

# Git – rad sa udaljenim repozitorijumom

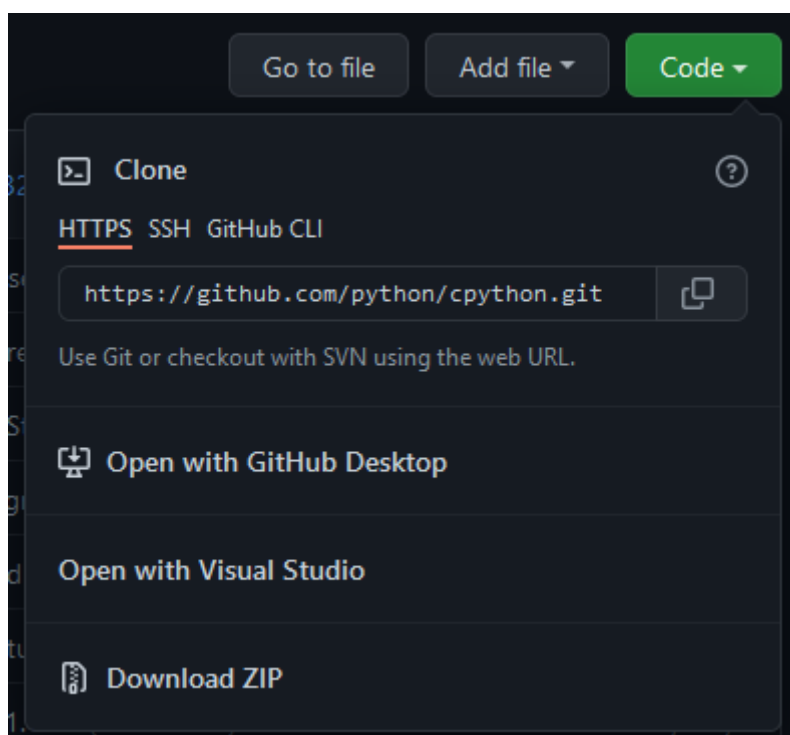
## Udaljeni repozitorijumi i udaljene grane

Naučili ste kako da radite sa gitom u lokalnu. Vrijeme je da naučite kako možete da razmjenjujete izmjene sa kolegama iz tima. Pored lokalnih repozitorijuma postoje i udaljeni repozitorijumi. Udaljeni repozitorijum je repozitorijum koji se nalazi na nekom drugom računaru ili serveru.

Udaljene repozitorijume ćemo praviti na nekoj od hosting platformi, kao što su *GitHub*/*GitLab*. *GitHub* repozitorijum je repozitorijum isti kao i tvoj lokalni, ali se nalazi na nekom *GitHub*-ovom serveru. Ove udaljene repozitorijume možete da podešavate tako da ih prilagodite svojim specifičnim potrebama. Na *GitHub*-u napravite novi prazan repozitorijum.

Da bi mogli da razmjenjujemo izmjene između lokalnog i udaljenog repozitorijuma, potrebno je da izvršimo određena podešavanja. Podešavanja uključuju dodavanje udaljenih repozitorijuma našem lokalnom repozitorijumu. Nakon dodavanja možemo slati i dobavljati izmjene sa udaljenog repozitorijuma.

Link do udaljenog repozitorijuma možete vidjeti tako što ćete pritisnuti na zeleno „Code“ dugme na *GitHub* stranici prikazano na slici ispod.



Možete da odaberete način na koji će se vršiti razmjena izmjena između repozitorijuma: preko HTTPS protokola ili putem SSH. Savijet je da ukoliko radite na svom računaru da koristite SSH. Ukoliko radite na računaru iz učionice, koristite HTTPS i kreirajte Personal Access Token koji ćete koristiti umjesto šifre. Kopirajte putanju do udaljenog repozitorijuma (putanja je oblika: [git@github.com:python/cpython.git](mailto:git@github.com:python/cpython.git) ili <https://github.com/python/cpython.git>). Uputstvo za podešavanje SSH je na ovom [tutorijalu](#), a Personal Access Token [ovdje](#).

Obično postoji jedan udaljeni repozitorijum po projektu i on se čseto zove `origin`. Da biste podesili udaljeni repozitorijum za vaš lokalni repozitorijum izvršite naredbu: `git remote add origin <url do udaljenog repozitorijuma>`. Praksa je da se glavni udaljeni repozitorijum zove `origin`. Za link do udaljenog repozitorijuma stavite link do praznog *GitLab*/*GitHub* repozitorijuma kojeg ste napravili ranije.

Da bi provjerili koji udaljeni repozitorijumi su dodati izvršite: `git remote -v`. Prodiskutujte sa asistentom rezultat izvršavanja ove naredbe.

Pored lokalnih grana postoje i udaljene grane. Udaljene grane se nalaze na udaljenim repozitorijumima. Više o granama u nastavku.

## Slanje izmjena

Kad ste podesili udaljeni repozitorijum, vrijeme je da na njega pošaljete sve što ste razvili u lokalnu (sve što se nalazi u lokalnom repozitorijumu). Da bi to odradili iskoristite komandu `push`. Ova komanda šalje sve izmjene sa lokalne grane na udaljenu granu. Šalju se samo nove izmjene koje ne postoje na udaljenoj grani. **Da biste izvršili komandu potrebno je da kažemo gitu na koji udaljeni repozitorijum šaljemo izmjene i na koju granu tog repozitorijuma.** Pošalji izmjene na udaljeni repozitorijum koji si ranije kreirao sa: `git push origin <ime udaljene grane>`. Obično ime udaljene grane odgovara imenu lokalne grane, mada to ne mora da bude slučaj. Ukoliko ime grane ne postoji na udaljenom repozitorijumu, nova grana sa tim imenom će biti kreirana.

Da ne biste svaki put kucao ime udaljenog repozitorijuma i udaljene grane možete podesiti podrazumijevanu udaljenu granu na koju će se slati izmjene sa tvoje lokalne grane. To radite sa `git push -u origin <ime udaljene grane>`. Kad ovo podesite, svaki idući put možete da šaljete izmjene samo sa `git push`.

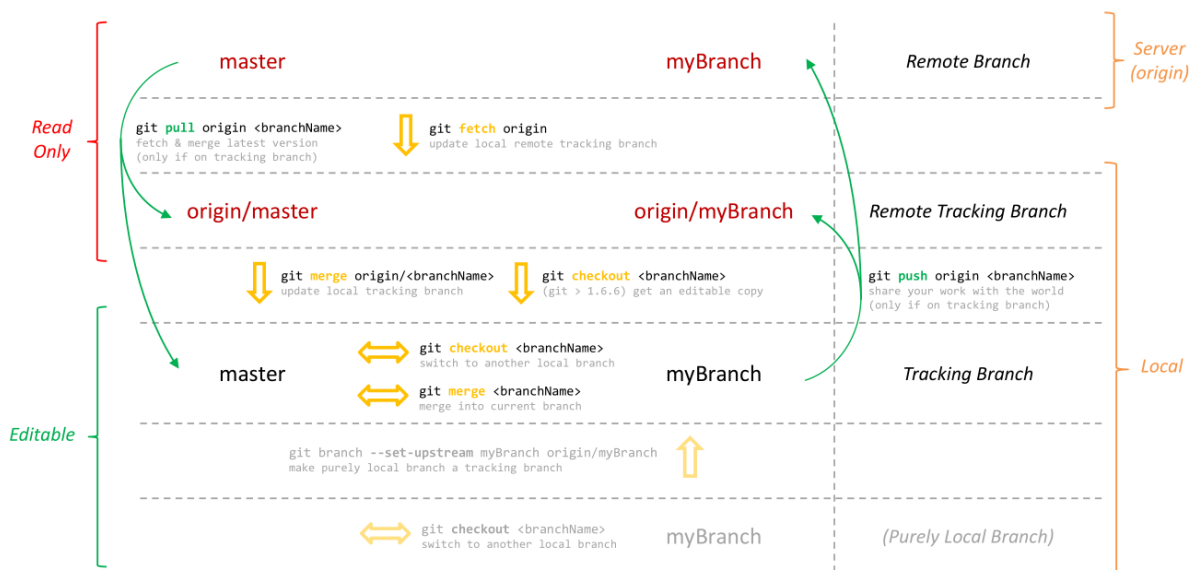
## Dobavljanje izmjena

Da biste dobavili izmjene sa udaljenog repozitorijuma koristimo naredbu `git fetch origin`. Ova naredba dovuče sve nove grane i izmjene na trenutnoj grani koje se nalaze na udaljenom repozitorijumu. Te izmjene su smještene u takozvanim *remote tracking* granama (granama koje prate udaljene grane). *Remote tracking* grane su oblika `origin/<ime grane>` i one su *read-only* (samo su za čitanje, ne mogu se slati izmjene na njih). Na *remote tracking* grane se dovuku izmjene sa udaljenih grana. Da bi te izmjene integrisali sa našim lokalnim granama radimo `git merge origin/<ime grane>`, isto kao kada spajamo dvije lokalne grane. Kako konflikti nastaju prilikom spajanja lokalnih grana, isto tako mogu nastati prilikom dobavljanja izmjena sa udaljene grane.

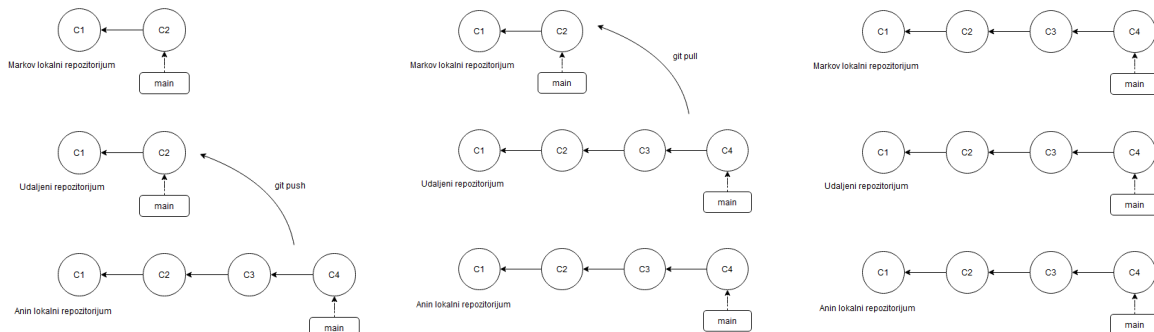
Kako se uvijek radi `git fetch origin`, pa `git merge origin/<ime grane>`, git ima skraćenicu za to: `git pull origin <ime grane>`. Savjet je da na početku rada koristite *fetch/merge* kombinaciju, dok se ne naviknete na tok rada, pa da vremenom pređete na pull zbog brzine. Slično kao i kod push komande, ako se podesi default-na push/pull grana, naredba se može pozivati samo sa `git pull`.

Na slici ispod su prikazani tipovi grana u gitu i komande koje koristimo da upravljamo njima. Detaljno analizirajte sliku sa asistentom i prodiskutujte sve vrste grana.

## Git branches for the rest of us



Na slikama ispod je prikazana razmjena izmjena. Miloš je napravio neke izmjene koje šalje na udaljeni repozitorijum. Kad su izmjene poslate na udaljeni repozitorijum, Marko može da ih preuzme u svoju lokalnu kopiju main-a i nastavi sa razvojem.



## Kloniranje udaljenog repozitorijuma

Postavlja se pitanje kako možemo da preuzmемо stanje udaljenog repozitorijuma kojeg nemamo lokalno na računaru. To radiš sa naredbom **clone** na idući način: `git clone <url do udaljenog repozitorijuma>`. Sa ovim će git preuzeti udaljeni repozitorijum, napraviti novi folder sa njegovim imenom i u njega će smjestiti sve fajlove i sve izmjene zabilježene u istoriji tog udaljenog repozitorijuma. Nakon clone-a možete nastaviti rad na repozitorijumu. Isprobajte ovu naredbu tako što ćeš klonirati idući repozitorijum: <https://github.com/SIIT-USI/git-clone-test>.

**Opciono:** Isprobajte clone naredbu još jednom tako što ćete klonirati repozitorijum koji je kreirao kolega iz klupe za izradu kalkulator zadatka.

**Opciono:** Otvorite kolegin projekat i napravi novu granu. Na toj grani napravite fajl pod nazivom README.md i kratko opišite šta program radi. Pošaljite novu granu na kolegin udaljeni repozitorijum.

*Opciono:* Nakon što ste to odradili na koleginoj grani, vratite se u svoj kalkulator projekat i u njemu dodajte README.md fajl sa opisom šta program radi i ko ga je napravio. Pošaljite sve izmjene na udaljeni repozitorijum.

## Vježbanje rada sa granama

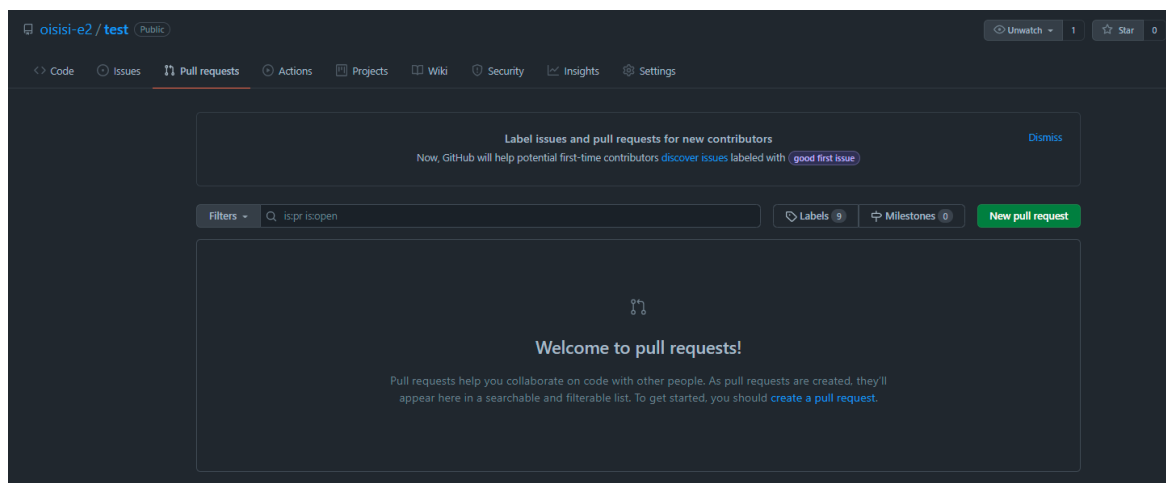
Da biste izvježbali rad sa granama, kako lokalnim, tako i udaljenim, pređite vježbe sa ovog sajta <https://learngitbranching.js.org/> vezane za rad sa granama. Za sad možete preskočiti vježbe vezane za rebase i interaktivni rebase.

## Pull request

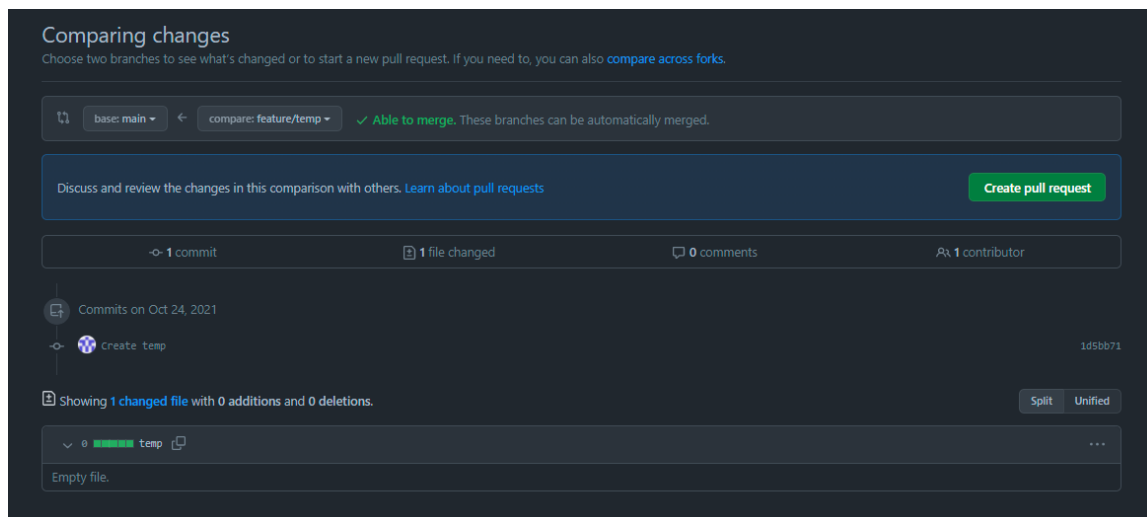
*Pull request* (PR) je proces putem kog se novi kod integriše na udaljeni repozitorijum. Ideja iza PR je da se kod prije spajanja sa repozitorijumom provjeri za greške, testira kako bi se povećao kvalitet koda i projekta. PR je kao neka kontrola merge naredbe, gdje neku granu umjesto da spojite lokalno, vi napravite PR pomoću kog ćete je spojiti. Tako se prije spajanja lako može provjeriti kod za greške. Spajanjem PR-a se i same grane spojaaju.

Napravite svoj prvi PR tako što ćete napraviti PR za granu koju ste napravili u kalkulator projektu na udaljenom repozitorijumu vašeg kolege. U koracima ispod je opisano kako možete napraviti novi PR.

PR možemo napraviti putem GitHub-a. Otvorimo Pull request tab repozitorijuma u kojem želimo da otvorite PR i pritisnimo na “New pull request”.

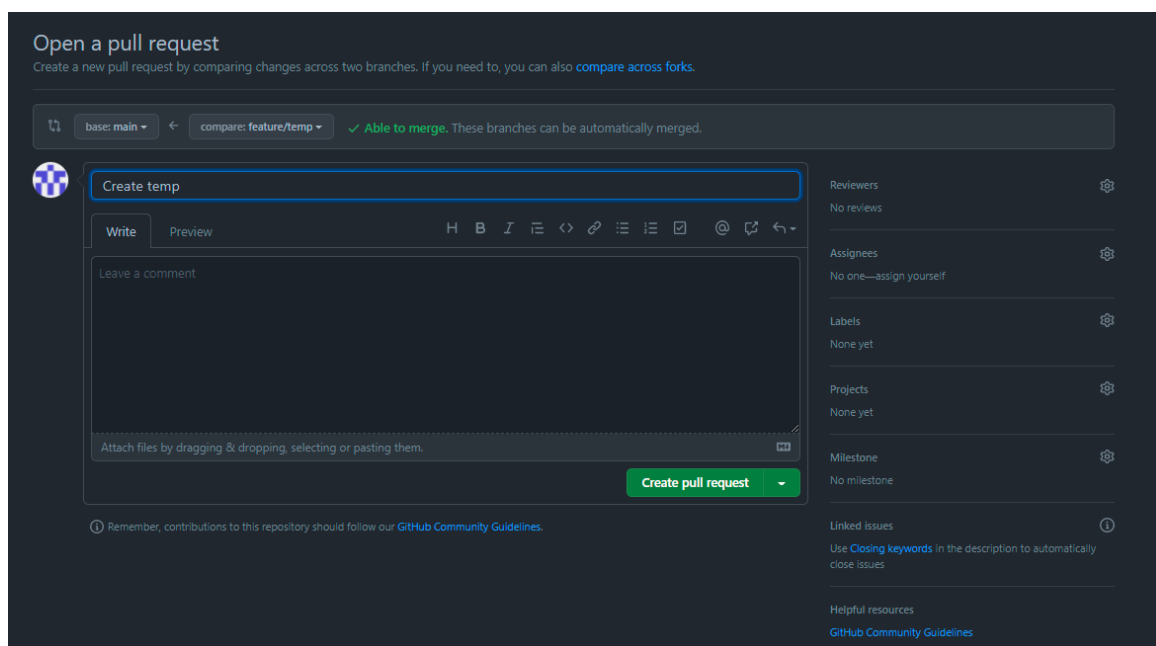


Idući korak je prikazan na slici ispod. Potrebno je da odaberemo koju granu želimo da spojimo na koju granu. Ispod možemo da pogledamo pregled commit-ova i pregled izmjena.

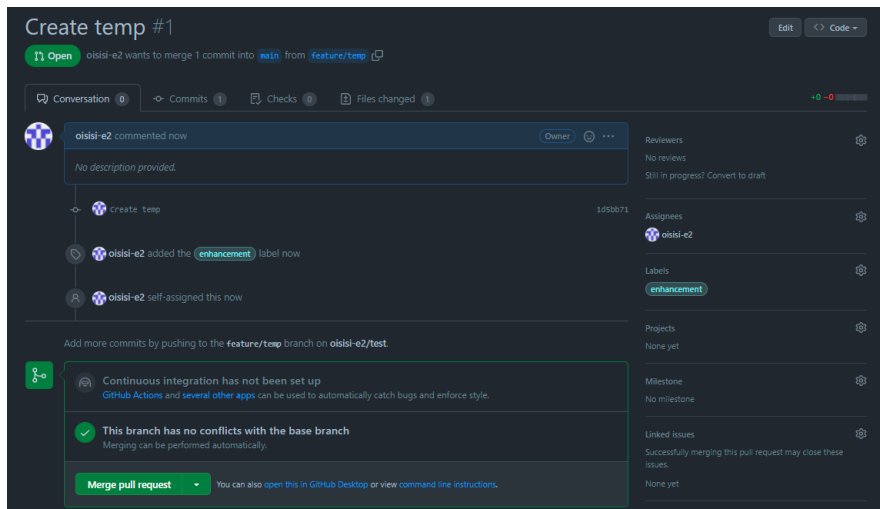


Nakon pritiska na dugme “Create pull request” otvori se prozor sa slike ispod. Tu možemo da podesimo naziv PR-a kao i opis – **šta smo izmijenili i zašto**. Dalje, moguće je podesiti “Reviewers” grupu ljudi koji treba da odrade review novih izmjene. “Assignees” grupa ljudi koja je zadužena za PR – ljudi koju su pravili izmjene (obično je to osoba koja pravi PR).

Kad sve popunimo, pritisemo na “Create pull request” da bi napravili PR.

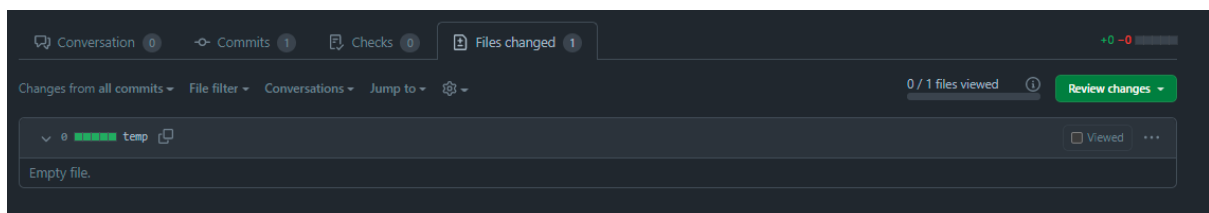


Otvoren PR je prikazan na slici ispod.

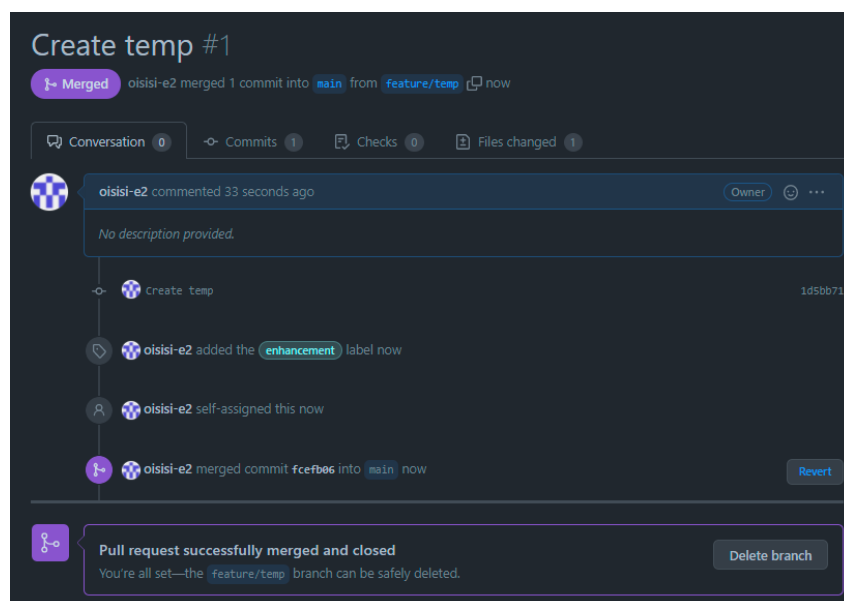


Pod tabom “Files changed” možemo da odradimo review PR-a. Review podrazumijeva da prođemo kroz sve izmjene, testiramo sve, uvjerimo se da radi kako je zamišljeno. Usput ostavljamo komentare na mjestima koje treba izmijeniti ili ako želimo da prodiskutujemo o načinu implementacije sa kolegom koji je napravio PR, i slično. Na dugme „Review changes“ kreiramo review. Možemo da:

- Odobrimo izmjene i dozvolimo kolegi da spoji izmjene u granu u koju želi
- samo ostavimo komentar bez eksplicitnog odobrenja ili
- možemo da zatražimo izmjene i onemogućimo spajanje dok se ne dodaju tražene izmjene.



Kad dobijemo sve dozvole za merge od svih review-era možemo da odradimo merge i spojimo svoje izmjene. Spojen PR je prikazan ispod.



Zatražite review od svog kolege. Kolega će spojiti tvoj PR nakon što ga odobri.

## Dodatna konfiguracija

Kako bi ubrzali rad sa gitom preporuka je da napravite git aliase. Alias je skraćeno ime za pisanje čestih git komandi. Pogledajte [dokumentaciju](#) za detalje i primjere.

## Literatura i inspiracija

- <https://git-scm.com/book/en/v2> - Pro git knjiga (git Biblija), preporučuje se da pročitate prva 3 poglavlja
- <https://missing.csail.mit.edu/2020/version-control/> - video i tekstualni materijali MIT-a o git-u
- <http://www.igordejanovic.net/courses/tech/git/> - odlični slajdovi koje je napravio profesor Dejanović, osnovni i napredni pojmovi su detaljno objašnjeni. Dodatno, sve je na srpskom.