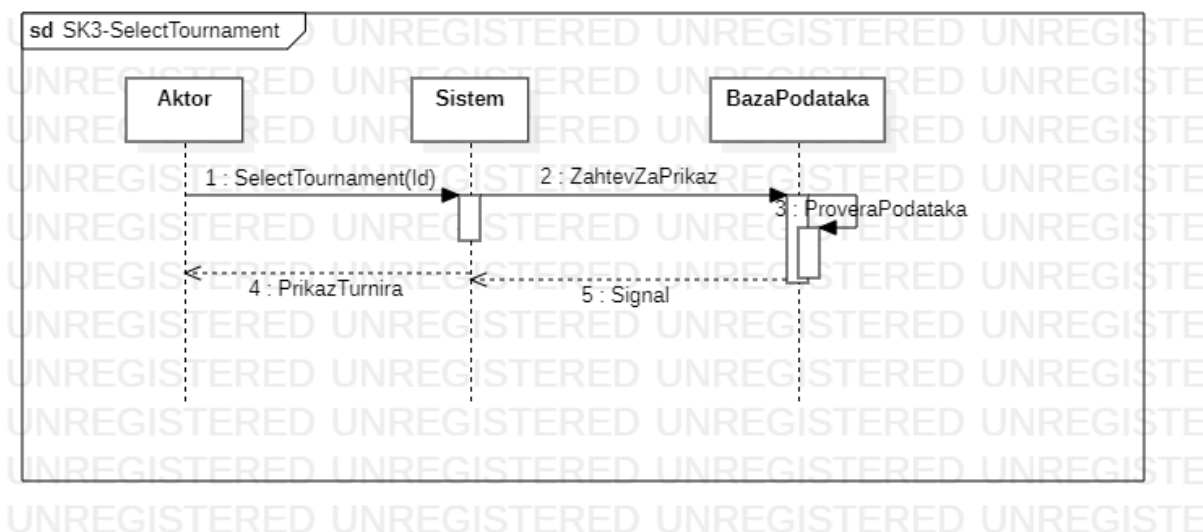
### Ugovor UG3: Selektovanje Turnira

Operacija: SelectTournament(Id)

Veza sa SK: SK3

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan i selektovao je turnir

Postuslovi: Podaci o selektovanom turniru postoje



**Slika 7.3.1 Dijagram sekvenci UG3 – SelectTournament**

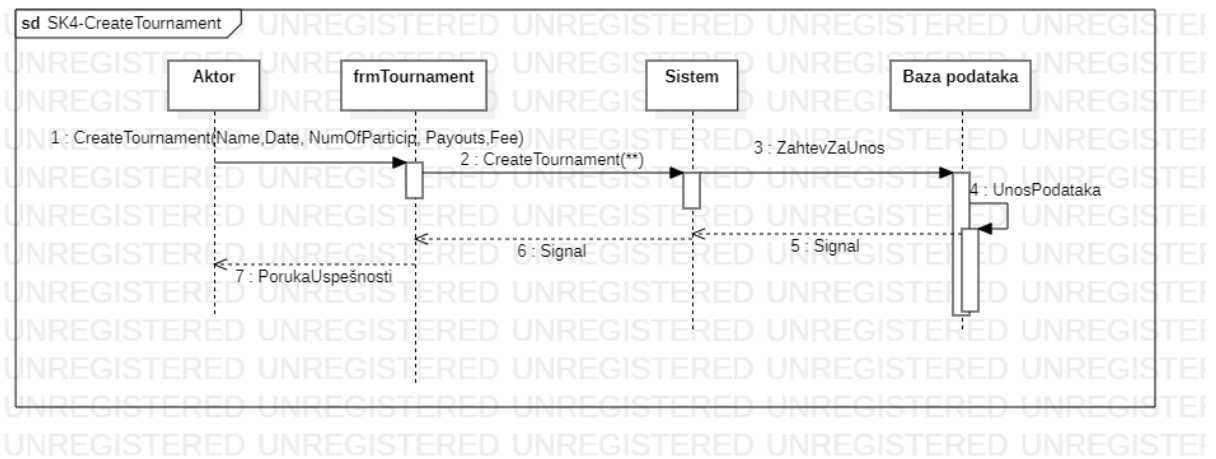
### Ugovor UG4: Kreiranje Turnira

Operacija: CreateTournament (Name, date, NumOfParcitiptants, ManagePayouts, Fee)

Veza sa SK: SK4

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao organizator ili admin, i prikazana je forma za kreiranje turnira

Postuslovi: Podaci o turniru su sačuvani u bazi podataka



**Slika 7.4.1 Dijagram sekvenci UG4 – Kreiranje turnira**

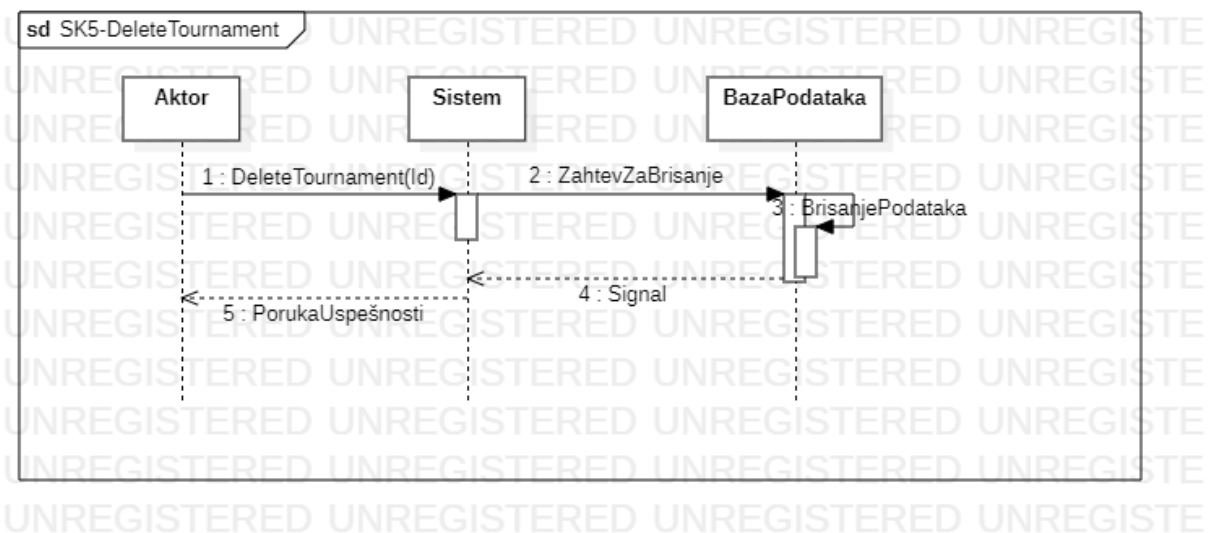
### Ugovor UG5: Brisanje turnira

Operacija: DeleteTournament(TournamentID)

Veza sa SK: SK5

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao admin ili organizator i selektovan je željeni turnir

Postuslovi: Turnir je obrisani iz baze podataka



**Slika 7.5.1 Dijagram sekvenci UG5 – Brisanje turnira**



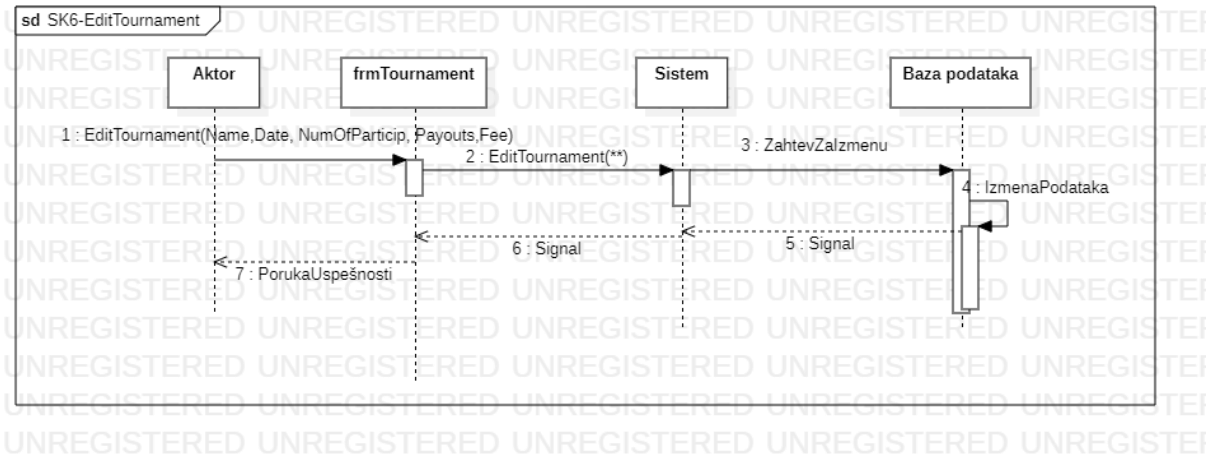
**Ugovor UG6: Izmena turnira**

Operacija: EditTournament(TournamentID,Name,date,NumOfParcicipants,ManagePayouts, Fee)

Veza sa SK: SK6

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao admin ili organizator i prikazana je forma za izmenu turnira

Postuslovi: Podaci o turniru su sačuvani u bazi podataka



**Slika 7.6.1 Dijagram sekvenci UG6 – IzmenaTurnira**

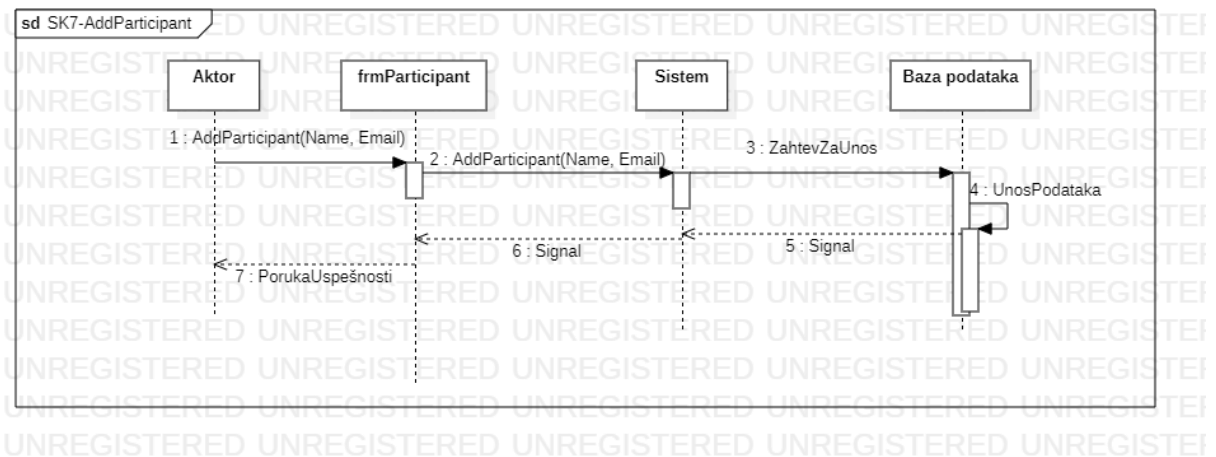
**Ugovor UG7: Dodavanje učesnika**

Operacija: AddParticipant(Name, Email)

Veza sa SK: SK7

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao organizator ili admin i otvorena je forma za unos novog takmičara

Postuslovi: Takmičar je sačuvan u bazi podataka



**Slika 7.7.1 Dijagram sekvenci UG7 – Dodavanje učesnika**

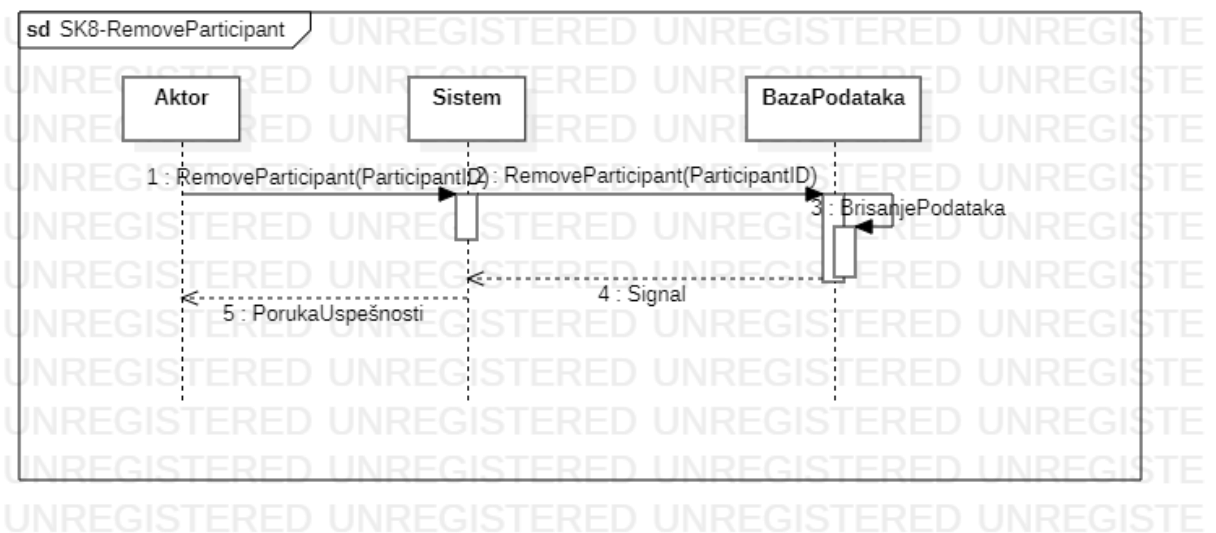
### Ugovor UG8: Uklanjanje učesnika

Operacija: RemoveParticipant(ParticipantID)

Veza sa SK: SK8

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao organizator ili admin I otvoren je prozor za upravljanje učesnicima

Postuslovi: Učesnik je obrisani iz baze podataka



**Slika 7.8.1 Dijagram sekvenci UG8 – Uklanjanje učesnika**

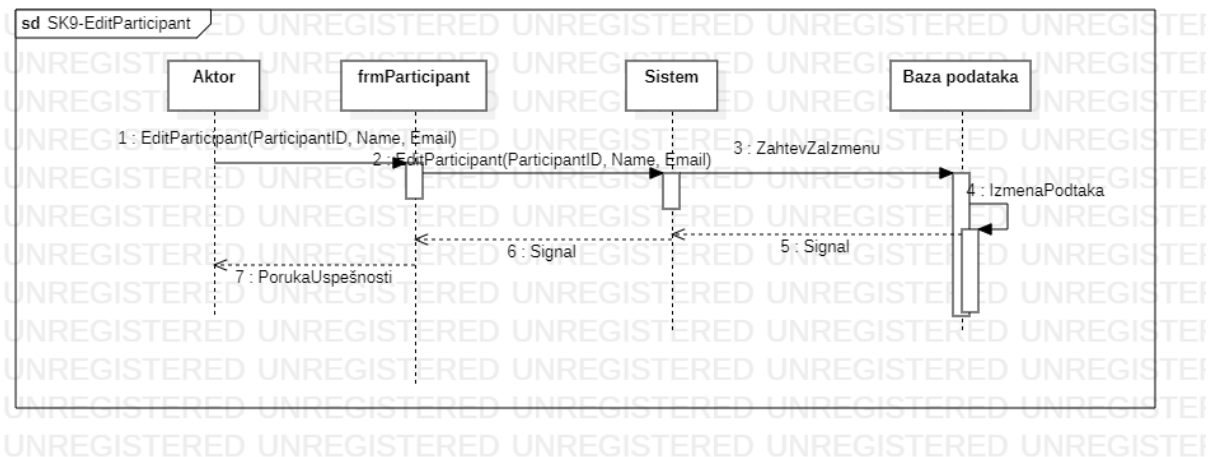
**Ugovor UG9: Izmena učesnika**

Operacija: EditParticipant(ParticipantID, Name, Email)

Veza sa SK: SK9

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao organizator ili admin I otvoren je prozor za upravljanje učesnicima

Postuslovi: Podaci su sačuvani u bazi podataka



**Slika 7.9.1 Dijagram sekvenci UG9 – Izmena takmičara**

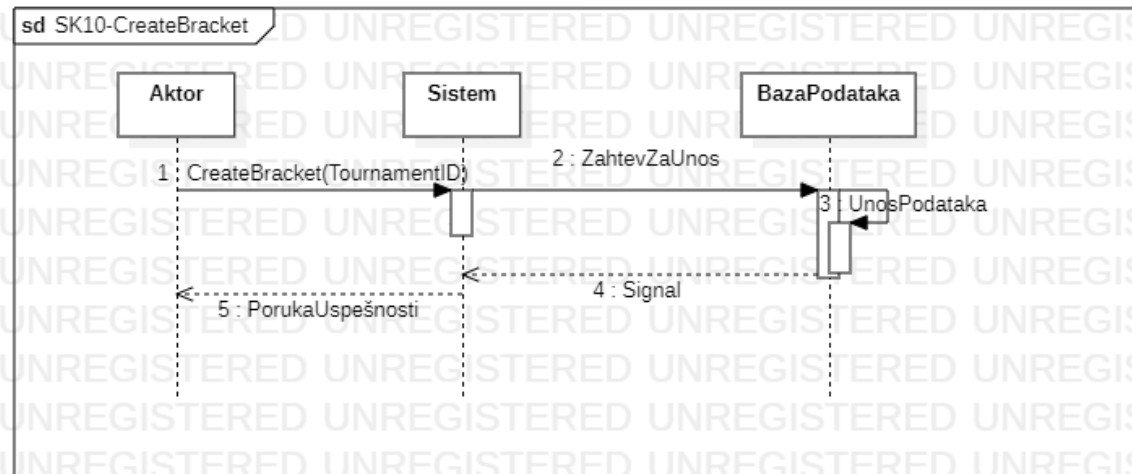
**Ugovor UG10: Kreiranje žreba**

Operacija: CreateBracket(TournamentID)

Veza sa SK: SK10

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao organizator ili admin i ispunjeni su vi uslovi za kreiranje žreba

Postuslovi: Podaci su sačuvani u bazi podataka



**Slika 7.10.1 Dijagram sekvenci UG10 – Kreiranje žreba**

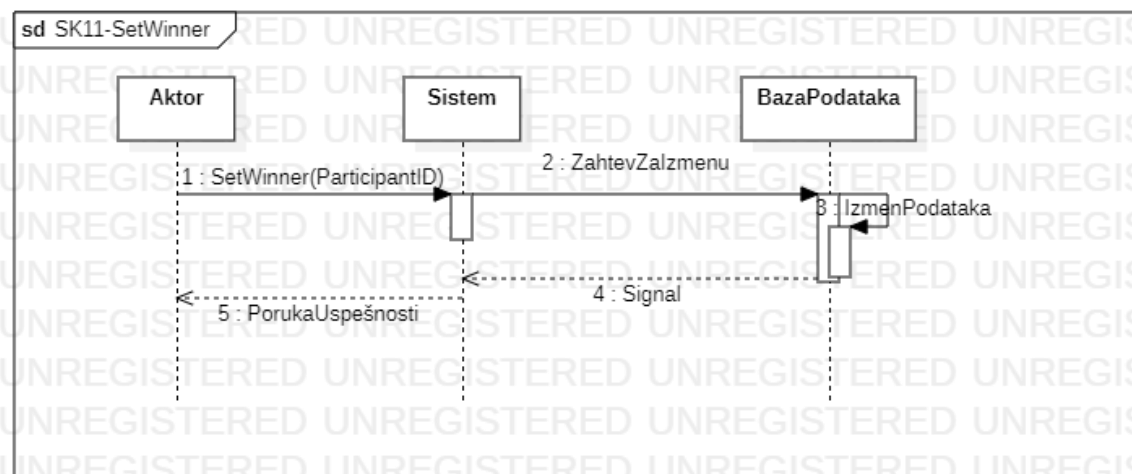
### **Ugovor UG11: Proglašavanje pobednika**

Operacija: SetWinner(ParticipantID)

Veza sa SK: SK11

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao organizator ili admin

Postuslovi: Podaci su sačuvani u bazi podataka



**Slika 7.11.1 Dijagram sekvenci UG11 – Proglašavanje pobednika**

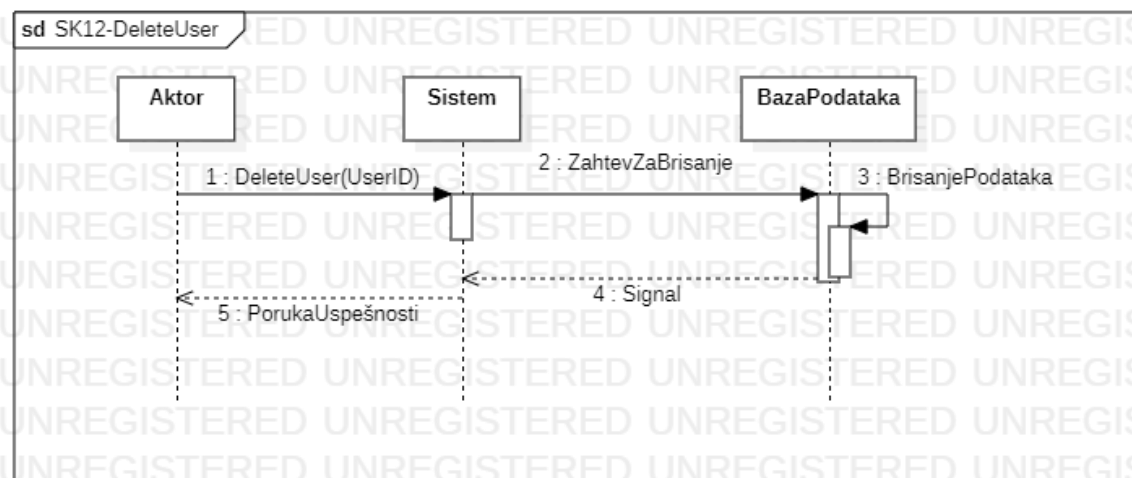
**Ugovor UG12: Brisanje korisnika**

Operacija: DeleteUser(UserID)

Veza sa SK: SK12

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao admin I selektovan je korisnik za brisanje

Postuslovi: Podaci su obrisani iz baze podataka



**Slika 7.12.1 Dijagram sekvenci UG12 – Brisanje korisnika**

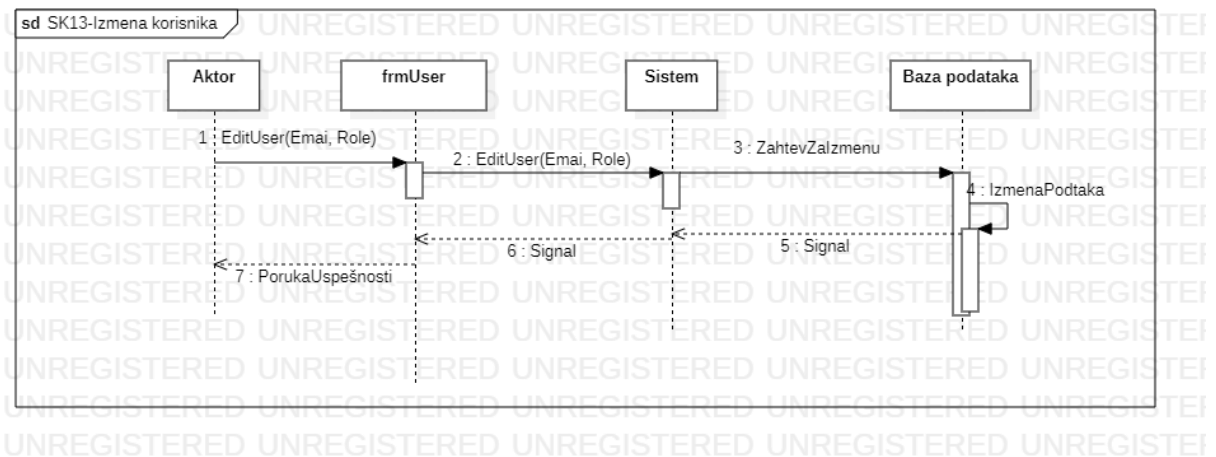
**Ugovor UG13: Izmena korisnika**

Operacija: EditUser(Email, Role)

Veza sa SK: SK13

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao admin I otvoren je prozor za upravljanje korisnicima

Postuslovi: Podaci su sačuvani u bazi podataka



**Slika 7.13.1 Dijagram sekvenci UG13 – Izmena korisnika**

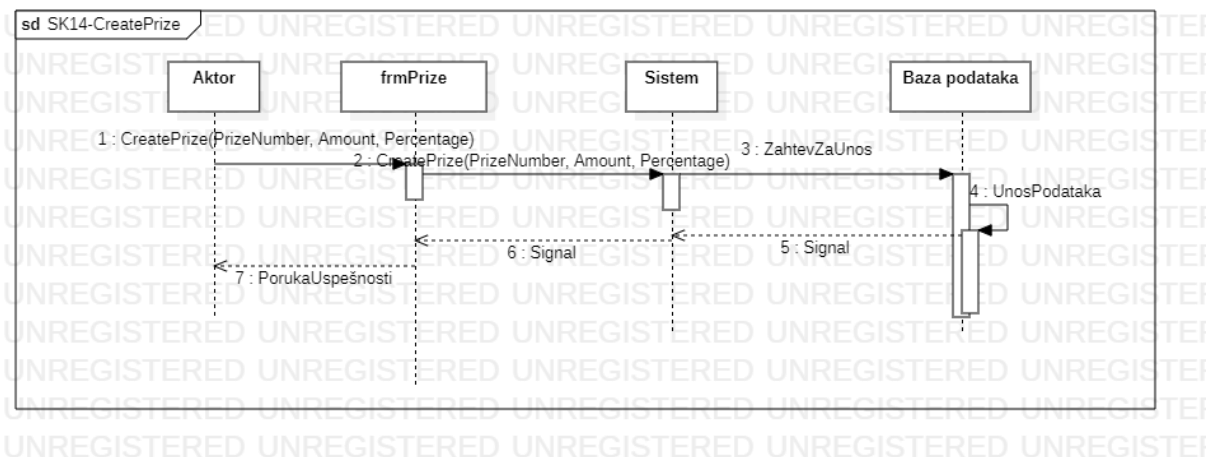
### Ugovor UG14: Kreiranje nagrade

Operacija: CreatePrize(PrizeNumber, Amount, Percentage)

Veza sa SK: SK14

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao organizator ili admin I otvoren je prozor za upravljanje nagradama

Postuslovi: Podaci su sačuvani u bazi podataka



**Slika 7.14.1 Dijagram sekvenci UG14 – Kreiranje nagrade**

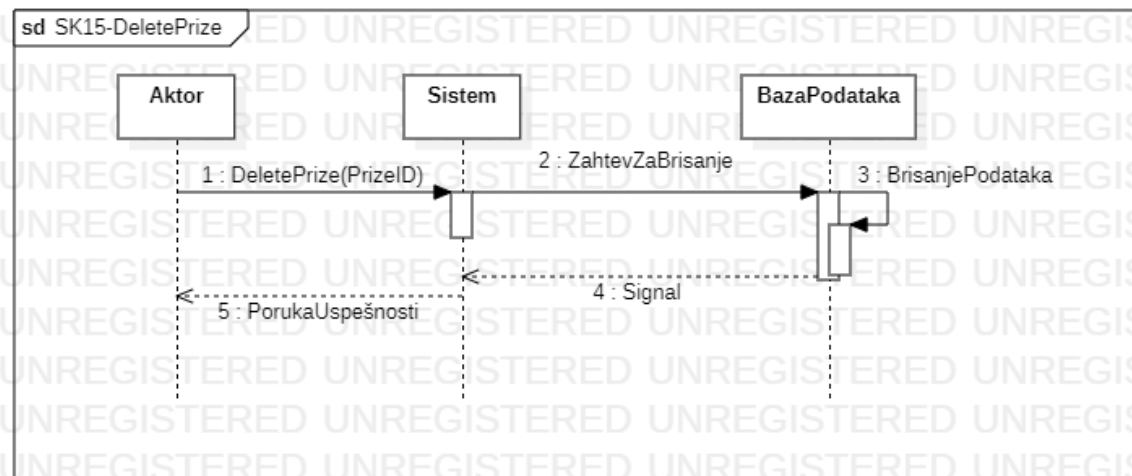
**Ugovor UG15: Brisanje nagrade**

Operacija: DeletePrize(PrizeID)

Veza sa SK: SK15

Preduslovi: Aplikacija je pokrenuta, korisnik je ulogovan kao organizator ili admin I otvoren je prozor za upravljanje nagradama

Postuslovi: Podaci o nagradi su obrisani iz baze podataka



**Slika 7.15.1 Dijagram sekvenci UG15 – Brisanje nagrade**

**7.2. PROJEKTOVANJE SKLADIŠTA PODATAKA**

Skladište podataka se projektuje na osnovu softverskih klasa strukture. Kao rezultat se dobijaju tabele relacionog sistema za upravljanje bazom podataka.



**Tabela: Roles**

	Column Name	Data Type	Allow Nulls
	RoleID	int	<input type="checkbox"/>
	RoleName	varchar(15)	<input type="checkbox"/>


**Tabela: Brackets**

	Column Name	Data Type	Allow Nulls
	BracketID	int	<input type="checkbox"/>
	TournamentID	int	<input type="checkbox"/>


**Tabela: Matches**

	Column Name	Data Type	Allow Nulls
	MatchID	int	<input type="checkbox"/>
	FirstParticipantID	int	<input checked="" type="checkbox"/>
	SecondParticipantID	int	<input checked="" type="checkbox"/>
	WinnerID	int	<input checked="" type="checkbox"/>
	NextMatch	int	<input type="checkbox"/>
	RoundID	int	<input type="checkbox"/>
	TournamentID	int	<input type="checkbox"/>


**Tabela: Participants**

	Column Name	Data Type	Allow Nulls
	ParticipantID	int	<input type="checkbox"/>
	Name	nvarchar(50)	<input type="checkbox"/>
	Email	varchar(50)	<input type="checkbox"/>



**Tabela: Prizes**

	Column Name	Data Type	Allow Nulls
	placeNumber	int	<input type="checkbox"/>
	prizeAmount	float	<input type="checkbox"/>
	prizePercentage	float	<input type="checkbox"/>
	TournamentID	int	<input type="checkbox"/>
	id	int	<input type="checkbox"/>

**Tabela: Rounds**

	Column Name	Data Type	Allow Nulls
	RoundID	int	<input type="checkbox"/>
	BracketID	int	<input type="checkbox"/>
	RoundNumber	int	<input type="checkbox"/>

**Tabela: TournamentParticipants**

	Column Name	Data Type	Allow Nulls
	TournamentID	int	<input type="checkbox"/>
	ParticipantID	int	<input type="checkbox"/>



**Tabela: Tournaments**

	Column Name	Data Type	Allow Nulls
	Name	nvarchar(50)	<input type="checkbox"/>
	Date	date	<input type="checkbox"/>
	NumberOfParticipant	int	<input type="checkbox"/>
	Payouts	int	<input type="checkbox"/>
	Fee	float	<input type="checkbox"/>
🔑	TournamentID	int	<input type="checkbox"/>
	Organizer	nvarchar(50)	<input type="checkbox"/>
	Winner	int	<input checked="" type="checkbox"/>

**7.3.PROJEKTOVANJE KORISNIČKOG INTERFEJSA****SK1:Registrowanje korisnika**

Osnovni scenario: Korisnik popunjava formu za registraciju i poziva sistem da ga registruje.  
Preduslovi: Korisnik je pokrenuo aplikaciju I nalazi se na formi za registrovanje.



Username:

Password:

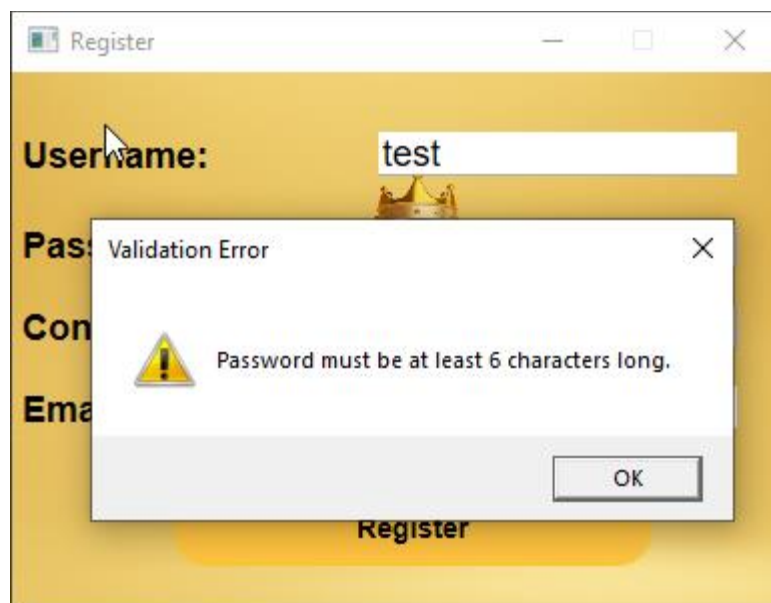
Confirm Password:

Email:

**Register**

**Slika 7.3.1.1 – Registrovanje korisnika**

Alternativni scenario: Korisnik neuspešno popunjava formu za aplikaciju I sistem javlja grešku.

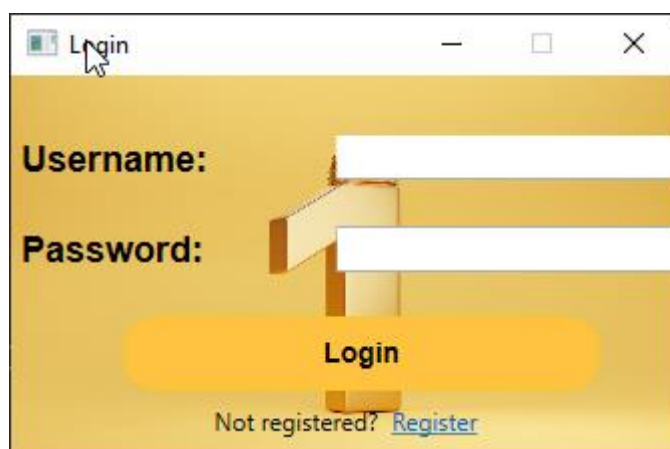


**Slika 7.3.1.2 - Neuspešno registrovanje korisnika**

#### **SK2: Logovanje na aplikaciju**

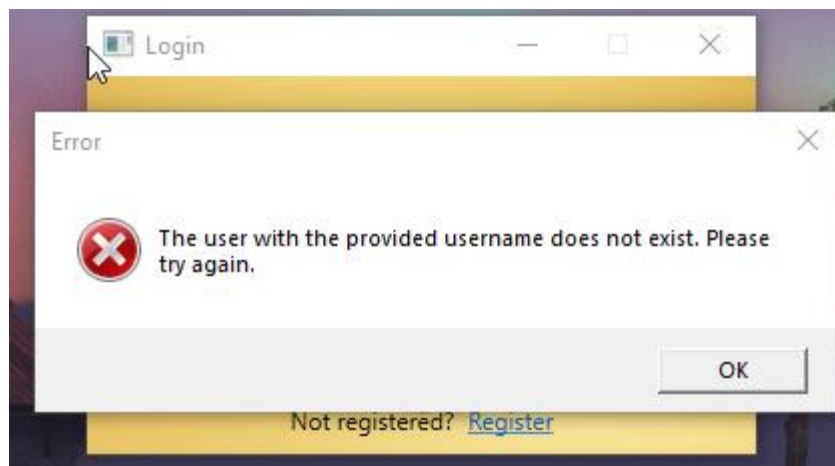
Osnovni scenario: Korisnik unosi podatke za logovanje i poziva sistem da ga uloguje.

Preduslovi: Aplikacija je pokrenuta i korisnik se nalazi na formi za logovanje.



**Slika 7.3.2.1 – Logovanje na aplikaciju**

Alternativni scenario: Korisnik ne uspeva da se loguje na sistem i sistem prikazuje grešku.



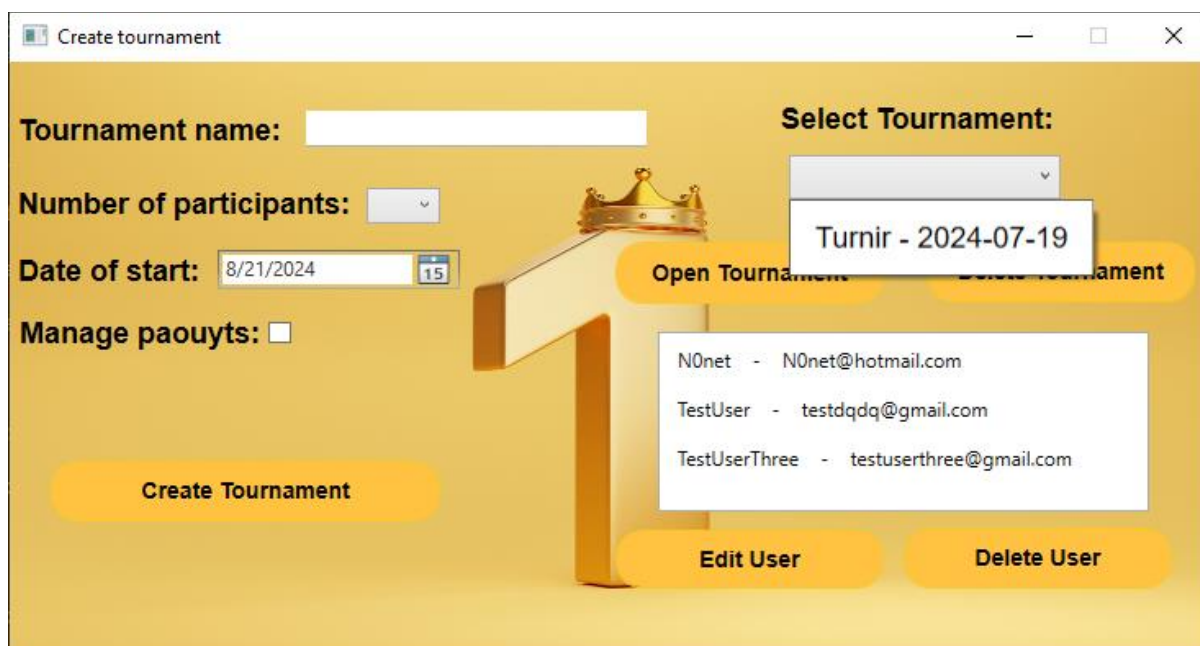
**Slika 7.3.2.2 – Neuspešno logovanje**

### **SK3: Selektovanje turnira**

Osnovni scenario:

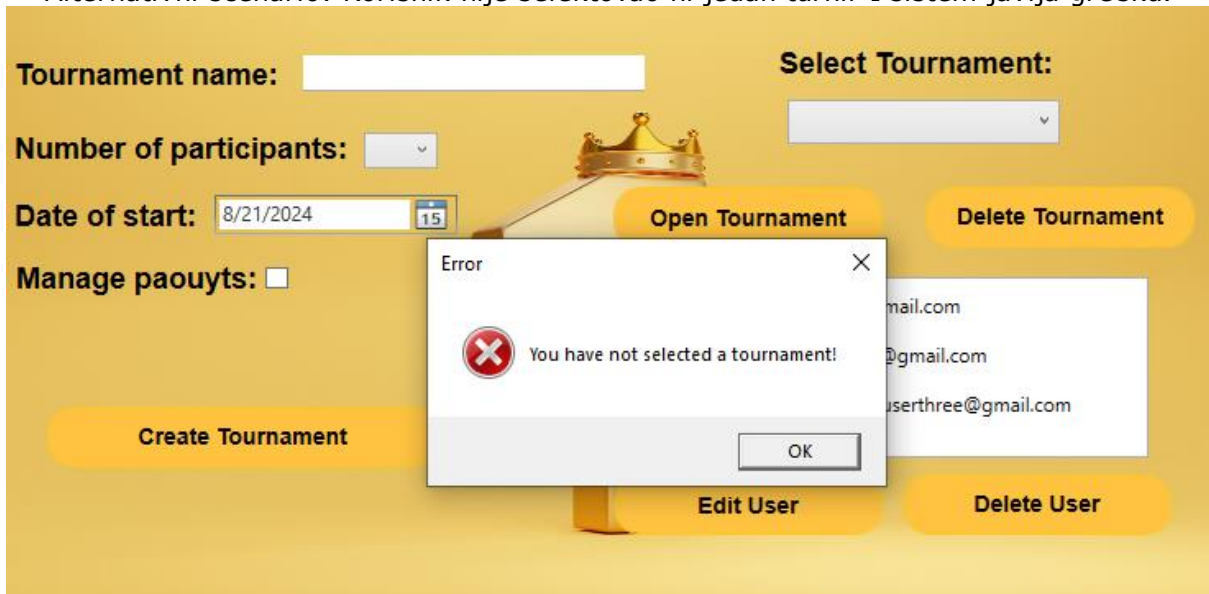
Korisnik poziva sistem da mu prikaže selektovani turnir

Preduslovi: Aplikacija je pokrenuta, korisnik sa rolom admin ili organizator je ulogovan.



**Slika 7.3.3.1 – Selektovanje turnira**

Alternativni scenario: Korisnik nije selektovao ni jedan turnir I sistem javlja grešku.

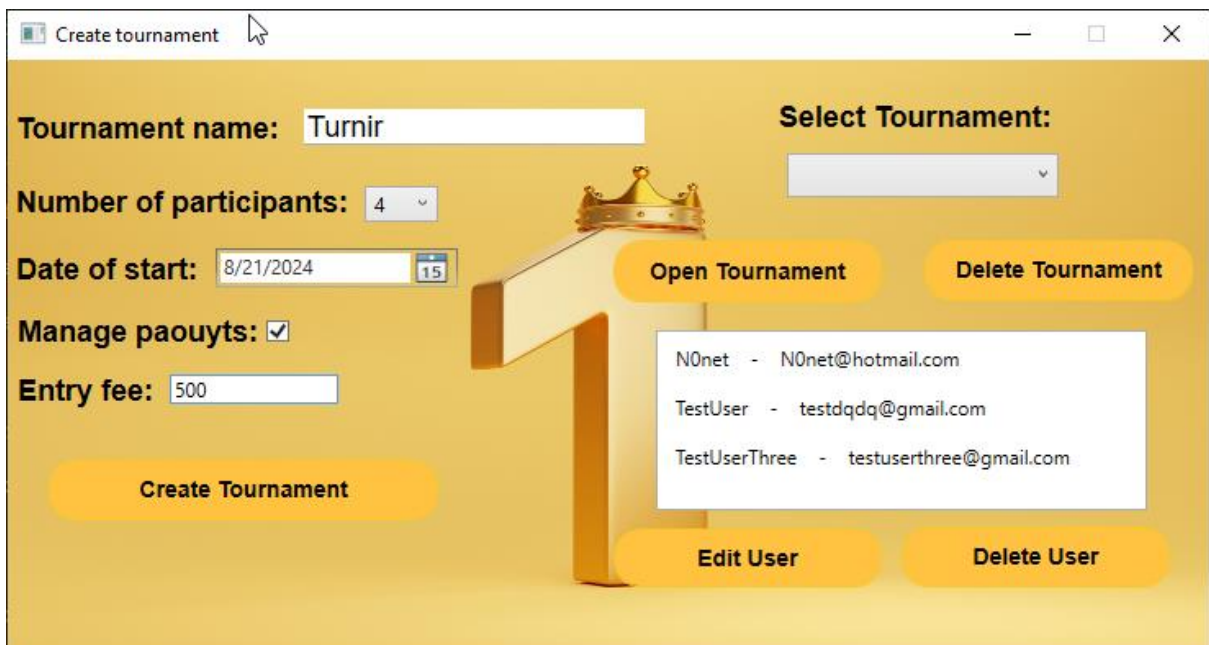


**Slika 7.3.3.2 – Neuspešno selektovanje turnira**

#### SK4: Kreiranje novog turnira

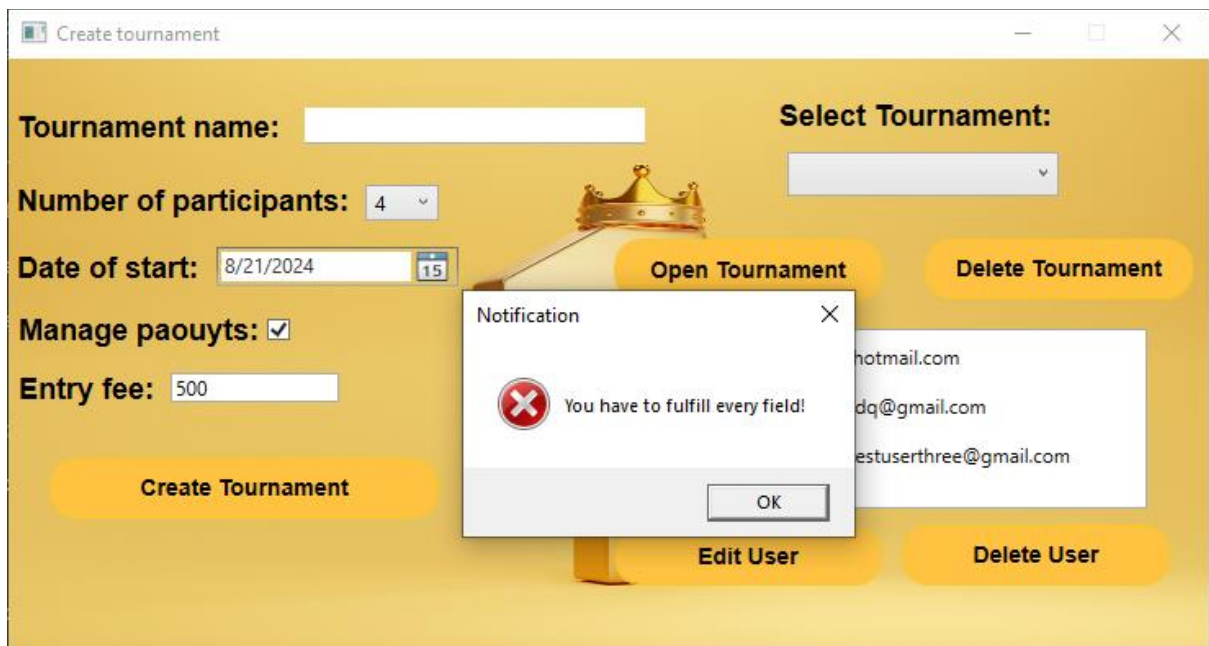
Osnovni scenario:

Korisnik unosi podatke o turniru I poziva sistem da kreira novi turnir



**Slika 7.3.4.1 – Kreiranje novog turnira**

Alternativni scenario: Korisnik nespešno popunjava podatke o turniru i sistem javlja grešku.

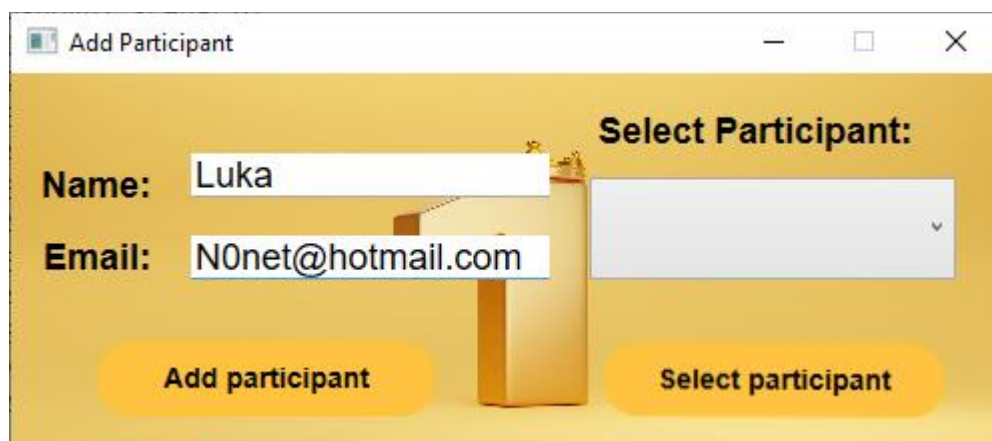


**Slika 7.3.4.2 – Kreiranje novog turnira**

#### **SK7: Dodavanje takmičara**

Osnovni scenario:

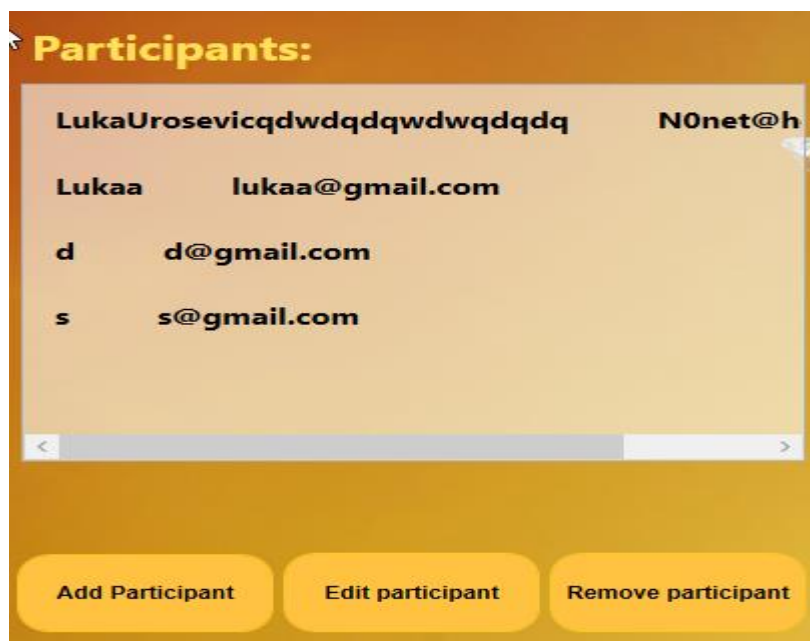
Korisnik unosi podatke o takmičaru ili selektuje već postojećeg takmičara i poziva sistem da doda takmičara



**Slika 7.3.7.1 – Dodavanje takmičara SK8: Brisanje takmičara**

Osnovni scenario:

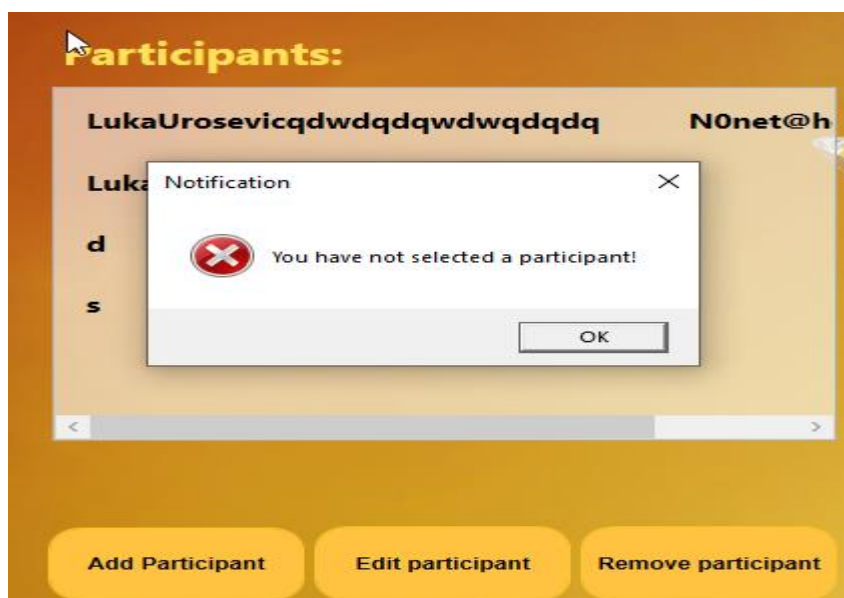
Korisnik selektuje takmičara i poziva sistem da ga obriše.



**Slika 7.3.8.1 -Brisanje takmičara**

Alternativni scenario:

Korisnik nije selektovao ni jednog takmičara i sistem javlja grešku

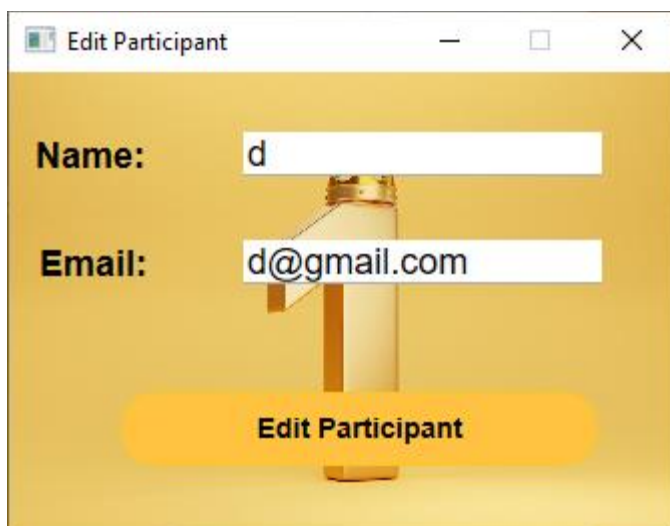


**Slika 7.3.8.2 - Neuspešno brisanje takmičara**

### **SK9: Izmena takmičara**

Osnovni scenario:

Korisnik unosi nove podatke o takmičaru I poziva sistem da ih sačuva u bazi podataka.

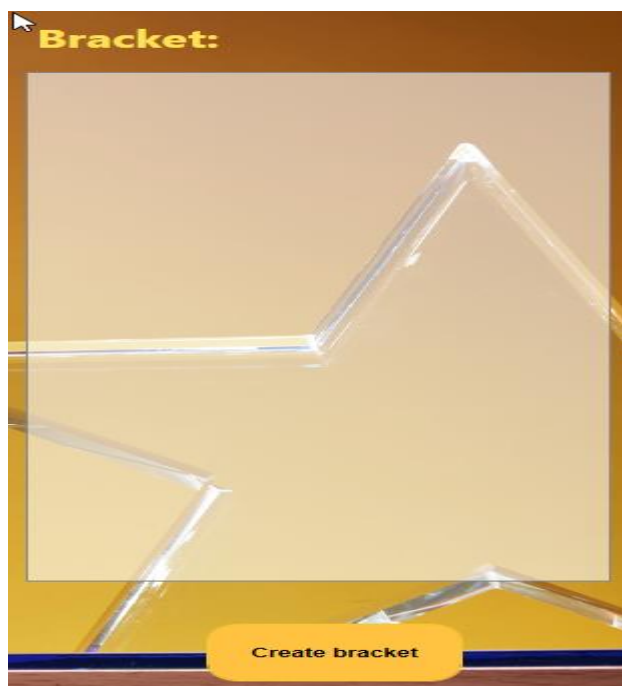


**Slika 7.3.9.1 – Izmena takmičara**

#### **SK10: Kreiranje žreba**

Osnovni scenario:

Korisnik poziva sistem da kreira žreb za takmičenje

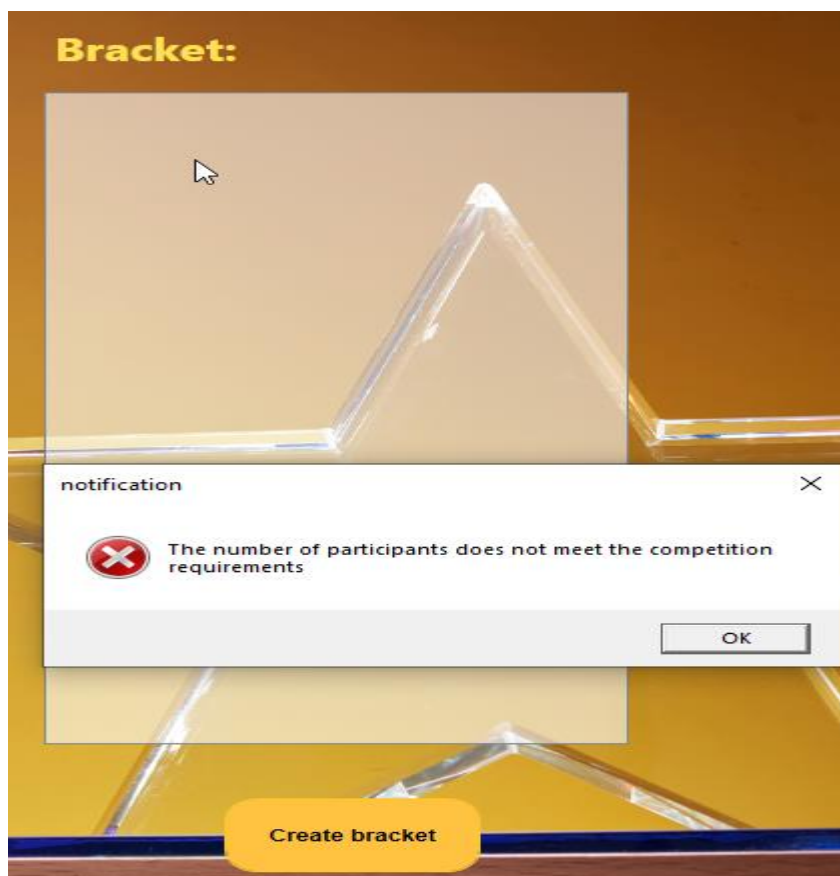


**Slika 7.3.10.1 – Kreiranje žreba**

Alternativni scenario:

Nisu ispunjeni uslovi za kreiranje žreba i sistem javlja grešku





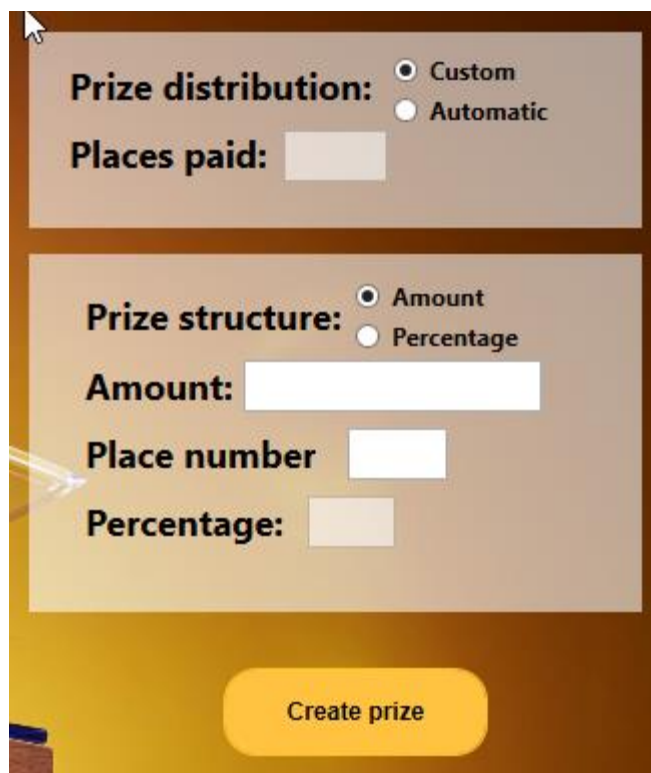
**Slika 7.3.10.2 – Neuspešno kreiranje žreba**

#### **SK16: Brisanje takmičara**

Osnovni scenario:

Korisnik bira opciji custom ili automatic za kreiranje nagrade. Nakon toga korisnik popunjava informacije potrebne za kreiranje nagrade i poziva sistem da kreira istu.





**Prize distribution:** ☒ Custom ☐ Automatic  
**Places paid:**

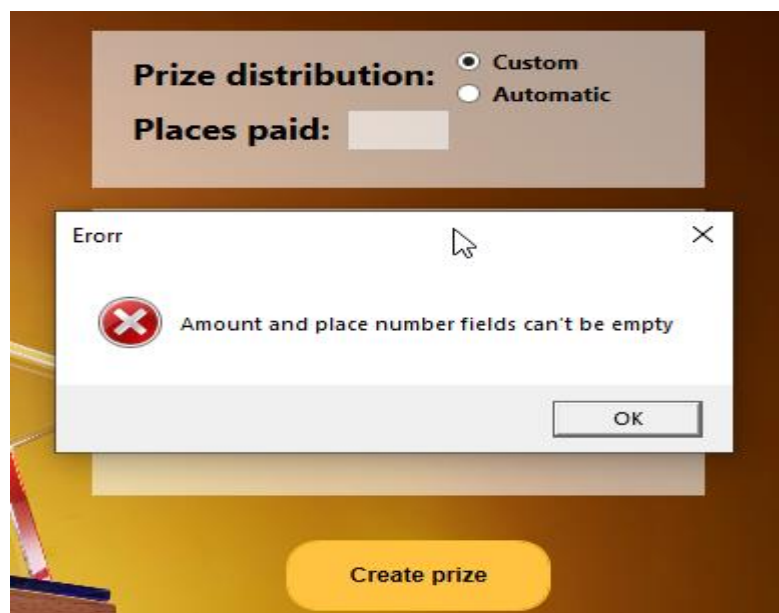
**Prize structure:** ☒ Amount ☐ Percentage  
**Amount:**   
**Place number**   
**Percentage:**

Create prize

**Slika 7.3.16.1 -Kreiranje nagrade**

Alternativni scenario:

Korisnik nije pravilno popunio formu za kreiranje nagrade I sistem javlja grešku.



**Slika 7.3.16.2 - Neuspešno kreiranje nagrade**

## 8. ZAKLJUČAK

Početna ideja aplikacije nastala je kao lična želja da kreiram aplikaciju koja će mi pomoći u organizaciji različitih vrsta turnira. Sve prikazano u dokumentaciji je prva faza aplikacije koja može da se koristi. U budućnosti je planirano da se aplikacija unapredi, doradi i iskoristi u realnoj upotrebi. Domen ove aplikacije je potencijalno veliki. Aplikacija može da se unapredi na mnogo načina.... Unapređenje grafičkog user interfacea(GUI-a), dodavanje funkcionalnosti kao što su: takmičenje između ekipa, dodavanje različitih sistema takmičenja, unaprediti users account management... Takođe aplikaciju je moguće dalje optimizovati da bi se omogućio scaling same aplikacije. Optimizacija bi mogla da se odradi na više načina kao što su: unapređenje algoritama i struktura podataka, korišćenje asinhronizovanih funkcija, unaprediti rad sa memorijom...

## 9. LITERATURA

### 9.1.Fizička literatura

1. Majstorović M: Uvod u baze podataka, ITS, Beograd, 2015.
2. Majstorović M: Baze podataka, skripta (na Studentskom portalu), ITS, Beograd, 2013.
3. Svetlana Anđelić, WPF i ASP.NET Framework - projektovanje i implementacija softvera, ITS, 2016
4. Majstorović M: Napredne baze podataka, predavanja za predmet u ppt formi, školske 2018/2019. (na Studentskom portalu)
5. "Scrum" - Knjiga Džefa Saterlanda
6. "The Pragmatic Programmer" – Knjiga David Thomasa I Andrew Hunta
7. "C# 10 and .NET 6 – Modern Cross-Platform Development" Knjiga Mark J. Price

### 9.2.Internet literatura

1. <https://www.w3schools.com/>
2. <https://www.udemy.com/>
3. <https://stackoverflow.com>
4. <https://geeksforgeeks.org/>