Отчет по лабораторной работе №2

Группа НПИбд-02-22

Стариков Данила Андреевич

Содержание

# 1 Цель работы

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git.

# 2 Основная часть

## 2.1 Выполнение лабораторной работы

### 2.1.1 Настройка github

Сделали предварительную конфигурацию git: указали имя и email владельца репозитория и определили параметры (Рисунок 1).

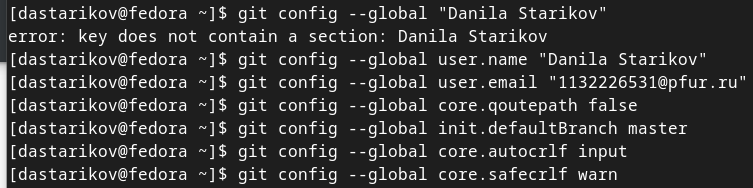


Рис. 1: Конфигурация git.

### 2.1.2 Создание SSH ключа.

Сгенерировали shh ключ для идентификации при работе с сервером репозитория с помощью команды ssh-keygen “Данила Стариков 1132226531@pfur.ru” (Рисунок 2).

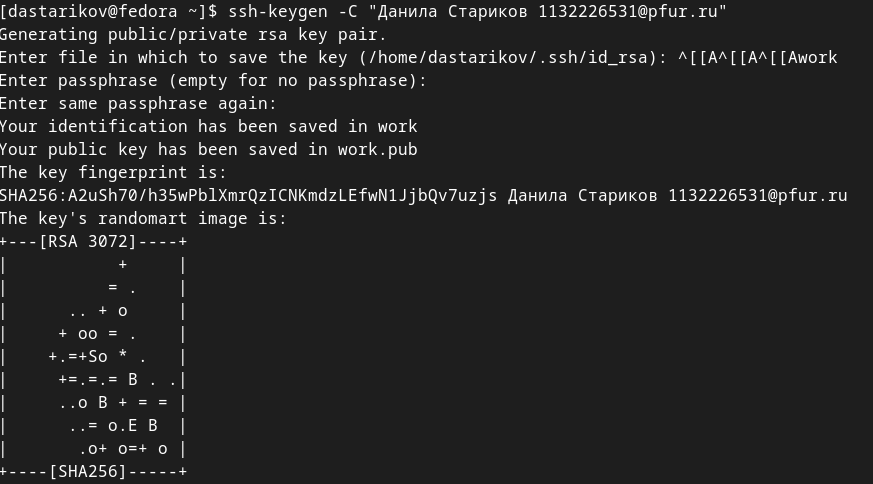


Рис. 2: Генеривание ssh ключа.

Далее сгенерированный открытый ключ необходимо загрузить на сайт <https://github.com/> в настройках учетной записи. Для этого копируем содержимое файла с ключом в буфер обмена с помощью утилиты xclip ((Рисунок 3), и затем добавляем в параметр SSH keys внастройках (Рисунок 4).

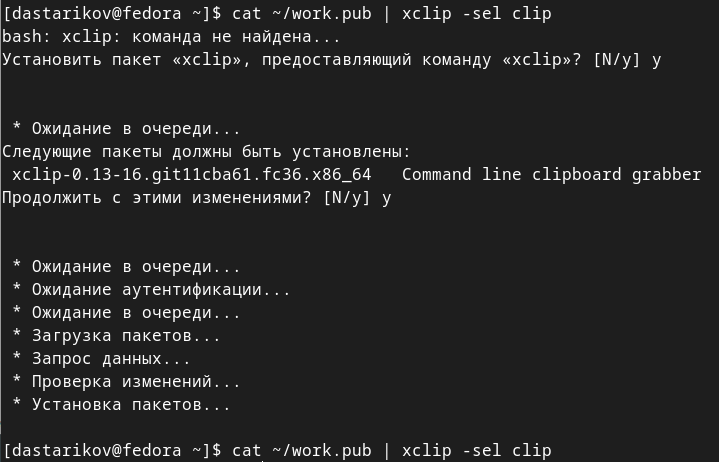


Рис. 3: Установка утилиты xclip и копирование ssh ключа в буфер обмена.

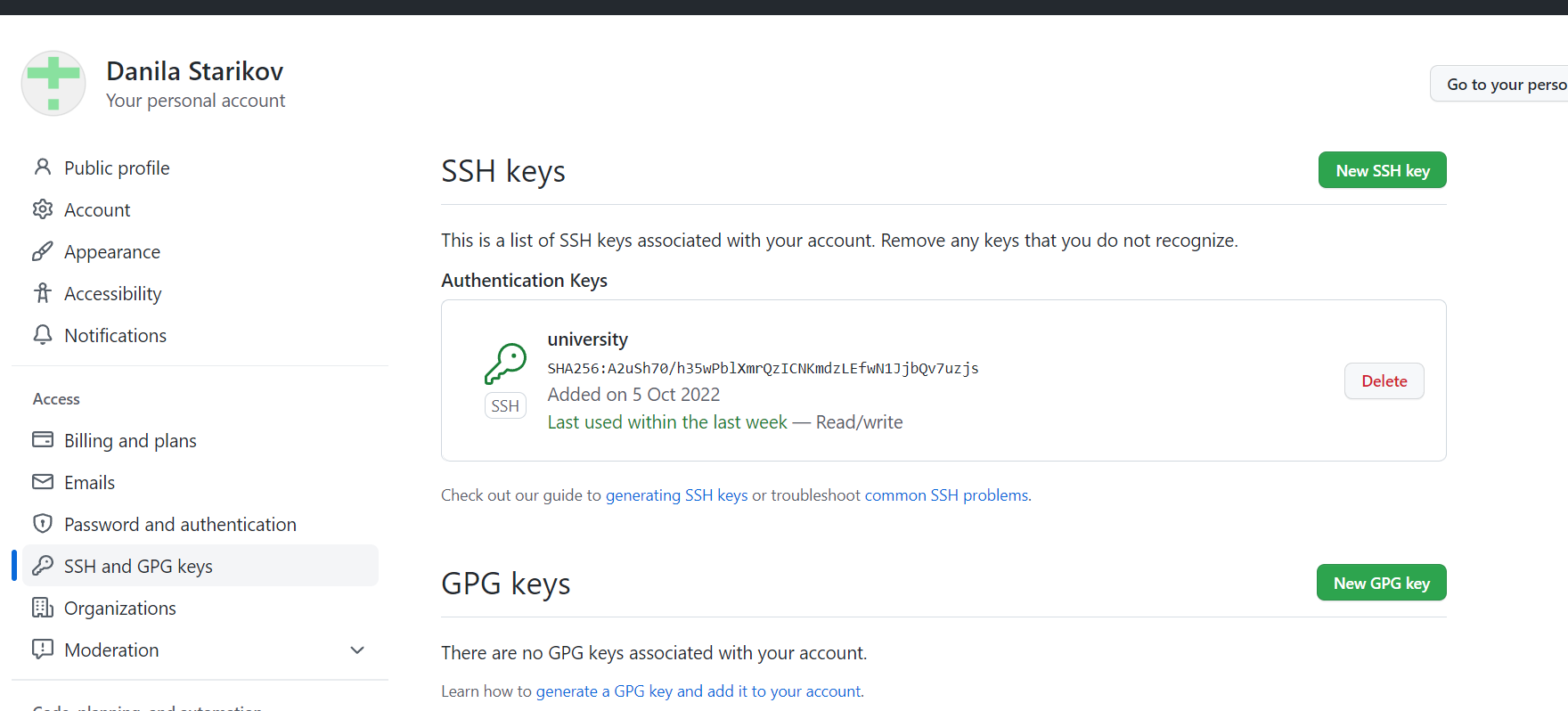


Рис. 4: Привязка сгенерированного ssh ключа к учетной записи на github.

### 2.1.3 Создание GPG ключа.

Сгенерировали GPG ключ для подтверждения автора коммита. Для этого запустили команду gpg --full-generate-key и прошли этапы его генерации (Рисунок 5).

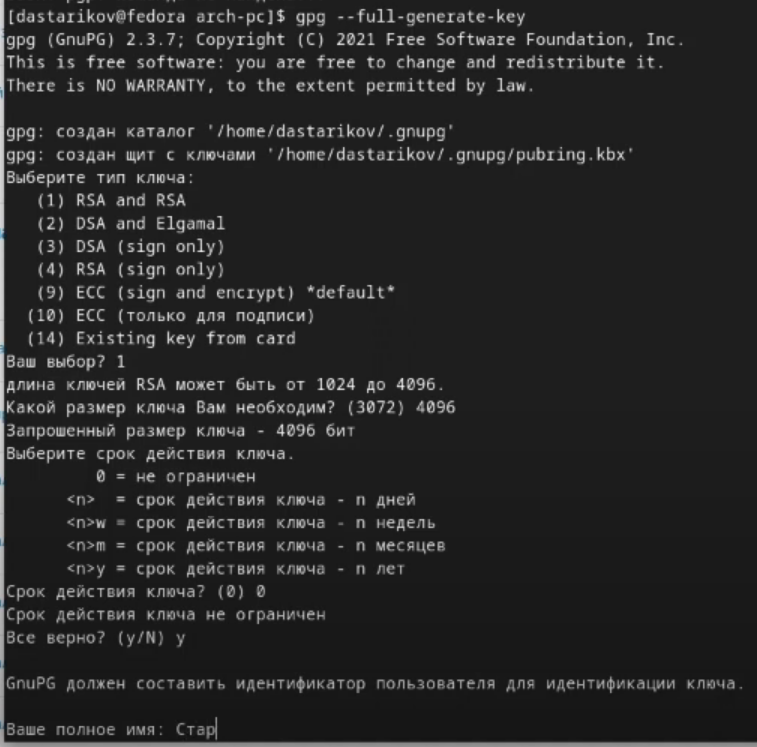


Рис. 5: Генерация GPG ключа.

После создания gpg ключа необходимо привязать его к аккаунту на Github. По команде gpg --list-secret-keys --keyid-format LONG получили отпечаток ключа, и затем скопировали сам ключ в буфер: gpg --armor --export <PGP Fingerprint> (Рис. 6 - 7).

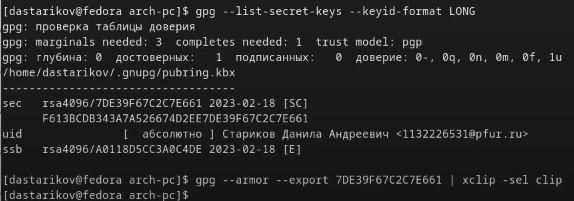


Рис. 6: Экспорт GPG ключа.

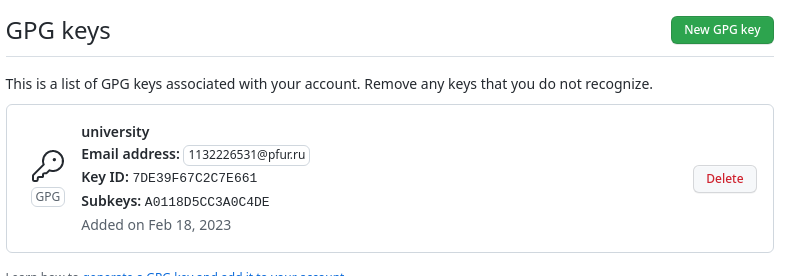


Рис. 7: Добавление GPG ключа к аккаунту на Github.

Последним этапом настроили автоматическую подпись коммитов в git (Рисунок 8).

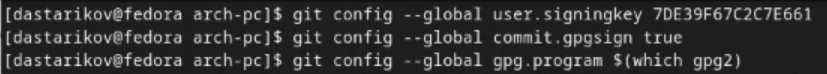


Рис. 8: Настройка автоматической подписи коммитов.

### 2.1.4 Сознание рабочего пространства и репозитория курса на основе шаблона.

Оформляем рабочее пространство для выполнения лабораторных работ, для этого создаем отдельный каталог для предмета «Операционные системы», где будут храниться все выполненные лабораторные работы (Рисунок 9).

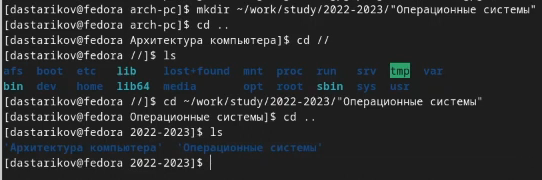


Рис. 9: Создание каталога для предмета «Операционные системы».

### 2.1.5 Сознание репозитория курса на основе шаблона.

Создадим репозиторий на основе шаблона, расположенного по адресу а <https://github.com/yamadharma/course-directory-student-template>. Для этого нажимаем кнопку «Use this template» (Рисунок 10), и в появившемся поле указываем имя репозитория (Рисунок 11).

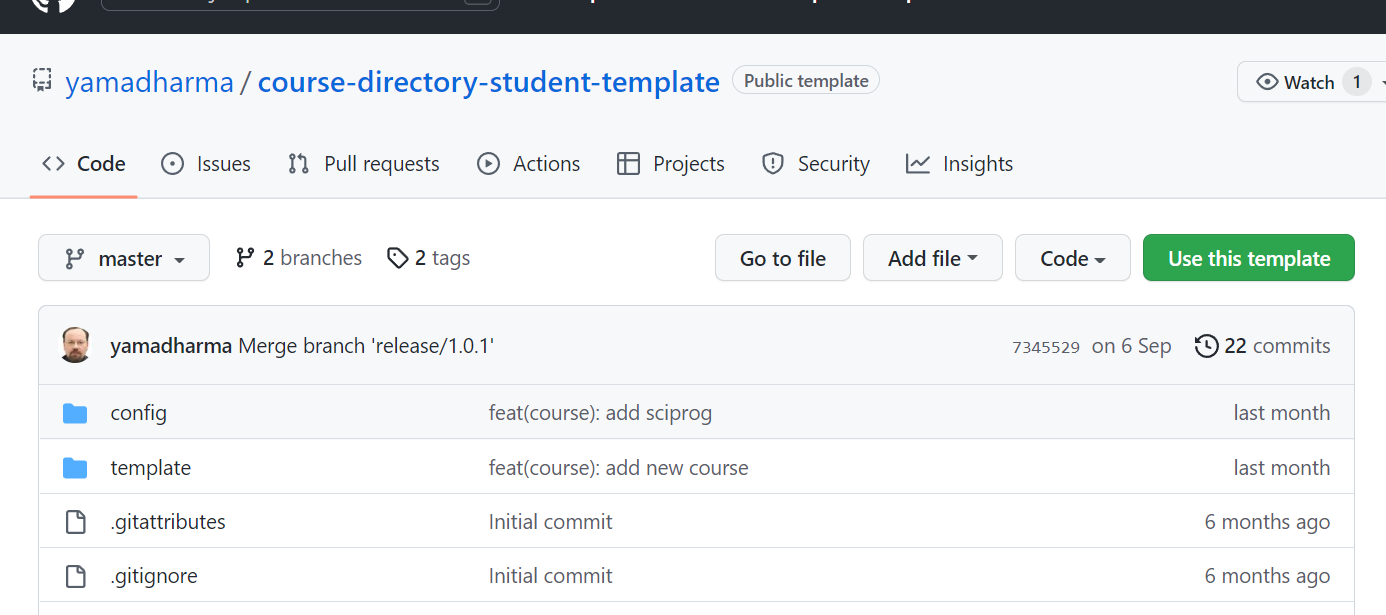


Рис. 10: Репозиторий, использованный в качестве шаблона.

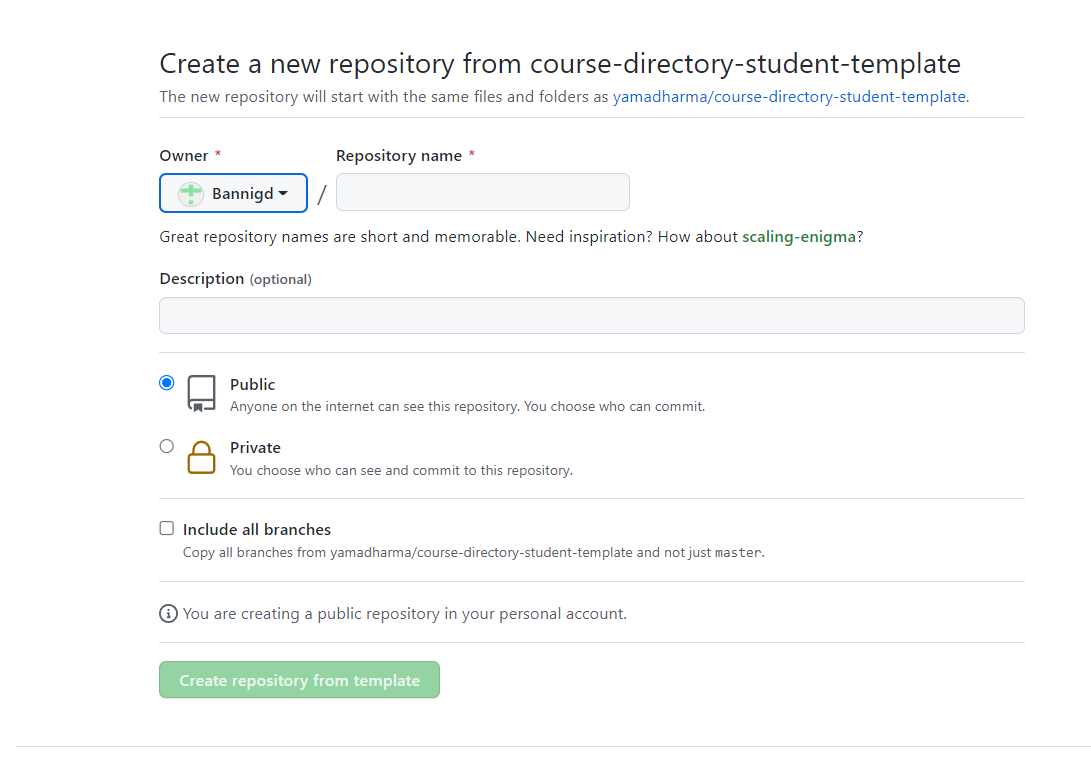


Рис. 11: Создание репозитория по шаблону.

Теперь необходимо клонировать созданный репозиторий в соответствующую папку, сначала добавим ssh ключ, чтобы возможность работать с репозиторием (Рисунок 12), затем клонируем наш репозиторий в папку предмета с помощью команды git clone –recursive git@github.com:Bannigd/study\_2022-2023\_os-intro os-intro (Рисунок 13).

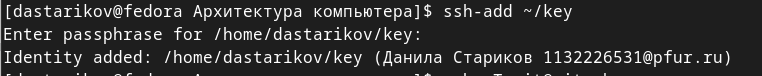


Рис. 12: Добавление ssh ключа.

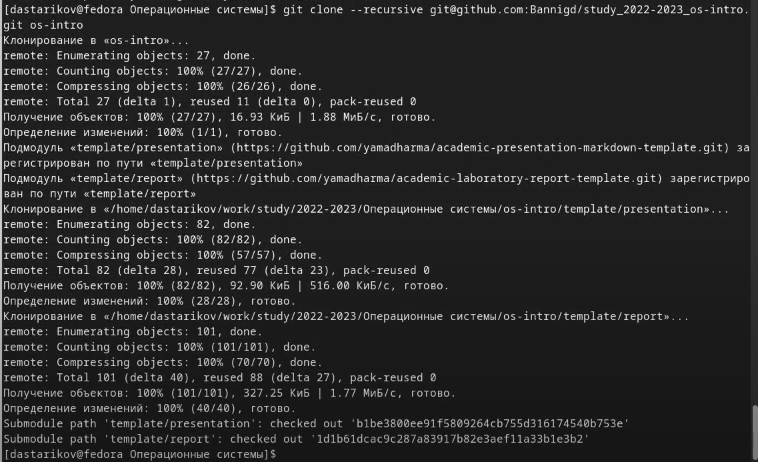


Рис. 13: Клонирование репозитория из github в каталог предмета.

### 2.1.6 Настройка каталога курса.

Для настройки репозитория для курса “Операционные системы”, удалили файл package.json, создали файл COURSE с именем шаблона os-intro и вызвали команду make, которая собрала необходимые для курса каталоги и файлы

Рис. 14: Настройка каталога курса.

Рис. 14: Настройка каталога курса.

Отправляем файлы на сервер (Рисунок 15).

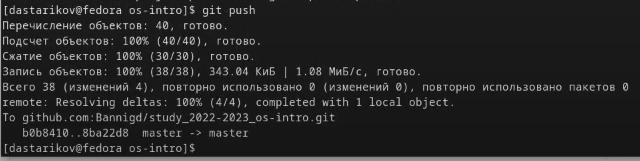


Рис. 15: Отправка изменений на сервер.

# 3 Выводы

При выполнении лабораторной работы изучили идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git.