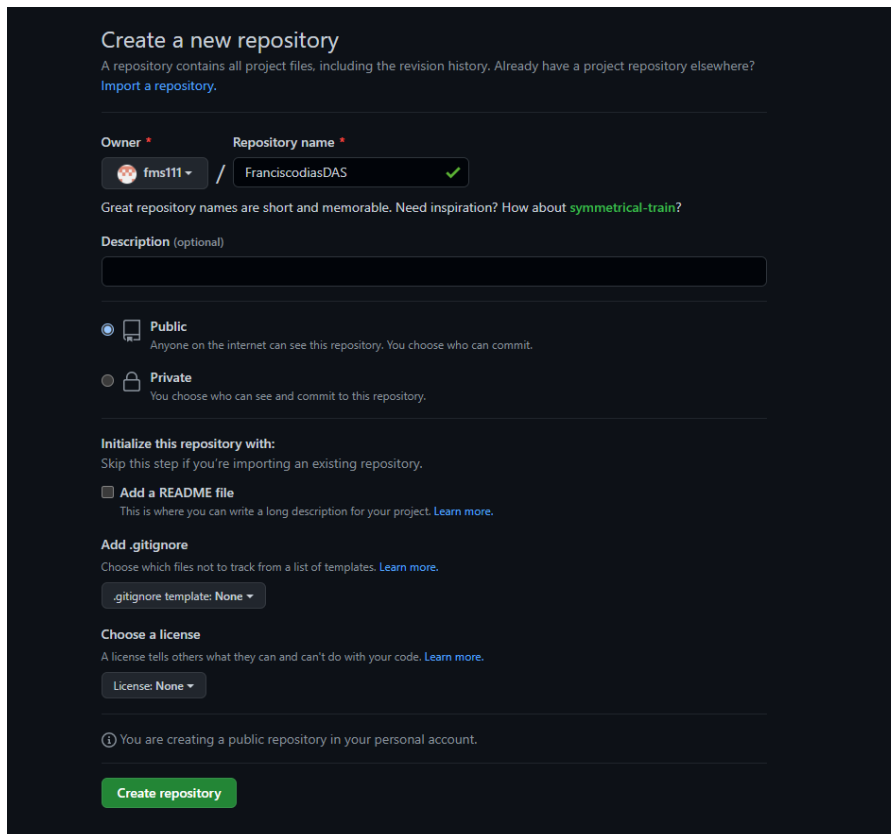


Relatório Trabalho Final DAS - Francisco Dias 2022095


1. Criar o Repositório no GitHub e as branches do mesmo



Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


Owner ^{*} Repository name ^{*}

 fms111 / FranciscodiasDAS ✓

Great repository names are short and memorable. Need inspiration? How about [symmetrical-train?](#)

Description (optional)

☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: **None**

Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

License: **None**

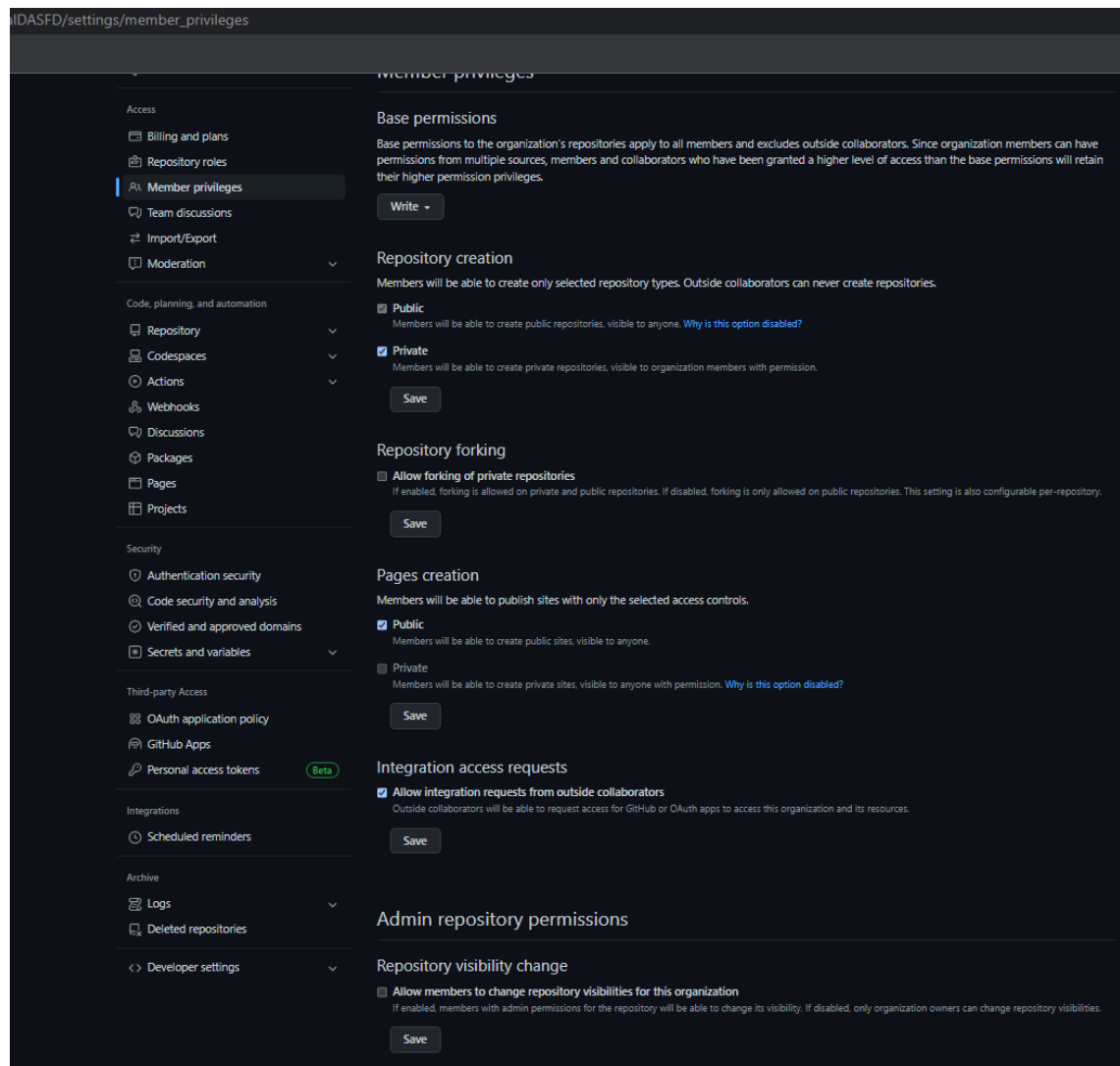
ⓘ You are creating a public repository in your personal account.

[Create repository](#)

```
$ 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/Utilizador/FranciscodiasDAS/.git/hooks]
```

2. Criei uma organização no GitHub e coloquei as regras em ordem



Altereí a permissão para Write

Desativei a Repository Visibility Change

3. Tornei obrigatória a revisão do código antes de ser aprovado um Pull Request

The screenshot shows the 'Protect your most important branches' settings in GitHub. The 'Branch name pattern' is set to 'master'. Under 'Protect matching branches', several options are checked: 'Require a pull request before merging', 'Require approvals' (with a dropdown set to 1), 'Require review from Code Owners', and 'Require status checks to pass before merging'. Other options like 'Dismiss stale pull request approvals' and 'Require approval of the most recent reviewable push' are unchecked.

Protect your most important branches
Branch protection rules define whether collaborators can delete or force push to the branch and set requirements for any pushes to the branch, such as passing status checks or a linear commit history.
Your GitHub Free plan can only enforce rules on its public repositories, like this one.

Branch name pattern *

master

Protect matching branches

- ☒ **Require a pull request before merging**
When enabled, all commits must be made to a non-protected branch and submitted via a pull request before they can be merged into a branch that matches this rule.
- ☒ **Require approvals**
When enabled, pull requests targeting a matching branch require a number of approvals and no changes requested before they can be merged.
Required number of approvals before merging: 1
- ☐ **Dismiss stale pull request approvals when new commits are pushed**
New reviewable commits pushed to a matching branch will dismiss pull request review approvals.
- ☒ **Require review from Code Owners**
Require an approved review in pull requests including files with a designated code owner.
- ☐ **Require approval of the most recent reviewable push**
Whether the most recent reviewable push must be approved by someone other than the person who pushed it.
- ☐ **Require status checks to pass before merging**
Choose which status checks must pass before branches can be merged into a branch that matches this rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to a branch that matches this rule after status checks have passed.

4. Adicionar um ficheiro .Gitignore para ignorar ficheiros .doc e .docx

The screenshot shows a nano text editor window titled '.gitignore'. The content of the file is: *.doc and *.docx. The status bar at the bottom indicates 'File Name to Write: .gitignore'.

```
GNU nano 6.4 .gitignore Modified
*.doc
*.docx

File Name to Write: .gitignore
```

The screenshot shows a terminal window with the following commands and output:

```
$ nano .gitignore
Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (master)
$ ls -la
./ ../.git/ .gitignore
Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (master)
$ cat .gitignore
*.doc
*.docx
```

5. Controlar as versões do relatório a entregar usando o repositório, sendo que é necessário que existam pelo menos:

a) 1 carregamento inicial do ficheiro no branch develop

Comandos para fazer push para o repositório:

```
Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore

nothing added to commit but untracked files present (use "git add" to track)

Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (master)
$ git add .
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it

Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (master)
$ git commit -m "Configuração do gitignore"
[master 86a228d] Configuração do gitignore
1 file changed, 2 insertions(+)
 create mode 100644 .gitignore

Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (master)
$ git push origin develop
Everything up-to-date

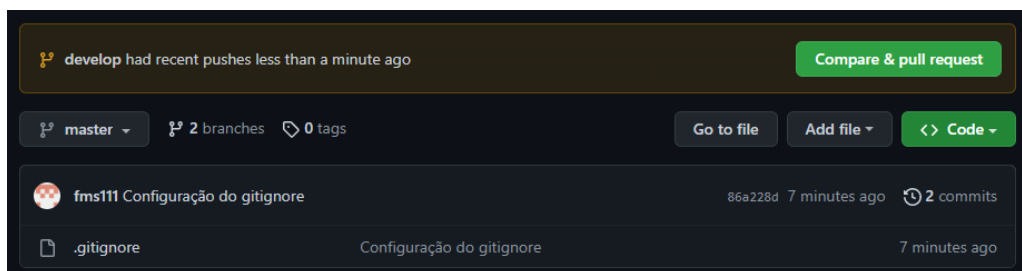
Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (master)
$ git push origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Writing objects: 100% (3/3), 274 bytes | 274.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/fms111/FranciscodiasDAS.git
 257baef..86a228d master -> master
```

```
Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/TrabalhofinalDAS (develop)
$ git add .

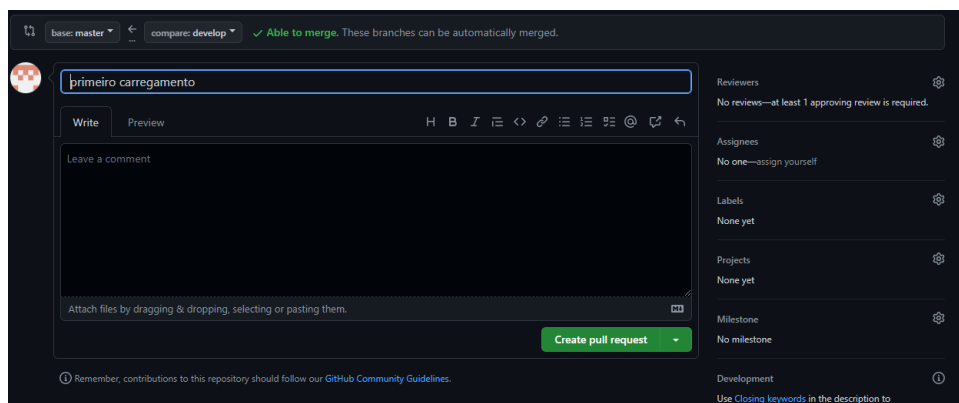
Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/TrabalhofinalDAS (develop)
$ git commit -m "Primeiro carregamento"
[develop 0fbd95a] Primeiro carregamento
1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Doc1.pdf

Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/TrabalhofinalDAS (develop)
$ git push origin develop
```

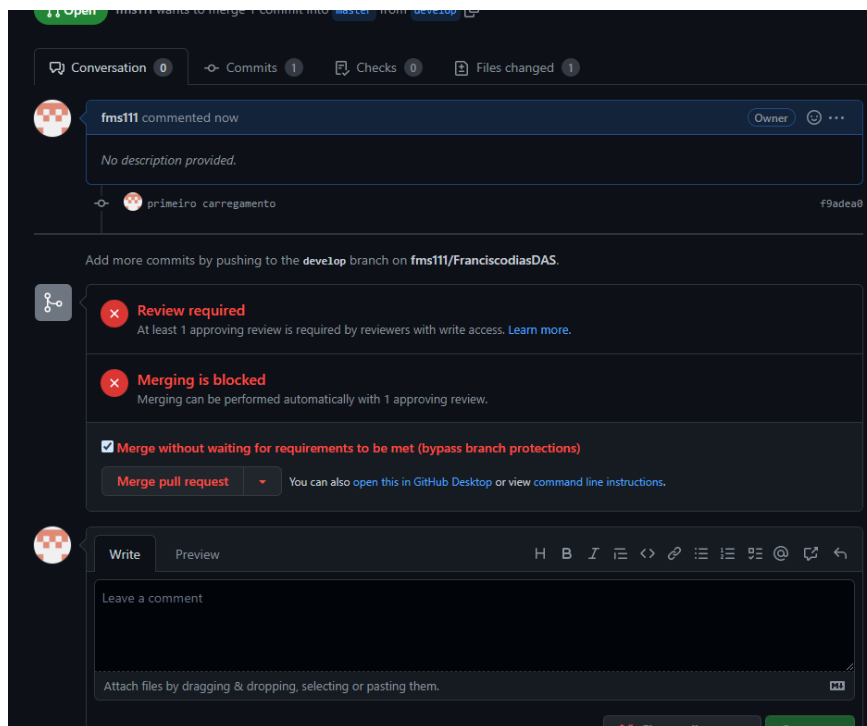
Print do Pull Request



Criação do pull request



Review / Aprovação do Pull Request



b) 5 alterações no próprio branch develop com os devidos comentários

-- Criação e alteração da feature1das --

```
Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (develop)
$ git flow feature start feature1das
Switched to a new branch 'feature/feature1das'

Summary of actions:
- A new branch 'feature/feature1das' was created, based on 'develop'
- You are now on branch 'feature/feature1das'

Now, start committing on your feature. When done, use:

    git flow feature finish feature1das

Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (feature/feature1das)
$ git add .

Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (feature/feature1das)
$ git commit -m "relatorio1"
[feature/feature1das 4d39de4] relatorio1
1 file changed, 0 insertions(+), 0 deletions(-)

Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (feature/feature1das)
$ git flow feature finish feature1das
Switched to branch 'develop'
Your branch is up to date with 'origin/develop'.
Updating f9adea0..4d39de4
Fast-forward
 Doc1.pdf | Bin 25662 -> 409984 bytes
1 file changed, 0 insertions(+), 0 deletions(-)
Deleted branch feature/feature1das (was 4d39de4).

Summary of actions:
- The feature branch 'feature/feature1das' was merged into 'develop'
- Feature branch 'feature/feature1das' has been locally deleted
- You are now on branch 'develop'

Utilizador@DESKTOP-CCS4PPJ MINGW64 ~/FranciscodiasDAS (develop)
$ git push origin develop
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 370.14 KiB | 16.82 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/fms111/FranciscodiasDAS.git
 f9adea0..4d39de4 develop -> develop
```