哈尔滨工业大学 计算学部

2024 年秋季学期《开源软件开发实践》

Lab4：开源软件开发中的 DevOps

学号	姓名	联系方式
2023120252	席鹏轩	1749367694@qq.com/15939201851

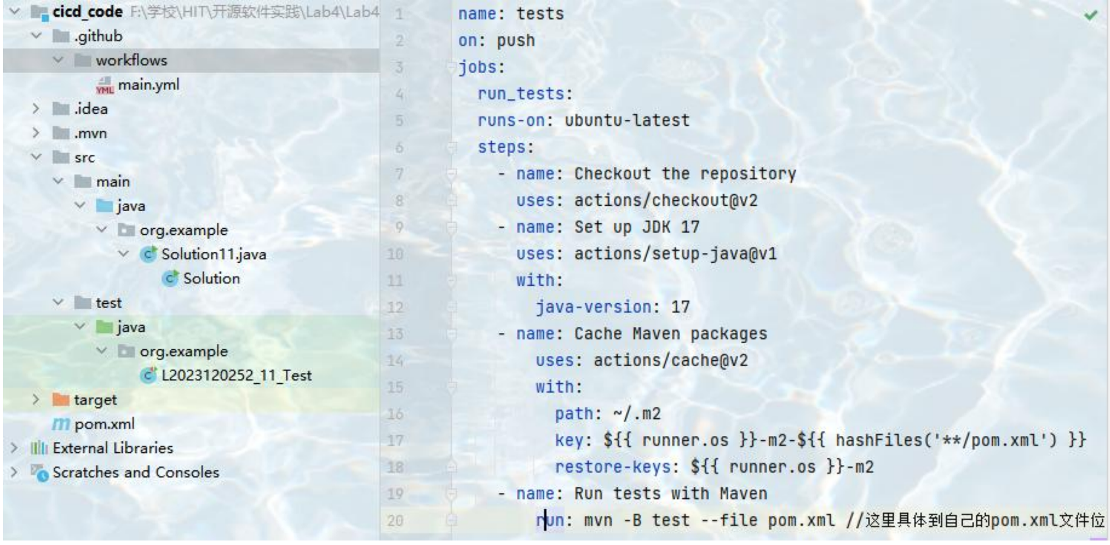
目　录

1 实验要求

本次实验训练开源软件开发中的基本 DevOps 操作：
1.掌握开源软件开发中的基本 DevOps 流程和工具的使用
2.熟悉利用 Github Actions 进行 DevOps
3.熟悉利用 Jenkins 进行 DevOps

2 实验内容 1 Github Actions DevOps 实践

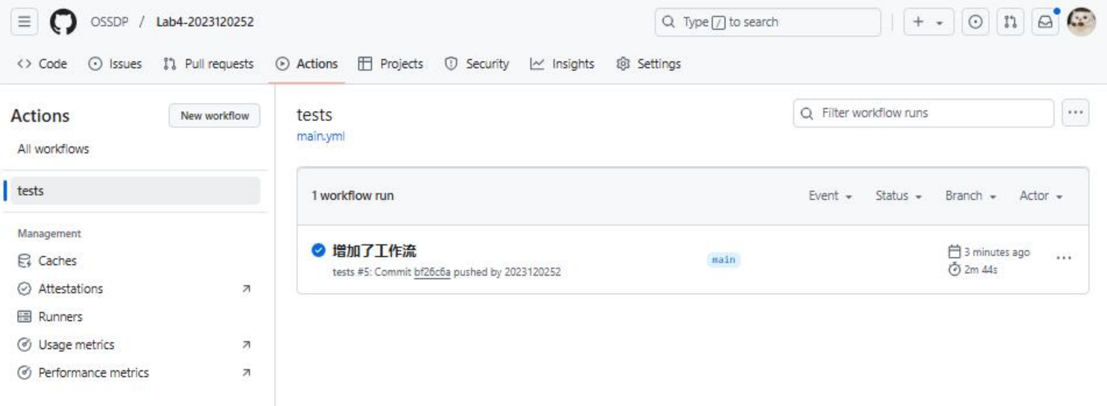
目录结构：



```
cicd_code  F:\学校\HIT\开源软件实践\Lab4\Lab4
  .github
    workflows
      main.yml
  .idea
  .mvn
  src
    main
      java
        org.example
          Solution11.java
            Solution
    test
      java
        org.example
          L2023120252_11_Test
  target
  pom.xml
External Libraries
Scratches and Consoles
```

```yaml
name: tests
on: push
jobs:
  run_tests:
  runs-on: ubuntu-latest
  steps:
    - name: Checkout the repository
      uses: actions/checkout@v2
    - name: Set up JDK 17
      uses: actions/setup-java@v1
      with:
        java-version: 17
    - name: Cache Maven packages
        uses: actions/cache@v2
        with:
          path: ~/.m2
          key: ${{ runner.os }}-m2-${{ hashFiles('**/pom.xml') }}
          restore-keys: ${{ runner.os }}-m2
    - name: Run tests with Maven
        run: mvn -B test --file pom.xml //这里具体到自己的pom.xml文件位
```

执行自动化测试成功界面：

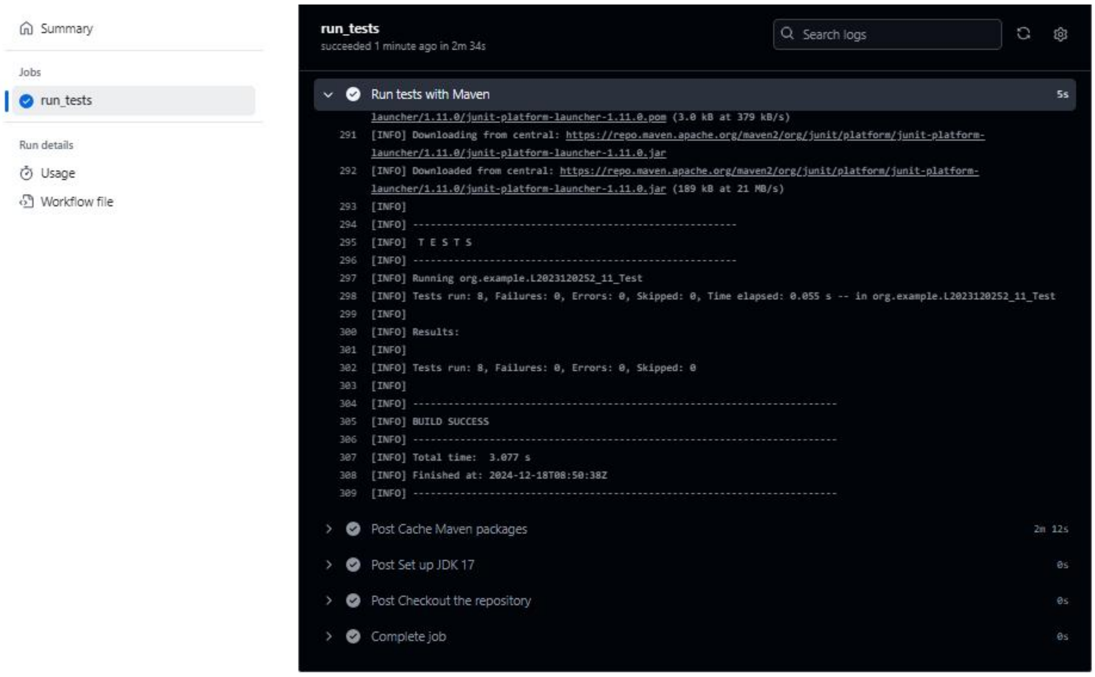


tests
main.yml

1 workflow run

✓ 增加了工作流
tests #5: Commit bf26c6a pushed by 2023120252　main　3 minutes ago　2m 44s

执行的具体信息：



run_tests
succeeded 1 minute ago in 2m 34s

Run tests with Maven 5s

```
launcher/1.11.0/junit-platform-launcher-1.11.0.pom (3.0 kB at 379 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/junit/platform/junit-platform-
launcher/1.11.0/junit-platform-launcher-1.11.0.jar
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/junit/platform/junit-platform-
launcher/1.11.0/junit-platform-launcher-1.11.0.jar (189 kB at 21 MB/s)
[INFO]
[INFO] -------------------------------------------------------
[INFO]  T E S T S
[INFO] -------------------------------------------------------
[INFO] Running org.example.L2023120252_11_Test
[INFO] Tests run: 8, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.055 s -- in org.example.L2023120252_11_Test
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 8, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  3.077 s
[INFO] Finished at: 2024-12-18T08:50:38Z
[INFO] ------------------------------------------------------------------------
```

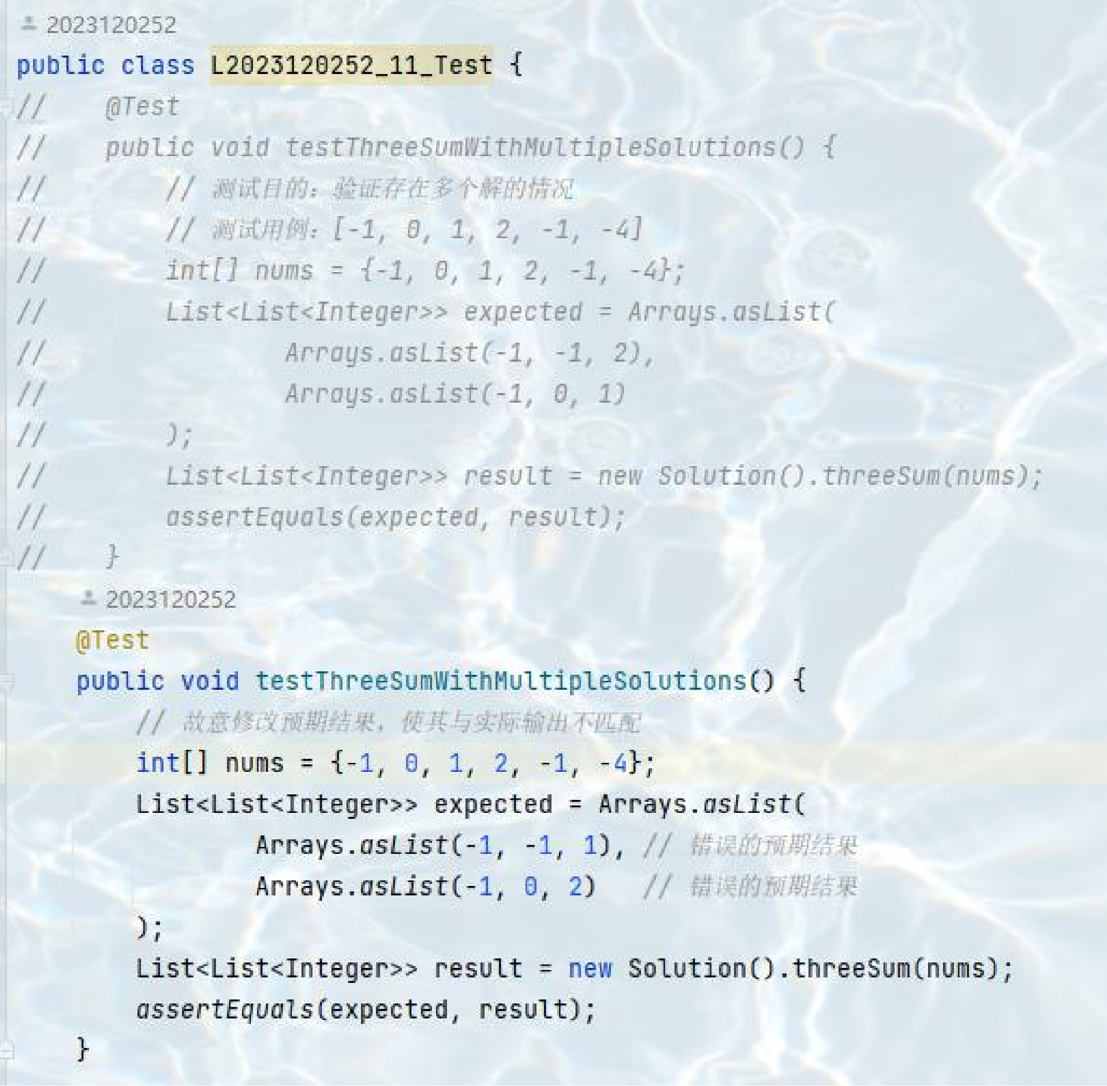
Post Cache Maven packages 2m 12s

Post Set up JDK 17 0s

Post Checkout the repository 0s

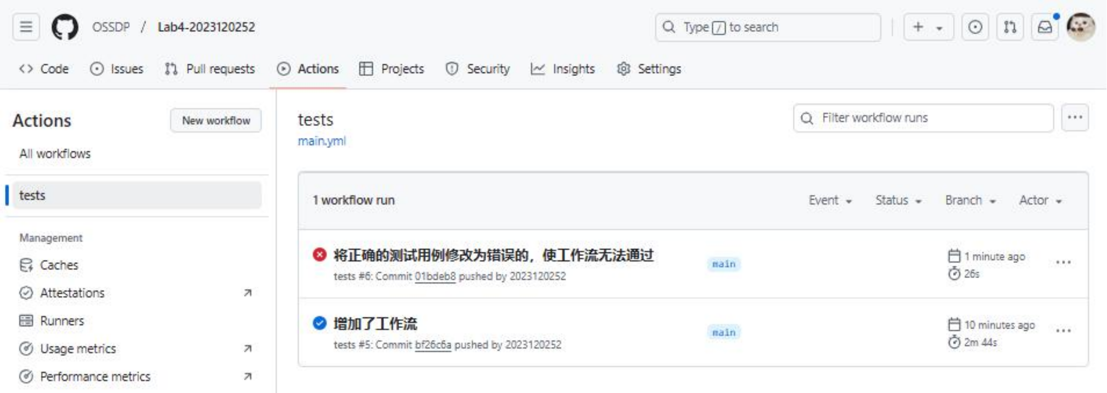
Complete job 0s

将正确的测试用例修改为错误的：



```java
2023120252
public class L2023120252_11_Test {
//    @Test
//    public void testThreeSumWithMultipleSolutions() {
//        // 测试目的：验证存在多个解的情况
//        // 测试用例：[-1, 0, 1, 2, -1, -4]
//        int[] nums = {-1, 0, 1, 2, -1, -4};
//        List<List<Integer>> expected = Arrays.asList(
//                Arrays.asList(-1, -1, 2),
//                Arrays.asList(-1, 0, 1)
//        );
//        List<List<Integer>> result = new Solution().threeSum(nums);
//        assertEquals(expected, result);
//    }
    2023120252
    @Test
    public void testThreeSumWithMultipleSolutions() {
        // 故意修改预期结果，使其与实际输出不匹配
        int[] nums = {-1, 0, 1, 2, -1, -4};
        List<List<Integer>> expected = Arrays.asList(
                Arrays.asList(-1, -1, 1), // 错误的预期结果
                Arrays.asList(-1, 0, 2)   // 错误的预期结果
        );
        List<List<Integer>> result = new Solution().threeSum(nums);
        assertEquals(expected, result);
    }
```

执行自动化测试失败界面：



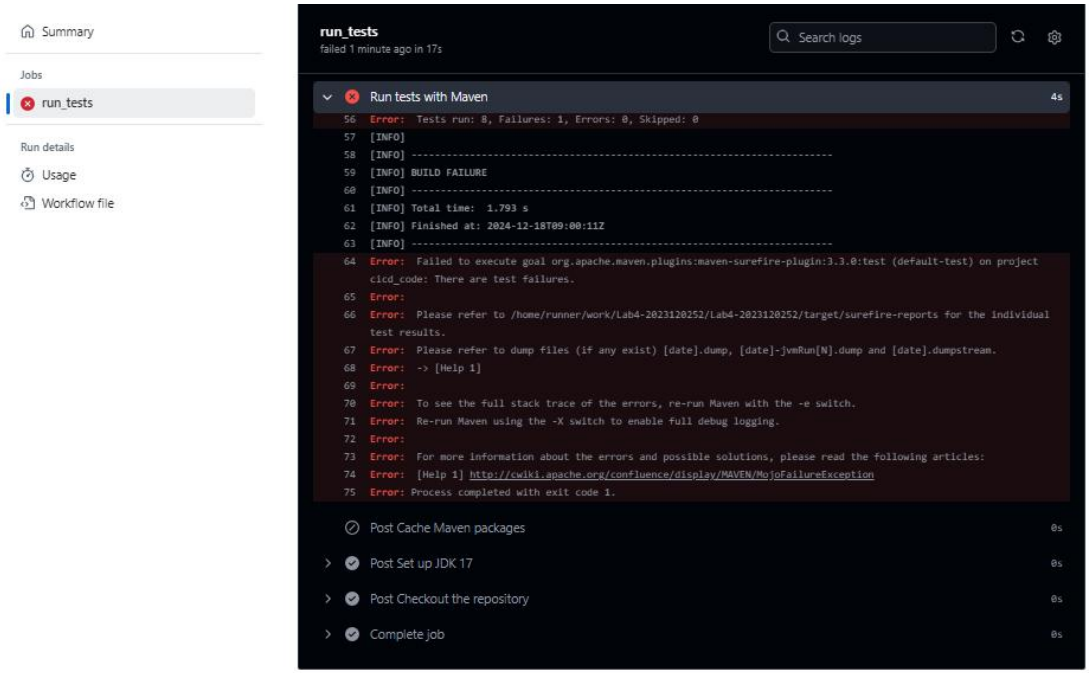
tests

main.yml

1 workflow run

✗ 将正确的测试用例修改为错误的，使工作流无法通过 tests #6: Commit 01bdeb8 pushed by 2023120252	main	1 minute ago 26s
✓ 增加了工作流 tests #5: Commit bf26c6a pushed by 2023120252	main	10 minutes ago 2m 44s

执行的具体错误信息：



run_tests

failed 1 minute ago in 17s

✗ Run tests with Maven 4s

```
Error:  Tests run: 8, Failures: 1, Errors: 0, Skipped: 0
[INFO]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD FAILURE
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  1.793 s
[INFO] Finished at: 2024-12-18T09:00:11Z
[INFO] ------------------------------------------------------------------------
Error:  Failed to execute goal org.apache.maven.plugins:maven-surefire-plugin:3.3.0:test (default-test) on project
cicd_code: There are test failures.
Error:
Error:  Please refer to /home/runner/work/Lab4-2023120252/Lab4-2023120252/target/surefire-reports for the individual
test results.
Error:  Please refer to dump files (if any exist) [date].dump, [date]-jvmRun[N].dump and [date].dumpstream.
Error:  -> [Help 1]
Error:
Error:  To see the full stack trace of the errors, re-run Maven with the -e switch.
Error:  Re-run Maven using the -X switch to enable full debug logging.
Error:
Error:  For more information about the errors and possible solutions, please read the following articles:
Error:  [Help 1] http://cwiki.apache.org/confluence/display/MAVEN/MojoFailureException
Error: Process completed with exit code 1.
```

Post Cache Maven packages 0s

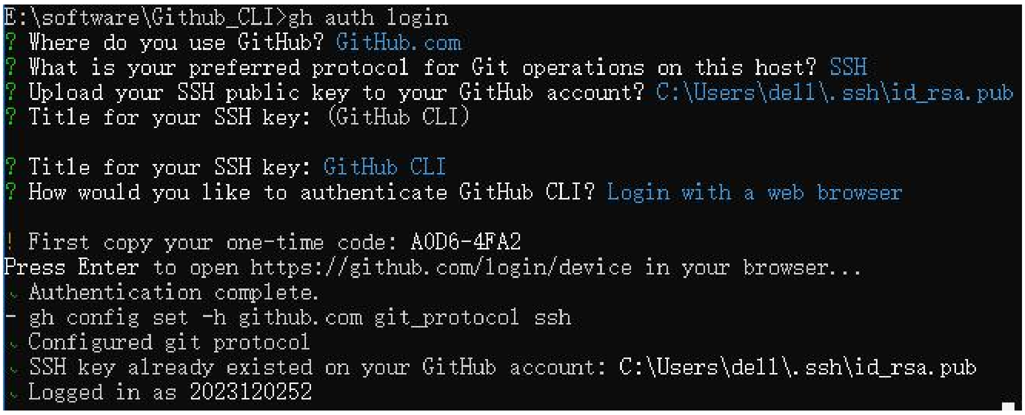
Post Set up JDK 17 0s

Post Checkout the repository 0s

Complete job 0s

3 实验内容 2 Jenkins DevOps 实践

准备工作：下载 Github CLI,并关联 Github 账户

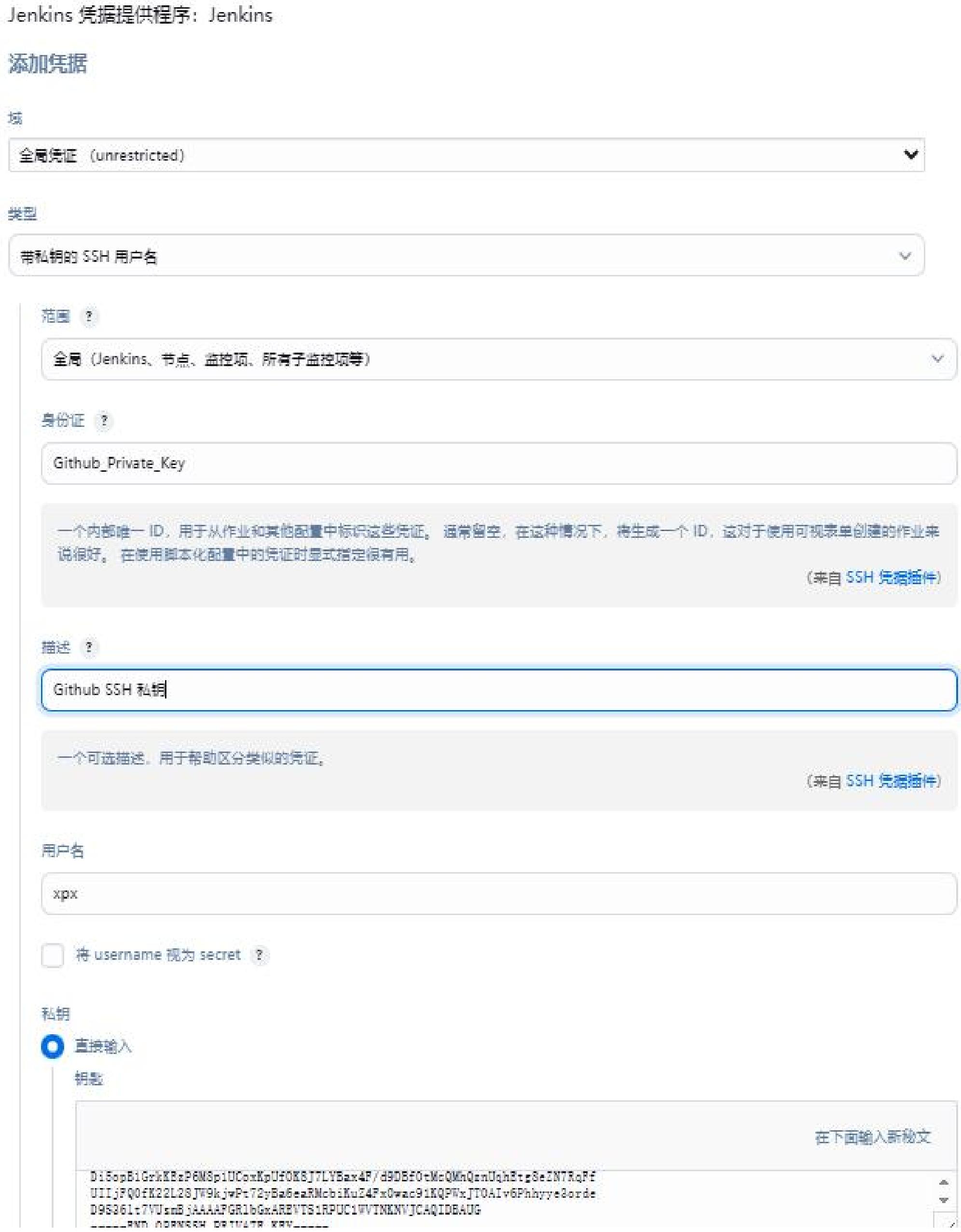


```
E:\software\Github_CLI>gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? SSH
? Upload your SSH public key to your GitHub account? C:\Users\dell\.ssh\id_rsa.pub
? Title for your SSH key: (GitHub CLI)

? Title for your SSH key: GitHub CLI
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: A0D6-4FA2
Press Enter to open https://github.com/login/device in your browser...
✓ Authentication complete.
- gh config set -h github.com git_protocol ssh
✓ Configured git protocol
✓ SSH key already existed on your GitHub account: C:\Users\dell\.ssh\id_rsa.pub
✓ Logged in as 2023120252
```

新建 item：



Jenkins 凭据提供程序：Jenkins

添加凭据

域

全局凭证 (unrestricted)

类型

带私钥的 SSH 用户名

范围 ?

全局 (Jenkins、节点、监控项、所有子监控项等)

身份证 ?

Github_Private_Key

一个内部唯一 ID，用于从作业和其他配置中标识这些凭证。 通常留空，在这种情况下，将生成一个 ID，这对于使用可视表单创建的作业来说很好。 在使用脚本化配置中的凭证时显式指定很有用。

（来自 SSH 凭据插件）

描述 ?

Github SSH 私钥

一个可选描述，用于帮助区分类似的凭证。

（来自 SSH 凭据插件）

用户名

xpx

□ 将 username 视为 secret ?

私钥

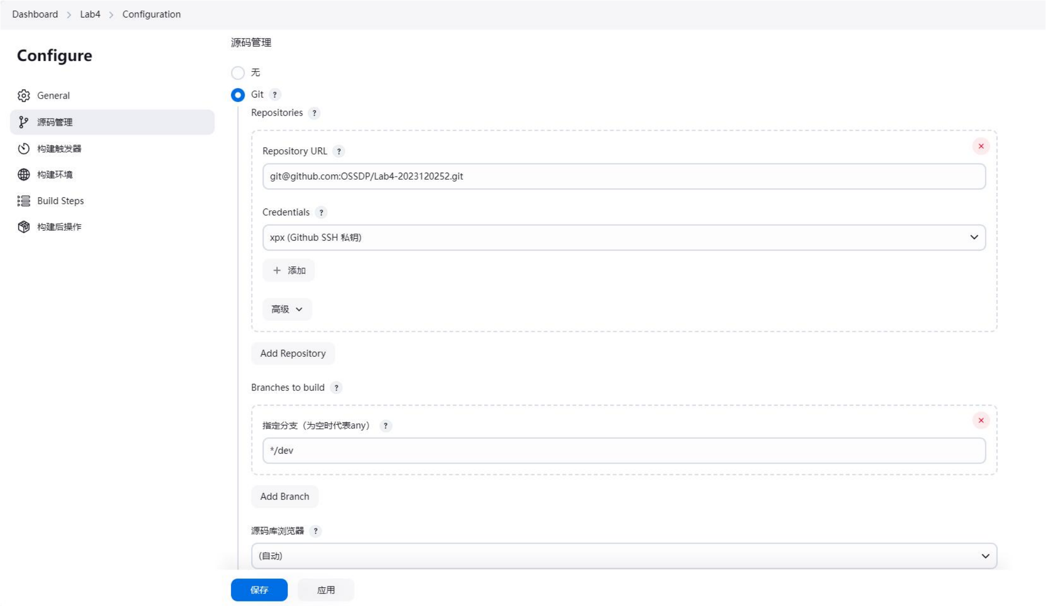
● 直接输入

钥匙

在下面输入新秘文

```
Di5opBiGrkKBzP6M3p1UCoxKpUfOK8J7LYBax4F/d9DBf0tMcQMnQznUqhEtgSeZN7RqFf
UIIjFQ0fK22L28JW9kjwPt72yBa6eaRMcbiKuZ4Fx0wac91KQPWxJTOAIv6Phhyye3orde
D9S361t7VUsmBjAAAAFGR1bGxAREVTS1RPUC1WVTNKNVJCAQIDBAUG
-----END OPENSSH PRIVATE KEY-----
```

源码设置：



源码管理

○ 无

● Git

Repositories

Repository URL: git@github.com:OSSDP/Lab4-2023120252.git

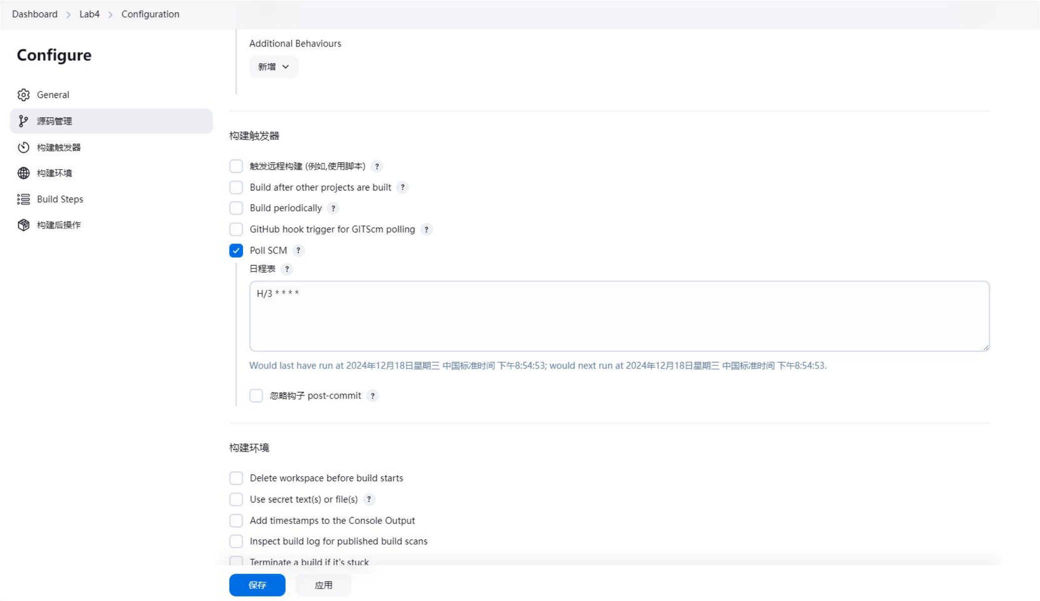
Credentials: xpx (Github SSH 私钥)

Branches to build

指定分支（为空时代表any）: */dev

源码库浏览器: (自动)

构建触发器设置：



构建触发器

□ 触发远程构建 (例如,使用脚本)

□ Build after other projects are built

□ Build periodically

□ GitHub hook trigger for GITScm polling

☑ Poll SCM

日程表

```
H/3 * * * *
```

Would last have run at 2024年12月18日星期三 中国标准时间 下午8:54:53; would next run at 2024年12月18日星期三 中国标准时间 下午8:54:53.

□ 忽略钩子 post-commit

构建环境

□ Delete workspace before build starts

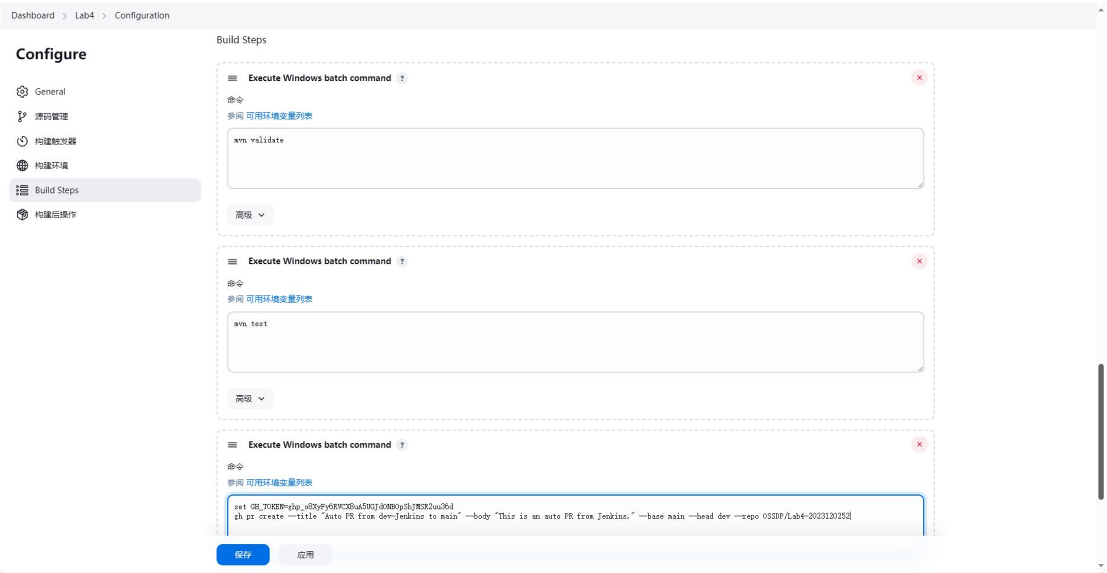
□ Use secret text(s) or file(s)

□ Add timestamps to the Console Output

□ Inspect build log for published build scans

□ Terminate a build if it's stuck

构建步骤：



Execute Windows batch command

命令

参阅 可用环境变量列表

```
mvn validate
```

Execute Windows batch command

命令

参阅 可用环境变量列表

```
mvn test
```

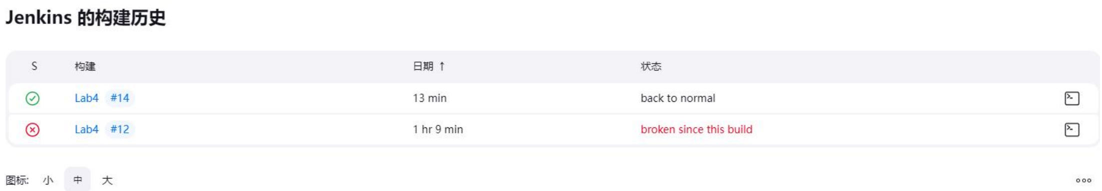
Execute Windows batch command

命令

参阅 可用环境变量列表

```
set GH_TOKEN=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
gh pr create —title ″Auto PR from dev-Jenkins to main″ —body ″This is an auto PR from Jenkins.″ —base main —head dev —repo OSSDP/Lab4-2023120252
```

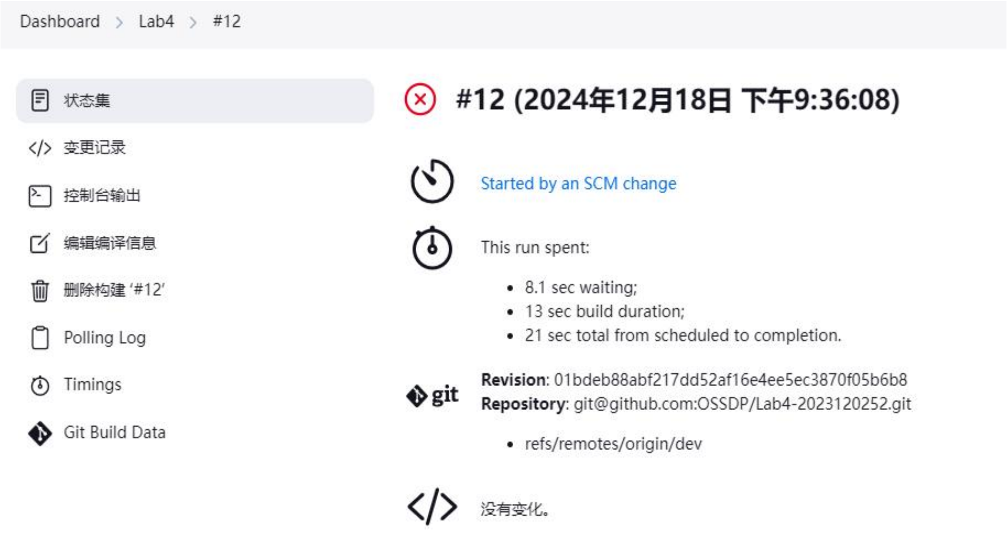
这里构建了两次，一次失败一次成功：



Jenkins 的构建历史

S	构建	日期 ↑	状态
✓	Lab4 #14	13 min	back to normal
✗	Lab4 #12	1 hr 9 min	broken since this build

构建失败截图：这是因为在前面的工作流部分将测试代码改成了错误代码。



#12 (2024年12月18日 下午9:36:08)

Started by an SCM change

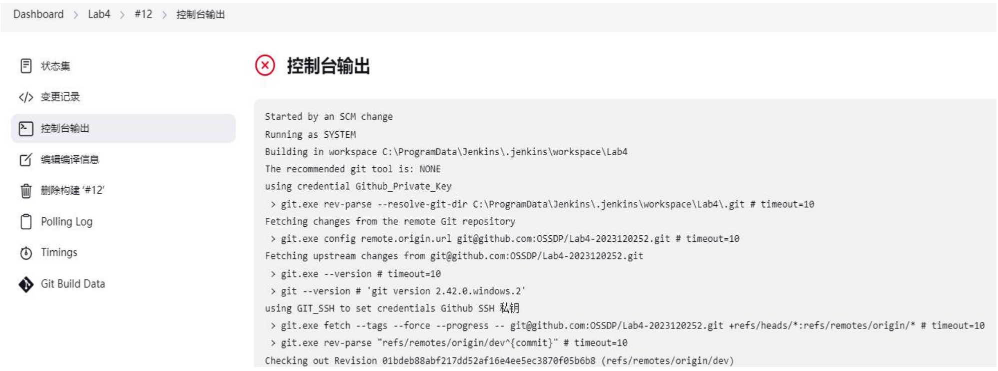
This run spent:

- 8.1 sec waiting;
- 13 sec build duration;
- 21 sec total from scheduled to completion.

Revision: 01bdeb88abf217dd52af16e4ee5ec3870f05b6b8
Repository: git@github.com:OSSDP/Lab4-2023120252.git

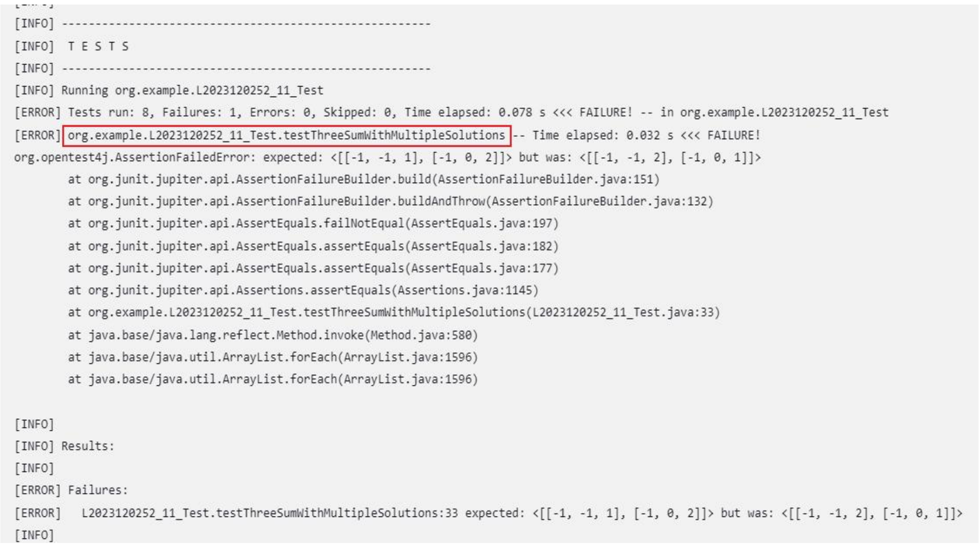
- refs/remotes/origin/dev

没有变化。



控制台输出

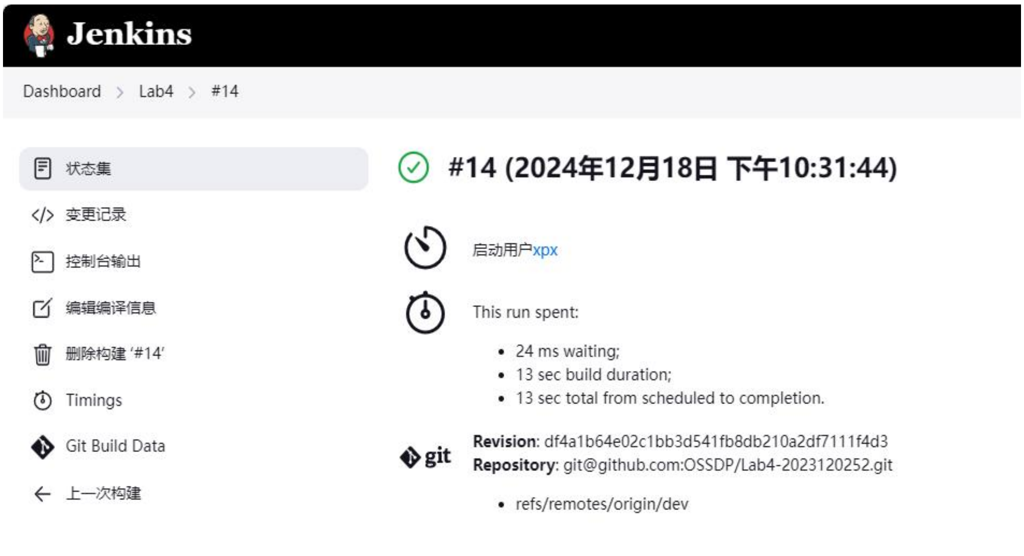
```
Started by an SCM change
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\Lab4
The recommended git tool is: NONE
using credential Github_Private_Key
 > git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\Lab4\.git # timeout=10
Fetching changes from the remote Git repository
 > git.exe config remote.origin.url git@github.com:OSSDP/Lab4-2023120252.git # timeout=10
Fetching upstream changes from git@github.com:OSSDP/Lab4-2023120252.git
 > git.exe --version # timeout=10
 > git --version # 'git version 2.42.0.windows.2'
using GIT_SSH to set credentials Github SSH 私钥
 > git.exe fetch --tags --force --progress -- git@github.com:OSSDP/Lab4-2023120252.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git.exe rev-parse "refs/remotes/origin/dev^{commit}" # timeout=10
Checking out Revision 01bdeb88abf217dd52af16e4ee5ec3870f05b6b8 (refs/remotes/origin/dev)
```



```
[INFO]
[INFO] -------------------------------------------------------
[INFO]  T E S T S
[INFO] -------------------------------------------------------
[INFO] Running org.example.L2023120252_11_Test
[ERROR] Tests run: 8, Failures: 1, Errors: 0, Skipped: 0, Time elapsed: 0.078 s <<< FAILURE! -- in org.example.L2023120252_11_Test
[ERROR] org.example.L2023120252_11_Test.testThreeSumWithMultipleSolutions -- Time elapsed: 0.032 s <<< FAILURE!
org.opentest4j.AssertionFailedError: expected: <[[-1, -1, 1], [-1, 0, 2]]> but was: <[[-1, -1, 2], [-1, 0, 1]]>
        at org.junit.jupiter.api.AssertionFailureBuilder.build(AssertionFailureBuilder.java:151)
        at org.junit.jupiter.api.AssertionFailureBuilder.buildAndThrow(AssertionFailureBuilder.java:132)
        at org.junit.jupiter.api.AssertEquals.failNotEqual(AssertEquals.java:197)
        at org.junit.jupiter.api.AssertEquals.assertEquals(AssertEquals.java:182)
        at org.junit.jupiter.api.AssertEquals.assertEquals(AssertEquals.java:177)
        at org.junit.jupiter.api.Assertions.assertEquals(Assertions.java:1145)
        at org.example.L2023120252_11_Test.testThreeSumWithMultipleSolutions(L2023120252_11_Test.java:33)
        at java.base/java.lang.reflect.Method.invoke(Method.java:580)
        at java.base/java.util.ArrayList.forEach(ArrayList.java:1596)
        at java.base/java.util.ArrayList.forEach(ArrayList.java:1596)

[INFO]
[INFO] Results:
[INFO]
[ERROR] Failures:
[ERROR]   L2023120252_11_Test.testThreeSumWithMultipleSolutions:33 expected: <[[-1, -1, 1], [-1, 0, 2]]> but was: <[[-1, -1, 2], [-1, 0, 1]]>
[INFO]
```

构建成功截图：将测试代码修改正确后的截图。



#14 (2024年12月18日 下午10:31:44)

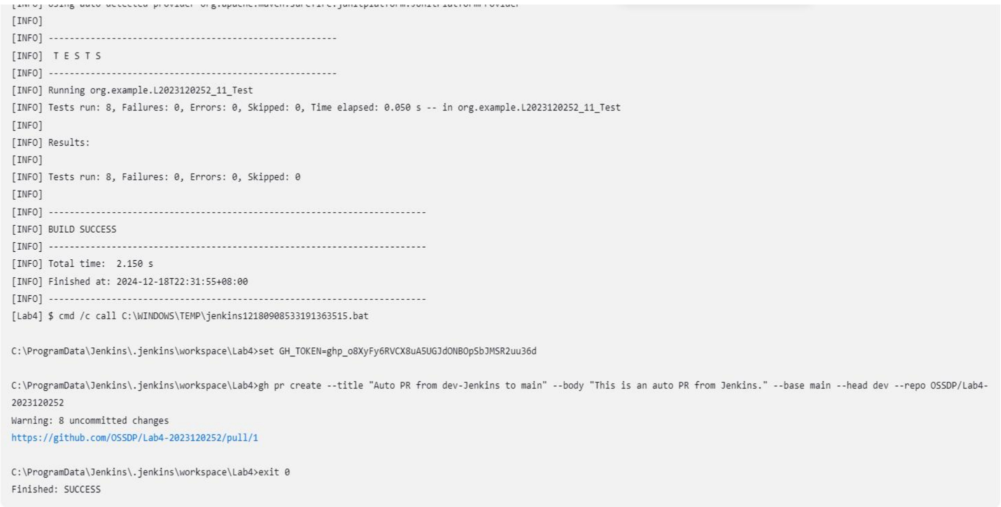
启动用户xpx

This run spent:

- 24 ms waiting;
- 13 sec build duration;
- 13 sec total from scheduled to completion.

Revision: df4a1b64e02c1bb3d541fb8db210a2df7111f4d3
Repository: git@github.com:OSSDP/Lab4-2023120252.git

- refs/remotes/origin/dev

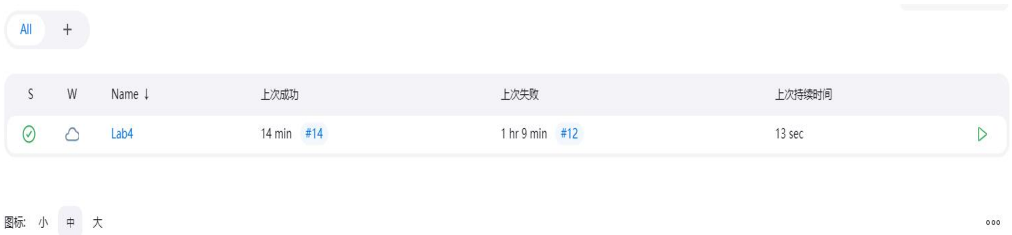


```
[INFO] Using auto detected provider org.apache.maven.surefire.junitplatform.JUnitPlatformProvider
[INFO]
[INFO] -------------------------------------------------------
[INFO]  T E S T S
[INFO] -------------------------------------------------------
[INFO] Running org.example.L2023120252_11_Test
[INFO] Tests run: 8, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.050 s -- in org.example.L2023120252_11_Test
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 8, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  2.150 s
[INFO] Finished at: 2024-12-18T22:31:55+08:00
[INFO] ------------------------------------------------------------------------
[Lab4] $ cmd /c call C:\WINDOWS\TEMP\jenkins12180908533191363515.bat

C:\ProgramData\Jenkins\.jenkins\workspace\Lab4>set GH_TOKEN=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

C:\ProgramData\Jenkins\.jenkins\workspace\Lab4>gh pr create --title "Auto PR from dev-Jenkins to main" --body "This is an auto PR from Jenkins." --base main --head dev --repo OSSDP/Lab4-
2023120252
Warning: 8 uncommitted changes
https://github.com/OSSDP/Lab4-2023120252/pull/1

C:\ProgramData\Jenkins\.jenkins\workspace\Lab4>exit 0
Finished: SUCCESS
```

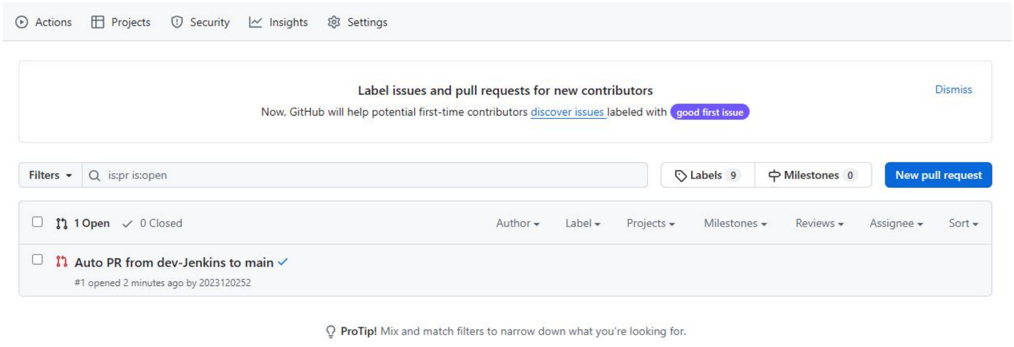


All +

S	W	Name ↓	上次成功	上次失败	上次持续时间	
✓	☁	Lab4	14 min #14	1 hr 9 min #12	13 sec	▷

图标: 小 中 大

Github 仓库的 PR 推送成功的截图：



Label issues and pull requests for new contributors

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

Dismiss

Filters ▾ is:pr is:open

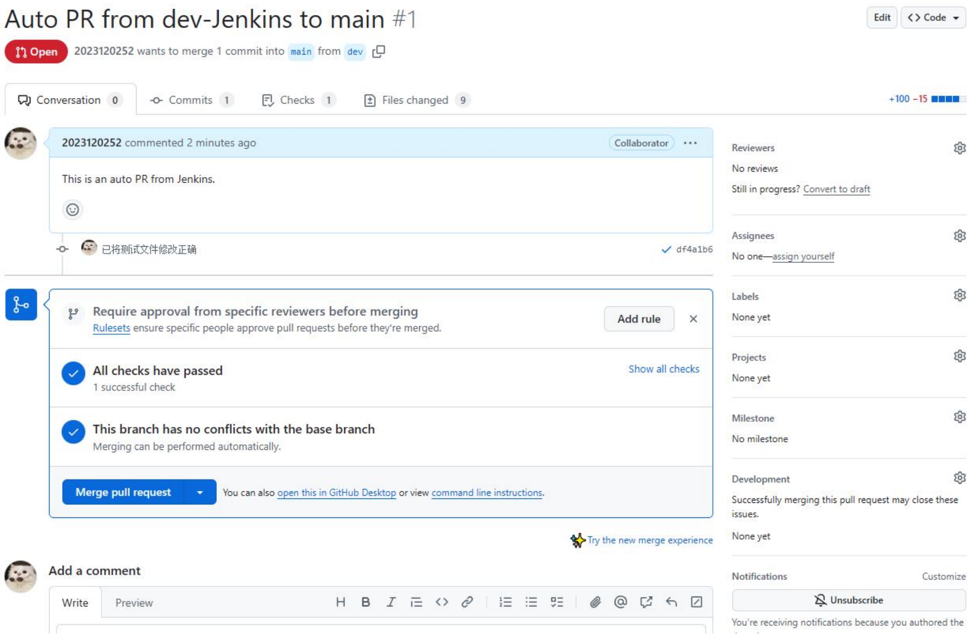
Labels 9 | Milestones 0 | New pull request

□ 1 Open ✓ 0 Closed — Author ▾ Label ▾ Projects ▾ Milestones ▾ Reviews ▾ Assignee ▾ Sort ▾

□ Auto PR from dev-Jenkins to main ✓

#1 opened 2 minutes ago by 2023120252

ProTip! Mix and match filters to narrow down what you're looking for.



Auto PR from dev-Jenkins to main #1

Open 2023120252 wants to merge 1 commit into main from dev

2023120252 commented 2 minutes ago — Collaborator

This is an auto PR from Jenkins.

已将测试文件修改正确 ✓ df4a1b6

Require approval from specific reviewers before merging

Rulesets ensure specific people approve pull requests before they're merged.

Add rule ×

All checks have passed

1 successful check

Show all checks

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.

Try the new merge experience

Add a comment

Write Preview

Reviewers: No reviews. Still in progress? Convert to draft

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

Milestone: No milestone

Development: Successfully merging this pull request may close these issues. None yet

Notifications: Customize — Unsubscribe

You're receiving notifications because you authored the

4 小结

本次实验让我对开源软件开发中的 DevOps 流程有了更深入的理解和实践。通过使用 GitHub Actions 和 Jenkins，我学习了如何自动化构建、测试和部署代码，并熟悉了相关的工具和配置。实验过程中，我遇到了一些挑战，例如配置 GitHub Actions 工作流、配置 Jenkins Git 凭据和构建触发器，以及确保 Maven 和 GitHub CLI 的正确安装和配置。通过克服这些挑战，我学到了很多，包括如何使用 GitHub Actions 和 Jenkins 进行自动化构建、测试和部署，如何配置 Git 凭据和构建触发器，以及如何编写构建脚本并使用 Maven 和 GitHub CLI 命令。总的来说，这次实验让我受益匪浅，我不仅学到了 DevOps 的理论知识，还通过实践掌握了相关的技能。