$ sudo apt-get install git-core git-gui git-doc

$ cd /home/linuxtechlab/git\_repo

$ git init

pierwsza rzecz po instalacji

– konfiguracja nazwy użytkownika i adresu email

$ git config --global user.name "Jan Nowak"

$ git config --global user.email jannowak@example.com

– global informuje, żeby Git od tej pory korzystał z tych

danych

– w razie potrzeby modyfikacji – bez opcji global

edytor

$ git config --global core.editor emacs

• narzędzie obsługi różnic

$ git config --global merge.tool vimdiff

• inne narzędzia obsługi różnic

– kdiff3, tkdiff, meld, xxdiff, emerge, vimdiff, gvimdiff, ecmerge,

Opendiff

• sprawdzanie ustawień

$ git config –-list

user.name=Jan Kowalski

user.email=jan@kowalski.pl

color.status=auto

color.branch=auto

color.interactive=auto

color.diff=auto

trzy sposoby wyświetlenia strony podręcznika

$ git help <polecenie>

$ git <polecenie> --help

$ man git-<polecenie>

• na przykład

$ git help config

Tworzenie repozytorium

• Dwa sposoby:

1. inicjalizacja Gita w istniejącym katalogu

$ git init

– tworzy nowy podkatalog o nazwie .git – szkielet

repozytorium Gita

– na razie żadna część projektu nie jest jeszcze śledzona

– rozpoczęcie kontroli wersji, na przykład

$ git add \*.c

$ git add README

$ git commit -m 'initial project version'

2. klonowanie istniejącego repozytorium

To create the new branch, issue the command:

git checkout -b BRANCH

After creating the repository, you will be presented with a URL to use for pushing our local repository. To do this, go back to the terminal window (still within ~/newproject) and issue the commands:

git remote add origin URL

git push -u origin master

### Pulling the project

Say your collaborators make changes to the code on the GitHub project and have merged those changes. You will then need to pull the project files to your local machine, so the files you have on your system match those on the remote account. To do this, issue the command (from within ~/newproject):

git pull origin master

The above command will pull down any new or changed files to your local repository.

Rozwiązanie problemu:

Using the --allow-unrelated-histories flag worked with a pull request in this way:

git pull origin branchname --allow-unrelated-histories