**OOAD 2019-2020**

**Studentski projekat**

**Grupa2 – Zamger**

**Studenti:**

Paša Džumhur

Rijad Handžić

Huso Hamzić

# Uvod

Visokofunkcionalan informacioni sistem za visokoškolske ustanove koji olakšava dnevni workflow kako studenata tako i profesora. Aplikacija daje uvid u ostvarene rezultate studenta kao i mogućnosti da student prijavi ispite i odazove se na sve fakultetske obaveze i zadaće. Omogućava nastavnom osoblju da ima uvid i kontrolu nad predmetima na kojim su zaduženi kao i kreiranje ispitnih termina, zadaća za iste. Također studentskoj službi olakšava svakodnevne obaveze oko potvrda i papirologije.

# Klase

**Student:**

* Apstraktna klasa
* Atributi:
* ime (string)
* prezime (string)
* broj indeksa (int)
* datumRođenja (date)
* mjestoPrebivališta (string)
* odsjek (string)
* username (string)
* email (string)
* inbox (list<Poruka>)
* outbox (list<Poruka>)
* brojBesplatnihZahtjeva (int)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* posaljiZahtjev – formira zahtjev koji se šalje studentskoj službi na odobravanje

**BachelorStudent:**

* Izvedena klasa iz klase **Student**
* Atributi:
* predmeti (list<PredmetZaStudente>)
* prosjek (double)
* godinaStudija (string)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* dodajPredmet – dodaje novi predmet u listu predmeta koje student pohađa
* izbaciPredmet – briše predmet primljen kao parametar iz studentove liste predmeta kojih pohađa
* dajPorukuNaDatum – vraća listu poruka primljenih na datum poslan kao parametar

**MasterStudent:**

* Izvedena klasa iz klase **Student**
* Atributi:
* predmeti (list<PredmetZaStudente>)
* prosjek (double)
* godinaStudija (string)
* prosjekNaBSC (double)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* dodajPredmet – dodaje novi predmet u listu predmeta koje student pohađa
* izbaciPredmet – briše predmet primljen kao parametar iz studentove liste predmeta kojih pohađa
* dajPorukuNaDatum – vraća listu poruka primljenih na datum poslan kao parametar

**NastavnoOsoblje:**

* Apstraktna klasa
* Atributi:
* ime (string)
* prezime (string)
* datumRođenja (date)
* mjestoPrebivališta (string)
* username (string)
* email (string)
* inbox (list<Poruka>)
* outbox (list<Poruka>)
* titula (string)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* unesiPrisustvo – unosi prisustvo za određenog studenta na predmetu

**Profesor:**

* Izvedena klasa iz klase **NastavnoOsoblje**
* Atributi:
* predmetiNaKojimPredaje (list<PredmetZaProfesore>)
* aktivniTermini (list<Aktivnost>)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* objaviTermin – omogućuje profesoru da objavi termin za ispit ili za zadaću na koji se studenti mogu prijaviti
* kreirajAnketu – stvara studentsku anketu koju studenti koji pohađaju određeni predmet mogu ispuniti
* unesiRezultateIspita – unosi rezultate studenata koji su radili određeni ispit
* unesiRezultateZadace – unosi rezultate studenata koji su radili određenu zadaću
* prikaziRezultateAnkete – prikazuje prosjecnu ocjenu predmeta kao i studentske komentare za određeni predmet
* prikaziRezultateIspita – povezuje studente sa njihovim rezultatima ostvarenim na nekom ispitu
* prikaziRezultateZadace – povezuje studente sa njihovim rezultatima ostvarenim na nekoj zadaći
* izmjeniRezultateIspitaZaStudenta – u slučaju neke greške prilikom unosa omogućava korekciju iste
* izmjeniRezultateZadaćeZaStudenta – u slučaju neke greške prilikom unosa omogućava korekciju iste

**Asistent:**

* Izvedena klasa iz klase **NastavnoOsoblje**
* Atributi:
* predmetiNaKojimPredaje (list<PredmetZaProfesore>)
* aktivniTermini (list<Aktivnost>)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* prikaziRezultateIspita – povezuje studente sa njihovim rezultatima ostvarenim na nekom ispitu
* prikaziRezultateZadace – povezuje studente sa njihovim rezultatima ostvarenim na nekoj zadaći
* izmjeniRezultateZadaćeZaStudenta – u slučaju neke greške prilikom unosa omogućava korekciju iste
* objaviTermin – omogućuje asistentu da objavi termin za ispit ili za zadaću na koji se studenti mogu prijaviti
* unesiRezultateZadace – unosi rezultate studenata koji su radili određenu zadaću

**StudentskaSluzba:**

* Atributi:
* username (string)
* bazaConn (string)
* Metode:
* unesiStudenta – unosi studenta u bazu/sistem
* dodajPredmet – unosi novi predmet u bazu/sistem
* dodajNastavnoOsoblje – unosi nastavno osoblje u bazu/sistem
* kreirajObavjestenje – kreira obavještenje koje je zajedničko za sve korisnike u sistemu
* izmjeniStudenta – mijenja studentove lične podatke
* izmjeniNastavnoOsoblje – mijenja lične podatke od nastavnog osoblja
* prikaziStudenta – prikazuje studentove lične podatke kao i njegov ostvaren uspjeh na svim predmetima koje je pohađao
* dajZahtjeve – vraća listu zahtjeva podnesenih od strane studenta
* prikaziNastavnoOsoblje – prikazuje lične podatke člana nastavnog osoblja
* pretraziOsobu – nalazi osobu u bazi podataka po imenu i prezimenu i vraća listu osoba koje odgovaraju upitu
* prikaziAnkete – prikazuje rezultate anketa za sve predmete na fakultetu
* objaviRezultateAnketa – objavljuje rezultate studentskih anketa
* prikaziPendingAktivnosti – prikazuje sve podnesene termine i aktivnosti
* odobriAktivnost – odobrava aktivnost i „pušta“ je u sistem

**PredmetZaStudenta:**

* Atributi:
* naziv(string)
* dostupnost(map<string,int>)
* ectsPoeni (double)
* ocjena (int)
* bodovi (double)
* zadace (list<Aktivnost>)
* ispiti (list<Aktivnost>)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**PredmetZaProfesore:**

* Atributi:
* naziv(string)
* dostupnost(map<string,int>)
* ectsPoeni (double)
* studenti (map<Student, int>)
* zadace (list<Aktivnost>)
* ispiti (list<Aktivnost>)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**Aktivnost:**

* Apstraktna klasa
* Atributi
* naziv(string)
* pocetniDatum (Date)
* krajnjiDatum (Date)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**Ispit:**

* Izvedena klasa iz klase **Aktivnost**
* Atributi
* kapacitet (int)
* predmet (Predmet)
* maxBrojBodova (int)
* brojBodovaZaProlaz (int)
* salaOdrzavanja (string)
* studenti (list<Student>)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**Zadaca:**

* Izvedena klasa iz klase **Aktivnost**
* Atributi
* kapacitet (int)
* predmet (Predmet)
* maxBrojBodova (int)
* studenti (list<Student>)
* file (File)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**Poruka:**

* Atributi
* posiljalacID (int)
* primalacID (int)
* naslov (string)
* text (string)
* datumSlanja (Date)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**API –** kalendar

**Interface**  - trenutno nemamo opravdan razlog za korištenje, ali kako iterativno radimo na projektu postoji mogućnost da će se ukazati potreba za njim.

**Baza –** Kompleksnija baza podataka sa dosta tabela i međutabela koje će sadržavati informacije o svim studentima, nastavnom osoblju i predmetima te rezultate iz ispita i zadaća sa svih predmeta.