

NKULULEKO SEDIBE - FULLSTACK DEVELOPER

Get it with Git & GitHub

Topic: GitHub

Submission Date: 02 July 2024

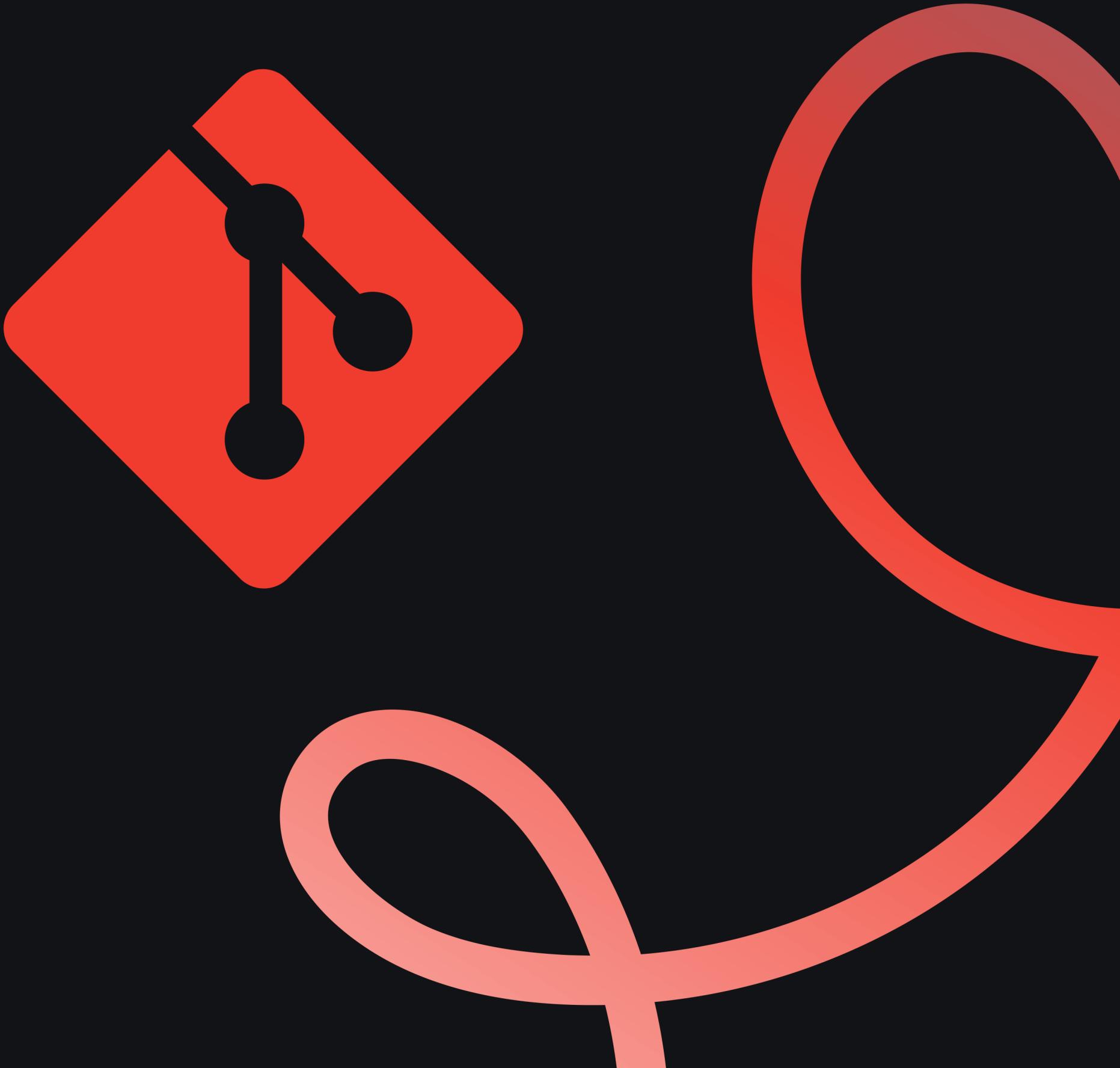
Time: 12:00

NIMBUS STUDIOS - iNKULU LAB

What is Git?

Git is a distributed version control system that tracks changes in source code during software development.

It allows developers to manage and coordinate work on a project by maintaining a history of changes, branches, and versions of files.



What is GitHub?

GitHub is a web-based Git repository hosting service. It provides features beyond Git's core functionality, such as access control, bug tracking, task management, and wikis.

Essentially, GitHub makes it easier to collaborate on projects, share code, and manage repositories.





Difference Between Git and GitHub

- 1 Git is the software itself, while GitHub is a service that hosts Git repositories.
- 2 Git is a command-line tool, whereas GitHub provides a graphical user interface.
- 3 Git is installed locally on your system, while GitHub is hosted on the web.
- 4 Git is maintained by the Linux community, while GitHub is maintained by Microsoft.
- 5 Git focuses on version control and code sharing, while GitHub focuses on centralized source code hosting.
- 6 Git manages source code history, while GitHub hosts Git repositories.

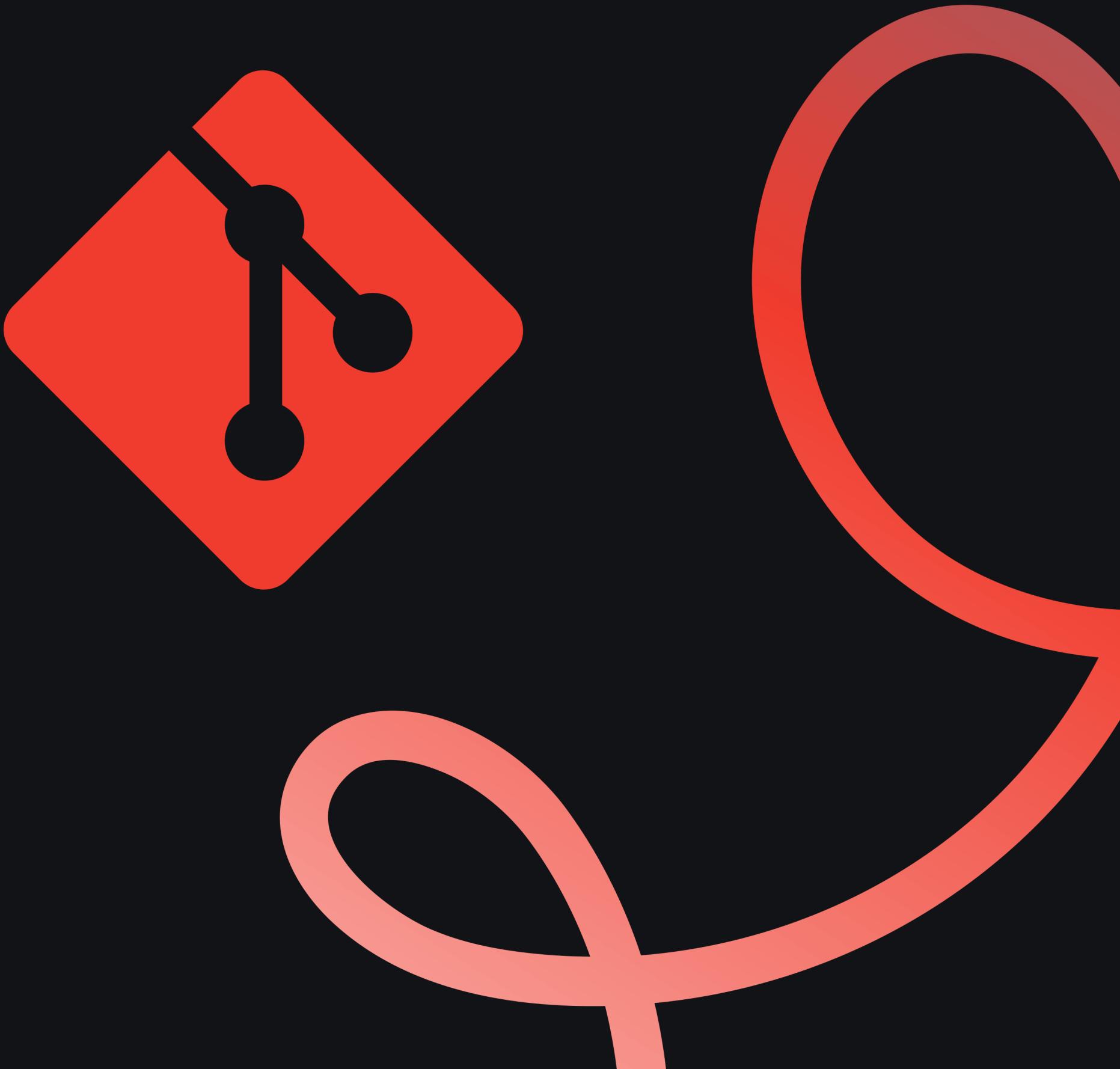


Advantages of Using Git Locally

Git allows you to track changes to your code over time.

It provides an efficient way to manage different versions of your project locally.

You can easily revert to previous versions or experiment with new features without affecting the main codebase.



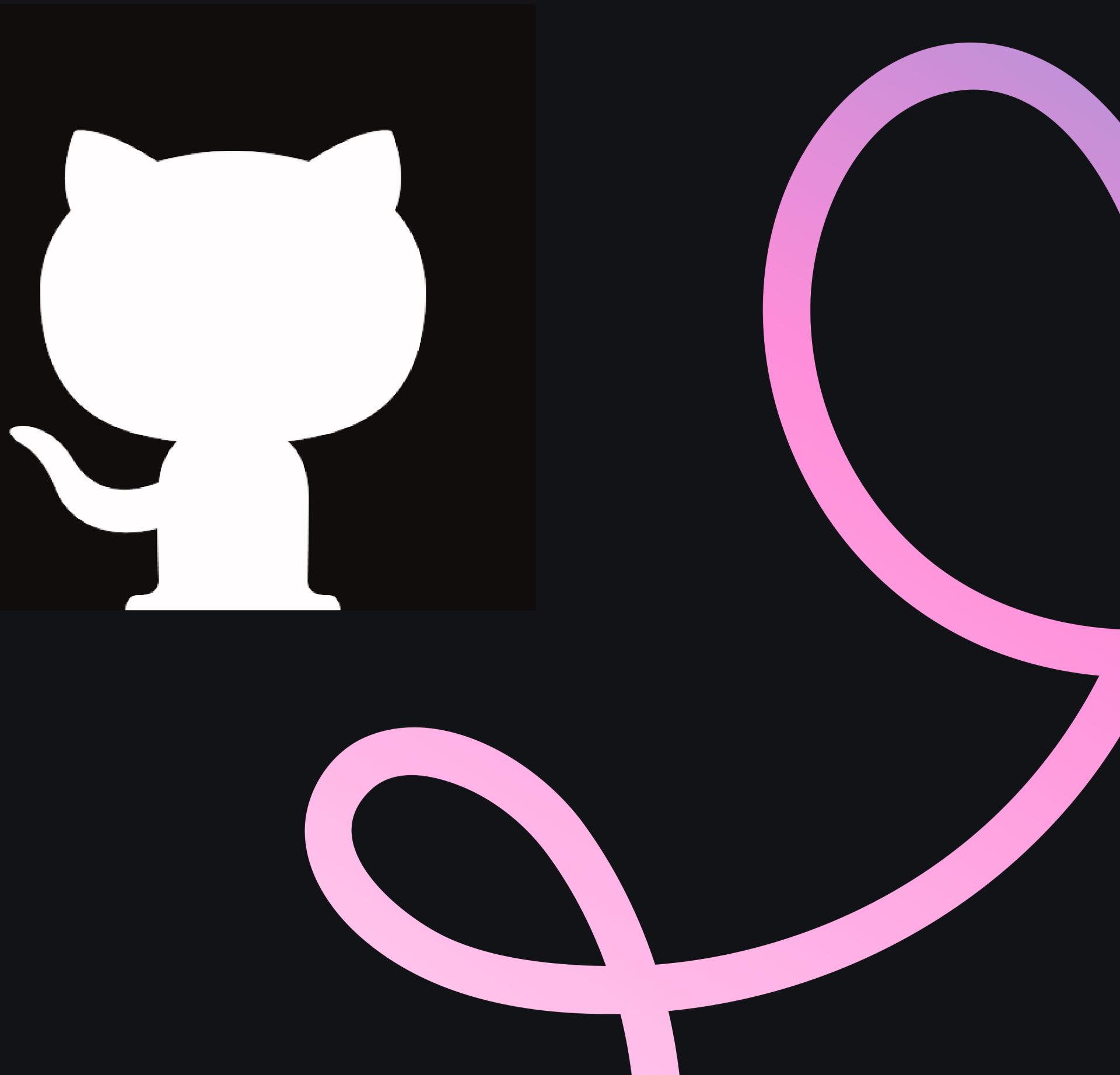
Advantages of Using GitHub

Collaboration: GitHub enables multiple developers to work together on the same project.

Version Control: It provides a complete timeline of changes, decisions, and project progression.

Code Sharing: GitHub allows you to share your code with others.

Features: GitHub offers additional tools like pull requests, code scanning, and secret scanning¹



Git Branches

Git branches allow developers to work on isolated changes without affecting the main codebase.

They provide an organized way to manage different features or bug fixes.

Common Git commands related to branches include creating, switching, merging, and deleting branches.



Pull Requests

A pull request (PR) is a way to propose changes from one branch (usually a feature branch) to another (often the main branch).

It facilitates code review, discussion, and collaboration before merging changes into the main codebase.



Protecting Project Code Stability

Use code reviews and pull requests to ensure quality.

Set up continuous integration (CI) to run automated tests.

Limit direct pushes to the main branch.

Implement branch protection rules to prevent accidental changes.



Working as a Team: Avoiding Bad Code

Regular code reviews and pair programming.

Clear coding standards and guidelines.

Automated testing and CI/CD pipelines.

Communication and collaboration within the team.



Protecting Public Code from Forked Repositories

Maintain a stable main branch.

Review and approve pull requests from forked repositories.

Use branch protection rules to prevent force pushes or destructive changes.

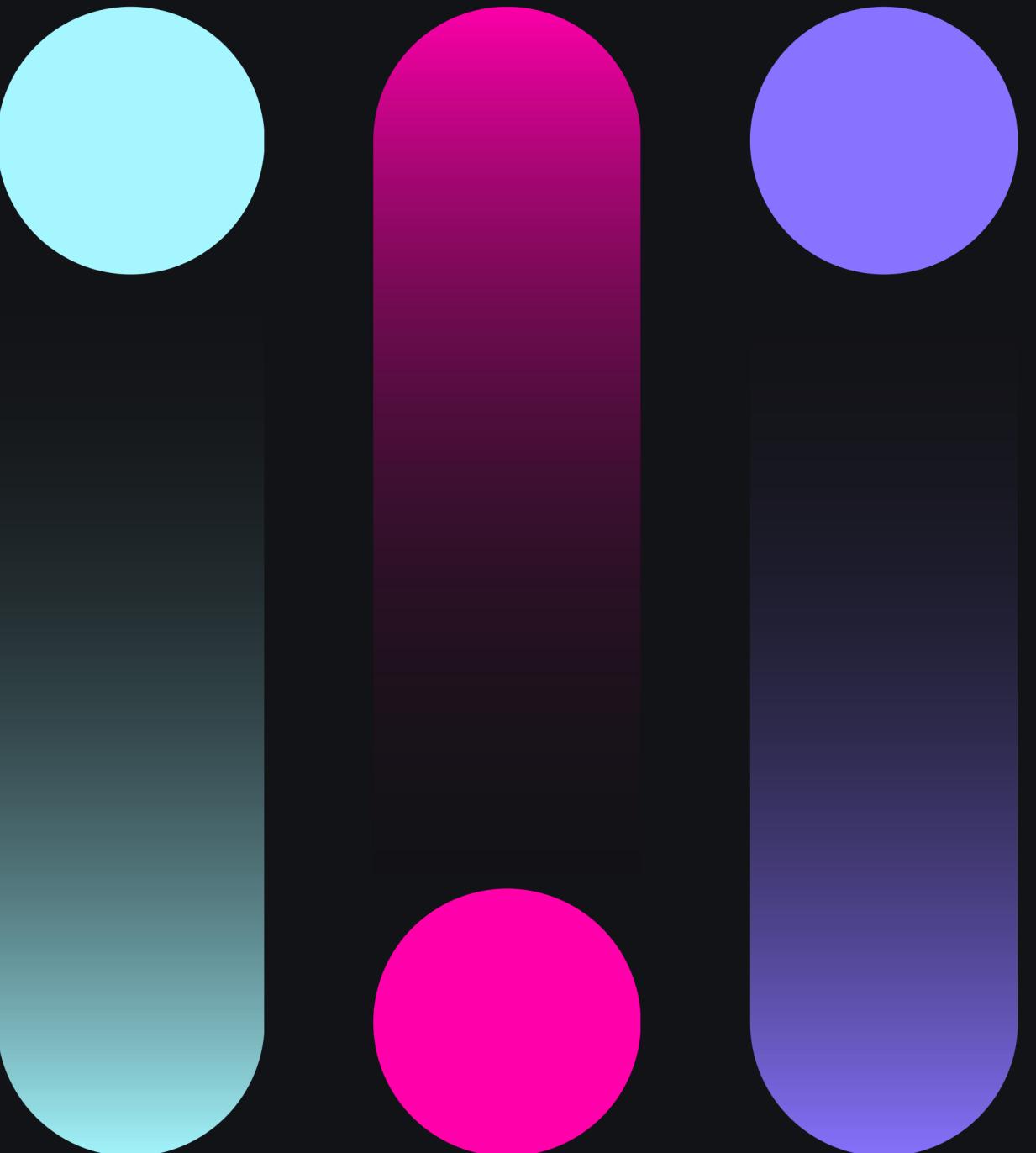


Alternatives to GitHub

GitLab

Bitbucket

AWS CodeCommit



References

- (1) Git - Wikipedia. <https://en.wikipedia.org/wiki/Git>.
- (2) GitHub - Wikipedia. <https://en.wikipedia.org/wiki/GitHub>.
- (3) Difference Between Git and GitHub - GeeksforGeeks. <https://www.geeksforgeeks.org/difference-between-git-and-github/>.
- (4) How to Use Git and GitHub – a Guide for Beginners and Experienced <https://www.freecodecamp.org/news/guide-to-git-github-for-beginners-and-experienced-devs/>.
- (5) What Is GitHub and Why Should You Use It? | Coursera. <https://www.coursera.org/articles/what-is-git>.
- (6) What is Git | Atlassian Git Tutorial. <https://www.atlassian.com/git/tutorials/what-is-git>.
- (7) Benefits of Using GitHub - DZone. <https://dzone.com/articles/benefits-of-using-github>.
- (8) GitHub: Let's build from here · GitHub. <https://github.com/>.
- (9) What is GitHub - W3Schools. <https://www.w3schools.com/whatis/whatis.github.asp>.
- (10) What Is GitHub? A Beginner's Introduction to GitHub - Kinsta. <https://kinsta.com/knowledgebase/what-is-github/>.
- (11) Should I use git if I'm just a single guy learning programming ... - Reddit. https://www.reddit.com/r/learnprogramming/comments/rlk5d9/should_i_use_git_if_im_just_a_single_guy_learning/.
- (12) About Git - GitHub Docs. <https://docs.github.com/en/get-started/using-git/about-git>.
- (13) Git - What is Git?. <https://git-scm.com/book/en/v2/Getting-Started-What-is-Git%3F>.
- (14) Why Use Git | Atlassian Git Tutorial. <https://www.atlassian.com/git/tutorials/why-git>.
- (15) What is Git? - Azure DevOps | Microsoft Learn. <https://learn.microsoft.com/en-us/devops/develop/git/what-is-git>.
- (16) Git | Overview | Workflow | Advantages | K21Academy. <https://k21academy.com/devops-job-bootcamp/git-overview-workflow-advantages/>.
- (17) A Beginner's Guide to Git vs GitHub - CareerFoundry. <https://careerfoundry.com/en/blog/web-development/git-vs-github/>.
- (18) Difference between Git and GitHub - Stack Overflow. <https://stackoverflow.com/questions/13321556/difference-between-git-and-github>.
- (19) How to Use Git and GitHub – Introduction for Beginners - freeCodeCamp.org. <https://www.freecodecamp.org/news/introduction-to-git-and-github/>.
- (20) About GitHub and Git - GitHub Docs. <https://docs.github.com/en/get-started/start-your-journey/about-github-and-git>.

Thank You

