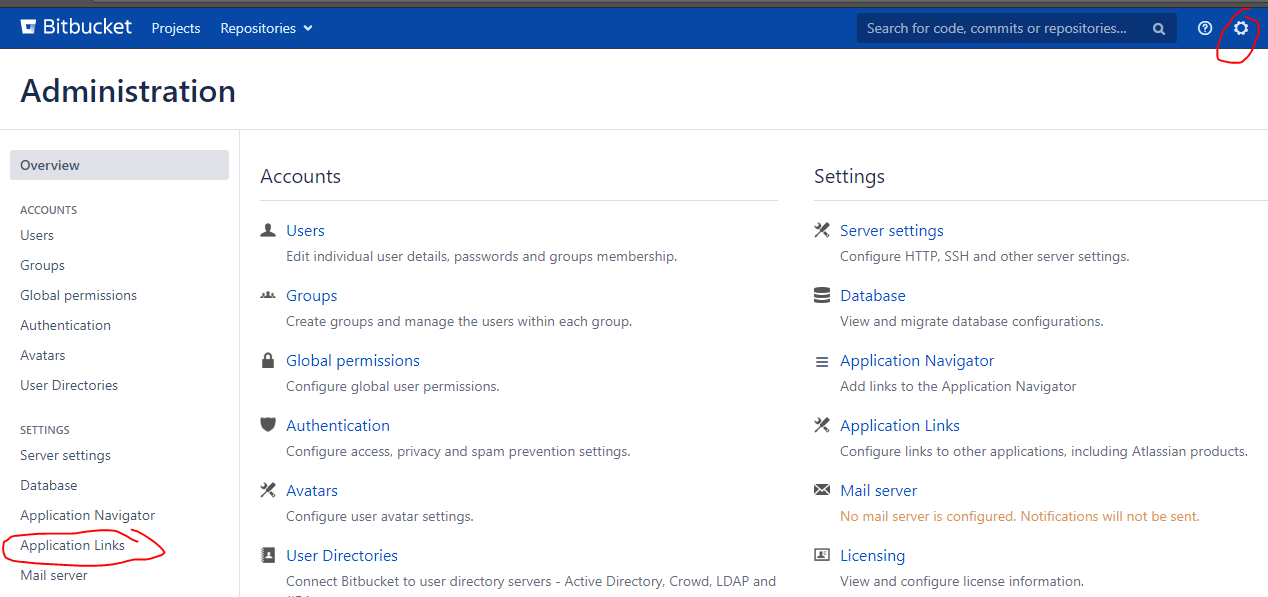
**Integrate JIRA with bitbucket server**

|  |
| --- |
| Note: Please view this image in “Web Layout” to see the complete image,  right side bottom of the word doc. |

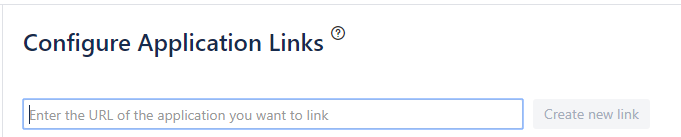
**Note: To practice with Phase-4, make sure JIRA is already installed and server is running.**

Integrate JIRA with bitbucket server, so that it will be easy for development, easy to divide the tasks among the dev team.

To integrate the JIRA with Bitbucket🡪 go to Bitbucket settings 🡪 click on Application Links.



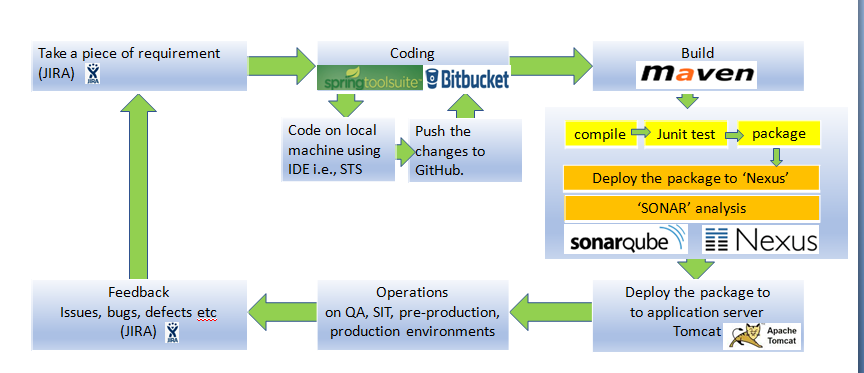
Enter the JIRA server URL: It will integrate with JIRA automatically.



**Let’s do some exercise:**

Take some requirement 🡪 coding on local machine 🡪 push the changes to Bitbucket with JIRA ticket reference 🡪 build with maven 🡪 deploy the package to tomcat server (skipping the part deploy the package to nexus as we already aware of it in Phase-1 & skipping another part analysing the sonar analysis as this is completed in Phase-3).

**Our goal:**



**Step-1:** Take some requirement: WEB-1

**Step-2:** Coding locally : Do the changes as per the requirement either in STS or in notepad editor.

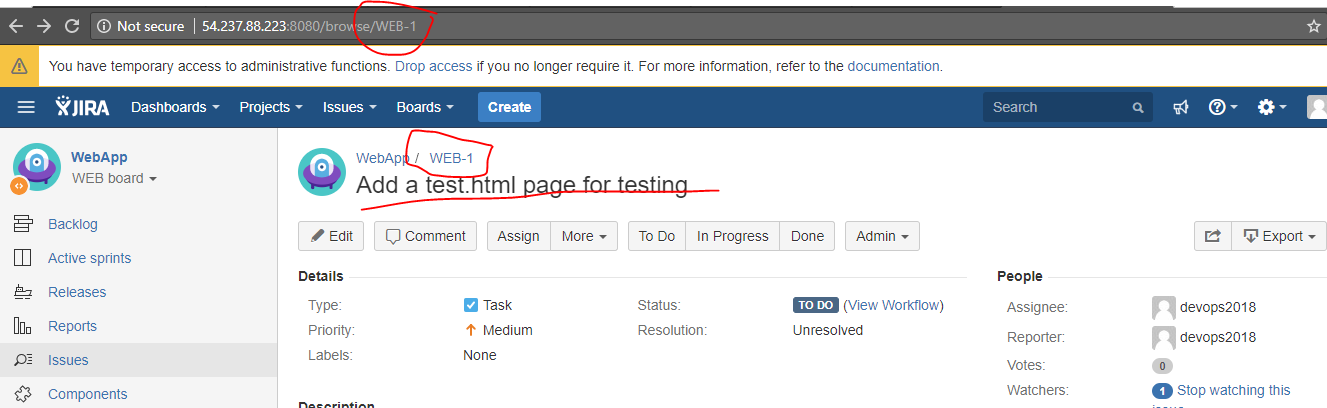
**Step-3:** push the changes to Bitbucket with JIRA ticket reference: Give the JIRA task WEB-1 in the commit message.

**Step-4:** Create a pull request to review by the team lead & merge and then approve for build & deploy.

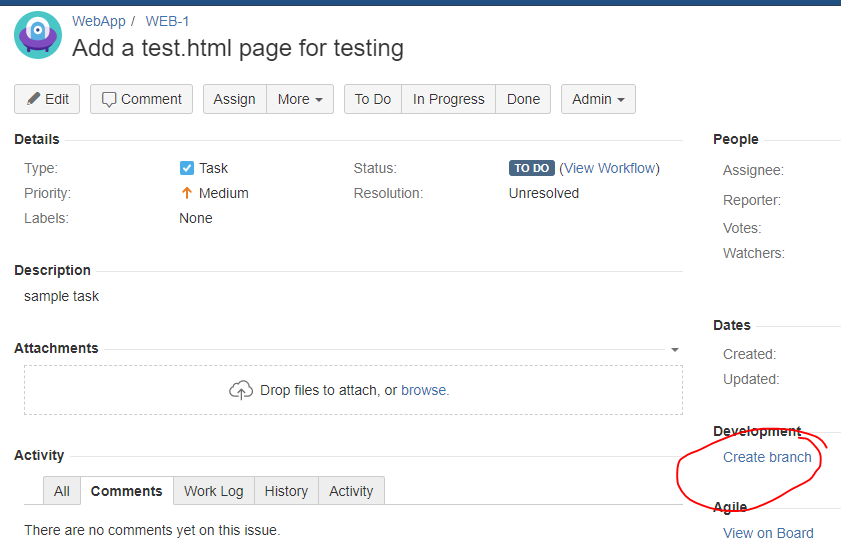
**Step-5:** build with maven: and deploy the package to tomcat server: Build a job in Jenkins as to package & deploy the new changes on tomcat server.

**Step-1:** Take some requirement: WEB-1

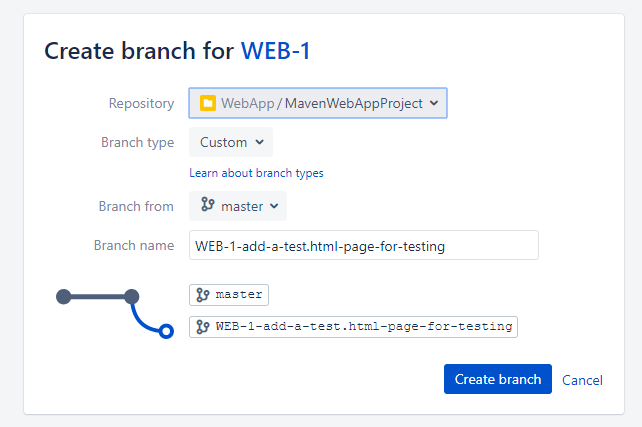
Add the test.html page to the code is our requirement. For this requirement creating a task in JIRA and assume assign the task to dev team.

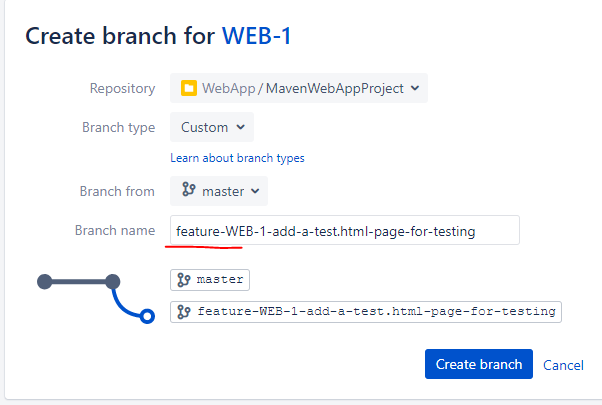


Dev team will this task and do the changes by creating a feature branch.



Click on “create branch”



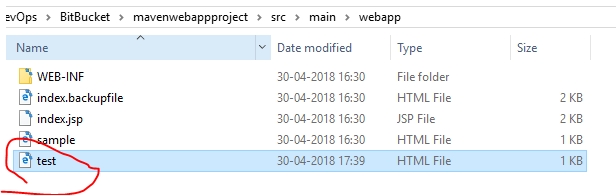


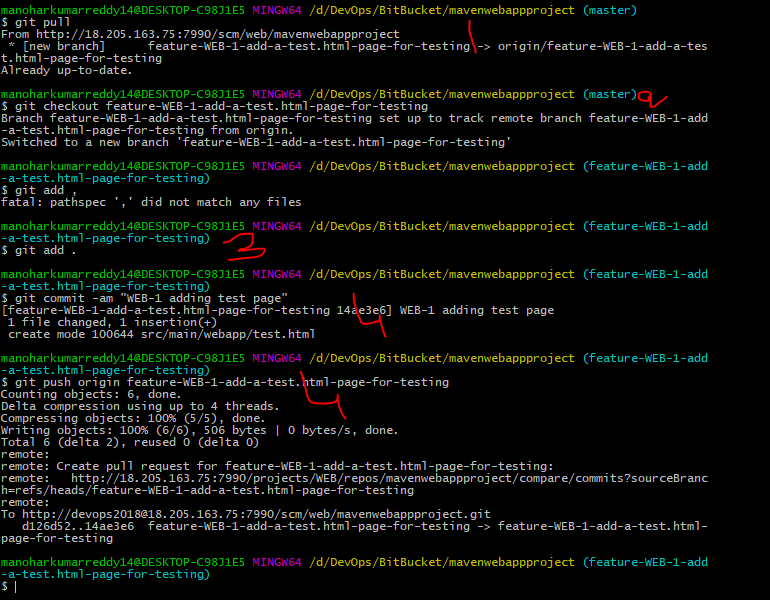
**Step-2 & 3:** Coding locally : Do the changes as per the requirement either in STS or in notepad editor.

As the direct commit & push to the ‘master’ branch is restricted (in real time), creating a feature branch for every dev team member or every change/requirement and then switch our local to the featire branch. So that dev team (whoever is going to work on feature branch) they will commit their changes.

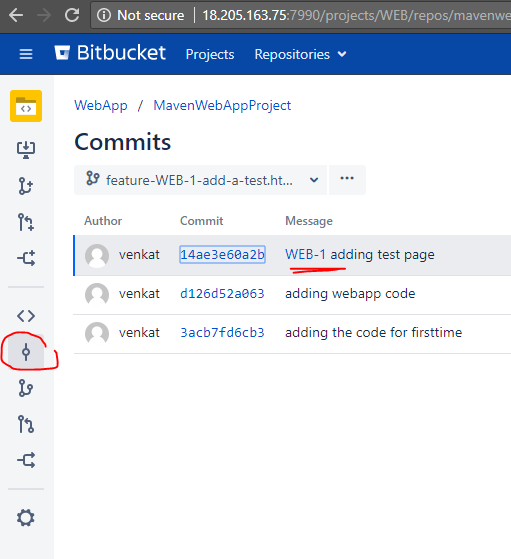
For this,

1. Git pull (update the local workspace if you already cloned) OR git clone.
2. Switch to the feature branch to complete the requirement.
3. Add the test.html page as per the requirement WEB-1.
4. Commit & push the changes with the JIRA task WEB-1 reference, see the commit message starts with WEB-1. (if you don’t give the JIRA task WEB-1 as a reference in the commit message, JIRA issue task WEB-1 can’t able to track these changes). We are giving JIRA issue task WEB-1 as a reference only for the purpose EASY tacking. In future we can easily find the answer for the question “who & why we did these changes in this project”.



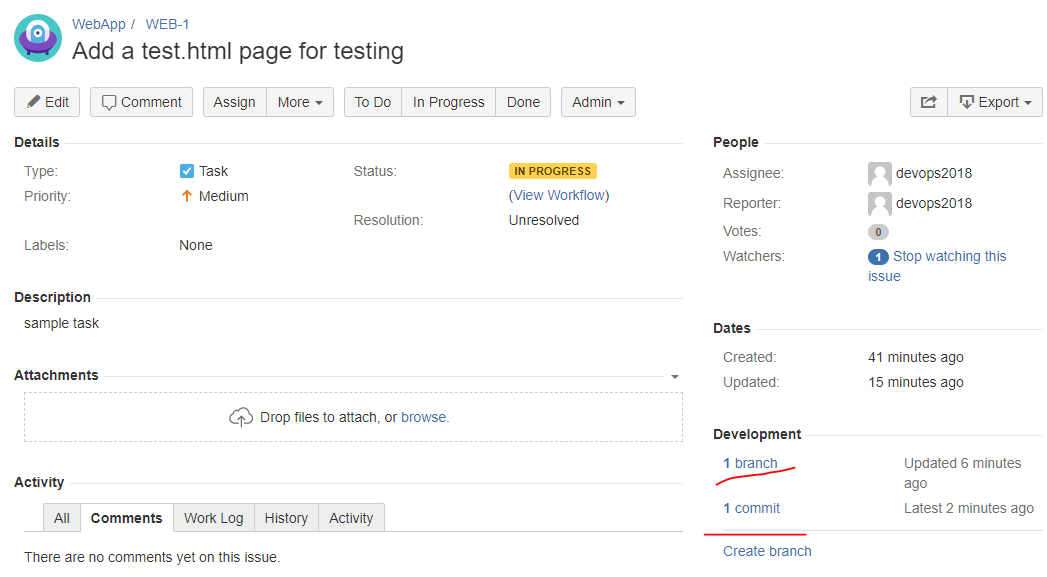


Check the commit history whether the local commit pushed to remote successfully or not.

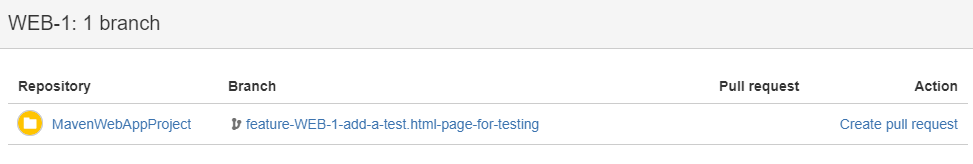


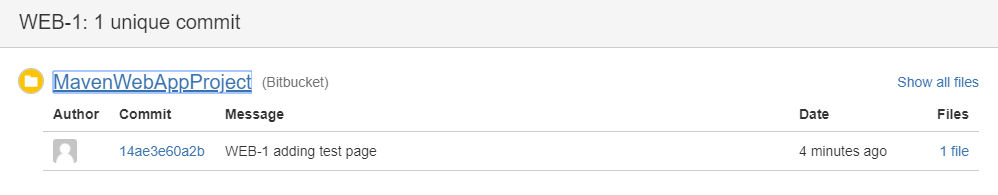
**Step-4:** Create a pull request to review by the team lead & merge and then approve for build & deploy.

Now, If you observe the task, WEB-1, its showing how many branches, & commits are created for this particular requirement. Click on commit & branch.



Click on commit & branch.

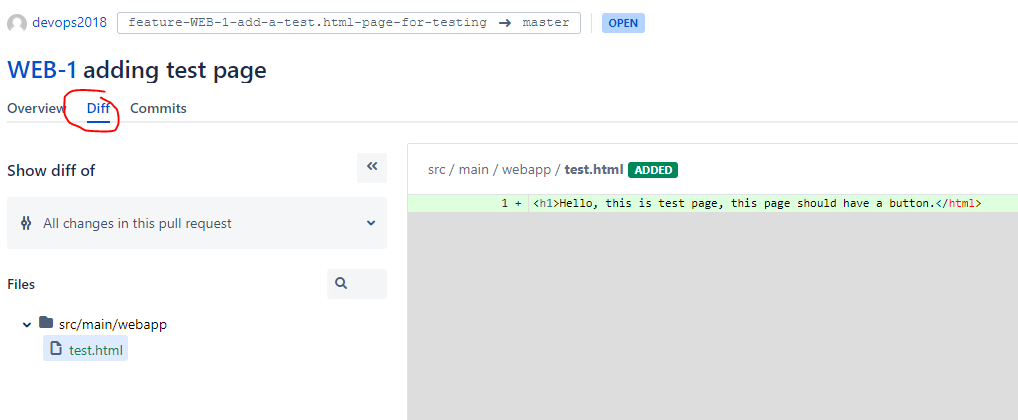


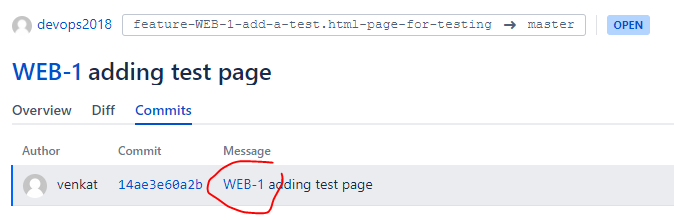


Now, create a pull request (click on branch🡪 Create pull request) to merge the developer changes to master. So that reviewer (team lead or manger) will check the changes and proceed with merge.

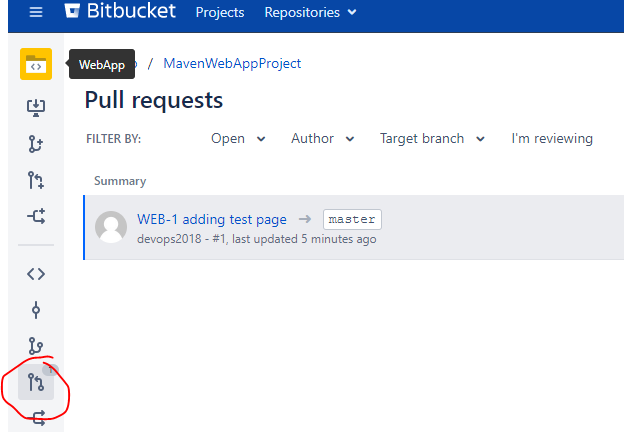


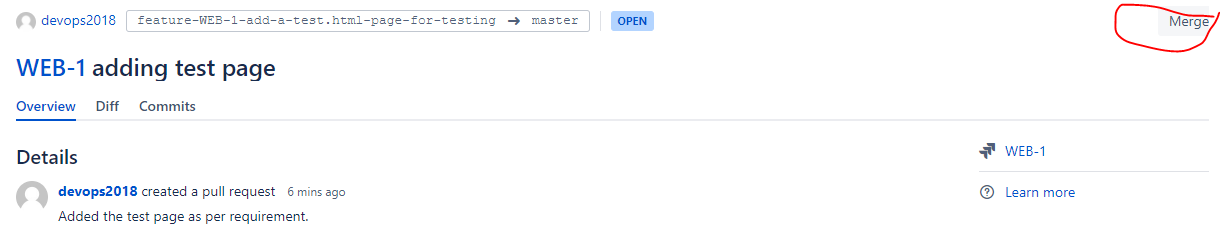
Once the PR (pull request) created reviewer will get email notification and then the reviewer will check the diff and then add some comments if additional changes required or proceed with merge the changes from feature branch to master branch.

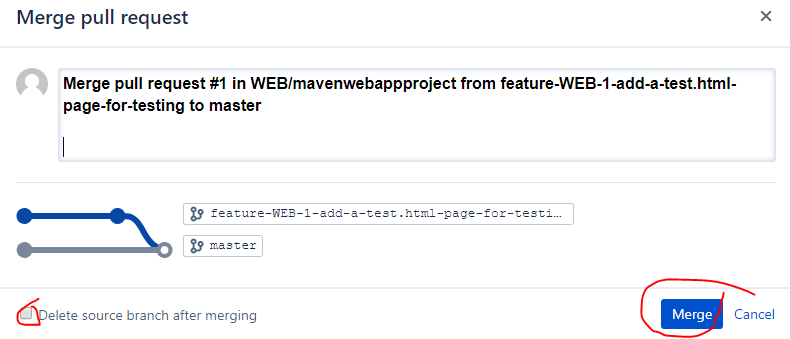


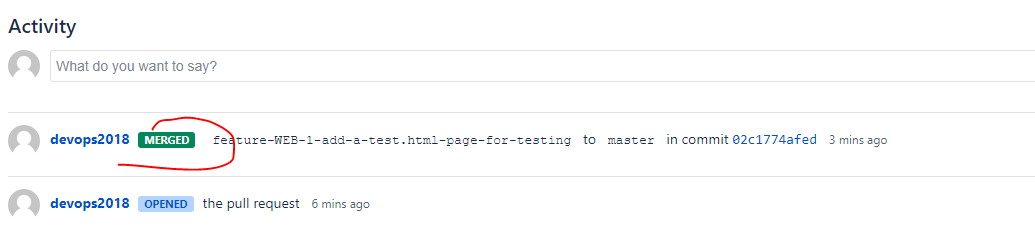


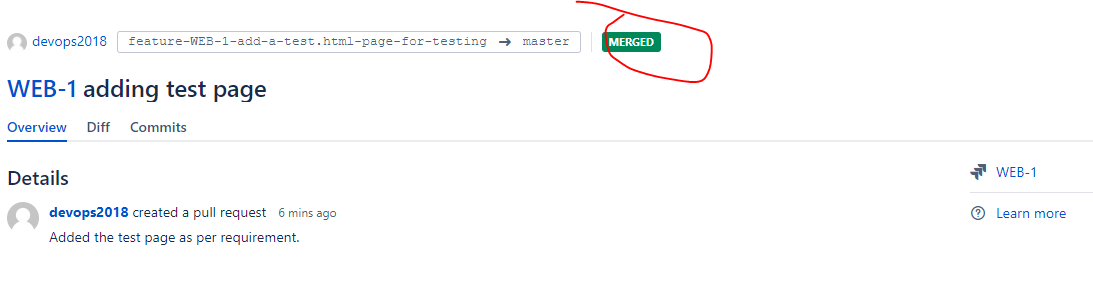
Click on pull requests to see all the pull requests:



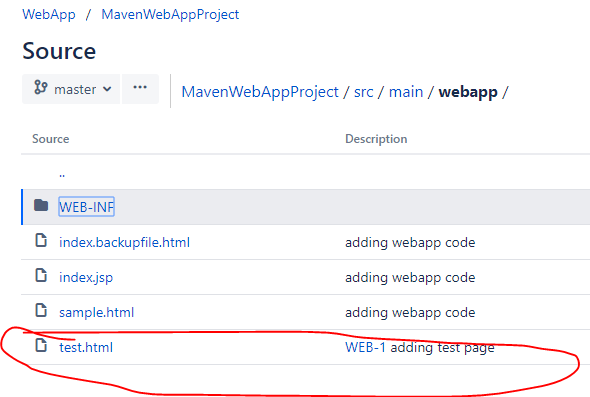






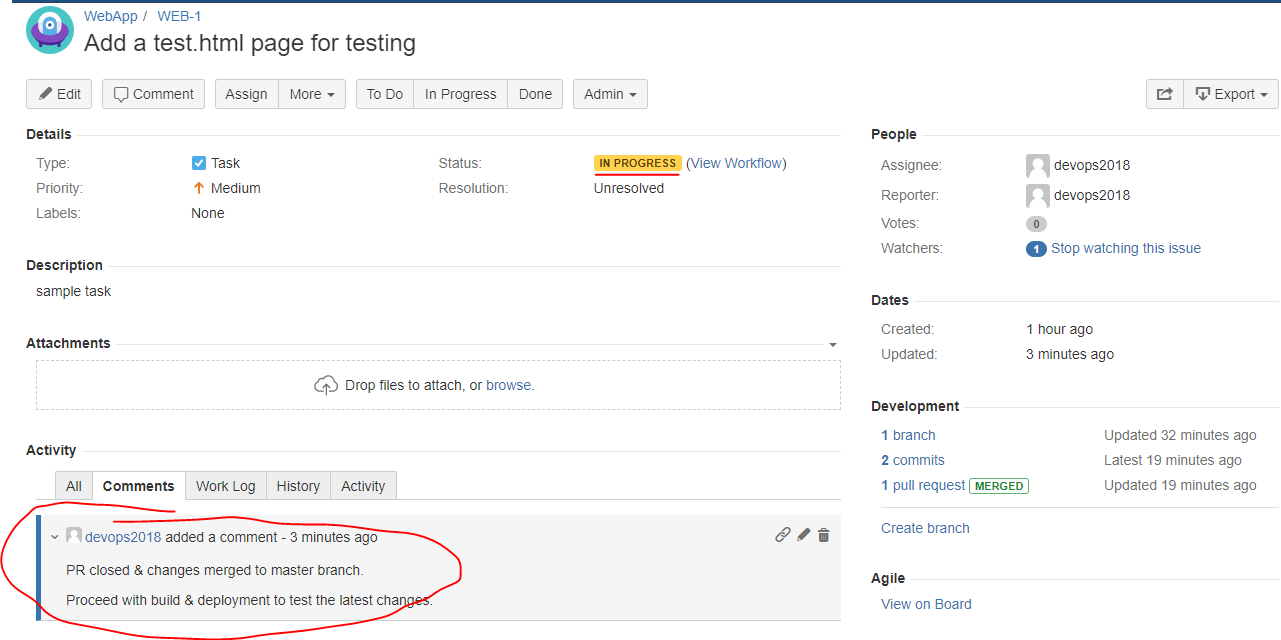


Merge is success & commits are merged to master branch.

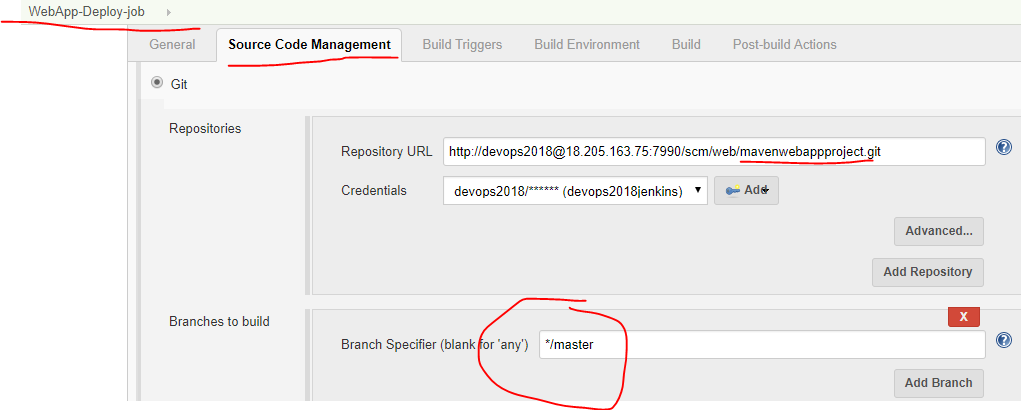


**Step-5:** build with maven: and deploy the package to tomcat server: Build a job in Jenkins as to package & deploy the new changes on tomcat server.

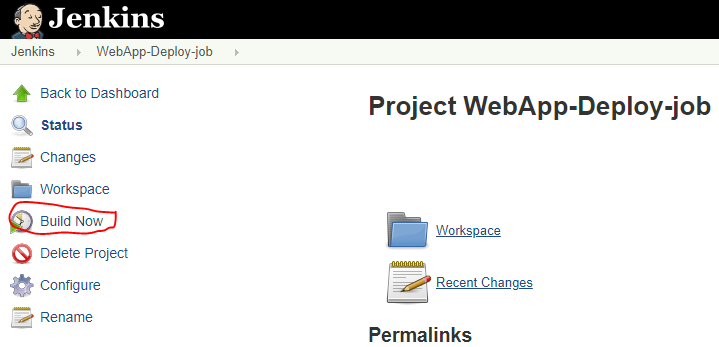
Now the changes are (latest code) in master branch. To check the application with latest changes dev lead wants to proceed with build & deploy.

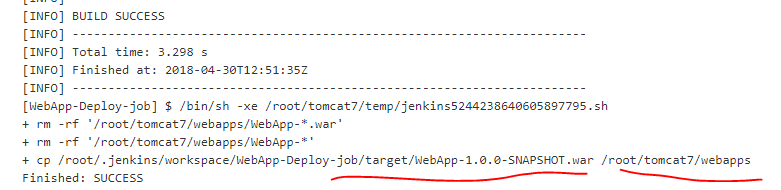


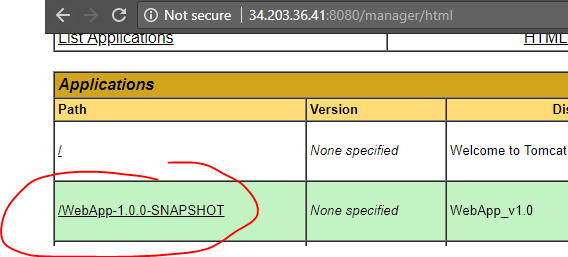
Go to Jenkins --> Click on “WebApp-Deploy-job” 🡪 see the Jenkins job configuration.

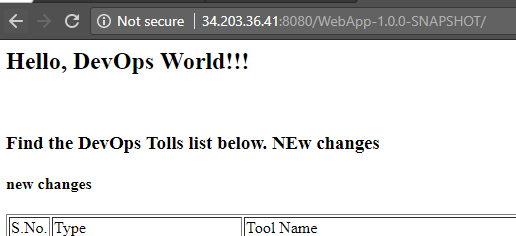






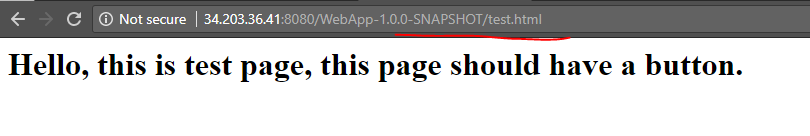






As per our changes, we have to check the newly added page ‘test.html’.

<http://34.203.36.41:8080/WebApp-1.0.0-SNAPSHOT/test.html>



So far we have done the below changes.

Take some requirement 🡪 coding in local system 🡪 push the changes to Bitbucket server with JIRA ticket reference 🡪 build with maven 🡪 deploy the package to tomcat server.

Refer the another doc for JIRA + Bitbucket + Jenkins integration and CI/CD process.