

Gérer un projet collaboratif en intégrant une démarche CI/CD

Notice explicative de la pipeline

La pipeline contient 4 jobs :

- 1) test-backend, il permet :
 - a. d'exécuter les tests du back grâce à Maven
 - b. de générer le rapport de tests grâce à Jacoco
 - c. rendre accessible le rapport de tests au téléchargement
- Screenshot de la configuration :

```
5  test-backend:
6    name: Backend tests
7    runs-on: ubuntu-latest
8    steps:
9      - name: Check out repository code
10       uses: actions/checkout@v3
11      - name: Set up JDK
12       uses: actions/setup-java@v1
13       with:
14         java-version: '11'
15      - name: Cache Maven packages
16       uses: actions/cache@v1
17       with:
18         path: ~/.m2
19         key: ${ runner.os }-m2-${ hashFiles('**/pom.xml') }
20         restore-keys: ${ runner.os }-m2
21      - name: Run Tests
22       working-directory: ./back
23       run: mvn -B test
24      - name: Jacoco Report
25       uses: 'actions/upload-artifact@v3.1.0'
26       with:
27         name: jacoco-report
28         path: ${ github.workspace }/back/target/site/jacoco
```

- 2) test-frontend, il permet :
 - a. d'exécuter les tests du front
 - b. de générer le rapport de tests
 - c. rendre accessible le rapport de tests au téléchargement
- Screenshot de la configuration :

```
29  test-frontend:
30    name: Frontend tests
31    runs-on: ubuntu-latest
32    steps:
33      - name: Check out repository code
34       uses: actions/checkout@v3
35      - name: Setup Node.js
36       uses: actions/setup-node@v3.5.1
37      - name: Build app
38       working-directory: ./front
39       run: npm ci
40      - name: Run tests
41       working-directory: ./front
42       run: npm run test-coverage
43      - name: Coverage Report
44       uses: 'actions/upload-artifact@v3.1.0'
45       with:
46         name: frontend-report
47         path: ${ github.workspace }/front/coverage/bobapp
```

- 3) code-quality, il permet :
- a. d'exécuter le scan sonar
 - b. le résultat est accessible sur le tableau de board Sonar Cloud
- Screenshot de la configuration :

```
48   code-quality:
49     name: Code Quality Check
50     runs-on: ubuntu-latest
51     steps:
52       - name: Check out repository code
53         uses: actions/checkout@v3
54         with:
55           fetch-depth: 0
56       - name: Set up JDK
57         uses: actions/setup-java@v1
58         with:
59           java-version: '11'
60       - name: Cache Maven packages
61         uses: actions/cache@v1
62         with:
63           path: ~/.m2
64           key: ${ runner.os }-m2-${ hashFiles('**/pom.xml') }
65           restore-keys: ${ runner.os }-m2
66       - name: Build Java binaries
67         working-directory: ./back
68         run: mvn package
69       - name: Setup Node.js
70         uses: actions/setup-node@v3.5.1
71       - name: Build app
72         working-directory: ./front
73         run: npm ci
74       - name: Run tests
75         working-directory: ./front
76         run: npm run test-coverage
77       - name: SonarCloud Scan
78         uses: SonarSource/sonarcloud-github-action@master
79         env:
80           GITHUB_TOKEN: ${ secrets.GITHUB_TOKEN }
81           SONAR_TOKEN: ${ secrets.SONAR_TOKEN }
```

- 4) builder-docker-images, il permet
- a. de construire l'image pour le back
 - b. de construire l'image pour le frontend
- Screenshot de la configuration :

```
82   build-docker-images:
83     name: Build docker images
84     runs-on: ubuntu-latest
85     steps:
86       - name: Check out repository code
87         uses: actions/checkout@v3
88       - name: Set up Docker Buildx
89         uses: docker/setup-buildx-action@v2
90       - name: Build backend
91         uses: docker/build-push-action@v3.2.0
92         with:
93           context: ./back
94           file: ./back/Dockerfile
95           tags: bobapp-backend:latest
96       - name: Build frontend
97         uses: docker/build-push-action@v3.2.0
98         with:
99           context: ./front
100          file: ./front/Dockerfile
101          tags: bobapp-frontend:latest
```