Software Requirements Specification

(Specifikacija softverskih zahtjeva)

za projekt

Aplikacija za automobile

Verzija 1.0

Autor:

Marijana Mrvelj

Ekonomski fakultet u Osijeku

Kolegij: Razvoj poslovnih aplikacija

6.6.2023.

Sadržaj

1. Uvod 1

1.1 Svrha aplikacije 1

1.2 Korisnici aplikacije 1

1.3 Koristi (benefiti) od aplikacije 1

2. Zahtjevi 1

2.1 Funkcijski zahtjevi 1

2.2 Sistemski, hardverski i mrežni zahtjevi 1

2.3 Sigurnost 1

2.4 Korisnički zahtjevi 1

2.5 Slučajevi (scenariji) korištenja (use-case dijagrami) 2

2.5.1. Slučaj korištenja 1: Pregled bicikala 2

2.5.2. Slučaj korištenja 2: Dodavanje novog bicikla 3

2.5.3. Slučaj korištenja 3: Uređivanje bicikla 3

2.5.4. Slučaj korištenja 4: Brisanje bicikla 3

2.6. Dijagrami klasa 4

3. Dizajn korisničkog sučelja 5

3.5. Glavni principi dizajna korišteni u aplikaciji 5

3.6. Wireframe-ovi i Mockup-i 6

# Uvod

## Svrha aplikacije

<Opisati kratko svrhu aplikacije – za što će ona služiti, koju će funkciju obavljati >

Kroz vježbu kolegija Razvoj poslovnih aplikacija radit će se jednostavna aplikacija za automobile koja podržava unos, uređivanje, brisanje, pretragu i opis automobila u bazi. Aplikacija također, treba omogućiti brisanje i prikaz detalja pojedinog automobila. Svaki unos podataka kroz aplikaciju treba uključivati provjeru valjanosti i za brisanje podataka je potrebna posebna potvrda korisnika.

## Korisnici aplikacije

<Tko će biti potencijalni korisnici aplikacije>

Aplikaciji će moći pristupiti svi korisnici koji imaju internet i internet preglednik te žele pristupiti svim

bitnim informacijama vezano za automobile na jednom centraliziranom mjestu.

## Koristi (benefiti) od aplikacije

<Koje koristi u smislu promjena u odnosu na postojeće stanje u tvrtci ili instituciji će ova aplikacija donijeti – što će automatizirati, što će se raditi brže, hoće li donijeti neke nove uvide (izvješća) u podatke, hoće li pridonijeti većoj dostupnosti informacija i kako to može povećati uspješnost poslovanja (očekuje li se smanjenje nekih troškova i kojih, ili povećanje prihoda i kojih)>

Stvorit će se jedna centralna baza podataka o automobilima dostupna svima. Korisnici kada budu htjeli potražiti informacije o biciklima neće više morati pretraživati putem Google-a i koristiti više izbora za dobivanje informacija o nekom automobila jer će kroz aplikaciju moći dobiti sve potrebne informacije na jednom mjestu. Aplikacija će biti dostupna putem interneta, zahvaljujući tome korisnik ima mogućnost korištenja aplikacije u bilo koje vrijeme.

# Zahtjevi

## Funkcijski zahtjevi

<Koje sve glavne funkcije, tj. operacije treba moći obavljati aplikacija, tj. što se sve treba moći napraviti pomoću aplikacije – ovdje navesti ukratko, a detaljni opis će ići pod korisničke zahtjeve>

Aplikacija mora omogućiti spremanje, uređivanje, pretraživanje, prikaz, traženje i brisanje automobila u bazi podataka.

## Sistemski, hardverski i mrežni zahtjevi

<U kojoj okolini aplikacija može raditi – na web poslužitelju tvrtke ili u oblaku (kojem?) ili lokalno ili na mobilnom uređaju? Na kojoj platformi – koji operativni sustav, koja baza podataka? Postoje li neki posebni zahtjevi za hardver (jačina procesora, memorija,disk…) da bi aplikacija mogla raditi bez problema?

Budući da će aplikacija biti razvijena u ASP.NET Core MVC-u ona treba biti smještena na Microsoft

Web poslužitelju (eng. server). Preporučuju se sljedeće hardverske specifikacije:

Minimum četverojezgreni procesor 2.2 GHz

Minimum 30GB RAM memorije

Minimum 1TB prostora

Operativni sustav Windows server 2019.

### 2.2.1 Web server

Preporučuje se korištenje Windows Azure-a za hostanje aplikacije.

Windows Azure može hostatii bilo koju ASP.NET Core MVC aplikaciju, uključujući i našu

predloženu aplikaciju u ovom dokumentu.Instaliranje je vrlo jednostavno jer je Microsoft odgovoran

za dodavanje resursa na poslužitelju za vrijeme visokog prometa.

Troškovi su minimalni, oni ovise o količini podataka koji se prikazuju posjetiteljima te održavanje

hardvera nije uključeno u njih.

### 2.2.2 Baze podataka

Preporučuje se korištenje SQL SERVER baze podataka unutar Windows Azure-a za temeljnu

bazu podataka aplikacije. Što se tiče Web poslužitelja, ova preporuka osigurava visoku dostupnost

za bazu podataka s dobrim omjerom vrijednosti za uložen novac. To posebno ima smisla ako je I

Web aplikacija hostana na Windows Azure-u.

## Sigurnost

<Kako će se u sustavu zaštititi osobni podaci korisnika (lozinke ili drugo) ako se koriste?>

U kasnijem razvoju aplikacije razvit će se sigurna identifikacija i zaštićena autentikacija gdje

korisnička imena i lozinke ne smiju biti spremljena u obična tekstualna polja i datoteke, a ostali

podaci korisnika kao što su adresa, telefonski brojevi, brojevi kreditnih kartica neće biti dostupni

anonimnim pristupom.

## Korisnički zahtjevi

<Ovdje navesti detaljni popis korisničkih zahtjeva – preporučeno u tablici, i to po vrstama korisnika - koje će vrste korisnika biti omogućene, što od operacija treba moći svaka vrsta korisnika napraviti, tj. tko će moći unositi, tko ispravljati, tko brisati, gdje će se spremati podaci, koja su izvješća potrebna, itd.>

Tablica. Korisnički zahtjevi

| Rb. | Zahtjev | Vrsta korisnika (user / admin) |
| --- | --- | --- |
| 1. | Prikaz svih automobila | Anonimni korisnik |
| 2. | Pretraga automobila po kategoriji I nazivu | Anonimni korisnik |
| 3. | Unos automobila | Registrirani korisnik |
| 4. | Uređivanje automobila | Registrirani korisnik |
| 5. | Brisanje automobila | Administrator |
| 6. | Provjera valjanosti podataka kod unosa I uređivanja | Registrirani korisnik |
| 7. | Potvrda s pitanjem “Jeste li sigurni?” kod brisanja automobila | Administrator |
| 8. | Prikaz detalja pojedinog automobila | Anonimni korisnik |
| 9. | Početna stranica dolaska na aplikaciji mora sadržavati osnovne informacije o svrsi aplikacije | Anonimni korisnik |

<dodati retke po potrebi>

## Slučajevi (scenariji) korištenja (use-case dijagrami)

<Ovdje umetnuti use-case dijagrame napravljene u UML jeziku u programu Astah Community. Dijagrame raditi prema Tablici korisničkih zahtjeva, ali povezivati aktivnosti>

Sljedeći slučajevi korištenja opisuju scenarije u kojima korisnici web aplikacije koriste predloženu aplikaciju za upravljanje automobilima. U tim slučajevima korištenja su uključene osnovne operacije, stoga ih ne treba smatrati konačnim. Kako napreduje razvoj dodatna funkcionalnost može biti dodana prema odluci SCRUM mastera.

### 2.5.1. Slučaj korištenja 1: Pregled automobila

Kada posjetitelj stranice pregledava Filmove koji se nalaze u web aplikaciji, odvijaju se sljedeći koraci:

1. Posjetitelj dolazi na početnu stranicu web mjesta kao anonimni korisnik ili klikne na link **Početna stranica** u izborniku ako se nalazio na drugoj stranici na istom web mjestu.
2. Početna stranica prikazuje osnovni opis web aplikacije i sadrži gumbe za prikaz, pretraživanje i dodavanje novih automobila.
3. Prikaz osnovnih informacija o razvojnom timu moguće je dobiti putem stranica **O nama** i **Kontakt**.
4. Ako anonimni korisnik želi vidjeti sve Automobile u bazi, mora kliknuti na link **Popis automobila** u glavnom izborniku ili gumb prikaži na **Početnoj stranici**.
5. Web aplikacija prikazuje popis automobila. Za svaki Automobil se prikazuje Naziv automobila, Godina proizvodnje automobila, Kategorija te Cijena.
6. Ako anonimni korisnik želi pretraživati Automobile u bazi po Kategoriji i Nazivu, mora kliknuti na link **Tražilica automobila** u glavnom izborniku.
7. Ako anonimni korisnik želi vidjeti detalje Automobila, mora kliknuti na link **Detalji** za taj Automobil.
8. Web aplikacija prikazuje detalje odabranog automobila –Naziv automobila, Datum Godina proizvodnje Automobila, Kategorija te Cijenu.

### Slučaj korištenja 2: Dodavanje novog automobila

Svi korisnici trebaju moći dodati novi Automobil. Kada korisnik dodaje novi Automobil, sljedeći koraci se odvijaju:

1. Korisnik klikne na gumb **Unos** na **Početnoj stranici** ili na link **Novi automobil** na stranicama **Popis automobila** ili **Tražilica automobila**.
2. Korisnik upisuje podatke o novom Automobilu.
3. Korisnik klikne na gumb **Spremi**.
4. Ako su upisani podaci ispravni, web aplikacija sprema Automobil u bazu i vraća korisnika na stranicu **Popis automobila**

### Slučaj korištenja 3: Uređivanje automobila

Kada korisnik uređuje Automobil, sljedeći koraci se odvijaju:

1. Korisnik klikne na link **Uredi** u popisu automobila na stranicama **Popis automobila** ili **Tražilica automobila**.
2. Korisnik mijenja postojeće podatke o automobilu.
3. Korisnik klikne gumb **Spremi promjene**.
4. Ako su upisani podaci točni, web aplikacija sprema promjene u bazi i prikazuje stranicu za **Popis automobila**.

### **Slučaj korištenja 4: Brisanje automobila**

Kad korisnik briše automobile iz baze podataka web aplikacije, sljedeći koraci se odvijaju:

1. Korisnik klikne na link **Obriši** u popisu automobila na stranicama **Popis automobila** ili **Tražilica automobila.**
2. Web aplikacija zahtijeva potvrdu o brisanju automobila.
3. Ako korisnik potvrđuje brisanje, automobil je uklonjen iz baze.
4. Web aplikacija prikazuje stranicu **Popis automobila**.

Slika na kojoj se prikazuje tekst, dijagram, krug, crta

Opis je automatski generiran

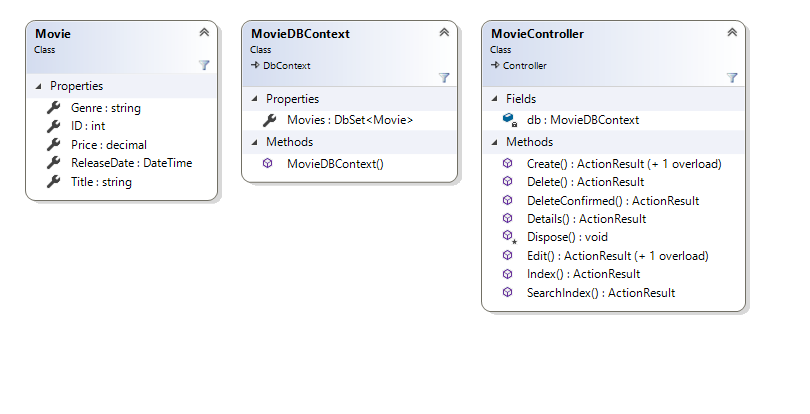
Slika 1: Dijagram slučajeva korištenja

## Dijagrami klasa

<Ovdje umetnuti dijagrame klasa napravljene u UML jeziku u programu Astah Community. Dijagrami trebaju biti na fizičkoj razini (sadržavati nazive klasa, nazive svojstava, tipove svojstava, nazive i tipove metoda, kao i veze između klasa>

Klasa **Automobil** je potrebna kako bi se u aplikaciji evidentirali matični podaci za svaki automobil. Svojstva koja opisuju neki automobil su: ID (identifikator automobile ), Naziv (naslov automobila, tekstualni podatak), Godina proizvodnje (godina, tekstualni podatak), Cijena (cijena, decimalni broj), Kategorija (kateogrija, tekstualni podatak)

Kako bi se podaci o automobililma mogli spremiti u bazu podataka, potrebno je napraviti klasu **AutomobilDBContext** koja koristi klasu Automobil kao model za izradu tablice u bazi pomoću Entity frameworka pa zbog toga i nasljeđuje klasu **DbContext**. Nakon toga treba pristupiti razvoju kontrolera AutomobilController koji mora naslijediti baznu klasu **Kontroler** s pripadajućim metodama za manipulaciju nad bazom.



Slika na kojoj se prikazuje tekst, snimka zaslona, Font, broj

Opis je automatski generiran

Slika 2: Dijagram klasa

# Dizajn korisničkog sučelja

## Glavni principi dizajna korišteni u aplikaciji

<Ovdje staviti opis glavnih principa dobrog dizajna koji su korišteni u aplikaciji uz referencu na izvor. Objasniti razloge za položaj, veličinu i raspored nekih dijelova u korisničkom sučelju, npr. zašto je slika zaglavlja na određenom mjestu, objasniti navigaciju, položaj izbornika, dugmadi i najvažnijih informacija u aplikaciji, boje, lakoću korištenja (usability) aplikacije i responzivnost dizajna. >

## Wireframe-ovi i Mockup-i

<Ovdje staviti wireframe-ove ili mockup-e za svaki ekran aplikacije. Ispod slika staviti kratke opise slike - koji slučaj korištenja prikazuju i ako treba objasniti kratko dizajn. Na kraju napisati je li front-end developer usvojio sve preporuke dizajnera u samoj aplikaciji

Slika na kojoj se prikazuje tekst, automobil, snimka zaslona, vozilo

Opis je automatski generiran

Slika 3. Mockup početne stranice

Slika 3 prikazuje elemente početne stranice, odnosno prve stranice koja se učitava u internet pregledniku kada korisnik upiše link web aplikacije. Početna stranica povezana je s prvim i drugim korakom prvog slučaja korištenja. Vidljivo je da ona sadrži na vrhu horizontalnu traku izbornika, zatim ispod toga sliku, a ispod slike u tri stupca glavne operacije koje se mogu izvršiti u aplikaciji, i to putem dugmadi koje vode na posebne poglede za svaku operaciju: Prikaži, Pretraživanje i Unos.

Slika na kojoj se prikazuje tekst, snimka zaslona, zaslon, softver

Opis je automatski generiran

Slika 4. Mockup stranice s popisom automobila

Slika 4 prikazuje elemente stranice “Popis automobila”. Stranica “Popis automobila povezana je s prvim slučajem korištenja I služi za prikaz svih filmova u bazi.

Popis automobila prikazivat će se u obliku jednostavne tablice sa svim poljima o svakom automobilu, s tim da se iza svakog retka nalaze linkovi na posebne stranice za operacije “Uredi”, “Detalji” i “Obriši” koje se odnose na automobil u tom retku.

Slika na kojoj se prikazuje tekst, snimka zaslona, zaslon, softver

Opis je automatski generiran

Slika 5. Mockup stranice za pretraživanje automobila

Slika 5 prikazuje elemente stranice “Tražilica automobila”. Stranica “Tražilica automobila” povezana je s prvim slučajem korištenja i služi za pretraživanje automobila po Kategoriji i Nazivu u bazi. Filter za pretraživanje po kategoriji je napravljen kao combo box s ponuđenim kategorijama iz baze, a filter po nazivu automobila je prikazan kao textbox u koji korisnik upisuje dio naziva automobila ili cijeli naziv automobila. Pretraživanje se aktivira nakon što korisnik klikne na dugme “Traži”, i tada se na istoj stranici ispod prikazuju samo oni automobili koji udovoljavaju traženim kriterijima. Filtrirani automobili se prikazuju u obliku iste tablice kao i kod popisa svih automobila.

Slika na kojoj se prikazuje tekst, snimka zaslona, zaslon, softver

Opis je automatski generiran

Slika 6. Mockup stranice “O nama”

Slika 6 prikazuje elemente stranice koja sadrži informacije o razvojnom timu. Opis je pružen u prvom slučaju korištenja.

Slika na kojoj se prikazuje tekst, snimka zaslona, zaslon, softver

Opis je automatski generiran

Slika 7. Mockup stranice “Kontakt”

Prethodna slika prikazuje elemente stranice koja sadrži informacije o kontakt informacijama razvojnog tima. Opis je pružen u prvom slučaju korištenja.

Slika na kojoj se prikazuje tekst, snimka zaslona, zaslon, softver

Opis je automatski generiran

Slika 8. Mockup stranice za dodavanje novog automobila

Slika 8 prikazuje elemente stranice koja se koristi za unos novog Automobila u bazi. Sva polja, prikazana kroz elemente, je potrebno ispuniti kako bi Automobil bio uspješno spremljen u bazu. Detaljan opis pružen je u drugom slučaju korištenja. Osim unosa i spremanja automobila korisnik ima prikazan i link na povratak na popis svih automobila.

Slika na kojoj se prikazuje tekst, snimka zaslona, zaslon, softver

Opis je automatski generiran

Slika 9. Mockup stranice za izmjene podataka automobila

Slika 9 prikazuje elemente stranice koja se koristi za izmjene postojećeg automobila u bazi. Detaljan opis pružen je u trećem slučaju korištenja. Obrazac ima isti oblik kao i za unos novog automobila, kako bi korisniku omogućio lakše snalaženje u aplikaciji, kao i dugme za spremanje promjena i ispod toga link na povratak na popis svih automobila.

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, zaslon

Opis je automatski generiran

Slika 10. Mockup stranice za brisanje automobila

Slika 10 prikazuje elemente stranice koja se koristi za korisnikovu potvrdu brisanja postojećeg automobila u bazi. Detaljan opis pružen je u četvrtom slučaju korištenja. Korisnik ima mogućnost brisanja ili povratka na popis svih automobila.

Slika na kojoj se prikazuje tekst, snimka zaslona, softver, zaslon

Opis je automatski generiran

Slika 11. Mockup stranice za prikaz detalja o automobilu

Slika 11 prikazuje elemente stranice koja se koristi za prikaz detalja pojedinačnog automobila u bazi. Opis je pružen u zadnjem koraku prvog slučaja korištenja. Korisnik ima mogućnost klika na link za uređivanje automobila ili povratka na popis automobila.