

**LAPORAN PRAKTIKUM
DASAR PEMROGRAMAN**

JOBSHEET 2

Sistem Version Control dan Kanbar Board

Oleh:

HIKMAH ALDRIN ABDILLAH

NIM: 2341720049



**PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
AGUSTUS 2023**

Praktikum

2.1 Percobaan 1: Menggunakan Github

Pertanyaan

1. Jelaskan perbedaan perintah git commit dan git push?

Jawab:

Git commit digunakan untuk "mengabadikan" perubahan yang telah Anda buat dalam repositori lokal Anda. Ini adalah langkah penting dalam siklus kerja Git.

Ketika Anda menjalankan **git commit**, Anda membuat titik sejarah baru (commit) yang mencatat perubahan-perubahan yang telah Anda buat dalam repositori lokal Anda. Setiap commit memiliki pesan komit yang menjelaskan perubahan yang telah dilakukan.

Git push digunakan untuk mengirim perubahan yang telah Anda commit di repositori lokal Anda ke repositori yang berada di server Git (misalnya, GitHub, GitLab, Bitbucket, dll.).

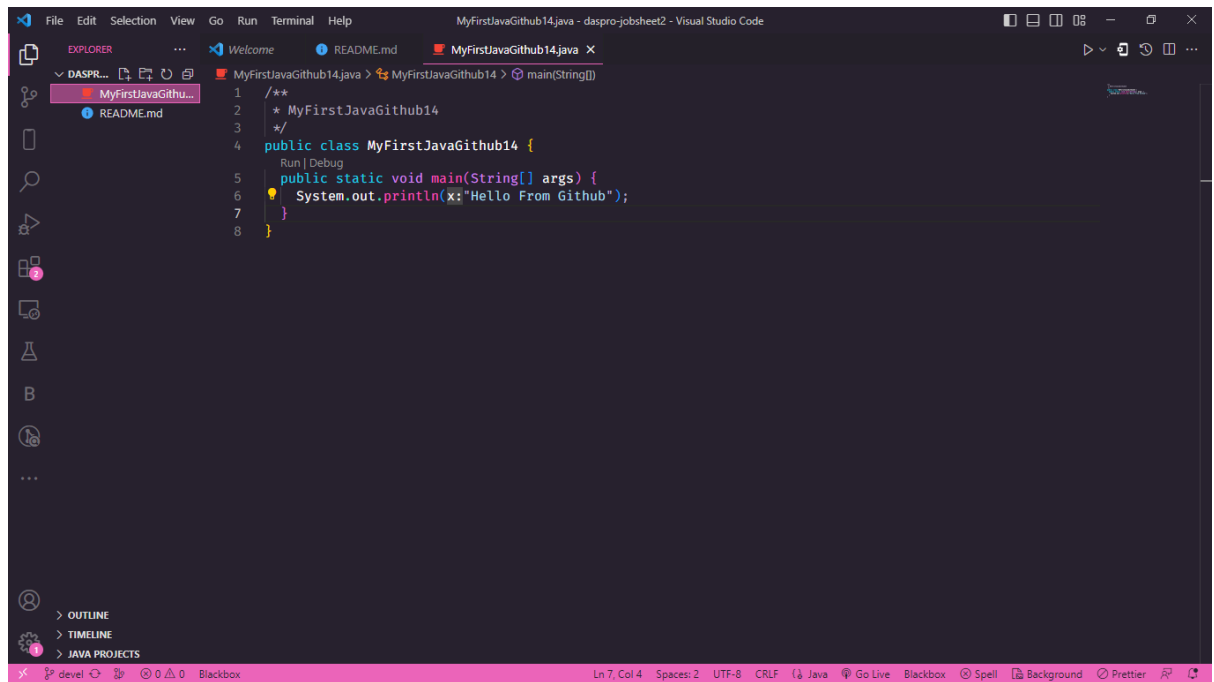
Dengan menjalankan **git push**, Anda memperbarui repositori pusat (atau repositori yang terpusat) dengan perubahan yang telah Anda lakukan di repositori lokal Anda. Ini memungkinkan kolaborator atau anggota tim lainnya untuk melihat dan mengakses perubahan Anda.

2. Apakah bisa alurnya dibalik, membuat folder atau proyek terlebih dahulu kemudian upload (push) ke Github? Jika bisa, buktikan!

Jawab:

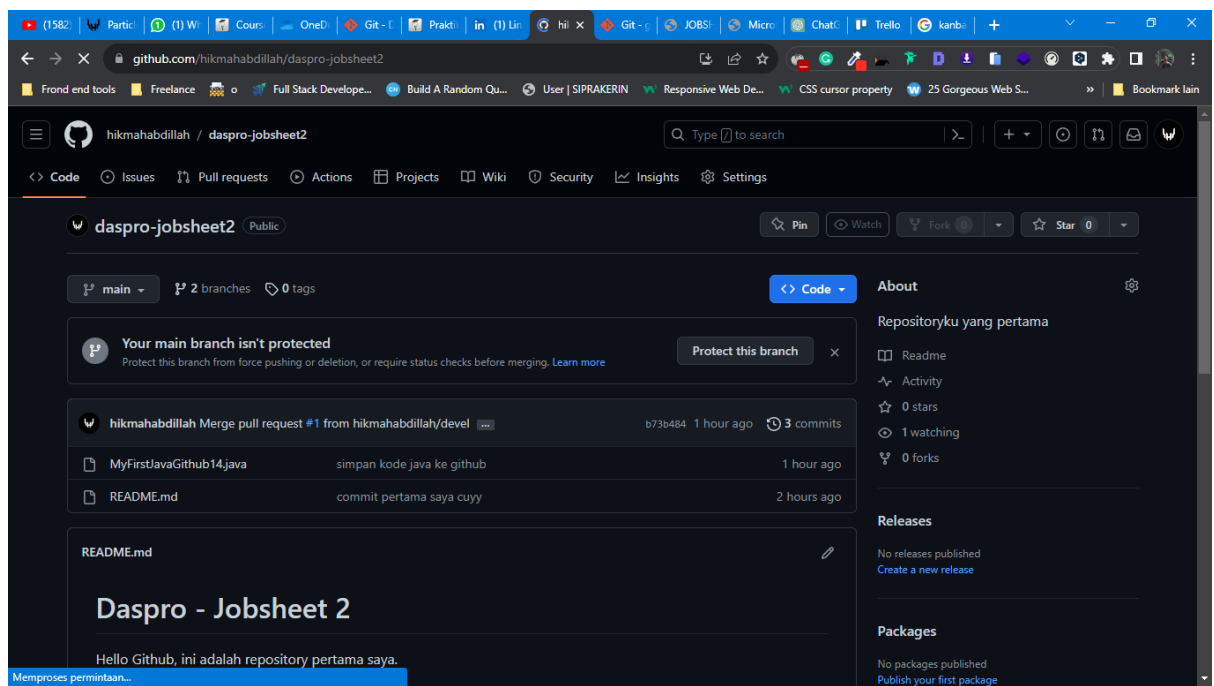
Tentu, Anda bisa membuat folder atau proyek terlebih dahulu di komputer Anda dan kemudian mengunggahnya (push) ke GitHub. Jadi, ya, Anda dapat membuat folder atau proyek terlebih dahulu di komputer Anda dan kemudian mengunggahnya (push) ke GitHub sesuai dengan langkah-langkah di atas. Ini adalah cara umum untuk memulai proyek baru di GitHub.

1. Membuat folder dan file



```
1 /**
2  * MyFirstJavaGithub14
3  */
4  public class MyFirstJavaGithub14 {
5      public static void main(String[] args) {
6          System.out.println("Hello From Github");
7      }
8  }
```

2. masuk ke Github dan membuat repository baru



3. masuk ke git bash dan ketikkan command berikut ini

```

hikmah@DESKTOP-OF0MEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2
$ git clone https://github.com/hikmahabdillah/daspro-jobsheet2.git
Cloning into 'daspro-jobsheet2'...
warning: You appear to have cloned an empty repository.

hikmah@DESKTOP-OF0MEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2
$ dir
daspro-jobsheet2

hikmah@DESKTOP-OF0MEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2
$ cd daspro-jobsheet2

hikmah@DESKTOP-OF0MEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2/daspro-jobsheet2 (main)
$ git add .

hikmah@DESKTOP-OF0MEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2/daspro-jobsheet2 (main)
$ git commit -m "commit pertama saya cuyy"
[main (root-commit) 86ab1d5] commit pertama saya cuyy
1 file changed, 3 insertions(+)
create mode 100644 README.md

hikmah@DESKTOP-OF0MEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2/daspro-jobsheet2 (main)
$ git push origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 293 bytes | 36.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/hikmahabdillah/daspro-jobsheet2.git
 * [new branch]      main -> main

hikmah@DESKTOP-OF0MEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2/daspro-jobsheet2 (main)
$ git config global user email "hikmahaldrin44@gmail.com"
usage: git config [<options>]

Config file location
  --global      use global config file
  --system      use system config file
  --local       use repository config file
  --worktree    use per-worktree config file
  -f, --file <file> use given config file
  --blob <blob-id> read config from given blob object

```

```

hikmah@DESKTOP-OFOMEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2/daspro-jo
bsheet2 (main)
$ git config global user.name "Hikmah Aldrin Abdillah"
usage: git config [<options>]

Config file location
  --global          use global config file
  --system          use system config file
  --local           use repository config file
  --worktree        use per-worktree config file
  -f, --file <file> use given config file
  --blob <blob-id>  read config from given blob object

Action
  --get             get value: name [value-pattern]
  --get-all        get all values: key [value-pattern]
  --get-regexp      get values for regexp: name-regex [value-pattern]
  --get-urlmatch    get value specific for the URL: section[.var] URL
  --replace-all    replace all matching variables: name value [value-pattern]

  --add            add a new variable: name value
  --unset          remove a variable: name [value-pattern]
  --unset-all     remove all matches: name [value-pattern]
  --rename-section rename section: old-name new-name
  --remove-section remove a section: name
  -l, --list       list all
  --fixed-value    use string equality when comparing values to 'value-pattern'
  -e, --edit       open an editor
  --get-color      find the color configured: slot [default]

```

```

hikmah@DESKTOP-OFOMEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2/daspro-jo
bsheet2 (main)
$ git branch devel

hikmah@DESKTOP-OFOMEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2/daspro-jo
bsheet2 (main)
$ git checout devel
git: 'checout' is not a git command. See 'git --help'.

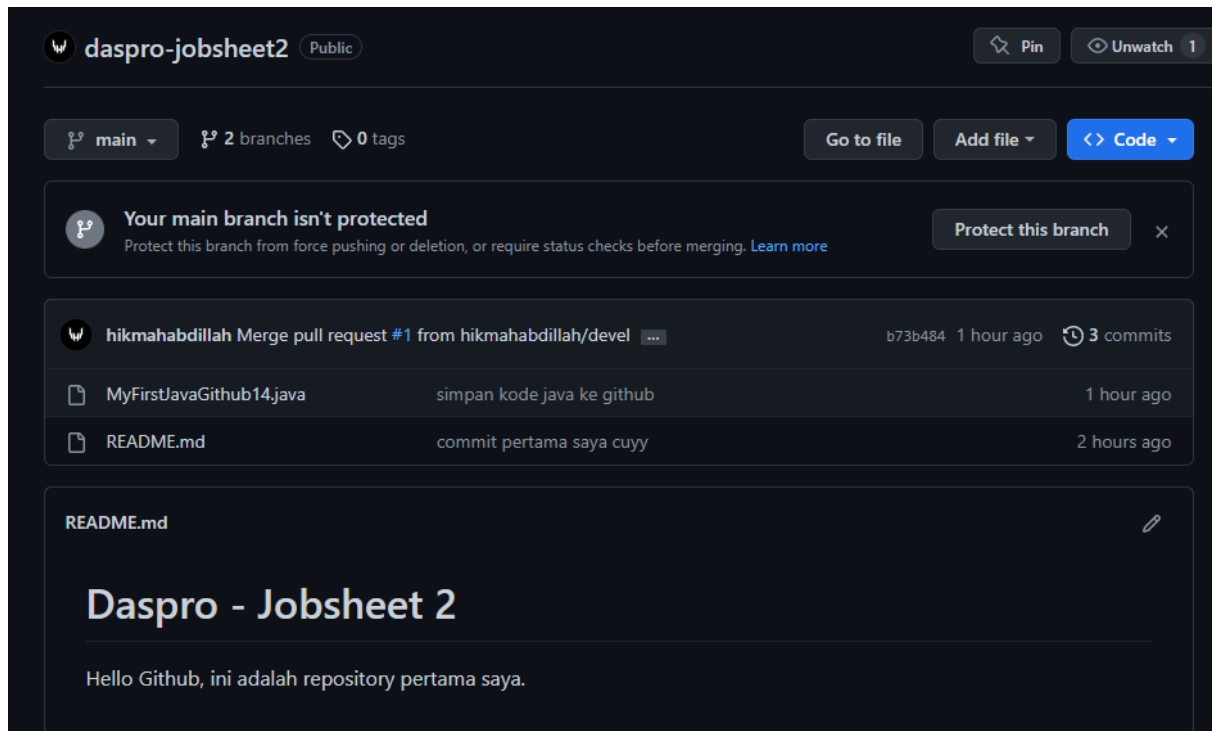
The most similar command is
  checkout

hikmah@DESKTOP-OFOMEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2/daspro-jo
bsheet2 (main)
$ git checkout devel
Switched to branch 'devel'

hikmah@DESKTOP-OFOMEVA MINGW64 ~/OneDrive/Dokumen/TUGAS DASPRO S1/Pt 2/daspro-jo
bsheet2 (devel)
$ |

```

4. kembali ke Github dan refresh browser



2.2 Percobaan 2 : Dasar Kolaborasi di Github

Pertanyaan!

1. Jelaskan fungsi dari Pull requests!

Jawab:

Pull request (sering disingkat menjadi PR) adalah fitur yang penting dalam sistem kontrol versi berbasis Git, seperti GitHub, GitLab, dan Bitbucket. Fungsi utama dari pull request adalah untuk memfasilitasi kolaborasi dan penggabungan perubahan antara cabang (branch) yang berbeda dalam sebuah repositori.

2. Mengapa kita perlu membuat sebuah branch, manfaatnya apa?

Jawab:

Membuat cabang (branch) dalam repositori Git memiliki banyak manfaat, terutama dalam konteks pengembangan perangkat lunak kolaboratif. Dengan kata lain, membuat cabang merupakan praktik yang sangat baik dalam pengelolaan kode sumber dan memungkinkan pengembangan perangkat lunak yang lebih terstruktur, terorganisir, dan kolaboratif. Cabang memfasilitasi kerja tim, pengujian, pemeliharaan kode, dan inovasi dalam proyek perangkat lunak.

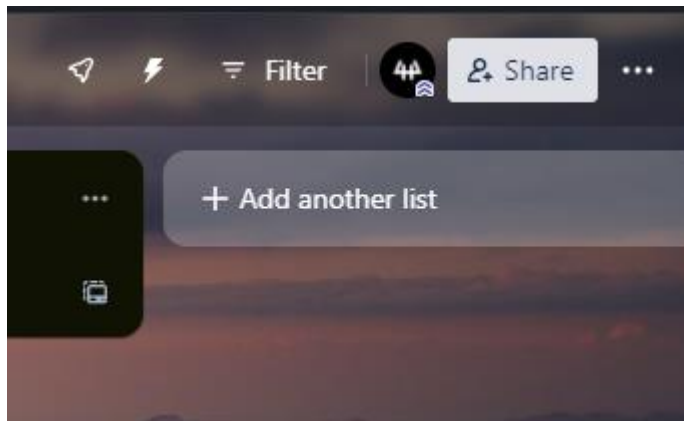
2.3 Percobaan 3 : Menggunakan Trello

Pertanyaan!

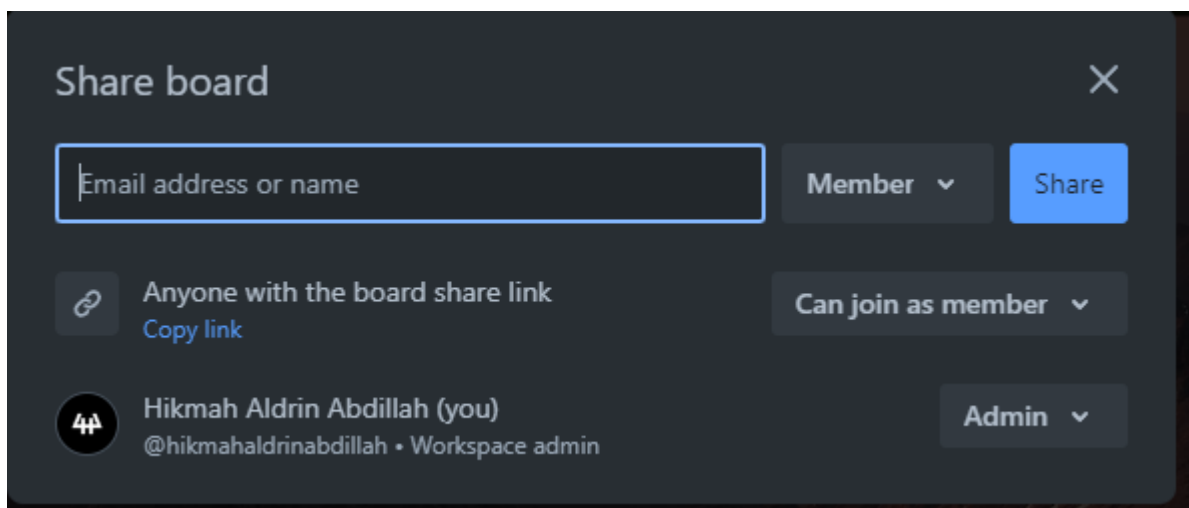
1. Bagaimana langkah yang dilakukan untuk mengundang anggota tim untuk bergabung ke dalam board Anda?

Jawab:

Klik menu share



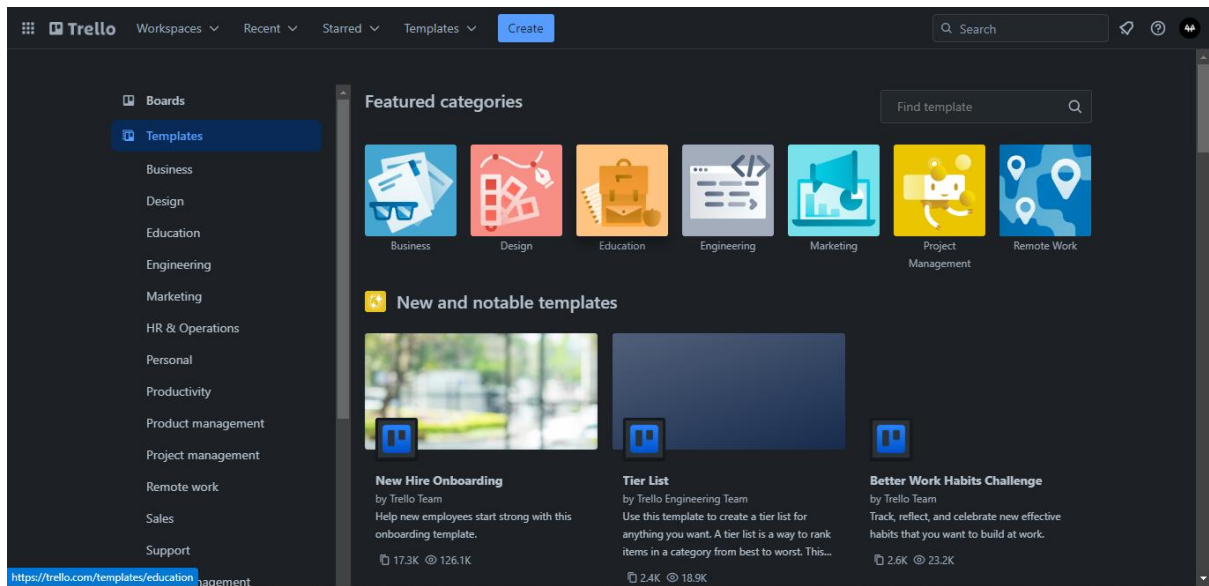
Dan disini kita bisa langsung invite via gmail atau copy link untuk mengundang anggota tim



2. Apakah memungkinkan untuk membuat Board tidak dari awal (menambahkan list satu per satu), jika mungkin bagaimana caranya?

Jawab:

Pada Trello, Anda dapat membuat board dengan lebih cepat dan mudah menggunakan fitur yang disebut "Template". Dengan template, Anda dapat membuat board yang sudah memiliki beberapa list dan kartu dasar yang dapat disesuaikan sesuai kebutuhan Anda. Sekarang Anda telah membuat board Trello dengan template yang sudah ada dan dapat memulai bekerja dengan lebih cepat. Anda dapat menyesuaikan board tersebut sesuai dengan proyek atau tugas Anda tanpa harus membuat list dan kartu dari awal.



1. Tugas

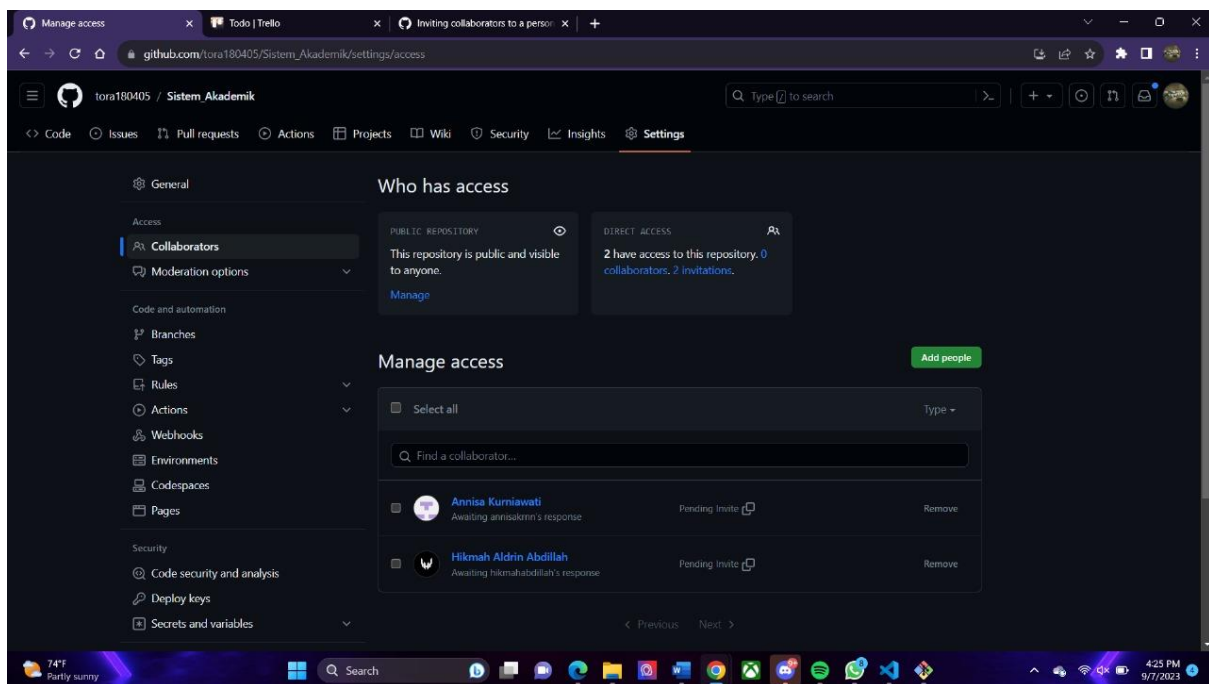
1. Buatlah kelompok yang terdiri dari 3 orang!

Jawab:

Sudah

2. Buatlah repository berdasarkan topik yang dipilih oleh kelompok Anda, undang semua anggota tim ke dalam repository tersebut.

Jawab:



3. Buatlah sebuah Kanban Board dengan judul board lengkap dengan card sesuai dengan topik proyek yang kelompok Anda pilih, undang semua anggota tim ke dalam board tersebut.

