**Korisnik**

* Apstraktna klasa
* Svi akteri su izvedeni iz ove klase
* Atributi:
* username (string)
* password (string)
* Od metoda ova klasa posjeduje samo getere i setere za svoje metode
* Ova klasa ima kontruktor sa dva parametra, kao i konstruktor bez parametara koji je namijenjen za gosta koji još nema username i password

**Student**

* Izvedena iz klase Korisnik (generalizacija)
* Pored atributa iz klase Korisnik, klasa Student sadrži i sljedeće atribute:
* id (int)
* licniPodaci (LicniPodaci)
* prebivaliste (PrebivalisteInfo)
* skolovanje (SkolovanjeInfo)
* paviljon(Paviljon)
* brojRucaka(int)
* brojVecera(int)
* Ova klasa ima sljedeće metode:
* getere i setera za svaki atribut
* popuniZahtjevZaPremještanje - pozivom ove metode otvara se UI u kojem student unosi podatke koji su potrebni za premještanje
* popuniCimeraj - pozivom ove metode otvara se UI u kojem student unosi podatke koji su potrebni za cimeraj
* posaljiZahtjev - ova metoda se poziva unutar ovih metoda i šalje zahtjev u bazu podataka na daljnju obradu
* Ova klasa ima defaultni konstruktor sa svim parametrima koji odgovaraju atributima

**PrebivalisteInfo**

* Ova klasa nam služi za čuvanje informacija o prebivalištu studenata
* Atributi ove klase su:
* adresa (String)
* kanton (String)
* općina (String)
* Ova klasa od metoda sadrži samo getere i setere za atribute ove klase
* Ova klasa ima defaultni konstruktor sa svim parametrima koji odgovaraju atributima

**SkolovanjeInfo**

* Ova klasa nam služi za čuvanje informacija o fakultetu studenata
* Atributi ove klase su:
* fakultet (String)
* brojIndeksa (int)
* ciklusStudija (int)
* godinaStudija (int)
* Ova klasa od metoda sadrži samo getere i setere za atribute ove klase
* Ova klasa ima defaultni konstruktor sa svim parametrima koji odgovaraju atributima

**Pol (ENUM) {MUSKO, ZENSKO}**

**Uprava**

* Ova klasa je izvedena iz klase Korisnik (generalizacija)
* Atributi ove klase:
* blagajna (Blagajna)
* Metode ove klase su:
* otvoriListuStudenata - pozivom ove metode, prikazuje se lista svih studenata
* prikaziStanjePoSobama - pozivom ove metode, prikazuje se raspored studenata po sobama
* otvoriBlagajnu - ova metoda služi za prikaz glavnog menija blagajne
* pregledajZahtjeve - ova metoda služi za prikaz svih zahtjeva koji čekaju na obradu
* prikaziZahtjev - ova metoda služi za prikaz odabranog zahtjeva
* pregledKartice - ova metoda se poziva u slučaju da je u obradi zahtjev za upis i želimo vidjeti izgled kartice studenta
* pregledKorisnickihPodataka - ova metoda se poziva u slučaju da je u obradi zahtjev za upis i želimo vidjeti korisničke podatke
* odobriZahtjev
* odbijZahtjev
* upisiStudenta - ova metoda se poziva prilikom odobravanja zahtjeva za upis studenta. U ovoj metodi dodajemo studenta u bazu podataka, te vršimo ažuriranje smještajnih kapaciteta.
* Ova klasa ima konstruktor bez parametara koji kreira instancu klase Blagajna

**Restoran**

* Ova klasa je izvedena iz klase Korisnik (generalizacija)
* Ova klasa implementira interfejs PregledStanjaBonova, kao i interfejs AzurirajStanjeBonova
* Atributi ove klase su:
* dnevniMeni (DnevniMeni)
* idTrenutnogStudenta (int)
* Metode ove klase su:
* pregledDnevnogMenija - ova metoda se poziva u slučaju da želimo vidjeti dnevni meni
* azurirajDnevniMeni - ova metoda se poziva u slučaju da želimo vršiti ažuriranje dnevnog menija
* otvoriZahtjevZaNabavkuNaminica - ova metoda otvara pozor za popunjavanje zahtjeva za nabavku namirnica
* dodajRucak
* dodajVeceru
* izbaciRucak
* izbaciVeceru
* Ova klasa ima konstruktor bez parametara u kojem se samo kreira instanca klase DnevniMeni

**Zahtjev**

* Ovo je apstraktna klasa
* Ova klasa nam služi da bismo iz nje izveli ostale zatjeve
* Sadrži atribute:
* id (int)
* podnosilacZahtjeva (Korisnik)
* datum (DateTime)
* odobren (Bool)
* Metode ove klase:
* posaljiZahtjev - zahtjev se sprema u bazu podataka i čeka se na njegovo odobrenje
* geteri i seteri
* Ova klasa sadrži konstruktor koji prima prva tri atribura, a odobren se automatski stavlja na false

**ZahtjevZaPremještanje**

* Ova klasa je izvedena iz klase Zahtjev (generalizacija)
* Atributi ove klase su:
* trenutniPaviljon(Paviljon)
* trenutnaSoba (Soba)
* noviPaviljon (Paviljon)
* novaSoba (Soba)
* razlogPremjestanja (String)
* Ova klasa nema dodatnih metoda za razliku od bazne klase osim getera i setera
* Konstruktor ove klase ima za parametre pored svojih atributa, i atribute bazne klase. On vrši pozivanje konstruktora bazne klase.

**ZahtjevZaUpis**

* Ova klasa je izvedena iz klase Zahtjev (generalizacija)
* Atributi ove klase su:
* licniPodaci (LicniPodaci)
* prebivaliste (PrebivalisteInfo)
* skolovanje (SkolovanjeInfo)
* Ova klasa nema dodatnih metoda za razliku od bazne klase osim getera i setera
* Konstruktor ove klase ima za parametre pored svojih atributa, i atribute bazne klase. On vrši pozivanje konstruktora bazne klase.

**ZahtjevZaCimeraj**

* Ova klasa je izvedena iz klase Zahtjev (generalizacija)
* Atributi ove klase su:
* paviljon (Paviljon)
* soba (Soba)
* prviCimer (int)
* drugiCimer (int)
* dodatneNapomene (String)
* Ova klasa nema dodatnih metoda za razliku od bazne klase osim getera i setera
* Konstruktor ove klase ima za parametre pored svojih atributa, i atribute bazne klase. On vrši pozivanje konstruktora bazne klase.

**ZahtjevZaNabavkuNamirnica**

* Ova klasa je izvedena iz klase Zahtjev (generalizacija)
* Atributi ove klase su:
* listaNamirnica (List<String>)
* narudzba (List<StavkaNarudzbe>)
* Ova klasa nema dodatnih metoda za razliku od bazne klase osim getera i setera
* Konstruktor ove klase ima za parametre pored svojih atributa, i atribute bazne klase. On vrši pozivanje konstruktora bazne klase.

**StavkaNarudzbe**

* Ova klasa nam služi kao klasa koja nam povezuje namirnicu koju treba naručiti sa količinom koju treba naručiti
* Atributi:
* namirnica (String)
* kolicina (double)
* Metode ove klase su geteri i seteri za atribute ove klase
* Ova klasa ima defaultni konstruktor sa dva parametra

**Blagajna**

* Ova klasa implementira interfejs AzurirajStanjeBonova
* Ova klasa se koristi za regulisanje uplata smještaja studenata studenskog doma i regulisanje stanja bonova
* Atributi ove klase su:
* stanjeBudgeta (double)
* Metode ove klase su:
* provjeriID - provjerava ID studenta i na osnovu njega pronalazi studenta u bazi podataka
* uplatiDomZaOdabraniMjesec - ova metoda ažurira stanje budžeta i poziva metode za ažuriranje stanja bonova studenta
* ucitajStanjeBudzeta - čita stanje budžeta iz baze podataka
* Ova klasa ima konstruktor bez parametara unutar kojeg se poziva metoda ucitajStanjeBudzeta

**Paviljon**

* Ova klasa nam služi kao kontejnerska klasa koja na jednom mjestu čuva sobe jednog paviljona
* Atributi ove klase su:
* naziv (String)
* sobe (List<Soba>)
* Ova klasa od metoda ima samo getere i setere za atribute
* Klasa posjeduje defaultni konstruktor sa dva parametra

**Soba**

* Atributi ove klase su:
* brojSobe (int)
* studenti (List<Student>)
* Metode ove klase:
* geteri i seteri za atribute ove klase
* dodajStudentaUSobu
* izbaciStudentaIzSobe
* Ova klasa posjeduje defaultni konstruktor sa dva parametra, kao i konstruktor sa jednim parametrom u slučaju da je soba prazna

**DnevniMeni**

* Atributi ove klase su:
* listaRucaka (List<String>)
* listaVecera (List<String>)
* Metode ove klase su:
* geteri i seteri
* dodajRucak
* dodajVeceru
* izbaciRucak
* izbaciVeceru
* Ova klasa pored konstruktora sa dva parametra posjeduje i konstruktor bez parametara koji služi za kreiranje praznog dnevnog menija

**LicniPodaci**

* Cilj ove klase je čuvanje ličnih podataka na jednom mjestu, odnosno grupisanje podataka
* Atributi ove klase su:
* prezime (String)
* ime (String)
* mjesto rođenja (String)
* pol (Pol)
* email (String)
* JMBG (long)
* datum rođenja (DateTime)
* mobitel (int)
* slika (String)
* Metode ove klase su seteri i geteri za atribute
* Ova klasa sadrži defaultni konstruktor sa svim parametrima koji predstavljaju odgovarajuće atribute

**Interfejs:PregledStanjeBonova**

* Pomoću ovog interfejsa imamo uvid u stanje bonova određenog studenta
* Metode ove klase su:
* dajBrojRucakaZaStudenta
* dajBrojVeceraZaStudenta

**Interfejs:AzurirajStanjeBonova**

* Pomoću ovog interfejsa vršimo ažuriranje stanja bonova određenog studenta
* Metode ove klase su:
* azurirajStanjeRucaka
* azurirajStanjeVecera