

Opis dijagrama klase

Za implementaciju e-Teretane će biti potrebne sljedeće klase: Korisnik, Clan, Trener, Recepcioner, Administrator, Oprema, Trening, Teretana i Novost. Pored navedenih klasa, potrebni su i enumi TipClanarine koji ima 3 vrijednosti vezane za dužinu trajanja članarine (JEDNOMJESECNA, TROMJESECNA, SESTOMJESECNA), TipTreninga koji posjeduje dvije vrijednosti i to INDIVIDUALNI i GRUPNI te TipZauzetostiOpreme koji može poprimiti sljedeće vrijednosti: SLOBODNO, IZNAJMLJENO, ZAUZETO.

U nastavku ćemo navesti potrebne atribute za navedene klase.

Korisnik predstavlja apstraktnu klasu koja kao svoje atribute ima ime, prezime, e-mail, korisničko ime i šifru (svi navedeni atributi su tipa String). Klasu Korisnik nasljeđuju klase Clan, Trener, Recepcioner i Administrator.

Clan pored navedenih atributa, kao atribute ima i TipClanarine koji govori koji je trenutni tip članarine tog korisnika, zatim atribut tipa Date koji označava datum učlanjivanja kako bi se moglo odrediti kada ističe aktivna članarina te, atribut tipa int koji govori koliko puta je korisnik bio u teretani za aktuelni mjesec (resetuje se na 0 početkom novog mjeseca) te atribut trenutnoPrisutan tipa bool koji govori da li je korisnik trenutno u teretani.

Klasa **Trener** pored navedenih atributa, kao atribut ima i mapu ocjene ciji je ključ tipa DateTime, a vrijednost tipa int, koja predstavlja ocjene koje je trener dobijao od strane korisnika.

Klasa **Recepcioner** pored navedenih atributa ima i dva dodatna atributa tipa DateTime, koji predstavljaju početak odnosno kraj njegovog radnog vremena, kako bi njegov rad bio ograničen samo na taj vremenski period.

Klasa **Admin** pored navedenih atributa ima i dodatni atribut prijavljen tipa bool, čime se osigurava da čak i ako neko pokuša unijeti ispravne podatke ne može izvršiti login dok se trenutno logovani Admin ne izloguje sa sistema.

Klasa **Oprema** sadrži atribute nazivOpreme tipa String, id (šifru) opreme tipa int, tipZauzetosti tipa TipZauzetostiOpreme, opis opreme tipa String, atribut naCekanju tipa bool koji govori da li postoji neki zahtjev za iznajmljivanje ove opreme koji je još uvijek na listi čekanja za odobrenje, dva atributa tipa DateTime koji su null kada je oprema slobodna, predstavljaju datum iznajmljivanja i rok za vraćanje opreme koja je iznajmljena, a vrijeme početka i kraja korištenja u slučaju kada se oprema koristite atribut tipa Clan koji je također null kada je oprema slobodna, a u suprotnom se odnosi na člana koji je iznajmio ili koristi opremu.

Klasa **Trening** sadrži sljedeće atribute: atribut tipa DateTime koji označava datum i vrijeme održavanja treninga, atribut kapacitet tipa int koji označava maksimalan broj ljudi koji se mogu prijaviti na taj trening, atribut opis tipa String koji sadrži opis treninga, atribut tipa Trener koji predstavlja trenera odgovornog za taj trening, listu Clanova koji su prijavljeni na taj trening te atribut tipa TipTreninga.

Klasa **Novost** ima 3 atributa tipa string, koji predstavljaju naziv novosti, tekst novosti te putanju za sliku vezanu za novost.

Klasa **Teretana** se sastoji od sljedećih atributa: admin tipa Administrator, mapu racuniClanova kod koje je ključ tipa Clan, a vrijednost tipa String i predstavlja broj racuna člana, lista čiji su članovi tipa Clan, lista čiji su članovi tipa Korisnik, lista čiji su članovi tipa Oprema, mapa zahtjeviZaIznajmljivanje koja kao ključ ima korisnika koji je poslao zahtjev, a kao vrijednost ima Opremu za koju je zahtjev poslat, kao i dvije liste čiji su članovi tipa Trening gdje jedna lista predstavlja aktuelne treninge, a druga treninge koji su se već održali.

Nakon što smo pobrojali attribute za klase potrebne za naš sistem, sada ćemo preći na njihove funkcionalnosti.

Metode klase Korisnik su sljedeće:

- Konstruktor sa 5 parametara tipa String koji odgovaraju njegovim atributima
- Getteri za sve navedene attribute
- Setteri za sve attribute osim za atribut korisnickoIme koji se ne može mijenjati

Metode klase Clan su sljedeće:

- Konstruktor sa 8 parametara, gdje 5 tih parametara odgovaraju atributima naslijeđenim iz klase Korisnik, a preostala 3 su za tipClanarine, datumUclanjivanja te brojDolazaka
- Getteri i setteri za 3 atributa karakteristična samo za ovu klasu
- Metodu **obnoviClanarinu** koja kao parametre prima novi datum i novi tip članarine za korisnika, nema povratnog tipa nego samo mijenja vrijednosti atributa.

Metode klase Recepcioner su sljedeće:

- Konstruktor sa 7 parametara koji odgovaraju atributima klase Recepcioner
- Gettere i Settere za dodatne attribute

Metode klase Trener su sljedeće:

- Konstruktor sa 5 parametara koji odgovaraju atributima klase Korisnik
- Gettere i Settere za dodatne attribute
- Metoda dodajOcenu koja kao parametre prima datum na koji je ocjena unesena i vrijednost ocjene, nema povratnog tipa nego samo dodaje novu ocjenu u mapu postojećih ocjena za datog trenera

Metode klase Administrator su sljedeće:

- Konstruktor sa 5 parametara koji odgovaraju atributima klase Korisnik
- Getter i Setter za dodatni atribut

Metode klase Oprema su sljedeće:

- Konstruktor sa parametrima koji odgovaraju atributima same klase
- Getteri i setteri za attribute

Metode klase Trening su sljedeće:

- Konstruktor sa parametrima koji odgovaraju atributima same klase

- Metodu **prijaviClana** koja vrši dodavanje člana primljenog kao parametar u listu prijavljenih članova za trening
- Gettere i settere za attribute

Metode klase Novost su sljedeće:

- Konstruktor sa parametrima koji odgovaraju atributima same klase
- Gettere i settere za attribute

Metode klase Teretana su sljedeće:

- Metode **dodajClana** i **dodajZaposlenog** koje kao parametre primaju podatke dovoljne da se kreira novi objekat odgovarajućeg tipa i doda u odgovarajuću listu
- Metode **dajSlobodnuOpremu**, **dajIznajmljenuOpremu**, **dajOpremuKojaSeKoristi** koje vraćaju respektivno liste koje se sastoje od opreme koja je slobodna za korištenje ili iznajmljivanje, zatim opreme koja je već iznajmljena i opremu koja se trenutno koristi
- Metodu **dajIznajmljenuOpremuZaClana** koja kao parametar prima objekat tipa Clan, a vraća listu opreme koju je on iznajmio
- Metode **dajAktivneTreningeZaClana** i **dajProsleTreningeZaClana** koje primaju kao parametar objekat tipa Clan, a vraćaju listu treninga koje respektivno sadrže treninge za koje je član trenutno prijavljen i treninge kojima je prisustvovao u prošlosti
- Metode **dajAktivneTreningeZaTrenera** i **dajProsleTreningeZaTrenera** koje kao parametar primaju objekat tipa Trener, a vraćaju listu treninga koje respektivno sadrže treninge za koje je član trenutno prijavljen i treninge koje je organizovao u prošlosti
- Metodu **iznajmiOpremu** koja kao parametre prima objekat tipa Clan koji označava člana koji želi iznajmiti opremu, kao i šifru opreme koju želi da iznajmi. Metoda vrši dodavanje parametara u mapu koja sadrži aktuelne zahtjeve za iznajmljivanje, te postavlja opremi atribut naCekanju na true.
- Metodu **odobriIznajmljivanje** koja kao parametar prima Clana koji je poslao zahtjev za iznajmljivanje, kao i opremu za koju je poslao zahtjev. U metodi se vrši uklanjanje ovog zahtjeva iz liste zahtjeva te promjena atributa opreme naCekanju na false, tipZauzetosti na IZNAJMLJENO, a atribut tipa Clan klase Oprema mijenja na člana koji ju je iznajmio. Koristi se Send Mail API za obavješćavanje korisnika kada se približi rok za vraćanje opreme.
- Metodu **zauzmiOpremu** koja ima iste parametre kao i metoda iznajmiOpremu, a vrši mijenjanje atributa tipZauzetosti na ZAUZETO, a atribut tipa Clan klase Oprema postavlja na člana koji ju trenutno koristi
- Metodu **oslobodiOpremu** koja prima iste parametre kao i metoda zauzmiOpremu, a služi za registrovanje vraćene opreme koja je bila iznajmljena ili za oslobađanje opreme koja se koristila (mijenja tipZauzetosti opreme na SLOBODNO)
- Metodu **dodajTrening** koja kao parametar prima trening koji se dodaje, te vrši dodavanje treninga u listu aktuelnih treninga
- Metodu **osvjeziTreninge** koja provjerava da li je prošao termin nekog od treninga (na osnovu trenutnog datuma i vremena) iz liste aktuelnih treninga, te isti prebacuje u listu prošlih treninga.

- Metodu **prijaviClanaNaTrening** koja kao parametre prima Clana koji se prijavljuje za trening te sam Trening na koji se prijavljuje. Vršiti dodavanje navedenog člana u listu prijavljenih članova unutar objekta tipa Trening.
- Metodu **dajDanasnjeTreninge** bez parametara koja vraća listu treninga koji se održavaju na današnji dan, a filtriraju se na osnovu trenutnog datuma
- Metodu **posaljiPristupnePodatke** koja prima parametar tipa Korisnik, te vrši slanje informacija o korisničkom imenu i šifri na e-mail novododanom korisniku (neovisno od toga da li je u pitanju dodavanje novog člana od strane recepcionera ili dodavanje novog člana/recepcionera/trenera od strane administratora sistema). Slanje e-maila će se vršiti uz pomoć Send Email API-a.
- Metodu **evidentirajClana** koja kao parametre prima člana koji je došao u teretanu i vrijednost true/false koja označava da li se korisnik prijavljuje ili odjavljuje iz teretane, te mu uvećava mu brojač koji broji prisustvo za 1 ukoliko je drugi parametar funkcije true i postavlja mu atribut trenutnoPrisutan na drugi parametar funkcije.
- Metode **dajPrisutneClanove** i **dajOdsutneClanove** koja vraća listu članova ali filtriranu na osnovu atributa Člana trenutnoPrisutan.
- Metodu **ukloniKorisnika** koja kao parametar tipa korisnika te ga uklanja iz odgovarajuće liste
- Metodu **promijeniAttributeKorisnika** koja kao parametre prima starog i novog Korisnika, te vrši zamjenu atributa starog korisnika sa atributima novog korisnika
- Metodu **dodajRacun** koja prima kao parametre Clana i String koji predstavlja njegov racun, te ih dodaje u mapu racuniClanova.
- Metodu **dajRacunZaClana** koja kao parametar prima člana a vraća String koji predstavlja njegov račun.
- Metodu **dajRacune** koja vraća mapu svih računa.
- Metodu **dajNovosti** koja vraća listu svih novosti.
- Metodu **dodajNovost** koja prima parametar tipa Novost, te vrši dodavanje nove novosti u listu postojećih novosti
- Metodu **ocijeniTrenera** koja prima parametar tipa Trener koji predstavlja trenera na kojeg se odnosi ocjena, parametar tipa int koji predstavlja visinu ocjene i parametar tipa DateTime koji predstavlja datum kada je ocjena unesena. Metoda nema povratnog tipa, već samo dodaje novu ocjenu u mapu ocjena za odgovarajućeg trenera
- Gettere i settere za attribute

Odnosi između klasa su sljedeći:

- Apstraktna klasa Korisnik predstavlja generalizaciju klasa Clan, Recepcioner, Trener i Administrator koje ju nasljeđuju
- Klasa Oprema u sebi sadrži atribut tipa Clan koji u slučaju kada je oprema iznajmljena ili se koristi predstavlja člana koji je iznajmljuje/koristi. U pitanju je agregacija jer klasa Clan postoji neovisno od klase Oprema.
- Klasa Trening sadrži atribut tipa Trener. U pitanju je kompozicija, jer svaki trening ima svog odgovornog trenera.
- Klasa Trening sadrži listu članova prijavljenih na trening. U pitanju je agregacija.
- Klasa Teretana u sebi sadrži kolekcije klasa Clan, Korisnik, Trening, Oprema i Novost. U pitanju je kompozicija jer te klase ne mogu postojati bez teretane.

- Klasa Teretana sadži i atribute tipa Administrator i RacuniClanova. U pitanju je kompozicija jer te klase ne mogu postojati bez teretane.
- Klasa Teretana u svojim metodama koristi instance klase Trener. U pitanju je asocijacija.