

# Despliegue de Joomla en OpenShift

---

## Bloque III: Trabajando con el hosting

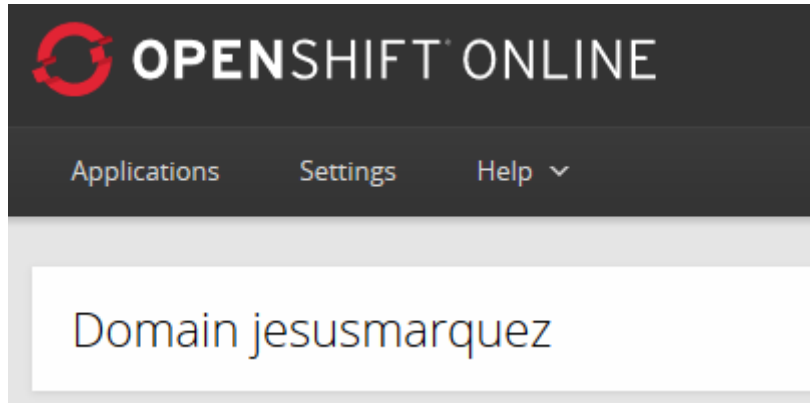
Jesús Márquez Racero

21/01/2016



## 1. Crea un nuevo namespace.

Cada cuenta de usuario en OpenShift Online está asociada a un “espacio de nombres” para generar un FQDN único para cada gear. En la configuración inicial de la cuenta habrá que seleccionar un espacio de nombres que sea único, este espacio de nombres se aplicará automáticamente a todos los gears que se creen.



## 2. Crea las claves ssh e incorpóralas a OpenShift.

Abrimos una consola en debian y con el siguiente comando crearemos un par de claves, una privada que solo tendremos nosotros y una pública que será con la que a continuación utilizaremos para sincronizarnos con OpenShift.

```
usuario@debian218:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/usuario/.ssh/id_rsa):
/home/usuario/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/usuario/.ssh/id_rsa.
Your public key has been saved in /home/usuario/.ssh/id_rsa.pub.
The key fingerprint is:
0f:43:a3:83:58:bd:77:0d:ed:c2:f3:59:59:a1:8e:14 usuario@debian218
The key's randomart image is:
+---[RSA 2048]---+
|      E      .|
|      . . . . .|
|    . . 0 . . . .|
|   o . + o . + o |
|  . . + S = . o . o |
|    o = + o      |
|      . o        |
+---+-----+
usuario@debian218:~$
```

A continuación nos vamos a .ssh que es la carpeta en la cual se guarda el par de claves, visualizaremos la clave pública y la copiaremos para vincularla con OpenShift.

```

usuario@debian218:~$ cd .ssh/
usuario@debian218:~/.ssh$ ls
id_rsa  id_rsa.pub  known_hosts
usuario@debian218:~/.ssh$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDS5Np78iYSVkiNrBM1avEqGfcKgGcDLnTGRFnwDSO
jZCXLorljNDRI2LCc7jKk50CM5hnIGExustqAHCkpolxwV4KSbjHpyEj/G5zCKv6IP1BEKfcVhBZLC
ntBxghCpS/ILR20zWTQLZHLpFIrSNyA6Wl6a9EW3752EUwEz0i1Pqu2X1bvRMjiQeYRnaZkh41kCrM6f
B80s0M9JgDf6sXc7j3iat0RdWfzN+Lo4x5PgQQS1cFTSGSunm21wRn7QyqL8EJXZjagIYTbjmh2WFjQ4
EQHYt+/SAJfNzoc9yUpiy3jfqV+YkU/R5wQqV040gFOAgxnSyHlaAxohbdTP usuario@debian218
usuario@debian218:~/.ssh$ █

```

Nos iremos a OpenShift y entramos en la pestaña *Settings*, una vez aquí elegimos añadir una nueva clave. Ahora pegaremos la clave pública que copiamos anteriormente en el cuadro blanco que se muestra a la derecha y guardamos.

Settings

You need to set a public key before you can work with application code

Public Keys

OpenShift uses a public key to securely encrypt the connection between your local machine and your application and to authorize you to upload code. You must create a private and public key on your local machine and then upload the public key before you can connect to your applications' Git repositories or remotely access your applications.

[Learn more about SSH keys.](#)

Paste the contents of your public key file (.pub)

U11PquZX1bVkmj1QeYRnaZkh41kCrM6fB80s0M9JgDf6sXc7j3iat0RdWfzN+Lo4x5PgQQS1cFTSGSunm21wRn7QyqL8EJXZjagIYTbjmh2WFjQ4EQHYt+/SAJfNzoc9yUpiy3jfqV+YkU/R5wQqV040gFOAgxnSyHlaAxohbdTP  
 usuario@debian218

Save

Y como vemos ya está nuestra clave pública creada y vinculada en OpenShift.

Settings

✓ Your public key has been created

Public Keys

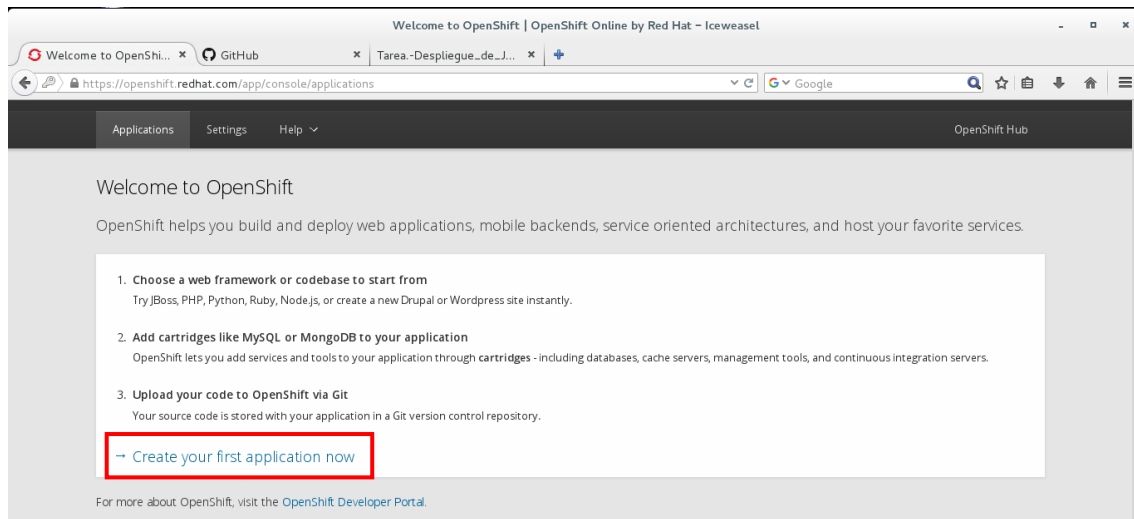
OpenShift uses a public key to securely encrypt the connection between your local machine and your application and to authorize you to upload code. [Learn more about SSH keys.](#)

Key name	Type	Contents	
default	ssh-rsa	AAAAB3Nza.AxohbdTP	Delete

Add a new key...

### 3. Crea una nueva aplicación: joomla basada en PHP 5.4. Añadele Mysql 5.5

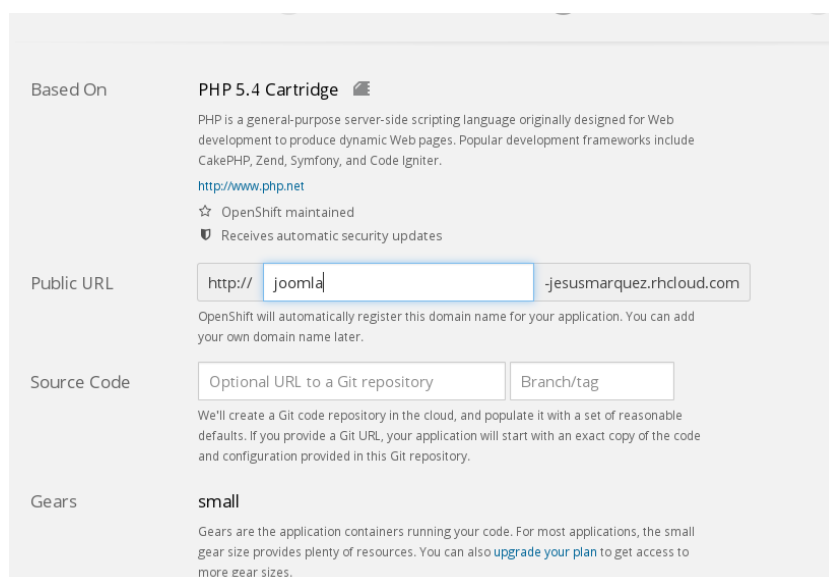
En primer lugar entramos en nuestra cuenta de OpenShift y elegiremos crear nuestra primera aplicación.



En la siguiente pestaña nos sale varios cartridge que nos ofrece OpenShift según el gear que queramos crear, nosotros en primer lugar instalaremos Php la versión 5.4 que es la que necesitaremos para instalar Joomla.

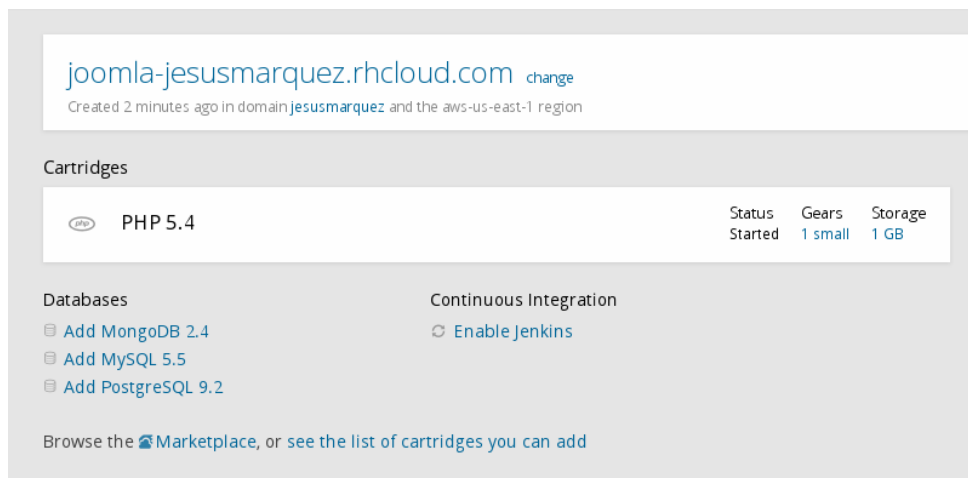


Ahora le daremos el nombre con el cual queremos que al escribirlo en el navegador nos salga a nuestra página de joomla.



Como podemos ver aquí tenemos instalado nuestro php5.4, y ahora tendremos que instalar también mysql la versión 5.5 y phpmyadmin aunque este no es necesario

para la instalación de joomla pero con el podremos administrar nuestras bases de datos en caso de cualquier error.



joomla-jesusmarquez.rhcloud.com [change](#)  
Created 2 minutes ago in domain [jesusmarquez](#) and the [aws-us-east-1](#) region

**Cartridges**

Icon	Cartridge	Status	Gears	Storage
	PHP 5.4	Started	1 small	1 GB

**Databases**

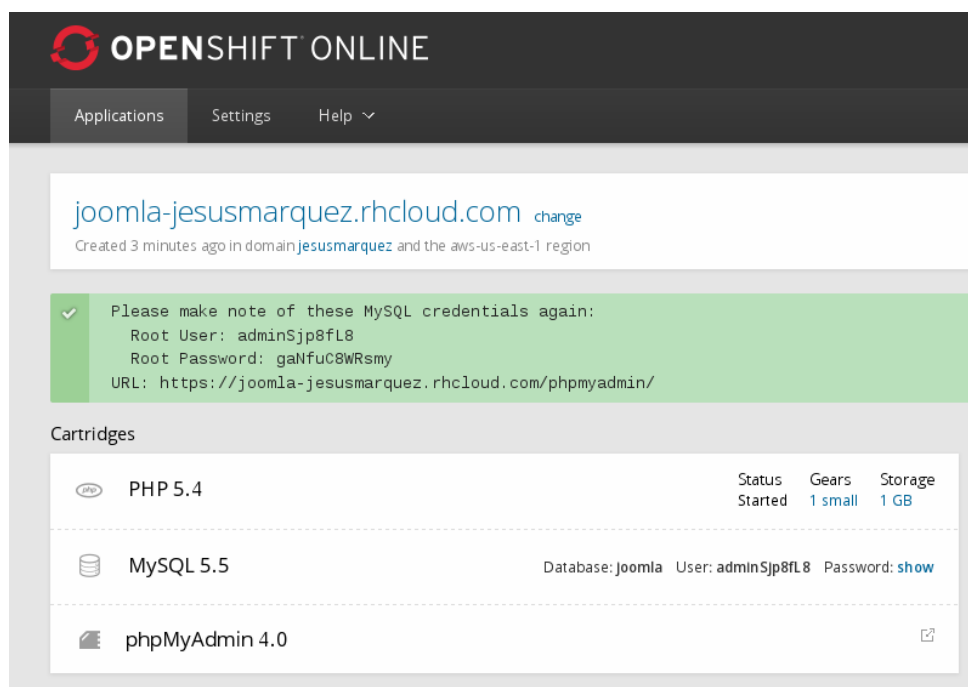
- [Add MongoDB 2.4](#)
- [Add MySQL 5.5](#)
- [Add PostgreSQL 9.2](#)

**Continuous Integration**

- [Enable Jenkins](#)

Browse the [Marketplace](#), or [see the list of cartridges you can add](#)

Como vemos aquí tenemos instalado todo los requisitos necesario para la instalación de joomla ya que apache viene instalado por defecto.



**OPENSIFT ONLINE**

Applications Settings Help ▾

joomla-jesusmarquez.rhcloud.com [change](#)  
Created 3 minutes ago in domain [jesusmarquez](#) and the [aws-us-east-1](#) region

✓ Please make note of these MySQL credentials again:  
Root User: adminSjp8fL8  
Root Password: gaNfuC8WRsmy  
URL: <https://joomla-jesusmarquez.rhcloud.com/phpmyadmin/>

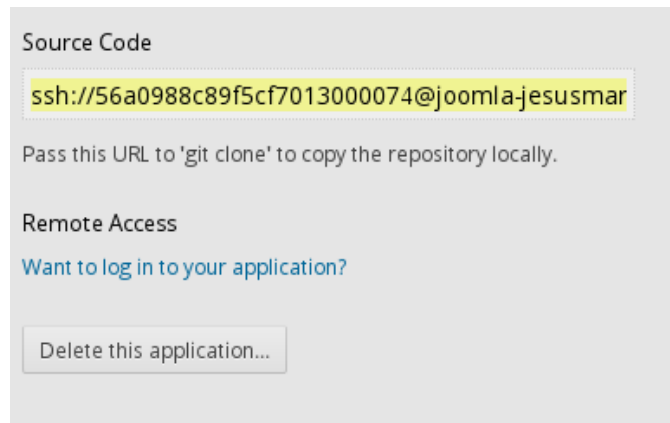
**Cartridges**

Icon	Cartridge	Status	Gears	Storage
	PHP 5.4	Started	1 small	1 GB
	MySQL 5.5			
	phpMyAdmin 4.0			

Database: joomla User: adminSjp8fL8 Password: [show](#)

#### 4. Clonamos el repositorio remoto a nuestro equipo local

Copiamos el código siguiente que se encuentra en nuestra aplicación creada, este código se utiliza para copiar la carpeta de nuestro repositorio remoto a local pudiendo sincronizarlos y subir archivos desde local a remoto.



Nos vamos a nuestra consola y clonamos nuestro repositorio.

```
usuario@debian218: ~  
Archivo  Editar  Ver  Buscar  Terminal  Ayuda  
usuario@debian218:~$ git clone ssh://56a0988c89f5cf7013000074@joomla-jesusmarque  
z.rhcloud.com/~git/joomla.git/  
Cloning into 'joomla'...  
The authenticity of host 'joomla-jesusmarquez.rhcloud.com (54.172.193.200)' can'  
t be established.  
RSA key fingerprint is cf:ee:77:cb:0e:fc:02:d7:72:7e:ae:80:c0:90:88:a7.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added 'joomla-jesusmarquez.rhcloud.com,54.172.193.200' (RSA  
) to the list of known hosts.  
remote: Counting objects: 19, done.  
remote: Compressing objects: 100% (15/15), done.  
remote: Total 19 (delta 0), reused 19 (delta 0)  
Receiving objects: 100% (19/19), 18.25 KiB | 0 bytes/s, done.  
Checking connectivity... done.  
usuario@debian218:~$
```

Una vez se ha clonado observamos que se ha creado una carpeta que se llama joomla y contiene un index.php que es la página que muestra por defecto.

```
usuario@debian218:~$ cd joomla/  
usuario@debian218:~/joomla$ ls  
index.php  
usuario@debian218:~/joomla$
```

## 5. Nos descargamos Joomla utilizando wget y lo descomprimos en nuestro entorno de OpenShift

A continuación con wget nos descargaremos joomla comprimido de la página oficial.

```
usuario@debian218:~$ wget https://github.com/joomla/joomla-cms/releases/download  
/3.4.8/Joomla_3.4.8-Stable-Full_Package.zip
```

En el siguiente paso creamos una carpeta que se llame joomla y el nombre de nuestro equipo para descomprimir aquí los ficheros de instalación y de configuración de joomla.

```
usuario@debian218:~/joomla$ mkdir joomla218
```

Ahora procedemos a descomprimirlo dentro de la carpeta que hemos creado.

```
usuario@debian218:~/joomla$ unzip /home/usuario/Joomla_3.4.8-Stable-Full_Package.zip -d joomla218/
```

Aquí vemos que todo se ha descomprimido correctamente.

```
usuario@debian218:~/joomla$ ls
index.php  joomla218
usuario@debian218:~/joomla$ ls joomla218/
administrator  htaccess.txt  joomla.xml  logs          robots.txt.dist
bin             images         language    media         templates
cache          includes      layouts     modules       tmp
cli            index.php     libraries   plugins       web.config.txt
components     installation  LICENSE.txt README.txt
```

## 6. Lo subimos a OpenShift

Ahora vamos a subir los archivos que hemos creado dentro de nuestro repositorio local al remoto mediante los siguientes comandos.

Con git add seleccionamos los archivos que hemos modificado o creado dentro del repositorio, en nuestro caso los seleccionaremos todos.

Con git comit -m pondremos un comentario de lo que vamos a subir para saber que hemos hecho en cada momento.

```
usuario@debian218:~/joomla$ git add .
usuario@debian218:~/joomla$ git commit -m "Instalacion de Joomla"
```

Por último con git push subiremos los archivos a OpenShift.

```

usuario@debian218:~/joomla$ git push
warning: push.default is unset; its implicit value has changed in
Git 2.0 from 'matching' to 'simple'. To squelch this message
and maintain the traditional behavior, use:

    git config --global push.default matching

To squelch this message and adopt the new behavior now, use:

    git config --global push.default simple

When push.default is set to 'matching', git will push local branches
to the remote branches that already exist with the same name.

Since Git 2.0, Git defaults to the more conservative 'simple'
behavior, which only pushes the current branch to the corresponding
remote branch that 'git pull' uses to update the current branch.

See 'git help config' and search for 'push.default' for further information.
(the 'simple' mode was introduced in Git 1.7.11. Use the similar mode
'current' instead of 'simple' if you sometimes use older versions of Git)

```

## 7. Iniciamos la instalación utilizando las variables de entorno.

Modificamos el archivo de configuración de joomla (configuration.php-dist) con las variables de entorno, estas variables se utilizan para que busque sola los parametros de configuración que necesita joomla (base de datos, usuario de la base de datos, etc) y no sea necesario introducirlos a la hora de la instalación por lo que nos ahorraremos un paso.

```

GNU nano 2.2.6      Fichero: configuration.php

public $captcha = '0';
public $list_limit = '20';
public $access = '1';

/* Database Settings */
public $dbtype = 'mysqli';           // Normally mysqli
public $host = 'getenv('OPENSIFT_MYSQL_DB_HOST') . ':' . getenv('OPENSIFT_MYSQL_DB_PORT'); // DB host
public $username = 'getenv('OPENSIFT_MYSQL_DB_USERNAME'); // DB username
public $password = 'getenv('OPENSIFT_MYSQL_DB_PASSWORD'); // DB password
public $db = 'getenv('OPENSIFT_APP_NAME'); // DB name
public $dbprefix = 'jos_';           // Do not change unless you need it

```

Ahora guardaremos el archivo con otro nombre (configuration.php) y borraremos el otro archivo de configuración (configuration-dist.php) ya que nos puede dar error a la hora de instalar.

```

usuario@debian218:~/joomla/joomla218/installation$ ls
application      COPYRIGHT      helper        LICENSES      sql
configuration.php CREDITS       index.php     localise.xml  template
configuration.php-dist favicon.ico    INSTALL      model         view
controller       form          language     response
usuario@debian218:~/joomla/joomla218/installation$ rm -r configuration.php-dist
usuario@debian218:~/joomla/joomla218/installation$

```



Volvemos a proceder al paso de subir al repositorio remoto los archivos que hemos modificado, para ello volveremos a proceder a hacer los pasos que hicimos anteriormente.

```
usuario@debian218:~/joomla$ git add .
usuario@debian218:~/joomla$ git commit -m "Archivos modificados"
[master 6fbec15] Archivos modificados
1 file changed, 4 insertions(+), 4 deletions(-)
rename joomla218/installation/{configuration.php-dist => configuration.php} (88%)
usuario@debian218:~/joomla$
```

```
usuario@debian218:~/joomla$ git push
warning: push.default is unset; its implicit value has changed in
Git 2.0 from 'matching' to 'simple'. To squelch this message
and maintain the traditional behavior, use:
```

Por último procedemos a la instalación para ello nos vamos al navegador y escribiremos la dirección que nos da del repositorio barra y el nombre de la carpeta que contiene los archivos de instalación y configuración.

En este paso de la instalación utilizaremos los datos que nos proporciona mysql en OpenShift.

Cartridges			
	PHP 5.4	Status Started	Gears 1 small Storage 1 GB
	MySQL 5.5	Database: joomla User: admin\$jp8fL8 Password: show	
	phpMyAdmin 4.0		

Rellenamos los campos con esos datos.

1 Configuración 2 Base de datos 3 Visión general

### Configuración de la base de datos

← Anterior → Siguiente

Tipo de base de datos \*   
Probablemente sea "mysqli"

Hospedaje \*   
Normalmente es "localhost"

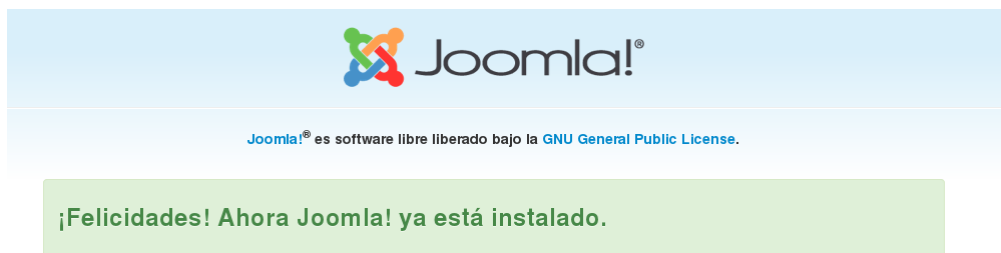
Usuario \*   
Algo como "root" o un nombre de usuario facilitado por quien le sirva el hospedaje

Contraseña   
Por cuestiones de seguridad, es primordial usar una contraseña para la cuenta de su base de datos.

Base de datos \*   
En algunos hospedajes solo se permite el nombre específico de una base de datos por sitio. En esos casos, si le interesa instalar más de un sitio, puede usar el prefijo de las tablas para distinguir entre los sitios de Joomla! que usen la misma base de datos.

Prefijo de las tablas \*   
Elija un prefijo para la base de datos o use el **generado aleatoriamente**. Lo óptimo es que sea de tres o cuatro caracteres de largo y que contenga solo caracteres alfanuméricos, y DEBE acabar con un guión bajo.

Como vemos ya se ha instalado Joomla, una vez instalado antes de administrarlo pulsaremos en el botón de eliminar carpeta de instalation para que nadie pueda volver a instalarlo en caso de que acceda a nuestros datos.



Aquí estamos ya en la página principal de nuestro Joomla!



***Definiciones importantes:***

- **Gear:** Es un contenedor dentro de una máquina virtual con unos recursos limitados para que pueda ejecutar sus aplicaciones un usuario de OpenShift.

- **Cartridge:** Son contenedores de software preparados para ejecutarse en un gear. En principio sobre cada gear pueden desplegarse varios cartridges.