

Trabalho Prático Individual Final - Git

Desenvolvimento Ágil de Software

Desenvolvimento de Software

Pedro Marques



Índice

Trabalho Prático Individual Final - Git

Introdução	2
1. Criação das Branches para a utilização do modelo GitFlow.	3
2. Níveis de acesso no GitHub	4
3. Revisão de código antes de um pull request	5
4. Adicionar GitIgnore	6
5. Controlar as versões	7-10

Introdução

Devemos começar por criar um repositório local com os seguintes comandos:

```
MINGW64:/c/Users/marques/projetofinal

marques@DESKTOP-F1F0C9I MINGW64 ~
$ mkdir projetofinal

marques@DESKTOP-F1F0C9I MINGW64 ~
$ cd projetofinal

marques@DESKTOP-F1F0C9I MINGW64 ~/projetofinal
$ git init
Initialized empty Git repository in C:/Users/marques/projetofinal/.git/
```

Depois de criado o repositório local devemos acessar o GitHub e criar o nosso repositório em: **Your repositories >>** 

Assim que concluído, para ligar o nosso repositório local ao nosso repositório do GitHub devemos digitar o seguinte comando:

```
marques@DESKTOP-F1F0C9I MINGW64 ~/projetofinal (develop)
$ git remote add origin https://github.com/DAS-ProjetoFinal/projetofinal.git
```

1. Criação das Branches para ser possível a utilização do modelo GitFlow.

Para criar todas as branches necessárias e configurar o modelo gitflow devemos começar por digitar o seguinte comando:

```
marques@DESKTOP-F1F0C9I MINGW64 ~/projetofinal (master)
$ git flow init
No branches exist yet. Base branches must be created now.
Branch name for production releases: [master]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Users/marques/projetofinal/.git/hooks]
```

Quando digitado o comando é recomendado pressionar **ENTER** até ao fim da configuração. Após o procedimento, estaremos por diante de 2 branches: Master e Develop.