

ACTA DE REUNIÓN

El día 03/01/2017 a las 20:13 GMT/UTC +1, horario de invierno, el grupo de alumnos de la asignatura Evolución y Gestión de la Configuración del Grado en Ingeniería Informática - Ingeniería del Software de la Escuela Técnica Superior de Ingeniería Informática formado por Javier Bonilla, Juan Domínguez, Daniel Moreno, Miguel Rodríguez y Álvaro Valencia Parra fueron convocados a reunión con el siguiente orden del día:

CONFIGURAR JENKINS

Javier Bonilla consiguió, de forma individual, configurar Jenkins para que fuese capaz de detectar los push efectuados a los repositorios de GitHub. Como primer punto del orden del día, Javier Bonilla explicó al resto de los miembros del grupo el procedimiento a seguir para dicha configuración, que es el siguiente:

- Descargar ngrock, descomprimirlo y ejecutarlo mediante el siguiente comando: `./ngrock http 8080`
- Ir al repositorio de GitHub, settings, webhooks y pulsar sobre añadir webhook.
- Añadir como URL la que nos ha dado ngrock (descrita como "forwarding" dentro de la consola de ngrock), y añadir a dicha URL lo siguiente: `"/github-webhook"`.
- Creamos el job en Jenkins, vamos a configure, y en project URL, ponemos la URL SSH de nuestro repositorio.
- Dentro de Source Code Management, todas las URL de los repositorios deben ser HTTPS. En el apartado de Repositories, la URL debe acabar en `.git`, en Branches to build se deben indicar las ramas en las que se deben detectar los push para activar la tarea, y en URL al final del cuadro de configuración la URL debe ser HTTPS. Repository browser debe ser `"githubweb"` (sin las comillas).
- En Build Triggers, indicamos que el build se debe ejecutar cuando se haga un push a GitHub.
- Guardamos los cambios, y el siguiente paso es ir a la configuración de Jenkins: `manage Jenkins > configure system`.
- Nos dirigimos al apartado de GitHub > GitHub Servers, e indicamos como API RUL la siguiente: `"https://api.github.com"`. Para añadir las credenciales, le damos a `Advanced > manage additional github actions > convert login and password to token > from login and password`. Nos pedirá nuestras credenciales de GitHub. Debemos indicarla, darle a guardar y actualizar la página. Nos dirigimos al mismo apartado (GitHub servers), y en credentials, seleccionamos la que acabamos de crear, y pulsamos `"test connection"`.

Una vez realizado todo esto, Jenkins será capaz de registrar los push de GitHub.

CONFIGURAR TAREAS DE JENKINS

El siguiente paso fue configurar la tarea a llevar a cabo cuando Jenkins detecte el push. La tarea sería ejecutar un fichero `.sh` dentro de la máquina de Vagrant de Agora. Para ello, instalamos el plugin de Vagrant, y le damos permiso a Jenkins para que fuese capaz de ejecutar comandos de Vagrant (este paso es necesario, porque instalamos Vagrant como root user). Para ello, hicimos lo siguiente:

- Modificar el fichero `/etc/default/jenkins`, y cambiar la variable `$JENKINS_USER` a `"root"` (comillas incluidas).

- A continuación, ejecutamos los siguientes comandos:

```
chown -R root:root /var/lib/jenkins
```

```
chown -R root:root /var/cache/jenkins
```

```
chown -R root:root /var/log/jenkins
```

- Reiniciamos Jenkins: `service jenkins restart`.

Con esto, Jenkins es capaz de ejecutar comandos de Vagrant como root. A continuación, añadimos la tarea a realizar con el build. Para ello, nos dirigimos a la configuración del proyecto, y en la sección de build, pulsamos sobre Add build step > Run a command in vagrant machine. Especificamos el Vagrantfile path (/home/developer/egc/agora/Vagrantfile), el nombre de la máquina de Vagrant (default), y el comando a ejecutar (bash /home/vagrant/ode/build-elections.sh). El equipo decidió que el repositorio con el que debe trabajar Jenkins debe estar situado en la carpeta /code/temp, dentro de la carpeta compartida de la máquina virtual. De esta forma, cada vez que se haga un cambio en la rama master o develop, se hará un pull en el repositorio correspondiente de la carpeta temp, y se efectuaría el build se ese repositorio.

CIERRE DE LA REUNIÓN

La reunión se cierra a las 0:42 GMT/UTC +1, horario de invierno, del día siguiente. Se convoca una reunión para el día siguiente, CUATRO de ENERO del año DOS MIL DIECISIETE, hora por determinar. Se acuerda que cada miembro del equipo debe configurar el resto de repositorios por su cuenta, realizando los pasos indicados.

Y para que conste, firman los asistentes a la reunión:

Javier Bonilla

Miguel Rodríguez

Juan Domínguez

Daniel Moreno

Álvaro Valencia

