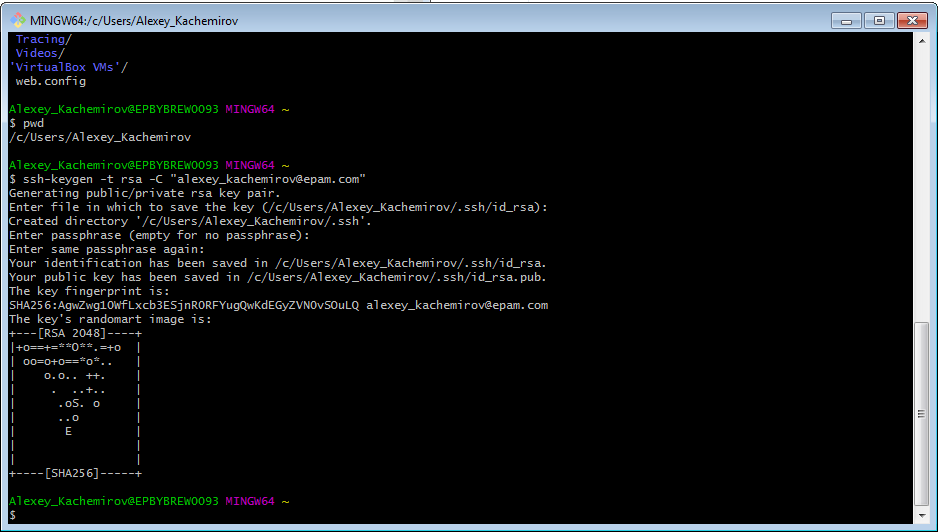
## task 1

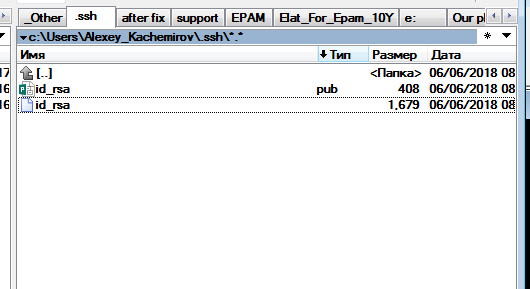
**Firstly I created SSH key in command line in Git Bash :**

Used commands:

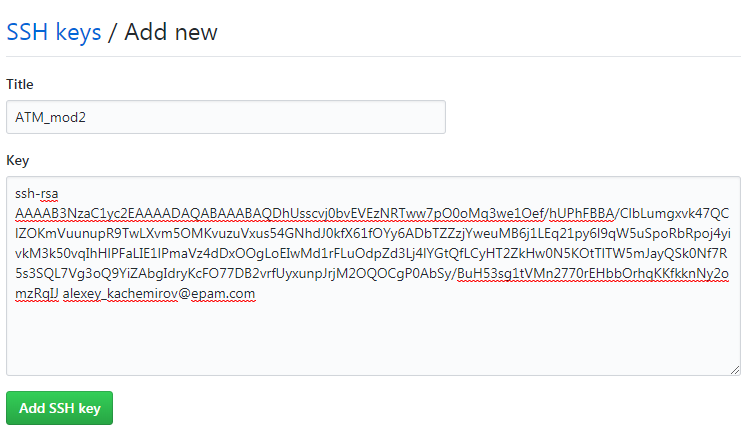
*ssh-keygen -t rsa -C "alexey\_kachemirov@epam.com"*



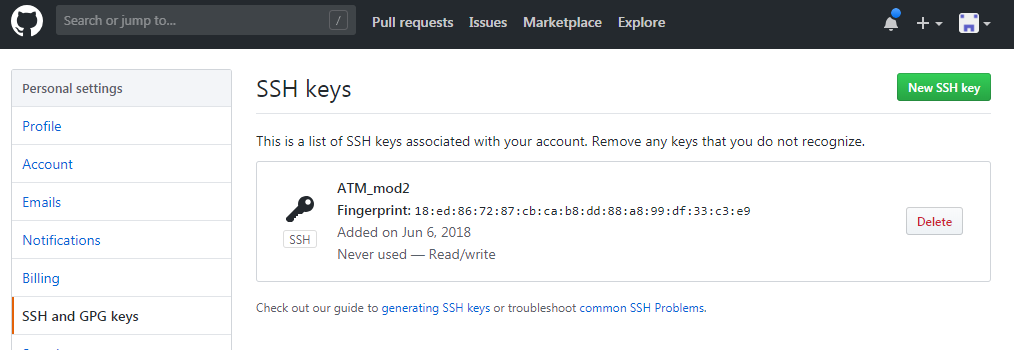
**Private and public keys were created:**



**Then I added public key in my github account:**



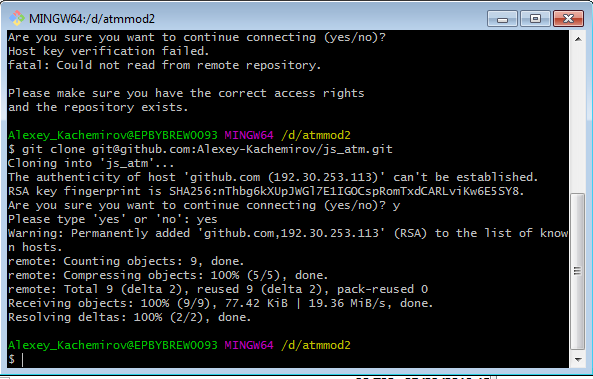
**Key was added:**



**After that I cloned my existing remote repository using SSH key:**

Used commands:

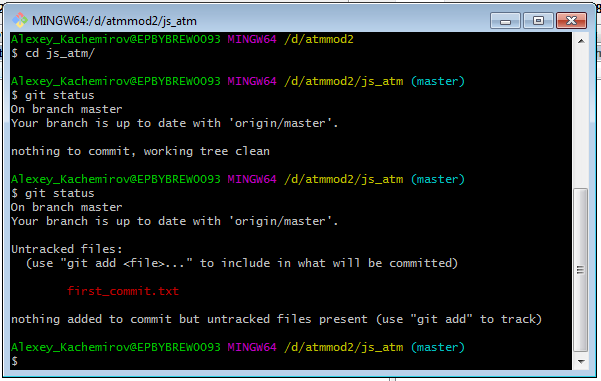
*git clone git@github.com:Alexey-Kachemirov/js\_atm.git*



**Finally I could start to make commits. I added new ‘txt’ file and checked for status:**

Used commands:

*git status*

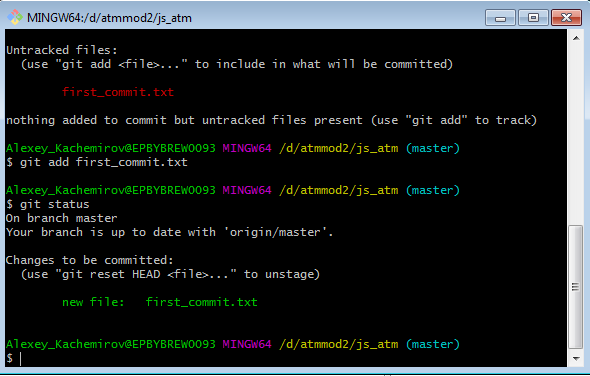


**Added to staging area (Index):**

Used commands:

*git add first\_commit.txt*

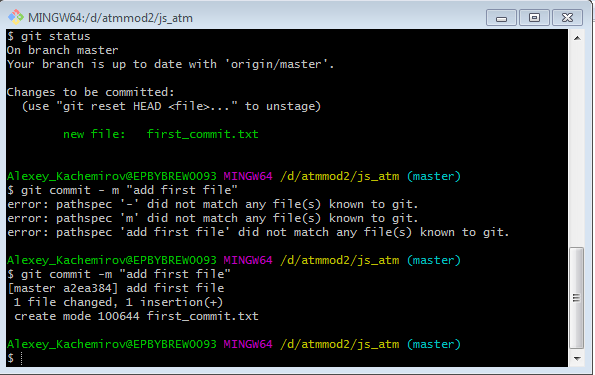
*git status*



**Committed 1st file:**

Used commands:

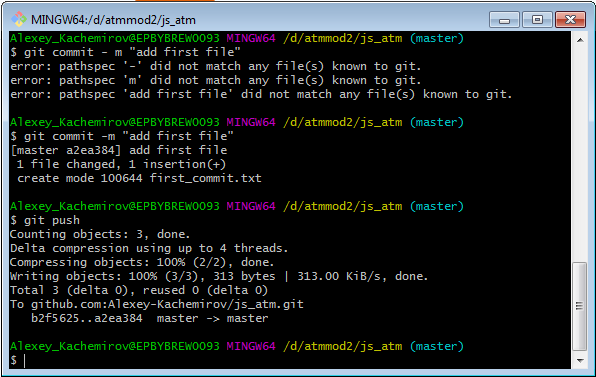
*git commit -m “add first file”*



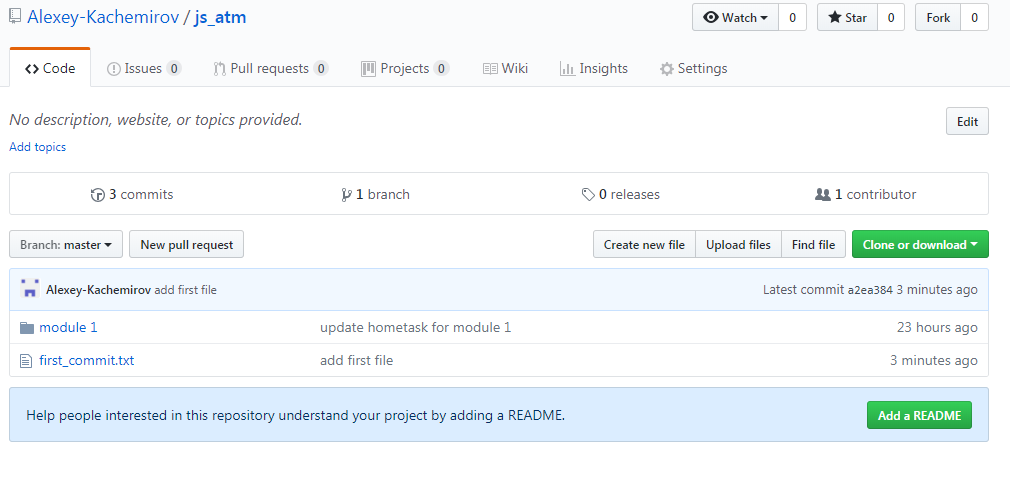
**And pushed to remote repo:**

Used commands:

*git push*



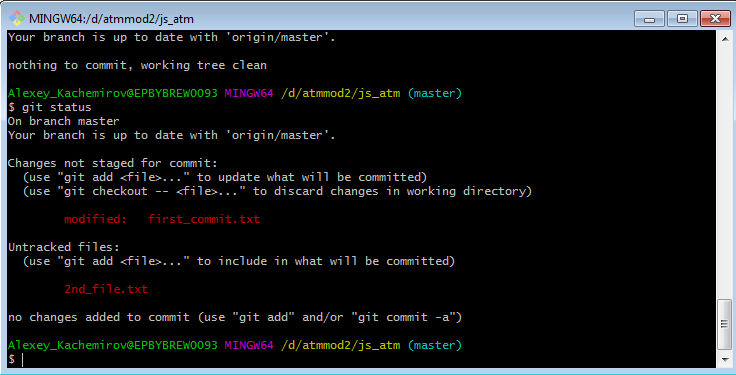
**Then I updated my remote repository and could observe my committed file:**



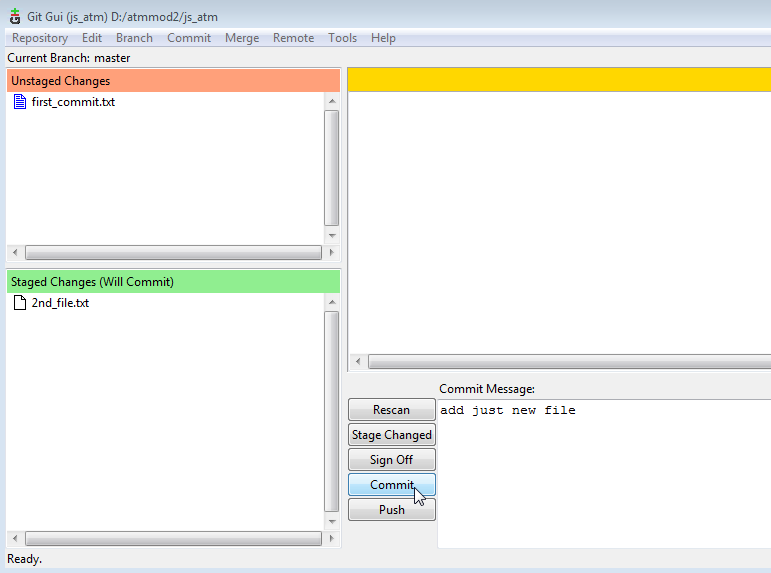
**In the 2nd commit I added one more file. Additionally I made some changes in existing file but didn’t include it in commit:**

Used commands:

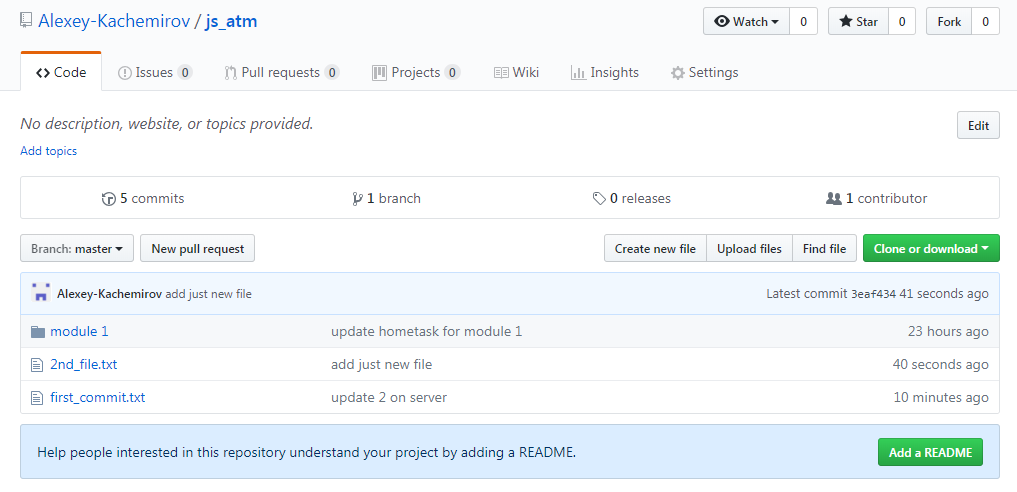
*git status*



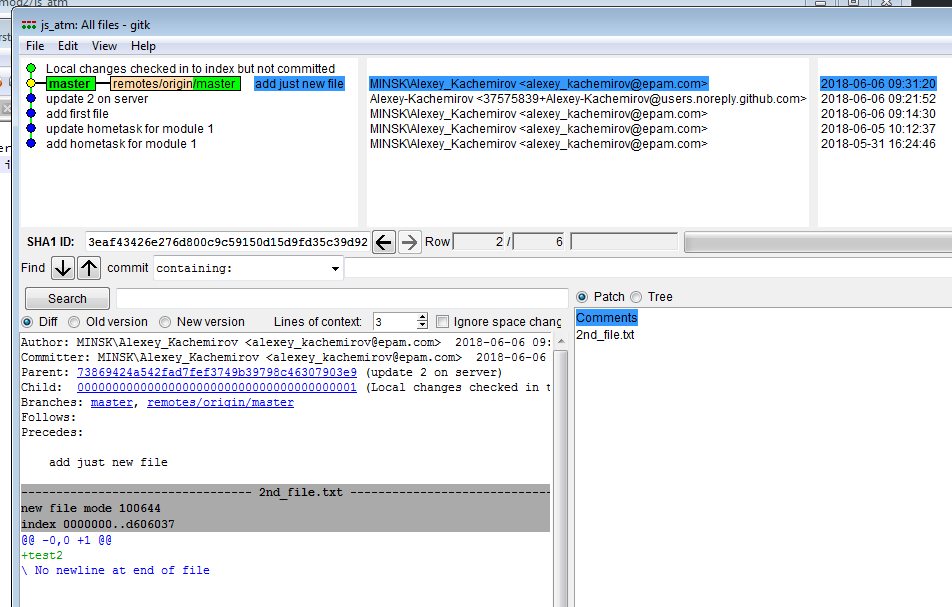
**For commit operation I used Git GUI:**



**After “push” command it appeared on remote repository:**



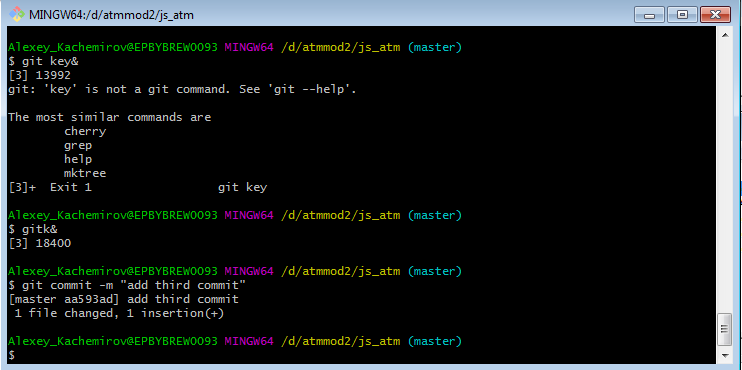
**I could track changes in Git CMD:**



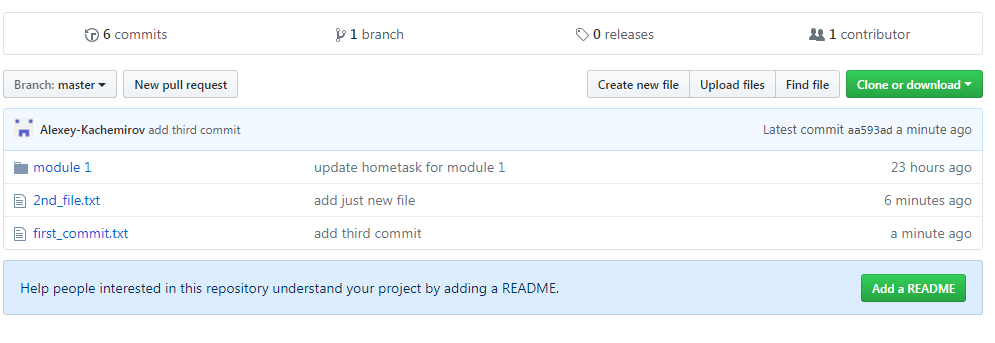
**At last I made the 3rd commit:**

Used commands:

*git commit –m “add third commit”*



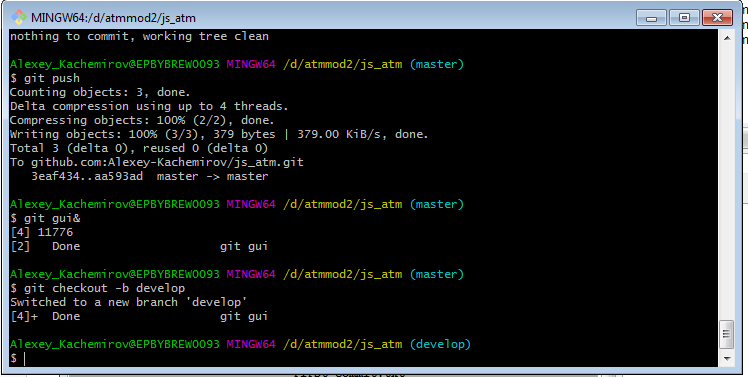
**Pushed it and refreshed repository:**



**Then I created new branch “develop” and navigated to it:**

Used commands:

*git checkout –b develop*



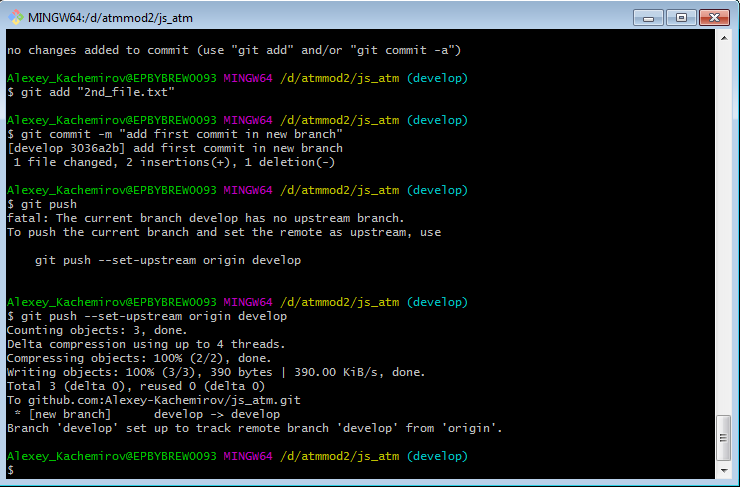
**I made 1st commit in “develop” branch:**

Used commands:

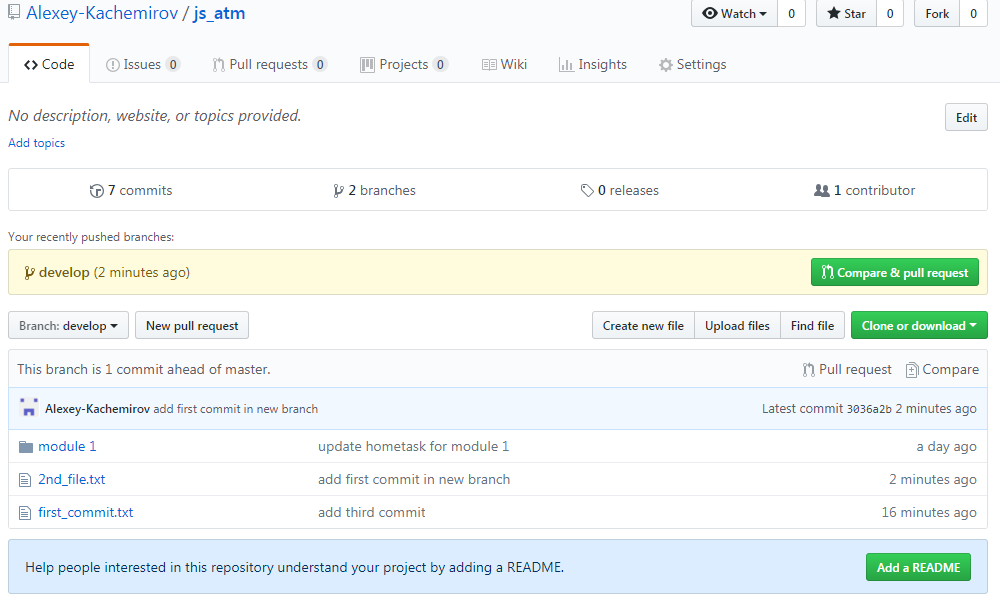
*git add*

*git commit –m “add first commit in new branch”*

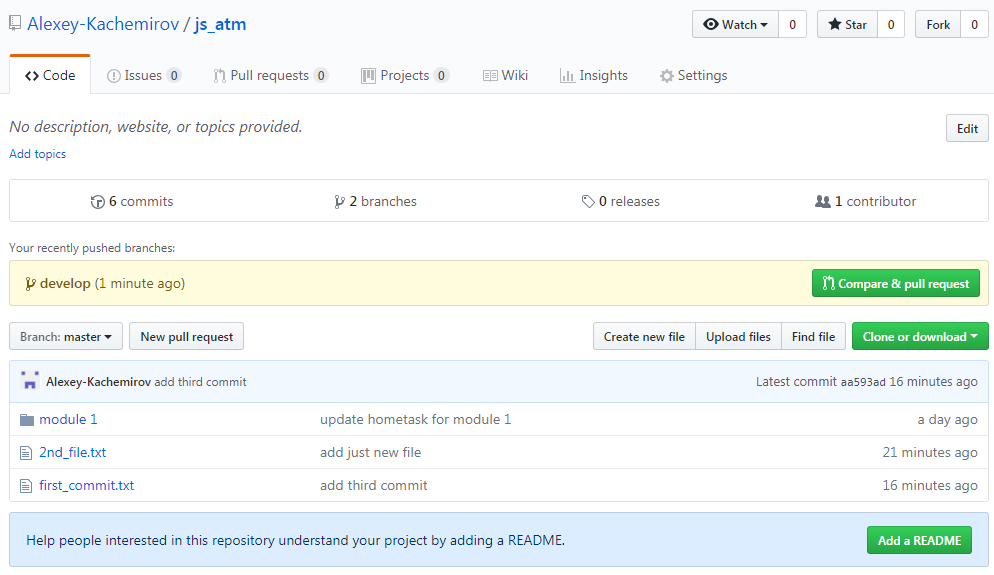
*git push –set-upstream origin develop*



**And pushed it to repository. You can see that number of commits in “master” branch and “develop” branch is not the same. “develop” branch has 7 commits:**



**And “master” branch has 6 commits as “develop” branch wasn’t merged to “master” branch.**



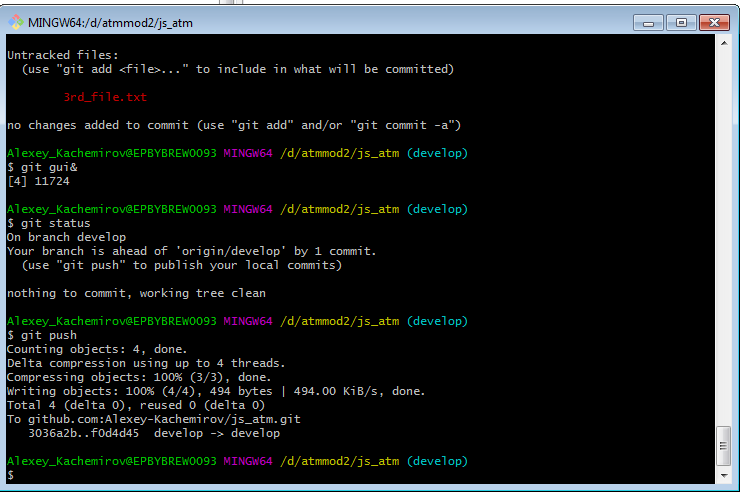
**I made 2nd commit in “develop” branch via Git GUI:**

Used commands:

*git status*

*git gui&*

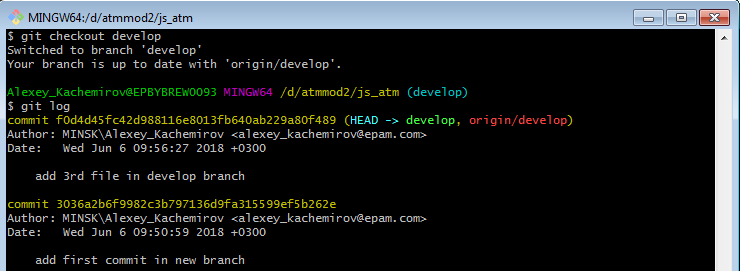
*git push*



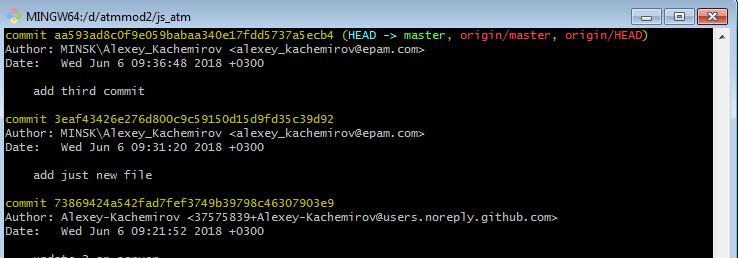
**Before merge in “master” branch I compare logs for 2 branches. “develop” branch has last commit at 09:56:27:**

Used commands:

*git log*



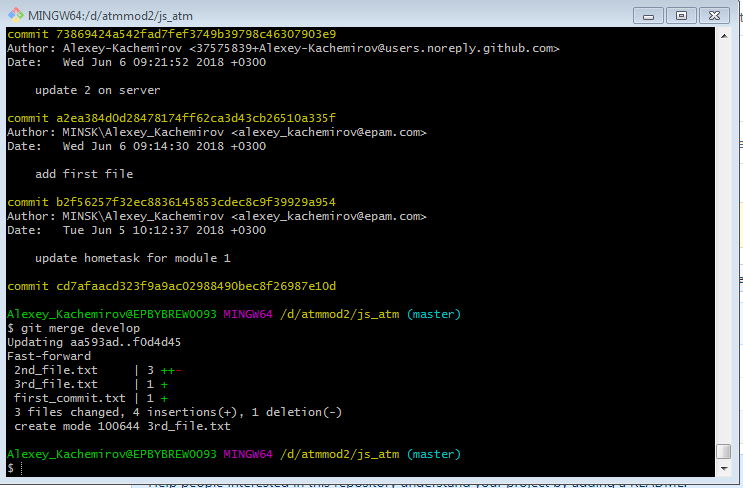
**“master” branch has last commit at 9:36:48.**



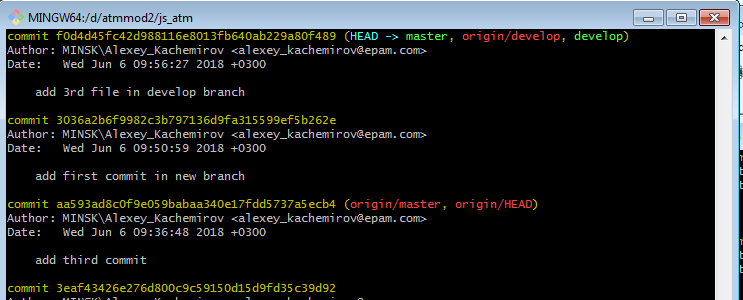
**I switched to master branch and merged “develop” branch:**

Used commands:

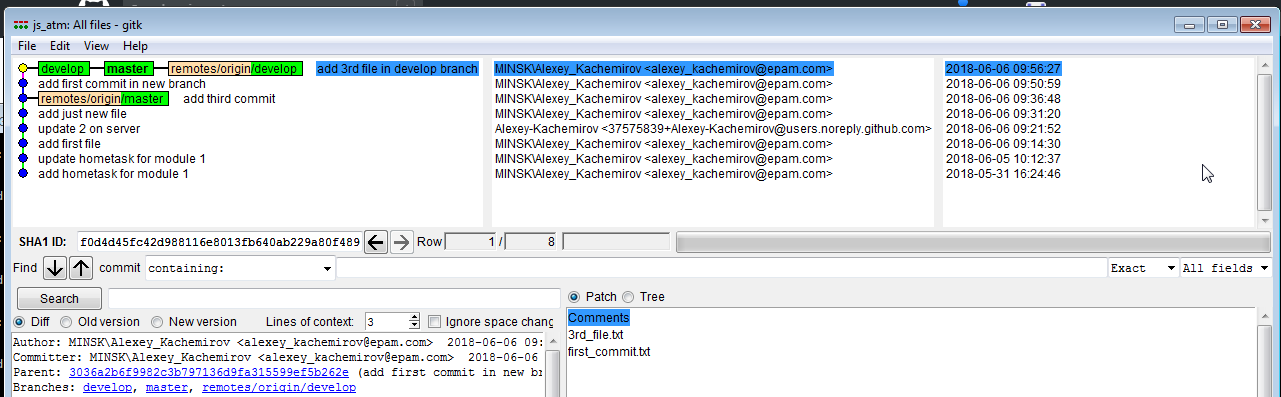
*git merge develop*



**We can check now last commits in “master” branch again. Commits from “develop” branch are available in “master” branch now:**

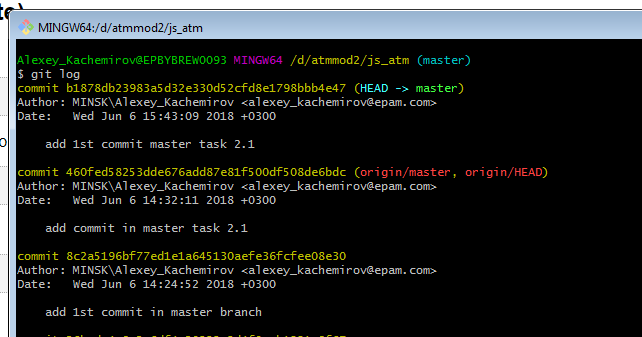


**Here are commits in Git CMD:**

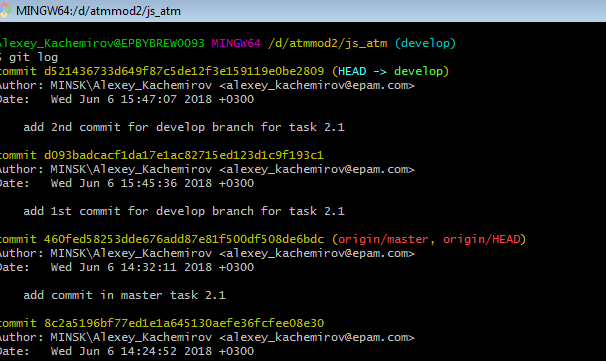


## task 2

**Created 1 commit in “master” branch. Commit time 15:43:09:**



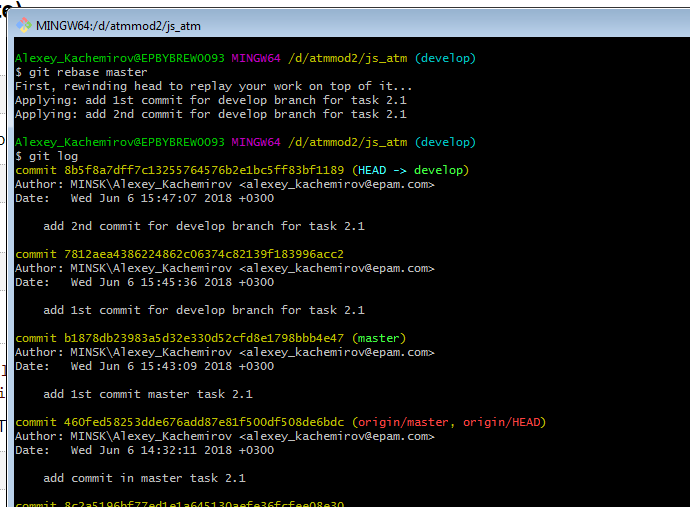
**Created 2 commits in “develop” branch. Commits time: 15:47:07 and 15:45:36**



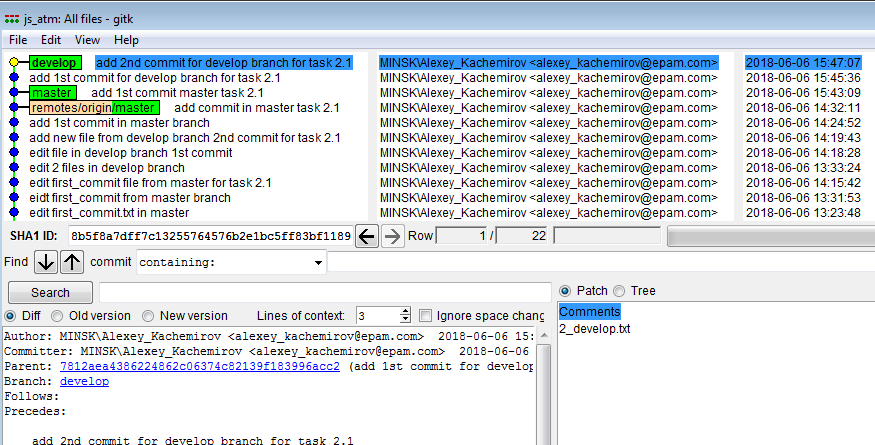
**Rebase “develop” from master and check in logs. As a result commit made in “master” branch is appearing now in “develop” branch**

Used commands:

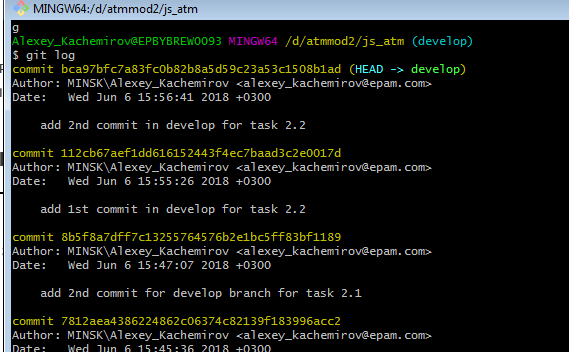
*git rebase master*



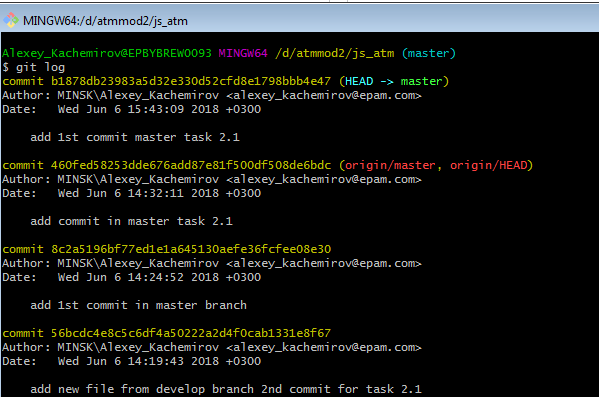
**View in Git CMD:**



**Created 2 commits in “develop” branch for cherry-pick. Commits time: 15:56:41 and 15:55:26**



**Before cherry-pick I checked last commits in “master” branch – 15:43:09**

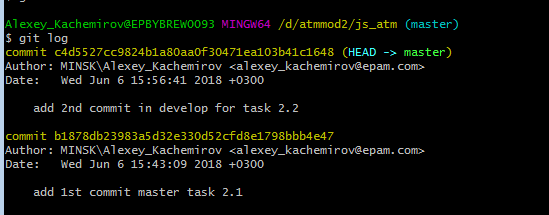


**Cherry-pick commit from “develop” to “master” and check the log. SHA1 – is hash of transferred commit from “develop” to master.**

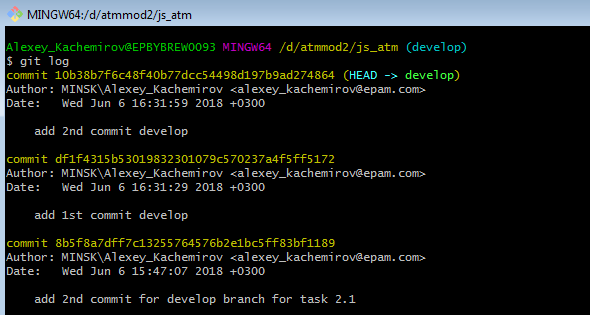
Used commands:

*git cherry-pick <sha1>*

*git log*



**Before revert commit from “develop” branch I checked the logs:**

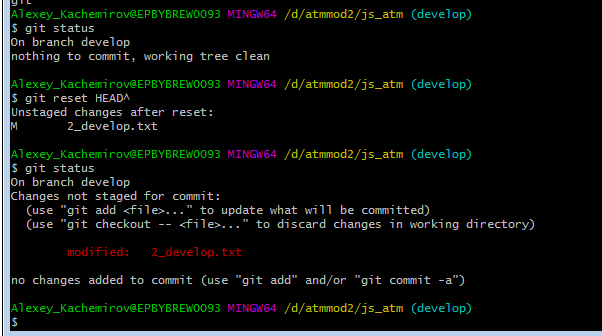


**And reverted last commit:**

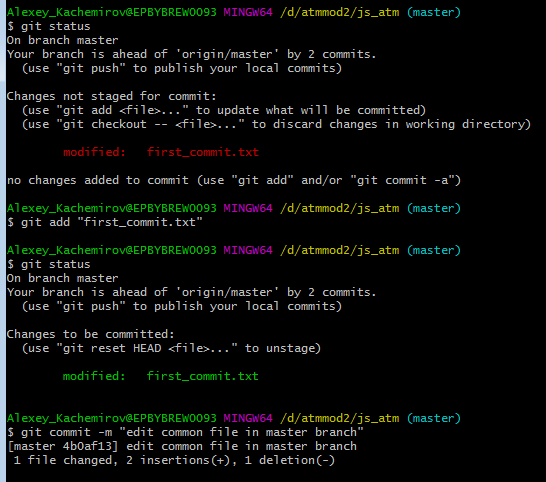
Used commands:

*git reset HEAD^*

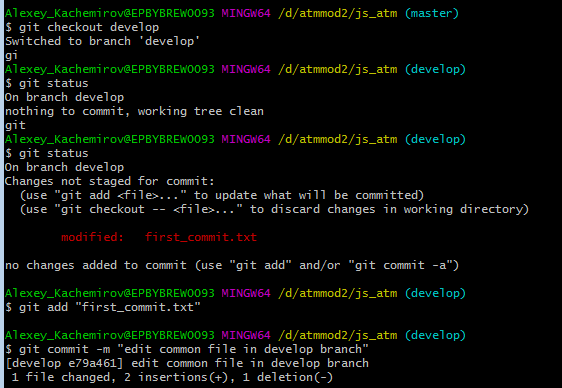
*git status*



**For generating conflict situation I edited the same file in 2 branches – “master” and “develop”. Firstly, committed in “master”:**



**Then switched to “develop” branch and committed there:**

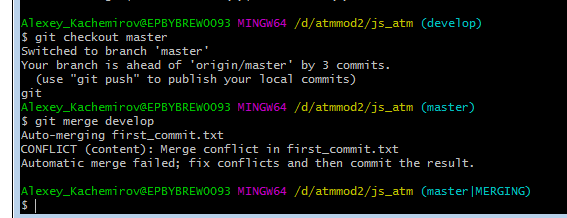


**After that I switched to “master” branch and tried to merge “develop” there:**

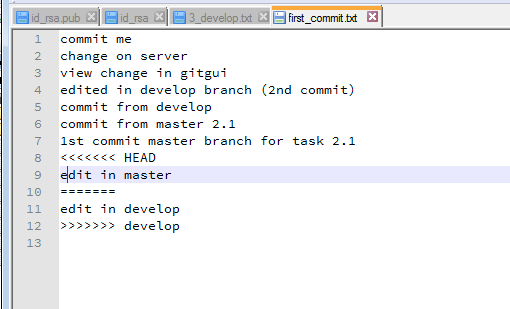
Used commands:

*git checkout master*

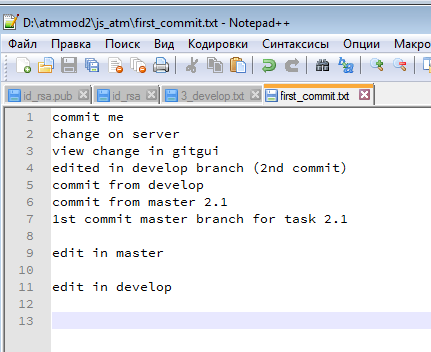
*git merge develop*



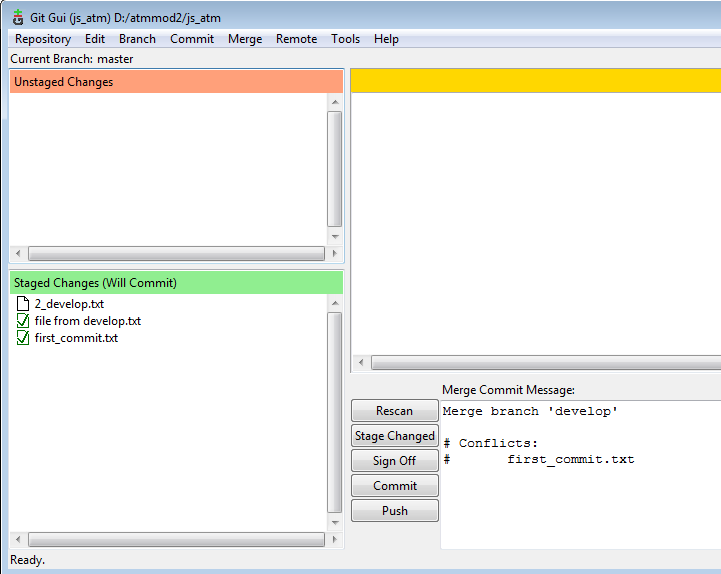
**I’ve got a conflict. “master|MERGING” note in brackets indicates that. So I opened conflict file and tried to resolve it manually. Before edit:**



**After edit:**



**Merge conflict was resolved and I committed again via Git GUI:**



**The whole picture of my actions in Git CMD:**

