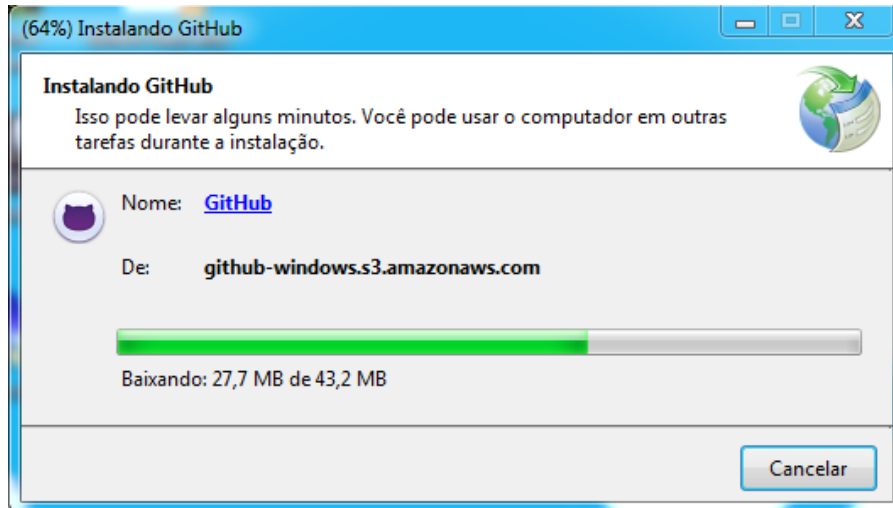


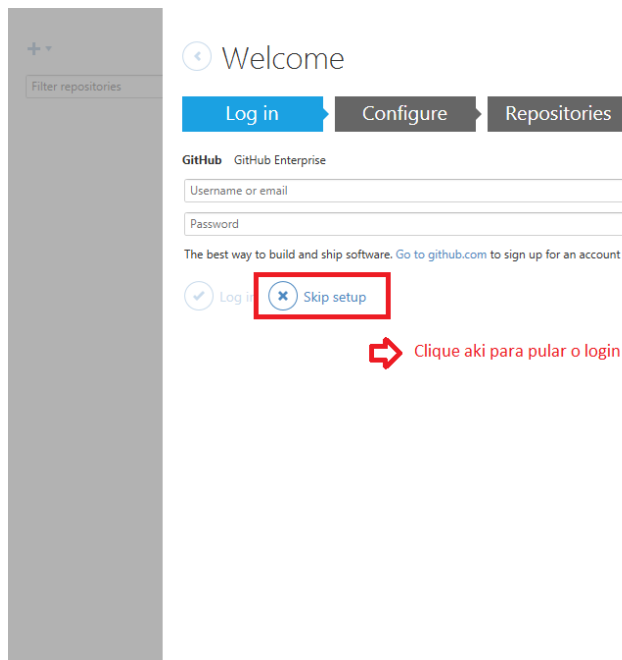
# Tutorial GitHub – Windows

## Preparando o Ambiente:

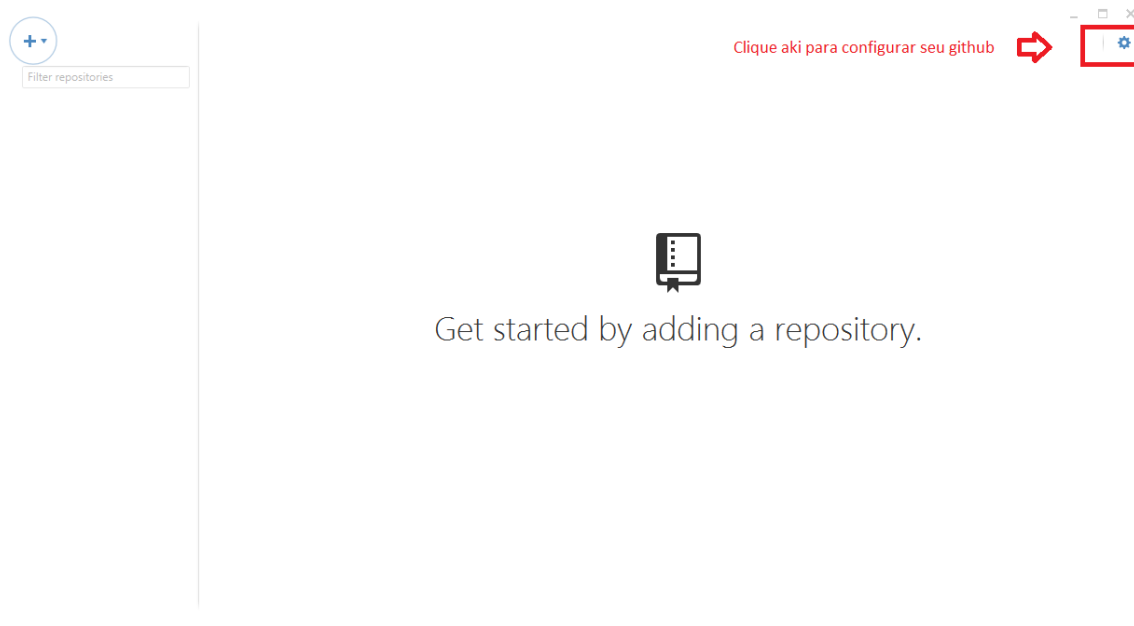
- 1 – Entre no site <https://windows.github.com/> e baixe.
- 2 – Instale o github no seu computador.



- 3 – Após a instalação, clique em “**Skip setup**” para pular as configurações de **login**.

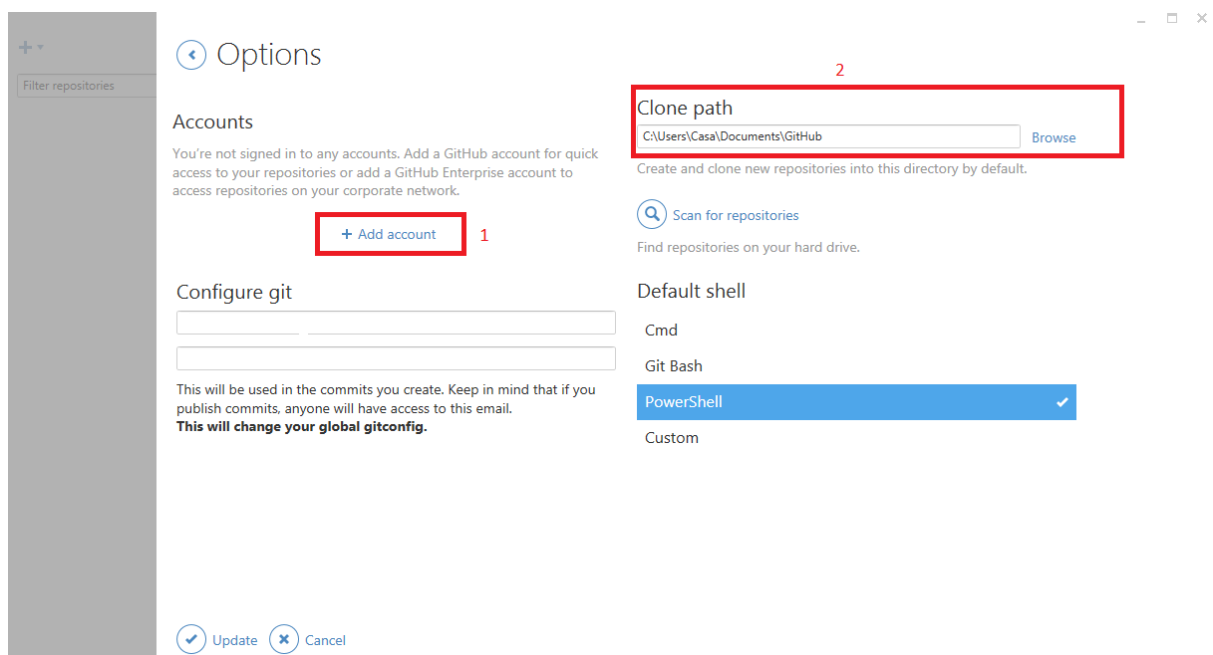


4 – Clique em “**Opções**” para configurar **manualmente** seu github.



5 – Insira seu **login** e senha, e altere o caminho padrão do github, este caminho ficará seu projeto na sua máquina.

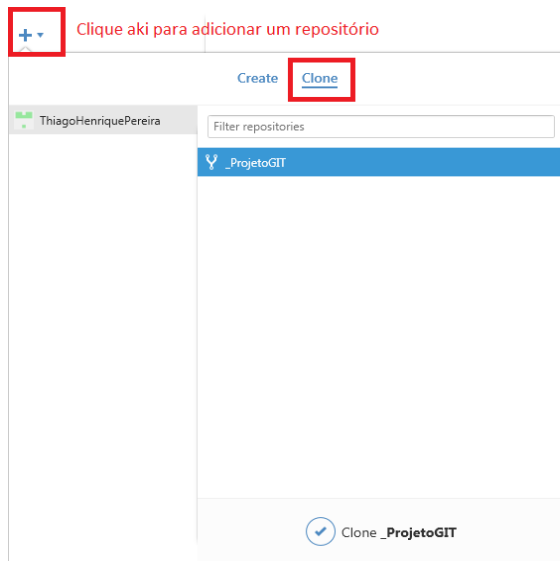
**Obs.:** Se estiver utilizando o “Apache” ou outro servidor qualquer, este caminho deverá ser o www do seu servidor, por **exemplo**, “**C:\Program Files\wampserver\www**”, este é meu caminho do wampserver.



6 – Abra seu navegador e entre no github <https://github.com/> , após efetuar **login** no github, escolha o repositório desejado e clique em **“FORK”**, feito isso, volte para o github do desktop e faça o clone do repositório.

6.1 – Clique no **“mais +”** para adicionar um repositório

6.2 – Escolha a aba **“Clone”** e selecione o repositório do projeto.



ted by adding a repository.

7 – Após realizar todos esses passos, seu ambiente para o desenvolvimento está configurado **corretamente**.

## Iniciando o desenvolvimento:

Após a configuração do seu ambiente, vamos começar a o desenvolvimento.

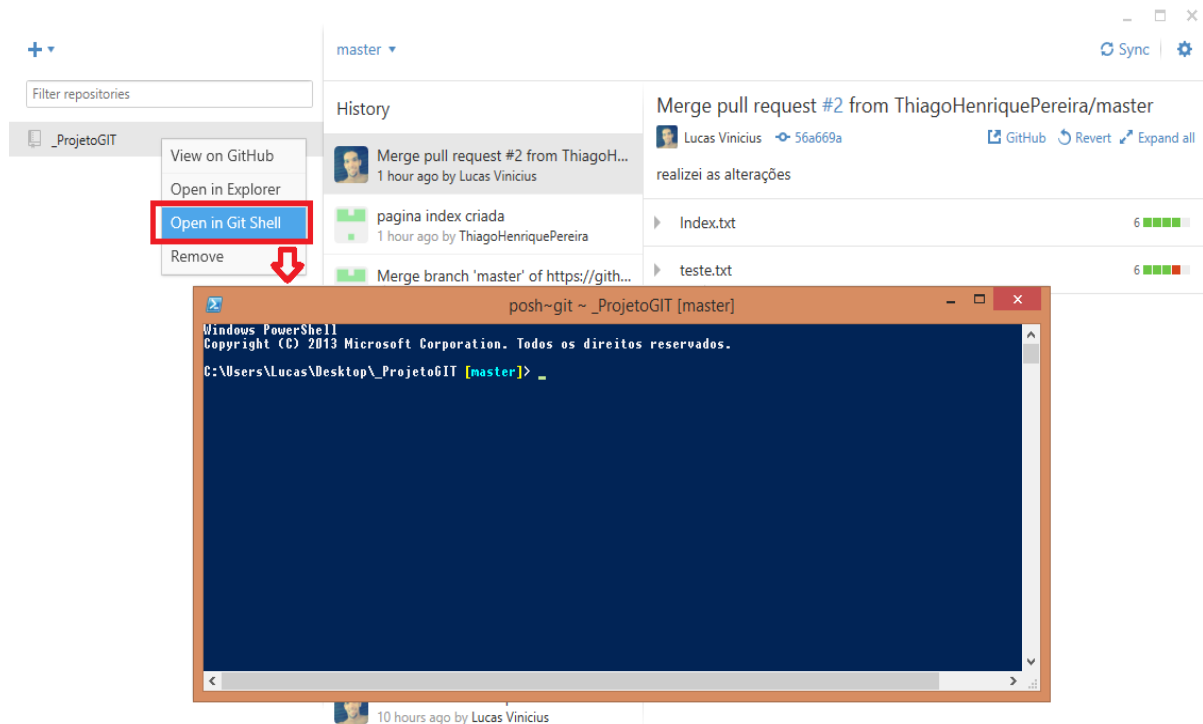
Antes de começar a desenvolver, é importante saber como funcionar o github.

Mais, por que utilizar o github (controle de versão)?

- 1 – Evita ter que ficar mandando arquivos '.zip' para outras pessoas.
- 2 – Mantem o código centralizado.
- 3 – Todos tem acesso ao projeto, não precisa mais centralizar em só uma pessoa.
- 4 – Todos integrantes do grupo desenvolvem.  
HAHAHAHAHA.

***Agora sim, mãos à massa.***

1 – Lembre-se toda vez que for 'mexer' no **código**, lembre-se que pode ter outra pessoa mexendo no mesmo código que você. Então é **importantíssimo** que você atualize seu repositório, para isso siga as instruções abaixo.



**Conforme a imagem acima:**

1.1 – Clique com botão direito no seu repositório, e escolha **“open in git Shell”**.

1.2 – Aparecerá uma tela preta semelhante à essa da imagem acima, digite o seguinte comando:

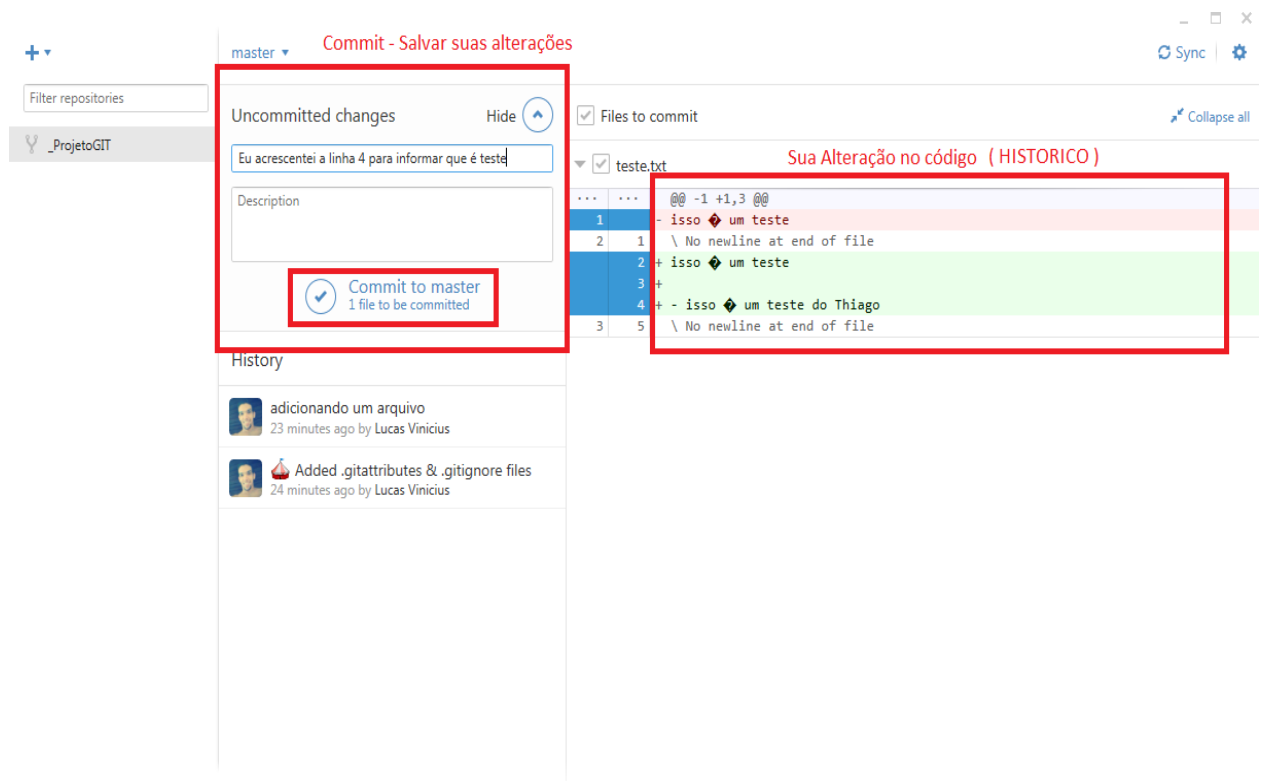
```
git pull upstream master
```

Este comando atualizará todo seu repositório de acordo com o repositório “Pai”.

***OBS.: Todos os DIAS antes de fazer qualquer coisa no código, digite esse comando, é de extrema importância ficar sempre atualizado seu repositório.***

2 – Vamos supor que você fez alguma alteração no código, **por exemplo**,

Criou um arquivo, criou uma página, fez qualquer alteração, e agora você quer **“Sinalizar”** que você realizou a alteração com sucesso.



2.1 – Toda alteração que você fizer, aparecerá um caixa “Uncommitted changes”, isso é para você salvar suas alterações, mas você pode se perguntar... eu já salvei o arquivo, apertei “CTRL + S” ... **sim**, você salvou o arquivo..., mas **não** salvou o arquivo no github, ele precisa saber quando é que você deseja salvar suas alterações.

2.2 – Toda alteração precisa ter um comentário, falando o porquê da alteração, confira na imagem acima.

2.3 – O github tem um histórico de alterações no código... haaaaa legal, então quando eu quiser recuperar um código que eu apaguei eu consigo???.... **SIM**, você consegue.

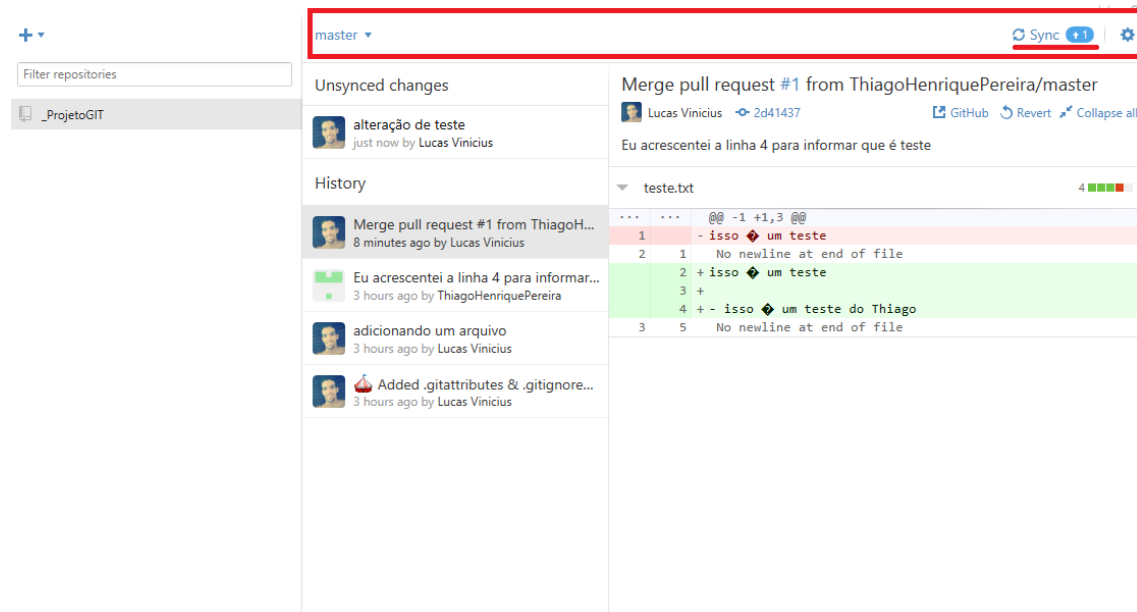
2.3 – Feito isso, apenas clique no **commit** para salvar suas alterações.

3 - Após o **commit**, note que no canto superior da tela o “**Sync**” ficará com um número na frente, isso representa o número de alterações realizadas no código.

O “**Sync**” é chamado de sincronizador, ele é responsável por sincronizar suas alterações com seu repositório online.

**OBS.:** Toda alteração no projeto, clique em “**Sync**” para sincronizar suas alterações no seu repositório online, **ATENÇÃO: Se não tiver internet, não tem como atualizar o repositório ONLINE, pois ele necessita de INTERNET.**

(Veja a imagem abaixo)



**Pronto...** feito isso, todas as alterações que você realizou está bem organizada, e controlada pelo github.

## ***Último passo... Enviando suas alterações para Repositório PAI.***

Você já realizou suas alterações no projeto, salvo todas alterações... e agora ???

Você deve se lembrar que lá em cima “**No início do tutorial – passo 6**”, que você **clonou o repositório PAI**, existe um repositório pai do projeto... CORRETO ??? Então, você precisa enviar suas alterações para ele...

Mas poxa? ... É muita coisa? ... Fiz alteração... salvei as alterações... sincronizei os arquivos com o repositório... e precisa de mais ???

SIM... precisa, veja bem...

Se você estivesse trabalhando sozinho no seu projeto, seria apenas isso que você aprendeu até agora, mas lembre-se que existem outras pessoas que também estão trabalhando no projeto, e que eles precisam das suas alterações para continuar o projeto.

Há sim... então tá... O que eu faço?

Faça o **pull request**... vá até o github **online** <https://github.com/> e vá até seu repositório em que você está trabalhando, repare que no canto direito, existe algumas opções, clique em **pull request** e clique em

“**create pull request**” e pronto... suas alterações foram enviadas para o repositório PAI...

Agora aguarde o repositório PAI aceitar ou recusar suas alterações.

(Veja as imagens abaixo)



## Passo 1

ThiagoHenriquePereira / **\_ProjetoGIT**  
forked from LucasViniciusPereira/\_ProjetoGIT

testando o git hub em equipe — Edit

3 commits 1 branch 0 releases 1 contributor

branch: master **\_ProjetoGIT / +**

This branch is 1 commit ahead of LucasViniciusPereira:master Pull Request Compare

Eu acrescentei a linha 4 para informar que é teste

ThiagoHenriquePereira authored 3 hours ago latest commit 59d495f2cb

.gitattributes	Added .gitattributes & .gitignore files	3 hours ago
.gitignore	Added .gitattributes & .gitignore files	3 hours ago
teste.txt	Eu acrescentei a linha 4 para informar que é teste	3 hours ago

Help people interested in this repository understand your project by adding a README! [Add a README](#)

HTTPS clone URL  
https://github.com/'

You can clone with HTTPS, SSH, or Subversion.

Clique aki para enviar suas alterações

[Code](#) [Pull Requests](#) 0

[Wiki](#)  
[Pulse](#)  
[Graphs](#)  
[Settings](#)

## Passo 2

Pull requests Labels Milestones Filters is:pr is:open New pull request

**Welcome to Pull Requests!**

Pull requests help you collaborate on code with other people. As pull requests are created, they'll appear here in a searchable and filterable list. To get started, you should [create a pull request](#).

Clique aki para criar um pull request

**ProTip!** Type `g` and then `i` on any issue or pull request to go back to this listing page.

### Passo 3

The screenshot shows the GitHub Pull Request interface. At the top, the base fork is 'LucasViniciusPereira\_Pro...' and the head fork is 'ThiagoHenriquePereira\_P...'. The base is 'master' and the compare is 'master'. A red box highlights the 'Create pull request' button. Below it, a yellow banner says 'Discuss and review the changes in this comparison with others.' A red text prompt says 'Clique aki para confirmar sua solicitação'. The commit list shows '1 commit' and '1 file changed'. The commit message is 'Eu acrescentei a linha 4 para informar que é teste'. The diff view shows changes to 'teste.txt' with 3 additions and 1 deletion. A red box highlights the 'Create pull request' button at the bottom right.

### Passo 4

The screenshot shows the GitHub Pull Request interface in the description view. The commit message is 'Eu acrescentei a linha 4 para informar que é teste'. The description field contains the text 'isso é um teste do pull-request'. A red box highlights the 'Create pull request' button. A green banner says 'Able to merge. These branches can be automatically merged.' A red text prompt says 'Clique aki para finalizar'. The commit list shows '1 commit' and '1 file changed'. The commit message is 'Eu acrescentei a linha 4 para informar que é teste'. The diff view shows changes to 'teste.txt' with 3 additions and 1 deletion. A red box highlights the 'Create pull request' button at the bottom right.

### Pronto... Finalizado

**Se você ainda estiver com dúvida, me procure para esclarecimento das dúvidas...**

**Agradeço a atenção, e Obrigado**

*By Lucas Vinicius Pereira*