

Priprema za završni test

Korišćenjem radnih okvira Spring Boot, AngularJS i Bootstrap razviti Web aplikaciju za vođenje evidencije o automobilima Rent a Car kompanija.

1.1 Aplikacija treba da obezbedi rad sa sledećim entitetima, uz čije atribute je data i validacija na nivou baze podataka:

Kompanija:

- Id - identifikator
- Naziv - tekstualna vrednost
- Adresa – tekstualna vrednost
- Telefon – tekstualna vrednost

Automobil:

- Id – identifikator
- Model – tekstualna **obavezna** vrednost
- Registracija – tekstualna **obavezna i jedinstvena** vrednost
- Godište – celobrojna **obavezna** vrednost
- Potrošnja – numerička vrednost
- Iznajmljen – boolean vrednost
- Kompanija - veza sa instancom klase Kompanija (jedan automobil može pripadati samo jednoj kompaniji, a jedna kompanija može imati više automobila). Suprotan deo veze kreirati kod kompanije.

1.2 Pomoću radnog okvira Spring Boot implementirati sledeći REST API:

- GET /api/kompanije - preuzimanje svih kompanija
- GET /api/kompanije/{id} - preuzimanje kompanija po zadatom id-u
- GET /api/automobili - preuzimanje svih automobila
- GET /api/automobili/{id} - preuzimanje automobila po zadatom id-u
- POST /api/automobili - dodavanje novog automobila
- PUT /api/automobili/{id} - izmena postojećeg automobila
- DELETE /api/automobili/{id} - brisanje postojećeg automobila
- GET /api/kompanije/{id_kompanije}/automobili - preuzimanje svih automobila za zadatu kompaniju

1.3 Na nivou API-ja validirati sledeće sledeće stavke:

- Da model automobila nije prazan string ili null vrednost
- Da je potrošnja maksimalno dvocifren broj
- Da je godište maksimalno četvocifreni broj.

2.1) Obezbediti unos novog automobila. Nakon unosa vrednosti u polja prikazana na slici 1. i klika na dugme dodaj, automobil se preko API-ja dodaje u aplikaciju i izmena se prikazuje u tabeli na slici 2. Podaci o novododatom automobilu se brišu iz forme. **Iznajmljenost se ne dodaje preko forme, i inicijalno je false!**

Model

Registracija

Godište

Potrošnja

Kompanija

Dodaj

Slika 1

Model:

Godište od:

Potrošnja do:

Pretraži

Slika 2

Nazad

Napred

Model	Registracija	Godište	Potrošnja	Kompanija	
Nissan Prairie	Su-123-fd	1991	10 l.	SuRent	<div><div>Izmeni</div><div>Obriši</div><div>Iznajmi</div></div>
Nissan Micra	Ns-021-nm	2007	8 l.	NsRent	<div><div>Izmeni</div><div>Obriši</div><div>Iznajmi</div></div>
Tesla Roadster	Su-111-aa	2017	0 l.	SuRent	<div><div>Izmeni</div><div>Obriši</div><div>Iznajmi</div></div>

Slika 3

2.2) Obezbediti brisanje automobila. Nakon klika na dugme obriši u tabeli za prikaz automobila (slika 3.), automobil se briše iz aplikacije i uklanja iz tabele.

2.3) Obezbediti izmenu automobila u zasebnoj stranici. Nakon klika na dugme izmeni, podaci o automobilu se prikazuju u zasebnoj stranici koja omogućava čuvanje izmena. Ukoliko korisnik odabere da sačuva podatke, izmenjeni podaci se čuvaju u aplikaciji i prelazi se na stranicu za unos i prikaz automobila. Dizajn zasebne stranice za izmenu automobila prepušten je polazniku kursa. **Vrednost polja iznajmljen se ne može menjati preko forme za izmenu.**

2.4) Obezbediti pretragu automobila, preko forme (slika 2). Prilikom pretrage korisnik može da unese model, donju granicu za godište ili gornju granicu potrošnje. Pronalaze se i prikazuju oni automobili koji zadovoljavaju sve kriterijume. Ukoliko korisnik neko polje ne unese, vrednost tog polja se ignoriše u pretrazi. Filtriranje se vrši na back-end delu aplikacije.

2.5) Obezbediti paginirani prikaz podataka. Dugme prethodno i dugme sledece iznad tabele za prikaz automobila (slika 3) 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 na back-end delu aplikacije.

3. Implementirati iznajmljivanje automobila, klikom na dugme “**Iznajmi**”, prikazano na slici 3. To podrazumeva da se polje Iznajmljen tog automobila pretvori u vrednost *true*. Ako je automobil iznajmljen, dugme “**Iznajmi**” je *disabled*. Pored toga je u okviru procesa iznajmljivanja potrebno kreirati zaseban entitet **Najam**, koji sadrži samo vezu ka entitetu automobila koji je iznajmljen. Implementacija iznajmljivanja bi trebalo da bude na nivou servisa na bekendu.