

## **Лабораторная работа №20**

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

#### **Контрольные вопросы:**

#### **Системы контроля версий GIT**

##### **1. Опишите назначение систем CVS.**

Системы контроля версий (CVS) предназначены для управления изменениями в коде, документах и других данных. Они позволяют отслеживать все модификации, возвращаться к предыдущим версиям, управлять совместной работой нескольких разработчиков и минимизировать конфликты при одновременной работе.

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

1) Локальные системы контроля версий – хранят изменения только на локальном компьютере, менее надежны из-за риска потери данных;

2) Централизованные системы (например, Subversion, Perforce) – все данные и история изменений хранятся на одном сервере, что упрощает совместную работу, но серверное зависание может стать проблемой;

3) Распределенные системы (например, GIT, Mercurial) – каждая копия репозитория содержит полную историю изменений, что обеспечивает надежность и возможность работать без постоянного подключения к серверу.

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

В GIT данные хранятся в виде снимков (snapshots) всего проекта. При каждом сохранении (коммите) GIT сохраняет текущую версию всех файлов, оптимизируя объем памяти.

#### **Преимущества:**

1) Скорость – локальные операции выполняются быстро;

2) Надежность – легко восстанавливать предыдущие версии;

3) Ветвление – простое и гибкое управление ветками;

4) Распределенность – можно работать автономно, а затем синхронизировать изменения.

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

Установка: скачать установочный файл с официального сайта и следовать инструкциям для соответствующей ОС.

Настройка: открыть терминал и выполнить команды:

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

```
git config --global user.email "ваш.email@example.com"
```

Опционально настроить предпочитаемый текстовый редактор.

## **GITHUB**

1. Опишите назначение сервиса GitHub.

GitHub – это облачная платформа для хранения репозиториях GIT, совместной работы над проектами, обмена кодом, отслеживания задач и управления командными проектами.

2. Опишите назначение сервиса Github Desktop.

GitHub Desktop – это удобное графическое приложение для работы с репозиториями на GitHub. Оно упрощает управление проектами GIT без необходимости использовать командную строку.

3. Поясните термины git и github:

1. Репозиторий – хранилище, содержащее проект и всю его историю изменений.

2. Рабочая область – локальные файлы проекта.

Хранилище – докальный или удаленный репозиторий с историей изменений.

3. Форк – копия чужого репозитория на вашем аккаунте для работы с кодом.

4. Ветка – отдельная линия разработки, позволяющая работать над новыми функциями, не затрагивая основную версию.

5. Мастер – основная ветка в репозитории, в которой обычно хранится стабильный код.

6. Коммит – сохранение изменений в локальном репозитории с описанием.

7. Пул – получение новых изменений из удаленного репозитория.

8. Пуш – отправка изменений в удаленный репозиторий.

9. Пулреквест – запрос на слияние изменений с основной веткой, который можно обсудить и проверить.

10. Мердж – слияние веток, объединение изменений в одну версию.

11. Кодревью – проверка и обсуждение изменений кода перед их слиянием.