Informacioni sistemi

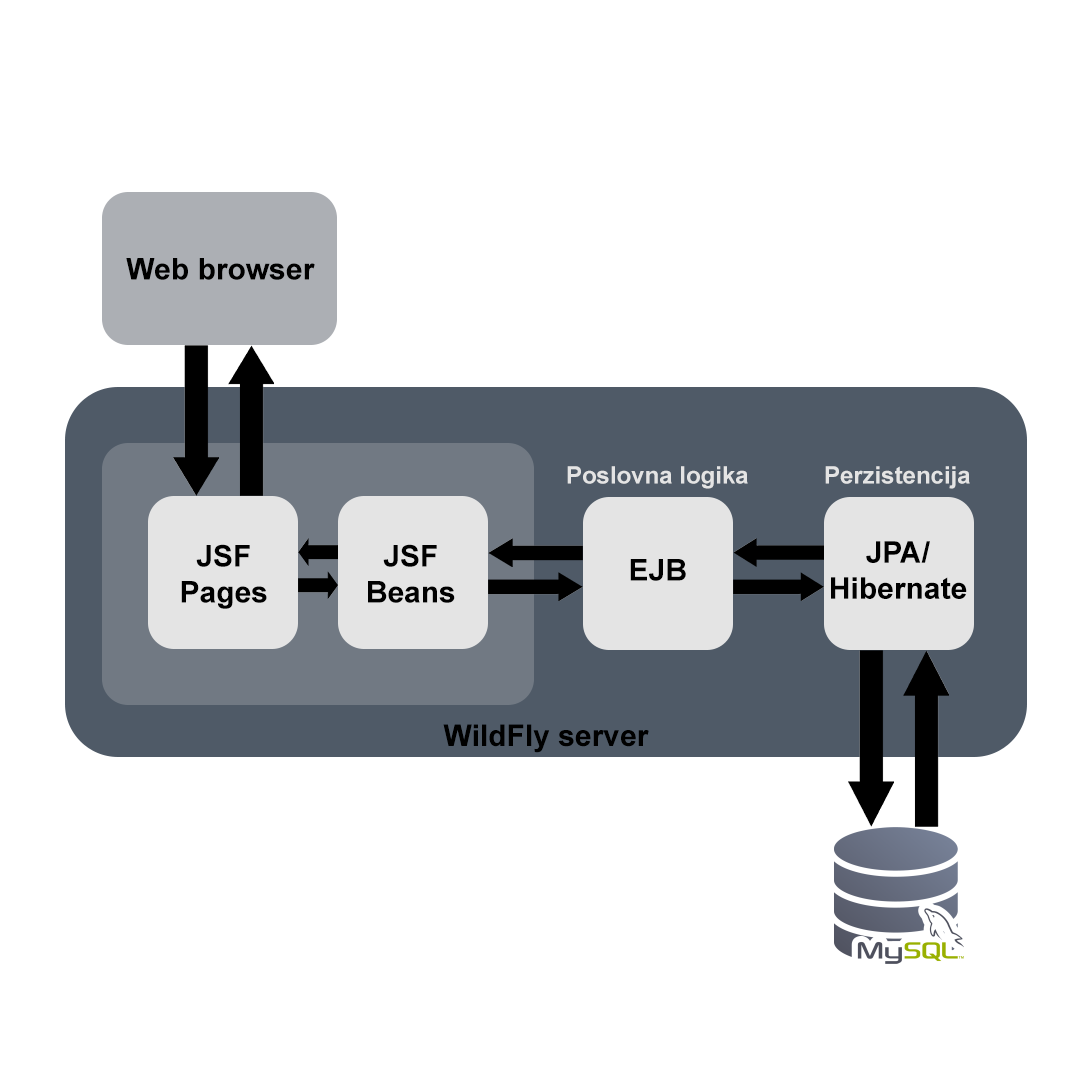
Drugi deo projekta

Student: Aleksa Ilić

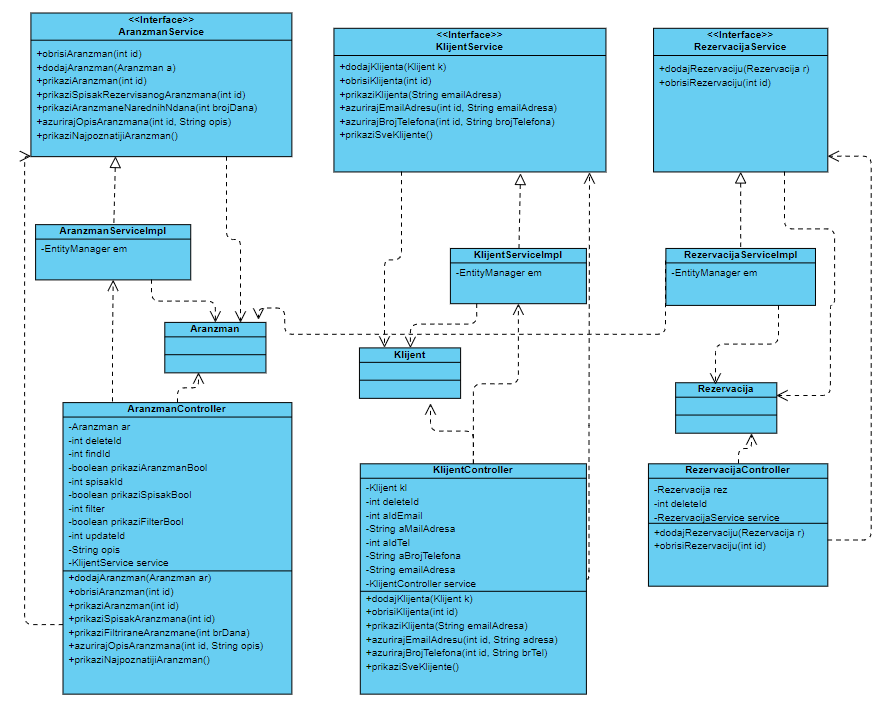
Broj indeksa: 18664

E-mail: [aleksa.ilic@elfak.rs](mailto:aleksa.ilic@elfak.rs)

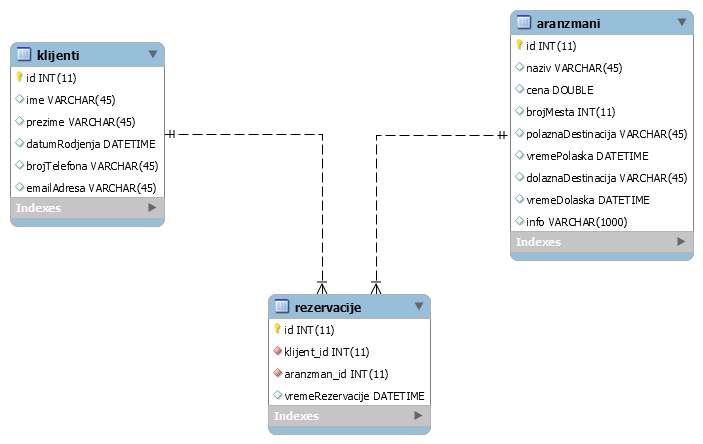
1. Arhitektura projekta



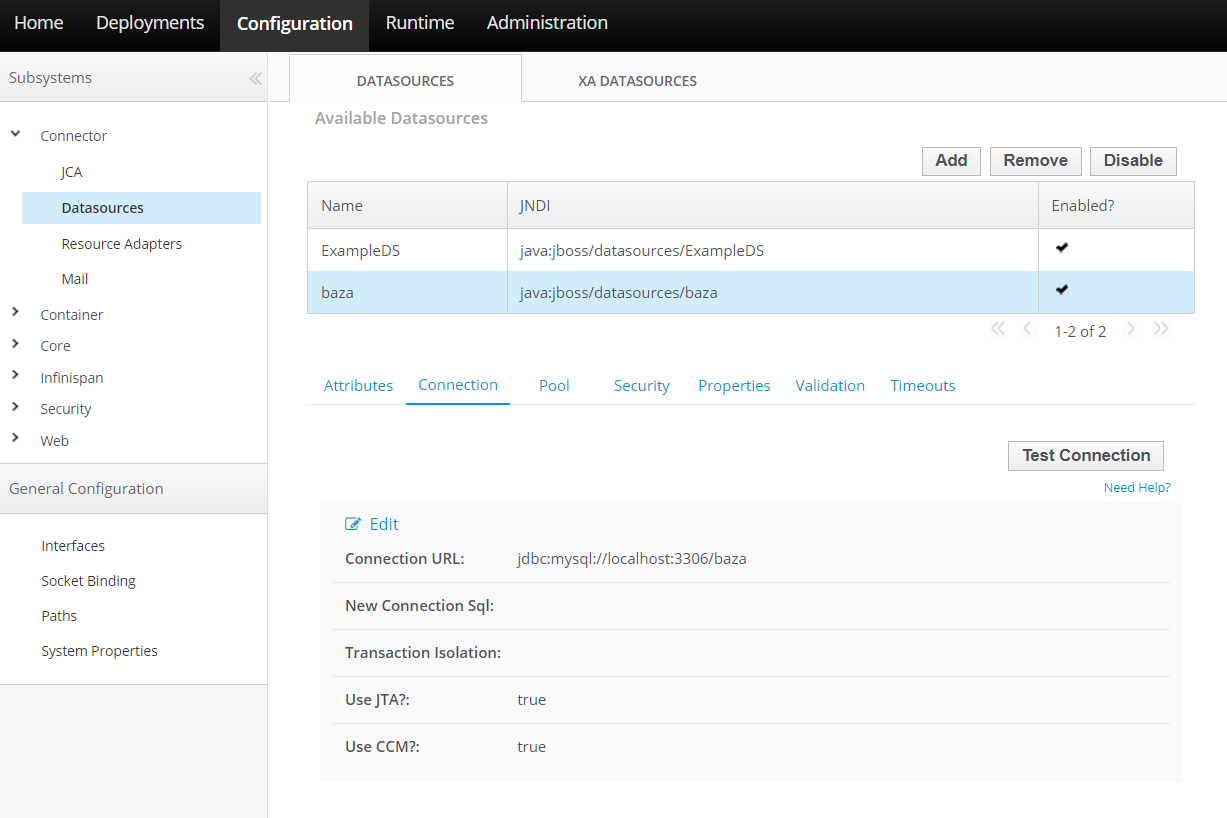
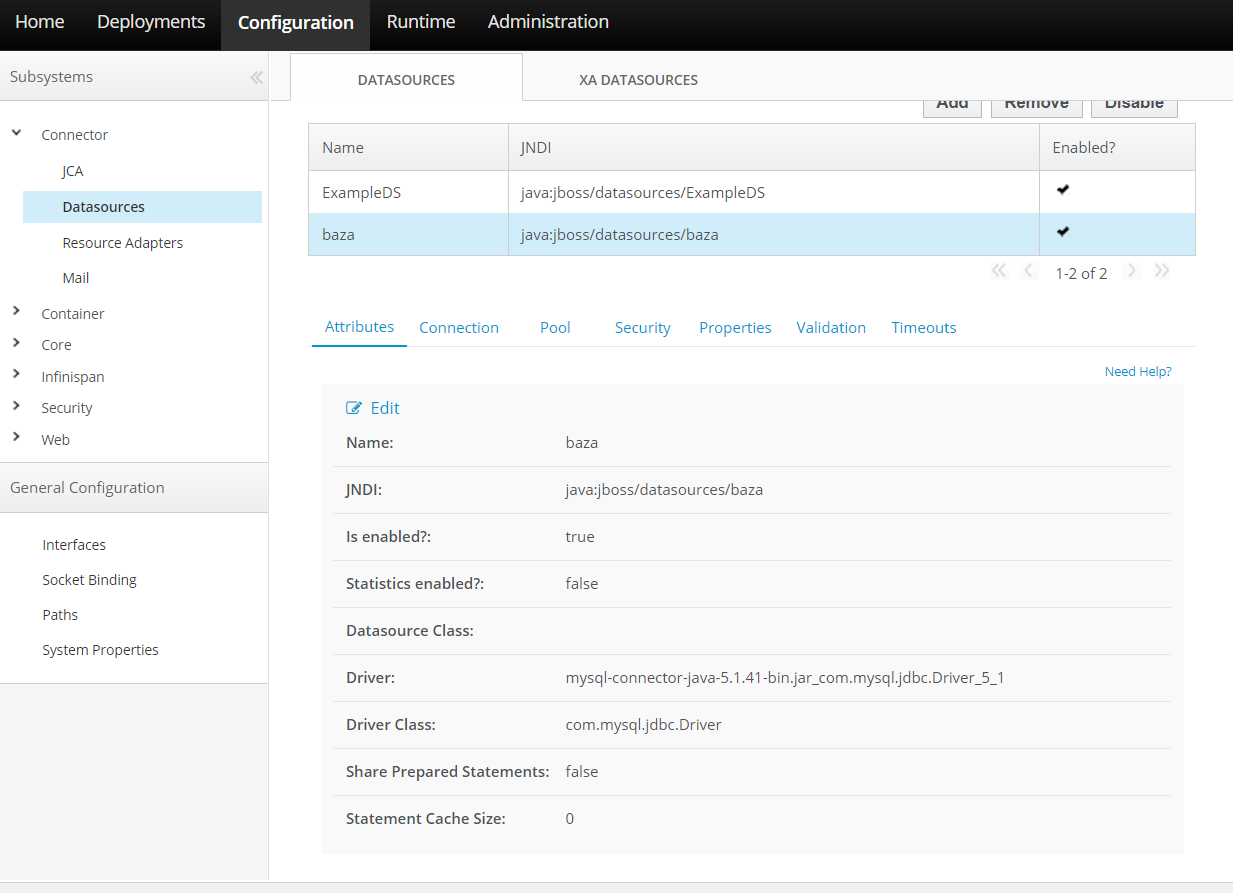
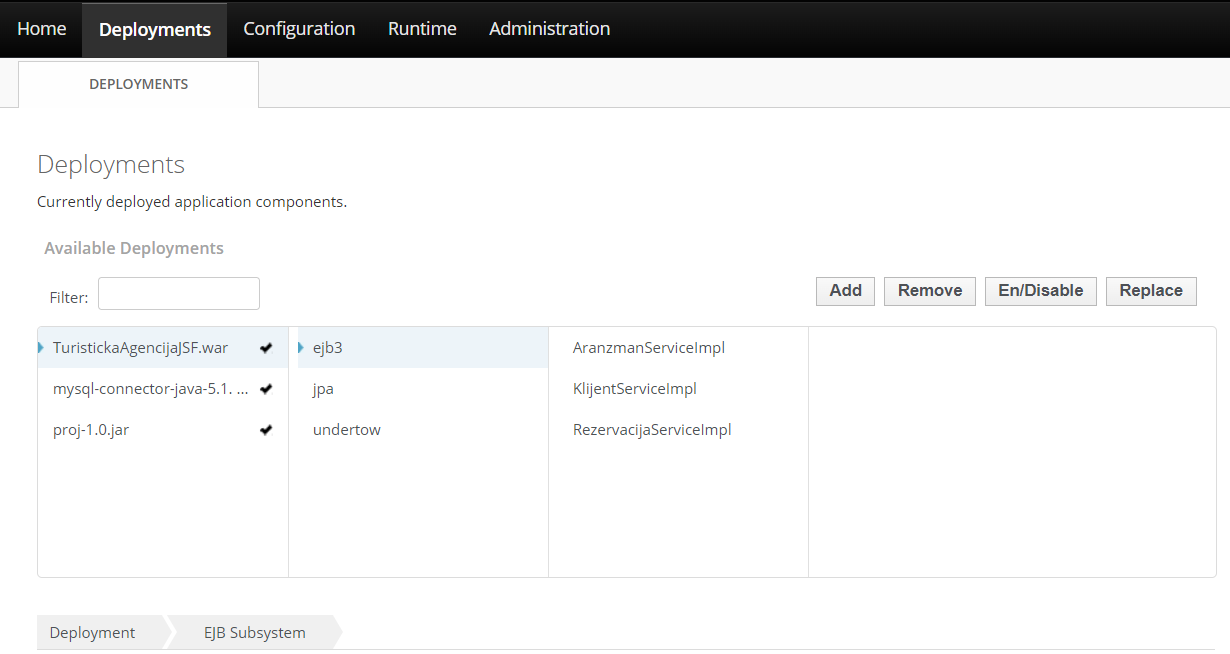
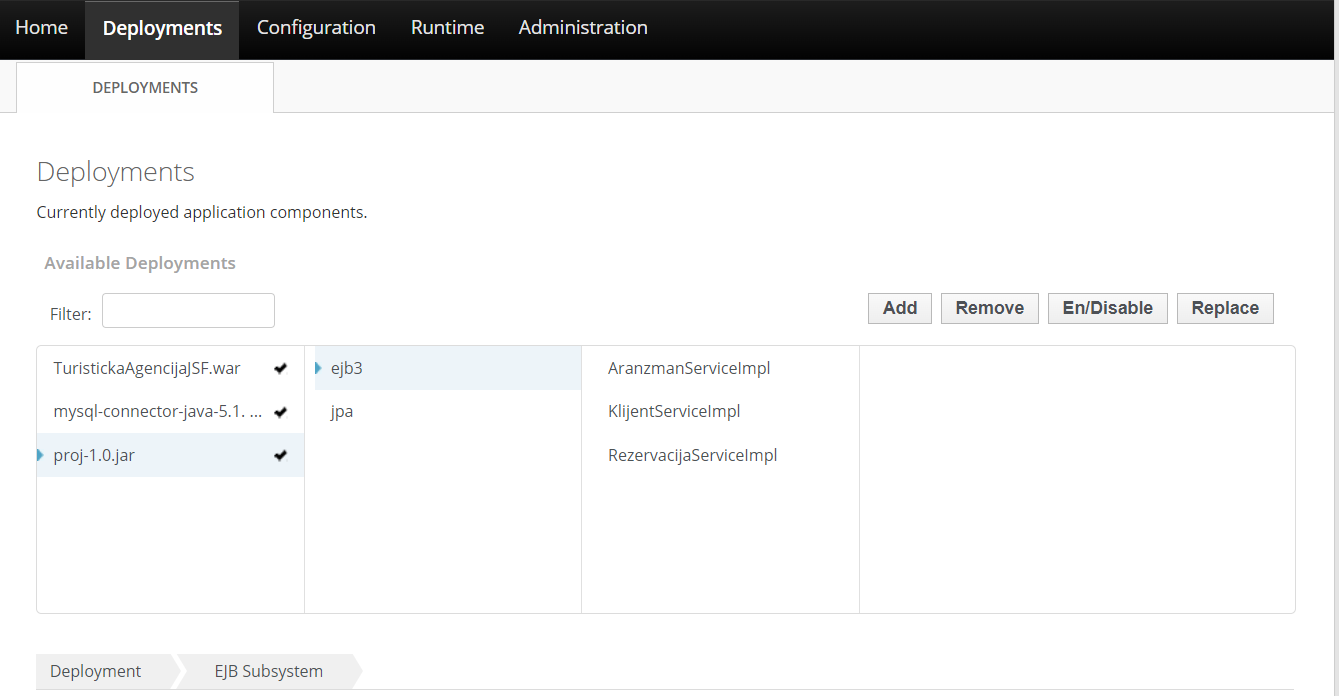
1. UML Klasni dijagram



1. Šema baze podataka



1. Konfiguracija deployment-a



1. Test cases

* Test funkcije za dodavanje novog klijenta

**Opis**: Testira se da li funkcija za dodavanje klijenata u bazu podataka radi adekvatno time što dodaje novog klijenta sa id-em 1000.

**Preduslovi**: Potrebno je da u bazi, u tabeli ‘klijenti’ ne postoji klijent sa id-em 1000.

**Koraci testa**:

1. Provera da li postoji klijent sa zadatom email adresom, kao i id-em.
2. Dodaje se novi klijent u bazu.
3. Provera da li je klijent uspešno dodat.
4. Prikaz podataka prethodno unetog klijenta.
5. Brisanje unetih podataka iz baze.

**Test podaci**: INSERT INTO `baza`.`klijenti` (`id`, `ime`, `prezime`, `datumRodjenja`, `brojTelefona`, `emailAdresa`) VALUES ('1000', 'Aleksa', 'Ilic', '2002-10-23 00:00:00', '060000000', 'aleksa@gmail.com');

**Očekivani rezultat**: Korišćene funkcije rade korektno i pozivanim redosledom ne uzrokuju nikakve promene u bazi podataka.

**Dobijeni rezultati**: Funkcije rade korektno i nema promena u bazi podataka.

**Status**: Uspešan.

* Test funkcije za kreiranje nove rezervacije

**Opis**: Testira se da li funkcija za dodavanje nove rezervacije u bazu podataka radi adekvatno time što dodaje novog klijenta sa id-em 1000, aranžman sa id-em 1000 i na kraju i samu rezervaciju između klijenta i aranžmana, takođe sa id-em 1000.

**Preduslovi**: Potrebno je da u bazi, u tabeli ‘klijenti’ ne postoji klijent sa id-em 1000, u tabeli ‘aranzmani’ ne postoji aranžman sa id-em 1000 i u tabeli ‘rezervacije’ ne postoji rezervacija sa id-em 1000.

**Koraci testa**:

1. Provera da li postoji klijent, aranžman i rezervacija sa id-em 1000.
2. Dodaju se novi klijent, aranžman i rezervacija.
3. Provera da li je rezervacija uspešno dodata.
4. Brisanje unetih podataka iz baze.

**Test podaci**:

INSERT INTO `baza`.`aranzmani` (`id`, `naziv`, `cena`, `brojSlobodnihMesta`, `brojMesta`, `polaznaDestinacija`, `vremePolaska`, `dolaznaDestinacija`, `vremeDolaska`, `info`) VALUES ('1000', 'Budva 2024', '150', '55', '55', 'Nis', '2024-06-15 00:00:00', 'Budva', '2024-06-25 00:00:00', 'Testiranje opisa aranzmana Budva 2024.');

INSERT INTO `baza`.`klijenti` (`id`, `ime`, `prezime`, `datumRodjenja`, `brojTelefona`, `emailAdresa`) VALUES ('1000', 'Petar', 'Peric', '1988-12-16 00:00:00', '065252522', 'petarperic@gmail.com');

INSERT INTO `baza`.`rezervacije` (`id`, `klijent\_id`, `aranzman\_id`, `vremeRezervacije`) VALUES ('1000', '1000', '1000', '2024-01-01 00:00:00');

**Očekivani rezultat**: Korišćene funkcije rade korektno i pozivanim redosledom ne uzrokuju nikakve promene u bazi podataka.

**Dobijeni rezultati**: Funkcije rade korektno i nema promena u bazi podataka.

**Status**: Uspešan.

* Test funkcije za prikaz aranžmana čije je vreme polaska u narednih N dana

**Opis**: Testira da li funkcija za prikazivanje aranžmana radi adekvatno, tako da prikazuje samo aranžmane čije je vreme polaska u narednih N=31 dana

**Preduslovi**: Potrebno je da u bazi, u tabeli ‘aranzmani’ ne postoje aranžmani sa id-evima 1000,1001 i 1002.

**Koraci testa**:

1. Provera da li postoje aranžmani sa id-evima 1000, 1001 i 1002.
2. Dodaju se novi aranžmani u bazu.
3. Prikazuju se aranžmani koji ispunjavaju uslov(vreme polaska u narednih 31 dan).
4. Brisanje dodatih aranžmana iz baze.

**Test podaci**:

INSERT INTO `baza`.`aranzmani` (`id`, `naziv`, `cena`, `brojSlobodnihMesta`, `brojMesta`, `polaznaDestinacija`, `vremePolaska`, `dolaznaDestinacija`, `vremeDolaska`, `info`) VALUES ('1000', 'Oslo Tura 1', '300', '20', '20', 'Novi Sad', '2024-01-29 00:00:00', 'Oslo', '2024-02-01 09:00:00', 'Opis 1');

INSERT INTO `baza`.`aranzmani` (`id`, `naziv`, `cena`, `brojSlobodnihMesta`, `brojMesta`, `polaznaDestinacija`, `vremePolaska`, `dolaznaDestinacija`, `vremeDolaska`, `info`) VALUES ('1001', 'Oslo Tura 2', '300', '20', '20', 'Subotica', '2024-01-19 00:00:00', 'Oslo', '2024-01-21 09:00:00', 'Opis 2');

INSERT INTO `baza`.`aranzmani` (`id`, `naziv`, `cena`, `brojSlobodnihMesta`, `brojMesta`, `polaznaDestinacija`, `vremePolaska`, `dolaznaDestinacija`, `vremeDolaska`, `info`) VALUES ('1002', 'Oslo Tura 3', '300', '20', '20', 'Novi Sad', '2024-02-15 00:00:00', 'Oslo', '2024-02-17 09:00:00', 'Opis 3');

**Očekivani rezultat**: Prikaz prva dva test aranžmana, jer je njihovo vreme polaska u narednih 31 dana.

**Dobijeni rezultati**: Uspešan prikaz prva dva test aranžmana.

**Status**: Uspešan.