



Dokumentacija

Web programiranje ASP

API Projekat

David Stević 58/20

Internet tehnologije – Web Programiranje

Sadržaj

1. Uvod.....	4
1.1 Korišćeni programski jezici i biblioteke	4
1.2 Tema API-ja	4
1.3 Opis funkcionalnosti.....	4
2. Organizacija	5
2.1 Baza	5
3. Uputstvo.....	6
3.1 Kreiranje baze i promena mesta connection stringu:	6
3.2 Kreiranje inicijalnih podataka	7
4. Funkcionalnosti	7
4.1 Registracija korisnika	7
4.1.1 Registracija korisnika	7
4.2 Vozila	7
4.2.1 Dohvatanje vozila	7
4.2.2 Dohvatanje specifičnog vozila	7
4.2.3 Dodavanje vozila	7
4.2.4 Izmena vozila	8
4.2.5 Brisanje vozila.....	8
4.3 Iznajmljivanje	8
4.3.1 Iznajmnjivanje vozila	8
4.3.2 Otkazivanje rente	8
4.4 Cene	8
4.4.1 Dodavanje cene.....	8
4.5 Popust	8
4.5.1 Izmena popusta.....	8
4.6 Specifikacije.....	8
4.6.1 Brisanje određene specifikacije	8
4.7 Korisnik.....	8
4.7.1 Dohvatanje korisnika	8
4.7.2 Izmena korisnika.....	9
4.7.3 Brisanje korisnika	9
4.8 Token	9

4.8.1	Pravljenje tokena.....	9
4.8.2	Brisanje tokena.....	9
4.9	Audit Log	9
4.9.1	Pregled audit log-a	9

1. Uvod

1.1 Korišćeni programski jezici i biblioteke

Za potrebe kreiranja API-ja su korišćene sledeće tehnologije:

- C#
- Fluent Validation
- Entity Framework Core, Tools, SQL Server, Poxies
- JWT Bearer

1.2 Tema API-ja

Tema koju obrađuje ovaj API je sistem za rentiranje automobila. Sistem za rentiranje automobila omogućava korisnicima da iznajme automobile za određeni period vremena. Ovaj sistem olakšava proces rezervacije i upravljanja vozilima za iznajmljivanje.

Baza podataka: Sistem koristi bazu podataka za skladištenje informacija o automobilima, korisnicima, rezervacijama, cenama, dostupnosti vozila i ostalim relevantnim podacima. Baza podataka omogućava efikasno čuvanje, pretragu i ažuriranje podataka.

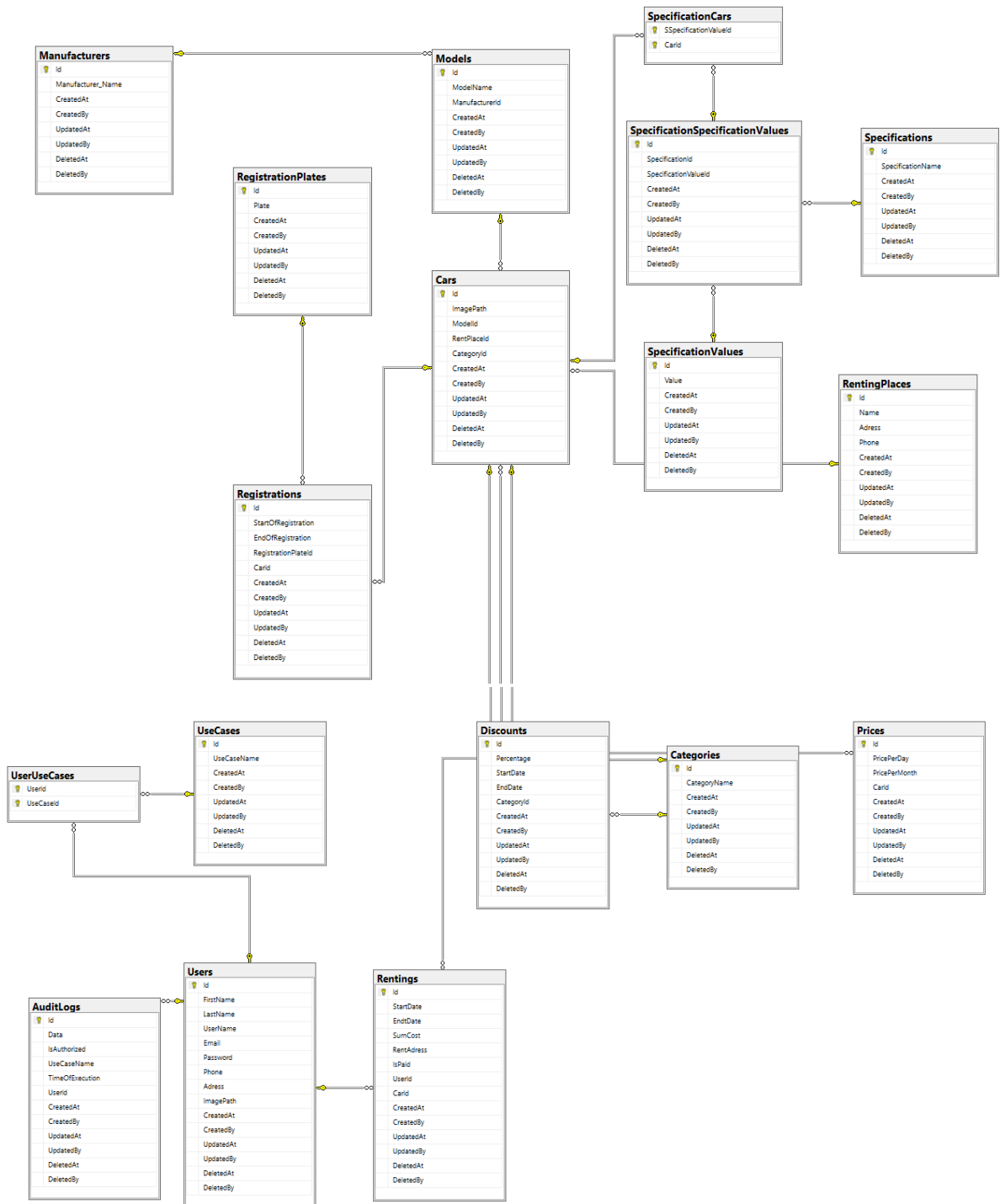
1.3 Opis funkcionalnosti

Funkcionalnosti API-ja su sledeće

- Inicijalno popunjavanje baze podataka fake podacima
- Logovanje i autorizacija korisnika Bearer JWT tokenom
- Registracija korisnika
- Izmena korisnika
- Brisanje korisnika
- Dohvatanje automobil
- Dohvatanje specifinog automobila
- Dodavanje automobila
- Brisanje automobila
- Izmena automobila
- Rentiranje automobila
- Brisanje rente
- Dodavanje cena automobila
- Izmena popusta
- Brisanje specifikacija
- Ispis izvršavanja slučajeva korišćenja

2. Organizacija

2.1 Baza



3. Uputstvo

3.1 Kreiranje baze i promena mesta connection stringu:

Postupak za korišćenja ovog API-ja je sledeći:

Pre prvog pokretanja API-ja neophodno je kreirati bazu podataka pod nazivom „CarRental“.

Inicijalna migracija je napravljena i sleće što radimo je promena mesta izvršavanja konekcionog stringa. U sloju pod nazivom „DataAccess“ potrebno je odkomentarisati vezu sa bazom u metodi „OnConfiguring“, a zakomentaristai pozivanje u konstruktoru i property User (kao u priloženom kodu).

```
/*public CarRentalContext(DbContextOptions options = null) : base(options)
{
}

public IApplicationUser User { get; }*/
protected override void OnConfiguring(DbContextOptionsBuilder
optionsBuilder)
{
    optionsBuilder.UseSqlServer(@"Data Source=.\SQLEXPRESS;Initial
Catalog=CarRental;Integrated Security=True");
}
```

Nakon ovoga potrebno je u API sloju u folderu Extensions u „ContainerExtensions“ klasi zakomentarisati njen sadržaj.

```
public static void AddCarRentalDbContext(this IServiceCollection services)
{
    /*services.AddTransient(x =>
    {
        var optionsBuilder = new DbContextOptionsBuilder();

        var connString = x.GetService<AppSettings>().ConnString;

        optionsBuilder.UseSqlServer(connString).UseLazyLoadingProxies();

        var options = optionsBuilder.Options;

        return new CarRentalContext(options);
    });*/
}
```

Poslednji korak je izvršavanje komande „update-database“ kojom će se kreirati baza podataka.

Kada završimo sa kreiranjem baze potrebno je sve vratiti kao što je bilo prethodno!

3.2 Kreiranje inicijalnih podataka

Nakon kreiranja baze potrebno je da ubacimo inicijalne podatke u bazu. Slanjem zahteva na url adresu: <http://localhost:5000/api/inital> izvršićemo inicijalni unos fejk podataka. Ovim se unose i svi UseCase-vi koji postoje na sistemu i dva korisnika koja su veoma bitni. Kreira se jedan anonimni korisnik, kao i jedan admin korisnik. Anonimni korisnik može samo da generiše token, dok admin korisnik može da uradi sve ostale funkcionalnosti u sistemu.

Kredencijali za Anonimnog korisnika su:

Email: anon@gmail.com

Password: Sifra123!

Kredencijali za Admin korisnika su:

Email: admin@gmail.com

Password: Sifra123!

Veoma je važno da ovo odradimo, inače nećemo moći da koristimo API.

4. Funkcionalnosti

Svaki endpoint na stranici SWAGGER ima opisano koje parametre prima, koje podatke vraća i koje statusne kodove vraća.

4.1 Registracija korisnika

4.1.1 Registracija korisnika

POST <http://localhost:5000/api/register> ovaj endpoint omogućava registraciju korisnika. Kada se korisnik uloguje automatski se omogućava da izvrši sledeće slučajeve korišćenja:

- Pregled vozila
- Pregled specficiranog vozila
- Rentiranje vozila

4.2 Vozila

4.2.1 Dohvatanje vozila

GET <http://localhost:5000/api/car> ovaj endpoint služi za pregled vozila u sistemu. Svaki korisnik može da pregleda vozila i da filtrira vozila po određenom nazivu modela, nazivu proizvođača, registarske oznake, datumu rentiranja.

4.2.2 Dohvatanje specifičnog vozila

GET <http://localhost:5000/api/car/id> ovaj endpoint služi za pregled specifičnog vozila u sistemu. Svaki korisnik može da pregleda određeno vozilo.

4.2.3 Dodavanje vozila

POST <http://localhost:5000/api/car/> ovaj endpoint služi za dodavanje vozila u sistem. Samo admin korisnici mogu dodavati nova vozila.

4.2.4 Izmena vozila

PUT <http://localhost:5000/api/car/id> ovaj endpoint služi za izmenu vozila u sistemu. Samo admin korisnici mogu izmeniti određeni automobil.

4.2.5 Brisanje vozila

DELETE <http://localhost:5000/api/car/id> ovaj endpoint služi za pregled vozila u sistemu. Svaki korisnik može da pregleda vozila i da filtrira vozila po određenom nazivu modela, nazivu proizvođača, registarske oznake, datumu rentiranja.

4.3 Iznajmljivanje

4.3.1 Iznajmljivanje vozila

POST <http://localhost:5000/api/rentings> ovaj endpoint služi za iznajmljivanje automobila. Svaki autorizovani korisnik može da iznajmi automobil u određenom vremenskom periodu ukoliko je vozilo tada slobodno. Nakon uspešnog iznajmljivanja vozila korisniku se pošalje mejl sa podacima o iznajmljivanju.

4.3.2 Otkazivanje rente

DELETE <http://localhost:5000/api/rentings/id> ovaj endpoint služi za otkazivanje rente. Samo admin korisnik može da otkáže određenu rentu.

4.4 Cene

4.4.1 Dodavanje cene

POST <http://localhost:5000/api/prices> ovaj endpoint služi za dodavanje cene automobila. Samo admin korisnik može da doda novu cenu za određeno vozilo

4.5 Popust

4.5.1 Izmena popusta

PUT <http://localhost:5000/api/discount> ovaj endpoint služi za promenu popusta određenoj kategoriji automobila. Samo admin korisnik može da izvrši ovaj slučaj korišćenja.

4.6 Specifikacije

4.6.1 Brisanje određene specifikacije

DELETE <http://localhost:5000/api/specification/id> ovaj endpoint služi za brisanje određene specifikacije. Samo admin korisnik može da izvrši ovaj slučaj korišćenja.

4.7 Korisnik

4.7.1 Dohvatanje korisnika

GET <http://localhost:5000/api/user> ovaj endpoint služi za dohvatanje svih korisnika u sistemu. Samo admin korisnik može da izvrši ovaj slučaj korišćenja.

4.7.2 Izmena korisnika

PUT <http://localhost:5000/api/user> ovaj endpoint služi za izmenu korisnika u sistemu. Samo admin korisnik može da izvrši ovaj slučaj korišćenja.

4.7.3 Brisanje korisnika

DELETE <http://localhost:5000/api/user/id> ovaj endpoint služi za brisanje korisnika u sistemu. Samo admin korisnik može da izvrši ovaj slučaj korišćenja.

4.8 Token

4.8.1 Pravljenje tokena

POST <http://localhost:5000/api/token> ovaj endpoint služi za dobijanje autorizaciju korisnika. Svaki korisnik koji postoji u sistemu može da ovo da izvrši.

4.8.2 Brisanje tokena

DELETE <http://localhost:5000/api/token> ovaj endpoint služi za brisanje tokena korisnika. Svaki korisnik koji ulogovan može da se odloguje.

4.9 Audit Log

4.9.1 Pregled audit log-a

GET <http://localhost:5000/api/auditlog> ovaj endpoint služi za pregled izvršavanja svakog slučaja korišćenja kao i podataka koji je korisnik prosledio. Moguća je pretraga po username-u korisnika ili usecase name-u, kao i pretraga po datumu od i do. Samo admin korisnik može da izvrši ovaj slučaj korišćenja.