**UNIVERZITET U BEOGRADU**

**FAKULTET ORGANIZACIONIH NAUKA**

Katedra za softversko inženjerstvo   
Seminarski rad iz predmeta:

**Projektovanje softvera**Softverski sistem za naručivanje hrane za zaposlene u kompaniji u Java okruženju

Profesor Student

Dr Dušan Savić Branislav Stojanović 79/18

**Beograd 2021**

Sadržaj

[1. Prikupljanje korisničkih zahteva 3](#_Toc96292602)

[1.1. Verbalni opis 3](#_Toc96292603)

[1.2. Slučajevi korišćenja 3](#_Toc96292604)

[1.2.1 SK1 – Kreiranje naloga zaposlenog 4](#_Toc96292605)

[1.2.2 SK2 – Deaktivacija naloga zaposlenog 6](#_Toc96292606)

[1.2.3 SK3 – Prijavljivanje korisnika 7](#_Toc96292607)

[1.2.4 SK4 – Kreiranje ponude jela (složen) 8](#_Toc96292608)

[1.2.5 SK5 – Izmena ponude jela (složen) 9](#_Toc96292609)

[1.2.6 SK6 – Pregled porudžbina 10](#_Toc96292610)

[1.2.7 SK7 – Kreiranje grupne porudžbine 11](#_Toc96292611)

[1.2.8 SK8 – Poručivanje (složen) 12](#_Toc96292612)

[1.2.9 SK9 – Izmena porudžbine 13](#_Toc96292613)

[2. Analiza 14](#_Toc96292614)

[2.1. Ponašanje softverskog sistema - Sistemski dijagram sekvenci 14](#_Toc96292615)

[2.1.1. DS1 - Kreiranje naloga zaposlenog 14](#_Toc96292616)

[2.1.2 DS2 – Deaktivacija naloga zaposlenog 16](#_Toc96292617)

[2.1.3 DS3 – Prijavljivanje korisnika 17](#_Toc96292618)

[2.1.4 DS4 – Kreiranje ponude jela 19](#_Toc96292619)

[2.1.5 DS5 – Izmena ponude jela 20](#_Toc96292620)

[2.1.6 DS6 – Pregled porudžbina 22](#_Toc96292621)

[2.1.7 DS7 – Kreiranje grupne porudžbine 24](#_Toc96292622)

[2.1.8 DS8 – Poručivanje 26](#_Toc96292623)

[2.1.9 DS9 – Izmena porudžbine 27](#_Toc96292624)

[2.2 . Ponašanje softverskog sistema - Definisanje ugovora sa sistemskim operacijama 29](#_Toc96292625)

[2.3. Sturktura softverskog sistema - Konceptualni (Domenski) model 31](#_Toc96292626)

[2.4. Struktura softverskog sistema – Relacioni model 31](#_Toc96292627)

[3. Projektovanje 37](#_Toc96292628)

[3.1 Projektovanje korisničkog interfejsa 37](#_Toc96292629)

[3.1.1 Projektovanje ekranskih formi 38](#_Toc96292630)

[SK3 – Prijavljivanje korisnika 44](#_Toc96292631)

[SK4 – Kreiranje ponude jela (složen) 47](#_Toc96292632)

[SK5 – Izmena ponude jela (složen) 51](#_Toc96292633)

[SK6 – Pregled porudžbina 53](#_Toc96292634)

[SK7 – Kreiranje grupne porudžbine 56](#_Toc96292635)

[SK8 – Poručivanje (složen) 59](#_Toc96292636)

[SK9 – Izmena porudžbine 62](#_Toc96292637)

[3.1.2 Projektovanje kontrolera korisničkog interfejsa 66](#_Toc96292638)

[3.2 Projektovanje aplikacione logike 66](#_Toc96292639)

[3.2.1 Kontroler aplikacione logike 66](#_Toc96292640)

[3.3.2 Poslovna logika 67](#_Toc96292641)

[3.3.3 Broker baze podataka 71](#_Toc96292642)

[3.4. Projektovanje skladišta podataka 71](#_Toc96292643)

[4. Implementacija 74](#_Toc96292644)

[5. Testiranje 74](#_Toc96292645)

# Prikupljanje korisničkih zahteva

## Verbalni opis

Ovaj sistem ima za zadatak da omogući zaposlenima u kompaniji da elektronskim putem vrše izbor jela za ručak za ordeđeni dan. Postoje dva tipa korisnika ovog sistema, to su administrator sistema i zaposleni.

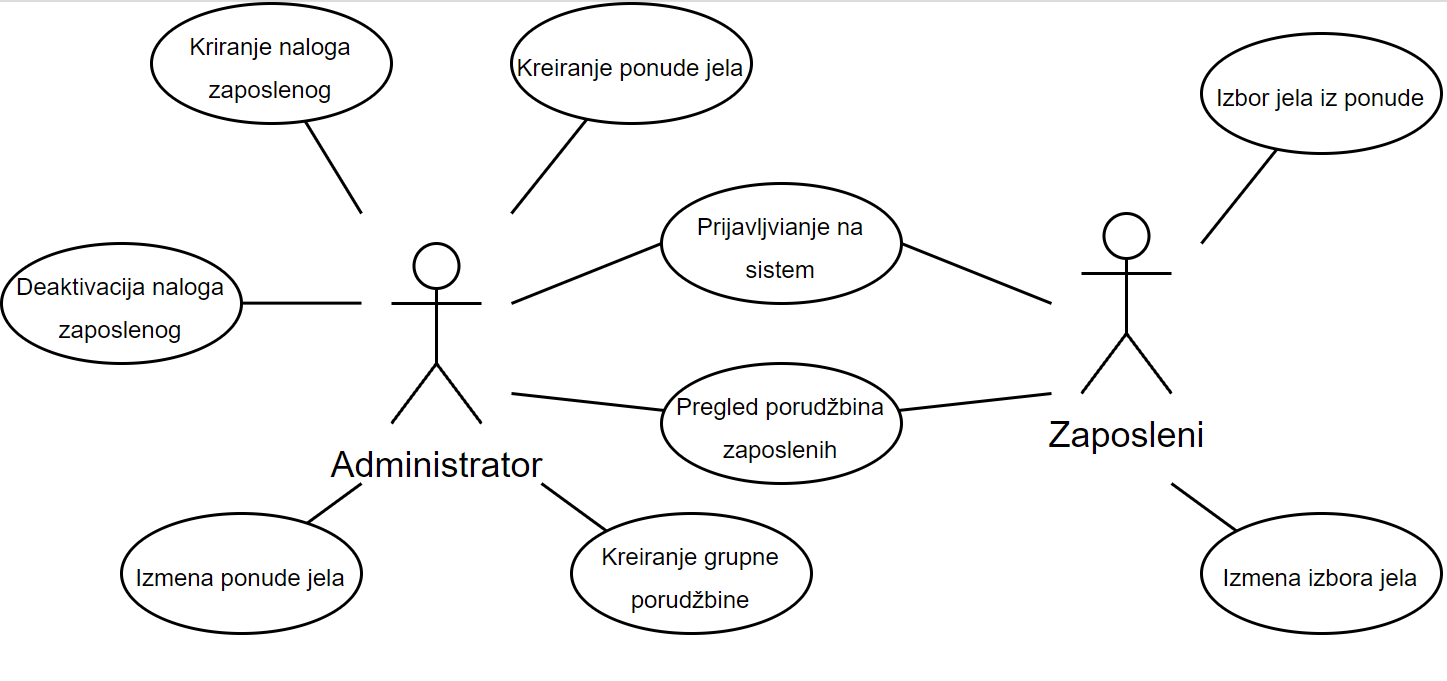
U ovom sistemu korisnik administrator zadužen je da pred za svaki dan predstojeće radne nedelje kreira ponudu jela na osnovu koje će zaposleni moći da biraju jela koja žele da jedu za predjelo, glavno jelo, salatu i dezert i na taj način formiraju svoj obrok za taj dan. Zaposleni se zatim prijavljuju na sistem, pregledaju ponudu jela i za svaki dan biraju po jedno od svake kategorije jela. Nakon što zaposleni izaberu jela koja žele za narednu nedelju, administrator sistema vrši pregled porudžbina i ukoliko neki zaposleni nije izabrao jela za narednu nedelju administrator ga podseća da to uradi. Administrator sistema pre početka svakog radnog dana pregleda porudžbine zaposlenih za taj dan, zatim kreira grupnu porudžbinu u kojoj je navedena ukupna količina svakog od jela koja je poručena za taj dan, a zatim grupnu porudžbinu šalje kompaniji koja je zadužena za pružanje usluga keteringa kompaniji koja koristi ovaj sistem.

## Slučajevi korišćenja

Identifikovano je 9 slučajeva korišćenja ovog softverskog sistema:

1. Kreiranje naloga zaposlenog
2. Deaktivacija naloga zaposlenog
3. Prijavljivanje korisnika
4. Kreiranje ponude jela (složen)
5. Izmena ponude jela (složen)
6. Pregled porudžbina
7. Kreiranje grupne porudžbine (složen)
8. Poručivanje
9. Izmena porudžbine

Figure 1 – Dijagram slučajeva korišćenja



### 1.2.1 SK1 – Kreiranje naloga zaposlenog

**Naziv SK:** Kreiranje naloga zaposlenog

**Aktori SK:** Administrator

**Učesnici SK:** Administrator i sistem

**Preduslov:** Sistem je pokrenut, Administrator je ulogovan na svoj nalog, prikazana je lista svih zaposlenih u firmi.

**Osnovni scenario SK:**

1. Administrator **bira** zaposlenog kojem želi da kreira nalog. (**APUSO**)

2. Administrator **unosi podatke** o nalogu zaposlenog (**korisničko** **ime** i **šifru**) u prazna polja. (**APUSO**)

3. Administrator **kontroliše** da li je korektno uneo podatke o nalogu zaposlenog. (**ANSO**)

4. Administrator poziva sistem da **zapamti** podatke o nalogu zaposlenog. (**APSO**)

5. Sistem **pamti** podatke o nalogu zaposlenog. (**SO**)

6. Sistem **prikazuje** adminstratoru poruku: “Nalo je uspešno kreiran “. (**IA**)

**Alternativni scenariji:**

2.1. Ukoliko je aktor izabrao zaposlenog koji već poseduje nalog prikazaće mu se poruka “Zaposleni već poseduje nalog”.(**IA**)

6.2. Ukoliko sistem ne može da kreira naog zaposlenog prikazaće poruku “Greška prilikom kreiranja naloga”.(**IA**)

### 1.2.2 SK2 – Deaktivacija naloga zaposlenog

**Naziv SK: Deaktivacija naloga zaposlenog**

**Aktori SK:** Administrator

**Učesnici SK:** Administrator, sistem

**Preduslov:** Sistem je pokrenut, administrator je ulogovan na svoj nalog, učitana je lista na kojoj su izlistani svi zaposleni.

**Osnovni scenario SK:**

1. Administrator **bira zaposlenog** čiji nalog želi da deaktivira. (**APUSO**)
2. Administrator **poziva** sistem da deaktivira nalog zaposlenog. (**APSO**)
3. Sistem traži od aktora **da potvrdi** poziv za deaktivacijom naloga (**APUSO**)

4. Sistem **deaktivira** nalog zaposlenog i **deaktivira** sve buduće porudžbine zaposlenog. (**SO**)

5. Sistem **prikazuje poruku** “Nalog je uspešno deaktiviran”. (**IA**)

**Alternativni scenariji:**

2.1. Ako zaposleni nema nalog prikazuje se poruka “Zaposleni ne poseduje nalog**.**” (**IA**)

### 1.2.3 SK3 – Prijavljivanje korisnika

**Naziv SK:** Prijavljivanje korisnika

**Aktori SK:** Administrator, zaposleni

**Učesnici SK:** Administrator, zaposleni, sistem

**Preduslov:** Sistem je pokrenut, prikazana je forma za unos korisničkog imena i lozinke

**Osnovni scenario SK:**

1. Aktor **unosi** korisničko ime i šifru. (**APUSO**)

2. Aktor **kontroliše** da li je korektno uneo korisničko ime i lozinku. (**ANSO**)

3. Aktor **poziva sistem** da ga prijavi. (**APSO**)

4. Sistem **prijavljuje** korisnika. (**SO**)

5. Sistem **prikazuje** korisniku glavnu formu. (**IA**)

**Alternativni scenariji:**

5.1. Ako sistem **ne pronađe** korisnika sa datim korisničkim imenom prikazuje se poruka “Nevalidno korisničko ime ili šifra”. (**IA**)

5.2. Ako je nalog korisnika koji želi da se prijavi **deaktiviran** prikazuje se poruka “Nalog je deaktiviran”. (**IA**)

### 1.2.4 SK4 – Kreiranje ponude jela (složen)

**Naziv SK:** Kreiranje ponude jela

**Aktori SK:** Administrator

**Učesnici SK:** Administrator, sistem

**Preduslov:** Sistem je pokrenut, administrator je ulogovan na svoj nalog, otvorena je forma za kreiranje ponude.

**Osnovni scenario SK:**

1. Administrator **bira** dan za koji hoće da kreira ponudu jela. (**APUSO**)
2. Administrator **vrši izbor** jela koje će zaposleni moći da biraju za taj dan. (**APUSO**)
3. Administrator kontroliše da li je pravilno popunio formu (**ANSO**)
4. Administrator **poziva** sistem da sačuva ponudu jela. (**APSO**)
5. Sistem **pamti** ponudu jela (**SO**)
6. Sistem **prikazuje** poruku “Ponuda jela je uspešno sačuvana”.(**IA**)

**Alternativni scenariji:**

6.1. Ukoliko je administrator izabrao dan za koje već postoji ponuda jela sistem prikazuje poruku: “Ponuda za datum već postoji”. (**IA**)

6.2. Ukoliko je korisnik izabrao za datum ponude neki dan iz prošlosti prikazuje se poruka: “Datum ponude ne sme biti u prošlosti”.(**IA**)

### 1.2.5 SK5 – Izmena ponude jela (složen)

**Naziv SK:** Izmena ponude jela

**Aktori SK:** Administrator

**Učesnici SK:** Administrator, sistem

**Preduslov:** Sistem je pokrenut, administrator je ulogovan na svoj nalog, učitana je lista svih dana za koje je kreirana ponuda.

**Osnovni scenario SK:**

1. Administrator **bira** dan za koju želi da promeni ponudu. (**APUSO**)
2. Administrator **poziva** sistem da učita ponudu za taj dan. (**APSO**)

3. Sistem **pronalazi** ponudu za izabrani dan. (**SO**)

4. Sistem **prikazuje** ponudu jela koje je administrator izabrao za taj dan. (**IA**)

5. Administrator **menja** jela. (**APUSO**)

6. Administrator **poziva** sistem da ažurira ponudu jela za taj dan. (**APSO**)

7. Sistem **ažurira** ponudu jela. (**SO**)

8. Sistem **prikazuje** poruku “Ponuda je uspešno sačuvana”. (**IA**)

**Alternativni scenariji:**

8.1. Ukoliko je administrator izabrao dan za koje već postoji ponuda jela sistem prikazuje poruku: “Ponuda za datum već postoji”. (**IA**)

8.2. Ukoliko je korisnik izabrao za datum ponude neki dan iz prošlosti prikazuje se poruka: “Datum ponude ne sme biti u prošlosti”.(**IA**)

### 1.2.6 SK6 – Pregled porudžbina

**Naziv SK:** Pregled porudžbina

**Aktori SK:** Zaposleni, Administrator

**Učesnici SK:** Administrator, Zaposleni, sistem

**Preduslov:** Sistem je pokrenut, aktor je ulogovan na svoj nalog, prikazana je lista porudžbina sortirana po danima (opadajuće).

**Osnovni scenario SK:**

1. Aktor **bira** ponudu za koju želi da pregleda poručena jela zaposlenih.(**APUSO**)

2. Aktor **poziva** sistem da učita poručena jela za tu ponudu. (**APSO**)

3. Sistem **učitava** poručena jela za izabranu ponudu. (**SO**)

4. Sistem **prikazuje** izabrana jela za taj dan za svakog zaposlenog. (**IA**)

**Alternativni scenariji:**

4.1. Ukoliko sistem ne može da učita podatke o porudžbini prikazuje poruku: “Greška prilikom učitavanja porudžbine”. (**IA**)

### 1.2.7 SK7 – Kreiranje grupne porudžbine

**Naziv SK:** Kreiranje grupne porudžbine

**Aktori SK:** Administrator

**Učesnici SK:** Administrator, sistem

**Preduslov:** Sistem je pokrenut, aktor je ulogovan na svoj nalog, prikazana je lista porudžbina sortirana po danima (opadajuće).

**Osnovni scenario SK:**

1. Aktor **poziva** sistem da učita porudžbine svih zaposlenih za željenu ponudu. (**APSO**)

2. Sistem **učitava** porudžbine svih zaposlenih za tu ponudu. (**SO**)

3. Sistem **prikazuje** izabrana jela za svakog zaposlenog. (**IA**)

4. Aktor **poziva** sistem da kreira grupnu porudžbinu za tu ponudu. (**APSO**)

5. Sistem **kreira** grupnu porudžbinu (**SO**)

6. Sistem **prikazuje formu** za čuvanje grupne pordužbine kao tekstualnog fajla. (**IA**)

7. Aktor bira lokaciju na fajl sistemu gde želi da sačuva grupnu porudžbinu kao i naziva fajla.(**APUSO**)

8. Aktor **poziva sistem** za sačuva grupnu porudžbinu.(**APSO**)

9. Sistem **čuva** grupnu porudžbinu kao tekstualni fajl. (**IA**)

**Alternativni scenariji:**

3.1. Ukoliko sistem ne može da učita podatke o porudžbini prikazuje poruku: “Greška prilikom učitavanja porudžbine”. (**IA**)

6.1. Ukoliko sistem **ne može** da kreira grupnu porudžbinu, prikazuje poruku “Greška prilikom kreiranja grupne porudžbine”. (**IA**)

### 1.2.8 SK8 – Poručivanje (složen)

**Naziv SK:** Poručivanje

**Aktori SK:** Zaposleni

**Učesnici SK:** Zaposleni, sistem

**Preduslov:** Sistem je pokrenut, zaposleni je ulogovan na svoj nalog, izlistani su svi dani za koje je administrator kreirao ponudu jela.

**Osnovni scenario SK:**

1. Aktor **bira** dan za koji hoće da izvrši poručivanje. (**APUSO**)
2. Aktor **poziva** sistem da učita ponuđena jela za taj dan. (**APSO**)
3. Sistem **učitava** ponuđena jela za izabrani dan. (**SO**)
4. Sistem **prikazuje** ponuđena jela za dan koji je aktor izabrao. (**IA**)
5. Aktor od ponuđenih jela **bira** po jedno jelo od svake od kategorija jela (predjelo, glavno jelo, salata i dezert). (**APUSO**)

5. Aktor **proverava** da li je dobro izabrao jela. (**ANSO**)

6. Aktor **poziva** sistem da sačuva poručena jela.(**APSO**)

7. Sistem **pamti** poručena jela zaposlenog. (**SO**)

8. Sistem prikazuje **poruku** “Porudžbina je uspešno sačuvana”. (**IA**)

**Alternativni scenariji:**

8.1. Ukoliko sistem ne može da sačuva poručena jela, prikazuje se poruku “Greška prilikom čuvanja porudžbine”. (**IA**)

### 1.2.9 SK9 – Izmena porudžbine

**Naziv SK:** Izmena porudžbine

**Aktori SK:** Zaposleni

**Učesnici SK:** Zaposleni, sistem

**Preduslov:** Sistem je pokrenut, zaposleni je ulogovan na svoj nalog, izlistani su svi dani za koje je administrator kreirao ponudu jela.

**Osnovni scenario SK:**

1. Aktor **bira** dan za koji hoće da izmeni porudžbinu. (**APUSO**)
2. Aktor **poziva** sistem da učita ponuđena jela za taj dan kao i jela koja je on poručio .(**APSO**)
3. Sistem **vraća** ponudjena jela za taj dan i jela koja je zaposleni poručio za taj dan.(**SO**)
4. Sistem **prikazuje** poručena jela zaposlenog(**IA**)
5. Aktor **menja** svoju porudžbinu. (**APUSO**)
6. Aktor **poziva** sistem da sačuva izmenjenu porudžbinu. (**APSO**)
7. Sistem **čuva** porudžbinu zaposlenog. (**SO**)
8. Sistem **prikazuje** poruku “Porudžbina je uspešno sačuvana”. (**IA**)

**Alternativni scenariji:**

8.1 Ukoliko sistem **ne može** da sačuva porudžbinu prikazuje se poruku “Greška prilikom čuvanja porudžbine.” (**IA**)

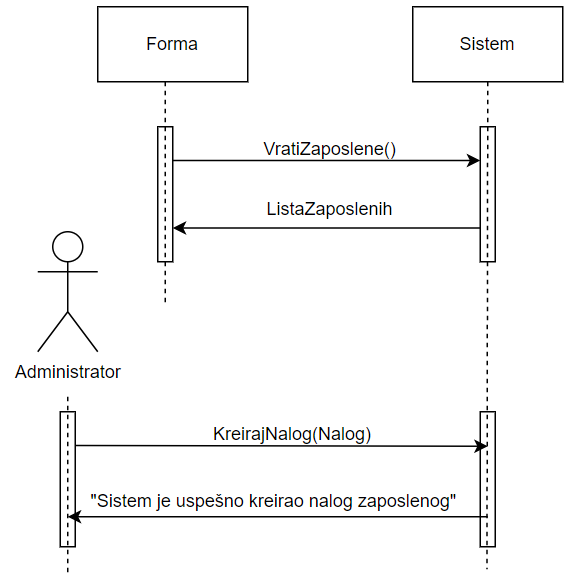
# Analiza

## Ponašanje softverskog sistema - Sistemski dijagram sekvenci

### DS1 - Kreiranje naloga zaposlenog

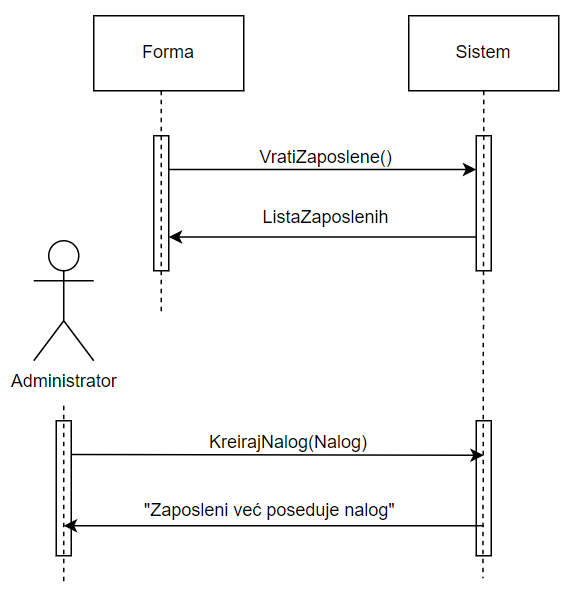
**Osnovni scenario**:

1. Forma **poziva** sistem da vrati listu svih zaposlenih. (**APSO**)
2. Sistem **vraća** listu svih zaposlenih. (**IA**)
3. Administrator **poziva** sistem da zapamti podatke o nalogu zaposlenog (**APSO**)
4. Sistem **prikazuje** adminstratoru poruku: “Nalo je uspešno kreiran “. (**IA**)

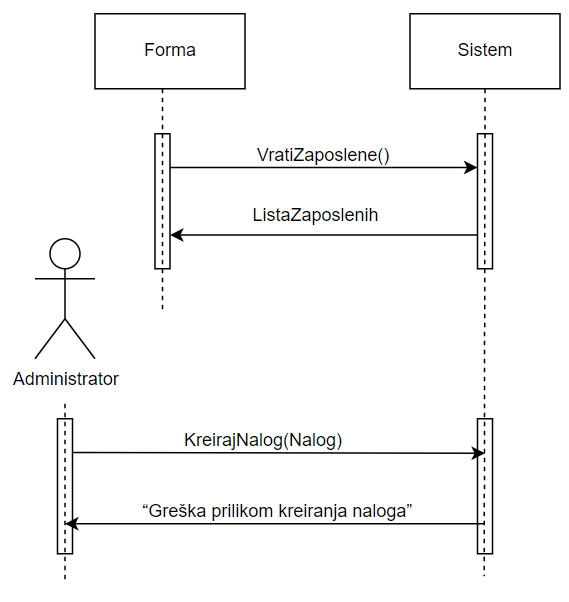


**Alternativni scenariji:**

4.1. Ukoliko je aktor izabrao zaposlenog koji već poseduje nalog prikazaće mu se poruka “Zaposleni već poseduje nalog”(**IA**)



4.2. Ukoliko sistem ne može da kreira naog zaposlenog prikazaće poruku “Greška prilikom kreiranja naloga”.(**IA**)



**Sa navedenih sekvencnih dijagrama uočavaju se dve sistemska operacija:**

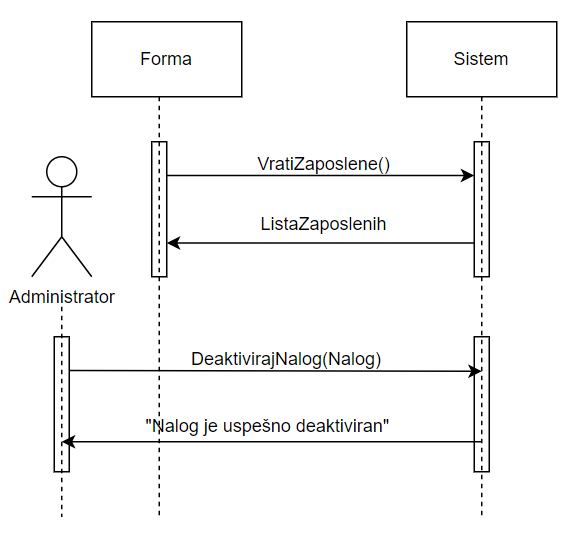
1. **VratiZaposlene**(): ListaZaposlenih
2. **KreirajNalog**(Nalog): Signal

### 

### 2.1.2 DS2 – Deaktivacija naloga zaposlenog

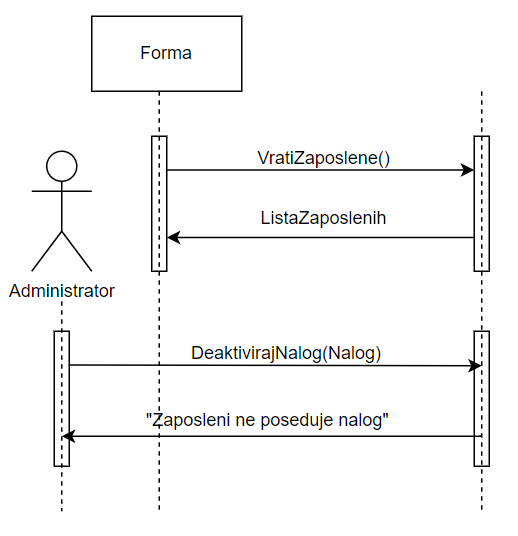
**Osnovni scenario SK:**

1. Forma **poziva** sistem da učita listu zaposlenih. (**APSO**)
2. Sistem **vraća** formi listu zaposlenih (**IA**)
3. Administrator **poziva** sistem da deaktivira nalog zaposlenog. (**APSO**)
4. Sistem **prikazuje poruku** “Nalog je uspešno deaktiviran”. (**IA**)



**Alternativni scenariji:**

2.1. Ako zaposleni nema nalog prikazuje se poruka “Zaposleni ne poseduje nalog.”



**Sa navedenih sekvencnih dijagrama uočavaju se dve sistemska operacija:**

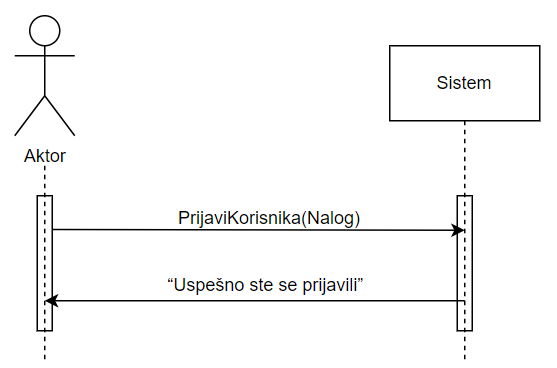
1. **VratiZaposlene**(): ListaZaposlenih
2. **DeaktivirajNalog**(Nalog): Signal

### 2.1.3 DS3 – Prijavljivanje korisnika

**Osnovni scenario SK:**

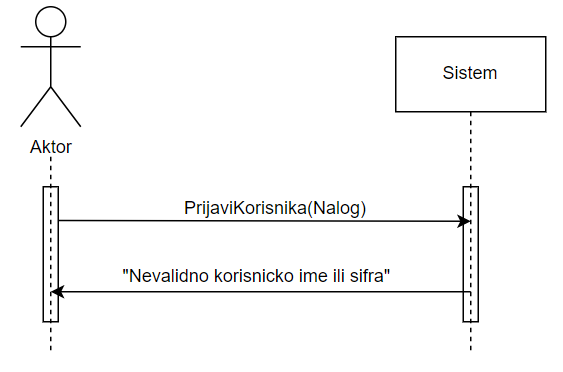
1. Aktor **poziva sistem** da ga prijavi. (**APSO**)

2. Sistem **prikazuje** aktoru **poruku**: “Uspešno ste se prijavili”. (**IA**)

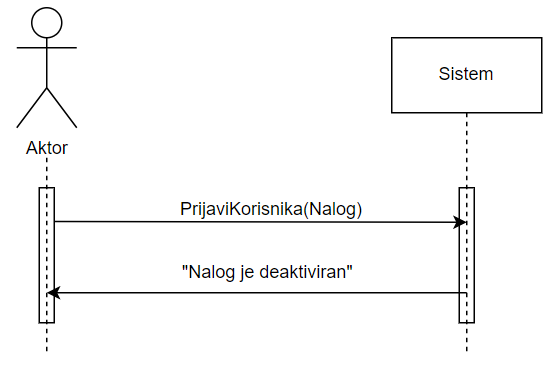


**Alternativni scenariji:**

2.1. Ako sistem **ne pronađe** nalog sa datim korisničkim imenom prikazuje se poruka “Nevalidno korisničko ime ili šifra”. (**IA**)



2.2. Ako je nalog korisnika koji želi da se prijavi **deaktiviran** prikazuje se poruka “Nalog je deaktiviran”. (**IA**)



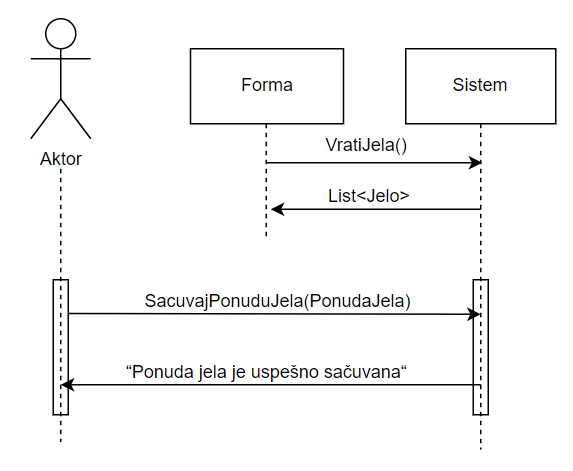
**Sa navedenih sekvencnih dijagrama uočava se jedna sistemska operacija:**

1. **PrijaviKorisnika**(Nalog): Signal

### 2.1.4 DS4 – Kreiranje ponude jela

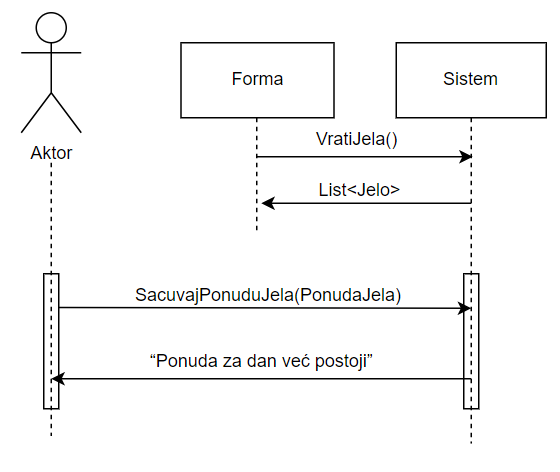
**Osnovni scenario:**

1. Forma poziva sistem da vrati sva jela. (**APSO**)
2. Sistem vraća formi listu jela.(**IA**)
3. Administrator **poziva** sistem da sačuva ponudu jela. (**APSO**)
4. Sistem **prikazuje** poruku “Ponuda jela je uspešno sačuvana”.(**IA**)

****

**Alternativni scenariji:**

4.1. Ukoliko je administrator izabrao dan za koje već postoji ponuda jela sistem prikazuje poruku: “Ponuda za dan već postoji”. (**IA**)



4.2. Ukoliko je korisnik izabrao za datum ponude neki dan iz prošlosti prikazuje se poruka: “Datum ponude ne sme biti u prošlosti”.(**IA**)



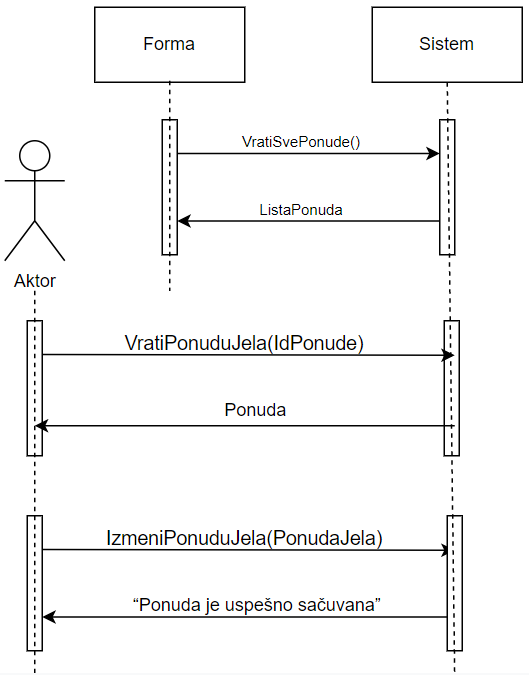
**Sa navedenih sekvencnih dijagrama uočavaju se dve sistemske operacija:**

1. **VratiJela(): List<Jelo>**
2. **SacuvajPonuduJela(PonudaJela)**

### 2.1.5 DS5 – Izmena ponude jela

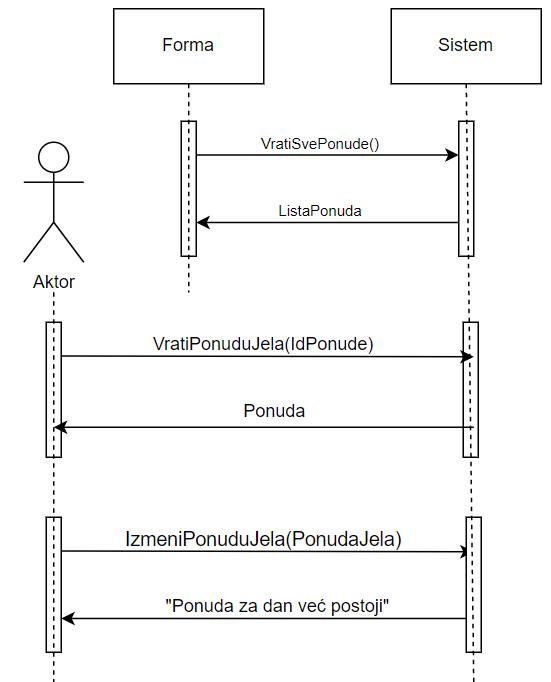
**Osnovni scenario SK:**

1. Forma **poziva** sistem da vrati ponude za sve dane za koje je administrator kreirao ponudu. (**APSO**)
2. Sistem **prikazuje** listu svih ponuda. (**IA**)
3. Administrator **poziva** sistem da učita ponudu za taj dan. (**APSO**)
4. Sistem **prikazuje** ponudu jela koje je administrator izabrao za taj dan. (**IA**)
5. Administrator **poziva** sistem da ažurira ponudu jela za taj dan. (**APSO**)
6. Sistem **prikazuje** poruku “Ponuda je uspešno sačuvana”. (**IA**)

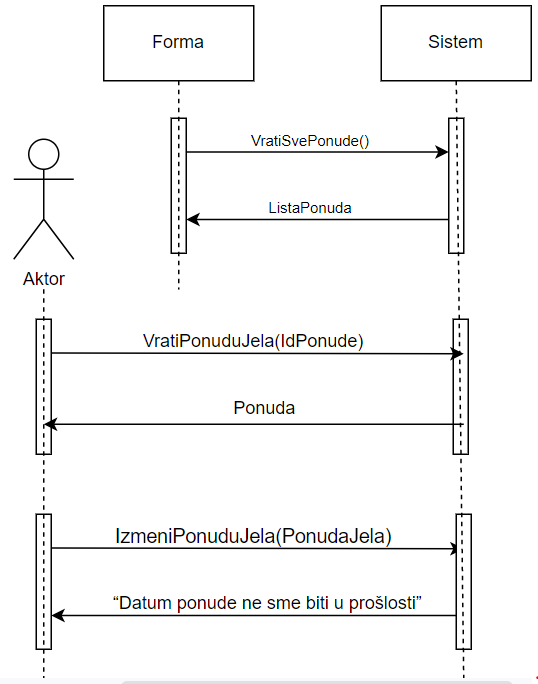


**Alternativni scenariji:**

* 1. Ukoliko je administrator izabrao dan za koje već postoji ponuda jela sistem prikazuje poruku: “Ponuda za dan već postoji”. (**IA**)



* 1. Ukoliko je korisnik izabrao za datum ponude neki dan iz prošlosti prikazuje se poruka: “Datum ponude ne sme biti u prošlosti”.(**IA**)



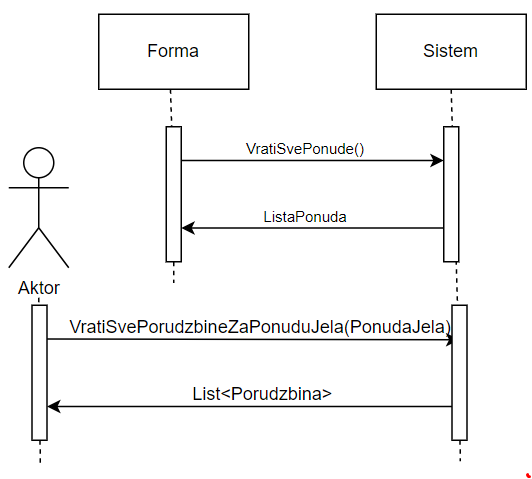
**Sa navedenih sekvencnih dijagrama uočavaju se tri sistemske operacija:**

1. **VratiSvePonude**(): ListaPonuda
2. **IzmeniPonuduJela** (PonudaJela): Signal
3. **VratiPonuduJela**(IdPonude): PonudaJela

### 2.1.6 DS6 – Pregled porudžbina

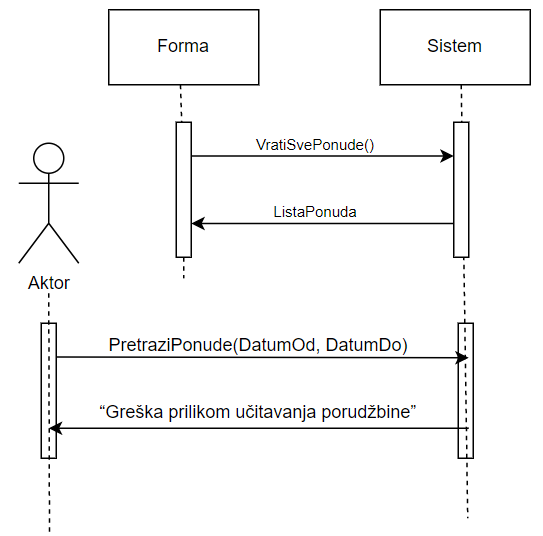
**Osnovni scenario SK:**

1. Forma **poziva** sistem da učita listu svih ponuda. (**APSO**)
2. Sistem **prikazuje** listu svih ponuda. (**IA**)
3. Aktor **poziva** sistem da učita poručena jela za izabranu ponudu. (**APSO**)
4. Sistem **prikazuje** izabrana jela za taj dan za svakog zaposlenog. (**IA**)



**Alternativni scenariji:**

* 1. Ukoliko sistem ne može da učita podatke o porudžbini prikazuje poruku: “Greška prilikom učitavanja porudžbine”. (**IA**)”.



**Sa navedenih sekvencnih dijagrama uočavaju se dve sistemske operacije:**

1. **VratiSvePonude()**: ListaPonuda
2. **VratiSvePorudzbineZaPonuduJela(PonudaJela)** : List<Porudzbina>

### 2.1.7 DS7 – Kreiranje grupne porudžbine

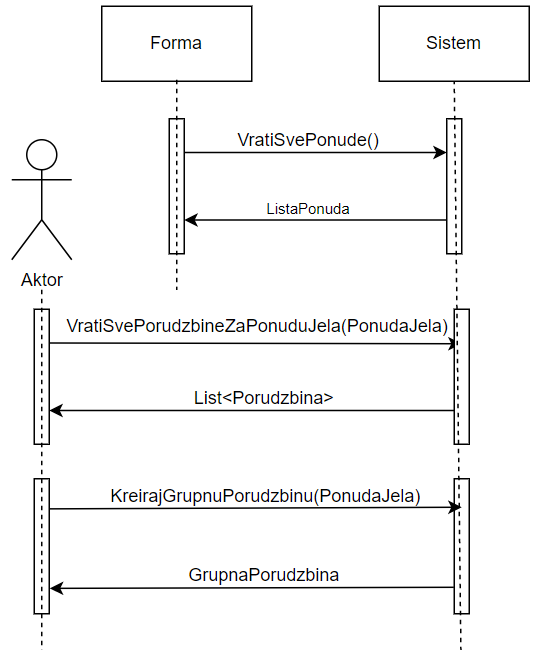
**Osnovni scenario SK:**

1. Forma **poziva** sistem da učita sve ponude. (**APSO**)
2. Sistem **prikazuje** listu svih ponuda. (**IA**)
3. Aktor **poziva** sistem da učita porudžbine svih zaposlenih za željenu ponudu. (**APSO**)
4. Sistem **prikazuje** izabrana jela za svakog zaposlenog. (**IA**)
5. Aktor **poziva** sistem da kreira grupnu porudžbinu za tu ponudu. (**APSO**)

6. Sistem **prikazuje formu** za čuvanje grupne pordužbine kao tekstualnog fajla (**IA**)

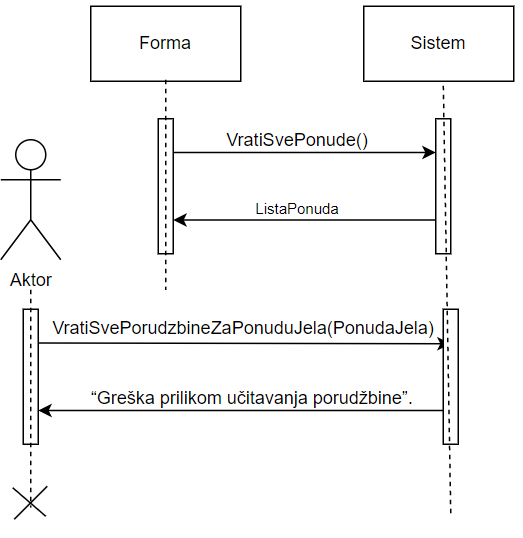
7. Aktor **poziva sistem** za sačuva grupnu porudžbinu.(**APSO**)

8. Sistem **čuva** grupnu porudžbinu kao tekstualni fajl. (**IA**)

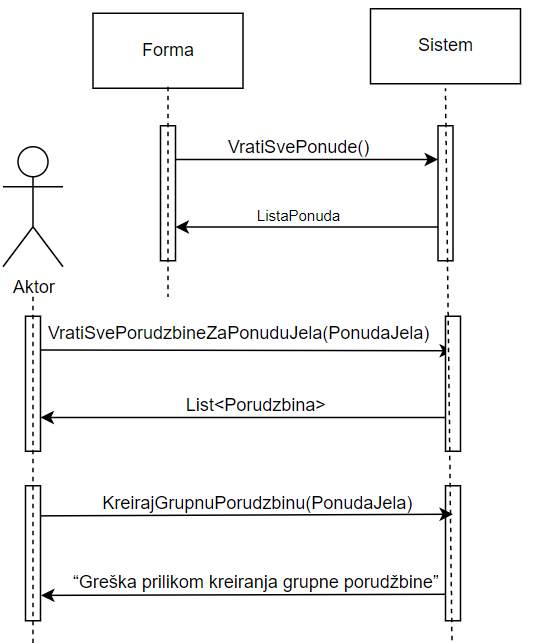


**Alternativni scenariji:**

* 1. Ukoliko sistem ne može da učita podatke o porudžbini prikazuje poruku: “Greška prilikom učitavanja porudžbine”. (**IA**)



* 1. Ukoliko sistem **ne može** da kreira grupnu porudžbinu, prikazuje poruku “Greška prilikom kreiranja grupne porudžbine”. (**IA**)



**Sa navedenih sekvencnih dijagrama uočavaju se tri sistemske operacije:**

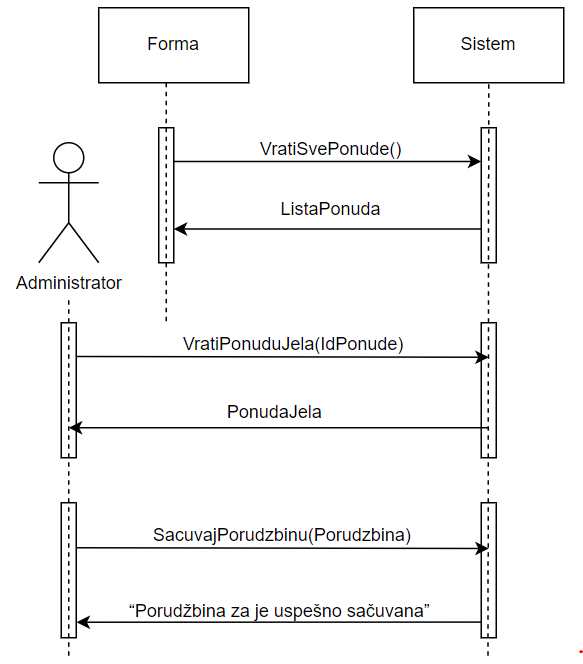
1. **VratiSvePonude**(): ListaPonuda
2. **VratiSvePorudzbineZaPonuduJela(PonudaJela): List<Porudzbina>**

3. **KreirajGrupnuPorudzbinu(**PonudaJela**):** String

### 2.1.8 DS8 – Poručivanje

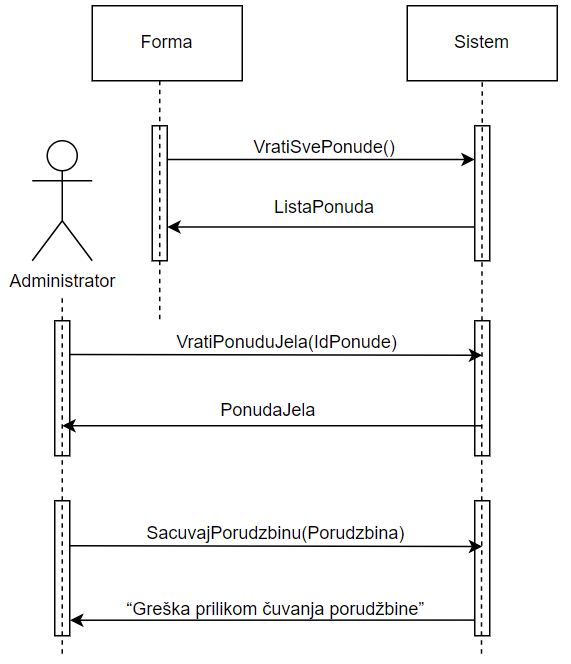
**Osnovni scenario SK:**

1. Forma **poziva** sistem da učita ponude za sve dane za koje je administrator kreirao ponude jela. (**APSO**)
2. Sistem **vraća** listu ponuda po danima. (**IA**)
3. Aktor **poziva** sistem da učita ponuđena jela za taj dan. (**APSO**)
4. Sistem **prikazuje** ponuđena jela za dan koji je aktor izabrao. (**IA**)
5. Aktor **poziva** sistem da sačuva poručena jela. (**APSO**)
6. Sistem **prikazuje** **poruku** “Porudžbina je uspešno sačuvana”. (**IA**)



**Alternativni scenariji:**

6.1. Ukoliko sistem **ne može** da sačuva poručena jela, prikazuju poruku “Greška prilikom čuvanja porudžbine”.



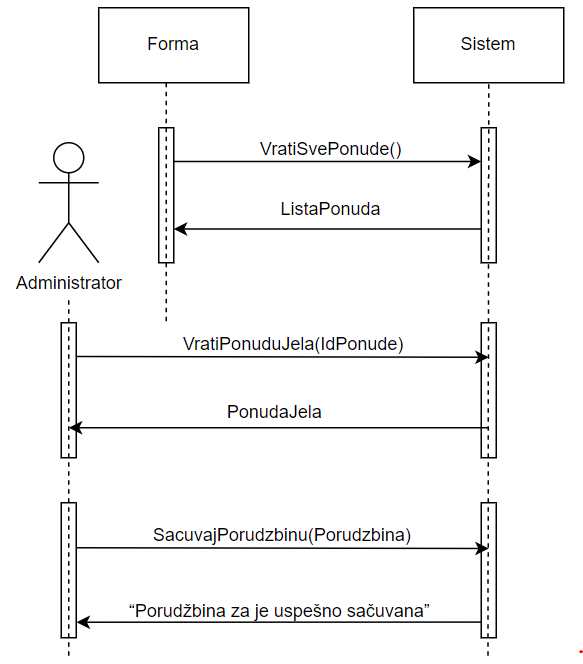
**Sa navedenih sekvencnih dijagrama uočavaju se tri sistemske operacije:**

1. **VratiSvePonude**(): ListaPonuda
2. **VratiPonuduJela**(IdPonude): PonudaJela
3. **SacuvajPorudzbinu**(Porudzbina): Signal

### 2.1.9 DS9 – Izmena porudžbine

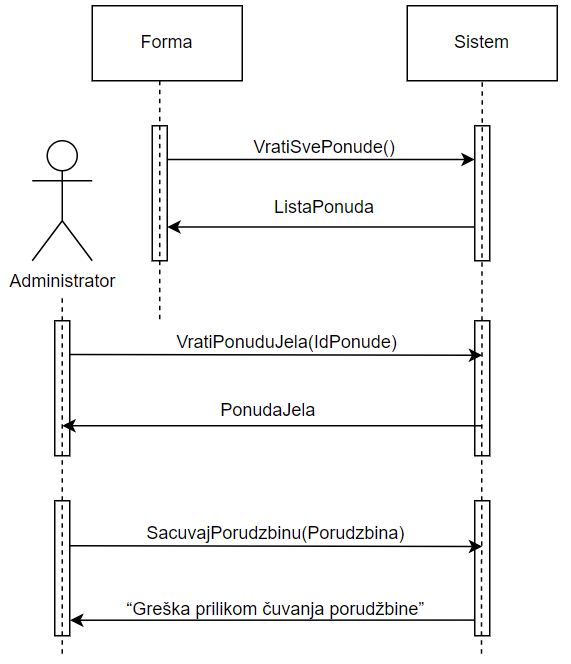
**Osnovni scenario SK:**

1. Forma **poziva** sistem da učita ponude za sve dane za koje je administrator kreirao ponude jela. (**APSO**)
2. Sistem **vraća** listu ponuda po danima. (**IA**)
3. Zaposleni **poziva** sistem da učita ponuđena jela za taj dan koji je izabrao kao i jela koja je on poručio .(**APSO**)
4. Sistem **prikazuje** poručena jela zaposlenog(**IA**)
5. Aktor **poziva** sistem da sačuva izmenjenu porudžbinu. (**APSO**)
6. Sistem **prikazuje** poruku “Porudžbina je uspešno sačuvana”. (**IA**)

****

**Alternativni scenariji:**

* 1. Ukoliko sistem **ne može** da sačuva porudžbinu prikazuju “Greška prilikom čuvanja porudžbine.”



**Sa navedenih sekvencnih dijagrama uočavaju se tri sistemske operacije:**

1. **VratiSvePonude**(): ListaPonuda
2. **VratiPonuduJela**(IdPonude): PonudaJela
3. **SacuvajPorudzbinu**(Porudzbina): Signal

Na onsovu analize scenarija dobijeno je 13 sistemskih operacija:

1. **KreirajNalog**(Nalog): Signal

2. **PrijaviKorisnika**(Nalog): Nalog

3. **DeaktivirajNalog**(Nalog): Signal

4. **SacuvajPonuduJela**(PonudaJela): Signal

5. **IzmeniPonuduJela**(PonudaJela): Signal

6. **VratiSvePonude**(): List<PonudaJela>

7. **VratiPonuduJela**(IdPonude): PonudaJela

8. **VratiZaposlene**(): List<Zaposleni>

9. **VratiJela**(): List<Jelo>

10. **SacuvajPorudzbinu**(Porudzbina): Signal

11. **VratiSvePorudzbineZaPonuduJela**(PonudaJela): List<Porudzbina>

12. **VratiPorudzbinuZaposlenog**(Zaposleni): List<Porudzbina>

13. **KreirajGrupnuPorudzbinu(**PonudaJela**):** String

### 2.2 . Ponašanje softverskog sistema - Definisanje ugovora sa sistemskim operacijama

**Ugovor UG1**: **KreirajNalog(**Nalog**): Signal**

**Veza sa SK**: SK1

**Preduslovi**: Vrednosna i strukturna ograničenja nad objektom **Nalog** moraju biti zadovoljena

**Postuslovi**: Nalog je kreiran

**Ugovor UG2**: **PrijaviKorisnika(**Nalog**): Nalog**

**Veza sa SK**: SK3

**Preduslovi**: Vrednosna i strukturna ograničenja nad objektom **Nalog** moraju biti zadovoljena

**Postuslovi**: /

**Ugovor UG3**: **DeaktivirajNalog(**Nalog**): Signal**

**Veza sa SK**: SK2

**Preduslovi**: Strukturna i vrednosna ograničenja nad objektom **Nalog** moraju biti zadovoljena

**Postuslovi**: Nalog je deaktiviran

**Ugovor UG4**: **SacuvajPonuduJela(PonudaJela): Signal**

**Veza sa SK**: SK4, SK5

**Preduslovi**: Strukturna i vrednosna ograničenja nad objektom **PonudaJela** moraju biti zadovoljena

**Postuslovi**: Ponuda je kreirana, Ponuđena jela su sačuvana

**Ugovor UG5**: **IzmeniPonuduJela(PonudaJela): Signal**

**Veza sa SK**: SK5

**Preduslovi**: Strukturna i vrednosna ograničenja nad objektom **PonudaJela** moraju biti zadovoljena

**Postuslovi**: Ponuda je izmenjena

**Ugovor UG6**: **VratiSvePonude (): List<PonudaJela>**

**Veza sa SK**: SK5, SK6, SK7, SK8, SK9

**Preduslovi**: /

**Postuslovi**: /

**Ugovor UG7:** **VratiPonuduJela(IdPonude): PonudaJela**

**Veza sa SK**: SK5, SK8, SK9

**Preduslovi**: /

**Postuslovi**: /

**Ugovor UG8**: **VratiJela(): List<Jelo>**

**Veza sa SK**: SK4, SK5

**Preduslovi**: /

**Postuslovi**: /

**Ugovor UG9**: **VratiZaposlene(): List<Zaposleni>**

**Veza sa SK**: SK1, SK2

**Preduslovi**: /

**Postuslovi**: /

**Ugovor UG10**: **SacuvajPorudzbinu**(Porudzbina): Signal

**Veza sa SK**: SK8, SK9

**Preduslovi**: Strukturna i vrednosna ograničenja nad objektima **PoručenoJelo** moraju biti zadovoljena

**Postuslovi**: Poručena jela su kreirana

**Ugovor UG11**: **VratiSvePorudzbineZaPonuduJela**(PonudaJela): List<Porudzbina>

**Veza sa SK**: SK6, SK7

**Preduslovi**: /

**Postuslovi**: /

**Ugovor UG12**: **VratiPorudzbinuZaposlenog**(Zaposleni): List<Porudzbina>

**Veza sa SK**: SK9

**Preduslovi**: /

**Postuslovi**: /

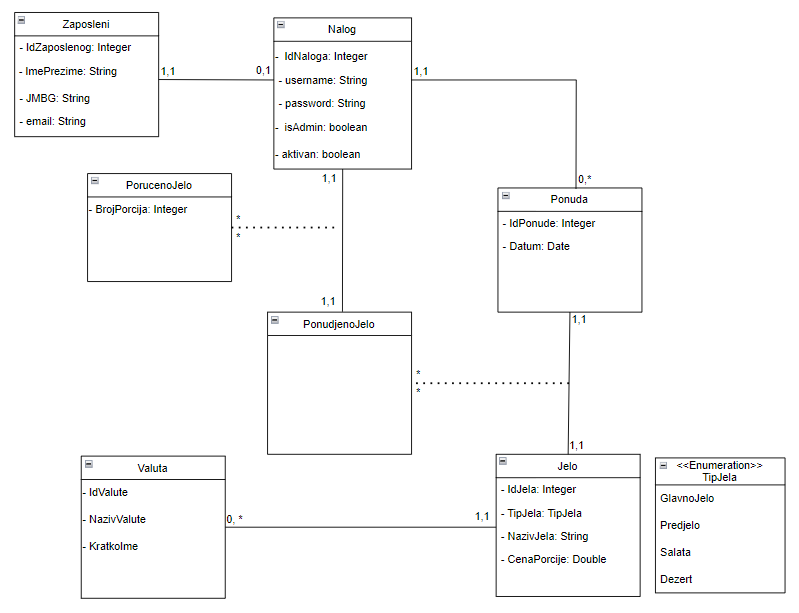
**Ugovor UG13**: **KreirajGrupnuPorudzbinu**(PonudaJela): String

**Veza sa SK**: SK7

**Preduslovi**: /

**Postuslovi**: /

### 2.3. Sturktura softverskog sistema - Konceptualni (Domenski) model



### 2.4. Struktura softverskog sistema – Relacioni model

**Ponuda**(**IdPonude**, Datum, *IdNaloga*)

**Jelo**(**IdJela**, TipJela, NazivJela, CenaPorcije, *IdValute*)

**PonudjenoJelo**(***IdPonude***, ***IdJela***)

**PorucenoJelo**(***IdPonude***, ***IdJela***, ***IdNaloga,*** *BrojPorcija*)

**Zaposleni**(**IdZaposlenog**, ImePrezime, JMBG, email)

**Nalog**(**IdNaloga**, username, password, isAdmin, aktivan, *IdZaposlenog*)

**Valuta**(**IdValute**, NazivValute, KratkoIme)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela **Ponuda** | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  Restricted Nalog  UPDATE  Restricted Nalog,  Cascades PonudjenoJelo  DELETE  Restricted PonudjenoJelo |
| IdPonude | int | Not null and >0 |  |  |
| Datum | date | Not null | Unique |  |
| IdNaloga | int | Not null and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela **Jelo** | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  Restricted Valuta  UPDATE  Restricted  Valuta,  Cascades PonudjenoJelo  DELETE  Restricted PonudjenoJelo |
| TipJela | TipJela | In (“Glavno Jelo”, “Predjelo”, “Salata”, “Dezert”) |  |  |
| NazivJela | string | Not null |  |  |
| CenaPorcije | Double | Not null and >0 |  |  |
| IdValute | int | Not null and >0 |  |  |

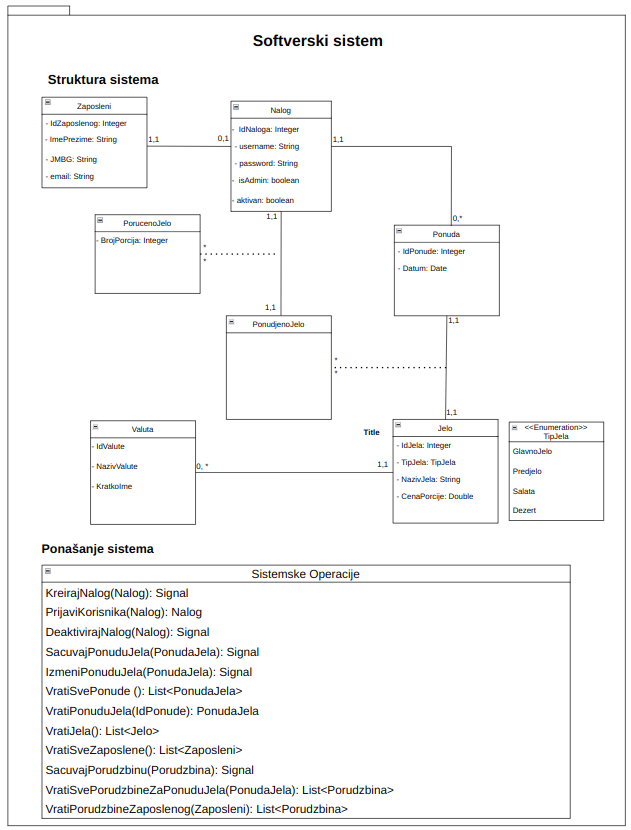
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela **PonudjenoJelo** | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  Restricted Ponuda, Jelo  UPDATE  Restricted  Ponuda, Jelo, Cascades PorucenoJelo  DELETE  Restricted PorucenoJelo |
| IdPonude | int | Not null and >0 |  |  |
| IdJela | int | Not null and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela **PorucenoJelo** | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  Restricted PonudjenoJelo, Nalog  UPDATE  Restricted PonudjenoJelo, Nalog  DELETE  / |
| IdPonude | int | Not null and >0 |  |  |
| IdJela | int | Not null and >0 |  |  |
| IdNaloga | int | Not null and >0 |  |  |
| BrojPorcija | Int | Not nul and > 0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela **Zaposleni** | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  /  UPDATE  Cascades Nalog  DELETE  Restricted Nalog |
| IdZaposlenog | Int | Not null and >0 and Unique |  |  |
| ImePrezime | string | Not null |  |  |
| JMBG | string | Not null and Unique |  |  |
| Email | string | Not null |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela **Nalog** | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  Restricted Zaposleni  UPDATE  Restricted Zaposleni,  Cascades Ponuda, PorucenoJelo  DELETE  Restricted Ponuda, PorucenoJelo |
| IdNaloga | Int | Not null and >0 and Unique |  |  |
| username | string | Not null |  |  |
| password | string | Not null and Unique |  |  |
| IsAdmin | boolean | Not null |  |  |
| Aktivan | boolean | Not null |  |  |
| IdZaposlenog | Int | Not null and >0 |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabela **Valuta** | | Prosto vrednosno ograničenje | | Složeno vrednosno ograničenje | | Strukturno ograničenje |
| Atributi | Ime | Tip atributa | Vrednost atributa | Međuzavisnost atributa jedne tabele | Međuzavisnost atributa više tabela | INSERT  /  UPDATE  Cascades Jelo  DELETE  Restricted Jelo |
| IdValute | Int | Not null and >0 |  |  |
| NazivValute | string | Not null |  |  |

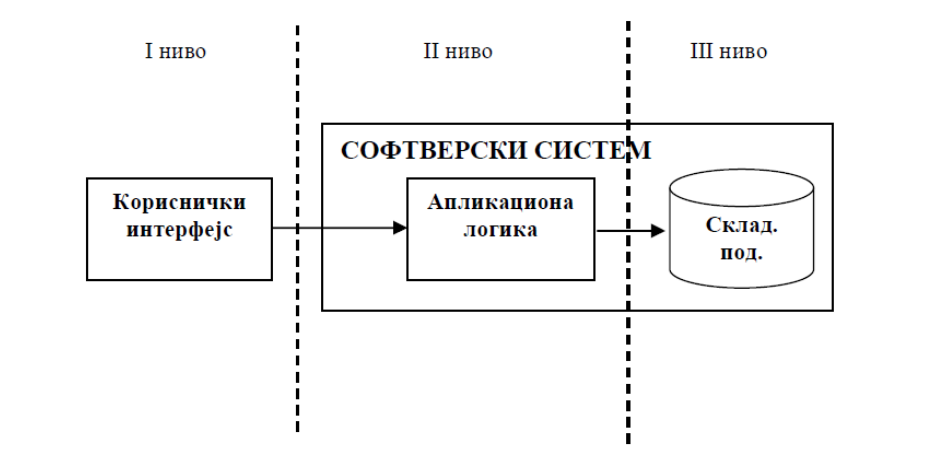
Rezultat faze analize su struktura softverskog sistema opisana preko dijagrama klasa i relacionog modela, i ponašanje softverskog sistema opisano na osnovu dijagrama sekvenci i sistemskih operacija. 

# 3. Projektovanje

U fazi projektovanja opisana je fizička struktura softverskog sistema, odnosno arhitektura u kojoj će sistem biti izrađen.

U ovom seminarskom radu koristi se tronivojska arhitektura softverskog sistema u kojoj se sistem sastoji iz 3 nivoa:

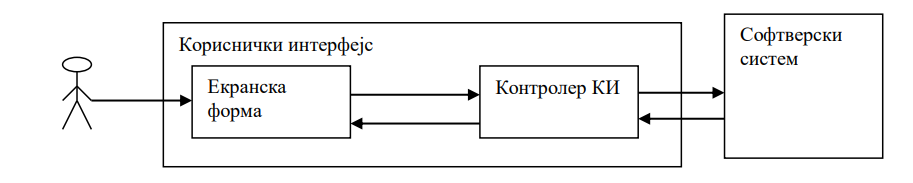
1. Nivo korisničkog interfejsa
2. Nivo aplikacione logike
3. Nivo skladišta podataka



## Projektovanje korisničkog interfejsa

Korisnički interfejs je deo sistema preko kojeg korisnik interaguje sa sistemom, generiše događaje, poziva akcije, unosi podatke i takođe dobija odgovore od sistema.

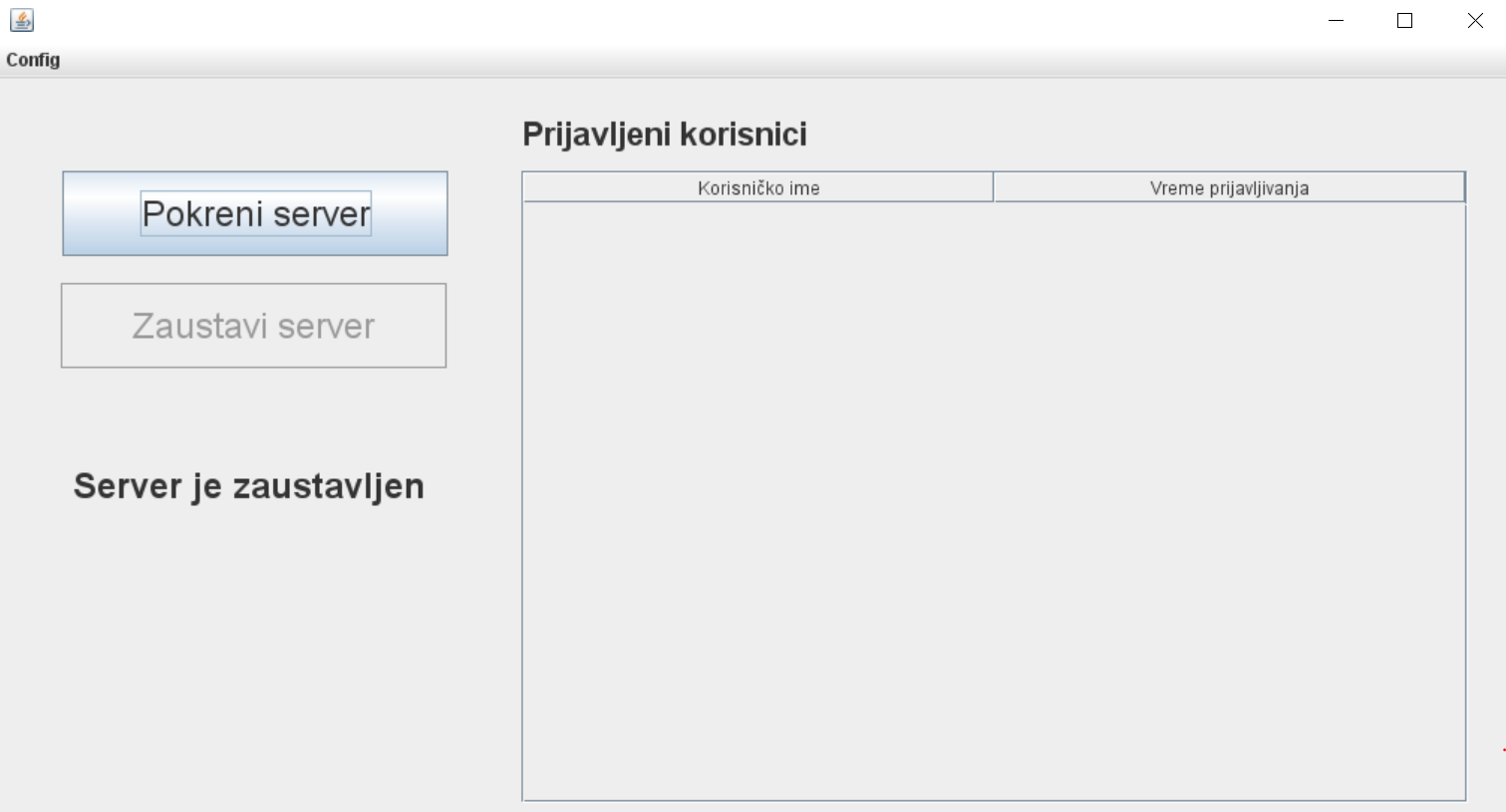
Korisnički interfejs se sastoji od kontrolera korisničkog interfejsa i skupa ekranskih formi.



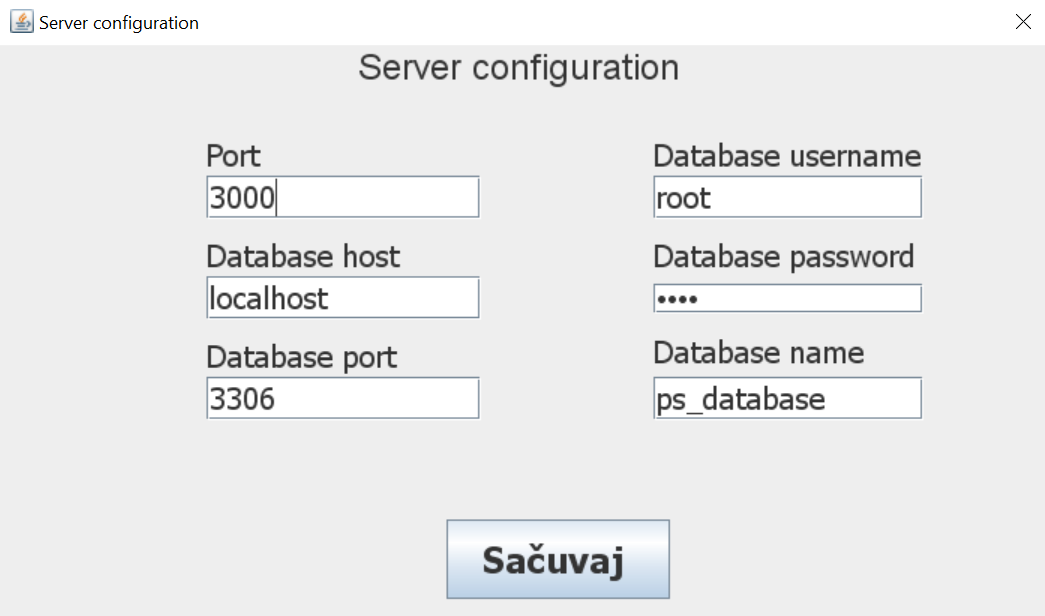
### 3.1.1 Projektovanje ekranskih formi

Scenarija korišćenja ekranskih formi direktno su povezana sa scenarijima slučajeva korisčćenja.

Za upravljanje i konfiguraciju servera, na serverskoj strani obezbeđena je forma koja izgleda ovako:



Klikom na stavku menija **Config** otvara se dijalog za podešavanje osnovnih kofiguracionih parametara servera.



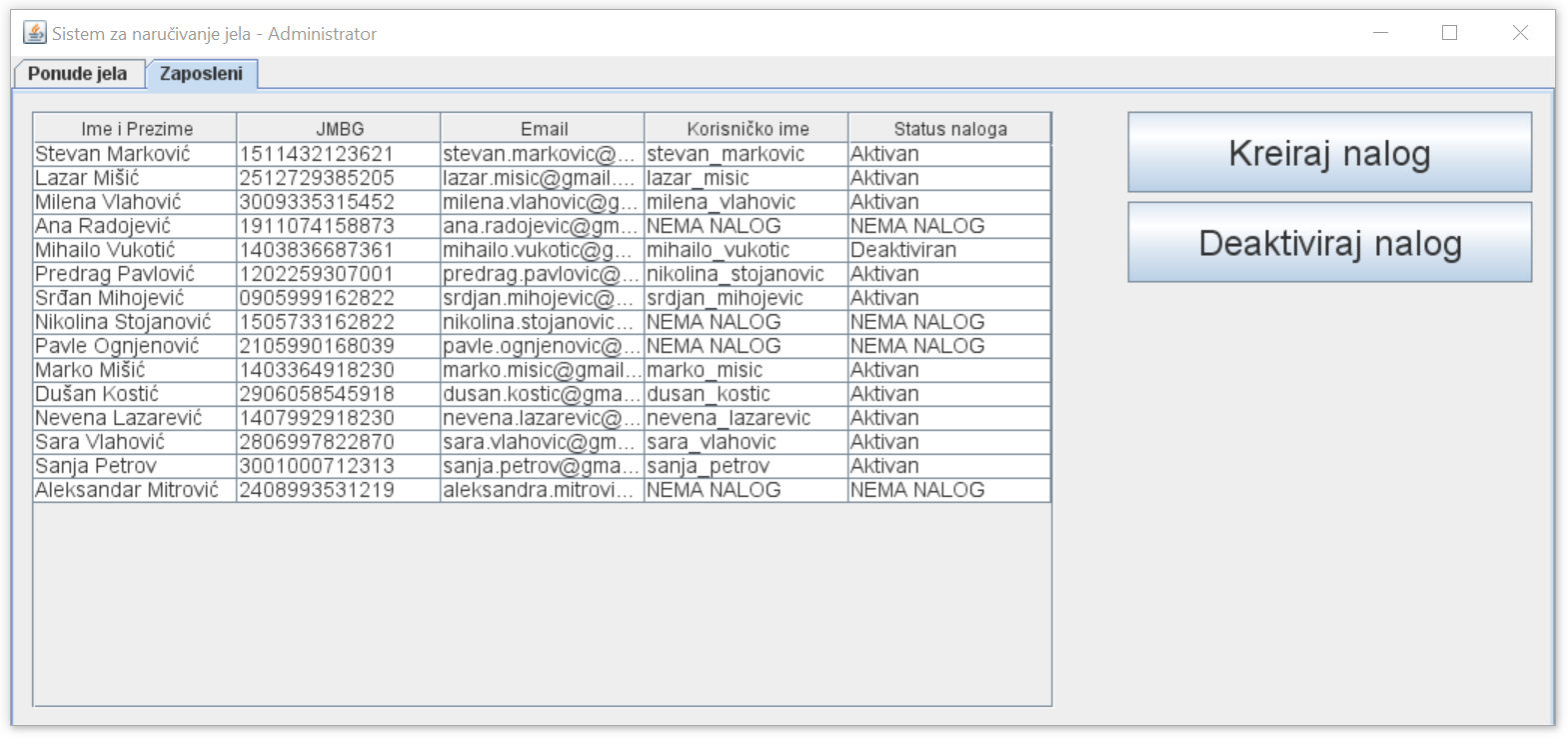
SK1 – Kreiranje naloga zaposlenog

**Naziv SK:** Kreiranje naloga zaposlenog

**Aktori SK:** Administrator

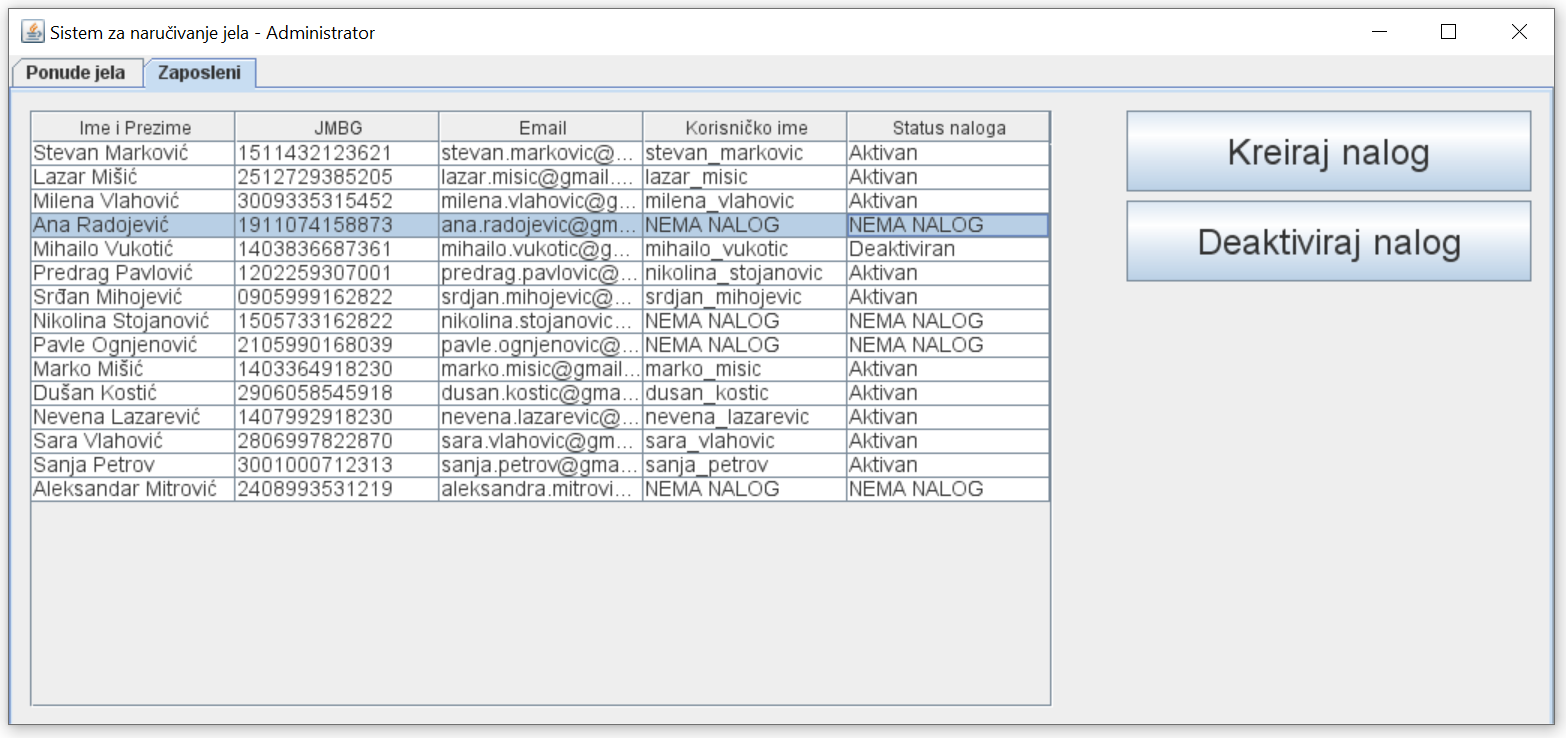
**Učesnici SK:** Administrator i sistem

**Preduslov:** Sistem je pokrenut, Administrator je ulogovan na svoj nalog, prikazana je lista svih zaposlenih u firmi.

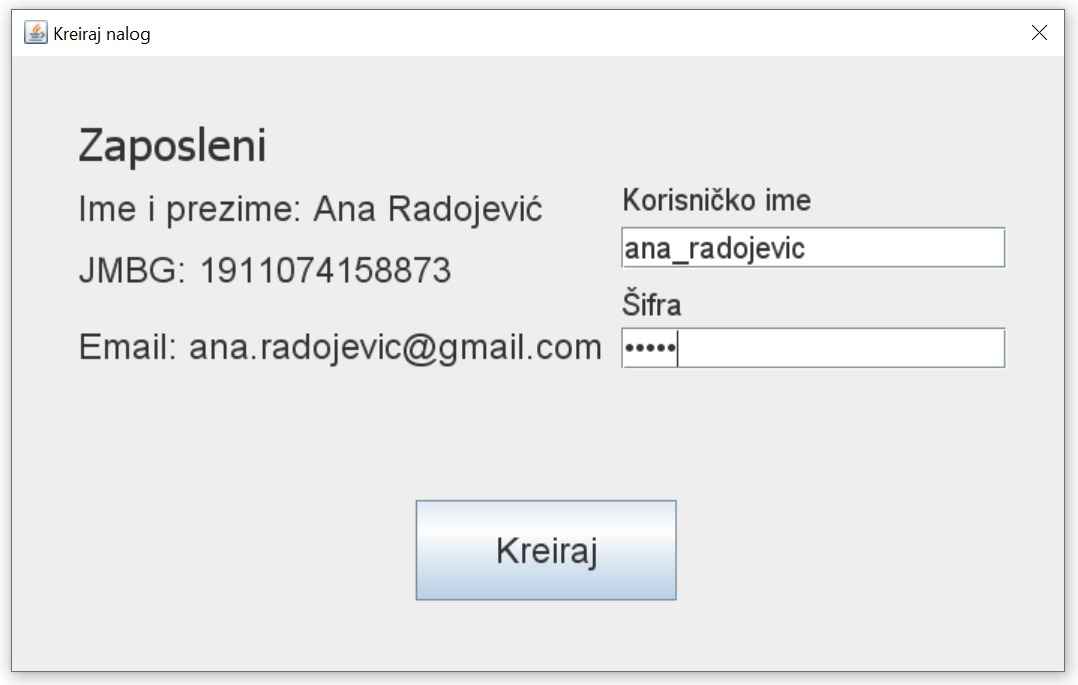


**Osnovni scenario SK:**

1. Administrator **bira** zaposlenog kojem želi da kreira nalog. (**APUSO**)



1. Administrator **unosi podatke** o nalogu zaposlenog (**korisničko** **ime** i **šifru**) u prazna polja. (**APUSO**)



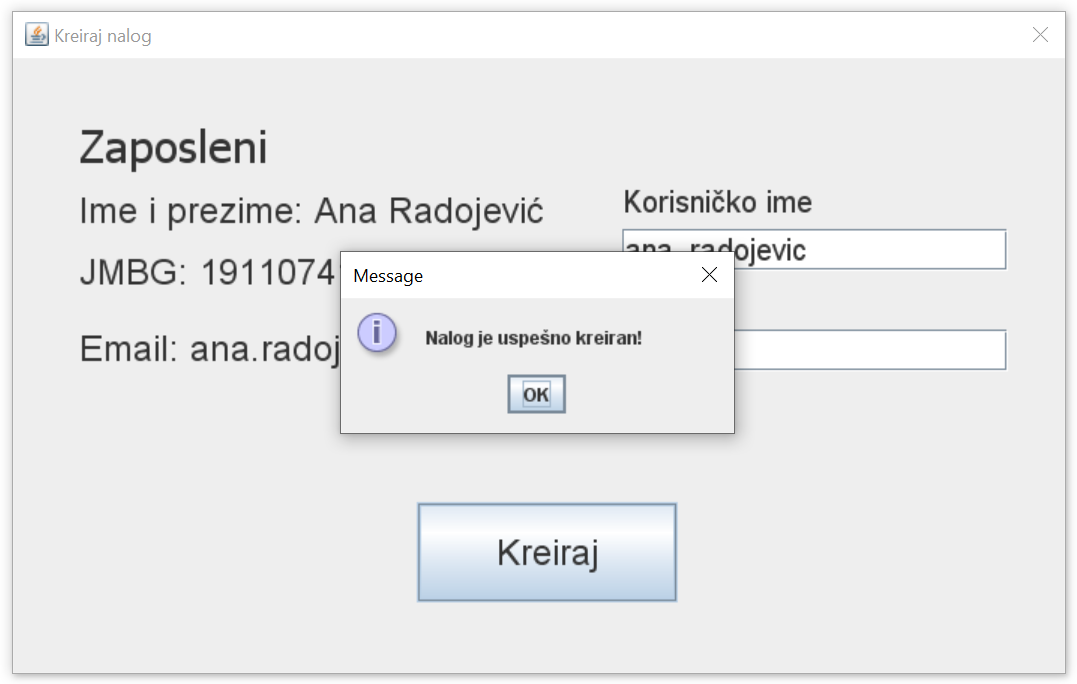
3. Administrator **kontroliše** da li je korektno uneo podatke o nalogu zaposlenog. (**ANSO**)

4. Administrator poziva sistem da **zapamti** podatke o nalogu zaposlenog. (**APSO**)

Opis akcije: Administrator klikom na dugme kreiraj poziva sistemsku operaciju KreirajNalog(Nalog)

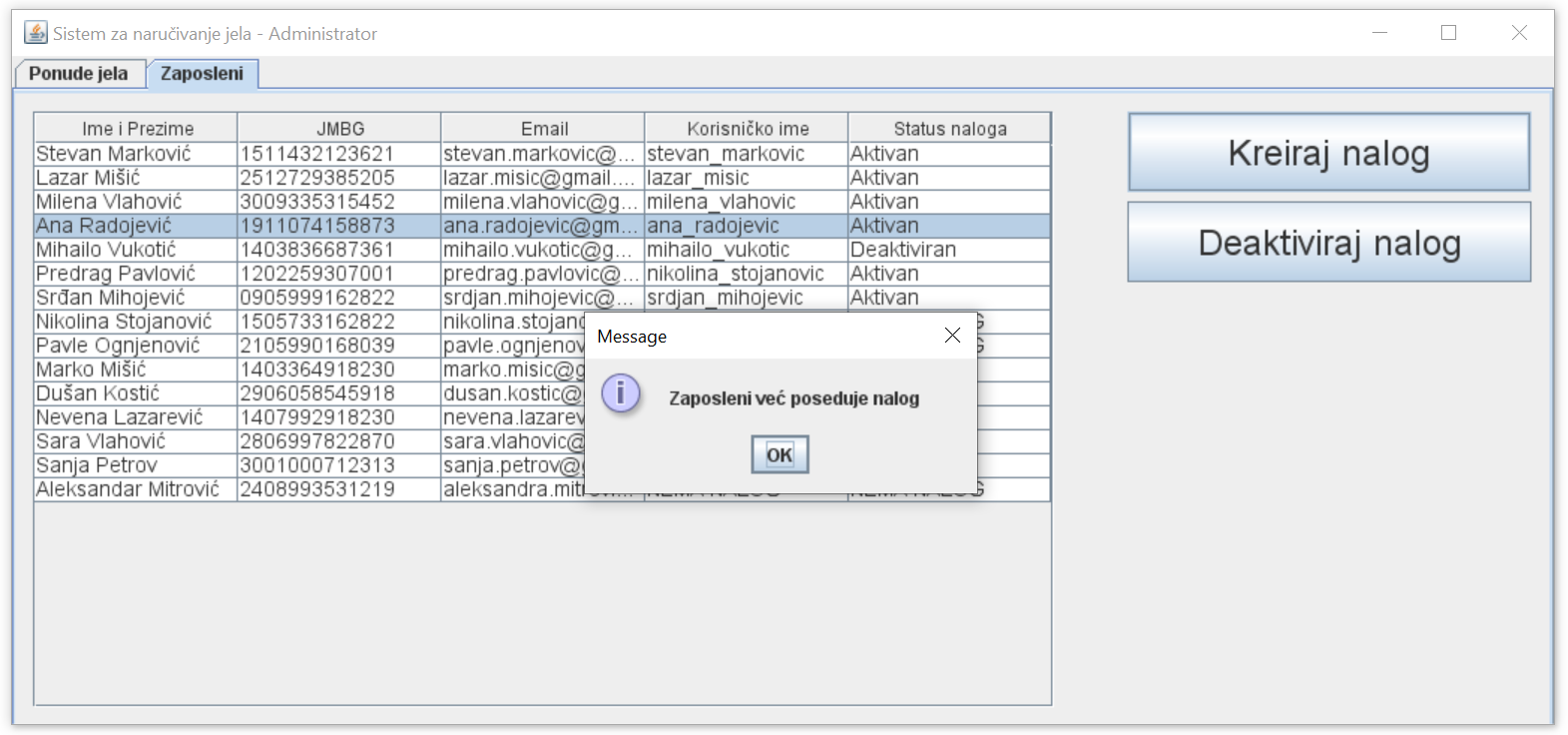
5. Sistem **pamti** podatke o nalogu zaposlenog. (**SO**)

6. Sistem **prikazuje** adminstratoru poruku: “Nalo je uspešno kreiran “. (**IA**)

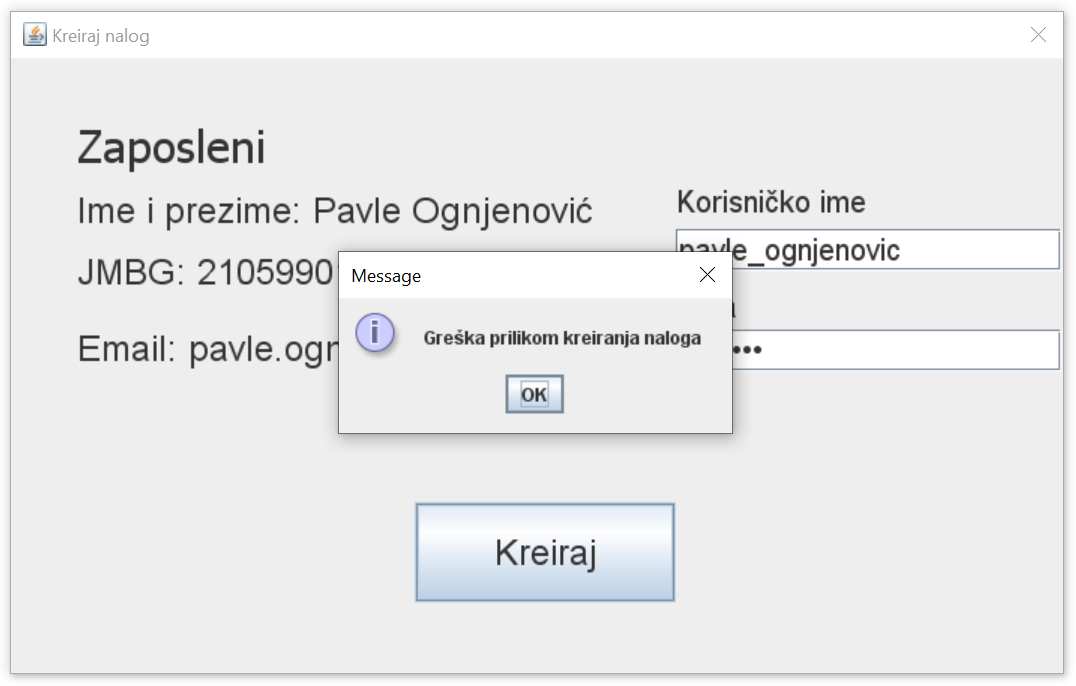


**Alternativni scenariji:**

* 1. Ukoliko je aktor izabrao zaposlenog koji već poseduje nalog prikazaće mu se poruka “**Zaposleni već poseduje nalog**”(



6.2. Ukoliko sistem ne može da kreira naog zaposlenog prikazaće poruku “**Greška prilikom kreiranja naloga**”.(**IA**)



SK2 – Deaktivacija naloga zaposlenog

**Naziv SK: Deaktivacija naloga zaposlenog**

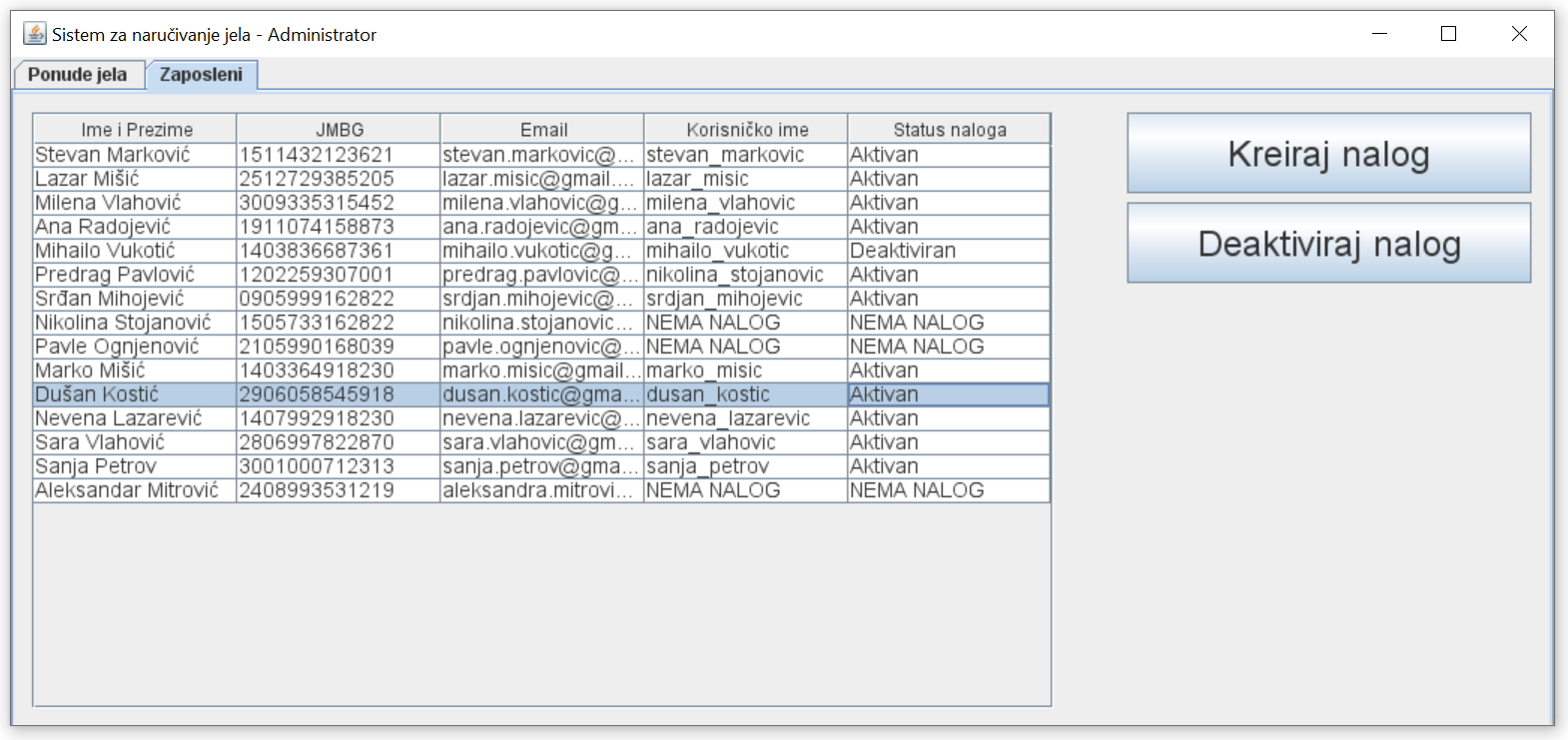
**Aktori SK:** Administrator

**Učesnici SK:** Administrator, sistem

**Preduslov:** Sistem je pokrenut, administrator je ulogovan na svoj nalog, učitana je lista na kojoj su izlistani svi zaposleni.

**Osnovni scenario SK:**

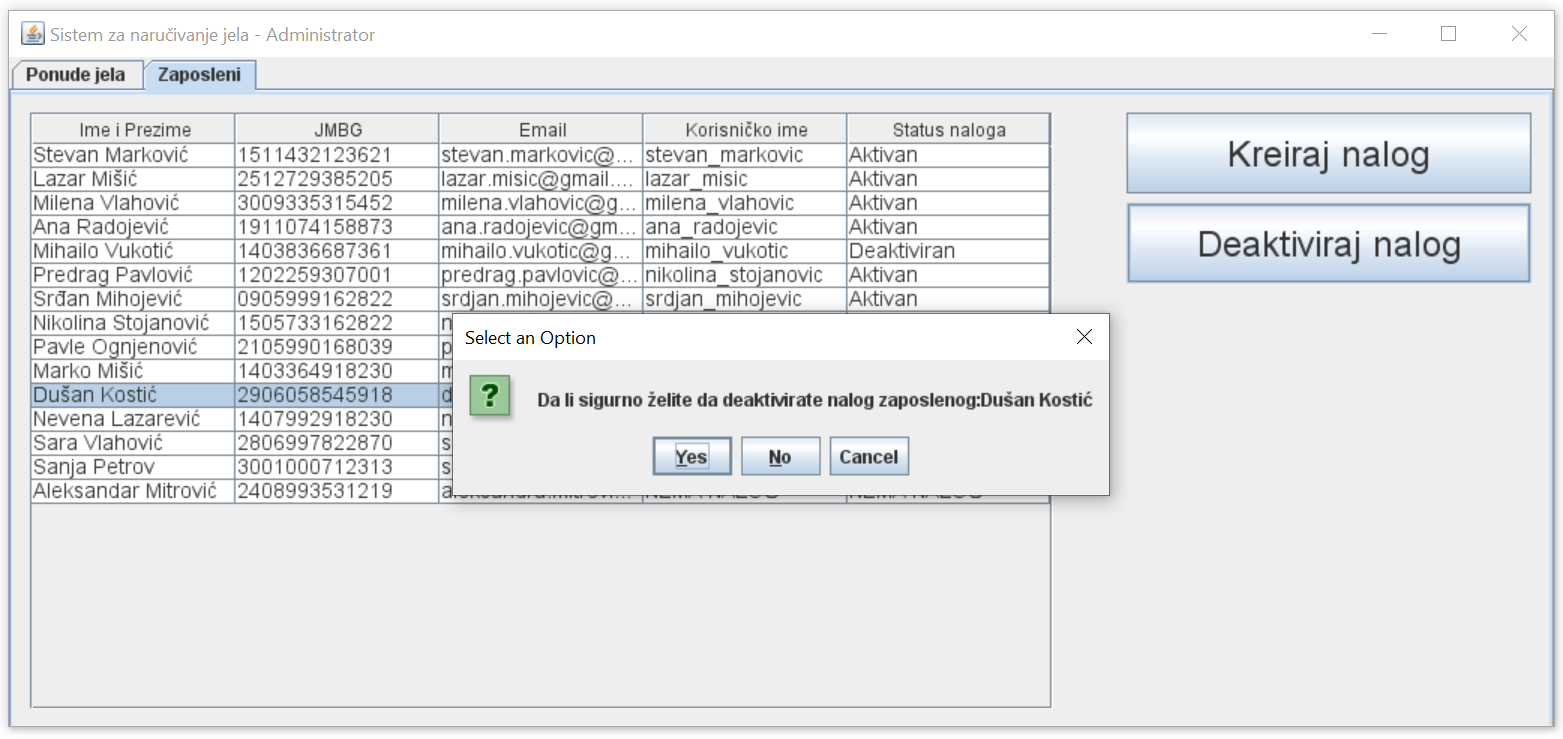
1. Administrator **bira zaposlenog** čiji nalog želi da deaktivira. (**APUSO**)



1. Administrator **poziva** sistem da deaktivira nalog zaposlenog. (**APSO**)

Opis akcije: Administrator pritiska dugme deaktiviraj nalog, čime se poziva sistemska operacija DeaktivirajNalog(Nalog)

1. Sistem traži od aktora **da potvrdi** poziv za deaktivacijom naloga (**APUSO**)

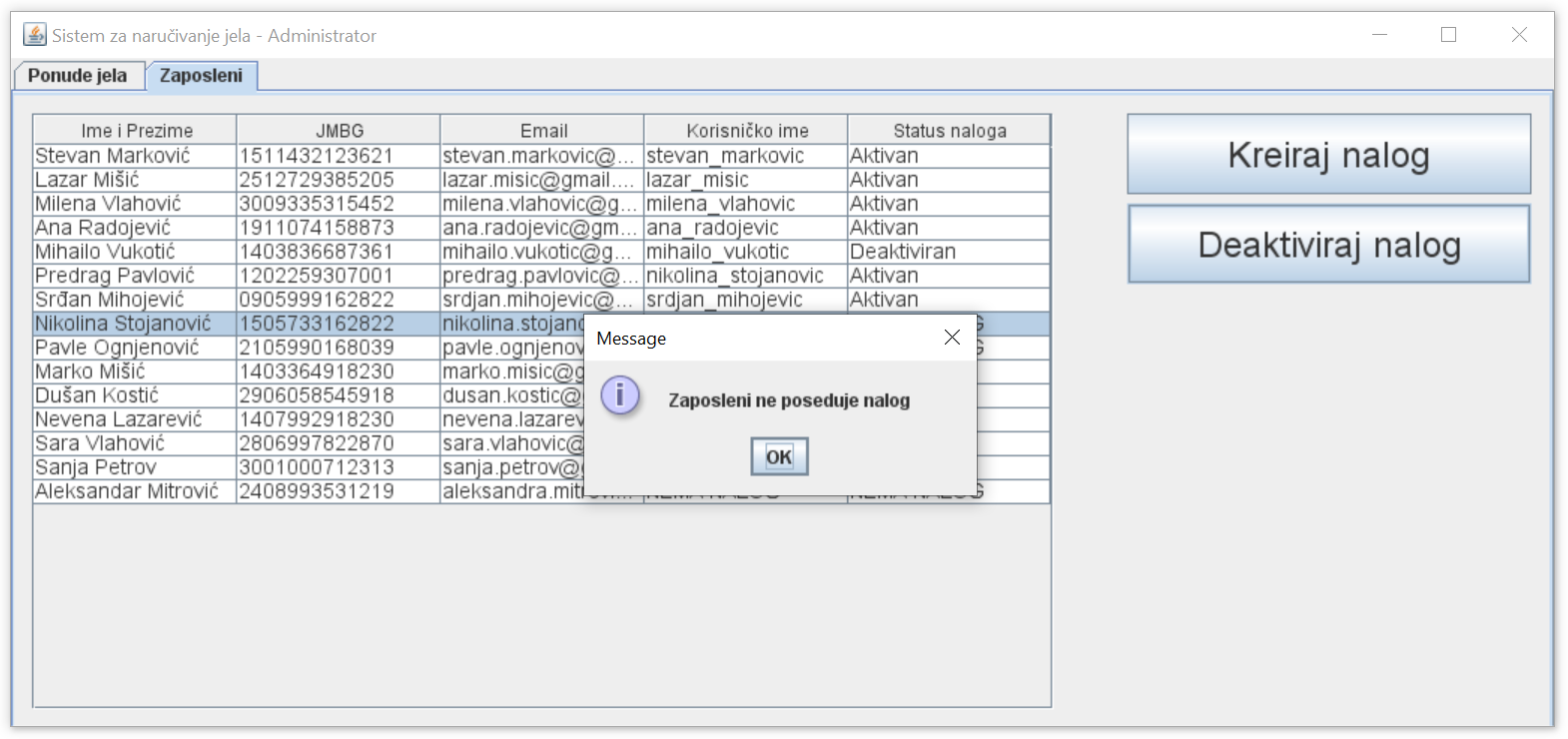
****

4. Sistem **deaktivira** nalog zaposlenog i **deaktivira** sve buduće porudžbine zaposlenog. (**SO**)

5. Sistem **prikazuje poruku** “Nalog je uspešno deaktiviran”. (**IA**)

**Alternativni scenariji:**

2.1. Ako zaposleni nema nalog prikazuje se poruka “**Zaposleni ne poseduje nalog.**” (**IA**)



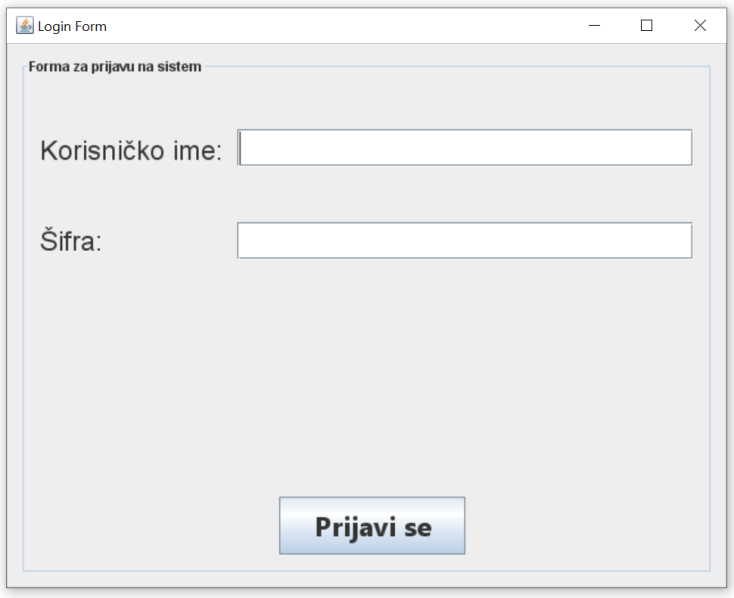
### SK3 – Prijavljivanje korisnika

**Naziv SK:** Prijavljivanje korisnika

**Aktori SK:** Administrator, zaposleni

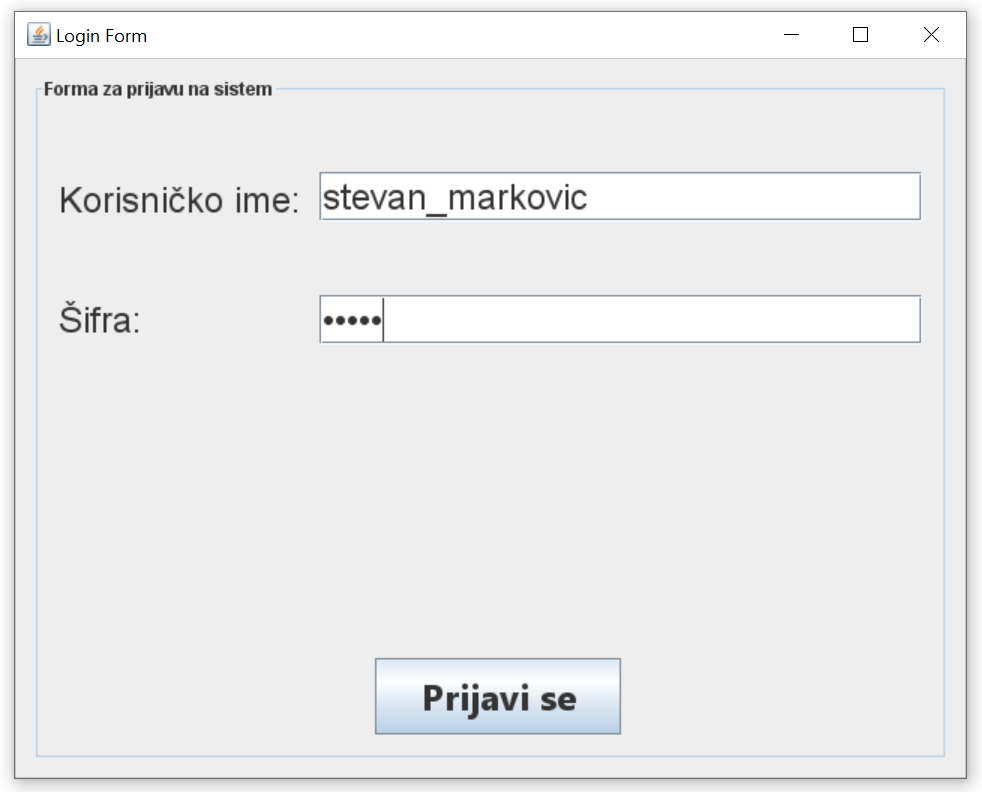
**Učesnici SK:** Administrator, zaposleni, sistem

**Preduslov:** Sistem je pokrenut, prikazana je forma za unos korisničkog imena i lozinke

****

**Osnovni scenario SK:**

1. Aktor **unosi** korisničko ime i šifru. (**APUSO**)



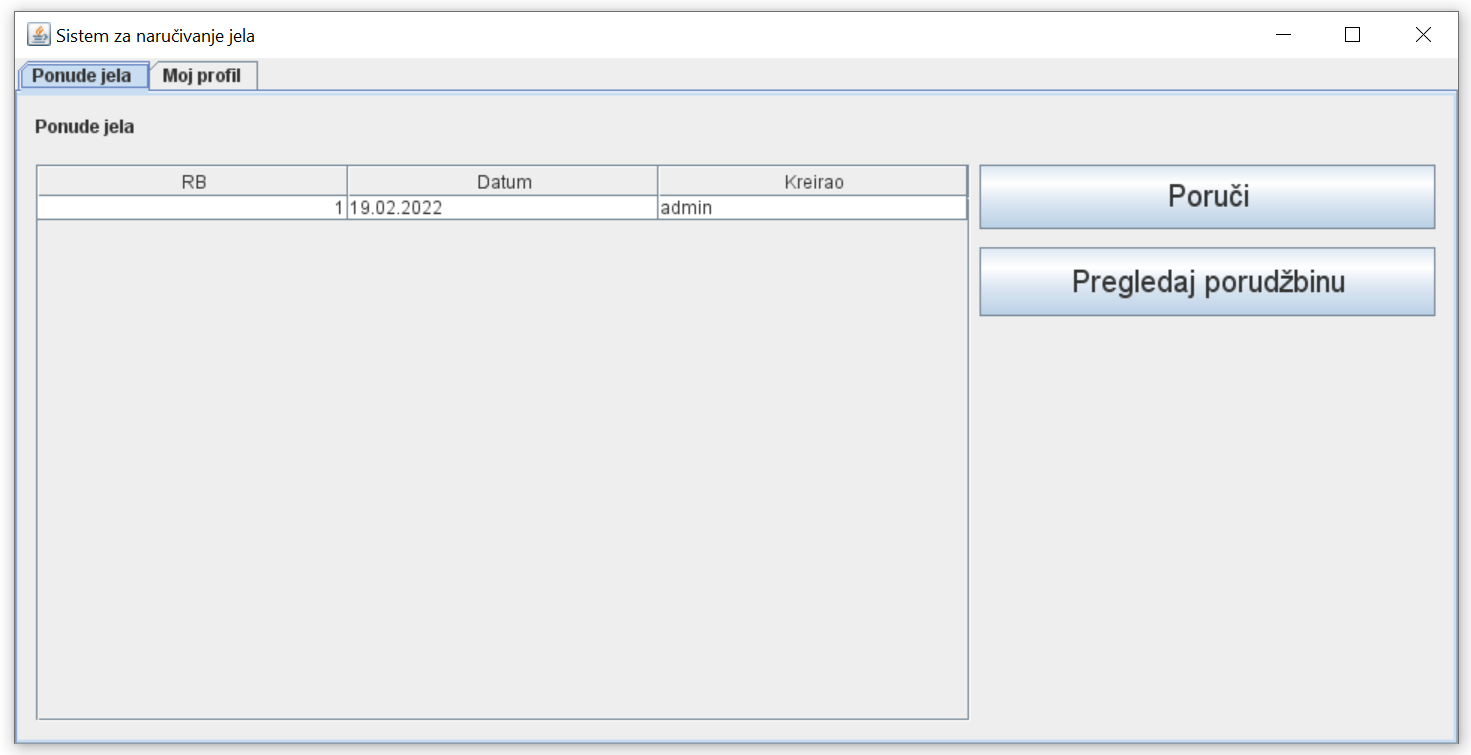
2. Aktor **kontroliše** da li je korektno uneo korisničko ime i lozinku. (**ANSO**)

3. Aktor **poziva sistem** da ga prijavi. (**APSO**)

Opis akcije: Korisnik pritiskom na dugme Prijavi se poziva sistemsku operaciju PrijaviKorisnika(Nalog)

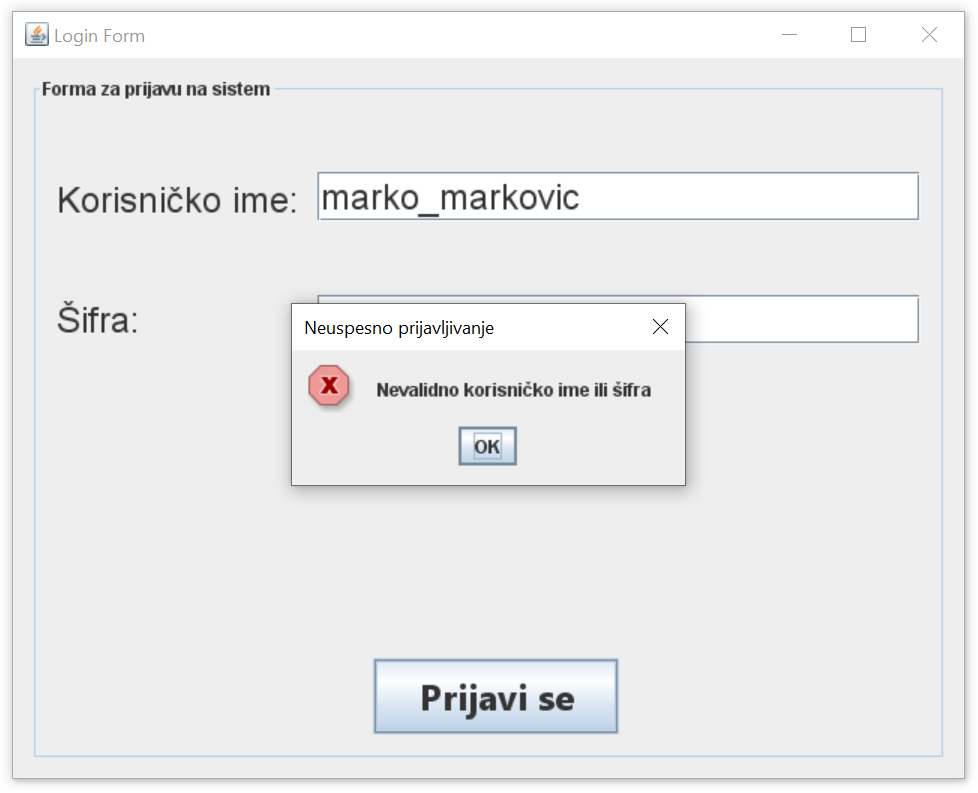
4. Sistem **prijavljuje** korisnika. (**SO**)

5. Sistem **prikazuje** korisniku glavnu formu. (**IA**)

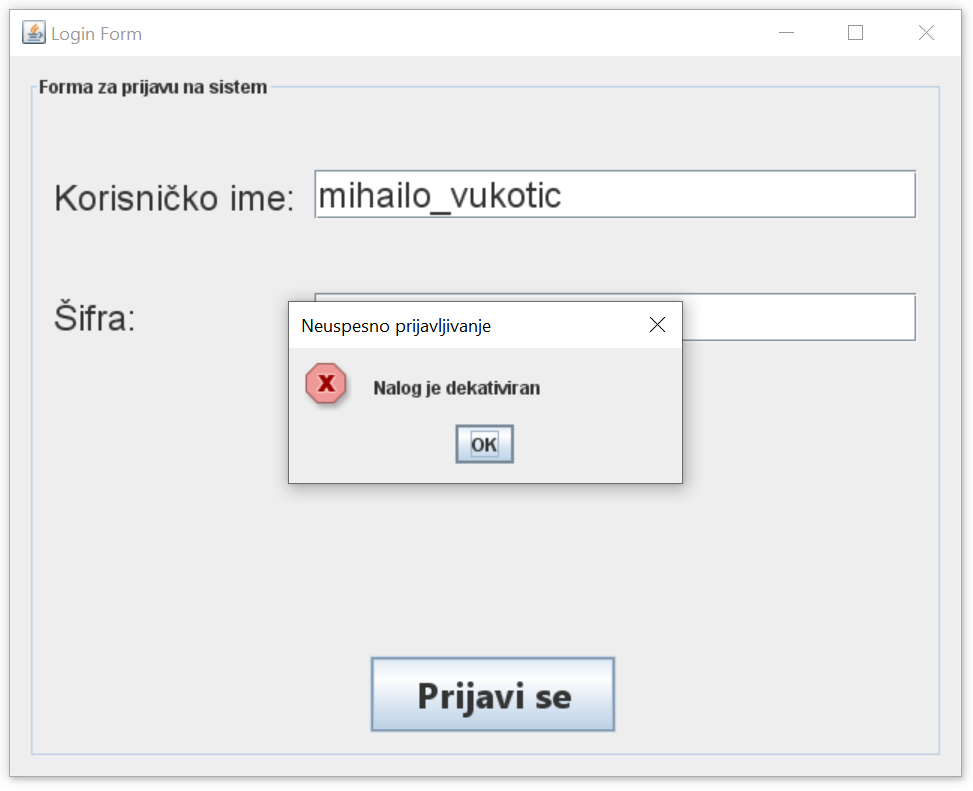
****

**Alternativni scenariji:**

5.1. Ako sistem **ne pronađe** korisnika sa datim korisničkim imenom prikazuje se poruka “Nevalidno korisničko ime ili šifra”. (**IA**)



5.2. Ako je nalog korisnika koji želi da se prijavi **deaktiviran** prikazuje se poruka “Nalog je deaktiviran”.



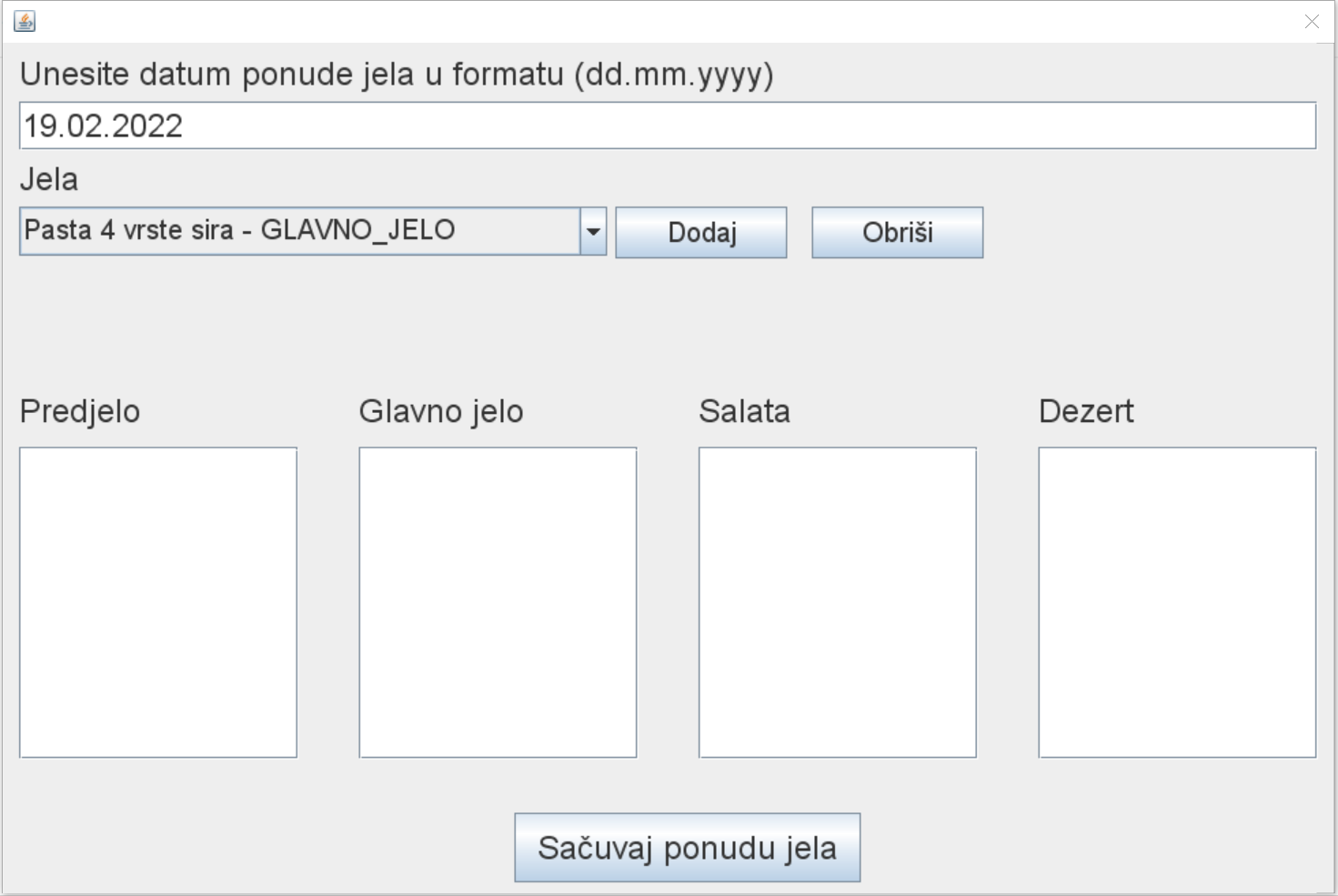
### SK4 – Kreiranje ponude jela (složen)

**Naziv SK:** Kreiranje ponude jela

**Aktori SK:** Administrator

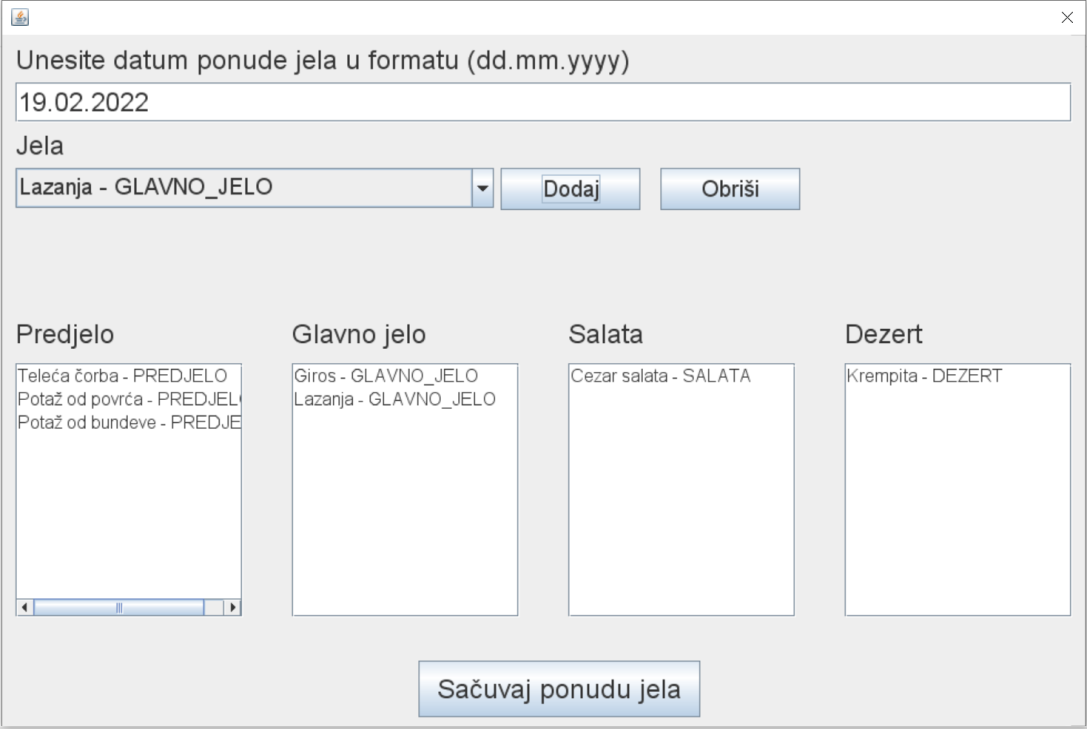
**Učesnici SK:** Administrator, sistem

**Preduslov:** Sistem je pokrenut, administrator je ulogovan na svoj nalog, otvorena je forma za kreiranje ponude.

****

**Osnovni scenario SK:**

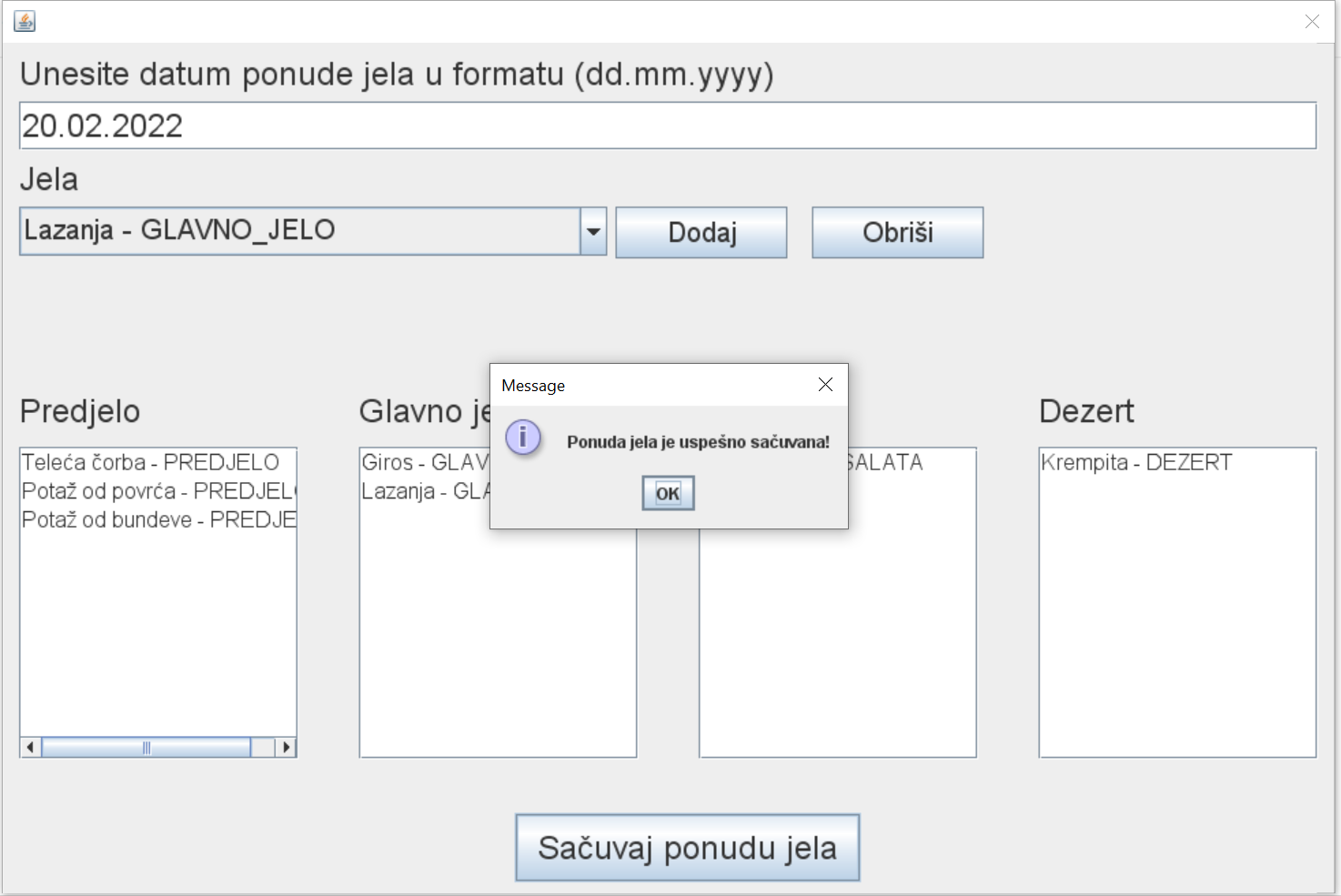
1. Administrator **bira** dan za koji hoće da kreira ponudu jela. (**APUSO**)
2. Administrator **vrši izbor** jela koje će zaposleni moći da biraju za taj dan. (**APUSO**)



1. Administrator kontroliše da li je pravilno popunio formu (**ANSO**)
2. Administrator **poziva** sistem da sačuva ponudu jela. (**APSO**)

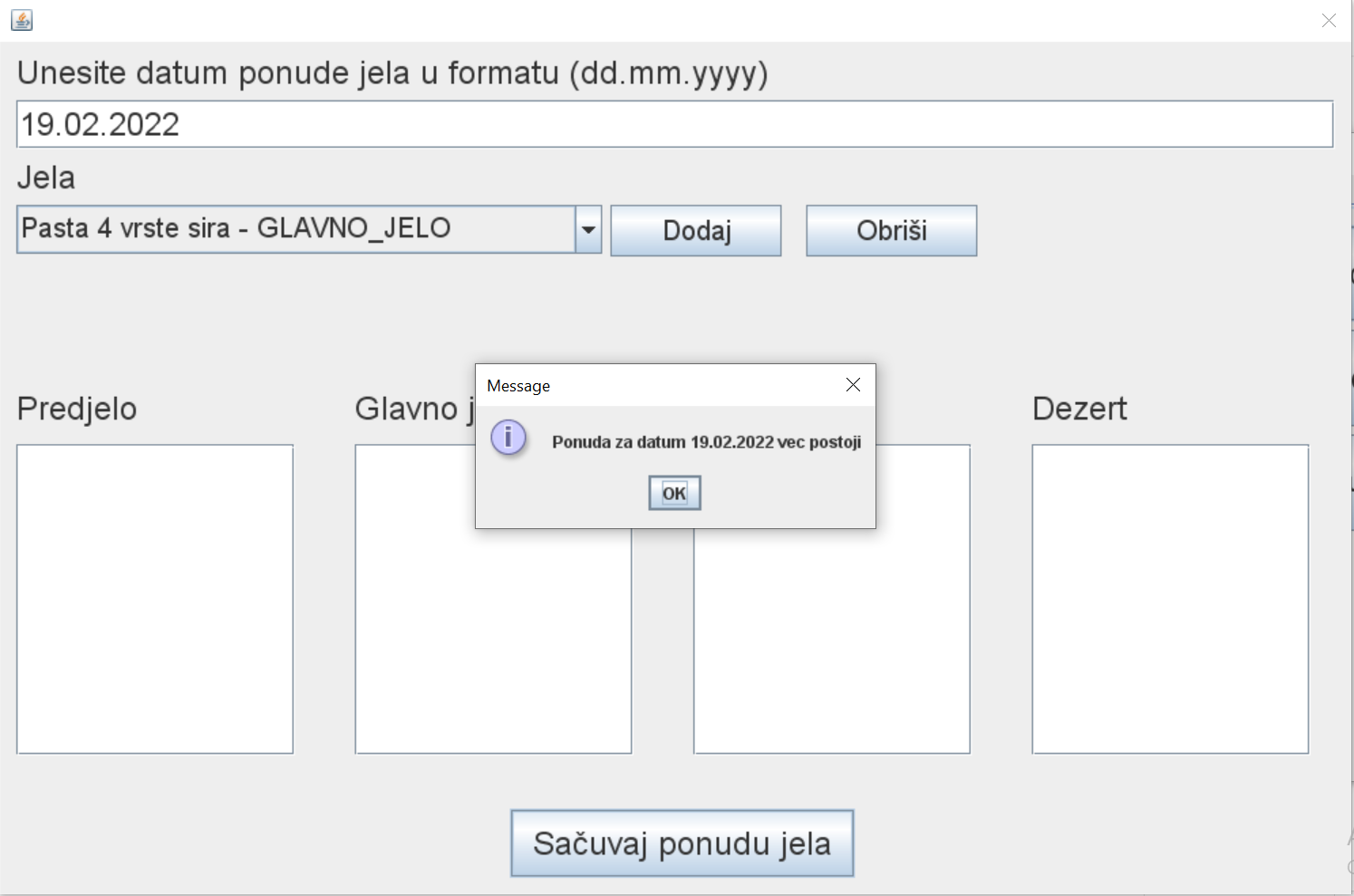
Opis akcije: Administrator pritiskom na dugme Sačuvaj ponudu jela poziva sistemsku operaciju SacuvajPonuduJela(PonudaJela): Signal

1. Sistem **pamti** ponudu jela (**SO**)
2. Sistem **prikazuje** poruku “Ponuda jela je uspešno sačuvana”.(**IA**)

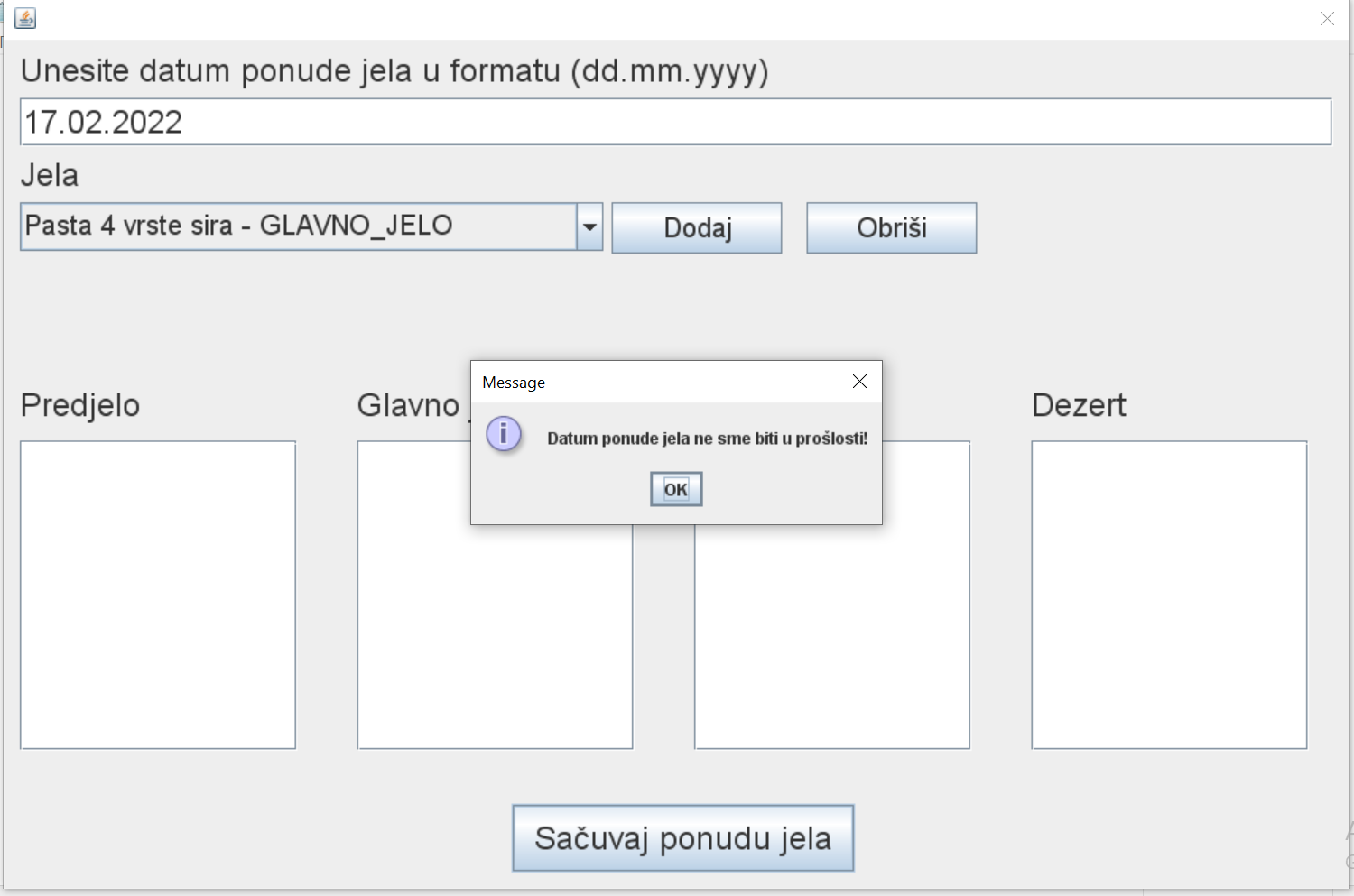
****

**Alternativni scenariji:**

6.1. Ukoliko je administrator izabrao dan za koje već postoji ponuda jela sistem prikazuje poruku: “Ponuda za dan već postoji”. (**IA**)



* 1. . Ukoliko je korisnik izabrao za datum ponude neki dan iz prošlosti prikazuje se poruka: “Datum ponude ne sme biti u prošlosti”.(**IA**)



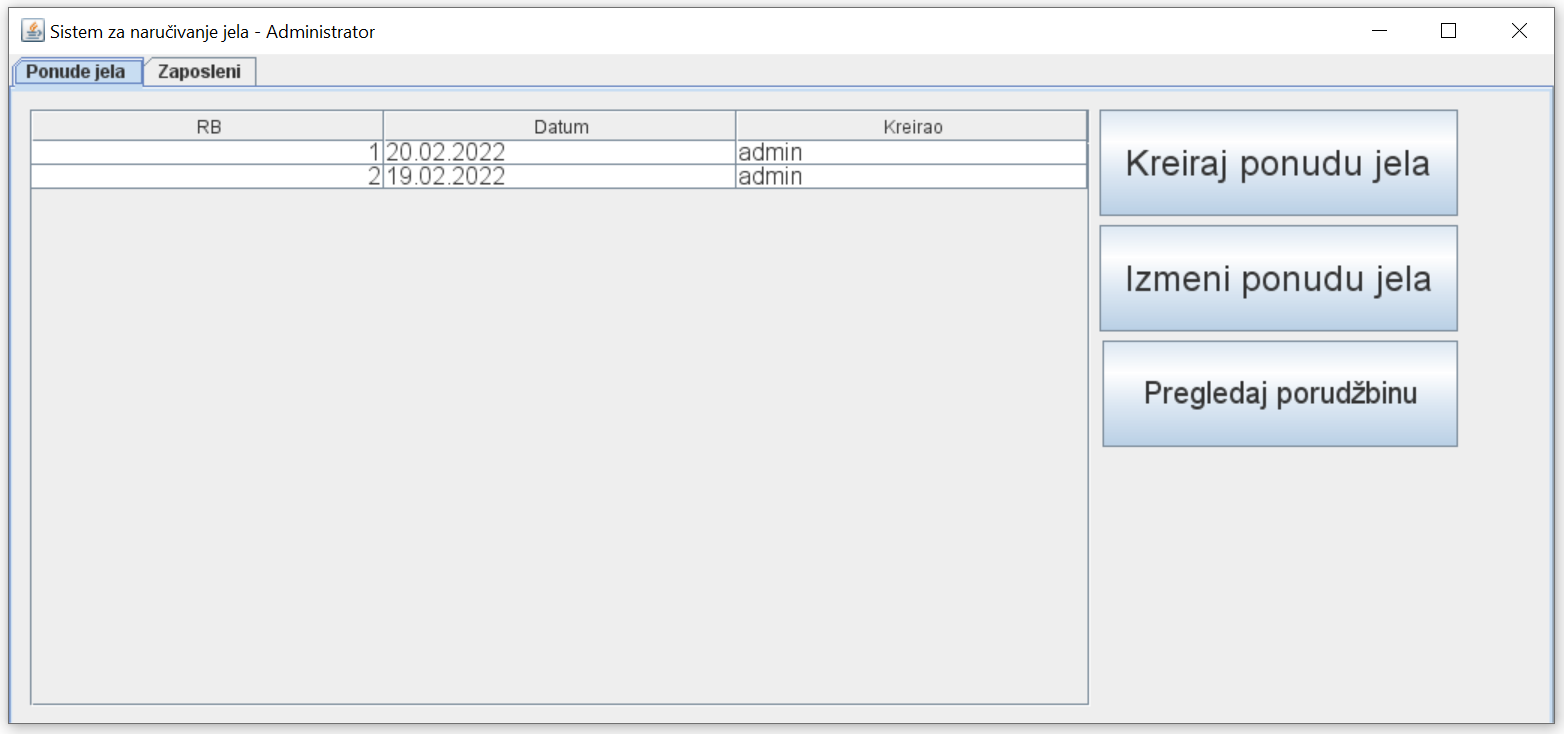
### SK5 – Izmena ponude jela (složen)

**Naziv SK:** Izmena ponude jela

**Aktori SK:** Administrator

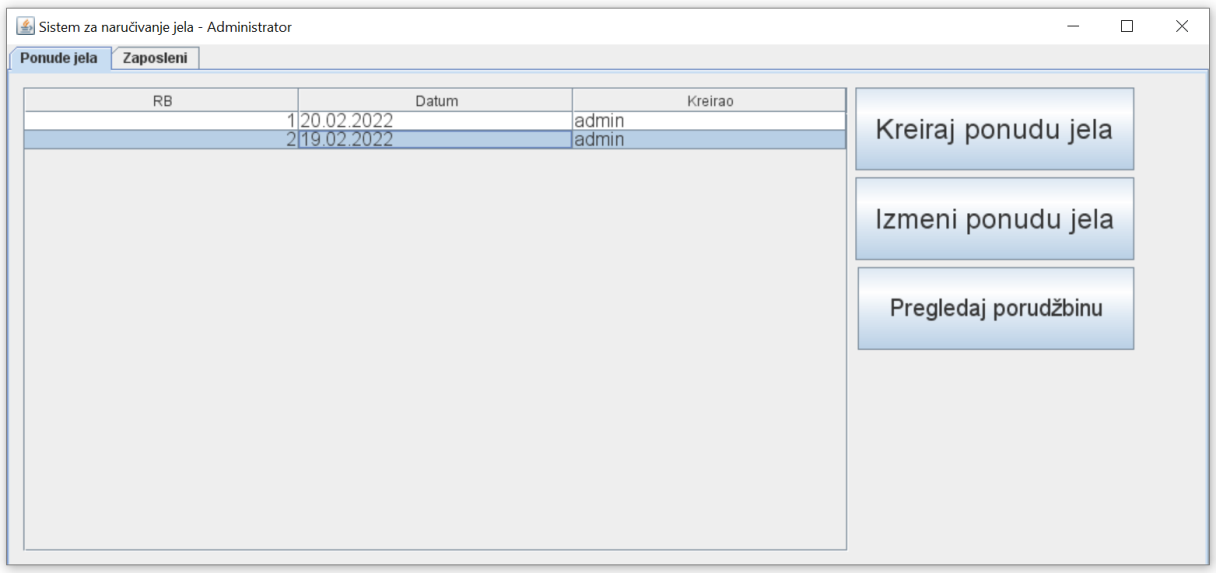
**Učesnici SK:** Administrator, sistem

**Preduslov:** Sistem je pokrenut, administrator je ulogovan na svoj nalog, učitana je lista svih dana za koje je kreirana ponuda.

****

**Osnovni scenario SK:**

1. Administrator **bira** dan za koju želi da promeni ponudu. (**APUSO**)



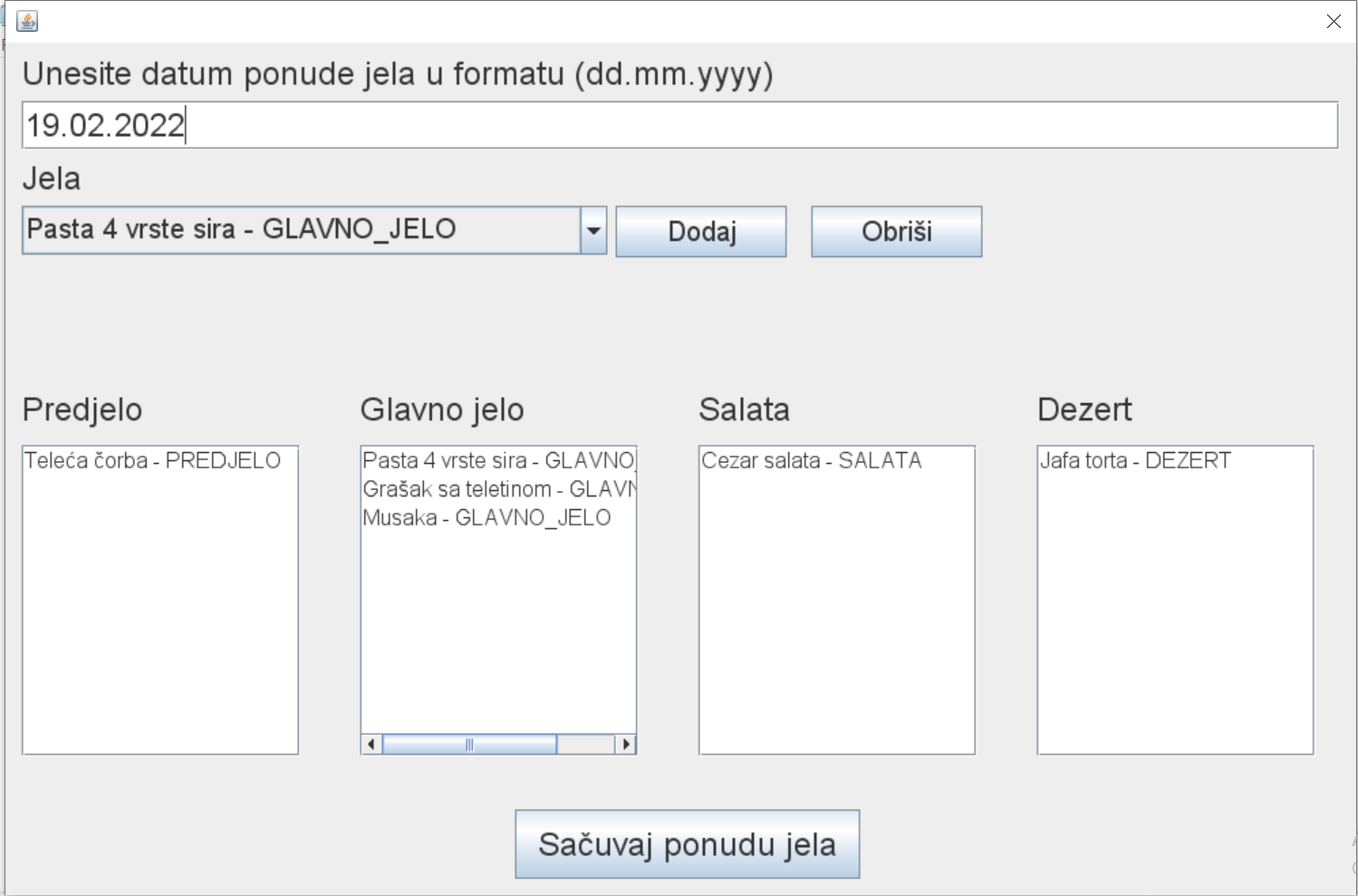
1. Administrator **poziva** sistem da učita ponudu za taj dan. (**APSO**)

Opis akcije: Administrator pritiskom na dugme Izmeni ponudu jela poziva sistem da poziva sistem da učita izabranu ponudu jela.

3. Sistem **pronalazi** ponudu za izabrani dan. (**SO**)

4. Sistem **prikazuje** ponudu jela koje je administrator izabrao za taj dan. (**IA**)

5. Administrator **menja** jela. (**APUSO**)



6. Administrator **poziva** sistem da ažurira ponudu jela za taj dan. (**APSO**)

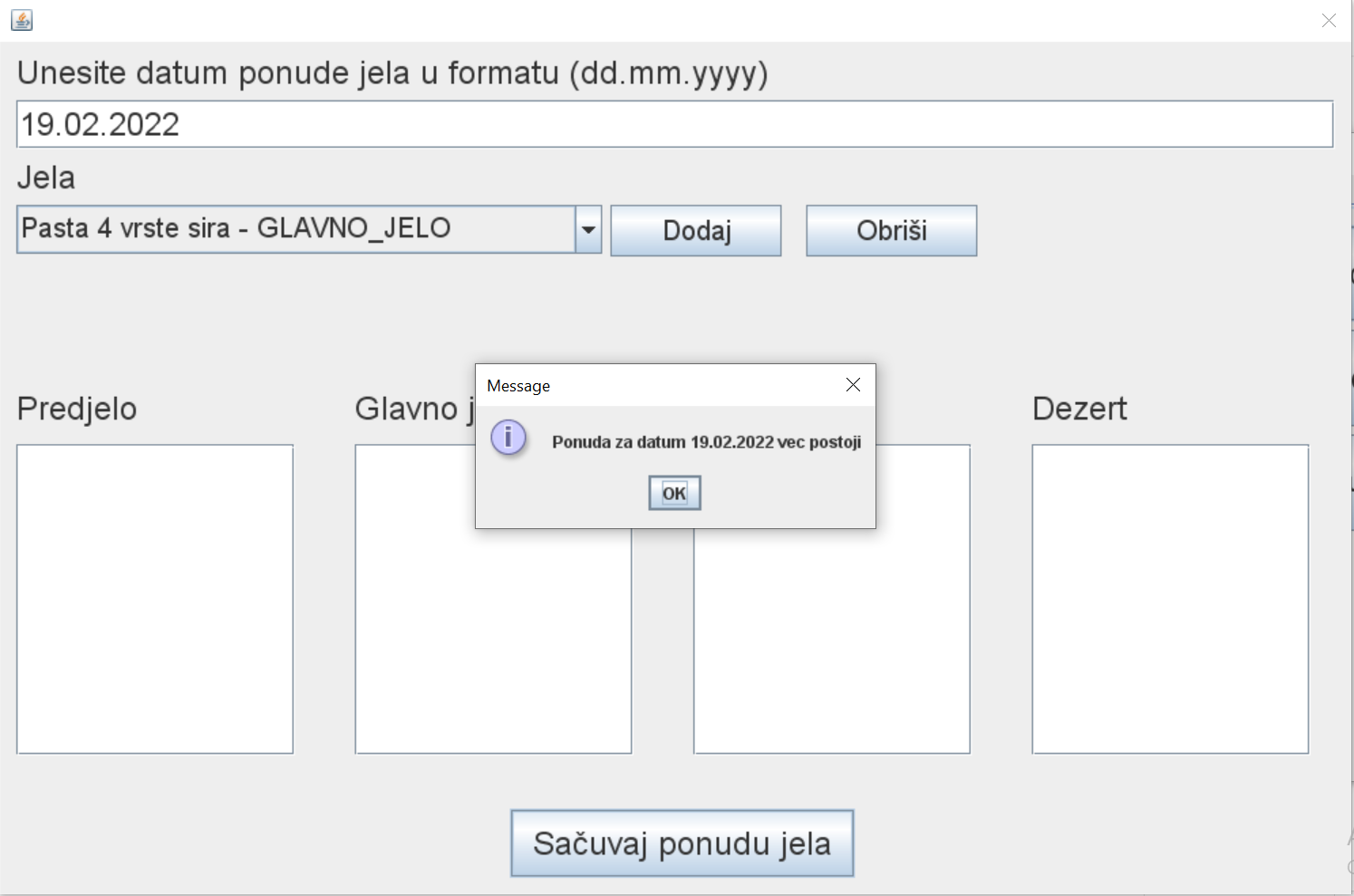
Opis akcije: Administrator pritiskom na dugme Sačuvaj ponudu jela poziva sistemsku operaciju IzmeniPonuduJela(PonudaJela): Signal

7. Sistem **ažurira** ponudu jela. (**SO**)

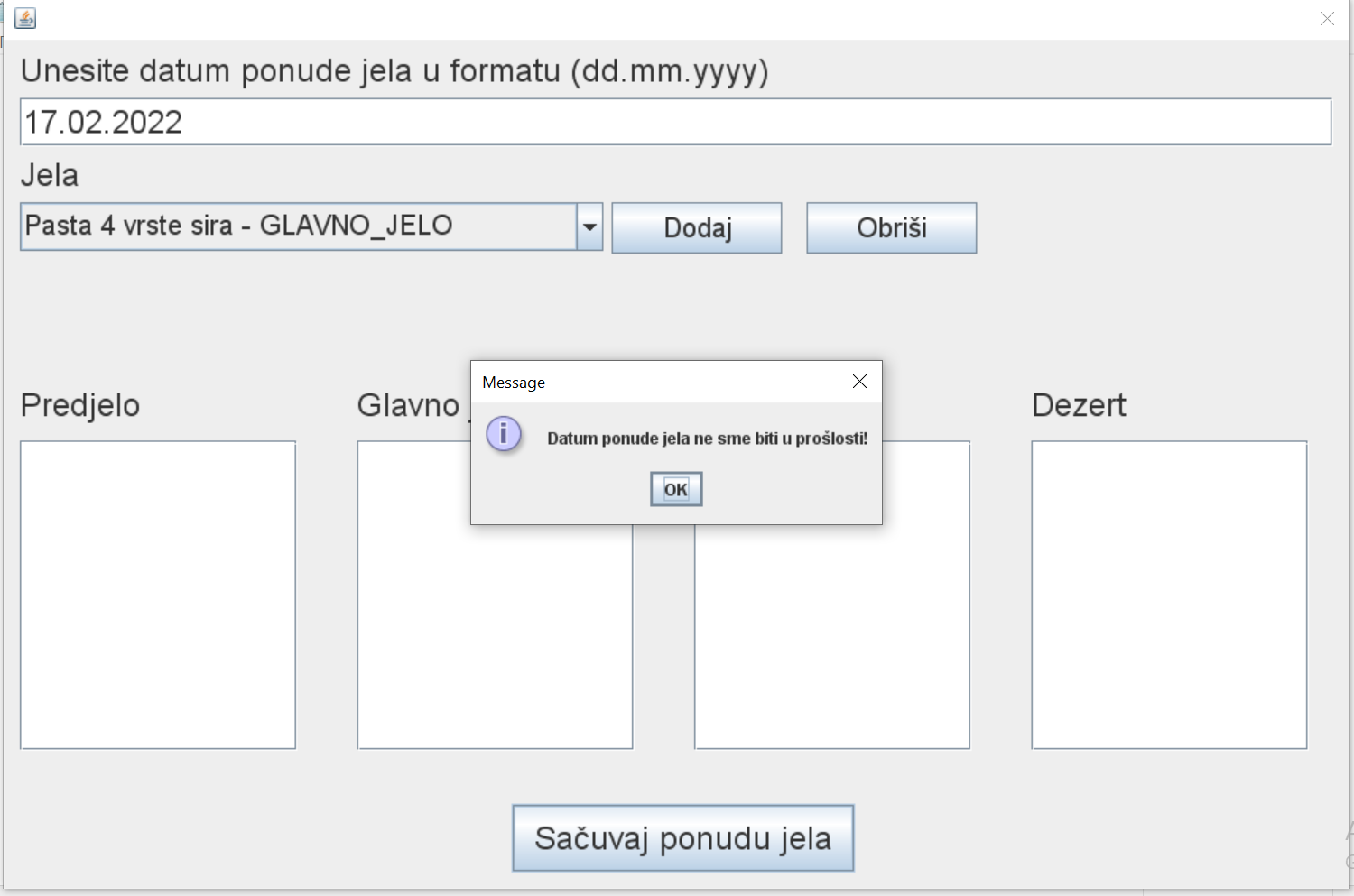
8. Sistem **prikazuje** poruku “Ponuda je uspešno sačuvana”. (**IA**)

**Alternativni scenariji:**

* 1. Ukoliko je administrator izabrao dan za koje već postoji ponuda jela sistem prikazuje poruku: “Ponuda za dan već postoji”. (**IA**)



6.2 . Ukoliko je korisnik izabrao za datum ponude neki dan iz prošlosti prikazuje se poruka: “Datum ponude ne sme biti u prošlosti”.(**IA**)



### SK6 – Pregled porudžbina

**Naziv SK:** Pregled porudžbina

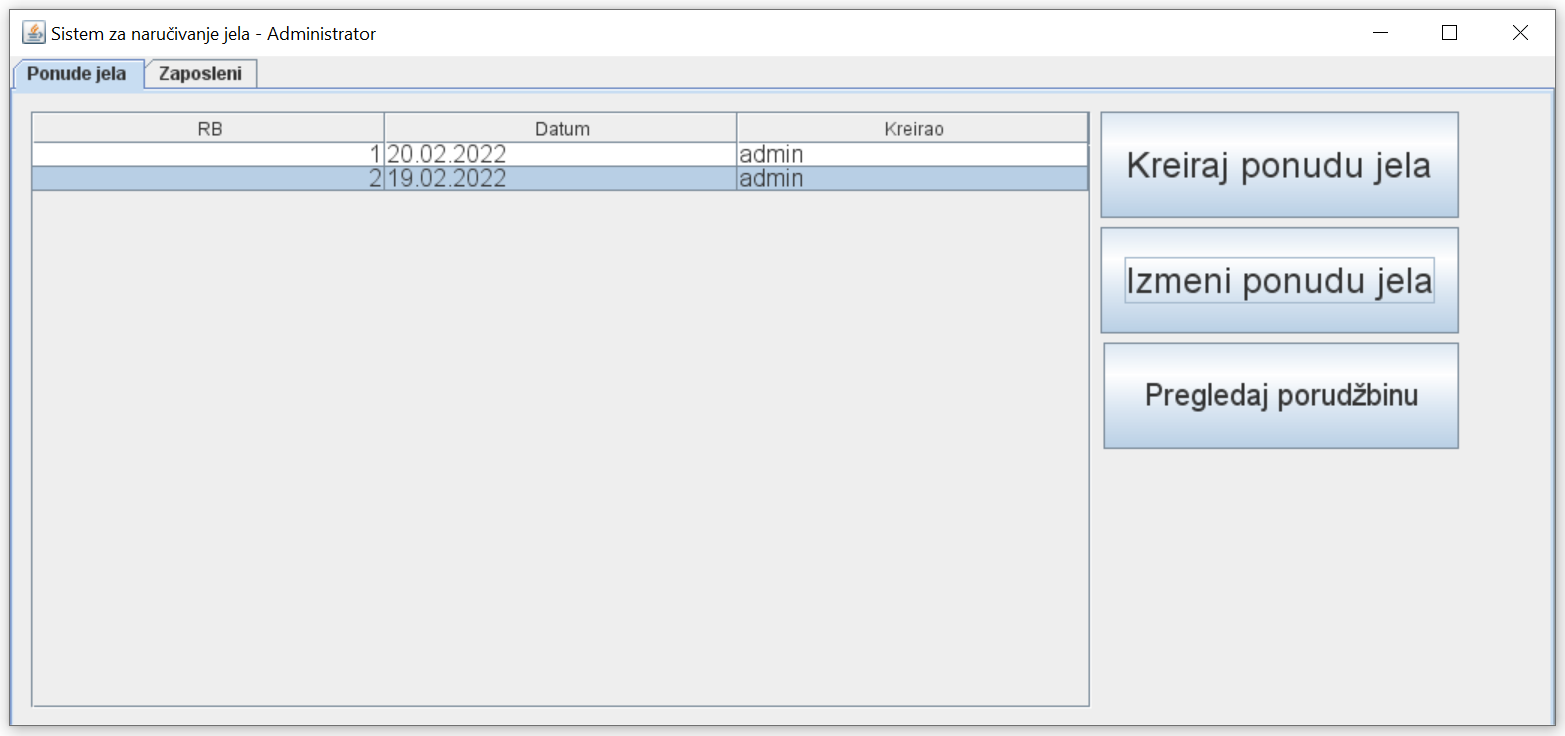
**Aktori SK:** Zaposleni, Administrator

**Učesnici SK:** Administrator, Zaposleni, sistem

**Preduslov:** Sistem je pokrenut, aktor je ulogovan na svoj nalog, prikazana je lista porudžbina sortirana po danima (opadajuće).

**Osnovni scenario SK:**

1. Aktor **bira** ponudu za koju želi da pregleda poručena jela zaposlenih.(**APUSO**)

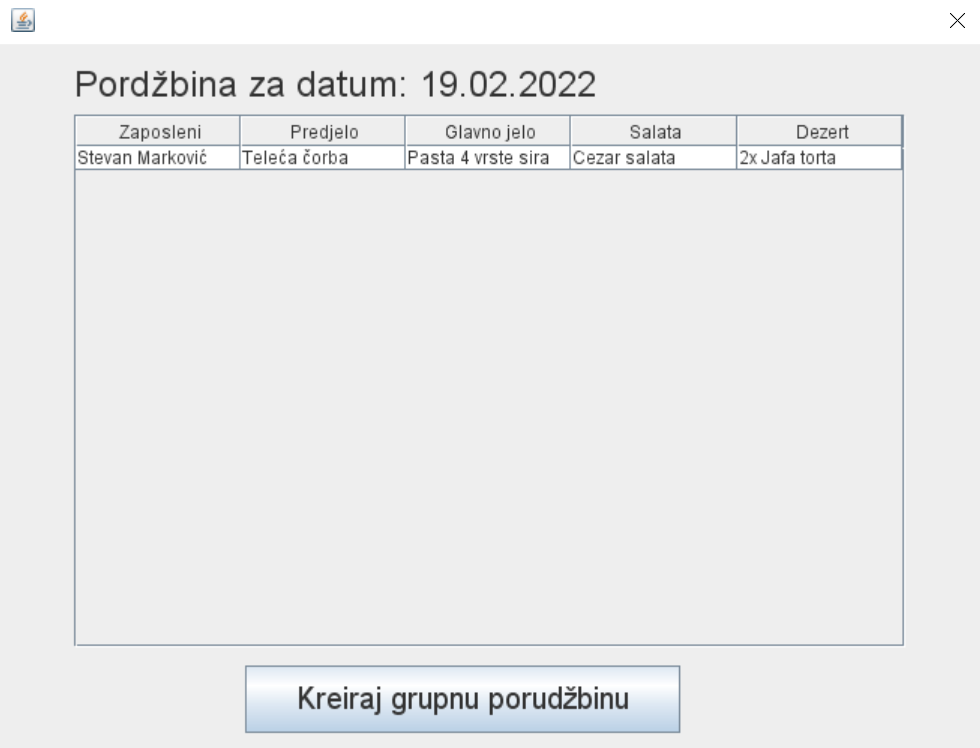


1. Aktor **poziva** sistem da učita poručena jela za tu ponudu. (**APSO**)

Opis akcije: Pritiskom na dugme Pregledaj produžbinu korisnik poziva sistemsku operaciju VratiSvePorudzbineZaPonuduJela(PonudaJela): List<Porudzbina>

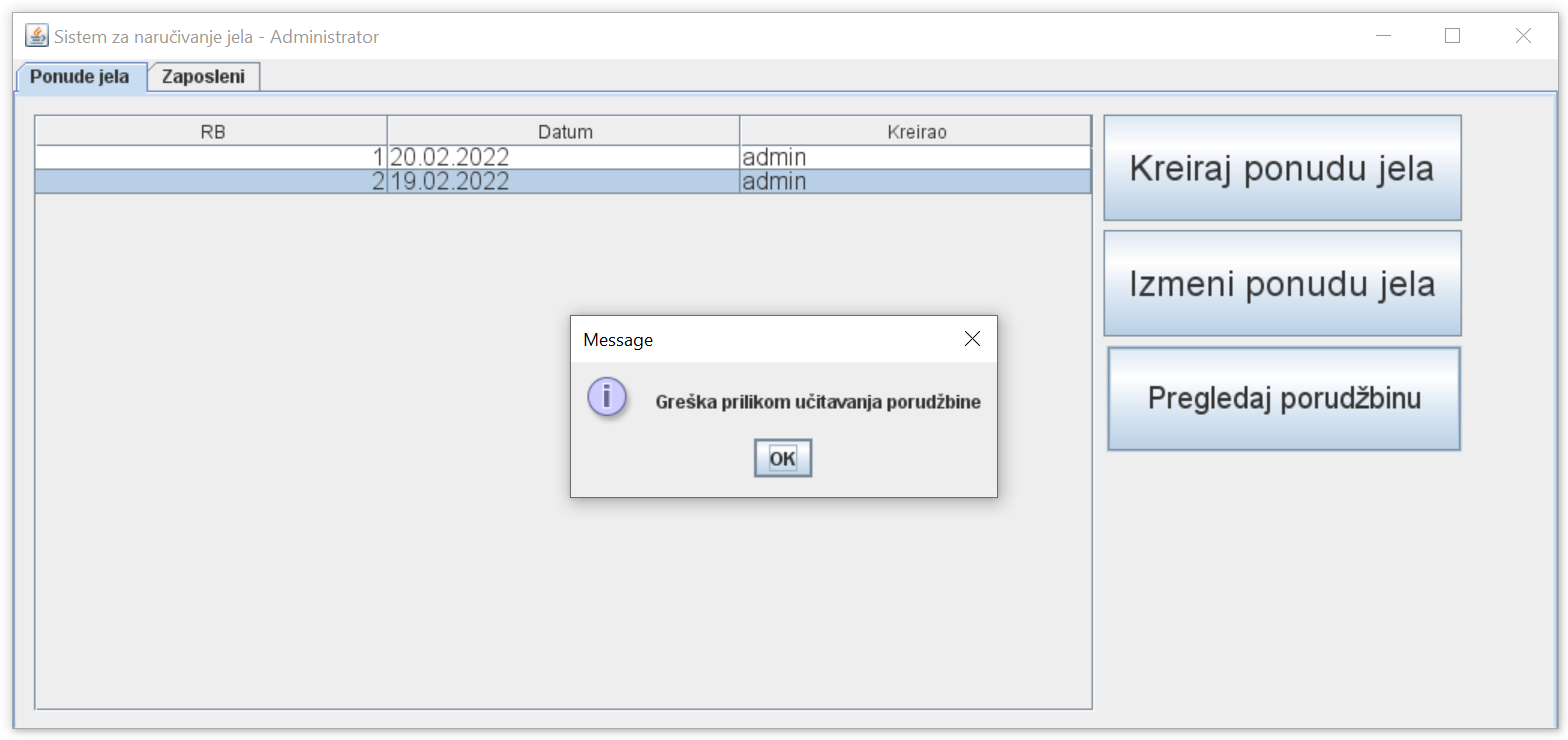
3. Sistem **učitava** poručena jela za izabranu ponudu. (**SO**)

4. Sistem **prikazuje** izabrana jela za taj dan za svakog zaposlenog. (**IA**)



**Alternativni scenariji:**

4.1. Ukoliko sistem ne može da učita podatke o porudžbini prikazuje poruku: “Greška prilikom učitavanja porudžbine”. (**IA**)



### SK7 – Kreiranje grupne porudžbine

**Naziv SK:** Kreiranje grupne porudžbine

**Aktori SK:** Administrator

**Učesnici SK:** Administrator, sistem

**Preduslov:** Sistem je pokrenut, aktor je ulogovan na svoj nalog, prikazana je lista porudžbina sortirana po danima (opadajuće).

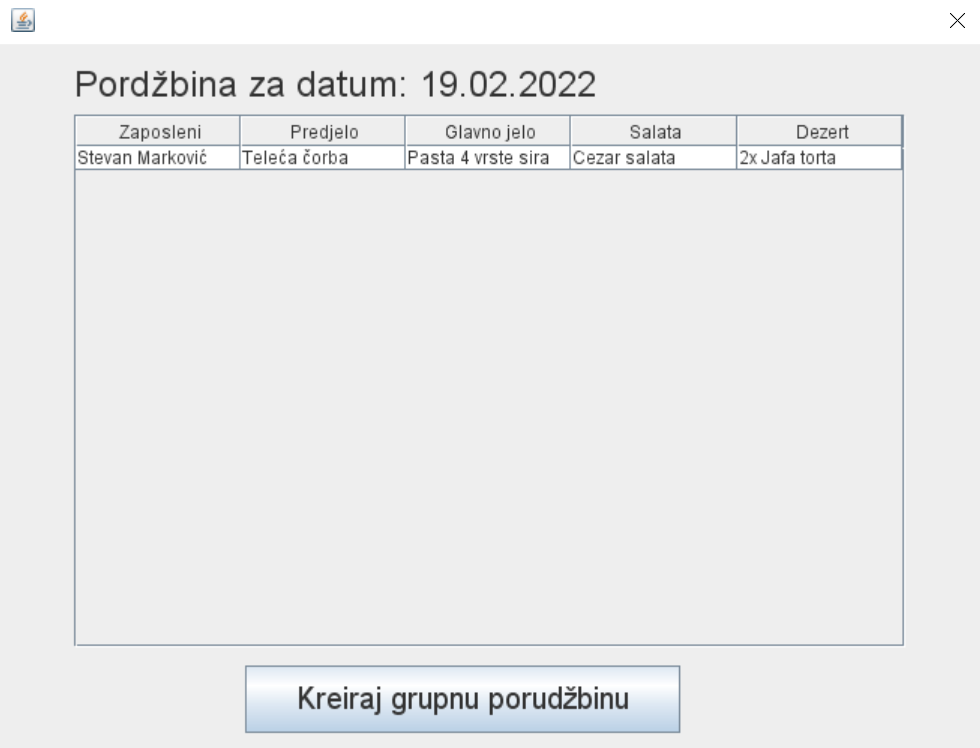
**Osnovni scenario SK:**

1. Aktor **poziva** sistem da učita porudžbine svih zaposlenih za izabranu ponudu (**APSO**)

Opis akcije: Pritiskom na dugme Pregled porudžbine aktor poziva sistemsku operaciju: VratiSvePorudzbineZaPonuduJela(PonudaJela): List<Porudzbina>

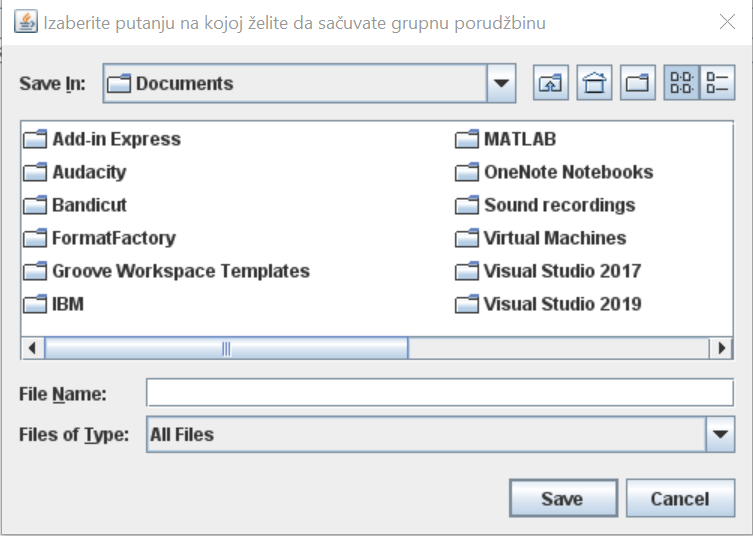
2. Sistem **učitava** porudžbine svih zaposlenih za tu ponudu. (**SO**)

3. Sistem **prikazuje** izabrana jela za svakog zaposlenog. (**IA**)



4. Aktor poziva sistem da **kreira** grupnu porudžbinu za tu ponudu. (**APSO**)

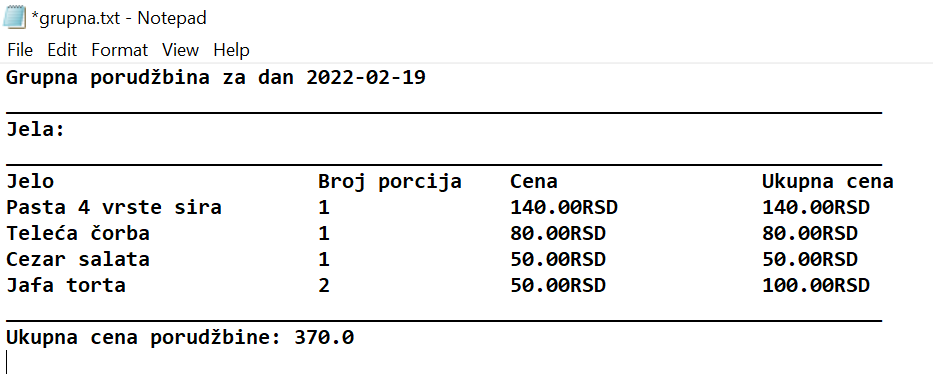
Opis akcije: Pritiskom na dugme Kreiraj grupnu porudžbinu aktor poziva sistemsku operaciju: KreirajGrupnuPorudzbinu(PonudaJela): String

1. Sistem **kreira** grupnu porudžbinu (SO)
2. Sistem **prikazuje** formu za čuvanje grupne pordužbine kao tekstualnog fajla. (IA) 

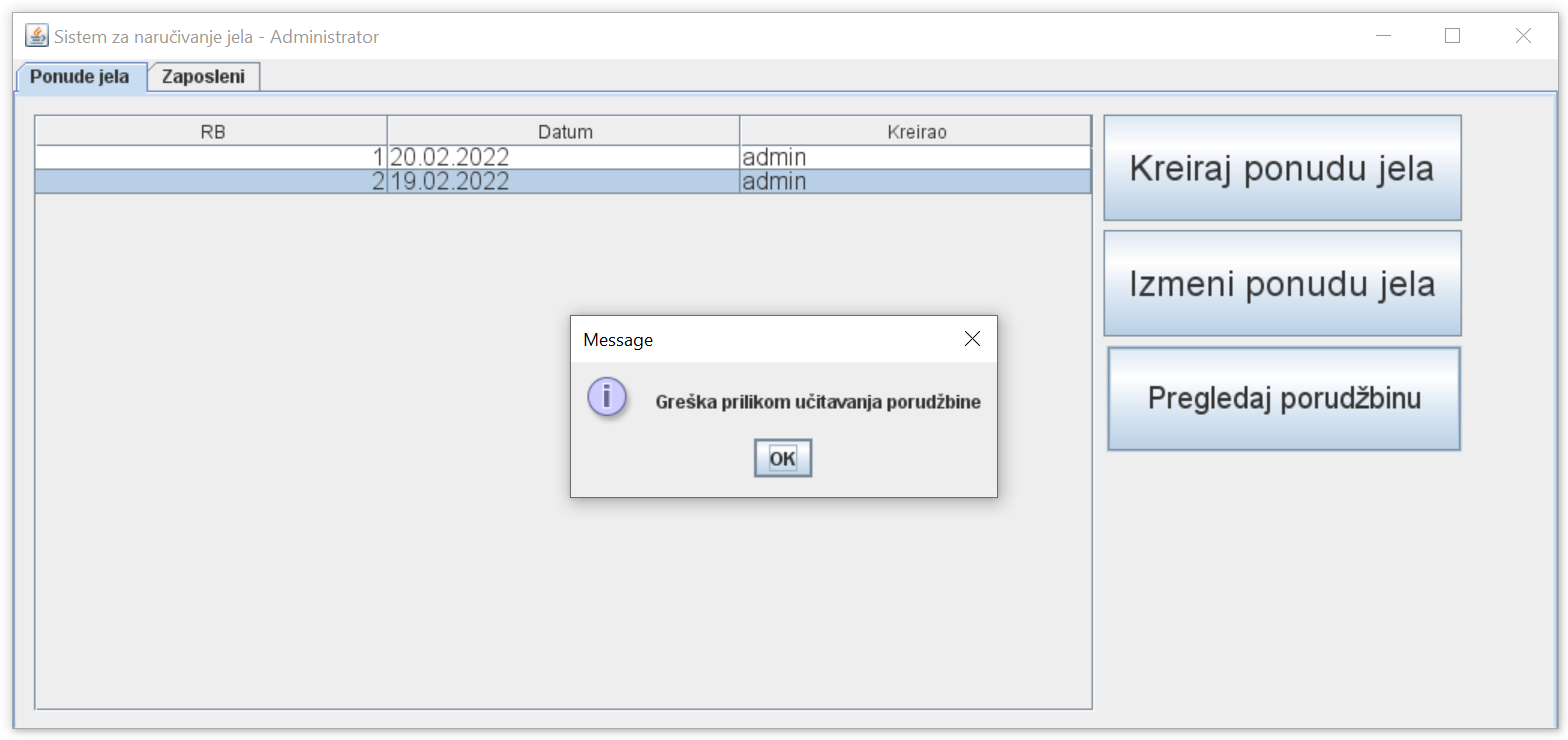
7. Aktor bira lokaciju na fajl sistemu gde želi da sačuva grupnu porudžbinu kao i naziva fajla.(**APUSO**)

7. Aktor **poziva** sistem za sačuva grupnu porudžbinu.(**APSO**)

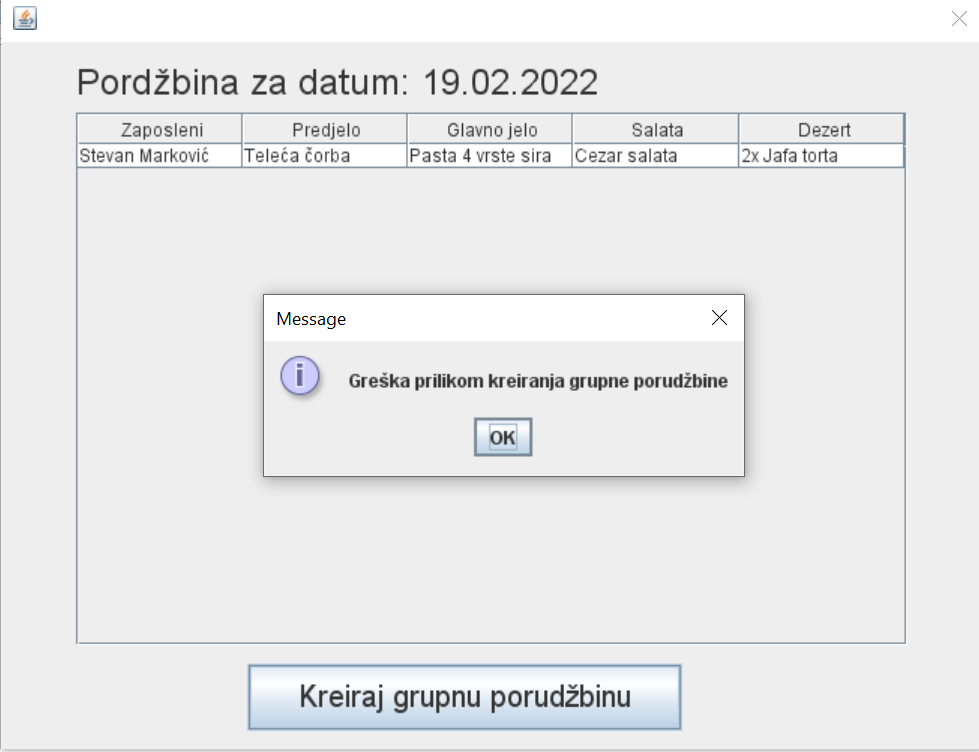
8. Sistem **čuva** grupnu porudžbinu kao tekstualni fajl. (**IA**)

****

**Alternativni scenariji:**

3.1. Ukoliko sistem ne može da učita podatke o porudžbini prikazuje poruku: “Greška prilikom učitavanja porudžbine”. (**IA**) 

5.1. Ukoliko sistem **ne može** da kreira grupnu porudžbinu, prikazuje poruku “Greška prilikom kreiranja grupne porudžbine”. (**IA**)



### SK8 – Poručivanje (složen)

**Naziv SK:** Poručivanje

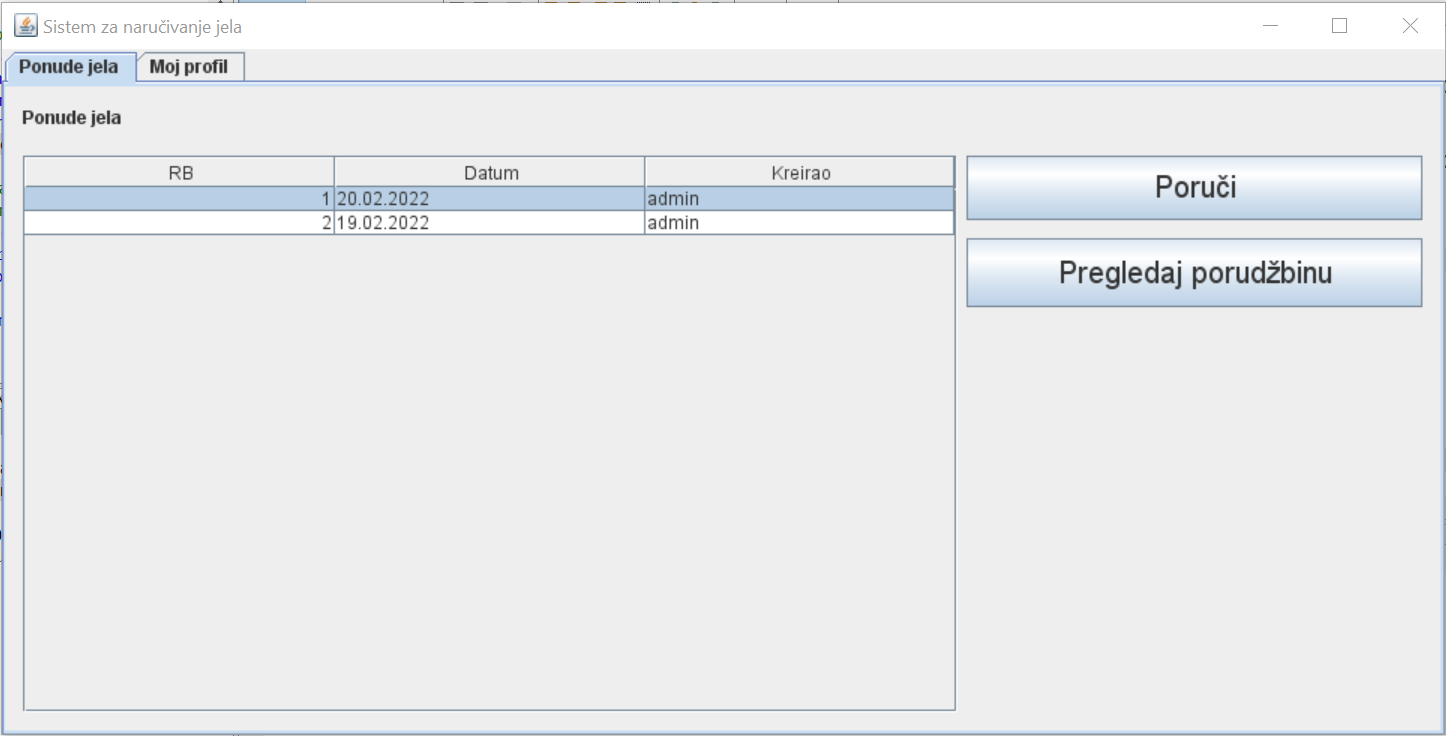
**Aktori SK:** Zaposleni

**Učesnici SK:** Zaposleni, sistem

**Preduslov:** Sistem je pokrenut, zaposleni je ulogovan na svoj nalog, izlistani su svi dani za koje je administrator kreirao ponudu jela.

**Osnovni scenario SK:**

1. Aktor **bira** dan za koji hoće da izvrši poručivanje. (**APUSO**)



1. Aktor **poziva** sistem da učita ponuđena jela za taj dan. (**APSO**)

Opis akcije: Pritiskom na dugme Poruči zaposleni poziva sistemsku operaciju VratiPonuduJela(IdPonude): PonudaJela

1. Sistem **učitava** ponudu jela sa ponuđenim jelima za izabrani dan. (**SO**)
2. Sistem **prikazuje** ponuđena jela za dan koji je aktor izabrao. (**IA**)
3. Aktor od ponuđenih jela **bira** po jedno jelo od svake od kategorija jela(predjelo, glavno jelo, salata i dezert) kao i broj porcija (**APUSO**)



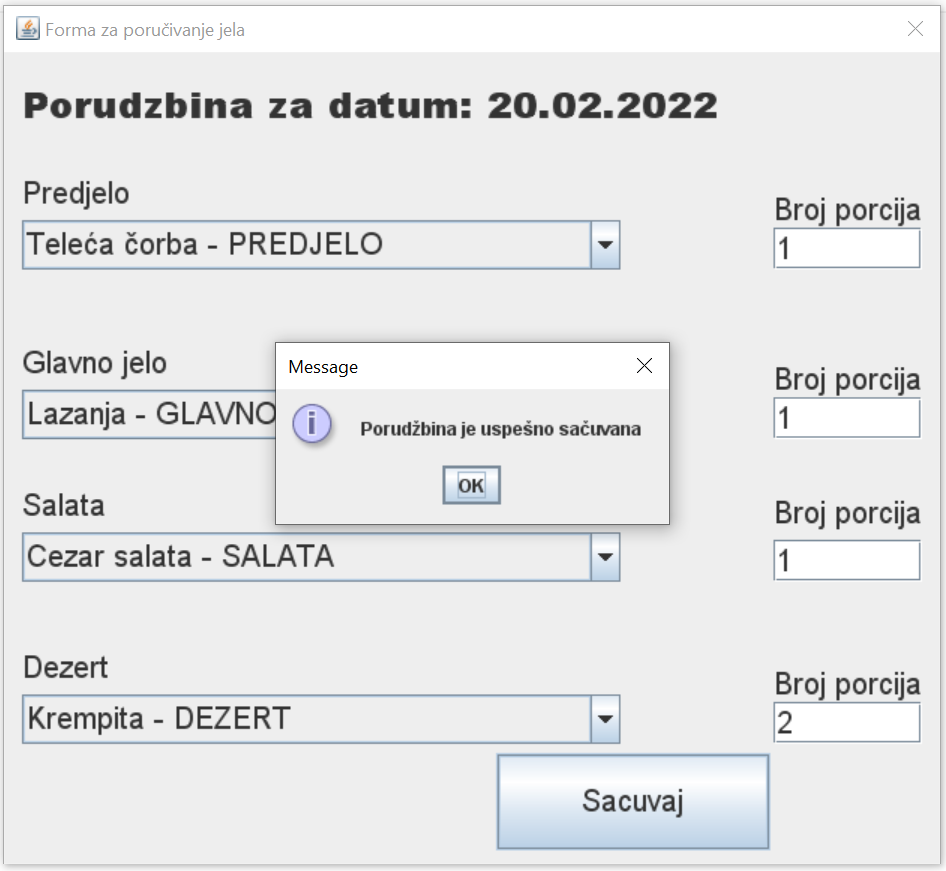
5. Aktor **proverava** da li je dobro izabrao jela. (**ANSO**)

6. Aktor **poziva** sistem da sačuva poručena jela.(**APSO**)

Opis akcije: pritiskom na dugme Sačuvaj aktor poziva sistemsku operaciju SacuvajPorudzbinu(Porudzbina): Signal

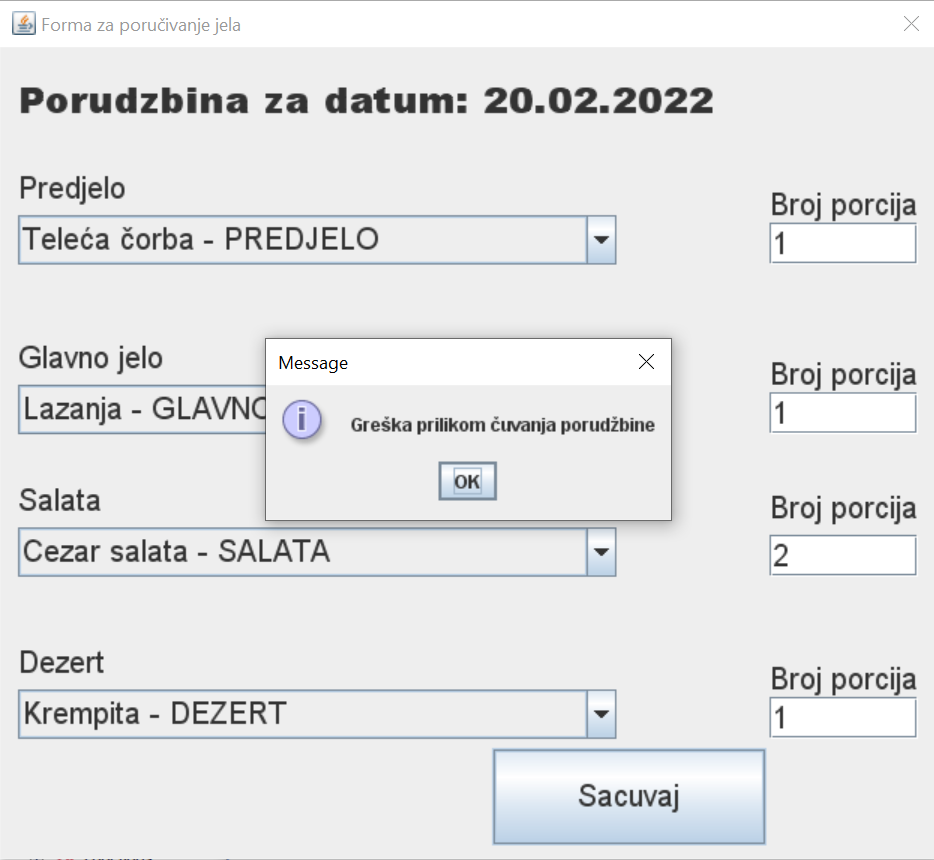
7. Sistem **pamti** poručena jela zaposlenog. (**SO**)

8. Sistem prikazuje **poruku** “Porudžbina je uspešno sačuvana”. (**IA**)



**Alternativni scenariji:**

8.1. Ukoliko sistem ne može da sačuva poručena jela, prikazuju poruku “Greška prilikom čuvanja porudžbine.” (**IA**)



### SK9 – Izmena porudžbine

**Naziv SK:** Izmena porudžbine

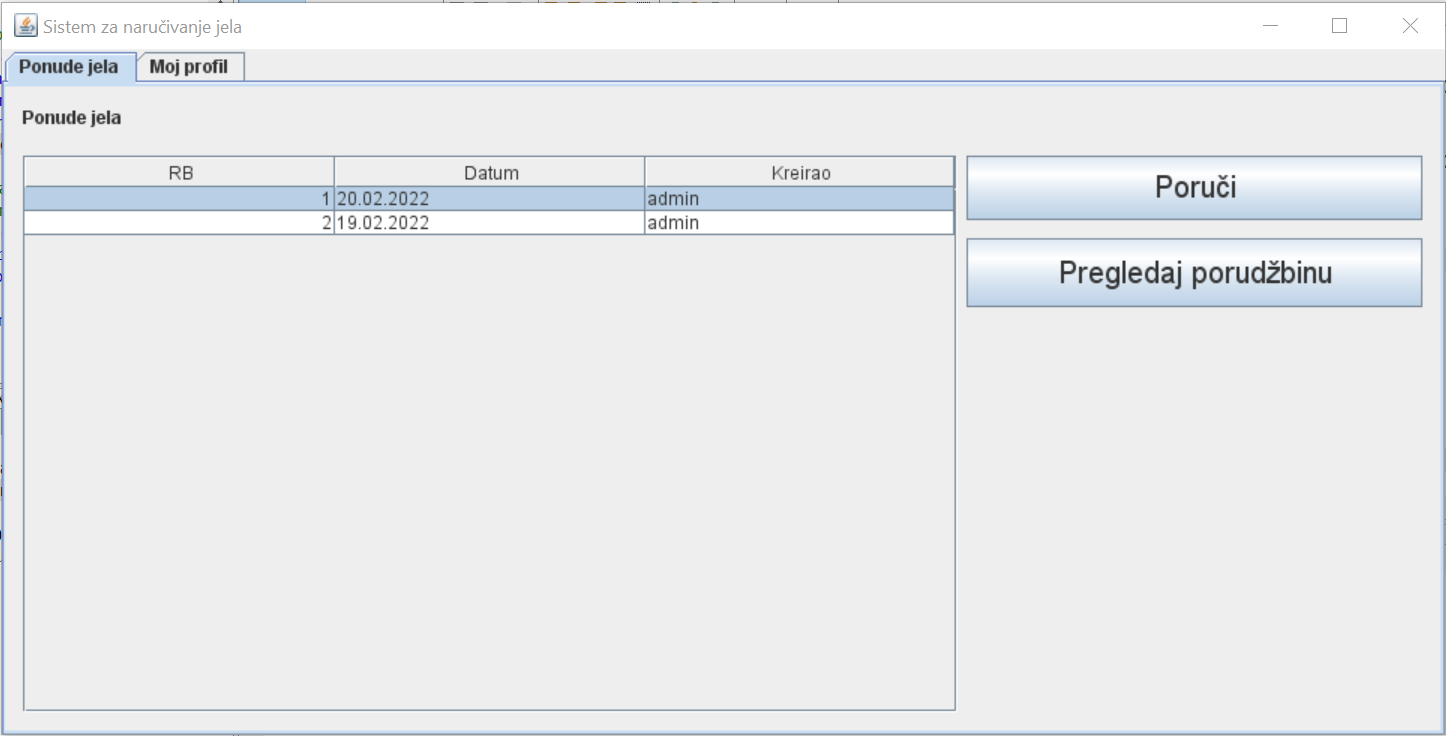
**Aktori SK:** Zaposleni

**Učesnici SK:** Zaposleni, sistem

**Preduslov:** Sistem je pokrenut, zaposleni je ulogovan na svoj nalog, izlistani su svi dani za koje je administrator kreirao ponudu jela.

**Osnovni scenario SK:**

1. Aktor **bira** dan za koji hoće da izmeni porudžbinu. (**APUSO**)



1. Aktor **poziva** sistem da učita ponuđena jela za taj dan.(**APSO**)

Opis akcije: pritiskom na dugme Poruči aktor poziva sistemsku operaciju VratiPonuduJela(Ponuda): Signal i VratiPorudzbinuZaposlenog(Zaposleni): List<Porudzbina>

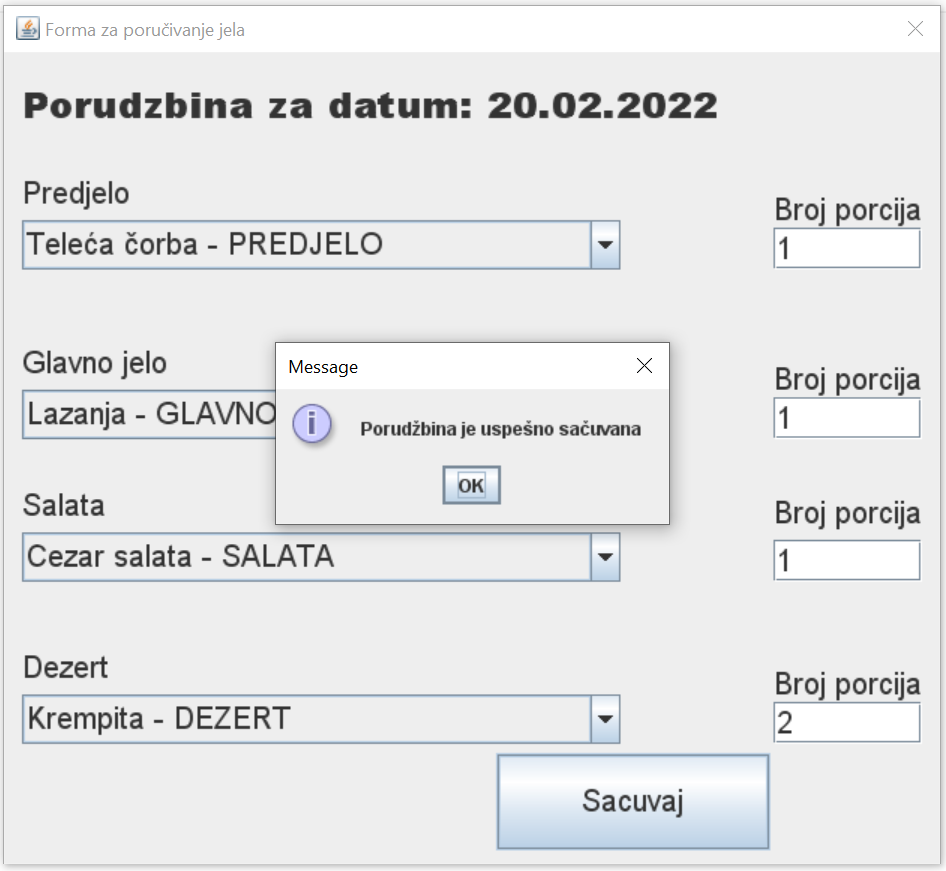
1. Sistem **vraća** ponuđena jela za taj dan i jela koja je zaposleni poručio za taj dan.(**IA**)



1. Aktor **menja** svoju porudžbinu. (**APUSO**)
2. Aktor **poziva** sistem da sačuva izmenjenu porudžbinu. (**APSO**)

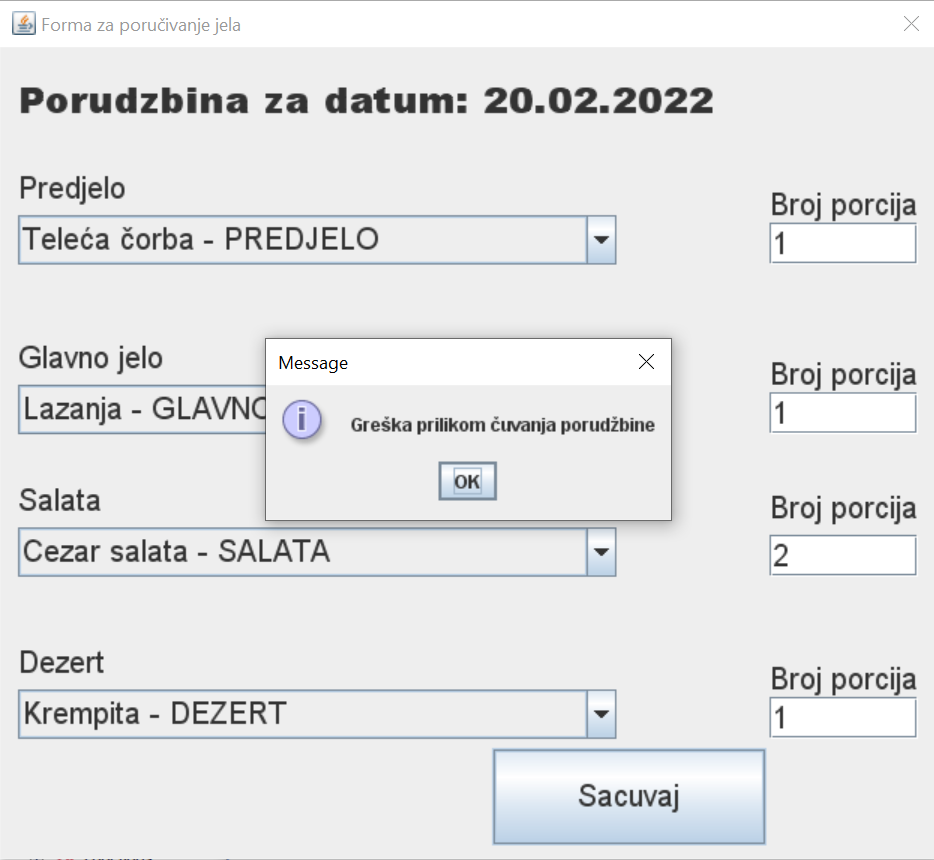
Opis akcije Pritiskom na dugme Sacuvaj aktor poziva sistemsku operaciju SacuvajPorudzbinu(Porudzbina): Signal

1. Sistem **čuva** porudžbinu zaposlenog. (**SO**)
2. Sistem **prikazuje** poruku “Porudžbina je uspešno sačuvana”. (**IA**)



**Alternativni scenariji:**

8.1 Ukoliko sistem ne može da sačuva porudžbinu prikazuje “Greška prilikom čuvanja porudžbine.” (IA)



### 3.1.2 Projektovanje kontrolera korisničkog interfejsa

Kontroler korisničkog interfejsa je odgovoran za:

1. Prihvatanje grafičkih objekata od ekranske forme.
2. Konvertovanje podataka koji se nalaze u grafičkim objektima u domenske objekte koji će biti proslijeđeni preko mreže do aplikacionog servera.
3. Konvertovanje domenskih objekata u grafičke objekte i prosleđuje ih do ekranske forme.

Kontroler je realziovan preko Singleton dizajn paterna zbog neophodnosti da sve ekranske forme koriste jednu instancu kontrolera.

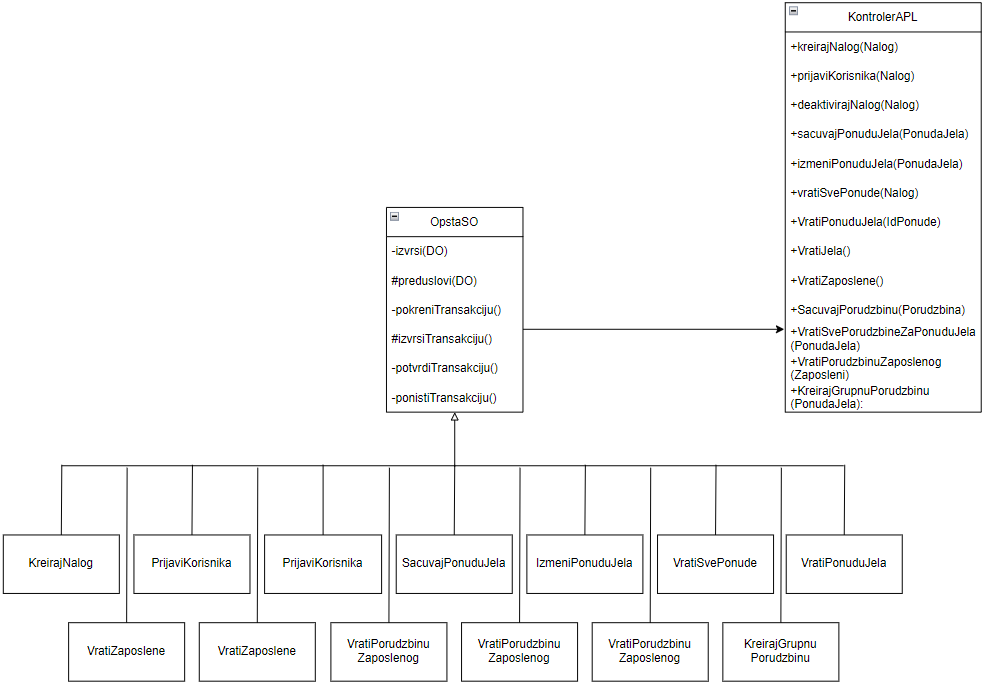
## Projektovanje aplikacione logike

Aplikaciona logika se sastoji od tri dela:

* Kontroler aplikacione logike
* Poslovna logika
* Broker baze podataka

### Kontroler aplikacione logike

Kontroler aplikacione logike prihvata od niti koja komunicira sa klijentom zahteve za izvršenje sistemskih operacija i poziva odgovarajuće klase poslovne logike koje su odgovorne za izvršenje tih operacija. Rezultate operacija zatim vraća do klijentske niti koja klijentskoj aplikaciji prosleđuje odgovor servera na zahtevanu operaciju.



### 3.3.2 Poslovna logika

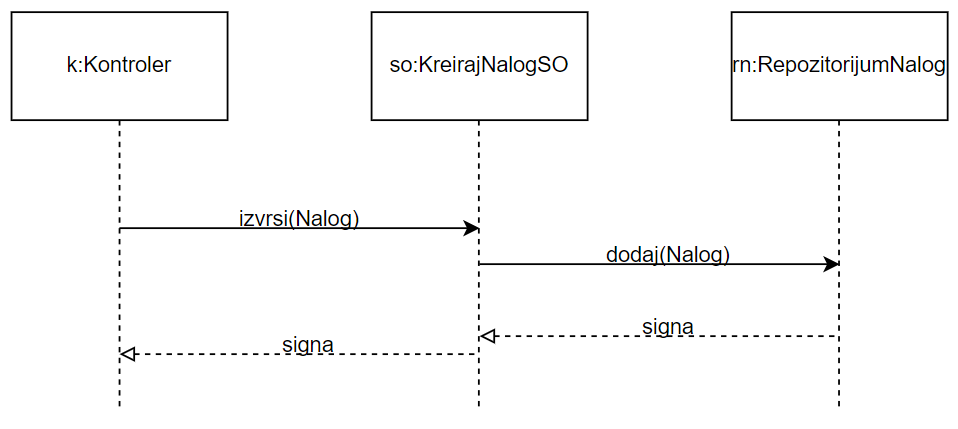
Poslovna logika realizovana je preko klasa od kojih je svaka zadužena za izvršavanje određene sistemske operacije.

**Ugovor UG1**: **KreirajNalog(**Nalog**): Signal**

**Veza sa SK**: SK1

**Preduslovi**: Vrednosna i strukturna ograničenja nad objektom **Nalog** moraju biti zadovoljena

**Postuslovi**: Nalog je kreiran

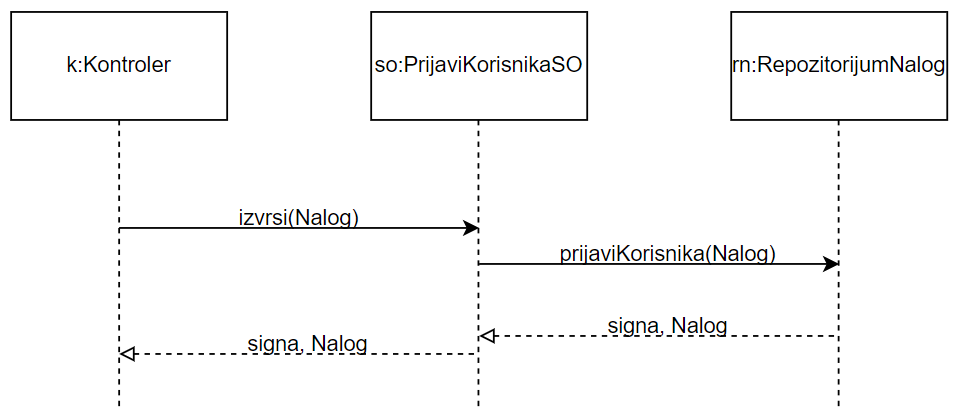


**Ugovor UG2**: **PrijaviKorisnika(**Nalog**): Nalog**

**Veza sa SK**: SK3

**Preduslovi**: Vrednosna i strukturna ograničenja nad objektom **Nalog** moraju biti zadovoljena

**Postuslovi**: /

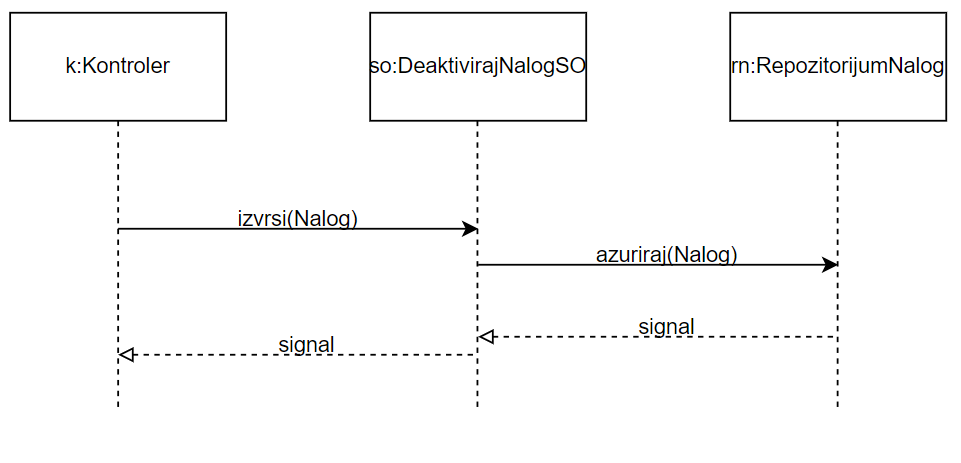


**Ugovor UG3**: **DeaktivirajNalog(**Nalog**): Signal**

**Veza sa SK**: SK2

**Preduslovi**: Strukturna i vrednosna ograničenja nad objektom **Nalog** moraju biti zadovoljena

**Postuslovi**: Nalog je deaktiviran

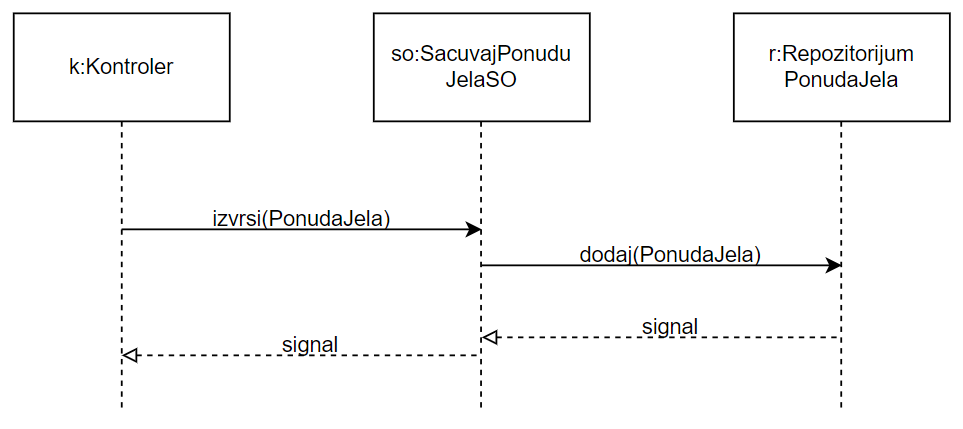


**Ugovor UG4**: **SacuvajPonuduJela(PonudaJela): Signal**

**Veza sa SK**: SK4, SK5

**Preduslovi**: Strukturna i vrednosna ograničenja nad objektom **PonudaJela** moraju biti zadovoljena

**Postuslovi**: Ponuda je kreirana, Ponuđena jela su sačuvana

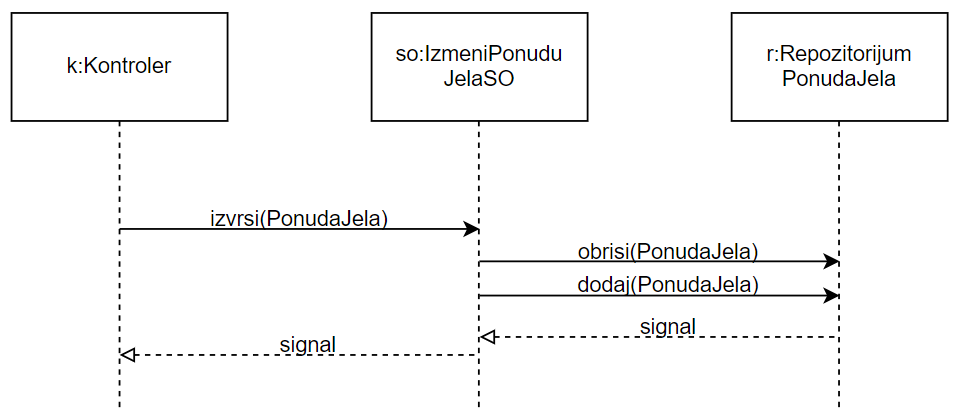


**Ugovor UG5**: **IzmeniPonuduJela(PonudaJela): Signal**

**Veza sa SK**: SK5

**Preduslovi**: Strukturna i vrednosna ograničenja nad objektom **PonudaJela** moraju biti zadovoljena

**Postuslovi**: Ponuda je izmenjena

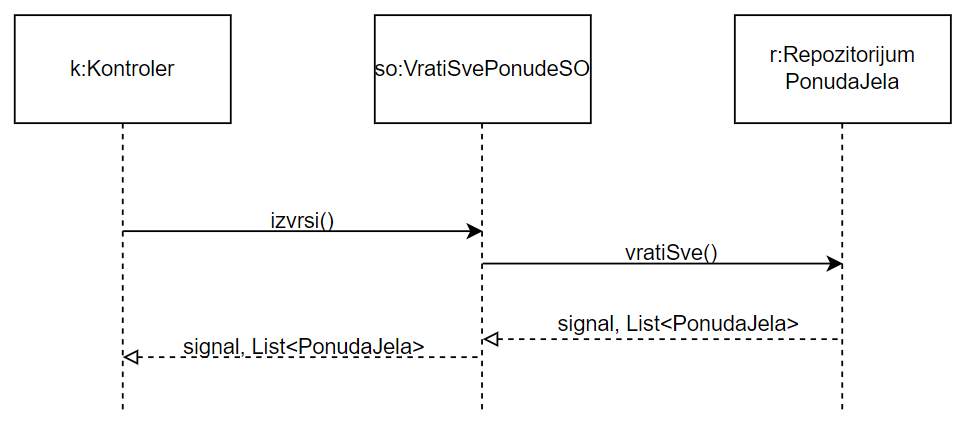


**Ugovor UG6**: **VratiSvePonude (): List<PonudaJela>**

**Veza sa SK**: SK5, SK6, SK7, SK8, SK9

**Preduslovi**: /

**Postuslovi**: /

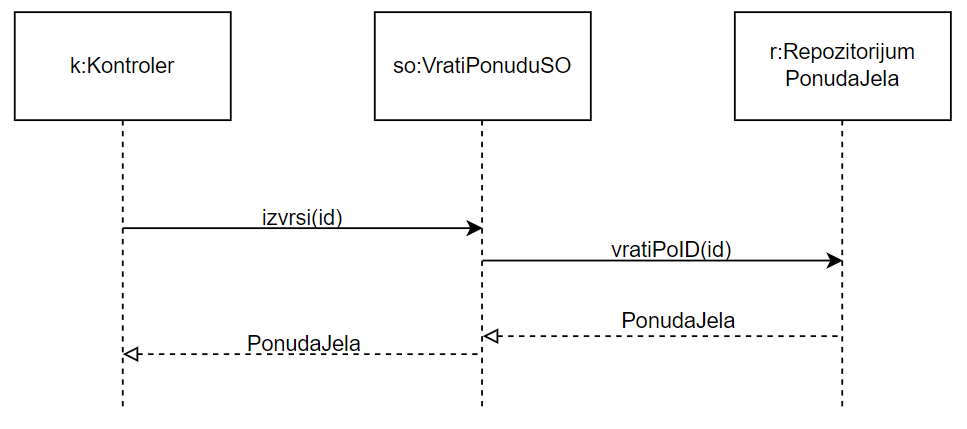


**Ugovor UG7:** **VratiPonuduJela(IdPonude): PonudaJela**

**Veza sa SK**: SK5, SK8, SK9

**Preduslovi**: /

**Postuslovi**: /

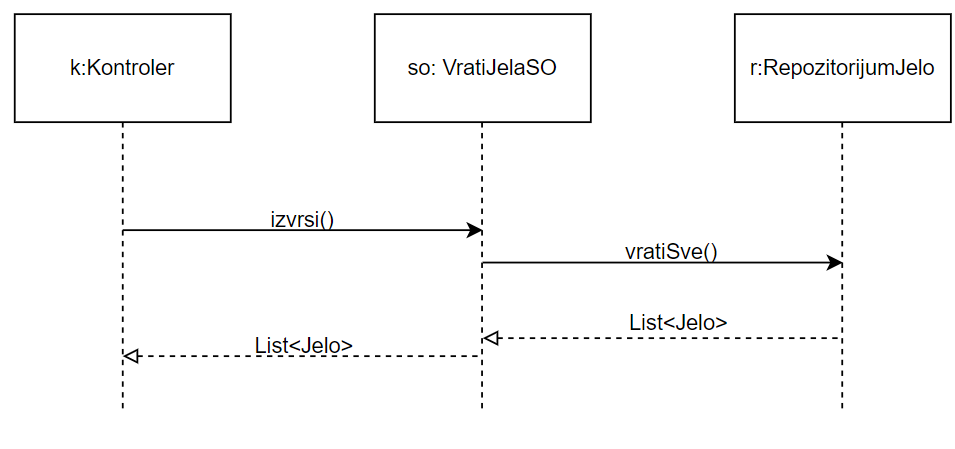


**Ugovor UG8**: **VratiJela(): List<Jelo>**

**Veza sa SK**: SK4, SK5

**Preduslovi**: /

**Postuslovi**: /

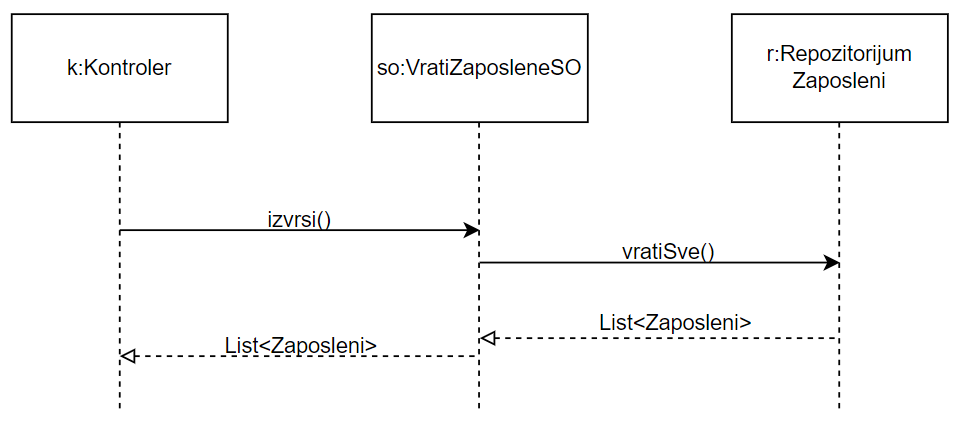


**Ugovor UG9**: **VratiZaposlene(): List<Zaposleni>**

**Veza sa SK**: SK1, SK2

**Preduslovi**: /

**Postuslovi**: /

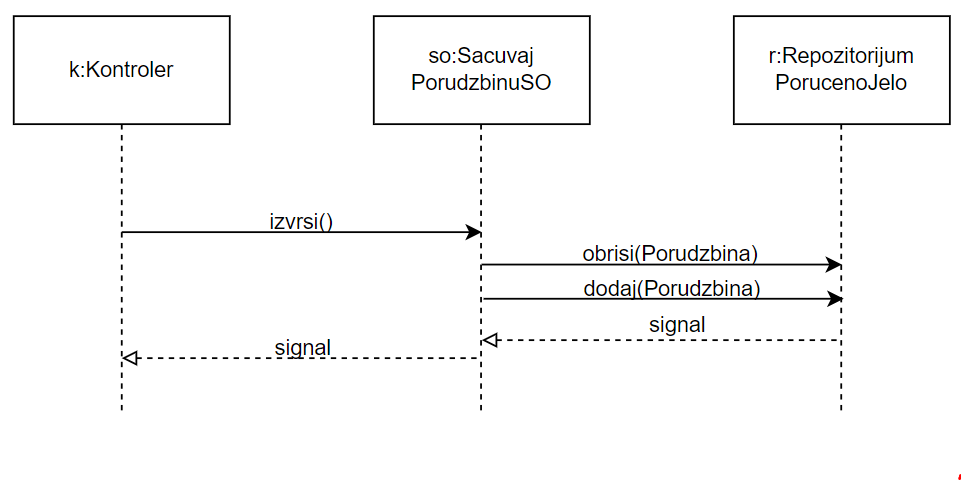


**Ugovor UG10**: **SacuvajPorudzbinu**(Porudzbina): Signal

**Veza sa SK**: SK8, SK9

**Preduslovi**: Strukturna i vrednosna ograničenja nad objektima **PoručenoJelo** moraju biti zadovoljena

**Postuslovi**: Poručena jela su kreirana

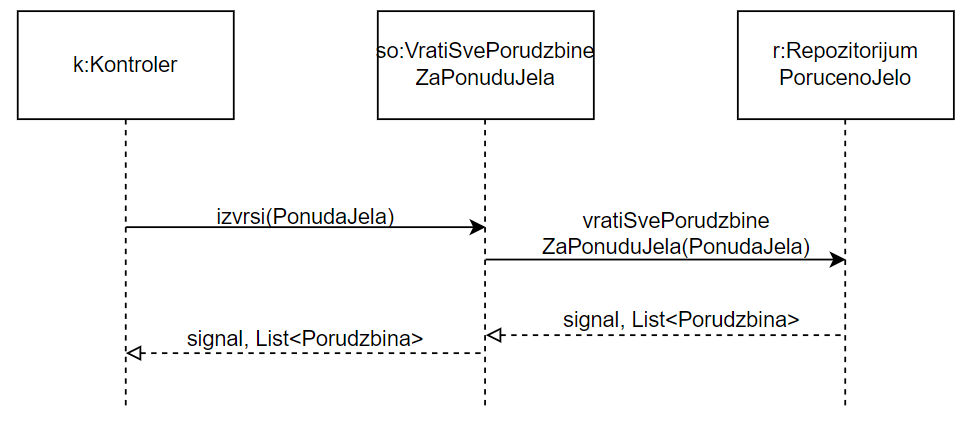


**Ugovor UG11**: **VratiSvePorudzbineZaPonuduJela**(PonudaJela): List<Porudzbina>

**Veza sa SK**: SK6, SK7

**Preduslovi**: /

**Postuslovi**: /

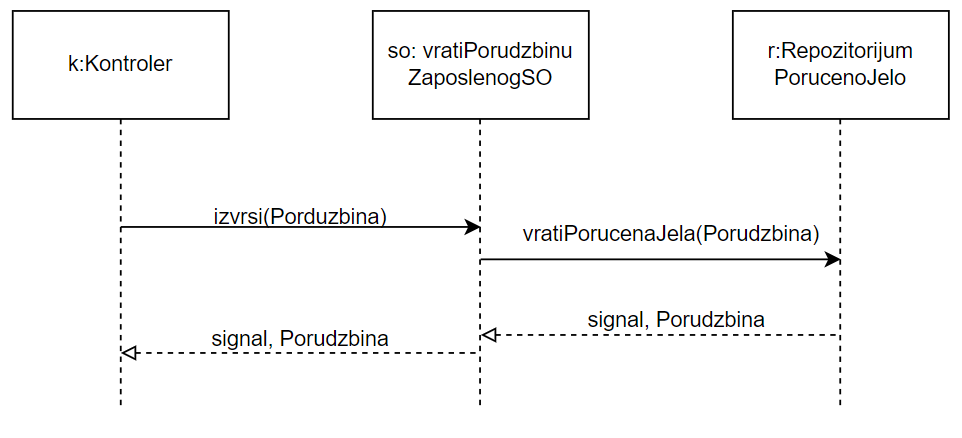


**Ugovor UG12**: **VratiPorudzbinuZaposlenog**(Zaposleni): List<Porudzbina>

**Veza sa SK**: SK9

**Preduslovi**: /

**Postuslovi**: /

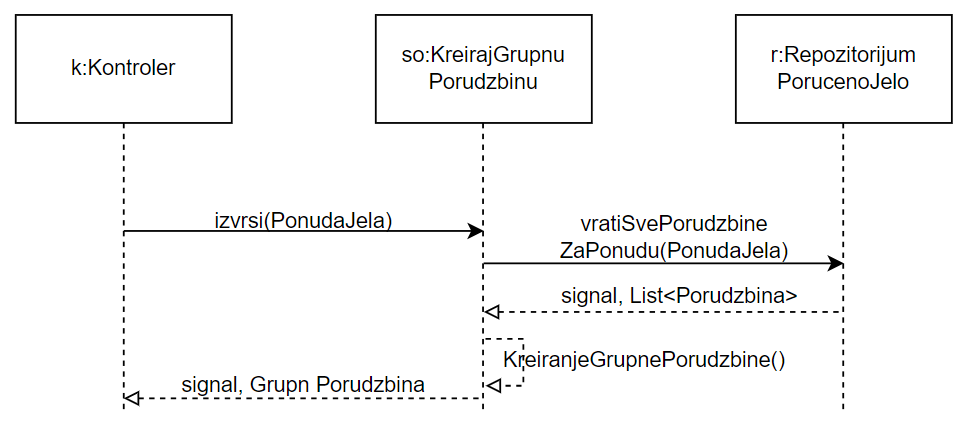


**Ugovor UG13**: **KreirajGrupnuPorudzbinu**(PonudaJela): String

**Veza sa SK**: SK7

**Preduslovi**: /

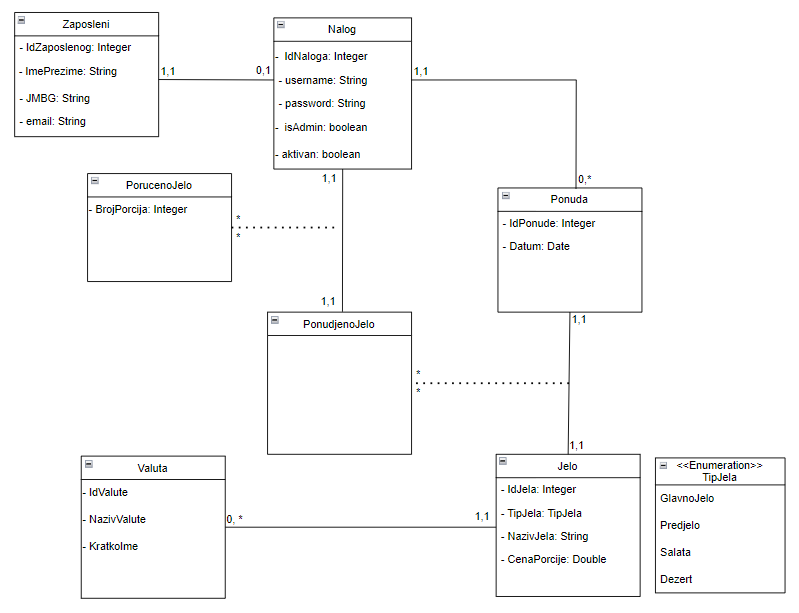
**Postuslovi**: /

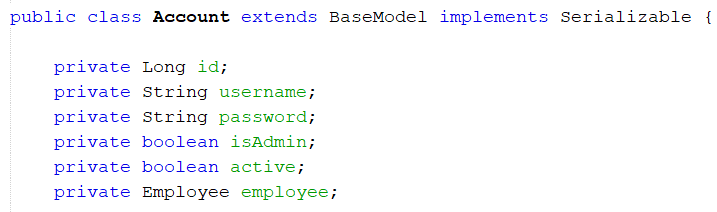


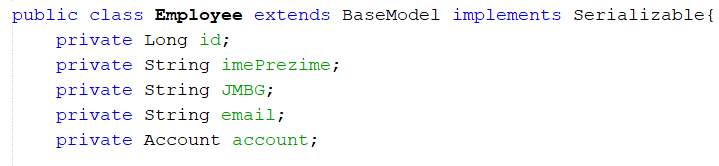
**Projektovanje strukture softverskog sistema**

Na osnovu domenskih klasa iz konceptualnog modela formiraju se softverske klase u kodu.

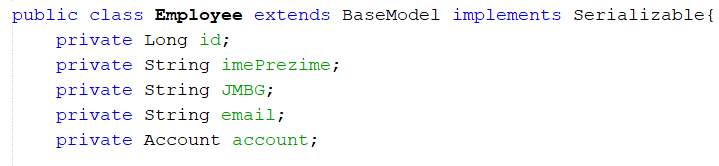
Konceptualne klase:



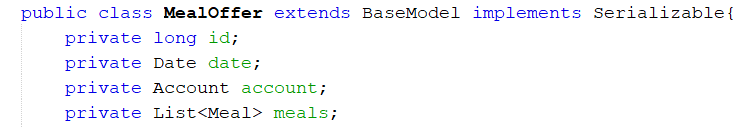
* Klasa Nalog implementirana je klasom Account
* Klasa Zaposleni implementirana je klasom Employee



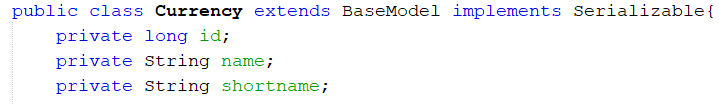
* Softverska klasa Meal reprezentuje klase Jelo, PonuđenoJelo i PoručenoJelo, koje zapravo predstavljaju koncept Jela u različitim stanjima



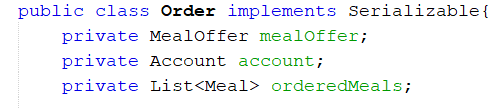
* PonudaJela implementirana je softverskom klasom MealOffer



* Valuta je implementirana klasom Currency



* Klasa Order(Porudzbina) nije klasa konceptualnog modela, i nema direktnu reprezentaciju u bazi ali se koristi radi lakšeg rada sa drugim modelima jer ona povezuje Ponudu jela, nalog koji vrši porudžbinu i jela koja je poručio.



### 3.3.3 Broker baze podataka

Uloga brokera baze podataka realizovana je preko Repository klasa. Postoji jedna generička klasa Repository koja implementira osnovne metode za rad nad tabelom relacione baze podataka:

* List<BaseModel> getAll()
* void add(BaseModel el)
* void update(BaseModel el)
* void delete(BaseModel el)
* BaseModel getById(BaseModel el)

Takođe domenski modeli imaju posebne Repository klase koje implementiraju neke domenski specifičnije metode. Postoje:

* AccountRepository
* EmployeeRepository
* MealOfferRepository
* MealRepository
* OrderedMealRepository

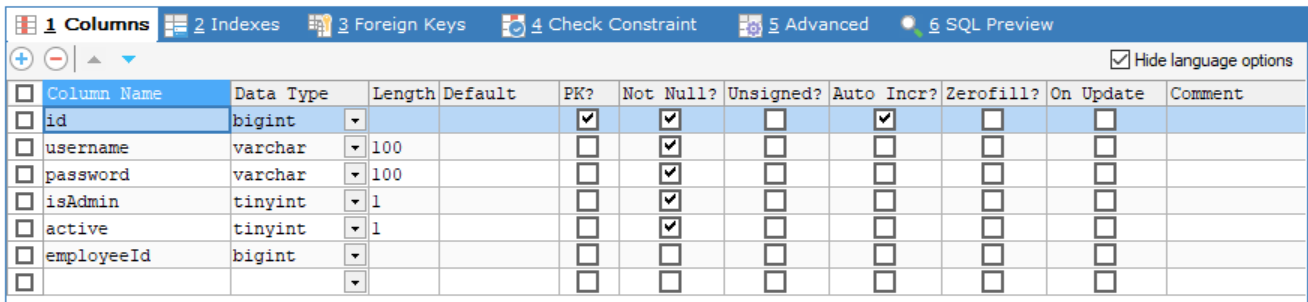
Svaka repository klasa implementira DBRepository interfejs sa sledećim metodama:

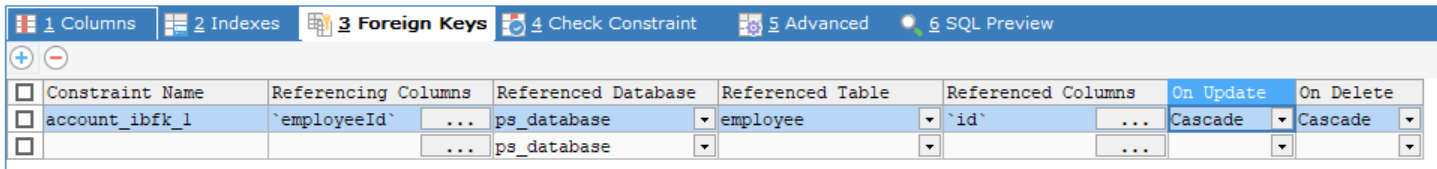
* public void **connect**()
* public void **disconnect**()
* public void **startTransaction**()
* public void **commit**()
* public void **rollback**()

## Projektovanje skladišta podataka

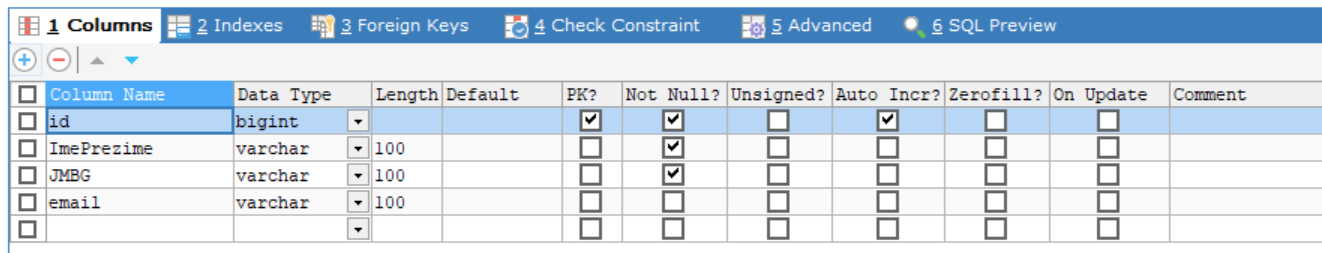
Na osnovu relacionog modele izvšeno je projektovanje tabela baze podataka

Nalog – tabela Account

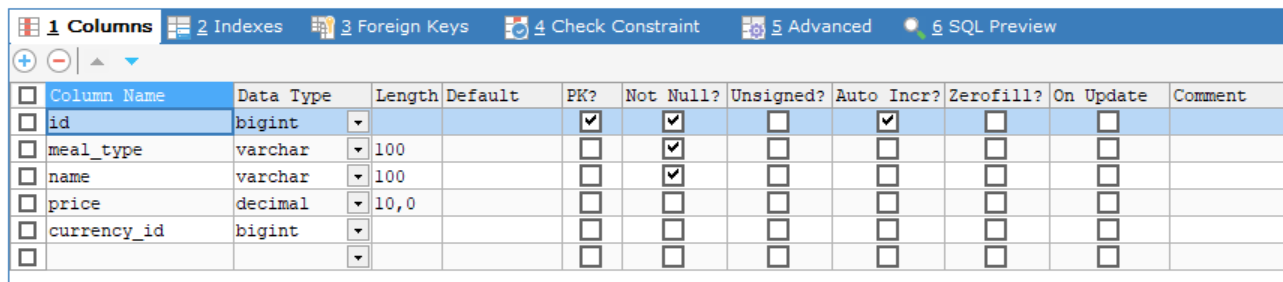


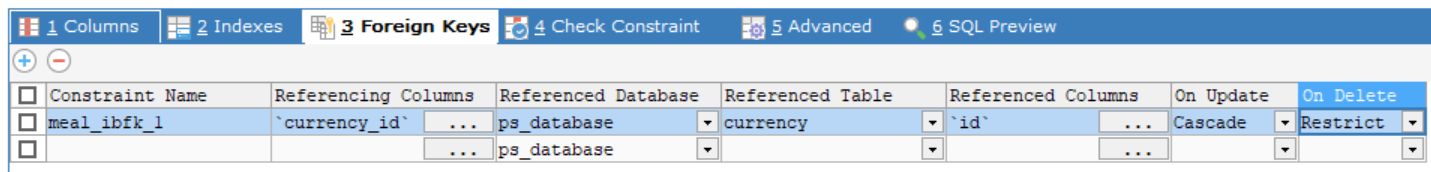


Zaposleni – tabela Employee

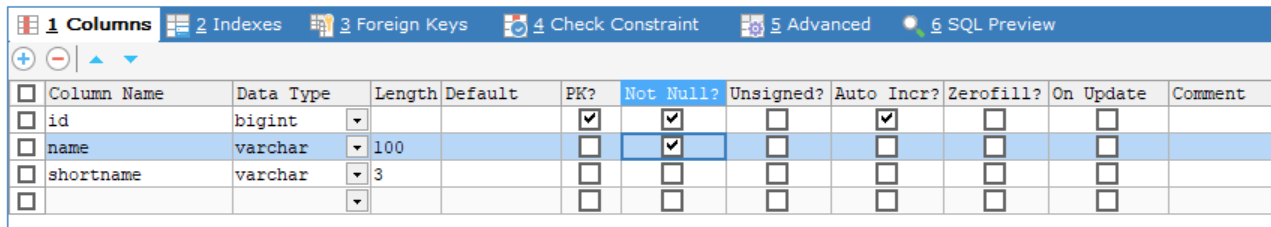


Jelo – tabela Meal

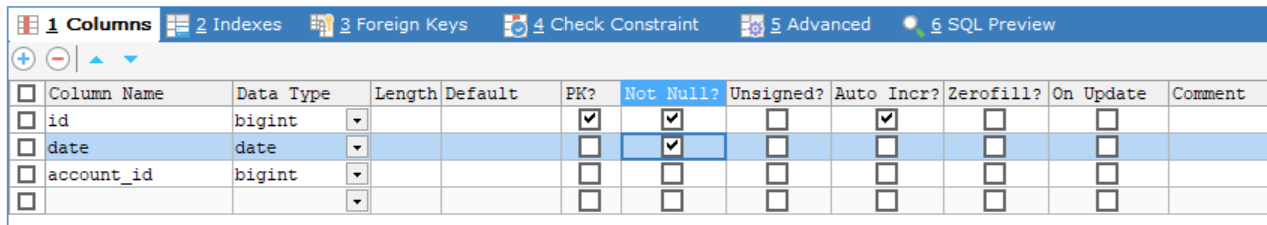


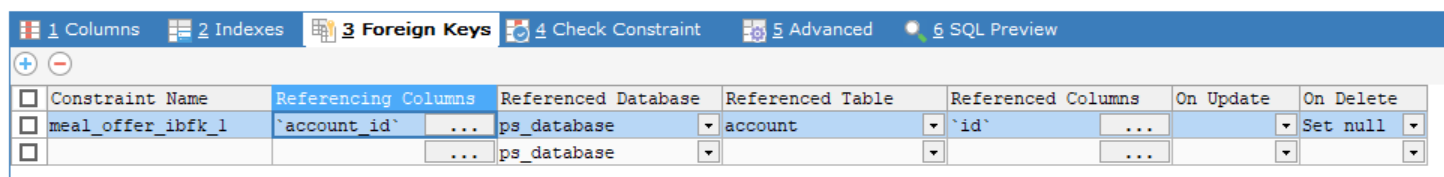


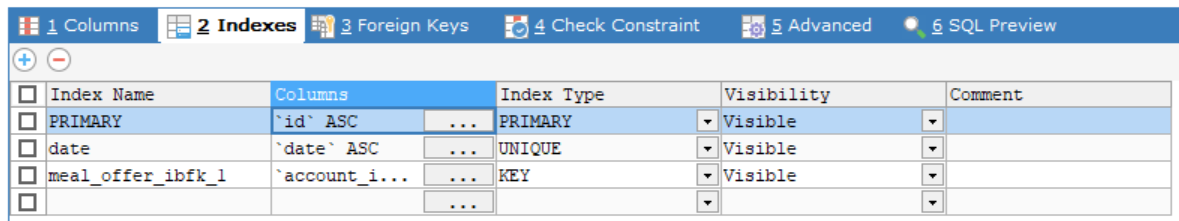
Valuta – tabela Currency



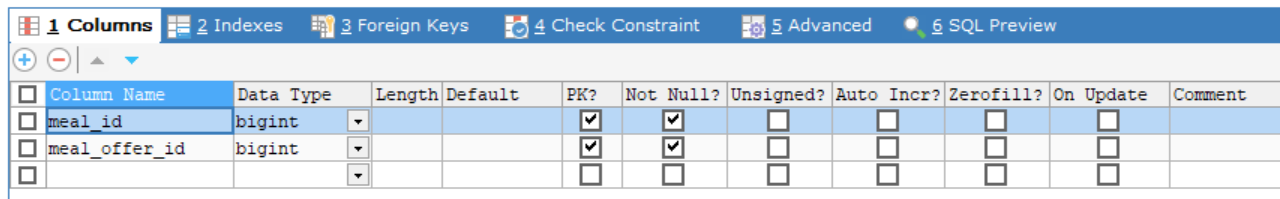
PonudaJela - MealOffer

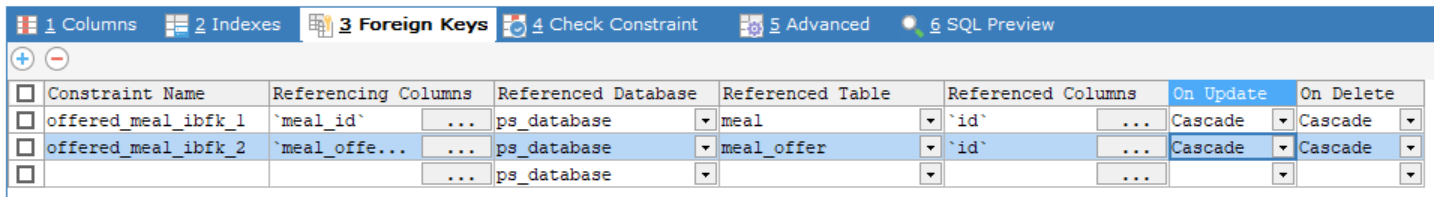




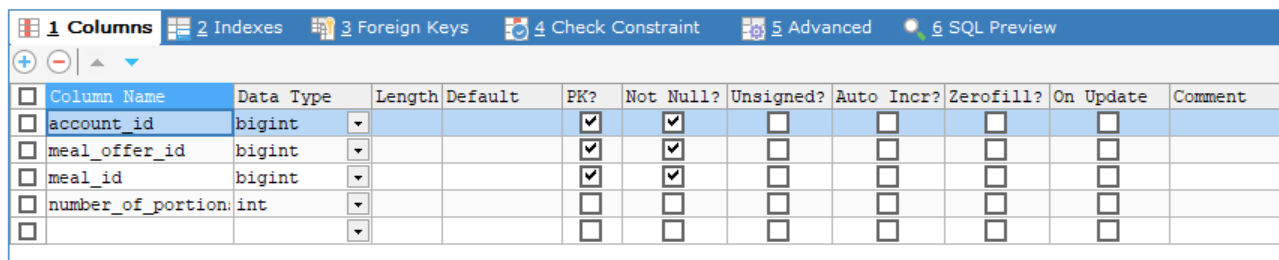


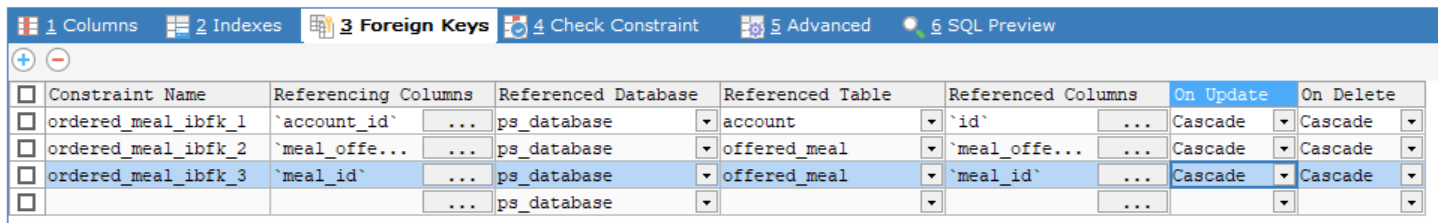
PonudjenoJelo – OfferedMeal





PoručenoJelo – OrderedMeal





# Implementacija

Tehnologije korišćene za implementaciju sistema su:

1. Programski jezik JAVA (verzija 1.8.0\_271)
2. Razvojno okruženje NetBeans
3. MySQL – sistem za upravljanje bazom podataka
4. JDBC – java API za interakciju DBMS-om.

# Testiranje

Sistem je testiran višestrukrim ponavljanjem slučajeva korišćenja u različitom redosledu, kao i unošenjem neočekivanih i nepodrazumevanih podataka kako bi se ispitala robusnost sistema. Skalabilnost sistema testirana je istovremenim korišćenjem sistema od strane 10 korisnika.