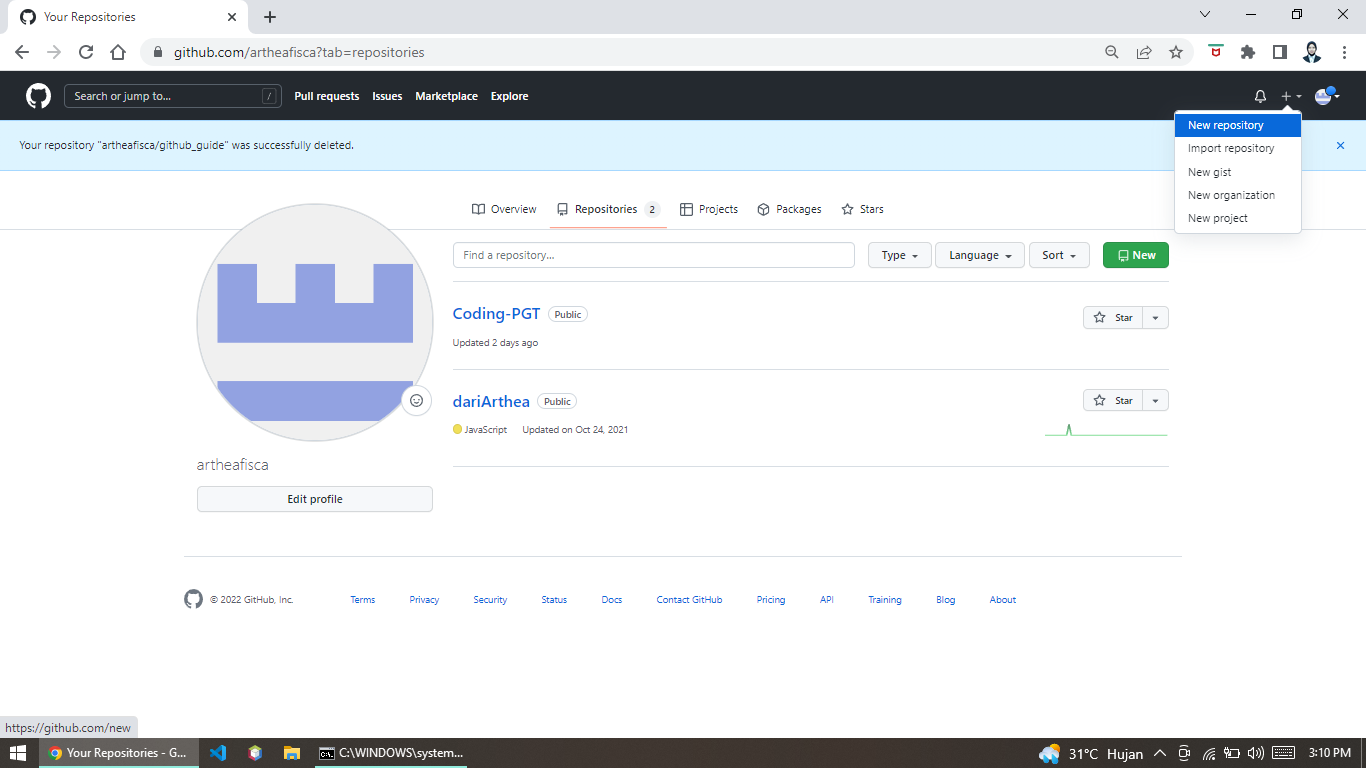
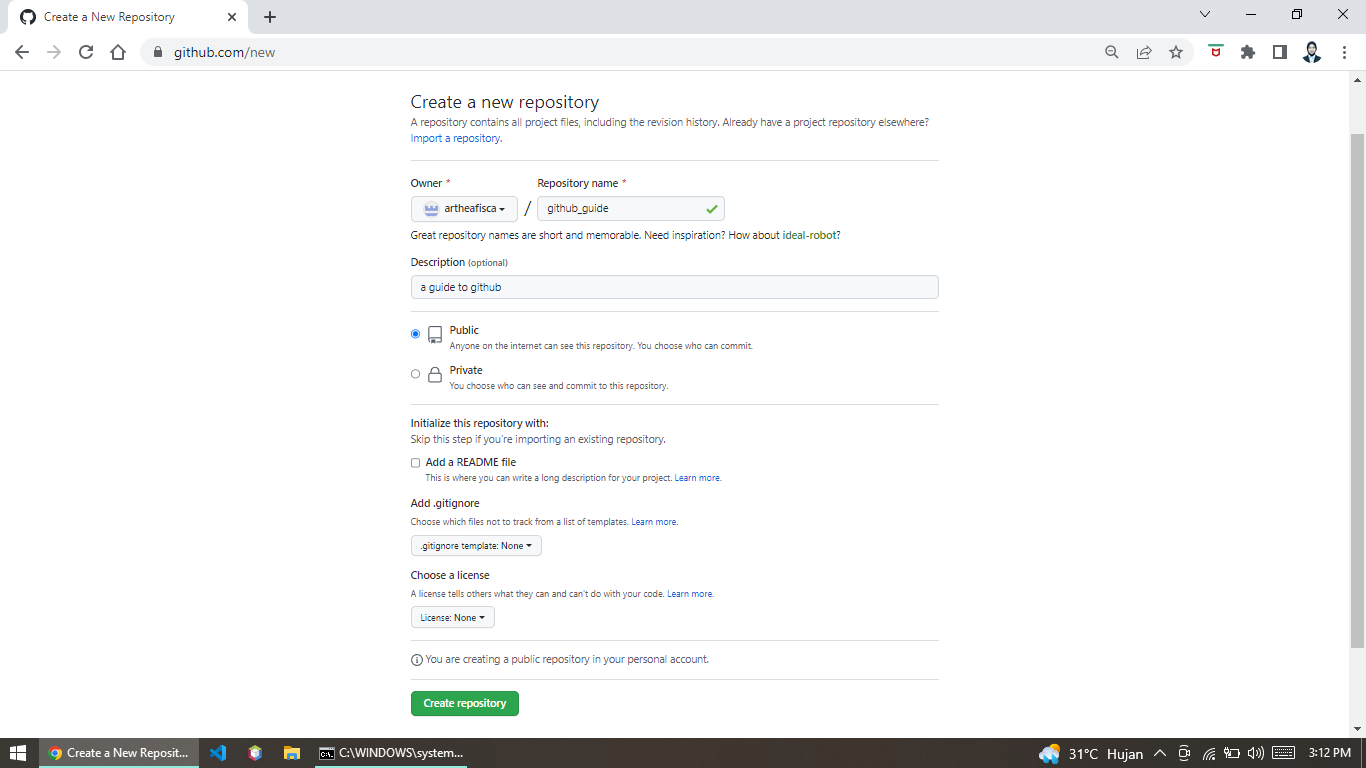
Step 1: Inisialisasi Proyek Baru

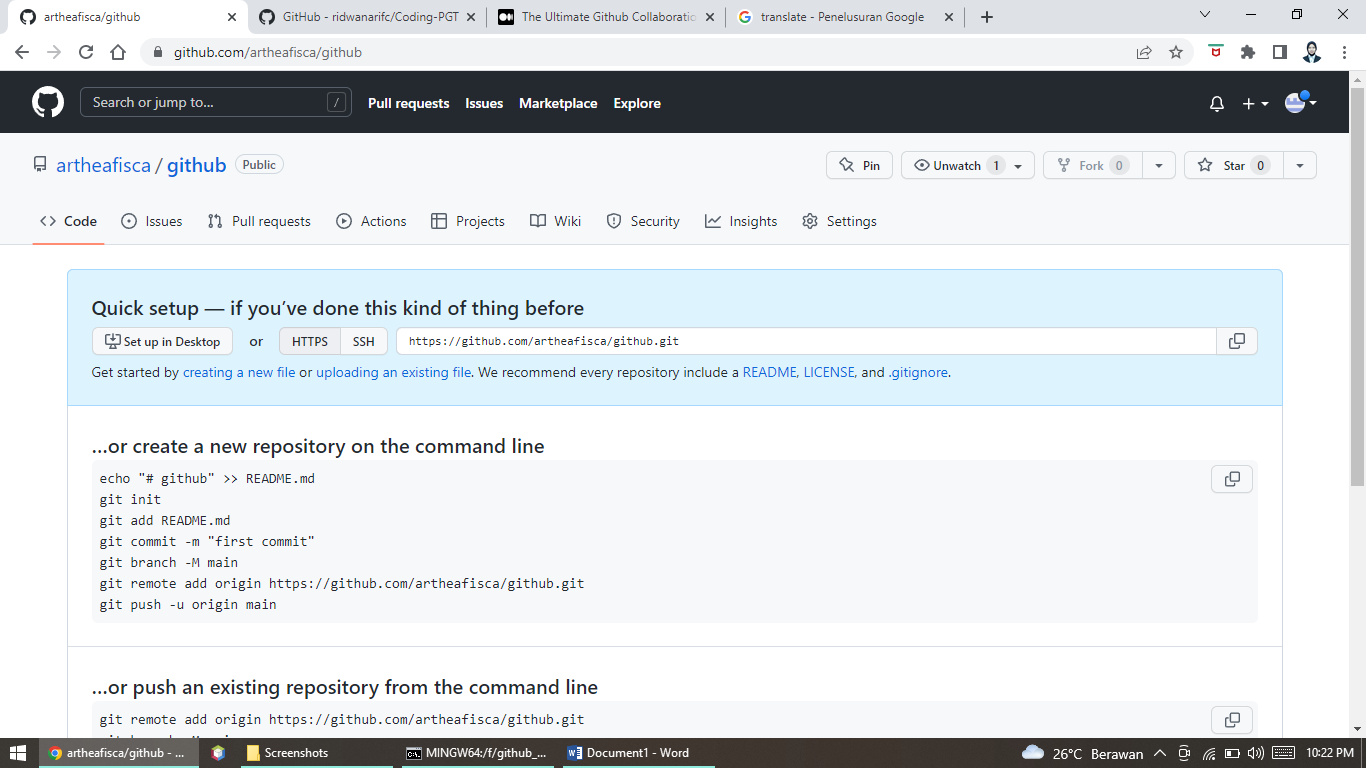
Buka Github dan klik tombol '+' di pojok kanan rop dan pilih 'New Repository'.



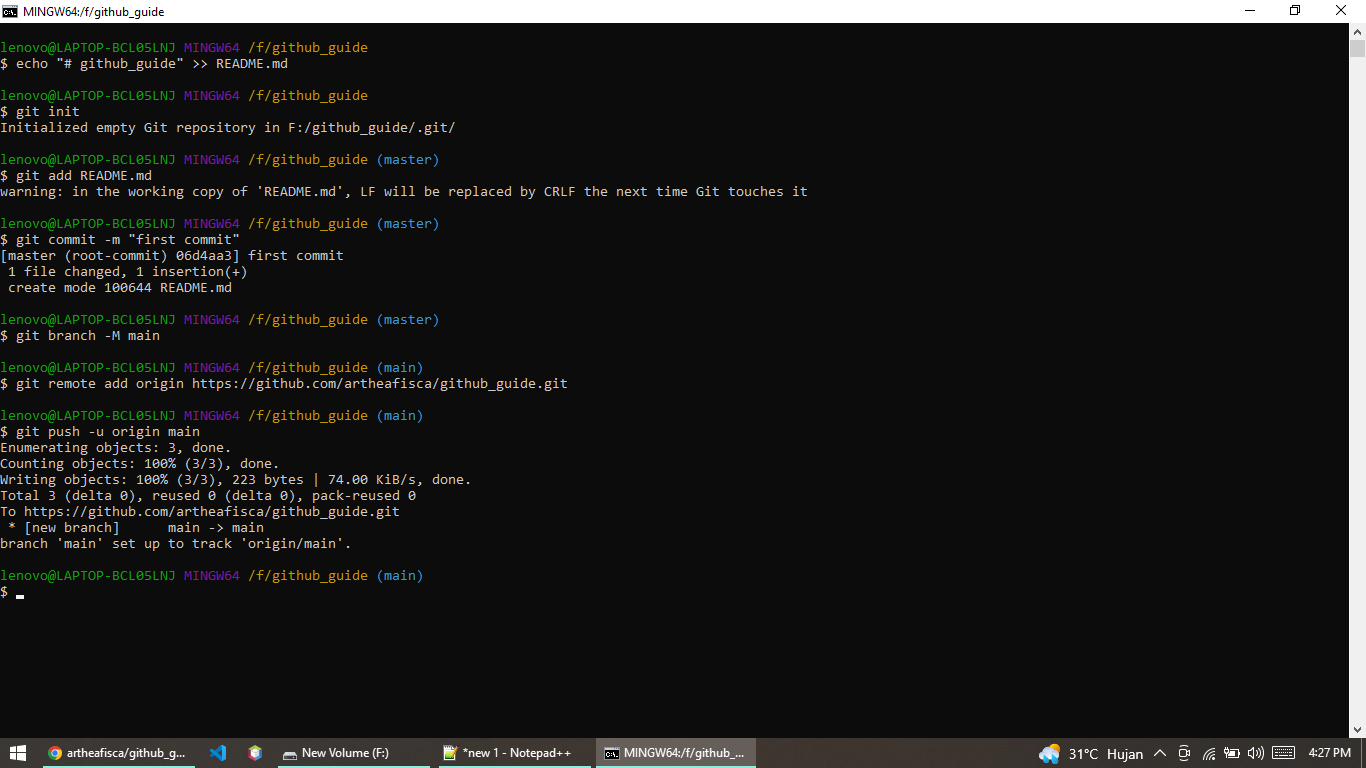
Kemudian isi kolom Repository name dan Description. Tetap publik, dan jangan "Initialize this repository with a README". Jangan mengubah apa pun. Klik "Create repository".



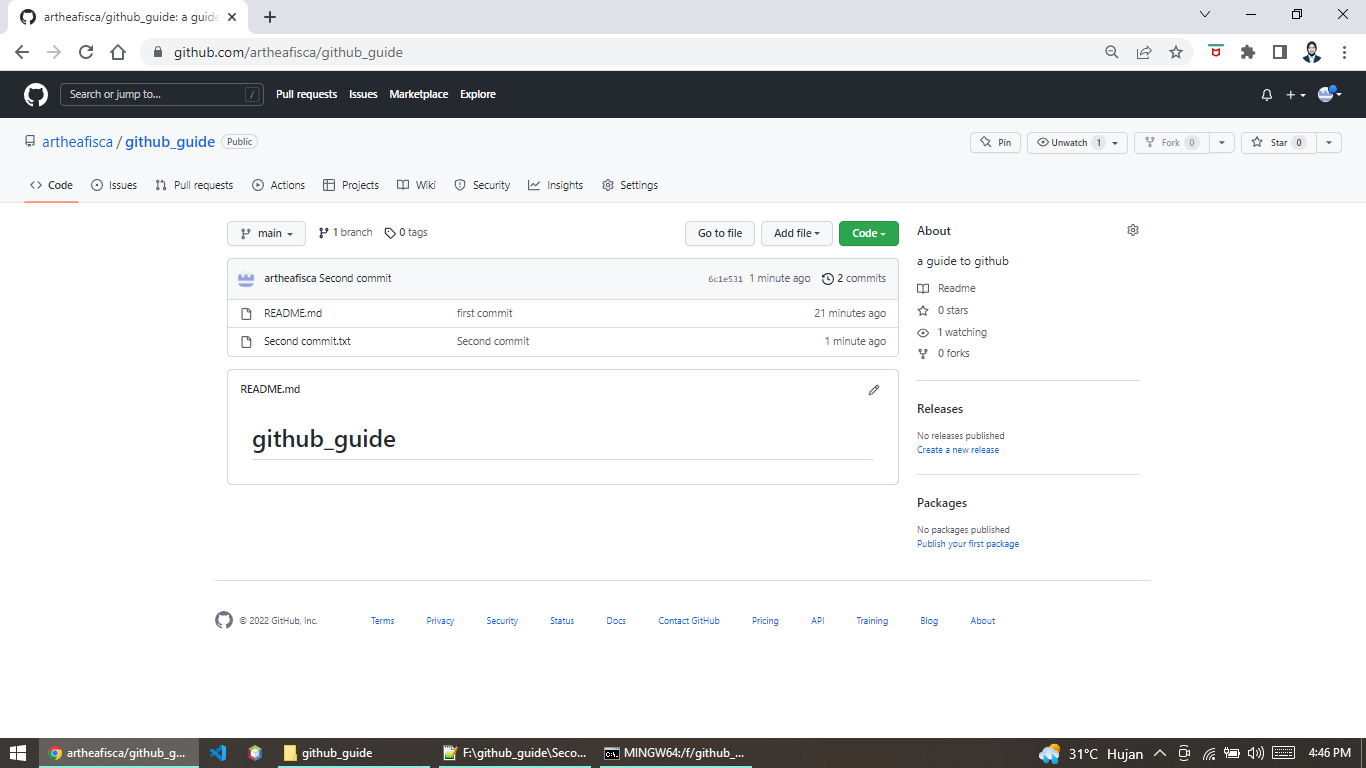
Selanjutnya Anda akan melihat halaman pengaturan. Ini adalah instruksi untuk menghubungkan Repo yang baru saja Anda buat di Github (Remote) ke direktori yang Anda buat di terminal Anda (Lokal).



Rekatkan baris di kotak merah baris demi baris yang dimulai dengan "echo ..." ke terminal saat Anda masuk ke direktori yang baru saja Anda buat secara lokal. Terminal Anda akan terlihat seperti ini setelah Anda selesai:



Dan jika Anda pergi ke halaman Repo Github Anda, Anda akan melihat ReadMe yang Anda inisialisasi dan referensi ke komit pertama yang Anda buat.

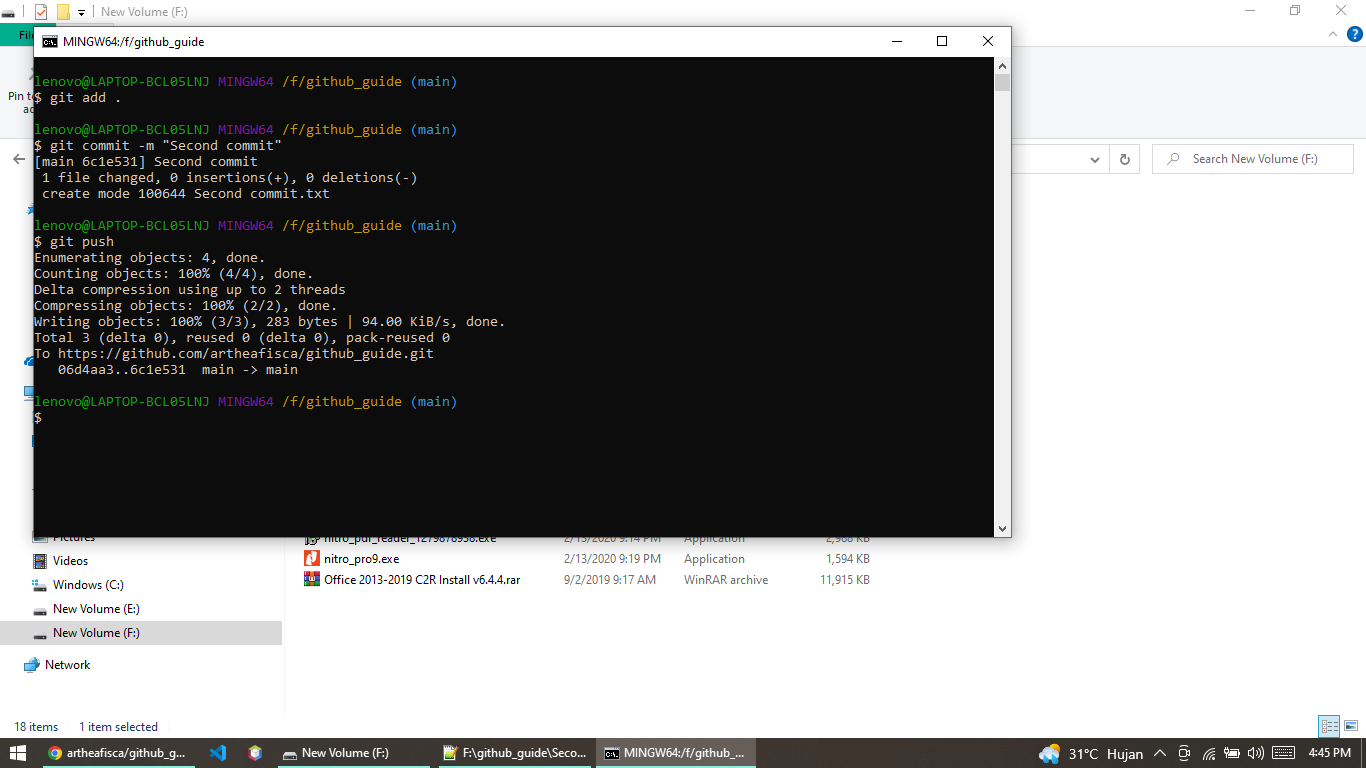


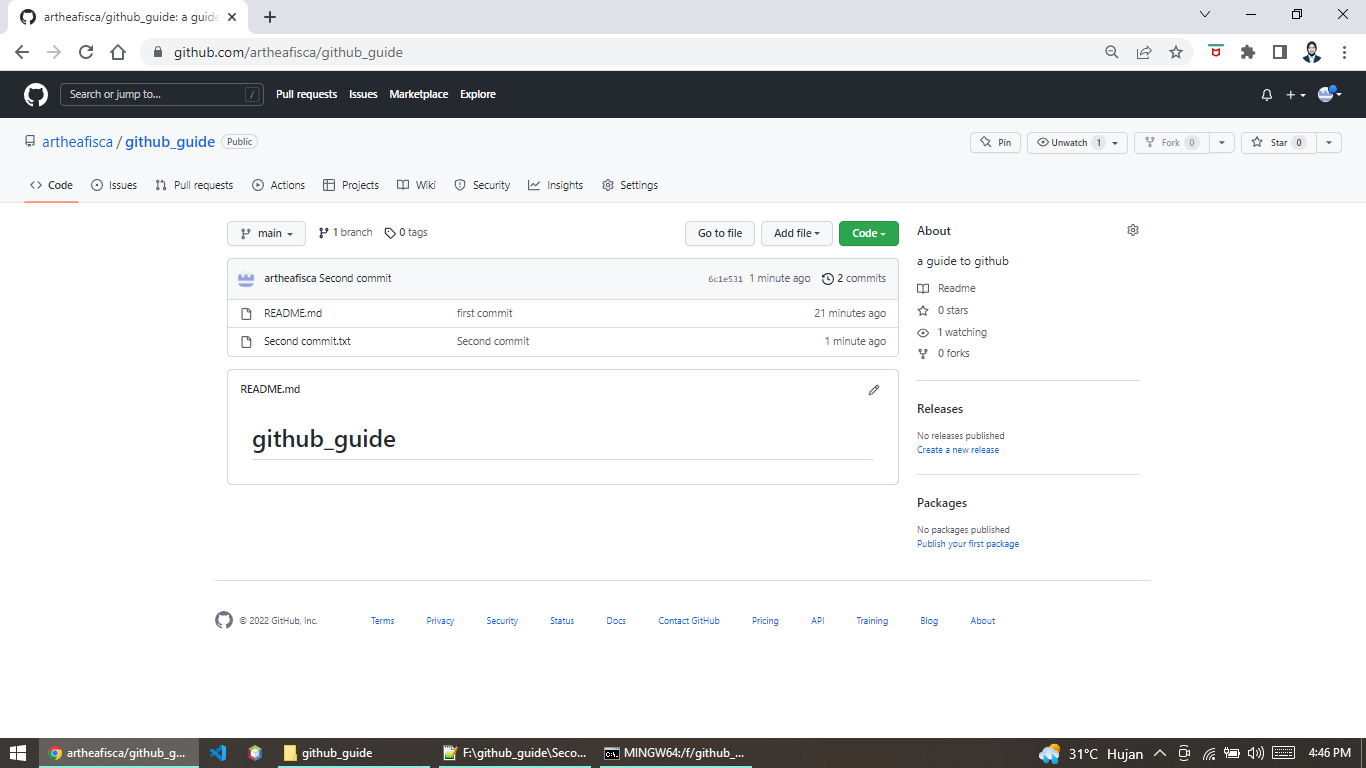
Sekarang mari kita perbarui Repo ini. Kembali ke terminal Anda dan git add, git commit, dan git Push:

$ git add .

$ git commit -m "Second commit"

$ git push

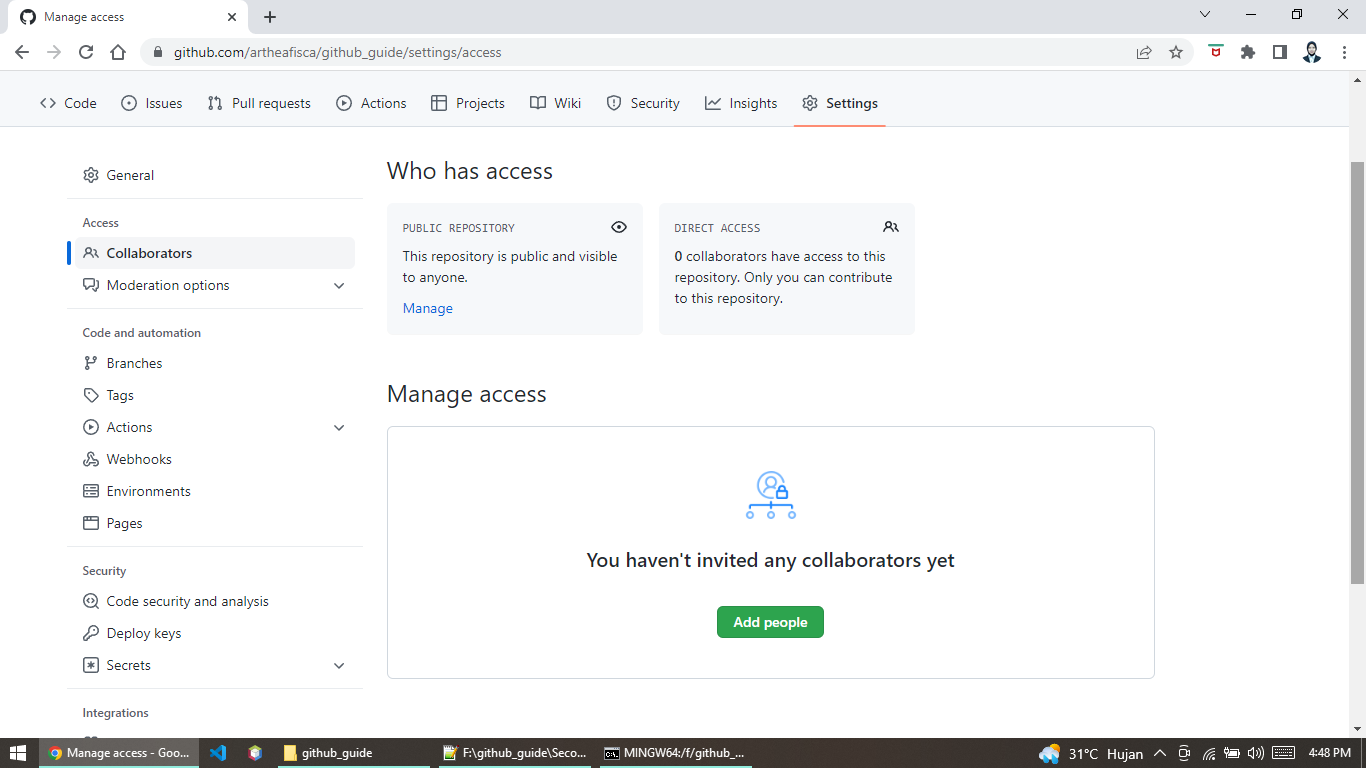


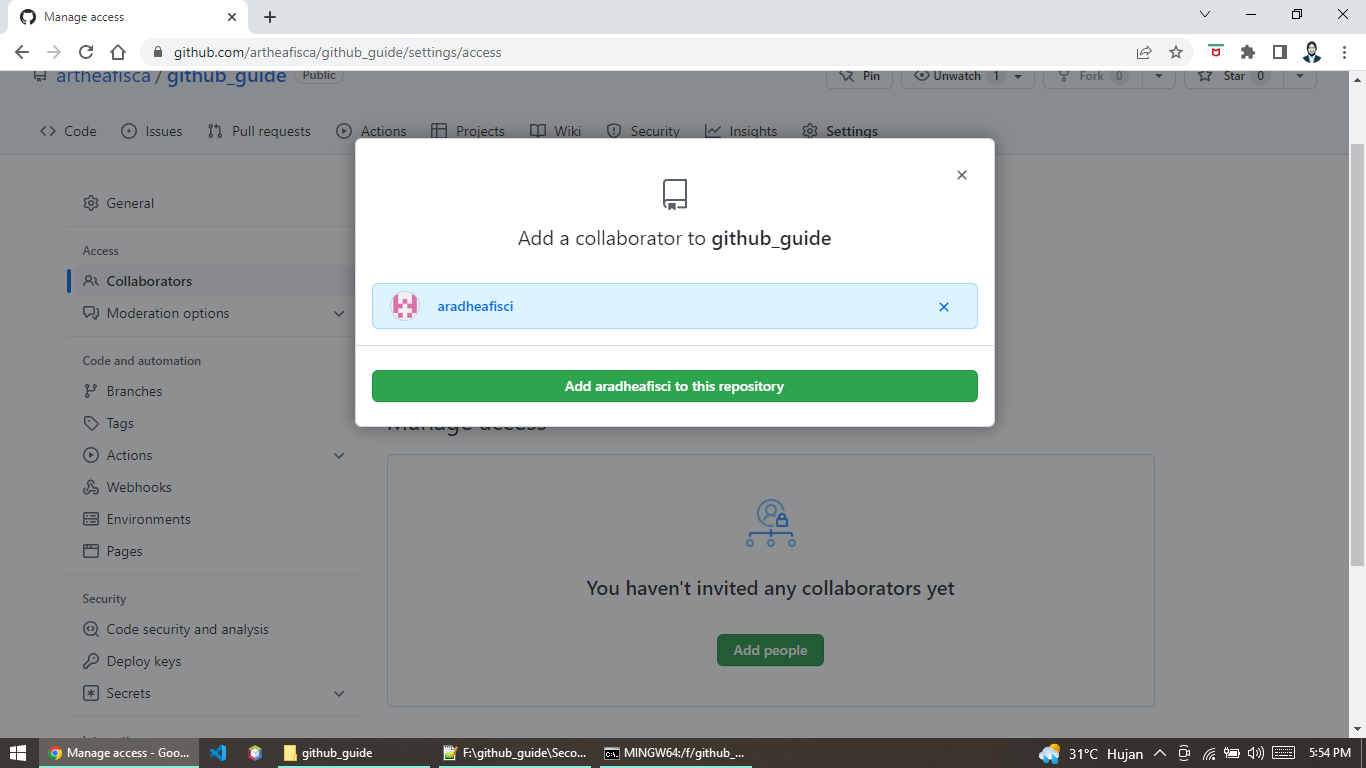


Anda telah diinisialisasi dan siap untuk mulai bekerja!

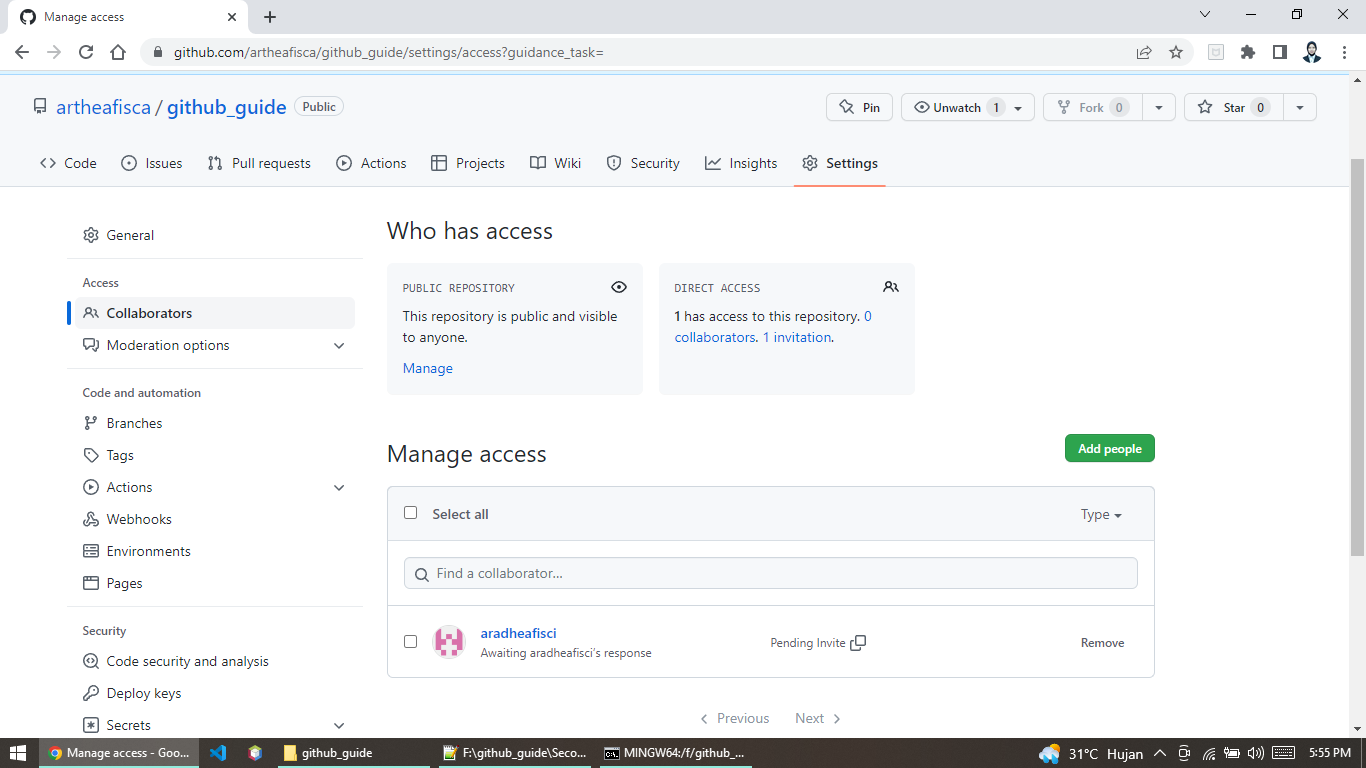
Step 2: Siapkan Tim Anda

Klik pada tab "Setting" perwakilan Anda, lalu "Collaborators" lalu cari pengguna Github dan tambahkan mereka dengan mengklik "Add Collaborator":

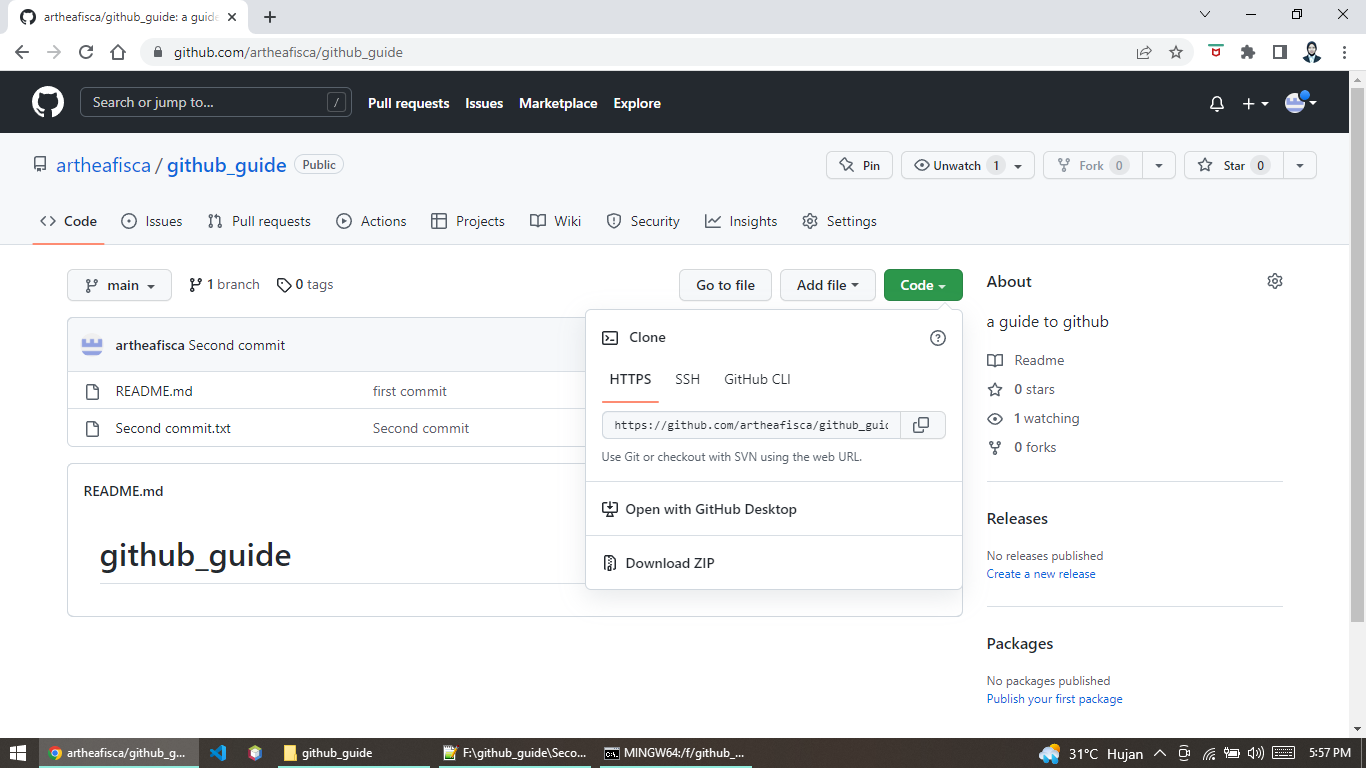




Mereka akan menerima email yang memberi tahu mereka bahwa Anda menambahkan mereka dan akan terdaftar sebagai kolaborator.



Jika Anda seorang kolaborator, buka halaman Github Repo, Git Clone the project, dan cd ke direktori. Jangan memotongnya! Forking akan menyalinnya di Repo baru ke halaman Github Anda, tetapi Anda tidak menginginkannya — Anda ingin berkolaborasi di Repo Github yang sama dengan rekan tim Anda.



$ git clone [git@github.com:MinesJA/github\_guide.git](mailto:git@github.com:MinesJA/github_guide.git)

$ cd github\_guide/

Dan sekarang Anda siap untuk berkolaborasi!

Step 3: Bekolaborasi

Saat Anda menggunakan git untuk mengerjakan proyek yang sama dengan banyak orang, ada satu aturan utama yang harus Anda ikuti:

CABANG UTAMA HARUS SELALU DIPLOYABLE

Cara agar Master dapat digunakan adalah dengan membuat cabang baru untuk fitur-fitur baru dan menggabungkannya ke dalam Master setelah selesai. Inilah cara kerjanya.

Step 3a: Branches

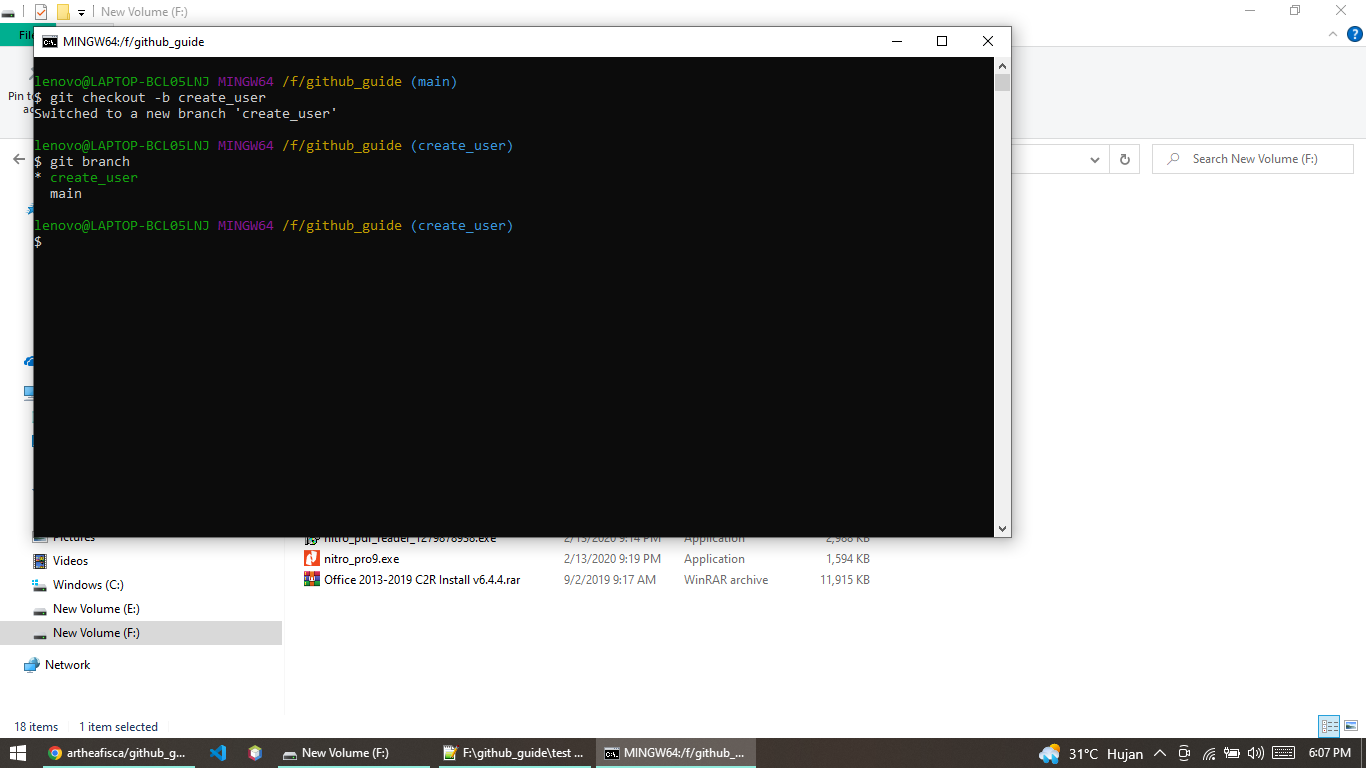
Jadi katakanlah Anda ingin membuat model Pengguna. Di terminal Anda, buat cabang baru:

$ git co -b create\_user

“co” adalah singkatan dari “checkout” yang digunakan untuk berpindah antar cabang. Menambahkan "-b" dan nama di akhir membuat cabang baru dan kemudian pindah ke cabang baru itu untuk kita.

Anda harus dapat memverifikasi ini dengan perintah:

$ git branch



Anda sekarang berada di cabang baru dan dapat mulai membuat kode.

Catatan: Sebagai aturan umum, Anda harus sering git add dan git commit ketika Anda menyelesaikan sesuatu yang memungkinkan kode Anda berfungsi (berakhir menjadi beberapa kali dalam satu jam). Misalnya, ketika Anda menyelesaikan suatu metode dan basis kode berfungsi, git commit seperti ini:

$ git commit -m "Added function to allow Users to say 'Hello World'"

Step 3b: Submitting Pull Requests

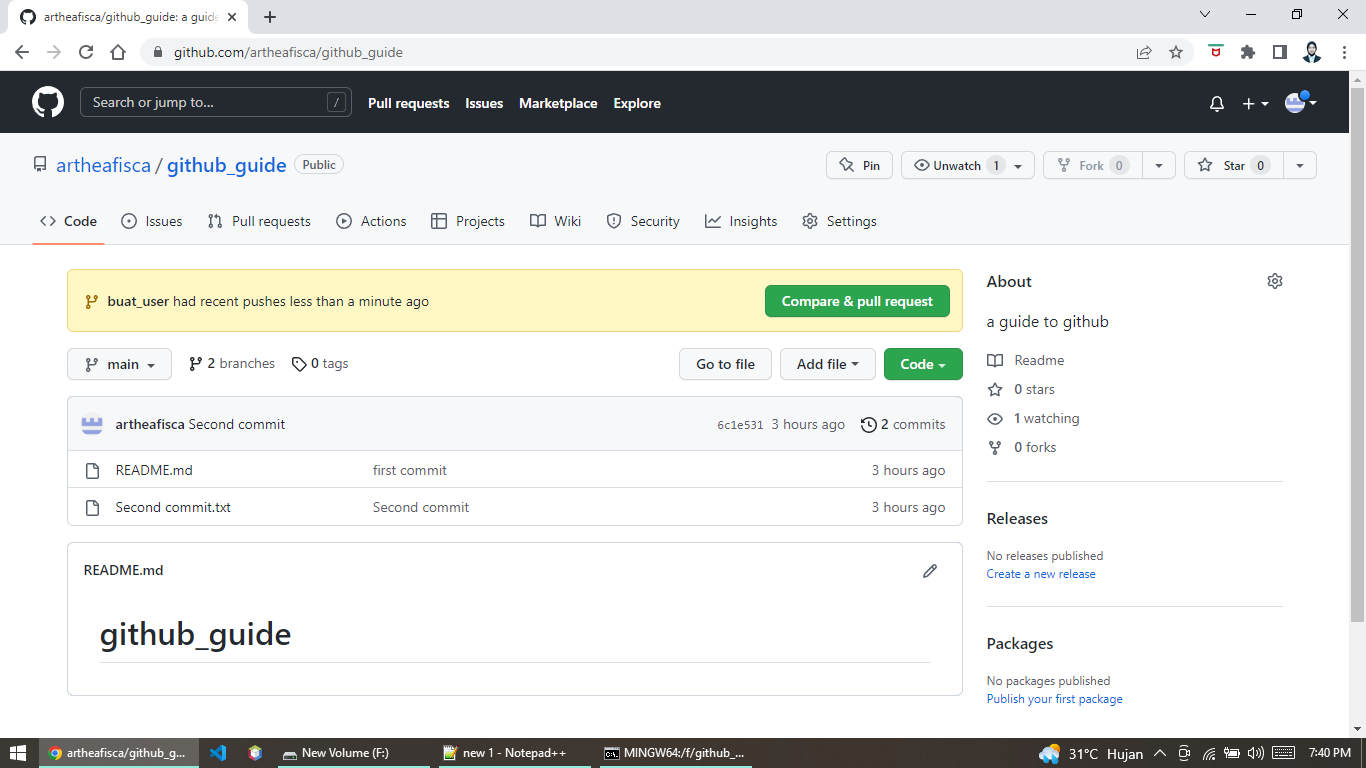
Pertama, tentukan siapa yang akan bertanggung jawab menangani penggabungan. Semakin sedikit orang yang bertindak secara independen dalam penggabungan, semakin baik sehingga untuk tim yang terdiri dari 4 orang, Anda mungkin perlu memiliki satu "Reviewer" atau "Merge Master" resmi.

Selanjutnya, minta semua orang git Push cabang mereka:

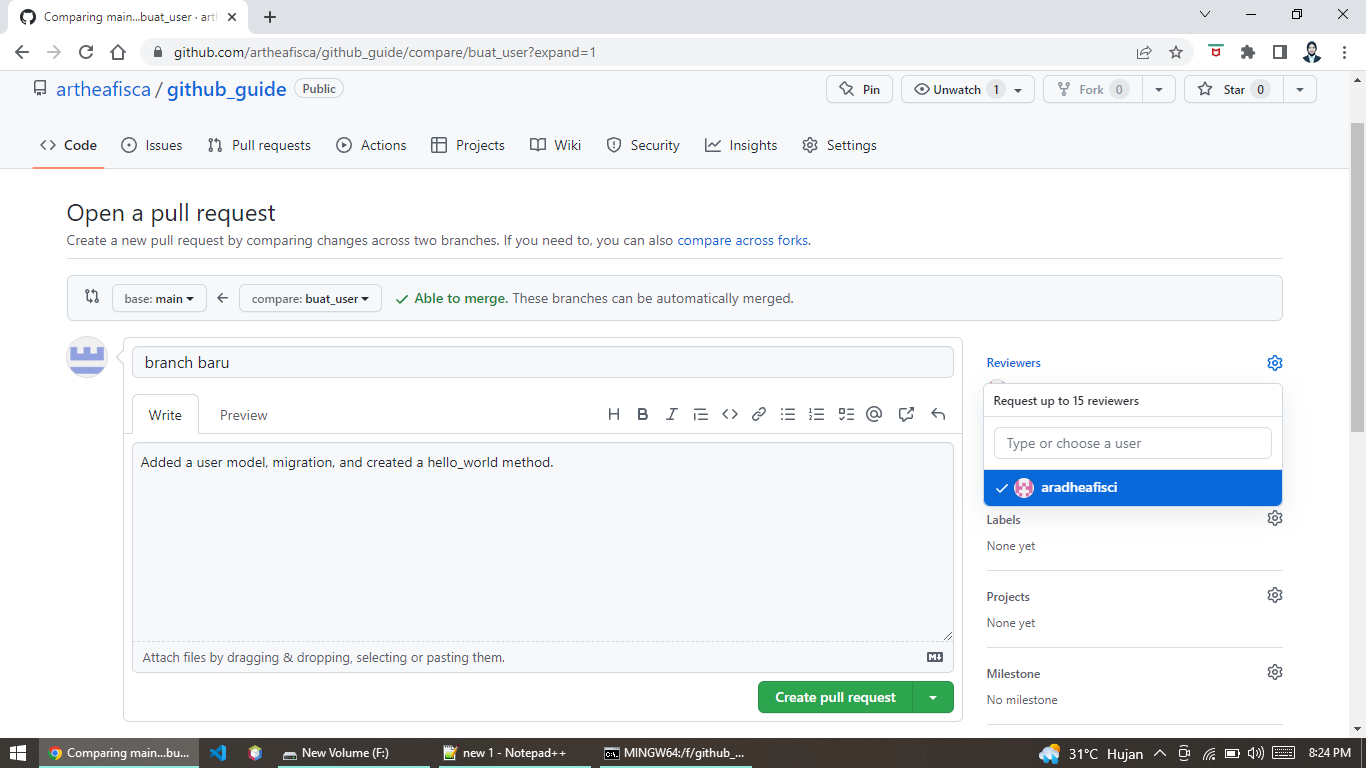
$ git push

Sekarang pergi ke halaman Repo Github. Anda akan melihat cabang yang Anda dorong di bilah kuning di bagian atas halaman dengan tombol "Compare & pull request”".

Catatan: Atau, Anda dapat memilih cabang di menu tarik-turun “Branch:” dan pilih cabang yang baru saja Anda tekan. Anda kemudian akan memiliki tombol "Pull Request" dan "Compare" di sisi kanan.



Klik "Compare & pull request Ini akan membawa Anda ke halaman "Open a pull request". Dari sini, Anda harus menulis deskripsi singkat tentang apa yang sebenarnya Anda ubah. Dan Anda harus mengklik tab “Reviewers” ​​dan memilih siapa pun yang diputuskan oleh tim Anda sebagai “Merge Master”. Setelah selesai, klik "Create ull request".

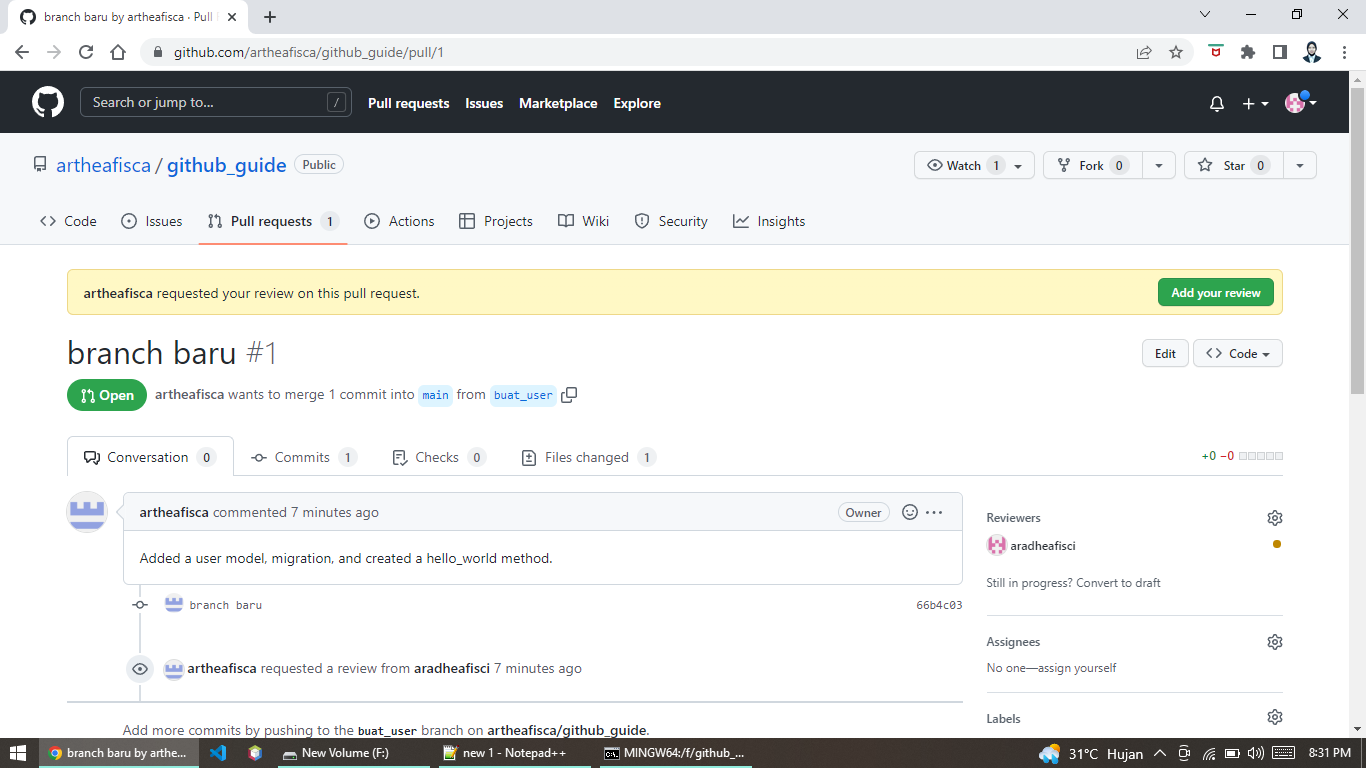


Step 3c: Merging Pull Requests

Perhatikan bahwa jika Anda seorang kolaborator, Anda dapat menggabungkan permintaan tarik Anda sendiri. Namun, sekali lagi, jika Anda bekerja dalam tim, lebih masuk akal jika satu orang melakukan semua penggabungan dan semua orang lainnya mengirimkan "Pull requests" dan menetapkan "Master Merger" sebagai peninjau hanya untuk memastikan Anda berurusan dengan konflik gabungan apa pun satu cabang pada satu waktu.

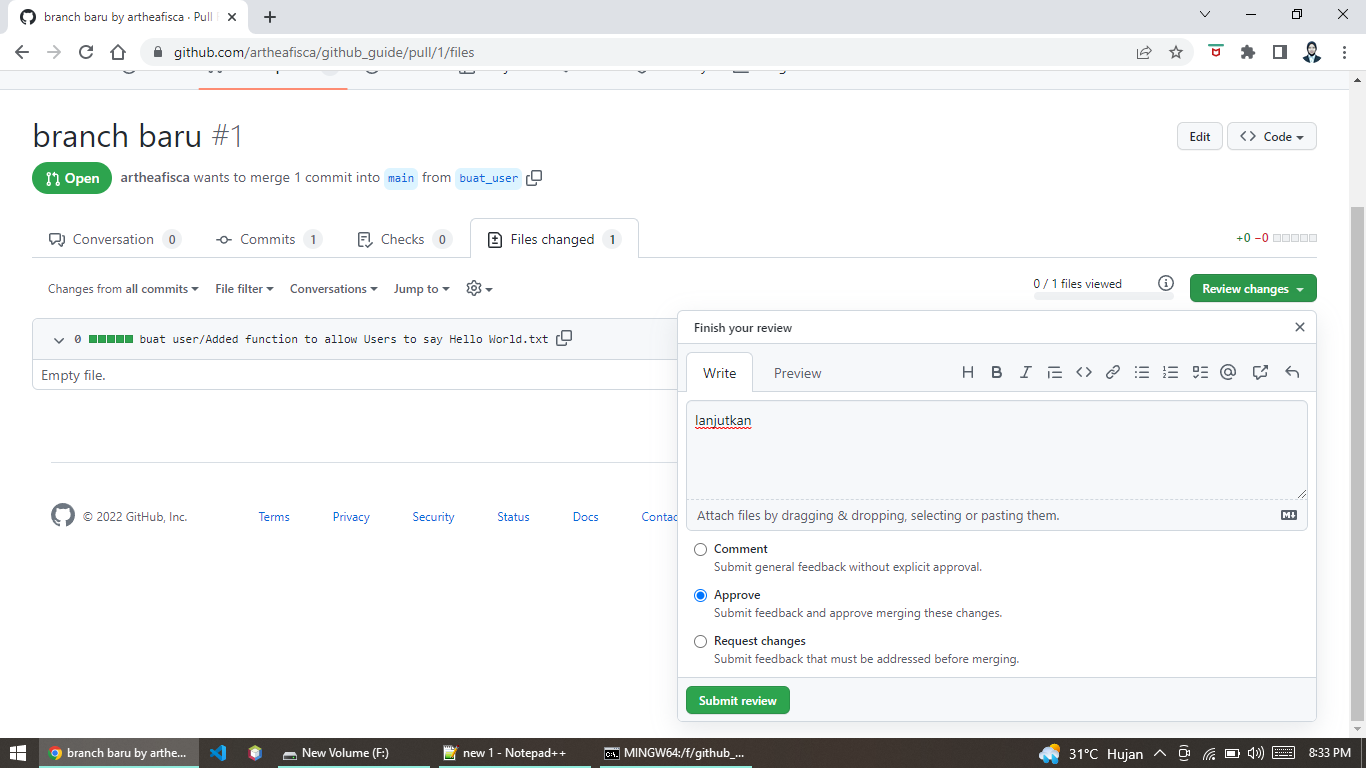
Jadi, dengan asumsi Anda adalah orang yang bertanggung jawab untuk mengurus semua penggabungan dan seseorang telah menugaskan Anda sebagai "Reviewer" pada permintaan tarik, ketika Anda masuk ke Github Anda, Anda akan melihat Anda memiliki pemberitahuan yang memberi tahu Anda bahwa seseorang telah menugaskan Anda sebagai pengulas. Anda juga akan melihat bilah kuning yang menunjukkan salah satu rekan tim Anda sebagai "requested on this pull request."

Silakan dan klik tombol “Add your reviewer”.

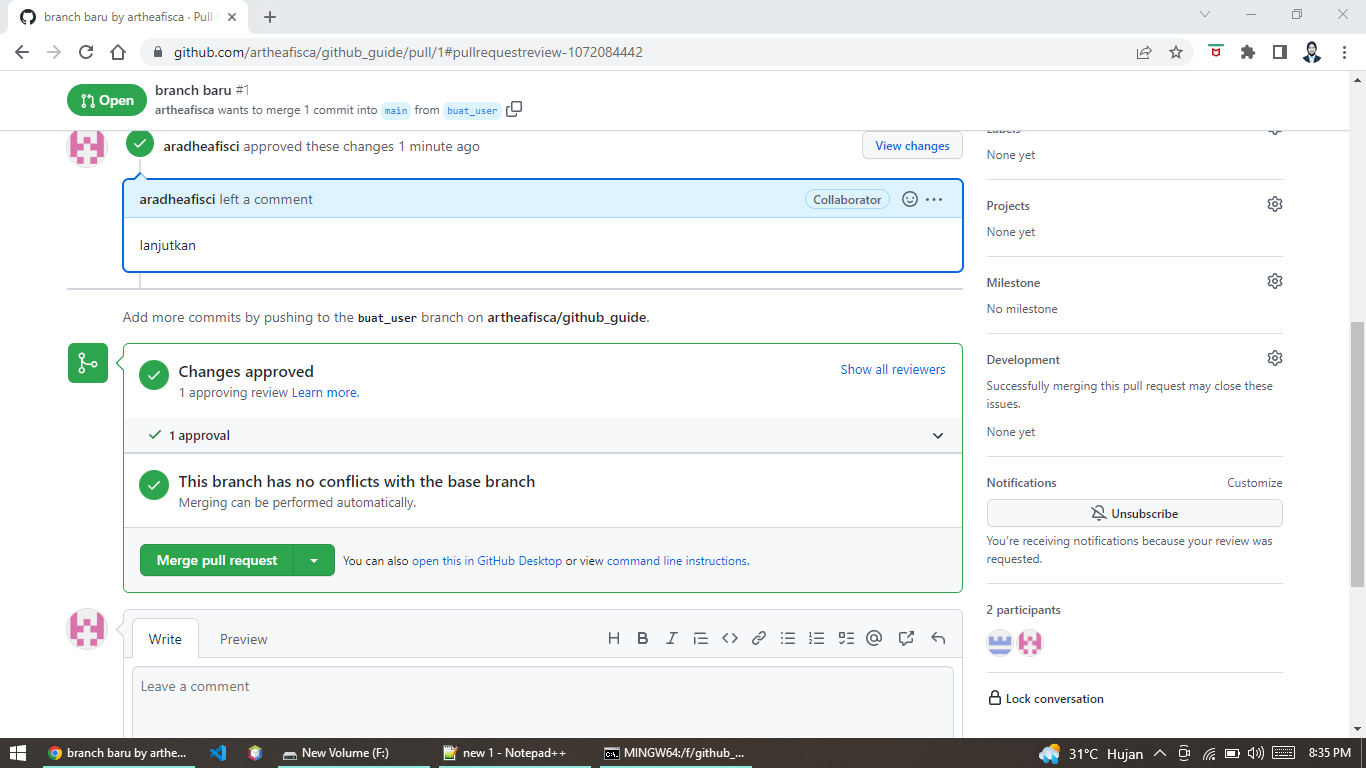


Ini akan membawa Anda ke halaman Permintaan Tarik. Bagaimana Anda bergerak maju dari sini terserah Anda dan tim Anda. Jika Anda bekerja bersama, saya akan menggunakan setiap pagi untuk duduk sebagai sebuah tim dan melakukan pull request bersama. Jika Anda melakukannya, Anda tidak perlu meninggalkan Ringkasan Ulasan yang bertele-tele dan terperinci.

Namun, jika Anda bekerja dari jarak jauh, ini akan menjadi alat utama Anda untuk memberi tahu pemohon jika mereka perlu membuat perubahan atau jika Anda akan menggabungkan permintaan mereka.



Ketika Anda mengklik "Submit review" pada tarik-turun "Review changes", ulasan Anda sekarang akan ada sebagai komentar di utas permintaan tarik.



Saat Anda puas dengan permintaan tarik, buka bagian bawah permintaan tarik dan klik "Merge pull request".

Anda kemudian akan melihat pesan "Pull request successfully merged and closed" dan tombol untuk "Delete branch" yang harus Anda klik.

