

# *E - Rezultati*

*Predmet: Softverski inženjering*

*Mentori:*

*Profesor: Dukić Nedžad*

*Asistent: Delalić Sead*

*Autori:*

*Ajka Hujdur, Naida Hujić, Maja Mihojević, Nejra Ajdinović, Čehajić Aida*

*Prirodno - matematički fakultet, Sarajevo, 2018.*

## SADRŽAJ

1. UVOD .....	3
1.1. ODABIR TEHNOLOGIJE I METODIKA RAZVOJA .....	4
2. PROJEKTNJA DOKUMENTACIJA .....	7
2.1. ZADUŽENJA NA PROJEKTU .....	7
2.2. OPIS PROJEKTA .....	9
2.2.1 OPIS KORISNIKA .....	9
2.3. FUNKCIONALNI ZAHTJEVI .....	10
3. Modeliranje – UML dijagrami .....	12
3.1. USE CASE DIJAGRAMI .....	12
3.1.1. Prijava na sistem .....	15
3.1.2. <b>Odjava korisnika</b> .....	15
3.1.3. Registracija korisnika - profesor .....	16
3.1.4. <b>Registracija korisnika - student</b> .....	16
3.1.5. Pregled kurseva - profesor .....	17
3.1.6. Kreiranje kursa .....	17
3.1.7. Kreiranje ispita .....	18
3.1.8. Kreiranje rezultata ispita .....	18
3.1.9. Ažuriranje rezultata ispita .....	19
3.1.10. Pregled svih kurseva - student .....	19
3.1.11. Prijava na kurs .....	19
3.1.12. Prijava na ispit .....	20
3.1.13. Pregled rezultata ispita .....	20
3.2. SEKVENCIJALNI DIJAGRAM .....	21
3.3. DIJAGRAM AKTIVNOSTI .....	24
3.4. ERA model .....	27
4. KORISNIČKE UPUTE .....	34
4.1. Prijava .....	34
4.2. Registracija .....	35
4.3. Početna stranica profesora .....	37
4.4. Početna stranica studenta .....	41
5. TESTIRANJE APLIKACIJE .....	44
5.1. SLUČAJEVI TESTIRANJA (TEST CASES) .....	44
5.2. IZVJEŠTAJ TESTIRANJA .....	53

## 1. UVOD

Softversko inženjerstvo predstavlja primjenu principa inženjerstva u cilju ostvarivanja ekonomičnog softvera, efikasnog i pouzdanog u realnosti na računarskim sistemima.

### **Njegova primjena uključuje:**

- · izradu pouzdanog softvera,
- · izradu softvera koji radi po originalnim specifikacijama,
- · izradu softvera koji je prilagodljiv promjenama u svim fazama životnog ciklusa,
- · izradu softvera koji se može ponovno koristiti na različitim sistemima,
- · razvoj zavisen od raspoloživih resursa,
- · razvoj izveden na vrijeme,
- · efektivan razvoj sa aspekta troškova,
- · razvoj koji zadovoljava korisničke zahtjeve i
- · razvoj koji obezbjeđuje kvalitetan proizvod.

Realizacija informacionog sistema ili projekta je složen proces. Neophodno za kvalitetnu analizu i specificiranje zahtjeva sistema je shvatanje motivacije za izradu istog. Također je bitno i što bolje razumijevanje potreba korisnika, te prilagođavanje principa realizacije softvera tržištu. Model kao proizvod dobre analize zahtjeva korisnika i shvatanja potrebe za izradu projekta treba da omogući dobru specifikaciju, vizualizaciju, konstruiranje, dokumentiranje sistema.

UML (Unified Modeling Language) omogućava konstruisanje šema koje modeliraju sistem opisujući konceptualne stvari, npr. proces poslovanja i funkcije sistema, te konkretne stvari, npr. klasne tipove, šeme baze podataka, softverske komponente i sl. Prvi korak predstavljanja je Use case dijagram (dijagram korištenja), koji omogućava predstavljanje osnovnih načina korištenja sistema i aktora (vanjski entitet koji ima interesa u interakciji sa sistemom) sistema. Use case dijagram jasno definiše funkcionalnost sistema iz perspektive različitih aktora sistema, te opisuje način na koji aktori mogu koristiti sistem, ali ne kako je ta funkcionalnost podržana unutar sistema. Sequence dijagram (dijagram komponentata) prati sami razvoj događanja i pozivanja opšte postavljanih programskih metoda za njihovo izvršavanje. Sequence dijagram prikazuje graf komponenti povezanih zavisnim vezama, kako bi se omogućila lakša analiza reakcije ostalih komponenti na promjene u jednoj komponenti. State dijagram (dijagram stanja) prikazuje mehanizam stanja. Omogućava prikaz dinamičkog ponašanja

elementa modela, te mogući slijed stanja i akcija kroz koje element može prolaziti, a koje su rezultat reakcije na neki signal ili poziv operacija. Activity dijagram (dijagram aktivnosti) je specijalni slučaj dijagrama stanja u kojem su sva stanja akcije ili podakcije, a promjena je inicirana završetkom akcije ili podakcije u izvornom stanju. Koristi se za opisivanje aktivnosti, načina korištenja i interakcija. Stanje objekta može biti prikazano u uglatim zgradama ispod imena objekta, kao npr. [planirano].

Ova aplikacija ima za cilj olakšati profesorima i studentima praćenje ispita, predmeta i ostalih aktivnosti profesora i studenata.

**Za izradu ovog projekta bilo je potrebno proći kroz četiri faze:**

1. Početna faza (Inception) predstavlja fazu definisanja obima projekta.
2. Faze razrade (Elaboration) predstavlja razradu i planiranje te izradu samog projekta.
3. Faza izgradnje (Construction) predstavlja fazu izgradnje sistema.
4. Faza prenosa (Transition) predstavlja fazu prenosa sistema krajnjim korisnicima.

## **1.1. ODABIR TEHNOLOGIJE I METODIKA RAZVOJA**

Kako bi razvili što kvalitetniji programski proizvod s što manje troškova, većinom ćemo se bazirati na korištenje tehnologija koje su nam dostupne preko fakulteta, a koje predstavljaju izuzetno kvalitetno rješenje. Kako bi izradili korisnu projektnu dokumentaciju koristit ćemo Microsoft Office paket. Za planiranje projekta koristit ćemo alat Trello koji organizuje projekat kao jednu tablu, i u kojem možemo detaljno planirati sve aktivnosti, njihov tijek odvijanja, sve resurse i njihova zaduženja te na kraju dobiti detaljne izvještaje u obliku gantograma ili izračuna proračuna projekta. Nadalje, za modeliranje ćemo koristiti alate dostupne na internetu u kojima izrađujemo sve UML dijagrame. Izrađujemo dijagrame slučajeva korištenja, dijagrame aktivnosti, sekvencijalni dijagram, i ERA model. Za programiranje koristimo razvojno okruženje Webstorm, gdje za Front-end koristimo HTML, CSS i Javascript, a za Back-end Node.js. Za bazu podataka koristit ćemo MySQL bazu podataka.

Dakle, kako smo i rekli, za izradu funkcionalnosti same aplikacije, tj na back-end-u koristili smo Node.js. Node.js je serverska JavaScript platforma koja se sastoji od minimalne core biblioteke pored bogatog ekosistema. Radi na V8 JavaScript engine-u, što znači da je veoma brz u izvršavanju.

Tradicionalno programiranje se obavlja sinhrono: Linija koda se izvršava, sistem čeka rezultat, rezultat se procesira, i zati se izvršavanje programa nastavlja. Ponekad taj sistem izvršavanja zahtjeva dugo čekanje, na primjer čitanje iz baze podataka.

JavaScript ima sasvim drugačiji pristup takvom problemu. Uvijek se izvršava samo jedna Nit. Kada se izvršavaju neke spore I/O operacije, kao što je čitanje podataka iz baze, program ne čeka, nego ide na izvršavanje sljedeće linije koda. Kada se I/O operacija vrati, pokreće se callback funkcija i rezultat se procesira.

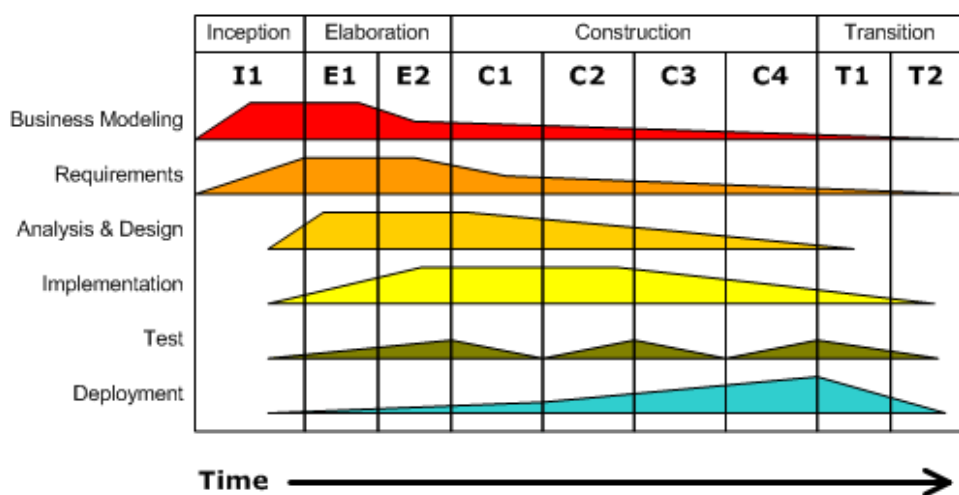
Node.js omogućava korištenje jedinstvenog jezika između front-end i back-end-a. To znači da kompletnu aplikaciju možete realizovati pomoću jednog programskog jezika.

Za bazu podataka koristili smo MySQL. MySQL je besplata, open – source sistem za upravljanje bazama podataka. Vrlo je stabilna, i ima dobro dokumentirane module i ekstenzije te podršku od brojnih programskih jezika: PHP, Perl, Java, Python...

Za frontend smo koristili Bootstrap. Bootstrap je open-source JavaScript framework, odnosno kombinacija HTML-a, CSS-a i JavaScript-a, razvijen sa ciljem da omogući i olakša razvoj web formi (interface-a, tj. *layout-a*) kao i razvoj naprednih web komponenti. Zato ga sa te strane možemo zvati *Front-End-Framework*. Razvijen je od strane *Twitter-a*. Bootstrap je, dakle, kolekcija razvijenih CSS i JavaScript alata i biblioteka. Isto tako je modularnog tipa, što omogućava njegovu dalju lakšu „nadogradnju“ i upotrebu sa različitim modulima koje izrađuju nezavisni developeri. Osim što omogućava i olakšava integrisanje raznih vrsta komponenti (formi, buttons-a, akcija sa tekstom, i drugim), odlično „sarađuje“ sa **JavaScript**-om i sa **jQuery** bibliotekama.

Realizaciji ovog projekta pristupili smo preko iterativnog pristupa razvoja aplikacije gdje se raspoređuje potrebno vrijeme na izradi projekta na nekoliko manjih projekata koje nazivamo iteracijama. To nam omogućava da se fokusiramo na kratkoročne ciljeve i tako ih postupno završavamo s velikom uspjehnošću. U prvoj iteraciji isplanirali smo nekoliko ciljeva koje moramo postići. Važno je da prilikom početka iterativnog procesa imamo isplanirano što u tom iterativnom procesu moramo napraviti, koji je prioritet pojedinih aktivnosti i vrijeme trajanja realizacije tih aktivnosti.

Najpoznatija iterativna metoda je RUP ( Rational Unified Process ), koja je i korištena u ovom projektu. Najvažnija ideja koju RUP donosi je jednostavna: konzistentnom primjenom RUP pravila igre, otklanjamo rizike od najranijih trenutaka u projektu (inception), gradimo arhitekturne prototipove čim imamo dovoljno informacija o projektu (elaboration) te na temelju njih zaključujemo kako i kamo dalje, kroz niz kratkih iteracija produciramo nove verzije aplikacije/produkta tako da u svakoj iteraciji nastane rezultat koji funkcionalno odgovara zahtjevima s početka iteracije te ga se može demonstrirati stakeholderima (construction), i na kraju pripremamo aplikaciju/proizvod za produkciju kroz završno korisničko testiranje, dotjerivanje dokumentacije, obuku korisnika i konačno stavljanje u rad u produkcijsku infrastrukturu (transition).



**Metodologija koja je korištena u ovom radu se sastoji od četiri koraka:**

1. Opis slučajeva korištenja
2. Analiza
3. Implementacija
4. ER Model

## 2. PROJEKTNÁ DOKUMENTACIJA

### 2.1. ZADUŽENJA NA PROJEKTU

Tim koji će raditi na ovom projektu sastoji se od pet članova. Dva člana rade izgled i dizajn aplikacije (Maja Mihojević i Nejra Ajdinović), dva člana rade pozadinu aplikacije i na njenoj funkcionalnosti (Ajka Hujdur i Naida Hujčić) i jedan član radi na kreiranju baze, tabela u bazi i njihovih veza( Čehajić Aida ).

Aktivnosti koje se izvršavaju na projektu prilikom izrade aplikacije:

opis i specifikacija korisničkih zahtjeva

kreiranje UML dijagrama

kreiranje ERA modela i baze podataka

izrada skica formi programa

programiranje aplikacije

testiranje aplikacije

U nastavku možete vidjeti Ganttov dijagram za dati projekat.

Ganttov dijagram, odnosno gantogram je vremenski dijagram koji prikazuje popis aktivnosti u vremenu. On nam omogućava da lakše pratimo tok našeg projekta i daje nam uvid u resurse koji se koriste na pojedinim aktivnostima. Preko gantograma možemo pratiti eventualna odstupanja i lakše se prilagoditi nekim neželjenim situacijama koje nastaju na projektu.

[illegible]



## 2.2. OPIS PROJEKTA

Projekat 'E-Rezultati' bazira se na izradi aplikacije koja omogućava profesoru i studentu lakši pregled ocjena i studenata koji pohađaju predmet. Pored toga aplikacija omogućava profesoru da kreira svoj kurs, a studentu da se prijavi na taj kurs. Profesor za odgovarajući kurs ima mogućnost da kreira nadolazeći ispit, na koji se student prijavljuje. I na kraju profesor kreira rezultate ispita, pri čemu se svakom studentu šalju samo njegovi rezultati, i ima uvid samo u svoje rezultate.

### 2.2.1 OPIS KORISNIKA

#### 2.2.1.1 Opis potencijalnog tržišta

Profesori koji žele lakši način za organizovanje svojih kurseva, i uvid koliko studenata polaže njegov kurs. Studenti koji imaju bolji uvid u aktivnosti na kursu (ispiti), kao i mogućnost da vide samo svoje rezultate.

#### 2.2.1.2 Profili korisnika

Postoje dva tipa korisnika koji će koristiti WEB aplikaciju E – Rezultati, profesor i student.

**Profesor:** ima pristup funkcijama preko kojih kreiraju kurseve i ispite na odgovarajućem kursu, kao i rezultate završenih ispita.

**Student:** ima pristup funkcija preko kojih se prijavljuje na kurseve koje sluša odgovarajući semestar, kao i na ispite koji su kreirani za taj kurs. Student može vidjeti i svoje rezultate nakon završetka ispita, koje će dobiti na e-mail.

### 2.2.1.3 Osnovne potrebe korisnika

Profesor ima potrebu da na jednostavan i brz način kreira kurseve na kojima drži nastavu u odgovarajućem semestru / godini, kako bi studenti na jednostavan način pristupili tom kursu.

Profesor ima potrebu i da upravlja podacima vezanim za kurseve tj da može da kreira ispite na tom kursu. Također profesor ima potrebu da ima uvid u to koji studenti su prijavljeni na njegov kurs, kao i koji studenti su prijavljeni na ispit.

Student ima potrebu da vidi sve kurseve koji su kreirani na odsjeku koji unese prilikom registracije. Također da se prijavi na odgovarajući kurs pomoću šifre koju dobije od profesora.

Student ima potrebu da se prijavi na ispit koji je kreiran za kurs na koji je on prijavljen. Pored toga student ima potrebu da dobije rezultate ispita na e-mail, kao i da te rezultate vidi na samoj stranici.

## 2.3. FUNKCIONALNI ZAHTJEVI

### 2.3.2.1 Registrovanje korisnika

Korisnik ima mogućnost da se registruje na sistem nakon čega će moći da koristi usluge kojima raspolaže aplikacija

### 2.3.2.2 Kreiranje kursa

Korisnik - profesor ima mogućnost da kreira kurs

### 2.3.2.3. Prijava na kurs

Korisnik - student ima mogućnost da se prijavi na kurs pomoću šifre koju dobije od profesora

### 2.3.2.4 Kreiranje ispita

Korisnik - profesor ima mogućnost da kreira ispit za odgovarajući kurs, prilikom čega navodi vrijeme i mjesto održavanja ispita.

#### 2.3.2.5 Prijava na ispit

Korisnik - student ima mogućnost da se prijavi na ispit, nakon čega korisnik - profesor ima uvid koliko studenata, kao i koji studenti su prijavljeni na ispit.

#### 2.3.2.6 Kreiranje rezultata ispita

Korisnik - profesor, nakon završetka ispita, ima mogućnost da kreira rezultate ispita za sve studente koji su pristupili ispitu.

#### 2.3.2.7 Pregled rezultata ispita

Korisnik - student ima mogućnost da na e-mail dobije rezultate ispita, kao i da iste pogleda na aplikaciji.

## 3. MODELIRANJE – UML DIJAGRAMI

### 3.1. USE CASE DIJAGRAMI

Dijagram slučaja korištenja (eng. UseCase) koristi se za opis osnovnih funkcionalnih cjelina i njihovo ponašanje, te za opis vanjskih učesnika i njihove interakcije sa sistemom. Uz pomoć UseCase dijagrama prikazali smo primjer rada sistema sa stajališta korisnika, odnosno učesnika. Osnovni koncepti UseCase dijagrama su: slučaj korištenja (eng. UseCase), učesnik (eng. Actor), veza između njih. Svaki slučaj korištenja ima svoju specifikaciju, odnosno opisni tekst scenarija koji se odvija kada korisnik obavlja neki slučaj korištenja. Imamo opisane sve moguće scenarije koji se očekuju pri radu sa aplikacijom. Specifikacije slučajeva korištenja su zapravo najvažniji dio modela slučajeva korištenja, dok dijagrami služe za ilustraciju scenarija.

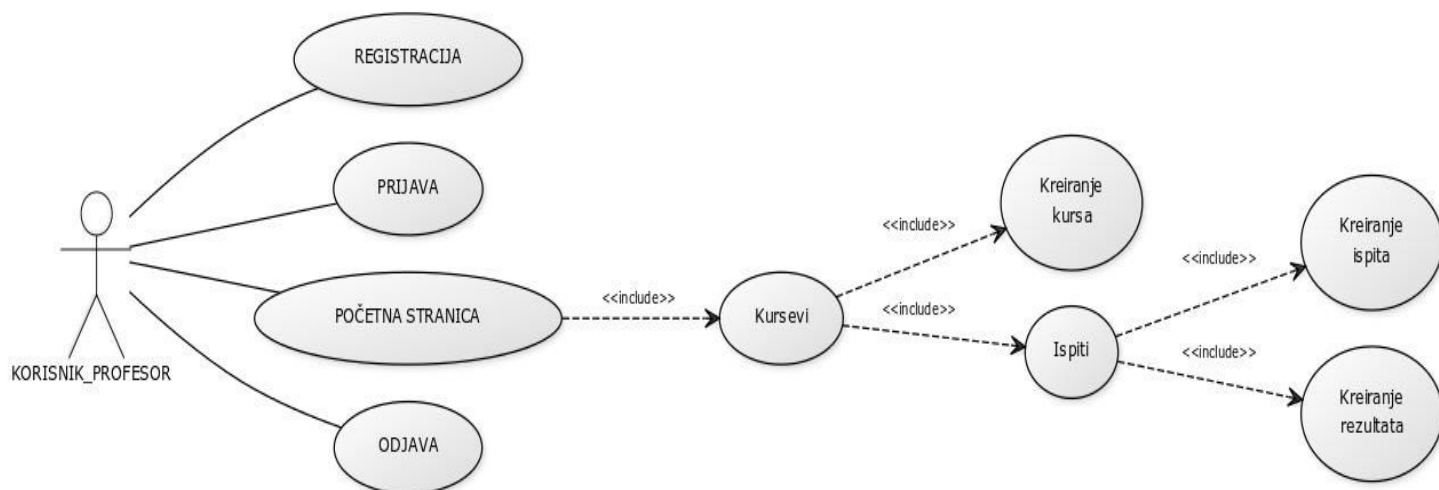
S obzirom da ova aplikacija ima dvije vrste korisnika, u nastavku će biti navedeni slučajevi korištenja za svaku vrstu.

Slučajevi korištenja za sve aktore:

1. Prijava na aplikaciju
2. Odjava sa sistema

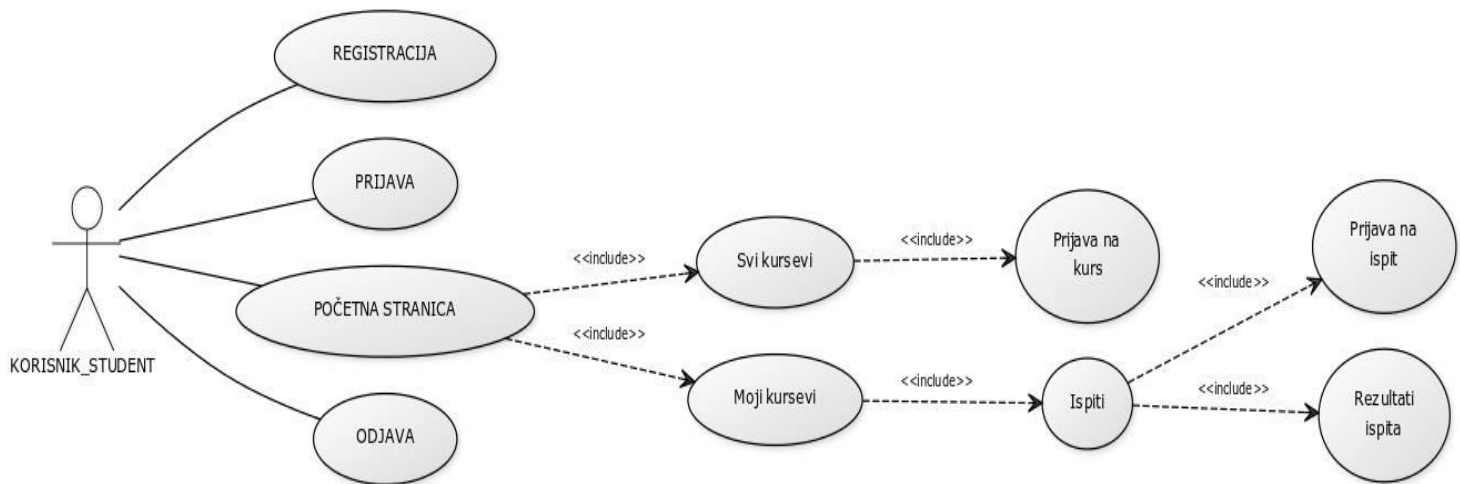
## Slučajevi korištenja za korisnika - profesor

1. Registracija
2. Pregled početne
3. Pregled njegovih kurseva
4. Kreiranje novog kursa
5. Pregled kreiranog kursa
6. Kreiranje ispita
7. Kreiranje rezultata ispita



## Slučajevi korištenja za korisnika - student

1. Registracija
2. Pregled početne
3. Pregled svih kreiranih kurseva na odsjeku
4. Prijava na kurs
5. Pregled kursa na koji je prijavljen
6. Prijava na ispit
7. Pregled rezultata ispita



U sljedećoj fazi nalazi se opis slučajeva korištenja. Tekstualni opis slučaja korištenja ima sljedeću strukturu:

- Kratki opis
- Akteri
- Preconditions
- Opis
- Postconditions

### 3.1.1. Prijava na sistem

**Kratak opis:** Logovanje korisnika.

**Akteri:** Profesor i student

**Preconditions:** Korisnik se nalazi na strani za logovanje

**Opis:**

1) Korisnik otvara stranicu za logovanje

izuzetak: računar nije povezan na mrežu

2) Korisnik upisuje korisničko ime i šifru u stranici za logovanje i pritiska dugme „Prijava“

izuzetak – korisnik nema korisničko ime i šifru

3) Sistem provjerava ime i šifru u bazi podataka i otvara novu stranicu sa sadržajem koji je namijenjen ovom korisniku

/izuzetak – korisnik je upisao pogrešnu šifru/

Izuzeci: - računar nije povezan na mrežu - potrebno je konektovati računar na mrežu

- korisnik nema korisničko ime i šifru - korisnik treba da izabere opciju registracija novog korisnika.

- korisnik je upisao pogrešnu šifru - sistem vraća korisnika na stranicu za logovanje

**Postconditions:** Korisnik je uspješno prijavljen na aplikaciju. Moguće je da dalje koristi ponuđeni sadržaj

### 3.1.2. Odjava korisnika

**Kratki opis:** Odjava korisnika sa sistema

**Akteri:** Profesor i student

**Preconditions:** Korisnik je prijavljen u aplikaciju

**Opis:**

1) Korisnik pritiska dugme za odjavu “Logout”

2) Sistem odjavljuje korisnika

Izuzetak - računar nije povezan na mrežu - potrebno je konektovati računar na mrežu

**Postconditions:** Korisnik je uspješno odjavljen, i preusmjeren na stranicu za prijavu korisnika

### 3.1.3. Registracija korisnika - profesor

**Kratak opis:** Registrovanje novog korisnika

**Akteri:** Profesor

**Preconditions:** Profesor želi da se prijavi na aplikaciju i zna podatke koji su potrebni za prijavu (ime, prezime, fakultet, odsjek, username, email, password)

**Opis:**

1) Profesor otvara stranicu za registrovanje profesora

izuzetak: računar nije povezan na mrežu

2) Profesor upisuje svoje podatke u stranicu za registrovanje i pritiska dugme „Registracija“

izuzeci: - korisnik je uneo šifru koja ima manje od šest karaktera ili korisnik je uneo pogrešan format e-mail

- uneseni e-mail se već koristi
- uneseni username se već koristi

3) Profesor se nalazi na stranici za prijavu korisnika

izuzeci: - računar nije povezan na mrežu ( potrebno je konektovati računar na mrežu)

**Postconditions:** Profesor je registrovan na aplikaciju.

### 3.1.4. Registracija korisnika - student

**Kratak opis:** Registrovanje novog korisnika

**Akteri:** Student

**Preconditions:** Student želi da se prijavi na aplikaciju i zna podatke koji su potrebni za prijavu (ime, prezime, fakultet, odsjek, smjer, broj indeksa, username, email, password)

**Opis:**

1) Student otvara stranicu za registrovanje studenta

izuzetak: računar nije povezan na mrežu

2) Student upisuje svoje podatke u stranicu za registrovanje i pritiska dugme „Registracija“

izuzeci: - korisnik je uneo šifru koja ima manje od šest karaktera ili korisnik je uneo pogrešan format e-mail

- uneseni e-mail se već koristi



- uneseni username se već koristi
- uneseni broj indeksa se već koristi

3) Student se nalazi na stranici za prijavu korisnika

izuzeci: - računar nije povezan na mrežu (potrebno je konektovati računar na mrežu)

**Postconditions:** Student je registrovan na aplikaciju.

### 3.1.5. Pregled kurseva - profesor

**Kratak opis:** Pregled kurseva koje je profesor kreirao

**Akteri:** Profesor

**Preconditions:** Korisnik je prijavljen na aplikaciju

**Opis:**

- 1) Korisnik je pritisnuo u sidebar-u dugme “Moji kursevi”
- 2) Otvara se nova stranica sa prikazom svih kreiranih kurseva od strane tog profesora u tabelarnom obliku

Izuzetak: -računar nije povezan na mrežu, -profesor još uvijek nije kreirao nijedan kurs

**Postconditions:**

### 3.1.6. Kreiranje kursa

**Kratak opis:** Kreiranje novog kursa

**Akteri:** Profesor

**Preconditions:** Profesor je prijavljen i nalazi se na stranici sa prikazom svih kurseva

**Opis:**

- 1) Profeso pritiska dugme “Dodaj kurs”
- 2) Otvara se popup prozor sa formom za kreiranje novog kursa
- 3) U input polja profesor unosi naziv kursa, semestar, ciklus i šifru kursa
- 4) Profesor pritiska dugme “Kreiraj”

Izuzeci: -računar nije povezan na mrežu, -unesena šifra se već koristi

**Postconditions:** Kurs se uspješno kreira, profesor je vraćen na stranicu sa svim kursevima

### 3.1.7. Kreiranje ispita

**Kratki opis:** Kreiranje ispita za odgovarajući kurs

**Akteri:** Profesor

**Preconditions:** Korisnik ima kreiran barem jedan kurs

**Opis:**

- 1) Korisnik pritiska dugme u sidebar-u “Moji kursevi”
- 2) Iz tabele bira kurs za koji želi da kreira ispit
- 3) Otvara se stranica od tog kursa
- 4) Pritiska dugme “+Novi ispit”
- 5) Otvara se popup prozor sa formom za kreiranje ispita
- 6) Profesor popunjava formu sa podacima dio ispita, mjesto održavanja i vrijeme održavanja ispita
- 7) Profesor pritiska dugme “Kreiraj”

Izuzeci: - računar nije povezan na mrežu,

**Postconditions:** Ispit se uspješno kreira, profesor je vraćen na stranicu od kreiranog kursa

### 3.1.8. Kreiranje rezultata ispita

**Kratki opis:** Profesor kreira rezultate ispita za prijavljene studente

**Akteri:** Profesor

**Preconditions:** Na ispit mora biti prijavljen barem jedan student

**Opis:**

- 1) Profesor unosi osvojene bodove za svakog studenta u tabeli
- 2) Profesor unosi maksimalne bodove
- 3) Profesor unosi datum uvida
- 4) Profesor pritiska dugme “Spasi rezultate”

**Postconditions:** Rezultati se uspješno kreiraju, i svakom studentu iz tabele se šalje e-mail sa osvojenim bodovima, koliko je maksimalno bodova i kada je uvid.

### 3.1.9. Ažuriranje rezultata ispita

**Kratki opis:** Profesor ažurira već kreiranje rezultate ispita

**Akteri:** Profesor

**Preconditions:** U tabelu već moraju biti uneseni podaci

**Opis:**

- 1) Profesor ažurira osvojene bodove za odgovarajućeg studenta
- 2) Profesor pritiska dugme “Spasi rezultate”

**Postconditions:** Rezultati se uspješno ažuriraju

### 3.1.10. Pregled svih kurseva - student

**Kratki opis:** Pregled svih kurseva koji su kreirani na odsjeku na kojem je student prijavljen

**Akteri:** Student

**Preconditions:** Student je prijavljen na aplikaciju

**Opis:**

- 1) Student je pritisnuo dugme “Svi kursevi” u sidebar-u

Izuzeci: - računar nije prijavljen na mrežu

**Postconditions:** Student vidi listu u tabelarnom prikazu svih kurseva kreiranih na tom odsjeku

### 3.1.11. Prijava na kurs

**Kratki opis:** Prijava na kreirani kurs

**Akteri:** Student

**Preconditions:** Student je prijavljen na aplikaciju

**Opis:**

- 1) Student je pritisnuo dugme “Svi kursevi” u sidebar-u
- 2) Student je pritisnuo dugme “Prijava” pored kursa na koji se želi prijaviti
- 3) Otvara se popup prozor gdje upisujemo šifru kursa koji smo dobili od profesora
- 4) Student pritiska dugme “Prijava” da bi se prijavio na kurs

Izuzeci: - računar nije povezan na mrežu, - šifra nije tačna, student ponovo unosi šifru

**Postconditions:** Student je uspješno prijavljen na ispit, i u opadajućem meniju “Moji kursevi” nalazi se kurs na koji je prijavljen

### 3.1.12. Prijava na ispit

**Kratki opis:** Prijava na kreirani ispit

**Akteri:** Student

**Preconditions:** Student je prijavljen na kurs

**Opis:**

- 1) Student bira kurs za koji je kreiran ispit na koji želi da se prijavi
- 2) Pritiska dugme “Prijava” pored ispita na koji želi da se prijavi

Izuzeci: računar nije povezan na mrežu

**Postconditions:** Student je uspješno prijavljen na ispit

### 3.1.13. Pregled rezultata ispita

**Kratki opis:** Student na stranici kursa može vidjeti rezultate ispita

**Akteri:** Student

**Preconditions:**

**Opis:**

- 1) Student bira kurs za koji želi vidjeti rezultate ispita

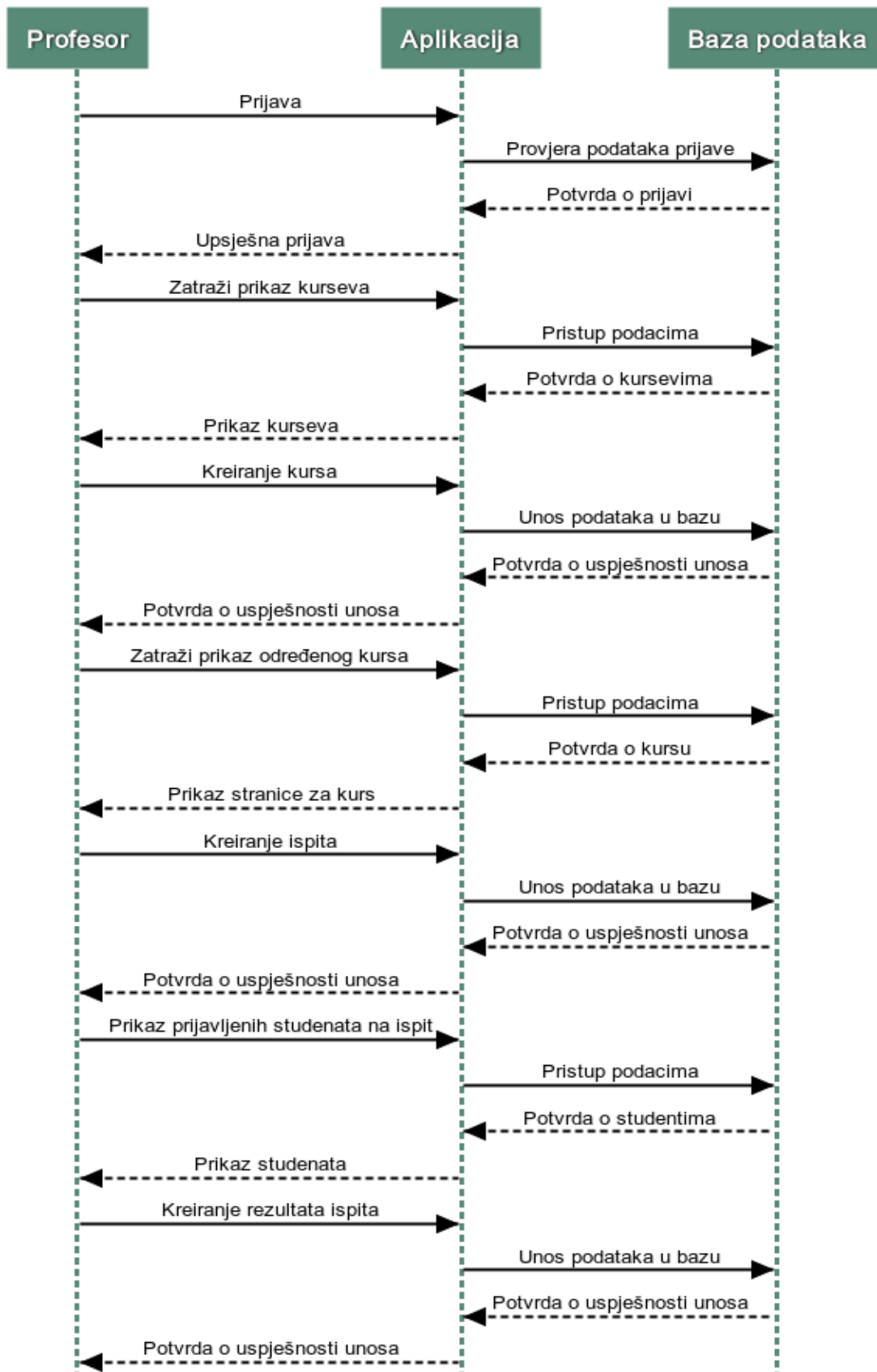
**Postconditions:** Na dnu stranice nalazi se tabela sa rezultatima, gdje je navedeno za koji dio ispita su rezultati, koliko je maksimalno bodova i kada je uvid.

## 3.2. SEKVENCIJALNI DIJAGRAM

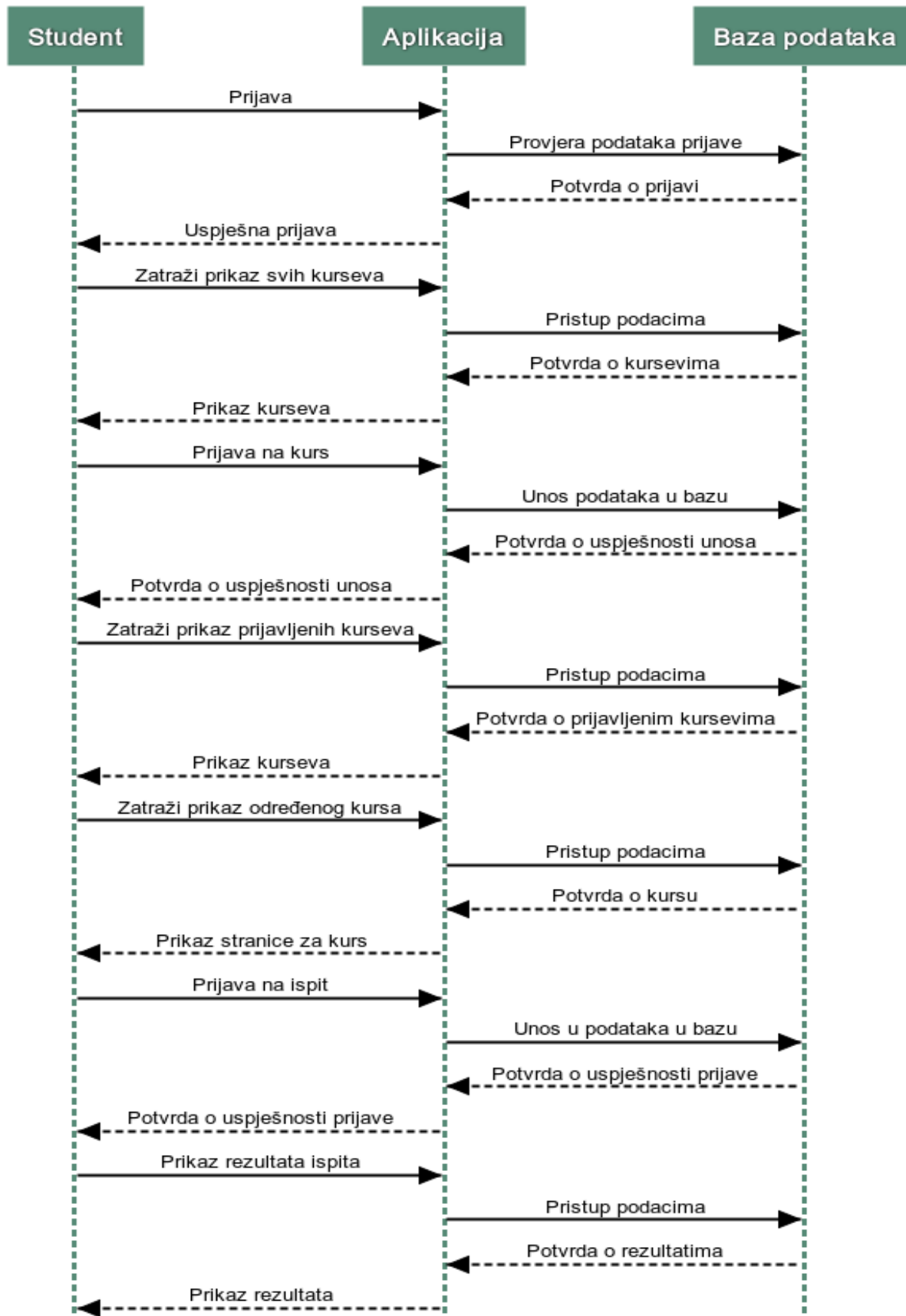
U fazi analize opisuje se logička struktura i ponašanje softverskog sistema (poslovna logika softverskog sistema). Ponašanje softverskog sistema opisujemo pomoću sekvencijalnih dijagrama, koji se prave za svakog korisnika, i pomoću ugovora o sistemskim operacijama, koje se dobijaju na osnovu sistemskih dijagrama sekvenci.

Dijagrami sekvenca se koriste da bi istražili i prikazali sekvence u kojima objekti međusobno djeluju između sebe. Objekti pri tom mogu biti: Korisnik, Aplikacija i Baza podataka. Dijagrami sekvenca obično opisuju sekvence poruka između više objekata, gdje je redoslijed poruka detaljno opisan.

## Sekvencijalni dijagram - profesor



## Sekvencijalni dijagram - student

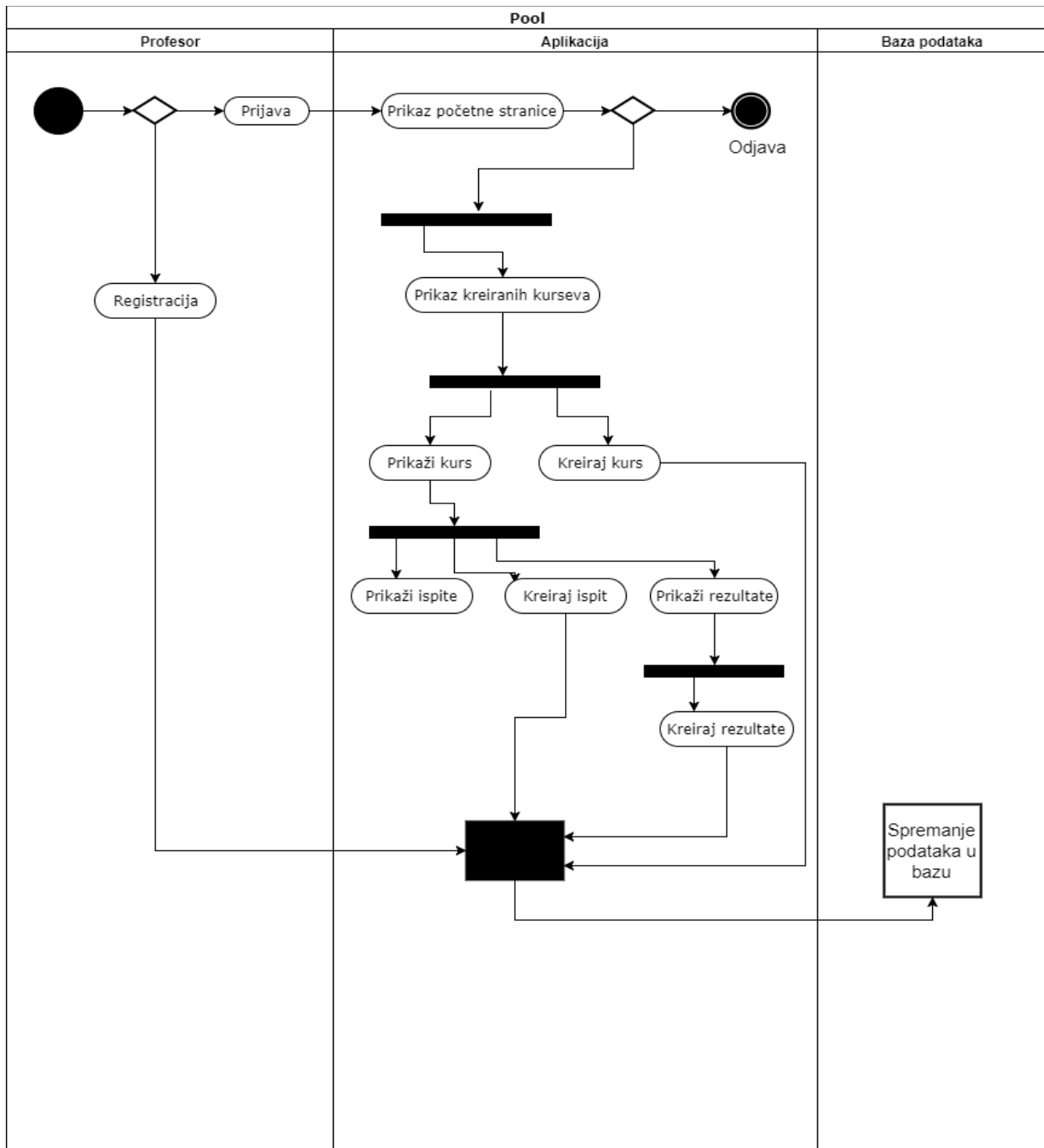


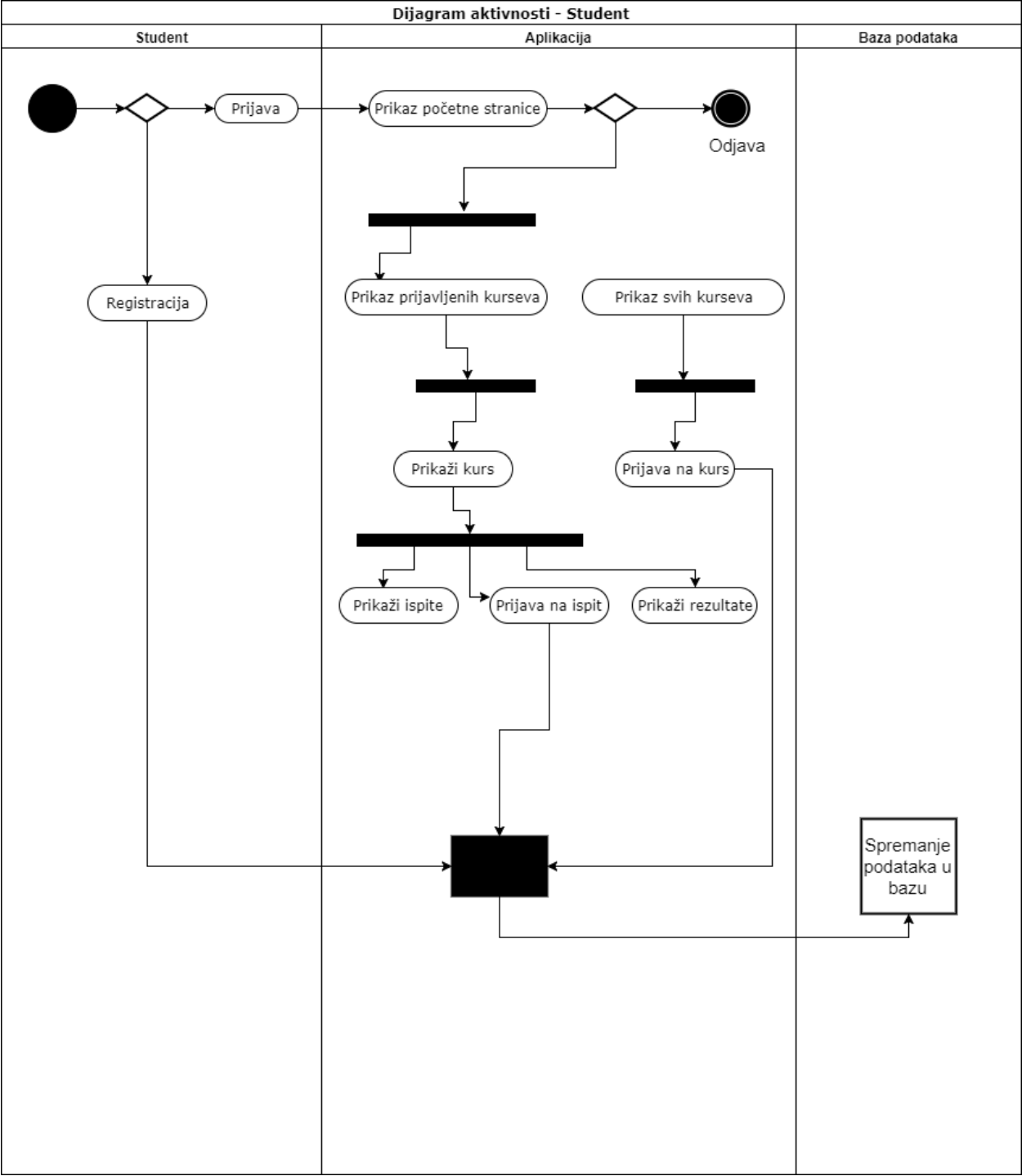
### 3.3. DIJAGRAM AKTIVNOSTI

Ili (eng. Activity Diagram) je specijalni oblik dijagrama prijelaza stanja, koji prikazuje stanja i prijelaze između slučajeva korištenja. Dijagrame aktivnosti smo organizirali u domene odgovornosti (eng. swimlane), kako bi ga uspjeli što bolje razumjeti. Svaki učesnik u dijagramu aktivnosti ima svoju plivačku stazu (eng. swimlane) uz pomoć koje lako očitamo koja aktivnost pripada kojem učesniku. Služe za opis logike procedura, također prikazuju primjenu određene operacije u određenoj klasi opisujući akcije i aktivnosti.

Dijagrami aktivnosti su namijenjeni modeliranju dinamičkih aspekata (ponašanja) sistema.







### 3.4 ERA MODEL

ERA model je skraćenica od engl. riječi Entity, Relationship, Attributes, što u prijevodu znači Entiteti, Veze, Atributi.

ERA model je grafička prezentacija znanja o objektima, vezama i svojstvima. Objekti mogu biti jaki i slabi. Jaki mogu postojati nezavisno od drugih objekata, a slabi egzistencijalno zavise o drugim objektima.

Kad govorimo o osobini (atributu), razlikujemo dvije stvari: deskriptor i identifikator. Deskriptor upotpunjuje opis objekta dok identifikator jednoznačno određuje svaku instancu objekta.

O vezama možemo govoriti na tri načina. Prvi način je kad govorimo o redu veze (unarna i binarna veza), drugi je način učešća (obvezno i opcionalno) i treći način je kad govorimo o tipu povezanosti (1:1, 1:M, M:N).

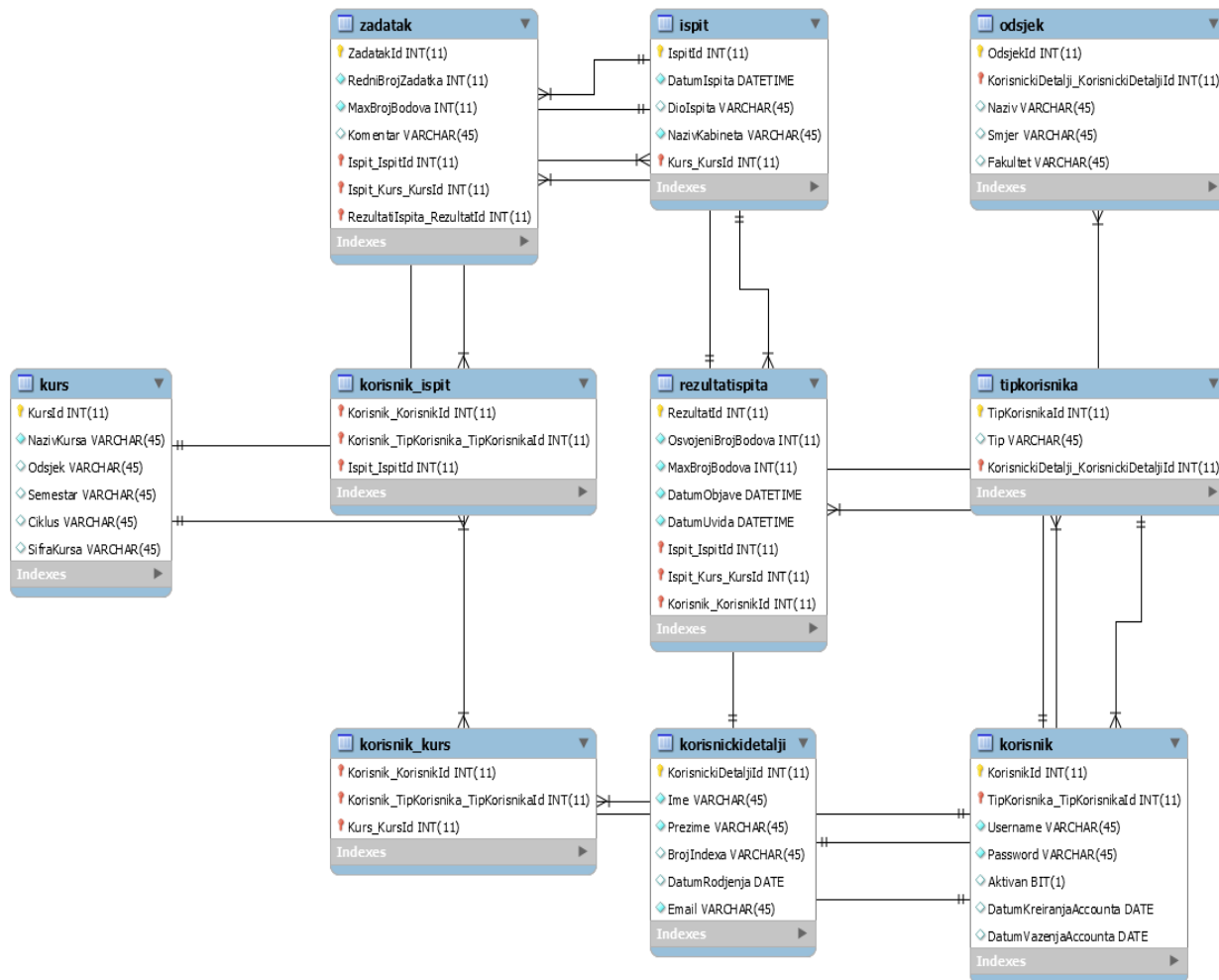
ERA model koristili smo kao metodu konceptualnog i fizičkog modeliranja podataka. Evo i osnovnih koncepata modela entiteta – veze koje smo koristili:

Entitet – je koncept ili objekt od interesa, bilo da je stvarni ili apstraktni predmet odnosno događaj u kojemu se u informacijskom sistemu pamte podaci, sadrži niz atributa

Atribut – govori o identifikaciji, klasifikaciji, kvantifikaciji, kvaliteti ili stanju entiteta, to je jedinstveno obilježje entiteta

Veza – je relacija dva entiteta, one su dvosmjerne i prikazuju međusobne odnose entiteta u oba smjera

Ograničenja – ili cjelobrojnosti veza, nam kazuju koliki broj entiteta jednog tipa može biti u vezi s entitetima drugom tipa.



Baza za projekat Ime\_Projekta se sastoji od 10 tabela:

1. Zadatak
2. Ispit
3. RezultatiIspita
4. Kurs
5. Korisnik\_Ispit
6. Odsjek
7. TipKorisnika
8. KorisnickiDetalji
9. Korisnik
10. Korisnik\_Kurs

**Tabela Zadatak:**



Ime kolone	Primarni ključ	Tip	Nullable	Unique	AutoIncrement	Opis
ZadatakID	DA	Integer	Ne	Da	Da	Primarni ključ
RedniBrojZadatka	NE	Integer	DA			Broj zadatka za neki Ispit
MaxBrojBodova	NE	Float	DA			Max broj bodova po zadatku
Komentar	NE	Varchar(45)	DA			Profesor upisuje dodatni komentar po zadatku
Ispit_IspitID	NE	Integer	Ne	NE	NE	Strani ključ za tabelu <b>Ispit</b>
Ispit_Kurs_KursID	NE	Integer	Ne			Strani ključ za tabelu <b>Kurs</b>
RezultatIspita_RezultatId	Ne	Integer	Ne			Strani ključ za tabelu <b>RezultatIspita</b>

**Tabela Ispit:**

Ime kolone	Primarni ključ	Tip	Nullable	Unique	AutoIncrement	Opis
IspitID	DA	Int	NE	Da	DA	Primarni ključ
DatumIspita	NE	DateTime	Ne	Ne	Ne	Termin za ispit
Diolspita	Ne	Varchar(45)				Koji dio ispita se polaže, (1.parcijala, integralno..)
NazivKabineta	Ne	Varchar(45)	Ne	Ne		Mjesto polaganja ispita
Kurs_KursId	Ne	Int	Ne	Ne		Strani ključ za tabelu <b>Kurs</b>

**Tabela RezultatiIspita:**

Ime kolone	Primarni ključ	Tip	Nullable	Unique	AutoIncrement	Opis
RezultatID	DA	Int	Ne	DA	Da	Primarni ključ
OsvojeniBrojBodova	Ne	Float				Osvojeni broj bodova po zadatku
MaxBrojBodova	Ne	Float	Da	Ne	Ne	Max.broj bodova po Ispitu
DatumObjave	Ne	DateTime	Da	Ne	Ne	Datum objave rezultata ispita
DatumUvida	Ne	DateTime	Da	Ne	Ne	Datum kad student može doći na uvid
Ispit_IspitId	Ne	Int	Ne	Ne	Ne	Strani ključ na tabelu Ispit
Ispit_Kurs_KursId	Ne	Int	Ne	Ne	Ne	Strani ključ na tabelu Kurs
Korisnik_KorisnikId	Ne	Int	Ne	Ne	Ne	Strani ključ na tabelu Korisnik

**Tabela Kurs:**

Ime kolone	Primarni ključ	Tip	Nullable	Unique	AutoIncrement	Opis
KursID	DA	Integer	NE	DA	DA	Primarni ključ
NazivKursa	NE	Varchar(45)	NE	NE		Ime kursa (predmeta)
Odsjek	NE	Varchar(45)				Odsjek na kojem se sluša predmet
Semestar	NE	Varchar(45)				Ljetni/Zimski
Ciklus						Prvi/Drugi
SifraKursa		Varchar(45)	NE	DA	NE	Jedinstvena sifra za kurs

**Tabela Korisnik\_Ispit ( vezna tabela = Korisnik Ispit)**

Ime kolone	Primarni ključ	Tip	Nullable	Unique	AutoIncrement	Opis
Korisnik_KorisnikId	DA	Integer	NE	DA	DA	Primarni ključ
Korisnik_TipKorisnika_TipKorisnikald	NE	Integer	NE	NE		Strani ključ za tabelu <b>TipKorisnika</b>
Ispit_IspitId	NE	Integer	NE			Strani ključ za tabelu Ispit

**Tabela TipKorisnika:**

Ime kolone	Primarni ključ	Tip	Nullable	Unique	AutoIncrement	Opis
TipKorisnikald	DA	Integer	NE	DA	DA	Primarni ključ
Tip	NE	Varchar(45)	NE	NE		Tip korisnika ( student ili profesor )
KorisnickiDetalji_KorisnickiDetaljiId	NE	Varchar(45)	NE			Strani ključ za tabelu <b>KorisnickiDetalji</b>

**Tabela KorisnikKurs ( vezna tabela Korisnik Kurs )**

Ime kolone	Primarni ključ	Tip	Nullable	Unique	AutoIncrement	Opis
Korisnik_KorisnikId	DA	Integer	NE	DA	DA	Primarni ključ
Korisnik_TipKorisnika_TipKorisnikald	NE	Integer	NE	NE		Strani ključ za tabelu <b>TipKorisnika</b>
Kurs_KursId	NE	Integer	NE			Strani ključ za tabelu <b>Kurs</b>

**Tabela Korisnik:**

Ime kolone	Primarni ključ	Tip	Nullable	Unique	AutoIncrement	Opis
KorisnikId	DA	Integer	Ne	Da	Da	Primarni ključ
TipKorisnika_TipKorisnikId	NE	Integer	Ne			Strani ključ za tabelu <b>TipKorisnika</b>
Username	NE	Varchar(45)	Ne	Da		Jedinstveni username korisnika
Password	NE	Varchar(45)	Ne			Sifra korisnika
Aktivan	NE	Bit	Da	NE	NE	Da li je Korisnik aktivan
DatumKreiranjaAccounta	NE	DateTime	Da			Datum trenutnog kreiranja računa
DatumVazenjaAccounta	Ne	DateTime	Da			Vrijeme dokad važi račun

**Tabela KorisnickiDetalji:**

Ime kolone	Primarni ključ	Tip	Nullable	Unique	AutoIncrement	Opis
KorisnickiDetaljId	DA	Integer	Ne	Da	Da	Primarni ključ
Ime	NE	Varchar(45)	Ne			Ime korisnika
Prezime	NE	Varchar(45)	Ne			Prezime korisnika
BrojIndexa	NE	Varchar(45)	Ne	DA		BrojIndexa (samo za studente)
DatumRodjenja	NE	DateTime	Ne	NE	NE	Datum rodjenja korisnika
Email	NE	Varchar(45)	Ne	Da		Email korisnika



**Tabela Odsjek:**

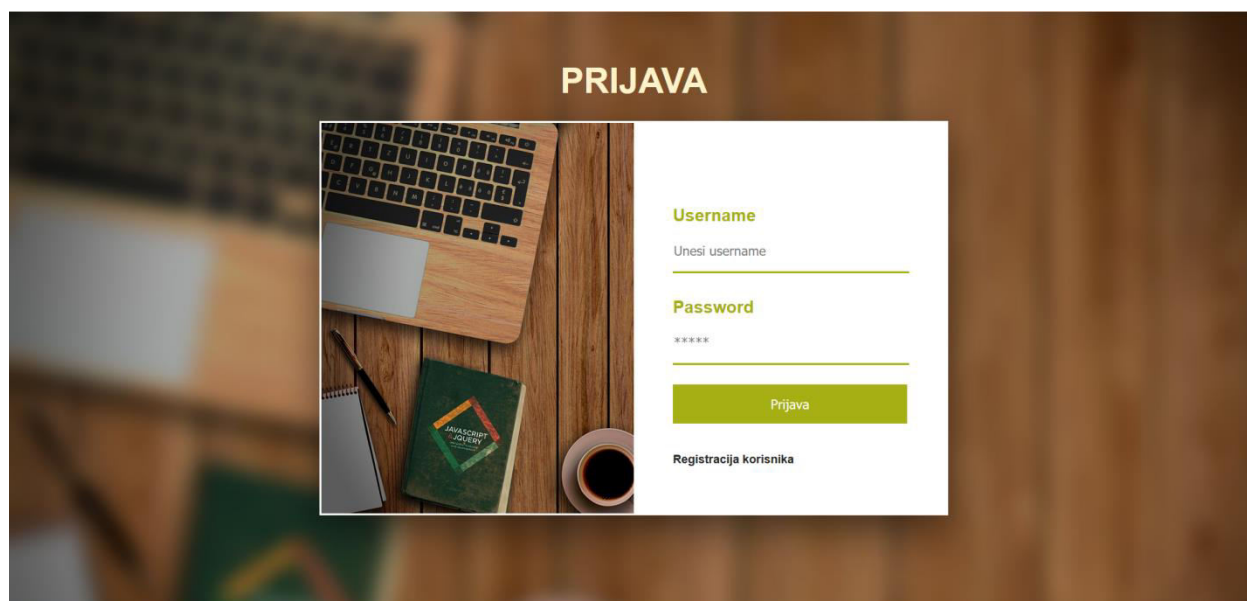
Ime kolone	Primarni ključ	Tip	Nullable	Unique	AutoIncrement	Opis
OdsjekId	DA	Integer	Ne	Da	Da	Primarni ključ
KorisnickiDetalji_KorisniDetaljiId	NE	Integer	Ne			Strani ključ za tabelu <b>KorisnickiDetalji</b>
Naziv	NE	Varchar(45)				Naziv odsjeka
Smjer	NE	Varchar(45)				Naziv smjera
Fakultet	NE	Varchar(45)				Ime ustanove

## 4. KORISNIČKE UPUTE

U ovom dijelu dat ćemo detaljne korisničke upute za korištenje naše aplikacije.

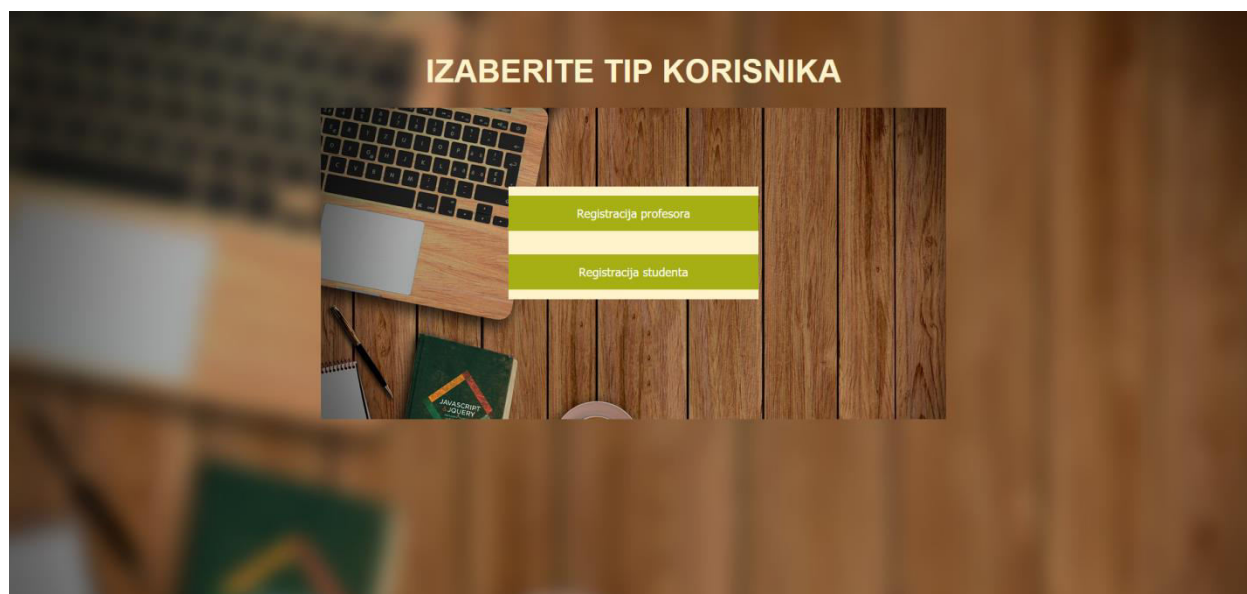
### 4.1. Prijava

Prva stranica koja se otvara je stranica za prijavu korisnika.



Korisnici koji su već registrovani u polja **Username** i **Password** upisuju svoj username i password pomoću kojih se prijavljuju na stranicu.

Ukoliko korisnik nije registrovan, klikom na **Registracija korisnika** otvara se nova stranica koja nudi korisniku da se registruje.



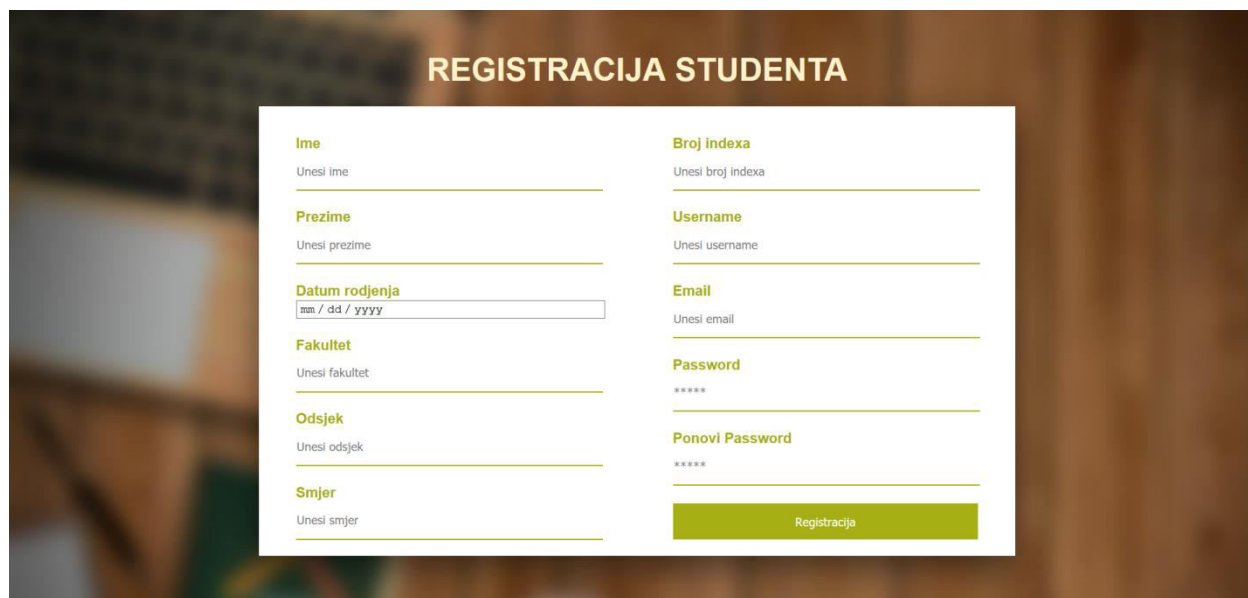
U zavisnosti da li se registruje student ili profesor klikom na određeno dugme otvara se nova stranica koja prikazuje polja za registraciju korisnika.

## 4.2. Registracija

Ukoliko je korisnik profesor, klikom na dugme **Registracija profesora** otvara se stranica za registraciju profesora:

A screenshot of a web application interface for professor registration. The title 'REGISTRACIJA PROFESORA' is centered at the top in a bold, sans-serif font. Below the title is a white rectangular form with a thin border. The form is divided into two columns. The left column contains four input fields with labels: 'Ime' (Unesi ime), 'Prezime' (Unesi prezime), 'Fakultet' (Unesi fakultet), and 'Odsjek' (Unesi odsjek). The right column contains three input fields with labels: 'Username' (Unesi username), 'Email' (Unesi email), and 'Password' (with a 'Ponovi Password' field below it). The password fields are masked with asterisks. At the bottom right of the form is a green button with the text 'Registracija' in white.

Ukoliko je korisnik student, klikom na dugme Registracija studenta otvara se stranica za registraciju studenta:



The image shows a web form titled "REGISTRACIJA STUDENTA" in a bold, yellow font. The form is divided into two columns. The left column contains fields for "Ime" (Name), "Prezime" (Surname), "Datum rođenja" (Date of birth) with a date picker, "Fakultet" (Faculty), "Odsjek" (Department), and "Smjer" (Direction). The right column contains fields for "Broj indexa" (Index number), "Username", "Email", "Password", and "Ponovi Password" (Repeat Password). A green "Registracija" button is at the bottom right. The form is set against a blurred background of a library or study area.

Ime	Broj indexa
Unesi ime	Unesi broj indexa
Prezime	Username
Unesi prezime	Unesi username
Datum rođenja	Email
mm / dd / yyyy	Unesi email
Fakultet	Password
Unesi fakultet	*****
Odsjek	Ponovi Password
Unesi odsjek	*****
Smjer	
Unesi smjer	Registracija

U formi za registraciju korisnik (student ili profesor) unosi podatke i klikom na dugme Registracija, pristupa se bazi gdje se u bazi provjerava da li postoji uneseni Email i Username, kao i Broj indexa u slučaju registracije Studenta. Ukoliko podaci ne postoje u bazi podataka, korisnik se preusmjerava na formu za login, a ukoliko podaci postoje korisniku se ispisuje validacijska greška da se podaci već koriste.

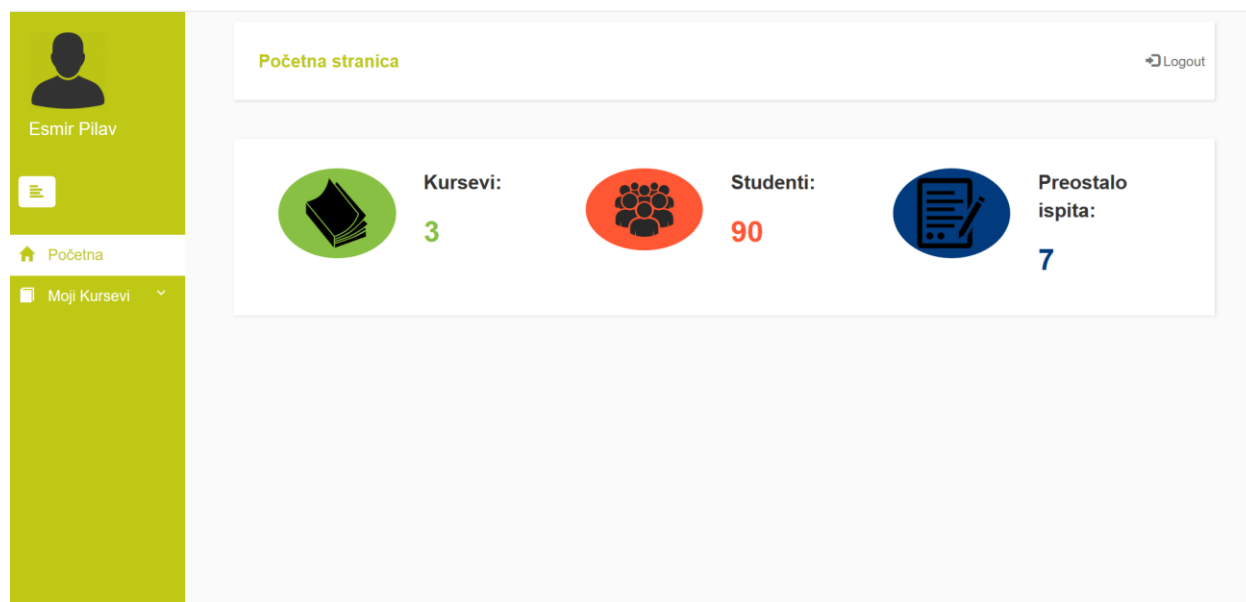
Na formi za login korisnik unosi Username i Password, i pritiskom na dugme Prijava, pristupa se bazi podataka gdje se provjerava da li postoji uneseni username i da li password odgovara unesenom username-u. Ukoliko dobijemo potvrđan odgovor od baze, korisnik se preusmjerava na početnu stranicu za profesora, odnosno studenta u zavisnosti od tipa korisnika. U slučaju negativnog odgovora, korisniku se ispisuje validacijska greška da Username ili Password nisu tačni.

### 4.3. Početna stranica profesora


Na početnoj stranici profesora, nalazi se ukupan broj kurseva koje sam profesor predaje, ukupan broj studenata koji slušaju njegove kurseve te broj preostalih ispita.


U desnom gornjem uglu se nalazi mogućnost odjave.


Sa lijeve strane se nalazi meni u kojem imamo kartice *Moji kursevi* u kojoj se nalaze kartice *Detaljno* te kartice za svaki kurs koji profesor predaje.




Klikom na karticu *Detaljno*, otvara se stranica gdje su prikazani svi kursevi na kojima je predavač prijavljeni profesor.


  
Esmir Pilav



 Početna

 Moji Kursevi

Detaljno

 Logout


Dodaj kurs


Naziv Kursa	Semestar	Ciklus	Šifra kursa
Analiza i sinteza algoritama	3	1	1234
Racunarske mreže	2	1	1111
kurs	2	2	kurs1


Dakle ovdje se nalaze nazivi kurseva prijavljenog profesora uz dodatni opis semestra i ciklusa u kojem profesor predaje kurs te šifra kursa.


U desnom gornjem uglu se nalazi dugme za dodavanje novog kursa.

Klikom na ovo dugme prikazuje se prozor u kojem popunjavamo tražena polja vezana za prijavu novog kursa.


  
Esmir Pilav



 Početna

 Moji Kursevi

Detaljno

 Logout

Dodaj kurs

Naziv Kursa	Semestar	Ciklus	Šifra kursa
Analiza i sinteza			1234
Racunarske mreže			1111
kurs			kurs1

Kreiranje kursa

Kurs

Unesi naziv kursa

Semestar

Unesi semestar

Ciklus

Unesi ciklus


Šifra kursa

Unesi šifru kursa


Kreiraj

Odustani

Klikom na svaki kurs, otvora se nova stranica koja nam prikazuje ispite te mogućnost pregleda i unosa rezultata za odabrani ispit.



Esmir Pilav



Početna

Moji Kursevi

### Analiza i sinteza algoritama


Logout

+ Novi ispit


#### Ispiti

Dio ispita	Mjesto održavanja	Vrijeme održavanja	Dosad prijavljeno	
1	ABG	20/01/2012 u 18:00	2	Rezultati
2	ABG	25/01/2018 u 11:00	2	Rezultati
3	aa	26/01/2018 u 14:14	2	Rezultati

Ukoliko kliknemo na dugme *Rezultati* otvorit će se sljedeći prozor gdje imamo mogućnost za svakog studenta unijeti rezultat ispita, maximalan broj bodova te datum uvida. Rezultati ispita se šalju svakom studentu i to samo njegovi rezultati, dakle student nema informaciju o rezultatima drugih studenata.



Esmir Pilav



Početna

Moji Kursevi

### Ispit iz predmeta Analiza i sinteza algoritama održan datuma 20.01.2012.

Logout

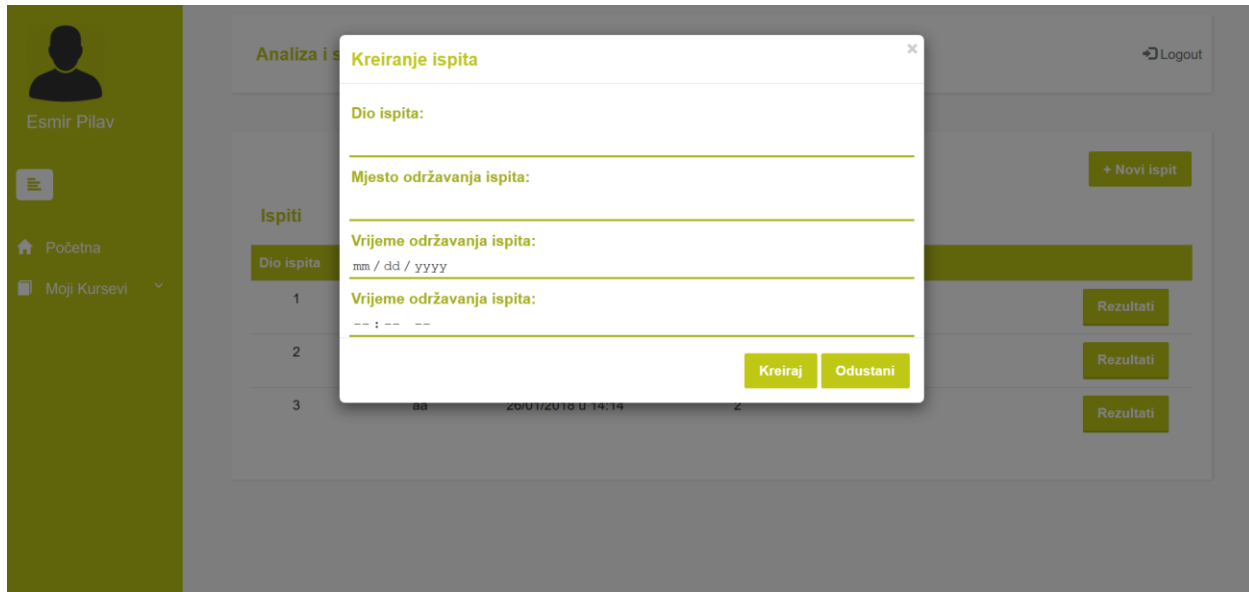
#### Rezultati

Ime i prezime	Broj indexa	Osvojeno bodova
Eldina Maslo	5555	50
Asmir Omeragic	6392	45

Maximalno bodova: 50      Uvid: 25/01/2018 mm / dd / yyyy      **Spasi rezultate**

U desnom gornjem uglu se nalazi dugme za dodavanje novog ispita.

Klikom na ovo dugme otvora se novi prozor gdje popunjavajući tražene podatke profesor kreira ispit za odabrani kurs.

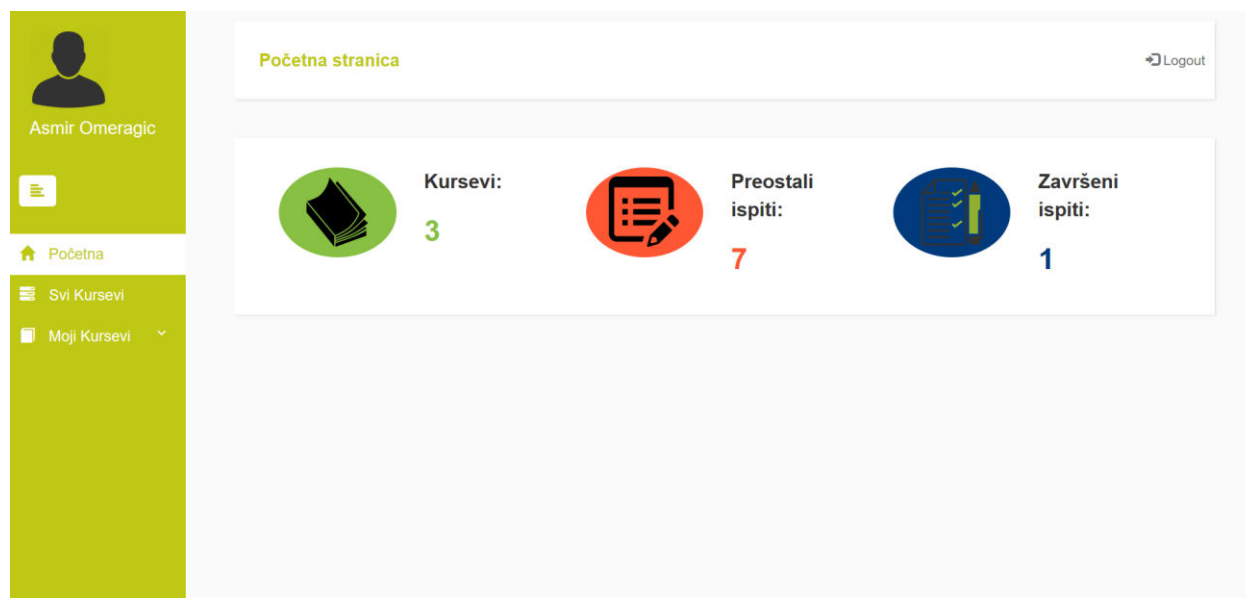




#### 4.4. Početna stranica studenta

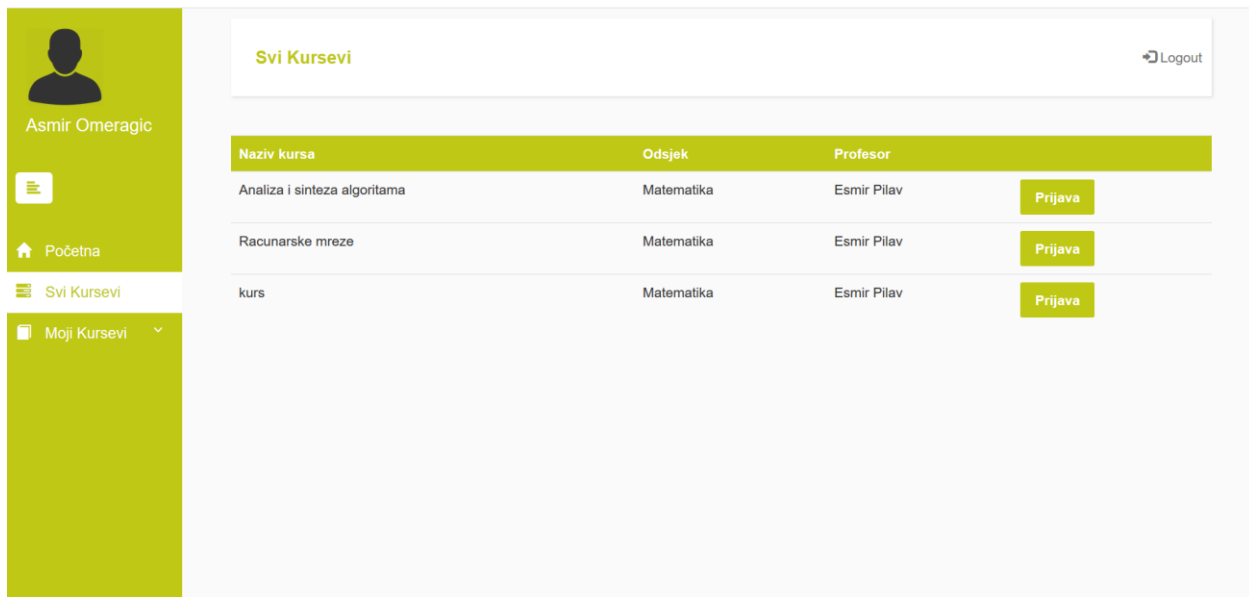
Na početnoj stranici studenta se nalazi ukupan broj kurseva koje student sluša, broj preostalih ispita te broj završenih ispita.

U desnom gornjem uglu se nalazi mogućnost odjave.

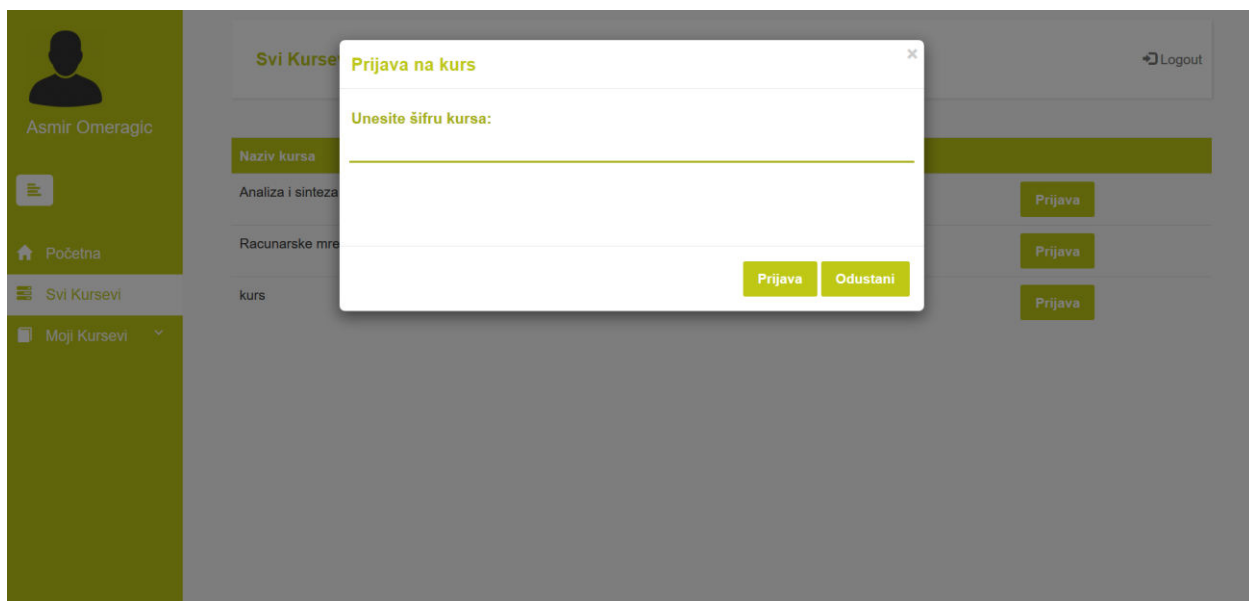


Sa lijeve strane se nalazi meni sa karticama *Svi kursevi* i *Moji kursevi*.

Klikom na karticu *Svi kursevi* otvora se nova stranica sa svim kursevima na koje se student može prijaviti.



Klikom na dugme *Prijavi se* otvora se novi prozor za prijavu na kurs, gdje student treba da unese šifru predmeta i klikom na prijava prijavljuje se na kurs.



Klikom na karticu *Moji kursevi* u meniju se prikazuju svi kursevi na koje je student prijavljen. Klikom na određeni kurs otvara se novi prozor gdje se prikazuju detalji o ispitima odabranog kursa te rezultati ispita.

  
Asmir Omeragic



 Početna

 Svi Kursevi

 Moji Kursevi 

Analiza i sinteza algoritama  Logout

Ispiti

Naziv ispita	Mjesto održavanja	Vrijeme održavanja	
2	ABG	25/01/2018 u 11:00	<a href="#">Prijava</a>
3	aa	26/01/2018 u 14:14	<a href="#">Prijava</a>

Rezultati ispita

Naziv ispita	Datum objave	Datum uvida	Osvojeno bodova	Maksimalan broj bodova
1	22.01.2018	25.01.2018	45	50

Student se može prijaviti na svaki ispit koji je ponuđen za određeni kurs klikom na dugme **Prijava**.

## 5. TESTIRANJE APLIKACIJE

Za pisanje slučajeva za testiranje (Test cases) koristili smo software na internetu koji se naziva TestRail. Nakon toga smo pokrenuli testiranje gdje smo za svaki test napisali da li je prošao ili nije. U nastavku možete vidjeti sve testove, a na kraju kako izgleda izvještaj i uspješnost testiranja aplikacije.

### 5.1. SLUČAJEVI TESTIRANJA (TEST CASES)

#### 1. Registracija profesora - uspješno

##### Preconditions

1. Korisnik mora imati internet konekciju

##### Steps

1. Unijeti ime
2. Unijeti prezime
3. Unijeti fakultet
4. Unijeti odsjek
5. Unijeti username
6. Unijeti e-mail
7. Unijeti password
8. Unijeti potvrdu password-a
9. Pritisnuti dugme Registracija

##### Expected Result

Korisnik je registrovan i preusmjeren na stranicu za prijavu

#### 2. Registracija profesora - e-mail već postoji

##### Preconditions

1. Korisnik mora imati internet konekciju

##### Steps

1. Popuniti input polja
2. Popuniti input polje za e-mail sa već postojećim e-mail-om
3. Kliknuti na dugme Registracija

##### Expected Result

Na stranici će se pojaviti poruka da je uneseni e-mail već postoji

#### 3. Registracija profesora - username već postoji

**Preconditions**

1. Korisnik mora imati internet konekciju

**Steps**

1. Popuniti input polja
2. Popuniti input polje za username sa već postojećim username-om
3. Kliknuti na dugme Registracija

**Expected Result**

Na stranici će se pojaviti poruka da uneseni username već postoji

#### 4. Registracija profesora - prazno input polje

**Preconditions**

1. Korisnik mora imati internet konekciju

**Steps**

1. Ostaviti prazno input polje
2. Kliknuti na dugme Registracija

**Expected Result**

Na stranici kod praznog input polja će se pojaviti poruka - Unesite podatke

#### 5. Registracija profesora - password nije dovoljno dug

**Preconditions**

1. Korisnik mora imati internet konekciju

**Steps**

1. Unijeti u polje za password manje od 6 karaktera

**Expected Result**

Na stranici će se pojaviti poruka da password nije dovoljno dug

## 6. Registracija profesora - password i potvrda passworda se ne poklapaju

### Preconditions

1. Korisnik mora imati internet konekciju

### Steps

1. Unijeti password
2. Unijeti potvrdu passworda koja se razlikuje od unesenog passworda

### Expected Result

Na stranici će se pojaviti poruka da se password i potvrda passworda ne poklapaju

## 7. Registracija studenta - uspješno

### Preconditions

1. Korisnik mora imati internet konekciju

### Steps

1. Unijeti ime
2. Unijeti prezime
3. Unijeti fakultet
4. Unijeti odsjek
5. Unijeti smjer
6. Unijeti username
7. Unijeti broj indeksa
8. Unijeti e-mail
9. Unijeti password
10. Unijeti potvrdu password-a
11. Pritisnuti dugme Registracija

### Expected Result

Korisnik je registrovan i preusmjeren na stranicu za prijavu

## 8. Registracija studenta - e-mail već postoji

### Preconditions

1. Korisnik mora imati internet konekciju

### Steps

1. Popuniti input polja
2. Popuniti input polje za e-mail sa već postojećim e-mail-om
3. Kliknuti na dugme Registracija

### Expected Result

Na stranici će se pojaviti poruka da je uneseni e-mail već postoji

## 9. Registracija studenta - username već postoji

### Preconditions

---

1. Korisnik mora imati internet konekciju

### Steps

---

1. Popuniti input polja
2. Popuniti input polje za username sa već postojećim username-om
3. Kliknuti na dugme Registracija

### Expected Result

---

Na stranici će se pojaviti poruka da uneseni username već postoji

## 10. Registracija studenta - broj indeksa već postoji

### Preconditions

---

1. Korisnik mora imati internet konekciju

### Steps

---

1. Popuniti input polja
2. Popuniti input polje za broj indeksa sa već postojećim brojem indeksa
3. Kliknuti na dugme Registracija

### Expected Result

---

Na stranici će se pojaviti poruka da uneseni broj indeksa već postoji

## 11. Registracija studenta - prazno input polje

### Preconditions

---

1. Korisnik mora imati internet konekciju

### Steps

---

1. Ostaviti prazno input polje
2. Kliknuti na dugme Registracija

### Expected Result

---

Na stranici kod praznog input polja će se pojaviti poruka - Unesite podatke

## 12. Registracija studenta - password nije dovoljno dug

### Preconditions

1. Korisnik mora imati internet konekciju

### Steps

1. Unijeti u polje za password manje od 6 karaktera

### Expected Result

Na stranici će se pojaviti poruka da password nije dovoljno dug

## 13. Registracija studenta - password i potvrda passworda se ne poklapaju

### Preconditions

1. Korisnik mora imati internet konekciju

### Steps

1. Unijeti password
2. Unijeti potvrdu passworda koja se razlikuje od unesenog passworda

### Expected Result

Na stranici će se pojaviti poruka da se password i potvrda passworda ne poklapaju

## 14. Prijava - uspješno

### Preconditions

1. Korisnik mora imati internet konekciju
2. Korisnik mora biti registrovan

### Steps

1. Unijeti Username
2. Unijeti Password
3. Kliknuti na dugme prijava

### Expected Result

Korisnik je preusmjeren na početnu stranicu od profesora/studenta u zavisnosti od toga kako je registrovan (profesor ili student)



## 15. Prijava - pogrešan username ili password

### Preconditions

---

1. Korisnik mora imati internet konekciju

### Steps

---

1. Unijeti username ili password koji nisu registrovani
2. Kliknuti na dugme prijava

### Expected Result

---

Na stranici će se pojaviti poruka "Pogresan username ili password"

## 16. Prijava - prazno input polje

### Preconditions

---

Korisnik mora imati internet konekciju

### Steps

---

1. Ostaviti prazno input polje
2. Kliknuti na dugme Prijava

### Expected Result

---

Na stranici će se pojaviti kod praznog input polja poruka - Unesite podatke

## 17. Kreiranje kursa - uspješno

### Preconditions

---

1. Korisnik mora biti prijavljen
2. Korisnik mora biti na stranici "Moji kursevi"
3. Korisnik mora kliknuti dugme "Dodaj kurs"

### Steps

---

1. Unijeti naziv kursa
2. Unijeti broj semestra
3. Unijeti broj ciklusa
4. Unijeti šifru kursa
5. Kliknut dugme Kreiraj

### Expected Result

---

- Na stranici će se pojaviti poruka da je kurs uspješno kreiran
- Korisnik će biti preusmjeren na stranicu Moji kursevi gdje će biti i novokreirani kurs

## 18. Kreiranje kursa - prazna input polja

### Preconditions

---

1. Korisnik mora imati internet konekciju
2. Korisnik mora biti prijavljen
3. Korisnik mora kliknuti na dugme Dodaj kurs

### Steps

---

1. Ostaviti neko od polja prazno
2. Kliknuti na dugme Kreiraj

### Expected Result

---

Na stranici će se kod praznog input polja pojaviti poruka da unesemo podatke

## 19. Prijava na kurs - uspješno

### Preconditions

---

1. Korisnik mora imati internet konekciju
2. Korisnik mora biti prijavljen
3. Korisnik mora kliknuti na dugme Prijava

### Steps

---

1. Unijeti šifru kursa
2. Kliknuti na dugme Prijavi

### Expected Result

---

Na stranici će se pojaviti poruka da smo uspješno prijavljeni na kurs  
U opadajućem meniju pojaviće se kurs na koji smo se prijavili

## 20. Prijava na kurs - pogrešna šifra

### Preconditions

---

1. Korisnik mora imati internet konekciju
2. Korisnik mora biti prijavljen
3. Korisnik mora kliknuti na dugme Prijava

### Steps

---

1. Unijeti pogrešnu šifru
2. Kliknuti na dugme Prijavi

### Expected Result

---

Na stranici će se pojaviti poruka "Pogriješili ste šifru. Pokušajte ponovo"

## 21. Prijava na kurs - već smo prijavljeni na kurs

### Preconditions

1. Korisnik mora imati internet konekciju
2. Korisnik mora biti prijavljen
3. Korisnik mora biti prijavljen na kurs
4. Korisnik mora kliknuti na dugme prijava

### Steps

1. Unijeti šifru kursa
2. Kliknuti na dugme prijava

### Expected Result

Na stranici će se pojaviti poruka da smo već prijavljeni na kurs

## 22. Prijava na kurs - prazno input polje

### Preconditions

1. Korisnik mora imati internet konekciju
2. Korisnik mora biti prijavljen
3. Korisnik mora kliknuti na dugme prijava

### Steps

1. Ostaviti prazno input polje
2. Kliknuti na dugme Prijavi

### Expected Result

Na stranici će se pored praznog input polja pojaviti poruka da unesemo podatke

## 23. Kreiranje ispita - uspješno

### Preconditions

1. Korisnik mora imati internet konekciju
2. Korisnik mora biti prijavljen
3. Korisnik mora imati kreiran bar jedan kurs
4. Korisnik mora biti na stranici kursa za koji želi kreirati ispit
5. Korisnik mora kliknuti na dugme Novi ispit

### Steps

1. Unijeti Dio ispita
2. Unijeti Mjesto održavanja ispita
3. Unijeti Datum održavanja ispita
4. Unijeti Mjesto održavanja ispita
5. Kliknuti na dugme Kreiraj

### Expected Result

- Na stranici će se pojaviti poruka da smo uspješno kreirali ispit
- Korisnik će ostati na stranici trenutnog kursa i imaće u tabeli kreirani ispit

## 24. Kreiranje ispita - prazna input polja

### Preconditions

1. Korisnik mora imati internet konekciju
2. Korisnik mora biti prijavljen
3. Korisnik mora imati bar jedan kreirani kurs
4. Korisnik mora biti na stranici kursa
5. Korisnik mora kliknuti na dugme Novi ispit

### Steps

1. Ostaviti prazno input polje
2. Kliknuti na dugme kreiraj

### Expected Result

Na stranici će se pored praznog input polja pojaviti poruka da unesemo podatke

## 25. Kreiranje rezultata - uspješno

### Preconditions

1. Korisnik mora imati internet konekciju
2. Korisnik mora biti prijavljen
3. Korisnik mora imati kreiran bar jedan ispit sa prijavljenim studentima
4. Korisnik mora kliknuti na dugme Rezultati

### Steps

1. Unijeti osvojene bodove za svakog studenta u tabeli
2. Unijeti maksimalan broj bodova
3. Unijeti datum uvida
4. Kliknuti na dugme Spasi rezultate

### Expected Result

Korisnik će ostati na stranici za dodavanje rezultata  
Biti će poslan e-mail svakom studentu sa njegovim osvojenim bodovima, koji su maksimalni bodovi i kada je datum uvida  
Na stranici (student) odgovarajućeg kursa biti će prikazani njegovi rezultati na određenom ispitu

## 26. Kreiranje rezultata - prazno input polje

### Preconditions

1. Korisnik mora imati internet konekciju
2. Korisnik mora biti prijavljen
3. Korisnik mora imati kreiran bar jedan ispit sa prijavljenim studentima
4. Korisnik mora kliknuti na dugme rezultati

### Steps

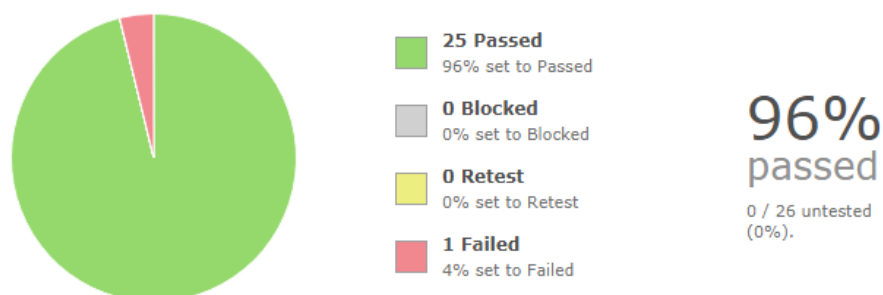
1. Ostaviti prazno input polje
2. Kliknuti na dugme Spasi rezultate

### Expected Result

Na stranici će se pojaviti poruka da moramo unijeti sve podatke

## 5.2. IZVJEŠTAJ TESTIRANJA

Na sljedećem grafikonu možemo vidjeti uspješnost same aplikacije. Vidimo da je 96% aplikacije, tj. 25 testova uspješno, dok je 4%, tj. 1 test neuspješan, od ukupno 26 testova.



Sada ćemo vidjeti koji su to testovi prošli a koji ne.

### Registracija profesora (6)

ID	Title	Assigned To	Status
T1	Registracija profesora - uspješno		Passed
T2	Registracija profesora - e-mail već postoji		Passed
T3	Registracija profesora - username već postoji		Passed
T4	Registracija profesora - prazno input polje		Passed
T5	Registracija profesora - password nije dovoljno dug		Passed
T6	Registracija profesora - password i potvrda passworda se ne poklapaju		Passed

### Registracija studenta (7)

ID	Title	Assigned To	Status
T7	Registracija studenta - uspješno		Passed
T8	Registracija studenta - e-mail već postoji		Passed
T9	Registracija studenta - username već postoji		Passed
T10	Registracija studenta - broj indeksa već postoji		Passed
T11	Registracija studenta - prazno input polje		Passed
T12	Registracija studenta - password nije dovoljno dug		Passed
T13	Registracija studenta - password i potvrda passworda se ne poklapaju		Passed

### Prijava (3)

ID	Title	Assigned To	Status
T14	Prijava - uspješno		Passed
T15	Prijava - pogresan username ili password		Passed
T16	Prijava - prazna input polja		Passed

#### Kreiranje kursa (2)

<input type="checkbox"/>	ID	Title	Assigned To	Status	
<input type="checkbox"/>	T17	Kreiranje kursa - uspješno		Passed ▾	
<input type="checkbox"/>	T19	Kreiranje kursa - prazna input polja		Passed ▾	

#### Prijava na kurs (4)

<input type="checkbox"/>	ID	Title	Assigned To	Status	
<input type="checkbox"/>	T20	Prijava na kurs - uspješno		Passed ▾	
<input type="checkbox"/>	T32	Prijava na kurs - pogrešna šifra		Passed ▾	
<input type="checkbox"/>	T33	Prijava na kurs - već smo prijavljeni na kurs		Passed ▾	
<input type="checkbox"/>	T21	Prijava na kurs - prazno input polje		Passed ▾	

#### Kreiranje ispita (2)

<input type="checkbox"/>	ID	Title	Assigned To	Status	
<input type="checkbox"/>	T22	Kreiranje ispita - uspješno		Passed ▾	
<input type="checkbox"/>	T23	Kreiranje ispita - prazna input polja		Passed ▾	

#### Kreiranje rezultata ispita (2)

<input type="checkbox"/>	ID	Title	Assigned To	Status	
<input type="checkbox"/>	T24	Kreiranje rezultata ispita - uspješno		Passed ▾	
<input type="checkbox"/>	T25	Kreiranje rezultat ispita - prazno input polje		Failed ▾	