

## Prva laboratorijska vežba iz predmeta Principi softverskog inženjerstva

### Prvi deo: Kreiranje repozitorijuma lokalno

1. Klonirati repozitorijum lokalno. Pri kloniranju Git Credential Manager će tražiti username i password. Username je prvi deo studentskog mejla (pre @), a šifra se generiše na Gerrit-u (ući u Settings za Vaš nalog, i u sekciji HTTP Credentials generisati novu šifru i zatim nju prekopirati u Git Credential Manager).
2. Dodatno, u okviru Git repozitorijuma je potrebno podesiti ime i email. U Git Bash-u u okviru repozitorijuma izvršiti komande **git config user.email <studentski\_mejl>** i **git config user.name <ime>**.
3. Po dogovoru, jedan član tima treba da ubaci fajlove iz prethodnih faza na repozitorijum i drugi članovi tima treba da odobre to ubacivanje, odobravanjem Pull Request-a (PR).
4. Preostali članovi tima povlače promene sa udaljenog repozitorijuma

### Drugi deo: Rad sa Pull Request-ovima

5. Svaki član tima treba da kreira u direktorijumu fajl <Ime\_člana\_tima>.txt sa nasumičnim sadržajem
6. Napraviti commit sa ovim fajlom i poslati ga na udaljeni repozitorijum (**git push origin HEAD:refs/for/master**).
7. S obzirom na to da se je ovo odradio svaki član tima, kreiraće se PR za svakog člana tima. Povuci lokalno izmene Pull Request-a na sledeći način - ući u izmenu nekog drugog člana tima koja još uvek nije prihvaćena, i u gornjem desnom uglu kliknuti na tri tačke zatim *Download patch*. Prekopirati komandu iz sekcije *Branch* i izvršiti je u lokalnom repozitorijumu. Izvršavanje te komande će rezultovati pravljenjem nove grane na Vašem lokalnom repozitorijumu, na kojoj se mogu videti sve izmene koje su napravljene u okviru datog commit-a, i eventualno testirati. Ta grana se posle *code review*-a može obrisati sa lokalnog repozitorijuma. Ostaviti komentare na PR (uz to ostaviti ocenu koja nije +2).
8. Pogledati komentare koji su drugi ostavili na Vašem PR i odgovoriti na komentare. Ukoliko se slažete, napravite promenu lokalno (**git commit --amend**), u skladu sa komentarom i ažurirati PR (**git push origin HEAD:refs/for/master**).
9. Kada vidite da su i drugi Pull Request-ovi ažurirani u skladu sa Vašim komentarom, odobriti ga sa ocenom +2.
10. Kada su zadovoljeni svi uslovi za odobravanje PR-a, spojiti ga sa master granom.

### Treći deo: Razrešavanje Merge konflikta

11. Napraviti fajl index.html sa sadržajem datim na materijalima. Jedan član tima po dogovoru treba da ubaci na repozitorijum i drugi članovi tima treba da odobre to ubacivanje, odobravanjem Pull Request-a.
12. Svaki član tima treba da promeni sliku koja se nalazi u html stranici i da napravi PR od toga.
13. Od jednog člana tima je potrebno odobriti Pull Request i spojiti ga sa master granom.
14. Drugi članovi tima treba da povuku promene koju je napravio prvi na master grani (**git pull --rebase**).
15. Merge konflikt razrešiti lokalno u korist master grane, a zatim gurnuti promene nazad na udaljeni repozitorijum (u commit poruci ne menjati Change-Id).

### Četvrti deo: Ignorisanje fajlova

16. Jedan član tima u dogovoru sa ostalima treba da napravi .gitignore fajl. U njega je potrebno staviti bilo šta što ne želite da vaš repozitorijum prati, a da će se fizički nalaziti u njemu (konfiguracioni fajlovi vezani za okruženje itd.).
17. Za potrebe ovog primera, napisati u .gitignore fajlu da se ignoriše fajl ignore.txt i okačiti taj fajl na repozitorijum tako da drugi mogu da ga preuzmu.
18. Kada se preuzme .gitignore fajl, kreirati fajl ignore.txt.
19. Odraditi komandu git status. Da li je git zapazio kreiranje ignore.txt?
20. Preuzeti neki koristan šablon .gitignore fajla sa interneta.