

Paas Público



como herramienta pedagógica

Cloud Computing

Carlos Serrano Sánchez

Contenido

1	Introducción	3
2	Openshift.....	3
2.1	Crear una cuenta OpenShift Gratuita.....	4
3	GitHub	7
4	Creación App en OpenShift.....	10
4.1	Cliente RHC.....	11
4.1.1	Instalar el cliente Git, si no lo tenemos aún instalado.	11
4.1.2	Instalar Ruby.....	12
4.1.3	Instalar cliente RHC	12
4.1.4	Configuración de RHC.....	13
5	Instalación de Catridges	14
6	Sincronización Openshift-Local	17
6.1	Comando RHC	20
7	Gestión de Base de datos.....	20
8	Putty para acceso SSH	23
9	Conclusiones.....	25
10	Bibliografía	26

1 Introducción

Este documento pretende ser una guía práctica de un caso de uso en el cual integramos un servidor local con Github y Openshift, en ningún momento se trata de un manual de uso y por tanto no asegura que la metodología de empleo sea la más óptima.

Para ello, voy a describir paso a paso dicho caso de uso, que aunque no es realmente práctico en un entorno real de desarrollo, sí nos da una buena idea del empleo de Openshift y Github.

La situación inicial en la que nos encontramos es la siguiente:

Tenemos un servidor Apache en local donde estamos desarrollando una aplicación web (PHP + MySQL). Para sincronizar nuestros repositorios y poder trabajar de modo efectivo con nuestro grupo vamos a crear un repositorio Github, que mantendremos sincronizado con el cliente de Windows.

En algún momento del desarrollo, vamos a decidir migrar la aplicación a pre-producción (lo llamamos así para hacer ver al alumno que no solo podemos usar OpenShift como entorno de producción sino también como entorno de desarrollo).

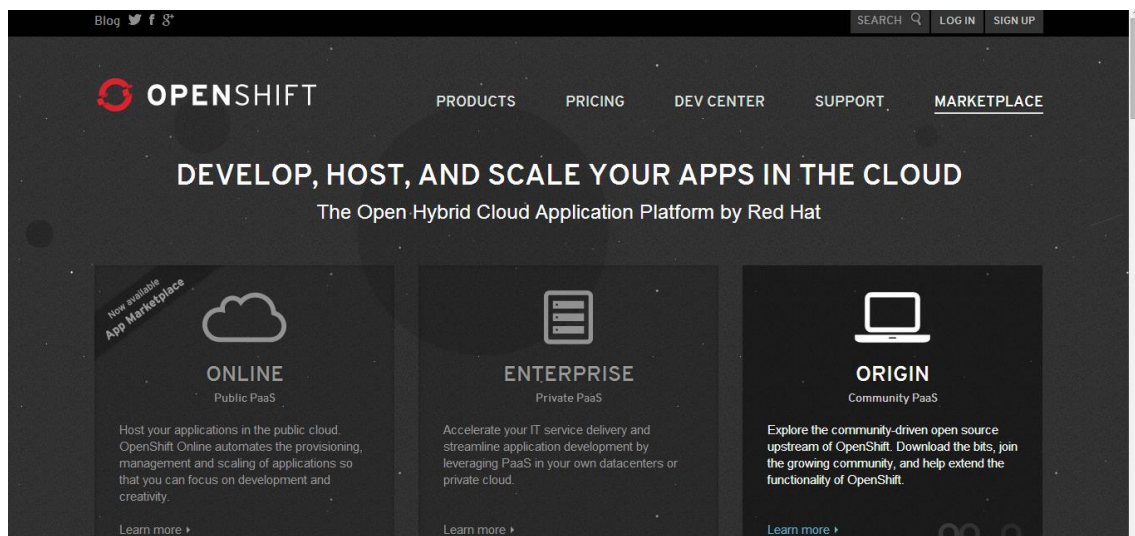
Openshift realizará un clonado del repositorio Github que hemos empleado en la primera fase y a partir de aquí iremos detallando más opciones de Openshift y su mantenimiento.

2 Openshift

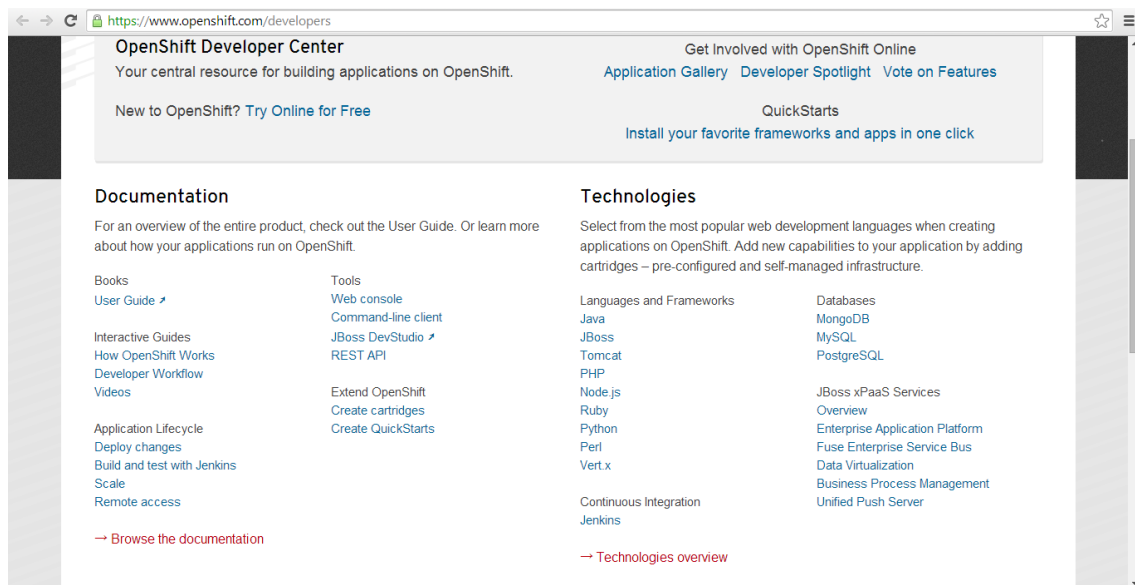
Antes de comenzar, vamos a introducir al protagonista de esta guía, Openshift.

Openshift.com es una empresa/sitio web que integra la solución PAAS Openshift desarrollada por Red Hat en su arquitectura física de servidores. Openshift, como hemos dicho, es un PAAS, es decir, Platform As A Service. Traducido, es una pila de servicios que nos ofrecen la posibilidad de configurar un servidor basado en una serie de utilidades, funcionalidades y compatibilidades: PHP, JAVA, Ruby, instalación de base de datos... Todo lo que ofrecía el hosting convencional, pero desde la visión del cloud computing y su elasticidad.

Por tanto, Openshift.com desarrolla dicha pila de servicios en sus servidores y la ofrece en varias modalidades, desde totalmente gratuito hasta de pago, en función de algunos parámetros como son los gears empleados. Algo así como las instancias de servidor que arranca, o lo que es lo mismo, potencia.



Existen muchas posibilidades de configuración de este PAAS, y cada vez irán integrando más.



2.1 Crear una cuenta OpenShift Gratuita

Vamos a crear una cuenta gratuita de OpenShift y estudiar algunas de sus posibilidades. Lo primero que necesitamos es una cuenta de correo activa y a continuación rellenar el formulario de alta.

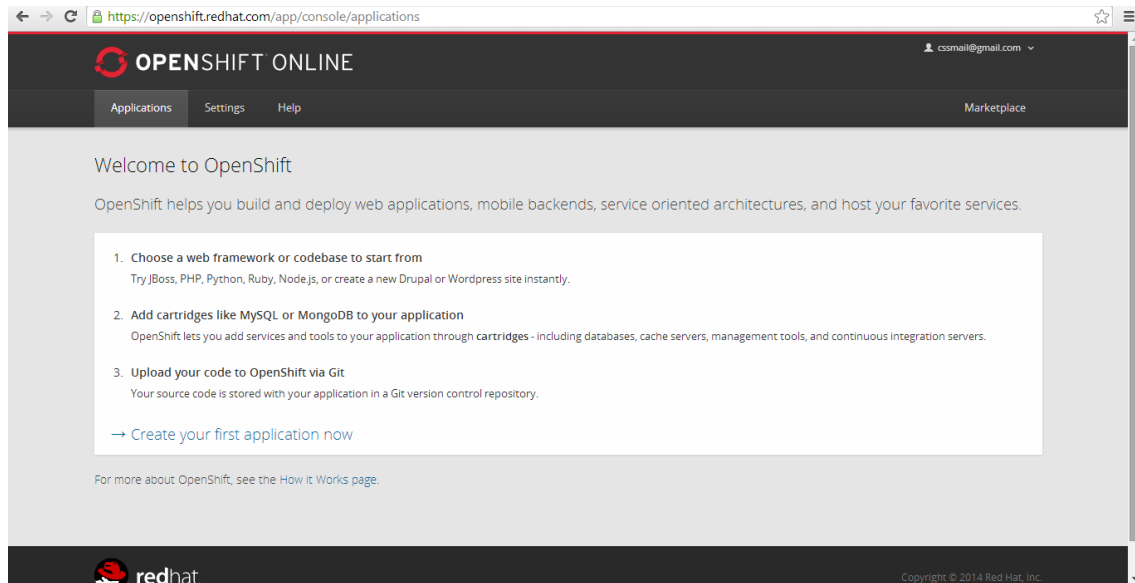
The screenshot shows the OpenShift account creation page. At the top, the OpenShift logo is displayed. Below it, the heading "Create an account" is followed by a form with the following fields: "Email address" (with a placeholder "Valid email address"), "Password" (with a placeholder "At least 6 characters"), "Password confirmation" (with a placeholder "Enter it again"), and "Are you a spam bot?" (with a placeholder "Type the words that appear below" and a CAPTCHA image). Below the CAPTCHA are links for "Get Another", "Get an audio CAPTCHA", and "Help". A small text line reads "reCAPTCHA provided by Google - help fight spam and fix books!". At the bottom, a link for "Terms of Service" is present. To the right of the form, the text "Already have an account? [Sign in](#)" is shown. Below this, a section titled "You'll love OpenShift because it has:" lists several features: built-in support for various languages, powerful command line and web management tools, pre-created quickstarts, a vibrant community, and a wide range of developer resources. A link "Learn more about [OpenShift](#)" is at the bottom right.

Tras verificar la cuenta, siguiendo las instrucciones del correo que recibimos, procedemos a autenticarnos en el sistema.

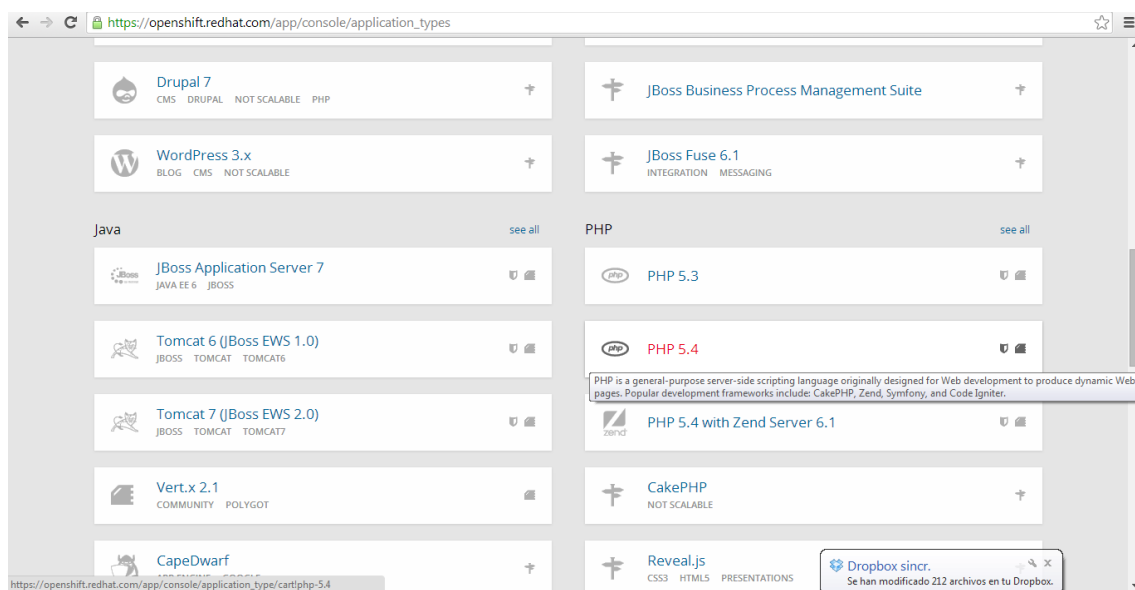
The screenshot shows the OpenShift login page. At the top, the OpenShift logo is displayed. Below it, the heading "Sign in to OpenShift" is followed by a form with the following fields: "OpenShift Online or Red Hat account" (with a placeholder "Case sensitive") and "Password" (with a placeholder "Forgot your password?"). Below the form is a "Sign in" button. To the right of the form, the text "Need an account? [Create One](#)" is shown. Below this, a section titled "Features Released — April 09, 2014" lists several features: enhanced log storage and rotation, new JBoss data virtualization cartridge, new RHC cloning and filtering commands, option to set dependencies on other web cartridges, and improvements to the 10gen-mms-agent. To the right of this section, a section titled "Upcoming Events" lists two events: "Webinar: How to Develop and Deploy PHP Apps in the Cloud" (June 18) and "MongoDB World 2014" (June 23 - New York New York). At the bottom left, a small text line reads "Esperando a googleads.g.doubleclick.net...". At the bottom right, a small text line reads "TiConf Amsterdam".

Es una plataforma que nos guía de una forma excepcionalmente clara el paso a paso a seguir, teniendo en cuenta la complejidad del sistema. La primera vez que accedemos, tras darnos la

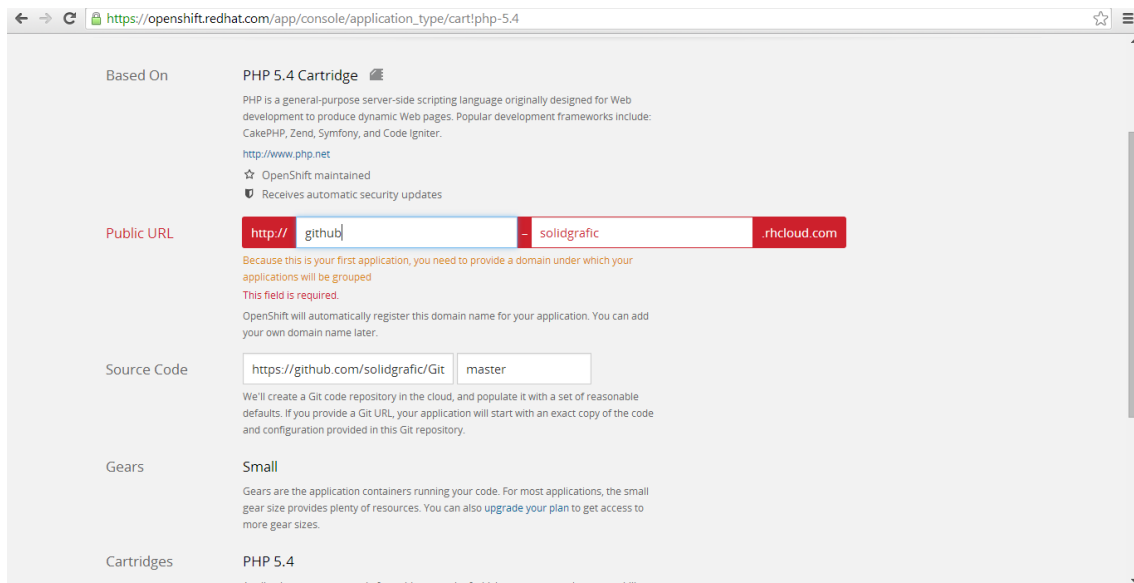
bienvenida, nos invita a crear nuestra primera aplicación.



Tal como hemos dicho existen una gran variedad de servicios que podemos activar en nuestra plataforma. Nosotros en esta guía vamos a optar por configurar la plataforma con compatibilidad con PHP5 , sin instalar ningún CMS concreto.



Una vez seleccionada la plataforma básica, toca configurar los parámetros más concretos de nuestra app.



Debemos asignar una url de acceso (podremos aparcas nuestro DNS si lo tenemos). Y lo más importante, podemos asignar un repositorio GitHub donde se encuentra el código fuente. De esta forma, se realiza una sincronización y copia automática de dichos contenidos y tendríamos nuestra app y su código subido a OpenShift automáticamente.

Y aquí es donde vamos a realizar un paréntesis. Aunque el objetivo principal de este documento no es Github como tal, vamos aprovechar para dar también un caso de uso de este servicio.

3 GitHub

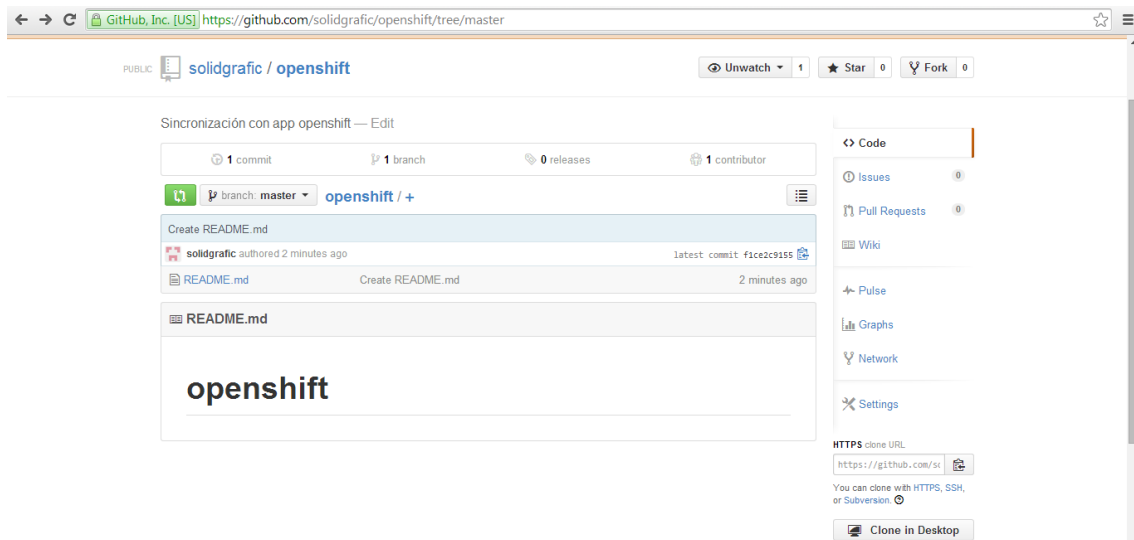
GitHub es una sitio web/empresa que ha llegado para quedarse durante mucho tiempo en el mundo de los desarrolladores. En su plataforma integra el famoso protocolo de control de versiones git, además de ofrecer un sinfín de posibilidades varias, como la creación de páginas...

OpenShift, de hecho, emplea esta plataforma para mantener sus repositorios. De hecho, más adelante explicaremos uno de los mecanismos para emplearlo.

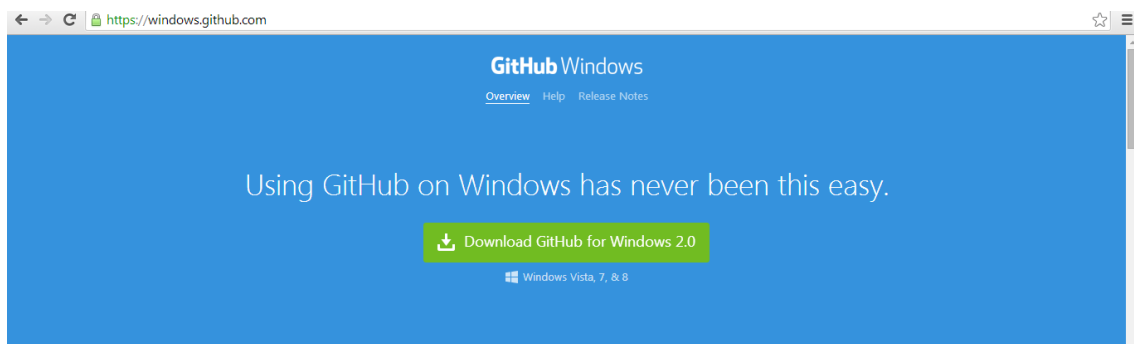
No obstante, en este punto nos vamos a centrar, exclusivamente, en el uso de GitHub. Vamos a partir de supuesto en el que estamos desarrollando una app en local y necesitamos tener nuestros repositorios sincronizados, puesto que trabajamos con un grupo de desarrolladores, gestores, diseñadores...

Me doy de alta en la web de Github, donde existe una versión de cuenta gratuita que te permite crear los repositorios que necesites, pero la condición es que han de ser públicos. En caso de que necesites que sean privados, deberás pasar a una versión de pago.

Tras el alta correspondiente, creamos un repositorio (normalmente tantos repositorios como proyectos).

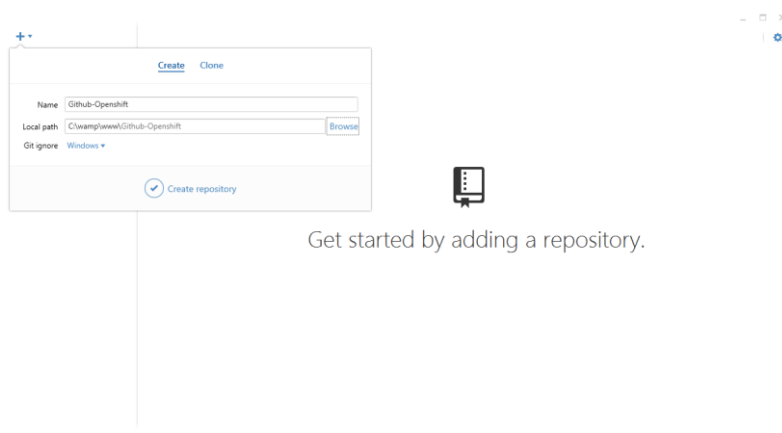


Hemos creado uno llamado openshift. Como se observa en la esquina derecha inferior, existe un acceso directo a la descarga de un cliente para clonado automático de repositorios (Clone in Desktop).

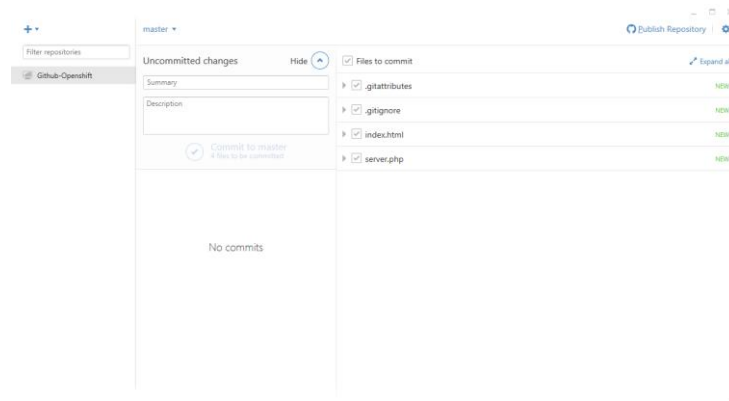


Tenemos dicho cliente disponible en diferentes plataformas. Yo, personalmente, lo voy a instalar en Windows 7 64 bits.

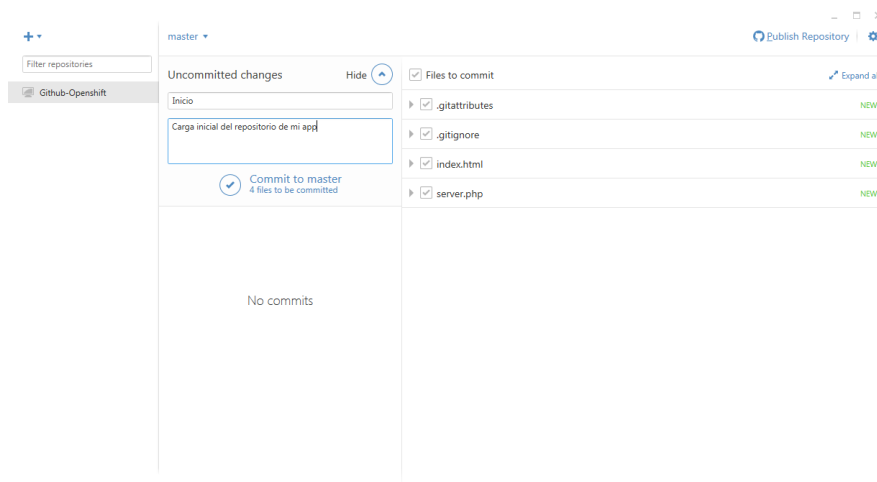
A partir de aquí su uso es fácil e intuitivo.



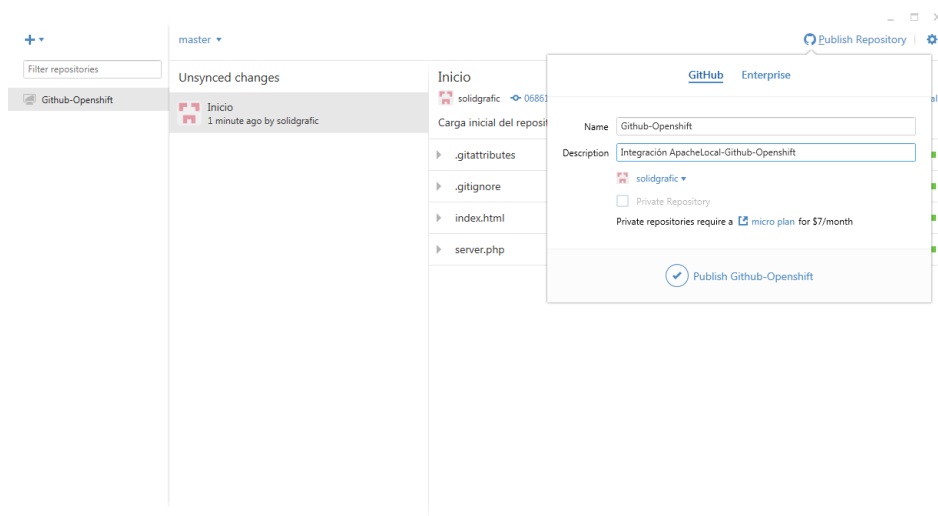
Te permite crear nuevos repositorio, o clonar repositorios existentes en Github. Sincroniza periódicamente o manualmente. Gestiona los comentarios de los "commit". Es realmente intuitivo.



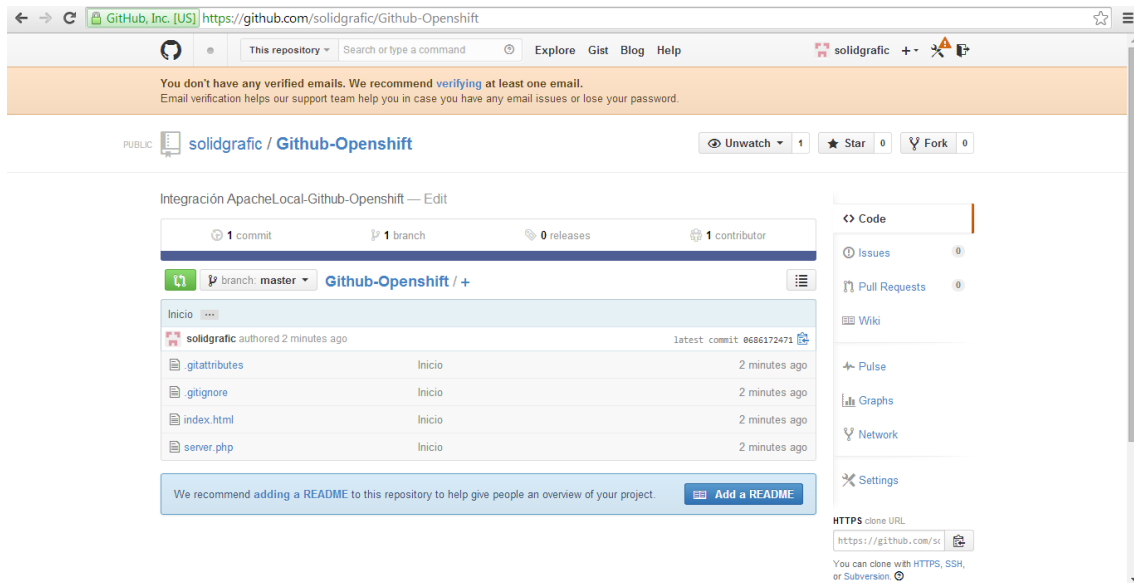
Como vemos, hacemos el simulacro de cargar nuestro app en dicho repositorio. Solo consta de dos archivos: server.php e index.html.



En la siguiente imagen, vemos que podemos crear un repositorio en local y luego publicar a nuestra cuenta de Github.

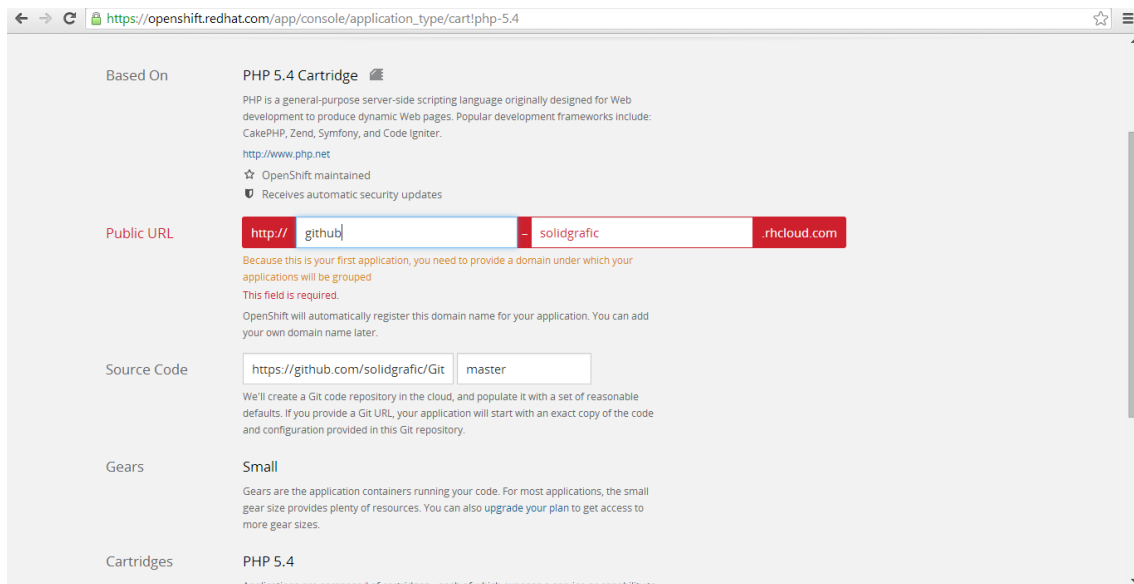


Automáticamente, tendremos nuestros archivos en la nube.

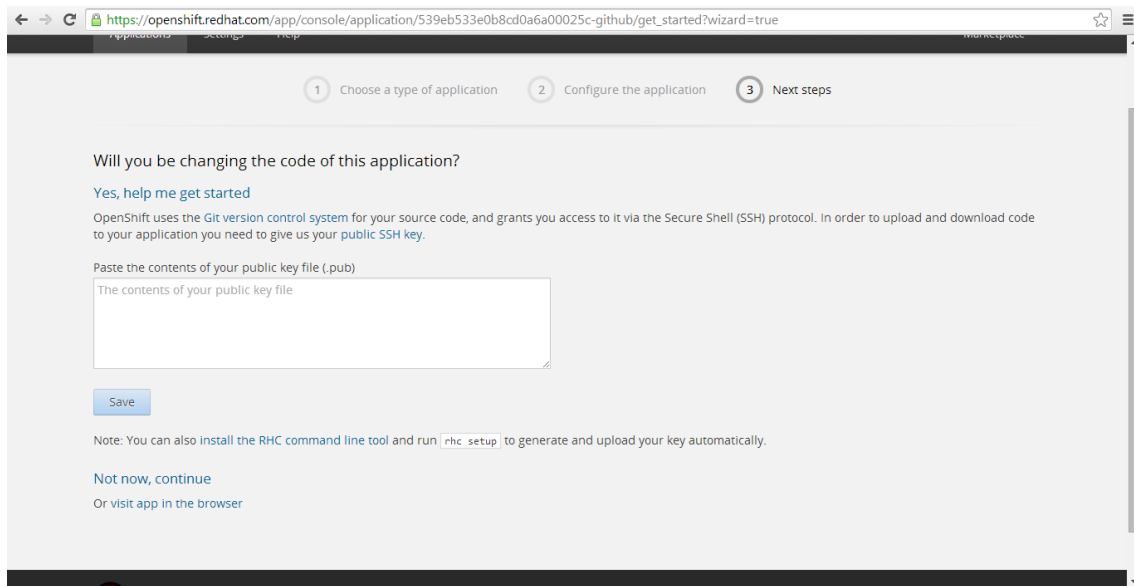


4 Creación App en OpenShift

Como vimos, anteriormente, cuando creamos una cuenta de openshift y accedemos por primera vez, nos invita a crear una app. Partamos de esta pantalla.



En ella podemos observar algunos parámetros. La URL la cual define nuestro punto de acceso a dicha app-web. Y el repositorio Git que contiene nuestro código. En este ejemplo, lo he “ligado”, con el repositorio que he creado antes para el desarrollo. De esta forma, en cuenta haga clic a siguiente, se creara la plataforma y la aplicación con el código que exista en dicho repositorio, concretamente: server.php e index.html.



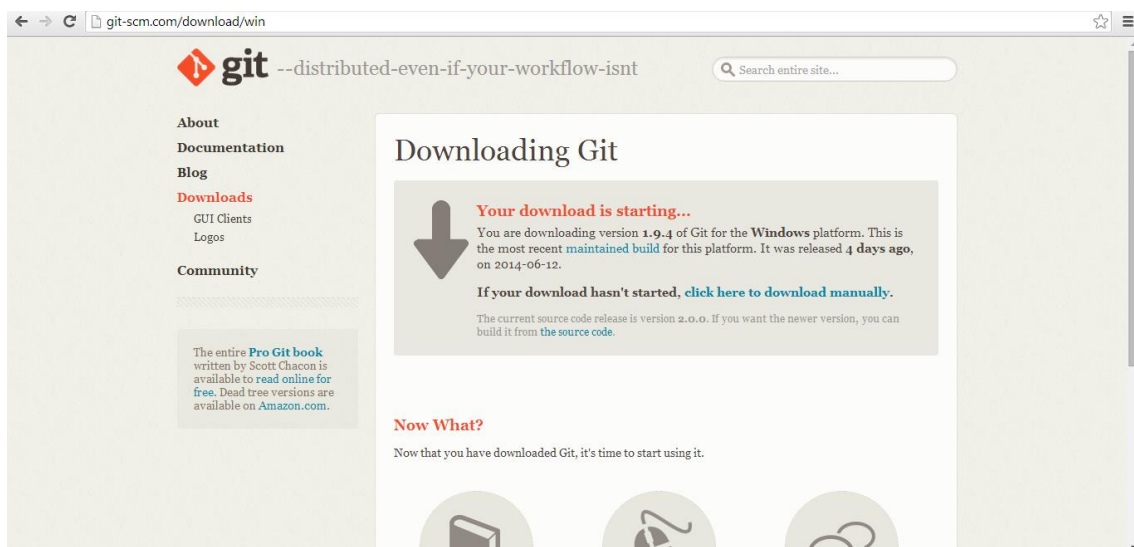
En la siguiente página nos invita a insertar la clave pública del servicio de Git para poder sincronizar los archivos automáticamente. En esta guía no se realiza este paso y a continuación nos centramos en el mantenimiento y puesta en marcha de nuestra app en OpenShift.

4.1 Cliente RHC

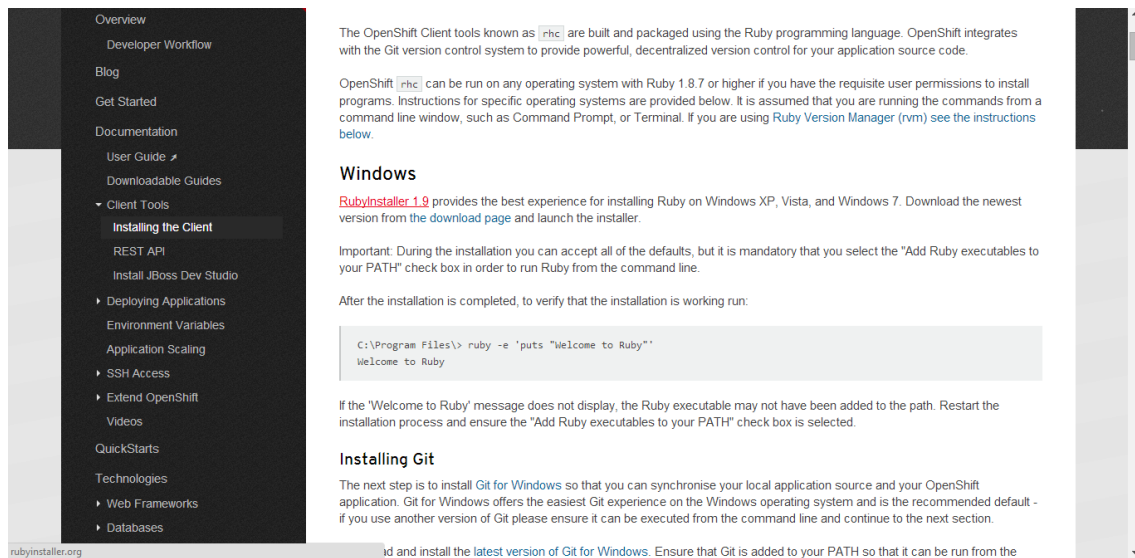
En la última página, antes visitada, ya nos advierte la web de OpenShift que nos recomienda instalar el cliente RHC para poder sincronizar nuestro repositorio local con el de OpenShift.

El manual de la propia web lo explica clarísimamente, pero los pasos fundamentales son:

4.1.1 Instalar el cliente Git, si no lo tenemos aún instalado.



4.1.2 Instalar Ruby



The screenshot shows the RubyInstaller 1.9 website. On the left is a dark sidebar with a navigation menu. The main content area has a white background with black text. The sidebar menu includes: Overview, Developer Workflow, Blog, Get Started, Documentation, User Guide (with a sub-menu arrow), Downloadable Guides, Client Tools (with a sub-menu arrow), Installing the Client (highlighted), REST API, Install JBoss Dev Studio, Deploying Applications, Environment Variables, Application Scaling, SSH Access, Extend OpenShift, Videos, QuickStarts, Technologies, Web Frameworks, and Databases. The main content area contains the following text:

The OpenShift Client tools known as `[rhc]` are built and packaged using the Ruby programming language. OpenShift integrates with the Git version control system to provide powerful, decentralized version control for your application source code.

OpenShift `[rhc]` can be run on any operating system with Ruby 1.8.7 or higher if you have the requisite user permissions to install programs. Instructions for specific operating systems are provided below. It is assumed that you are running the commands from a command line window, such as Command Prompt, or Terminal. If you are using [Ruby Version Manager \(rvm\)](#) see the instructions below.

Windows

[RubyInstaller 1.9](#) provides the best experience for installing Ruby on Windows XP, Vista, and Windows 7. Download the newest version from the [download page](#) and launch the installer.

Important: During the installation you can accept all of the defaults, but it is mandatory that you select the "Add Ruby executables to your PATH" check box in order to run Ruby from the command line.

After the installation is completed, to verify that the installation is working run:

```
C:\Program Files\> ruby -e 'puts "Welcome to Ruby"'
Welcome to Ruby
```

If the 'Welcome to Ruby' message does not display, the Ruby executable may not have been added to the path. Restart the installation process and ensure the "Add Ruby executables to your PATH" check box is selected.

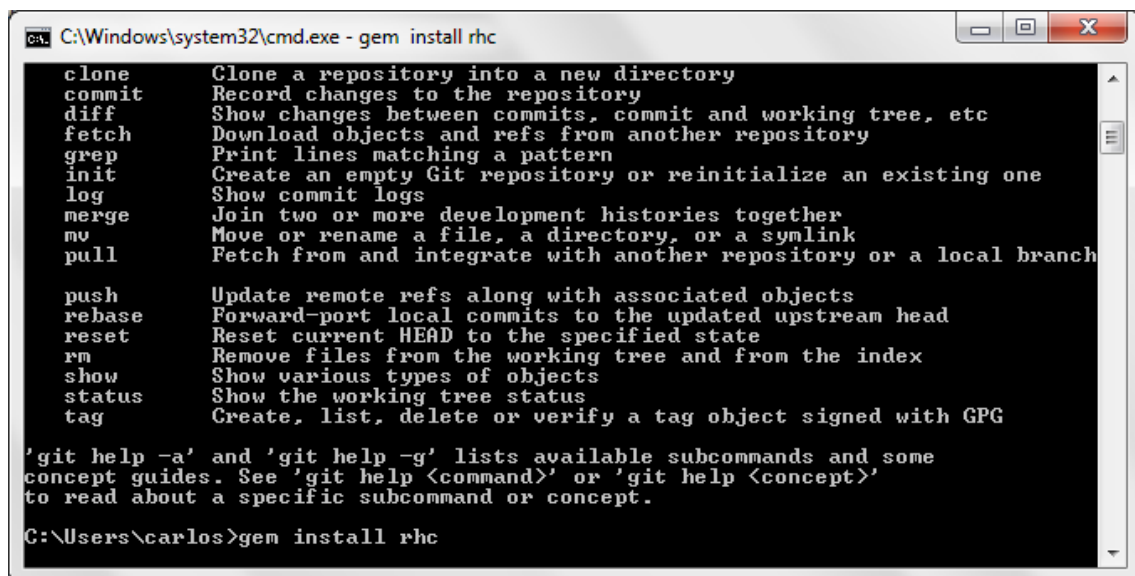
Installing Git

The next step is to install [Git for Windows](#) so that you can synchronise your local application source and your OpenShift application. Git for Windows offers the easiest Git experience on the Windows operating system and is the recommended default - if you use another version of Git please ensure it can be executed from the command line and continue to the next section.

id and install the latest version of Git for Windows. Ensure that Git is added to your PATH so that it can be run from the

Una vez instalado Git y Ruby, instalar el cliente RHC.

4.1.3 Instalar cliente RHC



The screenshot shows a Windows command prompt window with the title bar "C:\Windows\system32\cmd.exe - gem install rhc". The command prompt displays the output of the `gem install rhc` command, which lists various Git commands and their descriptions. The output is as follows:

```
clone      Clone a repository into a new directory
commit     Record changes to the repository
diff       Show changes between commits, commit and working tree, etc
fetch      Download objects and refs from another repository
grep       Print lines matching a pattern
init       Create an empty Git repository or reinitialize an existing one
log        Show commit logs
merge      Join two or more development histories together
mv         Move or rename a file, a directory, or a symlink
pull       Fetch from and integrate with another repository or a local branch

push       Update remote refs along with associated objects
rebase     Forward-port local commits to the updated upstream head
reset      Reset current HEAD to the specified state
rm         Remove files from the working tree and from the index
show       Show various types of objects
status     Show the working tree status
tag        Create, list, delete or verify a tag object signed with GPG

'git help -a' and 'git help -g' lists available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.

C:\Users\carlos>gem install rhc
```

La instalación tarda algunos minutos, pero suele ser limpia y finaliza sin problemas.

```
C:\Windows\system32\cmd.exe - gem install rhc
Fetching: rhc-1.25.3.gem (100%)
=====
If this is your first time installing the RHC tools, please run 'rhc setup'
=====
Successfully installed rhc-1.25.3
Parsing documentation for net-ssh-2.9.1
Installing ri documentation for net-ssh-2.9.1
Parsing documentation for net-scp-1.2.1
Installing ri documentation for net-scp-1.2.1
Parsing documentation for net-ssh-gateway-1.2.0
Installing ri documentation for net-ssh-gateway-1.2.0
Parsing documentation for net-ssh-multi-1.2.0
Installing ri documentation for net-ssh-multi-1.2.0
Parsing documentation for archive-tar-minitar-0.5.2
Installing ri documentation for archive-tar-minitar-0.5.2
Parsing documentation for highline-1.6.21
Installing ri documentation for highline-1.6.21
Parsing documentation for commander-4.2.0
Installing ri documentation for commander-4.2.0
Parsing documentation for httpclient-2.4.0
Installing ri documentation for httpclient-2.4.0
Parsing documentation for open4-1.3.4
```

4.1.4 Configuración de RHC

```
C:\Users\carlos>rhc setup
DL is deprecated, please use Fiddle
OpenShift Client Tools (RHC) Setup Wizard

This wizard will help you upload your SSH keys, set your application namespace,
and check that other programs like Git are properly installed.

Login to openshift.redhat.com:
```

Dicha configuración realizará una serie de operaciones, como es la creación de los pares de claves (privada y pública) necesarios para la sincronización de archivos con el repositorio git de OpenShift y la asociación de dicho repositorio con tu cuenta.

```
C:\Windows\system32\cmd.exe - rhc setup
Parsing documentation for httpclient-2.4.0
Installing ri documentation for httpclient-2.4.0
Parsing documentation for open4-1.3.4
Installing ri documentation for open4-1.3.4
Parsing documentation for rhc-1.25.3
Installing ri documentation for rhc-1.25.3
10 gems installed

C:\Users\carlos>rhc setup
DL is deprecated, please use Fiddle
OpenShift Client Tools (RHC) Setup Wizard

This wizard will help you upload your SSH keys, set your application namespace,
and check that other programs like Git are properly installed.

Login to openshift.redhat.com: cssmail@gmail.com
Password: *****

OpenShift can create and store a token on disk which allows to you to access the
server without using your password. The key is stored in your home directory and
should be kept secret. You can delete the key at any time by running 'rhc
logout'.
Generate a token now? (yes|no)
```

```
C:\Windows\system32\cmd.exe - rhc setup

This wizard will help you upload your SSH keys, set your application namespace,
and check that other programs like Git are properly installed.

Login to openshift.redhat.com: cssmail@gmail.com
Password: *****

OpenShift can create and store a token on disk which allows to you to access the
server without using your password. The key is stored in your home directory and
should be kept secret. You can delete the key at any time by running 'rhc
logout'.
Generate a token now? <yes/no> yes
Generating an authorization token for this client ... lasts about 1 month
Saving configuration to C:\Users\carlos\.openshift\express.conf ... done
No SSH keys were found. We will generate a pair of keys for you.

    Created: C:/Users/carlos/.ssh/id_rsa.pub

Your public SSH key must be uploaded to the OpenShift server to access code.
Upload now? <yes/no>
```

```
C:\Windows\system32\cmd.exe

found at https://www.openshift.com/developers/install-the-client-tools
We recommend these free applications:

* Git for Windows - a basic git command line and GUI client
https://github.com/msysgit/msysgit/wiki/InstallMSysGit
* TortoiseGit - git client that integrates into the file explorer
http://code.google.com/p/tortoisegit/
Checking common problems

Your private SSH key file should be set as readable only to yourself. Please
run 'chmod 600 C:\Users\carlos\.ssh\id_rsa'

Checking for a domain ... solidgrafic
Checking for applications ... found 1

github http://github-solidgrafic.rhcloud.com/

You are using 1 of 3 total gears
The following gear sizes are available to you: small

Your client tools are now configured.

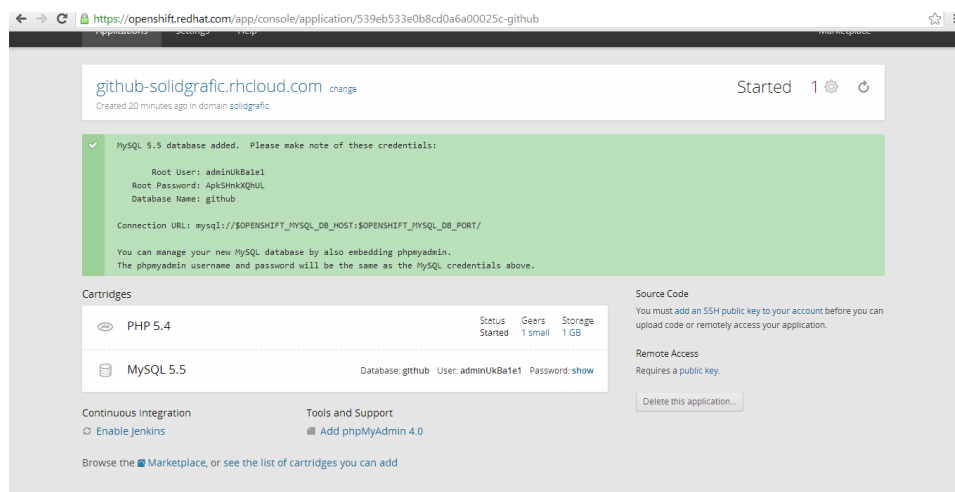
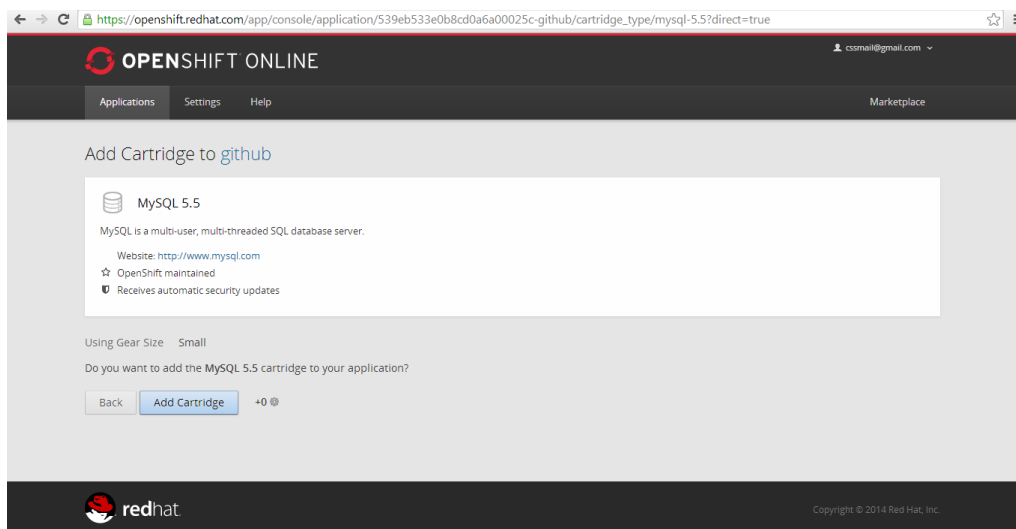
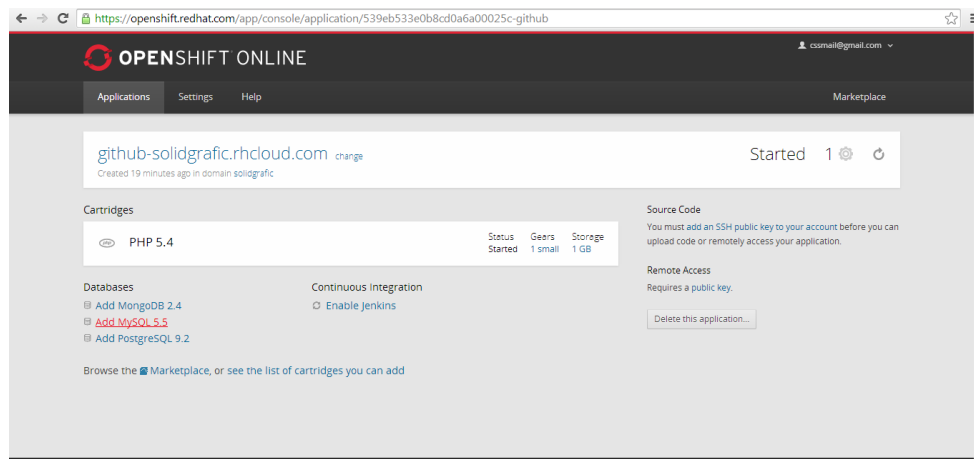
C:\Users\carlos>
```

Una vez finaliza dicho proceso estamos en disposición de mantener nuestra app. No obstante, primero acabaremos de configurarla.

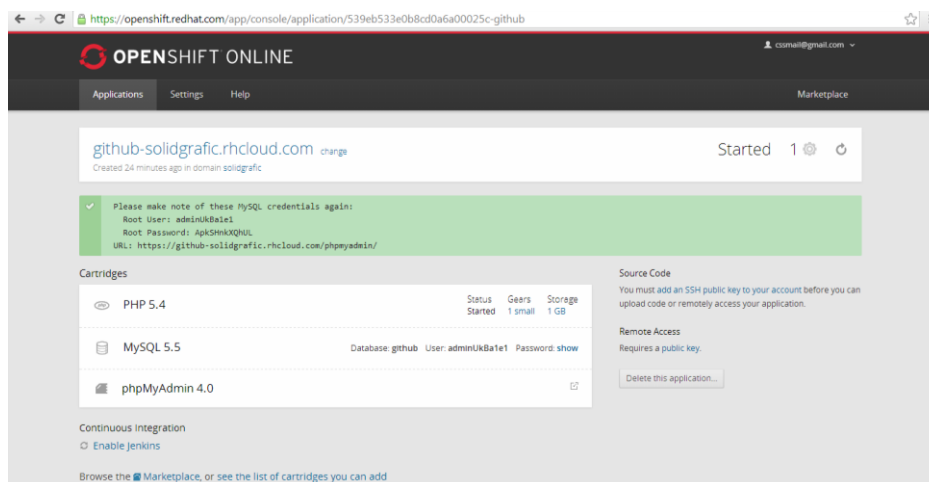
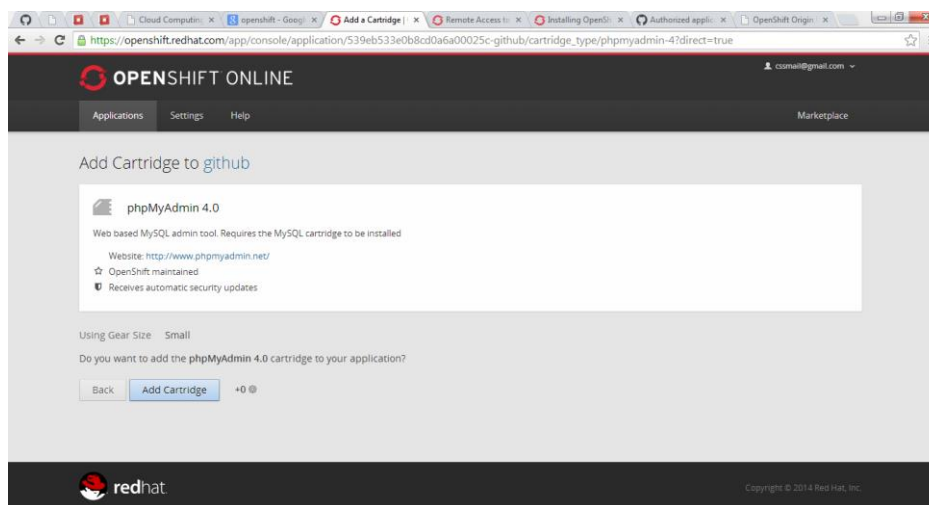
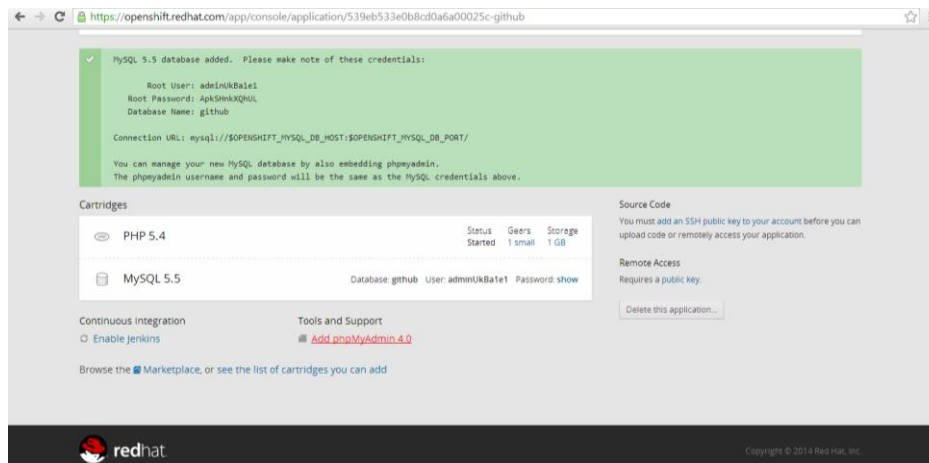
5 Instalación de Catridges

El concepto de catridge (cartucho) en Openshift viene a ser la posibilidad de activar alguno de los miembros de dicha pila de servicios que ofrece este PAAS. Es decir, instalar compatibilidad con PHP, o con una base de datos, o con un balanceador... Cada una de estas funcionalidades es un catridge.

A continuación vamos a activar la base de datos MySQL y PHPMyAdmin, en nuestra app que se llama github-solidgrafic.



Ahora añadimos PHPMyAdmin.



Ya estamos listos para comenzar a trabajar.

En primer lugar, visitamos nuestra URL para comprobar si nuestra app se ha sincronizado correctamente con nuestro código fuente en GitHub.

Trabajo final para el Curso de Cloud Computing

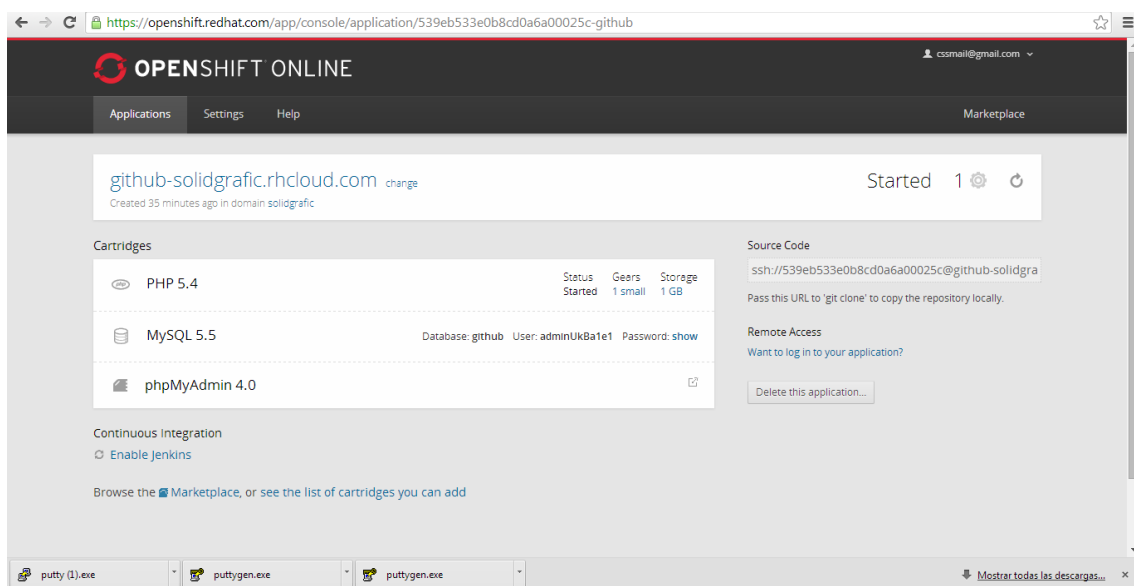
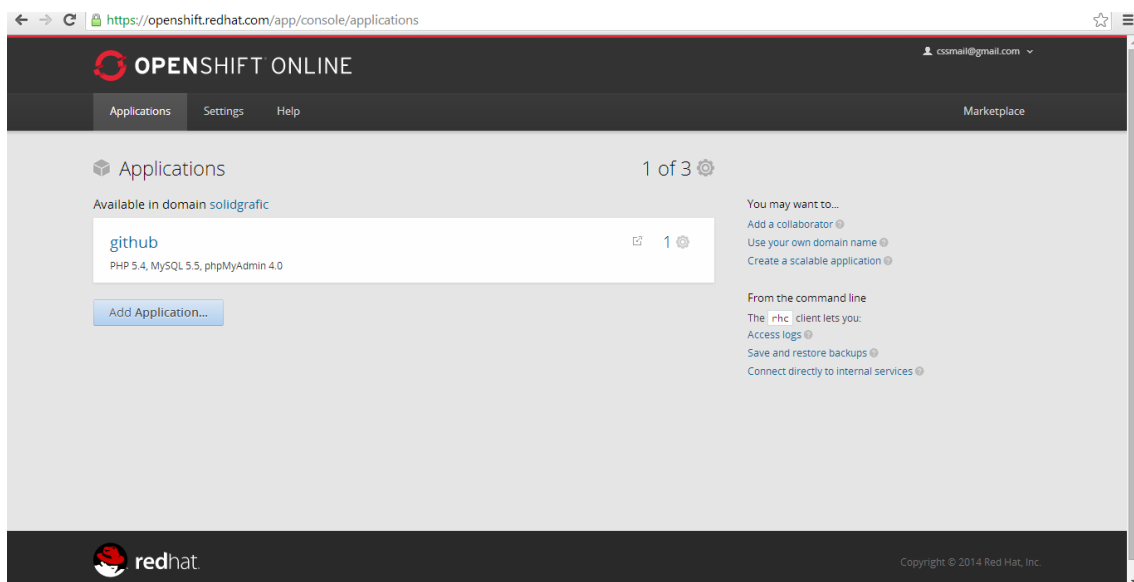
Sagrado Corazón de Jesús - La Salle

Carlos Serrano

Efectivamente, esta todo correcto. Pero nuestra app presenta problemas y queremos corregirlos. Concretamente la codificación e la página no es UTF-8. Pasemos al siguiente punto

6 Sincronización Openshift-Local

Antes de nada, necesitamos conocer los datos de acceso de nuestro repositorio. Para ello accedemos a nuestra cuenta de OpenShift y desde ahí podemos saber todo lo necesario.



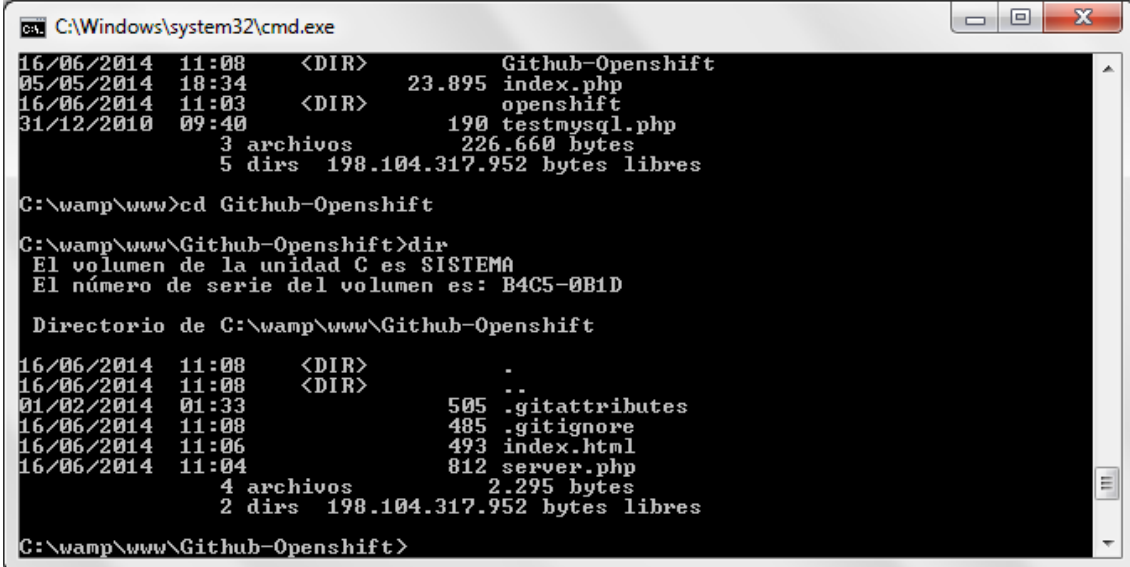
A la derecha de la web podemos observar la url de nuestro repositorio En este caso es:

ssh://539eb533e0b8cd0a6a00025c@github-solidgrafic.rhcloud.com/~/.git/github.git/

Más adelante, emplearemos esta información para conectarnos mediante ssh, en este caso la URL será:

539eb533e0b8cd0a6a00025c@github-solidgrafic.rhcloud.com

A continuación, me voy a mi carpeta de proyecto (en mi pc local).



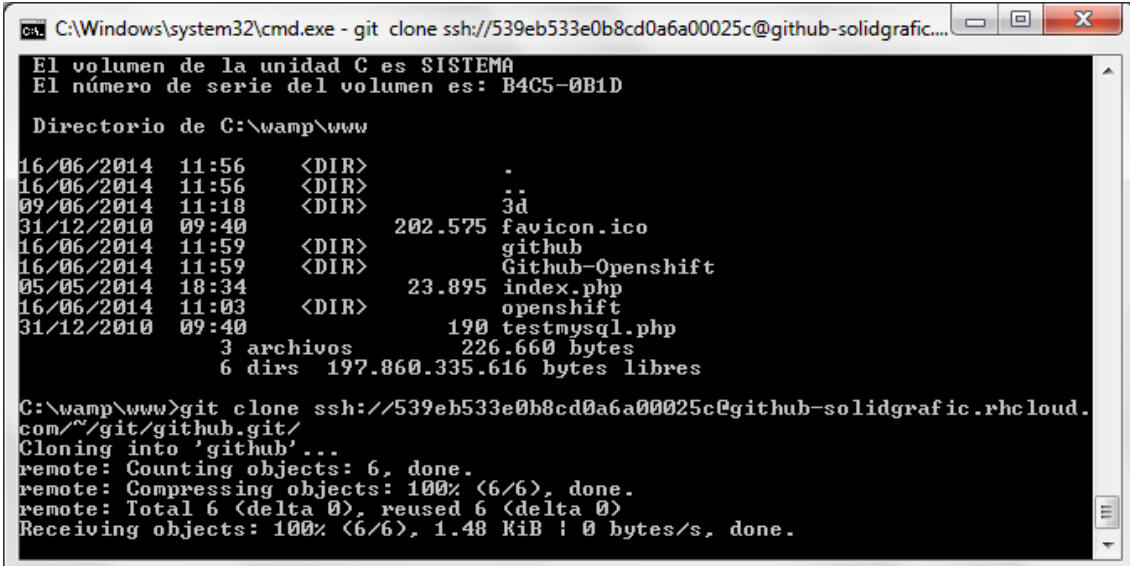
```
C:\Windows\system32\cmd.exe
16/06/2014 11:08 <DIR> Github-Openshift
05/05/2014 18:34 23.895 index.php
16/06/2014 11:03 <DIR> openshift
31/12/2010 09:40 190 testmysql.php
3 archivos 226.660 bytes
5 dirs 198.104.317.952 bytes libres

C:\wamp\www>cd Github-Openshift
C:\wamp\www\Github-Openshift>dir
El volumen de la unidad C es SISTEMA
El número de serie del volumen es: B4C5-0B1D

Directorio de C:\wamp\www\Github-Openshift
16/06/2014 11:08 <DIR> .
16/06/2014 11:08 <DIR> ..
01/02/2014 01:33 505 .gitattributes
16/06/2014 11:08 485 .gitignore
16/06/2014 11:06 493 index.html
16/06/2014 11:04 812 server.php
4 archivos 2.295 bytes
2 dirs 198.104.317.952 bytes libres

C:\wamp\www\Github-Openshift>
```

Y ahí realizo la clonación del repositorio.



```
C:\Windows\system32\cmd.exe - git clone ssh://539eb533e0b8cd0a6a00025c@github-solidgrafic....
El volumen de la unidad C es SISTEMA
El número de serie del volumen es: B4C5-0B1D

Directorio de C:\wamp\www
16/06/2014 11:56 <DIR> .
16/06/2014 11:56 <DIR> ..
09/06/2014 11:18 <DIR> 3d
31/12/2010 09:40 202.575 favicon.ico
16/06/2014 11:59 <DIR> github
16/06/2014 11:59 <DIR> Github-Openshift
05/05/2014 18:34 23.895 index.php
16/06/2014 11:03 <DIR> openshift
31/12/2010 09:40 190 testmysql.php
3 archivos 226.660 bytes
6 dirs 197.860.335.616 bytes libres

C:\wamp\www>git clone ssh://539eb533e0b8cd0a6a00025c@github-solidgrafic.rhcloud.
com/~/.git/github.git/
Cloning into 'github'...
remote: Counting objects: 6, done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 6 (delta 0), reused 6 (delta 0)
Receiving objects: 100% (6/6), 1.48 KiB | 0 bytes/s, done.
```

Una vez realizada esta operación, ya tengo los ficheros en local de Nuevo. Realizo la modificación oportuna, en este caso, conversión de index.html a UTF-8 y realizo la

sincronización.

```
C:\wamp\www\github>git add index.html
C:\wamp\www\github>
```

```
C:\wamp\www\github>git commit -m "UTF-8"
```

```
C:\wamp\www\github>git commit -m "UTF-8"
[master f17c5bb] UTF-8
1 file changed, 2 insertions(+), 2 deletions(-)
C:\wamp\www\github>git push
```

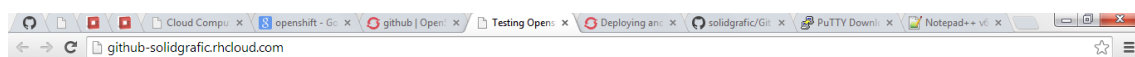
```
C:\Windows\system32\cmd.exe - git push

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)

Counting objects: 5, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 303 bytes | 0 bytes/s, done.
Total 3 (delta 2), reused 0 (delta 0)
remote: Stopping PHP 5.4 cartridge (Apache+mod_php)
remote: Waiting for stop to finish
remote: Waiting for stop to finish
remote: Stopping PHPMyAdmin cartridge
remote: Waiting for stop to finish
remote: Waiting for stop to finish
remote: Stopping MySQL 5.5 cartridge
remote: Building git ref 'master', commit f17c5bb
remote: Preparing build for deployment
remote: Deployment id is 2d2fbb5f
remote: Activating deployment
remote: Starting PHPMyAdmin cartridge
remote: Starting MySQL 5.5 cartridge
```

Cuando visitamos de Nuevo la web, observamos que los cambios se han realizado correctamente.

Nota: cuando se actualiza el repositorio se reinician todos los servicios del PAAS.



Trabajo final para el Curso de Cloud Computing

Sagrado Corazón de Jesús - La Salle

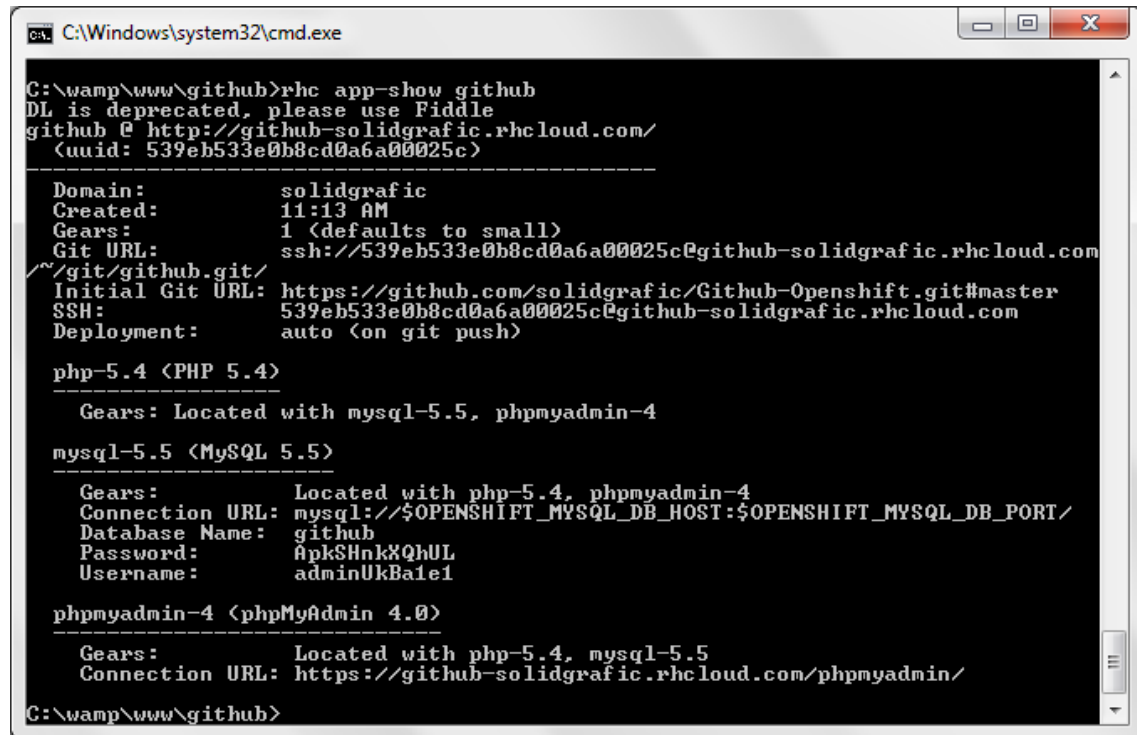
Carlos Serrano

6.1 Comando RHC

Existen comandos de rhc que nos permiten crear, reiniciar, destruir y configurar las app de OpenShift mediante la línea de comandos de nuestro pc.

A continuación, veremos un ejemplo muy sencillo.

Comando RHC para conocer detalles de la app



```
C:\Windows\system32\cmd.exe

C:\wamp\www\github>rhc app-show github
DL is deprecated, please use Fiddle
github @ http://github-solidgrafic.rhcloud.com/
(uuid: 539eb533e0b8cd0a6a00025c)

-----
Domain:          solidgrafic
Created:         11:13 AM
Gears:           1 <defaults to small>
Git URL:         ssh://539eb533e0b8cd0a6a00025c@github-solidgrafic.rhcloud.com
~/git/github.git/
Initial Git URL: https://github.com/solidgrafic/Github-Openshift.git#master
SSH:            539eb533e0b8cd0a6a00025c@github-solidgrafic.rhcloud.com
Deployment:      auto <on git push>

-----
php-5.4 <PHP 5.4>
-----
Gears: Located with mysql-5.5, phpmyadmin-4

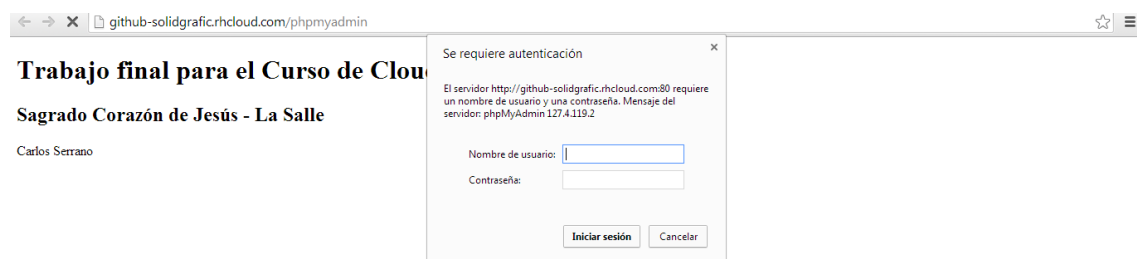
mysql-5.5 <MySQL 5.5>
-----
Gears: Located with php-5.4, phpmyadmin-4
Connection URL: mysql://$OPENSIFT_MYSQL_DB_HOST:$OPENSIFT_MYSQL_DB_PORT/
Database Name:  github
Password:       8pkSHnkXQhUL
Username:       adminUkBa1e1

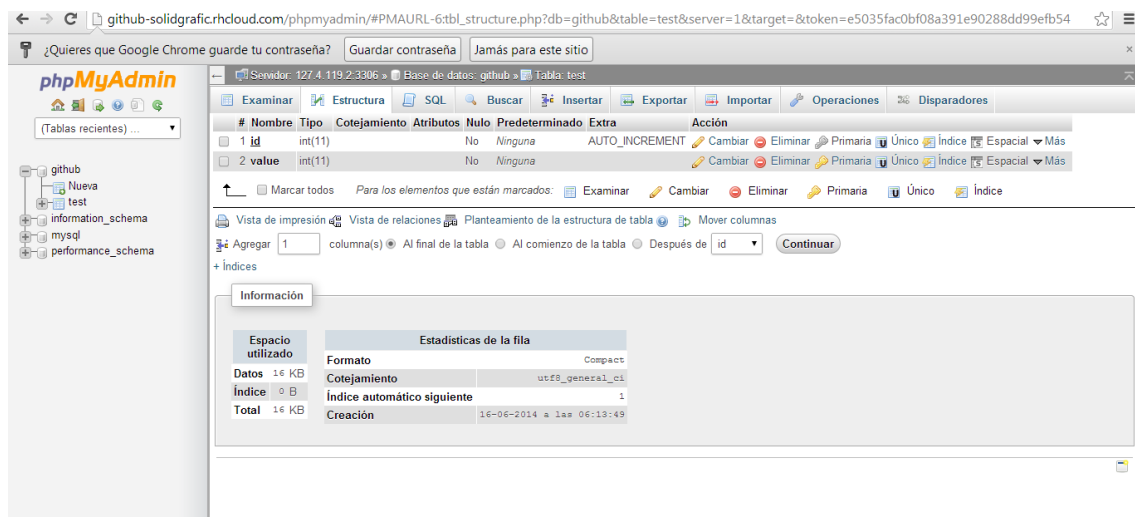
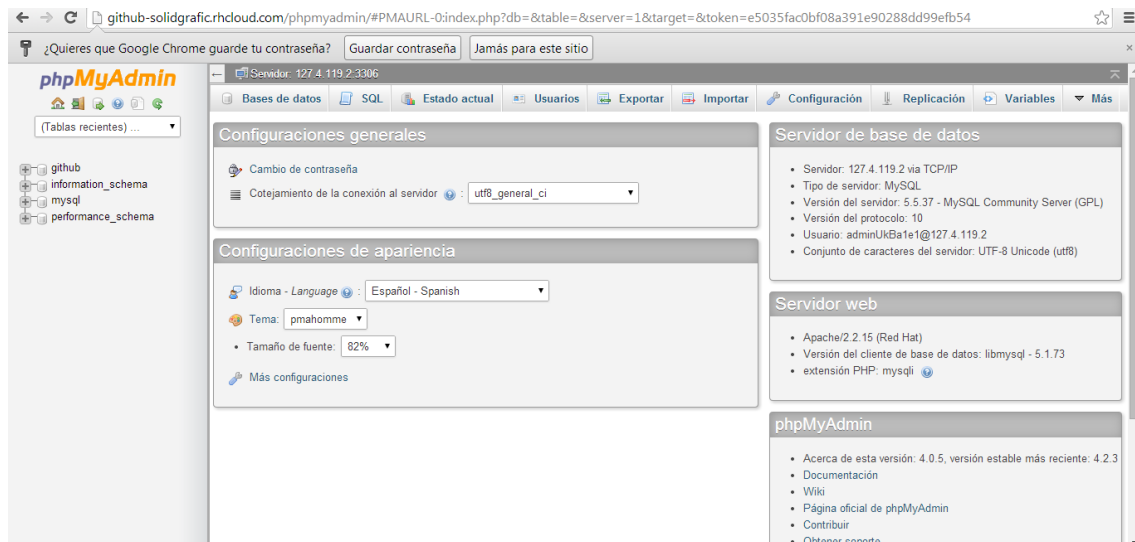
phpmyadmin-4 <phpMyAdmin 4.0>
-----
Gears: Located with php-5.4, mysql-5.5
Connection URL: https://github-solidgrafic.rhcloud.com/phpmyadmin/

C:\wamp\www\github>
```

7 Gestión de Base de datos

A continuación vamos a realizar la creación de una tabla en el servidor MySQL creado, mediante PHPMYAdmin.





Una vez creada la tabla, modifico el archivo server.php de mi app para que se conecte con el servidor. Dicho archivo queda:

```

1  <?php
2  // Conectando, seleccionando la base de datos
3  $link = mysql_connect('127.4.119.2:3306', 'adminUkBa1e1', 'ApkSHnkXQhUL')
4  or die('No se pudo conectar: ' . mysql_error());
5  echo 'Connected successfully';
6  mysql_select_db('github') or die('No se pudo seleccionar la base de datos');
7
8  // Realizar una consulta MySQL
9  $query = 'SELECT * FROM test';
10 $result = mysql_query($query) or die('Consulta fallida: ' . mysql_error());
11
12 // Imprimir los resultados en HTML
13 echo "<table>\n";
14 while ($line = mysql_fetch_array($result, MYSQL_ASSOC)) {
15     echo "\t<tr>\n";
16     foreach ($line as $col_value) {
17         echo "\t\t<td>$col_value</td>\n";
18     }
19     echo "\t</tr>\n";
20 }
21 echo "</table>\n";
22
23 // Liberar resultados
24 mysql_free_result($result);
25
26 // Cerrar la conexión
27 mysql_close($link);
28 ?>

```

Realizamos la sincronización.

```
C:\Windows\system32\cmd.exe
DL is deprecated, please use Fiddle
github @ http://github-solidgrafic.rhcloud.com/
(uuid: 539eb533e0b8cd0a6a00025c)

-----
Domain:      solidgrafic
Created:     11:13 AM
Gears:       1 <defaults to small>
Git URL:     ssh://539eb533e0b8cd0a6a00025c@github-solidgrafic.rhcloud.com
~/git/github.git/
Initial Git URL: https://github.com/solidgrafic/Github-Openshift.git#master
SSH:         539eb533e0b8cd0a6a00025c@github-solidgrafic.rhcloud.com
Deployment:   auto <on git push>

-----
php-5.4 <PHP 5.4>

Gears: Located with mysql-5.5, phpmyadmin-4

-----
mysql-5.5 <MySQL 5.5>

Gears: Located with php-5.4, phpmyadmin-4
Connection URL: mysql://$OPENSSHIFT_MYSQL_DB_HOST:$OPENSSHIFT_MYSQL_DB_PORT/
Database Name:  github
Password:       8pkSHnkXQhUL
Username:       adminUkBa1e1

-----
phpmyadmin-4 <phpMyAdmin 4.0>

Gears: Located with php-5.4, mysql-5.5
Connection URL: https://github-solidgrafic.rhcloud.com/phpmyadmin/

C:\wamp\www\github>git add server.php
C:\wamp\www\github>git commit -m "Actualización server.php con config"
```

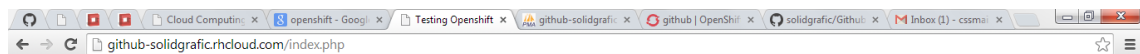
```
C:\Windows\system32\cmd.exe

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>

Counting objects: 5, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 371 bytes | 0 bytes/s, done.
Total 3 (delta 2), reused 0 (delta 0)
remote: Stopping PHP 5.4 cartridge (Apache+mod_php)
remote: Waiting for stop to finish
remote: Waiting for stop to finish
remote: Stopping PHPMyAdmin cartridge
remote: Waiting for stop to finish
remote: Waiting for stop to finish
remote: Stopping MySQL 5.5 cartridge
remote: Building git ref 'master', commit c378ee9
remote: Preparing build for deployment
remote: Deployment id is cc8001a7
remote: Activating deployment
remote: Starting PHPMyAdmin cartridge
remote: Starting MySQL 5.5 cartridge
remote: Starting PHP 5.4 cartridge (Apache+mod_php)
remote: Application directory "/" selected as DocumentRoot
remote:
remote: Git Post-Receive Result: success
remote: Activation status: success
remote: Deployment completed with status: success
To ssh://539eb533e0b8cd0a6a00025c@github-solidgrafic.rhcloud.com:~/git/github.git/
 f17c5bb..c378ee9  master -> master

C:\wamp\www\github>
```

Y finalmente, podemos observar nuestra app PHP trabajando con el servidor MySQL.



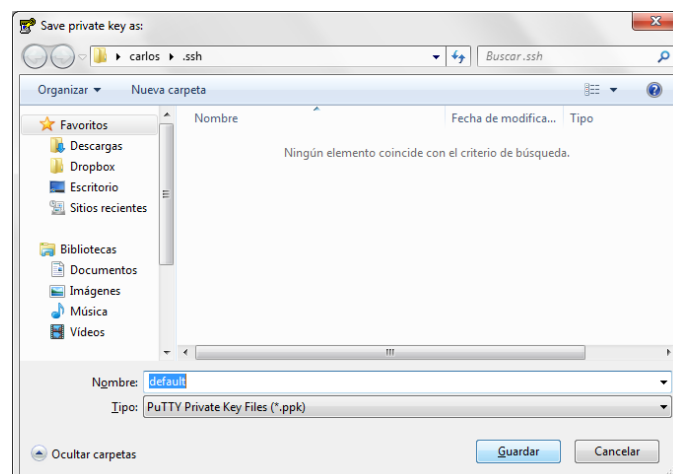
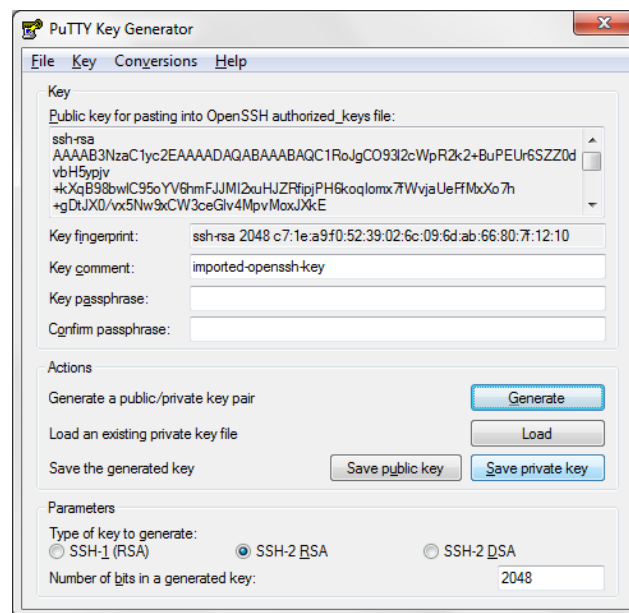
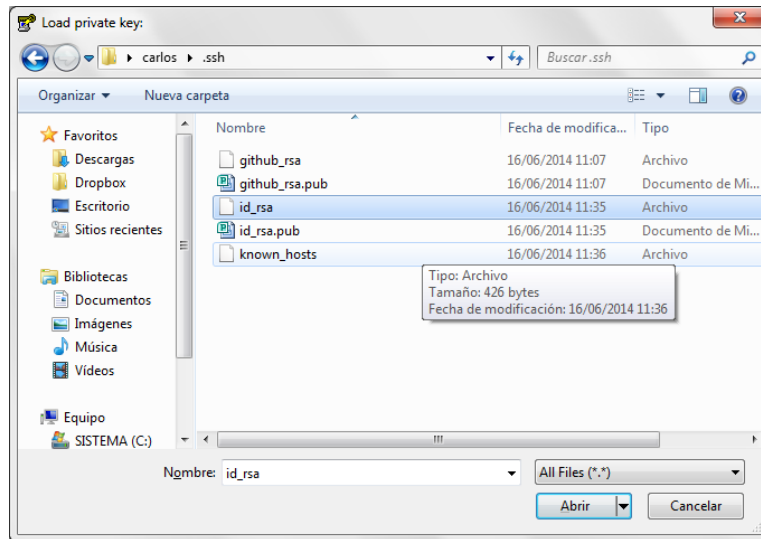
Trabajo final para el Curso de Cloud Computing

Sagrado Corazón de Jesús - La Salle

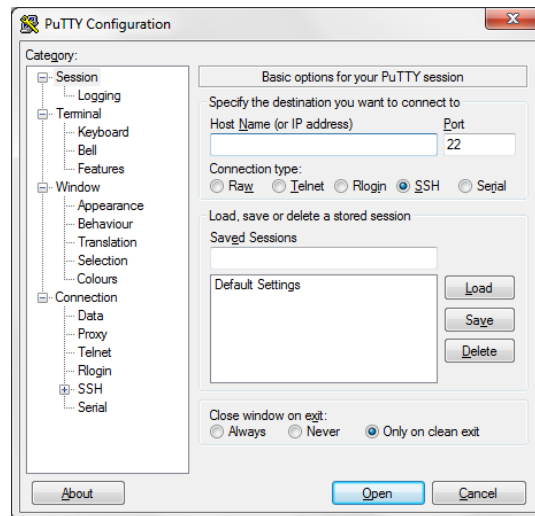
Conectando...
Connected successfully
1 1
2 2
3 3
Carlos Serrano

8 Putty para acceso SSH

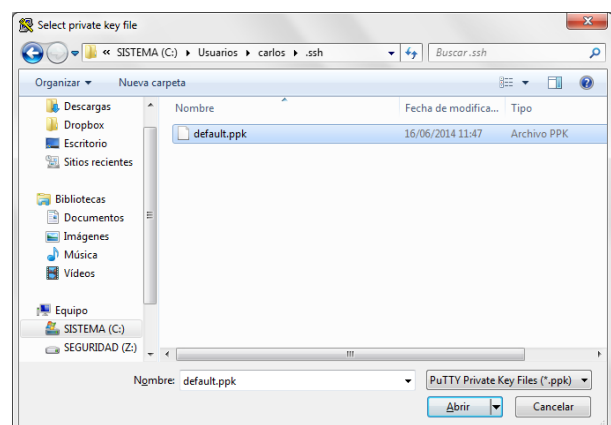
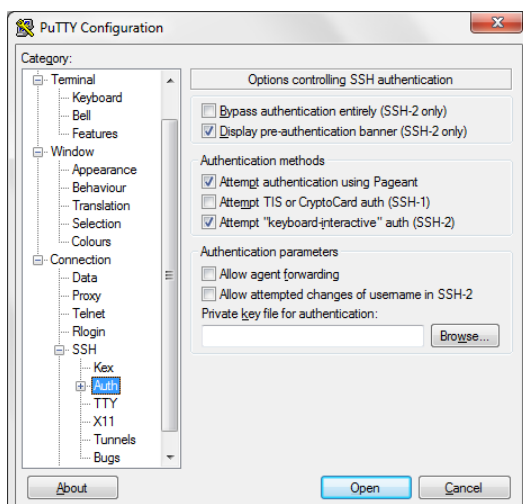
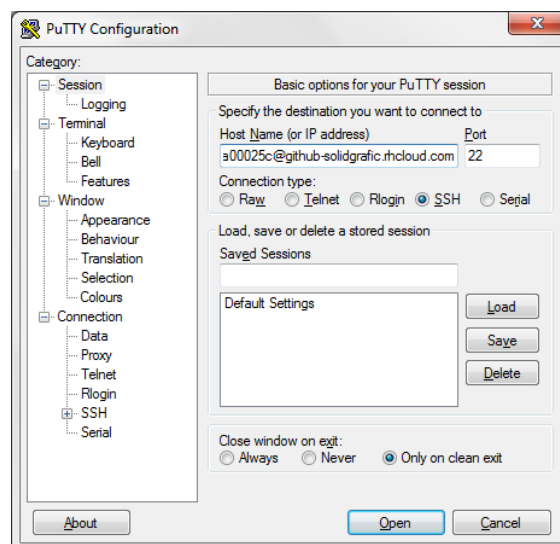
Instalamos putty y puttygen. Arrancamos éste último y en el menú Conversiones->importar clave, seleccionamos la que hemos creado con rhc.

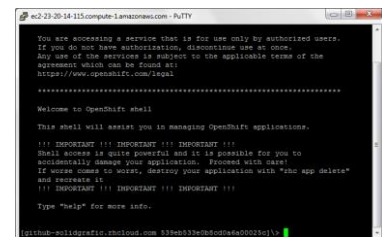
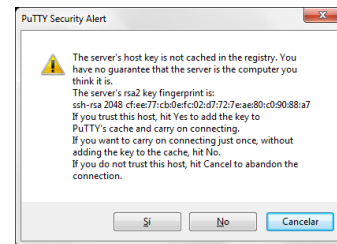
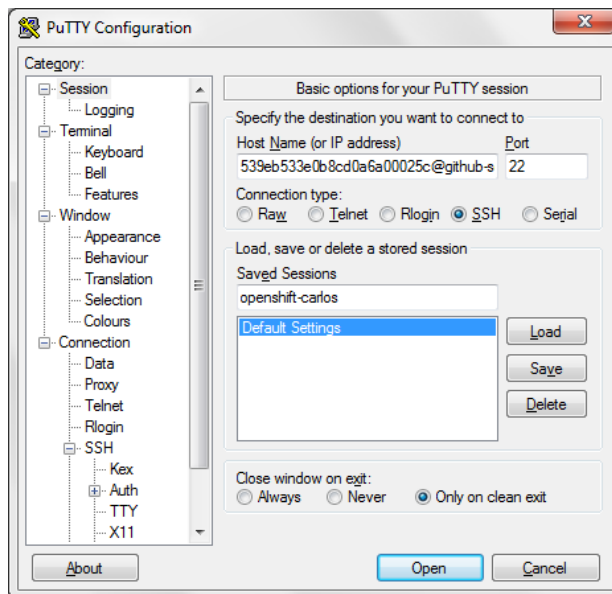


Una vez realizada la conversión a ppk y guardada en un directorio accesible, iniciamos Putty y configuramos los datos de conexión.



La URL la extraemos del apartado de sincronización git de este mismo documento.





9 Conclusiones

Como ya sabemos, el paradigma cloud computing ha llegado, y por lo visto para quedarse como mínimo hasta que surja algo más evolucionado. Las soluciones PaaS pueden ser un gran recurso a la hora de desarrollar nuestras aplicaciones. Son muy interesantes cuando hablamos de aplicaciones que tienen picos de actividad puntuales, donde las bondades del cloud computing se hacen necesarias irremediablemente.

No obstante, la solución propuesta por OpenShift, se me antoja una interesante plataforma para enseñar al alumnado a ponerse en contacto con soluciones Cloud-Computing. Es interesante para que conozcan el nuevo paradigma del almacenamiento de dichas aplicaciones y lo que aún resulta más interesante es la mezcla de tecnologías que requiere para su empleo. Sin duda, solo “echarlo a andar” supone un gran esfuerzo (si no lo hacemos con un tutorial y siguiente, siguiente...) que trae como recompensa la asimilación de conceptos complejos que en otro supuesto sería más difíciles de entender: infraestructura de clave pública y privada, ssh, control de versiones, alojamiento, cloud...

Sin embargo, veo un pequeño detalle que hace que para mí, OpenShift se convierta en una plataforma de pre-producción como he explicado en la guía. El hecho de que en cada sincronización todos los gears se reinicien hace, a simple vista, que tomemos a OpenShift como la plataforma de prueba final (o antes de la final, vista de cliente) en donde probamos nuestros códigos, una vez al día, a la semana... en función de nuestro proyecto, pero no lo veo práctico para un desarrollo ágil y constante.

10 Bibliografía

<http://openshift.github.io/>

<https://www.openshift.com>

<http://git-scm.com/>

<http://rubyinstaller.org/>

<https://github.com/>

<http://www.chiark.greenend.org.uk/~sgtatham/putty/>