

**Manual Básico de utilização do**

**Github**

**Tutorial criado com o intuito de ajudar na utilização do Github para projetos em equipe**.

**1º INTRODUÇÃO**

Primeira coisa que você precisa ter em mente são conceitos básicos do git, como por exemplo, o que é um repositório git, commit, branch, etc. além de que, este será um manual básico, por tanto, não terá informações aprofundadas sobre o github.

Github é um serviço web que usa o sistema de controle e versionamento de arquivos, onde por ele podemos criar diversos projetos individuais ou em grupo, facilitando a integridade e organização do grupo e do projeto. Através dele, criamos ou adicionamos e salvamos arquivos para um projeto, além de que, podemos fazer o controle de versão, evitando o risco de fazer alterações que sobrescrevam outros arquivos.

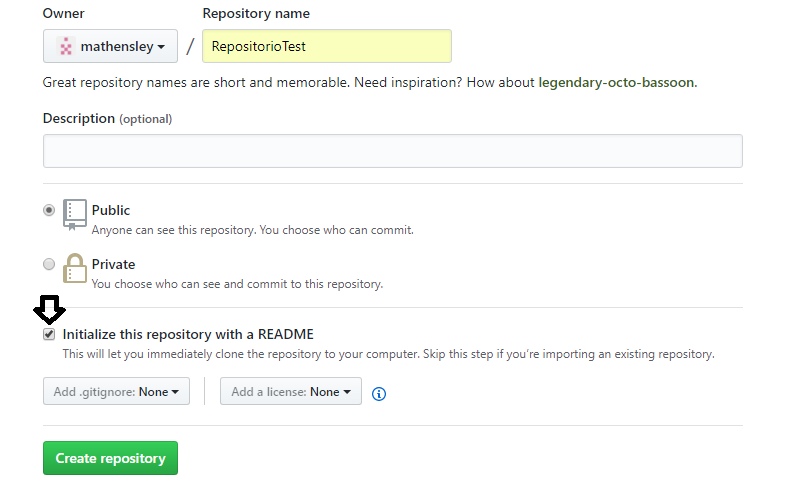
No github podemos commitar, ou seja, ”salvar” nossos arquivos no repositório, e clonar, que seria baixar o projeto, além disso, o git tem a opção de criar branches. Branches são versões que estão paralelas ao seu projeto, os arquivos salvos estão no projeto, porém não interferem com a branch principal (por padrão a master), fazendo com que você possa trabalhar livremente sem interromper a versão principal. Após o conteúdo da branch secundária estiver “completo”, você poderá dar um merge (juntar a sua parte com a parte principal do projeto) com a branch master para publicar as mudanças realizadas.

\*Seus projetos no git serão públicos (pode ser privado, porém é um serviço pago), então outras pessoas podem ter acesso ao conteúdo.

**2º CRIANDO SEU REPOSITÓRIO**

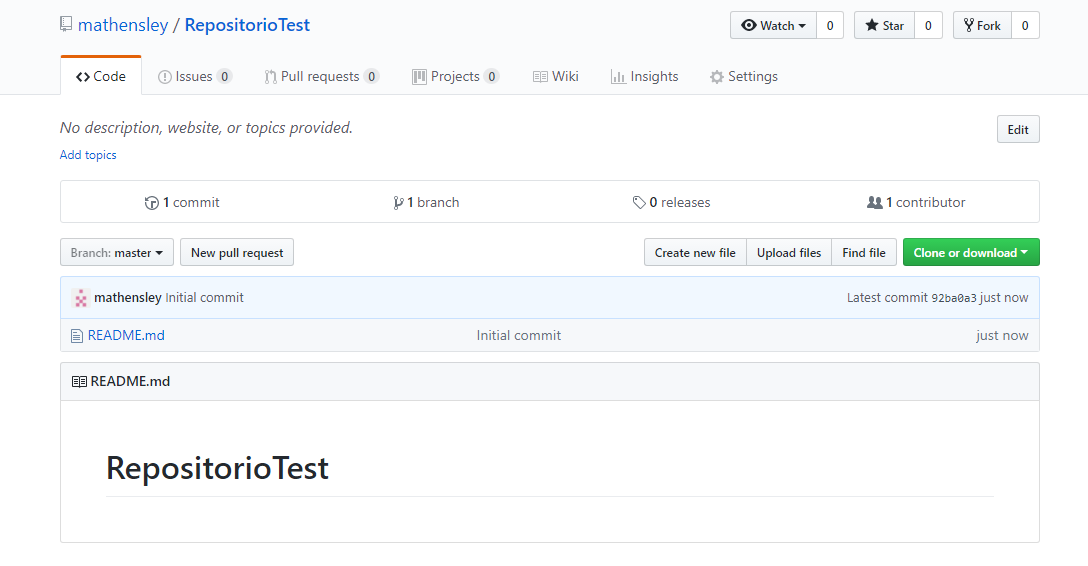


Após criar sua conta você pode criar um novo repositório (ou até mesmo contribuir com outros que estejam livres para qualquer pessoa commitar).

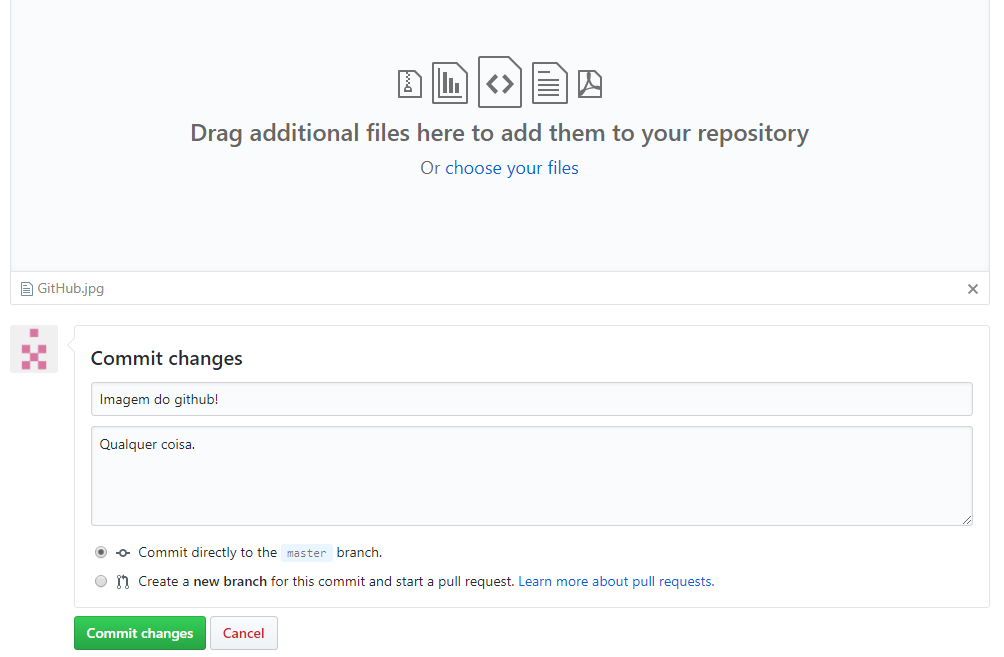


Em seguida aparecerá esta tela. Ao colocar o nome do repositório, e ou um comentário, clique em *Initialize this repository*, e você poderá começar.

Quando tiver criado, você verá a seguinte tela:

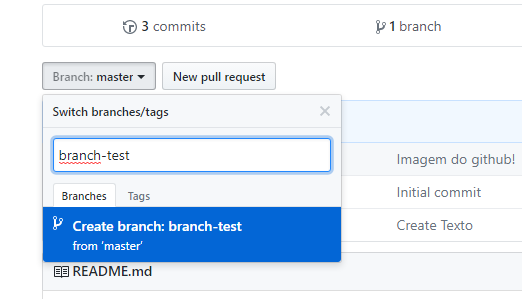


Apartir daí podemos começar a mexer no nosso repositório. A aba ‘<>Code’ é onde estará todo os arquivos do projeto, nela temos várias opções, começando pelo botão verde (‘Clone or download’), é onde podemos pegar a url do projeto, abrir no Desktop usando o aplicativo do github ou baixar o arquivo compactado do projeto. Além disso, temos a opção ‘Create new file’ onde você cria um novo arquivo de texto e o commita; ‘Upload file’ (o mais utilizado) é onde se faz o upload de todos os arquivos que você queira commitar no projeto:

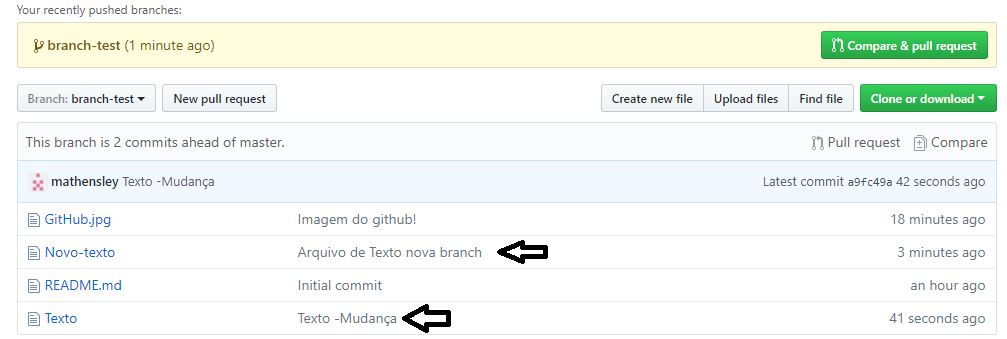


Você pode escolher ou arrastar um ou mais arquivos (até 100) e na parte de ‘Commit changes’ é onde você descreve as mudanças ou adições feitas neste commit, comenta (opcional), poderá também escolher a branch onde estará sendo commitado o arquivo. Após isso, seu arquivo irá aparecer na aba.

**3º BRANCHES**

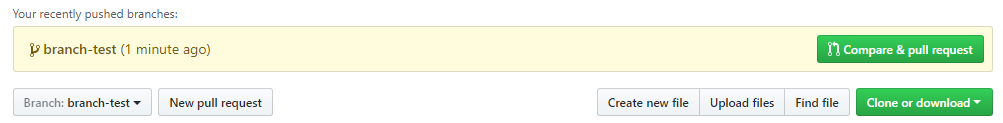


No menu ‘Branch’, você poderá ver todas as branches do projeto e também criar uma. Após criá-la, ela terá herdado todos os arquivos que tinham na branch a qual ela foi ligada (nesse caso a master), e a partir daí você pode criar novos arquivos e fazer mudanças.



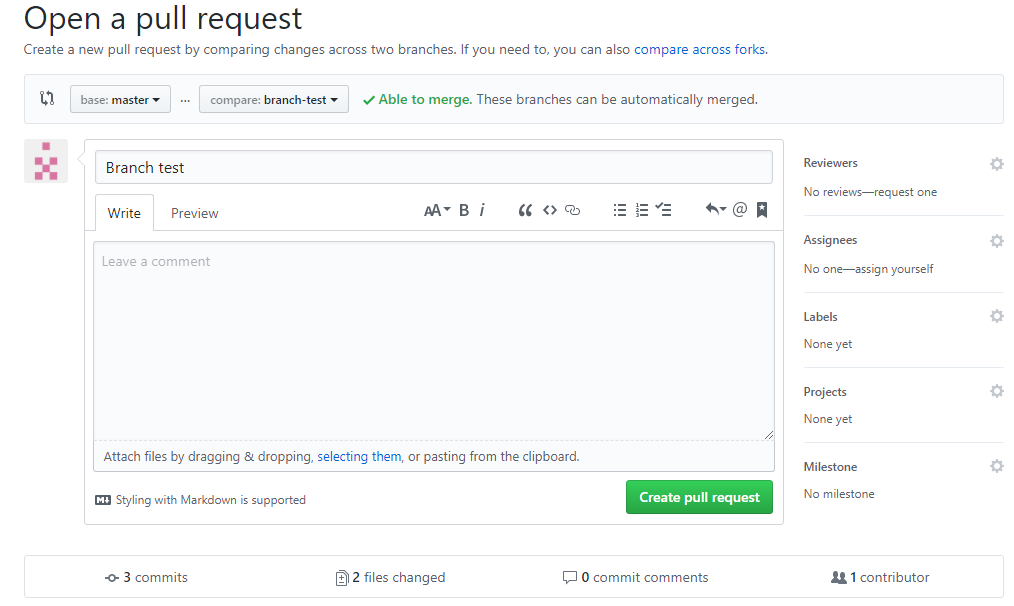
Ao fazer mais commits o git mostra que sua branch está mais à frente da principal, então o que fazer?

**4º PULL REQUESTS**

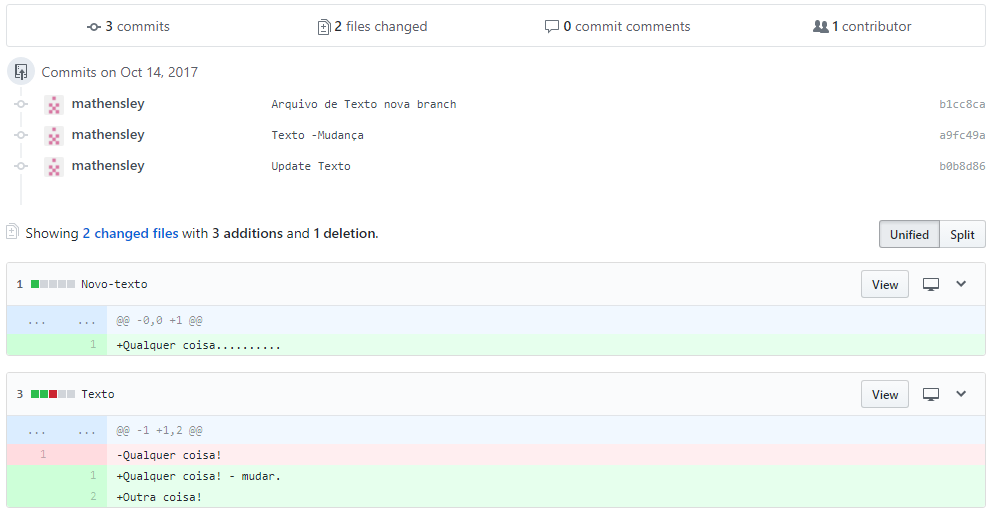


Pull Request é uma opção muito importante para quem está trabalhando em equipe, pois é por ela que você pode mandar todas as adições e mudanças para a branch principal com segurança.

Ao clicar em ‘New pull request’ (ou ‘Compare & pull request’), você estará mandando um pedido para que as mudanças e adições feitas sejam colocadas na branch principal:



Nesta página, você confere se poderá haver algum problema para dá o merge, onde ele compara a branch que você quer mandar os arquivos com a sua. Para mandar o request (pedido), você escreve um título, se quiser pode deixar um comentário servindo de sumário para as mudanças propostas, além de várias outras opções como, adicionar outros colaboradores para fazer um review, adicionar um administrador, etc.



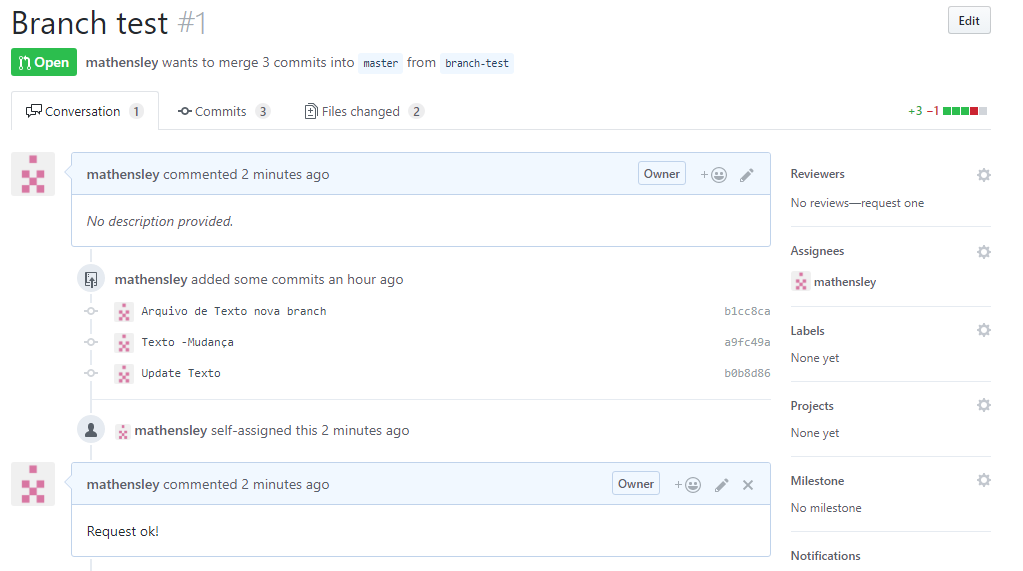
Mais abaixo na página, temos os commits realizados, e o principal que é o que será adicionado e mudado.

Toda a parte que for verde indica o que será adicionado, e caso haja alguma parte vermelha, indica que essa linha será descartada e substituída pela a próxima, você deve ter cuidado para ter certeza se deve mesmo ser substituída essa parte. É sempre bom deixar um comentário explicando o porquê da mudança.



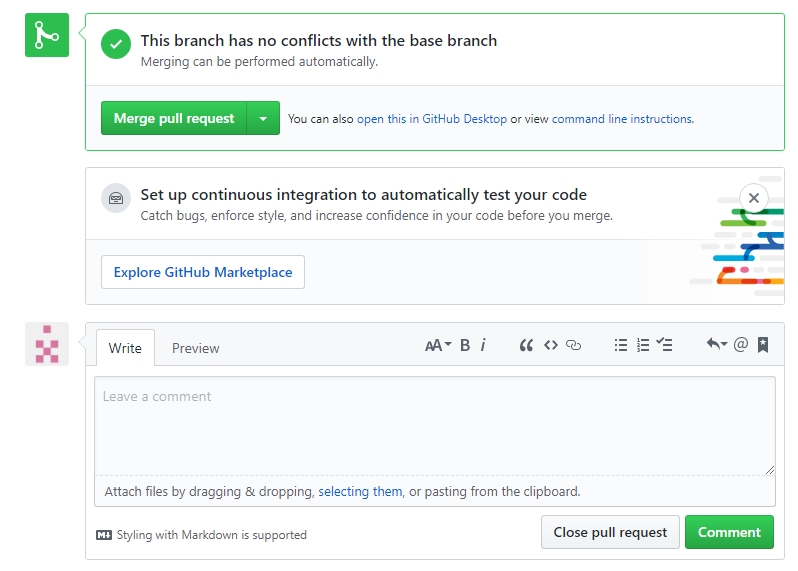
Após isso tudo, você pode criar o pull request. Depois dele ser criado irá aparecer nas abas o seu request:





Ao clicar no seu request você poderá ver o título, de qual e para qual branch você deseja fazer o merge, a descrição feita sobre as mudanças, os commits, e possíveis comentários de colaboradores do projeto.

Caso você seja um colaborador, então terá que esperar o administrador dá o merge.



Caso você seja o administrador do projeto, você poderá então conferir e dá o merge dos arquivos na branch principal, ou então fechar este request se houver problemas. Após o merge ser completado, você pode escolher deletar a branch caso ela não seja mais usada.

Ao fazer um pull request ele sempre irá mandar todos os arquivos da branch, então se você quiser que um só arquivo vá para branch principal ou que pelo menos um arquivo não vá, você terá que criar mais uma branch só para isso.

**5º OUTRAS CARACTERÍSTICAS**

Há outras opções interessantes no github, como a guia ‘contributor’ na aba ‘<>Code’, que mostra dados estatísticos de commits da branch e as contribuições dos colaboradores. A guia ‘branches’ onde mostra todas as branches do projeto, e a ‘commits’ que mostra todos os commits da branch.

Há também a aba ‘Issues’, onde você pode apontar erros de merge, etc. A aba ‘Wiki’, usada para criar páginas para descrever funções do projeto, características, etc.

Na aba ‘Settings’, você pode renomear o nome do repositório, desativar ou ativar funções do git, excluir o projeto dentre outras, você pode também adicionar colaboradores, informando o username, nome ou e-mail do usuário que você deseja adicionar como colaborador.

**6º CONSIDERAÇÕES FINAIS**

Github é uma ferramenta bem versátil para administração, organização e realização de um projeto individual ou em grupo, e é muito utilizada pela comunidade. Nela há muitas funções e características que você só irá aprender com o passar do tempo e a medida das necessidades, por tanto, o motivo deste ser só um manual básico.

Apesar disso, acredito este ser um manual de grande ajuda para quem estiver começando ou até mesmo para quem utiliza a algum tempo, porém não conheça algumas funções.