МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Донецкий национальный технический университет»

Факультет Интеллектуальных систем и программирования

Кафедра "Программная инженерия" им. Л.П. Фельдмана

Лабораторная работа № 2

на тему: «Работа с ветками»

по курсу: «Профессиональная практика программной инженерии»

Проверил:

Незамова Л.В.

Выполнил:

ст. гр. ПИ-20Б

Шатохин И.И.

Донецк-2024

Задание:

Для каждого из описанных модулей добавить несколько файлов (по желанию), имитирующие разработку программного кода на любом выбранном языке программирования используя не менее 3х веток. Описать имеющиеся ветки,

Описать, какие файлы реализовывают какой конкретно модуль и в какой ветке находятся.

Влить все ветки в master.

Изменить данные в одной из веток так, чтобы вызвать конфликтную ситуацию (продемонстрировать ситуацию), решить конфликтную ситуацию (также продемонстрировать).

Добавить изменения в master.

Удалить все ветки кроме master.

Для работы клонируем репозиторий через git bash.

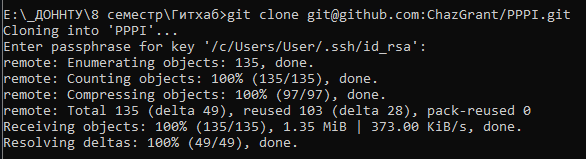


Рисунок 1 – Клонирование репозитория

1. Разделим модули по трём веткам и создадим ветки с помощью команды git branch имя\_ветки

Ветка 1: "Логика управления главным персонажем" – PlayerLogic

1. Модуль игрока – Player.cpp
2. Модуль оружия – Weapon.cpp

Ветка 2: "Взаимодействия с окружающим миром" – WorldInteractions

1. Модуль рендеринга и вывода сообщений – Interface.cpp
2. Модуль физического взаимодействия объектов в мире – World.cpp

Ветка 3: "Онлайн составляющая игры" – OnlineFeatures

1. Модуль вторжения в миры и модуль «Игрок против игрока» – PVPManagement.cpp

Ветка 4: "Сюжетная логика" – Storytelling

1. Модуль проигрыша заставок и сюжетных видео - StoryTriggers.py
2. Модуль взаимодействия с неигровыми персонажами и их историей - NPCManager.py

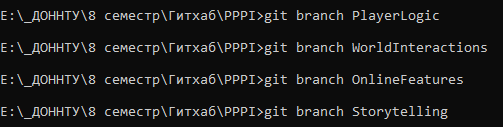
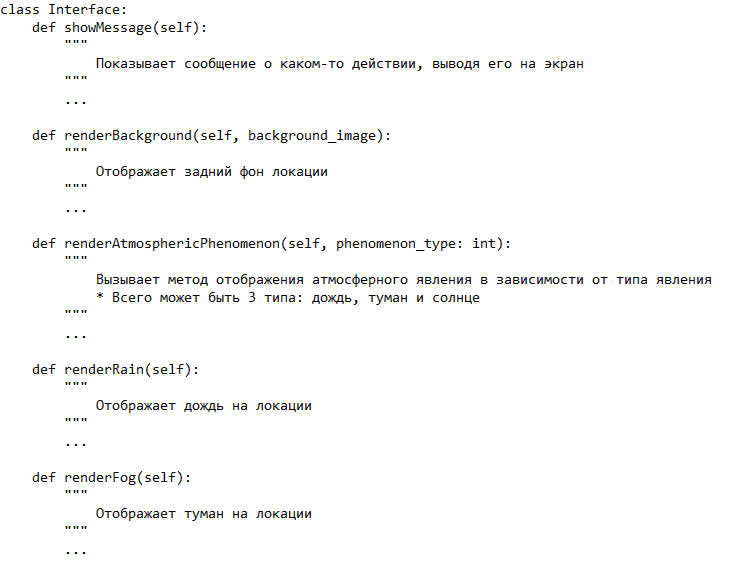


Рисунок 2 – Создание веток

2) Напишем псевдокод для каждого из модулей.

  
Рисунок 3 – Псевдокод файла Interface.py

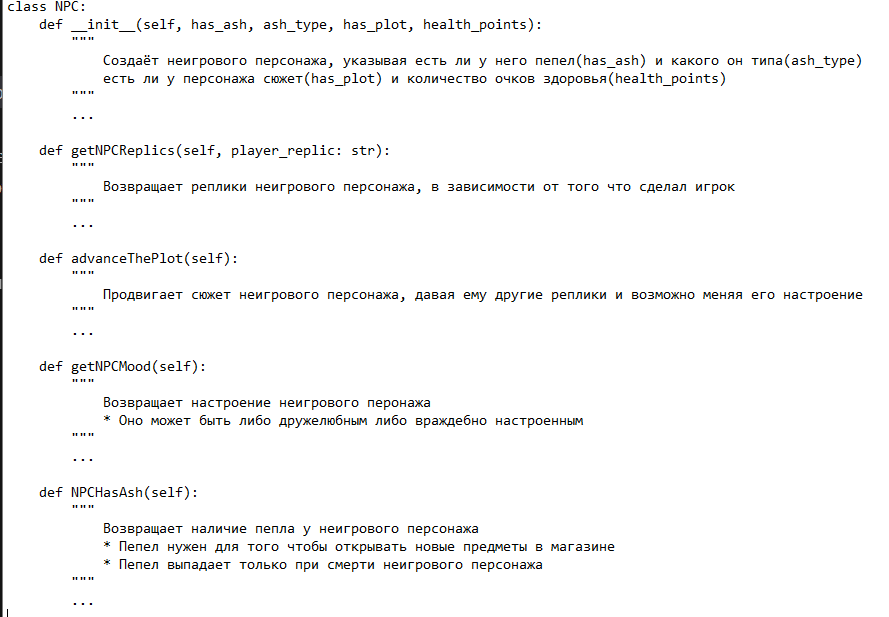


Рисунок 4 – Псевдокод файла NPCManager.py

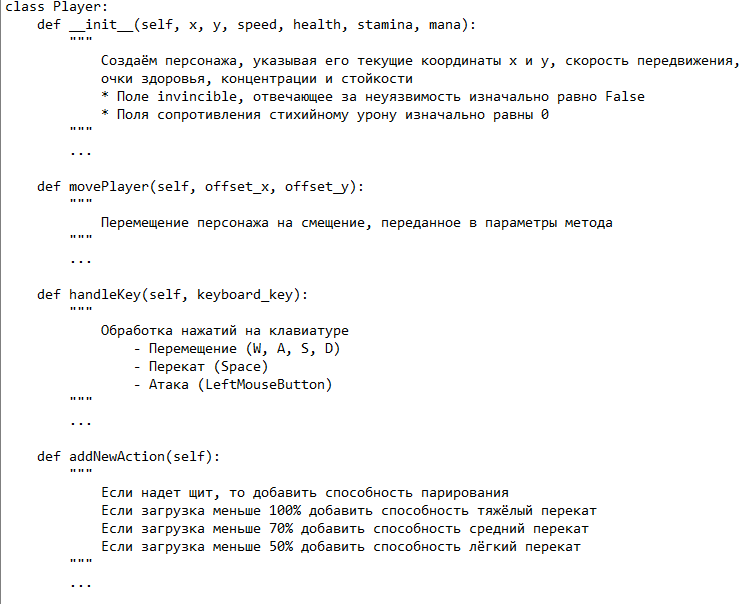
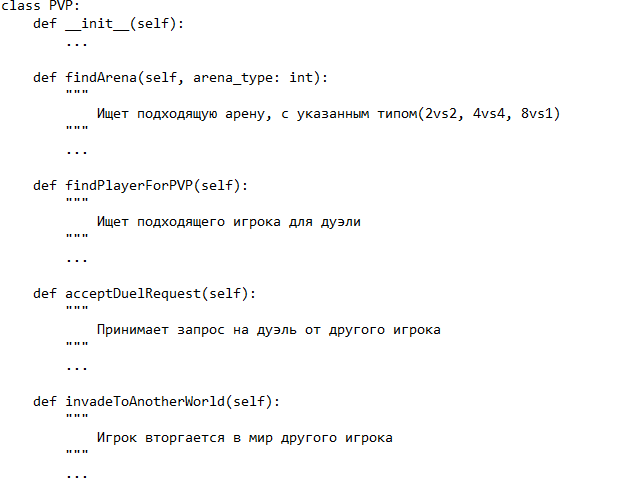


Рисунок 5 – Псевдокод файла Player.py

  
Рисунок 6 – Псевдокод файла PVPManagement.py

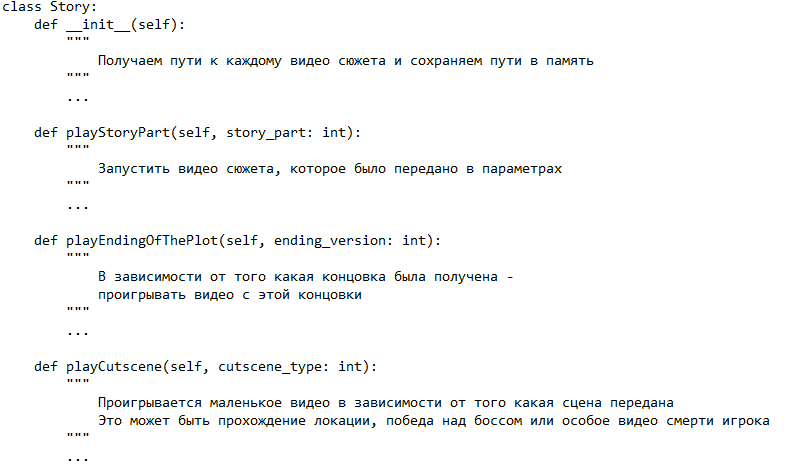


Рисунок 7 – Псевдокод файла StoryTriggers.py

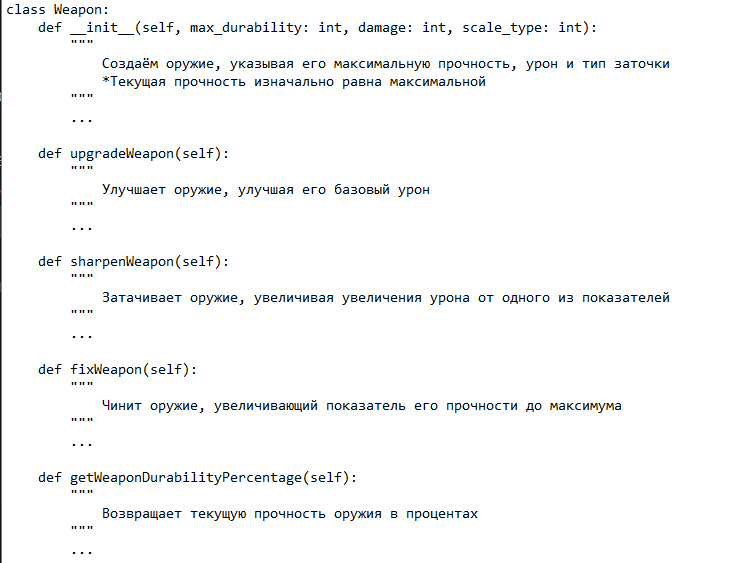


Рисунок 8 – Псевдокод файла Weapons.py

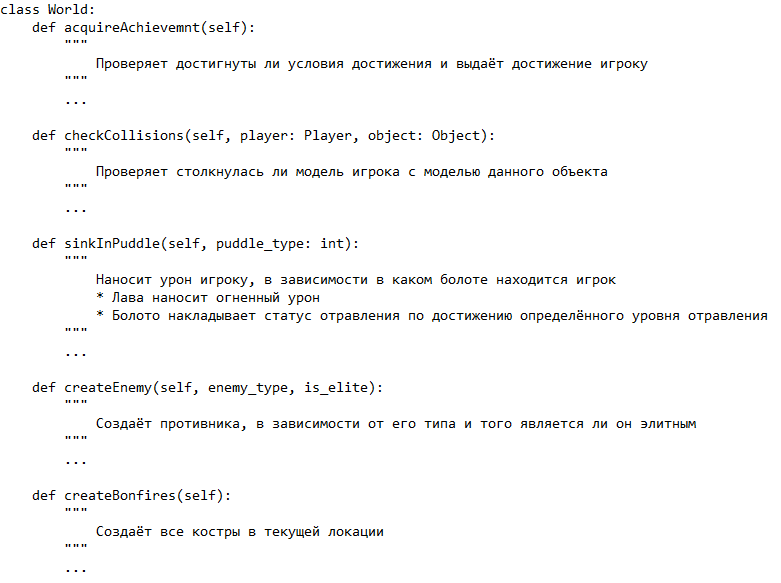


Рисунок 9 – Псевдокод файла World.py

3) Загружаем модули по разным веткам

Переключаемся на ветку PlayerLogic и добавляем туда файлы

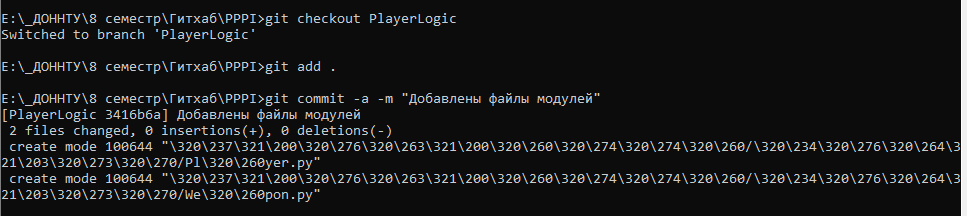


Рисунок 11 – Загрузка модулей в PlayerLogic

Переключаемся на ветку OnlineFeatures и добавляем туда файлы

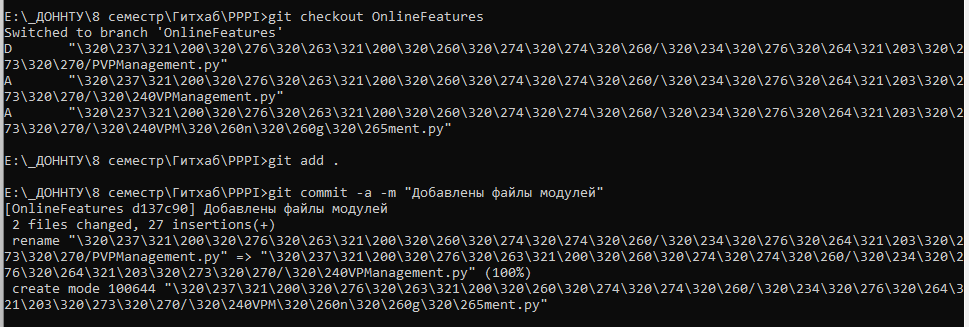


Рисунок 12 – Загрузка модулей в OnlineFeatures

Переключаемся на ветку Storytelling и добавляем туда файлы

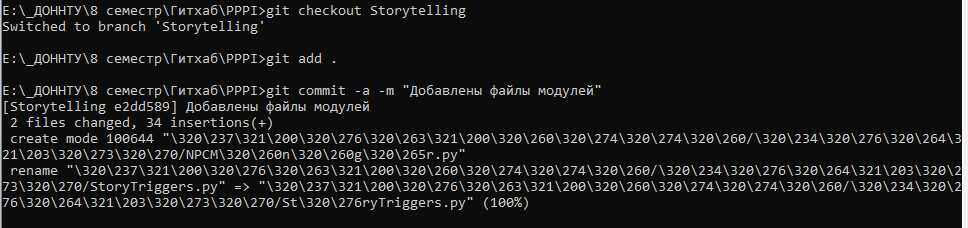
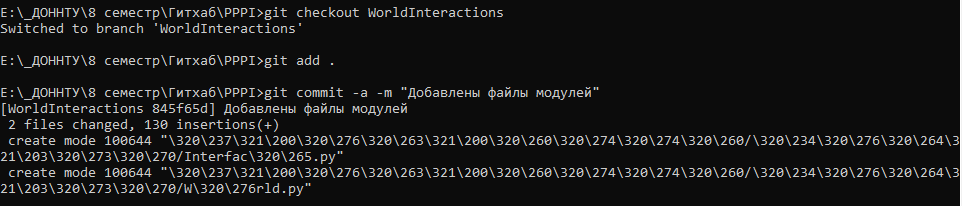


Рисунок 13 – Загрузка модулей в Storytelling

Переключаемся на ветку WorldInteractions и добавляем туда файлы



Проверяем лог ревизий для каждой ветки

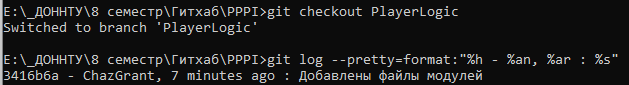


Рисунок 14 – Лог ревизий для PlayerLogic

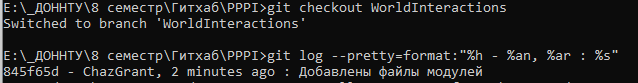


Рисунок 15 – Лог ревизий для WorldInteractions

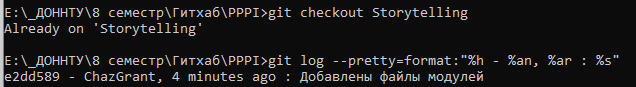


Рисунок 16 – Лог ревизий для Storytelling

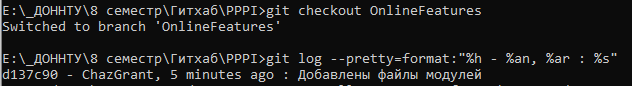


Рисунок 16 – Лог ревизий для OnlineFeatures

4) Чтобы получить конфликт слияния, добавим в ветке PlayerLogic файл с таким же названием как в ветке OnlineFeatures, но другим содержанием.

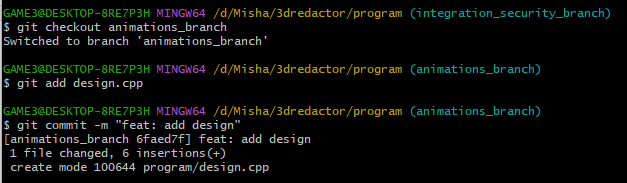


Рисунок 17 – Добавление файла для создания конфликта слияний

5) Переключаемся на ветку мастера и выполняем слияние веток.

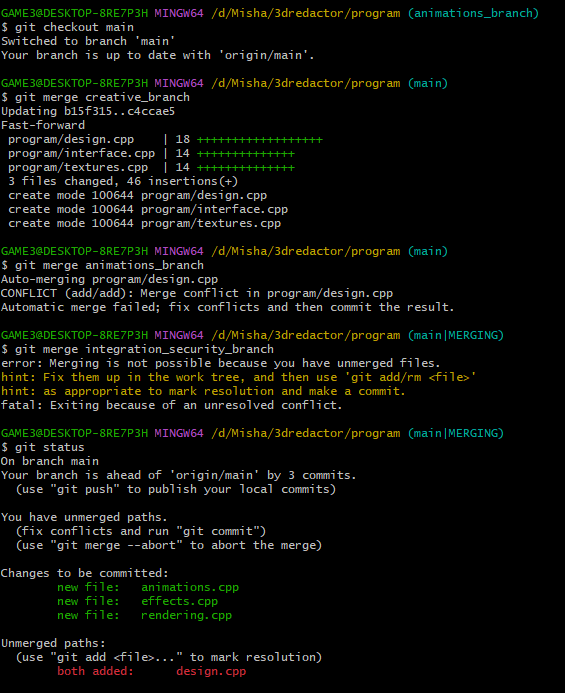


Рисунок 18 – Попытка слияния веток

Получаем конфликт слияния

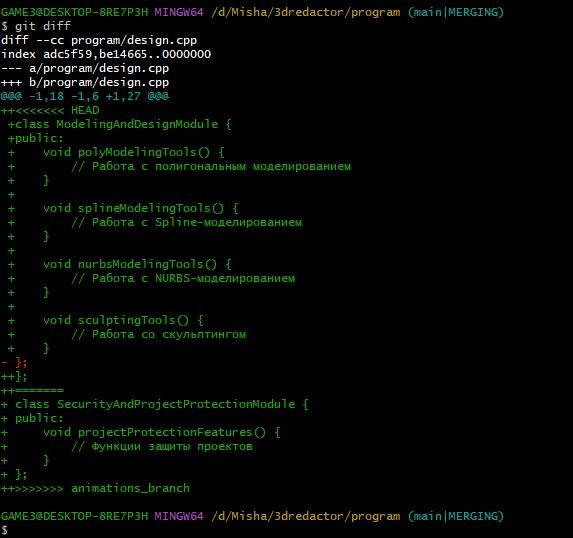


Рисунок 19 – Конфликт слияния

Чтобы разрешить конфликт достаточно отредактировать проблемный файл и закоммитить его. После чего снова попробовать слить ветки.

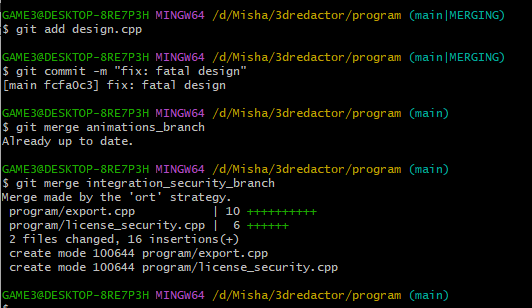


Рисунок 20 – Решение конфликта слияния

Лог ревизии

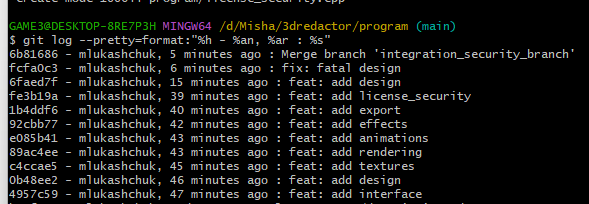


Рисунок 21 – Лог ревизии

6) Удаляем все ветки

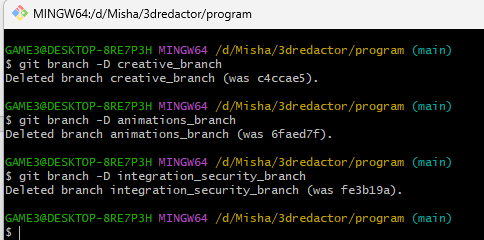


Рисунок 22 – Удаление всех веток кроме основной

В результате работы добавляются следующие файлы в репозиторий

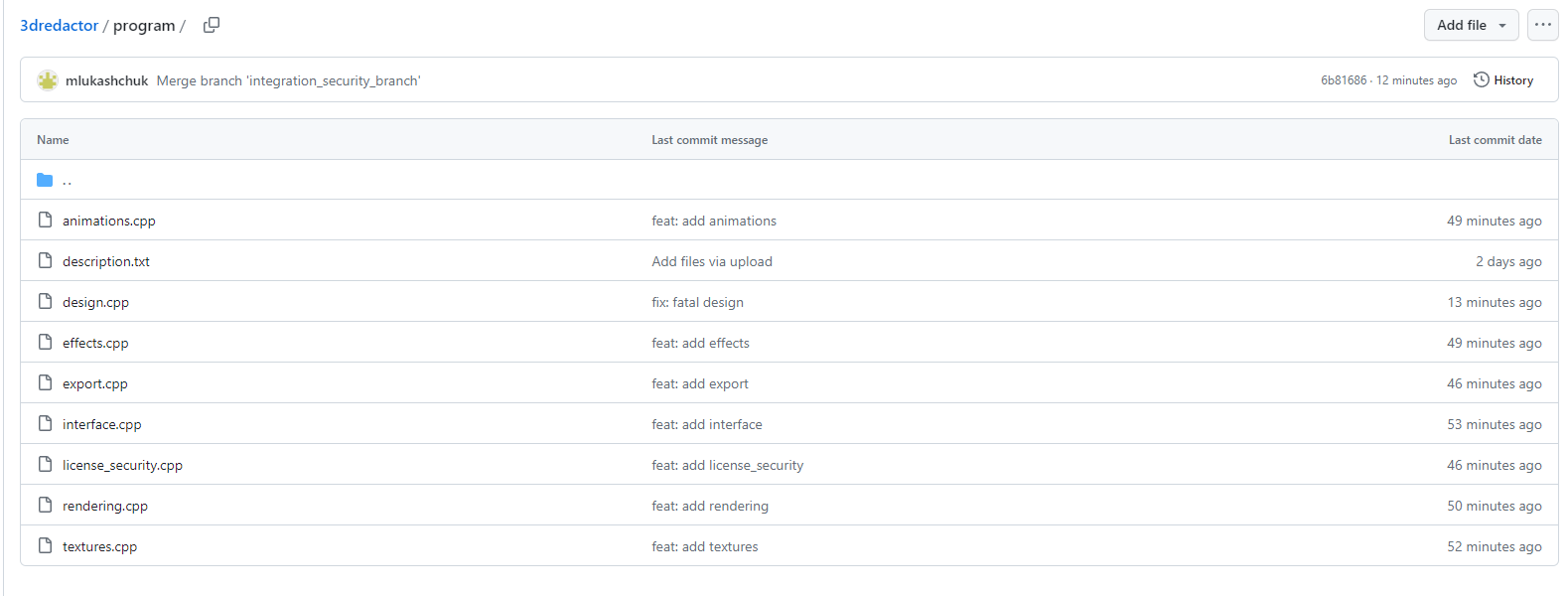


Рисунок 23 – Список добавленных файлов в репозитории

7) Загружаем отчёты в репозиторий

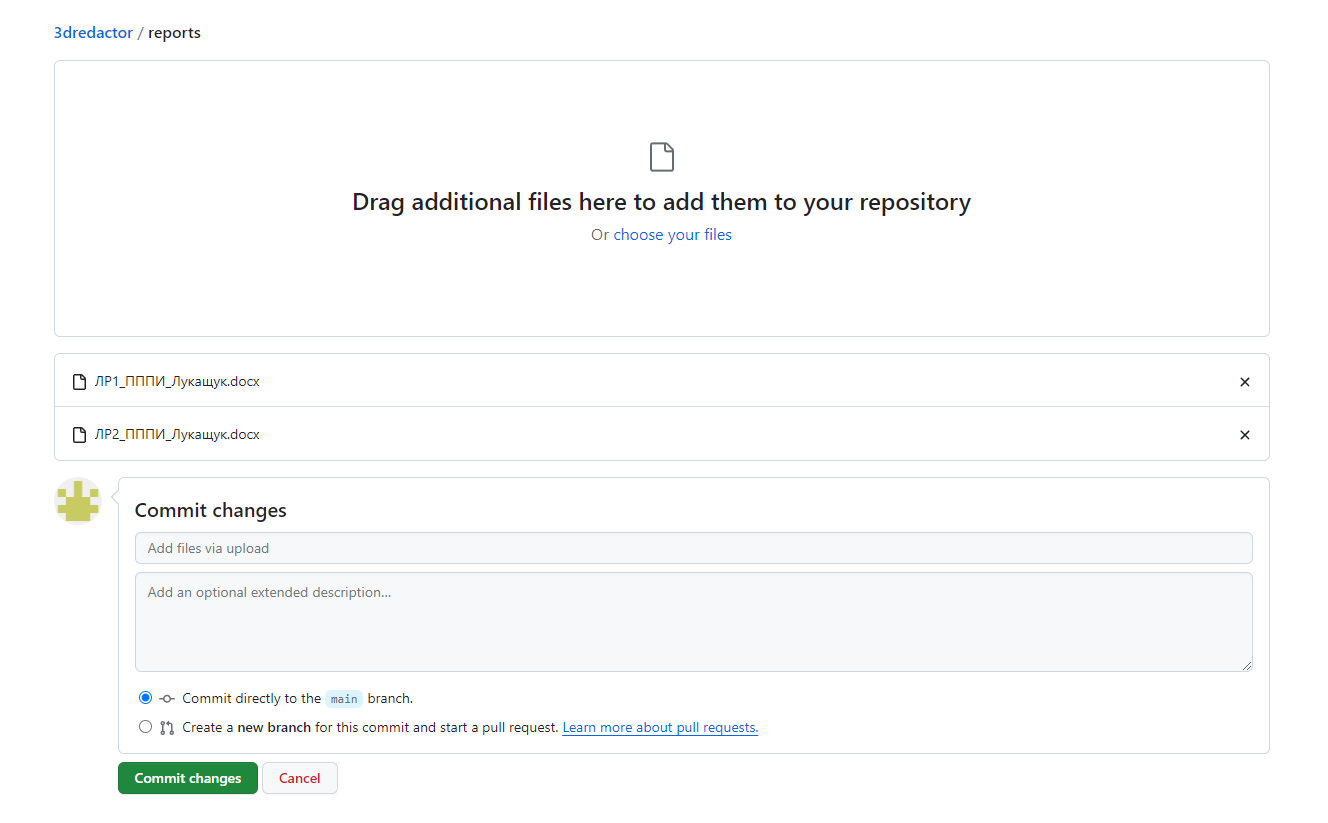


Рисунок 24 – Загрузка отчётов в репозиторий