Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

Tím č.5

Metodika verziovania programu

Vedúci tímu: Mgr. Martin Sabo, PhD., Ing. Marta Prnová, PhD.
Vypracovala: Alena Valová
Kontakt: mikasa.fiit@gmail.com

1 Práca v nástroji GitHub

Pre zachovanie rovnorodosti práce, podporu spolupráce tímu na projekte a prevencii vzniku nezrovnalostí vo verziách programu na projekte sú v tomto dokumente definované pravidlá manipulácie s verziami projektu.

Zvoleným kolaboratívnym nástrojom na projekte je Github. Slovenský jazyk sa bude využívať pri vytváraní:

- potvrdzujúcich správ, (ďalej commit message)
- názvov vetiev patriacich úlohám

V anglickom jazyku budú pomenované vetvy master a dev.

Vetvy

Master

Hlavnou vetvou je vetva master. Do tejto vetvy sa budú pridávať len hotové funkcionality spustiteľného programu. V tejto vetve by sa **nesmie** nachádzať program s nedokončenou funkcionalitou a s chybami.

Nad touto vetvou **môže** existovať len vetva dev. Do tejto vetvy sa bude vykonávať merge vetvy dev a to **prednostne** na konci šprintu po odsúhlasení v tíme. Pridaná časť programu **musí** mať hodnotu pre zákazníka. Nedokončená funkcionalita sa do tejto vetvy nepridáva.

Dev

Vetva určená na vývoj je vetva dev. Do tejto vetvy sa budú pridávať len ukončené funkcionality z pridelených úloh. Ak si obsah úlohy žiada testovanie, nesmie byť do vetvy pridaná neotestovaná časť programu. Výnimkou tohto prípadu je scenár, kedy je dané testovanie súčasťou inej úlohy.

Nad touto vetvou **môžu** vznikať len vetvy, ktoré sú vytvorené za účelov splnenia pridelenej úlohy. Do vetvy dev sa môže vykonať merge len s vetvou, ktorej:

- zodpovedajúca úloha je splnená na 100%
- pridaný program zodpovedajúci úlohe prešiel kontrolou druhej strany.

Spájanie vetiev bude vykonávať Jakub Kučečka. Riešenie konfliktov počas spájania bude riešiť osoba, ktorej úloha bola pridelená.

Vetvy úloh

Pre každú úlohu bude existovať vetva, ktorá sa vytvorí z vetvy dev tesne pred začatím programovania a zanikne po ukončení úlohy vykonaním merge do vetvy dev. Pre pod úlohy sa nemusí vytvárať vetva.

Vetvy budú nazývané podľa čísla úloh, priradené systémom Jira (bez diakritiky). Vzor:

ASR-1_vytvorenie_metodiky_verziovania_programu

Okolo pomlčky na nenachádzajú medzery. Za id úlohy nasleduje podčiarkovník a názov úlohy v slovenskom jazyku. Všetky začiatočné písmená sú malé a slová sú rozdelené podčiarkovníkom.

Commit messages

Príklad commit message:

ASR-1 pridanie funkcionality ....

Začína sa id úlohy, nasleduje popis k pridanej funkcionalite. Odporúča sa podrobný popis.

A Cheet Sheet

Odloženie lokálnych zmien

Príkaz	Popis
git stash	odloženie zmeny na vrch fronty (imaginárny priečinok)
git stash pop	vyberanie zmien z vrchu fronty

Pridávanie do globálneho repozitára

Príkaz	Popis
git push -u origin {vetva}	prvý krát je potrebné definovať origin vetvu
git push –set-upstream origin {vetva}	prvý krát pri klonovani je potrebné definovať upstream
git push	pridáva na globál
git push –force	uprednostní vaše zmeny, môže prepisovať pri paralelnom pridávaní
git push –force-with-lease	pridáva hneď ako bude môcť

Sťahovanie z gitu

Príkaz	Popis
git clone {url}	
git clone {html}	
git pull	sťahuje do lokálu a integruje zmeny do pracovných súborov
git fetch	sťahuje iba nové zmeny bez integrácie do pracovných súborov

Práca s vetvami

Príkaz	Popis
git checkout -b {nazov vetvy}	vytvorí vetvu a prepne sa do nej
git checkout {nazov vetvy}	prepne sa do vetvy
git branch	zoznam vetiev
git branch -v	zoznam s posledným commitom
git branch –merged	zoznam zmergovaných vetiev
git branch –no-merged	zoznam nezmergovaných vetiev
git branch -d {nazov vetvy}	mazanie vetvy

Práca s commitmi

Príkaz	Popis
git commit -m ”message”	
git rebase -i HEAD$\sim${$\#commitov$}	spája, maže, pripája, premenováva commity
git rebase –abort	ruší rebase
git rebase –continue	pokračuje v rebase (Dôležité pri merge confliktoch)
git commit –amend	zahodí posledný commit
git branch –no-merged	zoznam nezmergovaných vetiev
git reset –hard	vráti celý repozitár do stavu, v akom bol pri poslednom commite

Práca so súbormi

Príkaz	Popis
git add {zoznam súborov}	
git add .	pridá všetky zmenené súbory
git reset subor	vráti súbor do pôvodného stavu, v akom bol pri poslednom commite

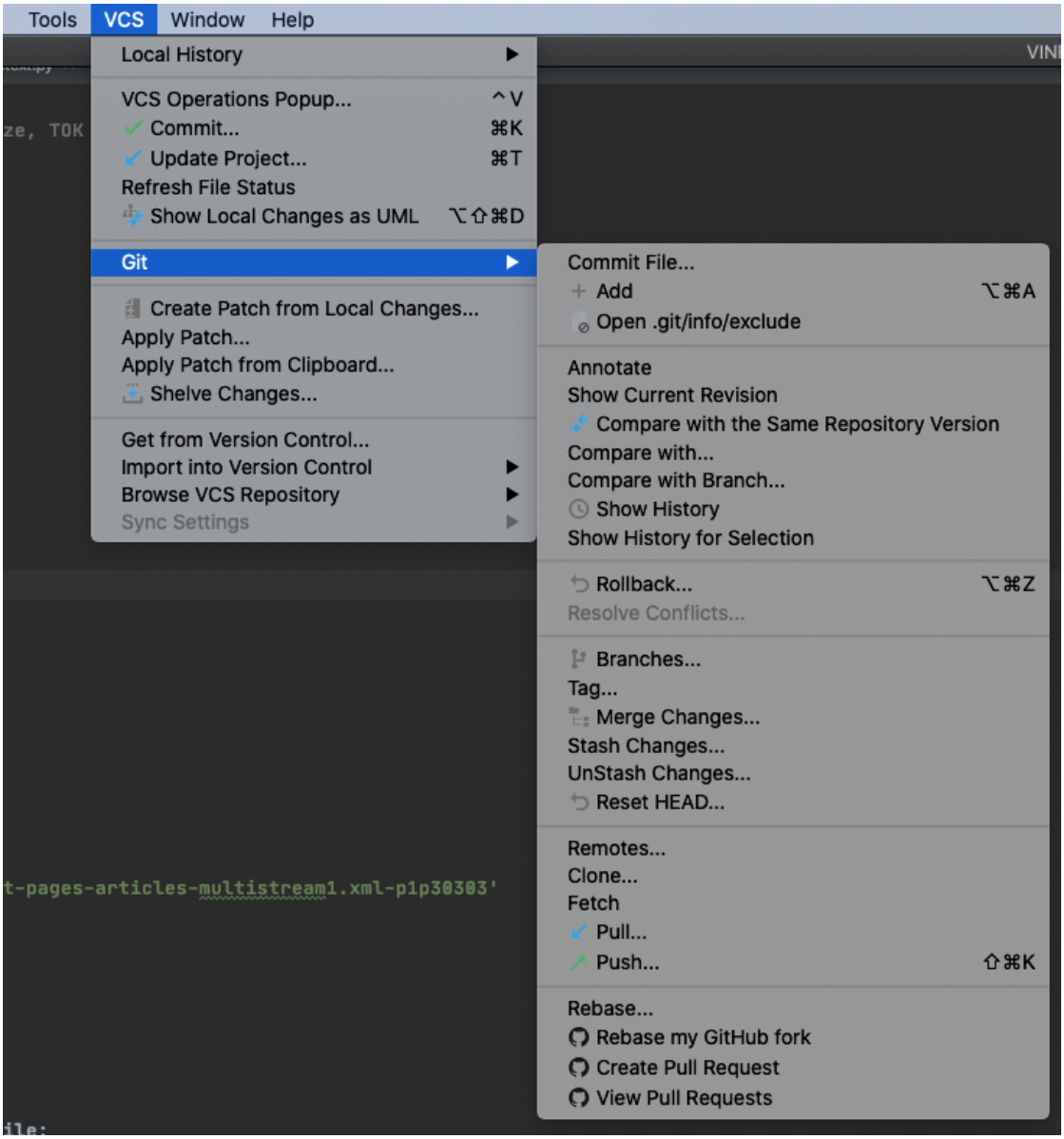
Informácie o repozitári

Príkaz	Popis
git status	stav súborov
git diff	zmeny od posledného pridania (git add)
git log	logovanie commitov
git remote -v	kontrola gitu

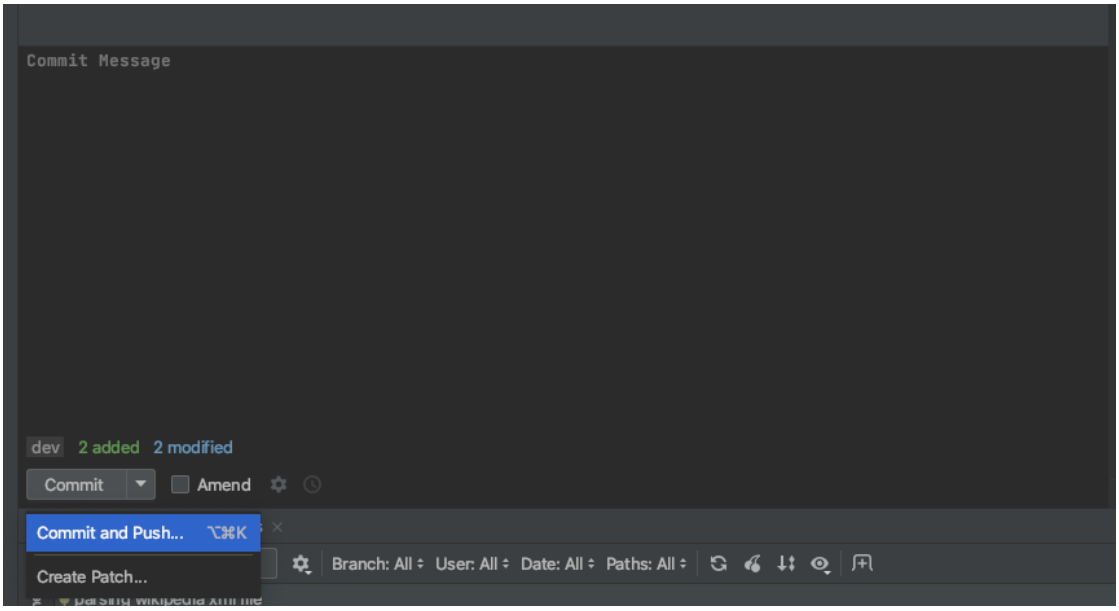
Iné

Príkaz	Popis
git cherry-pick topic	
git reset --merge ORIG_HEAD	
git cherrypick Xpatience topic	

B Práca s Gitom v IDE



Obr. 1: Možnosti v Pycharm



Obr. 2: Commit message. Vykonanie commit a push.

C História

Meno a Priezvisko	Dátum	Opis zmeny
Alena Valová	17.10.2020	Vytvorenie metodiky
Jakub Kučečka	18.10.2020	Vytvorenie Cheet Sheet pre GitHub
...	...	...