

Міністерство освіти і науки України

Національний університет «Львівська політехніка»



Лабораторна робота № 1

з дисципліни: «Автоматизоване проектування комп'ютерних систем»,

на тему: «Створення GIT репозиторію.»

Виконав:ст. гр. КІ-410

Ковальчук А.О.

Прийняв:

Кіцера А.О.

Львів – 2024

Task 1. Initiate GIT repository:

1. Create a github repo with the name “**csdt<YY₁YY₂><group><student's full name><student's number>**”, main branch develop.

YY₁ – Start studying year YY₂ – End studding year

For example: 2019-2020 -> **1920** Full example **csdt1920ki47spitzeras03**

2. Add access for the author.

3. Create a branch **feature/develop/<task number>**. For example **feature/develop/task1**.

4. Create **README** file with:

a. details about repo;

b. task details;

c. student number and details from **Table 1. Tasks by students**;

d. add details about technology, program language, and HW that will be used in next tasks.

5. Create **GIT TAG**: **<PROJECT NAME>_<VERSION>_WW<YYWWD>**

· YY – current year;

· WW – work week;

· D – current day number of weak.

6. Create pull request with name **task1** and submit **lecturer** as reviewer.

7. After the reviewer approved – merge request into **develop** branch.

Acceptance criteria:

The created repo link with the “develop” as default branch and README.ML file.

Вариант 8:

Student number	Game	config format
8	rock paper scissors	JSON

Теоретичні відомості

Git - це розподілена система керування версіями, розроблена для відстеження змін у програмному коді та спрощення спільної роботи над проектами між розробниками.

Git працює на основі концепції "знімків" (snapshots). Кожен коміт фіксує стан проекту на певний момент, що дозволяє легко відновлювати чи переглядати попередні версії коду. Це надає впевненість розробникам у тому, що їхні зміни зафіксовані та можна повертатися до будь-якого попереднього стану.

Гілкування в Git нагадує створення альтернативних реальностей для вашого проекту. Розробники можуть створювати гілки для експериментів, розробки нових функцій чи виправлення помилок, не турбуючись про вплив на головний код. Після успішних експериментів зміни можна об'єднати через злиття.

Git також робить спільну роботу більш ефективною за допомогою Pull Requests (запитів на злиття). Це механізм для обговорення та перегляду змін перед їхнім об'єднанням в основну гілку проекту. Це поліпшує якість коду та сприяє колективній відповідальності.

Коміт (Commit): Це фіксація змін у репозиторії. Кожен коміт містить інформацію про зміни, внесені до проекту, і включає у себе усю необхідну інформацію для відновлення коду в конкретний момент часу.

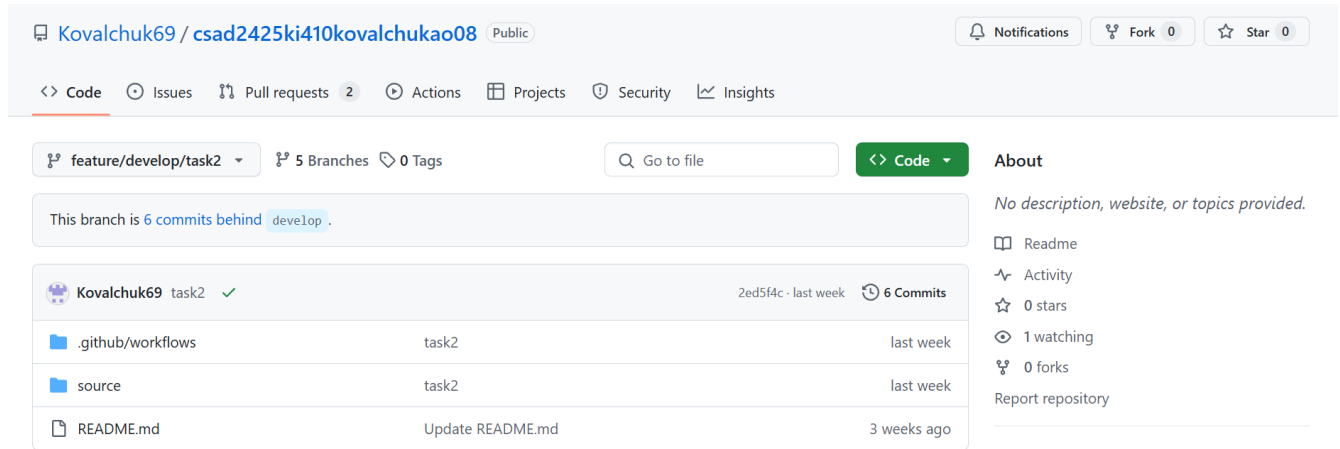
Гілка (Branch): Це ізольована лінія розвитку у репозиторії. Гілки дозволяють розробникам працювати паралельно над різними аспектами проекту, не впливаючи один на одного. Після завершення роботи над гілкою її можна злити з основною гілкою.

Злиття (Merge): Це процес об'єднання змін з однієї гілки в іншу. Git дозволяє злити зміни з однієї гілки в іншу, щоб створити єдину версію проекту.

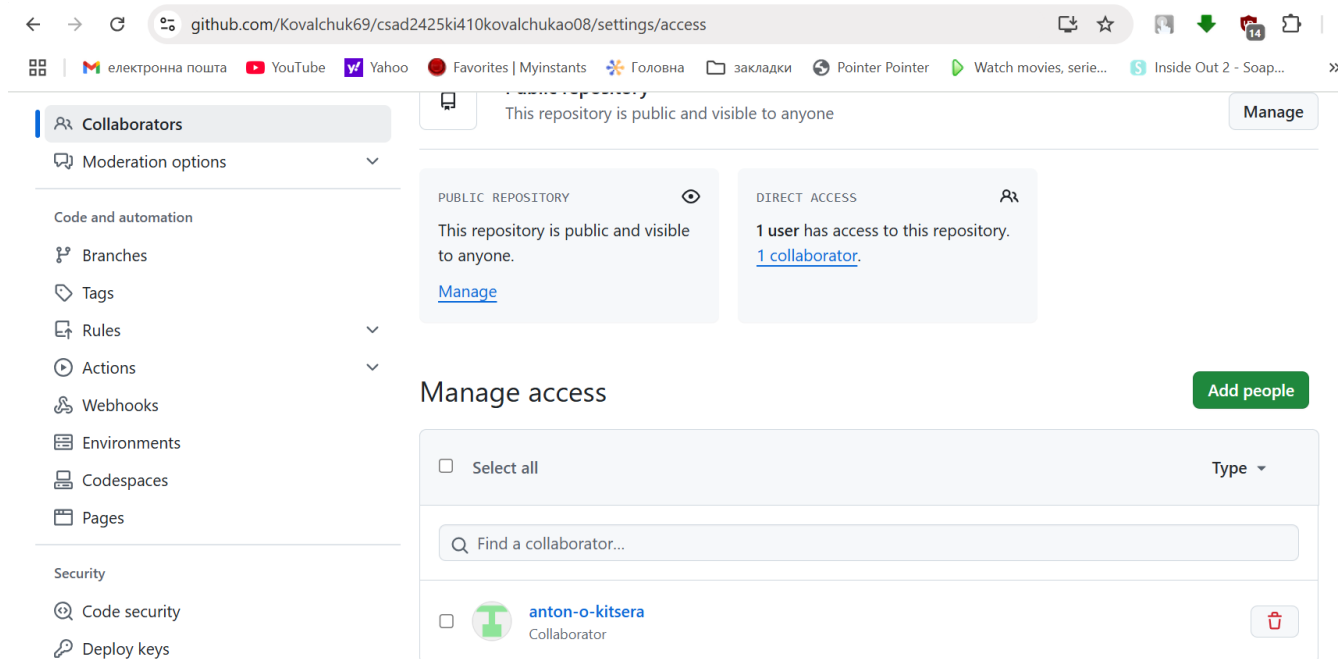
Pull Request (Запит на злиття): Це механізм, який дозволяє розробникам обговорювати та рецензувати зміни перед їхнім об'єднанням в основну гілку. Зазвичай використовується в співпраці розробників у великих проектах.

Хід роботи

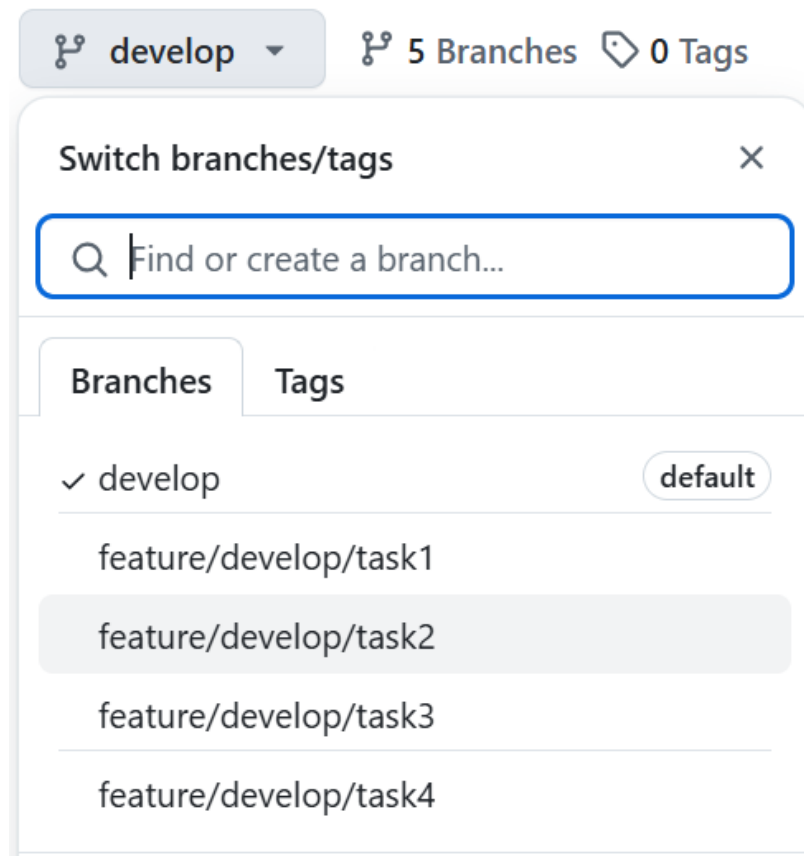
1. Створив новий репозиторій на GitHub: csad2425ki410kovalchukao08



2. Надав доступ до репозиторію викладачу.




3. Ініціалізував репозиторій та змінив назву основної гілки на "develop" та створив нову гілку feature/develop/task1.




4. Створив README файл і переглянув його наповнення.

[csad2425ki410kovalchukao08](#) / README.md

 **Kovalchuk69** Update README.md bcdfe9e · 2 weeks ago [History](#)

PreviewCodeBlame

18 lines (11 loc) · 737 Bytes

 Code 55% faster with GitHub Copilot

RawDownloadEdit

1.1. Details about repo

This repository is created to store lab works for the CSAD course. It will include completed labs and all the steps of development from start to finish.

1.2. Task details

This project is dedicated to the development of a digital version of the classic game Rock paper scissors.

1.3. Student number and details

Student number: 8 Game: Rock paper scissors Config format: JSON

1.4. Details about technology, program language, and HW

In this project i will use a client-server architecture. The client is an application that gets data from a server, represented by an Arduino Uno board that performs all the calculations and returns the results to the app. This project will be developed in C++ language.

5. Надіслав запит на об'єднання змін із гілки "task1" та вказав викладача як особу, що проведе рецензію. Після того, як викладач підтвердив зміни, здійснив злиття з гілкою develop.

The screenshot shows a GitHub pull request interface for a repository named 'task1 #1'. At the top, it indicates that 'Kovalchuk69' merged 4 commits into the 'develop' branch from the 'feature/develop/task1' branch last week. Below this, a summary bar shows 0 conversations, 4 commits, 0 checks, and 1 file changed. The main content area displays a comment from 'Kovalchuk69' stating 'No description provided.' and a commit history showing four 'Update README.md' commits. A review from 'anton-o-kitsera' is shown as approved. The right sidebar contains metadata: Reviewers (anton-o-kitsera), Assignees (None), Labels (None), Projects (None), Milestone (None), and Development (None). A 'Revert' button is visible at the bottom right.

Висновок:

Під час виконання лабораторної роботи створено репозиторій на GitHub під назвою "csad2425ki410kovalchukao08". Доступ до нього був наданий викладачу. Далі було ініціалізовано репозиторій та перейменовано основну гілку на "develop". Також створено гілку "feature/develop/task1" та файл README, а також був створений pull request. Після підтвердження викладачем змін, вони були об'єднані з гілкою "develop".