
Rešenja zadataka

Git

Autori:
Goran Savić
Milan Segedinac

1. Zadatak

Repozitorijum se kreira na www.gitlab.com klikom na dugme New project. Na novoj stranici se unese ime projekta i klikne na dugme Create project.

Kloniranje repozitorijuma na lokalni računar (ako je korisničko ime petar i projekat se zove helloworld):

```
git clone https://github.com/petar/helloworld.git
```

Pozicionirati se u helloworld folder koji je nastao kloniranjem:

```
cd helloworld
```

Konfigurisati repozitorijum (ako na računaru nisu globalno konfigurisani svi Git repozitorijumi):

```
git config user.name "Petar Petrovic"  
git config user.email "petar@mail.com"
```

1. U Eclipse ići na kreiranje novog Java projekta (File -> New -> Java Project). Pri kreiranju projekta, za lokaciju ne treba izabrati *default* lokaciju u Workspace, nego izabrati folder helloworld koji predstavlja git repozitorijum.

Napraviti jedan paket u projektu i u njemu klasu Hello koja ima main metodu koja ispisuje HelloWorld:

```
System.out.println("HelloWorld");
```

2. Da se ne bi u repozitorijum postavljali fajlovi iz bin foldera, u folderu helloworld napraviti .gitignore fajl sa sadržajem
/bin/

Iz komandne linije, ako smo pozicionirani u helloworld folder:

```
git add .  
git commit -m "Dodavanje ispisa HelloWorld"
```

3. U Eclipse promeniti kod za ispis:
System.out.println("Petar");

4. Iz komandne linije, ako smo pozicionirani u helloworld folder:

```
git add .  
git commit -m "Izmena ispisa"
```

5. git log

6. Ako prvi commit sa ispisom HelloWorld ima heš vrednost koja počinje sa c19a915

```
git checkout c19a915
```

7. git branch novagrana
git checkout novagrana

Izmeniti klasu HelloWorld tako da ispis bude:

```
System.out.print("HeloWorld");
```

Dodavanje izmenjenog fajla na granu:

```
git add .  
git commit -m "Ispis koriscenjem print"
```

8. Povratak na glavnu granu

```
git checkout master
```

Ubacivanje izmene sa grane u glavnu granu

```
git merge novagrana
```

Git će prijaviti konflikt jer je ista linija koda menjana u verzijama koje spaja.

Konflikt rešavamo tako što u Eclipse u klasi obrišemo sve Git markere (linije koje počinju sa <<<<, ===== i >>>>). Ostavimo samo liniju koda koja ispisuje HelloWorld sa print.

Nakon rešavanja konflikta, postavimo sreden fajl na repozitorijum:

```
git add .  
git commit -m "Resen konflikt sa ispisom"
```

9. Postavimo izmene na udaljeni repozitorijum:

```
git push origin master
```

2. Zadatak

Vođa tima kreira projekat na www.gitlab.com (na dugme New project).

Vođa tima vrši kloniranje repozitorijuma na svoj lokalni računar.

```
git clone https://github.com/petar/imena.git
```

U folderu u kojem je repozitorijum kreira novi fajl imena.txt i postavlja ga na repozitorijum:

```
git add .  
git commit -m "Dodavanje praznog fajla sa imenima"
```

Vođa tima postavlja izmene sa lokalnog repozitorijuma na udaljeni repozitorijum na www.gitlab.com:

```
git push origin master
```

Svi ostali članovi kloniraju repozitorijum na svoj lokalni računar:

```
git clone https://github.com/petar/imena.git
```

Svi članovi tima (uključujući i vođu tima) unose svoje ime u prvu praznu liniju fajla imena.txt.

Svi članovi tima postavljaju izmenu u svoj lokalni repozitorijum:

```
git add .  
git commit -m "Unos imena Milan"
```

Svi članovi tima postavljaju izmene na udaljeni repozitorijum:

```
git push origin master
```

Samo će najbrži član tima uspeti da izvrši operaciju. Ostali će dobiti grešku, jer je u međuvremenu udaljeni repozitorijum dobio novi *commit* koji nije integrisan u lokalni repozitorijum. U tom slučaju moraju se preuzeti izmene sa udaljenog repozitorijuma i spojiti sa lokalnim izmenama:

```
git pull
```

Operacija pull će prijaviti konflikt, jer je izmena na udaljenom repozitorijuma izvršena u fajlu imena.txt u istoj liniji kao i lokalna izmena. Git će ubaciti markere u fajl, koje je potrebno obrisati i srediti fajl tako da ostanu u njemu i sva imena koja su dodali drugi članovi tima, kao i ime člana tima koji radi tom na lokalnom računaru.

Nakon toga je potrebno ubaciti izmene u lokalni repozitorijum:

```
git add .  
git commit -m "Resen konflikt"
```

Izmene se onda šalju u udaljeni repozitorijum:

```
git push origin master
```

Operacija će proći uspešno, samo ako u međuvremenu neki drugi član tima nije dodao novu verziju na udaljeni repozitorijum. Ako se to desilo, neophodno je ponoviti postupak preuzimanja izmena, rešavanja konflikta i slanja na udaljeni repozitorijum sve dok operacija ne uspe (tj. dok god se ne desi da od trenutka preuzimanja izmena sa udaljenog repozitorijuma do slanja novih izmena na udaljeni repozitorijum nije bilo postavljanja novih verzija od strane drugih članova tima).