

Ministerul Educației al Republicii Moldova
Universitatea Tehnică a Moldovei
Facultatea Calculatoare, Informatică și Microelectronică
Departamentul Ingineria Software și Automatică

Raport

Lucrarea de laborator nr. 1

Disciplina: Programarea în rețea

Tema: Versionarea codului sursă utilizând GIT

A realizat

C. Olaru

St. gr. TI-144

A verificat

Lect. asis.

E. Donos

Chișinău 2017

Cuprins

Aplicarea în practică a sistemului GIT	2 - 8
Construirea proiectelor Java utilizind Apache Maven.....	8 - 11
Concluzie.....	12
Anexa A.....	13

Link repozitoriu: <https://github.com/olarucristian/pr>

Scopul lucrării: Lucrarea de laborator are ca scop studiul și înțelegerea principiilor de funcționare și utilizare a sistemului distribuit de control al versiunilor numit GIT.

Obiectivul lucrării: Crearea unui repozitoriu distant, localizat de serviciul gitlab.ati.utm.md, și sincronizarea tuturor modificărilor efectuate asupra repozitoriului local.

Sarcină tip pentru lucrarea de laborator: Reprezentarea vizuală a pașilor necesari pentru efectuarea lucrării de laborator include elemente grafice cu semnificații corespunzătoare

comenzilor git. Prin urmare este necesar de luat în considerare următoarele:

a) @ asociază repozitoriul distant prin `git remote add origin`;

b) + adaugă fișiere/directorii și modifică repozitoriul local prin `commit`;

c) - șterge fișiere/directorii și modifică repozitoriul local prin `commit`;

d) ~ modifică fișiere și repozitoriul local prin `commit`;

e) * (branch) îmbină o ramură în cea indicată, `git merge`;

f) ↓ `git pull`, trage modificările din repozitoriul distant;

g) ↓↑ `pull and push`, sincronizează repozitoriul local și distant;

În corespundere cu elementele de notație, pașii efectuați în indicațiile metodice pot corespunde grafului de mai jos :

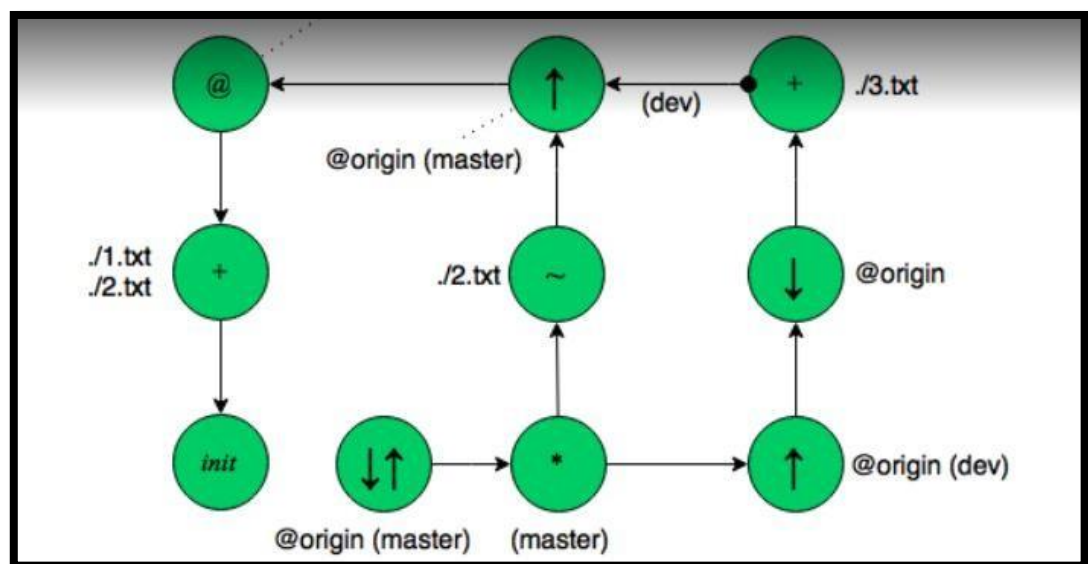


Figura 1 – Variantă pentru lucrare de laborator

1. Aplicarea în practică a sistemului GIT

Pentru a efectua această lucrare de laborator avem nevoie de sistemul de control Git pe care îl putem descărca de pe adresa: <http://git-scm.com/downloads>. Fișierele unui proiect pot fi păstrate atât local cât și pe <https://github.com/olarucristian/pr>. Pentru aceasta avem nevoie să creăm cont nou și să generăm o cheie ssh pentru a conecta repozitoriul local cu cel de pe server. Primul pas pe care trebuie să îl facem pentru a începe lucrul cu Git este să creăm un director local, să pornim de acolo Git și să inițializăm repozitoriul în acest director (vezi în figura 2).

```
Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2
$ mkdir lab1

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2
$ cd lab1

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1
$ git init
Initialized empty Git repository in C:/Users/Cristi/Desktop/pr2/lab1/.git/
```

Figura 2 – Initializare Git

Pentru a verifica cum sistemul funcționează am creat în directorul nou două fișier apoi am testat instrucțiunea *git status* (vezi în figura 3).

```
Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ touch file1.txt

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ touch file2.txt

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git status
On branch master

Initial commit

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        file1.txt
        file2.txt

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

Figura 3 – git status

Modificarea repozitoriului local se face prin instrucțiunea *git add* și *git commit*. Pentru ca aceste modificări să ajungă pe repozitoriul distant este nevoie de a testa instrucțiunea *git remote add origin git@gitlab.ati.utm.md:username/projectname.git* și respectiv *git push* (vezi în figura 4).

```

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git add .

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git commit -m "add file1 and file2 master"
[master (root-commit) 6ba5724] add file1 and file2 master
Committer: Olaru Cristi <Olaru Cristi>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file1.txt
create mode 100644 file2.txt

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git remote add origin https://github.com/olarucristian/pr.git

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git push -u origin master
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 218 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/olarucristian/pr.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.

```

Figura 4 – git add , commit.

Crearea unei ramure și trecerea de la una la alta se face cu ajutorul instrucțiunii *git branch **nume ramură*** și respectiv *git checkout **nume ramură***. Pe ramura **dev** a fost create un fișier nou *file3.txt* dupa care modificări au fost salvate cu **git add .** și **git commit-m**”---” și încărcate pe repoziitoriul distant .iar pentru trage modificările din repoziitoriul distant folosim **git pull**(vezi în figura 5,6,7 și 8).

```

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git branch dev

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git checkout dev
Switched to branch 'dev'

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (dev)
$ touch file3.txt

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (dev)
$ git status
On branch dev
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    file3.txt

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

```

Figura 5 – git branch, git checkout

```

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (dev)
$ git add .

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (dev)
$ git commit -m "add file3 dev"
[dev 6d4ed88] add file3 dev
  Committer: Olaru Cristi <Olaru Cristi>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file3.txt

```

Figura 6 – git add , commit.

```

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (dev)
$ git pull origin master
From https://github.com/olarucristian/pr
 * branch                master      -> FETCH_HEAD
Already up-to-date.

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (dev)
$ git push -u origin dev
Counting objects: 2, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 234 bytes | 0 bytes/s, done.
Total 2 (delta 0), reused 0 (delta 0)
To https://github.com/olarucristian/pr.git
 * [new branch]          dev -> dev
Branch dev set up to track remote branch dev from origin.

```

Figura 7 – git pull, git push

```

Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (dev)
$ git checkout master
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.

```

Figura 8 – git checkout master

Pentru a verifica cum sistemul functioneaz  am modificat **fi ier2** de pe master apoi am testat instruc iunea *git status* , apoi a fost adaugata si comentat (vezi  n figura 9).

```
Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/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:   file2.txt

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

Figura 9 – modificarea

Iar pentru a uni dou  ramure este nevoie de a testa instruc iunea *git merge* (vezi  n figura 10).

```
Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git merge dev
Merge made by the 'recursive' strategy.
 file3.txt | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 file3.txt
```

Figura 10 – git merge

Pentru ca toate aceste modific ri s  ajung  pe repozitoriul distant este nevoie de a folosi instruc iunea *git push* (vezi  n figura 11,12).

```
Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git push -u origin master
Counting objects: 5, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 466 bytes | 0 bytes/s, done.
Total 5 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/olarucristian/pr.git
 6ba5724..e7980a7 master -> master
Branch master set up to track remote branch master from origin.
```

Figura 11 –  nc rcarea pe repozitoriul distant

4 commits		2 branches		0 releases		0 contributors	
Branch: master ▾		New pull request		Create new file	Upload files	Find file	Clone or download ▾
Olaru Cristi Merge branch 'dev'		Latest commit e7980a7 6 minutes ago					
file1.txt	add file1 and file2 master						28 minutes ago
file2.txt	file2 testing modified						8 minutes ago
file3.txt	add file3 dev						21 minutes ago

Figura 12 – rezultatele modific rii repozitoriului distant

Pentru a vizualiza graficul comiturilor este nevoie de a folosi instrucțiunea **git log --graph --all** (vezi în figura 13).

```
Cristi@DESKTOP-TTLBDV6 MINGW64 ~/Desktop/pr2/lab1 (master)
$ git log --graph --all
* commit e7980a74cf808f96444d7d1f44dabe8eed3ce8e7
  Merge: d3e8c2c 6d4ed88
  Author: Olaru Cristi <Olaru Cristi>
  Date:   Wed Mar 29 20:08:01 2017 +0300

      Merge branch 'dev'

* commit 6d4ed8858cc35b337d396547cbdceca506bda5c5
  Author: Olaru Cristi <Olaru Cristi>
  Date:   Wed Mar 29 19:52:50 2017 +0300

      add file3 dev

* commit d3e8c2ccaf1014dde10b54b7dacd50b2283a2454
  Author: Olaru Cristi <Olaru Cristi>
  Date:   Wed Mar 29 20:05:41 2017 +0300

      file2 testing modified

* commit 6ba572424688e7c669629b850fc6445d3923618e
  Author: Olaru Cristi <Olaru Cristi>
  Date:   Wed Mar 29 19:45:41 2017 +0300

      add file1 and file2 master
```

Figura 13 – git log --graph --all

Construirea proiectelor Java utilizind Apache Maven

În capitolul respectiv , va fi analizata metoda de creare a proiectelor java, utilizind apache maven . În figurile - 2.1, 2.2 observam etapele initiale de creare a proiectului utilizind IDEA.

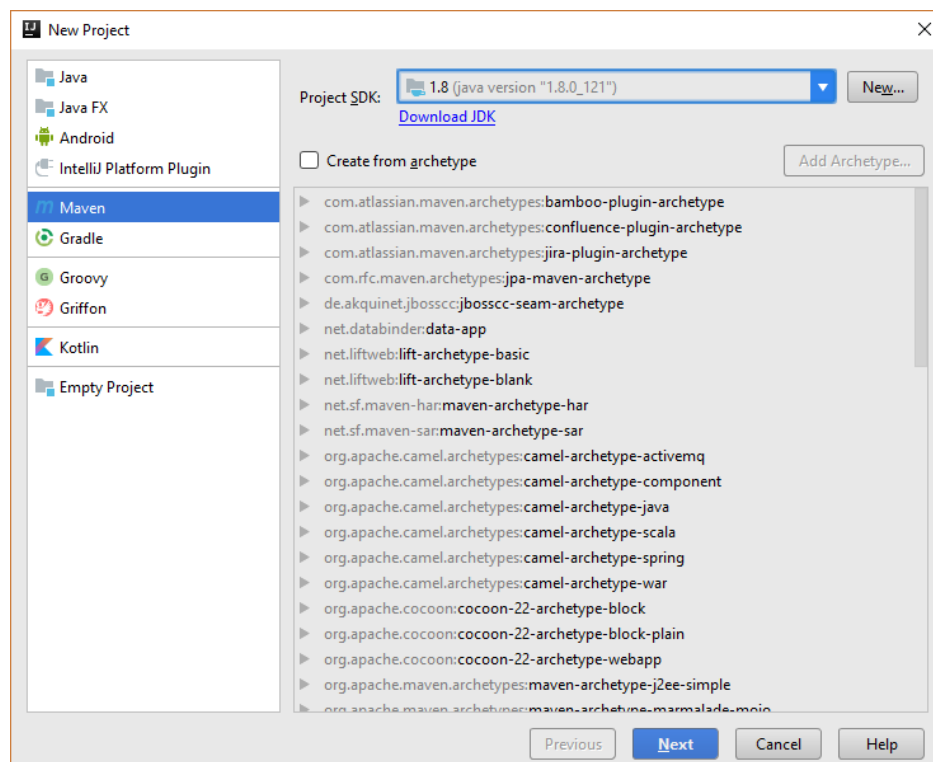


Figura 2.1 – Crearea unui proiect maven

GroupId	prlab1	<input checked="" type="checkbox"/> Inherit
ArtifactId	MAVEN app	
Version	1.0-SNAPSHOT	<input checked="" type="checkbox"/> Inherit

Figura 2.2 –GrupId,ArtifactId,Versiunea

În figura 2.3, poate fi observată descărcarea dependențelor inițiale și POM-ul Mavedeschis în editor:

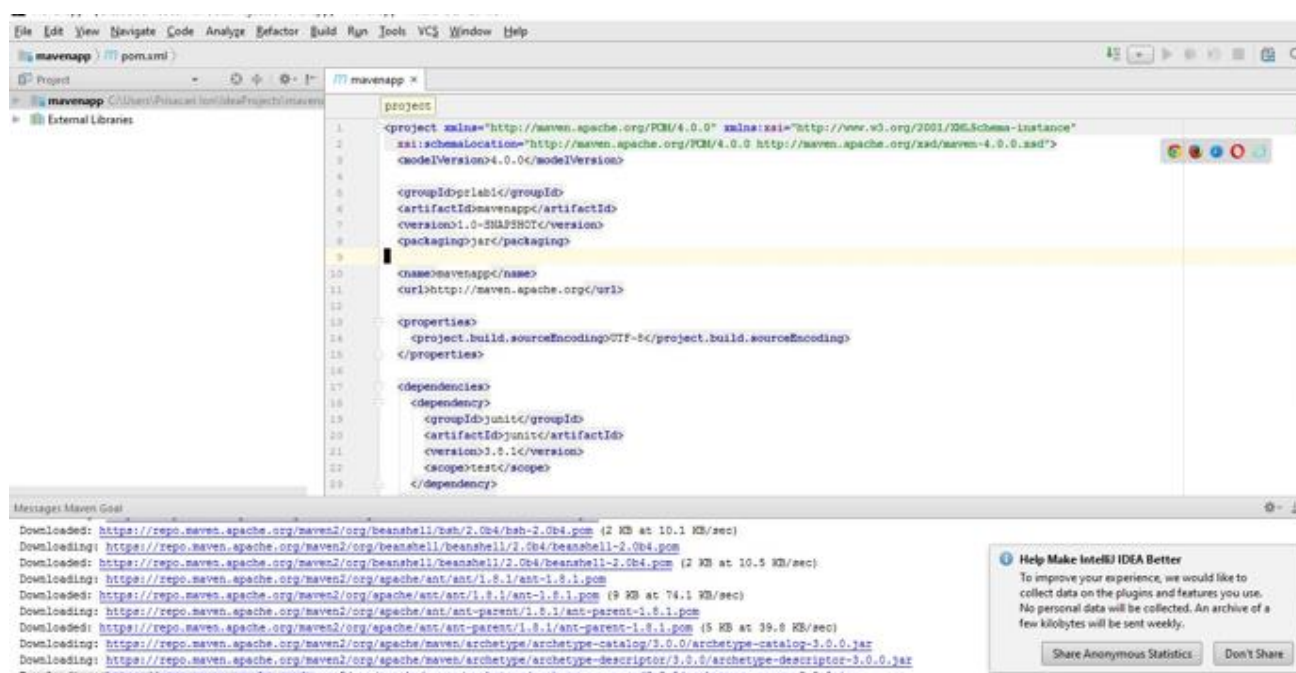


Figura 2.3 – Descărcarea dependențelor inițiale

În figura 2.4, poate fi observată configurarea *run/debug* necesară executării unui scop Maven:

<div> <div>+</div> <div>-</div> <div>📄</div> <div>✂</div> <div>↑</div> <div>↓</div> <div>📁</div> <div>🔍</div> </div> <div> <div>⚙️ Maven</div> <div>⚙️ Unnamed</div> <div>🔍 Defaults</div> </div>	<div>Name: Unnamed</div> <div> <input type="checkbox"/> Share <input type="checkbox"/> Single instance only </div> <div> <div>Parameters</div> <div>General</div> <div>Runner</div> <div>Logs</div> </div> <div>Working directory: C:\Users\Cristian\IdeaProjects\mavenapp</div> <div>Command line: clean install</div> <div>Profiles (separated with space):</div> <div>add prefix '-' to disable profile, e.g. "-test"</div> <div><input type="checkbox"/> Resolve Workspace artifacts</div>
---	--

În figura 2.4, poate fi observată configurarea *run/debug* necesară executării unui scop Maven:

În scopul verificării corectitudinii creării proiectului, a fost rulat un simplu *Hello World!*, ce se observă în figura 2.5:

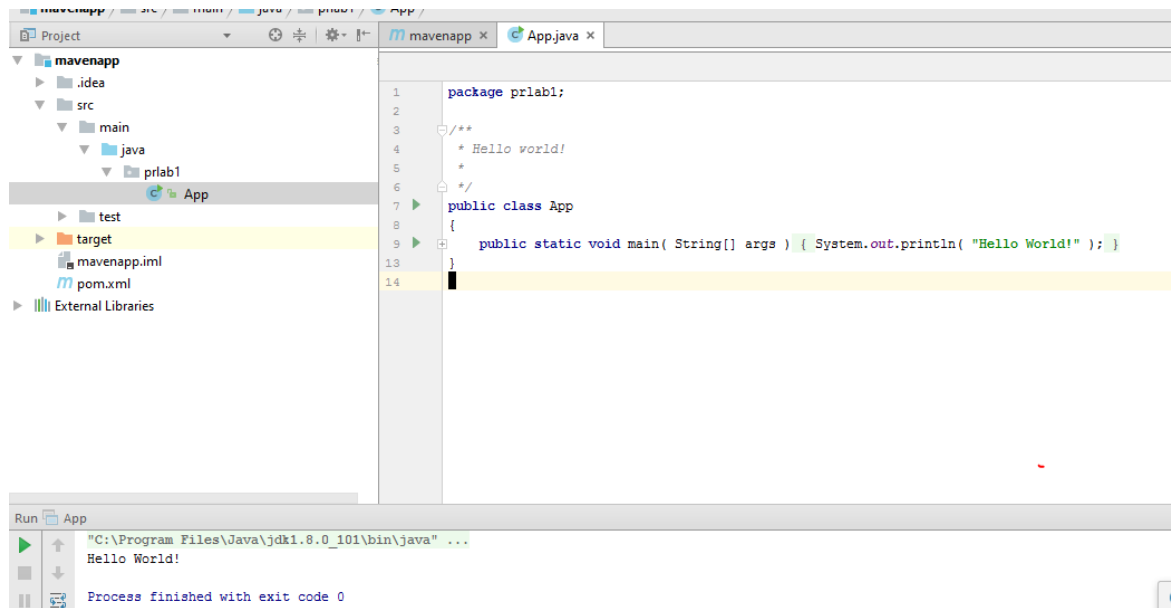


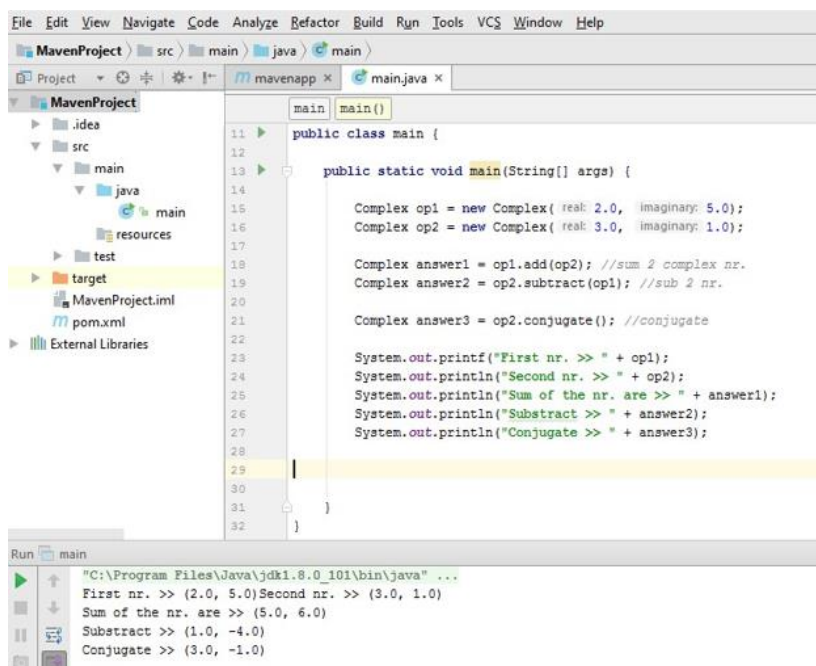
Figura 2.5 – Testarea proiectului

În scopul realizării lucrării de laborator, a fost utilizată dependența *commons-math* 3.2, ce a fost inclusă în proiect prin intermediul POM-ului (figura 2.6). Librăria respectivă include un set vast de opțiuni ce permit prelucrarea matematică a datelor.



Figura 2.6 – Dependența *commons-math* 3.2

În scopul testării dependenței, au fost inițializate 2 nr. complexe și realizate careva din operațiile standard asupra nr. complexe. Rezultatul poate fi analizat în figura 2.7:



The screenshot displays an IDE with a project named 'MavenProject'. The 'src/main/java' directory contains a file 'main.java'. The code defines a 'Complex' class and a 'main' method. The 'main' method creates two complex numbers: 'op1' (2.0 + 5.0i) and 'op2' (3.0 + 1.0i). It then performs three operations: addition (resulting in 5.0 + 6.0i), subtraction (resulting in 1.0 - 4.0i), and conjugation (resulting in 3.0 - 1.0i). The results are printed to the console. The 'Run' tab at the bottom shows the output of the program.

```
11 public class main {
12
13     public static void main(String[] args) {
14
15         Complex op1 = new Complex( real: 2.0, imaginary: 5.0);
16         Complex op2 = new Complex( real: 3.0, imaginary: 1.0);
17
18         Complex answer1 = op1.add(op2); //sum 2 complex nr.
19         Complex answer2 = op2.subtract(op1); //sub 2 nr.
20
21         Complex answer3 = op2.conjugate(); //conjugate
22
23         System.out.printf("First nr. >> " + op1);
24         System.out.println("Second nr. >> " + op2);
25         System.out.println("Sum of the nr. are >> " + answer1);
26         System.out.println("Substract >> " + answer2);
27         System.out.println("Conjugate >> " + answer3);
28
29     }
30
31 }
32 }
```

Run main

```
"C:\Program Files\Java\jdk1.8.0_101\bin\java" ...
First nr. >> (2.0, 5.0)Second nr. >> (3.0, 1.0)
Sum of the nr. are >> (5.0, 6.0)
Substract >> (1.0, -4.0)
Conjugate >> (3.0, -1.0)
```

Figura 2.7 – Rezultatul

Concluzie

În această lucrare de laborat am însușit modul de lucru cu sistemul de control al versiunelor Git. Am creat un repozitoriu local și unul distant făcând schimb de commit-uri între ele prin instrucțiunea *push* și *pull*. Conform condiției am creat o două ramuri prin instrucțiunea *branch* și le-am unit prin *merge*. Efectuând această lucrare de laborator am conștientizat importanța sistemelor control a versiunelor, ele fiind un lucru indispensabil pentru programatori.

```
import org.apache.commons.math3.linear.RealMatrix;
import org.apache.commons.math3.random.CorrelatedRandomVectorGenerator;
import org.apache.commons.math3.random.GaussianRandomGenerator;
import org.apache.commons.rng.UniformRandomProvider;
import org.apache.commons.rng.simple.RandomSource;

public class main {

    public static void main(String[] args) {

        Complex op1 = new Complex(2.0, 5.0);
        Complex op2 = new Complex(3.0, 1.0);

        Complex answer1 = op1.add(op2); //sum 2 complex nr.
        Complex answer2 = op2.subtract(op1); //sub 2 nr.

        Complex answer3 = op2.conjugate(); //conjugate

        System.out.printf("First nr. >> " + op1);
        System.out.println("Second nr. >> " + op2);
        System.out.println("Sum of the nr. are >> " + answer1);
        System.out.println("Substract >> " + answer2);
        System.out.println("Conjugate >> " + answer3);

    }
}
```