GitHub

설명:

GitHub는 가장 널리 사용되는 Git 호스팅 서비스 중 하나로, 오픈 소스 프로젝트와 상업용 프로젝트를 관리하는 데 널리 사용됩니다. GitHub는 코드 호스팅뿐만 아니라 협업 도구, 코드 검토, 이슈 트래킹, 프로젝트 관리 기능 등을 제공합니다.

장점:

큰 커뮤니티와 오픈 소스 프로젝트: GitHub에는 수많은 오픈 소스 프로젝트가 있으며, 개발자 커뮤니티가 매우 활발합니다.

GitHub Actions: CI/CD 파이프라인을 쉽게 설정할 수 있습니다.

풍부한 통합: 다양한 개발 도구 및 서비스와의 통합이 잘 되어 있습니다.

사용자 친화적 인터페이스: 직관적인 UI로 사용하기 쉽습니다.

단점:

비용: 프리미엄 기능과 대규모 팀을 위한 요금제가 비쌀 수 있습니다.

프라이빗 리포지토리 제한: 무료 요금제에서 프라이빗 리포지토리 사용 시 제한이 있을 수 있습니다.

기업 맞춤형 기능 제한: GitHub Enterprise가 필요한 경우 비용이 상당히 높아질 수 있습니다.

GitLab

설명:

GitLab은 Git 리포지토리 관리 외에도 DevOps 기능을 포함한 종합적인 소프트웨어 개발 플랫폼을 제공합니다. 자체 호스팅이 가능하며, CI/CD, 코드 리뷰, 이슈 관리 등의 기능이 포함되어 있습니다.

장점:

CI/CD 통합: GitLab CI/CD는 매우 강력하고 사용하기 쉽습니다.

자체 호스팅 가능: 온프레미스 설치가 가능하여 자체 서버에 호스팅할 수 있습니다.

종합적인 DevOps 툴: DevOps 라이프사이클 전체를 관리할 수 있는 기능 제공.

오픈 소스: GitLab Community Edition은 오픈 소스입니다.

단점:

복잡성: 기능이 많아서 처음 사용자가 익숙해지기까지 시간이 걸릴 수 있습니다.

성능 문제: 대규모 프로젝트에서는 성능 문제가 발생할 수 있습니다.

비용: 완전한 기능을 이용하려면 프리미엄 플랜이 필요합니다.

Bitbucket

설명:

Bitbucket은 Atlassian에서 제공하는 Git 호스팅 서비스로, 특히 Jira, Confluence와 같은 Atlassian 제품과의 통합이 강점입니다. Bitbucket은 Git 및 Mercurial 리포지토리를 모두 지원합니다.

장점:

Jira 통합: Atlassian의 Jira와의 원활한 통합으로 이슈 트래킹과 프로젝트 관리를 효율적으로 할 수 있습니다.

프라이빗 리포지토리: 무료 요금제에서 무제한 프라이빗 리포지토리를 제공합니다.

파이프라인: Bitbucket Pipelines를 사용하여 CI/CD를 쉽게 설정할 수 있습니다.

단점:

커뮤니티와 생태계: GitHub에 비해 상대적으로 작은 커뮤니티.

사용자 인터페이스: 일부 사용자에게는 UI가 직관적이지 않을 수 있습니다.

성능: 큰 프로젝트에서는 성능 문제가 있을 수 있습니다.

2번 깃 사용법

1. cd Working

git init

1. git config --global user.name "Yuhan"
2. git config --global user.email [your\_email@example.com](mailto:your_email@example.com)
3. git config --global core.quotepath false
4. git add . # 모든 변경 파일을 스테이징
5. git commit -m "Initial commit" # 커밋 메시지와 함께 커밋
6. git status # 현재 상태 확인
7. git log # 커밋 기록 확인
8. git remote add origin <https://github.com/Yuhan/Repo.git>
9. git reset, git revert, 그리고 git checkout 명령어를 사용하여 커밋을 되돌린다
10. git remote add origin <https://github.com/Yuhan/Repo.git>
11. git push -u origin master
12. git branch new-branch # 새로운 브랜치 생성
13. git checkout new-branch # 생성한 브랜치로 전환
14. git clone 원격 저장소 주소
15. git pull

마지막 브런치 합병 방법

git checkout main

git merge feature-branch

git checkout main

git merge feature-branch