

Web programiranje – Specifikacija zadatka za drugi kolokvijum - Grupa 1

Primenjeno Softversko Inženjerstvo 2019/2020

Zadatak za drugi kolokvijum

Napraviti veb aplikaciju za prodaju vozila koristeći **Visual Studio** (2015 ili 2017) okruženje i .NET Framework (<=4.6.1) MVC tehnologiju.

Uputstvo za izradu i predaju zadatka za drugi kolokvijum

1. Zadatak se radi samostalno i nije dozvoljeno kopirati ili koristiti kod od nekog drugog studenta
2. **Rok za završetak zadatka je 31.05.2020. U 23.59h.**
3. Sav kod aplikacije je potrebno da bude na **privatnom GitLab repozitorijumu** koji otvorite i na koji dodate asistentski nalog **web.programiranje.psi** i dodelite tom nalogu ulogu **Maintainer**-a. Ime repozitorijuma mora biti sledećeg formata: **WP-K2-XX-YYY-ZZZZ** pri čemu je XX-YYY-ZZZZ broj indeksa (primer naziva repozitorijuma: WP-K2-PR-047-2016).
4. Minut nakon isteka roka, skripta će automatski pokupiti sve projekte sa svih repozitorijuma studenata koji su izvršili prethodni (3.) korak i biće testirani na plagijate. Studenti čiji projekti se pokazu da su plagijati neće biti ocenjeni (kako onaj student koji je prepisao tako i onaj student od koga je prepisano)
5. Zadatak se radi u **Visual Studio** (2015 ili 2017) okruženju
6. Zadatak se radi u .NET Framework (<=4.6.1) MVC tehnologiji
7. Nije dozvoljeno koristiti bilo kakve dodatne biblioteke prilikom izrade zadatka sem onoga što je ugrađeno u sam Framework
8. Nije potrebno koristiti Javascript (niti bilo koju Javascript biblioteku)
9. Nije dozvoljeno korišćenje HTML helpera
10. Nije dozvoljen *scaffolding*, odnosno automatsko generisanje koda pomoću alata koje pruža Visual Studio
11. Sve podatke potrebne za pravilan rad aplikacije čuvati u tekstualnim datotekama pri čemu student samostalno bira format za čuvanje podataka u datotekama

Specifikacija zahteva

Potrebno je realizovati veb aplikaciju za sistem koji podržava prodaju vozila. Aplikaciju koriste 2 grupe (uloge) korisnika: Kupac (osoba koja želi da kupi Vozilo) i Administrator. Aplikacija rukuje sa sledećim entitetima:

Korisnik

- Korisničko ime (jedinstveno i minimalno 3 karaktera)
- Lozinka (minimalno 8 karaktera koji mogu biti slova i brojevi)
- Ime
- Prezime
- Pol
- Email
- Datum rođenja
- Uloga (Administrator, Kupac)

Vozilo

- Marka (minimalno 3 karaktera)
- Model
- Oznaka šasije
- Boja
- Broj vrata
- Opis
- Vrsta goriva/pogona (Benzin, Dizel, Benzin + Gas, Metan, Električni pogon, Hibridni pogon)
- Cena vozila
- Na stanju (true/false)

Kupovina

- Kupac (Korisnik)
- Vozilo
- Datum kupovine
- Naplaćena cena

Implementirati sledeće funkcionalnosti:

- Registracija - neregistrovan korisnik se registruje na aplikaciju popunjavajući polja koja su za to predviđena i nakon toga postaje Kupac. Ako je registracija neuspešna prikazuje se korisniku poruka greške. Ako je registracija uspešna redirektuje se na stranicu za prijavu.
- Administratori se programski učitavaju iz tekstualne datoteke i ne mogu se naknadno dodati.
- Prijavljivanje na sistem - neprijavljeni korisnik loguje se na sistem tako što unosi korisničko ime i lozinku korisnika za koji je registrovan. Ako je prijava neuspešna prikazuje se korisniku poruka greške. Ako je prijava uspešna redirektuje se na stranicu za pregled Vozila koja se prodaju.

1. Neregistrovani korisnik

- Može da vidi sva Vozila u sistemu koja je moguće kupiti u vidu tabele pri čemu su Vozila prikazana u redovima jedan ispod drugog
- Može da sortira Vozila po marki, po modelu ili ceni (opadajuće i rastuće)
- Može da pretražuje sva Vozila po marki, po modelu ili po ceni od-do (nije potrebno implementirati kombinovanu pretragu)
- Sortiranje i pretraga se implementiraju sa serverske strane

2. Administrator

- Može da vidi sve Kupce u sistemu u vidu tabele pri čemu su oni prikazani u redovima jedan ispod drugog
- Može da obriše Kupca (logički) nakon čega on ne može više da se prijavi
- Može da vidi listu **svih** Vozila (vidi **sva** Vozila koje je su kreirali on i drugi administratori), prodatih i neprodatih, pri čemu se kod prodatih Vozila vidi ko ih je kupio
- Može da dodaje nova Vozila koja se prodaju. Ako je dodavanje neuspešno prikazuje se korisniku poruka greške. Ako je dodavanje uspešno redirektuje se na stranicu koja prikazuje listu svih Vozila.
- Vozilo može da modifikuje pre kupovine (pri čemu je moguće ažurirati/izmeniti vrednost svakog atributa), a nakon kupovine Vozilo se ne može modifikovati (Administrator može da modifikuje vozilo koje je on kreirao ili koje su drugi administratori kreirali)

3. Kupac

- Može da vidi sve svoje kupovine (sa informacijama kupljenih Vozila) u vidu tabele pri čemu su one prikazane u redovima jedna ispod druge
- Može da vidi sva Vozila koja je moguće kupiti u sistemu u vidu tabele pri čemu su ona prikazana u redovima jedno ispod drugog
- Kada izabere jedno Vozilo prelazi na stranicu gde vidi detalje tog Vozila i može da ga kupi (jedno Vozilo ne mogu da kupe 2 kupca). Ako je kupovina neuspešna prikazuje se korisniku poruka greške. Ako je kupovina uspešna redirektuje se na stranicu koja prikazuje sve ranije kupovine.
- Kupac može samo jedno Vozilo da kupi po kupovini

Dodatne napomene:

1. Brisanje korisnika u sistemu je **logičko**.
2. U zavisnosti od konkretne implementacije, **studenti mogu proizvoljno proširiti date entitete ili dodavati druge**.
3. Nakon svakog dodavanja/brisanja entiteta potrebno je ažurirati sadržaj tekstualnih datoteka.
4. Svi datumi u aplikaciji treba da budu u formatu dd/MM/yyyy.
5. Obavezno validirati sve podatke na serverskoj strani prilikom unosa.
6. Nije potrebno raditi validaciju na klijentskoj strani.
7. Izgled aplikacije student može napraviti po želji.
8. Student je u obavezi da implementira navedene funkcionalnosti za maksimalan broj poena (15). Sve dodatno što student implementira se neće bodovati i neće moći da zameni postojeće funkcionalnosti. Bodovaće se samo one funkcionalnosti koje su navedene u specifikaciji.