Git and GitHub: Essential Tools for Version Control

Git and GitHub are essential tools for developers to track changes, collaborate with others, and manage code. Today, we'll explore the basics of Git and its benefits, along with key commands and the workflow for using Git.

by The XYZ Company

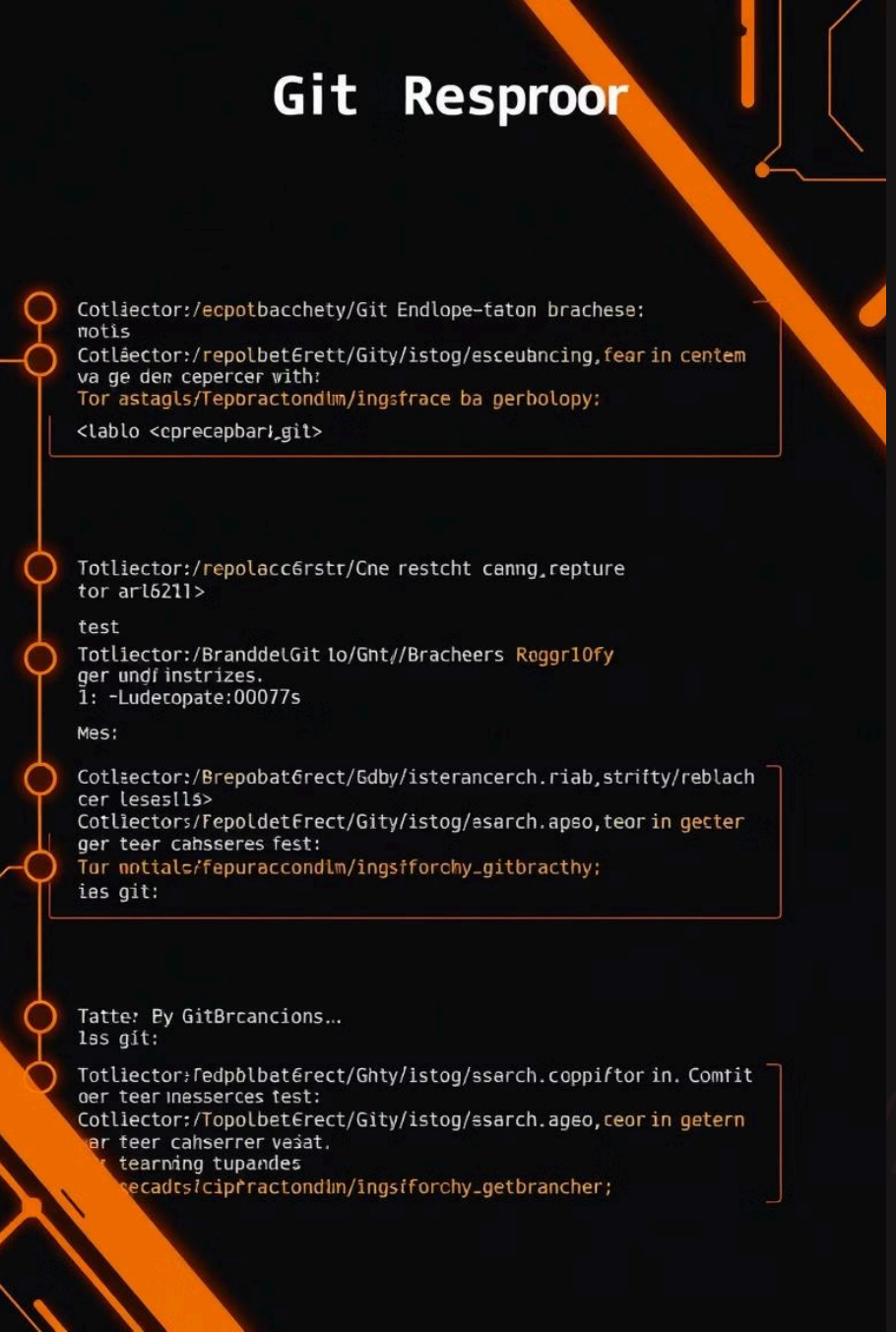
What is Git?

Version Control System

Git is a version control system that keeps track of changes made to files over time, allowing you to revert to previous versions, track code history, and collaborate with others on projects.

Distributed System

Git is a distributed system, meaning that each developer has a local copy of the project repository, enabling independent work and offline access to the codebase.



Git Basics: Repositories, Commits, and Branches

1 Repositories

A repository is a directory that contains all the files and directories related to a project.

2 Commits

Commits are snapshots of the project's state at a particular point in time.

3 Branches

Branches allow developers to work on separate features or bug fixes without affecting the main codebase.

Git Commands You Should Know

Git add

Stage changes to be included in the next commit.

Git commit

Create a snapshot of the current changes and add it to the project history.

Git push

Upload changes from your local repository to the remote server.

Git pull

Download changes from the remote server to your local repository.

Git clone

To clone remote repo to your local environment

Git status

check if your current repo

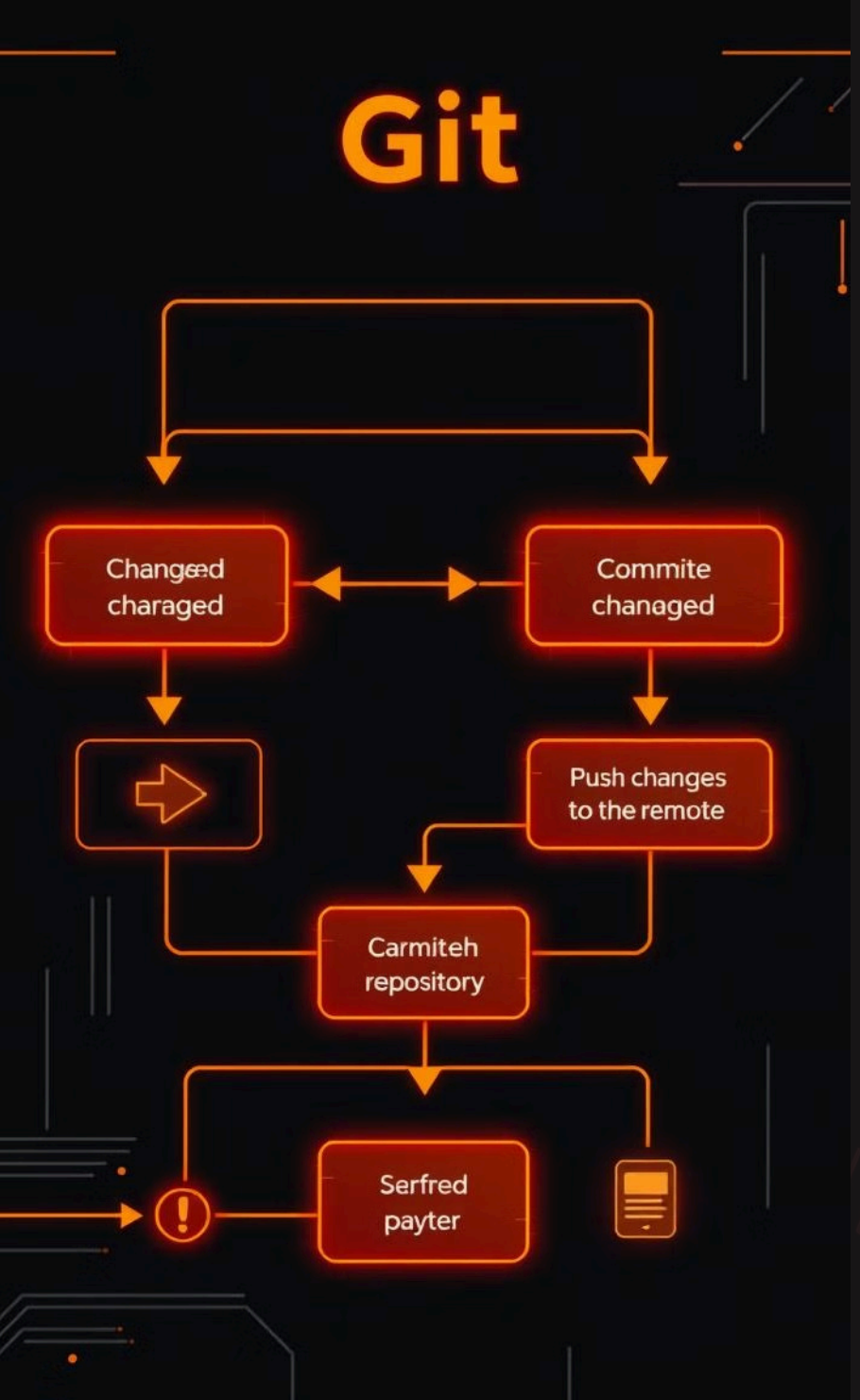
is clean or not

Demo to push local code to remote repo

git add .

git commit -m. "some message"

git push



Git Workflow: Add -> Commit -> Push

1

Git add

Stage your changes to be included in the next commit.

2

Git commit

Create a snapshot of the current changes and add it to the project history.

3

Git push

Upload your changes from your local repository to the remote server.