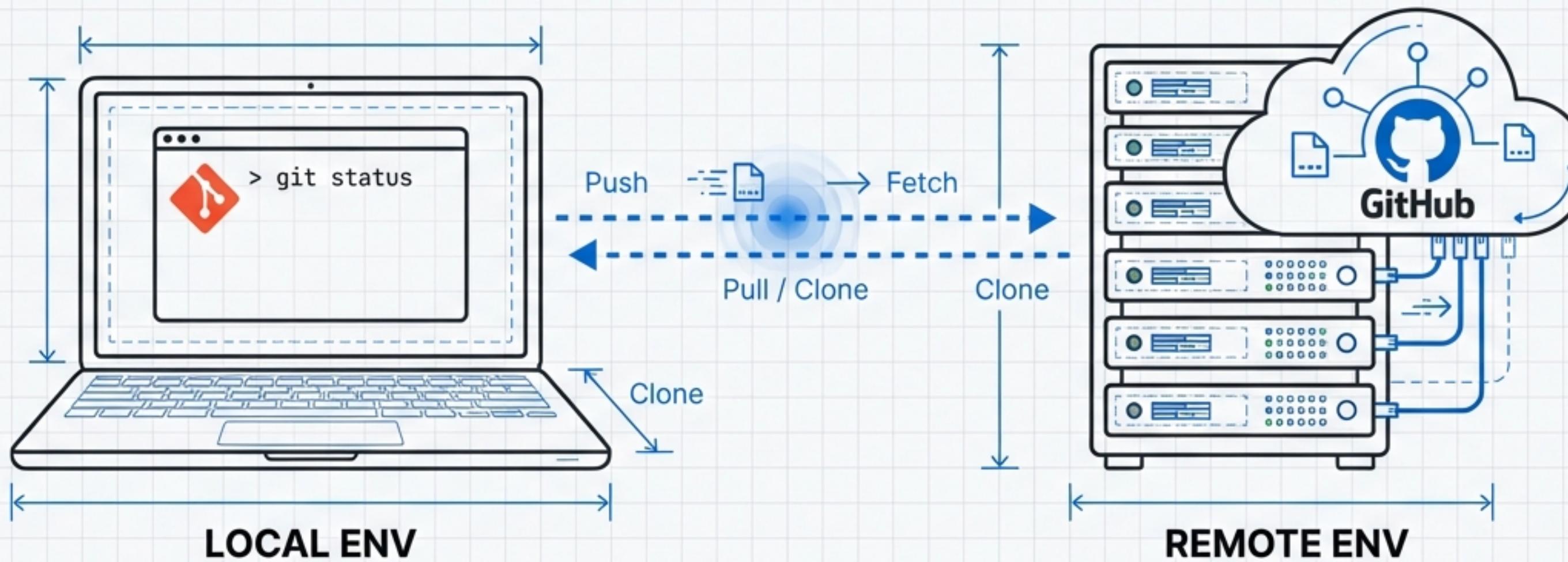


The Complete Guide to Git & GitHub

From Version Control Basics to Professional Collaboration



A comprehensive masterclass designed to take you from 'Zero to Hero' in version control, covering installation, architecture, and advanced team workflows.

The Power Duo: Git vs. GitHub

Git

The Tool (Local)



A version control system that runs offline on your machine.



Like MS Word: You create and save files locally.

GitHub

The Platform (Cloud)



A cloud-based service to host and share your Git repositories.



Like Google Docs: You collaborate and share online.

Tracking History

Team Collaboration

Getting Started: Installation & Configuration

Installation



Windows

Download from git-scm.com



Mac

`brew install git`



Linux

`sudo apt-get install git`



Verify install: run '`git --version`'

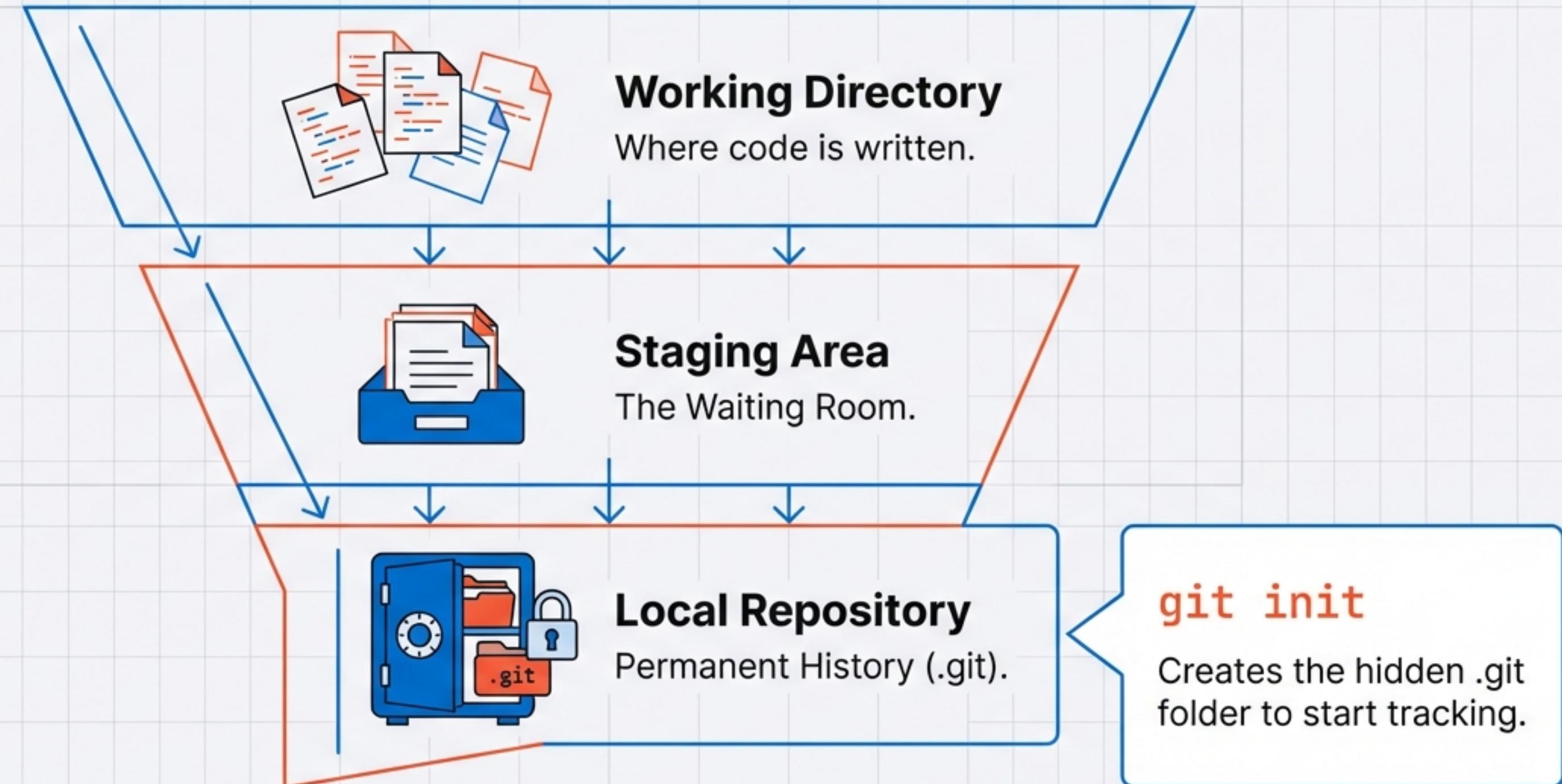
Identity Configuration

Who are you? (Identity Setup)

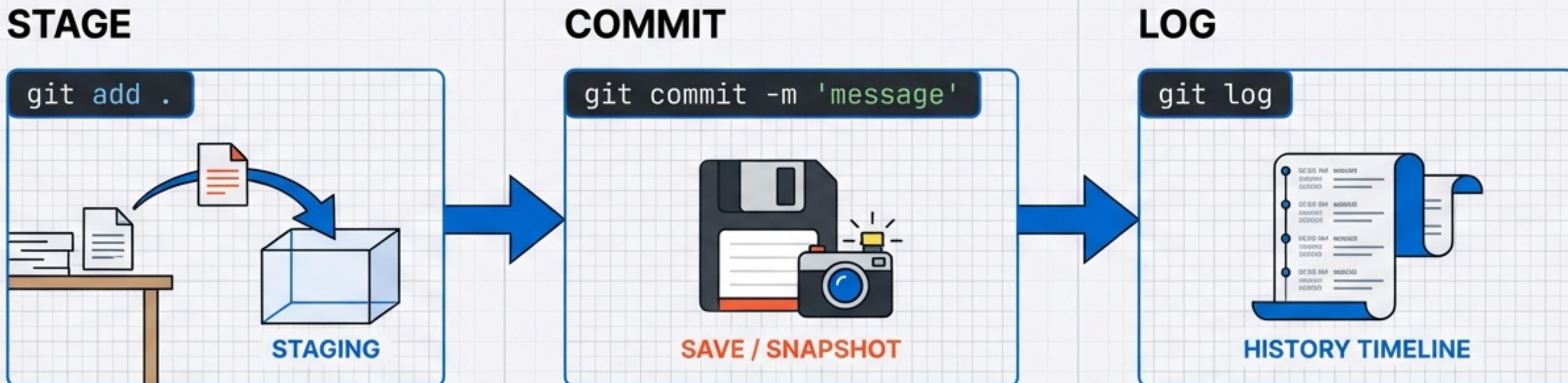
```
git config --global user.name 'Your Name'  
git config --global user.email 'you@email.com'  
git config --list
```

```
user.name=Your Name  
user.email=you@email.com  
user.atoon=Your Name  
user.global=you@email.com
```

The Architecture of Git



The Daily Loop: Add, Commit, Log



Prepares files for saving.

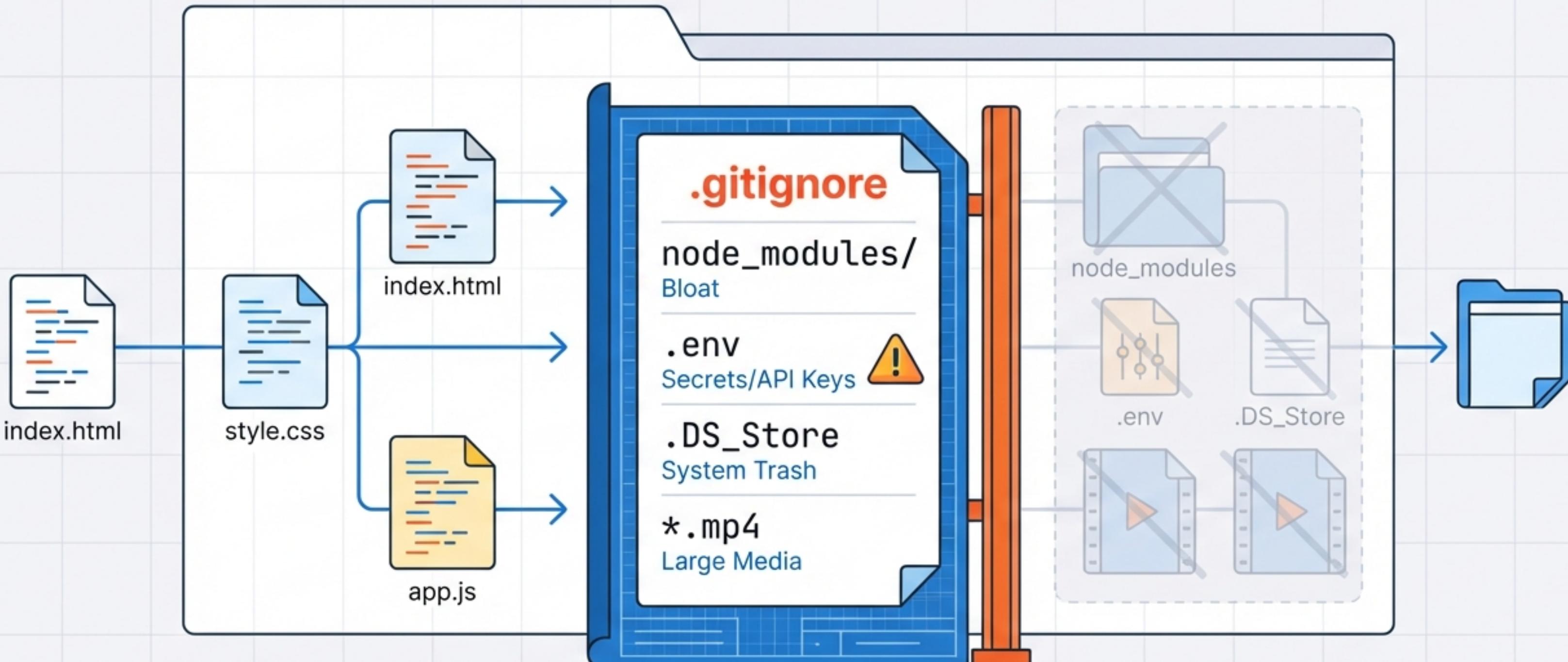
Metaphor Callout: Think 'Checkpoints', not just Ctrl+S.



 **Pro-Tip:** Write descriptive messages!

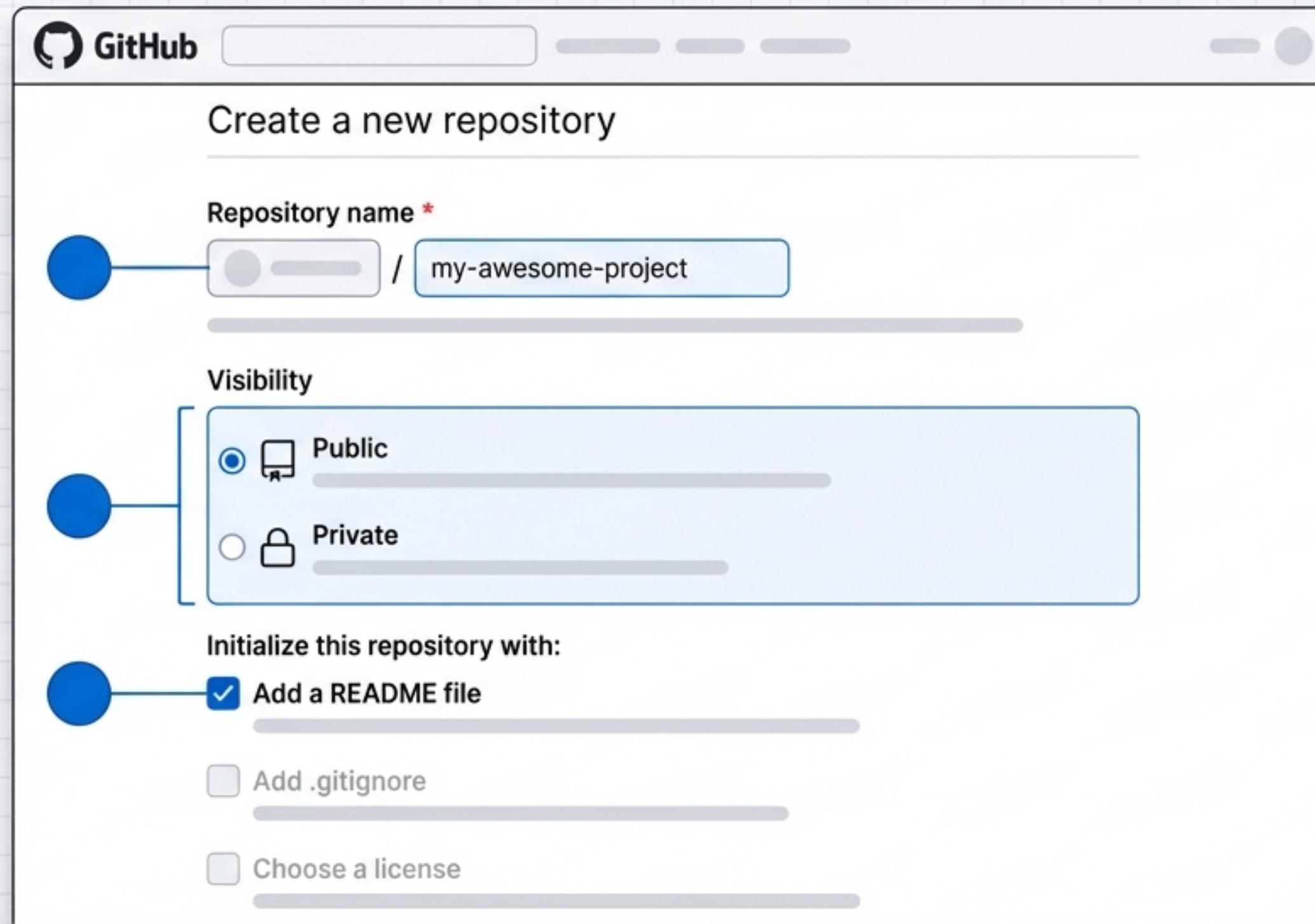
Review your timeline.

Keeping it Clean: The .gitignore File



Tell Git what to ignore to keep your repo secure and lightweight.

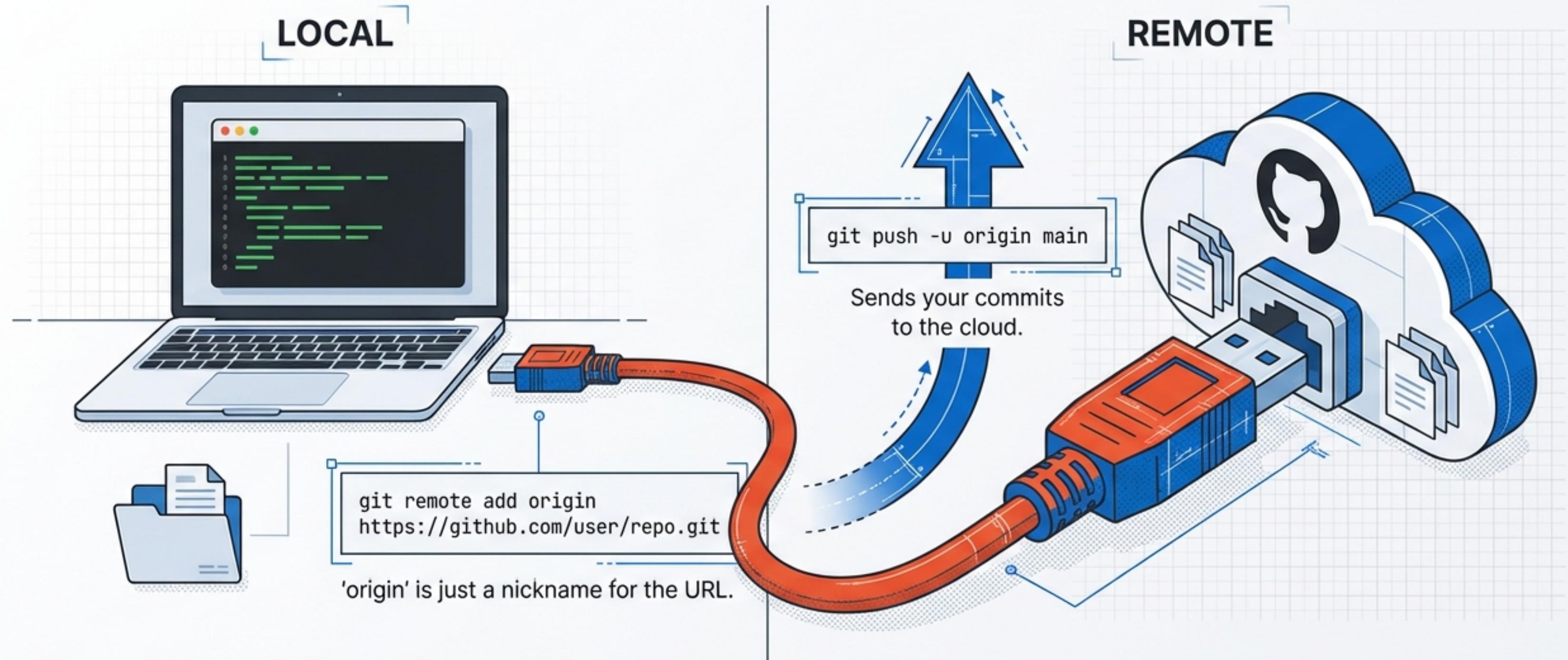
Going Global: GitHub Repository Setup



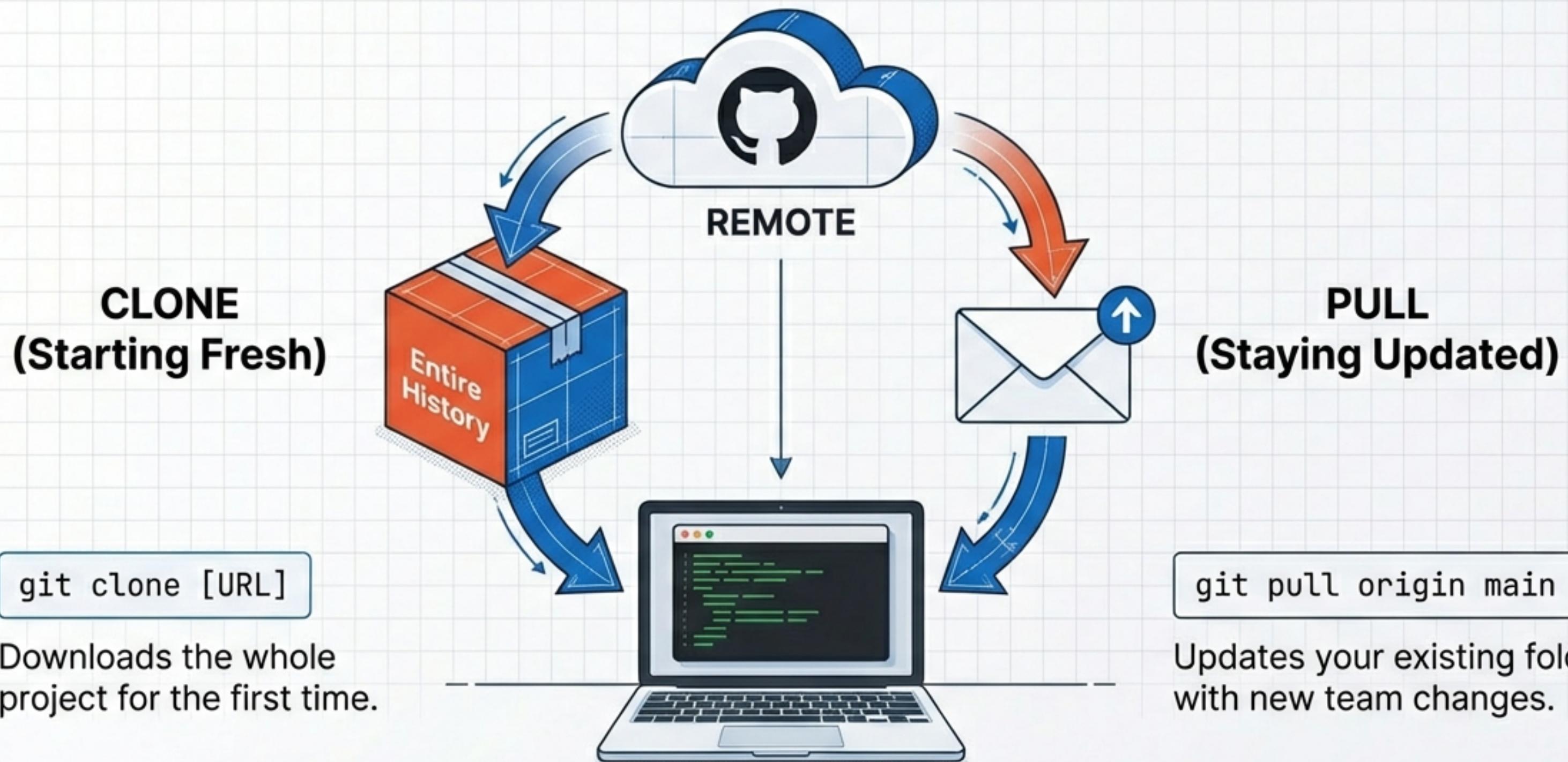
Profile Pro-Tips

- Add a Bio & Social Links.
- Use a README.md to introduce your project to the world.

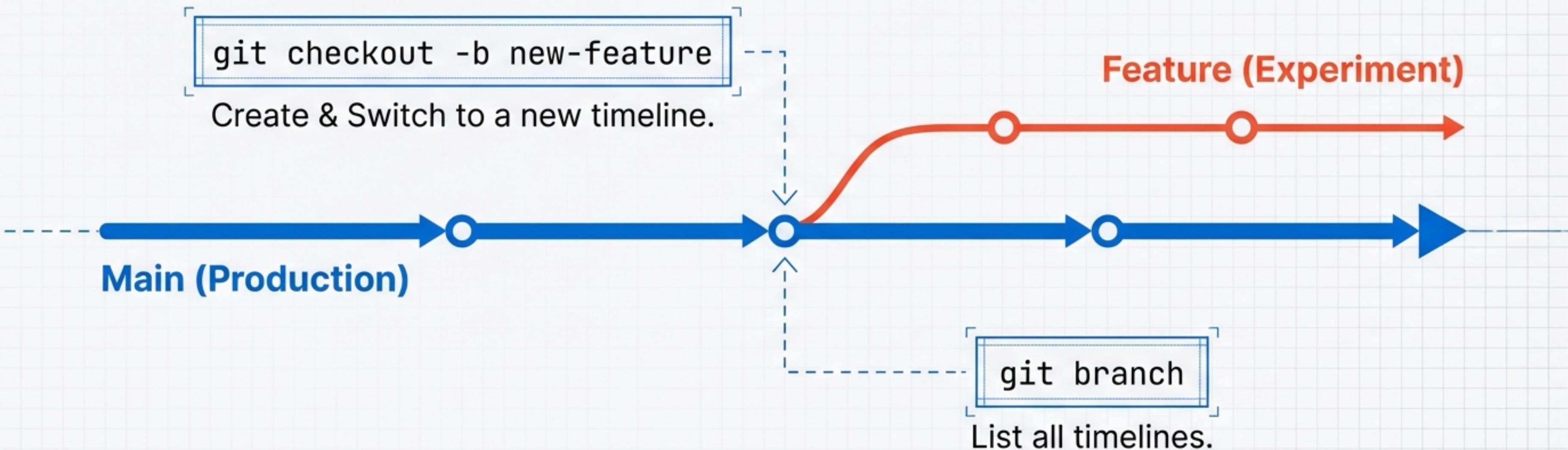
The Bridge: Linking Local to Remote



Two-Way Traffic: Clone & Pull

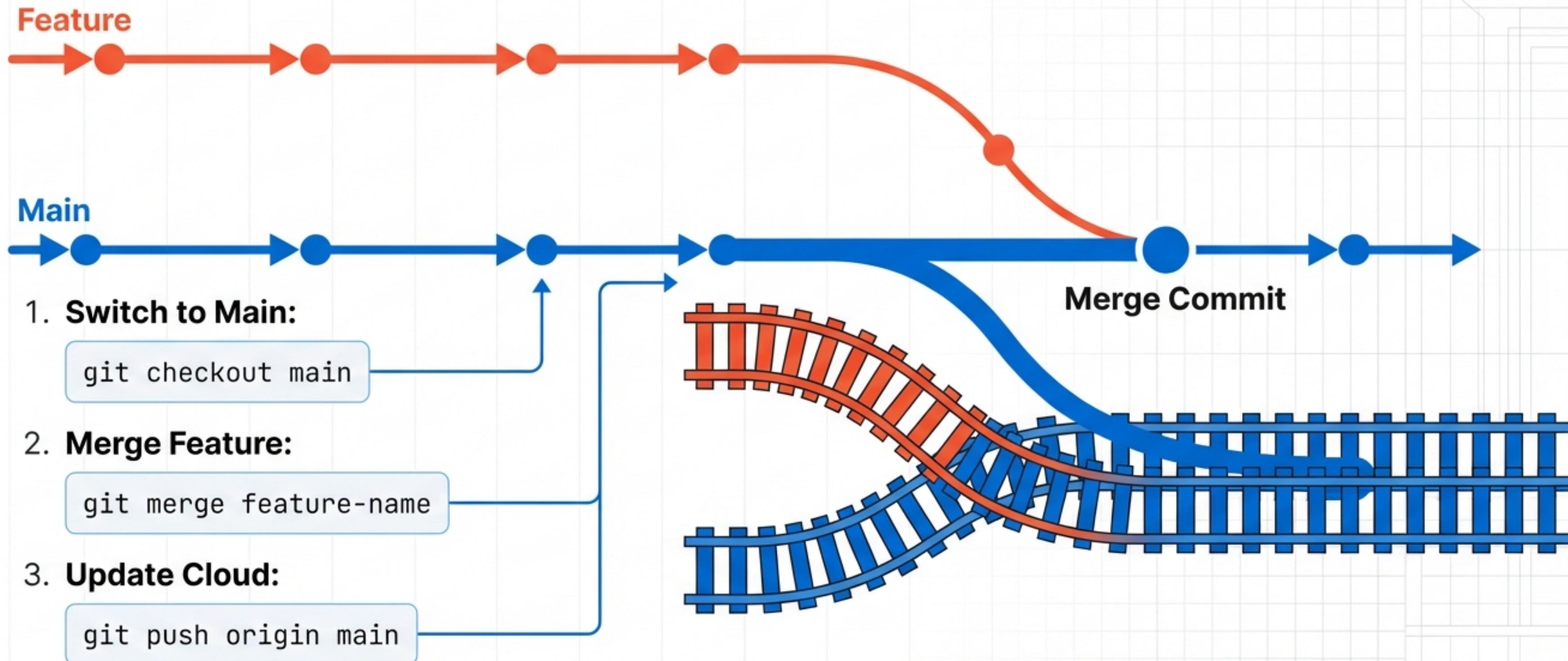


Multiverse Theory: Branching



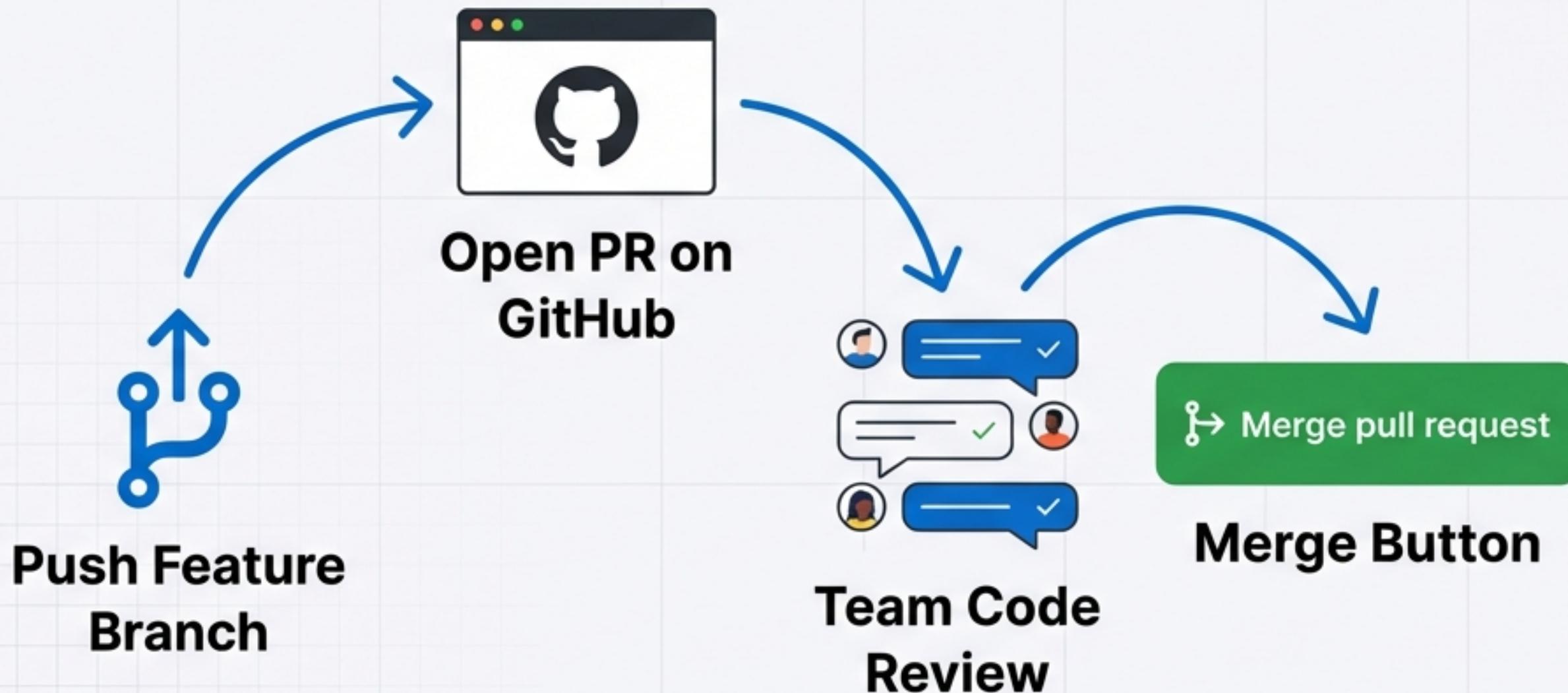
Why: Work on new ideas without breaking the live code.

The Merge: Combining Timelines



The Professional Standard: Pull Requests (PR)

⚠ NEVER push directly to `main` in a team setting.



Benefits

- Quality Control
- Discussion
- Safety

When Things Go Wrong: Merge Conflicts



A screenshot of a terminal window with a dark theme. The window title bar has three colored circles (red, yellow, green). The terminal content shows a merge conflict in a file named 'script.js'. The code is as follows:

```
1 <<<<< HEAD
2 var color = 'blue';
3 =====
4 var color = 'green';
5 >>>>> feature-branch
```

Explanation:

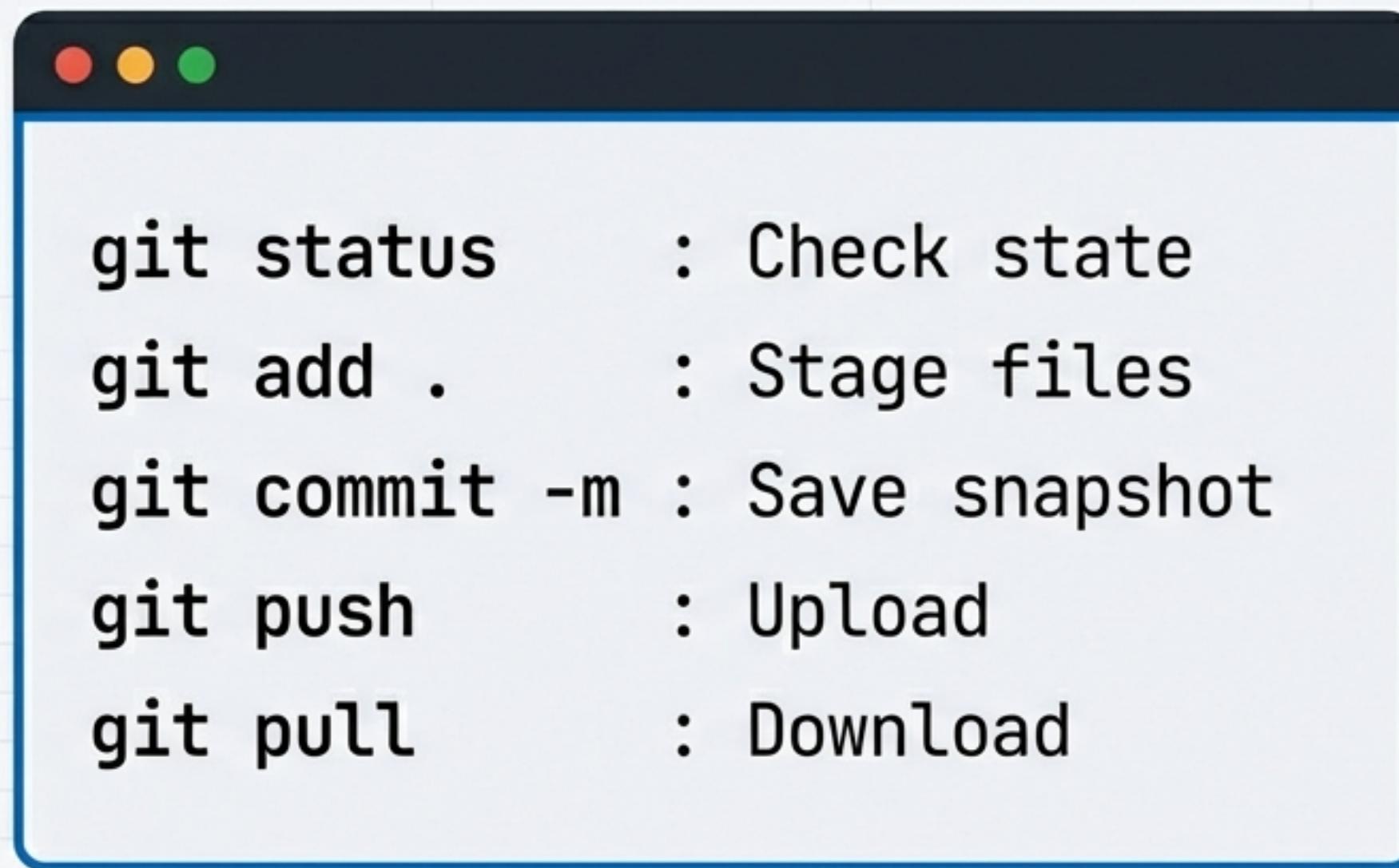
- ⚠ **Cause:** Two people changed the exact same line.
- ✓ **Fix:** Delete the markers. Choose the correct code. Save.

Resolution Command:

```
git add . && git commit
```

The Cheat Sheet & Best Practices

Commands



Golden Rules

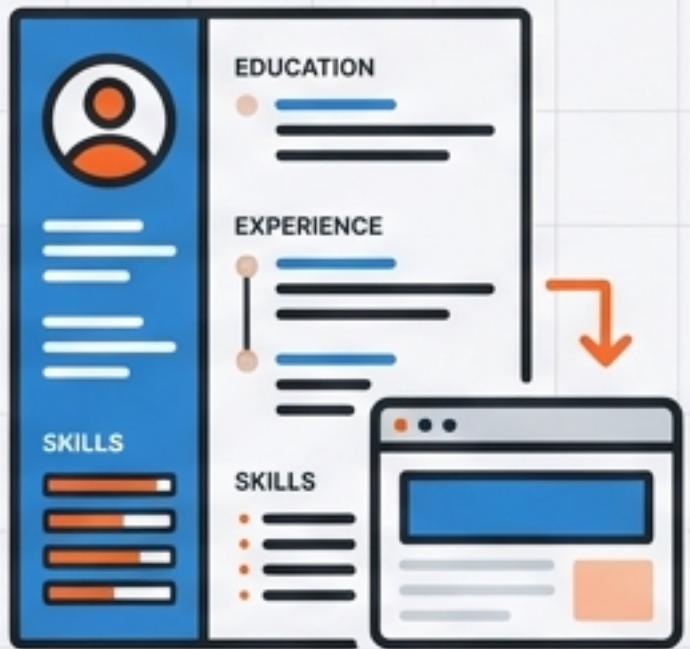
1. Commit Often
(Small, frequent saves).

2. Be Descriptive
(Clear messages help future-you).

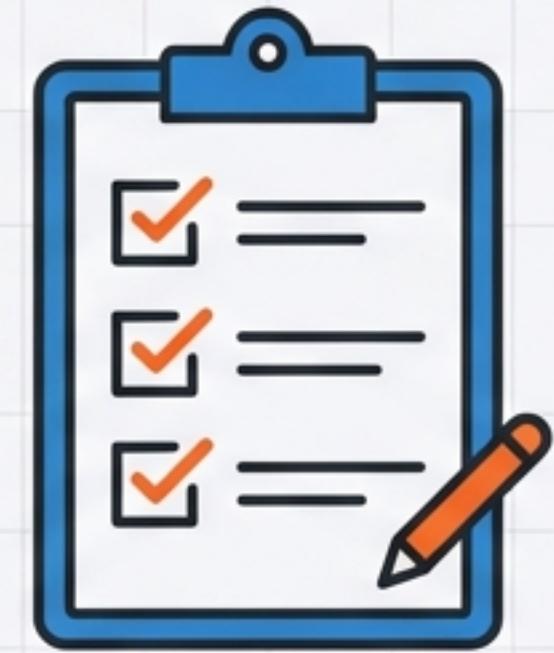
3. Pull First
(Always update before starting).

4. Secure
(No API keys in public repos).

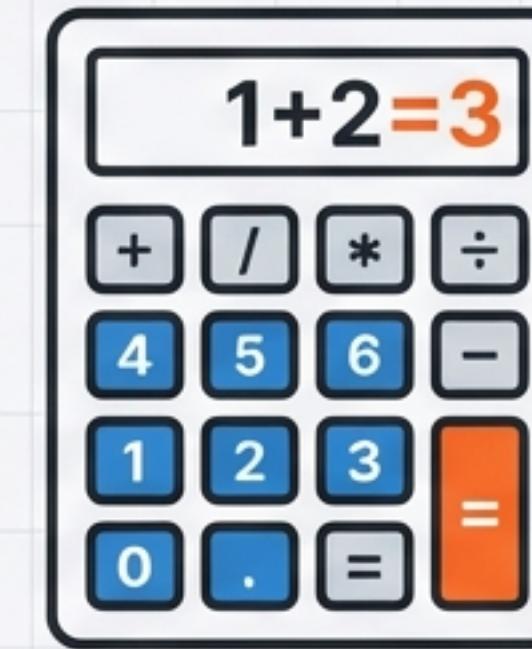
Start Building Today



Portfolio Site



To-Do List App

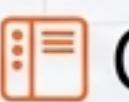


Calculator

Next Level



GitHub Issues



GitHub Projects



GitHub Actions

Create your first repository today. Consistency is key.