

**Jesenji semestar**

**2016/17 PROJEKTNI ZADATAK**

***Dokumentacija sistema za online kupovinu obuće***

**AUTOR:**

**Milanka Bjelić 2626**

**Smer - Softversko inženjersto**

Beograd, Mart 2017.

# Sadržaj:

[Sadržaj: 2](#_Toc440891238)

[1. Uvod 3](#_Toc440891239)

[1.1 Svrha dokumenta 3](#_Toc440891240)

[1.2 Pregled dokumenta 3](#_Toc440891242)

[1.3 Razvojne metodologije 4](#_Toc440891243)

[2. Opis 4](#_Toc440891244)

[2.1 Funkcije sistema 4](#_Toc440891245)

[2.2 Neophodni moduli/stranice 4](#_Toc440891246)

[2.3 Ograničenja sistema 4](#_Toc440891247)

[3. Zahtevi i specifikacije 5](#_Toc440891248)

[3.1 Uvod 5](#_Toc440891249)

[3.2 Korisnici sistema 5](#_Toc440891250)

[3.3Funkcionalni zahtevi 5](#_Toc440891251)

[3.3.1 Administratorski deo 5](#_Toc440891252)

[3.4.2 Deo za korisnika 6](#_Toc440891253)

[3.4 Nefunkcionalni zahtevi 8](#_Toc440891254)

[3.4.1 Performanse 8](#_Toc440891256)

[3.4.2 Lakoća upotrebe 8](#_Toc440891257)

[3.4.3 Zaštita 8](#_Toc440891258)

[3.4.4 Raspoloživost 8](#_Toc440891259)

[3.4.5 Lakoća održavanja 8](#_Toc440891260)

[3.4.6 Bezbednost 8](#_Toc440891261)

[3.6 Slučaji upotrebe (Use cases) 11](#_Toc440891262)

[3.6.1 Administrator 11](#_Toc440891263)

[3.6.2 Korisnik 11](#_Toc440891264)

[4. Dizajn 23](#_Toc440891265)

[4.1 Strukturni dizajn 23](#_Toc440891266)

[4.1.1 Klasni dijagram 23](#_Toc440891267)

[4.1.2 Dizajn baze podataka 24](#_Toc440891268)

[4.2 Arhitektura sistema 26](#_Toc440891269)

[4.3 Infrastruktura sistema 27](#_Toc440891270)

[4.4 Procesni dizajn(dizajn ponašanja) 28](#_Toc440891271)

[4.4.1 Dijagrami sekvenci 28](#_Toc440891272)

[4.4.2 Dijagram aktivnosti 66](#_Toc440891273)

[5. Implementacija i testiranje 80](#_Toc437018471)

[5.1 Implementacija sistema 80](#_Toc437018472)

[5.2 Testiranje sistema 81](#_Toc437018477)

[6. ZAKLJUČAK 81](#_Toc437018478)

[7. Literatura 82](#_Toc437018480)

**1. UVOD**

Ovaj projekat predstavlja dokumentaciju sistema koji uz određene modifikacije može da se koristi u različitim tipovima poslovanja. Ovde će, konkretno, biti dat komplentan uvid u dokumentaciju softvera za online narudžbu obuće. Korišćenjem ovog sistema, prodavnice obuće mogu da unaprede svoje poslovanje omogućavajući kupcima da naruče željeni proizvod i pregledaju trenutnu ponudu bez odlaska u prodajni objekat.

**1.1 Svrha dokumenta**

U okviru ove dokumentacije biće detaljno objašnjeni principi po kojima sistem funkcioniše. Da bi ti principi bili što jasniji i detaljniji, koristiće se različiti dijagrami i tabele koji će omogućiti razumevanje veza između pojedinih sekvenci sistema. Na taj način se sprečava bespotrebno trošenje vremena ukoliko dođe do potrebe da se deo sistema menja. Za izradu samih modela koristi se Power Designer, a što se tiče implementacije moguće je koristiti različite programske jezike.

**1.2 Pregled dokumenta**

Dokumet između ostalog sadrži:

1. Detaljnan uvid u dijagram slučaja korišćenja

2. Detaljan uvid u svaki slučaj korišćenja

3. Sekvencijalne dijagrame svih slučajeva korišćenja

4. Sistem Sekvencijalne dijagrame svih slučajeva korišćenja

5. Detaljan uvid u klasni dijagram

6. Prikaz svake klase, sa pojašnjenjem njenih metoda, atributa, i kolaboraciojom sa ostalim klasama

7. Konceptualni model baze podataka

8. Fizički model baze podataka

9. Detaljan prikaz infrastrukturnog modela

10. Detaljan uvid u dijagram arhitekture softvera

11. Rezultate testiranja pojedinih klasa

**1.3 Razvojne metodologije**

Što se tiče razvojnih metodologija s obzirom da se radi o sistemu u kome su od strarta duboko i precizno definisani opredelili bismo se vodopad(engl. Waterfall) metodu razvoja softvera u kombinaciji sa agilnom metodom koja bi se manifestovala kroz inkrementalni razvoj. To bi značilo da je akcenat na realizaciji početnih, tj. osnovnih zahteva zarad ispunjenja funkcionalnosti, a zatim će naredne verzije softvera proširivati sistem u skladu sa dodatnim potrebama.

**2. OPIS SISTEMA**

**2.1 Funkcije sistema**

Ovaj sistem podrazumeva softver koji omogućava funkcionalnosti za dve grupe korisnika softvera, odnosno za administratore i kupce. U zavisnosti o kojoj grupi je reč, sistem pruža različite funkcionalnosti. Ako je reč o administratorima, sistem pruža funkcionalnosti kao što su, menjanje podataka o proizvodu, dodavanje samog proizvoda u asortiman dostupan kupcima, brisanje i ažuriranje ponudjenog asortimana, itd. Korisniku sistem omogućava funkcionalnosti pomoću kojih može da razgleda proizvode, izabere odgovarajući i poruči ga.

**2.2 Neophodni moduli/stranice**

1. Za korisnike:

Potrebno je implementirati vebsajt kako bi bila omogućena realizacija svih osnovnih funkcionalnosti. Za početak je dovoljna osnovna HTML struktura koja će se kasnije upotpuniti. Proizvodi se unose direktno u bazu, kako bi mogle da se verifikuju funkcionalnosti za korisnike, kao što su izbor proizvoda i njegovo dodavanje u korpu. Kako bi mogli da testiramo komponente, parametri kao što su specifikacija proizvoda se unose ručno.

2. Za administratore:

Potreban je potpuno funkcionalan modul za administratora sistema. Dizajn samog modula treba da bude što jednostavniji. Kada je reč o funkcionalnosti, u pitanju je modul koji je CRUD (engl. Create Read Update Delete) upotpunjen sa dodatnim mogućnostima administratora kao što su pregled i brisanje korisničkih naloga ukoliko je to potrebno.

## 2.3 Ograničenja sistema

Ograničenje trenutnog sistema je što ne postoji globalna baza, u kojoj bi se skladištili podaci svih karika kompanije, već je rađena za funkcionisanje na nivou samo jedne, tačnije postoji lokalna baza u koju se smeštaju podaci samo unutar te prodavnice. Iako je dati nedostatak naveden kao ograničenje sistema, treba naglasiti da se ovaj nedostatak može prevaziđi pažljivim redizajnom određenih komponenata kako bi se postigla željena integracija.

**3. ZAHTEVI I SPECIFIKACIJE**

## 3.1 Korisnici sistema

S obzirom na sve funkcionalnosti koje sistem treba da pokrije, razlikujemo dve korisničke grupe, čije funkcionalnosti će biti različite. Jednu grupu čine korisnisi , tačnije klijeti koji imaju potrebu i želju da koriste usluge online prodavnice obuće, dok drugu grupu čine administratori sistema koji su zaduženi za upravljanje sistemom, tačnije njegovim sadržajem. Detaljan korisnički molud obe grupe korisnika biće posebno analiziran i projektovan u nastavku.

Role koje razlikujemo u okviru sistema su:

1. Administrator
2. Klijent

Zahtevi odnosno slučajevi upotrebe će se definisati na nivou svake role sistema.

## 3.2 Funkcionalni zahtevi

### 3.3.1 Administratorski deo

Administrator će imati mogućnost da se loguje na nalog. Pored toga on može da dodaje nove proizvode što podrazumeva unos njegovog naziva, cene, boje, veličine, opisa, slike i 3D prikaza. Admnistrator, takođe, ima funkcionalnost i da izmeni neki proizvod koji već postoji na sistemu, promenom veličina u kojima je dostupan, boja, promenom cene, opisa, itd. Isto tako, nakon što odabere da izmeni neki postojeći proizvod, adminstrator može taj proizvod da izbriše. Pre nego što izabere proizvod za izmenu ili birsanje.

#### Prijava na nalog

Kako bi adminstrator mogao da pristupi svom nalogu (modulu) neophodno je da se prijavi unošenjem korisničkog imena i lozinke nakon čega će mu se pružiti mogućnost obavljanja ostalih aktivnosti ukoliko uspešno prođe procese autentikacije i autorizacije.

Prijava administratora je neophodna kako bi se zaštitilo obavljanje funkcionalnosti za koje će samo administrator biti privilegovan.

#### Dodavanje proizvoda

Administrator ima mogućnost da doda novi proiyvod, pri čemu treba da unese njegovu cenu, veličine i boje u kojima je proizvod dostupan, sliku i 3D prikaz proizvoda, kao i detaljan opis proizvoda kako bi korisnik imao što bolji uvid u sve detalje proizvoda koji može da naruči.

##### ***Unos veličina proizvoda***

Pošto je izabrao opciju za dodavanje novog proizvoda, korisnik unosi dostupne veličine proizvoda. Veličine mogu biti: XXS, XS, S, M, L, XL i XXL.

##### ***Unos boja proizvoda***

Pošto je izabrao opciju za dodavanje novog proizvoda, korisnik unosi dostupne boje proizvoda.

##### ***Unos naziv proizvoda***

Pri unosu novog proizvoda, administrator dodeljuje naziv proizvodu.

##### ***Unos cene proizvoda***

Pri unosu proizvoda, administrator mora da doda i cenu proizvoda u validnom formatu.

##### ***Unos opisa proizvoda***

Administrator treba da doda opis koji korisniku daje detaljnije informacije o proizvodu (materijal, otpornost na vodu, uputstvo za održavanje itd.).

##### ***Unos slika proizvoda***

Administratoru treba da ubaci sliku prizvoda.

##### ***Unos 3D prikaza proizvoda***

Administratoru treba da ubaci 3D prikaz prizvoda koji će omogućiti korisniku da vidi proizvod iz svih perspektiva, što je vrlo značajno pri kupovini obuće preko interneta.

##### ***Izmena postojećeg proizvoda***

Administrator ima mogućnost da izmeni proizvod koji je već na sistemu. Administrator prvo treba da odabere proizvod unosom šifre proizvoda. Ako proizvod postoji, administrator će moći da izmeni proizvod. Nakon toga on ima mogućnost da ukloni dati proizvod i da izmeni cenu proizvoda.

##### ***Uklanjanje proizvoda***

Administrator može da ukloni unet proizvod. Odabirom ove opcije, administrator prima poruku da potvrdi opcije. Ukoliko potvrdi, proizvod će biti obrisan.

##### ***Izmena cena proizvoda***

Administrator ima mogućnost da izmeni cenu proizvoda upisom nove cene i potvrdom unosa.

### 3.4.2 Deo za korisnika

Kako bi koristio sistem, korisnik prvo mora da napravi nalog.Pošto je kreirao nalog, korisnik mora da se prijavi na svoj nalog kako bi mogao da koristi funkcionalnosti sistema. Korisnik prvo bira tip obuće nakon čega bira konkretan proizvod. Kada izabere konkretan proizvod, korisnik treba da izabere odgovarajuću veličinu i boju, i tek tada da ubaci proizvod u korpu ukoliko želi da ga naruči. Ukoliko želi da naruči proizvod, korisnik prvo mora da unese adresu za dostavu kao i da izabere način plaćanja. Ako izabere plaćanje karicom, mora da popuni i formu sa podacima sa kartice. Korisnik ima i opciju da ukloni proizvod iz korpe ukoliko se predomislio. Takođe može da vidi informacije o firmi i da kontaktira firmu preko telefona ili slanjem poruke na e-mail.

#### Kreiranje naloga

Kako bi mogao da koristi sistem, korisnik treba da napravi nalog. Forma za kreiranje naloga zahteva lične podatke korisnika kao što su: korisničko ime, lozinka, ime, prezime, email i adresa.

##### ***Prijavljivanje na nalog***

Ako je korisnik napravio nalog, može da se uloguje na sistem. Za prijavljivanje na sistem je potrebno da korisnik unese korisničko ime i lozinku koje je odabrao pri koreiranju svog naloga.

##### ***Biranje tipa obuće***

Korisnik bira tip obuće koji ga zanima, nakon čega od sistema dobije proizvode izabranog tipa. Tip obuće može biti: patike, cipele, espadrile, cizme, sandale, večernja obuća.

##### ***Biranje konkretnog proizvoda***

Odabirom tipa, korisnik dobija na uvid sve odevne komade ovog tipa. Nakon toga korisnik treba da izabere konkretan proizvod koji želi da kupi. Odabirom proizvoda, dobija mogućnost da odredi detaljne sprecifikacije proizvoda.

###### **Biranje veličine proizvoda**

Ukoliko korisnik želi da kupi odabran proizvod, treba da izabere veličinu proizvoda koja na početku nije određena ali korisnik može da vidi koje veličine postoje na sistemu. Nakon odabira, korisniku se prikazuje veličina koju je izabrao.

###### **Uvećanje slike**

Da bi imao detljan uvid u proizvod, korisnik može da uveća sliku proizvoda. On ima mogućnost da svaku od pet slika uveća i samim tim bliže pogleda proizvod.

###### **3D prikaz - rotacija**

Kako bi što bolje video izabrani proizvod, korisnik može prelazom preko 3D prikaza da rotira sliku proizvoda i na taj način vidi proizvod iz svih perspektiva.

##### ***Pregled informacija o firmi***

Korisnik ima mogućnost da vidi informacije o firmi (lokacije na kojima se nalazi, otkad postoji, itd.).

#### Kontaktiranje firme

Korisnik mora da ima nalog kako bi kontaktirao firmu. Sistem mu vraća formu koju treba da popuni. Forma traži ime, prezime, email i adresu. Zatim, forma traži poruku korisnika firmi. Potvrdom dobija poruku da je poruka uspešno poslata.

#### Dodavanje proizvoda u korpu

Pošto je prethodno odabrao proizvod, ako želi da naruči proizvod, korisnik treba da ubaci proizvod sa odabranim detaljima u svoju korpu. Korisnik automatski dobija korpu kada napravi nalog.

#### Potvrdi sadržaj/Naruči

Ukoliko je korisnik zadovljan sadržajem korpe, može da naruči proizvode koji se nalaze u korpi. Nakon toga, korisnik će morati da unese detalje o adresi i kreditnoj kartici ukoliko ne plaća pouzećem.

#### Unos adrese za narudžbinu

Nakon što je narudžbenica sastavljena, korisniku se traži da unese podatke o adresi na kojoj želi da primi dostavu. Pritom korsnik treba da unese državu, grad, ulicu i broj. Nakon ovog unosa narudžbina je završena.

#### Biranje načina plaćanja

Korisnik može da odabere da li će da plaća karticom ili pouzećem. Ukoliko bira karticu, neohodno je da unese podatke sa kartice kako bi mogao na taj način da poruči željeni proizvod.

#### Naručivanje

Kako bi konačno potvrdio svoju narudžbu, korisnik dobija formu za potvrdu da li je siguran da želi da naruči odabrane proizvode.

## 3.3 Nefunkcionalni zahtevi

### Nefunkcionalni zahtevi su zahtevi koji nisu direktno povezani sa servisima koje sistem treba da obezbedi svojim korisnicima, već sa svojstvim tih servisa, kao što su: performanse, pouzdanost, brzina odgovora, zauzeće memorije, bezbednost, raspoloživost, prenosivost, lakoća održavanja. Oni definišu ograničenja implementaciji sistema, kao što su na primer, svojstva U/I uređaja, ili predstavljanje podataka na interfejsu sa drugim sistemima.

### 3.3.1 Performanse

Performanse su krucijalni zahtev za većinu sistema. U slučaju web shop-a one nisu imperativ, iz razloga što sistem nije kritičan, ali to ne znači da celokupno kotiranje samog softvera neće biti uvećano ukoliko se sve izvršava u optimalnom vremenu. S obzirom da se sistem skladišti na dva servera (jedan za samu aplikaciju, drugi za bazu podataka) šanse za sporo učitavanje stranica ili izvlačenja sadržaja, tj. informacija iz baze podataka (putem krajnje jednostavnih upita, koji nisu zahtevni ni vremenski ni memorijski) su maksimalno redukovane.

### 3.3.2 Lakoća upotrebe

Sistem zahteva veoma jednostavnu, jednočasovnu obuku, u okviru koje će se administratori upoznati i ovladati svim funkcijama sistema, odnosno modula.

### 3.3.3 Zaštita

Sistem ne zahteva visok stepen zaštite, razlog je što je mogućnost manipulacije sa sadržajem baze podataka ograničen isključivo na modul za administratora, a sami administratori su obezbeđeni autentikacijom i autorizacijom.

### 3.3.4 Raspoloživost

Sajt treba da bude dostupan za korišćenje na svim tipovima uređaja poput tablet urađaja i mobilnih telefona što zahteva prilagodljiv (responsive) dizajn.

### 3.3.5 Lakoća održavanja

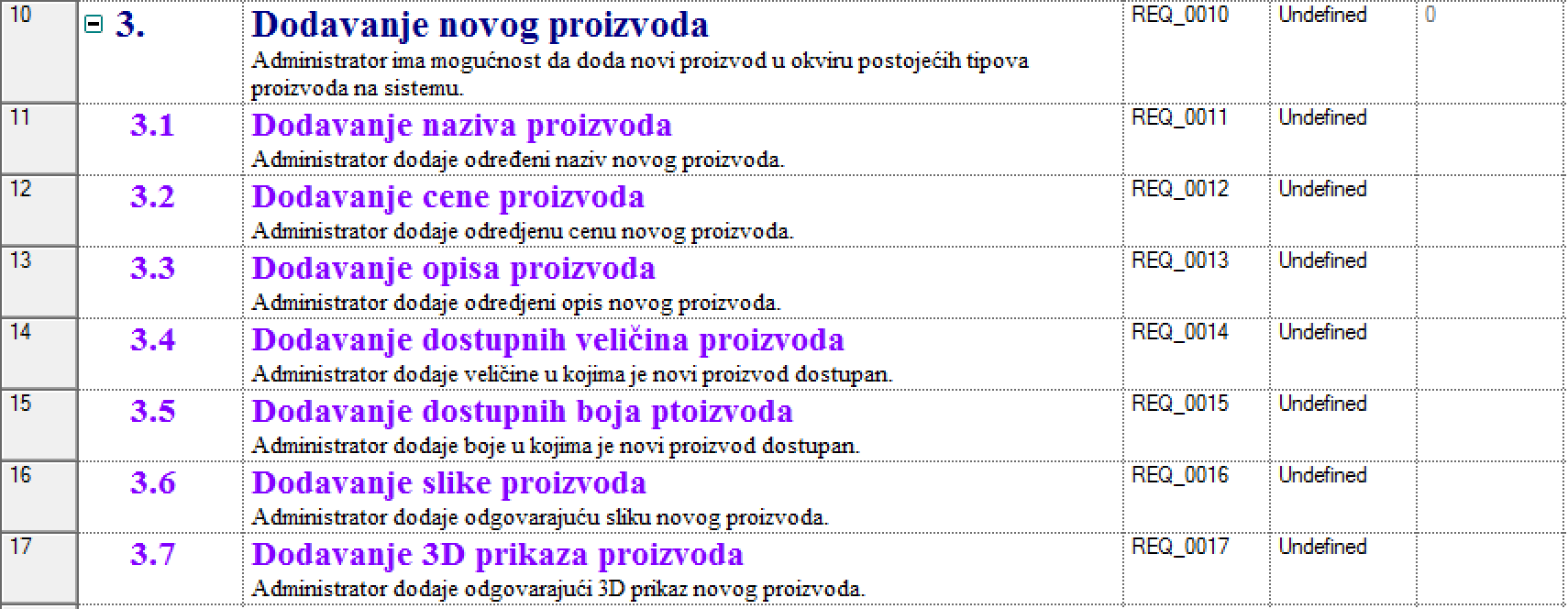
Akcenat je na tome da se olakša razvoj eventualnih naknadnih funkcionalnosti tako što treba težiti da se u što većoj meri koriste postojeće tj. već razvijene komponente.

### 3.3.6 Bezbednost

Najkritičnija komponenta sistema je sama mogućnost elektronskog plaćanja. S da se sve transkacije izvršavaju uz pomoć tzv. Trećeg lica, preostalo je samo da se iskoriste dobri metodi(koji podrazumevaju dobar izbor algoritama uz ''salt'' tehniku) za šifre korisnika pri beleženju i autentikaciji prilikom rada sa bazom podataka. Pri odabiru ''plaćanja po preuzimanju'' postoji verifikacija samog korisnika putem sms-a, što je standardna procedura.



Slika 1.1.1: Pogled na zahteve za administratora I



Slika 1.1.2: Pogled na zahteve za administratora II



Slika 1.2.1: Pogled na zahteve za korisnika I



Slika 1.2.2: Pogled na zahteve za korisnika II

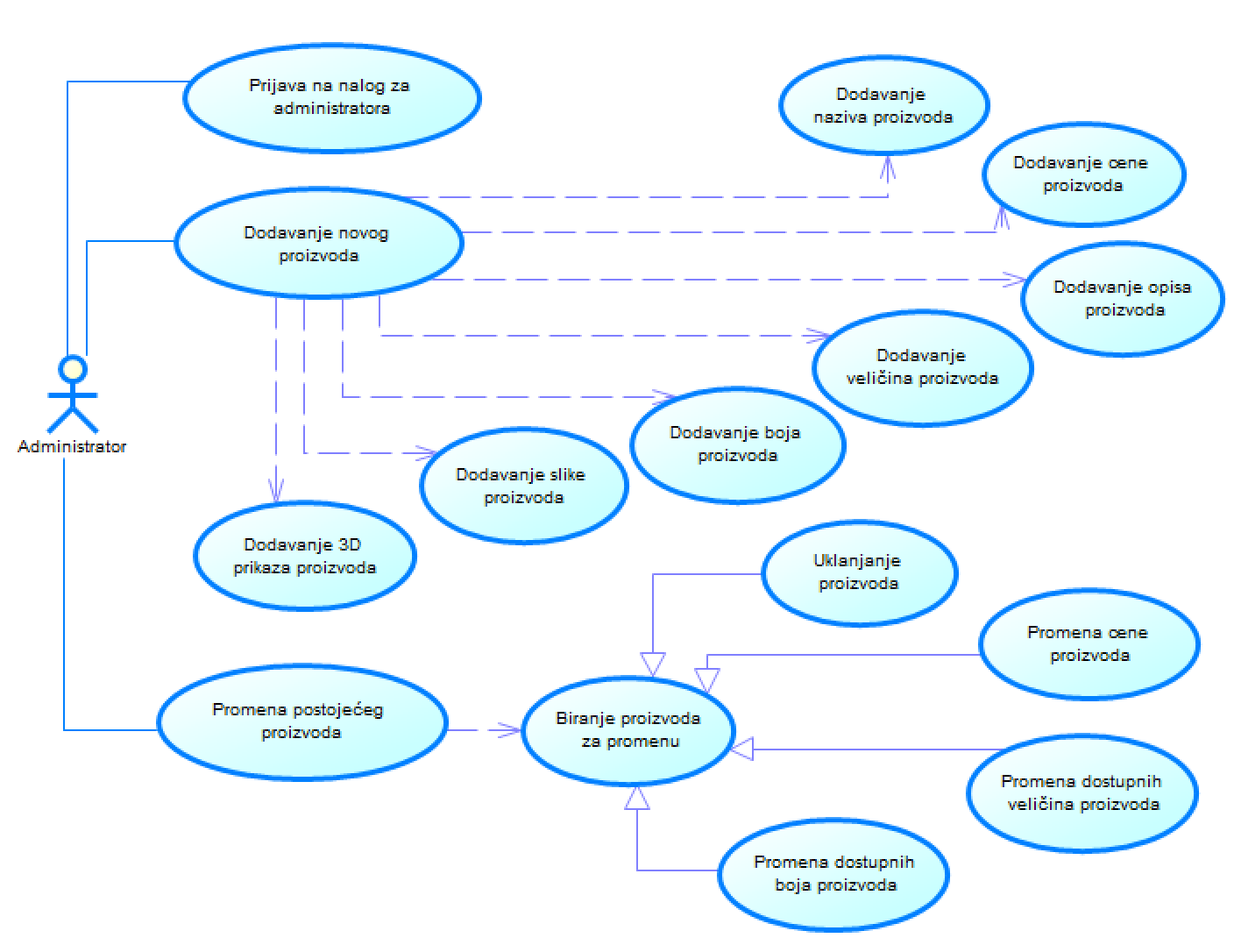


Slika 1.2.3: Pogled na zahteve za korisnika III

## 3.4 Slučaji upotrebe (Use cases)

### 3.4.1 Administrator

Na slici 2.1 možemo videti slučaje korišćenja koji odgovaraju administratoru.



Slika 2.1: Use Case dijagram za administratora

***> Prijava na nalog za administratora***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

Administrator unosi korisničko ime I lozinku kako bi pristupio nalogu I samim tim bio u mogućnosti da koristi aplikaciju kroz odgovarajući modul koji je dizajniran za njega.

**Preduslovi:**

1. Administrator ima pristup internetu.

**Osnovni scenario:**

1. Administrator pristupa stranici na kojoj se nalazi login forma koju treba da ispuni.
2. Popunjava login formu unosom koriničkog imena i lozinka i potvrđuje unos klikom na dugme.
3. Ukoliko su korisničko ime i lozinka odgovarajući, korisnik će biti obavešten u uspešnom logovanju i biće preusmeren na podsistem za administrator-a.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Korisničko ime i lozinka ne postoje, administratoru će se prikazati poruka da je uneo pogrešno korisničko ime ili šifru što će ga ponovo vratiti na korak 2.

2. Sistem prikazuje poruku o grešci i nudi korisniku opciju da kreira nalog.

**Postuslovi:**

1. Administrator je obavešten o uspešnom prijavljivanju, preusmeren je na svoj podsistem i može da nastavi sa korišćenjem aplikacije.

***> Dodavanje novog proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnistrator ima mogućnost da doda novi proizvoda nakon što se ulogovao na sistem.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.

**Osnovni scenario:**

1. Administrator bira opciju “Dodaj novi proizvod”.
2. Sistem vraća formu koju administrator treba da popuni potrebnim podacima.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Postuslovi:**

1. Administrator može da doda novi proizvod sa svim potrebnim podacima.

***> Dodavanje naziva proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator dodaje naziv proizvoda nakon što je izabrao opciju za dodavanje novog proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za dodavanje novog proizvoda.

**Osnovni scenario:**

1. Administrator bira opciju “Dodaj proizvod”.
2. Sistem vraća polje za unos naziva proizvoda.
3. Administrator unosi naziv prozivoda.
4. Sistem vraća poruku o uspešno unetom nazivu proizvoda.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator nije uneo validan naziv proizvoda.
2. Sistem vraća poruku o nevalidnom nazivu i ponovo vraća polje za unos naziva.

**Postuslovi:**

1. Administrator je obavešten o uspešnom dodavanju naziva proizvoda.

***> Dodavanje cene proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator dodaje cenu proizvoda nakon što je izabrao opciju za dodavanje novog proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za dodavanje novog proizvoda.

**Osnovni scenario:**

1. Administrator bira opciju “Dodaj proizvod”.
2. Sistem vraća polje za unos cene proizvoda.
3. Administrator unosi cenu prozivoda.
4. Sistem vraća poruku o uspešno unetoj ceni proizvoda.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator nije uneo validan format cene proizvoda.
2. Sistem vraća poruku o nevalidnom formatu cene i ponovo vraća polje za unos cene.

**Postuslovi:**

1. Administrator je obavešten o uspešnom dodavanju cene proizvoda.

***> Dodavanje opisa proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator dodaje opis proizvoda nakon što je izabrao opciju za dodavanje novog proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za dodavanje novog proizvoda.

**Osnovni scenario:**

1. Administrator bira opciju “Dodaj proizvod”.
2. Sistem vraća polje za unos opisa proizvoda.
3. Administrator unosi opis prozivoda.
4. Sistem vraća poruku o uspešno unetom opisu proizvoda.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator nije uneo validan opis proizvoda.
2. Sistem vraća poruku o nevalidnom opisu i ponovo vraća polje za unos naziva.

**Postuslovi:**

1. Administrator je obavešten o uspešnom dodavanju opisa proizvoda.

***> Dodavanje veličina proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator dodaje veličine proizvoda nakon što je izabrao opciju za dodavanje novog proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za dodavanje novog proizvoda.

**Osnovni scenario:**

1. Administrator bira opciju “Dodaj proizvod”.
2. Sistem vraća polje za unos veličina proizvoda.
3. Administrator unosi veličine prozivoda.
4. Sistem vraća poruku o uspešno unetim veličinama proizvoda.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator nije uneo ni jednu veličinu proizvoda.
2. Sistem ponovo vraća polje za unos veličina proizvoda.

**Postuslovi:**

1. Administrator je obavešten o uspešnom dodavanju veličina proizvoda.

***> Dodavanje boja proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator dodaje boje proizvoda nakon što je izabrao opciju za dodavanje novog proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za dodavanje novog proizvoda.

**Osnovni scenario:**

1. Administrator bira opciju “Dodaj proizvod”.
2. Sistem vraća polje za unos boja proizvoda.
3. Administrator unosi boje prozivoda.
4. Sistem vraća poruku o uspešno unetim bojama proizvoda.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator nije uneo ni jednu boju proizvoda.
2. Sistem vraća polje za unos boja.

**Postuslovi:**

1. Administrator je obavešten o uspešnom dodavanju boja proizvoda.

***> Dodavanje slike proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator dodaje sliku proizvoda nakon što je izabrao opciju za dodavanje novog proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za dodavanje novog proizvoda.

**Osnovni scenario:**

1. Administrator bira opciju “Dodaj proizvod”.
2. Sistem vraća polje za unos slike proizvoda.
3. Administrator unosi slike prozivoda.
4. Sistem vraća poruku o uspešnom unosu slike proizvoda.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator nije uneo sliku proizvoda.
2. Sistem vraća polje za unos slike.

**Postuslovi:**

1. Administrator je obavešten o uspešnom dodavanju slike proizvoda.

***> Dodavanje 3D prikaza proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator dodaje 3D prikaz proizvoda nakon što je izabrao opciju za dodavanje novog proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za dodavanje novog proizvoda.

**Osnovni scenario:**

1. Sistem bira opciju “Dodaj proizvod”.
2. Sistem vraća polje za dodavanje 3D prikaza proizvoda.
3. Administrator dodaje 3D prikaz prozivoda.
4. Sistem vraća poruku o uspešnom dodavanju 3D prikaza proizvoda.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator nije uneo 3D prikaz proizvoda.
2. Sistem vraća polje za dodavanje 3D prikaza.

**Postuslovi:**

1. Administrator je obavešten o uspešnom dodavanju 3D prikaza proizvoda.

***> Promena postojećeg proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator bira opciju za promenu proizvoda koji već postoji na sistemu.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za promenu postojećeg proizvoda.

**Osnovni scenario:**

1. Administrator bira opciju “Promeni postojeći proizvod”.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Postuslovi:**

1. Administrator je uspešno izabrao opciju za izmenu proizvoda.

***> Biranje proizvoda za promenu***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator bira proizvod kome želi da promeni nešto tako što unosi šifru proizvoda koji želi da menja.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za promenu postojećeg proizvoda.

**Osnovni scenario:**

1. Sistem vraća formu za odabir proizvoda za izmenu.
2. Admnistrator unosi šifru proizvoda koji želi da izmeni.
3. Sistem vraća stranicu za izmenu izabranog proizvoda.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator je uneo šifru proizvoda koja ne postoji.
2. Sistem ponovo vraća formu za unos šifre proizvoda.

**Postuslovi:**

1. Administrator je uspešno izabrao proizvod za izmenu.

***> Uklanjanje proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator bira opciju za uklanjanje proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za promenu postojećeg proizvoda.
4. Admnistrator je uspešno izabrao proizvod za izmenu.

**Osnovni scenario:**

1. Administrator bira opciju “Ukloni proizvod”.
2. Sistem vraća formu za potvrdu o uklanjanju proizvoda.
3. Administrator potvrđuje brisanje.
4. Sistem vraća poruku o uspešnom brisanju.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator bira „Ne“.
2. Sistem vraća formu za izmenu proizvoda.

**Postuslovi:**

1. Administrator je uspešno izbrisao proizvod.

***> Promena cene proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator bira opciju za promenu cene proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za promenu postojećeg proizvoda.
4. Admnistrator je uspešno izabrao proizvod za izmenu.

**Osnovni scenario:**

1. Administrator bira opciju “Izmeni cenu”.
2. Sistem vraća formu za izmenu cene proizvoda.
3. Administrator potvrđuje unos cene.
4. Sistem vraća poruku o uspešnoj promeni cene proizvoda.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator nije uneo validan format cene.
2. Sistem ponovo vraća formu za izmenu cene proizvoda.

**Postuslovi:**

1. Administrator je uspešno promenio cenu proizvoda.

***> Promena boja proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator bira opciju za promenu boja proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za promenu postojećeg proizvoda.
4. Admnistrator je uspešno izabrao proizvod za izmenu.

**Osnovni scenario:**

1. Administrator bira opciju “Promeni boje proizvoda”.
2. Sistem vraća formu za izmenu boja proizvoda.
3. Administrator menja boje proizvoda.
4. Sistem vraća poruku o uspešnoj promeni boja proizvoda.

**Izuzeci:**

1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator nije uneo ni jednu boju proizvoda.
2. Sistem ponovo vraća formu za izmenu boja proizvoda.

**Postuslovi:**

1. Administrator je uspešno promenio boje proizvoda.

***> Promena veličina proizvoda***

**Učesnici:**

Administrator i sistem.

**Opis slučaja korišćenja:**

1. Admnisitrator bira opciju za promenu veličina proizvoda.

**Preduslovi:**

1. Admnistrator ima pristup internetu.
2. Admnistrator se prijavio na nalog.
3. Admnistrator je izabrao opciju za promenu postojećeg proizvoda.
4. Admnistrator je uspešno izabrao proizvod za izmenu.

**Osnovni scenario:**

1. Administrator bira opciju “Izmeni veličine”.
2. Sistem vraća formu za izmenu veličina proizvoda.
3. Administrator menja veličine proizvoda.
4. Sistem vraća poruku o uspešnoj promeni veličina proizvoda.

**Izuzeci:**

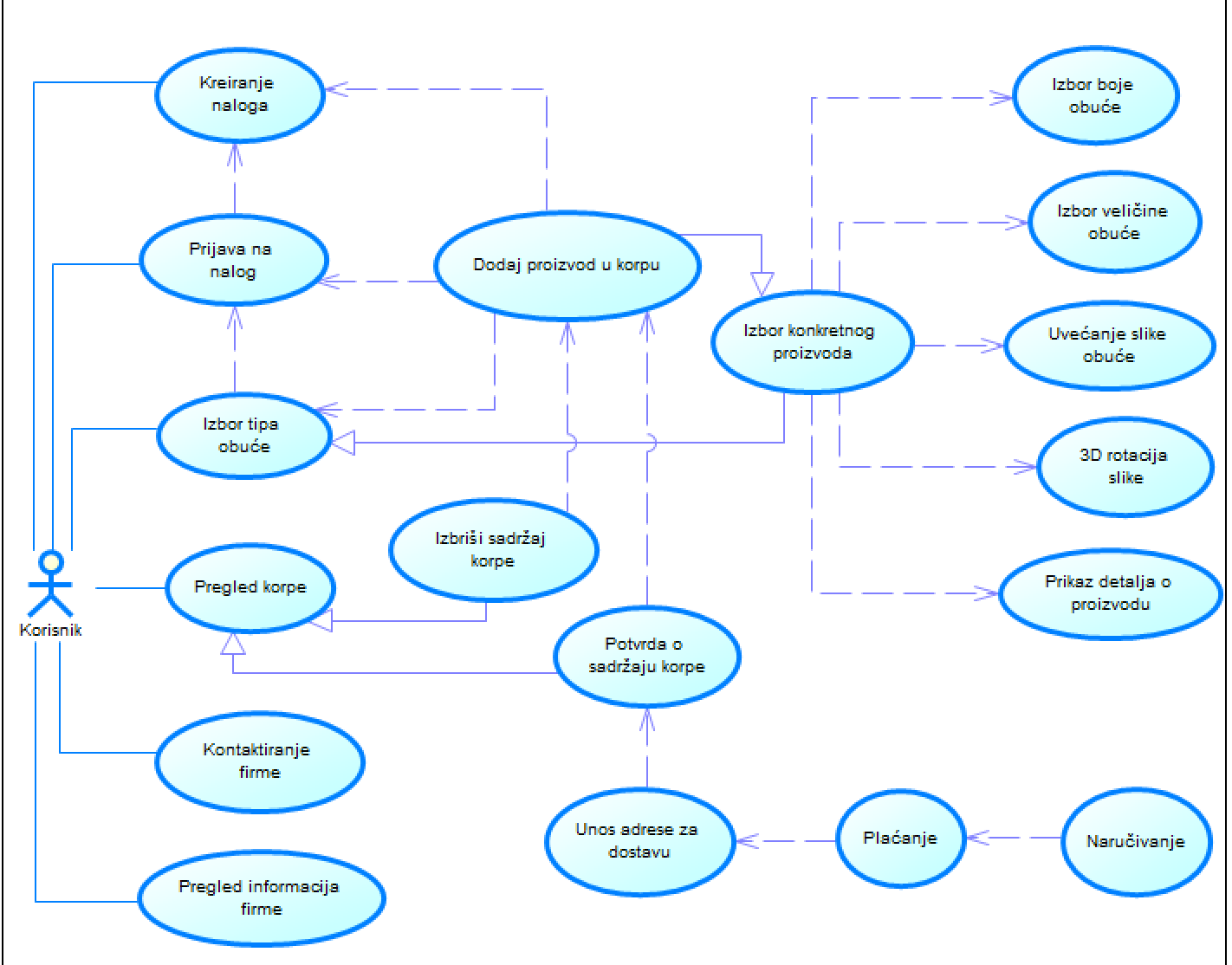
1. Administrator je izgubio konekciju sa internetom.

**Alternativni scenariji:**

1. Administrator nije uneo ni jednu veličinu proizvoda.
2. Sistem ponovo vraća formu za izmenu veličina proizvoda.

**Postuslovi:**

1. Administrator je uspešno promenio veličine proizvoda.



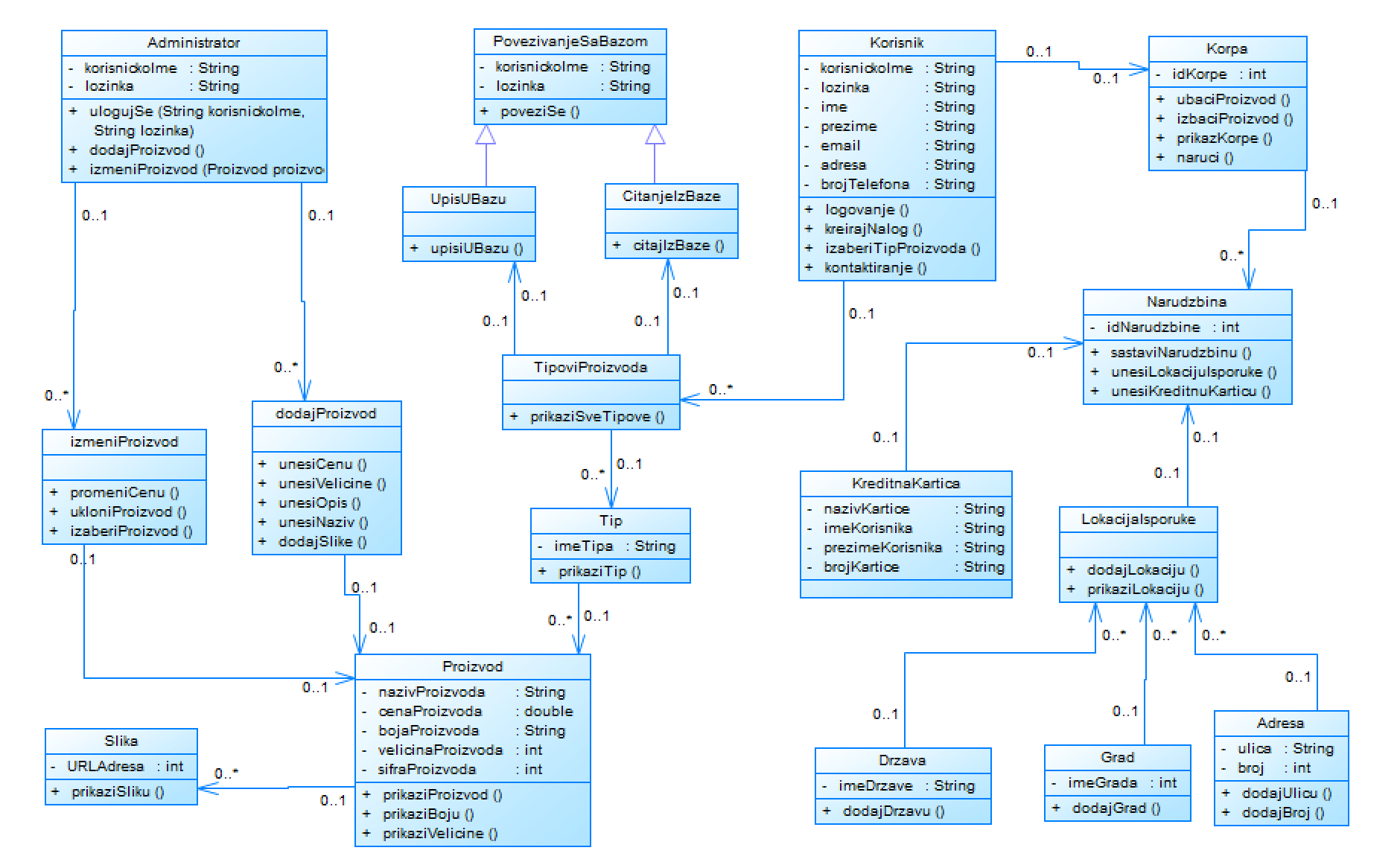
Slika 2.2: Use Case dijagram za korisnika

## 4. Dizajn

## 4.1 Strukturni dizajn

### 4.1.1 Klasni dijagram

Na slici 5 se može videti prikaz klasnog dijagrama gde je definisana primarna struktura klasa kao i relacije između njih.



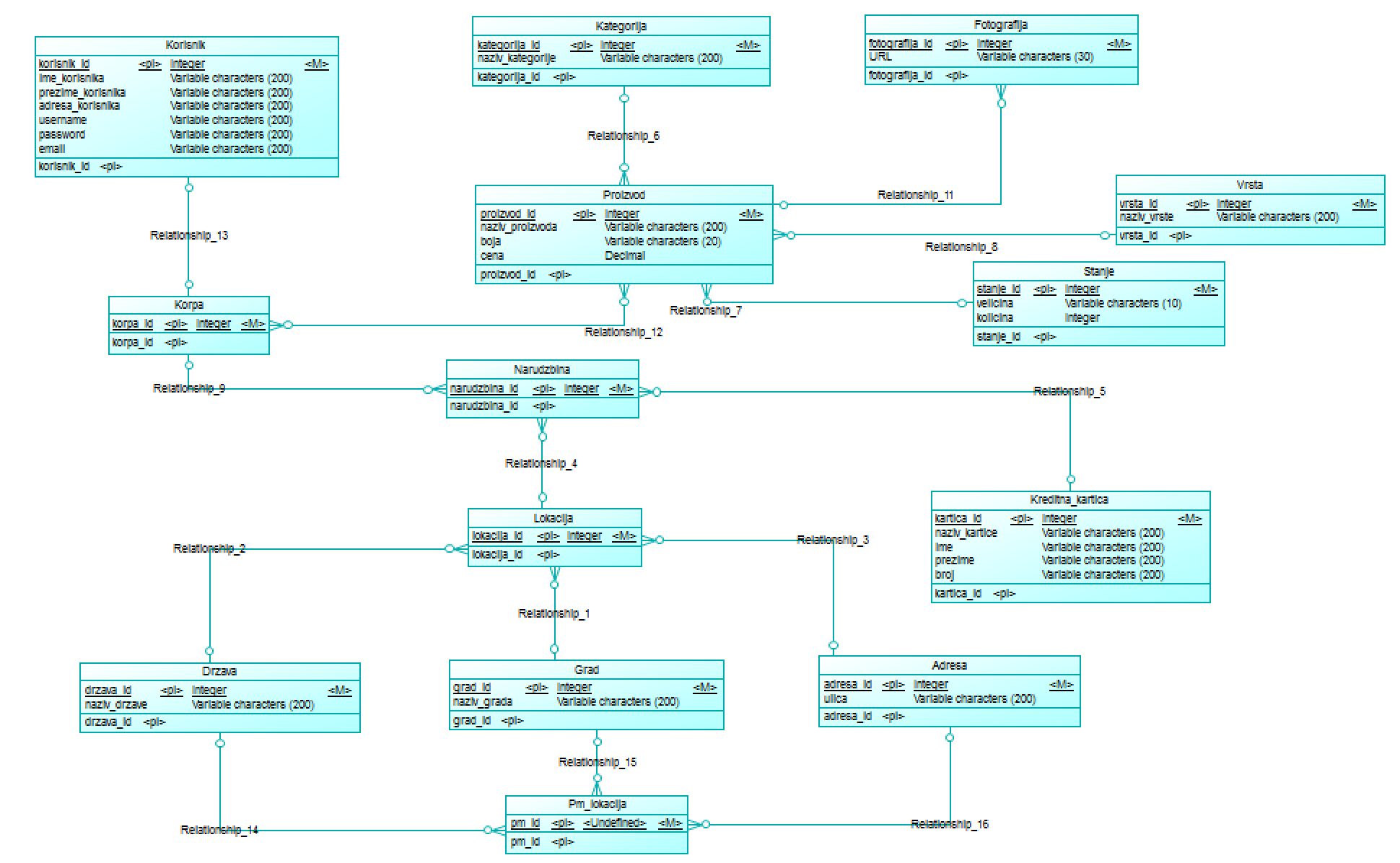
Slika 3: Struktura entitet klasa izgenerisanih na osnovu baze podataka

Na slici se može videti da je ključna klasa Proizvod koja direktno ili indirektno povezana sa svim ostalim klasama. Sve klase predstavljaju Java POJO klase što podrazumeva da sadrže metode getere i setere za svaki atribut kao i toString metodu koje su izostavljene u dijagramu zbog preglednosti. Dakle, s obzirom da će se implementirati MVC arhitektura projekta pomenute klase predstavljaju model i koriste se za predstavljanje podataka sa kojima će aplikacija manipulisati.

### 4.1.2 Dizajn baze podataka

#### Konceptualni dijagram

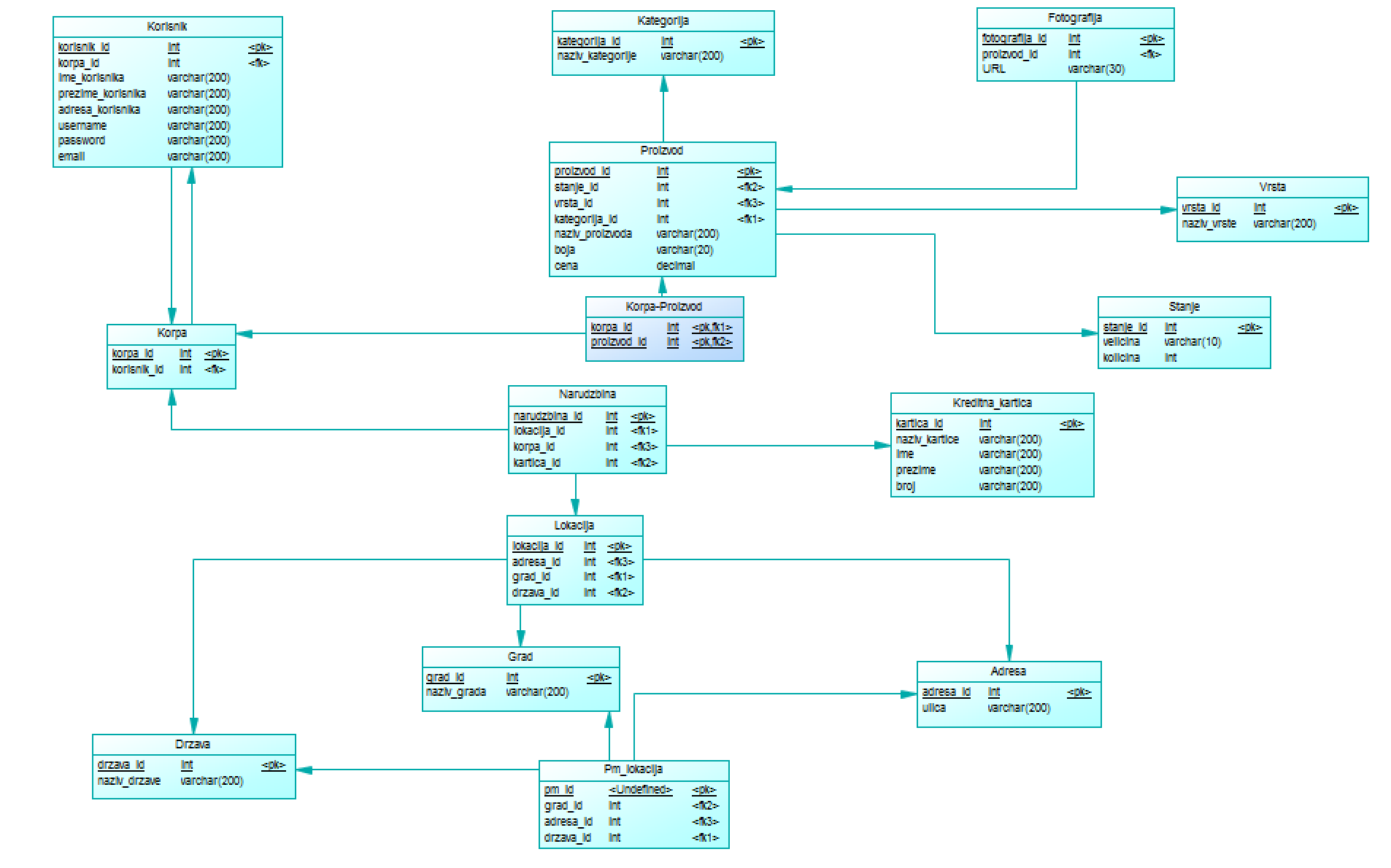
Na slici 4 se može videti konceptualni dijagram klasa.



Slika 4: Konceptualni model baze podataka

#### Fizički model baze podataka

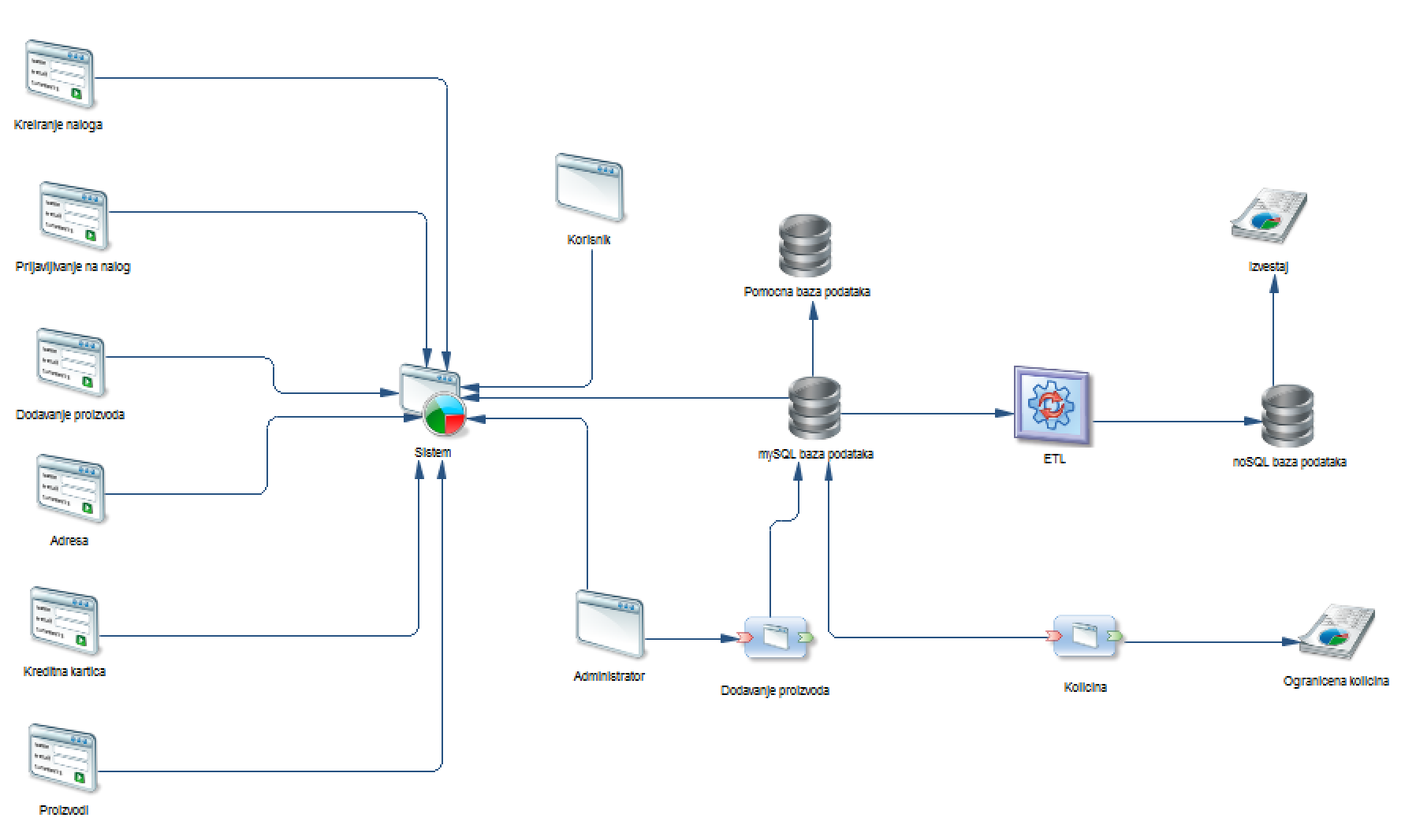
Na slici 5 se može videti fizički model baze podataka.



Slika 5: Fizički model baze podataka

## 4.2 Arhitektura sistema

Na slici 6 je prikazan dijagram arhitekture sistema.



Slika 6: Dijagram arhitekture

Prvenstveno je potrebno obezediti bazu podataka koja će predstavljati relacionu mySQL bazu podataka. U slučaju nepredviđenih problema(kao što su pad servera) kompanija, ne sme da dozvoli sebi prekid usluga, stoga je rezervna baza od krucijalnog značaja. Servis Insertion služi za manipulaciju nad bazom podataka s tim što ovaj servis nudi i osnovne CRUD (Create Read Update Delete) operacije, a I malo kompleksnije upite. Servis Quantity služi za izveštavanje administratora o tome da je neki proizvod dostigao najmanju dozvoljenu granicu kada se radi o količini. (I da bi zalihe trebalo obnoviti). Treba naglasiti da je ETL (Extract, Transform, Load) servis usklađen sa regularnim promenama stanja baze podataka. On smešta podatke u noSQL bazu, radi dalje analize.

Komponente Login, Credit Card, Products, Location, Create Account, Add Product predstavljaju komponente koje će biti u unutar stranica samog sistema.

*Login* – predstavlja front-end komponentu koja se prikazuje prilikom pokušaja logovanja.

*Credit Card* – predstavlja takođe front-end komponentu koja se prikazuje prilikom unošenja podataka o kreditnoj kartici.

*Product –* predstavlja front end komponentu koja će biti ispunjena informacijama o proizvodu.

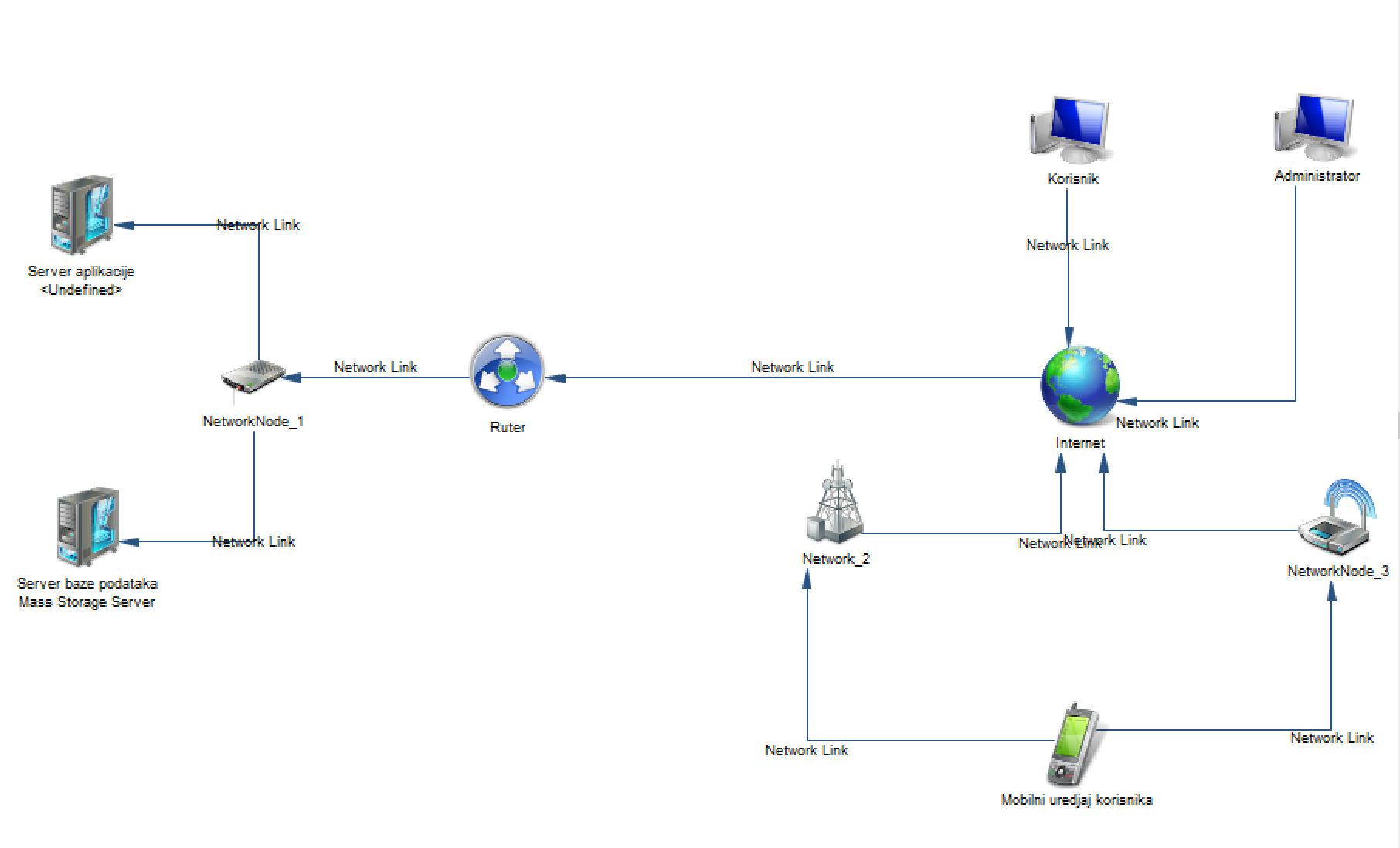
*Location -*  predstavlja front end komponentu, tj. formu koju korisnik treba da ispuni informacijama o lokaciji na kojoj želi da primi pošiljku.

*Create Account -* predstavlja front end komponentu, tj. formu koju korisnik treba da ispuni osnovnim informacijama o sebi.

*Add Product –* takođe predstavlja front end komponentu, ali u okviru modula za administratore. (Funkcionalnost podrazumeva osnove CRUD operacije).

## 4.3 Infrastruktura sistema

Na slici 7 se može videti dijagram infrastrukture sistema.



Slika 7: Dijagram infrastrukture

Ovaj dijagram ukazuje na krajnje jednostavnu infrastrukturu – uključuje najosnovnije elemente, neopohodne za funkcionisanje jednog veb sistema. Administrator može pristupiti sistemu putem interneta preko bilo kog računara (izostavljena je mogućnost pristupa putem mobilnog telefona, iz bezbednosnih razloga), dok je klijentima omogućen pristup sistemu takođe koristeći računar, ali i putem mobilnih uređaja. Firewall nije neizostavan element, s obzirom da dobro konfigurisani serveri nemaju nikakve beneficije od istog. Treba istaći hub koji kontroliše tok saobraćaja, između dva servera, jedan koji služi za skladištenje same aplikacije, drugi kao skladište same baze podataka. Na taj način se može izbeći preterano opterećenje jednog od ova dva servera, ukoliko sam sistem mora da odgovara na zahteve mnogobrojnih korisnika. Na serveru bi se nalazili svi elementi koji su navedeni u dijagramu arhitekture softvera.

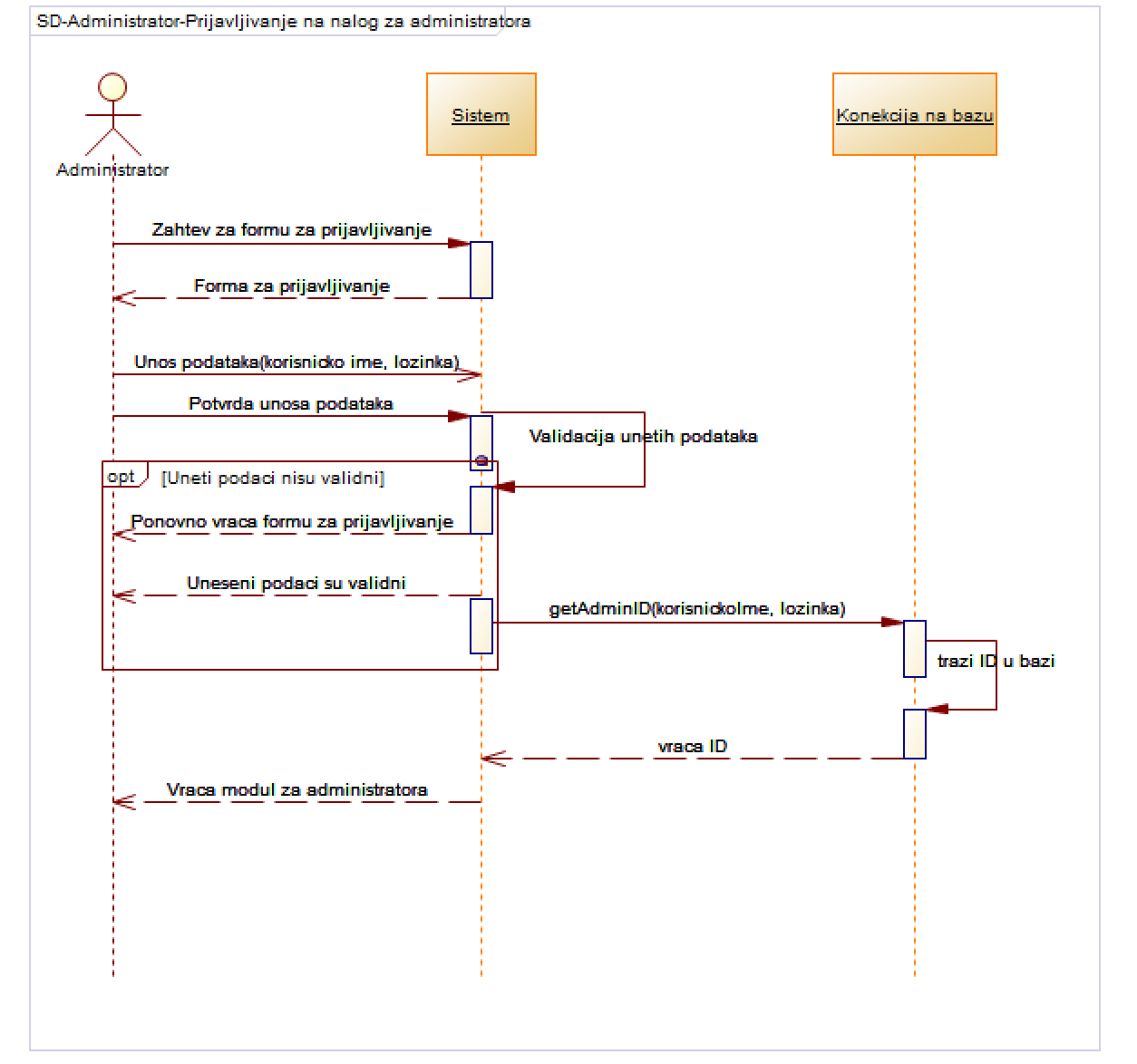
## 4.4 Procesni dizajn(dizajn ponašanja)

### 4.4.1 Dijagrami sekvenci

Za svaki od osnovnih scenarija koji su opisani biće reprezentovan dijagram sekvenci i pored toga će biti opisan tok interakcije u izuzetnim situacijama.

#### Administrator

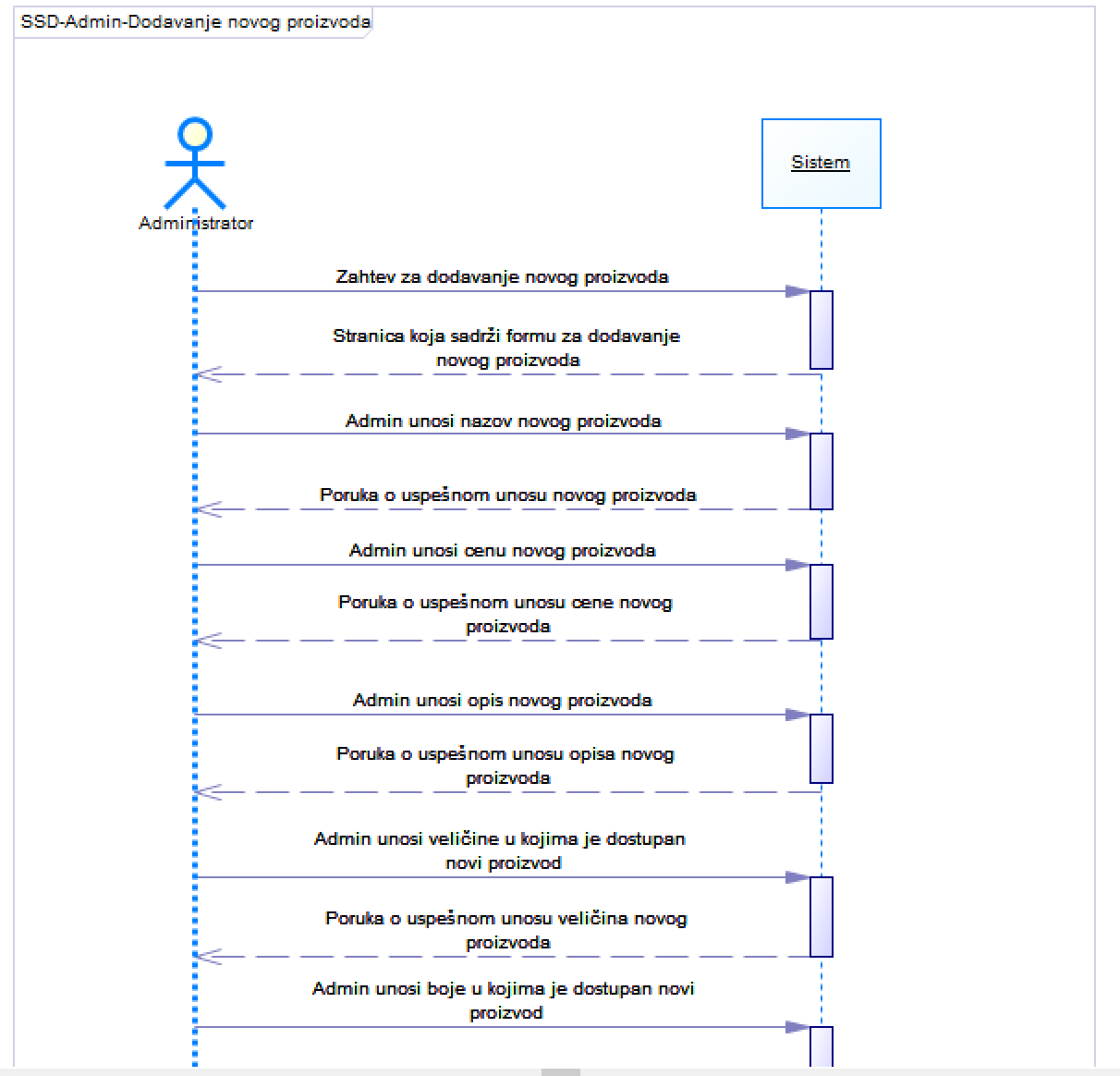
#### Prijavljivanje na nalog



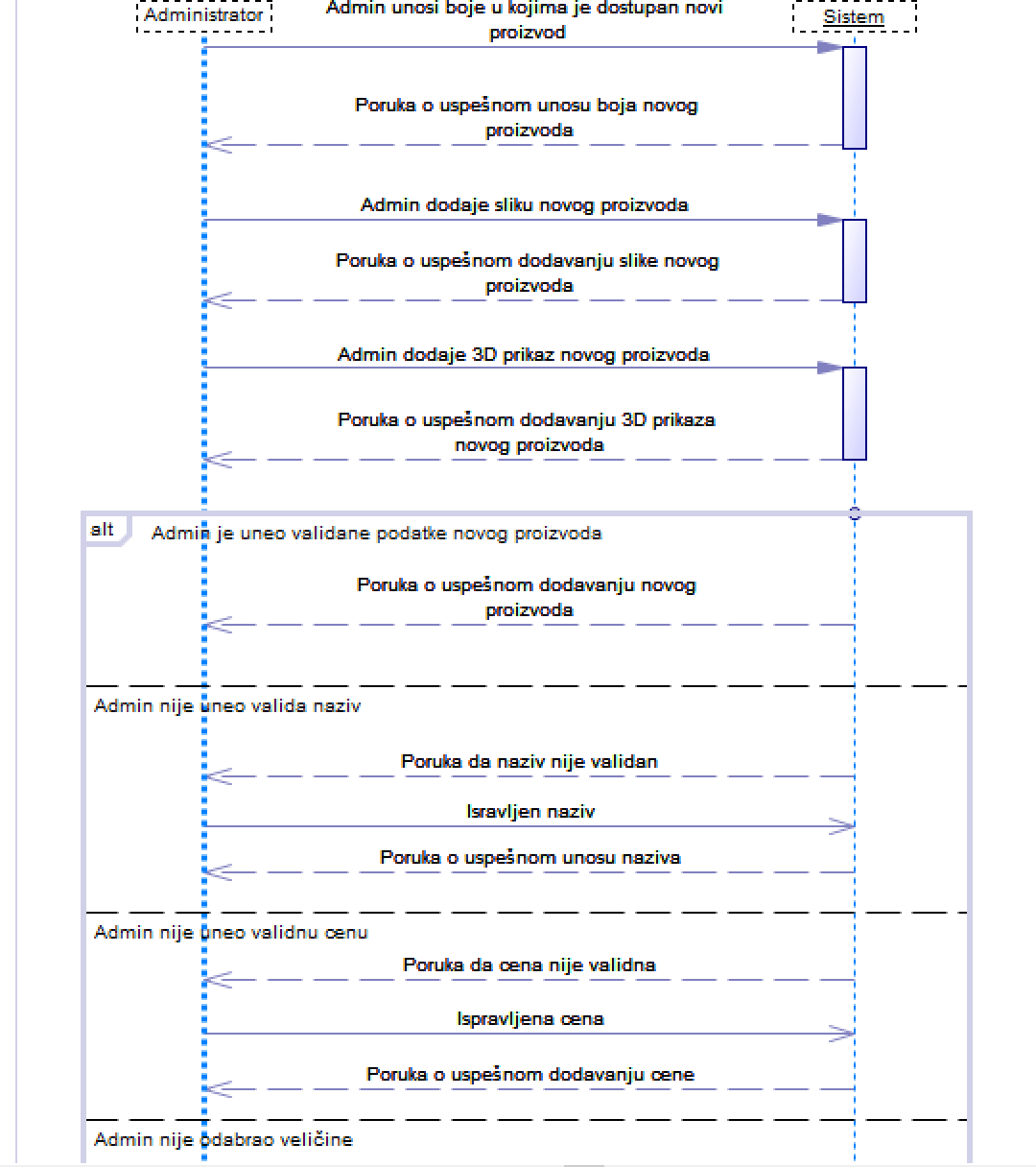
Slika 8.1: SD- Prijava administratora

U okviru provere korisničkog imena i lozinke sistem proverava da li je administratoru kreiran nalog sa definisanim korisničkim imenom kao i da li je šifra ispravno uneta. Na slici je prikazan scenarijo ukoliko su uneti odgovarajuće koriničko ime i šifra. U protivnom sistem će administratoru prikazati poruku da je uneo pogrešno korisničko ime ili šifru i na tom mestu bi se dijagram završio.

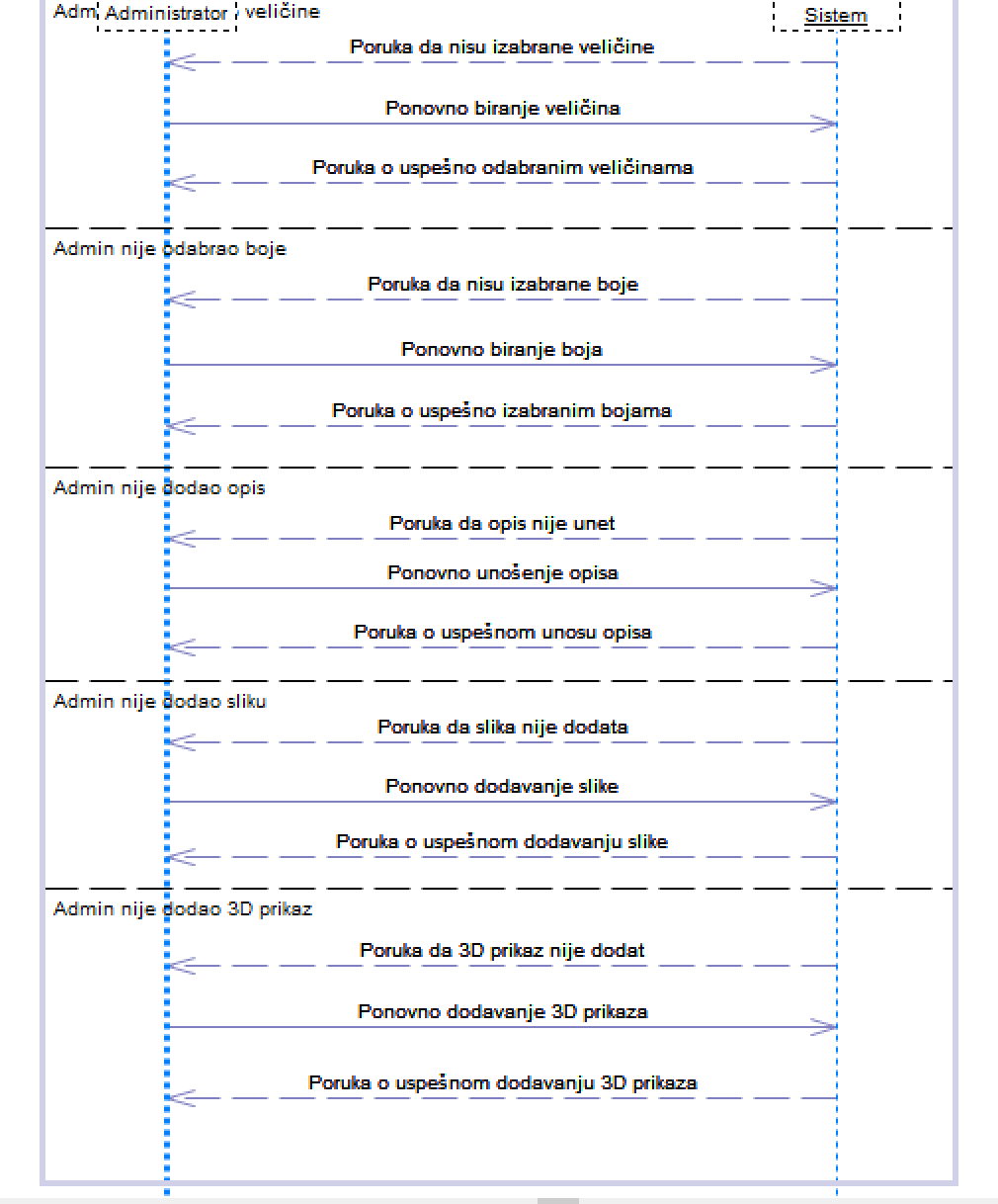
#### Dodavanje proizvoda



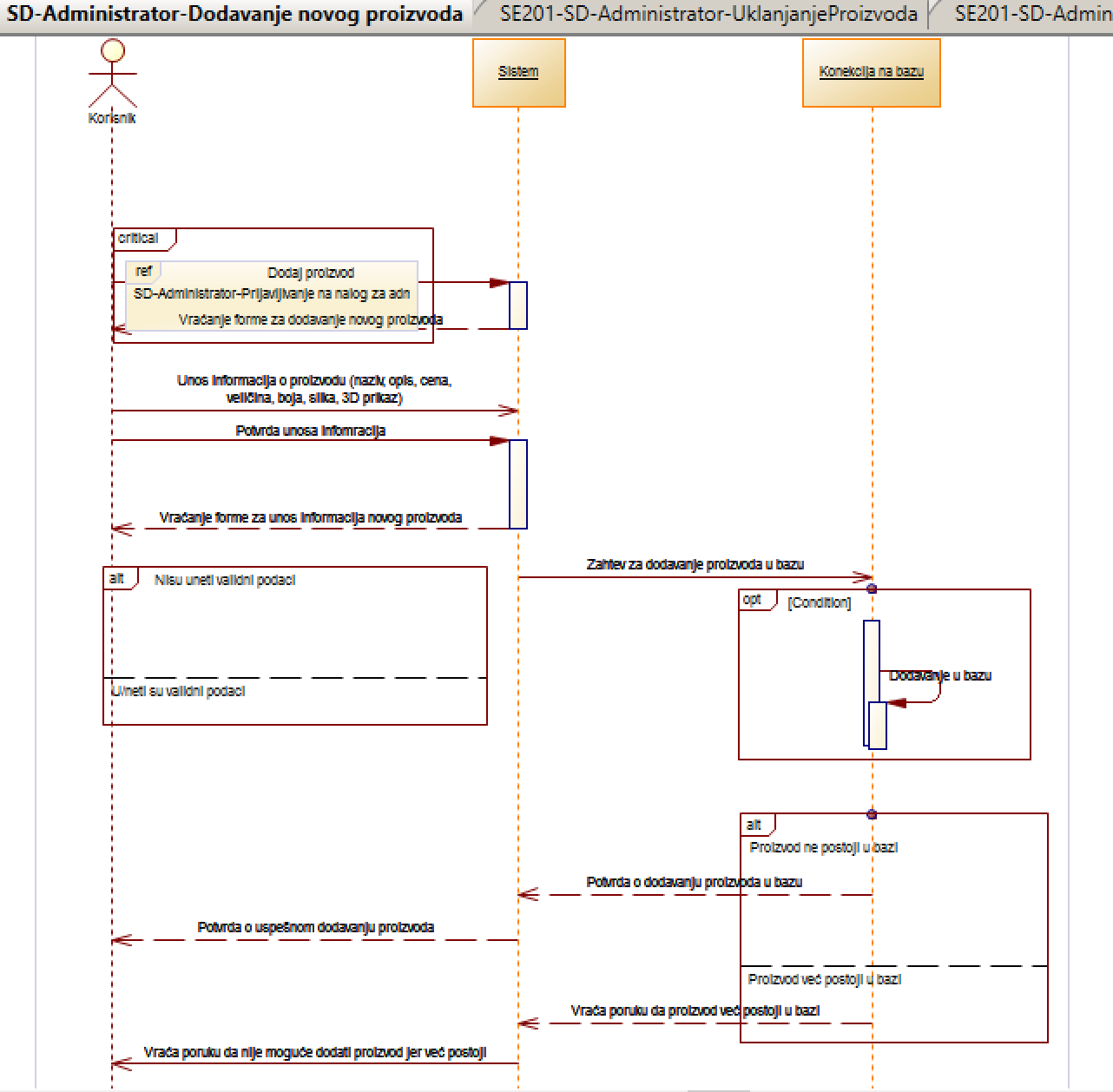
Slika 8.2.1: SSD-Dodavanje proizvoda I



Slika 8.2.2: SSD-Dodavanje proizvoda II



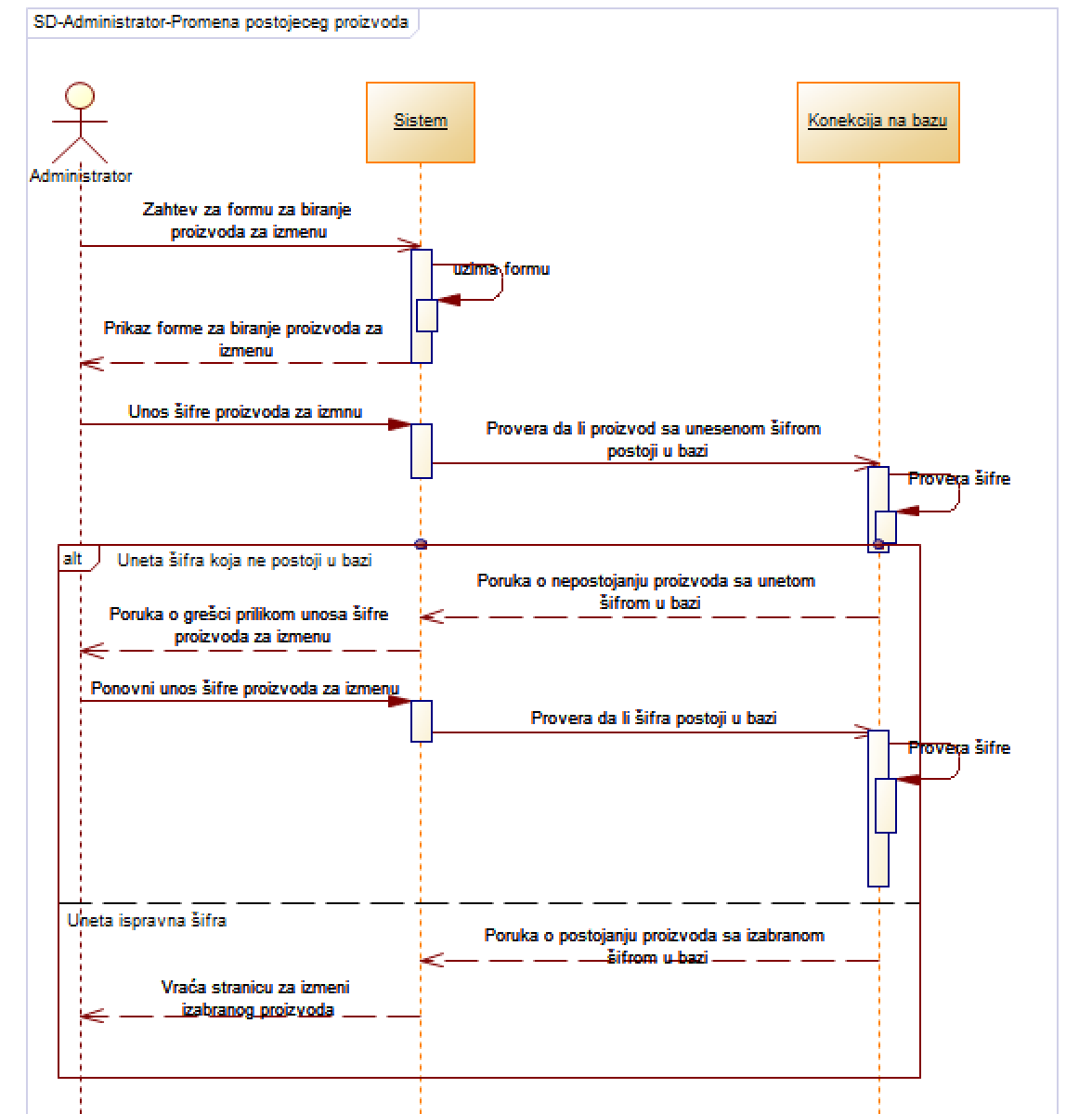
Slika 8.2.3: SSD-Dodavanje proizvoda III



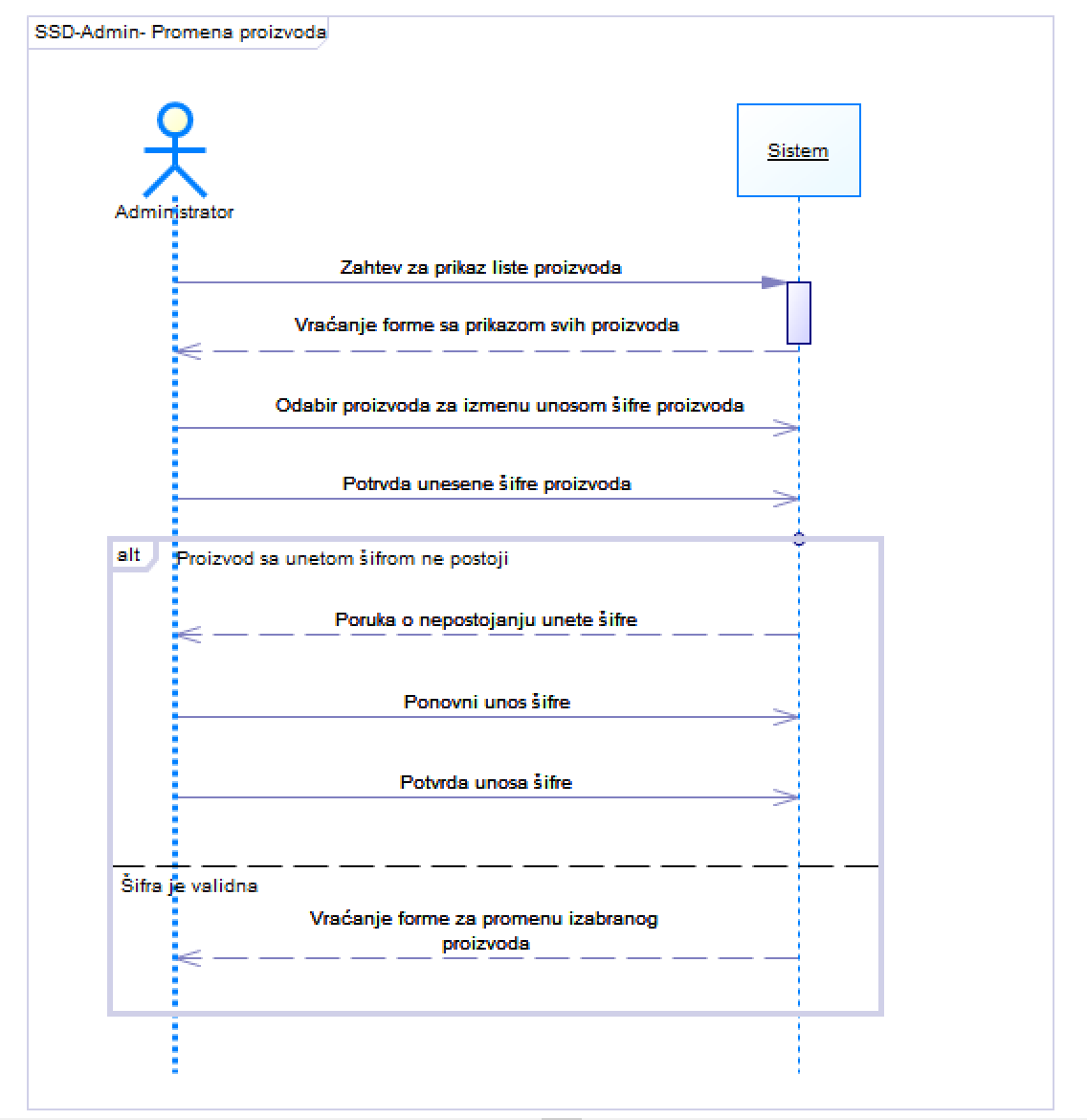
Slika 8.2.4: SD-Dodavanje proizvoda

U okviru dijagrama za dodavanje proizvoda, detaljno su predstavljeni svi uslovi koje je potrebno ispuniti kako bi bilo omogućeno unošenje proizvoda. Takođe su prikazani i svi alternativni scenariji koji se tiču ovog slučaja korišćenja. U sistem sekvencijalnom dijagramu vidimo poruke koje sistem razmenjuje sa sistemom, dok u sekvencijalnom postoji prikaz rada u samom sistemu kao i razmene poruka među objektima.

***Promena proizvoda***

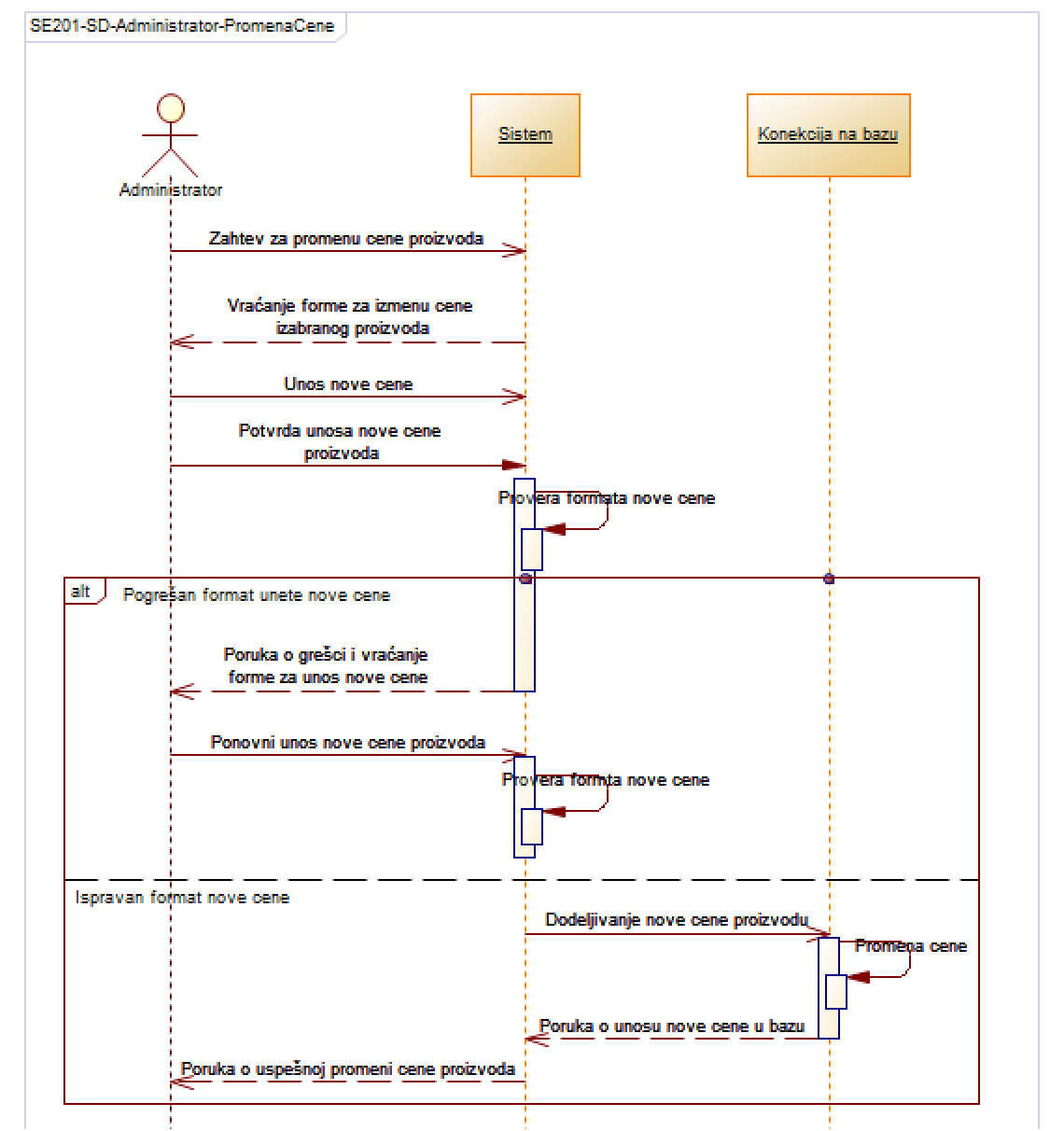


Slika 8.3.1: SD dijagram za izmenu proizvoda

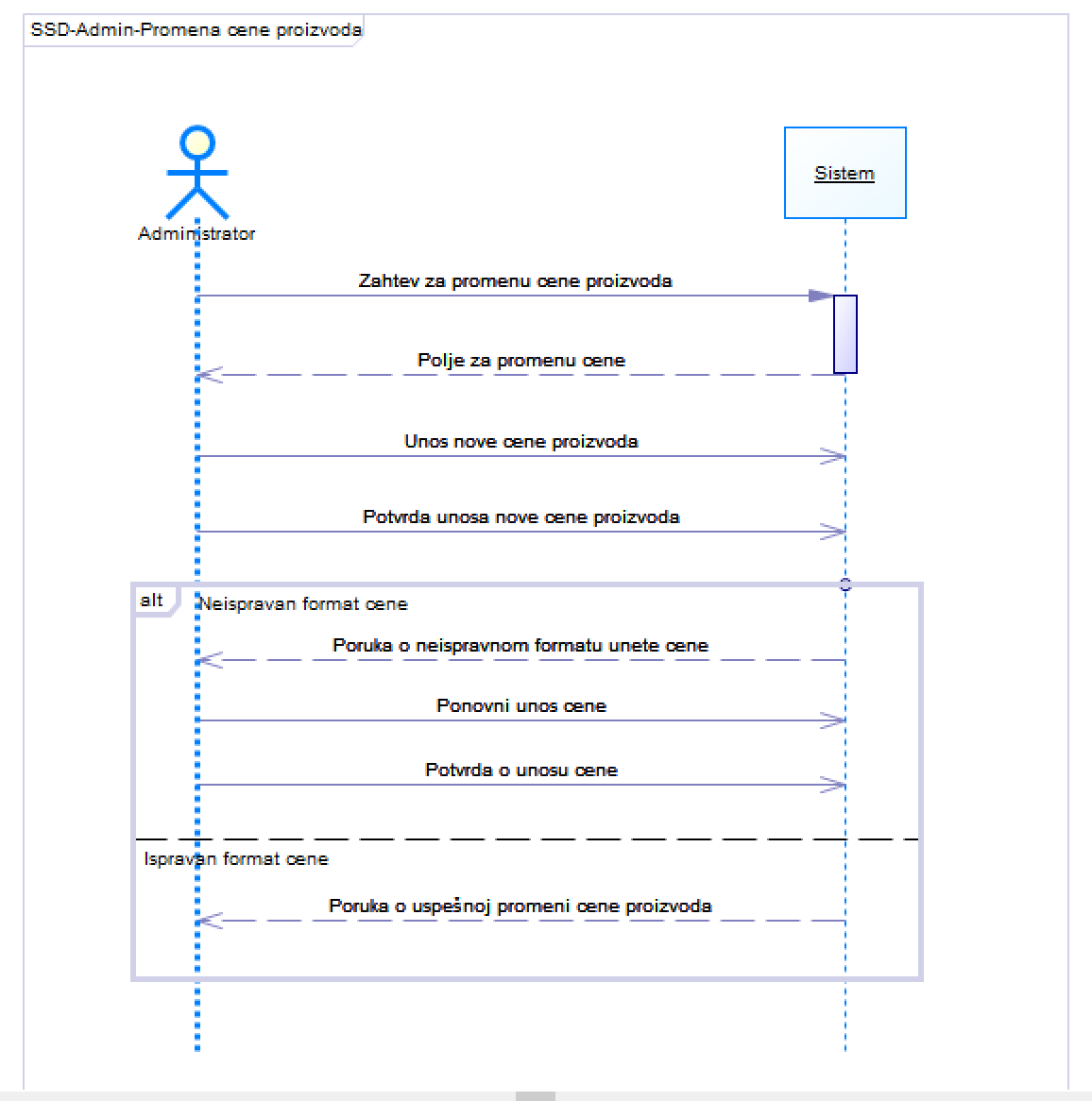


Slika 8.3.2: SSD dijagram za izmenu proizvoda

***Promena cene proizvoda***



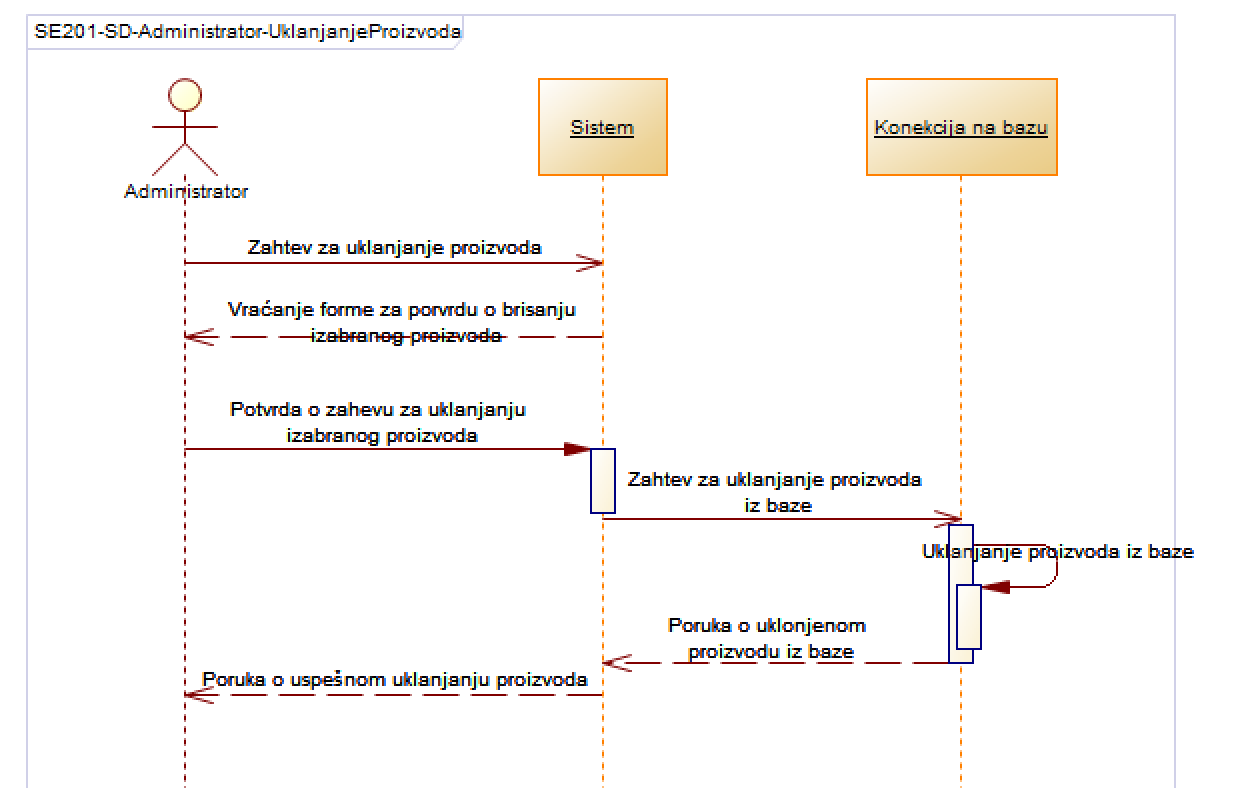
Slika 8.4.1: SD za izmenu cene proizvoda



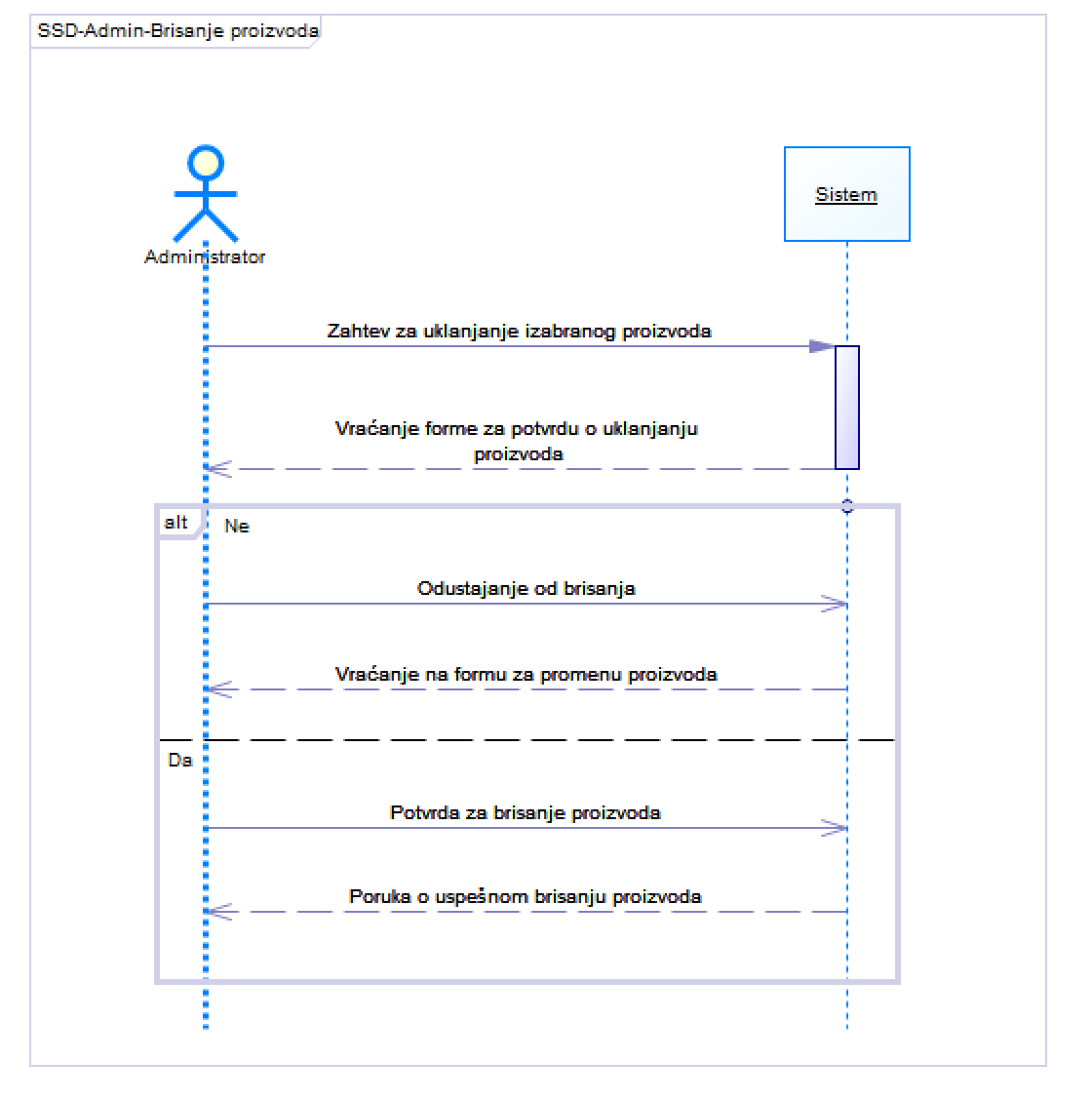
Slika 8.4.2: SSD za izmenu cene proizvoda

SD i SSD dijagrami koji opisuju sistem pri izmeni cene proizvoda pokazuju kako sistem razmenjuje poruke sa korisnikom i u kojim situacijama postoje alternativni scenariji. Sistem treba da pri svakom unosu cene proveri da li je cena validnog formata, što je relevantno u ovakvom sistemu. Na kraju, ukoliko je sve pravilno urađeno, sistem ubacuje novu cenu u bazu i prikazuje je uz proizvod.

***Uklanjanje proizvoda***

******

Slika 8.5.1: SD za uklanjanje proizvoda

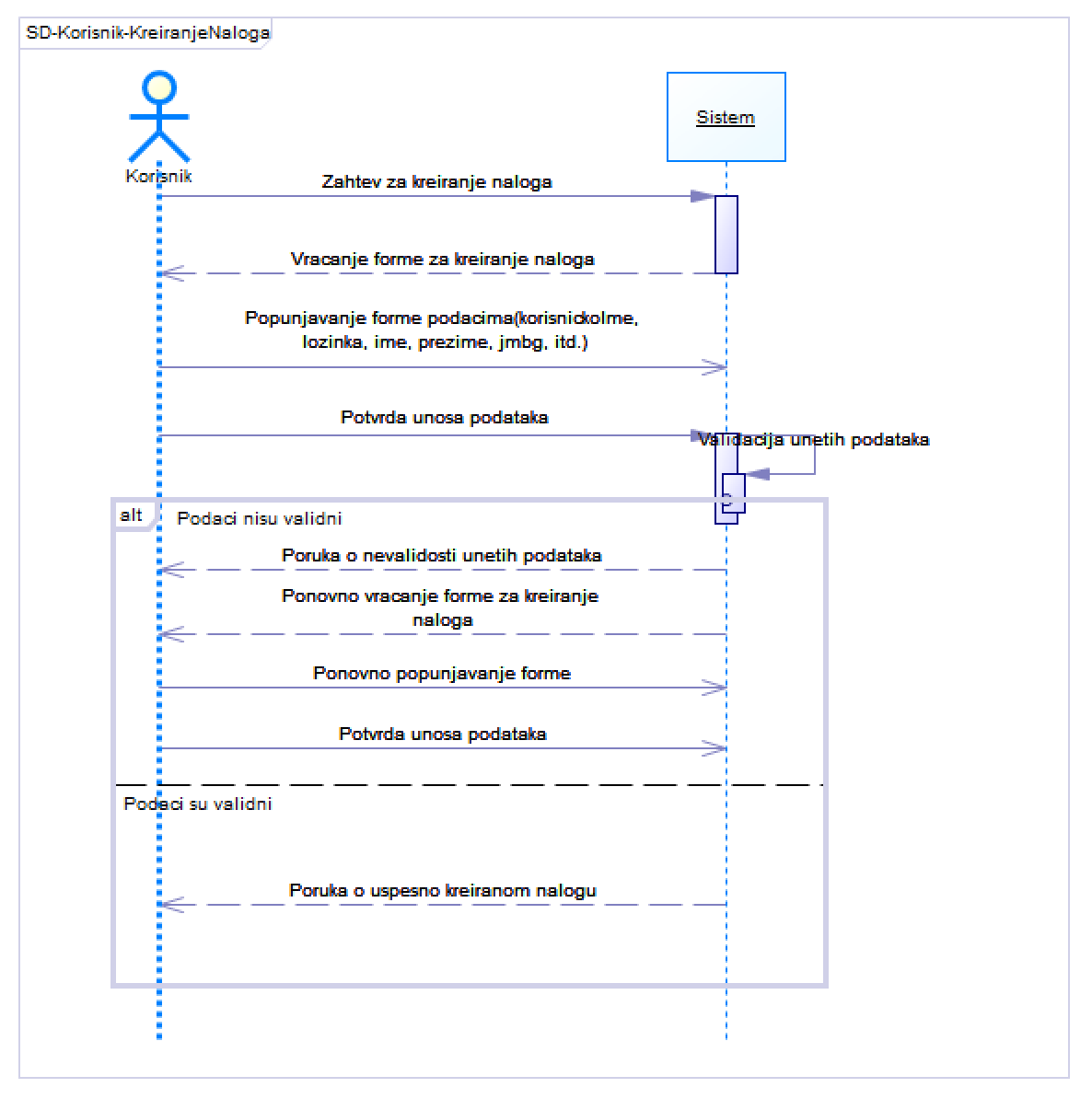


Slika 8.5.2: SSD za uklanjanje proizvoda

Kako bi administrator uklonio proizvod, sistem treba da obavesti korisnika porukom da je izabrao datu opciju i da još jednom proveri da li je siguran u svoj odabir. Ukoliko administrator odgovori potvrdno, sistem uklanja proizvod i iz baze i sa sajta. U suprotnom, sistem vraća administratora na stranicu za izmenu proizvoda.

#### Korisnik

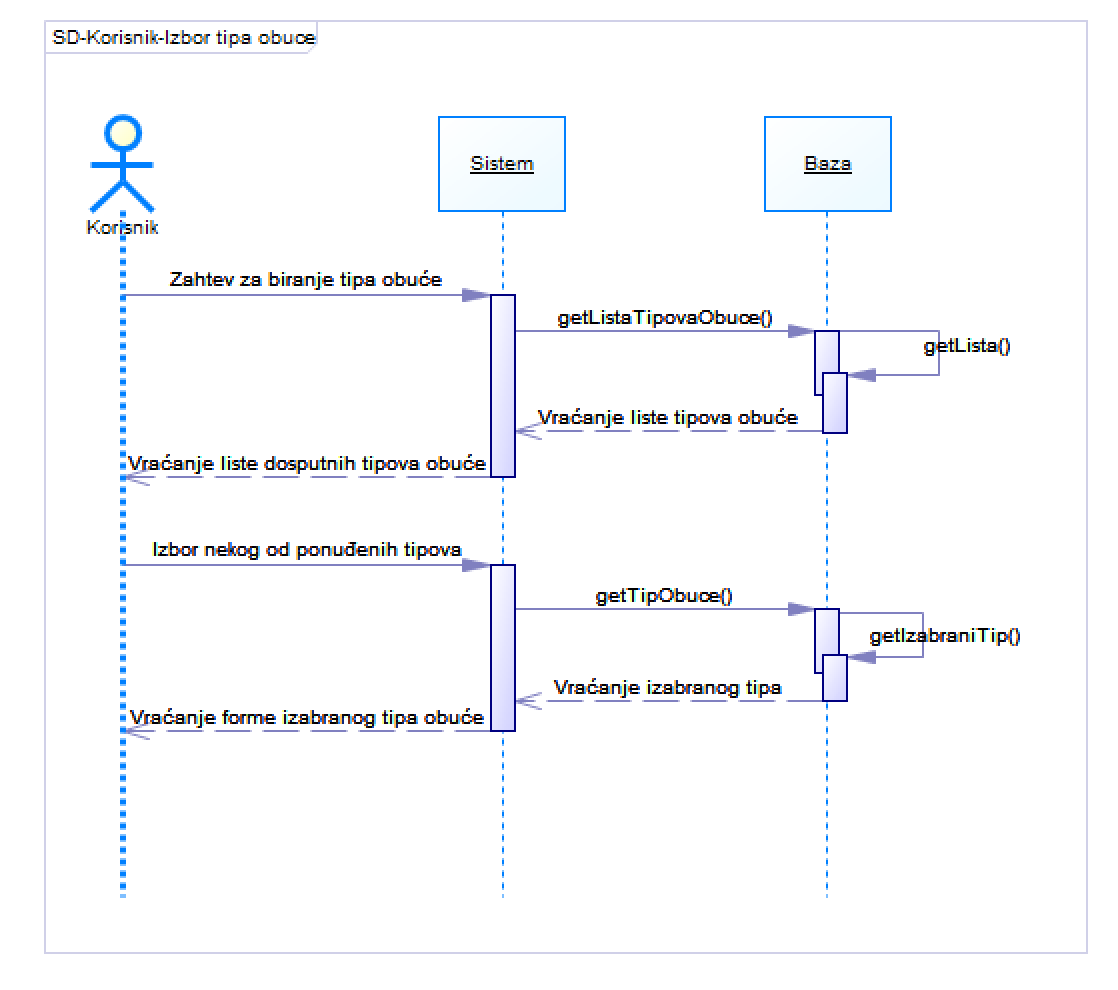
#### Kreiranje naloga



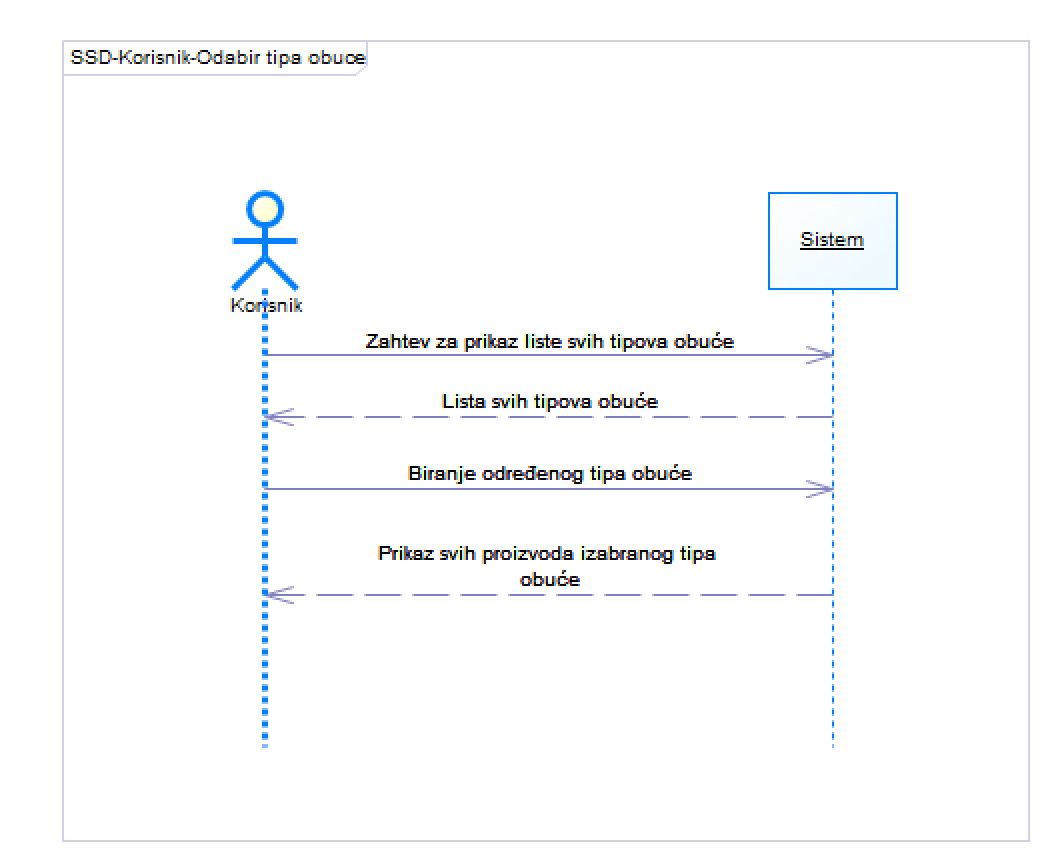
Slika 8.6.1: SD za kreiranje naloga korisnika

Kada korisnik kreira nalog, sistem treba da proveri da li je korisnik uneo podatke validnog formata. Ukoliko se to ispunjeno, sistem pravi nalog korsinku. U surotnom, vraća mu formu za ponovno popunjavanje forme podacima.

***Biranje tipa obuće***

******

Slika 8.7.1: SD za biranje tipa obuće

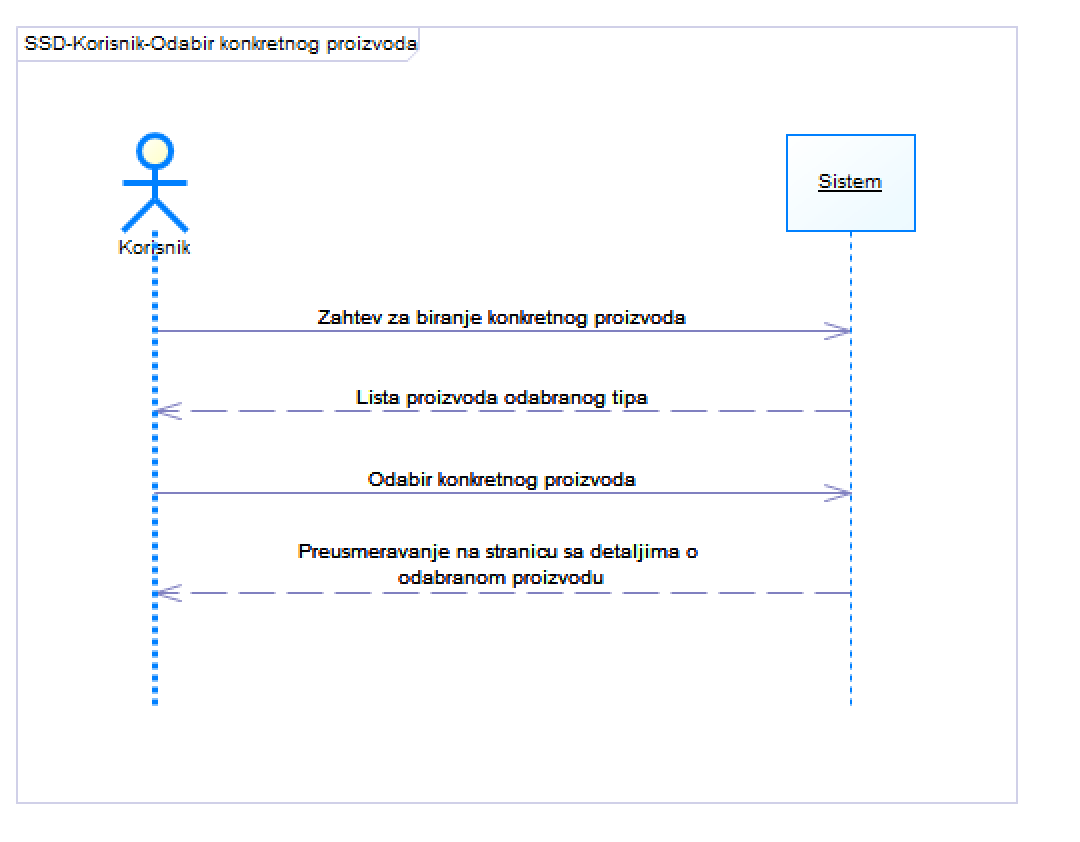


Slika 8.7.2: SSD za biranje tipa obuće

***Biranje konkretnog proizvoda***

******

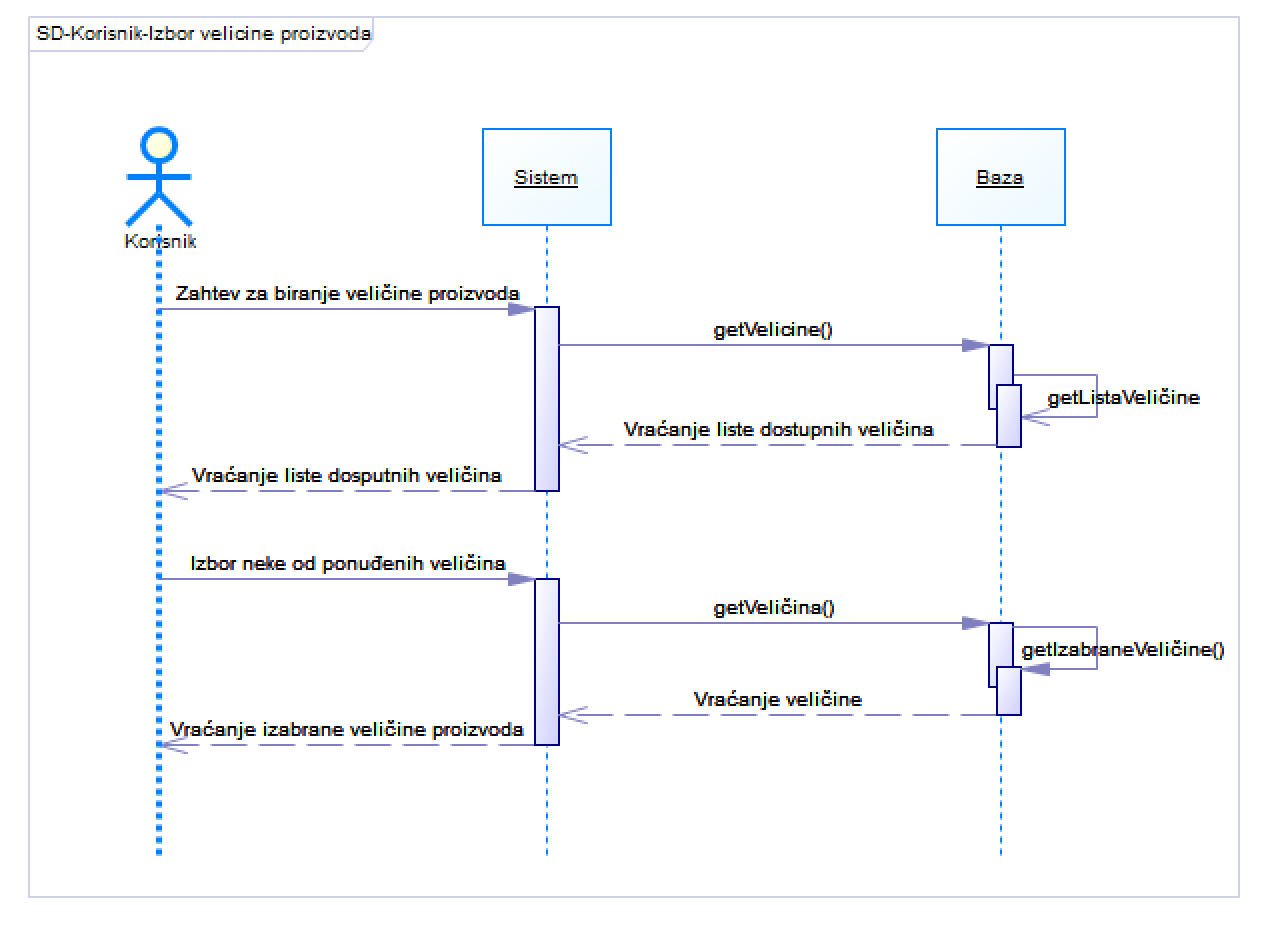
Slika 8.8.1: SD za biranje konkretnog proizvoda



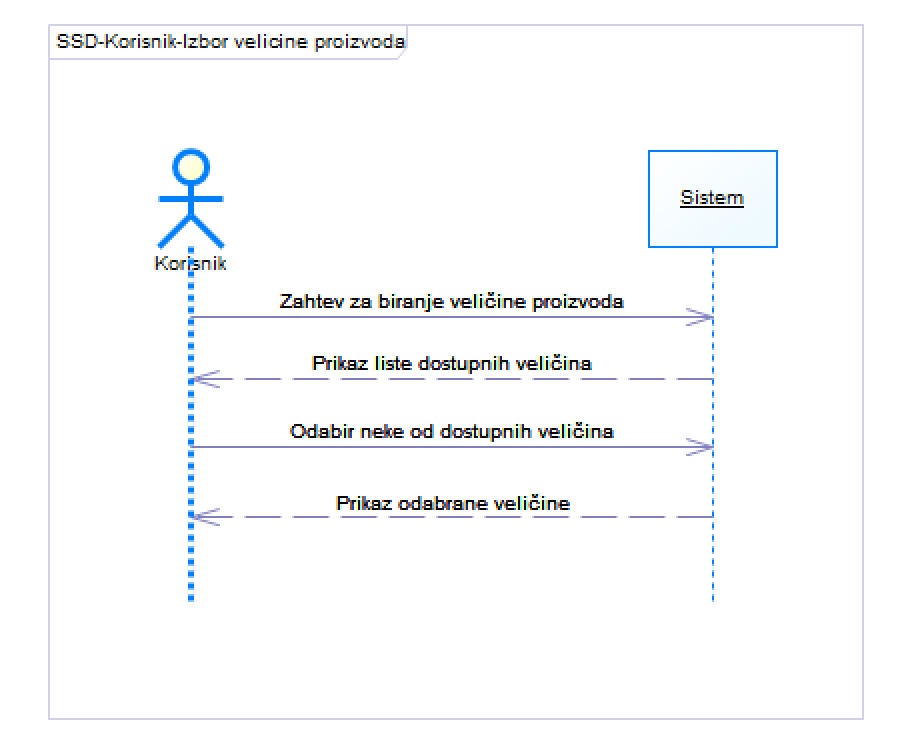
Slika 8.8.2: SSD za biranje konkretnog proizvoda

Pri izboru konkretnog proizvoda, korisnik prvo bira tip obuće. Nakon odabranog tipa, sledi biranje konkretnog komada koji pripada toj vrsti i kolekciji. U jednom trenutku, korisnik može da pritupi samo jednom komadu.

***Biranje veličine proizvoda***



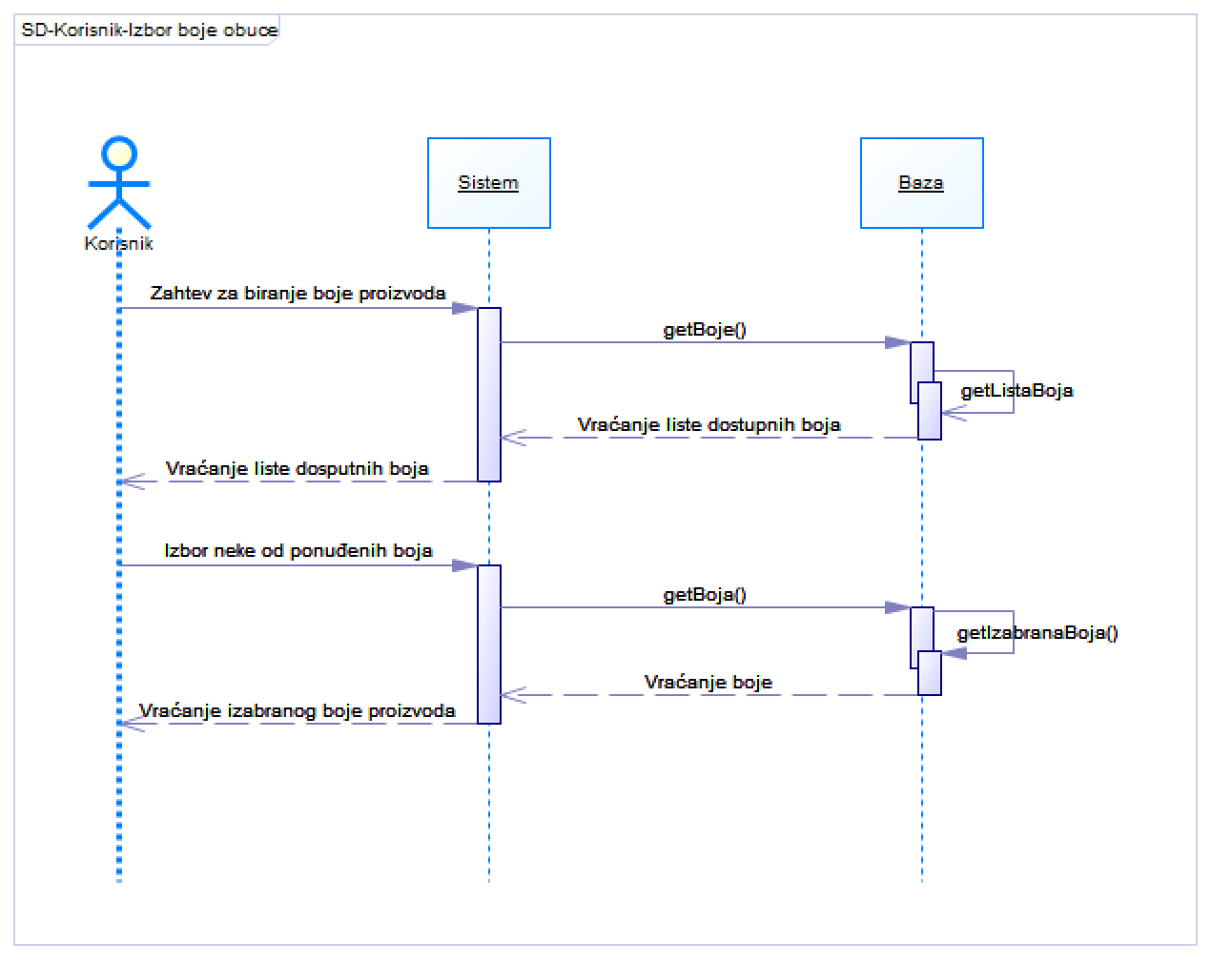
Slika 8.9.1: SD za biranje veličine proizvoda



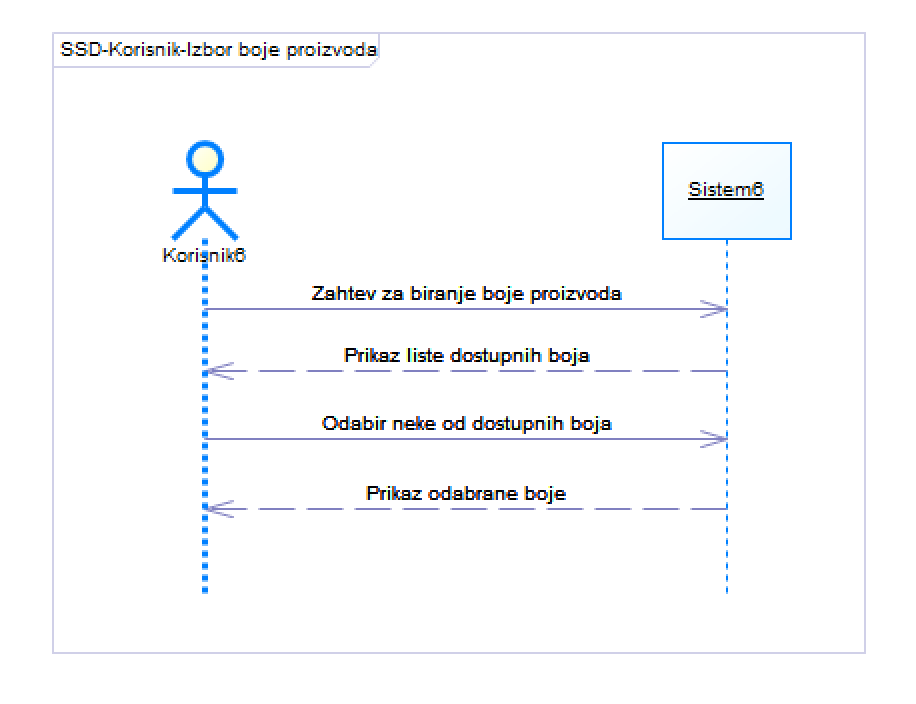
Slika 8.9.2: SSD za biranje veličine proizvoda

Kada je izabrao konkretan proizvod, korisniku ostaje da odabere veličinu i boju odevnog komada koji želi da kupi. Upravo taj proces je prikazan na slikama iznas. Tu ne postoje alternativni scenariji jer je korisniku ponuđena lista.

***Biranje boje proizvoda***

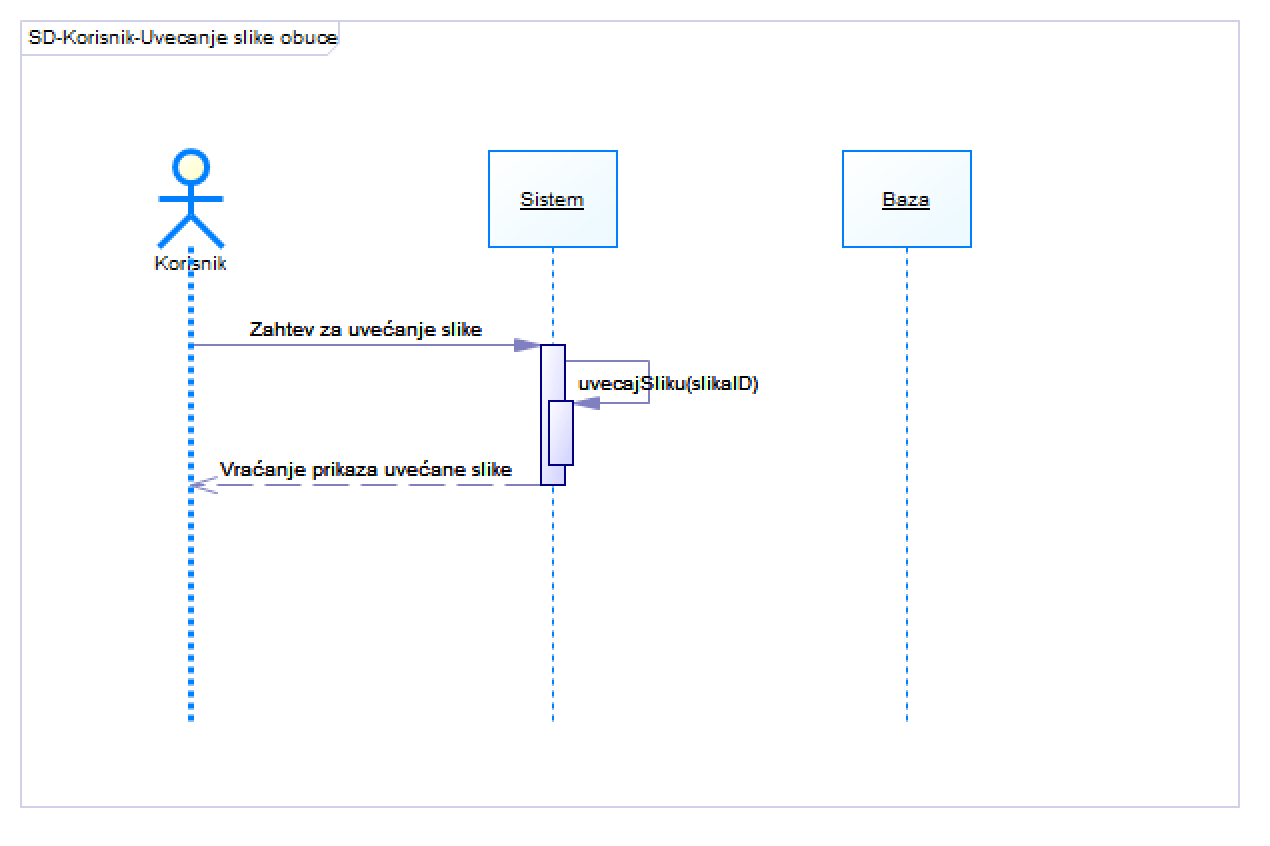


Slika 8.10.1: SD za biranje boje proizvoda



Slika 8.10.2: SSD za biranje boje proizvoda

***Uvećanje slike proizvoda***

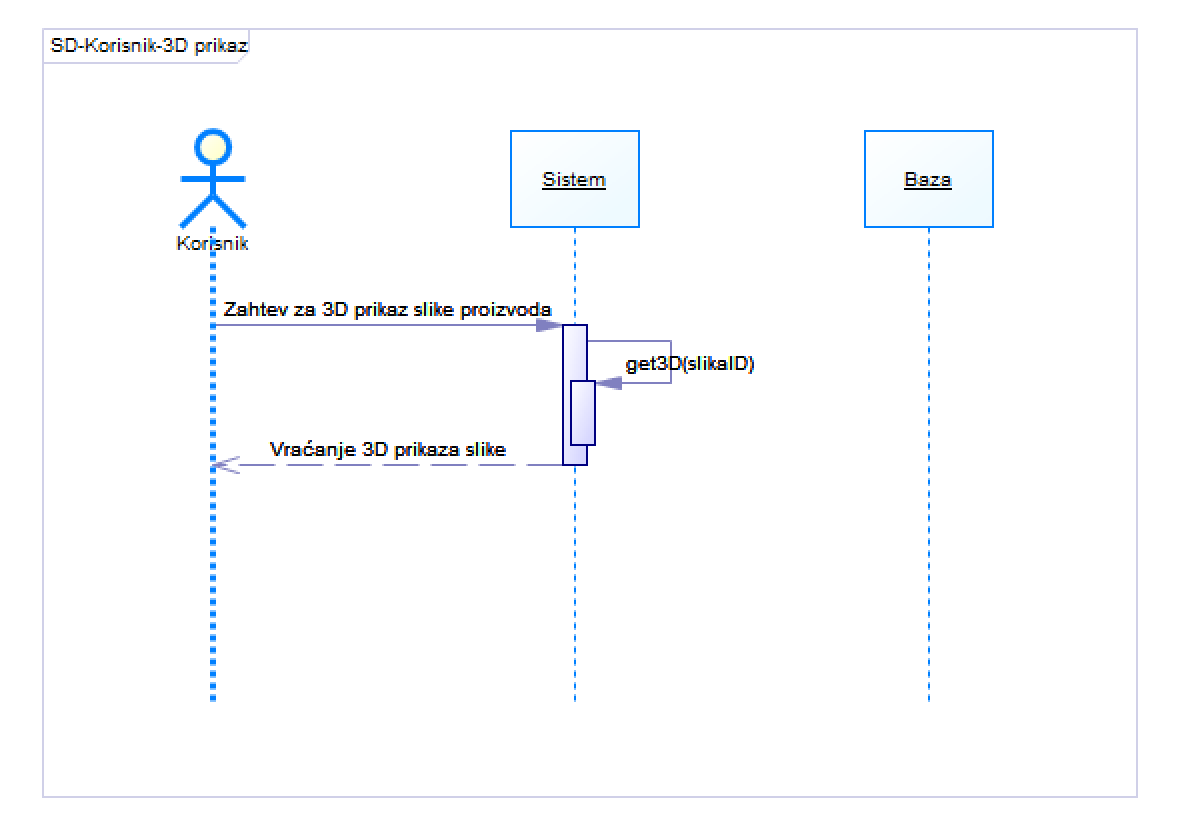
******

Slika 8.11.1: SD za uvećanje slike proizvoda

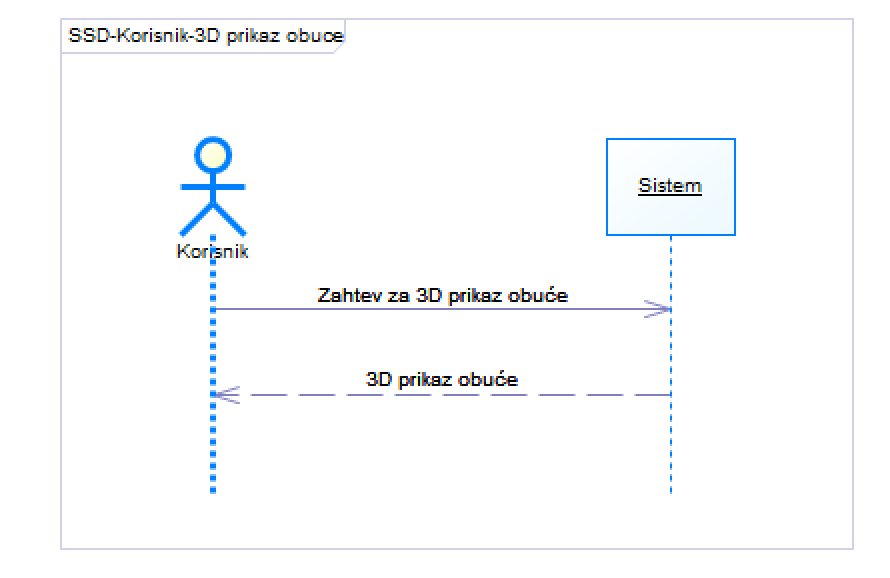


Slika 8.11.2: SSD za uvećanje slike proizvoda

***3D prikazivanje proizvoda***

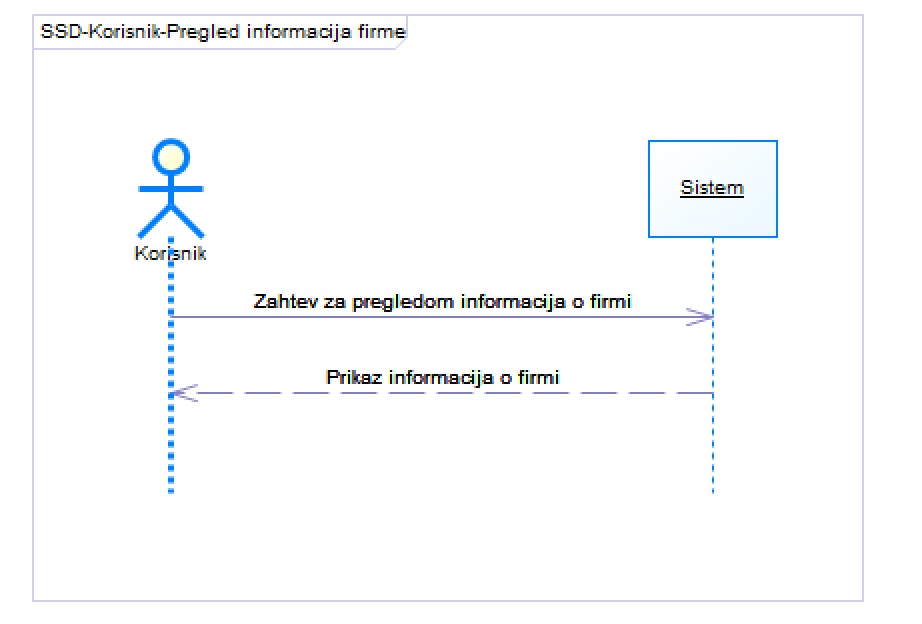
******

Slika 8.12.1: SD za prikazivanje 3D prikaza proizvoda

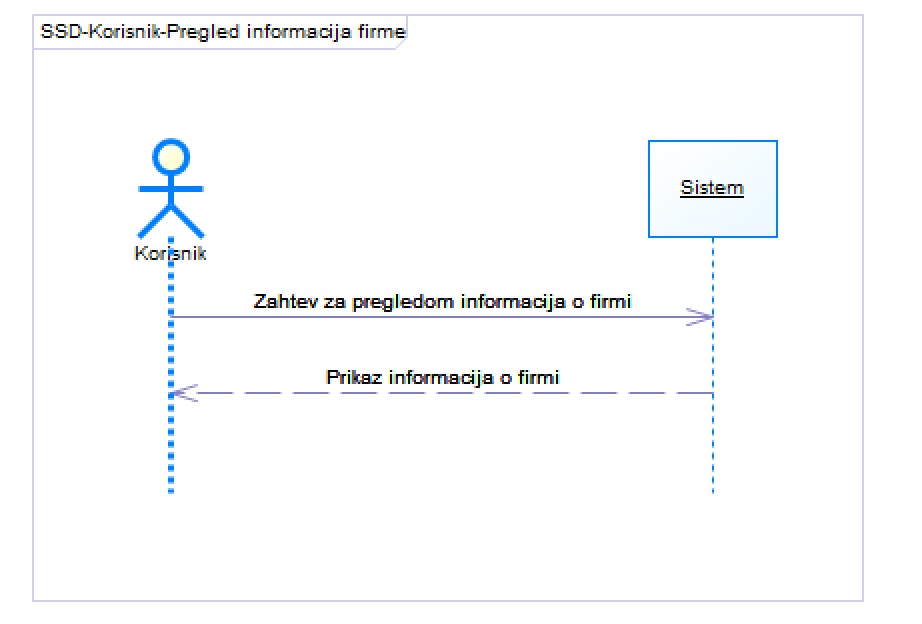


Slika 8.12.2: SSD za prikazivanje 3D prikaza proizvoda

***Pregled informacija o kompaniji***



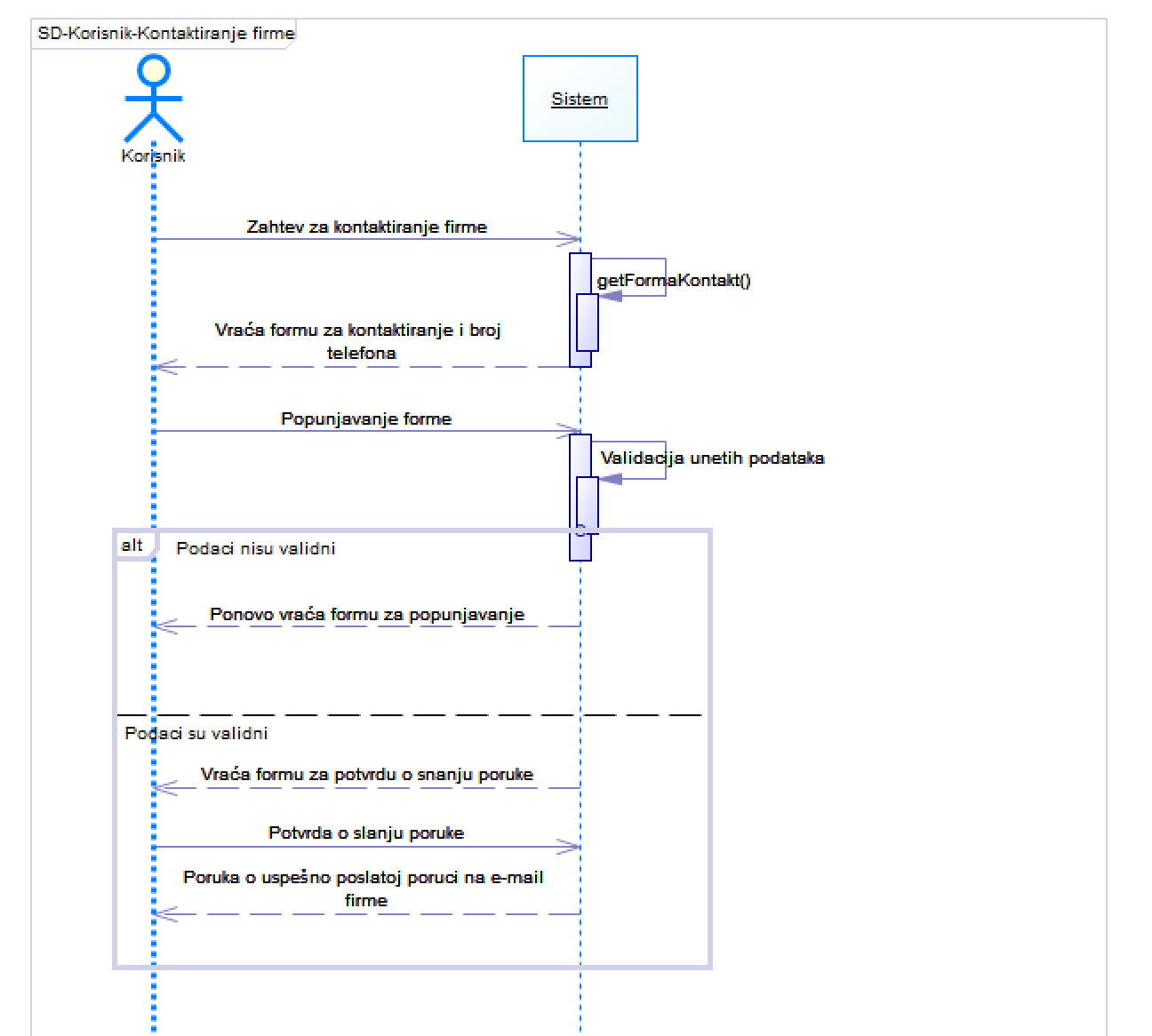
Slika 8.13.1: SD za prikazivanje informacija o firmi



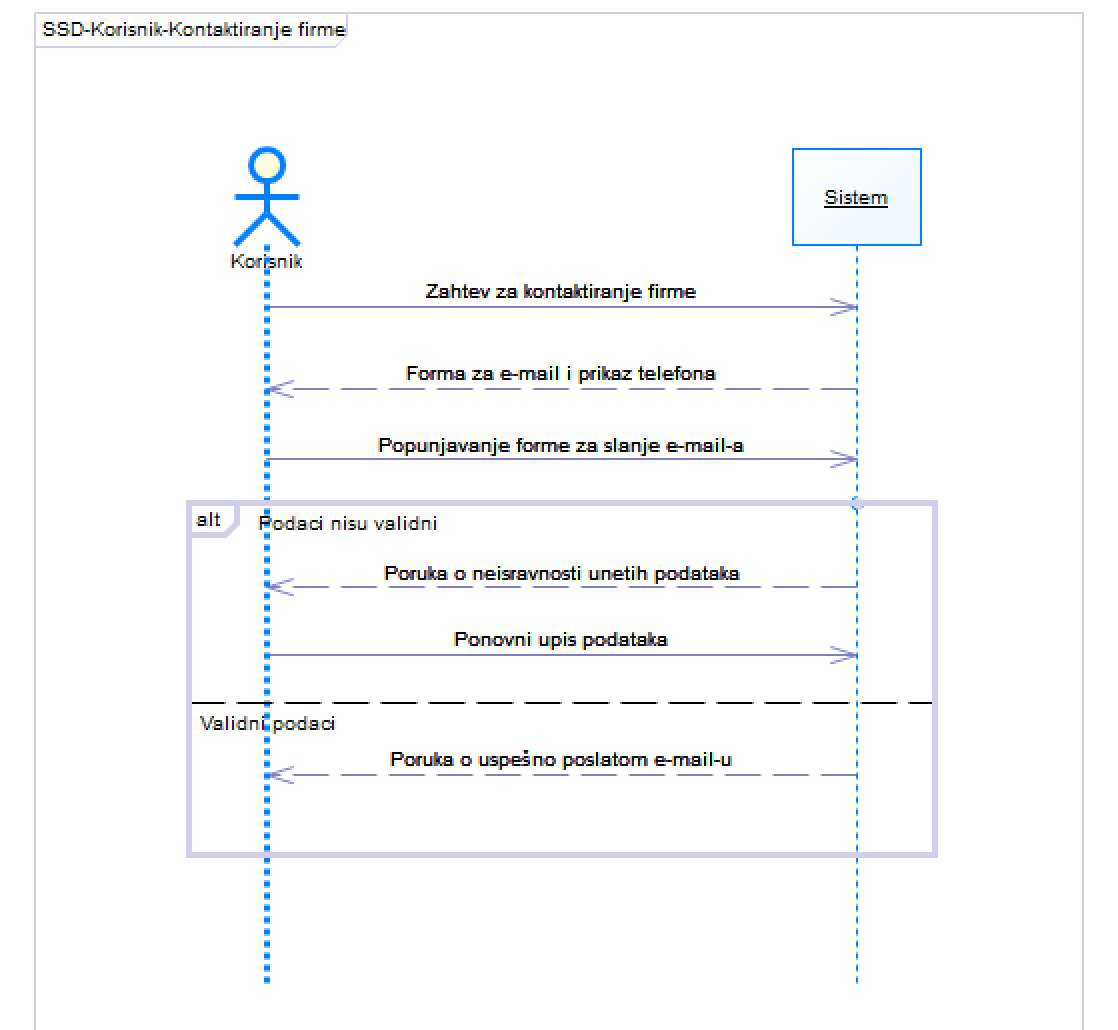
Slika 8.13.2: SSD za prikazivanje informacija o firmi

Nakon što se korisnik uloguje, dobija opciju da pregleda informacije o kompaniji. Pritom se podrazumeva da vidi na kojim lokacijama se kompanija nalazi i tačnu adresu. Ovi podaci se čitaju iz baze i prikazuju korisnicima.

***Kontaktiranje firme***

******

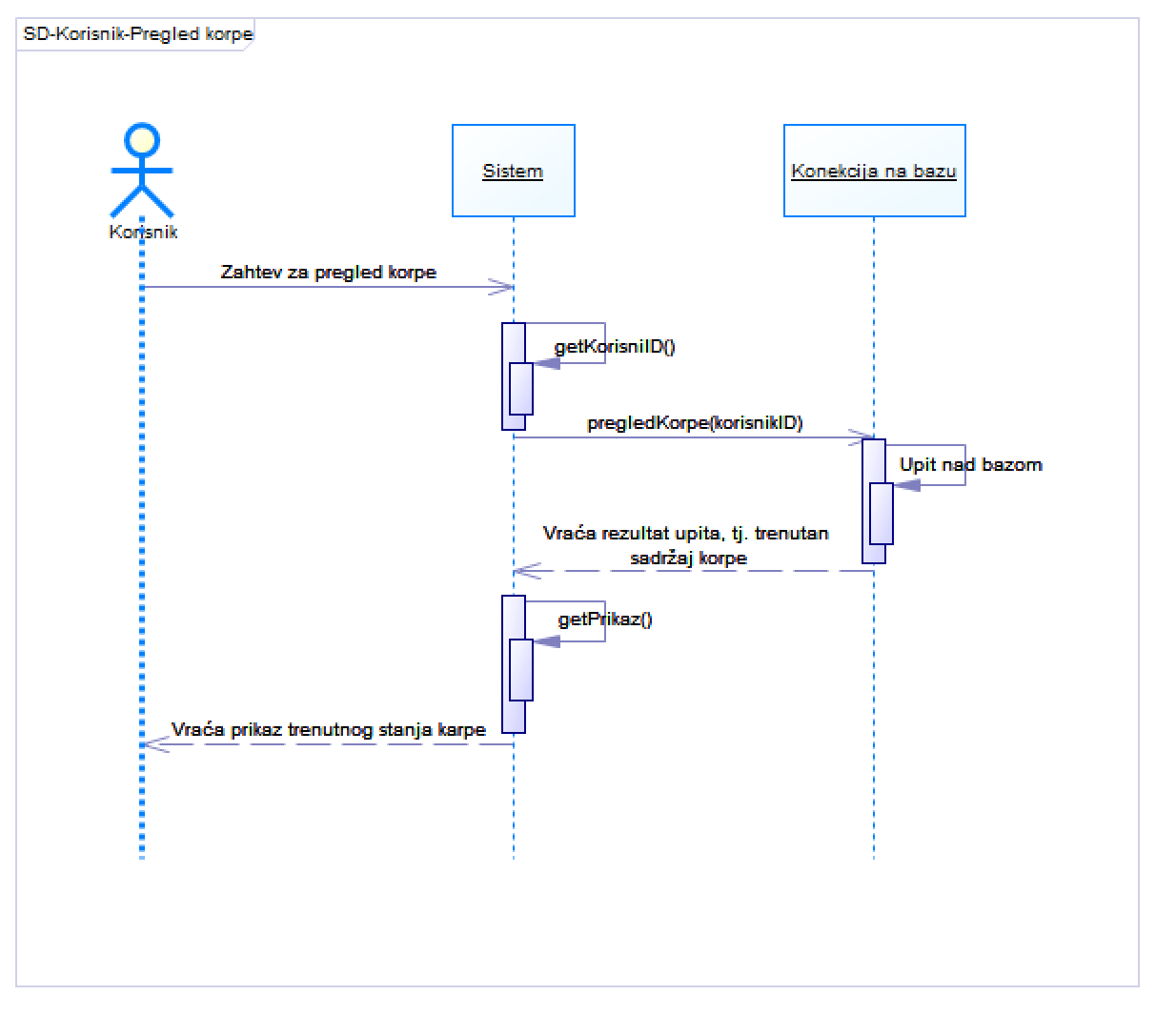
Slika 8.14.1: SD za Kontaktiranje firme



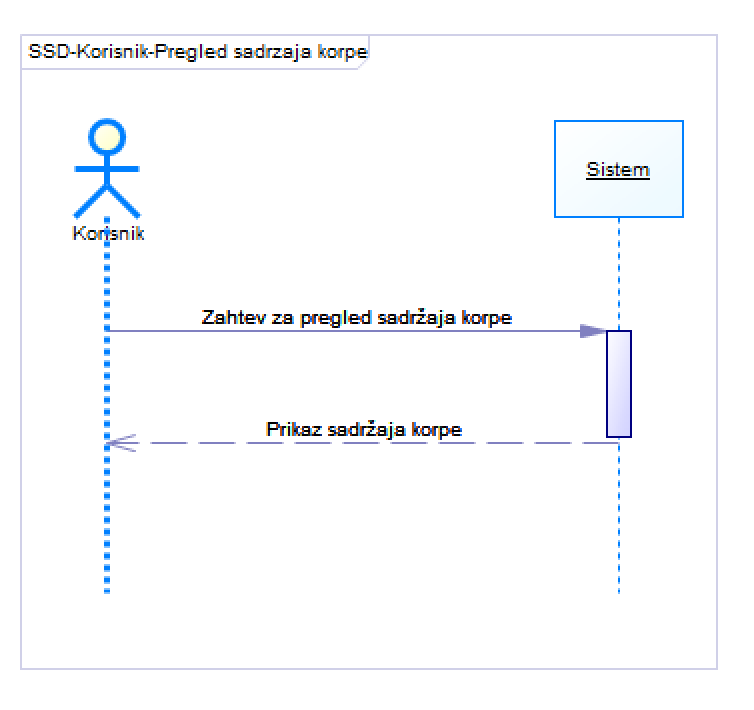
Slika 8.14.2: SSD za Kontaktiranje firme

Ako korisnik želi da kontaktira firmu, mora da popuni priloženu formu. Pri svakom unosu podatka, sistem proverava validnost podataka. Na kraju, ukoliko je sve ispravno, poruka se upisuje u bazu kako bi ostala sačuvana za evidenciju.

***Pregled sadržaja korpe***



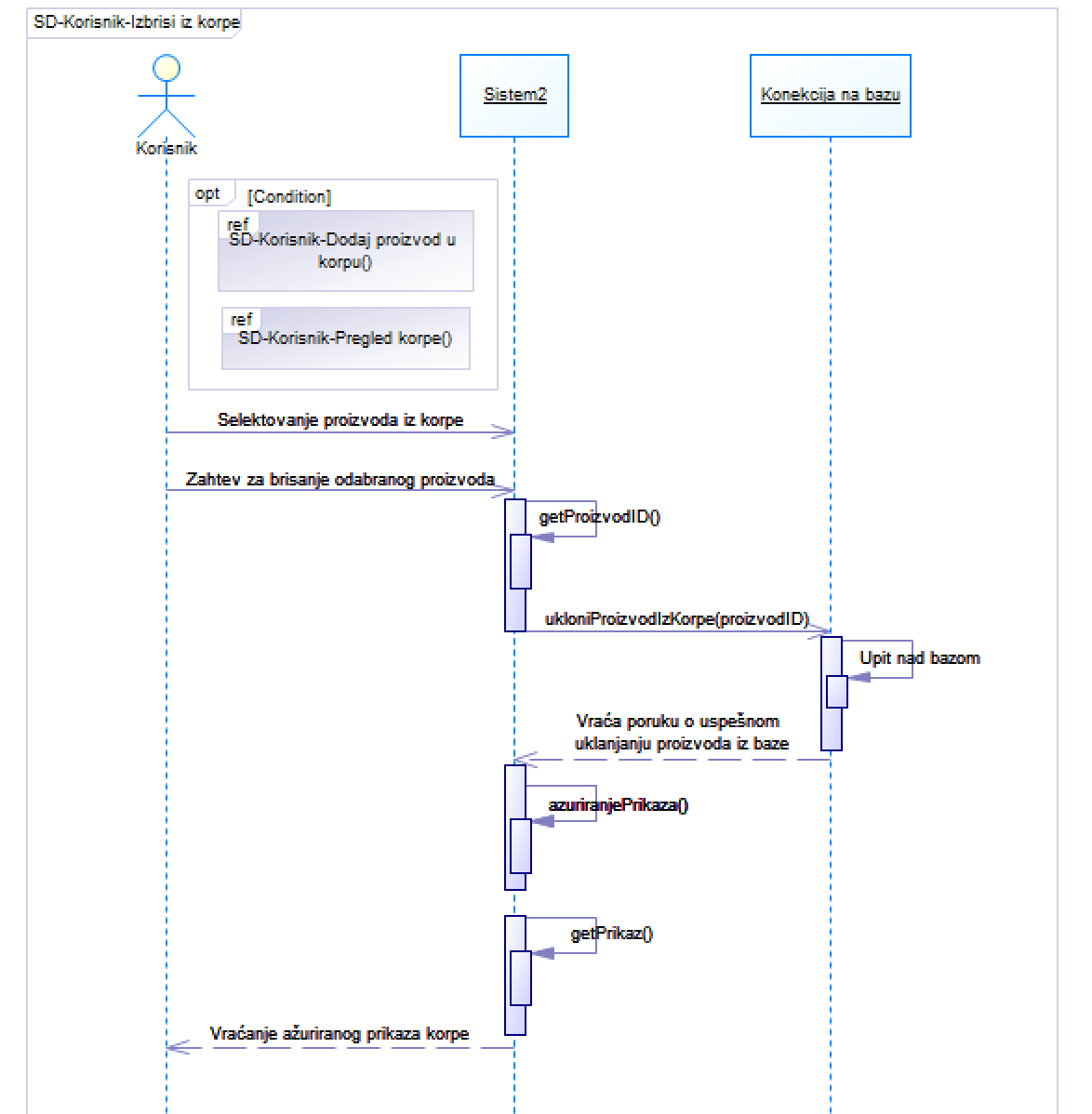
Slika 8.15.1: SD za pregled trenutnog sadržaja korpe



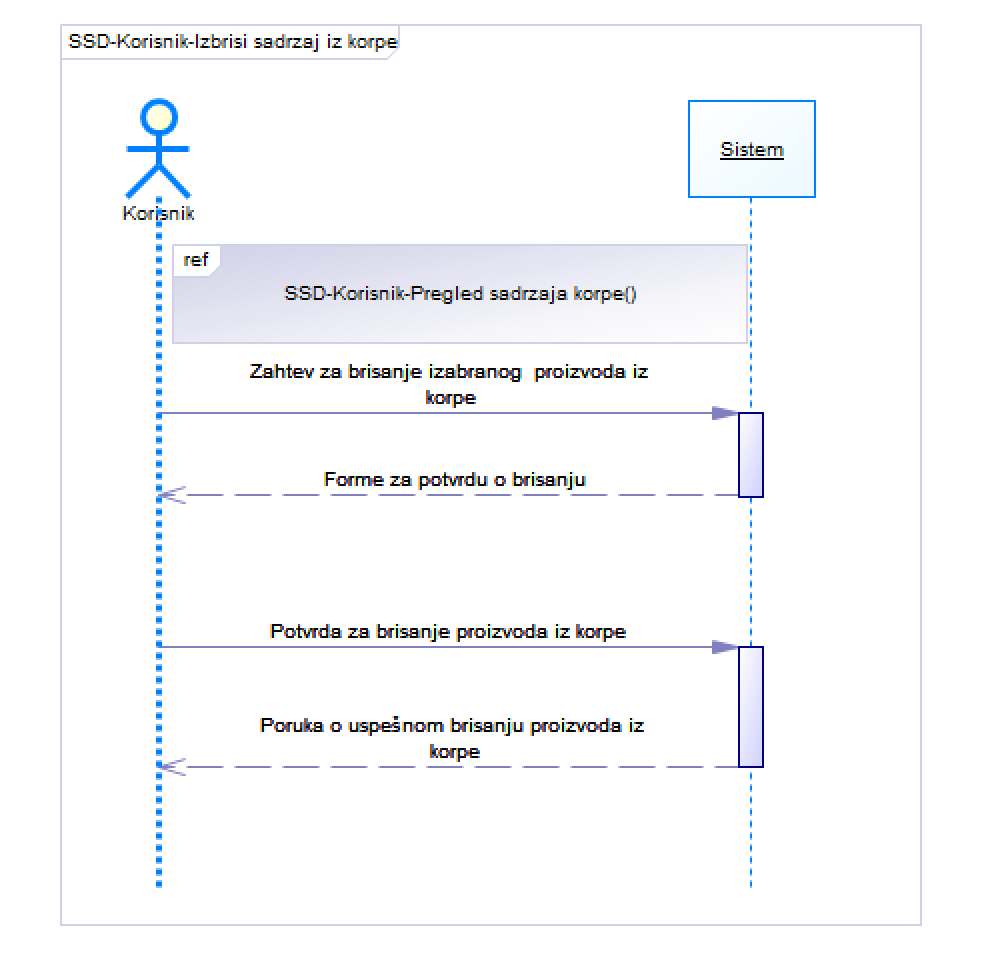
Slika 8.15.2: SSD za pregled trenutnog sadržaja korpe

Jedna od mogućnosti korisnika je da pregleda korpu. Naravno, preduslov za to je da ima nešto u korpi već. Ukoliko ne postoiji nista u korpi, korisnik dobija poruku da je korpa prazna. U suprotnom, korisnik dobija sadržaj korpe na uvid.

***Ukloni proizvod iz korpe***



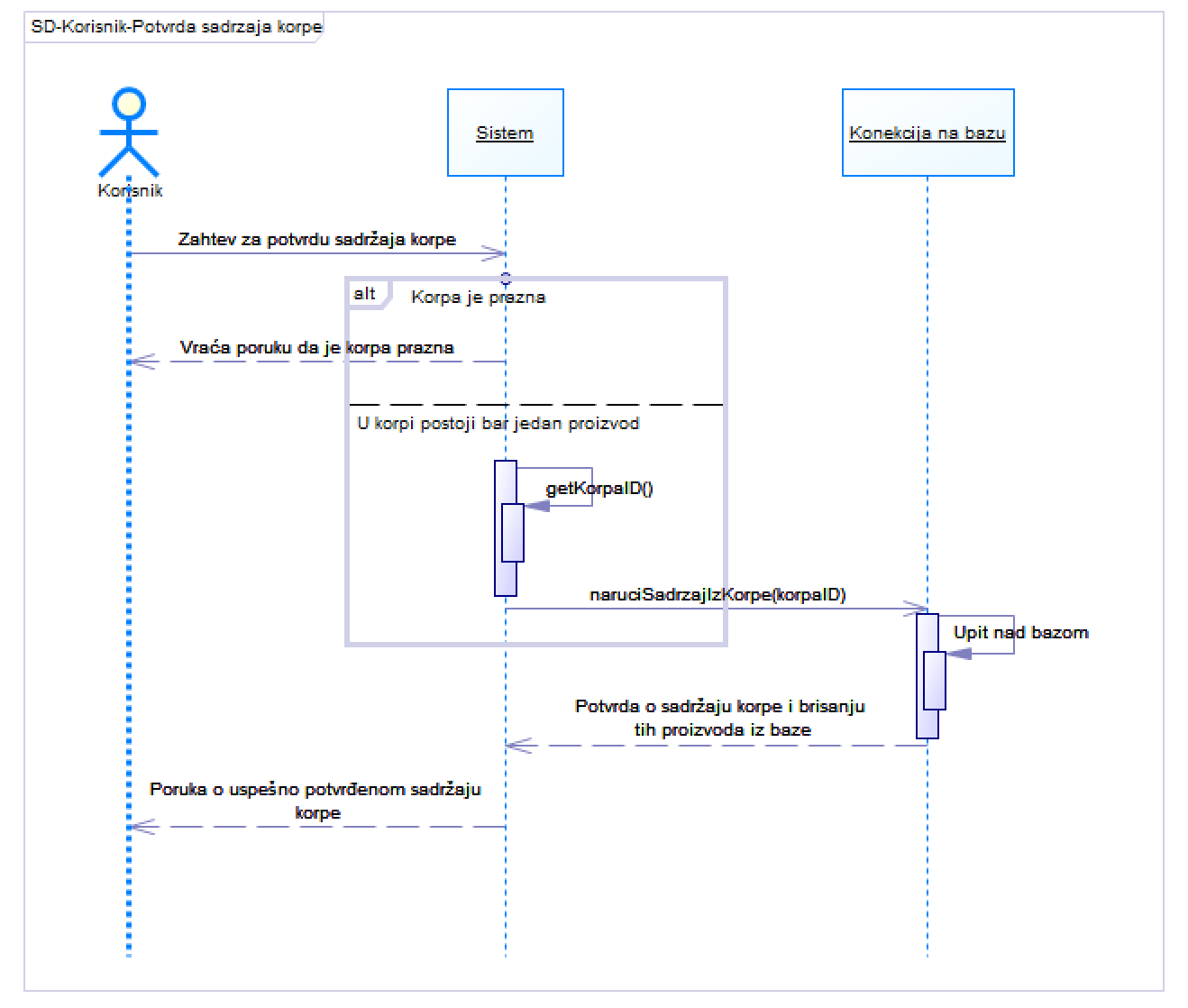
Slika 8.16.1: SD za uklanjanje proizvoda iz korpe



Slika 8.16.2: SSD za uklanjanje proizvoda iz korpe

Korisnik ima mogućnost ako ima nalog da briše željeni proizvod iz korpe. Ovde vidimo način na koji sistem komunicira sa bazom u tom trenutku. Da bi se obrisao proizvod, naravno, korpa mora da sadrži taj proizvod i ukoliko je korisnik siguran da to želi da uradi, u bazi se menja sadržaj korpe i prenosi na ekran.

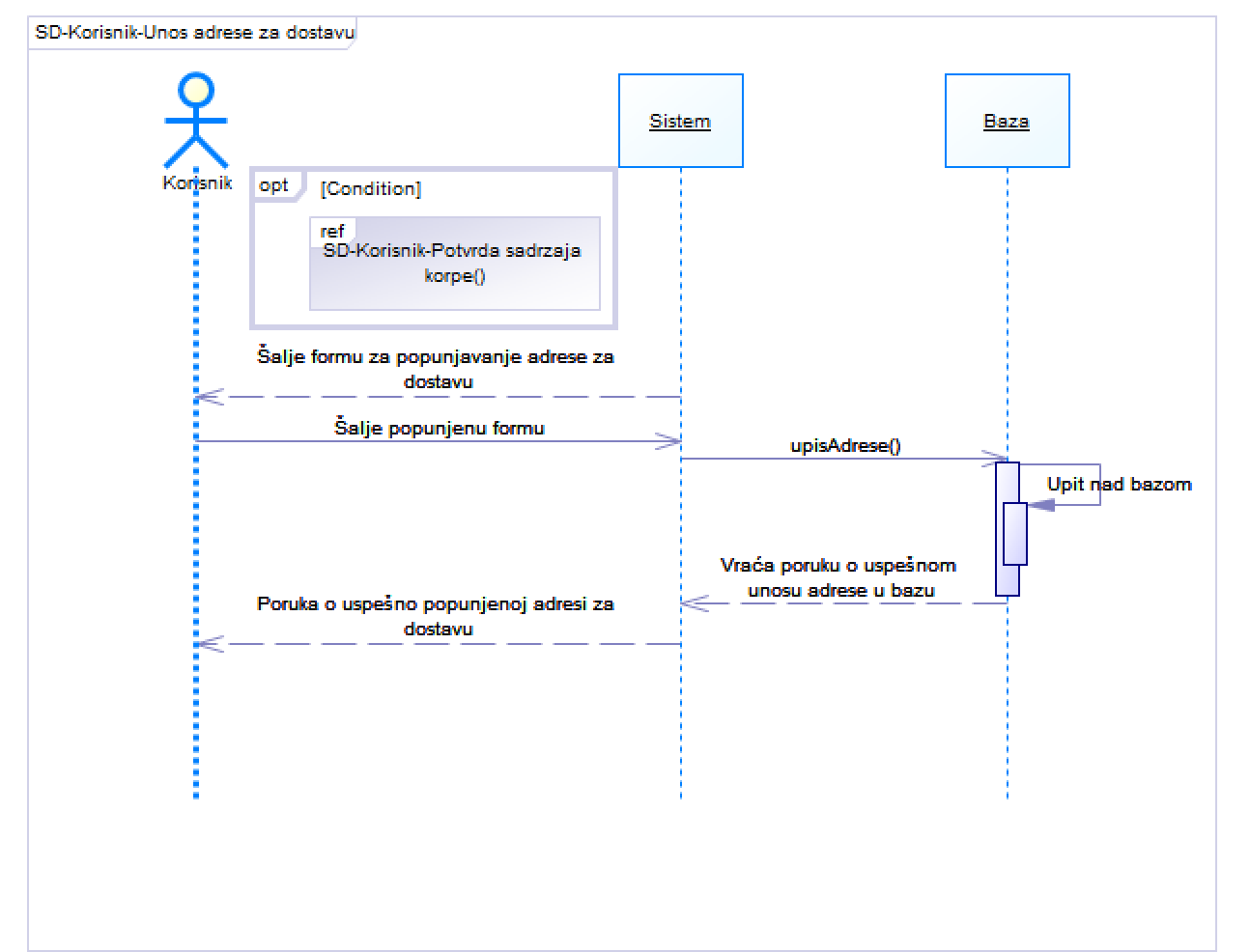
***Potvrđivanje sadržaja korpe***

******

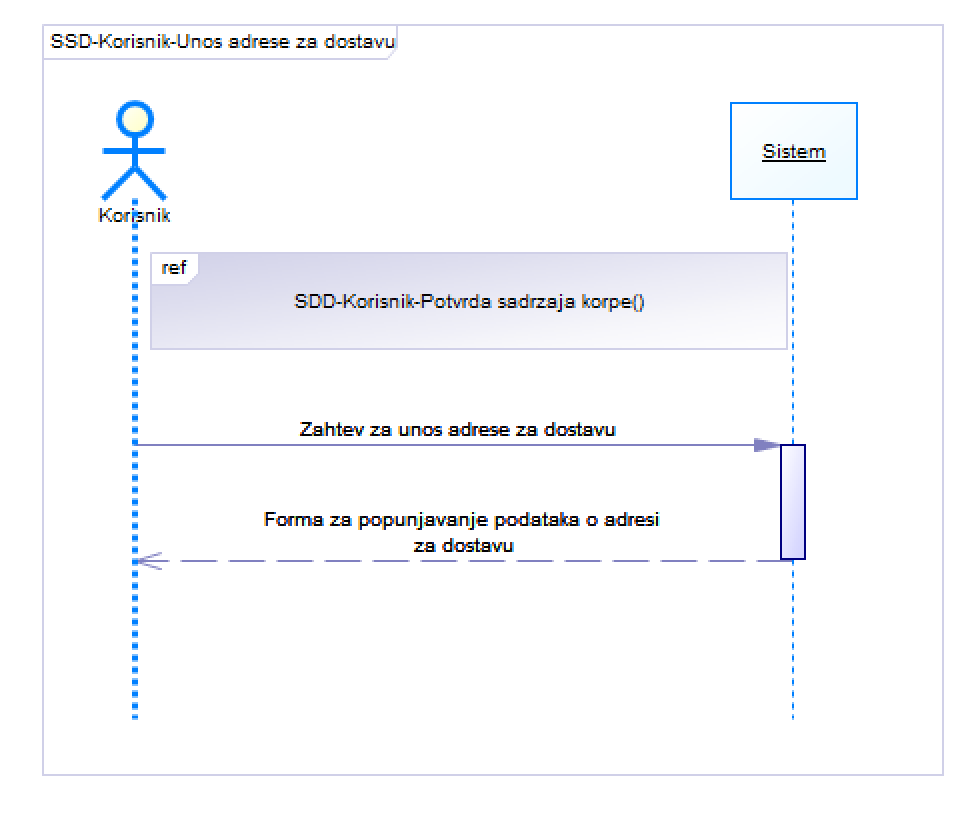
Slika 8.17.1: SD za potvrdu sadržaja iz korpe

Neophodno je da korisnik pregleda korpu i da pre svega ima nešto u korpi kako bi mogao da potvrdi njen sadržaj. Bez tog uslova, ovaj slučaj korišćenja je onemogućen. Takođe je bitno sadržaj korpe upisti u bazu kako bi se čuvali podaci za evidenciju.

***Unos adrese za dostavu***

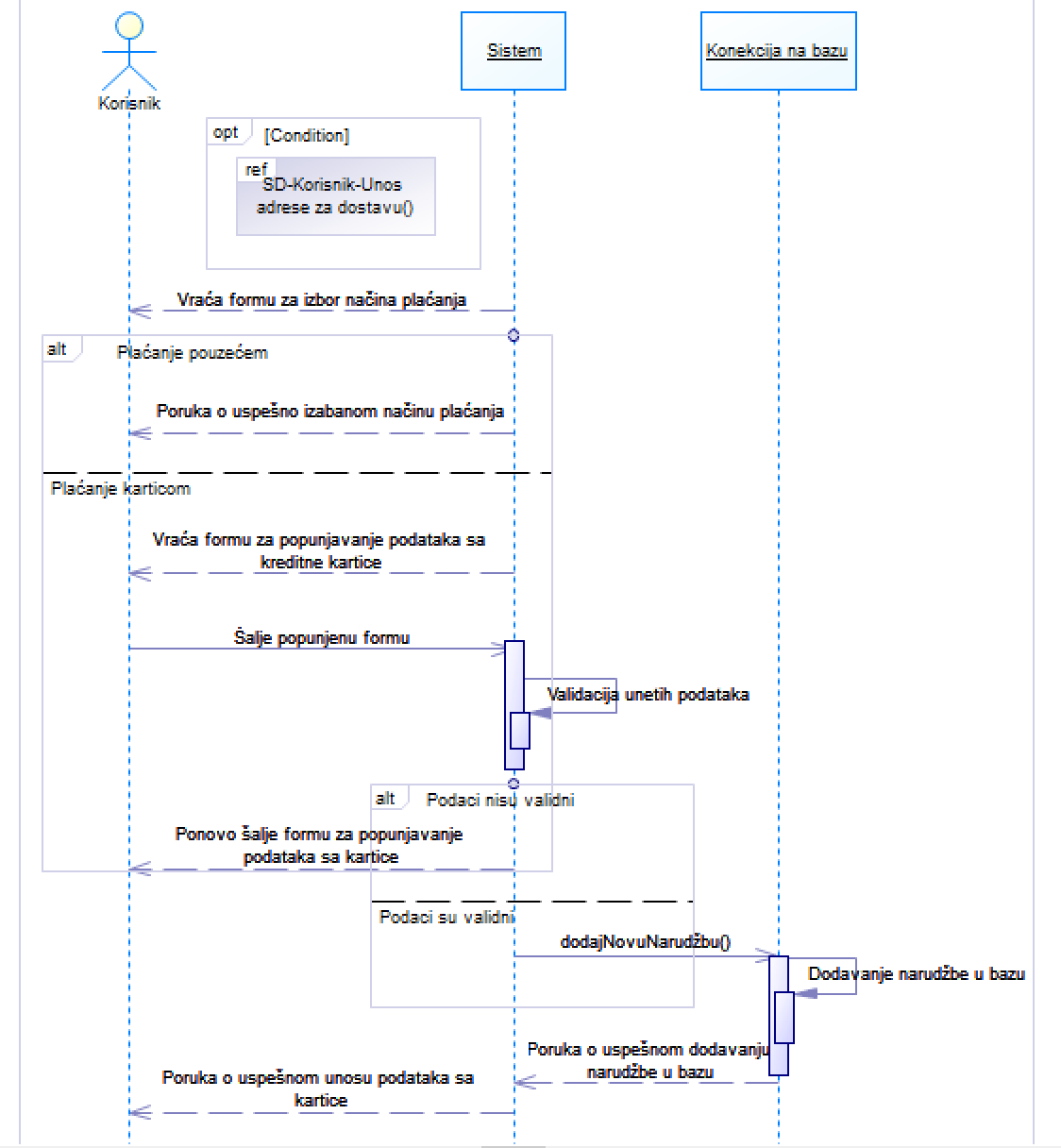
******

Slika 8.18.1: SD za unos adrese za dostavu

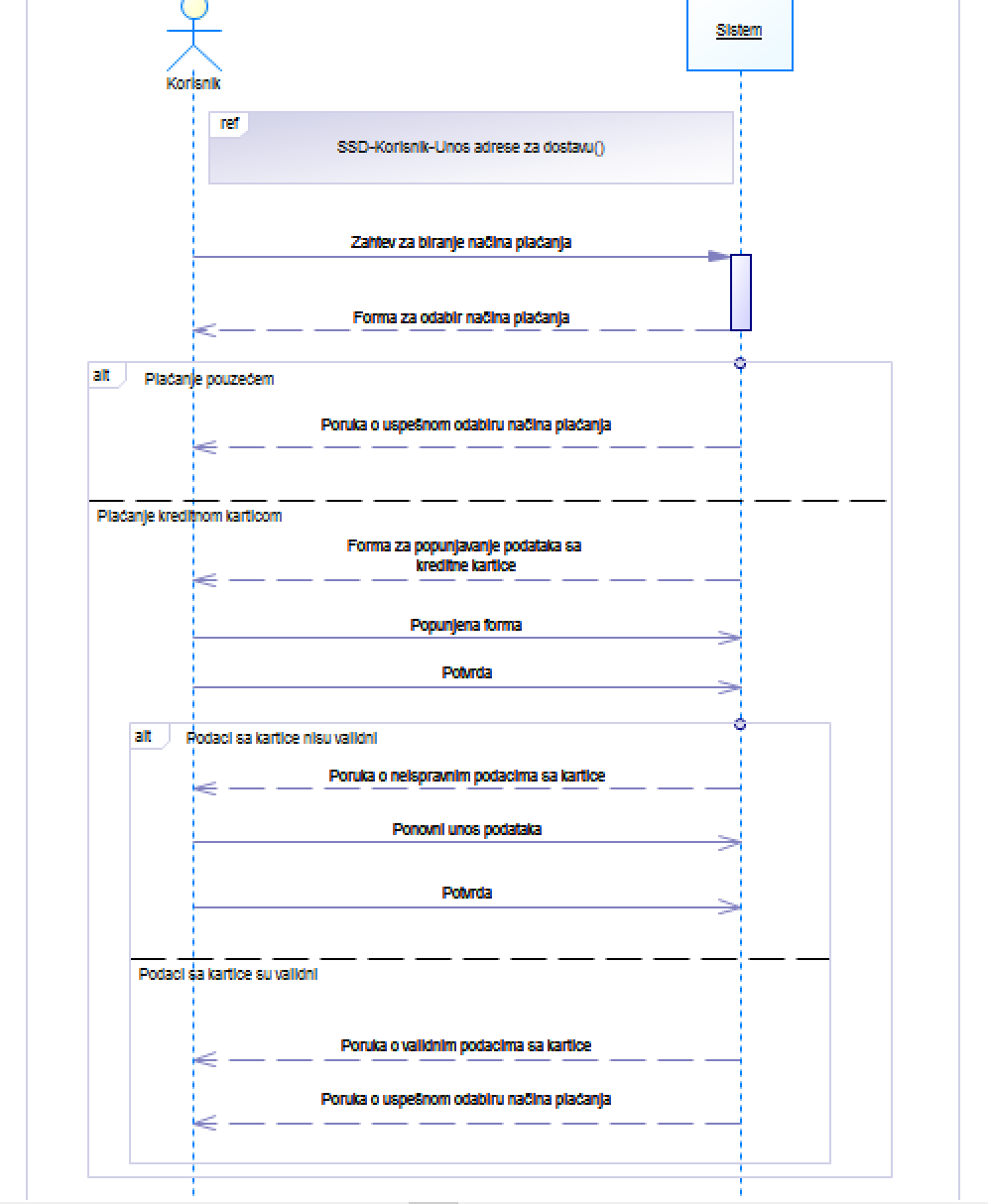


Slika 8.18.2: SSD za unos adrese za dostavu

***Biranje načina plaćanja***

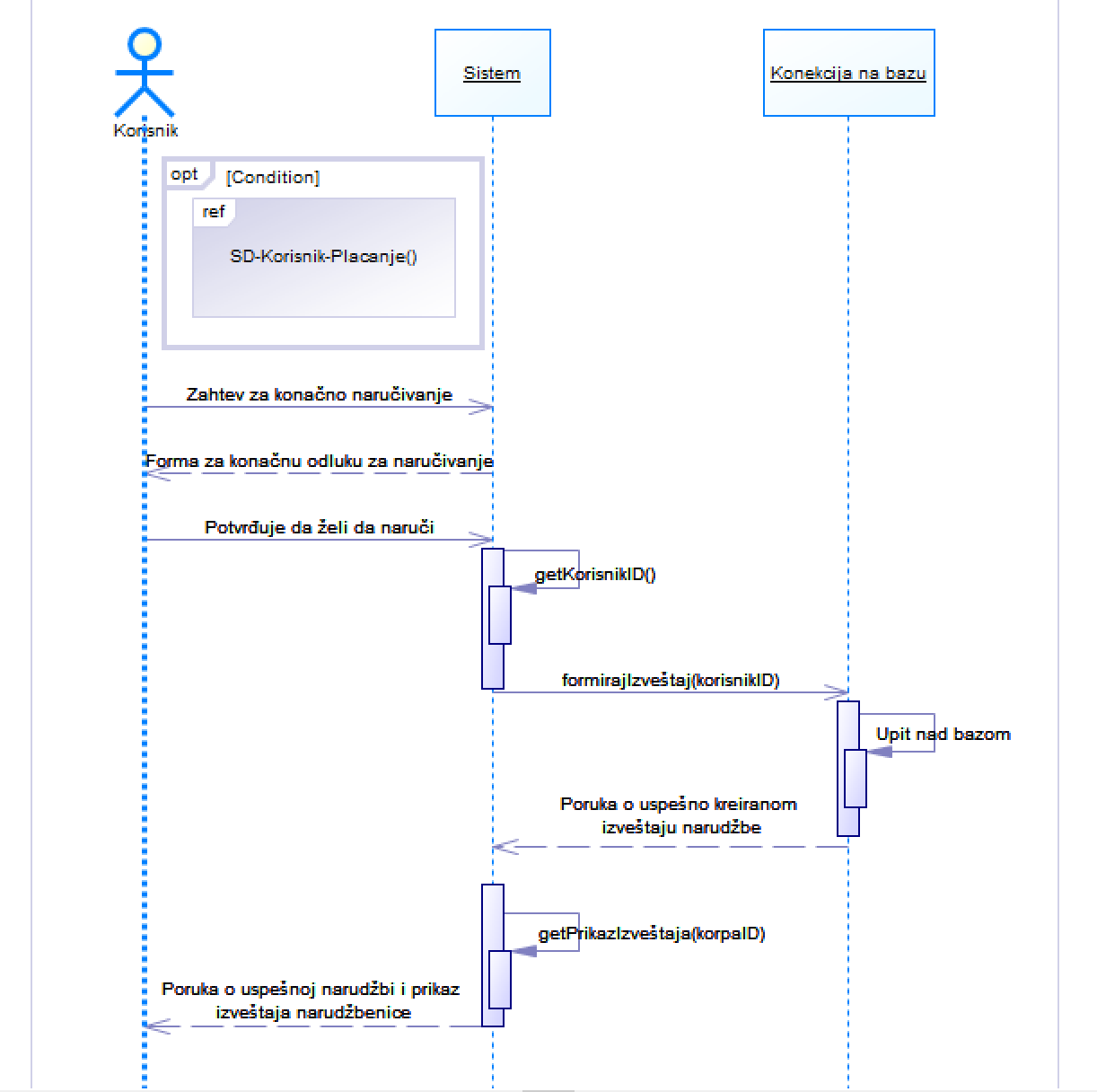
******

Slika 8.19.1: SD za izbor načina plaćanja

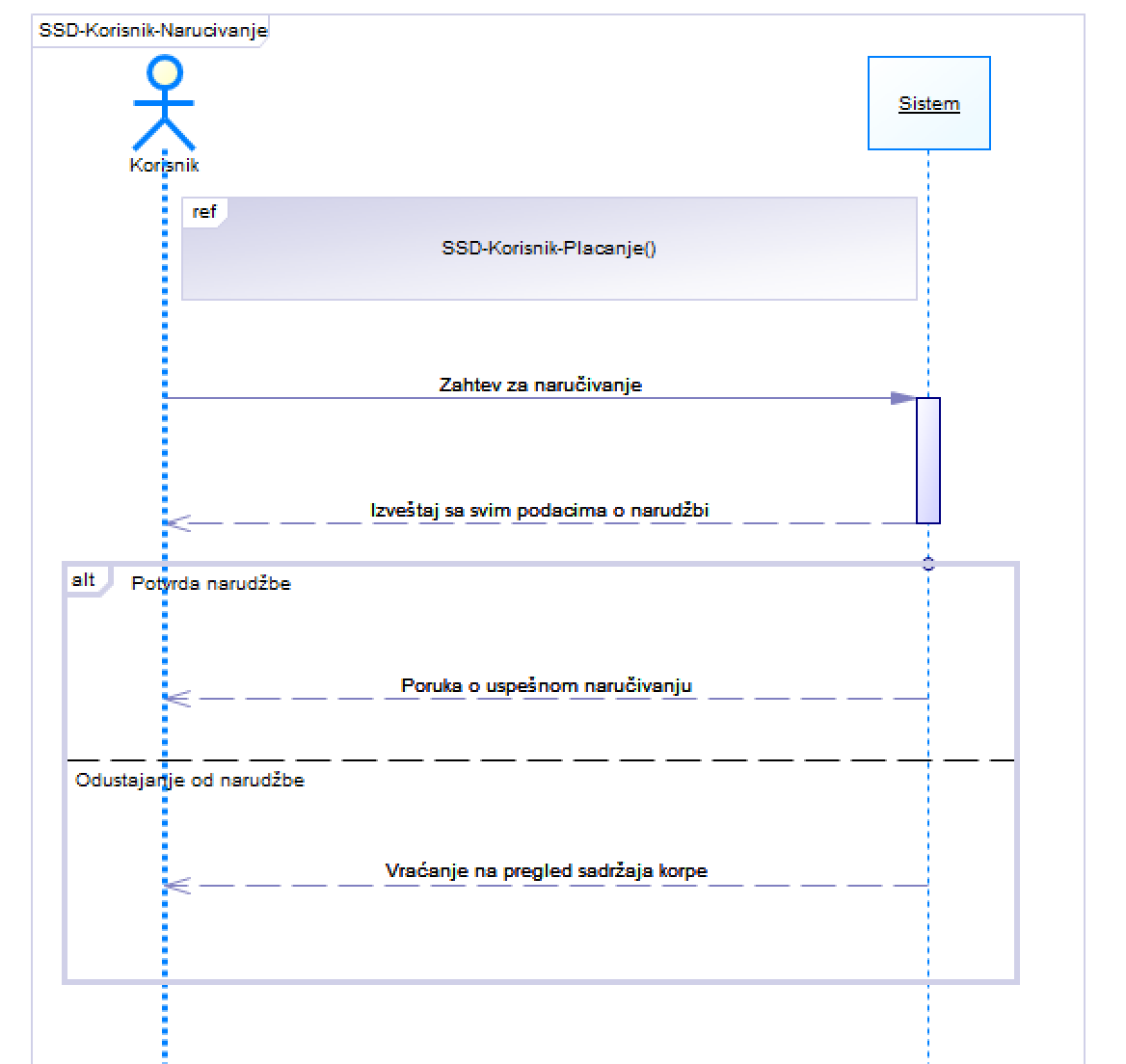


Slika 8.19.2: SSD za izbor načina plaćanja

***Naručivanje***

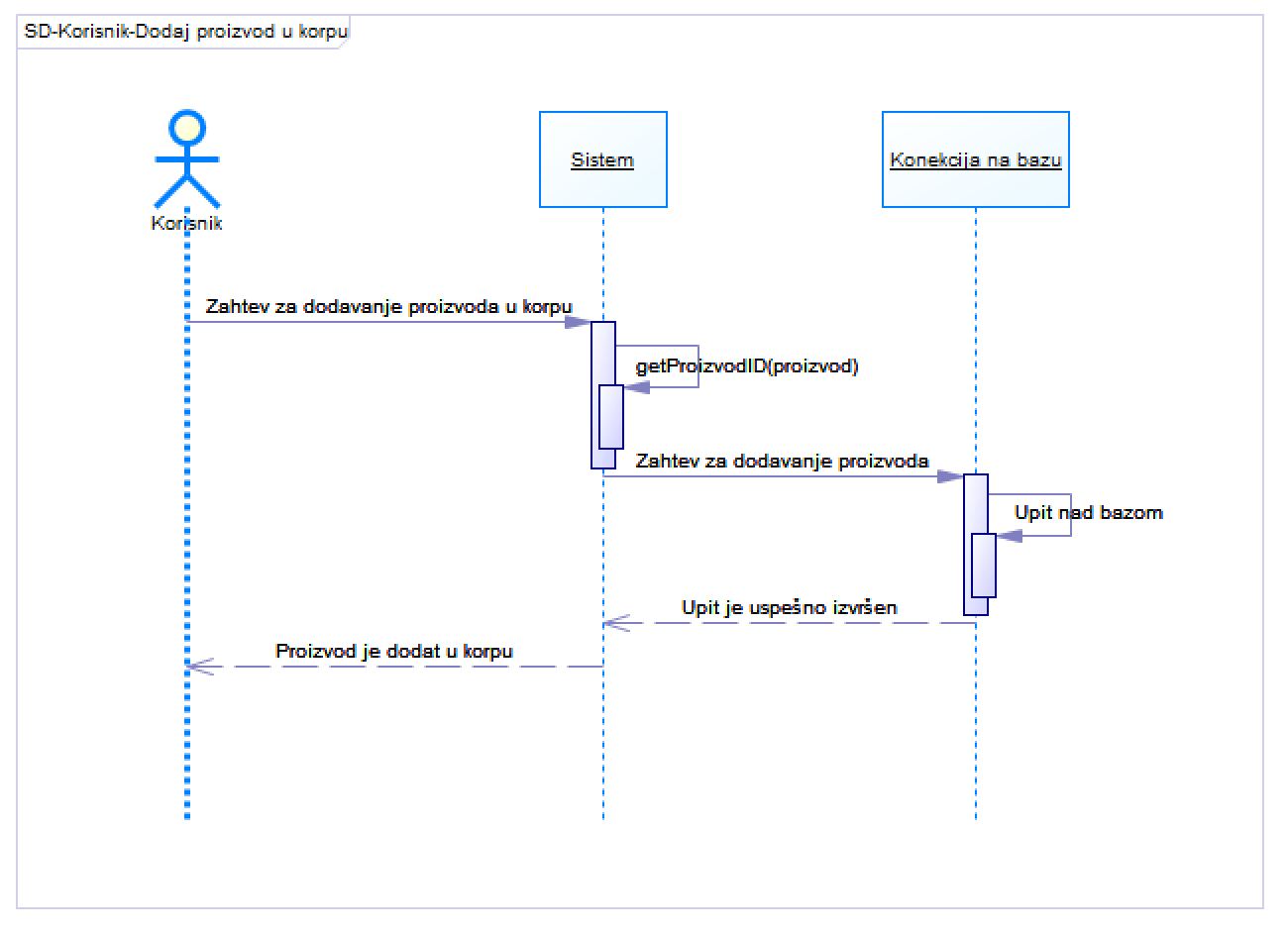
******

Slika 8.20.1: SD za naručivanje

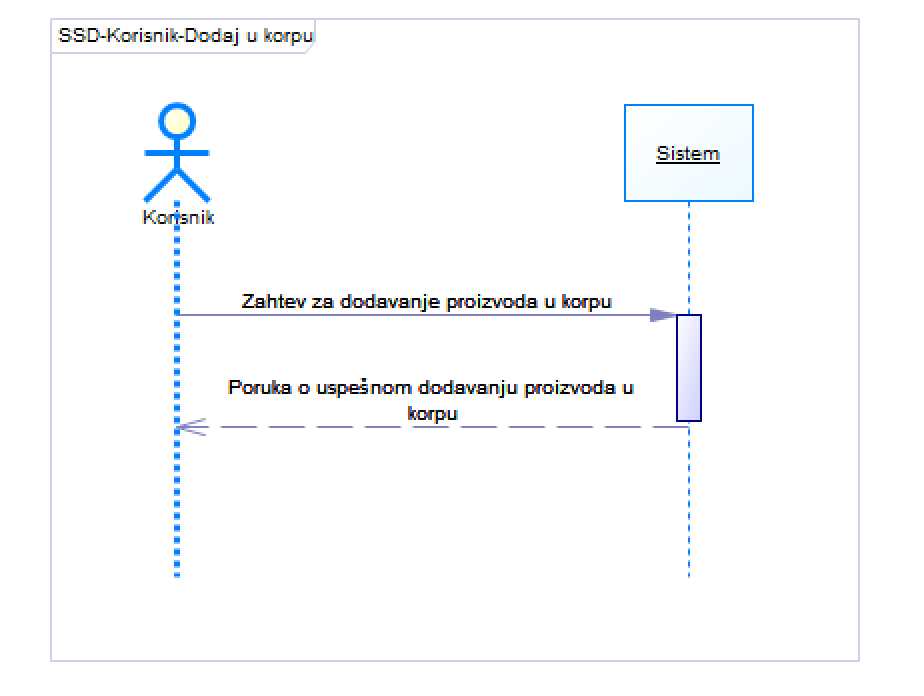


Slika 8.20.2: SSD za naručivanje

***Dodavanje proizvoda u korpu***



Slika 8.21.1: SD za dodavanje proizvoda u korpu



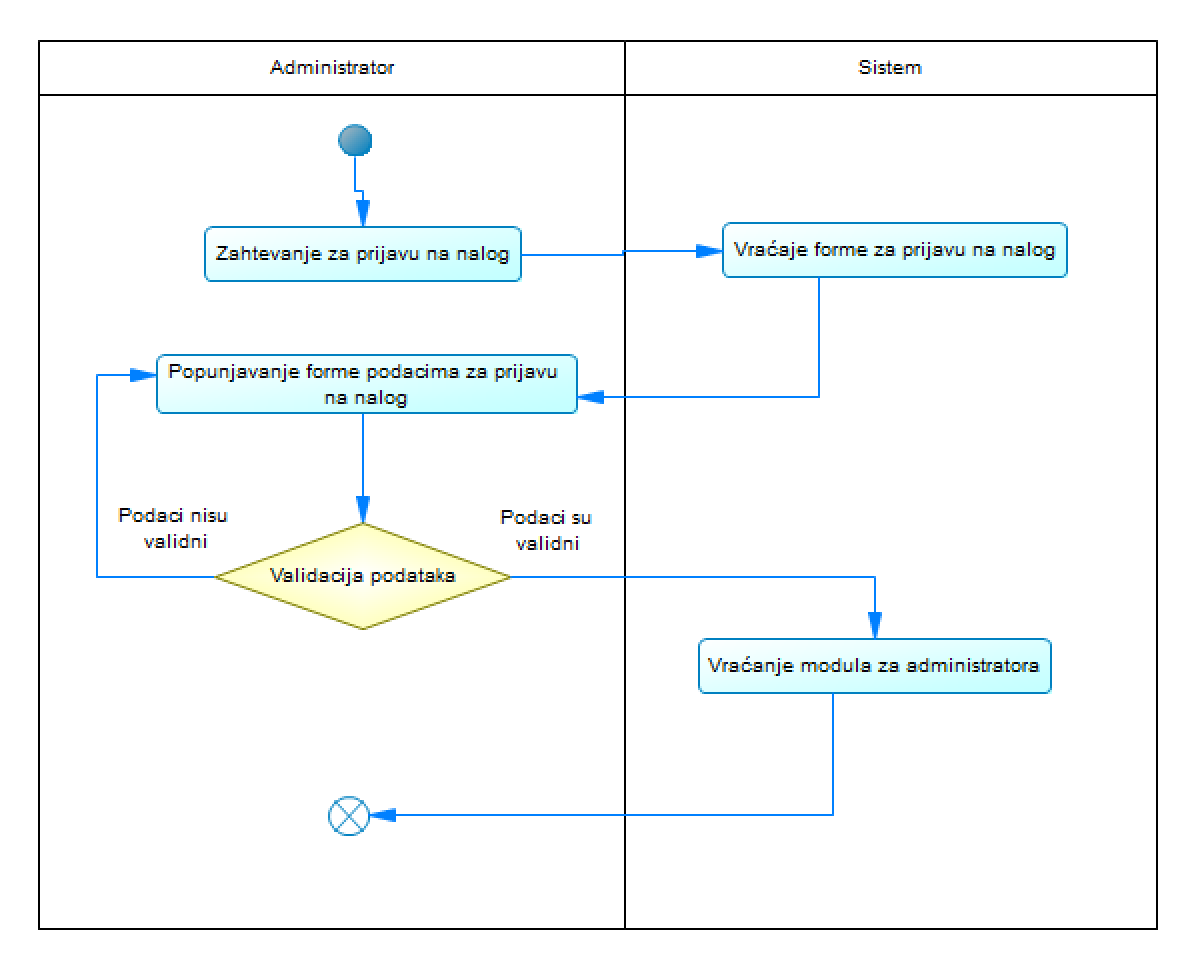
Slika 8.21.2: SSD za dodavanje proizvoda u korpu

### 4.4.2 Dijagram aktivnosti

U okviru dijagrama aktivnosti za svaki slučaj korišćenja biće prikazan scenarijo upotrebe uključujući i izuzetne situacije koje nisu grafički prikazane u okviru sekvencijalnih dijagrama.

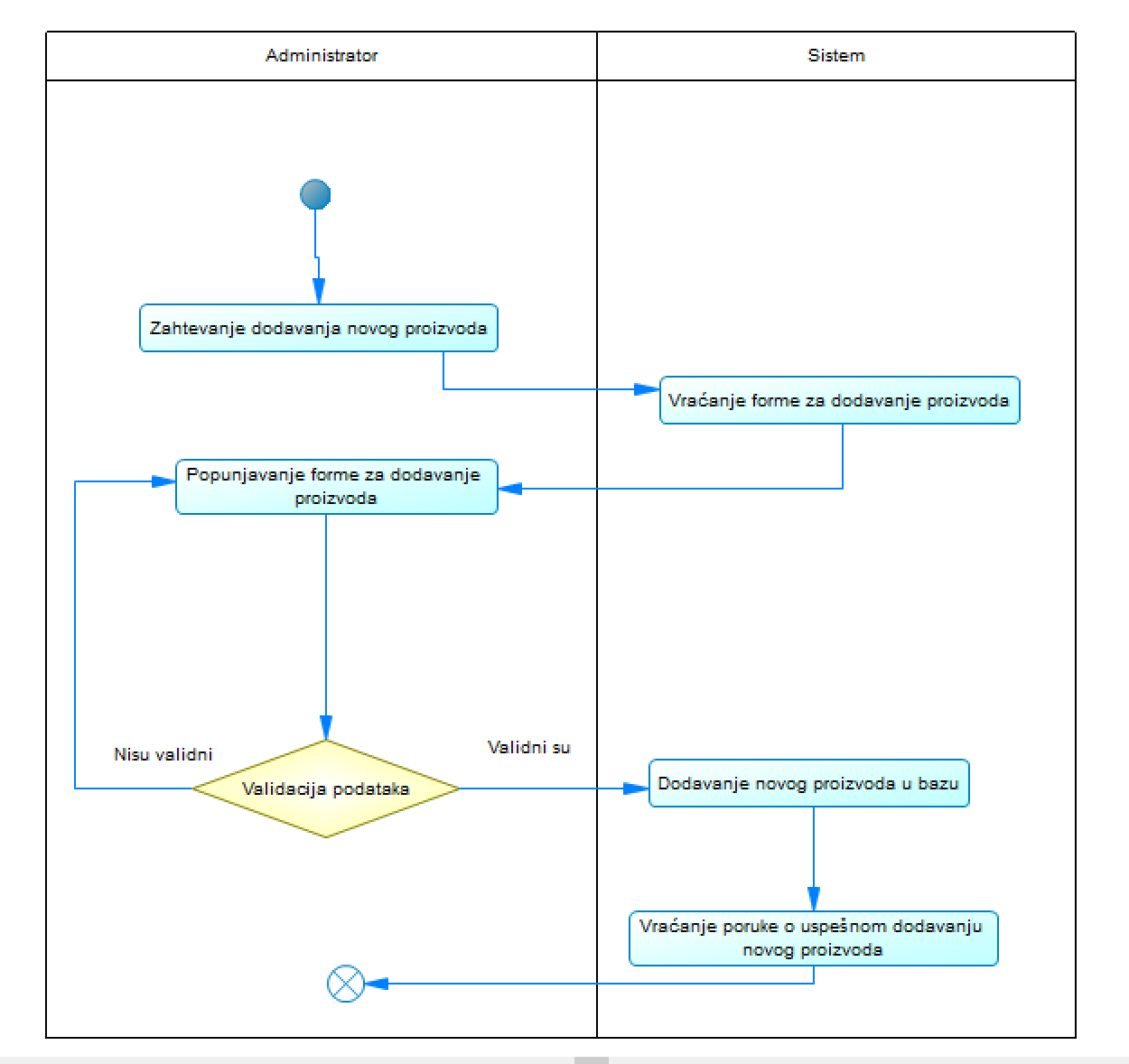
#### Administrator

##### ***Prijavljivanje na nalog***



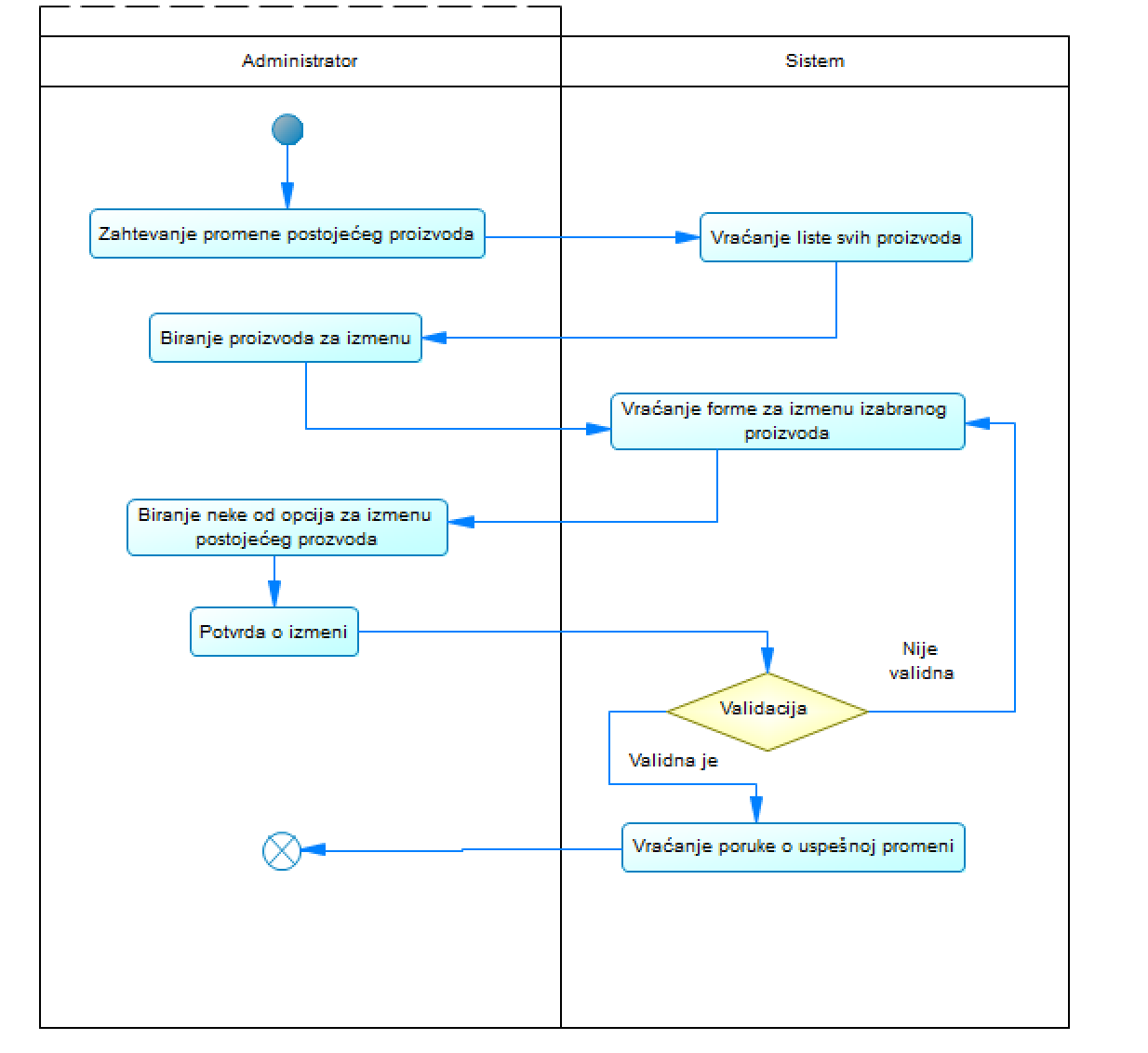
Slika 9.1: DA za prijavljivanje na nalog

##### ***Dodavanje proizvoda***



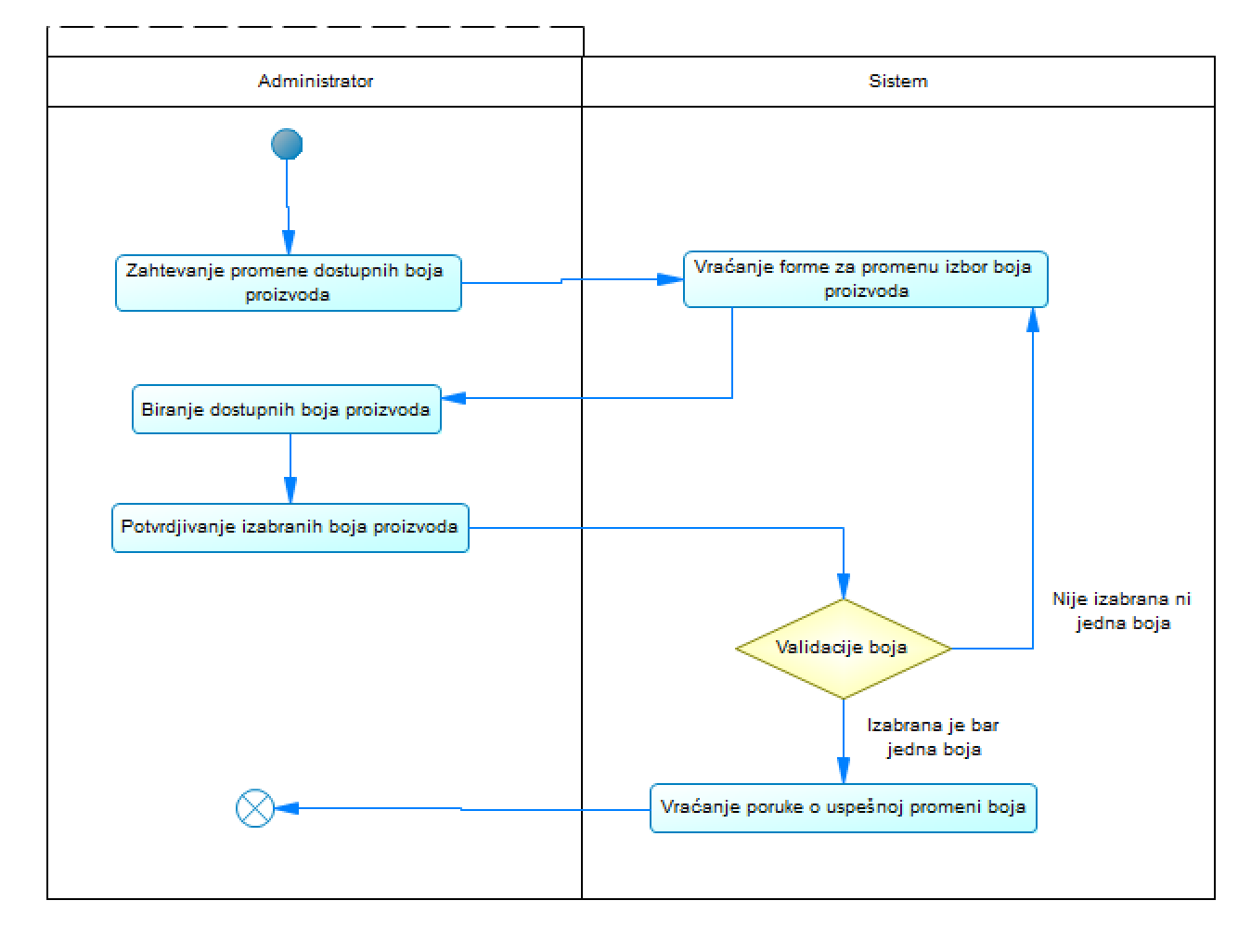
Slika 9.2: DA za dodavanje proizvoda

##### ***Promena postojećeg proizvoda***



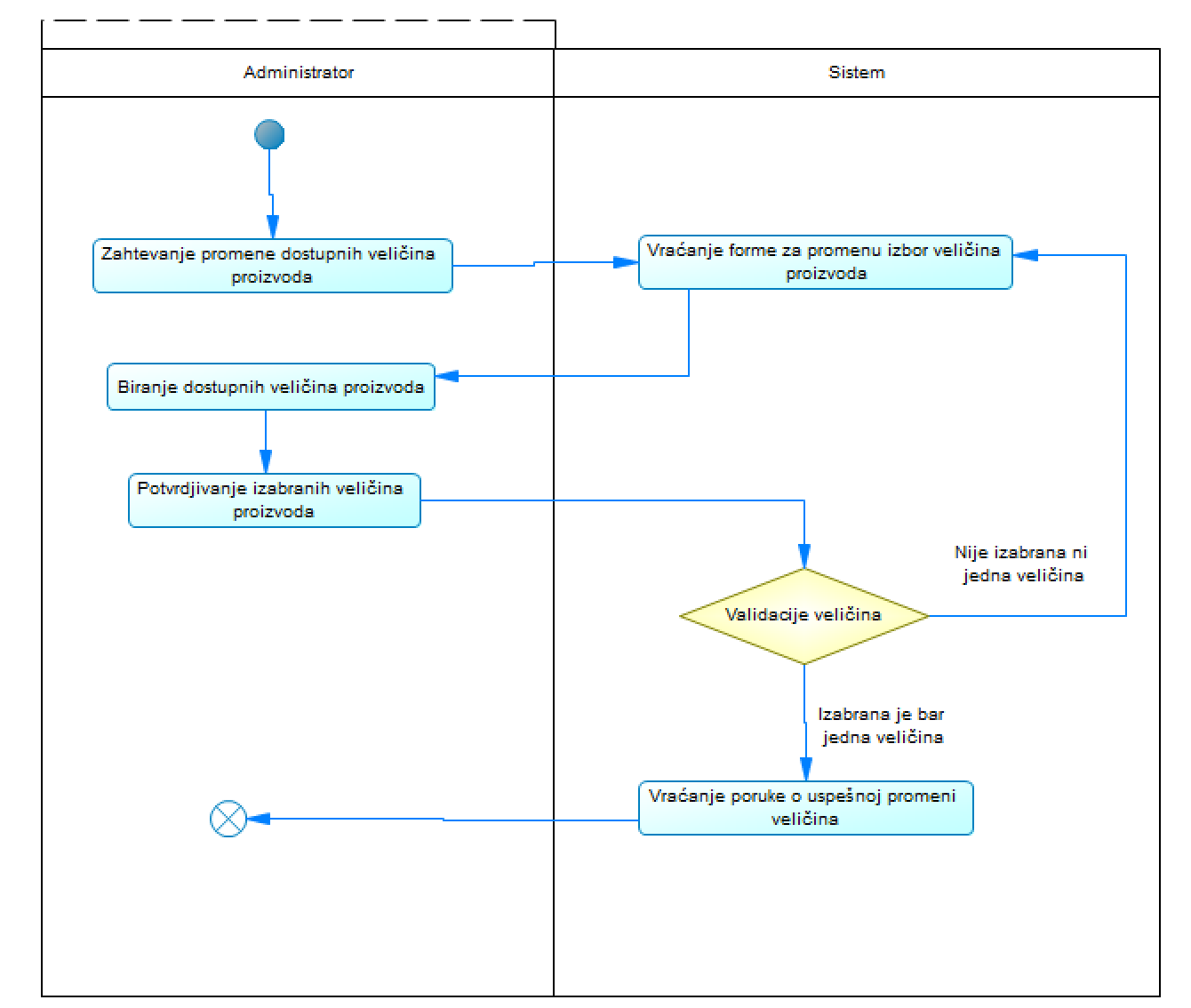
Slika 9.3: DA za promenu postojećeg proizvoda

##### ***Promena boja proizvoda***



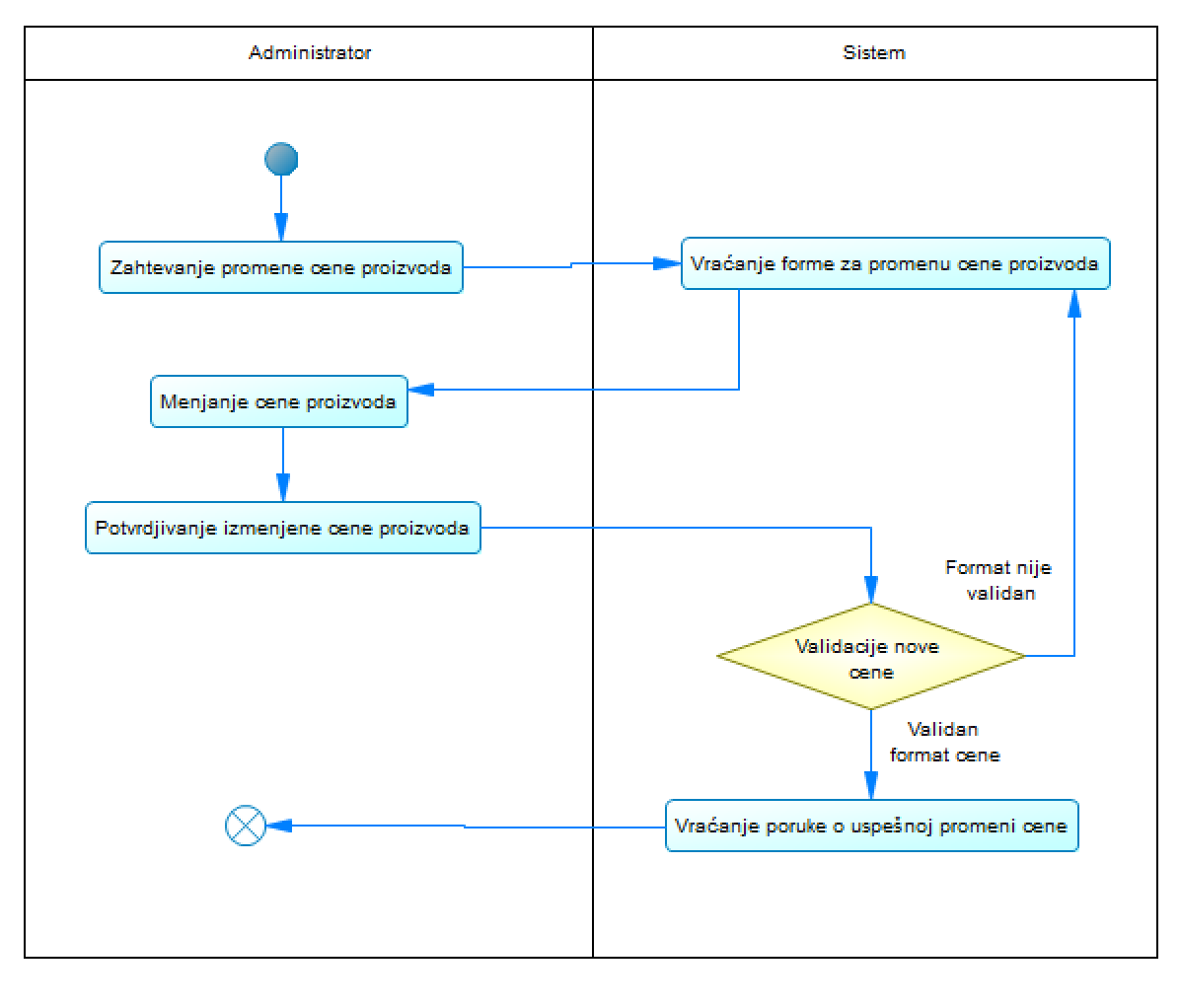
Slika 9.4: DA za promenu boja proizvoda

##### ***Promena veličina proizvoda***



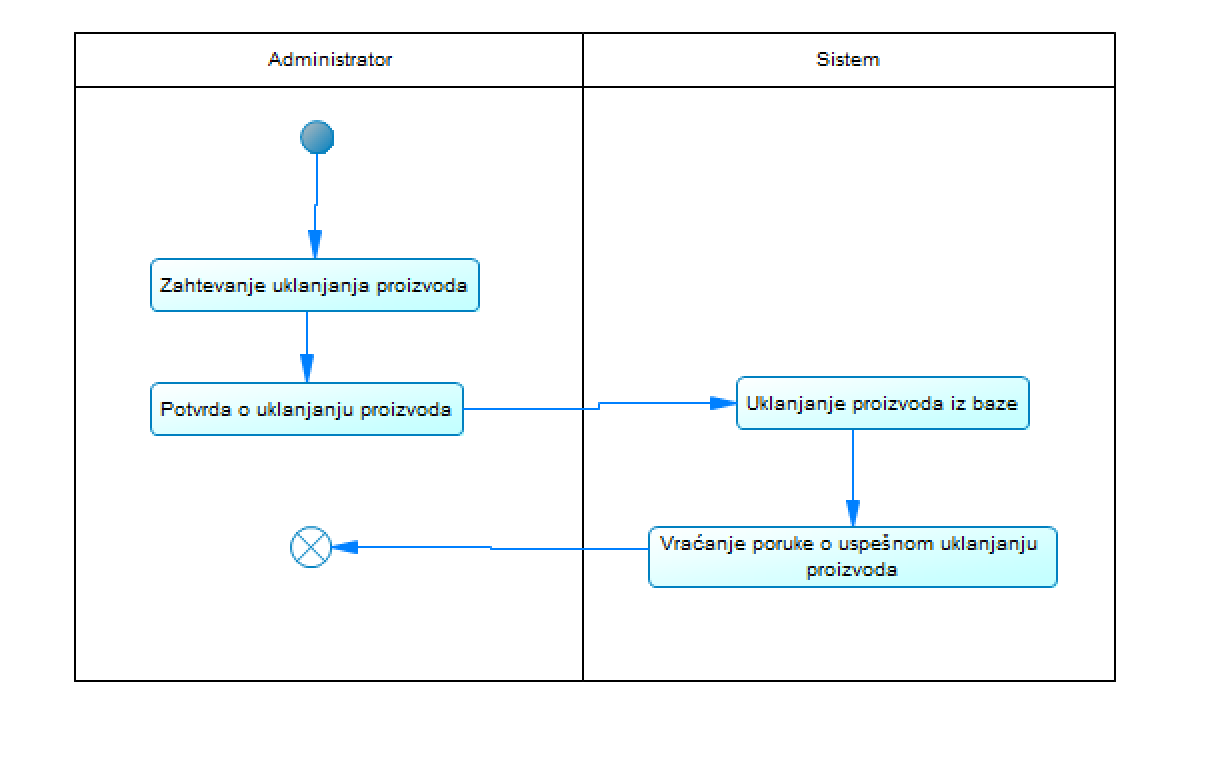
Slika 9.5: DA za promenu veličina proizvoda

##### ***Promena cene proizvoda***



Slika 9.6: DA za promenu cene proizvoda

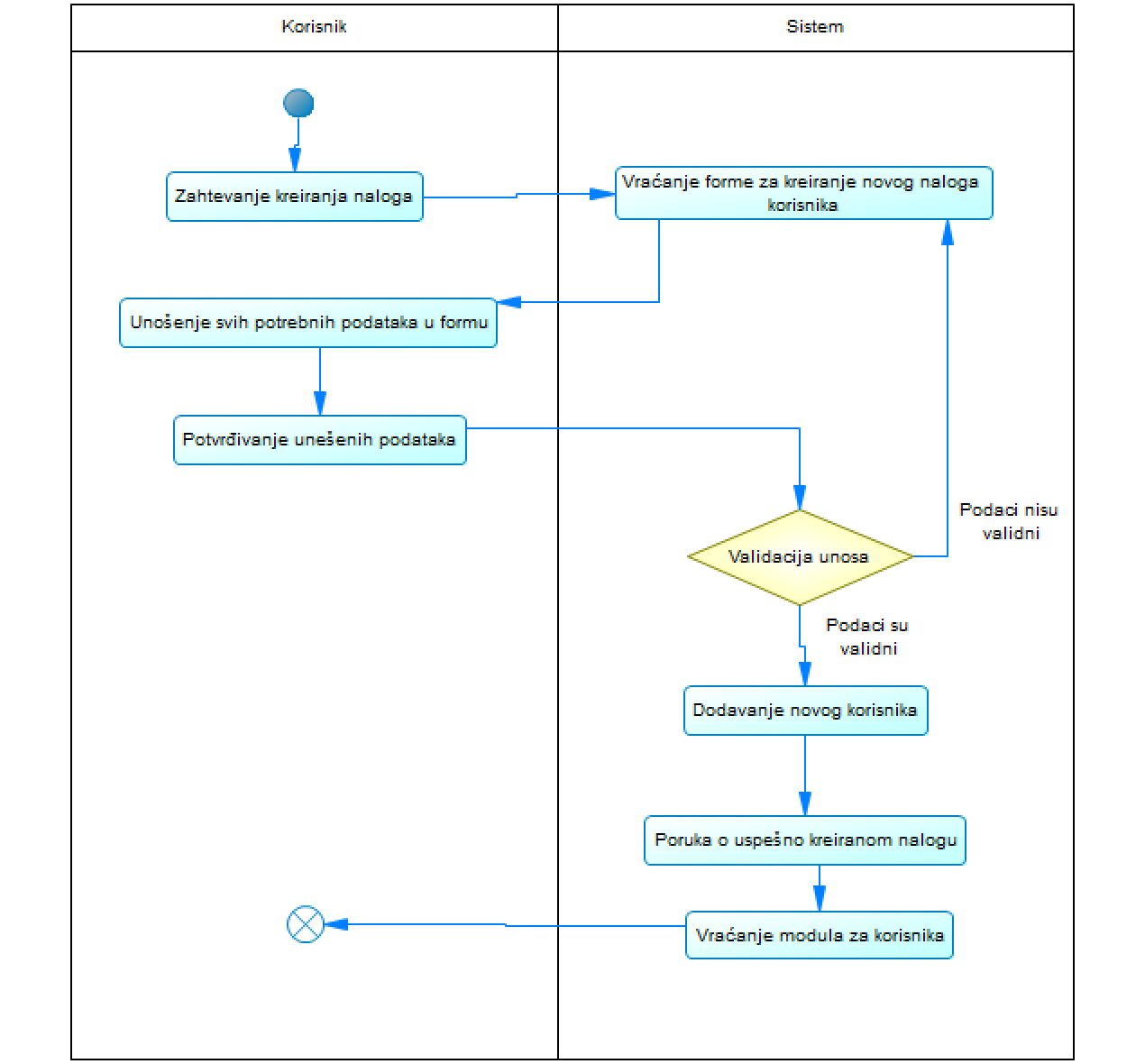
##### ***Brisanje proizvoda***



Slika 9.7: DA za brisanje proizvoda

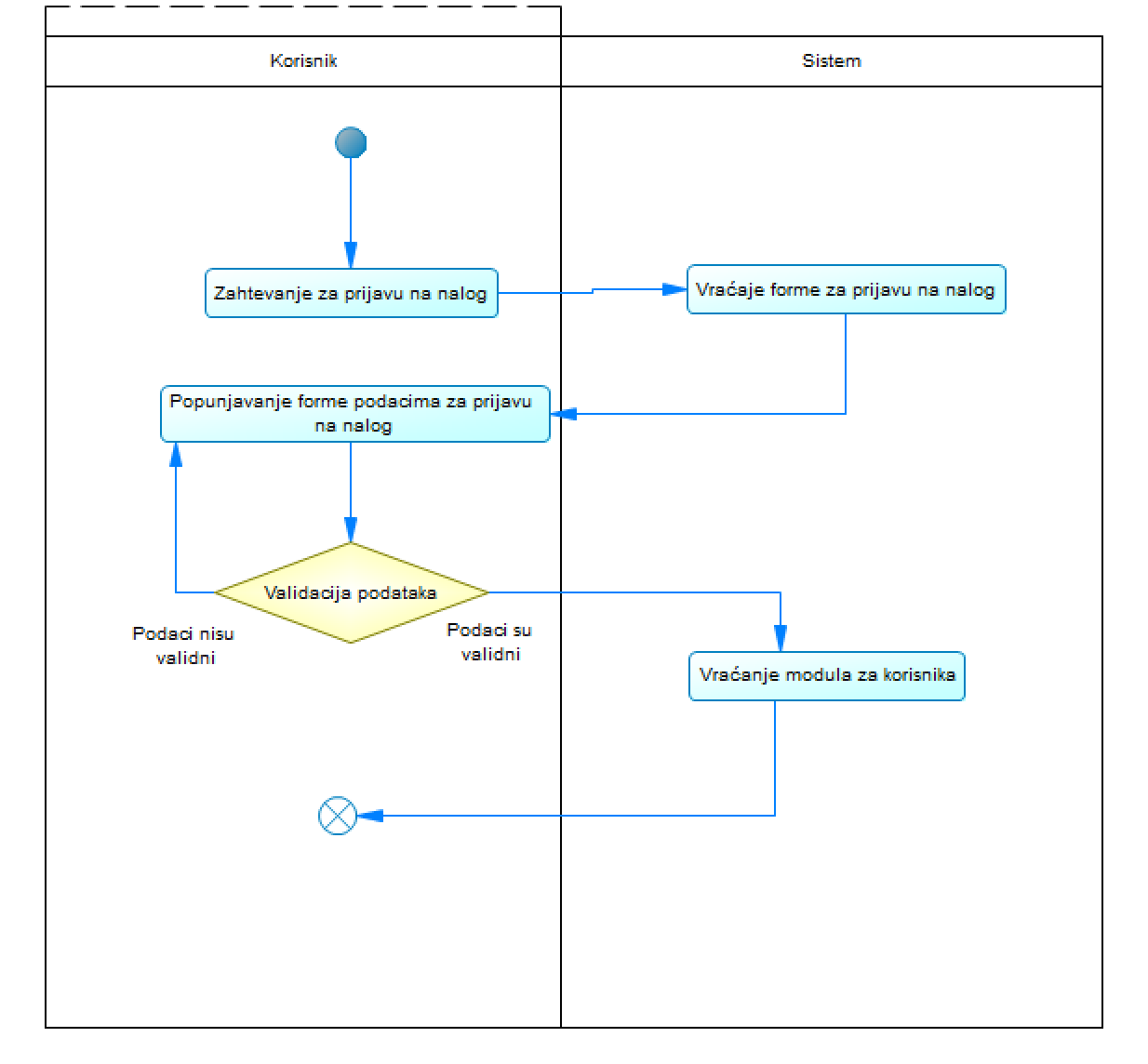
##### ***Korisnik***

##### ***Kreiranje naloga***



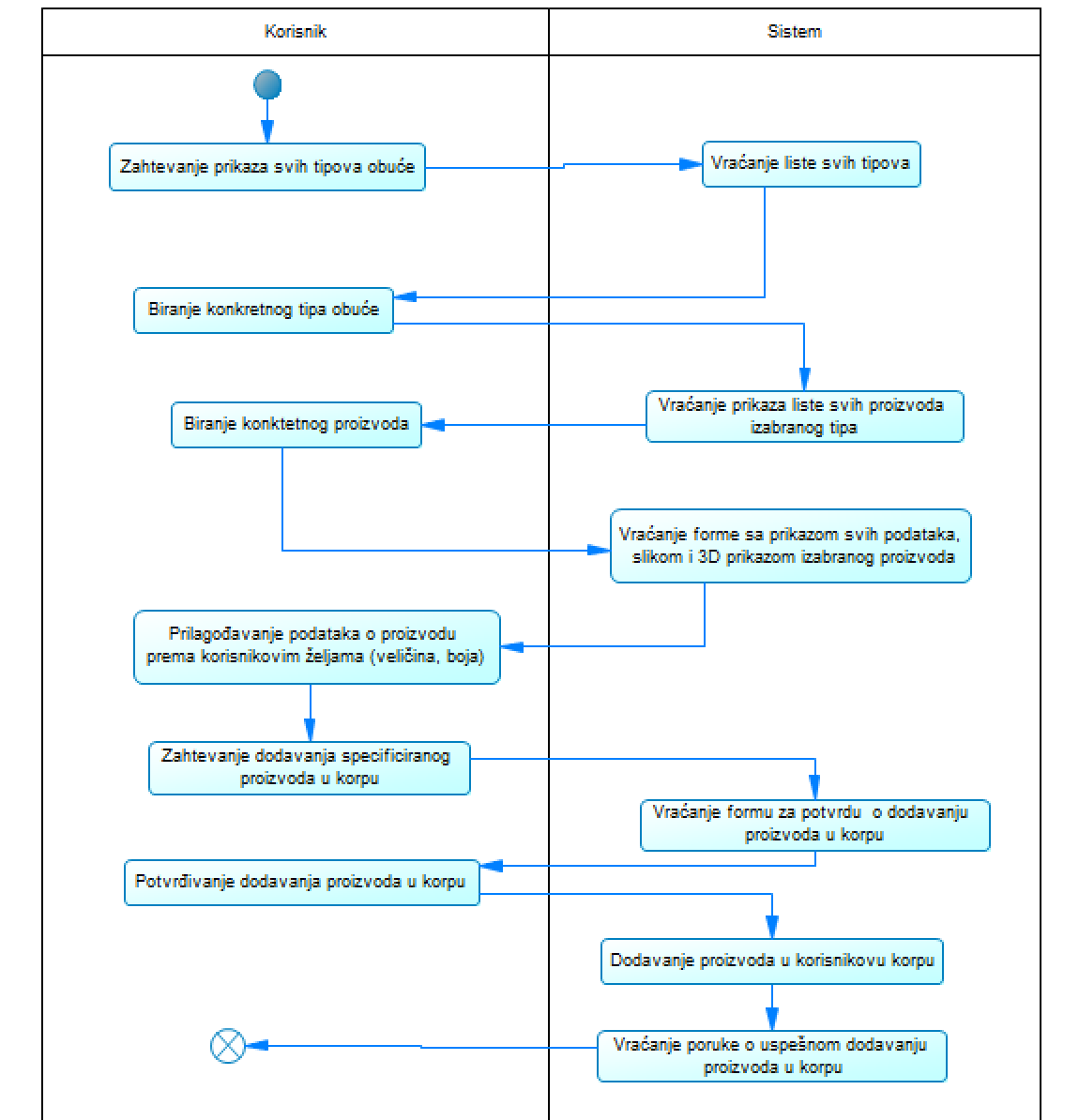
Slika 10.1: DA za kreiranje naloga

##### ***Prijavljivanje na sistem***



Slika 10.2: DA za prijavljivanje na sistem

##### ***Izbor obuće i dodavanje u korpu***



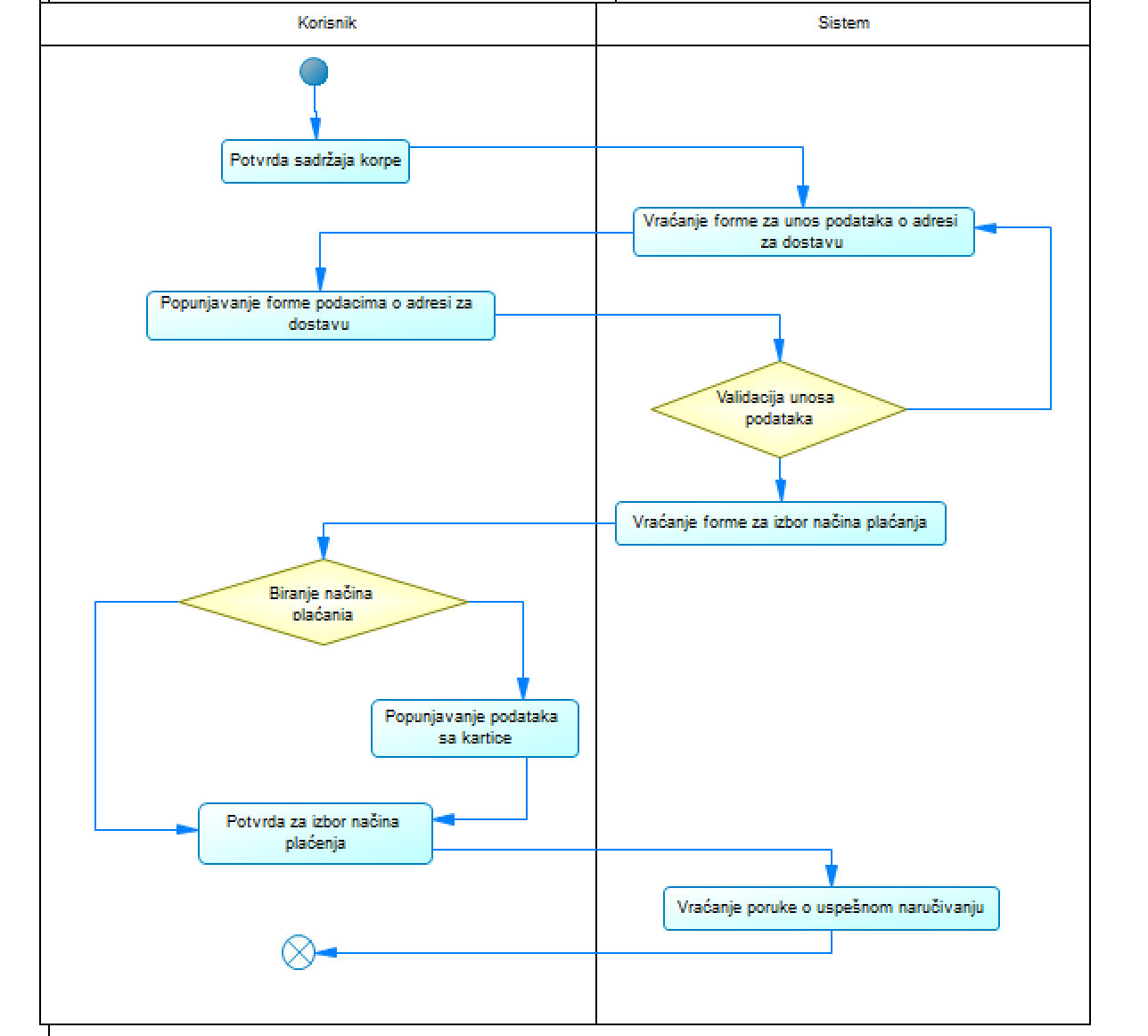
Slika 10.3: DA za izbor obuće i dodavanje u korpu

##### ***Pregled sadržaja korpe***



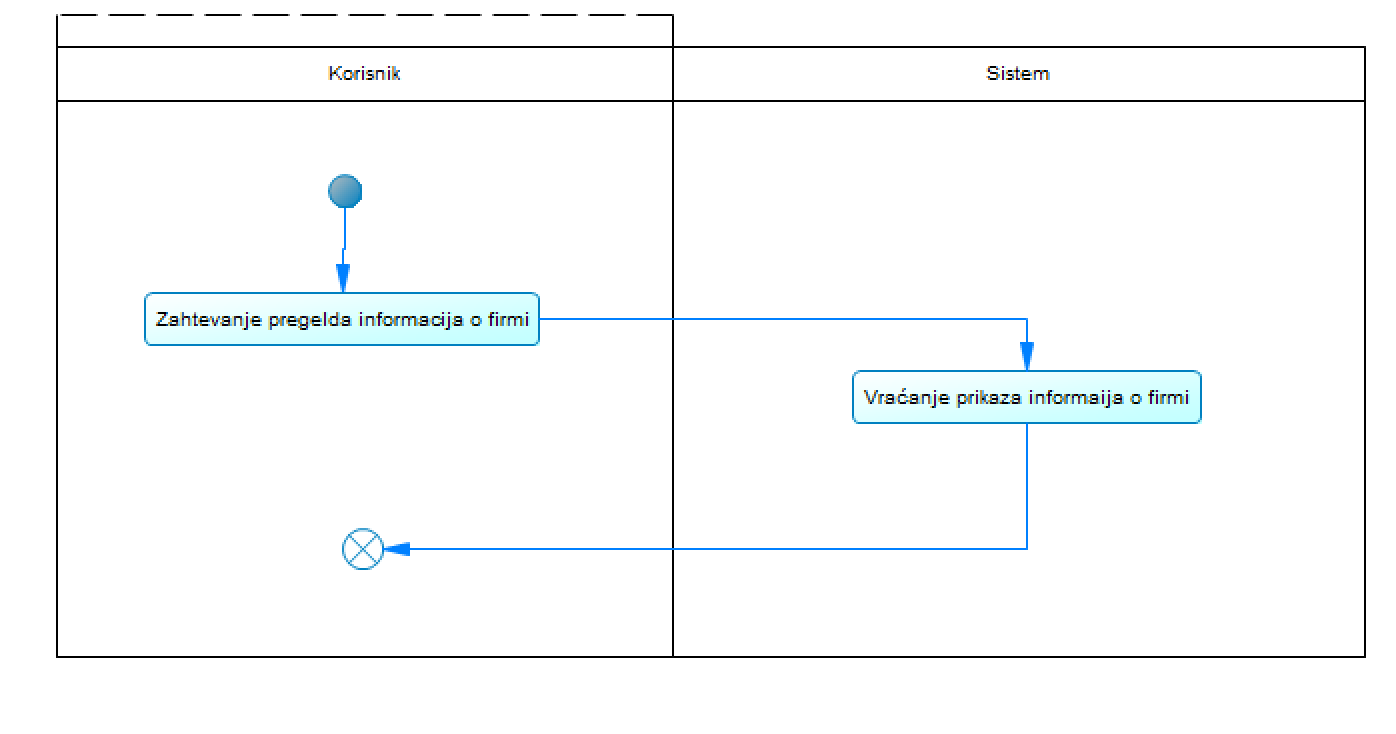
Slika 10.4: DA za pregled sadržaja korpe

##### ***Naručivanje***



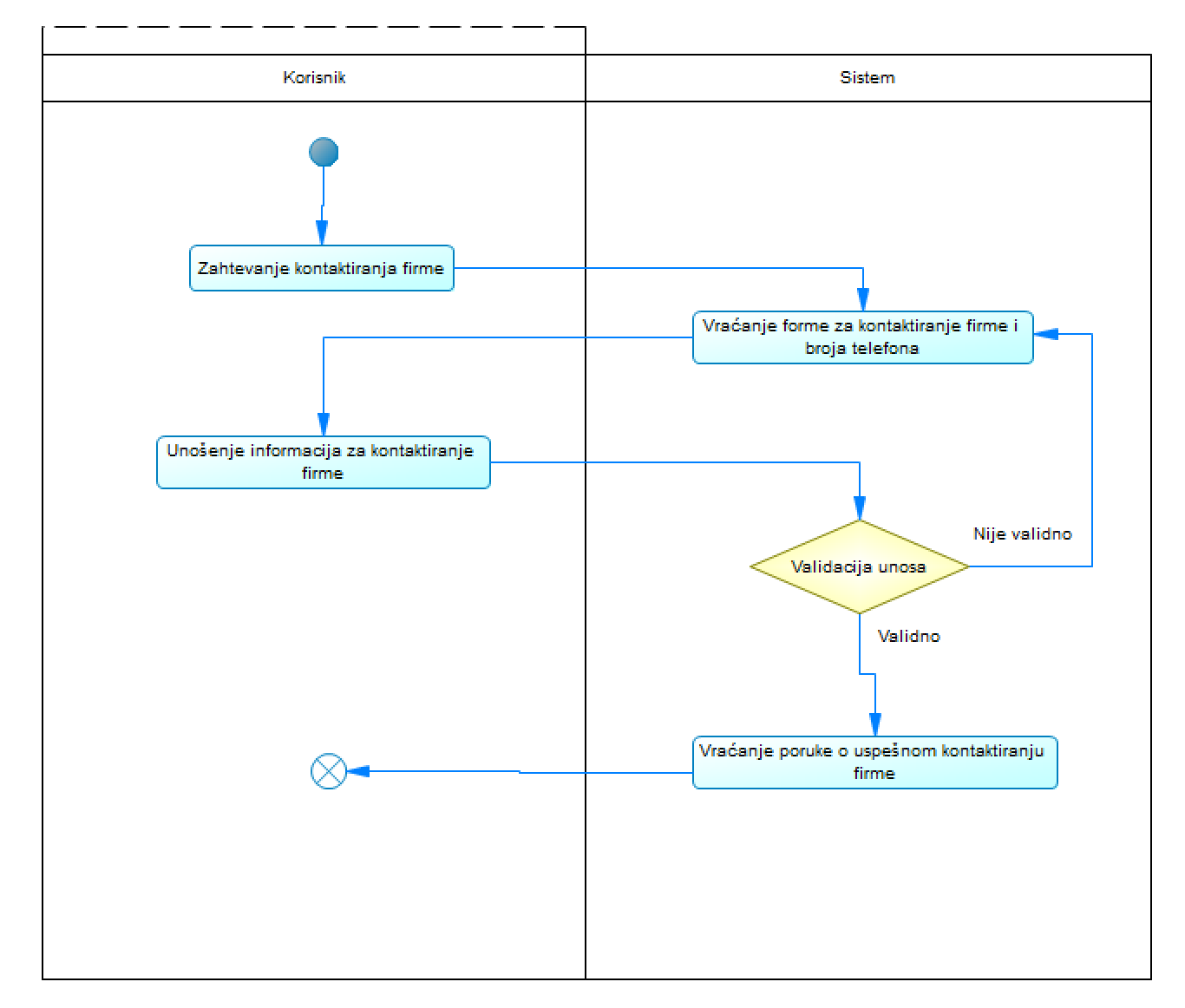
Slika 10.5: DA za naručivanje

##### ***Pregled informacija firme***



Slika 10.6: DA za pregled informacija firme

##### ***Kontektiranje firme***



Slika 10.7: DA za kontaktiranje firme

**5. Implementacija i testiranje**

## Korišteni dizajn paterni

#### MVC (Model View Controller)

MVC razdvaja kod u tri sloja, to jeste: Model, View I Controler.

Važno je naglasiti da ovi moduli komuniciraju i iniciraju radnje međusobne. Ako se opredelimo za Spring framework, potrebno je da se izaberemo i tehnologiju za mapiranja entiteta iz baze u same klase (engl. Object oriented programming).

Da bismo mogli da mapiramo tabele u odgovarajuće entitet klase koristimo Hibernate ORM (Object Relational Mapping) okvir. Mapiranje svake klase možemo definisati i podesiti u okviru konfiguracionog fajla samog Hibernate sloja. Pomoću Hibernate okvira moguće je vršiti manipulaciju nad bazom bez direktnog korišćenja SQL jezika. Hibernate nam nudi klase u okviru kojih su implementirane metode za manipulaciju sa bazom podataka i to se realizuje prethodnom manipulacijom sa entitet klasama. Pored toga Hibernate omogućava da kreiramo upite korišćenjem HQL (Hibernate Query Language) jezika.

Kod Java Spring okvira vizuelni deo, tj. sloj (View) čine html fajlovi u kojima se definiše izgled veb aplikacije pisanjem HTML koda koji se stilizuje pomoću CSS tehnologije. Za stilizovanje je korišćen CSS-ov okvir Bootstrap radi lakše prilagodljivosti različitim veličinama ekrana. U okviru View segmenta mogu se koristiti i odgovarajuće grafičke komponente koje nam nudi sam framework. Pored korišćenja gotovih komponenti u okviru razvoja aplikacije kreirane su i sopstvene komponente.

Kada je reč o delu namenjenom za razvoj same logike (engl. Business logic) odnosno kontroleru on je realizovan kreiranjem Javinih klasa koje čine sastavni deo svake strane. U okviru java klasa koje pripadaju kontroleru definišu se svi objekti potrebni za realizaciju odgovarajućih biznis problema. Kontroler je međusloj koji komunicira i sa View-om i sa Modelom. Dakle, kontroler ima zadatak da obrađuje podatke koje koristi iz model-a i da ih nakon obrade prosleđuje view sloju gde će oni biti reprezentovani krajnjim korisnicima. Svi podaci koji su nastali obradom u okviru kontrolera mogu se proslediti view-u kako bi ih prikazao, dok se komunikacija kontrolera sa modelom obavlja pomoću dodatnog DAO (Data Access Object) sloja koji je u dijagramu arhitekture nazvan ‘’Insertion’’.

Takođe treba napomenuti da je za bazu podataka odabran mySQL RDBMS (Relational Data Base Menagment Sistem). Dakle, koristiće se relaciona baza podataka sa kojom će se raditi u kombinaciji sa Hibernate ORM okvirom koji je već pomenut.

## Testiranje sistema

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | Ime | Opis | **Koraci** | | | **Satus** |
| Korak | **Očekivano** | **Realno** |
| **Cena proizvoda** | **Validacija unete cene proizvoda** | **Verifikacija cene** | **Format unete cene proizvoda sadrži karaktere** | **Greška: ‘’Niste uneli cenu u validnom formatu** |  | **F** |
| **Format unete cene proizvoda sadrži samo brojeve** | **Cena je uspešno dodata. Vraća se poruka o uspešnom dodavanju cene** |  | **P** |
| **Korisnik** | **Validacija lozinke** | **Da bi lozinka bila validna mora da sadrži male i velike karaktere kao i bar jedan broj** | **Unesena lozinka sadži samo jednu vrstu karaktera** | **Prikazuje se poruka sa greškom “Unesite lozinku koja sadrži mala, velika slova, kao i bar jednu cifru”** |  | **F** |
| **Unesena lozinka sadrzi i mala i velika slova, ali i bar jednu cifaru** | **Prikaz poruke o validnom unosu lozinke** |  | **P** |

**6. Zaključak**

Ovaj projekat predstavlja sistem čija upotreba nije nepoznata, tj. većina korisnika je upoznata sa principom kupovine preko interneta. Pored dela koji korisnik vidi, u ovom projetu je prikazan i deo koji nije vidljiv korisnicima tj. način na koji sistem funkcioniše. Različite vrste dijagrama su korištene kako bi se prikazao način rada samog sistema i način njegove interakcije sa korisnicima sistema. Sistem je implementiran za dve vrste korisnika, a to su administratori sistema, i kupci. Ovaj dokument je značajan za prikaz detaljne realizacije čitavog projekta, što je značajno za održavanje sistema. Dokumentacija je vrlo značajna kako za korisnike sistema, tako i za ljude koji se nalaze iza samog sistema ali i za one koji su pravili sistem od početka(kao podsetnik na neke detalje koji su značajni, a lako se zaboravljaju).

**7. Literatura**

1. Dragan Domazet, Aleksandar Slijepčević. Uvod u Softversko Inženjerstvo (Prvo izdanje), Beograd 2014.
2. Ian Sommerville. *Software Engineering* (Ninth edition) Addison-Wesley, 2011.