

Programsko Inženjerstvo

Ak. god. 2022./2023.

Studensis

Dokumentacija

Grupa: *Studensis*

Voditelj: *Matija Fućek*

Datum predaje: *12.12.2022.*

Nastavnik: *Vlado Sruk*

Sadržaj

O timu Studensis

Okupljena je ekipa od sedmero visoko motivirana člana, od kojih svaki doprinosi svojim usmjerenjem tako da pokrivamo cjeloukupan stack potreban za razvoj platforme.

1. Dnevnik promjena dokumentacije

2. Opis projektnog zadatka

Projekt "Studensis" je osmišljen kao platforma koja objedinjuje set alata potrebnih za rad proizvoljne obrazovne ustanove, prvenstveno fakulteta, na velikom broju studenata, kolegija i posebnih programa.

Cilj projekta je kontribuirati boljem iskustvu svakog studenta i profesora diljem Sveučilišta. Posebno smo motivirani da omogućimo fakultetima pored FER-a kvalitetnije rješenje za upravljanje kolegijima i pružimo bolje načine komunikacije profesora i studenata.

Komponente sustava

⚠ Crvenom pozadinom su dijelomično navedene značajke i funkcionalnosti koje nebi ulazile u opseg projekta za Programsko Inženjerstvo.

Alat za upravljanje korisnicima

Poseban alat koji bi pretežno djelovao automatski, ali i uz mogućnost ručnih izmjena je alat za upravljanje korisnicima na razini samog sustava.

Alat za upravljanje kolegijima

Neregistrirani korisnici bi imali uvid u javne informacije za svaki kolegij kao što su opis predmeta, način bodovanja, popis profesora, asistenata i ostalih službenih sudionika.

Registrirani korisnici, kojima su dodijeljena prava dubljeg pregleda u kolegij, bi mogli prolaziti kroz sam sadržaj kolegija u obliku lekcija, materijala, te kvizova i zadataka za vježbu.

Posebno za svaki kolegij određuju se registrirani korisnici koji bi dobili ulogu Editora. Svaki editor bi mogao mijenjati javno dostupne stranice, te sam sadržaj lekcija unutar kolegija. Pored sadržaja, editori (ili potencijalno još neka visa rola) bi mogli mijenjati i samu listu registriranih korisnika kojima je dano pravo dubljeg pregleda.

Sustav za sadržaj

Od profila korisnika do sadržaja lekcija unutar kolegija, sustav za sadržaj pobrinuo bi se da se diljem stranice sadržaj organizira na što pregledniji način, te bi olakšavao i pregled i uređivanje, naravno sa širokim mogućnostima proširenja kroz ekstenzije.

3. Specifikacija programske potpore

Neregistriranom korisniku prikazuje se naslovna stranica s osnovnim informacijama vezanim za sustav radnog imena Studensis. Neregistrirani korisnik može se prijaviti u sustav sa svojim korisničkim podacima.

Registracijom u sustav korisnik dobiva razinu prava određenu ulogom u sustavu. Postojat će uloge administratora, fakultetskog osoblja (nositelja kolegija, profesora te asistenata), upisanih studenata, posjetitelja I razvojnog tima.

Svaka od ovih uloga daje pristup određenim funkcionalnostima I informacijama na sustavu.

Student prijavom u sustav dobiva prikaz glavne stranice. Na njoj su mu vidljivi njegovi predmeti za koje za svaki piše skupljeni broj bodova. Piše i lista položenih predmeta te prosjek ocjena koji je student ostvario u obrazovnoj ustanovi. Nadalje ima kratki prikaz nadolazećih obaveza, a klikom na tu sekciju, otvara se stranica sa svim obavezama, koje također imaju svaka svoju stranicu. Također ima vidljiv jedan manji kalendar, koje mu je inicijalno stanje da prikazuje dnevni kalendar, ali postoji mogućnost podešavanja na veće vremensko razdoblje. Klikom na kalendar otvara se zasebna stranica s kalendarom, s inicijalnim stanjem pregleda trenutnog tjedna. Također moguće su opcije podešavanja na veće i manje razdoblje. U kalendaru su vidljive sve nadolazeće obaveze, koje nude i daljnje informacije. Student također ima mogućnost dodavanja vlastitih zapisa u kalendar. Nadalje, na glavnoj stranici ispod manje verzije kalendara, vidljiv je popis upisanih predmeta. Klikom na neki od njih, otvara se stranica predmeta. Student ovdje može vidjeti osnovne informacije o predmetu te ima pristup materijalima i svojim rezultatima na predmetu. Dostupni materijali su mu prenesene prezentacije, videi i sl. Također na našoj platformi omogućili bismo pristup interaktivnim lekcijama. Njih kreira profesor i to omogućuje studentima bolje usavršavanje gradiva, a profesorima opciju da temeljem rezultata daje određeni broj bodova. Slično, ali ne isto, napravili bismo odjeljak s funkcionalnostima sličnim aplikaciji Kahoot koja bi se mogla koristiti na predavanjima uživo, sa svrhom sličnom interaktivnim lekcijama. Studentu su vidljive obaveze vezane za otvoreni predmet. Na glavnoj stranici, ispod popisa predmeta, nalaze se najnovije novosti i obavijesti o fakultetu. Također će student imati pristup detaljnijim obavijestima koje može filtrirati ili se na njih pretplaćivati.

Fakultetsko osoblje dijeli se na nositelje predmeta, predavače te asistente koji imaju ovisno o tome manja ili veća prava. Prijavom u sustav, osoblje također dobiva prikaz glavne stranice, drugačije od one kakvu vidi student. Vidljiv im je dnevni kalendar, kojem, kao i studenti, mogu podesiti vremenski interval na veći. U njemu su vidljive nadolazeće obaveze. Klikom na kalendar otvara se zasebna stranica s većim, ali također podesivim, prikazom. Ovdje osoblje ima mogućnost unosa svojih zapisa, ali također i zapisa kojima mogu podesiti vidljivost. Recimo, žele održati konzultacije u određeno vrijeme za određenu grupu studenata. Na glavnoj stranici, ispod kalendara, nalazi se popis predmeta na kojima sudjeluju. Na njima imaju različite uloge. Za svaki predmet, postoji zasebna stranica. Na njoj imaju pristup listi studenata upisanih na odabrani predmet.

Daje im se mogućnost unosa bodova. Na ovoj stranici nalazit će se QR kod, koji će studenti moći skenirati te se na taj način može evidentirati dolaznost na predavanja. Naravno, imat će mogućnost i oduzimanja bodova za prisutnost u slučaju da primjete da je student napustio predavanje.

Za svaki predmet, postojat će repozitorij materijala, koji se mogu dodavati, brisati te uređivati. Daljim razvojem servisa, omogućili bismo neke funkcionalnosti inspirirane alatom Notion. Nudili bismo prijepis materijala na način da ga profesor može uređivati, a studenti mogu na svaki odlomak ostavljati komentare, te na taj način dobivati povratne informacije od profesora ili od drugih studenata. Time bismo omogućili bolju i

lakšu komunikaciju, a time i usvajanje gradiva. Osim materijala, postojao bi pristup kreaciji, brisanju i uređivanju interaktivnih lekcija te ranije navedenim funkcionalnostima sličnim Kahootu. Osoblje ima pristup popisu održavanih provjera na predmetu, te unos bodova za svaku od njih. Nadalje, mogu slati obavijesti vezane za predmet. Ispod popisa predmeta na glavnoj stranici, osoblju se prikazuju obavijesti, slično kao i studentima.

Ulogu posjetitelja dobivaju studenti upisani na fakultet, ali oni koji nisu upisani na određeni predmet. Njima se na stranici tog predmeta prikazuju osnovne informacije o predmetu.

Administrator ima najviša prava, te ih dodjeljuje drugim ulogama.

Studenti i osoblje imaju pristup funkcionalnosti chata, putem kojeg mogu komunicirati s ostalim osobljem ili studentima.

Funkcionalni zahtjevi

Ostali zahtjevi

- Sustav treba omogućiti rad više korisnika u stvarnom vremenu
- Korisničko sučelje i sustav moraju podržavati hrvatsku abecedu (dijakritičke znakove) pri unosu i prikazu tekstualnog sadržaja
- Izvršavanje dijela programa u kojem se pristupa bazi podataka ne smije trajati duže od nekoliko sekundi
- Sustav treba biti implementiran kao web aplikacija koristeći objektno-orijentirane jezike
- Neispravno korištenje korisničkog sučelja ne smije narušiti funkcionalnost i rad sustava
- Sustav treba biti jednostavan za korištenje
- Nadogradnja sustava ne smije narušavati postojeće funkcionalnosti sustava
- Veza s bazom podataka mora biti otporna na vanjske greške
- Pristup sustavu mora biti omogućen iz javne mreže pomoću HTTPS.

Obrasci uporabe

Glavni sudionik: Klijent

- cilj: Pristupiti korisničkom računu
- sudionici: Baza podataka, Google
- preduvjet: –
- opis osnovnog tijeka:
 1. Klijent ode na stranicu Studius i klikne na "prijava"
 2. Unese svoje korisničko ime i lozinku
 3. Ulogira se na Studius
 4. Opis mogućih odstupanja: Uneseni podaci ne postoje u bazi

UC2 Registracija putem linka

- Glavni sudionik: Klijent
- cilj: Stvoriti korisnički račun
- sudionici: Baza podataka, Google
- preduvjet: –

- opis osnovnog tijeka:
 1. Klijent ode na stranicu Studius i klikne "registracija"
 2. Unese ime, prezime, jmbag, email i lozinku
 3. Podaci se spremaju u bazu i stvara se novi korisnički račun

UC3 Pregledati javne detalje o kolegiju

- Glavni sudionik: Klijent
- cilj: Pregledati javne podatke o željenom kolegiju
- sudionici: Baza podataka
- preduvjet: -
- opis osnovnog tijeka:
 1. Klijent ode na stranicu Studius i zatraži jedan od kolegija
 2. Klikne na taj kolegij
 3. Prikažu mu se puno ime kolegija, njegov opis, način bodovanja i opterećenje (ECTS)
- Glavni sudionik: Klijent
- cilj: Pregledati globalne obavijesti
- sudionici: Baza podataka
- preduvjet: -
- opis osnovnog tijeka:
 1. Klijent ode na stranicu Studius i klikne na "obavijesti"
 2. Prikaže mu se lista obavijesti (od najnovijih prema starijima)

UC5 Pregledati popis oglasa svih nadolazećih predloženih Seminara

- Glavni sudionik: Klijent
- cilj: Pregledati oglase nadolazećih seminara
- sudionici: Baza podataka
- preduvjet: -
- opis osnovnog tijeka:
 1. Klijent ode na stranicu Studius i klikne na "nadolazeći seminari"
 2. Prikaže mu se lista nadolazećih seminara

UC6 Uređivanje opisa

- Glavni sudionik: Registrirani korisnik
- cilj: Urediti svoj opis
- sudionici: Baza podataka
- preduvjet: Korisnik ima svoj račun i ulogiran je u sustav
- opis osnovnog tijeka:
 1. Korisnik klikne na "uredi svoj opis"
 2. Zapiše svoje titule, radove, područja interesa
- Glavni sudionik: Registrirani korisnik
- cilj: Provjeriti javne detalje o željenom korisniku
- sudionici: Baza podataka
- preduvjet: Korisnik ima svoj račun i ulogiran je u sustav
- opis osnovnog tijeka:
 1. Korisnik klikne na tražilicu za korisnike

2. Pretraži željenog korisnik tako što utipka njegovo ime i prezime

UC8 Najava dolaska na seminar

- Glavni sudionik: Registrirani korisnik
- cilj: Najaviti dolazak na seminar u svrhu prebrojavanja ljudi i napomeni o premašenom
- kapacitetu dvorane
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav
- opis osnovnog tijeka:
 1. Korisnik pregleda oglase javnih seminara
 2. Zabilježi da će doći na željeni seminar

UC9 Pregledati popis korisnika na predmetima koje predaje

- Glavni sudionik: Profesor
- cilj: Pregledati sve korisnike na predmetu na kojem predaje
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom profesora
- opis osnovnog tijeka:
 1. Profesor odabere jedan od predmeta na kojem predaje
 2. Dobije pregled svih studenata upisanih na taj predmet
- Glavni sudionik: Profesor
- cilj: Obaviještavati studente s događanjima relevantnih za predmet
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom profesora
- opis osnovnog tijeka:
 1. Profesor odabere jedan od predmeta na kojem predaje
 2. Postavi obavijest relevantnu za taj predmet

UC11 Evidencija dolaska

- Glavni sudionik: Profesor
- cilj: Evidentirati dolazak doktorandima kako bi mogli uspješno položiti predmet
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom profesora
- opis osnovnog tijeka:
 1. Profesor na seminaru da listić za potpisivanje doktorandima
 2. Zapisana imena zabilježi u sustav

UC12 Odobravanje prijavljenog seminara

- Glavni sudionik: Profesor, doktorand
- cilj: Odobriti ili odbiti prijavljeni seminar ovisno je li relevantan za upisan predmet
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom profesora
- opis osnovnog tijeka:
 1. Doktorand prijavi da želi održati određen seminar kako bi mogao uspješno položiti predmet
 2. Profesor može odobriti ili odbiti ovisno je li relevantan za taj predmet

- Glavni sudionik: Profesor
- cilj: Pregledati statistiku
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom nositelja
- opis osnovnog tijeka:
 1. Profesor pregleda listu predmeta na kojima predaje
 2. Dobije pregled svih statistika od svih godina

UC14 Najava i prethodno izlaganje radova za konferencije (generalna proba)

- Glavni sudionik: Profesor, doktorand
- cilj: Napraviti probu seminara prije službenog izlaganja
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom nositelja
- opis osnovnog tijeka:
 1. Nositelj se dogovori s doktorandom za termin s određenim datumom vremenom i mjestom
 2. Termin se zabilježi u bazu
 3. Održe probu

UC15 Upisati dostupni kolegij

- Glavni sudionik: Student
- cilj: Upisati kolegije koje može
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom studenta
- opis osnovnog tijeka:
 1. Student ode na aplikaciju za upis
 2. Vidi listu ponuđenih predmeta
 3. Odabere predmete koje želi upisati (ako za neke predmete nema preduvjete sustav neće dopustiti odabir)
- Glavni sudionik: Student
- cilj: Pregledati popis kolegija na koje je upisan
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom studenta
- opis osnovnog tijeka:
 1. Student kad je ulogiran vidi listu aktivnih predmeta

UC17 Detalji o kolegiju

- Glavni sudionik: Student
- cilj: Pregledati detalje o kolegiju
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom studenta
- opis osnovnog tijeka:
 1. Student ode na listu aktivnih predmeta
 2. Odabere željeni predmet
 3. Vidi neke podatke koje registrirani korisnik ne vidi, svoji rezultati na predmetu, uvjeti polaganja i statistike ispita ako su objavljene

UC18 Pregledati obavijesti vezanih uz kolegije na koje je povezan

- Glavni sudionik: Student
- cilj: Pregledati obavijesti na aktivnim kolegijima
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom studenta
- opis osnovnog tijeka:
 1. Student ode na listu aktivnih kolegija
 2. Odabere željeni predmet
 3. Pregleda obavijesti na predmetu
- Glavni sudionik: Studentska služba
- cilj: Pregledati podatke iz baze
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom člana studentske službe
- opis osnovnog tijeka:
 1. Član studentske službe pregleda željene podatke iz baze

UC20 Pisanje u bazu

- Glavni sudionik: Studentska služba
- cilj: Zapisati podatke u bazu
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom člana studentske službe
- opis osnovnog tijeka:
 1. Član studentske službe zapiše željene podatke u bazu

UC21 Izmjena u bazi

- Glavni sudionik: Studentska služba
- cilj: Izmjeniti podatke u bazi
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom člana studentske službe
- opis osnovnog tijeka:
 1. Član studentske službe izmjeni željene podatke u bazi

UC22 Brisanje u bazi

- Glavni sudionik: Studentska služba
- cilj: Izbrisati podatke iz baze
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom člana studentske službe
- opis osnovnog tijeka:
 1. Član studentske službe izbriše željene podatke iz baze
- Glavni sudionik: Studentska služba
- cilj: Kreirati linkove za upis
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom člana studentske službe
- opis osnovnog tijeka:
 1. Član studentske službe kreira linkove za upis predmeta ili studija

UC24 Kreirati kolegij

- Glavni sudionik: Studentska služba
- cilj: Kreirati kolegij
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom člana studentske službe
- opis osnovnog tijeka:
 1. Član studentske službe kreira kolegij

UC25 Odobravanje kolegija

- Glavni sudionik: Studentska služba
- cilj: Odobravanje kolegija
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom člana studentske službe
- opis osnovnog tijeka:
 1. Član studentske službe potvrdi studentu upis na kolegij

UC26 Postavljanje globalnih obavijesti

- Glavni sudionik: Studentska služba
- cilj: Postavljanje globalnih obavijesti
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom člana studentske službe
- opis osnovnog tijeka:
 1. Postavljanje obavijesti relevantne za cijeli fakultet

UC27 Kvalifikacijski ispiti za studente

- Glavni sudionik: Studentska služba
- cilj: Odobriti kvalifikacijske ispite studentima
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom člana studentske službe
- opis osnovnog tijeka:
 1. Član studentske službe pregleda zahtjev za kvalifikacijski ispit
 2. Odobri ga ili odbije

UC28 Razgovor o temi i doprinosu znanstvenog rada

- Glavni sudionik: Studentska služba
- cilj: Porazgovarati o temi i doprinosu znanstvenog rada sa doktorandom
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom člana studentske službe
- opis osnovnog tijeka:
 1. Član studentske službe se dogovori sa doktorandom termin za razgovor
 2. Zabilježe datum, vrijeme i mjesto održavanja

UC29 Postavljanje rola korisnika na kolegiju

- Glavni sudionik: Nositelj

- cilj: Postaviti role korisnicima na kolegiju
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom nositelja
- opis osnovnog tijeka:
 1. Nositelj odredi profesore i asistente što će predavati na predmetu i držati laboratorijske vježbe i organizira rad

UC30 Postavljanje uvjeta na kolegiju

- Glavni sudionik: Nositelj
- cilj: Postaviti uvjete polaganja na kolegiju
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom nositelja
- opis osnovnog tijeka:
 1. Profesor odabere kolegij s liste na kojima ima nositeljska prava
 2. Promjeni uvjete za izlazak na ispit
- Glavni sudionik: Nositelj
- cilj: Postaviti materijale na kolegiju
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom nositelja
- opis osnovnog tijeka:
 1. Profesor odabere kolegij s liste na kojima ima nositeljska prava
 2. Postavi materijale za učenje studentima

UC32 Postaviti kolegij je li dostupan za upisivanje

- Glavni sudionik: Nositelj
- cilj: Postaviti kolegij je li dostupan za upisivanje
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom nositelja
- opis osnovnog tijeka:
 1. Profesor odabere kolegij s liste na kojima ima nositeljska prava
 2. Dopusti izvanredno upisivanje nekolicini ili svim studentima ako je potrebno

UC33 Urediti opis kolegija na kojima ima nositeljska prava

- Glavni sudionik: Nositelj
- cilj: Urediti opis predmeta na kojima je nositelj
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom nositelja
- opis osnovnog tijeka:
 1. Profesor odabere jedan predmet s liste na kojima ima nositeljska prava
 2. Uredi opis ako želi

UC34 Prikaz i rasprava istraživačkih postignuća (novoobjavljeni radovi vlastiti i tuđi)

- Glavni sudionik: Nositelj
- cilj: Prikazati i raspraviti o istraživačkim postignućima novoobjavljenih radova
- sudionici: Baza podataka, doktorand

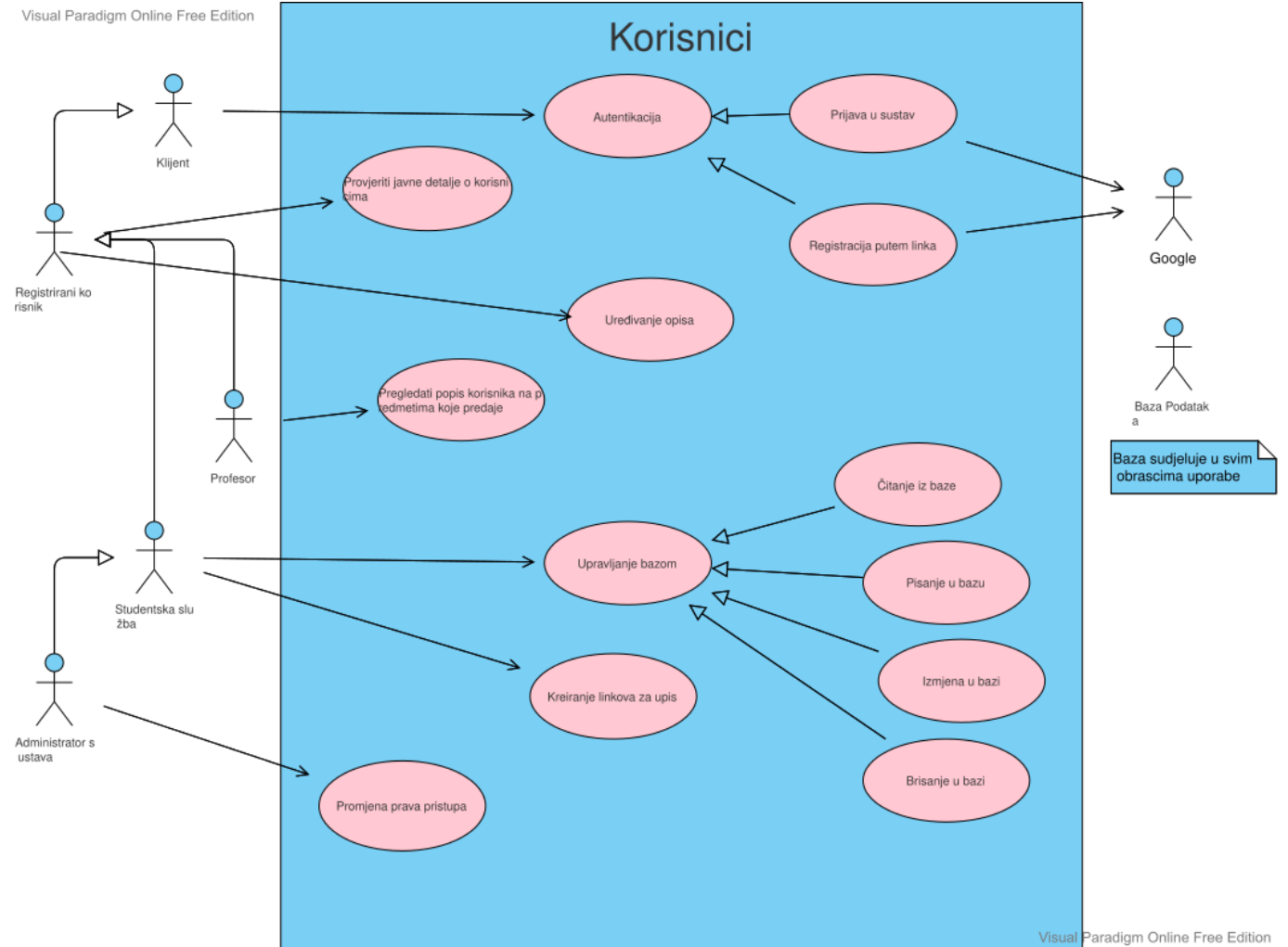
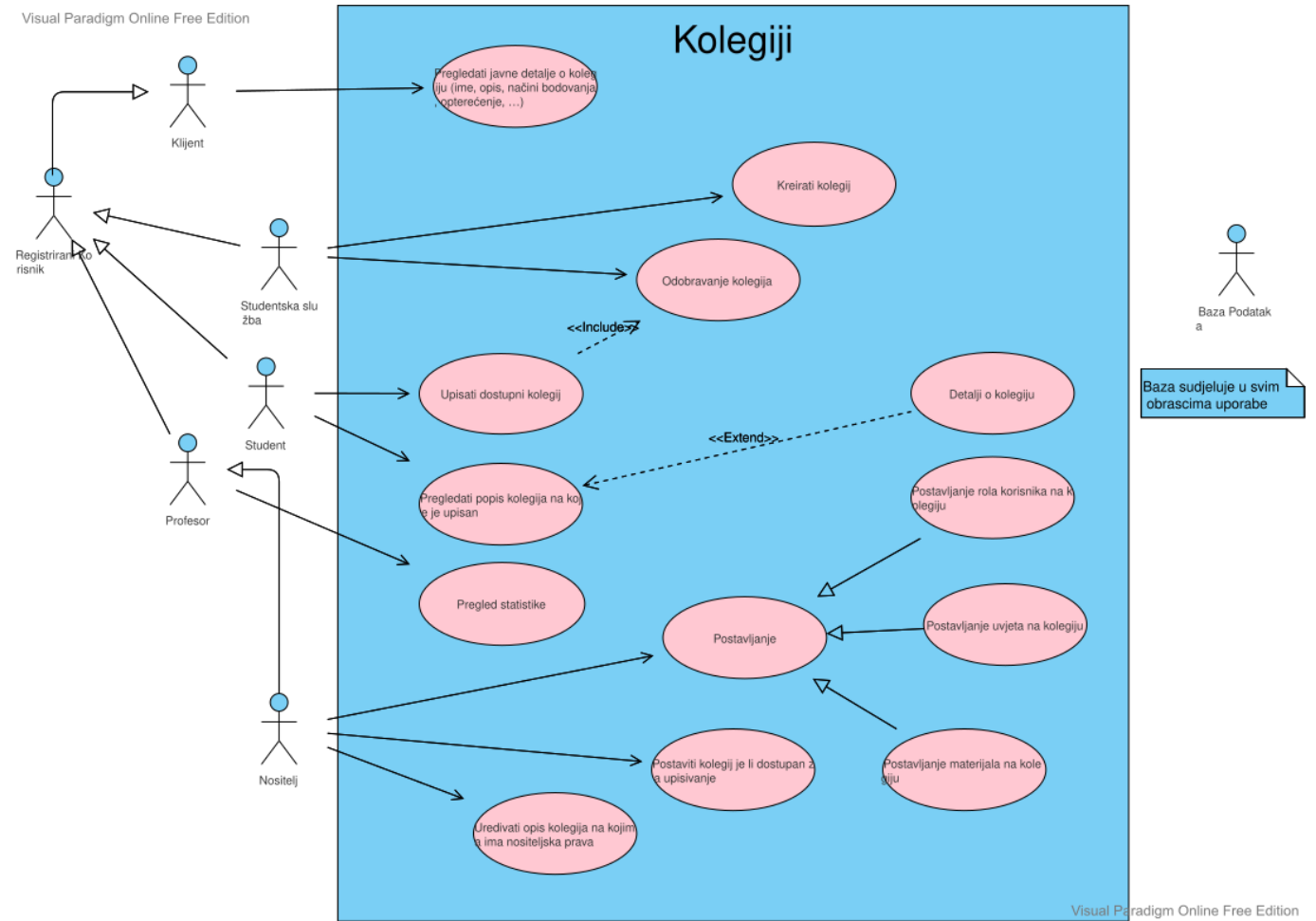
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom nositelja
- opis osnovnog tijeka:
 1. Nositelj se dogovori s doktorandom za termin s određenim datumom vremenom i mjestom
 2. Termin se zabilježi u bazu
 3. Raspravi o istraživačkim postignućima novoobjavljenih radova

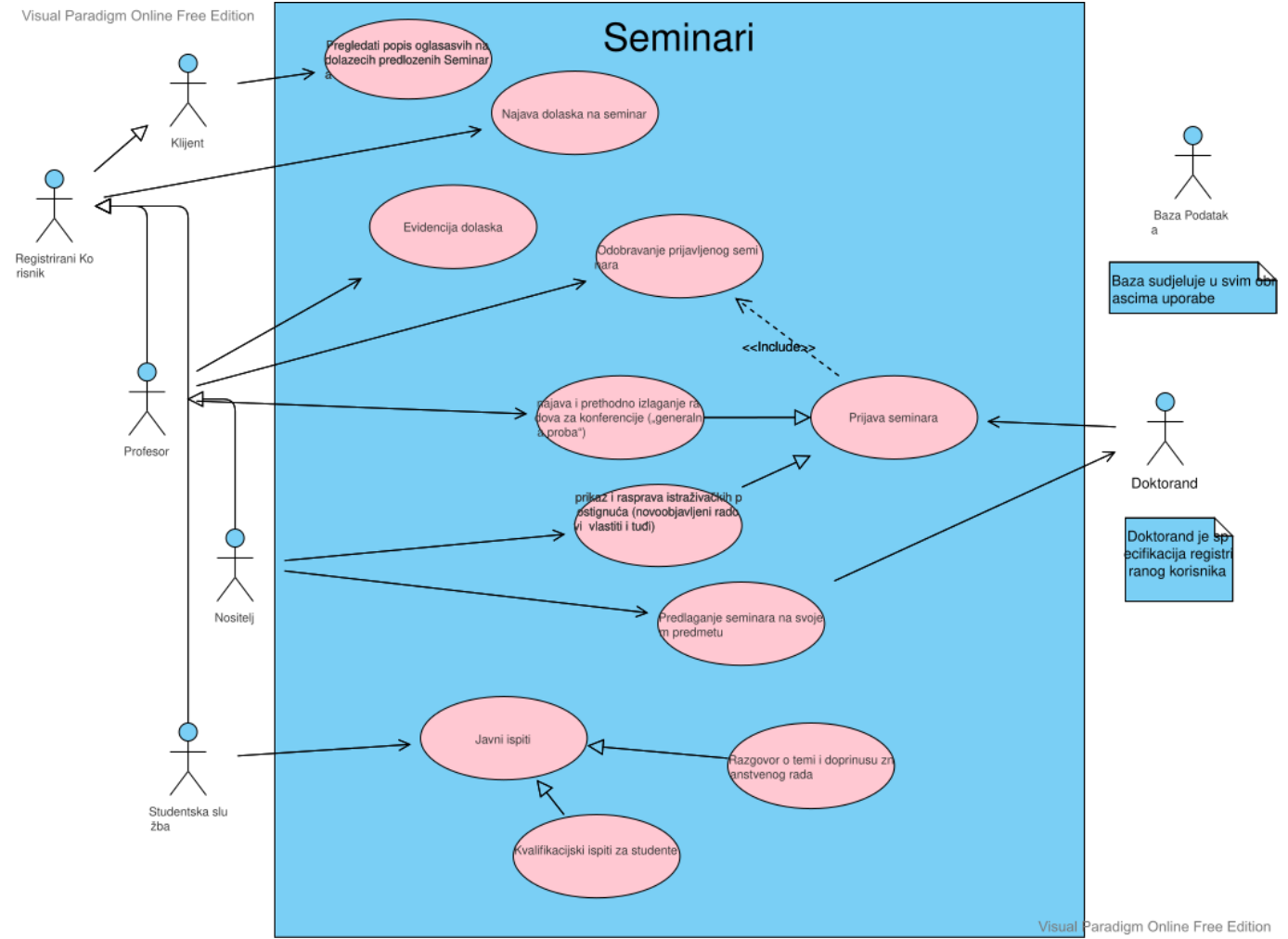
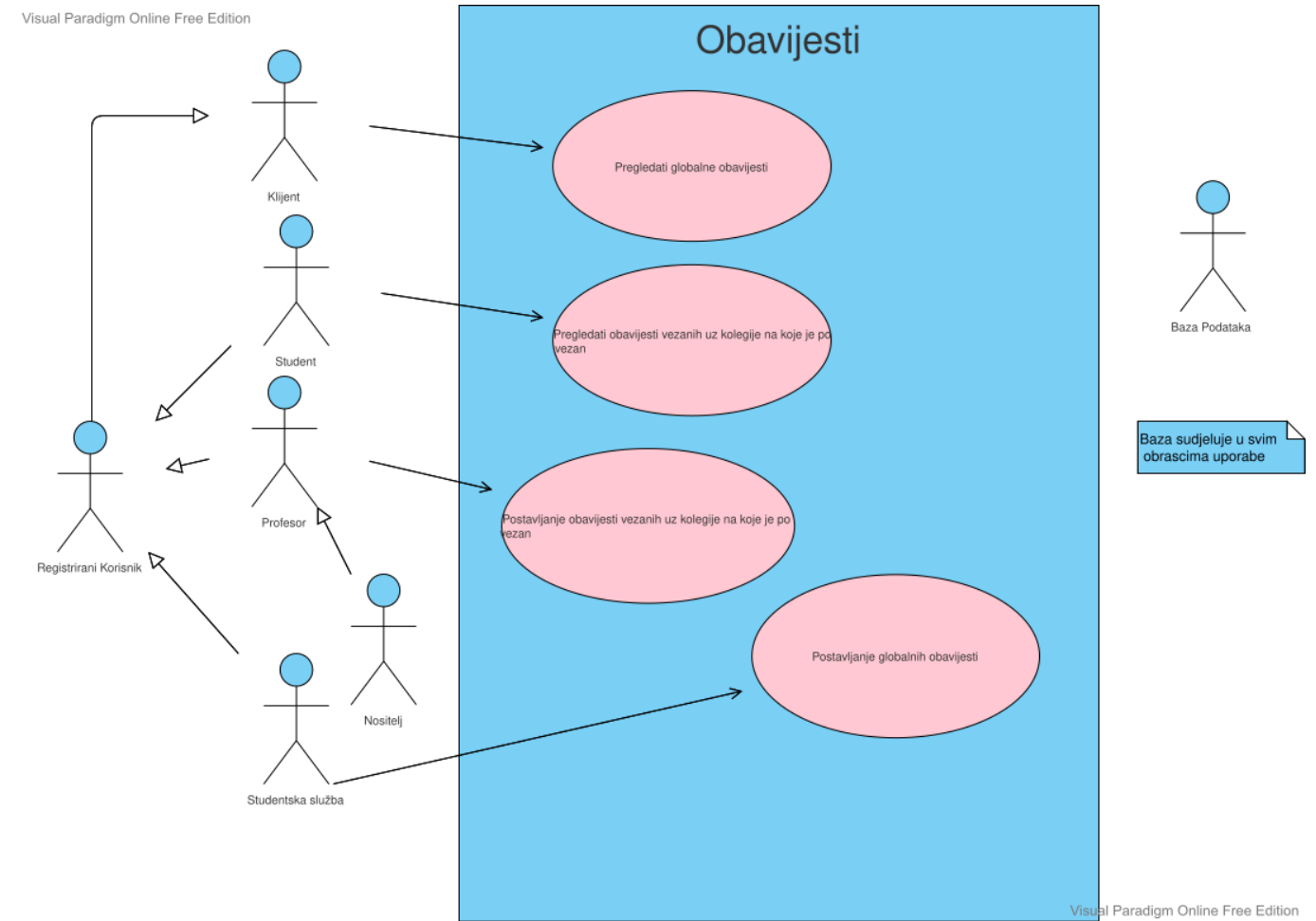
UC35 Predlaganje seminara na svojem predmetu

- Glavni sudionik: Nositelj
- cilj: Predložiti seminare na svojem predmetu
- sudionici: Baza podataka, doktorand
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom nositelja
- opis osnovnog tijeka:
 1. Nositelj pregleda listu oglasa seminara
 2. Predloži doktorandima da dođe na seminare koji su vezani za njegov predmet

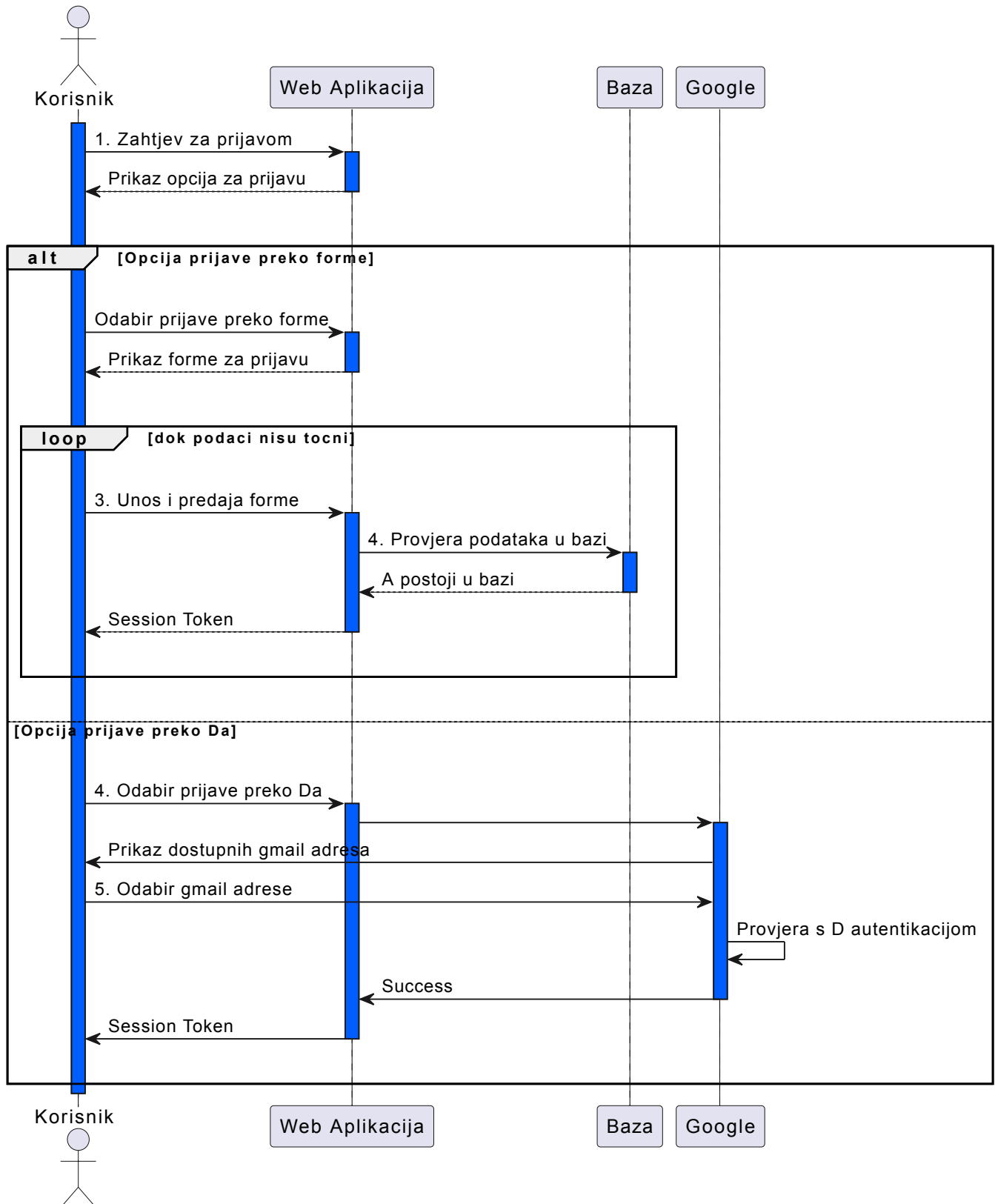
UC36 Promjena prava pristupa

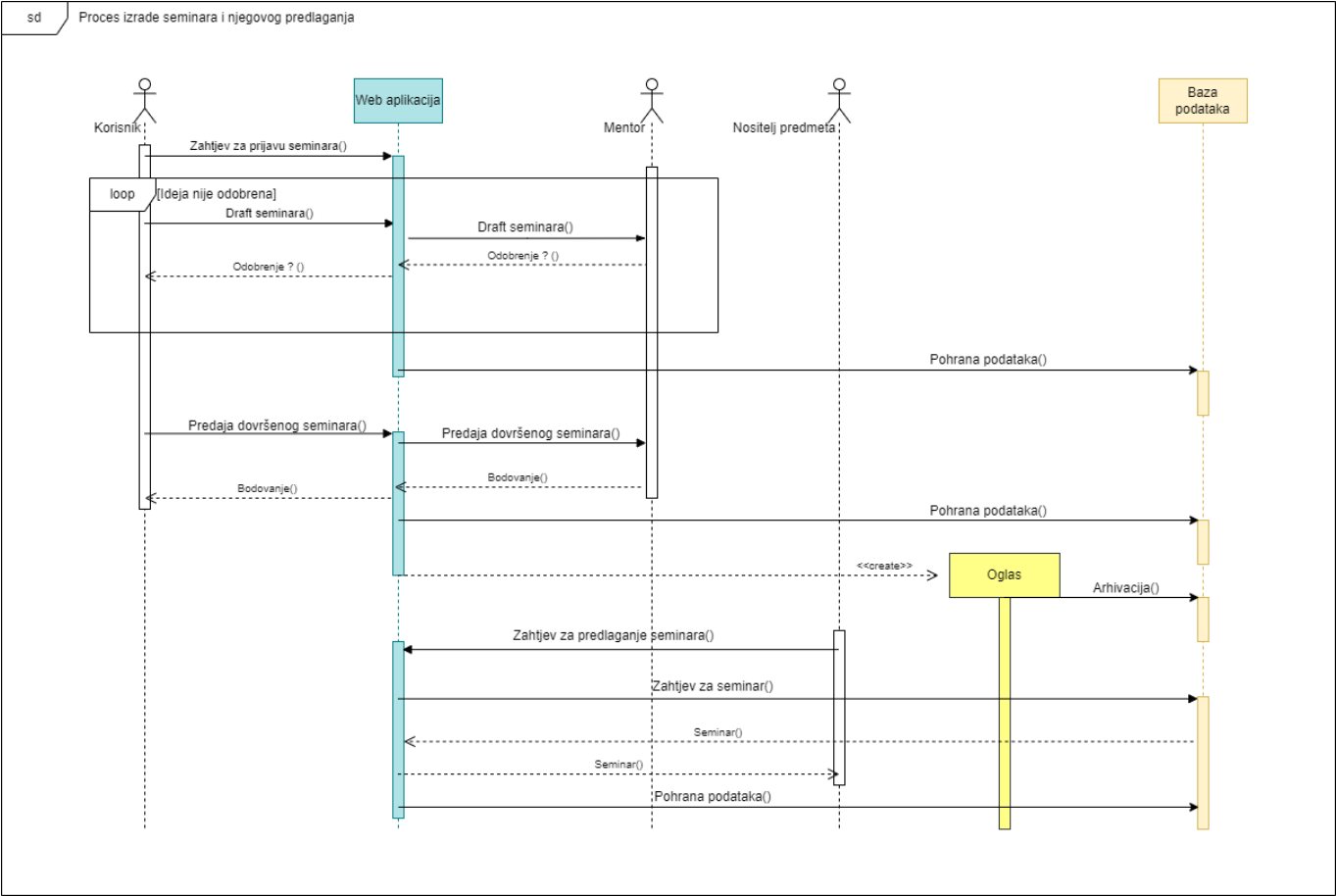
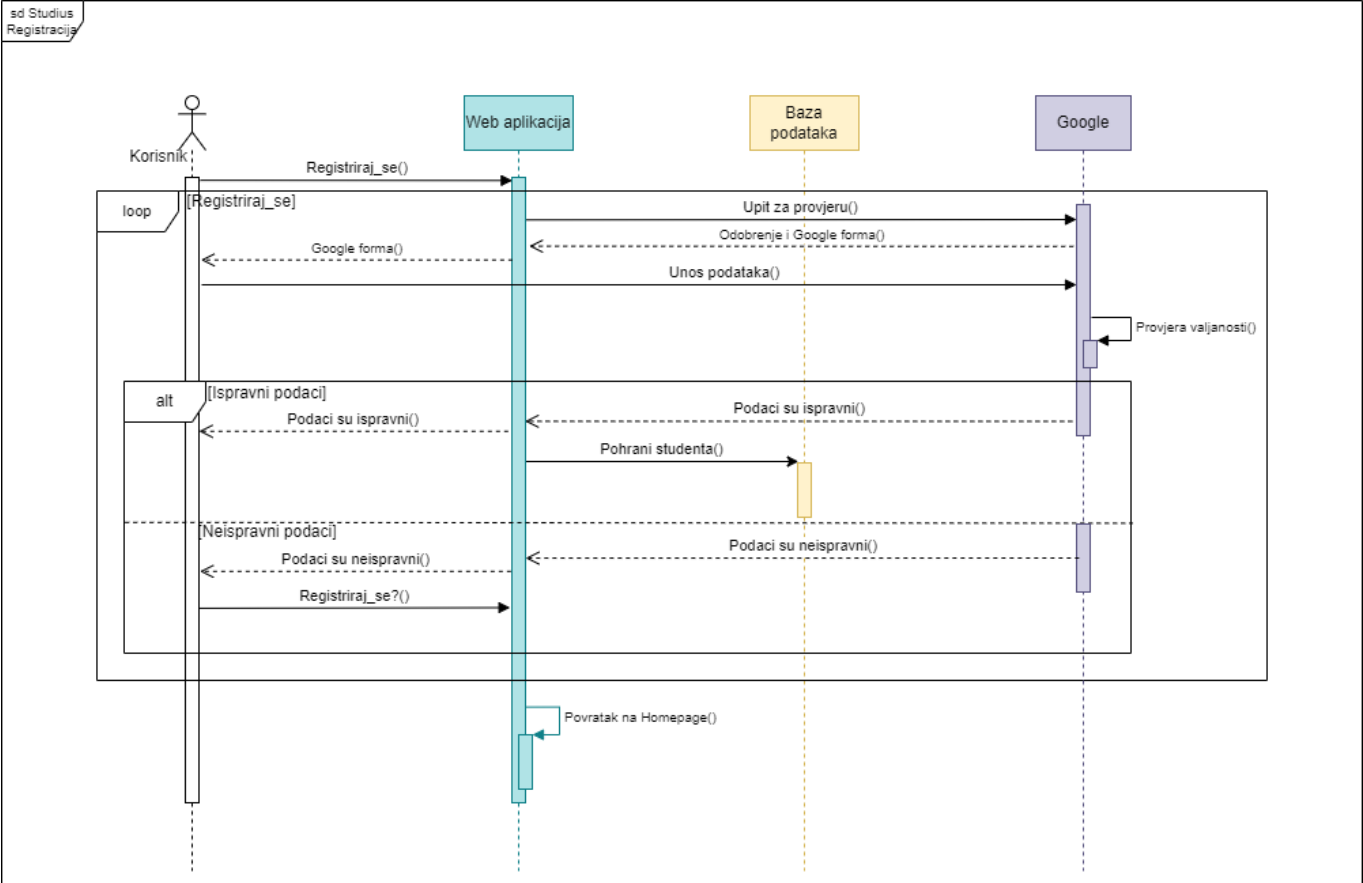
- Glavni sudionik: Administrator sustava
- cilj: Promjeniti prava pristupa
- sudionici: Baza podataka
- preduvjet: Korsnik ima svoj račun i ulogiran je u sustav sa statusom administratora sustava
- opis osnovnog tijeka:
 1. Administrator sustava može bilo kojem korisniku zabraniti pristup određenom sadržaju
 2. ako smatra da je to potrebno zbog sigurnosti Administrator sustava može mijenjati role kad zatreba (npr iz studenta u doktoranda, iz doktoranda u asistenta itd.)





Sekvencijski dijagrami





Dionici

- 1. Klijent
- 2. Registrirani korisnik

3. Student
4. Profesor
5. Nositelj
6. Doktorand
7. Studentska služba
8. Administrator sustava

Aktori i njihovi funkcionalni zahtjevi

1. Klijent
 - Prijava u sustav
 - Registriracija putem linka
 - Pregledati globalne obavijesti
 - Pregledati popis oglasa svih nadolazecih predlozenih Seminara
2. Registrirani korisnik
 - Uređivanje opisa
 - Provjeriti javne detalje o korisnicima
 - Najava dolaska na seminar
3. Student
 - Upisati dostupni kolegij
 - Pregledati popis kolegija na koje je upisan
 - Detalji o kolegiju
 - Pregledati obavijesti vezanih uz kolegije na koje je povezan
4. Pofesor
 - Pregledati popis korisnika na predmetima koje predaje
 - Postavljanje obavijesti vezanih uz kolegije na koje je povezan
 - Evidencija dolaska
 - Odobravanje prijavljenog seminara
 - Pregled statistike
 - Najava i prethodno izlaganje radova za konferencije (generalna proba)
5. Nositelj
 - Postavljanje rola korisnika na kolegiju
 - Postavljanje uvjeta na kolegiju
 - Postavljanje materijala na kolegiju
 - Postaviti kolegij je li dostupan za upisivanje
 - Urediti opis kolegija na kojima ima nositeljska prava
 - Prikaz i rasprava istraživačkih postignuća (novoobjavljeni radovi vlastiti i tuđi)
 - Predlaganje seminara na svojem predmetu
6. Doktorand
 - Prikaz i rasprava istraživačkih postignuća (novoobjavljeni radovi vlastiti i tuđi)
 - Najava i prethodno izlaganje radova za konferencije (generalna proba)
7. Studentska služba
 - Čitanje iz baze
 - Pisanje u bazu
 - Izmjena u bazi
 - Brisanje u bazi
 - Kreiranje linkova za upis

- Kreirati kolegij
- Odobravanje kolegija
- Postavljanje globalnih obavijesti
- Kvalifikacijski ispiti za studente
- Razgovor o temi i doprinosu znanstvenog rada

8. Administrator sustava

- Promjena prava pristupa

4. Arhitektura i dizajn sustava

Arhitektura se može podijeliti na tri glavne cjeline:

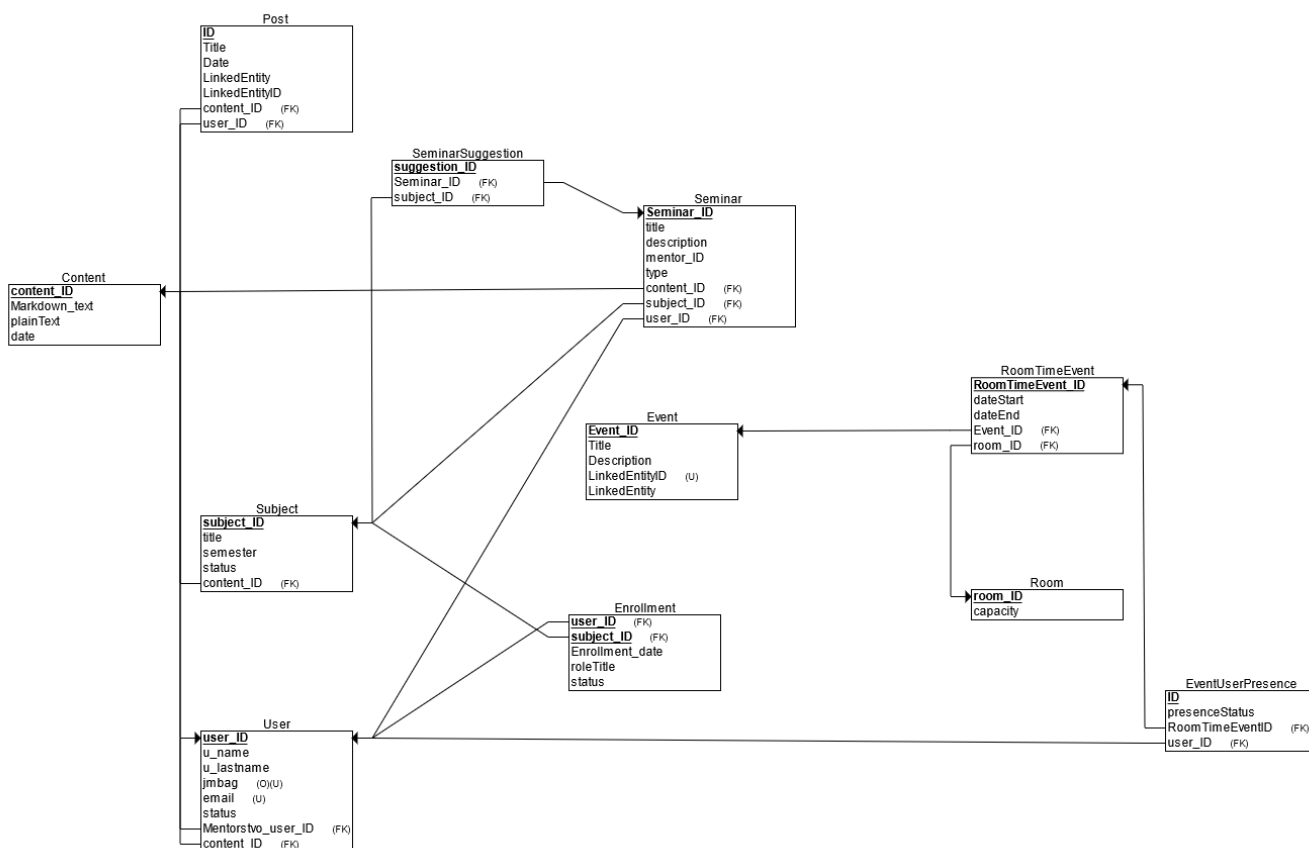
1. Klijentska aplikacija
2. Frontend servis
3. Backend podsustavi

Klijentska aplikacija je program koji korisniku pruža sučelje za komunikaciju s ostatkom sustava. Sama aplikacija je dizajnirana po vizualnim načelima dobrog UX i UI dizajna kako bi korisnik imao što manje trenja u realizaciji svojih ideja.

Frontend servis, zahvaljujući na korištenju NextJS-a kao radnog okvira, sama web aplikacija bi bila optimizirana za što veće performanse i minimalno kašnjenje kroz uporabu caching metoda i pre-renderiranim komponentama.

Backend podsustavi su najkompleksnija podcjelina same arhitekture. Backend je po uzoru na moderne mikroservisne arhitekture

4.1 Baza podataka



Post entitet sadrži sve relevantne podatke o obavijestima unutar sustava. Povezan je One-to-One vezom sa entitetom Content, te Many-to-one vezom sa entitetom User.

Post		
ID	INT	Identifikacijski broj objave
Title	VARCHAR(50)	Naslov objave
Date	DATETIME	Datum stvaranja objave
LinkedEntity	VARCHAR(50)	Za koji od entiteta (Subject, Seminar, User) je objava vezana
LinkedEntityID	INT	Identifikacijski broj (uuid) entiteta za koji je objava vezana
Content_ID	INT	Identifikacijski broj sadržaja objave (Content.content_ID)
User_ID	INT	Identifikacijski broj korisnika koji je stvorio objavu

Content entitet sadržava relevantne podatke o sadržaju, te sam sadržaj obavijesti te informacija o kolegijima I seminarima.

Content		
Content_ID	INT	Identifikacijski broj sadržaja
Markdown_text	VARCHAR	Sadržaj obavijesti ili informacije o predmetu zapisane u Markdown alatu za formatiranje teksta
plainText	VARCHAR	Zapis sadržaja u običnom text formatu
Date	DATETIME	Datum stvaranja sadržaja

Subject entitet sadržava bitne informacije o kolegijima na fakultetu. Vezom One-to-many vezan je za entitet Content

Subject		
Subject_id	INT	Jedinstveni identifikacijski broj kolegija
title	VARCHAR	Naziv kolegija
description	VARCHAR	Kratki opis kolegija
Semester	VARCHAR	Informacija o tome da li se kolegij održava u ljetnom ili zimskom semestru
Status	VARCHAR	Status kolegija, da li ga je moguće upisati ili se više ne predaje
Content_ID	VARCHAR	Content.content_ID

User entitet sadrži važne informacije o korisnicima sustava. Vezom One-to-many vezano je za entitet Content.

User		
user_ID	INT	Jedinstveni identifikacijski broj korisnika
u_firstname	VARCHAR	Ime korisnika
U_lastname	VARCHAR	Prezime korisnika
Jmbag	VARCHAR	Za sve korisnike osim studenata ima vrijednost NULL
Email	VARCHAR	e-mail korisnika
Status	VARCHAR	Određuje koju razinu pristupa ima korisnik u našem sustavu

Mentorstvo_user_ID	INT	ID mentora koji može biti dodijeljen studentu(user.user_ID)
Content_ID	INT	Content.content_ID

Enrollment entitet je identifikacijski i egzistencijalno slabi identitet koji predstavlja instancu upisa na određeni kolegij. Povezan je sa Many-to-one vezama za User i Subject entitete, čiji primarni ključevi predstavljaju njegov kompozitni ključ.

Enrollment		
User_ID	INT	User.user_ID
Subject_ID	INT	Subject.subject_ID
Enrollment_date	DATETIME	Datum ostvarenja upisa
Status	VARCHAR	Označava status pojedinog upisa (aktivan, arhiviran)

Seminar entitet sadrži važne informacije o seminarima koji doktorandi predaju na sustav. Povezan je One-to-one vezom sa entitetom Content, Many-to-one vezom sa entitetom Subject, te One-to-one sa Userom

Seminar		
Seminar_ID	INT	Jedinstveni identifikacijski broj seminara
Title	VARCHAR	Naslov seminara
Description	VARCHAR	Kratki opis seminara
Mentor_ID	INT	Jedinstveni identifikacijski broj mentora

Type	VARCHAR	Tip seminara
Content_ID	INT	Content.content_ID
Subject_ID	INT	Subject.subject_ID
User_ID	INT	User.user_ID

SeminarSuggestion entitet koji sadržava bitne informacije o seminaru koji nositelji odabiru kao prijedlog za studente svojeg predmeta. Vezom One-to-one vezan je za entitete Seminar I Subject

SeminarSuggestion		
Suggestion_ID	INT	Jedinstveni identifikacijski broj prijedloga
Seminar_ID	INT	Seminar.Seminar_ID
Subject_ID	INT	Subject.subject_ID

Room entitet sadrži važne informacije o dvoranama I prostorijama na fakultetu. One-to-Many vezom je spojen sa entitetom RoomTimeEvent.

Room		
Room_ID	INT	Jedinstveni identifikacijski broj prostorije
Capacity	INT	Kapacitet prostorije

RoomTimeEvent entitet sadrži važne informacije o održavanju događaja na fakultetu. Spaja događaj

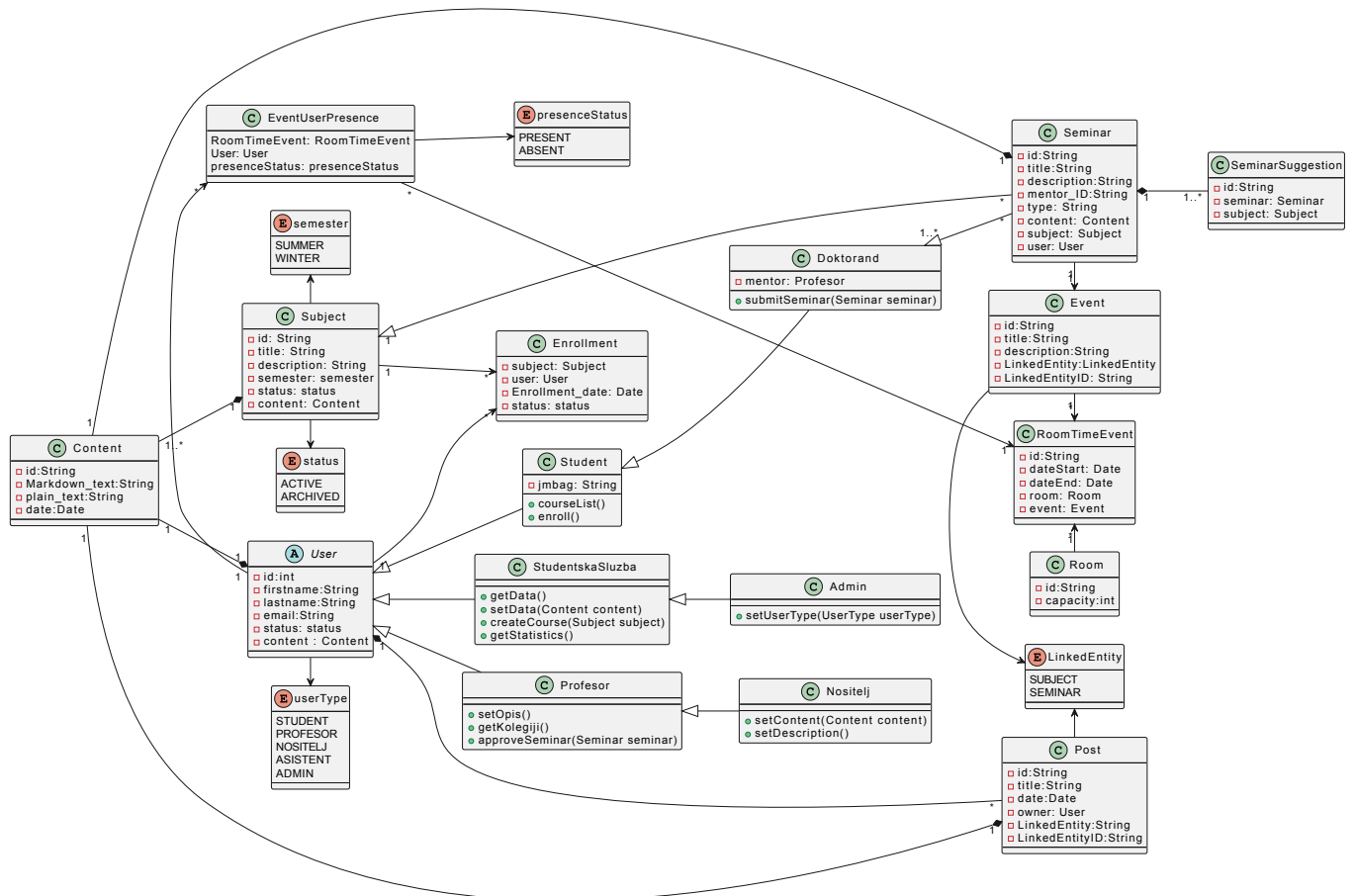
sa prostorom I vremenom održavanja. Many-to-one vezama je spojen sa entitetima Room I Event.

RoomTimeEvent		
RoomTimeEventID	INT	Jedinstveni identifikacijski broj održavanja događaja
dateStart	DATETIME	Vrijeme početka događaja
dateEnd	DATETIME	Vrijeme završetka događaja
Event_ID	INT	Event.event_ID
Room_ID	INT	Room.room_ID

EventUserPresence entitet sadrži bitne informacije o prisutnosti studenata na događajima gdje se njihova prisutnost očekuje. Many-to-one vezom je spojena sa User entitetom te Many-to-One vezom je spojena na RoomTimeEvent entitet.

EventUserPresence		
presence_ID	INT	Identifikacijski broj evidencije
presenceStatus	BOOL	True/False, je li student bio prisutan
RoomTimeEvent_ID	INT	RoomTimeEvent.RoomTimeEvent_ID
User_ID	INT	User.user_ID

4.2 Dijagram razreda



4.3 Dijagram stanja

4.4 Dijagram aktivnosti

4.5 Dijagram komponenti

5. Implementacija i korisničko sučelje

5.1 Korištene tehnologije i alati

Development frontenda bi bio baziran na dizajnu izrađenom u **Figma**. Definiran bi bio kompletan Design System, i svaka nova komponenta bi bila dokumentirana.

Frontend bi bio implementiran koristeći React, posebno **NextJS** uz **Typescript**. Za stilove bi bio korišten **Tailwind CSS**.

Backend bi bio pisan u **Nodeu** kroz Typescript, te bismo koristili **Supabase** kao rješenje za pohranu podataka.

Većina funkcionalnosti na frontu bi bila zamotana kroz **Serverless** APIje, koje nam omogućuje hostanje preko **Vercela** (kreatori NextJS-a)

5.2 Ispitivanje programskog rješenja

5.3 Dijagram razmještaja

5.4 Upute za puštanje u pogon

Izvođenje razvoja

U dogovoru s cijelom ekipom, i pripadnim mentorima, koja bi radila na projektu, radili bismo prateći tjedne sprintove uz prilagođen oblik SCRUM-a.

Način organizacije koji smo odlučili koristiti kao razvojni tim je SCRUM.

koji ćemo kao razvojni tima pratiti

Sprintovi

Sprintovi su tjedna ili dvotjedna razdoblja na čijem se početku određuje niz zadataka i tema na koje se fokusira većina razvojnog procesa.

Tjedni / dvotjedni sastanci

Obično traju oko sat vremena. Cilj je imati viši pregled nad onime što je dovršeno u prethodnom sprintu te koji će zadaci ući u sljedeći sprint.

Kraj svakog sprinta obilježen je ovakvim sastankom, gdje gledamo koliko su uspješno bili postavljeni zadaci, koji su sve ciljevi postignuti te se reflektiramo na sam proces (što bi moglo biti bolje)

Bilješke ovakvih sastanaka vodimo kroz alat **Notion** te na temelju tih zapisnika na mjesečnoj bazi stvaramo dokument koji prosljeđujemo svim mentorima vezanim uz projekt.

Dnevni sastanci

Dnevni bi sastanci trebali trajati manje od 10 minuta svaki dan i nisu obavezni. Cilj je uskladiti zadatke koje svaki član rješava.

Mjesečni sastanci

Cilj ovih sastanaka je usklađivanje s mentorima iz raznih zavoda, te iznošenje i skupno razmišljanje o napretku te idućim koracima.

Proces izvedbe razvoja

Dijelovi projekta su već započeti, naime Korisničko Putovanje (User Journey), model baze podataka za osnovni set funkcionalnosti, te istraživanje tehnologija koje bismo primjenjivali.

Frontend tijekom