

Pengertian Repository:

Secara sederhana arti dari kata repository adalah tempat penimbunan atau tempat penyimpanan. Secara etimologi, repository dapat diartikan sebagai tempat untuk menyimpan (archiving). Repository adalah tempat disimpannya berbagai macam program atau aplikasi yang telah dibuat sedemikian rupa sehingga bisa diakses melalui internet. Repository tidak hanya dapat diakses melalui internet saja tetapi kita juga dapat menggunakan alternatif repository lewat distribusi pada media lain seperti DVD yang tentunya sangat membantu sekali buat kita yang tidak memiliki koneksi internet yang cepat.

Repository adalah tempat penyimpanan dari kumpulan software atau aplikasi yang bisa didownload untuk digunakan atau dapat diartikan sebuah arsip software yang ada pada media penyimpanan atau internet. Bergantung pada apakah sistem kendali versi yang digunakan itu didistribusikan (misalnya, Git atau Mercurial) atau terpusat (misalnya, Subversion atau Perforce). Seluruh rangkaian informasi dalam repository dapat diduplikasi pada sistem setiap pengguna atau dapat dipelihara pada satu peladen. Beberapa metadata yang ada di repository antara lain adalah:

- Catatan riwayat perubahan dalam repository.
- Sekumpulan objek komit (commit)
- Kumpulan referensi untuk objek komit yang disebut heads.

Dengan kata lain repository ini adalah paket-paket khusus untuk sebuah sistem operasi yang kemudian paket-paket tersebut di instal untuk mendapatkan kinerja lebih baik dari sebuah sistem operasi. Di dalam konteks kepustakawanan repository merupakan suatu tempat dimana dokumen, informasi atau data disimpan, dipelihara dan digunakan. Kadang-kadang istilah repository dipakai untuk menyatakan hal yang sama.

- Menurut pendapat Reitz (2004) menyatakan bahwa “Repository is the physical space (building, room, area) reserved for permanent or intermediate storage of archival materials (manuscripts, rare books, government documents, papers, photographs etc)”. artinya Repository adalah ruang fisik (gedung, ruangan, area) yang disediakan untuk penyimpanan permanen atau menengah dari bahan arsip (manuskrip, buku langka, dokumen pemerintah, kertas, foto, dll). Perpustakaan sebenarnya adalah sebuah repository akan tetapi dalam ruang lingkup yang lebih luas.
- Menurut pendapat Pfister (2008) mengatakan bahwa perguruan tinggi yang berbasis repository adalah satu set layanan yang menawarkan berbagai bahan digital yang

dihasilkan oleh lembaga tersebut ataupun yang dihasilkan lembaga lain yang dikelolanya kepada masyarakat penggunaanya

- Menurut Clifford Lynch (2003) mendefinisikan repository pada perguruan tinggi adalah serangkaian pelayanan yang diberikan oleh perguruan tinggi kepada anggota komunitasnya untuk mengelola dan menyebarkan bahan-bahan digital yang dihasilkan oleh institusi tersebut.

Alasan Dibangunnya Repository:

Ada berbagai alasan untuk membangun repository. Pfister (2008) mengemukakan sedikitnya ada tiga alasan membangun repository diantara adalah sebagai berikut:

- Peningkatan visibilitas dan dampak dari output penelitian. Para peneliti dan lembaga mendapatkan manfaat dari repository dalam cara yang sama yaitu mengetahui kejelasan dan dampak dari hasil penelitian. Membangun dan mempertahankan reputasi dalam komunitas ilmiah sangat penting bagi kegiatan akademik dan institusi dan hal itu dapat dicapai dengan repository.
- Berkaitan dengan perubahan dalam paradigma publikasi ilmiah. Munculnya gerakan untuk menyediakan akses gratis terhadap publikasi ilmiah. Content ilmiah dihasilkan dan dipublikasikan sendiri dan penyediaan akses gratis terhadap bahan-bahan tersebut adalah merupakan aktivitas utama dalam gerakan akses terbuka (open access movement). Suatu perguruan tinggi akan lebih leluasa memberikan akses terbuka terhadap bahan-bahan yang mencerminkan kekayaan intelektual dari perguruan tinggi itu sendiri adalah melalui pendirian repository.
- Membangun repository adalah didasarkan atas kemungkinan perbaikan komunikasi internal. Dengan menyediakan penyimpanan bahan-bahan digital secara terpusat akan mendapatkan manfaat dari bahan yang telah dipublikasikan pada satu sisi, dan pada sisi yang lain menjadi dasar untuk mengetahui bahan-bahan yang belum dipublikasikan secara digital. Sehingga repository menjadi salah satu upaya untuk mendorong agar bahan-bahan lain yang bukan kategori ilmiah seperti laporan kegiatan, panduan dan sebagainya untuk dipublikasikan dalam format digital, karena bahan-bahan tersebut juga merupakan bagian dari pengetahuan organisasi dan sebaiknya dapat diakses oleh setiap orang dalam suatu organisasi. Repository mendorong upaya digitalisasi terhadap dokumen-dokumen perguruan tinggi yang bukan kategori ilmiah, sehingga akses terhadap dokumen tersebut lebih mudah.

Keuntungan Membangun Repository

Membangun sebuah repository akan menghasilkan keuntungan baik untuk individu maupun untuk lembaga. Hasil dari penelitian-penelitian, artikel ilmiah, makalah, tesis, disertasi dan karya ilmiah lainnya yang tersedia secara online dapat diakses, di download, dan atau dikutip lebih cepat dan lebih sering dibanding dengan yang tersedia dalam format tercetak. Repository pada perguruan tinggi menjadi sarana penting untuk mempublikasikan penelitian dan karya-karya akademik yang dimilikinya. Reputasi perguruan tinggi akan semakin dikenal melalui peran repositor. Karya akademik perguruan tinggi tidak hanya tersebar melalui jurnal, akan tetapi dapat juga melalui repository, sehingga akan meningkatkan visibilitas dan prestise.

Tujuan Repository

Tujuan utama repository adalah untuk menyimpan sekumpulan berkas dan juga riwayat perubahan pada berkas tersebut. Banyak perbedaan sistem kendali versi dalam menangani penyimpanan perubahan: misalnya, Subversion dulu mengandalkan pangkalan data dan sekarang pindah menyimpan perubahan langsung ke sistem berkas. Perbedaan-perbedaan metode ini membuat bermacam-macam kendali versi digunakan oleh kelompok-kelompok yang berbeda, tergantung pada kebutuhan mereka. Jadi intinya yang dinamakan repository itu adalah berbagai macam program atau aplikasi yang sudah dibuat sedemikian rupa sehingga kita mengakses program tersebut secara online.

Repository institusi dipandang sebagai suatu yang sangat penting, sistem yang memudahkan akses terhadap informasi ilmiah dan sebagai salah satu pendorong munculnya karya ilmiah yang bermutu. Akses terbuka (open access) adalah merupakan ciri dari repository institusi. Aplikasi akses terbuka ini dapat diatur lebih lanjut oleh pengelola repository dengan kontributor bahan repository (dokumen). Untuk itu, ke depan perlu ditetapkan pengelola yang khusus menangani repository institusi ini.

Version Control System (VCS)

Version Control System (VCS) disebut juga pengontrol versi atau pengontrol revisi, merupakan suatu sistem pengelolaan berbagai revisi atas perubahan dari suatu unit informasi baik berupa dokumen, source code, ataupun informasi lainnya yang disimpan dalam media penyimpanan komputer. Setiap entitas perubahan biasanya diidentifikasi dalam bentuk kode numerik ataupun serangkaian karakter yang diistilahkan sebagai "nomor revisi" atau

sederhananya disebut "revisi". Misalnya sekumpulan berkas yang disertakan saat inisialisasi disebut sebagai "revisi 1", kemudian saat terjadi perubahan atas satu atau beberapa berkas yang ada disebut sebagai "revisi 2", dan seterusnya. Tiap-tiap revisi diasosiasikan dengan stempel waktu serta pengguna yang melakukan perubahan tersebut. Sebuah revisi dapat diperbandingkan, dikembalikan, ataupun digabungkan dengan beberapa jenis berkas yang lain.

Kebutuhan untuk mengelola revisi atas perubahan yang dilakukan pada dasarnya telah lama muncul bersamaan dengan saat kemampuan menulis dikenal oleh manusia, hanya saja kebutuhan tersebut menjadi sangat penting seiring dengan dimulainya era komputasi. "Edisi" yang biasanya terdapat pada buku misalnya, merupakan contoh umum penggambaran atas manajemen revisi pada dunia percetakan.

Version Control System (VCS) atau sistem kendali versi umumnya merupakan aplikasi mandiri, namun ada pula yang berfungsi sebagai sub modul dalam berbagai jenis aplikasi seperti pengolah kata (misalnya: OpenOffice Writer, Microsoft Word, KOffice), lembar kerja (spreadsheet) (misalnya: OpenOffice Calc, Microsoft Access), dan berbagai macam sistem manajemen konten. Integrasi terhadap kendali revisi ini merupakan salah satu fitur penting yang dimiliki perangkat lunak wiki seperti DokuWiki, TWiki, dan lain-lain. Dalam wiki, keberadaan kendali revisi memungkinkan pengembalian artikel kembali ke revisi sebelumnya yang sangat penting bagi penulis, ataupun kontributor untuk melacak perubahan yang dilakukan oleh pengguna lainnya, mengoreksi kesalahan, dan mempertahankan wiki terbuka dari vandalisme, ataupun spam.

Perangkat lunak pembantu untuk mengendalikan revisi saat ini semakin lebih dikenal karena keberadaannya semakin diperlukan utamanya oleh organisasi yang memiliki tim pengembang, yang melibatkan lebih dari satu orang penulis kode dalam pengerjaan project perangkat lunak. Sistem ini utamanya digunakan dalam proses pengembangan perangkat lunak dimana tiap anggota dalam tim pengembang dapat saling melakukan perubahan terhadap sebuah berkas yang sama. Dalam dunia rekayasa perangkat lunak, pengelolaan revisi merupakan bagian dari praktik yang dilakukan untuk menelusuri serta mengendalikan setiap perubahan yang dilakukan terhadap source code. Pengembang perangkat lunak terkadang menggunakan pula sistem ini untuk manajemen perubahan atas dokumen ataupun berkas-berkas konfigurasi seperti halnya source code.

Saat sebuah tim mendesain, mengembangkan, dan mendistribusikan sebuah perangkat lunak adalah hal yang umum, beberapa versi dari sebuah perangkat lunak yang sama didistribusikan di beberapa situs yang berbeda, dan terhadap para pengembang perangkat lunak pun, pemanfaatan pengelolaan revisi memungkinkan mereka untuk secara simultan melakukan perbaikan kesalahan, penambahan fitur, serta mendistribusikan versi yang lebih baru.

GIT

Git adalah sebuah software atau VCS terdistribusi untuk mengelola perubahan file didalam folder (repository atau repo). Git merupakan alat menyimpan riwayat perubahan file yang disimpan menggunakan serangkaian commit. Git adalah perangkat lunak pengendali versi atau project manajemen kode perangkat lunak yang diciptakan oleh Linus Benedict Torvalds seorang rekayasawan perangkat lunak Finlandia yang dikenal sebagai perintis pengembangan Kernel Linux. Git pada awalnya ditujukan untuk pengembangan kernel Linux, desain Git terinspirasi oleh BitKeeper dan Monotone. Git pada awalnya hanya dirancang sebagai mesin tingkat rendah yang dapat digunakan oleh tampilan muka (*front end*) lain seperti Cogito atau StGIT. Namun selanjutnya project inti Git telah berkembang menjadi pengendali revisi lengkap yang dapat digunakan langsung.

#GITHUB

GitHub adalah sebuah website atau layanan hos web bersama untuk project pengembangan perangkat lunak yang menggunakan sistem kendali versi Git dan layanan hosting internet. Hal ini banyak digunakan untuk kode komputer.

Istilah didalam GIT:

- Repo adalah folder project
- Commit adalah rekaman
- Hash adalah penanda unik
- Checkout adalah berpindah
- Branch adalah cabang
- Merga adalah menggabung branch
- Remote adalah Sumber
- Clone adalah mengambil repo dari remote
- Push adalah mengirim commit
- Pull adalah mengambil commit

PENGENALAN GITHUB

GitHub adalah sebuah layanan repository untuk menyimpan project secara online, project yang kita upload di github bersifat open-source yang bisa dikembangkan oleh programmer lain. Kelebihan Github adalah ketika kita membuat sebuah repository project, kemudian menguploadnya lalu kita membuat perubahan atau penambahan pada project tersebut dan di upload ulang ke repository itu maka project yang lama tidak akan hilang, jadi kita bisa membuat beberapa versi dari project yang dibuat dan hal ini sangat memudahkan bila kita ingin ke versi sebelumnya. Ini memberikan kontrol akses dan beberapa fitur kolaborasi seperti pelacakan bug, permintaan fitur, manajemen tugas untuk setiap project. GitHub menawarkan paket repository pribadi dan gratis pada akun yang sama dan digunakan untuk project perangkat lunak open source.

GitHub merupakan layanan repository git yang sangat besar dan telah dikenal luas. Banyak varian repository yang bisa diakses secara gratis sehingga menjadikan GitHub menjadi terkenal dengan layanan penyimpanan source code. Jika project kita bersifat open source, GitHub bisa menjadi pilihan yang tepat. Namun kelemahannya, jika kita menginginkan privasi untuk project, kita harus membayar GitHub. Pada bulan April 2017, GitHub melaporkan bahwa mereka mempunyai lebih dari 20 juta pengguna dan lebih dari 57 juta repository, menjadikannya layanan terbesar dari source code di dunia. Pada tahun 2018 dilaporkan bahwa Microsoft mengakuisisi GitHub.

Perintah Dasar Github

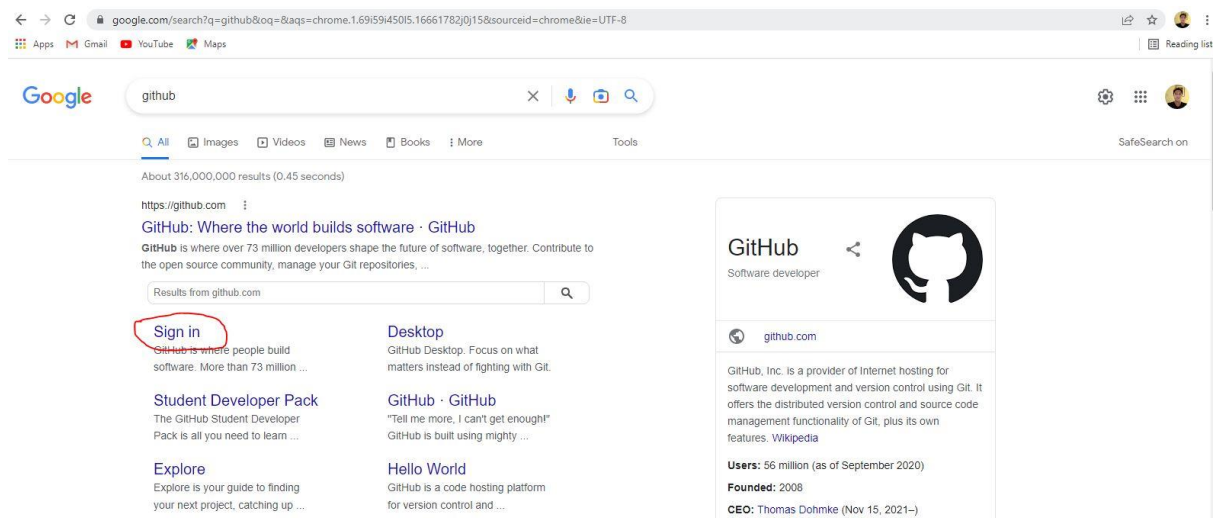
Ada beberapa perintah dasar yang biasanya digunakan, perintah dimulai dari kata “git” :

- **git init** untuk membuat repository lokal di dalam folder project
- **git clone** meng-clone atau meng-copy projek dari repository
- **git status** untuk menampilkan status pada repository lokal
- **git add** menambah file baru pada repository yang dipilih
- **git commit** menyimpan perubahan, dan setiap perubahan ini wajib memberikan keterangan pada setiap perubahan
- **git push** mengirimkan perubahan file setelah di commit ke repository
- **git pull** mengambil file yang sudah di ubah dan di upload
- **git branch** melihat branch yang tersedia pada repository
- **git merge** menggabungkan semua branch yang ada pada repository.

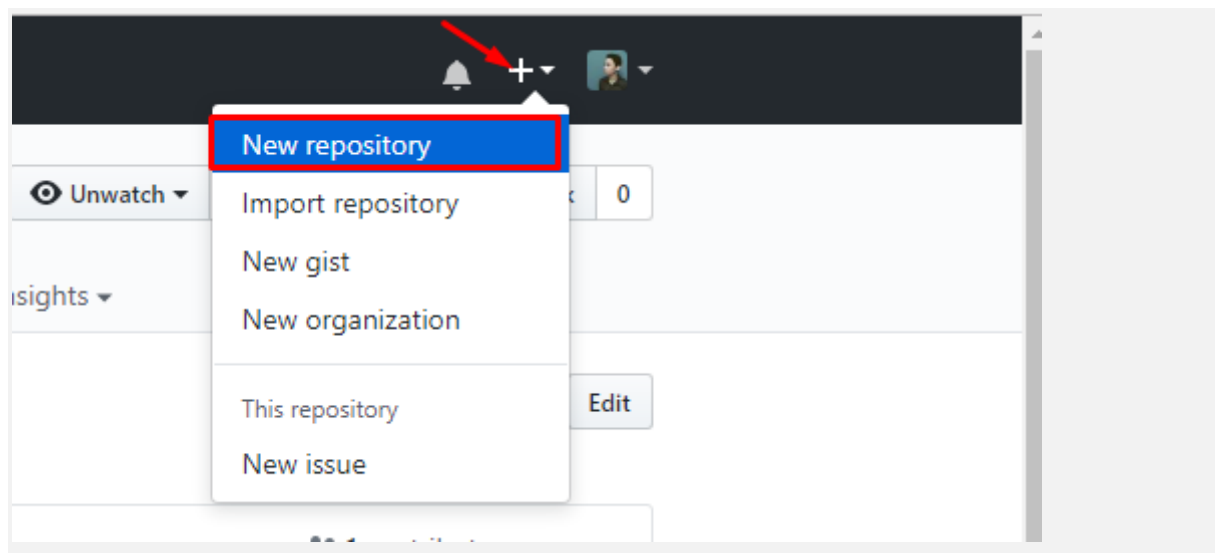
Cara upload project di github secara public

Berikut langkah-langkahnya:

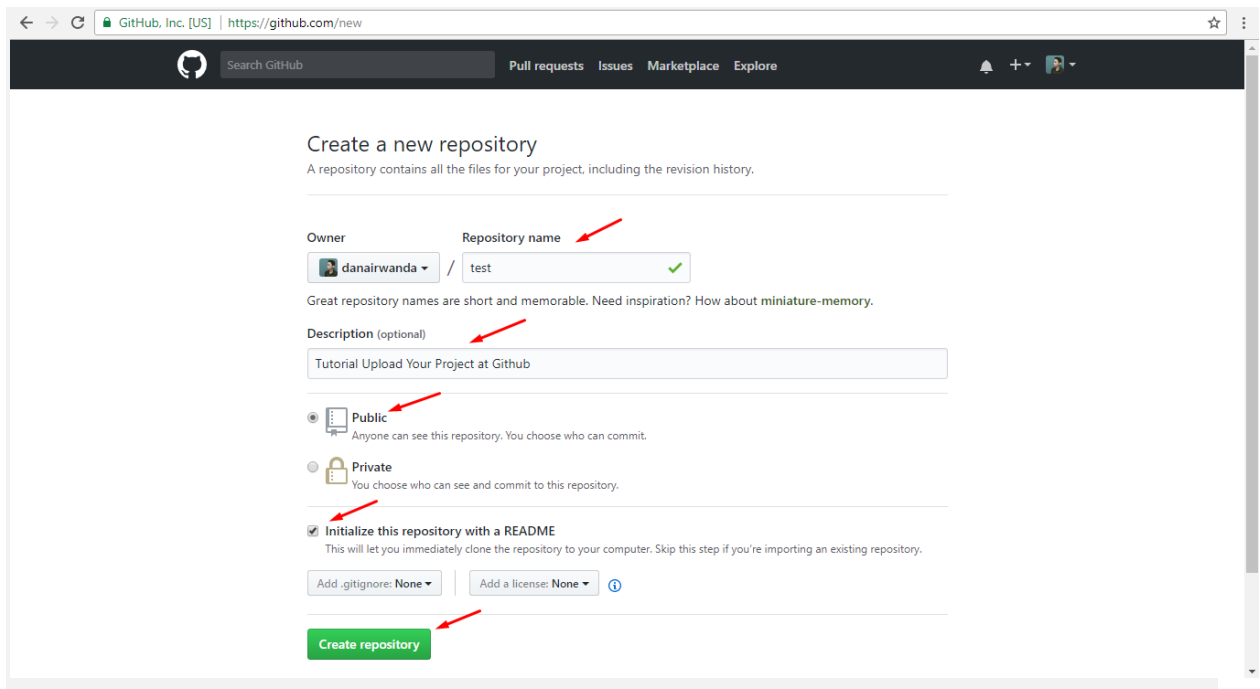
1. Silahkan buat akun terlebih dahulu di *github.com*



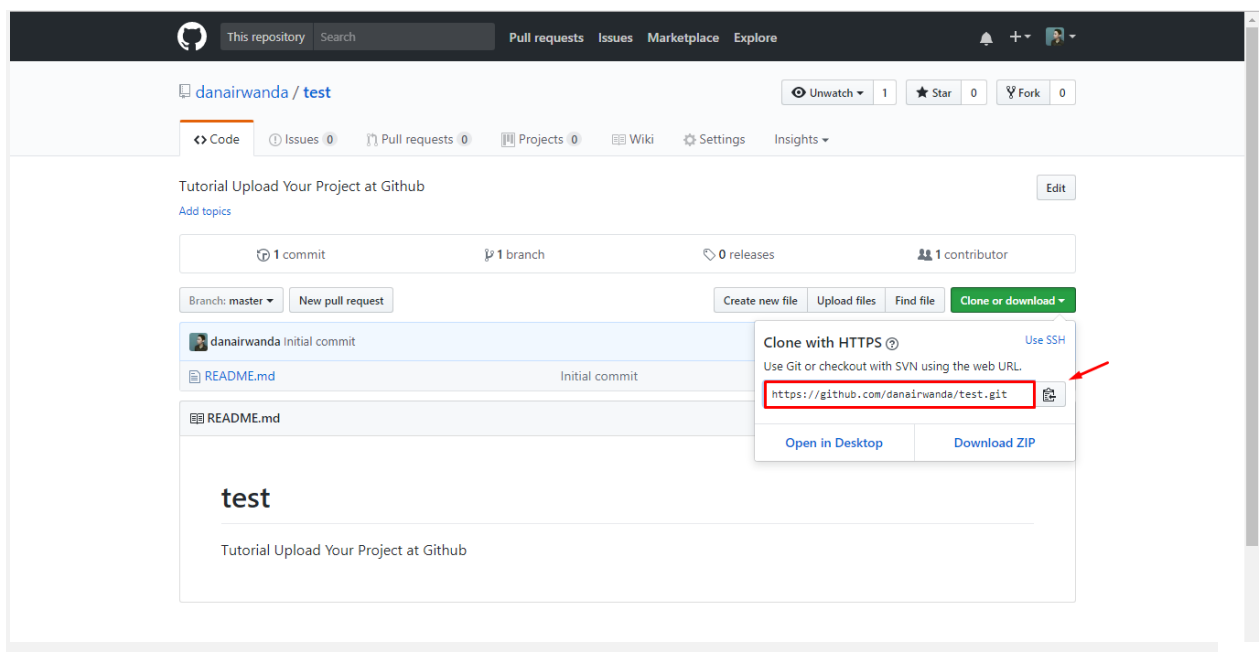
2. Download software Git disini. Sesuaikan dengan sistem operasi yang digunakan, pada tutorial ini kami menggunakan Windows
3. Lakukan instalasi Git seperti software pada umumnya
4. Login ke Github.com dan buatlah sebuah repository baru dengan mengeklik tombol yang terletak pada kanan atas seperti gambar berikut:



5. Buat repository baru, misalnya dengan nama “test” kemudian klik tombol *Create repository* lihat gambar:

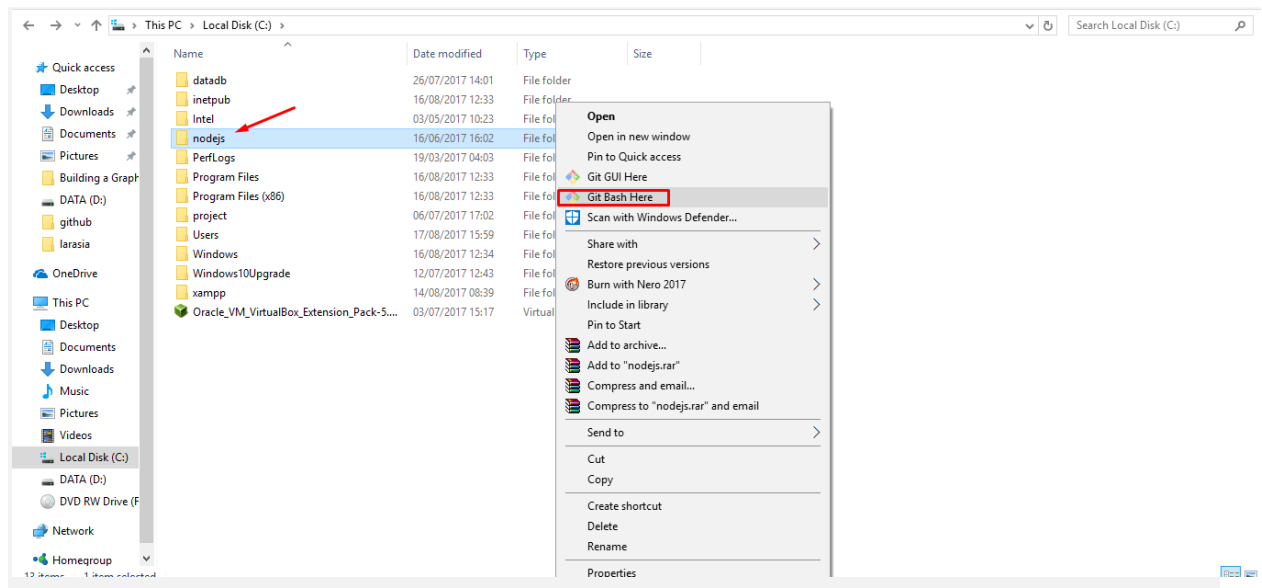


6. Sekarang kita bisa mengakses remote repository dengan url `https://github.com/IchsanAlisan/test.git`



7. Arahkan pada direktori tempat project akan diupload ke repository di Github

8. Untuk pengguna Windows klik kanan folder project dan pilih Git Bash. Berikut gambarnya:



9. Kemudian akan muncul command prompt / CMD

10. Jika baru pertama kali menggunakan software Git, sebaiknya konfigurasi username dan email dulu
 Git config --global user.name "username anda"
 Git config --global user.email "email anda"

```
PS-TM@DESKTOP-U3VRCC9 MINGW64 /c/nodejs (master)
$ Git config --global user.email "danairwanda@icloud.com"
```

11. Setelah melakukan konfigurasi username dan email, kemudian lakukan inisiasi dengan menulis perintah berikut

Git init

```
PS-TM@DESKTOP-U3VRCC9 MINGW64 /c/nodejs
$ git init
Initialized empty Git repository in C:/nodejs/.git/
```

12. Kemudian tambahkan semua file yang ada dalam folder project kita, ketikan
 Git add *

```
PS-TM@DESKTOP-U3VRCC9 MINGW64 /c/nodejs (master)
$ git add *
warning: LF will be replaced by CRLF in node_modules/bignumber.js/package.json.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/core-util-is/LICENSE.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/core-util-is/README.md.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/core-util-is/float.patch.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/core-util-is/lib/util.js.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/core-util-is/package.json.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/core-util-is/test.js.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/inherits/LICENSE.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/inherits/README.md.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/inherits/inherits.js.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/inherits/inherits_browser.js.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/inherits/package.json.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in node_modules/isarray/README.md.
```

13. Buat commit projectnya, sebagai contoh disini commit “versi 1.0”

Git commit -m "test"

```
PS-TM@DESKTOP-U3VRCC9 MINGW64 /c/nodejs (master)
$ git commit -m "test"
```

14. Setelah kita buat commit untuk projectnya, remote repository yang telah dibuat tadi, contoh <https://github.com/danairwanda/test.git>,

Ketikan Git remote add origin <https://github.com/danairwanda/test.git>

```
PS-TM@DESKTOP-U3VRCC9 MINGW64 /c/nodejs (master)
$ Git remote add origin https://github.com/danairwanda/test.git
```

15. Setelah meremote repository, kemudian kita pull projectnya dengan perintah

Git pull origin master

```
PS-TM@DESKTOP-U3VRCC9 MINGW64 /c/nodejs (master)
$ Git pull origin master
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/danairwanda/test
* branch                master       -> FETCH_HEAD
* [new branch]          master       -> origin/master
```

16. Kirim project ke repository kita

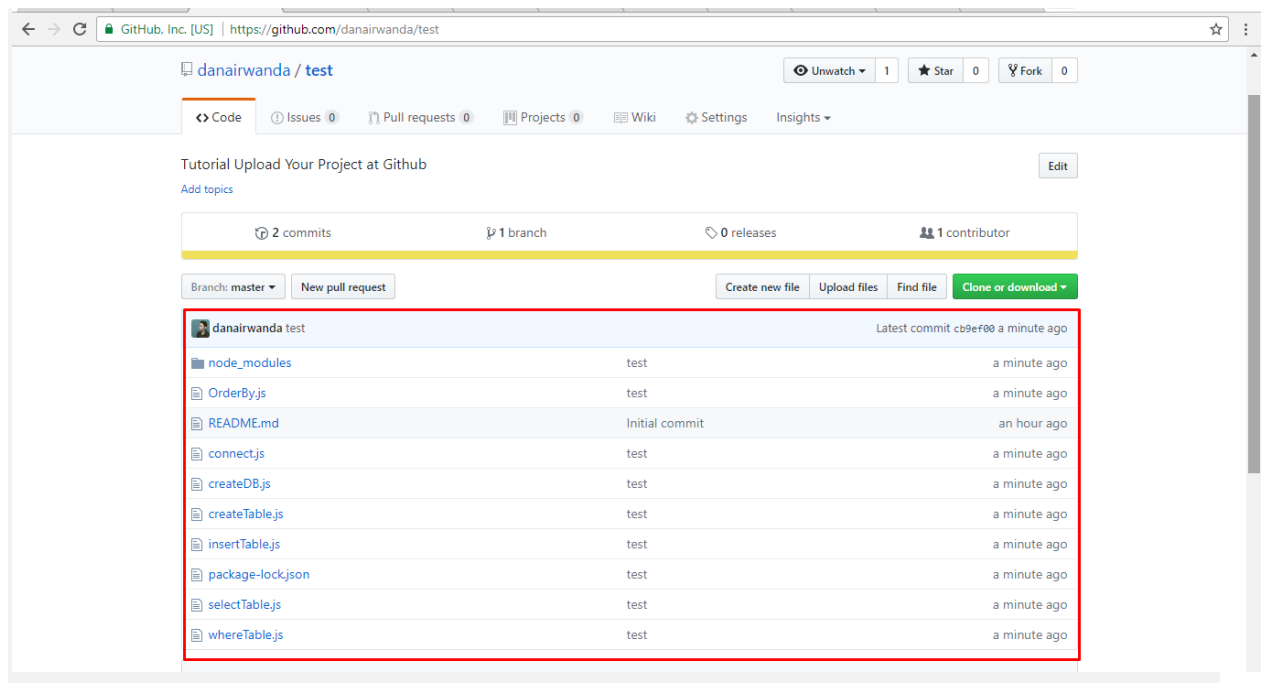
Git push origin master

```
PS-TM@DESKTOP-U3VRCC9 MINGW64 /c/nodejs (master)
$ git push origin master
Counting objects: 135, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (127/127), done.
Writing objects: 100% (135/135), 214.30 KiB | 0 bytes/s, done.
Total 135 (delta 14), reused 0 (delta 0)
remote: Resolving deltas: 100% (14/14), done.
To https://github.com/danairwanda/test.git
1defd91..cb9ef00 master -> master
```

17. Terakhir, kita akan diminta username dan password kita. Untuk password mungkin kita tidak akan melihat password yang kita ketikan

18. Tunggu sampai project selesai di upload

19. Jika proses upload selesai, cek pada Github lalu refresh repository, maka file kita sudah terupload disana



The screenshot shows the GitHub repository page for 'danairwanda / test'. The repository has 2 commits, 1 branch, 0 releases, and 1 contributor. The file list is as follows:

File	Commit	Time
node_modules	test	a minute ago
OrderBy.js	test	a minute ago
README.md	Initial commit	an hour ago
connect.js	test	a minute ago
createDB.js	test	a minute ago
createTable.js	test	a minute ago
insertTable.js	test	a minute ago
package-lock.json	test	a minute ago
selectTable.js	test	a minute ago
whereTable.js	test	a minute ago