

GitHub: Teoretični del naloge

AleksDuhanaJ

September 23, 2024

1 Uvod

GitHub je spletna platforma za gostovanje in razvoj programske opreme, ki temelji na sistemu za nadzor različic Git. Omogoča razvijalcem, da učinkovito sodelujejo na projektih, delijo kodo, sledijo spremembam in upravljajo različne različice programov. GitHub ima ključno vlogo pri odprtokodnih projektih, pa tudi pri komercialnem razvoju programske opreme [1].

2 Kaj je GitHub in kako deluje?

GitHub je gostujoča storitev za repozitorije Git, ki omogoča razvijalcem, da shranjujejo in upravljajo svoje projekte. Git je distribuiran sistem za nadzor različic (DVCS), ki vsakemu razvijalcu omogoča, da ima celotno zgodovino projekta in dostop do vseh datotek lokalno [2]. Glavne značilnosti GitHub-a vključujejo:

- **Repozitoriji:** Repozitorij ali Git projekt vsebuje vse datoteke in mape povezane z določenim projektom, skupaj z zgodovino revizij. GitHub omogoča, da so ti repozitoriji javni ali zasebni [2].
- **Veje (branches):** Veje omogočajo razvijalcem, da delajo na ločenih verzijah kode brez vpliva na glavno vejo. To je uporabno pri razvoju novih funkcionalnosti ali odpravljanju napak [3].
- **Pull requests:** Z uporabo pull requestov lahko razvijalci predlagajo spremembe v projektu, ki jih drugi člani ekipe pregledajo in združijo (merge) v glavno vejo [1].
- **Integracija z drugimi orodji:** GitHub ponuja široko paleto integracij z drugimi orodji za razvoj programske opreme, kot so CI/CD sistemi, projektni menedžment in še več [4].

3 Uporaba GitHub-a v ekipnem razvoju

GitHub se pogosto uporablja v ekipah, ki razvijajo spletne aplikacije, igre ali namizne programe. Ekipa lahko uporablja GitHub na naslednje načine:

- **Sledenje spremembam:** Z uporabo Git-a lahko vsak član ekipe vidi, katere spremembe so bile narejene, kdo jih je naredil in zakaj [2].
- **Delo z vejami:** Razvijalci lahko delajo na lastnih vejah in nato združijo svoje spremembe, ko so pripravljeni [3].
- **Pregled kode (Code review):** GitHub omogoča pregled kode prek pull requestov, kjer lahko drugi člani ekipe podajo povratne informacije [1].
- **Avtomatizacija:** GitHub Actions omogoča avtomatizacijo nalog, kot so testiranje in razhroščevanje [4].

4 Kaj je *merge* v okolju GitHub in kako pripomore pri ekipnem delu?

Merge ali združevanje je proces kombiniranja sprememb iz ene veje v drugo. V GitHub-u se *merge* najpogosteje uporablja za vključitev sprememb iz razvojne veje (*feature branch*) v glavno vejo (*main branch*). To je ključno za ekipno delo, saj omogoča, da razvijalci delajo na različnih delih projekta neodvisno, nato pa svoje spremembe združijo, ko so pripravljene [3].

Proces združevanja vključuje:

- Ustvarjanje pull requesta, ki opozori ekipo na pripravljene spremembe.
- Pregled sprememb in povratne informacije od drugih članov ekipe.
- Združitev sprememb v glavno vejo, ko je ekipa zadovoljna z izvedenimi popravki.

Merge omogoča, da ekipa dela vzporedno na različnih funkcionalnostih in hkrati ohranja stabilnost glavne kode. Prav tako preprečuje konflikte in izboljša sledljivost sprememb [2].

5 Zaključek

GitHub je nepogrešljivo orodje za razvoj programske opreme, saj omogoča učinkovito sodelovanje med razvijalci, sledenje spremembam in upravljanje različic. S svojimi funkcionalnostmi, kot so veje, pull requesti in merge, poenostavi delo ekipam, ki razvijajo različne vrste programov.

References

- [1] GitHub Docs. *Get started with GitHub documentation*. Dostopno na: <https://docs.github.com/en/get-started>.
- [2] GitHub Docs. *About Git*. Dostopno na: <https://docs.github.com/en/get-started/using-git/about-git>.
- [3] GitHub Docs. *Using Git*. Dostopno na: <https://docs.github.com/en/get-started/using-git>.
- [4] GitHub Docs. *Exploring integrations*. Dostopno na: <https://docs.github.com/en/rest/guides>.