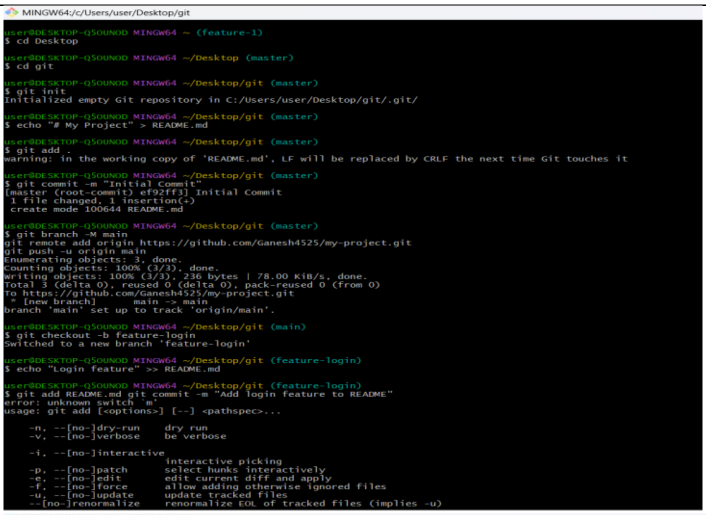
VIT Vidyalankar Institute of Technology ACCREDITED A+ BY NAAC	**Department of Information Technology**

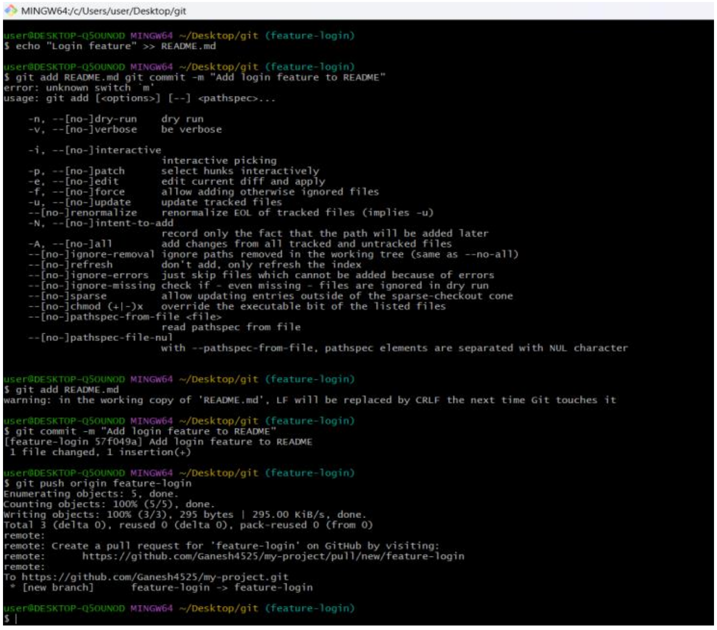
Semester	T.E. Semester VI – INFT
Subject	DevOps Lab
Laboratory Teacher:	Prof. Rohit Barve
Laboratory	CC02

Student Name	Ganesh Waghmare	
Roll Number	22101A0014	
Grade and Subject Teacher's Signature		

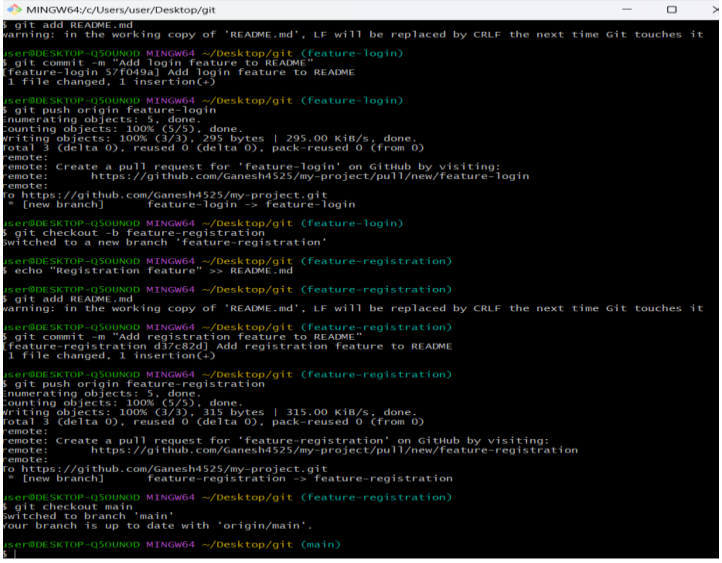
Experiment Number	1	
Experiment Title	Study Git Hub and Git Bash	
Problem Statement		
Resources / Apparatus Required	Hardware: Desktop/Laptop	Software: Git Hub and Git Bash



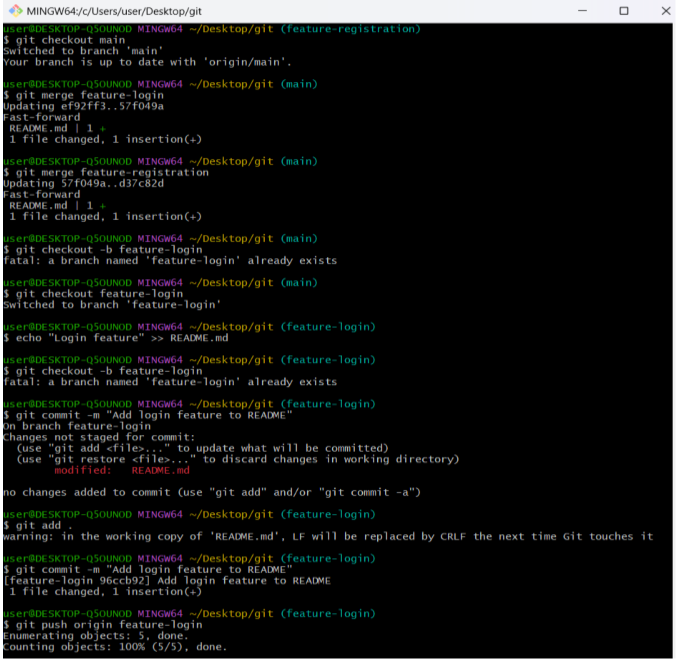
```
user@DESKTOP-Q5OUNOD MINGW64 ~ (feature-1)
$ cd Desktop

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop (master)
$ cd git

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (master)
$ git init
Initialized empty Git repository in C:/Users/user/Desktop/git/.git/

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (master)
$ echo "# My Project" > README.md

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (master)
$ git add .
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (master)
$ git commit -m "Initial Commit"
[master (root-commit) ef92ff3] Initial Commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (master)
$ git branch -M main
git remote add origin https://github.com/Ganesh4525/my-project.git
git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 236 bytes | 78.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Ganesh4525/my-project.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git checkout -b feature-login
Switched to a new branch 'feature-login'

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ echo "Login feature" >> README.md

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git add README.md git commit -m "Add login feature to README"
error: unknown switch `m'
usage: git add [<options>] [--] <pathspec>...

    -n, --[no-]dry-run    dry run
    -v, --[no-]verbose    be verbose

    -i, --[no-]interactive
                          interactive picking
    -p, --[no-]patch      select hunks interactively
    -e, --[no-]edit       edit current diff and apply
    -f, --[no-]force      allow adding otherwise ignored files
    -u, --[no-]update     update tracked files
    --[no-]renormalize    renormalize EOL of tracked files (implies -u)
```



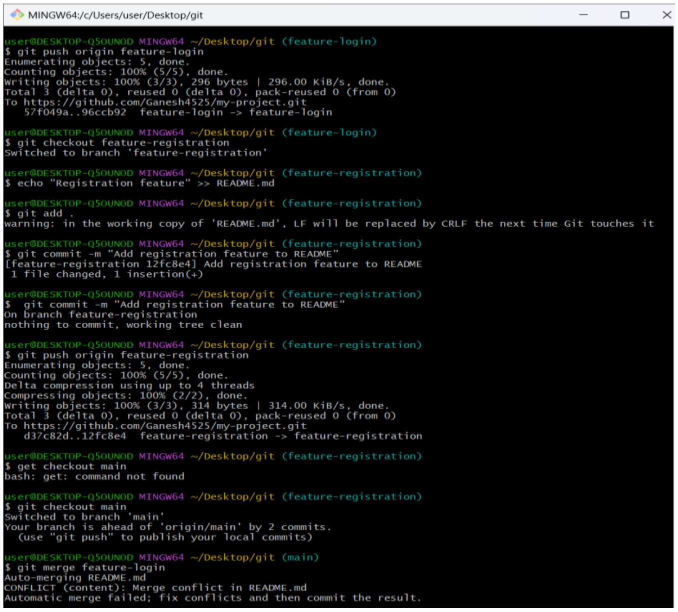
```
user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ echo "Login feature" >> README.md

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git add README.md git commit -m "Add login feature to README"
error: unknown switch `m'
usage: git add [<options>] [--] <pathspec>...

    -n, --[no-]dry-run    dry run
    -v, --[no-]verbose    be verbose

    -i, --[no-]interactive
                          interactive picking
    -p, --[no-]patch      select hunks interactively
    -e, --[no-]edit       edit current diff and apply
    -f, --[no-]force      allow adding otherwise ignored files
    -u, --[no-]update     update tracked files
    --[no-]renormalize    renormalize EOL of tracked files (implies -u)
    -N, --[no-]intent-to-add
                          record only the fact that the path will be added later
    -A, --[no-]all        add changes from all tracked and untracked files
    --[no-]ignore-removal ignore paths removed in the working tree (same as --no-all)
    --[no-]refresh        don't add, only refresh the index
    --[no-]ignore-errors  just skip files which cannot be added because of errors
    --[no-]ignore-missing check if - even missing - files are ignored in dry run
    --[no-]sparse         allow updating entries outside of the sparse-checkout cone
    --[no-]chmod (+|-)x   override the executable bit of the listed files
    --[no-]pathspec-from-file <file>
                          read pathspec from file
    --[no-]pathspec-file-nul
                          with --pathspec-from-file, pathspec elements are separated with NUL character


user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git commit -m "Add login feature to README"
[feature-login 57f049a] Add login feature to README
 1 file changed, 1 insertion(+)

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git push origin feature-login
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 295 bytes | 295.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature-login' on GitHub by visiting:
remote:      https://github.com/Ganesh4525/my-project/pull/new/feature-login
remote:
To https://github.com/Ganesh4525/my-project.git
 * [new branch]      feature-login -> feature-login

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ |
```



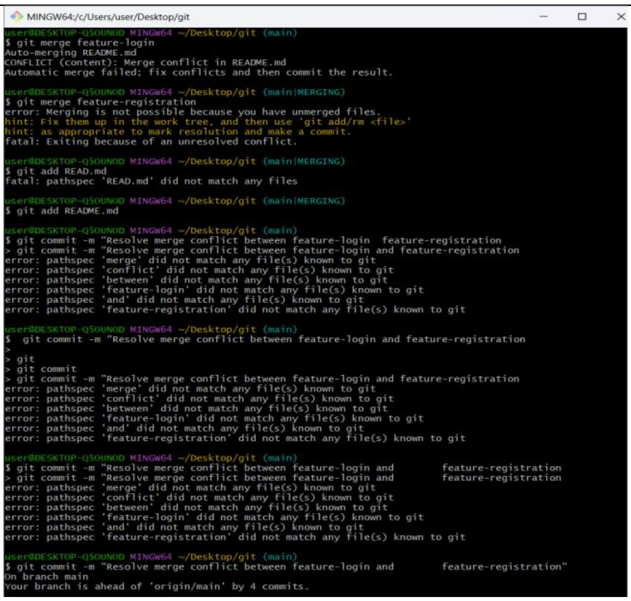
```
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git commit -m "Add login feature to README"
[feature-login 57f049a] Add login feature to README
 1 file changed, 1 insertion(+)

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git push origin feature-login
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 295 bytes | 295.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature-login' on GitHub by visiting:
remote:      https://github.com/Ganesh4525/my-project/pull/new/feature-login
remote:
To https://github.com/Ganesh4525/my-project.git
 * [new branch]      feature-login -> feature-login

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git checkout -b feature-registration
Switched to a new branch 'feature-registration'

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ echo "Registration feature" >> README.md

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ git commit -m "Add registration feature to README"
[feature-registration d37c82d] Add registration feature to README
 1 file changed, 1 insertion(+)

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ git push origin feature-registration
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 315 bytes | 315.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature-registration' on GitHub by visiting:
remote:      https://github.com/Ganesh4525/my-project/pull/new/feature-registration
remote:
To https://github.com/Ganesh4525/my-project.git
 * [new branch]      feature-registration -> feature-registration

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ |
```



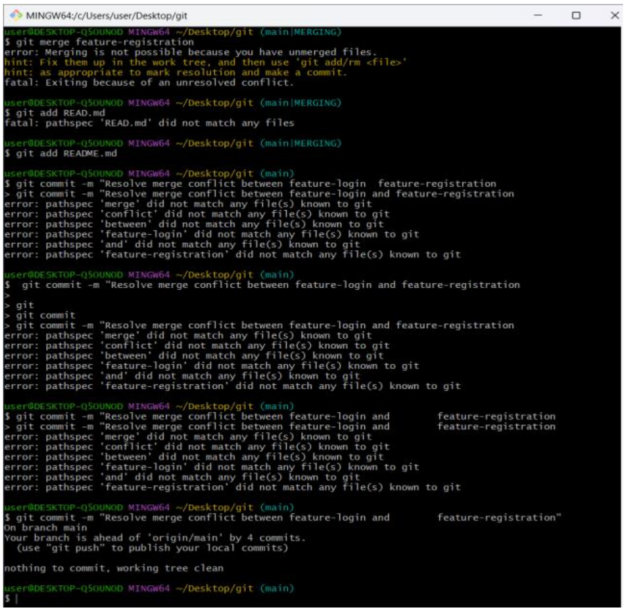
```
user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git merge feature-login
Updating ef92ff3..57f049a
Fast-forward
 README.md | 1 +
 1 file changed, 1 insertion(+)

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git merge feature-registration
Updating 57f049a..d37c82d
Fast-forward
 README.md | 1 +
 1 file changed, 1 insertion(+)

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git checkout -b feature-login
fatal: a branch named 'feature-login' already exists

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git checkout feature-login
Switched to branch 'feature-login'

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ echo "Login feature" >> README.md

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git checkout -b feature-login
fatal: a branch named 'feature-login' already exists

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git commit -m "Add login feature to README"
On branch feature-login
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   README.md

no changes added to commit (use "git add" and/or "git commit -a")

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git add .
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git commit -m "Add login feature to README"
[feature-login 96ccb92] Add login feature to README
 1 file changed, 1 insertion(+)

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git push origin feature-login
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
```



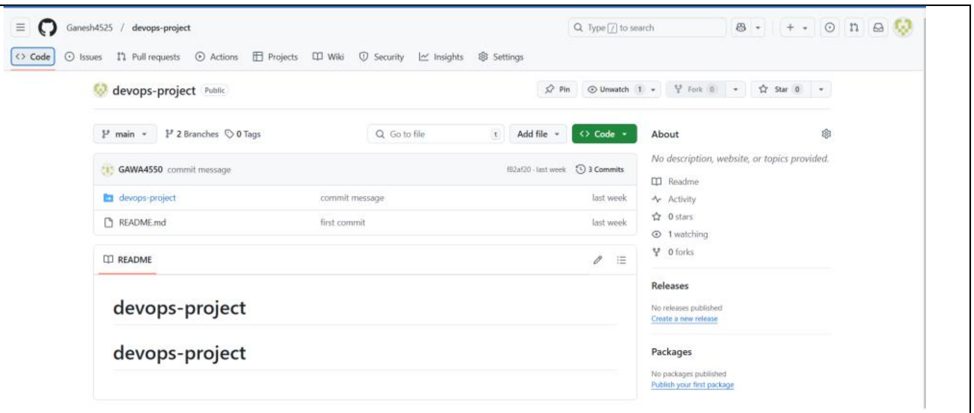
```
user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git push origin feature-login
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 296 bytes | 296.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Ganesh4525/my-project.git
   57f049a..96ccb92  feature-login -> feature-login

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-login)
$ git checkout feature-registration
Switched to branch 'feature-registration'

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ echo "Registration feature" >> README.md

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ git add .
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ git commit -m "Add registration feature to README"
[feature-registration 12fc8e4] Add registration feature to README
 1 file changed, 1 insertion(+)

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$  git commit -m "Add registration feature to README"
On branch feature-registration
nothing to commit, working tree clean

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ git push origin feature-registration
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 314 bytes | 314.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Ganesh4525/my-project.git
   d37c82d..12fc8e4  feature-registration -> feature-registration

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ get checkout main
bash: get: command not found

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (feature-registration)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git merge feature-login
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
```



```
user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git merge feature-login
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main|MERGING)
$ git merge feature-registration
error: Merging is not possible because you have unmerged files.
hint: Fix them up in the work tree, and then use 'git add/rm <file>'
hint: as appropriate to mark resolution and make a commit.
fatal: Exiting because of an unresolved conflict.

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main|MERGING)
$ git add READ.md
fatal: pathspec 'READ.md' did not match any files

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main|MERGING)
$ git add README.md

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git commit -m "Resolve merge conflict between feature-login  feature-registration
> git commit -m "Resolve merge conflict between feature-login and feature-registration
error: pathspec 'merge' did not match any file(s) known to git
error: pathspec 'conflict' did not match any file(s) known to git
error: pathspec 'between' did not match any file(s) known to git
error: pathspec 'feature-login' did not match any file(s) known to git
error: pathspec 'and' did not match any file(s) known to git
error: pathspec 'feature-registration' did not match any file(s) known to git

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$  git commit -m "Resolve merge conflict between feature-login and feature-registration
>
> git
> git commit
> git commit -m "Resolve merge conflict between feature-login and feature-registration
error: pathspec 'merge' did not match any file(s) known to git
error: pathspec 'conflict' did not match any file(s) known to git
error: pathspec 'between' did not match any file(s) known to git
error: pathspec 'feature-login' did not match any file(s) known to git
error: pathspec 'and' did not match any file(s) known to git
error: pathspec 'feature-registration' did not match any file(s) known to git

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git commit -m "Resolve merge conflict between feature-login and        feature-registration
> git commit -m "Resolve merge conflict between feature-login and        feature-registration
error: pathspec 'merge' did not match any file(s) known to git
error: pathspec 'conflict' did not match any file(s) known to git
error: pathspec 'between' did not match any file(s) known to git
error: pathspec 'feature-login' did not match any file(s) known to git
error: pathspec 'and' did not match any file(s) known to git
error: pathspec 'feature-registration' did not match any file(s) known to git

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git commit -m "Resolve merge conflict between feature-login and        feature-registration"
On branch main
Your branch is ahead of 'origin/main' by 4 commits.
```



```
user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main|MERGING)
$ git merge feature-registration
error: Merging is not possible because you have unmerged files.
hint: Fix them up in the work tree, and then use 'git add/rm <file>'
hint: as appropriate to mark resolution and make a commit.
fatal: Exiting because of an unresolved conflict.

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main|MERGING)
$ git add READ.md
fatal: pathspec 'READ.md' did not match any files

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main|MERGING)
$ git add README.md

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git commit -m "Resolve merge conflict between feature-login  feature-registration
> git commit -m "Resolve merge conflict between feature-login and feature-registration
error: pathspec 'merge' did not match any file(s) known to git
error: pathspec 'conflict' did not match any file(s) known to git
error: pathspec 'between' did not match any file(s) known to git
error: pathspec 'feature-login' did not match any file(s) known to git
error: pathspec 'and' did not match any file(s) known to git
error: pathspec 'feature-registration' did not match any file(s) known to git

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$  git commit -m "Resolve merge conflict between feature-login and feature-registration
>
> git
> git commit
> git commit -m "Resolve merge conflict between feature-login and feature-registration
error: pathspec 'merge' did not match any file(s) known to git
error: pathspec 'conflict' did not match any file(s) known to git
error: pathspec 'between' did not match any file(s) known to git
error: pathspec 'feature-login' did not match any file(s) known to git
error: pathspec 'and' did not match any file(s) known to git
error: pathspec 'feature-registration' did not match any file(s) known to git

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git commit -m "Resolve merge conflict between feature-login and        feature-registration
> git commit -m "Resolve merge conflict between feature-login and        feature-registration
error: pathspec 'merge' did not match any file(s) known to git
error: pathspec 'conflict' did not match any file(s) known to git
error: pathspec 'between' did not match any file(s) known to git
error: pathspec 'feature-login' did not match any file(s) known to git
error: pathspec 'and' did not match any file(s) known to git
error: pathspec 'feature-registration' did not match any file(s) known to git

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ git commit -m "Resolve merge conflict between feature-login and        feature-registration"
On branch main
Your branch is ahead of 'origin/main' by 4 commits.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/git (main)
$ |
```



devops-project Public

GAWA4550 commit message		f82af20 · last week 3 Commits
devops-project	commit message	last week
README.md	first commit	last week

README

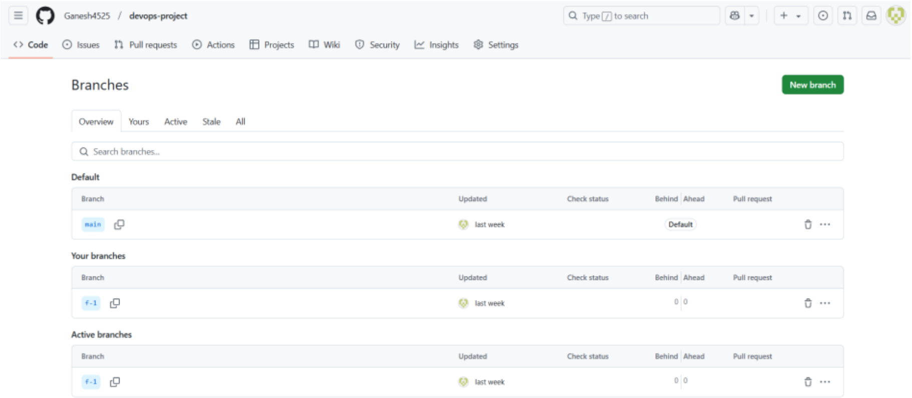
devops-project

devops-project

No description, website, or topics provided.

No releases published
Create a new release

No packages published
Publish your first package



Default

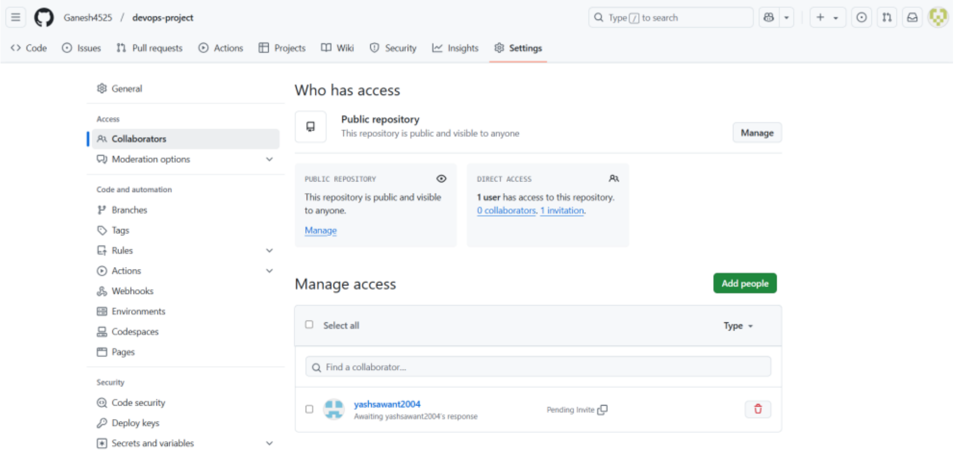
Branch	Updated	Check status	Behind | Ahead	Pull request
main	last week		Default	

Your branches

Branch	Updated	Check status	Behind | Ahead	Pull request
f-1	last week		0 | 0	

Active branches

Branch	Updated	Check status	Behind | Ahead	Pull request
f-1	last week		0 | 0	



Who has access

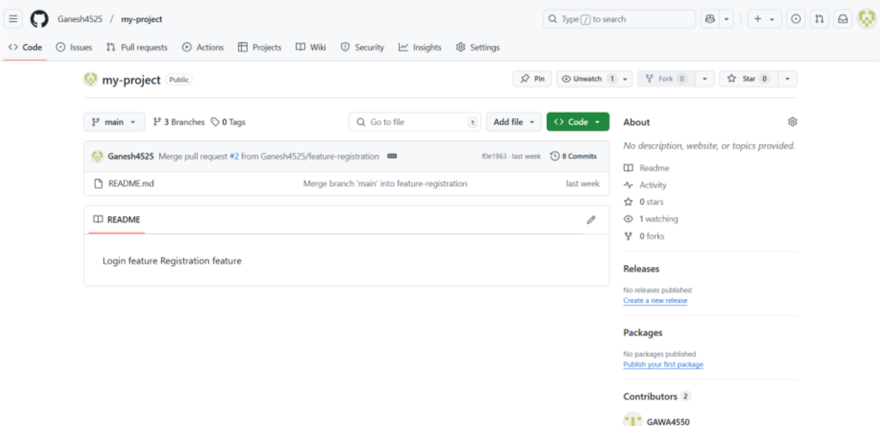
Public repository
This repository is public and visible to anyone

PUBLIC REPOSITORY
This repository is public and visible to anyone.
Manage

DIRECT ACCESS
1 user has access to this repository. 0 collaborators. 1 invitation.

Manage access

□ yashsawant2004 Awaiting yashsawant2004's response Pending Invite



my-project Public

Ganesh4525 Merge pull request #2 from Ganesh4525/feature-registration		f0e1963 · last week 8 Commits
README.md	Merge branch 'main' into feature-registration	last week

README

Login feature Registration feature

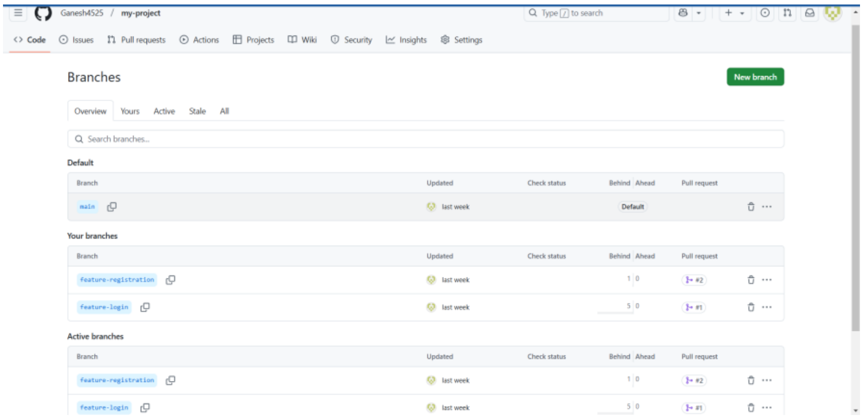
No description, website, or topics provided.

No releases published
Create a new release

No packages published
Publish your first package

Contributors 2

GAWA4550



Ganesh4525 / my-project

Branches

Default

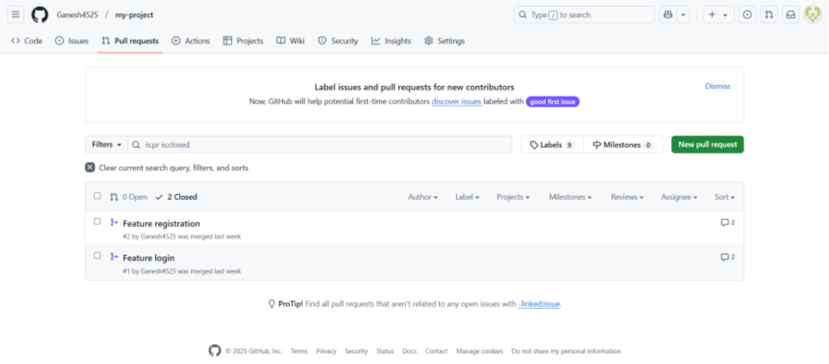
Branch	Updated	Check status	Behind | Ahead	Pull request
`main`	last week		Default	

Your branches

Branch	Updated	Check status	Behind | Ahead	Pull request
`feature-registration`	last week		1 | 0	#2
`feature-login`	last week		5 | 0	#1

Active branches

Branch	Updated	Check status	Behind | Ahead	Pull request
`feature-registration`	last week		1 | 0	#2
`feature-login`	last week		5 | 0	#1



Ganesh4525 / my-project

Label issues and pull requests for new contributors

Now, GitHub will help potential first-time contributors discover issues labeled with good first issue

is:pr is:closed

Clear current search query, filters, and sorts

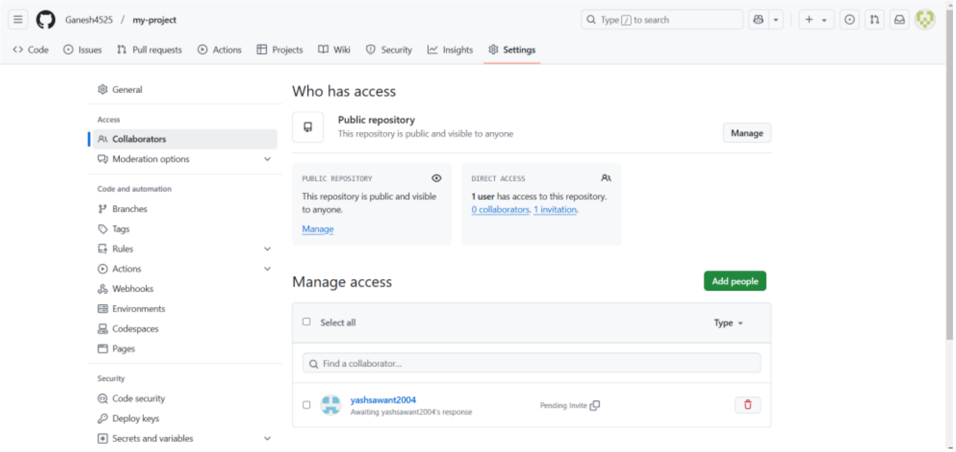
Feature registration 2

#2 by Ganesh4525 was merged last week

Feature login 2

#1 by Ganesh4525 was merged last week

ProTip! Find all pull requests that aren't related to any open issues with -linked:issue.



Ganesh4525 / my-project

Who has access

Public repository

This repository is public and visible to anyone

PUBLIC REPOSITORY

This repository is public and visible to anyone.

Manage

DIRECT ACCESS

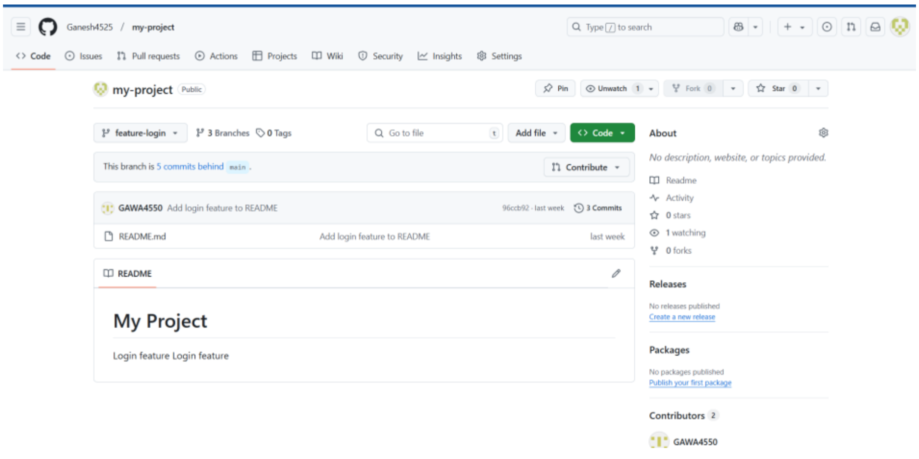
1 user has access to this repository. 0 collaborators. 1 invitation.

Manage access

yashsawant2004

Awaiting yashsawant2004's response

Pending Invite



Ganesh4525 / my-project

my-project Public

This branch is 5 commits behind `main` .

GAWA4550 Add login feature to README 96ccb92 · last week 3 Commits

README.md Add login feature to README last week

README

My Project

Login feature Login feature

About

No description, website, or topics provided.

Readme

Activity

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

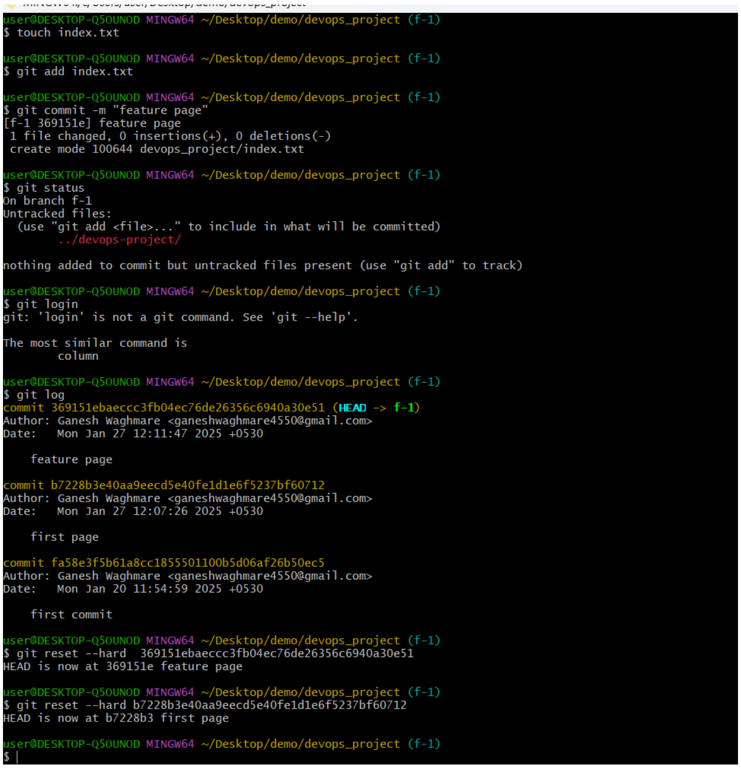
Packages

No packages published

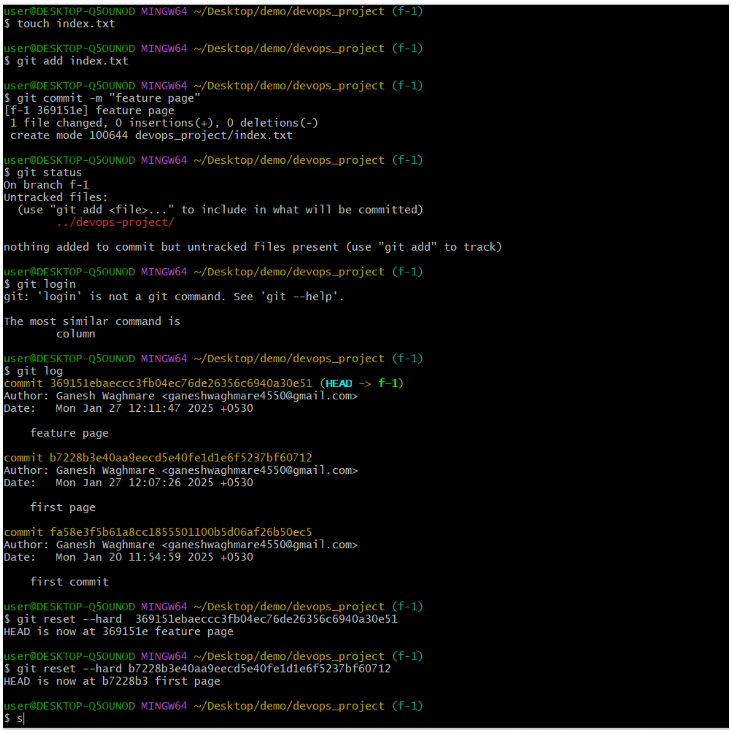
Publish your first package

Contributors 2

GAWA4550



```
user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ touch index.txt

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git add index.txt

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git commit -m "feature page"
[f-1 369151e] feature page
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 devops_project/index.txt

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git status
On branch f-1
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ../devops-project/

nothing added to commit but untracked files present (use "git add" to track)

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git login
git: 'login' is not a git command. See 'git --help'.

The most similar command is
        column

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git log
commit 369151ebaeccc3fb04ec76de26356c6940a30e51 (HEAD -> f-1)
Author: Ganesh Waghmare <ganeshwaghmare4550@gmail.com>
Date:   Mon Jan 27 12:11:47 2025 +0530

    feature page

commit b7228b3e40aa9eecd5e40fe1d1e6f5237bf60712
Author: Ganesh Waghmare <ganeshwaghmare4550@gmail.com>
Date:   Mon Jan 27 12:07:26 2025 +0530

    first page

commit fa58e3f5b61a8cc1855501100b5d06af26b50ec5
Author: Ganesh Waghmare <ganeshwaghmare4550@gmail.com>
Date:   Mon Jan 20 11:54:59 2025 +0530

    first commit

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git reset --hard  369151ebaeccc3fb04ec76de26356c6940a30e51
HEAD is now at 369151e feature page

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git reset --hard b7228b3e40aa9eecd5e40fe1d1e6f5237bf60712
HEAD is now at b7228b3 first page

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ |
```



```
user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ touch index.txt

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git add index.txt

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git commit -m "feature page"
[f-1 369151e] feature page
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 devops_project/index.txt

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git status
On branch f-1
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ../devops-project/

nothing added to commit but untracked files present (use "git add" to track)

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git login
git: 'login' is not a git command. See 'git --help'.

The most similar command is
        column

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git log
commit 369151ebaeccc3fb04ec76de26356c6940a30e51 (HEAD -> f-1)
Author: Ganesh Waghmare <ganeshwaghmare4550@gmail.com>
Date:   Mon Jan 27 12:11:47 2025 +0530

    feature page

commit b7228b3e40aa9eecd5e40fe1d1e6f5237bf60712
Author: Ganesh Waghmare <ganeshwaghmare4550@gmail.com>
Date:   Mon Jan 27 12:07:26 2025 +0530

    first page

commit fa58e3f5b61a8cc1855501100b5d06af26b50ec5
Author: Ganesh Waghmare <ganeshwaghmare4550@gmail.com>
Date:   Mon Jan 20 11:54:59 2025 +0530

    first commit

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git reset --hard  369151ebaeccc3fb04ec76de26356c6940a30e51
HEAD is now at 369151e feature page

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ git reset --hard b7228b3e40aa9eecd5e40fe1d1e6f5237bf60712
HEAD is now at b7228b3 first page

user@DESKTOP-Q5OUNOD MINGW64 ~/Desktop/demo/devops_project (f-1)
$ s|
```

Conclusion:	Using GitHub for collaborative project management helps teams work efficiently by organizing code, tracking changes, and resolving conflicts smoothly. With branches, pull requests, and proper commit history, teams can ensure a clean and structured workflow, making development more manageable and teamwork more effective.