

Ministerul Educației a Republicii Moldova
Universitatea Tehnică a Moldovei
Facultatea Calculatoare, Informatică și Microelectronică
Catedra Automatica și Tehnologii Informaționale

Raport

Lucrarea de laborator Nr.2

La disciplina Medii interactive de dezvoltare a produselor soft

Tema: Version control systems și modul de setare a unui server

A efectuat

Vasîlc Iurie

St.gr.TI-144

A verificat

Irina Cojan

Lect.univ.

Chișinău 2016

Scopul lucrării:

- Studierea sistemului de control a versiunilor Git.

Sarcina lucrării:

- Să se studieze sistemul de control al versiunilor Git. Să se realizeze un proiect în repositoryul local. Proiectul să se încarce în repositoryul GitHub.

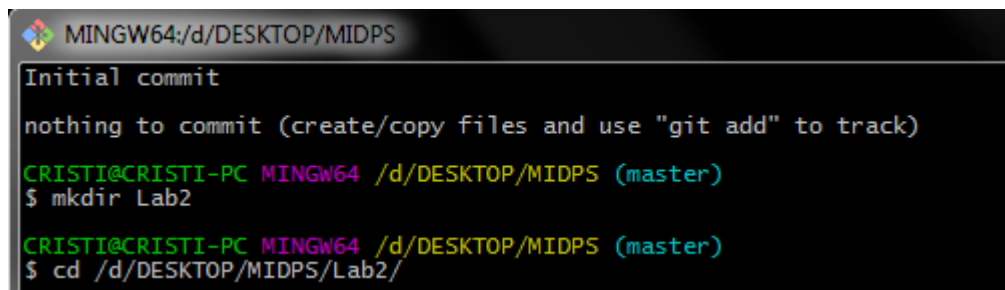
Etapele executării lucrării de laborator:

În cadrul lucrării de laborator respective, a fost propus studiul sistemului de control al versiunilor Git, a particularităților acestuia și a instrucțiunilor principale/generale.

Pentru a realiza sarcina propusă, va fi creat un repozitoriu public pe **GitHub**, cu denumirea **MIDPS-Lab2** [1]. Inițial, într-un repozitoriu public este gol, acesta nu va fi clonat. Schimbările proiectului, realizat în repozitoriul local, vor fi încărcate pe cel public în timpul dezvoltării acestuia.

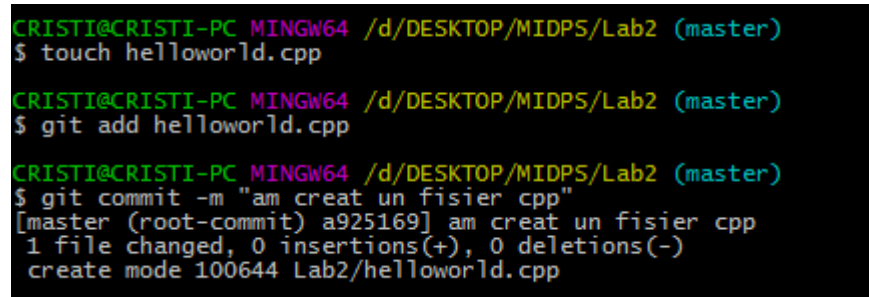
Etapele realizării proiectului (în repozitoriul local):

Inițial, se crează o mapă, MIDPS-Lab2, ce va conține repozitoriul local. Apoi, este rulat Git-ul, și inițializat repozitoriul local:



```
MINGW64:/d/DESKTOP/MIDPS
Initial commit
nothing to commit (create/copy files and use "git add" to track)
CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS (master)
$ mkdir Lab2
CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS (master)
$ cd /d/DESKTOP/MIDPS/Lab2/
```

În cadrul lucrării, va fi creat un fișier cu *extensia .cpp*, în cadrul căruia se va scrie un program de gen „HelloWorld”:



```
CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ touch helloworld.cpp
CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git add helloworld.cpp
CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git commit -m "am creat un fisier cpp"
[master (root-commit) a925169] am creat un fisier cpp
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Lab2/helloworld.cpp
```

Gitul observă că au fost efectuate modificări în cadrul fișierului:

```
CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git status
On branch master
nothing to commit, working directory clean

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   helloworld.cpp

no changes added to commit (use "git add" and/or "git commit -a")
```

Fișierul modificat va fi adăugat în repoziitoriul local:

```
CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git add helloworld.cpp

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git commit -m "am adaugat un comentariu"
[master 235b248] am adaugat un comentariu
1 file changed, 1 insertion(+)

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git status
On branch master
nothing to commit, working directory clean
```

Se adaugă funcția main a programului, commit la modificări:

```
CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git commit -m "am scris functia main"
[master 6f9443c] am scris functia main
1 file changed, 7 insertions(+), 1 deletion(-)

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git status
On branch master
nothing to commit, working directory clean
```

Deci, carcasa programului a fost scrisă. Urmează o decizie: să creez o funcție pentru afișarea mesajului „Hello World”, sau să-l afișez prin intermediul funcției *cout*? Dilema respectivă se soluționează prin utilizarea unui *branch*, ce va permite dezvoltarea concomitentă a fișierului pe cele 2 căi stabilite. Branch-ul respectiv va fi numit *slave*, în cadrul acestuia, programul va fi realizat prin utilizarea unei funcții proprii de afișare. Pe branch-ul inițial, *master*, va fi utilizată funcția *cout*:

```

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git branch
* master

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git branch metal

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git branch
* master
  metal

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (master)
$ git checkout metal
Switched to branch 'metal'

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (metal)
$ git add helloworld.cpp

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (metal)
$ git commit -m "lucram pe alt branch"
[metal 1151c31] lucram pe alt branch
1 file changed, 3 insertions(+)

```

A fost adăugat corpul funcției în fișier, commit la modificari:

```

CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS/Lab2 (metal)
$ git commit -m "am adaugat careva linii"
[metal 57540df] am adaugat careva linii
1 file changed, 2 insertions(+)

```

Modificarea se adaugă în repozitoriu, commit la modificări.

Revenim la starea inițială a fișierului, branch-ul *master*:

Se observă faptul că nu a fost efectuată nici o modificare, aceasta se explică prin faptul că codul scris anterior, a fost scris utilizând o alta ramură de dezvoltare a programului. În cadrul branch-ului *master*, vom scrie programul, utilizând doar funcția *cout*:

Se păstrează modificările, commit la modificări.

Graph-ul branch-ului master este următorul:

Deci, programul a fost dezvoltat pe 2 căi, căile respective sunt:

```
CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS (master)
$ git log --graph
* commit bc25038b93e35e72e58b8a2804c32e7a1b56a75e
  Author: NironesCristian <rateofficial@yandex.com>
  Date: Tue May 31 09:19:51 2016 +0300

    am adaugat gitignore si readme pe master

* commit 13822ae427921558d36bd6267e30c85539383d24
  Merge: 9c75218 57540df
  Author: VasilicIura <iiuurraa@list.ru>
  Date: Tue May 31 09:12:16 2016 +0300

    Merge pull request #1 from VasilicIura/metal

    Metal

* commit 57540dff28e0f8994e80e6e01f679a6e02b92549
  Author: NironesCristian <rateofficial@yandex.com>
  Date: Tue May 31 09:10:13 2016 +0300

    am adaugat careva linii

* commit 1151c31c3b2e5085e2649f3ef0368008fd5aca13
  Author: NironesCristian <rateofficial@yandex.com>
  Date: Tue May 31 09:08:22 2016 +0300

    lucram pe alt branch

* commit 9c75218f005c3a8b362bf6908486e9979f13627f
  Author: NironesCristian <rateofficial@yandex.com>
  Date: Tue May 31 09:05:19 2016 +0300

    am adaugat un newline
```

```
* commit 13822ae427921558d36bd6267e30c85539383d24
  Merge: 9c75218 57540df
  Author: VasilicIura <iiuurraa@list.ru>
  Date: Tue May 31 09:12:16 2016 +0300

    Merge pull request #1 from VasilicIura/metal

    Metal

* commit 57540dff28e0f8994e80e6e01f679a6e02b92549
  Author: NironesCristian <rateofficial@yandex.com>
  Date: Tue May 31 09:10:13 2016 +0300

    am adaugat careva linii

* commit 1151c31c3b2e5085e2649f3ef0368008fd5aca13
  Author: NironesCristian <rateofficial@yandex.com>
  Date: Tue May 31 09:08:22 2016 +0300

    lucram pe alt branch

* commit bc25038b93e35e72e58b8a2804c32e7a1b56a75e
  Author: NironesCristian <rateofficial@yandex.com>
  Date: Tue May 31 09:19:51 2016 +0300

    am adaugat gitignore si readme pe master

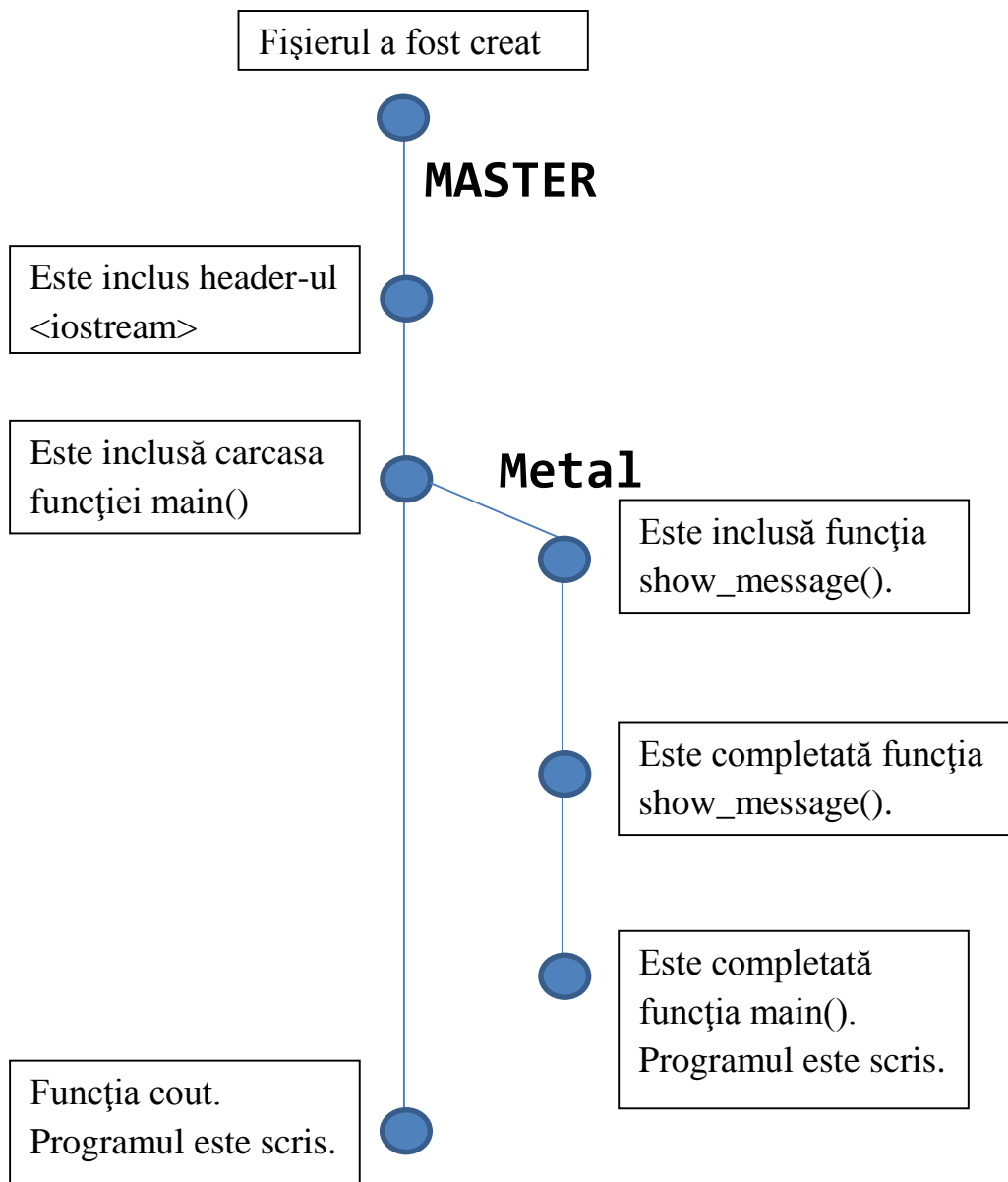
* commit 13822ae427921558d36bd6267e30c85539383d24
  Merge: 9c75218 57540df
  Author: VasilicIura <iiuurraa@list.ru>
  Date: Tue May 31 09:12:16 2016 +0300

    Merge pull request #1 from VasilicIura/metal

    Metal
```

Se observă faptul că fișierul a început să fie modificat pe branch-ul *metal*, începînd cu commit-ul „file modified – main”, etapa la care a fost scrisă carcasa funcției *main*.

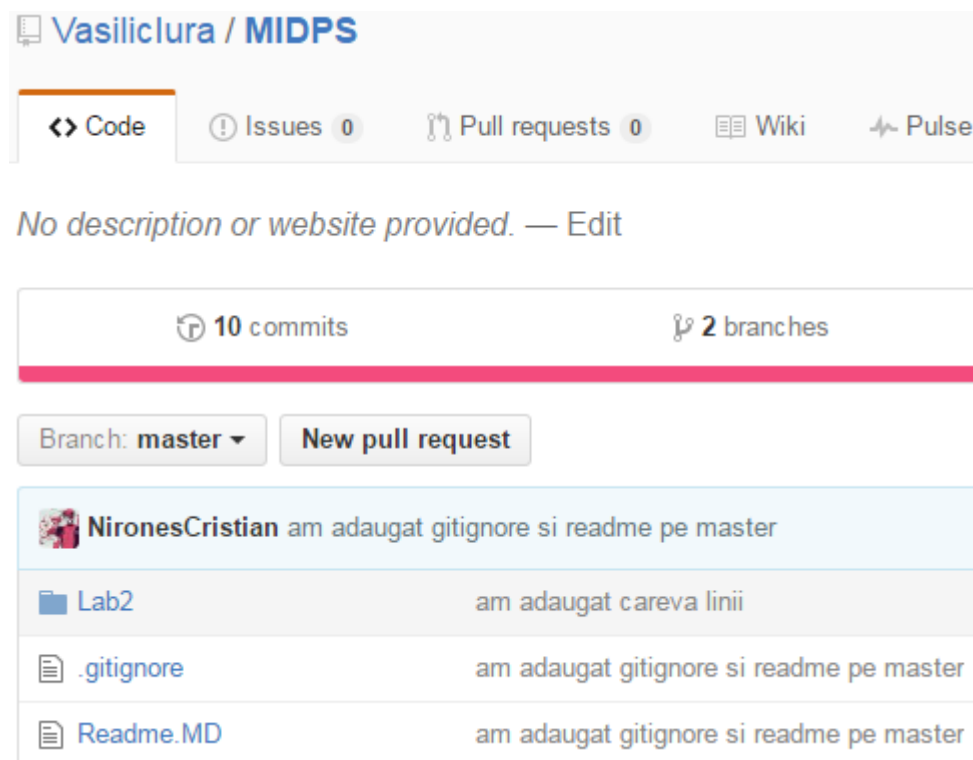
Dezvoltarea pe branch-uri, schematic, este următoarea:



Branch-ul *master* se încarcă pe repozitoriul public:

```
CRISTI@CRISTI-PC MINGW64 /d/DESKTOP/MIDPS (master)
$ git push origin master
Counting objects: 4, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 2.41 KiB | 0 bytes/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/VasilicIura/MIDPS.git
13822ae..bc25038 master -> master
```

Comituri master:



VasilicIura / MIDPS

Code Issues 0 Pull requests 0 Wiki Pulse

No description or website provided. — Edit

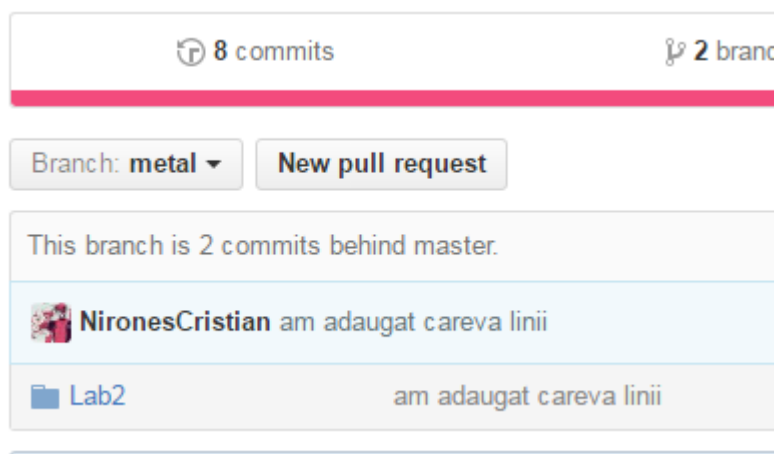
10 commits 2 branches

Branch: master New pull request

NironesCristian am adaugat gitignore si readme pe master

Lab2	am adaugat careva linii
.gitignore	am adaugat gitignore si readme pe master
Readme.MD	am adaugat gitignore si readme pe master

Branch-ul *slave* se încarcă pe repozitoriul public:



8 commits 2 branches

Branch: metal New pull request

This branch is 2 commits behind master.

NironesCristian am adaugat careva linii

Lab2	am adaugat careva linii
------	-------------------------

Showing 1 changed file with 2 additions and 0 deletions.

2 Lab2/helloworld.cpp		
		@@ -9,6 +9,8 @@ int main()
9	9	
10	10	cout<<"Hello, world!"<<endl;
11	11	cout<<"Hi! My name is..."<<endl;
	12	+ cout<<"Who! My name is..."<<endl;
	13	+ cout<<"<scratch sounds> Slim Shadyyy?"<<endl;
12	14	
13	15	return 0;
14	16	} 0+

În rezultat, fișierul *hello.cpp* este încărcat pe repositoryul public pe 2 branch-uri pentru o ulterioară dezvoltare.

Conform condiției, este necesar a crea un file README, în care descriu ce am realizat în cadrul lucrării, ce dificultăți am întâlnit. Întru-cât am descris cele realizate în cadrul raportului, în fișier vor fi descrise dificultățile întâlnite.

Cele scrise în fișier nu vor fi incluse în raport. Va fi realizat un *push* pentru fiecare schimbare.

Notă finală:

Pe parcursul proiectului, a fost creat branch-ul slave, care ulterior nu a fost dezvoltat. A fost posibilă realizarea procedurii merge, însă conținutul fișierului hello.cpp este diferit pe ambele ramuri, respectiv aceasta va crea un conflict. Pentru a nu „complica” conținutul proiectului, conflictul se va soluționa în studiul ulterior al Git-ului.

Concluzie

În cadrul lucrării de laborator nr.2, a fost studiat sistemul de control al versiunilor, Git, particularitățile acestuia, comenzile *bash* generale. Au fost create/modificate repozitorii locale/publice. S-a concluzionat faptul că Git-ul este un sistem necesar oricărui programator/developer, întrucât permite dezvoltarea și structurarea pas cu pas a proiectelor și versionarea fișierelor.

Un posibil neajuns al sistemului este logica complicată a modului de structurare al informației (dezvoltarea pe branch-uri).

Au fost obținute noi cunoștințe în domeniul mediilor interactive de dezvoltare a produselor soft.

Bibliografie

1. Repozitoriul public GitHub:

<https://github.com/NironesCristian/MIDPSLab2>

2. Tutorial Git:

<https://githowto.com/>