**Міністерство освіти і науки України**

**Національний технічний університет**

**«Дніпровська політехніка»**

****

**ЗВІТ**

**Практична робота №2**

**з дисципліни**

# «Програмування в середовищі Java»

**Виконав:**

студент гр. 122-21-3

Кабаченко О. В.

**Прийняв:**

Доцент каф. САУ

Алексєєв О. М.

**м. Дніпро**

**2025  рік**

**Хід роботи**

1) Для початку була відкрита веб-сторінка GitHub, після переходу по посиланню <https://github.com>. Зовнішній вигляд сторінки показано на рисунку 1.

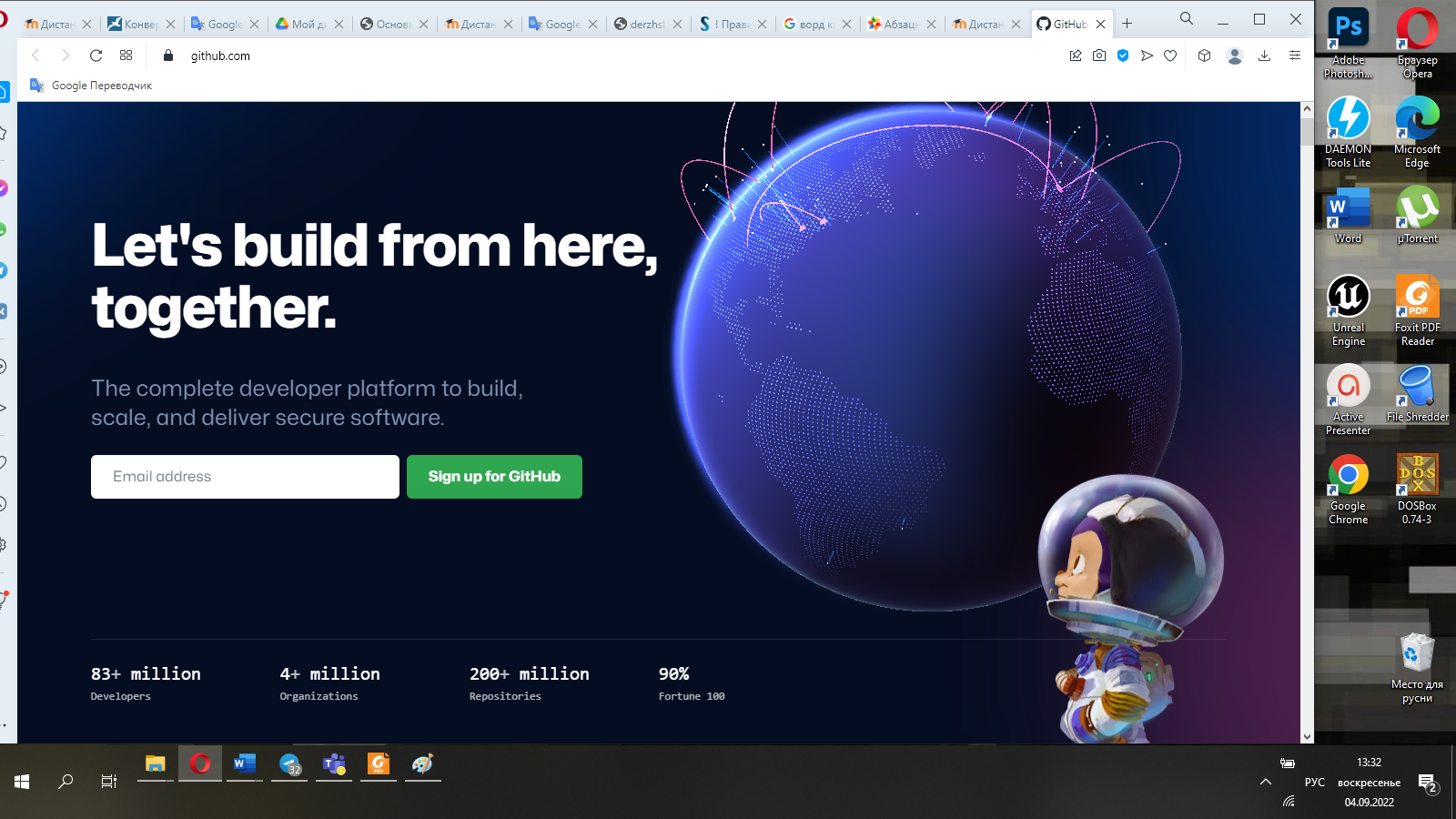


Рисунок 1 – Зовнішній вигляд сторінки GitHub

2) Далі натиснуто на розділ “Sign UP” та уведено необхідну для реестрації інформацію(результат зображено на рисунку 2).

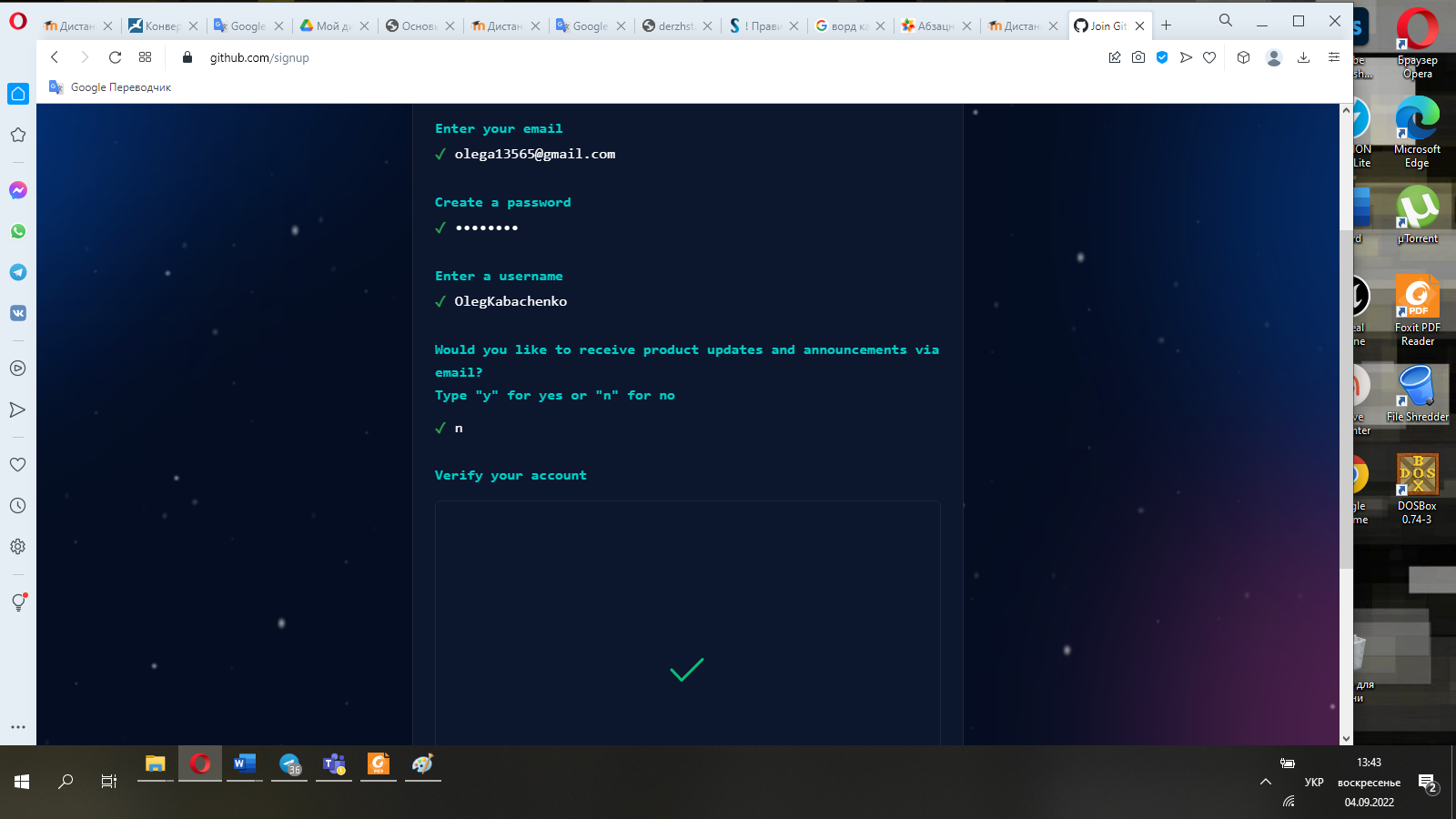


Рисунок 2 – Реєстрація на сайті

3) Після попереднього кроку натиснуто кнопка “Create account”, та уведено код, який було надіслано на поштову адресу. Переглянути сторінку з уведенням коду можливо на рисунку 3.

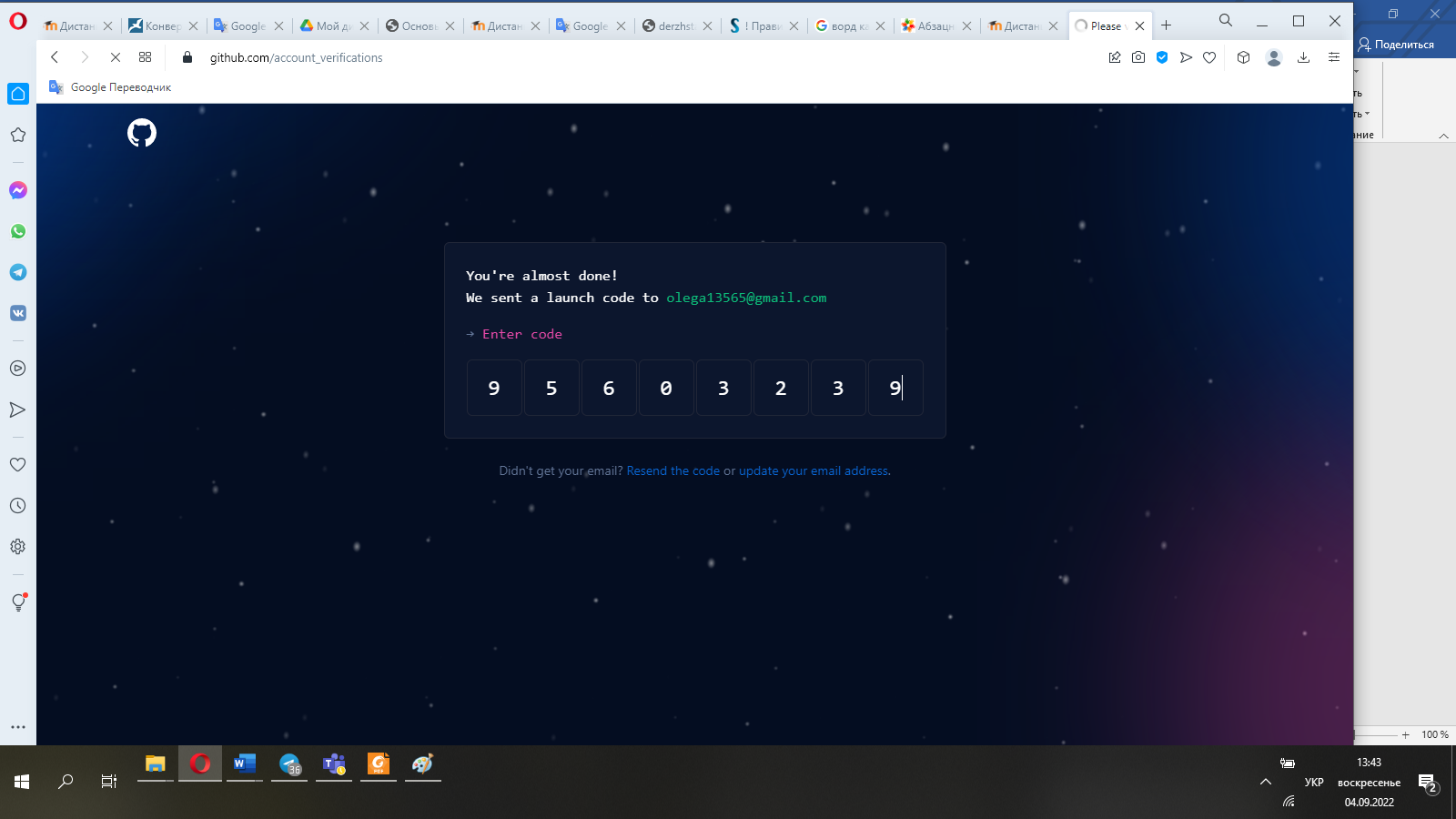


Рисунок 3 – Уведення коду підтвердження

4) Після підтвердження коду, автоматично була відкрита сторінка персоналізації, яку було попущено шляхом натискання на кнопку “Skip Personalization” (зовнішній вигляд сторінки зображено на рисунку 4). Потім система відкрила так званий “dashboard”, який зображено на рисунку 5.

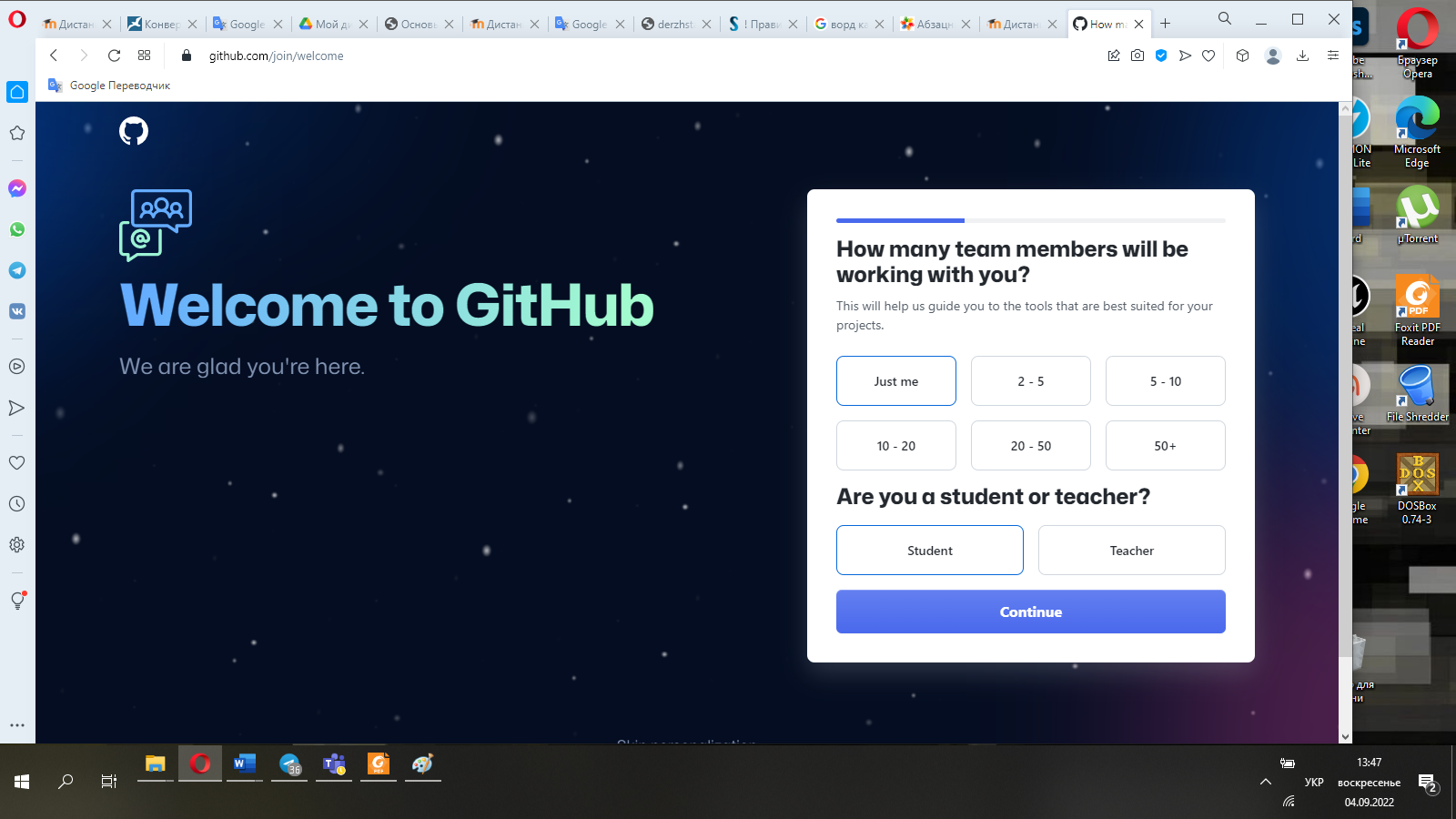


Рисунок 4 – Сторінка персоналізації

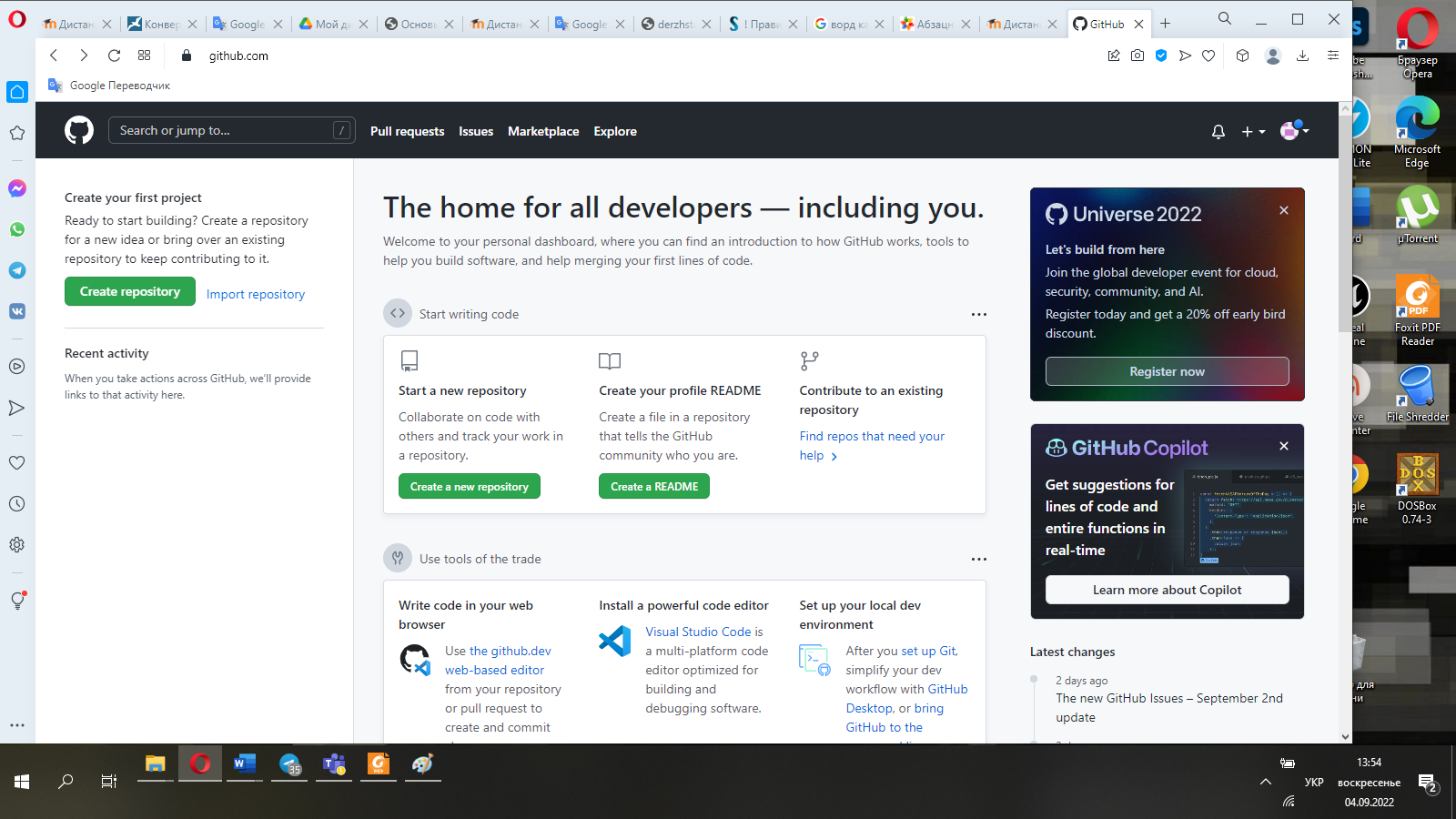


Рисунок 5 – Сторінка “dashboard”

5) Далі було створено новий репозиторій під назвою «Kabachenko-java-labs». На рисунку 6 наведено процес створення репозиторію, а на рисунку 7 – його початковий вміст

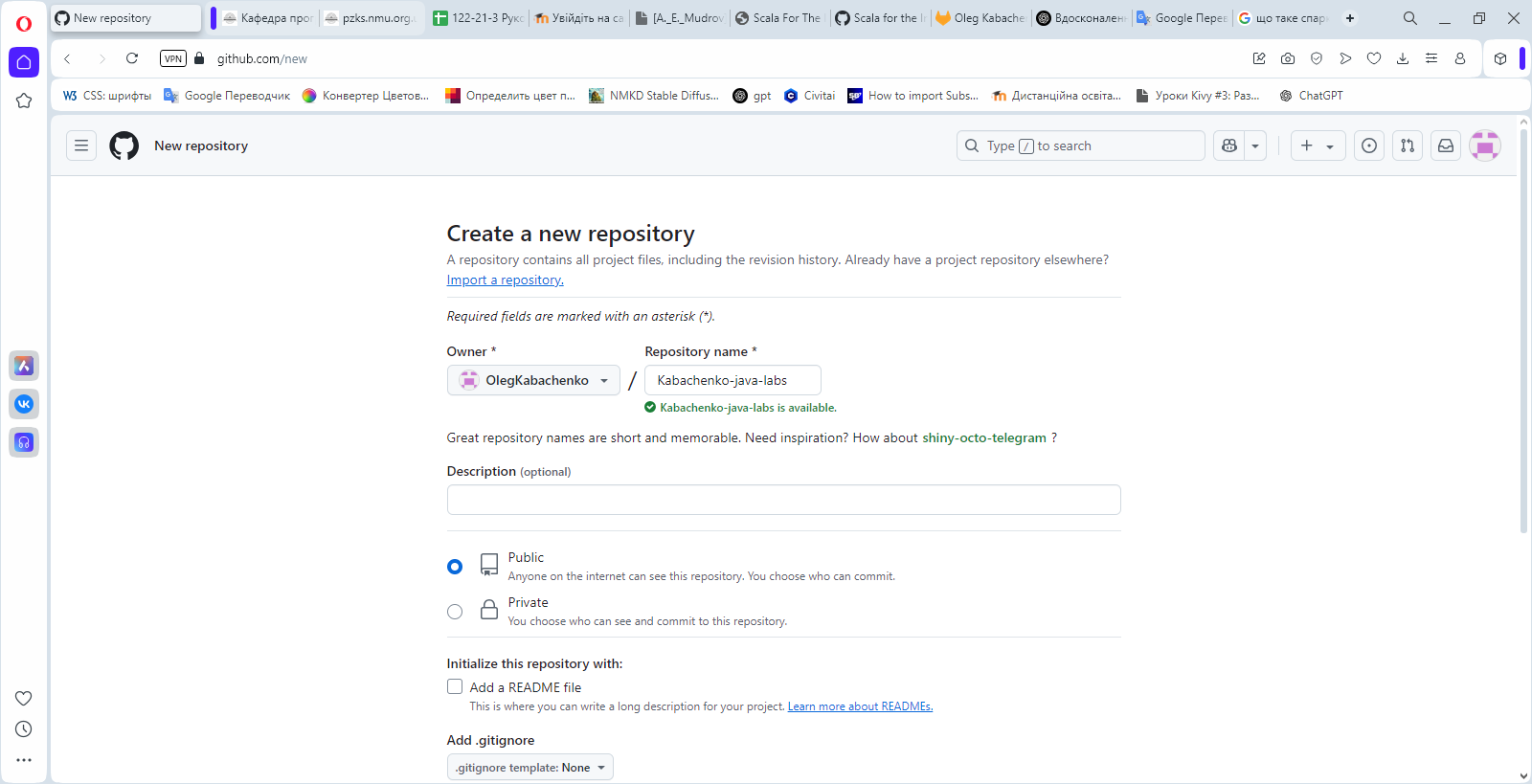


Рисунок 6 – Створення репозиторію

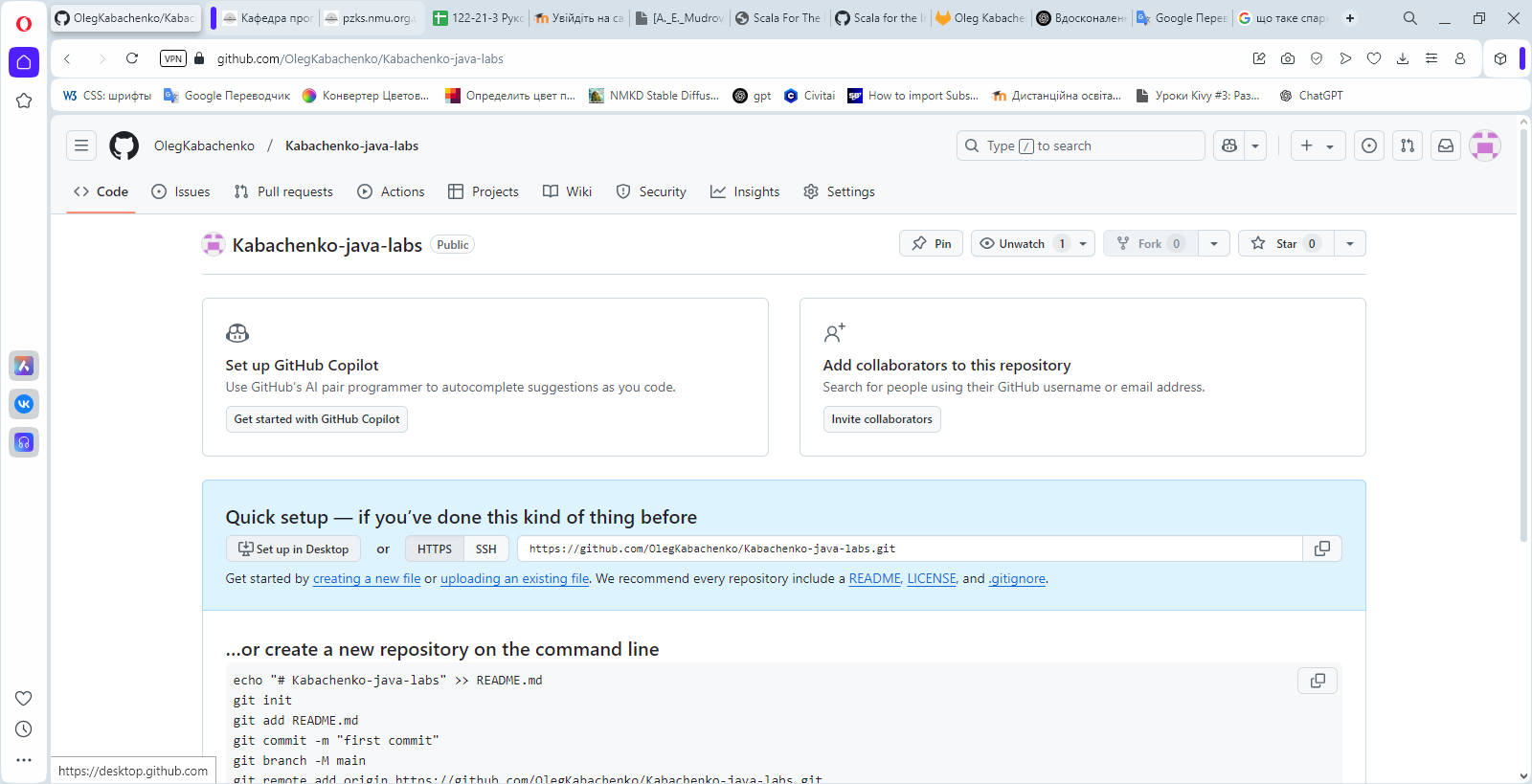


Рисунок 7 – Пустий репозиторій

6) Далі за допомоги Intellij Idea, в корені проекту з попередньої лабораторної роботи ініціалізовано Git-репозиторій, як це зображено на рисунку 8.

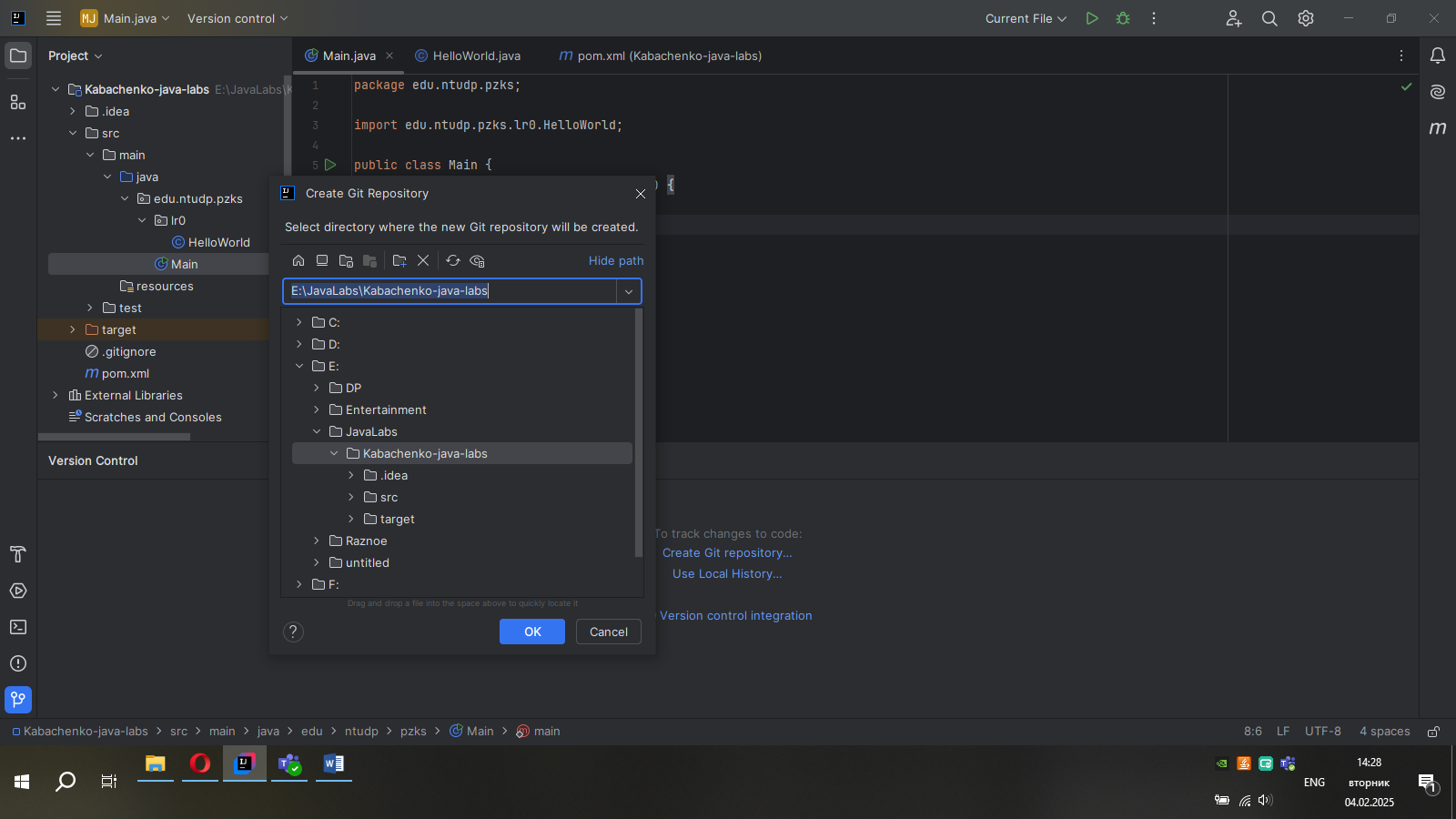


Рисунок 8 – Ініціалізація локального Git-репозиторію

7) На цьому кроці підключимо локальний репозиторій до хмарного, використовуючи термінал та команду git remote add, як це наведено на рисунку 9.

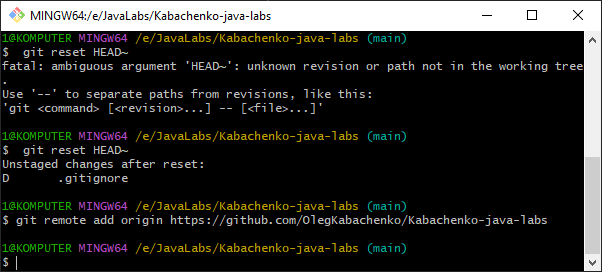


Рисунок 9 – З’єднання локального репозиторію з хмарним

8) Далі знову було відкрито Intellij Idea, та обрано вікно створення комітів, у якому обрано необхідні файли(усі, крім файлів з кодом та файлу звіту з попередньої лабораторної роботи), уведено коментарій та натиснуто «Commit and Push». Цей процес зображено на рисунку 10, а на рисунку 11 можливо переглянути оновлений вміст хмарного репозиторію.

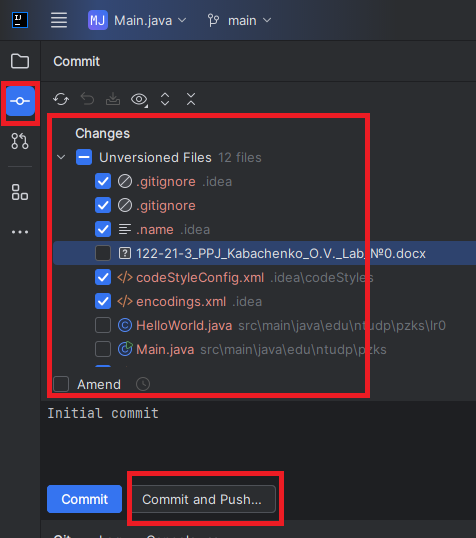


Рисунок 10 – Створення першого коміту

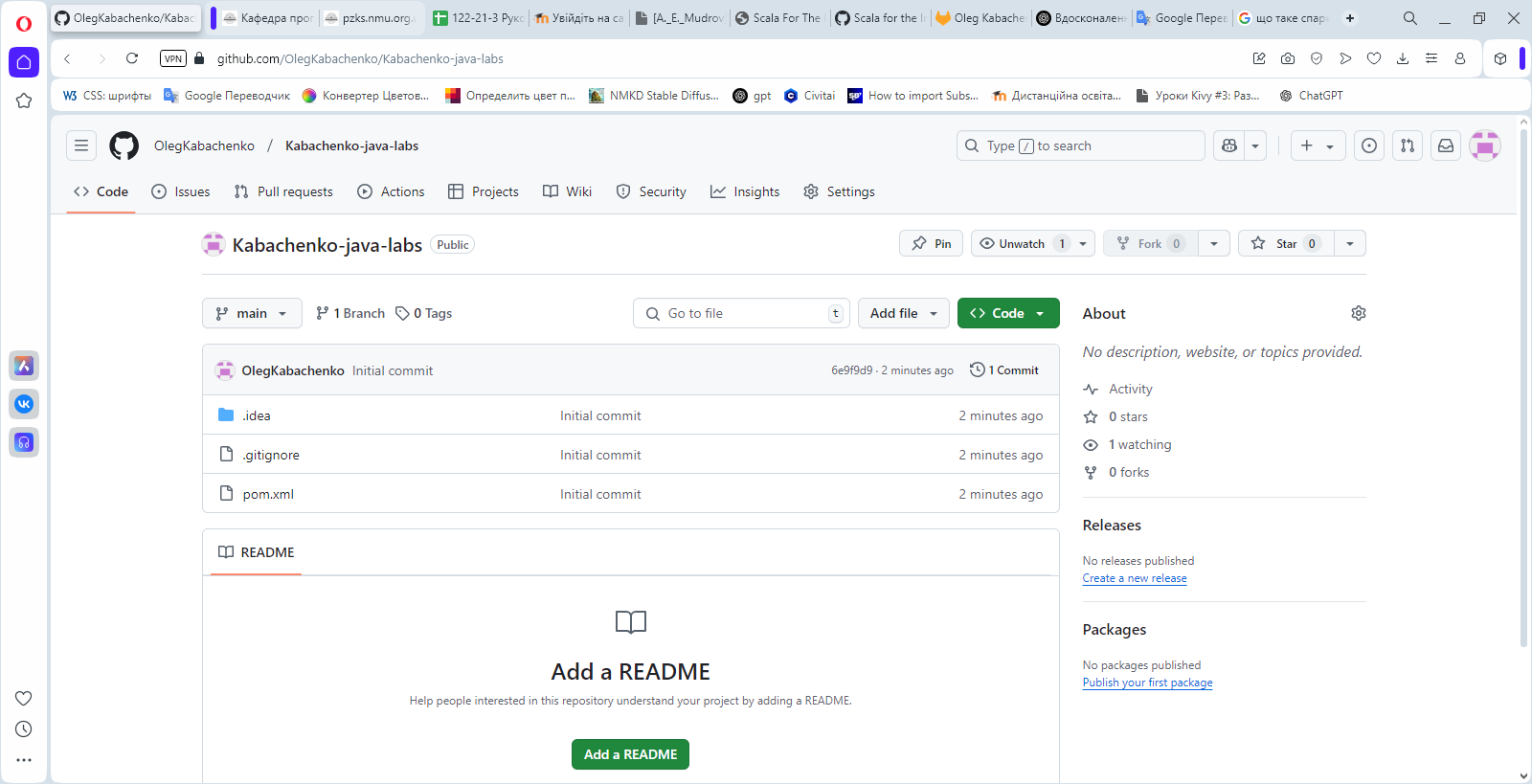


Рисунок 11 – Оновлений вміст хмарного репозиторію.

9) На цьому кроці створено нову гілку проекту, шляхом обрання кнопки, зображеної на рисунку 12, та введення необхідної назви гілки(рисунок 13).

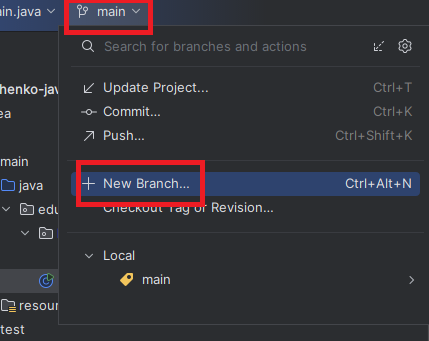


Рисунок 12 – Створення нової гілки

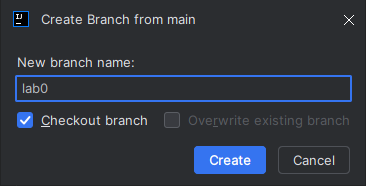


Рисунок 13 – Введення назви для нової гілки

10) Після попереднього кроку створено новий коміт, який додає до нової гілки файли з кодом та файл звіту з попередньої лабораторної роботи, як це наведено на рисунку 14. На рисунку 15 можливо переглянути оновлений вміст хмарного репозиторію.

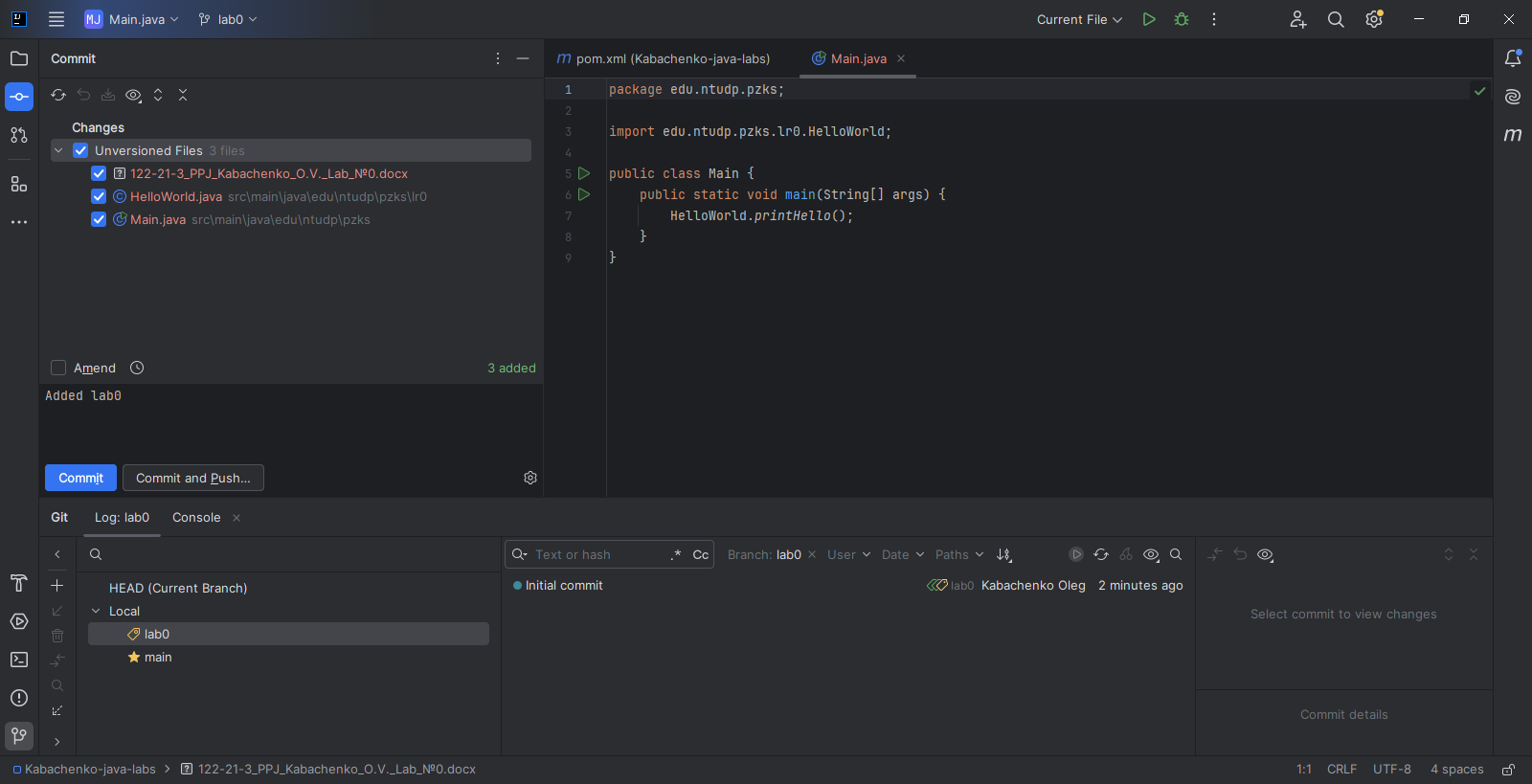


Рисунок 14 – Новий коміт

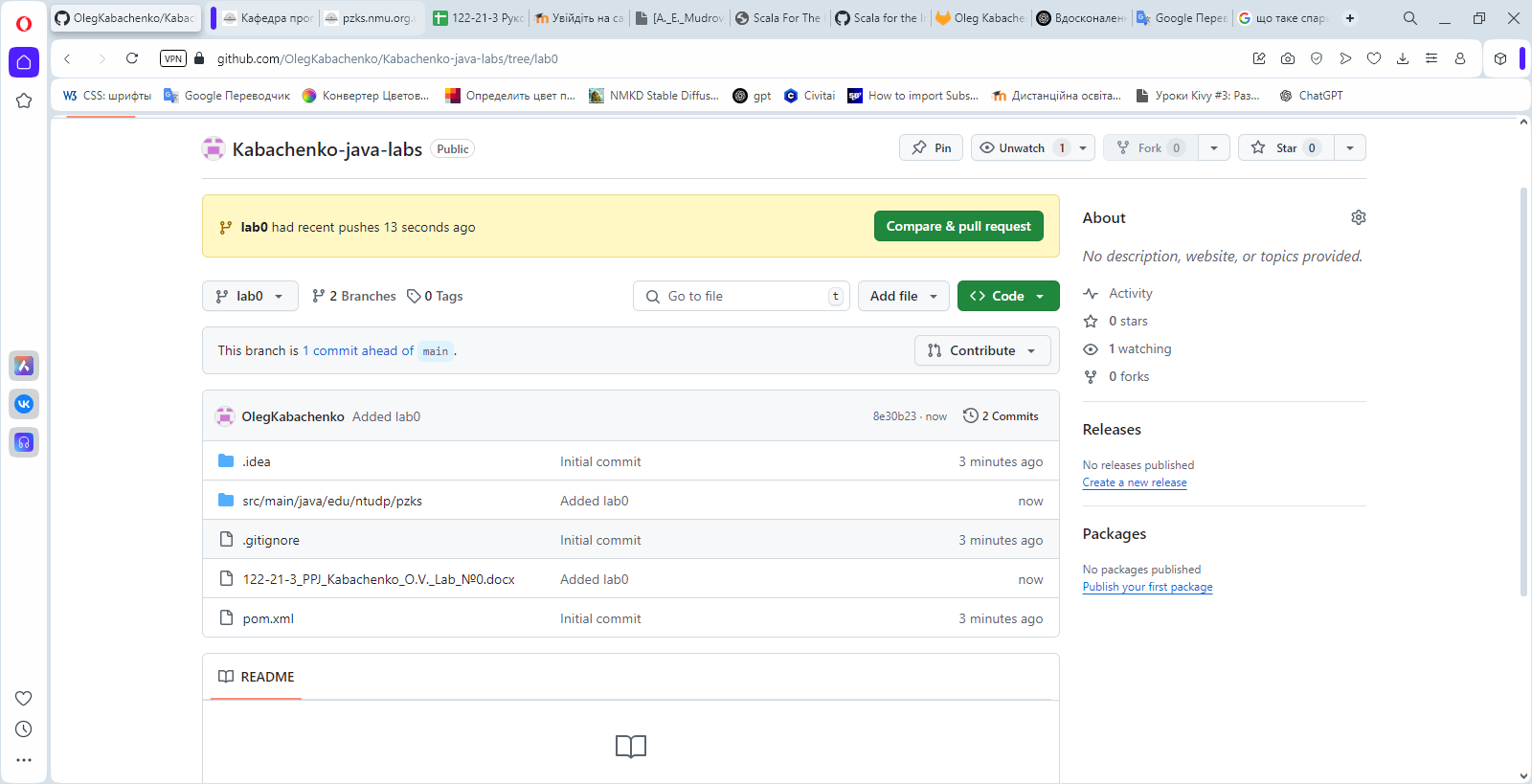


Рисунок 15 – Хмарний репозиторій з новою гілкою та комітом

11) Далі було відкрито головну гілку проекту (рисунок 16) та злито до неї побічну гілку lab0(рисунок 17). Після цих дій, виконано команду Push, для того щоб відправити зміни на хмарний репозиторій(рисунки 18-19)

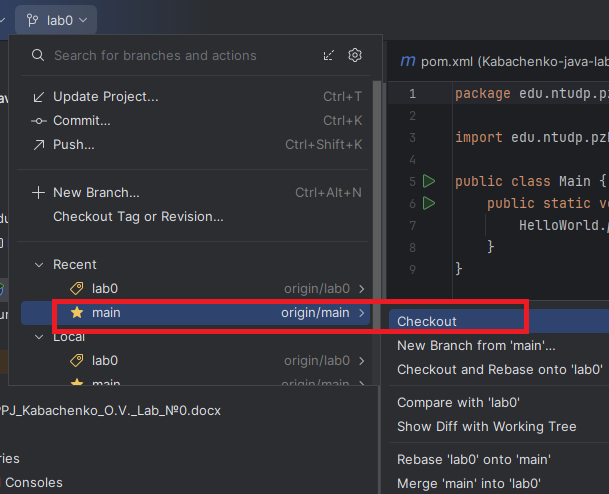


Рисунок 16 – Зміна гілки

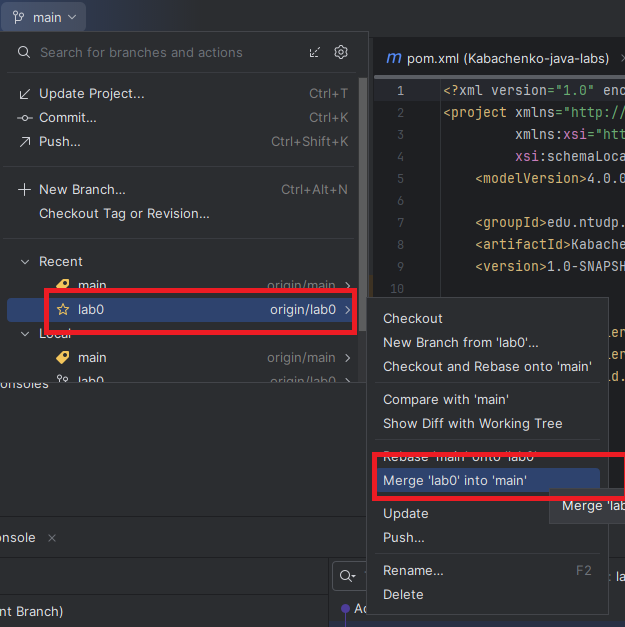


Рисунок 17 – Злиття побічної гілки до головної

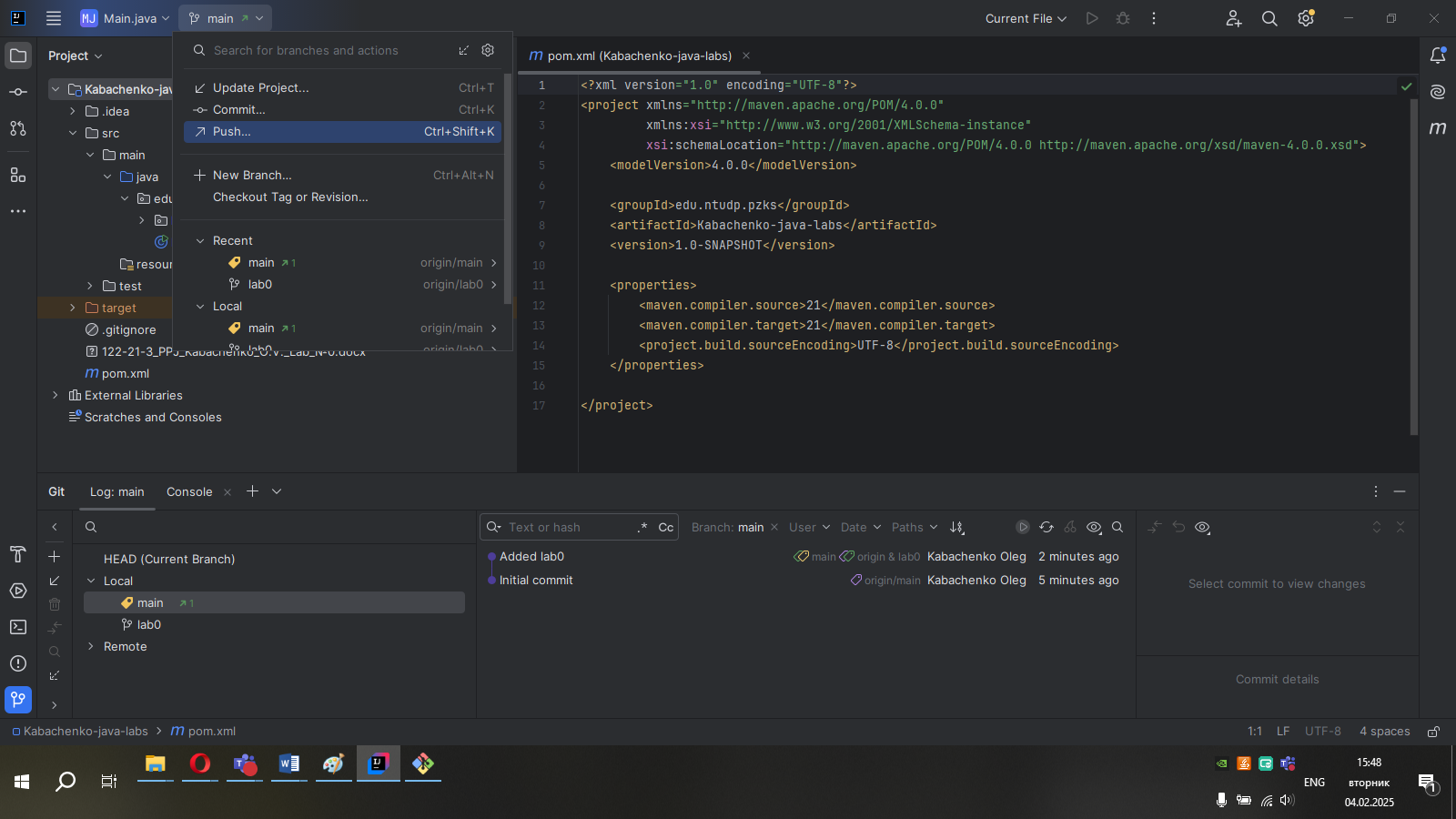


Рисунок 18 – Відправка змін до хмарного репозиторію

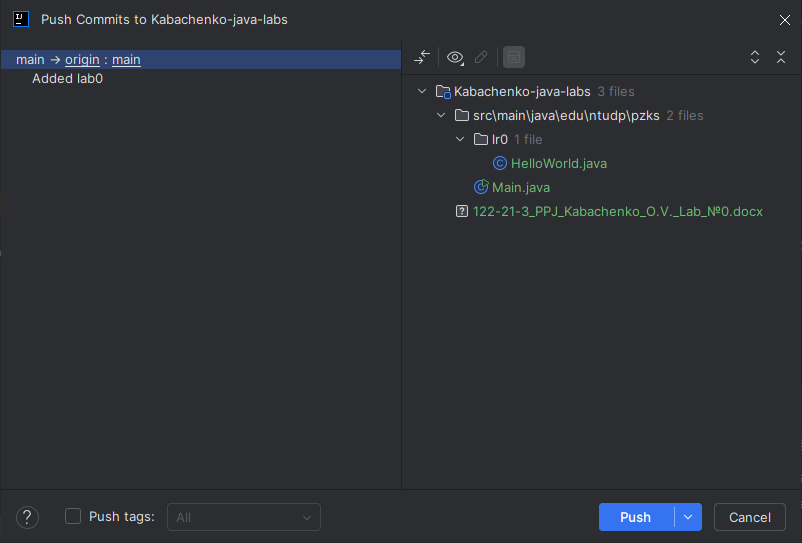


Рисунок 19 – Підтвердження дії

12) Після написання звіту для цієї лабораторної роботи, його аналогічним чином було завантажено спочатку до окремої гілки, а потім цю гілку злито до головної.

**Висновки**

У ході виконання лабораторної роботи було розглянуто створення власного акаунту на github , а також процес під'єднання свого проекту intellij Idea до github. Додатково вивчено порядок дій для створення нових гілок та злиття однієї гілки в іншу.