Universitatea Tehnică a Moldovei

Departamentul Ingineria Software si Automatică

**RAPORT**

Lucrare de laborator Nr.1

Tehnologii si Instrumente de dezvoltare a PP*.*

A efectuat: st. gr. TI-161, Ciobanu Alina

A verificat: lect. univ. [Cristian Rusu](https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=9&cad=rja&uact=8&ved=2ahUKEwiUn7HI7LDdAhVoh4sKHXz8D9YQFjAIegQIBRAB&url=https%3A%2F%2Fwww.facebook.com%2Fpublic%2FCristian-Rusu&usg=AOvVaw2RlhYBAtcChwVbsLEB4h-C)

.

Chișinău 2018

**Tema:** Versionarea codului sursa utilizand GIT

**Scop**: Studiul și înțelegerea principiilor de funcționare și utilizare a sistemului distribuit de control al versiunilor numit GIT.



Fig. 1. Clonarea directory-ului de pe git cu numele ”space”



Fig. 2. Crearea unui branch din linia de comanda



Fig. 3. Introducem modificarile personale

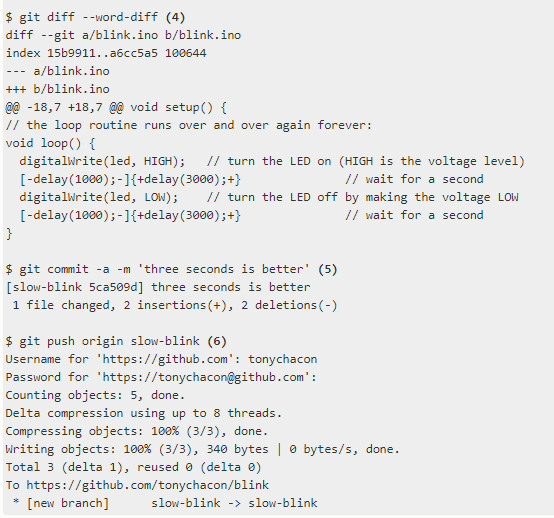


Fig. 4. Verificam daca sunt modificari

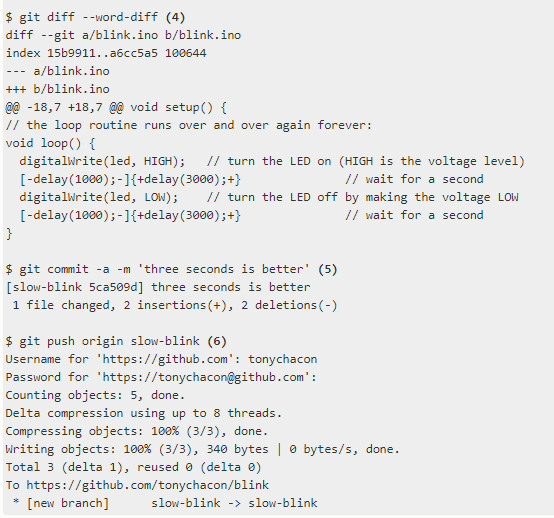


Fig. 5. Fixam modificarile in branch-ul nostru

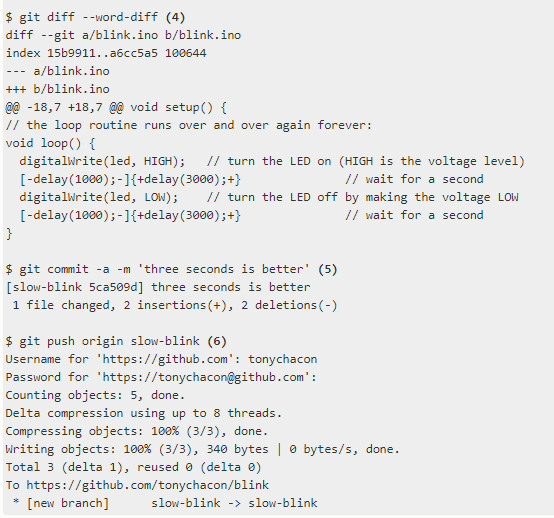


Fig.6. Transmitem datele noi in branch-ul creat anterior

$ git remote add upstream https://github.com/schacon/blink **(1)**

$ git fetch upstream **(2)**

remote: Counting objects: 3, done.

remote: Compressing objects: 100% (3/3), done.

Unpacking objects: 100% (3/3), done.

remote: Total 3 (delta 0), reused 0 (delta 0)

From https://github.com/schacon/blink

\* [new branch] master -> upstream/master

$ git merge upstream/master **(3)**

Auto-merging blink.ino

CONFLICT (content): Merge conflict in blink.ino

Automatic merge failed; fix conflicts and then commit the result.

$ vim blink.ino **(4)**

$ git add blink.ino

$ git commit

[slow-blink 3c8d735] Merge remote-tracking branch 'upstream/master' \

into slower-blink

$ git push origin slow-blink **(5)**

Counting objects: 6, done.

Delta compression using up to 8 threads.

Compressing objects: 100% (6/6), done.

Writing objects: 100% (6/6), 682 bytes | 0 bytes/s, done.

Total 6 (delta 2), reused 0 (delta 0)

To https://github.com/tonychacon/blink

ef4725c..3c8d735 slower-blink -> slow-blink

1. Add the original repository as a remote named “upstream”
2. Fetch the newest work from that remote
3. Merge the main branch into your topic branch
4. Fix the conflict that occured
5. Push back up to the same topic branch

**Concluzie**:

În urma efectuării lucrării date de laborator am facuta cunostință cu instrumentul gitHub, s-a facut cunostinta cu modul de instalare Git. Am cunoscut o serie de comenzi utile în lucru cu versificarea proiectului cum ai fi: **git init, git status, git merge, commit –m, gitt add, git push origin master, git pull, git diff, git checkout –b, git clone.**