

Fase 4 - Activitat 11.3: Descarrega automàtica dins de la màquina de producció de noves versions d'una aplicació pujades a Github des de la màquina de desenvolupament utilitzant Jenkins, ngrok i Webhooks de Github.

0- Identificació del grup i activitat:

Curs: ASIX2

Projecte: PJ9 DevOps i Cloud Computing

Fase: 4

Activitat: 11.3

Grup/Individual: Individual

Membres:

1.- Objectiu de l'activitat

L'objectiu d'aquesta activitat és la creació d'un projecte de **Jenkins** per descarregar automàticament dins de la màquina **producció** cada nova versió del codi de projecte **pizzas** que es puja al seu dipòsit de **Github** des de la màquina **desenvolupament**. Per aconseguir aquest objectiu hem de:

- Configurar el projecte **pizzas** de **Github** per treballar amb **Webhooks**.
- Crear i configurar un projecte de **Jenkins** que treballi amb el dipòsit de **pizzas** de **Github** i amb les seves **Webhooks**.
- Caldrà **ngrok** perquè **Github** i **Jenkins** es puguin comunicar.
- Caldren les **Webhooks** del dipòsit **pizzas** de **Github** per començar automàticament el procés d'enviament de la nova versió del codi a la màquina de **producció** via **Jenkins**.

2.- Abans de començar

a) Posa en marxa les màquines virtuals **desenvolupament** i **producció**.

b) Dins de la màquina **producció** entra a la carpeta **ngrok** i executa:

```
ngrok http 8080
```

per aconseguir fer pública l'aplicació **Jenkins** de la teva màquina virtual.

c) Comprova que la màquina és accessible de manera pública.

3- Generació d'un token API Secret de Jenkins per ser utilitzat pels Webhooks de Github

a) Selecciona dins del **Dashboard** de **Jenkins** l'opció **Configuració de Jenkins** → **Users** (secció **Security**). A continuació selecciona el teu usuari d'accés a **Jenkins**.

b) A continuació selecciona l'opció **Security**.

c) Selecciona **API Token** → **Add new token** → **Generate** i genera un nou **Token** de **Jenkins**.

d) Copia el valor generat per poder utilitzar-lo amb **Webhooks** de **Github** al següent apartat.

e) Torna al **Dashboard**.

4.- Configuració del projecte de Jenkins per treballar amb Webhooks de Github

- a) Accedeix al projecte **pizzas** de **Jenkins**. Selecciona l'opció **Configura**.
- b) A la secció **Build Triggers** selecciona: **GitHub hook trigger for GITScm polling**
- c) Selecciona **Desa** per desar la nova configuració.

Ara **Jenkins** ha estat configurat perquè cap cop que es pugin des de la màquina **desenvolupament** noves versions de l'aplicatiu **pizzas** al dipòsit **pizzas** del teu compte d'usuari de **Github**, automàticament es dispararà la tasca per descarregar-les dins de la màquina **produccio**.

5.- Configuració de Webhooks de Github

- a) Accedeix al teu dipòsit **pizzas** de **Github**. A continuació accedeix a **Settings** → **Webhooks** i fes clic a **Add Webhook**.
- b) A la secció **Payload** escriu l'adreça URL pública que has obtingut amb **ngrok** afegint **/github-webhook/** a la URL. Per exemple si **ngrok** et dona **https://47ff-79-153-220-213.ngrok-free.app**, s'haurà d'escriure **https://47ff-79-153-220-213.ngrok-free.app/github-webhook/**
- c) A la secció **Content type** selecciona: **applications/json**.
- d) Afegeix el **Token** creat a l'apartat **3** dins de la secció **Secret**.
- e) A la secció **SSL Verification** selecciona: **Enable SSL verification**.
- f) A la secció **Which events would you like to trigger this webhook?** selecciona: **Just de push event**.
- g) Habilita l'opció **Active**.
- h) Fes clic a **Add Webhook**.

Ara **Github** ha estat configurat per avisar automàticament a **Jenkins** que s'ha pujat una nova versió del codi al dipòsit **pizzas**.

6.- Prova de funcionament abans de lliurar l'activitat

- a) Entra a la carpeta **pizzas** dins de la màquina **desenvolupament**. Modifica **pizza.html**. Canvia la línia **17** i canvia **dacomo** per **xyyzz** (a on **xx** són les dues primeres lletres del teu nom, **yy** les dues primeres lletres del teu primer cognom i **zz** les dues primeres lletres del teu segon cognom).
- b) Actualitza el dipòsit local i sincronitza-ho amb el remot. Torna a la carpeta **pizzas** (que és a on tens el dipòsit Git local) i executa:

```
git add *
```

```
git commit -m "Commit 2 de l'aplicatiu pizzas"
```

```
git push -u origin main
```

i comprova que s'ha pujat la nova versió de **pizzas** a **Github**.

c) Fes les següents comprovacions per assegurar-te que la nova versió s'ha descarregat automàticament a la màquina producció:

- Accedeix a la web d'administració del teu compte de **Github** i a continuació accedeix a **Settings** → **Webhooks**. Refresca la pàgina. Comprova que el Webhook que havies creat a l'apartat 5 té un tick verd ✓ i el missatge **Last delivery was successful**.
- Accedeix a la web d'administració de l'eina **Jenkins** instal·lada a la màquina **produccio** i selecciona **pizzas**. Refresca la pàgina. Comprova que s'ha creat un nou **Build** que abans no existia i que tu **NO** has posat en marxa manualment.
- Entra dins del **Build** que s'ha creat automàticament i comprova a la dreta que surt un missatge informant que el **Build** va ser efectivament va ser iniciat per un **push** del teu usuari de **Github**.
- Dins de la tasca **pizzas** de **Jenkins**, a la secció **Espai de treball Build**, comprova que el fitxer **pizzas.html** té el mateix codi que el fitxer **pizzas.html** dins de la màquina **desenvoluament**.
- Comprova que l'hora donada per **Jenkins** de descarrega a la màquina **produccio** coincideix amb l'hora donada per **Github** de pujada del codi.

Lliurament de l'activitat

NOTA: Les següents tasques s'han de fer al meu davant de manera que jo pugui veure personalment la comprovació de funcionament i poder validar el lliurament de l'activitat.

a) Entra a la carpeta **pizzas** dins de la màquina **desenvolupament**. Modifica **pizza.php**. Afegeix el paràgraf **Autor** amb el teu nom complet.

b) Actualitza el dipòsit local i sincronitza-ho amb el remot. Executa:

```
git add *  
git commit -m "Commit 3 de l'aplicatiu pizzas"  
git push -u origin main
```

d) Comprova que s'ha descarregat la nova versió del projecte dins del directori /home/vagrant/projectes.

e) Data límit per obtenir el **100%** de la nota: **dimarts 5-2-25** a les **19.10**.