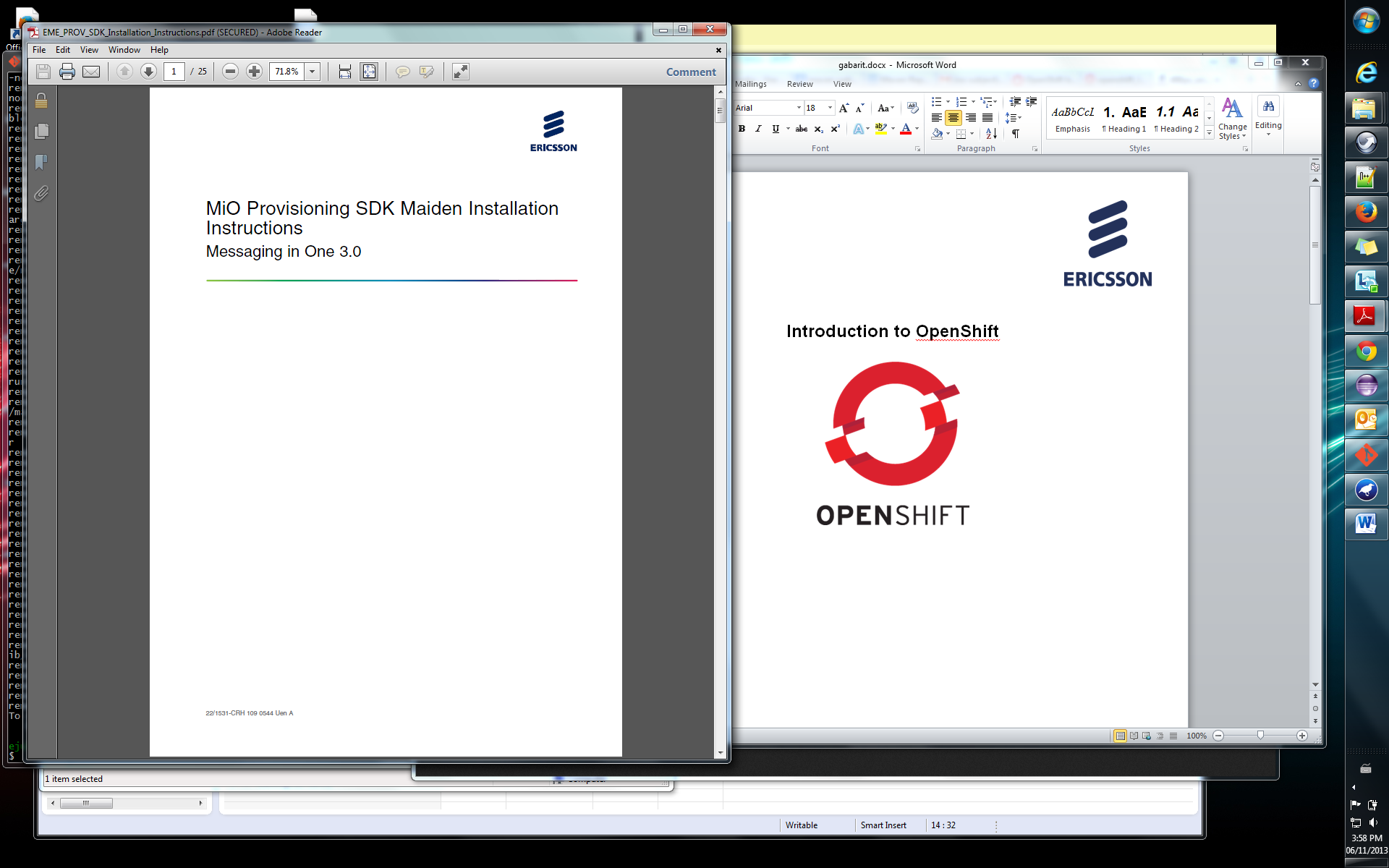
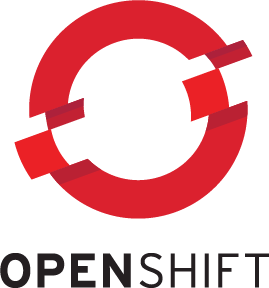


**Introduction to OpenShift**





**Table of Contents**

[1. Introduction 3](#_Toc371593794)

[1.1 What is OpenShift? 3](#_Toc371593795)

[1.2 Goal of this document 3](#_Toc371593796)

[2. Set up procedures 4](#_Toc371593797)

[2.1 Signing up 4](#_Toc371593798)

[2.2 Creating a Web Application 4](#_Toc371593799)

[2.3 Add your SSH Key 8](#_Toc371593800)

[2.4 Clone your OpenShift Repo 9](#_Toc371593801)

[2.5 Launching your app 10](#_Toc371593802)

[3. Adding Cartridges 11](#_Toc371593803)

[4. OpenShift Eclipse Plugin 13](#_Toc371593804)

[5. Maven dependencies 13](#_Toc371593805)

# Introduction

## What is OpenShift?

OpenShift is a cloud application development and hosting services developed by Red Hat. OpenShift automates the provisioning, management and even scaling of the projects to allow developers to deploy their applications on the cloud. OpenShift uses git repositories to help with the revision control when developing applications. Every time you push code to your git repo, OpenShift will recompile and launch your application.

To learn more about the interworking of OpenShift you can visit: <https://www.openshift.com/products/online>

## Goal of this document

As a criteria of McGill codeJam 2013, every team or contestants must use an openShift to publish their code for the judges to see. This documented is provided to guide you thru the steps of setting up an account and launching a web application with OpenShift.

# Set up procedures

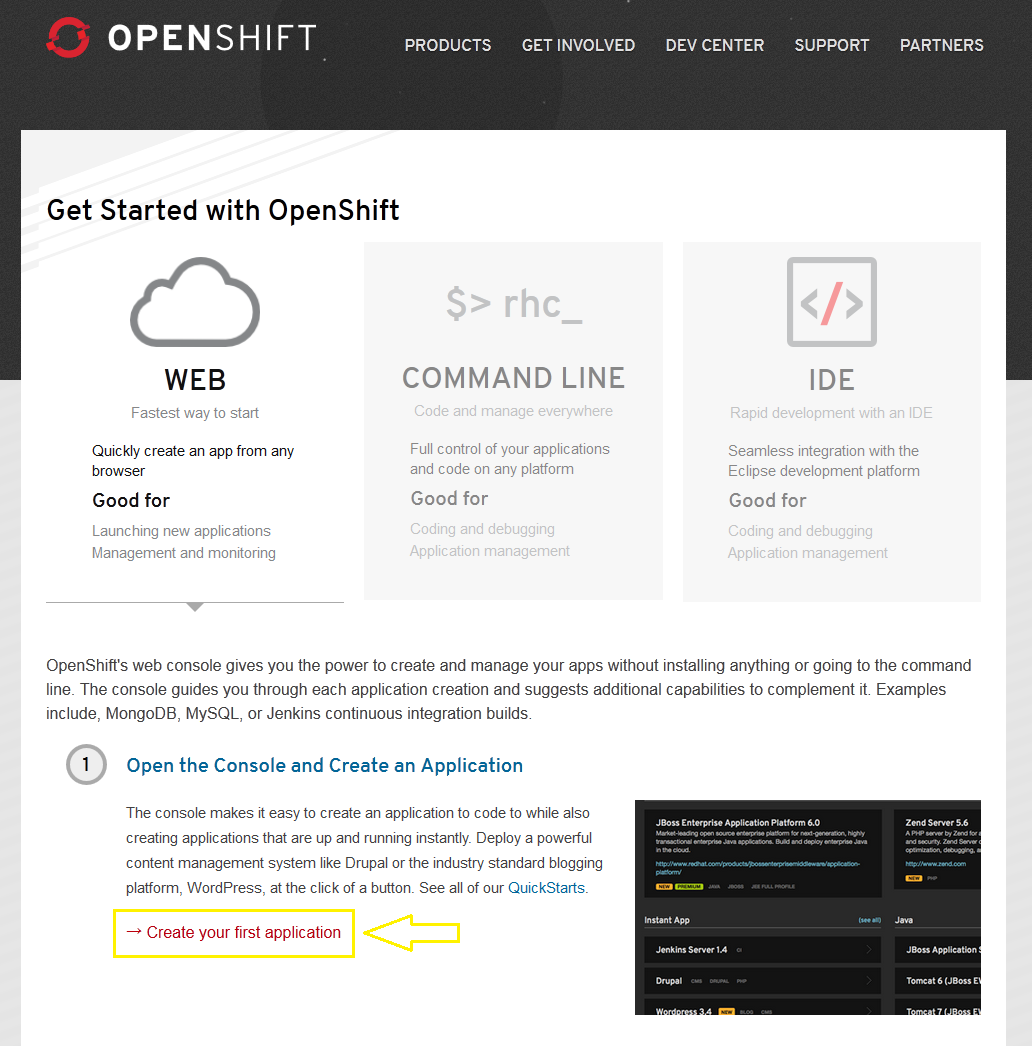
## Signing up

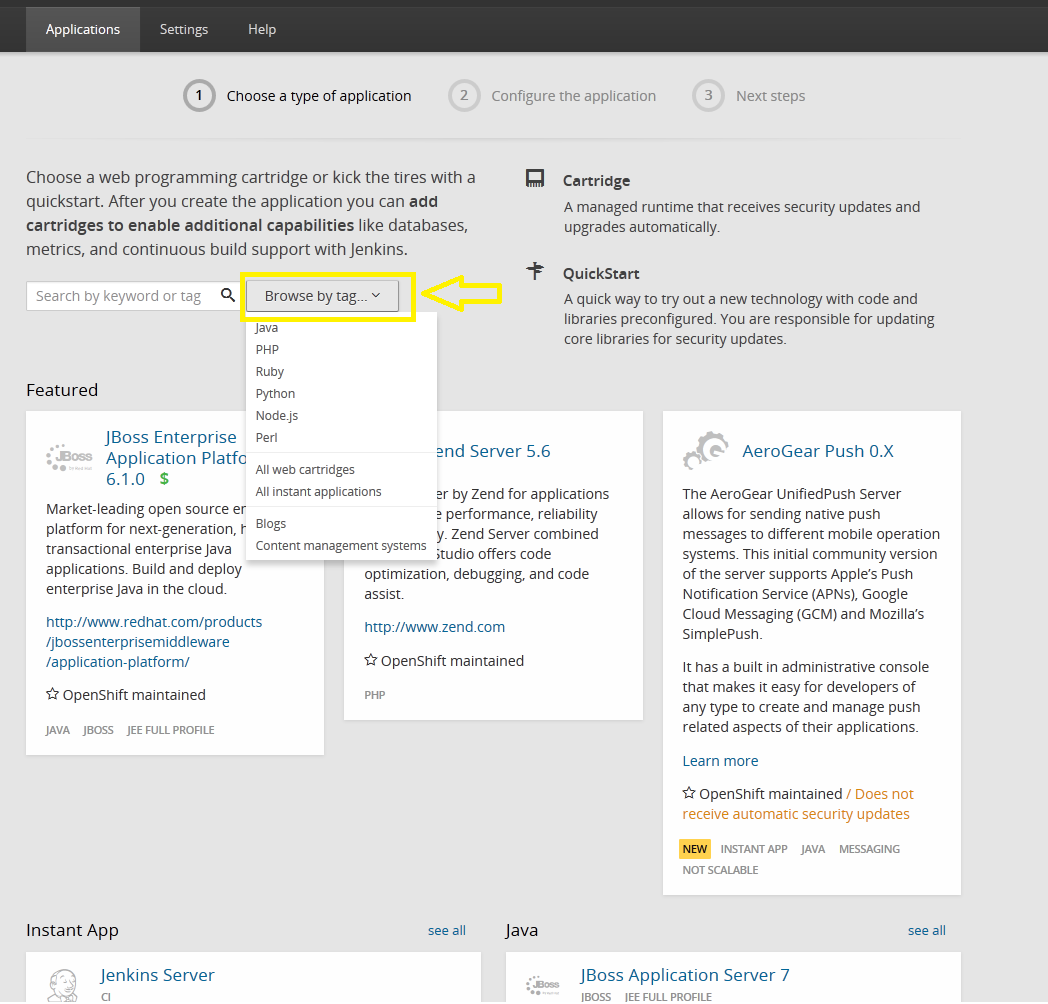
The first step to start using OpenShift is to create an account. Go to <https://www.openshift.com/> and click the “Sign Up for free” under Online: Public PaaS



## Creating a Web Application

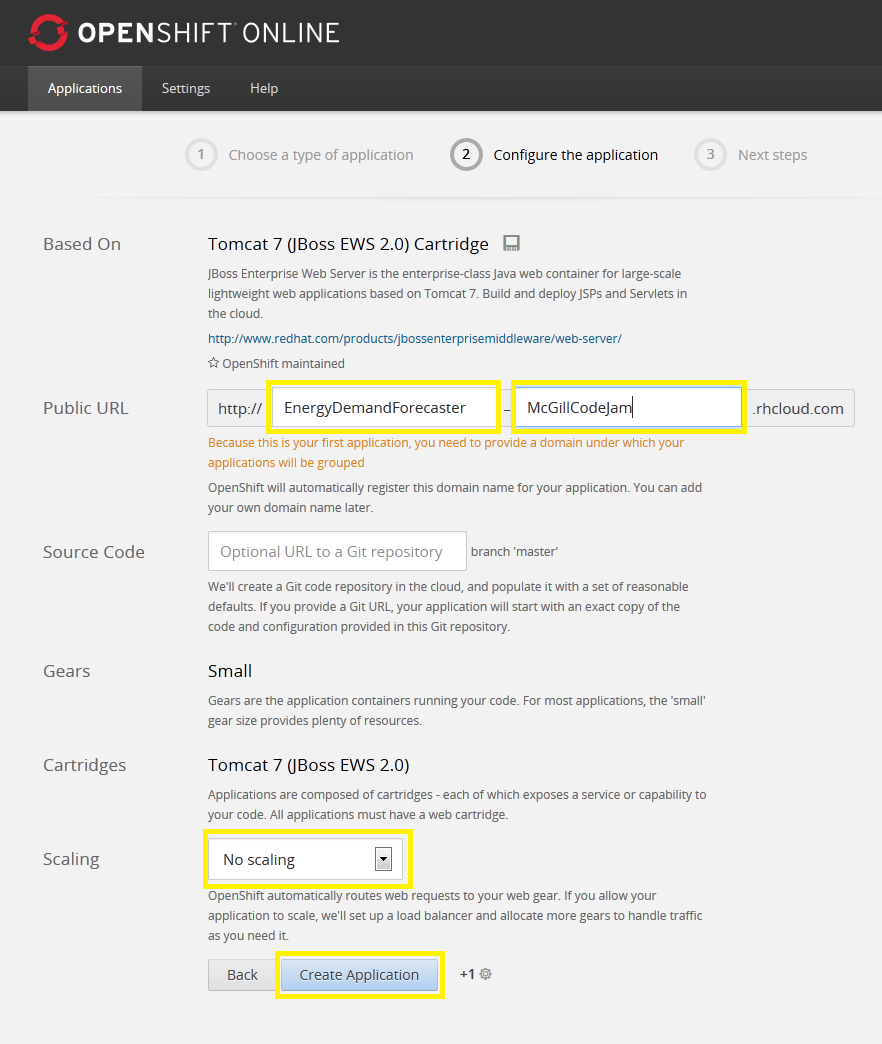
After entering your credentials, you will receive an email validation to activate your account. Once this is done and you are able to log into OpenShift you will be redirected to the getting Started Page. Once you’re here click on the “Create your First Application” button

After you click the link, there will be a list of things to do before you can start coding. You can read this if you wish, or else just click on “Create your first application now”



You’ll see that if you press on Browse by tag you’ll see the supported languages on OpenShift. By clicking on a specific language tag you’ll be able to see all the Web Server choices you have for that language. Once you’ve found the Web Server script language you like, click on it and you will be redirected to a new page

\*For the purpose of this demo, I’ll be creating a Tomcat 7 webapp running with JBoss EWS 2.0.



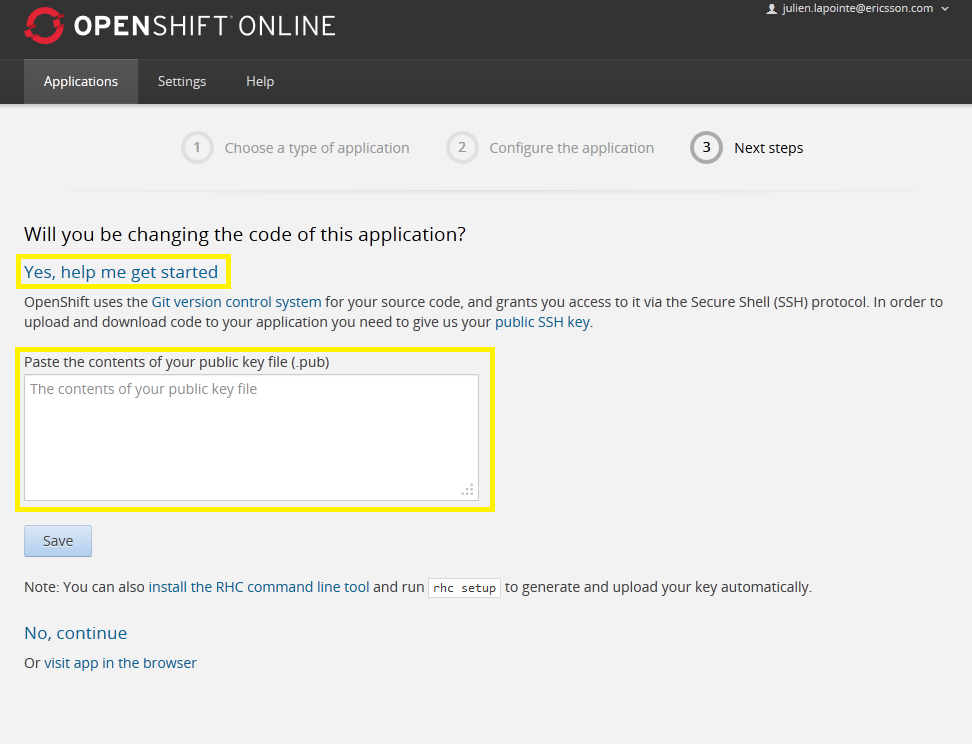
On this page you will see different options related to your project. First you have to set the URL of web project. The first box of the Public URL is for the name of your project. The second one is to specify the domain of your application; you can put whatever name you want as the domain isn’t relevant for CodeJam.

You can add source code to an OpenShift project; however for the competition we’ll leave this field empty.

The last option you’ll see is scaling, choose no scaling and click the create application button.

## Add your SSH Key

Once the loading is done, you will be directed to a page asking if you will be modifying the source code. Click “Yes, Help me get started”



You will then be asked to paste your SSH public Key used for git. You can usually find the .pub file that contains your key under. C:\Users\USERACCOUNT\.ssh

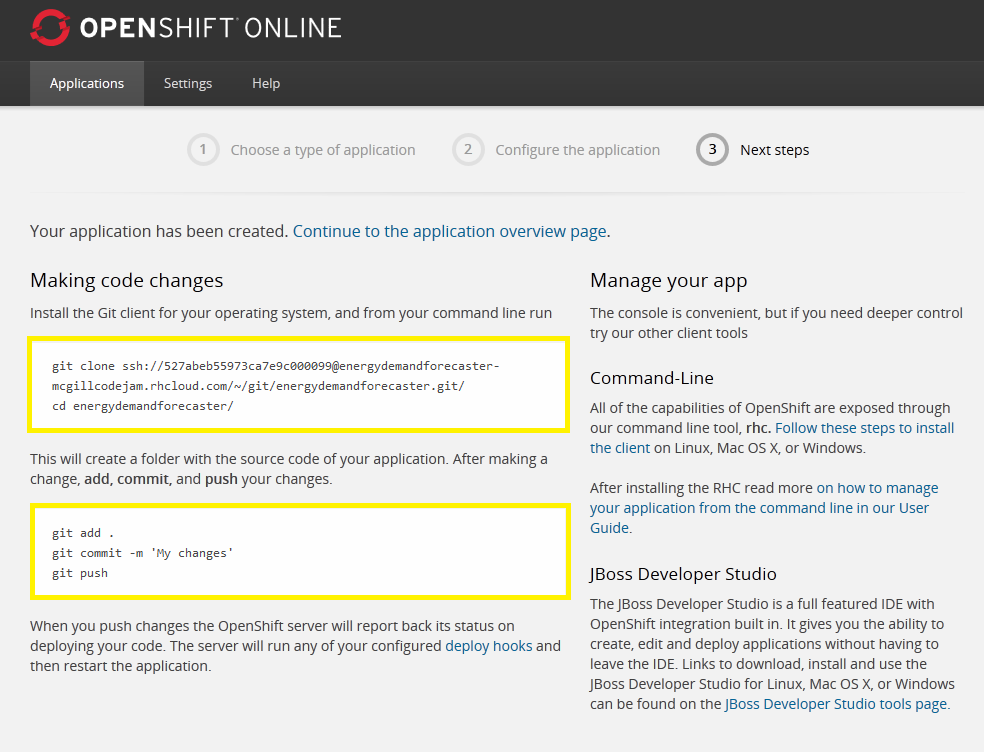
If you don’t have a key, you may want to look into the procedures to generate one here:

<https://help.github.com/articles/generating-ssh-keys>

Once you have your public key, paste it into the box and press save

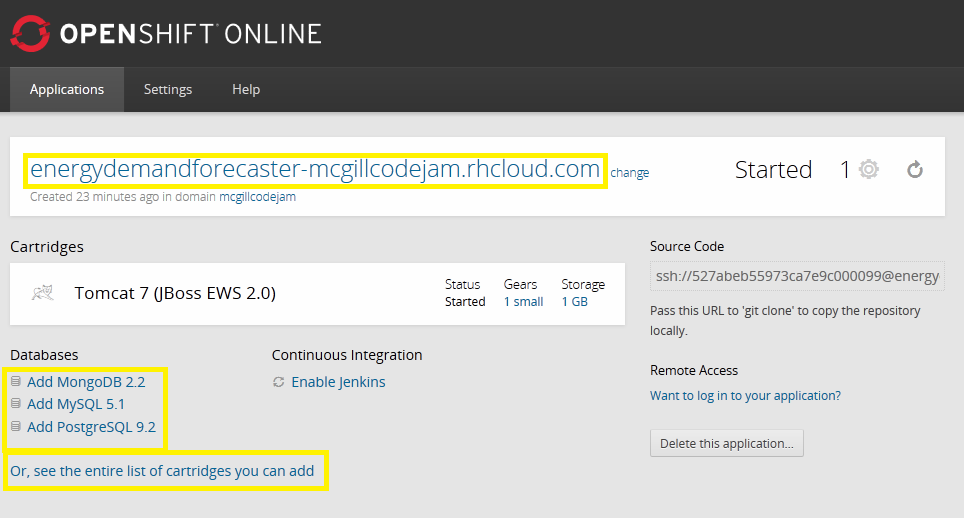
\*Your SSH key usually starts with ssh-rsa and ends with your USERNAME@ComputerName

## Clone your OpenShift Repo



On the page you’ll be redirected to, you can see the link to clone the OpenShift Repo so you can start Coding. Under this are the basic git commands you have to do to publish your code to the OpenShift cloud.

## Launching your app



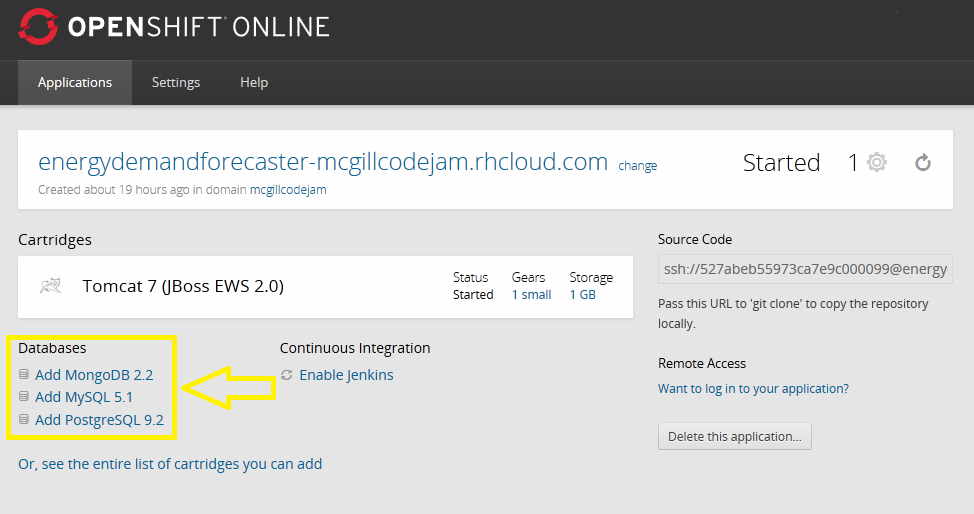
Once you’ve completed the application creation, you’ll be redirected to page containing your project information. Here you’ll find the link to your application on the top, the link to clone your git repo, and options to add cartridges (more information in section 3).

You can start your application by clicking on the link to your application (in this case energydemandforecaster-mcgillcodejam.rhcloud.com) you’ll be redirected to your application URL and you’ll see a default OpenShift html page being displayed. If you cloned the OpenShift repo, the page being displayed is index.html under YOUR\_PROJECT\src\main\webapp.

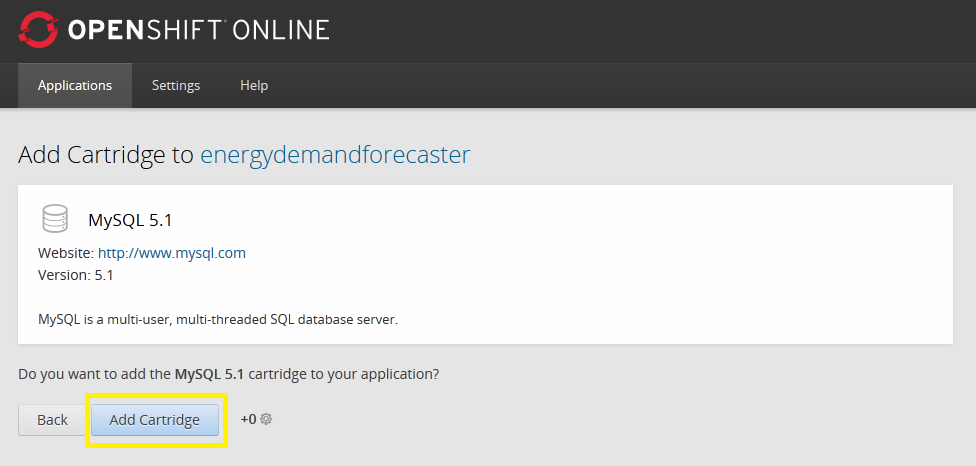
Happy coding ☺

# Adding Cartridges

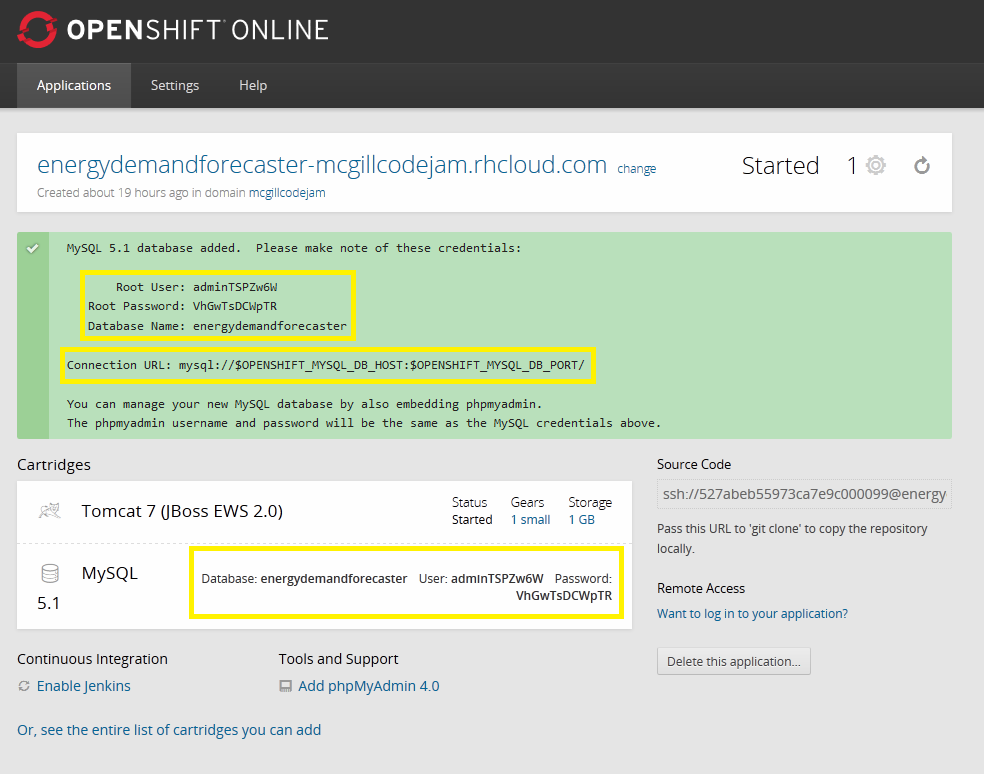
OpenShift uses cartridges to easily add databases or other tools to your project. Adding a database to your project is very simple using the cartridge system. On your application page click on the databse you want to use under the “Databases” section.



There are other useful cartriges like PHPMyAdmin if you click on the “see the entire list of cartridges” link. But for now we’ll just create a database. Once you clicked on the type of database you want to you you’ll be redirected to a new page.



Simply click on the Add “Cartridge” button and the database will be added to your project! Once this is done, you’ll be able to see the information related to your database.



After the database was created we can see the database’s Root User, Root Password, and Database Name. This information is used to connect your web application to the database. You don’t need to write this information down since you’ll be able to get it anytime on your OpenShift application page right next to the “MySQL” cartridge. You may be thinking that the Connection URL of your database seems odd. This is because OpenShift runs on a remote linux server and provides your application with many useful environment variables.

$OPENSHIFT\_MYSQL\_DB\_HOST would equal to the IP to connect to your database

$OPENSHIFT\_MYSQL\_DB\_USERNAME would be equal to your database Root User (or adminTSPZw6w in this example)

An example to get these values in java is System.getenv(“Variable\_Name“)

To get a list of all the environment variables you could use click here:

<https://www.openshift.com/page/openshift-environment-variables>

# OpenShift Eclipse Plugin

If you’ll be coding in eclipse there is a useful eclipse plugin you could use to push code straight from Eclipse. It also includes a multitude of other tools such as creating an OpenShift application from Eclipse, importing one, integration with git repos and more. This plugin is very useful if you take a bit of time to get to know what it does. You can find OpenShift Eclipse Plugin installation procedures here:

<https://www.openshift.com/blogs/getting-started-with-eclipse-paas-integration>

# Maven dependencies

If you are unfamiliar with Maven, it’s a software management tool created by apache and is used to build applications. The interesting thing with Maven is that it downloads any missing project dependencies listed in your Maven POM (Project Object Model). This means that if you add library dependencies to your POM file, you don’t have to manually add libraries to your project!

In the top level of your git repo, you’ll find a POM.xml file, if you open it you’ll see that OpenShift already added some project dependencies for you.

Adding a dependency to your project is very simple: Just add the dependency between the

**<dependencies> </dependencies>** tags for example the highlighted lines are the new dependency:

<dependencies>

<dependency>

<groupId>org.postgresql</groupId>

<artifactId>postgresql</artifactId>

<version>9.2-1003-jdbc4</version>

</dependency>

<dependency>

<groupId>org.apache.commons</groupId>

<artifactId>commons-math3</artifactId>

<version>3.0</version>

</dependency>

</dependencies>

Maven supports a multitude of different libraries; you can find all the supported libraries and the tags to use here <http://mvnrepository.com/>