

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ЛЬВІВСЬКА ПОЛІТЕХНІКА



**АВТОМАТИЗОВАНЕ ПРОЕКТУВАННЯ  
КОМП'ЮТЕРНИХ СИСТЕМ**

Завдання 1: “Ініціалізація GIT репозиторію”

**Виконав:**

студент групи КІ-410

Іванюк О.О.

**Приняв:**

Кіцера А. О.

## Методичні вказівки

### Опис завдання

#### №1

Створити репозиторій на GitHub з назвою "csad<YY1YY2><група><повне ім'я студента><номер студента>", де основна гілка — develop. YY1 – рік початку навчання

YY2 – рік закінчення навчання

Наприклад: 2019-2020 -> 1920

Повний приклад: csad1920ki47spitzeras03

#### №2

Додати доступ для автора.

#### №3

Створити гілку feature/develop/<номер завдання>. Наприклад, feature/develop/task1.

#### №4

Створити файл README, де вкажіть:

- a. деталі про репозиторій;
- b. деталі завдання;
- c. номер студента та інформацію.
- d. додати інформацію про технології, мову програмування та обладнання, яке буде використовуватися в наступних завданнях.

#### №5

Створити GIT TAG: <ІМ'Я ПРОЄКТУ>\_<ВЕРСІЯ>\_WW<YYWWD> · YY – поточний рік;

WW – робочий тиждень;

D – номер дня тижня.

#### №6

Створити pull request з назвою task1 та надайте викладача як рецензента.

#### №7

Після затвердження рецензента – виконайте злиття запиту в гілку develop.

## Критерії прийняття

Посилання на створений репозиторій з гілкою "develop" як основною та файлом README.ML.

## Теорія

### Git

Розподілена система контролю версій, яка використовується для відстеження змін у файлах і координації роботи між кількома розробниками над одним проектом. Git дозволяє користувачам зберігати знімки стану проекту (коміти), повертатися до попередніх версій файлів і об'єднувати зміни, зроблені різними розробниками в одну гілку (branch).

**Репозиторій** — це сховище, яке містить проект і всю його історію змін, відслідковану за допомогою системи контролю версій Git.

**Branch (гілка)** — це окрема версія історії комітів у Git, яка дозволяє працювати над змінами або новими функціями незалежно від основної гілки проекту. Гілки дозволяють одночасно розвивати різні частини проекту, не впливаючи на основну робочу версію, поки зміни не будуть готові.

**Tag (тег)** — це мітка, яка вказує на певний коміт в історії проекту. Теги зазвичай використовуються для позначення важливих версій проекту, таких як випуски релізів (наприклад, версії 1.0.0, 2.0.1 тощо). Тег фіксує конкретний стан репозиторію на певному етапі, дозволяючи легко повернутися до цього коміту або використовувати його як орієнтир.

# Виконання завдання

## Хід виконання

### Створення репозиторію

Створив локально GIT репозиторій із гілкою за замовчуванням: “develop”. Після створення репозиторію додавив README.md файл з текстом “There's nothing here yet)” і створив ініціалізуючий коміт: “Initial commit”.

Створив віддалений репозиторій на GitHub з назвою “[csad2425ki410ivaniukoo07](#)”, яка відповідає заданому патерну назви у методичних вказівках. Завантажив зміни локального репозиторію у віддалений.

### Створення гілки завдання

Створив нову гілку “feature/develop/task1”, де змінив вміст файлу README.md, описавши інформацію, яка вимагалася у методичних вказівках. Вміст файлу наведений у лістингу 1.

Лістинг 1. Новий вміст файлу README.md.

```
# CSAD 2024-2025 Group KI-410 Ivaniuk Oleh 07

## Project Overview
This repository is created for the Computer Systems Automated Design course. It contains the implementation of assigned tasks using Arduino and C++ for hardware and software development.

## Task Details
- **Student Number**: 07
- **Game**: Tic-Tac-Toe 3x3
- **Configuration Format**: XML
- Play Modes:
  - Man vs AI
  - Man vs Man
  - AI vs AI (Random Move or Win Strategy)

## Technology Stack
Programming Language: C++
Hardware Platform: Arduino
Development Tools: Arduino IDE, Git, GitHub
```

Створивши коміт із змінами прив’язав до нього тег з назвою “[CSAD 1.0 WW24401](#)”, який назвав згідно методичних вказівок.

### Внесення змін в репозиторій

Здійснивши всі дії вище, залив гілку і тег у віддалений репозиторій, після чого створив pull request. Додавши викладача до репозиторію, додав його як reviewer-а до pull request-у. Отримавши погодження на заливання змін, змержив гілку feature/develop/task1 у develop.

## **Висновки**

Протягом цього завдання було створено GIT репозиторій, де в подальшому будуть виконуватися наступні завдання. Під час виконання цього завдання було визначено стандартний алгоритм роботи над завданнями цього курсу.

## **Список використаних джерел**

Список інтернет ресурсів наведено нижче:

1. C# Guide - .NET managed language | Microsoft Learn  
[Електронний ресурс] – Режим доступу до ресурсу:  
<https://learn.microsoft.com/en-us/dotnet/csharp/>
2. .NET Multi-platform App UI (.NET MAUI) | .NET  
[Електронний ресурс] – Режим доступу до ресурсу:  
<https://dotnet.microsoft.com/en-us/apps/maui>
3. Blazor | Build client web apps with C# | .NET  
[Електронний ресурс] – Режим доступу до ресурсу:  
<https://dotnet.microsoft.com/en-us/apps/aspnet/web-apps/blazor>
4. UNO R3 - Arduino Docs  
[Електронний ресурс] – Режим доступу до ресурсу:  
<https://docs.arduino.cc/hardware/uno-rev3/>
5. Home > xUnit.net  
[Електронний ресурс] – Режим доступу до ресурсу:  
<https://xunit.net/>