

# Jenkins

Mettre en place la CI/CD





## Compétence demandée : Comprendre le principe de pipelines



- 1. Nous sommes DevOps
- 2. Cl
- 3. CD
- 4. Installation Jenkins
- 5. Configuration Jenkins
- 6. Jobs freestyle
- 7. Le Jenkinsfile



# Nous sommes DevOps



Le but des DevOps est d'accélérer la livraison de logiciels de qualité en automatisant et en optimisant les processus de développement, de test, de déploiement et de gestion des infrastructures.









- 1. Nous sommes DevOps
- 2. Cl
- 3. CD
- 4. Installation Jenkins
- 5. Configuration Jenkins
- 6. Jobs freestyle
- 7. Le Jenkinsfile



## CI



L'intégration continue (CI) désigne la pratique qui consiste à automatiser l'intégration des changements de code réalisés par plusieurs contributeurs dans un seul et même projet de développement.





## Que serait le monde sans CI?





Sans elle, les développeurs doivent se coordonner manuellement et communiquer lorsqu'ils contribuent au code du produit final. Cette coordination s'étend au-delà des équipes de développement et touche aussi les opérations et le reste de l'organisation.

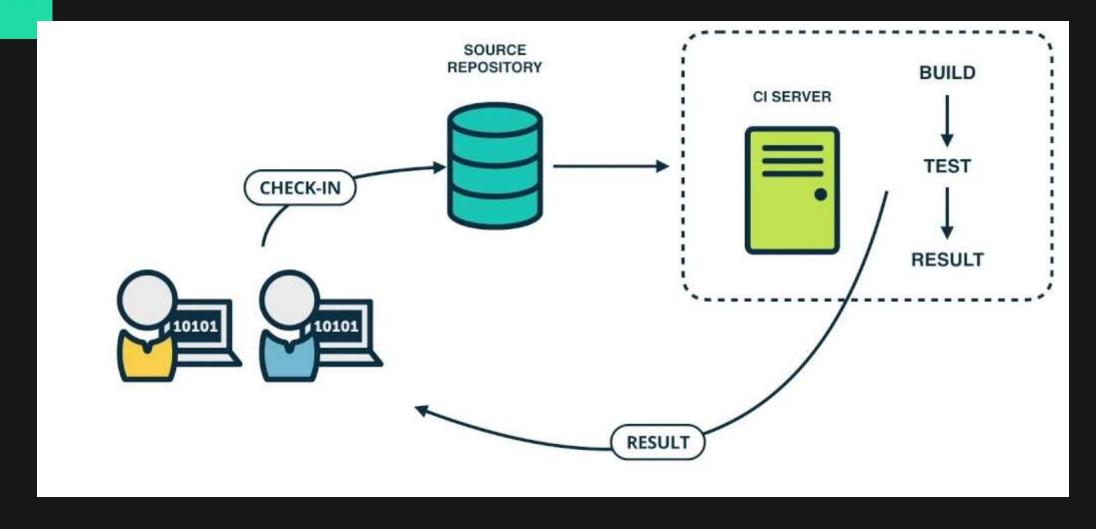


Dans un environnement sans Cl, la communication peut devenir une tâche de synchronisation complexe et intriquée, ce qui ajoute des coûts d'administration inutiles aux projets.

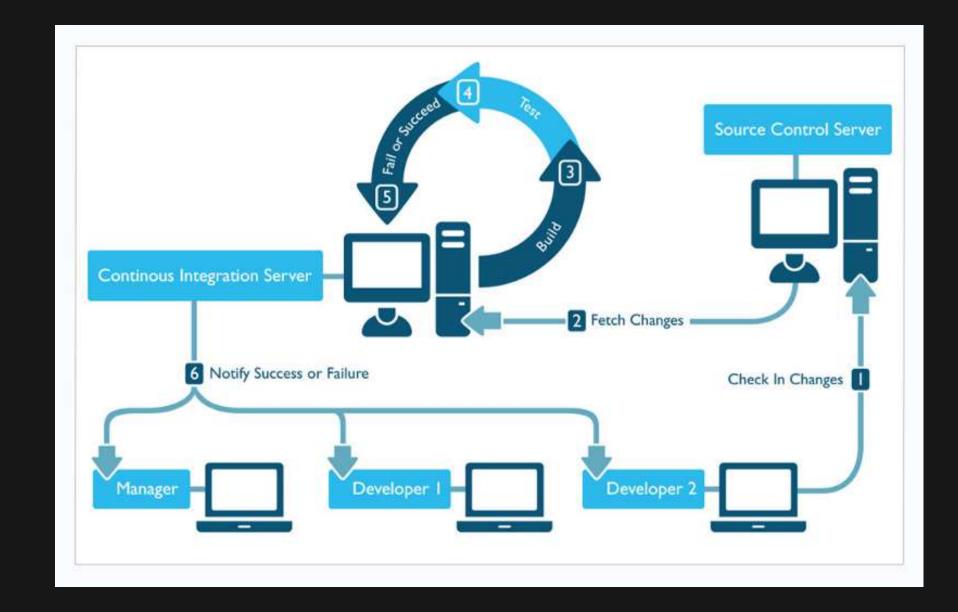
Les livraisons de code sont ralenties et les taux de défaillance augmentent, car les développeurs doivent gérer les intégrations de façon sensible et réfléchie.



Il s'agit d'une bonne pratique DevOps principale, permettant aux développeurs de logiciels de merger fréquemment des changements de code dans un dépôt central où les builds et les tests s'exécutent ensuite.



#### CERTIF ACADEMY





Des outils automatisés sont utilisés pour affirmer l'exactitude du nouveau code avant son intégration.

Les outils de CI se trouvent sur un autre serveur qui acte comme un agent de vérification et d'intégration.









- 1. Nous sommes DevOps
- 2. Cl
- 3. CD
- 4. Installation Jenkins
- 5. Configuration Jenkins
- 6. Jobs freestyle
- 7. Le Jenkinsfile



# CD



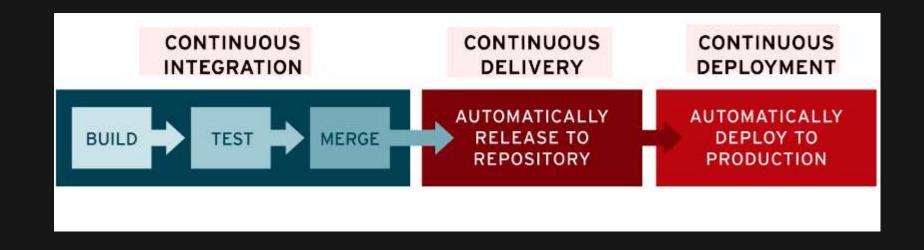
La livraison continue (en anglais : continuous delivery, CD) est une approche d'ingénierie logicielle dans laquelle les équipes produisent des logiciels dans des cycles courts, ce qui permet de le mettre à disposition à n'importe quel moment. Le but est de construire, tester et diffuser un logiciel plus rapidement.



Attention Continuous Delivery et Continuous Deployment n'est pas la même chose!

Continuous Deployment = Continuous Delivery + Automatic Deployment













- 1. Nous sommes DevOps
- 2. Cl
- 3. CD
- 4. Installation Jenkins
- 5. Configuration Jenkins
- 6. Jobs freestyle
- 7. Le Jenkinsfile



# **Installation Jenkins**











## Qu'est-ce que Jenkins?





Jenkins est un outil open source de serveur d'automatisation. Il aide à automatiser les parties du développement logiciel liées au build, aux tests et au déploiement, et facilite l'intégration continue et la livraison continue.



Jenkins est l'outil de CI le plus utilisé sur le marché avec 58%!



- 1. INSTALLATION & INTERFACE
- 2. BUILD JOBS
- 3. LES PLUGINS
- 4. ARCHETYPES DE PIPELINE
- 5. CI BEST PRACTICES



### INSTALLATION & INTERFACE



https://www.jenkins.io/



```
PS C:\Users\sam> java --version
java 17.0.1 2021-10-19 LTS
Java(TM) SE Runtime Environment (build 17.0.1+12-LTS-39)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.1+12-LTS-39, mixed mode, sharing)
PS C:\Users\sam>
```





## Quelle version de Java est requise ?





## https://www.jenkins.io/doc/ad ministration/requirements/java/

Java 11 ou Java 17

CERTIF

**២** Download Jenkins 2.361.3 LTS for: Download Jenkins 2.376 for: Generic Java package (.war) Generic Java package (.war) SHA-256: f39cb8d09fd17c72dc096511ce50f245fc3004d1022aaaf60421a536f740c9b9 SHA-256: 98530ed12eaf0849098a3e3dd0b745292bb8b9fb2f9a102be5884b04e87dc106 Docker Docker Ubuntu/Debian Ubuntu/Debian CentOS/Fedora/Red Hat CentOS/Fedora/Red Hat Windows Windows openSUSE openSUSE FreeBSD 🛍 Arch Linux 🛍 Gentoo 🛍 FreeBSD 🛍 macOS 🛍 Gentoo 🛍 macOS 🛍 OpenBSD 🛍





Vous allez télécharger un fichier d'installation (ici un msi pour Windows), d'une taille moyenne de 100 Mo

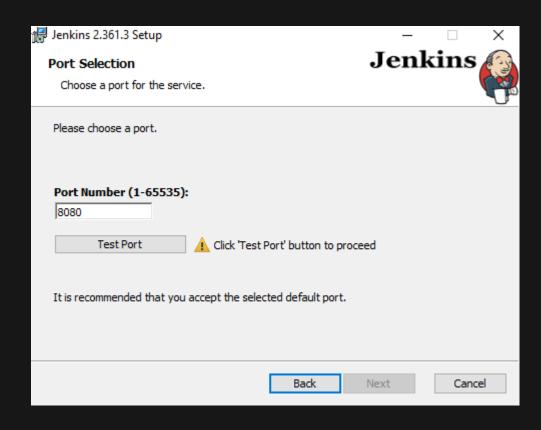




Un service permet à jenkins de tourner en arrière-plan, on peut choisir local system pour l'instant. Cela dit, cela veut dire qu'il aurait plein droit sur le système.

Sur chaque système, il reste important de savoir gérer ses utilisateurs, ses groupes et leurs droits d'accès!





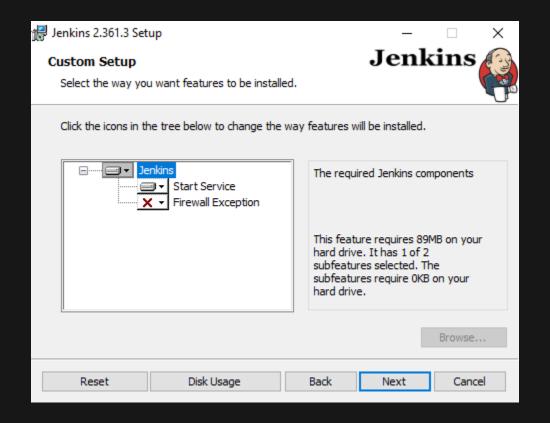
Comme tout serveur, jenkins demande un port libre sur lequel écoute les requetes.





Si vous avez plusieurs JRE sur votre ordinateur, choisissez la bonne version de Java!





Configurez le firewall si vous en avez un <u>plus tard</u>.

Habituellement, on configure les firewalls manuellement.

Tout le monde sait ce qu'est un firewall ?





## Qu'est-ce qu'un firewall?





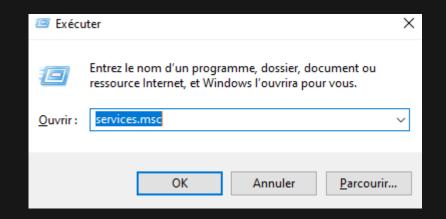


L'installation débute ...



C'est fini!





Vous pouvez gérer les services sur le serveur! ©

Savez-vous tous comment gérer les services ?

#### CERTIF ACADEMY

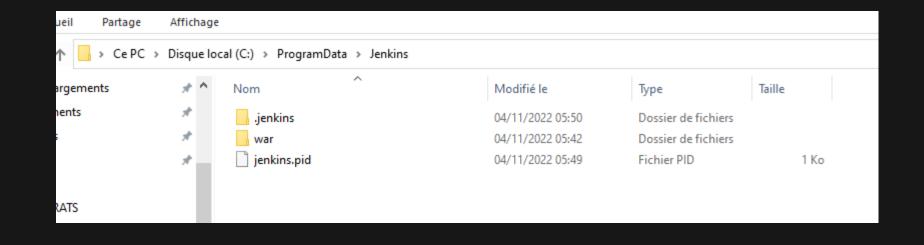
Services (local)					
kins	Nom	Description	État	Type de démarrage	Ouvrir une session en tant que
<u>tter</u> le service <u>émarrer</u> le service	Interface de services d'invit	Fournit une		Manuel (Déclenche	Système local
	Interruption SNMP	Reçoit les m		Manuel	Service local
	🖳 Isolation de clé CNG	Le service d'	En co	Manuel (Déclenche	Système local
cription : cins Automation Server	Jenkins Jenkins	Jenkins Aut	En co	Automatique	Système local
	🖳 Jeu sauvegardé sur Xbox Live	Ce service s		Manuel (Déclenche	Système local
	🖳 Journal d'événements Wind	Ce service g	En co	Automatique	Service local
	🖳 Journaux & alertes de perfo	Le service d		Manuel	Service local



Nom	Modifié le	Туре	Taille	
jenkins.err.log	23/11/2021 11:57	Document texte	396 Ko	
📧 jenkins.exe	02/11/2022 07:32	Application	606 Ko	
🛍 jenkins.exe.config	02/11/2022 09:56	XML Configuratio	1 Ko	
jenkins.out.log	23/11/2021 11:57	Document texte	1 Ko	
] Jenkins.war	02/11/2022 09:41	Fichier WAR	91 311 Ko	
jenkins.wrapper.log	04/11/2022 05:42	Document texte	24 Ko	
jenkins.xml	04/11/2022 05:42	Document XML	3 Ko	
jenkins.xml.backup	18/11/2021 03:17	Fichier BACKUP	3 Ko	

L'installation débute ...

#### CERTIF ACADEMY



#### CERTIF ACADEMY

→ Ce PC → Disque local	(C:) > ProgramData > Jenkins > .jenkins			
nents * ^	Nom	Modifié le	Туре	Taille
オ	jobs	18/11/2021 03:51	Dossier de fichiers	
7th	logs	18/11/2021 03:18	Dossier de fichiers	
*	nodes	18/11/2021 03:18	Dossier de fichiers	
	plugins	04/11/2022 05:42	Dossier de fichiers	
	secrets	22/11/2021 13:52	Dossier de fichiers	
	updates	22/11/2021 07:06	Dossier de fichiers	
	userContent	18/11/2021 03:18	Dossier de fichiers	
	users	22/11/2021 20:31	Dossier de fichiers	
	workflow-libs	18/11/2021 03:20	Dossier de fichiers	
	workspace	21/11/2021 23:09	Dossier de fichiers	
NATIS GROUP	.lastStarted	22/11/2021 19:09	Fichier LASTSTART	0 Ko
onversation Microsof	owner	23/11/2021 01:01	Fichier OWNER	1 Ko
	config.xml	23/11/2021 11:57	Document XML	6 Ko
UP	ailed-boot-attempts.txt	04/11/2022 05:50	Document texte	1 Ko
	hudson.model.UpdateCenter.xml	04/11/2022 05:50	Document XML	1 Ko
(ERROUCHA - Docui	hudson.plugins.git.GitTool.xml	18/11/2021 03:20	Document XML	1 Ko
ndation - Documen	identity.key.enc	18/11/2021 03:18	Fichier ENC	2 Ko
	jenkins.install.InstallUtil.lastExecVersion	22/11/2021 19:09	Fichier LASTEXEC	1 Ko
	jenkins.install.UpgradeWizard.state	18/11/2021 03:35	Fichier STATE	1 Ko
	jenkins.model.JenkinsLocationConfigura	18/11/2021 03:35	Document XML	1 Ko
	$\begin{tabular}{ll} \hline & jenkins.security.apitoken.ApiTokenProper \\ \hline \end{tabular}$	22/11/2021 20:34	Document XML	1 Ko
	jenkins.security.QueueltemAuthenticator	22/11/2021 20:34	Document XML	1 Ko
	jenkins.security.UpdateSiteWarningsConf	22/11/2021 20:34	Document XML	1 Ko
	jenkins.telemetry.Correlator.xml	18/11/2021 03:18	Document XML	1 Ko
ents	nodeMonitors.xml	22/11/2021 19:09	Document XML	1 Ko
	org.jenkinsci.plugins.resourcedisposer.As	22/11/2021 16:45	Document XML	1 Ko
(C:)	org.jenkinsci.plugins.workflow.flow.Flow	22/11/2021 14:05	Document XML	1 Ko
(/	queue.xml	04/11/2022 05:50	Document XML	1 Ko
RW (F:)	queue.xml.bak	22/11/2021 16:47	Fichier BAK	1 Ko
	secret.key	18/11/2021 03:18	Fichier KEY	1 Ko
	secret.key.not-so-secret	18/11/2021 03:18	Fichier NOT-SO-S	0 Ko
_				



Il est important de retenir les 2 dossiers : jobs et workspace.

Jobs : contient toutes les informations sur les jobs (dont le build) géré de manière interne par Jenkins

Workspace: peut être n'importe où et sert d'espace de travail lors des builds par exemple. La gestion des workspace se repose sur le fait qu'un workspace peut être supprimé à tout moment. Il se reconstruira au lancement du job.







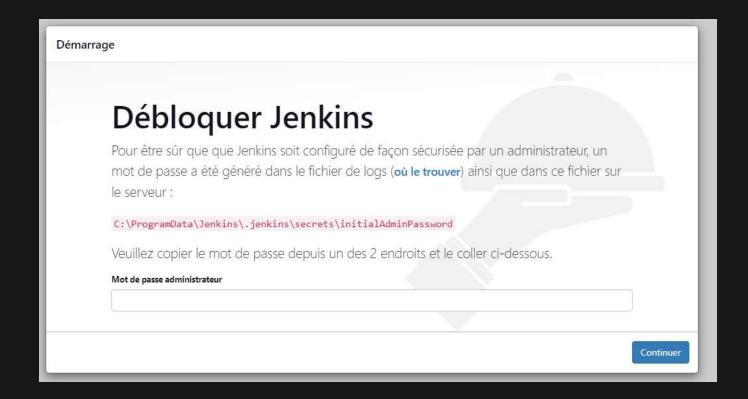


- 1. Nous sommes DevOps
- 2. Cl
- 3. CD
- 4. Installation Jenkins
- 5. Configuration Jenkins
- 6. Jobs freestyle
- 7. Le Jenkinsfile



# Configuration Jenkins





Pour des raisons de sécurité, pour vérifier que vous avez les droits nécessaires sur la machine, vous devez entrer un code



#### **Personnaliser Jenkins**

Les plugins étendent Jenkins avec des fonctionnalités additionnelles pour satisfaire différents besoins.

#### Installer les plugins suggérés

Installer les plugins que la communauté Jenkins trouve les plus utiles.

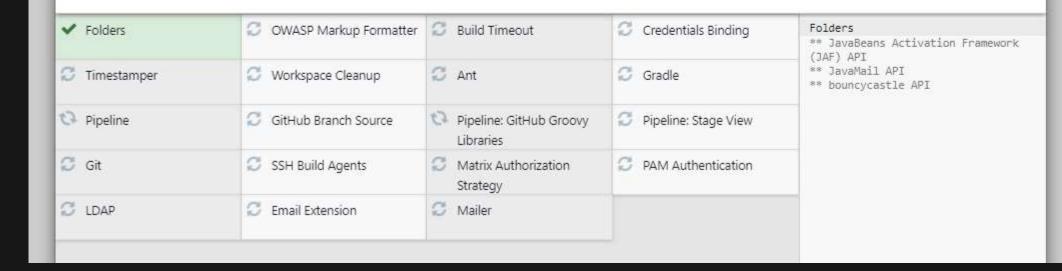
## Sélectionner les plugins à installer

Sélectionner et installer les plugins les plus utiles à vos besoins.



Installation en cours...

### Installation en cours...







#### Créer le 1er utilisateur Administrateur

Nom d'utilisateur:
Not de passe:
Confirmation du mot de passe:
Nom complet:
Adresse courriel:



#### Démarrage

#### Créer le 1er utilisateur Administrateur

om d'utilisateur:
admin
lot de passe:
onfirmation du mot de passe:
••••
om complet:
admin admin
dresse courriel:
admin@admin.com



#### Démarrage

## Configuration de l'instance

#### URL de Jenkins:

http://localhost:8080/

L'URL de Jenkins est utilisée pour fournir l'URL de base pour les liens absolus vers les diverses ressources Jenkins. Cela signifie que cette valeur est nécessaire pour le bon fonctionnement de nombreuses fonctionnalités de Jenkins, notamment les notifications par mail, les mises à jour des statuts des pull requests, et la variable d'environnement BUILD\_URL fournie pour les étapes de build.

La valeur par défaut affichée n'est pas encore sauvegardée et est générée à partir de la requête actuelle, lorsque c'est possible. Il est fortement

recommandé d'utiliser comme valeur l'URL qui est censée être utilisée par les utilisateurs. Cela évitera des confusions lors du partage ou de la visualisation de liens.









- 1. Nous sommes DevOps
- 2. Cl
- 3. CD
- 4. Installation Jenkins
- 5. Configuration Jenkins
- 6. Jobs freestyle
- 7. Le Jenkinsfile



## Jobs freestyle



- 1. Créer un projet free-style
- 2. Ajouter des build steps
- 3. Archiver les artefacts
- 4. Ajouter un SCM
- 5. Ajouter des triggers pour lancer les jobs automatiquement



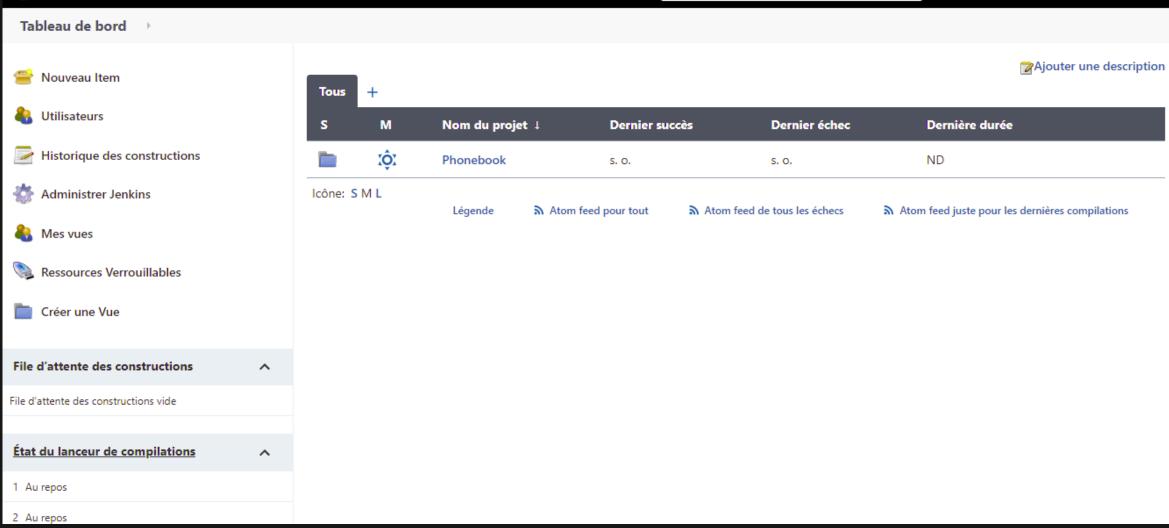
- 1. Créer un projet free-style
- 2. Ajouter des build steps
- 3. Archiver les artefacts
- 4. Ajouter un SCM
- 5. Ajouter des triggers pour lancer les jobs automatiquement



## CREER UN PROJET FREESTYLE



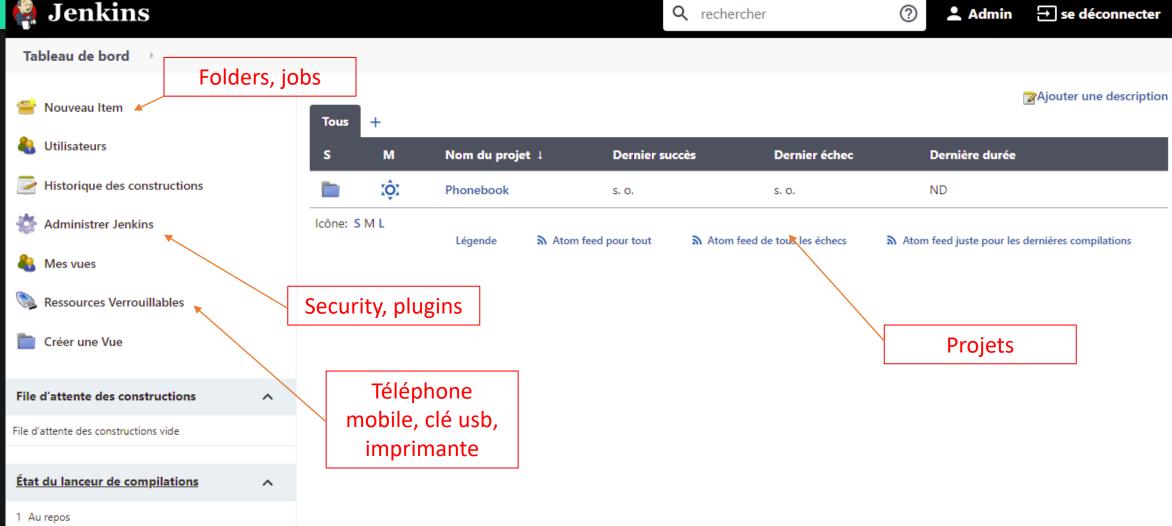






2 Au repos

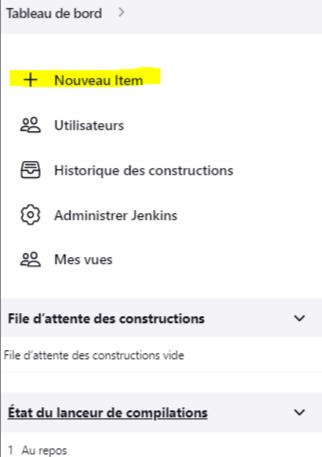








2 Au repos



## **CERTIF ACADEMY**

#### Saisissez un nom

Erphrense

» Champ obligatoire



#### Construire un projet free-style

Ceci est la fonction principale de Jenkins qui sert à builder (construire) votre projet. Vous pouvez intégrer tous les outils de autre chose qu'un build logiciel.



#### **Pipeline**

Organise des activités de longue durée qui peuvent sétendre sur plusieurs agents de construction. Adapté pour la création qui ne sadaptent pas facilement à des tâches de type libre.



### Construire un projet multi-configuration

Adapté aux projets qui nécessitent un grand nombre de configurations différentes, comme des environnements de test mu



#### Dossier

Crée un conteneur qui stocke des objets imbriqués. Utile pour grouper ensemble des éléments. Contrairement à une vue q plusieurs éléments du même nom tant qu'ils se trouvent dans des dossiers différents.



### **Organization Folder**

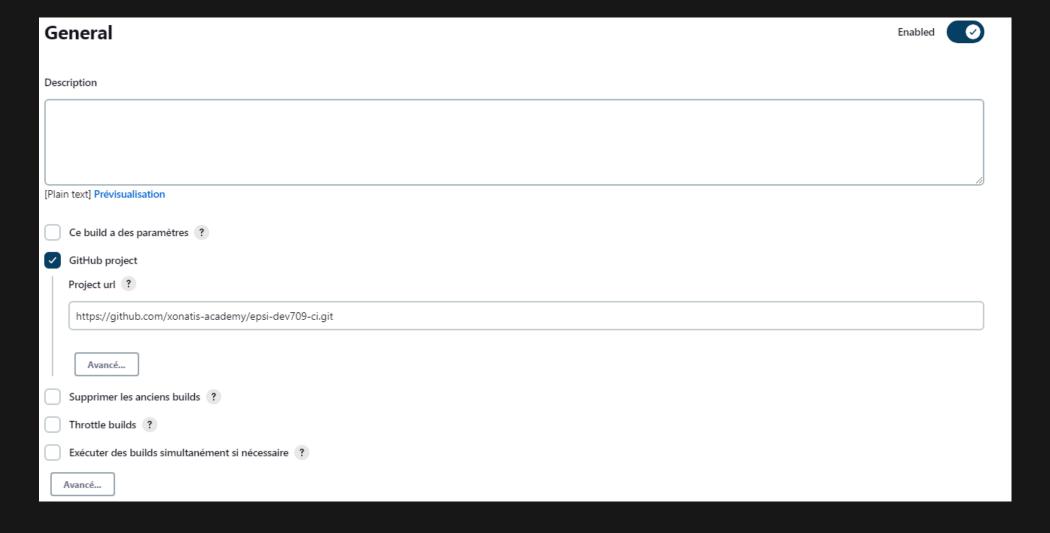
Creates a set of multibranch project subfolders by scanning for repositories.



### **Pipeline Multibranches**

Crée un ensemble de projets Pipeline en se basant sur les branches détectées dans le dépôt dun gestionnaire de code sour







- 1. Créer un projet free-style
- 2. Ajouter des build steps
- 3. Archiver les artefacts
- 4. Ajouter un SCM
- 5. Ajouter des triggers pour lancer les jobs automatiquement



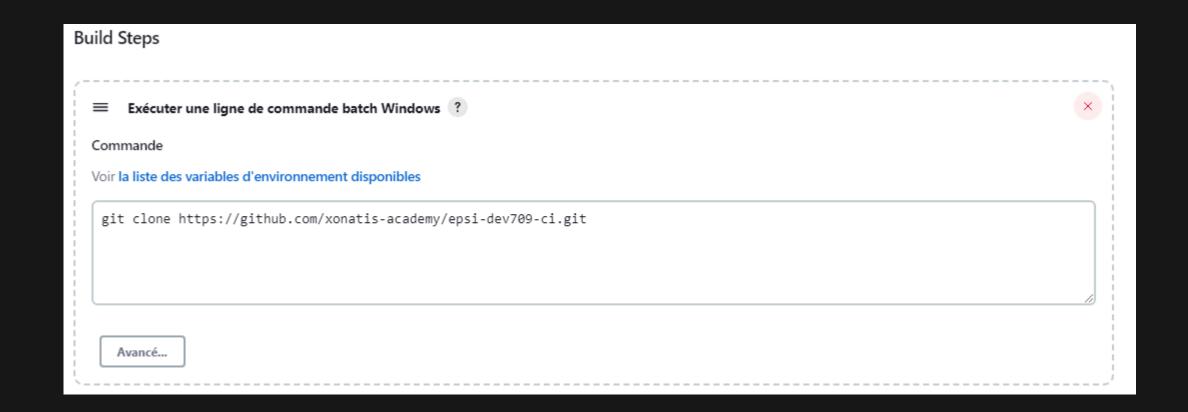
## **BUILD STEPS**





Ajouter une étape de build qui permet de cloner le projet, vérifiez le workspace ©









Ajouter une étape de build qui permet de cleaner le projet, vérifiez le workspace ©









Ajouter une étape de build qui permet de compiler le projet, vérifiez le workspace ©



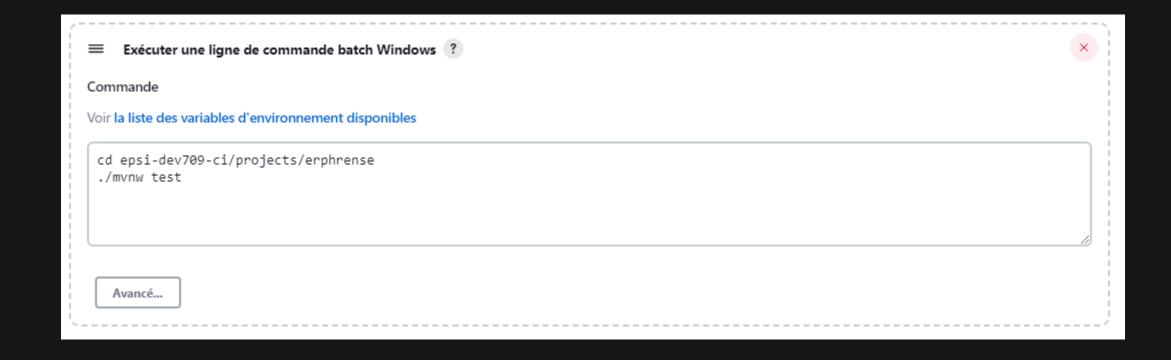






Ajouter une étape de build qui permet de tester le projet, vérifiez le workspace ©



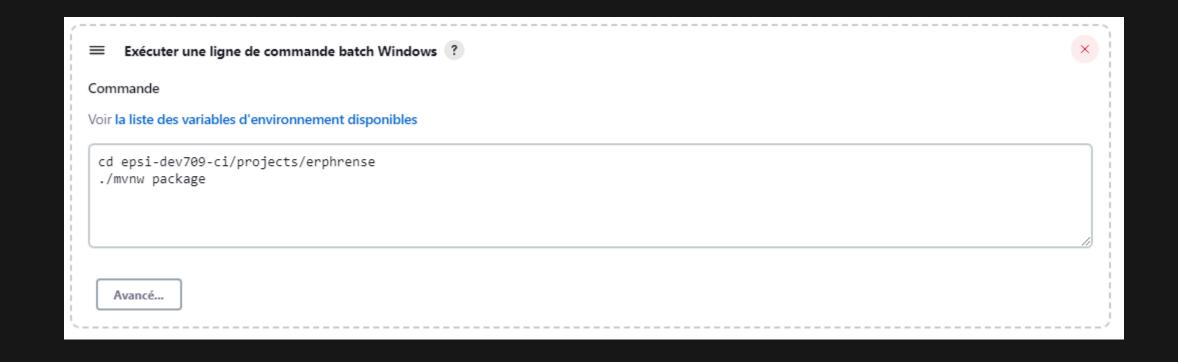






Ajouter une étape de build qui permet de packager le projet, vérifiez le workspace ©







- 1. Créer un projet free-style
- 2. Ajouter des build steps
- 3. Archiver les artefacts
- 4. Ajouter un SCM
- 5. Ajouter des triggers pour lancer les jobs automatiquement



## ARTEFACT



Un artifact est le résultat produit par un job. Par exemple, pour un build job, l'artifact récupéré peut être la version packagée et distribuable de l'application.

Habituellement, l'artifact (parlé) est le résultat du processus de build.





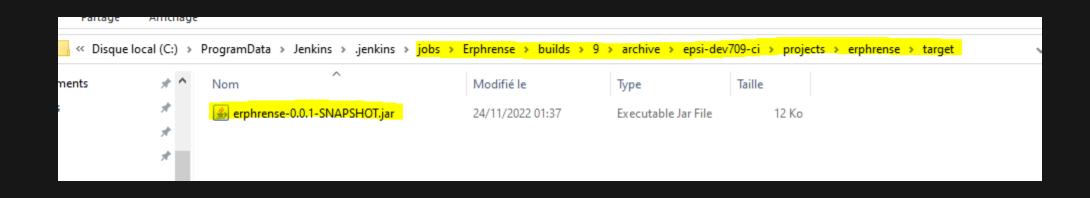
# Archivez l'artefact du build dans les actions après le build ©





Archiver des artefacts ?	×
Fichiers à archiver ?	
epsi-dev709-ci\projects\erphrense\target\erphrense-0.0.1-SNAPSHOT.jar	
Exclusions ?	
Ne pas faire échouer la construction si l'archive retournée est vide ?	
Archive les artifacts seulement si la construction est un succès	
Marquer d'une empreinte numérique tous les artefacts archivés	
Utilise les exclusions par défaut ?	
Treat include and exclude patterns as case sensitive ?	
Follow symbolic links ?	











Artefacts du build





Aucun changement.



Lancé par un utilisateur anonyme





# Modifiez le nom de l'artefact en fonction du build ©





Dans le cadre du application Maven, il est préférable de modifier la version du fichier pom. Cela dit, il est plus intéressant de savoir comment utiliser les variables d'environnement pour l'ensemble de vos besoins!



Exécuter une ligne de commande batch Windows ?	×
Commande	
Voir la liste des variables d'environnement disponibles	
rename projects\erphrense\target\erphrense-0.0.1-SNAPSHOT.jar erphrense-%BUILD_NUMBER%.jar	
Avancé	,
Ajouter une étape au build ▼	
Tyoutal and etape ad band	
Actions à la suite du build	
<u> </u>	
Archiver des artefacts ?	×
Fichiers à archiver ?	
projects/erphrense/target/erphrense- <mark>*.</mark> jar	
Avancé	



Attention, en fonction de votre terminal, l'écriture peut-être différente : par exemple, sous bash, écrivez plutôt \${BUILD\_NUMBER}

## CERTIF ACADEMY





Artefacts du build



11,51 KB view



No changes.



Lancé par un utilisateur anonyme



Revision: f50df222005182daef1258810bf9a1fa6d8f90fc

Repository: https://github.com/xonatis-academy/epsi-dev709-ci.git

· refs/remotes/origin/main



- 1. Créer un projet free-style
- 2. Ajouter des build steps
- 3. Archiver les artefacts
- 4. Ajouter un SCM
- 5. Ajouter des triggers pour lancer les jobs automatiquement



## SCM



SCM (Source Control Management) est l'outil de gestion de version de votre application.

Il se connecte sur Jenkins séparément habituellement.





## Ajoutez un SCM à votre projet ©





Gestion de code source	
○ Aucune	
● Git ?	
Repositories ?	
Repository URL ?	×
https://github.com/xonatis-academy/epsi-dev709-ci.git	
Credentials ?	
- aucun -	~
+ Ajouter	
Avancé	
Add Repository	
Branches to build ?	
Branch Specifier (blank for 'any') ?	×
*/main	
Add Branch	



Nous pouvons également supprimer l'étape du git clone car Jenkins le fera automatiquement.

Par ailleurs, il faudrait potentiellement modifier les étapes du build pour enlever le dossier intermédiaire (reportez-vous au workspace).







## Construction #11 (24 nov. 2022, 01:54:15)



Artefacts du build





No changes.



Lancé par un utilisateur anonyme



Revision: 1f489a92c15700fb965f714e54e3d3a63757af6c

Repository: https://github.com/xonatis-academy/epsi-dev709-ci.git

refs/remotes/origin/main



- 1. Créer un projet free-style
- 2. Ajouter des build steps
- 3. Archiver les artefacts
- 4. Ajouter un SCM
- 5. Ajouter des triggers pour lancer les jobs automatiquement



### **TRIGGERS**



Chaque build job peut être déclenché par un événement (trigger), habituellement par des webhooks ou du polling.

Un SCM est très souvent utilisé pour savoir quand builder!





### Qu'est-ce que les webhooks?







### Qu'est-ce que le polling?







Modifiez votre job pour qu'il se lance dès qu'un commit est enregistré dans votre remote (polling toutes les 5 minutes) ©





Ce qui déclenche le build
Déclencher les builds à distance (Par exemple, à partir de scripts) ?
Construire après le build sur d'autres projets ?
Construire périodiquement ?
GitHub hook trigger for GITScm polling ?
✓ Scrutation de l'outil de gestion de version ?
Planning ?
H/5 * * * *
Aurait été lancé à jeudi 24 novembre 2022 à 01:56:10 heure normale d'Europe centrale; prochaine exécution à jeudi 24 novembre 2022 à 02:01:10 heure normale d'Europe centrale.
Ignore post-commit hooks ?



Le mieux reste d'utiliser les webhooks car ils fonctionnent sur la notification. Cela permet d'éviter de gâcher des ressources mais cela demande un adresse IP publique.



#### Dernier Log du dernier accès à Git

```
Started on 24 nov. 2022, 02:41:00

Using strategy: Default

[poll] Last Built Revision: Revision 1f489a92c15700fb965f714e54e3d3a63757af6c (refs/remotes/origin/main)

The recommended git tool is: NONE

No credentials specified

> git.exe --version # timeout=10

> git --version # 'git version 2.31.0.windows.1'

> git.exe ls-remote -h -- https://github.com/xonatis-academy/epsi-dev709-ci.git # timeout=10

Found 1 remote heads on https://github.com/xonatis-academy/epsi-dev709-ci.git

[poll] Latest remote head revision on refs/heads/main is: 1f489a92c15700fb965f714e54e3d3a63757af6c - already built by 11

Done. Took 0,5 s

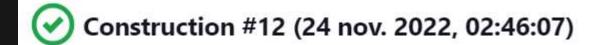
No changes
```



Lorsqu'un développeur(euse) effectue un commit sur la branch main du remote ou l'un des merges de PR

. . .

#### CERTIF ACADEMY





Artefacts du build

erphrense-0.0.1-SNAPSHOT.jar 11,51 KB view



Changes

1. Adding readme (details / githubweb)



Lancé par un changement dans la base de code



Revision: f50df222005182daef1258810bf9a1fa6d8f90fc

Repository: https://github.com/xonatis-academy/epsi-dev709-ci.git

refs/remotes/origin/main



- 1. Rappel sur les projets Maven
- 2. Qu'est-ce que la Cl?
- 3. Qu'est-ce que la CD?
- 4. Installation de Jenkins
- 5. Configuration
- 6. Utilisation
- 7. Recap'



# **RECAP'**





### Qu'est-ce qu'on a appris?





- 1. On peut automatiser n'importe quelle tâche dans le terminal
- 2. On a pu ajouter Git comme SCM
- 3. On a pu ajouter des triggers pour déclencher le job à par des changements dans le SCM
- 4. On sait archiver des artefacts, prêt à être livrés
- 5. On sait numéroter ces artefacts et facilement revenir en arrière









- 1. Nous sommes DevOps
- 2. Cl
- 3. CD
- 4. Installation Jenkins
- 5. Configuration Jenkins
- 6. Jobs freestyle
- 7. Le Jenkinsfile



## Le Jenkinsfile



Il est possible de déporter le code du pipeline directement dans le SCM (git par exemple) afin de centraliser le code du projet :

- Code source de l'application
- Code du CI/CD du pipeline



Pour ce faire, il faut créer une fichier nommé « Jenkinsfile » (par convention)



PHRENSE
.mvn
SFC
target
.gitignore
Jenkinsfile
mvnw
mvnw.cmd
pom.xml
readme.md

```
Jenkinsfile
      pipeline {
          agent any
          triggers {
           pollSCM 'H/5 * * * *'
        stages {
          stage('checkout') {
            steps {
             checkout
 15
          stage('clean') {
            steps {
               dir('projects/erphrense') {
                 bat './mvnw clean'
          stage('compile') {
            steps {
               dir('projects/erphrense') {
                 bat './mvnw compile'
          stage('test') {
            steps {
               dir('projects/erphrense') {
                 bat './mvnw test'
```



	Declarative: Checkout SCM	checkout	clean	compile	test	package	archive
Average stage times: (Average <u>full</u> run time: ~31s)	15	1s	3s	6s	7s	7s	573ms
#16 Nov 24 2 05:09 commits	1s	1s	3s	6s	7s	7s	573ms