

Министерство Образования Республики Молдова
Технический Университет Молдовы
Кафедра Автоматики и Информационных Технологий

Лабораторная работа №2

По дисциплине: «MIDPS»

Тема: «Setting server environment. Version Control Systems»

Выполнил:

студент группы ТІ-145:
Куликов Игорь

Проверил:

старший преподаватель:
Кожокару Светлана

Цель работы:

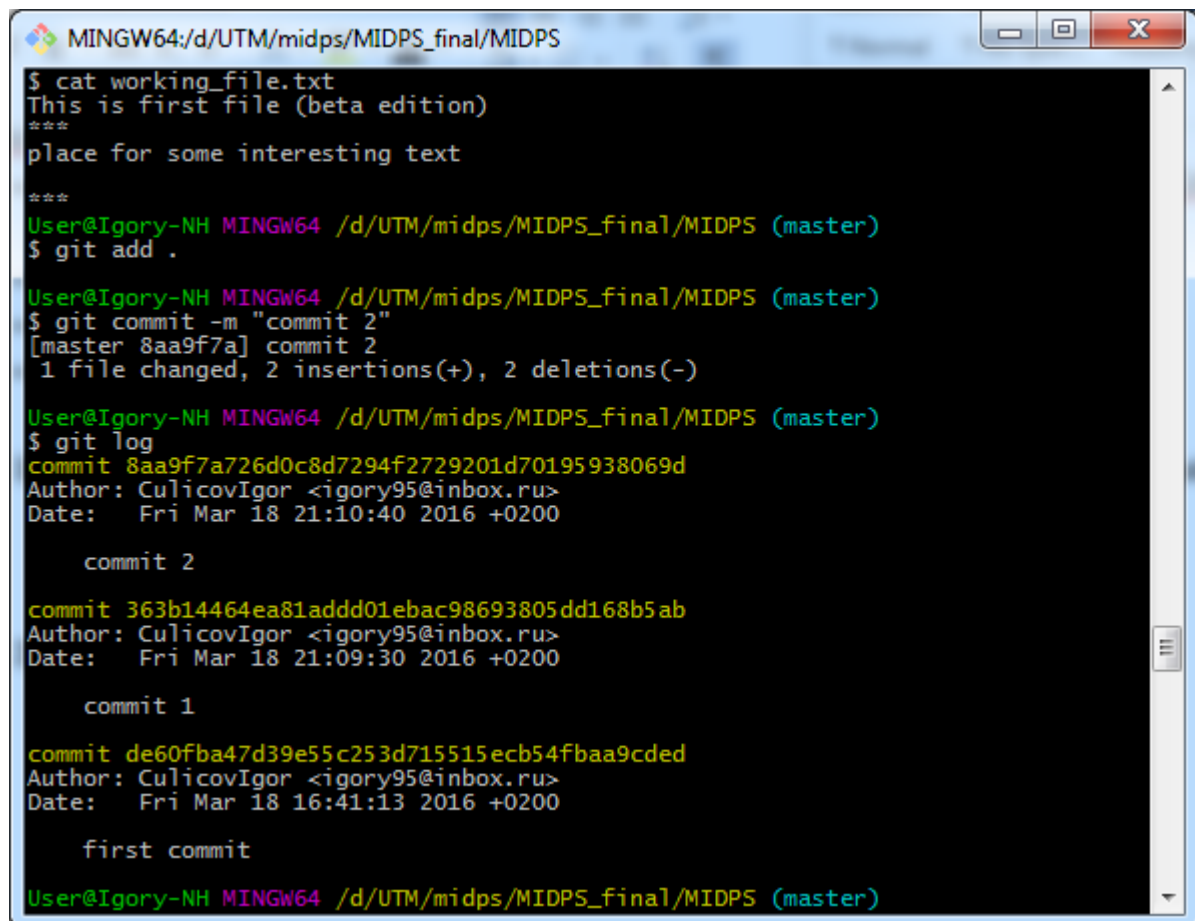
- Intelegerea si folosirea CLI (basic level)
- Administrarea remote a masinilor linux machine folosind SSH (remote code editing)
- Version Control Systems (git || mercurial || svn)
- Compileaza codul C/C++/Java/Python prin intermediul CLI, folosind compilatoarele gcc/g++/javac/python

Задание к работе:

Advanced Level (nota 9 || 10):

- 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
- merge 2 branches
- conflict solving between 2 branches

- 1) В созданном нами репозитории создаем текстовый файл и записываем в него какой-нибудь текст и вносим данные изменения с помощью команд add * и commit.



```
MINGW64:/d/UTM/midps/MIDPS_final/MIDPS
$ cat working_file.txt
This is first file (beta edition)
***
place for some interesting text
***
User@IgorY-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (master)
$ git add .
User@IgorY-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (master)
$ git commit -m "commit 2"
[master 8aa9f7a] commit 2
1 file changed, 2 insertions(+), 2 deletions(-)
User@IgorY-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (master)
$ git log
commit 8aa9f7a726d0c8d7294f2729201d70195938069d
Author: CulicovIgor <igory95@inbox.ru>
Date: Fri Mar 18 21:10:40 2016 +0200

    commit 2

commit 363b14464ea81addd01ebac98693805dd168b5ab
Author: CulicovIgor <igory95@inbox.ru>
Date: Fri Mar 18 21:09:30 2016 +0200

    commit 1

commit de60fba47d39e55c253d715515ecb54fbba9cde
Author: CulicovIgor <igory95@inbox.ru>
Date: Fri Mar 18 16:41:13 2016 +0200

    first commit
User@IgorY-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (master)
```

- 2) Далее создаем новую ветку mybranch с помощью команды checkout, в которую будут включены все наши предыдущие изменения.

```
MINGW64:/d/UTM/midps/MIDPS_final/MIDPS
User@Igor-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (master)
$ git branch mybranch

User@Igor-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (master)
$ git checkout mybranch
Switched to branch 'mybranch'

User@Igor-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (mybranch)
$ git log
commit 8aa9f7a726d0c8d7294f2729201d70195938069d
Author: CulicovIgor <igory95@inbox.ru>
Date: Fri Mar 18 21:10:40 2016 +0200

    commit 2

commit 363b14464ea81add01ebac98693805dd168b5ab
Author: CulicovIgor <igory95@inbox.ru>
Date: Fri Mar 18 21:09:30 2016 +0200

    commit 1

commit de60fba47d39e55c253d715515ecb54fbba9cded
Author: CulicovIgor <igory95@inbox.ru>
Date: Fri Mar 18 16:41:13 2016 +0200

    first commit
```

- 3) С помощью команды reset откатываем наше последнее изменение. Теперь в ветке mybranch 2 commit-а, а в ветке master 3. Это сделано для последующего возникновения конфликта при слиянии 2 веток.

```
MINGW64:/d/UTM/midps/MIDPS_final/MIDPS
User@Igor-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (mybranch)
$ git reset --hard HEAD~1
HEAD is now at 363b144 commit 1

User@Igor-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (mybranch)
$ git log
commit 363b14464ea81add01ebac98693805dd168b5ab
Author: CulicovIgor <igory95@inbox.ru>
Date: Fri Mar 18 21:09:30 2016 +0200

    commit 1

commit de60fba47d39e55c253d715515ecb54fbba9cded
Author: CulicovIgor <igory95@inbox.ru>
Date: Fri Mar 18 16:41:13 2016 +0200

    first commit
```

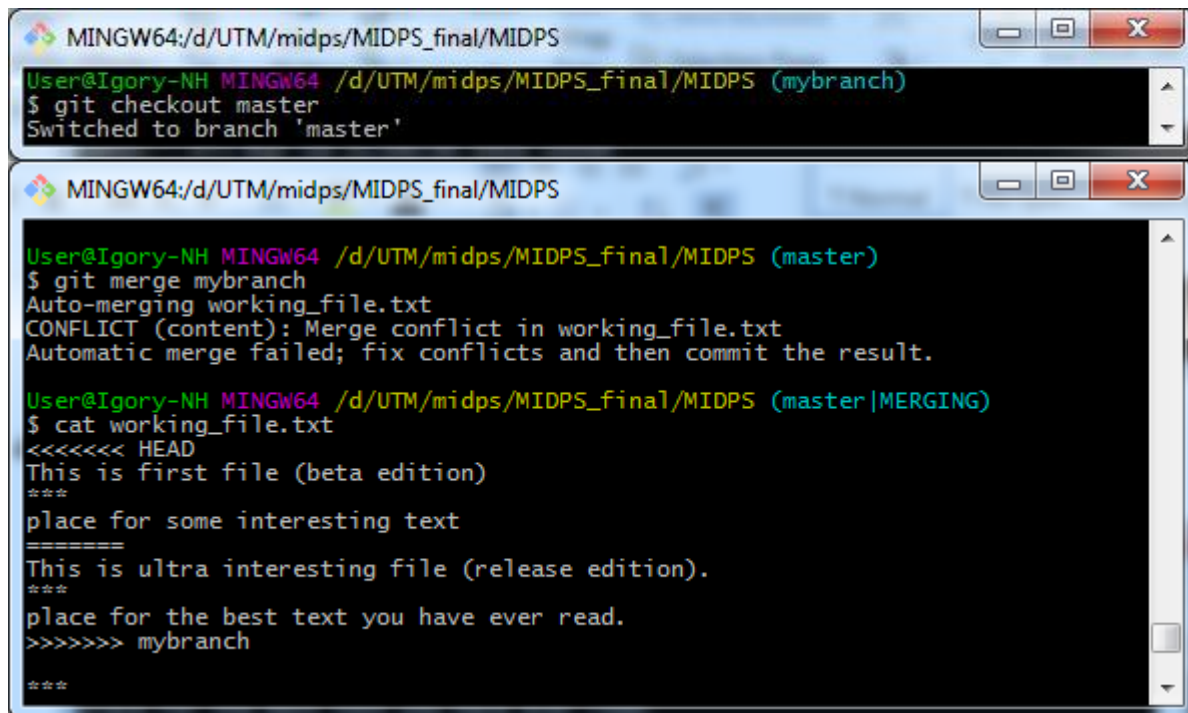
- 4) Теперь в наш текстовый файл вносим другие изменения которые и приведут к возникновению конфликта.

```
MINGW64:/d/UTM/midps/MIDPS_final/MIDPS
User@Igor-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (mybranch)
$ cat working_file.txt
This is ultra interesting file (release edition).
***
place for the best text you have ever read.
***

User@Igor-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (mybranch)
$ git add .

User@Igor-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (mybranch)
$ git commit -m "conflict commit"
[mybranch 8009c3e] conflict commit
1 file changed, 2 insertions(+), 2 deletions(-)
```

- 5) Переходим в ветку master и пытаемся объединить mybranch и master ветки. При попытке объединения нам говорится о возникновении конфликта в нашем текстовом файле. Git помечает в каком месте у нас возникает конфликт.

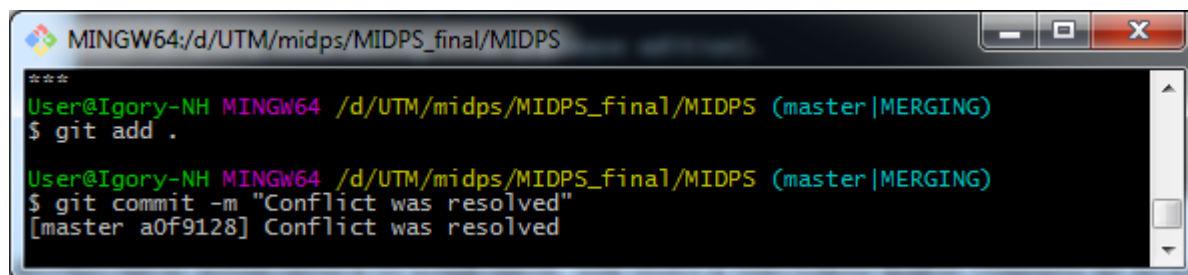


```
MINGW64:/d/UTM/midps/MIDPS_final/MIDPS
User@IgorY-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (mybranch)
$ git checkout master
Switched to branch 'master'

MINGW64:/d/UTM/midps/MIDPS_final/MIDPS
User@IgorY-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (master)
$ git merge mybranch
Auto-merging working_file.txt
CONFLICT (content): Merge conflict in working_file.txt
Automatic merge failed; fix conflicts and then commit the result.

User@IgorY-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (master|MERGING)
$ cat working_file.txt
<<<<<< HEAD
This is first file (beta edition)
***
place for some interesting text
=====
This is ultra interesting file (release edition).
***
place for the best text you have ever read.
>>>>>> mybranch
***
```

- 6) Вручную избавляемся от конфликта, для этого в текстовом файле убираем не нужную нам часть и сохраняем его. И с помощью команд add, commit и push добавляем эти изменения в наш репозиторий. Конфликт был разрешен.



```
MINGW64:/d/UTM/midps/MIDPS_final/MIDPS
***
User@IgorY-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (master|MERGING)
$ git add .

User@IgorY-NH MINGW64 /d/UTM/midps/MIDPS_final/MIDPS (master|MERGING)
$ git commit -m "Conflict was resolved"
[master a0f9128] Conflict was resolved
```

Выводы

Выполняя данную лабораторную работу, пришлось ознакомиться с принципами и методами работы веб-сервиса для хостинга IT-проектов и их совместной разработки - GitHub. Для этого изначально был создан репозиторий, в который в последующем был загружен текстовый файл. Для ознакомления с работой веток в данном сервисе была создана ветка, в которой необходимо было откатить последнее изменение, для дальнейшего слияния этой ветки с другой веткой и возникновения конфликта, который необходимо решить. Что и было сделано выше. В завершение могу добавить, что это удобный веб-сервис, который в будущем может пригодится мне как программисту. Он достаточно понятен и легок в использовании не смотря на его большой функционал.

Библиография:

- 1) <http://rogerdudler.github.io/git-guide/>
- 2) https://githowto.com/ru/resolving_conflicts?ref=driverlayer.com