Лабораторная работа № 20

Изучение систем контроля версий GIT, GITHUB, GitLAB и др.

**Теоретическая часть:**

СИСТЕМЫ КОНТРОЛЯ ВЕРСИЙ GIT

1. Опишите назначение систем CVS.

Назначение системы контроля версий (CVS): Системы контроля версий предназначены для управления изменениями в исходном коде, документах и других файловых ресурсах в проектах разработки программного обеспечения. Они обеспечивают возможность отслеживать изменения, восстанавливать предыдущие версии, совместно работать над проектом и избегать конфликтов между разработчиками.

2. Какие виды CVS вам известны? Какими особенностями они обладают?  
ы

Централизованные системы контроля версий (Centralized Version Control Systems, CVCS), такие как CVS и Subversion (SVN), хранят все версии файлов на сервере, и разработчики могут работать с ними через клиентские приложения. Особенностью CVCS является централизованное хранилище версий.

Распределенные системы контроля версий (Distributed Version Control Systems, DVCS), такие как Git и Mercurial, каждому разработчику предоставляют полную копию репозитория, и они могут работать над проектом независимо, а затем обмениваться изменениями с другими разработчиками. Особенностью DVCS является децентрализованность и ускоренный доступ к истории версий.

3. На примере системы контроля версиями GIT опишите особенности подхода к работе с данными. Какими преимуществами данная система обладает?

Git - это распределенная система контроля версий, что позволяет каждому разработчику иметь полный репозиторий на своем локальном компьютере.

Работа с данными в Git основана на коммитах. Разработчик делает изменения, затем коммитит их, создавая новую версию данных.

Git предоставляет мощные инструменты для ветвления и слияния, что позволяет создавать и объединять ветки разработки.

Одной из важных особенностей Git является скорость и эффективность при работе с большими проектами и историями версий.

4. Опишите процесс установки И настройки GIT на свой ПК .

Загрузите установщик Git с официального сайта (https://git-scm.com/) и запустите его.

Следуйте инструкциям установщика, выбрав необходимые параметры.

После установки Git, откройте командную строку (терминал) и убедитесь, что Git установлен, введя команду git --version. Это покажет текущую версию Git.

Настройте глобальные параметры, такие как имя пользователя и адрес электронной почты, с помощью команд:

git config --global user.name "Ваше имя"

git config --global user.email "ваш@адрес.почты"  
**GITHUB**  
1. Опишите назначение сервиса GitHub.

Назначение: GitHub - это веб-сервис и платформа для хостинга репозиториев, созданных с использованием системы контроля версий Git. Он предоставляет разработчикам и командам возможность совместной работы над проектами, управления версиями кода, отслеживания изменений и обмена кодом.

Особенности: GitHub предоставляет функциональность для совместной разработки, управления задачами, отслеживания ошибок, создания веток разработки, а также обеспечивает инструменты для управления проектами и код-ревью.  
2. Опишите назначение сервиса Github Desktop.

Назначение: GitHub Desktop - это графический клиент для работы с репозиториями на GitHub. Он предоставляет удобный способ управления репозиториями без использования командной строки Git.

Особенности: GitHub Desktop позволяет пользователям клонировать репозитории, создавать ветки, коммитить изменения, создавать запросы на извлечение (пул-реквесты) и выполнять другие действия для совместной разработки. Это интуитивно понятное приложение для начинающих разработчиков.  
3. Поясните термины git и github:

1. Репозиторий - это место, где хранится и управляется исходный код проекта. Репозиторий может быть локальным (на вашем компьютере) или удаленным (на сервере, как GitHub). Он содержит историю версий и различные ветки разработки.  
2. Рабочая область и хранилище - это место, где вы редактируете файлы. Хранилище (или индекс) - это промежуточная зона, где вы подготавливаете файлы к коммиту. Вы перемещаете изменения из рабочей области в хранилище перед фиксацией (коммитом) изменений.  
3. Форк - это копия чужого репозитория, которую вы создаете, чтобы внести в нее изменения. Это часто используется для внесения вклада в проекты с открытым исходным кодом.  
4. Ветка - это параллельная линия разработки в репозитории, которая позволяет вам работать над функциональностью или исправлениями багов независимо от основной ветки. Ветки могут быть созданы, изменены и объединены.  
5. Мастер - это основная ветка в репозитории, которая содержит стабильную версию проекта. В нее обычно сливаются другие ветки после завершения разработки.  
6. Коммит - это фиксация изменений в репозитории. Он создает новую версию данных и сохраняет их в истории версий.  
7. Пул - это операция, при которой изменения из удаленного репозитория объединяются с вашим локальным репозиторием. Это позволяет вам получать обновления от других разработчиков.  
8. Пуш - это операция, при которой вы отправляете свои коммиты в удаленный репозиторий, обновляя его версию.  
9. Пулрекквест - это запрос на включение ваших изменений из вашей ветки в другую ветку, часто в основную ветку проекта. Он часто используется для обсуждения и проверки изменений другими разработчиками.  
10. Мердж - это операция, при которой изменения из одной ветки объединяются с другой веткой или мастером.  
11. Кодревью - это процесс, при котором другие разработчики анализируют и оценивают ваш код перед его внесением в репозиторий, обеспечивая качество кода и выявляя ошибки.

2. Выполнить практическую часть работы:  
  
Задание 1.  
  
1. Создать свой профиль на github.

2. Установить на свой ПК GitHub Desktop и создать в нем новый репозиторий.

3. Разместить в нем свой КУРСОВОЙ ПРОЕКТ ( программу-исходники, файл SETUP, ПЗ, ГЧ)

4. Сделать интеграцию вашего репозитория с GitHub.

3. Ссылку на свой профиль в github на Курсовой проект сбросить в google class в ЛР работу.

Задание 2.  
На примере любого вашего проекта по c# , python И др. Необходимо показать основные возможные Типовые действия с проектом:  
Создание репозитория

Создание мастера, ветки, форка  
Выполнение действий: пуш, пул, коммит, пулреквест, мердж, Кодревью.