**Лабораторная работа 20**

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

**Контрольные вопросы**

**Системы контроля версий CVS**

1. **Назначение системы CVS**: CVS (Concurrent Versions System) предназначен для управления изменениями в файлах и директориях с помощью версионного контроля. Он позволяет отслеживать историю изменений, восстанавливать предыдущие версии и сотрудничать с другими разработчиками.
2. **Виды CVS и их особенности**:
   * **CVS**: Оригинальная система контроля версий, которая поддерживает параллельную работу нескольких разработчиков.
   * **CVSNT**: Расширенная версия CVS, добавляющая поддержку Windows и улучшенные функции.
   * **Subversion (SVN)**: Современная система контроля версий, которая улучшает многие недостатки CVS и предлагает более мощные инструменты для управления проектами.

**Система контроля версий GIT**

1. **Особенности подхода к работе с данными**: GIT - это распределенная система контроля версий, которая позволяет каждому разработчику иметь полную копию всей истории проекта. Это обеспечивает высокую гибкость и независимость от центрального сервера.
2. **Преимущества GIT**:
   * **Распределенность**: Каждый разработчик имеет полную копию проекта, что упрощает синхронизацию и восстановление данных.
   * **Быстрая работа**: Операции с GIT выполняются быстро благодаря локальному хранению данных.
   * **Мощные инструменты для совместной работы**: Ветвление, слияние, откат изменений и многое другое.

**Процесс установки и настройки GIT на ПК**

1. **Установка GIT**: Скачать и установить GIT с официального сайта.
2. **Настройка GIT**: Настроить имя пользователя и адрес электронной почты с помощью команды git config --global user.name "Ваше имя" и git config --global user.email "ваша\_почта@example.com".

**Сервис GitHub**

1. **Назначение сервиса GitHub**: GitHub - это платформа для хостинга и совместной работы над проектами с использованием системы контроля версий GIT.
2. **Назначение сервиса GitHub Desktop**: GitHub Desktop - это графический интерфейс для работы с GitHub, который упрощает процессы коммита, пуша и другие операции.

**Термины Git и GitHub**

1. **Репозиторий**: Коллекция файлов и истории изменений.
2. **Рабочая область и хранилище**: Рабочая область - это текущая копия файлов проекта, хранилище - это место, где хранится история всех изменений.
3. **Форк**: Копия репозитория на другом аккаунте.
4. **Ветка**: Параллельная линия разработки внутри репозитория.
5. **Мастер**: Основная ветка проекта.
6. **Коммит**: Запись о изменениях в репозитории.
7. **Пул**: Коллекция всех коммитов в репозитории.
8. **Пуш**: Операция отправки изменений из локального репозитория в удаленный.
9. **Пулрекквест**: Операция получения изменений из удаленного репозитория в локальный.
10. **Мердж**: Операция объединения нескольких веток в одну.
11. **Кодревью**: Процесс проверки и ревизии кода другими разработчиками перед его принятием.