

MINISTERUL EDUCATIEI DIN REPUBLICA MOLDOVA  
UNIVERSITATEA TEHNICA DIN MOLDOVA  
FACULTATEA "CALCULATOARE, INFORMATICA SI MICROELECTRONICA  
FILIERA ANGLOFONA

---

# Raport

LUCRARE DE LABORATOR NR. 1  
LA  
MEDII INTERACTIVE DE DEZVOLTARE A PRODUSELOR SOFT

---

*A efectuat:*

st.gr. FAF-131

Tudor Plugaru

*A verificat:*

Sava Virtosu

Chisinau 2015

## Connect to a remote server via SSH:

I have installed Ubuntu Server on virtualbox on my linux machine. I started the virtual machine and I have connected to it by typing the following command in Terminal:

```
ssh 192.168.1.239
```

After this I have entered the user password from my server and the connection succeeded.

## Initialize a repository on server & Create a file in repository folder, write in your name, save it and commit it:

```
cd IDE/Lab_1
vim README.md
#pressed i to enter insert mode
#pressed ESC and typed
:wq #write & quit
cd ../../
git init
git remote add origin git@bitbucket.org:PlugaruT/ide.git
git status
git add . #add necessary files to git
git commit -am 'Lab#1'
git push origin master
```

I have created a real repository on my BitBucket account and pushed all my work there.

## Create 2 more branches with at least one unique committed file per branch:

```
git checkout -b branch1 #created switched to a new branch "branch1"
vim branch1.txt #created a file and wrote some text in it
git add branch1.txt
git commit -m 'branch1.txt added'
git push origin branch1 #pushed the file to branch1
```

```
git checkout master #switched back to master branch
```

```
git checkout -b branch2 #created branch2
vim branch2.txt #created a file and wrote some text in it
git add branch2.txt
git commit -m 'branch2.txt added'
git push origin branch2
```

```
git checkout master
git merge branch2 #merged branch2 with master branch
#Updating 17432e1..8512197
#Fast-forward
#Lab#1/branch2.txt | 1 +
#1 file changed, 1 insertion(+)
#create mode 100644 Lab#1/branch2.txt
git branch -d branch2 #deleted branch2
#Deleted branch branch2 (was 8512197).
```

## Learn 10 vim commands:

Commands I have used and know are:

- i *#enter insert mode*
- esc
- dd *#delete current line*
- gg *#move cursor to first line*
- :wq *#write and quit*
- h j k l *#move through file*
- /word *#search "word" in file*
- dG *#delete all lines*
- u *#undo last change*
- v *#select text*

## Create a VCS alias:

In order to create a VCS alias I have edited the `/.gitconfig` file.

```
vim ~/.gitconfig
```

This is how the section alias looks:

```
[alias]
  st = status
  co = checkout
  lg = log --graph --pretty=format: '%C(bold red)%h%Creset -%C(yellow)%d%Creset %s %Cgreen(%cr) %C(bold green)<%an>%Creset'
      --abbrev-commit --date=relative
```

Now I can use shortcuts like "git st" instead of "git status".

## Create a VCS merge conflict and solve it:

For this I have created a new branch branch3 and edited the file branch2.txt. In master branch this file looked like this:

```
This is file from Branch master
```

In branch3 same file looked like this:

```
This is file from Branch master3
```

To merge this two branches I typed the following in Terminal:

```
git checkout master #switched in master branch
git merge branch3
#Auto-merging Lab#1/branch2.txt
#CONFLICT (content): Merge conflict in Lab#1/branch2.txt
#Automatic merge failed; fix conflicts and then commit the result.
vim branch2.txt
```

The content of file branch2.txt:

```
<<<<<<< HEAD
This is file from Branch master
=====
This is file from Branch branch
>>>>>>> branch3
```

To solve this conflict I have deleted the changes from branch3 and merging this two branches succeeded.

## **Install a code-highlighter plugin in your CLI text editor:**

I have installed a vim plugin manager named Vundle. Using this plugin manager it is very easy to install plugins for vim. I have installed a C++ syntax highlighter using this commands:

```
vim ~/.vimrc
```

I added this line in file:

```
Plugin 'octol/vim-cpp-enhanced-highlight'
```

and then in vim typed:

```
:PluginInstall
```

Now the plugin is installed and ready for use.

## **Conclusion:**

After this laboratory work I have got familiar with VCS systems and gained skills working with git. VCS systems are very useful when working in a team or in a big company. Many developers can work on the same project from different places and having possibility to have the latest version of project. Also, using CLI editors, like vim, I found out that this kind of editors are not so easy to use, but very powerful when you learn a little bit about them. I think that I will continue to use vim from now on because I liked it a lot.