Exploring Git Collaboration

Aziz Zai

24.04.2024

IT'S YOUR TURN!!!

1. How can you use Git as a collaboration platform?

- **Version Control**: We use Git in our product's frontend as a version control system to manage changes to our codebase.
- **Branching**: We work on different branches, each specific to a feature, bug fix, or refactoring action, which helps isolate changes and manage development flow.
- **Remote Repositories**: We use GitHub for hosting our remote repositories, enabling centralized access and collaboration.
- **Hooks**: We utilize Git commit and push hooks to perform checks before committing and pushing code, ensuring code quality and consistency.

2. What online platforms host Git?

- The main platforms that host Git repositories include:
  - GitHub
  - GitLab
  - Bitbucket
  - Azure Repos
  - AWS CodeCommit

3. Are there differences between online platforms?

- **Yes, significant differences exist between online platforms**:
  - Features vary widely: For instance, GitHub offers GitHub Actions for continuous integration and deployment.
  - Integration capabilities differ: Each platform has its strengths depending on the ecosystem (e.g., GitHub with Microsoft, GitLab with Kubernetes).
  - Pricing structures and plans can differ, affecting the choice depending on organizational needs and budget.

4. Add your findings to your script and commit the changes.

- **Action Item**: Incorporate this information into your project script.
- **Commit**: Use Git commands to stage (`git add`), commit (`git commit`), and push (`git push`) the updates to your remote repository.