|  |
| --- |
| **МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ** |
| **ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ** |
| **УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ** |
| **«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»** |
|  |
| Институт информационных технологий и управления в технических системах |
| (полное название института) |
|  |
| кафедра «Информационные системы» |
| (полное название кафедры) |

**ОТЧЁТ**

по лабораторной работе №6

на тему«Исследование распределенных системы контроля версий Git при коллективной разработке программных продуктов»

по дисциплине **«**Технологии создания программных продуктов»

Выполнил

студент ИИТУТС

группы ИС/б-18-2-о

Радыгина Екатерина

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | | | | | | |
|  | (должность, учёная степень преподавателя) | | | | | | | | |
|  |  | | | | | | | | |
|  | (ФИО преподавателя) | | | | | | | | |
|  | « |  | » |  |  |  | 20 | 20 | г. |
|  |  | | | | | | | | |
|  | (оценка) | | | | | | | | |
|  |  | | | | | | | | |
|  |  | | | | | | | | |

Севастополь 2020

# 6.1 Цель работы

Исследовать основные подходы к организации взаимодействия команды разработчиков с использованием распределенной системы контроля версий (DVCS). Приобрести практические навыки установки и настойки DVCS Git, организации ветвей разработки и осуществление слияния.

# 6.2 Постановка задачи

В качестве предметной области взят геймплей игры «1410b». Геймплей игры должен реализовывать возможность игрока передвигаться по локации и уничтожать врагов, защищая точки. Игрок может использовать различные инструменты, за использование которых с баланса снимаются монеты, которые игрок может подобрать, уничтожив врага. Враг подразделяется на различный типы, сейчас реализован Soldier. Враг постоянно движется к активной в данный момент точке (о том, какая точка активна, враг узнаёт, обращаясь в LevelManager), когда враг подходит на определённое расстояние к точке, то останавливается, и начинает стрельбу (Стрельба осуществляется появлением экземпляра класса Bullet перед врагом). Если игрок входит в зону видимости врага (Scale), враг начинает двигаться в сторону игрока и активирует стрельбу.

# 6.3 Ход работы

6.3.1 Для данного проекта была использована методология разработки SCRUM. Проект разрабатывался четырьмя разработчиками, для каждого была создана отдельная ветвь для работы с удалённым репозиторием (рисунок 6.1). Для работы с гитом была выбрана программа SmartGit. В качестве удалённого репозитория использовался MS Azure.

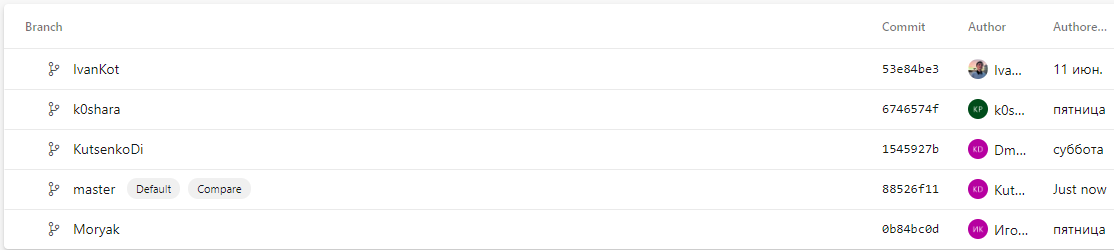


Рисунок 6.1 – Ветки

6.3.2 Создадим изменения в одном локальном репозитории, сохраним их в удаленном. Перед началом работы следует синхронизироваться с главной веткой. Для этого преходим на эту ветку и нажимаем pull (рисунок 6.2), после обратно на свою ветку, затем merge с главной веткой, в окне выбираем Create merge commit. Добавим в локальную ветку 3D модель одного из главных персонажей и закоммитим (рисунок 6.3). Затем перейдем в Azure и создадим pull request (рисунок 6.4) и подождем, пока его проверят и примут члены команды. После принятия изменения добавятся на главную ветку.

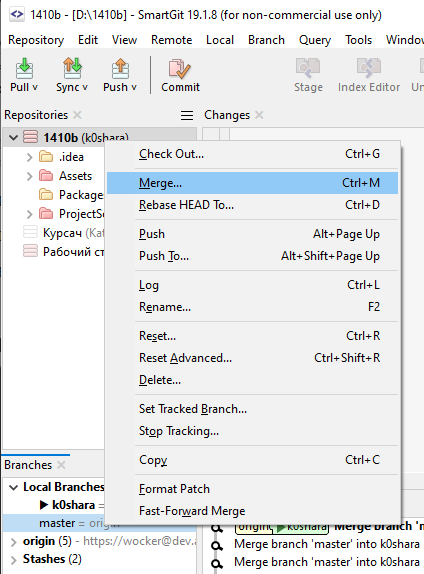
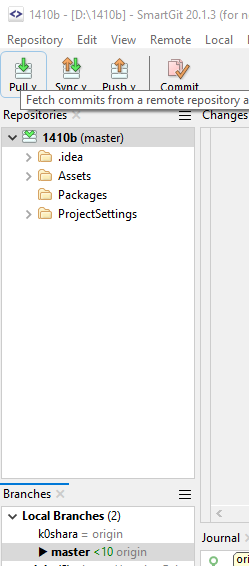


Рисунок 6.2 – Синхронизация веток

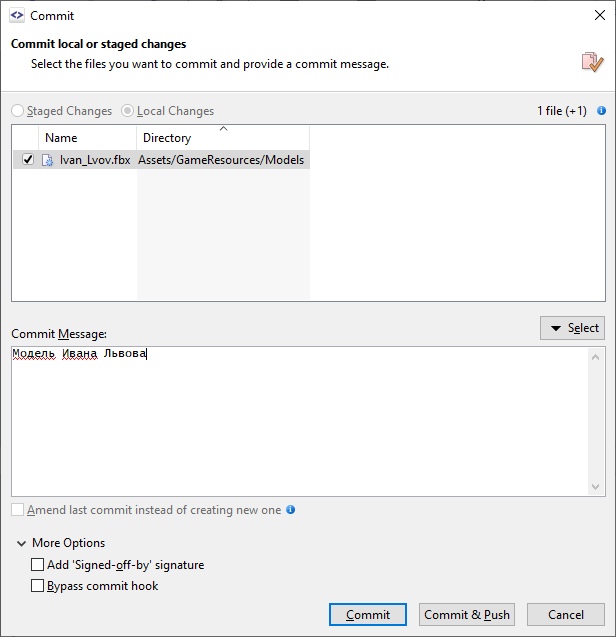


Рисунок 6.3 – Коммит на главную ветку

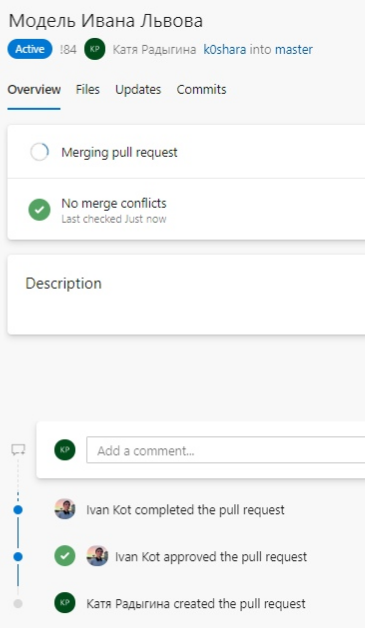
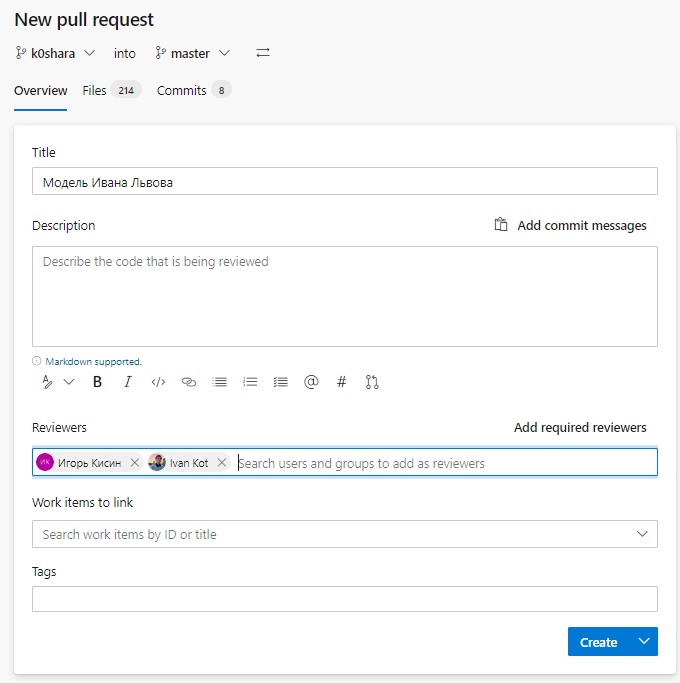


Рисунок 6.4 – Создание и принятие pull request

6.3.3 Внесем одновременно разные изменения в ветки и продемонстрируем их слияние. Отредактируем заранее созданный текстовый файл на главной ветке и локальной. После попытки добавления измененного файла из локальной в глобальную ветку, происходит конфликт (рисунок 6.5). Заходим в файл и решаем его (рисунок 6.6). Через stage принимаем изменения, rebase head to и добавляем файл в локальную ветку (рисунок 6.7).

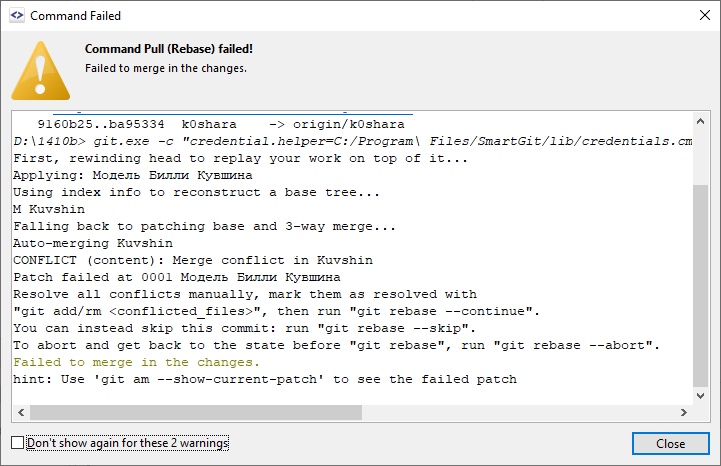
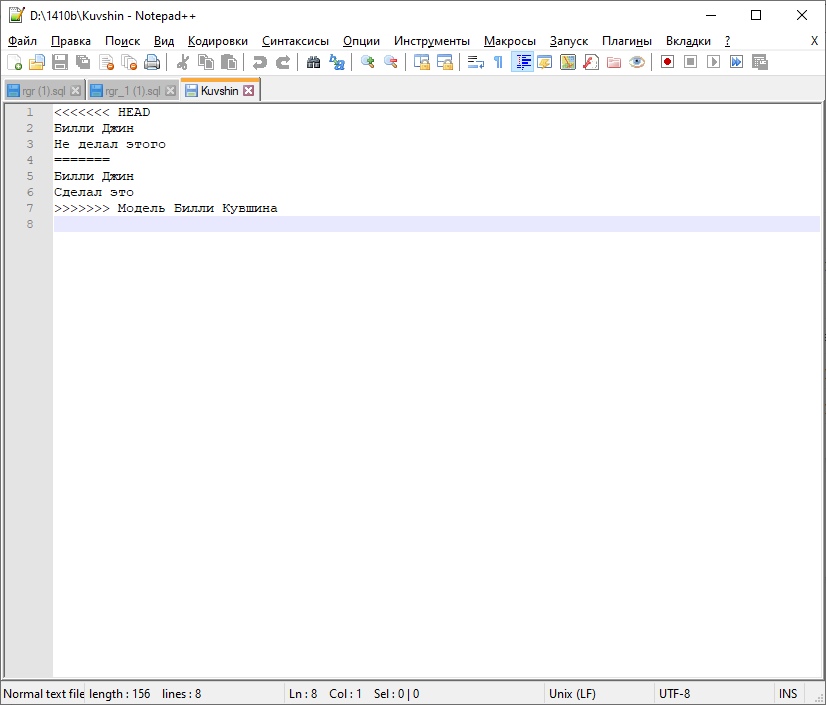


Рисунок 6.5 – Конфликт



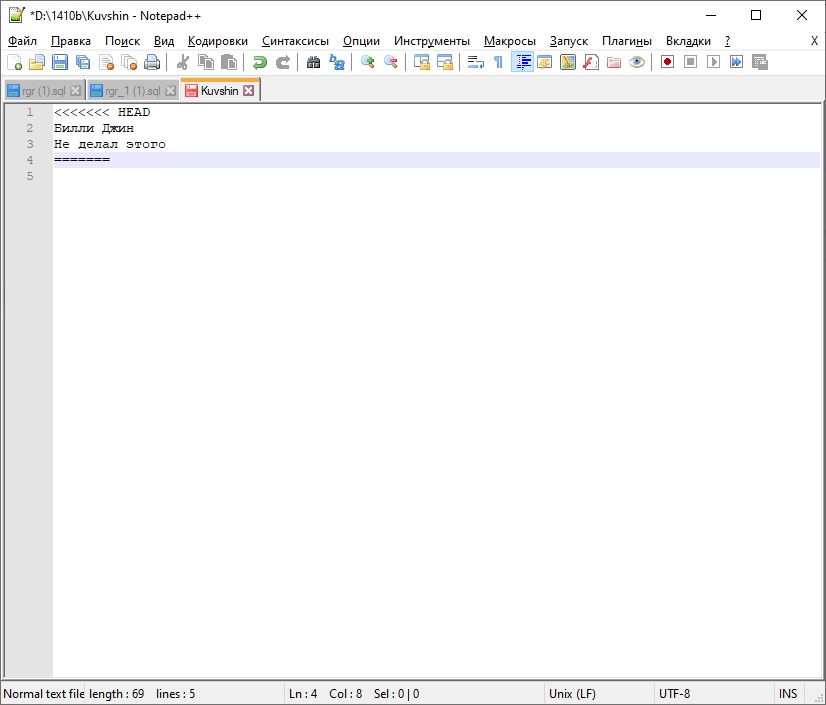
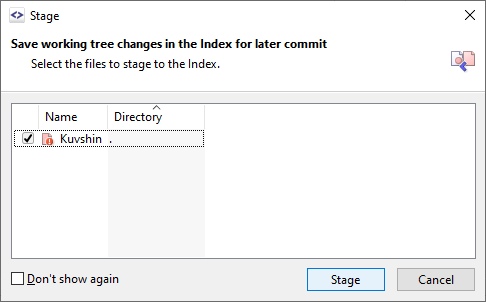


Рисунок 6.6 – Решение конфликта



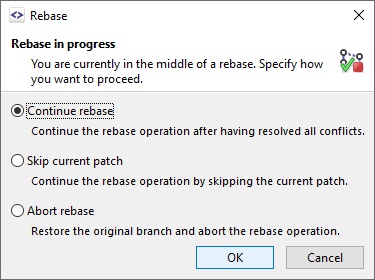


Рисунок 6.7 – Принятие решения конфликта

# Вывод

В ходе работы над проектом были исследованы основные подходы к организации взаимодействия команды разработчиков с использованием распределенной системы контроля версий (DVCS). Приобретены практические навыки установки и настойки DVCS Git, организации ветвей разработки и осуществление слияния.