

Система контроля версий GitCAD.io

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

В настоящем руководстве применены следующие термины с соответствующими определениями:

репозиторий: централизованное хранилище данных, представляющее собой структурированный архив файлов, который доступен для коллективного использования и распространения посредством сетевых протоколов.

коммит: фиксированное состояние проекта, которое сохраняется в репозитории. Коммит включает в себя изменения, произведенные в файлах и каталогах, и сопровождается метаданными, такими как идентификатор автора, дата и сообщение с описанием изменений. Коммиты формируют цепочки изменений, обеспечивая возможность отслеживания истории проекта и восстановления предыдущих состояний.

ветка: отдельная линия разработки внутри репозитория, которая позволяет параллельно вести разработку различных частей проекта без взаимных помех. Каждая ветвь представляет собой независимую копию репозитория, начиная с определенного момента (точки ветвления). При создании новой ветви текущие изменения сохраняются в оригинальной ветви, тогда как новая ветвь развивается независимо до момента их возможного слияния.

слияние: процедура интеграции изменений между различными ветвями проекта. В ходе слияния система управления версиями объединяет изменения из одной ветви в другую, создавая единый поток изменений. Этот процесс требует разрешения возможных конфликтов, возникающих при изменении одних и тех же файлов в обеих ветвях.

1 РАБОТА С ВЕТВЯМИ

При инициализации репозитория создается главная ветвь, традиционно называемая `main` или `master`, которая считается основным направлением разработки проекта. Все последующие изменения происходят в отдельных ветвях, создаваемых специально для новых функций или исправлений. Разработчик, приступивший к работе над новым компонентом, создает отдельную ветвь, производную от основной, и работает в ней, добавляя или модифицируя файлы. По завершении работы в данной ветви и успешной проверке стабильности изменений, ветвь сливается обратно в главную ветвь.

Ветви могут быть созданы не только из основной, но и из любых других существующих ветвей. Однако рекомендуется избегать чрезмерной разветвленности, поскольку это усложняет дальнейшую синхронизацию и поддержку проекта.

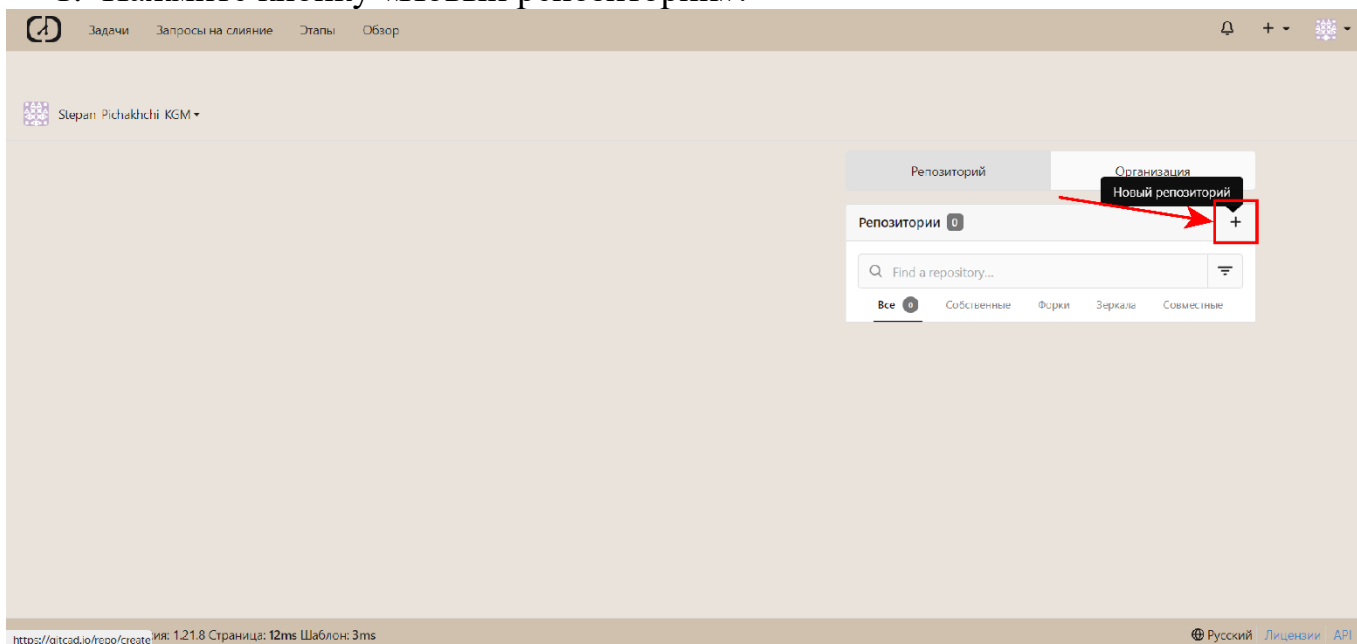
В ветвях могут храниться разные исполнения проекта, но это затруднит их синхронизацию. Поэтому так лучше не делать, а для исполнений проекта использовать другие инструменты основной CAD программы.

2 УСТАНОВКА

1. Зарегистрируйтесь на сервисе <https://gitcad.io/>
2. Создайте учетную запись на платформе <https://github.com/>
ВАЖНО: используйте одинаковый адрес электронной почты для регистрации на обоих ресурсах!!!
3. Скачать приложение GitHub Desktop с сайта <https://github.com/apps/desktop>

3 СОЗДАНИЕ НОВОГО РЕПОЗИТОРИЯ


1. Нажмите кнопку «Новый репозиторий».



2. Укажите имя репозитория и его описание. Имя должно отражать суть проекта, а описание может включать дополнительные детали.

Новый репозиторий

A repository contains all project files, including revision history. Already hosting one elsewhere? [Migrate repository.](#)

Владелец *  Stepan_Pichakhchi_KGM

Некоторые организации могут не отображаться в раскрывающемся списке из-за максимального ограничения количества репозиториях.

Название репозитория *

Лучшие названия репозиториях состоят из коротких, легко запоминаемых и уникальных ключевых слов.

Видимость ☐ Сделать репозиторий приватным

Только владелец или члены организации, при наличии прав, смогут увидеть это.

Описание

Добавьте краткое описание (необязательно)

Шаблон

Метки задач

Выберите набор ярлыков задачи.

3. Поставьте галочку «Сделать репозиторий приватным», если не хотите, чтобы другие пользователи имели к нему доступ. Для совместной работы в одном

репозитории доступна функция подключения соавторов. Об этом будет сказано отдельно.

4. Выберите галочку «Инициализировать репозиторий».

Лицензия определяет, что другие люди могут, а что не могут делать с вашим кодом. Не уверены, какая лицензия подходит для вашего проекта? Смотрите [Выберите лицензию](#).

README

Это место, где вы можете написать подробное описание вашего проекта.

☒ Инициализировать репозиторий (Добавляет .gitignore, LICENSE and README)

Ветка по умолчанию

Ветка по умолчанию является базовой веткой для запросов на слияние и коммитов кода.

Модель доверия подписи

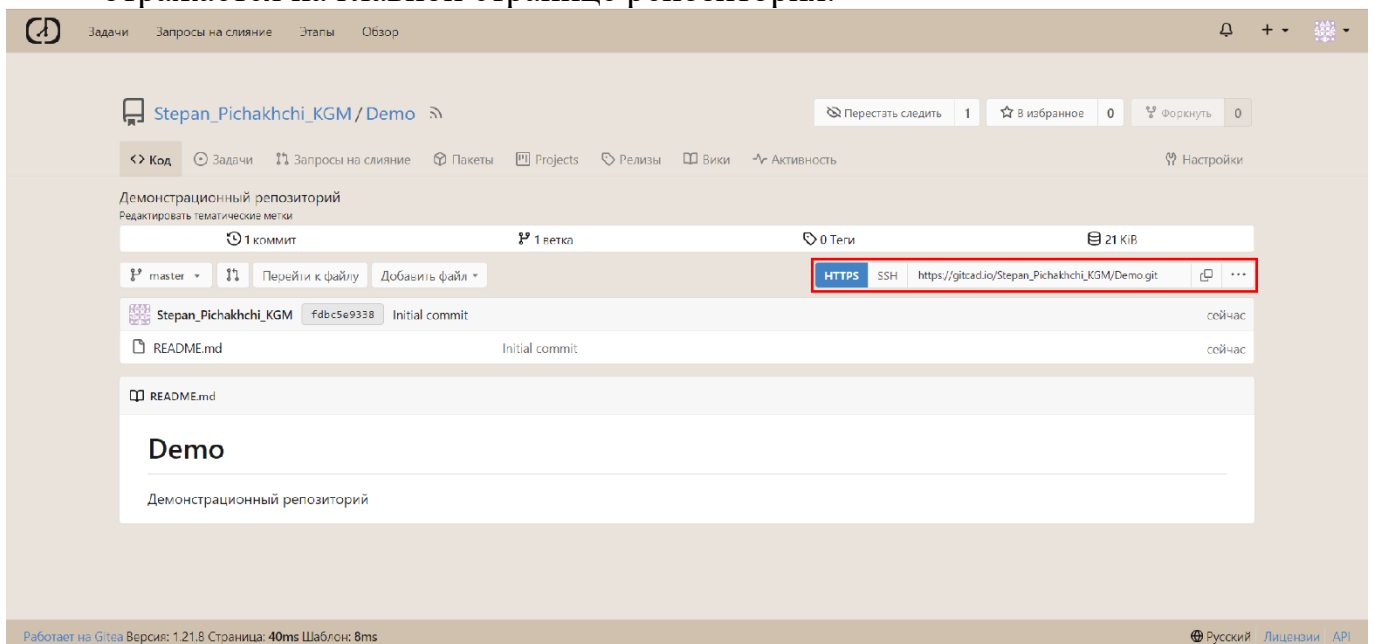
Выберите модель доверия для проверки подписи. Возможные варианты:

- Соавтор: Доверять подписям соавторов
- Автор коммита: доверять подписям, соответствующим авторам коммитов
- Соавтор+Коммитер: Доверять подписям соавторов, которые соответствуют автору коммита
- По умолчанию: используйте модель доверия по умолчанию для этой установки

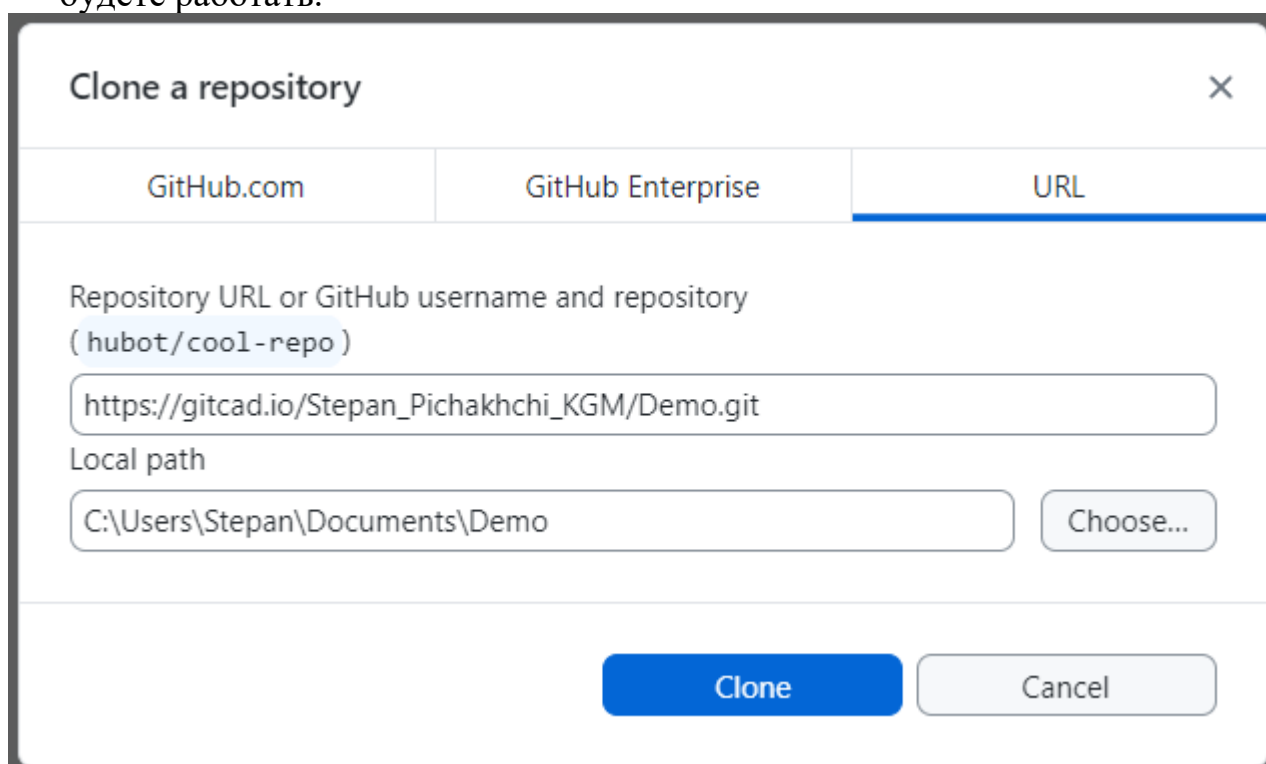
Шаблон ☐ Сделать репозиторий шаблоном

5. После заполнения данных внизу нажать кнопку «Создать репозиторий»

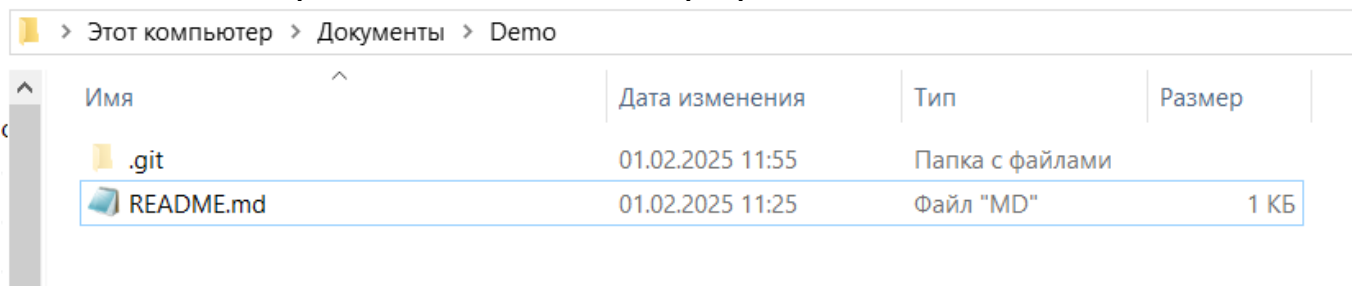
6. Вы создали репозиторий на сервере GitCAD. Сейчас в вашем репозитории только один файл «README.md», созданный автоматически. По умолчанию он содержит название и описание репозитория. Информация из этого файла отражается на главной странице репозитория.



7. Необходимо скопировать ссылку на этот репозиторий и запустить приложение GitHub Desktop. В приложении GitHub Desktop надо будет войти через аккаунт GitHub.
8. Начальное окно GitHub Desktop предлагает несколько вариантов для создания репозитория на вашем ПК: 1 – копировать репозиторий из интернета; 2 – создать репозиторий из папки на жестком диске; 3 – если папка-репозиторий уже существует на вашем ПК, то ее можно подключить к GitHub Desktop. Нам нужен первый вариант, так как сам репозиторий расположен на сервере GitCAD. Для этого вставляем скопированную ранее ссылку и выбираем папку, где на компьютере будут храниться файлы из репозитория, с которыми вы будете работать.

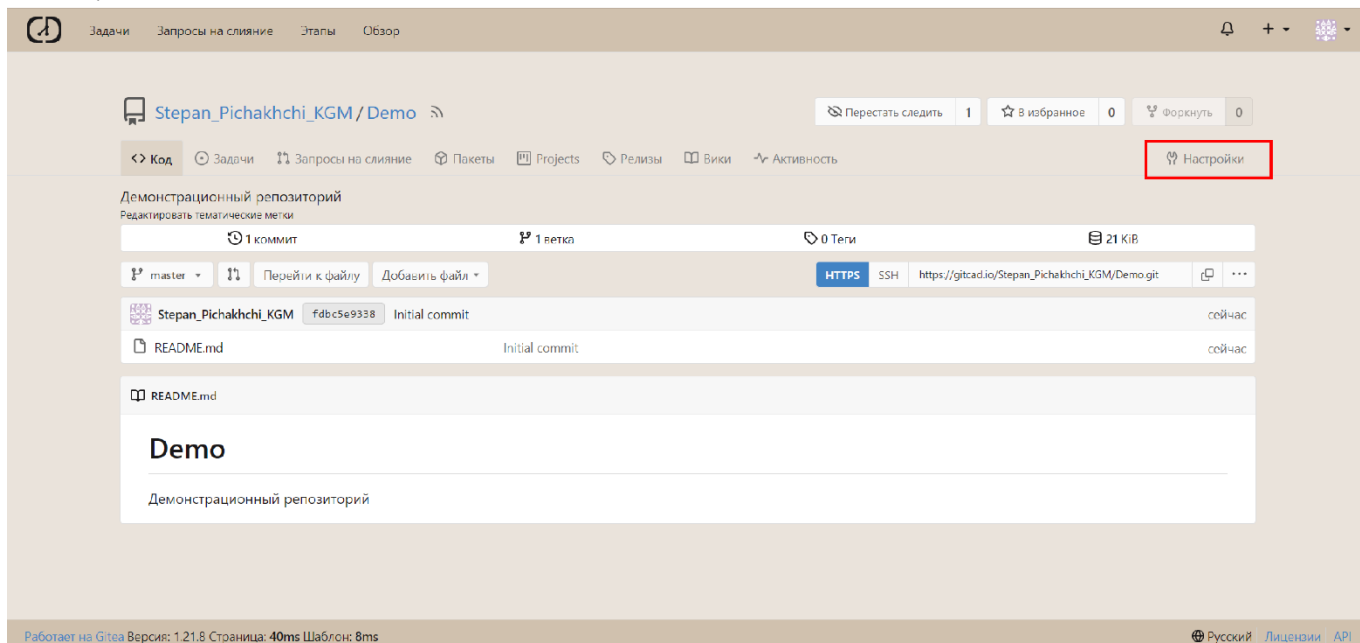


9. По указанному пути появилась папка с файлами, скопированными из указанного репозитория (для нового репозитория это файл «Readme.md») и скрытой папкой с метаданными репозитория. Вы можете работать в этой папке и сохранять в ней файлы моделей. Через приложение GitHub Desktop вы сможете отправлять изменения на сервер в виде коммитов.

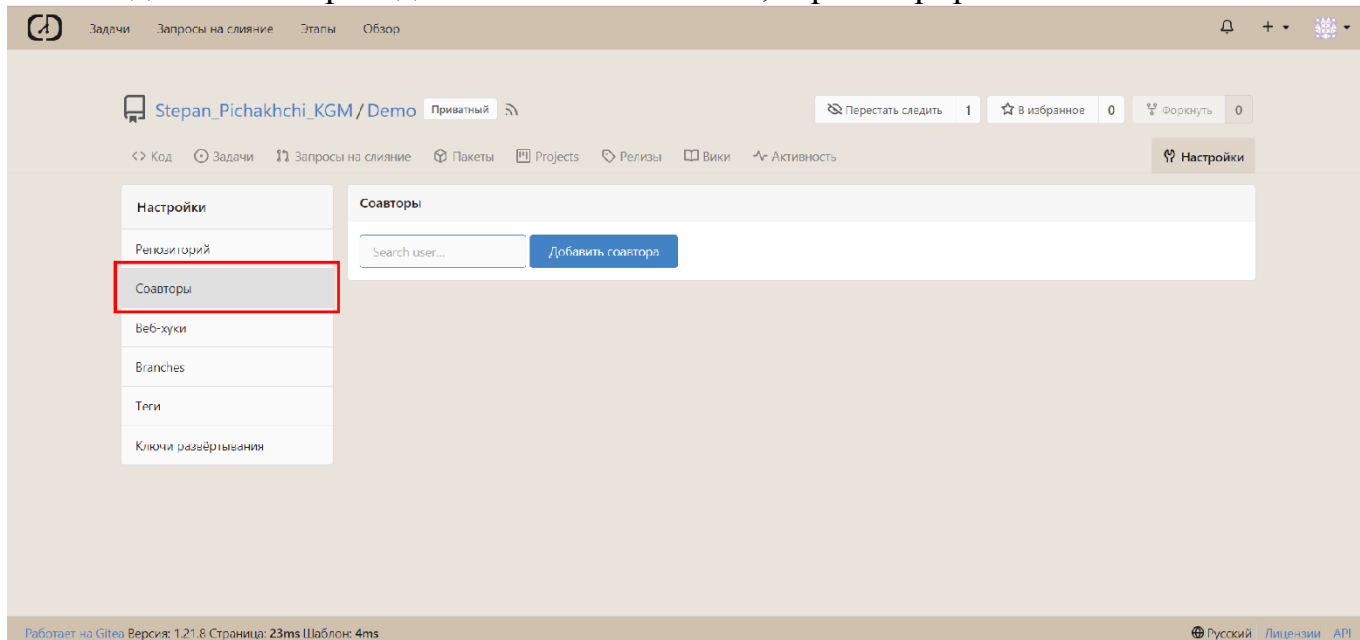


4. КОПИРОВАНИЕ УЖЕ СУЩЕСТВУЮЩЕГО РЕПОЗИТОРИЯ

Чтобы получить копию уже существующего репозитория, следуйте шагу 8 из процедуры создания нового репозитория, указывая URL нужного вам репозитория. Если это приватный репозиторий, убедитесь, что владелец добавил вас в список соавторов через настройки доступа. Добавить соавторов можно через настройки на сайте:



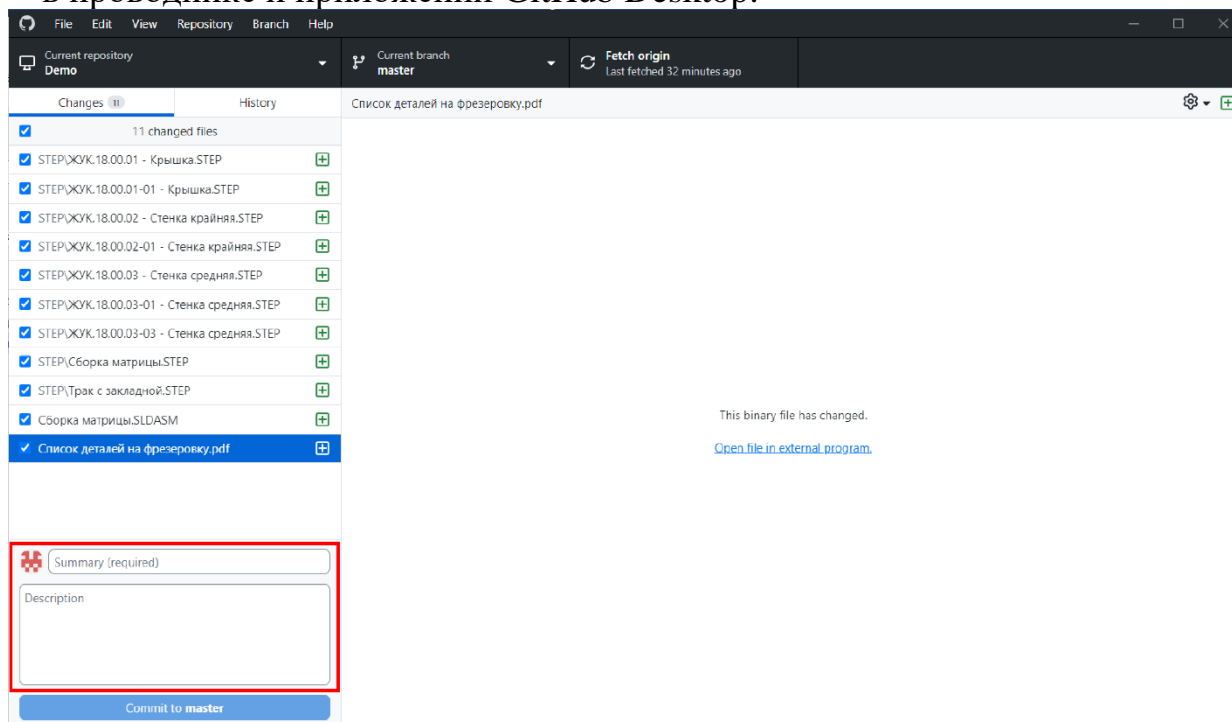
Во вкладке «Соавторы» добавить пользователя, зарегистрированного в GitCAD.



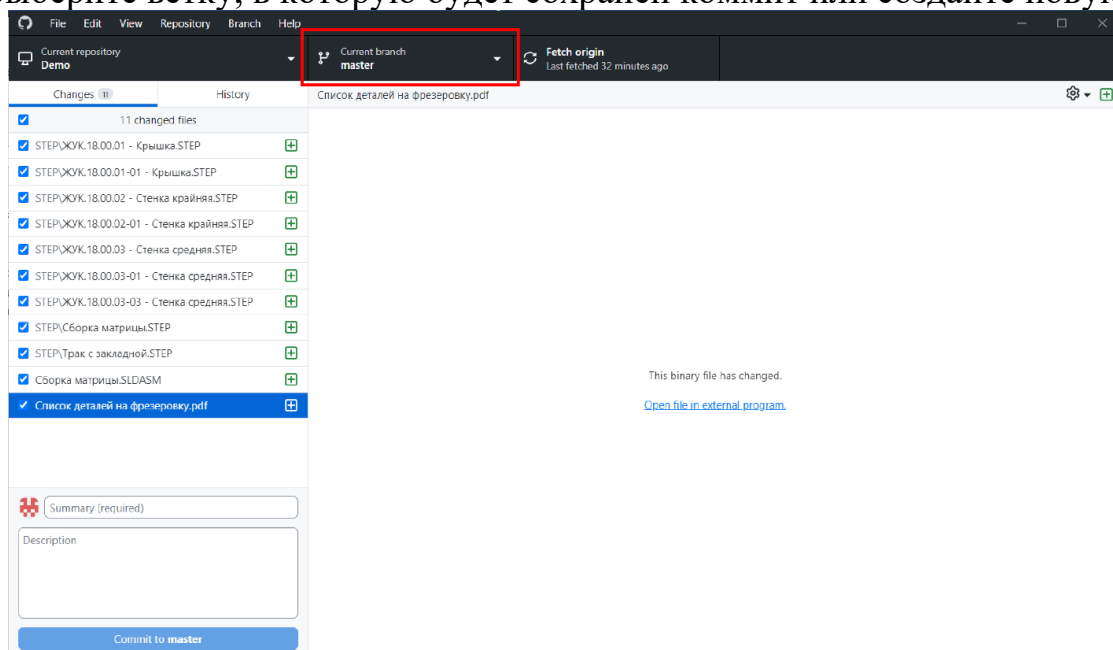
5 ЗАГРУЗКА ИЗМЕНЕНИЙ НА СЕРВЕР

Загрузка изменений на сервер – это основной инструмент контроля версий. Для этого выполните следующую последовательность действий.

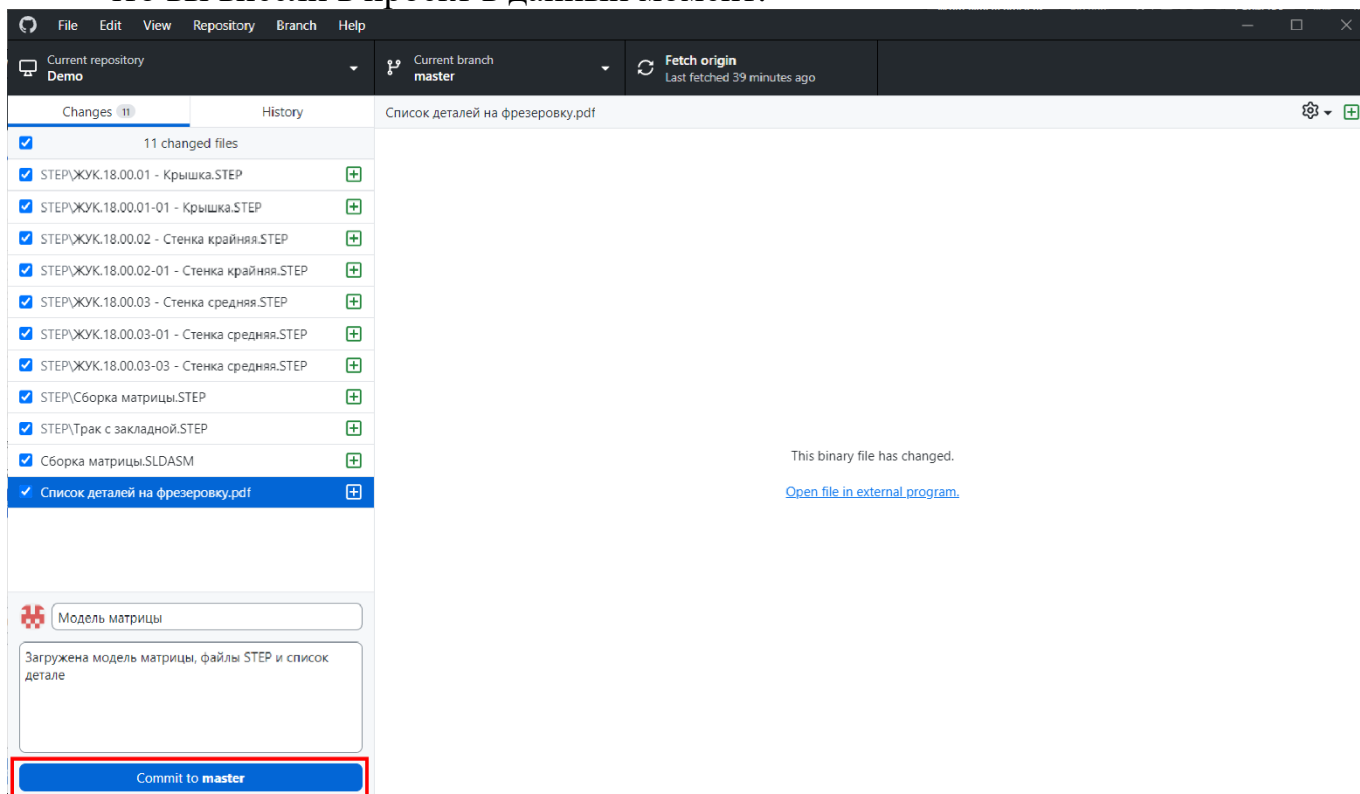
1. После внесения изменений в рабочие файлы обновленные версии появятся в вашей локальной копии репозитория. Новые и измененные файлы отобразятся в проводнике и приложении GitHub Desktop.



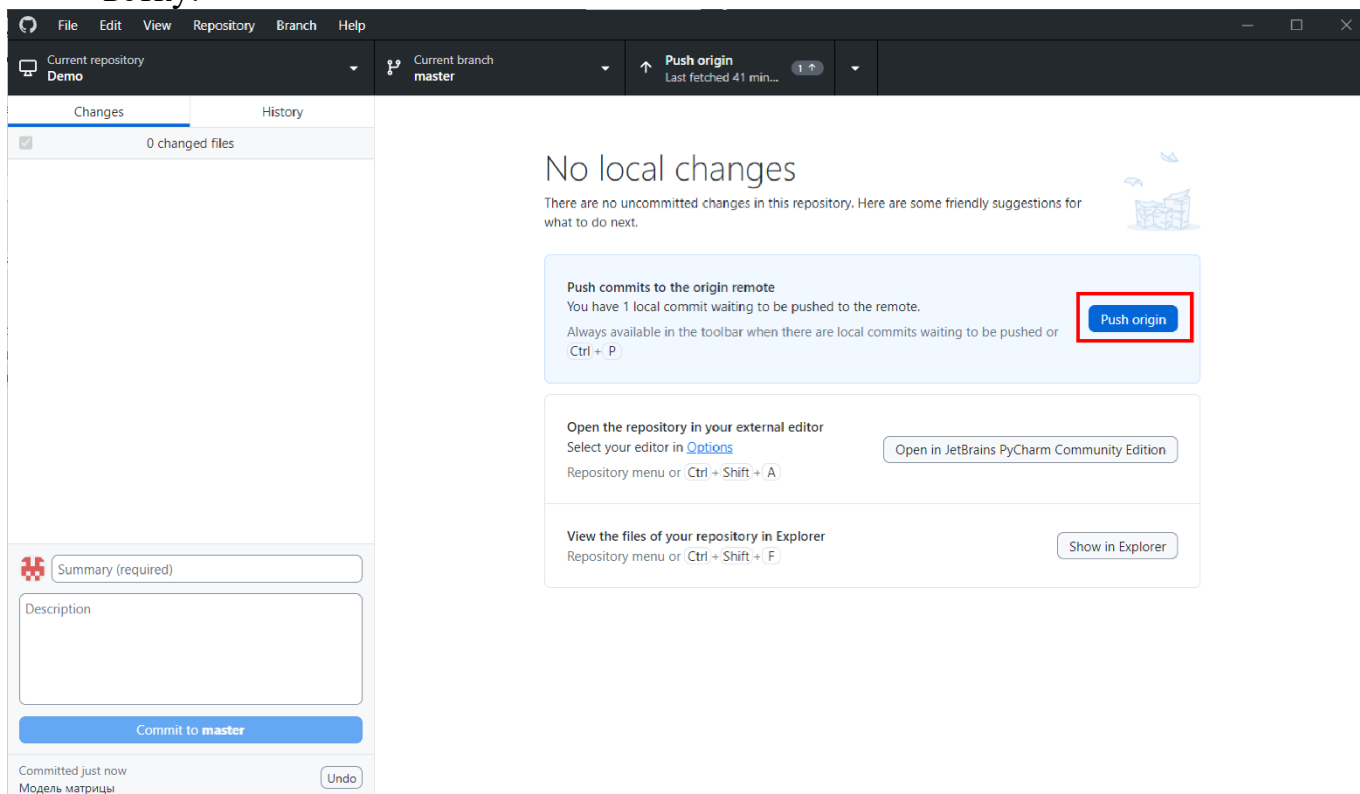
2. Заполните информацию по изменениям, которые вы внесли в проект. Summary – название коммита (может содержать дату или основные изменения), Description – подробно описывает изменения, которые вы внесли. Уровень подробности определяет сам автор по согласованию с соавтором, если он есть.
3. Выберите ветку, в которую будет сохранен коммит или создайте новую.



4. Нажмите кнопку «Commit to ...». Это действие создаст коммит тех изменений, что вы внесли в проект в данный момент.



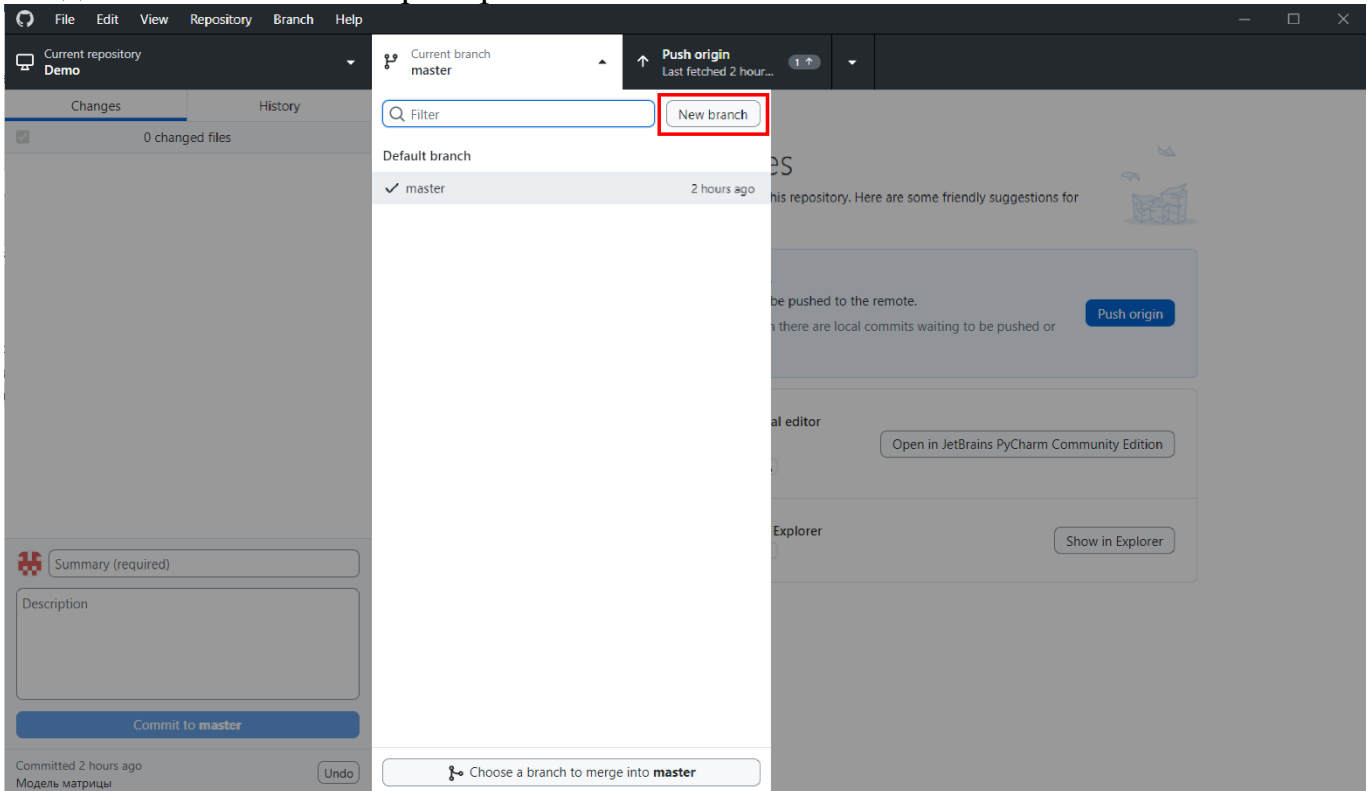
5. После этого остается только отправить изменения на сервер кнопкой «Push origin». После этого ваши изменения будут загружены на сервер в указанную ветку.



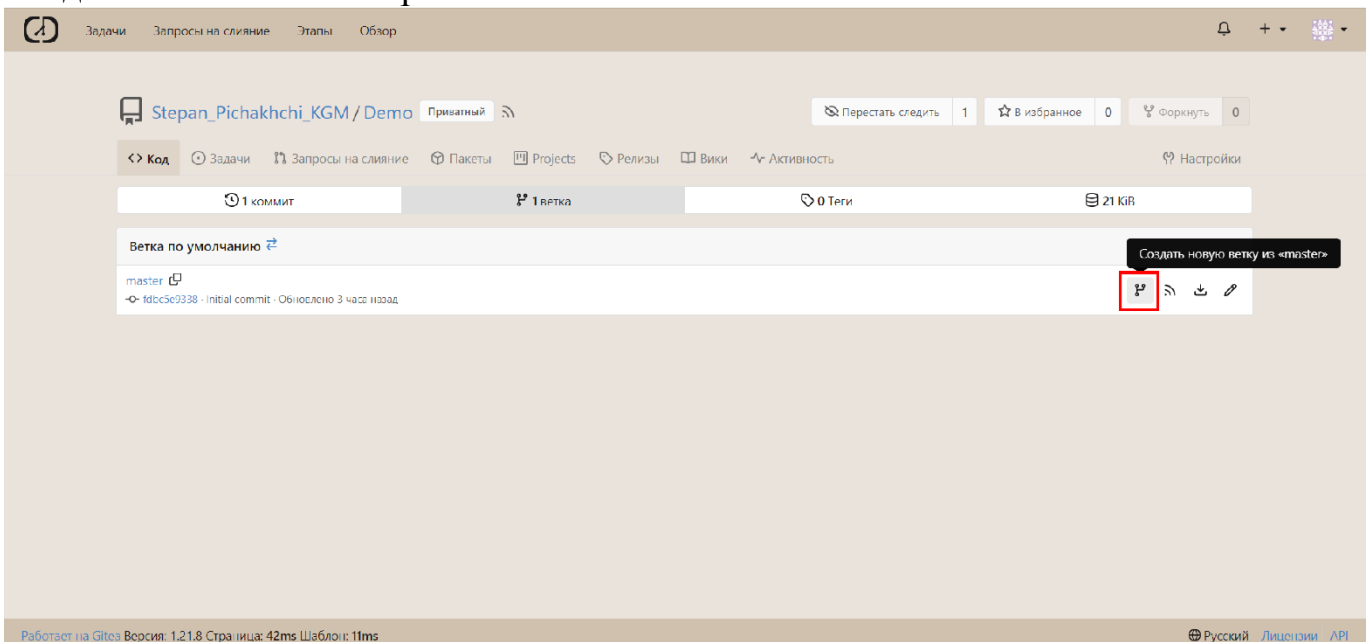
6 СОЗДАНИЕ НОВОЙ ВЕТВИ

Создать новую ветвь можно двумя способами: через интерфейс GitHub Desktop или веб-страницу репозитория.

Создание новой ветви через приложение:



Создание новой ветви через сайт:



7 СЛИЯНИЕ ВЕТВЕЙ

7.1 ОТПРАВКА ЗАПРОСА

Запросы на слияние отправляются автором изменений или владельцем репозитория. Рекомендуется осуществлять эту процедуру через сайт, чтобы иметь возможность предварительного просмотра всех измененных файлов. Автор запроса может предоставить комментарии и сопроводительные материалы для обоснования своих изменений.

Отправка запроса на слияние через сайт:

The screenshot displays the GitHub web interface for creating a new pull request. The page title is "Новый запрос на слияние" (New pull request). The interface shows a comparison between the "master" branch and a new branch named "new_material". The commit list shows one commit by "PichKH" with the message "Новая ветвь". The file diff for README.md is visible, showing changes to the repository description.

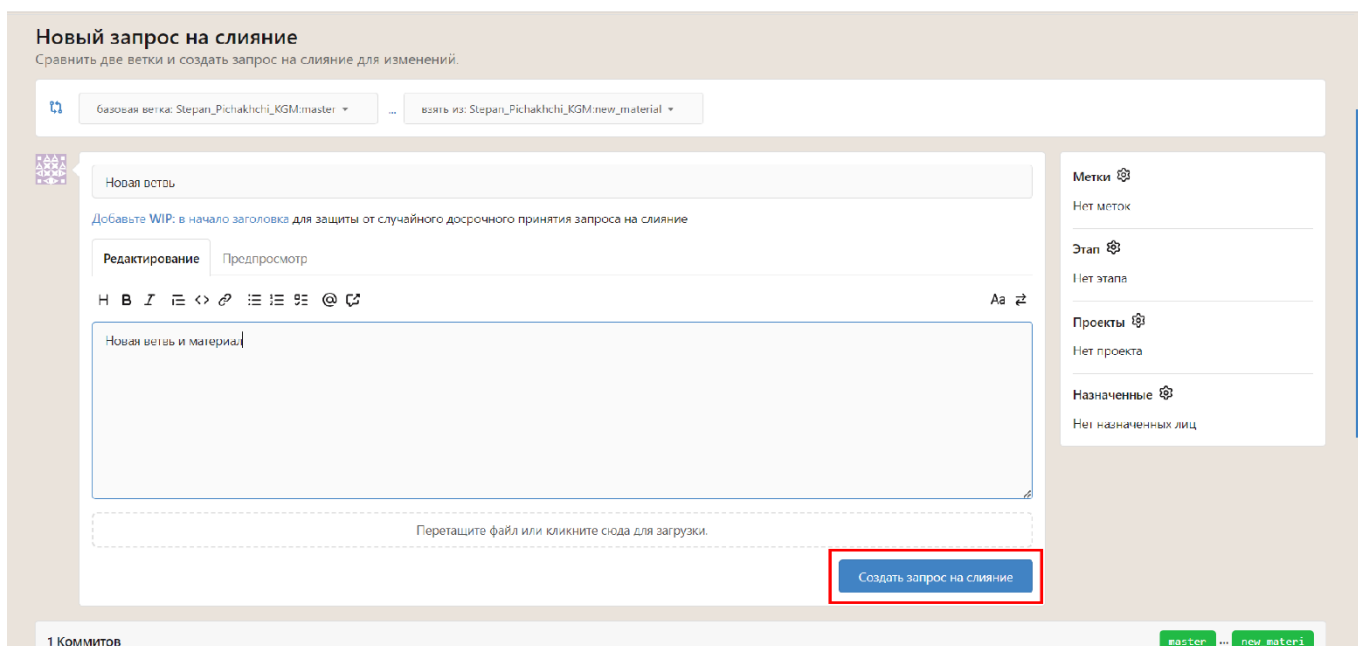
1 Коммитов

| Автор | SHA1 | Сообщение | Дата |
|--------|------------|-------------|---------------|
| PichKH | ca5ca9b687 | Новая ветвь | 5 минут назад |

1 изменённых файлов: 3 добавлений и 1 удаление

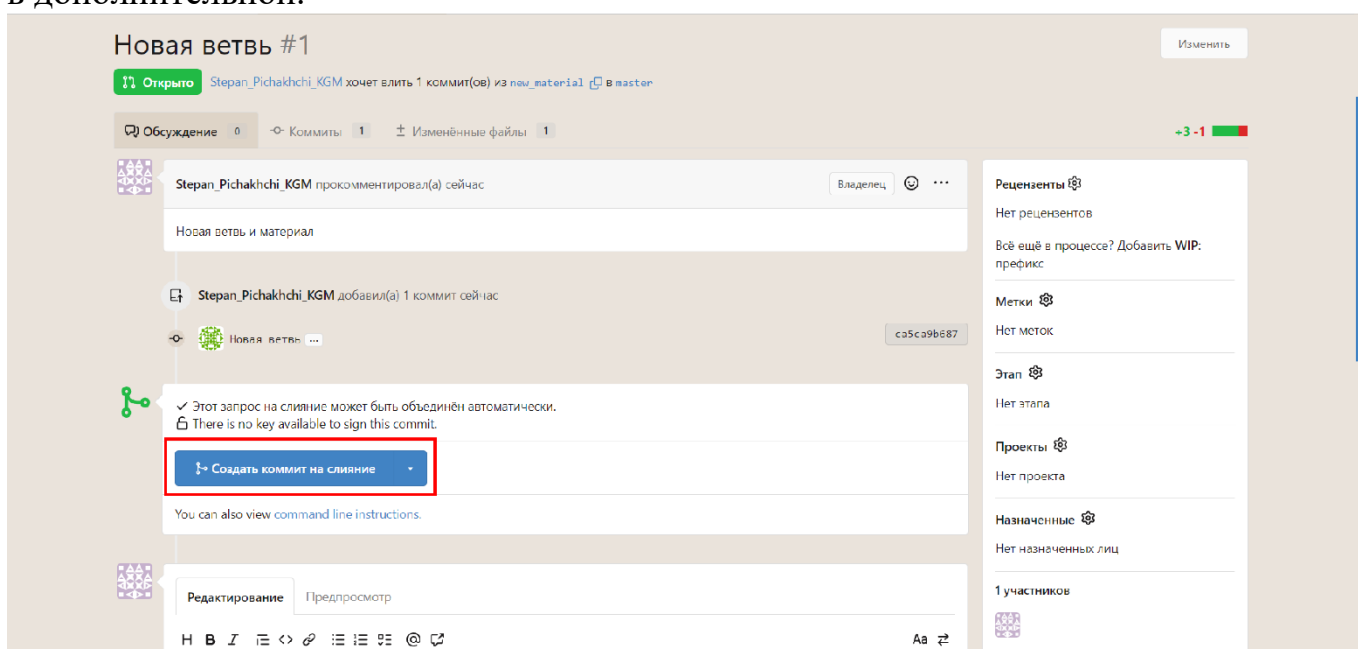
README.md

```
@@ -1,3 +1,5 @@
1 1 # Demo
2 2
3 3 - Демонстрационный репозиторий
4 4 + Демонстрационный репозиторий
> 5 - В новой ветви задан новый материал.
```



7.2 СЛИЯНИЕ ВЕТВЕЙ ЧЕРЕЗ СОЗДАНИЕ КОММИТА

После отправки запроса владелец репозитория или ответственное лицо проверяет изменения и создает коммит в основной ветви с изменениями, сделанными в дополнительной.



ЗАКЛЮЧЕНИЕ

Система управления версиями GitCAD предоставляет эффективный механизм для организации коллективной работы над проектами, обеспечивая контроль за изменениями и облегчая интеграцию результатов труда различных участников коллектива.