

Ministerul Educatiei din Republica Moldova
Universitatea Tehnica din Moldova
FCIM

Catedra Automatică și Tehnologii Informaționale

Raport

Lucrare de laborator nr.1
MIDPS

Tema: *Version Control Systems si modul de setare a unui server*

A efectuat: st.gr TI-154

Cobîlaș V.

A verificat:

Gojin V.

Chisinau 2017

1.Scopul lucrării: Insusirea notiunii de Version Control Systems si a modului de setare a unui server. Version Control Systems (git - bitbucket - mercurial - svn)

2.Sarcina lucrarii : Sa se studieze sistemul de control al versiunilor Git. Sa se realizeze un proiect in repozitoriul local. Proiectul sa se incarce in repozitoriul GitHub. Sa se efectueze diferite modificari.

3. Efectuarea lucrarii de laborator

3.1 Sarcinile propuse pentru efectuare lucrarii de laborator

Basic Level (nota 5 - 6) :

- initializeaza un nou repository
- configureaza-ti VCS - crearea branch-urilor (creeaza cel putin 2 branches)
- commit pe ambele branch-uri (cel putin 1 commit per branch)

Normal Level (nota 7 - 8):

- seteaza un branch to track a remote origin pe care vei putea sa faci push (ex. Github, Bitbucket or custom server)
- reseteaza un branch la commit-ul anterior salvarea temporara a schimbarilor care nu se vor face commit imediat.
- folosirea fisierului .gitignore

Advanced Level (nota 9 - 10):

- merge 2 branches
- rezolvarea conflictelor a 2 branches
- comezile git care trebuie cunoscute

3.2 Realizarea lucrarii de laborator

Basic Level (nota 5 - 6) :

- Primul pas in executarea acestei lucrari de laborator a fost crearea unui repository, apasind butonul New de pe pagina utilizatorului, tab-ul cu denumirea Repositories. Dupa setarea numelui pentru repository, s-a apasat create repository.
- Configurarea VCS. S-a crearea unui ssh key si s-a copiat in lista de key in account-ul github. Apoi s-a clonat repositoryul local.

- S-a configurat config-ul la git prin intermediul comenzilor git config user.name "VasileCobilas" si git config user.email "cobilas2013@mail.ru" [Figura 1]

- Au fost create 2 branchuri ,comenzile care au fost folosite pentru crearea branchurilor: git checkout -b 'denumire branchului' si git checkout -b 'denumire branchului'. Apoi cu ajutorul comenzii git push origin 'denumirea branchului' au fost incarcate pe github.[Figura 2,3]

- Apoi s-a realizat cite un commit pentru fiecare din branchurile create folosind comenzile: git checkout 'denumirea branchului' (pentru a schimba ramificarea directorului git.) git add . (adauga fisiere la commit) git commit -m "descriere" (salveaza schimbarile in head) git push origin 'denumirea branchului' (Trimite/publica ramificare curenta). [Figura4.1 , 4.2]

Normal Level (nota 7 - 8):

- S-a setat un branch to track pe care s-a facut push cu comenzile: git checkout --track -b new origin/master (seteaza un branch to track) git add . (adauga fisiere la commit) git commit -m "add new branch" (salveaza schimbarile in head) git push origin new (Trimite/publica ramificare curenta.)[Figura 5]

- S-a resetat branchul branch two la commtiul anterior cu ajutorul comenzilor: git log --graph --oneline (pentru a vedea evolutia directorului git.) git reset --hard (Reinitializeaza indexul si directorul de lucru la starea ultimului commit)[Figura 6]

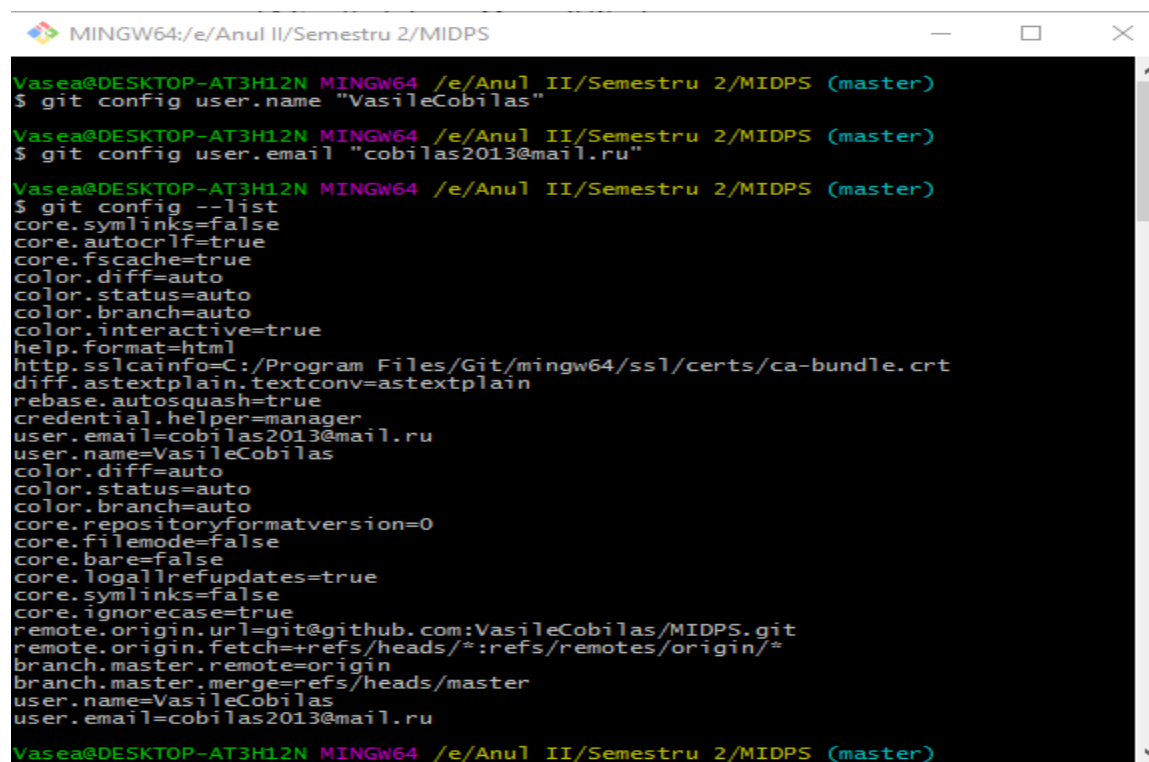
- S-a adaugat un fisier in .gitignore ce a facut comenzile git sa nu il considere ca un fisier din proiect. Dupa cum se observa in screen ca dupa ce am creat fisierul Ignore.lib si apoi am selectat comanda 'git status' a aparut mesajul ca nu a fost facut nici o modificare in repozitoriu, ceea ce inseamna ca fisierul a fost ignorat.Comenzile folosite: echo "Ignore" Ignore.lib (Creaza un fisier sau inscrie ceva in fisier) git status (arata starea fisierelor)[Figura 7]

Advanced Level (nota 9 – 10) :

- S-a facut merge la 2 branchuri cu ajutorul comenzii git merge si denumirea la branch-ul cu care s-a dorit sa se faca merge[Figura 8]

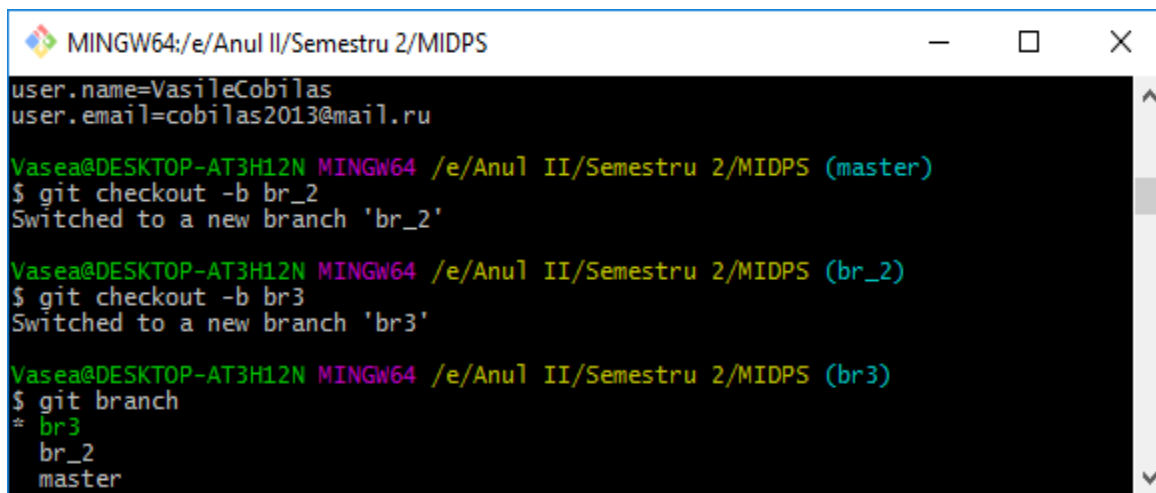
- S-a facut cunostinta cu majoritatea comenzilor Git care au fost oferite de catre profesor. <https://www.siteground.com/tutorials/git/commands.htm>

- S-a creat un tag nou [Figura 9]



```
MINGW64:/e/Anul II/Semestru 2/MIDPS
Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git config user.name "VasileCobilas"
Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git config user.email "cobilas2013@mail.ru"
Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git config --list
core.symlinks=false
core.autocrlf=true
core.fscache=true
color.diff=auto
color.status=auto
color.branch=auto
color.interactive=true
help.format=html
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
diff.astextplain.textconv=astextplain
rebase.autosquash=true
credential.helper=manager
user.email=cobilas2013@mail.ru
user.name=VasileCobilas
color.diff=auto
color.status=auto
color.branch=auto
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
remote.origin.url=git@github.com:VasileCobilas/MIDPS.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.master.remote=origin
branch.master.merge=refs/heads/master
user.name=VasileCobilas
user.email=cobilas2013@mail.ru
Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
```

Figura 1.Configurare



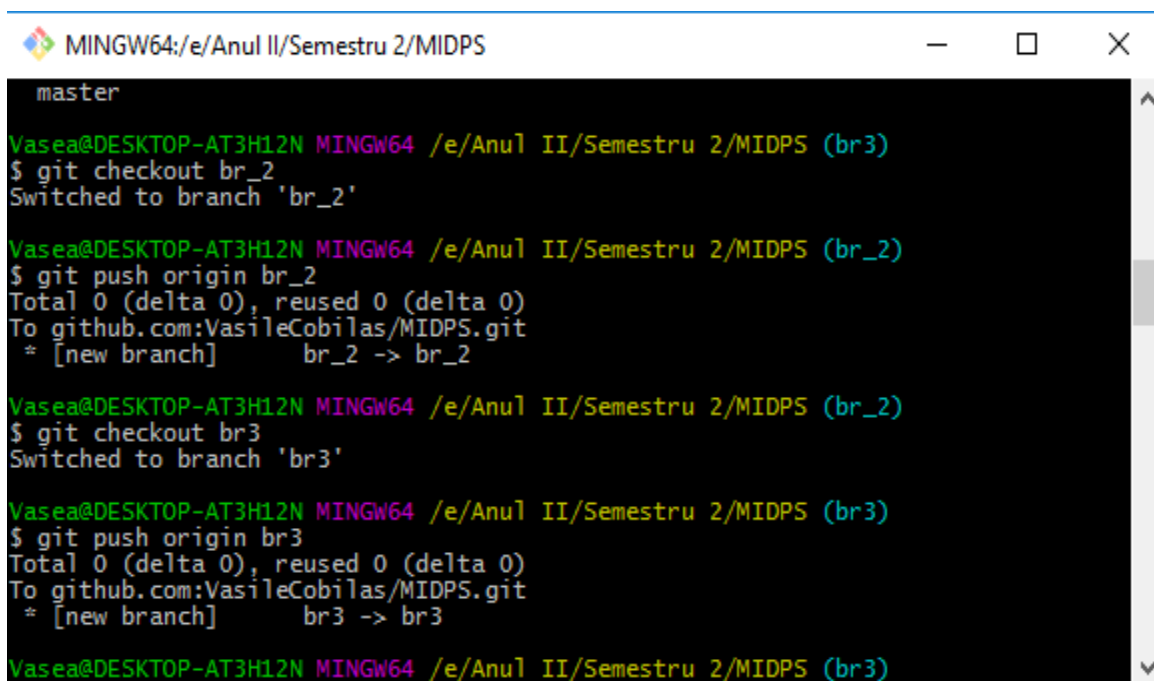
```
MINGW64:/e/Anul II/Semestru 2/MIDPS
user.name=VasileCobilas
user.email=cobilas2013@mail.ru

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git checkout -b br_2
Switched to a new branch 'br_2'

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br_2)
$ git checkout -b br3
Switched to a new branch 'br3'

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br3)
$ git branch
* br3
  br_2
  master
```

Figura 2.Crearea a doua branch-uri



```
MINGW64:/e/Anul II/Semestru 2/MIDPS
master

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br3)
$ git checkout br_2
Switched to branch 'br_2'

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br_2)
$ git push origin br_2
Total 0 (delta 0), reused 0 (delta 0)
To github.com:VasileCobilas/MIDPS.git
 * [new branch]      br_2 -> br_2

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br_2)
$ git checkout br3
Switched to branch 'br3'

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br3)
$ git push origin br3
Total 0 (delta 0), reused 0 (delta 0)
To github.com:VasileCobilas/MIDPS.git
 * [new branch]      br3 -> br3

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br3)
```

Figura 3.Încărcarea branch-urilor

```
MINGW64:/e/Anul II/Semestru 2/MIDPS
Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br3)
$ git checkout br_2
Switched to branch 'br_2'

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br_2)
$ git status
On branch br_2
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        br2.txt

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

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br_2)
$ git add .

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br_2)
$ git commit -m "cream un nou branch"
[br_2 162006f] cream un nou branch
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 br2.txt

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br_2)
$ git push origin br_2
Counting objects: 2, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 244 bytes | 0 bytes/s, done.
Total 2 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local objects.
To github.com:VasileCobilas/MIDPS.git
    1740e96..162006f  br_2 -> br_2
```

Figura 4.1 Comit pe br_2

```
MINGW64:/e/Anul II/Semestru 2/MIDPS
Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br_2)
$ git checkout br3
Switched to branch 'br3'

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br3)
$ git status
On branch br3
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        br3.txt

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

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br3)
$ git add .

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br3)
$ git commit -m "comit pe al 3 branch"
[br3 86c2310] comit pe al 3 branch
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 br3.txt

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br3)
$ git push origin br3
Counting objects: 2, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 246 bytes | 0 bytes/s, done.
Total 2 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local objects.
To github.com:VasileCobilas/MIDPS.git
    1740e96..86c2310  br3 -> br3
```

Figura 4.2 Commit pe br3

```
MINGW64:/e/Anul II/Semestru 2/MIDPS

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (br3)
$ git checkout --track -b new origin/master
Branch new set up to track remote branch master from origin.
Switched to a new branch 'new'

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (new)
$ git add .

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (new)
$ git commit -m "adaugam un branch nou"
On branch new
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (new)
$ git push origin new
Total 0 (delta 0), reused 0 (delta 0)
To github.com:VasileCobilas/MIDPS.git
 * [new branch]      new -> new
```

Figura 5. Setam un branch to track

```
MINGW64:/e/Anul II/Semestru 2/MIDPS

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git stash
Saved working directory and index state WIP on master: cf86613 Corectare
HEAD is now at cf86613 Corectare

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git add .

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git commit -m "reset"
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git log --graph --all --oneline

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git log --graph --all --oneline
* aeca2a6 WIP on master: cf86613 Corectare
|
| * 956f1d8 index on master: cf86613 Corectare
|
| * cf86613 Corectare
| * 67bd5b4 Mersul
| * 380ac15 raport nefinisat
| * 86c2310 comit pe al 3 branch
|
| * 162006f cream un nou branch
|
| * 1740e96 curaj
| * d4252e2 Folder
| * 7aba97e Initial commit
|
$
$

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git reset --hard HEAD
HEAD is now at cf86613 Corectare

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git reset HEAD~
Unstaged changes after reset:
M README.md

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git reset HEAD~
Unstaged changes after reset:
M README.md

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git status
On branch master
```

```

Your branch is behind 'origin/master' by 2 commits, and can be fast-forwarded.
(use "git pull" to update your local branch)
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:   README.md

no changes added to commit (use "git add" and/or "git commit -a")
Vasea@DESKTOP-AT3H12N MINGW64 /e/Anu1 II/Semestru 2/MIDPS (master)
$ |

```

Figura 6 .Resetam un branch la comitutul anterior

```

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anu1 II/Semestru 2/MIDPS (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working tree clean

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anu1 II/Semestru 2/MIDPS (master)
$ echo "Ignore" >> Ignore.lib

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anu1 II/Semestru 2/MIDPS (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        Ignore.lib

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

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anu1 II/Semestru 2/MIDPS (master)
$ git add .
warning: LF will be replaced by CRLF in Ignore.lib.
The file will have its original line endings in your working directory.

```

Figura 7 .Ignorarea unui file

```

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anu1 II/Semestru 2/MIDPS (new)
$ git merge master
Updating 380ac15..cf86613
Fast-forward
 README.md | 10 ++++++---
 1 file changed, 9 insertions(+), 1 deletion(-)

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anu1 II/Semestru 2/MIDPS (new)

```

Figura 8.Merge la doua branch-uri

```

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git tag

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git tag v1.0

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git tag
v1.0
fatal: tag 'v1.0' already exists

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ git push origin --tags
Total 0 (delta 0), reused 0 (delta 0)
To github.com:VasileCobilas/MIDPS.git
 * [new tag]          v1.0 -> v1.0

Vasea@DESKTOP-AT3H12N MINGW64 /e/Anul II/Semestru 2/MIDPS (master)
$ |

```

Figura 9. Folosirea tagurilor

Concluzie: In urma efectuării lucrării de laborator nr.1, a fost studiat sistemul de control al versiunilor, Git, particularitățile acestuia, comenzile bash generale, s-a creat noi branch-uri prin terminalul git bash. S-a încărcat în branch-uri fișiere cu ajutorul instrucțiunii git commit pe repositoryul creat MIDPS, în același timp a fost studiată noțiunea de version control system (este o categorie de instrumente software pentru a ajuta grupurile care se ocupă de software să poată dirija cu schimbările în codurile sale în orice timp la dorința acestora). Au fost create și modificate repository-uri locale și publice. S-a concluzionat faptul că Git-ul este un sistem necesar oricărui programator și developer, într-un anumit grad 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. Repositoryul public <https://github.com/VasileCobilas/MIDPS>
2. Tutorial Git: <http://all.webng.md/traduceri/everyday-ro.html>