



Introduction to Git and GitHub

MARCH 2024

Contents

- 01** Git
- 02** Git Installation
- 03** GitHub
- 04** Git vs GitHub
- 05** GitHub Setup
- 06** Git – Setting your credentials
- 07** Git – Clone/Download a Repository
- 08** Git – Main Commands
- 09** Fix Git/GitHub Integration
- 10** Homework

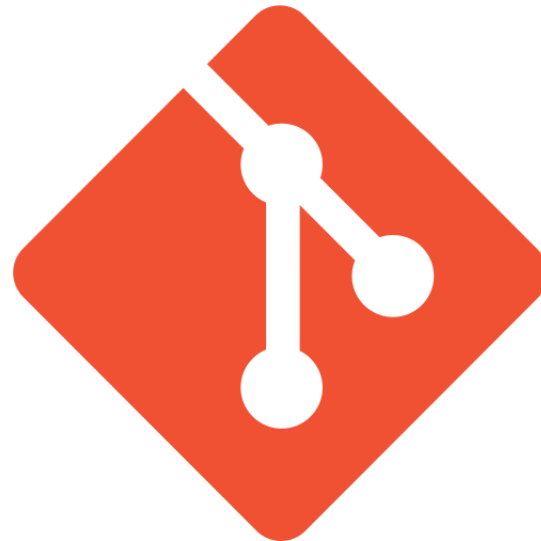
Git (Version Control System - VCS)

Imagine you and your friends are working together on a big group project, like creating a video game or writing a story.

Now, let's say you're all working on different parts of the project at the same time.

How do you keep track of everyone's changes and make sure you're all working on the same version?

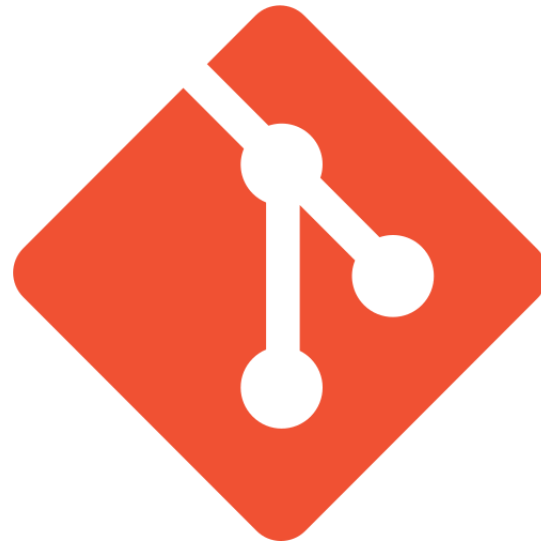
That's where Git comes in!



git

Git (Version Control System - VCS)

Git is like a magical tool that helps you and your friends work together on projects without stepping on each other's toes



git

Git (Installation)

1. Access <https://git-scm.com/downloads> and download accordingly to your Operational System (Windows, Linux, Mac)
2. Run the file downloaded
3. Click on the NEXT buttons till find the INSTALL button
4. Click on the INSTALL button and wait
5. After the installation is finished, check if is installed:
 - Open a Command line (terminal) application
 - Type: `git -version` and then hit the enter button

```
C:\Users\Eric_Dantas>git --version  
git version 2.37.2.windows.2
```

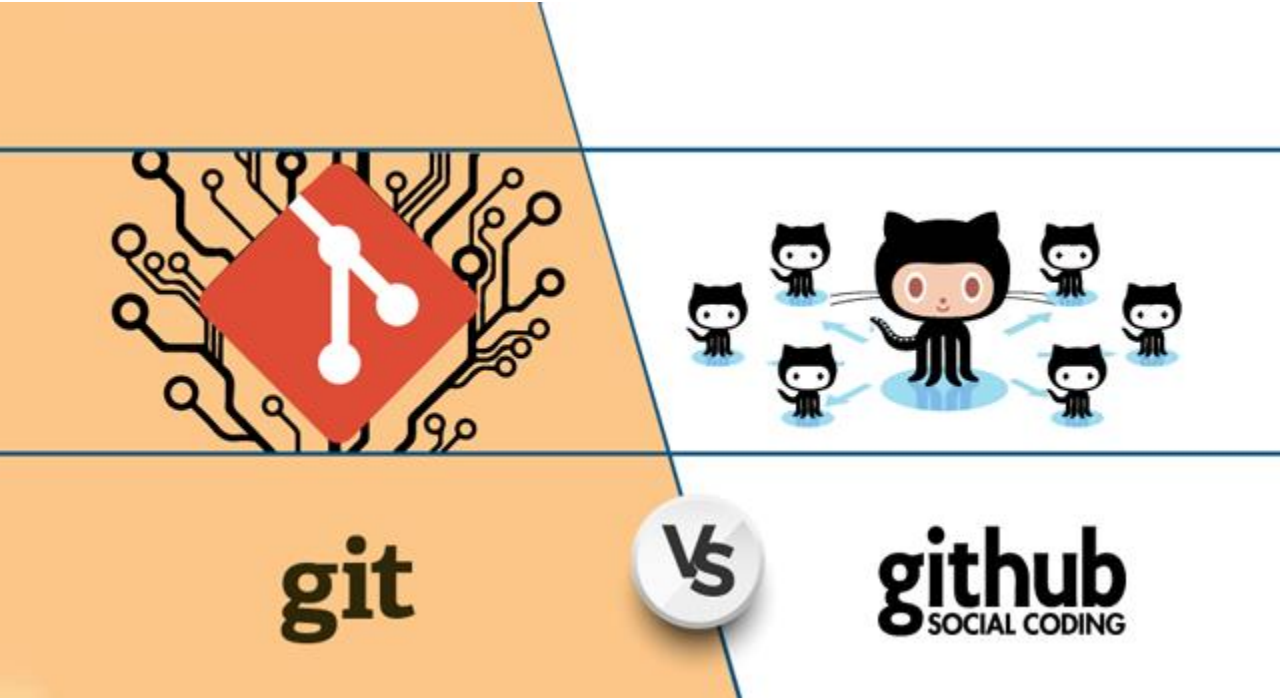
GitHub (Hosting Platform)

Imagine you've written a fantastic story with your friends using Git, but now you want to share it with the world or maybe even get feedback from other people.

That's where GitHub comes in!

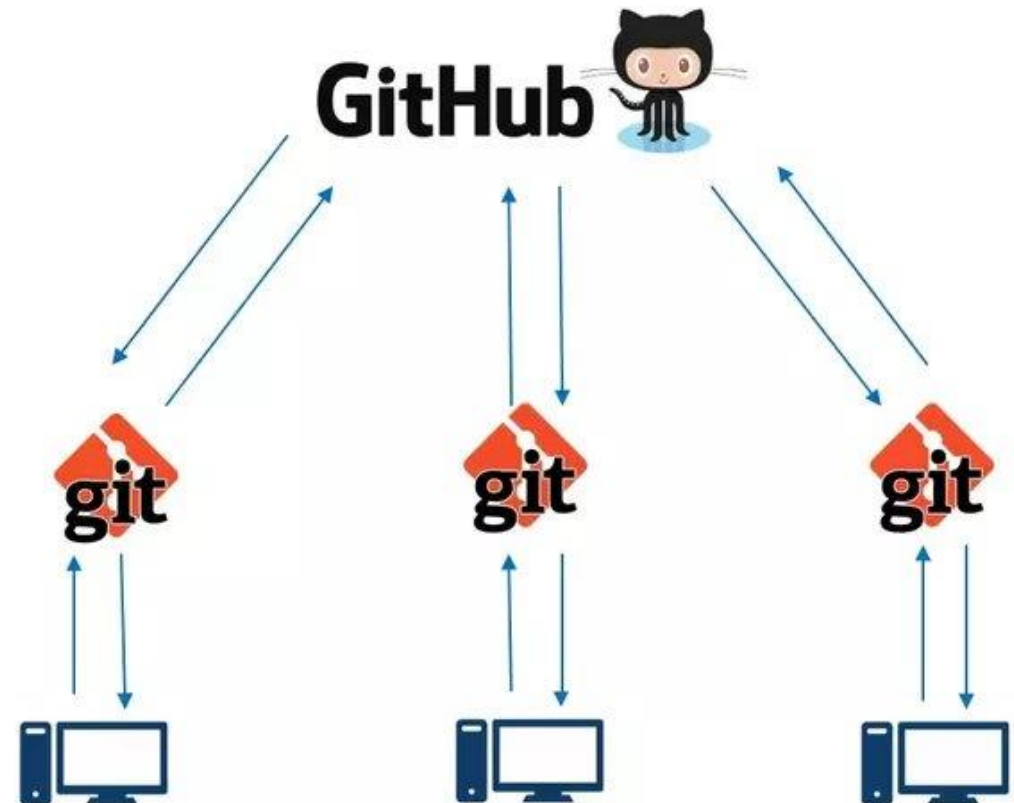
GitHub is like a giant library where you can store your projects and collaborate with people from all over the world.





Git is the tool that helps you manage changes in your project locally on your computer.

GitHub is the online platform where you can store your Git projects, collaborate with others, and showcase your work to the world.



GitHub (Setup – Create an Account)

Note: Before work with GitHub, make sure Git is already installed

1. Access the link <https://github.com/> and create an account by clicking on the Sign Up button (top right on the screen)
2. Type your email and click on the Continue button
3. Create a password and click on the Continue button
4. Type a username and click on the Continue button
5. Click on the Continue button to complete
6. Verify your account by clicking on the Verify button. After that, you need to solve the puzzle. The puzzle can vary in methods to solve
7. GitHub will send an 8 numbers code to your email. Copy it and type inside the squares shown by GitHub. If the code is correct, you will be redirected to your GitHub space. Everything is done!

Git (Setting your credentials)

1. From a terminal (command line), set your username:

```
git config --global user.name "Segun Ajibola"
```

2. Confirm that your username is set:

```
git config --global user.name
```

3. Set your email:

```
git config --global user.email "youremail@gmail.com"
```

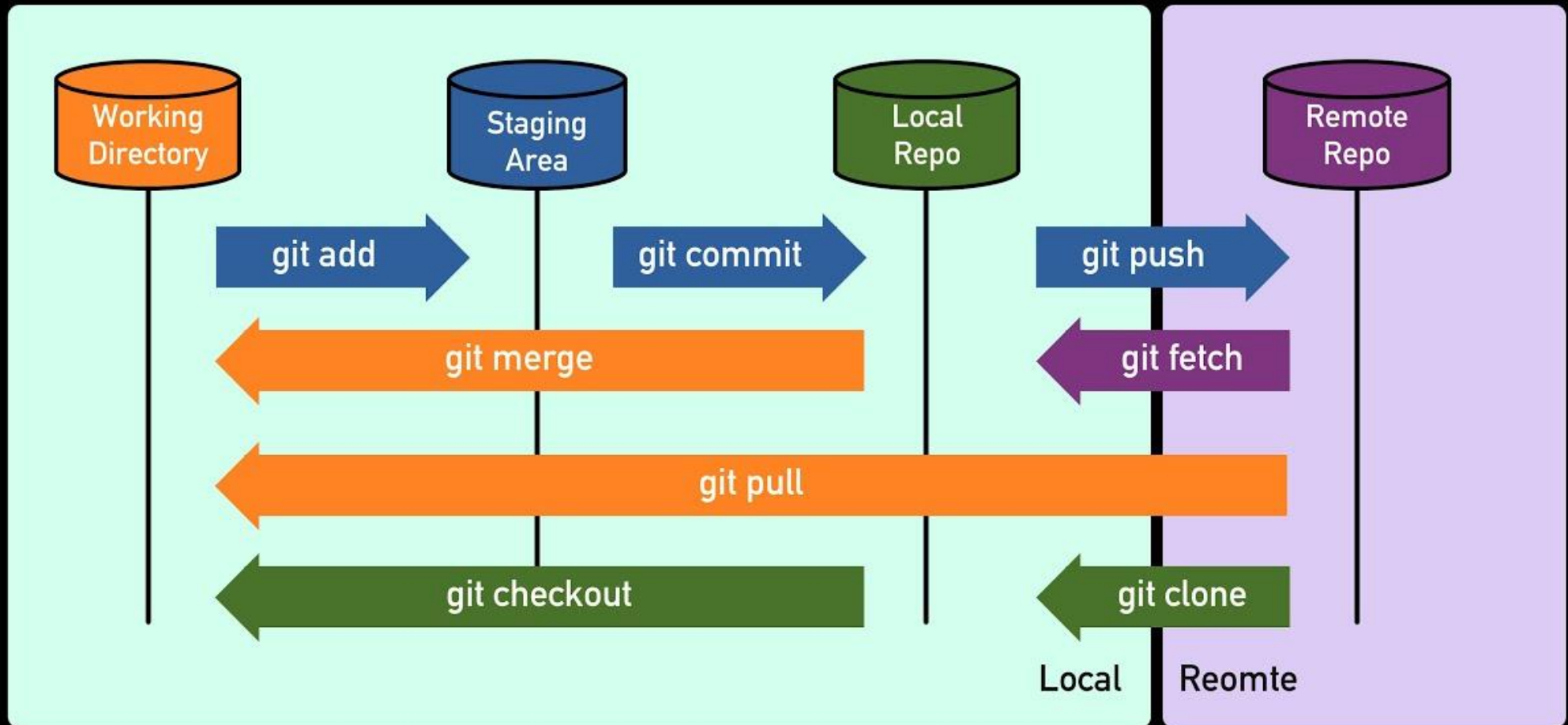
4. You will be asked to authenticate your GitHub account, so just sign in with the same email to confirm

Git (Clone/Download a Repository)

1. Open a terminal from a folder where you want to have your project
 - Right-click mouse
 - Click on Open in Terminal
 - Type the next commands
2. `git clone` is like making a copy of a project from GitHub onto your computer
 - `git clone https://github.com/ericrommel/ekids.git` (in general)
 - `git clone git@github.com:ericrommel/ekids.git` (only if you have defined a public SSH key and also only in your own machine)
3. Go to the folder where you cloned the project and have fun!

Git (Main commands)

How Git Actually Works



Git (Main commands)

- Change to the main branch
 - `git checkout main`
- Pull the up-to-date files from Github
 - `git pull`
- Change to your branch
 - `git checkout my-branch-name-here`
- Merge the changes from remote to your local
 - `git merge main`
- See what files were changed
 - `git status`
- Add files that you want to save on Git
 - `git add file-name` or `git add .`
- Commit your changes
 - `git commit -m "A nice message to understand what was done"`
- Push your changes
 - `git push`

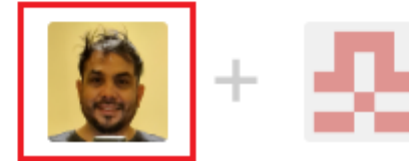
Fix Git/GitHub Integration

- **To allow you to push changes** to the ekids' repository, **you need to accept** a collaboration invitation **sent to your email** connected to your GitHub account
- **Before clicking on anything**, please confirm that the invitation is a valid one (check the highlighted squares)
- Click on **“accept”** or **“View invitation”**
- You need to be logged in to your GitHub account to accept it

ericrommel invited you to ericrommel/ekids Σ Inbox x

Eric Dantas <noreply@github.com>
to me ▼

GitHub



@ericrommel has invited you to collaborate on the **ericrommel/ekids** repository

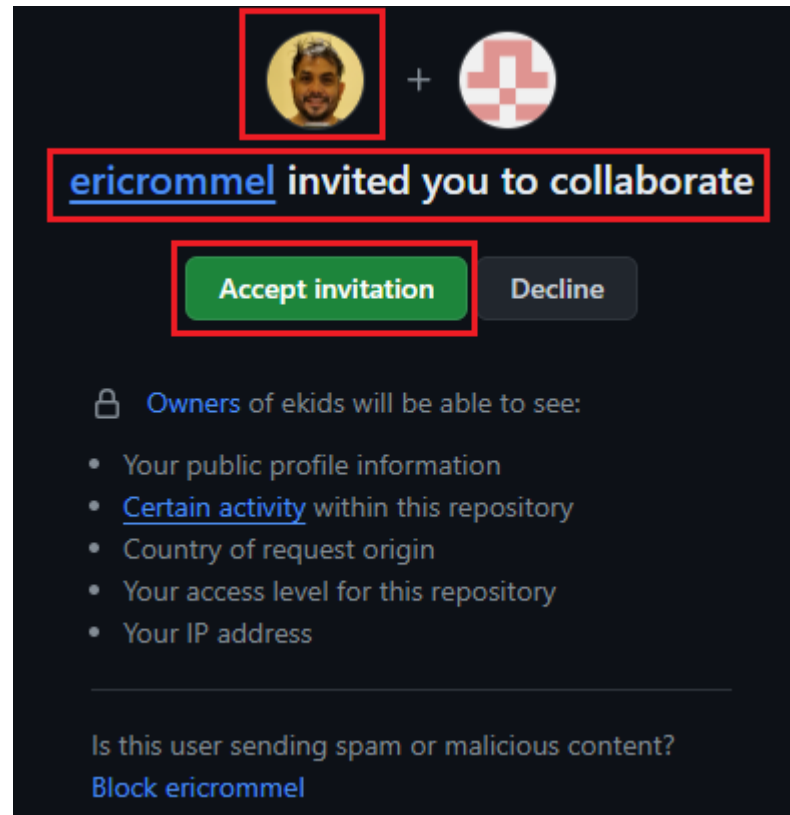
You can **accept** or [decline](#) this invitation. You can also head over to <https://github.com/ericrommel/ekids> to check out the repository or visit [@ericrommel](#) to learn a bit more about them.

This invitation will expire in 7 days.

View invitation

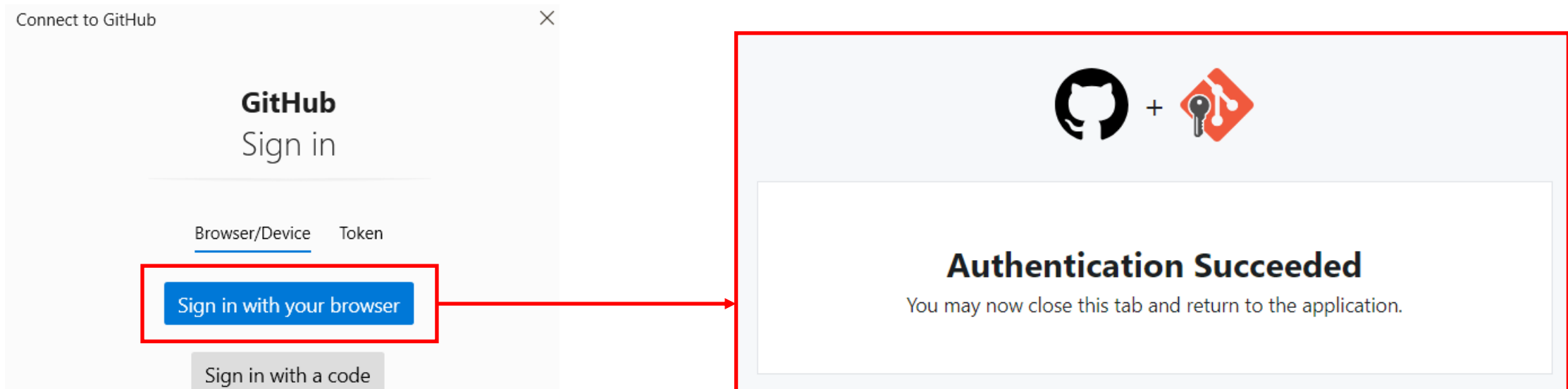
Fix Git/GitHub Integration

- GitHub will ask you to accept the invitation. Just click on the green button labeled “Accept invitation”



Fix Git/GitHub Integration

- At EPAM, as you are using different laptops each time, clone the repository using <https://github.com/ericrommel/ekids.git>
- During the push, you might be prompted to log in using the browser access



Fix Git/GitHub Integration

- If the below message is appearing for you:

```
PS C:\Users\Eric_Dantas\projects\epam\ekids> git push
git@github.com: Permission denied (publickey).
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.
```

See that the issue is with the public key. Probably you don't have it saved in your machine or in your GitHub account

- You need to follow the next slide steps (use Git Bash terminal)

NOTE: For safety reasons, only do it from your OWN device

Fix Git/GitHub Integration

```
EPAM+Eric_Dantas@EPESMALW01EA MINGW64 ~/caio-rommel/ekids (caio)
$ eval "$(ssh-agent -s)" 1
Agent pid 1605

EPAM+Eric_Dantas@EPESMALW01EA MINGW64 ~/caio-rommel/ekids (caio)
$ ssh-keygen -t ed25519 -C "ericrommel@gmail.com" 2
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/Eric_Dantas/.ssh/id_ed25519): 3
Enter passphrase (empty for no passphrase): 4
Enter same passphrase again: 5
Your identification has been saved in /c/Users/Eric_Dantas/.ssh/id_ed25519
Your public key has been saved in /c/Users/Eric_Dantas/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:pZra19l/iAXiNgZ4WHSIJJ2E+g301Ra97iia9Xoszw4 ericrommel@gmail.com
The key's randomart image is:
+--[ED25519 256]--+
|      . = 0 + . 0 0 0      |
|      0 . + + . 0 .      |
|      0 . = . . .      |
|      . . + 0 0 . 0      |
|      . 0 . So 0 .      |
|      . 0 = . .      |
|      0 E = * 0 .      |
|      0 0 ++ * + . .      |
|      . + 0 . B = . . .      |
+-----[SHA256]-----+

EPAM+Eric_Dantas@EPESMALW01EA MINGW64 ~/caio-rommel/ekids (caio)
$ ssh-add c:/Users/Eric_Dantas/.ssh/id_ed25519 6
Enter passphrase for c:/Users/Eric_Dantas/.ssh/id_ed25519: 7
Identity added: c:/Users/Eric_Dantas/.ssh/id_ed25519 (ericrommel@gmail.com)

EPAM+Eric_Dantas@EPESMALW01EA MINGW64 ~/caio-rommel/ekids (caio)
$ clip < ~/.ssh/id_ed25519.pub 8
EPAM+Eric_Dantas@EPESMALW01EA MINGW64 ~/caio-rommel/ekids (caio)
$
```

1. Turn on *ssh-agent*
2. Generate a *new SSH key* on your local machine (*replace* to use your email)
3. Just press *Enter*
4. Type a passphrase (*don't forget it*)
5. Re-type the passphrase
6. Add your SSH key to the *ssh-agent*
7. Type your passphrase defined
8. Copy the *SSH public key* to your clipboard (memory)

Fix Git/GitHub Integration

- Now, go to your GitHub account and save your SSH KEY copied before

The screenshot shows the GitHub account settings page for Eric Dantas (ericrommel). The page is divided into a left sidebar with navigation links and a main content area with settings sections.

Left Sidebar (Navigation):

- Set status
- Your profile
- Add account
- Your repositories
- Your projects
- Your Copilot
- Your organizations
- Your enterprises
- Your stars
- Your sponsors
- Your gists
- Upgrade
- Try Enterprise
- Feature preview
- Settings** (highlighted with a red box and number 1)

Account Information:

- Eric Dantas (ericrommel)
- Your personal account [Switch settings context](#)

Settings Categories:

- Public profile
- Account
- Appearance
- Accessibility
- Notifications
- Access
 - Billing and plans
 - Emails
 - Password and authentication
 - Sessions
 - SSH and GPG keys** (highlighted with a red box and number 2)
 - Organizations
 - Enterprises
 - Moderation

Main Content Area:

- SSH keys**
 - There are no SSH keys associated with your account.
 - Check out our guide to [connecting to GitHub using SSH keys](#) or troubleshoot [common SSH problems](#).
 - New SSH key** (highlighted with a red box and number 3)
- GPG keys**
 - There are no GPG keys associated with your account.
 - Learn how to [generate a GPG key and add it to your account](#).
 - New GPG key**
- Vigilant mode**
 - ☐ **Flag unsigned commits as unverified**
 - This will include any commit attributed to your account but not signed with your GPG or S/MIME key. Note that this will include your existing unsigned commits.
 - [Learn about vigilant mode.](#)

Fix Git/GitHub Integration

- Now, go to your GitHub account and save your SSH KEY copied before

Add new SSH Key

Title

Caio's laptop

Add any title you want for your SSH KEY

Key type

Authentication Key

Key

```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIEligHGy9NyH6G2SROHx6kqn99hYKj/yRVxTvhvlpqJ ericrommel@gmail.com
```

Ctrl+V (paste) your SSH KEY saved from Git Bash

Add SSH key

After clicking on that button, your SSH KEY is saved. You can go back and push your changes

Homework

Note: Add a screenshot for each completed task below on your Google Classroom

- Install Git
- Create a GitHub account
- Clone the repository from <https://github.com/ericrommel/ekids> (git clone)
- Make any change in (e.g.: any code homework available)
- Add your changes (git add)
- Commit your changes (git commit)
- Push your changes (git push)

Docs and Books

- Git Doc: <https://git-scm.com/doc>
- GitHub Doc: <https://docs.github.com/en>
- Book: [Beginning Git and GitHub: A Comprehensive Guide to Version Control, Project Management, and Teamwork for the New Developer](#)
- Book: [Version Control with Git and GitHub](#)

Thank you!
