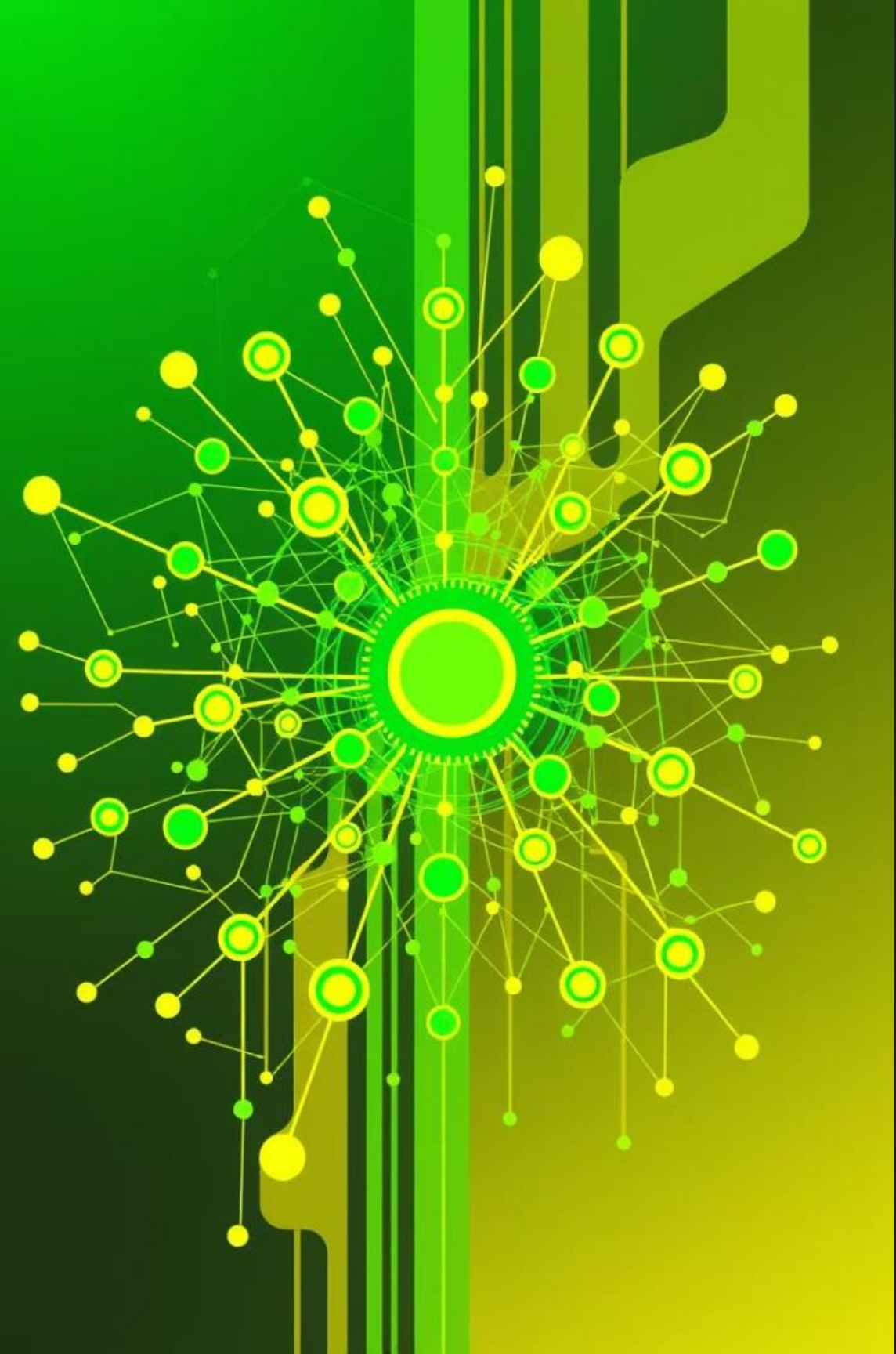


# Outline

- Introduction to GitLab
- Setting Up GitLab
- Creating and Managing Projects
- Version Control with Git
- Collaboration and Sharing
- CI/CD Pipelines
- Conclusion and Q&A

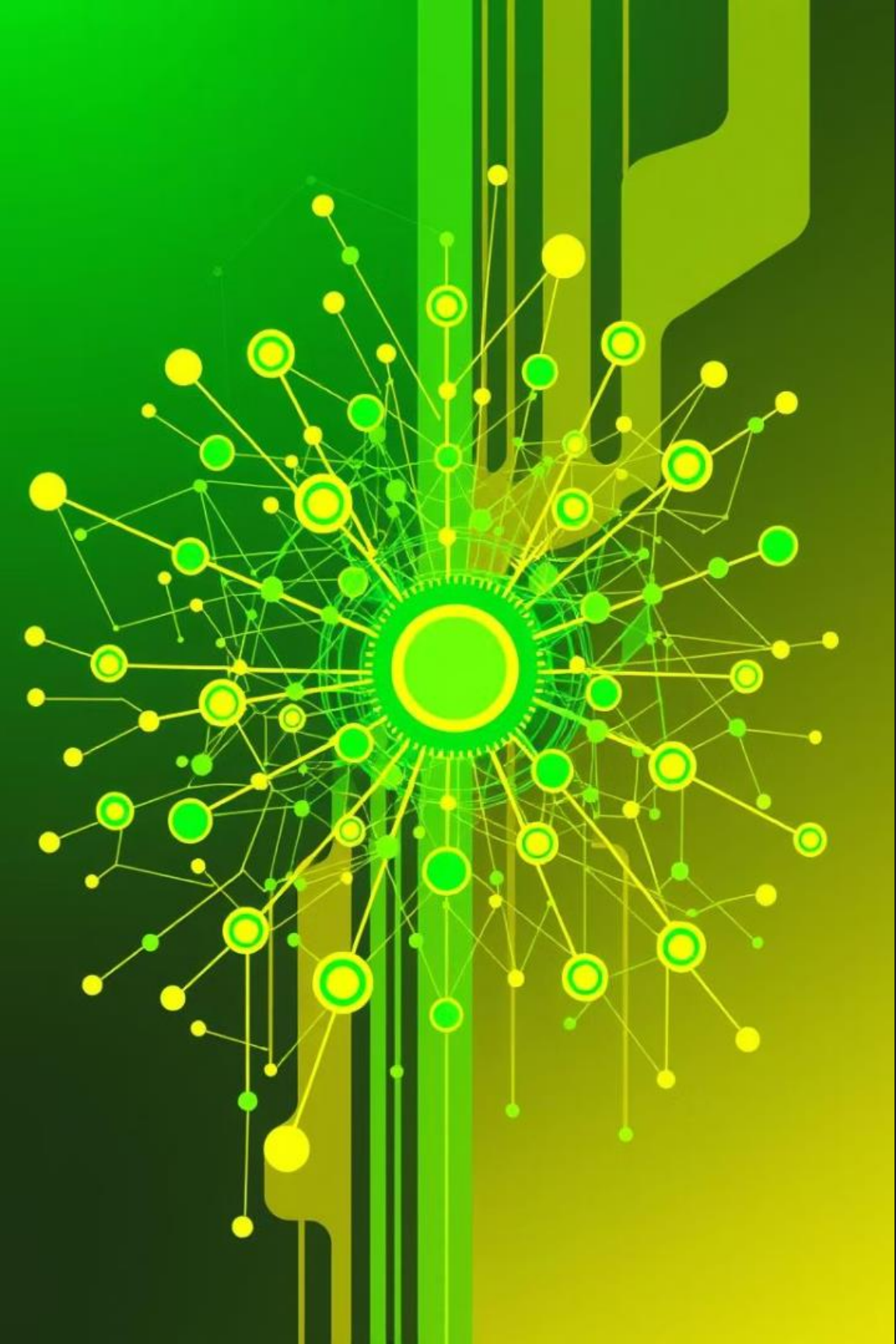


# Introduction to GitLab

**GitLab** is a web-based DevOps lifecycle tool that provides a Git repository manager with wiki, issue-tracking, and CI/CD pipeline features.

## Key Features:

- Version Control
- Continuous Integration/Continuous Deployment (CI/CD)
- Code Collaboration



# Difference between GitLab and GitHub

Both GitLab and GitHub are popular platforms that leverage Git for version control.

## GitLab

- Store Your Code
- Work with Others
- Track Changes
- Automate Tasks
- Manage Projects
- Open Source
- Self-Hosted

## GitHub

- Store Your Code
- Work with Others
- Track Changes
- Showcase Your Work
- Cloud- Hosted

# Setting up a GitLab account

To use GitLab, you'll need to create a free account on their platform.



The image shows a laptop screen displaying the GitLab sign-up page. At the top left is the GitLab logo, a yellow dog head, followed by the text "itLab" in blue and red, and "sign up" in black. Below this is the text "Welcome to create a new account?". There are two input fields: "Email address:" and "Username:". Below the fields is a yellow button with the text "Completed:". At the bottom of the form area, it says "Learn to your scout the you'a account."

1

## Visit GitLab Website

Open your web browser and navigate to the GitLab website (gitlab.com).

2

## Sign Up

Click on the "Sign up" button and provide your email address, username, and password.

3

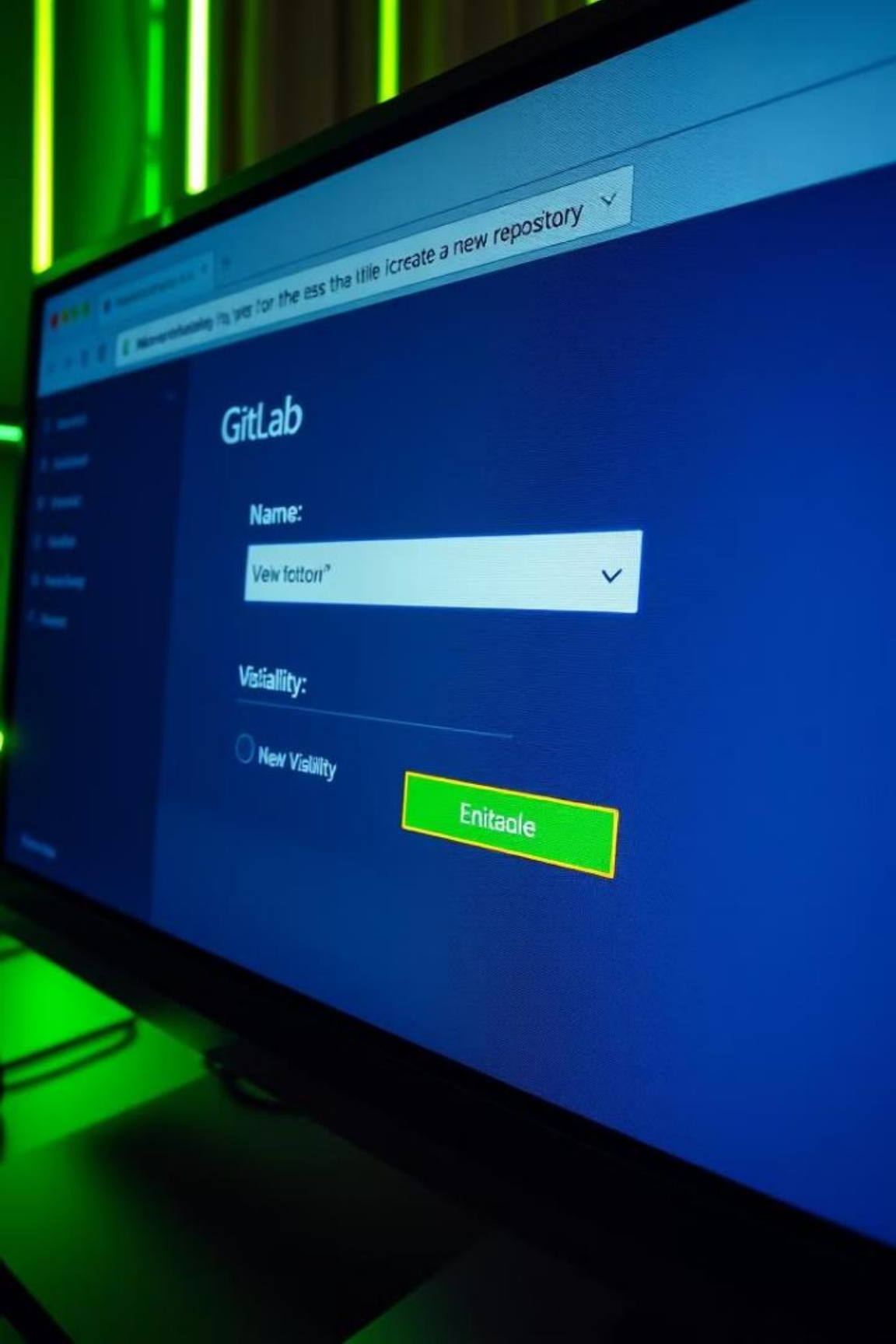
## Verify Email

Check your email inbox for a verification link and click on it to activate your account.



# Creating a new repository

A repository is a central location where you store your code and track changes.



1

## Log In

Log in to your GitLab account.

2

## New Repository

Navigate to the "Projects" section and click on the "New project" button.

3

## Name and Initialize

Give your repository a descriptive name and select options for visibility (public or private) and initialization (with a README file).

4

## Create Repository

Click on the "Create repository" button to create your new repository.



# Cloning a repository

Cloning a repository creates a local copy of a remote repository on your computer.

## Git Clone Command

Use the `git clone` command followed by the URL of the remote repository to clone it.

## Repository Directory

The `git clone` command creates a directory with the same name as the repository in your local machine.

## Local Copy

You now have a local copy of the repository that you can work with independently.



# Committing changes to a repository

Committing changes saves your changes to the local repository, keeping track of what you've modified.

1

## Modify Files

Make the desired changes to the files in your local repository.

2

## Stage Changes

Use the `git add` command to stage the files you want to commit.

3

## Commit Changes

Use the `git commit` command to create a snapshot of your changes with a descriptive commit message.





# What is Git and why use it?

Git is a version control system that allows you to track changes in your code over time.

## 1 Track Changes

Git helps you record every change made to your code, enabling you to revert back to previous versions if needed.

## 3 Manage Code History

Git provides a comprehensive history of your codebase, allowing you to understand how the code evolved and identify the origin of specific changes.

## 2 Collaborate Effectively

Git facilitates collaboration by enabling multiple developers to work on the same project simultaneously, merging their changes seamlessly.

## 4 Prevent Data Loss

Git ensures that your code is backed up and safe, even if your computer crashes or you accidentally delete files.



# Pushing changes to a remote repository

Pushing changes updates the remote repository with the changes you made in your local repository.



## Git Push

The `git push` command sends your committed changes from your local repository to the remote repository.



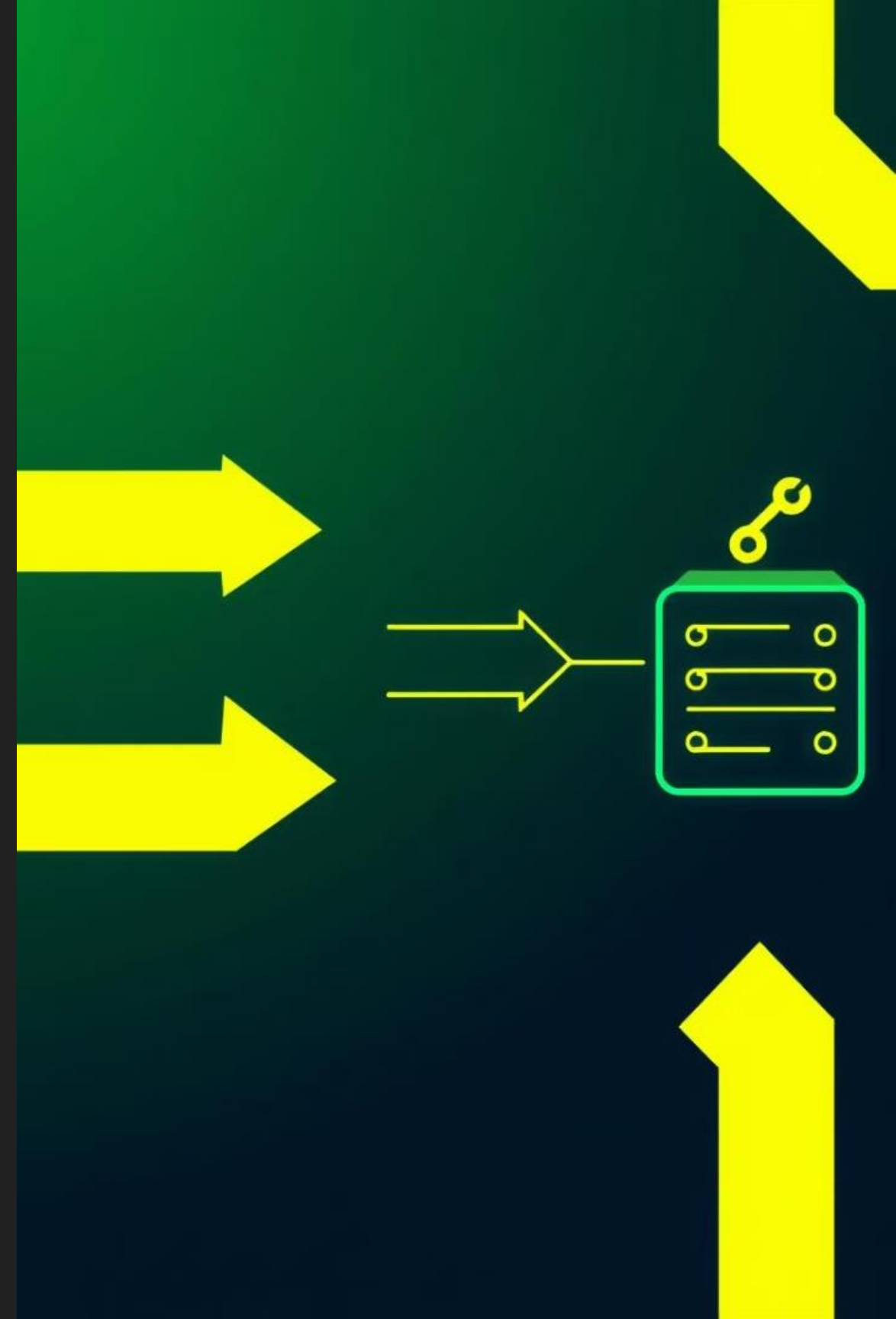
## Remote Update

Your changes are reflected in the remote repository, making them available to other collaborators.



## Synchronized

Both your local and remote repositories now contain the latest changes, ensuring everyone is working on the same codebase.



# Collaborating on a project using GitLab

GitLab provides tools for seamless collaboration on software projects.

Feature

Description

Issue Tracking

Create, manage, and track issues, bugs, and feature requests.

Merge Requests

Propose changes to the main branch and collaborate on code reviews.

Branching

Create separate branches to work on new features or bug fixes without affecting the main branch.

Discussion

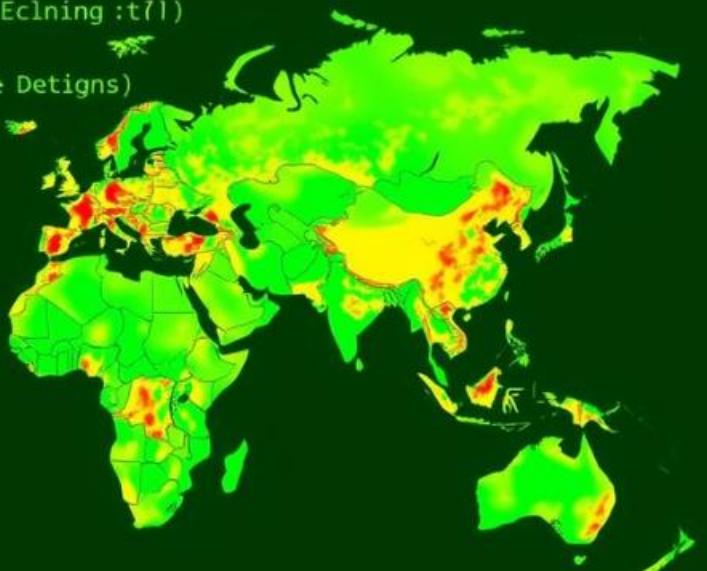
Engage in discussions on issues, merge requests, and other project aspects.



# Integrating GitLab with Python for climate data analysis

GitLab can be seamlessly integrated with Python to streamline your climate data analysis workflow.

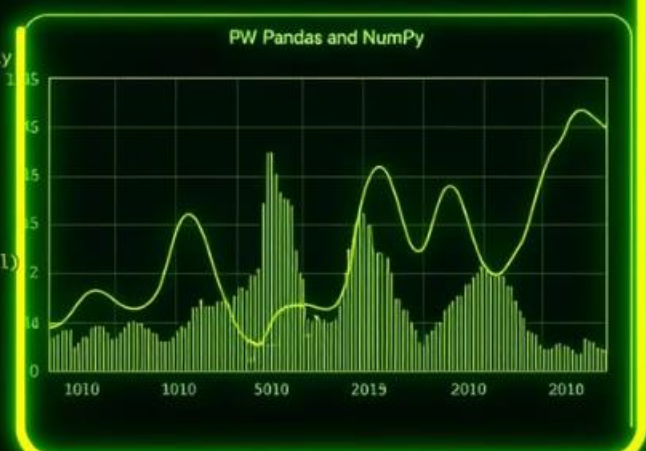
```
API Version taticl, time data sarion {
  API: Catal fobe,
  Care: Fiscital corceations,
  Cose: Cnd Matson clwmer Clomated(Eclning :t71)
  Crast (Lunde-data 0(T1);
  Ctacl: Doscflod make tdstrrer labe Detigns)
  PP ();
  Download and is clitate atray:
}
```



## API Integration

Use Python libraries to interact with the GitLab API and download climate data from repositories.

```
1 PROCESS PLANELLIMATE DATA IT (LUSTACT)
2   Temperature ->0 122)
3   Paandaas - oor 809.85"
4   Tylii ar -fisssoq
5   Vesnt temraable dsor tempetore)
6   Cemperate. Atarlacts s fr- climate rabtins, 224,2987.219.774).
7   Vfbst the rent(channe(9lo, for - climate data)
8   Vesnt thal conterent change thatts (or-77,11.05.49,22.30 /es8d),
9   Chanpettge tabet, for reecalate, im pinotentD1)
10
11   Veist enpitiorw ands in camu dil
12   Panddas- 42:s:700
13   Vabce - dasy in Rariic, this for1el wn 5 pralcaly
14   at?:
15
16   Ptatics case - 4- 12:401)
17   Viast percernerd limat temperture. Wip)
18
19   Clooe efs levde:
20   5, sullctaton
21   Tts tahes fow to lat for idel4935:
22   Tesauring turee chaarge, clinet* - tr rarerante datl)
23   or finet:
24
25
```



## Data Analysis

Utilize Python libraries like Pandas and NumPy to analyze, process, and visualize climate data to gain valuable insights.