



E-BOOK PARA INICIANTES GIT & GITHUB NO VISUAL STUDIO

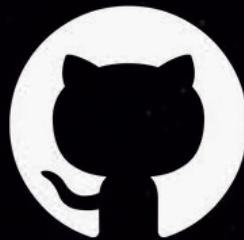
Criado por estudantes da Universidade Paulista

SUMÁRIO:

1. INTRODUÇÃO AO GIT E GITHUB.....	3
2. CRIANDO UMA CONTA NO GITHUB.....	9
3. CONECTANDO GIT NO VISUAL STUDIO.....	22
4. O QUE É O GIT CHANGES NO VISUAL STUDIO... .	29
5. CRIANDO UM REPOSITÓRIO NO GITHUB.....	31
6. CRIANDO UMA BRANCH.....	38
7. FAZENDO UM COMMIT.....	48
8. FAZENDO UM PUSH.....	56
9. DESFAZENDO UMA ALTERAÇÃO.....	60
10. FAZENDO UM FETCH.....	64
11. FAZENDO UM PULL.....	67
12. COMO FAZER UM MERGE.....	70
13. CONCLUSÃO.....	79

INTRODUÇÃO AO GIT E GITHUB

SUAS DIFERENÇAS





GIT

O Git é um sistema de controle de versão que permite gerenciar e acompanhar as mudanças no código-fonte de um projeto localmente



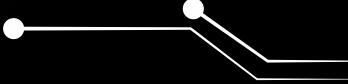
GITHUB

O GitHub é uma plataforma de hospedagem de repositórios Git online. Ele oferece ferramentas adicionais para colaboração, permitindo que equipes trabalhem juntas remotamente em projetos usando Git como base

INTRODUÇÃO AO GIT E GITHUB

DICIONÁRIO





REPOSITÓRIO LOCAL

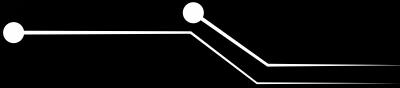
É o espaço no seu computador onde você armazena o projeto e pode fazer todas as mudanças. É como um "caderno" privado que só você tem acesso até decidir compartilhar.

REPOSITÓRIO REMOTO

É o repositório "online", onde o projeto fica guardado em servidores (exemplo: GitHub, GitLab). Ele é compartilhado, permitindo que várias pessoas colaborem no mesmo projeto.

BRANCH

É como uma linha do tempo paralela no seu projeto. Você cria uma branch para desenvolver algo sem mexer no trabalho principal (geralmente chamado de "main" ou "master").



COMMIT

É o momento de "salvar" as mudanças que você fez no código. Quando você faz um commit, você registra um ponto de progresso, e pode voltar a ele se precisar.

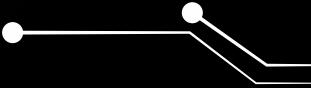
MENSAGENS DE COMMIT

São as descrições que você escreve para explicar o que foi mudado no commit. Serve para você (e para outras pessoas) saber o que foi feito naquela mudança.

PUSH

Quando você termina de fazer o commit no seu repositório local e quer enviar essas mudanças para o repositório remoto (online), você usa o push.





FETCH

O fetch é um comando para baixar as atualizações feitas no repositório remoto, mas sem integrar elas no seu projeto. Ele só pega as mudanças, sem fazer nada com elas ainda.

PULL

O fetch é um comando para baixar as atualizações feitas no repositório remoto, mas sem integrar elas no seu projeto. Ele só pega as mudanças, sem fazer nada com elas ainda.

MERGE

É o processo de juntar duas branches diferentes. Se você tem uma branch de desenvolvimento e quer juntar ela com a principal (geralmente "main"), você faz um merge para unir o código de ambos os lugares.



CRIANDO UMA CONTA NO GITHUB





PASSO 01 - ACESSSE O SITE OFICIAL DO GITHUB PELO LINK:

[HTTPS://GITHUB.COM/](https://github.com/)



A screenshot of the GitHub homepage. The header features the GitHub logo and navigation links. The main title is "Vamos construir a partir daqui" with the subtitle "A plataforma de desenvolvimento com tecnologia de IA líder no mundo." Below the title are input fields for email and password, and a "Início" button. To the right, there's a "Enviar um teste empresarial gratuito" button. At the bottom, logos of various companies are shown, including 3M, KPMG, Mercedes-Benz, SAP, P&G, and TELUS. The background has a dark theme with some abstract geometric shapes.



PASSO 02 - DIGTE SEU EMAIL NO CAMPO
“ENDERECO DE EMAIL” PARA COMEÇAR A CRIAR
SUA CONTA

Endereço de email

seuemail@email.com

Inscreve-se no GitHub

PASSO 03 - CLIQUE EM “INSCREVA-SE NO
GITHUB” PARA CONTINUAR



PASSO 04 - CONFIRME SE SEU EMAIL ESTÁ CORRETO E CLIQUE EM CONTINUAR



A screenshot of the GitHub sign-up process. The text "Welcome to GitHub! Let's begin the adventure." is displayed above a form field. The field contains the placeholder "Enter your email*" and the value "github.suporte@gmail.com". To the right of the input field is a green "Continue" button.



PASSO 05 - CRIE UMA SENHA PARA SUA CONTA E CLIQUE EM CONTINUAR

Bem-vindo ao GitHub!
Vamos começar a aventura

Insira seu e-mail *

✓ `github.suporte@gmail.com`

Criar uma senha *

→ `*****`

A senha é forte

Certifique-se de que tenha pelo menos 16 caracteres OU pelo menos 9 caracteres, incluindo um número e uma letra minúscula.

[Continuar](#)

This image shows a screenshot of the GitHub sign-up process. It's the 'Create a password' step. The user has entered their email as 'github.suporte@gmail.com'. They have also entered a password, shown as a series of asterisks. A green progress bar at the bottom indicates that the password is strong. Below the progress bar, there is a note about password requirements: 'Certifique-se de que tenha pelo menos 16 caracteres OU pelo menos 9 caracteres, incluindo um número e uma letra minúscula.' There is a large 'Continuar' (Continue) button at the bottom right.

OBS: UTILIZE UMA SENHA CONTENDO LETRAS
MAÍSCULAS E MINÚSCULAS, NÚMEROS E
CARACTERES ESPECIAIS



PASSO 06 - DIGITE UM NOME DE USUÁRIO E CLIQUE EM CONTINUAR

Bem-vindo ao GitHub!
Vamos começar a aventura.

Insira seu e-mail *

✓ `github.suporte@gmail.com`

Criar uma senha *

✓ `*****`

Digite um nome de usuário *

→ `joaoMazzoni123`

Continuar

joaoMazzoni123 está disponível.

OBS: O NOME DE USUÁRIO É COMO O SEU PERFIL SERÁ EXIBIDO PARA OUTROS USUÁRIOS DO GITHUB



PASSO 07 - MARQUE A CAIXA SE QUISER RECEBER ANÚNIOS E ATUALIZAÇÕES, CASO CONTRÁRIO, APENAS CLIQUE EM CONTINUAR

Bem-vindo ao GitHub!

Vamos começar a aventura

Insira seu e-mail *

✓ `github.suporte@gmail.com`

Criar uma senha *

✓ `*****`

Digite um nome de usuário *

✓ `joaoMazzoni123`

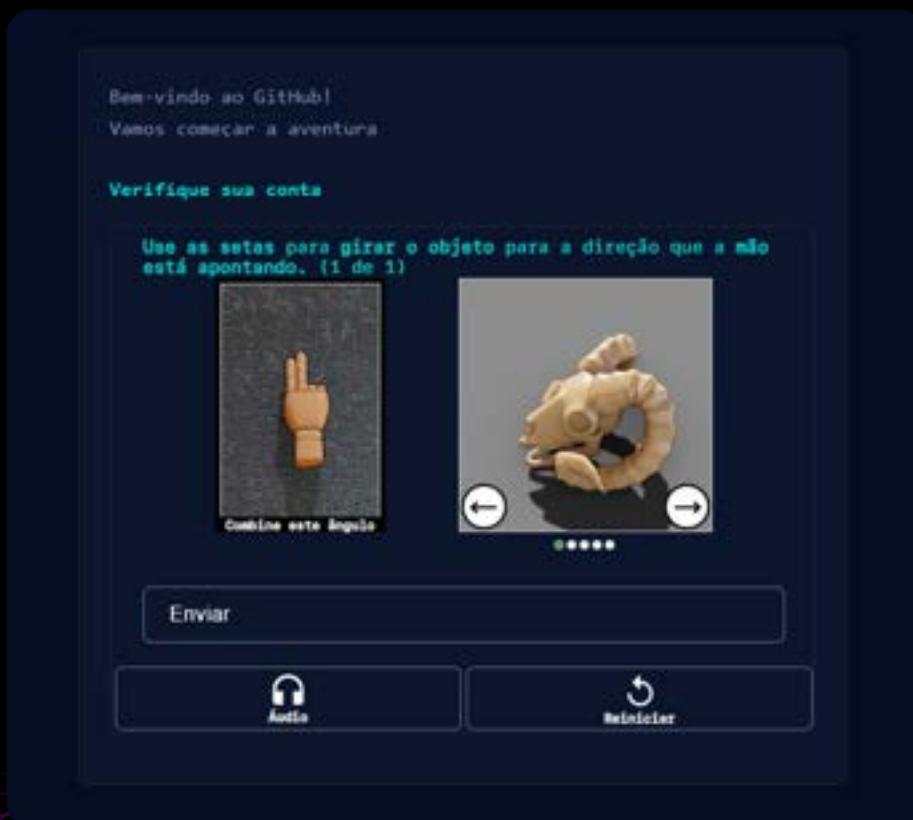
Preferências de e-mail

Receba atualizações e anúncios ocasionais de produtos.

Continuar



PASSO 08 - COMPLETE A VERIFICAÇÃO DE SEGURANÇA DO GITHUB



Bem-vindo ao GitHub!

Vamos começar a aventura.

Verifique sua conta

Use as setas para girar o objeto para a direção que a mão está apontando. (1 de 1)

Combine este ângulo

← →

Enviar

Áudio

Reiniciar

This image shows a screenshot of a mobile application interface for a security verification challenge. At the top, it says "Bem-vindo ao GitHub!" and "Vamos começar a aventura.". Below that, it says "Verifique sua conta". Then, there's a text instruction: "Use as setas para girar o objeto para a direção que a mão está apontando. (1 de 1)". Below the text are two images: one showing a hand pointing up and another showing a hand pointing right. Between them are left and right arrow buttons. Below the images is a progress bar with four green dots. At the bottom of the screen are three buttons: "Enviar" (Send), "Áudio" (Audio), and "Reiniciar" (Restart).



PASSO 09- UM CÓDIGO DE VERIFICAÇÃO SERÁ ENVIADO PARA SEU EMAIL CADASTRADO NO GITHUB

A screenshot of a dark-themed web page showing a two-factor authentication step. At the top, it says "Você está quase terminando!" and "Enviamos um código de lançamento para github-suporte@gmail.com". Below that, it says "Digite o código:" followed by eight empty square input fields. At the bottom, there is a link "Não recebeu seu e-mail? Recuse o código ou atualize seu endereço de e-mail".

Você está quase terminando!
Enviamos um código de lançamento para github-suporte@gmail.com

→Digite o código:

Não recebeu seu e-mail? [Recuse o código ou atualize seu endereço de e-mail](#).



PASSO 10 - ACESSE SEU EMAIL E COPIE O CÓDIGO ENVIADO PELO GITHUB

A screenshot of an email from GitHub showing a launch code. The email header includes the GitHub logo and the text "Here's your GitHub launch code!". Below this is a graphic of a GitHub cat and a rocket ship. The text "Continue signing up for GitHub by entering the code below:" is followed by the code "80557088". A green button labeled "Open GitHub" is present. At the bottom, there is a note about completing the sign-up process and a link to paste into a browser if unable to enter the code.

Here's your GitHub launch code!



Continue signing up for GitHub by entering the code below:

80557088

Open GitHub

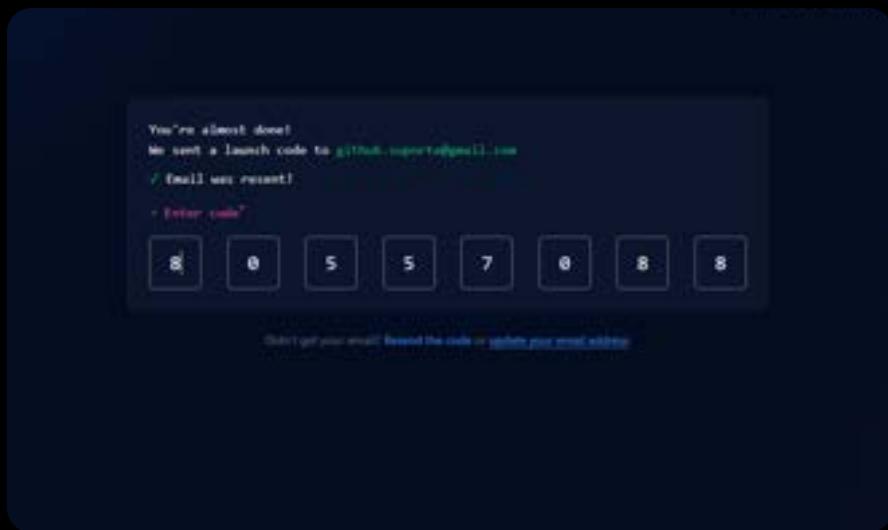
Once completed, you can start using all of GitHub's features to explore, build, and share projects.

Not able to enter the code? Paste the following link into your browser:
<https://github.com/account/verifications/confirm/24b887a6-cf90-4e6a-9bef-95c774b1d11d/b0557088>

Terms • Privacy • Sign in to GitHub

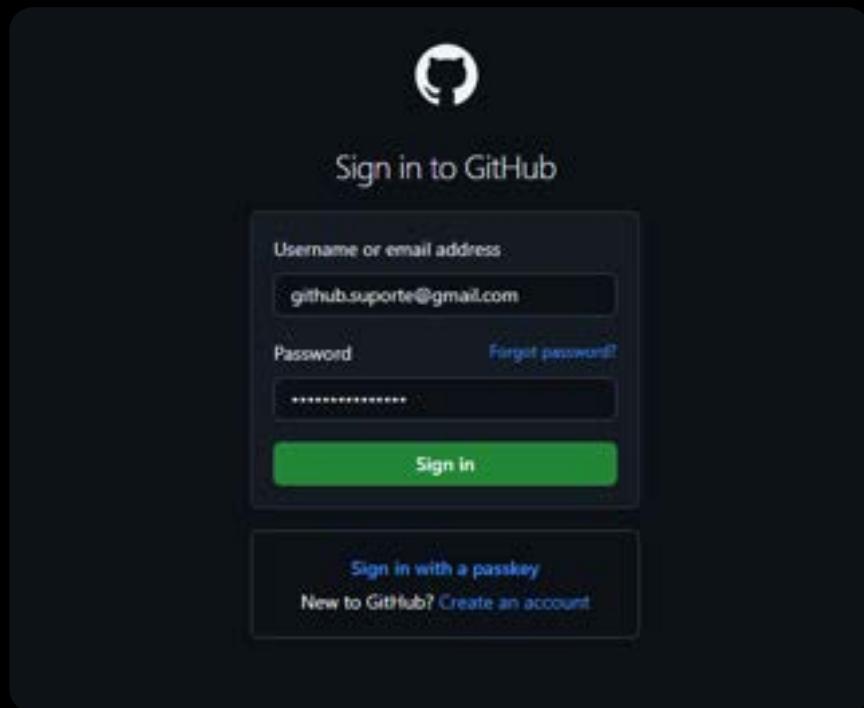


PASSO 11 - COLE O CÓDIGO NOS CAMPOS EM BRANCO





PASSO 12 - DIGITE O EMAIL E A SENHA QUE
FORAM CADASTRADOS E CLIQUE EM SIGN IN





PASSO 13 - PRONTINHO! SUA CONTA GITHUB ESTÁ CRIADA

Home

[Send feedback](#)[Filter](#)

Start writing code

Start a new repository for `joseMacconi123`

A repository contains all of your project's files, version history, and collaborator discussions.

Repository name *

name your new repository...

Public
Anyone on the Internet can see this repository.

Private
You choose who can see and comment to the repository.

[Create a new repository](#)

Introduce yourself with a profile README

Share information about yourself by creating a profile README, which appears at the top of your profile page.

`josemacconi123 / README.md` [Create](#)

1 - 🌟 I'm the `@joseMacconi123`
2 - 🌐 I'm interested in ...
3 - 📚 I'm currently learning ...
4 - 🎨 I'm looking to collaborate on ...
5 - 📧 How to reach me ...
6 - 🎯 Personal ...
7 - 🍺 Fun facts ...

Use tools of the trade

Simplify your development workflow with a GUI

Install GitHub Desktop to visualize, commit, and push changes without ever touching the command line.

Get AI-based coding suggestions

Try GitHub Copilot free for 30 days, which suggests entire functions in real-time, right from your editor.



CONECTANDO O GITHUB NO VISUAL STUDIO





PASSO 01 - CASO AINDA NÃO TENHA O VISUAL STUDIO INSTALADO, ACESSE O LINK ABAIXO:

[HTTPS://VISUALSTUDIO.MICROSOFT.COM/PT-BR/DOWNLOADS/](https://visualstudio.microsoft.com/pt-br/downloads/)

Downloads



Visual Studio 2022 | Windows

O IDE mais abrangente para desenvolvedores .NET e C++ no Windows para criação de web, nuvens, desktop, aplicativos móveis, serviços e jogos.

Versão prévia

Obtenha acesso antecipado aos recursos mais recentes que ainda não estão na versão principal.

[Saiba mais →](#)

Comunidade

IDE poderoso, gratuito para estudantes, colaboradores de código aberto e instituições

[Download gratuito](#)

Profissional

IDE profissional mais adequado para equipes pequenas

[Teste gratuito](#)

Enterprise

Solução escalável e completa para equipes de qualquer tamanho

[Teste gratuito](#)



[Notas sobre a versão →](#)

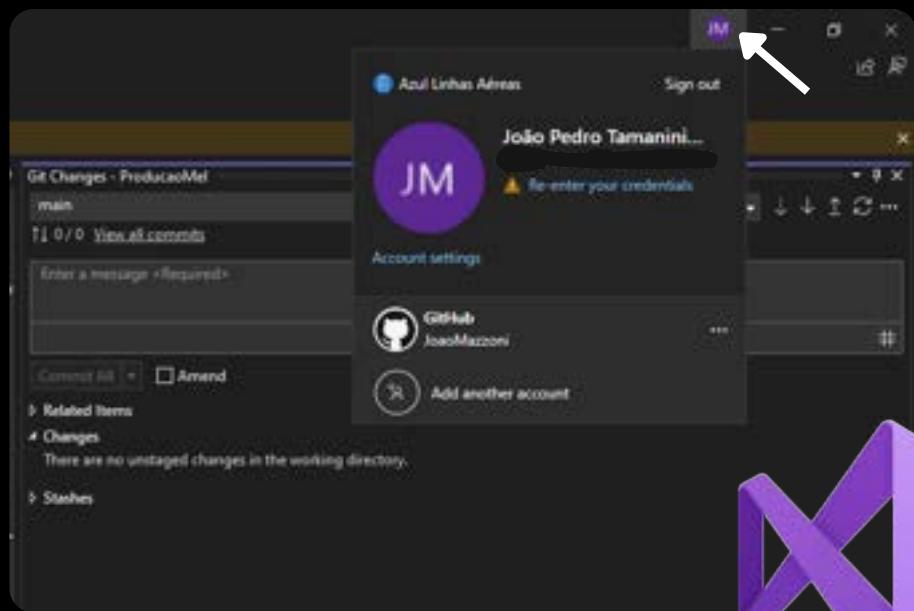
[Comparar Edições →](#)

[Como fazer a instalação offline →](#)

[Termos de Licença →](#)

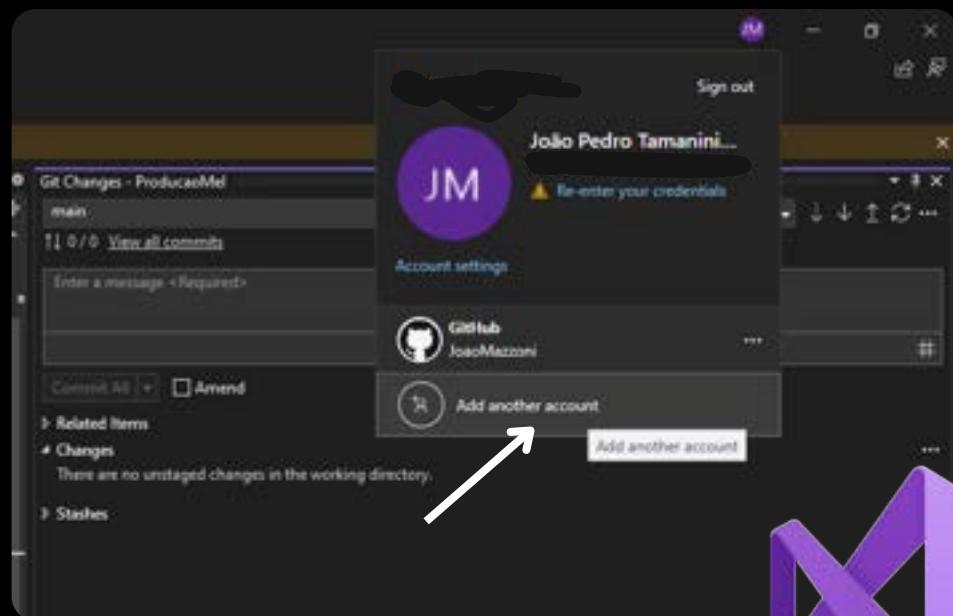


PASSO 02 - COM O VISUAL STUDIO ABERTO
CLIQUE NO ÍCONE DE USUÁRIO NO CANTO
SUPERIOR DIREITO



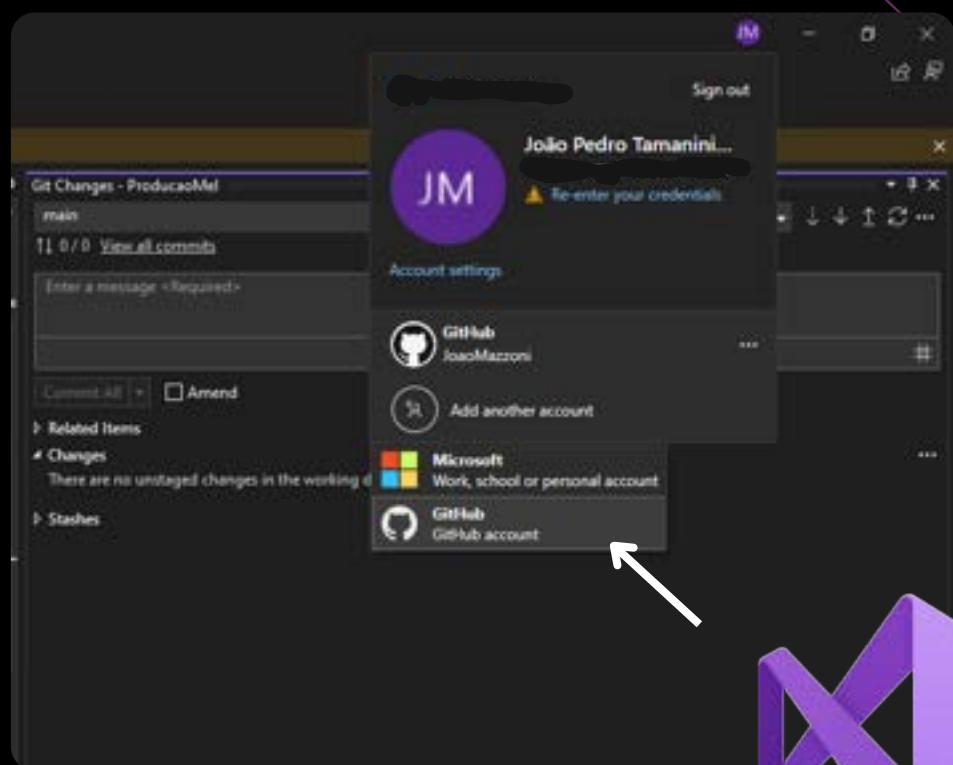


PASSO 03 - NA JANELA ABERTA, CLIQUE EM
“ADD ANOTHER ACCOUNT”



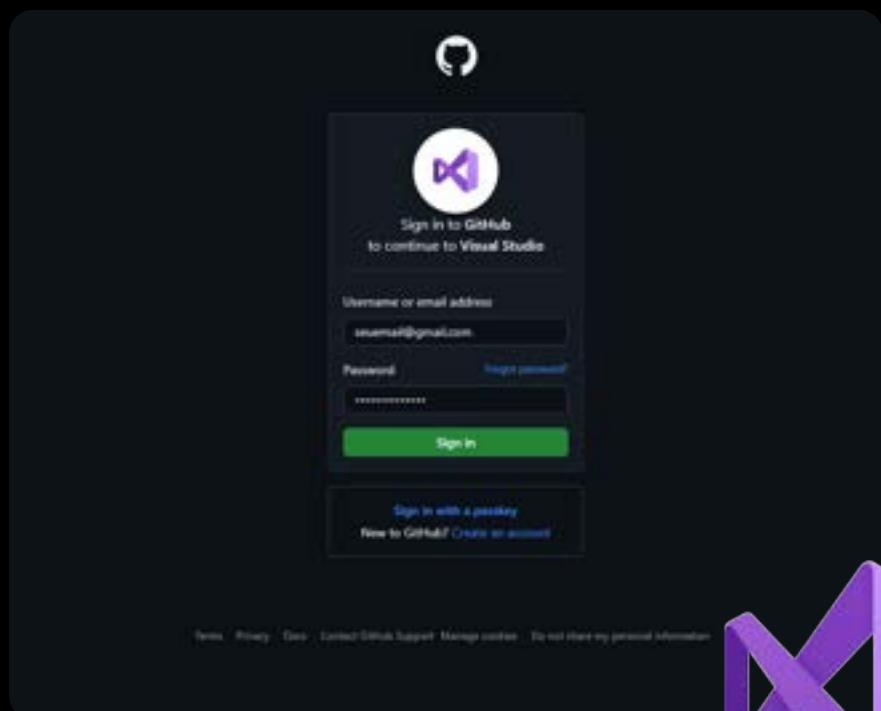


PASSO 04 - CLIQUE EM "GITHUB"





PASSO 05 – UMA JANELA ABRIRÁ NO NAVEGADOR
DIGITE SEU LOGIN DO GITHUB E CLIQUE EM
“SIGN IN”





PASSO 06 - PROTINHO!
AGORA SUA CONTA GITHUB ESTÁ VINCULADA NO
VISUAL STUDIO



Sucesso!

Sua autorização foi bem-sucedida. Agora você pode retornar ao Visual Studio.



O QUE É O GIT CHANGES NO VISUAL STUDIO?



GIT CHANGES



O Git Changes no Visual Studio é uma ferramenta integrada que facilita o uso do Git diretamente no ambiente de desenvolvimento. Ele serve para gerenciar mudanças no código, como adicionar, confirmar (commit), sincronizar e enviar alterações para repositórios DO GitHub.

FACILIDADE COM GIT

O Git Changes no Visual Studio facilita o uso do Git com:

- 1- Visualização das mudanças no código em tempo real.
- 2-Comandos Git integrados (commit, push, pull) sem sair do editor.
- 3-Gerenciamento fácil de branches e merges.
- 4-Comparação rápida de versões e histórico.
- 5-Feedback imediato para resolver conflitos e
- 6-erros



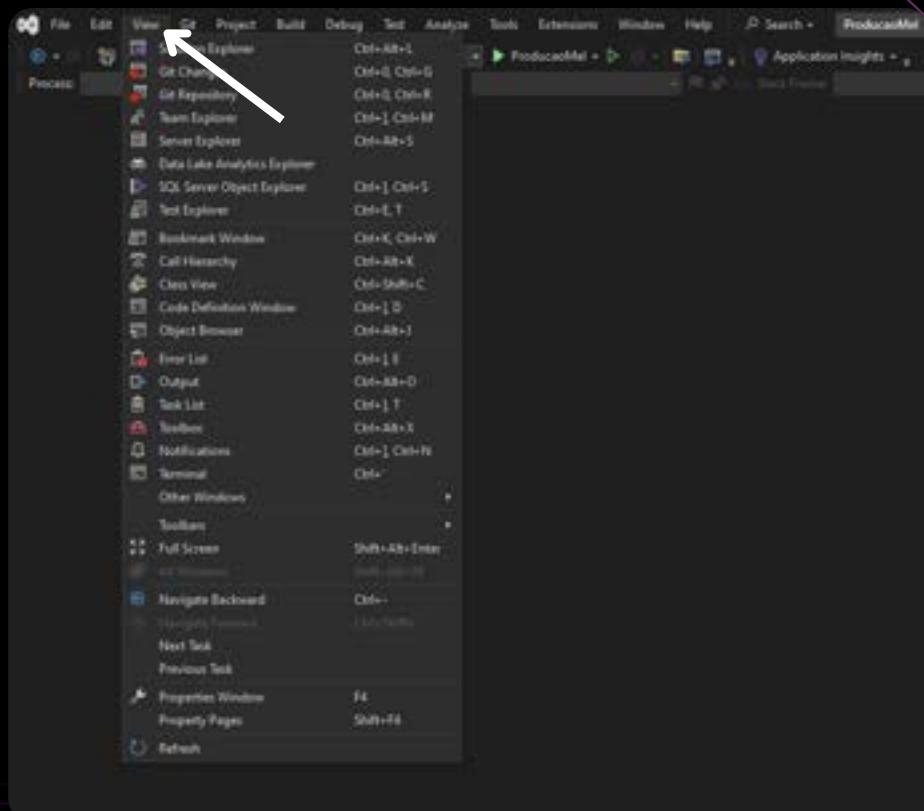


CRIANDO UM REPOSITÓRIO NO GITHUB



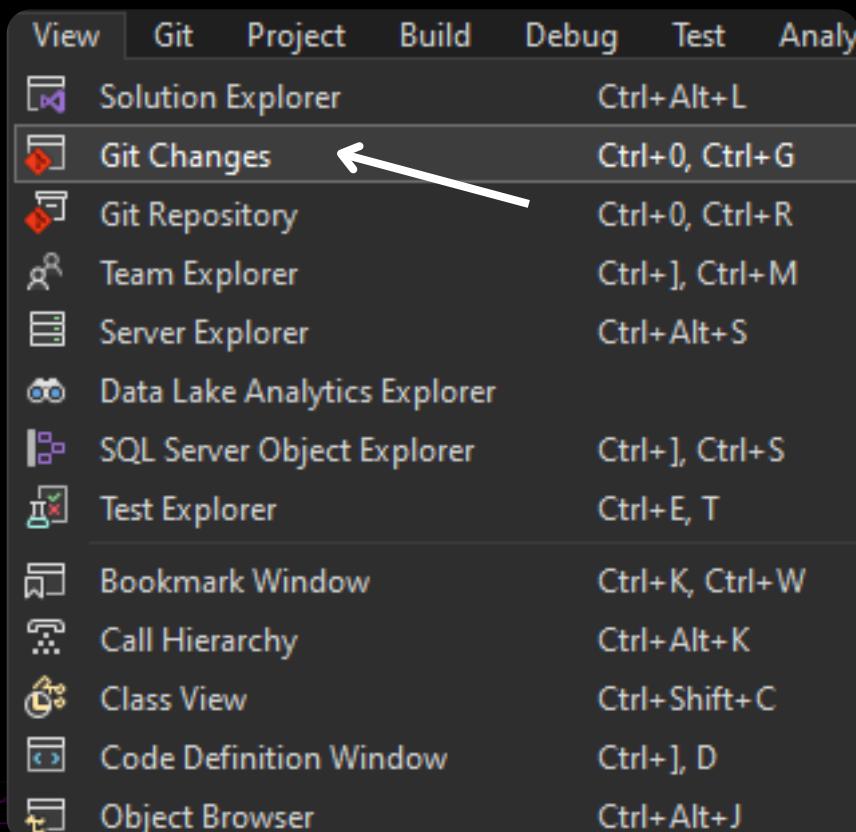
CRIANDO UM REPOSITÓRIO NO GITHUB

PASSO 01 - NO MENU SUPERIOR DO VISUAL STUDIO CLIQUE NA OPÇÃO “VIEW”



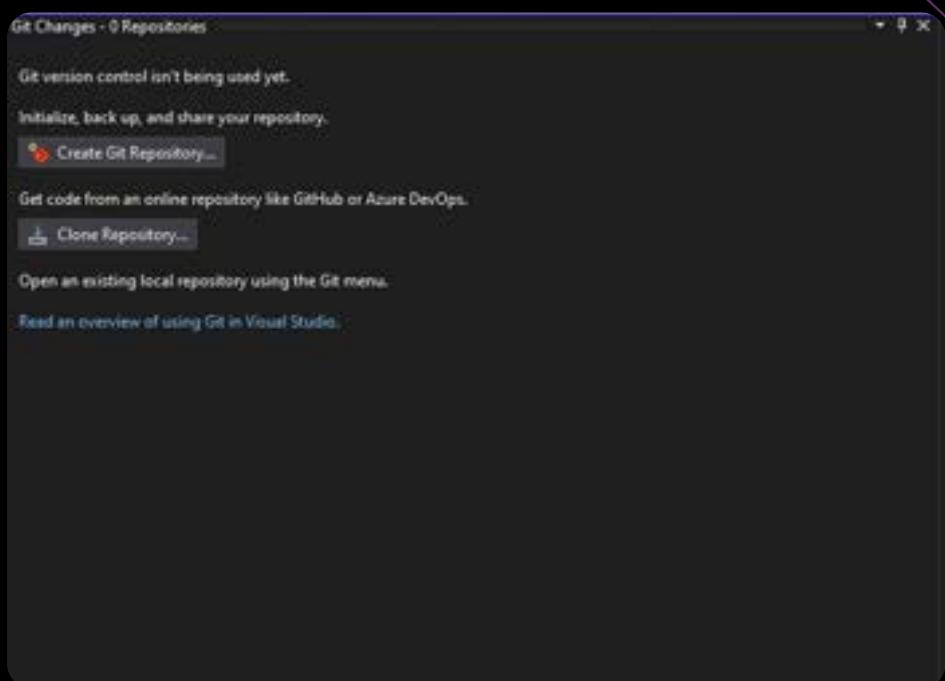
CRIANDO UM REPOSITÓRIO NO GITHUB

PASSO 02 - CLIQUE NA SEGUNDA OPÇÃO “GIT CHANGES”



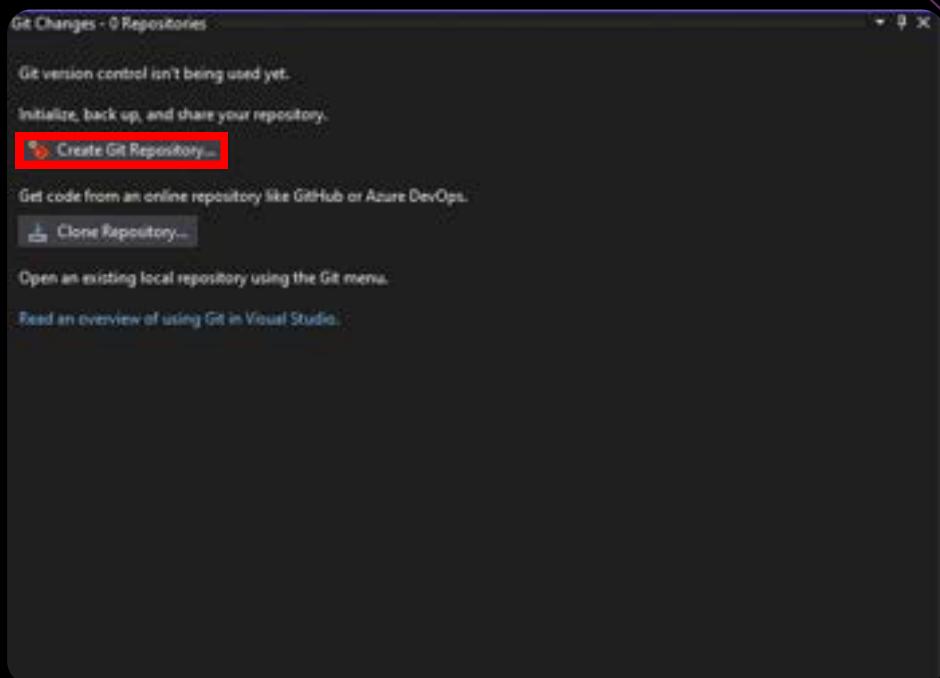
CRIANDO UM REPOSITÓRIO NO GITHUB

PASSO 03 - UMA JANELA COMO ESSA SERÁ ABERTA NO SEU VISUAL STUDIO



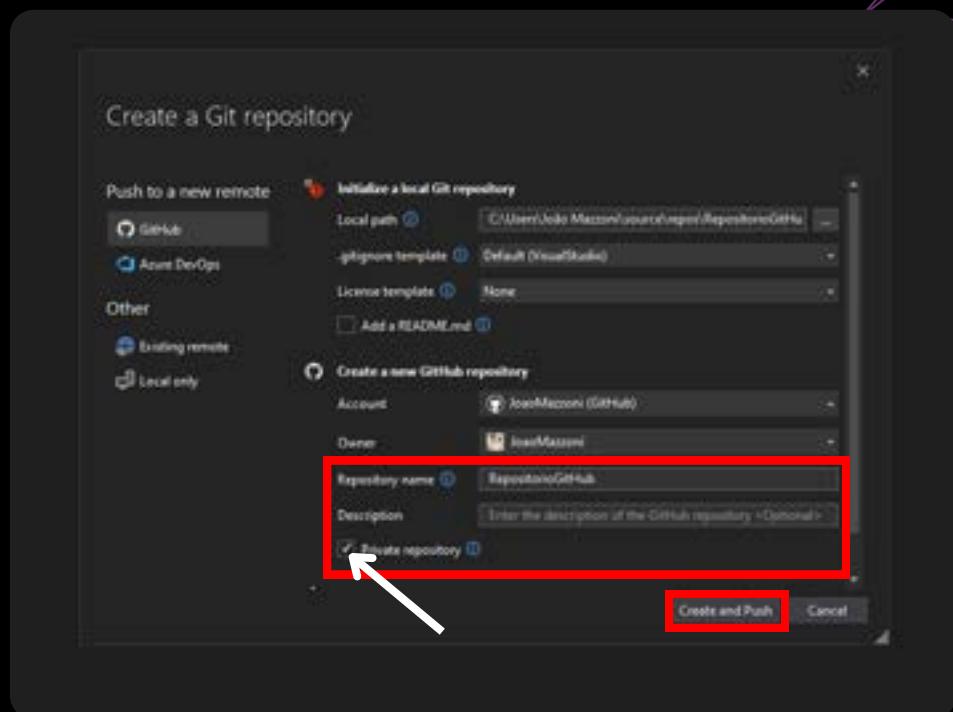
CRIANDO UM REPOSITÓRIO NO GITHUB

PASSO 04 - CLIQUE EM “CREATE A GIT REPOSITORY”



CRIANDO UM REPOSITÓRIO NO GITHUB

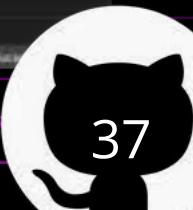
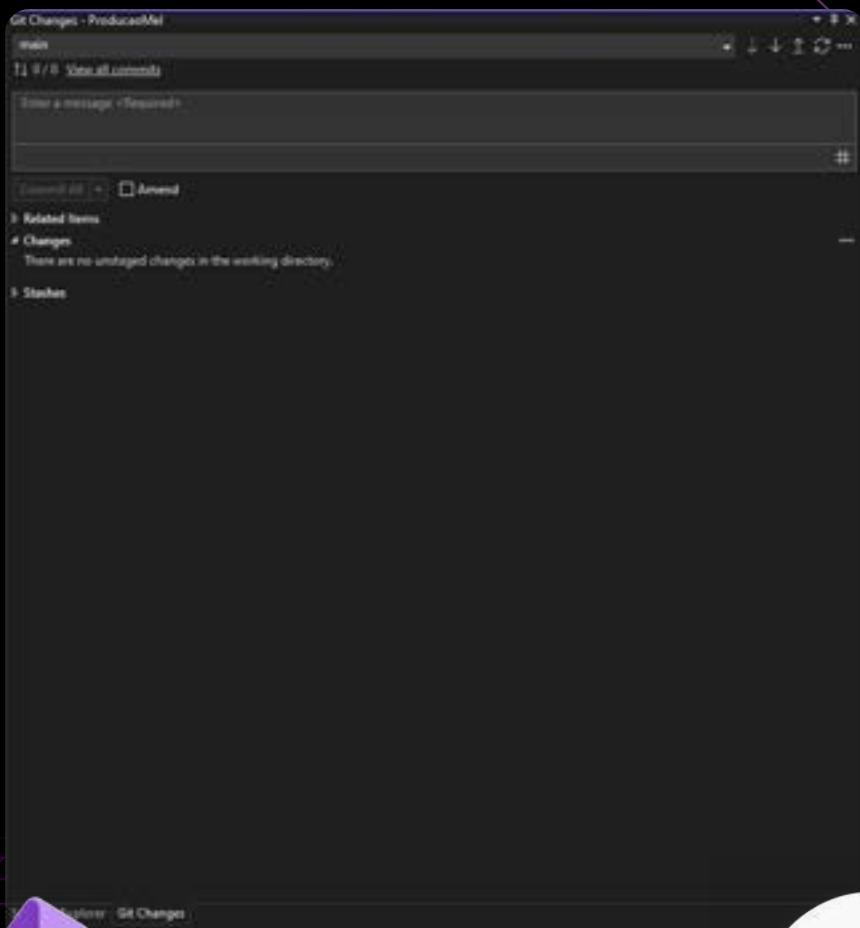
PASSO 05 - DIGITE O NOME DE SEU REPOSITÓRIO E UMA DESCRIÇÃO (CASO DESEJAR)



PASSO 06 - MARQUE A CAIXA "PRIVATE REPOSITORY" PARA UM REPOSITÓRIO PRIVADO OU DESMARQUE PARA DEIXAR PÚBLICO CLIQUE EM "CREATE AND PUSH"

CRIANDO UM REPOSITÓRIO NO GITHUB

PASSO 07 - PRONTINHO!
SEU REPOSITÓRIO FOI CRIADO LOCALMENTE E
REMOTAMENTE NO GITHUB



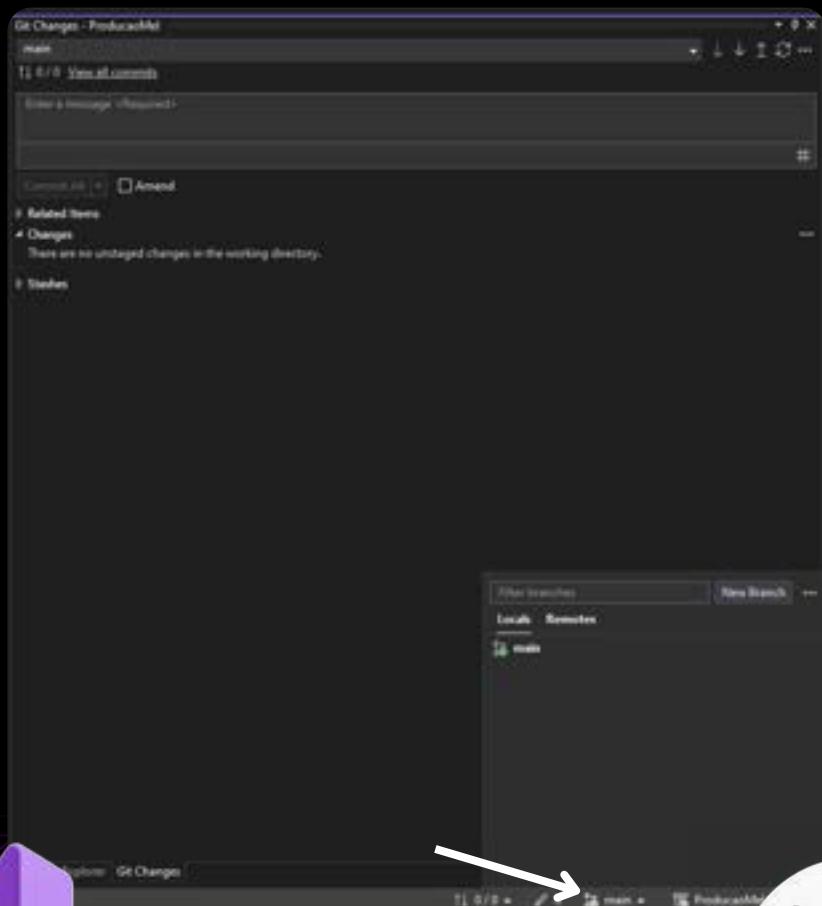


CRIANDO UMA BRANCH



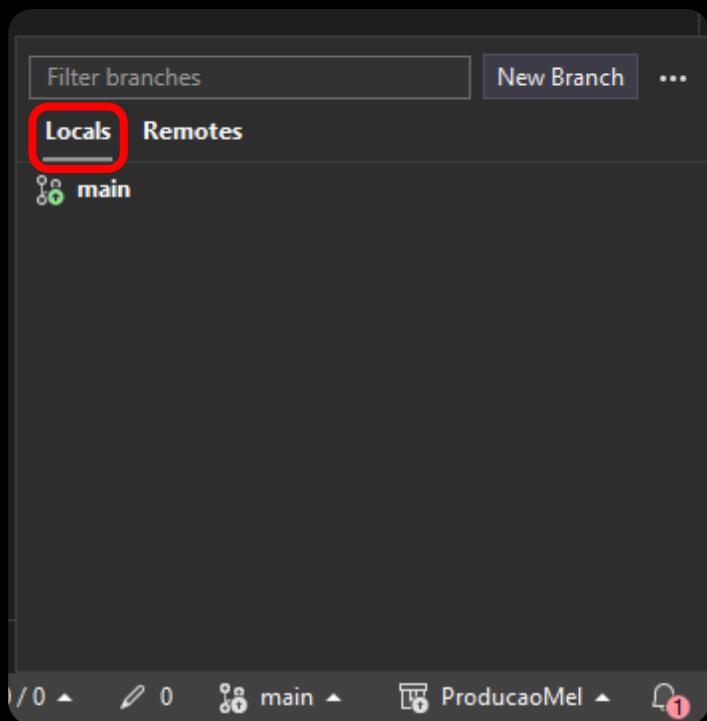
criando uma branch

PASSO 01 - NO CANTO INFERIOR DIREITO DA JANELA DO GIT CHANGES CLIQUE EM "MAIN"



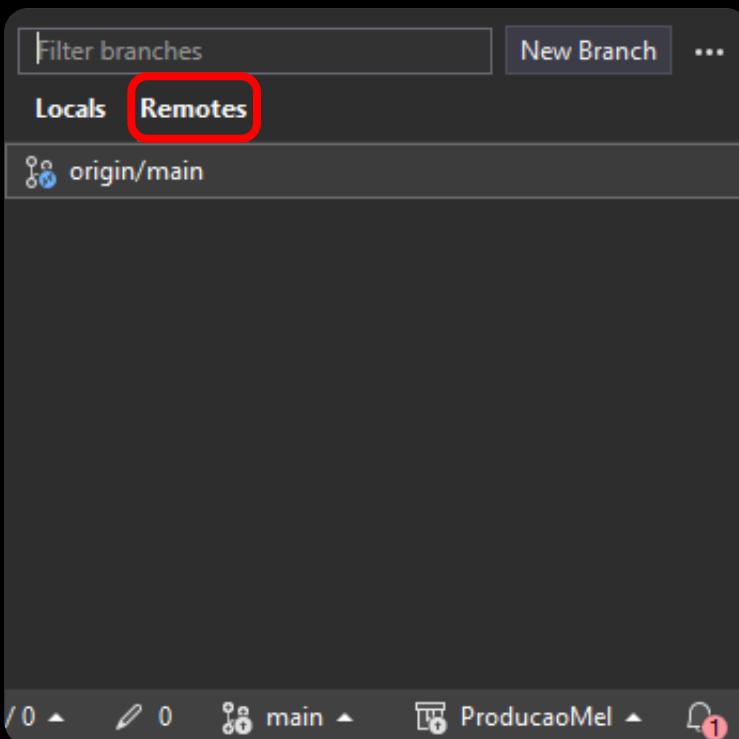
CRIANDO UMA BRANCH

PASSO 02 - NESSA JANELINHA É POSSÍVEL VISUALIZAR TODAS AS SUA BRANCHS LOCAIS CRIADAS PARA AQUELE PROJETO



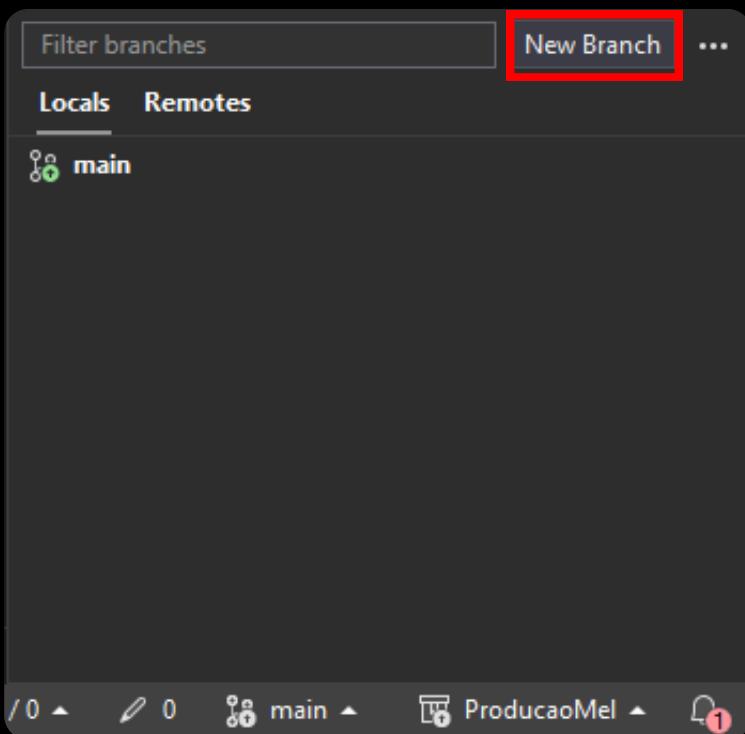
CRIANDO UMA BRANCH

PASSO 03 - NA ABA “REMOTES” É POSSÍVEL VISUALIZAR AS BRANCHS QUE FORAM ENVIADAS PARA O REPOSITÓRIO DO GITHUB



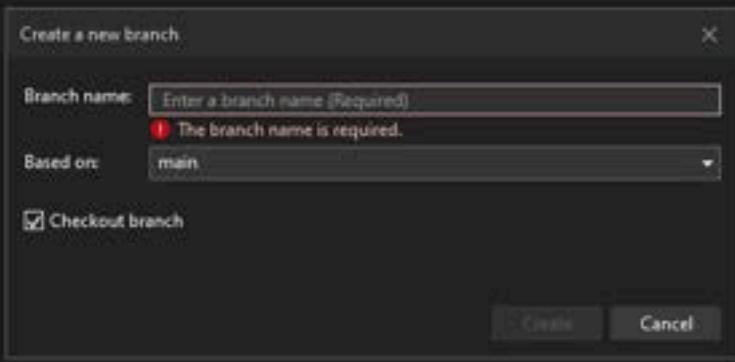
CRIANDO UMA BRANCH

PASSO 04 - CLIQUE EM “NEW BRANCH” PARA CRIAR UMA NOVA BRANCH



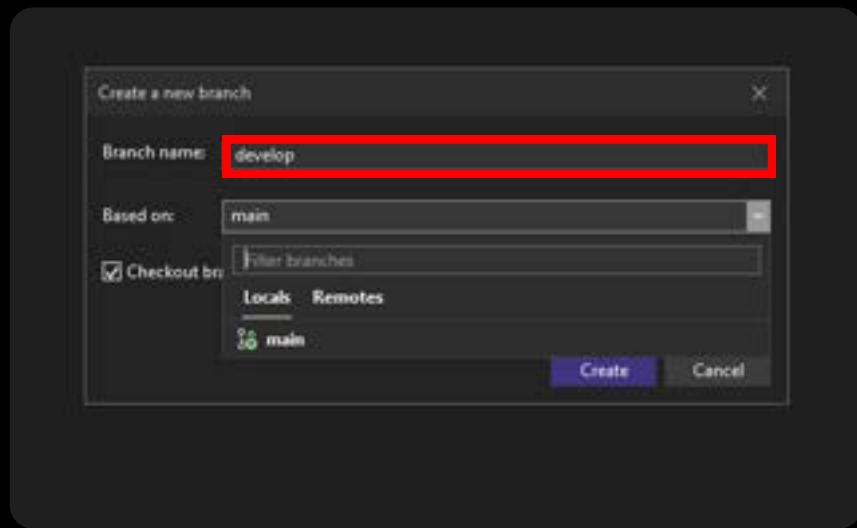
CRIANDO UMA BRANCH

PASSO 05 - UMA JANELA SERÁ ABERTA PARA A CRIAÇÃO DA BRANCH LOCAL NOVA



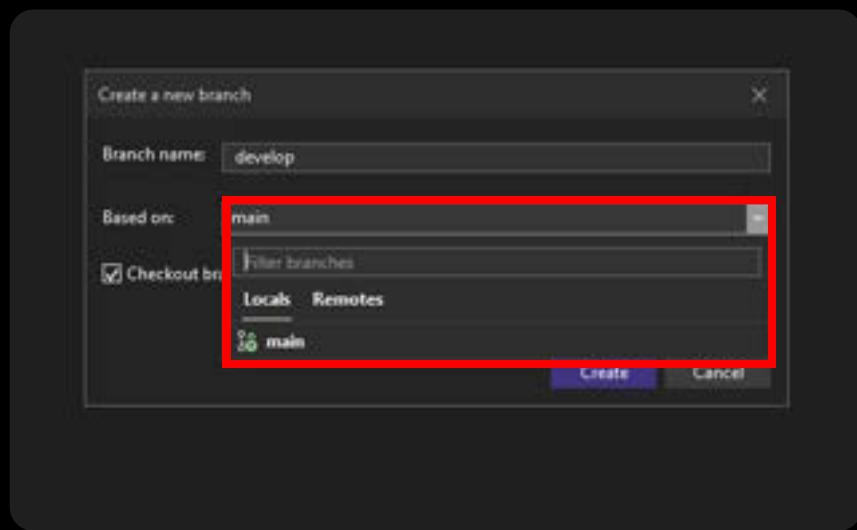
CRIANDO UMA BRANCH

PASSO 06 - EM "BRANCH NAME" DIGITE O NOME DA BRANCH NOVA



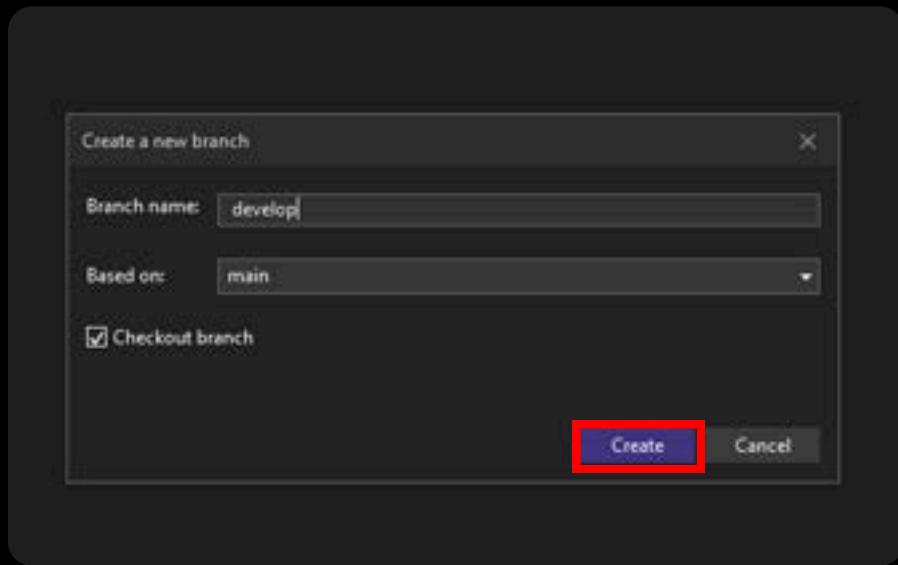
CRIANDO UMA BRANCH

PASSO 07 - EM “BASED ON” VOCÊ IRÁ SELECIONAR QUAL BRANCH SERÁ COPIADA COMO BASE PARA FAZER A NOVA



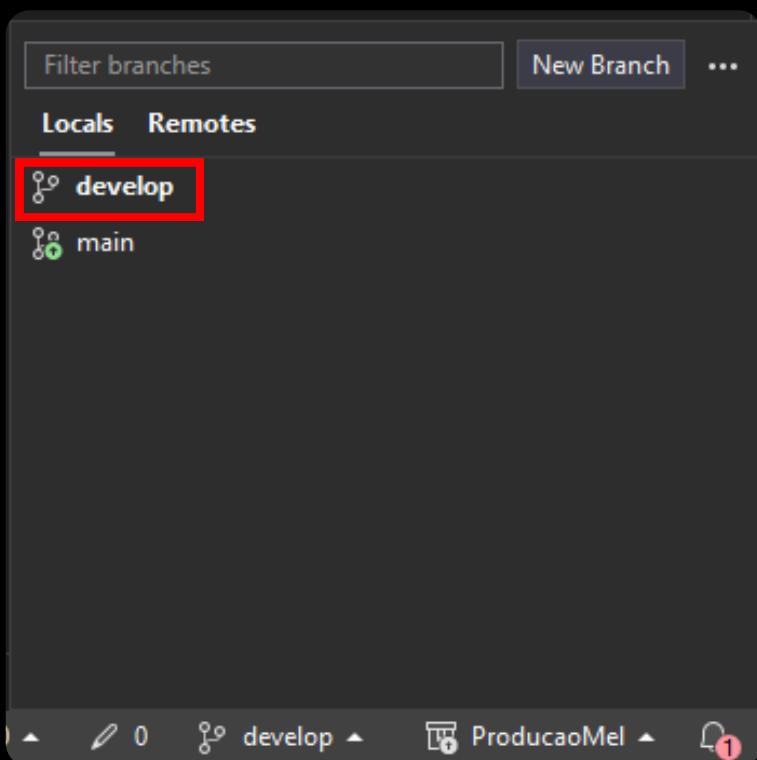
CRIANDO UMA BRANCH

PASSO 08 - CLIQUE EM "CREATE" PARA CRIAR A BRANCH NOVA



CRIANDO UMA BRANCH

PASSO 09 - PRONTINHO!
A BRANCH NOVA ESTÁ CRIADA, AGORA VOCÊ PODE SELECIONAR EM QUAL DESEJA REALIZAR AS ALTERAÇÕES

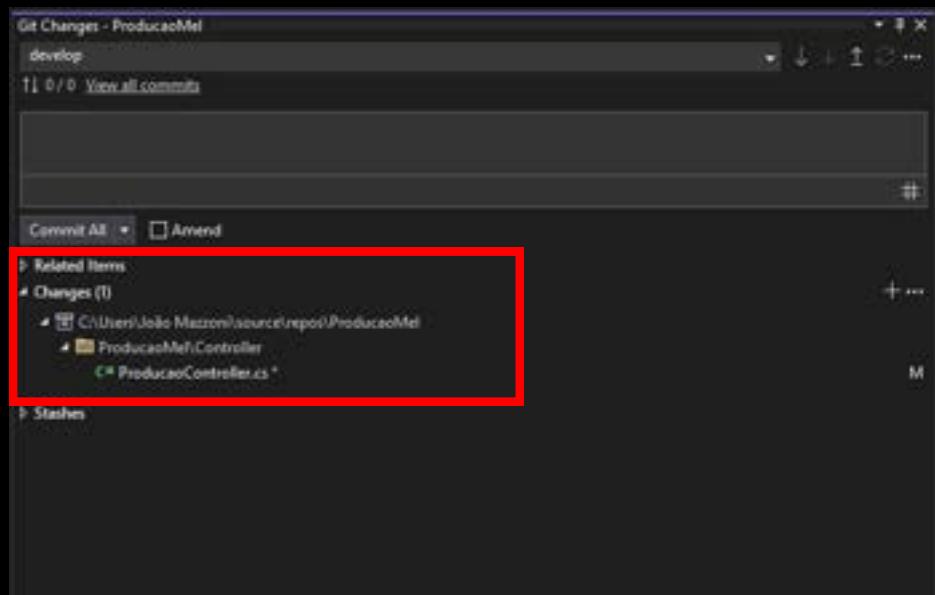


FAZENDO UM COMMIT



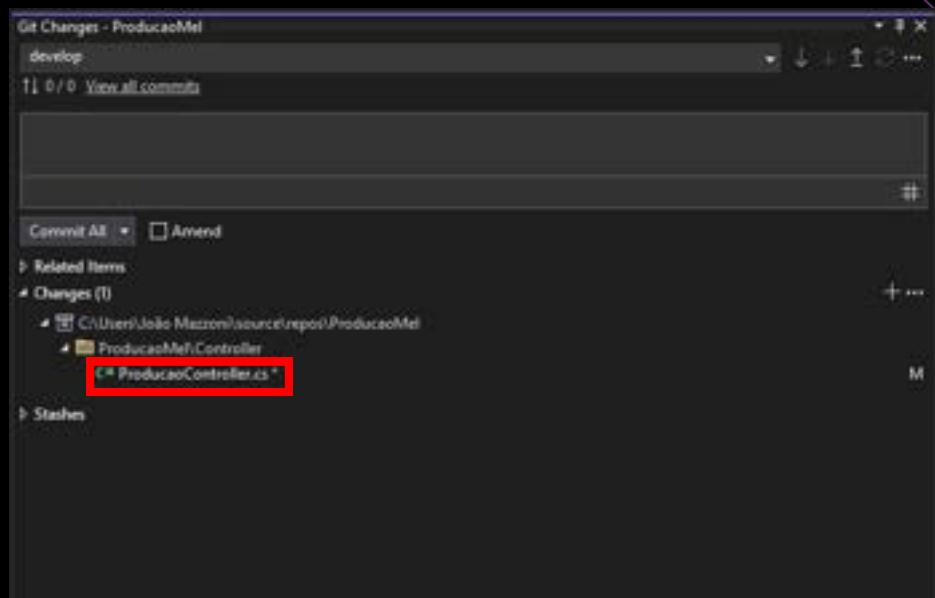
FAZENDO UM COMMIT

PASSO 01 - QUANDO UMA ALTERAÇÃO É FEITA NO SEU PROJETO, AS MUDANÇAS E ARQUIVOS ALTERADOS SERÃO EXIBIDOS AQUI



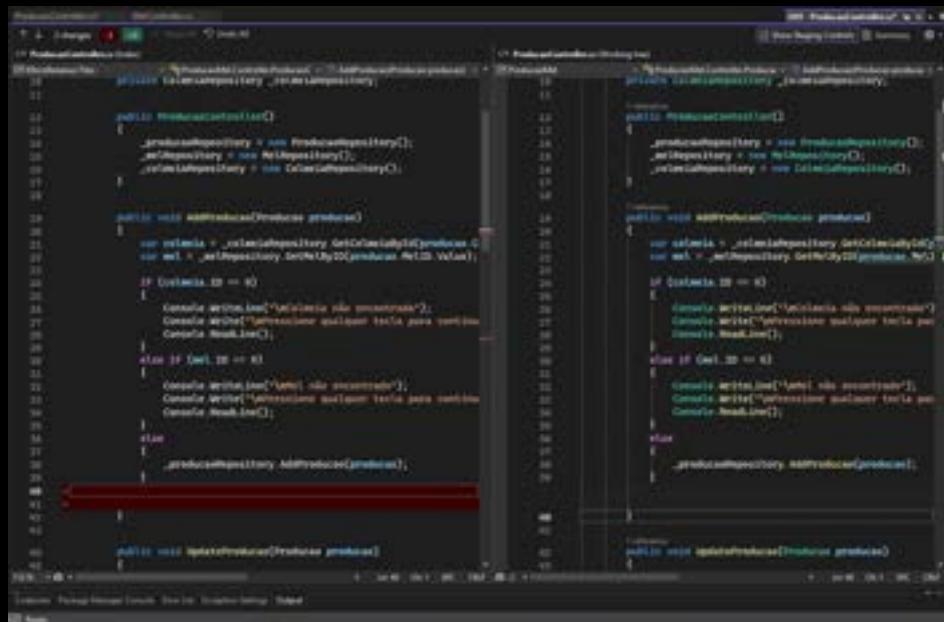
FAZENDO UM COMMIT

PASSO 02 - VOCÊ PODE CLICAR NO ARQUIVO MODIFICADO PARA VER SUAS ALTERAÇÕES



FAZENDO UM COMMIT

PASSO 03 - O LADO ESQUERDO MOSTRA COMO O ARQUIVO ERA ANTES E O LADO DIREITO COMO ELE FICOU APÓS A ALTERAÇÃO



```
private void addProduco(Produtor produtor) {
    _produtodrepository <-- Produtodrepository();
    _categoriarepository <-- Categoriarepository();
    _produtorepository <-- Produtorepository();
}

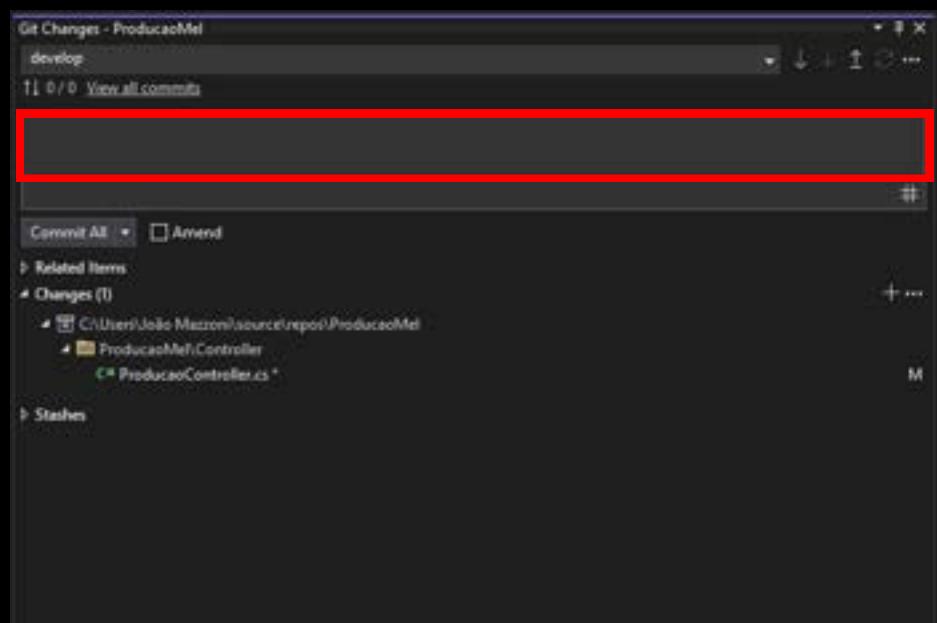
public void addProduco(Produtore produtore) {
    var categoria = _categoriarepository.getByName(produtore.categoria);
    var val = _produtorepository.getByName(produtore.refId, Val);
    if (val == null) {
        Console.println("Custos da categoria não encontrada");
        Console.println("Insira uma categoria qualquer tecida para continua");
        Console.readLine();
    } else if (val == null) {
        Console.println("Val não encontrado");
        Console.println("Insira um novo valor tecida para continua");
        Console.readLine();
    } else {
        _produtorepository.addProdutore(produtore);
    }
}

public void UpdateProdutore(Produtore produtore) {
}

public void UpdateProdutore(Produtore produtore) {
}
```

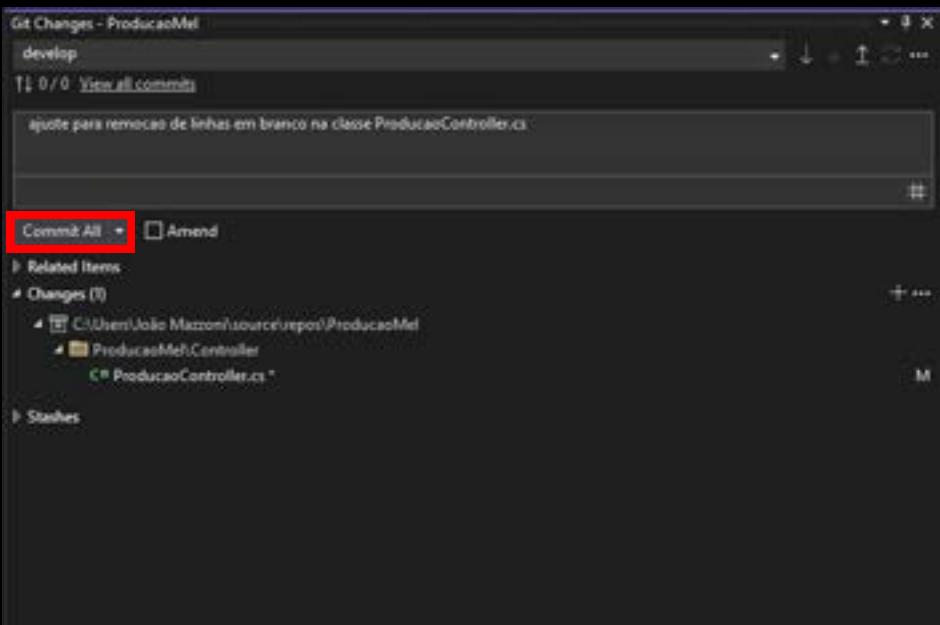
FAZENDO UM COMMIT

PASSO 04 - NESTE CAMPO, DIGITE A MENSAGEM DO SEU COMMIT, DESCREVENDO O QUE FOI MODIFICADO EM SEU PROJETO DE FORMA OBJETIVA



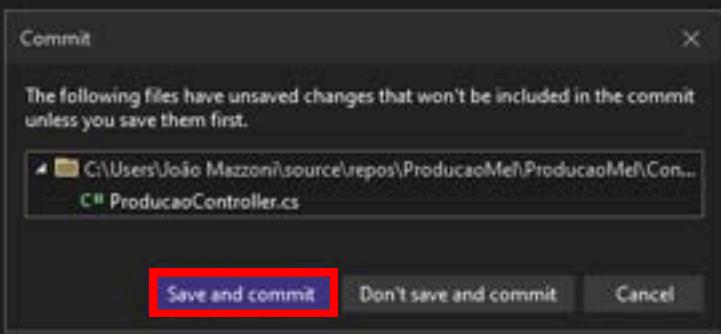
FAZENDO UM COMMIT

PASSO 05 - APÓS ESCREVER A MENSAGEM
CLIQUE EM “COMMIT ALL”



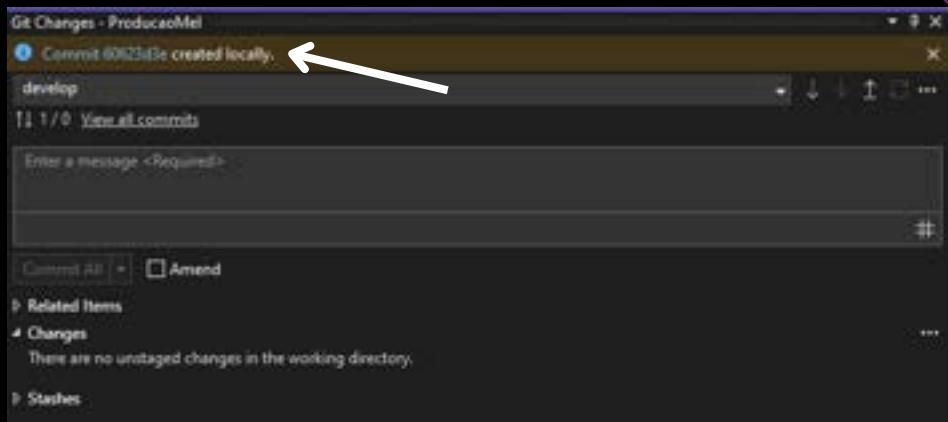
FAZENDO UM COMMIT

PASSO 06 - CASO UMA JANELA SEJA ABERTA,
CLIQUE EM “SAVE AND COMMIT”

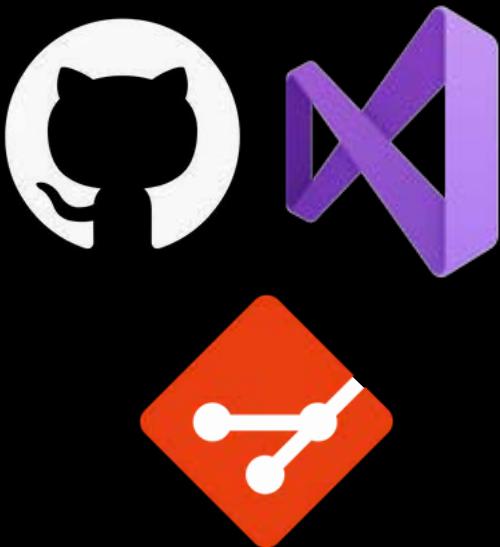


FAZENDO UM COMMIT

PASSO 07- PRONTINHO!
UMA MENSAGEM SERÁ EXIBIDA MOSTRANDO QUE
SEU COMMIT FOI FEITO COM SUCESSO



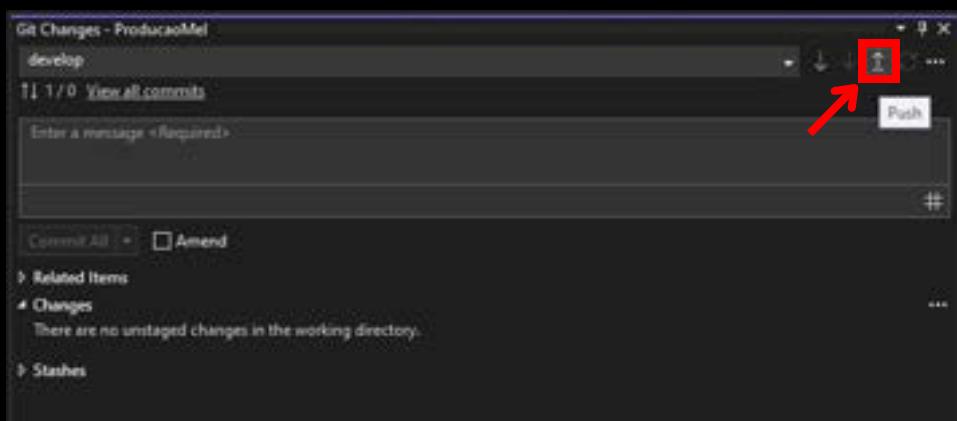
FAZENDO UM PUSH



FAZENDO UM PUSH

AGORA VAMOS ENVIAR O COMMIT PARA O REPOSITÓRIO REMOTO!

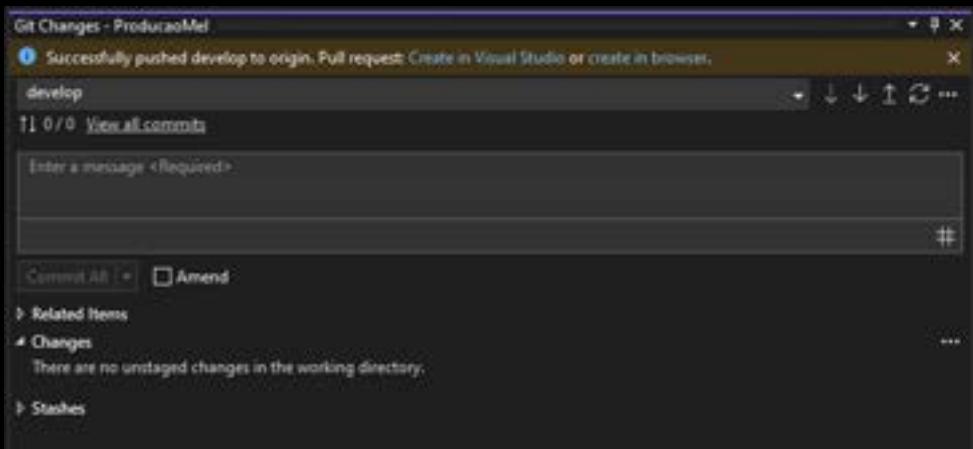
PASSO 01 - NO CANTO SUPERIOR DIREITO DO GIT CHANGES CLIQUE NA “SETA PARA CIMA” ONDE É EXIBIDO A PALAVRA “PUSH”



FAZENDO UM PUSH

PASSO 02 - PRONTINHO!

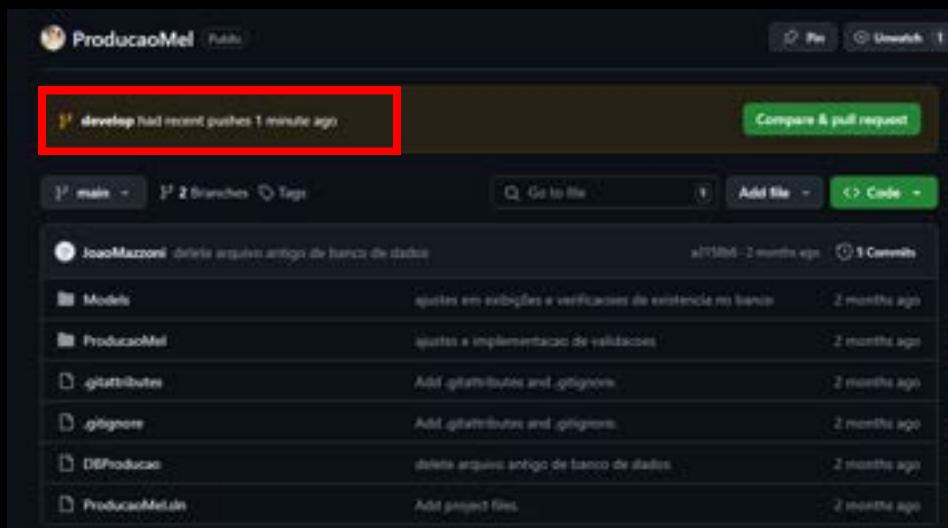
UMA MENSAGEM SERÁ EXIBIDA MOSTRANDO QUE
SEU PUSH FOI FEITO COM SUCESSO



A screenshot of a Git Changes interface in a dark-themed application. The title bar says "Git Changes - ProducaoMed". A yellow notification bar at the top states "Successfully pushed develop to origin. Pull request: Create in Visual Studio or create in browser." Below this, a list shows a single commit: "develop" with "1 0 / 0 View all commits". There is a text input field for a commit message with placeholder text "Enter a message <Required>". At the bottom, there are buttons for "Commit All" and "Amend", and sections for "Related Items", "Changes" (which notes "There are no unstaged changes in the working directory"), and "Stashes".

FAZENDO UM PUSH

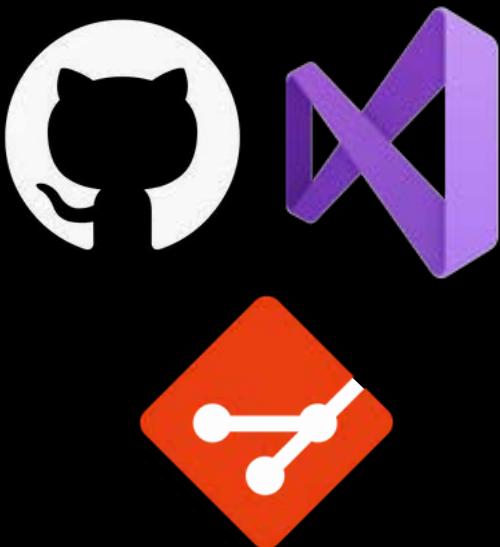
PASSO 03 – AGORA O REPOSITÓRIO COM AS ALTERAÇÕES ESTÃO NO GIT HUB



The screenshot shows a GitHub repository named 'ProducaoMel' with a public status. A red box highlights the top notification bar which says '1 push to develop just now pushed 1 minute ago'. Below the header, there are dropdown menus for 'main' branch, '2 Branches', and 'Tags', along with search and file addition buttons. The main area displays a list of recent commits:

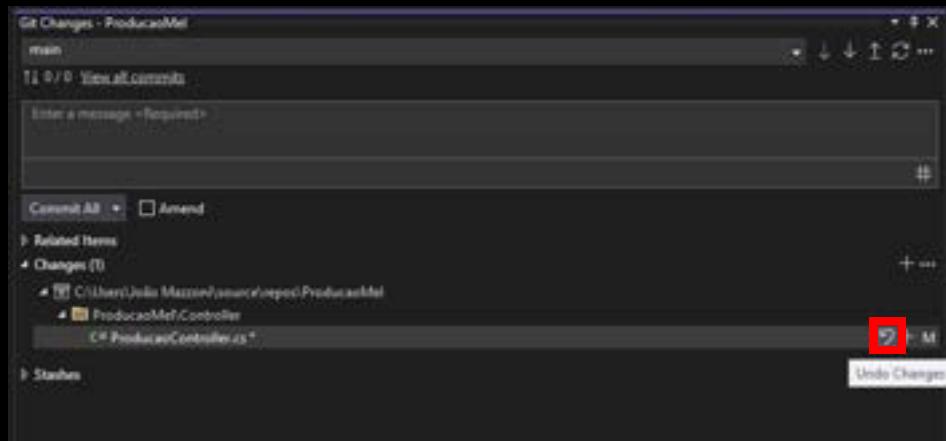
Author	Commit Message	Date	Commits
JoaoMazzoni	deletar arquivo antigo de banco de dados	at 11:56 · 2 months ago	1 Commit
Model	ajustes em exigências e verificações de existência no banco	2 months ago	
ProducaoMel	ajustes e implementação de validações	2 months ago	
.gitattributes	Add .gitattributes and .gitignore	2 months ago	
.gitignore	Add .gitattributes and .gitignore	2 months ago	
DBProducao	deletar arquivo antigo de banco de dados	2 months ago	
ProducaoMel	Add project files.	2 months ago	

DESFAZENDO UMA ALTERAÇÃO



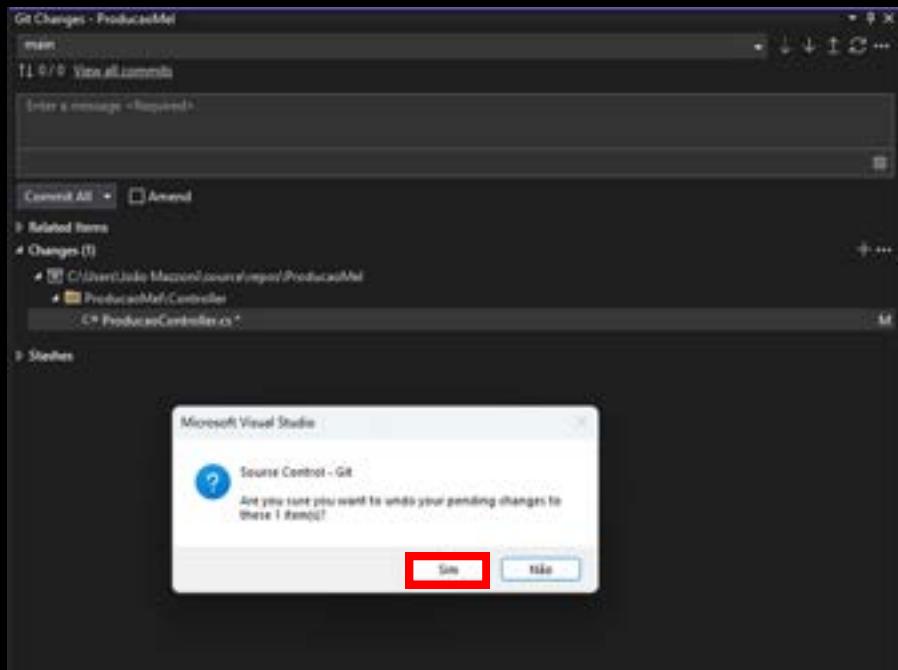
DESFAZENDO UMA ALTERAÇÃO

PASSO 01 – CASO ALGUMA ALTERAÇÃO PRECISE SER DESFEITA, CLIQUE NA SETA “UNDO CHANGES” DO ARQUIVO DESEJADO PARA RETORNAR AO COMMIT ANTERIOR DAQUELE ARQUIVO



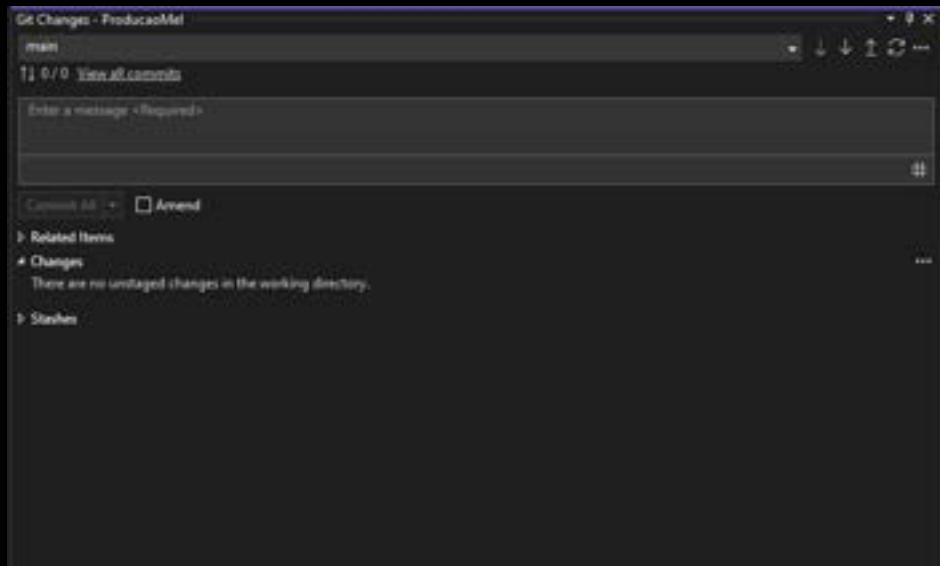
DESFAZENDO UMA ALTERAÇÃO

PASSO 02 - CLIQUE EM "SIM" PARA CONFIRMAR



DESFAZENDO UMA ALTERAÇÃO

PASSO 03 - PRONTINHO!
AS ALTERAÇÕES FORAM DESFEITAS

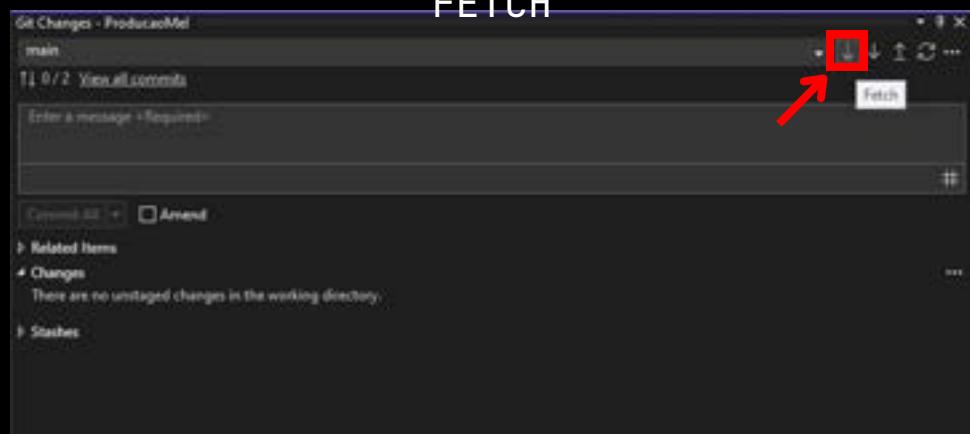


FAZENDO UM FETCH



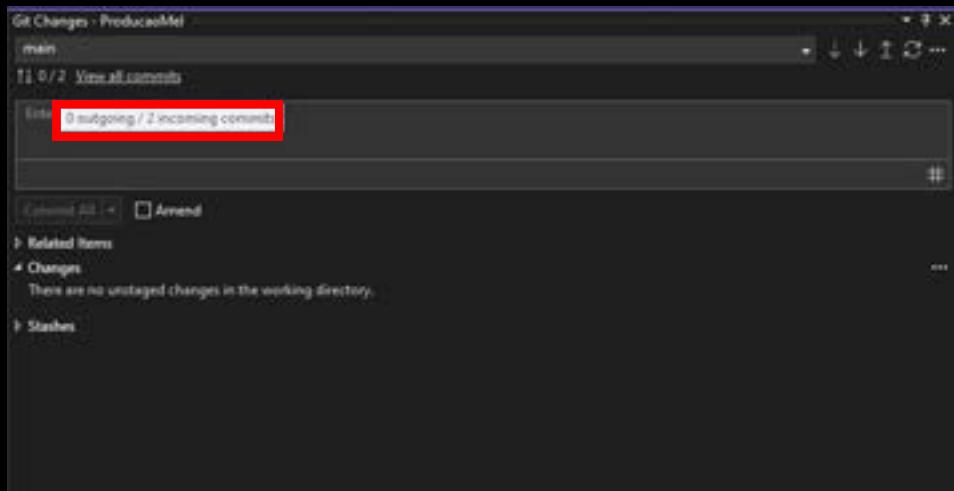
FAZENDO UM FETCH

PASSO 01 - NO CANTO SUPERIOR DIRETO DO
GIT CHANGES CLIQUE NA "SETA PONTILHADA
PARA BAIXO" ONDE É EXIBIDO A PALAVRA
"FETCH"



FAZENDO UM FETCH

PASSO 02 - O FETCH TRARÁ ALTERAÇÕES DO REPOSITÓRIO REMOTO DO GITHUB, MAS NÃO AS MESCLARÁ NO CÓDIGO ATUAL AUTOMATICAMENTE



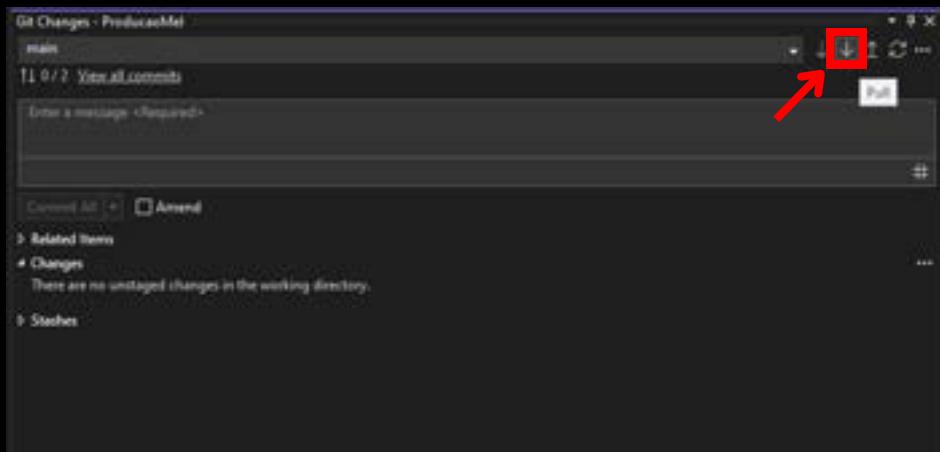
O “INCOMING” INDICA QUE HÁ 2 ALTERAÇÕES NOVAS VINDAS DO REPOSITÓRIO REMOTO, AGUARDANDO PARA SEREM MESCLADAS NO CÓDIGO ATUAL

FAZENDO UM PULL



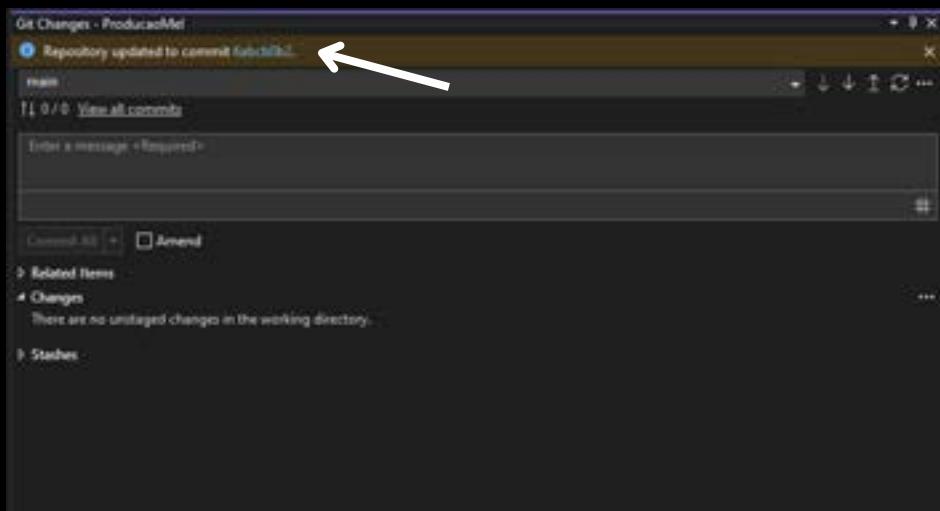
FAZENDO UM PULL

PASSO 01 - NO CANTO SUPERIOR DIREITO DO
GIT CHANGES CLIQUE NA “SETA PARA BAIXO”
ONDE É EXIBIDO A PALAVRA “PULL”



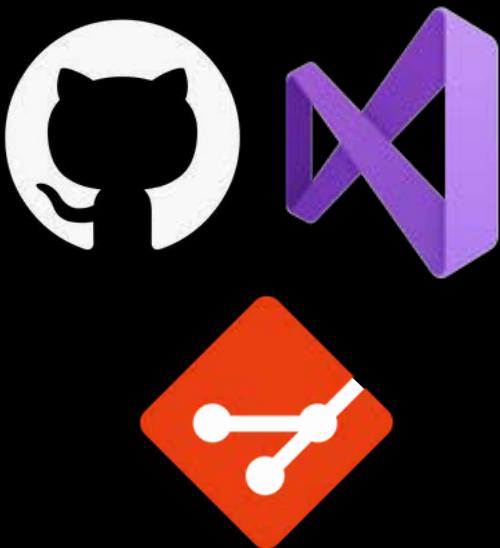
FAZENDO UM PULL

PASSO 02 - O PULL TRARÁ ALTERAÇÕES DO REPOSITÓRIO REMOTO DO GITHUB E DIFERENTEMENTE DO "FETCH" AS MESCLARÁ AUTOMATICAMENTE NO CÓDIGO ATUAL



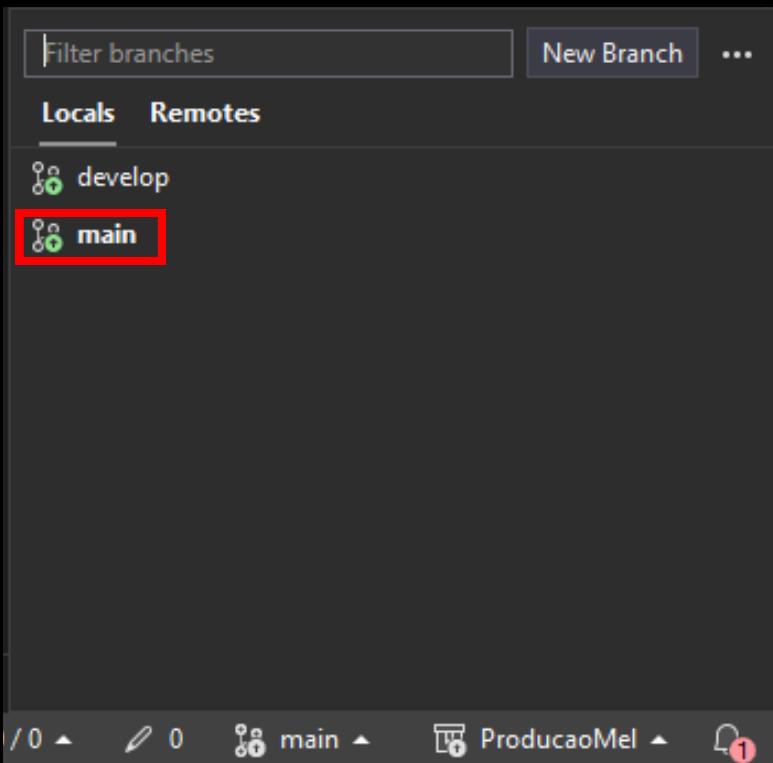
O COMANDO PULL É MUITO IMPORTANTE PARA SEMPRE MANTER SUA BRANCH ATUALIZADA, PRINCIPALMENTE SE ESTIVER TRABALHANDO EM EQUIPE!

COMO FAZER UM MERGE



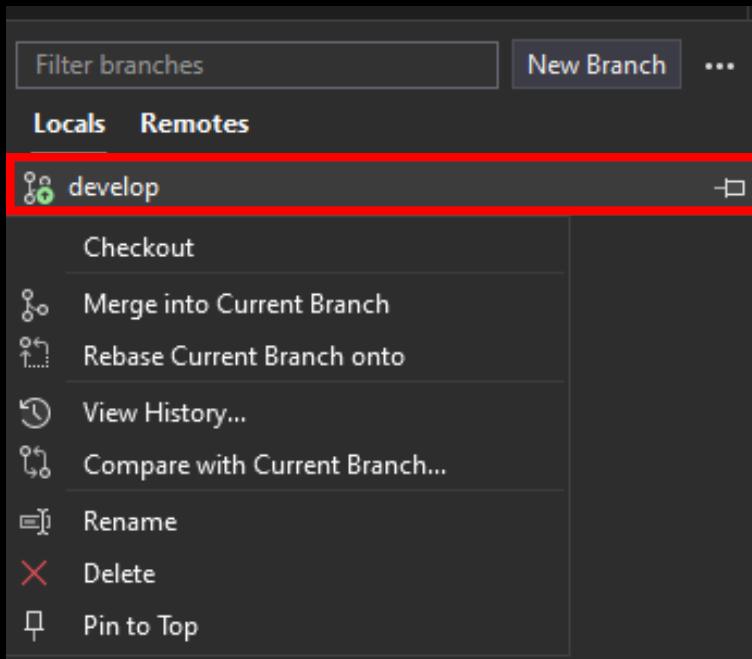
FAZENDO UM MERGE

PASSO 01 - NA JANELA DE SELEÇÃO DE BRANCHS CLIQUE NA BRANCH QUE RECEBERÁ AS ALTERAÇÕES DE OUTRA BRANCH



FAZENDO UM MERGE

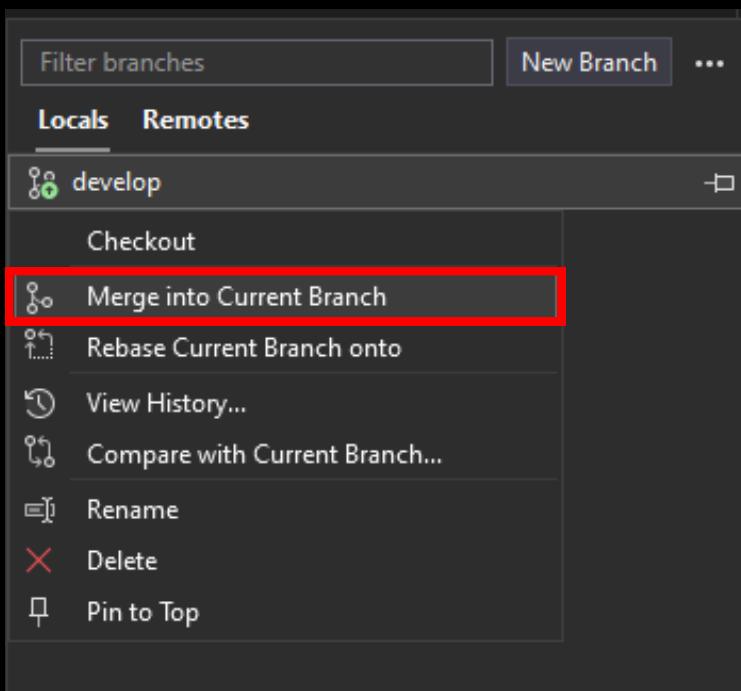
PASSO 02 - AGORA CLIQUE COM O BOTÃO DIREITO NA BRANCH QUE SERÁ MESCLADA NA BRANCH SELECIONADA ANTERIORMENTE



NO CASO A BRANCH “MAIN” RECEBERÁ AS ALTERAÇÕES DA BRANCH “DEVELOP”

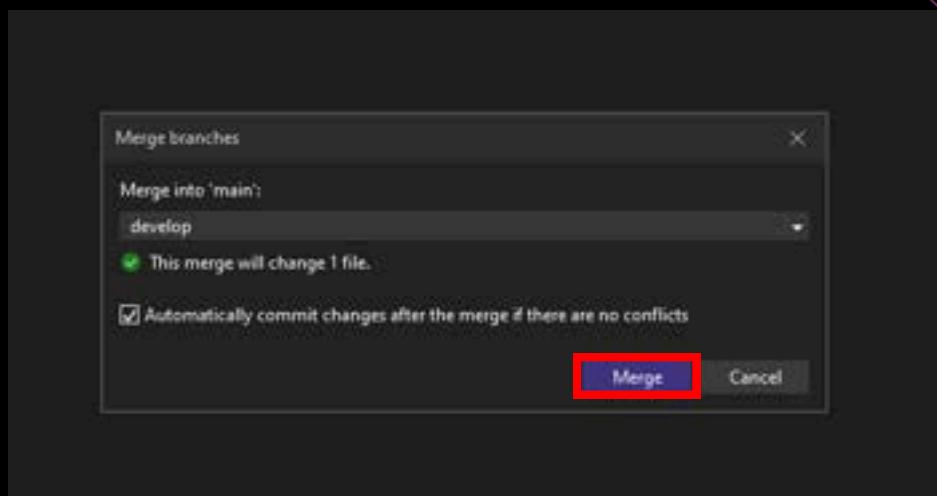
FAZENDO UM MERGE

PASSO 03 - CLIQUE NA OPÇÃO “MERGE INTO CURRENT BRANCH”



FAZENDO UM MERGE

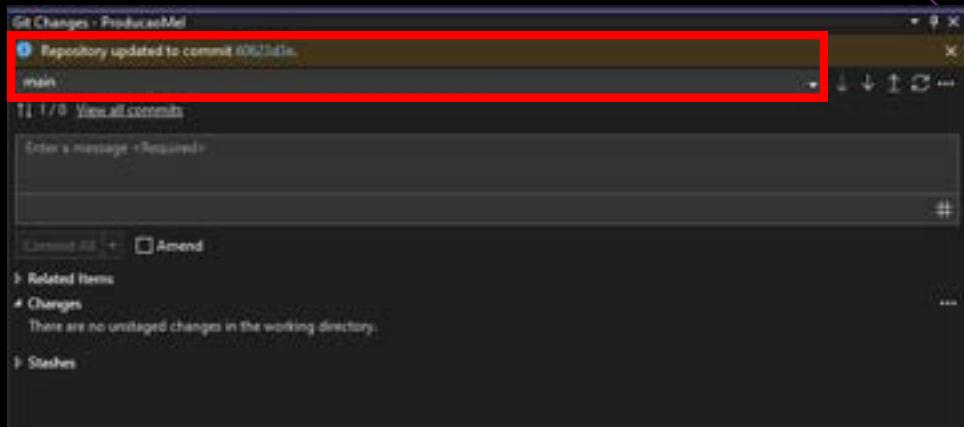
PASSO 04 - AQUI AS ALTERAÇÕES DA BRANCH “DEVELOP” SERÃO ENVIADAS E MESCLADAS COM O CÓDIGO DA BRANCH “MAIN” COMBINANDO AS DUAS



PASSO 05 - CLIQUE EM “MERGE”

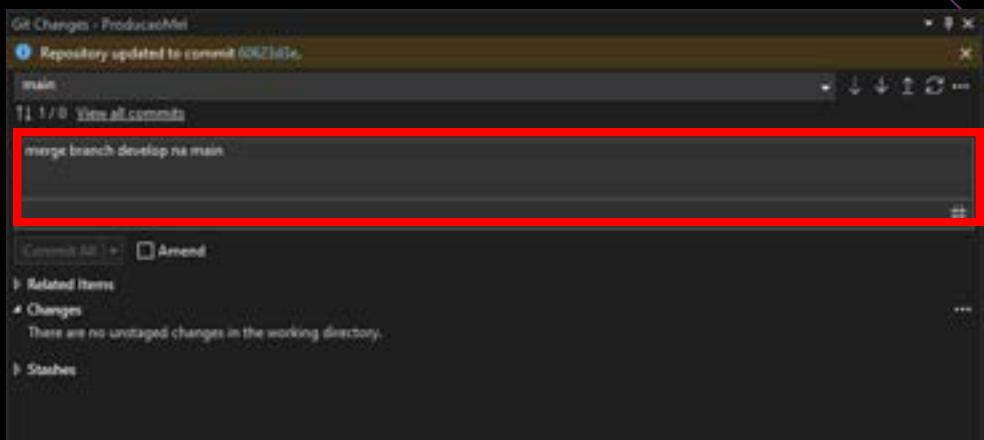
FAZENDO UM MERGE

PASSO 06 - AGORA A BRANCH "MAIN" ESTÁ ATUALIZADA COM AS MODIFICAÇÕES VINDAS DO MERGE COM A BRANCH "DEVELOP"



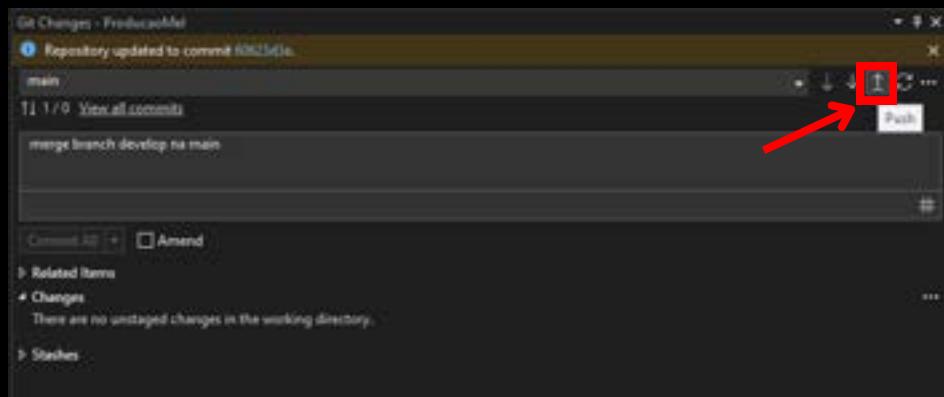
FAZENDO UM MERGE

PASSO 07 - NO CAMPO MENSAGEM, DIGITE UM
TEXTO OBJETIVO DESCREVENDO O MERGE
REALIZADO



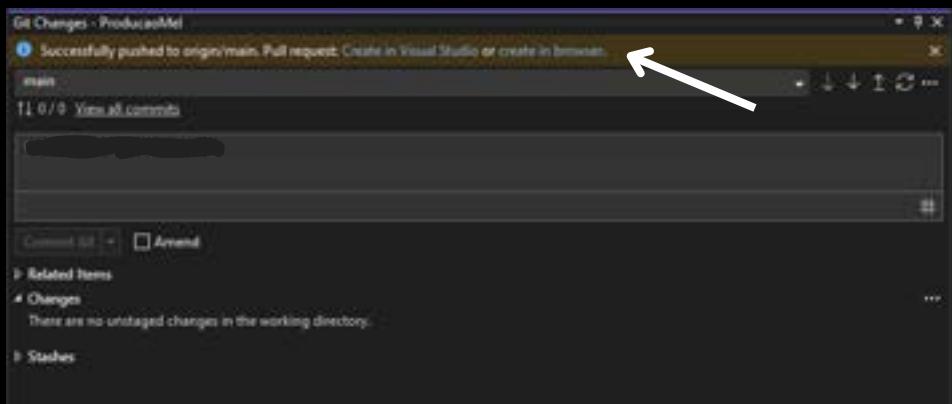
FAZENDO UM MERGE

PASSO 08 – CLIQUE NA OPÇÃO “PUSH” PARA ENVIAR AS ALTERAÇÕES DO MERGE PARA O REPOSITÓRIO REMOTO DO GITHUB



FAZENDO UM MERGE

PASSO 09 - PRONTINHO!
O MERGE ENTRE SUAS BRANCHS FOI REALIZADO
COM SUCESSO



CONCLUSÃO

Neste e-book, exploramos as principais funcionalidades e conceitos que envolvem o uso do Git e GitHub, especialmente integrados ao Visual Studio. Desde a criação de uma conta no GitHub e configuração do Git até o trabalho com commits, branches, merges e controle de alterações, abordamos os aspectos fundamentais para que você possa iniciar ou aprofundar sua jornada no controle de versões.

A prática desses comandos e operações é essencial para um entendimento sólido e para garantir a segurança e rastreabilidade no desenvolvimento de qualquer projeto. Com o domínio dessas ferramentas, você terá mais confiança para colaborar em projetos, trabalhar de forma organizada e manter um histórico claro das suas modificações.

Esperamos que este guia tenha esclarecido as etapas e boas práticas, facilitando seu caminho no aprendizado de Git e GitHub. Lembre-se de que o aprendizado contínuo e a prática são a chave para tirar o máximo proveito dessas tecnologias. Boa sorte em seus projetos e continue explorando as possibilidades que o Git e o GitHub oferecem!



**CRIADO POR ESTUDANTES DA UNIVERSIDADE
PAULISTA**

Amanda Aparecida Ferreira
Carine dos Reis Souza
Danielly Celestrini Burkowski
João Pedro Tamanini Mazzoni
Maria Luiza Cyrino Guedes Leandro
Rian Freire Garcia Ortega