Završni test JWD 61-62

Korišćenjem radnih okvira Spring Boot, React i Bootstrap razviti Web aplikaciju za azil i usvajanje životinja (PetAdopt).

1.1 Aplikacija bi trebalo da obezbedi rad sa sledećim entitetima, uz čije atribute su data i ograničenja na nivou baze podataka:

Kategorija:

ld - identifikator

Naziv - tekstualna vrednost, **obavezna, jedinstvena** (npr. pas, mačka)

Ljubimac (Pet):

ld - identifikator

Ime - tekstualna vrednost, obavezna

Starost - numerička oznaka izražena brojem meseci

Vakcinisan - boolean vrednost, obavezna

Pol - tekstualna vrednost, **obavezna** (muški/ženski)

Težina - numerička oznaka izražena u kg, obavezna (npr. 2kg, 0.8kg...)

Opis - tekstualna vrednost, obavezna

Kateogorija - veza sa klasom Kategorija (svaki ljubimac pripada tačno jednoj kategoriji, dok jedna kategorija može da sadrži više ljubimaca)

Udomljavanje:

Id - identifikator

Datum udomljavanja ljubimca - datum i vreme, **obavezna** vrednost Ljubimac - veza sa klasom Ljubimac (u okviru udomljavanja moguće je udomiti tačno jednog ljubimca)

1.2 Pomoću radnog okvira Spring Boot implementirati sledeći REST API:

GET /api/kategorije - preuzimanje svih kategorija	svi
GET /api/ljubimci - preuzimanje svih ljubimaca (paginirano)	svi
GET /api/ljubimci/{id} - preuzimanje ljubimca po zadatom id-u	admin, user
POST /api/ljubimci - dodavanje novog ljubimca	admin, user
PUT /api/ljubimci/{id} - izmena postojećeg ljubimca	admin, user
DELETE /api/ljubimci/{id} – brisanje postojećeg ljubimca	admin

1.3 Na nivou API-ja validirati sledeće stavke:

Ime ljubimca - tekstualna vrednost, ne sme biti prazan string Težina - numerička nenegativna vrednost (ne sme biti nula ili negativna)

2.1 Obezbediti unos novog ljubimca na **zasebnoj stranici** (slika 1). Prilikom kreiranja novog ljubimca administrator unosi podatke o ljubimcu kroz formu. Prilikom unosa podataka o ljubimcu, podrazumeva se da je ljubimac nevakcinisan tj. ta vrednost se ne prosleđuje prilikom kreiranja. Na stranici za prikaz i pretragu ljubimaca postoji dugme za prelazak na stranicu za unos novog ljubimca (slika 2).

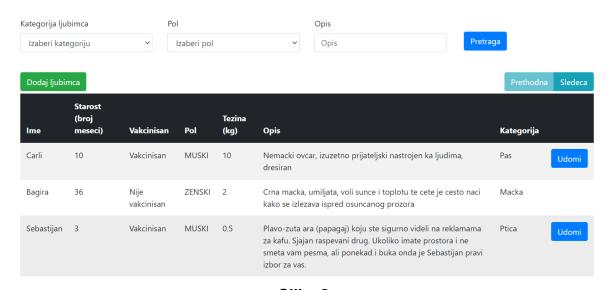
Nakon unosa vrednosti u polja prikazana na slici 1 i klika na dugme "*Dodaj ljubimca*", on se preko API-ja dodaje u aplikaciju i potom se korisnik vraća na stranicu za prikaz i pretragu.

lme		
Ime ljubimca		
Starost		
Starost (broj meseci)		
Pol		
Izaberi pol ljubimca		~
Tezina		
Tezina (kg)		
Opis		
Opis ljubimca		
Kategorija		

Slika 1.

- 2.2 Obezbediti prikaz svih ljubimaca. U svakom redu tabele ispisati podatke o ljubimcima kao što je dato na slici 2. Za polje vakcinisan, koje se u bazi čuva kao boolean vrednost, u tabeli je potrebno ispisati rečima *Vakcinisan/Nije vakcinisan*.
- 2.3 Obezbediti brisanje ljubimaca. Ova funkcionalnost dostupna je samo administratorima sistema te i dugme treba da bude vidljivo samo ako je prijavljeni korisnik administrator (dugme je dato na slici 3 koja predstavlja prikaz svih ljubimaca ukoliko je ulogovani korisnik administrator sistema). Nakon klika na dugme "Obriši" u tabeli za prikaz svih ljubimaca, on se briše iz aplikacije i uklanja iz tabele.
- 2.4 Nadograditi prikaz svih tako da podrži paginaciju. Dugme "*Prethodna*" i dugme "*Sledeća*" iznad tabele za prikaz ljubimaca (slika 2) omogućuju promenu stranice. Ukoliko se korisnik nalazi na prvoj stranici onemogućiti dugme prethodno, a ukoliko se nalazi na poslednjoj onemogućiti dugme sledeće. *Paginacija se vrši na back-end delu aplikacije*.

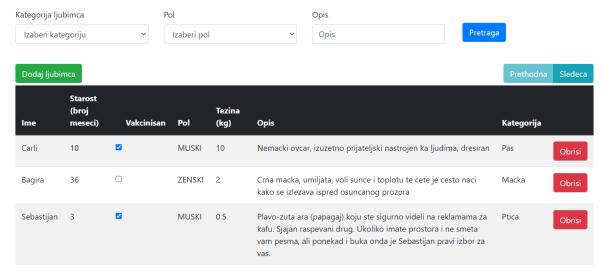
2.5 Obezbediti pretragu ljubimaca, preko forme za filtriranje koja je prikazana u okviru slike 2. Prilikom pretrage korisnik može da filtrira po polu (izbor između ponuđenih vrednosti *muško/žensko*), kategoriji ljubimca (izbor jedne od dostupnih kategorija) i opisu (za tekst koji se unese u polje za pretragu treba proveriti da li je sadržan u opisu ljubimca, a ne da je potpuno jednak). Pronalaze se i prikazuju ljubimci koje zadovoljavaju ove kriterijume. Ukoliko korisnik neko polje ne unese, vrednost tog polja se ignoriše u pretrazi. Pretraga se vrši na *back-end* delu aplikacije.



Slika 2.

- 2.5 Kompletnu izmenu podataka o ljubimcu **nije potrebno** realizovati na *front-end* strani, putem popunjene forme. Izmenu kompletnog objekta implementirati na *back-end* delu aplikacije.
- 3. Implementirati funkcionalnost vakcinacije ljubimaca za administratore:
 - a) Ukoliko je ulogovan administrator potrebno je u prikazu svih (tabela) umesto reči Vakcinisan/Nije vakcinisan prikazati checkbox, kao na slici 3 u nastavku. Checkbox je označen ukoliko je dati ljubimac vakcinisan ili neoznačen ukoliko ljubimac nije vakcinisan.
 - b) Klikom na checkbox potrebno je izmeniti stanje polja vakcinisan za izabranog kućnog ljubimca. Ukoliko se označi promeniti vrednost polja vakcinisan na true, u suprotnom na false.

Napomena: i dalje ako ulogovani korisnik nije administrator prikazivati za tražena polja tekst Vakcinisan/Nije vakcinisan.



Slika 3

- 4. Implementirati udomljavanje ljubimca za obične korisnike (ukoliko je ulogovani korisnik administrator dugme ne treba da bude vidljivo). Klikom na dugme "*Udomi*" omogućuje se udomljavanje ljubimca, gde korisnik može da udomi ljubimca samo ukoliko je vakcinisan. Ukoliko ljubimac nije vakcinisan, dugme "*Udomi*" treba da je sakriveno (kao što je slučaj sa drugim redom u tabeli datoj u primeru na slici 2). Kreirati u bazi novi objekat Udomljavanje za odabranog ljubimca.
- 5. Aplikacija bi trebalo da bude obezbeđena i sa *back-end*, i sa *front-end* strane što znači da samo prijavljeni korisnik može da joj pristupi. Korisnik se prijavljuje na sistem preko *login* stranice. Za ovaj deo, dozvoljeno je koristiti postojeće materijale kao takve. HTTPS nije potrebno iskonfigurisati. Role koje mogu pristupiti određenim endpoint-ima aplikacije su date kod definicije API-a.

Obavezno uz rešenje priložiti i test podatke za bazu podataka.