

*Datum: 05.03.2023.*

## Upute za izradu seminarskog rada

U nastavku su date upute za izradu seminarskog rada iz predmeta Razvoj softvera II.

- Seminarski rad se radi samostalno ili u grupi ukoliko za to postoje ispunjeni preduslovi vezani za obim rada i specifičnost teme.
- Tema seminarskog rada može biti iz domene elektronskog poslovanja, ali ne smije uključiti klasičnu prodaju proizvoda koja se implementira u sklopu nastave na ovom predmetu. Tačnije, odabrana tema ne može biti u potpunosti zasnovana na funkcionalnostima zajedničkog projekta *eProdaja*.

### Prijava teme seminarskog rada

- Temu rada predlaže svaki student bez obzira da li se odlučio seminarski rad raditi samostalno ili kao član grupe. Ukoliko se radi o grupnom radu, svaki student predaje zasebnu prijavu unutar koje su opisane sve funkcionalnosti rada sa jasnom naznakom koji član tima će biti zadužen za svaku od njih. Svaki student, ako se radi o grupnom radu, u okviru svoje prijave, na početku navodi sebe kao prvog člana tima, a nakon toga i ostale članove.
- Članovi grupe trebaju podjednako raditi na svim dijelovima projekta (API, desktop i mobile).
- Tema seminarskog rada se predlaže u formi prijave (.doc fajl) unutar koje su pobrojane i detaljnije opisane sve funkcionalnosti. Pored opisa funkcionalnosti, prijava seminarskog rada treba sadržavati i skice (engl. mockups) pojedinih dijelova interfejsa koje će biti implementirane. Sve skice moraju biti konzistentne, prateći prihvatljive UI/UX principe, te ne mogu biti skice gotovih, već implementiranih, interfejsa.
- Prijava seminarskog rada se vrši putem DL sistema, tačnije putem sekcije Zadaci. Korištenje sekcije Zadaci je pojašnjeno u sekciji „Predaja seminarskog rada“ ovog dokumenta. U okviru pomenute sekcije studenti će, tokom perioda u kojima se ne organizuju ispiti, najkasnije do 15-og u mjesecu, dobijati komentare i eventualne zahtjeve za korekcijom prijave.
- Tek nakon odobrenja prijave može se krenuti sa implementacijom projekta.

- Svaki projekat treba posjedovati dva korisnička interface-a: desktop (Flutter) i mobilni (Flutter), koja će koristiti zajedničku mikroservisnu infrastrukturu za komunikaciju sa bazom podataka. Pomenuta mikroservisna infrastruktura podrazumijeva implementaciju dva ASP.NET Core Web API projekta pri čemu jedan predstavlja glavni API, a drugi, pomoćni, služi za demonstraciju rada sa mikroservisima koji komuniciraju koristeći neki od asinhronih mehanizama npr. RabbitMQ. Pomoćni API treba osigurati obradu podataka neophodnih za funkcionisanje aplikacije.
- Prijave koje su predate nakon definisanih rokova neće biti pregledane.

## Implementacija seminarskog rada

- Za izradu aplikacije koristiti Visual Studio 2022 uz upotrebu C# programskog jezika.
- Baza podataka treba da sadrži minimalno 10 tabela (ne uključujući referentne tabele), a za njenu izradu je potrebno koristiti SQL Server 2017 (ili noviju verziju) ili neki drugu relacionu bazu podataka. Bazu podataka imenovati brojem indeksa (bez prefiksa IB npr. 180081).
- Tokom implementacije aplikacije može se koristiti jedan od sljedećih pristupa: Code First, Database First, Code First from Database i sl., sa ili bez upotrebe Stored procedura.
- Desktop dio predstavlja administrativni dio aplikacije, sa obaveznim dijelom za izvještavanje. Pregled svih podataka mora da uključi minimalno jedan parametar za pretragu, osim ako to nije opravdano i primjenjivo drugačije, npr. prikaz 5 najboljih klijenata.
- Mobilni dio aplikacije treba da omogući klijentima pregled uslužnih ili drugih djelatnosti kompanije, eventualno kreiranje narudžbe, pregled historije aktivnosti (prethodne narudžbe, korištene usluge i sl.) za registrovane klijente, te pregled i izmjenu profila. Pomenuto svakako zavisi od teme i područja za koju se razvija aplikacija.
- U skladu sa funkcionalnostima, potrebno je implementirati jednostavniji modul sistema preporuke upotrebom nekog od poznatih algoritama. Umjesto algoritma preporuke, moguće je raditi na implementaciji Identity Server-a u cilju osiguranja naprednijih oblika autentifikacije korisnika.

## Predaja seminarskog rada

- Neposredno pred svaki ispitni rok će se definisati termin za predaju seminarskih radova. U slučaju da se organizuju dva termina, period za predaju će biti znatno kraći.
- Na jednom ispitnom terminu nije moguće izvršiti prijavu teme i predaju seminarskog rada.

- Seminarski rad potrebno je postaviti na vlastiti github repozitorij, te tom prilikom voditi računa da se ispravno podesi .gitignore fajl.
- Github repozitorij na kome je postavljen projekat treba biti javan (eng. public).
- Seminarski radovi će se pregledati isključivo nakon definisanog roka za predaju. Prilikom predaje seminarskog rada potrebno je pratiti sljedeće korake:
  1. U sklopu github repozitorija, unutar *readme.md*, potrebno je navesti sve kredencijale za prijavu na desktop/mobilnu aplikaciju.
  2. U sklopu projekta, **baza zajedno sa podacima kao i API mikroservisi moraju biti dokerizirani.**
  3. Link na github repozitorij postaviti na DLWMS na sljedeći način:
    - a. Sa glavnog menija odabrati *Nastava*
    - b. Odabrati godinu studija 3 i predmet *Razvoj softvera II*
    - c. Opcija: *Zadaci*
      - i. Na trenutno aktivnom zadatku odabrati link *Prijavi temu*<sup>1</sup>
      - ii. Nakon prijave teme iz liste Moji radovi odabrati link *Detalji*
      - iii. Postaviti link na projekat
  4. Korisnički podaci za pristup aplikaciji seminarskog rada trebaju biti (obavezno ih navesti unutar *readme.md* fajla):
    - a. Desktop verzija:
      - i. Korisničko ime: **desktop**
      - ii. Lozinka: **test**
    - b. Mobile verzija:
      - i. Korisničko ime: **mobile**
      - ii. Lozinka: **test**
    - c. Ukoliko aplikacija posjeduje više korisničkih uloga tada pristupni podaci trebaju biti u sljedećoj formi:
      - i. Korisničko ime: **nazivUloge**
      - ii. Lozinka: **test**
  5. Konekcijski string (eng. connection string) treba biti unutar konfiguracijskog fajla aplikacije.
  6. Baza podataka treba posjedovati sve podatke neophodne za testiranje aplikacije. Također, podaci se mogu kreirati i prilikom pokretanja aplikacije (engl. migrations). Radovi koji sadrže nedovoljan broj zapisa u bazi neće biti detaljnije evaluirani.
  7. Programski kod koji se ne koristi u projektu ne smije biti sastavni dio projekta.

<sup>1</sup> Ukoliko ste temu ranije prijavili i zadatak je još uvijek aktivan, preskačete ovaj korak

8. Kontrole (engl. widgets) za koje ne postoji implementirana funkcionalnost ne smiju postojati na formi/page-u.
9. Kontrole trebaju učitavati samo one podatke za koje su namijenjene.
10. Sve forme moraju biti validirane s odgovarajućim upozorenjima.

➤ **Osnovni preduslov za evaluaciju jeste da se rad/aplikacija može pokrenuti<sup>2</sup> bez dodatnih intervencija koje podrazumijevaju bilo kakvu modifikacija programskog koda, dodavanja referenci ili biblioteka, modifikaciju linkova, portova, konekcijskih stringova i sl. Također, rad/aplikacija mora biti stabilna prilikom korištenja.**

- Radovi koji su modifikovani nakon definisanog termina za predaju, bez obzira na šta se odnosila ta modifikacija, neće biti evaluirani na tom roku.
- Prije predaje, obavezno testirati aplikaciju i na drugim okruženjima mimo razvojnog, te otkloniti sve uočene nedostatke. Prethodno se posebno odnosi na pokretanje projekta bez potrebe za bilo kakvim intervencijama.
- Radovi koji na bilo koji način budu odstupali od pravila definisanih u ovom dokumentu ili budu prijavljivali greške prilikom korištenja osnovnih funkcionalnosti neće biti prihvaćeni. Pomenuto se posebno odnosi na radove koji, nakon korekcija, budu posjedovali nedostatke na koje im je već ukazano u nekoj od ranijih verzija aplikacije.
- Nakon pregleda, lista prihvaćenih seminarskih radova će biti objavljena putem obavijesti na DL sistemu.
- Studenti kojima seminarski rad ne bude prihvaćen će dobiti kraći komentar o razlozima neprihvatanja, a oni su dužni korigovati iste ili slične nedostatke u cijelom radu.

➤ **Rok za završetak svih obaveza na predmetu je posljednji apsolventski rok tj. april naredne akademske godine. Studenti koji ne završe sve obaveze na predmetu dužni su svoje obaveze realizovati prema uputama za novu akademsku godinu.**

## Odbrana seminarskog rada

1. Nakon prihvatanja aplikacije (seminarskog rada) studentima će se omogućiti pristup njegovoj odbrani.

<sup>2</sup> Pod pokretanjem se podrazumijeva mogućnost korištenja i testiranja osnovnih funkcionalnosti aplikacije.

2. U toku odbrane rada studenti dobivaju zadatke u okviru kojih se zahtijeva implementacija određenih funkcionalnosti (API, Flutter-desktop i mobile) u okviru vlastitog ili templejt projekta.
3. U slučaju da je seminarski rad rađen u grupi, članovi grupe moraju biti detaljno upoznati sa svim dijelovima aplikacije, bez obzira na funkcionalnosti koje su pojedinačno implementirali.

U okviru nastave će biti kreirana aplikacija *eProdaja* sa svim prethodno navedenim komponentama. Studenti će imati pristup video materijalima koji će biti dostupni putem FIT servisa.

U slučaju potrebe, dodatna pojašnjenja ili dopune procedura koje se odnose na izradu seminarskih radova će biti objavljene u okviru zasebnih dokumenata i/ili obavijesti na predmetu.