

Web programiranje – Specifikacija projektnog zadatka

Primenjeno Softversko Inženjerstvo 2020/2021

Projektni zadatak

Upotrebom tehnologija i tehnika primenjivanih na predavanjima i vežbama projektovati veb aplikaciju za turističku agenciju prema sledećoj specifikaciji.

Informacija o roku za izradu projekta će naknadno biti postavljena na zvaničnom sajtu predmeta (<https://canvas.ftn.uns.ac.rs/courses/2541>).

Pravila izrade projektnog zadatka

- Svaki student radi **samostalno** projekat bez pomoći drugih; nije dozvoljeno plagiranje koda drugog studenta
- Prilikom izrade projektnog zadatka dozvoljeno je koristiti **isključivo** tehnologije (i verzije istih) koje se navedene u dokumentu “Web programiranje - tehnologije.pdf”
- Sve podatke potrebne za pravilan rad aplikacije neophodno je **trajno** čuvati **isključivo u tekstualnim datotekama** (student bira jedan od sledećih formata: json, xml, csv, tsv ili neki proizvoljni dsv (delimiter-separated value, odnosno delimiter kojim će odvajati vrednosti unutar datoteke)
- Projektni zadatak se može braniti u junu ili septembru (ako studentska služba bude dozvolila biće održan još jedan dodatni rok kasnije)
- Aplikacija mora da ima **unapred ubačene test podatke** za demonstraciju svih implementiranih funkcionalnosti
- Dizajn korisničkog interfejsa kao i stilovi (CSS) su prepušteni studentu i obavezni su
- Web server na kome aplikacija mora biti podignuta je IIS Express
- Obavezna upotreba sistema kontrole verzije u vidu **GitLab-a**, a uputstvo za upotrebu možete pronaći na Canvas portalu u dokumentu “Uputstva za Git.pdf”
- 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: **XXYYY-ZZZZ-Web-projekat** gde XX šifra vašeg smera (npr. PW, PR, RA, E3, itd.), YY predstavlja broj indeksa i ZZZZ godinu upisa (primer naziva repozitorijuma: **PR047-2016-Web-projekat**).
- Sva pitanja vezano za razumevanje projektnog zadatka potrebno je postaviti u otvorenoj diskusiji u **Discussions** sekciji na **Canvas** platformi, a odgovori na postavljena pitanja će biti navedeni na kraju javno dostupnog **dokumenta “Česta pitanja”**

(https://docs.google.com/document/d/1ZseblzGahpDGkB06-Q7ZZMEyryZ_-z31KQ_7hNXZjw4/edit?usp=sharing). Na pitanja koja se odnose na detalje implementacije neće biti odgovoreno pošto će upravo ti segmenti biti ocenjeni i njih studenti moraju samostalno da reše.

- Sve informacije o datumima odbrane, rokovima za završetak izrade projekta i dodatnim informacijama o polaganju projektnog zadatka biće blagovremeno objavljeni u vidu obaveštenja u **Announcements** sekciji na **Canvas** portalu

Ocenjivanje projektnog zadatka

- Maksimalan broj bodova koji se može osvojiti je **40**
- Da bi projektni zadatak bio položen potrebno je osvojiti **minimalno 20 bodova** na odbrani
- Svaka pravilno implementirana funkcionalnost donosi određen broj bodova (broj bodova zavisi od kompleksnosti i težine implementacije)
- Pre odbrane projektnog zadatka, isti dan (rano ujutro), se održava izrada eliminacionog zadatka. Studenti koji uspešno urade eliminacioni zadatak mogu da brane projekat u tom roku. Studenti koji ne urade uspešno eliminacioni zadatak ne mogu da brane projektni zadatak u tom roku ni za jednu ocenu. Eliminacioni zadatak se ocenjuje po principu 1 ili 0 - ako je sve u zadatku urađeno student dobija ocenu 1 i može da brani projektni zadatak, a u suprotnom dobija 0 i ne može da brani projekat u tom roku.
- Izlazak na eliminacioni zadatak nije obavezan i u tom slučaju studenti mogu da osvoje mnogo manje bodova usled čega ne mogu na kraju semestra ostvariti veću ocenu

Ocenjivanje projektnog zadatka ako student položi eliminacioni zadatak

- Implementacija rešenja uz korišćenje jQuery biblioteke, REST-a (ASP.NET Web API 2) i AJAX poziva, pri čemu se između klijenta i servera razmenjuju JSON objekti, nosi više bodova nego implementacija rešenja korišćenjem MVC 5 tehnologije
- Konačan osvojeni broj bodova na odbrani projektnog zadatka za studente koji uspešno urade eliminacioni zadatak se računa na sledeći način:

$$\text{konačan_broj_bodova} = \text{osvojeni_bodovi} * \text{koef_za_tehnologiju} * \text{koef_za_rok}$$

- *konačan_broj_bodova* - konačan broj bodova koji student ostvari na osnovu formule
- *osvojeni_bodovi* - ukupan broj bodova koji student ostvari i koji se dobija zbirom bodova ostvarenih za pravilnu implementaciju funkcionalnosti
- *koef_za_tehnologiju* - koeficijent za tehnologiju
 - 1.0 ako je student implementirao rešenje uz korišćenje jQuery biblioteke, REST-a (ASP.NET Web API 2) i AJAX poziva, pri čemu se između klijenta i servera razmenjuju JSON objekti
 - 0.8 ako je studenti implementirao rešenje korišćenjem MVC 5 tehnologije
- *koef_za_rok* - koeficijent za odbranu u određenom roku
 - 1.0 ako je student odbranio projektni zadatak u prvom (junskom roku)

- 0.9 ako je student odbranio projektni zadatak u drugom (septembarskom) roku
- u slučaju da bude održan treći rok za odbranu projekta, koeficijent u tom roku će iznositi 0.8
- Primeri:
 - **Primer 1:** Student položi eliminacioni zadatak i odbrani projekat u prvom roku koristeći MVC 5 tehnologiju, a pritom implementira svaku funkcionalnost što iznosi maksimalnih 40 bodova -> konačan_broj_bodova = $40 * 0.8 * 1.0 = 32$ -> **student je položio projektni zadatak**
 - **Primer 2:** Student položi eliminacioni zadatak i odbrani projekat u prvom roku koristeći Web API 2 tehnologiju, a pritom ne implementira svaku funkcionalnost te ostvari 25 bodova -> konačan_broj_bodova = $25 * 1.0 * 1.0 = 25$ -> **student je položio projektni zadatak**
 - **Primer 3:** Student položi eliminacioni zadatak i odbrani projekat u drugom roku koristeći MVC 5 tehnologiju, a pritom ne implementira svaku funkcionalnost te ostvari 27 bodova -> konačan_broj_bodova = $27 * 0.8 * 0.9 = 19.44$ -> **student nije položio projektni zadatak jer nije ostvaren minimum od 20 bodova**

Ocenjivanje projektnog zadatka ako student ne izađe na eliminacioni zadatak

- Student koji ne izađe na eliminacioni zadatak može da koristi bilo koju tehnologiju iz spiska dozvoljenih tehnologija (nema razlike u osvojenom broju bodova ako se radi MVC 5 ili Web API 2)
- Potrebno je osvojiti **minimalno 20 bodova**
- U zavisnosti od roka u kojem student polaže projekat, u slučaju da student osvoji veći broj bodova, konačan broj bodova će biti skaliran na sledeću vrednost:
 - **25** (maksimum za ovaj rok) ako student odbrani projektni zadatak u prvom roku i osvoji više od toliko bodova
 - **23** (maksimum za ovaj rok) ako student odbrani projektni zadatak u drugom roku i osvoji više od toliko bodova
 - u slučaju da bude održan treći rok za odbranu projekta, ako student odbrani projektni zadatak u trećem roku biće mu skalirani bodova na **21** (maksimum za ovaj rok) ako osvoji više od toliko bodova
- Ako student ostvari više bodova od definisanog maksimuma za određeni rok, bodovi će biti skalirani na vrednost definisanu za rok u kojem je projektni zadatak položen
- Primeri:
 - **Primer 1:** Student ne izađe na eliminacioni zadatak i odbrani projekat u prvom roku koristeći MVC 5 tehnologiju, a pritom implementira svaku funkcionalnost što iznosi maksimalnih 40 bodova -> skalirani broj bodova je **25** -> **student je položio projektni zadatak**
 - **Primer 2:** Student ne izađe na eliminacioni zadatak i odbrani projekat u prvom roku koristeći Web API 2 tehnologiju, a pritom ne implementira svaku funkcionalnost te ostvari 35 bodova -> skalirani broj bodova je **25** -> **student je položio projektni zadatak**

- **Primer 3:** Student ne izađe na eliminacioni zadatak i odbrani projekat u drugom roku koristeći MVC 5 tehnologiju, a pritom ne implementira svaku funkcionalnost te ostvari 19 bodova -> **student nije položio projektni zadatak jer nije ostvaren minimum od 20 bodova**
- **Primer 4:** Student ne izađe na eliminacioni zadatak i odbrani projekat u drugom roku koristeći MVC 5 tehnologiju, a pritom ne implementira svaku funkcionalnost te ostvari 28 bodova -> skalirani broj bodova je **23 -> student je položio projektni zadatak**

Specifikacija zahteva

Potrebno je realizovati veb aplikaciju za informacioni sistem turističke agencije koji omogućava rezervaciju putovanja i informisanje o turističkim destinacijama. Aplikaciju koriste 3 grupe (uloge) korisnika: Turista, Menadžer i Administrator. Aplikacija rukuje sa sledećim entitetima:

Korisnik

- Korisničko ime (jedinstveno)
- Lozinka
- Ime
- Prezime
- Pol
- Email
- Datum rođenja (čuvati u formatu dd/MM/yyyy)
- Uloga (Administrator, Menadžer, Turista)
- Lista svih rezervacija aranžmana bez obzira na njihov status (ako je korisnik Turista)
- Lista svih kreiranih aranžmana (ako je korisnik Menadžer)

Aranžman

- Naziv
- Tip aranžmana (noćenje sa doručnom, polupansion, pun pansion, all inclusive, najam apartmana)
- Tip prevoza (autobus, avion, autobus+avion, individualan, ostalo)
- Lokacija gde se putuje (grad, država ili regija)
- Datum početka putovanja (čuvati u formatu dd/MM/yyyy)
- Datum završetka putovanja (čuvati u formatu dd/MM/yyyy)
- Mesto nalaženja (gde se grupa nalazi, tj. odakle kreće putovanje)
- Vreme nalaženja (kada se grupa nalazi, tj. kada počinje putovanje; čuvati u formatu HH:mm)
- Maksimalan broj putnika
- Opis aranžmana
- Program putovanja (opis)
- Poster aranžmana (slika)
- Smeštaj

Mesto nalaženja (primer na slici 1)

- Adresa u formatu: ulica i broj, mesto/grad, poštanski broj
- Geografska dužina
- Geografska širina



Slika 1.

Smeštaj

- Tip smeštaja (hotel, motel, vila)
- Naziv
- Broj zvezdica (za hotele)
- Postojanje bazena (da/ne)
- Postojanje spa centra (da/ne)
- Prilagođeno za osobe sa invaliditetom (da/ne)
- Postoji wifi (da/ne)
- Lista svih smeštajnih jedinica

Smeštajna jedinica

- Dozvoljen broj gostiju
- Dozvoljen boravak kućnim ljubimcima (da/ne)
- Cena za celu smeštajnu jedinicu

Rezervacija

- Jedinstveni identifikator rezervacije (15 karaktera)
- Turista koji vrši rezervaciju
- Status (aktivna, otkazana)
- Izabrani aranžman za koji se vrši rezervacija
- Izabrana smeštajna jedinica

Komentar

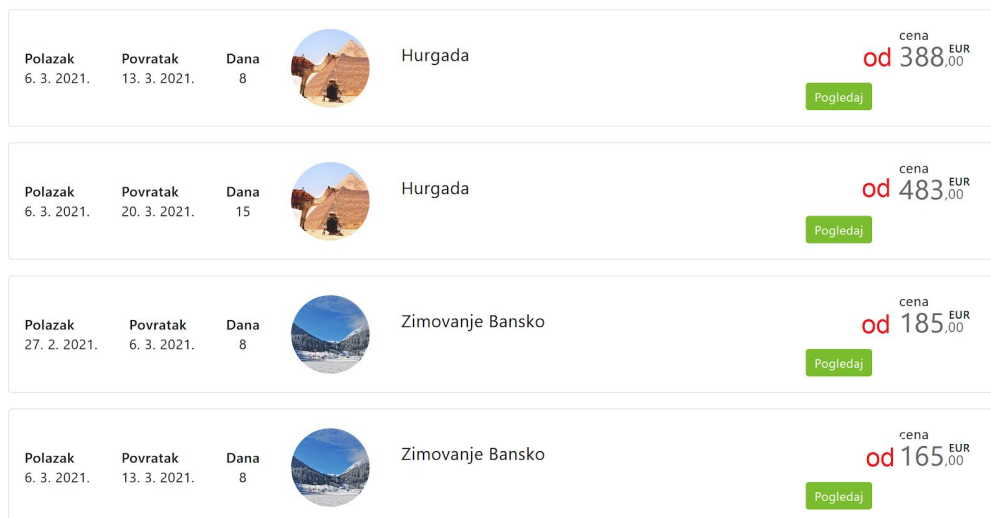
- Turista koji je učestvovao na putovanju i koji je ostavio komentar
- Aranžman na koji se komentar odnosi
- Tekst komentara
- Ocena (na skali od 1 do 5)

Napomena: Entiteti koji su navedeni su obavezni i obavezno je da sadrže navedene atribute. Dozvoljeno je dodati još entiteta kao i dopuniti postojeće entitete sa još atributa ako za to imate potrebe.

Funkcionalnosti za implementaciju

Neprijavljeni korisnik

- Na početnoj strani može da gleda sve predstojeće aranžmane (sortiranje od najskorijih prema najdaljim) koje pruža turistička agencija (primer na slici 2)



Slika 2

- Može da pređe na stranu gde se vide svi aranžmani koji su prošli ili su u toku (sortirani od najskorije završenih ka najdalje završenim)
- Može da gleda detaljne informacije svakog aranžmana (pri čemu se vide **svi** detalji aranžmana uključujući mesto nalaženja koje osim tekstualnog opisa sadrži i prikaz lokacije **na mapi**)
- Može da pregleda informacije o smeštaju za aranžman (pri čemu se vide svi detalji smeštaja) i tom prilikom vidi dostupne i zauzete smeštajne jedinice
- Može da pretražuje aranžmane na stranicama za pregled (budućih i prošlih) i sortira ih
- Može da gleda komentare koje su ostavili turisti na aranžmanima koji su prošli
- Može da sortira i pretražuje smeštajne jedinice u okviru stranice za pregled smeštaja
- Registracija - registruje se na aplikaciju popunjavajući polja koja su za to predviđena i nakon toga postaje Turista
- Prijavljivanje na sistem - loguje se na sistem tako što unosi korisničko ime i lozinku, a ako je prijava uspešna može da izvršava aktivnosti predviđene njegovom ulogom.

Prijavljeni korisnik

- Nakon što se korisnik prijavi, na početnoj strani vidi isti sadržaj (buduće aranžmane) kao i neprijavljeni korisnik i može da ih pretraži i sortira
- U skladu sa ulogom prijavljenog korisnika, sa ove stranice korisnik može otići na druge stranice koje odgovaraju ulozi tog korisnika
- Svaki prijavljeni korisnik može da pregleda svoj profil i uređuje ga
- Prijavljeni korisnik može da radi sve isto što i neprijavljeni**

Spisak funkcionalnosti u zavisnosti od uloge:

Turista

- Može da izvrši rezervaciju nekog aranžmana (pronalazi slobodnu smeštajnu jedinicu u okviru smeštaja za određeni aranžman i rezerviše je)
- Može da otkaže aranžman nakon čega se izabrana smeštajna jedinica u okviru aranžmana oslobađa (ako je aranžman prošao ne može da se otkaže rezervacija)
- Može da pregleda sve svoje rezervacije (buduće i prethodne; aktivne i otkazane)
- Može da sortira i pretražuje svoje rezervacije u sklopu stranice za pregled rezervacija
- Kada aranžman prođe i turista se vrati sa putovanja, može da ostavi komentar na taj aranžman
 - Nakon kreiranja taj komentar nije vidljiv nikome sem menadžeru koji je kreirao taj aranžman i koji može da ga odobri nakon čega taj komentar postaje vidljiv svima

Menadžer

- Kreira, modifikuje, pregleda i briše svoje aranžmane
 - U okviru kreiranja aranžmana definiše se mesto nalaženja tako što se izabere tačna lokacija na mapi; prilikom odabira lokacije (i kasnijeg njenog prikaza) koristiti **OpenLayers mape** (<https://openlayers.org/>) ili neku alternativu sa sličnim funkcionalnostima
 - Svaki menadžer je interno zadužen za određene aranžmane i isključivo ta osoba ih uređuje (menadžer može da vidi aranžmane koje su kreirali drugi menadžeri prilikom pregleda svih aranžmana što je dostupno svim korisnicima, ali može da uređuje samo one koje je taj menadžer kreirao)
 - Brisanje aranžmana je **logičko**
 - Brisanje aranžmana nije dozvoljeno ako postoji neka rezervacija za taj aranžman
- Kreira, modifikuje, pregleda i briše smeštaje
 - Brisanje smeštaja je **logičko**
 - Brisanje smeštaja nije dozvoljeno ako postoji aranžman u budućnosti sa izabranim smeštajem
- Kreira, modifikuje, pregleda i briše smeštajne jedinice u okviru smeštaja
 - Brisanje smeštajnih jedinica je **logičko**
 - Brisanje smeštajne jedinice nije dozvoljeno ako postoji neka rezervacija te jedinice u nekom aranžmanu koji predstoji u budućnosti
 - Modifikacija broja kreveta nije dozvoljena ako postoji neka rezervacija te jedinice u nekom aranžmanu koji predstoji u budućnosti
- Prilikom pregleda aranžmana, smeštaja i smeštajnih jedinica u okviru nekog smeštaja može da podatke sortira i pretraži
- Pregleda sve rezervacije u okviru svojih aranžmana i klikom na rezervaciju može da vidi njene detalje
- Može da vidi sve komentare u okviru svojih aranžmana
- Kada turista kreira komentar, menadžer može da ga odobri ili odbije
 - Odobren komentar biva automatski vidljiv svima

- Odbijen komentar je vidljiv samo tom menadžeru
- Mogu se odobriti/odbijati isključivo komentari na aranžman koji je taj menadžer kreirao

Administrator



- **Učitavaju se programski** iz tekstualnog fajla i **ne mogu se naknadno dodati**
- Regstruje nove menadžere
- Pregleda sve korisnike u sistemu pri čemu može da ih sortira i pretraži
- Prikaz sumnjivih turista - prikaz turista koji su 2 ili više puta otkazali rezervacije
- Blokiranje sumnjivih turista - blokiranjem sumnjivog turista taj blokirani korisnik više ne može da se prijavi na sistem

Funkcionalnosti pretrage i sortiranja

- Svaka pretraga treba biti **kombinovanog karaktera** - korisnik unosi vrednosti za više parametara pretrage i dobija rezultate koji ispunjavaju svaki kriterijum pretrage (primer kombinovane pretrage prikazuje slika 3)

Pretraga aranžmana

Datum početka putovanja:

Min:	Max:
22/06/2021 	do 

Datum završetka putovanja:

Min:	Max:
Od 	18/07/2021 

Tip prevoza

Tip aranžmana

Naziv

Traži

Slika 3

- Prilikom pretrage, ako korisnik unosi sadržaj u polje **tekstualnog** tipa, proveriti da li vrednosti atributa objekta **sadrži** unetu vrednosti (npr. ako se pretražuju aranžmani i u polje za unos naziva aranžmana unese "di", potrebno je pronaći sve aranžmane čiji naziv sadrži ovaj tekstualni sadržaj ("Divčibare zimovanje", "Leto na Maldivima", "Čari Vladivostoka" itd.)
- Aranžmani
 - Pretraga
 - Donja granica datuma za početak putovanja
 - Gornja granica datuma za početak putovanja
 - Donja granica datuma za završetak putovanja

- Gornja granica datuma za završetak putovanja
 - Tip prevoza
 - Tip aranžmana
 - Naziv
- Sortiranje
 - Naziv (rastuće i opadajuće)
 - Datum početka putovanja (rastuće i opadajuće)
 - Datum završetka putovanja (rastuće i opadajuće)
- Smeštaji
 - Pretraga
 - Tip
 - Naziv
 - Postojanje bazena (da/ne)
 - Postojanje spa centra (da/ne)
 - Prilagođeno za osobe sa invaliditetom (da/ne)
 - Postoji wifi (da/ne)
 - Sortiranje:
 - Naziv (rastuće i opadajuće)
 - Ukupan broj smeštajnih jedinica (rastuće i opadajuće)
 - Broj slobodnih smeštajnih jedinica (rastuće i opadajuće)
- Smeštajne jedinice
 - Pretraga
 - Donja granica za dozvoljen broj gostiju
 - Gornja granica za dozvoljen broj gostiju
 - Dozvoljen boravak kućnim ljubimcima (da/ne)
 - Cena za celu smeštajnu jedinicu
 - Sortiranje
 - Dozvoljen broj gostiju (rastuće i opadajuće)
 - Cena za celu smeštajnu jedinicu (rastuće i opadajuće)
- Korisnici
 - Pretraga
 - Ime
 - Prezime
 - Uloga
 - Sortiranje
 - Ime (rastuće i opadajuće)
 - Prezime (rastuće i opadajuće)
 - Uloga (rastuće i opadajuće)