



# GitHub4Women

Domain 2: Working with GitHub  
Repositories



# Agenda

- 1 Navegação no Repositório
- 2 Criando um Repositório
- 3 Templates de Repositório
- 4 Arquivos recomendados
- 5 Escrevendo um bom README
- 6 Mantendo um Repositório



# Agenda

- 7 Clonar um Repositório
- 8 Criar uma branch
- 9 Adicionando arquivos
- 10 Insights de um Repositório
- 11 Stars
- 12 Feature previews



# Navegação repositório



main 34 Branches 3 Tags

Go to file

Add file ▾

<> Code ▾

In this tutorial for C# development with ASP.NET Core, you create a C# ASP.NET Core web app in Visual Studio.

[dotnet.microsoft.com/en-us/](https://dotnet.microsoft.com/en-us/)

 [Readme](#)

 MIT license

 **Security policy**

 Activity

#### Custom properties

☆ 2 stars

👁 2 watching

0 forks


Releases 3

**Release 3.0.0** Latest  
on Dec 1, 2023

## Packages




No packages published  
Publish your first package

Contributors 4

 **leandromsft** Leandro Prado

 **leandro-dev01** Leandro DEV01

leandromsft Merge branch 'main' of https://github.com/prado-org/dotnetcore-webapp ✓ db3f9f9 · last week 439 Commits		
.devcontainer	Add port forwarding for APP and API services	2 months ago
.github	Update ci.yml	last week
.vscode	disable cSpell	2 months ago
docs	Update About.md	6 months ago
iac	Remove unused parameter in main-common.bicep	3 months ago
src	The most important changes involve enhancing the `Weathe...	last week
.gitignore	Update .gitignore to exclude Mono auto generated files	2 months ago
CODEOWNERS	Update CODEOWNERS	5 months ago
CONTRIBUTING.md	rename file	5 months ago
LICENCE.md	Create LICENCE.md	3 months ago
README.md	Update README.md	2 months ago
SECURITY.md	Create SECURITY.md	6 months ago

 [README](#)    [MIT license](#)    [Security](#)

## My First Project

**Warning**

This code is used for example purposes only.

# Criando um repositório

## Create a new repository

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

*Required fields are marked with an asterisk (\*).*

### Repository template

No template ▾

Start your repository with a template repository's contents.

Owner \*



prado-org ▾

/

Repository name \*

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

Description (optional)

- ☐ **Public**  
Anyone on the internet can see this repository. You choose who can commit.
- ☐ **Internal**  
Pedro's Inc [enterprise members](#) can see this repository. You can choose who can commit.
- ☒ **Private**  
You choose who can see and commit to this repository.

Initialize this repository with:

☐ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs](#).

## Add .gitignore

.gitignore template: None ▾

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

## Choose a license

License: None ▾

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

## Grant your Marketplace apps access to this repository

prado-org is subscribed to 1 Marketplace app.



**Azure Boards** *(auto-installed)*

Connects Azure Boards with GitHub to plan, track, and discuss work across your teams

You are creating a private repository in the prado-org organization (Pedro's Inc).

Create repository



# Tipos de Repositórios

## Public

Se sua conta não for uma conta de usuário gerenciada, você poderá criar repositórios públicos. Os repositórios públicos são acessíveis a todos na Internet.

## Internal

Repositórios internos podem ser acessados por todos os integrantes da empresa, estando disponíveis apenas em contas enterprise.

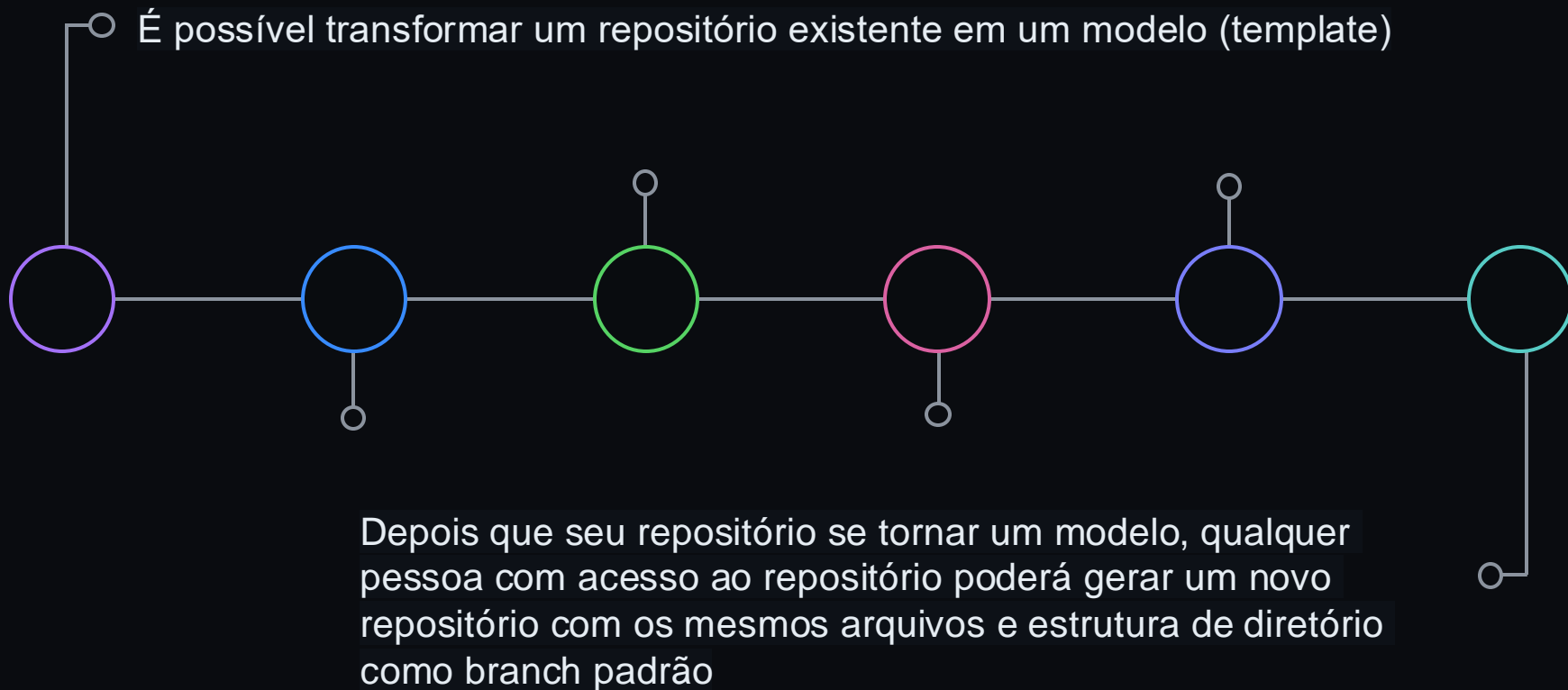
**Apenas contas Enterprise**

## Private

Os repositórios só podem ser acessados por você, pelas pessoas com as quais você compartilha explicitamente o acesso e, para repositórios da organização, por determinados integrantes da organização.



# Templates







Demo

# Arquivos recomendados



# Arquivos recomendados

Os arquivos recomendados comunicam expectativas para o seu projeto, ajudam a gerenciar contribuições e também ajudam outros usuários a navegar pelo seu trabalho

README

CODEOWNERS

LICENSE

CONTRIBUTING



# LICENSE

O licenciamento do projeto é o que define o que outros podem e não podem fazer com seu código



Dentro do escopo de *open source* existem variações nas permissões atribuídas



De acordo com os Termos de Serviço, desde que seu código esteja em um repositório público, outros podem ver e fazer um *fork* no seu código



Arquivo LICENSE.md (ou .txt ou .rst) geralmente é encontrado na raiz do repositório



*License picker* na criação dos projetos

## Choose a license

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

License: None ▼



# CONTRIBUTING

É utilizado para orientar sobre como outras pessoas podem contribuir com seu projeto

- Contém orientações sobre como as contribuições serão absorvidas
- Os usuários serão direcionados a esse arquivo quando criarem um issue ou abrir um pull request
- Visa estimular interações úteis e assertivas no contexto do projeto



# CODEOWNERS

Arquivo utilizado para definir os responsáveis por aprovações e revisões de códigos do repositório



O arquivo CODEOWNERS deve estar ou em `.github`, ou na raiz ou em `docs/`



Pode haver mais de um arquivo, mas a varredura é realizada conforme a ordem acima



Cada branch pode ter um CODEOWNERS diferente



Os "code owners" devem ter permissão de gravação no repositório



Caso uma equipe seja a revisora, ela deve estar visível e ter permissões de gravação



# README

É, frequentemente, o primeiro item que um visitante verá ao acessar seu repositório. Ele é utilizado para detalhar o trabalho contido naquele repositório e, geralmente, responde às perguntas:

- Qual a finalidade do projeto?
- Como outros usuários podem utilizar esse projeto?
- Onde os usuários podem buscar ajuda sobre o projeto?
- Quem são os responsáveis pela manutenção do projeto?



# Escrevendo um bom README

## Título e Descrição

---

Comece com um título claro e conciso que descreva o propósito do seu projeto.

---

## Instruções de Instalação

---

Forneça instruções claras e detalhadas sobre como instalar seu projeto.

---

## Exemplos de Uso

---

Forneça exemplos práticos de como usar o seu projeto.

---







Demo

# Configurações dos repositórios



# Mantendo um repositório

Configurações do  
Repositório

Branches

Pull Requests (PRs)

Issues

Wikis

Proteção de  
Branches

# Proteção de Branches

## Sintaxe fnmatch

### Escopo da regra

Branch específica

Todas as  
branches

Match Name

### Prevalência

Nome completo

Mais antiga

# Proteção de Branches

## Sintaxe fnmatch

### Escopo da regra

Branch específica

Todas as  
branches

Match Name

### Prevalência

Nome completo

Mais antiga

**Apenas uma  
regra é  
aplicada por  
branch**

# Mantendo um repositório

Configurações do  
Repositório

Branches

Pull Requests (PRs)

Issues

Wikis

Proteção de  
Branches

# Clonar um repositório

1

- Em GitHub.com, acesse a página inicial do repositório de interesse

2

- Clique em código (acima da lista de arquivos) e copie a URL do repositório

3

- Abra o terminal (Git Bash ou CMD) e navegue até o diretório onde o clone será armazenado

4

- Execute o comando de clone: `$ git clone https://github.com/<proprietario>/<repositorio>`



# Criar uma *branch*

Primeiro, mude para o ramo (branch) a partir do qual você deseja criar o novo ramo. Por exemplo, se você tem os seguintes ramos: master, dev e branch1, e deseja criar um novo ramo chamado “subbranch\_of\_b1” sob o ramo chamado “branch1”, execute o seguinte comando:

```
git checkout branch1
```



Agora, crie o novo ramo com o seguinte comando

```
git checkout -b subbranch_of_b1
```





# Adicionando arquivos

## Interface GitHub.com

1

• Em GitHub.com, acesse a página principal do repositório.

2

• Acima da lista de arquivos, selecione o menu suspenso “Adicionar arquivo” e clique em “Carregar arquivos”.

3

• Alternativamente, você pode arrastar e soltar os arquivos diretamente no seu navegador.

4

• Escolha os arquivos que deseja carregar, arrastando e soltando o arquivo ou clicando em “escolher seus arquivos”.

5

• No campo “Mensagem de commit”, digite uma mensagem curta e significativa que descreva a alteração feita no arquivo.

6

• Abaixo dos campos de mensagem de commit, decida se deseja adicionar o commit ao ramo (branch) atual ou a um novo ramo. Se o ramo atual for o ramo padrão, escolha criar um novo ramo para o seu commit e, em seguida, crie uma solicitação de pull.

7

• Clique em “Propor alterações”.



# Adicionando arquivos

## Linha de comando

1

• Abra o terminal, por exemplo: CMD ou Git Bash

2

• Clone um repositório usando o commando : `$ git clone https://github.com/<proprietario>/<repositorio>.git`

3

• Adicione arquivos: `$ git add <arquivo1.txt> <arquivo2.js>`

4

• Faça o *commit* dos arquivos adicionados: `$ git commit -m "<Adicionando arquivos>"`

5

• Faça o push (envio) dos arquivos: `$ git push origin <ramo>`



Termos entre <> devem ser substituídos pelos termos adequados ao seu cenário



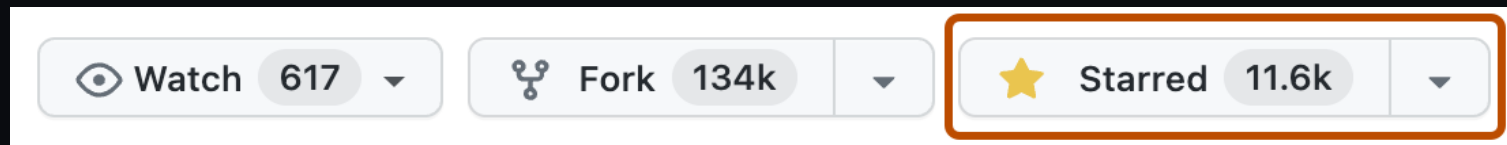


# Insights de um Repositório

1. No GitHub.com, navegue até a página principal do repositório.
2. Clique em **Insights**.



# Stars



1

Salvando um Repositório com Estrelas

2

Visualizando Quem Marcou com Estrela um Repositório

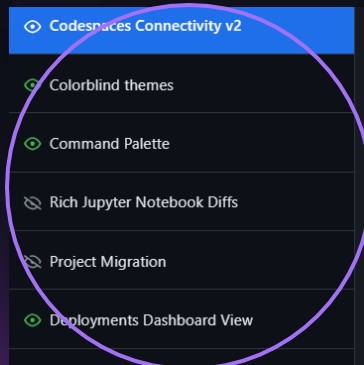
3

Organizando Repositórios Marcados com Estrelas em Listas





Demo



# Feature previews

- Alpha: Recursos em pleno desenvolvimento, não documentados e sem suporte técnico.
- Beta: Prontos para distribuição mais ampla, mas sem obrigação de suporte técnico.
- Disponibilidade Geral (GA): Totalmente testados e disponibilizados publicamente com suporte técnico





Demo



# Hora de praticar!

Atividade Módulo 2





# Thank you

