Министерство образования Новосибирской области

ГБПОУ НСО «Новосибирский авиационный технический колледж имени Б.С. Галущака»

|  |  |  |
| --- | --- | --- |
| УТВЕРЖДАЮ  Председатель цикловой  комиссии УГС 09.00.00 Информатика и вычислительная техника  \_\_\_\_\_\_\_\_\_\_\_ О.О. Чекушкина  Протокол № 1 от  « 15 » сентября 2023 г | УТВЕРЖДАЮ  Председатель цикловой  комиссии УГС 09.00.00 Информатика и вычислительная техника  \_\_\_\_\_\_\_\_\_\_\_ О.О. Чекушкина  Протокол №\_\_\_\_\_ от  « \_\_ » \_\_\_\_\_\_\_\_\_\_ 20\_\_\_ г | УТВЕРЖДАЮ  Председатель цикловой  комиссии УГС 09.00.00 Информатика и вычислительная техника  \_\_\_\_\_\_\_\_\_\_\_ О.О. Чекушкина  Протокол №\_\_\_\_\_ от  « \_\_ » \_\_\_\_\_\_\_\_\_\_ 20\_\_\_ г |

**Публикация проекта в Git**

Методические указания к самостоятельной работе студента 1

Профессиональный модуль: ПМ.01 Разработка модулей программного обеспечения

для компьютерных систем

Междисциплинарный курс: МДК.01.01 Разработка программных модулей

Специальность: 09.02.07 Информационные системы и программирование

Разработал:

Катаев М.Д.

2023

**1 Цели**

1.1 В ходе выполнения работы студенты осваивают:

1.1.1 Общие компетенции:

* ОК.01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам;
* ОК.02 Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности;
* ОК.09 Использовать информационные технологии в профессиональной деятельности.

1.1.2 Профессиональные компетенции:

* ПК.1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
* ПК.1.2 Разрабатывать программные модули в соответствии с техническим заданием.

1.2 В результате выполнения практических заданий студенты:

1.2.1 Осваивают умения:

* применять архитектурный подход для разработки программного продукта;
* разрабатывать и специфицировать требования к программным модулям;
* проектирования информационную систему.

1.2.2 Усваивают знания:

* определение модуля и модульной разработки программного обеспечения;
* принципы модульной разработки программного обеспечения;
* принципы архитектурного подхода для разработки программного продукта.

**2 Дидактическое обеспечение**

2.1 методические рекомендации по выполнению работы;

2.2 пример выполнения самостоятельной работы (Приложение А);

**3 Форма организации:** индивидуальная

**4 Инструктаж**

4.1 работа выполняется в виде электронного документа;

4.2 работа сдается преподавателю в СДО Moodle путем загрузки файла электронного документа.

4.3 Время выполнения работы — 90 минут

**5 Порядок выполнения**

5.1 ознакомиться с методическими указаниями по выполнению работы;

5.2 ознакомиться с содержанием задания;

5.4 выполнить задание в соответствии с индивидуальной темой разработки;

5.5 отправить работу на проверку преподавателю.

**6 Методические рекомендации**

Методические рекомендации представлены в приложении А

**7 Форма отчета**

7.1 Номер и название практического занятия

7.2 Цели практического занятия

7.3 Ход выполнения работы

**8 Содержание работы**

Создать репозиторий в системе контроля версий Git, задать ему название «MyFirstApp». После чего произвести установку GitBash на свой персональный компьютер. Затем разместить проекта из практического занятия 1, в созданный репозиторий. Оформить отчет по выполненной работе. В отчете должно быть подробно описана установка командной строки, создание репозитория, размещения проекта в репозитории, а также расписаны вызываемые команды.

**9 Критерии оценок**

9.1 При оценивании результатов выполнения задания учитываются следующие критерии:

* практическое задание выполнено в соответствии с методическими рекомендациями, в установленное время;
* репозиторий назван в соответствии с заданием;
* размещение проекта в репозитории совершено при помощи командной строки Git Bash;
* отчет выполнен согласно заданию;
* наличие синтаксических/орфографических ошибок в тексте.

9.2 В основу оценки выполненных заданий положен принцип:

* «отлично» — выполнен полный объем заданий, в соответствии с п. 9.1, и сдачи задания в заданные сроки;
* «хорошо» — выполнен полный объем заданий в соответствии с п. 9.1, и сдачи задания в заданные сроки, но допущены ошибки;
* «удовлетворительно» — выполнен полный объем задания в соответствии с п. 9.1, с допущением незначительных ошибок по оформлению документа;
* «неудовлетворительно» — задание не выполнено, в соответствии с п. 9.1.

ПРИЛОЖЕНИЕ А

(справочное)

**Пример выполнения самостоятельной работы**

**Установка GitBash**

Официальная сборка доступна для скачивания на официальном сайте Git. Перейдите на страницу <https://git-scm.com/download/win> в браузере, после чего выберите необходимую версию, загрузка начнется автоматически.

Ход установки:

1. Перейдите в папку загрузок и запустите загруженный файл;
2. Укажите путь до каталога, в который будет установлен Git;
3. Настройте необходимые параметры установки;
4. Дождитесь завершения процесса установки.

**Создание репозитория**

Сначала необходимо зарегистрироваться и авторизоваться на сайте <https://github.com/> . После чего откроется окно с системой контроля версий, после чего необходимо нажать кнопку «New», рисунок А.1, расположенную в левом верхнем углу окна.

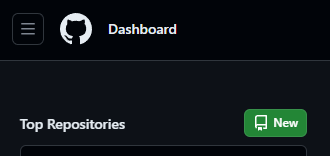


Рисунок А.1 – Кнопка создания нового репозитория

В открывшемся окне необходимо ввести название репозитория, и обозначить его публичным репозиторием, рисунок А.2, затем нажать кнопку «Create a new repository».

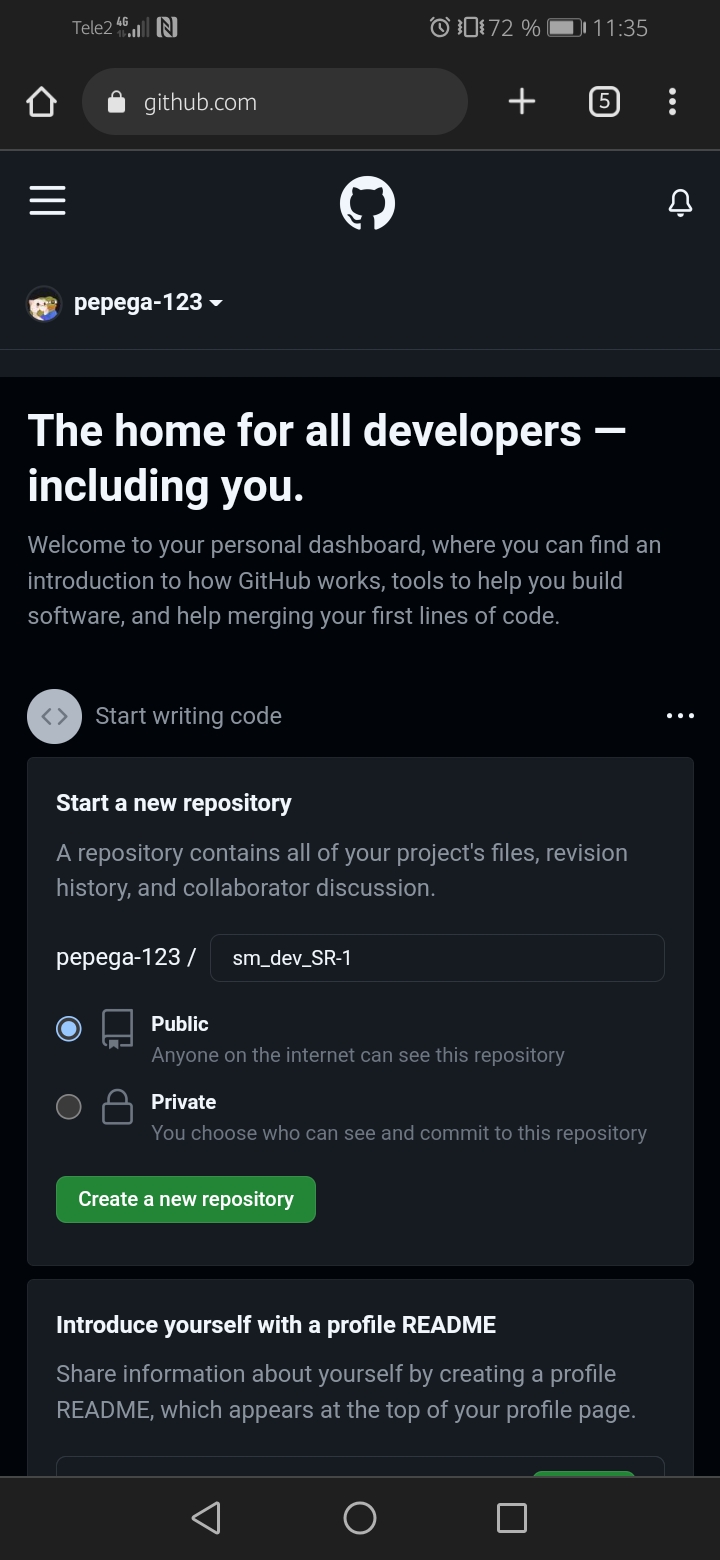


Рисунок А.2 – Создание репозитория

После чего откроется окно, представленное на рисунке А.3, а значит репозиторий был создан успешно.

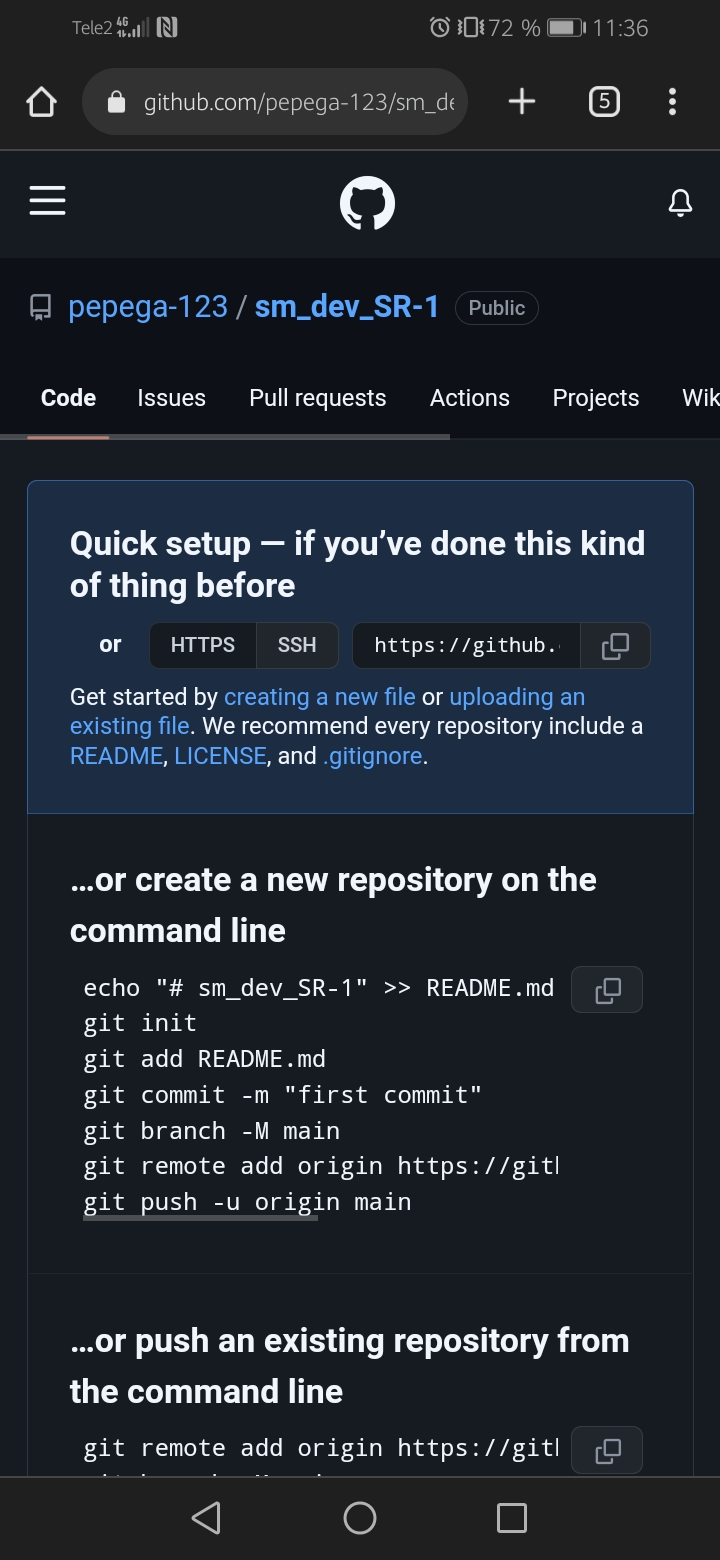


Рисунок А.4 – Страница репозитория

**Добавление проекта в репозиторий**

Сначала необходимо открыть каталог с проектом, который необходимо загрузить в систему контроля версий, после в пустой части каталога следует нажать правой кнопкой мыши (ПКМ) и выбрать пункт «Git Bush Here», рисунок А.4.

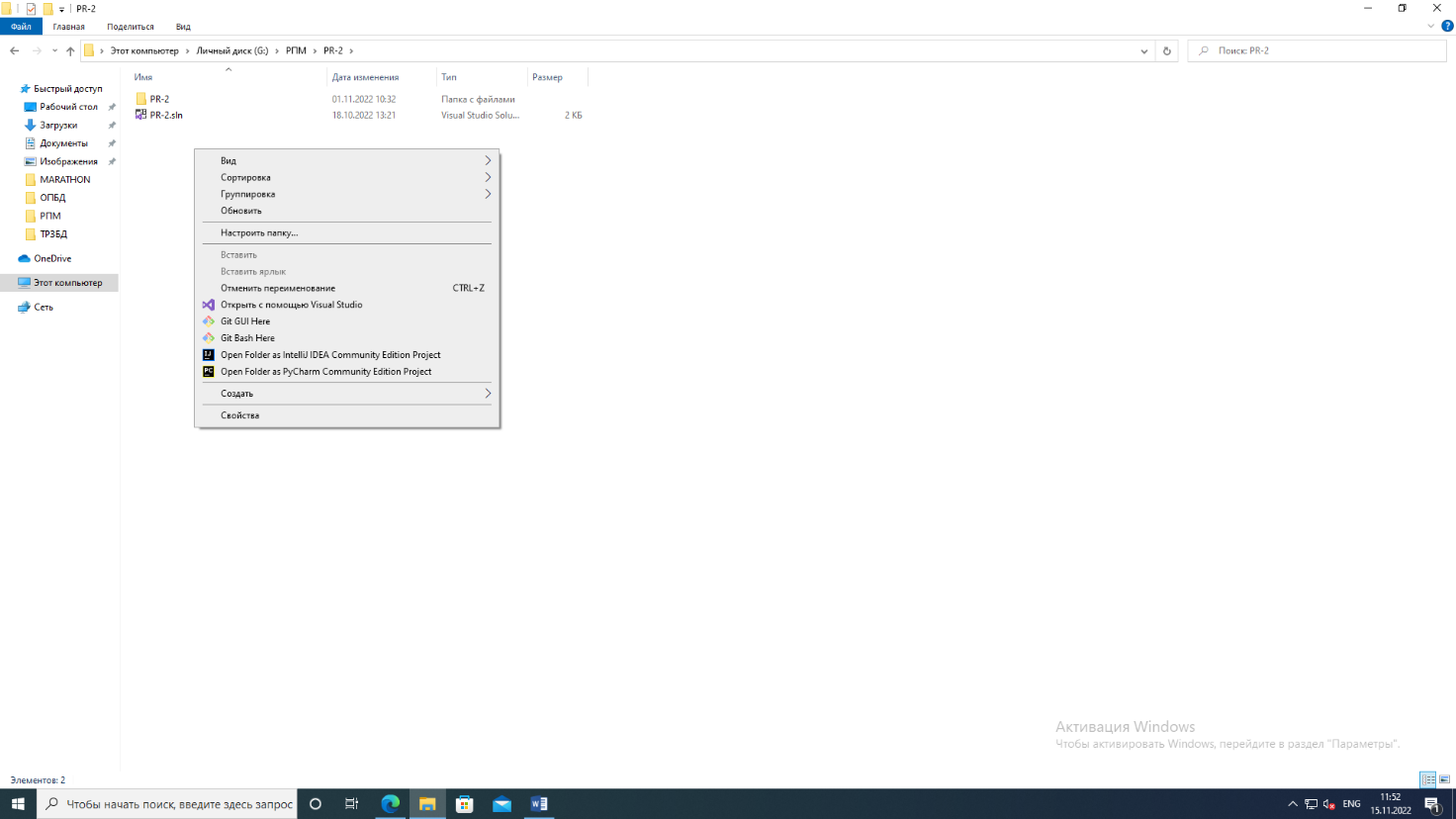


Рисунок А.4 – Вызов командной строки Git Bush

После чего откроется командная строка Git Bash, рисунок А.5

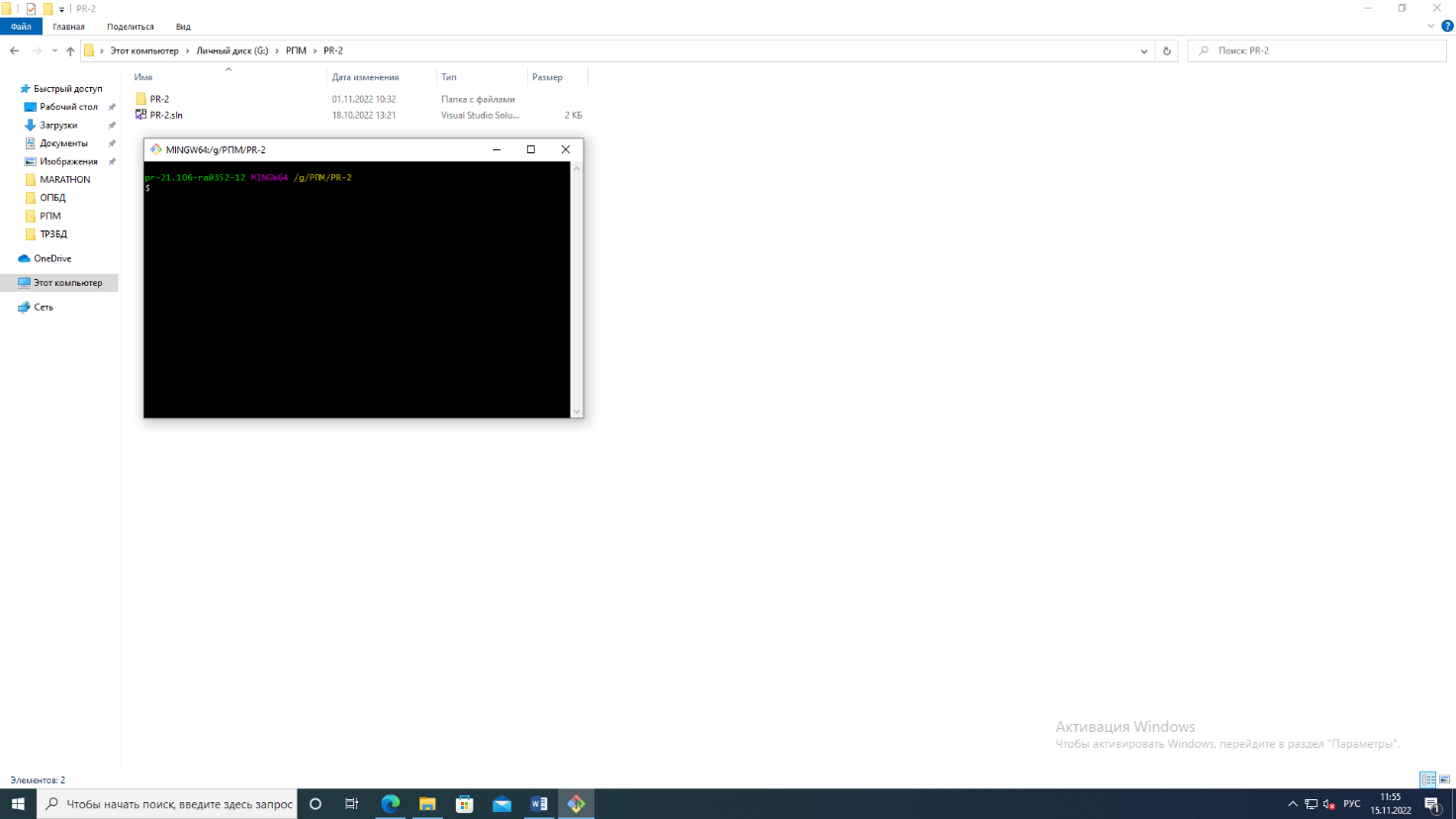


Рисунок А.5 – Командная строка Git Bush

Для того чтобы начать отслеживать содержимое папки в системе, необходимо выполнить представленные ниже команды в командной строке, рисунок А.6:

git init

git add .

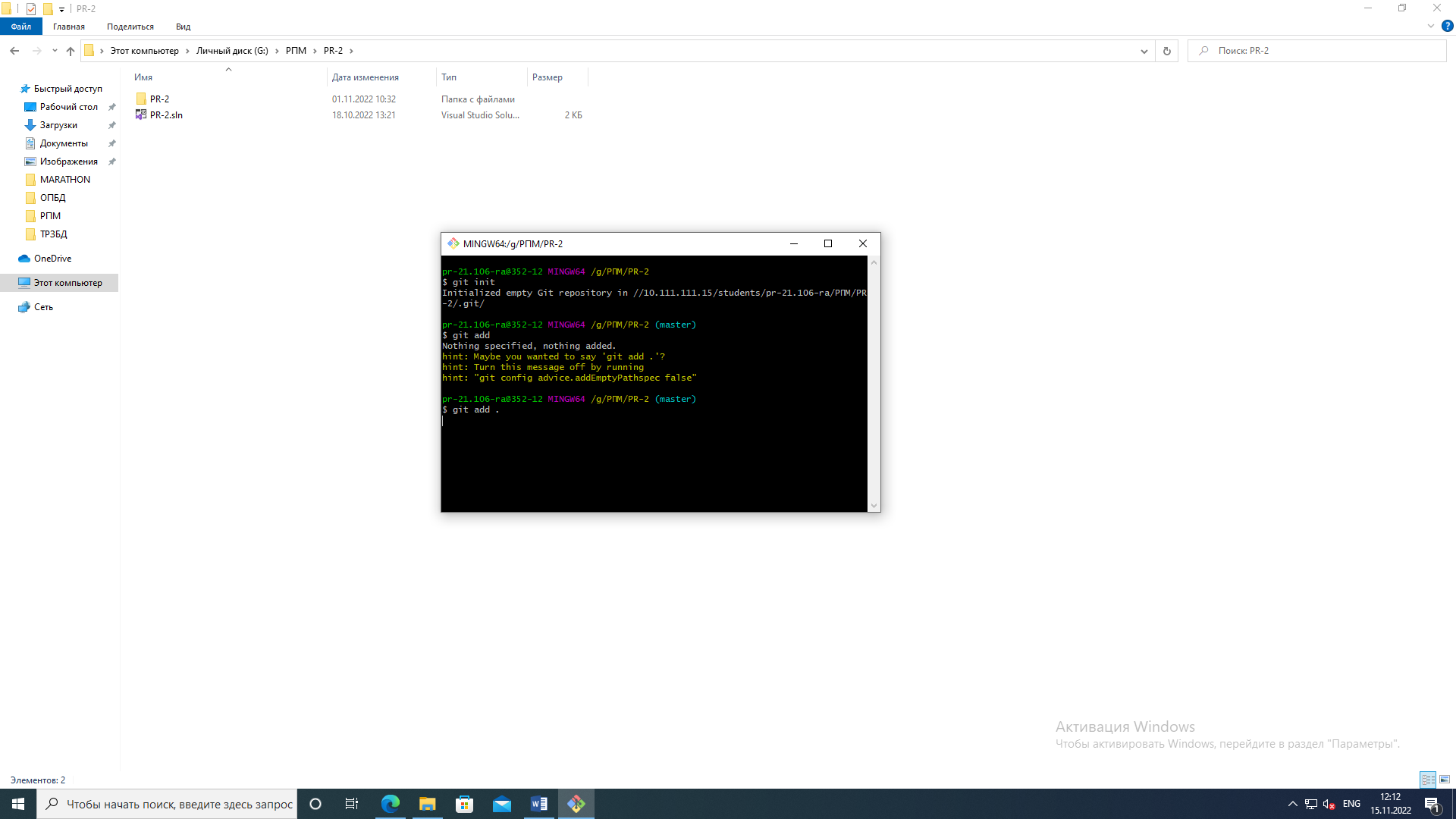


Рисунок А.6 – Команды инициализации локального репозитория

Для настройки необходимо указать ваше имя и электронную почту, рисунок А.7.

git config --local user.email “you@example.com”

git config --local user.name “Ваше имя”

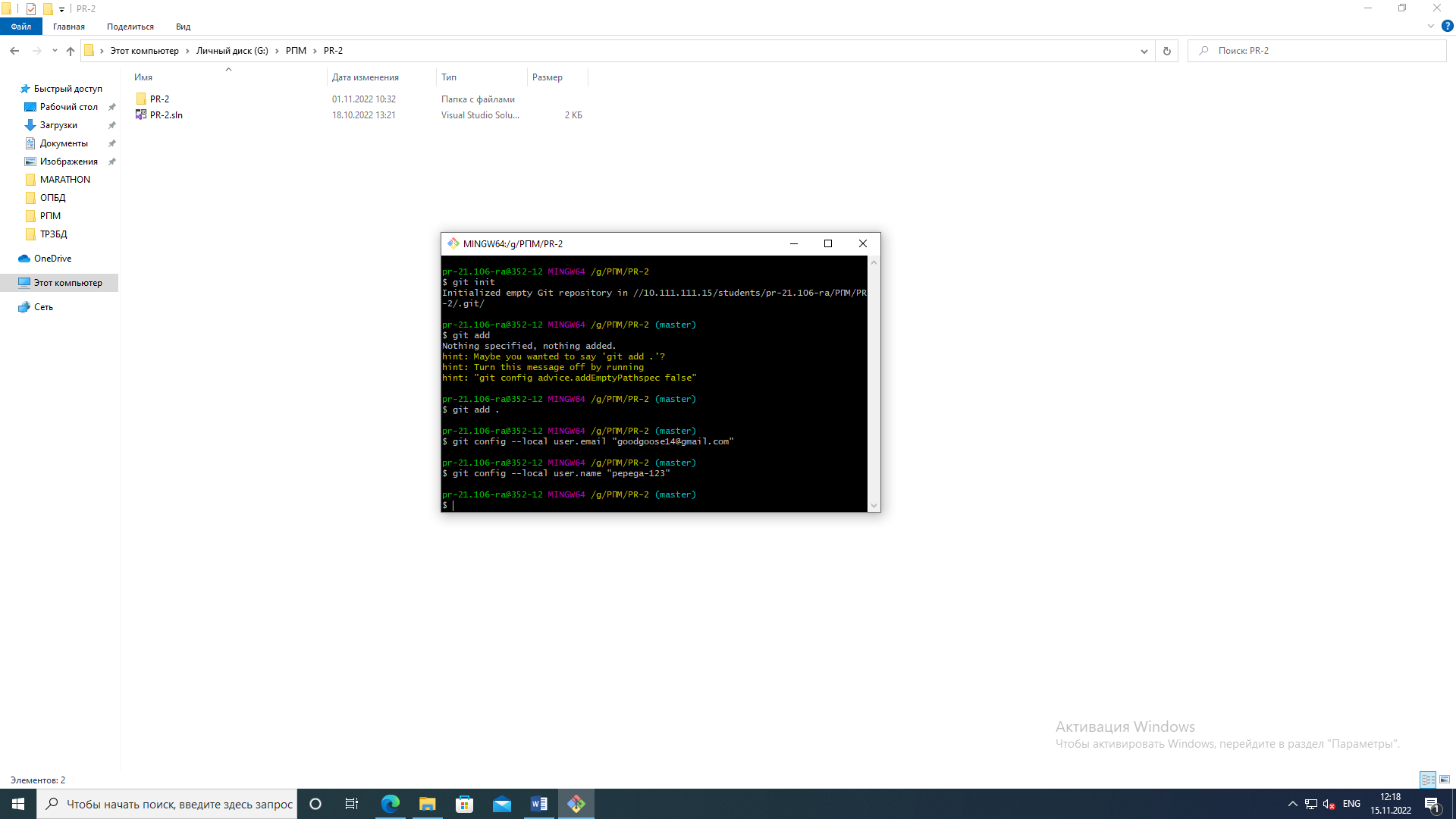


Рисунок А.7 – Подключение к аккаунту системы контроля версий

Дальше необходимо выполнить первый коммит, при помощи команды git commit -m “Initial commit”, рисунок А.8

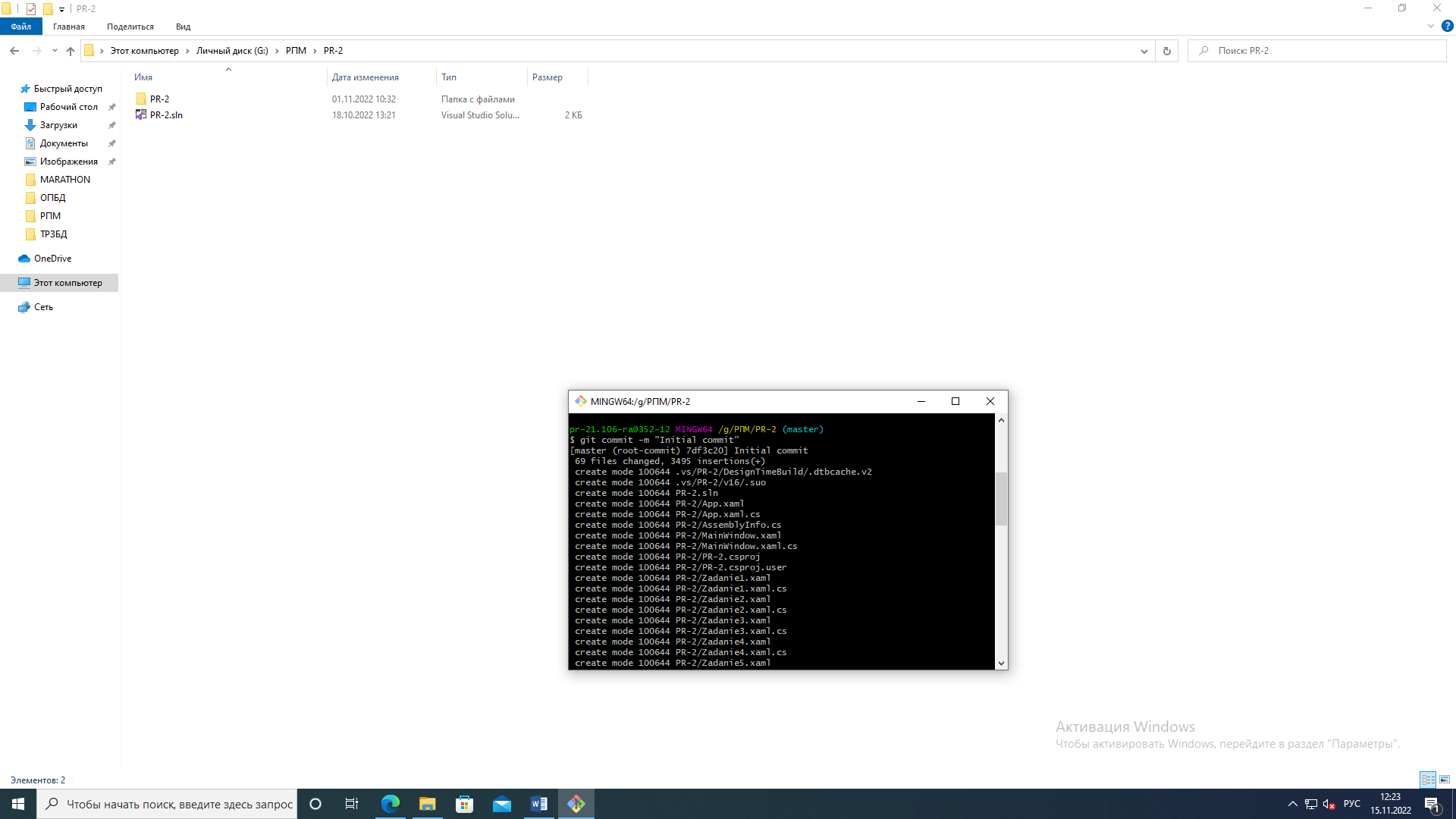


Рисунок А.8 – Первый коммит

Для добавления изменений локального репозитория в созданный репозиторий, необходимо выполнить команды, рисунок А.9:

git remote add origin https://github.com/пользователь/репозиторий.git

git push -u origin master

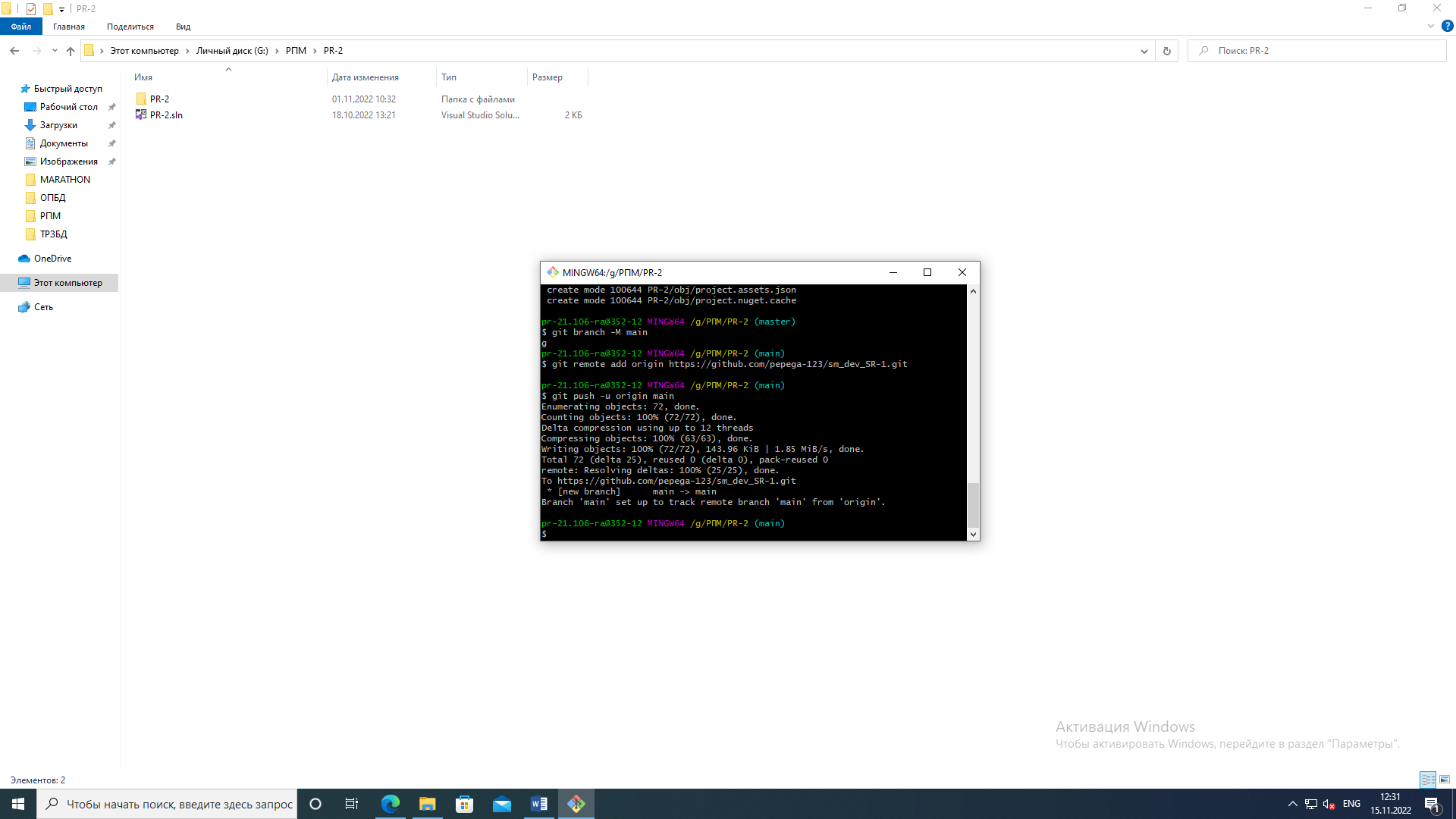


Рисунок А.9 – Добавление изменений в удаленный репозиторий

В конечном итоге, созданный проект будет размещен в репозитории системы контроля версий Git.