

## **Лабораторная работа №25**

### **Работа с системой контроля версий**

#### **1 Цель работы**

Научиться применять систему контроля версий git в процессе разработки программного обеспечения.

#### **2 Литература**

2.1 Git – Book. – URL: <https://git-scm.com/book/ru/v2>. – Режим доступа: свободный. – Текст : электронный.

#### **3 Подготовка к работе**

3.1 Повторить теоретический материал (см. п.2).

3.2 Изучить описание лабораторной работы.

#### **4 Основное оборудование**

4.1 Персональный компьютер.

#### **5 Задание**

Для локального хранения репозитория создать на ПК папку, в которой будут храниться локальные копии репозитория:

C:/Temp/isppXX/repos/

Вместо XX – номер группы

##### **5.1 Создание удаленного репозитория**

В браузере перейти на страницу <https://gitgen.softsol.local/> и зарегистрироваться:

- имя пользователя: isppXXNN (вместо NN – номер рабочего места. Например: ispp2101, ispp3514)

- Email: isppXXNN@arcotel.ru

- Пароль: тот же, что и логин (isppXXNN)

Нажать на «+» и «Новый репозиторий» для создания репозитория

Указать настройки репозитория:

- имя репозитория: LabWork25

- видимость: отметить сделать репозиторий приватным

- описание: указать ЛР№25 по МДК.01.01, группу, фамилии авторов

- инициализировать репозиторий: да

Нажать «Создать репозиторий». На открывшейся странице отобразится содержимое основной ветви проекта (main) и текст, находящийся в файле README.md.

##### **5.2 Оформление readme.md**

Чтобы открыть файл README.md на редактирование, нужно нажать на него.

На открывшейся странице нажать кнопку карандаш («Редактировать файл») и заменить на следующий текст содержимое открывшегося на редактирование файла, используя инструменты форматирования Git (текст, выделенный желтым, заменить на актуальные данные о том, что используется на лабораторных по МДК.01.01):

**# Лабораторная работа по МДК.01.01**

**## Авторы**

**Фамилия Имя, группа**

**## Используемое для разработки ПО**

**1. язык программирования F#**

**2. .NetFramework 4.0**

**3. среда разработки Microsoft Visual Studio Community 2010**

**## Требования к запуску приложения на стороне клиента:**

**\* .NetFramework 4.0**

При нажатии на кнопку «Предпросмотр» можно увидеть, как будет выглядеть файл. Если все корректно, то следует нажать на кнопку «Сохранить правки».

Если некорректно (нет заголовков разных уровней, нумерованного и маркированного списков), то нажать на «Редактировать файл» и внести изменения.

При нажатии на «Сохранить правки» отображается текст комментария к изменению. При необходимости его можно изменить перед отправкой изменения в репозиторий.

После нажатия на «История» и выбор коммита отображается сравнение старой и новой версий файла.

### **5.3 Работа с репозиторием (в браузере)**

#### **5.3.1 Создать файл в браузере**

Перейти на страницу ветки main, нажать на «Добавить файл – Новый файл» в верхней части страницы.

В открывшемся окне указать имя файла Program.cs в поле ввода и содержимое файла в многострочном поле ввода:

```
class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("Hello, world!");
        Console.ReadLine();
    }
}
```

В разделе «Фиксация изменений» указать описание «Добавление файла Program.cs, выводящего hello world».

#### **5.3.2 Загрузить файл с компьютера.**

Перейти на страницу ветки main, нажать на «Добавить файл – Загрузить файл» в верхней части страницы. Для загрузки выбрать текстовый документ (\*.txt) или документ с листингом (\*.cs).

### 5.3.3 Удалить созданный и загруженный файлы.

Перейти к редактированию файла, нажать пиктограмму корзины («Удалить файл»)

### 5.3.4 Просмотреть выполненные изменения.

Перейти на страницу ветки main, нажать на пиктограмму с часами («n коммитов»)

Изучить выполненные коммиты:

- для просмотра определенного выбрать хэш: SHA1
- для просмотра графа нажать «Граф коммитов»

## 5.4 Работа с ветками (в браузере)

5.4.1 Создать ветвь dev на основе основной ветви.

5.4.2 Добавить в ветви dev новый файл и отредактировать файл readme.md

5.4.3 Создать новый запрос на слияние:

- базовая ветка main
- взять из dev

5.4.4 Принять запрос на слияние

5.4.5 Изучить изменения, появившиеся в ветке, и граф коммитов.

**Задания 5.5-5.8 выполнять, используя git bash. Все команды, результаты и действия сохранить в виде скриншотов в документе LabWork25.docx.**

## 5.5 Создание рабочей копии на своем ПК путем клонирования

5.5.1 Для локального хранения репозитория создать на ПК папку, в которой будут храниться локальные копии репозитория:

C:/Temp/isppXX/repos/

Вместо XX – номер группы

5.5.2 Создать локальную копию репозитория в папке для репозитория  
C:/Temp/isppXX/repos/Repository1

Для открытия Git Bash в контекстном меню выбрать Git Bash Here

## 5.6 Внесение изменений в рабочую копию и загрузка изменений в новую ветку

Создать в папке с рабочей копией проект на C#. Содержимое скрытой папки .git не заменять (в ней настройки локальной копии репозитория).

Загрузить файлы из локальной копии в ветку на сайте.

## 5.7 Оформление .gitignore

Добавить в папку с рабочей копией файл .gitignore.

Запретить в нем загрузку в репозиторий:

- папок bin и obj
- файлов pdf и docx

Отправить изменения на сервер

## 5.8 Получение и отправка изменений

5.8.1 Получение изменений из удаленного репозитория

Изменить в удаленном репозитории содержимое файла README.md, заменив «Лабораторная работа №25 по МДК.01.01» на «Лабораторная работа №25» и сохранить изменения.

Получить изменения из удаленного репозитория

5.8.2 Отправка изменений в удаленный репозитория

Внести изменение в файл приложения в рабочей копии (например в файле с кодом добавить комментарии, переименовать переменные).

Сохранить изменения в удаленном репозитории.

5.8.3 Просмотреть граф коммитов

## **6 Порядок выполнения работы**

6.1 Выполнить все задания из п.3.

6.2 Ответить на контрольные вопросы.

## **7 Содержание отчета**

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

## **8 Контрольные вопросы**

6.1 Что такое «репозиторий»?

6.2 Что указывается в файле readme.md?

6.3 Что указывается в файле .gitignore?

6.4 Какое программное обеспечение может применяться для управления git-репозиторием?

6.5 Где может располагаться репозиторий?