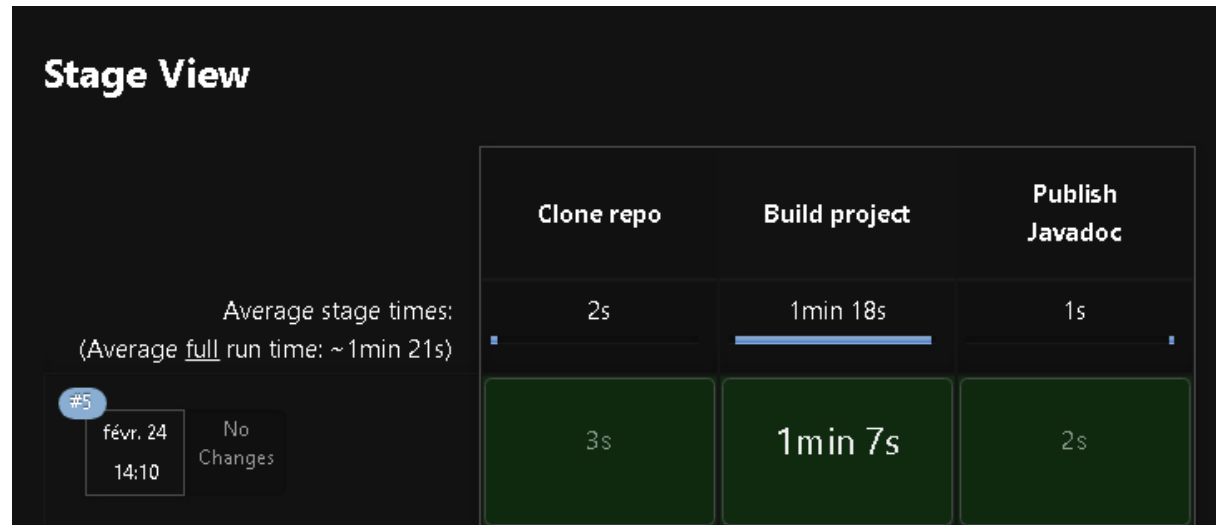
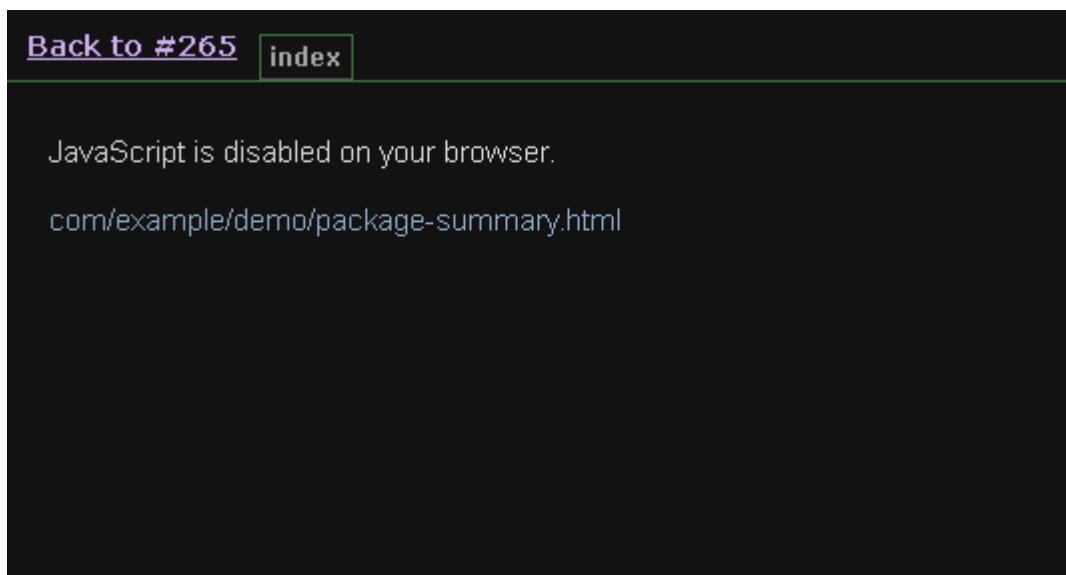
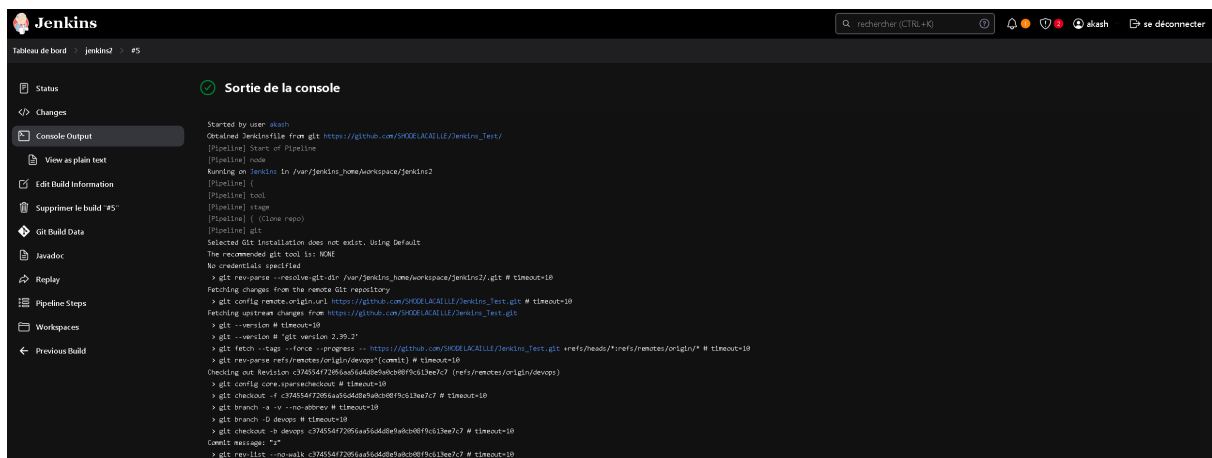


TP DevOps :

Compilation :



```
1 pipeline {
2   agent any
3
4   triggers {
5     pollSCM('*/*3 * * * *')
6   }
7
8   tools {
9     maven 'maven-3.5.2'
10  }
11
12  stages {
13    stage('Clone repo') {
14      steps {
15        git branch: 'devops', url: 'https://github.com/SHODELACAILLE/Jenkins_Test.git'
16      }
17    }
18
19    stage('Build project') {
20      steps {
21        dir('demo') {
22          sh "${tool 'maven-3.5.2'}/bin/mvn' -B -DskipTests clean package"
23
24          // Génère le Javadoc
25          sh "${tool 'maven-3.5.2'}/bin/mvn' javadoc:javadoc"
26        }
27      }
28    }
29
30    stage('Publish Javadoc') {
31      steps {
32        dir('demo/target/site/apidocs') {
33          publishHTML(target: [
34            reportDir: '.',
35            reportFiles: 'index.html',
36            reportName: "Javadoc",
37            keepAll: true,
38            allowMissing: false
39          ])
40        }
41      }
42    }
43  }
44 }
45
46
47
48
49
50
```



Maven :

Jenkinsfile DemoApplicationTests.java DemoApplication.java pom.xml

demo > src > test > java > com > example > demo > DemoApplicationTests.java

```
1 package com.example.demo;
2
3 import org.junit.jupiter.api.Test;
4 import static org.assertj.core.api.Assertions.assertThat;
5
6 public class DemoApplicationTests {
7
8     // Classe Calculatrice définie directement dans le fichier de test
9     static class Calculatrice {
10
11         public int add(int a, int b) {
12             return a + b;
13         }
14
15         public int subtract(int a, int b) {
16             return a - b;
17         }
18
19         public int multiply(int a, int b) {
20             return a * b;
21         }
22     }
23
24     @Test
25     public void testAddition() {
26         Calculatrice calculatrice = new Calculatrice();
27         assertThat(calculatrice.add(2, 3)).isEqualTo(5);
28     }
29
30     @Test
31     public void testSubtraction() {
32         Calculatrice calculatrice = new Calculatrice();
33         assertThat(calculatrice.subtract(5, 3)).isEqualTo(2);
34     }
35
36     @Test
37     public void testMultiplication() {
38         Calculatrice calculatrice = new Calculatrice();
39         assertThat(calculatrice.multiply(2, 3)).isEqualTo(6);
40     }
41 }
42
```

```

    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-test</artifactId>
      <scope>test</scope>
    </dependency>

    <dependency>
      <groupId>org.junit.jupiter</groupId>
      <artifactId>junit-jupiter-api</artifactId>
      <version>5.8.2</version>
      <scope>test</scope>
    </dependency>
    <dependency>
      <groupId>org.junit.jupiter</groupId>
      <artifactId>junit-jupiter-engine</artifactId>
      <version>5.8.2</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

  <build>
    <plugins>
      <plugin>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-maven-plugin</artifactId>
      </plugin>
    </plugins>
  </build>
</project>

```

```

pipeline {
    agent any

    tools {
        // Spécifier l'installation de Maven
        maven 'maven-3.5.2'
    }

    stages {
        stage('Build project') {
            steps {
                // Naviguer vers le répertoire du projet
                dir('demo') {
                    sh 'chmod +x mvnw'

                    sh './mvnw clean package'
                    sh './mvnw javadoc:javadoc'
                }
            }
        }
    }

    post {
        success {
            script {
                if (fileExists('**/target/surefire-reports/*.xml')) {
                    junit '**/target/surefire-reports/*.xml'
                }
            }
        }
    }
}

```

maven

Stage View

Average stage times:
(Average full run time: ~1min 32s)

#28

févr. 27
06:16

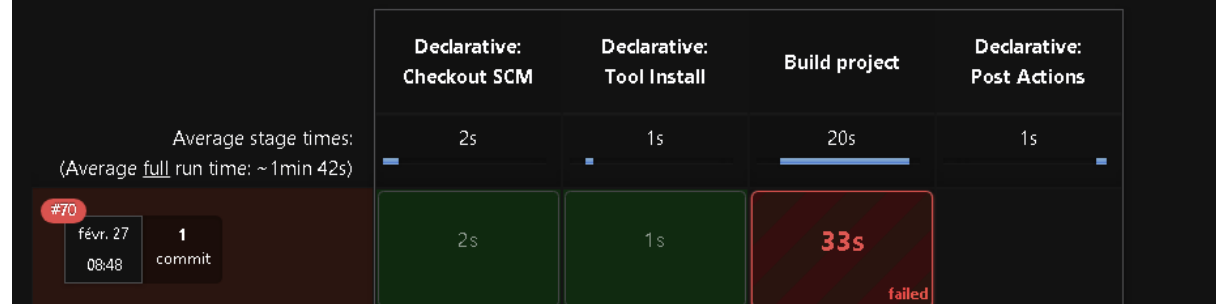
1
commit

Declarative: Checkout SCM	Declarative: Tool Install	Build project	Declarative: Post Actions
2s	1s	34s	1s
2s	1s	1min 21s	1s

Faire Échouer un Test :

```
@Test
public void testAddition() {
    Calculatrice calculatrice = new Calculatrice();
    assertThat(calculatrice.add(2, 3)).isEqualTo(6);
}
```

Stage View



```
expected: 6
but was: 5

at java.base/jdk.internal.reflect.NativeConstructorAccessorImpl.newInstance@((Native Method)
at java.base/jdk.internal.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:77)
at java.base/jdk.internal.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45)
at java.base/java.lang.reflect.Constructor.newInstanceWithCaller(Constructor.java:499)
at com.example.demo.DemoApplicationTests.testAddition(DemoApplicationTests.java:27)
at java.base/java.lang.reflect.Method.invoke(Method.java:568)
at java.base/java.util.ArrayList.forEach(ArrayList.java:1511)
at java.base/java.util.ArrayList.forEach(ArrayList.java:1511)

[INFO]
[INFO] Results:
[INFO]
[ERROR] Failures:
[ERROR]   DemoApplicationTests.testAddition:27
expected: 6
but was: 5
[INFO]
[ERROR] Tests run: 3, Failures: 1, Errors: 0, Skipped: 0
[INFO]
[INFO] BUILD FAILURE
[INFO]
[INFO] Total time: 23.400 s
[INFO] Finished at: 2024-02-27T07:49:25Z
[INFO]
[ERROR] Failed to execute goal org.apache.maven.plugins:maven-surefire-plugin:3.1.2:test (default-test) on project demo: There are test failures.
[ERROR]
[ERROR] Please refer to /var/jenkins_home/workspace/maven/demo/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://wiki.apache.org/confluence/display/MAVEN/MojofailureException
```

Ignorer un Test :

```

@Disabled
@Test
public void testMultiplication() {
    Calculatrice calculatrice = new Calculatrice();
    assertEquals(calculatrice.multiply(2, 3), 6);
}

```

```

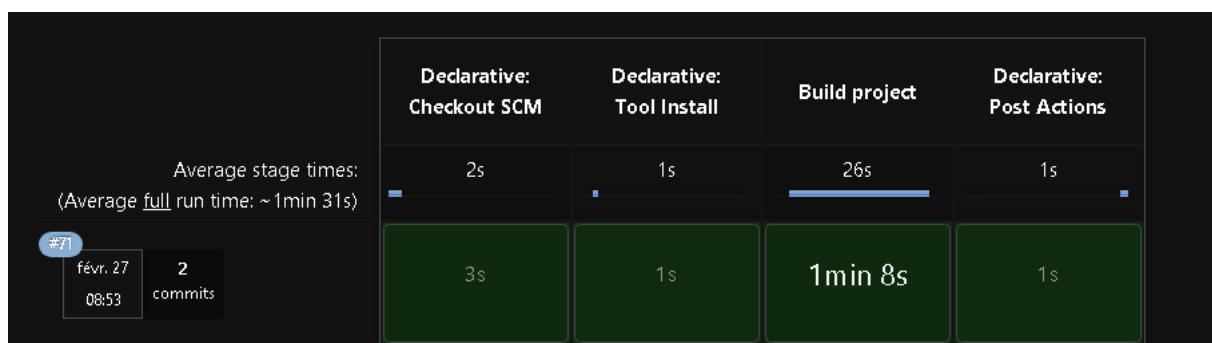
pipeline {
    agent any

    tools {
        // Spécifier l'installation de Maven
        maven 'maven-3.5.2'
    }

    stages {
        stage('Build project') {
            steps {
                // Naviguer vers le répertoire du projet
                dir('demo') {
                    sh 'chmod +x mvnw'

                    sh './mvnw -DskipTests clean package'
                    sh './mvnw javadoc:javadoc'
                }
            }
        }
    }
}

```



Sonar :

utilisation de ngrok pour <http://localhost:9000>.

Create Project

Search for projects...

Perspective

Overall Status

Sort by

Name

2 project(s)

demo PUBLIC

Passed

Last analysis: 3 minutes ago • 68 Lines of Code • XML, Java

E 1

A 0

A 4

A —

0.0%

0.0%

Security

Reliability

Maintainability

Hotspots Reviewed

Coverage

Duplications

sonarqube

ProjectsIssuesRulesQuality ProfilesQuality GatesAdministrationMore

Analysis_Testmain

OverviewIssuesSecurity HotspotsMeasuresCodeActivity

Project SettingsProject Information

Analysis Method

Use this page to manage and set-up the way your analyses are performed.

How do you want to analyze your repository?

With Jenkins

With GitHub Actions

With Azure Pipelines

Other CI

Locally

```
1 pipeline {
2   agent any
3
4   tools {
5     // Specify the installation of Maven
6     maven "maven-3.5.2" // Make sure this version is configured in your Jenkins tools
7   }
8
9   stages {
10    stage('Checkout SCM') {
11      steps {
12        checkout scm // This checks out source code from the configured SCM repository
13      }
14    }
15
16    stage('Build and Analyze') {
17      steps {
18        dir('demo') {
19          sh 'chmod +x mvnw'
20
21          withSonarQubeEnv('MySonar') {
22            sh './mvnw sonar:sonar -Dsonar.projectKey=com.example:demo -Dsonar.host.url=https://219f-2a01-e0a-1dc-a678-d4a9-cc43-bea9-888-ngrok-free.app -Dsonar.token=sq_2620b0d6078983b95ac6233ef3db78058617f3c'
23          }
24        }
25      }
26    }
27  }
28}
```

Sonar

Stage View

Average stage times:
(Average full run time: ~3min 10s)

#42

mars 03
04:18

1
commit

Declarative: Checkout SCM	Declarative: Tool Install	Checkout SCM	Build and Analyze
2s	1s	3s	1min 9s
2s	1s	3s	2min 57s

