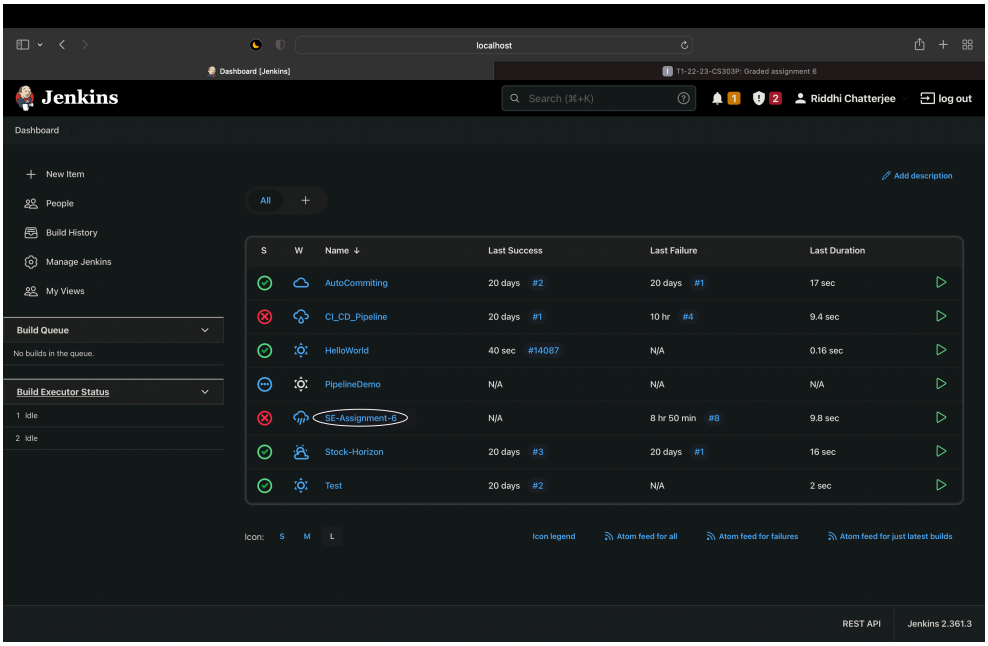
Assignment - 6
Report

Name: Riddhi Chatterjee
Roll Number: IMT2020094

Setting up the project in the Jenkins environment:

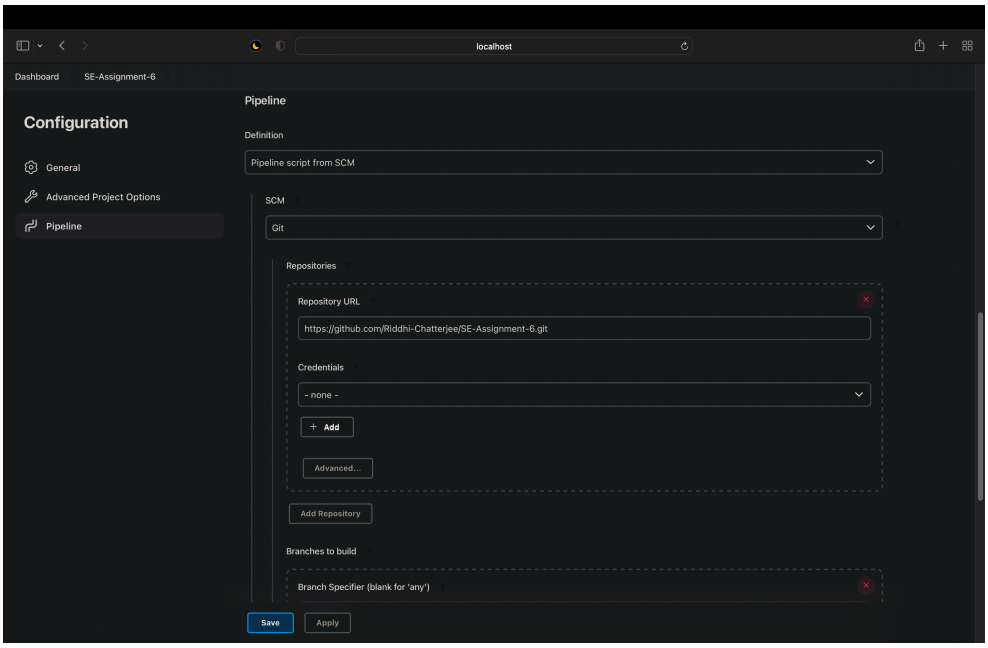
We can see our pipeline project “SE-Assignment-6” in the Jenkins dashboard.



Linking GitHub source code:

We select the “Pipeline script from SCM” option and provide the link to our git repository. Currently our repository is public, so we don’t need to provide credentials.

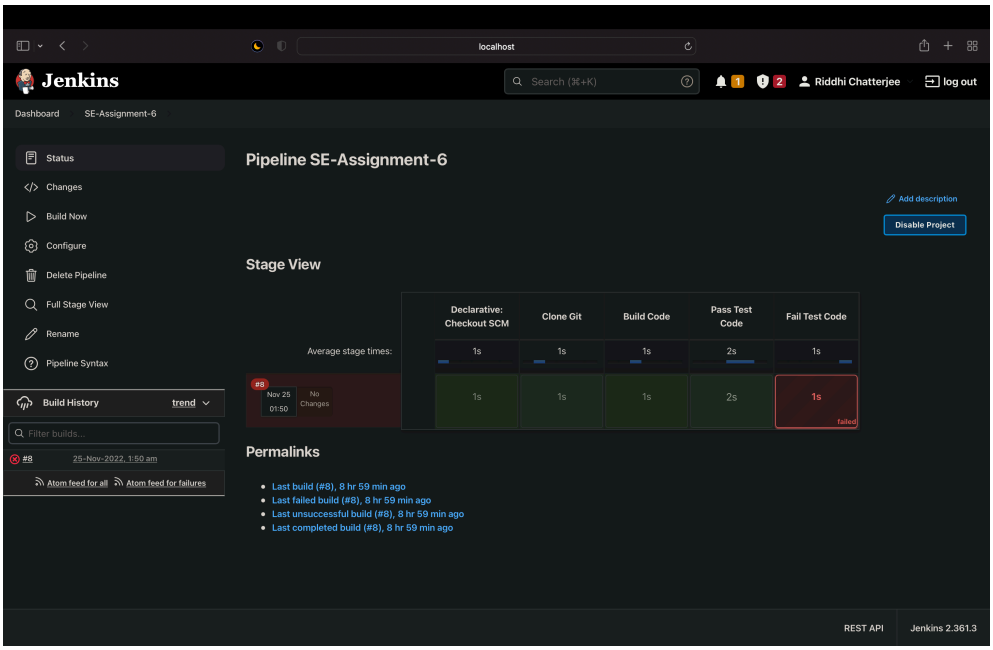
Link to git repository: https://github.com/Riddhi-Chatterjee/SE-Assignment-6



Execution of pipeline stages:

The following figure shows the execution of the pipeline stages. The stages get executed sequentially and Jenkins shows us the execution time and pass/fail status of each stage.

We can see that all stages have passed except the “Fail Test Code” stage. This is in accordance with what we had expected because the “Fail Test Code” stage executes the unit test cases that were built to fail.

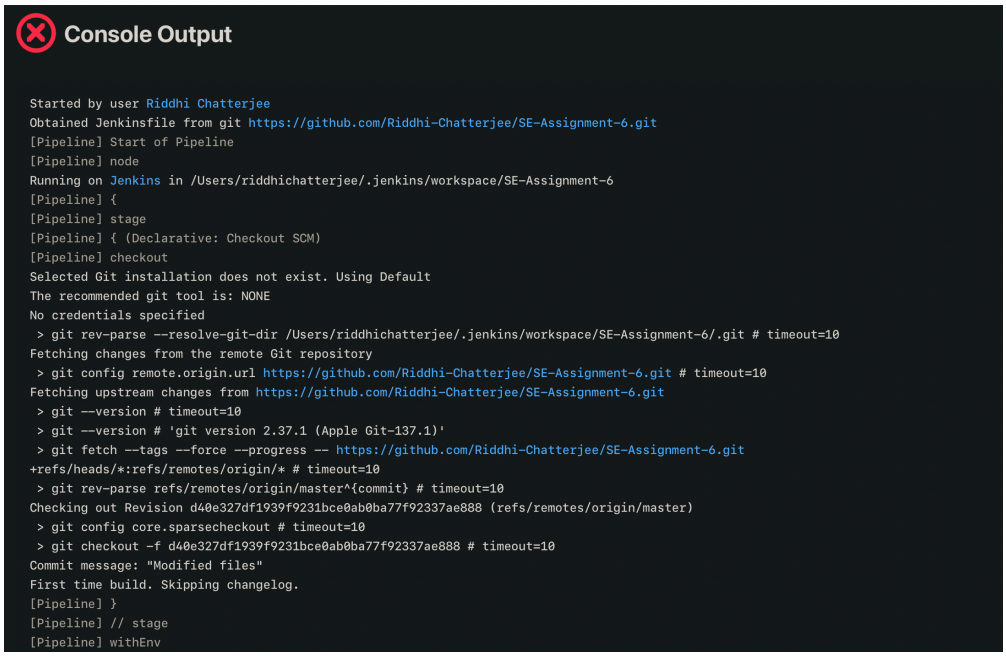


The console output for each of the pipeline stages is as follows:

- ## Declarative: Checkout SCM stage:

  The multi-branch workflow checkouts SCM repositories in a special stage “Declarative: Checkout SCM" (if it is not disabled by the skipDefaultCheckout option).

  The checkout step will checkout code from source control; scm is a special variable which instructs the checkout step to clone the specific revision which triggered this Pipeline run.



- **Clone Git stage**

```
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Clone Git)
[Pipeline] git
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
 > git rev-parse --resolve-git-dir /Users/riddhichatterjee/.jenkins/workspace/SE-Assignment-6/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/Riddhi-Chatterjee/SE-Assignment-6 # timeout=10
Fetching upstream changes from https://github.com/Riddhi-Chatterjee/SE-Assignment-6
 > git --version # timeout=10
 > git --version # 'git version 2.37.1 (Apple Git-137.1)'
 > git fetch --tags --force --progress -- https://github.com/Riddhi-Chatterjee/SE-Assignment-6 +refs/heads/*:refs/remotes/origin/* #
timeout=10
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision d40e327df1939f9231bce0ab0ba77f92337ae888 (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f d40e327df1939f9231bce0ab0ba77f92337ae888 # timeout=10
 > git branch -a -v --no-abbrev # timeout=10
 > git branch -D master # timeout=10
 > git checkout -b master d40e327df1939f9231bce0ab0ba77f92337ae888 # timeout=10
Commit message: "Modified files"
[Pipeline] }
[Pipeline] // stage
```

- **Build Code stage:**

```
[Pipeline] stage
[Pipeline] { (Build Code)
[Pipeline] sh
+ chmod u+x Division.py
[Pipeline] sh
+ ./Division.py
[Pipeline] }
[Pipeline] // stage
```

- **Pass results of unit test cases:**

We can see that the three unit test cases which were built to pass have passed indeed.
The test cases are as follows:

1. **test_case_pass_1:** Computes 10/5 and asserts the resultant value to be 2.0
2. **test_case_pass_2:** Computes 1/4 and asserts the resultant value to be 0.25
3. **test_case_pass_3:** Computes -4/2 and asserts the resultant value to be -2.0

```
[Pipeline] stage
[Pipeline] { (Pass Test Code)
[Pipeline] sh
+ chmod u+x PassTestCases.py
[Pipeline] sh
+ ./PassTestCases.py
...
----------------------------------------------------------------------
Ran 3 tests in 0.000s

OK
test_case_pass_1 passed!
test_case_pass_2 passed!
test_case_pass_3 passed!
[Pipeline] }
[Pipeline] // stage
```

- ## Fail results of unit test cases:

  We can see that the two unit test cases which were built to fail have failed indeed.
  The test cases are as follows:

  1. **test_case_fail_1:** Tries to compute 10/0 and thus results in ZeroDivisionError
  2. **test_case_fail_2:** Computes 100/2 but asserts the resultant value to be 500.0 and thus results in AssertionError

```
[Pipeline] stage
[Pipeline] { (Fail Test Code)
[Pipeline] sh
+ chmod u+x FailTestCases.py
[Pipeline] sh
+ ./FailTestCases.py
EF
======================================================================
ERROR: test_case_fail_1 (__main__.Tests)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "/Users/riddhichatterjee/.jenkins/workspace/SE-Assignment-6/./FailTestCases.py", line 11, in test_case_fail_1
    result = divide(a, b)
  File "/Users/riddhichatterjee/.jenkins/workspace/SE-Assignment-6/Division.py", line 5, in divide
    return a/b
ZeroDivisionError: division by zero

======================================================================
FAIL: test_case_fail_2 (__main__.Tests)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "/Users/riddhichatterjee/.jenkins/workspace/SE-Assignment-6/./FailTestCases.py", line 19, in test_case_fail_2
    self.assertEqual(result, 500.0)
AssertionError: 50.0 != 500.0

----------------------------------------------------------------------
Ran 2 tests in 0.001s

FAILED (failures=1, errors=1)
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
ERROR: script returned exit code 1
Finished: FAILURE
```