

2024_jan_02

Osnove rabe oddaljenih repozitorijev kode, Git in BlueJ

pojmi:

korenski imenik je običajno korenski imenik datotečnega sistema; Windows loči datotečne sistem- vsak nosilec lahko (če ga ne vpne) predstavlja lasten datotečni sistem: vsak datotečni sistem ima lasten korenski imenik pri 'ostalih' imamo navadno en sam koren, vsi nosilci pa so vpeti in so predstavljeni kot podimeniki :

xNix sistemi : /

windows : c:\, d:\, ...

domači imenik je tisti v katerem pristanete, ko se prijavite v nek sistem; je lasten uporabniku;

windows: kot zažene 'cmd' vam okno odpre v tem imeniku

xNix: ~

windows: zapisan v %userprofile%, powershell pozna ~

delovni imenik (working directory/folder) je tisti v katerem snek trenutek nekaj delate običajno je delovni imenik tudi trenutni imenik (current directory), ni pa to nujno

prog. oprema za uporabo pri izvedbi:

- Git For Windows : <https://github.com/git-for-windows/git/releases/> (<https://gitforwindows.org/>)
- BlueJ : <https://www.bluej.org>

opomba:

xNix : verjetno boste za namestitev uporabili OS-u lasten upravljelec paketov (npr. apk: apk add git)

navodila v nadaljevanju bodo specifična za običajno windows lupino (cmd), za xNix lupine dodajamo v opombah

1. Naloga Osnovo spoznavanje

V domačem imeniku ustvarite nov delovni imenik z imenom naloga1. Naj postane ta imenik vaš trenutni imenik. Uporabite brskalnik (Firefox, Opera, PaleMoon, Edge, ...), naslovite https://gitlab.vegova.si/serhio.w02/g3_2023_24 .

- ali opazite, da na spisku dobite več datotek, nekaj jih je celo v lastnih mapah ?
- ali si lahko ogledate vsebino posamezne datoteke, tako da na spisku kliknete nanjo ?
- vsebina katere izmed datotek s spiska je prikazana pod spiskom datotek ?

Oglejte si URI v naslovni vrstici brskalnika; serhio.w02 področje/prostor, ki pripada enemu uporabniku git(laba), navadno je to kar njegovo uporabniško ime. g3_2023_24 je ime enega repozitorija/področja v lasti uporabnika, ki navadno pripada enemu projektu. Vse javne repozitorije uporabnika lahko dosežete enostavno tako, da iz URi odstranite ime projekta

- kateri so javni repozitoriji uporabnika serhio.w02 ?

Recimo, da se vrnemo nazaj na https://gitlab.vegova.si/serhio.w02/g3_2023_24. Desno je velik moder gumb za napisom 'Code'. S klikom nanj dobite 3 vrste izbir; spodaj so prenosi v obliki ene datoteka(zip, tar,...), nad tem možnost direktnega 'odpiranja' z enim od maloštevilnih IDE, na vrhu pa je naslov git v obliki git in spletnega naslova (NASLOV!).

2024_jan_02

Osnove rabe oddaljenih repozitorijev kode, Git in BlueJ

e) kateri IDE so s strani gitlab Vegove podprti?

Posamezno datoteko prenesete tako, da izberete datoteko (kliknete na ime) in nato poiščete povezavo/gumb na prenos datoteke. V splošnem redko potrebujemo mehanizem prenosa posamezne datoteke, ker so te običajno v kodi povezane oz. odvisne med seboj. Tipično prenašamo celoten repozitorij. Celotno strukturo in vsebino lahko prenesemo z enkratnim prenosom v obliki npr. zip datoteke. Po prenosu jo moramo ročno razpakirati. Celotno operacijo prenosa brez dodatnega posega zna opraviti klient. Za ta primer bomo uporabili interaktivni odjemalec git, inteligentno poimenovanem: git.

f) preverite, če je git odjemalec dosegljiv. V poročilo naj gre rezultat izvedbe : ...\\naloga1>git -v .

V primeru, da git ni dosegljiv imete dve možnosti, namestiti ga ali izpustiti nadaljevanje.

če je git dosegljiv, izvedete :

```
....\\naloga1>git clone https://gitlab.vegova.si/serhio.w02/g3\_2023\_24.git
```

(NASLOV! - dobite na gumbu 'Code' pod 'Clone with HTTPS')

po izvedbi ukaza lahko ugotovite, da se je v vašem delovnem imeniku pojavila mapa g3_2023_24, v delovni mapo izvedite

```
....\\naloga1>tree /F
```

in rezultat ukaza dodajte v poročilo. xNix : uporabite tree, če ga nimate nameščenega kot poseben paket bo dovolj tudi rezultat ukazov ls -alR ali du iz delovnega imenika

g) prenešana vsebina je BlueJ paket. Preverite, če lahko projekt odprete iz mape g3_2023_24.

Odgovor podajte v poročilo skupaj s tem, ali se je v BlueJ urejevalniku aktivirala opcija Teamwork (oz. so v Tools->Team omogočene več kot 2 možnosti?)

h) vsebine repozitorija, ki ste jih prenesli, so v bistvu 'trash projekt', oz. rezultat parih testiranj, brez dodatnega posega z vsaj nekaj vizualizacije deluje samo tisti Molecule ... Preskus ni obvezen.

i) ah, da, za vsak slučaj poskusite še enkrat prenesti oddaljen repo z git clone na isto lokacijo:

```
....\\naloga1>git clone https://gitlab.vegova.si/serhio.w02/g3\_2023\_24.git
```

verjetno ne bo šlo; clone vedno hoče narediti NOV lokalni repozitorij; pa dajmo, odstranimo ga s celotno vsebino vred:

```
....\\naloga1>rd /S g3_* (xNix: rm -r g3_*) in nato ponovimo
```

```
....\\naloga1>git clone https://gitlab.vegova.si/serhio.w02/g3_2023_24.git
```

! -----

! - git interaktivno omogoča hiter prenos repozitorija

! - BlueJ ne zazna avtomatično git projekta in se tako ne poveže z njim

! - pred nadaljevanjem:

! odjemalec v BlueJ sicer deluje, vendar ne podpira ravno vseh funkcij git-a

! odstranite še enkrat celoten folder g3_2023_24 (nov bo drugje, pa da s tem ne bo zmede)

2024_jan_02

Osnove rabe oddaljenih repozitorijev kode, Git in BlueJ

2. Naloga GitLab, Git, Teamwork in BlueJ

----- Repozitorij z vsebinami obstaja ----- ::

Odprite BlueJ. Če se slučajno odpre kakšen projekt, ga enostavno zaprite. Uporabite;

Tools -> Team -> Checkout Project

V TeamSettings dialogu izpolnite : Location URI naj vsebuje (NASLOV!) s protokolom https (glej git clone), branch pustite prazen, Personal: vpišite prva tri polja, e-mail format naj bo pravi, veljavnost vpisanih podatkov je nerelevantna, dokler za v oddaljenem repozitoriju operacije ni zahtevana avtentifikacija. Pravilnost(ustreznost) in dosegljivos URI preverite s Check Connection.

Po kliku na OK najprej določite: ime projekta naj bo g3_2023_24, lokacija delovna mapa (....\naloga1) in izvedite.

Test prepisa vsebine z oddaljenega (checkout) izvedete z :\naloga1>tree /F rezultat mora biti iz iste lokacije enak kot c predhodnih primerih. Če ne, ponavljate, dokler operacija ne uspe.

Opomba: če za ime mape izberete karkoli drugega kot g3_2023_24, bo pač ime mape drugačno oz. ime projekta drugačno

Na levi strani osnovnega okna BlueJ okolja, kjer imate vertikalno gumbe 'New class', puščica, 'Compile' se delu Teamwork po uspešnem prenosu in kreiranju lokalnega repozitorija pojavijo aktivirani gumbi Update - Commit/Push - Status. Če jih ni, poiščite mali trikotniček nekje spodaj in ga kliknite. Hkrati imate v Tools -> Team omogočene še ostale možnosti.

Pripravite zaslonsko sliko osnovnega BlueJ okna: pri prikazanih gumbih Teamwork na levi strani izvedite Project->Save As z odprtim dialogom Save As zajemite zaslonsko sliko in jo dodajte poročilu za to nalogo.

..... -! - preverite datotečni sistem

..... -! - lahko izvedete test prevajanja in izvajanja katerega od prenesenih primerov

! -----

! Opisan postopek je relevanten zgolj za bralen dostop, vsi pisalni dostopi zahtevajo

! avtentifikacijo; torej:

! : realen e-poštni naslov (:), predvsem pa uporabniško ime in geslo, ter jasno v skladu s tem

! repozitorij, kjer imate pravice za pisanje:

! - repozitorij, ki je lastniško vaš,

! - repozitorij, kjer vam je dal nekdo pravico pisanja (:no

!

!-----

2024_jan_02

Osnove rabe oddaljenih repozitorijev kode, Git in BlueJ

3. Naloga Gitlab, računi na Gitlab, zahteva za dostop do učiteljevega repozitorija

Na <https://gitlab.vegova.si> ustvarite račun. Pomnite geslo, še ga boste potrebovali. Prijavite se v platformo. Prijavljeni izvršite dostop do repozitorija g3_2023_24 (https://gitlab.vegova.si/serhio.w02/g3_2023_24). Desno v vrstici z imenom projekta so 3 vertikalne pike in za tem menu z opcijo, s katero lahko zahtevate dostop do repozitorija. Izvedite zahtevo za dostop.

4. Naloga Deljenje svojega dela, objavljanje oz. potiskanje vsebine v oddaljen repozitorij

Če oddaljen repozitorij z vsebinami obstaja, lahko sledimo postopkom iz naloge2 s tem, da uporabimo prijavo v oddaljen sistem

- ! - **oddaljenega repozitorija (novega projekta) ne moremo narediti iz odjemalca**, torej je postopek:

- I. pridobimo dostop do oddaljenega sistema, ustvarimo si račun ali uporabimo obstoječega.
- II. na oddaljenem sistemu kreiramo nov (PAZI!) prazen projekt. Dobesedno prazen, če vam sistemu ponuja kreiranje kakršnekoli datoteke, zavrnite.
- III. Odprite BlueJ in kreirajte projekt z imenom Testirnik. V projekt dodajte razred z imenom Testni1, katerega glavna metoda naj izpiše 'Dober dan, petek 26. januarja'. Za vsak slučaj program zaženite in preverite njegovo delovanje.
- IV. Iz BlueJ-eve stranske vrstice izberite Share ali iz menija Tools->Team-Share this project.
- V. Izpolnite vnosna polja; URI je URI vašega oddaljenega repozitorija, ostali podatki o računu morajo biti realni podatki, sicer se ne boste uspeli prijaviti, še manj pa pošiljati v oddaljeni repozitorij. Preverite, ali je povezava lahko uspešna (Check connection) in dokončajte 'share'.

V poročilo dodajte sliko oddaljenega repozitorija, delček, kjer je videti ime vašega repozitorija in objavljeni datoteki (vsaj 2?)

5. Naloga Nekaj osnovnih operacij pri delom z Git, BlueJ -Commit/Push upstream

V nadaljevanju; po uspešnem 'share' BlueJ v levem meniju ponuja 3 gumbe : update, commit/push, status. S status preverite ali je prišlo do kakšne spremembe lokalno ali v oddaljenem repozitoriju, z update lokalno uveljavite vse morebitne spremembe, ki so bile narejene v oddaljenem repozitoriju, s commit/push uveljavite lokalne spremembe: pošljete jih v oddaljen repozitorij.

- I. Naredimo spremembo; recimo dodamo opis v README datoteko (besedilna datoteka v BlueJ) /shranimo,zapremo Readme/, nato
- II. preverimo, ali je git zaznal spremembo : Status (Tool->Team->Status), v poročilo prilepite sliko okna statusa.
- III. Če ni sprememb, ponovite I. Sicer uporabimo Commit/Push (Tools->Team->Commit/Push to repository), vpišite Commit Comment (npr. Dopolnitev README datoteke). Pritisnite gumb 'Commit' in dodajte zaslonsko sliko v poročilo. Nato pritisnite gumb push.

2024_jan_02

Osnove rabe oddaljenih repozitorijev kode, Git in BlueJ

IV. oglejte si vsebino README datoteke v oddaljenem repozitoriju preko brskalnika

6. Naloga Nekaj osnovnih operacij pri delom z Git, BlueJ -Update

Update uporabimo, ko je bila narejena sprememba v oddaljenem repozitoriju, pa te spremembe lokalno še nimamo.

I. Odprite datoteko README oddaljenega repozitorija preko brskalnika, spremenite vsebino in uveljavite spremembo na strežniku.

II. V BlueJ kliknite na gumb 'Update' v levem meniju. Zaslonsko sliko dodajte v poročilo. Izvedite update in lokalno preverite ali vsebina datoteke lokalno ustreza spremembam, narejenim na strežniku.

! - vsakič, ko odprete projekt BlueJ, ki je bil kadarkoli povezan z GIT, se bo ta nastavitev avtomatično detektirala in podatki za dostop do oddaljenega repo. bodo vpisani, ukazi Team bodo omogočene. (geslo se ne shranjuje).

! - če pač delate lokalni repozitorij na novo, ker zgornjega nimate, pač vpišete podatke za dostop do GIT repozitorija (glej nalogo 2) in izvedete 'checkout'. Checkout vedno naredi nov lokalni repozitorij in vanj prepíše vso vsebino z oddaljenega repozitorija.

! - če potrebujete samo prenos datotek, je najhitreje z git clone (naloga 1, točka 3); ta podobno kot checkout naredi lokalni git repozitorij, le da ga BlueJ ne zazna kot takega.

2024_jan_02

Osnove rabe oddaljenih repozitorijev kode, Git in BlueJ

Prostor za zapiske/rešitve

2024_jan_02

Osnove rabe oddaljenih repozitorijev kode, Git in BlueJ
