Indeks: **IB150051**

Student: Edin Pinjić

**Studenski servis**

PRIJAVA TEME SEMINARSKOG RADA

1. **Uvod**

Decentralizovani podaci je jako čest i u većini slučajeva komplikovan problem u smislu pronalaženja željenih podataka u trenutku kada nam su isti podaci neophodni. Naročito se taj problem odnosi na studente kojima je svaka ušteda vremena na dobavljanje potrebnih informacija i pravovremeno dobijanje traženih informacija jako bitna.

Kao jedno od mogućih rješenja, koje i nije idealno, podrazumijeva kontaktiranje studentske službe i samih profesora, gdje može doći do preopterećenja samih uposlenika studentske službe, ali i profesora, naročito u periodu ispita i ovjere semestara. U cilju pronalaženja adekvatnog rješenja za pomenute slučajeve, u ovom seminarskog radu će biti opisan razvoj mobilne aplikacija pod nazivom Studentski servis koja će omogućiti adekvatno dobavljanje potrebnih informacija.

1. **Ciljevi projekta**

* Centralizovani podaci
* Dostupnost potrebnih podataka u svakom trenutku
* Lakše upravljanje podacima

1. **Funkcionalnosti aplikacije Studentski servis**

Aplikacija Studenski servis će svojim korisnicima osigurati sljedeće funkcionalnosti:

* Upravljanje uplatama
* Upravljanje podacima uposlenika akademije i studenata
* Upravljanje važnim događajima
* Upravljanje obavijestima i ispitima
  1. **Upravljanje uplatama**

U okviru ove funkcionalnosti, korisnici administracija će moći upravljati svim uplatama studenata, tj. vršenje unosa, izmjene i brisanja. Korisnici studenti će imati samo mogućnost na uvid svojih vlastitih izvršenih uplata, tj. prikazivanje datuma, iznosa uplate, svrhe, te ko od uposlenika je evidentirao izvršenu uplatu, tako da u slučaju neslaganja, student odmah zna kome treba da se obrati. Sve navedene informacije će se moći dobiti klikom na dugme „Detalji“, kao što je to u nastavku prikazano.

****

Slika 1. Prikaz uplata

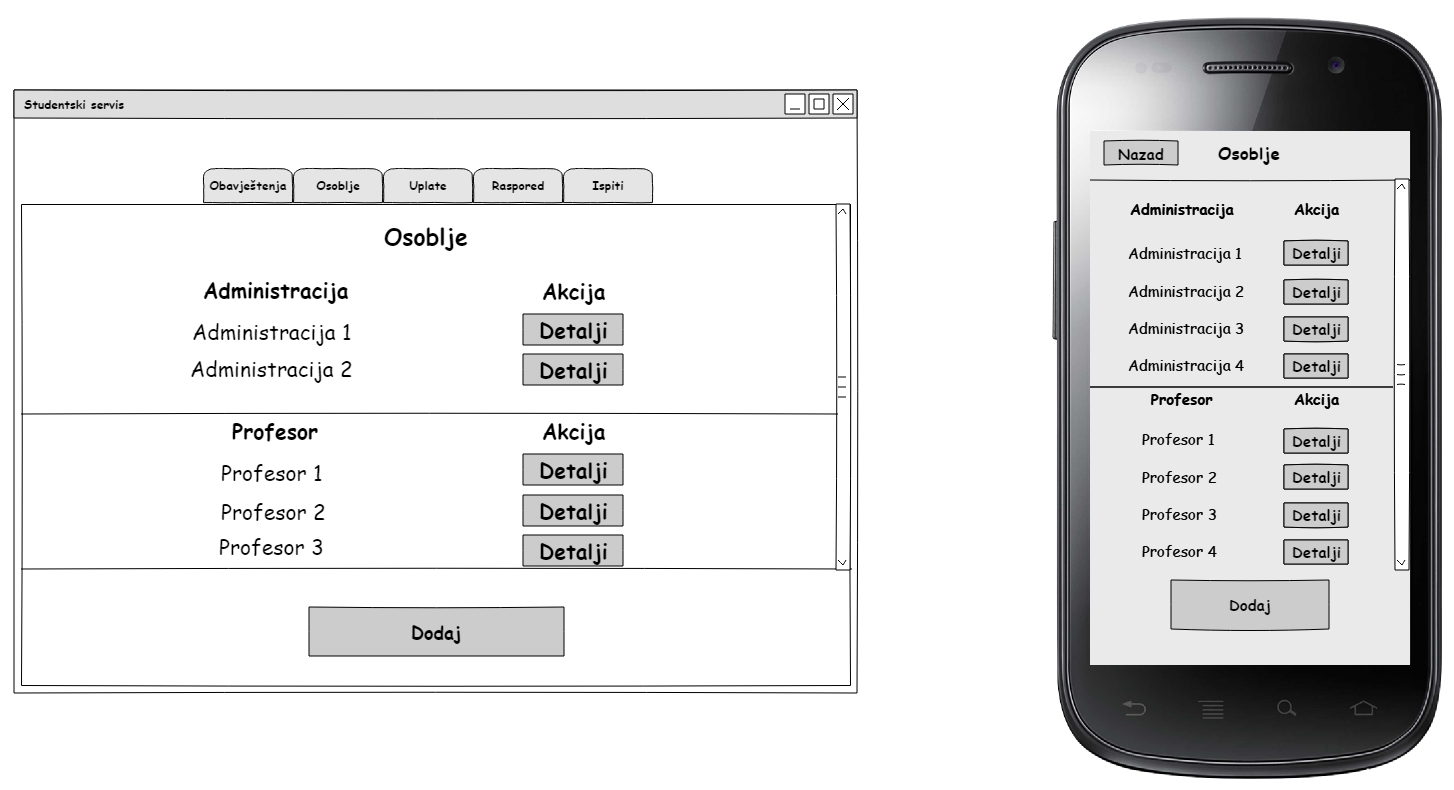


Slika 2. Unos uplata

* 1. **Upravljanje podacima uposlenika akademije i studenata**

Ova funkcionalnost se odnosi na uposlenike akademije, tačnije korisnike Administracija, koji će kroz jednostavne korake izvršiti unos novih članova akademije, tj. unos podataka poput imena, prezimena, pozicije u akademiji, godina studija/godine staža i slično, te osim toga imaće mogućnost uređivanja i brisanja podataka o istom. Korisnici Studenti će također imati ovu funkcionalnost, ali moći će samo imati na uvid liste osoblja, a sve detaljnije informacije poput kontakt telefona, mail-a i sl. će dobiti klikom na dugme detalji.



Slika 3. Unos podataka o novom osoblju

Slika 4. Prikaz osoblja

* 1. **Upravljanje važnim događajima**

Ova funkcionalnost se odnosi na Student korisnike aplikacije. Putem ove funkcionalnosti, studenti će imati uvid u već snimljene događaje važnih za same njih, ali i dodavanje događaja, gdje će biti naznačen naziv, datum, opis događaja, te oznaka da li je događaj važan za njih. Prvo će biti prikazana lista svih događaja, te klikom na dugme detalji, dobijaju opis snimljenog događaja, tj. snimljeni sadržaj za određeni događaj.



Slika 5. Prikaz događaja



Slika 6. Unos događaja

* 1. **Upravljanje obavijestima i ispitima**

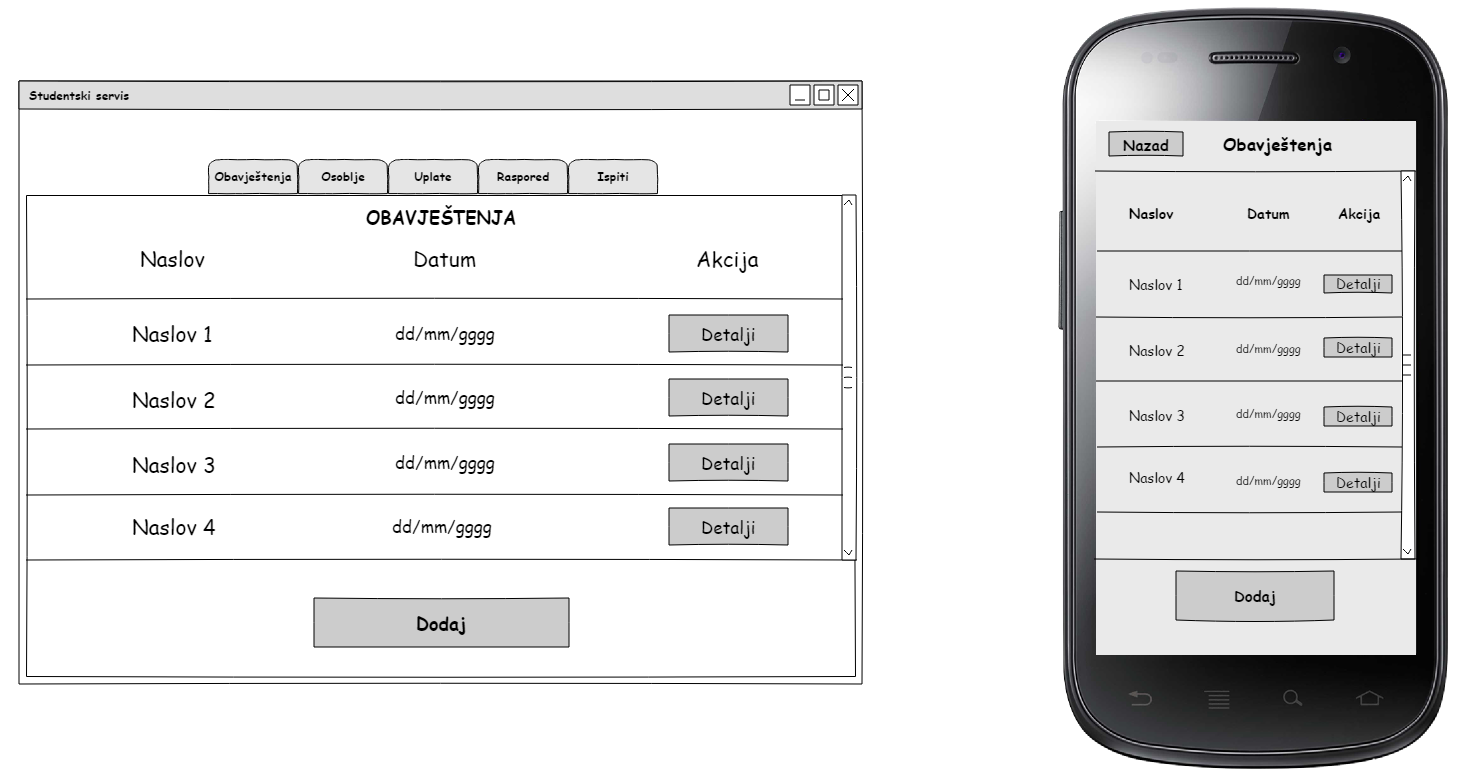
Pomenuta funkcionalnost se odnosi na sve korisnike, međutim samo korisnik profesor ima mogućnost da izvrši dodavanje nove obavijesti i termine novih ispita, dok korisnik student ima samo mogućnost pregleda istog.



Slika 7. Prikaz zakazanih ispita



Slika 8. Unos ispita



Slika 9. Prikaz obavještenja



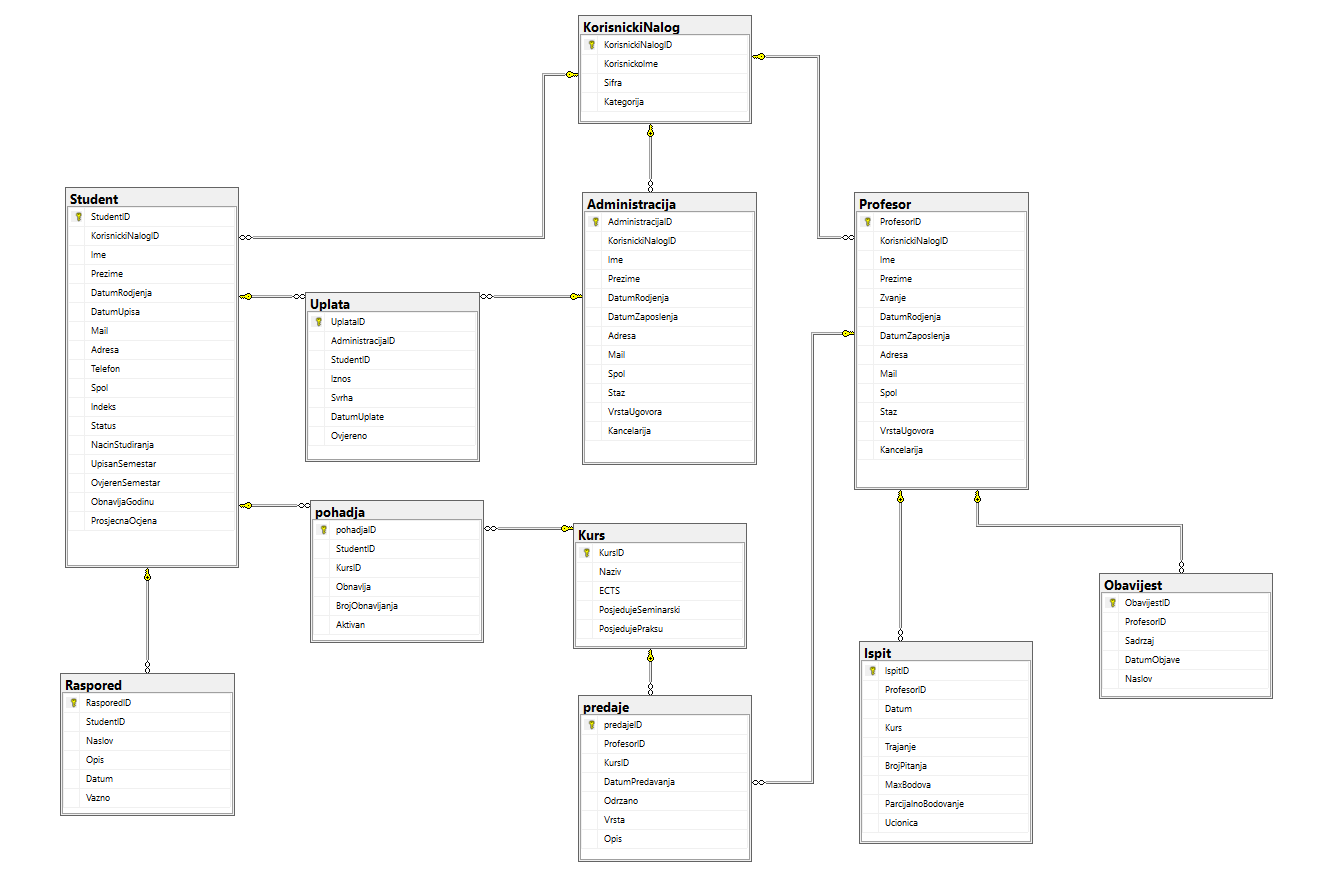
Slika 10. Unos obavještenja

1. **Opis web servisa**

Stil softverske arhitekture koji će se koristiti jeste REST, koristit će se Web API i korstit će se windows forme za prikaz i upravljanje funkcijama. Platforma koja će se koristiti za ostvarenje pomenutog rješenja jeste ASP.NET Core. Posjedovat će kontrolere za uplate, obavještenja, ispite, osoblje i raspored, te će imati kontroler za autentifikaciju/autoriziaciju. U principu, metode koje će se ovdje koristi su Get (Dobavljanje podataka), Post (Dodavanje podataka), Put (Izmjena postojećih podataka) i Delete (brisanje podataka) metode, koje će se koristiti uz pomoć flurl http NuGet paketa. Naravno, biće primijenjeni interface-i, te samim tim i dependency injection-i.

1. **Opis baze podataka**

Softver koji će se za ovo rješenje koristiti jeste MS SQL Server 2017, te će se relaciona baza podataka koristiti. Baza podataka je već gotova i sastoji se od jedanaest (11) tabela i obuhvata sve potrebne podatke koji su potrebni za pomenutu aplikaciju. Relaciona šema pomenute baze podataka je prikazana u nastavku.



Slika . Šema relacione baze podataka za aplikaciju Studentski servis

**Zaključak**

U ovoj projektnoj dokumentaciji predstavljeno je softversko rješenje, koje će koristiti web API i windows forme za upravljanje svih pomenutih elemenata. Svrha rješenja jeste poboljšanje suradnje između studenata i uposlenika fakulteta, te unaprijeđenje poslovanja i olakšanje poslovnih procesa i čiji su ciljevi postizanje centralizacije i dostupnosti podataka, te jednostavnije upravljanje samim podacima. Rezultat ovog projekta čini funkcionalni sistem, koji će biti javno dostupna svim članovima fakulteta. Zahvaljujući ovom rješenju, uz funkcionalnosti obavještenja, ispiti, raspored i uplate, polaznici će moći u par klikova saznati sve što im je potrebno za efikasan i napredan rad. Eliminisat će telefonske pozive prema studentskoj službi, koja i nije baš često dostupna, vrijeme izgubljeno na čekanju nekih jednostavnih i kratkih informacija. Pored toga, administrativnom osoblju će uz funkcionalnosti upravljanje uplata i fakultetskom osoblju biti olakšano regulisanje papirologije, te neće tako lahko doći do grešaka, koje su na klasičnom načinu jako česte. Projekat, dakle, ne samo da je koristan za sve polaznike, već i za osoblje fakulteta, koji će imati kvalitetan i siguran pristup svom poslovanju, s potpunom kontrolom i evidencijom njihovog poslovanja, u skladu s najvišim standardima. Mana ove aplikacije je što su obuhvaćene sve potrebne funkcionalnosti, ali to se nastoji vremenom promijeniti. Treba napomenuti da će trenutna aplikacija obuhvatati neke od najosnovnijih funkcionalnosti, navedene u ranijem dijelu dokumenta, te za buduće verzije bi se mogle implementirati funkcionalnosti poput na primjer ovjere semestara, polaganje online testova, postavljanje i preuzimanje nastavnog materijala i slično.