**МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ЛЬВІВСЬКА ПОЛІТЕХНІКА**



**АВТОМАТИЗОВАНЕ ПРОЕКТУВАННЯ**

**КОМП’ЮТЕРНИХ СИСТЕМ**

**Task1: Initiate GIT repository**

**Виконав:**

**КІ-401**

**Янкович Н. Р.**

**Прийняв: Федак П. Р.**

**Львів 2024**

**Завдання:**

1. Create a github repo with the name “csad<YY1YY2><group><student's full

name><student's number>”, main branch develop

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>

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

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

**Теорія:**

GitHub — веб-сервіс для зберігання та керування проєктами на основі системи контролю версій Git. Він дозволяє розробникам спільно працювати над кодом, відстежувати зміни, вносити виправлення та обмінюватися проєктами. GitHub також широко використовується для відкритого програмного забезпечення та співпраці над різними IT-проєктами.

Git — система контролю версій, яка дозволяє відстежувати зміни у файлах, зокрема в програмному коді. Вона допомагає керувати різними версіями проєкту, зберігати історію змін і дозволяє кільком людям працювати над одним проєктом без конфліктів. За допомогою Git можна повертатися до попередніх версій, зливати зміни з різних гілок (branch) та працювати над кодом паралельно з іншими розробниками.

Git Bash — командний інтерфейс для Windows, який дозволяє користувачам виконувати команди Git у середовищі, схожому на термінал Linux. Він встановлюється разом з Git для Windows і надає можливість використовувати Git та інші командні утиліти Unix прямо у Windows.

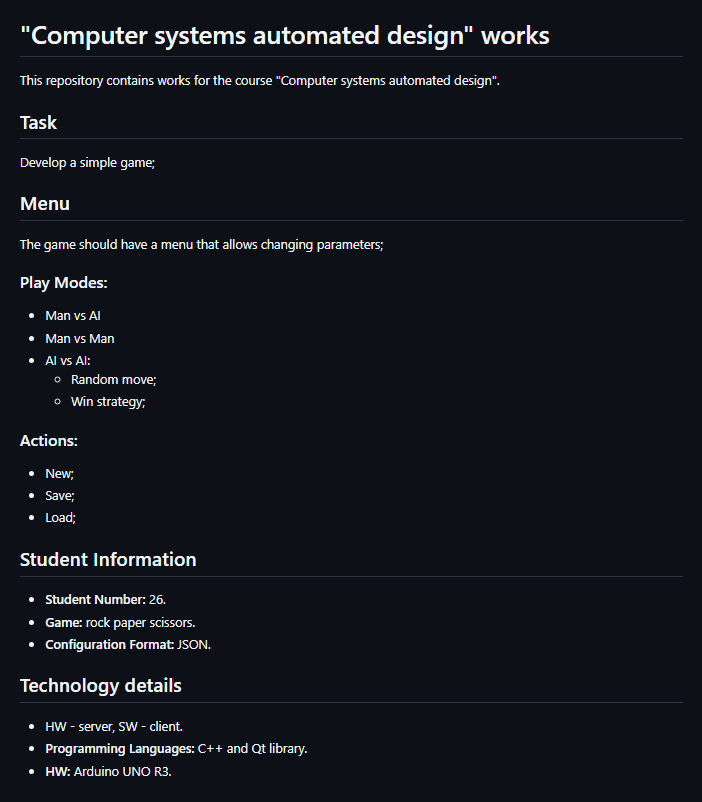
**Реалізація:**

1. Створив репозиторій у Github з відповідним ім’ям.
2. Запустив git bash, ініціалізував репозиторій командою git init.
3. Написав команду git remote для прив’язки репозиторію.
4. Створив гілку develop.
5. Ввів команду git commit –allow-empty для дозволу на пустий коміт.
6. Відправив гілку develop на репозиторій комадною git push -u origin develop.
7. Створив гілку feature/develop/task1 комадною git checkout -b feature/develop/task1.
8. Створив файл README.md командою touch.
9. Написав у файл потрібну і відредаговану інформацію.
10. Створив git tag відповідно до завдання та відправив його:  
    git tag csad2125ki401yankovychnr26\_1.0\_WW24406;

git push origin csad2125ki401yankovychnr26\_1.0\_WW24406.

1. Завантажив файл на гілку feature/develop/task1 та створив pull request з ім’ям task1.

Далі наведено вміст README.md та його вигляд на Github.

****

*Рис.1. Вмістиме README.md*

**Висновок:** виконуючи цю лабораторну роботу, я створив репозиторій, основну та додаткову для завдання гілки, створив README файл про мій репозиторій та завдання, оформив його. Створив git tag. Виконав pull request.

**Списки використаної літератури:**

1. https://en.wikipedia.org/wiki/GitHub
2. Oggl, B., & Kofler, M. (2022). Git: Project Management for Developers and DevOps. Apress. 408 cторінок
3. Чакон, С., & Штрауб, Б. (2020). GIT для професійного програміста. 496 сторінок.