



I STRAŽIVAČKA STANICA PETNICA

Uvod u GIT

Nina Jakovljević

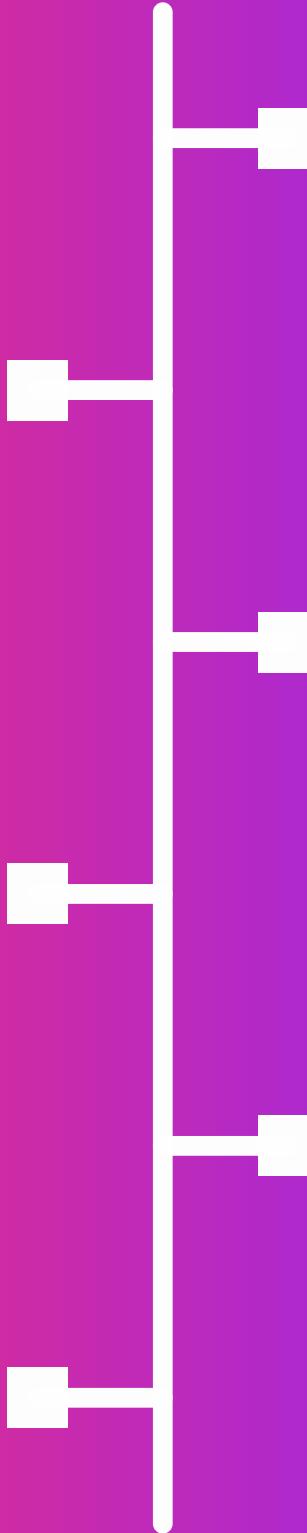
Zajedničko predavanje na seminaru fizike i elektronike



IS Petnica, jun 2025.

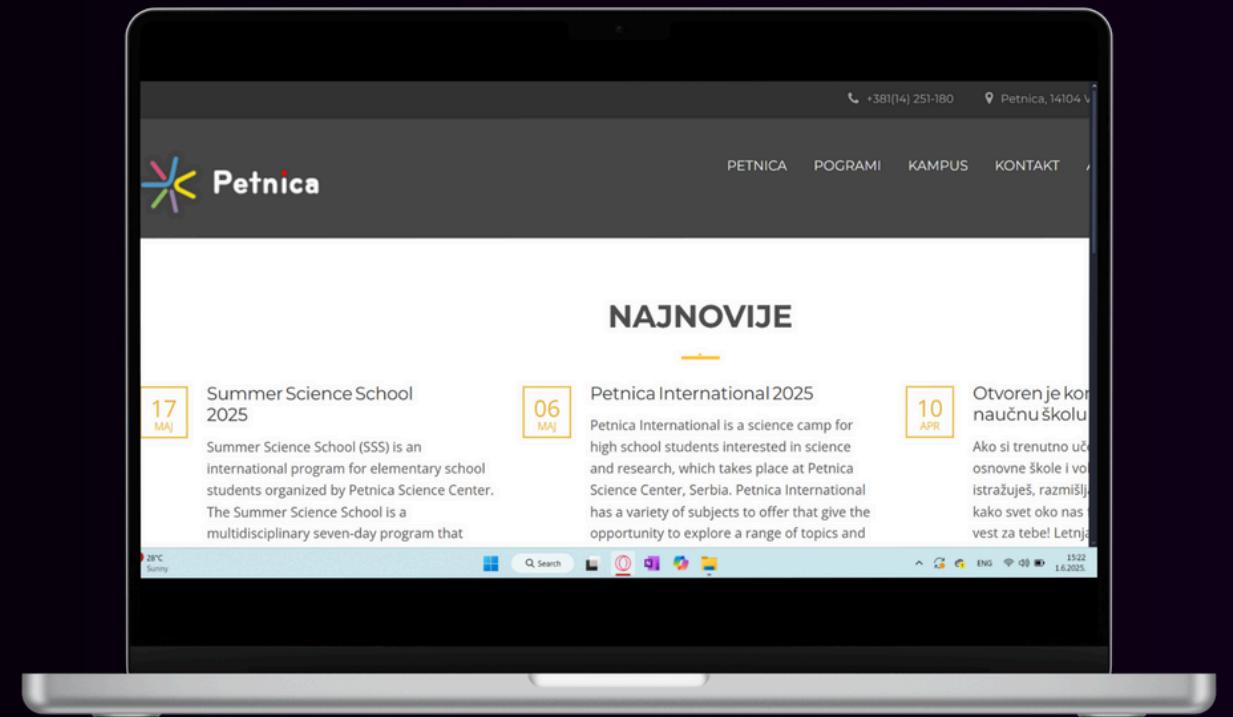


Overview



- Introduction to GIT
- Installing GIT
- Initializing GIT repository
- GIT branches
- Working with Remote Repositories
- Cloning, Pushing, Pulling & Fetching
- Pull requests
- Merge Conflicts
- Forks
- Cherry Picking
- Reseting & Reverting

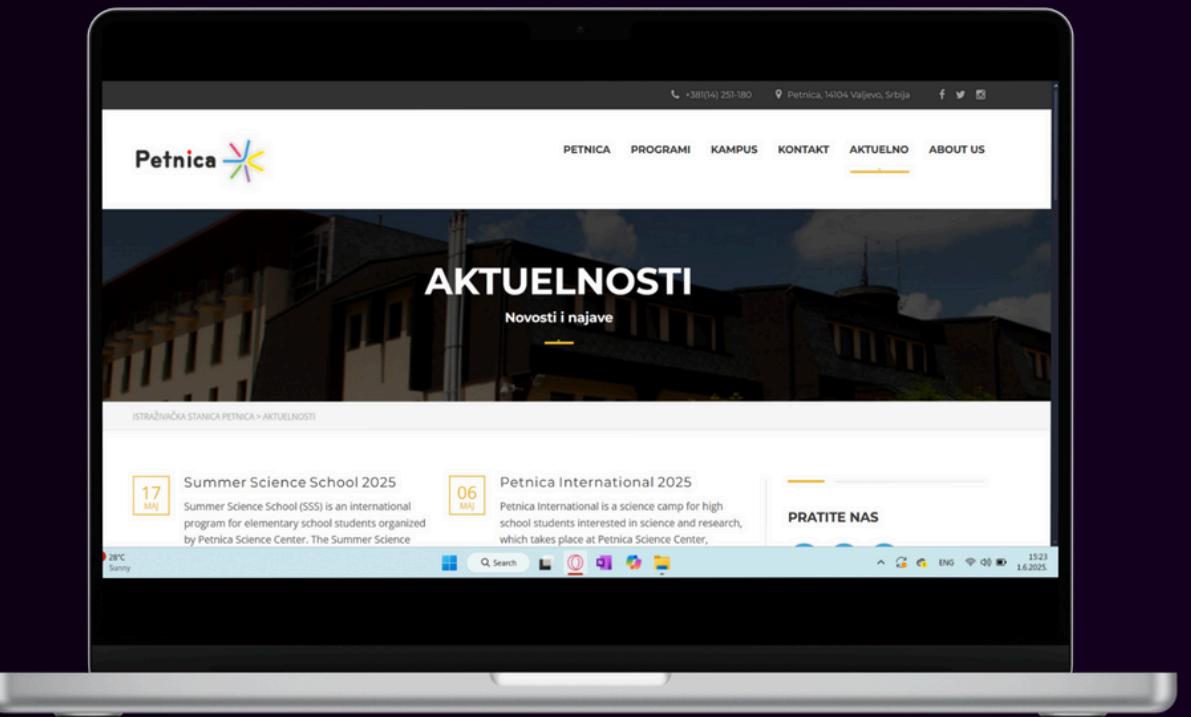
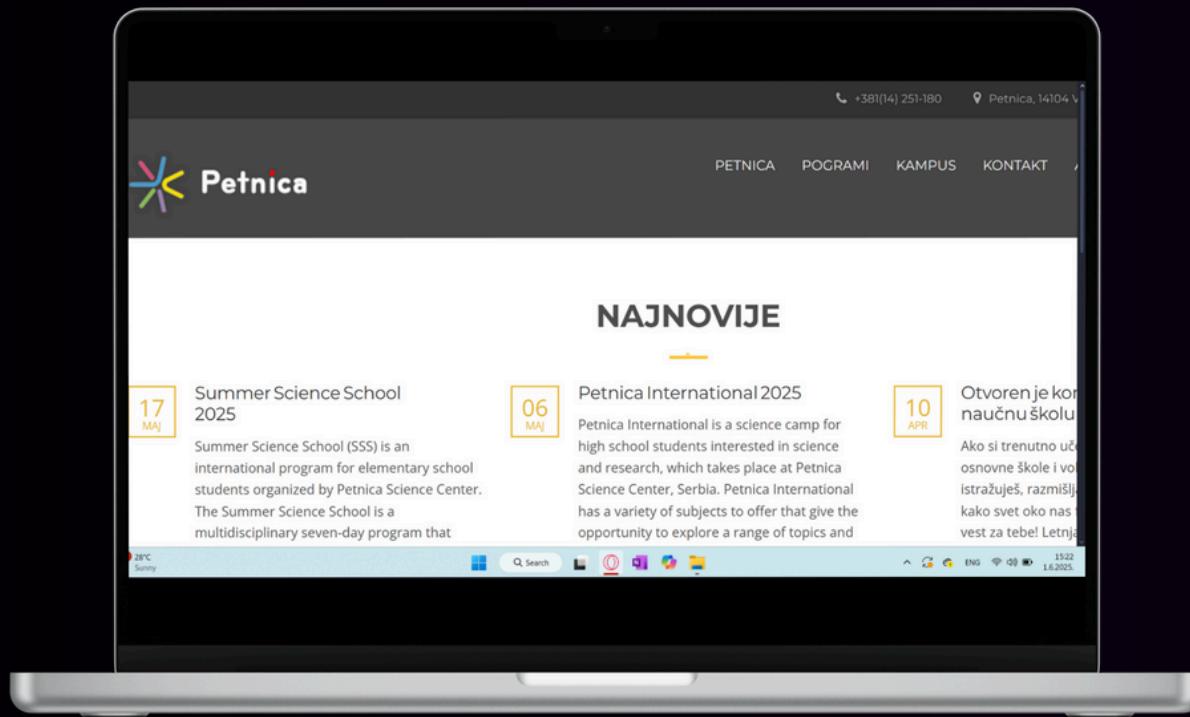
www.mywebsite.com



IS Petnica, jun 2025.

www.mywebsite.com

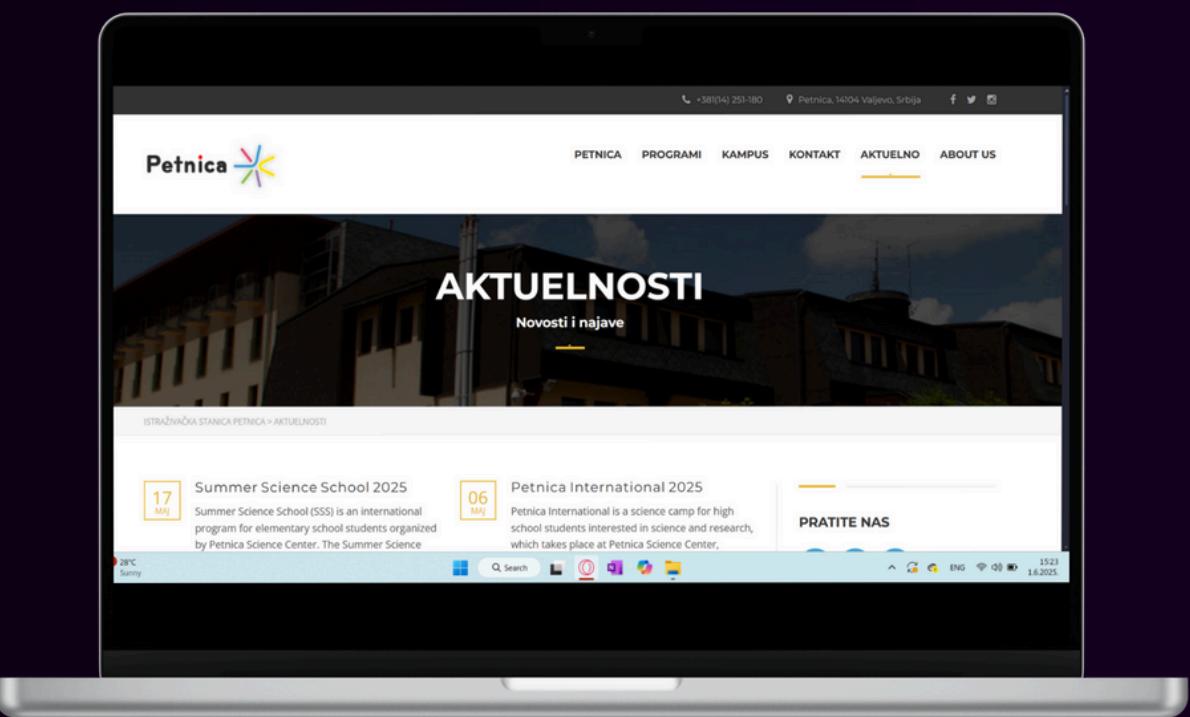
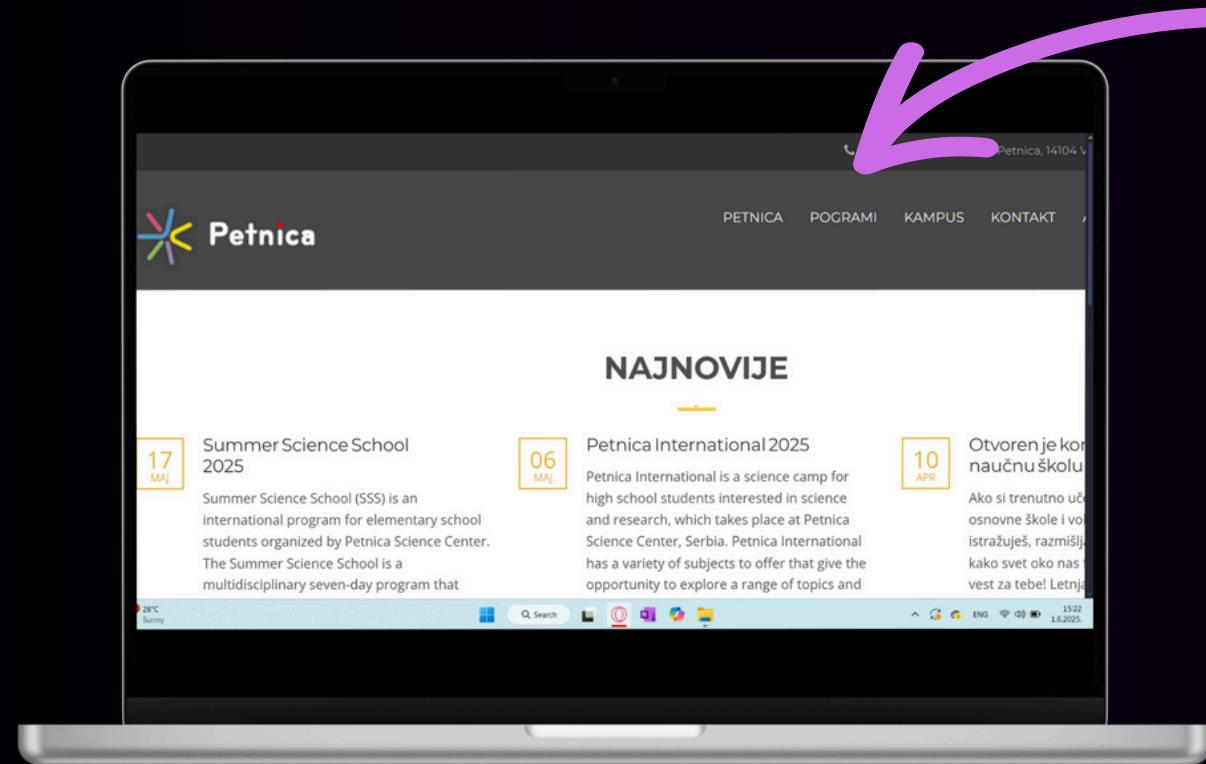
local



IS Petnica, jun 2025.

www.mywebsite.com

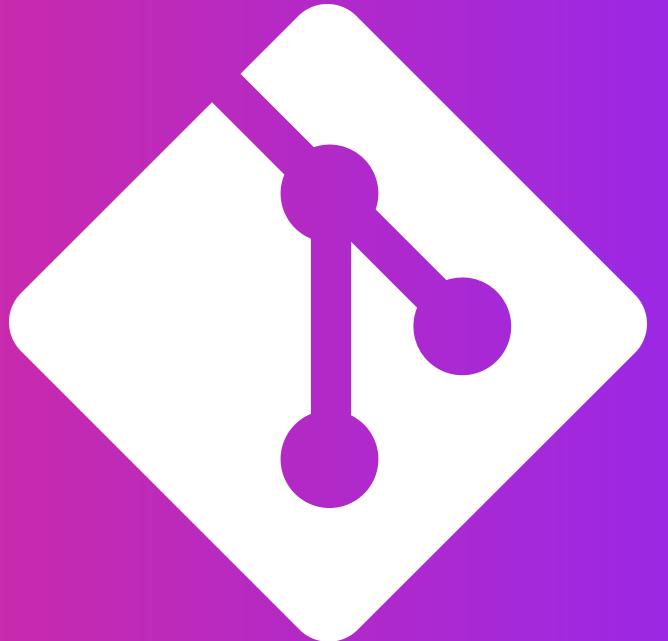
local



IS Petnica, jun 2025.



Introduction



- Content Tracker
- Distributed Version Control System

GIT can help you out!

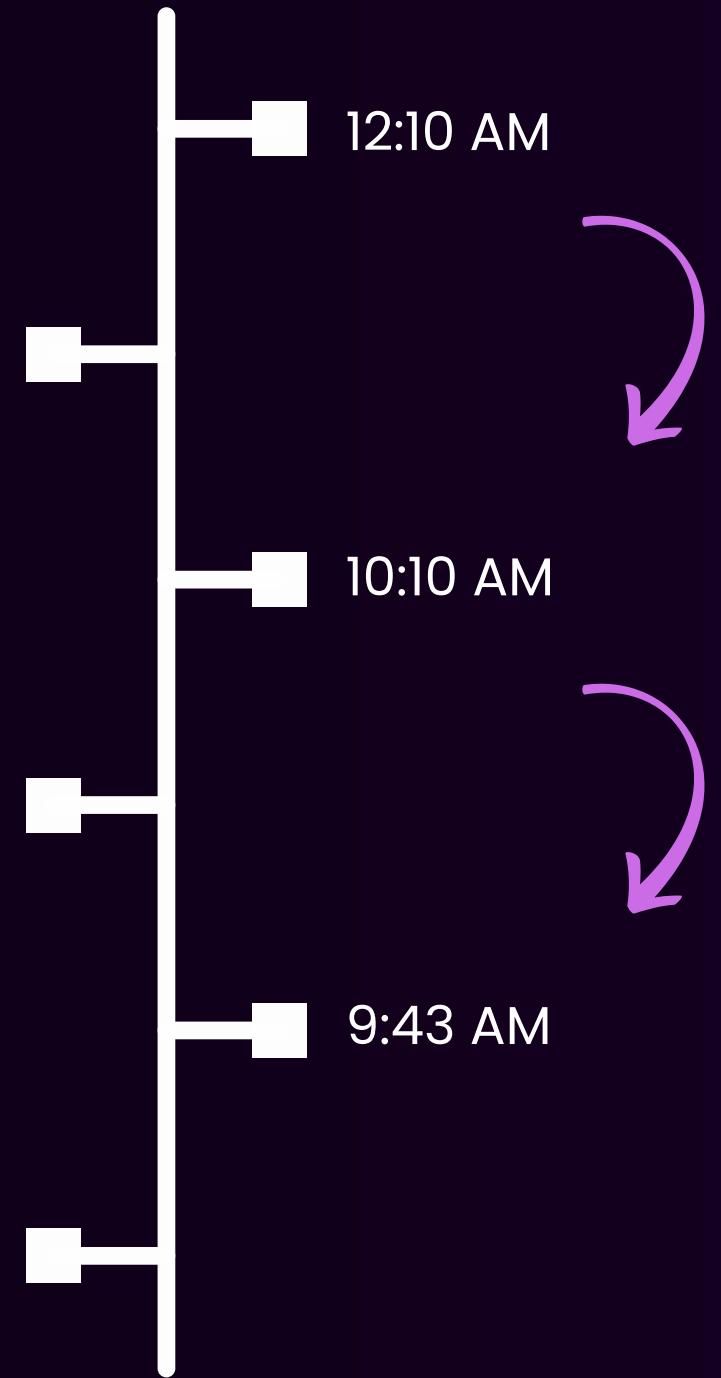
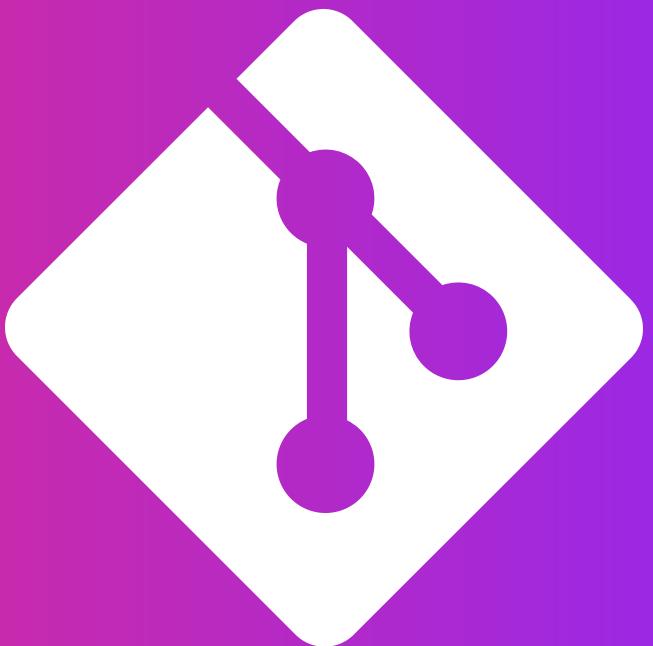




Introduction

- Content Tracker
- Distributed **Version Control System**

GIT can help you out!



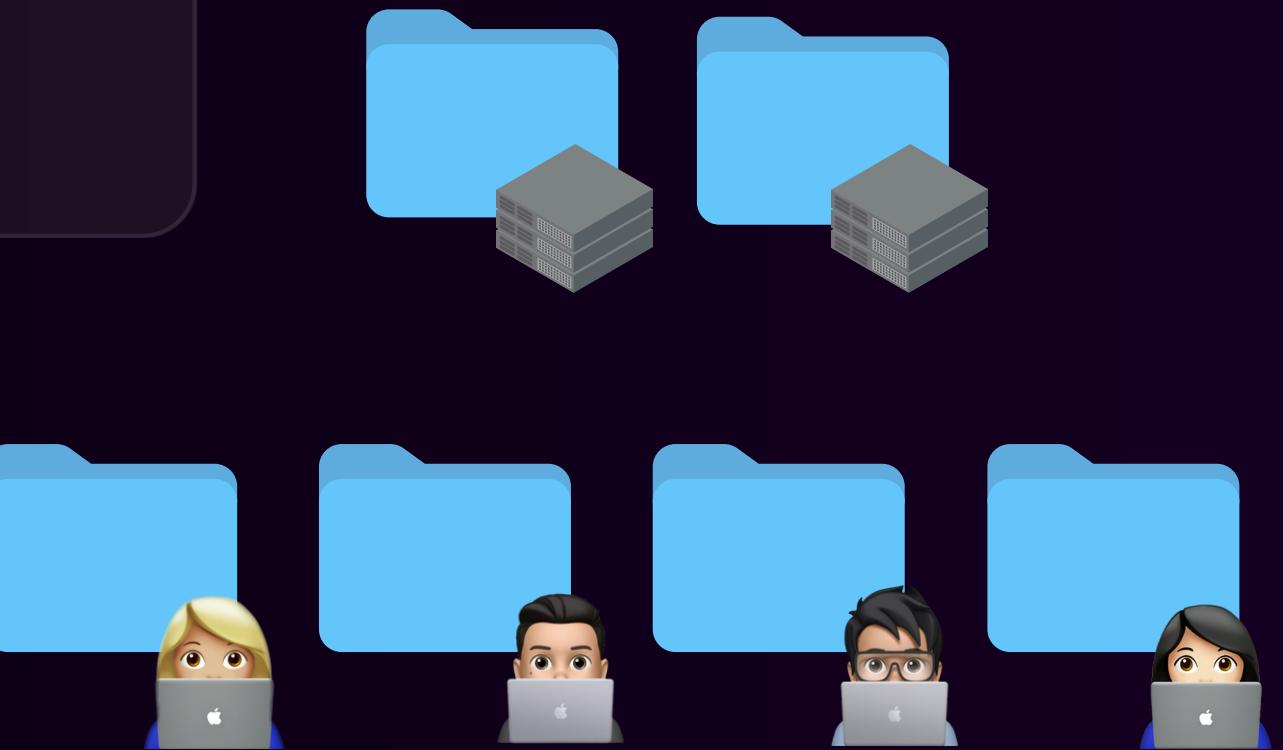


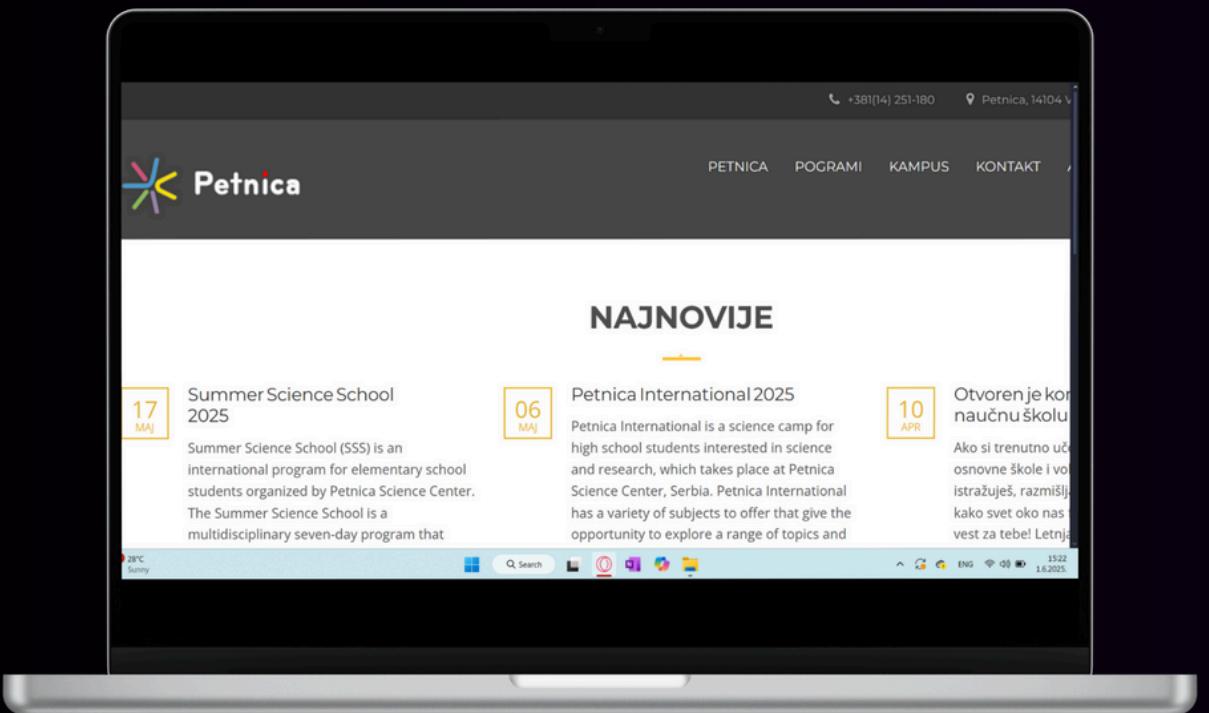
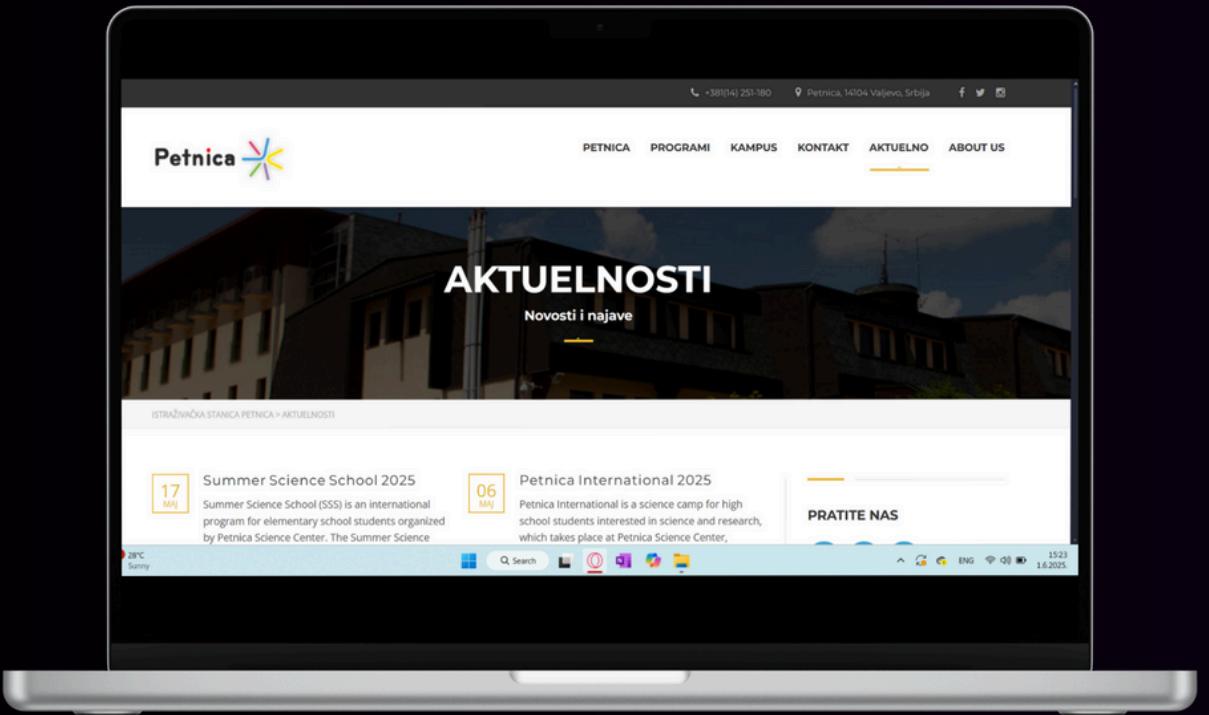
Introduction



- Content Tracker
- **Distributed** Version Control System

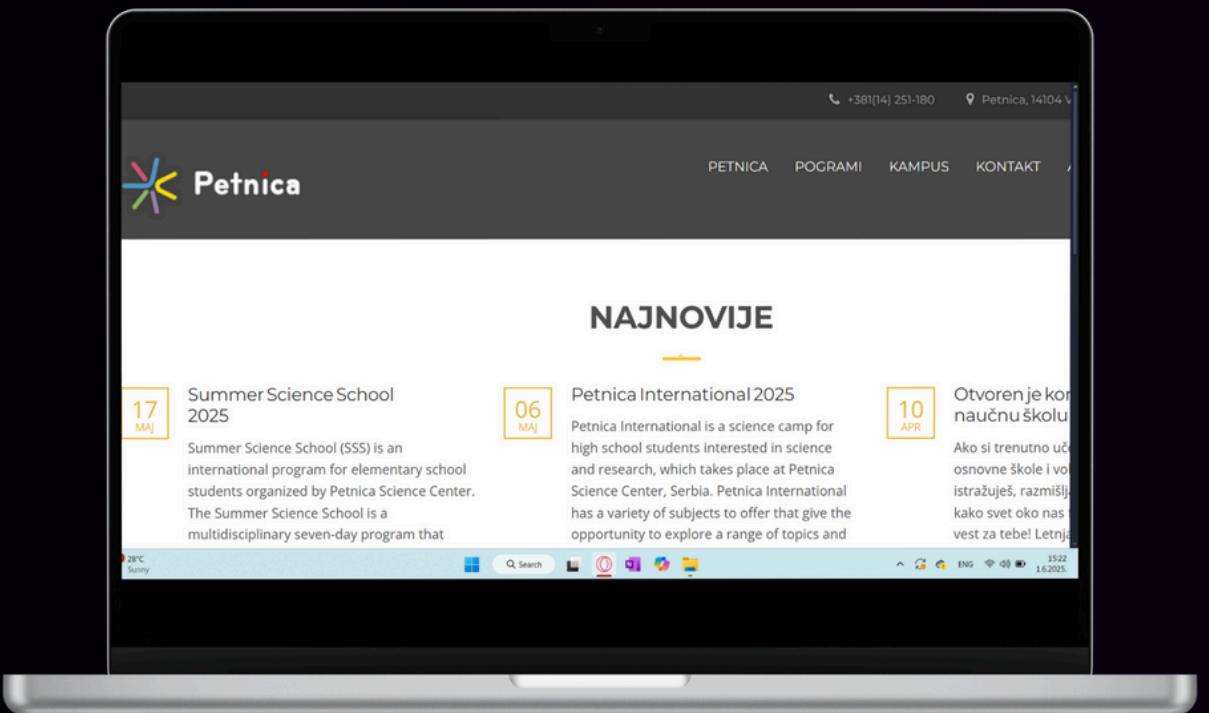
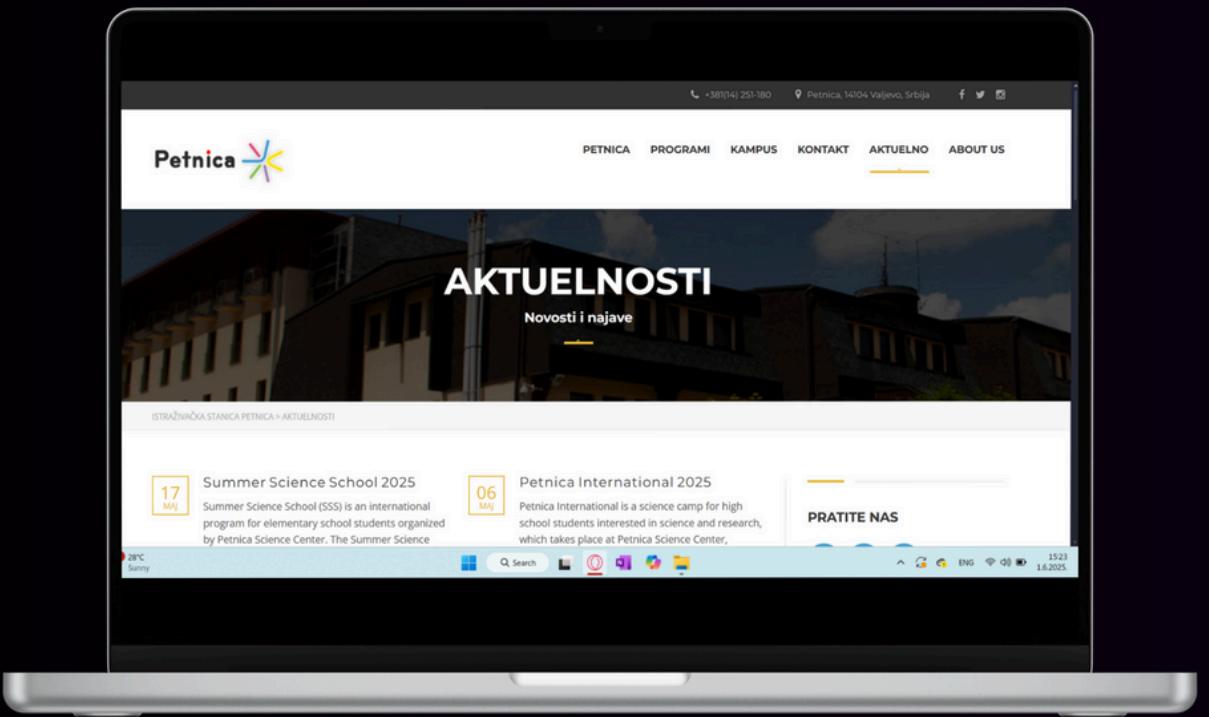
GIT can help you out!





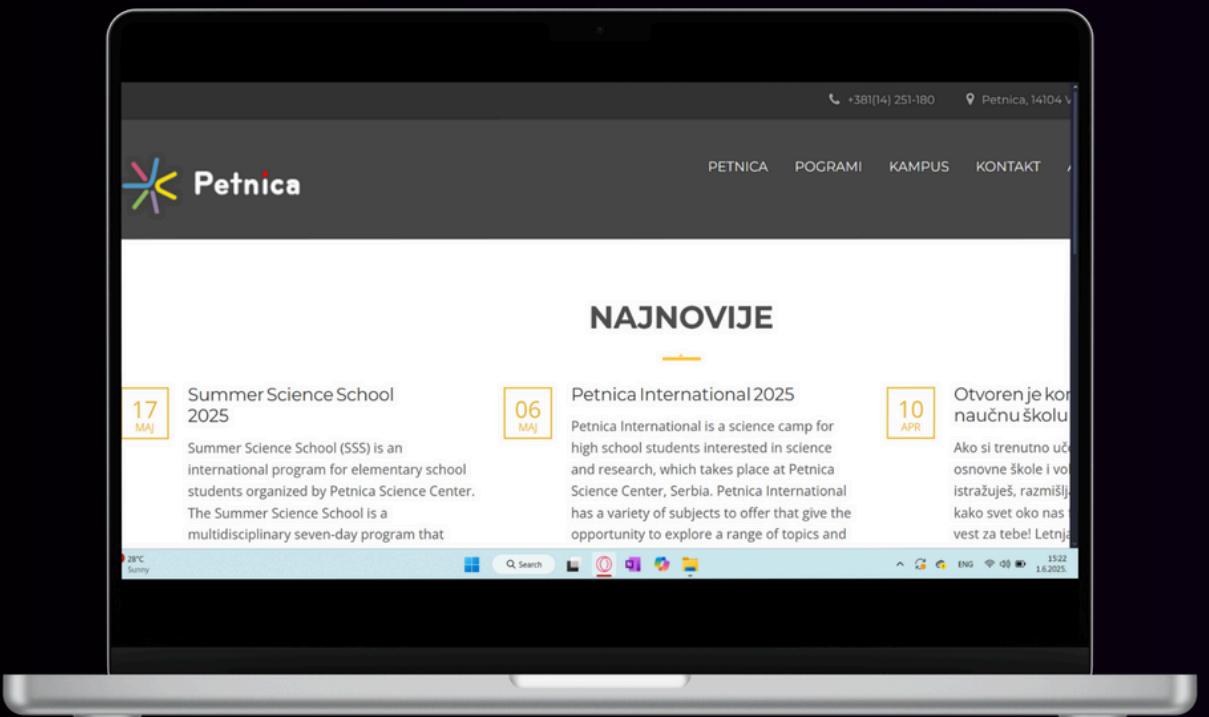
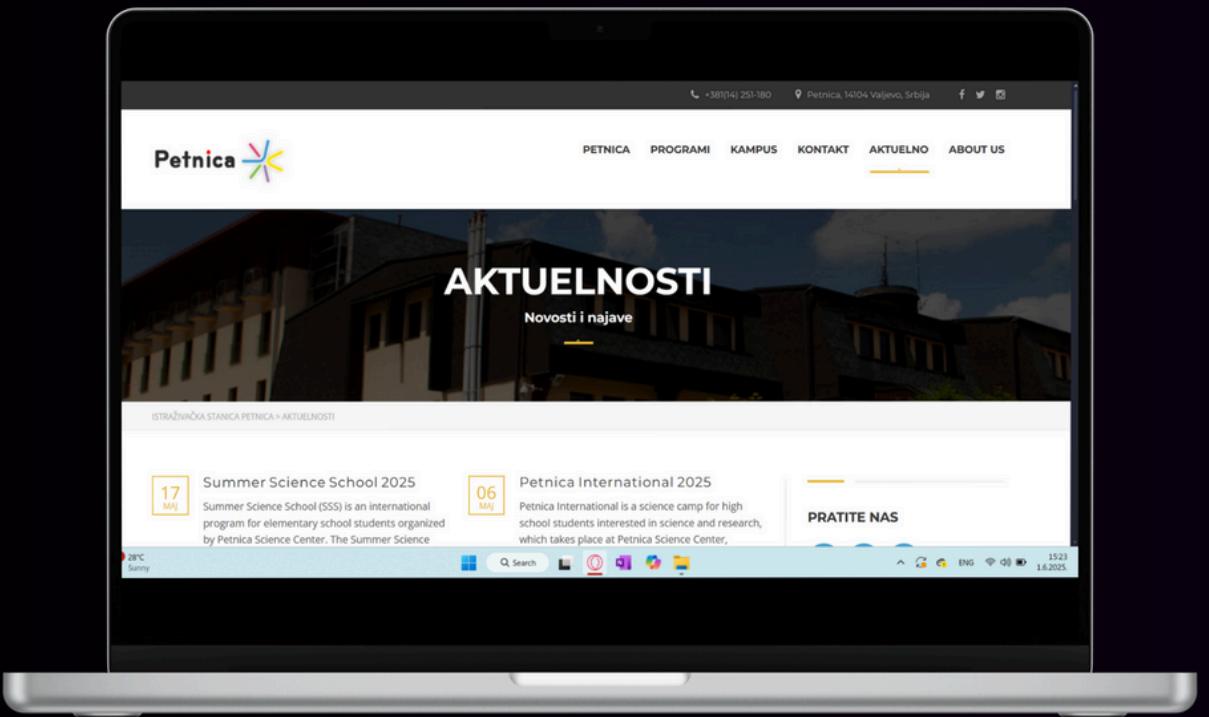
New state of your website with local changes

Hosted Version of your website



New state of your website with local changes

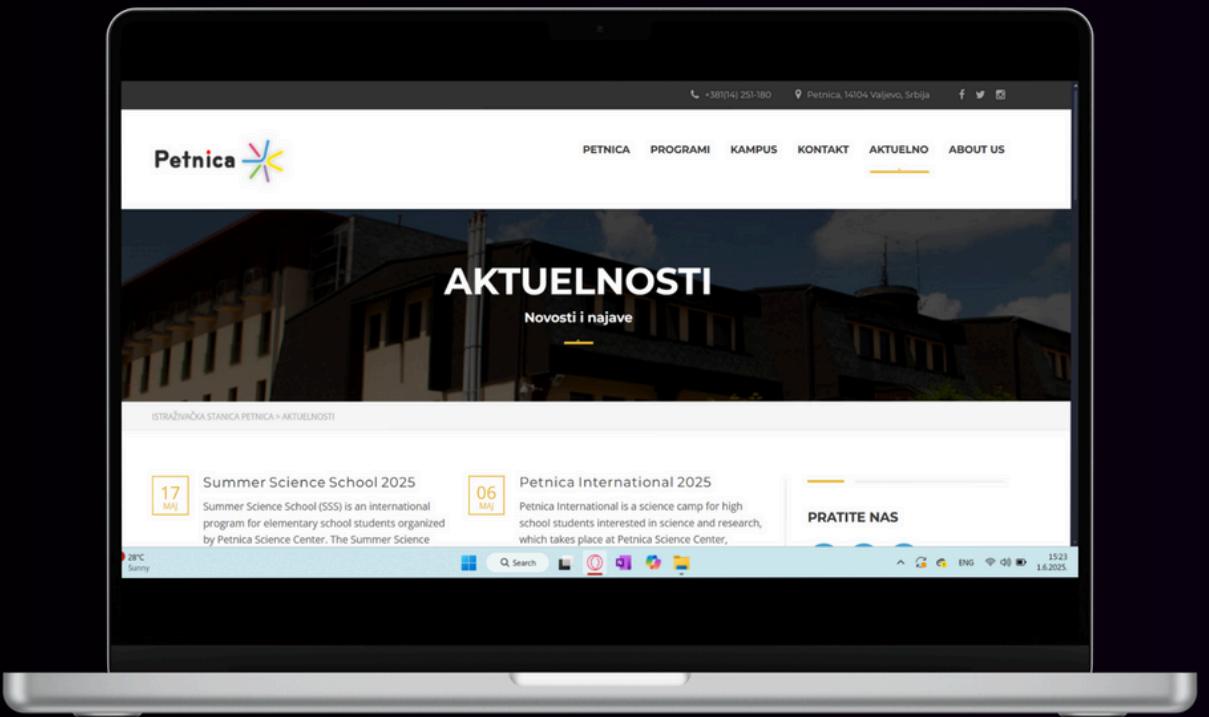
Hosted Version of your website



New state of your website with local changes

Hosted Version of your website





New state of your website with local changes

Hosted Version of your website



Fix button CSS @Aleksa



15:48 PM

Changed background image @Nikola



12:10 PM

Added subtitle @Maja



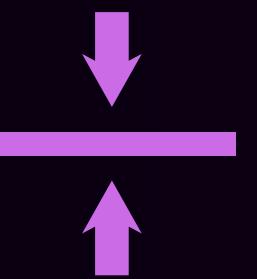
10:10 AM

Added front page @Nikola

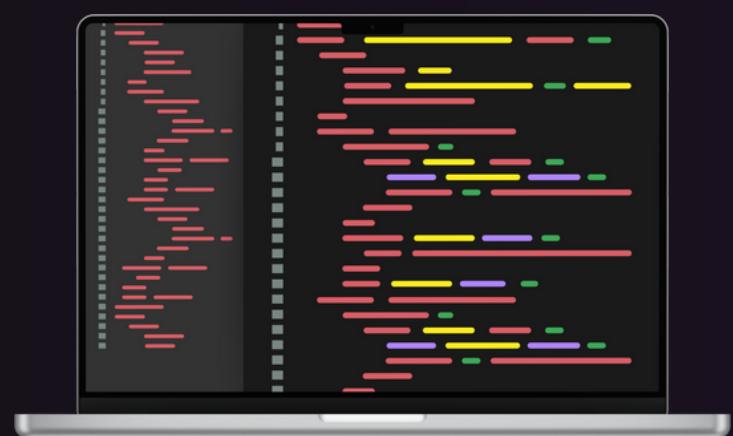


8:10 AM

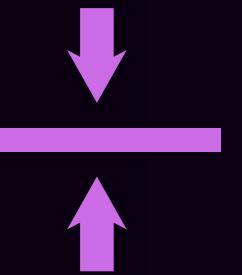
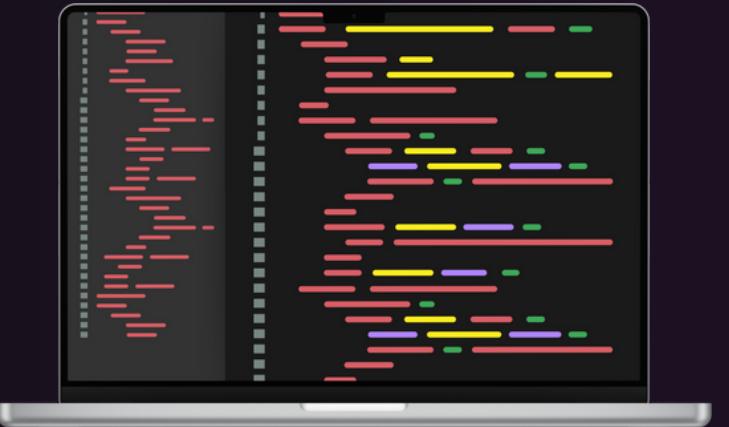
Remote Repository



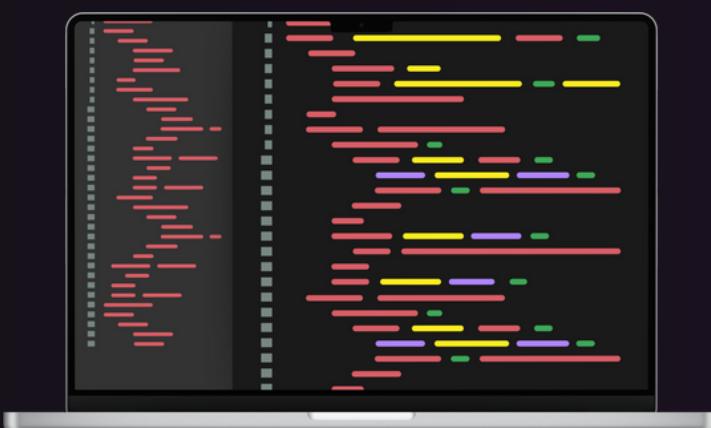
Local Repository



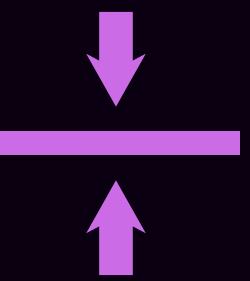
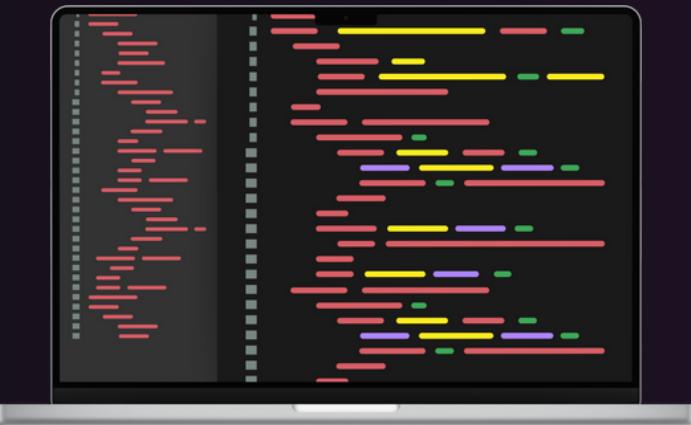
Remote Repository



Local Repository



Remote Repository

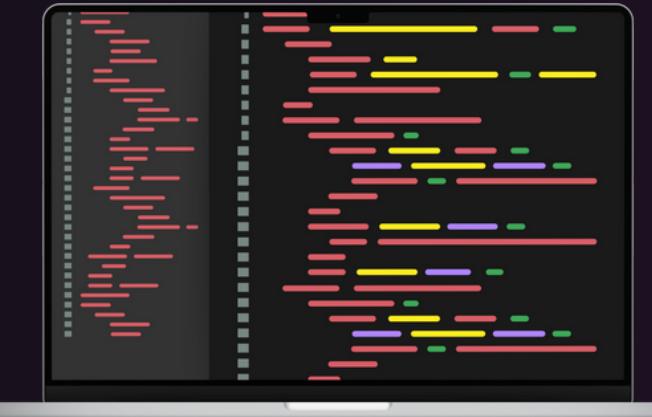


Local Repository

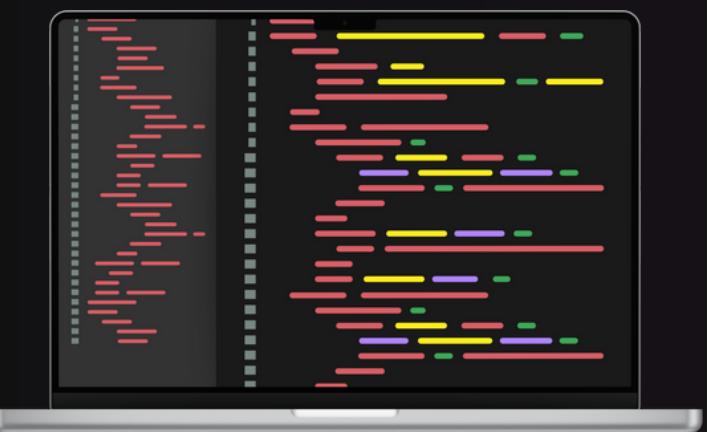
Local Repository



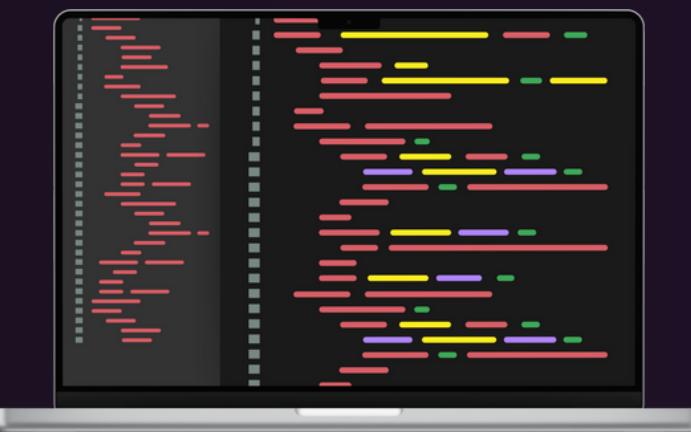
Remote Repository



Local Repository

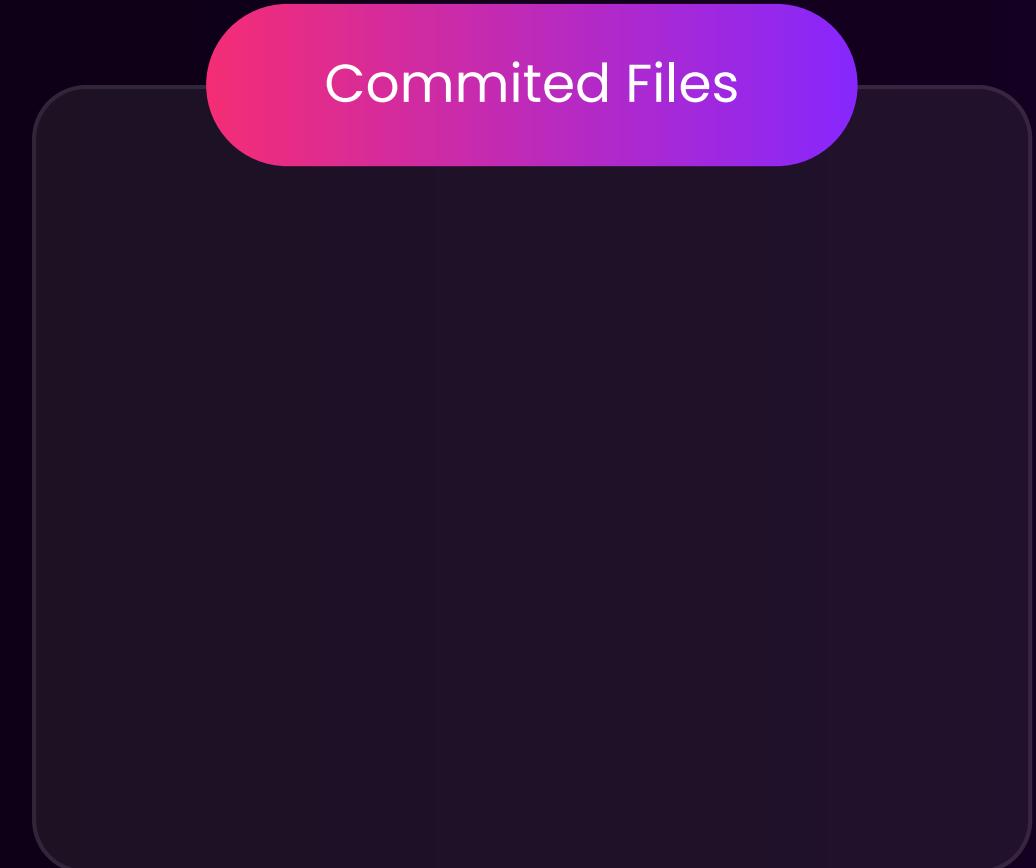
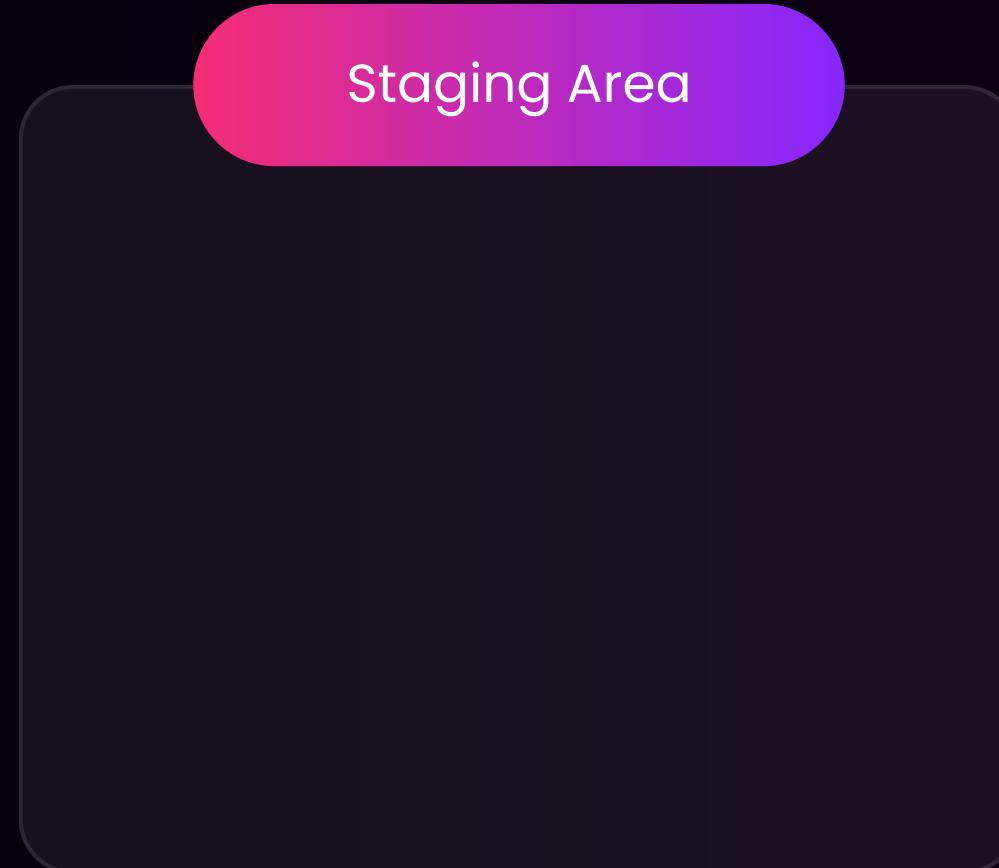
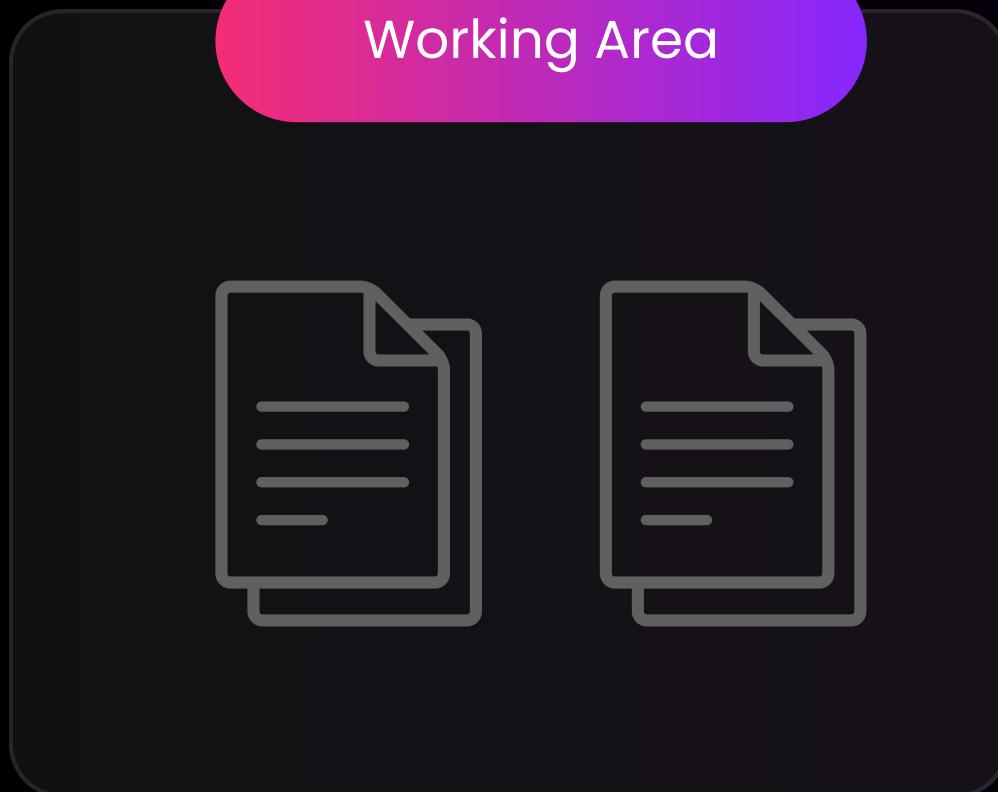


Local Repository





Local Repository





Local Repository

Working Area

Staging Area

Committed Files





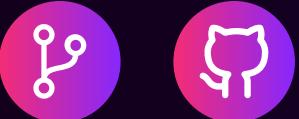
Local Repository

Working Area

Staging Area

Committed Files





Fix button CSS @Aleksa



15:48 PM

Commit

Changed background image @Nikola



12:10 PM

Commit

Added subtitle @Maja



10:10 AM

Commit

Added front page @Nikola



8:10 AM

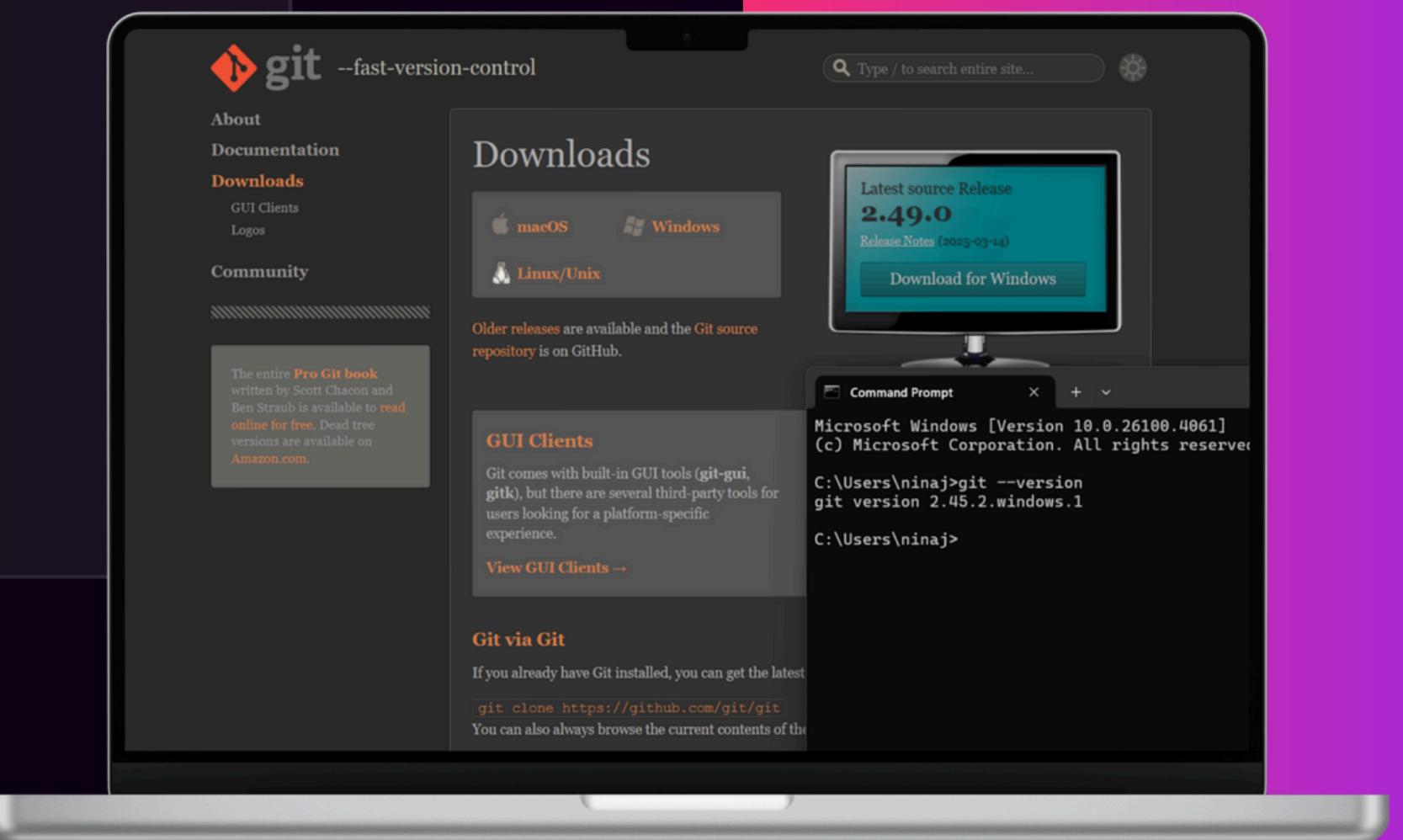
Commit

Installing Git

You can easily find the install scripts for your specific machine on the website:

git-scm.com

To check whether GIT has been installed correctly type: **git --version**





Initializing Git Repository

git init

dir

dir /a

git status



```
C:\Users\ninaj\MyProject>git status
On branch master

No commits yet

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

nothing added to commit but untracked files pr
esent (use "git add" to track)
```



Initializing Git Repository

git init - Initializes a new Git repository in the current directory. This command creates a hidden .git folder that stores all version control information, allowing you to start tracking changes in your project.

dir - Lists all files and directories in the current directory when used in Windows Command Prompt. It helps users see the contents of their current folder.

dir /a - Displays all files and directories, including those with specific attributes.

git status - Shows the current state of your working directory and staging area in a Git repository. It displays which changes are staged for the next commit, which are not, and which files aren't being tracked by Git.





Initializing Git Repository

A commit records the change of a repository compared to its previous state.

git add

```
C:\Users\ninaj\MyProject>git add New.txt

C:\Users\ninaj\MyProject>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   New.txt
```

C:\Users\ninaj\MyProject>



Initializing Git Repository

A commit records the change of a repository compared to its previous state.

```
git add
```

```
git commit -m "cracked enigma's code"
```

```
C:\Users\ninaj\MyProject>git commit -m "cracked  
enigma's code"
```

```
[master (root-commit) f32f12c] cracked enigma'  
s code
```

```
 1 file changed, 1 insertion(+)  
 create mode 100644 New.txt
```

```
C:\Users\ninaj\MyProject>git status
```

```
On branch master
```

```
nothing to commit, working tree clean
```

```
C:\Users\ninaj\MyProject>
```





Initializing Git Repository

- Your default branch is the master branch as we haven't made any new branches yet.
- Storing changes in our local git repository is called committing.
- Before we can commit a file, we first have to put it in the staging area.
- With every commit, we save the current state of the repository and show the changes that we made to it.
- Before we can commit a file, we first have to put it in the staging area.





Displaying commits

The command `git log` shows the information that you need to know about all the commits, such as the commit hash, the author name, the date of the commit and the commit message.

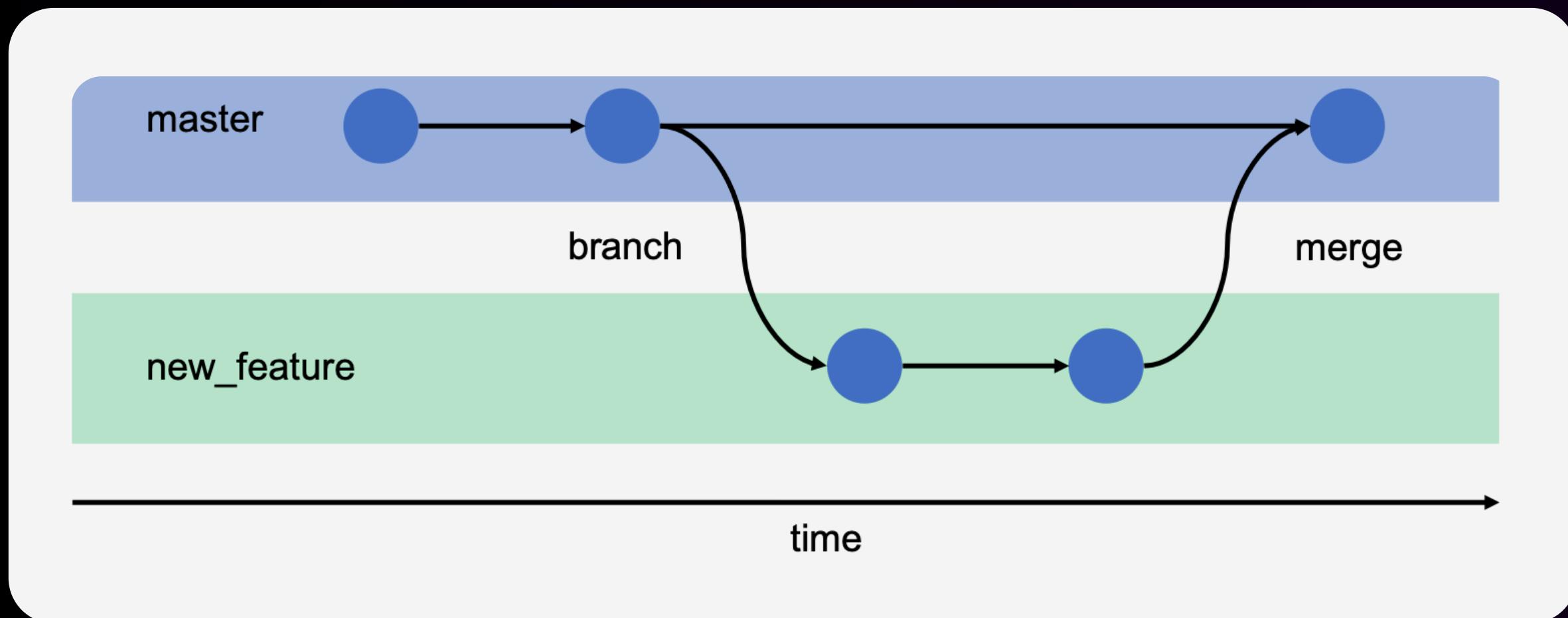
```
git log  
git log --oneline
```

```
C:\Users\ninaj\MyProject>git log  
commit f32f12c4a81d7636df57f885906f2a835577051  
2 (HEAD -> master)  
Author: njakov <nina.jakovljevic11@gmail.com>  
Date:   Sun Jun 1 17:14:37 2025 +0200  
  
        cracked enigma's code  
  
C:\Users\ninaj\MyProject>git log --oneline  
f32f12c (HEAD -> master) cracked enigma's code  
  
C:\Users\ninaj\MyProject>
```

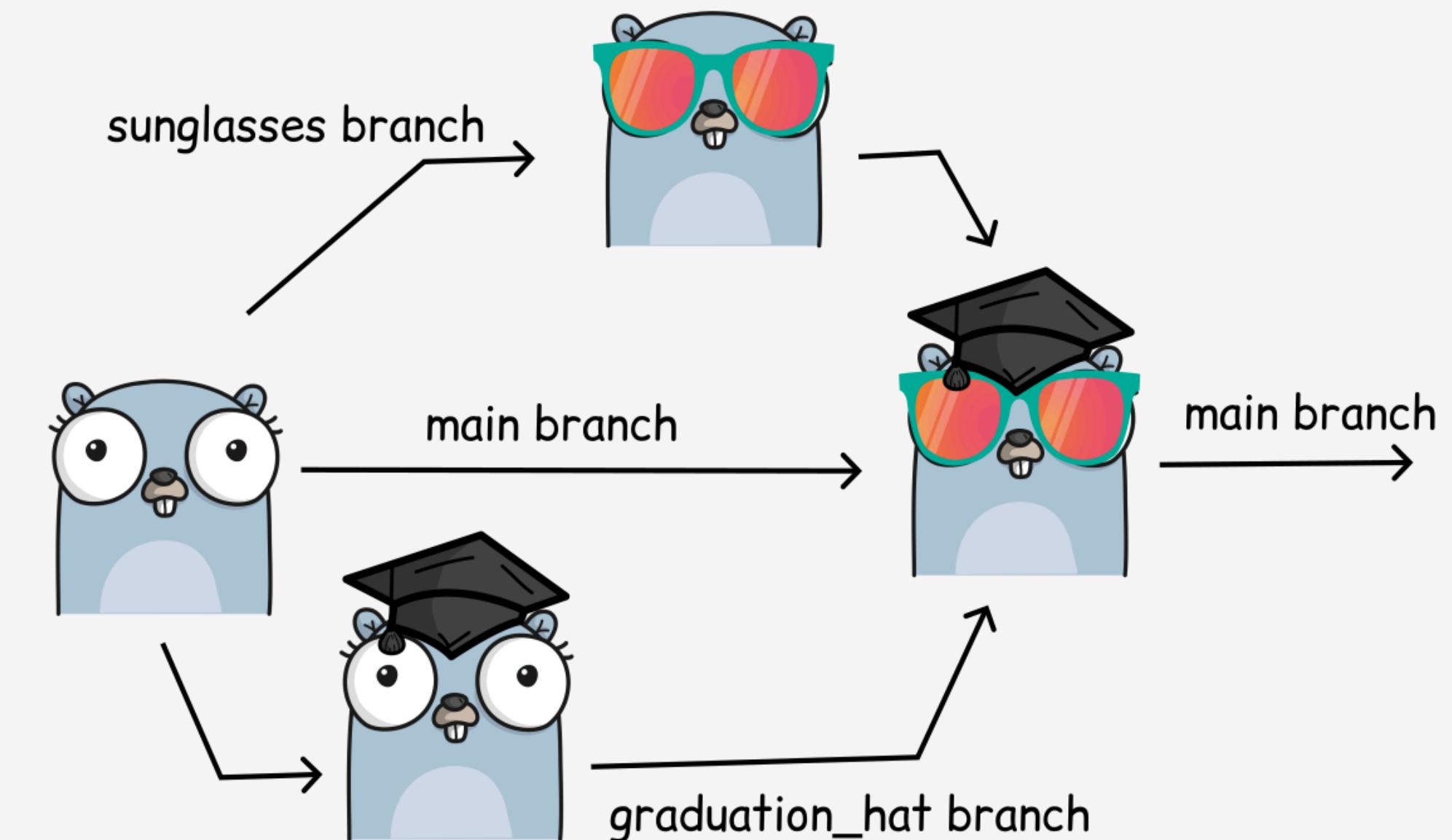




Branches



Branches





Branches

git branch

git branch branch_name

git checkout branch_name

git branch -d branch_name

```
MINGW64:/c/Users/ninaj/MyProject
ninja@Nina MINGW64 ~/MyProject (master)
$ git status
On branch master
nothing to commit, working tree clean

ninja@Nina MINGW64 ~/MyProject (master)
$ git branch
* master

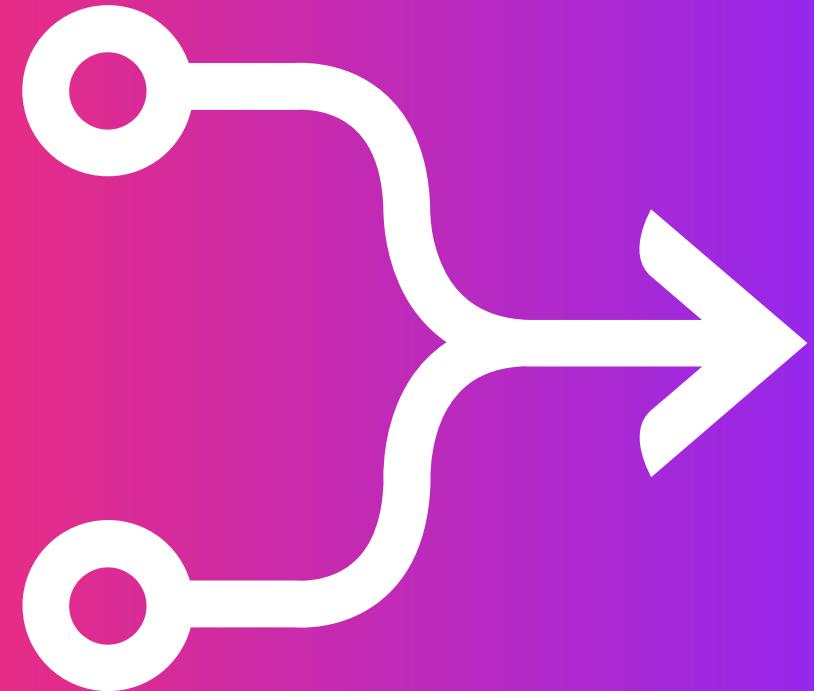
ninja@Nina MINGW64 ~/MyProject (master)
$ git checkout -b nina
Switched to a new branch 'nina'

ninja@Nina MINGW64 ~/MyProject (nina)
$
```





Merging



The git merge command receives the name of the branch that we want to merge into our current branch.

`git merge feature/new`





Remote Repositories



GitLab



BitBucket



GitHub

GitHub





Remote repository

git remote add origin

git remote -v

In order to keep our local and remote repo in sync, we have to push the data from our local repo to the remote repo.

git push origin master

```
ninaj@Nina MINGW64 ~/MyProject (nina)
$ git remote add origin https://github.com/njakov/my-web-app.git
```

```
ninaj@Nina MINGW64 ~/MyProject (nina)
```

```
$ git remote -v
origin https://github.com/njakov/my-web-app.git (fetch)
origin https://github.com/njakov/my-web-app.git (push)
```

```
ninaj@Nina MINGW64 ~/MyProject (nina)
```

```
$
```





Cloning

git clone [ssh link]

By default, it will create a folder locally with the same name as the remote repository.

```
ninaj@Nina MINGW64 ~/MyProject (nina)
$ git remote add origin https://github.com/njakov/my-web-app.git

ninaj@Nina MINGW64 ~/MyProject (nina)
$ git remote -v
origin  https://github.com/njakov/my-web-app.git (fetch)
origin  https://github.com/njakov/my-web-app.git (push)

ninaj@Nina MINGW64 ~/MyProject (nina)
$
```



Useful links



 GitHub Docs

The AI powered developer platform



GitHub.com Help Documentation

Get started, troubleshoot, and make the most of GitHub. Documentation for new users, developers, administrators, and all of GitHub's products.

 GitHub Docs



10 Visualized Commands That Will Transform Your Workflow!

Although Git is a very powerful tool, I think most people would agree when I say it can also be... a total nightmare 😞 I've always found it...

M Dev Genius / Nov 17, 2024

1 Setting up git | Git & GitHub Tutorial for Scientists: It's Not Only for Programmers

Git & GitHub Tutorial for Scientists: It's Not Only for Programmers

dallasdatascience.com /



Thank You

For Your Attention



Email

nina.jakovljevic11@gmail.com

