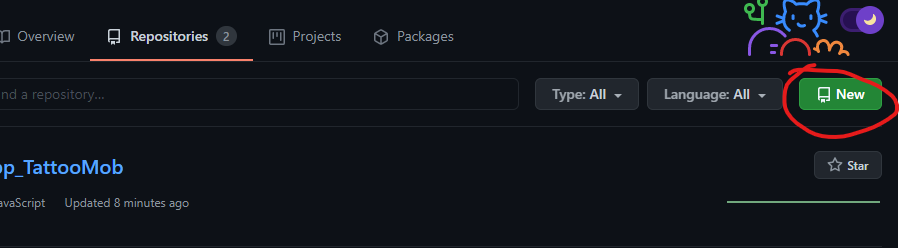
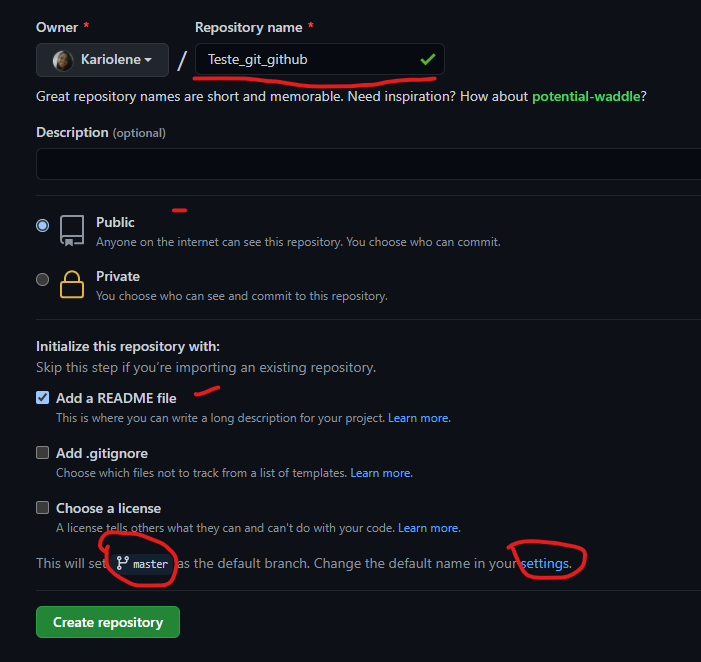
Lembrando que é necessário já possuir uma conta no github e instalar e configurar o git na máquina. Se ainda não fez isso, nas referencias existe os tutoriais para isso também.

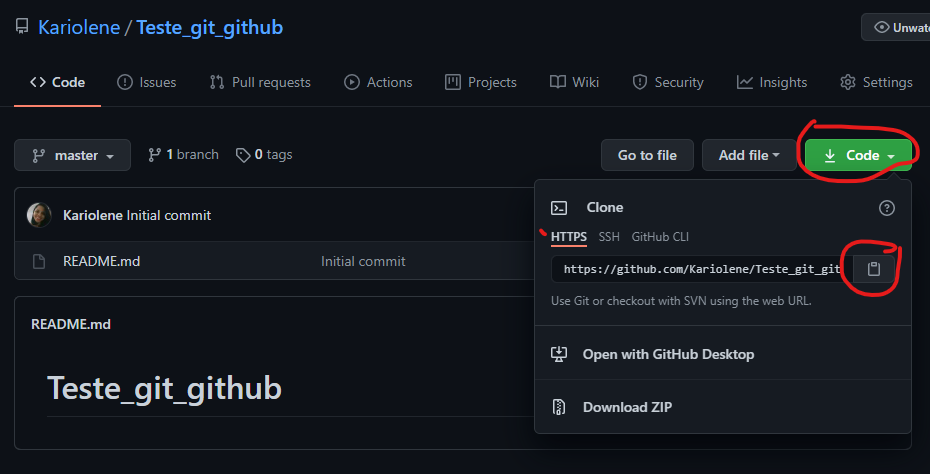
**Cria o repositório no github:**

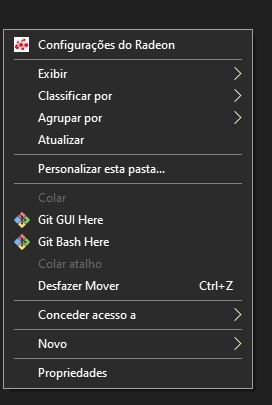


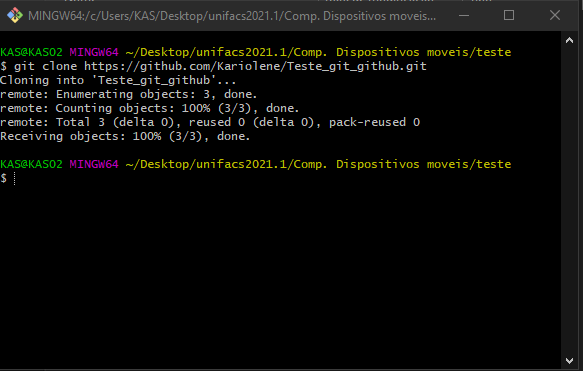


**Clonando repositório na máquina.**

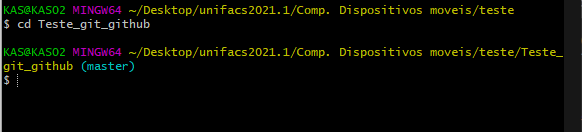
1. Acessa o repositório no github, copia o link para clonar.



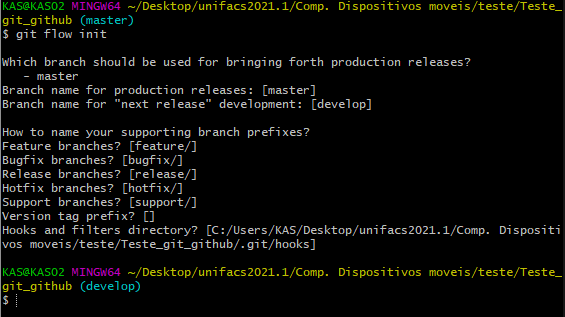
1. Escolhe uma pasta no computador, botão direito escolhe: “Git Bash Here”
2. Digita “git clone” + clica com botão direito do mouse e cola o link copiado, enter:



1. Digita “cd” + “nome do repositório criado”.

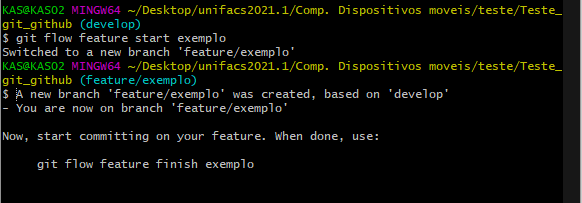


1. Digita: git flow init.
2. Enter em todas as perguntas e o git flow já estará inicializado no repositório local.

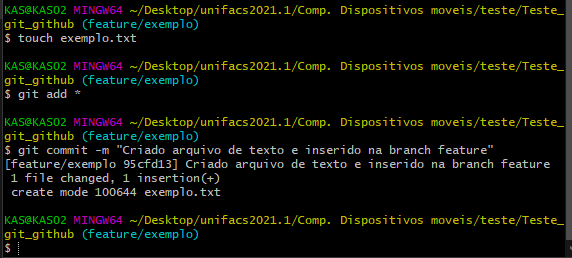


Atribuindo projeto inicial seguindo fluxo do git flow:

1. Ainda no terminal do git, ou se tiver saído do terminal, basta ir até a pasta do repositório, botão direito e abre o git bash . Em seguida digita-se “git flow feature start “nome\_de\_sua\_preferencia”” depois enter:

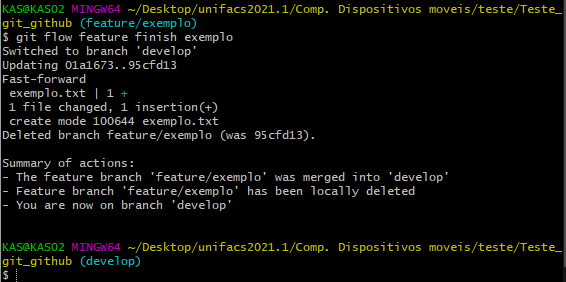


1. Depois copia o arquivo e cola no diretório, voltando ao terminal neste ponto, vamos digitar: “git add \*” e dar enter.
2. A seguir digitar : git commit -m “[descrição sobre a alteração, mantem as aspas e remove os colchetes ]” .

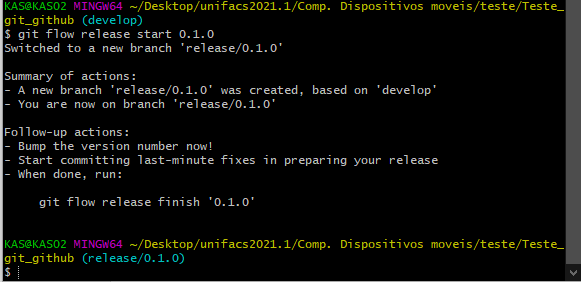


OBs.: “touch” não é necessário, ele apenas ajudou a criar um arquivo de texto para podermos segui com o exemplo, mas poderia apenas copiar um projeto e colar na pasta do repositório e em seguida voltar ao terminar e digitar o “git add \*” e depois o “git commit -m +descrição ”, que também funciona. Agora precisamos subir essa alteração para develop.

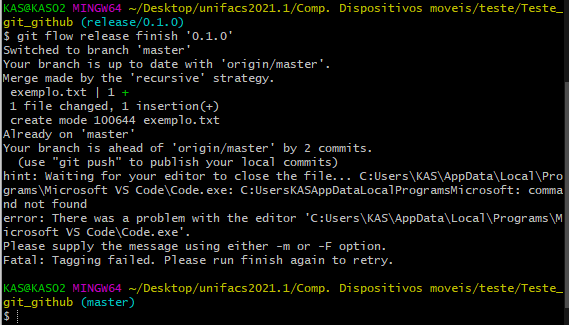
1. Para subir a alteração para a branch develop, precisamos digitar “git flow feature finish + [nome atribuído ao criar a feature, nesse exemplo: “exemplo” ]” :



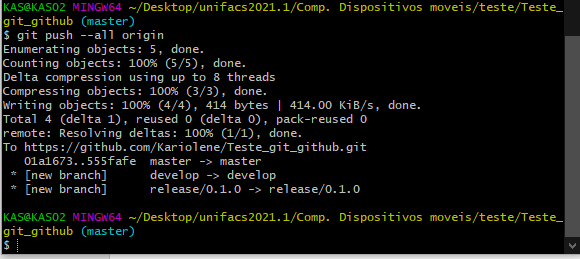
1. Em seguida precisamos criar uma branch release: git flow release start 0.1.0 para que a alteração ou o projeto inicial chegue até a master. Lembrando que para envio a master é preciso ter sido testado e aprovado, a master é como o ambiente de produção e a release é como o ambiente de qualidade.



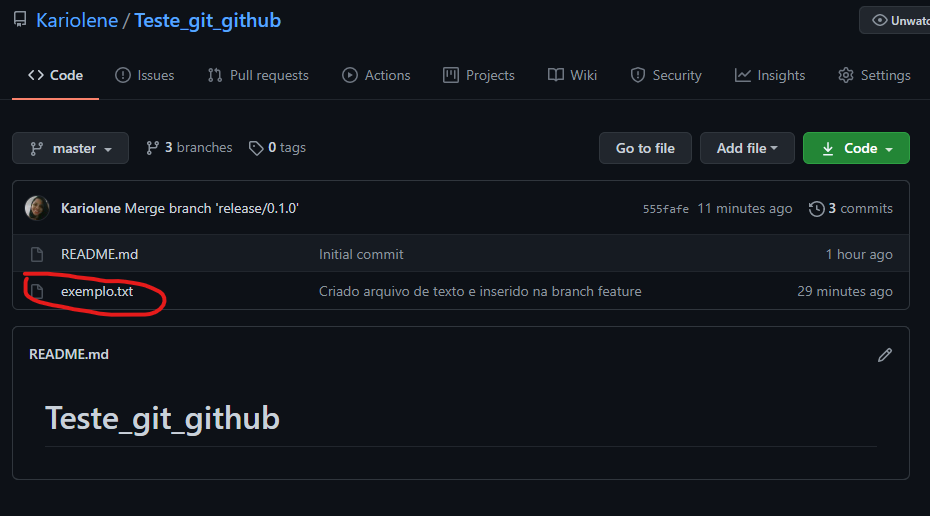
1. Quando finalizamos a release, então a master recebe um merge e o develop também é atualizado, é possível fazer alterações no release, mas somente quando necessário. Para finalizar release: git flow release finish ‘[digito da versão criada]’ e enter.



1. Agora precisamos enviar ao github, podemos usar o git push –all origin, para enviar todas as branch’s ao repositório remoto ou apenas uma branch usando o comando: git push origin [nome da branch], vou enviar tudo.



As próximas alterações basta criar uma feature, alterar, finalizar e enviar o develop para o repositório remoto usando : git push origin [nome da branch].



Referencias:

Envio do git flow ao github:

<https://fjorgemota.com/para-onde-enviar-meus-repositorios-git/#more-1450>

inicializando git flow:

<https://fjorgemota.com/git-flow-uma-forma-legal-de-organizar-repositorios-git/>

git flow na prática:

<https://www.youtube.com/watch?v=wzxBR4pOTTs>

Configurando o git:

<https://www.youtube.com/watch?v=SOxafinthys>

Criando repositório:

<https://www.youtube.com/watch?v=h1-XZ9Kh1H4>