ОГБПОУ «ТОМСКИЙ ТЕХНИКУМ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»

Отчет по лабораторно-практической работе №1

учебной дисциплины   
*МДК. 03.01 Моделирование и анализ программного обеспечения*

*Тема: «Создание и изучение возможностей репозитория проекта»*

Выполнили:

студенты 601 группы

Демин К.И. Мальцева Е.В., Белослудцева Н.А.

Проверил:  
преподаватель

Михайлов М.А.

г. Томск – 2022 г

**Тема:** «Создание и изучение возможностей репозитория проекта».

**Цель работы:** получение первоначальных навыков использования систем контроля версий исходного кода программ, получение первоначальных навыков организации разработки программного обеспечения. Создание в системе контроля версий репозитория для нового проекта и выполнение всех основных действий с исходным кодом программы, связанных с контролем версий.

1. Создадим аккаунт на github

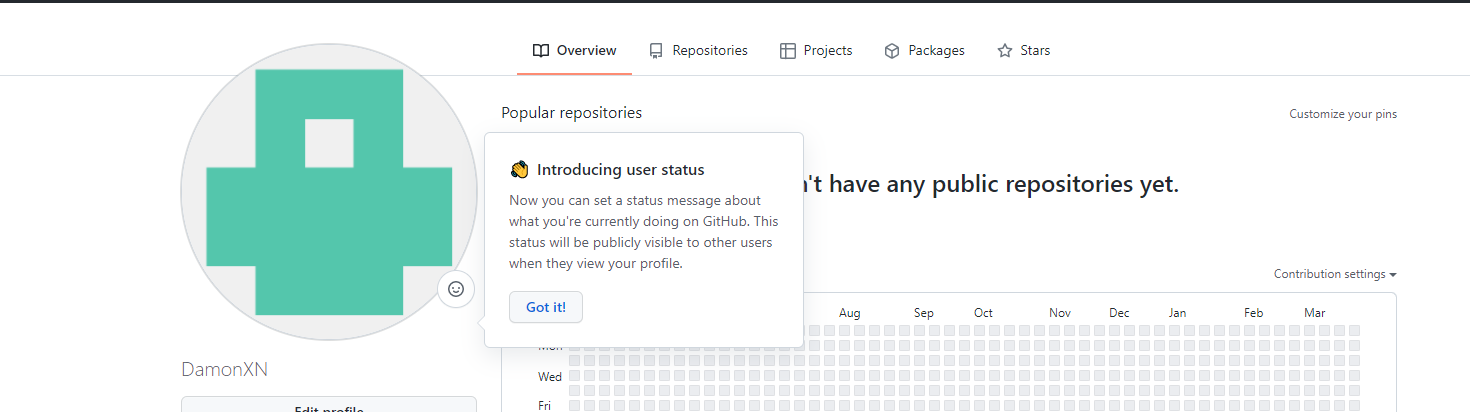


Рис. 1 Аккаунт на GitHub

1. Создадим репозиторий

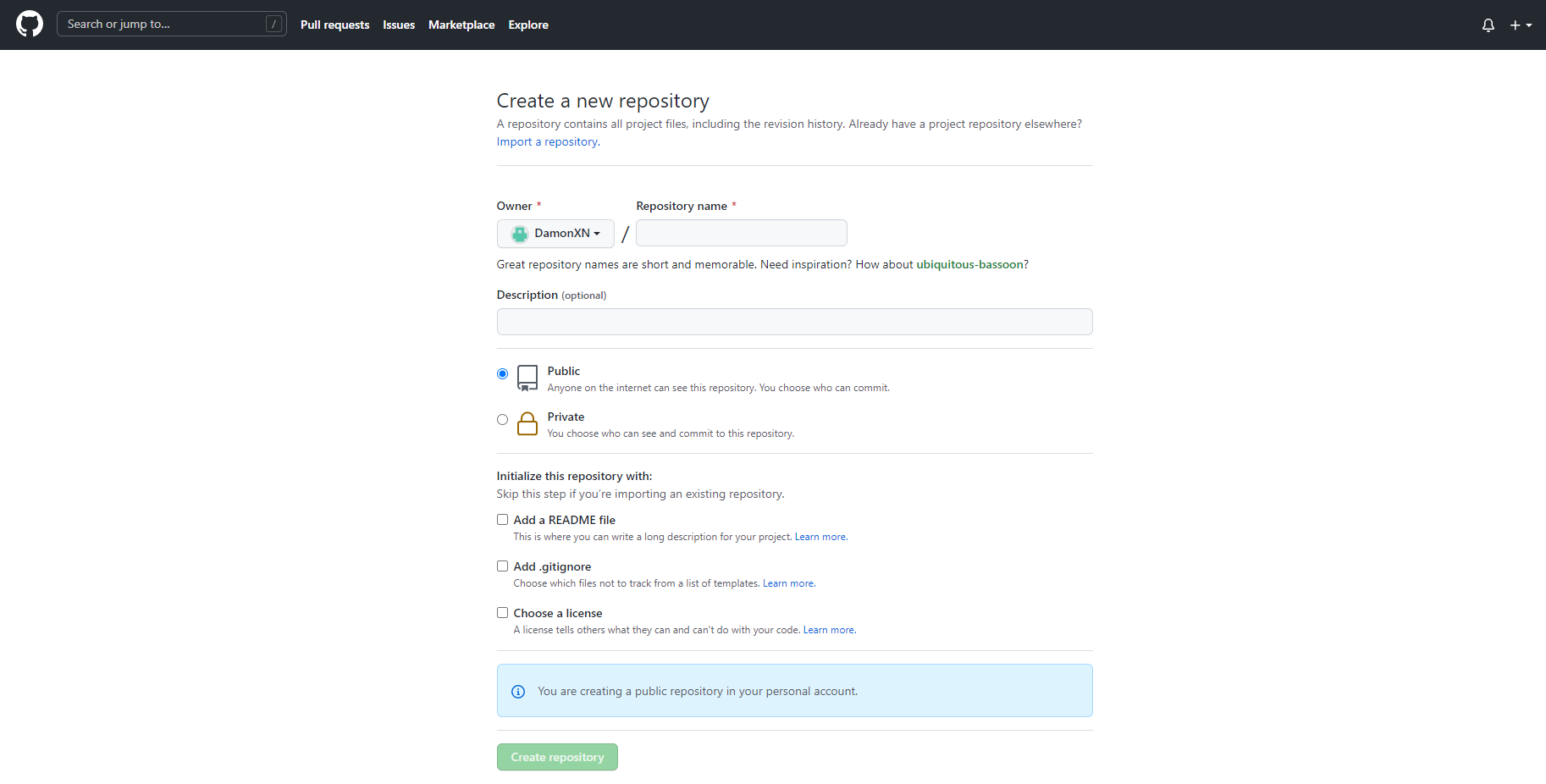


Рис. 2 Создание репозитория

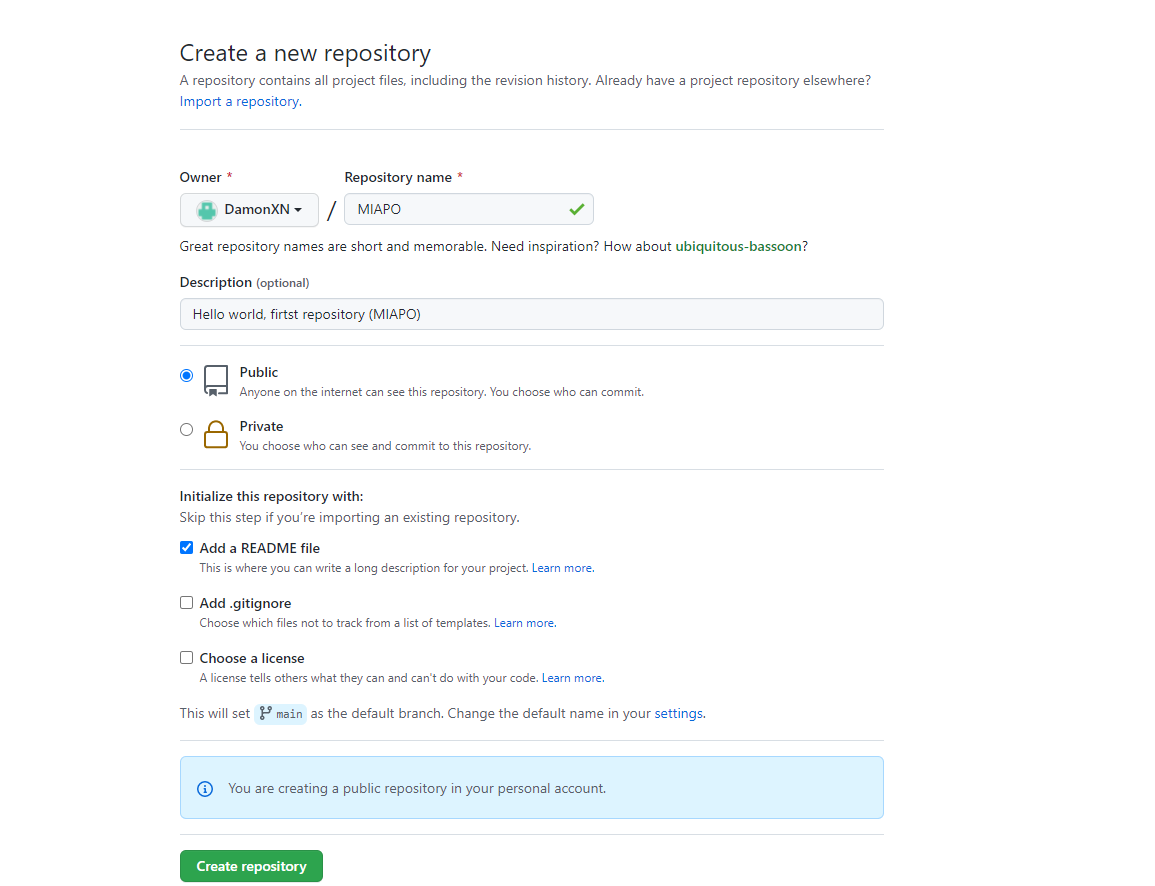


Рис. 3 Заполнение создания репозитория

1. Добавляем ветки в репозиторий

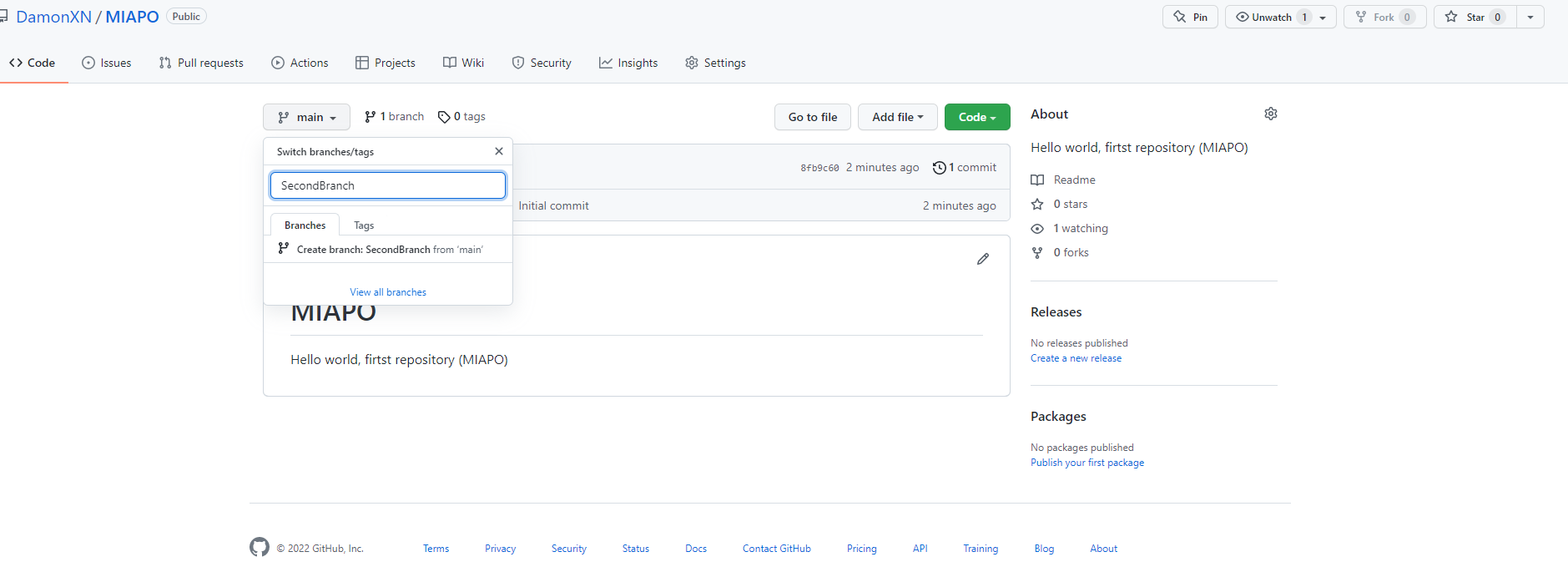


Рис. 4 Добавление названия ветки

1. Изменяем файлы и коммиты

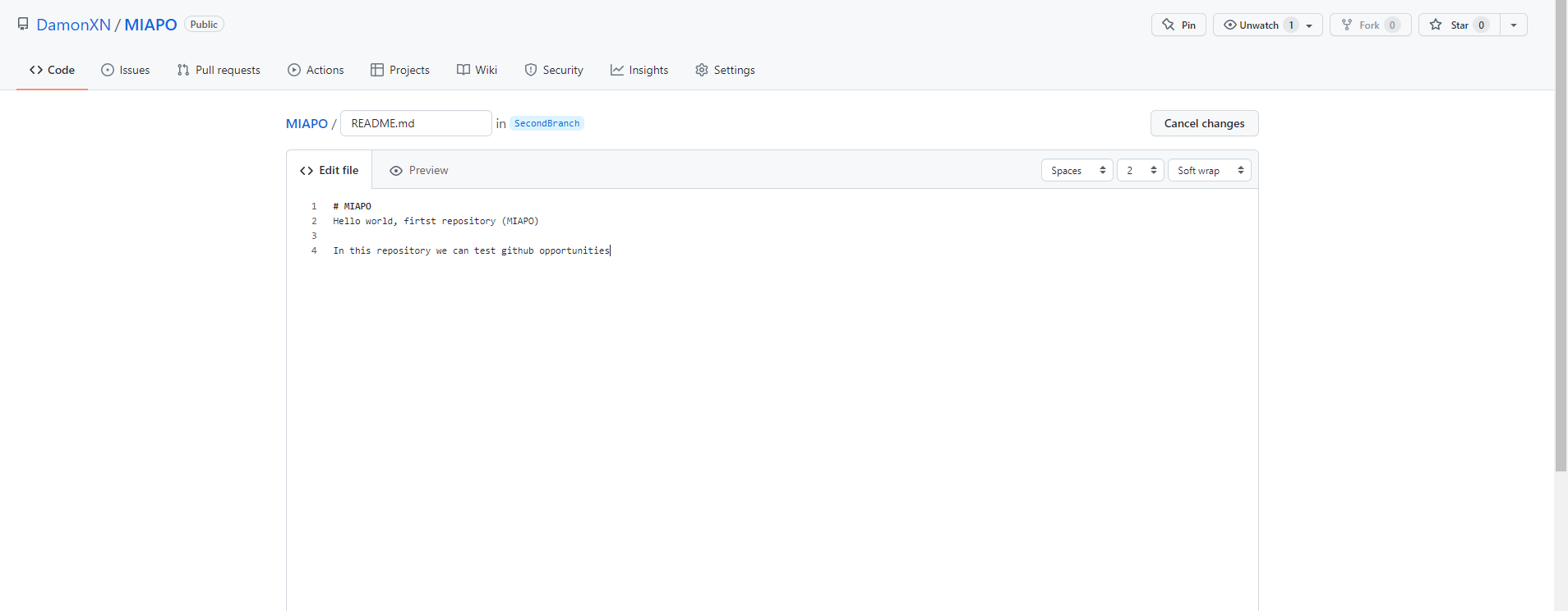


Рис. 5 Измененный ReadME

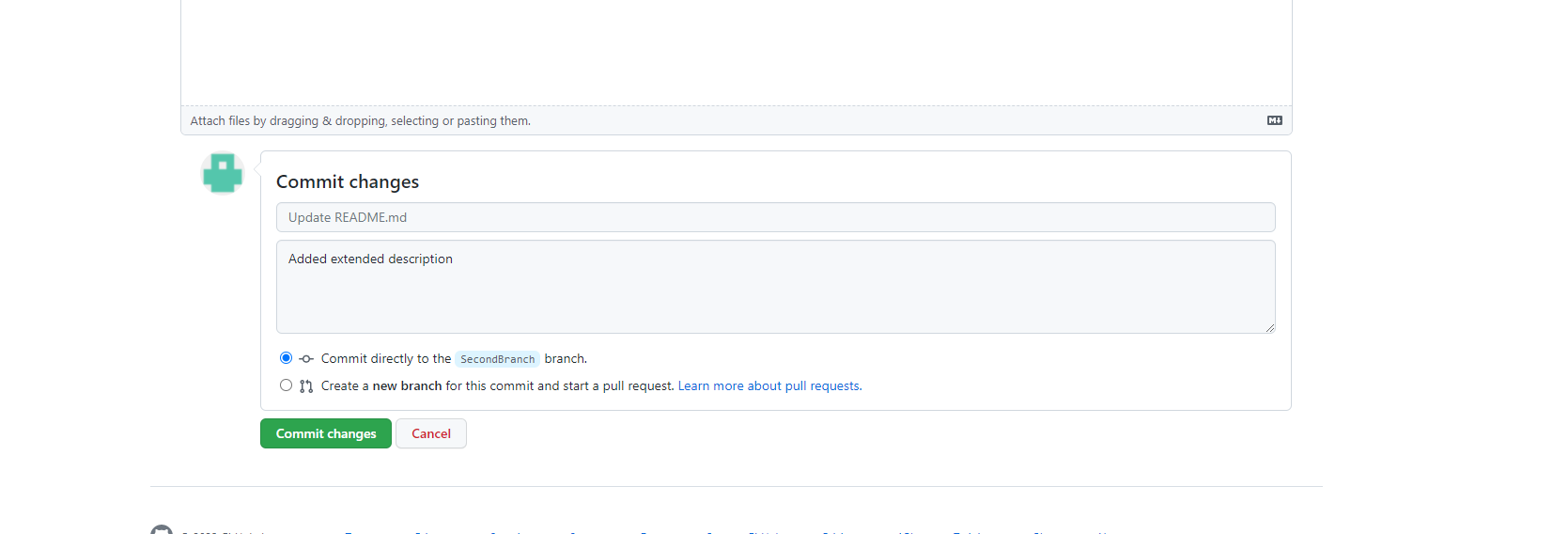


Рис. 6 Краткое изменение коммита

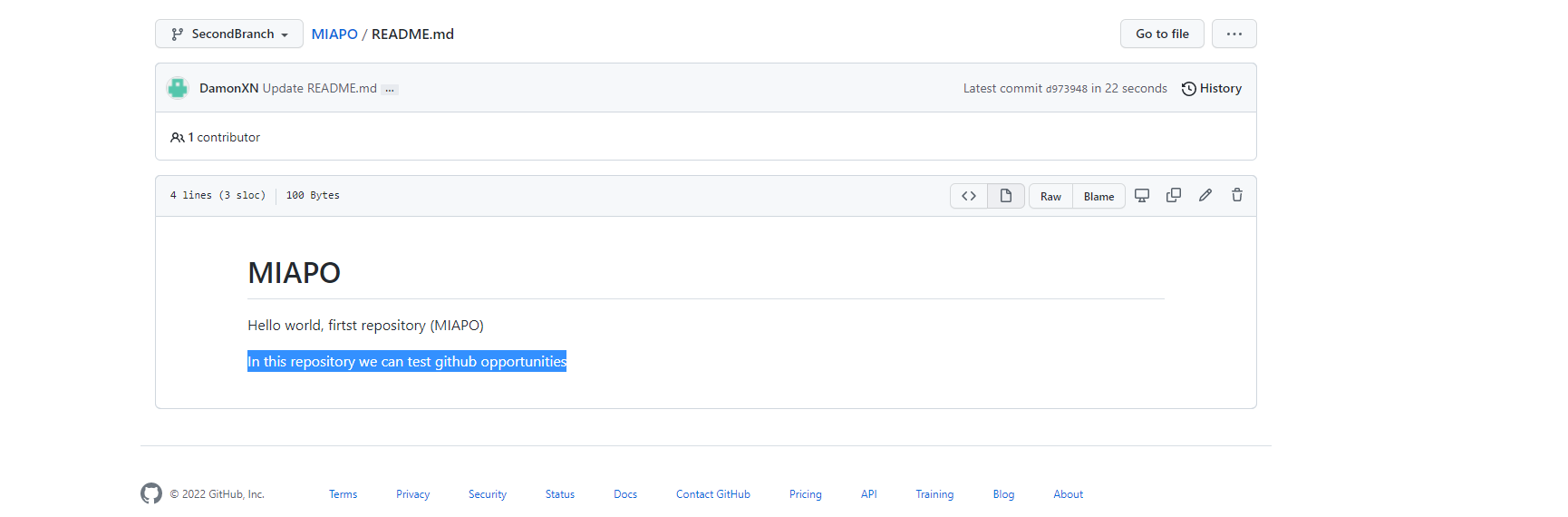


Рис. 7 Результат изменений

1. Создание запросов слияния

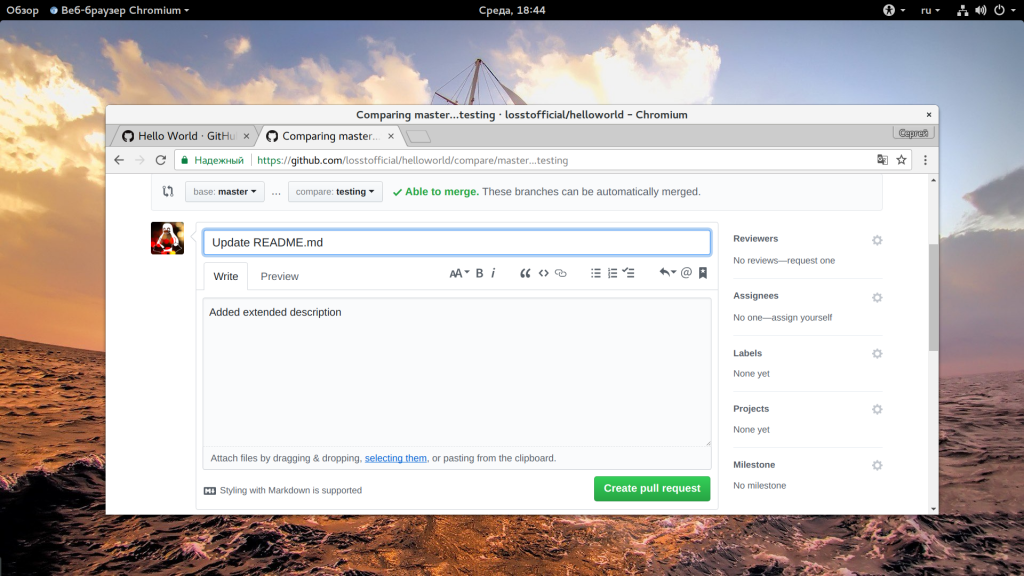
[](https://losst.ru/wp-content/uploads/2017/08/Snimok-ekrana-ot-2017-08-23-18-44-29.png)

Рис. 8 Создаем описание и нажимаем на кнопку create pull request

1. Делаем просмотр и одобрение запросов на слияние

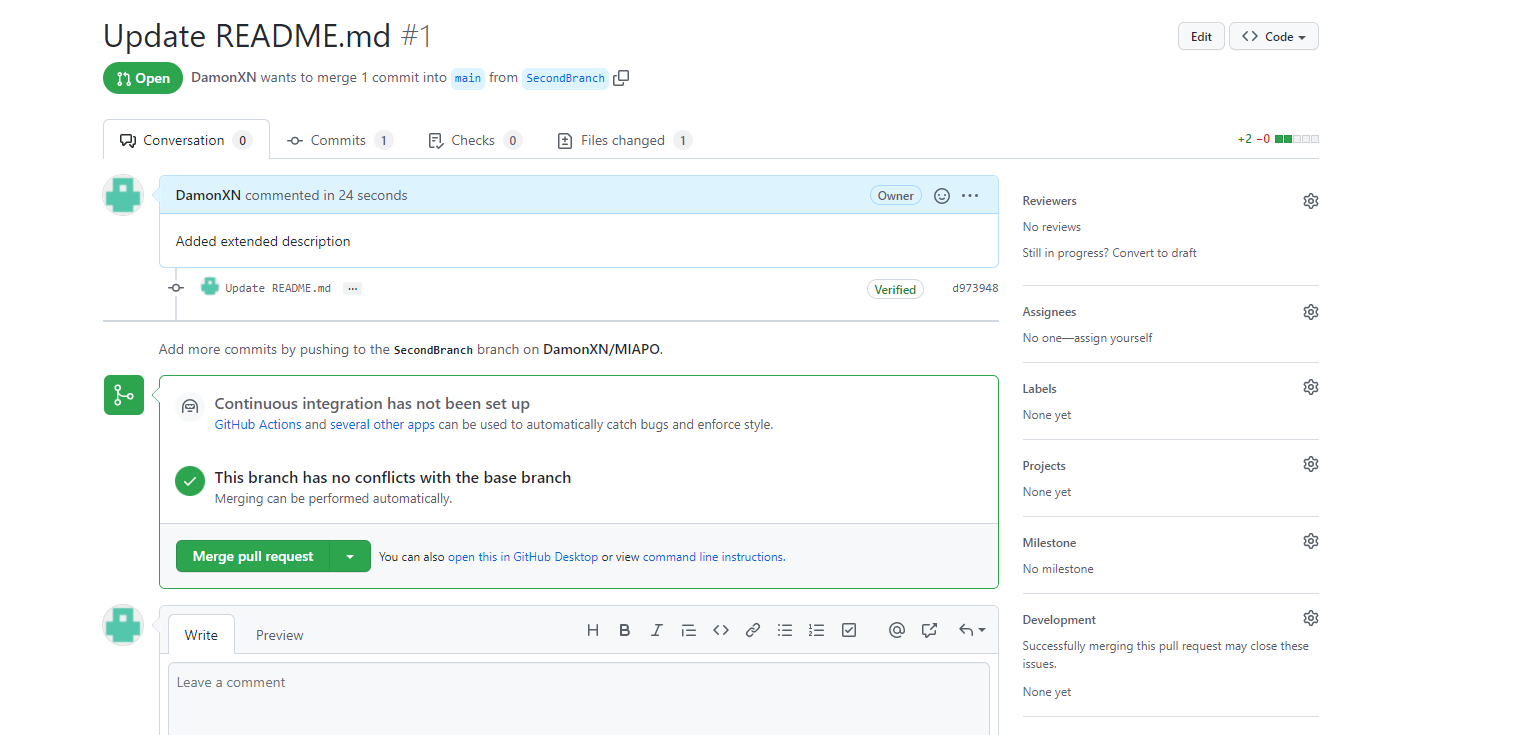


Рис. 9 Принимаем запрос на слияние

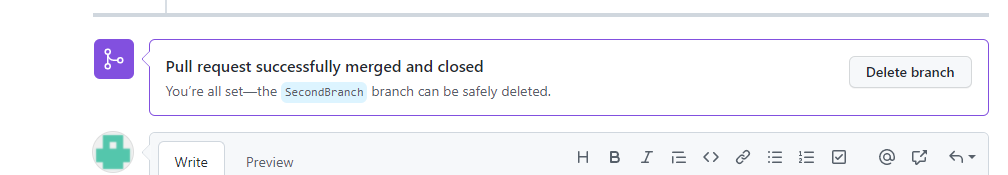


Рис. 10 Подтверждаем Pull Request

1. Делаем отчеты об ошибках

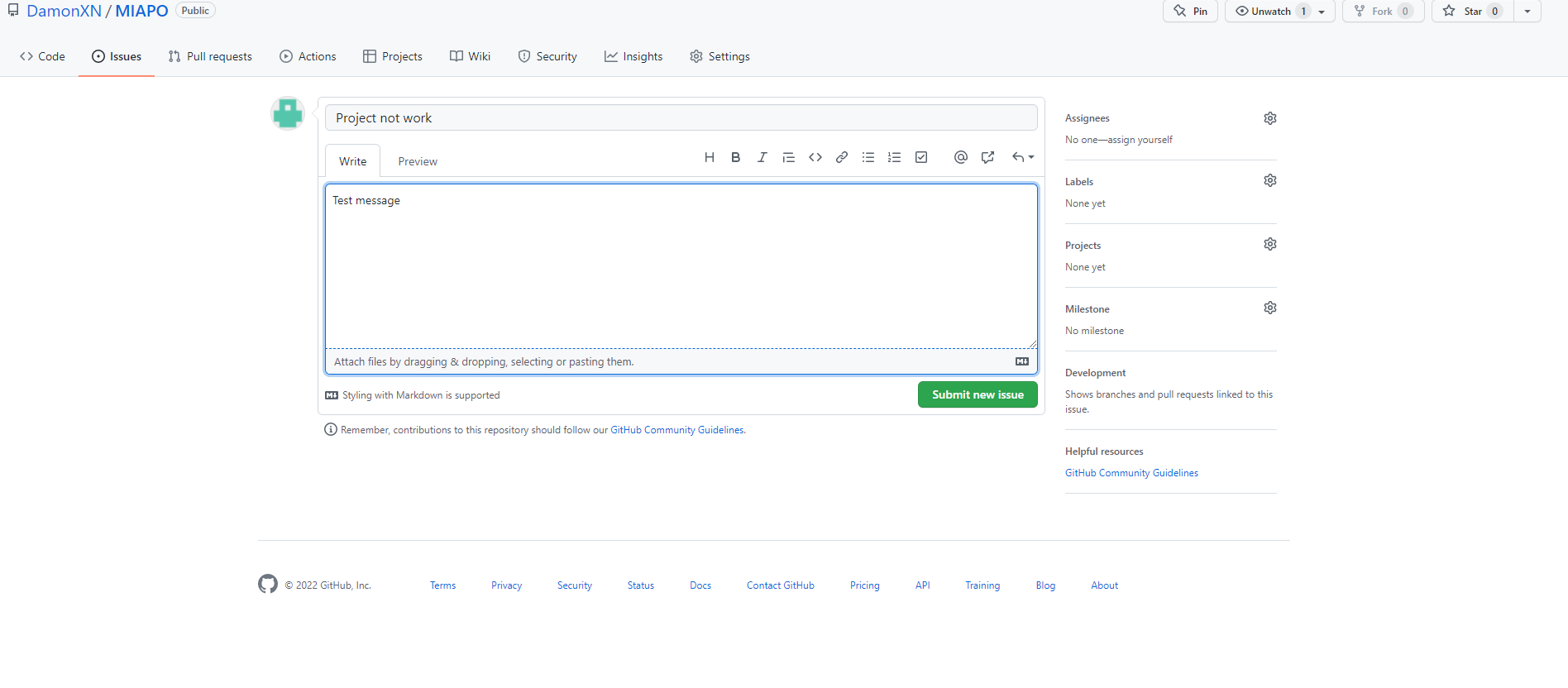


Рис. 11 Создаем сообщение об ошибках, вводим заголовок и текст

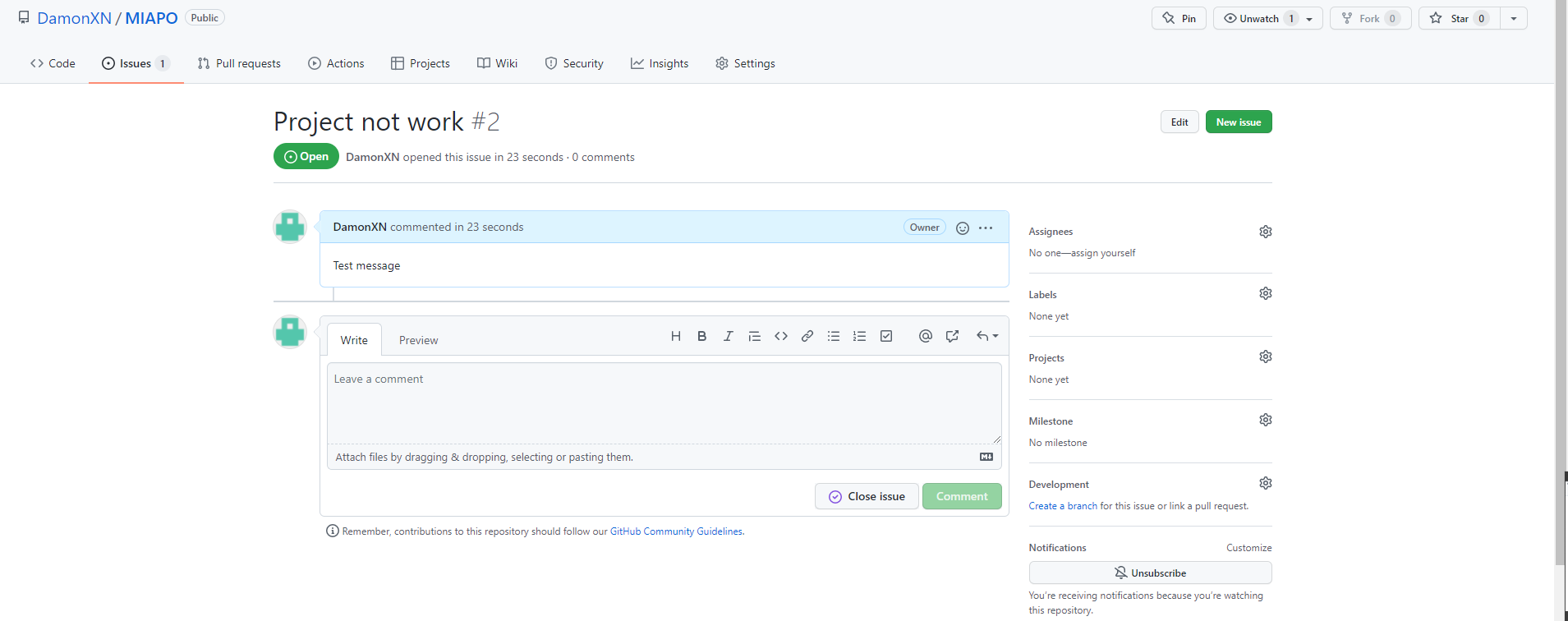


Рис. 12 Созданный отчет об ошибках

1. Делаем релизы

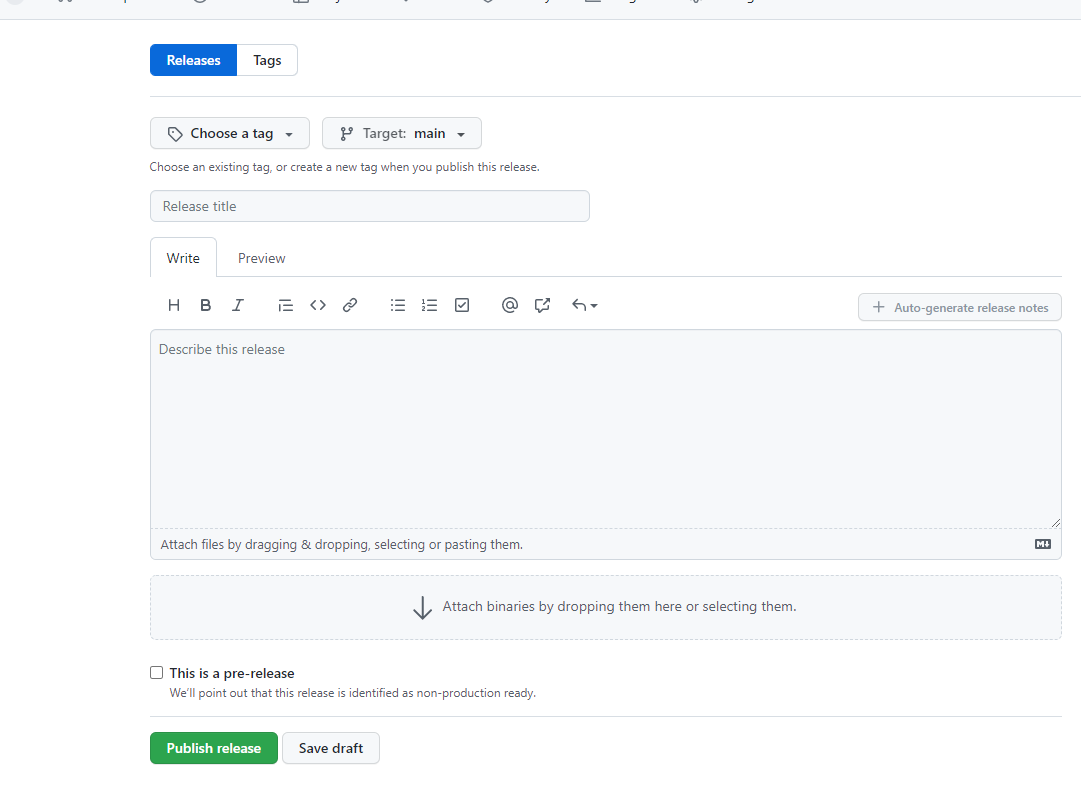


Рис. 13 Создаем новый релиз и вводим определенные данные

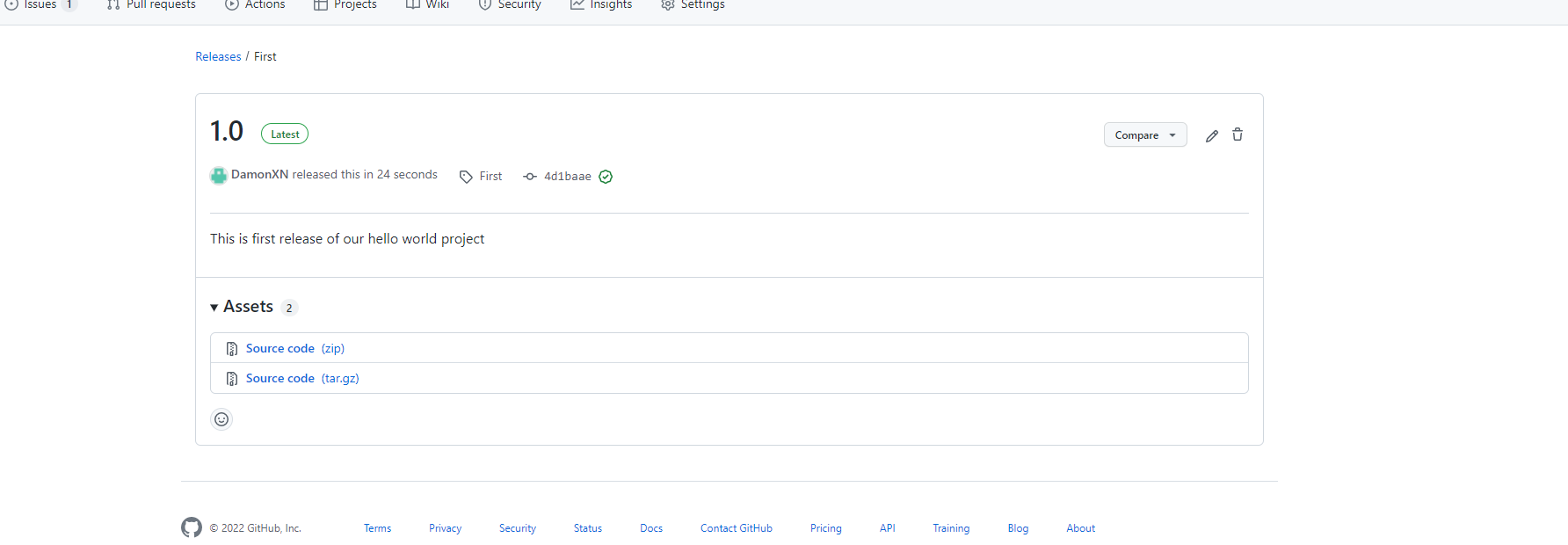


Рис. 14 Созданная страница релиза

**3. Контрольные вопросы**

1. Что такое система контроля версий?

Системы контроля версий (СКВ, VCS, Version Control Systems) позволяют разработчикам сохранять все изменения, внесённые в код. Поэтому в случае, описанном выше, они могут просто откатить код до рабочего состояния вместо того, чтобы тратить часы на поиски маленькой ошибки или ошибок, ломающих весь код.

1. Какие системы контроля версий бывают?

Существует три типа СКВ: локальная, централизованная и распределённая.

1. Что такое Git?

Git — распределённая система контроля версий, которая даёт возможность разработчикам отслеживать изменения в файлах и работать совместно с другими разработчиками.

1. Какие преимущества есть у Git?

**Бесплатный и open-source**. Это значит, что его можно бесплатно скачать и вносить любые изменения в исходный код;

**Небольшой и быстрый**. Он выполняет все операции локально, что увеличивает его скорость. Кроме того, Git локально сохраняет весь репозиторий в небольшой файл без потери качества данных;

**Резервное копирование**. Git эффективен в хранении бэкапов, поэтому известно мало случаев, когда кто-то терял данные при использовании Git;

**Простое ветвление**. В других СКВ создание веток— утомительная и трудоёмкая задача, так как весь код копируется в новую ветку. В Git управление ветками реализовано гораздо проще и эффективнее.

1. В чем отличия Git от GitHub?

Git — это инструмент, позволяющий реализовать распределённую систему контроля версий, а GitHub — это сервис для проектов, использующих Git.

**4.Выводе по работе**

Мы научились первоначальным навыкам использования систем контроля версий исходного кода программы, получили навыки организации разработки программного обеспечения. Создали в системе контроля версии репозиторий для нового проекта и выполнили все основные действия с исходным кодом, связанные с контролем версий.