Министерство образования и науки РФ

Институт информационных технологий и управления в технических системах

Севастопольский государственный университет

Лабораторная работа №6

по дисциплине: «ТСПП»

«**Исследование распределенных систем контроля версий Mercurial при**

**коллективной разработке программных продуктов**»

Выполнил:

ст.гр. ИСб/22о

Воронин И.Ю.

Проверил:

Дрозин А.Ю.

г. Севастополь

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 Mercurial, организации ветвей разработки и осуществление слияния.

1. Вариант задания

2.1. Разработать модель командной работы согласно варианту, полученному у преподавателя.

2.2. Создать необходимое количество репозиториев, разработать соглашение по предназначению репозиториев.

2.3. Создать изменения в одном локальном репозитории, сохранить их в удаленном.

2.4. Получить набор изменений из удаленного репозитория в репозиторий отличный от описанного в п.2.3, внести дополнительные изменения и сохранить их в удаленном репозитории.

2.5. Внести одновременно разные изменения в локальные репозитории сохранить их все в удаленном, продемонстрировать процесс слияния.

2.6 Внести изменения в разных локальных репозиториях в одинаковые файлы, в одинаковых строках. Продемонстрировать процесс слияния при наличии конфликтов.

2.7. Продемонстрировать создание именованных веток в локальном репозитории.

2.8 Дать краткое описание команд распределенной системы контроля версий, используемых при выполнении работы.

2.9. Проанализировать результаты работы, сделать выводы.

3. Ход работы

#hg init

#hg add InsuranceTerminal

#hg add

#hg status

Все файлы данного проекта будут отобразиться с атрибутом А (рис.3.1).

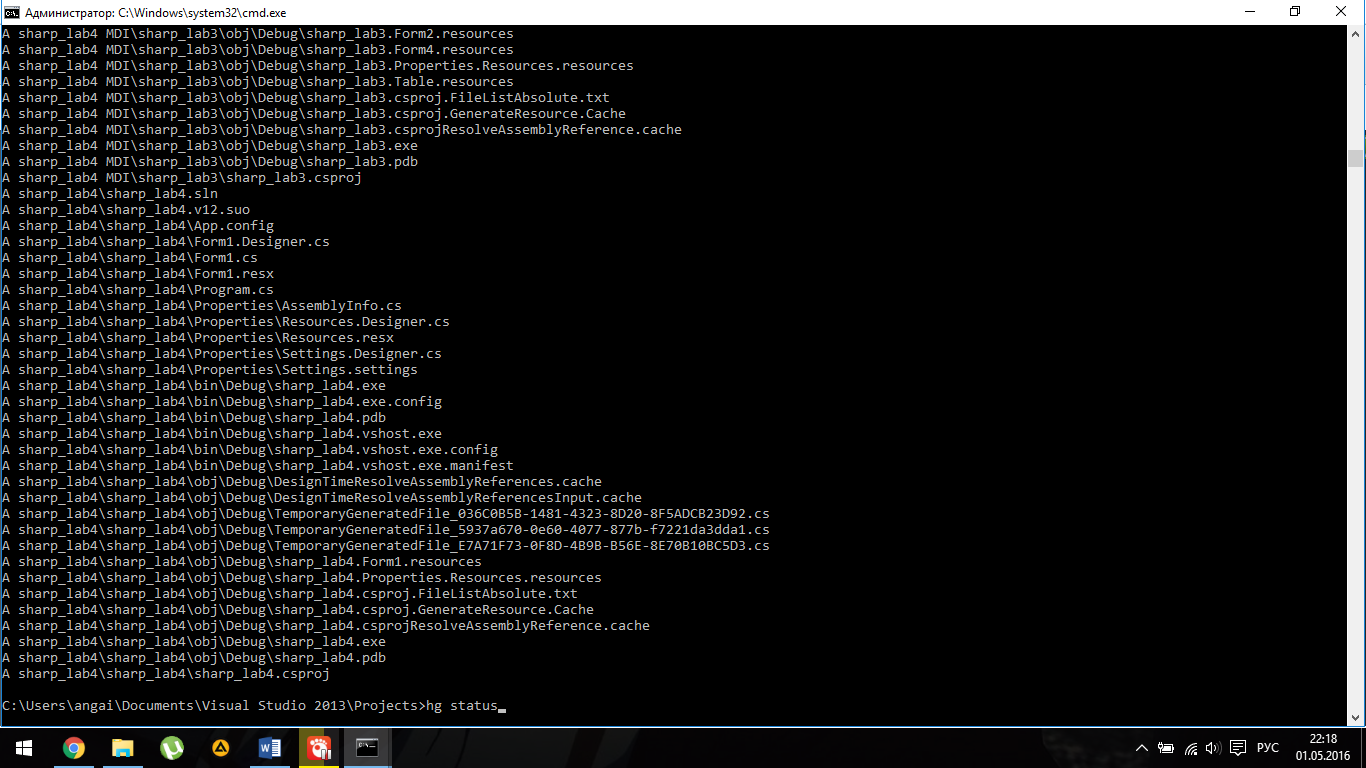


Рисунок 3.1 – Статус файлов

#hg config –edit

Записываем username

#hg commit

Добавляем комментарий и подтверждаем создание ревизии. (рис.3.2)

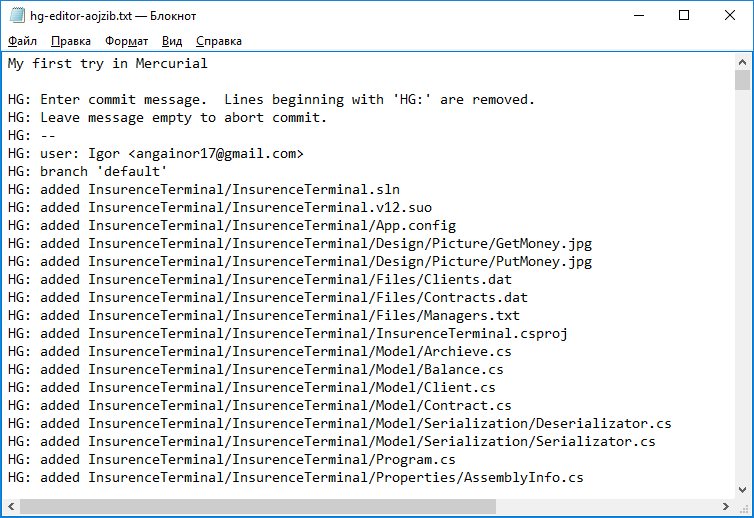


Рисунок 3.2 – Сохранение изменений.

Все изменения сохранены в локальном репозитории. (рис.3.3)

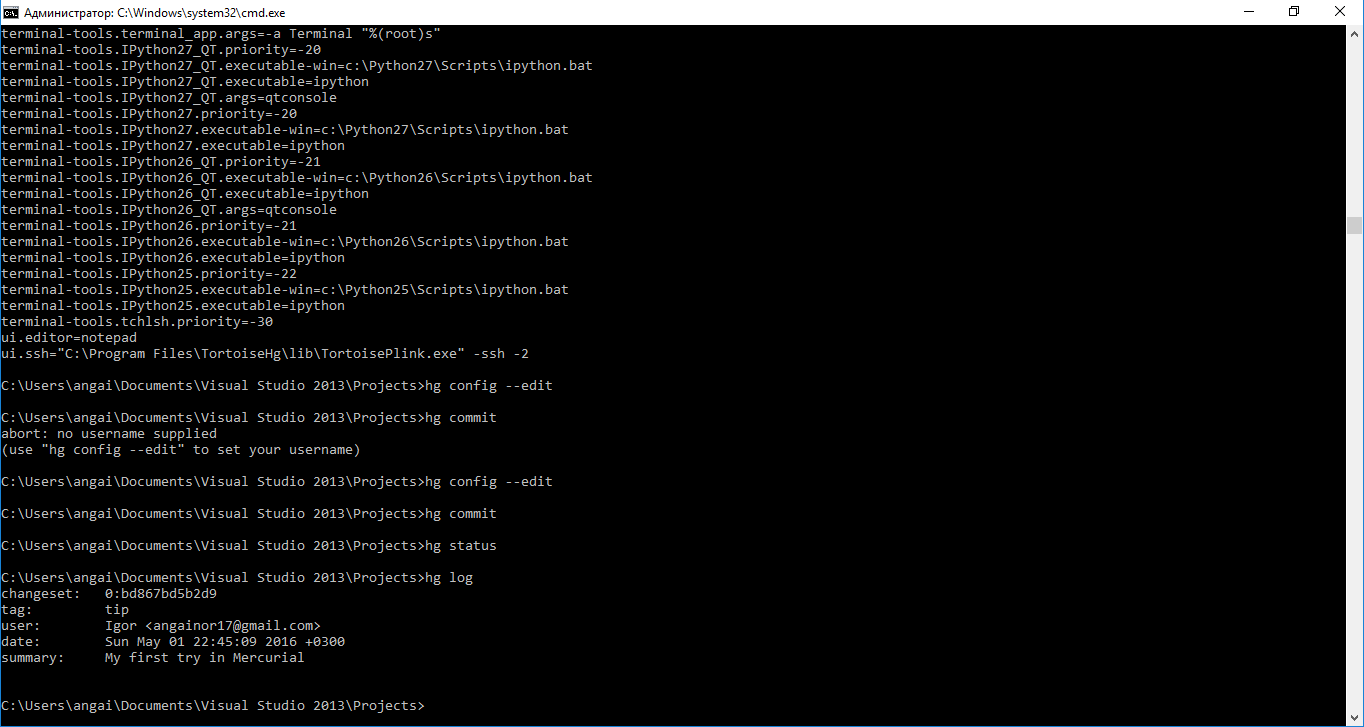


Рисунок 3.3 – Данные о ревизии.

Можем просмотреть наши данные на сайте, адрес которого указан на рисунке 3.4.

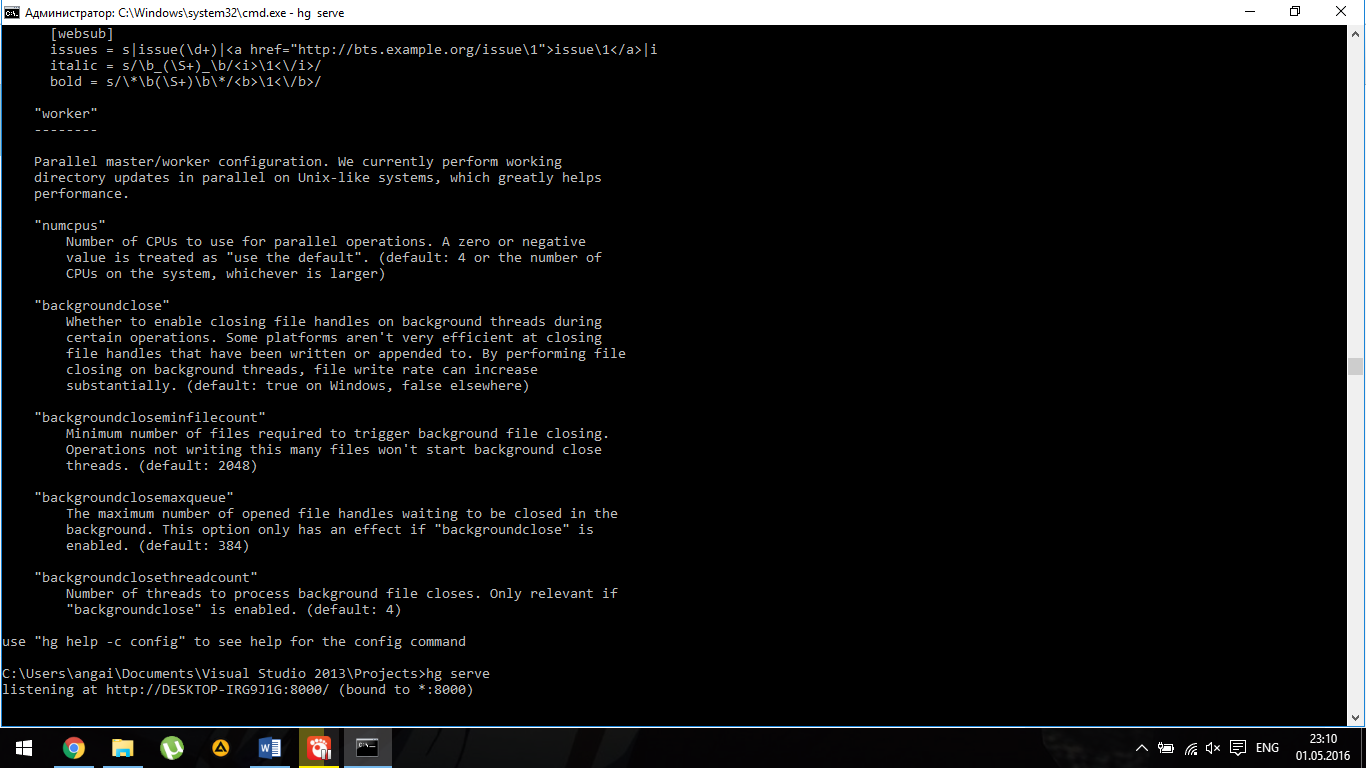


Рисунок 3.4 – Отображение данных

Передадим изменения в репозиторий, находящий в папке «1». (рис.3.5)

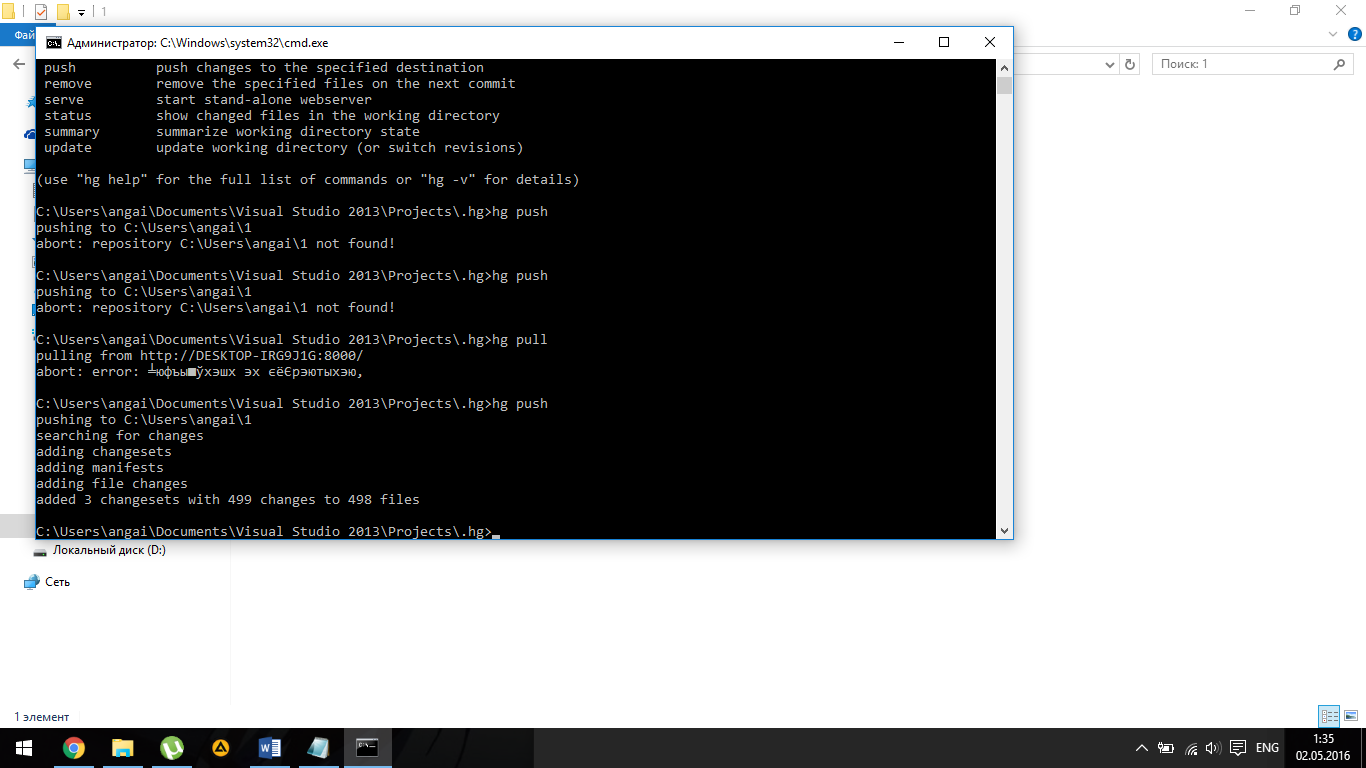


Рисунок 3.5 – Использование push.

Изменим исходный репозиторий и после снова его обновим. (рис.3.6)

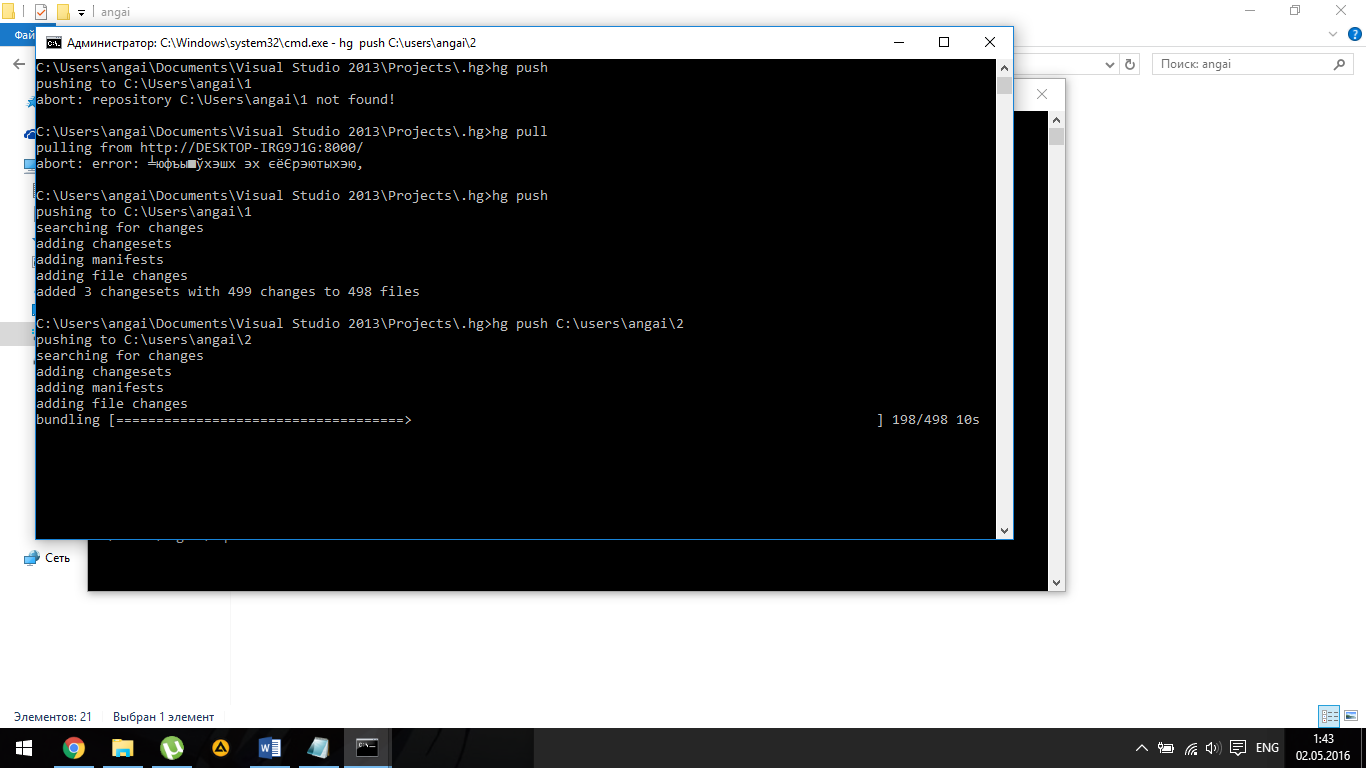


Рисунок 3.6 – Обновление репозитория.

Получим исходный репозиторий, который будет отличаться, от данного состояния рабочего проекта. (рис.3.7)

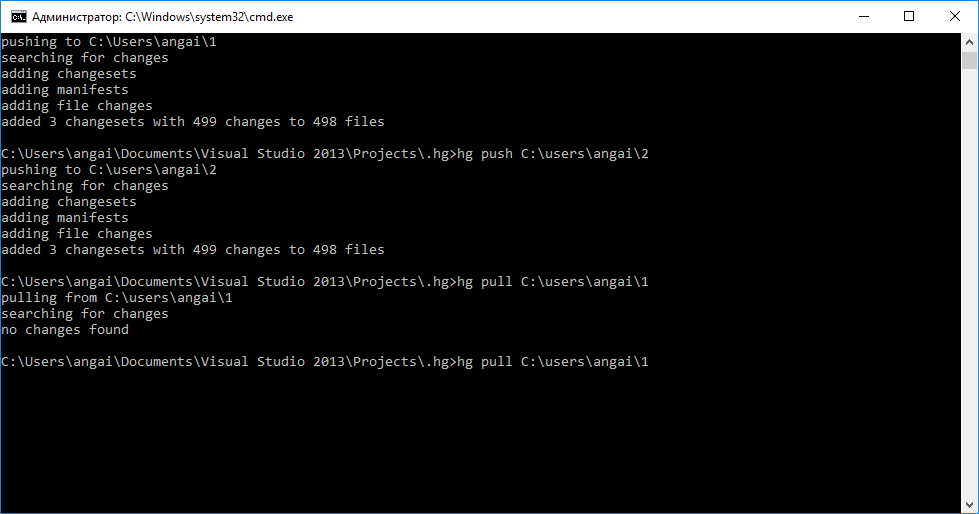


Рисунок 3.7 – Согласование репозиториев.

Загрузим в старый репозиторий новые изменения. (рис.3.8)

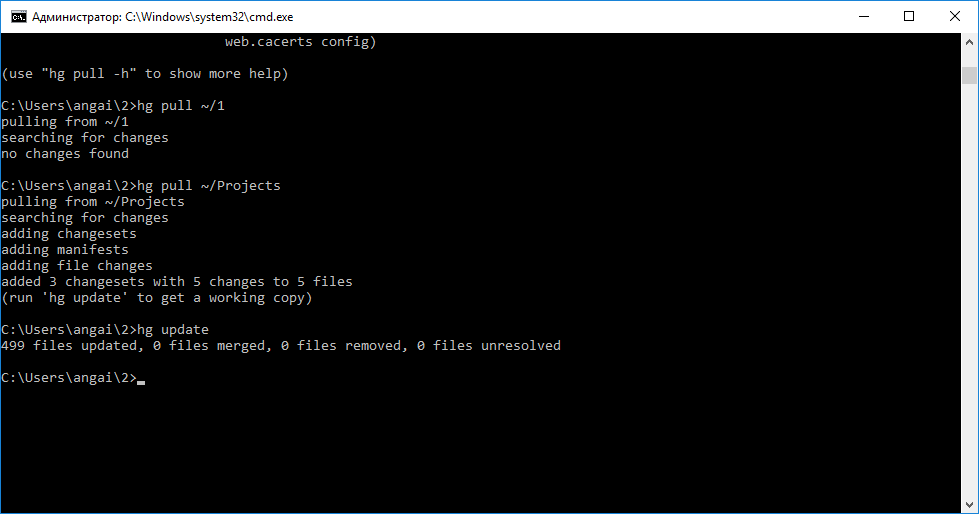


Рисунок 3.8. – Операция pull.

Создадим новую ветвь в одном из репозиториев. (рис.3.9)

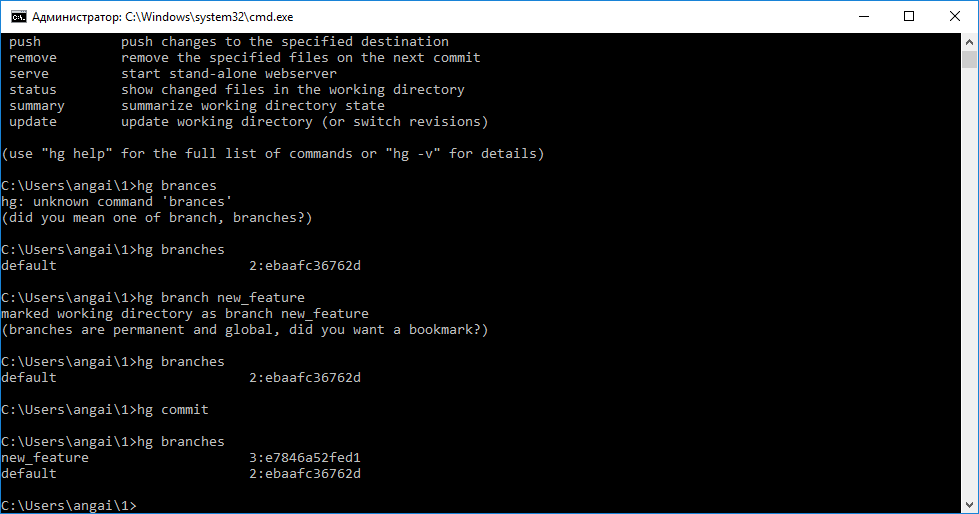


Рисунок 3.9 – Создание ветвей.

ВЫВОДЫ

В данной лабораторной работе были изучены распределённые системы контроля версий и особенности работы с ними на примере Mercurial. Были использованы ранее написанные проекты, созданы различные репозитории, ревизии и ветвления.