**IT AKADEMIJA**

SMER - JAVA DEVELOPMENT

GRUPA - V

**TURISTIČKA AGENCIJA - JAVA EE WEB APLIKACIJA**

Dokumentacija za završni rad

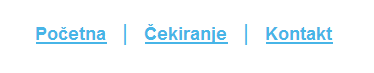
Mentor: Polaznik:

Prof. Vlada Marić Đorđije Čupić

**TURISTIČKA AGENCIJA - DOKUMENTACIJA ZA KORISNIKE**

Suština aplikacije je jednostavno rezervisanje i plaćanje izabranog hotela.

Na početnoj stranici aplikacije su tri linka ("Početna", "Čekiranje", "Kontakt") . Ova tri linka se takođe nalaze na svakoj stranici web sajta. Prvi link ("Početna") uvek vodi ka početnoj stranici sajta. Drugi link ("Čekiranje") je objašnjen kasnije u tekstu. Treći link ("Kontakt") uvek vodi ka stranici na kojoj su razni podaci koji se odnose na turističku agenciju (npr. lokacija agencije, email, kontakt telefon...).



Pored ova tri linka, na početnoj stranici je i padajuća lista država u kojoj korisnik klikom na ime države bira državu u koju želi da putuje.



Kada korisnik izabere državu pretraživač prelazi na stranicu na kojoj će moći da bira jedno od mesta, iz prethodno odabrane države, u kojem želi da rezerviše hotel. U slučaju da se predomisli, takođe na istoj stranici može da izabere neku drugu državu.



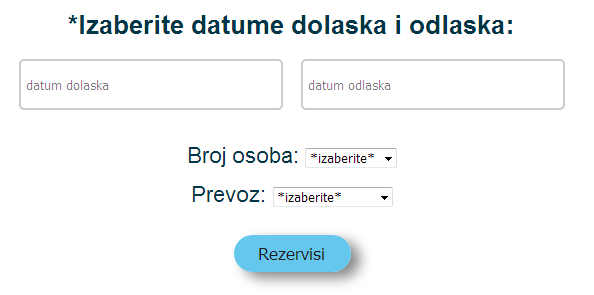
Korisnik potom bira odredište klikom na ime željene destinacije iz padajuće liste. Pretraživač sada prelazi na stranicu na kojoj je kratak opis izabrane destinacije i ponuda hotela.



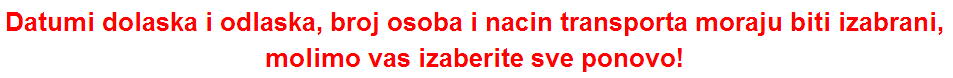
Klikom bilo na ime, zvezdice ili sliku hotela, pretraživač prelazi na stranicu koja opisuje detalje izabranog hotela. Na ovoj stranici je kratak opis hotela, zatim cene prenoćišta, cenovnik prevoznih sredstava, kao i nekoliko fotografija hotela i okoline.



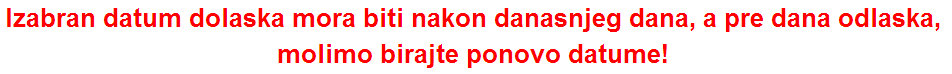
Klikom na "Dalje" korisnik dolazi na stranicu na kojoj bira datume rezervacije, broj osoba (tj. tip sobe - npr. dvokrevetna, trokrevetna...), kao i prevozno sredstvo. Datumi dolaska i odlaska se biraju u kalendarima koji se pojavljuju klikom na polja datuma (dolaska i odlaska) . Broj osoba i prevozno sredstvo se biraju iz padajuće liste.



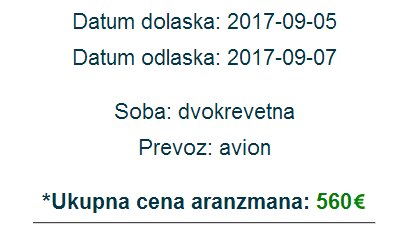
\* Sva polja moraju biti popunjena (datumi, kako bi bio nastavljen proces rezervacije. U slučaju da je neko polje ostalo prazno, korisnik će dobiti obaveštenje da nisu sva polja popunjena:



\* U slučaju da je pogrešno izabran datum (npr. datum dolaska posle datuma odlaska ili datum koji je već prošao) korisnik će takođe dobiti obaveštenje:



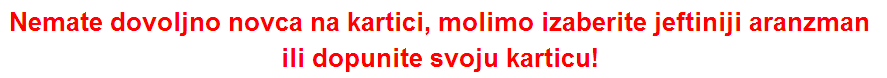
Klikom na "Rezerviši" korisnik prelazi na stranicu za unos ličnih podataka i podataka koji se odnose na online plaćanje (biranje kartice za plaćanje i unos podataka). Takođe, na ovoj stranici su prikazani prethodno izabrani datumi, tip sobe, prevozno sredstvo i ukupan iznos aranžmana koji korisnik treba da plati.

Kako bi rezervisao sobu u hotelu, korisnik mora da popuni sva polja u koja unosi lične podatke, da izabere tip kreditne kartice kojom plaća aranžman i unese podatke sa kreditne kartice. Zatim da klikne na "Plati".

\* Ukoliko nisu popunjena sva polja korisnik će dobiti obaveštenje da su sva polja obavezna.

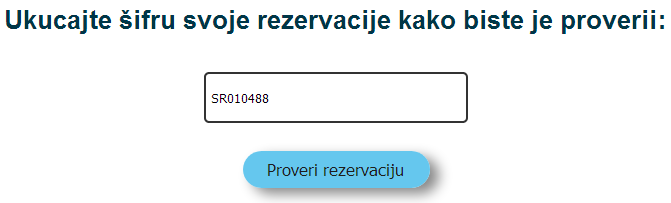
\* Ukoliko korisnik nema na kartici dovoljno novca kako bi platio ukupan iznos aranžmana dobiće obaveštenje:



Ukoliko plaćanje prođe uspešno, pretraživač prikazuje stranicu sa podacima o uplaćenom aranžmanu i šifru rezervacije.



Sada da objasnimo link "Čekiranje". Klikom na ovaj link sa bilo koje strane, dolazi se na stranicu na kojoj korisnik unosi šifru rezervacije prethodno rezervisanog aranžmana.



Šifra mora biti identična (osim veličine slova - eng. case insensitive) kao ona koju je dobio korisnik pri rezervaciji. Ukoliko je ispravno uneta šifra, klikom na "Proveri rezervaciju" pretraživač ide na stranicu sa prikazom podataka o prethodno rezervisanom aranžmanu.



\*Napomena: za korišćenje ove web aplikacije potrebno je koristiti pretraživač Google Chrome ili Mozilla Firefox. U slučaju korišćenja Mozille, neophodno je prethodno ukucati u Mozillinu traku za adrese about:config i postaviti da vrednost (eng. value) dom.forms.datetime bude 'true'. Ovo omogućava da Mozilla učita HTML 5 input type date koji je korišćen u aplikaciji.

**TURISTIČKA AGENCIJA - DOKUMENTACIJA ZA DEVELOPERE**

RADNO OKRUŽENJE

Za izradu projekta korišćeni su:

Jezici:

- CSS (blue island template -> http://www.free-css.com/free-css-templates/page56/blue-island)

- HTML

- JAVA EE

- SQL i HQL

- XML

Alati:

- Hibernate

- phpMyAdmin

Server za bazu podataka:

- MariaDB

Okruženje:

- NetBeans IDE 8.2

Serverska platforma:

- GlassFish Server

- XAMPP

Biblioteke (eng. Libraries):

- JDK 1.8

- Hibernate 4.3x(JPA2.1)

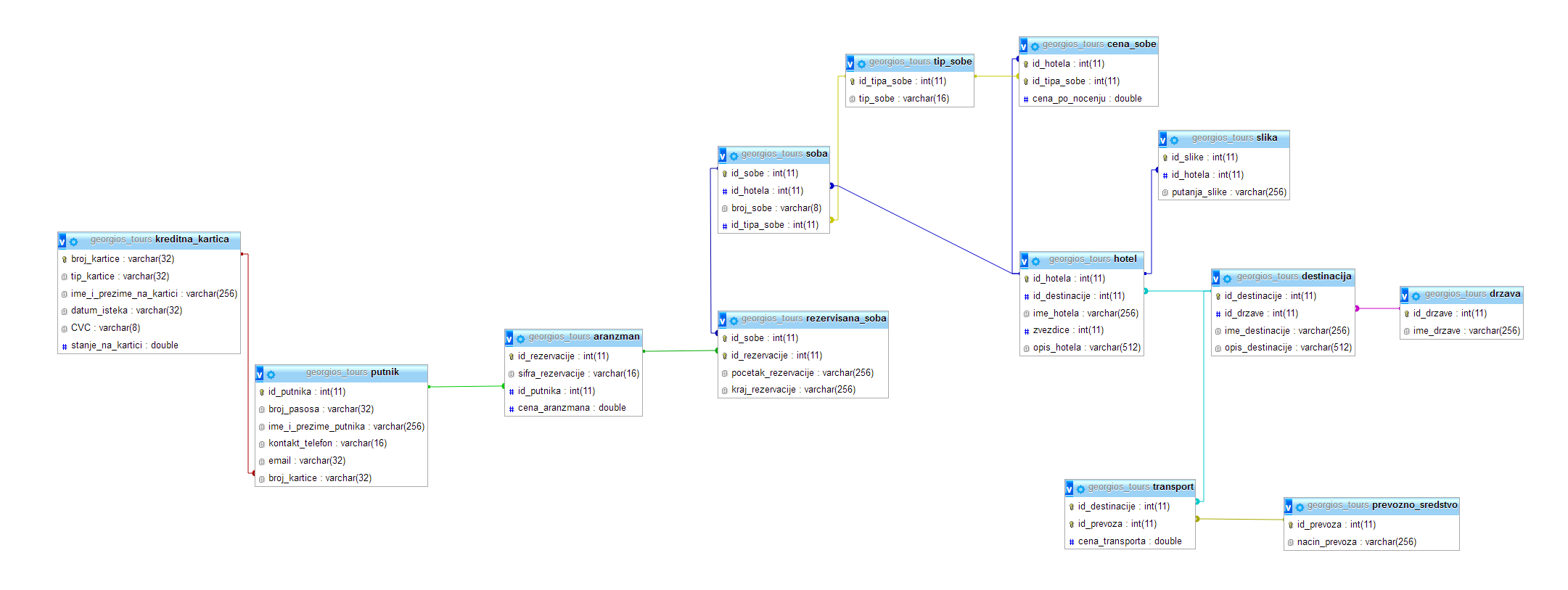
- MySQL JDBC Driver

- GlassFish Server

- joda-time 2.9.9

BAZA PODATAKA

Prikaz tabela i relacija u UML-u (eng. Unifed Modeling Language):



PAKETI

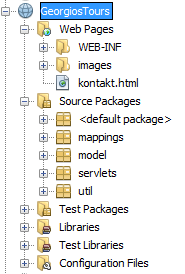
Infrastrukturu koda aplikacije čine četiri paketa (+ defaultni paket) redom:

1) mappings

2) model

3) servlets

4) util



Paket 'mappings' sadrži mapiranja svih entiteta iz baze podataka (primer - vidi sliku ispod) , kao i njihove relacije kreirane putem XML-a. Za mapiranje je korišćen alat Hibernate.

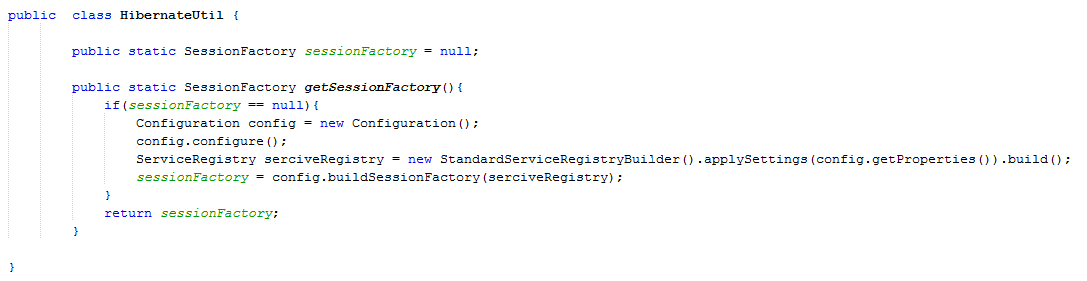


Paket 'model' sadrži sve entitete koji predstavljaju tabele iz baze podataka. Ovi entiteti predstavljaju klasične Java POJO (eng. Plain Old Java Object) objekte (primer - vidi sliku ispod).

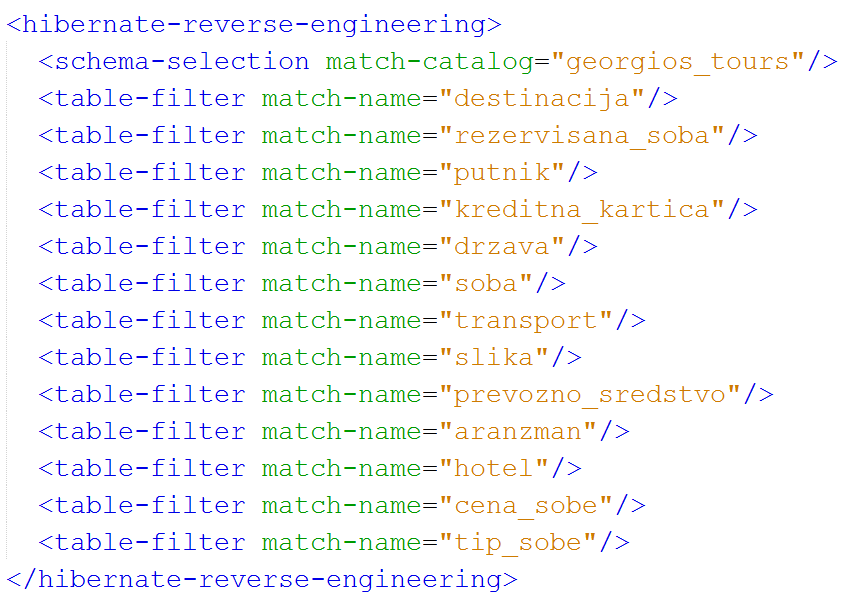


Paket 'servlets' sadrži sve servlete aplikacije. U servletima su potpuno spojeni logika (eng. controller) i izgled (eng. view) aplikacije. Logika je urađena upotrebom programskog jezika Java uz ORM (eng. Object Relational Mapping) alat Hibernate, a izgled upotrebom kombinacije HTML-a i CSS-a. Servleti su detaljnije opisani dalje u tekstu.

Paket 'util' sadrži pomoćnu klasu HibernateUtil (vidi sliku ispod) koja omogućava pozivanje metode getSessionFactory() i dobijanje objekta sesije koji će biti korišćen za dalju manipluaciju objekatima. Za ovu klasu primenjen je Singleton šablon (eng. pattern).



U defaultnom paketu su dva fajla napisana u XML-u. Hibernate-ov konfiguracioni fajl (hibernate.cfg.xml) neophodan za korišćenje Hibernate alata i mapiranje objekata, i reverse-engineering fajl (hibernate.reveng.xml) koji je generisan pomoću Hibernate Reverse Engineering Wizarda.

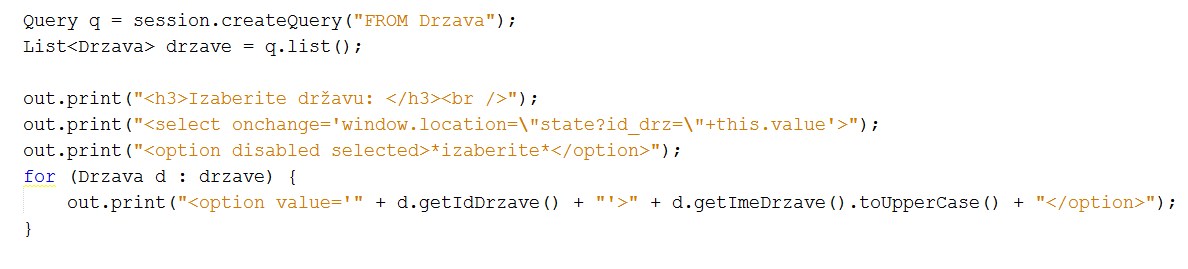
 

SERVLETI

U paketu 'servlets' se nalazi čitava logika koda aplikacije napisana u Javi. Paket sadrži devet servleta (choose, state, destination, hotel, book, payment, reservation, check\_in\_form, check\_in) ekstenzija .java.

1) servlet choose:

Sastoji se od padajuće liste država. Države dobijamo iz baze pomoću upita napisnog u HQL-u (eng. Hibernate Query Language), a prikazujemo ih prolaskom kroz listu generičke klase (List <Drzava>) upotrebom enchanced for petlje.

<select onchange = 'window.location =...'> html tag omogućava da klikom na bilo koju državu iz liste prelazimo na sledeću stranicu (servlet state.java) kojoj smo prosledili određene parametre (state?id\_drz=this.value). Ovi parametri su u stvari vrednosti koje uzimamo iz taga option (atribut value) iz enchanced for petlje.

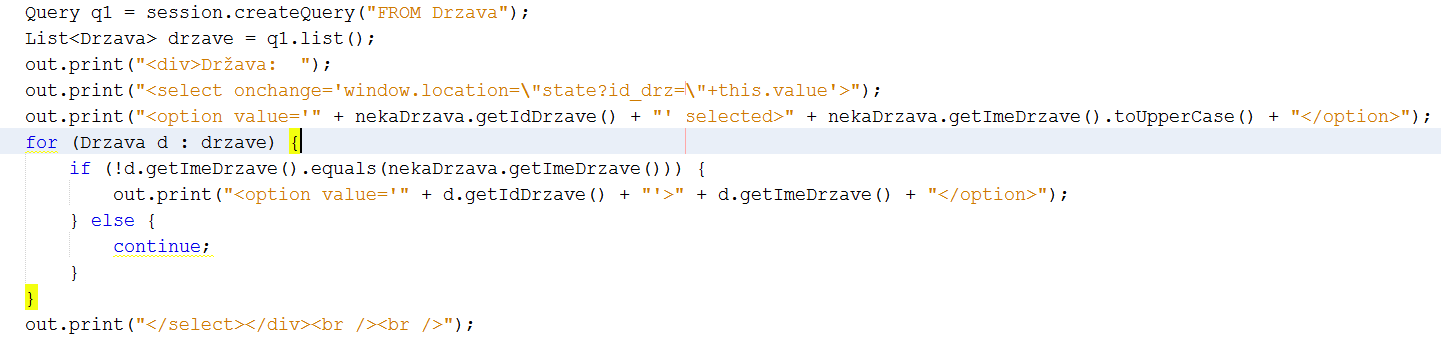
2) servlet state:

Parametre poslate sa prethodne stranice hvatamo na sledeći način:

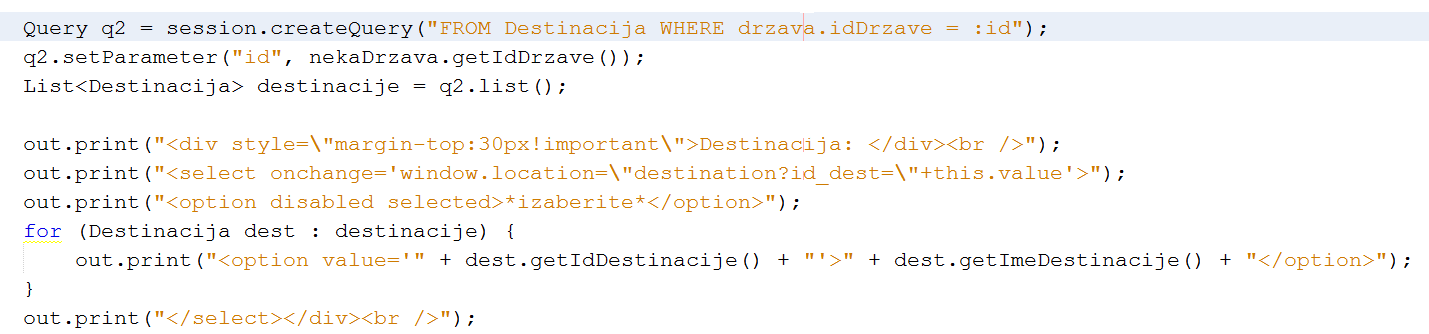
C:\Users\Djordjije\Desktop\za programere\parametri.png

Na osnovu uhvaćenog parametra dobijamo državu koju je korisnik izabrao.

Prvi upit predstavlja sličnu padajuću listu kao na prethodnom servletu, s tim što je modifikovana kako korisnik ne bi mogao da izabere državu koju je već izabrao.



Drugi upit, koji je parametrizovan, uzima iz baze destinacije iz prethodno izabrane države. Zatim prikazuje padajuću listu destinacija u kojoj korisnik bira željenu destinaciju na isti način kao što je ranije birao i državu. Odabirom destinacije, prelazimo na sledeći servlet (destination.java) kome smo, po istom principu kao i pre, poslali određene parametre.

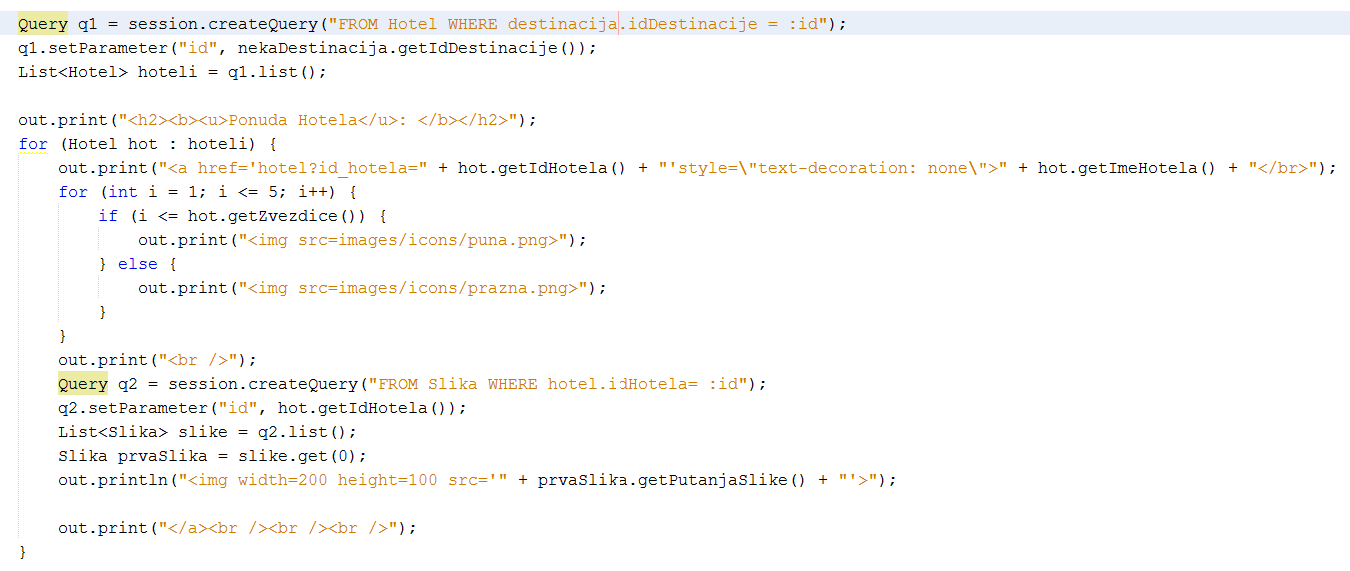


3) servlet destination:

Po istom principu slanja i hvatanja parametra, određujemo destinaciju koju je korisnik izabrao.

Sada koristimo upit kojim uzimamo iz baze hotele za tu destinaciju. Prolazimo kroz listu i prikazujemo imena hotela, zvezdice (jednostavan if-else upit), kao i dodatnim upitom prvu sliku svakog hotela (slike.get(0)).

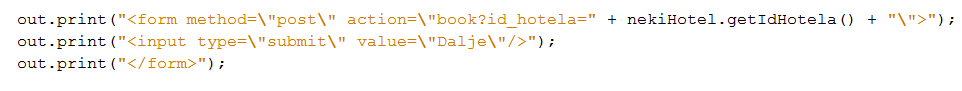
Usput smo takođe postavili linkove (<a href> tag) i to parametrizovane, kako bi prešli na stranicu sa detaljnijim opisom hotela (hotel.java), klikom na željeni hotel (bilo na ime, zvezdice ili sliku hotela).



4) servlet hotel:

Za ovu stranu koristimo tri upita (po istom principu kao i pre) za dobijanje detalja hotela (cena noćenja, cena transporta i slike hotela).

Pored toga, koristimo i HTML form tag sa parametrizovanim atributom action, koji korisnika, klikom na 'Dalje', šalje na sledeću stranicu (book.java).



5) servlet book:

Na ovoj stranici korisnik preko forme bira datume, tip sobe i prevozno sredstvo. Zatim se prosleđuju izabrani podaci preko parametrizovanog action atributa i input tipa submita na sledeću stranicu (payment.java).

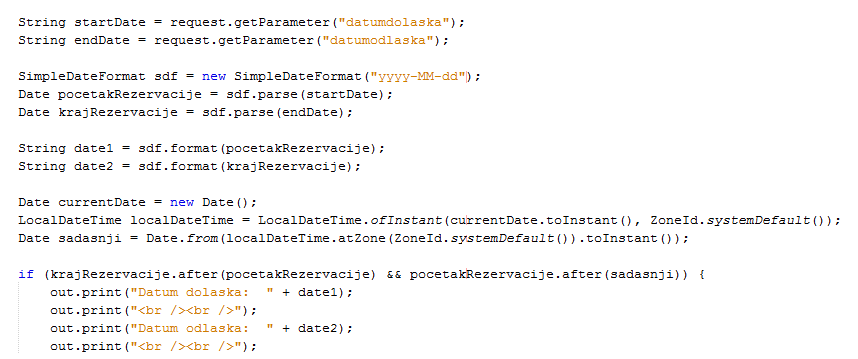


\* Napomena :

Kako je korišćen input type date (HTML 5), za njegov prikaz, u slučaju da se koristi pretraživač Mozilla Firefox neophodno je prethodno pokrenuti Mozillu , ukucati u adress traci about:config i postaviti da vrednost (eng. value) za dom.form.datetime bude 'true'. Internet Explorer još ne podržava HTML 5 input type date, a Google Chrome potpuno podržava bez ikakvih podešavanja.

6) servlet payment:

Na ovoj stranici hvatamo parametre sa prethodne stranice i prikazujemo ih. Za prikaz datuma koristimo Javin SimpleDateFormat preko koga formatiramo datume. Kako bi proverili ispravnost izabranih datuma dolaska i odlaska koristimo funkciju iz util.Date paketa x.after(y). Takođe poredimo izabrani datum dolaska sa sadašnjim datumom (\* napomena : za dobijanje sadašnjeg datuma koristimo funkcije iz LocalDateTime i ZoneId iz paketa java.time zbog čega je neophodno da importujemo ovaj paket) kako ne bi mogao da bude izabran datum koji je već prošao (takođe ni današnji dan) za rezervisanje hotelske sobe.



Dalje, pravimo upite kojima uzimamo cenu noćenja i transporta iz baze. Za formiranje ukupne cene aranžmana potrebna nam je razlika u danima između datuma dolaska i datuma odlaska koje je korisnik izabrao (\* napomena : za dobijanje ove razlike neophodno je prethodno skinuti (eng. download) biblioteku joda.time i dodati je u projekat).

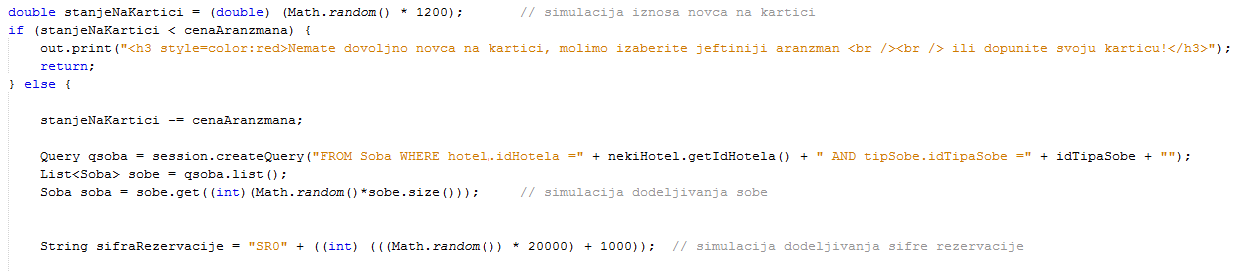


Zatim imamo formu za unos ličnih i podataka sa kreditne kartice, koje korisnik treba sam da upiše. Ova forma nas vodi na sledeću stranicu (reservation.java)



6) servlet reservation:

Ovaj servlet služi da obavi simulaciju dodeljivanja korisniku i rezervisanja sobe, kao i skidanja novca sa korisnikove kreditne kartice.



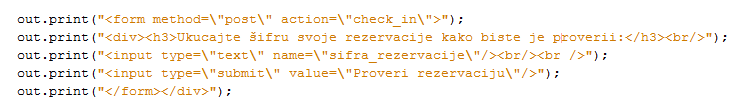
U slučaju da sve prođe kako treba, podaci o aranžmanu, putniku i kreditnoj kartici korisnika će biti ubačeni u bazu podataka, a korisnik će na pretreživaču dobiti prikaz detalja rezervisanog aranžmana i šifru svoje rezervacije koja služi za proveru informacija o aranžmanu preko stranice 'Čekiranje'.



7) servlet check in\_form:

Klikom na link 'Čekiranje' aktivira se ova stranica.

Na njoj se nalazi forma pomoću koje korisnik šalje unetu šifru rezervacije radi provere informacija aranžmana na sledećoj stranici (check\_in.java).



8) servlet check in:

Proveravamo poslatu šifru rezervacije s prethodne stranice da li odgovara šifri iz baze. U slučaju da odgovora, uzimamo podatke o aranžmanu iz baze i prikazujemo korisniku na pretraživaču.



Link 'Kontakt' vodi ka običnoj HTML stranici za prikaz osnovnih informacija o turističkoj agenciji.