

**VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE  
TEHNOLOGIJE**



**ITS** INFORMATION  
TECHNOLOGY  
SCHOOL

---

VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA IT

Završni rad:

**Projektovanje i realizacija web aplikacije za  
podršku u radu akademije kurseva**

Mentor:

dr Aleksandar Simović

Student:

David Božić 90/19

Beograd,

Februar 2023

## SADRŽAJ:

<b>POGLAVLJA</b>	<b>Str.</b>
REZIME	1
KLJUČNE REČI	1
1. UVOD	2
2. METODA RAZVOJA SOFTVERA	3
2.1. Model vodopad	3
2.2. Spiralni model	4
2.3. Scrum metodologija	5
2.4. Larmanova metodologija	7
3. KORIŠĆENE TEHNOLOGIJE	9
3.1. C#	9
3.2. JavaScript	9
3.3. Microsoft SQL Server	9
3.4. Bootsptrap	10
4. SPECIFIKACIJA KORISNIČKIH ZAHTEVA	11
4.1. Verbalni opis model	11
4.2. Slučajevi korišćenja	12
4.3. Opis slučajeva korišćenja	16
5. FAZA ANALIZE	27
5.1. Sistemski dijagrami sekvence	27
5.2. Definisanje ugovora o sistemskim operacijama	35
5.3. Konceptualni model	41
5.4. Relacioni model	42
6. FAZA PROJEKTOVANJA	43
6.1. Dijagrami sekvenci i kolaboracioni dijagrami za systemske operacije	43
6.2. Projektovanje baze podataka	48
6.3. Projektovanje korisničkog interfejsa	50
7. ZAKLJUČAK	57
8. LITERATURA	58

## REZIME:

U završnom radu projektovana je i implementirana Web aplikacija za podršku u radu akademije kurseva primenom programskog jezika C#-a i Microsoft SQL Server-a. Ovi jezici su primenjeni za manipulisanje bazom podataka, takođe primenjen je i Bootstrap okvir koji uključuje kastomizaciju CSS-a i JS-a za prikaz sadržaja i validaciju korisničkog unosa. Aplikacija nudi svim korisnicima uvid u ponude online kurseva dostupnih u akademiji, a namenjena je: posetiocima, registrovanim korisnicima, moderatorima i administratoru.

Aplikacija je posvećena korisnicima kojima finansije ili obaveze ne dozvoljavaju da pohađaju tradicionalnu skolu. Da bi posetilac mogao da koristi aplikaciju i sve njene benefite, pre svega je potrebno da kreira svoj jedinstveni nalog i da se uloguje. Nakon što je korisnik uspešno uneo svoje podatke, u mogućnosti je da pohađa kurseve koje akademija ima u ponudi. Akademija raspolaže sa raznim kursevima pretežno iz oblasti programiranja i digitalnog marketinga, korisnicima su dostupni video snimci predavanja kao i PDF materijali za utvrđivanje stečenog znanja. Pored toga što klijent mora da ima jedinstveni nalog na koji je ulogovan da bi mogao da pristupi određenom kursu, on takođe mora da unese i informacije o sredstvu plaćanja. Kada su sve potrebne informacije unete, klijent može da se preplati na bilo koji kurs koji akademija ima u ponudi, preplata na kurs je jednokratna i pristup je doživotan, osim ako klijent eksplicitno ne želi da otkaže kurs na koji je preplaćen. Osim samog pristupa kursevima, klijent u svakom trenutku može da pristupi svom jedinstvenom nalogu, da u okviru naloga ima pregled o svojim informacijama i aktivnostima, da te informacije menja, kao i da obriše svoj nalog.

Moderator je zadužen za manipulaciju kursevima na platformi, takođe on može da ima većinu privilegija registrovanog korisnika u cilju provere unetih ili izmenjenih kurseva.

Administrator je osoba sa najvećim pravom pristupa na platformi, on ima sve privilegije moderatora pored njih on takođe može da ima uvid u sve korisnike platforme, da unese moderatora, da ukine pristup bilo kom korisniku u aplikaciji da pregleda komentare korisnika i da ih po potrebi obriše ukoliko se kose sa politikom aplikacije.

Web aplikacije je projektovana u C# programskom jeziku verzije 10.0. Za manipulacije i smeštanje podataka korišćen je Microsoft SQL Server Management Studio verzije 18.12.1. Za stilizaciju frontend-a korišćen je HTML verzije 5 i Bootstrap CSS okvir verzije 5.

## KLJUČNE REČI:

Web aplikacija, akademija kurseva, kurs, C#, Microsoft SQL Server, Bootstrap

## 1. UVOD:

Online usluge i prodavnice su nastale kao posledica razvoja informacionih tehnologija i sve više dolaze do značaja i dokazuju svoju primenu u savremenom dobu. Pojam online usluge, u osnovi, predstavlja lakše obavljanje određenih aktivnosti putem interneta.

U savremenom dobu, veoma je bitno da ustanova koja se bavi obrazovanjem, odnosno stručnim usavršavanjem klijenata poseduje adekvatnu online platformu (aplikaciju). Prednost online obrazovanja je u tome što klijenti na dosta lakši način mogu da svoje obaveze usklade sa nastavom, predavanja su snimljena i pohranjena na samoj platformi tako da im je moguće pristupiti u bilo koje vreme, sa bilo kog mesta preko njihovih tehnoloških uređaja gde postoji pristup internetu. Ovakav način obrazovanja takođe ostavlja priliku i zaposlenima da se dokvalifikuju ili prekvalifikuju jer nema potrebe da izostaju sa posla ili menjaju svoje prebivalište da bi mogli fizički biti prisutni na predavanjima. Takođe, lica sa invaliditetom mogu da pohađaju kurseve iz udobnosti svog doma bez potrebe da bilo gde odlaze. Zbog toga već sada se može reći da je online obrazovanje postalo naša svakodnevica i nešto što će tek imati široku primenu u budućnosti.

Ova aplikacije je napravljena kako bi omogućila svojim korisnicima da u bilo kom momentu sa bilo kog mesta mogu da prate predavanja koja ih zanimaju, jer je ovakav način obrazovanja dosta fleksibilniji i pristupačniji od tradicionalnog.

Glavni cilj u završnom radu je da obuhvati sve što bi Web aplikacija ovog tipa trebalo da poseduje, kao i problematiku kreiranja i implementacije online usluga.

Kompletan razvoj Web aplikacije prikazan je u završnom radu i opisan kroz sledeća poglavlja:

Metoda razvoja softvera baziran je na principu "Larmanove metode" koja definiše načine upravljanja slučajevima korišćenja i objektno-orijentisanu metodu projektovanja softvera.

Poglavljje, korišćene tehnologije, obuhvata kratak opis tehnologija korišćenih za izradu projekta. To su: C# (eng. *C Sharp*), Microsoft SQL Server (eng. *Microsoft Structured Query Language Server*), HTML (eng. *HyperText Markup Language*), CSS (eng. *Cascading Style Sheets*), *JavaScript* i *Bootstrap*.

U poglavlju Specifikacija korisničkih zahteva dat je slikovni prikaz u vidu dijagrama slučajeva korišćenja za svakog aktera pojedinačno. Takođe, dat je spisak svih slučajeva korišćenja i njihov opis.

Faza analize obuhvata prikaze dijagrama sekvenci za pet karakterističnih slučajeva korišćenja. Za svaku sistemsku operaciju kreiraju se ugovori koji opisuju njeno ponašanje. Takođe su dati i konceptualni i relacioni model kreiranja baza podataka.

U fazi projektovanja dati su dijagrami sekvenci i kolaboracioni dijagram za pet karakterističnih slučajeva korišćenja. Predstavljeno je projektovanje baze podataka kroz slike i opise pet karakterističnih tabela. Prikazani su dizajn i funkcionalnosti korisničkog interfejsa kroz slike i opise za pet karakterističnih slučajeva korišćenja.

## 2. METODA RAZVOJA SOFTVERA

Postoji veliki broj metoda pomoću kojih softverski inženjer može da razvije svoj softver a i od ključnog značaja je da odabere i prepozna odgovarajući metod za razvoj svog softvera.

Korišćenjem ovih metoda i izborom alata, omogućava softverskom inženjeru da realizuje softversko rešenje i ostvari što bolji finalni rezultat.

Neki od najkorišćenijih modela razvoja softvera su: Vodopad model, Spiralni model i Scrum metodologija.

### 2.1 MODEL VODOPAD (WATERFALL MODEL)

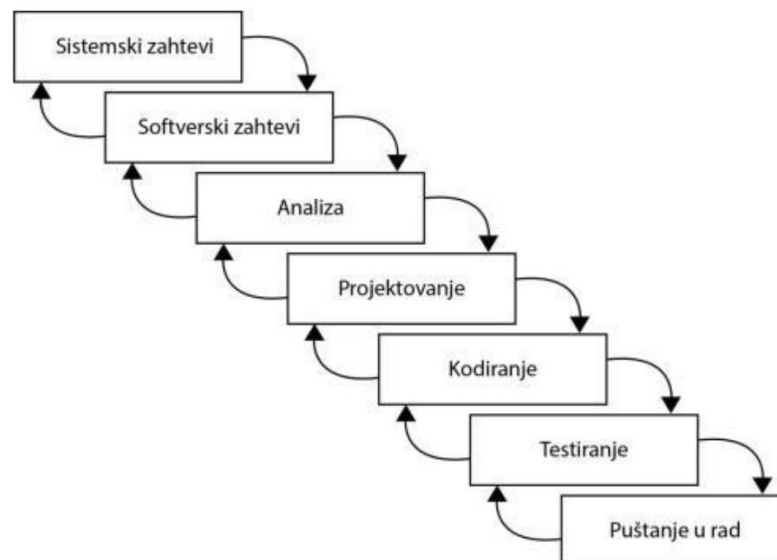
Model vodopada (Slika 2.1.1) je model koji ima pogled na vrlo visokom nivou apstrakcije. Projektnom timu ukazuje na redosledu događaja u procesu razvoja. Da bi se počelo sa narednom fazom, u potpunosti treba da se okonča prethodna faza.

Ono što je karakteristično za model vodopada je što rukovodiocima projekata nudi mogućnost za praćenje napretka projekta, na taj način što se posle svake faze razvoja definiše postignuta dostignuća.

Model vodopada je jedan od najstarijih modela za razvoja softvera, ali se i dalje koristi u širokoj primeni.

Ovaj modela je veoma jednostavan. Doprinosi lakšoj komunikaciji inženjera sa njegovim klijentima. U savremenom dobu, jedan od najvećih izazova predstavlja komunikacija između inženjera i klijenata, koristeći ovaj model razvoja ti problemi se otklanjaju.

Model vodopada ugrubo pokazuje faze razvoja, ne uzimajući u obzir da se softver zapravo gradi kroz mnogo veći broj iteracija. Time se može tvrditi da je ovo najveća mana modela vodopada [1].



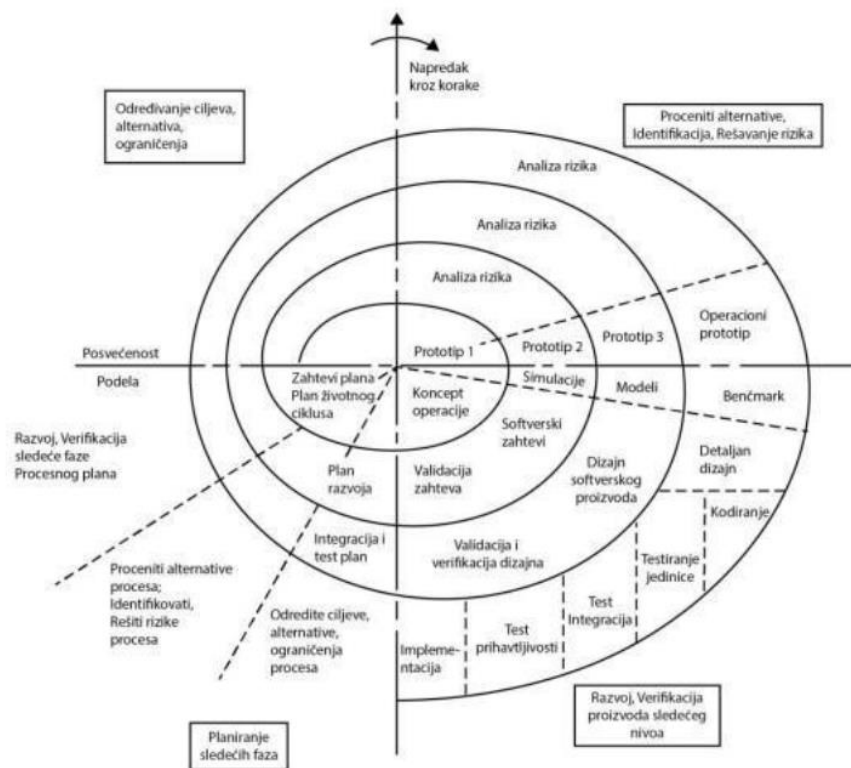
Slika 2.1.1 Modela vodopada [1]

Model vodopada se sastoji iz nekoliko faza koje su kaskadno poređane [1]:

- Sistemski zahtevi
- Softverski zahtevi
- Analiza
- Projektovanje
- Kodiranje
- Testiranje
- Puštanje u rad

## 2.2 SPIRALNI MODEL

Spiralni model (Slika 2.1.2) je model koji je namenjen za razvoj velikih, skupih i kompleksnih projekata.



Slika 2.1.2 Spiralni model [1]

Postoje dve ključne karakteristike ovog modela [1]:

1. Naziv je dobio po cikličnom obliku, podeljen je u četiri kvadranta, prateći razvoj u smeru kazaljke na satu, sa početnom tačkom u centru kruga. Kružnim prolaženjem, prolazi se kroz sve faze razvoja softvera.
2. Druga značajna karakteristika spiralnog modela je sagledavanje i analiza rizika u svakoj iteraciji, što je važna aktivnost u svakom razvoju. Upravo zato se koriste prototipi kao mehanizam za smanjivanje rizika za svaku fazu razvoja.

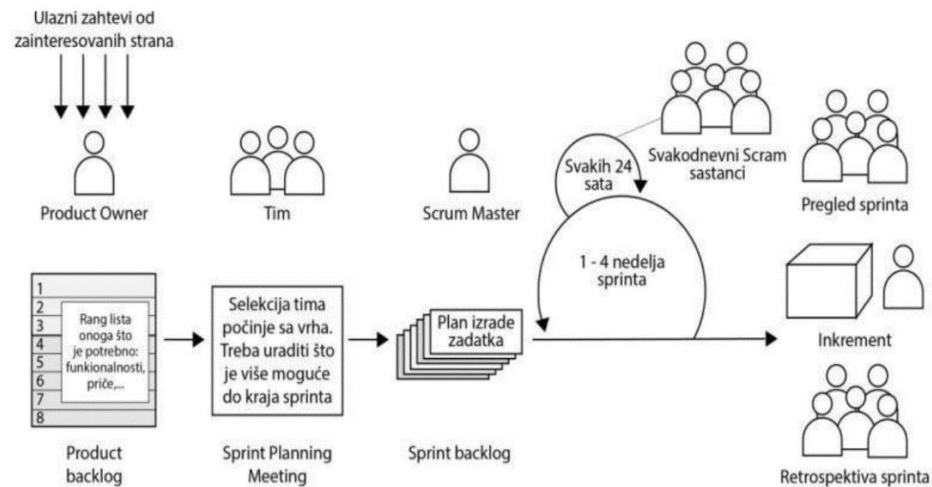
## 2.3 SCRUM METODOLOGIJA

Scrum je metodologija (Slika 2.1.3.) koja služi za razvoj softvera, koristi tehnike iz iterativnog i inkrementalnog razvoja. Svaki petnaest do trideset dana se održava sastanak, koji se naziva "Sprint". Cilj sprinta je da okupi ceo razvojni tim i da zajedno implementiraju zahteve najvišeg nivoa.

Scrum je okvir za organizovanje i upravljanje radom koji je zasnovan na skupu vrednosti, principa i praksi, na koju svaka organizacija dodaje svoju implementaciju, što daje jedinstvenu verziju Scrum-a za tu organizaciju.

Sprint predstavlja period koji traje od 7 do 30 dana u kom se kreira softverski proizvod. Skup funkcija koje bi trebalo realizovati tokom sprint-a koje se određuju u Product Backlog-u, i predstavljaju skup zahteva rangiranih po važnosti.

Sprint planning meeting je sastanak u kojem se određuje koji backlog item-i će ući u sprint. Product Owner je osoba koja informiše tim o item-ima koji bi trebalo da se realizuju tokom sprinta. [1]



Slika 2.1.3 Scrum framework [1]

Karakteristično za Scrum metodologiju su njene uloge. Postoje tri glavne uloge u Scrum metodologiji [1];

- a) **Product owner**  
*Product owner* učestvuje pripremanju *product backlog*-a, planira i donosi ključne odluke i sarađuje sa svim zainteresovanim učesnicima. Mora da postavlja prioritete i vodi računa o profitu organizacije. Kao osoba, *product owner* je vizionar, vidi izgled proizvoda u budućnosti, vodi tim i učesnike ka uspehu projekta.
- b) **Scrum master**  
*Scrum master* je odgovaran za sprovođenju Scrum metodologije u timu, kao i za vođenje tima u razvoj. Članovi tima bi trebalo da gledaju na *scrum master*-a kao mentora ili trenera, neko ko će im pomoći da reše zadate problema sa kojima nisu upoznati i neko ko će ih naučiti nečemu novom. *Scrum master* štiti razvojni tim od spoljnih uticaja i ostalih zainteresovanih učesnika. Otklanja sve prepreke koje se nalaze na putu razvojnog tima i time doprinosi njihovoj produktivnosti. Osoba

koja radi kao *Scrum master* treba da poseduje dovoljno znanja, da bude strpljiv u radu sa timom, da je komunikativan i da bude spreman na saradnju radi uspešnog završetka posla.

- c) **Development team**  
*Development team* je odgovoran za razvoj i isporuku zahteva *product owner*-a. U *development team* su uračunati programeri, administratori baza podataka, dizajneri grafičkog interfejsa, kao i ostali inženjeri koji zajedničkim snagama razvijaju i dizajniraju softversko rešenje. Tim se obično sastoji od pet do devet članova, jer se to u praksi pokazalo kao najbolje rešenje. *Development team* dizajnira, razvija, integriše i testira funkcionalnosti softvera. Članovi tima moraju međusobno efikasno da komuniciraju. Dobra komunikacija je ključ stvaranja kvalitetnog proizvoda.

*Daily scrum* predstavlja sastanak koji traje 15 minuta ili kraće u cilju da svaki od članova tima odgovori na sledeća pitanja [1]:

- a) Šta sam uradio od poslednjeg *Daily scrum*-a?
- b) Na čemu planiram da radim do sledećeg *Daily scrum*-a?
- c) Šta me sprečava u ostvarivanju napretka?

*Sprint review* je razgovor fokusiran na upravo završene karakteristike u kontekstu sveukupnog razvoja, u trajanja od najviše dva sata za 15 dana sprinta. Dešava se nakon završenog *sprinta*.

*Sprint retrospective* je druga aktivnost koja se dešava posle *sprint*-a i predstavlja diskutovanje između *Scrum master*-a sa menadžmentom o tome šta je bilo dobro, a šta ne i na koji način može da se unapredi proces [1].



## 2.4 LARMANOVA METODA

Ova metoda primenjena je za projektovanje Web aplikacije za podršku u radu akademije kurseva. Po uzoru na prethodno navedene metode razvoja sistema, Larmanova metoda poseduje iterativni i inkrementalni model životnog ciklusa softvera, predstavljena je kroz slučajeve korišćenja (eng. *Use Case*) koji definišu interakciju korisnika sa sistemom. Koristi objektno-orijentisanu metodu projektovanja koja uključuje i UML (eng. *Unified Modeling Language*) dijagrame. Objektno-orijentisan razvoj softvera opisuje sistem analogno realnom sistemu u realnom životu i definiše relacije i ponašanje objekata u sistemu kao i u realnosti. UML predstavlja jezik koji povezanim dijagramima na vizuelan način prikazuje konceptualnu (može i konkretnu) strukturu i specifikira sistem. UML nije programski jezik, već alat koji pospešuje razvoj softvera. Larmanova metoda čini enkapsuliranim slučajeve korišćenja od sistemskih operacija. Metoda zahteva upotrebu objektno-orijentisanog načina programiranja po kome se izrađuju paterni (uzori) sa kojima se postiže najoptimalnije rešenje. Larmanova metoda je između ostalog jednostavnija od drugih starijih metoda. Naročito je okrenuta prema programerima jer je prednost data fazi analize i fazi projektovanja. Jedinstveno za Larmanovu metodu je pravljenje ugovora za sistemске operacije što ostale metode ne poseduju. Ukoliko se u fazi analize ne definiše ugovor kasnije, recimo u fazi projektovanja, dolazi do problema u definisanju šta i kako sistemska operacija treba da radi. Jedna od mana Larmanove metode u procesu razvoja softvera je u tome što ne poseduje adekvatan menadžment koji povezuje tim koji učestvuje u razvoju. Nije definisan intenzitet sastanaka niti sama komunikacija između ljudi u timu čime se pospešuje fleksibilnost u izmenama i popravkama u softveru. Sledeće faze čine Larmanovu metodu: specifikacija korisničkih zahteva, analiza zahteva, projektovanje, implementiranje i testiranje softvera [2]:

### *Specifikacija zahteva*

Specifikacija zahteva je faza u kojoj konstantno sarađuju i komuniciraju projektanti i eksperti određenog domena. Zahtevi se prikupljaju, a zatim se opisuju pomoću modela slučajeva korišćenja.

### *Faza analize*

Faza analize je faza u kojoj je ključno da se dobro definišu i objasne slučajevi korišćenja. Faza analize opisuje ponašanje aplikacije. U ovoj fazi kreiraju se: sistemski dijagrami sekvenci, ugovori, konceptualni model i relacioni model.

### *Faza projektovanja*

Opisuje ponašanje aplikacije i fizičku strukturu. Na osnovu već napravljenih dijagrama sekvenci i ugovora, kreiraju dijagrami sekvenci za sve sistemске operacije. Nakon toga se kreira dijagram klasa. Konceptualni dijagram i dijagram sekvenci za sistemске operacije predstavlja osnov za formiranje dijagrama klasa. Nakon definisanih dijagrama sekvenci i kolaboracionih dijagrama za sistemске operacije kreira se baza podataka na osnovu klasa strukture. Kao rezultat se dobijaju tabele relacionog sistema za upravljanje bazom podataka. Na kraju se projektuje korisnički interfejs. Za svaki slučaj korišćenja je izveden ekranski prikaz koji opisuje akcije.

### *Faza implementacije*

U fazi implementacije se piše programski kod u nekom od objektno orijentisanih jezika. Primeri takvih jezika su: PHP, Java, C#.

*Faza testiranja*

Testiranje je važna i nezaobilazna faza u procesu razvoja softvera. Zahtevana pouzdanost i ispravnost koje softver mora da ima rastu srazmerno sa porastom složenosti funkcionalnosti koje on pruža. Od faze testiranja direktno zavisi završetak razvojnog procesa softvera mada nije retkost da se faza implementacije odvija paralelno sa fazom testiranja. Testiranje može uzeti i oko polovinu vremena koje je potrebno za celokupan razvoj softvera te ga vlasnici softvera često izbegavaju jer povećava vreme i poskupljuje razvoj.

### 3. KORIŠĆENE TEHNOLOGIJE

#### 3.1 C#

C# (C Sharp) je moderan, objektno orijentisan programski jezik. C# omogućava programerima da naprave mnoge tipove sigurnih i robusnih aplikacija koje rade u .NET-u. C# ima svoje korene u C porodici jezika i odmah će biti poznat C, C++, Java i JavaScript programerima.

C# obezbeđuje jezičke konstrukcije koje direktno podržavaju koncepte objektno orijentisanog i komponentno orijentisanog programiranja, čineći C# prirodnim jezikom za kreiranje i korišćenje softverskih komponenti.

Od svog nastanka, C# je dodao funkcije koje podržavaju nova radna opterećenja i nove prakse dizajna softvera. U svojoj srži, C# je objektno orijentisani jezik. Vi definišete tipove i njihovo ponašanje. [3].

#### 3.2 JAVASCRIPT

JavaScript je programski jezik koji služi za pravljenje interfejsa koji korisnicima omogućava, bez potrebe za ponovnim učitavanjem, visoku funkcionalnost.

JavaScript nudi visok nivo interaktivnosti, mogućnost rada u lokalu, bogatiji interfejs, trenutne povratne informacije.

Za razliku od programskih jezika koji se izvršavaju na serverskoj strani, JavaScript se izvršava na klijentskoj strani.

Danas se JavaScript smatra jednim od najpopularnijim programskim jezicima, a posebno je poznat po tome što je on podržan od strane svih Web pretraživača koji se danas najviše koriste [4].

#### 3.3 Microsoft SQL Server

Microsoft SQL Server je sistem za upravljanje relacionim bazama podataka (RDBMS) koji podržava širok spektar aplikacija za obradu transakcija, poslovne inteligencije i analitike u korporativnim IT okruženjima. Microsoft SQL Server je jedna od tri vodeće tehnologije baza podataka na tržištu, zajedno sa Oracle Database i IBM-ovim DB2.

Kao i drugi RDBMS softver, Microsoft SQL Server je izgrađen na vrhu SQL -a, standardizovanog programskog jezika koji administratori baza podataka (DBA) i drugi IT profesionalci koriste za upravljanje bazama podataka i ispitivanje podataka koje sadrže. SQL Server je vezan za Transact-SQL (T-SQL), implementaciju SQL-a iz Microsofta koja standardnom jeziku dodaje skup vlasničkih programskih ekstenzija. [5]

### 3.4 BOOTSTRAP

Bootstrap je CSS okvir (eng. *framework*) koji služi za izradu Web stranica i Web aplikacija. Predstavlja kolekciju CSS i HTML pravila i JavaScript ekstenzija koja se koriste za izradu Web stranice.

Od verzije 2.0. potpuno podržava izradu *responsive* (prilagodljivih) Web stranica, što je danas jedna od bitnijih karakteristika koje sajt treba da poseduje iz razloga što veliki broj korisnika pristupa sajtovima preko mobilnih uređaja. Takođe, treba napomenuti da Bootstrap ima *grid* sistem sa četiri nivoa prilagodljivost, što znači da je namenjen za različite veličine ekrana (velike monitore, laptop računare, table uređaje i mobilne telefone) [6].

Bootstrap u sebi sadrži JavaScript funkcije koje nude korisnicima razne mogućnost, poput *Carousel*-a ili *Dropdown* animacija.

Danas je Bootstrap sastavni deo tehnologija koje Web dizajneri koriste za izradu Web stranica, upravo zbog toga što nudi mogućnost lake izrade prilagodljivih sajtova [7].

## 4. SPECIFIKACIJA KORISNIČKIH ZAHTEVA

### 4.1 VERBALNI OPIS MODEL

Potrebno je projektovati i implementirati aplikaciju za podršku u radu akademije kurseva. Web aplikacija sadrži četiri role: Posetilac, Registrovani korisnik, Moderator i Administrator.

Posetioци pristupaju web platformi na kojoj im se izlistava spisak svih kurseva koji postoje u akademiji, pored toga posetioци mogu da pročitaju detalje o samom kursu, kao i komentare korisnika, da se registruju i da se uloguju. Svaki posetilac nakon registracije i logovanja na samoj platformi dobija status registrovanog korisnika.

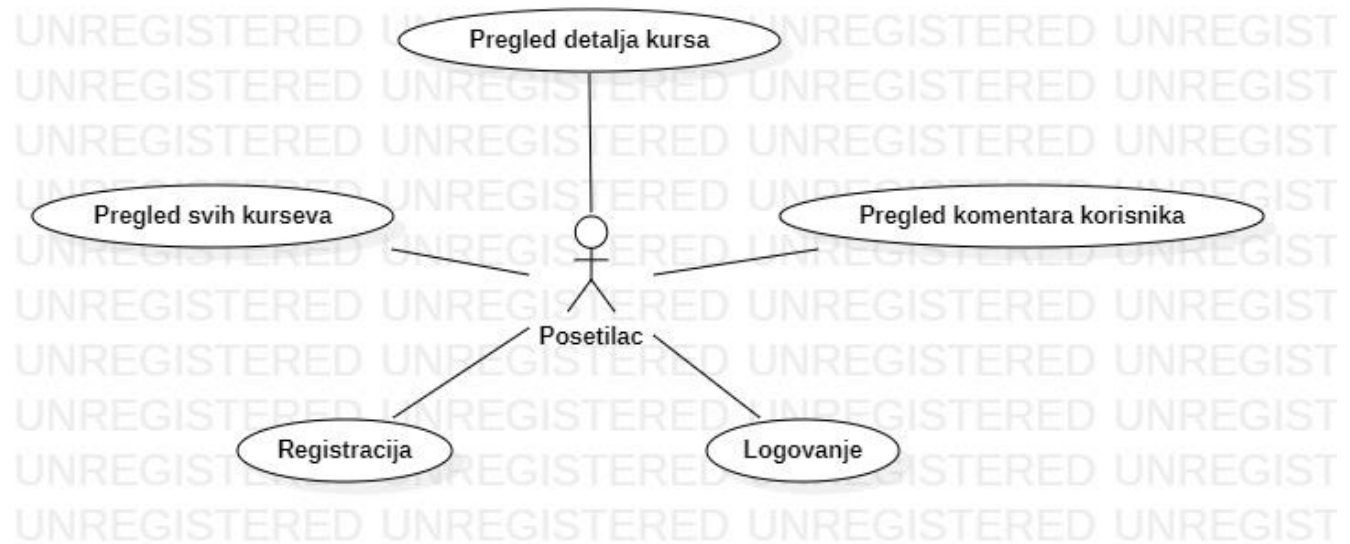
Kao registrovan korisnik neophodno je popuniti podatke o platnoj kartici da bi bilo moguće otključavanje određenog kursa. Registrovan korisnik može da se uloguje, izloguje, da izmeni šifru ukoliko je zaboravio, otključa kurs, pregleda kurs, preuzme materijale sa predavanja, otkaže kurs koji je već pregledao, izmeni svoje podatke koje je uneo prilikom registrovanja, da unese komentar na određeni kurs, obriše komentar, pročita komentare drugih korisnika, kao i da obriše svoj nalog na platformi.

Moderator može da se: uloguje, izloguje, izmeni svoje podatke, da izmeni šifru ukoliko je zaboravio, da doda kurs, pregleda kurs, preuzme materijale sa predavanja, pročita komentare korisnika, izmeni kurs, obriše kurs, kao i da obriše svoj nalog na platformi.

Administrator akademije kurseva može da se: uloguje, izloguje, da izmeni šifru ukoliko je zaboravio, izmeni svoje podatke, pregleda koji sve korisnici postoje u akademiji, da unese moderatora, da ukine pristup korisniku/moderatoru, da doda novi kurs, pregleda kurs, preuzme materijale sa predavanja, pregleda komentare korisnika, obriše komentar korisnika, obriše kurs, kao i da izmeni podatke o određenom kursu.

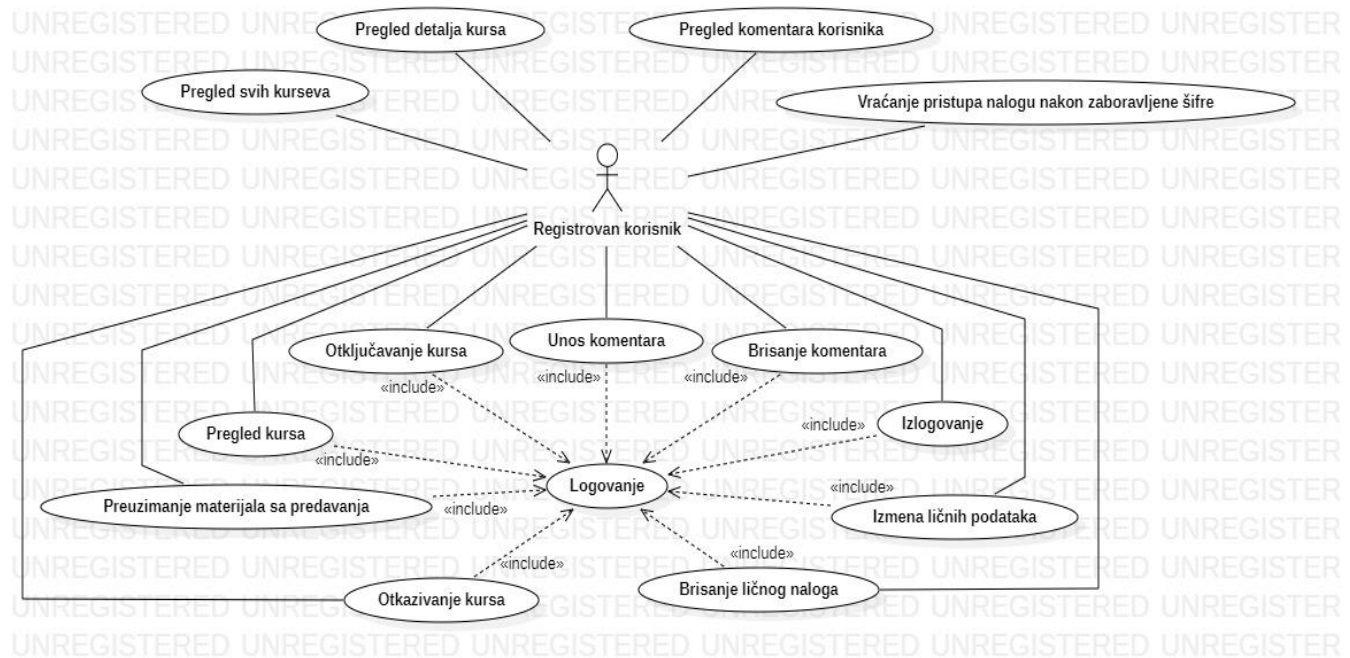
## 4.2 SLUČAJEVI KORIŠĆENJA

Na slici 4.2.1. prikazani su svi slučajevi korišćenja Posetioca



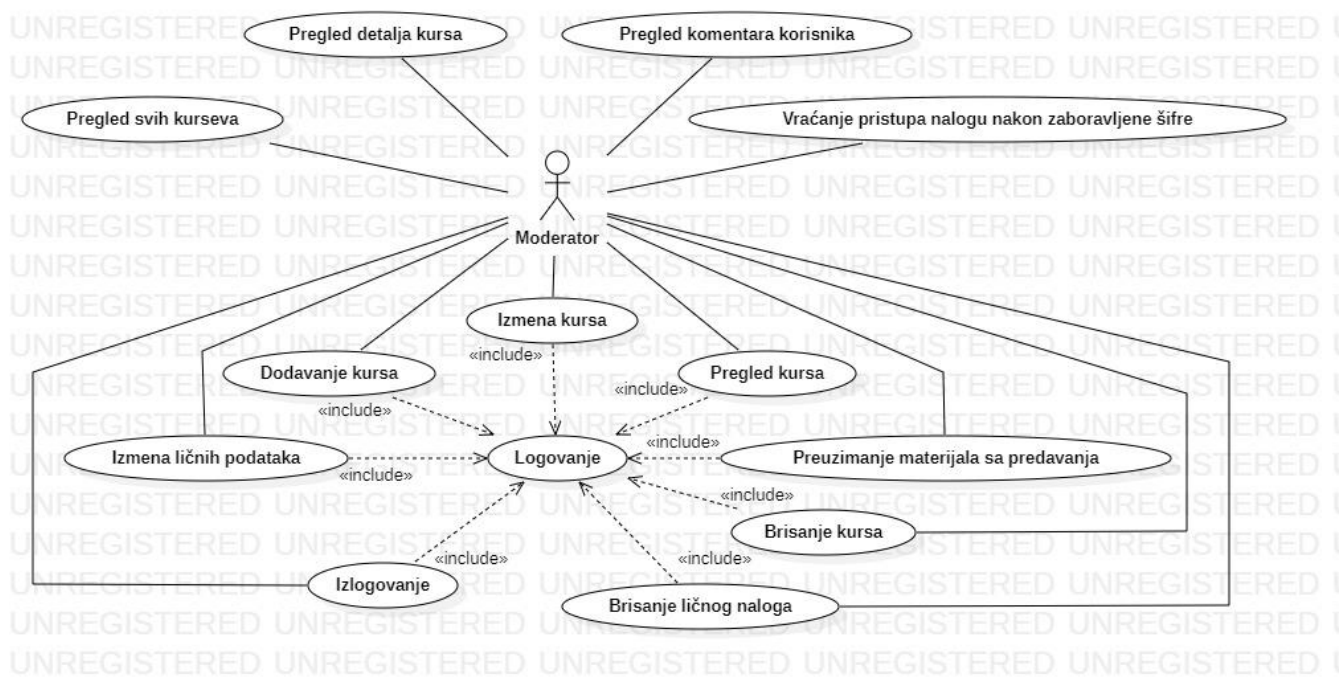
Slika 4.2.1. Dijagram slučajeve korišćenja – Posetilac

Na slici 4.2.2. prikazani su svi slučajevi korišćenja – Registrovanog korisnika



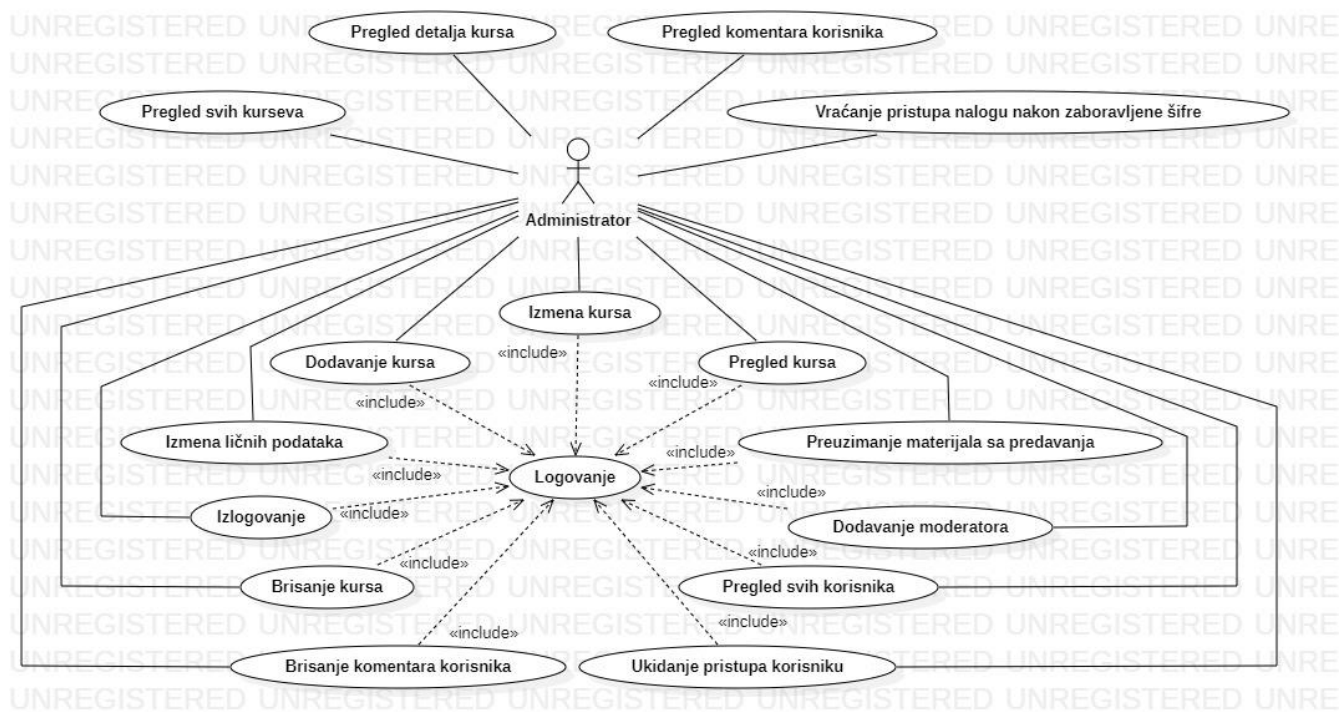
Slika 4.2.2. Dijagram slučajeve korišćenja – Registrovan korisnik

Na slici 4.2.3. prikazani su svi slučajevi korišćenja – Moderatora



Slika 4.2.3. Dijagram slučajeva korišćenja – Moderator

Na slici 4.2.4. prikazani su svi slučajevi korišćenja – Administrator



Slika 4.2.4. Dijagram slučajeva korišćenja – Administrator

Na osnovu verbalnog opisa uočeni su sledeći slučajevi korišćenja:

Slučajevi korišćenja za Posetioca:

1. Pregled svih kurseva
2. Pregled detalja kursa
3. Pregled komentara korisnika
4. Registracija
5. Logovanje

Slučajevi korišćenja za Registrovanog korisnika:

1. Pregled svih kurseva
2. Pregled detalja kursa
3. Pregled komentara korisnika
4. Logovanje
5. Izlogovanje
6. Otključavanje kursa
7. Pregled kursa
8. Preuzimanje materijala sa predavanja
9. Otkazivanje kursa
10. Unos komentara
11. Brisanje komentara
12. Izmena ličnih podataka
13. Brisanje ličnog naloga
14. Vraćanje pristupa nalogu nakon zaboravljene šifre

Slučajevi korišćenja za Moderatora:

1. Pregled svih kurseva
2. Pregled detalja kursa
3. Pregled komentara korisnika
4. Vraćanje pristupa nalogu nakon zaboravljene šifre
5. Logovanje
6. Izlogovanje
7. Dodavanje kursa
8. Izmena kursa
9. Brisanje kursa
10. Pregled kursa
11. Preuzimanje materijala sa predavanja
12. Izmena ličnih podataka
13. Brisanje ličnog naloga



Slučajevi korišćenja za Administratora:

1. Pregled svih kurseva
2. Pregled detalja kursa
3. Pregled komentara korisnika
4. Vraćanje pristupa nalogu nakon zaboravljene šifre
5. Logovanje
6. Izlogovanje
7. Dodavanje kursa
8. Izmena kursa
9. Brisanje kursa
10. Pregled kursa
11. Preuzimanje materijala sa predavanja
12. Izmena ličnih podataka
13. Pregled svih korisnika
14. Dodavanje moderatora
15. Ukidanje pristupa korisniku
16. Brisanje komentara korisnika

#### 4.3 OPIS SLUČAJEVA KORIŠĆENJA

**SK1:** Pregled svih kurseva

**Naziv:** Pregled svih kurseva

**Akteri:** Korisnik (Posetilac, Registrovani korisnik, Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma sa listom svih kurseva

Osnovni scenario:

1. Korisnik **poziva** sistem da prikaže sve kurseve (APSO)
2. Sistem **pretražuje** bazu podataka (SO)
3. Sistem **prikazuje** korisniku kurseve (IA)

Alternativni scenario:

- 2.1. Sistem ne uspeva da pretraži bazu, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

**SK2:** Pregled detalja kursa

**Naziv:** Pregled detalja kursa

**Akteri:** Korisnik (Posetilac, Registrovani korisnik, Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za prikaz detalja kursa

Osnovni scenario:

1. Korisnik **poziva** sistem da prikaže detalje izabranog kursa (APSO)
2. Sistem **pretražuje** bazu podataka (SO)
3. Sistem **prikazuje** korisniku detalje izabranog kursa (IA)

Alternativni scenario:

- 2.1. Sistem ne uspeva da pretraži bazu, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

**SK3:** Pregled komentara korisnika

**Naziv:** Pregled komentara korisnika

**Akteri:** Korisnik (Posetilac, Registrovani korisnik, Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za prikaz detalja kursa

Osnovni scenario:

1. Korisnik **poziva** sistem da prikaže detalje izabranog kursa (APSO)
2. Sistem **pretražuje** bazu podataka (SO)
3. Sistem **prikazuje** korisniku sve komentare vezane za kurs u sklopu detalja kursa (IA)

Alternativni scenario:

- 2.1. Sistem ne uspeva da pretraži bazu, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

**SK4:** Registracija

**Naziv:** Registracija

**Akteri:** Korisnik (Posetilac)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za registraciju

Osnovni scenario:

1. Korisnik **unos** podatke za registraciju (APUSO)
2. Korisnik **poziva** sistem da izvrši proveru podataka za registrovanje (APSO)
3. Sistem **kreira** novog registrovanog korisnika sa unetim podacima (SO)
4. Sistem **prikazuje** formu za logovanje (IA)

Alternativni scenario:

- 3.1. Sistem je pronašao korisnika sa istim korisničkim imenom u bazi, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)
- 3.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

**SK5:** Logovanje

**Naziv:** Logovanje

**Akteri:** Korisnik (Registrovani korisnik, Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za logovanje

Osnovni scenario:

1. Korisnik **unos**i podatke za logovanje (APUSO)
2. Korisnik **poziva** sistem da izvrši proveru podataka za logovanje (APSO)
3. Sistem **pretražuje** bazu podataka (SO)
4. Sistem **prikazuje** formu sa listom svih kurseva (IA)

Alternativni scenario:

- 3.1. Sistem nije pronašao korisnika koji odgovara unetim podacima u bazi, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)
- 3.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

**SK6:** Izlogovanje

**Naziv:** Izlogovanje

**Akteri:** Korisnik (Registrovani korisnik, Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je bilo koja forma u aplikaciji

Osnovni scenario:

1. Korisnik **poziva** sistem da ga izloguje iz aplikacije (APSO)
2. Sistem **izvršava** traženu operaciju (SO)
3. Sistem **prikazuje** formu sa listom svih kurseva (IA)

Alternativni scenario:

- 2.1. Sistem ne uspeva da izvrši traženu operaciju, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

**SK7:** Vraćanje pristupa nakon zaboravljene šifre

**Naziv:** Vraćanje pristupa nakon zaboravljene šifre

**Akteri:** Korisnik (Registrovani korisnik, Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za vraćanje pristupa nalogu

Osnovni scenario:

1. Korisnik **unosí** svoju email adresu (APUSO)
2. Korisnik **poziva** sistem da izvrši slanje mail-a za vraćanje pristupa nalogu (APSO)
3. Sistem **generiše** mail za vraćanje pristupa nalogu (SO)
4. Sistem **šalje** mail za vraćanje pristupa nalogu (IA)
5. Korisnik **pristupa** platformi putem mail-a za vraćanje pristupa nalogu (ANSO)

Alternativni scenario:

- 3.1. Sistem nije generisao mail za vraćanje pristupa nalogu nakon zaboravljene šifre, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)
- 3.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

**SK8:** Otključavanje kursa

**Naziv:** Otključavanje kursa

**Akteri:** Korisnik (Registrovan korisnik)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za otključavanje kursa

Osnovni scenario:

1. Korisnik **unosí** podatke sa platne kartice (APUSO)
2. Korisnik **poziva** sistem da izvrši kreiranje računa (APSO)
3. Sistem **kreira** račun korisnika za određeni kurs (SO)
4. Sistem **prikazuje** formu sa detaljima kursa i sa svim dostupnim materijalima koje dobija korisnik koji se preplatio na kurs (IA)

Alternativni scenario:

- 3.1. Sistem nije uspeo da kreira račun korisnika za određeni kurs, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)
- 3.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

**SK9:** Pregled kursa

**Naziv:** Pregled kursa

**Akteri:** Korisnik (Registrovani korisnik, Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za prikaz detalja kursa

Osnovni scenario:

1. Korisnik **poziva** sistem da prikaže video materijal kursa (APSO)
2. Sistem **pretražuje** bazu podataka (SO)
3. Sistem **prikazuje** formu sa video materijalom kursa (IA)

Alternativni scenario:

- 2.1. Sistem ne uspeva da izvrši traženu operaciju, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

**SK10:** Preuzimanje materijala sa predavanja

**Naziv:** Preuzimanje materijala sa predavanja

**Akteri:** Korisnik (Registrovani korisnik, Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za prikaz detalja kursa

Osnovni scenario:

1. Korisnik **poziva** sistem da preuzme materijale sa predavanja (APSO)
2. Sistem **pretražuje** bazu podataka (SO)
3. Sistem **izvršava** preuzimanje materijala sa predavanja na lokalni računar (IA)

Alternativni scenario:

- 2.1. Sistem ne uspeva da izvrši traženu operaciju, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

**SK11:** Otkazivanje kursa

**Naziv:** Otkazivanje kursa

**Akteri:** Korisnik (Registrovani korisnik, Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za prikaz detalja kursa

Osnovni scenario:

1. Korisnik **poziva** sistem da otkáže kurs na koji je korisnik preplaćen (APSO)
2. Sistem **pretražuje** bazu podataka (SO)
3. Sistem **izvršava** otkazivanje kursa na koji je korisnik preplaćen (IA)

Alternativni scenario:

- 2.1. Sistem ne uspeva da izvrši traženu operaciju, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

**SK12:** Izmena ličnih podataka

**Naziv:** Izmena ličnih podataka

**Akteri:** Korisnik (Registrovani korisnik, Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za izmenu ličnih podataka

Osnovni scenario:

1. Korisnik **unosí** nove lične podatke (APUSO)
2. Korisnik **poziva** sistem da izvrši proveru podataka (APSO)
3. Sistem **evidentira** unete izmene (SO)
4. Sistem **prikazuje** formu za izmenu ličnih podataka (IA)

Alternativni scenario:

- 3.1. Sistem nije uspeo da evidentira unete izmene, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)
- 3.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)
- 3.3. Korisnik nije napravio nikakvu izmenu, sistem mu **prikazuje** odgovarajuću poruku posetiocu i prekida dalje izvršavanje scenarija (IA)

**SK13:** Unos komentara

**Naziv:** Unos komentara

**Akteri:** Korisnik (Registrovani korisnik)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za unos komentara

Osnovni scenario:

1. Korisnik **unos**i komentar na određen kurs (APUSO)
2. Korisnik **poziva** sistem da kreira nov komentar na određen kurs (APSO)
3. Sistem **kreira** nov komentar na određen kurs (SO)
4. Sistem **prikazuje** formu za prikaz detalja kursa (IA)

Alternativni scenario:

- 3.1. Sistem nije uspeo da kreira nov komenar korisnika za određeni kurs, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)
- 3.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

**SK14:** Brisanje komentara

**Naziv:** Brisanje komentara

**Akteri:** Korisnik (Registrovani korisnik i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za prikaz detalja kursa

Osnovni scenario:

1. Korisnik **poziva** sistem da obriše komentar korisnika na određeni kurs (APSO)
2. Sistem **briše** komentar korisnika na određeni kurs (SO)
3. Sistem **prikazuje** formu za prikaz detalja kursa (IA)



Alternativni scenario:

- 2.1. Sistem ne uspeva da izvrši traženu operaciju, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

**SK15:** Brisanje ličnog naloga

**Naziv:** Brisanje ličnog naloga

**Akteri:** Korisnik (Registrovani korisnik i Moderator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za brisanje naloga

Osnovni scenario:

1. Korisnik **unosí** svoju šifru da bi potvrdio brisanje naloga (APUSO)
2. Korisnik **poziva** sistem da izvrši brisanje naloga (APSO)
3. Sistem **briše** nalog korisnika (SO)
4. Sistem **prikazuje** formu za logovanje (IA)

Alternativni scenario:

- 3.1. Sistem nije uspeo da obriše nalog korisnika, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)
- 3.2. Sistem je proverio ispravnost šifre, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

**SK16:** Pregled korisnika platforme

**Naziv:** Pregled korisnika platforme

**Akteri:** Korisnik (Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma „Admin panel“

Osnovni scenario:

1. Korisnik **poziva** sistem prikaže listu korisnika (APSO)
2. Sistem **pretražuje** bazu podataka (SO)
3. Sistem **prikazuje** listu svih korisnika (IA)

Alternativni scenario:

- 2.1. Sistem ne uspeva da izvrši traženu operaciju, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

**SK17:** Unos moderatora

**Naziv:** Unos moderatora

**Akteri:** Korisnik (Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za dodelu uloge korisniku

Osnovni scenario:

1. Korisnik **bira** ulogu za izabranog korisnika (APUSO)
2. Korisnik **poziva** sistem da izvrši dodelu uloge izabranom korisniku (APSO)
3. Sistem **izvršava** dodelu uloge izabranom korisniku (SO)
4. Sistem **prikazuje** formu „Admin panel“ (IA)

Alternativni scenario:

- 3.1. Sistem nije uspeo da dodeli ulogu moderatora izabranom korisniku, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)
- 3.2. Korisnik nije dodelio ni jednu ulogu za izabranog korisnika, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

**SK18:** Ukidanje pristupa korisniku

**Naziv:** Ukidanje pristupa korisniku

**Akteri:** Korisnik (Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za ukidanje pristupa korisniku

Osnovni scenario:

1. Korisnik **poziva** sistem da izvrši ukidanje pristupa korisniku (APSO)
2. Sistem **briše** izabranog korisnika (SO)
3. Sistem **prikazuje** formu „Admin panel“ (IA)

Alternativni scenario:

- 2.1. Sistem ne uspeva da izvrši traženu operaciju, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

**SK19:** Dodavanje kursa

**Naziv:** Dodavanje kursa

**Akteri:** Korisnik (Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za dodavanje kursa

Osnovni scenario:

1. Korisnik **unosí** podatke o novom kursu (APUSO)
2. Korisnik **poziva** sistem da izvrši kreiranje kursa (APSO)
3. Sistem **kreira** novi kurs sa unetim podacima (SO)
4. Sistem **prikazuje** formu sa listom svih kurseva (IA)

Alternativni scenario:

- 3.1. Sistem nije uspeo da izvrši kreiranje novog kursa, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)
- 3.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

**SK20:** Izmena kursa

**Naziv:** Izmena kursa

**Akteri:** Korisnik (Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za izmenu podataka kursa

Osnovni scenario:

1. Korisnik **unosí** nove podatke o kursu (APUSO)
2. Korisnik **poziva** sistem da evidentira unete izmene podataka (APSO)
3. Sistem **evidentira** unete izmene podataka (SO)
4. Sistem **prikazuje** formu sa listom svih kurseva (IA)

Alternativni scenario:

- 3.1. Sistem nije uspeo da evidentira unete izmene, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)
- 3.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

**SK21:** Brisanje kursa

**Naziv:** Brisanje kursa

**Akteri:** Korisnik (Moderator i Administrator)

**Učesnici:** Korisnik i sistem

**Preduslovi:** Sistem je uključen i prikazana je forma za brisanje kursa

Osnovni scenario:

1. Korisnik **poziva** sistem da izvrši brisanje kursa (APSO)
2. Sistem **briše** izabran kurs (SO)
3. Sistem **prikazuje** formu sa listom svih kurseva (IA)

Alternativni scenario:

- 2.1. Sistem ne uspeva da izvrši traženu operaciju, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

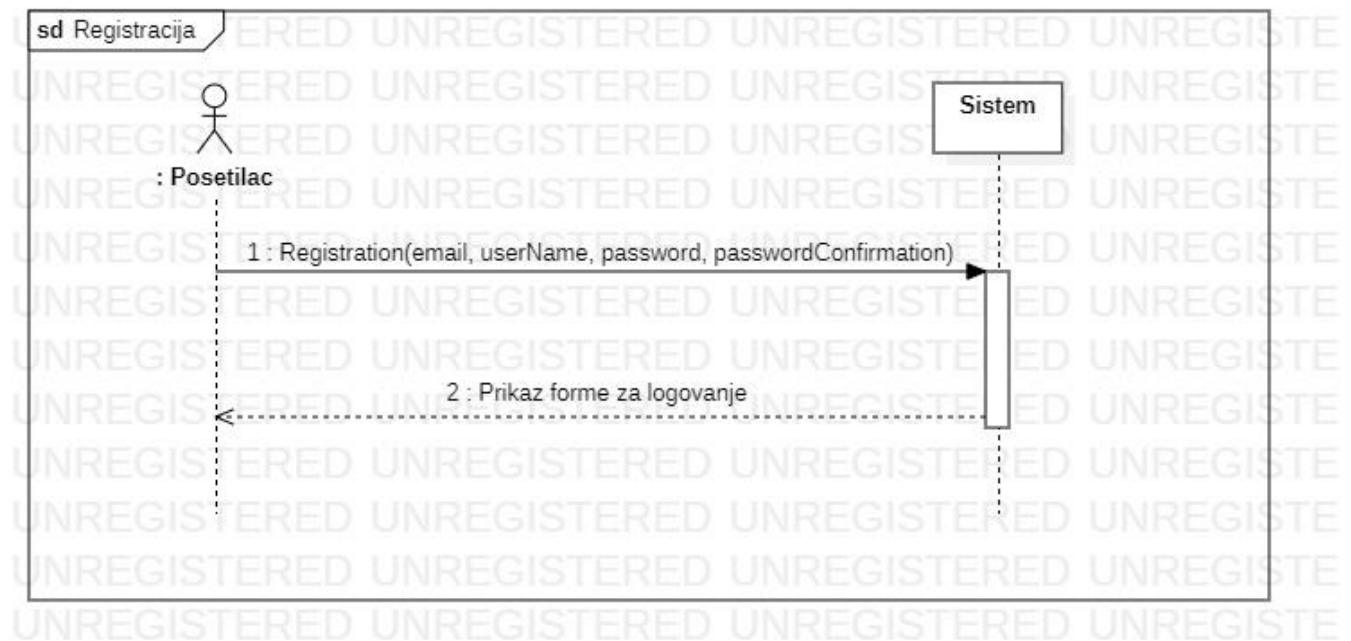
## 5. FAZA ANALIZE

### 5.1. Sistemski dijagrami sekvence

#### **DSSK4:** Registracija

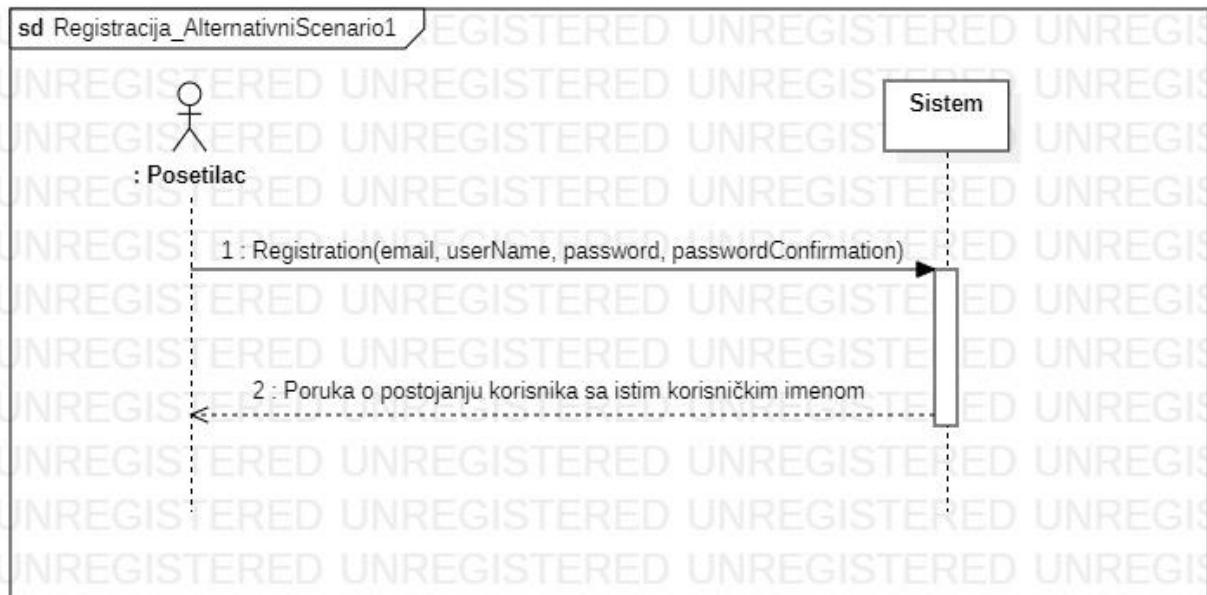
##### Osnovni scenario:

1. Korisnik **poziva** sistem da izvrši proveru podataka za registrovanje (APSO)
2. Sistem **prikazuje** formu za logovanje (IA)

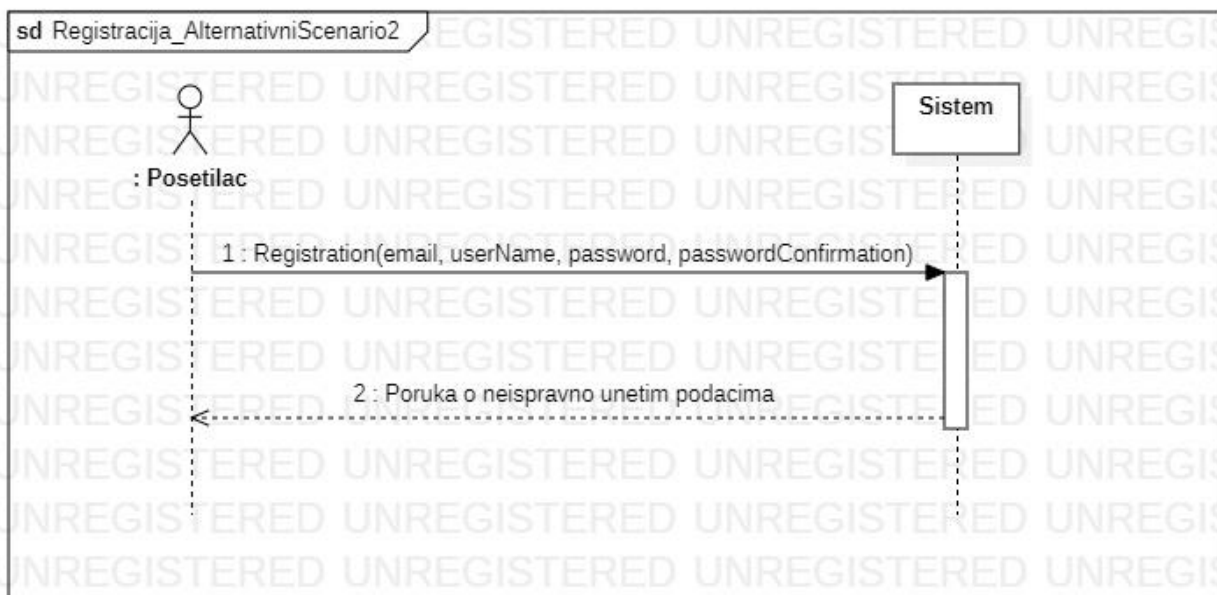


Alternativni scenario:

- 1.1. Sistem je pronašao korisnika sa istim korisničkim imenom u bazi, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

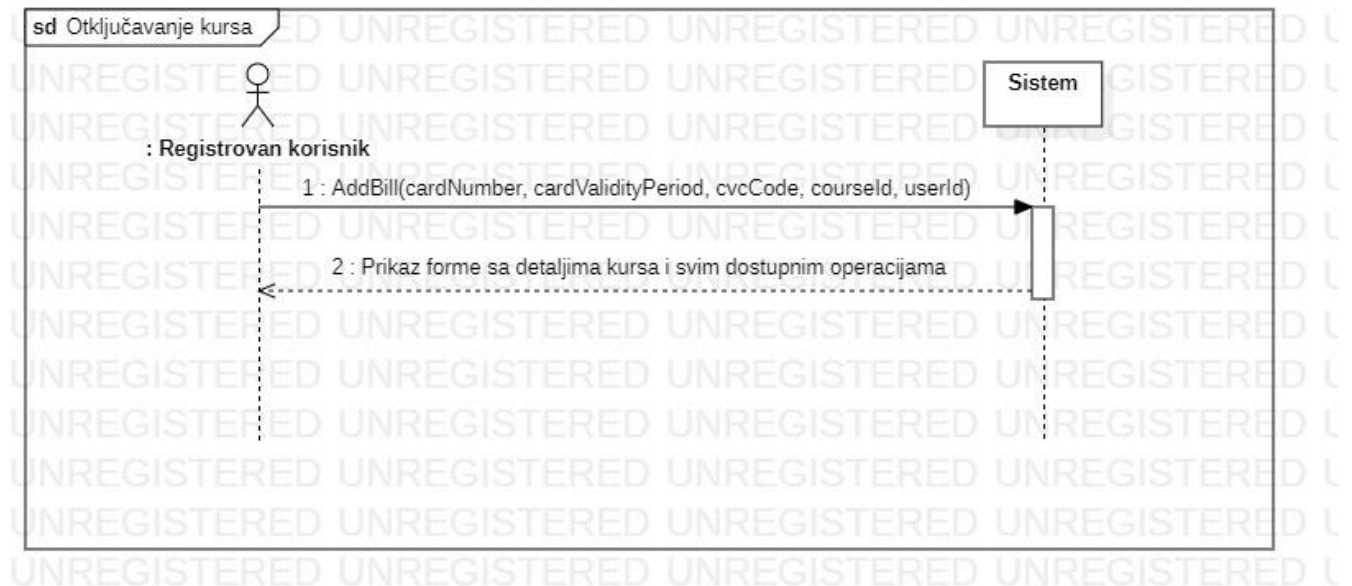


- 1.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

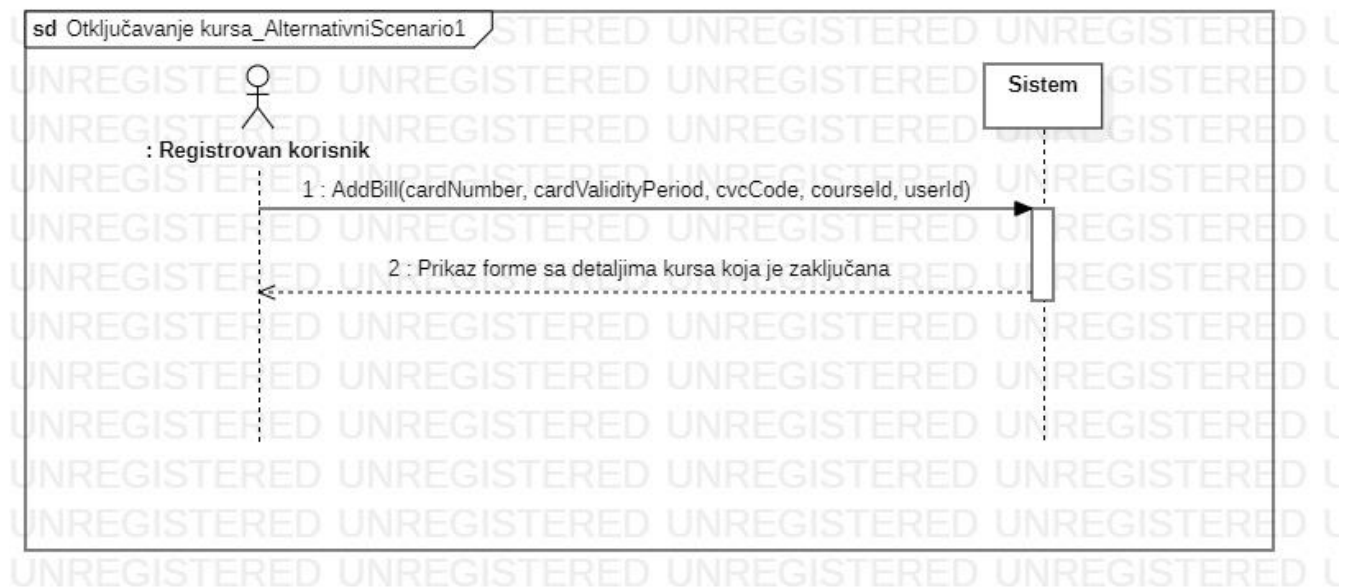


**DSSK8:** Otključavanje kursaOsnovni scenario:

1. Korisnik **poziva** sistem da izvrši kreiranje računa (APSO)
2. Sistem **prikazuje** formu sa detaljima kursa i sa svim dostupnim materijalima koje dobija korisnik koji se preplatio na kurs (IA)

Alternativni scenario:

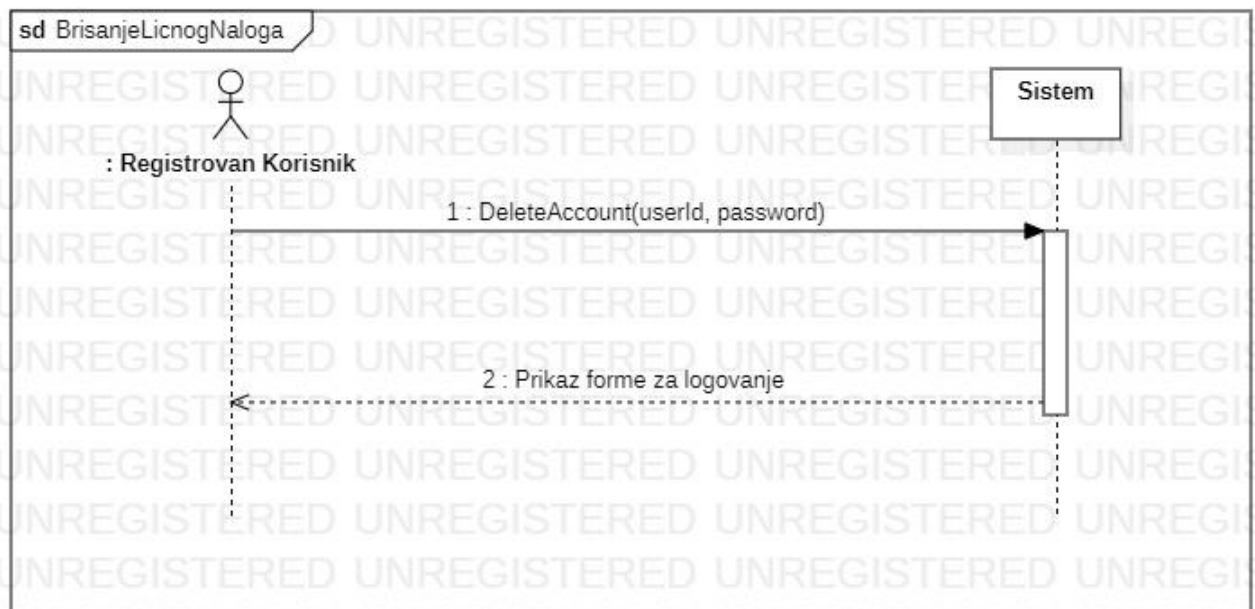
- 1.1. Sistem nije uspeo da kreira račun korisnika za određeni kurs, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)



- 1.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)

**DSSK15:** Brisanje ličnog nalogaOsnovni scenario:

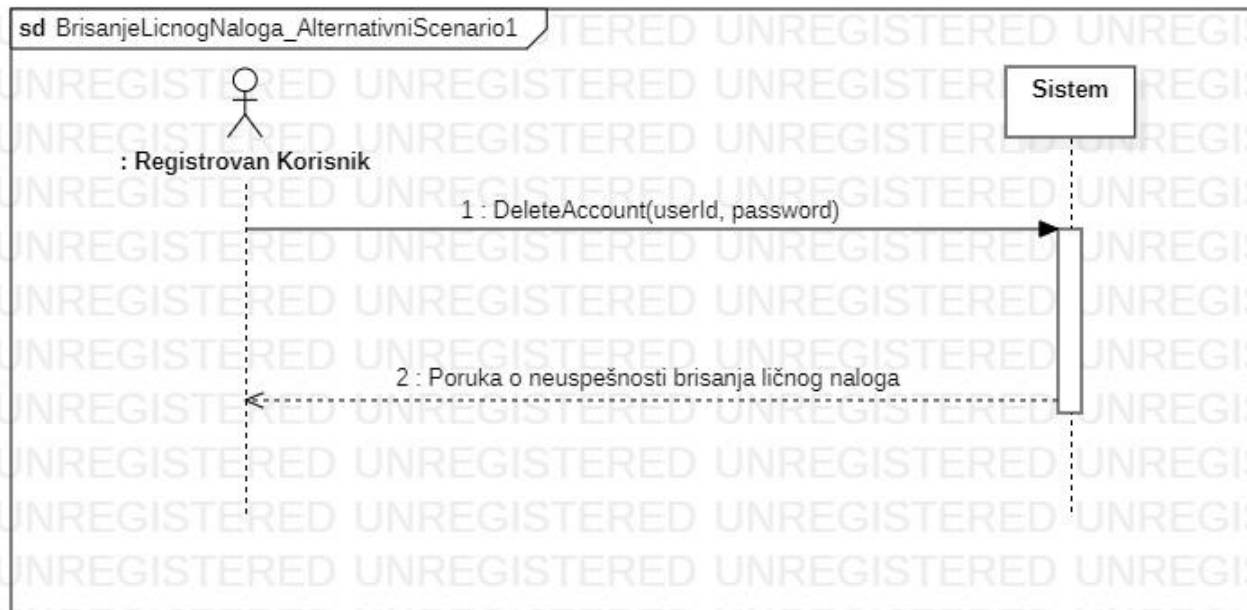
1. Korisnik **poziva** sistem da izvrši brisanje naloga (APSO)
2. Sistem **prikazuje** formu za logovanje (IA)





Alternativni scenario:

- 1.1. Sistem nije uspeo da obriše nalog korisnika, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)

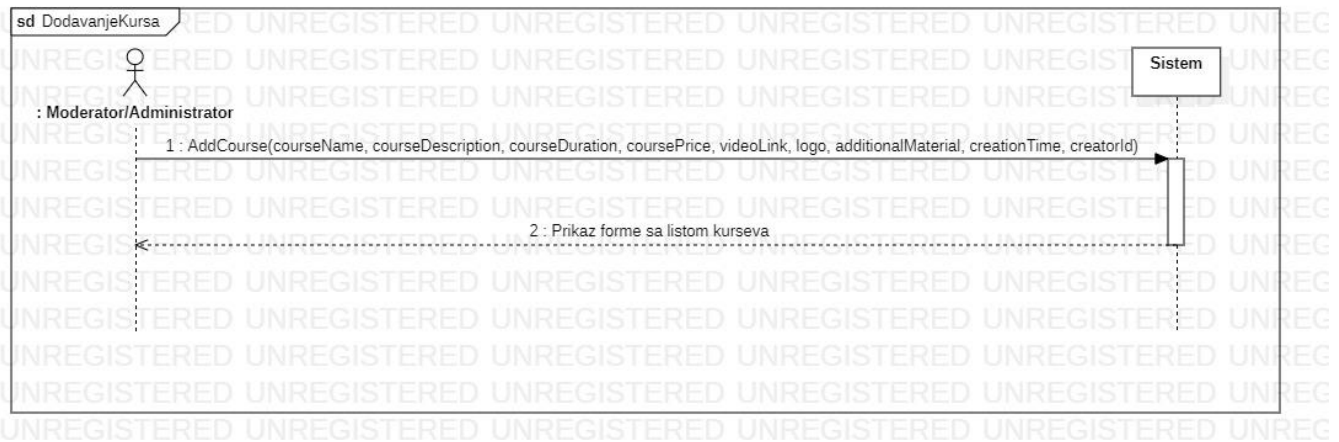


- 1.2. Sistem je proverio ispravnost šifre, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)



**DSSK19:** Dodavanje kursaOsnovni scenario:

1. Korisnik **poziva** sistem da izvrši kreiranje kursa (APSO)
2. Sistem **prikazuje** formu sa listom svih kurseva (IA)

Alternativni scenario:

- 1.1. Sistem nije uspeo da izvrši kreiranje novog kursa, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)



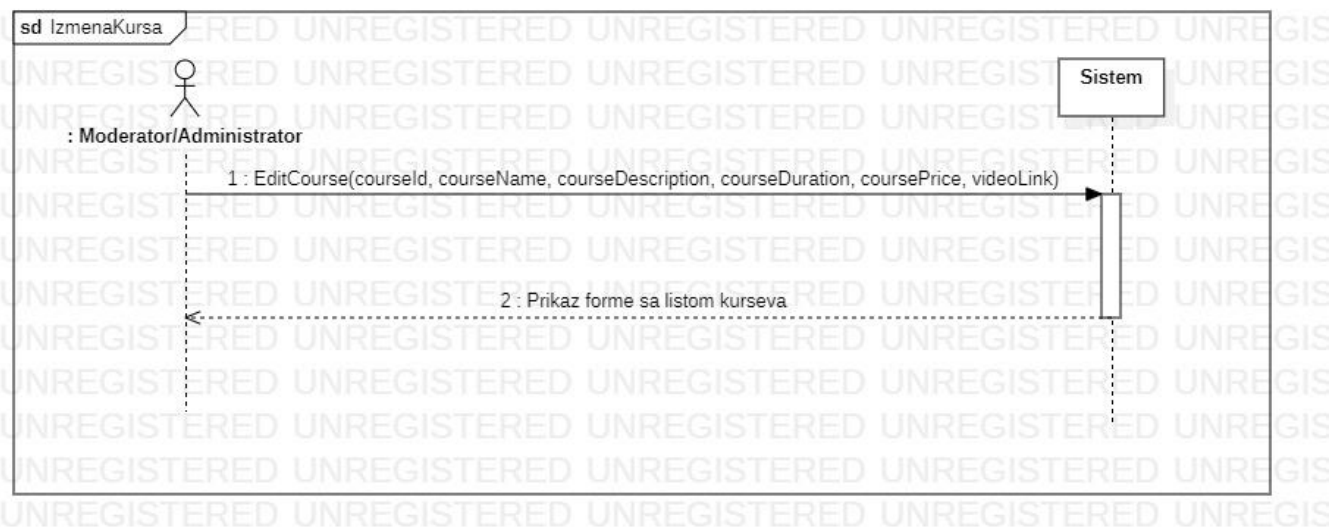
- 1.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)



## DSSK20: Izmena kursa

### Osnovni scenario:

1. Korisnik **poziva** sistem da evidentira unete izmene podataka (APSO)
2. Sistem **prikazuje** formu sa listom svih kurseva (IA)



Alternativni scenario:

- 1.1. Sistem nije uspeo da evidentira unete izmene, **prikazuje** odgovarajuću poruku i prekida dalje izvršavanje scenarija (IA)



- 1.2. Sistem je proverio ispravnost podataka prema šablonu za validaciju, uočio nepoklapanje, **prikazuje** odgovarajuću poruku i prekida izvršavanje scenarija (IA)



## 5.2. Definisanje ugovora o sistemskim operacijama

Za svaku od sistemskih operacija prave se ugovori (eng. *Contracts*). Ugovori opisuju ponašanje sistemske operacije, tako što opisuju šta operacija radi. Jedan ugovor je za jednu sistemsku operaciju.

1. Pregled svih kurseva
2. Pregled detalja kursa
3. Pregled komentara korisnika
4. Registracija
5. Logovanje
6. Izlogovanje
7. Vraćanje pristupa nakon zaboravljene šifre
8. Otključavanje kursa
9. Pregled kursa
10. Preuzimanje materijala sa predavanja
11. Otkazivanje kursa
12. Izmena ličnih podataka
13. Unos komentara
14. Brisanje komentara
15. Brisanje ličnog naloga
16. Pregled korisnika platforme
17. Unos moderatora
18. Ukidanje pristupa korisniku
19. Dodavanje kursa
20. Izmena kursa
21. Brisanje kursa

### **Ugovor UG1: Pregled svih kurseva**

Operacije: CoursesOverview()

Veza sa SK: SK1

Preduslovi: Baza podataka normalno funkcionira i postoje kursevi u akademiji

Postuslovi: Prikazani su svi kursevi

### **Ugovor UG2: Pregled detalja kursa**

Operacije: CourseDetails(courseId)

Veza sa SK: SK2

Preduslovi: Baza podataka normalno funkcionira i postoje kursevi u akademiji

Postuslovi: Prikazani su detalji kursa

### **Ugovor UG3: Pregled komentara korisnika**

Operacije: CommentsOverview(courseId)

Veza sa SK: SK3

Preduslovi: Baza podataka normalno funkcionira i postoje komentari korisnika na kurs

Postuslovi: Prikazani su komentari korisnika

### **Ugovor UG4: Registracija**

Operacije: Registration(email, userName, password, passwordConfirmation)

Veza sa SK: SK4

Preduslovi: Korisnik sa unetim korisničkim imenom ne postoji u bazi

Postuslovi: Podaci o novom korisniku se upisuju u bazu podataka

### **Ugovor UG5: Logovanje**

Operacije: Login(userName, password)

Veza sa SK: SK5

Preduslovi: Postoji sačuvan korisnički nalog

Postuslovi: Korisnik je ulogovan

### **Ugovor UG6: Izlogovanje**

Operacije: Logout()

Veza sa SK: SK6

Preduslovi: Korisnik je ulogovan

Postuslovi: Korisnik je izlogovan

### **Ugovor UG7: Vraćanje pristupa nakon zaboravljene šifre**

Operacije: ForgotPassword(email)

Veza sa SK: SK7

Preduslovi: Postoji sačuvan korisnički nalog

Postuslovi: Korisnik ima pristup svom nalogu

### **Ugovor UG8: Otključavanje kursa**

Operacije: AddBill(cardNumber, cardValidityPeriod, cvcCode, courseId, userId)

Veza sa SK: SK8

Preduslovi: Korisnik je ulogovan i nije preplaćen na kurs

Postuslovi: Korisniku su dostupni svi materijali kursa

### **Ugovor UG9: Pregled kursa**

Operacije: Video(courseId)

Veza sa SK: SK9

Preduslovi: Korisnik je ulogovan i preplaćen na određen kurs

Postuslovi: Korisnik pregleda video materijale kursa

### **Ugovor UG10: Preuzimanje materijala sa predavanja**

Operacije: AdditionalMaterial(courseId)

Veza sa SK: SK10

Preduslovi: Korisnik je ulogovan i preplaćen na određen kurs

Postuslovi: Korisnik je dobio materijale sa predavanja na svoj lokalni računar

### **Ugovor UG11: Otkazivanje kursa**

Operacije: DeleteBill(billId)

Veza sa SK: SK11

Preduslovi: Korisnik je ulogovan i preplaćen na određen kurs

Postuslovi: Korisnik više nema pristup materijalima kursa

### **Ugovor UG12: Izmena ličnih podataka**

Operacije: EditUser()

Veza sa SK: SK12

Preduslovi: Korisnik je ulogovan

Postuslovi: Korisnik je promenio svoje lične podatke

### **Ugovor UG13: Unos komentara**

Operacije: AddComment(courseId, userId)

Veza sa SK: SK13

Preduslovi: Korisnik je ulogovan, preplaćen na kurs i ima ulogu „User“ u sistemu

Postuslovi: Sačuvan je komentar korisnika na određen kurs

### **Ugovor UG14: Brisanje komentara**

Operacije: DeleteComment(commentId)

Veza sa SK: SK14

Preduslovi: Korisnik je ulogovan i ima ulogu „User“ ili „Administrator“ u sistemu

Postuslovi: Komentar korisnika na određeni kurs je obrisao

### **Ugovor UG15: Brisanje ličnog naloga**

Operacije: DeletePersonalData()

Veza sa SK: SK15

Preduslovi: Postoji sačuvan korisnički nalog i korisnik je ulogovan

Postuslovi: Korisnički nalog je obrisao



### **Ugovor UG16: Pregled korisnika platforme**

Operacije: UsersOverview()

Veza sa SK: SK16

Preduslovi: Korisnik je ulogovan i ima ulogu „Administrator“ u sistemu

Postuslovi: Prikazani su svi korisnici sistema

### **Ugovor UG17: Unos moderatora**

Operacije: ManageRole(userId, roleId, roleName)

Veza sa SK: SK17

Preduslovi: Korisnik je ulogovan, ima ulogu „Administrator“ u sistemu i postoji kreiran nalog drugog korisnika kome će biti dodeljena uloga „Moderator“.

Postuslovi: Sačuvan je novi Moderator u sistemu

### **Ugovor UG18: Ukidanje pristupa korisniku**

Operacije: DeleteUser(userId)

Veza sa SK: SK18

Preduslovi: Korisnik je ulogovan i ima ulogu „Administrator“ u sistemu

Postuslovi: Određenom korisniku je obrisani nalog

### **Ugovor UG19: Dodavanje kursa**

Operacije: AddCourse(courseName, courseDescription, courseDuration, coursePrice, videoLink, logo, additionalMaterial, creationTime, creatorId)

Veza sa SK: SK19

Preduslovi: Korisnik je ulogovan i ima ulogu „Administrator“ ili „Moderator“ u sistemu

Postuslovi: Nov kurs je sačuvan u bazi podataka

### **Ugovor UG20: Izmena kursa**

Operacije: EditCourse(courseId, courseName, courseDescription, courseDuration, coursePrice, videoLink)

Veza sa SK: SK20

Preduslovi: Korisnik je ulogovan, ima ulogu „Administrator“ ili „Moderator“ u sistemu i kurs postoji u bazi

Postuslovi: Novi podaci izabranog kursa su sačuvani u bazi

### **Ugovor UG21: Brisanje kursa**

Operacije: DeleteCourse(courseId)

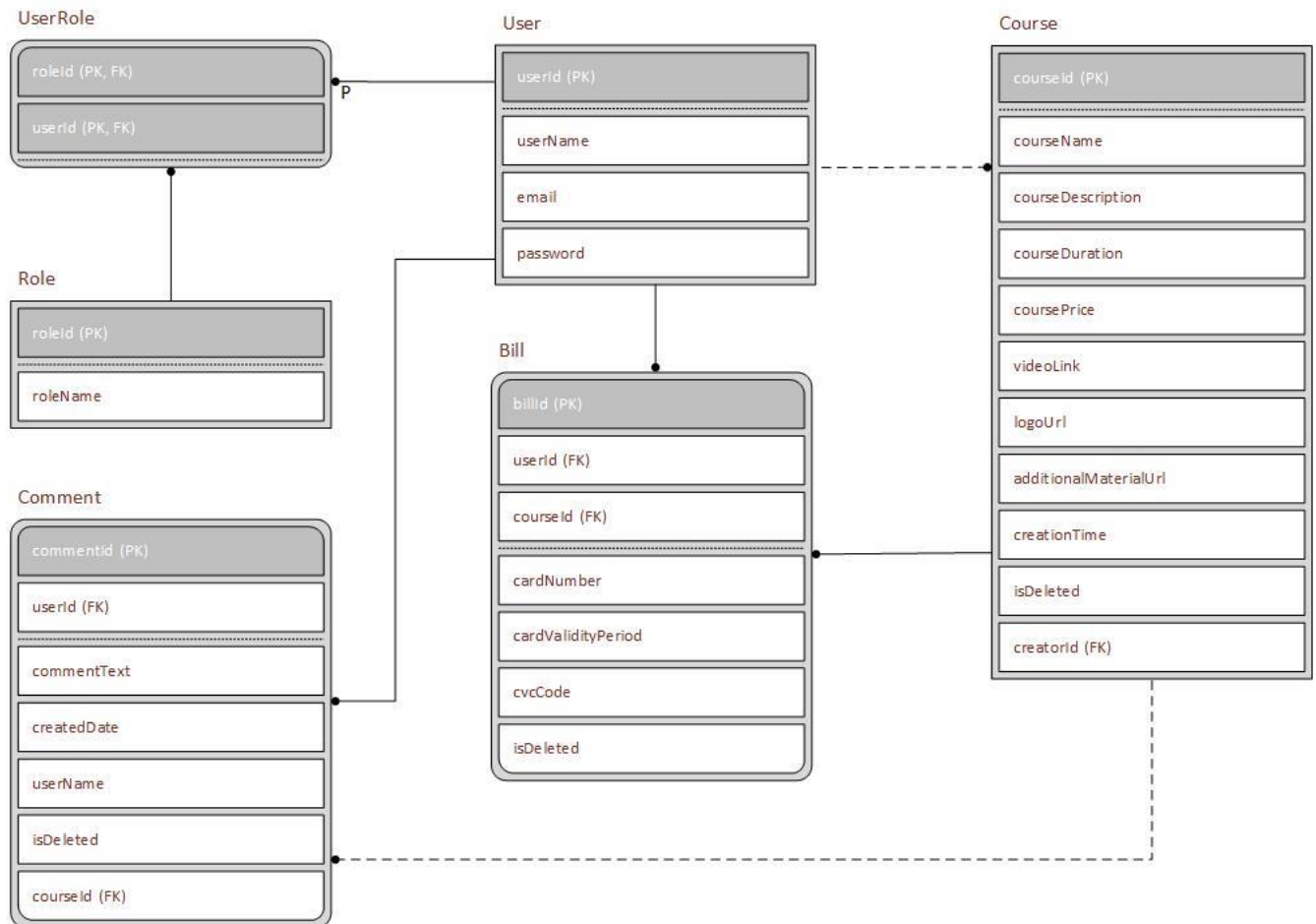
Veza sa SK: SK21

Preduslovi: Korisnik je ulogovan, ima ulogu „Administrator“ ili „Moderator“ u sistemu i kurs postoji u bazi

Postuslovi: Izabrani kurs je obrisani iz baze podataka

### 5.3. Konceptualni model

Konceptualni model nastaje na osnovu podataka iz funkcionalnog zahteva i slučajeva korišćenja (Slika 5.3.1).



Slika 5.3.1. Konceptualni model

#### 5.4. Relacioni model

User (userId, userName, email, password)

Role (roleId, roleName)

UserRole (roleId, userId)

UserRole (roleId) referencira Role (roleId)

UserRole (userId) referencira User (userId)

Course (courseId, creatorId, courseName, courseDescription, courseDuration, coursePrice, videoLink, logoUrl, additionalMaterialUrl, creationTime, isDeleted)

Course (creatorId) referencira User (userId)

Comment (commentId, userId, courseId, commentText, createdAt, userName, isDeleted)

Comment (userId) referencira User (userId)

Comment (courseId) referencira Course (courseId)

Bill (billId, userId, courseId, cardNumber, cardValidityPeriod, cvcCode, isDeleted)

Bill (userId) referencira User (userId)

Bill (courseId) referencira Course (courseId)

## 6. FAZA PROJEKTOVANJA

### 6.1 Dijagrami sekvenci i kolaboracioni dijagrami za sistemske operacije

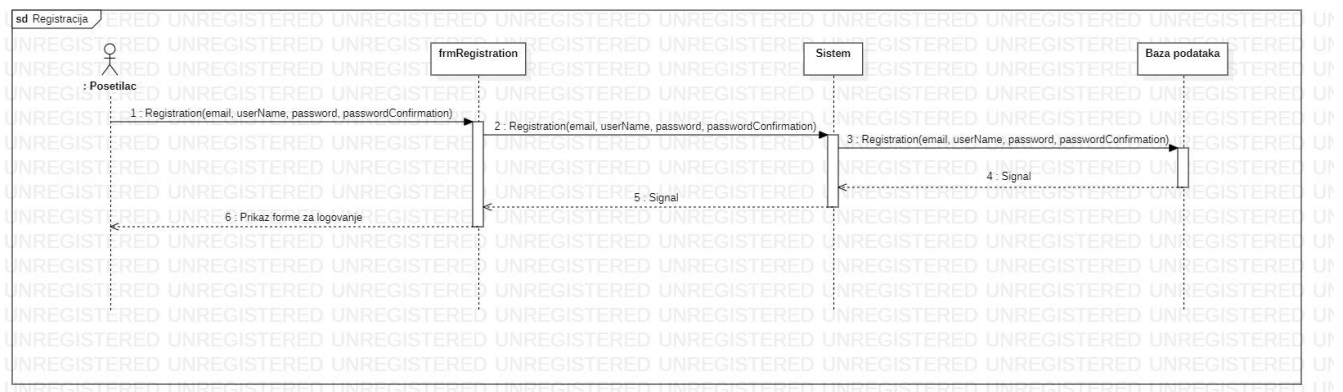
#### Ugovor UG4: Registracija

Operacije: Registration(email, userName, password, passwordConfirmation)

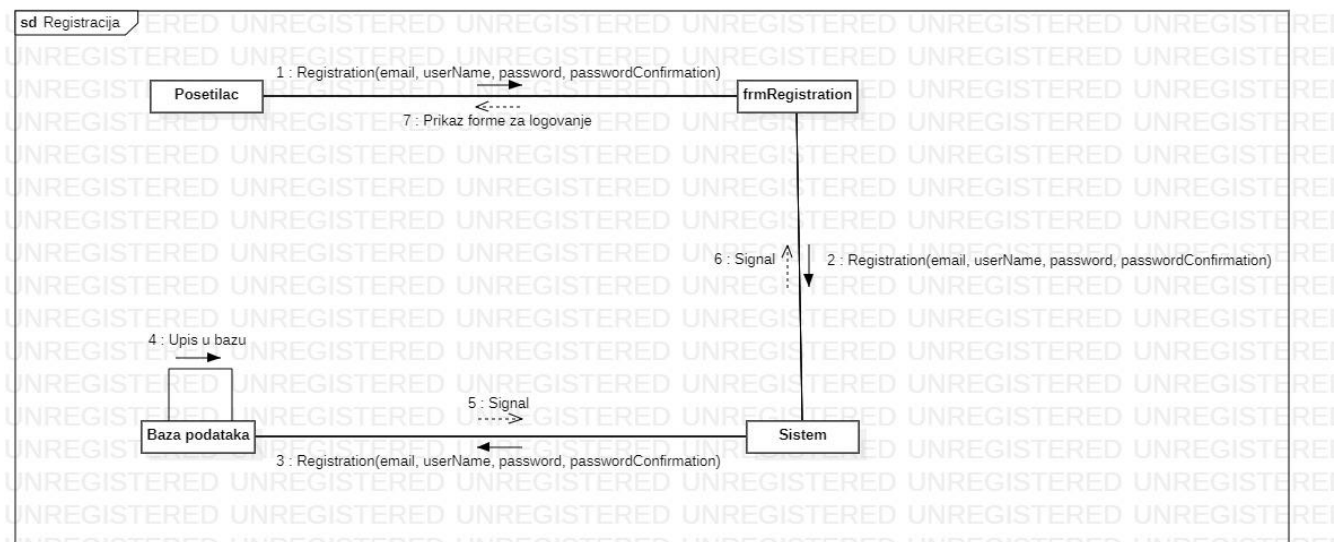
Veza sa SK: SK4

Preduslovi: Korisnik sa unetim korisničkim imenom ne postoji u bazi

Postuslovi: Podaci o novom korisniku se upisuju u bazu podataka



Slika 6.1.1. Dijagram sekvenci UG4 – Registracija



Slika 6.1.2. Dijagram kolaboracije UG4 – Registracija

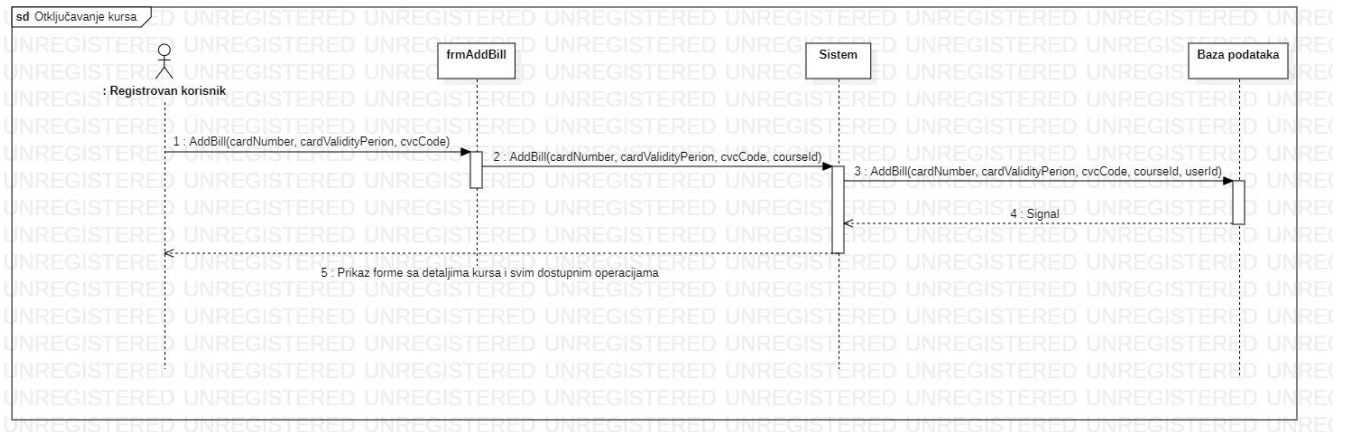
**Ugovor UG8: Otključavanje kursa**

Operacije: AddBill(cardNumber, cardValidityPeriod, cvcCode, courseId, userId)

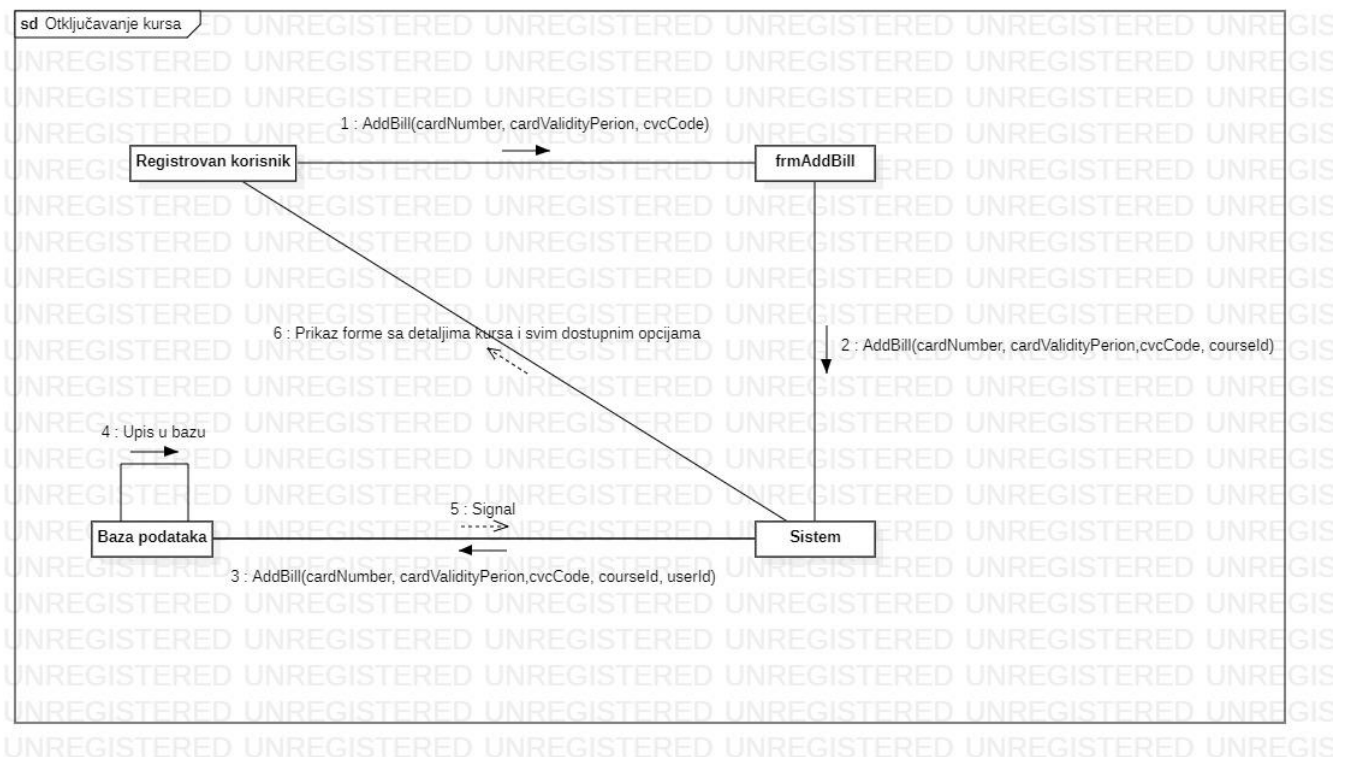
Veza sa SK: SK8

Preduslovi: Korisnik je ulogovan i nije preplaćen na kurs

Postuslovi: Korisniku su dostupni svi materijali kursa



Slika 6.1.3. Dijagram sekveneci UG8 – Otključavanje kursa



Slika 6.1.4. Dijagram kolaboracije UG8 – Otključavanje kursa

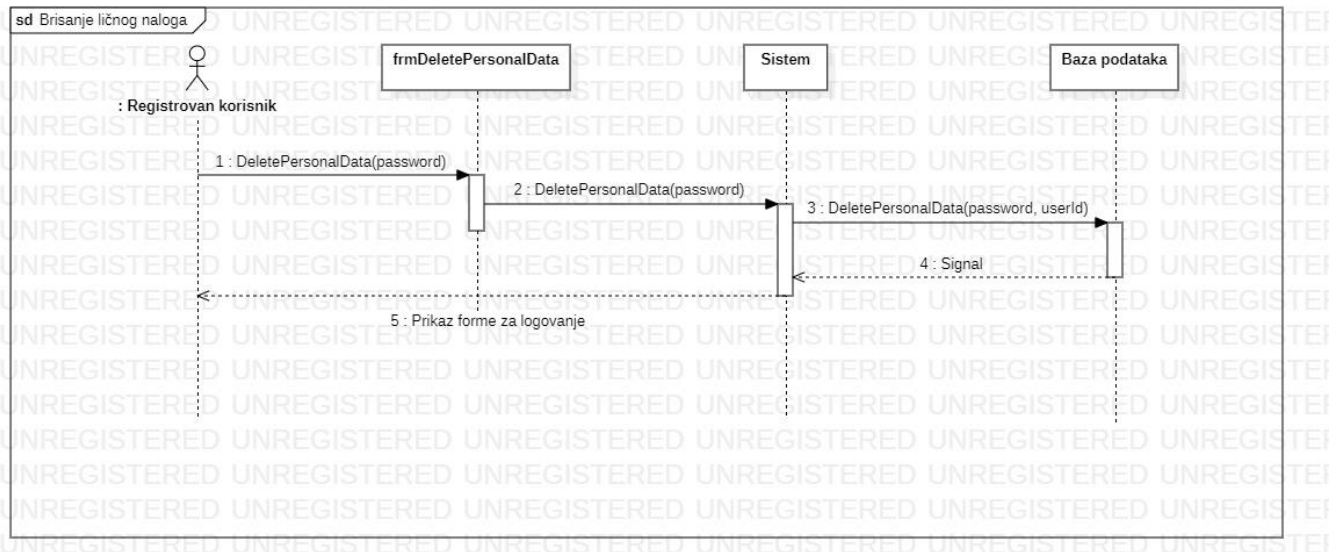
**Ugovor UG15: Brisanje ličnog naloga**

Operacije: DeletePersonalData()

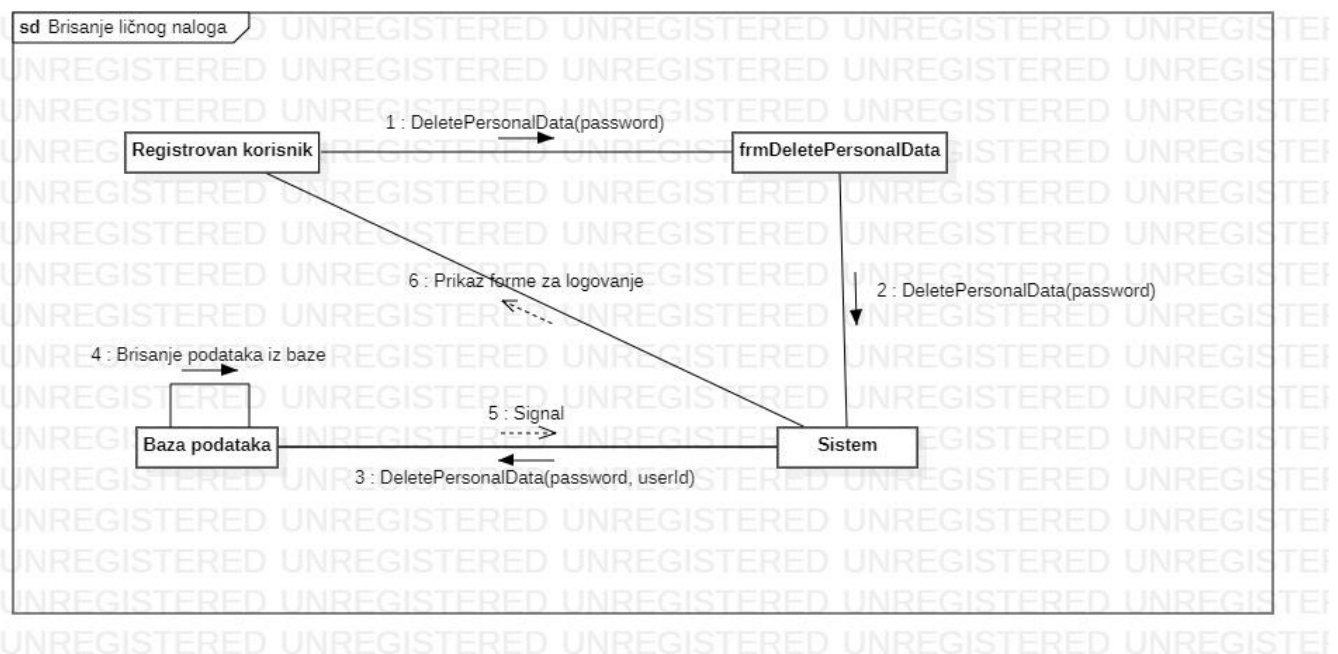
Veza sa SK: SK15

Preduslovi: Postoji sačuvan korisnički nalog i korisnik je ulogovan

Postuslovi: Korisnički nalog je obrisao



Slika 6.1.5. Dijagram sekvenci UG15 – Brisanje ličnog naloga



Slika 6.1.6. Dijagram kolaboracije UG15 – Brisanje ličnog naloga

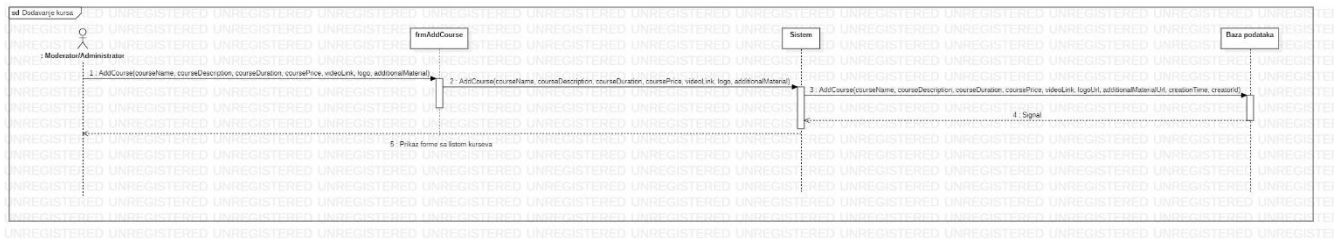
**Ugovor UG19: Dodavanje kursa**

Operacije: AddCourse(courseName, courseDescription, courseDuration, coursePrice, videoLink, logo, additionalMaterial, creationTime, creatorId)

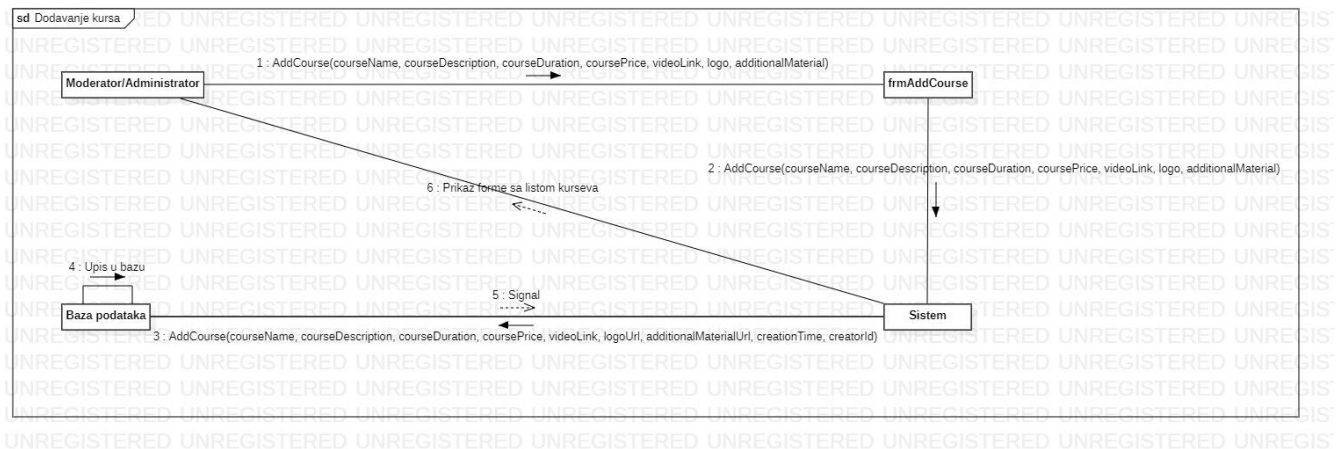
Veza sa SK: SK19

Preduslovi: Korisnik je ulogovan i ima ulogu „Administrator“ ili „Moderator“ u sistemu

Postuslovi: Nov kurs je sačuvan u bazi podataka



Slika 6.1.7. Dijagram sekvenci UG19 – Dodavanje kursa



Slika 6.1.8. Dijagram kolaboracije UG19 – Dodavanje kursa



**Ugovor UG20: Izmena kursa**

Operacije: EditCourse(courseId, courseName, courseDescription, courseDuration, coursePrice, videoLink)

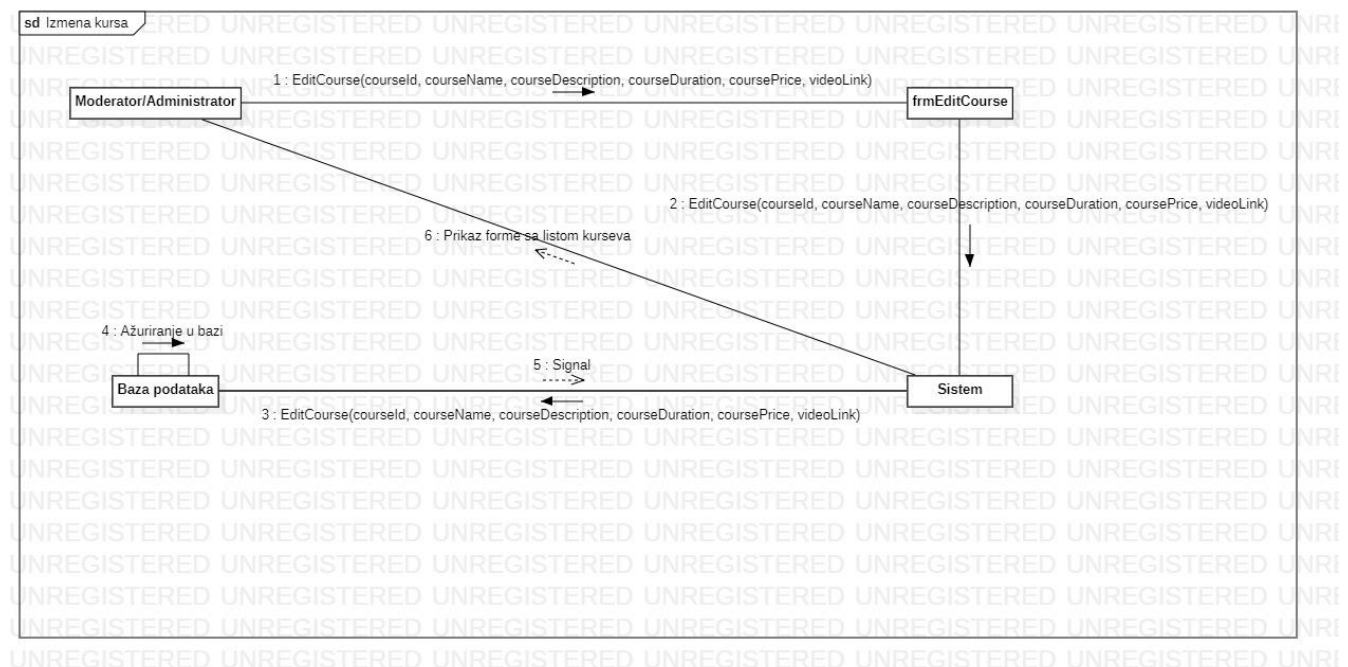
Veza sa SK: SK20

Preduslovi: Korisnik je ulogovan, ima ulogu „Administrator“ ili „Moderator“ u sistemu i kurs postoji u bazi

Postuslovi: Novi podaci izabranog kursa su sačuvani u bazi



Slika 6.1.9. Dijagram sekvenci UG20 – Izmena kursa





Slika 6.1.10. Dijagram kolaboracije UG20 – Izmena kursa

## 6.2 Projektovanje baze podataka



Baza podataka se projektuje na osnovu softverskih klasa strukture. Kao rezultat se dobijaju tabele relacionog sistema za upravljanje bazom podataka. U daljem radu predstavljene su i opisane tri karakteristične tabele iz baze podataka.

Na sledećoj slici (Slika 6.2.1.) prikazana je tabela „Course“ iz baze podataka koja je namenjena za čuvanje podataka o kursu. Kurs ima svoj jedinstven id koji ga jednoznačno određuje, naziv, opis, dužinu trajanja, cenu, vreme kreiranja, id kreatora, polje koje pamti da li je kurs dostupan(obrisan), link video materijala, kao i polja u kojima se čuva putanja do ikonice kursa i dodatnih PDF materijala sa predavanja.

	Column Name	Data Type	Allow Nulls
	courseId	int	<input type="checkbox"/>
	courseName	nvarchar(30)	<input type="checkbox"/>
	courseDescription	nvarchar(MAX)	<input type="checkbox"/>
	courseDuration	real	<input type="checkbox"/>
	coursePrice	real	<input type="checkbox"/>
	videoLink	nvarchar(MAX)	<input type="checkbox"/>
	logoUrl	nvarchar(MAX)	<input type="checkbox"/>
	additionalMaterialUrl	nvarchar(MAX)	<input type="checkbox"/>
	creationTime	datetime2(7)	<input type="checkbox"/>
	isDeleted	bit	<input type="checkbox"/>
	creatorId	nvarchar(MAX)	<input type="checkbox"/>
			<input type="checkbox"/>

Slika 6.2.1. Tabela Course

Na sledećoj slici (Slika 6.2.2.) prikazana je tabela „Bill“ iz baze podataka koja je namenjena za čuvanje podataka o računu korisnika koji se preplatio na određen kurs. Račun ima svoj id, zatim broj kartice, rok važenja kartice, cvc kod, polje koje pamti da li je račun dostupan(obrisan), kao i strane ključeve prema korisniku i prema kursu za koji je kreiran.

	Column Name	Data Type	Allow Nulls
	billId	int	<input type="checkbox"/>
	cardNumber	bigint	<input type="checkbox"/>
	cardValidityPeriod	nvarchar(MAX)	<input type="checkbox"/>
	cvcCode	int	<input type="checkbox"/>
	isDeleted	bit	<input type="checkbox"/>
	courseId	int	<input type="checkbox"/>
	userId	nvarchar(MAX)	<input type="checkbox"/>
			<input type="checkbox"/>

Slika 6.2.2. Tabela Bill

Na sledećoj slici (Slika 6.2.3.) prikazana je tabela „Comment “ iz baze podataka koja je namenjena za čuvanje podataka o komentarima korisnika na određeni kurs. Komentar ima svoj id, zatim tekst komentara, korisničko ime kreatora, vreme kreiranja, polje koje pamti da li je komentar dostupan(obrisan), kao i strane ključeve prema korisniku i prema kursu za koji je kreiran.

	Column Name	Data Type	Allow Nulls
🔑	commentId	int	<input type="checkbox"/>
	commentText	nvarchar(MAX)	<input type="checkbox"/>
	createdDate	datetime2(7)	<input type="checkbox"/>
	userName	nvarchar(256)	<input type="checkbox"/>
	userId	nvarchar(MAX)	<input type="checkbox"/>
	courseId	int	<input type="checkbox"/>
	isDeleted	bit	<input type="checkbox"/>
▶			<input type="checkbox"/>

Slika 6.2.3. Tabela Comment

Kompletna šema baze podataka nalazi se na slici 6.2.4.

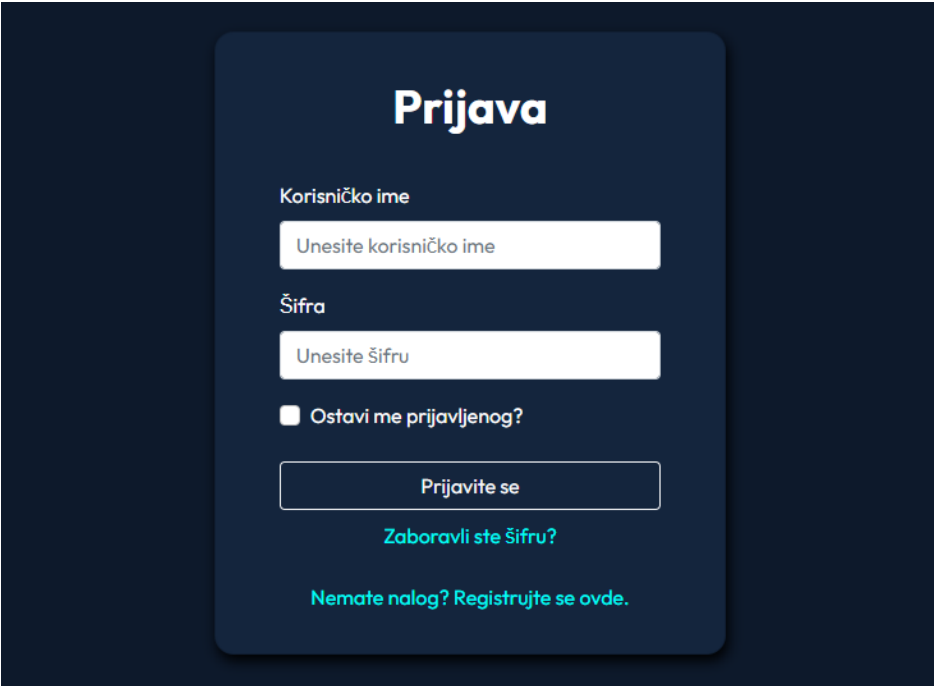


Slika 6.2.4. Šema baze podataka

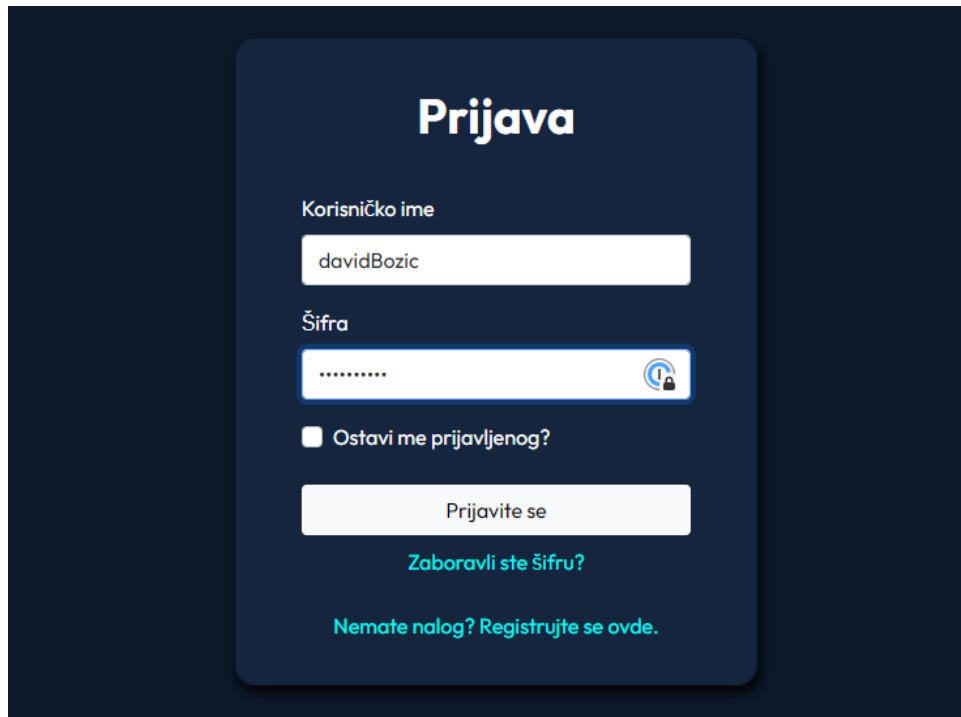
### 6.3 Projektovanje korisničkog interfejsa

Zbog ograničenog obima završnog rada, u ovom poglavlju prikazan je izgled korisničkog interfejsa aplikacije za četiri karakteristična slučaja korišćenja. U nastavku dat je prikaz interfejsa iz ugla određenih aktera, sa opisom interakcije između sistema i aktera ispod svake slike.

Klikom na opciju „Prijavite se“ u desnom delu navigacionog menija prikazuje se ekran kao na slici 6.3.1. i 6.3.2 gde se korisnik može ulogovati unosom korisničkog imena i šifre. Klikom na dugme „Prijavi se“ sistem proverava da li korisnik sa unetim podacima postoji. Ako sistem ne pronađe korisnika sa unetim podacima, korisnik dobija poruku kao na slici 6.3.3. Ukoliko sistem pronađe korisnika sa unetim podacima, šifra ulogovanog korisnika i njegova rola se čuvaju u sesiju i tako postaje dostupna svakoj stranici radi autorizacije pristupa sadržaju kojem nije moguće pristupiti bez odgovarajućeg nivoa pristupa. Nakon uspešnog logovanja korisnik dobija prikaz kao na slici 6.3.4.

The image shows a login form titled "Prijava" (Login) on a dark blue background. The form is a lighter blue rounded rectangle. It contains two input fields: "Korisničko ime" (Username) with placeholder text "Unesite korisničko ime" and "Šifra" (Password) with placeholder text "Unesite šifru". Below the password field is a checkbox labeled "Ostavi me prijavljenog?". At the bottom of the form is a button labeled "Prijavite se". Below the button are two links: "Zaboravili ste šifru?" and "Nemate nalog? Registrujte se ovde."

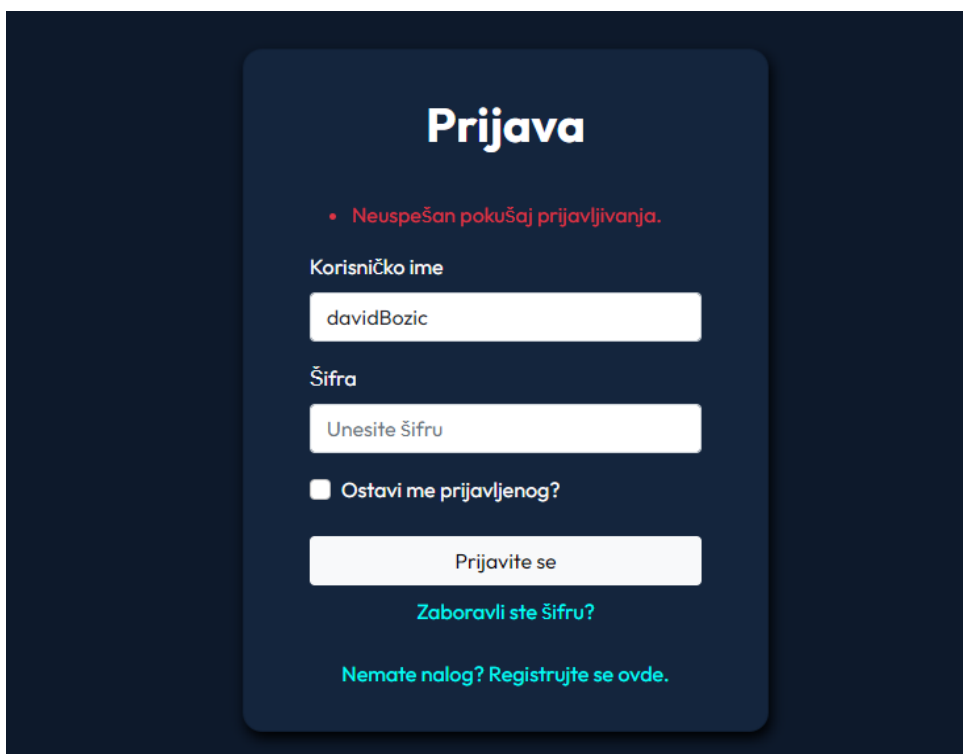
Slika 6.3.1. Forma za logovanje



The screenshot shows a login form titled "Prijava" on a dark blue background. The form is centered and contains the following elements:

- Korisničko ime:** A text input field containing "davidBozic".
- Šifra:** A password input field with masked characters "\*\*\*\*\*" and a toggle icon on the right.
- ☐ **Ostavi me prijavljenog?**
- Prijavite se** button
- [Zaboravli ste šifru?](#)
- [Nemate nalog? Registrujte se ovde.](#)

Slika 6.3.2. Forma za logovanje – Unos podataka

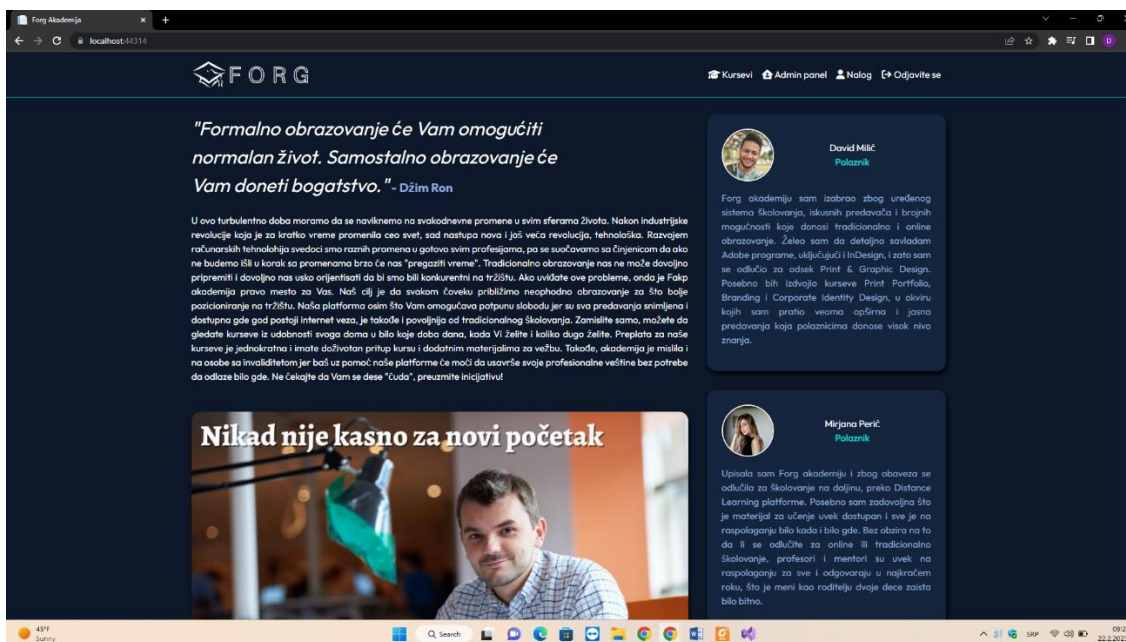


The screenshot shows the same login form as in Slika 6.3.2, but with an error message displayed at the top:

- **Neuspešan pokušaj prijavljivanja.**

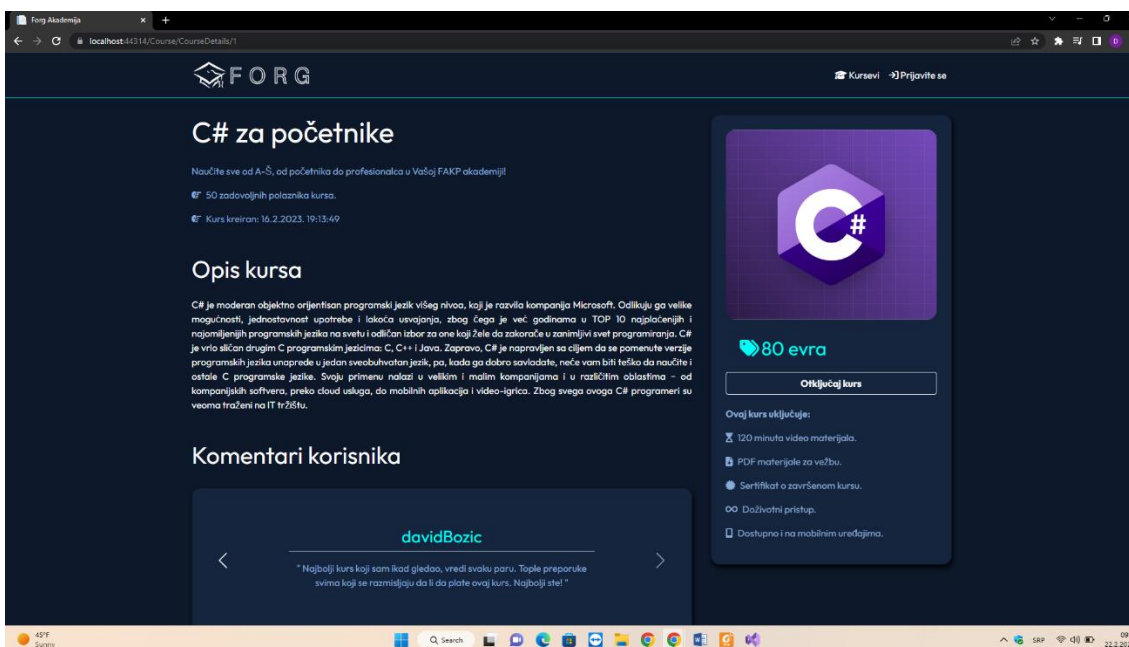
The form fields and buttons remain the same, but the password field is now empty and labeled "Unesite šifru".

Slika 6.3.3. Forma za logovanje – Poruka o neuspešnom logovanju

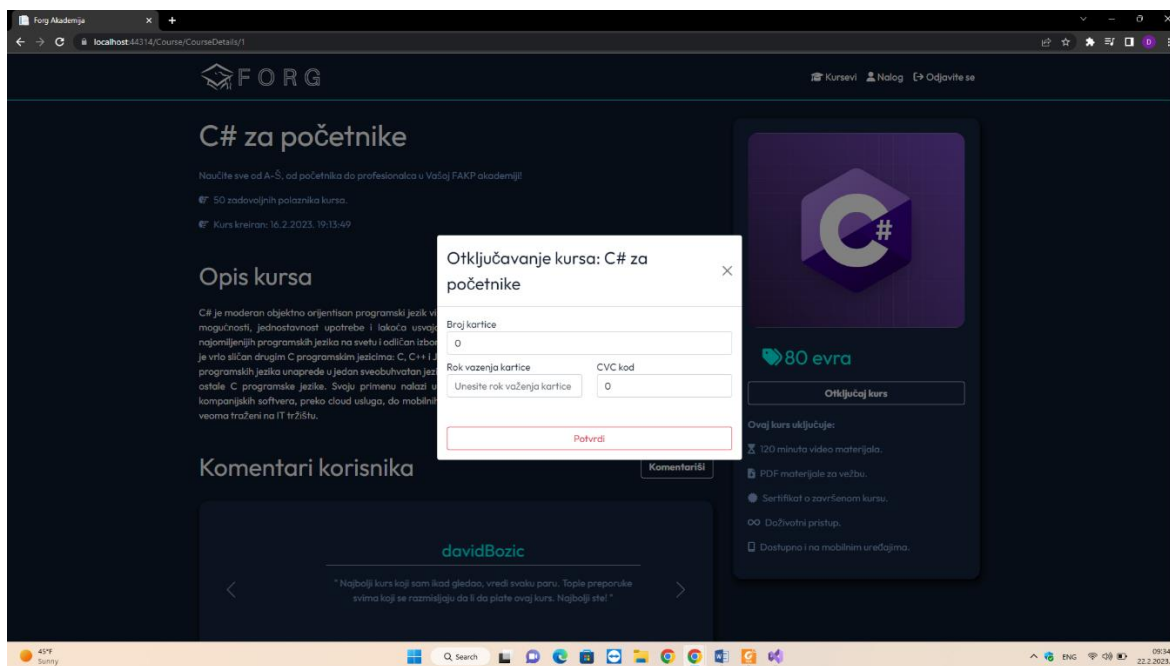


Slika 6.3.4. Početna strana nakon uspješnog logovanja

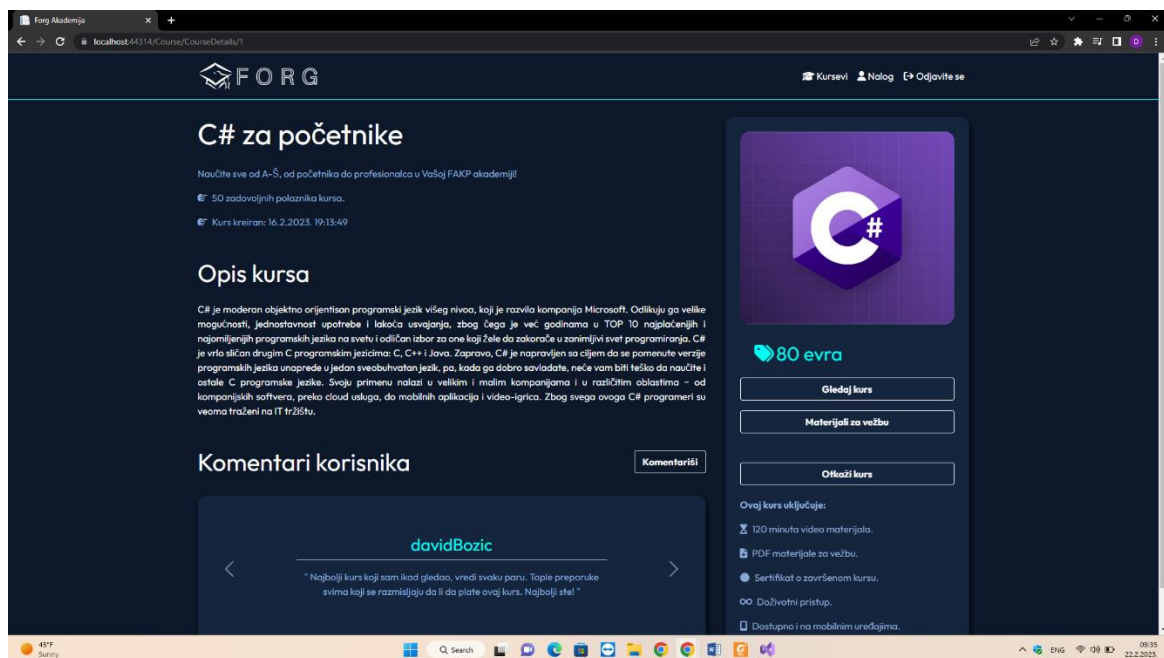
Na slici 6.3.5. prikazana je Web stranica za pregled detalja kursa kojoj može pristupiti ulogovani i neulogovani korisnik. U slučaju da neulogovani korisnik klikne na otključavanje kursa, sistem ga usmerava na formu za logovanje, dok ulogovanog korisnika usmerava na modal za otključavanje kursa ukoliko nije preplaćen na kurs slika 6.3.6., odnosno na sve dostupne materijale kursa ukoliko jeste preplaćen slika 6.3.7.



Slika 6.3.5. Detalji kursa neulogovanog korisnika i ulogovanog koji nije preplaćen

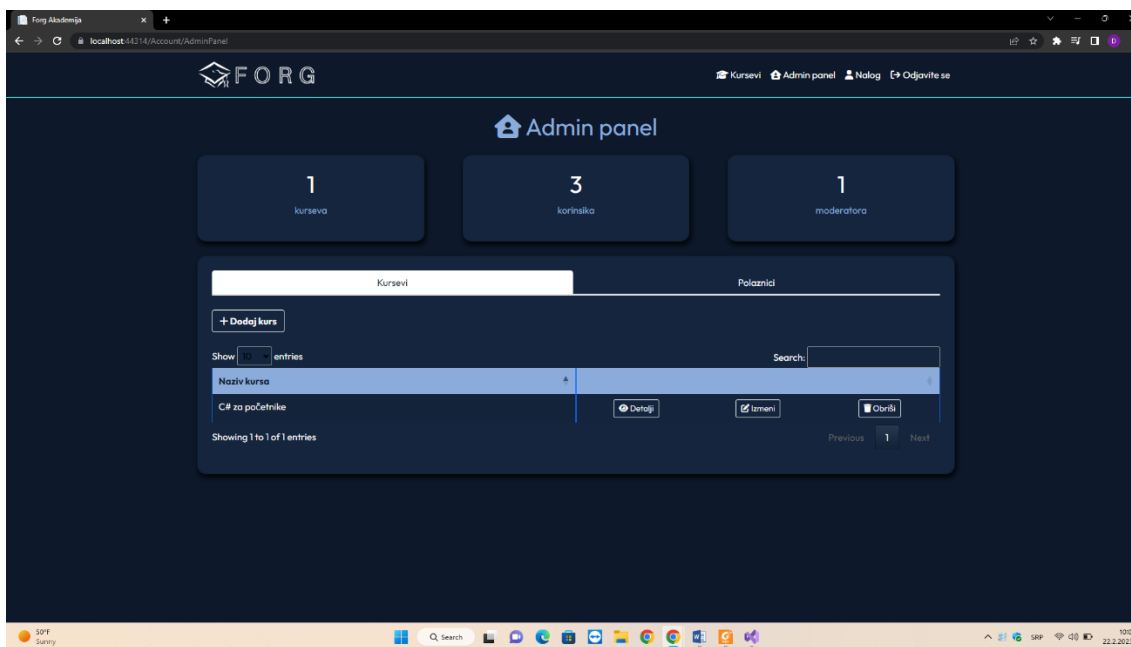


Slika 6.3.6. Detalji kursa otključavanje

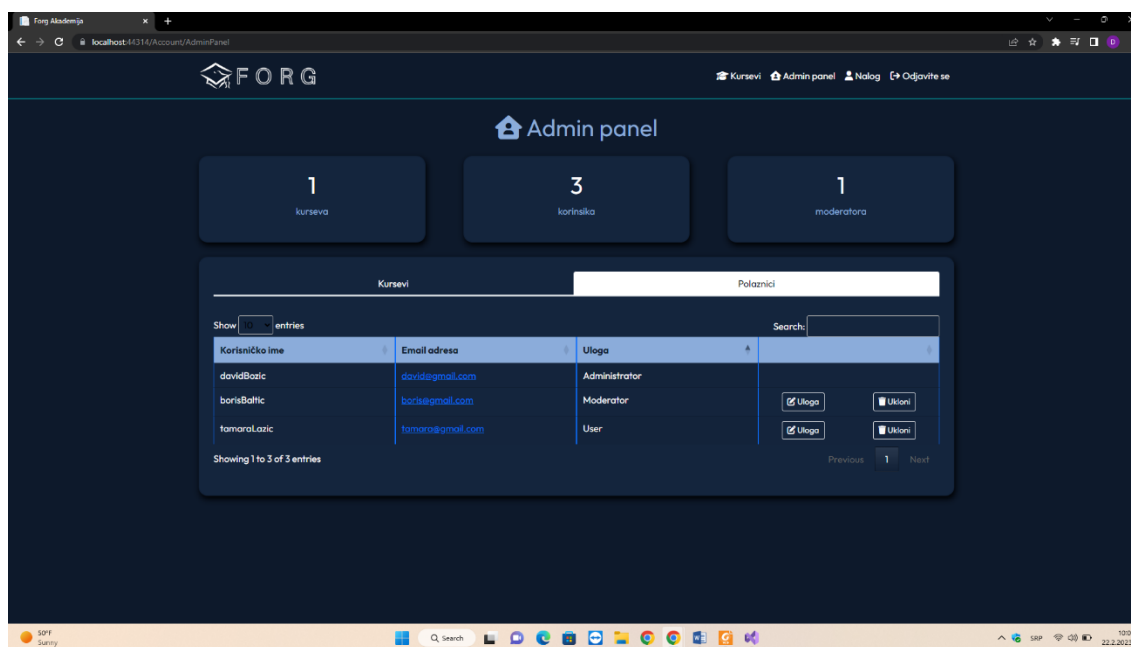


Slika 6.3.7. Detalji kursa ulogovan i preplaćen korisnik

Na slici 6.3.8. je prikazana Web stranica „Admin panel“ kojoj može pristupiti ulogovan korisnik koji ima ulogu „Administrator“ u sistemu, na toj stranici se po defaultu ispisuje broj korisnika platforme, broj dostupnih kurseva, kao i broj moderatora na platformi. Inicijalno je otvoren prvi tab za pregled svih kurseva i operacije koje su nad njima dostupne, a pritiskom na drugi tab se izlistavaju svi korisnici i operacije koje su nad njima dostupne slika 6.3.9.



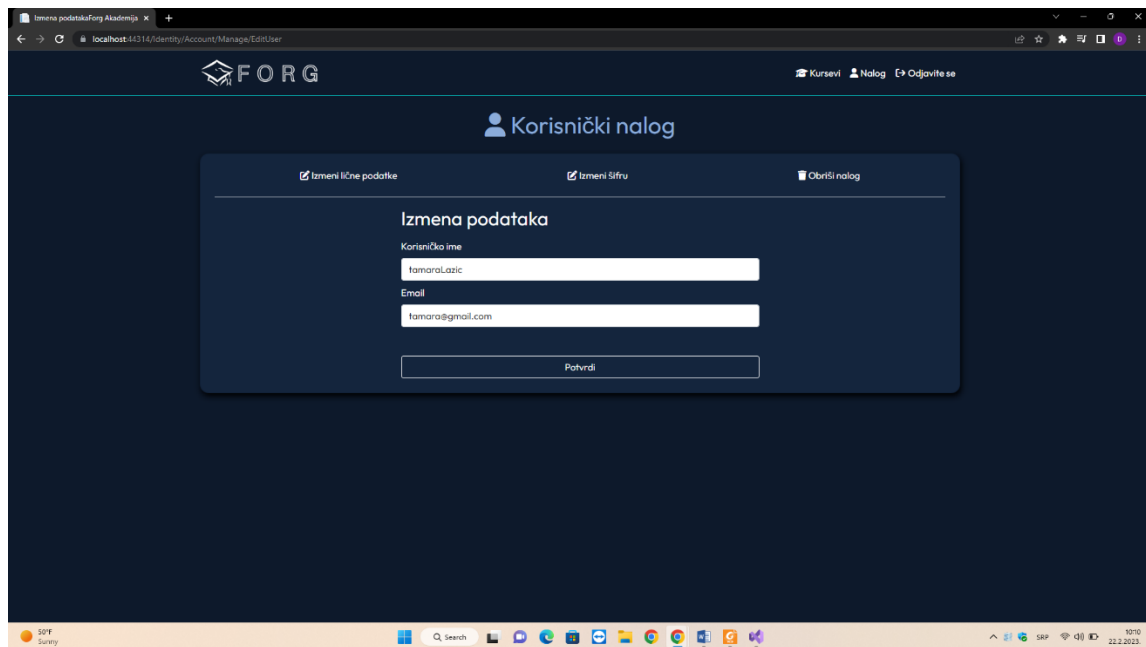
Slika 6.3.8. Admin panel



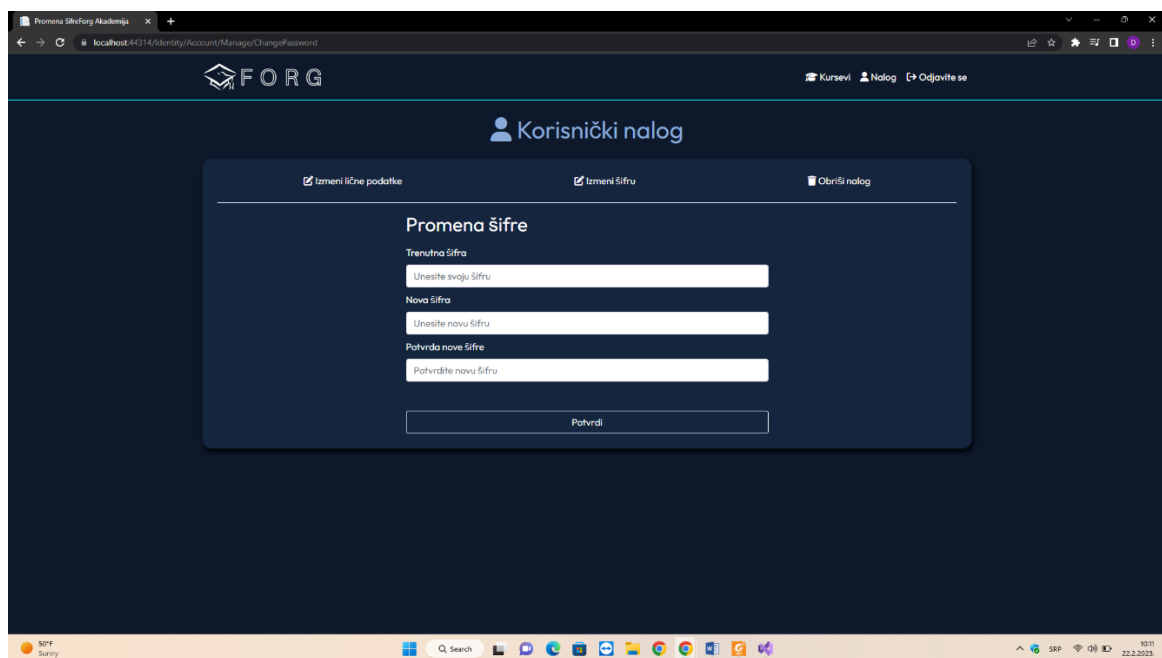
Slika 6.3.9. Admin panel - Korisnici

Na slici 6.3.10. je prikazana Web stranica „Nalog“ kojoj može pristupiti svaki ulogovan korisnik, namena te stranice je manipulacija ličnim informacijama. Na toj stranici je po difoltu otvoren prvi tab za izmenu korisničkog imena i email adrese, a pritiskom na drugi tab se otvara forma za promenu korisničke šifre slika 6.3.11. Ukoliko ulogovan korisnik ima ulogu „User“ ili „Moderator“ u sistemu njemu je dostupan i treći tab na kom se nalazi forma za brisanje ličnog naloga slika 6.3.12.

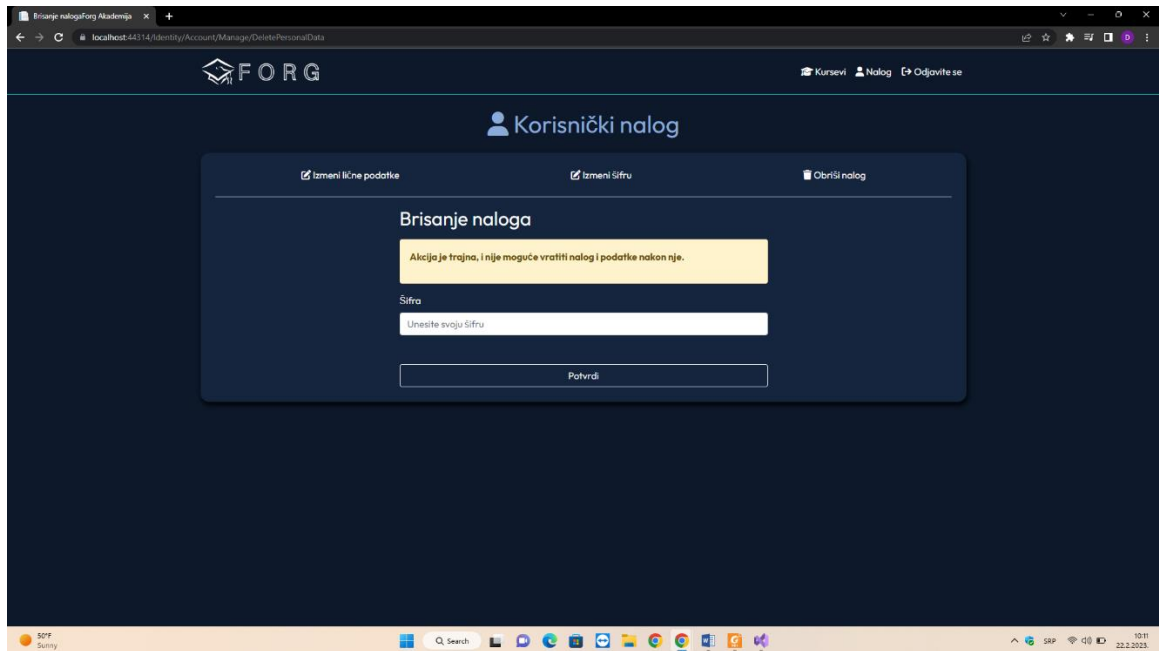




Slika 6.3.10. Nalog – Izmjena ličnih podataka



Slika 6.3.11. Nalog – Izmjena šifre



Slika 6.3.12. Nalog – Brisanje naloga

## 7. ZAKLJUČAK:

U završnom radu prikazan je proces razvoja Web aplikacije za podršku u radu akademije kurseva primenom C# programskog jezika, Microsoft SQL Server-a za upravljanje bazom podataka i Bootstrap-a koji je korišćen za dizajn Web aplikacije. Ono što je veoma važno je da ovakve aplikacije ne samo da predstavljaju budućnost obrazovanja na daljinu, nego korisnicima olakšavaju sam čin učenja i štede dosta vremena i novca.

Web aplikacija je napravljena kao prototip sa mnogim mogućnostima unapređenja. U cilju unapređenja aplikacije, neke od funkcionalnosti koje mogu naknadno da se dodaju su:

- Kreiranje bloga koji će približiti polaznicima kako izgleda IT svet
- Kreiranje online testova gde bi mogli da provere stečeno znanje

Kreiranje bloga koji će približiti polaznicima kako izgleda IT svet

- Uvođenjem bloga na platformi akademije, korisnicima iste se pruža mogućnost edukacije o aktuelnim trendovima na tržištu, zatim o raznim događajima, radionicama, jednostavno jednom rečju pomoć pri usmerenju ka željenom zanimanju.

Kreiranje online testova gde bi mogli da provere stečeno znanje

- Osim materijala sa predavanja koje korisnik ima dostupne kada se preplati na kurs, značajano unapređenje bi bilo i uvođenje online testova za proveru stečenog znanja da bi polaznik tačno mogao da proceni na kom se nivou nalazi. Na taj način polaznik bi mogao objektivno da sagleda svoj napredak bez pomoći neke druge osobe.

Informacione tehnologije, iako mlada inženjerska disciplina, su doživele neverovatnu ekspanziju naročito u zadnjih 10-15 godina. Gotovo da je nemoguće zamisliti normalan život bez njihove pomoći. U doba kada je svet stao zbog najveće pandemije virusa, mnoge grane privrede su opstale isključivo zbog računara i interneta, ali nije privreda jedina koja je koristila te mogućnosti, veliki broj škola je otvorio platforme za učenje na daljinu, bez kojih bi više generacija bilo uskraćeno za redovno obrazovanje. Zatim, osobe sa invaliditetom mogu neuporedivo lakše da se obrazuju jer nisu u obavezi da napuštaju svoj dom da bi im predavanje bilo dostupno. Takođe učenje na daljinu je dosta povoljnije, i dosta fleksibilnije od tradicionalnog školovanja.

## 8. LITERATURA:

- [1] Radić G: Softversko inženjerstvo, ITS, Beograd 2017.
- [2] Anđelić, S: WPF I ASP.NET Framework – projektovanje i implementacija softvera, ITS, Beograd, 2021.
- [3] C#, 2023, dostupno na: <https://learn.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/>
- [4] JavaScript, 2023, dostupno na: <https://www.oxfordwebstudio.com/da-li-znate/staje-javascript.html>
- [5] Microsoft SQL Server, 2023, dostupno na: <https://www.techtarget.com/searchdatamanagement/definition/SQL-Server>
- [6] Bootstrap, 2023, dostupno na: <https://www.slideshare.net/DamjanPavlica/vodi-zarad-sa-bootstrapom-69948458>
- [7] Bootstrap, 2023, dostupno na: <https://falcon-tech.rs/blog/sta-je-bootstrap/>