**Лабораторная работа № 20**

Тема: Изучение систем контроля версий GIT, GITHUB

Ход работы

Задание 1

* Создать свой профиль на github.
* Установить на свой ПК GitHub Desktop и создать в нем новый репозиторий.
* Разместить в нем свой КУРСОВОЙ ПРОЕКТ (программу-исходники, файл SETUP, ПЗ, ГЧ)
* Сделать интеграцию вашего репозитория с GitHub.
* Ссылку на свой профиль в github на Курсовой проект сбросить в google class в ЛР работу.

Ссылка на проект https://github.com/TheFezzz/cp

Контрольные вопросы

1. Назначение систем CVS

Системы контроля версий (CVS — Concurrent Versions System) предназначены для управления изменениями в документах, коде и других файлах. Основные задачи CVS включают:

• Отслеживание изменений: Позволяют пользователям видеть, кто и когда вносил изменения.

• Сохранение истории: Хранят все версии файлов, что позволяет вернуться к предыдущим состояниям.

• Совместная работа: Обеспечивают возможность нескольких пользователей работать над одним проектом одновременно, минимизируя конфликты.

• Управление версиями: Позволяют создавать, объединять и удалять версии файлов.

2. Виды CVS

Существует несколько видов систем контроля версий:

• Централизованные системы (например, CVS, Subversion):

• Все изменения хранятся на центральном сервере.

• Пользователи получают доступ к последней версии проекта и могут вносить изменения.

• Проблема: при сбое сервера доступ к данным может быть потерян.

• Распределенные системы (например, Git, Mercurial):

• Каждое рабочее пространство содержит полную копию репозитория.

• Пользователи могут работать офлайн и синхронизировать изменения позже.

• Преимущества: повышенная надежность и гибкость.

3. Особенности подхода к работе с данными в Git

Git — это распределенная система контроля версий, обладающая следующими особенностями:

• Локальные репозитории: Каждый разработчик имеет полную копию репозитория на своем компьютере, что позволяет работать без подключения к интернету.

• Снимки (snapshots): Git хранит изменения как снимки состояния файлов, а не как различия между ними. Это упрощает управление версиями.

• Гибкое ветвление и слияние: Создание и объединение веток происходит быстро и эффективно, что позволяет легко экспериментировать с новыми функциями.

• Эффективность: Git оптимизирован для быстрого выполнения операций, таких как коммиты, переключение между ветками и слияние.

Преимущества Git:

• Высокая скорость работы.

• Возможность работы в оффлайн-режиме.

• Легкость в управлении ветками и слияниями.

• Расширенные возможности для совместной работы.

4. Процесс установки и настройки Git на ПК

Установка Git

1. Скачивание:

• Перейдите на официальный сайт Git (https://git-scm.com/).

• Выберите версию для вашей операционной системы (Windows, macOS, Linux).

2. Установка:

• Для Windows: запустите загруженный установщик и следуйте инструкциям. Рекомендуется оставить все настройки по умолчанию.