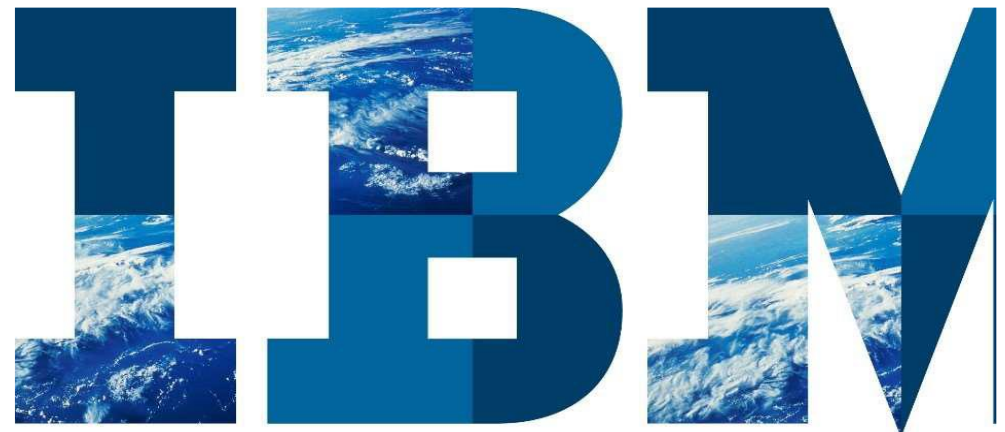




IBM APP CONNECT & OPENSIFT PIPELINE (TEKTON) - INTEGRACIÓN

DOCUMENTO DE INSTALACIÓN,
CONFIGURACIÓN & MANEJO

Versión 1.0



Contact Detail IBM	Cesar Guerra - cesar.guerra@ibm.com
Fecha	Septiembre 2023



ÍNDICE:

I. INTRODUCCIÓN.....3

II. INSTALACIÓN: *OPENSIFT PIPELINES*.....5

III. INSTALACIÓN: *TEKTON DASHBOARD*.....9

IV. DEMO: *INTEGRACIÓN*.....13

IBM APP CONNECT & OPENSIFT PIPELINES (TEKTON) - INTEGRACIÓN

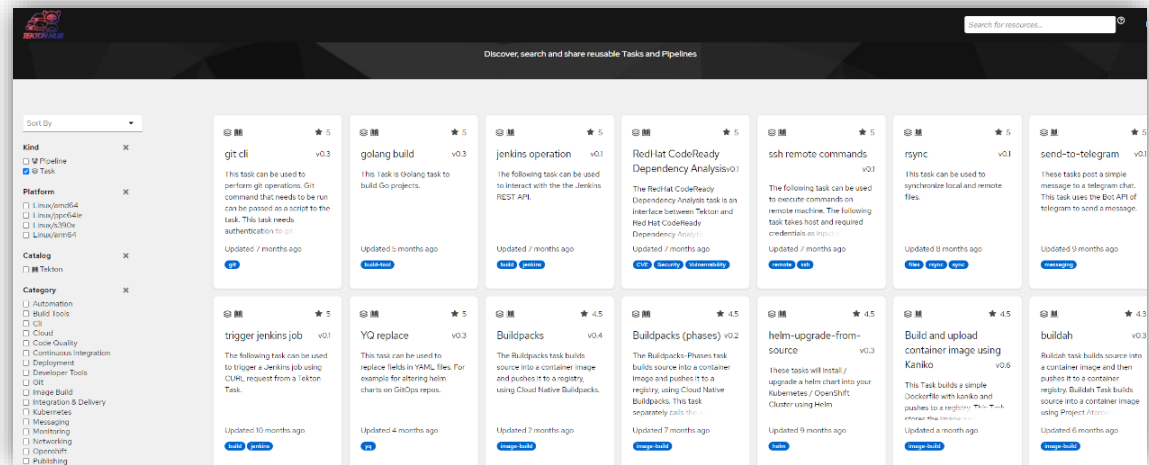


El presente documento abarca la **INTEGRACIÓN** de **IBM APP CONNECT 12** con **OPENSIFT PIPELINES (TEKTON)**:

DESCRIPCIÓN	DETALLE																
1. INTRODUCCIÓN:																	
<p>✓ TEKTON es una capacidad de infraestructura nativa de KUBERNETES, para la creación de sistemas de integración & entrega/despliegue continuo (CI/CD). Así mismo, ayuda a modernizar dicha continuidad ofreciendo especificaciones de la industria para el manejo de: pipelines, en múltiples entorno: On-Cloud, On-Premise e Híbrido, supervisados por un panel de control integrado (DASHBOARD).</p> <p>✓ OPENSIFT PIPELINES es una capacidad de OPENSIFT basada en TEKTON, para proporcionar una experiencia de CI/CD, a través de una excelente integración con las herramientas de OPENSIFT (RECURSOS).</p>																	
<p>En TEKTON se manejan varios CONCEPTOS asociados a sus RECURSOS, que son utilizados en KUBERNETES durante el ciclo de vida de ejecución, estos son:</p>	<table> <tr> <td>STEP:</td><td>Es una operación independiente del flujo de CI/CD, donde en cada STEP se puede correr una IMAGEN contenerizada.</td></tr> <tr> <td>TASK</td><td>Es una colección de STEPS en orden, donde cada TASK se representa como un POD. <ul style="list-style-type: none"> ▪ <i>Compilar código.</i> ▪ <i>Ejecutar test.</i> ▪ <i>Crear imagen de contenedor.</i> </td></tr> <tr> <td>TASK-RUN</td><td>Es el recurso asociado a la acción de ejecutar & mostrar el resultado una TASK.</td></tr> <tr> <td>PIPELINE</td><td>Es una colección de TASKs en orden.</td></tr> <tr> <td>PIPELINE-RUN</td><td>Es el recurso asociado a la acción de ejecutar & mostrar el resultado una PIPELINE.</td></tr> <tr> <td>PIPELINE -RESOURCE</td><td>Este recurso define un objeto de ENTRADA (<i>como un repositorio de Git</i>) o una SALIDA (<i>como una imagen de Docker</i>) del PIPELINE.</td></tr> <tr> <td>PARAMs</td><td>Es la información que requiere un PIPELINE o TASK para su ejecución.</td></tr> <tr> <td>WORKSPACES</td><td>Es un FileSystem utilizado en RUNTIME (TASK-RUN), para compartir información entre TASKs.</td></tr> </table>	STEP:	Es una operación independiente del flujo de CI/CD , donde en cada STEP se puede correr una IMAGEN contenerizada.	TASK	Es una colección de STEPS en orden, donde cada TASK se representa como un POD . <ul style="list-style-type: none"> ▪ <i>Compilar código.</i> ▪ <i>Ejecutar test.</i> ▪ <i>Crear imagen de contenedor.</i> 	TASK-RUN	Es el recurso asociado a la acción de ejecutar & mostrar el resultado una TASK .	PIPELINE	Es una colección de TASKs en orden.	PIPELINE-RUN	Es el recurso asociado a la acción de ejecutar & mostrar el resultado una PIPELINE .	PIPELINE -RESOURCE	Este recurso define un objeto de ENTRADA (<i>como un repositorio de Git</i>) o una SALIDA (<i>como una imagen de Docker</i>) del PIPELINE .	PARAMs	Es la información que requiere un PIPELINE o TASK para su ejecución.	WORKSPACES	Es un FileSystem utilizado en RUNTIME (TASK-RUN) , para compartir información entre TASKs .
STEP:	Es una operación independiente del flujo de CI/CD , donde en cada STEP se puede correr una IMAGEN contenerizada.																
TASK	Es una colección de STEPS en orden, donde cada TASK se representa como un POD . <ul style="list-style-type: none"> ▪ <i>Compilar código.</i> ▪ <i>Ejecutar test.</i> ▪ <i>Crear imagen de contenedor.</i> 																
TASK-RUN	Es el recurso asociado a la acción de ejecutar & mostrar el resultado una TASK .																
PIPELINE	Es una colección de TASKs en orden.																
PIPELINE-RUN	Es el recurso asociado a la acción de ejecutar & mostrar el resultado una PIPELINE .																
PIPELINE -RESOURCE	Este recurso define un objeto de ENTRADA (<i>como un repositorio de Git</i>) o una SALIDA (<i>como una imagen de Docker</i>) del PIPELINE .																
PARAMs	Es la información que requiere un PIPELINE o TASK para su ejecución.																
WORKSPACES	Es un FileSystem utilizado en RUNTIME (TASK-RUN) , para compartir información entre TASKs .																

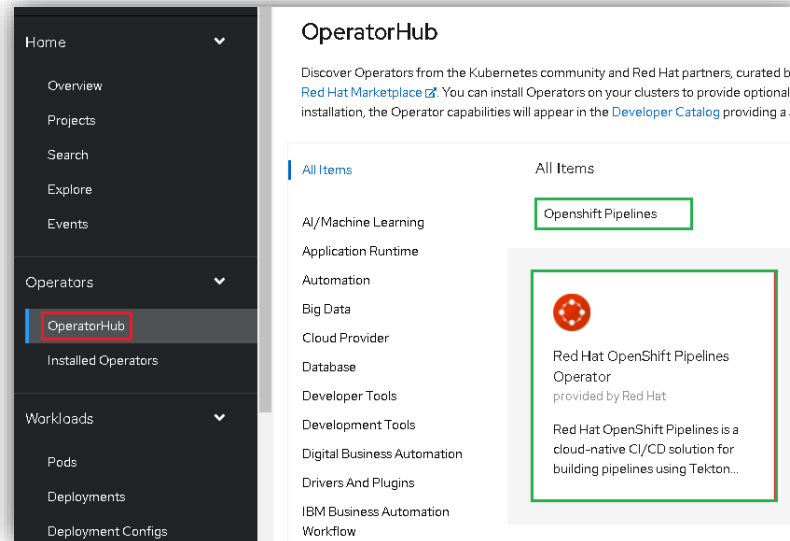
	<table border="1"> <tr> <td>EVENT-LISTENER</td><td>Este encargará de escuchar los EVENTOS automáticos, recibir las PETICIONES & llamar a los TRIGGERS.</td></tr> <tr> <td>TRIGGER</td><td>Este define lo que se ejecutará cuando el recurso: EVENT-LISTENER detecta un evento. Así mismo, internamente hace referencia e invoca a los recursos: TRIGGER-TEMPLATE, TRIGGER-BINDING, opcionalmente a: INTERCEPTOR.</td></tr> <tr> <td>TRIGGER-TEMPLATE</td><td>Es utilizado para crear RECURSOS como PIPELINE-RUNS & TASK-RUNS.</td></tr> <tr> <td>TRIGGER-BINDING</td><td>Es utilizado para capturar los campos (PAYLOAD), de un EVENTO & guardarlo como PARÁMETRO.</td></tr> <tr> <td>INTERCEPTOR</td><td>Permite realizar un filtrado del WORKLOAD, para la verificación, definir, probar condiciones & otros procesos útiles. Modifican el comportamiento de TRIGGERS.</td></tr> </table>	EVENT-LISTENER	Este encargará de escuchar los EVENTOS automáticos, recibir las PETICIONES & llamar a los TRIGGERS .	TRIGGER	Este define lo que se ejecutará cuando el recurso: EVENT-LISTENER detecta un evento. Así mismo, internamente hace referencia e invoca a los recursos: TRIGGER-TEMPLATE , TRIGGER-BINDING , opcionalmente a: INTERCEPTOR .	TRIGGER-TEMPLATE	Es utilizado para crear RECURSOS como PIPELINE-RUNS & TASK-RUNS .	TRIGGER-BINDING	Es utilizado para capturar los campos (PAYLOAD), de un EVENTO & guardarlo como PARÁMETRO .	INTERCEPTOR	Permite realizar un filtrado del WORKLOAD , para la verificación, definir, probar condiciones & otros procesos útiles. Modifican el comportamiento de TRIGGERS .
EVENT-LISTENER	Este encargará de escuchar los EVENTOS automáticos, recibir las PETICIONES & llamar a los TRIGGERS .										
TRIGGER	Este define lo que se ejecutará cuando el recurso: EVENT-LISTENER detecta un evento. Así mismo, internamente hace referencia e invoca a los recursos: TRIGGER-TEMPLATE , TRIGGER-BINDING , opcionalmente a: INTERCEPTOR .										
TRIGGER-TEMPLATE	Es utilizado para crear RECURSOS como PIPELINE-RUNS & TASK-RUNS .										
TRIGGER-BINDING	Es utilizado para capturar los campos (PAYLOAD), de un EVENTO & guardarlo como PARÁMETRO .										
INTERCEPTOR	Permite realizar un filtrado del WORKLOAD , para la verificación, definir, probar condiciones & otros procesos útiles. Modifican el comportamiento de TRIGGERS .										
<p>Luego, en las IMÁGENES se muestra la relación entre los RECURSOS de tipo: TASK, PIPELINE, PIPELINE-RUN, TASK-RUN asociados a los PODs. Así como la secuencia de STEPS que se maneja a nivel de: TASKs.</p>	<div> </div>										

Finalmente, vale mencionar la existencia de: **TEKTON-HUB**, donde se podrá encontrar un gran listado con la **DOCUMENTACIÓN** de **TASK** predefinidas & listas para ser utilizadas: <https://hub.tekton.dev/?kind=Task>



2. INSTALACIÓN: **OPENSIFT PIPELINES**


Ingresar en **OPERATOR-HUB** & filtrar por: **Openshift Pipelines**.



Luego, desde aquí antes de **INSTALAR**, se requiere **DESCARGAR** el cliente de **TEKTON**.

Otra **OPCIÓN** es realizar la **DESCARGA** desde aquí:

https://docs.openshift.com/container-platform/4.10/cli_reference/tkn_cli/installing-tkn.html



Red Hat OpenShift Pipelines
1111 provided by Red Hat

[Install](#)

Infrastructure features
 Disconnected
 Proxy-aware

Valid Subscriptions
 OpenShift Container Platform
 OpenShift Platform Plus

Repository
<https://github.com/tektoncd/operator>

Container image
 N/A

Created at
 N/A

Support
 Red Hat

- Scale pipelines on-demand
- Portable across any Kubernetes platform
- Designed for microservices and decentralized team
- Integrated with OpenShift Developer Console

Installation

Red Hat OpenShift Pipelines Operator gets installed into a single namespace (openshift-operators) which would then install Red Hat OpenShift Pipelines into the openshift-pipelines namespace. Red Hat OpenShift Pipelines is however cluster-wide and can run pipelines created in any namespace.

Components

- Tekton Pipelines: v0.47.4
- Tekton Triggers: v0.24.1
- Pipelines as Code: 0.19.4
- Tekton Chains: v0.16.0
- Tekton Hub (tech-preview): v1.13.0
- Tekton Results (tech-preview): v0.6.0

Getting Started

In order to get familiar with OpenShift Pipelines concepts and create your first pipeline, follow the [OpenShift Pipelines Docs](#).

CLI (tkn)

Tekton Pipelines cli project provides a CLI (tkn) for interacting with OpenShift Pipelines.

[Download tkn](#) links will be available on [command line tools](#) page after this operator is installed).

tkn - OpenShift Pipeline Command Line Interface (CLI)

The OpenShift Pipeline client `tkn` is a CLI tool that allows you to manage OpenShift Pipeline resources.

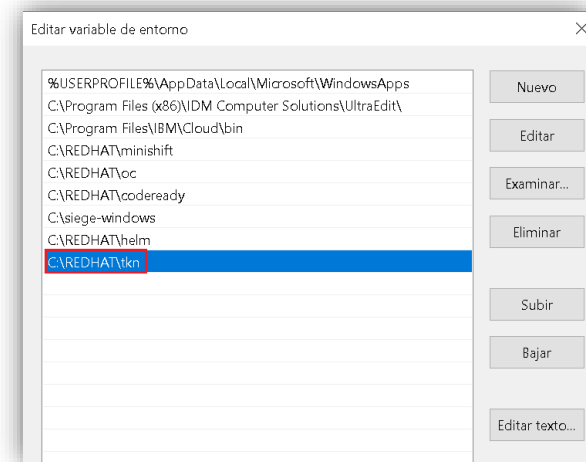
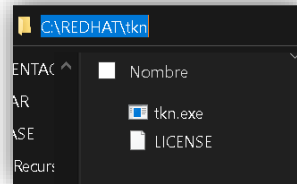
- Download tkn and tkn-pac for Linux x86_64
- Download tkn and tkn-pac for Linux ARM 64
- Download tkn and tkn-pac for IBM Power
- Download tkn and tkn-pac for IBM Z
- Download tkn and tkn-pac for Mac x86_64
- Download tkn and tkn-pac for Mac ARM 64
- Download tkn and tkn-pac for Windows x86_64
- Download tkn and tkn-pac for Windows ARM 64

CLIENTE TEKTON:

En este caso para fines de la presentación, **DESCARGAR** para **WINDOWS** el archivo: **tkn-windows-amd64.zip**, & descomprimir en un **FILESYSTEM** & referenciarlo a nivel de **VARIABLES DE ENTORNO**, como se muestra en **IMAGEN**.

Así mismo, luego de instalar se **VALIDA** el cliente de **TEKTON** instalado:

```
$ tkn version
```



```
GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ tkn version
Client version: 0.31.2
Chains version: v0.16.0
Pipeline version: v0.47.4
Triggers version: v0.24.1
Dashboard version: v0.19.0
Operator version: v0.67.0
```

Luego, regresando a **OPENSIFT**, se procede a **INSTALAR** el **OPERADOR**, aplicando la configuración como se muestra en la **IMAGEN** (*all namespaces*).

IMPORTANTE: "Siempre utilizar la versión mayor del **CHANNEL** existente".

OperatorHub > Operator Installation

Install Operator

Install your Operator by subscribing to one of the update channels to keep the Operator up to date. The strategy determines either manual or automatic updates.

Update channel *

- ☒ latest
- ☐ pipelines-1.10
- ☐ pipelines-1.11
- ☐ pipelines-1.8
- ☐ pipelines-1.9

Installation mode *

- ☒ All namespaces on the cluster (default)
Operator will be available in all Namespaces.
- ☐ A specific namespace on the cluster
This mode is not supported by this Operator

Installed Namespace *

openshift-operators

Update approval *

- ☒ Automatic
- ☐ Manual



Install **Cancel**

Luego, se valida el estado de **OPERADOR** que figure **EXITOSO**.

Project: openshift-operators

Installed Operators

Installed Operators are represented by ClusterServiceVersions within this Namespace. For more information, see the [Understanding Operators documentation](#). Or create an Operator and ClusterServiceVersion using the [Operator SDK](#).

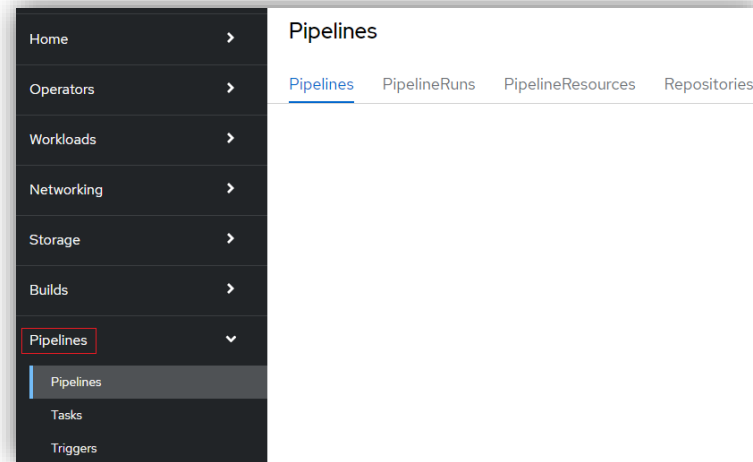
Name	Managed Namespaces	Status	Last updated
 Red Hat OpenShift Pipelines 1.11.1 provided by Red Hat	All Namespaces	 Succeeded Up to date	28 ago. 2023, 14:32

Luego, se valida la existencia del **SECRET** asociado llamado: **pipeline**, si figura quiere decir que ya se tiene los **PERMISOS** para el manejo de **PIPELINES** en **OPENSIFT**:

`$ oc get serviceaccount pipeline`

```
GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc get serviceaccount pipeline
NAME      SECRETS  AGE
pipeline  1        23d
```

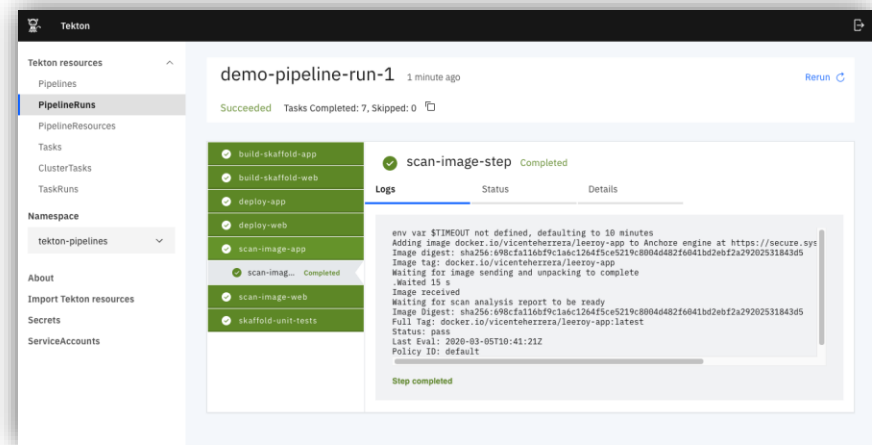

Finalmente, se aprecia a nivel del **MENÚ PRINCIPAL**, que se ha generado un **BLOQUE** nuevo llamado: **PIPELINES** (antes **NO** existía).



3. INSTALACIÓN: TEKTON DASHBOARD

TEKTON-DASHBOARD es una **CONSOLA WEB** con el propósito de la administración & ejecución **RECURSOS: TEKTON**, esta permite lo siguiente:

- Filtrar recursos.
- Vista en runtime de los recursos: **PipelineRun** & **TaskRun**
- Ver detalles de recursos existentes & **YAMLS**.
- Agregar funcionalidad a través de extensiones.



Luego, para la instalación del **DASHBOARD** se deberá realizar los siguientes pasos:

En este punto debe **EXISTIR** ya el **NAMESPACE: openshift-pipelines** (este será creado por el OPERATOR en la instalación).

`$ oc get all -n openshift-pipelines`

- **TEKTON-DASHBOARD:** contiene el **GITHUB** el **script YAML** con los recursos requeridos para la instalación del **DASHBOARD**.



**0_dummy-appconn
ect-tekton-dashboa**

Después, se procede con la **CREACIÓN:**

`$ oc create -f`
https://raw.githubusercontent.com/makutp/dummy-tekton-ace/main/scripts/0_dummy-appconnect-tekton-dashboard.yaml

```

GMY+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc get all -n openshift-pipelines
NAME                                     READY   STATUS    RESTARTS   AGE
pod/pipelines-as-code-controller-6d5df69996-jbtbt  1/1     Running   0           16d
pod/pipelines-as-code-watcher-7fb6d9667-8kp4g     1/1     Running   0           16d
pod/pipelines-as-code-webhook-7b8cdf6448-s9d8c    1/1     Running   0           16d
pod/tekton-chains-controller-745585dd48-wpc5d     1/1     Running   0           16d
pod/tekton-dashboard-ff6f46b77-8s51r             1/1     Running   0           5d2h
pod/tekton-operator-proxy-webhook-7db8979468-twbvk 1/1     Running   0           16d
pod/tekton-pipelines-controller-79fd798df9-5n66t  1/1     Running   0           16d
pod/tekton-pipelines-remote-resolvers-6458b48c46-crbst 1/1     Running   0           16d
pod/tekton-pipelines-webhook-5bfcf4ffd-c67br      1/1     Running   0           16d
pod/tekton-triggers-controller-549db989cb-jnrfn   1/1     Running   0           16d
pod/tekton-triggers-core-interceptors-84467b88d4-kr9hb 1/1     Running   0           16d
pod/tekton-triggers-webhook-6656d65cb4-vg2gq      1/1     Running   0           16d
pod/tnk-cli-serve-68c67fd6d6-m6jsk               1/1     Running   0           16d

NAME                                     TYPE          CLUSTER-IP      EXTERNAL-IP      PORT(S)
service/pipelines-as-code-controller      ClusterIP      172.30.133.141  <none>            8080/TCP,9090/TCP
service/pipelines-as-code-watcher         ClusterIP      172.30.158.40   <none>            9090/TCP
service/pipelines-as-code-webhook         ClusterIP      172.30.36.176   <none>            443/TCP
service/tekton-dashboard                   ClusterIP      172.30.186.35   <none>            9097/TCP
service/tekton-operator-proxy-webhook     ClusterIP      172.30.17.203   <none>            443/TCP
service/tekton-pipelines-controller       ClusterIP      172.30.172.227  <none>            9090/TCP,8080/TCP,8080/TCP
service/tekton-pipelines-webhook          ClusterIP      172.30.191.31   <none>            9090/TCP,8080/TCP,443/TCP,8080/TCP
service/tekton-triggers-controller        ClusterIP      172.30.95.227   <none>            9080/TCP
service/tekton-triggers-core-interceptors ClusterIP      172.30.40.251   <none>            8443/TCP
service/tekton-triggers-webhook           ClusterIP      172.30.211.242  <none>            443/TCP
service/tnk-cli-serve                     ClusterIP      172.30.231.150  <none>            8080/TCP

NAME                                     READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/pipelines-as-code-controller  1/1     1             1           16d
deployment.apps/pipelines-as-code-watcher    1/1     1             1           16d
deployment.apps/pipelines-as-code-webhook     1/1     1             1           16d
deployment.apps/tekton-chains-controller      1/1     1             1           16d
deployment.apps/tekton-dashboard              1/1     1             1           5d2h
deployment.apps/tekton-operator-proxy-webhook 1/1     1             1           16d
deployment.apps/tekton-pipelines-controller  1/1     1             1           16d
deployment.apps/tekton-pipelines-remote-resolvers 1/1     1             1           16d
deployment.apps/tekton-pipelines-webhook      1/1     1             1           16d
deployment.apps/tekton-triggers-controller    1/1     1             1           16d
deployment.apps/tekton-triggers-core-interceptors 1/1     1             1           16d

```

```

GMY+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc create -f https://raw.githubusercontent.com/makutp/dummy-tekton-ace/main/scripts/0_dummy-appconnect-tekton-dashboard.yaml
customresourcedefinition.apiextensions.k8s.io/extensions.dashboard.tekton.dev created
serviceaccount/tekton-dashboard created
clusterrole.rbac.authorization.k8s.io/tekton-dashboard-backend created
clusterrole.rbac.authorization.k8s.io/tekton-dashboard-dashboard created
clusterrole.rbac.authorization.k8s.io/tekton-dashboard-pipelines created
clusterrole.rbac.authorization.k8s.io/tekton-dashboard-tenant created
clusterrole.rbac.authorization.k8s.io/tekton-dashboard-triggers created
clusterrolebinding.rbac.authorization.k8s.io/tekton-dashboard-backend created
service/tekton-dashboard created
W0828 15:20:03.805791 2108 warnings.go:70] would violate PodSecurity "restricted:latest": allowPrivilegeEscalation != false
d capabilities (container "tekton-dashboard" must set securityContext.capabilities.drop=["ALL"]), seccompProfile (pod or conta
st")
deployment.apps/tekton-dashboard created
rolebinding.rbac.authorization.k8s.io/tekton-dashboard-pipelines created
rolebinding.rbac.authorization.k8s.io/tekton-dashboard-dashboard created
rolebinding.rbac.authorization.k8s.io/tekton-dashboard-triggers created
clusterrolebinding.rbac.authorization.k8s.io/tekton-dashboard-tenant created
route.route.openshift.io/tekton-dashboard created

```

Luego, se debe validar todos los **RECURSOS** creados en el **NAMESPACE**, resaltando los asociados al: **DASHBOARD** como se muestra en **IMAGEN**:

`$ oc get pods,deployments,services,routes -n openshift-pipelines`

```

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc get pods,deployments,services,routes -n openshift-pipelines
NAME                                     READY   STATUS    RESTARTS   AGE
pod/pipelines-as-code-controller-6d5df69996-jbtbt  1/1     Running   0           16d
pod/pipelines-as-code-watcher-7fb66d9667-8kp4g    1/1     Running   0           16d
pod/pipelines-as-code-webhook-7b8cdf6446-s9d8c    1/1     Running   0           16d
pod/tekton-chains-controller-745585dd48-wpc5d     1/1     Running   0           16d
pod/tekton-dashboard-ff6f46b77-qqj6s             1/1     Running   0           3m4s
pod/tekton-operator-proxy-webhook-7db8979468-twbvk 1/1     Running   0           16d
pod/tekton-pipelines-controller-79fd798df9-5n66t   1/1     Running   0           16d
pod/tekton-pipelines-remote-resolvers-6458b48c46-crbst 1/1     Running   0           16d
pod/tekton-pipelines-webhook-5bfcf4ffd-c67br      1/1     Running   0           16d
pod/tekton-triggers-controller-549db989cb-jnrfn    1/1     Running   0           16d
pod/tekton-triggers-core-interceptors-84467b88d4-kr9hb 1/1     Running   0           16d
pod/tekton-triggers-webhook-6656d65cb4-vg2gq      1/1     Running   0           16d
pod/tkn-cli-serve-68c67fd6d6-m6jsk               1/1     Running   0           16d

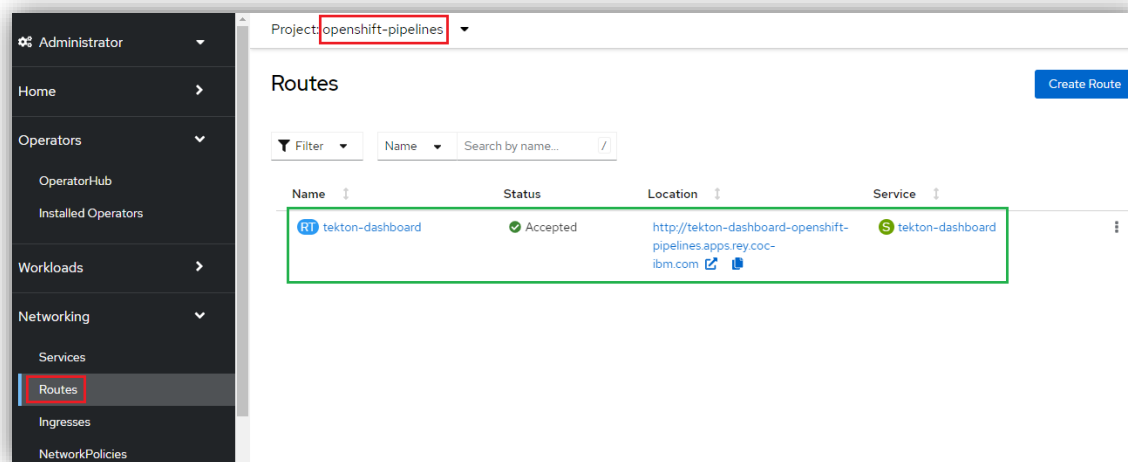
NAME                                     READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/pipelines-as-code-controller 1/1     1             1           16d
deployment.apps/pipelines-as-code-watcher    1/1     1             1           16d
deployment.apps/pipelines-as-code-webhook    1/1     1             1           16d
deployment.apps/tekton-chains-controller     1/1     1             1           16d
deployment.apps/tekton-dashboard             1/1     1             1           3m4s
deployment.apps/tekton-operator-proxy-webhook 1/1     1             1           16d
deployment.apps/tekton-pipelines-controller  1/1     1             1           16d
deployment.apps/tekton-pipelines-remote-resolvers 1/1     1             1           16d
deployment.apps/tekton-pipelines-webhook     1/1     1             1           16d
deployment.apps/tekton-triggers-controller   1/1     1             1           16d
deployment.apps/tekton-triggers-core-interceptors 1/1     1             1           16d
deployment.apps/tekton-triggers-webhook      1/1     1             1           16d
deployment.apps/tkn-cli-serve                1/1     1             1           16d

NAME                                     TYPE          CLUSTER-IP      EXTERNAL-IP      PORT(S)
service/pipelines-as-code-controller     ClusterIP     172.30.133.141  <none>           8080/TCP,9090/TCP
service/pipelines-as-code-watcher        ClusterIP     172.30.158.40   <none>           9090/TCP
service/pipelines-as-code-webhook         ClusterIP     172.30.36.176   <none>           443/TCP
service/tekton-dashboard                  ClusterIP     172.30.229.100  <none>           9097/TCP
service/tekton-operator-proxy-webhook     ClusterIP     172.30.17.203   <none>           443/TCP
service/tekton-pipelines-controller       ClusterIP     172.30.172.227  <none>           9090/TCP,8008/TCP,8080/TCP
service/tekton-pipelines-webhook          ClusterIP     172.30.191.31   <none>           9090/TCP,8008/TCP,443/TCP,8080/TCP
service/tekton-triggers-controller        ClusterIP     172.30.95.227   <none>           9000/TCP
service/tekton-triggers-core-interceptors ClusterIP     172.30.40.251   <none>           8443/TCP
service/tekton-triggers-webhook           ClusterIP     172.30.211.242  <none>           443/TCP

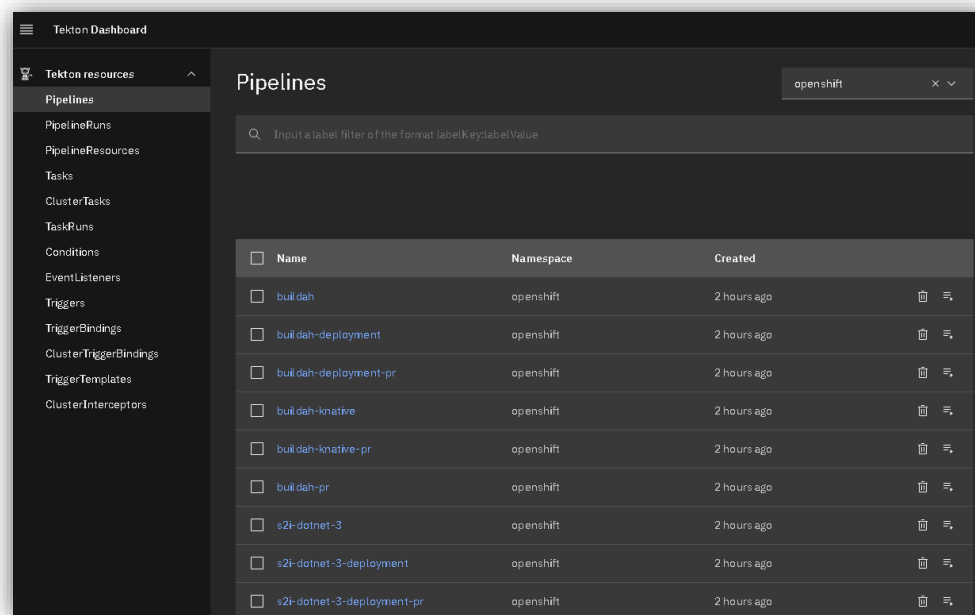
```

Luego, se ingresa gráficamente en **OPENSIFT / ROUTES** filtrando por **NAMESPACE: openshift-pipelines**, & seleccionar: **tekton-dashboard**, ingresando al **LINK** similar generado:

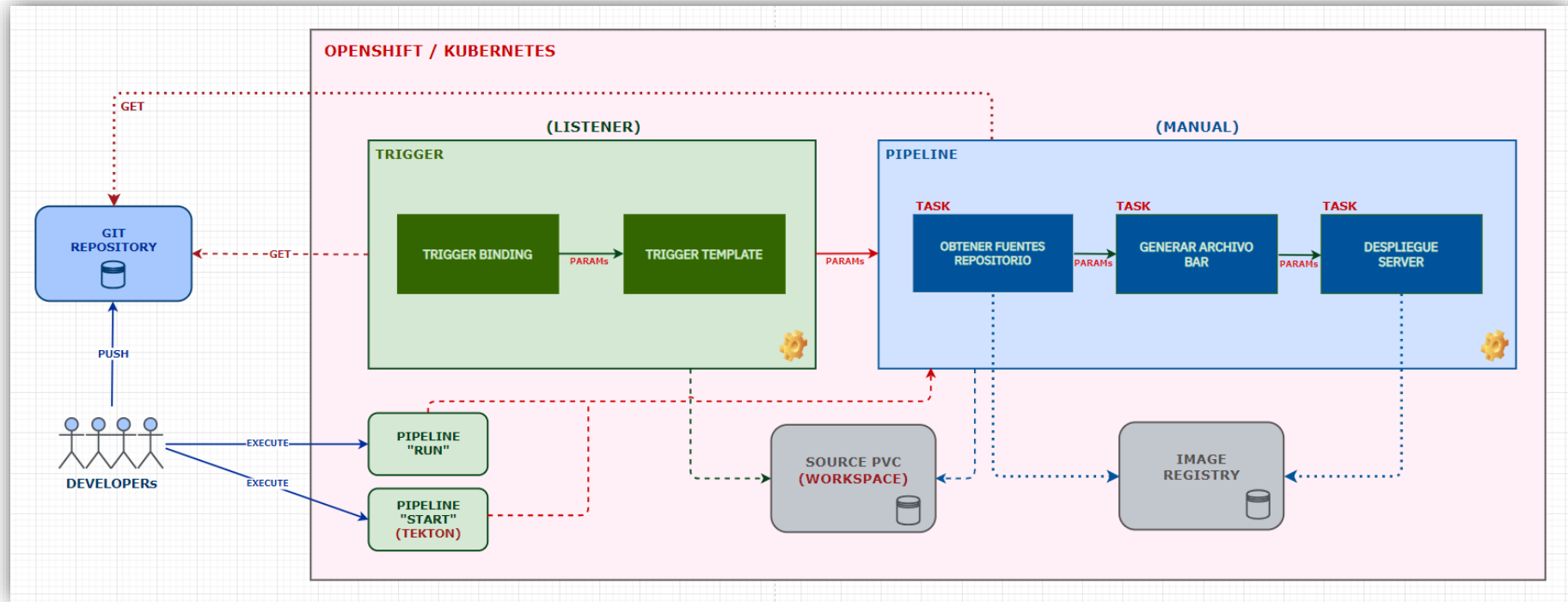
<http://tekton-dashboard-openshift-pipelines.apps.rey.coc-ibm.com>



Finalmente, se aprecia como debería visualizarse el **TEKTON DASHBOARD** desplegado.



4. DEMO: **INTEGRACIÓN**



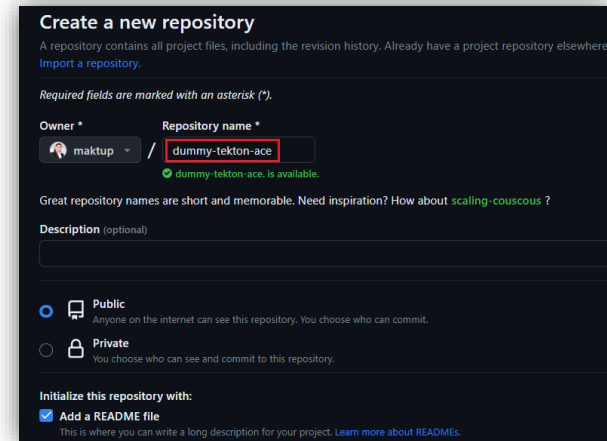
TIPOS DE DESPLIEGUE:

1. Recurso **PIPELINE**: Utilizado para la ejecución **MANUAL** (**PIPELINE RUN**, **PIPELINE START**).
2. Recurso **TRIGGER**: Complemento utilizado para la ejecución **AUTOMATIZADA**.

Por tema de orden, se recomienda manejar un **REPOSITORIO: GIT** independiente por **MicroServicio**, en el cual se ubiquen los **Script** utilizados.

En este caso se maneja:

REPOSITORIO: <https://github.com/maktup/dummy-tekton-ace.git>



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner * maktup / Repository name * dummy-tekton-ace

dummy-tekton-ace is available.

Great repository names are short and memorable. Need inspiration? How about [scaling-couscous](#)?

Description (optional)

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

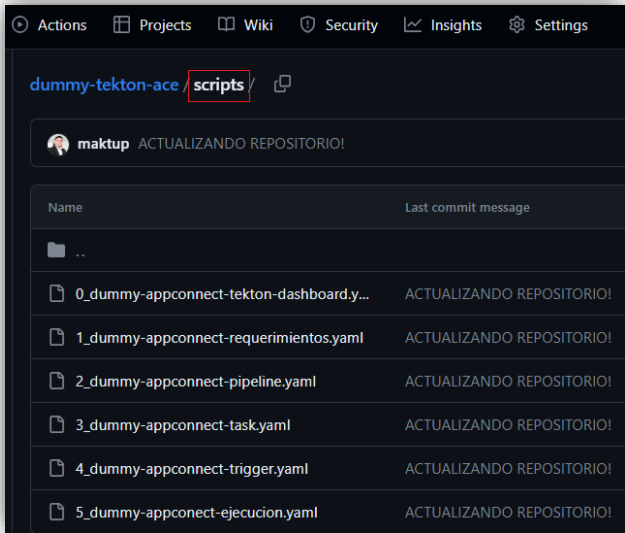
Initialize this repository with:

☒ Add a README file
This is where you can write a long description for your project. [Learn more about READMEs.](#)




Los **DIRECTORIOS** internos en **GITHUB** que se manejan en este punto, contienen:


- **scripts**: contiene los **scripts YAML** para ser instalados **ORDENADAMENTE**, para la construcción del flujo **DEVOPs** con **TEKTON**.



NOMBRE	DETALLE
0_dummy-appconnect-tekton-dashboard.yaml	Script YAML utilizado para la instalación del DASHBOARD de TEKTON .
1_dummy-appconnect-requerimientos.yaml	Script YAML utilizado para la instalación de TODOS los REQUERIMIENTOS previos al manejo de OPENSIFT PIPELINE (<i>acceso, seguridad, etc</i>).
2_dummy-appconnect-pipeline.yaml	Script YAML utilizado para la instalación del RECURSO con la estructura de PIPELINE .
3_dummy-appconnect-task.yaml	Script YAML utilizado para la instalación de los RECURSOS de tipo TASK con la LÓGICA de flujo (+ IMPORTANTE).
4_dummy-appconnect-trigger.yaml	Script YAML utilizado para la instalación de los RECURSOS para la AUTOMATIZACIÓN del despliegue.
5_dummy-appconnect-ejecucion.yaml	Script YAML utilizado para el TEST por medio del PIPELINE RUN .

-





Container software

library

Get entitlement key

View library

Access your container software

Below you'll find your entitlement key—Use it to verify and gain access to your container software. Check out the [container software library](#) to see details and a full list of the container software you own.

Entitlement key

Issued Date: September 15, 2021

```

p39n3m3AGN3y3y3G3C3m3o3u3G3B3U3E3M3P3y3p3C3M3Q3O3D3U3T3P3y3C3u3S3U3E3U3C3P3Y3Q3C3Q3P3M3U3T3P3d3o3u3E3n3g3S3S3G3S3H3Y3M3O3D3U3T3P3Y3E3S3m3M3N3U3Y3M3D3C3m3E3u3X3U3C3P3Q3n3D3d3W3r3C3G3F3M3P3P3d3C3u3C3u3

```

[Copy key](#)
[Get new key](#)

```
$ oc create secret docker-registry ibm-entitlement-key --
docker-username=cp --docker-
password=eYJ0eXaI0iJKV1QiLChJbGci0iJlUz1NiJ9.eYJ
pc3Mi0iJlQk0qTWFya2V0cGxhY2UiLChJpYXQiOiJlUz1NiJ9
mMj12NDYslmp0aSI6IjhiMjg5MmQ0NTIjZDQwYmRhMzJk
NmJiNjc1MTRmMjVkl0dipBDFU4Hv2mTNNe4Ma2QNo
7AdBW-Oh75JuWa-6VjIm --docker-server=cp.icr.io --
namespace=dummy-tekton-appconnect
```

```

$ oc create secret docker-registry ibm-entitlement-key --docker-username=cp --docker-password=eyJ0eXAiOiJKV1QiCjBhcGciOiJIUzI1Ni9yeyJpc3MiOiJJK0k0dWVya2V0cGxhY2UuIHR5cGU6ImJ2NDY5Imp0aSI6ImJmMjM0QmNTI1ZDQ0YmRhZmZkNmJiJnJjMTRmMjVkbWV0Ln0.dlpBDFU4Hv2mTNN4Ma2QNo7ADBw-Oh75J0Ja-6Vj1M --docker-server=cp.icr.io --namespace=dummy-tekton-appconnect
secret/ibm-entitlement-key created

```

```
$ oc create ns dummy-tekon-appconnect
```



```
GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc create ns dummy-tekton-appconnect
namespace/dummy-tekton-appconnect created
```

Luego, es necesario **DESCARGAR** & **TAGEAR** localmente hacia un **IMAGE REPOSITORY**, una **IMAGE** de **ACE12**, para ello se debe ejecutar:

```
$ docker logout
$ docker login docker.io
Username: XXX
Password: XXX
Login Succeeded
```

Se debe crear el **IMAGE REPOSITORY** con el nombre: **maktup/ibm-appconnect-12**

Identificar la **VERSIÓN** de la **IMAGE** de:

https://www.ibm.com/docs/en/app-connect/containers_cd?topic=obtaining-app-connect-enterprise-server-image-from-cloud-container-registry

```
$ docker pull icr.io/appc-dev/ace-server@sha256:a41f7501fe4025d2705bcabf1ad2ff523bc
af9ec263b98054501de1fc0cf5f62
```

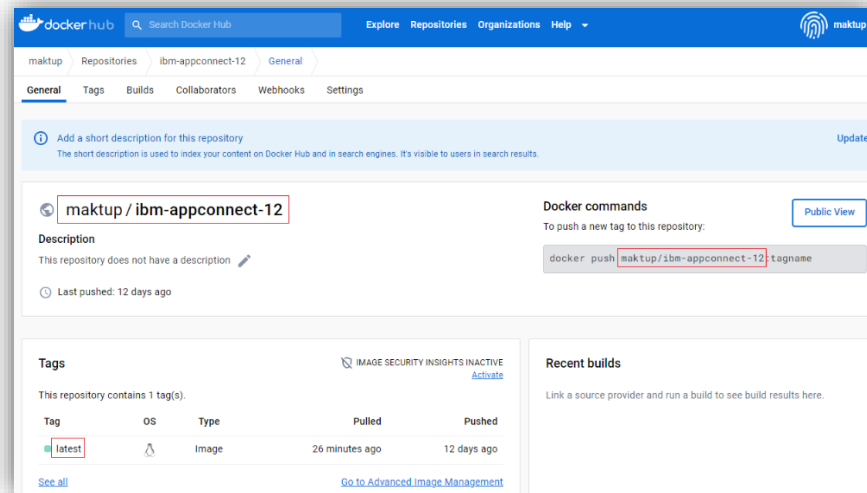
```
$ docker images
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
icr.io/appc-dev/ace-server <none>       164ac6f477db     15 months ago   1.86GB
```

```
$ docker tag 164ac6f477db docker.io/maktup/ibm-appconnect-12:latest
```

```
$ docker images
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
maktup/ibm-appconnect-12 latest        164ac6f477db     15 months ago   1.86GB
icr.io/appc-dev/ace-server <none>       164ac6f477db     15 months ago   1.86GB
```

```
$ docker push maktup/ibm-appconnect-12:latest
```

```
GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ docker images
REPOSITORY          TAG          IMAGE ID          CREATED          SIZE
docker/getting-started latest        3e4394f6b72f     8 months ago    47MB
maktup/ibm-appconnect-12 latest        164ac6f477db     15 months ago   1.86GB
icr.io/appc-dev/ace-server <none>       164ac6f477db     15 months ago   1.86GB
```



Luego, para la instalación de cada **RECURSO**, referenciando a los **scripts** existentes en **GITHUB**.

//Creación REQUERIMIENTOS:

```
$ oc create -f https://raw.githubusercontent.com/makutp/dummy-tekton-ace/main/scripts/1_dummy-appconnect-requerimientos.yaml
```

//Creación PERMISOS:

```
$ oc adm policy add-scc-to-user privileged -z pipeline -n dummy-tekton-appconnect
```

```
$ oc adm policy add-role-to-user edit -z pipeline -n dummy-tekton-appconnect
```

//Creación de PIPELINE:

```
$ oc create -f https://raw.githubusercontent.com/makutp/dummy-tekton-ace/main/scripts/2_dummy-appconnect-pipeline.yaml
```

//Creación de TASKs:

```
$ oc create -f https://raw.githubusercontent.com/makutp/dummy-tekton-ace/main/scripts/3_dummy-appconnect-task.yaml
```

//Creación de TRIGGERS:

```
$ oc create -f https://raw.githubusercontent.com/makutp/dummy-tekton-ace/main/scripts/4_dummy-appconnect-trigger.yaml
```

```
GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc create -f https://raw.githubusercontent.com/makutp/dummy-tekton-ace/main/scripts/1_dummy-appconnect-requerimientos.yaml
persistentvolumeclaim/source-pvc-pipe created
imagestream.image.openshift.io/ace-server created
clusterrole.rbac.authorization.k8s.io/pipeline-deployer-aceflows-role created
clusterrolebinding.rbac.authorization.k8s.io/pipeline-deployer-aceflows-rolebinding created
rolebinding.rbac.authorization.k8s.io/edit created
secret/github-interceptor-secret created

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc adm policy add-scc-to-user privileged -z pipeline -n dummy-tekton-appconnect
clusterrole.rbac.authorization.k8s.io/system:openshift:scc:privileged added: "pipeline"

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc adm policy add-role-to-user edit -z pipeline -n dummy-tekton-appconnect
clusterrole.rbac.authorization.k8s.io/edit added: "pipeline"

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc create -f https://raw.githubusercontent.com/makutp/dummy-tekton-ace/main/scripts/2_dummy-appconnect-pipeline.yaml
pipeline.tekton.dev/pipeline-build-and-deploy created

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc create -f https://raw.githubusercontent.com/makutp/dummy-tekton-ace/main/scripts/3_dummy-appconnect-task.yaml
task.tekton.dev/ace-obtain-repository-sources created
task.tekton.dev/ace-generate-bar-file created
task.tekton.dev/ace-create-integration-or-runtime-server created

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc create -f https://raw.githubusercontent.com/makutp/dummy-tekton-ace/main/scripts/4_dummy-appconnect-trigger.yaml
trigger.triggers.tekton.dev/trig-dummy-micro-01 created
triggerbinding.triggers.tekton.dev/trig-bind-dummy-micro-01 created
triggertemplate.triggers.tekton.dev/trig-temp-dummy-micro-01 created
eventlistener.triggers.tekton.dev/listener-dummy-micro-01 created
route.route.openshift.io/el-listener-dummy-micro-01 created
```

Luego, de ejecutados los **SCRIPTs**, se procede a validar la existencia de algunos **RECURSOS** importantes de **TEKTON**:

//Listar PIPELINES existentes en el NAMESPACE:

```
$ tkn pipeline ls -n dummy-tekton-appconnect
```

//Listar TAREAS (CREADAS MANUALMENTE) existentes en el NAMESPACE:

```
$ tkn task ls -n dummy-tekton-appconnect
```

```
GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ tkn pipeline ls -n dummy-tekton-appconnect
NAME                                AGE      LAST RUN    STARTED    DURATION    STATUS
pipeline-build-and-deploy          3 minutes ago    ---        ---        ---        ---

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ tkn task ls -n dummy-tekton-appconnect
NAME                                DESCRIPTION                                AGE
ace-create-integration-or-runtime-server  3 minutes ago
ace-generate-bar-file                    3 minutes ago
ace-obtain-repository-sources            ESTAS TAREAS SON 'T... 3 minutes ago
```

A. EJECUCIÓN "MANUAL":

El **TEST** de ejecución **MANUAL** del flujo **PIPELINE**, se puede realizar de **2** formas:

A. PIPELINE RUN: en base a un Script **YAML**

```
$ oc create -f
https://raw.githubusercontent.com/makut/dummy-tekton-ace/main/scripts/5_dummy-appconnect-ejecucion.yaml
```



```
GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc create -f https://raw.githubusercontent.com/makut/dummy-tekton-ace/main/scripts/5_dummy-appconnect-ejecucion.yaml
pipelinerun.tekton.dev/dummy-pipeline-run created
```

Project: dummy-tekton-appconnect

Pipelines

Pipelines PipelineRuns PipelineResources Repositories

Filter Name Search by name... / /

Name	Status	Task status	Started
PLR dummy-pipeline-run	Succeeded		28 ago. 2023, 18:58

Project: dummy-tekton-appconnect

PipelineRuns > PipelineRun details

PLR dummy-pipeline-run Succeeded

Details YAML TaskRuns Parameters Logs Events

Task	Log
pipe-obtener-fuentes-repositorio	
pipe-generar-archivo-bar	
pipe-despliegue-server	<pre>pipe-despliegue-server integrationruntime.appconnect.ibm.com "csm-microservicio-ir" force deleted - EL RECURSO ==> [IntegrationRuntime: csm-microservicio-ir] HA SIDO 'ELIMINADO' EXITOSAMENTE. 18.1. ==> ESPERANDO PARA LA ELIMINACION DEL 'INTEGRATION-RUNTIME' (15s) 19. ==> CREANDO 'INTEGRATION-RUNTIME' [YAML]: - NOMBRE: [csm-microservicio-ir] - NAMESPACE: [dummy-tekton-appconnect] - LICENCIA: [I-APPH-CUOCNR] - MODALIDAD: [CloudPakForIntegrationNonProductionFREE] - IMAGEN 'MICROSERVICIO': [image-registry.openshift-image-registry.svc:5000/dummy-tekton-appconnect/csm-microservicio:6ee64ffd0e4b0a01a3ee9673b202506972d4ae2e] - REPLICAS: [1] - CPU 'REQUEST': [400m] - MEMORIA 'LIMIT': [400Mi] - MEMORIA 'REQUEST': [300Mi] - VERSION: [12.0.7.0-r3] integrationruntime.appconnect.ibm.com/csm-microservicio-ir created 20. ==> LISTANDO 'INTEGRATION-RUNTIME' EXISTENTES: - EJECUTANDO ==> [oc get IntegrationRuntime -n dummy-tekton-appconnect -o wide] NAME RESOLVEDVERSION STATUS REPLICAS AVAILABLEREPLICAS URL AGE CUSTOMIMAGES csm-microservicio-ir 12.0.7.0-r3 Pending 1 0 ls true</pre>

B. PIPELINE START: en base a COMANDOS.

```
$ tkn pipeline start pipeline-build-and-deploy -w
name=workspace-pipeline,claimName=source-pvc-pipe -p
nombre-integration-server=csm-microservicio -p nombre-
namespace-appconnect=dummy-tekton-appconnect -p url-
repositorio-git=https://github.com/maktup/dummy-tekton-
ace.git -p nombre-subdirectorio-
git=dummy_csm_microservice -p nombre-repositorio-
git=dummy-tekton-ace -p branch-repositorio-git=main -p
nombre-proyecto-appconnect=dummy_csm_microservice -p
version-image-appconnect="12.0.7.0-r3" -p licencia-image-
appconnect="L-APEH-CJUCNR" -p ruta-image-
appconnect="docker.io/maktup/ibm-appconnect-12:latest" -p
tipo-licencia-
appconnect="CloudPakForIntegrationNonProductionFREE" -
p tipo-despliegue-servidor="3" -n dummy-tekton-appconnect
```

IMPORTANTE: “Los campos **RESALTADOS** hacen referencia a los **RECURSOS** & **PARÁMETROS** para enviar, propiamente al **RECURSO PIPELINE** existente en el **NAMESPACE**”.

“Solo deberán ser enviarán los **PARÁMETROS** que se requiere que sean **DINÁMICOS**, los que **NO** sean enviados serán seteados en **HardCode** desde: **pipeline-build-and-deploy**”.

“Las opciones (1,2,3) ingresadas en el **PARÁMETRO: tipo-despliegue-servidor** definen:”

1. Despliega/Actualiza sobre **INTEGRATION SERVER**.
2. Despliega/Actualiza sobre **INTEGRATION RUNTIME**.
3. Despliega/Actualiza sobre **AMBOS**.

Además, los **LOGs** generados sería **SIMILARES** a este:

```
$ tkn pipelinerun logs pipeline-build-and-deploy-run-k5mrx -f -n
dummy-tekton-appconnect
```


dummy-pipeline-run.log

```
GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ tkn pipeline start pipeline-build-and-deploy -w name=workspace-pipeline,claimName=source-pvc-pipe -
p nombre-integration-server=csm-microservicio -p nombre-namespace-appconnect=dummy-tekton-appconnect
-p url-repositorio-git=https://github.com/maktup/dummy-tekton-ace.git -p nombre-subdirectorio-git=dum
my_csm_microservice -p nombre-repositorio-git=dummy-tekton-ace -p branch-repositorio-git=main -p nomb
re-proyecto-appconnect=dummy_csm_microservice -p version-image-appconnect="12.0.7.0-r3" -p licencia-i
mage-appconnect="L-APEH-CJUCNR" -p ruta-image-appconnect="docker.io/maktup/ibm-appconnect-12:latest"
-p tipo-licencia-appconnect="CloudPakForIntegrationNonProductionFREE" -p tipo-despliegue-servidor="3"
-n dummy-tekton-appconnect
PipelineRun started: pipeline-build-and-deploy-run-k5mrx

In order to track the PipelineRun progress run:
tkn pipelinerun logs pipeline-build-and-deploy-run-k5mrx -f -n dummy-tekton-appconnect
```

Project: dummy-tekton-appconnect		
Pipelines		
Pipelines	PipelineRuns	PipelineResources Repositories
Filter Name Search by name...		
Name	Status	Task status
PLD pipeline-build-and-deploy-run-k5mrx	Succeeded	

Project **dummy-tekton-appconnect**

PipelineRuns > PipelineRun details

PLR pipeline-build-and-deploy-run-k5rnx Succeeded

Details YAML TaskRuns Parameters **Logs** Events

- pipe-obtener-fuentes-repositorio
- pipe-generar-archivo-bar
- pipe-despliegue-server**

```

pipe-despliegue-server

integrationruntime.appconnect.ibm.com "csm-microservicio-ir" force deleted
- El RECURSO ==> [[IntegrationRuntime: csm-microservicio-ir] HA SIDO 'ELIMINADO' EXITOSAMENTE.

18.1. ==> ESPERANDO PARA LA ELIMINACION DEL 'INTEGRATION-RUNTIME' (15s)

19. ==> CREANDO: 'INTEGRATION-RUNTIME' [YAML]:
- NOMBRE: [csm-microservicio-ir]
- NAMESPACE: [dummy-tekton-appconnect]
- LICENCIA: [I-APEN-CJUCNR]
- MODALIDAD: [CloudPakForIntegrationNonProductionFREE]
- IMAGEN 'MICROSERVICIO': [image-registry.openshift-image-registry.svc:5000/dummy-tekton-appconnect/csm-microservicio:6ee6ff3d0e4b0a01a3ee9673b202506972ddee2e]
- REPLICAS: [1]
- CPU 'LIMIT': [400m]
- CPU 'REQUEST': [300m]
- MEMORIA 'LIMIT': [400M]
- MEMORIA 'REQUEST': [300M]
- VERSION: [12.0.7.0-r3]

integrationruntime.appconnect.ibm.com/csm-microservicio-ir created

20. ==> LISTANDO 'INTEGRATION-RUNTIME' EXISTENTES:
- EJECUTANDO ==> [oc get IntegrationRuntime -n dummy-tekton-appconnect -o wide]
NAME              RESOLVEDVERSION STATUS  REPLICAS  AVAILABLEREPLICAS  URL  AGE  CUSTOMIMAGES
csm-microservicio-ir 12.0.7.0-r3 Pending 1          0              1s  true

```

Luego, se procede a validar el **REQUEST** enviado
¿QUÉ PARÁMETROS FUERON ENVIADO AL
PIPELINE? en: **RUNTIME**:

```
$ tkn pipelinerun list -n dummy-tekton-appconnect
```

```
$ tkn pipelinerun describe pipeline-build-and-deploy-run-4xt75 -n  
dummy-tekton-appconnect
```

IMPORTANTE: “Reemplazar el valor del **PIPELINE** en
la **2da** sentencia de código”.

```
GOX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ tkn pipelinerun list -n dummy-tekton-appconnect
NAME                               STARTED          DURATION         STATUS
pipeline-build-and-deploy-run-k5rnx 12 minutes ago   3m33s            Succeeded

GOX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ tkn pipelinerun describe pipeline-build-and-deploy-run-k5rnx -n dummy-tekton-appconnect
Name:                             pipeline-build-and-deploy-run-k5rnx
Namespace:                         dummy-tekton-appconnect
Pipeline Ref:                       pipeline-build-and-deploy
Service Account:                    pipeline
Labels:
  tekton.dev/pipeline=pipeline-build-and-deploy
Annotations:
  chains.tekton.dev/signed=true

Status

STARTED          DURATION         STATUS
13 minutes ago   3m33s            Succeeded

Timeouts
Pipeline:        1h0m0s

Params

NAME                               VALUE
branch-repositorio-git             main
licencia-image-appconnect          L-APEH-CJUCNR
nombre-integration-server          csm-microservicio
nombre-namespace-appconnect        dummy-tekton-appconnect
nombre-proyecto-appconnect          dummy_csm_microservice
nombre-repositorio-git             dummy-tekton-ace
nombre-subdirectorio-git           dummy_csm_microservice
ruta-image-appconnect              docker.io/maktup/ibm-appconnect-12:latest
tipo-despliegue-servidor           3
tipo-licencia-appconnect           CloudPakForIntegrationNonProductionFREE
url-repositorio-git                https://github.com/maktup/dummy-tekton-ace.git
version-image-appconnect            12.0.7.0-r3

Workspaces

NAME          SUB PATH          WORKSPACE BINDING
workspace-pipeline --- PersistentVolumeClaim (claimName=source-pvc-pipe)

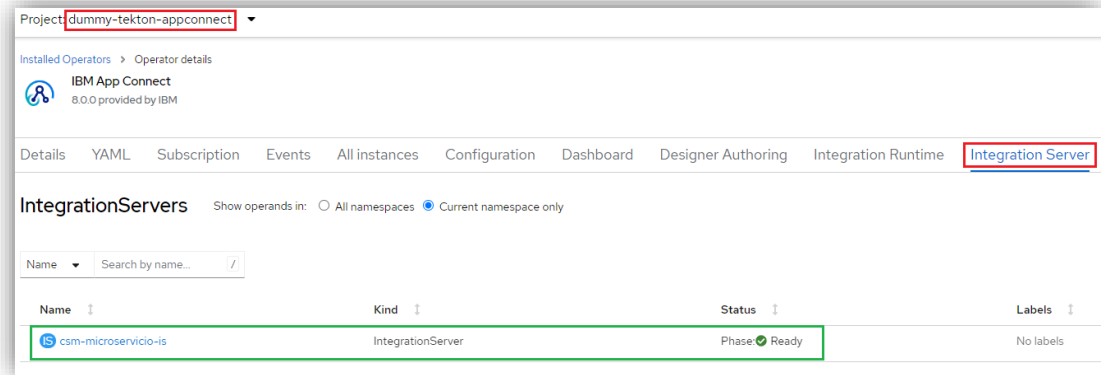
Taskruns

NAME                               TASK NAME                               STARTED          DURATION         STATUS
pipeline-build-and-deploy-run-k5rnx-pipe-despliegue-server  pipe-despliegue-server  12 minutes ago   2m48s            Succeeded
pipeline-build-and-deploy-run-k5rnx-pipe-generar-archivo-bar  pipe-generar-archivo-bar  13 minutes ago   30s              Succeeded
pipeline-build-and-deploy-run-k24a1a63221a8879c408b293b88758a84  pipe-obtener-fuentes-repositorio  13 minutes ago   15s              Succeeded
```

Luego, ingresando en: **OPERATOR / INSTALLED OPERATOR / IBM App Connect**, se aprecia que la **AUTOMATIZACIÓN** ha logrado **DESPLEGAR** el **.bar** en las modalidades:

- ✓ **INTEGRATION SERVER.**
- ✓ **INTEGRATION RUNTIME.**

IMPORTANTE: “Este **INTEGRATION RUNTIME** es una nueva característica de **IBM APP CONNECT**, que habilita crear **SERVERLESS** integration runtimes”.



Project: dummy-tekton-appconnect

Installed Operators > Operator details

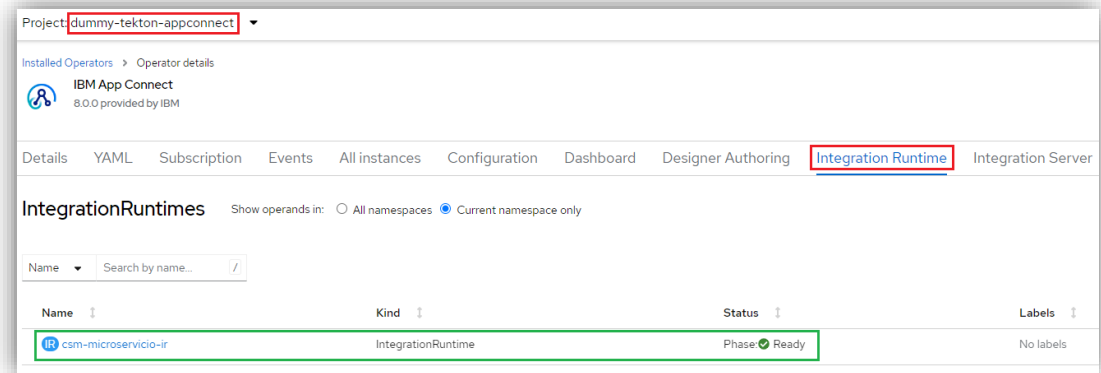
IBM App Connect
8.0.0 provided by IBM

Details | YAML | Subscription | Events | All instances | Configuration | Dashboard | Designer Authoring | Integration Runtime | **Integration Server**

IntegrationServers Show operands in: ☐ All namespaces ☒ Current namespace only

Name Search by name... /

Name	Kind	Status	Labels
csm-microservicio-is	IntegrationServer	Phase: Ready	No labels



Project: dummy-tekton-appconnect

Installed Operators > Operator details

IBM App Connect
8.0.0 provided by IBM

Details | YAML | Subscription | Events | All instances | Configuration | Dashboard | Designer Authoring | **Integration Runtime** | Integration Server

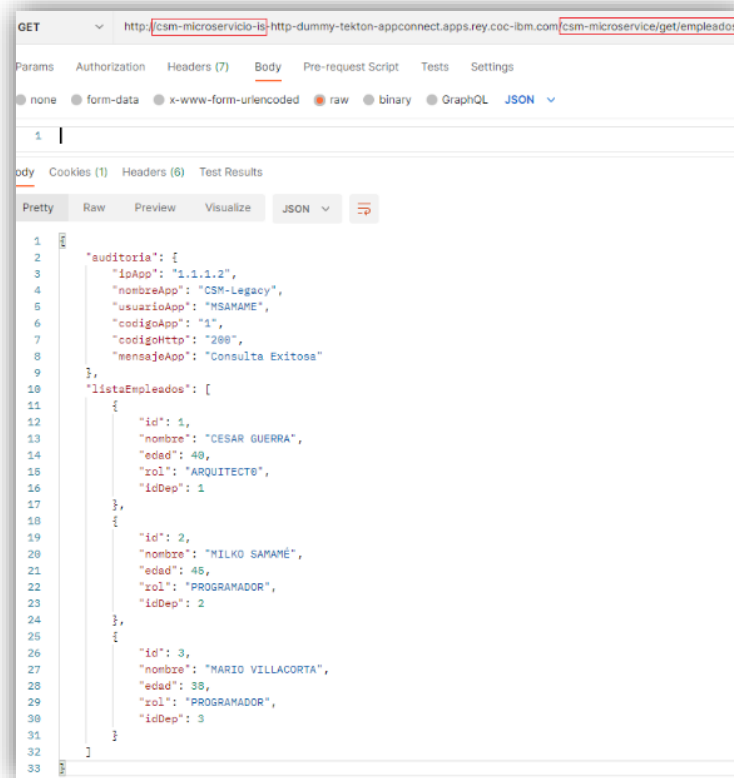
IntegrationRuntimes Show operands in: ☐ All namespaces ☒ Current namespace only

Name Search by name... /

Name	Kind	Status	Labels
csm-microservicio-ir	IntegrationRuntime	Phase: Ready	No labels

Finalmente, el resultado del **TEST** es el siguiente:

- ✓ **INTEGRATION SERVER.**
<http://csm-microservicio-is-http-dummy-tektion-appconnect.apps.rey.coc-ibm.com/csm-microservice/get/empleados>
- ✓ **INTEGRATIO RUNTIME.**
<http://csm-microservicio-ir-http-dummy-tektion-appconnect.apps.rey.coc-ibm.com/csm-microservice/get/empleados>



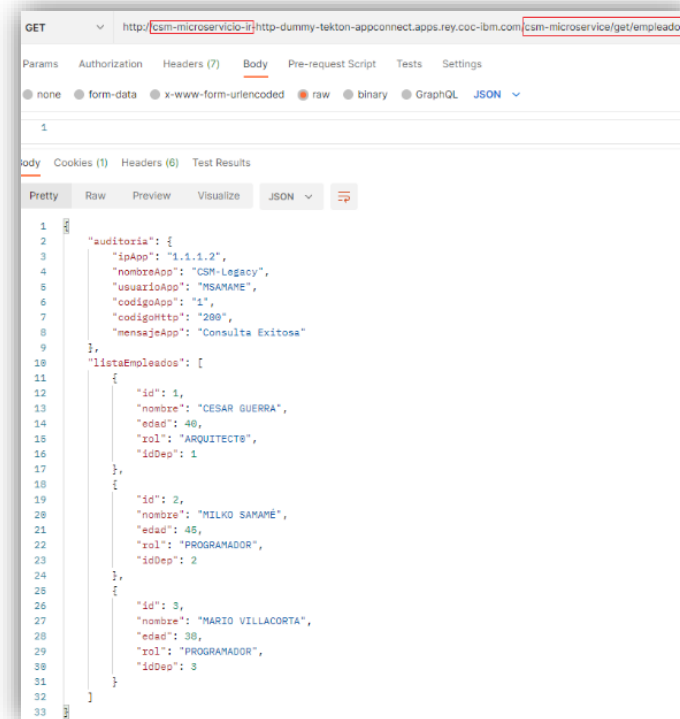
```

GET http://csm-microservicio-is-http-dummy-tektion-appconnect.apps.rey.coc-ibm.com/csm-microservice/get/empleados

Body
none form-data x-www-form-urlencoded raw binary GraphQL JSON

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33

{
  "auditoria": [
    {
      "ipApp": "1.1.1.2",
      "nombreApp": "CSM-Legacy",
      "usuarioApp": "MSANAME",
      "codigoApp": "1",
      "codigoHttp": "200",
      "mensajeApp": "Consulta Exitosa"
    }
  ],
  "listaEmpleados": [
    {
      "id": 1,
      "nombre": "CESAR GUERRA",
      "edad": 49,
      "rol": "ARQUITECTE",
      "idDep": 1
    },
    {
      "id": 2,
      "nombre": "MILKO SAMAMÉ",
      "edad": 45,
      "rol": "PROGRAMADOR",
      "idDep": 2
    },
    {
      "id": 3,
      "nombre": "MARIO VILLACORTA",
      "edad": 38,
      "rol": "PROGRAMADOR",
      "idDep": 3
    }
  ]
}
  
```

```

GET http://csm-microservicio-1f-http-dummy-tekton-appconnect.apps.rey.coc.ibm.com/csm-microservicio/get/empleados

Body
none form-data x-www-form-urlencoded raw binary GraphQL JSON
1
body Cookies (1) Headers (6) Test Results
Pretty Raw Preview Visualize JSON
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
{
  "auditoria": {
    "ipApp": "1.1.1.2",
    "nombreApp": "CSM-Legacy",
    "usuarioApp": "MSAAME",
    "codigoApp": "i",
    "codigoHttp": "200",
    "mensajeApp": "Consulta Exitosa"
  },
  "listaEmpleados": [
    {
      "id": 1,
      "nombre": "CESAR GUERRA",
      "edad": 40,
      "rol": "ARQUITECTO",
      "idDep": 1
    },
    {
      "id": 2,
      "nombre": "MILKO SAMAME",
      "edad": 45,
      "rol": "PROGRAMADOR",
      "idDep": 2
    },
    {
      "id": 3,
      "nombre": "MARIO VILLACORTA",
      "edad": 38,
      "rol": "PROGRAMADOR",
      "idDep": 3
    }
  ]
}

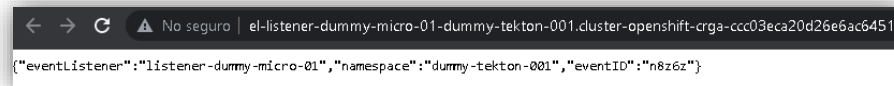
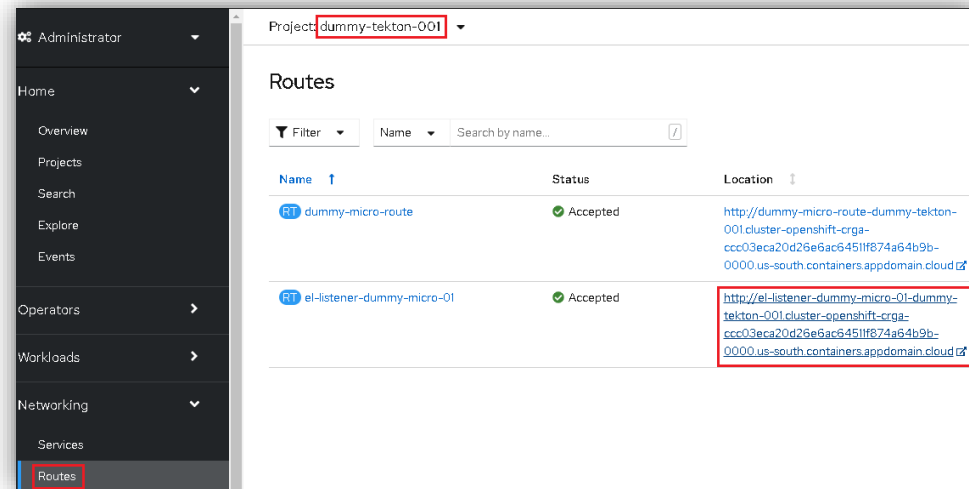
```

B. EJECUCIÓN “AUTOMÁTICA”:

Para el **TEST** de ejecución **AUTOMÁTICA** del flujo **PIPELINE**, es necesario el uso de un recurso: **ROUTE** para asociado al: **EVENT-LISTENER**.

Este servirá en la comunicación con el **REPOSITORIO (GITHUB)**, & se obtienen ingresando en **OPENSIFT: NETWORKING / ROUTES** se selecciona el **NAMESPACE** manejado, se selecciona el **ROUTE** con el nombre: **(el-listener-*)**, se validar que esté activo & se muestre el **JSON**.

IMPORTANTE: "se debe de guardar la **URL** del **EVENT-LISTENER** generada: <http://el-listener-dummy-micro-01-dummy-tekton-appconnect.apps.rev.coc-ibm.com>, ya que será requerido más adelante como **PARÁMETRO: Payload URL**".



Luego, se procede a obtener el valor del **SECRET** registrado para **REGISTRADO** en el repositorio **GIT**, para **CONFIGURAR** la **CONEXIÓN**:

```
$ oc get secrets github-interceptor-secret -n dummy-tekton-appconnect -o yaml
```

```
$ echo -n MTIzNDU2Nzg= | base64 --decode
```

IMPORTANTE: "El **SECRET** debe ser actualizado en la **2da** sentencia de comando, como se muestra en **IMAGEN**".

```

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ oc get secrets github-interceptor-secret -n dummy-tekton-appconnect -o yaml
apiVersion: v1
data:
  secretToken: MTIzNDU2Nzg=
kind: Secret
metadata:
  creationTimestamp: "2023-08-28T23:25:16Z"
  name: github-interceptor-secret
  namespace: dummy-tekton-appconnect
  resourceVersion: "101542727"
  uid: 9df755a2-444b-4bb7-b6ef-d1d703718cf1
type: Opaque

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop
$ echo -n MTIzNDU2Nzg= | base64 --decode
12345678

```

Luego, la **URL** anterior del **EVENT-LISTENER** se ingresa en la ruta de **GITHUB: SETTINGS/WEBHOOKS** para crear una **CONEXION**. Además, el valor del **SECRET** obtenido se ingresa, como se muestra en **IMAGEN**.

Esto hará que toda modificación que **IMPACTE** contra el **REPOSITORIO** de **GIT**, desencadenará & gatillará la activación del flujo: **PIPELINE**.

IMPORTANTE: “Esta conexión se debe realizar por **CLÚSTER** que se desee conectar”.

Wiki Security Insights **Settings**

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Webhooks

Environments

Codespaces

Pages

Security

Code security and analysis

Deploy keys

Secrets and variables

Integrations

GitHub Apps

Webhooks / Add webhook

We'll send a post request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in [our developer documentation](#).

Payload URL *

`http://el-listener-dummy-micro-01-dummy-tekton-appconnect.apf`

Content type

application/json

Secret

12345678

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.

Add webhook

Webhooks **Add webhook**

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

☒ `http://el-listener-dummy-micro-01-...` (all events) **Edit Delete**

EJECUCIÓN AUTOMÁTICA:

Para gatillar el flujo **PIPELINE**, se necesitará una **MODIFICACIÓN** en cualquier **FUENTE** de **ACE12** dentro del **REPOSITORIO** de **GITHUB**,

En la **IMAGEN** se aprecia como se procede a **MODIFICAR** un campo en el **ESQL** de **ACE12**. Luego, el directorio completo del **PROYECTO** es reemplazado localmente.

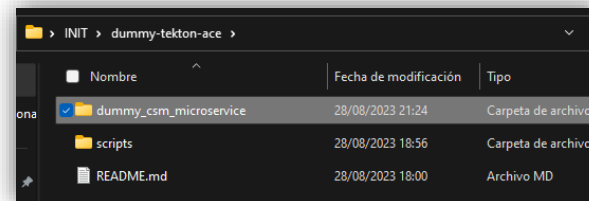
Después los **COMANDOS GIT** para la ejecutar son:

```
$ git init
$ git clone https://github.com/maktup/dummy-tektion-ace.git
$ cd dummy_csm_microservice
$ git add .
$ git status
$ git commit -m "ACTUALIZANDO REPOSITORIO!"
$ git push -f origin main
```

IMPORTANTE: "Este funcionará para el escenario de este **MICROSERVICIO**, para otros de debe **ADAPTAR** (URL, Directorio, etc)".

The screenshot shows the 'dummy_csm_microservice' project in the 'Application Development' environment. The project structure includes 'REST API Description', 'Resources', 'BARs', and 'Independent Resources'. The 'consultarEmpleadosAll.esql' file is open, displaying a series of SQL-like declarations and assignments for employee data. The 'SET vUsuarioApp' line is highlighted with a red box.

```
19 DECLARE vId2 INTEGER;
20 DECLARE vNombre2 CHARACTER;
21 DECLARE vEdad2 INTEGER;
22 DECLARE vRo12 CHARACTER;
23 DECLARE vIdDep2 INTEGER;
24
25 DECLARE vId3 INTEGER;
26 DECLARE vNombre3 CHARACTER;
27 DECLARE vEdad3 INTEGER;
28 DECLARE vRo13 CHARACTER;
29 DECLARE vIdDep3 INTEGER;
30
31 -- REGISTRAR DATOS 'RESPONSE':
32 SET vIpApp = '1.1.1.2';
33 SET vNombreApp = 'CSM-Legacy';
34 SET vUsuarioApp = 'SAMAME 2023';
35 SET vCodigoApp = '1';
36 SET vCodigoHttp = '200';
37 SET vMensajeApp = 'Consulta Exitosa';
38
39 SET vId1 = 001;
40 SET vNombre1 = 'CESAR GUERRA';
41 SET vEdad1 = 40;
42 SET vRo11 = 'ARQUITECT0';
43 SET vIdDep1 = 1;
44
45 SET vId2 = 002;
46 SET vNombre2 = 'MILKO SAMAME';
47 SET vEdad2 = 45;
48 SET vRo12 = 'PROGRAMADOR';
49 SET vIdDep2 = 2;
50
51 SET vId3 = 003;
52 SET vNombre3 = 'MARIO VILLACORTA';
53 SET vEdad3 = 38;
54 SET vRo13 = 'PROGRAMADOR';
55 SET vIdDep3 = 3;
56
57 CREATE LASTCHILD OF OutputRoot DOMAIN( 'JSON' );
58
```



```

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop/INIT/dummy-tekton-ace (main)
$ git add .
warning: in the working copy of 'dummy_csm_microservice/consultarEmpleadosAll.ConsultarEmpleado.esql', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'dummy_csm_microservice/gen/dummy_csm_microservice.msgflow', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'dummy_csm_microservice/restapi.descriptor', LF will be replaced by CRLF the next time Git touches it

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop/INIT/dummy-tekton-ace (main)
$ oc status
In project default on server https://api.rey.coc-ibm.com:6443

svc/openshift - kubernetes.default.svc.cluster.local
svc/kubernetes - 172.30.0.1:443 -> 6443

View details with 'oc describe <resource><name>' or list resources with 'oc get all'.

GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop/INIT/dummy-tekton-ace (main)
$ git commit -m "ACTUALIZANDO REPOSITORIO!"
[main e446b00] ACTUALIZANDO REPOSITORIO!
1 file changed, 1 insertion(+), 1 deletion(-)

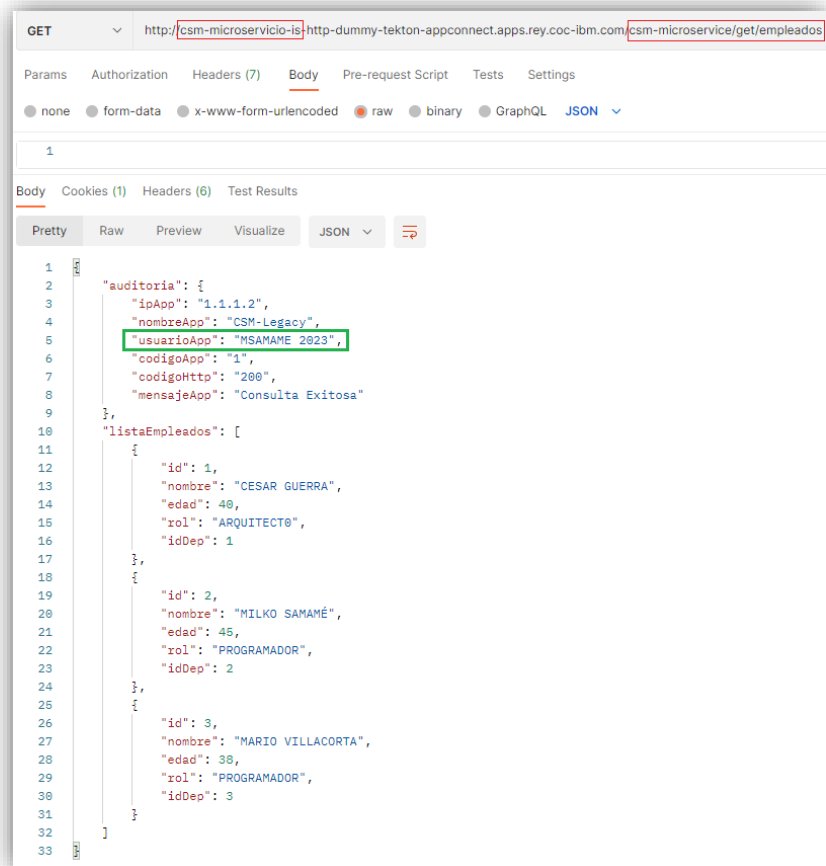
GMX+000996815@IBM-PF2PC7D4 MINGW64 ~/Desktop/INIT/dummy-tekton-ace (main)
$ git push -f origin main
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 454 bytes | 151.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/makutp/dummy-tekton-ace.git
   6ee64ff..e446b00  main -> main
  
```

Luego, como se muestra en las **IMÁGENES** el **DESPLIEGUE AUTOMÁTICO** del **MICROSERVICIO** se ha iniciado & después de un rato este será **COMPLETADO** a nivel del: **PIPELINE**.

The screenshot shows the Tekton Dashboard interface. On the left, a sidebar lists navigation options: Tekton resources, Pipelines, PipelineRuns, PipelineResources, Tasks, ClusterTasks, TaskRuns, Conditions, EventListeners, Triggers, TriggerBindings, ClusterTriggerBindings, TriggerTemplates, and ClusterInterceptors. The 'PipelineRuns' section is selected, showing a list of runs. One run, 'build-deploy-dummy-tekton-ace-sj249', is highlighted with a green box and labeled 'Succeeded'. Below this, a dropdown menu shows the selected pipeline 'ejecutar-creacion' with a green checkmark. The main panel displays the details of this pipeline run, including its duration (3 minutes 38 seconds) and a 'Logs' tab. The logs show the execution of a Docker build and deployment process, with a final status of 'Completed'.

Finalmente, luego de **TERMINADO** el procesamiento del **PIPELINE**, se realiza el **TEST** de validación del **MICROSERVICIO** desplegado funcione correctamente, con la **MODIFICACIÓN** realizada:

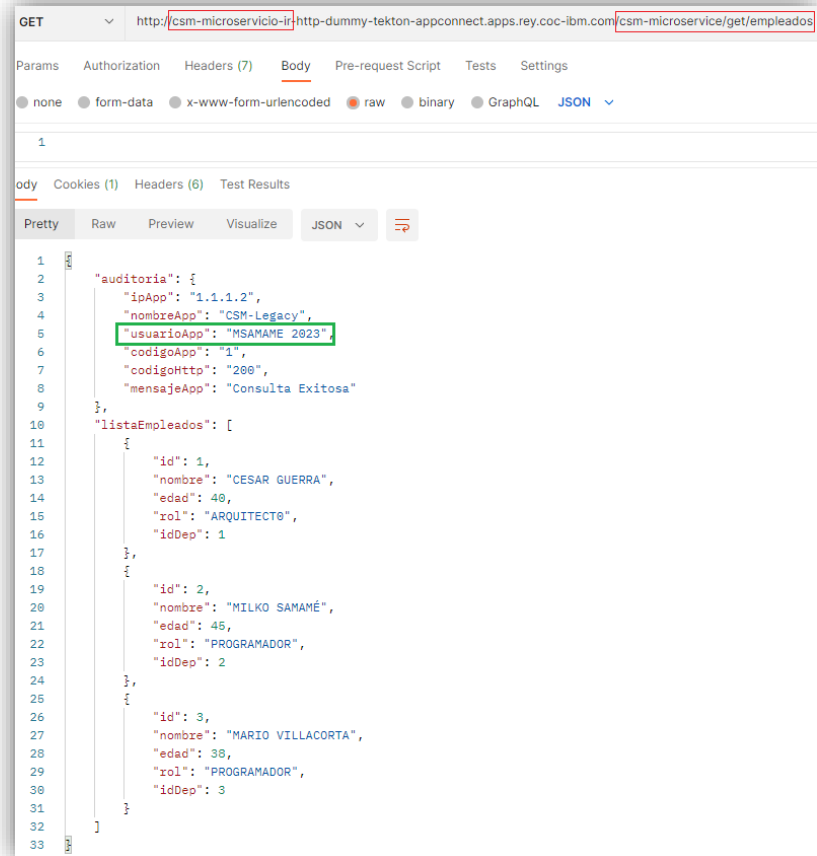
- ✓ **INTEGRATION SERVER.**
<http://csm-microservicio-is-http-dummy-tekton-appconnect.apps.rey.coc-ibm.com/csm-microservice/get/empleados>
- ✓ **INTEGRATIO RUNTIME.**
<http://csm-microservicio-ir-http-dummy-tekton-appconnect.apps.rey.coc-ibm.com/csm-microservice/get/empleados>



```

GET http://csm-microservicio-is-http-dummy-tekton-appconnect.apps.rey.coc-ibm.com/csm-microservice/get/empleados

Body
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
[{"auditoria": [{"ipApp": "1.1.1.2", "nombreApp": "CSM-Legacy", "usuarioApp": "MSAMAME 2823", "codigoApp": "1", "codigoHttp": "200", "mensajeApp": "Consulta Exitosa"}], "listaEmpleados": [{"id": 1, "nombre": "CESAR GUERRA", "edad": 40, "rol": "ARQUITECTO", "idDep": 1}, {"id": 2, "nombre": "MILKO SAMAM\u00c9", "edad": 45, "rol": "PROGRAMADOR", "idDep": 2}, {"id": 3, "nombre": "MARIO VILLACORTA", "edad": 38, "rol": "PROGRAMADOR", "idDep": 3}]}]
  
```



```

GET http://csm-microservicio-ii-http-dummy-tekton-appconnect.apps.rey.coc.ibm.com/csm-microservice/get/empleados

Params Authorization Headers (7) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON

1

body Cookies (1) Headers (6) Test Results
Pretty Raw Preview Visualize JSON

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
[{"auditoria": {"ipApp": "1.1.1.2", "nombreApp": "CSM-Legacy", "usuarioApp": "MSAMAME 2823", "codigoApp": "1", "codigoHttp": "200", "mensajeApp": "Consulta Exitosa"}, "listaEmpleados": [{"id": 1, "nombre": "CESAR GUERRA", "edad": 40, "rol": "ARQUITECTO", "idDep": 1}, {"id": 2, "nombre": "MILKO SAMAM\u00c9", "edad": 45, "rol": "PROGRAMADOR", "idDep": 2}, {"id": 3, "nombre": "MARIO VILLACORTA", "edad": 30, "rol": "PROGRAMADOR", "idDep": 3}]}]
```

REFERENCIAS:

- ✓ <https://tekton.dev/docs>
- ✓ https://www.ibm.com/docs/en/app-connect/containers_cd?topic=releases-release-notes-app-connect-operator-820
- ✓ https://www.ibm.com/docs/en/app-connect/containers_cd?topic=obtaining-app-connect-enterprise-server-image-from-cloud-container-registry
- ✓ <https://dalelane.co.uk/blog/?p=4676#image-builder-pipeline>