Отчет по лабораторной работе №2

управление версиями

Анна Ломакина

Содержание

# Цель работы

Изучить идеологию и применение средств контроля версий.

# Задание

## 1.Настройка git

1. Создайте учётную запись на https://github.com.
2. Настройте систему контроля версий git, как это описано выше c использованием сервера репозиториев https://github.com/.
3. Создайте структуру каталога лабораторных работ согласно пункту М.2. ## 2. Подключение репозитория к github
4. Создайте репозиторий на GitHub. Для примера назовём его os-intro.
5. Рабочий каталог будем обозначать как laboratory. Вначале нужно перейти в этот каталог: cd laboratory
6. Инициализируем системы git: git init
7. Создаём заготовку для файла README.md: echo “# Лабораторные работы” >> README.md git add README.md
8. Делаем первый коммит и выкладываем на github: git commit -m “first commit” git remote add origin git@github.com:/sciproc-intro.git git push -u origin master ## 3. Первичная конфигурация
9. Добавим файл лицензии: wget https://creativecommons.org/licenses/by/4.0/legalcode.txt ↪ -O LICENSE
10. Добавим шаблон игнорируемых файлов. Просмотрим список имеющихся шаблонов: curl -L -s https://www.gitignore.io/api/list Затем скачаем шаблон, например, для C: curl -L -s https://www.gitignore.io/api/c >> .gitignore Можно это же сделать через web-интерфейс на сайте https://www.gitignore. io/.
11. Добавим новые файлы: git add .
12. Выполним коммит: git commit -a
13. Отправим на github: git push ## 4. Конфигурация git-flow
14. Инициализируем git-flow git flow init Префикс для ярлыков установим в v.
15. Проверьте, что Вы на ветке develop: git branch
16. Создадим релиз с версией 1.0.0 Кулябов Д. С. и др. Операционные системы 21 git flow release start 1.0.0
17. Запишем версию: echo “1.0.0” >> VERSION
18. Добавим в индекс: git add . git commit -am ‘chore(main): add version’
19. Зальём релизную ветку в основную ветку git flow release finish 1.0.0
20. Отправим данные на github git push –all git push –tags
21. Создадим релиз на github.

# Выполнение лабораторной работы

Создаем учётную запись на https://github.com,настраиваем систему контроля версий git,с помощью сервера репозиториев https://github.com/.Создаем структуру каталога лабораторных работ. - Cоздаем репозиторий labaOS на Github. - Рабочий каталог будем обозначать как laboratory.Вначале нужно перейти в этот каталог: cd laboratory - Инициализируем системы git:git init - Создаем заготовку для файла README.md: echo“# Лабораторные работы”>>README.md

[] (image/1.jpg)

* Делаем первый коммит и выкладываем на github



* Добавим шаблон игнорируемых файлов.Просмотрим список имеющихся шаблонов:curl -l -s ,затем скачали шаблон например для C:curl -L -s >> .gitignore.Можно это же сделать через web-интерфейс на сайте



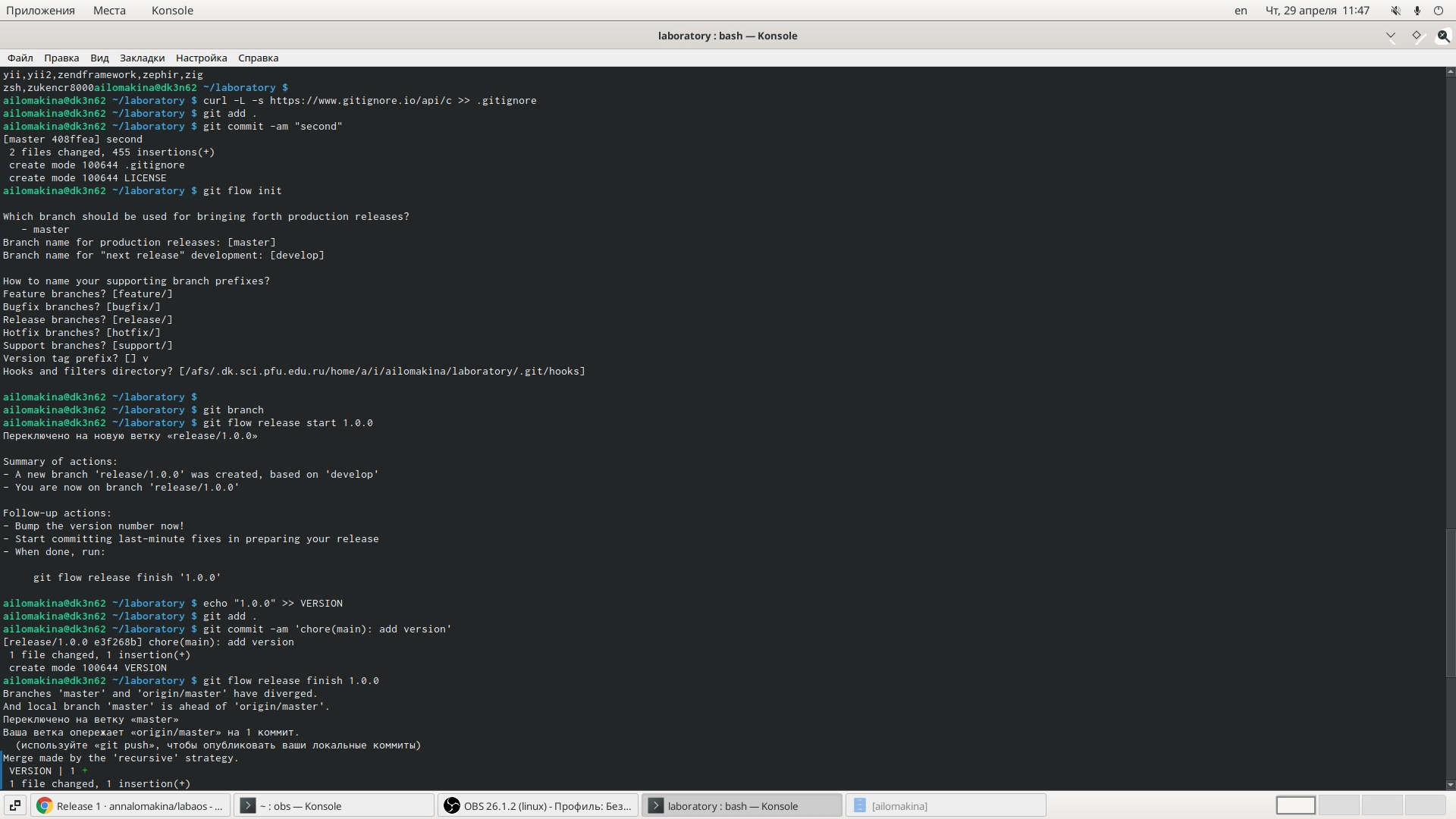
* Добавим новые файлы,выполним коммит,отправим на github



* Инициализируем git-flow
* Инициализируем git-init
* Префикс для ярлыков - v
* Создаем релиз с версией 1.0.0



* Отправляем данные на github,cоздаем релиз.



# Выводы

Я поняла как работать с репозиториями и Github.