



**MANAGING CODE USING BITBUCKET GUI**

**MANAGING CODE USING BITBUCKET GUI**

*CONTENTS:*

1. About ………………….………………………………………………………………………………………………………………….1
2. Cloning a repository ………………………………………………………………………………………………………………. 1
3. Creating branch.………………………………………………………………………………………………………............... 1
4. File creating, Add, Commit, Push to repository ………………………………………………………................. 2
5. Editing and Committing File using Bitbucket GUI ……………………………………………………….............. 4
6. Checking difference from previous file ………………………………………………………………….................... 5
7. Change History ………………………………….......................................................................................... 5
8. Checking difference from previous file ………………………………………………………………….................... 5
9. Viewing the file in RAW format ..……….......................................................................................... 6
10. Bitbucket Blame ………………………………......................................................................................... 6
11. Raising Pull Request ………………………….………………………………………………………………….................... 6
12. Approving Pull Request …………………….......................................................................................... 8
13. Rejecting Pull Request ……………………….………………………………………………………………….................. 10
14. Reviewing the difference ………...………........................................................................................ 10
15. Merging the Pull request ….………………....................................................................................... 11
16. Declining Pull Request ……………………….………………………………………………………………….................. 11
17. Forking a Repository …………...………............................................................................................ 12
18. Commits Navigation ………….………………....................................................................................... 14
19. Branches Navigation ..……….………………....................................................................................... 14
20. Pull Request Navigation ……………………....................................................................................... 15

**Managing Bitbucket using GUI**

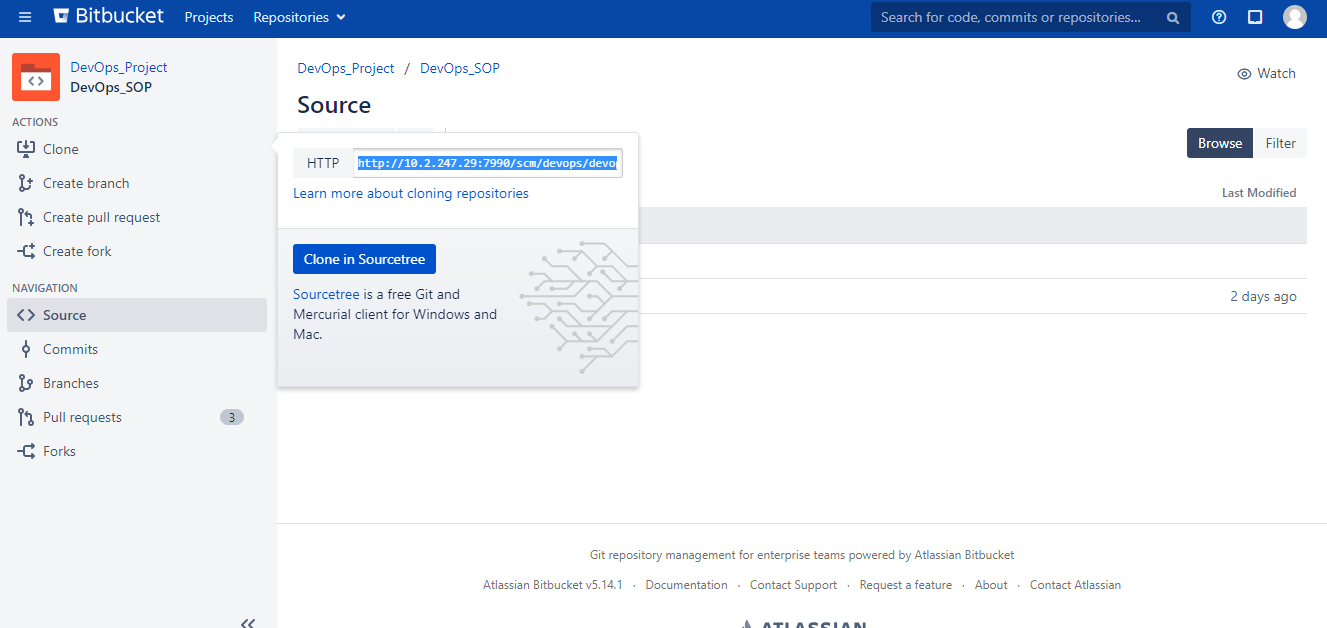
About Bitbucket

Bitbucket is a revision control system and web-based hosting service that allows teams to collaborate, securely store their code, and source control. Bitbucket allows anyone from your team to access the code, provides a code review system, tracks modifications history, and more. In addition, you have the capability to [integrate code deployment tools to make code deployment seamless](https://deploybot.com/bitbucket).

Cloning a Repository

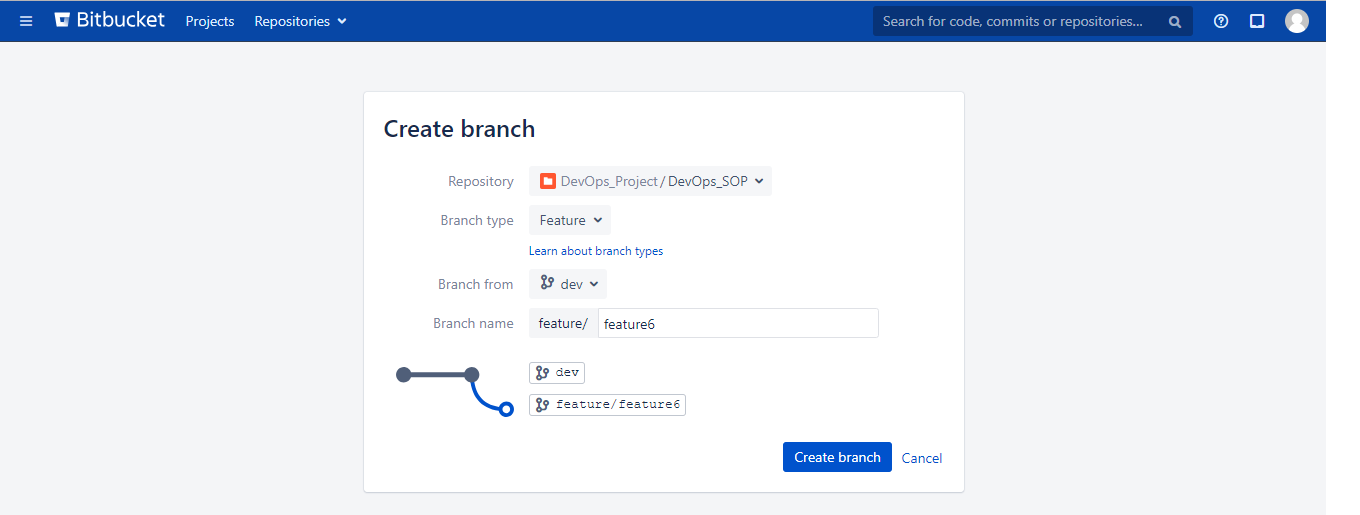
Cloning a repository refers to making exact copy of your repository. Following command will be used to make a clone of the repository

* git clone <git\_url>

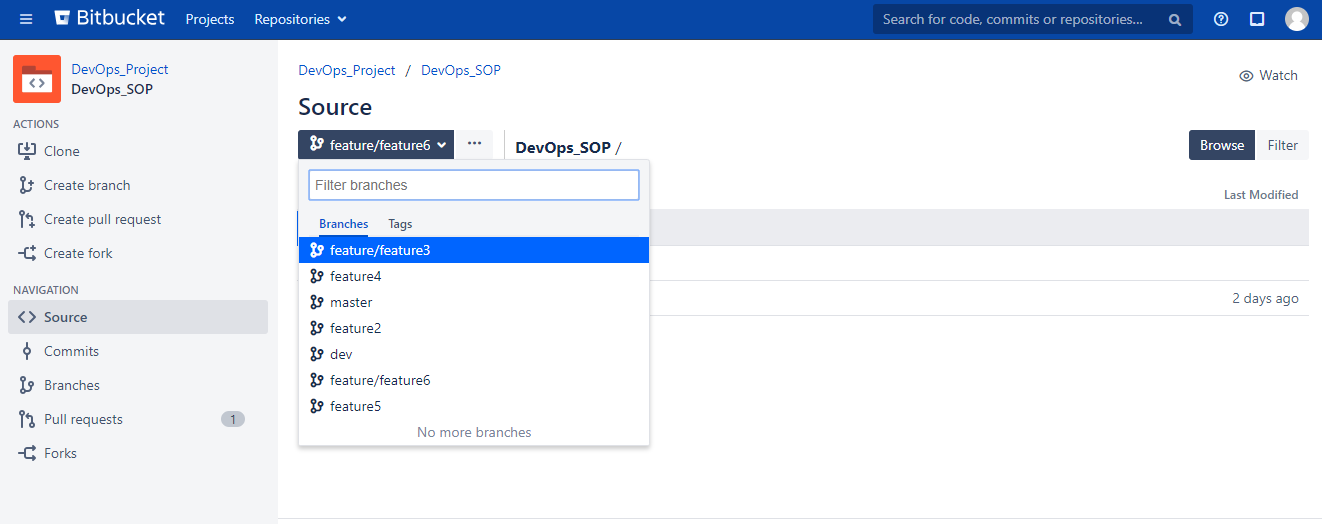


Branch Creation

Branching offers a way to work on a new feature without affecting the main codebase. You can create a branch from Bitbucket, Jira Software, or from your terminal. After you make changes, you push your branch to Bitbucket so that you can get it reviewed with a pull request.



After creation of a branch, it will look like this



File Creation, Add, Commit and Push to the Repository

To push a committed file into the repository requires following steps –

1. Checkout to the branch

* git checkout <branch\_name>

Ex - git checkout feature5

1. Create a file

* type “Comment” > <Filename>

Ex – type “comment” > README.md

1. Add it into the GIT

* git add <Filename>

Ex – git add README.md

1. Commit the changes into the file

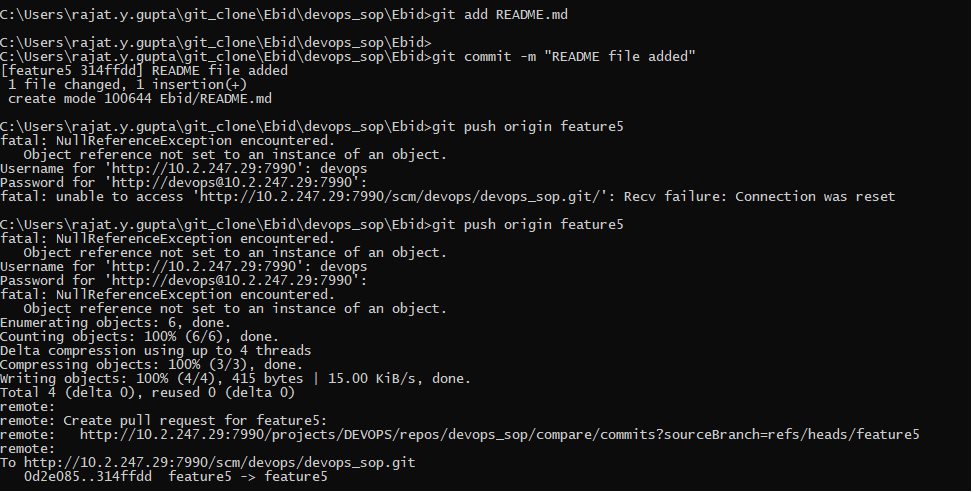
* git commit -m “type your message here”

Ex – git commit -m “README.md file added”

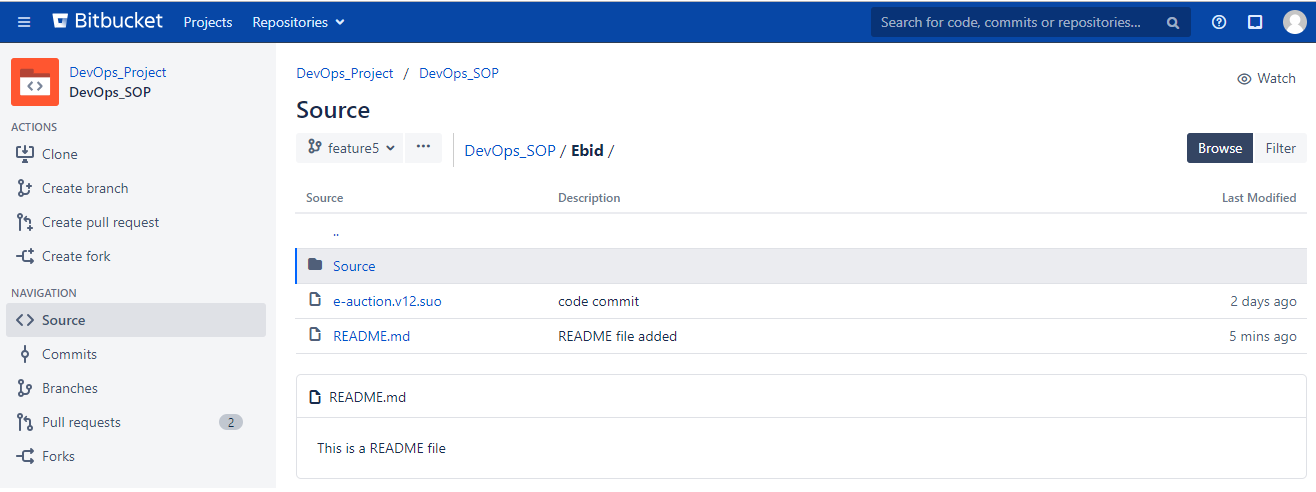
1. Pushing the code into the repository

* git push origin <branch\_name>

Ex – git push origin feature5

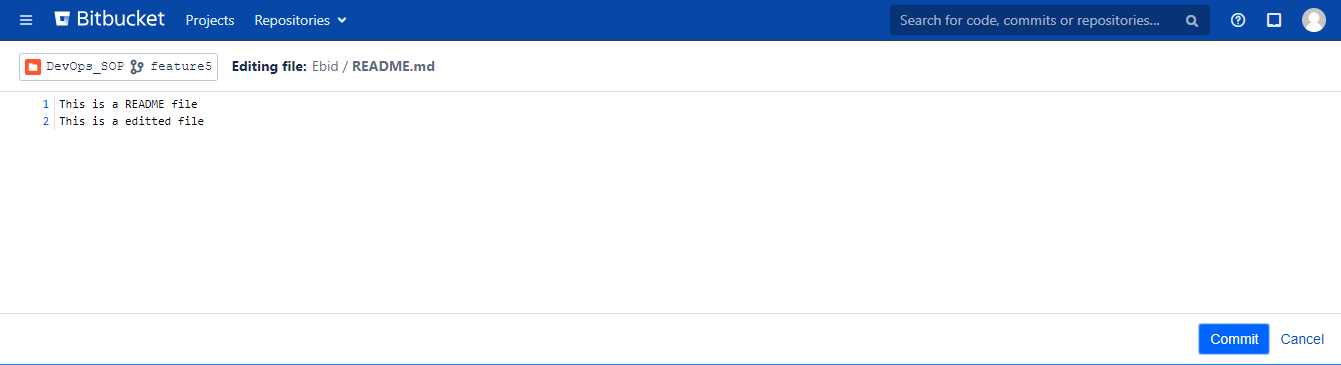


After push the changes from the GIT prompt, you can view the file at the GUI

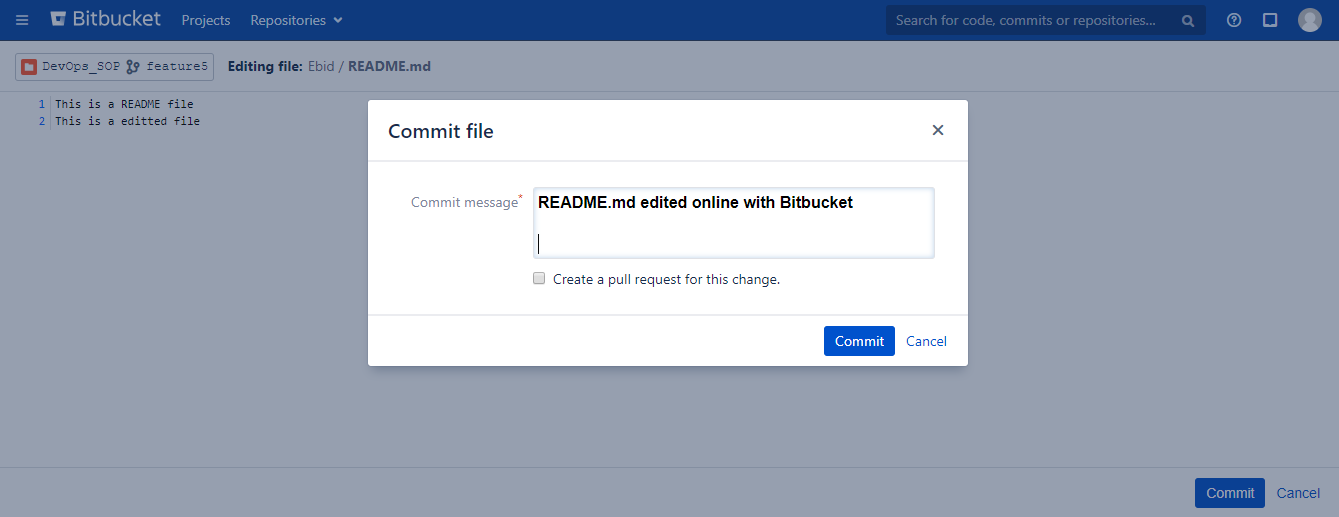


Editing and Committing File using Bitbucket GUI

Bitbucket GUI provides a way to edit your existing file.

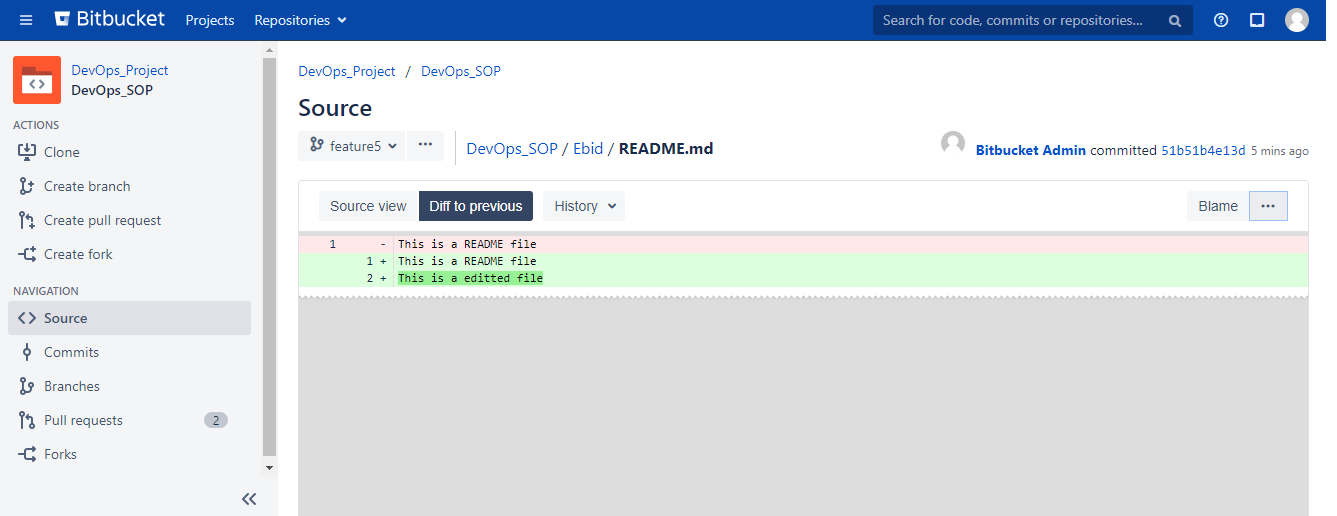


After making changes in the file, you can commit the changes from the GUI



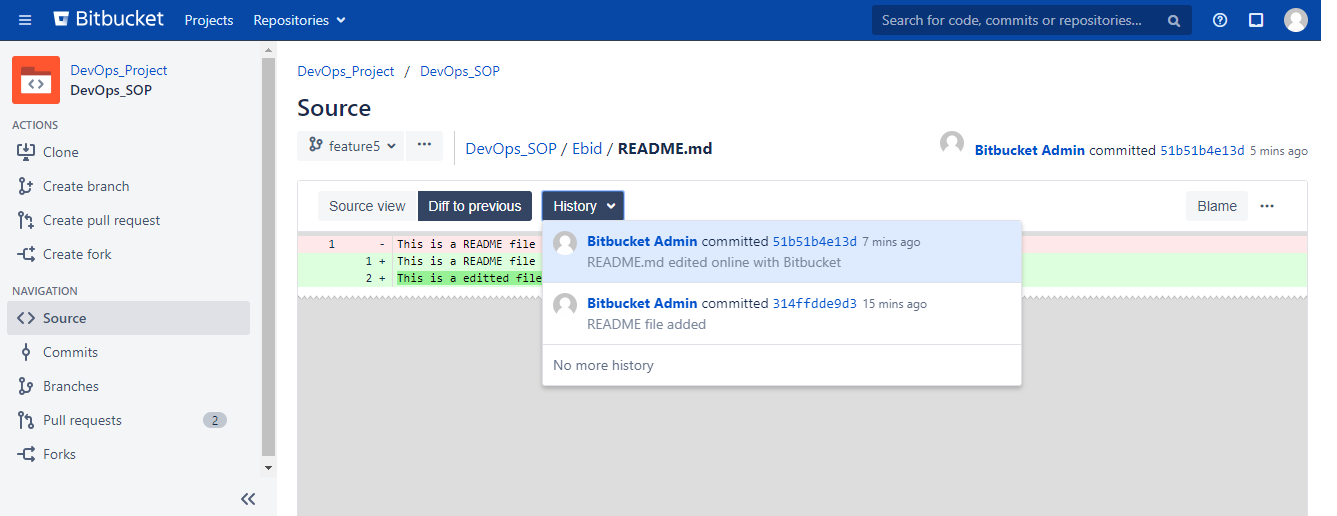
Checking difference from Previous file

After updating the file, you will able to check the changes done by you previously.

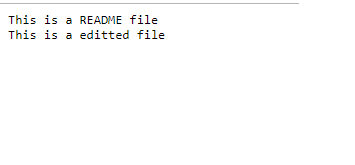


Change History

It provides you the details about the user who has changes the content in the file

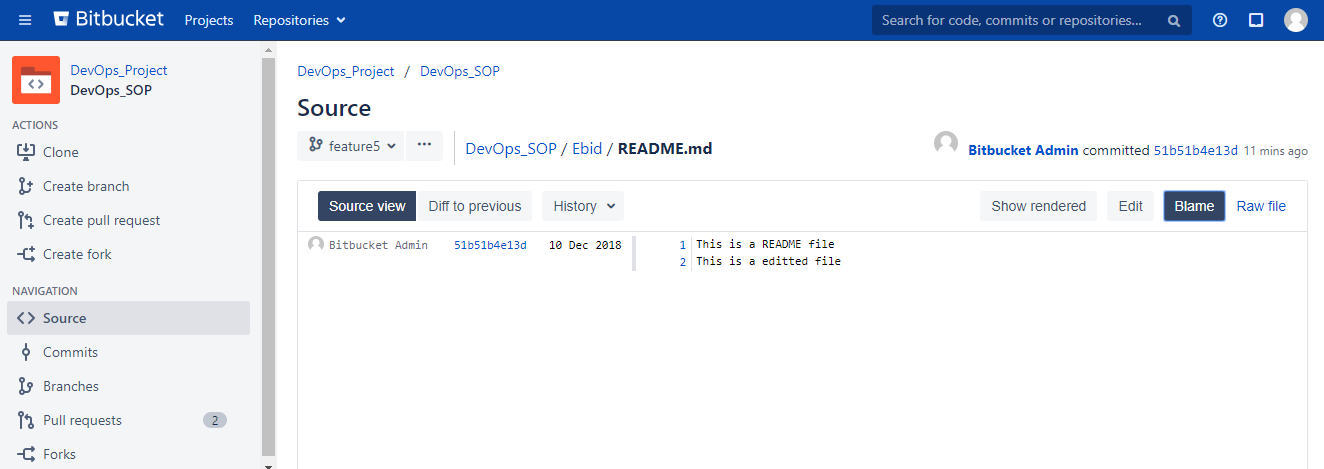


Viewing the file in RAW format

It will show the content of the file in the RAW format

Bitbucket Blame

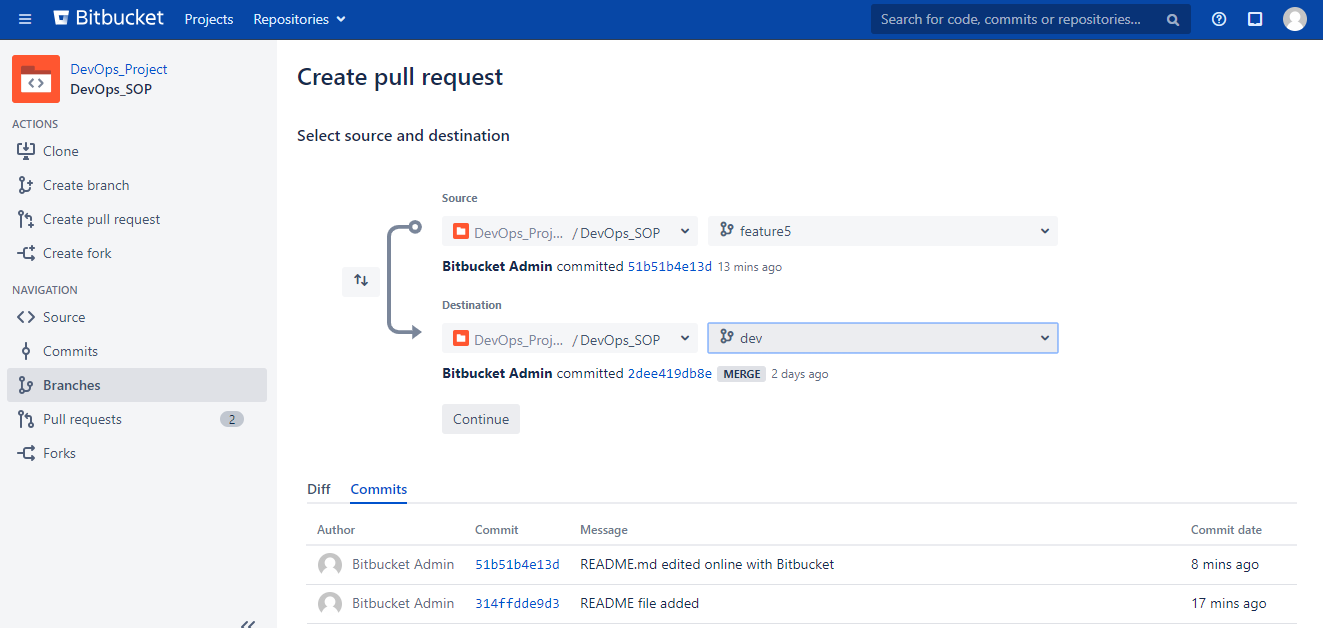
The high-level function of git blame is the display of author metadata attached to specific committed lines in a file. This is used to examine specific points of a file’s history and get context as to who the last author was that modified the line. This is used to explore the history of specific code and answer questions about what, how, and why the code was added to a repository.



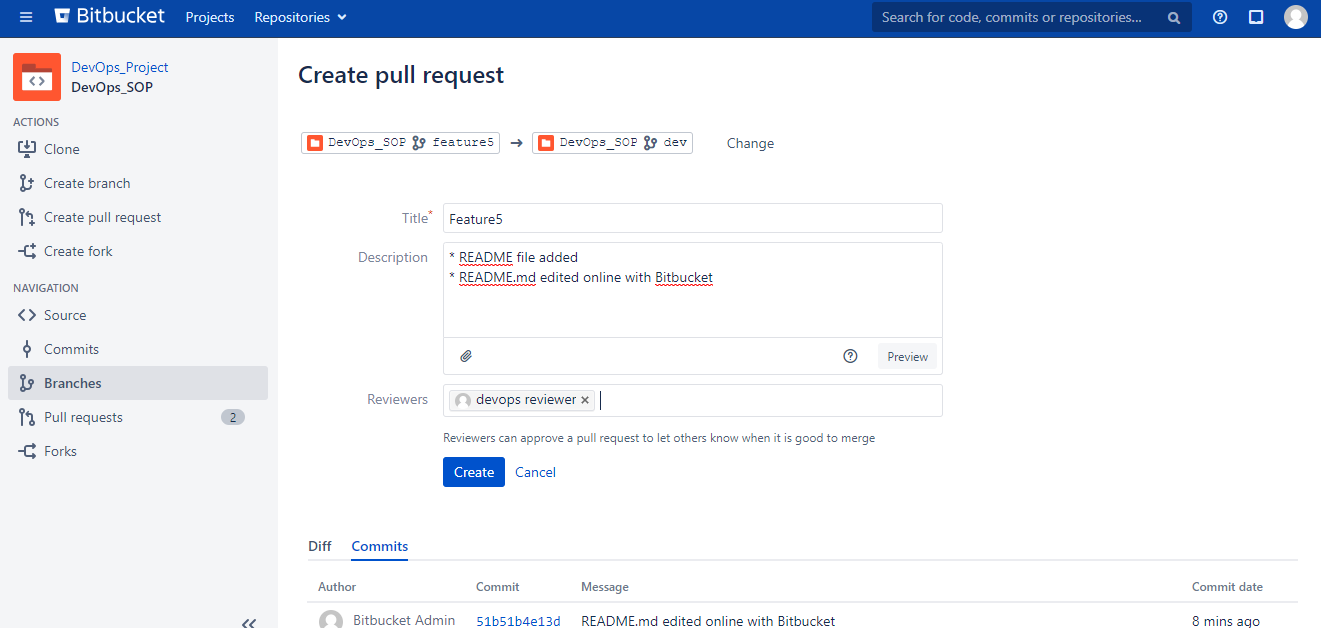
Raising Pull Request

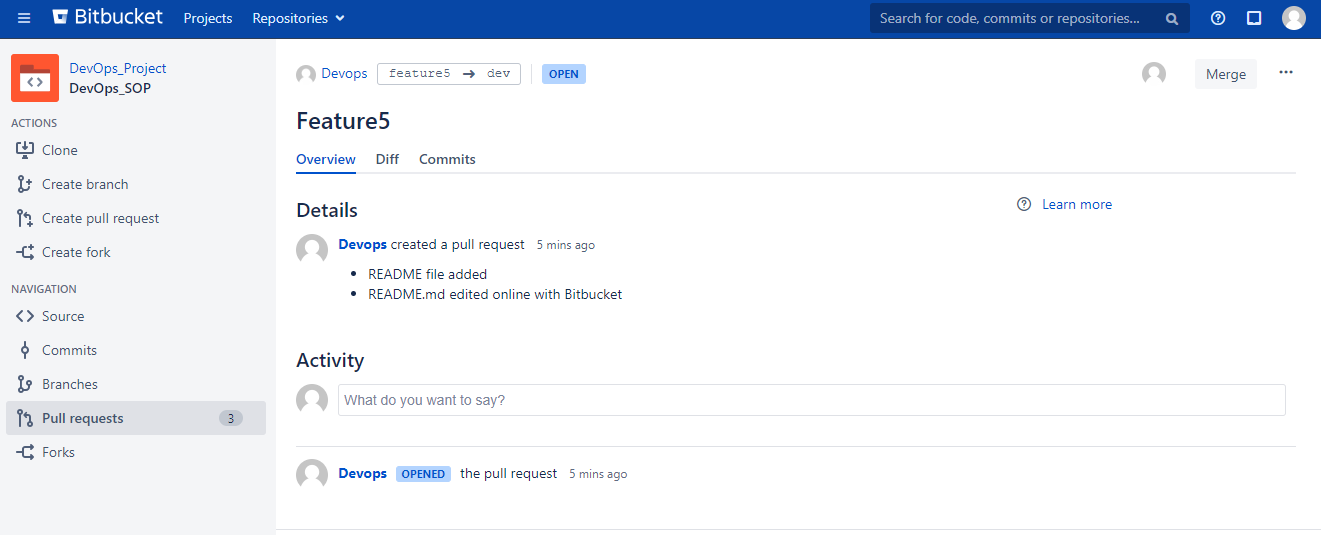
Pull request in bitbucket provide a user-friendly web interface for discussing proposed changes before integrating them into the official project.

After clicking the pull request option in bitbucket, the window will ask source branch and destination branch. By selecting them from the drop-down menu, you can move to the next window.

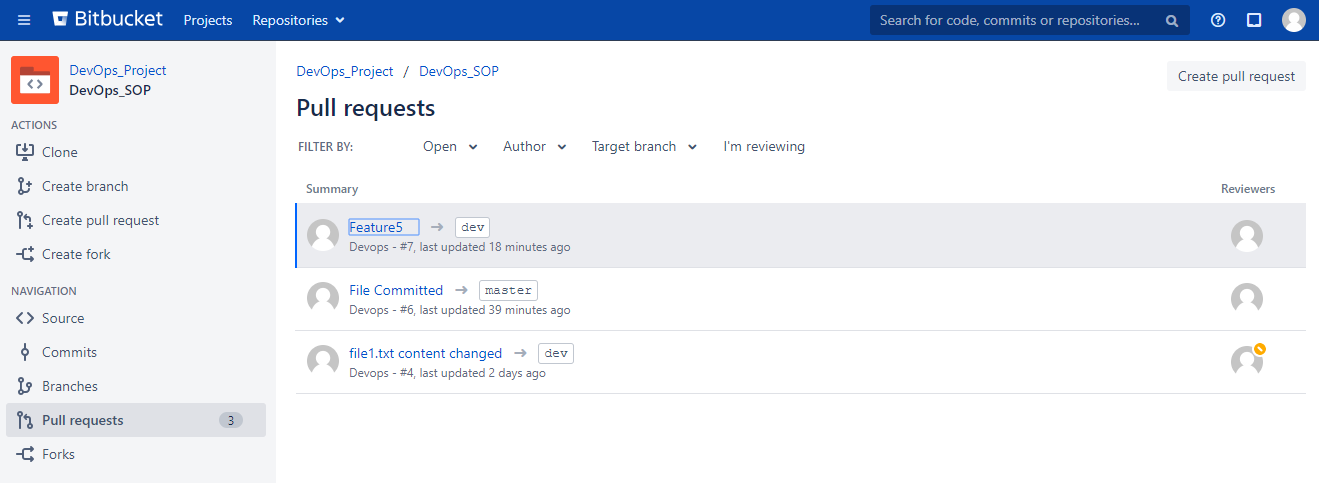


In this window, bitbucket will ask you to add the title, description and the user name for the review.

l



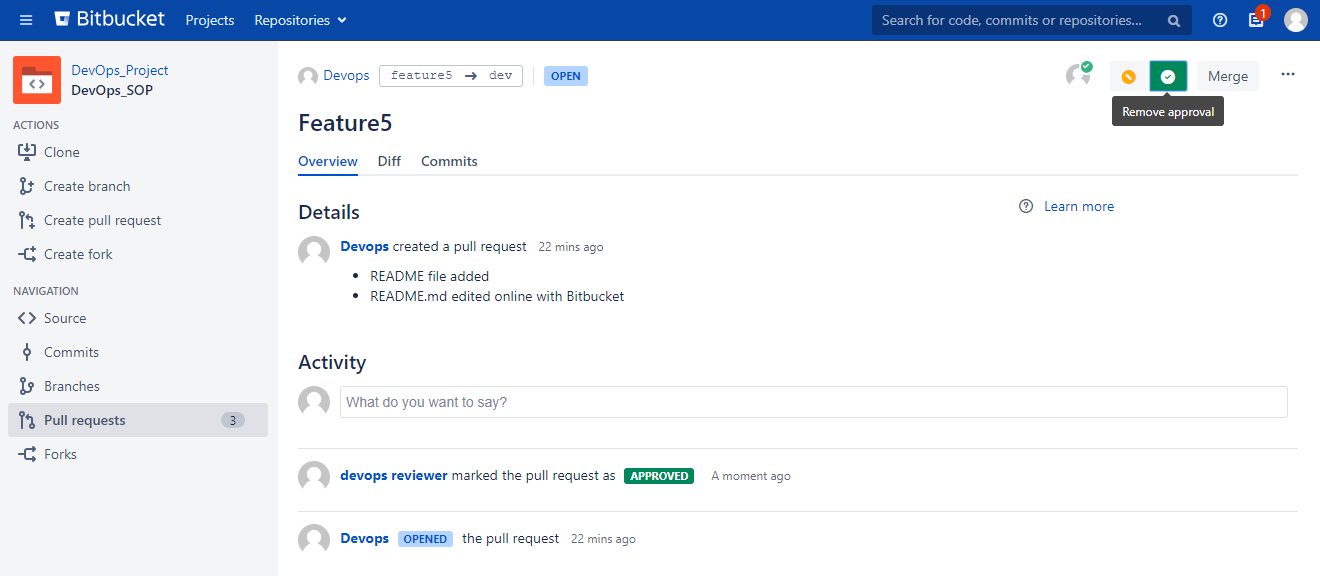
Reviewer can check all the pull requests. Edit, Reject and merge it.



Approving Pull Request

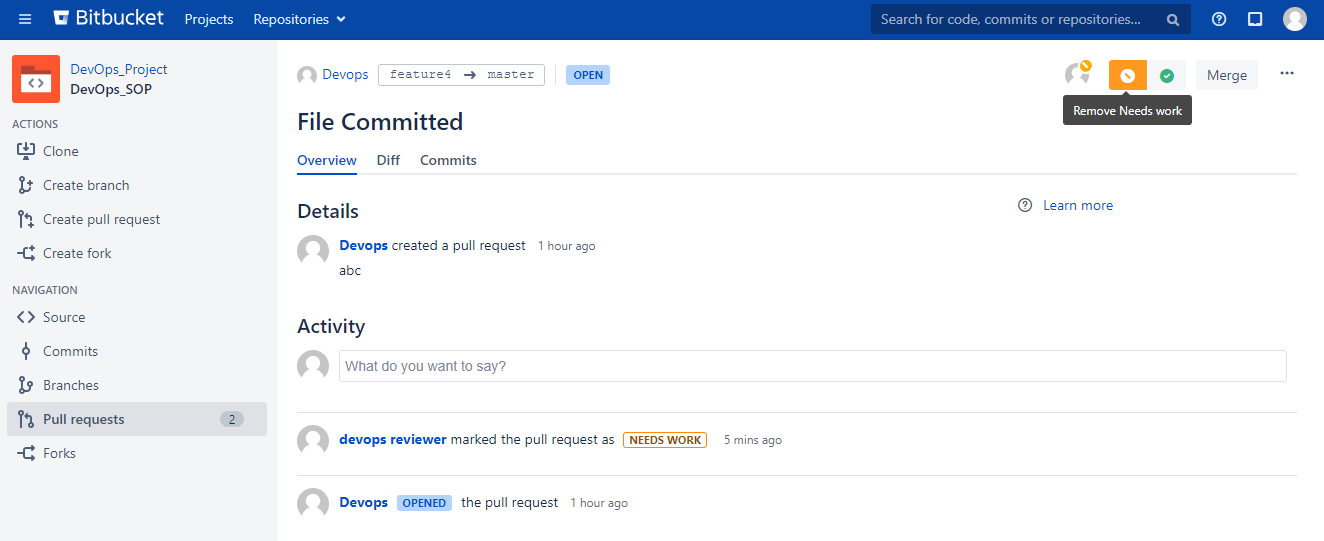
In this section, reviewer has multiple option to perform. Reviewer can approve, merge, reject, detach himself as a reviewer or ask the developer to rework.

Particularly, we are talking about how the reviewer can approve the pull request.



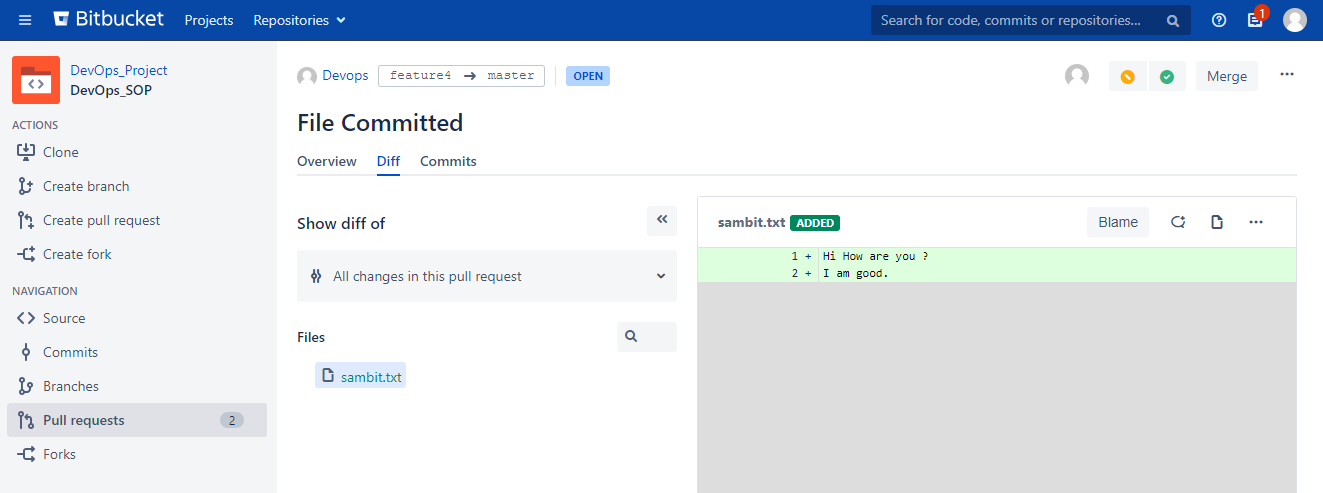
Rejecting Pull Request

Reviewer can ask developer to rework on the code. Below image will show, how to do that.



Reviewing the difference

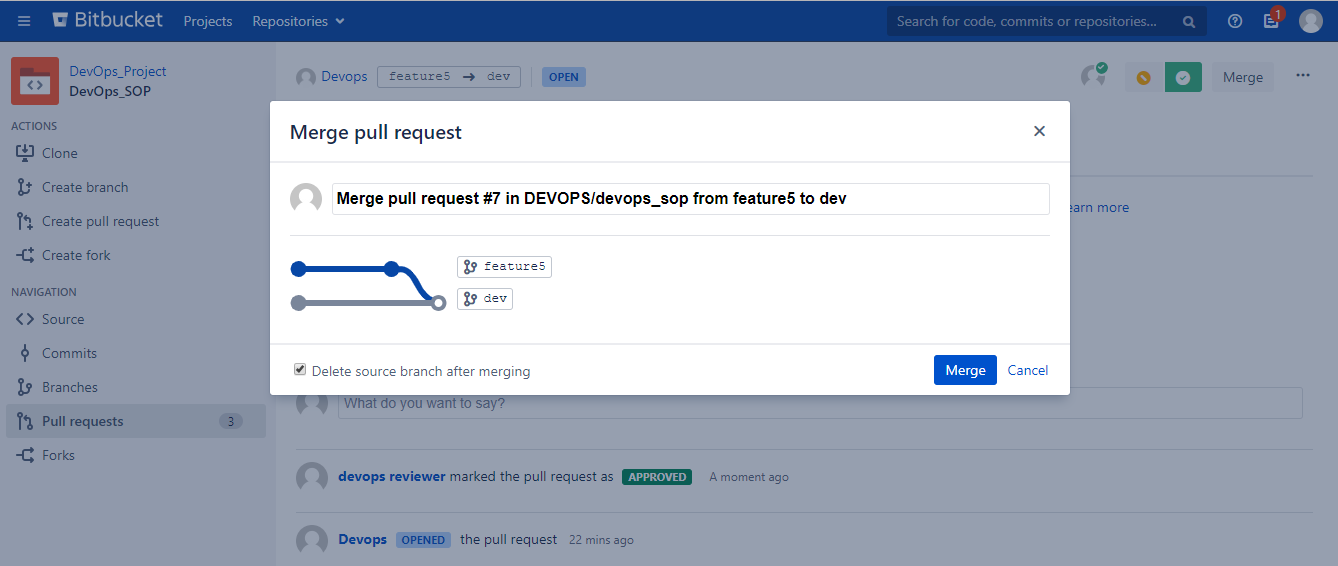
In this section, reviewer can check the changes done by the developer in code. And analyze the code impact.



Merging the Pull Request

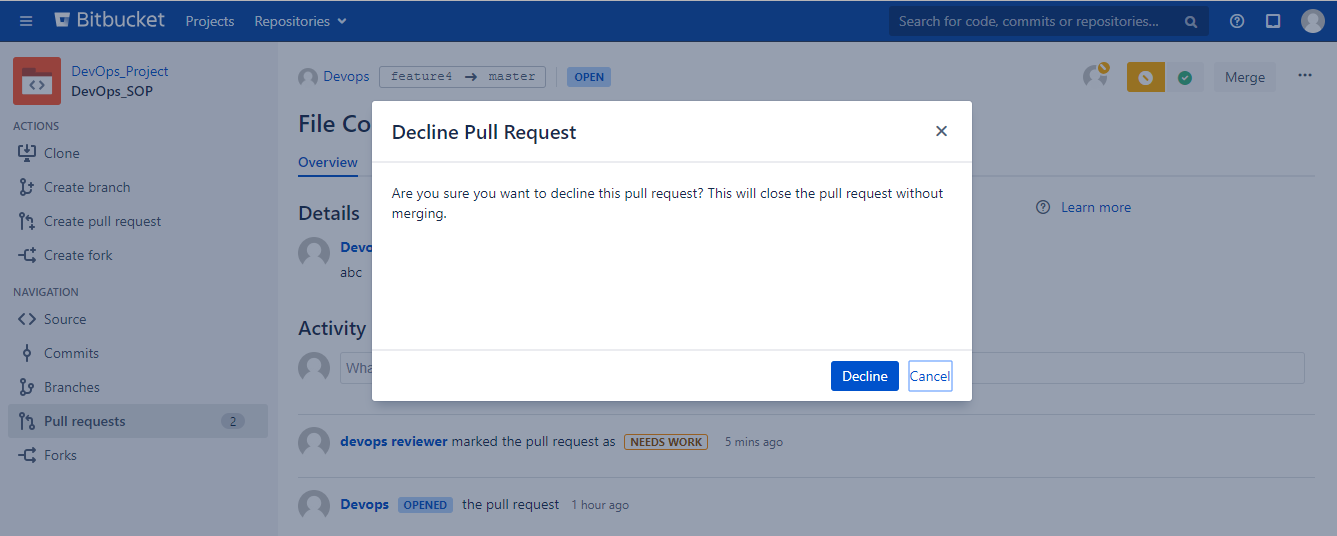
When the desired number of reviewers have approved a pull request, you can merge the pull request if you have written (or admin) permission on the repository. If you've been touching the same code as someone else, you may have [a merge conflict that you need to resolve locally](https://confluence.atlassian.com/bitbucket/resolve-merge-conflicts-704414003.html). After you merge a pull request, you can revert the pull request to remove the merge commit from the repository.

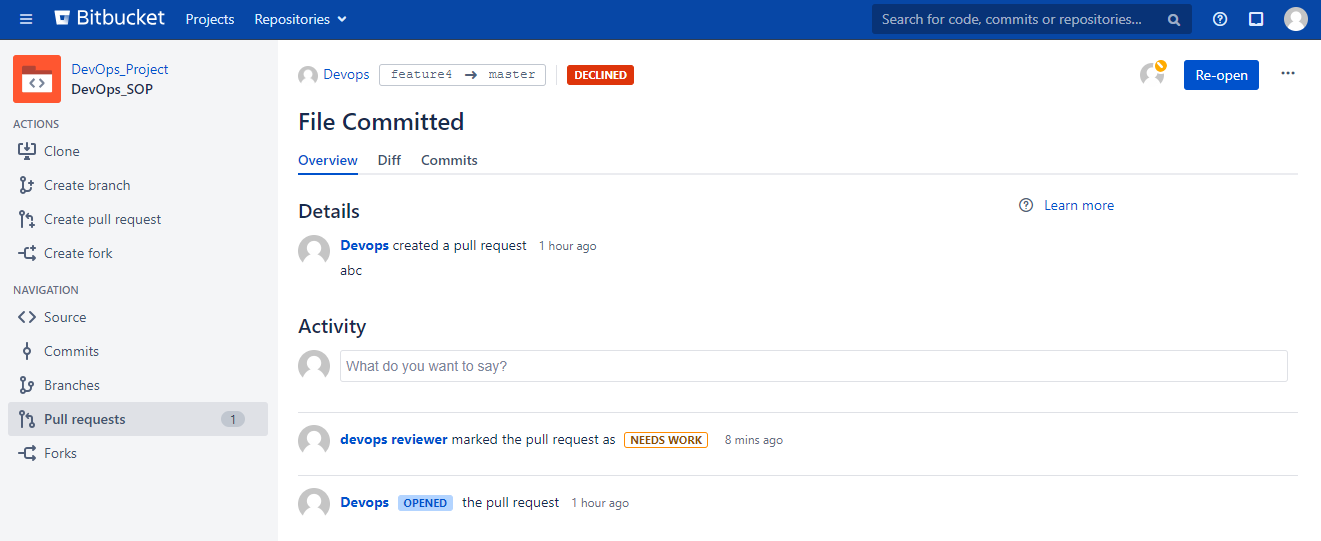
Merging your changes is the final stage of the pull request process. To merge a pull request, you Have to click the merge button after getting approved.



Decline Pull Request

Reviewing can decline the pull request if he got any wrong changes or bug in the code. So that, merge will not happen.

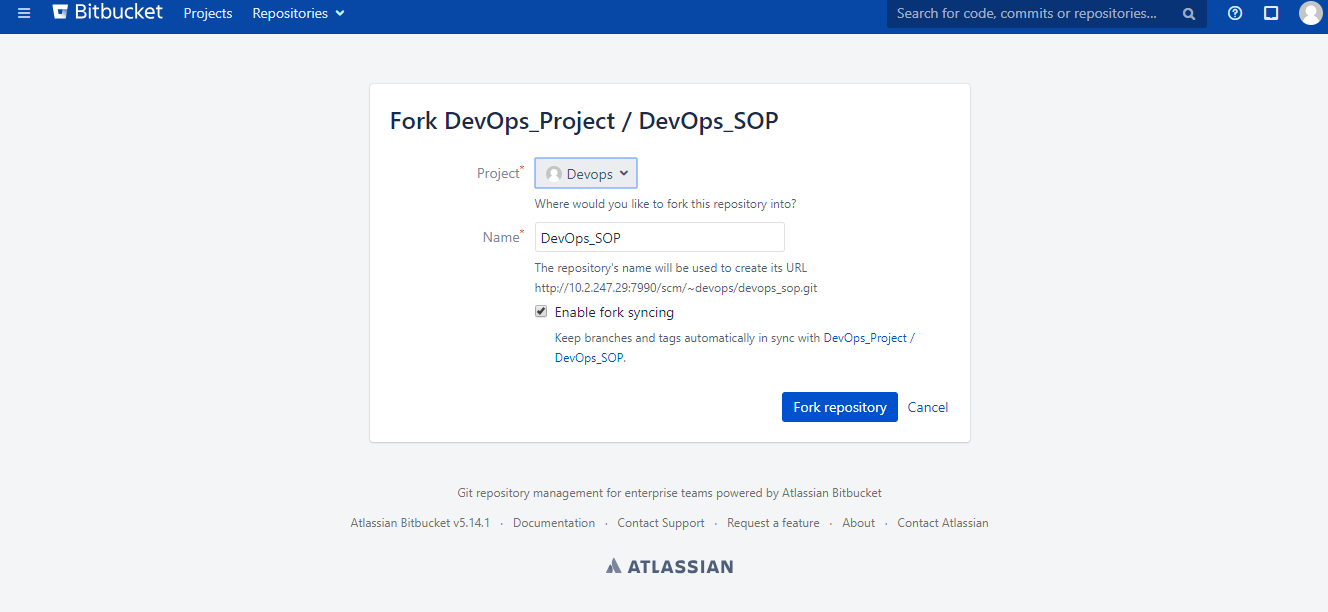




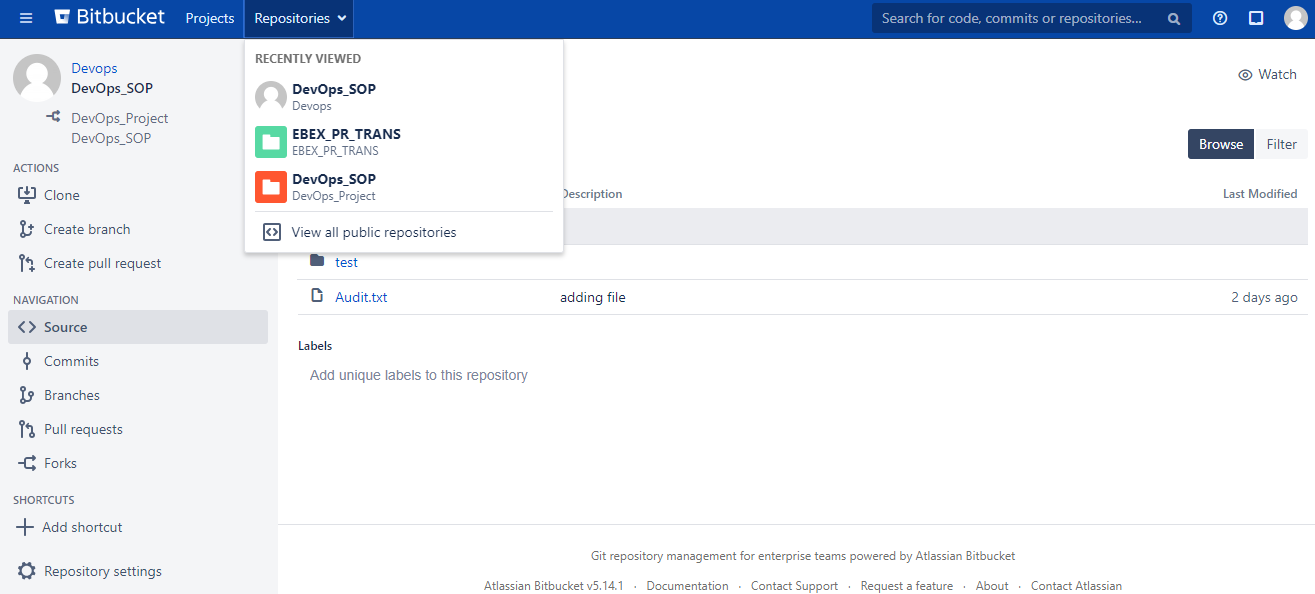
Forking a Repository

In git and Mercurial, you create branches by starting with either the head/trunk or an existing branch. When you do this, your changes become part of the main project repository. If you want to work on a fully separate copy of the project, you may want to consider making a Fork.

Forking is a way for you to clone a repository at a specific point, and to modify it from there. Fork is just another way of saying clone. Bitbucket Cloud manages the relationship between the original repository and the fork for you. Forking is particularly useful if you want to do some major development work that you may or may not later merge back into the repository.

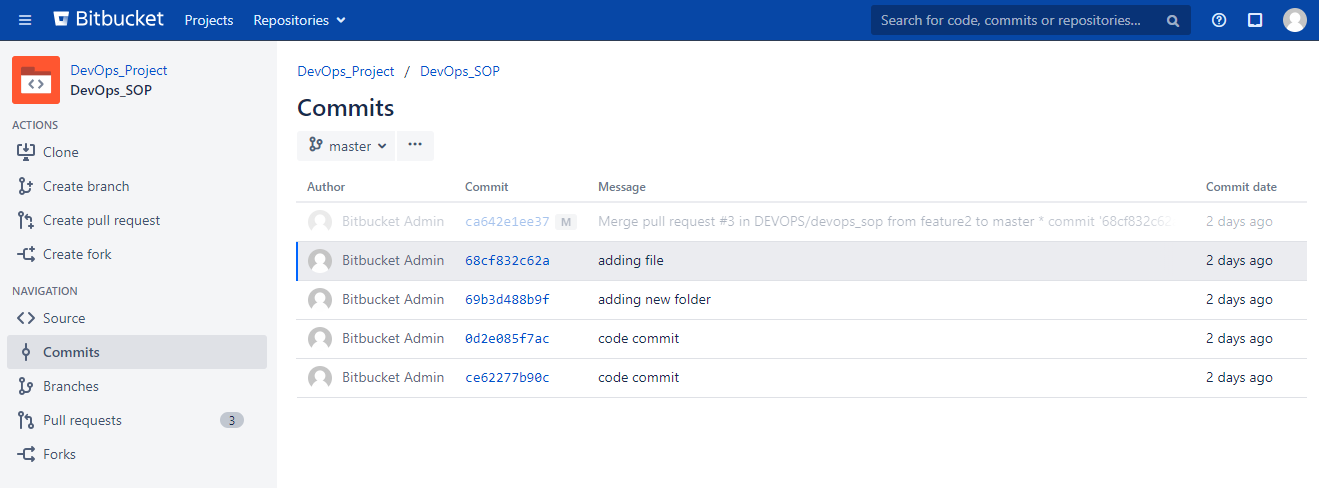


In the below image, forked repository is showing.



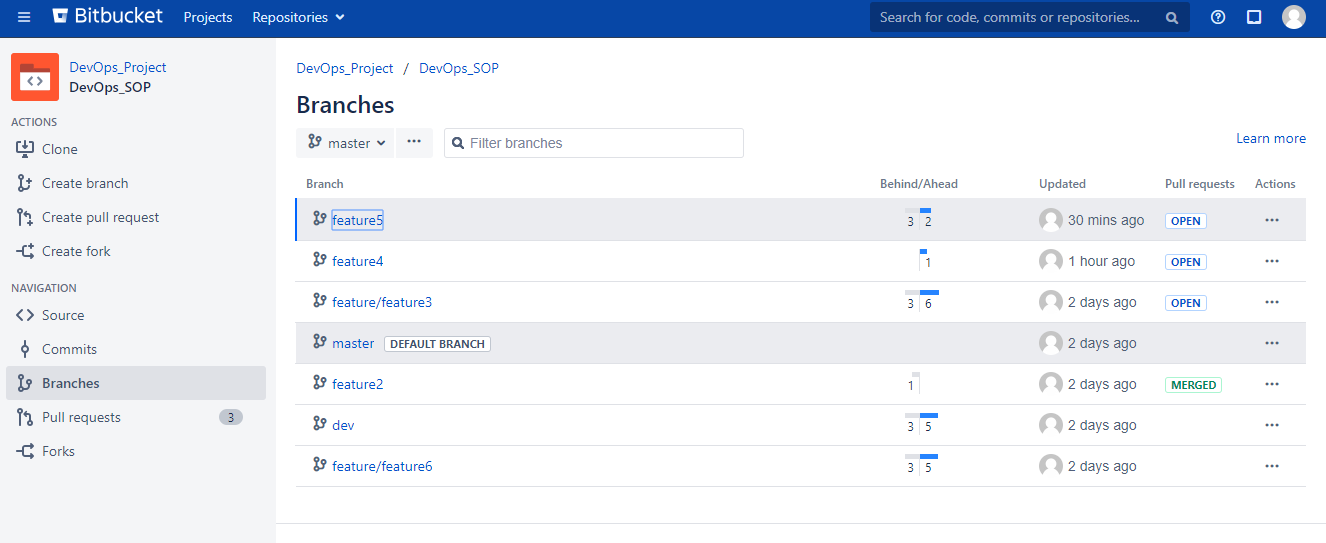
Commits Navigation

Here you can check all the commits done previously on the repository



Branches Navigation

Here you can check all the branches created previously



Pull Requests Navigation

Here you can check all the pull requests raised previously

