**Тема:** Работа с системой контроля версий Git

**Цель:** научиться создавать и управлять репозиториями, использовать тэги для версионирования, реализовать взаимодействие между фронтенд и бэкенд частями на разных версиях.

**Описание л.р.:**

Необходимо создать два отдельных репозитория: frontend и backend.

В каждом репозитории создать ветку main.

Реализация версий 1.0:

Frontend 1.0:

Создайте страницу с полем для ввода текста и кнопкой "Отправить".

Реализуйте логику, при которой нажатие на кнопку отправляет введенные данные на бэкенд.

Backend 1.0:

Реализуйте бэкенд, который принимает данные, отправленные с фронтенда.

Сохраните принятые данные в текстовый файл (data.txt) на сервере.

В каждом репозитории добавьте тэг v1.0 после завершения разработки.

Реализация версии 2.0:

Frontend 2.0:

Модифицируйте версию 1.0, добавив дополнительное поле для ввода и кнопку.

Реализуйте функционал, при котором нажатие на новую кнопку отправляет запрос на бэкенд и выводит результат запроса в отдельное новое поле на странице.

Backend 2.0:

Модифицируйте версию 1.0, добавив функционал чтения данных из data.txt.

Реализуйте логику, при которой бэкенд возвращает содержимое файла (data.txt) в ответ на запрос от фронтенда.

В каждом репозитории добавьте тэг v2.0 после завершения разработки.

**Требования:**

1. Используйте систему контроля версий Git для управления кодом.
2. Для каждой версии фронтенда и бэкенда должны быть отдельные тэги (v1.0, v2.0).
3. При переходе между версиями (v1.0 и v2.0) все работает корректно.

**Результат выполнения:**

1. Два репозитория (frontend и backend), каждый из которых имеет две версии (1.0 и 2.0).
2. Каждая версия помечена соответствующим тэгом.
3. Функциональная реализация взаимодействия между фронтенд и бэкенд частями приложения на обеих версиях.
4. Отчёт о проделанной работе.

**Примечание:**

Репозитории: Gitlab или Github

ЯП: Python или Java

Отчёт должен содержать: титульный лист и описание хода работы.