**МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ЛЬВІВСЬКА ПОЛІТЕХНІКА**



**АВТОМАТИЗОВАНЕ ПРОЕКТУВАННЯ**

**КОМП’ЮТЕРНИХ СИСТЕМ**

**Task1: Initiate GIT repository**

**Виконав:**

**КІ-404**

**Копій О.Р.**

**Прийняв: Федак П.Р.**

**Львів 2024**

**Опис завдання:** В даній лабораторній роботі потрібно ініціювати репозиторій у Github, а саме:

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.

**Теоретичні відомості:**

GitHub — один з найбільших вебсервісів для спільної розробки програмного забезпечення. Існують платні та безплатні тарифні плани користування. Базується на системі керування версіями Git і розроблений на Ruby on Rails і Erlang компанією GitHub, Inc (спочатку вона називалася Logical Awesome).

Git — розподілена система керування версіями файлів та спільної роботи. Проєкт створив Лінус Торвальдс для керування розробкою ядра Linux, а сьогодні підтримується Джуніо Хамано (англ. Junio C. Hamano). Git є однією з найефективніших, надійних і високопродуктивних систем керування версіями, що надає гнучкі засоби нелінійної розробки, що базуються на відгалуженні та злитті гілок. Для забезпечення цілісності історії та стійкості до змін заднім числом використовуються криптографічні методи, також можлива прив'язка цифрових підписів розробників до тегів і комітів.

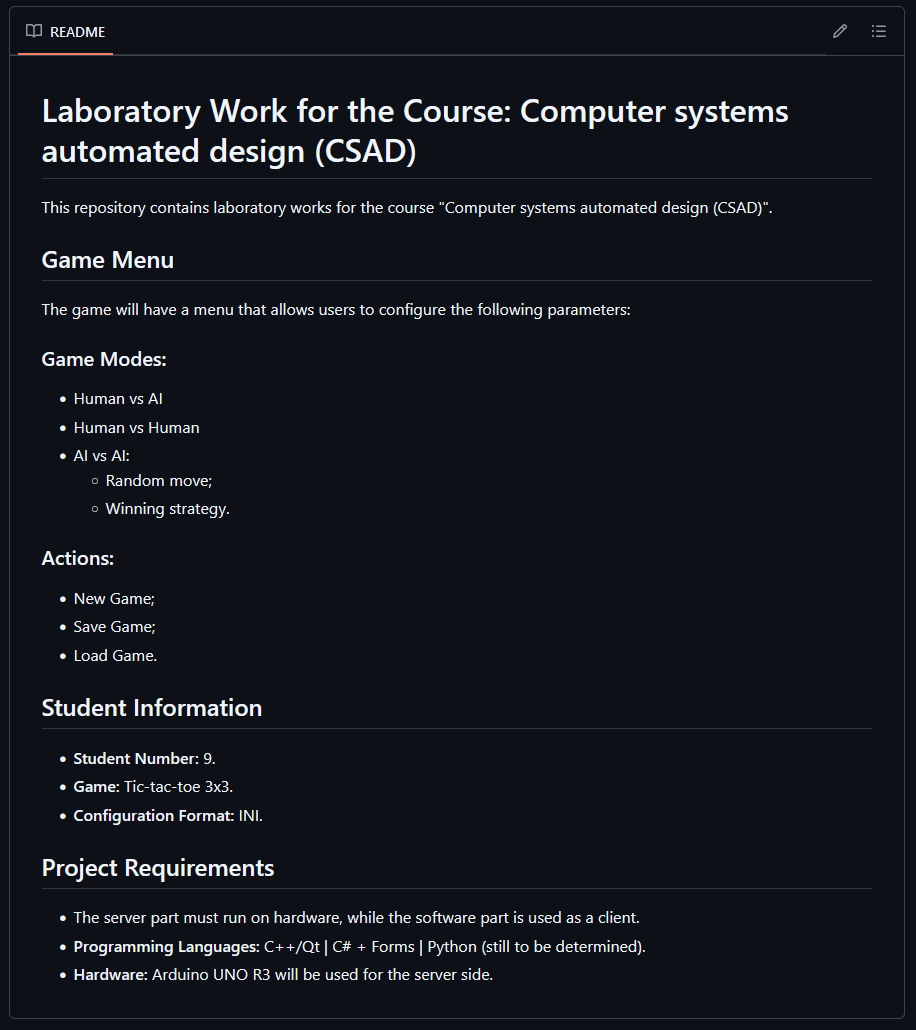
Основні команди в git:

* Git add
* Git status
* Git diff
* Git commit
* Git reset
* Git rm
* Git mv
* Git clean

**Деталі реалізації:**

Було виконано наступні кроки для реалізації завдання:

* Створення репозиторію у Github
* Ініціалізація репозиторію завдяки команді git init у git bash
* Прив’язка локального репозиторію до хмарного завдяки команді git remote add origin <посилання на репозиторій Github>
* Створення нової default вітки develop git checkout -b develop
* Створення першого пустого коміту для злиття вітки develop на хмару git commit --allow-empty -m "Initial empty commit on develop"
* Команда git push -u origin develop для відправки локальної вітки на віддалений репозиторій
* Команда git checkout -b feature/develop/task1 для створення нової вітки згідно завдання
* Створення README файлу touch README.md
* Редагування README файлу згідно до вимог завдання
* Створення гіт тегу: git tag csad2125ki404kopiyor09\_1.0\_WW24406
* Відправка гіт тегу: git push origin csad2125ki404kopiyor09\_1.0\_WW24406
* Відправка README файлу: git add README.md та git push -u origin feature/develop/task1

****

*Рис.1.Вмістиме файлу README.md*

**Висновок:** На даній лабораторній роботі я створив репозиторій у Github, створив декілька віток,а саме: develop(default branch) та feature/develop/task1. Створив README файл з деталями про репозиторій, моє завдання та план роботи, також було створено git tag та виконано pull request. Відповідно, за результатами ми отримали новостворений Github з двома вітками, тегом та файлом README.md.

**Списки використаної літератури:**

1. Bell, P., & Beer, B. (2014). Introducing GitHub: A Non-Technical Guide. 139 сторінок.
2. Chacon, S., & Straub, B. (2014). Pro Git (2-ге видання). 440 сторінок.
3. https://en.wikipedia.org/wiki/GitHub