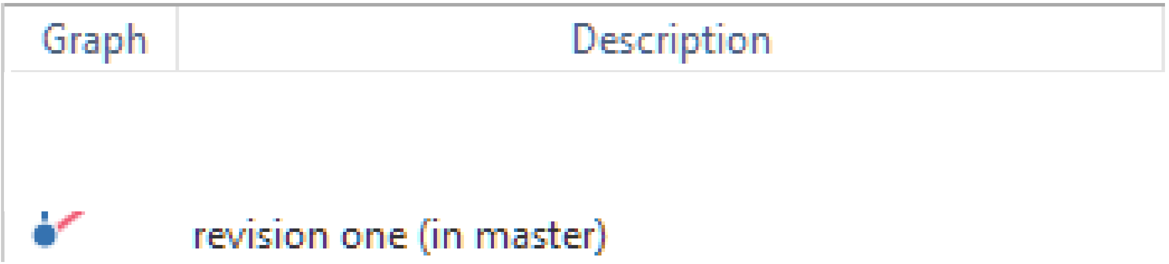
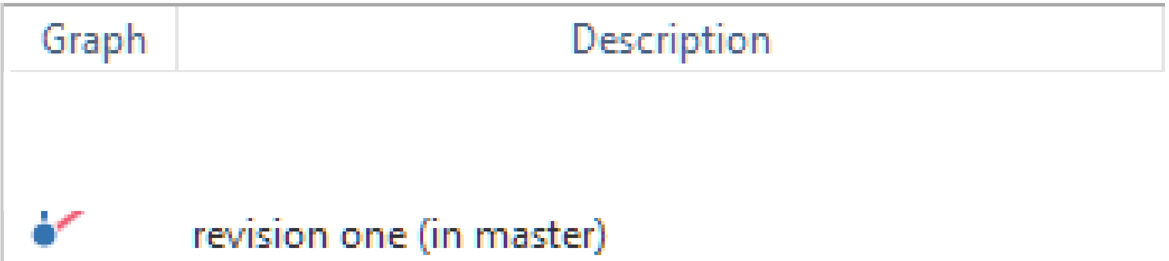
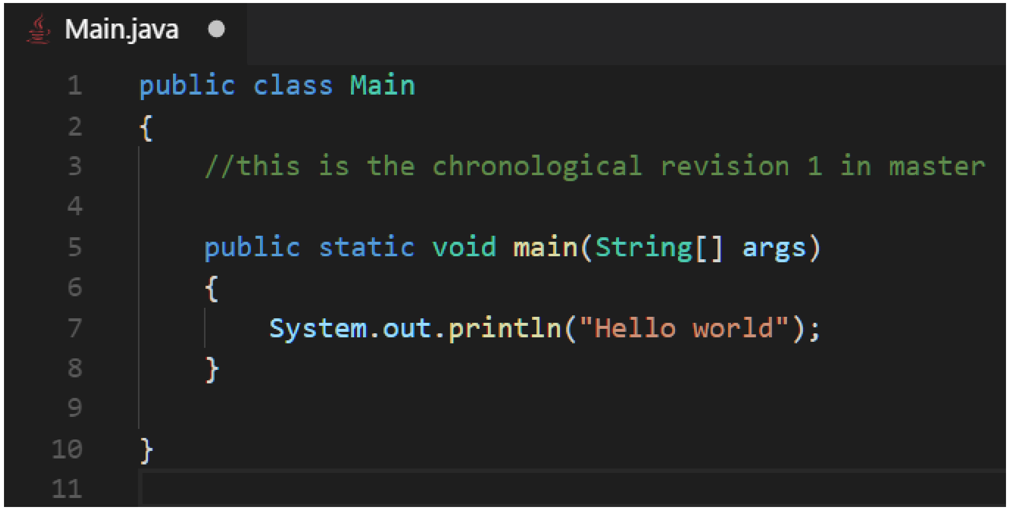
Example of Merge Conflict in Git



Graph	Description
	revision one (in master)

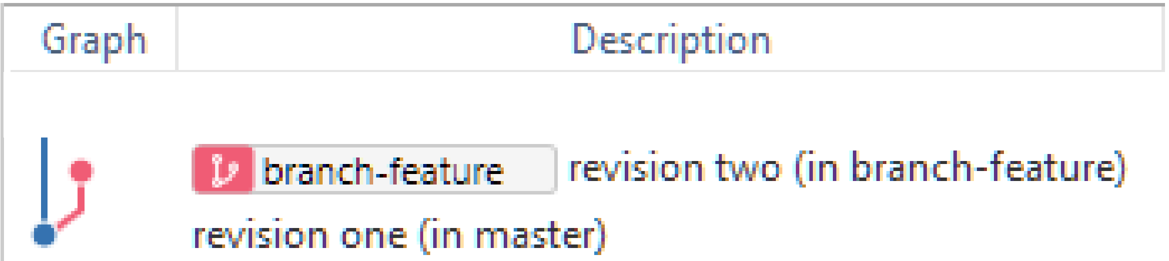


Graph	Description
	revision one (in master)



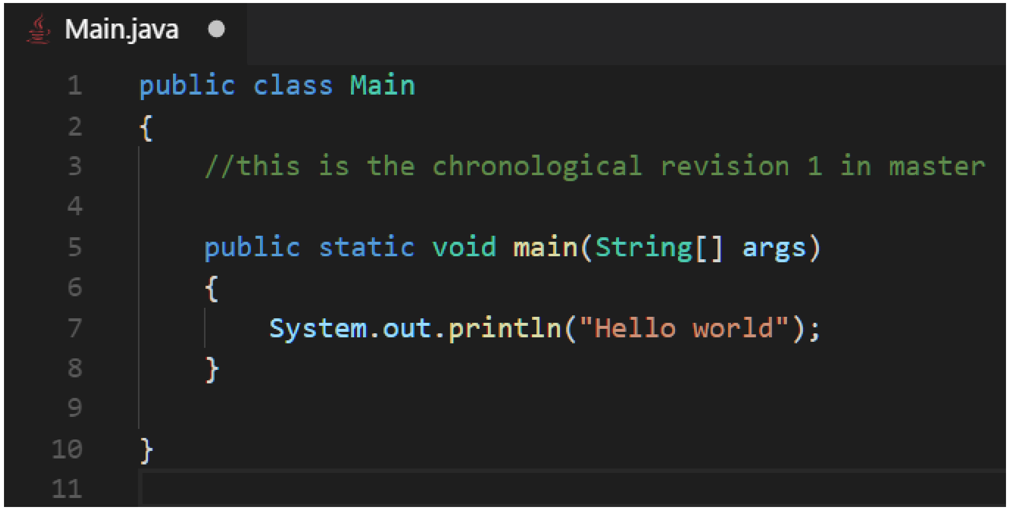
Main.java ●

```java
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

}

```

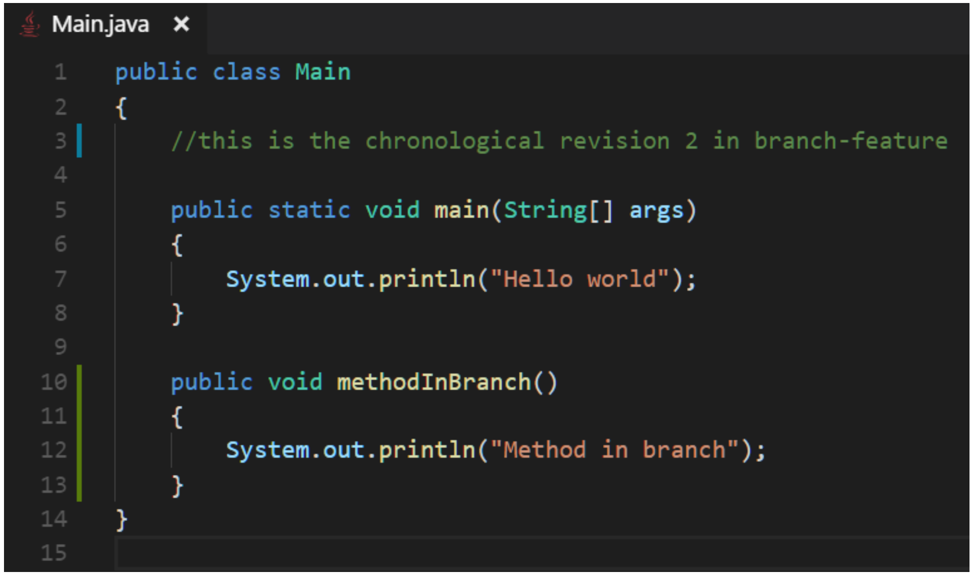


branch-feature revision two (in branch-feature)

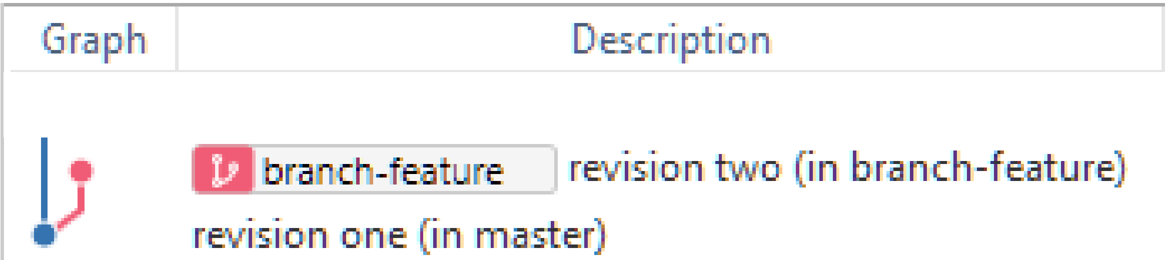
revision one (in master)



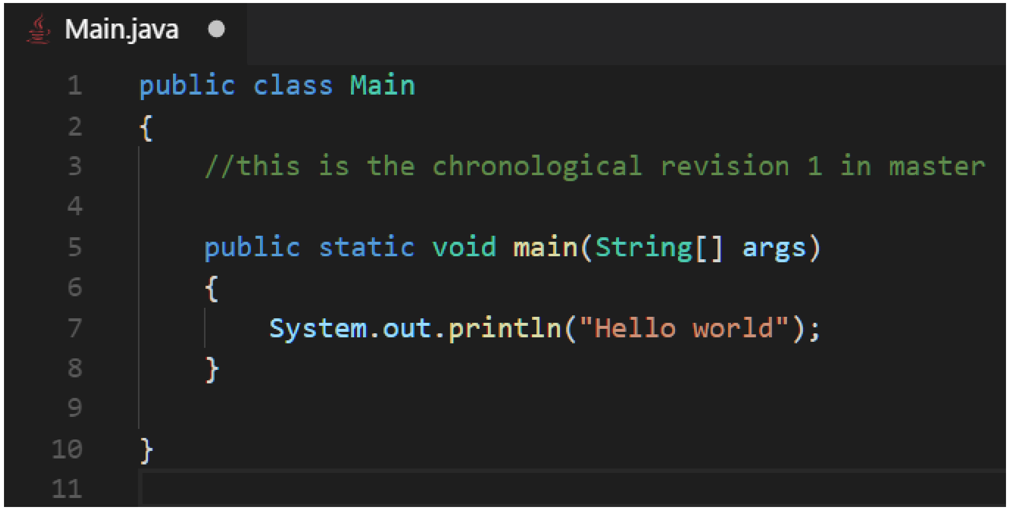
```java
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

}

```



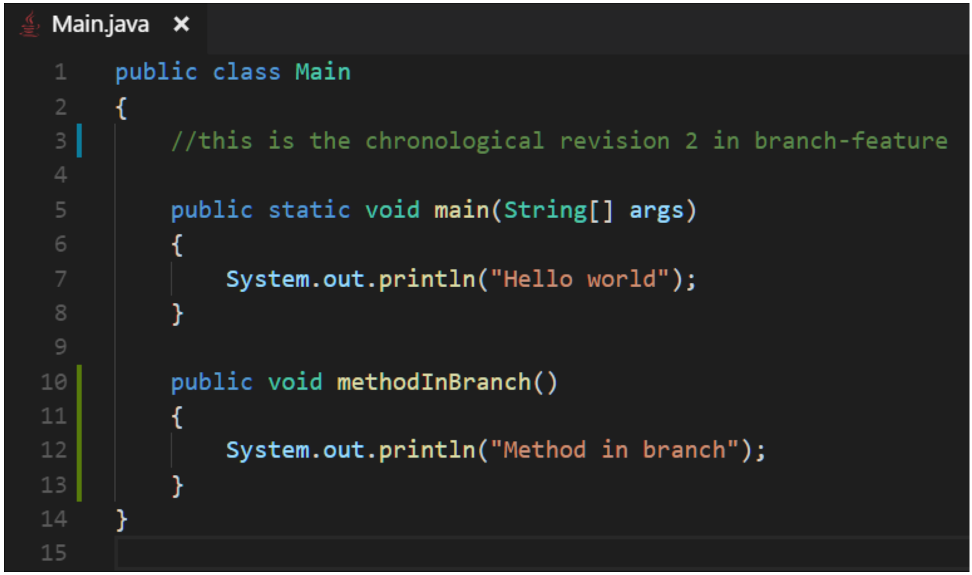
```java
public class Main
{
    //this is the chronological revision 2 in branch-feature

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}

```



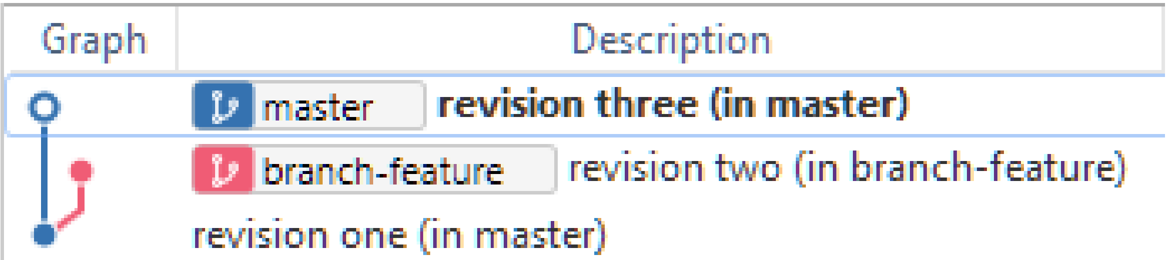
Graph	Description
	branch-feature revision two (in branch-feature)
	revision one (in master)



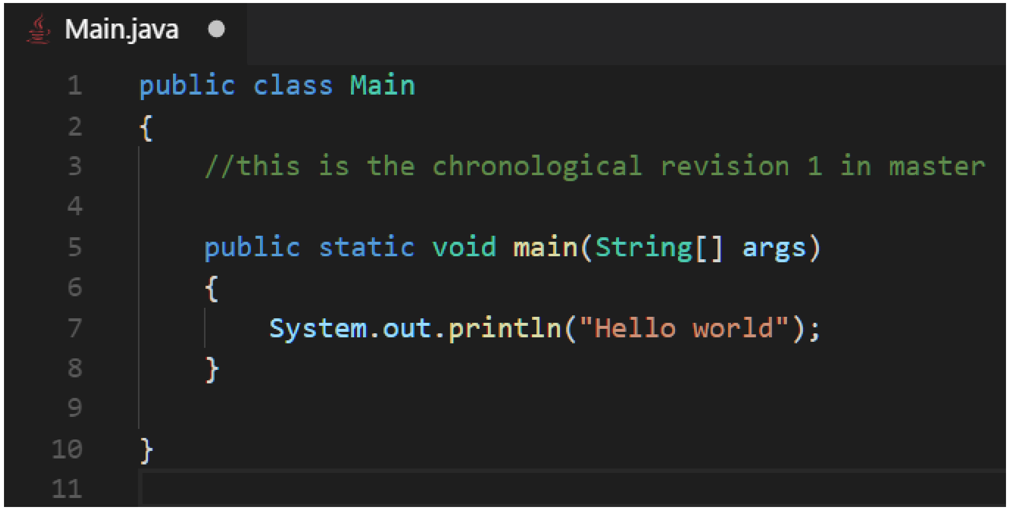
```java
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

}

```



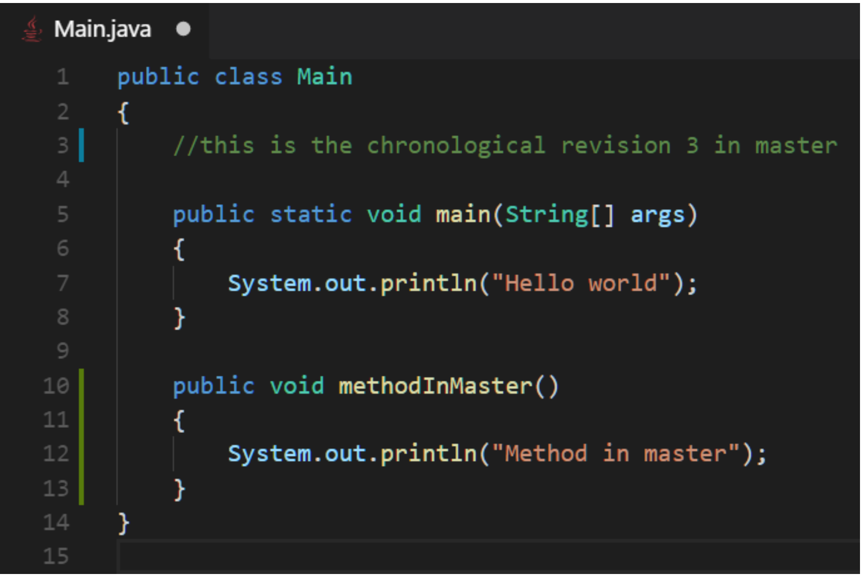
```java
public class Main
{
    //this is the chronological revision 2 in branch-feature

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}

```



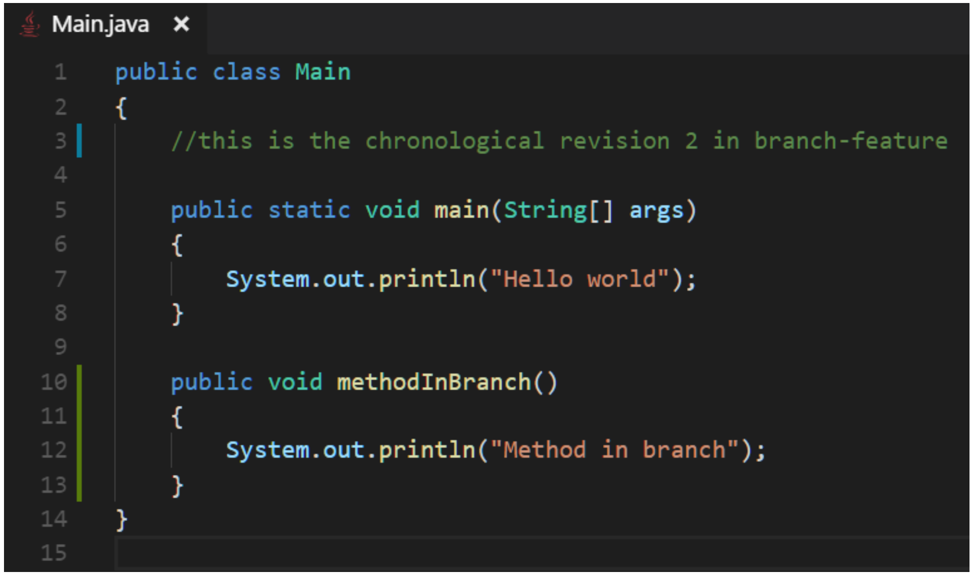
Graph	Description
	master **revision three (in master)**
	branch-feature revision two (in branch-feature)
	revision one (in master)



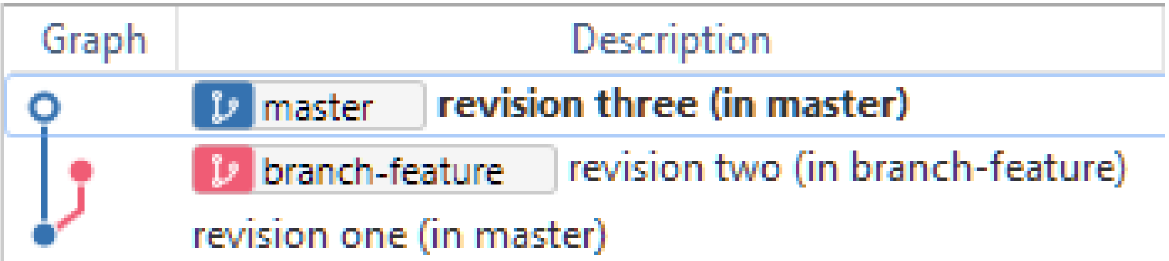
```java
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

}

```



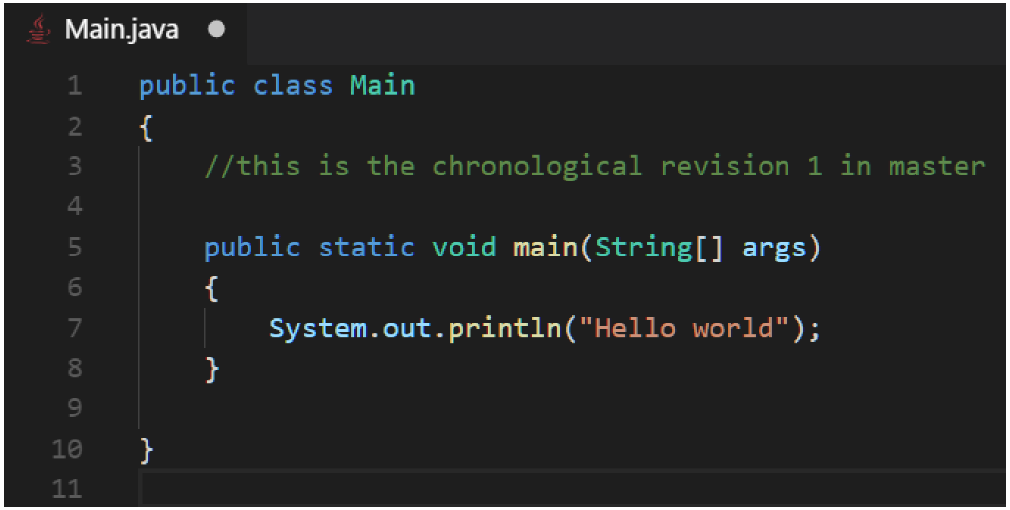
```java
public class Main
{
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```



```java
public class Main
{
    //this is the chronological revision 2 in branch-feature

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```

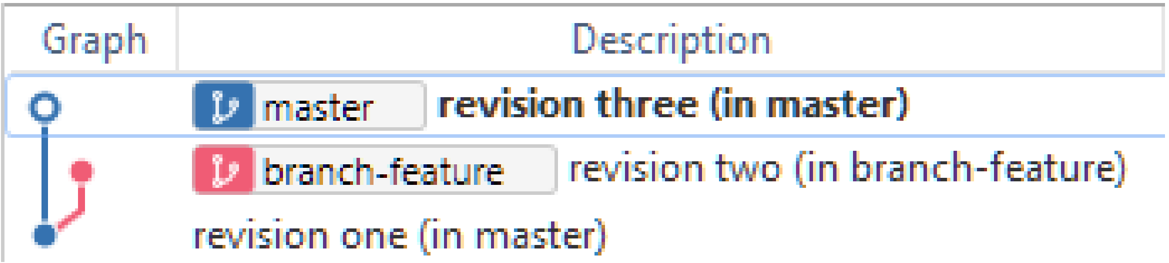


Graph	Description
○	master **revision three (in master)**
●	branch-feature revision two (in branch-feature)
●	revision one (in master)



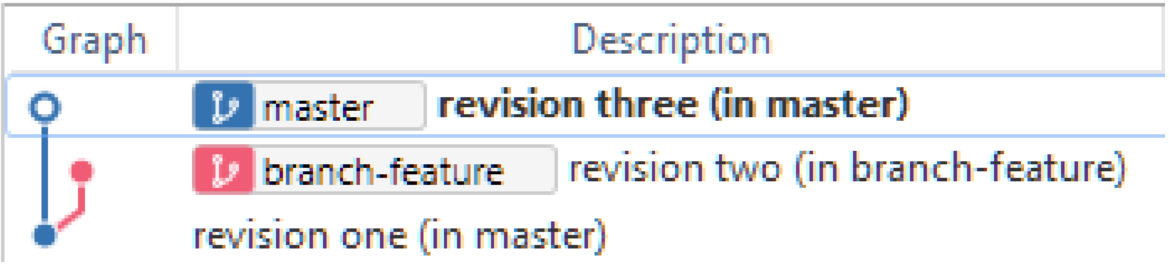
```java
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

}
```

Resolution of Merge Conflict with 2-way-merge



Graph	Description
○	master **revision three (in master)**
●	branch-feature revision two (in branch-feature)
●	revision one (in master)

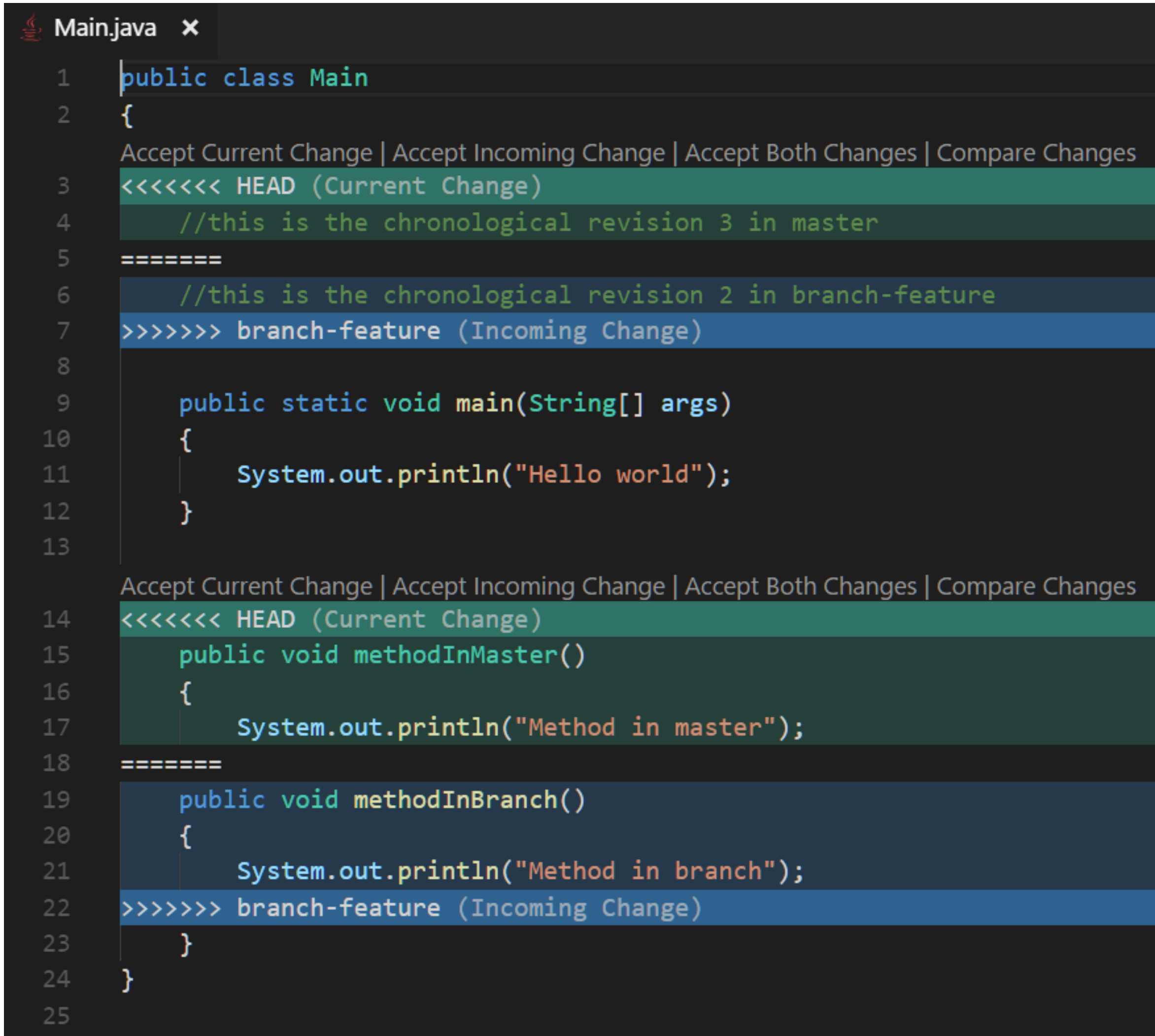
Resolution of Merge Conflict with 2-way-merge



Graph	Description
	master **revision three (in master)**
	branch-feature revision two (in branch-feature)
	revision one (in master)

In master, run:

```
git merge branch-feature
```



```
public class Main
{
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<<< HEAD (Current Change)
    //this is the chronological revision 3 in master
=======
    //this is the chronological revision 2 in branch-feature
>>>>>>> branch-feature (Incoming Change)

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<<< HEAD (Current Change)
    public void methodInMaster()
    {
        System.out.println("Method in master");
=======
    public void methodInBranch()
    {
        System.out.println("Method in branch");
>>>>>>> branch-feature (Incoming Change)
    }
}

```

Changes are visualized with:

CODE

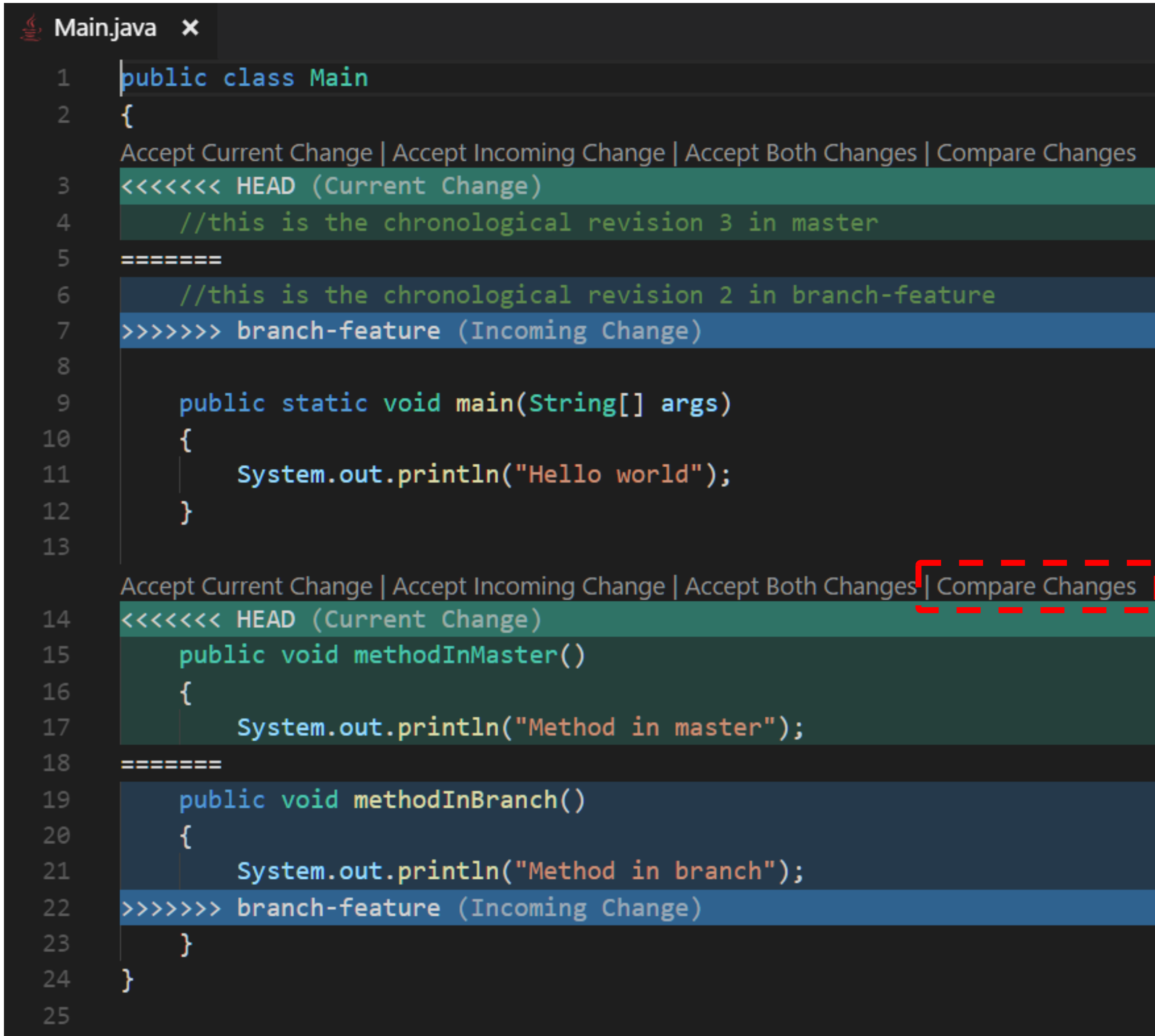
Visual Studio Code

https://github.com/Microsoft/vscode

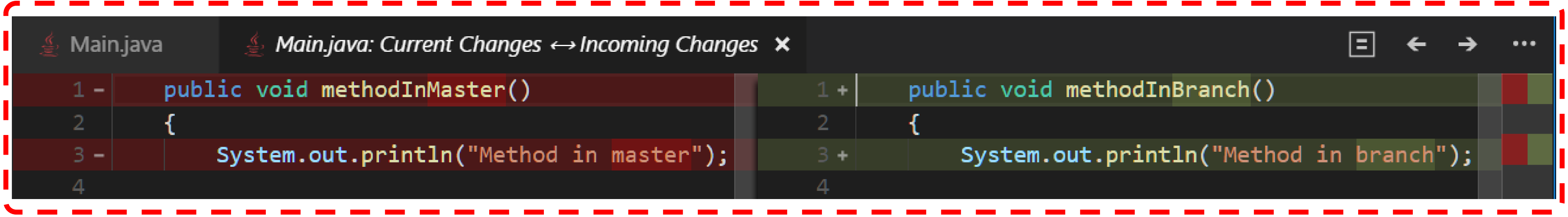
Changes are visualized with:

CODE
Visual Studio Code

https://github.com/Microsoft/vscode

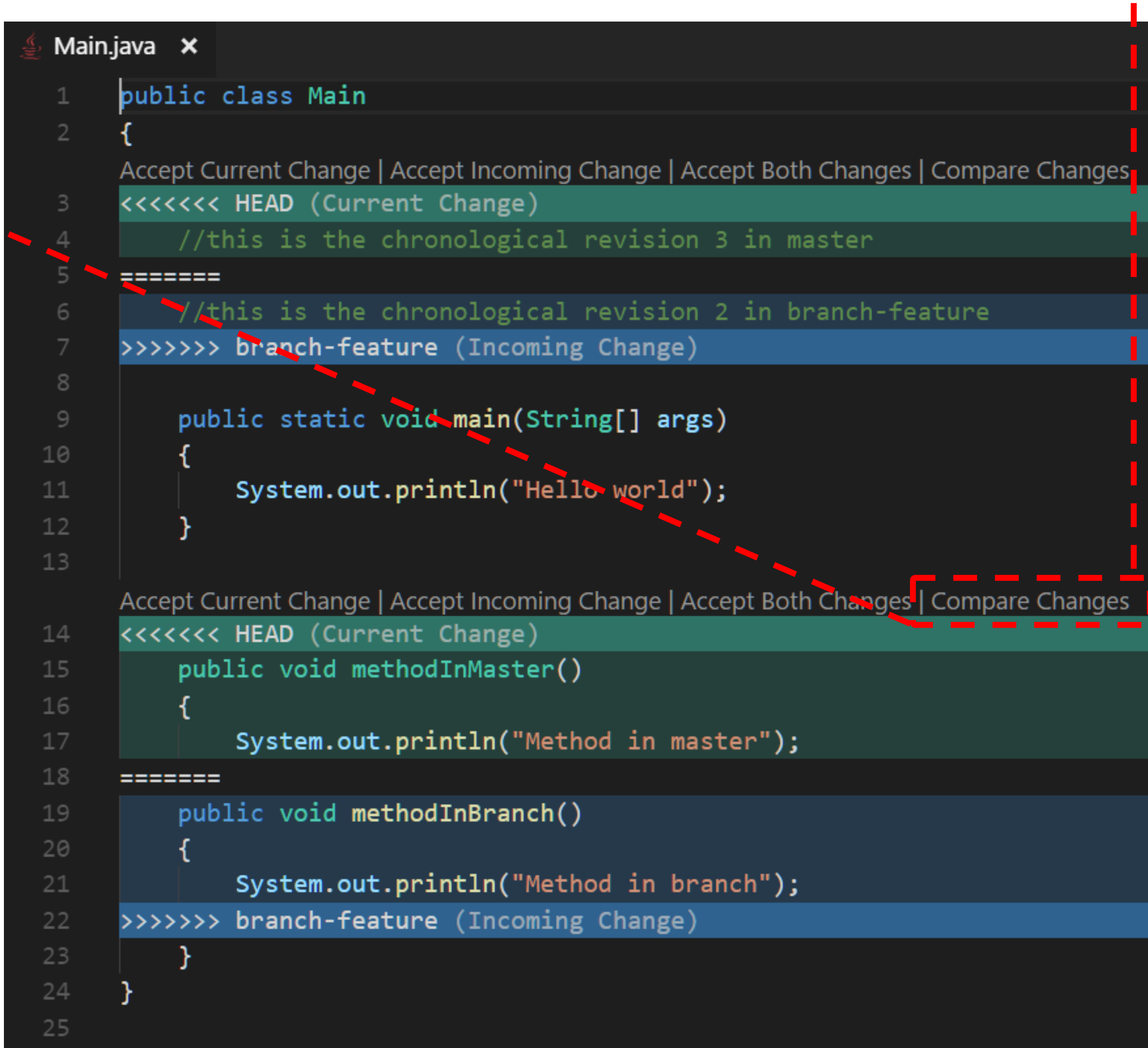


```
public class Main
{
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<<< HEAD (Current Change)
    //this is the chronological revision 3 in master
=======
    //this is the chronological revision 2 in branch-feature
>>>>>>> branch-feature (Incoming Change)

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<<< HEAD (Current Change)
    public void methodInMaster()
    {
        System.out.println("Method in master");
=======
    public void methodInBranch()
    {
        System.out.println("Method in branch");
>>>>>>> branch-feature (Incoming Change)
    }
}
```



```
public void methodInMaster()
{
    System.out.println("Method in master");
```

```
public void methodInBranch()
{
    System.out.println("Method in branch");
```



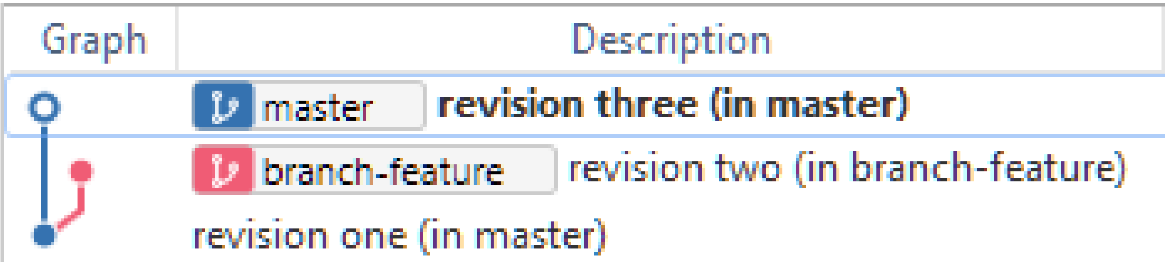
```
public class Main
{
Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<<< HEAD (Current Change)
    //this is the chronological revision 3 in master
=======
    //this is the chronological revision 2 in branch-feature
>>>>>>> branch-feature (Incoming Change)

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
<<<<<<< HEAD (Current Change)
    public void methodInMaster()
    {
        System.out.println("Method in master");
=======
    public void methodInBranch()
    {
        System.out.println("Method in branch");
>>>>>>> branch-feature (Incoming Change)
    }
}
```

Changes are visualized with:

Visual Studio Code

https://github.com/Microsoft/vscode

Resolution of Merge Conflict with 3-way-merge

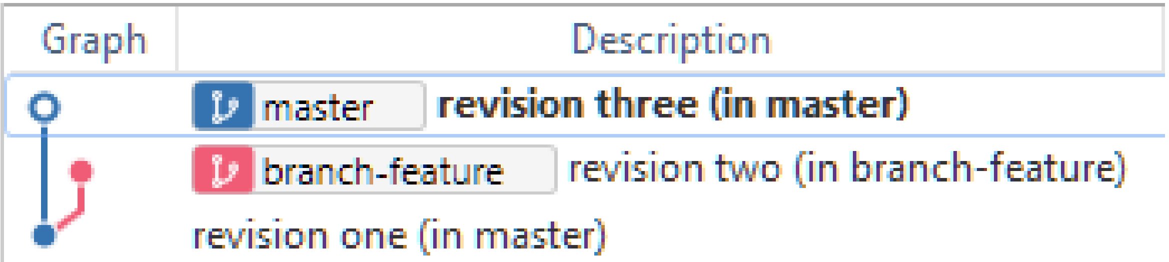


Graph	Description
	master **revision three (in master)**
	branch-feature revision two (in branch-feature)
	revision one (in master)

```
In master, run:
git merge branch-feature
git mergetool
```

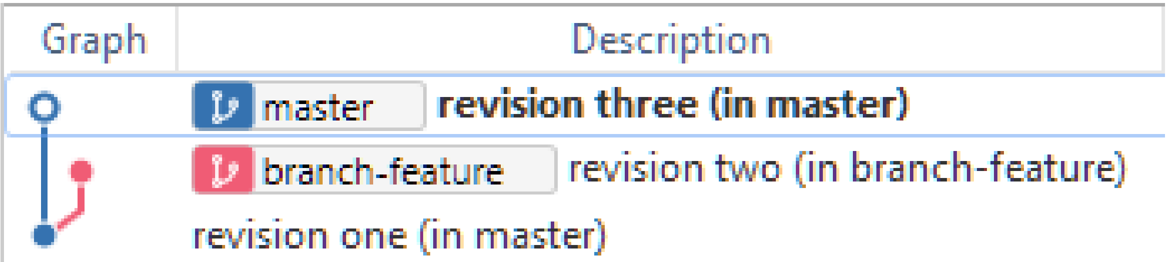
- Note: The `git mergetool` was configured to be Perforce P4Merge (from the default vimdiff)
  - https://www.perforce.com/products/helix-core-apps/merge-diff-tool-p4merge

3-way merge (Perforce P4Merge)

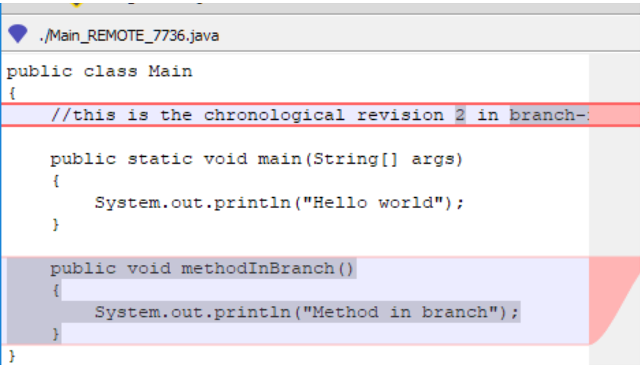


Graph	Description
	master **revision three (in master)**
	branch-feature revision two (in branch-feature)
	revision one (in master)

3-way merge (Perforce P4Merge)



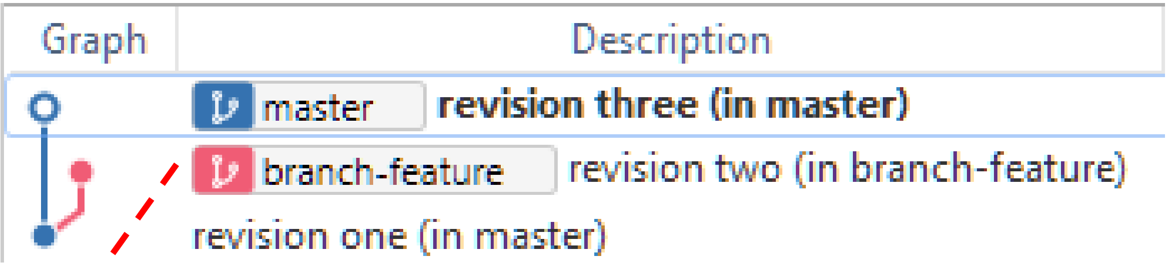
Graph	Description
	master revision three (in master)
	branch-feature revision two (in branch-feature)
	revision one (in master)



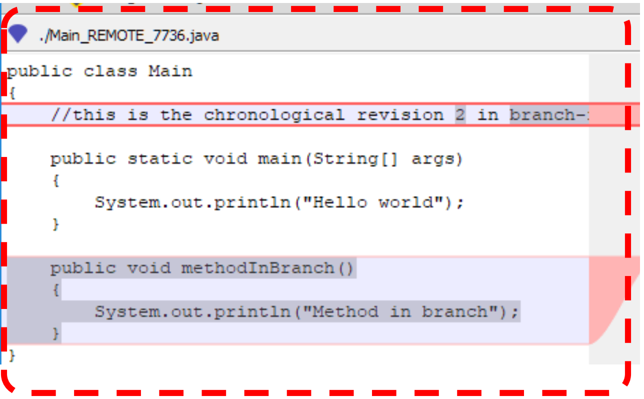
./Main_REMOTE_7736.java

```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```

3-way merge (Perforce P4Merge)



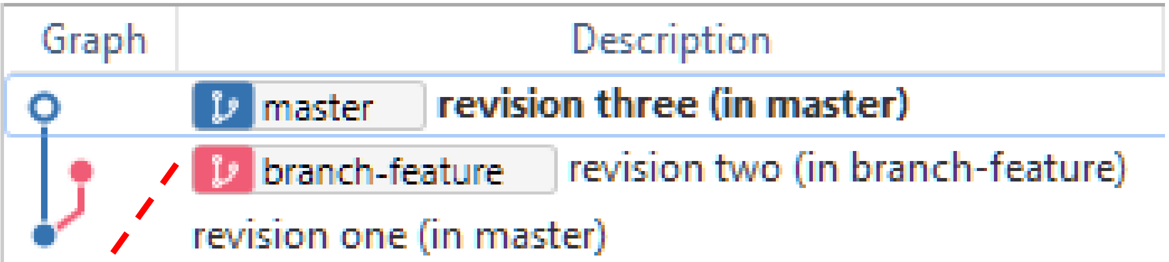
Graph	Description
○	master **revision three (in master)**
●	branch-feature revision two (in branch-feature)
●	revision one (in master)



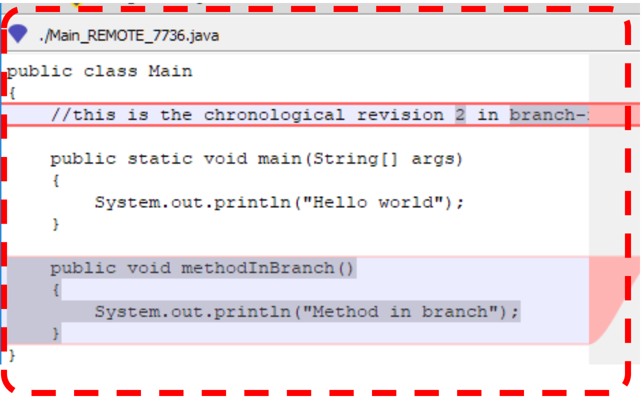
./Main_REMOTE_7736.java

```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```

3-way merge (Perforce P4Merge)

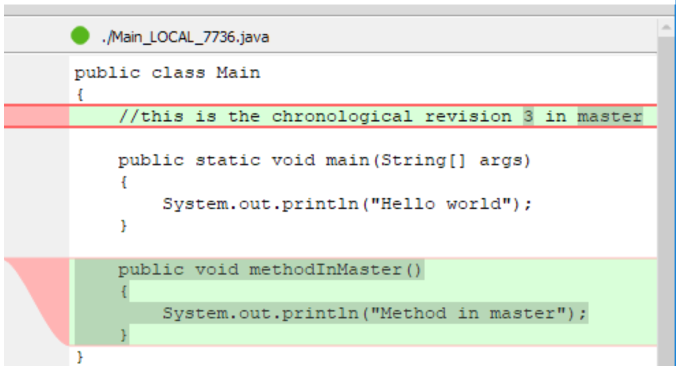


Graph	Description
	master revision three (in master)
	branch-feature revision two (in branch-feature)
	revision one (in master)



./Main_REMOTE_7736.java

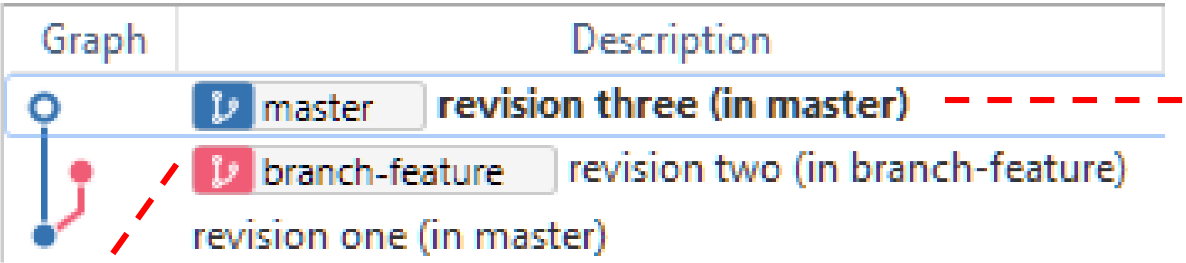
```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```



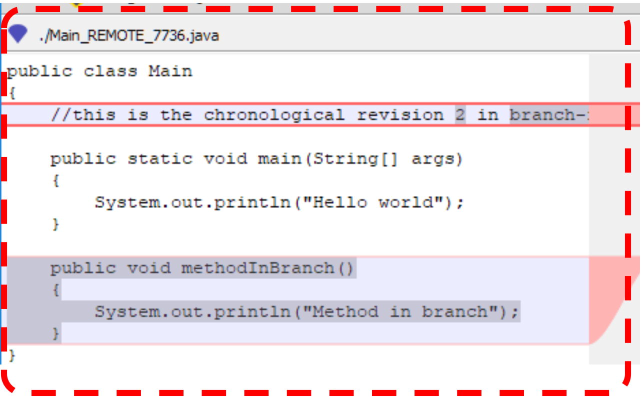
./Main_LOCAL_7736.java

```
public class Main
{
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

3-way merge (Perforce P4Merge)

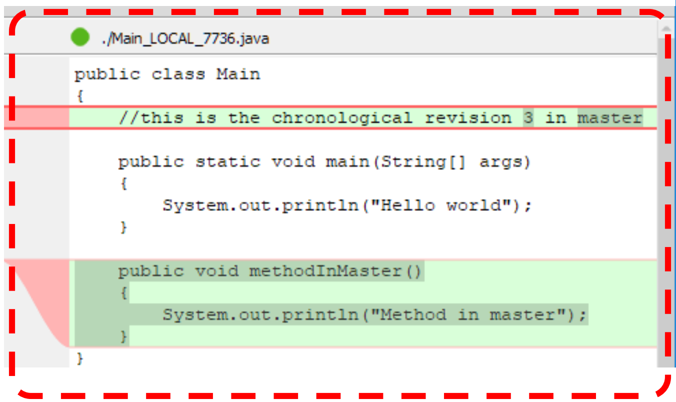


Graph	Description
○	master **revision three (in master)**
●	branch-feature revision two (in branch-feature)
●	revision one (in master)



./Main_REMOTE_7736.java

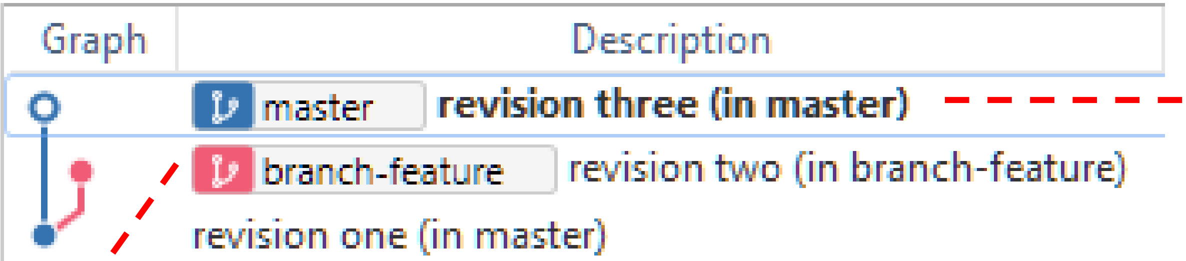
```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```



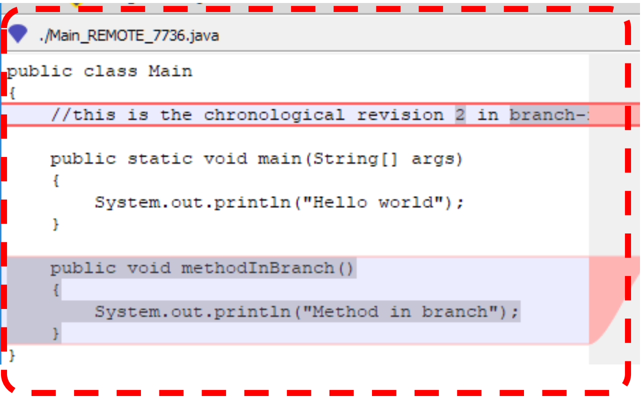
./Main_LOCAL_7736.java

```
public class Main
{
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

3-way merge (Perforce P4Merge)

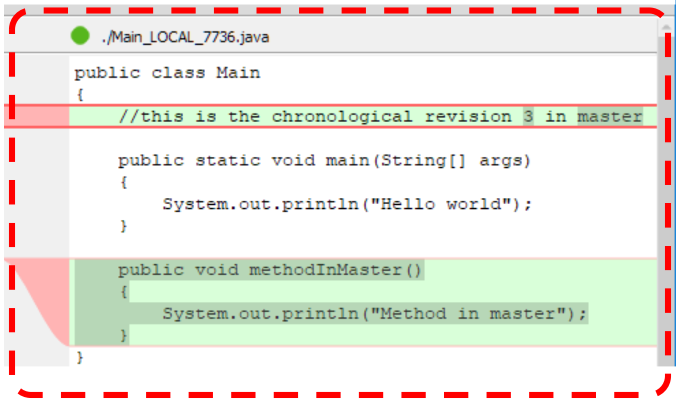


Graph	Description
	master **revision three (in master)**
	branch-feature revision two (in branch-feature)
	revision one (in master)



./Main_REMOTE_7736.java

```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```

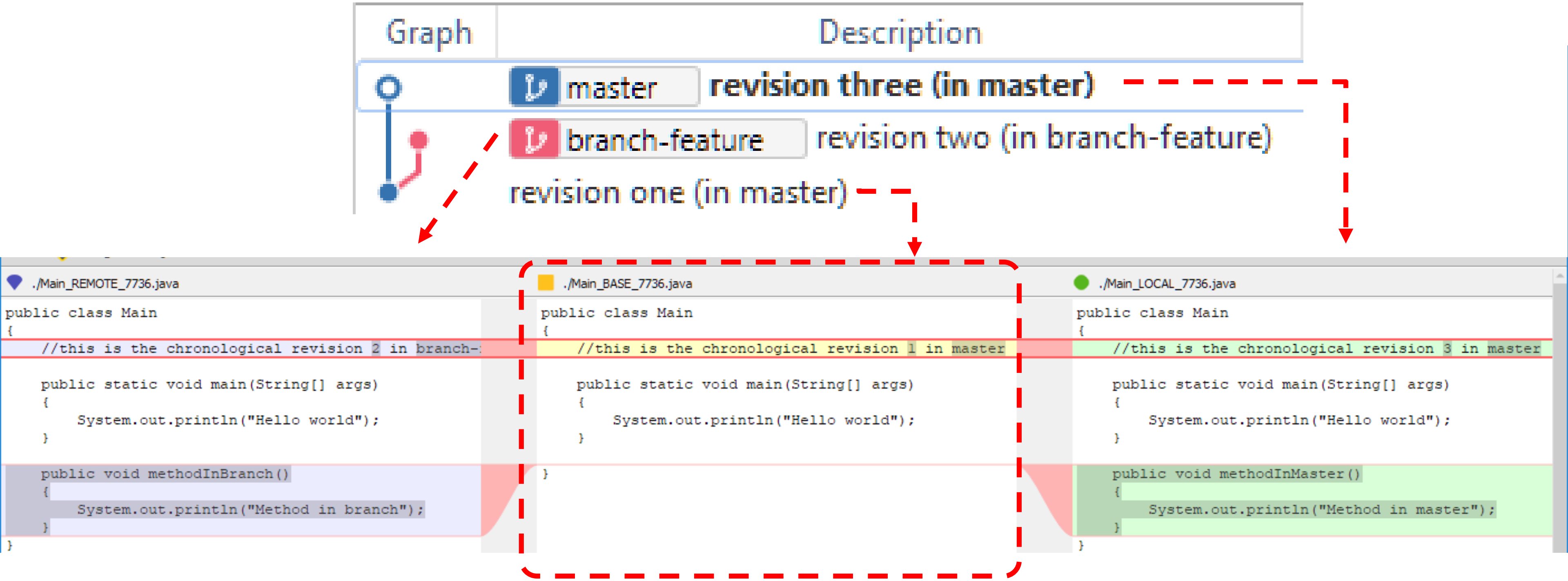


./Main_LOCAL_7736.java

```
public class Main
{
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

Shows the latest changes in both branches (same as a 2-way merge)

3-way merge (Perforce P4Merge)



Graph	Description
	master revision three (in master)
	branch-feature revision two (in branch-feature)
	revision one (in master)

./Main_REMOTE_7736.java

```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```

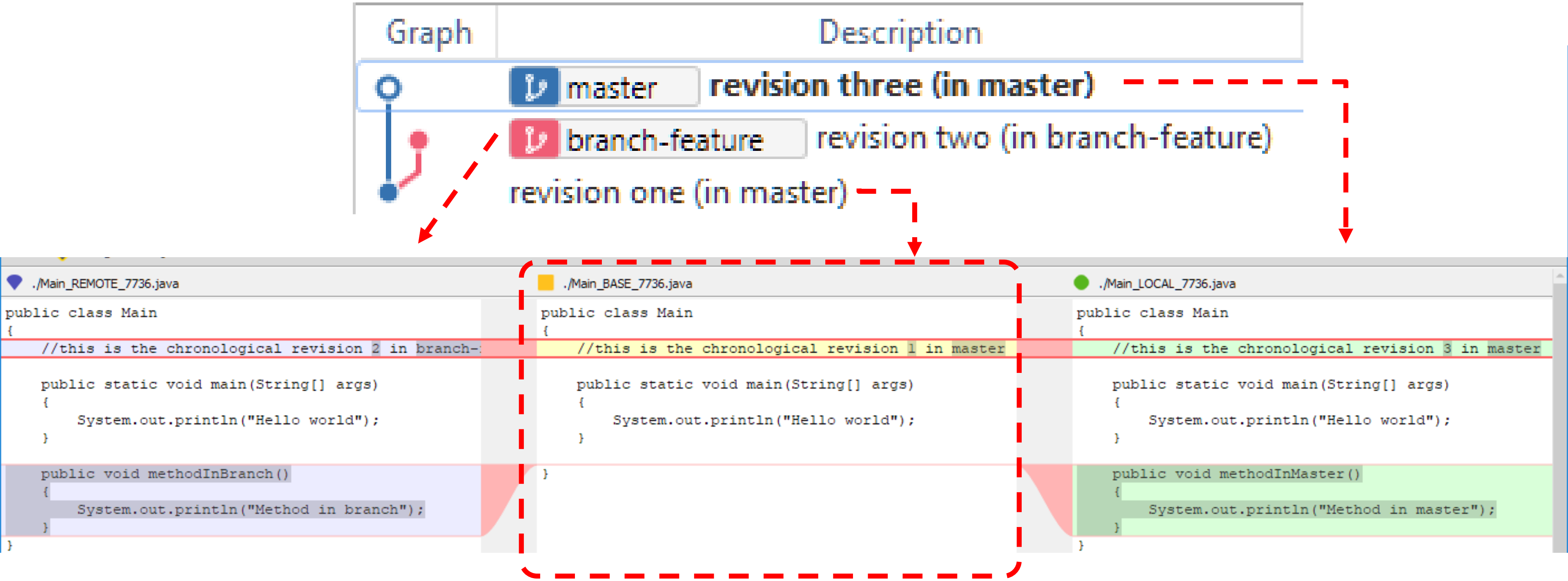
./Main_BASE_7736.java

```
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }
}
```

./Main_LOCAL_7736.java

```
public class Main
{
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

3-way merge (Perforce P4Merge)



./Main_REMOTE_7736.java

```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```

./Main_BASE_7736.java

```
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

}
```

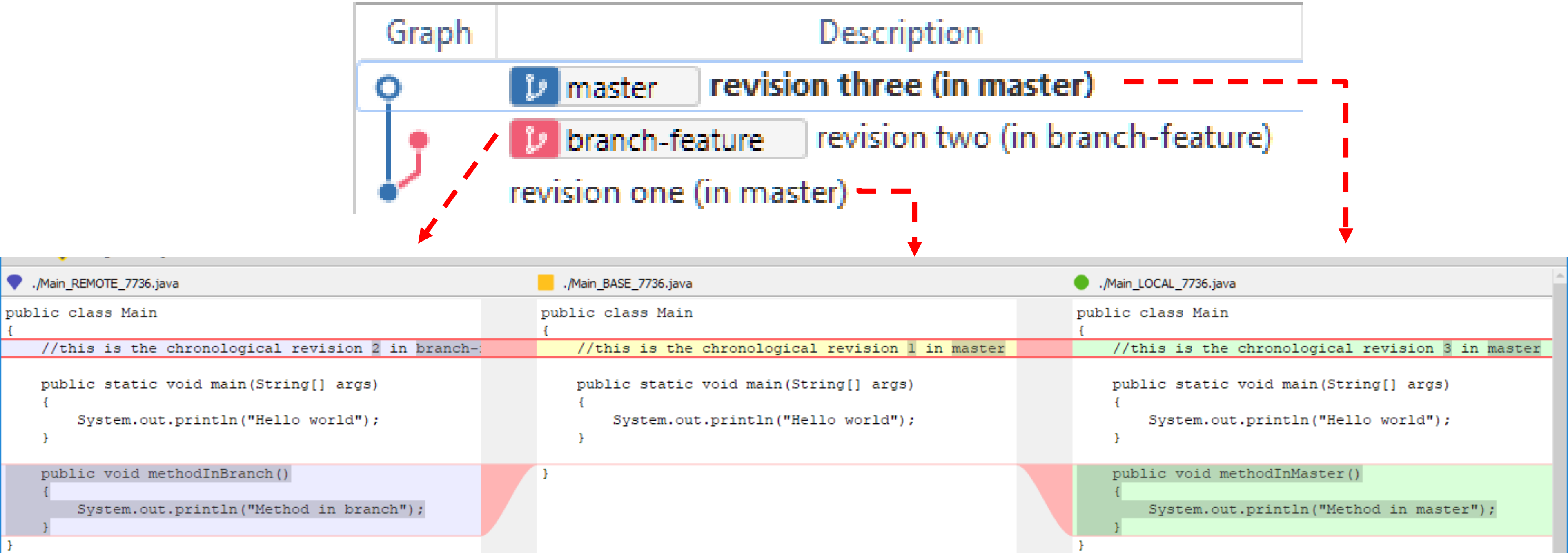
./Main_LOCAL_7736.java

```
public class Main
{
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

Common ancestor (i.e., original version of the file before the changes in master and branch-feature)

- Provides context of changes
- Makes 3-way merge less error-prone than 2-way merge

3-way merge (Perforce P4Merge)



Graph	Description
	master **revision three (in master)**
	branch-feature revision two (in branch-feature)
	revision one (in master)

./Main_REMOTE_7736.java

```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```

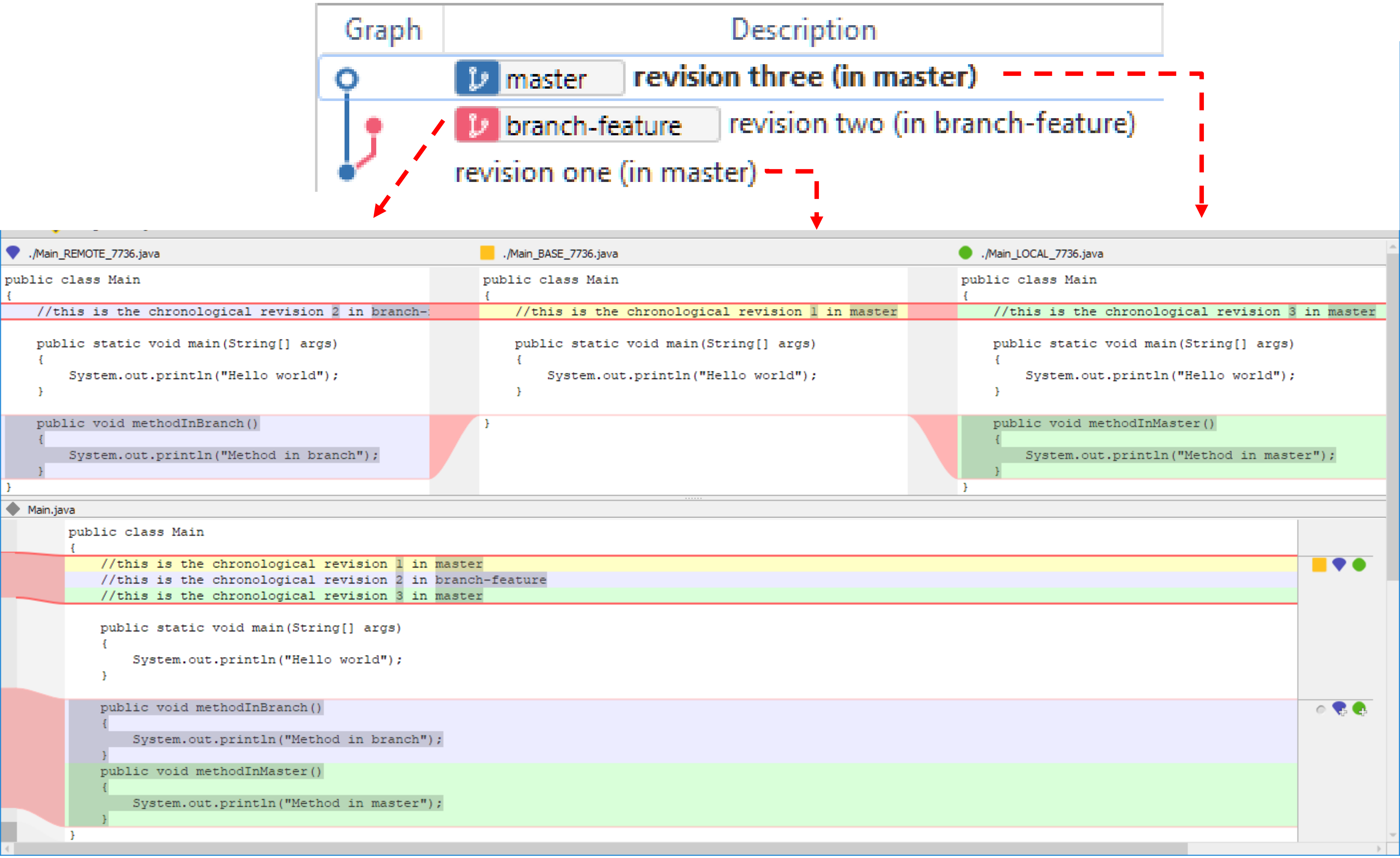
./Main_BASE_7736.java

```
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }
}
```

./Main_LOCAL_7736.java

```
public class Main
{
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

3-way merge (Perforce P4Merge)



Graph	Description
	master revision three (in master)
	branch-feature revision two (in branch-feature)
	revision one (in master)

./Main_REMOTE_7736.java

```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```

./Main_BASE_7736.java

```
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

}
```

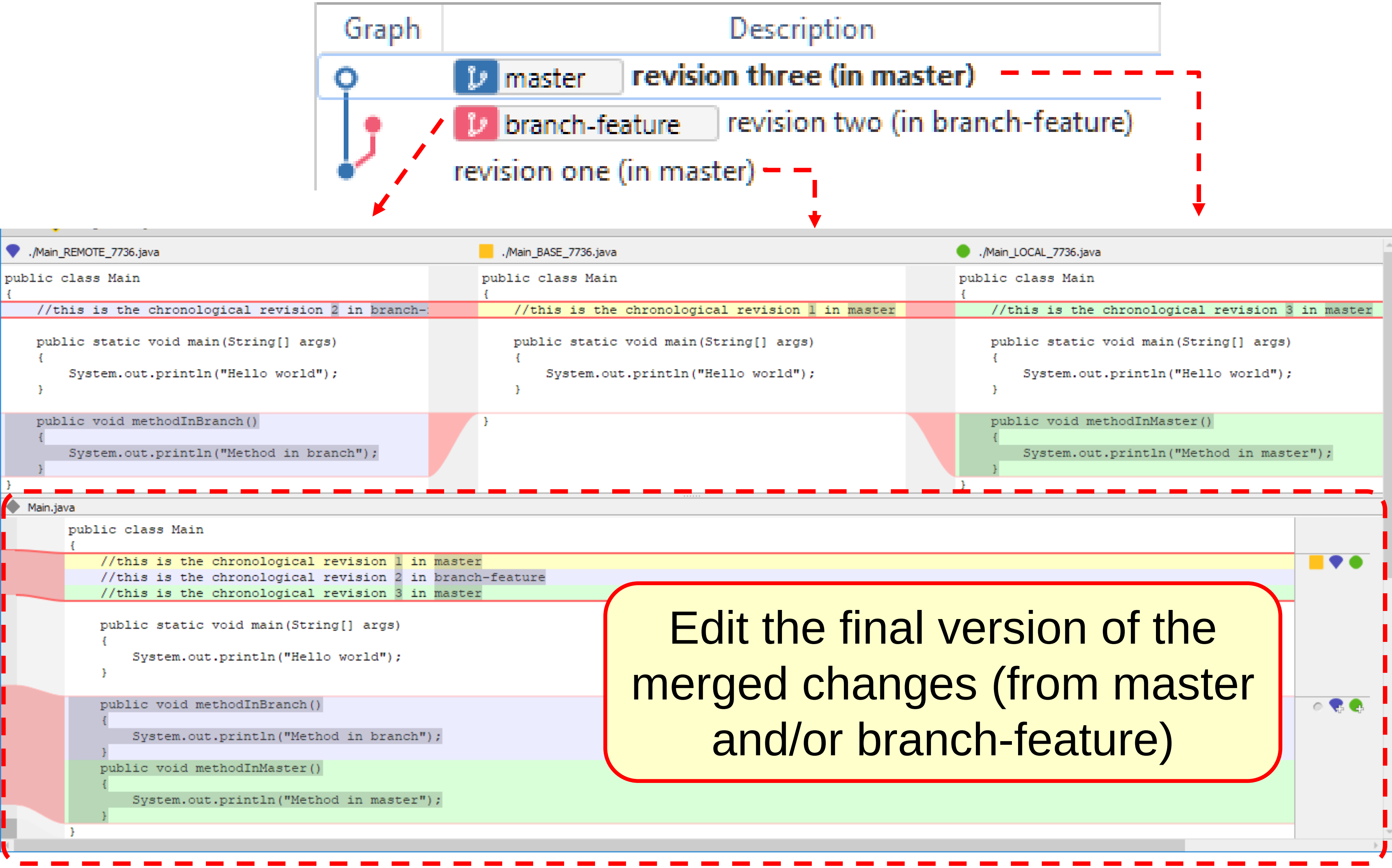
./Main_LOCAL_7736.java

```
public class Main
{
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

Main.java

```
public class Main
{
    //this is the chronological revision 1 in master
    //this is the chronological revision 2 in branch-feature
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

3-way merge (Perforce P4Merge)



Graph	Description
	master revision three (in master)
	branch-feature revision two (in branch-feature)
	revision one (in master)

./Main_REMOTE_7736.java

```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```

./Main_BASE_7736.java

```
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }
}
```

./Main_LOCAL_7736.java

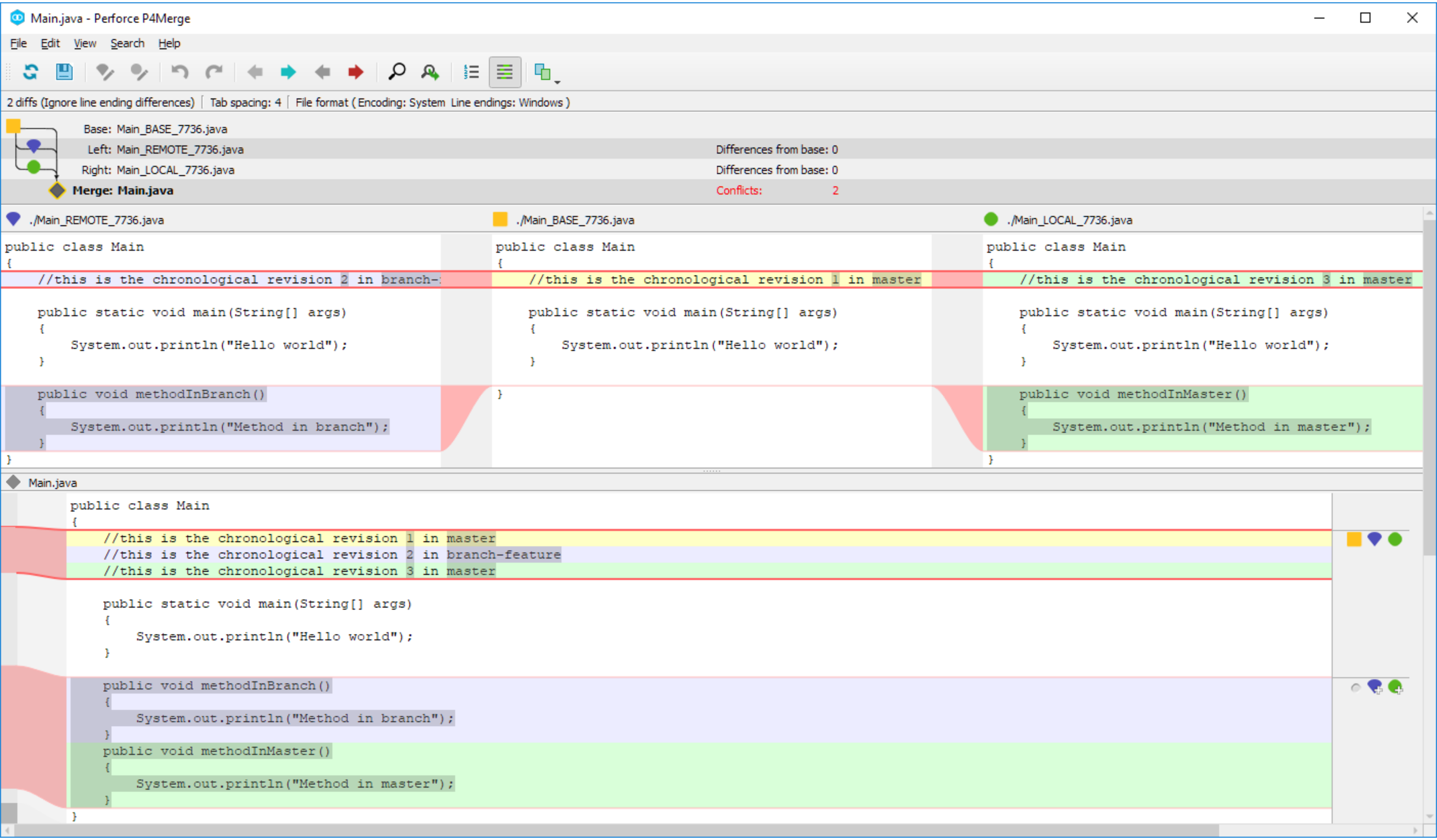
```
public class Main
{
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

Main.java

```
public class Main
{
    //this is the chronological revision 1 in master
    //this is the chronological revision 2 in branch-feature
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

Edit the final version of the merged changes (from master and/or branch-feature)

3-way merge (Perforce P4Merge)



2 diffs (Ignore line ending differences) | Tab spacing: 4 | File format (Encoding: System Line endings: Windows)

Base:	Main_BASE_7736.java	
Left:	Main_REMOTE_7736.java	Differences from base: 0
Right:	Main_LOCAL_7736.java	Differences from base: 0
Merge:	**Main.java**	Conflicts: 2

./Main_REMOTE_7736.java

```
public class Main
{
    //this is the chronological revision 2 in branch-

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
}
```

./Main_BASE_7736.java

```
public class Main
{
    //this is the chronological revision 1 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

}
```

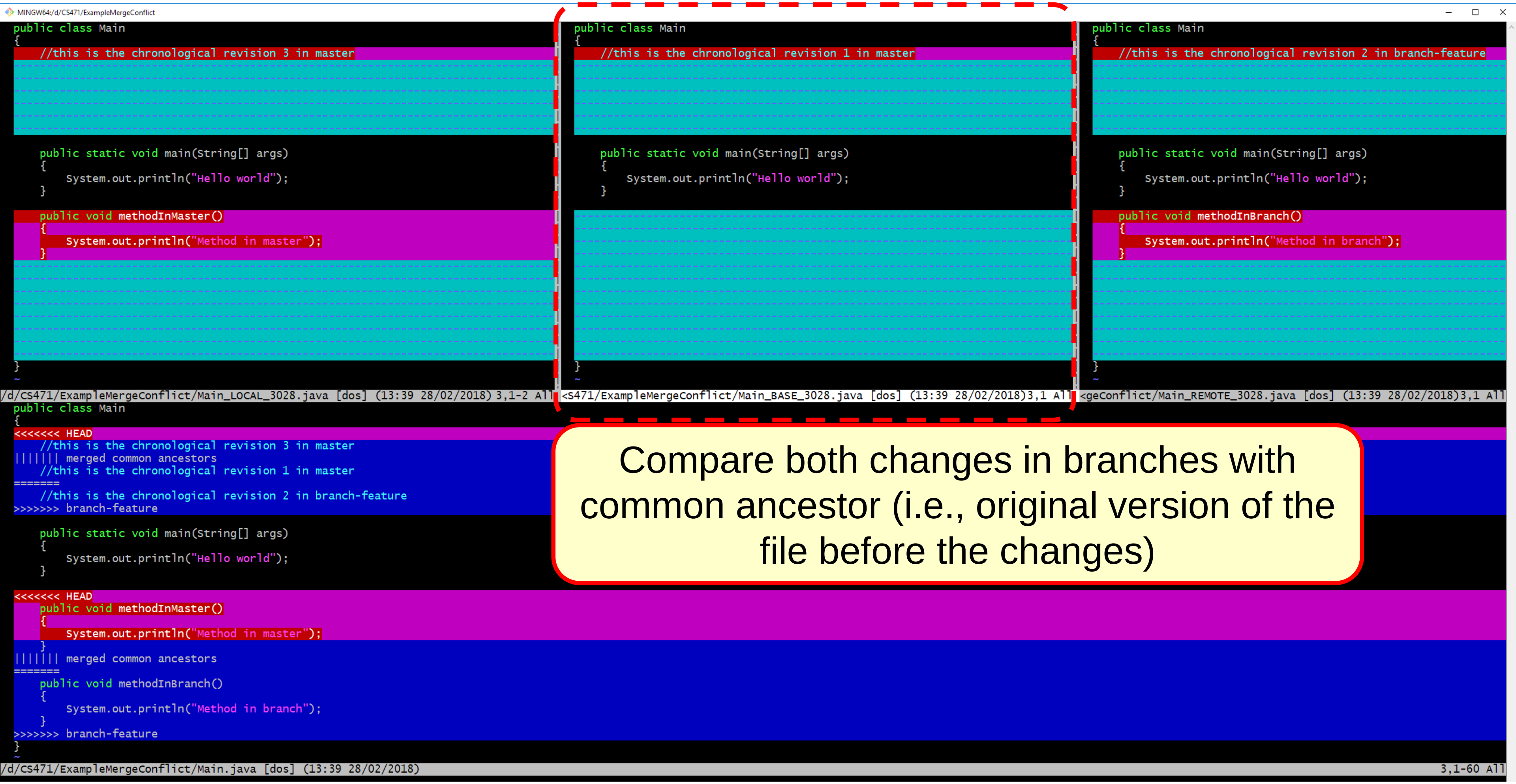
./Main_LOCAL_7736.java

```
public class Main
{
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

Main.java

```
public class Main
{
    //this is the chronological revision 1 in master
    //this is the chronological revision 2 in branch-feature
    //this is the chronological revision 3 in master

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
}
```

The default 3-way merge using vimdiff follows the same rationale/structure as P4Merge



MINGW64:/d/CS471/ExampleMergeConflict

```
public class Main
{
    //this is the chronological revision 3 in master


    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInMaster()
    {
        System.out.println("Method in master");
    }

}
~
/d/CS471/ExampleMergeConflict/Main_LOCAL_3028.java [dos] (13:39 28/02/2018) 3,1-2 All
```

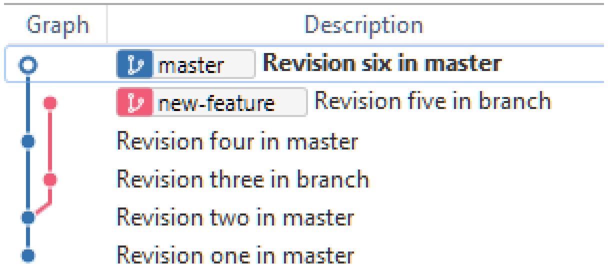
```
public class Main
{
    //this is the chronological revision 1 in master


    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

}
~
<S471/ExampleMergeConflict/Main_BASE_3028.java [dos] (13:39 28/02/2018)3,1 All
```

```
public class Main
{
    //this is the chronological revision 2 in branch-feature


    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }

}
~
<geConflict/Main_REMOTE_3028.java [dos] (13:39 28/02/2018)3,1 All
```

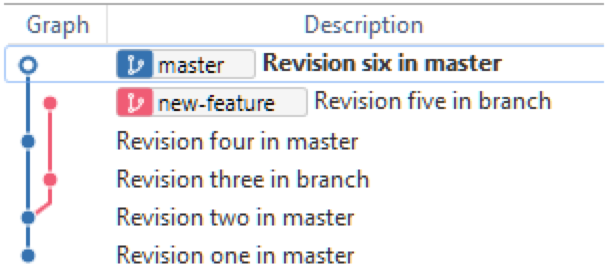
```
public class Main
{
<<<<<<< HEAD
    //this is the chronological revision 3 in master
||||||| merged common ancestors
    //this is the chronological revision 1 in master
=======
    //this is the chronological revision 2 in branch-feature
>>>>>>> branch-feature

    public static void main(String[] args)
    {
        System.out.println("Hello world");
    }

<<<<<<< HEAD
    public void methodInMaster()
    {
        System.out.println("Method in master");
    }
||||||| merged common ancestors
=======
    public void methodInBranch()
    {
        System.out.println("Method in branch");
    }
>>>>>>> branch-feature
}
~
/d/CS471/ExampleMergeConflict/Main.java [dos] (13:39 28/02/2018)                3,1-60 All
```

Compare both changes in branches with common ancestor (i.e., original version of the file before the changes)

Git Merge vs. Rebase Example



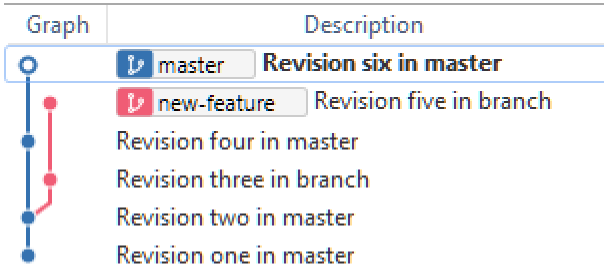
Graph	Description
○	master **Revision six in master**
●	new-feature Revision five in branch
●	Revision four in master
●	Revision three in branch
●	Revision two in master
●	Revision one in master



Graph	Description
	master **Revision six in master**
	new-feature Revision five in branch
	Revision four in master
	Revision three in branch
	Revision two in master
	Revision one in master

In master, run:

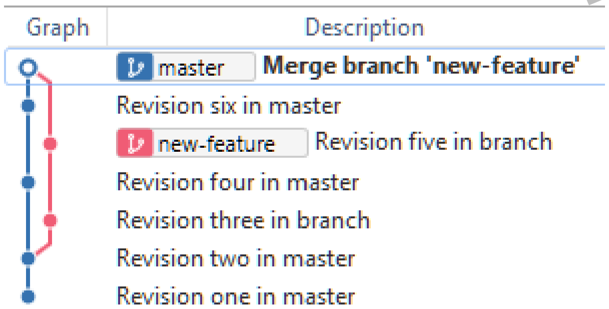
```
git merge new-feature
```



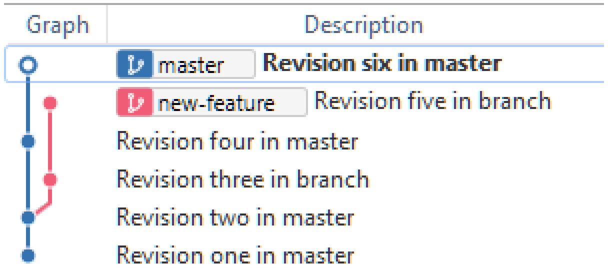
Graph	Description
	master **Revision six in master**
	new-feature Revision five in branch
	Revision four in master
	Revision three in branch
	Revision two in master
	Revision one in master

In master, run:

```
git merge new-feature
```



Graph	Description
	master **Merge branch 'new-feature'**
	Revision six in master
	new-feature Revision five in branch
	Revision four in master
	Revision three in branch
	Revision two in master
	Revision one in master



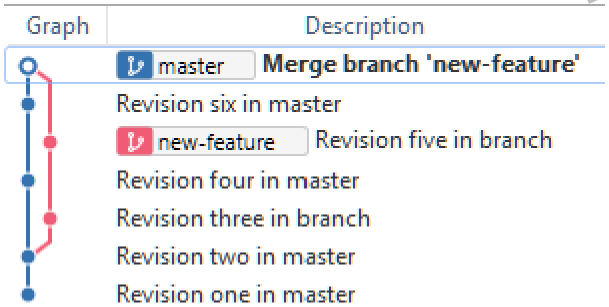
Graph	Description
	master **Revision six in master**
	new-feature Revision five in branch
	Revision four in master
	Revision three in branch
	Revision two in master
	Revision one in master

In branch, run:

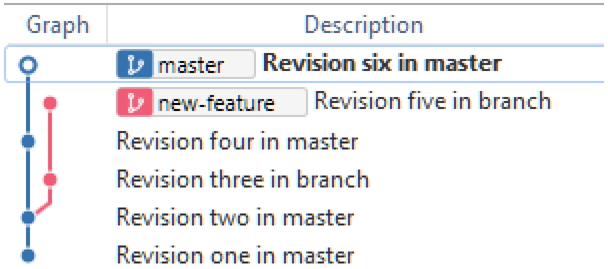
```
git rebase master
```

In master, run:

```
git merge new-feature
```



Graph	Description
	master **Merge branch 'new-feature'**
	Revision six in master
	new-feature Revision five in branch
	Revision four in master
	Revision three in branch
	Revision two in master
	Revision one in master



Graph Description

master **Revision six in master**

new-feature Revision five in branch

Revision four in master

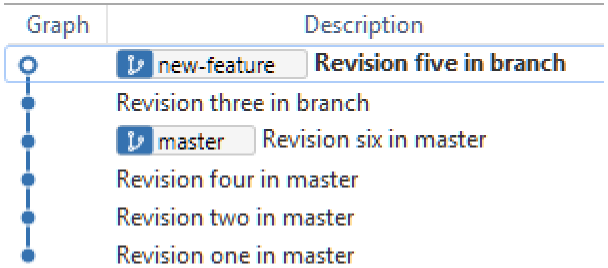
Revision three in branch

Revision two in master

Revision one in master

In branch, run:

```
git rebase master
```



Graph Description

new-feature **Revision five in branch**

Revision three in branch

master Revision six in master

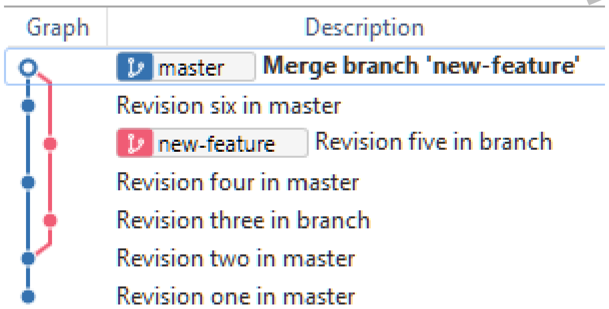
Revision four in master

Revision two in master

Revision one in master

In master, run:

```
git merge new-feature
```



Graph Description

master **Merge branch 'new-feature'**

Revision six in master

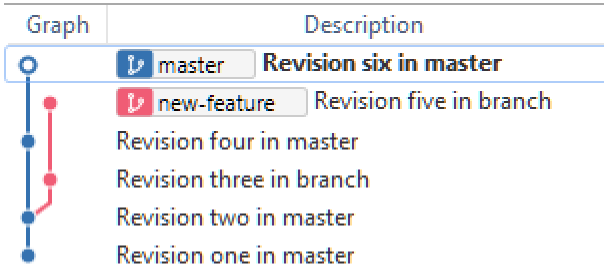
new-feature Revision five in branch

Revision four in master

Revision three in branch

Revision two in master

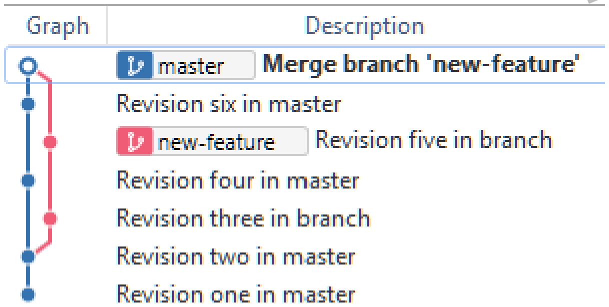
Revision one in master



Graph	Description
	master **Revision six in master**
	new-feature Revision five in branch
	Revision four in master
	Revision three in branch
	Revision two in master
	Revision one in master

In master, run:

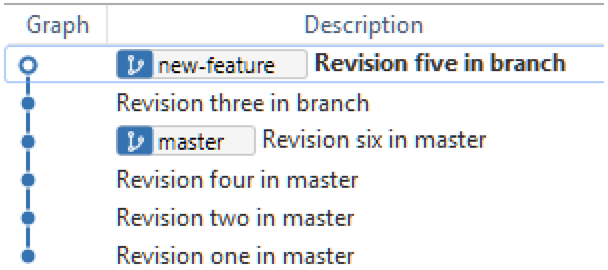
```
git merge new-feature
```



Graph	Description
	master **Merge branch 'new-feature'**
	Revision six in master
	new-feature Revision five in branch
	Revision four in master
	Revision three in branch
	Revision two in master
	Revision one in master

In branch, run:

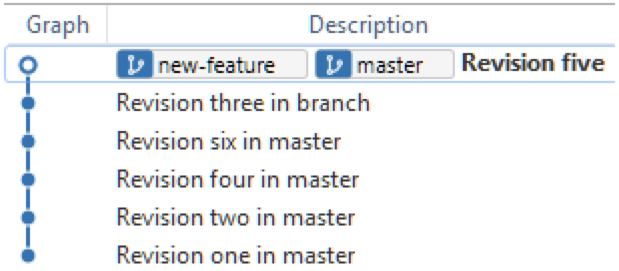
```
git rebase master
```



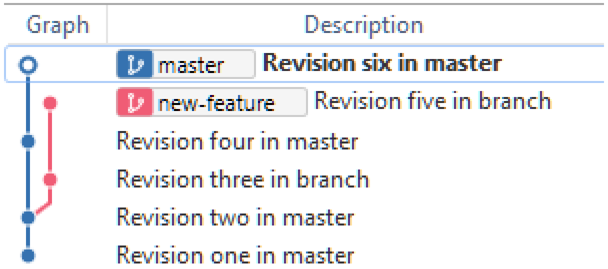
Graph	Description
	new-feature **Revision five in branch**
	Revision three in branch
	master Revision six in master
	Revision four in master
	Revision two in master
	Revision one in master

In master, run:

```
git merge new-feature
```



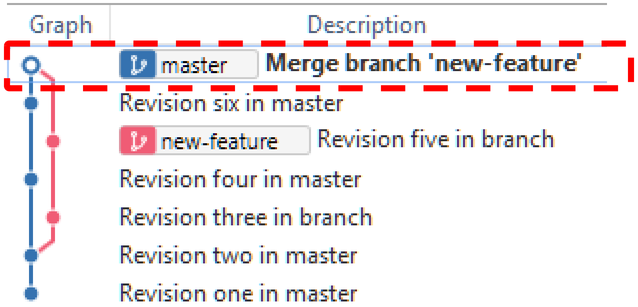
Graph	Description
	new-feature master **Revision five**
	Revision three in branch
	Revision six in master
	Revision four in master
	Revision two in master
	Revision one in master



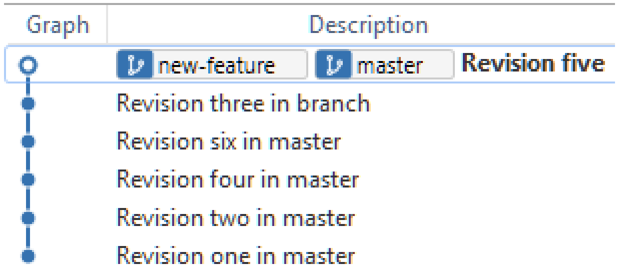
Graph	Description
	master **Revision six in master**
	new-feature Revision five in branch
	Revision four in master
	Revision three in branch
	Revision two in master
	Revision one in master

merge

rebase/merge



Graph	Description
	master **Merge branch 'new-feature'**
	Revision six in master
	new-feature Revision five in branch
	Revision four in master
	Revision three in branch
	Revision two in master
	Revision one in master



Graph	Description
	new-feature master **Revision five**
	Revision three in branch
	Revision six in master
	Revision four in master
	Revision two in master
	Revision one in master

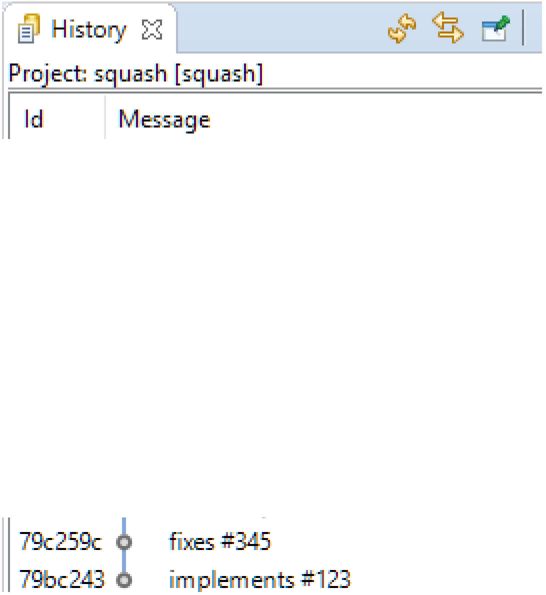
- Extra merge commit
- Non linear history
+ Preserves time information

+ No Extra commit
+ Linear history
- “Rewrites” history

Git Squash Example

Git Squash Example

- Open the repository history in Eclipse (“Team” → ”Show in History”)



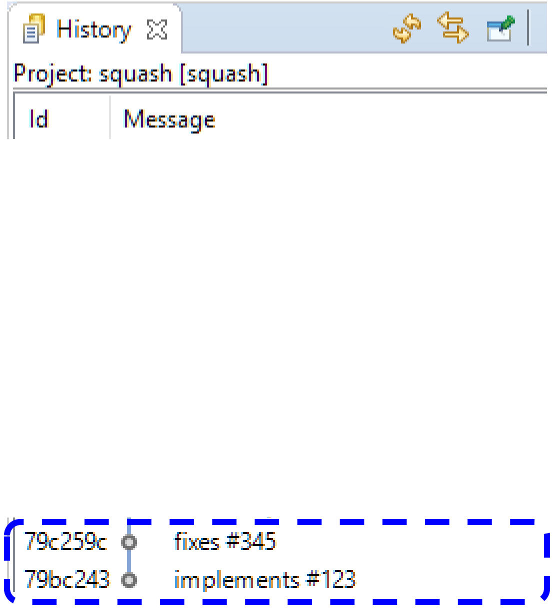
History

Project: squash [squash]

Id	Message
79c259c	fixes #345
79bc243	implements #123

Git Squash Example

- Open the repository history in Eclipse ("Team" → "Show in History")



History

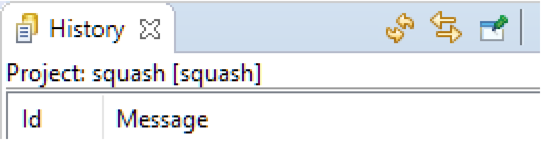
Project: squash [squash]

Id	Message
79c259c	fixes #345
79bc243	implements #123

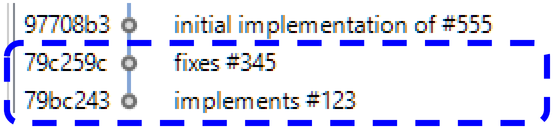
Existing commits

Git Squash Example

- We start working on #555, and keep adding commits related to it



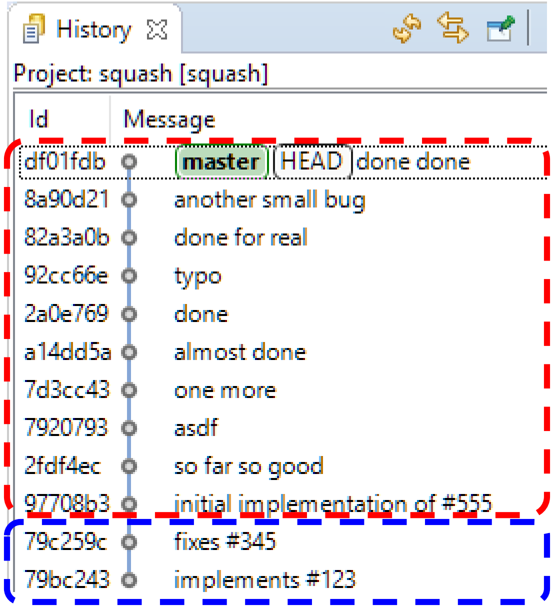
Project: squash [squash]



Id	Message
97708b3	initial implementation of #555
79c259c	fixes #345
79bc243	implements #123

Existing commits

Git Squash Example



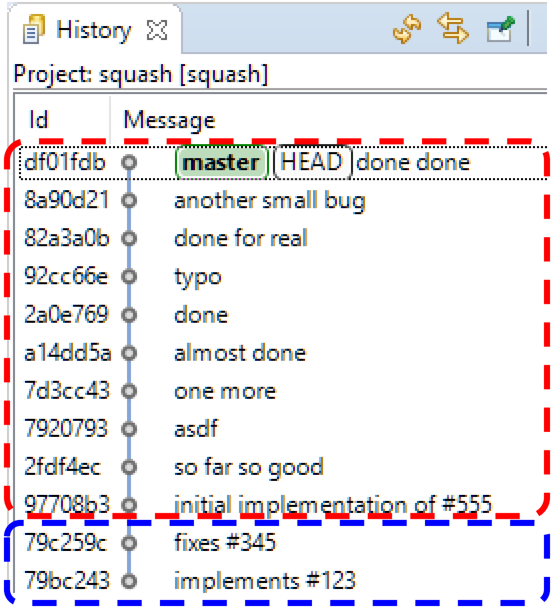
Project: squash [squash]

Id	Message
df01fdb	master HEAD done done
8a90d21	another small bug
82a3a0b	done for real
92cc66e	typo
2a0e769	done
a14dd5a	almost done
7d3cc43	one more
7920793	asdf
2fdf4ec	so far so good
97708b3	initial implementation of #555
79c259c	fixes #345
79bc243	implements #123

Multiple commits related to #555

Existing commits

Git Squash Example



Project: squash [squash]

Id	Message
df01fdb	master HEAD done done
8a90d21	another small bug
82a3a0b	done for real
92cc66e	typo
2a0e769	done
a14dd5a	almost done
7d3cc43	one more
7920793	asdf
2fdf4ec	so far so good
97708b3	initial implementation of #555
79c259c	fixes #345
79bc243	implements #123

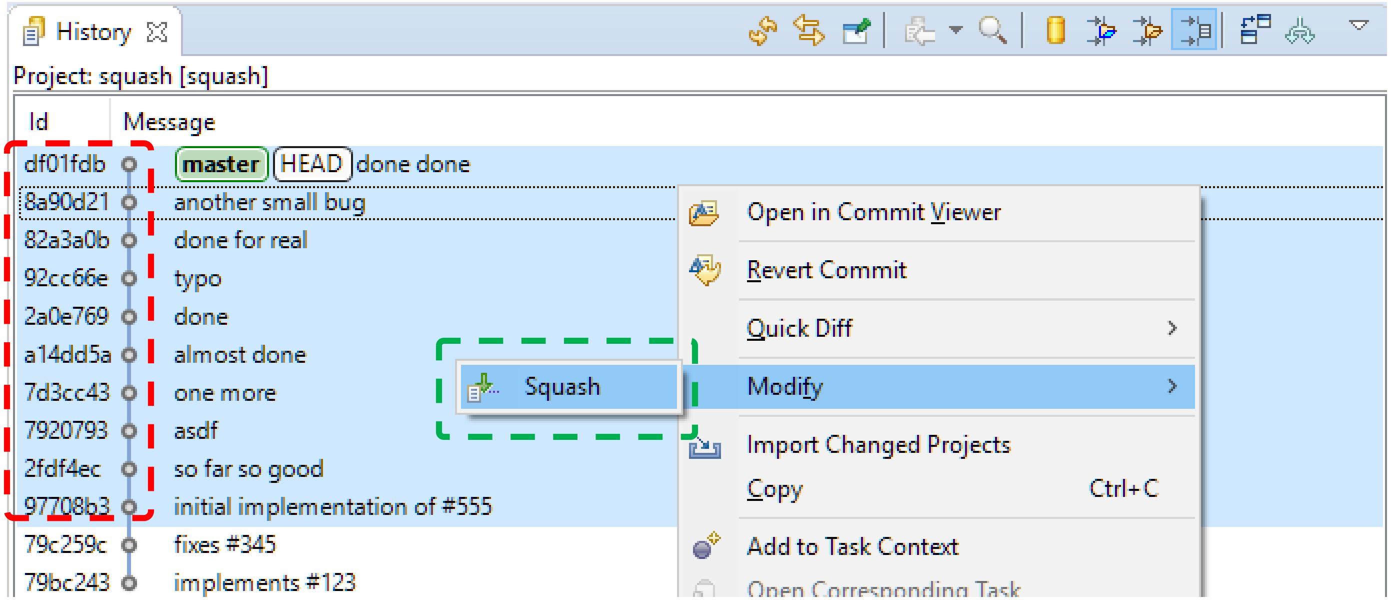
Existing commits

Multiple commits related to #555

- Good for debugging/implementing
  - Allow “checkpoints”
- Bad for other team members to read
  - History clutter
- Solution: squash multiple related commits into one

Git Squash Example

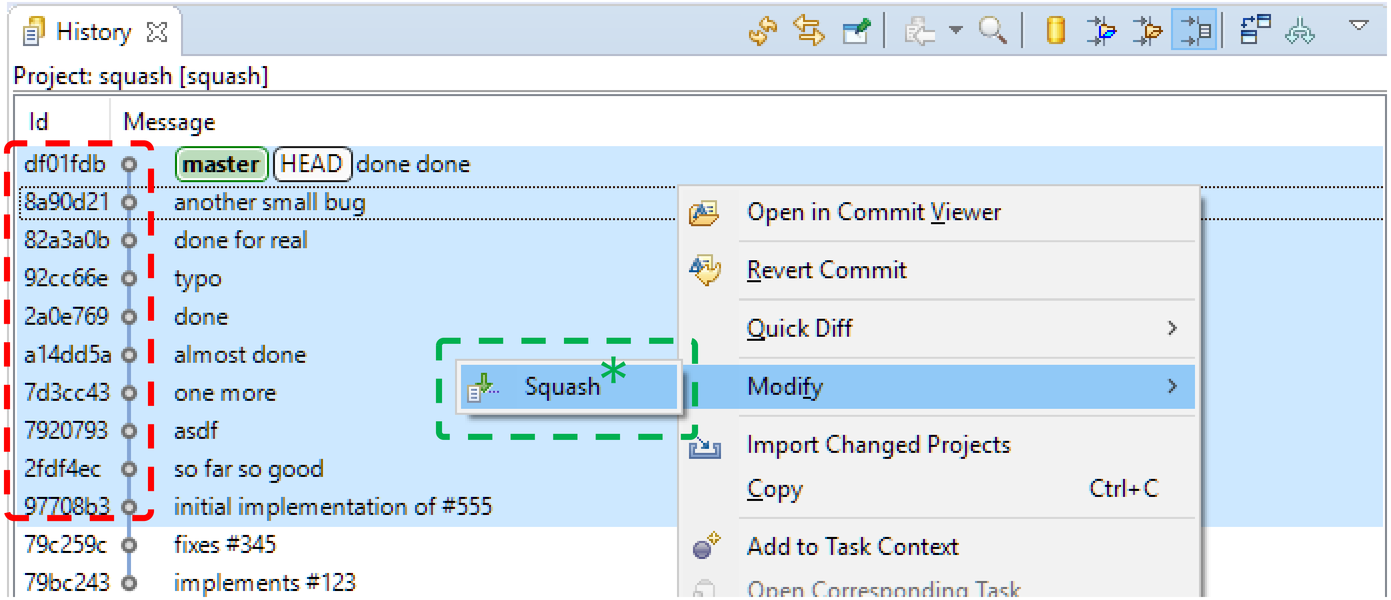
- Select group of related commits that will be “compressed” / ”squashed” into a new commit



Id	Message
df01fdb	master HEAD done done
8a90d21	another small bug
82a3a0b	done for real
92cc66e	typo
2a0e769	done
a14dd5a	almost done
7d3cc43	one more
7920793	asdf
2fdf4ec	so far so good
97708b3	initial implementation of #555
79c259c	fixes #345
79bc243	implements #123

Git Squash Example

- Select group of related commits that will be "compressed" / "squashed" into a new commit



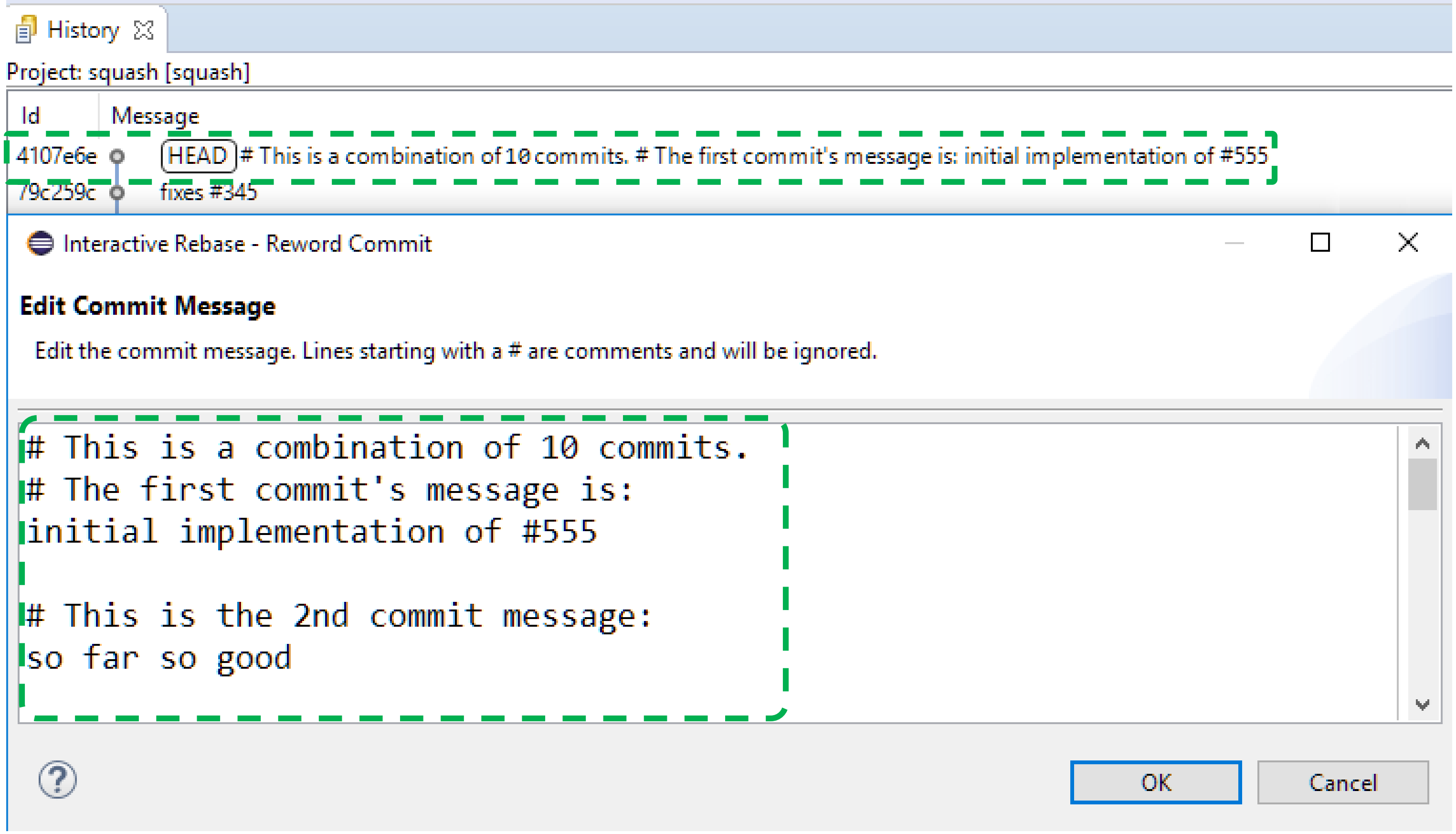
Project: squash [squash]

Id	Message
df01fdb	master HEAD done done
8a90d21	another small bug
82a3a0b	done for real
92cc66e	typo
2a0e769	done
a14dd5a	almost done
7d3cc43	one more
7920793	asdf
2fdf4ec	so far so good
97708b3	initial implementation of #555
79c259c	fixes #345
79bc243	implements #123

```
* git rebase -i HEAD~10
```

Git Squash Example

- The 10 commits were squashed into one, and we can edit the commit log message of the resulting commit



History

Project: squash [squash]

Id	Message
4107e6e	HEAD # This is a combination of 10 commits. # The first commit's message is: initial implementation of #555
79c259c	fixes #345

Interactive Rebase - Reword Commit

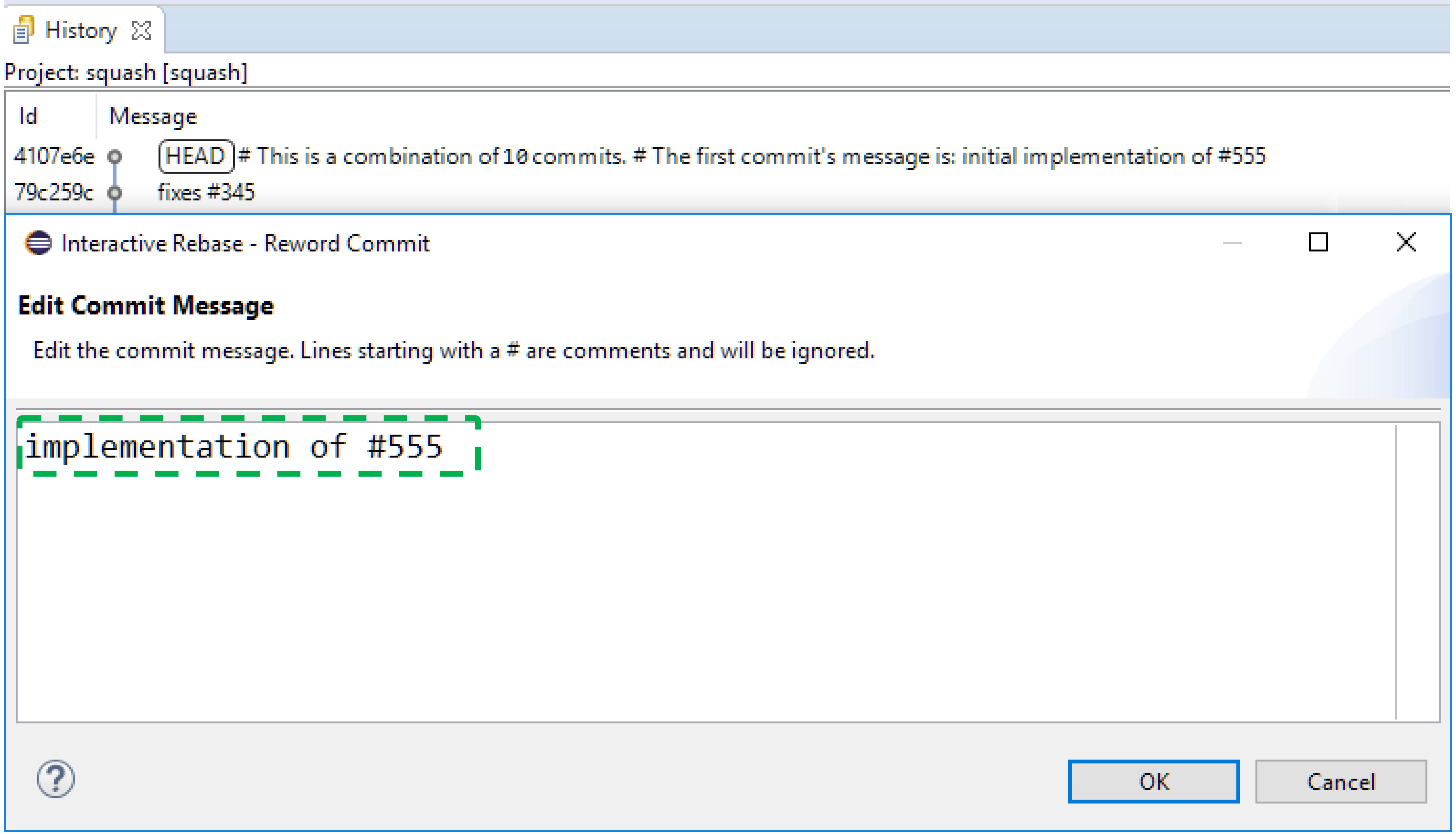
Edit Commit Message

Edit the commit message. Lines starting with a # are comments and will be ignored.

```
# This is a combination of 10 commits.
# The first commit's message is:
initial implementation of #555

# This is the 2nd commit message:
so far so good
```

Git Squash Example

- Updated commit log message



Project: squash [squash]

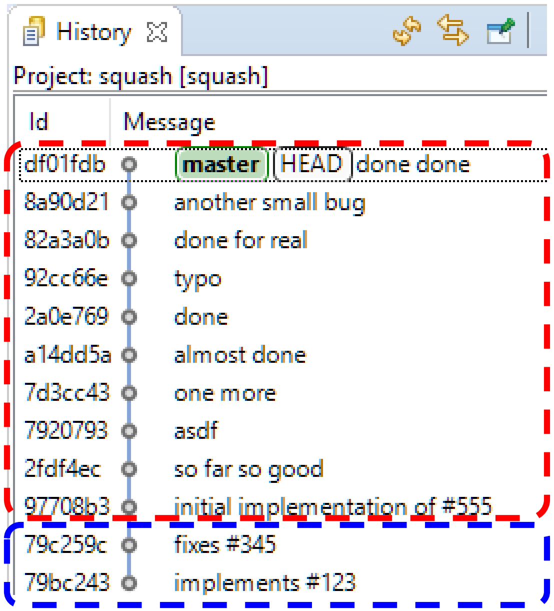
Id	Message
4107e6e	HEAD # This is a combination of 10 commits. # The first commit's message is: initial implementation of #555
79c259c	fixes #345

Edit Commit Message

Edit the commit message. Lines starting with a # are comments and will be ignored.

```
implementation of #555
```

Git Squash Example



Project: squash [squash]

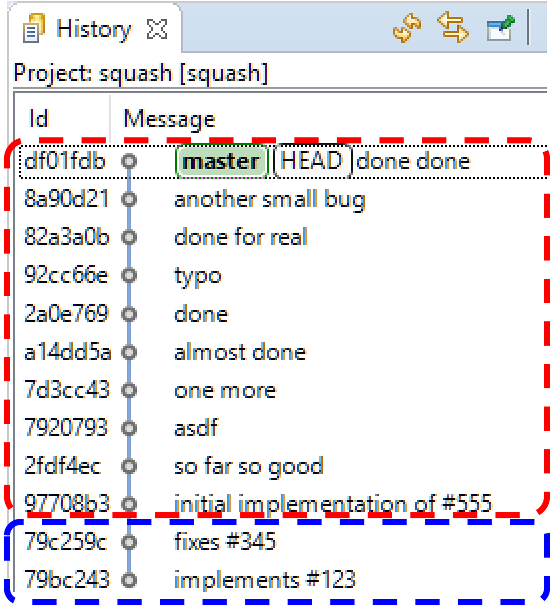
Id	Message
df01fdb	master HEAD done done
8a90d21	another small bug
82a3a0b	done for real
92cc66e	typo
2a0e769	done
a14dd5a	almost done
7d3cc43	one more
7920793	asdf
2fdf4ec	so far so good
97708b3	initial implementation of #555
79c259c	fixes #345
79bc243	implements #123

Multiple commits related to #555

Existing commits

Git Squash Example

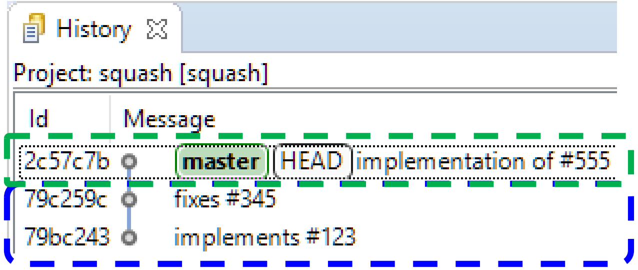
- "squash"-ing works best on task branches and with pull-requests



Id	Message
df01fdb	master HEAD done done
8a90d21	another small bug
82a3a0b	done for real
92cc66e	typo
2a0e769	done
a14dd5a	almost done
7d3cc43	one more
7920793	asdf
2fdf4ec	so far so good
97708b3	initial implementation of #555
79c259c	fixes #345
79bc243	implements #123

Existing commits

Multiple commits related to #555 were squashed into one commit

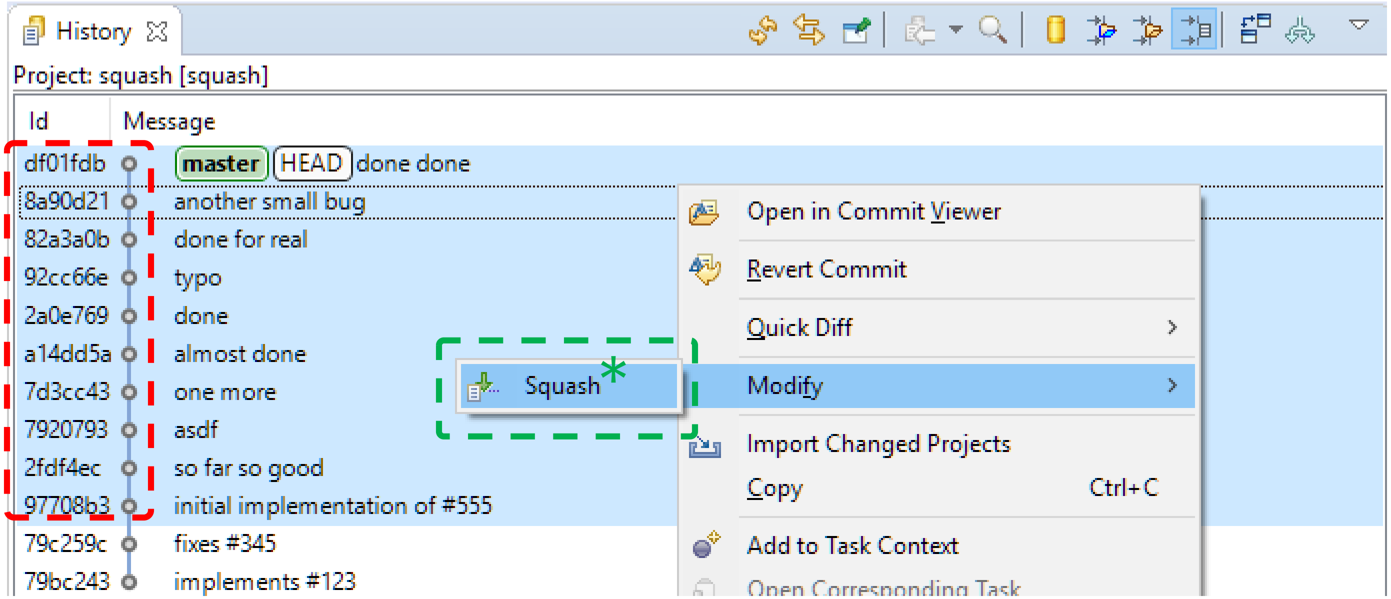


Id	Message
2c57c7b	master HEAD implementation of #555
79c259c	fixes #345
79bc243	implements #123

Existing commits

Git Squash Example

Eclipse GUI

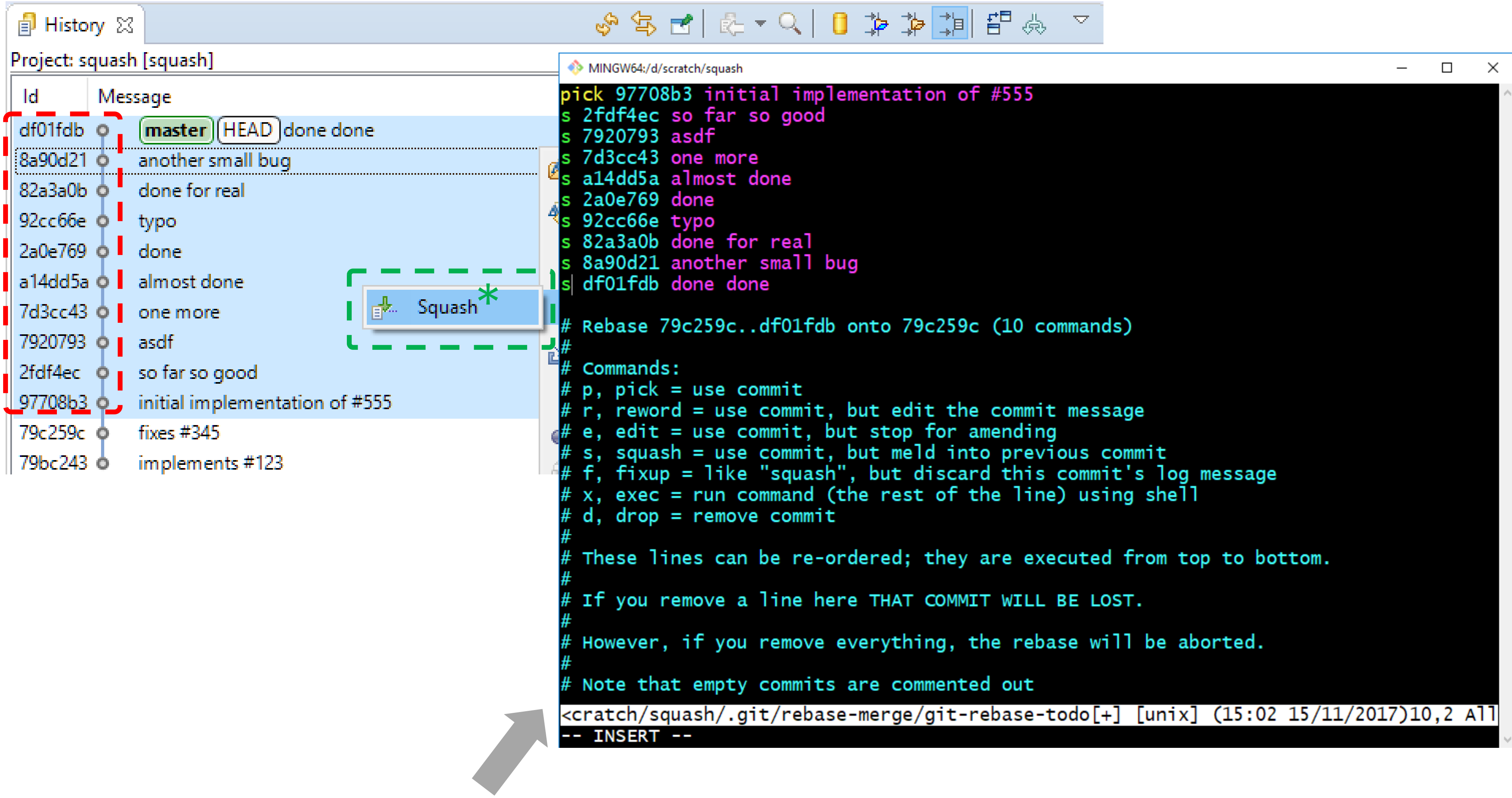


Id	Message
df01fdb	master HEAD done done
8a90d21	another small bug
82a3a0b	done for real
92cc66e	typo
2a0e769	done
a14dd5a	almost done
7d3cc43	one more
7920793	asdf
2fdf4ec	so far so good
97708b3	initial implementation of #555
79c259c	fixes #345
79bc243	implements #123

```
* git rebase -i HEAD~10
```

Git Squash Example

Eclipse GUI vs. command line



Project: squash [squash]

Id	Message
df01fdb	master HEAD done done
8a90d21	another small bug
82a3a0b	done for real
92cc66e	typo
2a0e769	done
a14dd5a	almost done
7d3cc43	one more
7920793	asdf
2fdf4ec	so far so good
97708b3	initial implementation of #555
79c259c	fixes #345
79bc243	implements #123

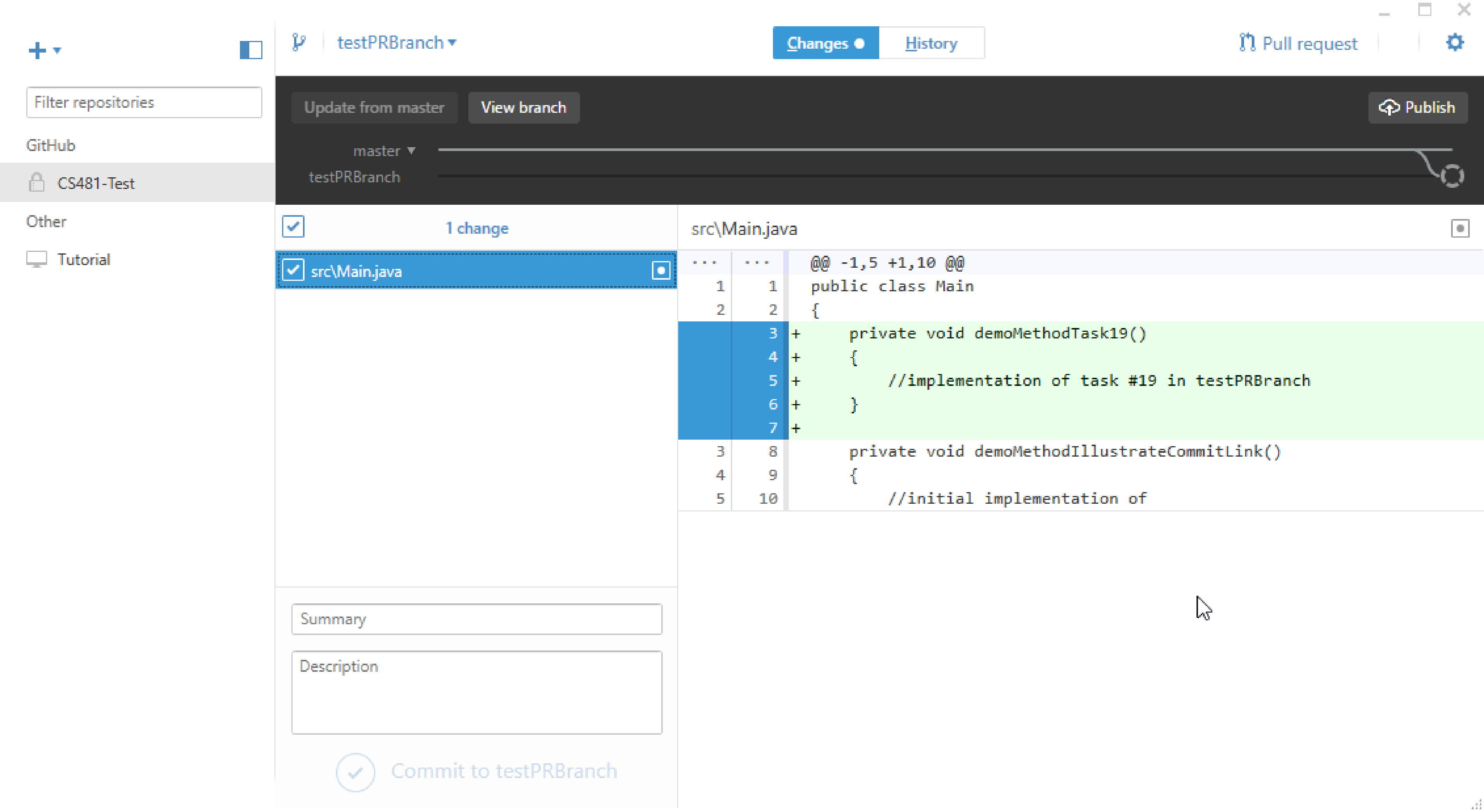
MINGW64:/d/scratch/squash

```
pick 97708b3 initial implementation of #555
s 2fdf4ec so far so good
s 7920793 asdf
s 7d3cc43 one more
s a14dd5a almost done
s 2a0e769 done
s 92cc66e typo
s 82a3a0b done for real
s 8a90d21 another small bug
s df01fdb done done

# Rebase 79c259c..df01fdb onto 79c259c (10 commands)
#
# Commands:
# p, pick = use commit
# r, reword = use commit, but edit the commit message
# e, edit = use commit, but stop for amending
# s, squash = use commit, but meld into previous commit
# f, fixup = like "squash", but discard this commit's log message
# x, exec = run command (the rest of the line) using shell
# d, drop = remove commit
#
# These lines can be re-ordered; they are executed from top to bottom.
#
# If you remove a line here THAT COMMIT WILL BE LOST.
#
# However, if you remove everything, the rebase will be aborted.
#
# Note that empty commits are commented out
<cratch/squash/.git/rebase-merge/git-rebase-todo[+]  [unix]  (15:02 15/11/2017)10,2 All
-- INSERT --
```

```
* git rebase -i HEAD~10
```

Pull Request Example

In testPRBranch (created from master) make some minor changes to the file src/Main.java



src\Main.java

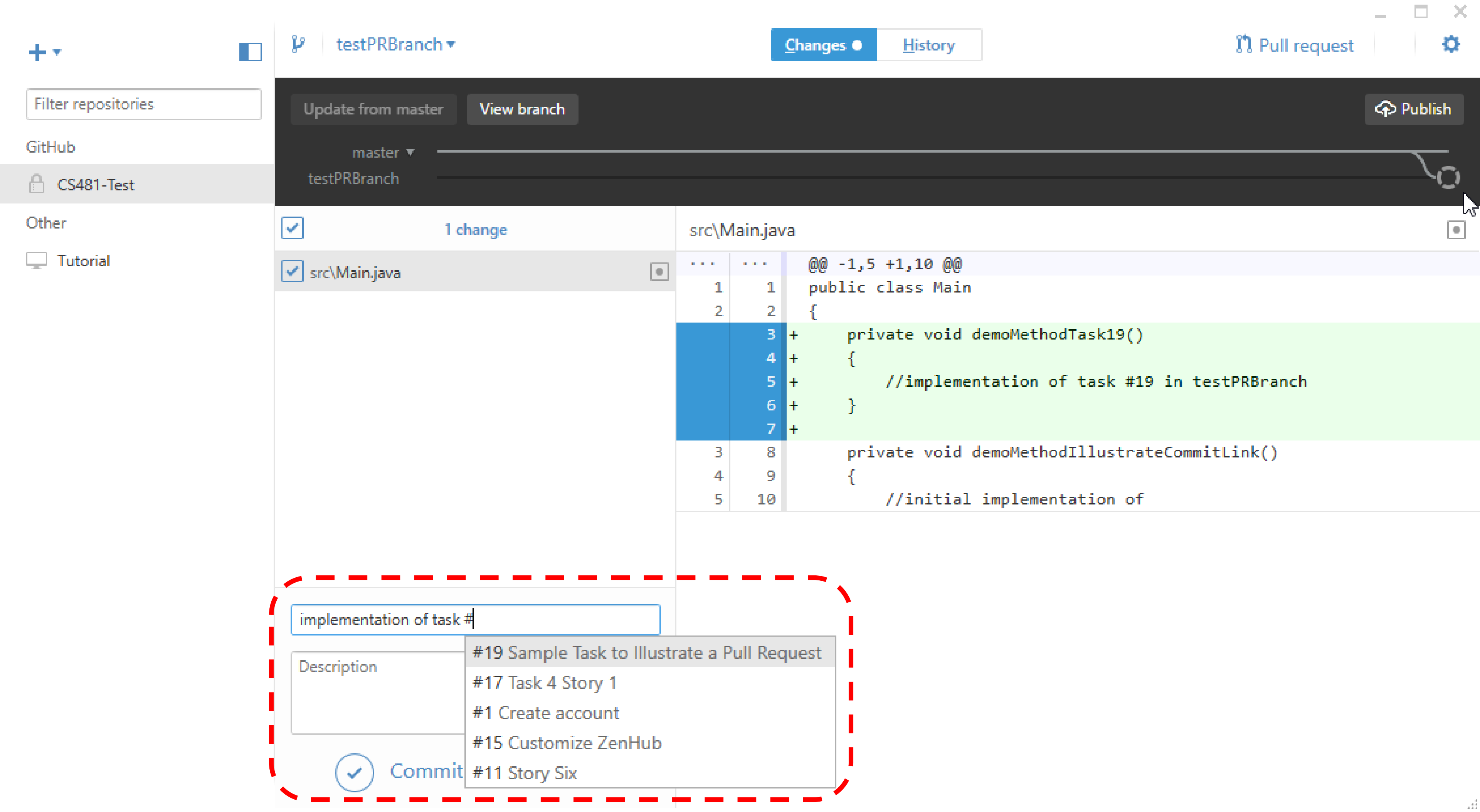
```
@@ -1,5 +1,10 @@
 public class Main
 {
+    private void demoMethodTask19()
+    {
+        //implementation of task #19 in testPRBranch
+    }
+
     private void demoMethodIllustrateCommitLink()
     {
         //initial implementation of
```

What would be a good commit log message?

- Ideally, we want to keep a “single source of truth”, namely the issue/task that the commit is supposed to implement
  - There is no need to add duplicate / long / error-prone / insufficiently detailed messages and commit descriptions (i.e., no need to “reinvent the wheel”)

- Solution: every commit should link to the ID of the issue / task / bug it supposed to implement / fix (e.g., “Implements #123”)

In the commit log message specify the task/bug ID to associate the implementation with the task/bug it implements/fixes



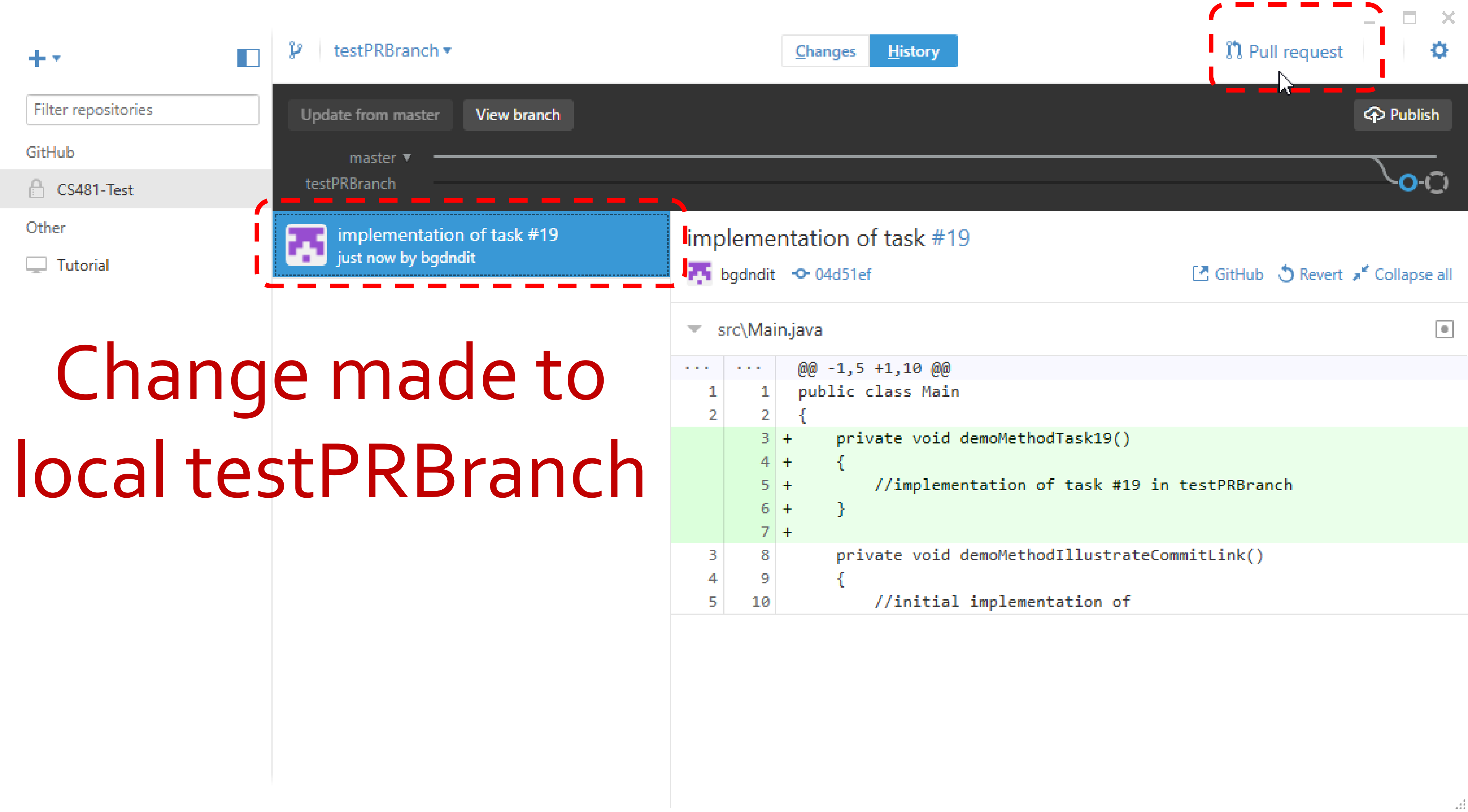
```
@@ -1,5 +1,10 @@
 public class Main
 {
+    private void demoMethodTask19()
+    {
+        //implementation of task #19 in testPRBranch
+    }
+
     private void demoMethodIllustrateCommitLink()
     {
         //initial implementation of
```

implementation of task #

Description

- #19 Sample Task to Illustrate a Pull Request
- #17 Task 4 Story 1
- #1 Create account
- #15 Customize ZenHub
- #11 Story Six

Prepare Pull Request



implementation of task #19
just now by bgdndit

implementation of task #19

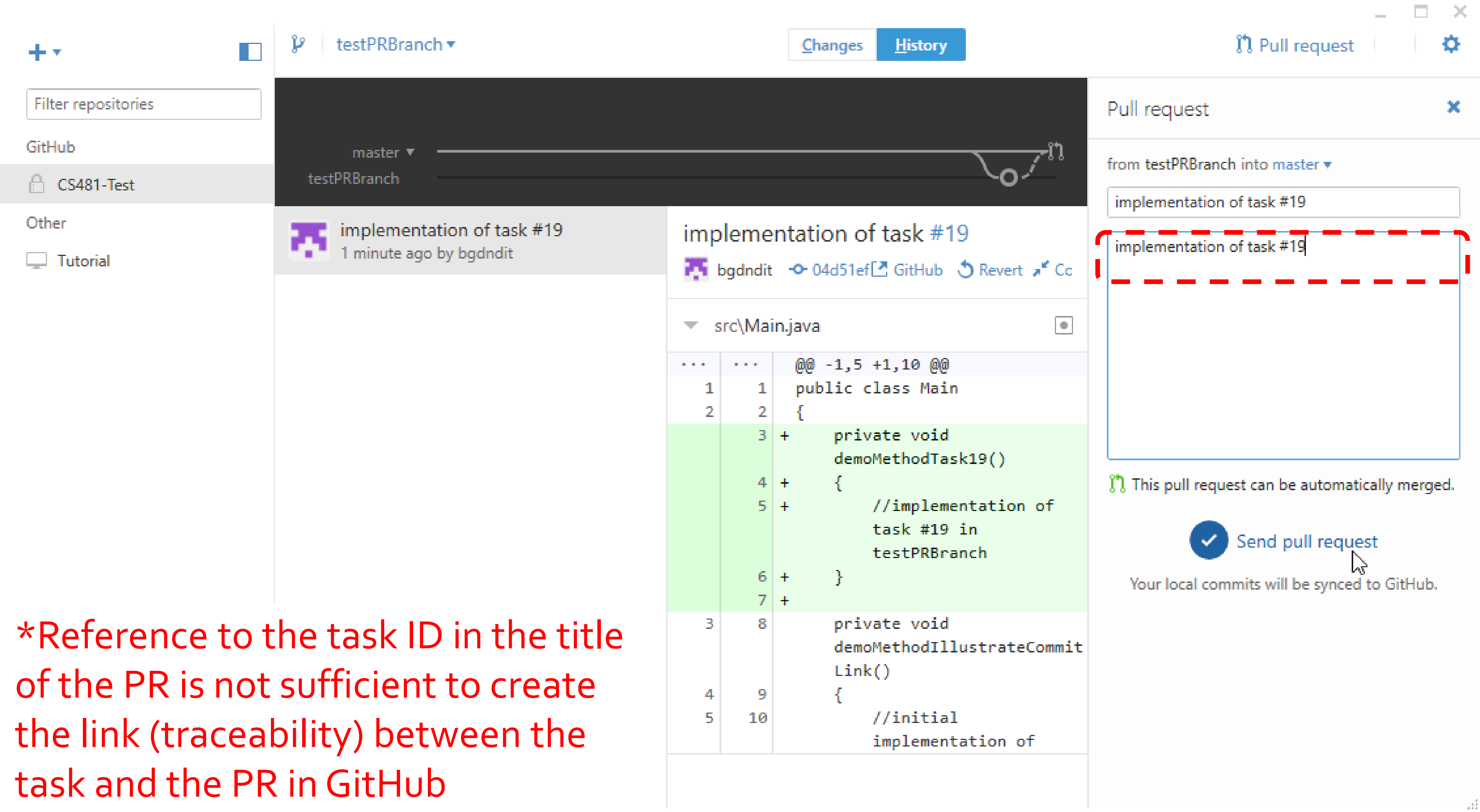
bgdndit 04d51ef

src\Main.java

```
@@ -1,5 +1,10 @@
 public class Main
 {
+    private void demoMethodTask19()
+    {
+        //implementation of task #19 in testPRBranch
+    }
+
     private void demoMethodIllustrateCommitLink()
     {
         //initial implementation of
```

Change made to local testPRBranch

Description of PR should include a reference to the task ID*



implementation of task #19
1 minute ago by bgdndit

implementation of task #19

src\Main.java

```
                @@ -1,5 +1,10 @@
  1     1       public class Main
  2     2       {
        3   +       private void
                    demoMethodTask19()
        4   +       {
        5   +           //implementation of
                        task #19 in
                        testPRBranch
        6   +       }
        7   +
  3     8           private void
                    demoMethodIllustrateCommit
                    Link()
  4     9           {
  5    10               //initial
                        implementation of
```

Pull request

from testPRBranch into master

implementation of task #19

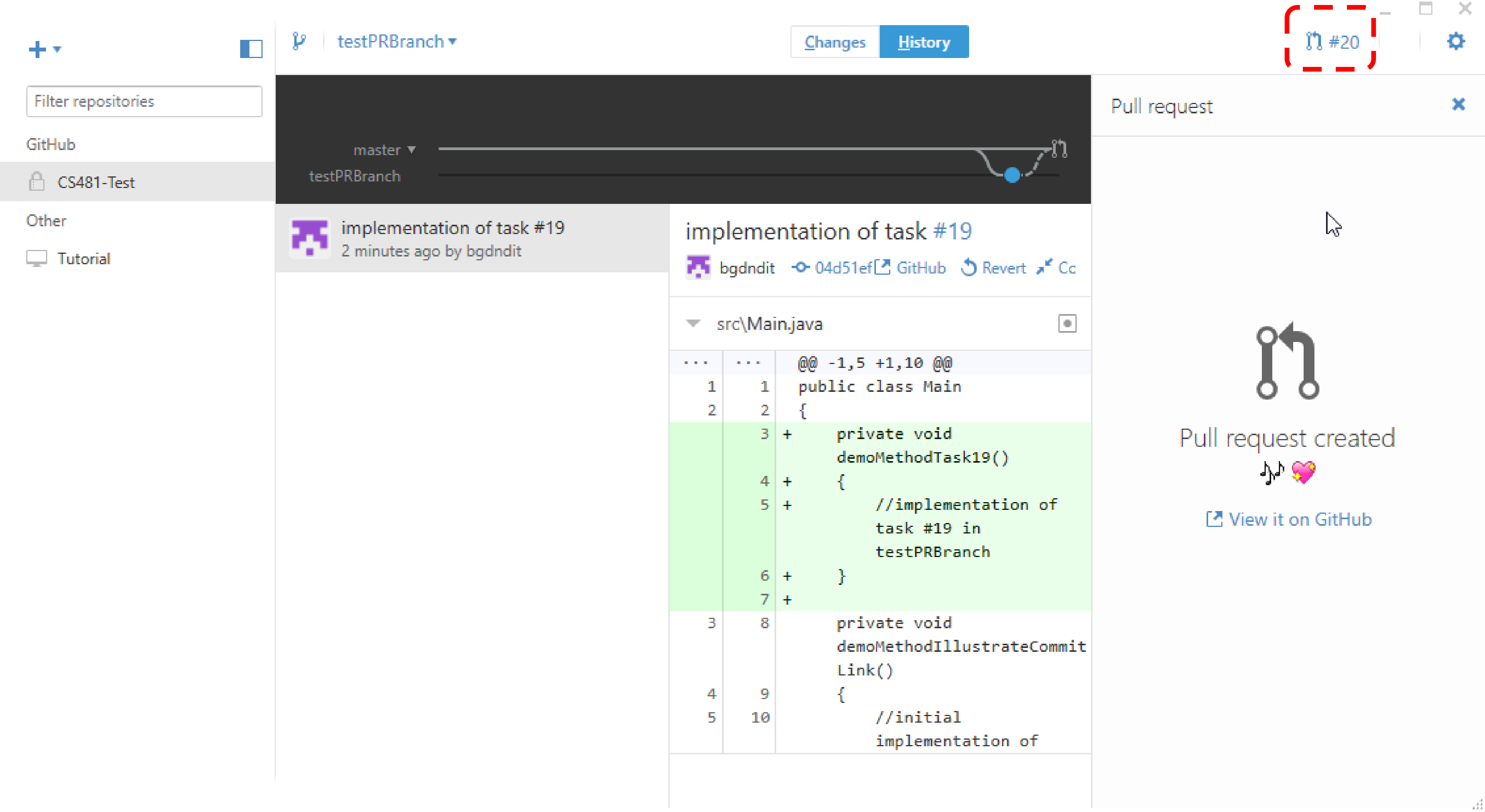
implementation of task #19

This pull request can be automatically merged.

Your local commits will be synced to GitHub.

*Reference to the task ID in the title of the PR is not sufficient to create the link (traceability) between the task and the PR in GitHub

A new Pull Request (#20) was created



implementation of task #19
2 minutes ago by bgdndit

implementation of task #19

src\Main.java

```
@@ -1,5 +1,10 @@
 public class Main
 {
+    private void demoMethodTask19()
+    {
+        //implementation of task #19 in testPRBranch
+    }
+
     private void demoMethodIllustrateCommitLink()
     {
         //initial implementation of
```

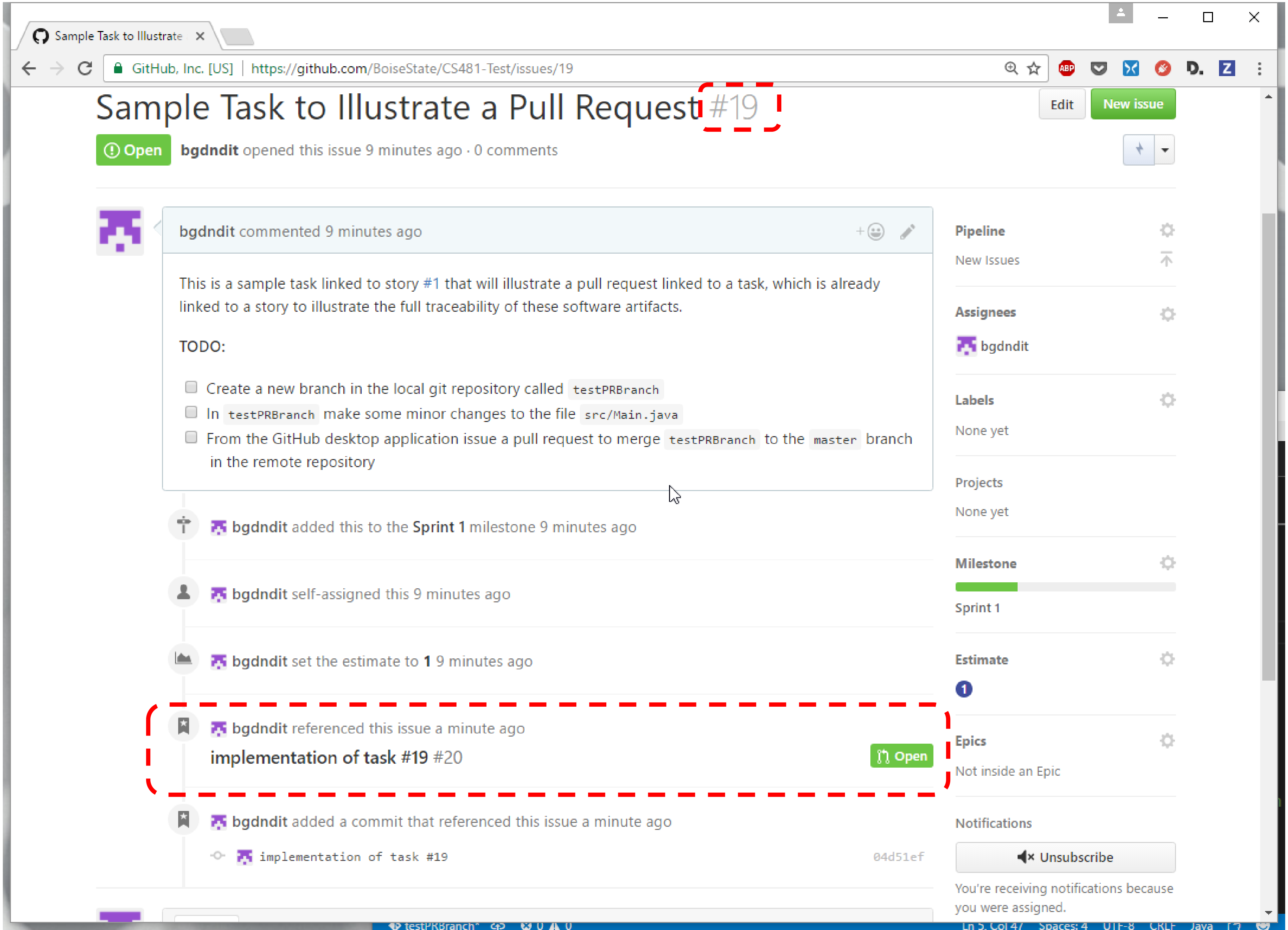
Pull request created
🎶💖

View it on GitHub

Why is Traceability Important?

- Establishing a traceability (link) between task #19 and PR #20
  - allows developers who view task #19 to navigate / investigate PR #20 (containing the commits and the changed code)
  - (vice-versa) allows developers to navigate from PR #20 back to task #19, in order to understand the context / requirement for the change
    - task #19 should further link to the a user story, for additional context
- Note: Traceability will be heavily emphasized in the group project

Task #19 is linked to the PR #20



Sample Task to Illustrate
GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/issues/19
Edit
New issue

Sample Task to Illustrate a Pull Request #19

Open **bgdndit** opened this issue 9 minutes ago · 0 comments

bgdndit commented 9 minutes ago

This is a sample task linked to story #1 that will illustrate a pull request linked to a task, which is already linked to a story to illustrate the full traceability of these software artifacts.

TODO:

- □ Create a new branch in the local git repository called `testPRBranch`
- □ In `testPRBranch` make some minor changes to the file `src/Main.java`
- □ From the GitHub desktop application issue a pull request to merge `testPRBranch` to the `master` branch in the remote repository

bgdndit added this to the **Sprint 1** milestone 9 minutes ago

bgdndit self-assigned this 9 minutes ago

bgdndit set the estimate to **1** 9 minutes ago

bgdndit referenced this issue a minute ago

implementation of task #19 #20 Open

bgdndit added a commit that referenced this issue a minute ago

`implementation of task #19` 04d51ef

Pipeline
New Issues

Assignees
bgdndit

Labels
None yet

Projects
None yet

Milestone
Sprint 1

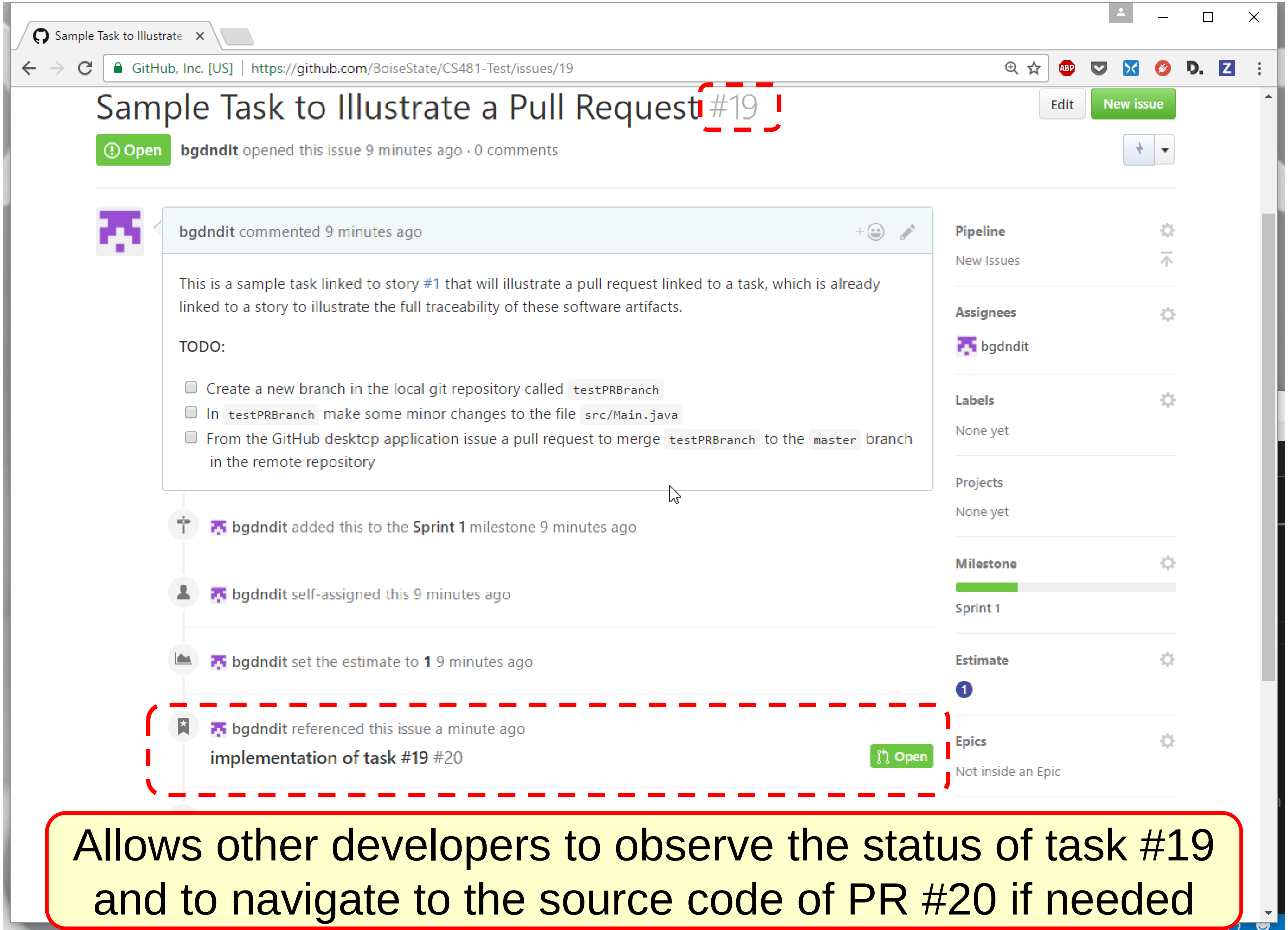
Estimate
1

Epics
Not inside an Epic

Notifications

You're receiving notifications because you were assigned.

Task #19 is linked to the PR #20



Sample Task to Illustrate
GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/issues/19
Edit
New issue

Sample Task to Illustrate a Pull Request #19

Open bgdndit opened this issue 9 minutes ago · 0 comments

bgdndit commented 9 minutes ago

This is a sample task linked to story #1 that will illustrate a pull request linked to a task, which is already linked to a story to illustrate the full traceability of these software artifacts.

TODO:

- □ Create a new branch in the local git repository called `testPRBranch`
- □ In `testPRBranch` make some minor changes to the file `src/Main.java`
- □ From the GitHub desktop application issue a pull request to merge `testPRBranch` to the `master` branch in the remote repository

bgdndit added this to the **Sprint 1** milestone 9 minutes ago

bgdndit self-assigned this 9 minutes ago

bgdndit set the estimate to **1** 9 minutes ago

bgdndit referenced this issue a minute ago

implementation of task #19 #20 Open

Pipeline: New Issues

Assignees: bgdndit

Labels: None yet

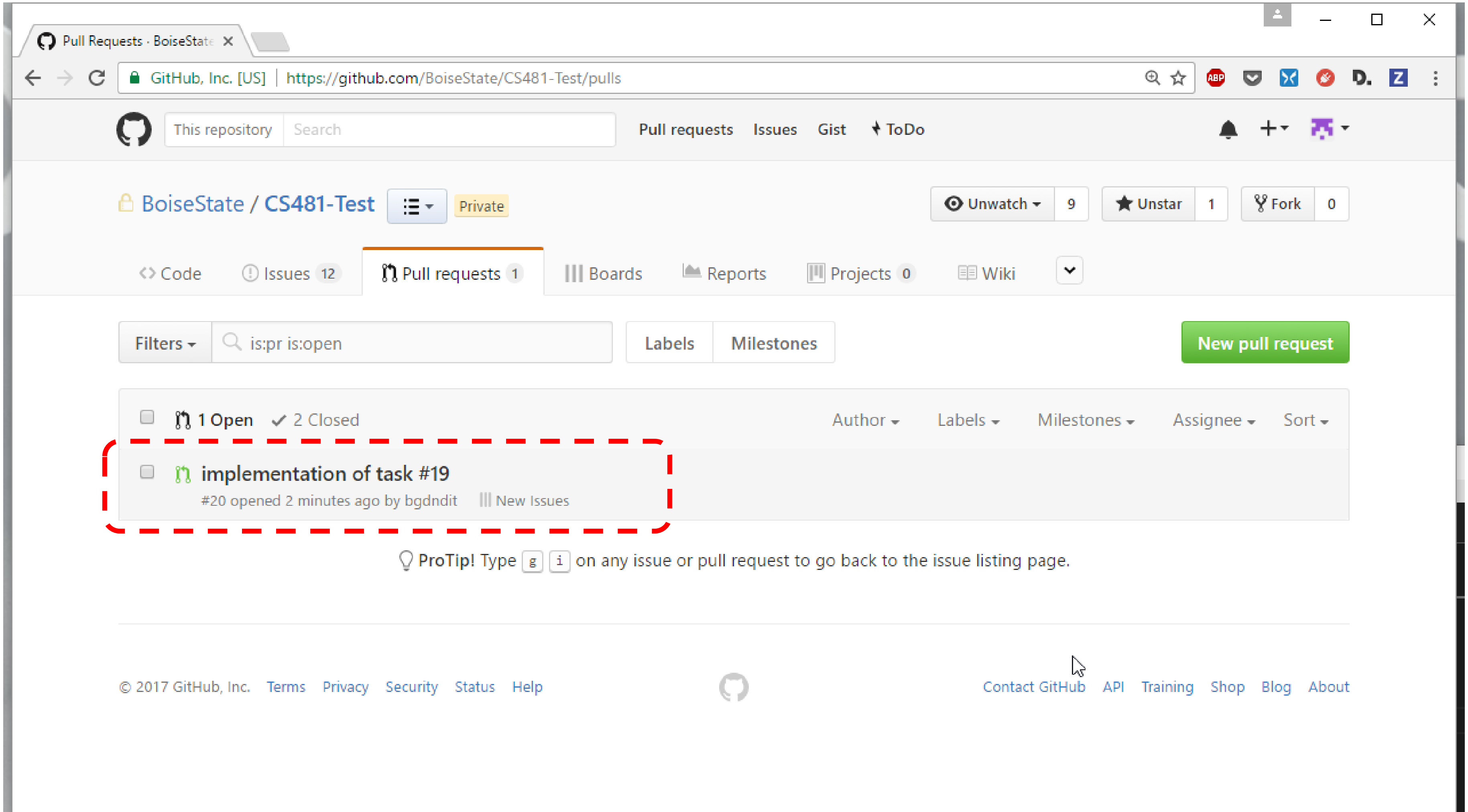
Projects: None yet

Milestone: Sprint 1

Estimate: 1

Epics: Not inside an Epic

Allows other developers to observe the status of task #19 and to navigate to the source code of PR #20 if needed



Pull Requests · BoiseState

GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pulls

This repository Search

Pull requests Issues Gist ToDo

BoiseState / CS481-Test Private

Unwatch 9 Unstar 1 Fork 0

Code Issues 12 Pull requests 1 Boards Reports Projects 0 Wiki

Filters is:pr is:open Labels Milestones New pull request

□ 1 Open ✓ 2 Closed Author Labels Milestones Assignee Sort

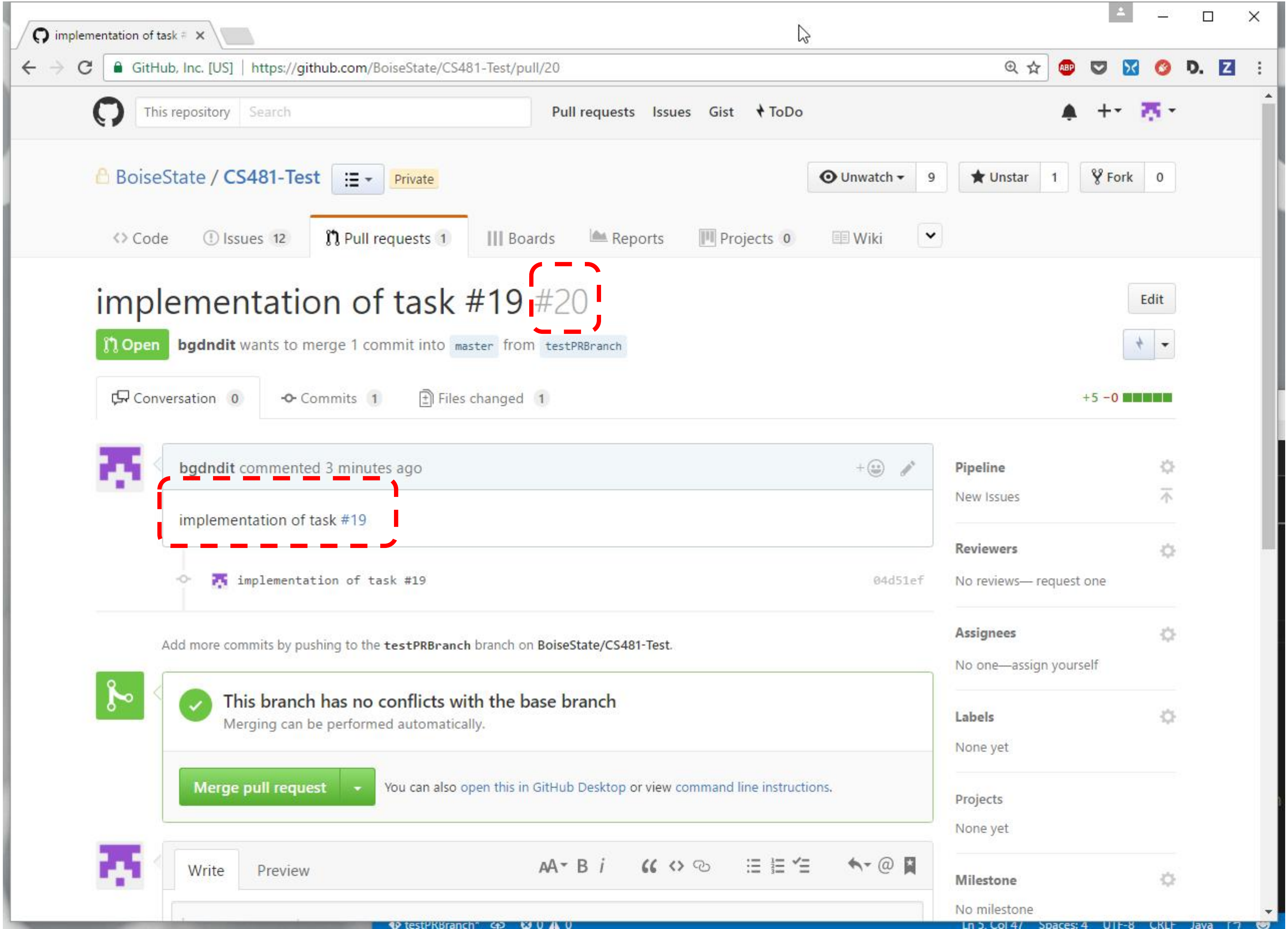
□ implementation of task #19
#20 opened 2 minutes ago by bgdndit New Issues

ProTip! Type g i on any issue or pull request to go back to the issue listing page.

© 2017 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub API Training Shop Blog About

Alternative way to access PRs (i.e., see all open PRs that other developers can contribute to…)

PR #20 referencing task #19



implementation of task #19 #20

Open bgdndit wants to merge 1 commit into master from testPRBranch

Conversation 0 Commits 1 Files changed 1 +5 −0

bgdndit commented 3 minutes ago

implementation of task #19

implementation of task #19 04d51ef

Add more commits by pushing to the testPRBranch branch on BoiseState/CS481-Test.

This branch has no conflicts with the base branch

Merging can be performed automatically.

Merge pull request You can also open this in GitHub Desktop or view command line instructions.

Pipeline: New Issues

Reviewers: No reviews— request one

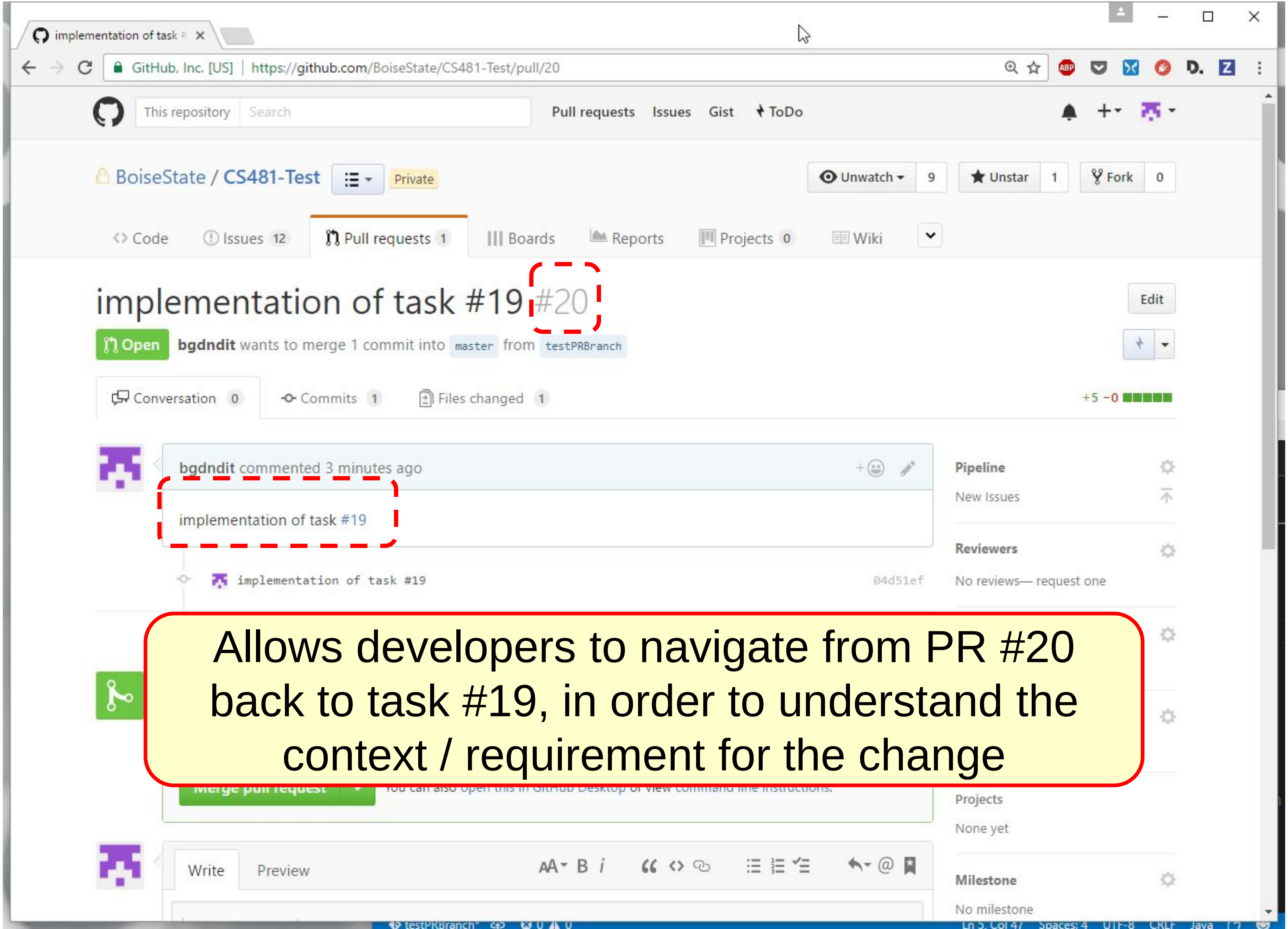
Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

Milestone: No milestone

PR #20 referencing task #19



implementation of task #19 #20

Open bgdndit wants to merge 1 commit into master from testPRBranch

Conversation 0 Commits 1 Files changed 1 +5 −0

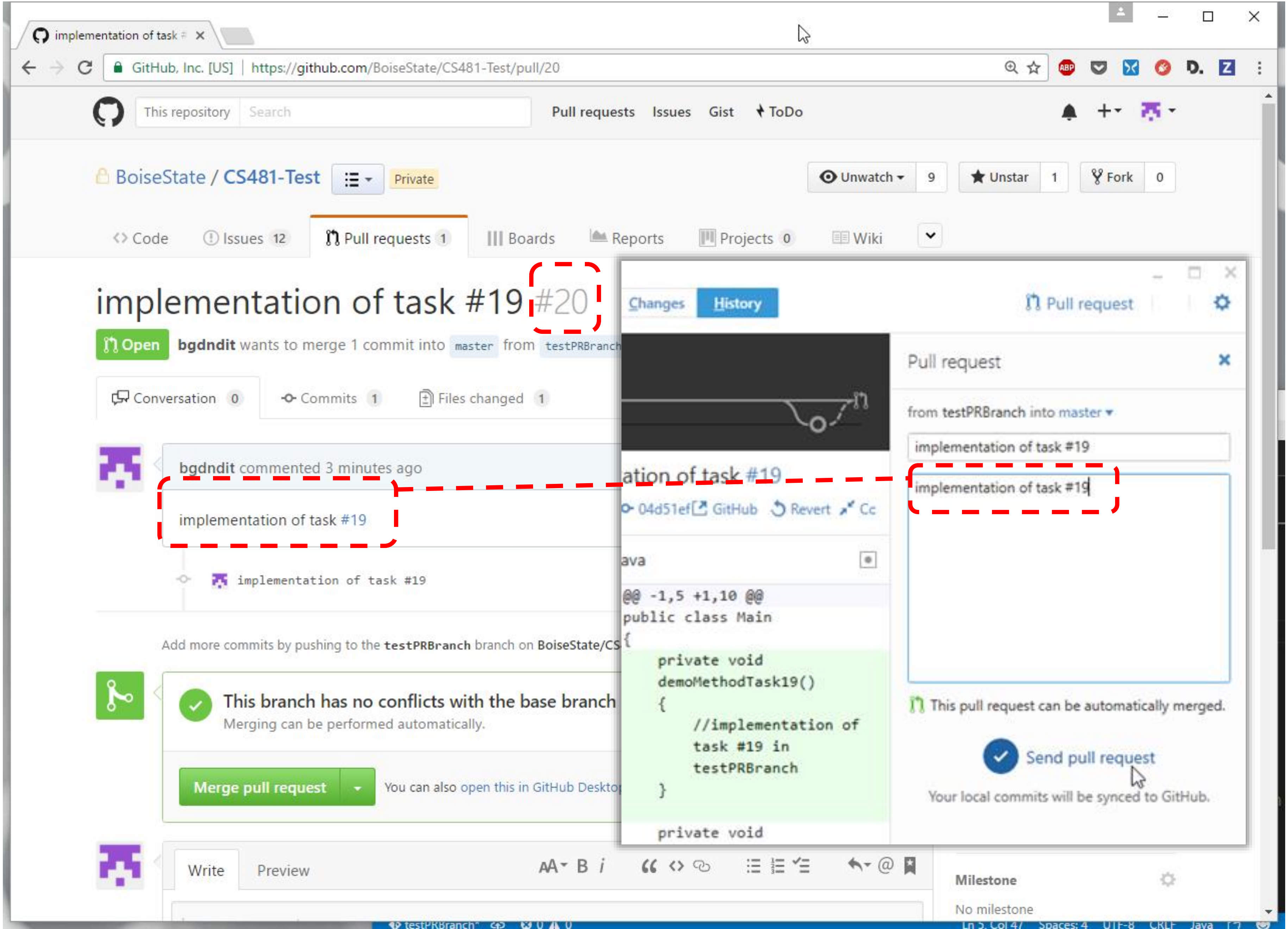
bgdndit commented 3 minutes ago

implementation of task #19

implementation of task #19 04d51ef

Allows developers to navigate from PR #20 back to task #19, in order to understand the context / requirement for the change

PR #20 referencing task #19



implementation of task #19 ×

GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pull/20

This repository | Search | Pull requests | Issues | Gist | ToDo

BoiseState / CS481-Test | Private

Unwatch 9 | Unstar 1 | Fork 0

Code | Issues 12 | Pull requests 1 | Boards | Reports | Projects 0 | Wiki

implementation of task #19 #20

Open **bgdndit** wants to merge 1 commit into `master` from `testPRBranch`

Conversation 0 | Commits 1 | Files changed 1

bgdndit commented 3 minutes ago

implementation of task #19

`implementation of task #19`

Add more commits by pushing to the **testPRBranch** branch on BoiseState/CS

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request | You can also open this in GitHub Deskto

Write | Preview

Milestone
No milestone

Changes | History | Pull request

ation of task #19
04d51ef GitHub | Revert | Cc

```
ava
@@ -1,5 +1,10 @@
public class Main
{
    private void
    demoMethodTask19()
    {
        //implementation of
        task #19 in
        testPRBranch
    }

    private void
```

Pull request ×

from testPRBranch into master

implementation of task #19

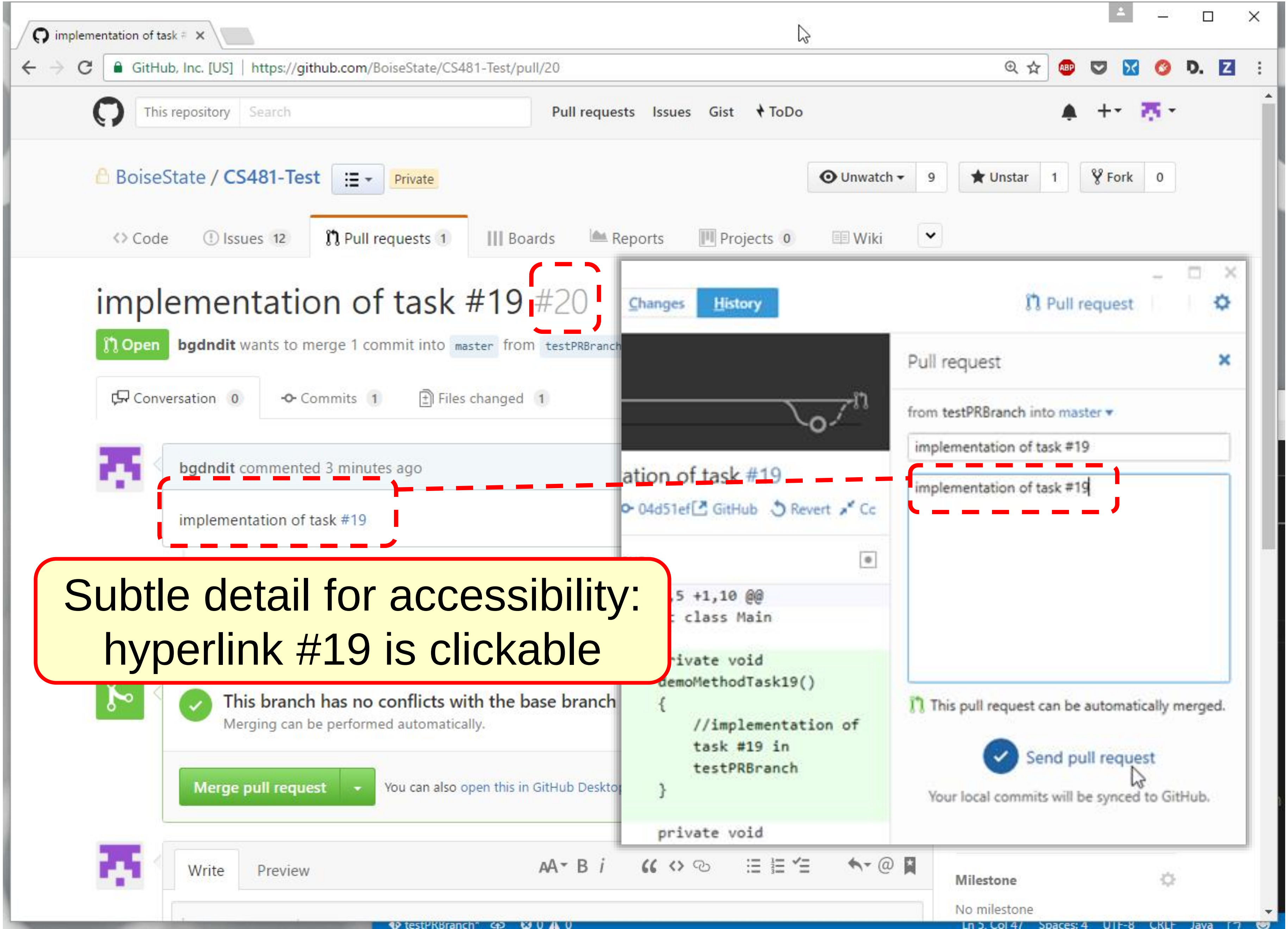
implementation of task #19

This pull request can be automatically merged.

Send pull request

Your local commits will be synced to GitHub.

PR #20 referencing task #19



implementation of task #19 #20

Open bgdndit wants to merge 1 commit into master from testPRBranch

bgdndit commented 3 minutes ago

implementation of task #19

This branch has no conflicts with the base branch

Merging can be performed automatically.

You can also open this in GitHub Desktop

Pull request

from testPRBranch into master

implementation of task #19

implementation of task #19

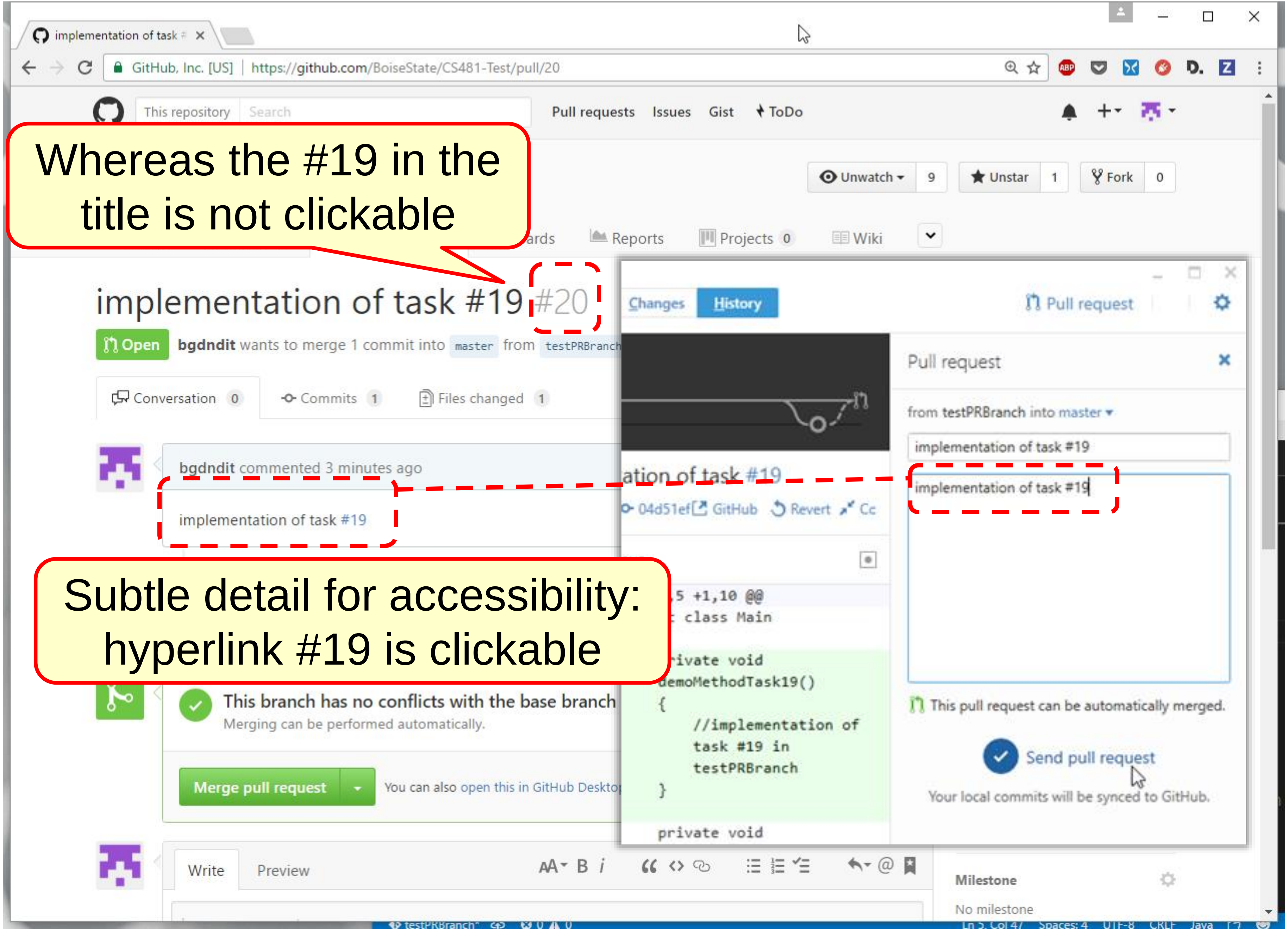
This pull request can be automatically merged.

Your local commits will be synced to GitHub.

```
5 +1,10 @@
 class Main

 ivate void
 demoMethodTask19()
 {
     //implementation of
     task #19 in
     testPRBranch
 }

 private void
```

Subtle detail for accessibility:
hyperlink #19 is clickable

PR #20 referencing task #19



Whereas the #19 in the title is not clickable

Subtle detail for accessibility: hyperlink #19 is clickable

implementation of task #19 #20

Open bgdndit wants to merge 1 commit into master from testPRBranch

bgdndit commented 3 minutes ago

implementation of task #19

This branch has no conflicts with the base branch

Merging can be performed automatically.

You can also open this in GitHub Desktop

Pull request

from testPRBranch into master

implementation of task #19

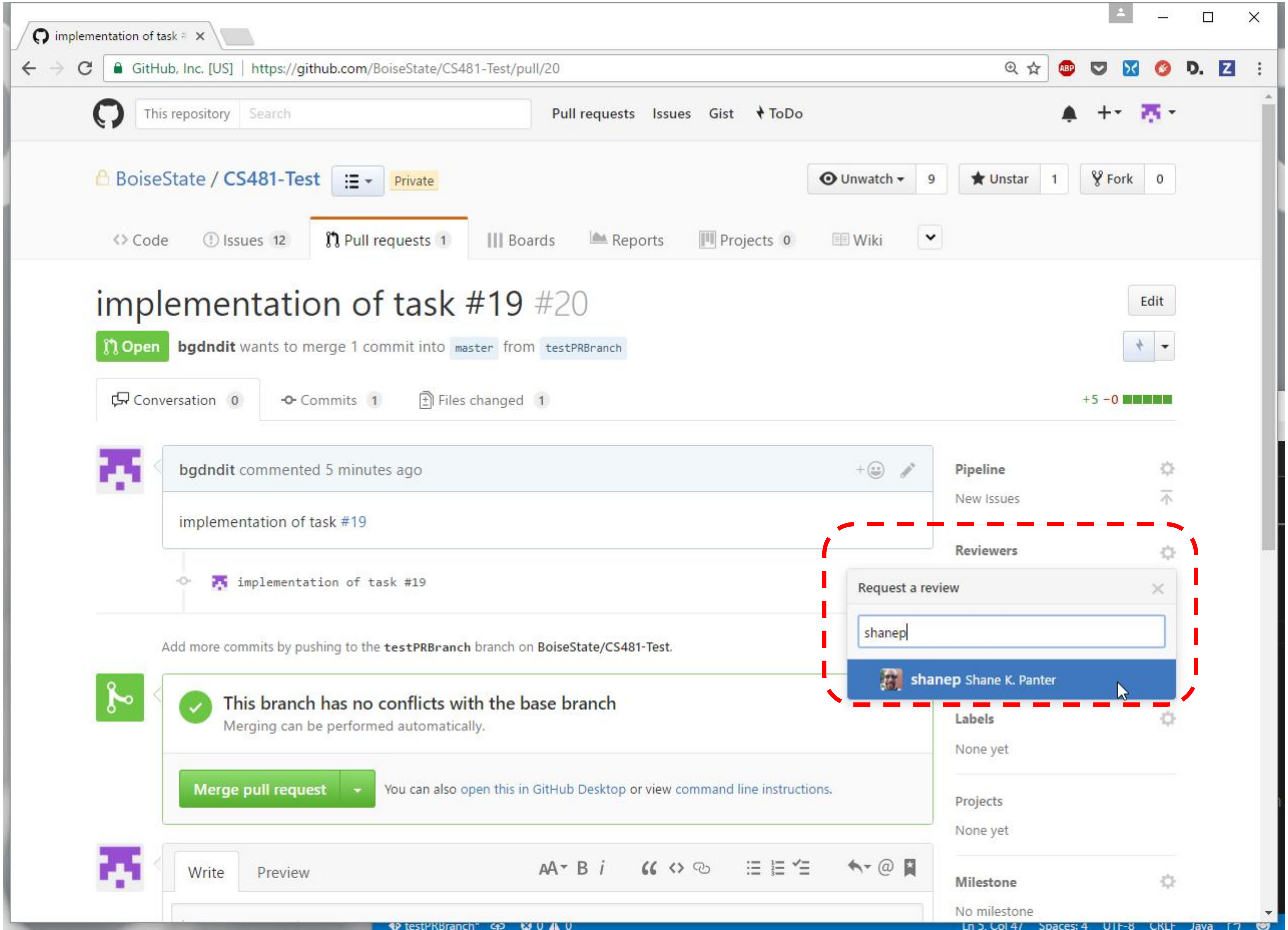
implementation of task #19

This pull request can be automatically merged.

Your local commits will be synced to GitHub.

```
@@ -1,5 +1,10 @@
class Main
private void
demoMethodTask19()
{
    //implementation of
    task #19 in
    testPRBranch
}
private void
```

PR allows you to Request Reviewer



implementation of task #19 #20

Open bgdndit wants to merge 1 commit into master from testPRBranch

bgdndit commented 5 minutes ago

implementation of task #19

implementation of task #19

Add more commits by pushing to the testPRBranch branch on BoiseState/CS481-Test.

This branch has no conflicts with the base branch

Merging can be performed automatically.

You can also open this in GitHub Desktop or view command line instructions.

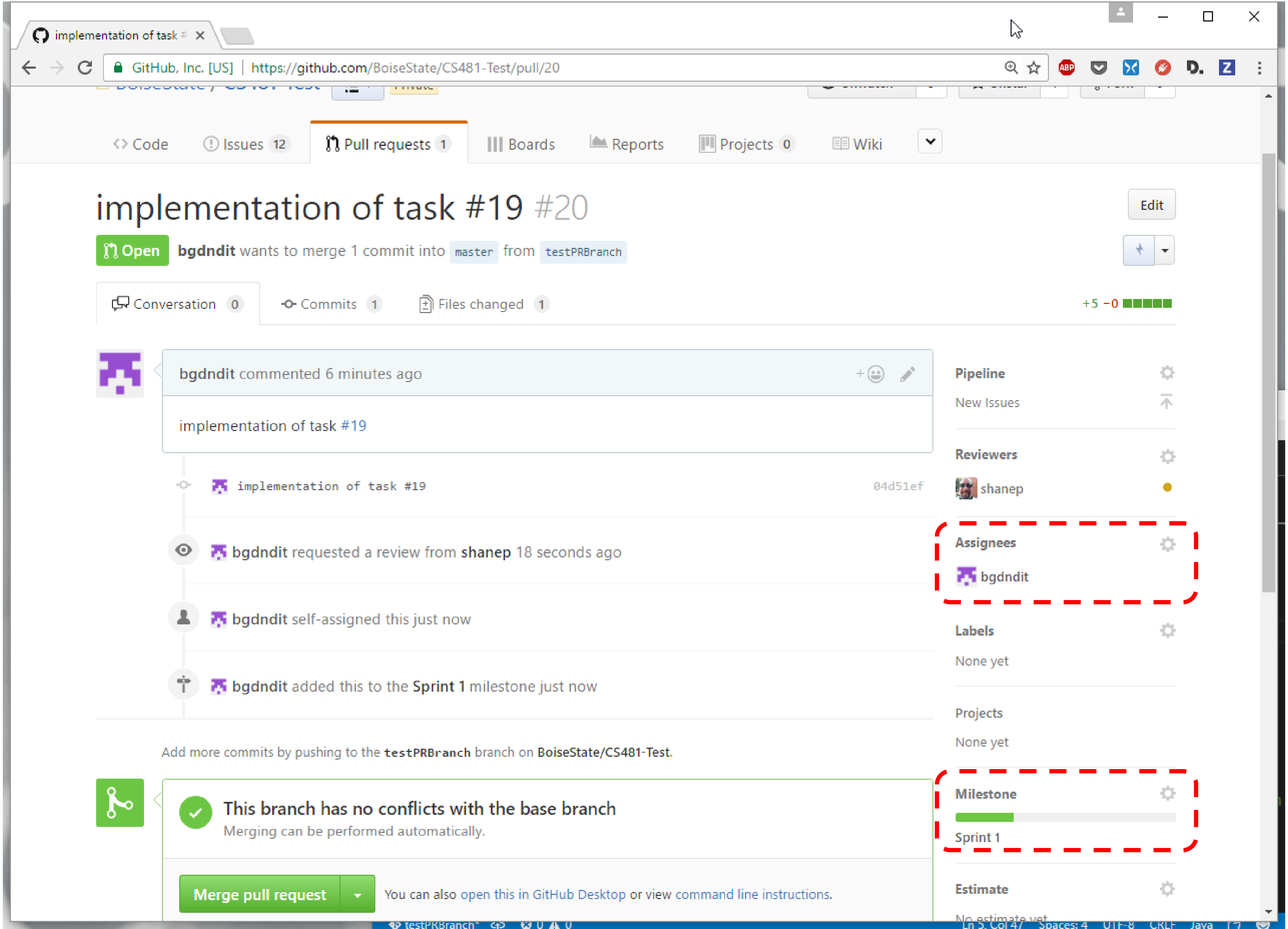
Reviewers

Request a review

shanep

shanep Shane K. Panter

Setup Assignees and Milestone



implementation of task #19 #20

Open bgdndit wants to merge 1 commit into master from testPRBranch

bgdndit commented 6 minutes ago

implementation of task #19

implementation of task #19 04d51ef

bgdndit requested a review from shanep 18 seconds ago

bgdndit self-assigned this just now

bgdndit added this to the Sprint 1 milestone just now

Add more commits by pushing to the testPRBranch branch on BoiseState/CS481-Test.

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request

You can also open this in GitHub Desktop or view command line instructions.

Pipeline
New Issues

Reviewers
shanep

Assignees
bgdndit

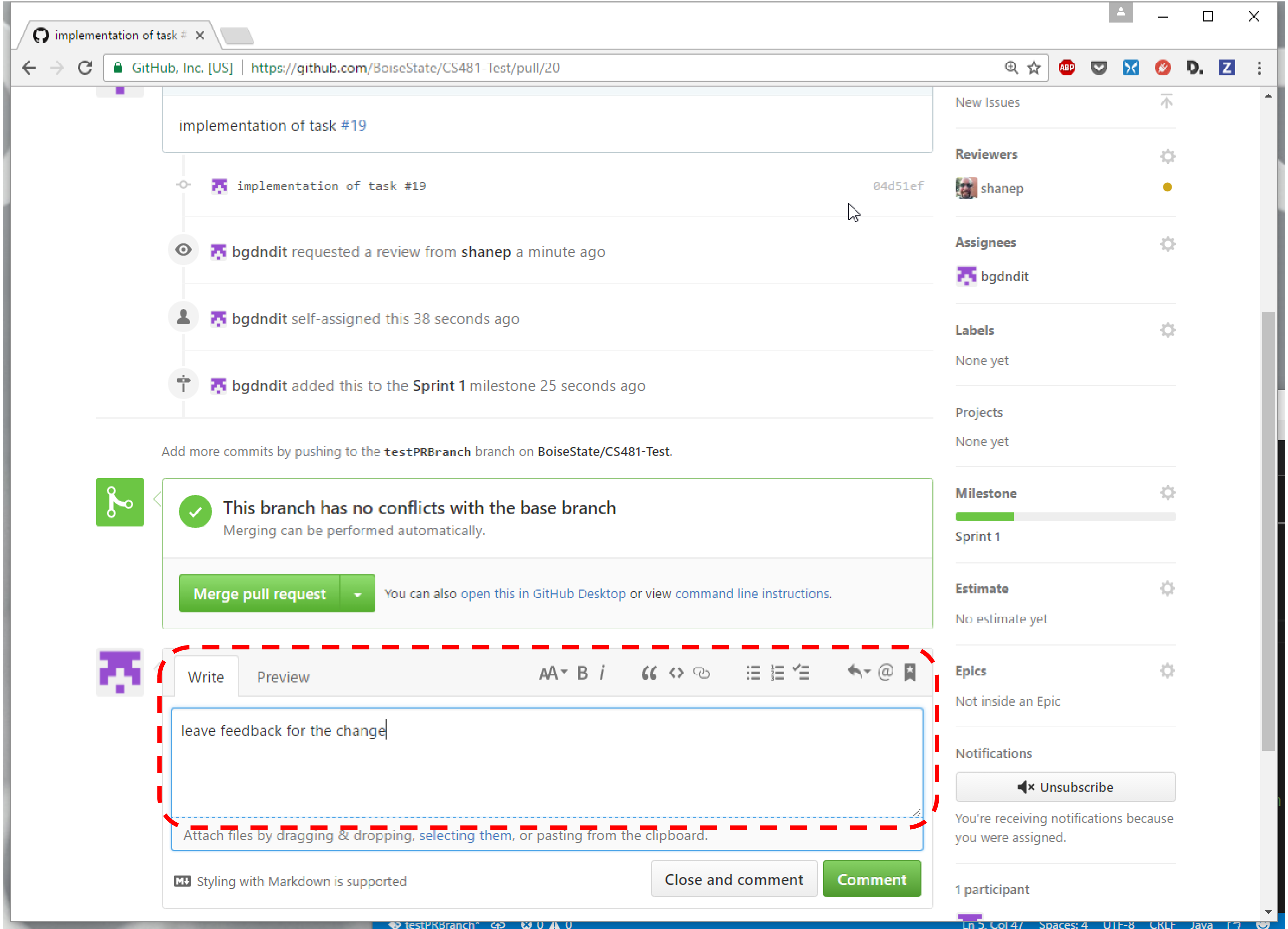
Labels
None yet

Projects
None yet

Milestone
Sprint 1

Estimate
No estimate yet

Leave feedback for the change



GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pull/20

implementation of task #19

```
implementation of task #19                                04d51ef
```

bgdndit requested a review from shanep a minute ago

bgdndit self-assigned this 38 seconds ago

bgdndit added this to the Sprint 1 milestone 25 seconds ago

Add more commits by pushing to the `testPRBranch` branch on BoiseState/CS481-Test.

This branch has no conflicts with the base branch

Merging can be performed automatically.

You can also open this in GitHub Desktop or view command line instructions.

leave feedback for the change

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

New Issues

Reviewers: shanep

Assignees: bgdndit

Labels: None yet

Projects: None yet

Milestone: Sprint 1

Estimate: No estimate yet

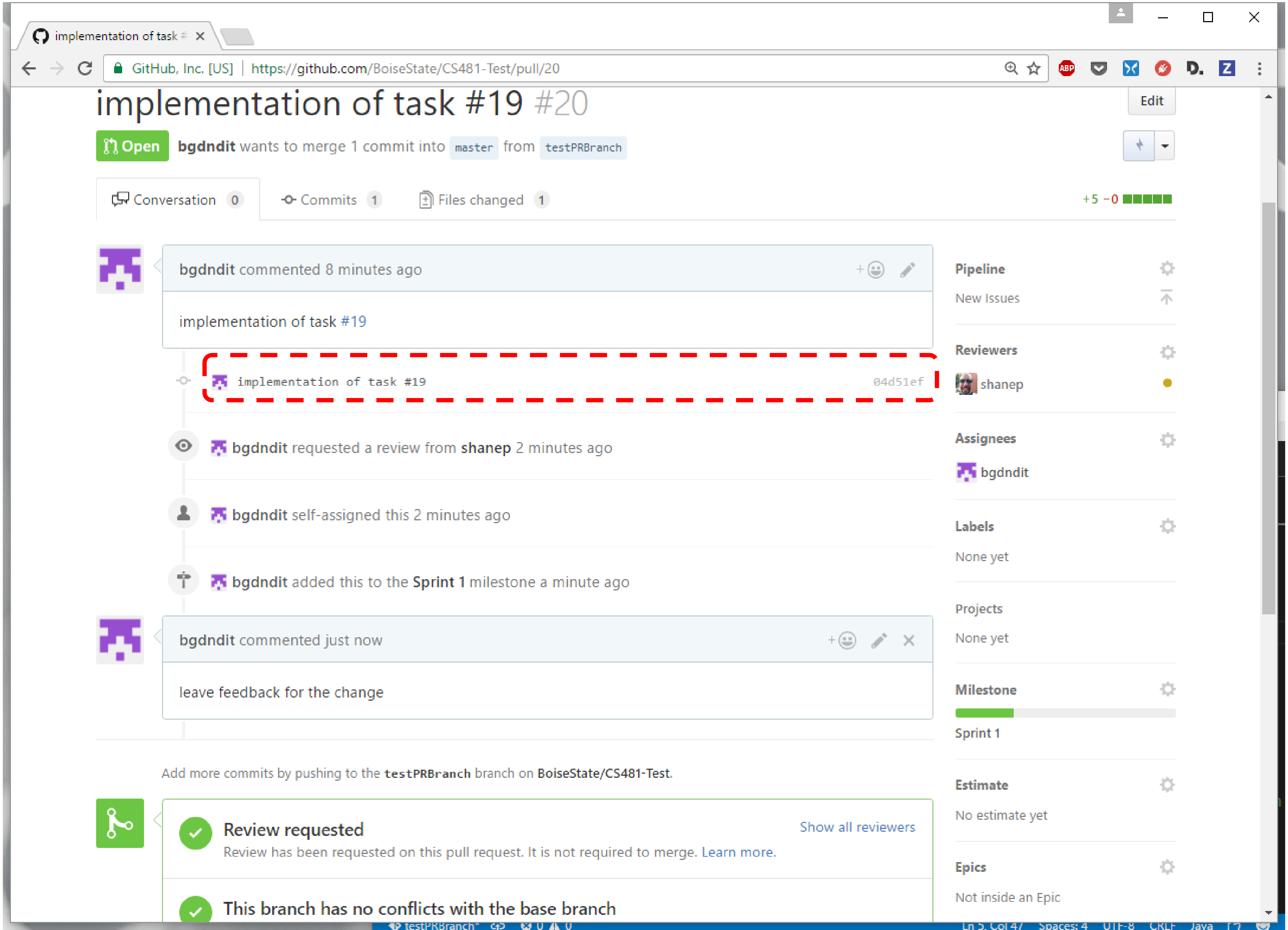
Epics: Not inside an Epic

Notifications

You're receiving notifications because you were assigned.

1 participant

Reviewer can inspect the code



implementation of task #
GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pull/20
Edit

implementation of task #19 #20

Open **bgdndit** wants to merge 1 commit into `master` from `testPRBranch`

bgdndit commented 8 minutes ago

implementation of task #19

`implementation of task #19` `04d51ef`

bgdndit requested a review from **shanep** 2 minutes ago

bgdndit self-assigned this 2 minutes ago

bgdndit added this to the **Sprint 1** milestone a minute ago

bgdndit commented just now

leave feedback for the change

Add more commits by pushing to the **testPRBranch** branch on **BoiseState/CS481-Test**.

Review requested

Review has been requested on this pull request. It is not required to merge. Learn more.

Show all reviewers

This branch has no conflicts with the base branch

Pipeline
New Issues

Reviewers
shanep

Assignees
bgdndit

Labels
None yet

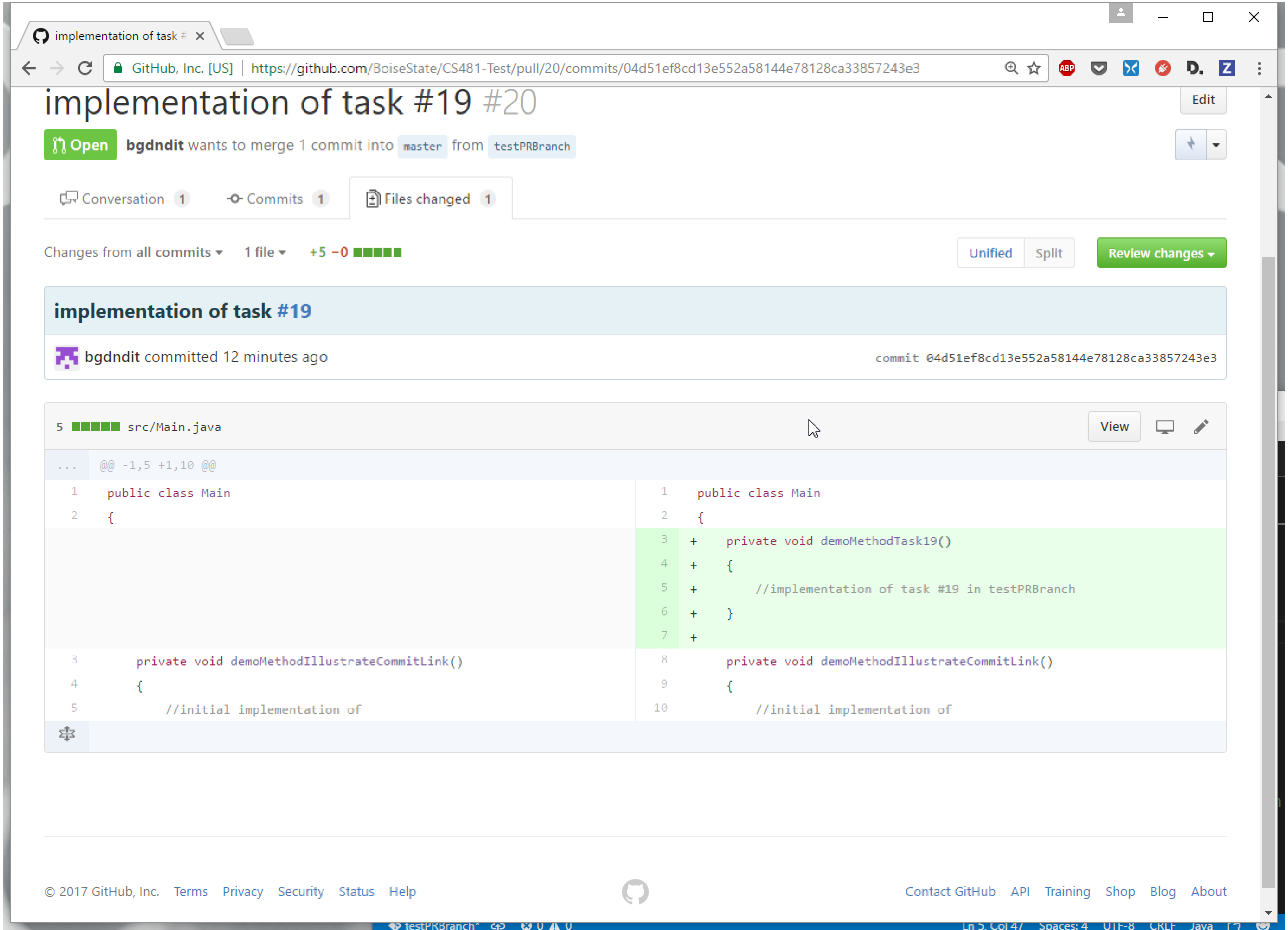
Projects
None yet

Milestone
Sprint 1

Estimate
No estimate yet

Epics
Not inside an Epic

Inspect code



implementation of task #
GitHub, Inc. [US]
https://github.com/BoiseState/CS481-Test/pull/20/commits/04d51ef8cd13e552a58144e78128ca33857243e3
Edit

implementation of task #19 #20

Open **bgdndit** wants to merge 1 commit into `master` from `testPRBranch`

+5 −0

implementation of task #19

bgdndit committed 12 minutes ago

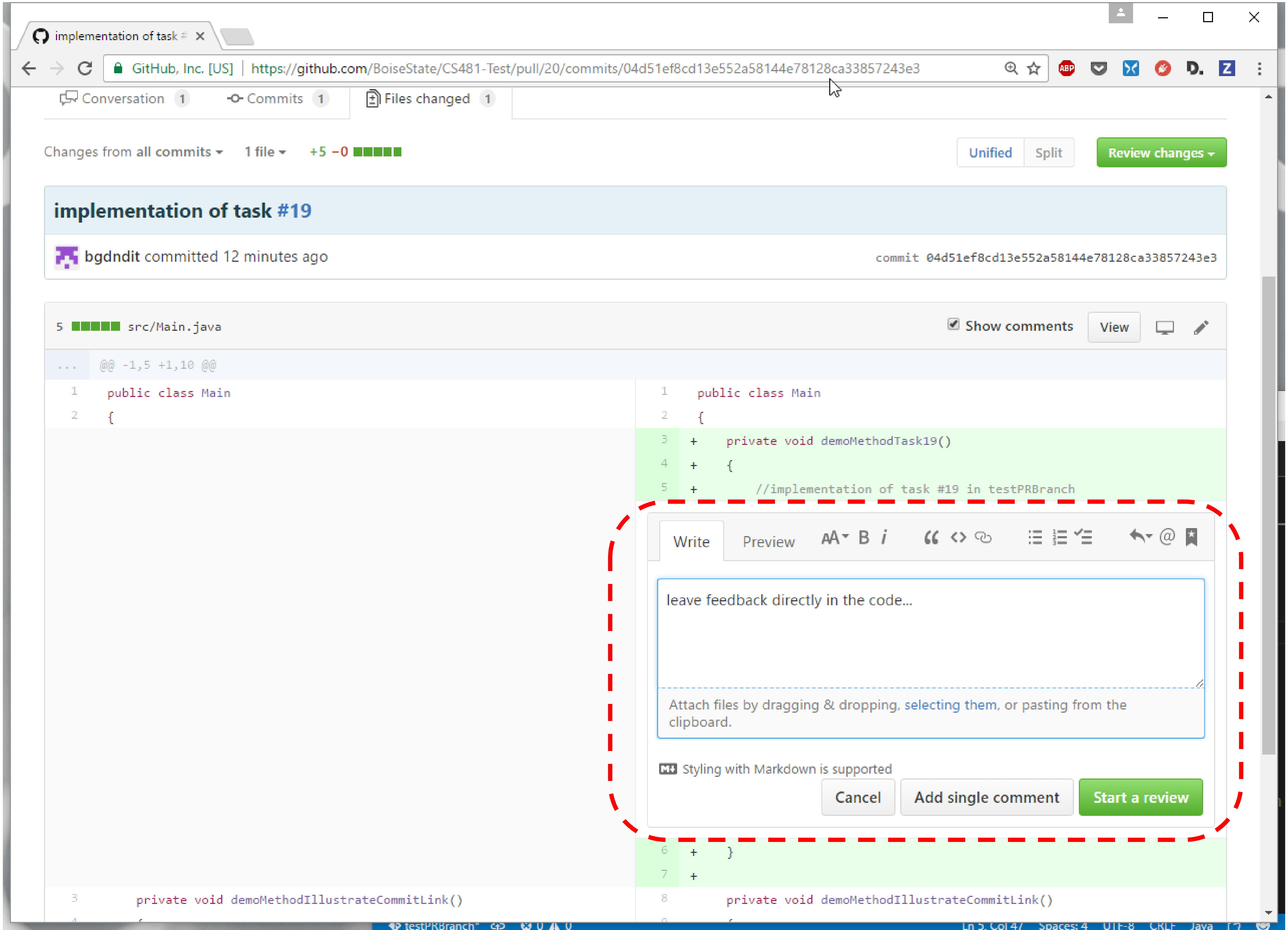
commit 04d51ef8cd13e552a58144e78128ca33857243e3

5 src/Main.java

```
@@ -1,5 +1,10 @@
public class Main                                  public class Main
{                                                  {
                                                   +    private void demoMethodTask19()
                                                   +    {
                                                   +        //implementation of task #19 in testPRBranch
                                                   +    }
                                                   +
    private void demoMethodIllustrateCommitLink()      private void demoMethodIllustrateCommitLink()
    {                                                  {
        //initial implementation of                        //initial implementation of
```

© 2017 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub API Training Shop Blog About

Leave feedback directly in the code…


implementation of task #
GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pull/20/commits/04d51ef8cd13e552a58144e78128ca33857243e3
Conversation 1
Commits 1
Files changed 1
Changes from all commits ▾
1 file ▾
+5 −0
Unified
Split
Review changes ▾

implementation of task #19

bgdndit committed 12 minutes ago

commit 04d51ef8cd13e552a58144e78128ca33857243e3

```
@@ -1,5 +1,10 @@
```

```
public class Main
{

    private void demoMethodIllustrateCommitLink()
```

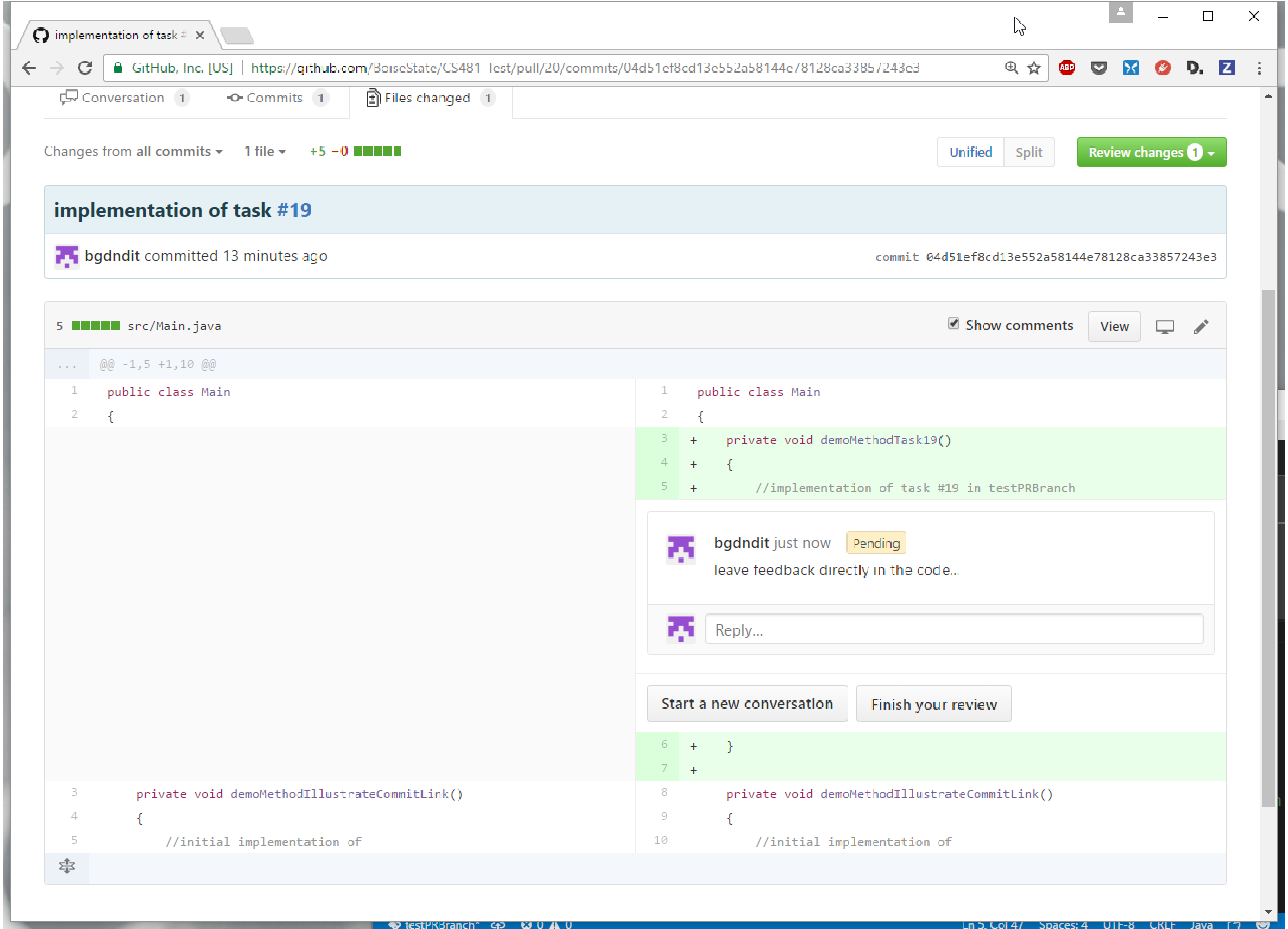
```
 public class Main
 {
+    private void demoMethodTask19()
+    {
+        //implementation of task #19 in testPRBranch
+    }
+
     private void demoMethodIllustrateCommitLink()
```

leave feedback directly in the code…

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

Starting a review...



implementation of task #19

bgdndit committed 13 minutes ago

commit 04d51ef8cd13e552a58144e78128ca33857243e3

5 src/Main.java

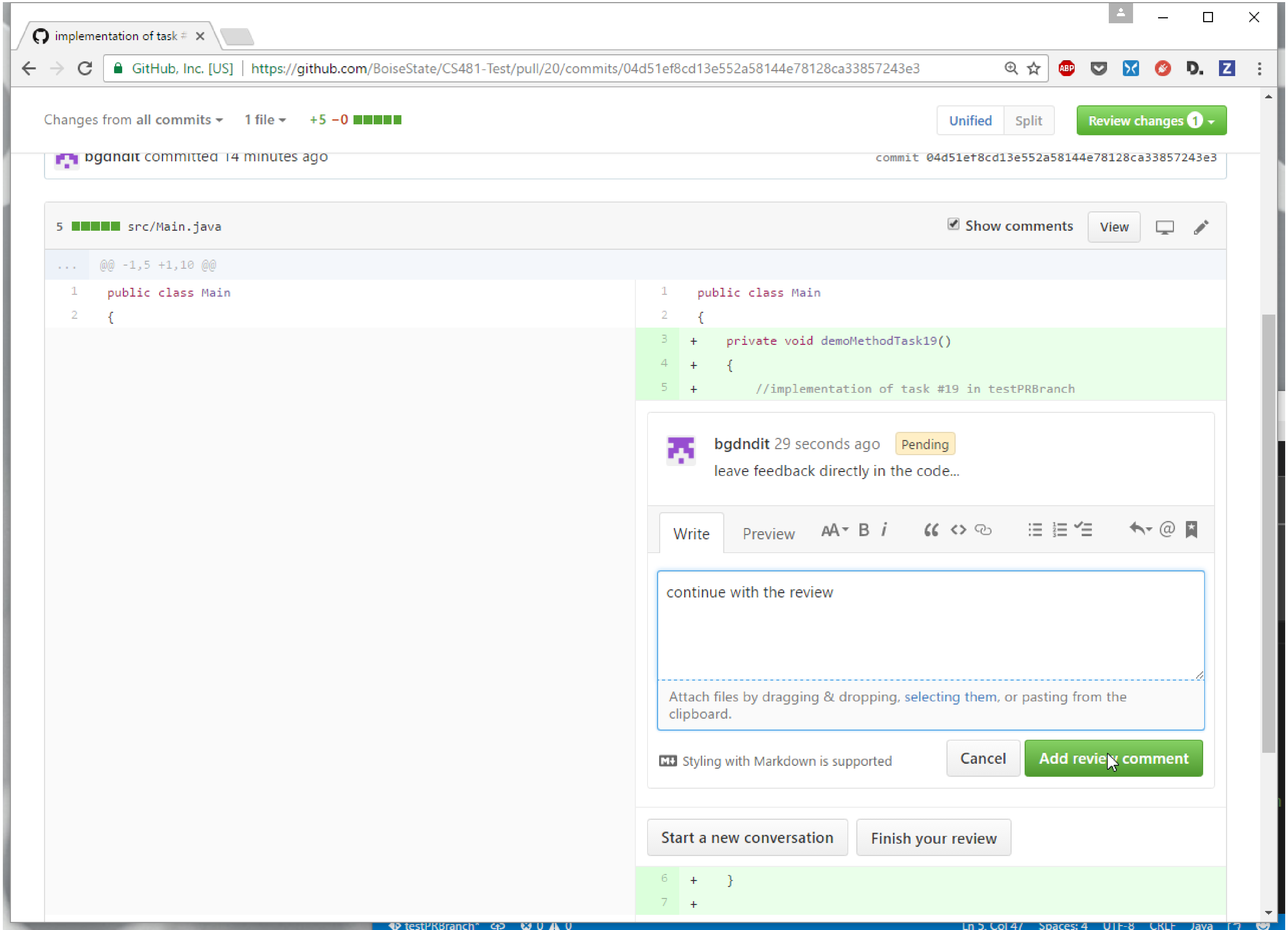
```
@@ -1,5 +1,10 @@
public class Main                              public class Main
{                                              {
                                             +     private void demoMethodTask19()
                                             +     {
                                             +         //implementation of task #19 in testPRBranch
```

bgdndit just now Pending

leave feedback directly in the code...

```
                                             +     }
                                             +
    private void demoMethodIllustrateCommitLink()      private void demoMethodIllustrateCommitLink()
    {                                                  {
        //initial implementation of                        //initial implementation of
```

Continue with the review



implementation of task #

GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pull/20/commits/04d51ef8cd13e552a58144e78128ca33857243e3

Changes from all commits ▾ 1 file ▾ +5 −0

Unified Split Review changes 1

bgdndit committed 14 minutes ago commit 04d51ef8cd13e552a58144e78128ca33857243e3

5 src/Main.java ☑ Show comments View

```
@@ -1,5 +1,10 @@
 public class Main              public class Main
 {                              {
                              +     private void demoMethodTask19()
                              +     {
                              +         //implementation of task #19 in testPRBranch
```

bgdndit 29 seconds ago Pending

leave feedback directly in the code...

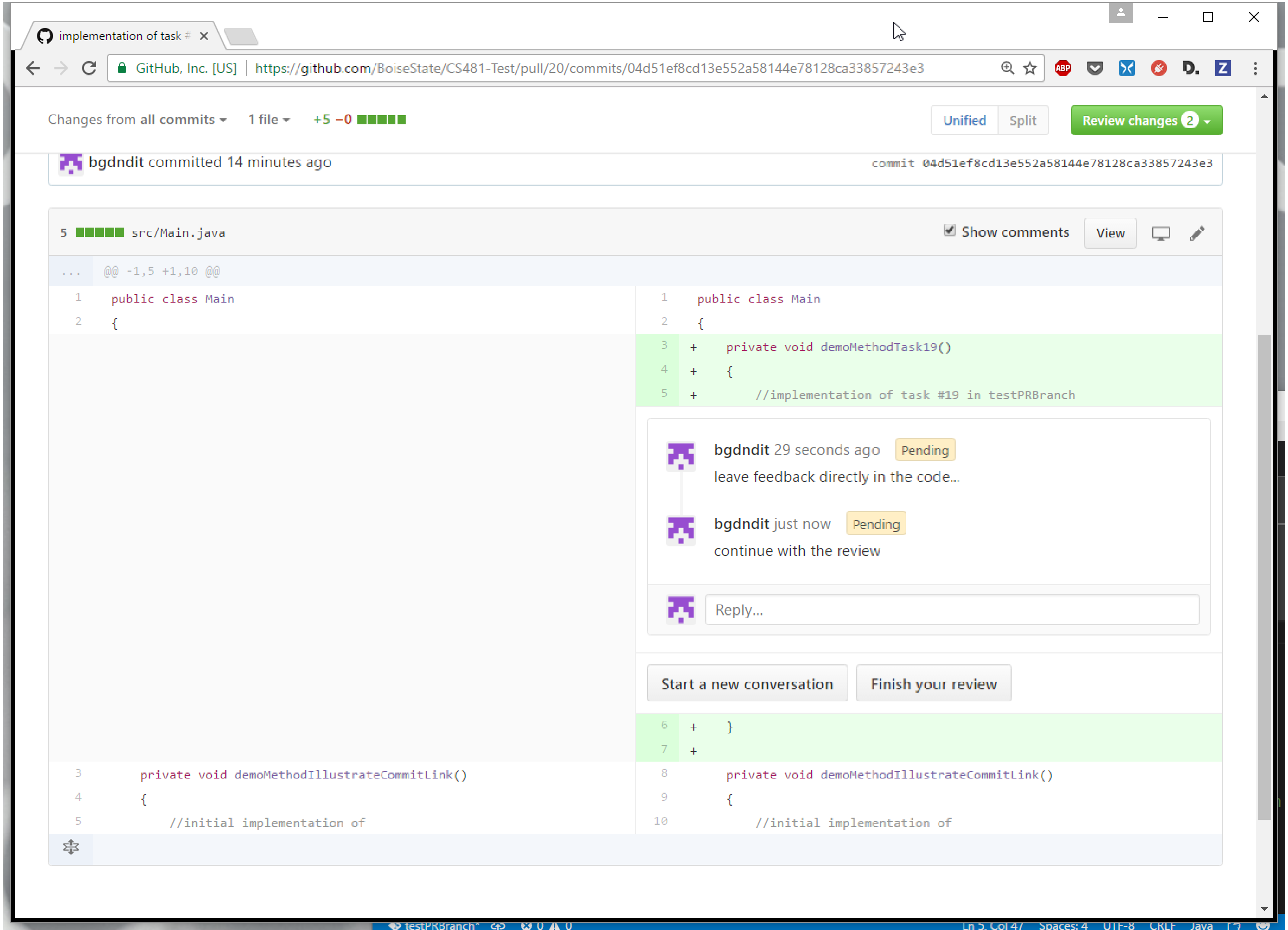
continue with the review

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

```
+     }
+
```

Finish the review



GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pull/20/commits/04d51ef8cd13e552a58144e78128ca33857243e3
Changes from all commits ▾ 1 file ▾ +5 −0

bgdndit committed 14 minutes ago commit 04d51ef8cd13e552a58144e78128ca33857243e3

5 src/Main.java

```
@@ -1,5 +1,10 @@
 public class Main
 {
+    private void demoMethodTask19()
+    {
+        //implementation of task #19 in testPRBranch
```

bgdndit 29 seconds ago Pending

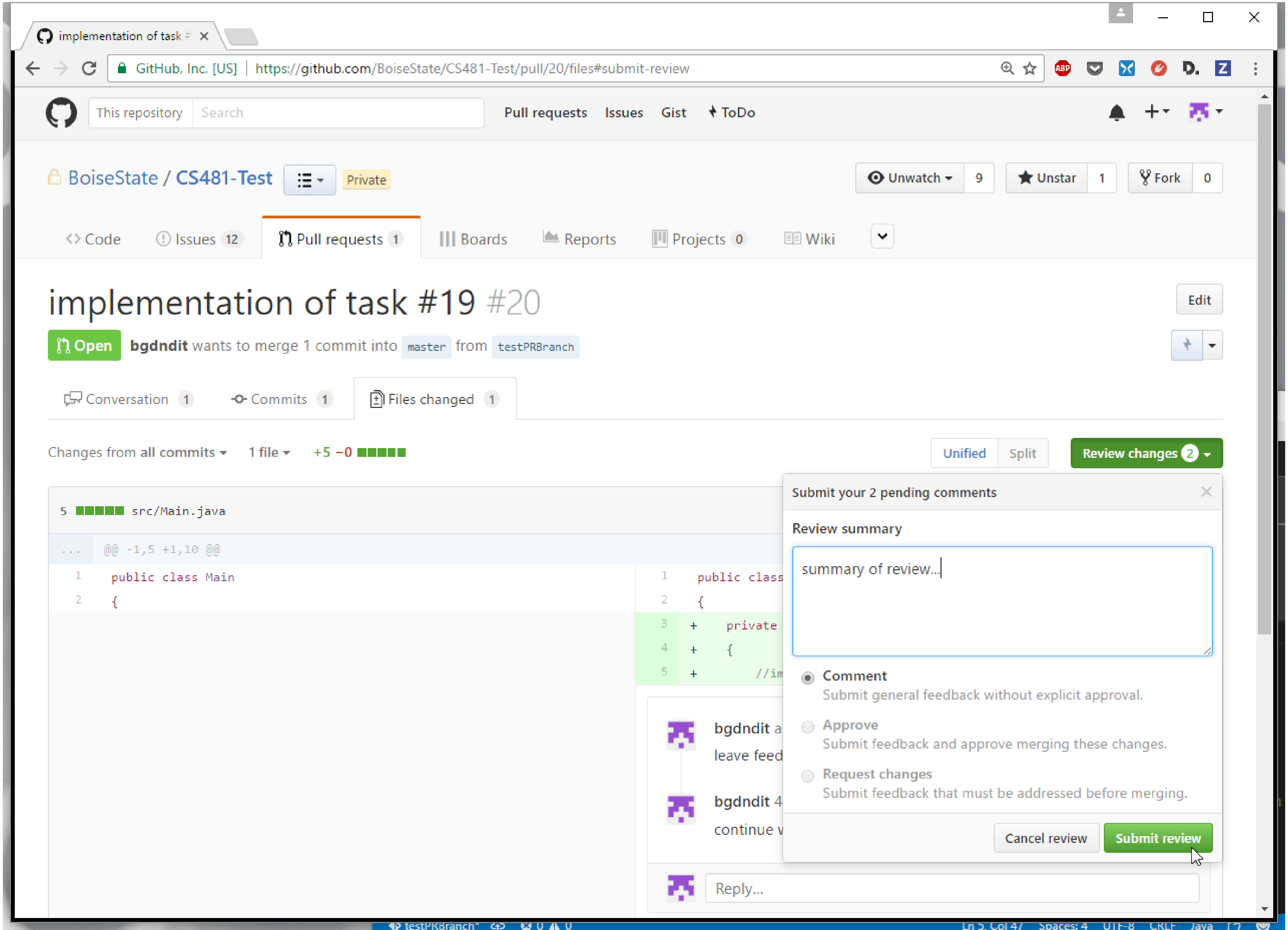
leave feedback directly in the code...

bgdndit just now Pending

continue with the review

```
+    }
+
     private void demoMethodIllustrateCommitLink()
     {
         //initial implementation of
```

Finish the review and provide summary



implementation of task #19 #20

Open bgdndit wants to merge 1 commit into `master` from `testPRBranch`

5 src/Main.java

```
@@ -1,5 +1,10 @@
public class Main          public class
{                          {
                         +     private
                         +     {
                         +         //im
```

bgdndit a
leave feed

bgdndit 4
continue v

Submit your 2 pending comments

Review summary

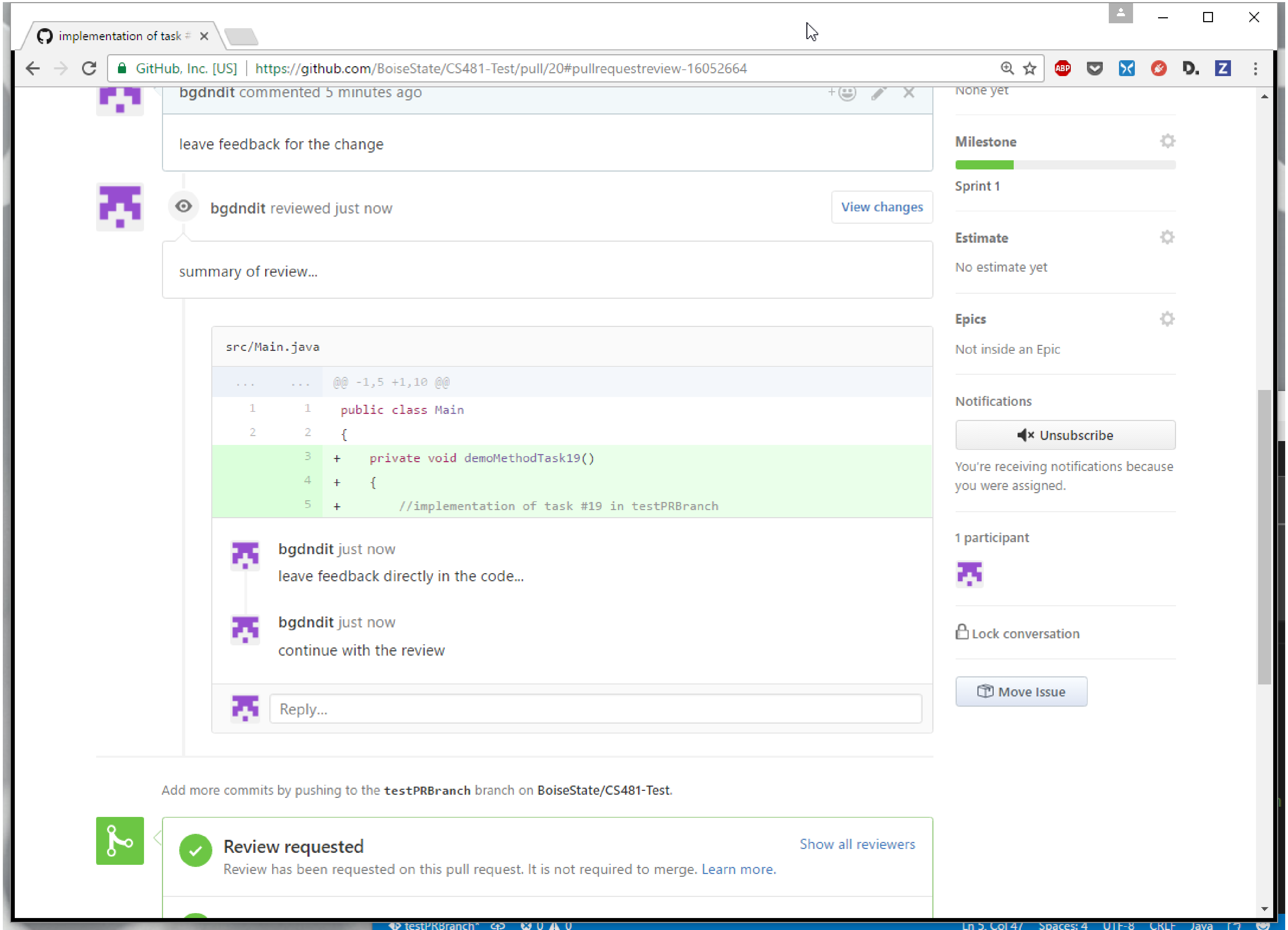
summary of review...

● Comment
Submit general feedback without explicit approval.

○ Approve
Submit feedback and approve merging these changes.

○ Request changes
Submit feedback that must be addressed before merging.

The review session inside the PR #20



implementation of task #
GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pull/20#pullrequestreview-16052664

bgdndit commented 5 minutes ago

leave feedback for the change

bgdndit reviewed just now

summary of review...

src/Main.java

```
...     ...   @@ -1,5 +1,10 @@
1       1      public class Main
2       2      {
        3     +    private void demoMethodTask19()
        4     +    {
        5     +        //implementation of task #19 in testPRBranch
```

bgdndit just now

leave feedback directly in the code...

bgdndit just now

continue with the review

Add more commits by pushing to the **testPRBranch** branch on **BoiseState/CS481-Test**.

Review requested

Review has been requested on this pull request. It is not required to merge. Learn more.

None yet

Milestone

Sprint 1

Estimate

No estimate yet

Epics

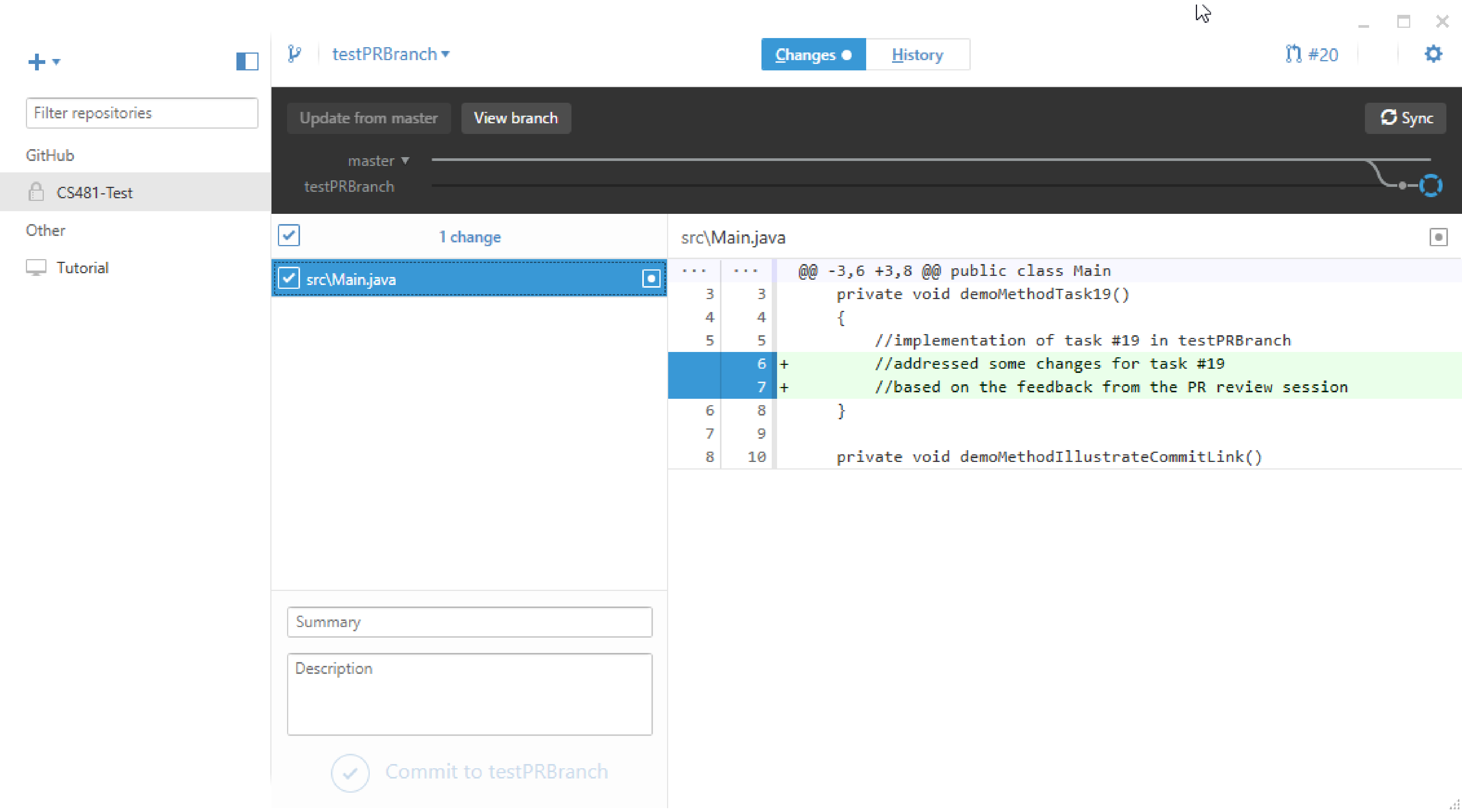
Not inside an Epic

Notifications

You're receiving notifications because you were assigned.

1 participant

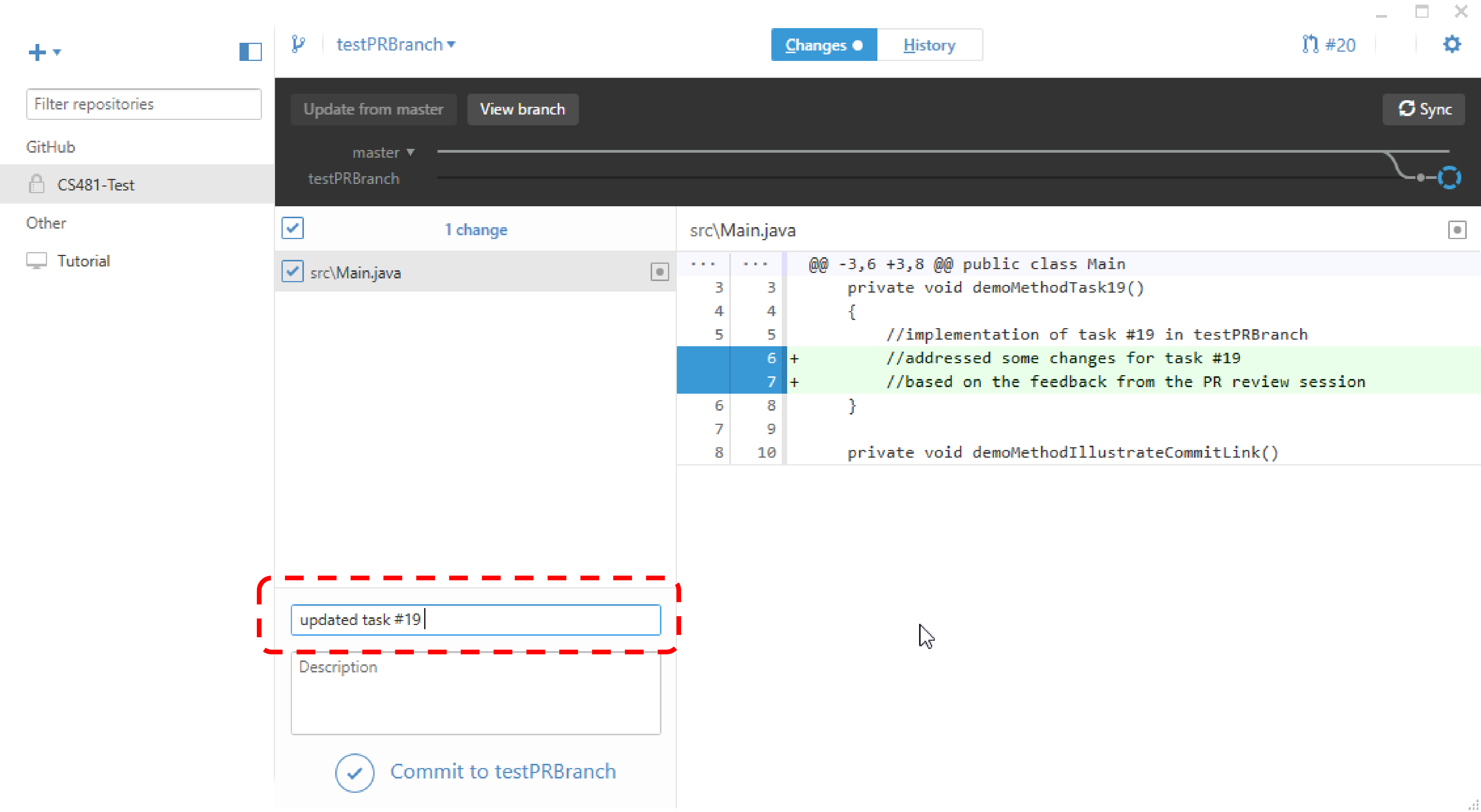
Address the changes for task #19 based on the feedback from the PR review session



src\Main.java

```
@@ -3,6 +3,8 @@ public class Main
    private void demoMethodTask19()
    {
        //implementation of task #19 in testPRBranch
+       //addressed some changes for task #19
+       //based on the feedback from the PR review session
    }

    private void demoMethodIllustrateCommitLink()
```

Add a new commit to the testPRBranch



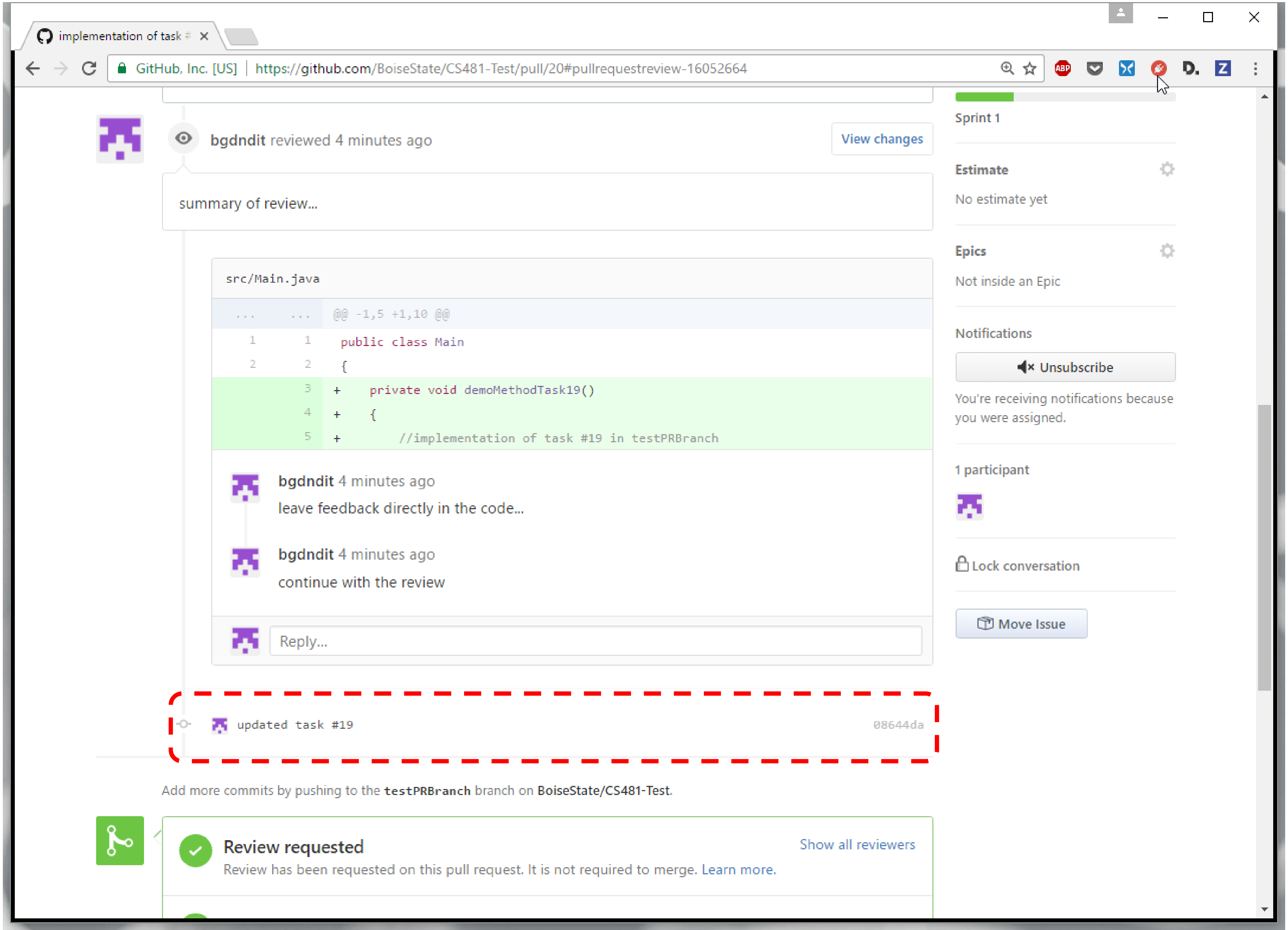
src\Main.java

```
...  ...  @@ -3,6 +3,8 @@ public class Main
3    3        private void demoMethodTask19()
4    4        {
5    5            //implementation of task #19 in testPRBranch
     6  +         //addressed some changes for task #19
     7  +         //based on the feedback from the PR review session
6    8        }
7    9
8    10       private void demoMethodIllustrateCommitLink()
```

updated task #19

Description

The new commit changes appear in PR #20



implementation of task ×

GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pull/20#pullrequestreview-16052664

bgdndit reviewed 4 minutes ago

summary of review...

src/Main.java

```
...   ...  @@ -1,5 +1,10 @@
1     1     public class Main
2     2     {
      3   +     private void demoMethodTask19()
      4   +     {
      5   +         //implementation of task #19 in testPRBranch
```

bgdndit 4 minutes ago

leave feedback directly in the code...

bgdndit 4 minutes ago

continue with the review

updated task #19 08644da

Add more commits by pushing to the **testPRBranch** branch on BoiseState/CS481-Test.

Review requested

Review has been requested on this pull request. It is not required to merge. Learn more.

Sprint 1

Estimate

No estimate yet

Epics

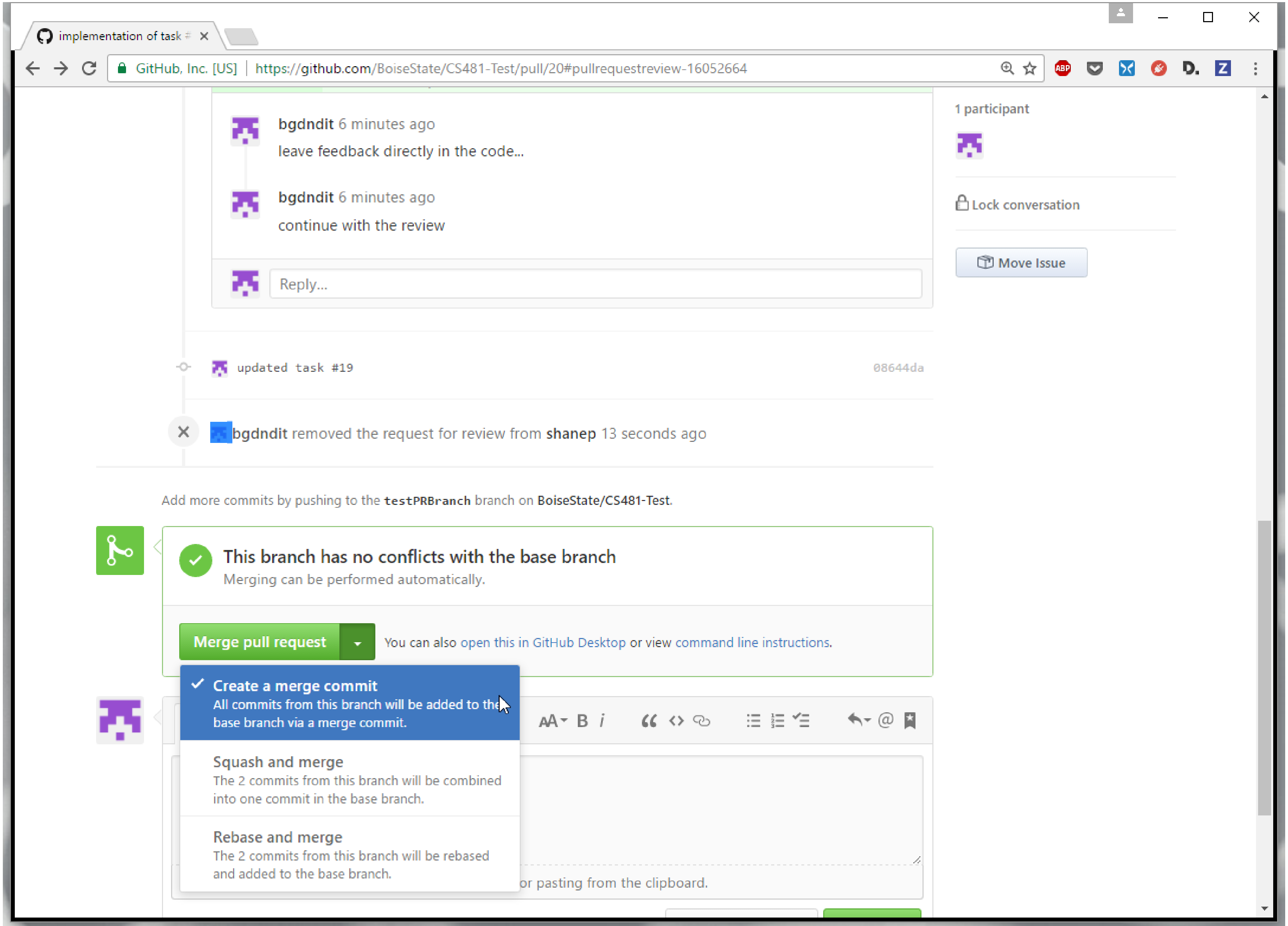
Not inside an Epic

Notifications

You're receiving notifications because you were assigned.

1 participant

Merge the pull request into the master branch



implementation of task #
GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pull/20#pullrequestreview-16052664

bgdndit 6 minutes ago

leave feedback directly in the code...

bgdndit 6 minutes ago

continue with the review

Reply...
1 participant
Lock conversation
Move Issue

updated task #19 08644da

bgdndit removed the request for review from **shanep** 13 seconds ago

Add more commits by pushing to the `testPRBranch` branch on BoiseState/CS481-Test.

This branch has no conflicts with the base branch

Merging can be performed automatically.

You can also open this in GitHub Desktop or view command line instructions.

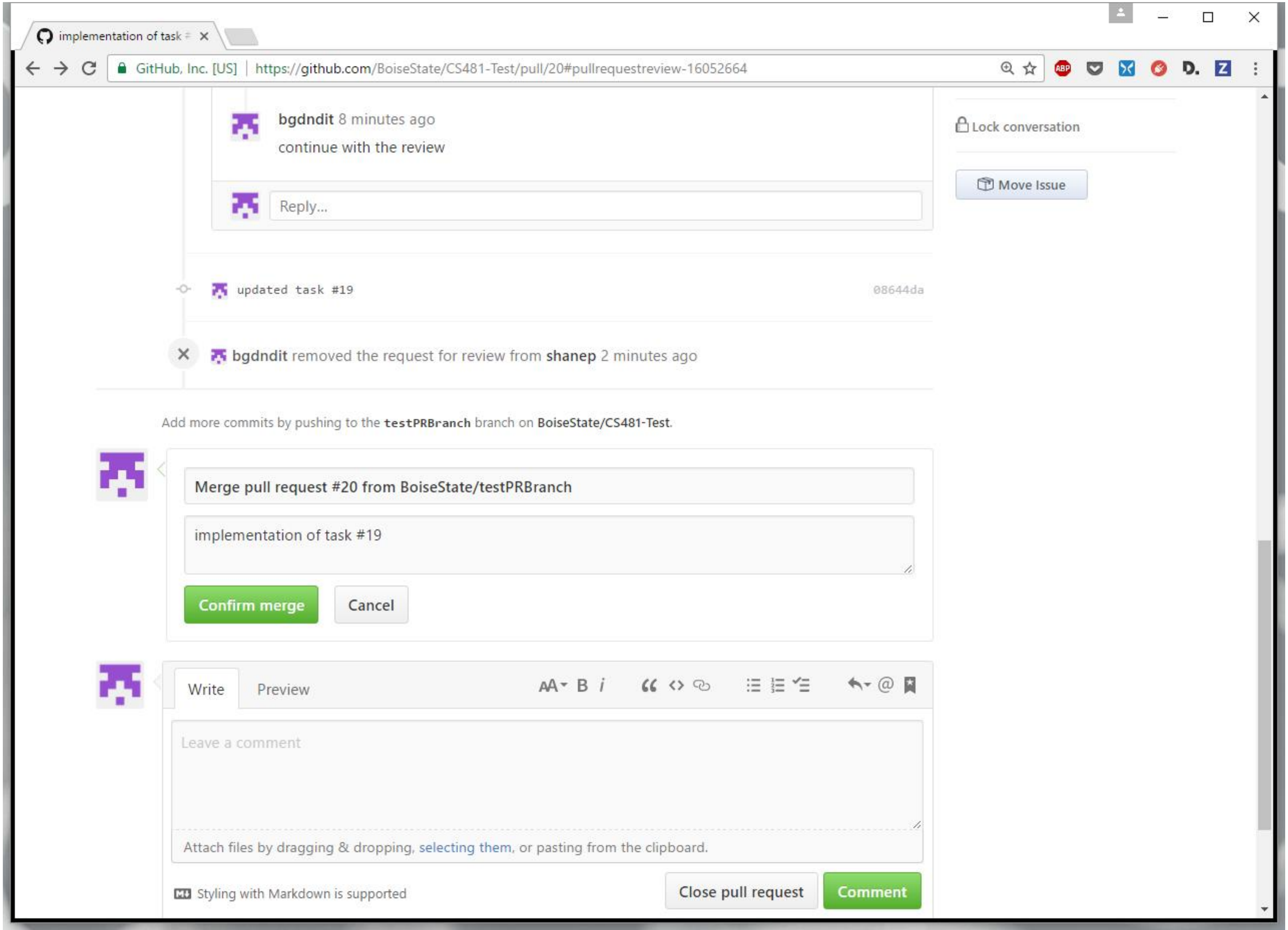
✓ **Create a merge commit**
All commits from this branch will be added to the base branch via a merge commit.

Squash and merge
The 2 commits from this branch will be combined into one commit in the base branch.

Rebase and merge
The 2 commits from this branch will be rebased and added to the base branch.

or pasting from the clipboard.

Finish the PR by merging it into the master branch



implementation of task
GitHub, Inc. [US] | https://github.com/BoiseState/CS481-Test/pull/20#pullrequestreview-16052664

bgdndit 8 minutes ago

continue with the review

Reply...

updated task #19 08644da

bgdndit removed the request for review from shanep 2 minutes ago

Lock conversation
Move Issue

Add more commits by pushing to the testPRBranch branch on BoiseState/CS481-Test.

Merge pull request #20 from BoiseState/testPRBranch

implementation of task #19

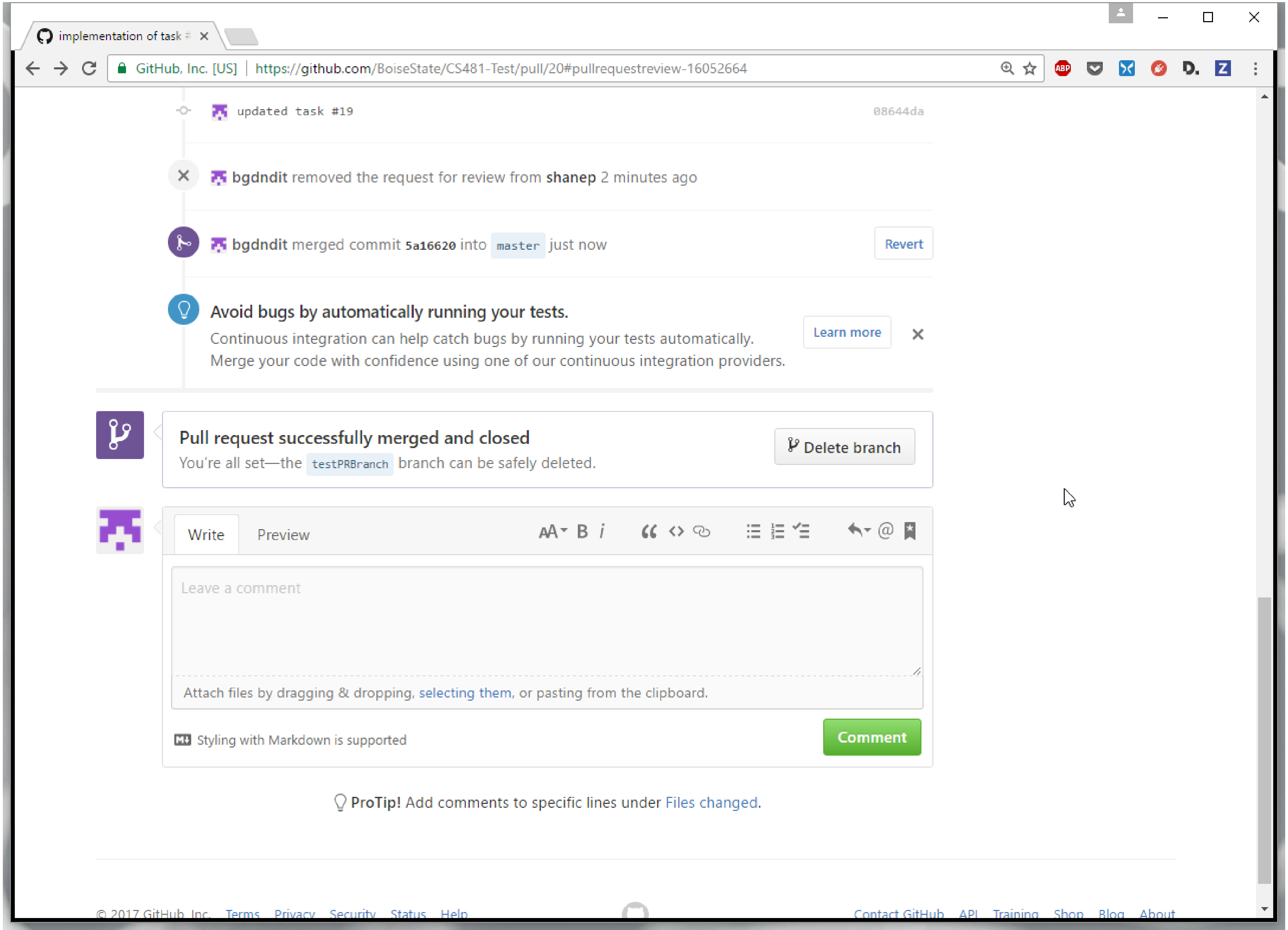
Confirm merge
Cancel
Write
Preview

Leave a comment

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

After successful PR merge and close, delete the branch



implementation of task #
GitHub, Inc. [US]
https://github.com/BoiseState/CS481-Test/pull/20#pullrequestreview-16052664
ABP
Revert
Learn more
Delete branch
Write
Preview
AA B i
Comment
© 2017 GitHub, Inc. Terms Privacy Security Status Help
Contact GitHub API Training Shop Blog About

updated task #19 08644da

bgdndit removed the request for review from shanep 2 minutes ago

bgdndit merged commit 5a16620 into master just now

Avoid bugs by automatically running your tests.

Continuous integration can help catch bugs by running your tests automatically. Merge your code with confidence using one of our continuous integration providers.

Pull request successfully merged and closed

You're all set—the testPRBranch branch can be safely deleted.

Leave a comment

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

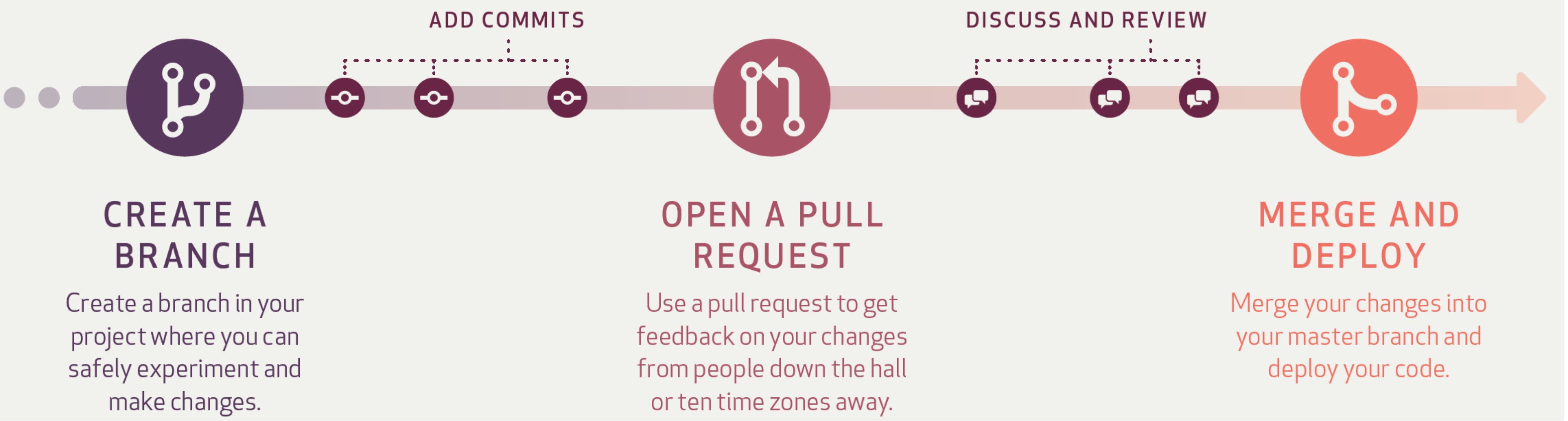
Styling with Markdown is supported

ProTip! Add comments to specific lines under Files changed.

GitHub Workflow

GitHub Workflow

- https://guides.github.com/introduction/flow/



CREATE A BRANCH

Create a branch in your project where you can safely experiment and make changes.

OPEN A PULL REQUEST

Use a pull request to get feedback on your changes from people down the hall or ten time zones away.

MERGE AND DEPLOY

Merge your changes into your master branch and deploy your code.

- Best suited for open-source software development