

Procédure de Migration des depots DevOps (Partie Jenkins)

1 Migration du repository « Install_Webapp » :

Ci-dessous les URLs GitBucket et SGitHub du repository Install_Webapp :

- Url GitBucket : https://gitbucket.cgasg.local/gitbucket/git/DEVOPS/Install_Webapp.git
- Url SGitHub : [SGF/Install_Webapp\(world.socgen\)](SGF/Install_Webapp(world.socgen))
- Url Job Jenkins « Install_Webapps_Backend » : [Install_Webapps_Backend \[Jenkins\] \(sgf.socgen\)](#)
- Url Job Jenkins « Install_Webapps_Fatjar » : [Install_Webapps_Fatjar \[Jenkins\] \(sgf.socgen\)](#)
- Url Job Jenkins « Install_Webapps_Frontend » : [Install_Webapps_Frontend \[Jenkins\] \(sgf.socgen\)](#)

Le repository Install_Webapp utilise 3 fichiers JenkinsFile :

- JenkinsFileBackend
- JenkinsFileFatjar
- JenkinsFileFrontend

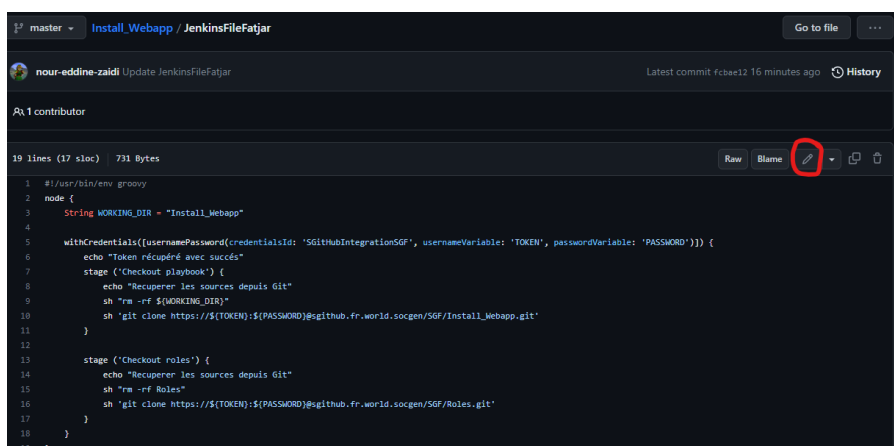
Chaque fichier JenkinsFile, correspond à un Job sur l'interface web de Jenkins admin-automation :

- Install_Webapps_Backend
- Install_Webapps_Fatjar
- Install_Webapps_Frontend

1.1 Fatjar :

1.1.1 Modification du fichier JenkinsFileFatjar :

- Allez sur https://sgithub.fr.world.socgen/SGF/Install_Webapp/blob/master/JenkinsFileFatjar
- Cliquez sur l'icône avec le crayon :



```
1 #!/usr/bin/env groovy
2 node {
3     String WORKING_DIR = "Install_Webapp"
4
5     withCredentials([usernamePassword(credentialsId: 'SGitHubIntegrationSGF', usernameVariable: 'TOKEN', passwordVariable: 'PASSWORD')]) {
6         echo "Token récupéré avec succès"
7         stage ("Checkout playbook") {
8             echo "Recuperer les sources depuis Git"
9             sh "rm -rf ${WORKING_DIR}"
10            sh "git clone https://$TOKEN:$PASSWORD@github.fr.world.socgen/SGF/Install_Webapp.git"
11        }
12
13        stage ("Checkout roles") {
14            echo "Recuperer les sources depuis Git"
15            sh "rm -rf Roles"
16            sh "git clone https://$TOKEN:$PASSWORD@github.fr.world.socgen/SGF/Roles.git"
17        }
18    }
19 }
```

- Remplacez le script par le nouveau ci-dessous ; On mettra les stages « Test Playbook » et « Run Playbook » du pipeline en commentaire pour le moment :

```
#!/usr/bin/env groovy
node {
```

```

String WORKING_DIR = "Install_Webapp"

withCredentials([usernamePassword(credentialsId: 'SGitHubIntegrationSGF',
usernameVariable: 'TOKEN', passwordVariable: 'PASSWORD')]) {

    stage ('Checkout playbook') {
        echo "Recuperer les sources depuis Git"
        sh "rm -rf ${WORKING_DIR}"
        sh 'git clone
https://${TOKEN}:${PASSWORD}@sgithub.fr.world.socgen/SGF/Install_Webapp.git'
    }

    stage ('Checkout roles') {
        echo "Recuperer les sources depuis Git"
        sh "rm -rf Roles"
        sh 'git clone
https://${TOKEN}:${PASSWORD}@sgithub.fr.world.socgen/SGF/Roles.git'
    }

    /*stage ('Test playbook') {
        echo "Test du playbook"
        sh "cd ${WORKING_DIR} && \
        ansible-playbook installFatJar.yml \
        --inventory environment/${ENVIRONNEMENT}/inventory/hosts.yml \
        --extra-var 'env=${ENVIRONNEMENT}' \
        --extra-var @environment/${ENVIRONNEMENT}/${APPLI}.yml \
        --syntax-check"
    }

    stage ('Run playbook') {
        ansiColor('xterm') {
            echo "Lancement du playbook"
            if (MAJ_CERTIFICAT_UNIQUEMENT == "false") {
                sh "cd ${WORKING_DIR} && \
                ansible-playbook installFatJar.yml \
                --inventory environment/${ENVIRONNEMENT}/inventory/hosts.yml \
                --extra-var 'env=${ENVIRONNEMENT}' \
                --extra-var @environment/${ENVIRONNEMENT}/${APPLI}.yml"
            } else {
                sh "cd ${WORKING_DIR} && \
                ansible-playbook installFatJar.yml \
                --inventory environment/${ENVIRONNEMENT}/inventory/hosts.yml \
                --extra-var 'env=${ENVIRONNEMENT}' \
                --extra-var @environment/${ENVIRONNEMENT}/${APPLI}.yml \
                --tags update_cert"
            }
        }
    }
}*/
}

```



- Cliquez sur “Commit changes”.

1.1.2 Modification du job Jenkins « Install_Webapps_Fatjar » correspondant :

- Allez sur https://admin-automation.sgf.socgen/job/Install_Webapps_Fatjar/ :
- Cliquez sur « Configure » :

Jenkins Search (CTRL+K) ZAIID Nour-Eddine (A) log out

Dashboard > Install_Webapps_Fatjar

Install_Webapps_Fatjar
Installation d'un fatjar

Left sidebar: Status, Changes, Build with Parameters, **Configure** (circled), Delete Pipeline, Move, Full Stage View, Rename, Pipeline Syntax.

Build History: #356 (Jun 18, 2024, 9:49 AM), #355 (May 27, 2024, 4:02 PM).

Stage View

	Checkout playbook	Checkout roles	Test playbook	Run playbook
Average stage times: (Average full run time: ~2min 29s)	8s	8s	946ms	2min 2s
#356 Jun 18 09:49 No Changes	10s	4s	897ms	3min 26s
#355 May 27 16:02 No Changes	8s	10s	1s	3min 51s
#354 May 22 17:30 No Changes	10s	12s	901ms	3min 11s

Right sidebar: Edit description, Disable Project.

- Descendez au champs « Definition > SCM > Repositories > Repository URL » et changez l'uri de GitBucket par la nouvelle de SGitHub :
- Sélectionnez « MyGitHubRepoToken/***** (SGitHub Integration SGF) » dans la section « Credentials ».

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

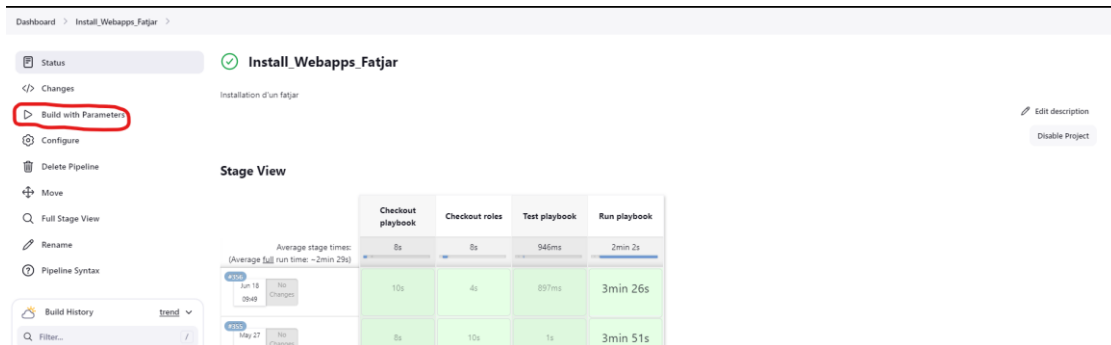
https://sgithub.fr.world.socgen/SGF/Install_Webapp.git

Credentials ?

MyGitHubRepoToken/***** (SGitHub Integration SGF)

+ Add

- Testez que le Job Jenkins arrive à récupérer les sources depuis Git et effectuer le git clone avec succès :
 - o Allez sur https://admin-automation.sgf.socgen/job/Install_Webapps_Fatjar/
 - o Cliquez sur « Build with Parameters » :



- Sélectionnez dans « Environnement » la variable « dev », puis cliquez sur « Build » :

Pipeline Install_Webapps_Fatjar

This build requires parameters:

Environnement
Environnement

dev

Appli

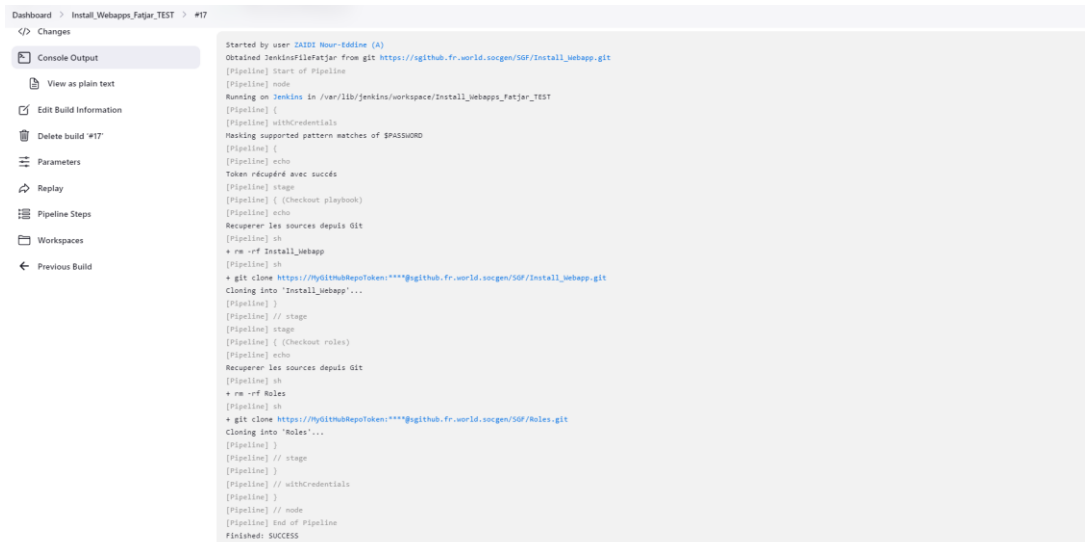
Fichier de définition (sans le .yaml)

☐ Maj_certificat_uniquement

Build

Cancel

- Le résultat du build devra retourner un output similaire à la capture d'écran ci-dessous :

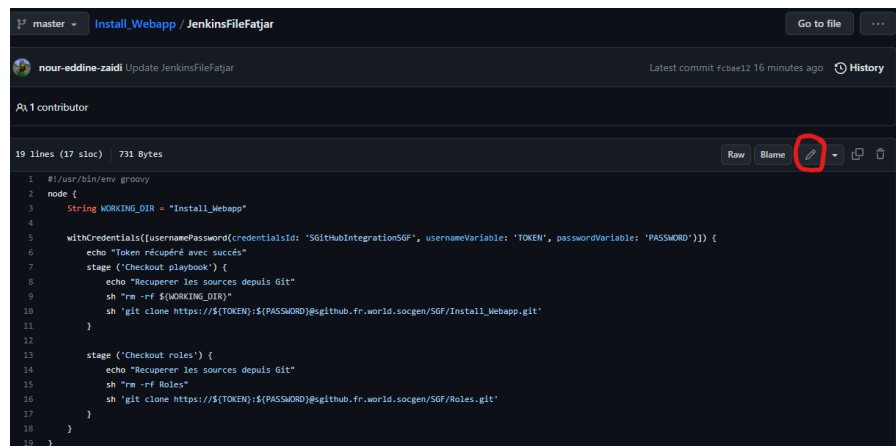


1.2 Backend :

1.2.1 Modification du fichier JenkinsFileBackend :

- Allez sur https://sgithub.fr.world.socgen/SGF/Install_Webapp/blob/master/JenkinsFileBackend

- Cliquez sur l'icône avec le crayon :



- Remplacez le script par le nouveau ci-dessous ; On mettra les stages « Test Playbook » et « Run Playbook » du pipeline en commentaire pour le moment :

```
#!/usr/bin/env groovy
node {
    String WORKING_DIR = "Install_Webapp"
    withCredentials([usernamePassword(credentialsId:
'SGithHubIntegrationSGF', usernameVariable: 'TOKEN', passwordVariable:
'PASSWORD')]) {
        echo "Token récupéré avec succès"
        stage ('Checkout playbook') {
            echo "Recuperer les sources depuis Git"
            sh "rm -rf ${WORKING_DIR}"
            sh 'git clone
https://${TOKEN}:${PASSWORD}@sgithub.fr.world.socgen/SGF/Install_Webapp.
git'
        }

        stage ('Checkout roles') {
            echo "Recuperer les sources depuis Git"
            sh "rm -rf Roles"
            sh "git clone
https://${TOKEN}:${PASSWORD}@sgithub.fr.world.socgen/SGF/Roles.git"
        }

        /*stage ('Test playbook') {
            echo "Test du playbook"
            sh "cd ${WORKING_DIR} && \
            ansible-playbook installBackend.yml \
            --inventory environment/$ENVIRONNEMENT/inventory/hosts.yml
\
            --extra-var 'env=$ENVIRONNEMENT' \
            --extra-var @environment/$ENVIRONNEMENT/${APPLI}.yml \
            --syntax-check"
        }
        stage ('Run playbook') {
```

```

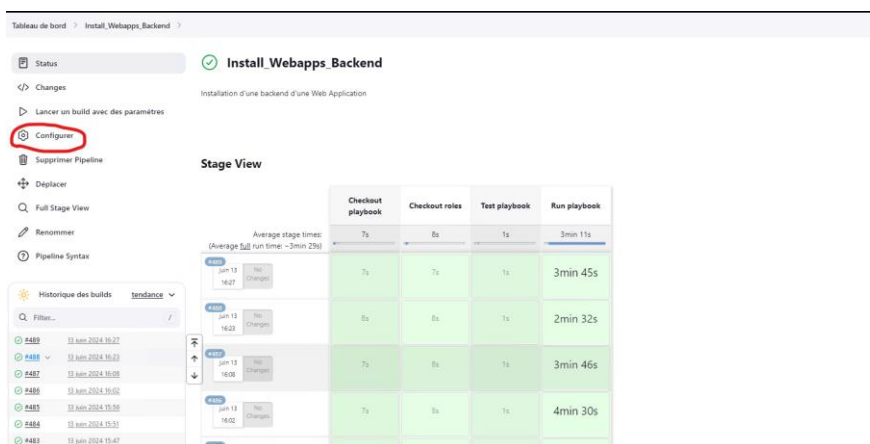
    ansiColor('xterm') {
        echo "Lancement du playbook"
        if (MAJ_CERTIFICAT_UNIQUEMENT == "false") {
            sh "cd ${WORKING_DIR} && \
                ansible-playbook installBackend.yml \
                --inventory
environment/${ENVIRONNEMENT}/inventory/hosts.yml \
                --extra-var 'env=${ENVIRONNEMENT}' \
                --extra-var @environment/${ENVIRONNEMENT}/${APPLI}.yml"
        } else {
            sh "cd ${WORKING_DIR} && \
                ansible-playbook installBackend.yml \
                --inventory
environment/${ENVIRONNEMENT}/inventory/hosts.yml \
                --extra-var 'env=${ENVIRONNEMENT}' \
                --extra-var @environment/${ENVIRONNEMENT}/${APPLI}.yml \
                --tags update_cert"
        }
    }
}
}
}

```

- Cliquez sur "Commit changes".

1.2.2 Modification du job Jenkins « Install_Webapps_Backend» correspondant :

- Allez sur https://admin-automation.sgf.socgen/job/Install_Webapps_Backend/
- Cliquez sur « Configure » :



- Descendez au champs « Definition > SCM > Repositories > Repository URL » et changez l'url de GitBucket par la nouvelle de SGitHub :
`https://sgithub.fr.world.socgen/SGF/Install_Webapp.git`
- Sélectionnez « MyGitHubRepoToken/***** (SGitHub Integration SGF) » dans la section « Credentials » :

Pipeline

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

https://github.fr:world.socgen:SGF/Install_Webapps.git

Credentials ?

MyGitHubRepoToken***** (SGF) GitHub Integration SGF

Appliquer

Avancé

Add Repository

- Testez que le Job Jenkins arrive à récupérer les sources depuis Git et effectuer le git clone avec succès :
 - o Allez sur https://admin-automation.sgf.socgen/job/Install_Webapps_Backend/
 - o Cliquez sur « Build with Parameters » :

Dashboard > Install_Webapps_Backend >

Status

Install_Webapps_Backend

Installation d'une backend d'une Web Application

Build with Parameters

Configure

Delete Pipeline

Move

Full Stage View

Rename

Pipeline Syntax

Stage View

Average stage times:
(Average full run time: ~3min 29s)

Checkout roles	Test playbook	Run playbook
8s	1s	3min 11s
7s	1s	3min 45s

- o Sélectionnez dans « Environnement » la variable « dev », puis cliquez sur « Build »

Tableau de bord > Install_Webapps_Backend >

Status

Changes

Lancer un build avec des paramètres

Configurer

Supprimer Pipeline

Déplacer

Full Stage View

Renommer

Pipeline Syntax

Pipeline Install_Webapps_Backend

Ce build nécessite des paramètres:

Paramètres

dev

Appel

Revenir au définition dans le job

Mod. certificat uniquement

Build

Cancel

Historique des builds

12 juin 2024 15:02

12 juin 2024 15:02

12 juin 2024 15:02

12 juin 2024 15:02

12 juin 2024 15:02

12 juin 2024 15:02

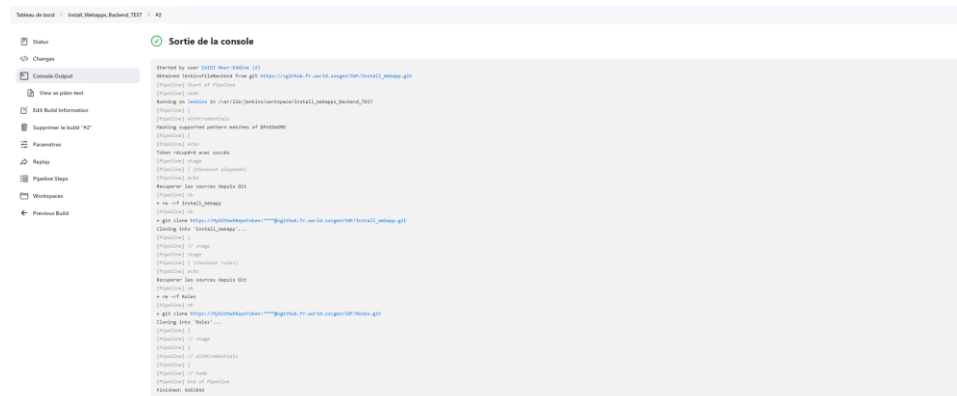
12 juin 2024 15:02

12 juin 2024 15:02

12 juin 2024 15:02

12 juin 2024 15:02

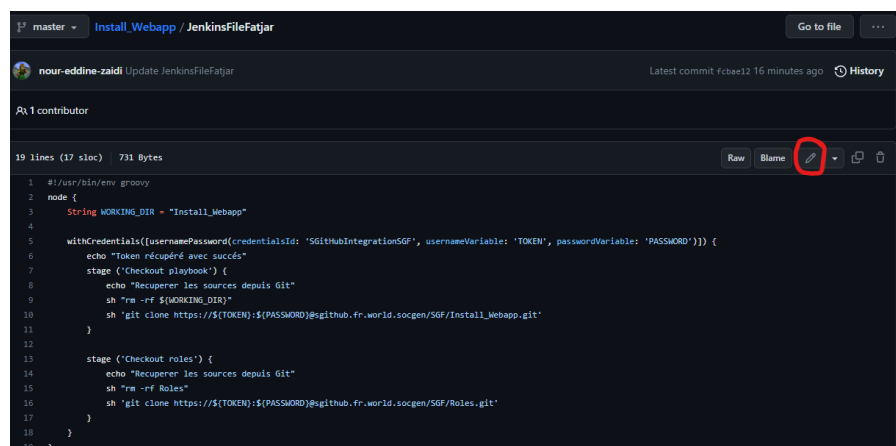
- Le résultat du build devra retourner un output similaire à la capture d'écran ci-dessous :



1.3 Frontend :

1.3.1 Modification du fichier JenkinsFileFrontend :

- Allez sur https://sgithub.fr.world.socgen/SGF/Install_Webapp/blob/master/JenkinsFileFrontend
- Cliquez sur l'icône avec le crayon :



- Remplacez le script par le nouveau ci-dessous ; On mettra les stages « Test Playbook » et « Run Playbook » du pipeline en commentaire pour le moment :

```
#!/usr/bin/env groovy
node {
    String WORKING_DIR = "Install_Webapp"

    withCredentials([usernamePassword(credentialsId: 'SGitHubIntegrationSGF', usernameVariable: 'TOKEN', passwordVariable: 'PASSWORD')]) {
        stage('Checkout playbook') {
            echo "Recuperer les sources depuis Git"
            sh "rm -rf ${WORKING_DIR}"
            sh 'git clone https://${TOKEN}:${PASSWORD}@sgithub.fr.world.socgen/SGF/Install_Webapp.git'
        }

        stage('Checkout roles') {
            echo "Recuperer les sources depuis Git"
            sh "rm -rf Roles"
            sh 'git clone https://${TOKEN}:${PASSWORD}@sgithub.fr.world.socgen/SGF/Roles.git'
        }
    }
}
```



```

- stage ('Checkout roles') {
-     echo "Recuperer les sources depuis Git"
-     sh "rm -rf Roles"
-     sh 'git clone
https://${TOKEN}:${PASSWORD}@sgithub.fr.world.socgen/SGF/Roles.git'
- }

- /*stage ('Test playbook') {
-     echo "Test du playbook"
-     ansiColor('xterm') {
-         dir ( WORKING_DIR ) {
-             ansiblePlaybook playbook: 'installFrontend.yml',
-             inventory:
'environment/'+ENVIRONNEMENT+'/inventory/hosts.yml',
-             extras: '-e env='+ENVIRONNEMENT+' -e
@environment/'+ENVIRONNEMENT+'/'+'APPLI+'.yml --syntax-check',
-             colored: true
-         }
-     }
- }

- stage ('Run playbook') {
-     echo "Run du playbook"
-     ansiColor('xterm') {
-         dir ( WORKING_DIR ) {
-             if (UPGRADE_APACHE_UNIQUEMENT == 'true') {
-                 ansiblePlaybook playbook: 'installFrontend.yml',
-                 inventory:
'environment/'+ENVIRONNEMENT+'/inventory/hosts.yml',
-                 extras: '-e env='+ENVIRONNEMENT+' -e
@environment/'+ENVIRONNEMENT+'/'+'APPLI+'.yml',
-                 tags: 'upgrade',
-                 colored: true
-             } else {
-                 if ((MAJ_CERTIFICAT_UNIQUEMENT == 'true') &&
(MAJ_VIRTUAL_HOST_UNIQUEMENT == "true")) {
-                     ansiblePlaybook playbook:
'installFrontend.yml',
-                     inventory:
'environment/'+ENVIRONNEMENT+'/inventory/hosts.yml',
-                     extras: '-e env='+ENVIRONNEMENT+' -e
@environment/'+ENVIRONNEMENT+'/'+'APPLI+'.yml',
-                     tags: 'update_vhost, update_cert',
-                     colored: true
-                 } else if ((MAJ_CERTIFICAT_UNIQUEMENT == 'true')
&& (MAJ_VIRTUAL_HOST_UNIQUEMENT == 'false')) {
-                     ansiblePlaybook playbook:
'installFrontend.yml',

```

```

inventory:
'environment/'+ENVIRONNEMENT+'/inventory/hosts.yml',
    extras: '-e env='+ENVIRONNEMENT+' -e
@environment/'+ENVIRONNEMENT+'/'+'APPLI+'.yaml',
    tags: 'update_cert',
    colored: true
    } else if ((MAJ_CERTIFICAT_UNIQUEMENT ==
'false') && (MAJ_VIRTUAL_HOST_UNIQUEMENT == 'true')) {
    ansiblePlaybook playbook:
'installFrontend.yml',
    inventory:
'environment/'+ENVIRONNEMENT+'/inventory/hosts.yml',
    extras: '-e env='+ENVIRONNEMENT+' -e
@environment/'+ENVIRONNEMENT+'/'+'APPLI+'.yaml',
    tags: 'update_vhost',
    colored: true
    } else {
    ansiblePlaybook playbook:
'installFrontend.yml',
    inventory:
'environment/'+ENVIRONNEMENT+'/inventory/hosts.yml',
    extras: '-e env='+ENVIRONNEMENT+' -e
@environment/'+ENVIRONNEMENT+'/'+'APPLI+'.yaml',
    colored: true
    }
    }
    }
    }
    }*/
    }
}

```

- Cliquez sur "Commit changes".

1.3.2 Modification du job Jenkins « Install_Webapps_Backend » correspondant :

- Allez sur https://admin-automation.sgf.socgen/job/Install_Webapps_Backend/
- Cliquez sur « Configurer » :

Tableau de bord

Install_Webapps_Frontend

Status

Changements

Lancer un build avec des paramètres

Configurer

Supprimer Pipeline

Déplacer

Full Stage View

Renommer

Pipeline Syntax

Install_Webapps_Frontend

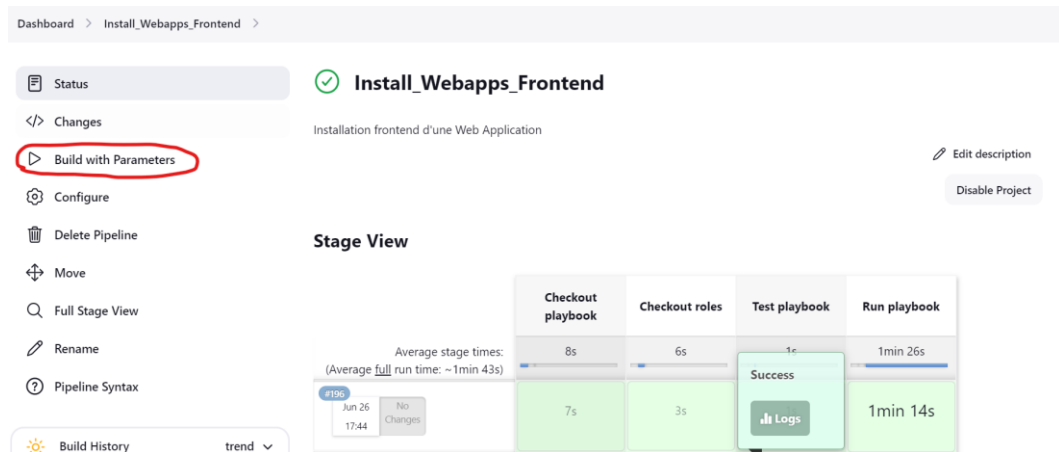
Installation frontend d'une Web Application

Stage View

	Checkout playbook	Checkout rules	Test playbook	Run playbook
Average stage time: (Average job run time - 1min 43s)	6s	6s	1s	1min 26s
20 juin 2024 17:44	7s	3s	1s	1min 14s
20 juin 2024 17:42	12s	5s	1s	1min 25s
20 juin 2024 17:40	7s	3s	1s	29s

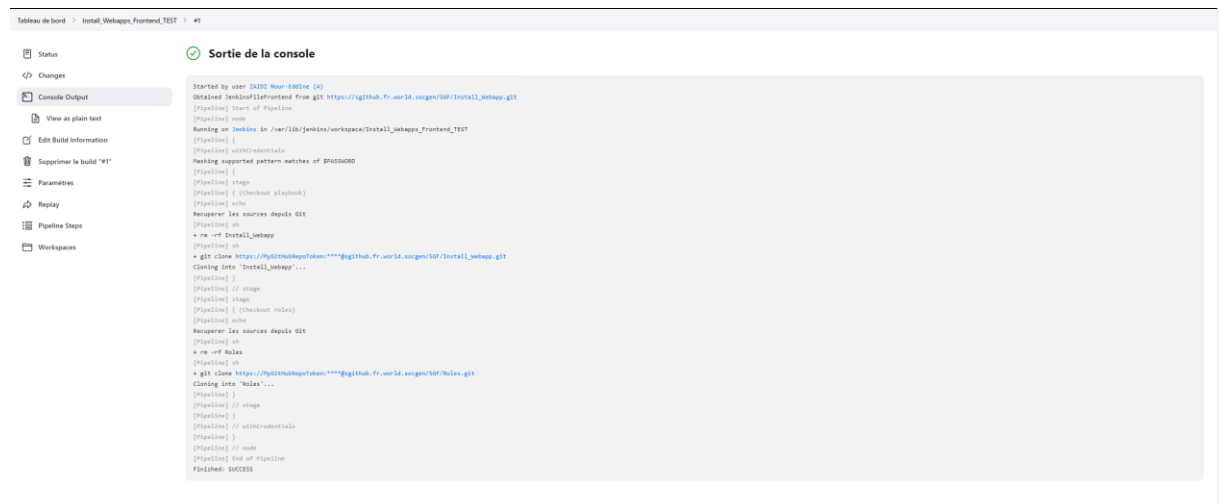
- Descendez au champs « Definition > SCM > Repositories > Repository URL » et changez l'url de GitBucket par la nouvelle de SGitHub :
https://sgithub.fr.world.socgen/SGF/Install_Webapp.git
- Sélectionnez « MyGitHubRepoToken/***** (SGitHub Integration SGF) » dans la section « Credentials » :

- Testez que le Job Jenkins arrive à récupérer les sources depuis Git et effectuer le git clone avec succès :
 - o Allez sur https://admin-automation.sgf.socgen/job/Install_Webapps_Backend/
 - o Cliquez sur « Build with Parameters » :



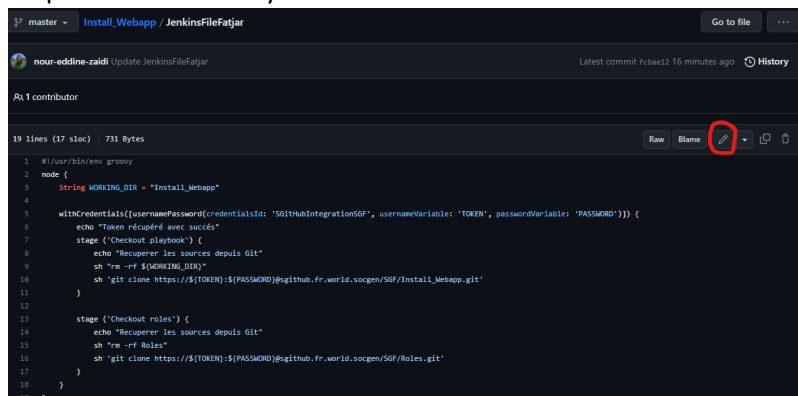
- o Sélectionnez dans « Environnement » la variable « dev », puis cliquez sur « Build »

- Le résultat du build devra retourner un output similaire à la capture d'écran ci-dessous :



1.4 Décommenter les stages « Test Playbook » et « Run Playbook » dans les fichiers JenkinsFile :

- Allez à https://github.fr.world.socgen/SGF/Install_Webapp.git
- Allez aux fichiers JenkinsFileBackend, JenkinsFileFrontend puis JenkinsFileFatjar
- Cliquez sur l'icône crayon :



- Décommenter les stages « Test Playbook » et « Run Playbook » en virant les « /* » et « */ ».