

Hacker Rank - Proactive DevOps Engineer - Fabian Alejandro Rodriguez Peñuela

1. Installation tools

- a. Create docker images with necessary components
 - i. docker-images/jenkins-master-node.dockerfile
 - ii. docker-images/jenkins-linux-node.dockerfile

```
jenkins-master-node.dockerfile U X
docker > jenkins-master-node.dockerfile
22 RUN /usr/local/bin/install-plugins.sh ws-cleanup
23 RUN /usr/local/bin/install-plugins.sh remote-file
24 RUN /usr/local/bin/install-plugins.sh kiwanjenkinsPlugin
25 RUN /usr/local/bin/install-plugins.sh sonar
26
27 # Plugins for better UX (not mandatory)
28 RUN /usr/local/bin/install-plugins.sh ansicolor
29 RUN /usr/local/bin/install-plugins.sh greenballs
30
31 # Plugin for scaling Jenkins agents
32 RUN /usr/local/bin/install-plugins.sh kubernetes
33
34
35 #nodejs and tools
36 RUN apt update -y
37 RUN apt-get install curl -y
38 RUN apt install zip -y
39 RUN apt install unzip -y
40 RUN apt install tar -y
41 RUN curl -sL https://deb.nodesource.com/setup_14.x | bash -
42 RUN apt-get install -y nodejs
43
44 #kubernetes
45 RUN apt-get update && apt-get install -y apt-transport-https gnupg2 curl
46 RUN curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | apt-key add -
47 RUN echo "deb https://apt.kubernetes.io/ kubernetes-xenial main" | tee -a /etc/apt/sources.list.d/kubernetes.list
48 RUN apt-get update
49 RUN apt-get install -y kubectl
50 RUN mkdir /root/.kube
51 # COPY config /root/.kube/
52 COPY .npmrc /root/
53
54 #install gcp
55 RUN echo "deb [signed-by=/usr/share/keyrings/cloud.google.gpg] https://packages.cloud.google.com/apt cloud-sdk main" | tee -a /etc/apt/sources.list.d/google-cloud-sdk.list
56 RUN apt-get install apt-transport-https ca-certificates gnupg
57 RUN curl https://packages.cloud.google.com/apt/doc/apt-key.gpg | apt-key --keyring /usr/share/keyrings/cloud.google.gpg add -
58 RUN apt-get update && apt-get install google-cloud-sdk
59
60 #USER jenkins
```

- b. Build and publish images

The screenshot shows the Docker Hub profile for user 'fabianrp7'. The profile lists four public repositories:

Repository	Status	Stars	Downloads	Visibility
fabianrp7 / timeoff	Not Scanned	0	59	Public
fabianrp7 / jenkins-master-custom	Not Scanned	0	5	Public
fabianrp7 / jenkins-master-node	Not Scanned	0	139	Public
fabianrp7 / jenkins-nodejs-agent	Not Scanned	0	259	Public

- c. Deploy Jenkins, nexus and Docker In Docker, in Kubernetes
 - i. k8s-deplyments/docker-in-docker-deployment.yaml
 - ii. k8s-deplyments/jenkins-deployment.yaml
 - iii. k8s-deplyments/nexus-deployment.yaml

```
PS C:\Users\fabian.rodriquez\documents\Fabian\repos> kubectl get all -o wide -n gorilla-logic
```

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED	NODE	READINESS	GATES
pod/dood	1/1	Running	0	37h	10.8.1.10	gke-cluster-gorilla-logic-default-pool-bbc01338-lzc1	<none>	<none>	<none>	<none>
pod/jenkins-6c94c8b7f5-pm686	0/1	ContainerCreating	0	18h	<none>	gke-cluster-gorilla-logic-default-pool-bbc01338-d1h3	<none>	<none>	<none>	<none>
pod/jenkins-86657f8c9f-jznd7	1/1	Running	0	2d	10.8.1.8	gke-cluster-gorilla-logic-default-pool-bbc01338-lzc1	<none>	<none>	<none>	<none>
pod/nexus-757b976dcd-sv555	1/1	Running	0	42m	10.8.2.3	gke-cluster-gorilla-logic-default-pool-bbc01338-d1h3	<none>	<none>	<none>	<none>

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE	SELECTOR
service/jenkins	NodePort	10.12.15.120	<none>	8080:30700/TCP	3d13h	app=jenkins
service/jenkins-jnlp	ClusterIP	10.12.1.44	<none>	50000/TCP	3d8h	app=jenkins
service/nexus	NodePort	10.12.7.170	<none>	8081:30707/TCP	3d13h	app=nexus

NAME	READY	UP-TO-DATE	AVAILABLE	AGE	CONTAINERS	IMAGES	SELECTOR
deployment.apps/jenkins	1/1	1	1	3d13h	jenkins	fabianrp7/jenkins-master-node:latest	app=jenkins
deployment.apps/nexus	1/1	1	1	44m	nexus	sonatype/nexus3:latest	app=nexus

NAME	DESIRED	CURRENT	READY	AGE	CONTAINERS	IMAGES	SELECTOR
replicaset.apps/jenkins-675cd49dc5	0	0	0	3d13h	jenkins	fabianrp7/jenkins-master-node:latest	app=jenkins, pod-template-hash=675cd49dc5
replicaset.apps/jenkins-6c94c8b7f5	1	1	0	18h	jenkins	fabianrp7/jenkins-master-node:latest	app=jenkins, pod-template-hash=6c94c8b7f5
replicaset.apps/jenkins-86657f8c9f	1	1	1	3d13h	jenkins	fabianrp7/jenkins-master-node:latest	app=jenkins, pod-template-hash=86657f8c9f
replicaset.apps/nexus-8dcfc6b786	0	0	0	44m	nexus	sonatype/nexus3:latest	app=nexus, pod-template-hash=8dcfc6b786
replicaset.apps/nexus-757b976dcd	1	1	1	42m	nexus	sonatype/nexus3:latest	app=nexus, pod-template-hash=757b976dcd

```
PS C:\Users\fabian.rodriquez\documents\Fabian\repos> |
```

- d. Jenkins kubernetes cloud and nodejs-agent configuration

←

→

↺

⚠ Not secure

34.122.90.105:30700/configureClouds/

Dashboard > Configure Clouds

Pod Templates

Pod Template

Name 1

nodejs-agent

Namespace 2

gorilla-logic

Labels 3

nodejs-agent

Usage 4

Only build jobs with label expressions matching this node

Pod template to inherit from 5

Containers 6

Container Template

Name 7

jnlp

Docker image 8

fabianrp7/jenkins-nodejs-agent

☒ Always pull image

Working directory 9

/home/jenkins/agent

Command to run 10

Save

Apply

e. Repo <https://github.com/fabianrp7/timeoff-mgmt-app> (Webhook configuration)

The screenshot shows the GitHub repository page for `fabianrp7/timeoff-mgmt-app`. The repository is public and has 2 branches and 20 tags. The main branch is `develop`. The repository was created by Fabian Alejandro Rodriguez Peña Solté fix, who has 1,071 commits. The repository contains a file tree with the following files and their last commit dates:

File	Last Commit
<code>bin</code>	Refactor carry over logic into reusable function (3 years ago)
<code>config</code>	Introduce redis as alternative session storage. (4 months ago)
<code>docs</code>	Introduce redis as alternative session storage. (4 months ago)
<code>lib</code>	Introduce redis as alternative session storage. (4 months ago)
<code>migrations</code>	Migration to compress email audit data (2 years ago)
<code>public</code>	Prevent double submission from browser (5 months ago)
<code>scss</code>	Expose leave details as a popup (13 months ago)
<code>t</code>	Prevent double submission from browser (5 months ago)
<code>views</code>	Prevent double submission from browser (5 months ago)
<code>.codeclimate.yml</code>	Discontinue node versions <6 (3 years ago)
<code>.gitignore</code>	Auto approval flag feature. (5 years ago)
<code>.npmrc</code>	Jenkinsfile and nexus configuration (6 hours ago)
<code>.travis.yml</code>	Upgrade version of NODEJS (2 years ago)

The right sidebar shows the repository's metadata, including the README, MIT License, 0 stars, 1 watching, and 0 forks. It also shows the Releases section with 20 tags and the Packages section with no published packages. The Contributors section shows 23 contributors.

The screenshot shows the GitHub Webhooks configuration page for the repository `fabianrp7/timeoff-mgmt-app`. The page is titled "Webhooks / Manage webhook" and has a left sidebar with the following sections:

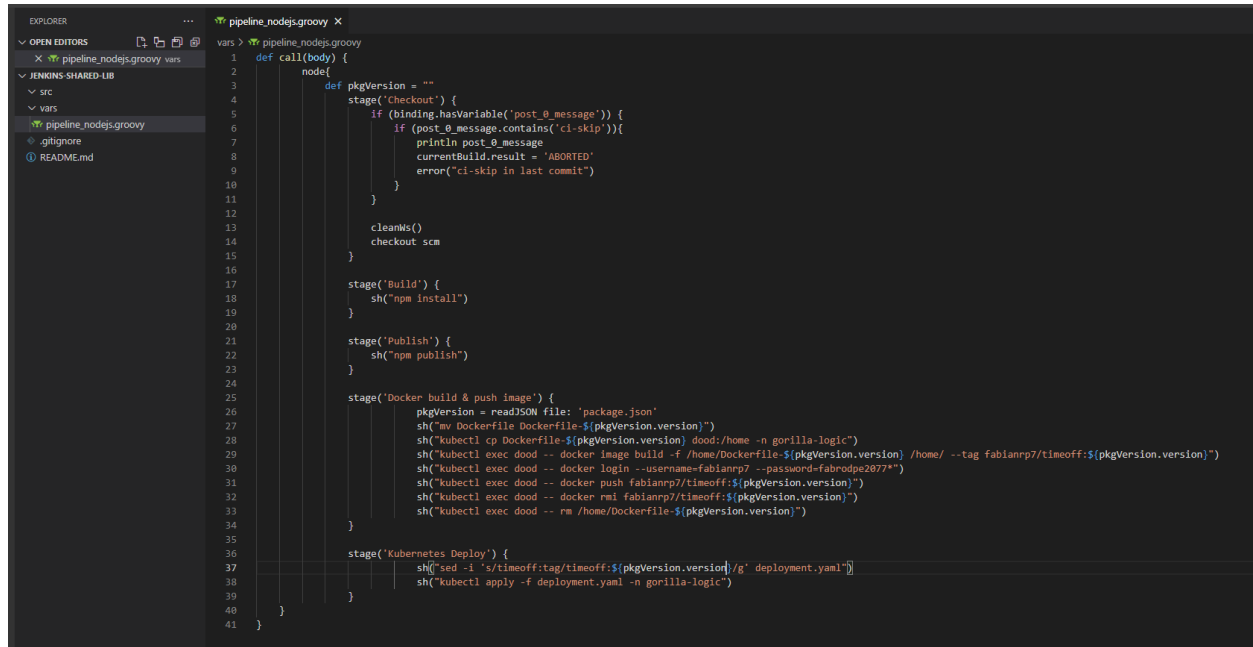
- General
- Access
 - Collaborators
 - Moderation options
- Code and automation
 - Branches
 - Actions
 - Webhooks**
 - Environments
 - Pages
- Security
 - Code security and analysis
 - Deploy keys
 - Secrets
- Integrations
 - Integrated apps
 - Email notifications

The main content area shows the configuration for a new webhook:

- Payload URL:** `http://34.122.90.105:30700/generic-webhook-trigger/invoke`
- Content type:** `application/json`
- Secret:** (empty field)
- Which events would you like to trigger this webhook?**
 - ☒ Just the push event.
 - ☐ Send me everything.
 - ☐ Let me select individual events.
- Active:** ☒ (We will deliver event details when this hook is triggered.)

At the bottom, there are two buttons: "Update webhook" and "Delete webhook".

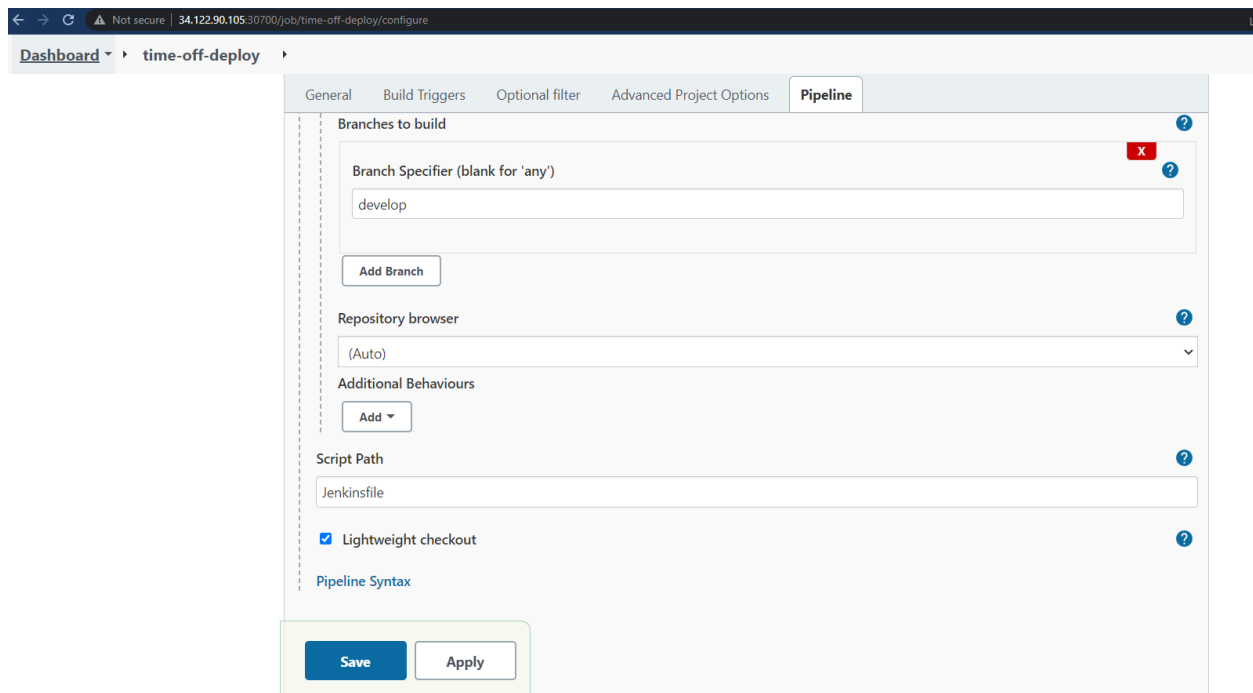
f. Jenkins shared lib creation <https://github.com/fabianrp7/jenkins-shared-lib>



The screenshot shows a Jenkins shared library script named `pipeline_nodes.groovy` in a code editor. The script defines a `call` function that takes a `body` block. Inside the function, it defines a `pkgVersion` variable and a `stage` named 'Checkout'. The 'Checkout' stage contains a conditional check for a post-build message containing 'ci-skip'. If it does, the build is aborted. Otherwise, it performs a checkout from the 'scm' provider. Following the checkout, there are stages for 'Build' (running `npm install`) and 'Publish' (running `npm publish`). The final stage is 'Docker build & push image', which reads the `package.json` file, builds a Docker image, logs into Docker Hub, pushes the image, and removes the local image. The script concludes with a 'Kubernetes Deploy' stage that applies a deployment manifest.

```
1 def call(body) {
2     node {
3         def pkgVersion = ""
4         stage('Checkout') {
5             if (binding.hasVariable('post_build_message')) {
6                 if (post_build_message.contains('ci-skip')) {
7                     println post_build_message
8                     currentBuild.result = 'ABORTED'
9                     error("ci-skip in last commit")
10                }
11            }
12
13            cleanWs()
14            checkout scm
15        }
16
17        stage('Build') {
18            sh("npm install")
19        }
20
21        stage('Publish') {
22            sh("npm publish")
23        }
24
25        stage('Docker build & push image') {
26            pkgVersion = readJSON file: 'package.json'
27            sh("mv Dockerfile Dockerfile-${pkgVersion.version}")
28            sh("kubect1 cp Dockerfile-${pkgVersion.version} dood:/home -n gorilla-logic")
29            sh("kubect1 exec dood -- docker image build -f /home/Dockerfile-${pkgVersion.version} /home/ --tag fabianrp7/timeoff:${pkgVersion.version}")
30            sh("kubect1 exec dood -- docker login --username=fabianrp7 --password=fabrope2077")
31            sh("kubect1 exec dood -- docker push fabianrp7/timeoff:${pkgVersion.version}")
32            sh("kubect1 exec dood -- docker rmi fabianrp7/timeoff:${pkgVersion.version}")
33            sh("kubect1 exec dood -- rm /home/Dockerfile-${pkgVersion.version}")
34        }
35
36        stage('Kubernetes Deploy') {
37            sh("sed -i 's/timeoff:tag/timeoff:${pkgVersion.version}/g' deployment.yaml")
38            sh("kubect1 apply -f deployment.yaml -n gorilla-logic")
39        }
40    }
41 }
```

g. Jenkins pipeline creation



The screenshot displays the Jenkins Pipeline configuration page for a job named 'time-off-deploy'. The 'Pipeline' tab is selected, showing the 'Branches to build' section. The 'Branch Specifier' is set to 'develop'. Below this, the 'Repository browser' is set to '(Auto)'. The 'Additional Behaviours' section has an 'Add' button. The 'Script Path' is set to 'Jenkinsfile'. The 'Lightweight checkout' checkbox is checked. At the bottom, there are 'Save' and 'Apply' buttons.

Dashboard ▾ time-off-deploy ▾

General Build Triggers Optional filter Advanced Project Options **Pipeline**

Branches to build

Branch Specifier (blank for 'any')

develop

Add Branch

Repository browser

(Auto)

Additional Behaviours

Add ▾

Script Path

Jenkinsfile

☒ Lightweight checkout

Pipeline Syntax

Save Apply

GCP require authentication of user through web browser, from each machine that use a config file to connect to GCP clusters, based on that, I decided to use the master node to run the pipeline instead of on-demand nodes-agent

```
PS C:\Users\fabian.rodriguez\documents\Fabian\repos> kubectl exec -it nodes-agent-ds2h3 -- /bin/sh
# kubectl get all -o wide -n gorilla-logic
error: You must be logged in to the server (Unauthorized)
# cd /root/.kube/
# cat /root/.kube/config
apiVersion: v1
clusters:
- cluster:
    certificate-authority-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURSB0tLS0tCk13UUVVNDQ0XzI0F3SUJBZ0tLSQUL4dHZxTFphczZQTzS9vQU0waErnNmd3RFFZSktyVWkLodmNOQVFFTEJRQXcKTHpFdE1De0dBWVVF
8xUZzVZVEF8WVRnelkyRTB0akpsTjJ0aApNQ0FYRFRJeU1ESXh0akEwTURBeU15b1Eekl3LRJd01qQTVNRFRV3TURJevdqqXZNUz83S3d3RFRZRUURFeVE1Cl6VmpabVptTLXxbVL6YzRRVFEtTm1Rde9EBghNQzF0tR0aallUu
FFQKFRVUFBNELCand0d2dnR0t8b0lCZ1FERWS0b2NGbnNGT3oxM9MVWZVNFZDVHgyekgVb2RrXQzRSBfV6WgoS31phdWfNu1ppRW110HJSWnRhcx4NzdKMMR4L8dzMuq2NjBTU8IvOV8veXXQW9CSF16vY4cEUvZGphdUZOCn
5RXFTN1VO283ldmhBk319cncB8QnJWcmxSNDVsbmNHzmpBM3EKVVTZTalg3QSt8amVxd1ZuS213TGszcXhEQ089PS85Kd0LhSnM5UHRtdEZuSdd1RGLZRmcvSHJTCkxHSDgybDZzTqPIUwLLYXhNjJlkcS9RUTJZdGd50VdyQ18zZLh
eC9nVUZH25Cc1ocGpnZutnRW1KSPNHZ3pDN1YyUVRkNFk4OHZVT1Nxs04zKzVCTGhSVy9ZWDEraVpCn0LsoFRvRStOMjIiek9kbEpTunJYakV5a8N00GNWZnLqMitwmtjZzhYVlNMcnB1WFBvOTHYd9QtzFJm0N1TDJwclFV
HhkCetSQ25yMwo1V2xhwS81V0FZSm9jY3VSQ0JZaxdMT8NnQmtk52hzTwdGVHNIclpmbmozRHJDNDM2VLRBNFdiChdFcWNCQ085xTUNBd0VBQWFO001FOXDEZ1LEVLIwUEFRSC9CQVFEQndJR1B0EcKQTFVZE3RUIvd1FGTUFNQ
pJNjRyM2JndzZrZkc1b1dyTUEwRwpDU3FHU0liM0RRRUJDd1VBQTRJQmdRQVZ0VHl0Q0tHOVpaL3hwZ3JmY1BjbmFVTLB1MlZKL1V4N28wSGpPMmt6CkJraGxZbFV3Rlh0RDJCcwszTG9aZU83VvhlQ3NuMWNPMVhVdEtoEDNFM
4b09TeSt1K2wrZ1V5dFVhQ081ek5C1fPb0RGAjZnc3ZWakVrcy9VbHdmBfO2cVdOb3p5cU2uSG9uNQpvsjg2OXFOVnRwc3pxR3VkvVHR2dmtLWGJJawtQ50Iwc0JlWldzV21CSjhHdT0T1ZgbVp1dEZRZE2d2d3NlcU5UcNBVwLB
eDhpSE5LdWd0MmmlVGt8dFyZ0Ho3UH0pSyteGgNPbjQKNDljdk5nallZSHhpcDNGNTd2W604MLZvYnZzRGkxM0oveURhk1Btu3RPRIFMNm0MDFBwNTN1hKS0JXWEd6agpsVupkdVBvanIrU29KWENUbTJfdFrl1QwRwXrdUd
1hXQk1icM08M1ZTCwRTSzdMU0h0K0Fqegh3UUFramvsbHhol2pZnNlXWXA2cGVUTW5DZk5jNfOxWXJpT3hTaUzVNU9MVVKKUwocE5vb1BFbGcxYlplZGl1N3BpQk09Ci0tLS0tRU5EIEFUFURJRKLDQVRFLS0tLS0K
server: https://34.72.172.36
name: gke_gorilla-logic-devops_us-central1-c_cluster-gorilla-logic
contexts:
- context:
    cluster: gke_gorilla-logic-devops_us-central1-c_cluster-gorilla-logic
    user: gke_gorilla-logic-devops_us-central1-c_cluster-gorilla-logic
    name: gke_gorilla-logic-devops_us-central1-c_cluster-gorilla-logic
current-context: gke_gorilla-logic-devops_us-central1-c_cluster-gorilla-logic
kind: Config
preferences: {}
users:
- name: gke_gorilla-logic-devops_us-central1-c_cluster-gorilla-logic
  user:
    auth-provider:
      config:
        cmd-args: config config-helper --format=json
        cmd-path: /usr/lib/google-cloud-sdk/bin/gcloud
        expiry-key: '{.credential.token_expiry}'
        token-key: '{.credential.access_token}'
      name: gcp
# kubectl get all
error: You must be logged in to the server (Unauthorized)
# gcloud container clusters get-credentials cluster-gorilla-logic --zone us-central1-c --project gorilla-logic-devops
Fetching cluster endpoint and auth data.
ERROR: (gcloud.container.clusters.get-credentials) ResponseError: code=403, message=Request had insufficient authentication scopes.

If you are in a compute engine VM, it is likely that the specified scopes during VM creation are not enough to run this command.
See https://cloud.google.com/compute/docs/access/service-accounts#accessscopesiam for more information about access scopes.
See https://cloud.google.com/compute/docs/access/create-enable-service-accounts-for-instances#changeserviceaccountandscope for how to update access scopes of the VM.
# exit
command terminated with exit code 1
```