Оглавление

[1. Однопользовательский, 1 ветка. 1](#_Toc520213473)

[2. Однопользовательский, 3 ветки 3](#_Toc520213474)

[3. Многопользовательский совместный 3](#_Toc520213475)

[4. Многопользовательский с архитектором 3](#_Toc520213476)

[5. Многопользовательский, 2 общих ветки (продакшен и препродакшен) 3](#_Toc520213477)

# Однопользовательский, 1 ветка.

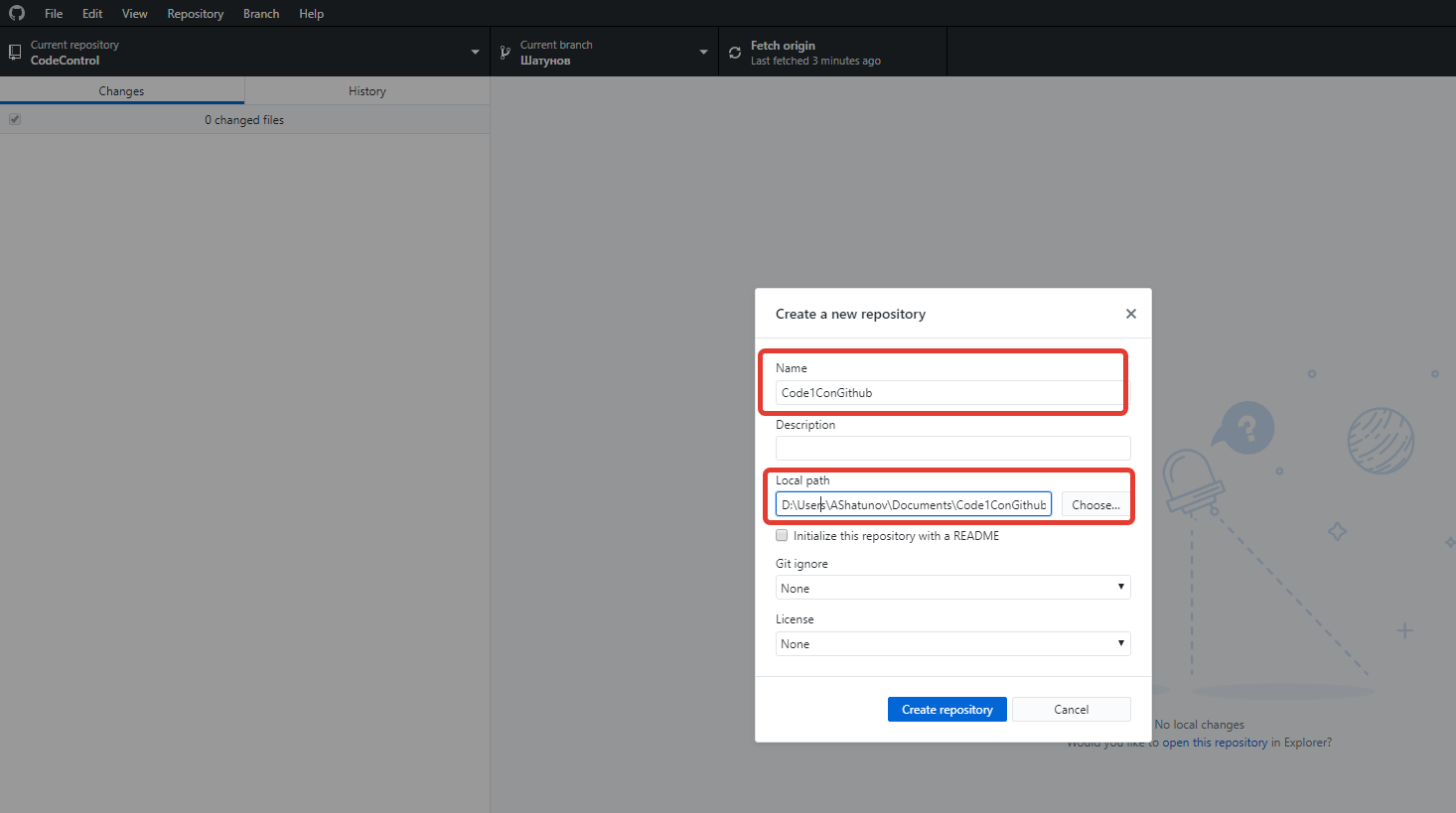
В этом случае мы извлекаем 2 функции гитхаба – систему контроля версий и облачное хранилище.

Ситуация: есть несколько компьютеров с интернетом, пользователь ведет разработку чего-либо.

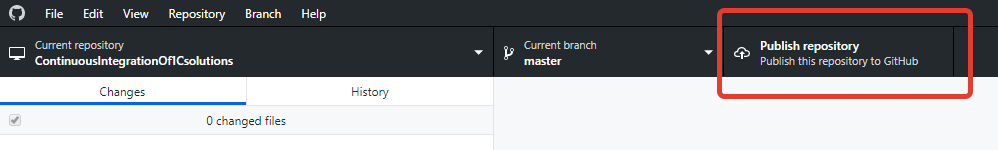
* 1. Создание репозитория

На любом из компьютеров открыть клиент и выполнить File/New repository

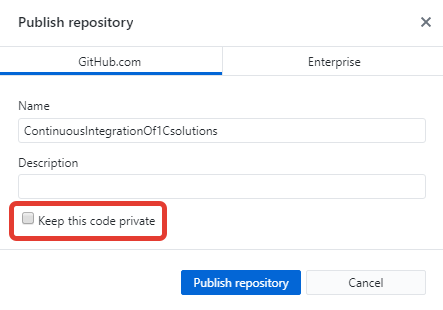
В появившемся окне указать имя проекта и путь к локальному хранилищу.



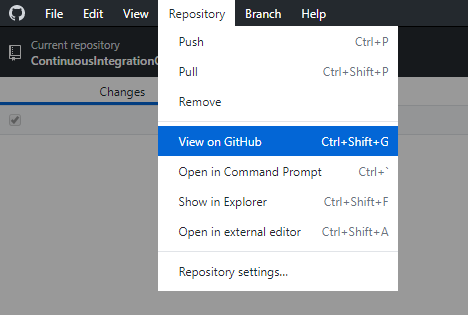
После создания нового проекта нужно отправить его в глобальный репозиторий. Для этого выполнить появившуюся команду Publish repository



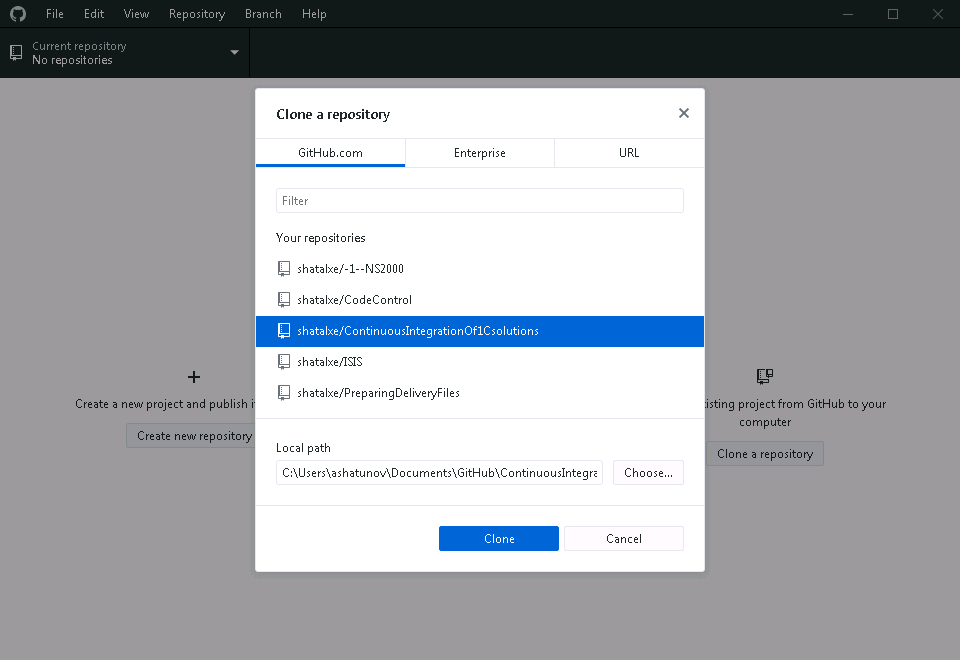
В новом окне убрать флаг Keep this code private



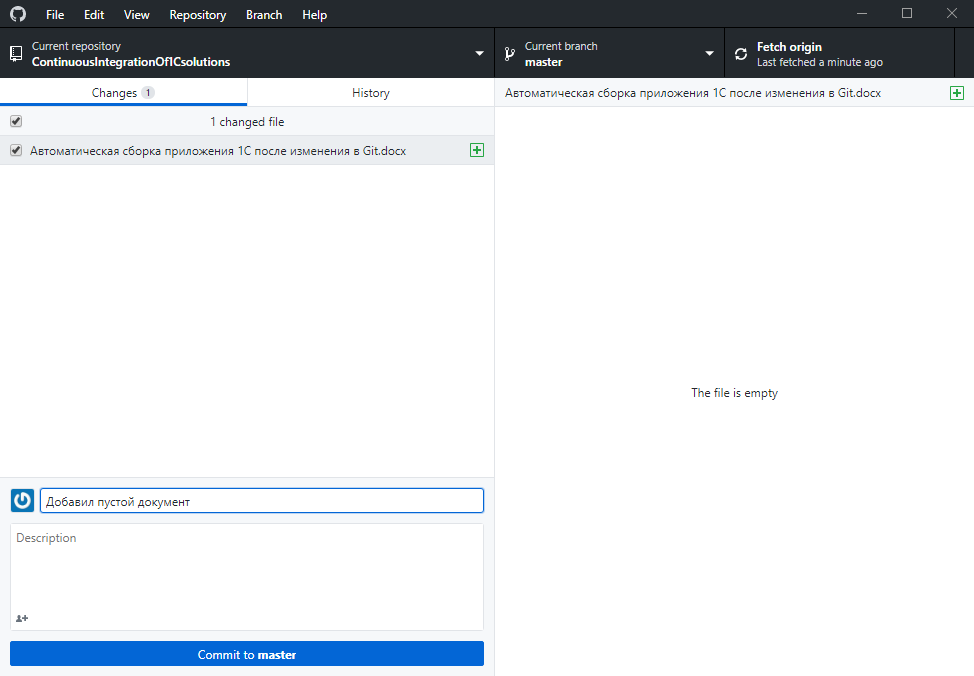
После публикации, за проектом можно следить через Веб-клиент.



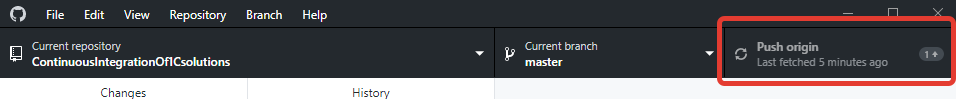
* 1. Подключение нового компьютера к репозиторию
     1. На втором устройстве открыть клиент GitHub
     2. Авторизоваться
     3. Выполнить File/Clone repository
     4. Указать нужный проект, нажать Clone



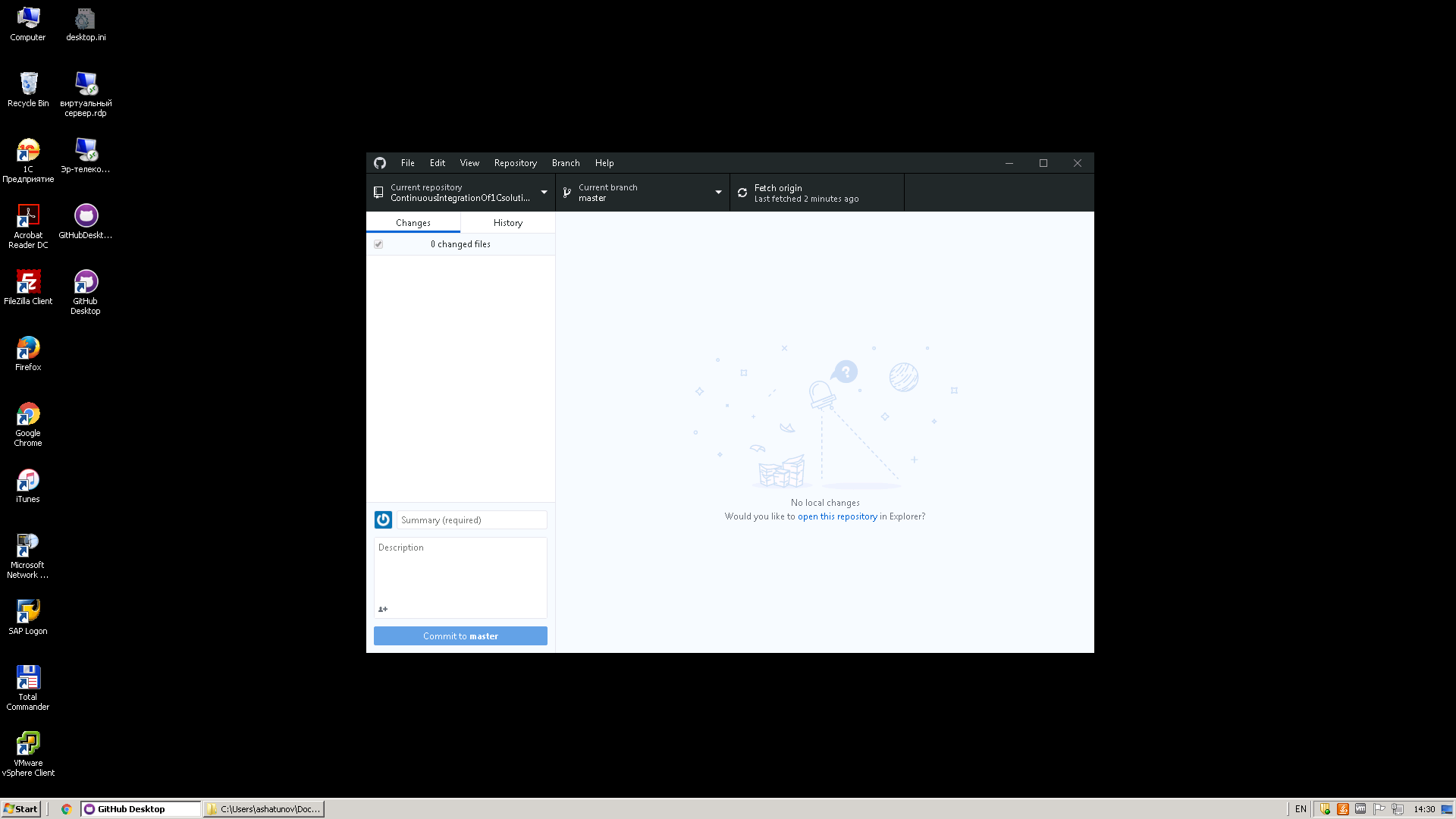
* 1. Выгрузка изменений в глобальный репозиторий
     1. Внести изменения в каталог проекта. Клиент GitHub покажет эти изменения.



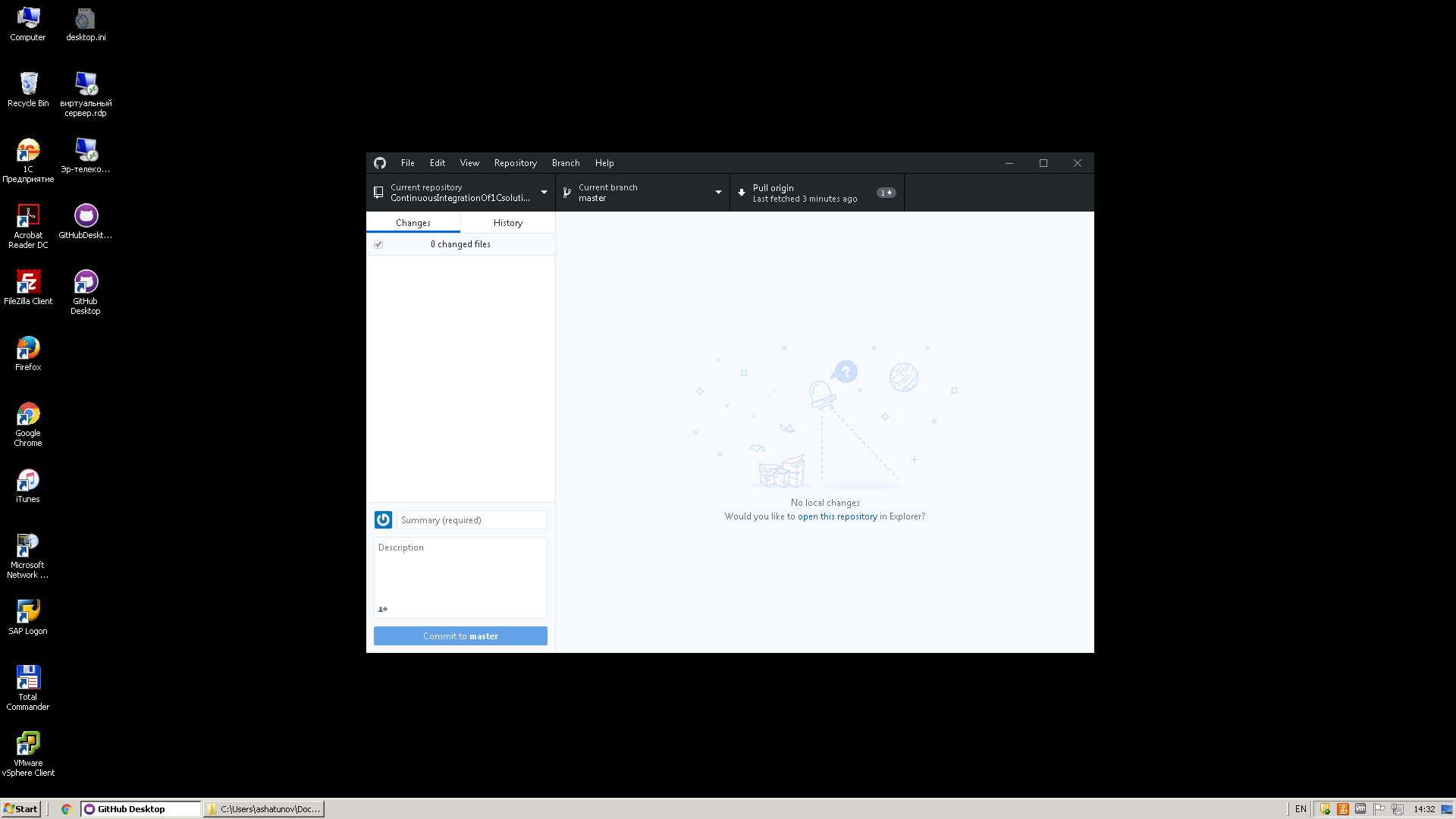
* + 1. В поле summary нужно ввести краткое описание аносимых изменений
    2. Выполнить commit to master
    3. Выполнить появившуюся команду Push origin



* 1. Загрузка из глобального репозитория
     1. На втором устройстве выполнить Fetch origin



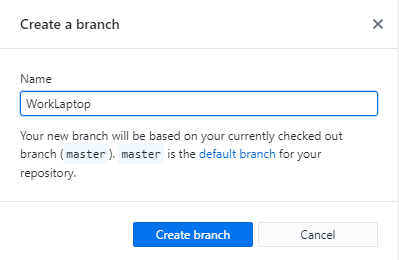
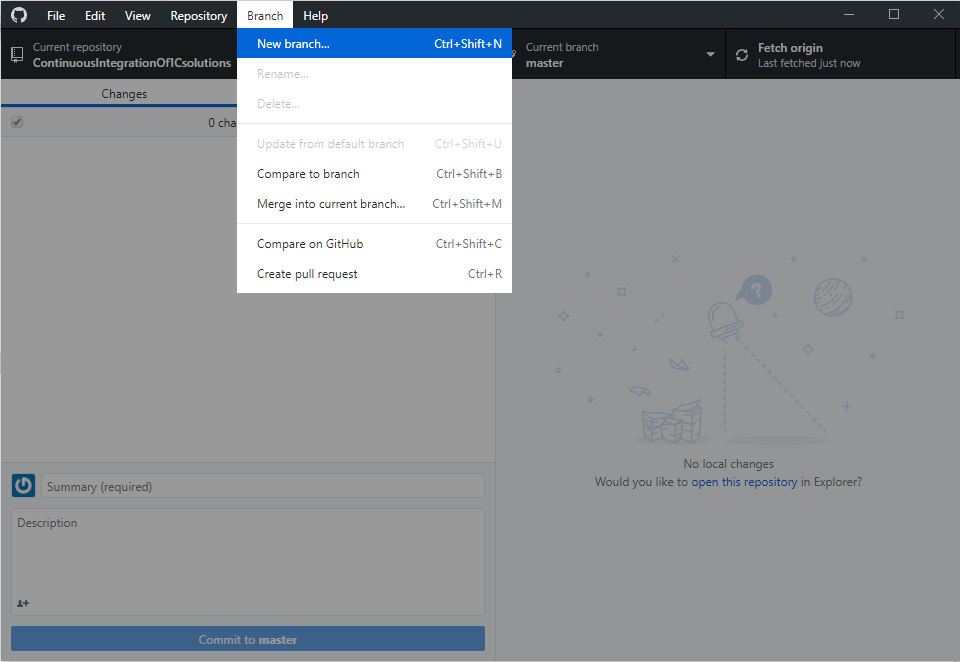
* + 1. Выполнить Pull origin



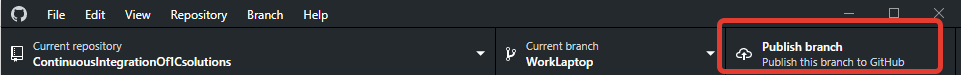
# Однопользовательский, 3 ветки

Вынесем работу на разных устройствах в разные ветки. Время от времени изменения будем отправлять в ветку master и обновляться из неё.

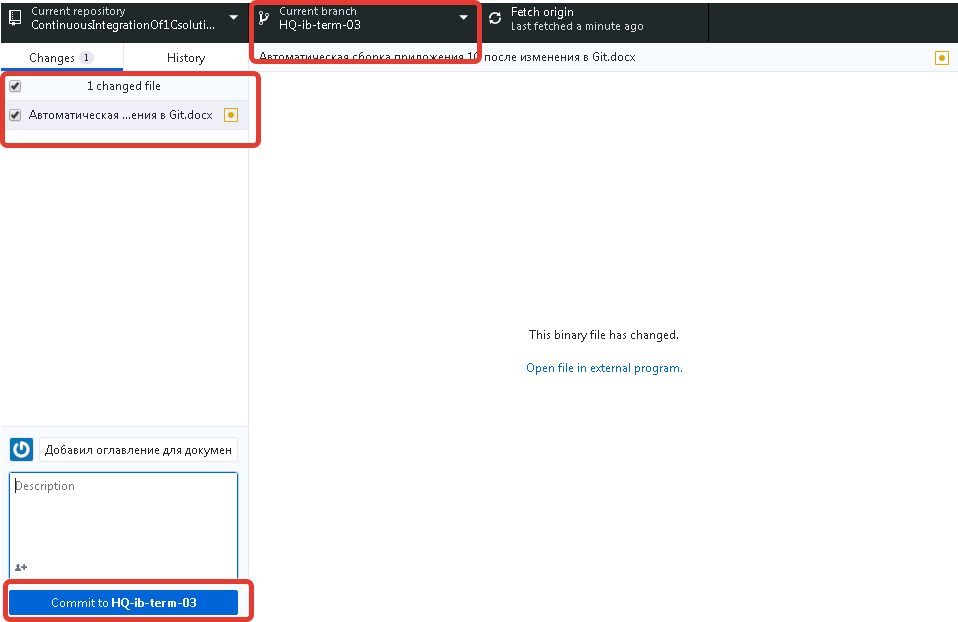
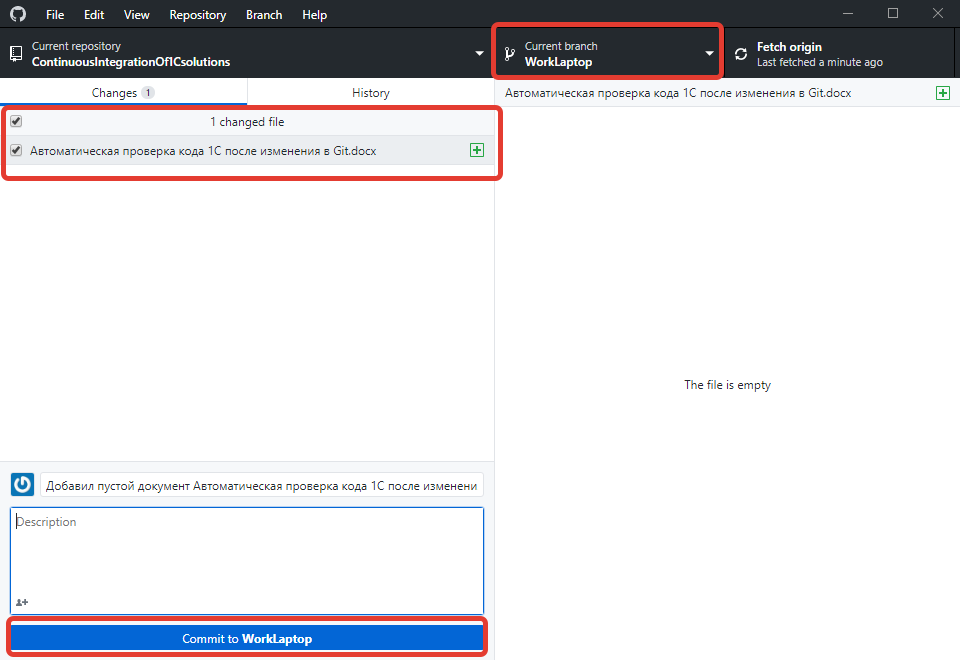
* 1. На обоих устройствах создадим по ветке



* 1. Отправим их в облако командой Publish branch



* 1. Добавим по коммиту на обоих устройствах.

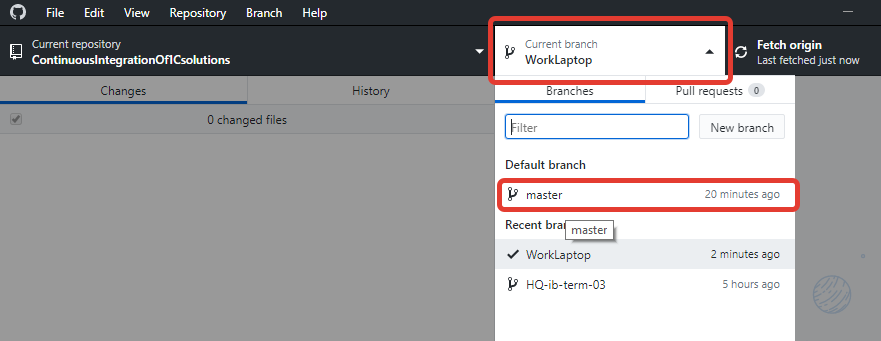


* 1. Отправим изменения в облако

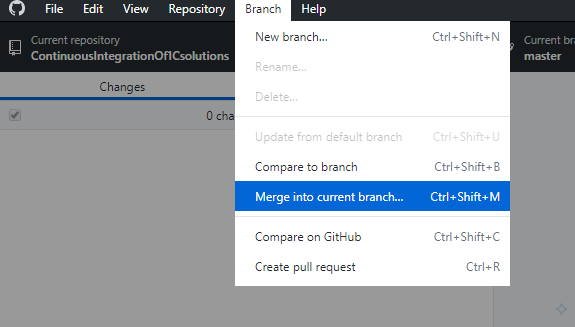
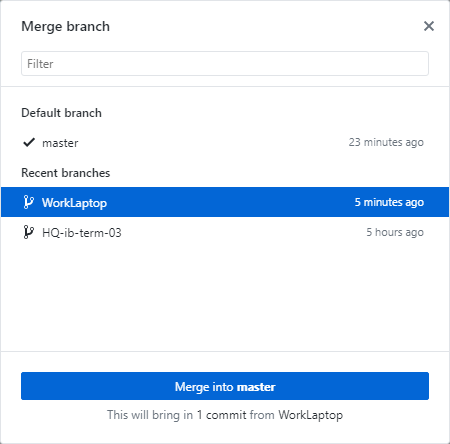


Когда завершаем работу над задачей, и хотим обновить ветку master:

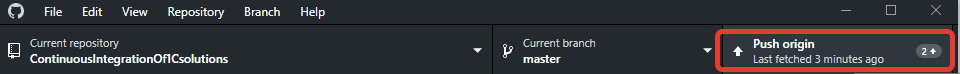
* 1. Переключаемся на ветку Master



* 1. Выполняем слияние с нужной веткой. Нажимаем Merge into [имя ветки] (если ветки равны, то команда не будет доступна)

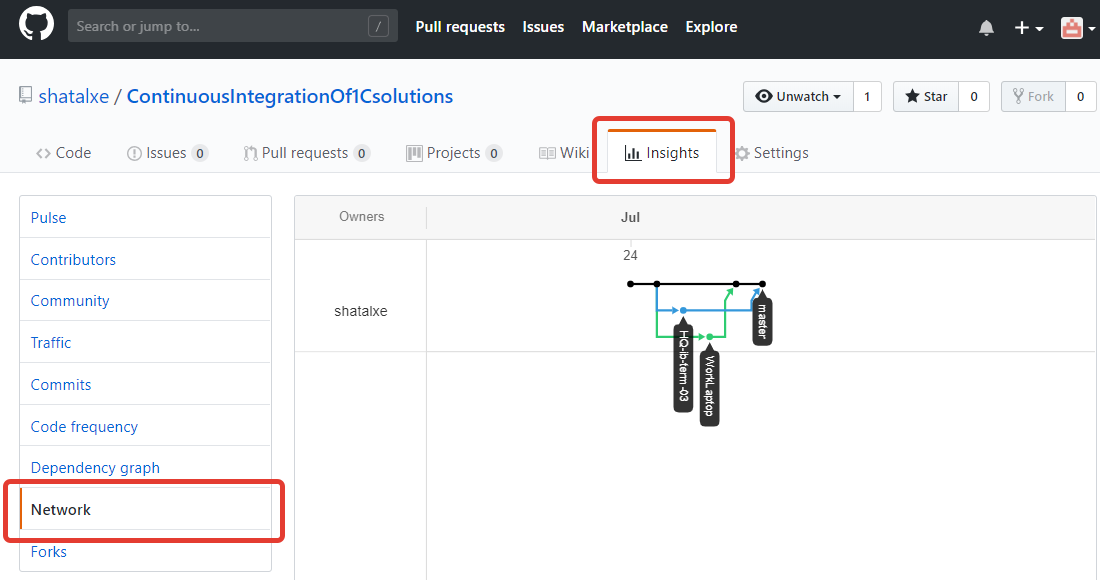
 

* 1. Отправляем изменения в облако



* 1. Повторяем процедуру слияния на других устройствах, предварительно обновив локальный репозиторий.

Изменения глобального репозитория в графическом виде можно посмотреть в веб-интерфейсе



# Многопользовательский с архитектором

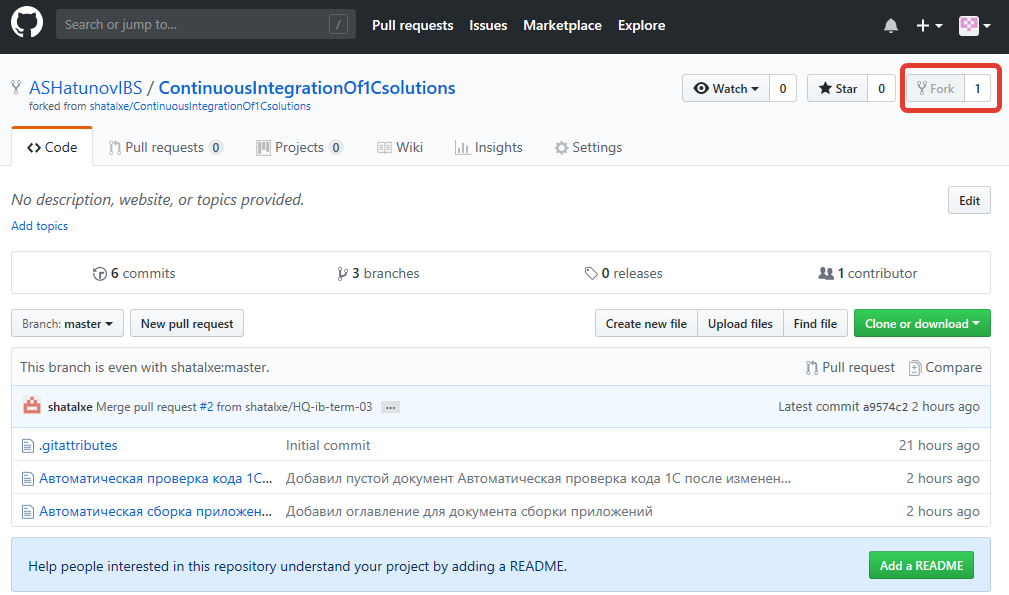
Архитектором будем считать пользователя, создавшего проект либо приглашенного по инвайту в список соучастников (Collaborators).

Все остальные пользователи клонируют репозиторий и отправляют Pull Request в основной.

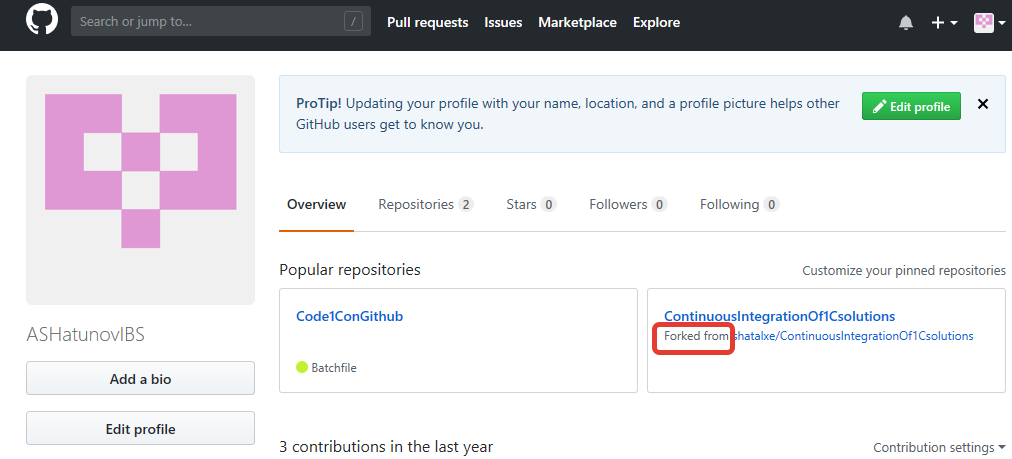
Подключим к нашему проекту нового пользователя. Для этого ему нужно сделать форк проекта и склонировать основной репозиторий.

В Веб-интерфейсе:

* 1. Авторизовываемся под новым пользователем, открываем страницу проекта.
  2. Делаем форк

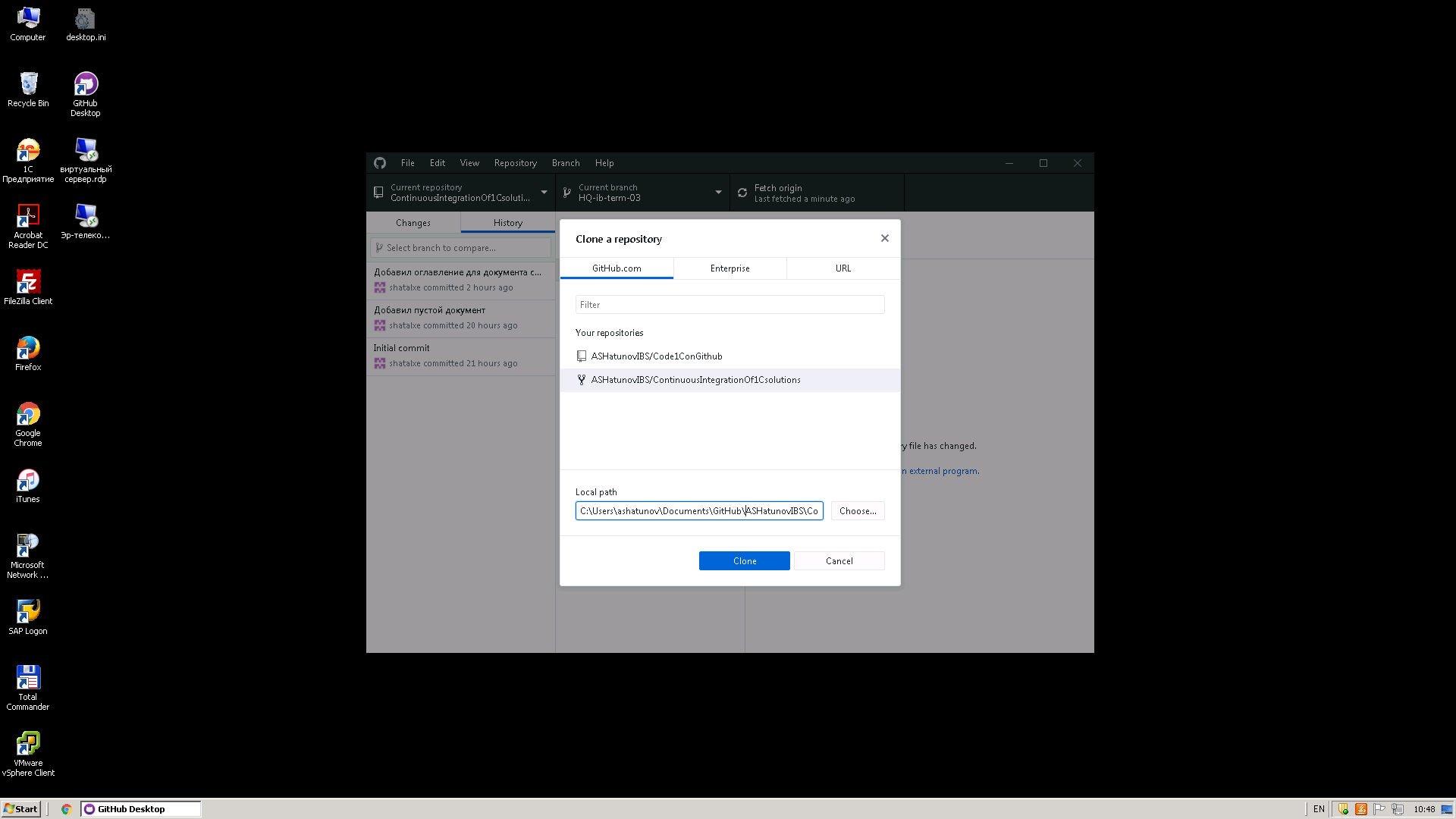


В профиле пользователя появится информация о форке

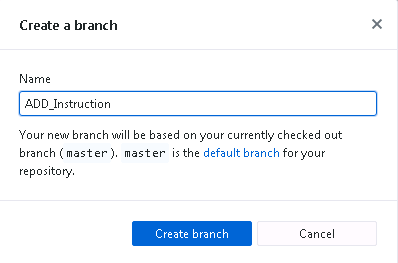


В клиенте windows:

* 1. Авторизовываемся
  2. Клонируем репозиторий



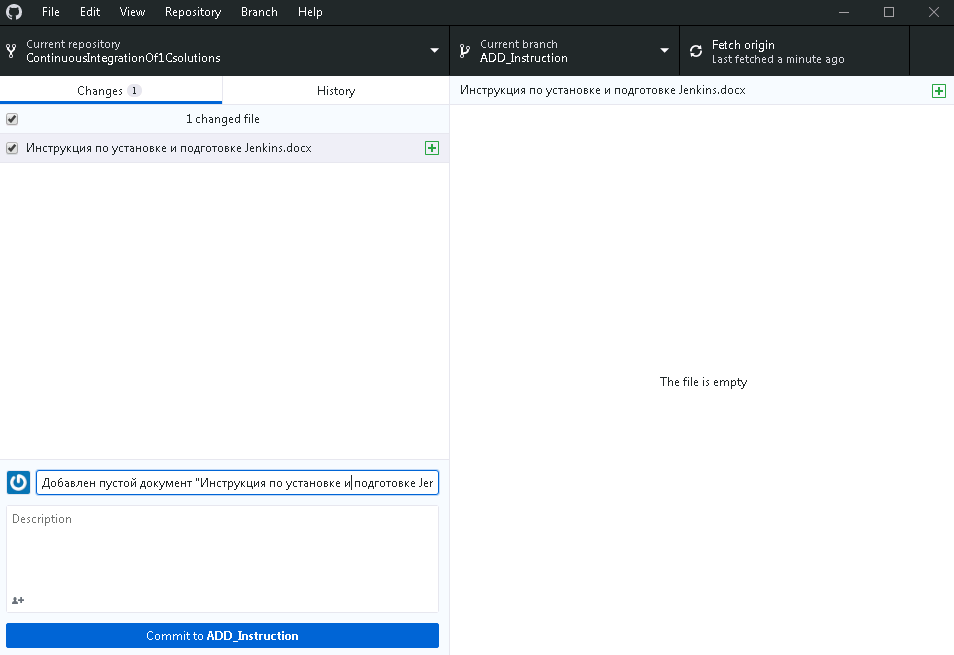
* 1. Добавляем новую ветку



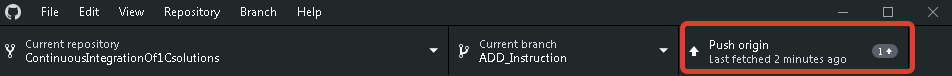
* 1. Отправляем её в облако



* 1. Вносим изменения, делаем коммит

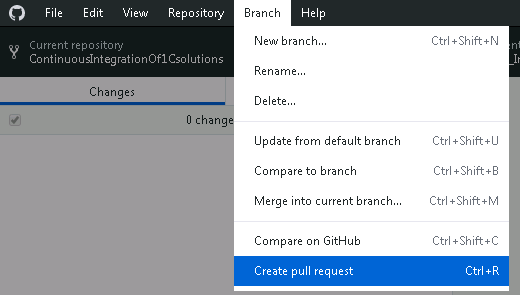


* 1. Пушим в облако



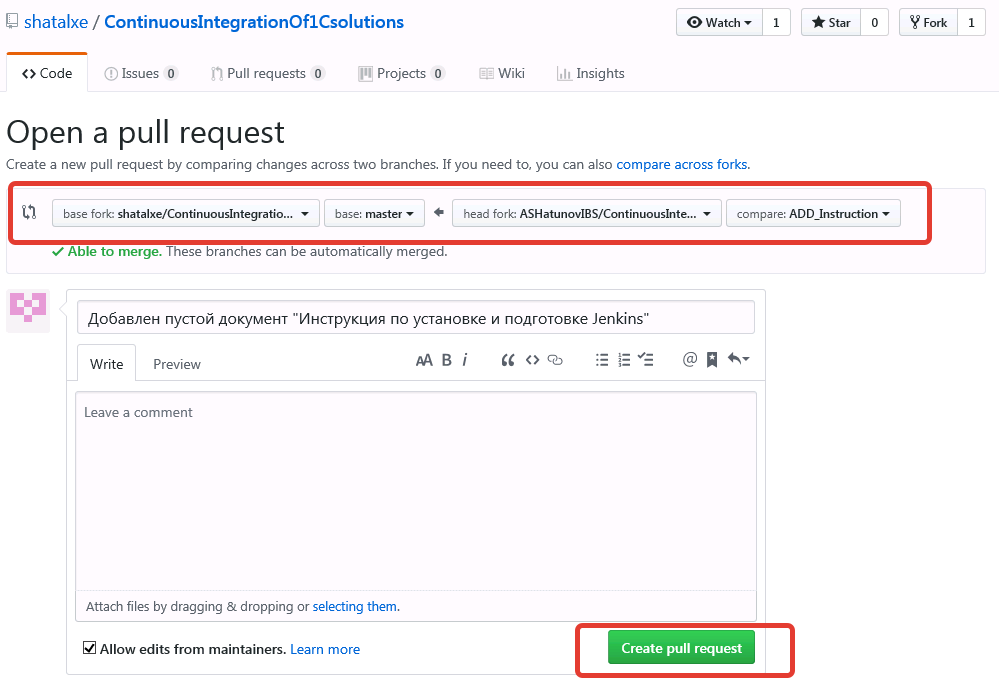
Когда мы закончим работу над задачей, нужно отправить наши изменения в главный репозиторий. Делается это с помощью запросов Pull request

* 1. Открываем страницу создания Pull request



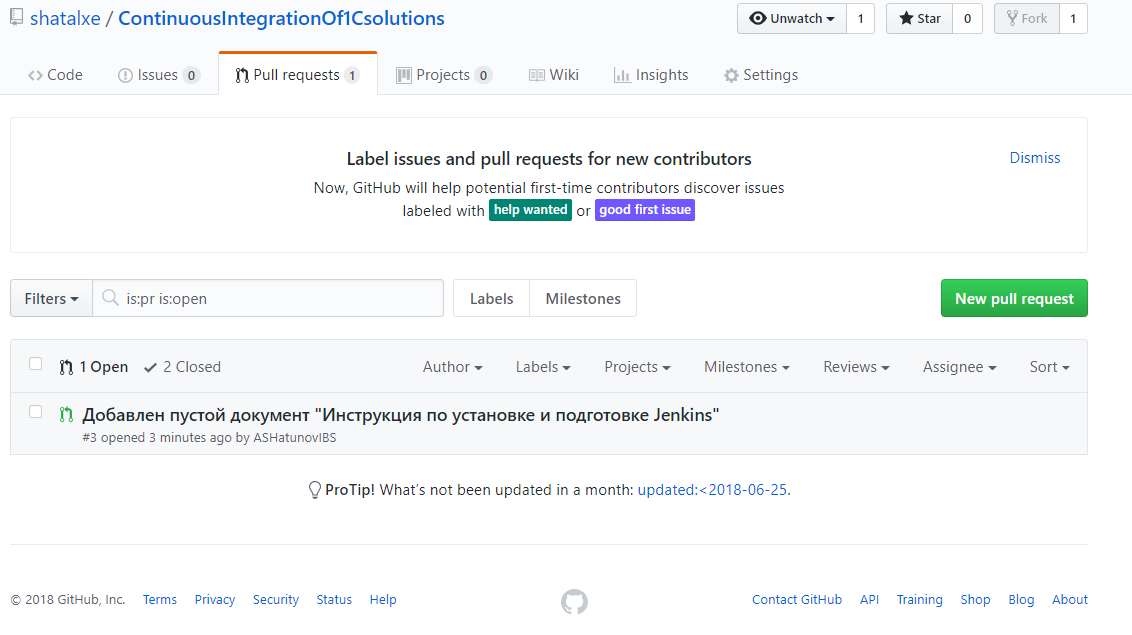
В веб-клиенте:

* 1. Убеждаемся, что слияние идет туда, куда нам надо
  2. Жмем Create pull request

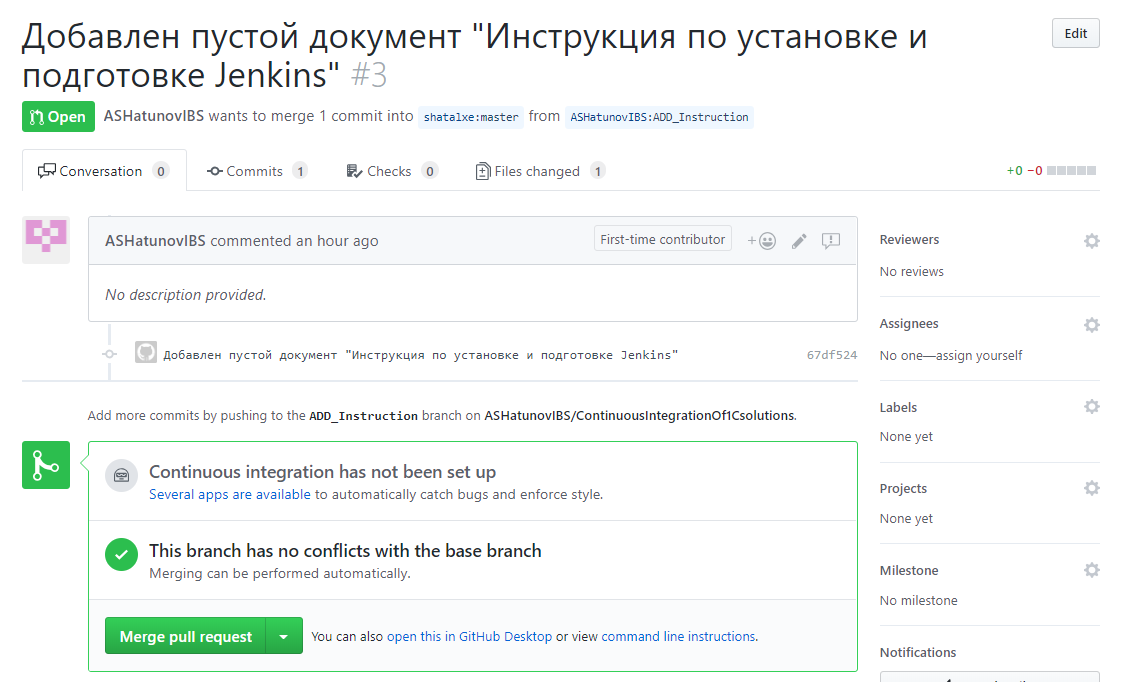


Владелец главного репозитория получит уведомление.

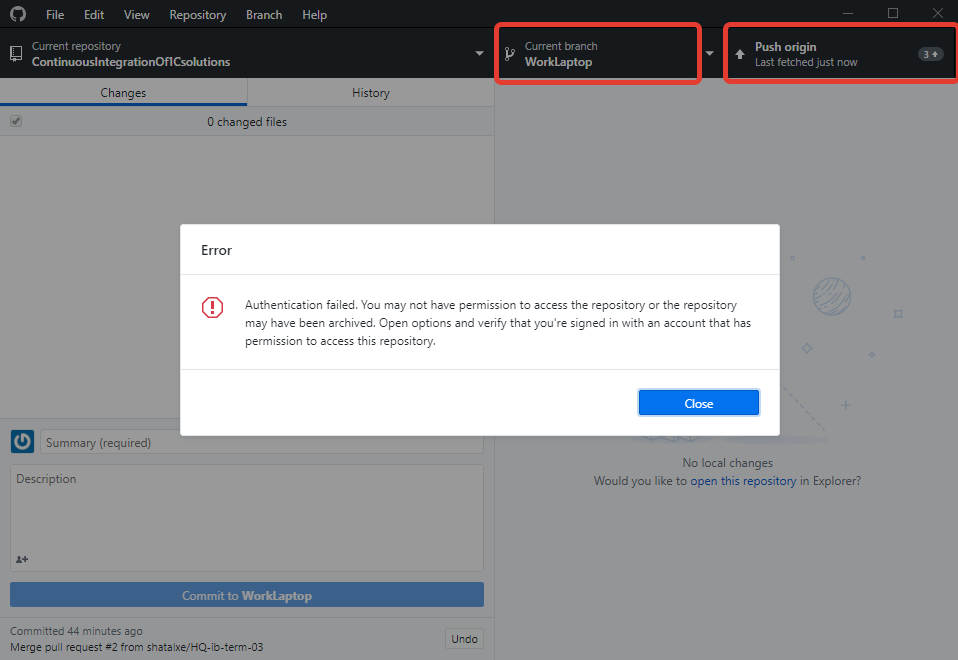
* 1. В главном репозитории (находясь под учетной записью его участника) переходим в раздел Pull requests
  2. Выбираем интересующий Pull request



* 1. Merge pull request



*При попытке обновить репозиторий, к которому у нас нет доступа, получим предупреждение:*



# Многопользовательский совместный

# 5. Многопользовательский, 2 общих ветки (продакшен и препродакшен)