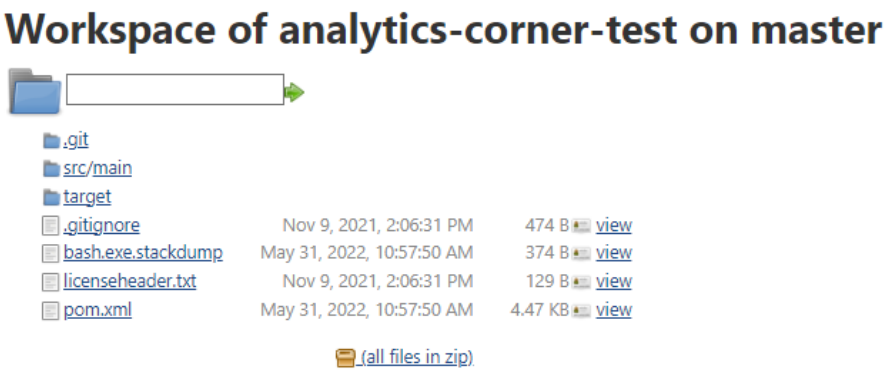
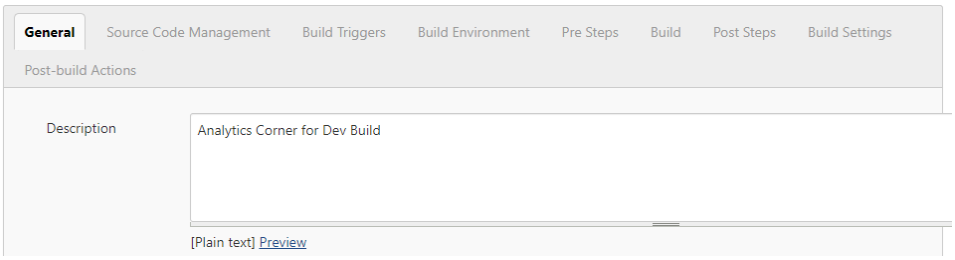
Jenkins-Maven Project Job

*) **Workspace**



***) Description**

Analytics Corner for Dev Build



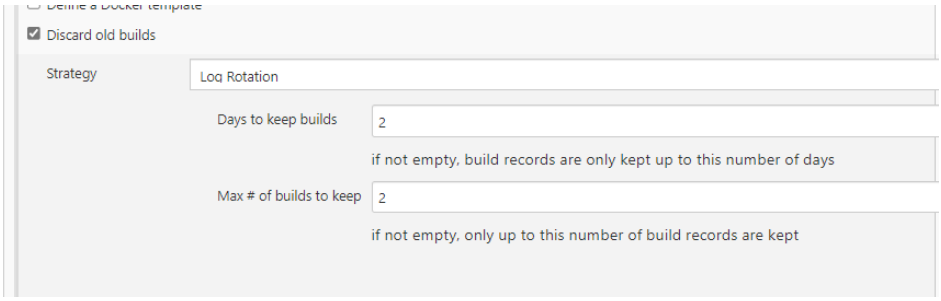
*) **Discard old builds**

Strategy

Log Rotation

Days to keep build: 2

Max # of builds to keep: 2

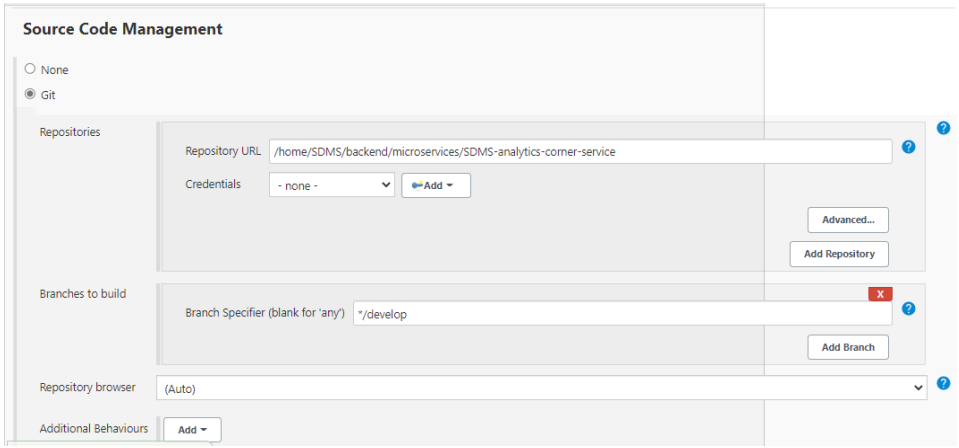


*) **Source Code Management**

Git-Repositories-Repository URL

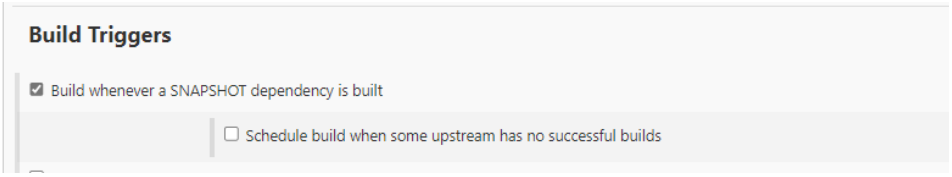
Branches to build */develop

Repository browser: (Auto)



*) **Build Triggers**

Build whenever a SNAPSHOT dependency is Built



*)**Build**

Root POM

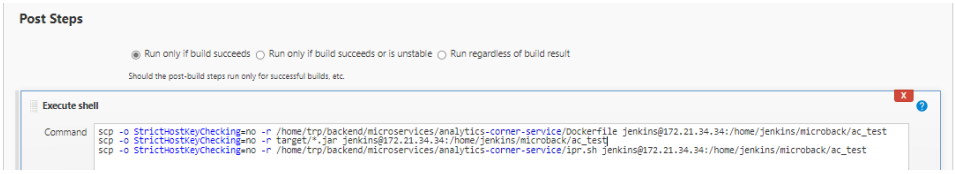
pom.xml

Goals and options

clean install package

*) **Post Steps-Run only if build succeeds-Execute Shell-Command**

Above image does not have the command to send the Dockerfile from jenkins hosted server to destination server



Send the docker file to a particular directory from local jenkins server to VM

scp -o StrictHostKeyChecking=no -r /home/trp/backend/microservices/analytics-corner-service/Dockerfile jenkins@172.21.34.34:/home/jenkins/microback/ac_dev

Send the artifact to a particular directory from jenkins workspace to VM

scp -o StrictHostKeyChecking=no -r target/*.jar jenkins@172.21.34.34:/home/jenkins/microback/ac_dev

Send ipr.sh file from local jenkins server to VM

scp -o StrictHostKeyChecking=no -r /home/trp/backend/microservices/analytics-corner-service/ipr.sh jenkins@172.21.34.34:/home/jenkins/microback/ac_dev

Actual Command

scp -o StrictHostKeyChecking=no -r /home/trp/backend/microservices/analytics-corner-service/Dockerfile jenkins@172.21.34.34:/home/jenkins/microback/ac_dev

scp -o StrictHostKeyChecking=no -r target/*.jar jenkins@172.21.34.34:/home/jenkins/microback/ac_dev

scp -o StrictHostKeyChecking=no -r /home/trp/backend/microservices/analytics-corner-service/ipr.sh jenkins@172.21.34.34:/home/jenkins/microback/ac_dev

***) Post-build Actions**

Send build artifacts over SSH-SSH Publisher-SSH Server

Name: TCSINOMICS04

Transfer-Exec command

cd microback/ac_dev // Go to the directory

chmod u+x ipr.sh // Give execute permission to the user

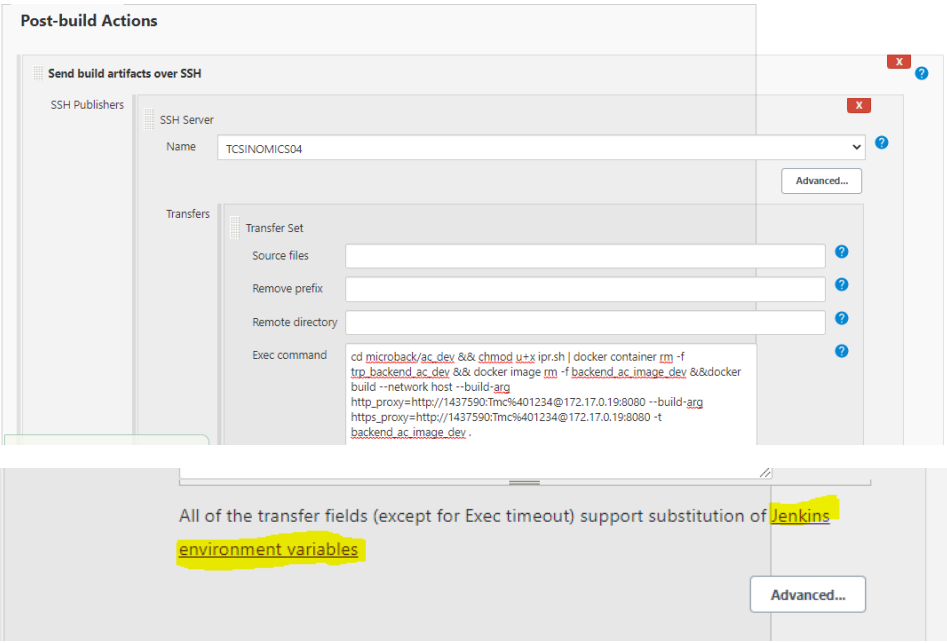
docker container rm -f trp_backend_ac_dev // Force remove earlier running container

docker image rm -f backend_ac_image_dev // Force remove earlier created image

docker build --network host --build-arg http_proxy=http://1437590:Tmc%401234@172.17.0.19:8080 --build-arg https_proxy=http://1437590:Tmc%401234@172.17.0.19:8080 -t backend_ac_image_dev .
// Build a new image, and networking option as host

Actual Command

cd microback/ac_dev && chmod u+x ipr.sh | docker container rm -f trp_backend_ac_dev && docker image rm -f backend_ac_image_dev &&docker build --network host --build-arg http_proxy=http://1437590:Tmc%401234@172.17.0.19:8080 --build-arg https_proxy=http://1437590:Tmc%401234@172.17.0.19:8080 -t backend_ac_image_dev .

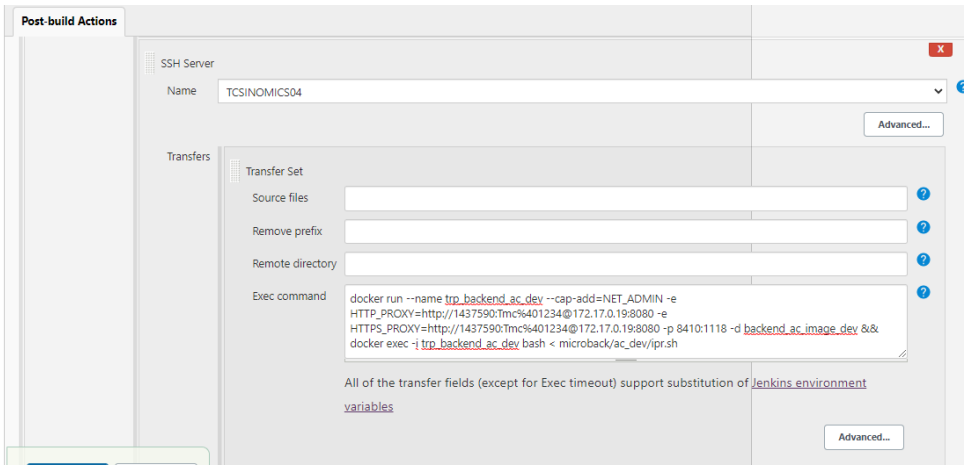


Name: TCSNOMICS04

Transfers-Exec command

docker run --name trp_backend_ac_dev --cap-add=NET_ADMIN -e HTTP_PROXY=http://1437590:Tmc%401234@172.17.0.19:8080 -e HTTPS_PROXY=http://1437590:Tmc%401234@172.17.0.19:8080 -p 8410:1118 -d backend_ac_image_dev // Run the newly created image to be a container

docker exec -i trp_backend_ac_dev bash < microback/ac_dev/ipr.sh



Actual Command

docker run --name trp_backend_ac_dev --cap-add=NET_ADMIN -e HTTP_PROXY=http://1437590:Tmc%401234@172.17.0.19:8080 -e HTTPS_PROXY=http://1437590:Tmc%401234@172.17.0.19:8080 -p 8410:1118 -d backend_ac_image_dev && docker exec -i trp_backend_ac_dev bash < microback/ac_dev/ipr.sh

****

Dockerfile and ipr for dev

FROM java:8

WORKDIR /

COPY ./analytics*.jar ac.jar

EXPOSE 1118

#CMD java -Xmx1024m -jar es.jar

CMD java -Xmx256m -Dhttps.proxyUser=1437590 -Dhttps.proxyPassword="Tmc@1234" -Dhttp.proxyUser=1437590 -Dhttp.proxyPassword="Tmc@1234" -Dhttps.proxyHost=172.17.0.19 -Dhttps.proxyPort=8080 -Dhttp.proxyHost=172.17.0.19 -Dhttp.proxyPort=8080 -Dhttp.nonProxyHosts="localhost|127.0.0.1|10.*.*.*|172.21.34.*" -jar ac.jar

```
[root@TCSINOMICS01 analytics-corner-service]# ls
Dockerfile  ipr.sh  readme
[root@TCSINOMICS01 analytics-corner-service]# cat Dockerfile
FROM java:8
WORKDIR /
COPY ./analytics*.jar ac.jar
EXPOSE 1118
#CMD java -Xmx1024m -jar es.jar
CMD java -Xmx256m -Dhttps.proxyUser=1437590 -Dhttps.proxyPassword="Tmc@1234" -Dhttp.proxyUser=1437590 -Dhttp.proxyPassword="Tmc@1234" -D
https.proxyHost=172.17.0.19 -Dhttps.proxyPort=8080 -Dhttp.proxyHost=172.17.0.19 -Dhttp.proxyPort=8080 -Dhttp.nonProxyHosts="localhost|12
7.0.0.1|10.*.*.*|172.21.34.*" -jar ac.jar
[root@TCSINOMICS01 analytics-corner-service]# vi Dockerfile
[root@TCSINOMICS01 analytics-corner-service]#
```

cat ipr.sh--> ip route add 172.17.0.19/32 via 172.17.0.1 dev eth0

```
[root@TCSINOMICS01 analytics-corner-service]# cat ipr.sh
ip route add 172.17.0.19/32 via 172.17.0.1 dev eth0
```