**VISOKA SKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE TEHNOLOGIJE**

****

Web programiranje

**CompShop – online prodavnica računara i računarske opreme**

Predmetni nastavnik: Student:

**Prof. Dr Selver Pepić Lazar Adjancic 530/17**

# **UVOD**

# **Korišćeni programski jezici**

Tokom projekta su korišćeni sledeći programski I markup jezici:

* HTML
* CSS
* JavaScript
* JQuery
* Bootstrap
* AJAX
* PHP

# **Opis funkcionalnosti**

## **1.21 Registracija I logovanje**

Na sajtu postoje stranice za registraciju i logovanje. Prilikom uspešne registracije korisnik kreira svoj nalog I upisuje se u bazu. Provera podataka se nalazi na serverskoj strani. Nakon registracije korisnik se moze ulogovati na stranici za logovanje. Prilikom uspešnog logovanja kreira se sesija sa svim podacima o korisniku. Provera se takodje vrši na serverskoj strani I ukoliko uneti podaci nisu validni korisnik dobija prikaz o postojanju greške.

## **1.22 Slanje mail-a I reklamacija**

Na sajtu postoji kontank stranica. Kontakt stranici imaju pristum svi korisnici (i sa i bez korisnickog naloga). Predstavlja formu preko koje se mogu slati pitanja adminu putem mail-a, kao i kritike, pohvale ili reklamacije. Potrebno je samo uneti neke osnovne podatke o korisniku. Nakon uspešno poslatog mail-a, mail se upisuje direktno u bazu.

## **1.23 Proizvodi**

Proizvodi se dinamički učitavaju iz baze. Na stranici na kojoj se nalaze proizvodi, postoji mogućnost pretraživanja proizvoda po id-u, nazivu ili po tipu proizvoda(desktop, laptop, dodatna oprema). Proizvodi se takodje mogu sortirati po ceni ili imenu.

## **1.24 Kupovina**

Klikom na dugme “Dodaj u korpu“ proizvod se automtski pojavljuje na stranici gde se nalazi korpa. Moze se odrediti količina odredjenog proizvoda koji će se dodati u korpu. U korpi se prikazuje lista proizvoda za kupovinu, cena svakog pojedinačnog proizvoda, cena svakog proizvoda na osnovu količine(cena\*kolicina) I ukupna cena svih proizvoda u korpi. Nakon obavljene kupovine kreira se narudzbenica u txt fajlu, narudzbenica se automatski upisuje u bazu.

# **1.3 Tema**

Tema projetka je Cyberspace, online prodavnica računara I računarske opreme. Korisnicki mogu izvršiti kupovinu online porzvoda koji su u ponudi, pretražiti sekciju proizvoda.

# **Organizacija**

# **Organizaciona šema sajta**

Sa svake strane se moze pristupiti svakoj strani, osim stranica za dodavanje, brisanje I azuriranje podaka o korisnicima I proizvodima koje su dostupne isljučivo administratorima sajta.

# **USE CASE – korisnik**



Slika 1. Use Case dijagram korišćenja

Na osnovu use case dijagrama uočavamo sledeće slučajeve korišćenja:

1. Registracija
2. Logovanje
3. Slanje email-a
4. Pretraživanje proizvoda
5. Kupovina

# **Verbalni opis**

**UseCase1: Registracija**

Naziv: Registracija

Aktor: Korisnik

Učesnici: Korisnik i aplikacija

Preduslov: Aplikacija je pokrenuta, korisnik nema korisnički nalog i pristupa registrovanju.

Osnovni scenario:

1. Aplikacija prikazuje formu za registraciju
2. Korisnik unosi podatke o sebi (ime, prezime, email, korisničko ime, sifru)
3. Korisnik poziva aplikaciju da kreira korisnika sa zadatim podacima
4. Aplikacija proverava postojanje korisnika sa zadatim korisničkim imenom
5. Aplikacija kreira korisnika i upisuje ga u bazu

Alternativni scenario:

1. Aplikacija pronalazi korisnika sa zadatim korisničkim imenom, i prikazuje poruku da takav korisnik već postoji
2. Aplikacija ne moze da kreira korisnika sa unetim podacima, i prikazuje odgovarajuću poruku

**UseCase2: Logovanje**

Naziv: Logovanje

Aktor: Korisnik

Učesnici: Korisnik i aplikacija

Scenario: Aplikacija je pokrenuta, korisnik ima napravljen korisnički nalog i pristupa logovanju.

Osnovni scenario:

1. Korisnik unosi korisničko ime i lozinku
2. Korisnik poziva aplikaciju da proveri postojanje korisnika sa zadatim korisničkim imenom i lozinkom
3. Aplikacija proverava postojanje korisnika
4. Aplikacije prebacuje korisnika na početnu stranu i otpočinje sesiju

Alternativni scenario:

1. Aplikacija ne pronalazi korisnika, nema korisničkog imena
2. Aplikacija pronalazi korisnika sa zadatim korisničkim imenom, ali je lozinka neispravna i prikazuje poruku o grešci

**UseCase3: Slanje email-a**

Naziv: Slanje email-a

Aktor: Korisnik

Učesnici: Korisnik i aplikacija

Scenario: Aplikacija je pokrenuta i korisnik se nalazi na kontekt strani.

Osnovni scenario:

1. Korisnik popunjava formu za slanje mail-a
2. Korinsik poziva aplikaciju da pošalje mail
3. Aplikacija šalje mail administratoru
4. Aplikacija prikazuje poruku o uspešno poslatom mail-u

Alternativni scenario:

1. Aplikacija ne uspeva da pošalje mail administratoru

**UseCase4: Pretraživanje proizvoda**

Naziv: Pretraživanje korisnika

Aktor: Korisnik

Učesnici: Aplikacija i korisnik

Preduslov: Aplikacija je pokrenuta i korisnik se nalazi na strani sa proizvodima.

Osnovni scenario:

1. Korisnik pretrazuje proizvode po id-u, imenu ili tipu proivoda
2. Korisnik poziva aplikaciju da pretraži proizvode
3. Aplikacija provera postojanje korisnika sa unetim karakteristikama
4. Aplikacija izbacuje sve proizvode koji odgovaraju traženim kategorijama
5. Korisnik poziva aplikaciju da izvrši sortiranje po ceni ili nazivu
6. Aplikacija izvršava sortiranje

Alternativni scenario:

1. Aplikacija ne pronalazi proizvod sa unetim karateristikama.

**UseCase5: Kupovina**

Naziv: Kupovina

Aktor: Korisnik

Učesnici: Aplikacija i korisnik

Preduslov: Aplikacija je pokrenuta, korisnik je ulogovan i nalazi se na stranici sa proizvodima.

Osnovni scenario:

1. Korisnik bira proizvode i količinu proizvoda koje zeli da kupi
2. Aplikacija prebacuje izabrane proizvode u korpu
3. Korisnik poziva aplikaciju da napravi narudzbenicu
4. Aplikacija kreira narudzbenicu tu txt fajlu i ubacuje je u bazu

Alternativni scenario:

1. Aplikacija ne uspeva da kreira narudzbenicu i prikazuje poruku o grešci

# **Faza analize – korisnik**

**DSSK1: Registracija**

1. Aplikacija prikazuje formu za registraciju
2. Korisinik poziva sistem da proveri postojanje korisnika sa zadatim korisničkim imenom
3. Korisnik poziva aplikaciju da kreira korisnika
4. Aplikacija prikazuje korisniku poruku o uspešnoj registraciji



Slika 2. Dijagram sekvenci za DSSK1-Registracija

Alternativni scenario:



Slika 3. Dijagram sekvenci za DSSK1-korisnik postoji

Uvedene sistemske operacije:

1. TraziKorisnika(username)
2. UpisiNovogKorisnika(ime,prezime,email,username,password)

**DSSK2: Logovanje**

1. Korisnik poziva aplikaciju da proveri da li su korisničko ime i šifra validni
2. Aplikacija prikazuje poruku o uspešnom logovanju i otpočinje sesiju



Slika 4. Dijagram sekvenci za DSSK2 – Logovanje

Alternativni scenario:



Slika 5. Dijagram sekvenci za DSSK2 – Netačno korisničko ime/lozinka

Uvedene sistemske operacije:

1. Logovanje(username,password)

**DSSK3: Slanje email-a**

1. Korisnik poziva aplikaciju da pošalje mail
2. Aplikacija prikazuje poruku o uspešno poslatom mail-u



Slika 6. Dijagram sekvenci za DSSK3 – Slanje email-a

Alternativni scenario:



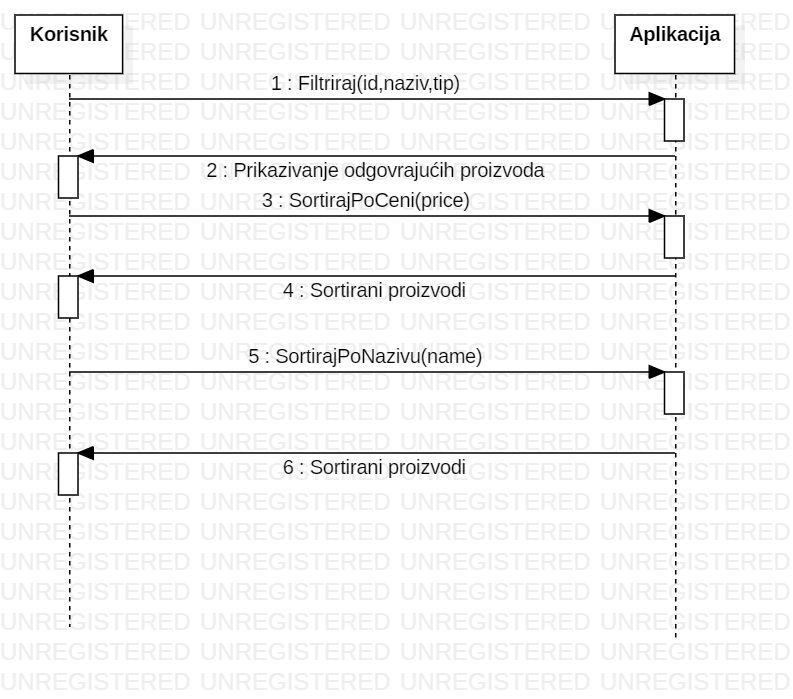
Slika 7. Dijagram sekvenci za DSSK3 – Neuspešno slanje mail-a

Uvedene sistemske operacije:

1. PošaljiMail(imePrezime,email,subject,message)

**DSSK4: Pretrazivanje proizvoda**

1. Korisnik poziva aplikaciju da filtrira proizvode po id-u, nazivu ili tipu
2. Aplikacija prikzuje proizvode koji odgovaraju filteru
3. Korisnik poziva aplikaciju da sortira proizvode po ceni ili nazivu
4. Aplikacija prikazuje sortirane proizvode



Slika 8. Dijagram sekvenci za DSSK4 – Pretraživanje proizvoda

**Alternativni scenario:**



Slika 9. Dijagram sekvenci za DSSK4 – Ne postoji proizvod

Uvedene sistemske opercije:

1. Filtriraj(id,naziv,tip)
2. SortirajPoCeni(price)
3. SortirajPoNazivu(nema)

**DSSK5: Kupovina**

1. Korisnik dodaje proizvod u korpu
2. Aplikacija prikazuje korpu sa odabranim prozvodima, cenama u ukupnom cenom
3. Korisnik poziva aplikaciju da kreira naružbenicu
4. Aplikacija kreira narudzbenicu i ubacuje je u bazu



Slika 10. Dijagram sekvenci za DSSK5 – Kupovina

Alternativni scenario:



Slika 11. Dijagram sekvenci za DSSK5 – Proizvod je već u korpi

Uvedene sistemske operacije:

1. DodajUKorpu(id)
2. KreirajNarudzbenicu(id,username,price,adress,file)

# Faza projektovanja – korinsik

**UG1: TraziKorisnika(username)**

Operacija: TraziKorisnika(username): bool

Preduslov: Korisnik ne postoji u bazi

Posuslov: Korinsik nije pronadjen, omogucena registracija



Slika 12. Dijagram sekvenci UG1: TraziKorisnika

**UG2: UpisiNovogKorisnika**

Operacija: UpisiNovogKorisnika (ime,prezime,email,username,password):bool

Preduslov: Korisnik nije registrovan

Postuslovi:Korisnik je registrovan u bazi



Slika 13. Dijagram sekvenci UG2: UpisiNovogKorisnika

**UG3: Logovanje**

Operacija : Logovanje(username,password) : bool

Preduslov: Korisnik je registrovan ima napravljen nalog

Postuslovi: Korisnik je ulogovan i sesija sa svim njegovim podacima je otpoceta



Slika 14. Dijagram sekvenci UG3: Logovanje

**UG4: PosaljiMail**

Operacija: PosaljiMail(imePrezime,email,subject,message) : bool

Preduslov: Korisnik je ispunio formu za slanje maila

Postuslovi: Mail je uspesno poslata adminu



Slika 15. Dijagram sekvenci UG: PosaljiMail

**UG5: DodajUKorpu**

Operacija: DodajUKorpu(idProizvoda): bool

Preduslovi: Proizvod koji se dodaje mora postojati u bazi

Postuslovi: Dodati proizvod dodaje se u sesiju korpe



Slika 16. Dijagra sekvenci UG5: DodajUKorpu

# Use Case- Admin



Slika 17. Use Case -Admin

Na osnovu slucajeva korišćenja uočavamo sledeće slučajeve korišćenja:

1. Dodaj proizvod
2. Brisi proizvod
3. Izmeni proizvod
4. Brisi korisnika
5. Izmeni korisnika
6. Dodaj korisnika

## Faza analize – admin

**SK1: Dodaj proizvod**

Naziv: Dodaj deo

Aktor: Admin

Učesnici: Admin i aplikacija

Preduslov: Aplikacija je uključena

Primeri za osnovni scenario:

1. Admin unosi podatke za unos dela(APUSO)
2. Admin poziva aplikaciju da unese deo(APSO)
3. Aplikacija unosi deo(SO)

Alternativni scenario:

1. Korisnik nije uneo sve podatke, prikazuje grešku(IA)

**SK2: Brisi proizvod**

Naziv: Brisi deo

Aktor: Admin

Učesnici: Admin i aplikacija

Preduslov: Aplikacija je uključena

Primeri za osnovni scenario:

1. Admin bira deo koji želi da brise(APUSO)
2. Admin poziva aplikaciju da obriše deo(APSO)
3. Aplikacija briše deo(SO)

Alternativni scenario:

1. Aplikacija ne može da obrise deo, izbacuje grešku(IA)

**SK3: Izmeni proizvod**

Naziv: Izmeni proizvod

Aktor: Admin

Učesnici: Admin i aplikacija

Preduslov: Aplikacija je uključena

Primeri za osnovni scenario:

1. Admin unosi podatke za izmenu proizvoda(APUSO)
2. Admin poziva aplikaciju da izmeni proizvod(APSO)
3. Aplikacija menja podatke o proizvodu(SO)

Alternativni scenario:

1. Aplikacija ne može da obrise proizvod, izbacuje grešku(IA)

2. Admin nije uneo sve podatke za izmenu, aplikacija izbacuje grešku(IA)

**SK4: Brisi korisnika**

Naziv: Brisi korisnika

Aktor: Admin

Učesnici: Admin i aplikacija

Preduslov: Aplikacija je uključena

Primeri za osnovni scenario:

1. Admin bira korisnika koji želi da brise(APUSO)
2. Admin poziva aplikaciju da obriše korisnika (APSO)
3. Aplikacija briše korisnika (SO)

Alternativni scenario:

1. Aplikacija ne može da obrise korisnika, izbacuje grešku(IA)

**SK5: Izmeni korisnika**

Naziv: Izmeni korisnika

Aktor: Admin

Učesnici: Admin i aplikacija

Preduslov: Aplikacija je uključena

Primeri za osnovni scenario:

1. Admin unosi podatke za izmenu korisnika (APUSO)
2. Admin poziva aplikaciju da izmeni korisnika (APSO)
3. Aplikacija menja podatke o delu(SO)

Alternativni scenario:

1. Aplikacija ne može da izmeni deo, izbacuje grešku(IA)

2. Admin nije uneo sve podatke za izmenu, aplikacija izbacuje grešku(IA)

**SK6: Dodaj korisnika**

Naziv: Dodaj korisnika

Aktor: Admin

Učesnici: Admin i aplikacija

Preduslov: Aplikacija je uključena i

Primeri za osnovni scenario:

1. Admin unosi podatke za unos dela(APUSO)
2. Admin poziva aplikaciju da unese deo(APSO)
3. Aplikacija unosi deo(SO)

Alternativni scenario:

1. Korisnik nije uneo sve podatke, prikazuje grešku(IA)

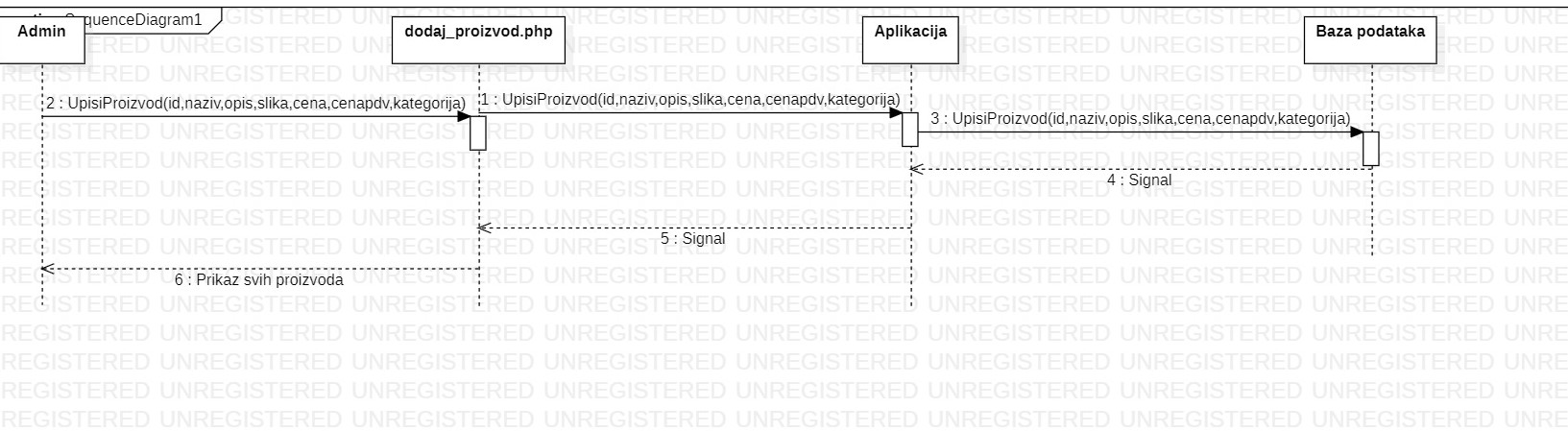
## Faza projektovanja -admin

**UG1: UpisiProizvod**

Operacija: UpisiProizvod (id,naziv,opis,slika,cena,kategorija)

Preduslov: Proizvod ne postoji u bazi

Posuslovi: Proizvod je upisan u bazu



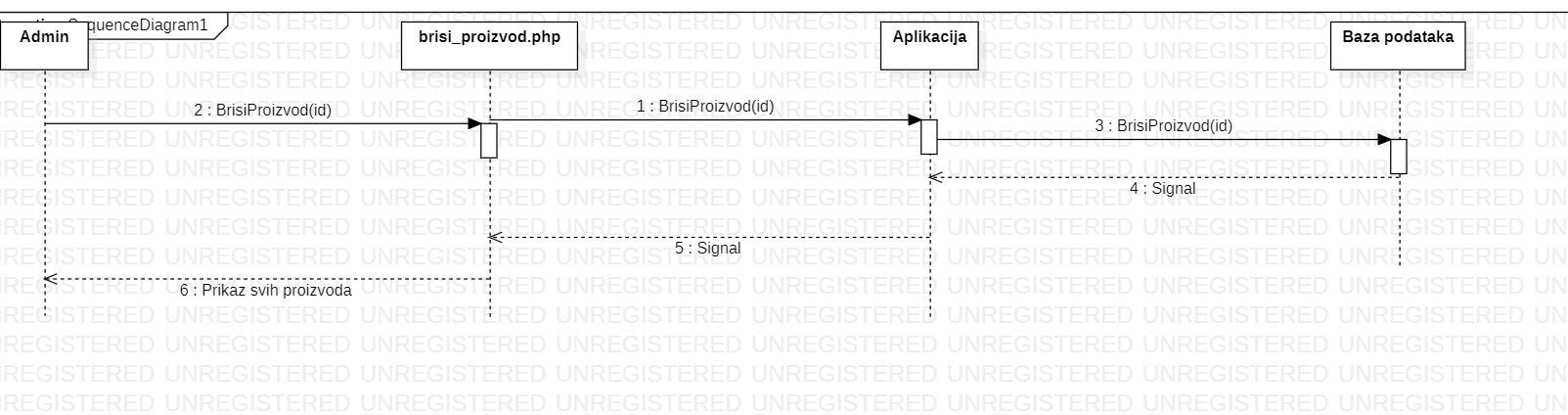
Slika 18. Dijagram sekvence SK1: Dodaj proizvod

**UG2: BrisiProizvod**

Operacija: BrisiProizvod (id) :bool

Preduslov: Proizvod postoji u bazi

Posuslovi: Proizvod je izbrisan iz baze



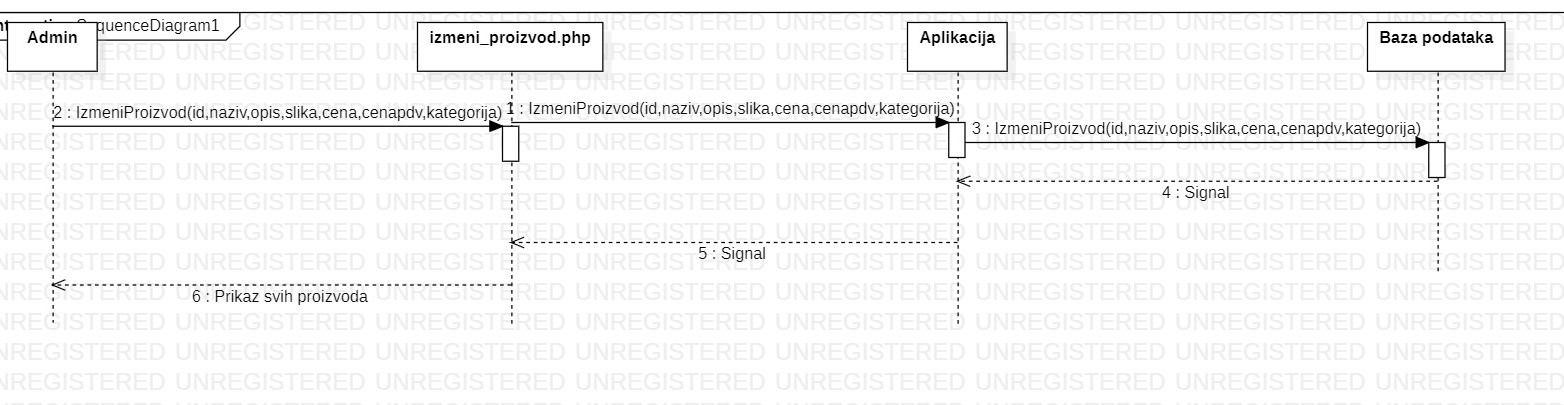
Slika 19. Dijagram sekvence SK2: Brisi proizvod

**UG3: IzmeniProizvod**

Operacija: IzmeniProizvod (id,naziv,opis,slika,cena,cenapdv,kategorija):bool

Preduslov: Proizvod postoji u bazi

Posuslovi: Proizvod je izmenjen iz baze



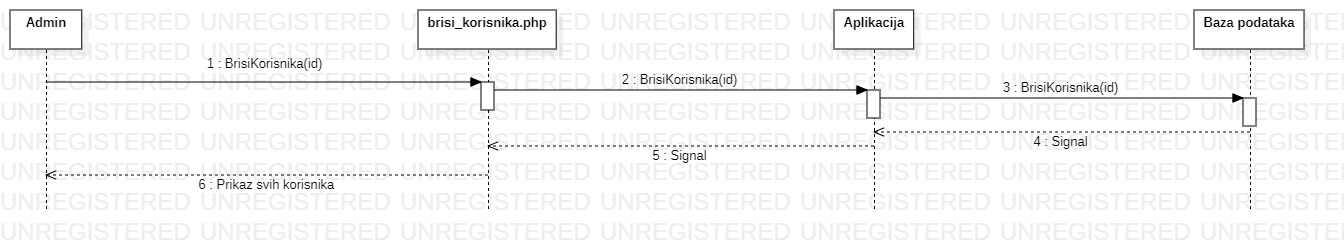
Slika 20. Dijagram sekvenci – Izmeni proizvod

**UG4: BrisiKorisnika**

Operacija: BrisiKorisnika (id) :bool

Preduslov: Korisnik postoji u bazi

Posuslovi: Korisnik je izbrisan iz baze



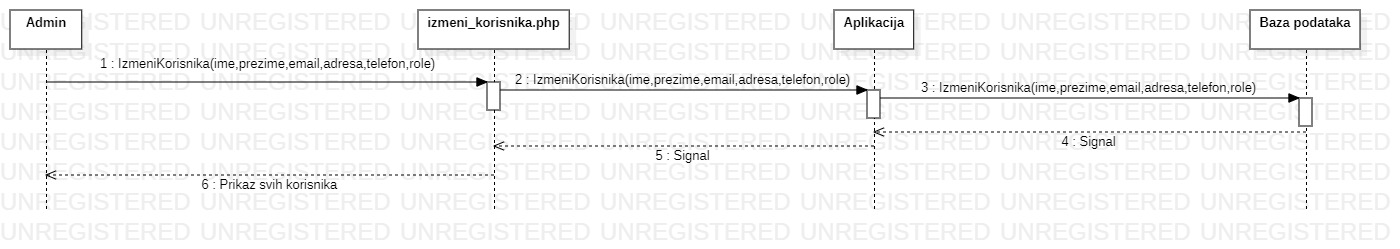
Slika 21. Dijagram sekvenci – Brisi korisnika

**UG5: IzmeniKorisnika**

Operacija: IzmeniKorisnika (ime,naziv,email,telefon,adresa,role) :bool

Preduslov: Korisnik postoji u bazi

Posuslovi: Korisnik je izmenjen iz baze



Slika 22. Dijagram sekvence – Izmeni korisnika