

Restful Booker API – Dokumentacija za Testnu Automatizaciju

Autor: Kenan Bejtić
Datum: 26.04.2025

Sadržaj

1. Kako pokrenuti automatizirane testove	3
2. Ciljevi	4
Provjera osnovnih funkcionalnosti	4
Osigurati Pouzdanost Servisa	4
Simulacija Realnih Korisničkih Scenarija	4
3. Testovi Osnovne Funkcionalnosti	5
3.1. Testovi Autentifikacije	5
3.2. Testovi Kreiranja Rezervacije	5
3.3. Testovi Čitanja Rezervacije	6
3.4. Testovi Ažuriranja Rezervacije	6
3.5. Testovi Brisanja Rezervacije	7
4. Smoke Testovi	8
5. Testovi Realističnog Korisničkog Toka	9
5.1. Tok Potpunog Životnog Ciklusa Rezervacije	9
5.2. Tok Pretrage i Filtriranja	10
6. Greške	11
6.1. Koraci za reprodukciju greške sa atributom checkin	11
6.3. Table of bugs relating to checkin attribute	13
6.4. Koraci za reprodukciju greške vezane za atribut totalprice	15
6.5. Tabela grešaka vezanih za atribut totalprice	16

1. Kako pokrenuti automatizirane testove

Opcija 1:

Pod uvjetom da imate instaliran Python, otvorite PowerShell i pokrenite sljedeću naredbu:

```
Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass
```

Zatim pokrenite skriptu `run_tests.ps1` unutar glavnog direktorija (foldera) gdje se skripta nalazi koristeći naredbu:

```
".\run_tests.ps1"
```

Ovo će automatski instalirati potrebne Python biblioteke i pokrenuti testove. Testovi će prikazati 10 upozorenja na kraju, koja su objašnjena u poglavlju 6 pod nazivom „Greške“ na stranici 11.

Opcija 2:

Pod uvjetom da imate instaliran Python, potrebno je otvoriti bilo koji terminal i pokrenuti sljedeću naredbu:

```
pip install pytest requests
```

Nakon što je to završeno, jednostavno trebate pokrenuti `pytest -v` unutar glavnog direktorija, i svi testovi će se pokrenuti.

2. Ciljevi

Provjera osnovnih funkcionalnosti

- **CRUD operacije:** Validirati kreiranje (POST /booking), čitanje (GET /booking i GET /booking/{id}), ažuriranje (PUT i PATCH /booking/{id}) i brisanje (DELETE /booking/{id}) da se ponašaju prema specifikacijama, vraćajući ispravne status kodove i podatke.
- **Autentifikacija i autorizacija:** Osigurati da validne vjerodajnice preko POST /auth daju upotrebljiv token, te da zaštićene krajnje tačke odbijaju zahtjeve bez validnog tokena ili s isteklim/nevažećim tokenom.
- **Validacija podataka i rukovanje greškama:** Potvrditi da neispravni ili nepotpuni zahtjevi vraćaju odgovarajuće 4xx greške, te da operacije nad nepostojećim resursima vraćaju 404 Not Found (ili dogovorene kodove).

Osigurati Pouzdanost Servisa

- **Smoke testovi:** Uspostaviti minimalni skup testova (ping, autentifikacija, kreiranje, čitanje, brisanje) za brzo otkrivanje većih kvarova u servisu.

Simulacija Realnih Korisničkih Scenarija

- **Potpuni životni ciklus rezervacije:** Simulirati korisničko putovanje od prijave, kreiranja rezervacije, potpunog zamjenjivanja, djelimičnog ažuriranja, preuzimanja, do brisanja, kako bi se osigurala cjelovita funkcionalnost od početka do kraja.
- **Pretraga i filtriranje:** Oponašati scenarije sa više korisnika kreiranjem više rezervacija sa zajedničkim atributima i pretragom preko parametara upita (npr. ime, prezime, datum prijave/odjave) radi provjere ispravnosti filtriranja.

3. Testovi Osnovne Funkcionalnosti

3.1. Testovi Autentifikacije

Testni Slučaj	Opis	Koraci	Očekivani Rezultat	Tip
TC-Auth-01	Validni kredencijali	1. POST /auth sa ispravnim korisničkim imenom/lozinkom	200 OK i odgovor u JSON formatu sadrži neprazan token	Pozitivan
TC-Auth-02	Nevalidni kredencijali	1. POST /auth sa pogrešnim korisničkim imenom/lozinkom	200 OK i tijelo JSON odgovora sadrži reason="Bad credentials"	Negativan

3.2. Testovi Kreiranja Rezervacije

Testni Slučaj	Opis	Koraci	Očekivani Rezultat	Tip
TC-CR-01	Kreiranje rezervacije (validno)	1. Dobaviti token putem TC-Auth-01 2. POST /booking sa kompletnim i validnim JSON tijelom koje uključuje: firstname, lastname, totalprice, depositpaid, bookingdates, additionalneeds	200 OK i tijelo odgovora sadrži bookingid plus objekt rezervacije koji odgovara poslanim podacima	Pozitivan
TC-CR-02	Kreiranje rezervacije (nedostaje podatak)	1. POST /booking izostavljajući obavezno polje (npr. lastname)	500 Bad Request sa opisnom porukom o grešci	Negativan

3.3. Testovi Čitanja Rezervacije

Testni Slučaj	Opis	Koraci	Očekivani Rezultat	Tip
TC-RD-01	Preuzimanje postojeće rezervacije po ID-u	1. Kreirati rezervaciju putem TC-CR-01 2. GET /booking/{id}	200 OK i tijelo odgovora sadrži detalje rezervacije tačno kako su kreirani	Pozitivan
TC-RD-02	Lista rezervacija sa filterom (nema rezultata)	1. GET /booking?firstname=NonExistentName	200 OK i prazan JSON niz ([])	Negativan
TC-RD-03	Preuzimanje rezervacije koja ne postoji	1. GET /booking/999999	404 Not Found	Negativan

3.4. Testovi Ažuriranja Rezervacije

Testni Slučaj	Opis	Koraci	Očekivani Rezultat	Tip
TC-UP-01	Potpuno ažuriranje (PUT) (validno)	1. Kreirati rezervaciju putem TC-CR-01 2. PUT /booking/{bookingid} sa potpuno novim validnim tijelom zahtjeva i validnim tokenom	200 OK i JSON odgovor odgovara novom tijelu zahtjeva	Pozitivan
TC-UP-02	Djelimično ažuriranje (PATCH) (validno)	1. Kreirati rezervaciju putem TC-CR-01 2. PATCH /booking/{bookingid} sa JSON tijelom koje sadrži jedno ili dva polja (npr. firstname) i validnim tokenom	200 OK i samo ažurirana polja su izmijenjena u odgovoru	Pozitivan
TC-UP-03	Neautorizirano ažuriranje	1. Kreirati rezervaciju putem TC-CR-01 2. PATCH /booking/{bookingid} bez slanja tokena	403 Forbidden	Negativan

3.5. Testovi Brisanja Rezervacije

Testni Slučaj	Opis	Koraci	Očekivani Rezultat	Tip
TC-DEL-01	Brisanje postojeće rezervacije (validno)	1. Kreirati rezervaciju putem TC-CR-01 2. DELETE /booking/{bookingid} sa validnim tokenom 3. GET /booking/{bookingid}	201 ili 204 na DELETE; naknadni GET vraća 404 Not Found	Pozitivan
TC-DEL-02	Brisanje nepostojeće rezervacije	1. DELETE /booking/999999 sa validnim tokenom	404 Not Found ili 405 Method Not Allowed	Negativan
TC-DEL-03	Brisanje postojeće rezervacije koristeći osnovni autorizacijski header	1. Kreirati rezervaciju putem TC-CR-01 2. DELETE /booking/{bookingid} sa autorizacijskim headerom kao alternativom za Cookie	201 or 204 on DELETE	Pozitivan

4. Smoke Testovi

Minimalne validacije kritične putanje kako bi se osigurala ispravnost servisa.

Testni Slučaj	Opis	Koraci	Očekivani Rezultat	Tip
TC-SM-01	Provjera zdravlja sistema i autentifikacija	1. Pošalji GET /ping 2. Pošalji POST /auth sa validnim korisničkim imenom/lozinkom	GET /ping vraća Status 201, POST /auth vraća Status 200 i odgovor sadrži token	Smoke
TC-SM-02	Kreiranje, preuzimanje i brisanje rezervacije	1. Dobaviti token 2. Poslati POST /booking sa validnim tijelom zahtjeva i zabilježiti bookingid 3. Poslati GET /booking/{bookingid} 4. Poslati DELETE /booking/{bookingid} sa tokenom	POST vraća Status 200 i bookingid, GET vraća Status 200, sadržaj odgovora odgovara kreaciji, DELETE vraća Status 201/204, naknadni GET vraća 404	Smoke
TC-SM-03	Lista rezervacija (osnovno)	1. Pošalji GET /booking	Status 200, vraća JSON niz ID-ova rezervacija (barem uključujući one novo kreirane)	Smoke

5. Testovi Realističnog Korisničkog Toka

5.1. Tok Potpunog Životnog Ciklusa Rezervacije

Testni Slučaj	Opis	Koraci	Očekivani Rezultat	Tip
TC-E2E-01	Tok potpunog životnog ciklusa rezervacije	<ol style="list-style-type: none">1. Autentifikacija: POST /auth (validni kredencijali), zabilježiti token2. Kreiranje: POST /booking (validan payload, sa tokenom), zabilježiti bookingid3. Provjera kreacije: GET /booking/{bookingid}, payload odgovara kreaciji4. Potpuna zamjena: PUT /booking/{bookingid} (novi payload, sa tokenom)5. Provjera zamjene: GET /booking/{bookingid}, payload odgovara zamjeni6. Djelimično ažuriranje: PATCH /booking/{bookingid} (npr. promjena firstname, sa tokenom)7. Provjera djelimičnog ažuriranja: GET /booking/{bookingid}, samo izmijenjena polja su promijenjena8. Brisanje: DELETE /booking/{bookingid} (sa tokenom)9. Provjera brisanja: GET /booking/{bookingid} vraća 404 Not Found	<p>Autentifikacija vraća 200 + token</p> <p>Kreacija vraća 200 + bookingid</p> <p>GET vraća 200 + tačne podatke</p> <p>PUT vraća 200 + nove podatke</p> <p>PATCH vraća 200 + ažurirane podatke</p> <p>DELETE vraća 201/204</p> <p>Završni GET vraća 404 Not Found</p>	End-to-End

5.2. Tok Pretrage i Filtriranja

Testni Slučaj	Opis	Koraci	Očekivani Rezultat	Tip
TC-E2E-02	Pretraga i filtriranje rezervacija	<p>1. Kreirati A: POST /booking (firstname="NameA", bookingdates A)</p> <p>2. Kreirati B: POST /booking (firstname="NameA", bookingdates B)</p> <p>3. Kreirati C: POST /booking (firstname="NameC")</p> <p>4. Filtrirati po imenu: GET /booking?firstname=NameA</p> <p>5. Filtrirati po prezimenu: GET /booking?lastname=<B_lastname></p> <p>6. Filtrirati po datumima: GET /booking?checkin=<A.checkin>&checkout=<A.checkout></p>	<p>Filtriranje firstname=NameA vraća niz koji sadrži ID-eve A i B</p> <p>Filtriranje lastname=B_lastname vraća niz koji sadrži ID od B</p> <p>Filtriranje po datumima A vraća niz koji sadrži ID od A samo</p>	End-to-End

6. Greške

Problem: API filter ?checkin= ne vraća novo kreirane rezervacije. Samo upiti bazirani na checkin datumu su pogođeni ovim problemom. Svi ostali filteri (firstname, lastname, checkout) rade ispravno. Također postoji greška vezana za atribut totalprice koji se vraća prilikom upita za rezervaciju. Testovi koji se odnose na ove greške nalaze se u /tests/test_bug.py.

6.1. Koraci za reprodukciju greške sa atributom checkin

- **Autentifikacija**

Prvo je potrebno regularno autentificirati našu sesiju.

```
curl -s -X POST https://restful-booker.herokuapp.com/auth \
-H "Content-Type: application/json" \
-d '{"username":"admin","password":"password123"}
```

- **Kreirati rezervaciju i zabilježiti vraćeni bookingid**

```
curl -s -X POST https://restful-booker.herokuapp.com/booking \
-H "Content-Type: application/json" \
-H "Cookie: token=<token>" \
-d '{
  "firstname":"BugTest",
  "lastname":"User",
  "totalprice":111,
  "depositpaid":false,
  "bookingdates":{
    "checkin":"2025-11-01",
    "checkout":"2025-11-05"
  },
  "additionalneeds":"None"
}
```

}'

- **Primijeniti filter “checkin”**

```
curl -s -X GET "https://restful-booker.herokuapp.com/booking?checkin=2025-11-01"
```

- **Pregledati odgovor**

There will be an array of { "bookingid": ... } objects. The newly created bookingid will not appear.

6.3. Table of bugs relating to checkin attribute

Testni Slučaj	Opis	Koraci	Očekivani Rezultat	Tip
TC-BUG-01	Filtriranje samo po checkin-u	1. Kreirati rezervaciju A 2. GET /booking?checkin=2025-11-01	200 OK, niz odgovora NE sadrži { "bookingid": <A> } (greška); rezervacija A i dalje postoji	Greška
TC-BUG-02	Filtriranje po firstname + checkin	1. Kreirati rezervaciju A 2. GET /booking?firstname=ComboTest &checkin=2025-11-01	200 OK, niz odgovora NE sadrži { "bookingid": <A> } (greška); rezervacija A i dalje postoji	Greška
TC-BUG-03	Filtriranje po lastname + checkin	1. Kreirati rezervaciju A 2. GET /booking?lastname=User&checkin=2025-11-01	200 OK, niz odgovora NE sadrži { "bookingid": <A> } (greška); rezervacija A i dalje postoji	Greška
TC-BUG-04	Filtriranje po firstname + lastname + checkin	1. Kreirati rezervaciju A 2. GET /booking?firstname=ComboTest &lastname=User&checkout=2025-11-05	200 OK, niz odgovora NE sadrži { "bookingid": <A> } (greška); rezervacija A i dalje postoji	Greška

TC-BUG-05	Filtriranje po firstname + checkin + checkout	1. Kreirati rezervaciju A 2. GET /booking?firstname=ComboTest &checkin=2025-11-01&checkout =2025-11-05	200 OK, niz odgovora NE sadrži { "bookingid": <A> } (greška); rezervacija A i dalje postoji	Greška
TC-BUG-06	Filtriranje po lastname + checkin + checkout	1. Kreirati rezervaciju A 2. GET /booking?lastname=User&check in=2025-11-01&checkout=2025- 11-05	200 OK, niz odgovora NE sadrži { "bookingid": <A> } (greška); rezervacija A i dalje postoji	Greška
TC-BUG-07	Filtriranje po firstname + lastname + checkin + checkout	1. Kreirati rezervaciju A 2. GET /booking?firstname=ComboTest &lastname=User&checkin=2025 -11-01&checkout=2025-11-05	200 OK, niz odgovora NE sadrži { "bookingid": <A> } (greška); rezervacija A i dalje postoji	Greška

6.4. Koraci za reprodukciju greške vezane za atribut totalprice

- **Autentifikacija**

Prvo je potrebno regularno autentificirati našu sesiju.

```
curl -s -X POST https://restful-booker.herokuapp.com/auth \
-H "Content-Type: application/json" \
-d '{"username":"admin","password":"password123"}
```

- **Kreirati rezervaciju sa cijenom u obliku broja s floating-point**

```
curl -s -X POST https://restful-booker.herokuapp.com/booking \
-H "Content-Type: application/json" \
-H "Cookie: token=<token>" \
-d '{
  "firstname":"BugTest",
  "lastname":"Precision",
  "totalprice":123.45,
  "depositpaid":true,
  "bookingdates":{"checkin":"2025-11-01","checkout":"2025-11-05"},
  "additionalneeds":"None"
}'
```

Zabilježiti vraćeni bookingid.

- **Preuzeti rezervaciju**

```
curl -s -X GET https://restful-booker.herokuapp.com/booking/{bookingid}
```

- **Pregledati odgovor**

Očekivano: "totalprice":123.45

Stvarno: izgubljena preciznost (npr. "totalprice":123 ili neki drugi cijeli broj)

6.5. Tabela grešaka vezanih za atribut totalprice

Testni Slučaj	Opis	Koraci	Očekivani Rezultat	Tip
TC-BUG-08	Gubitak decimalnih vrijednosti (123.45 postaje 123)	1. Kreirati rezervaciju sa totalprice: 123.45 2. GET /booking/{id}	200 OK, JSON odgovor sadrži "totalprice": 123.45	Greška
TC-BUG-09	Gubitak malih decimalnih vrijednosti (0.99 postaje 0)	1. Kreirati rezervaciju sa totalprice: 0.99 2. GET /booking/{id}	200 OK, JSON odgovor sadrži "totalprice": 0.99	Bug