



Curso de Extensão
Tecnologias Microsoft



INF-0991

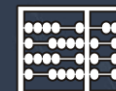
Controle de Versão

AULA 2

Prof. Dr. Rodrigo Bonacin

RBonacin@unicamp.br

03 de Setembro de 2022



INSTITUTO DE
COMPUTAÇÃO

Agenda – Aula 2

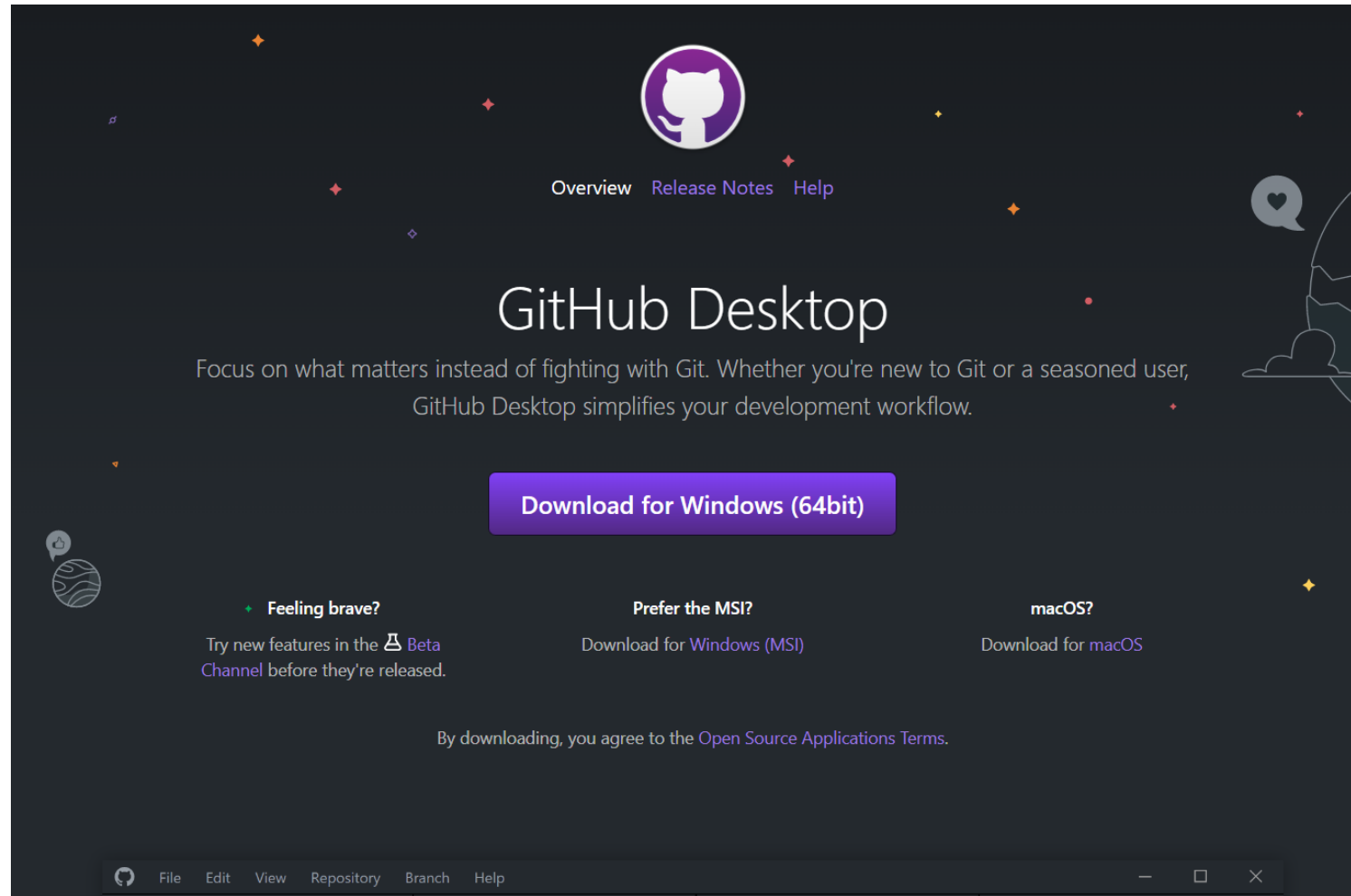


1. Correção Atividade 1 (outro arquivo de slides)
2. Ferramentas
 - GitHub Desktop
 - Visual Studio Code com Git/GitHub
3. Boas práticas
4. Atividades Práticas com Ferramentas em Sala de Aula
5. Atividade 2

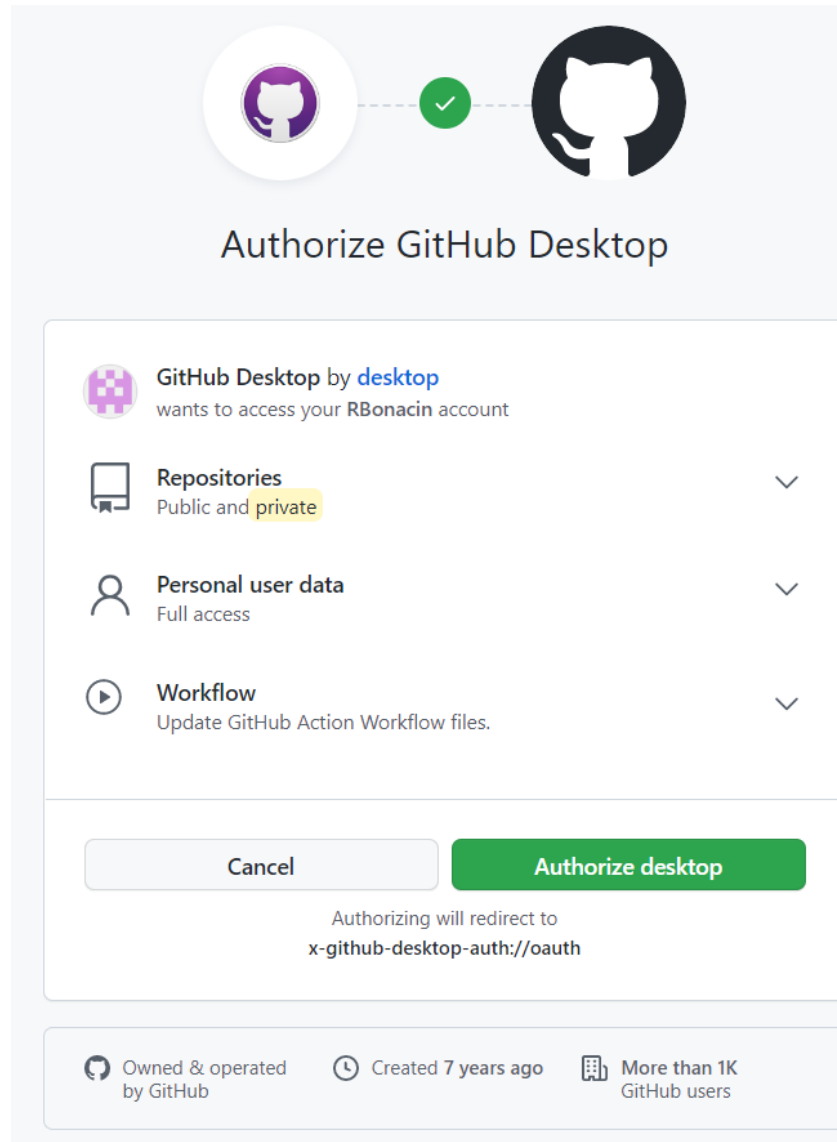
The background features a network of gray lines connecting various colored circles (orange, yellow, light blue, green) in a non-linear fashion. A dark blue horizontal band with a light blue border is positioned across the middle of the slide.

Ferramentas – GitHub Desktop

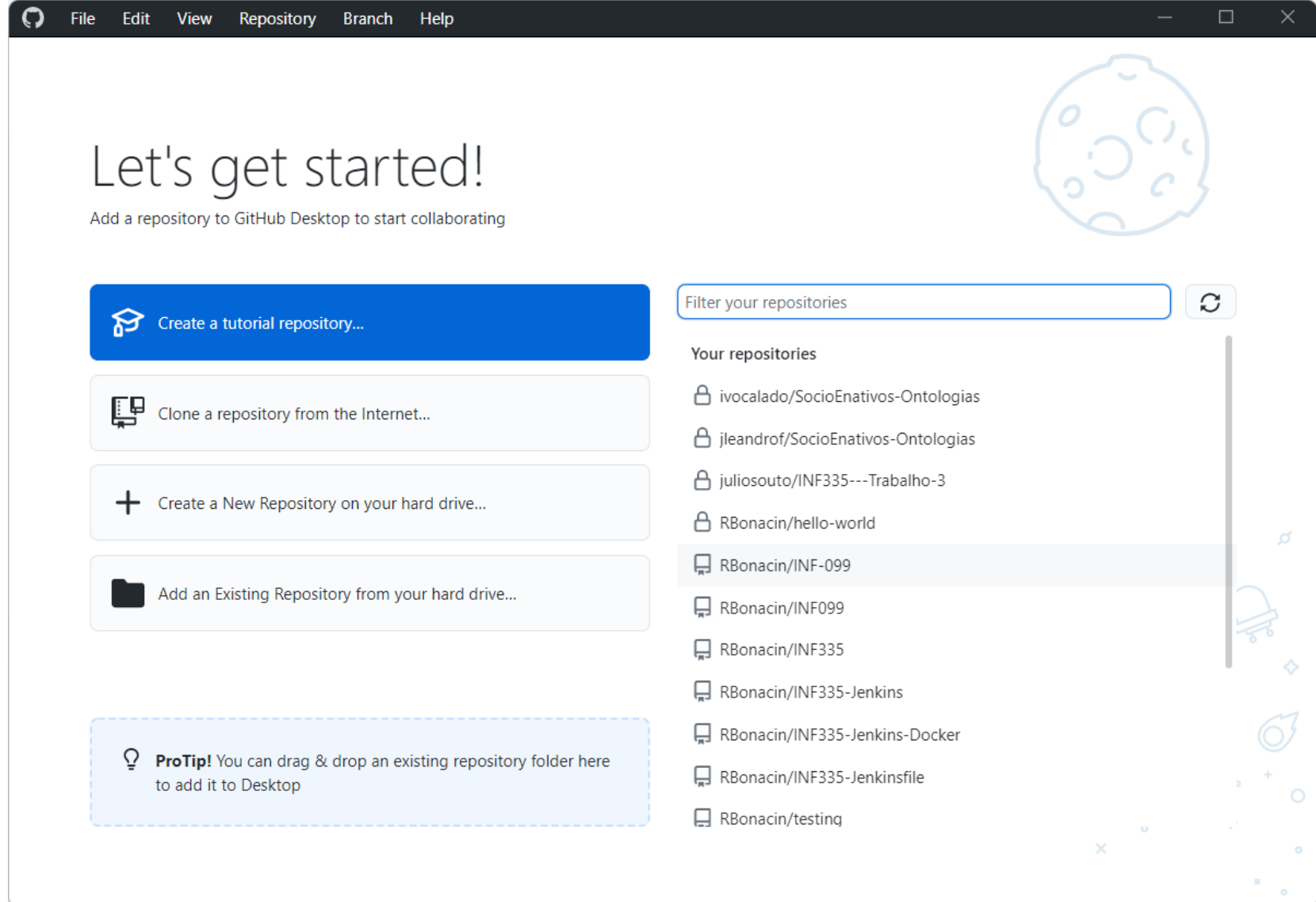
GitHub Desktop



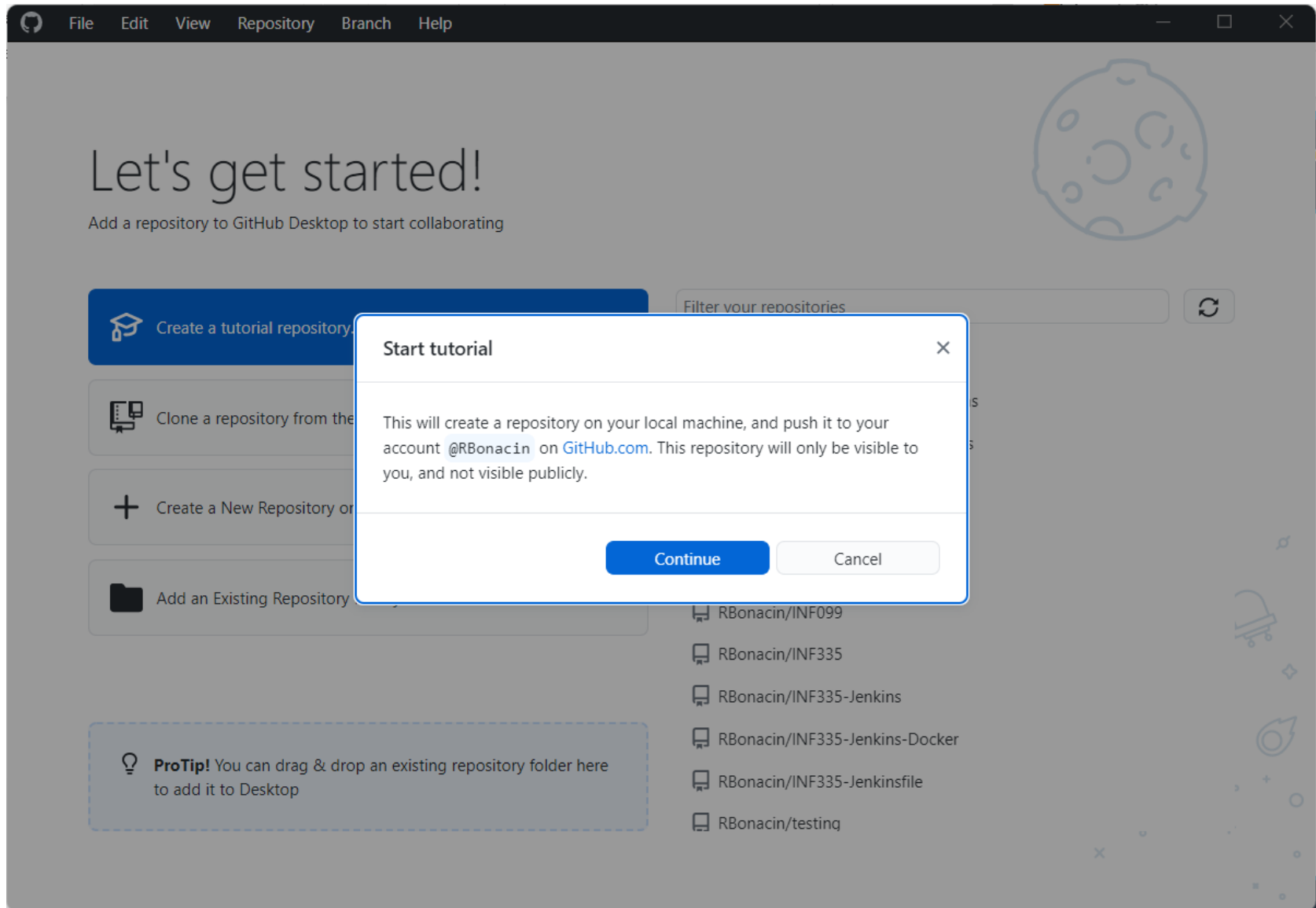
<https://desktop.github.com/>



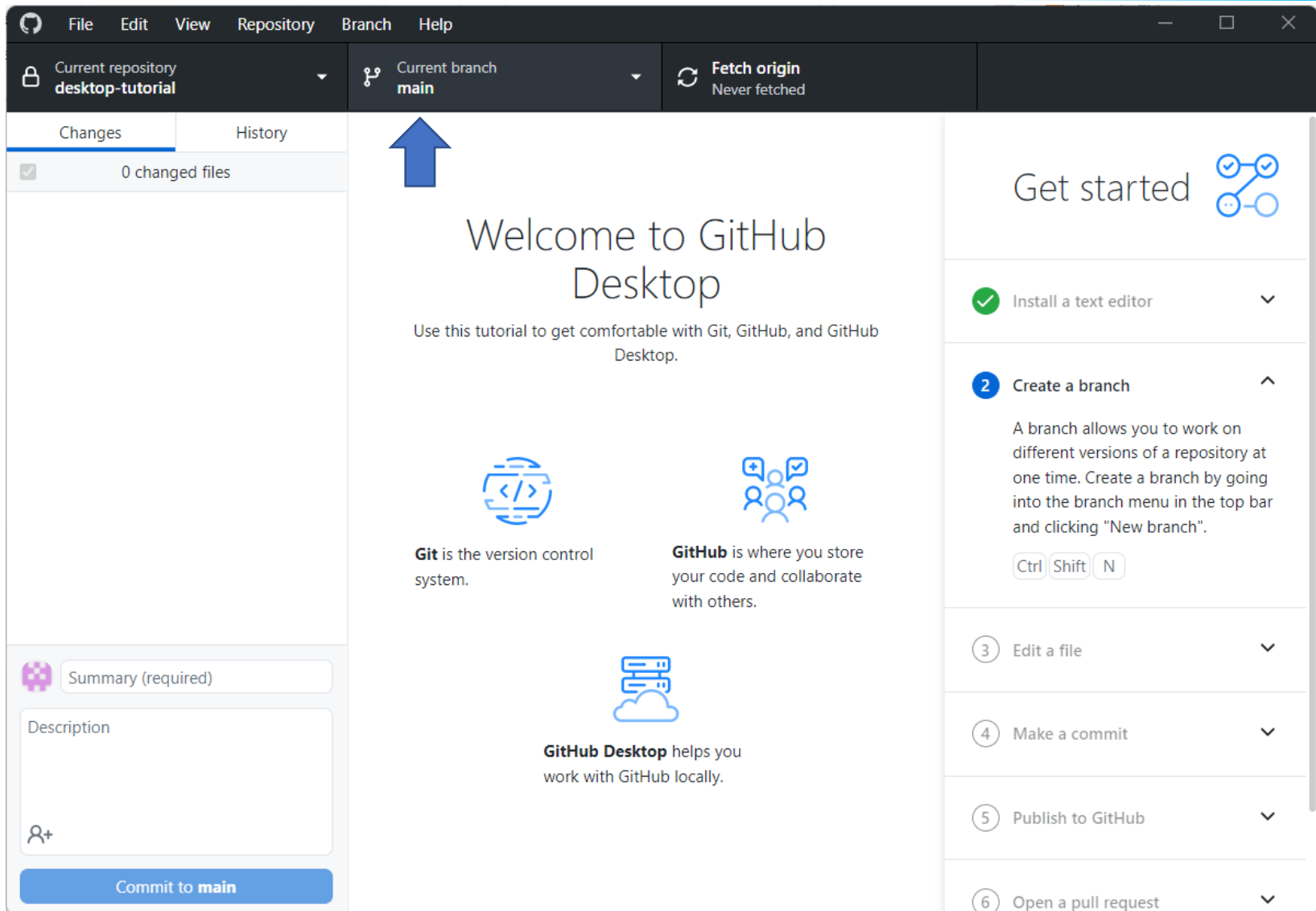
GitHub Desktop



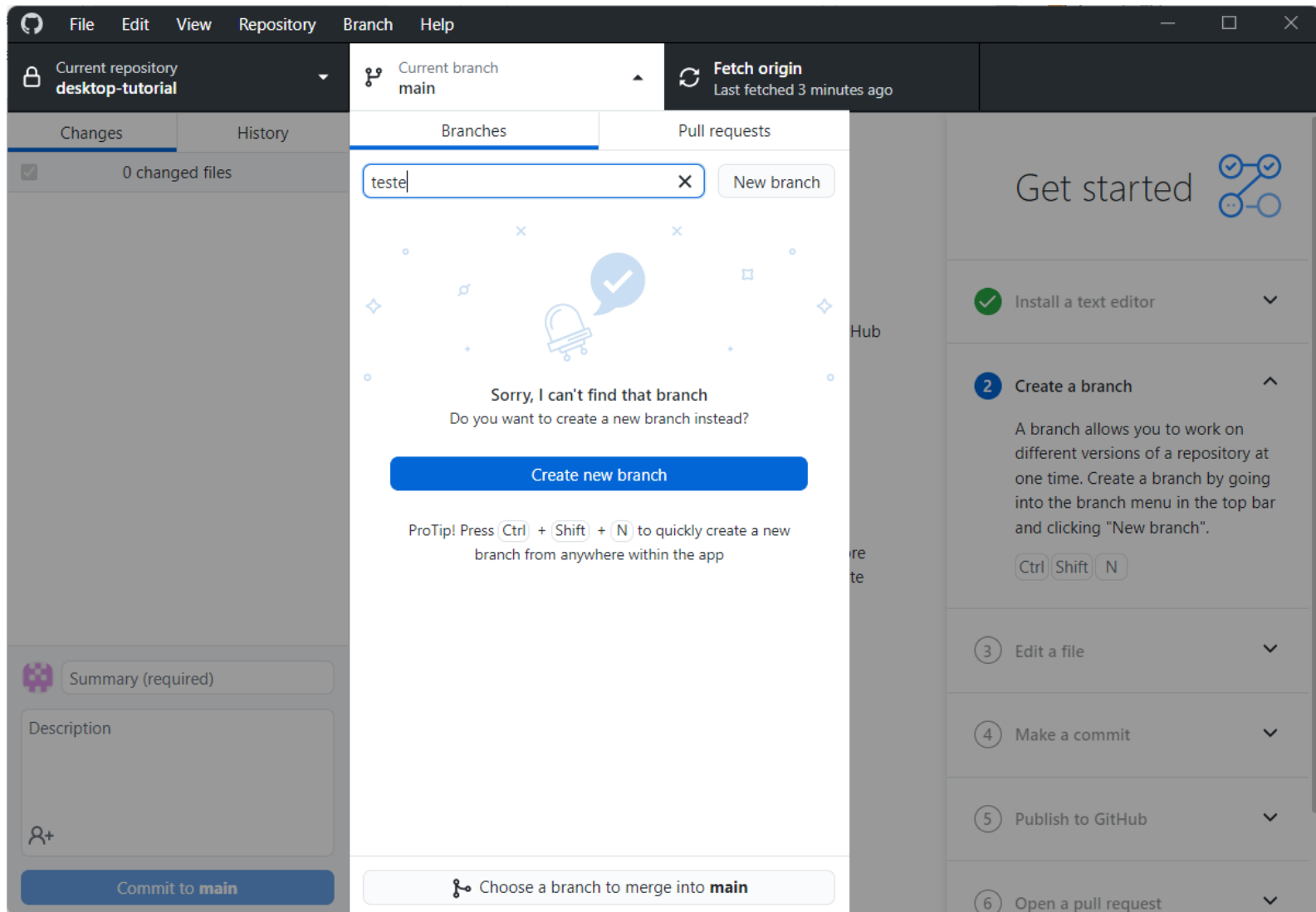
GitHub Desktop



GitHub Desktop



GitHub Desktop



GitHub Desktop



File Edit View Repository Branch Help

Current repository: **desktop-tutorial**

Current branch: **teste**

Publish branch
Publish this branch to GitHub

Changes History

0 changed files

Welcome to GitHub Desktop

Use this tutorial to get comfortable with Git, GitHub, and GitHub Desktop.

Git is the version control system.

GitHub is where you store your code and collaborate with others.

GitHub Desktop helps you work with GitHub locally.

Get started

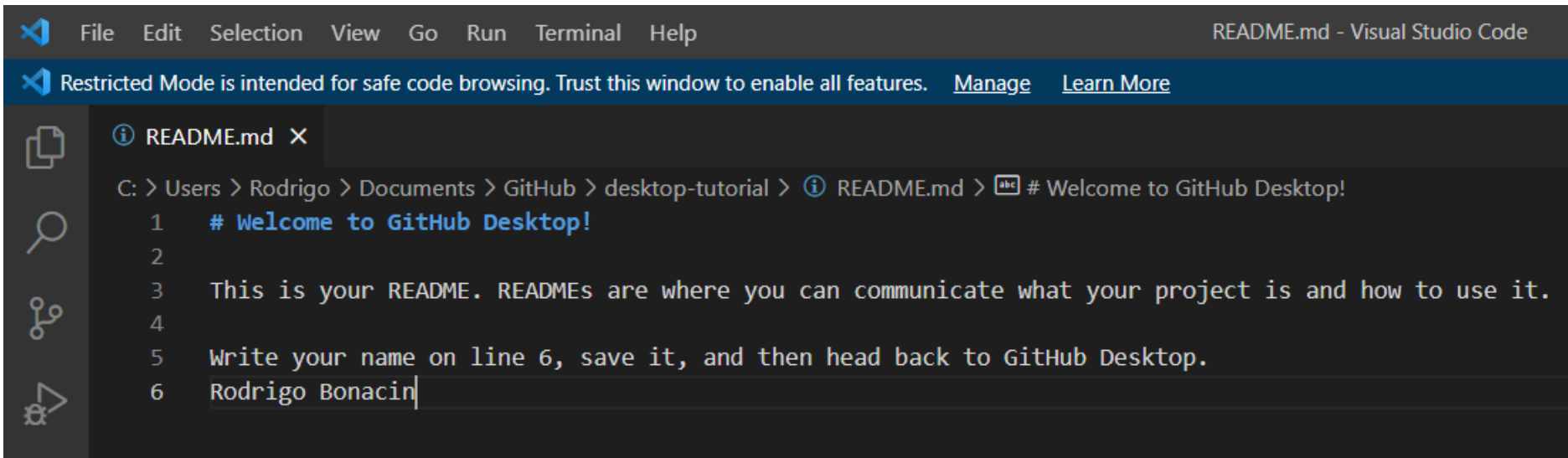
- ✓ Install a text editor
- ✓ Create a branch
- 3** Edit a file
Open this repository in your preferred text editor. Edit the `README.md` file, save it, and come back.
Open editor `Ctrl` `Shift` `A`
- 4 Make a commit
- 5 Publish to GitHub
- 6 Open a pull request

Summary (required)

Description

Commit to teste

GitHub Desktop

A screenshot of the Visual Studio Code editor interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The title bar shows 'README.md - Visual Studio Code'. A blue banner at the top states: 'Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More'. The left sidebar shows icons for Explorer, Search, Source Control, and Run and Debug. The Explorer view is active, showing a file named 'README.md'. The main editor area displays the content of the README file. The path bar shows 'C: > Users > Rodrigo > Documents > GitHub > desktop-tutorial > README.md'. The file content is as follows:

```
1  # Welcome to GitHub Desktop!
2
3  This is your README. READMEs are where you can communicate what your project is and how to use it.
4
5  Write your name on line 6, save it, and then head back to GitHub Desktop.
6  Rodrigo Bonacin
```

GitHub Desktop



The screenshot shows the GitHub Desktop application window. The top bar includes a menu (File, Edit, View, Repository, Branch, Help) and status information (Current repository: desktop-tutorial, Current branch: teste, Publish branch button). The main area is divided into three panes: 'Changes' (showing 1 changed file: README.md), 'History', and a file editor for README.md. The editor shows a diff with a new line added: '+Rodrigo Bonacin'. The bottom left pane contains the commit summary field with the text 'Inclusão do Nome' and a 'Commit to teste' button. On the right, a 'Get started' sidebar lists a sequence of steps: 1. Install a text editor, 2. Create a branch, 3. Edit a file, 4. Make a commit (circled in blue), 5. Publish to GitHub, and 6. Open a pull request. The 'Make a commit' step includes a description: 'A commit allows you to save sets of changes. In the "summary" field in the bottom left, write a short message that describes the changes you made. When you're done, click the blue Commit button to finish.'

GitHub Desktop



File Edit View Repository Branch Help

Current repository: **desktop-tutorial**

Current branch: **teste**

Publish branch
Publish this branch to GitHub

Changes History

0 changed files

Welcome to GitHub Desktop

Use this tutorial to get comfortable with Git, GitHub, and GitHub Desktop.

Git is the version control system.

GitHub is where you store your code and collaborate with others.

GitHub Desktop helps you work with GitHub locally.

Get started

- ✓ Install a text editor
- ✓ Create a branch
- ✓ Edit a file
- ✓ Make a commit
- 5 Publish to GitHub**
Publishing will "push", or upload, your commits to this branch of your repository on GitHub. Publish using the third button in the top bar.
- 6 Open a pull request

Summary (required)

Description

Commit to **teste**

Committed just now
Inclusão do Nome

Undo

GitHub Desktop



The screenshot shows the GitHub Desktop application window. The top bar includes a menu (File, Edit, View, Repository, Branch, Help) and status information: 'Current repository: desktop-tutorial', 'Current branch: teste', and 'Fetch origin' (Last fetched 1 minute ago). The left sidebar has tabs for 'Changes' (showing 0 changed files) and 'History'. The main area displays a 'Welcome to GitHub Desktop' message with instructions to use the tutorial. It features three icons: a code editor icon for 'Git is the version control system.', a group of people icon for 'GitHub is where you store your code and collaborate with others.', and a server icon for 'GitHub Desktop helps you work with GitHub locally.' The right sidebar shows a 'Get started' checklist with five items: 'Install a text editor', 'Create a branch', 'Edit a file', 'Make a commit', and 'Publish to GitHub', all marked with green checkmarks. The sixth item, '6 Open a pull request', is circled in blue and includes a 'Skip' link. Below it is a description of pull requests and buttons for 'Open pull request' and 'Exit tutorial'. At the bottom left, there is a 'Commit to teste' button and a 'Summary (required)' field.

File Edit View Repository Branch Help

Current repository: desktop-tutorial

Current branch: teste

Fetch origin
Last fetched 1 minute ago

Changes History

0 changed files

Welcome to GitHub Desktop

Use this tutorial to get comfortable with Git, GitHub, and GitHub Desktop.

Git is the version control system.

GitHub is where you store your code and collaborate with others.

GitHub Desktop helps you work with GitHub locally.

Get started

- ✓ Install a text editor
- ✓ Create a branch
- ✓ Edit a file
- ✓ Make a commit
- ✓ Publish to GitHub
- 6 Open a pull request** [Skip](#)

A pull request allows you to propose changes to the code. By opening one, you're requesting that someone review and merge them. Since this is a demo repository, this pull request will be private.

[Open pull request](#) [Ctrl + R](#)

[Exit tutorial](#)


Summary (required)

Description

[Commit to teste](#)

GitHub Desktop




 Search or jump to... / [Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)

[RBonacin / desktop-tutorial](#) Private Unwatch 1 Fork 0 Star 0

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#) [Settings](#)

Open a pull request


Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

 base: main

←































compare: teste

✓ Able to merge. These branches can be automatically merged.

 Inclusão do Nome

Write

Preview

H B I                              

GitHub Desktop



Inclusão do Nome #1

Open RBonacin wants to merge 1 commit into `main` from `teste`

Conversation 0 Commits 1 Checks 0 Files changed 1

RBonacin commented 2 minutes ago

No description provided.

Inclusão do Nome 3f3f5fb

Add more commits by pushing to the `teste` branch on RBonacin/desktop-tutorial.

- Require approval from specific reviewers before merging**
Ensure specific people or teams approve pull requests before they're merged into your main branch.
- Continuous integration has not been set up**
GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.
- This branch has no conflicts with the base branch**
Merging can be performed automatically.

Merge pull request You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Write

Preview

H B I

Leave a comment

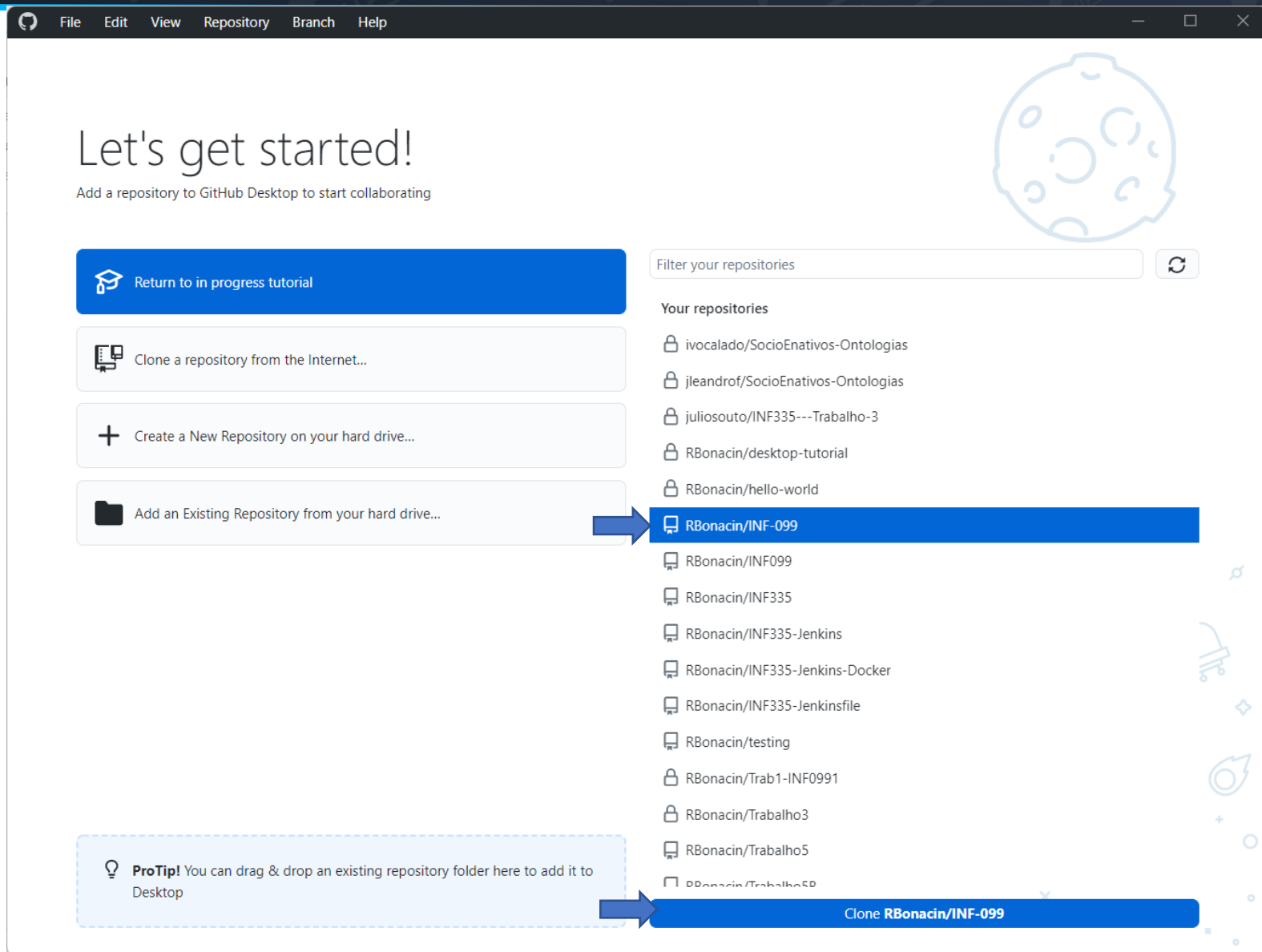
Attach files by dragging & dropping, selecting or pasting them.

Close pull request Comment

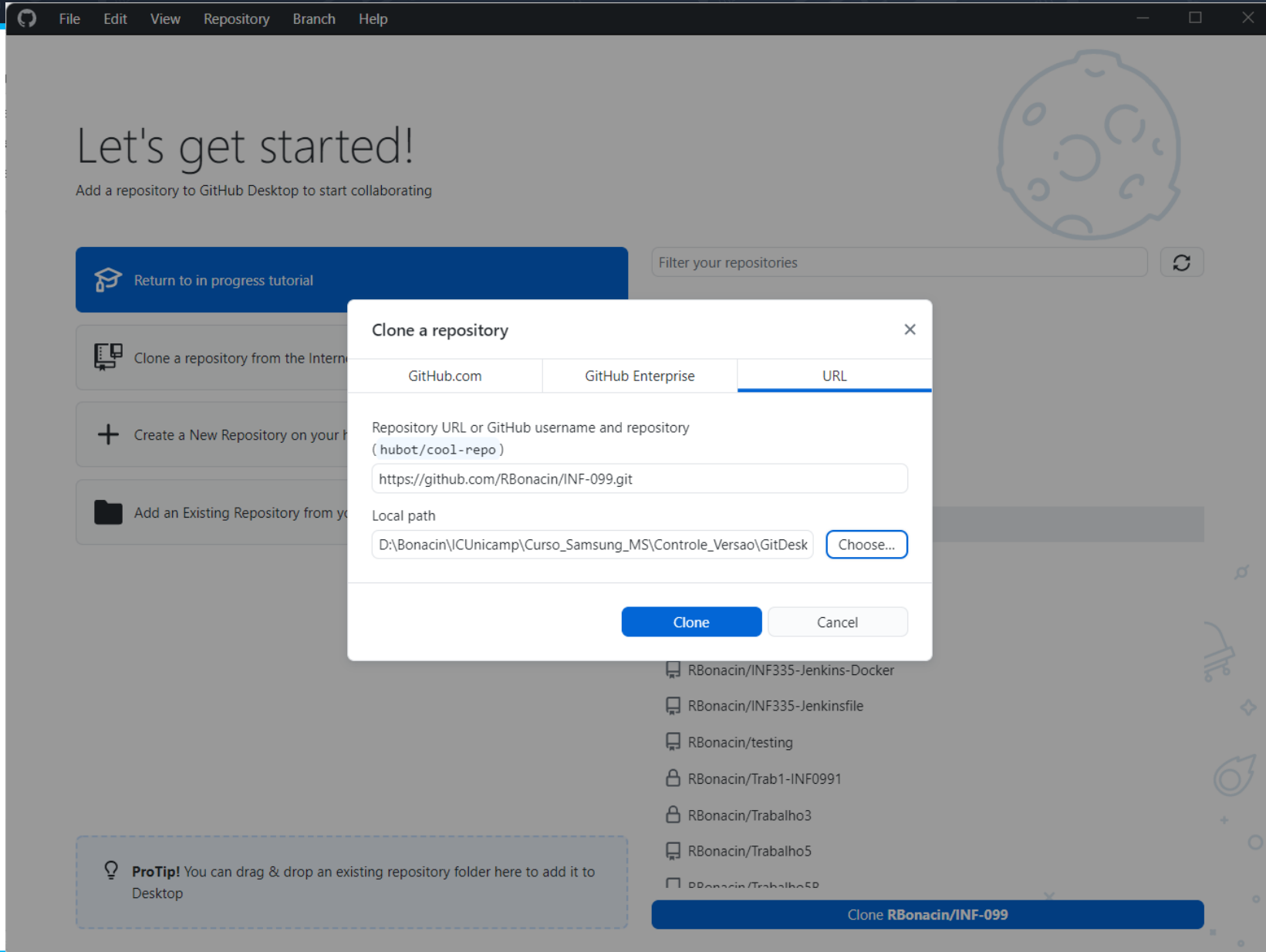
Inclusão do Nome #1

Merged RBonacin merged 1 commit into `main` from `teste` now

GitHub Desktop



GitHub Desktop



GitHub Desktop



FileEditViewRepositoryBranchHelp

Current repository
INF-099

Current branch
main

Fetch origin
Last fetched 1 minute ago

ChangesHistory

0 changed files

No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Open the repository in your external editor

Select your editor in [Options](#)

Repository menu or `Ctrl` `Shift` `A`

Open in Visual Studio Code

View the files of your repository in Explorer

Repository menu or `Ctrl` `Shift` `F`

Show in Explorer

Open the repository page on GitHub in your browser

Repository menu or `Ctrl` `Shift` `G`

View on GitHub

Summary (required)

Description

+

Commit to main



D:\Bonacin\ICUnicamp\Curso_Samsung_MS\Controle_Versao\GitDesktop\INF-099\README.md - Notepad++

Arquivo Editar Localizar Visualizar Formatar Linguagem Configurações Ferramentas Macro Executar Plugins Janela ?



README.md

```
1 INF-099 - Controle de Versão
2 Curso Extensão Tecnologias Microsoft - IC - UNICAMP
3 Este repositório tem fins didáticos para a referida disciplina.
4 Professor Rodrigo Bonacin
5 Setembro de 2022.
```

GitHub Desktop



FileEditViewRepositoryBranchHelp

Current repository
INF-099

Current branch
main

Fetch origin
Last fetched 10 minutes ago

Changes 1History

1 changed file

README.md

README.md

@@ -2,4 +2,4 @@ INF-099 - Controle de Versão

2 2 Curso Extensão Tecnologias Microsoft - IC - UNICAMP

3 3 Este repositório tem fins didáticos para a referida disciplina.

4 4 Professor Rodrigo Bonacin

5 -Agosto de 2022.

5 +Setembro de 2022.

Atualização da data

Description

Commit to main

GitHub Desktop



FileEditViewRepositoryBranchHelp

Current repository
INF-099

Current branch
main

Push origin
Last fetched 12 minutes ago

1 ↑

ChangesHistory

No branches to compare

Atualização da data

Rodrigo Bonacin • 1 minute ago

Boa Noite

Rodrigo Bonacin • 7 days ago

Merge README.md

Rodrigo Bonacin • 7 days ago

Inclusão da data

Rodrigo Bonacin • 7 days ago

Autor

Rodrigo Bonacin • 7 days ago

Inclusao IC - UNICAMP

Rodrigo Bonacin • Aug 14, 2022

Incluir Boa tarde

Rodrigo Bonacin • Aug 14, 2022

Inclusao do nome do curso

Rodrigo Bonacin • Aug 14, 2022

Comentário adicionado

Rodrigo Bonacin • Aug 13, 2022

Create OlaMundo.cs

Rodrigo Bonacin • Aug 13, 2022

Create README.md

Rodrigo Bonacin • Aug 13, 2022

Atualização da data

Rodrigo Bonacin ec450fb ± 1 changed file +1 -1

README.md

@@ -2,4 +2,4 @@ INF-099 - Controle de Versão

2 2 Curso Extensão Tecnologias Microsoft - IC - UNICAMP

3 3 Este repositório tem fins didáticos para a referida disciplin

4 4 Professor Rodrigo Bonacin

5 -Agosto de 2022.

5 +Setembro de 2022.

The background features a network of gray lines connecting various colored circles (orange, yellow, green, blue) in a non-linear fashion. A dark blue horizontal band with a light blue border is positioned across the middle of the slide.

Ferramentas – VS Code + GitHub



<https://code.visualstudio.com/docs/editor/github>

Visual Studio Code Docs Updates Blog API Extensions FAQ Learn

Search Docs

Overview

SETUP

GET STARTED

USER GUIDE

Basic Editing

Extension Marketplace

IntelliSense

Code Navigation

Refactoring

Debugging

Version Control

Working with GitHub

Working with GitHub in VS Code

Edit

GitHub is a cloud-based service for storing and sharing source code. Using GitHub with Visual Studio Code lets you share your source code and collaborate with others right within your editor. There are many ways to interact with GitHub, for example, via their website at <https://github.com> or the [Git](#) command-line interface (CLI), but in VS Code, the rich GitHub integration is provided by the [GitHub Pull Requests and Issues](#) extension.

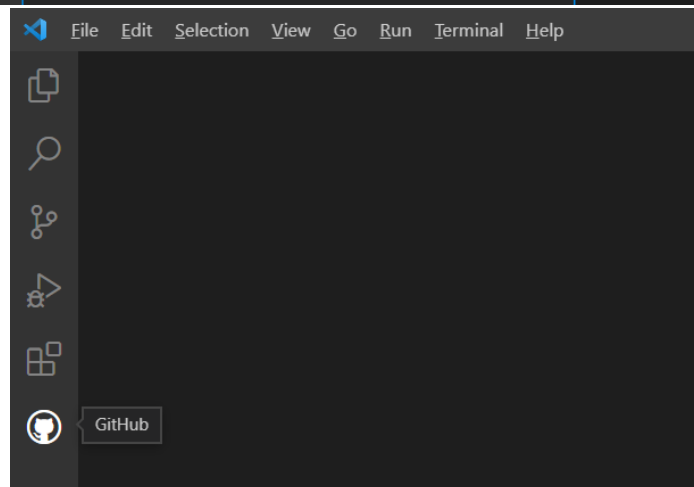
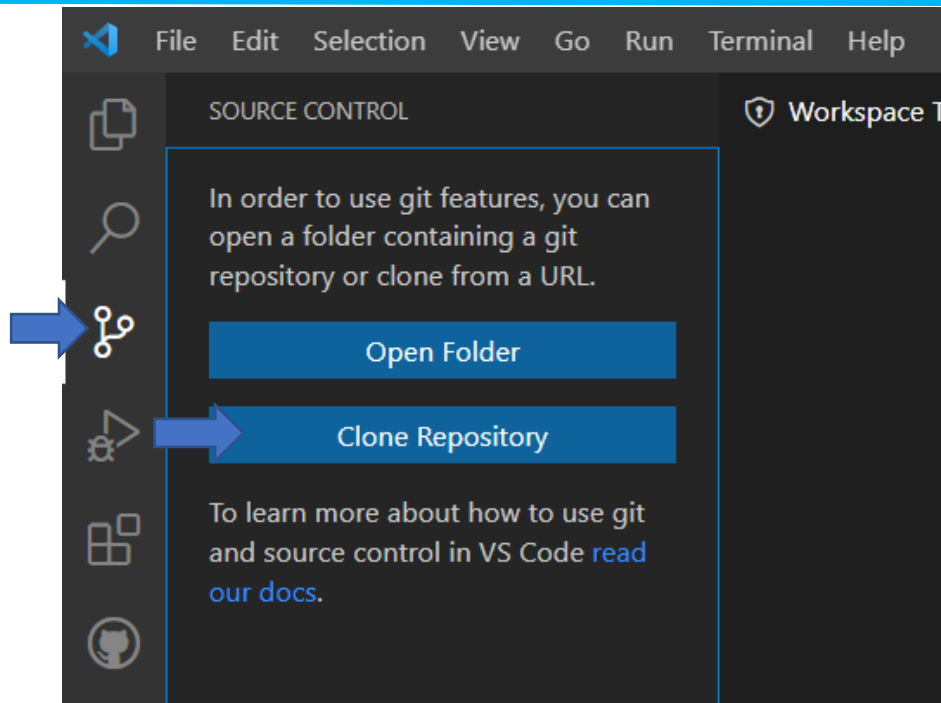
Install the GitHub Pull Requests and Issues extension



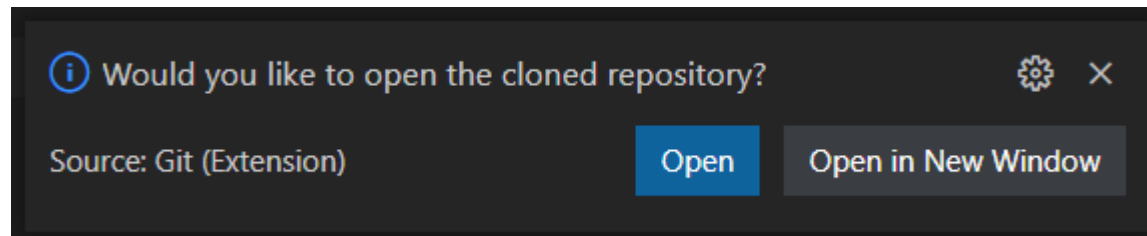
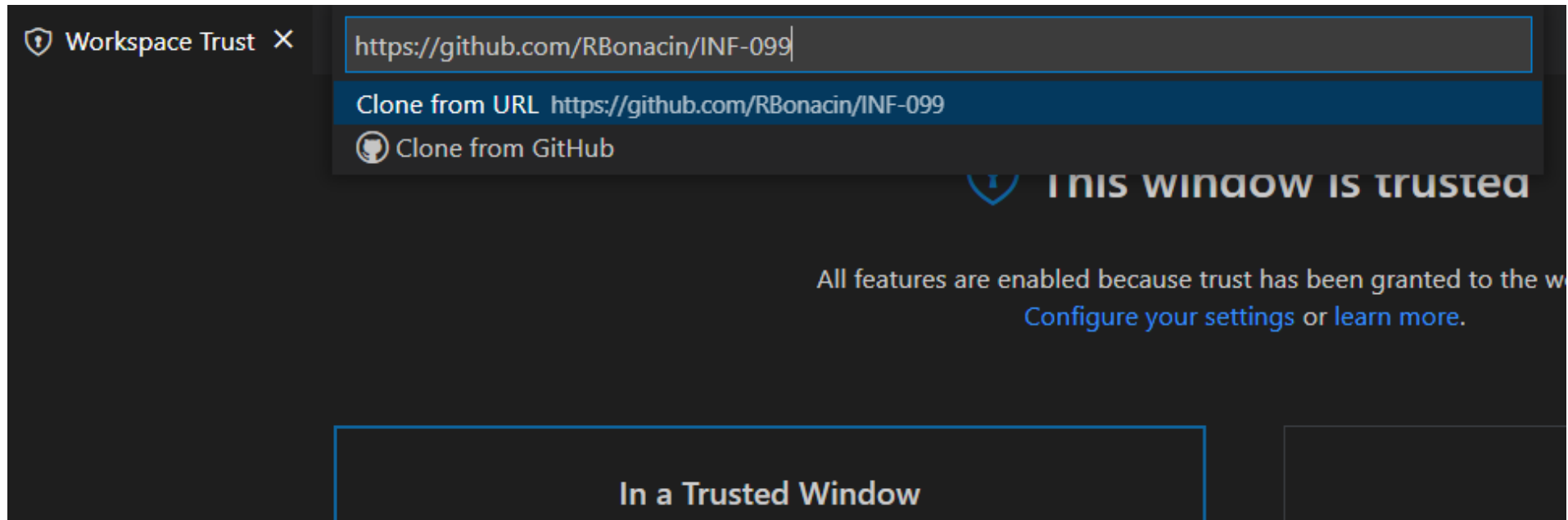
To get started with the GitHub in VS Code, you'll need to install [Git](#), [create a GitHub account](#) and install the [GitHub Pull Requests and Issues](#) extension. In this topic, we'll demonstrate how you can use some of your favorite parts of GitHub without leaving VS Code.

If you're new to source control or want to learn more about VS Code's basic Git support, you can start with the [Version Control](#) topic.

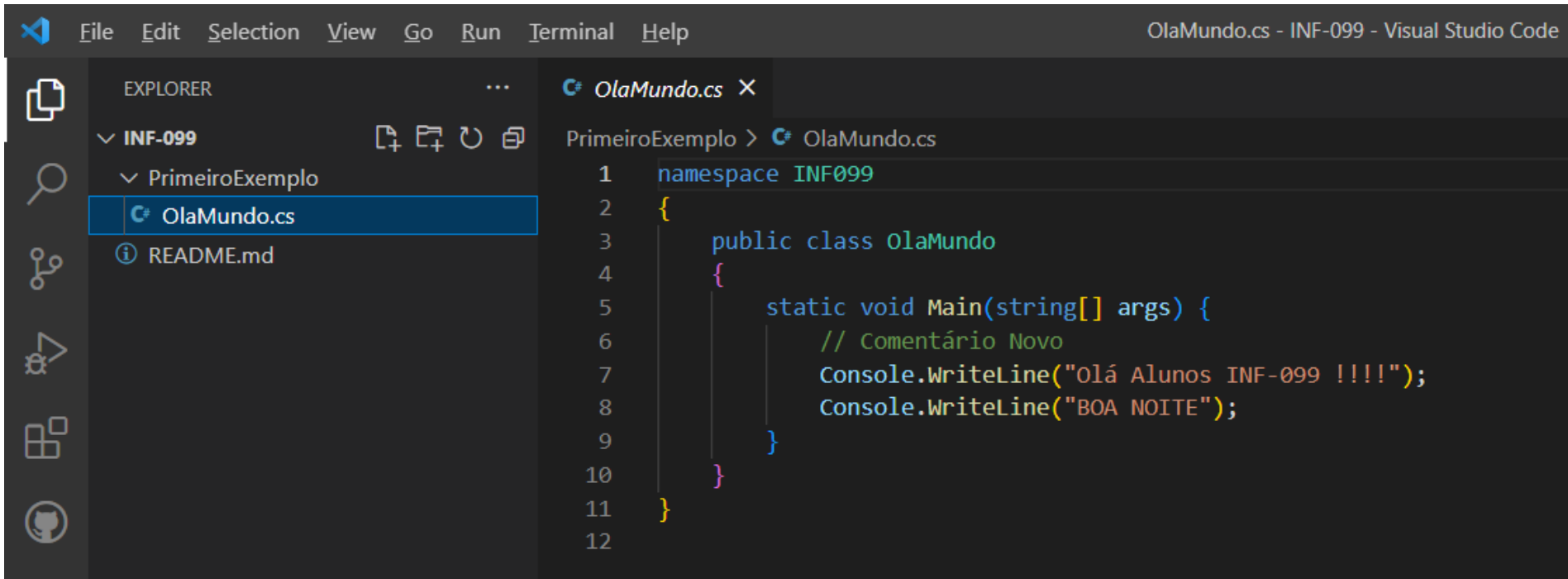
VS Code + GitHub



VS Code + GitHub



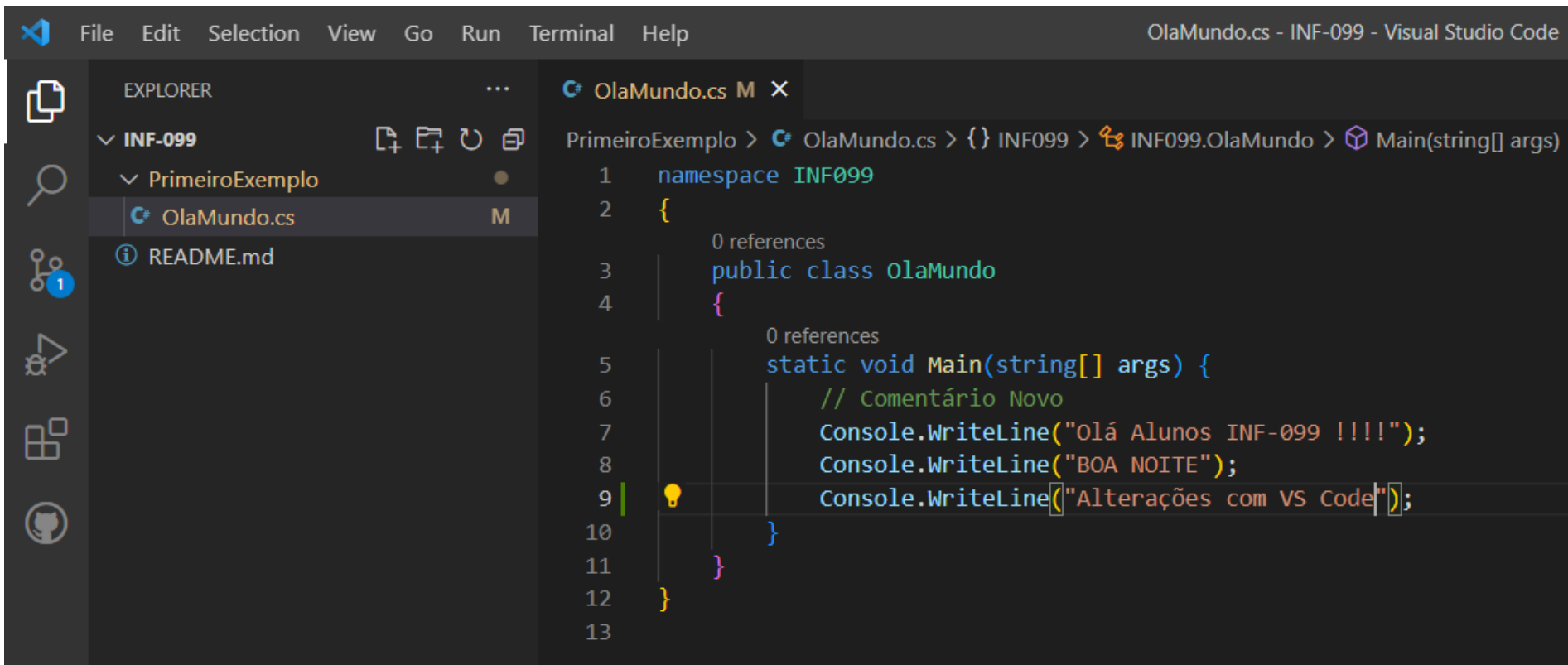
VS Code + GitHub



The screenshot shows the Visual Studio Code editor interface. The Explorer sidebar on the left displays a project structure with a folder named 'INF-099' containing a subfolder 'PrimeiroExemplo' and a file 'OlaMundo.cs'. The file 'OlaMundo.cs' is selected and open in the editor. The editor window title is 'OlaMundo.cs - INF-099 - Visual Studio Code'. The code in the editor is as follows:

```
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9         }
10    }
11 }
12
```

VS Code + GitHub



The screenshot displays the Visual Studio Code editor interface. The Explorer sidebar on the left shows a project structure with a folder named 'INF-099' containing a subfolder 'PrimeiroExemplo' and a file 'OlaMundo.cs'. The main editor area shows the code for 'OlaMundo.cs'. The code is written in C# and defines a namespace 'INF099' containing a public class 'OlaMundo'. The class has a static method 'Main' that takes an array of strings as an argument. Inside the 'Main' method, there are three lines of code that write to the console: 'Olá Alunos INF-099 !!!!', 'BOA NOITE', and 'Alterações com VS Code'. The code is as follows:

```
1 namespace INF099
2 {
3     0 references
4     public class OlaMundo
5     {
6         0 references
7         static void Main(string[] args) {
8             // Comentário Novo
9             Console.WriteLine("Olá Alunos INF-099 !!!!");
10            Console.WriteLine("BOA NOITE");
11            Console.WriteLine("Alterações com VS Code");
12        }
13    }
```

VS Code + GitHub



The screenshot shows the Visual Studio Code interface with the file `OlaMundo.cs` open. The editor displays the following C# code:

```
PrimeiroExemplo > OlaMundo.cs > {} INF099 > INF099.OlaMundo > Main(string[] args)
1 namespace INF099
2 {
3     0 references
4     public class OlaMundo
5     {
6         0 references
7         static void Main(string[] args) {
8             // Comentário Novo
9             Console.WriteLine("Olá Alunos INF-099 !!!!");
10            Console.WriteLine("BOA NOITE");
11            Console.WriteLine("Alterações com VS Code");
12        }
13 }
```

The Source Control view on the left shows a commit message "Alteração com VS Code" and a "Commit" button. A dialog box titled "Visual Studio Code" is displayed in the foreground, containing the following text:

Visual Studio Code

⚠ There are no staged changes to commit.

Would you like to stage all your changes and commit them directly?

Buttons: Yes, Always, Never, Cancel

VS Code + GitHub



The screenshot shows the Visual Studio Code interface with the file `OlaMundo.cs` open. The editor displays the following C# code:

```
PrimeiroExemplo > C# OlaMundo.cs > {} INF099 > INF099.OlaMundo > Main(string[] args)
1 namespace INF099
2 {
3     0 references
4     public class OlaMundo
5     {
6         0 references
7         static void Main(string[] args) {
8             // Comentário Novo
9             Console.WriteLine("Olá Alunos INF-099 !!!!");
10            Console.WriteLine("BOA NOITE");
11            Console.WriteLine("Alterações com VS Code");
```

A dialog box titled "Visual Studio Code" is displayed in the foreground, warning that the action will pull and push commits from and to 'origin/main'. The dialog includes a yellow warning icon and three buttons: "OK", "OK, Don't Show Again", and "Cancel".

VS Code + GitHub



main ▾

INF-099 / PrimeiroExemplo / OlaMundo.cs / <> Jump to ▾



RBonacin Alteração com VS Code

1 contributor

12 lines (12 sloc) | 287 Bytes

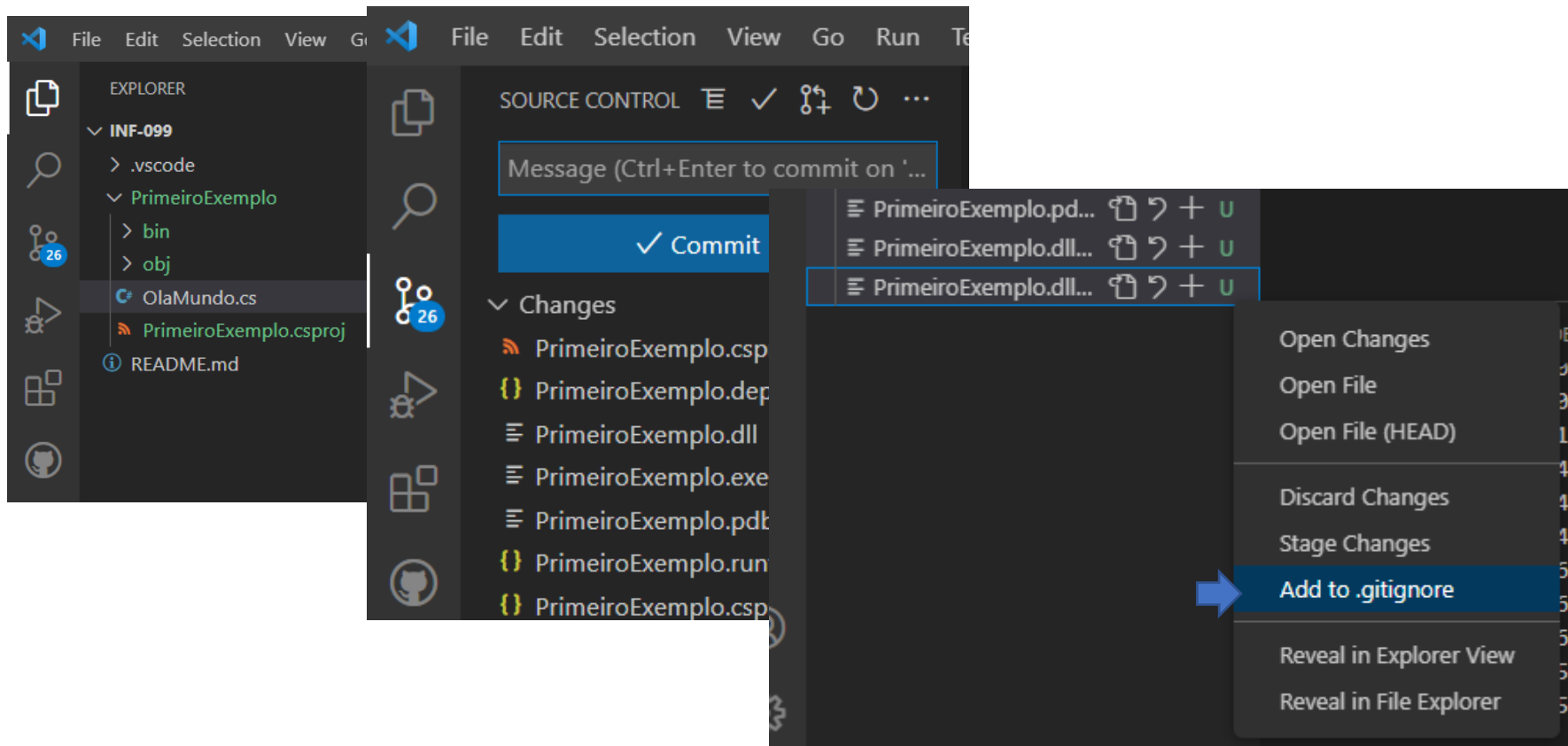
```
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10        }
11    }
12 }
```

VS Code + GitHub

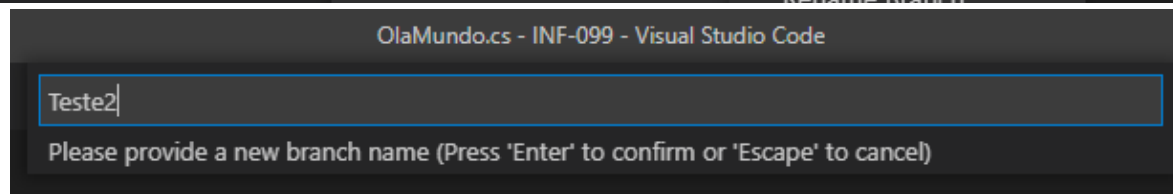
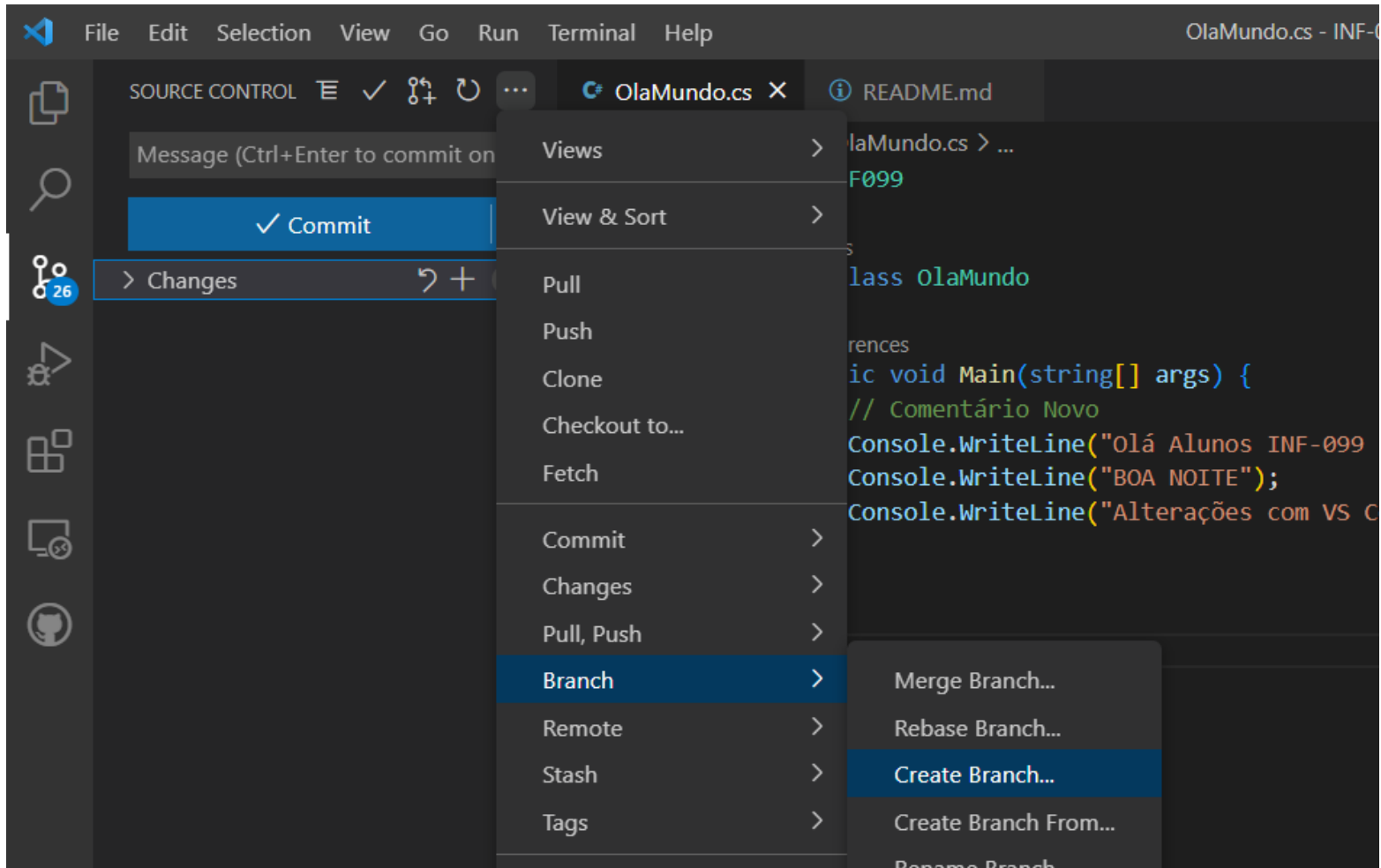


```
\INF-099\PrimeiroExemplo> dotnet new console --framework net6.0
```

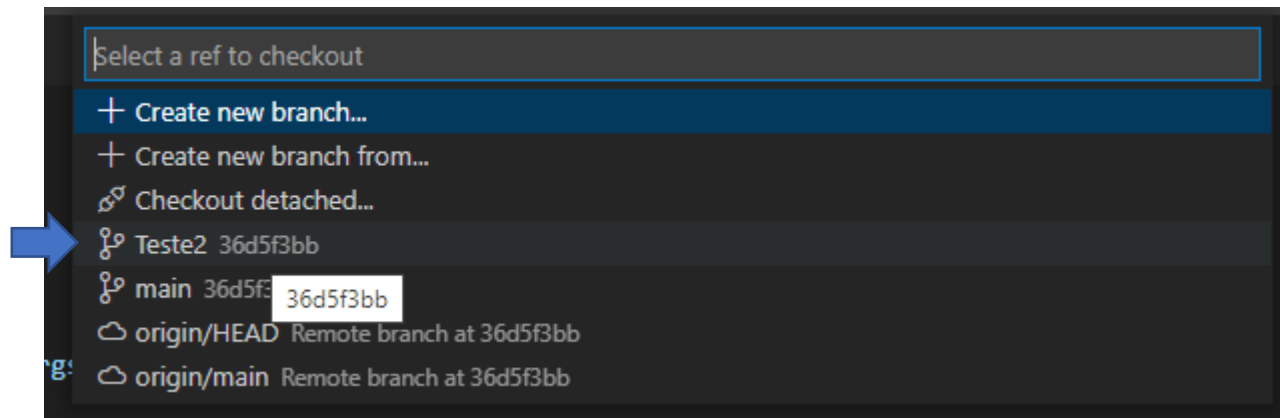
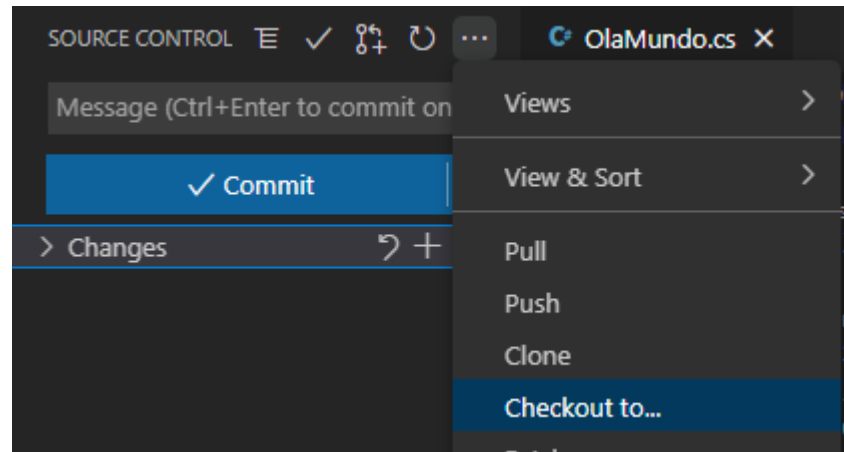
```
PS D:\Bonacin\ICUnicamp\Curso_Samsung_MS\Controle_Versao\VSCode\INF-099\PrimeiroExemplo> dotnet run
Olá Alunos INF-099 !!!!
BOA NOITE
Alterações com VS Code
```



VS Code + GitHub



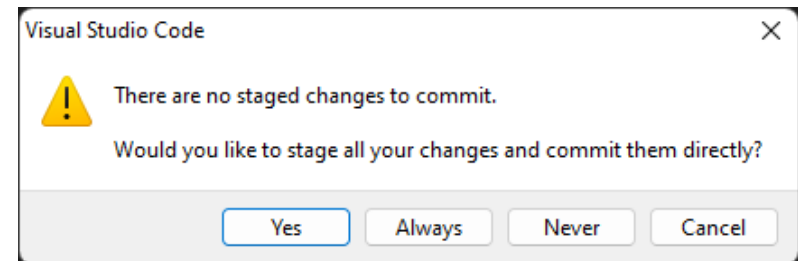
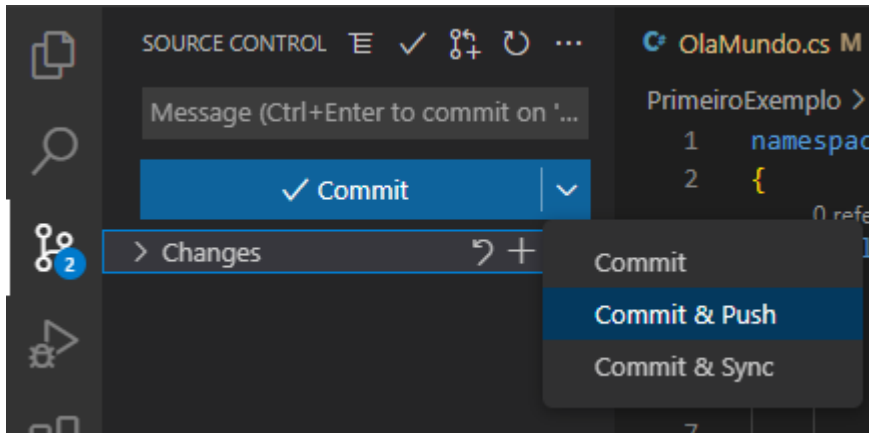
VS Code + GitHub



VS Code + GitHub



```
OlaMundo.cs M X
PrimeiroExemplo > OlaMundo.cs > {} INF099 > INF099.OlaMundo > Main(string[] args)
1 namespace INF099
2 {
3     0 references
4     public class OlaMundo
5     {
6         0 references
7         static void Main(string[] args) {
8             // Comentário Novo
9             Console.WriteLine("Olá Alunos INF-099 !!!!");
10            Console.WriteLine("BOA NOITE");
11            Console.WriteLine("Alterações com VS Code");
12            // Mais um comentário
13        }
14    }
15 }
```



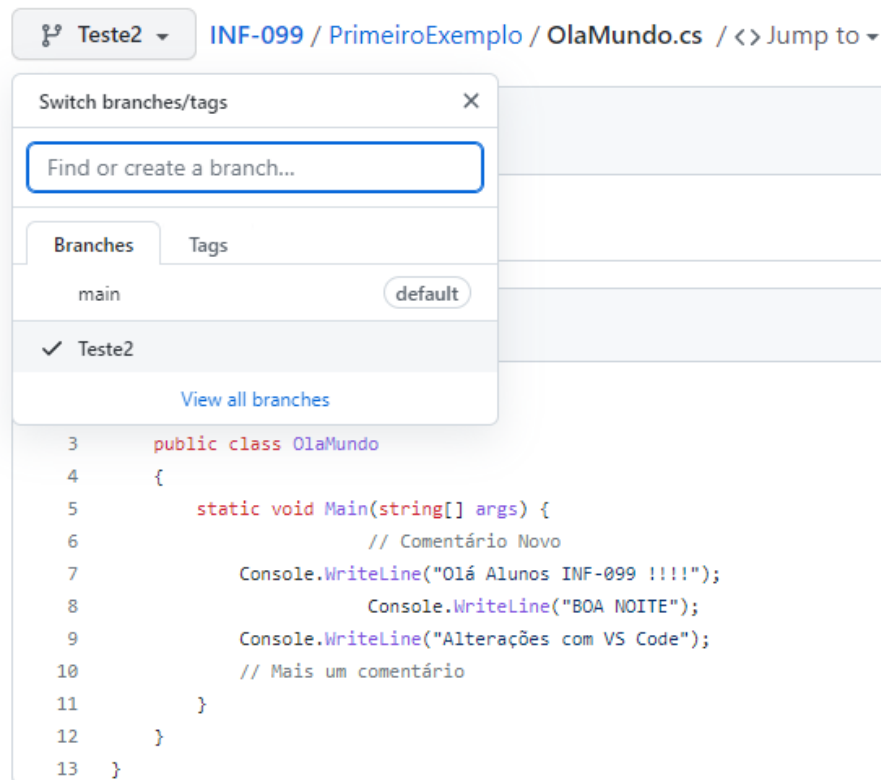
VS Code + GitHub



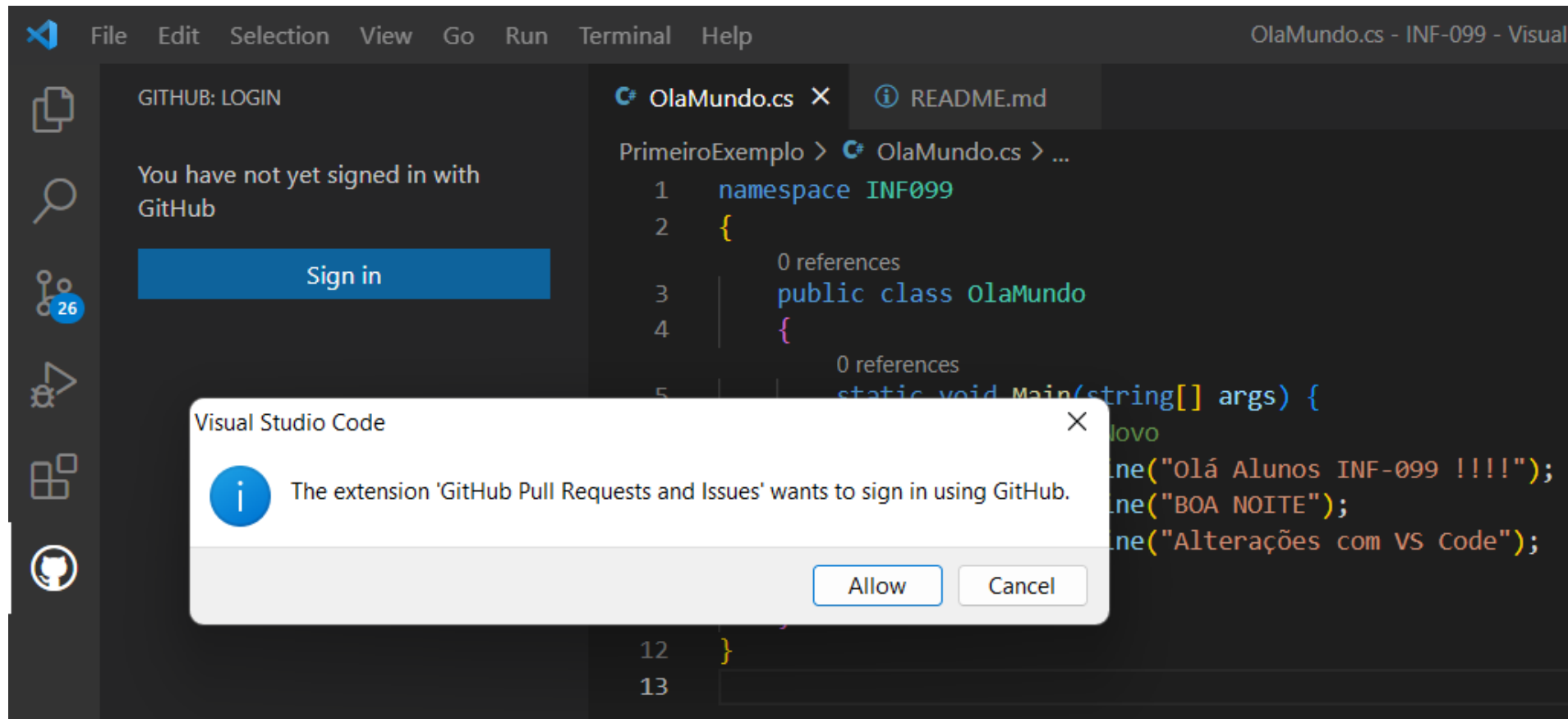
The screenshot shows the Visual Studio Code interface with a commit message editor open. The editor contains the following text:

```
.git > COMMIT_EDITMSG
1 Mais um comentario
2 # Please enter the commit message for your changes. Lines starting
3 # with '#' will be ignored, and an empty message aborts the commit
4 #
5 # On branch Teste2
6 # Changes to be committed:
7 #   new file:   .gitignore
8 #   modified:   PrimeiroExemplo/OlaMundo.cs
9 #
10
```

A dialog box titled "Visual Studio Code" is displayed in the foreground, asking: "The branch 'Teste2' has no remote branch. Would you like to publish this branch?". The dialog has "OK" and "Cancel" buttons. A blue arrow points to the "Accept Cc" button in the top right corner of the commit editor.



VS Code + GitHub



VS Code + GitHub



Authorize GitHub for VS Code



GitHub for VS Code by **Visual-Studio-Code**

wants to access your **RBonacin** account



Personal user data

Email addresses (read-only), profile information (read-only)



Repositories

Public and private



Workflow

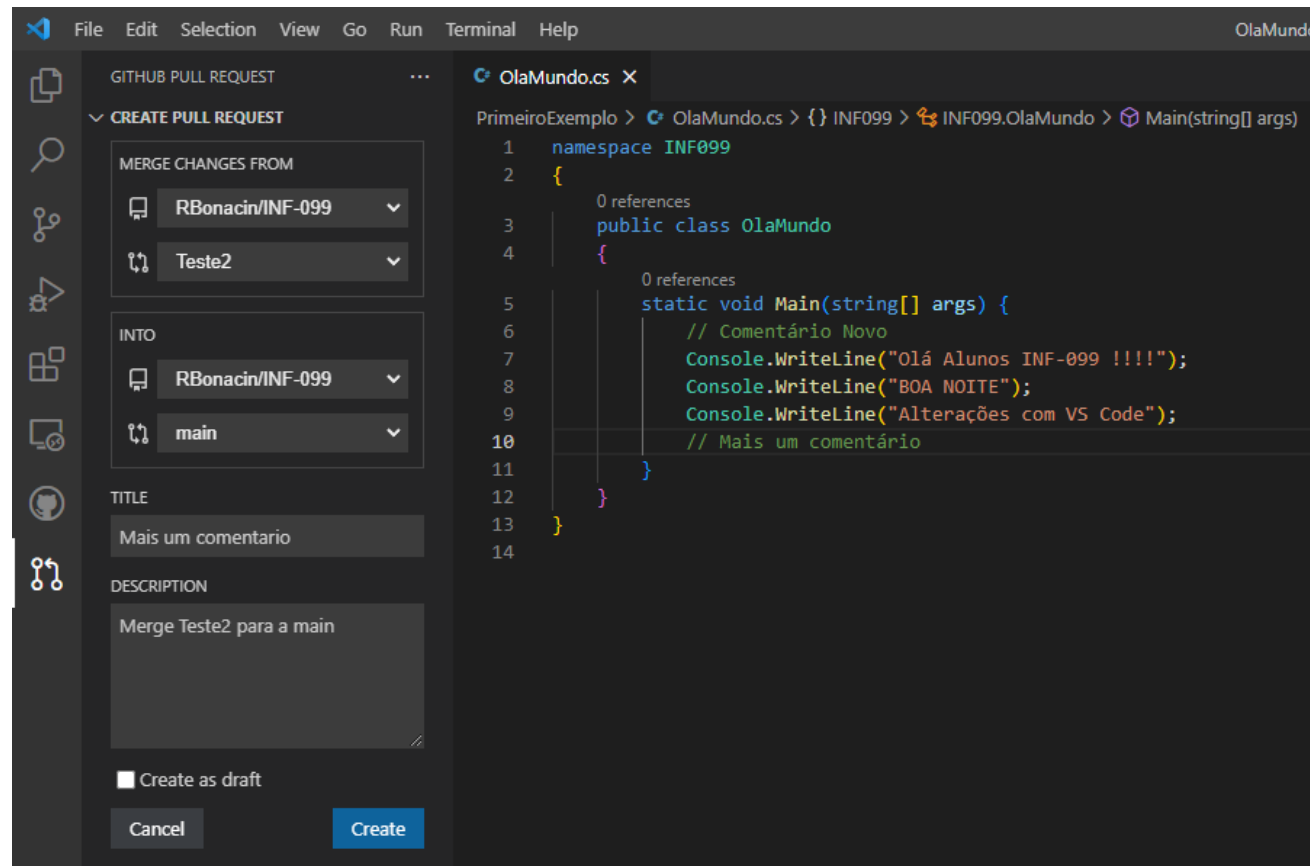
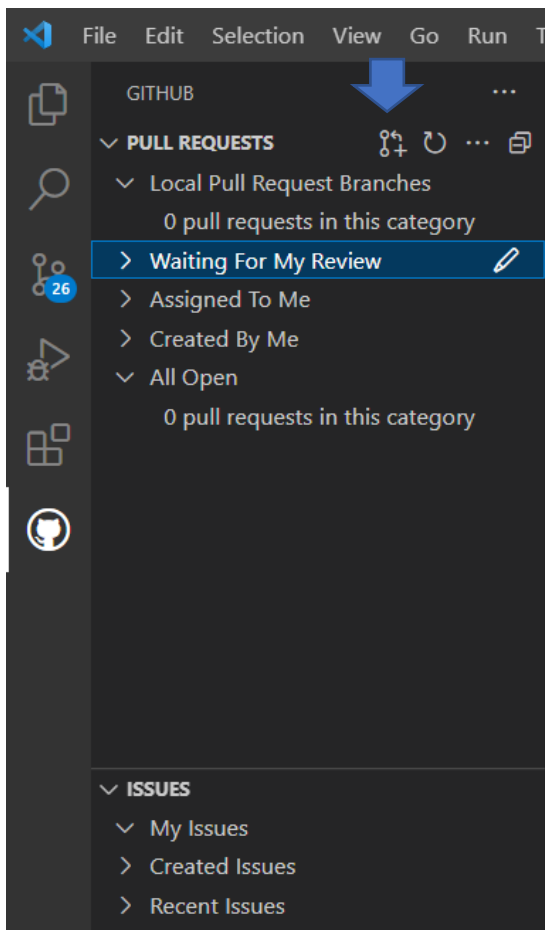
Update GitHub Action Workflow files.



Cancel

Authorize Visual-Studio-
Code

VS Code + GitHub



VS Code + GitHub



Mais um comentario #1

✓ Checked Out

Checkout 'main'

Refresh

Open

RBonacin wants to merge changes into **RBonacin:main** from **RBonacin:Teste2**

Created seconds ago

RBonacin commented seconds ago

Merge Teste2 para a main

RBonacin Mais um comentario

e19daed 37 minutes ago

✓ This branch has no conflicts with the base branch.

Merge Pull Request

 using method

Create Merge Commit

Leave a comment

Close Pull Request

Comment

Reviewers +
None yet

Assignees +
None yet—[assign yourself](#)

Labels +
None yet

Milestone +
No milestone

VS Code + GitHub



main ▾

INF-099 / PrimeiroExemplo / OlaMundo.cs / <> Jump to ▾



RBonacin Mais um comentario

1 contributor

13 lines (13 sloc) | 322 Bytes

```
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10            // Mais um comentário
11        }
12    }
13 }
```

VS Code + GitHub



2 Selected OK

✓ Delete remote branch origin/Teste2 <https://github.com/RBonacin/INF-099>

✓ Delete local branch Teste2

Checked Out Checkout 'main' Refresh

Merged RBonacin merged changes into RBonacin:main from RBonacin:Teste2 Created 2 minutes ago

RBonacin commented 4 minutes ago

Merge Teste2 para a main

RBonacin Mais um comentario e19daed 38 minutes ago

Pull request successfully merged. Delete branch...

Leave a comment

Close Pull Request Comment

Reviewers +
None yet

Assignees +
None yet—[assign yourself](#)

Labels +
None yet

Milestone +
No milestone

RBonacin deleted the Teste2 branch seconds ago

This pull request is closed.

main INF-099 / PrimeiroExemplo

Switch branches/tags

Find or create a branch...

Branches Tags

✓ main default

[View all branches](#)

1 namespace INF099

The background features a network of gray lines connecting various colored circles (orange, yellow, light blue, green) in a non-linear fashion. A solid blue horizontal band spans the width of the slide, serving as a backdrop for the title.

Boas Práticas em Controle de Versão



- Manter sempre seu repositório atualizado
 - Atualizar todos os arquivos antes de alterar ou utilizar algo
- Inserir comentários nos *commits*
- Não adicionar itens que não pertençam ao projeto
- Usar *branch* locais para fazer testes locais



- Cada equipe com seu *branch*
 - Crie *branch* internas se necessário
- Não fazer *rebase* com o *branch* de outras equipe (ou membros)
- O *branch* “master/main” deve estar sempre funcional
 - *Features* novas (e não testadas) em *branches* separados
 - E só depois integradas



<https://opensource.com/article/20/7/git-best-practices>

- Formalize as convenções do Git para sua equipe
 - Instruções sobre boas práticas, convenções de nomes, instruções básicas de uso ...
- Mesclar as alterações corretamente
- *Rebase* sua ramificação com frequência
- *Squash* (vários em 1) os *commits* antes do merge
- Utilizar *Tags*
- Faça o software executável imprimir a *Tag*



<https://deepsources.io/blog/git-best-practices/>

- Faça *commits* limpos e de propósito único
 - Tornar mais fácil para outras pessoas da equipe analisarem sua alteração, tornando as revisões de código mais eficientes.
 - Se o *commit* tiver que ser revertido completamente, é muito mais fácil fazê-lo
 - É simples acompanhar essas alterações com seu sistema de *ticketing*.



<https://deepsources.io/blog/git-best-practices/>

- Escreva mensagens de *commit* significativas

```
feat: add beta sequence
^--^  ^-----^
|      |
|      +--> Summary in present tense.
|
+-----> Type: chore, docs, feat, fix, refactor, style, or test.
```

- Se estiver usando um sistema de *ticket* (controle de incidência, configuração, mudança ...), incluir também o *id* do ticket na descrição



<https://deepsources.io/blog/git-best-practices/>

- Faça *commit* sempre, faça *commit* com frequência
 - Em vez de esperar para fazer o *commit* perfeito, é melhor trabalhar em pequenos pedaços e continuar confirmando seu trabalho (não é fazer *commit* de versões instáveis)
 - Se você estiver trabalhando em um *branch* que pode levar algum tempo para ser concluído, isso ajuda a manter seu código atualizado com as alterações mais recentes para evitar conflitos



<https://deepsources.io/blog/git-best-practices/>

- Não altere o histórico publicado
 - Uma vez que um *commit* tenha sido mesclado em um *branch* padrão (e esteja visível para outros), é altamente recomendável não alterar o histórico
 - Embora o *rebase* seja um recurso útil, ele deve ser usado apenas em *branches* com os quais você está trabalhando



<https://deepsources.io/blog/git-best-practices/>

- Não faça *commit* de arquivos gerados
 - Normalmente, apenas fazemos *commit* (controlamos) arquivos que exigiram esforço manual, ou seja, que não podem ser regenerados automaticamente
 - É útil adicionar um arquivo *.gitignore* na raiz do seu repositório para informar automaticamente ao Git quais arquivos ou caminhos você não deseja rastrear (ex: *.dll, *.exe ...)



<https://sourcelevel.io/blog/7-git-best-practices-to-start-using-in-your-next-commit>

1. Não faça *git push* direto no *master*. Ramifique-o!

- Enviar o código diretamente para o *master* não promove a colaboração
- O Git simplifica a comparação de código entre dois *branches*
 - Isso pode gerar discussões saudáveis, melhorar a qualidade do código base e propagar conhecimento entre os desenvolvedores



<https://sourcelevel.io/blog/7-git-best-practices-to-start-using-in-your-next-commit>

2. Configure adequadamente a autoria do *commit*

- Você deve pelo menos definir seu nome e endereço de e-mail corretamente
- *Commits* são ferramentas de comunicação, saber quem fez uma alteração específica pode ajudá-lo no futuro

```
D:\Bonacin\ICUnicamp\Curso_Samsung_MS\Controle_Versao\MeuProjeto\INF-099>git blame README.md
^9e3098f (Rodrigo Bonacin 2022-08-12 16:43:13 -0300 1) INF-099 - Controle de Versão
22d99455 (Rodrigo Bonacin 2022-08-14 12:22:39 -0300 2) Curso Extensão Tecnologias Microsoft - IC - UNICAMP
^9e3098f (Rodrigo Bonacin 2022-08-12 16:43:13 -0300 3) Este repositório tem fins didáticos para a referida disciplina.
782cdd6a (Rodrigo Bonacin 2022-08-14 18:13:18 -0300 4) Professor Rodrigo Bonacin
2f2d5695 (Rodrigo Bonacin 2022-08-14 19:19:47 -0300 5) Agosto de 2022.
```

- Mas não use “blame” para culpar



<https://sourcelevel.io/blog/7-git-best-practices-to-start-using-in-your-next-commit>

3. Escreva mensagens de confirmação descritivas e significativas

<https://cbea.ms/git-commit/>

DEVELOPMENT

How to Write a Git Commit Message

Commit messages matter. Here's how to write them well.



CBEAMS
31 AUG 2014 • 11 MIN READ

	COMMENT	DATE
○	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
○	ENABLED CONFIG FILE PARSING	9 HOURS AGO
○	MISC BUGFIXES	5 HOURS AGO
○	CODE ADDITIONS/EDITS	4 HOURS AGO
○	MORE CODE	4 HOURS AGO
○	HERE HAVE CODE	4 HOURS AGO
○	AAAAAAA	3 HOURS AGO
○	ADKFJSLKDFJSDKLFJ	3 HOURS AGO
○	MY HANDS ARE TYPING WORDS	2 HOURS AGO
○	HAAAAAAAANDS	2 HOURS AGO

AS A PROJECT DRAGS ON, MY GIT COMMIT MESSAGES GET LESS AND LESS INFORMATIVE.



<https://sourcelevel.io/blog/7-git-best-practices-to-start-using-in-your-next-commit>

4. Faça *commit* apenas de trabalhos relacionados

- Faça *commit* de uma quantidade mínima de linhas que façam sentido juntas
 - O *commit* não deve ser de x e y e z (que não são relacionados)
 - *single-responsibility principle*



<https://sourcelevel.io/blog/7-git-best-practices-to-start-using-in-your-next-commit>

5. Evite reescrever a história do *branch mestre*

- *Git rebase* é útil, mas usar git rebase em *branches* publicados, como o master, pode gerar muitos conflitos para os outros contribuidores
- Em *git push -force* , sempre especificar o branch destino para evitar erros
 - Evite em qualquer *branch* publicado



<https://sourcelevel.io/blog/7-git-best-practices-to-start-using-in-your-next-commit>

6. *Rebase* seu *branch* de trabalho com frequência

- É crucial sempre manter seu *branch rebase* com o código mais recente
- Você deve fazer *rebase* para evitar bugs, retrabalho e o trabalho tedioso de resolver conflitos com o *branch upstream* (remoto)

- `git checkout <upstream_branch>`
- `git pull`
- `git checkout -`
- `git rebase <upstream_branch>`



<https://sourcelevel.io/blog/7-git-best-practices-to-start-using-in-your-next-commit>

7. Conheça a ferramenta. Não tenha medo de usá-la.

- O git tem diversos comandos úteis que vão além
 - Veja: *git help -a*
 - Ex:
 - `git cherry-pick`
 - `git diff` and `git apply`
 - `git stash`
 - `git bisect`



<https://www.git-tower.com/learn/git/ebook/en/command-line/appendix/best-practices>

- a) Fazer *commit* de alterações relacionadas
...
- b) Fazer *commit* frequentemente
...
- c) Não fazer *commit* de trabalho pela metade
...
- d) Teste antes de fazer *commit*
...
- e) Escreva boas mensagens de commit
...



<https://www.git-tower.com/learn/git/ebook/en/command-line/appendix/best-practices>

f) O controle de versão não é um sistema de backup

- Fazer backup de seus arquivos em um servidor remoto é um bom efeito colateral de ter um sistema de controle de versão
- Mas você não deve usar seu VCS como se fosse um sistema de backup
- Ao fazer o controle de versão, você deve prestar atenção a semântica do *commit* (veja “mudanças relacionadas”) – você não deve apenas amontoar arquivos



<https://www.git-tower.com/learn/git/ebook/en/command-line/appendix/best-practices>

g) Utilizar *Branches*

- É fácil de usar, é um requisito básico do git ...

h) Concordar em um fluxo de trabalho

- O Git permite que você escolha entre muitos fluxos de trabalho diferentes: *branches* de longa duração, *branches* de tópicos, *merge* ou *rebase*, *git-flow*...
- A escolha depende de alguns fatores:
 - Seu projeto, seus fluxos de trabalho gerais de desenvolvimento e implantação e (talvez o mais importante) das preferências pessoais suas e de seus colegas de equipe
- Seja como for que você escolha trabalhar, apenas certifique-se de concordar com um fluxo de trabalho comum que todos seguem



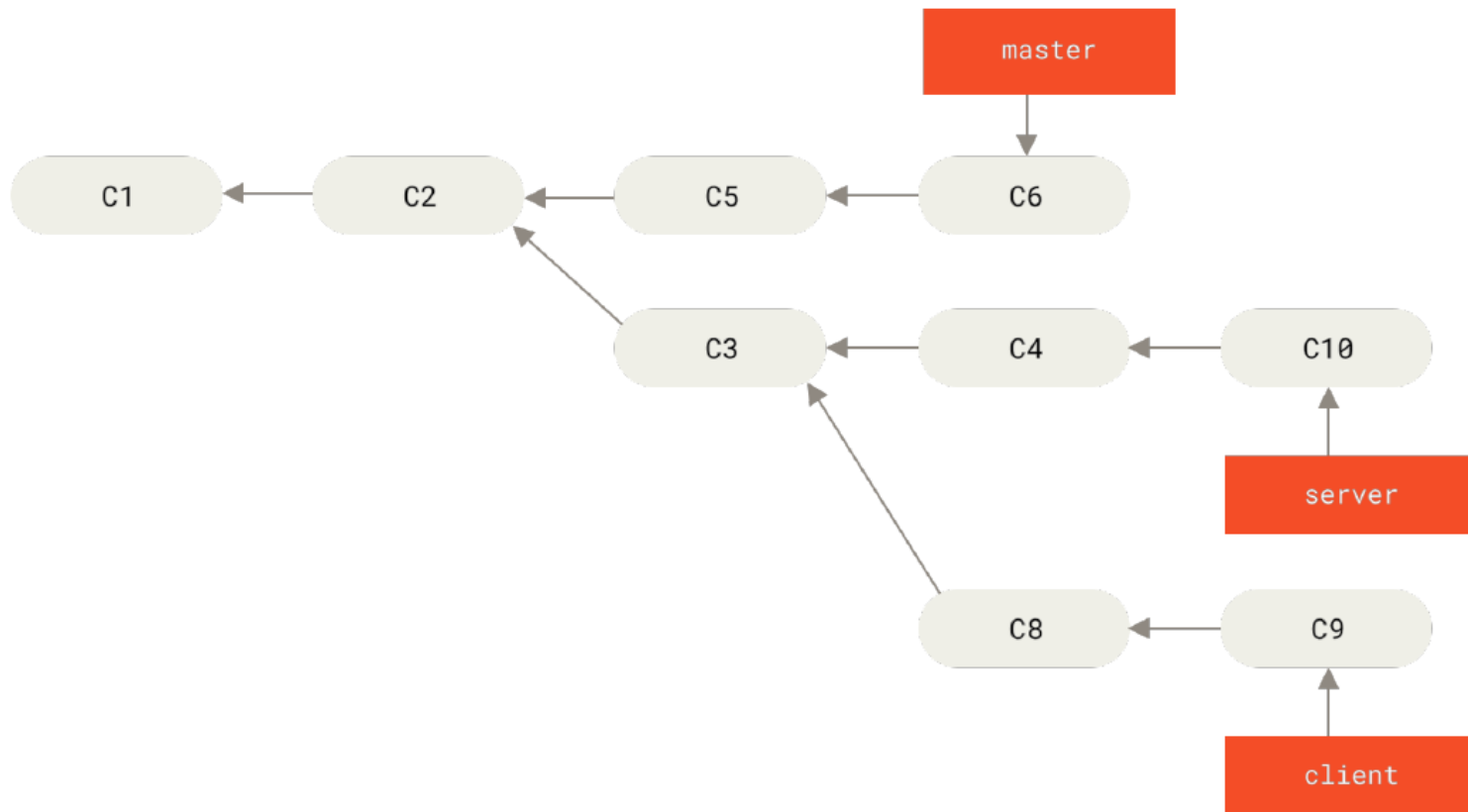
• • •

The background features a network of gray lines connecting various colored circles (orange, yellow, light blue, green) in a non-linear fashion. A dark blue horizontal band with a light blue border is positioned across the middle of the slide.

Prática com VS Code + GitHub



1) Vamos construir situação equivalente com VS Code + GitHub



VS Code + GitHub

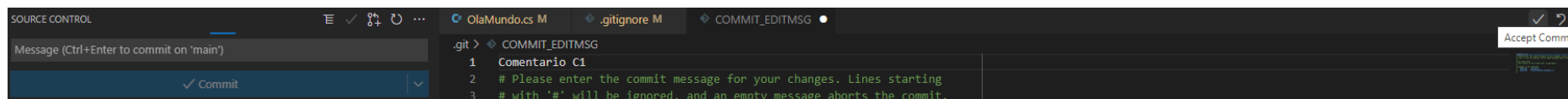


The Explorer sidebar on the left shows the project structure:

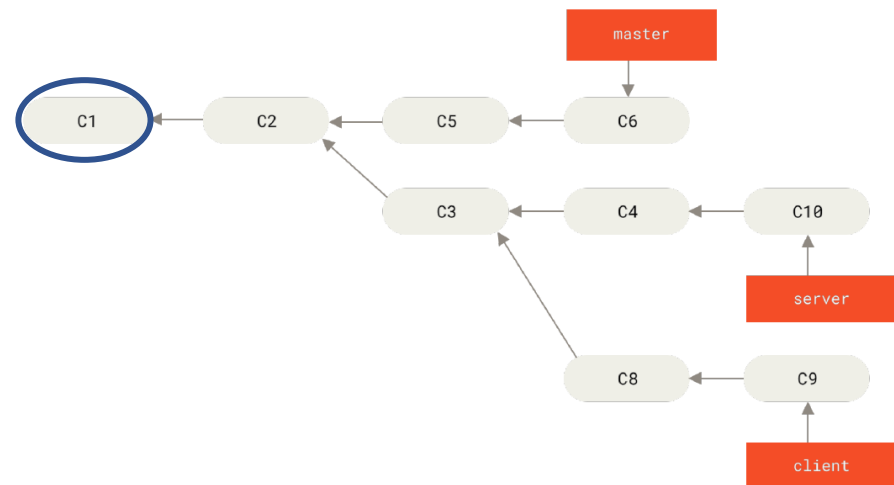
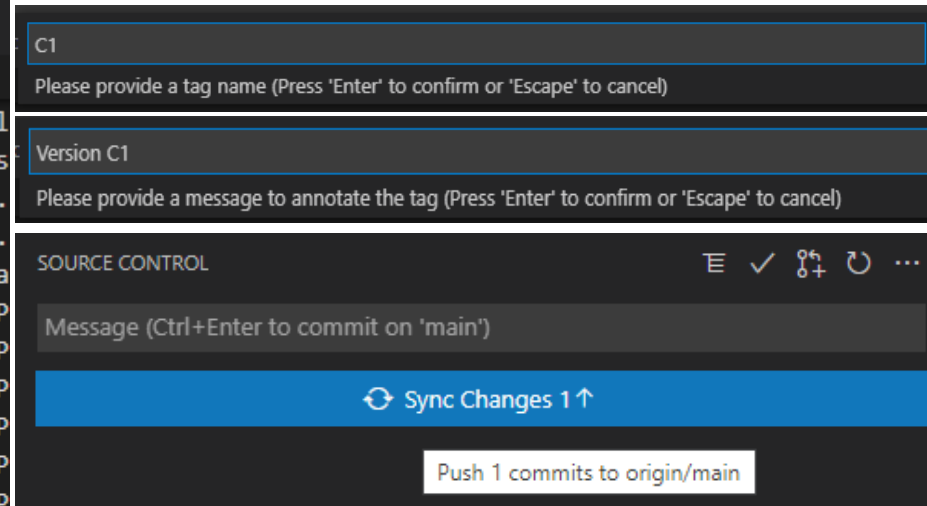
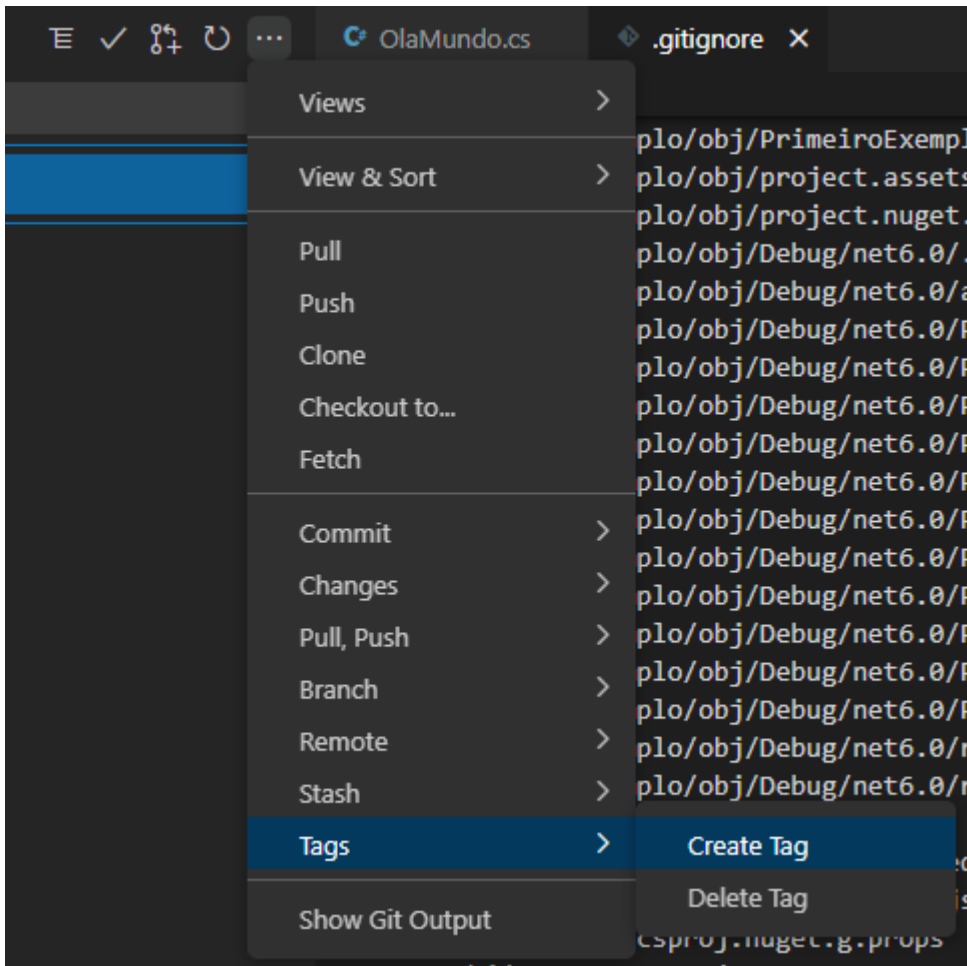
- INF-099
 - .vscode
 - bin
 - obj
 - PrimeiroExemplo
 - bin
 - obj
 - OlaMundo.cs (selected)
 - PrimeiroExemplo.csproj
 - .gitignore
 - INF-099.csproj
 - README.md

The Editor shows the content of `OlaMundo.cs`:

```
PrimeiroExemplo > OlaMundo.cs > {} INF099 > INF099.OlaMundo > Main(string[] args)
1 namespace INF099
2 {
3     0 references
4     public class OlaMundo
5     {
6         0 references
7         static void Main(string[] args) {
8             // Comentário Novo
9             Console.WriteLine("Olá Alunos INF-099 !!!!");
10            Console.WriteLine("BOA NOITE");
11            Console.WriteLine("Alterações com VS Code");
12            // Mais um comentário
13            // C1
14        }
15    }
16 }
```



VS Code + GitHub



VS Code + GitHub



PrimeiroExemplo > OlaMundo.cs > {} INF099 > INF099.OlaMundo > Main(string[] args)

```
1 namespace INF099
2 {
3     0 references
4     public class OlaMundo
5     {
6         0 references
7         static void Main(string[] args) {
8             // Comentário Novo
9             Console.WriteLine("Olá Alunos INF-099 !!!!");
10            Console.WriteLine("BOA NOITE");
11            Console.WriteLine("Alterações com VS Code");
12            // Mais um comentário
13            // C1
14            // C2
15        }
16    }
```

Stage Changes

main INF-099 / PrimeiroExemplo / OlaMundo.cs / <> Jump to

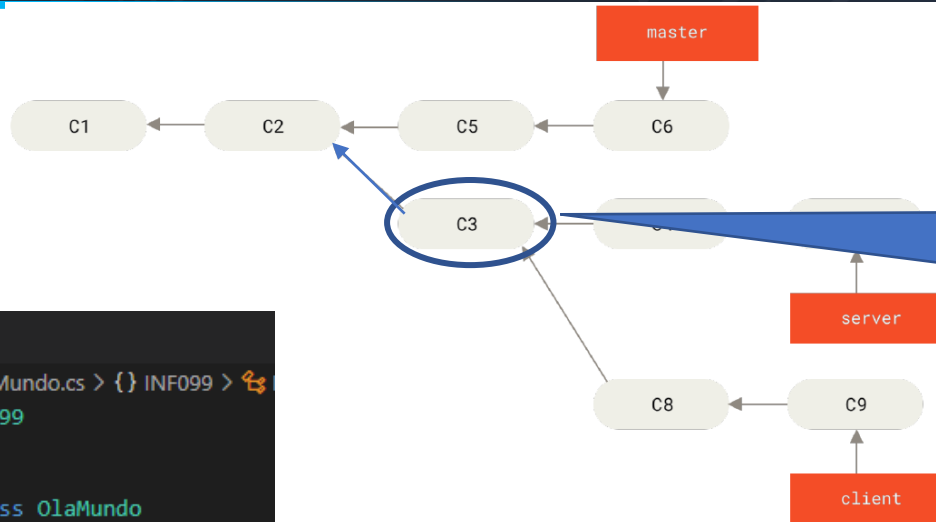
RBonacin Version C2

1 contributor

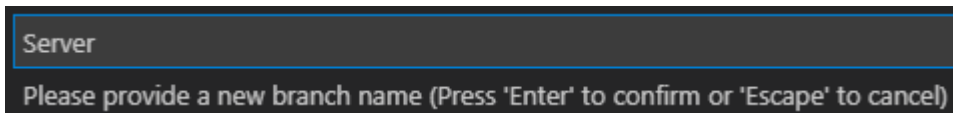
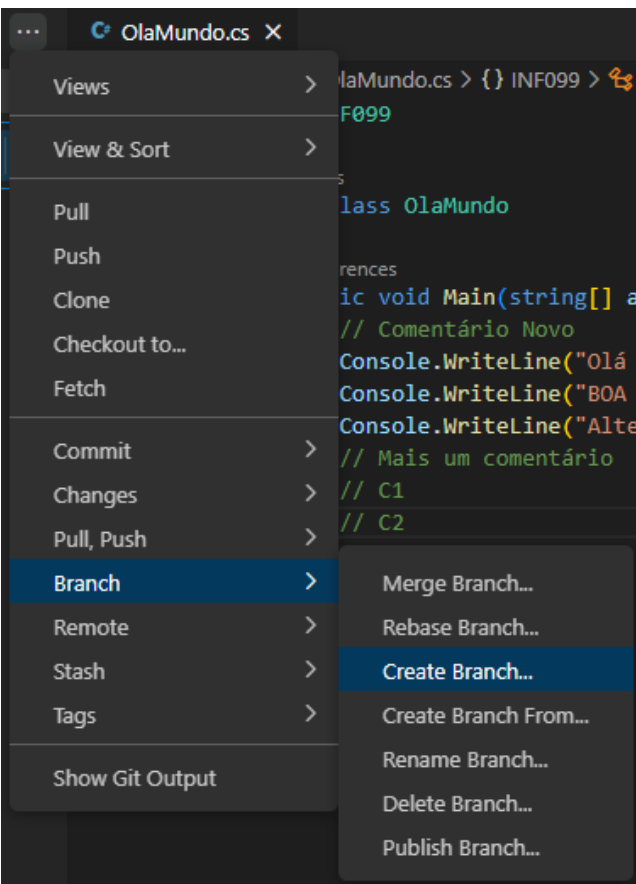
15 lines (15 sloc) | 359 Bytes

```
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10            // Mais um comentário
11            // C1
12            // C2
13        }
14    }
15 }
```

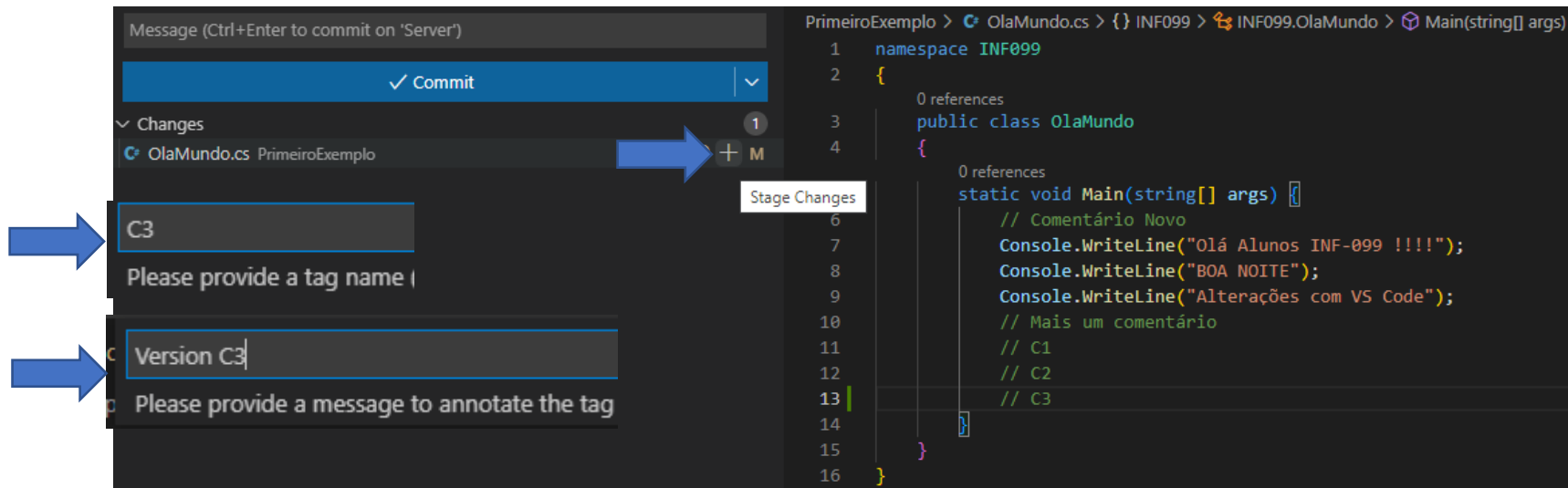
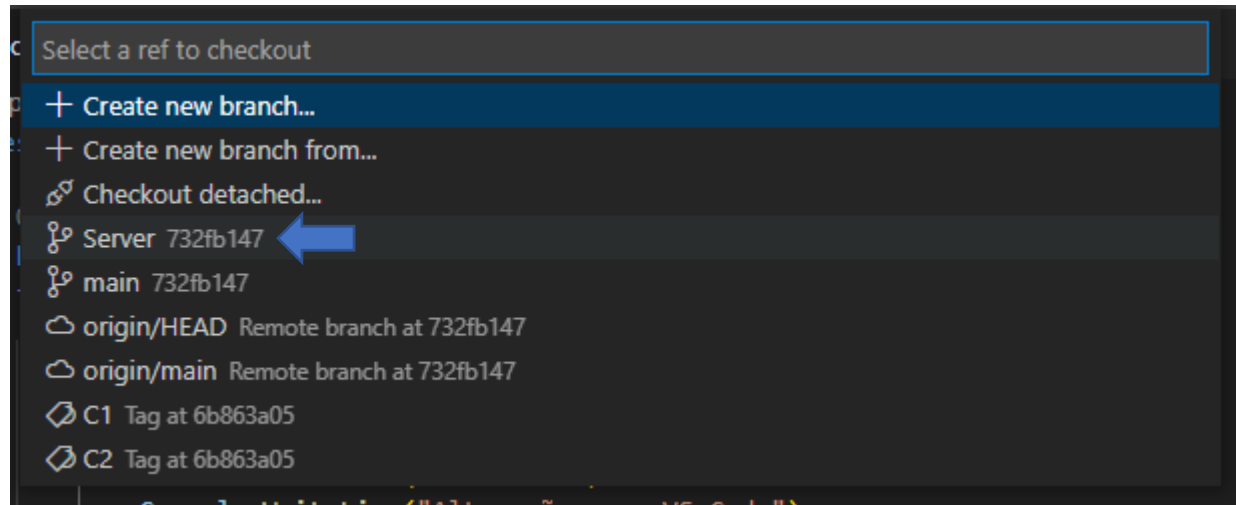
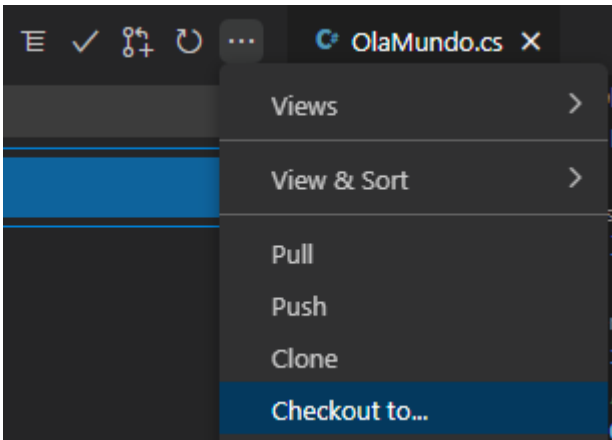
VS Code + GitHub



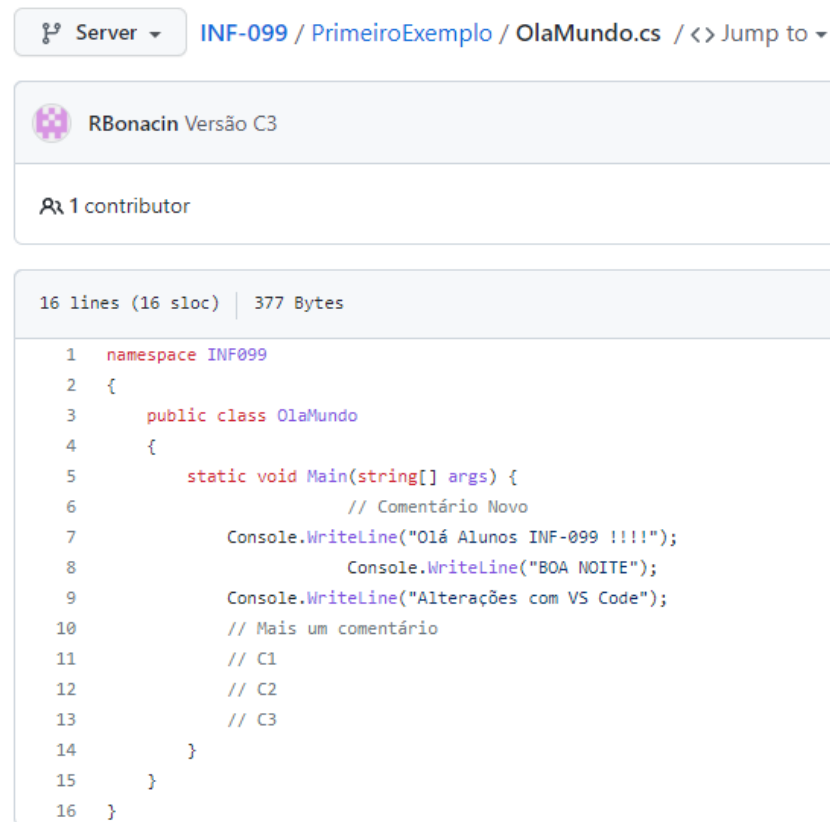
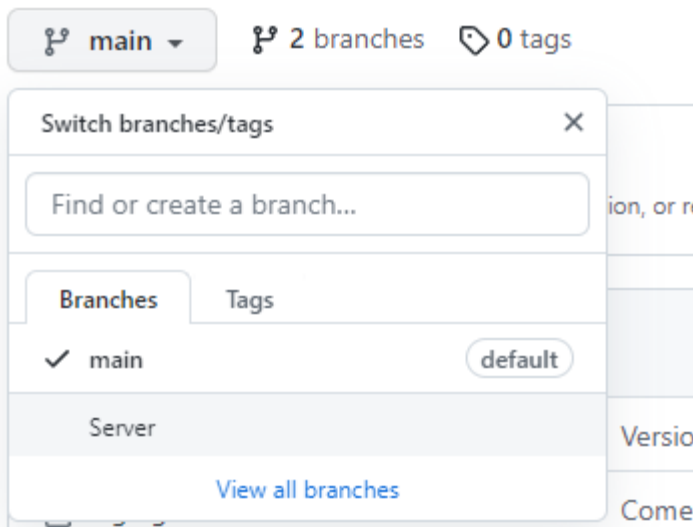
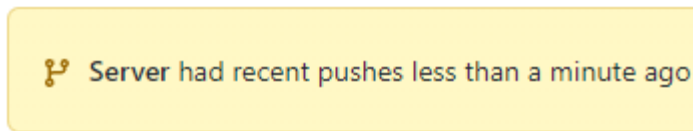
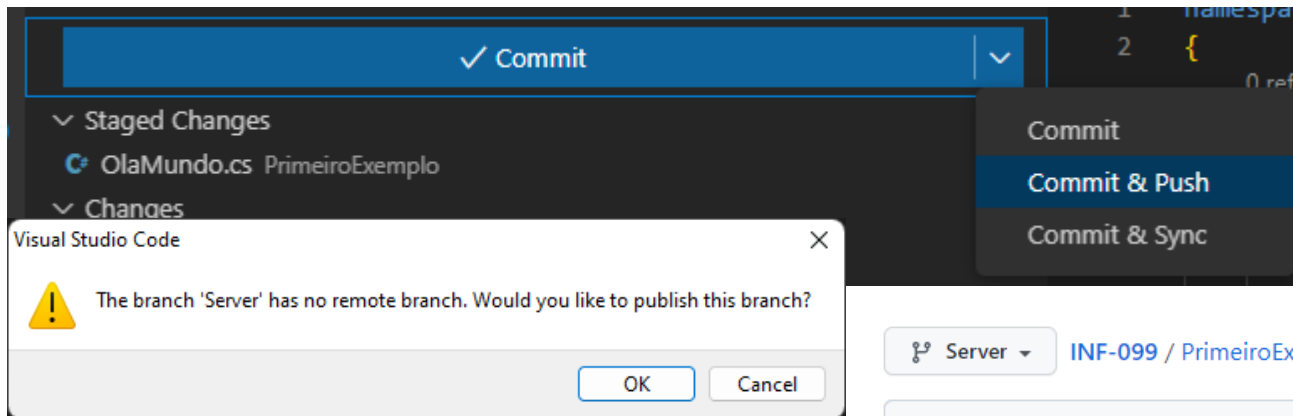
Criar Branch Server,
fazer checkout, alterar
arquivo, fazer commit
e push



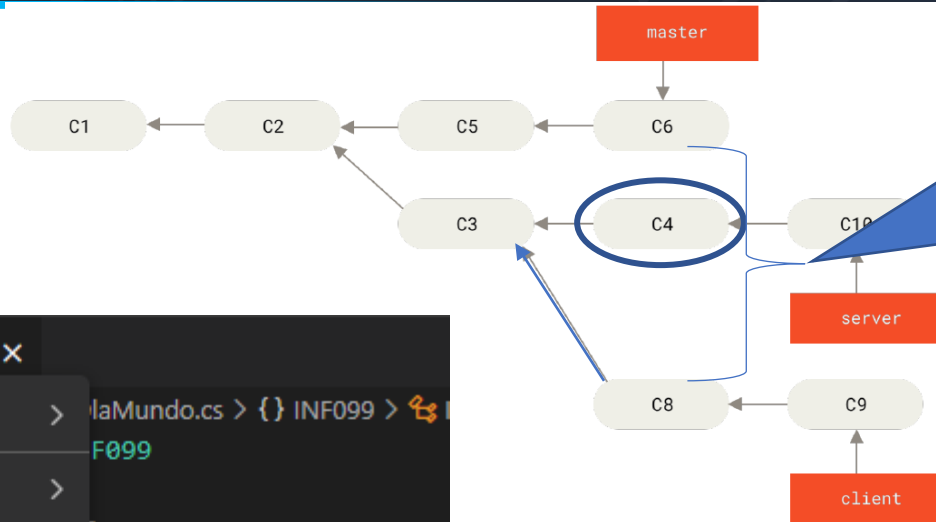
VS Code + GitHub



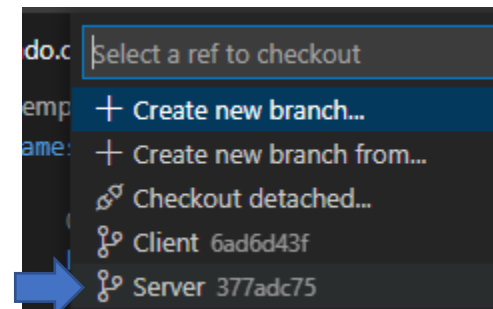
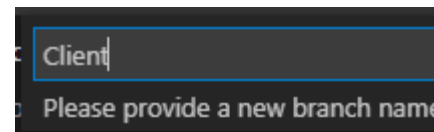
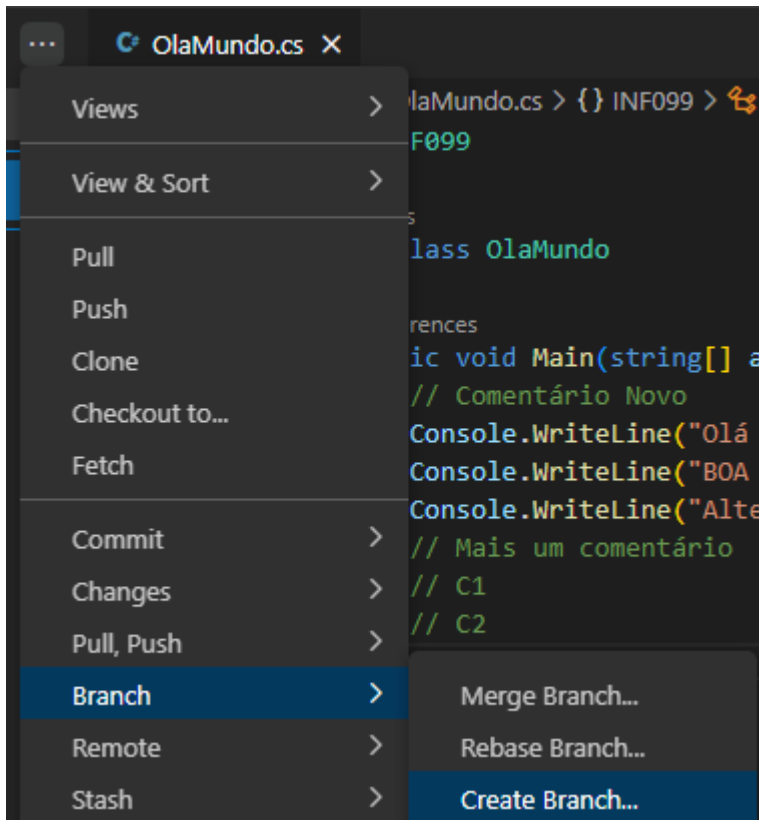
VS Code + GitHub



VS Code + GitHub



Criar Branch Client, mas fazer checkout "na Server". Alterar arquivo, fazer commit e push "na server"



VS Code + GitHub



SOURCE CONTROL

Message (Ctrl+Enter to commit on 'Client')

✓ Commit

Staged Changes: 1

OlaMundo.cs PrimeiroExemplo M

Changes: 0

```
PrimeiroExemplo > OlaMundo.cs > {} INF099 > INF099.OlaMundo > Main(string[] args)
1 namespace INF099
2 {
3     0 references
4     public class OlaMundo
5     {
6         0 references
7         static void Main(string[] args) {
8             // Comentário Novo
9             Console.WriteLine("Olá Alunos INF-099 !!!!");
10            Console.WriteLine("BOA NOITE");
11            Console.WriteLine("Alterações com VS Code");
12            // Mais um comentário
13            // C1
14            // C2
15            // C3
16            // C4
17        }
18    }
19 }
```

OlaMundo.cs M

Views: OlaMundo.cs > {} INF099 > INF099

View & Sort: >

Pull: class OlaMundo

Push: rences

Clone: ic void Main(string[]

Checkout to...: // Comentário Novo

Fetch: Console.WriteLine("Ol

Commit: Console.WriteLine("BO

Changes: Console.WriteLine("Al

Pull, Push: // Mais um comentário

Branch: // C1

Remote: // C2

Stash: // C3

Tags: // C4

Show Git Output: Create Tag

Delete Tag

C4

Please provide a tag name

Version C4

Please provide a message to annotate the tag

VS Code + GitHub



SOURCE CONTROL

Message (Ctrl+Enter to commit on 'Client')

✓ Commit

Staged Changes

OlaMundo.cs PrimeiroExemplo

Changes

OlaMundo.cs M

PrimeiroExemplo >

1 namespace

2 {

0 refe

1

Commit

Commit & Push

Commit & Sync

Server INF-099 / PrimeiroExemplo / OlaMundo.cs / <> Jump to

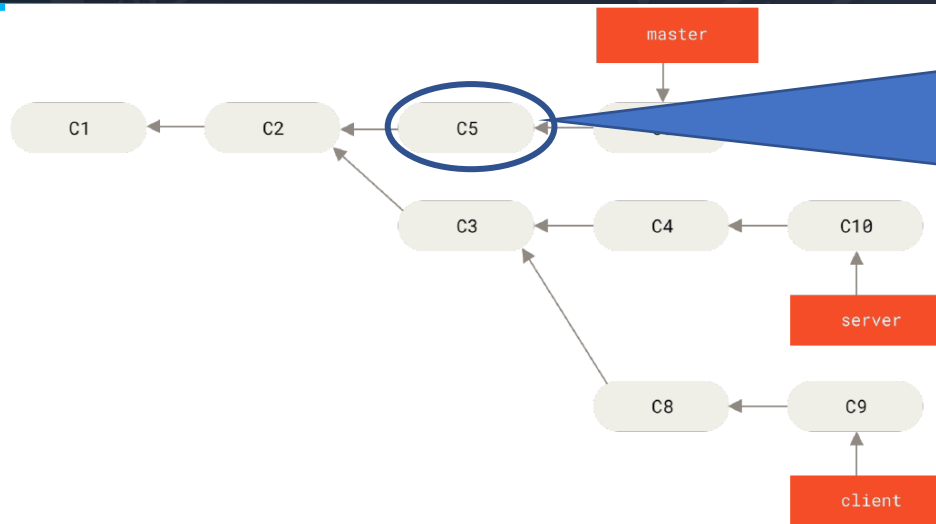
RBonacin Versão C4

1 contributor

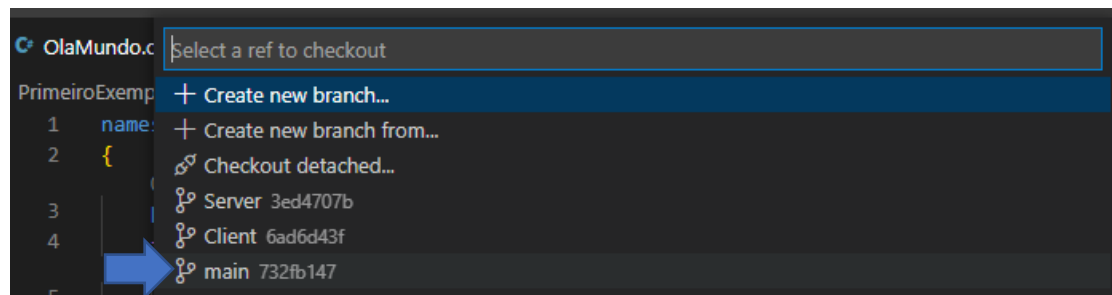
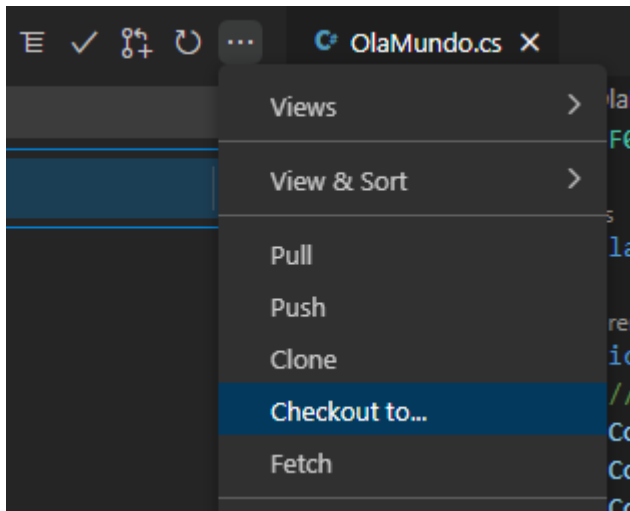
17 lines (17 sloc) | 395 Bytes

```
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10            // Mais um comentário
11            // C1
12            // C2
13            // C3
14            // C4
15        }
16    }
17 }
```

VS Code + GitHub



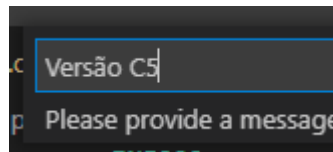
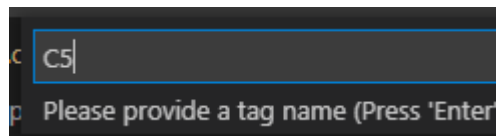
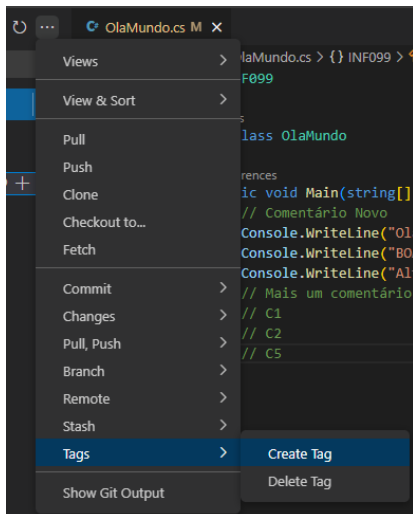
Fazer checkout “na Main/Master”. Alterar arquivo, fazer commit e push



VS Code + GitHub



```
namespace INF099
{
    public class OlaMundo
    {
        static void Main(string[] args) {
            // Comentário Novo
            Console.WriteLine("Olá Alunos INF-099 !!!!");
            Console.WriteLine("BOA NOITE");
            Console.WriteLine("Alterações com VS Code");
            // Mais um comentário
            // C1
            // C2
            // C5
        }
    }
}
```



VS Code + GitHub



SOURCE CONTROL

Message (Ctrl+Enter to commit on 'Client')

✓ Commit

Staged Changes

OlaMundo.cs PrimeiroExemplo

Changes

main INF-099 / PrimeiroExemplo / OlaMundo.cs / <> Jump to



RBonacin Version C5

1 contributor

16 lines (16 sloc) | 377 Bytes

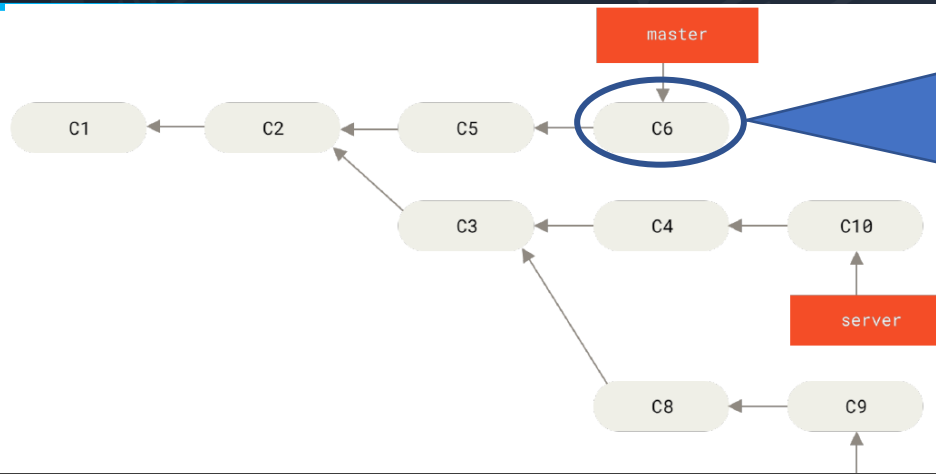
```
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10            // Mais um comentário
11            // C1
12            // C2
13            // C5
14        }
15    }
16 }
```

Commit

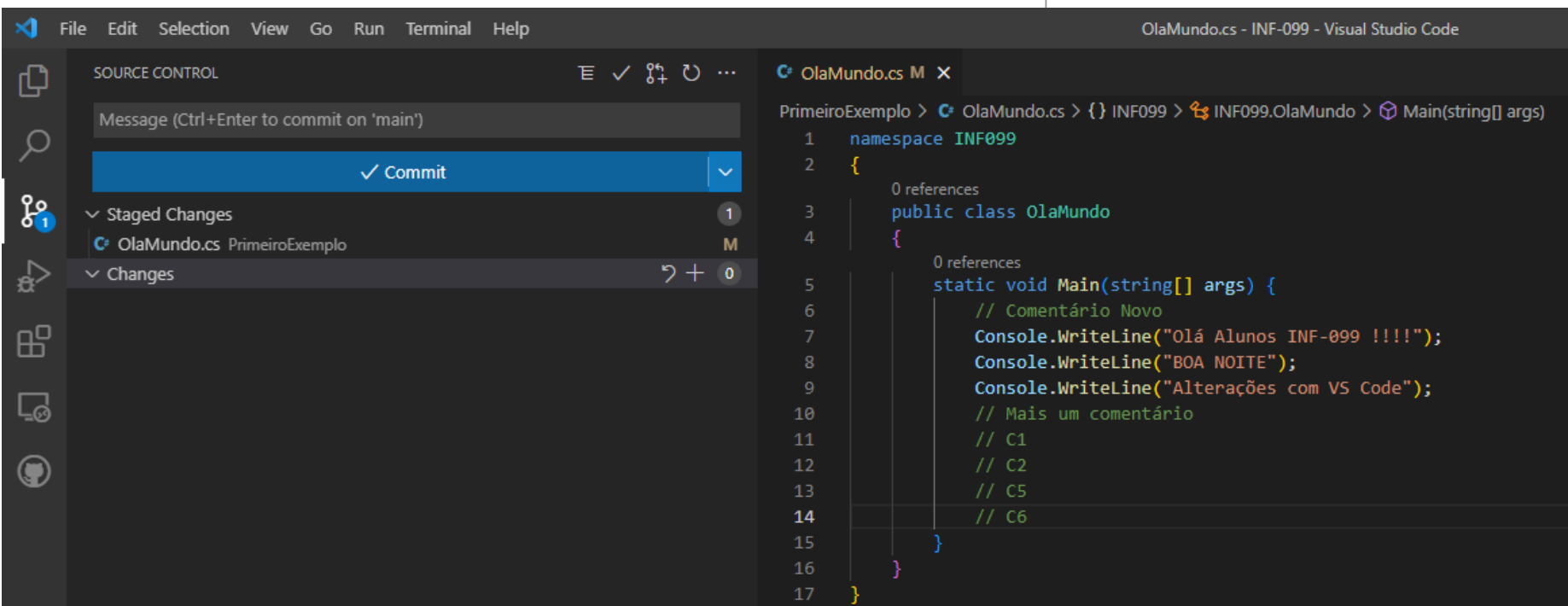
Commit & Push

Commit & Sync

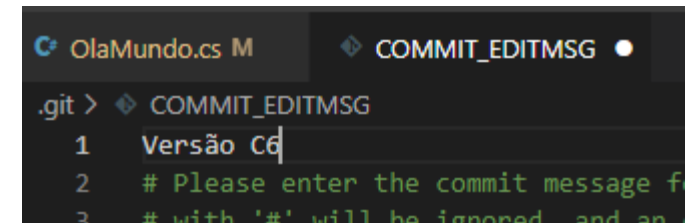
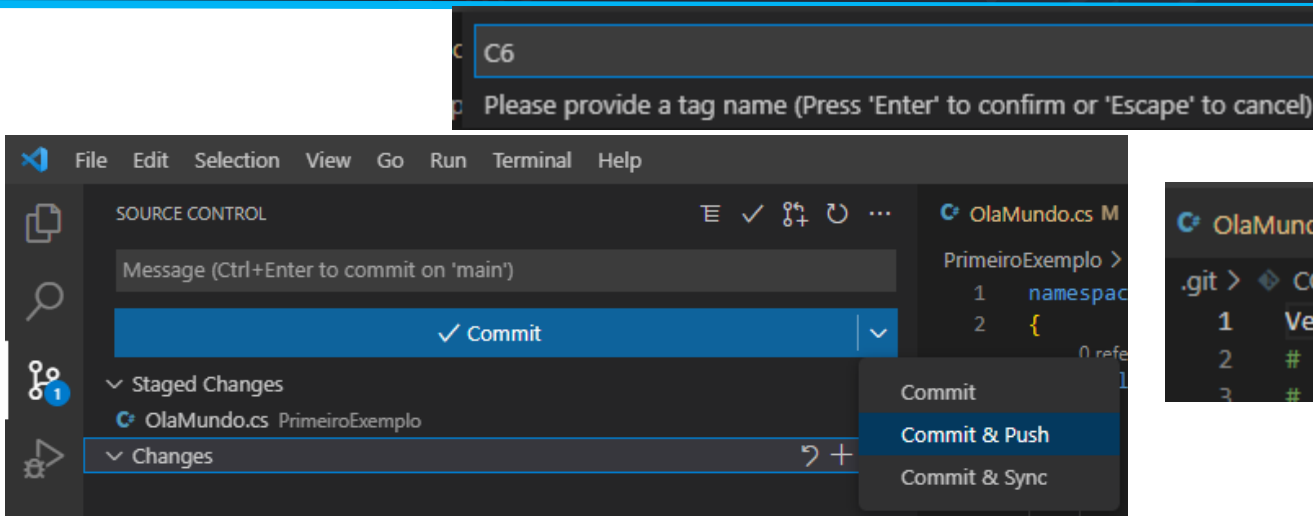
VS Code + GitHub



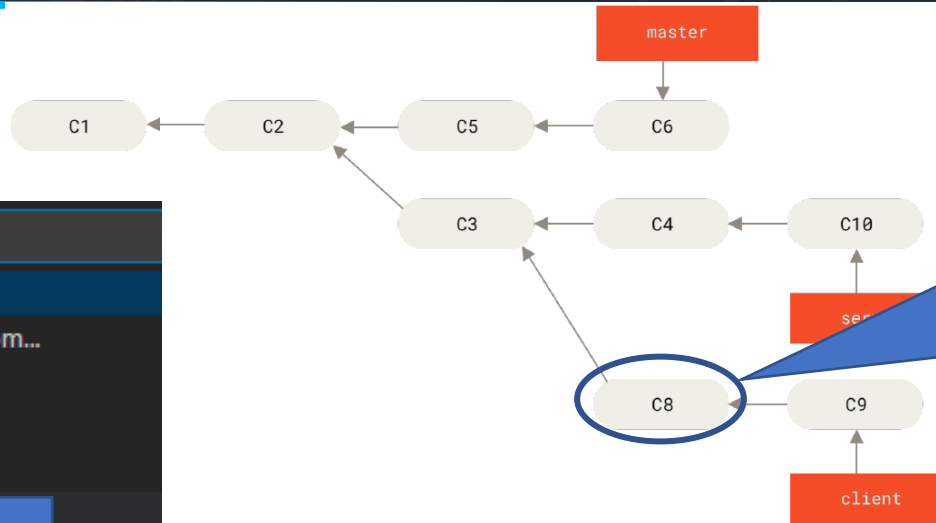
Continuar “na Main/Master”. Alterar arquivo, fazer commit e push



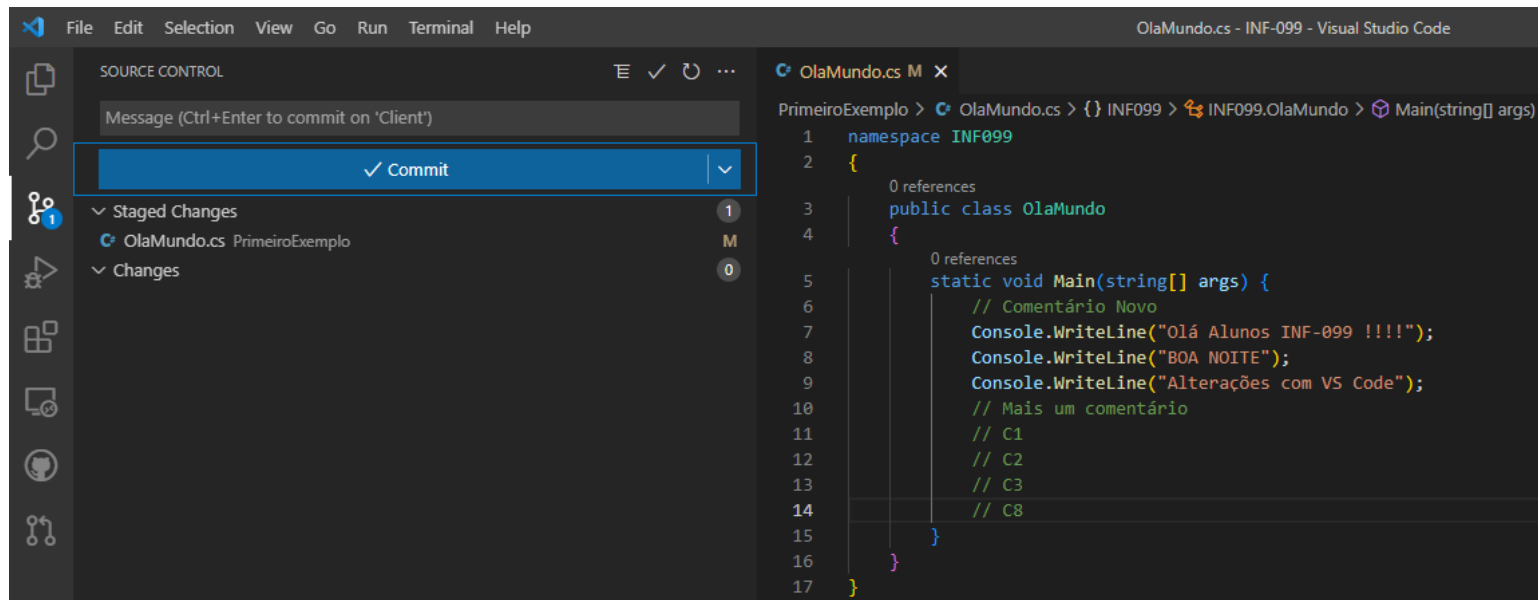
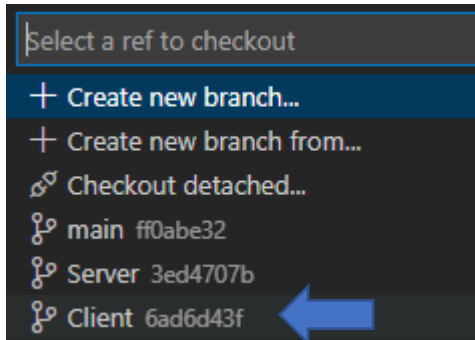
VS Code + GitHub



VS Code + GitHub



Fazer checkout para "Client". Alterar arquivo, fazer commit e push



VS Code + GitHub



C8

Please provide a tag name (Press 'Enter' to confirm or 'Escape' to cancel)

```
.git > COMMIT_EDITMSG
1  Version C8
2  # Please enter the commit message for
3  # your change. Lines starting with
4  # will be ignored, and an empty
```

Commit
Commit & Push
Commit & Sync

Client INF-099 / PrimeiroExemplo / OlaMundo.cs / <> Jump to

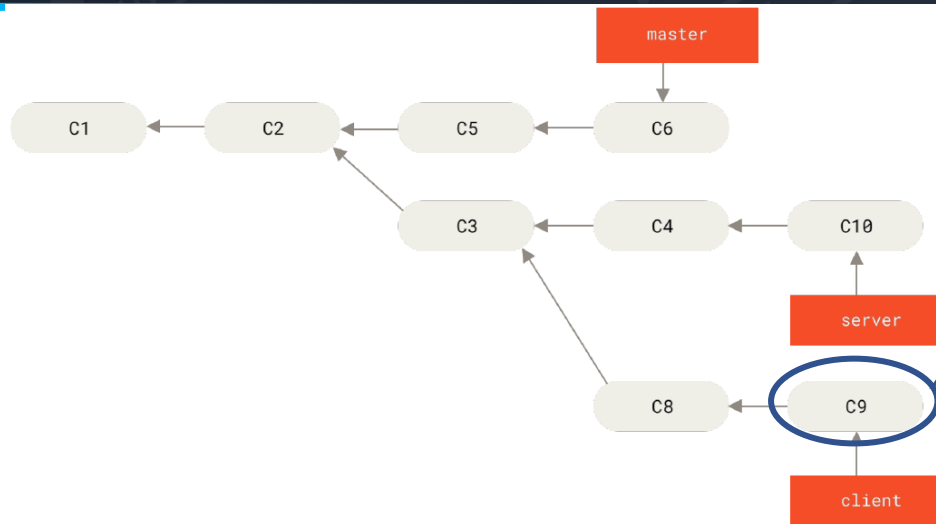
RBonacin Version C8

1 contributor

17 lines (17 sloc) | 395 Bytes

```
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10            // Mais um comentário
11            // C1
12            // C2
13            // C3
14            // C8
15        }
16    }
17 }
```


VS Code + GitHub



```
File Edit Selection View Go Run Terminal Help
SOURCE CONTROL
Message (Ctrl+Enter to commit on 'Client')
Commit
Staged Changes 1
OlaMundo.cs PrimeiroExemplo M
Changes 0

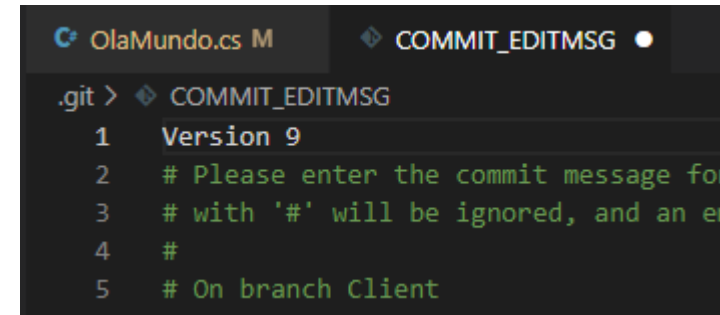
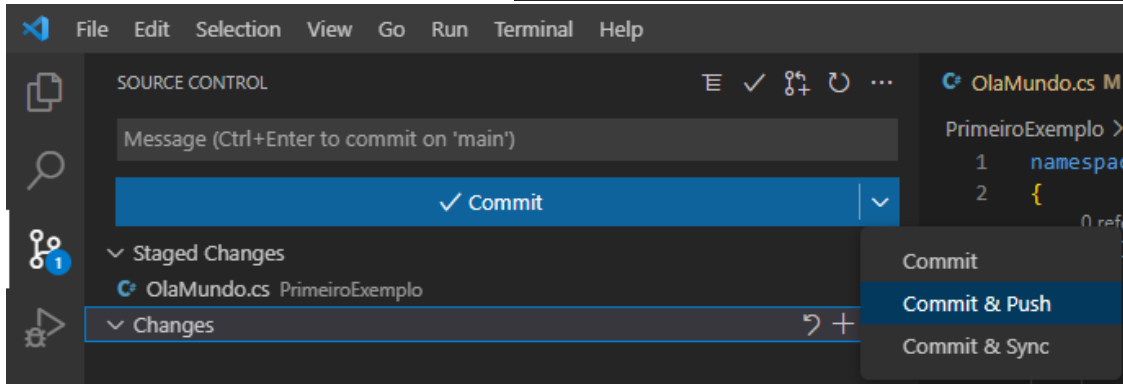
OlaMundo.cs M x
PrimeiroExemplo > OlaMundo.cs > {} INF099 > INF099.OlaMundo > Main(string[] args)
1 namespace INF099
2 {
3     0 references
4     public class OlaMundo
5     {
6         0 references
7         static void Main(string[] args) {
8             // Comentário Novo
9             Console.WriteLine("Olá Alunos INF-099 !!!!");
10            Console.WriteLine("BOA NOITE");
11            Console.WriteLine("Alterações com VS Code");
12            // Mais um comentário
13            // C1
14            // C2
15            // C3
16            // C8
17            // C9
18        }
19    }
20 }
```

VS Code + GitHub

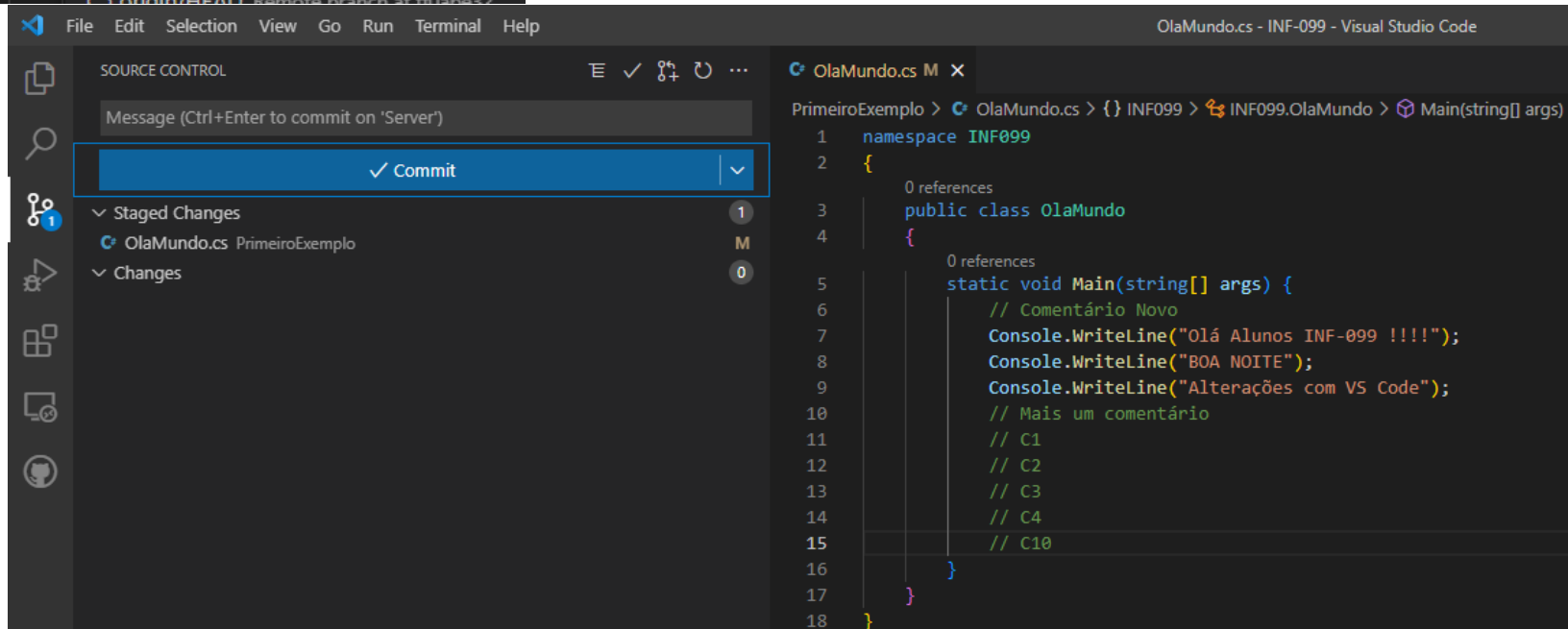
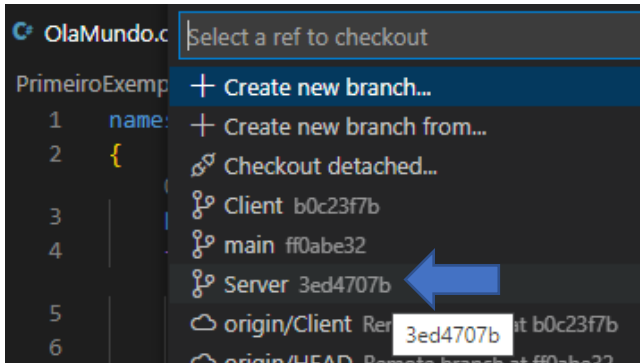
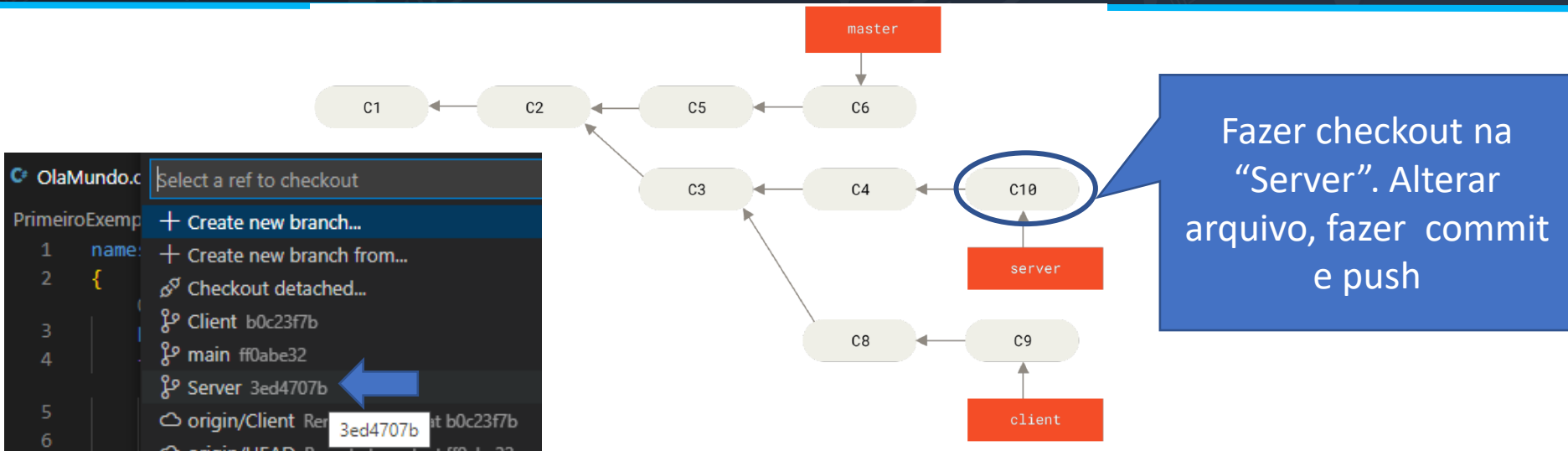


v9

Please provide a tag name (Press 'Enter' to confirm or 'Escape' to cancel)



VS Code + GitHub

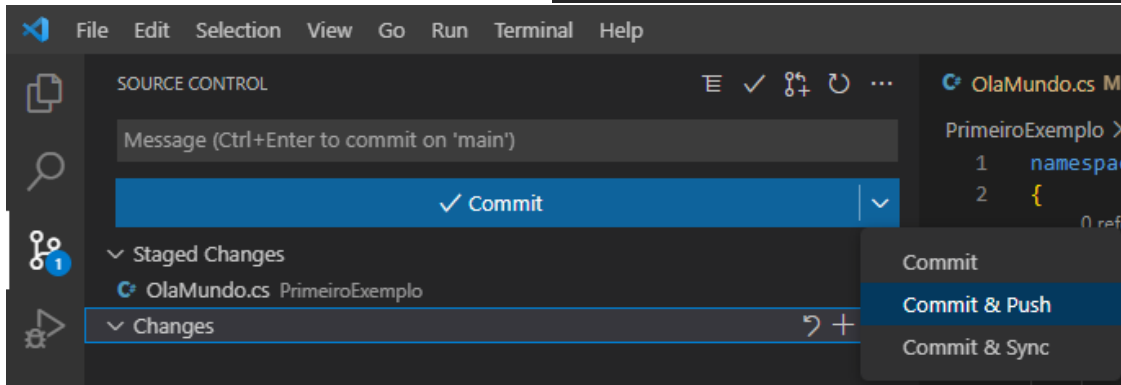


VS Code + GitHub



C10

Please provide a tag name (Press 'Enter' to confirm or 'Escape' to cancel)



```
.git > COMMIT_EDITMSG
1  Version C10
2  # Please enter the commit message
3  # with '#' will be ignored, and a
4  #
5  # On branch Server
```

Server INF-099 / PrimeiroExemplo / OlaMundo.cs / <> Jump to

RBonacin Version C10

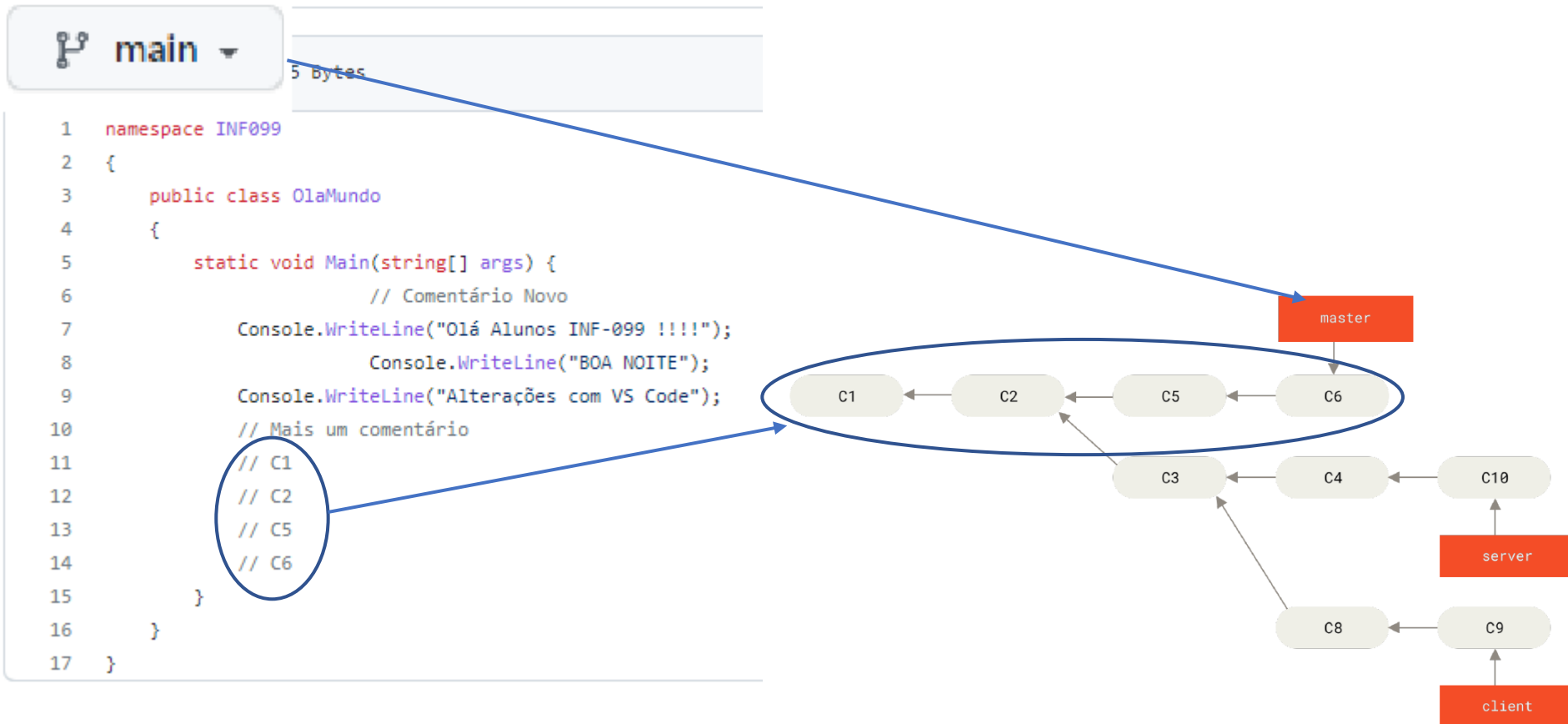
1 contributor

18 lines (18 sloc) | 414 Bytes

```
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10            // Mais um comentário
11            // C1
12            // C2
13            // C3
14            // C4
15            // C10
16        }
17    }
18 }
```



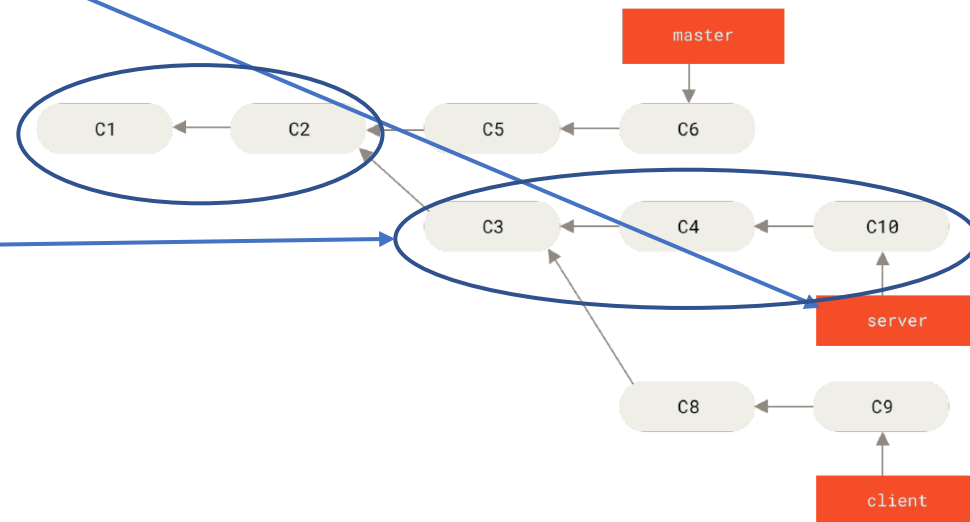
2) Conferir se está correto ...





2) Conferir se está correto ...

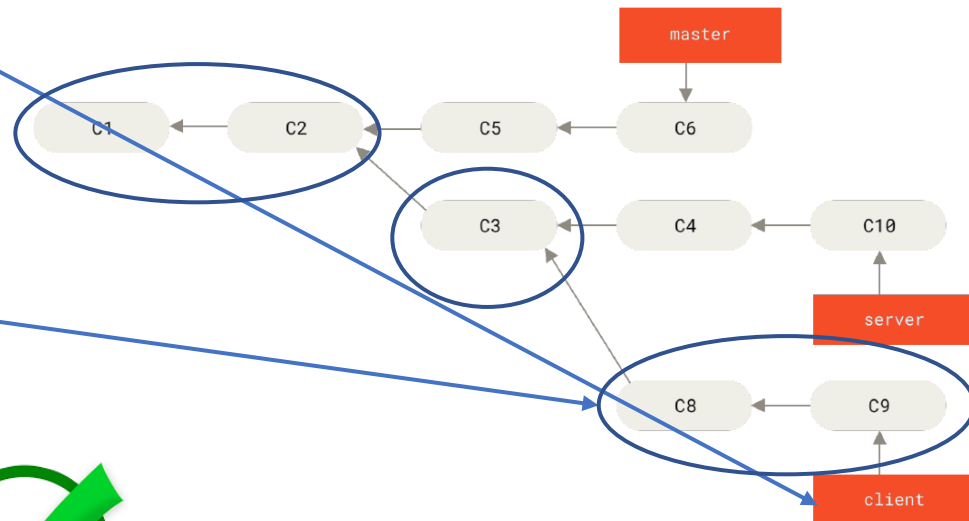
```
Server 414 Bytes
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10            // Mais um comentário
11            // C1
12            // C2
13            // C3
14            // C4
15            // C10
16        }
17    }
18 }
```





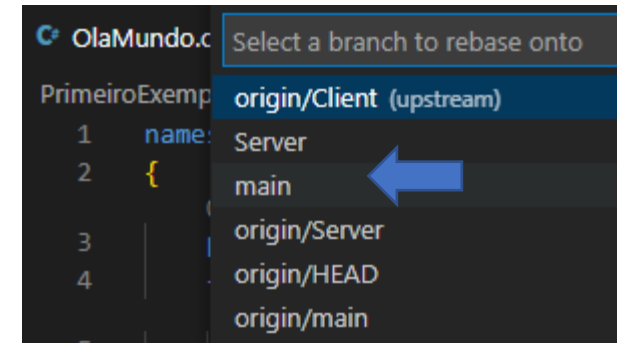
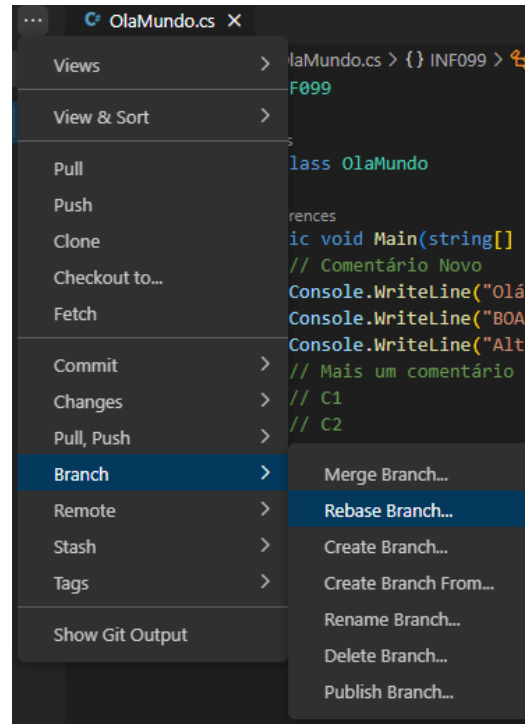
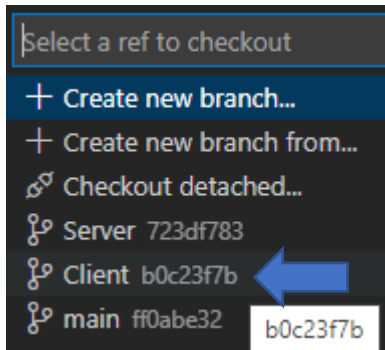
2) Conferir se está correto ...

```
Client 413 Bytes
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10            // Mais um comentário
11            // C1
12            // C2
13            // C3
14            // C8
15            // C9
16        }
17    }
18 }
```



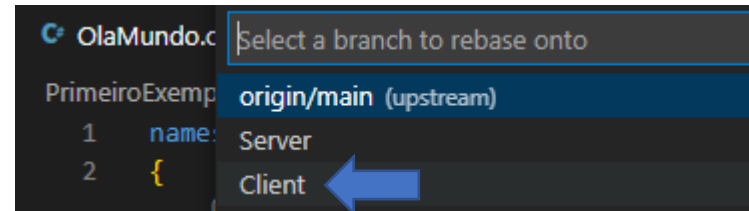
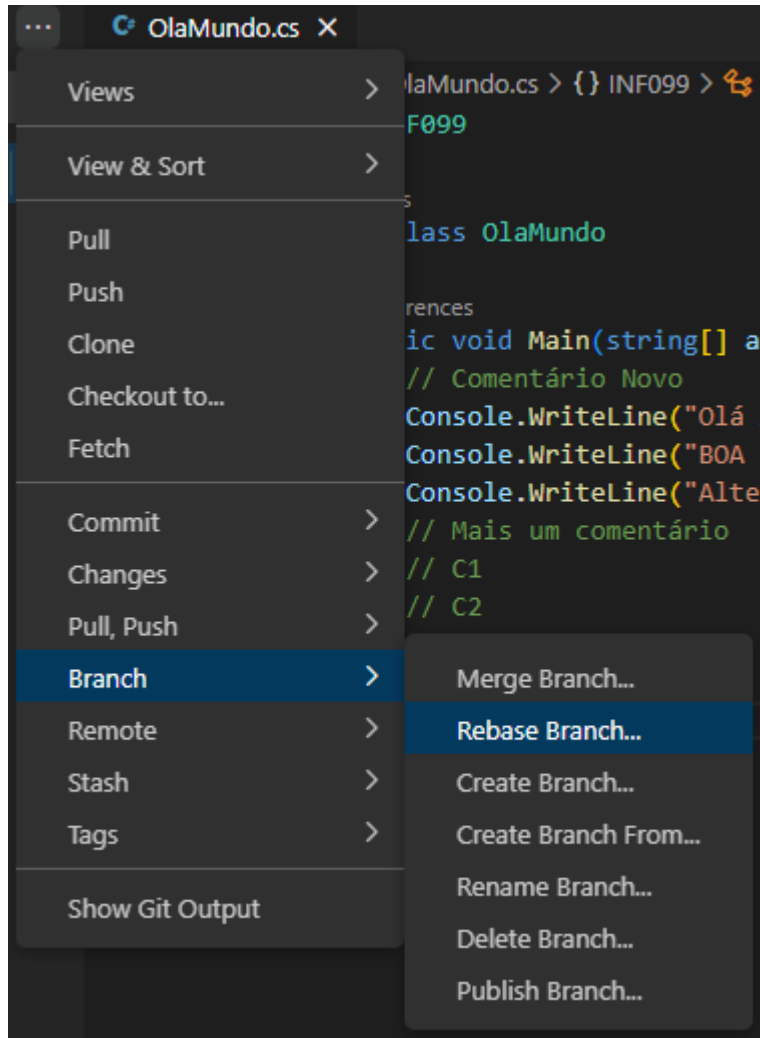


3) Fazer *rebase* da “client” na “master” (*main*)





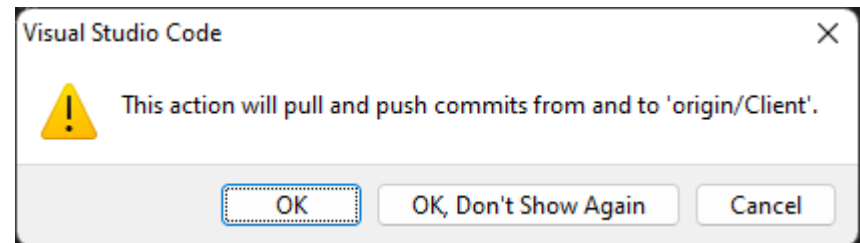
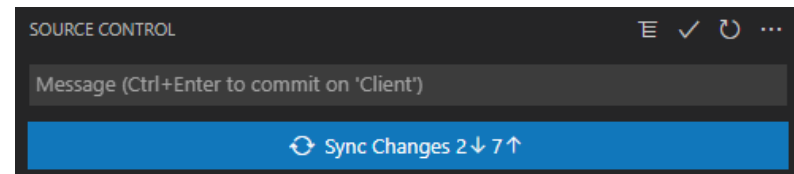
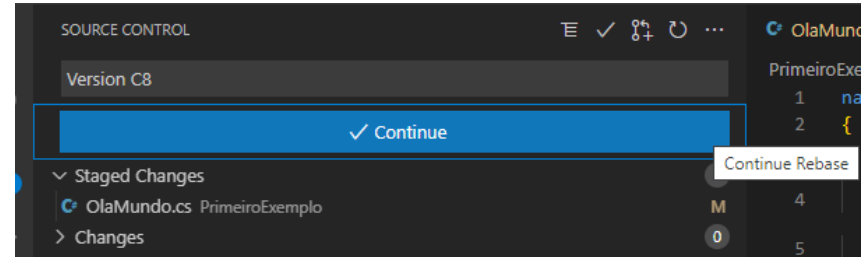
3) Fazer *rebase* da “client” na “master” (*main*)





3) Fazer *rebase* da “client” na “master” (*main*)

```
OlaMundo.cs ! X
PrimeiroExemplo > OlaMundo.cs > {} INF099 > INF099.OlaMundo > Main(string[] args)
1 namespace INF099
2 {
3     0 references
4     public class OlaMundo
5     {
6         0 references
7         static void Main(string[] args) {
8             // Comentário Novo
9             Console.WriteLine("Olá Alunos INF-099 !!!!");
10            Console.WriteLine("BOA NOITE");
11            Console.WriteLine("Alterações com VS Code");
12            // Mais um comentário
13            // C1
14            // C2
15            // C5
16            // C6
17            // C3
18            // C8
19            // C9
20        }
21    }
22 }
```



🔗 Client had recent pushes 1 minute ago



Compare & pull request



Merge pull request

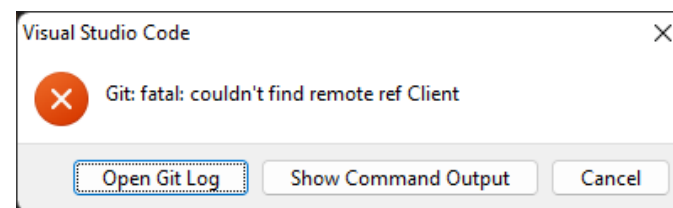
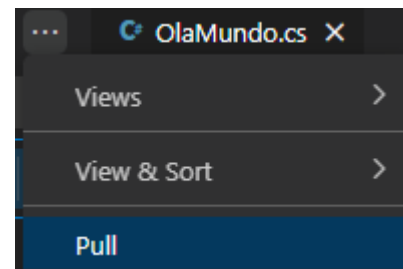


Delete branch



3) Fazer *rebase* da “client” na “master” (*main*)

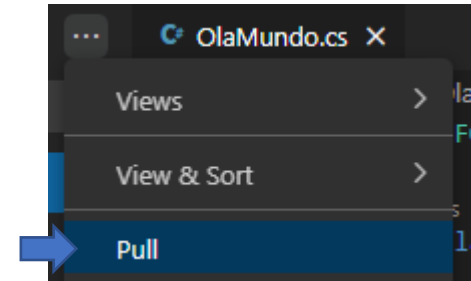
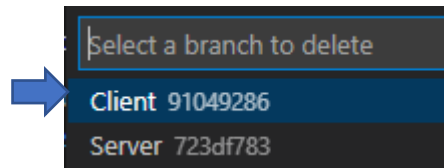
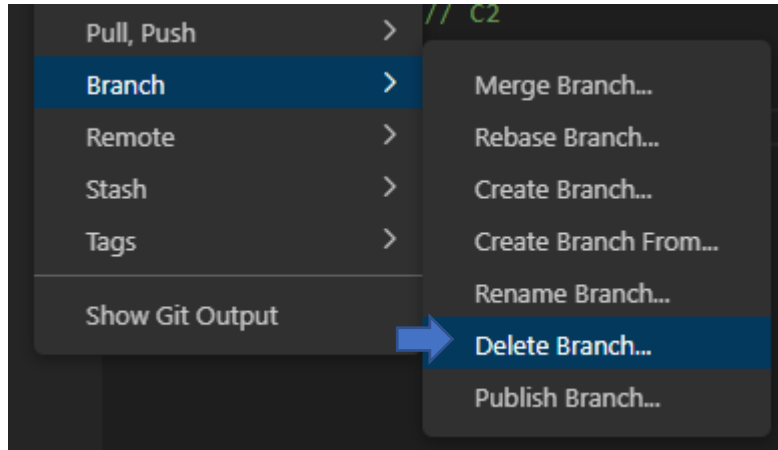
```
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10            // Mais um comentário
11            // C1
12            // C2
13            // C5
14            // C6
15            // C3
16            // C8
17            // C9
18        }
19    }
20 }
```



A branch “Client” não existe mais no GitHub (porque estou com head nela)



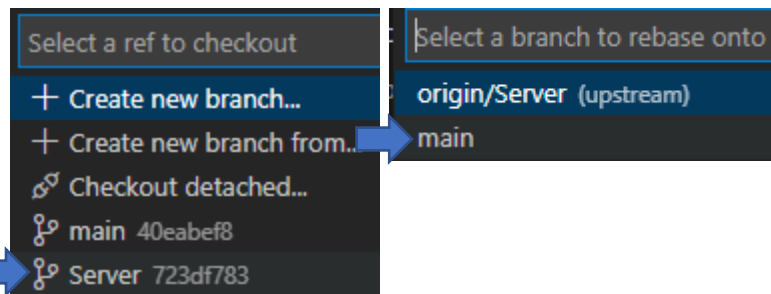
3) Fazer *rebase* da “client” na “master” (*main*)



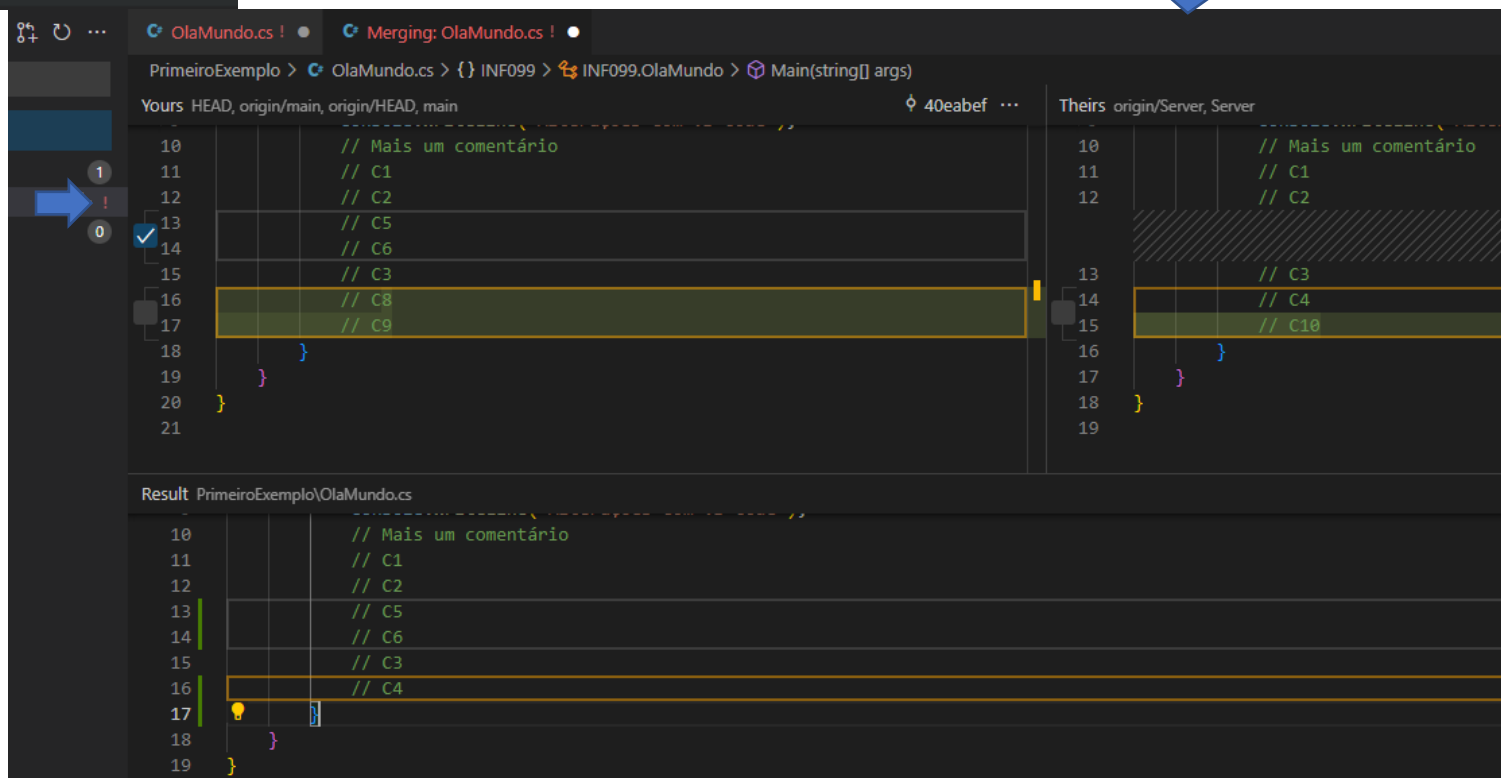
```
OlaMundo.cs X
PrimeiroExemplo > OlaMundo.cs > {} INF099 > INF099.OlaMundo > Main(string[] args)
1 namespace INF099
2 {
3     0 references
4     public class OlaMundo
5     {
6         0 references
7         static void Main(string[] args) {
8             // Comentário Novo
9             Console.WriteLine("Olá Alunos INF-099 !!!!");
10            Console.WriteLine("BOA NOITE");
11            Console.WriteLine("Alterações com VS Code");
12            // Mais um comentário
13            // C1
14            // C2
15            // C5
16            // C6
17            // C3
18            // C8
19            // C9
20        }
21    }
22 }
```



3) Fazer *rebase* da “server” na “master” (*main*) -

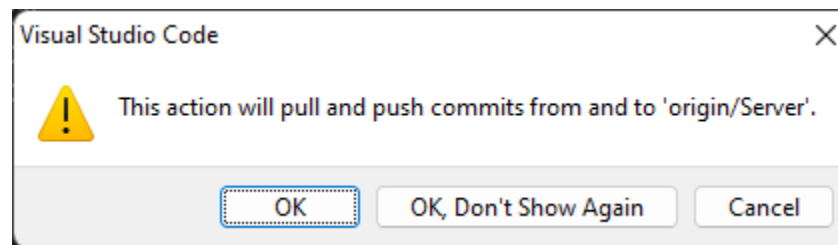
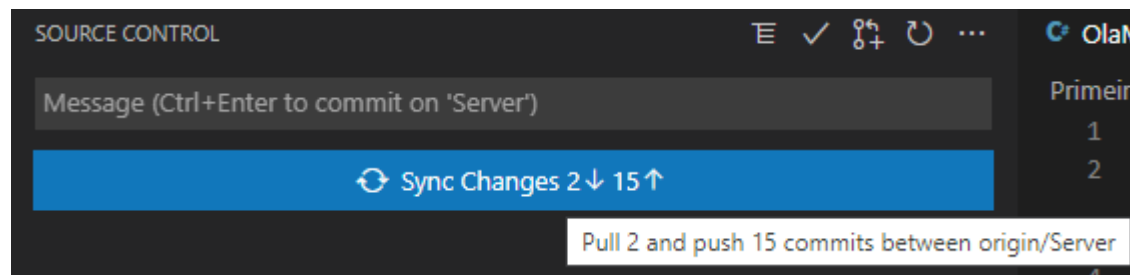
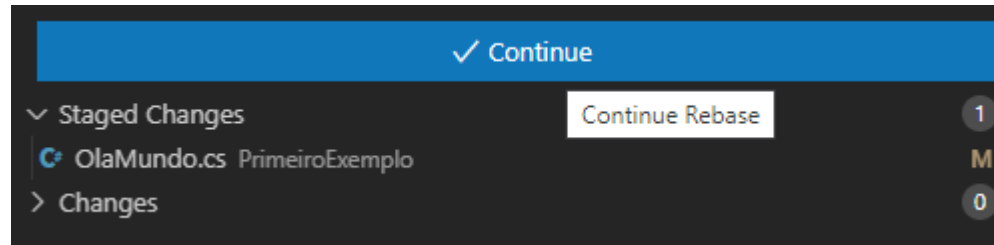


Ferramenta de
Merge do VS Code



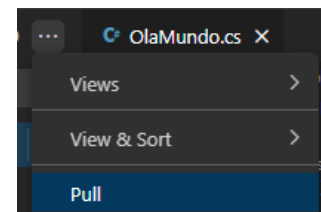


3) Fazer *rebase* da “server” na “master” (*main*) -



Merge pull request

Delete branch



Select a branch to delete

Server 48462066



4) Verificar

main INF-099 / PrimeiroExemplo / OlaMundo.cs / <> Jump to

Switch branches/tags

Find or create a branch...

Branches Tags

✓ main default

View all branches

```
1 namespace INF099
2 {
3     public class OlaMundo
4     {
5         static void Main(string[] args) {
6             // Comentário Novo
7             Console.WriteLine("Olá Alunos INF-099 !!!!");
8             Console.WriteLine("BOA NOITE");
9             Console.WriteLine("Alterações com VS Code");
10            // Mais um comentário
11            // C1
12            // C2
13            // C5
14            // C6
15            // C3
16            // C8
17            // C9
18            // C4
19            // C10
20        }
21    }
22 }
```

```
C:\VSCode\INF-099> git log --oneline --decorate --graph --all
```

```
* 9c62845 (HEAD -> main, origin/main, origin/HEAD) Merge pull request #4 from RBonacin/Server
|
| * 4846206 (origin/Server) Merge branch 'Server' of https://github.com/RBonacin/INF-099 into Server
| |
| | * 723df78 Version C10
| | * 3ed4707 (tag: C10) Versão C4
| | * 4cae2f6 Version C10
| | /
| /
* 40eabef Merge pull request #3 from RBonacin/Client
```

The background features a network of gray lines connecting various colored circles (orange, yellow, green, blue) and a dark blue horizontal band with a circuit-like pattern.

Trabalho 2



INF-0991 – Controle de Versão

Curso de Extensão Tecnologias Microsoft
Universidade Estadual de Campinas – UNICAMP

Prof. Rodrigo Bonacin
rbonacin@unicamp.br

03/setembro/2022

TRABALHO 2

1 Objetivos

Este trabalho tem como objetivo desenvolver habilidades de uso de sistema de controle de versão “Git” via ferramentas do Visual Studio Code. Este trabalho compõe a avaliação da disciplina INF0991, e deverá ser realizado em grupos de duas até quatro pessoas.



2 Atividade

As seguintes tarefas devem ser realizadas na execução desta atividade.

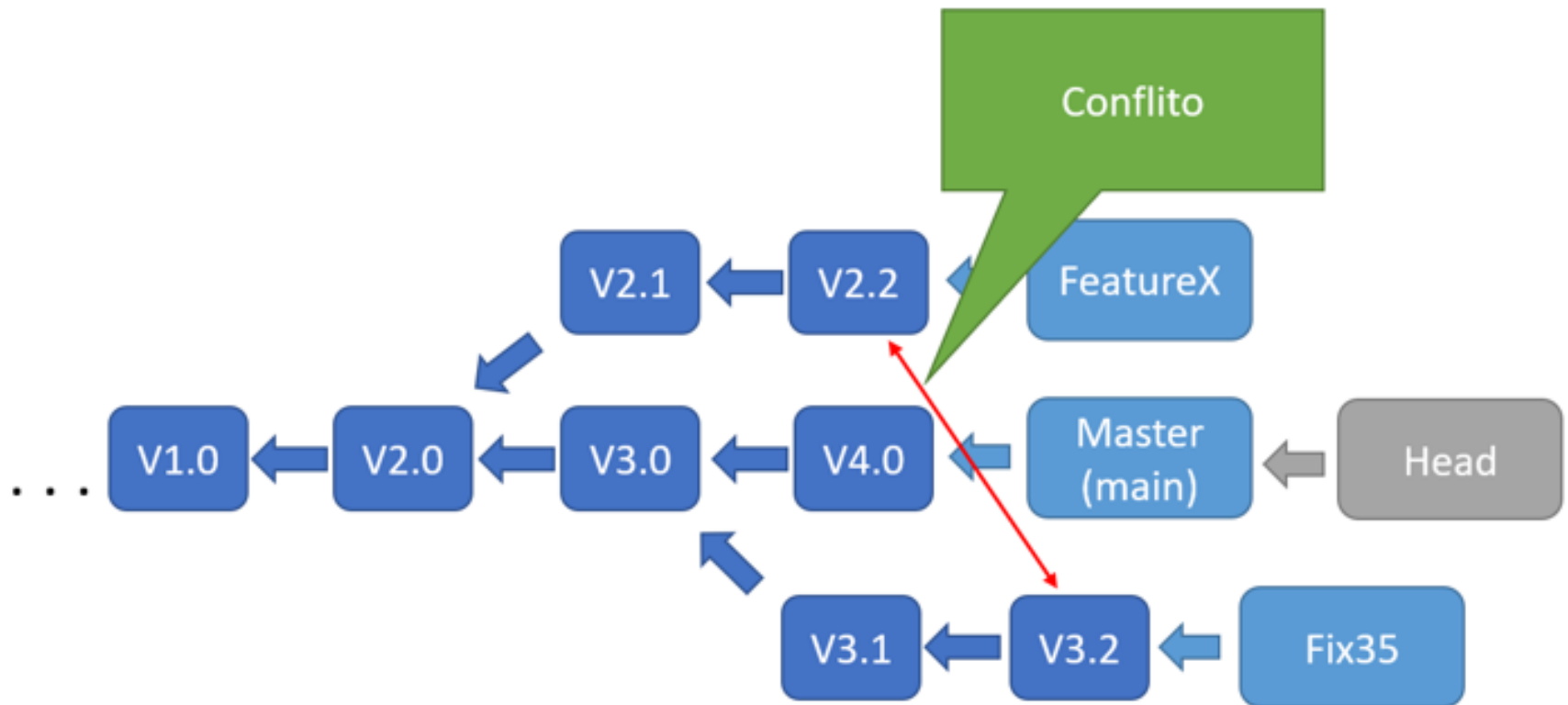
1. Inserir código C# no GitHub de um membro do grupo (preferencialmente, código gerado na disciplina INF-0990 Programação em C#).
2. Inserir alterações via Visual Studio Code por diferentes membros da equipe (em *branches* separados) de modo que tenhamos a seguinte configuração (utilizar *tags* para identificar as versões):



Trabalho 2



3. Nesta configuração deve existir (pelo menos) um conflito entre as versões V2.2 e V3.2:





4. Fazer “*rebase*” dos *branches* “Fix35” e “FeatureX” para o “Master (main)” de modo a resultar na seguinte configuração:



5. Eliminar eventuais conflitos.
6. Atualizar o GitHub (*push*).
7. Verificar alterações no GitHub.



3 Entrega

Os seguintes itens devem ser entregues:

1. Link com repositório GitHub com históricos de alterações, desde a inclusão dos arquivos até a *rebase* e solução de conflitos.
2. Relatório com cada tela (*print*) da execução no repositório local com todos os comandos executados na interface do Visual Studio Code.
3. Print com log de alterações/*commits* no repositório local (*git log --oneline --decorate --graph --all*) a cada *commit*.
4. Ou seja, cada tarefa supracitada deve estar documentada de modo a ilustrar sua execução.

Deverá ser entregue um arquivo por grupo, constando os respectivos nomes e números de matrículas. As entregas deverão ser realizadas pelo Moodle.

DATA FINAL DE ENTREGA: 09/09/2022