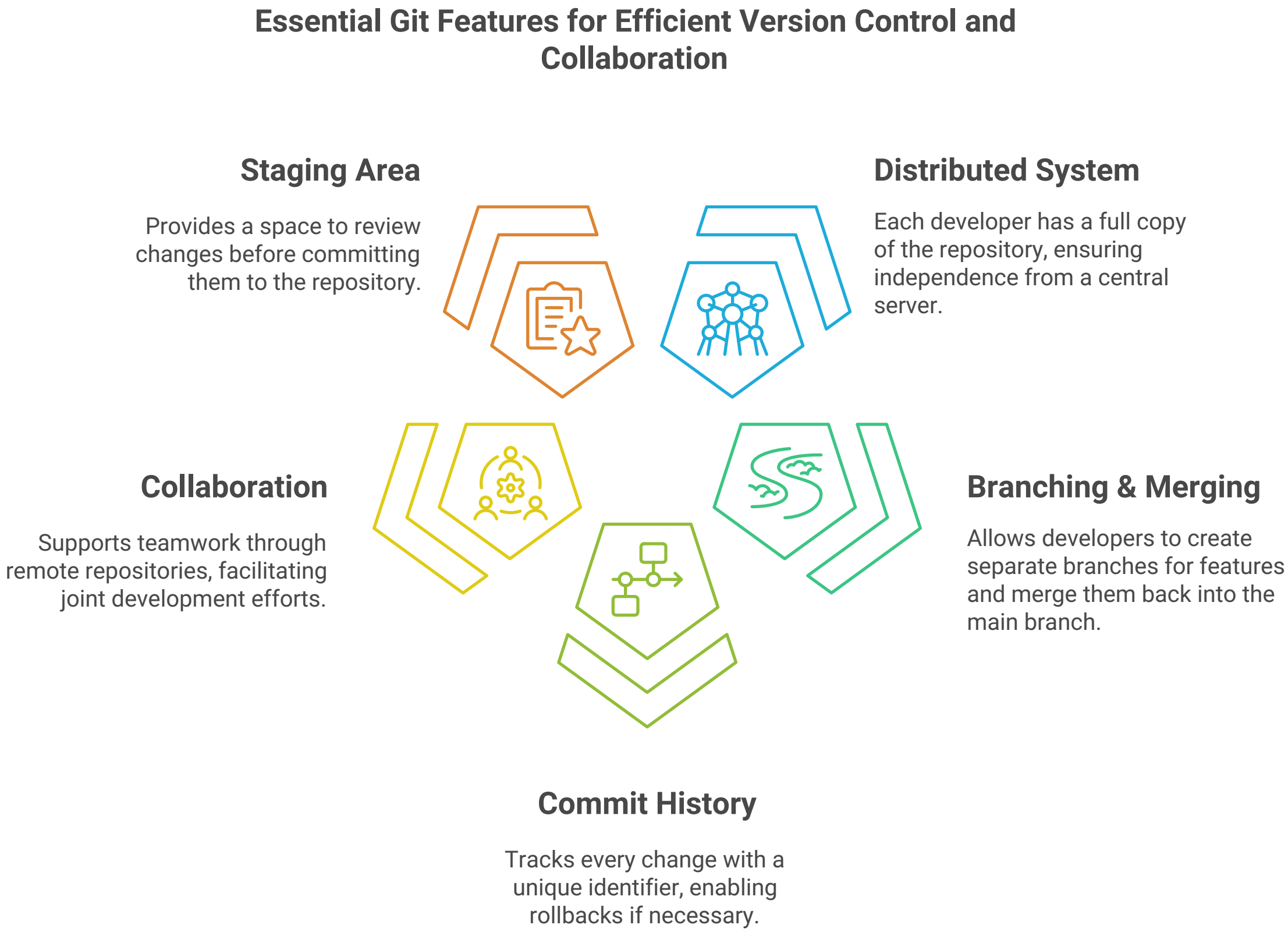


Understanding Git: A Comprehensive Overview

Git is a distributed version control system [VCS] that is essential for tracking changes in source code during software development. It highlights Git's key features, basic workflow, and its significance in collaborative development environments. By understanding Git, developers can enhance their productivity and streamline their coding processes.

Key Features of Git:

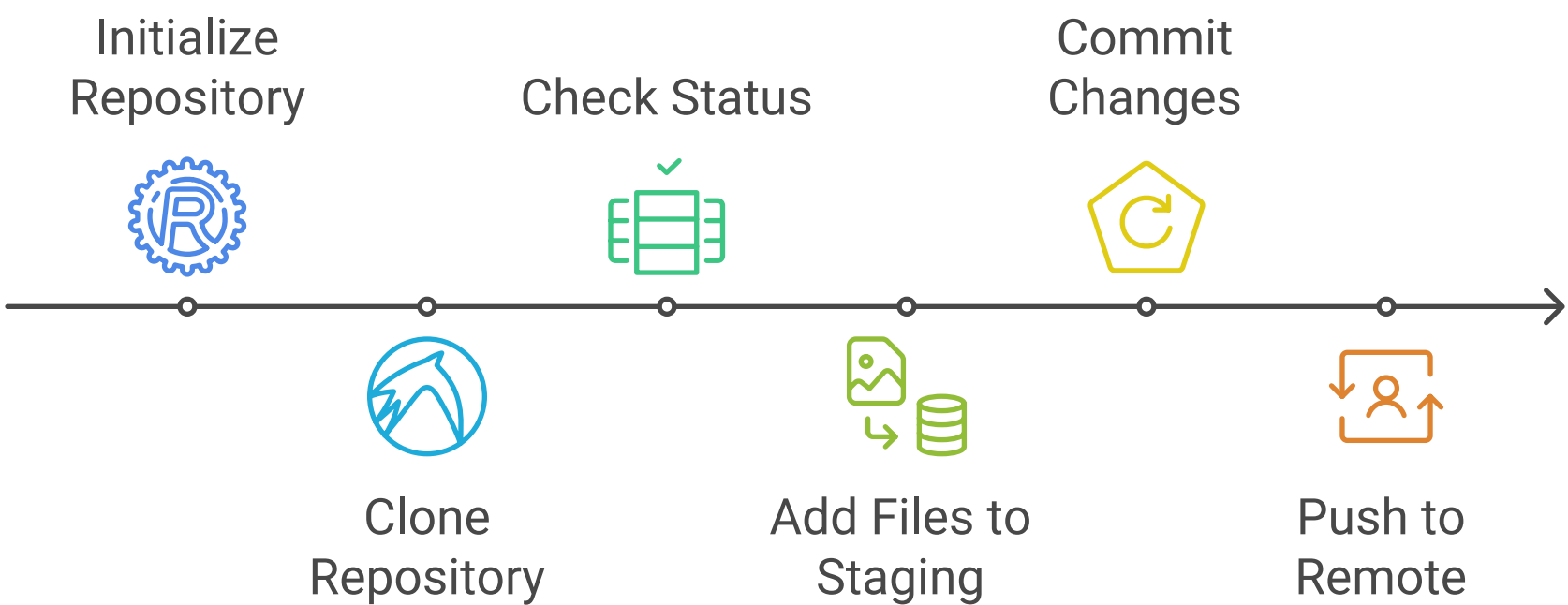
- Distributed System** – Every developer has a full copy of the repository, making it independent of a central server.
- Branching & Merging** – Developers can create separate branches for new features and merge them back into the main branch.
- Commit History** – Tracks every change with a unique identifier (SHA hash), allowing rollbacks if needed.
- Collaboration** – Supports teamwork via remote repositories [e.g., GitHub, GitLab, Bitbucket].
- Staging Area** – Changes can be reviewed before committing them to the repository.
- Efficient Storage** – Uses compression and delta encoding to store changes efficiently.



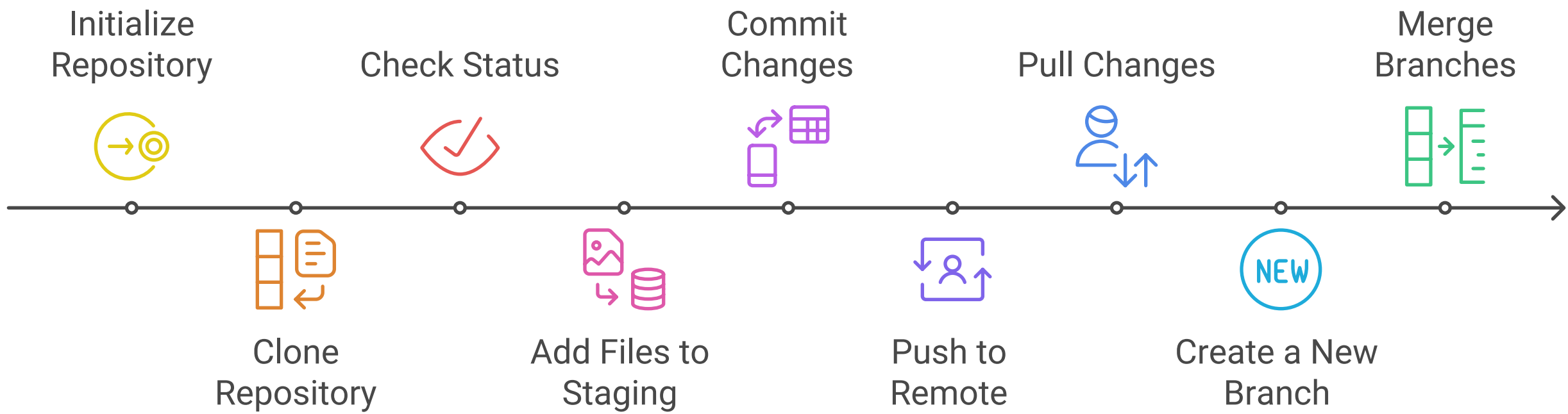
Basic Git Workflow:

- Initialize a Repository** – `git init`
- Clone a Repository** – `git clone <repo_url>`
- Check Status** – `git status`
- Add Files to Staging** – `git add <file>`
- Commit Changes** – `git commit -m "Message"`
- Push to Remote** – `git push origin <branch>`

Basic Git Workflow Sequence



Basic Git Workflow Sequence



Git is widely used in software development, DevOps, and version control, making it an essential tool for developers. Understanding its features and workflow can significantly improve collaboration and efficiency in coding projects.

