**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

**Korisnik:**

* Apstraktna klasa
* Atributi:
* ime (string)
* prezime (string)
* datumRođenja (date)
* mjestoPrebivališta (string)
* username (string)
* email (string)
* inbox (list<Poruka>)
* outbox (list<Poruka>)
* obavještenja (List<Obavještenje>)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* dodajPorukuUInbox - poruku primljenu kao parametar dodaje u inbox
* dodajPorukuUOutbox - poruku primljenu kao parametar dodaje u outbox
* dajPorukeNaDatum - vraća listu svih poruka primljenih/poslanih na datum primljen kao parametar
* dodajObavještenje - obavještenje primljeno kao parametar dodaje u atribut obavještenja

**Student:**

* Izvedena klasa iz klase Korisnik
* Atributi:
* broj indeksa (int)
* odsjek (string)
* predmeti(List<PredmetZaStudenta>)
* godinaStudija(int)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* dodajPredmet - dodaj predmet primljen kao parametar u listu predmeta koje student pohađa(atribut predmeti)

**MasterStudent:**

* Izvedena klasa iz klase **Student**
* Atributi:
* prosjekNaBSC (double)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**NastavnoOsoblje:**

* Izvedena klasa iz klase Korisnik
* Atributi:
* titula (string)
* aktivnosti(List<Aktivnost>)
* predmetiNaKojimPredaje

(List<PredmetZaNastavnoOsoblje>)

* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* dodajAktivnost - dodaje aktivnost primljenu kao parametar u atribut aktivnosti

**Profesor:**

* Izvedena klasa iz klase **NastavnoOsoblje**
* Atributi:
* ankete (List<Anketa>)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* dodajAnketu - anketu primljenu kao parametar dodaje u listu anketa (atribut ankete)

**StudentskaSluzba:**

* Atributi:
* username (string)
* email (string)
* idStudentske (int)
* zahtjevi (List<Zahtjev>)
* obavještenja (List<Obavještenja>)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* dajNeobrađeneZahtjeve - vraća listu neobrađenih zahtjeva
* dodajObavještenje - dodaje obavještenje primljeno kao parametar u listu obavještenja(atribut obavještenja)

**PredmetZaStudenta:**

* Atributi:
* naziv(string)
* ectsPoeni (double)
* ocjena (int)
* bodovi (double)
* aktivnosti (list<Aktivnost>)
* idPredmeta(int)
* idStudenta(int)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**PredmetZaNastavnoOsoblje:**

* Atributi:
* naziv(string)
* ectsPoeni (double)
* studenti (Map<int,Student>)
* aktivnosti (Map<int,Map<int,Aktivnost>>)
* idPredmeta(int)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* dajBrojStudenataNaPredmetu - vraća broj studenata na predmetu
* dodajRezultateZadaca - dodaje mapu rezultata jedne zadaće u atribut aktivnosti
* dodajRezultateIspita - dodaje mapu rezultata jednog ispita u atribut aktivnosti
* getIspitiNaPredmetu - vraća mapu ispita na predmetu
* getZadaceNaPredmetu - vraća mapu zadaca na predmetu

**Aktivnost:**

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

**Ispit:**

* Izvedena klasa iz klase **Aktivnost**
* Atributi
* idIspita (int)
* maxBrojBodova (int)
* brojBodovaZaProlaz (int)
* idStudenta(int )
* ostvareniBrojBodova(double)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**Zadaca:**

* Izvedena klasa iz klase **Aktivnost**
* Atributi
* maxBrojBodova (int)
* ostvareniBrojBodova (int)
* idStudenta(int)
* redniBrojZadace(int)
* rjesenjeZadace (File)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**Poruka:**

* Atributi
* posiljalac (int)
* primalac (int)
* naslov (string)
* sadržaj(string)
* datumSlanja (Date)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**Anketa:**

* Atributi
* naziv (string)
* datumIsteka(DateTime)
* pitanja (List<string>)
* idAnketa (int)
* komentar(string)
* odgovori(List<OdgovorNaAnketu>)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima
* dodajOdgovorNaAnketu - dodaje odgovor na anketu primjen kao parametar u atribut odgovori

**OdgovorNaAnketu:**

* Atributi
* idAnkete(int)
* idStudenta(int)
* odgovori (List<String>)
* komentar (String)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**Obavještenje:**

* Atributi
* naslov (String)
* sadržaj (String)
* vrijemeObavještenja (DateTime)
* Metode:
* Getteri i setteri
* Konstruktor sa svim parametrima

**API –** kalendar

**Interface**  - Mada trenutno ne koristimo nijedan interface, njihovu upotrebu ćemo zasigurno razmotriti pri implementiranju dizajn paterna u sljedećim laboratorijskim vježbama.

**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.