**Лабораторна робота №1: Основи систем контролю версій та мінімальний вебпроєкт**

**Мета:**

Ознайомитися з системами контролю версій, навчитися працювати з гілками та репозиторіями, а також створити мінімальний вебпроєкт на базі обраних технологій.

**Загальні вимоги:**

Максимальна кількість балів за виконання роботи: 10 балів.

Обовʼязкові завдання (на 8 з 10 балів).

Додаткові завдання (на 2 з 10 балів).

**Обов’язкові завдання (8 балів):**

* **Вибір технологій:**

Оберіть мову програмування та відповідний вебфреймворк. Можливі варіанти:

Java + Spring Boot

JavaScript + Node.js

Python + Django

Ruby + Rails

Ви також можете вибрати іншу мову програмування або вебфреймворк за вашим бажанням.

* **Створення мінімального вебпроєкту:**

Використовуючи обрані технології, створіть проєкт, що запускає вебсервер.

Реалізуйте один ендпоінт GET /products/{productId}, де productId — це ціле число.

Ендпоінт має повертати JSON у форматі:

{ "id": "{productId}", "name": "{productId} name" }

Наприклад, при запиті GET /products/1, відповідь повинна бути:

{ "id": "1", "name": "1 name" }

* **Ініціалізація репозиторію:**

Ініціалізуйте локальний Git-репозиторій для вашого проєкту.

Створіть основну гілку (main).

* **Синхронізація з віддаленим репозиторієм:**

Створіть віддалений репозиторій на одному з сервісів (GitHub, GitLab, Bitbucket тощо).

Синхронізуйте локальний репозиторій з віддаленим (push до віддаленого репозиторію).

* **Гілка release-candidate:**

Створіть гілку release-candidate з основної гілки (main).

Відправте її на віддалений репозиторій.

* **Робота з гілками:**

Створіть нову гілку для функціональності від гілки main або гілку виправлення дефекту з release-candidate.

Додайте в неї зміни та зробіть push на віддалений репозиторій.

Створіть запит на злиття (merge request або pull request) цієї гілки з однією з основних гілок (main, release-candidate).

**Демонстрація:**

Під час захисту роботи продемонструйте створення гілки, внесення змін та створення запиту на злиття.

**Додаткові завдання (2 бали):**

* Налаштуйте віддалений репозиторій так, щоб обовʼязково використовувалася стратегія злиття Merge without commit (стратегія fast-forward) і потребувався rebase перед злиттям, якщо головна гілка пішла далі.
* Налаштуйте захищену гілку (protected branch) для гілки release-candidate.
* Налаштуйте доступ до віддаленого репозиторію за допомогою SSH-ключа. Забороніть використання логіна та пароля для доступу до репозиторію.