

### **REC-S - servis za recenzije**

- Servis zadužen za ocene i komentare (zajedno - recenzije) koje ostavljaju korisnici sistema.
- Operacije koje servis vrši: dodavanje, čitanje i editovanje recenzija.
- U klasi RecenzijeController će se nalaziti endpoint-i:
  - makeRecenziju(RecenzijaDTO recenzijaData) - komunicira sa servisom radi ostavljanja recenzije (koristi korisnik)
  - getAllRecenzijeByOglasId(Long id) - komunicira sa servisom radi vraćanja svih recenzija po oglasu
  - leaveKomentar(KomentarDTO komentarData) - komunicira sa servisom radi ostavljanja komentara (koristi agent)
  - approveRecenzija(RecenzijaDTO recenzijaData) - komunicira sa servisom radi editovanja statusa recenzije (koristi administrator)
- U klasi RecenzijeService će se nalaziti implementirane metode:
  - leaveRecenzija(RecenzijaDTO recenzijaData) - metoda odgovara leaveRecenzija endpoint-u i služi za čuvanje recenzije.
  - getAllRecenzijeByOglasId(Long id) - metoda odgovara getAllRecenzijeByOglasId endpoint-u i služi za vraćanje svih recenzija ostavljenih na nekom oglasu.
  - leaveKomentar(KomentarDTO komentarData) - metoda odgovara leaveKomentar endpoint-u i služi za čuvanje komentara koje ostavljanju agenti kao odgovore na ostavljene komentare na svojim oglasima.
  - approveRecenzija(RecenzijaDTO recenzijaData) - metoda odgovara approveRecenzija endpoint-u i služi za prihvatanje ili odbijanje ostavljene recenzije od strane administratora. Jedini izmenjeni atribut recenzija je flag "approved".
- U klasi RecenzijeRepository će se nalaziti navedne metode koje nisu deo standardnih metoda jpa repozitorijuma, a to je u ovom slučaju findAllRecenzijeByOglasId(Long id) gde će se pretraga recenzija izvršiti preko id-a oglasa.

### **ZAH-S - servis za zahteve i razmenu poruka**

- Servis zadužen za rukovanje zahtevima i porukama.
- Operacije koje servis vrši: dodavanje, čitanje i editovanje zahteva, kao i dodavanje i čitanje razmenjenih poruka.
- U klasi ZahtevController će se nalaziti endpoint-i:
  - makeZahtev(ZahtevDTO zahtevData)
  - changeZahtevStatus(ZahtevDTO zahtevData)
  - getZahtevByUserId(Long id)
  - getZahtevByOglasId(Long id)
  - cancelZahtev(ZahtevDTO zahtevData)
  - approveZahtev(ZahtevDTO zahtevData)
- U klasi ZahtevService će se nalaziti implementirane metode:
  - makeZahtev(ZahtevDTO zahtevData) - metoda odgovara makeZahtev endpoint-u i služi za čuvanje poslatog zahteva.
  - changeZahtevStatus(ZahtevDTO zahtevData) - metoda odgovara changeZahtevStatus endpoint-u i služi za izmenu statusa zahteva posle plaćanja.
  - getZahtevByUserId(Long id) - metoda odgovara getZahtevByUserId endpoint-u i služi za prikaz svih zahteva prosleđenog korisnika (istorija zahteva korisnika).

- `getZahtevByOglasId(Long id)` - metoda odgovara `getZahtevByOglasId` endpoint-u i služi za prikaz svih zahteva koji postoje za neki određeni oglas.
- `cancelZahtev(ZahtevDTO zahtevData)` - metoda odgovara `cancelZahtev` endpoint-u i služi za otkazivanje zahteva od strane korisnika.
- `approveZahtev(ZahtevDTO zahtevData)` - metoda odgovara `approveZahtev` endpoint-u i služi za prihvatanje ili odbijanje primljenog zahteva od strane agenta. Jedini izmenjeni atribut recenzija je flag "approved".

- U klasi `ZahtevRepository` će se nalaziti navedne metode koje nisu deo standardnih metoda jpa repozitorijuma, a to su u ovom slučaju:

- `findZahtevByUserId(Long id)` - vrši pretragu zahteva preko korisničkog id-a
- `findZahtevByOglasId(Long id)` - vrši pretragu zahteva preko id-a oglasa.

- U klasi `ChatController` nalaziće se endpoint-i:

- `sendPoruka(PorukaDTO porukaData)`
- `getPorukaById(Long id)`

- U klasi `ChatService` će se nalaziti implementirane poruke:

- `savePoruka(PorukaDTO porukaData)` - metoda odgovara `savePoruka` endpoint-u i služi za čuvanje poruke.
- `getPorukaById(Long id)` - metoda odgovara `getPorukaById` endpoint-u i služi za prikaz poruka.

### **AUT-S - servis za autentifikaciju**

-Servis koji služi za autentifikaciju korisnika.

-Koristi se jwt token i spring security.

- U klasi `AuthController` se nalazi endpoint-i:

`@PostMapping("/signin")`

`public ResponseEntity<?> authenticateUser(@Valid @RequestBody LoginForm loginRequest)`

- Koji prepoznaje korisnika iz baze i omogućava korisniku da pristupi sistemu.

`@PostMapping("/signup")`

`public ResponseEntity<?> registerUser(@Valid @RequestBody SignUpForm signUpRequest)`

- Registruje novog korisnika koji onda može da se prijavi. Email je jedinstven i nije dozvoljeno da se korisnik registruje sa mail-om koji već postoji u bazi.

Posle uspešnog kreiranja korisnika aut-s ce komunicirati sa mail-s radi slanja email-a o uspešnoj registraciji.

### **MAIL-S - servis za slanje mail-ova**

Servis koji šalje mailove, detalje dobija od drugih servisa.

Koristiće se message queue za komunikaciju između servisa.

- `sendMail(String sender,String receiver, String Head, String body)` - ova metoda šalje poruku odgovarajućem primaocu.

## **VOZ-S - servis za vozila**

- Servis zadužen za vozila dostupna u rent-a-car ponudi.
- Operacije koje servis vrši: dodavanje, pretragu i editovanje vozila.
- U klasama MarkaVozila, ModelVozila, KlasaVozila, TipMenjaca, VrstaGoriva ce se nalaziti:
  - createMarkaVozila - napravi novu marku vozila
  - deleteMarkaVozilaById - brise marku vozila po id-u
  - getAllMarkaVozilaById - komunicira sa servisom radi vraćanja svih marki vozila po id-u
  - editMarkaVozilaById - menja marku vozila, trazi je po id-u

(Sve nabrojane klase imaju ove metode.)

- U klasi VoziloService će se nalaziti:
  - createVozilo - napravi novo vozilo
  - deleteVoziloById - obriše postojeće vozilo
  - editVoziloById - promeni karakteristike vozila
  - findVoziloById - pronadje vozilo preko id-a
  - findAllVoziloByModel - pronadje sva vozila po modelu
  - findAllVoziloByMarka - pronadje sva vozila po marki automobila
  - findAllVoziloByTipMenjaca - pronadje sva vozila po tipu menjaca
  - findAllVoziloByVrstaGoriva - pronadje sva vozila po vrsti goriva
  - findAllVoziloByKlasa - pronadje sva vozila po klasi automobila
  - updateKilometrazaById - Izmjena kilometraze. Vrsi se posle koriscenja vozila.

U klasi IzvestajService:

- createIzvestaj - Pravljenje novog izvestaja
- deleteIzvestajById - Brisanje postojećeg izvestaja po Id-u
- getAllIzvestajByVoziloId - pretraga izvestaja po Id-u vozila
- readIzvestajById - pretstavi postojeći izvestaj radi citanja.

U klasi CenovnikService:

- getAllCenaPoDanu - izlistavanje svih cena
- deleteCenaPoDanu - brisanje postojećih cena
- createCenaPoDanu - pravljenje novih cena

U klasi PopustService:

- createPopust - kreiranje novog popusta
- deletePopust - brisanje postojećeg popusta
- applyPopust - apliciranje popusta na cenu

U klasi OglasService:

- createOglas - kreiranje novog oglasa
- editOglasById - editovanje postojećeg oglasa Id-u
- deleteOglasById - brisanje postojećeg oglasa po Id-u
- getAllOglasByVoziloId - listanje svih oglasa po vozilo Id-u