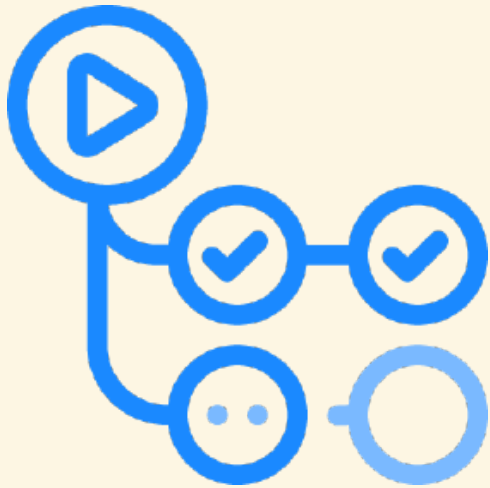


# GITHUB ACTIONS



# INTRODUCTION

# DÉFINITION DE GITHUB ACTIONS

- Service d'automatisation intégré directement dans GitHub, permettant d'automatiser diverses tâches de développement, de test et de déploiement directement à partir de votre référentiel GitHub.

# OBJECTIFS ET AVANTAGES

- Conçues pour simplifier et automatiser les workflows de développement logiciel.
- Leur intégration directe avec GitHub offre plusieurs avantages, notamment une automatisation transparente, une intégration continue (CI) et un déploiement continu (CD) faciles, ainsi qu'une grande flexibilité pour créer des workflows personnalisés.

# STRUCTURE DE BASE

# WORKFLOW YAML

- Un workflow GitHub est défini dans un fichier YAML situé dans le répertoire `.github/workflows` de votre dépôt.
- Ce fichier spécifie les actions à exécuter, les déclencheurs pour démarrer le workflow et d'autres configurations.

# ÉVÉNEMENTS DÉCLENCHEURS (TRIGGERS)

- Les GitHub Actions sont déclenchées par des événements tels que les pushes de code, les pull requests, les créations de branches, les labels, etc.
- Vous pouvez définir des workflows spécifiques pour réagir à ces événements.

# JOBS ET ÉTAPES (STEPS)

- Un workflow GitHub est composé de un ou plusieurs **jobs**, qui sont des séquences d'étapes (**steps**).
- Chaque **steps** représente une action à exécuter, comme la construction de code, les tests unitaires, le déploiement, etc.
- Les étapes peuvent être exécutées en parallèle ou en séquence, selon les besoins du workflow.



**FONCTIONNEMENT**

# DÉCLENCHEMENT DES ACTIONS

- Les actions GitHub sont déclenchées automatiquement en réponse à des événements spécifiques définis dans le workflow.
- Par exemple, un push de code vers une branche déclenchera un workflow de construction et de test.

# EXÉCUTION DES WORKFLOWS

- Une fois déclenchées, les GitHub Actions exécutent les workflows en fonction des configurations spécifiées dans le fichier YAML.
- Chaque étape est exécutée dans un environnement d'exécution isolé, ce qui garantit la cohérence et la reproductibilité des résultats.

# GESTION DES RÉSULTATS

- À la fin de l'exécution, les GitHub Actions fournissent des informations détaillées sur le résultat de chaque étape du workflow.
- Vous pouvez consulter les journaux d'exécution pour identifier les erreurs et les problèmes éventuels.

# PERSONNALISATION ET CONFIGURATION AVANCÉE

- Les GitHub Actions offrent de nombreuses options de personnalisation et de configuration avancée, telles que la définition de variables d'environnement, l'utilisation de conditionnels, l'intégration avec des outils externes, etc.

# **INTÉGRATION AVEC LES WORKFLOWS DE DÉVELOPPEMENT**

# INTÉGRATION CONTINUE (CI)

- GitHub Actions peut être utilisée pour mettre en œuvre des pipelines d'intégration continue (CI), automatisant ainsi les processus de construction, de test et de validation de code à chaque modification du référentiel.

# DÉPLOIEMENT AUTOMATIQUE (CD)

- Elles peuvent également être utilisées pour mettre en œuvre des pipelines de déploiement continu (CD), permettant ainsi d'automatiser le déploiement et la mise à jour d'applications sur divers environnements, tels que les environnements de test, de pré-production et de production.



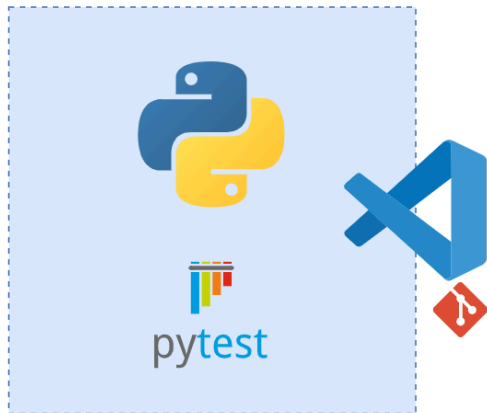
# AUTOMATISATION DES TÂCHES RÉCURRENTES

- Les GitHub Actions peuvent automatiser une grande variété de tâches récurrentes, telles que la génération de documentation, la notification des développeurs, la mise à jour des versions, la gestion des tickets, etc.

# TD - CRÉATION DE NOS GITHUB ACTIONS

[par ici](#)

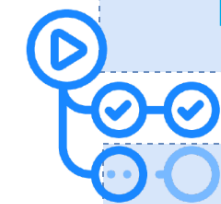
Dépôt de code local



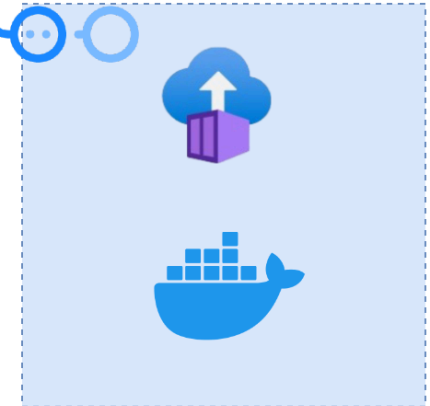
Chaine de Github Actions



Docker Engine



L'application est testée.  
Puis si les tests réussissent,  
une autre action build l'image,  
et la pousse sur Dockerhub



DockerHub

**BONNES PRATIQUES**

# ORGANISATION ET MODULARITÉ DES WORKFLOWS

- Il est recommandé d'organiser les workflows en modules réutilisables et de les maintenir séparément pour faciliter la gestion et la maintenance à long terme.

# GESTION DES SECRETS ET DES VARIABLES D'ENVIRONNEMENT

- Il est important de gérer les secrets et les informations sensibles de manière sécurisée en utilisant les fonctionnalités intégrées de gestion des secrets et des variables d'environnement dans GitHub Actions.

# TESTS ET VALIDATION DES WORKFLOWS

- Il est crucial de tester et de valider les workflows GitHub Actions régulièrement pour garantir leur bon fonctionnement et leur efficacité, en particulier lorsque des modifications sont apportées au code ou aux configurations.

# EN RÉSUMÉ

- Les GitHub Actions offrent une solution puissante et flexible pour automatiser les workflows de développement, d'intégration continue et de déploiement continu, directement à partir de votre dépôt GitHub.



# LA SUITE

Manifest Kubernetes

