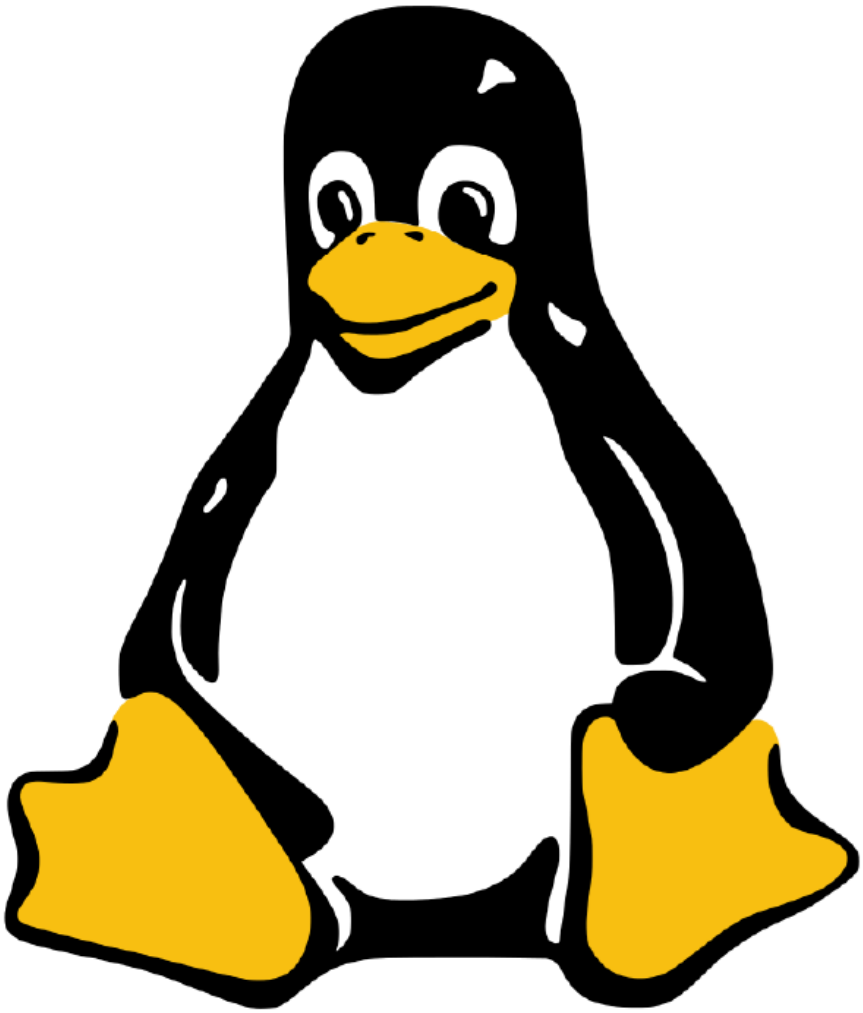
MANUAL MODUL 2
SISTEM OPERASI E
"Github, String, Fork, Thread"



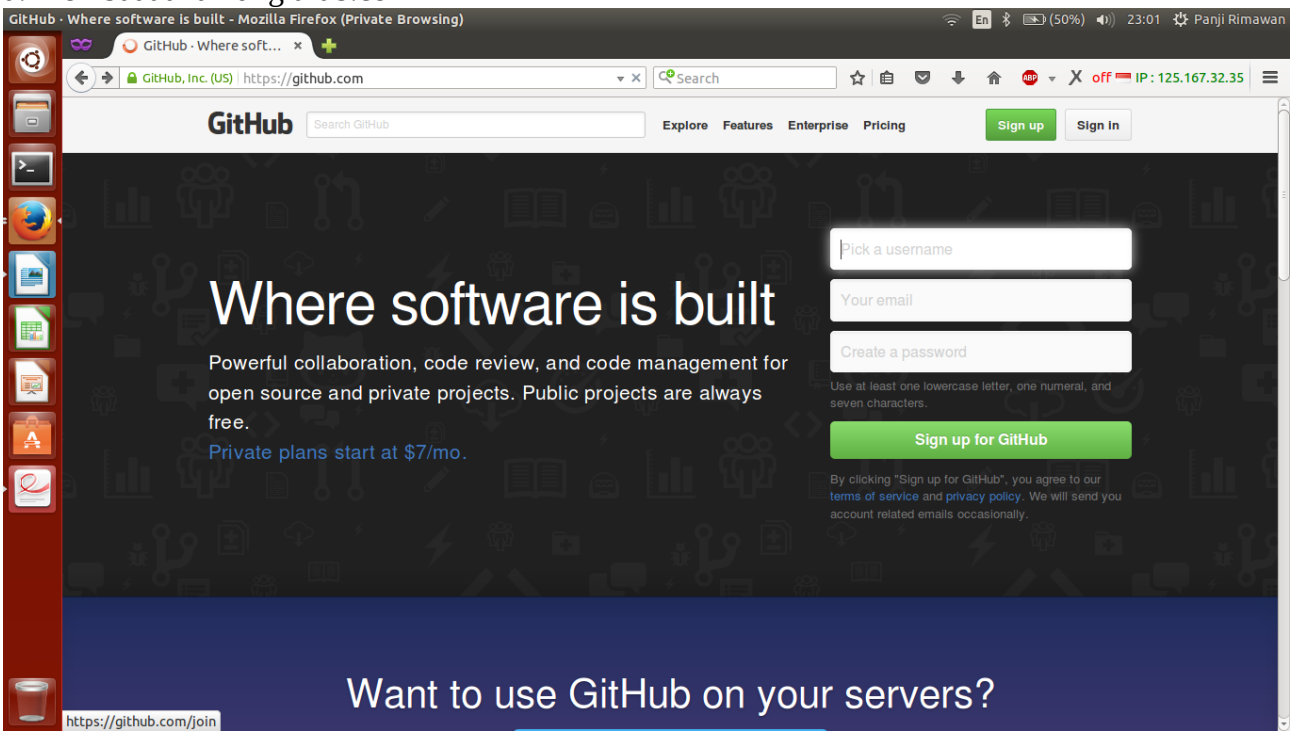
Oleh:
Kelompok E-5
Panji Rimawan (5114100075)
Steven Kurniawan (5114100100)

1. GIT

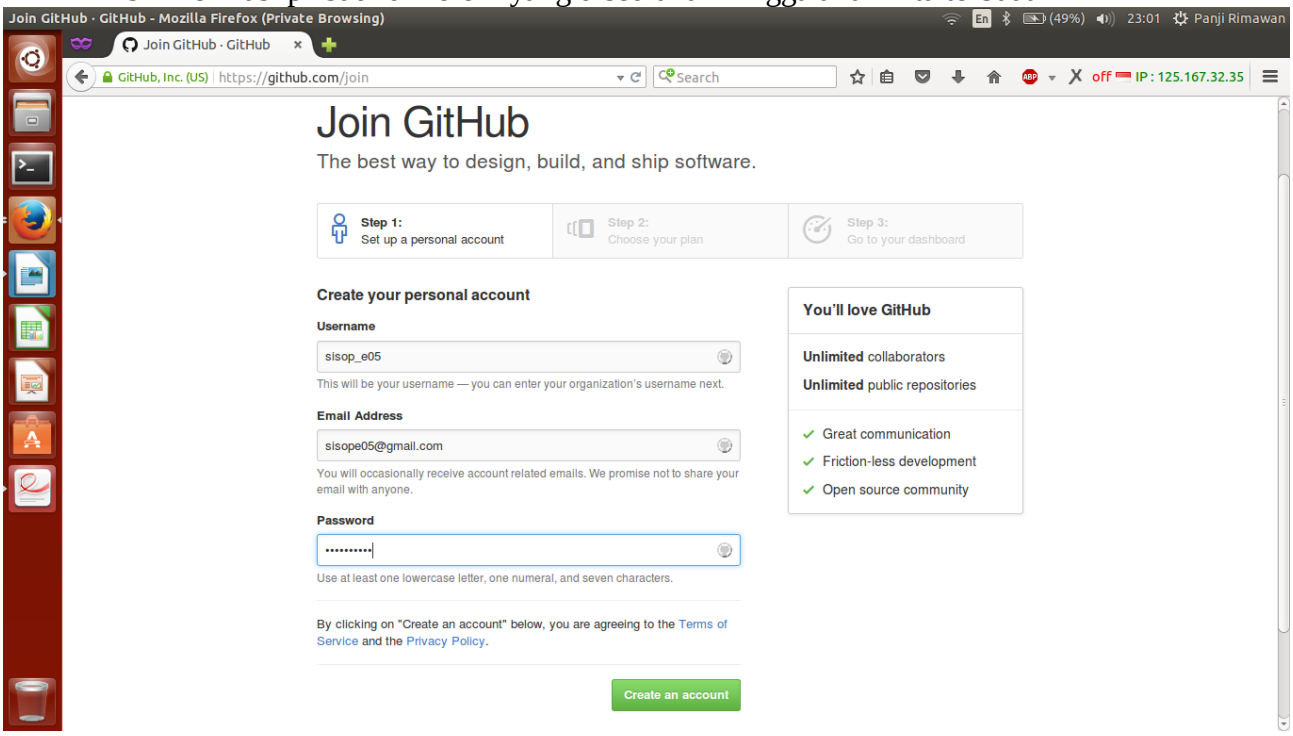
a. Pengertian GitHub?

GitHub adalah layanan penginangan web bersama untuk proyek pengembangan perangkat lunak yang menggunakan sistem pengontrol versi Git. Situs ini menyediakan fungsionalitas jejaring sosial seperti umpan web, pengikut, wiki (menggunakan perangkat lunak Wiki gollum) dan grafik jejaring sosial untuk menampilkan bagaimana para pengembang menggarap versi repositori mereka. (*source: wikipedia.org*)

a. Membuat akun di github.com

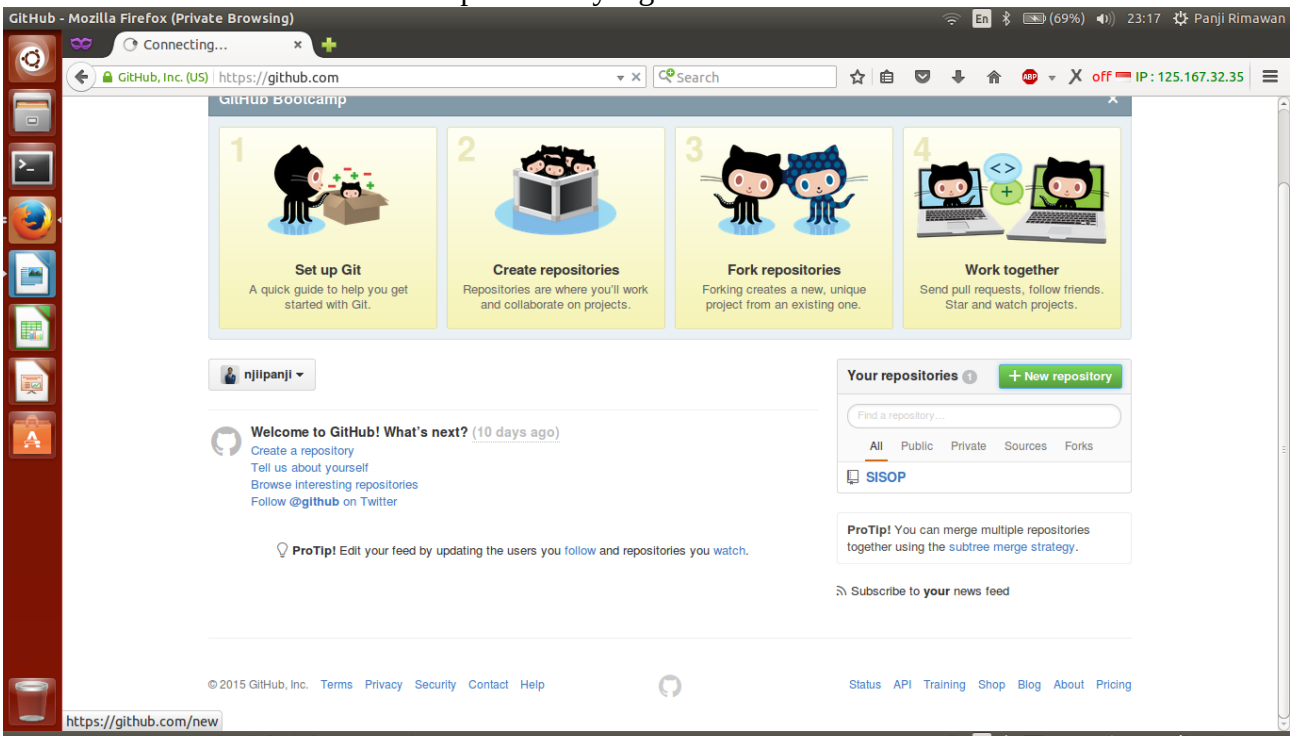


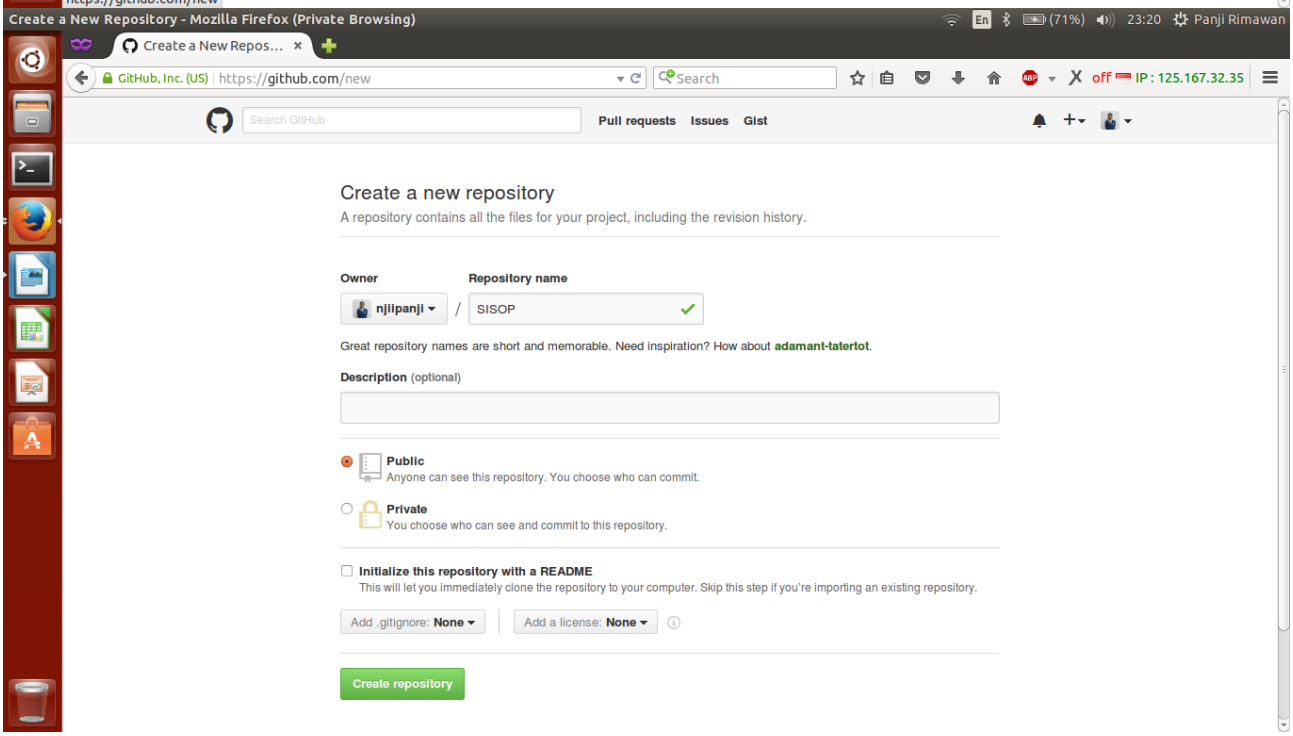
- Buka link github.com
- Pilih register untuk membuat akun
- Isi informasi pribadi di kolom yang disediakan hingga akun kita terbuat



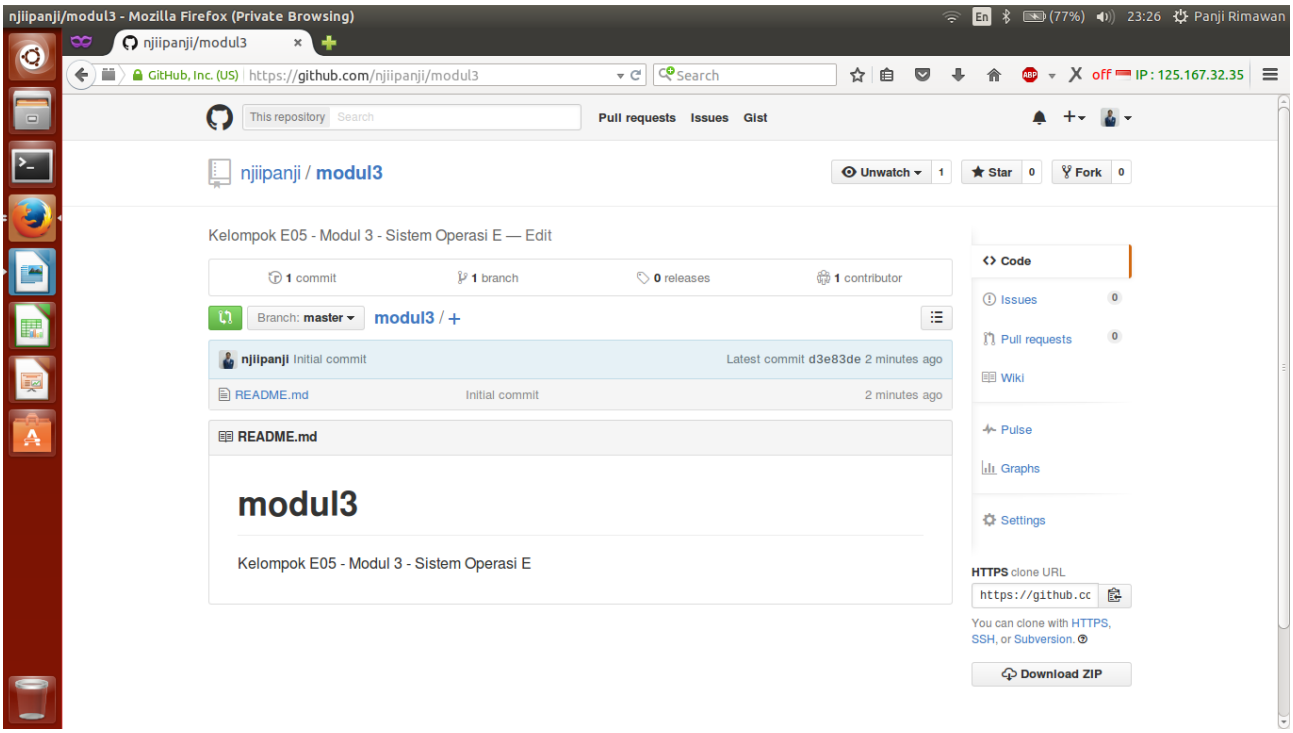
b. Membuat repositories baru di akun github

- Klik tombol “New Repository”, lalu akan muncul halaman baru yang berisikan permintaan untuk memasukkan nama repositories yang akan dibuat



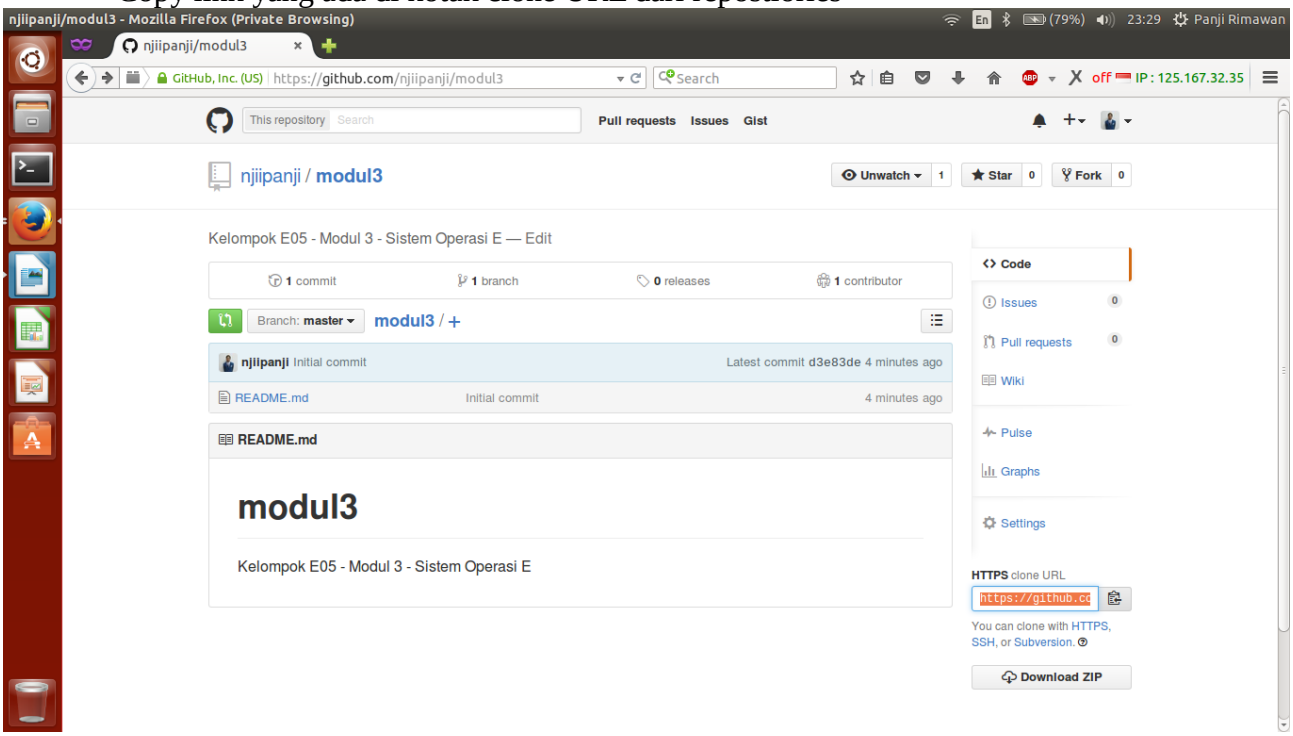


- Setelah selesai memberi nama, langkah terakhir yaitu klik tombol “Create Repository”. Maka akan muncul halaman seperti di bawah ini.

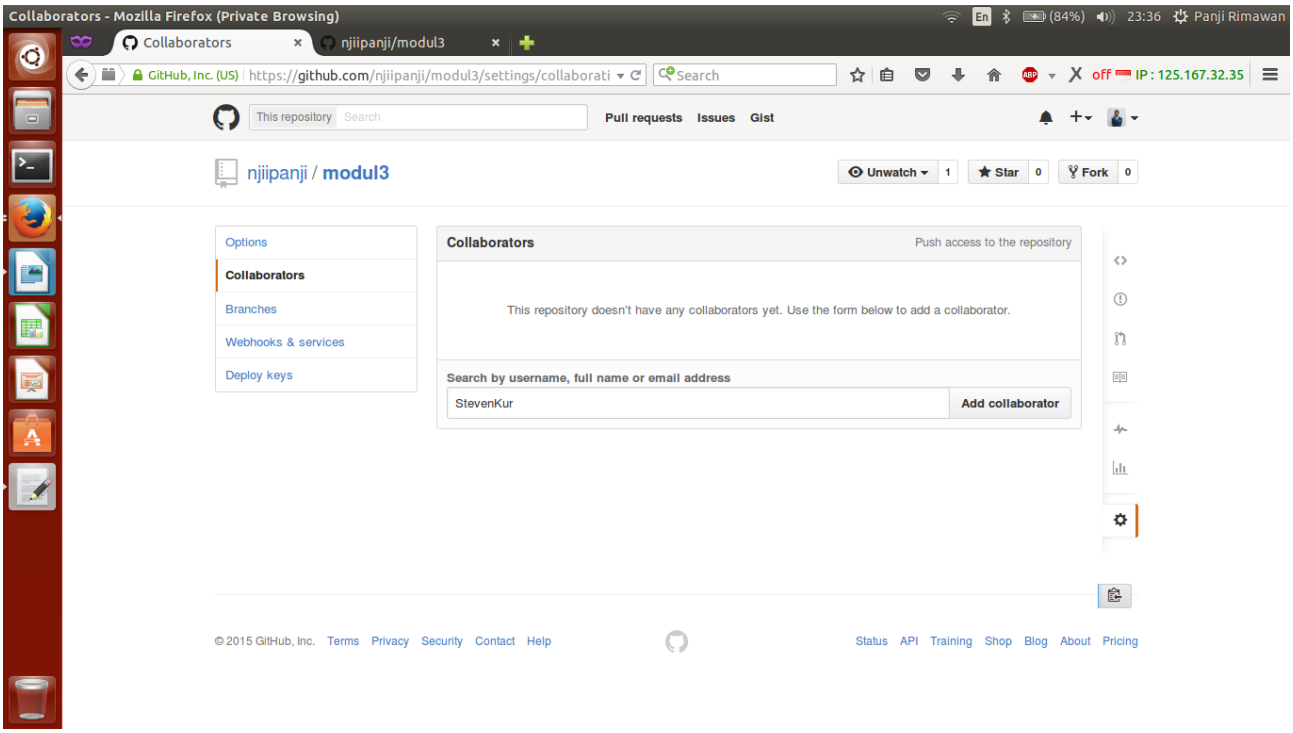


c. Menambahkan collaborator

- Copy link yang ada di kotak clone URL dari repostiories



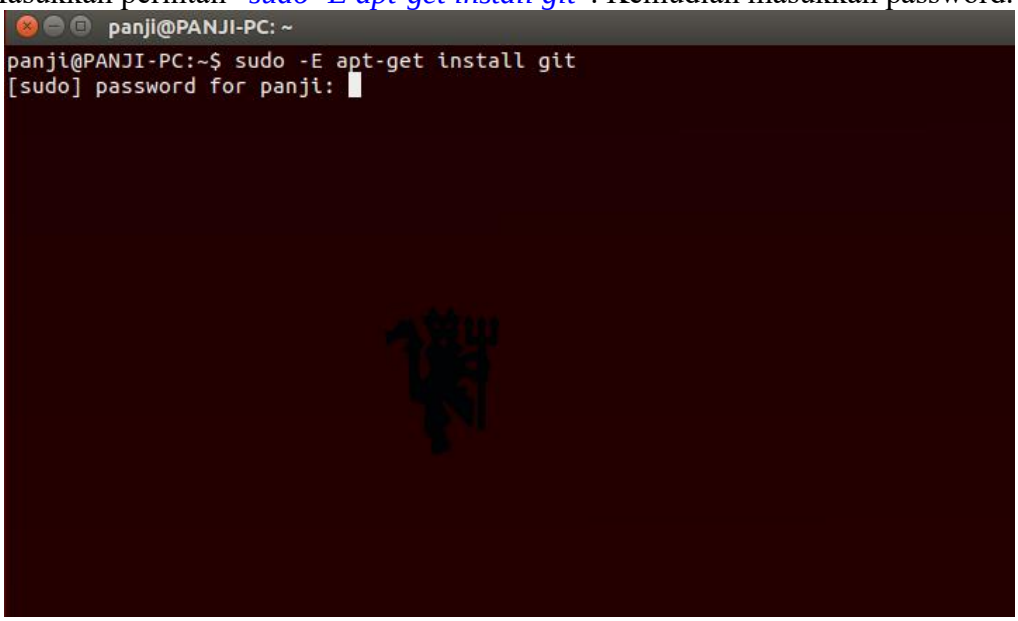
- Klik settings di repositories, lalu pindah ke tab Collaborator
- Masukkan username dari teman/member lain yang 1 project, kemudian klik “Add Collaborator”



- Akan muncul username dari teman/member lain yang kita tambahkan barusan

d. Install git di linux melalui terminal

- Masukkan perintah "*sudo -E apt-get install git*". Kemudian masukkan password.



- Cek versi git yang telah terinstall di ubuntu dengan mengetikkan perintah "*git --version*"

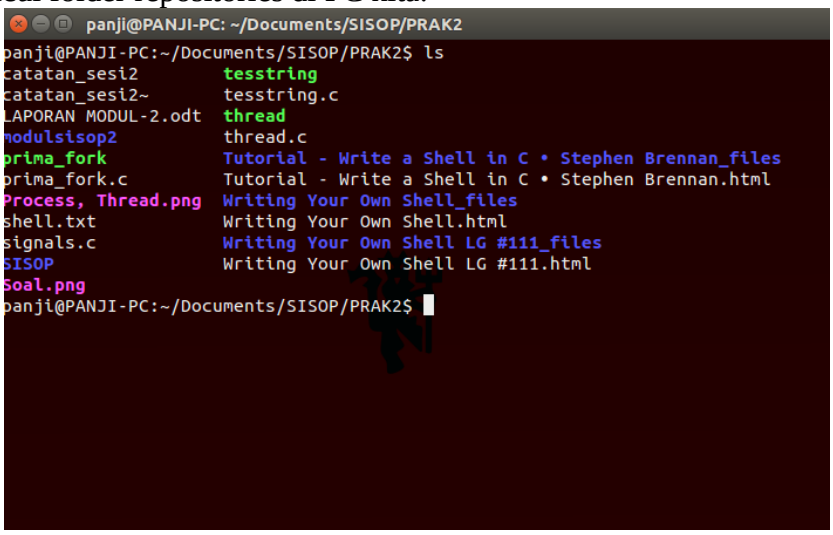
e. Perintah git yang dapat dieksekusi melalui terminal

- git clone *alamat_folder_repositories* -> membuat clone dari repositories ke PC user.
- git add *namafile* -> menambahkan file untuk ditrack .
- git commit -a -> menyetujui adanya perubahan dari repositories.
- git push -> upload file-file terbaru ke repositories .
- git pull -> download file-file yang terbaru dari repositories.

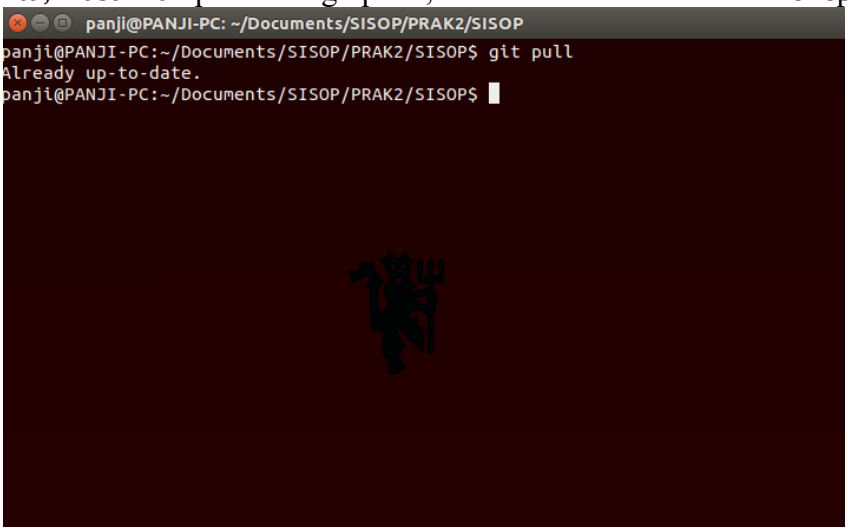
f. Implementasi penggunaan git di linux

1. Buat/masuk ke folder dimana kita akan menyimpan/mengunduh repositories kita ke dalam PC
2. Ketikkan “clone *alamat_folder_repositories*” repositories yang akan kita download ke PC. Folder clone nantinya akan otomatis menjadi folder repositories yang ada di PC. Link alamat untuk clone bisa dicopy dari repositories di github.com. Hal ini hanya dilakukan

Akan muncul folder repositories di PC kita:

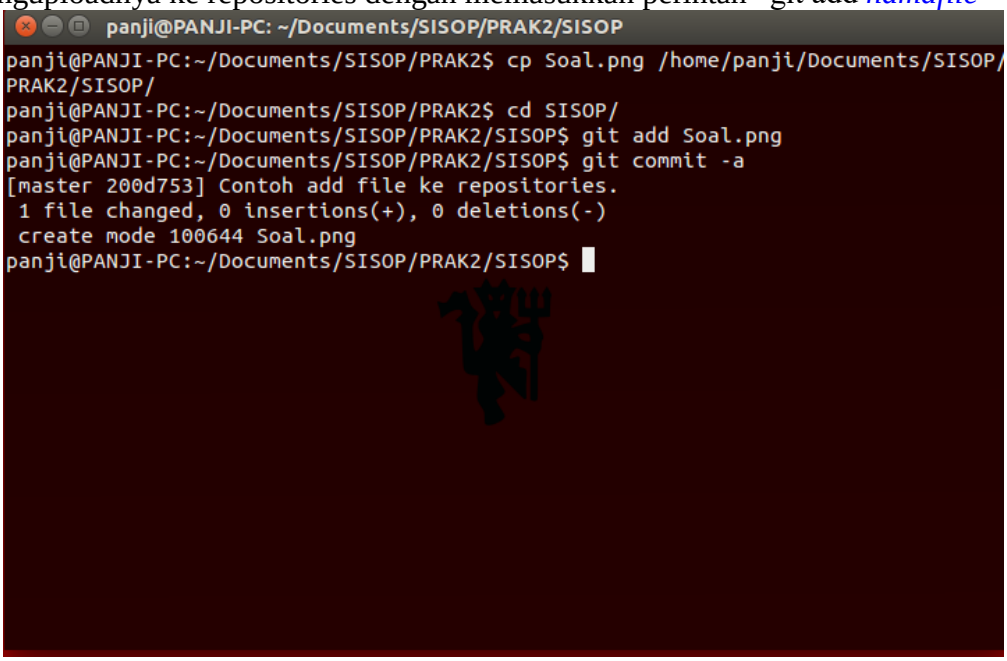


3. Setelah itu, masukkan perintah “git pull”, untuk mendownload file-file repositories terbaru.

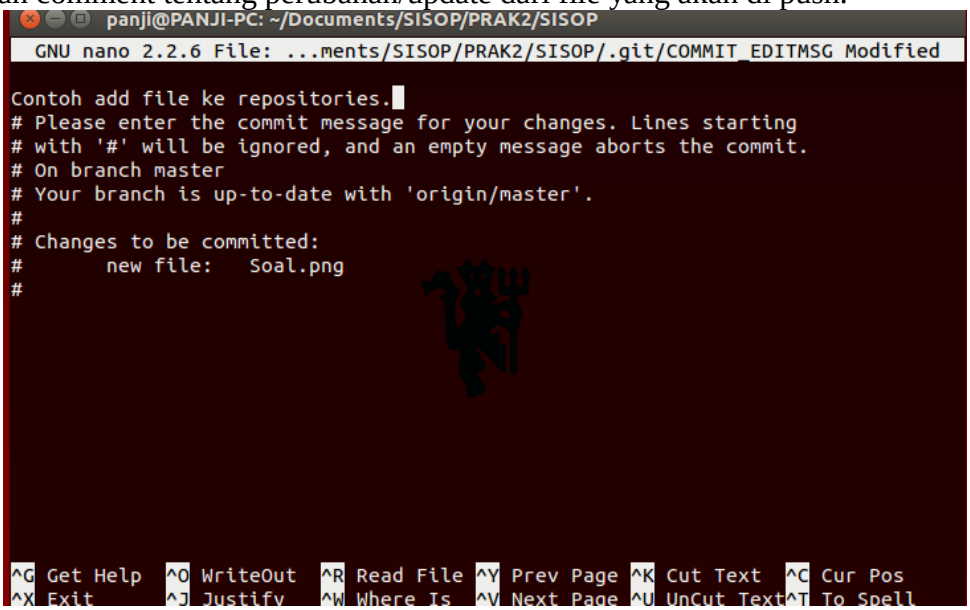


4. Jika sudah mengambil file-file terbaru, maka kita bisa memulai edit file / add file ke dalam folder tersebut untuk ditambahkan/diubah isi repositories. Jangan sampai member dari project yang sama di repositories tersebut mengedit sebuah file yang sama dalam waktu yang bersamaan (harus berbeda).

5. Setelah selesai menambahkan file / mengubah isi suatu file, kita bisa langsung menguploadnya ke repositories dengan memasukkan perintah "git add *namafile*"

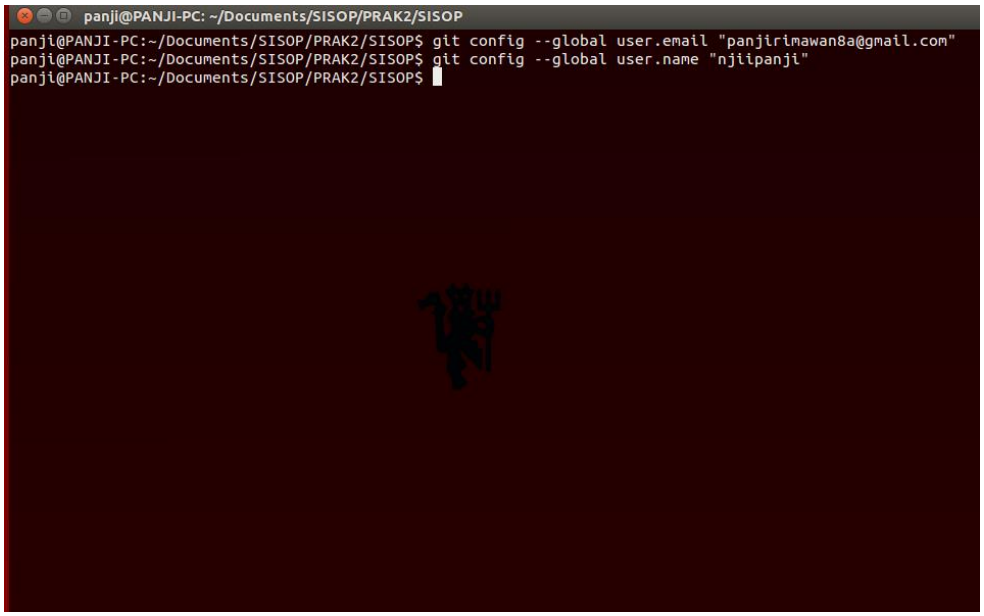


6. Kemudian dilanjutkan dengan memasukkan perintah "git commit -a" untuk membuat sebuah comment tentang perubahan/update dari file yang akan di push.

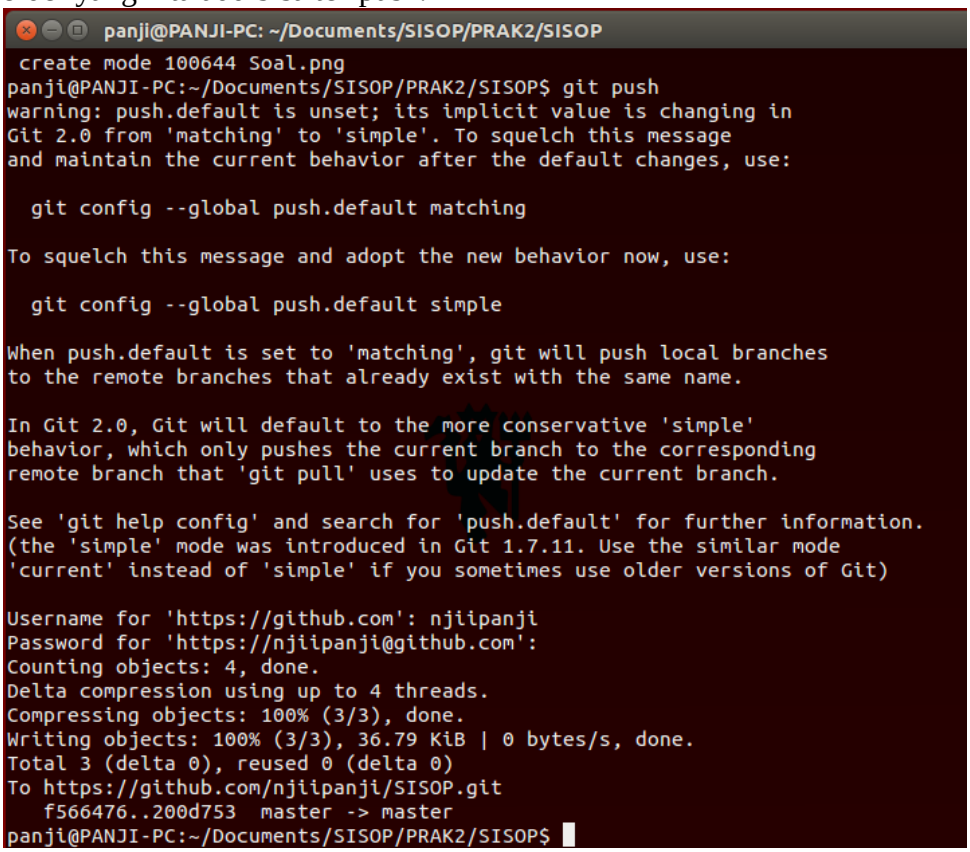


7. Sebelum di push, masukkan terlebih dahulu username.email &username.git sebelum push ke git, dengan memasukkan perintah

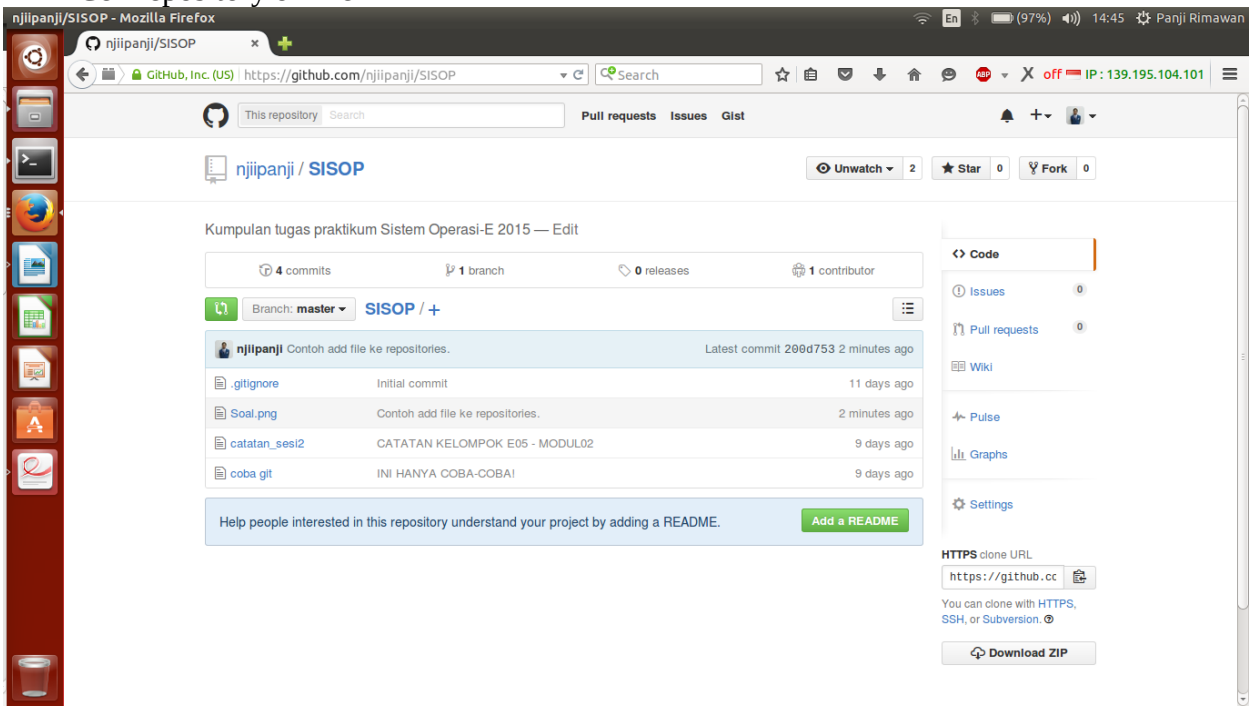
git config --global user.email "*email yang terdaftar di github/collaborator*"
git config --global user.name "*usernamegit*"



8. Setelah selesai, langkah terakhir yaitu push / upload file-file yang sudah ditambahkan di folder, dengan memasukkan perintah "git push". Saat proses, akan muncul tampilan seperti di bawah, dan kita perlu untuk memasukkan **username github** beserta **password** nya, agar file/folder yang kita add bisa ter-push.



Cek repository online



2. THREAD

a. Deklarasi fungsi pthread

Jika ingin menggunakan thread dalam program, perlu include library “pthread.h”. Setelah itu deklarasi fungsi thread di dalam program yang syntax nya adalah:

namafungsi(*variabel pthread*, NULL, *nama fungsi yang ingin dijalankan*, NULL);

contoh:

pthread_t tid[2]; // inisialisasi array untuk menampung thread

pthread_create(&(tid[0]), NULL, &playAndCount, NULL);

pthread_join(*variable pthread*, NULL) → Perintah untuk menunggu thread yang lain selesai terlebih dahulu sebelum program mati

contoh implementasi program:

panji@PANJI-PC: ~/Documents/SISOP/PRAK2/modulsisop2

```
#include <stdio.h>
#include <string.h>
#include <pthread.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>

// install vlc terlebih dahulu
// sesuaikan path file lagu sesuai folder masing2
// compile

pthread_t tid[2];       // inisialisasi array untuk menampung thread (dalam kasus ini terdapat 2 thread)
int length=20;          // inisialisasi jumlah counter

void* playAndCount(void *arg) { // fungsi yang akan dijalankan sbg thread
        unsigned long i=0;
        pthread_t id=pthread_self();
        int iter;
        if(pthread_equal(id, tid[0])) {
                system("clear");
                // printf("\n First thread processing\n");
                for(iter=length; iter>0; iter--) {
                        printf("%d", iter);
                        fflush(stdout);
                        sleep(1);
                        system("clear");
                }
        }
        else if(pthread_equal(id, tid[1])) {
                printf("jalan!\n");
                system("cvlc --play-and-exit --quite coba.mp3");
        }
        return NULL;
```

```
int main (void)
{
        int i=0;
        int err;
        while(i<2) {    // looping untuk membuat thread sebanyak 2x
                err = pthread_create(&(tid[i]), NULL, &playAndCount, NULL); // membuat thread
                if (err!=0)
                        printf("\nCan't create thread: [%s]", strerror(err));
                else
                        // printf("\n Thread created successfully\n");
                        i++;
        }
        pthread_join(tid[0], NULL); // thread join untuk menunggu thread pertama selesai
        pthread_join(tid[1], NULL); // thread join untuk menunggu thread kedua selesai

        return 0;
}
                                                                    52,1          Bot
```

Output:

```
panji@PANJI-PC: ~/Documents/SISOP/PRAK2/modulsisop2
20VLC media player 2.1.6 Rincewind (revision 2.1.6-0-gea01d28)
vlc: unknown option or missing mandatory argument `--quite'
Try `vlc --help' for more information.
```

```
panji@PANJI-PC: ~/Documents/SISOP/PRAK2/modulsisop2
14
```

b. Cara compile program yang menggunakan thread

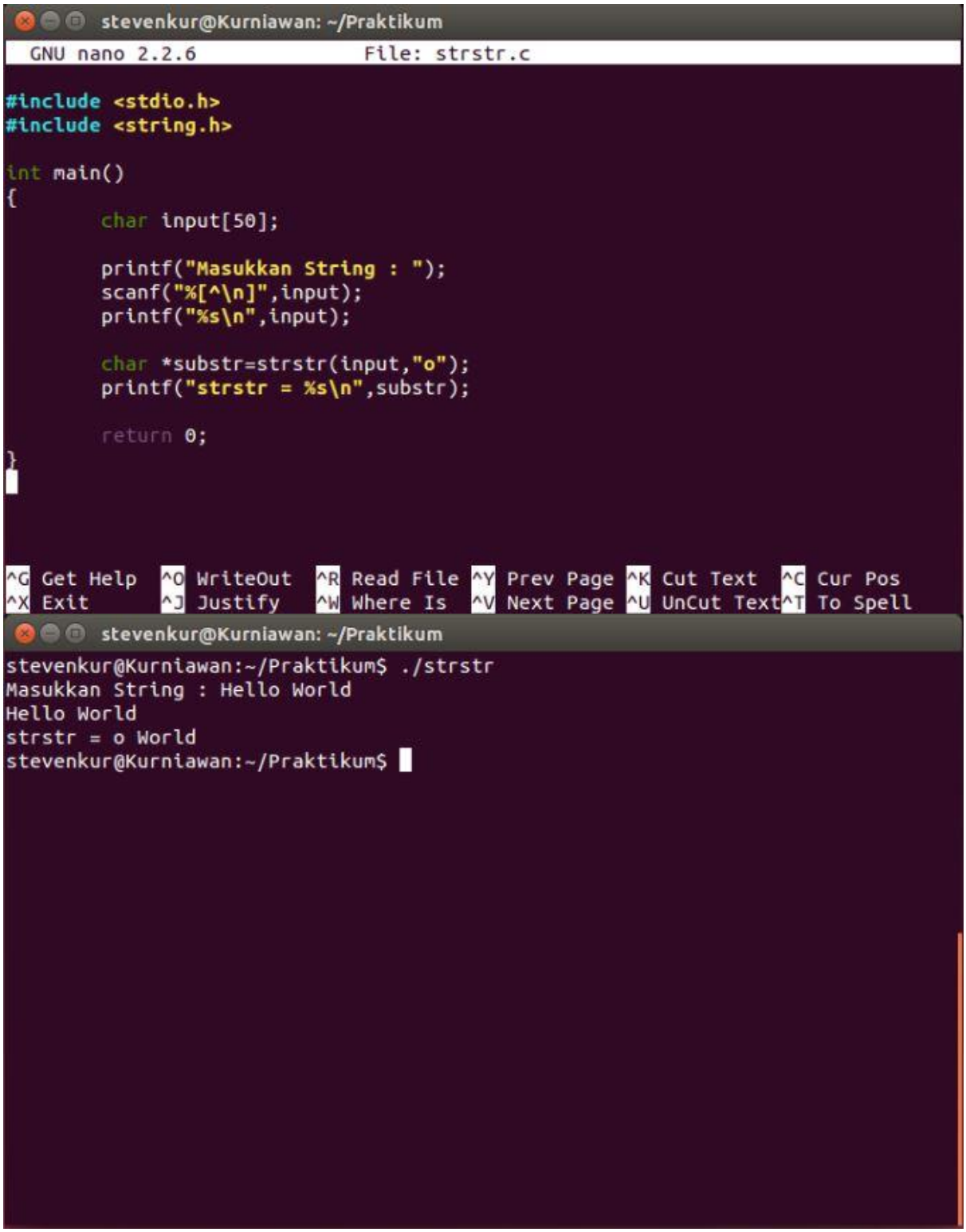
Dengan menambahkan parameter “-lpthread” di akhir perintah

gcc -o *fileexecutable filesourcecode* -lpthread

ctt: POSIX Thread : Harus return value (void*), pointer cuma 1 (*void)

3. STRING

Fungsi strstr: fungsi ini akan mencari lokasi substring dari suatu string. Cara penulisannya adalah strstr(string,index). Contoh:

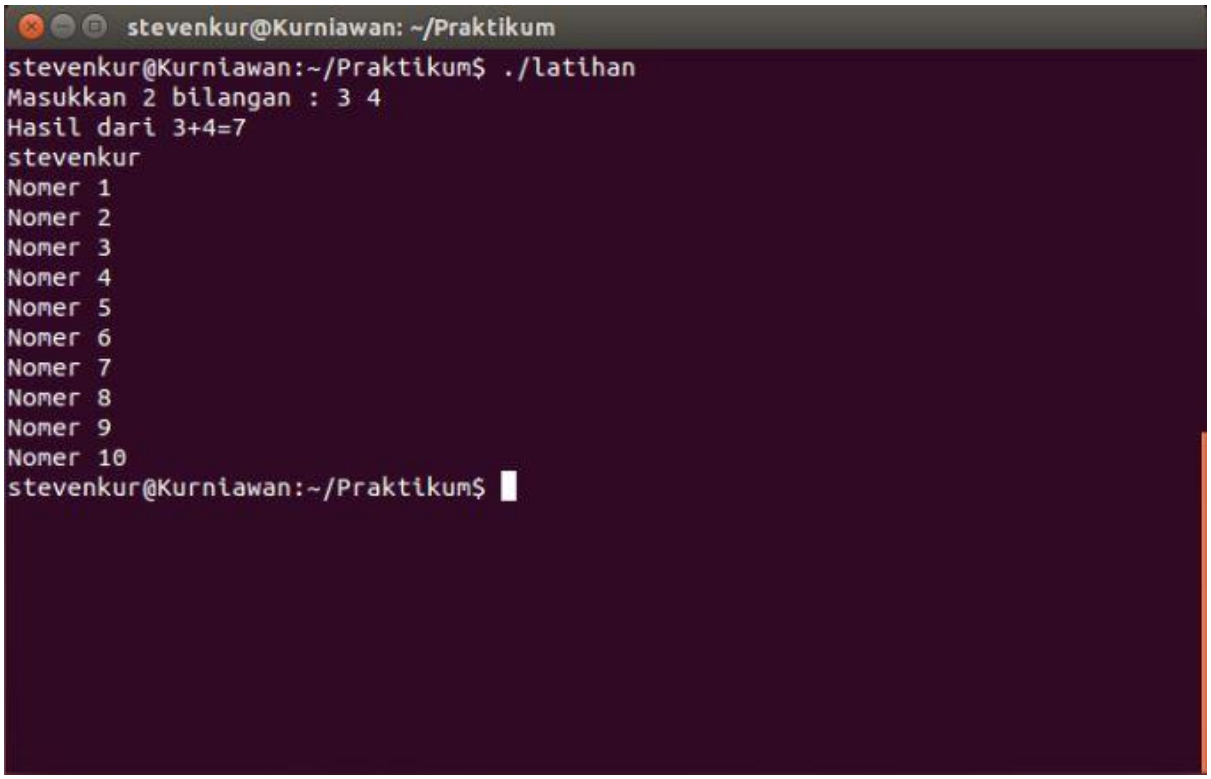


Fungsi strtok: fungsi ini membagi string menjadi beberapa bagian yang dibatasi oleh karakter yang telah ditentukan. Cara penulisannya adalah strtok(string,index). Apabila kita ingin melanjutkan bagian yang kedua dan seterusnya, maka fungsi strtok ditulis dengan strtok(NULL,index) dan dilakukan looping sampai fungsi strtok!=0. Contoh:

4. FORK

Fork & Exec: *system call* fork yang berfungsi untuk membuat proses baru. Proses yang memanggil *system call* fork ini akan dibagi jadi dua, proses induk dan proses turunan yang identik. Analoginya seperti pembelahan sel, dimana satu sel membelah jadi dua sel yang identik. Proses induk dan turunan independen satu sama lain dan berjalan bersamaan. *Return code* dari *system call* ini adalah suatu integer. Untuk proses anak *return code*-nya adalah 0 sementara untuk proses induk *return code*-nya adalah nomor identifikasi proses (PID) dari turunannya. Ada juga *system call* exec yang berguna untuk membuat proses turunan yang terbentuk memiliki instruksi yang

berbeda dengan proses induknya. Dengan kata lain, proses induk dan proses turunan tidak lagi identik tapi masing-masing punya instruksi berbeda. Contoh:



5. SIGNALING

Signal adalah sebuah event yang dikirimkan ke suatu proses oleh proses itu sendiri atau proses lain. Kadang juga digunakan untuk mengirimkan notifikasi kepada suatu proses bahwa ada suatu event. Penggunaannya harus menggunakan library sebagai berikut.

```
#include <sys/types.h>
#include <signal.h>
```

Untuk menangkap signal dibutuhkan signal handler kita bisa membuat signal handler. Contoh penggunaan fungsinya:

```
void signalhandler()
```

Untuk memanggilnya bisa menggunakan system call signal (signal, namafungsi)