

**Sistemska administracija**

**Verzioniranje**

**Avtor: Tine Jereb**

**Maribor, marec 28 2024**

KAZALO

[1 UVOD 3](#_Toc162547810)

[2 USTVARJANJE REPOZITORIJA 4](#_Toc162547811)

[3 COMMIT 6](#_Toc162547812)

[4 NOVA VEJA 7](#_Toc162547813)

[5 MERGE 9](#_Toc162547814)

[6 LOGS 10](#_Toc162547815)

[6.1 Git log 10](#_Toc162547816)

[6.2 Git reflog 11](#_Toc162547817)

[7 POVEZAVA NA REPOZITORIJ 12](#_Toc162547818)

KAZALO SLIK

[Slika 1: Ustvarjanje repozitorija 4](#_Toc162545549)

[Slika 2: Prvi commit 5](#_Toc162545550)

[Slika 3: Dodajanje datotek na GitHub 6](#_Toc162545551)

[Slika 4: Commit in push 7](#_Toc162545552)

[Slika 5: Commit in push 7](#_Toc162545553)

[Slika 6: Ustvarjanje in premik v novo vejo 7](#_Toc162545554)

[Slika 7: Commit na novo vejo 8](#_Toc162545555)

[Slika 8: Commit na novo vejo 8](#_Toc162545556)

[Slika 9: Merge 9](#_Toc162545557)

[Slika 10: git log 10](#_Toc162545558)

[Slika 11: git reflog 11](#_Toc162545559)

# UVOD

Pri tej nalogi sem se naučil uporabljati osnovne funkcija Git-a in GitHub-a. Začel sem z ustvarjanjem novega lokalnega repozitorija in ga povezal z novim GitHub repozitorijem. V repozitoriju sem naredil 10 različnih commitov, kjer sem sprotoma dodajal nove funkcionalnosti pri nalogi za učno enoto Razvoj aplikacij za internet, kjer smo naredili preprosto spletno stran za objavljanje in urejanje novic v jeziku php. Kot izhodišče smo dobili zgled iz vaj, katerega smo nadgradili. Obvezni del naloge sem objavljal na glavni veji, za neobvezni del pa sem ustvaril novo vejo. Po končani nalogi sem združil glavno vejo z novo vejo. Za celoten postopek sem si pomagal s spletom. Na začetku sem se še malo lovil, a ko sem dojel delovanje Gita in GitHuba, dokončevanje naloge ni bilo več problematično.

# USTVARJANJE REPOZITORIJA

Na GitHub-u sem ustvaril nov prazen repozitorij z imenom “SA\_Verzioniranje”. Nato sem na računalniku izbral mapo, kjer bom delal projekt in v njej odprl Git Bash. V CLI sem prilepiv ukaze, ki mi jih je pripravil GitHub za povezavo na repozitorij.

A screenshot of a computer

Description automatically generated

Slika : Ustvarjanje repozitorija

A screenshot of a computer program

Description automatically generated

Slika : Prvi commit

# COMMIT

Projekt sem začel s pripravi zgleda, ki smo ga dobili za nadgradnjo. Te datoteke sem porinil na “main” vejo.

A screen shot of a computer

Description automatically generated

Slika : Dodajanje datotek na GitHub

Med commit-i sem pozabil ugasniti Word datoteko za poročilo, zaradi katerega mi ni omogočilo posodabljanje na GitHub.

A screenshot of a computer program

Description automatically generated

Slika : Prva težava pri commit

A screen shot of a computer program

Description automatically generated

Slika : Commit in push

# NOVA VEJA

Po dokončanem obveznem delu naloge, sem ustvaril novo vejo “Neobvezne funkcionalnosti”. Na to vejo sem objavljal neobvezne dele naloge.

A black background with yellow text

Description automatically generated

Slika : Ustvarjanje in premik v novo vejo

Pri push-u na novo vejo je bilo potrebno uporabiti poseben ukaz, katerega mi je CLI sam priporočal.

A computer screen shot of text

Description automatically generated

Slika : Prvi commit na novo vejo

A screen shot of a computer program

Description automatically generated

Slika : Commit na novo vejo

# MERGE

Po končani celotni nalogi, sem združil glavno vejo z novo. Pri prvem poizkusu sem bil nepozoren in narobe napisal ime druge veje. Za to napako sem potreboval nekaj časa, da sem jo videl in odpravil. Na srečo nisem imel konfliktov in ju je združilo brez kakšnih problemov.

A computer screen with colorful text

Description automatically generated

Slika : Merge

# LOGS

## Git log

V prikazanem izgodu vidimo prikaz zgodovine commit-ov v repozitorij. Vsaka vrstica v zapisu predstavlja en commit in vsebuje naslednje informacije:

* Hash – je unimatna koda, ki identificira posamezen commit.
* Avtor – ime in e-pošto uporabnika, ki je naredil commit.
* Datum – datum in čas, kdaj je bil commit izveden.
* Sporočilo – kratek opis sprememb, ki jih commit prinaša.

Git log omogoča spremljanje celotne zgodovine repozitorija. Prikaže kdaj so commit-i narejeni, kdo in kdaj jih je naredil in kakšne spremembe je prinesel. To je pomembno za sledenje in razvoja projekta.

A screen shot of a computer program

Description automatically generated

Slika : git log

## Git reflog

Git reflog prikazuje dnevnik referenc oz. zgodovino sprememb glave repozitorija in ostalih referenc. Ne prikazuje samo zgodovine commit-ov, ampak tudi zgodovino premikanja po projektni devesni zgradbi.

* Prikazuje vse nedavne dejavnosti v repozitoriju.
* Vsaka vrstica predstavlja premik reference in vsebuje hash.
* Omogoča vračanje na prejšnja stanja repozitorija.
* Izpis je krajši kot pri git log, saj ne prikazuje vseh informacij.

A black screen with green and yellow text

Description automatically generated

Slika : git reflog

# POVEZAVA NA REPOZITORIJ

https://github.com/TineJereb/SA\_Verzioniranje