МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Донецкий национальный технический университет»

Факультет ИСП

Кафедра ПИ им Л.П.Фельдмана

Лабораторная работа № 1

на тему: «Базовые принципы работы с системами контроля версий»

по дисциплине «Профессиональная практика программной инженерии»

Выполнил: студент группы ПИ-20а

\_\_\_\_\_\_\_\_\_\_\_\_\_\_Синяева Э.В.

(подпись) «\_\_\_» \_\_\_\_\_\_\_2021г.

Принял:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Незамова Л.В.

(подпись) «\_\_\_» \_\_\_\_\_\_\_2021г.

2021г.

Цель работы – получить практические навыки использования систем контроля версий.

Вариант 17. Эмулятор портативных устройств (GameBoy, PSP, Pokemon Mini).

1. Регистрация на GitHub (рис. 1). Аккаунт: Ludmilalala, ссылка: https://github.com/Ludmilalala/Portable\_Emulator/tree/main

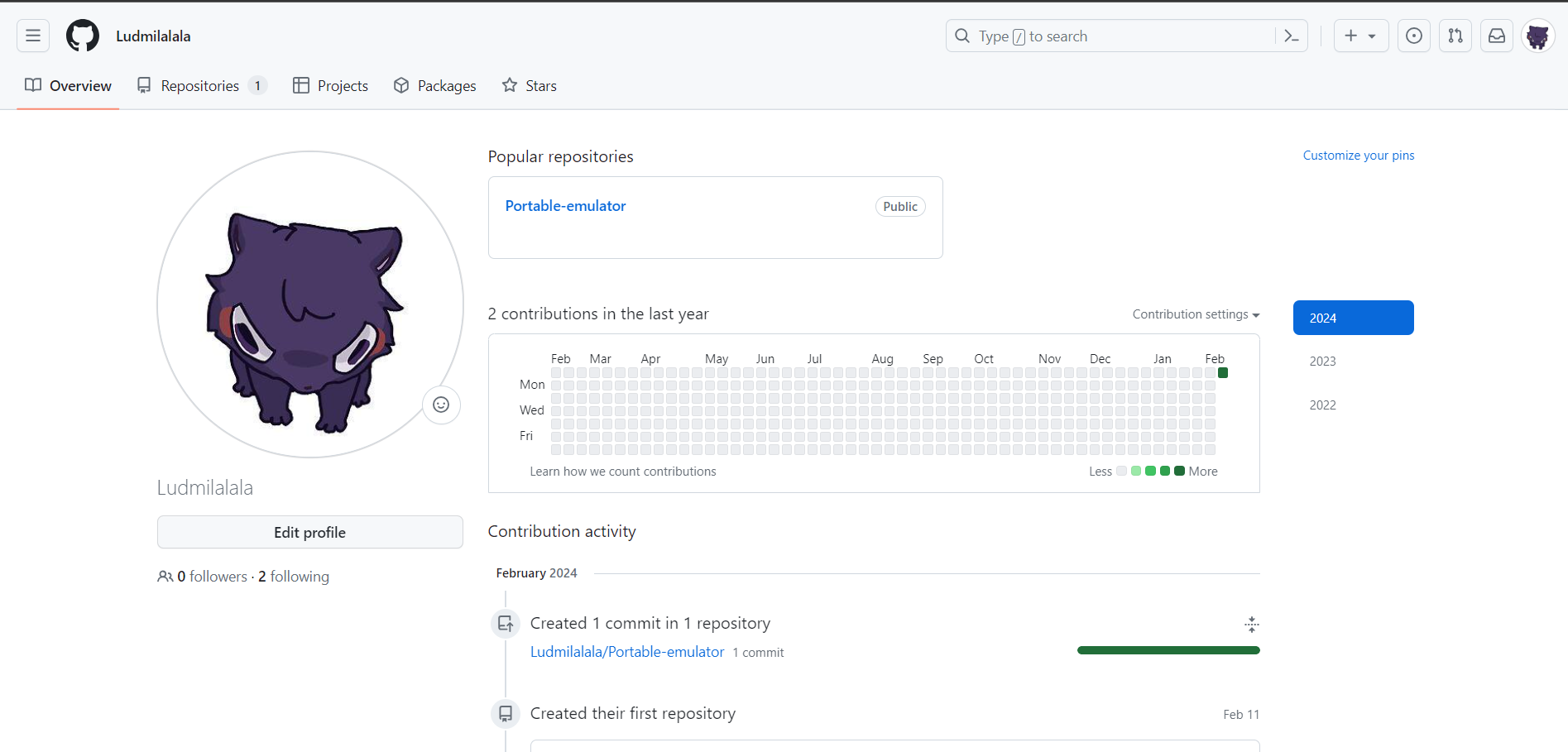


Рисунок 1 – Регистрация на GitHub

2. Создание репозитория и трёх папок в нём – программа, отчёты и команда (рис. 2).

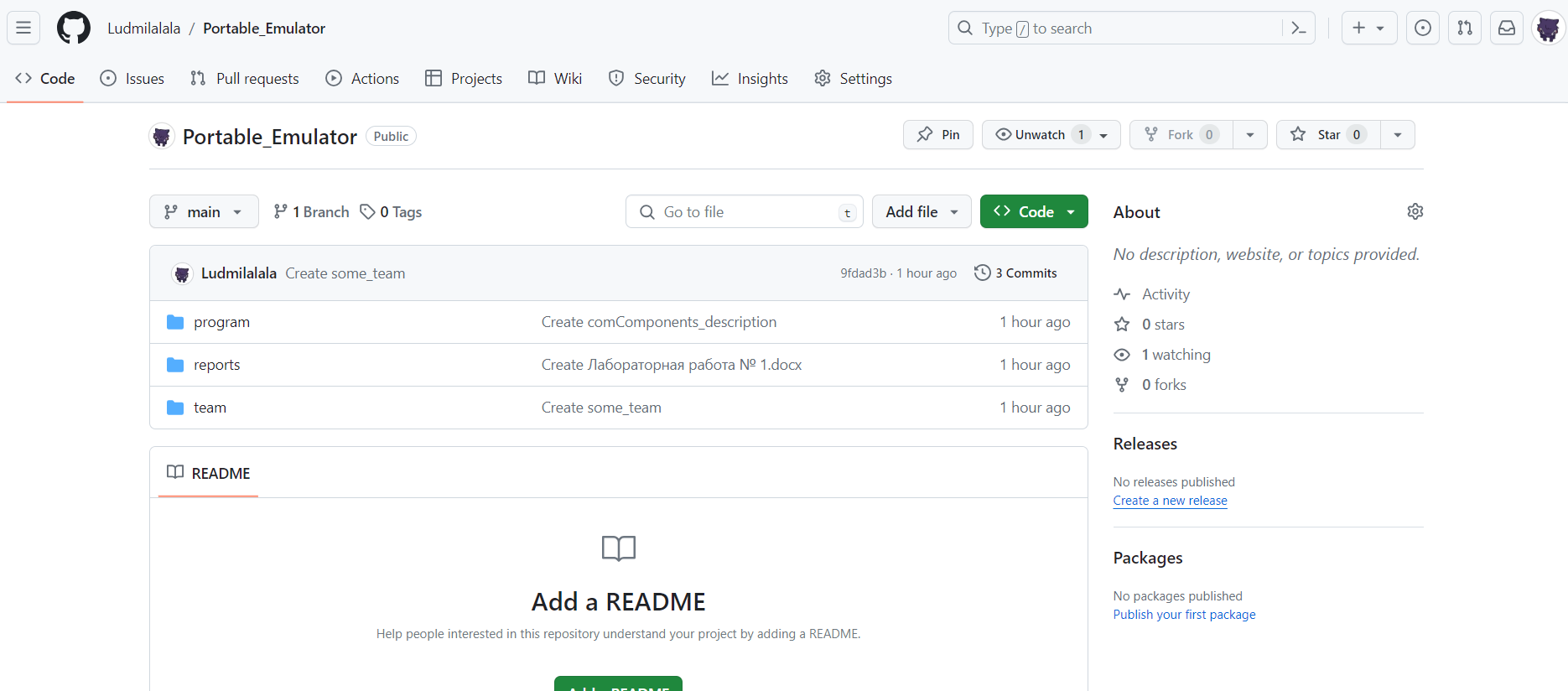


Рисунок 2 – Папки в репозитории

3. Краткое описание разрабатываемых компонентов/модулей моделируемого проекта:

Проект «Эмулятор портативных устройств (GameBoy, PSP, Pokemon Mini)» представляет собой комплексную систему, предназначенную для эмуляции различных портативных игровых устройств и пользователей, наслаждающихся классическими играми на современных устройствах. Разрабатываемые компоненты и модули этого проекта включают в себя:

1. Эмуляция аппаратной части: создание игровой среды
2. Графический интерфейс пользователя: разработка удобного и понятного пользовательского интерфейса.
3. Управление настройками: модуль для настройки параметров эмуляции, включения разрешения экрана, звуковых настроек.
4. Совместимость с играми.
5. Сохранение прогресса.
6. Интеграция с онлайн-сервисами: модуль для подключения к онлайн-сервисам.
7. Улучшение графиков и звука: возможность улучшить качество графиков и звука с помощью различных фильтров и эффектов.
8. Поддержка дополнительных функций: включение в эмулятор различных дополнительных функций, таких как чит-коды, мультиплеер и т.д.
9. Обновления и поддержка: обеспечение регулярных обновлений и техническая поддержка.
10. Безопасность и конфиденциальность: обеспечение мер безопасности для защиты данных пользователей.
11. Интеграция с облачными сервисами: возможность сохранения данных слота и прогресса в облачном хранилище.
12. Геймпад-эмуляция: поддержка подключения внешних геймпадов или контроллеров для более аутентичного игрового опыта.
13. Модуль управления эмуляцией: позволяет настраивать параметры, такие как скорость игры, сохранение/загрузка режима.
14. Поддержка мультиязычности.
15. Автоматическое обновление игр: интеграция функционала автоматического обновления игр.
16. Поддержка сохранения и загрузки процесса игры.
17. Поддержка различных форматов игровых образов: обеспечение совместимости с различными форматами игровых образов.
18. Возможность записи игрового процесса.
19. Поддержка мобильных устройств: оптимизация эмулятора для работы на мобильных устройствах.
20. Интеграция с настройками сетей: возможность делиться игровым прогрессом, достижениями и снимками экрана.
21. Расширенные настройки графики и звука.
22. Режимы эмуляции специальных устройств: включение возможности эмулирования специальных устройств, таких как геймпады.

На рисунке 3 представлено краткое описание модулей в папке “program”.

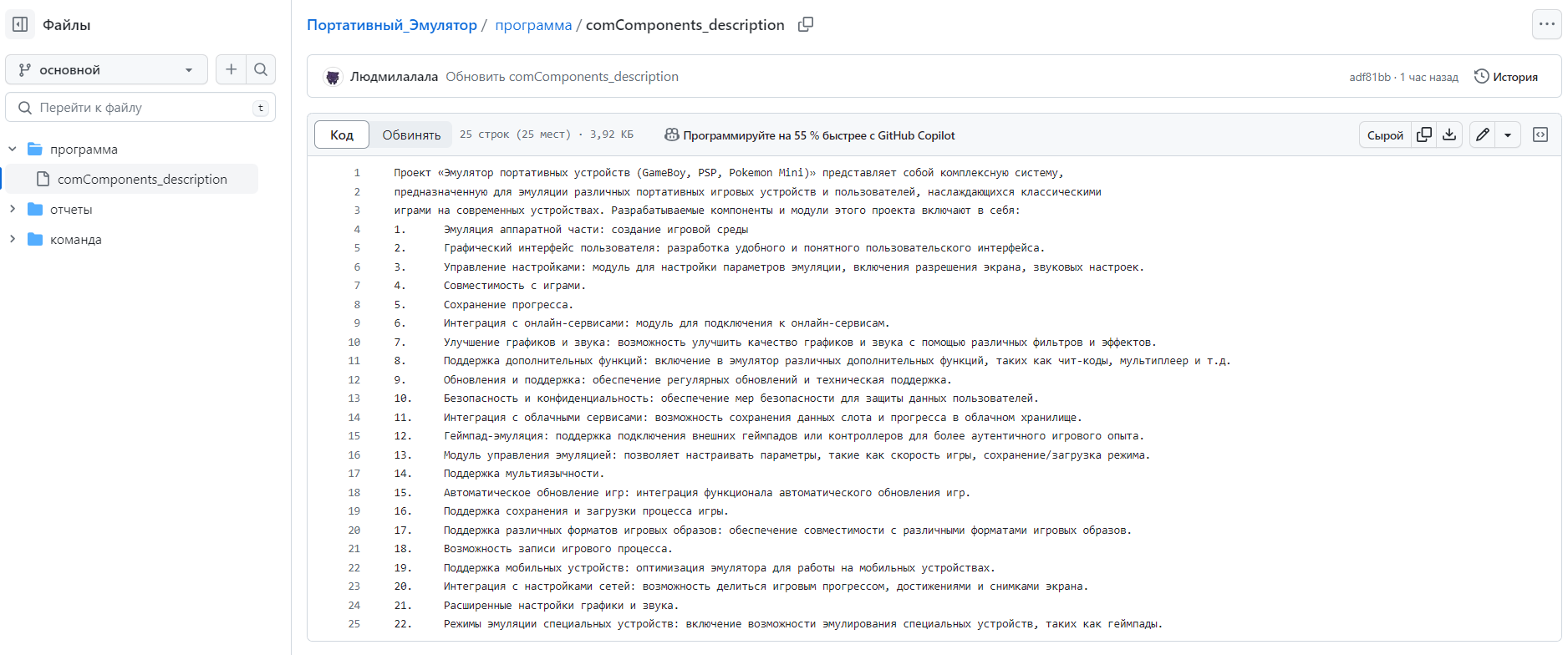


Рисунок 3 – Краткое описание модулей

4. Выполнение команды git log --pretty=format:\"%h %ad | %s%d [%an]\" --graph --date=short (рис. 4).

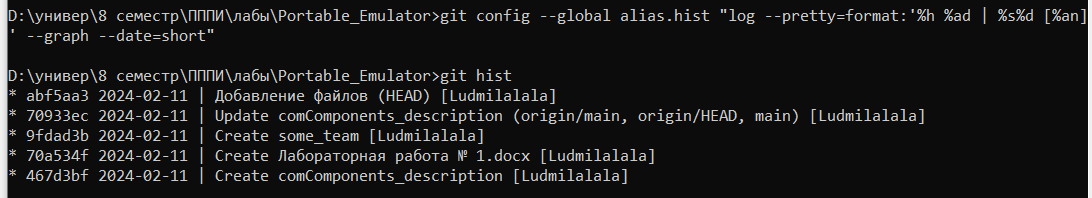


Рисунок 4 – Вывод команды git hist

команда git config --global alias.hist "log --pretty=format:'%h %ad | %s%d [%an]' --graph --date=short" создает псевдоним с именем hist для команды git log целью формирования результатов.

Конкретно эта команда задает следующее форматирование для вывода истории коммитов:

%h: сокращенный хеш коммита

%ad: дата коммита

%s: комментарий коммита

%d: теги коммита

%an: имя автора коммита

--graph: вывод графа коммитов

--date=short: даты заключения в кратком формате

5. Выполнение команды git diff (рис. 5).

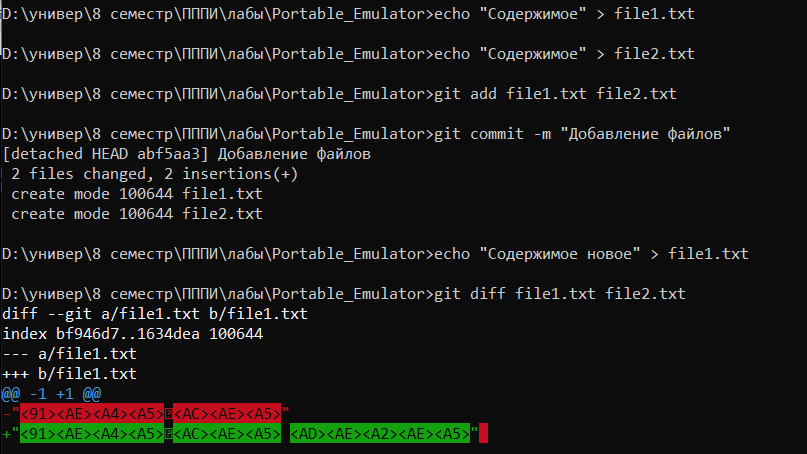


Рисунок 5 – Вывод команды git diff

**echo "Содержимое" > file1.txt**: Создает файл file1.txtи записывает в него текст "Содержимое". Это действие имитирует создание или изменение файла в вашем проекте.

**echo "Содержимое" > file2.txt**: Аналогично предыдущая группа создает файлы file2.txtи записывает в них текст «Содержимое».

**git add file1.txt file2.txt**: Добавляет файлы file1.txtи file2.txtв индекс Git, подготавливая их к коммиту. Это действие подготавливает изменения для исправления в репозитории.

**git commit -m "Добавление файлов"**: Создал коммит в репозитории Git, фиксирующий текущее состояние индекса. Сообщение коммита "Добавление файлов" является описанием изменений, внесенных в рамках данного коммита.

**echo "Содержимое новое" > file1.txt**: Изменяет критерий файла file1.txt, перезаписывая его текст "Содержимое новое". Это эмулирует изменение файла в вашем проекте после исправления изменений.

**git diff file1.txt file2.txt**: Сравнивает изменения между файлами file1.txtи файлами file2.txt, которые происходят в индексе и рабочей системе. Это позволяет увидеть, какие изменения были сделаны после последнего коммита и еще не были подготовлены к следующему коммиту.

6. Добавление файла отчёта по лабораторной работе (рис. 6).

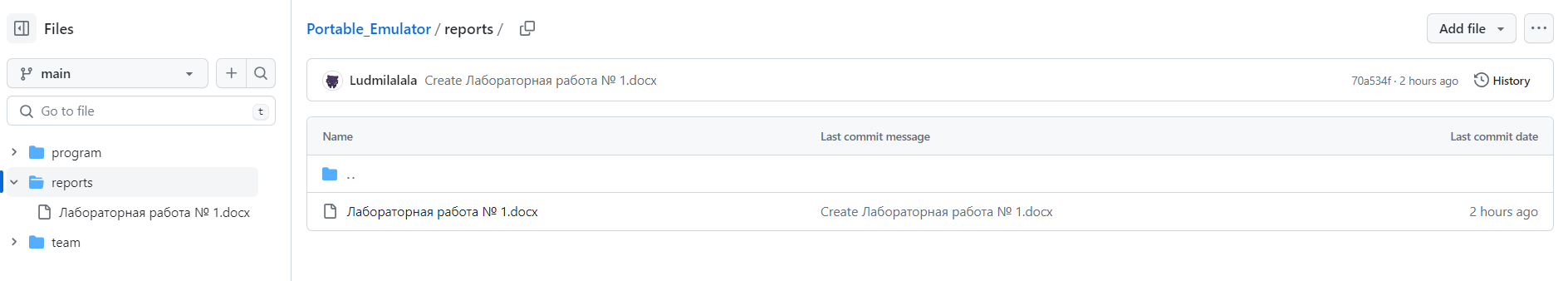


Рисунок 6 – Отчёт по лабораторной работе