**[Cara Menggunakan GitHub dan Membuat Repository Baru](http://www.renotekno.net/2016/06/cara-menggunakan-github-dan-membuat.html)**

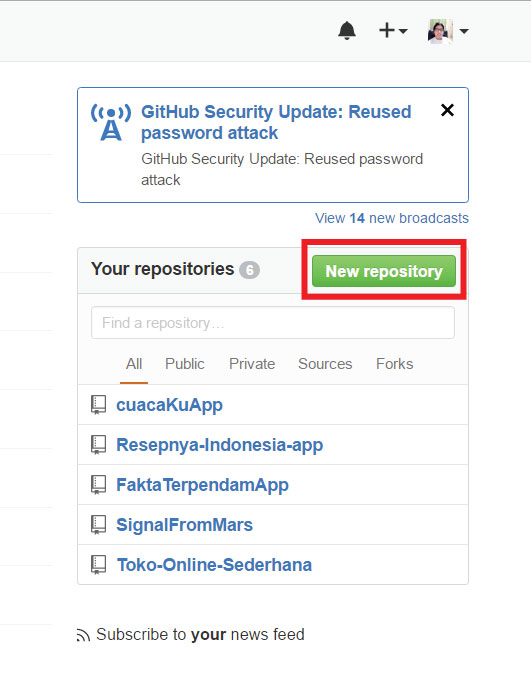
[AndroidDev](http://www.renotekno.net/search/label/AndroidDev), [WebDev](http://www.renotekno.net/search/label/WebDev) 20:03[1 Comment](http://www.renotekno.net/2016/06/cara-menggunakan-github-dan-membuat.html#comment-form)

Kali ini mimin akan share beberapa tutorial cara menggunakan VCS GitHub. Bagi kalian yang belum mengenal GitHub, kalian dapat membaca artikel mengenai [Apa itu GitHub ? dan Apa fungsi dari GitHub](http://www.renotekno.net/2016/06/apa-itu-github-dan-apa-fungsi-dari.html). Sebelumnya jika kalian belum memiliki akun GitHub kalian dapat membuat akun GitHub baru secara gratis di [halaman utama pada GitHub](https://github.com/).

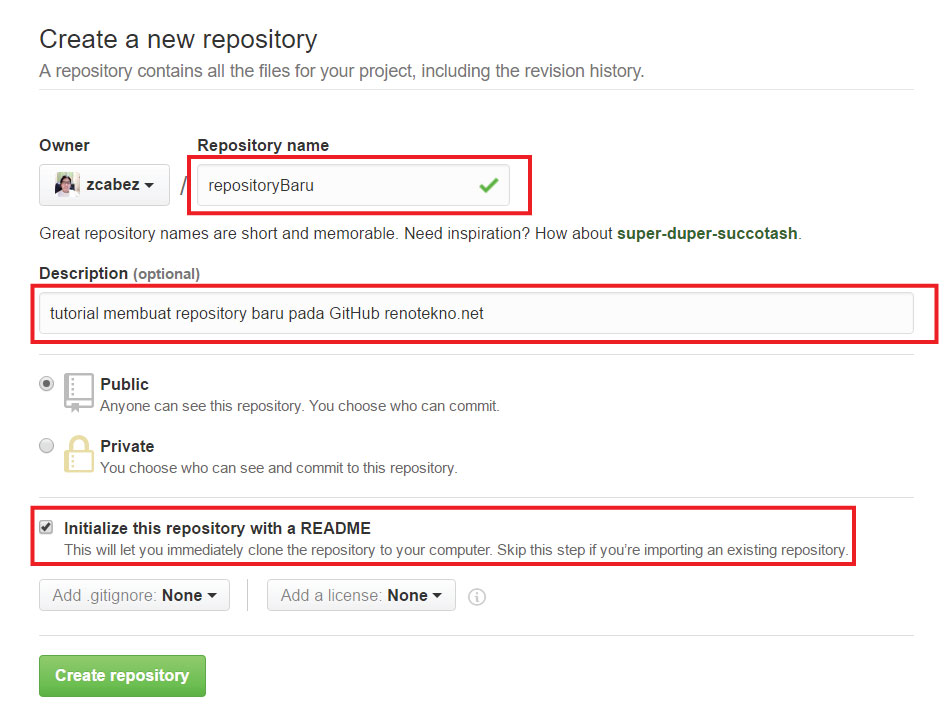
**Membuat Repository Baru**

Repository merupakan fungsi utama dalam menggunakan GitHub, repository merupakan tempat kalian menyimpan file-file dari project yang kalian buat. Repository juga akan melacak setiap perubahan yang kalian lakukan pada file-file. Berikut cara membuat repository baru pada GitHub:

1. Pada halaman utama setelah Login klik pada **New Repository**. Dapat kalian lihat juga pada bagian bawahnya merupakan daftar repository yang telah kalian buat

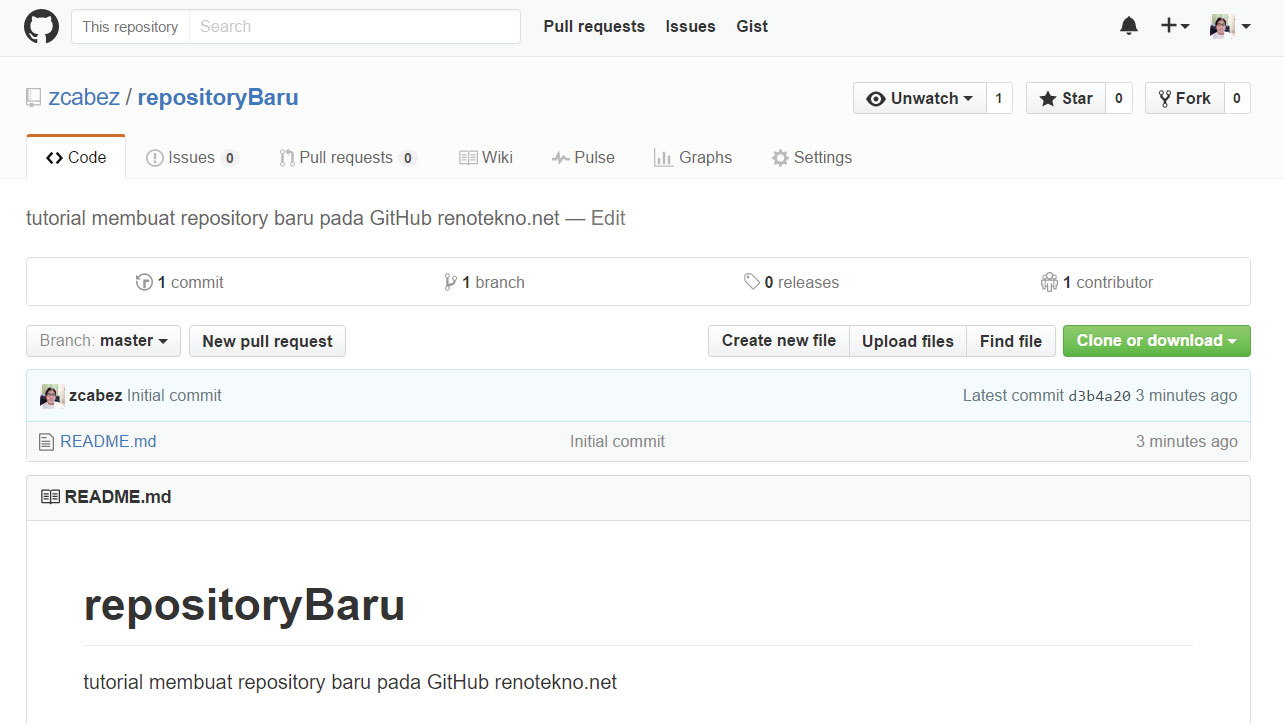


1. Isikan data sesuai dengan yang kalian inginkan. Pastikan kalian mengisi data sesuai dengan project yang ingin kalian buat agar dapat membedakan dengan repository lainya.



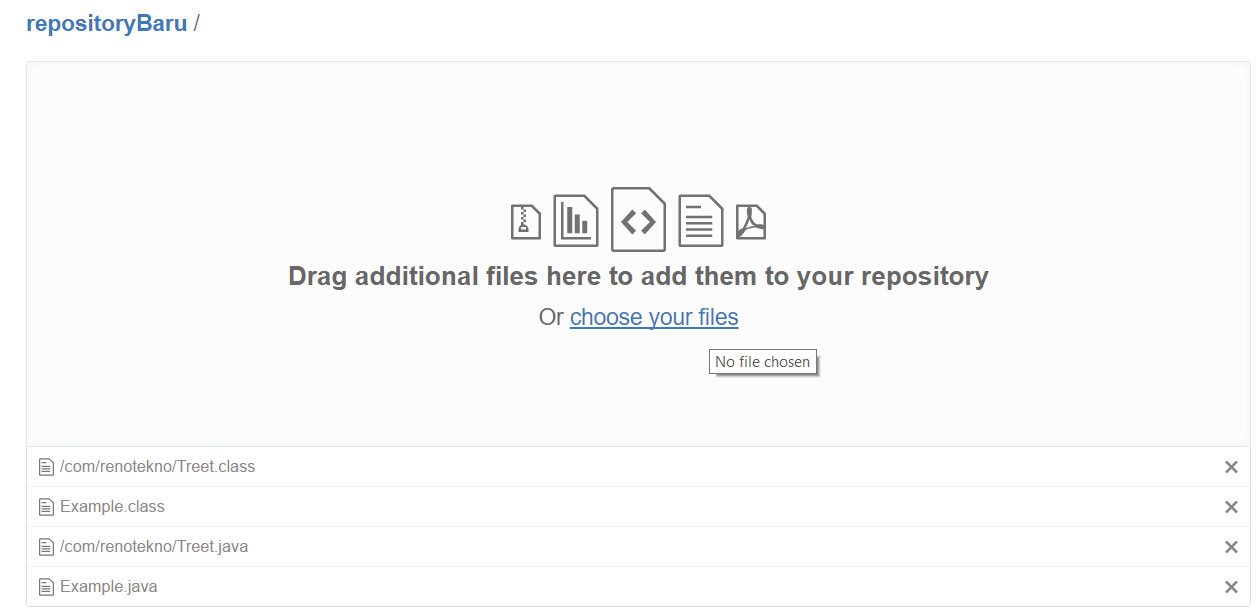
**Repository name** - Masukkan nama repository kalian.  
**Description** - Masukkan deskripsi singkat mengenai project kalian.  
Centang juga pada **Initialize this repository with a README** untuk memberikan informasi kepada orang lain mengenai project kalian. Setelah selesai mengisi data, klik pada **Create repository** untuk membuat repository.

**Upload File Project ke Repository**



Ini merupakan tampilan halaman setelah kalian selesai membuat repository. Dapat kalian lihat pada repository yang kalian buat terdapat 1 file yaitu **README.md** yang kita buat saat pertamakali pembuatan repository. Kalian juga dapat mengedit file README.md agar mempunyai informasi yang lebih detail mengenai project yang kalian buat. Langkah selanjutnya adalah menambahkan file-file project ke dalam repository dengan cara:

1. Klik pada tombol **Upload files**.
2. Klik pada **Choose your files** untuk memilih file yang ingin kalian masukkan ke repository kalian. Kalian juga bisa melakukan **drag & drop** pada kotak upload agar lebih cepat.



1. Langkah selanjutnnya adalah melakukan **Commit**. Kalian juga harus menyediakan keterangan singkat mengenai file yang di upload. Kalian juga dapat menambahkan deskripsi (opsional). Keterangan singkat ini pada tiap commit yang kalian lakukan berguna agar kalian dapat sekilas melihat perubahan yang dilakukan.

**Melakukan Perubahan Pada File**

Tentunya file yang kita upload bisa mengalami perubahan. Untuk melakukan perubahan kalian bisa dengan mengupload file dengan nama yang sama ataupun langsung mengeditnya pada GitHub dengan cara:

1. Klik pada salah satu file yang ingin kalian edit.
2. Klik pada lambang pencil di sebelah kanan.

Seperti biasa, setelah melakukan perubahan dan sebelum melakukan **commit** maka sebaiknya kalian memberikan keterangan singkat mengenai perubahan yang kalian lakukan.

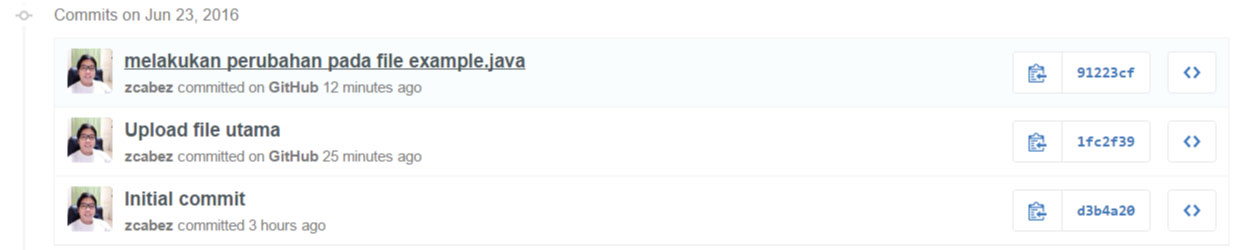
**Melihat Perubahan Yang Terjadi Pada Setiap Commit**

Kalian bisa melihat detail perubahan yang terjadi pada setiap commit dengan cara:

1. Buka halaman utama repository kalian. Pada status bar, kalian dapat melihat total commit yang telah kalian lakukan. Klik pada tulisan **commit** tersebut.



1. Pilih pada commit mana yang ingin kalian lihat perubahanya. Dapat kalian lihat setiap commit mempunyai keterangan singkat yang kalian masukkan.



1. Kalian dapat melihat detail perubahan yang terjadi. Baris yang berwarna Hijau merupakan baris yang baru ditambahkan dan baris yang berwarna Merah merupakan baris yang telah di hapus.



Bagaimana mudah bukan menggunakan GitHub ? Tentunya masih banyak fitur lainya yang dimiliki GitHub yang belum mimin bahas. Informasi di atas merupakan cara umum menggunakan GitHub dalam membuat sebuah project baik web ataupun aplikasi android. Dengan GitHub kalian bisa melacak setiap perubahan yang terjadi dan membandingkannya. Adapun tutorial lengkap yang disediakan GitHub dalam bentuk bahasa inggris di sini:

* [GitHub Guides](https://guides.github.com/activities/hello-world/" \t "_blank)

[**Apa itu GitHub ? Dan Apa Fungsi Dari GitHub**](http://www.renotekno.net/2016/06/apa-itu-github-dan-apa-fungsi-dari.html)

[AndroidDev](http://www.renotekno.net/search/label/AndroidDev), [WebDev](http://www.renotekno.net/search/label/WebDev) 22:32[No Comments](http://www.renotekno.net/2016/06/apa-itu-github-dan-apa-fungsi-dari.html#comment-form)

Kali ini mimin akan membahas mengenai salah satu VCS terkenal yaitu GitHub. GitHub sekarang ini banyak digunakan bagi kalian yang bekerja secara kelompok ataupun sendiri. Bagi kalian yang belum mengetahui GitHub, berikut informasi mengenai GitHub beserta manfaatnya. GitHub termasuk VCS, jadi VCS dan GitHub sama saja.

**Apa Itu VCS**

VCS atau singkatan dari *Version Control System* merupakan sebuah media yang mengatur versi atau file-file dari project atau aplikasi yang kalian buat. Tempat untuk menyimpan file-file tersebut pada VCS disebut juga **repositories**. Contohnya aplikasi line memiliki versi:

* 6.3.2
* 6.3.1
* 6.3.0
* dst

Dengan menggunakan VCS maka perubahan pada tiap file akan bisa terlacak. Jadi semisal kalian merilis aplikasi terbaru kalian, dan kalian ingin melihat perubahan atau perbedaan apa saja yang dimiliki oleh versi terbaru dan versi sebelumnya. Nah disinilah VCS yang bertugas sebagai pembanding, karena dialah yang melacak setiap perubahan yang kita lakukan.

VCS banyak digunakan bagi para developer, baik web maupun android agar dapat memanage atau mengatur file-file mereka dan mencatat semua perubahan yang terjadi sehingga dapat menghemat waktu.

**Jenis-jenis VCS**

VCS terdapat 2 jenis yaitu **lokal** dan **online**. Maksudnya lokal adalah kalian dapat membuat repositories kalian sendiri pada komputer atau laptop kalian secara offline sehingga tidak membutuhkan koneksi internet. VCS lokal biasanya menggunakan command line untuk memanipulasi data. Terdapat juga beberapa VCS lokal yang menggunakan GUI yaitu GIT. Biasanya untuk VCS lokal para developer menggunakan [software Git](https://git-scm.com/downloads" \t "_blank).

Untuk VCS yang online tentunya untuk membuat repositories kalian membutuhkan koneksi internet. Terdapat banyak  VCS online di luar sana, contohnya:

1. **[GitHub](https://github.com/" \t "_blank)**
2. Bitbucket
3. CodePlex

**Fungsi Dari GitHub**

GitHub merupakan VCS online yang sangat bermanfaat bagi mereka yang bekerja secara team maupun sendiri. GitHub sekarang ini banyak digunakan oleh para developer untuk mengatur file-file project yang sedang mereka jalankan baik secara team maupun perorangan.

GitHub merupakan VCS yang sudah menggunakan GUI sehingga mempermudah dalam mengatur file-file. Terdapat juga beberapa fitur-fitur tambahan pada GitHub yang menarik sehingga *interface* nya sangat nyaman untuk digunakan. Mimin juga menggunakan GitHub dalam mengatur file-file project yang mimin buat.  
  
Untuk cara menggunakan GitHub kalian bisa membaca artikel [cara menggunakan GitHub dan membuat repository baru](http://www.renotekno.net/" \l "coming-soon" \t "_blank).

**Hidup Tanpa VCS**

Bayangkan jika kalian bekerja pada sebuah team dalam suatu project. ketika kalian melakukan perubahan dan team kalian tidak mengetahui maka project yang kalian bangun akan menjadi berantakan. Atau kalian bisa menginformasikan perubahan yang kalian buat melalui SMS, Line, atau media sosial lainya. Tetapi itu semua tidak efektif.

Dengan VCS online atau *remote repository*, para anggota team hanya dengan Login dan masuk pada repository maka mereka dapat melihat perubahan yang kalian lakukan, karena VCS memberikan notifikasi setiap perubahan yang dilakukan.

Perusahaan-perusahaan besar seperti teamtreehouse, facebook, dan twitter juga menggunakan VCS untuk mengatur atau mengelola file-file mereka. Hanya saja VCS yang mereka gunakan dibuat private jadi hanya orang-orang tertentu saja yang dapat melihat. Pada interview pekerjaan juga jika kalian bisa menggunakan VCS baik secara *command line* maupun GUI terutama GitHub maka kalian akan menjadi pilihan yang dipertimbangkan lebih dari yang lain yang tidak mengerti VCS.

# Membuat dan Menggunakan Repository di GitHub

# GitHub

Pada artikel kali ini saya akan mencoba membuat repository Git pada GitHub. Sebelumnya apa itu GitHub? Github adalah layanan yang menggunakan Git sebagai Version Control untuk mengelola dan memanajemen file. Dan ini sangatlah berguna dalam hal distribusi file, khususnya dalam dunia pemrograman. Berikut ini adalah tutorial dasar untuk mempermudah pekerjaan dengan mudah menggunakan [GitHub](http://www.github.com)**.**

### Step 1. Membuat repository baru di GitHub

Dalam menggunakan Version controll, anda akan selalu melakukan Commit pada Git, tempat tersebut dinamakan sebuah repository (a**.**k**.**a. “repo”). Dan untuk menyimpan project yang anda buat pada GitHub, anda perlu memiliki GitHub Repository. Berikut ini adalah cara untuk membuat repository pada GitHub.

Click [*New Repository*](https://github.com/repositories/new)***.***

Selanjutnya masukkan informasi tentang anda**.** jika sudah di isi, selanjutnya click “Create Repository.”

Selamat**!** Sampai pada tahap ini, anda sudah memiliki Repo

**Installasi Git pada komputer anda**

* Sistem operasi Linux Ubuntu

Jika anda pengguna Sistem operasi Linux Ubuntu, anda hanya mengetikkan perintah dibawah ini untuk melakukan instalasi Git pada komputer anda.

$ sudo apt-get install git

* Sistem operasi Windows

Jika anda pengguna Windows  anda harus mendownload aplikasi  github dari <http://windows.github.com/>

### Membuat file README untuk repo.

File  README bukanlah bagian yang diperlukan dari repo GitHub, tapi kita akan membuat file tersebut untuk uji coba apakah repo dapat digunakan. File README adalah catatan yang bagus untuk menggambarkan proyek Anda atau menambahkan beberapa dokumentasi seperti cara menginstal atau menggunakan proyek Anda. Anda mungkin ingin menyertakan informasi kontak person – jika proyek Anda menjadi orang yang populer akan ingin membantu Anda suatu saat.

#### 

sekarang masuklah ke command prompt, lalu buat folder sesuai nama project :

mkdir ~/Hello-World #Membuat direktori "Hello-World"

setelah itu, masuk ke folder tersebut:

*cd ~/Hello-World #Masuk ke folder yang anda buat*

setelah masuk, ketik perintah tersebut untuk meng initialize folder tersebut:

*git init #Sets up the necessary Git files*

setelah masuk, ketik perintah tersebut untuk meng initialize folder tersebut:

*touch README# Creates a file called "README" in your Hello-World directory*

nah sekarang kita coba file README  di folder Hello-World tersebut menggunakan teks editor anda. tambahkan tulisan “Hello World!” di file tersebut, lalu di simpan **.**

#### Step 2. Commit file README

Sekarang file README sudah siap untuk di commit. Untuk meng -commit dibutuhkan snapshot dari semua file di project anda pada waktu yang sama**.** pada command prompt, ketikan perintah:

git add README #untuk menambahkan file README

git commit -m 'first commit' #Commit file anda, dengan menambahkan pesan "first commit"

#### Step 3: Push  commit anda

Sampai pada step ini, anda telah melakukan commit pada lokal repository anda, tapi masih belum melakukan apapun pada repo GitHub**.**

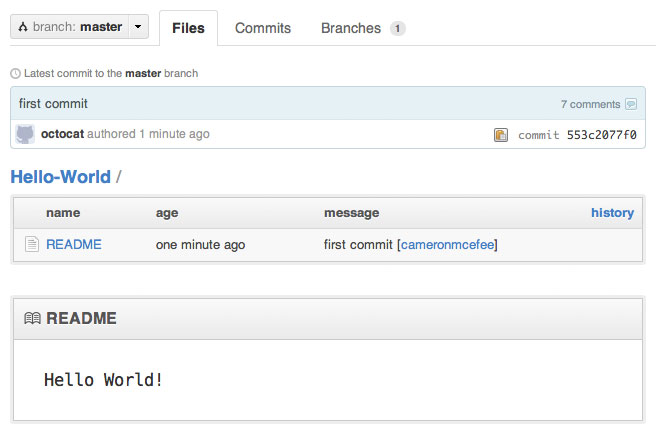
Untuk menghubungkan lokal koneksi anda ke GitHub, anda perlu me remote repo dan melakukan push pada commit  anda.

Masukkan perintah ini untuk melakukannya :

*git remote add origin https://github.com/namauser/Hello-World.git # membuat remote dengan nama "origin" pointing pada GitHub repo*

*git push origin master# Mengirimkan perintah commit sebagai "master" branch pada GitHub*

Sekarang anda dapat melihat repository anda di GitHub, anda akan dapat melihat file README ada di repo **.**



### Selamat…

Congratulations**!** anda sudah membuat sebuah repository di GitHub, membuat README, meng commit nya , dan push di GitHub**.**

**Seluruh Command**

### membuat repo menggunakan command

*touch README.md*

*git init*

*git add README.md*

*git commit -m "first commit"*

*git remote add origin https://github.com/username/Hello-World.git*

*git push -u origin master*

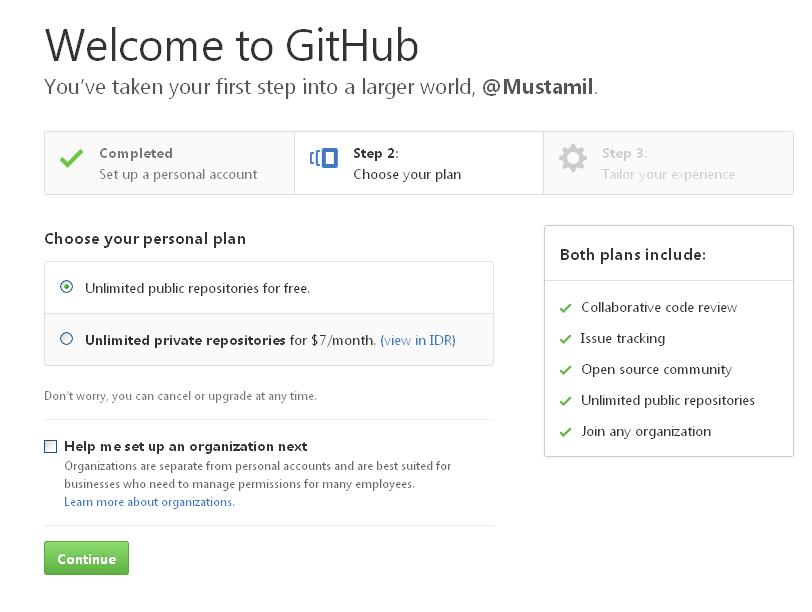
### *Push  repository dari command*

*git remote add origin https://github.com/usernam/Hello-World.git*

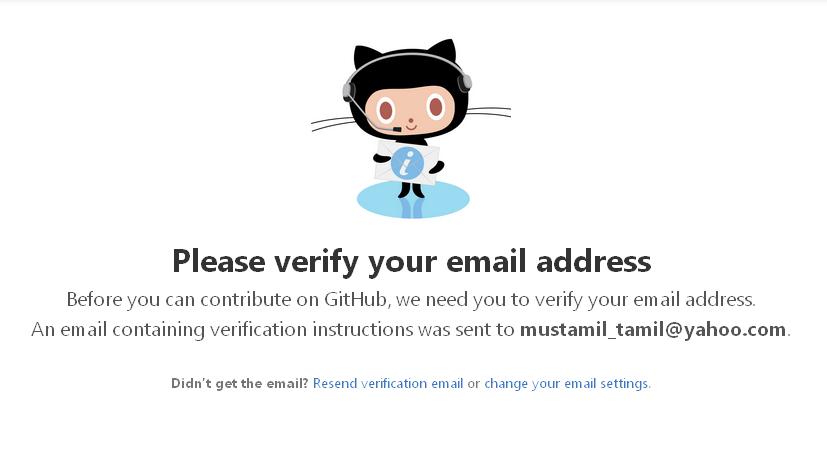
*git push -u origin master*

**Reference**

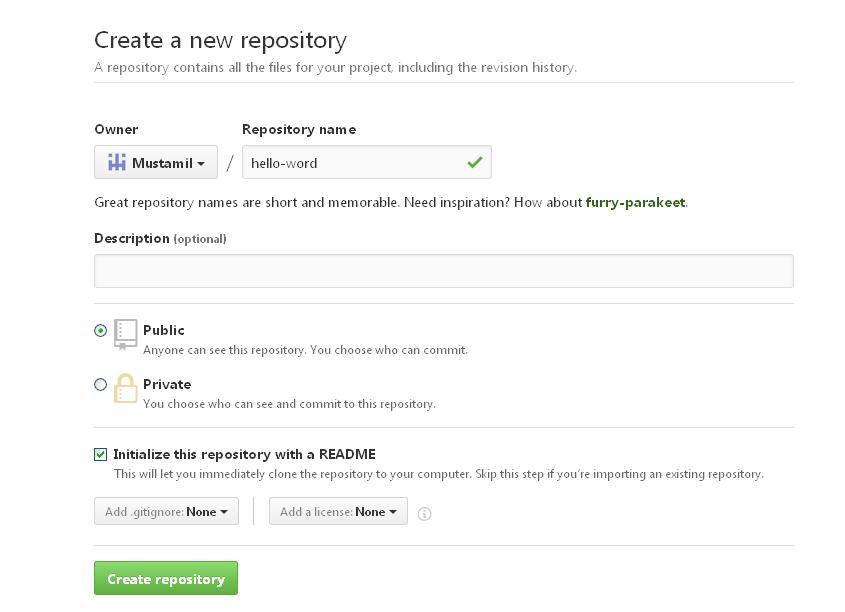
<https://help.github.com/articles/create-a-repo>



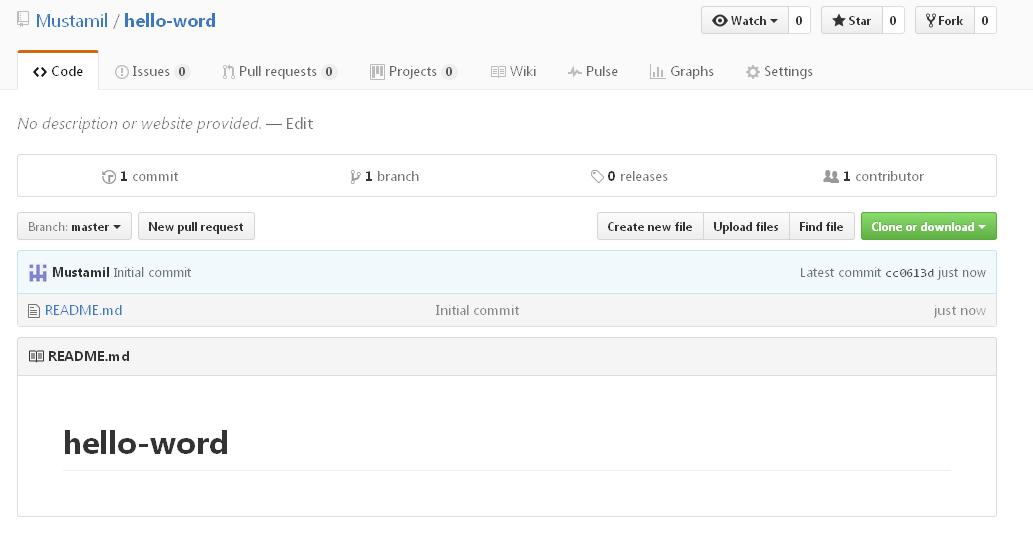
Lalu Pilih continue dan mengonfirmasi melalui email



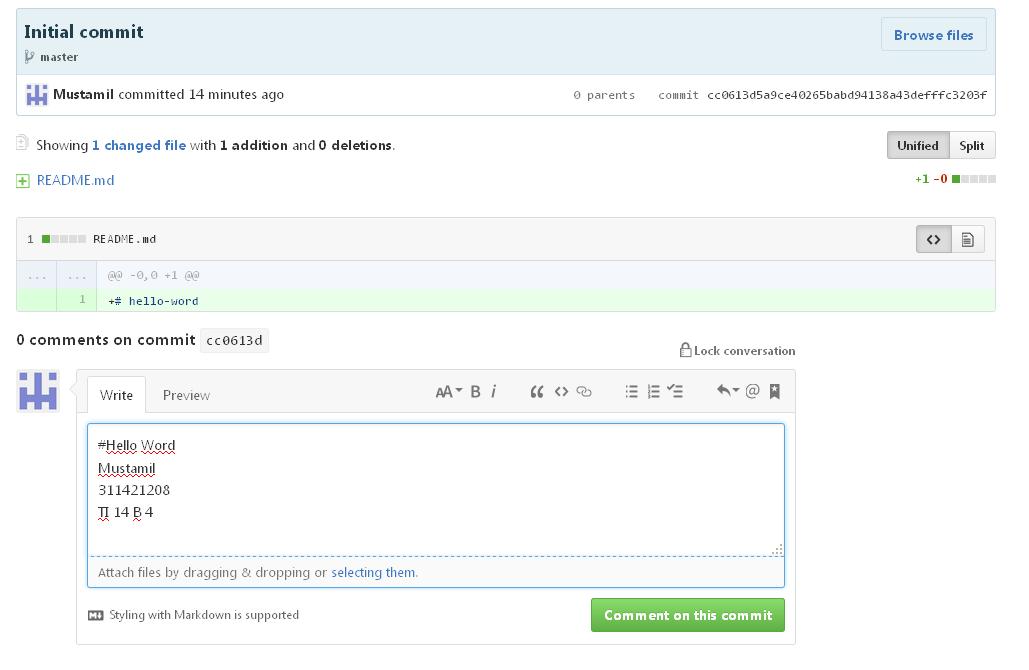
Membuat repository : mengisi repository name pilih public dan pilih Create repository



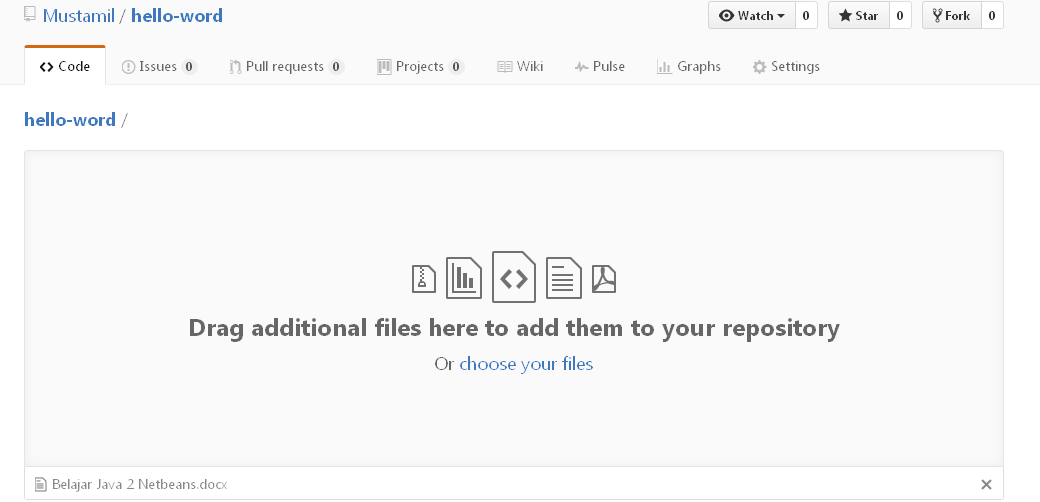
Lalu mengupload file project ke repository



Mengedit commit



Meng upload file dari computer ke GITHUB



Hasil upload data ke github

