

FACULTATEA CALCULATOARE, INFORMATICĂ ȘI MICROELECTRONICĂ

UNIVERSITATEA TEHNICĂ A MOLDOVEI

MEDII INTERACTIVE DE DEZVOLTARE A PRODUSELOR SOFT

LUCRARE DE LABORATOR #2

Version Control Systems Si Modul De Setare A Unui Server

*Autor:*

st. gr. TI-142

Damaschin Ion

*lector asistent:*

Irina COJANU

*lector superior:*

Svetlana COJOCARU

## LUCRARE DE LABORATOR #2

### 1. Scopul lucrării

Înșușirea noțiunii de Version Control Systems și a modului de setare a unui server.

### 2. Obiectivele lucrării

- a) Înțelegerea și folosirea CLI (basic level)
- b) Administrarea remote a mașinilor linux machine folosind SSH (remote code editing)
- c) Version Control Systems (git || mercurial || svn)
- d) Compilează codul C/C++/Java/Python prin intermediul CLI, folosind compilatoarele gcc/g++/javac/python

### 3. Efectuarea lucrării de laborator

#### 3.1. Task-uri implementate

- *Advanced Level* (nota 9 || 10):
  - setează un branch to track a remote origin pe care vei putea sa faci push (ex. Github, Bitbucket or custom server)
  - resetează un branch la commit-ul anterior
  - merge 2 branches
  - conflict solving between 2 branches

### 3.2. Realizarea lucrării de laborator

- *Advanced Level* (nota 9 || 10):
  - setează un branch to track a remote origin pe care vei putea sa faci push (ex. Github, Bitbucket or custom server)

```
MINGW64:/c/Users/cubyk/MIDPS
cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (master)
$ git branch
* master

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (master)
$ git checkout -b Lab4
Switched to a new branch 'Lab4'

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ .....
```

```
cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git init
Reinitialized existing Git repository in C:/Users/cubyk/MIDPS
/.git/

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git config --global user.name "CubykRubyk"

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git config --global user.email "ion.damaschin@hotmail.com"

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git status
On branch Lab4
nothing to commit, working directory clean

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git status
On branch Lab4
Untracked files:
  (use "git add <file>..." to include in what will be committ
ed)

       Lab#4/

nothing added to commit but untracked files present (use "git
add" to track)

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git add .
```

```

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git push origin Lab4
Enter passphrase for key '/c/Users/cubyk/.ssh/id_rsa':
Counting objects: 36, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (35/35), done.
Writing objects: 100% (36/36), 21.99 KiB | 0 bytes/s, done.
Total 36 (delta 9), reused 0 (delta 0)
To git@github.com:CubykRubyk/MIDPS.git
 * [new branch]      Lab4 -> Lab4

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$

```

- o resetează un branch la commit-ul anterior

```

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git log --graph --all --decorate
* commit d05404656c4e86bf5377821597e3d4ab340e3b70 (HEAD -> Lab4, origin/Lab4)
  Author: CubykRubyk <ion.damaschin@hotmail.com>
  Date: Thu Mar 31 08:18:30 2016 +0300

    Add lab4

* commit 4500d4baa1982fcd778b75b2eb0b82ad599b7537 (origin/master, origin/HEAD, master)
  Author: Ion Damaschin <Ion Damaschin>
  Date: Mon Mar 21 20:53:11 2016 +0200

    Replace calculator.java

* commit 7d797a8b8b5203bf5a154de3b34196044b612eca
  Merge: bfa9aab 7473a62
  Author: Ion Damaschin <Ion Damaschin>
  Date: Mon Mar 21 20:51:38 2016 +0200

    Replace calculator.java

* commit 7473a626686cf47c868d50e51a9439cf07d7f5f4
  Author: CubykRubyk <ion.damaschin@hotmail.com>
  Date: Mon Mar 21 20:50:01 2016 +0200

    Delete calculator.java

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git checkout 4500d4baa1982fcd778b75b2eb0b82ad599b7537
Note: checking out '4500d4baa1982fcd778b75b2eb0b82ad599b7537'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

  git checkout -b <new-branch-name>

HEAD is now at 4500d4b... Replace calculator.java
cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS ((4500d4b...))
$

```

- merge 2 branches

```
cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (master)
$ git merge master Lab4
Updating 4500d4b..d054046
Fast-forward
 Lab#4/.DS_Store | Bin 0 -> 8196 bytes
 Lab#4/README.md | 4 +
 Lab#4/admin/add-post.php | 107 ++++++++
 Lab#4/admin/add-user.php | 105 ++++++++
 Lab#4/admin/edit-post.php | 131 ++++++++
 Lab#4/admin/edit-user.php | 147 ++++++++
 Lab#4/admin/index.php | 87 ++++++
 Lab#4/admin/login.php | 53 +++++
 Lab#4/admin/logout.php | 9 +
 Lab#4/admin/menu.php | 9 +
 Lab#4/admin/users.php | 94 ++++++
 Lab#4/classes/class.password.php | 213 ++++++++
 Lab#4/classes/class.user.php | 56 +++++
 Lab#4/db.sql | 50 +++++
 Lab#4/images/.DS_Store | Bin 0 -> 6148 bytes
 Lab#4/images/magnify.gif | Bin 0 -> 970 bytes
 Lab#4/includes/config.php | 44 +++++
 Lab#4/index.html | 37 +++++
 Lab#4/index.php | 63 ++++++
 Lab#4/scripts/.DS_Store | Bin 0 -> 6148 bytes
 Lab#4/scripts/custom.js | 47 +++++
 Lab#4/search.php | 111 ++++++++
 Lab#4/style/.DS_Store | Bin 0 -> 6148 bytes
 Lab#4/style/main.css | 95 ++++++++
 Lab#4/style/normalize.css | 501 ++++++++
```

- rezolvarea conflictelor a 2 branches

```
MINGW64:/c/Users/cubyk/MIDPS

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (master)
$ echo "test = 1" >> test.txt

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (master)
$ git add . && git commit -m "test"
warning: LF will be replaced by CRLF in test.txt.
The file will have its original line endings in your working directory.
[master warning: LF will be replaced by CRLF in test.txt.
The file will have its original line endings in your working directory.
7d17cea] test
warning: LF will be replaced by CRLF in test.txt.
The file will have its original line endings in your working directory.
1 file changed, 1 insertion(+)

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (master)
$ git checkout Lab4
Switched to branch 'Lab4'

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ echo "test = 0" >> test.txt

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git add . && git commit -m "test"
warning: LF will be replaced by CRLF in test.txt.
The file will have its original line endings in your working directory.
[Lab4 warning: LF will be replaced by CRLF in test.txt.
The file will have its original line endings in your working directory.
221c19a] test
warning: LF will be replaced by CRLF in test.txt.
The file will have its original line endings in your working directory.
1 file changed, 1 insertion(+)

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (Lab4)
$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (master)
$ git merge Lab4
Auto-merging test.txt
CONFLICT (add/add): Merge conflict in test.txt
Automatic merge failed; fix conflicts and then commit the result.

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (master|MERGING)
$ !
```

```
cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (master|MERGING)
$ git add .

cubyk@WINCTRL-G7MU81Q MINGW64 ~/MIDPS (master|MERGING)
$ git commit -m "rezolvare conflict"
[master ec1f4d4] rezolvare conflict
```

## Concluzie

În urma realizării laboratorului nr.2 la tema: *"Version Control Systems si modul de setare a unui server"*, am însușit modul de utilizare a CLI, de administrarea remote a mașinilor linux machine folosind SSH.

Am efectuat conexiunea la un remote server folosind SSH (drept server remote, am folosit o mașină virtuală).

Am creat 2 branch-uri asupra cărora am efectuat un commit din nou.

În timpul efectuării laboratorului am lucrat cu comenzi ca :

- git init - crearea unui repository dintr-un fișier existent
- git remote add origin - pentru interconectarea repositoryului local cu cel de pe github.
- git commit -m - pentru înregistrarea unei schimbări(snapshot), ca git add.
- git config --global - operațiuni cu fișierul de configurație GIT Bash
- git checkout - pentru selectarea ramurii curente de lucru.
- git status - informații despre starea fișierelor
- git merge - actualizarea schimbărilor de pe două sau mai multe ramuri.

În concluzie, am pus în practică VCS-ul GIT Bash creând un repository și inițializându-l, clonând repository și efectuând diverse comenzi în el.

## Bibliografie

1. <https://git-scm.com>
2. <https://git-scm.com/docs/git-help>