

CI/CD avec GitHub Actions

 Yoan Ancelly  Lyazid Bahajjoub  Jamil Abdel Hamid

last commit **last tuesday**

 Automatisation de l'intégration continue **passing**

Description du projet

Sommaire

- Automatisation de l'intégration continue de l'API
 - Les étapes d'intégration continue de l'API
 - 1) Automatisation des tests unitaires
 - 2) Automatisation des tests d'intégration
 - 3) Automatisation des tests d'intégration
- 4) Mise en cache des dépendances
- Références

Automatisation de l'intégration continue de l'API

Les étapes d'intégration continue de l'API

- Installation des dépendances : `yarn install`
- Exécution des tests unitaires : `yarn test`
- Exécution des tests d'intégration : `yarn test:e2e`
- Build de l'application : `yarn build`
- Construction de l'image Docker.
- Partage de l'image Docker sur le DockerHub.

1) Automatisation des tests unitaires

```
name: Automatisation de l'intégration continue
on: [push]
jobs:
  build-and-test:
    name: Run unit tests
    runs-on: ubuntu-latest
    defaults:
      run:
        working-directory: ./api
    steps:
      - uses: actions/checkout@v3
      - uses: actions/setup-node@v3
      - name: Install dependencies
```

```
run: yarn install
- name: Run unit tests
  run: yarn test
```

- **name** : Nom du workflow.
- **on** : Déclencheur du workflow.
- **jobs** : Liste des jobs.
- **build-and-test** : Nom du job.
- **name** : Nom du job.
- **runs-on** : Plateforme d'exécution du job.
- **defaults** : Configuration par défaut du job.
- **run** : Configuration par défaut des étapes du job.
- **working-directory** : Répertoire de travail par défaut des étapes du job.
- **steps** : Liste des étapes du job.
- **uses** : Action à exécuter.
- **actions/checkout@v3** : Action permettant de récupérer le code source.
- **actions/setup-node@v3** : Action permettant d'installer Node.js.
- **name** : Nom de l'étape.
- **run** : Commande à exécuter.
- **yarn install** : Installation des dépendances.
- **yarn test** : Exécution des tests unitaires.

2) Automatisation des tests d'intégration

```
push-to-docker:
  name: Push on Docker Hub
  runs-on: ubuntu-latest
  defaults:
    run:
      working-directory: ./api
  steps:
    - name: Check out the repo
      uses: actions/checkout@v3

    - name: Install dependencies
      run: yarn install

    - name: Build app
      run: yarn build

    - name: Log in to Docker Hub
      uses: docker/login-action@v2
      with:
        username: ${ secrets.TP3_DOCKERHUB_USERNAME }
        password: ${ secrets.TP3_DOCKERHUB_SECRET }

    - name: Setup Docker Buildx
      uses: docker/setup-buildx-action@v2
```

```
- name: Build and push
  uses: docker/build-push-action@v3
  with:
    context: ./api
    push: true
    tags: yoanc/tp3-github-actions:${{ github.sha }}
```

node_modules not found > Install dependencies

Error buildx > Setup buildx

Error Dockerfile not found > Add context

3) Automatisation des tests d'intégration

```
build-and-test-e2e:
  name: Run integration tests
  runs-on: ubuntu-latest
  defaults:
    run:
      working-directory: ./api
  services:
    mongo:
      image: mongo
      ports:
        - 27017:27017
      env:
        MONGO_INITDB_ROOT_USERNAME: ${MONGODB_USERNAME}
        MONGO_INITDB_ROOT_PASSWORD: ${MONGODB_PASSWORD}
  steps:
    - name: Check out the repo
      uses: actions/checkout@v3

    - uses: actions/setup-node@v3

    - name: Install dependencies
      run: yarn install

    - name: Run integration Tests
      run: yarn test:e2e
      env:
        MONGODB_URI:
        mongodb://${MONGODB_USERNAME}:${MONGODB_PASSWORD}@localhost:27017
```

Running jobs directly on runner machine > use localhost to access service container > Must maps ports to enable access to service container

Add **needs** to **push-to-docker** job > prevent pushing to DockerHub if tests fail

4) Mise en cache des dépendances

```
- name: Setup NodeJS
  uses: actions/setup-node@v3
  with:
    cache: 'yarn'
    cache-dependency-path: '**/yarn.lock'

- name: Install dependencies
  run: yarn install --immutable
```

yarn cache not found > cache-dependency-path > it check in root folder by default

Add --immutable

Références

- [Workflow syntax for GitHub Actions - GitHub Docs](#) > needs > wait for other jobs to finish