

# Release Pipeline for Microservices

In this interactive workshop, you will learn how to achieve faster release cycles for your microservices using CI/CD approach. You will create a release pipeline that is automated and highly reliable, so that the risks of deploying updates are minimized. You will utilize open source tools along with declarative pipeline and shared libraries provided by Jenkins.

This guide provides you with the steps that will be shown during the workshop. Make sure to checkout the resources section at the end of this document. For updated version of this pdf, go to :

<https://github.com/GHCdemo/Demo-WorkSheets/blob/master/Worksheet.pdf>

## Speakers:

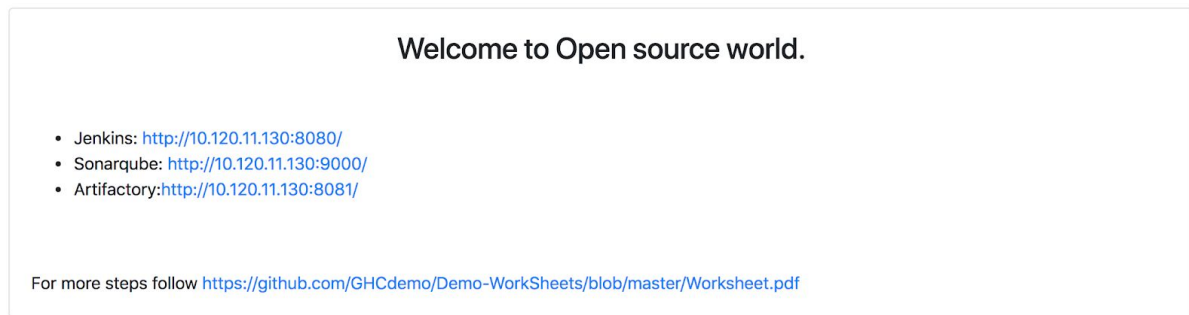
Srujana Bobba | @srujanabobba | <http://github.com/bobsru>  
Jyotsna Chatradhi | @jyotsnac | <http://github.com/JyotsnaC>

**Note:** If you need your environment for a longer time (maximum a day), please email us at [releasepipeline@gmail.com](mailto:releasepipeline@gmail.com) with your environment ip address. By default, the environments will be deleted an hour after the workshop.

## Step 1: Setup a Workspace

For this entire workshop, each one of you is going to get your own environment with the applications already installed on it. Follow the below steps to get an environment.

- Go to <https://assignworkspace.herokuapp.com/>
- Enter your email address and click on 'Create Workspace' button .You should see links for the installed applications similar to the ones shown below.



## Step 2: Fork github repositories

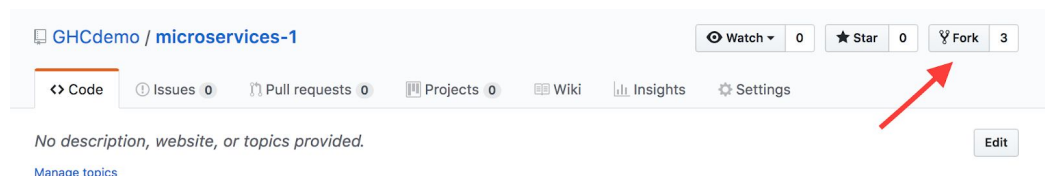
On our [github account](#), we added 3 microservices that we will be using in this workshop. Follow the below steps to fork these repositories into your github account.

- Open <https://github.com> and log into your account. If you do not have one, [create a new account](#).
- Click on each link below and fork the repositories into your github account. See the image for the fork button.

<https://github.com/GHCdemo/microservices-1>

<https://github.com/GHCdemo/microservices-2>

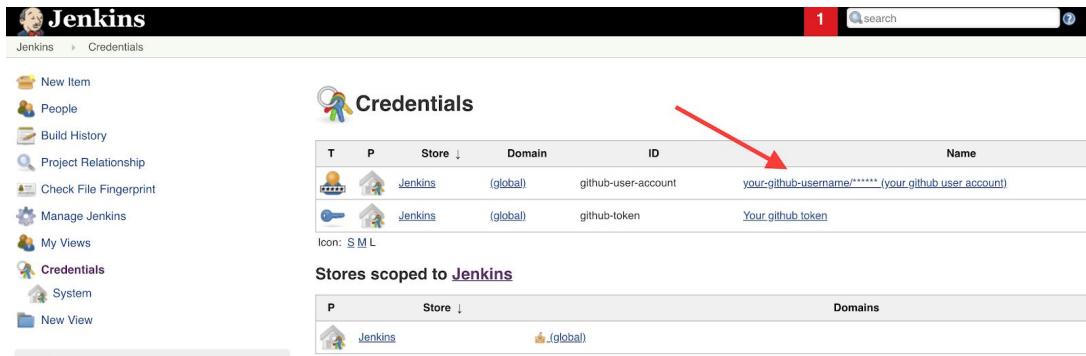
<https://github.com/GHCdemo/microservices-3>



## Step 3: Update your Github credentials in Jenkins

- Go to your workspace page you got after doing Step 1
- Click on the Jenkins URL
- Login with the below credentials:  
Username: admin  
Password: admin

- On the left menu, click on 'Credentials' link. Dummy credentials are already added for this setup. Click on 'your-github-username...' link.



- Click on update from the left side menu
- Update your github username (not email) in the username field (It is case sensitive. So make sure you are entering as it is)
- Enter your github password in the 'Password' field
- Click on save

#### Step 4: Scan your Github account

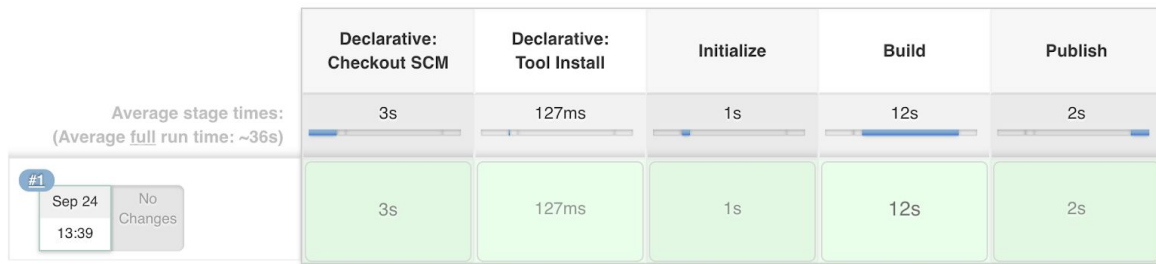
- Go to Jenkins home, click on 'github-org'.
- Click on 'Configure' from the left menu.
- Give your Github username in owner field (It is case sensitive).



- Click on Save at the end of the page.
- In the navigation bar, you should see 'github-org' if your Github profile has empty name. If not, you should see your Github name. Click on the link.

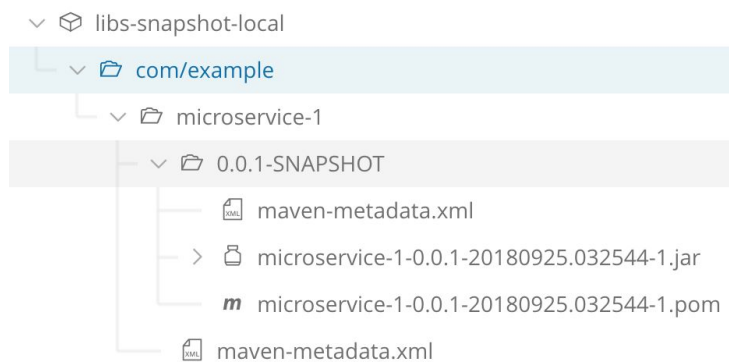
You should see all your repositories which has Jenkinsfile in it at the root level. Since you forked all 3 microservices repositories, you should now see microservices-1 repository. A build for 'master' branch should trigger automatically by now. (To check the build, click on 'microservices-1' and then 'master' links. You should see the pipeline view)

## Stage View



During 'Publish' stage in the pipeline, the artifacts will be published to artifactory. You can see the published artifacts at this location:

<http://<your-environment-ip>:8081/artifactory/webapp/#/artifacts/browse/tree/General/libs-snapshot-local/com/example/microservice-1/0.0.1-SNAPSHOT>



## Step 5: Where is the configuration/Jenkinsfile given for the microservices-1 ?

A file called 'Jenkinsfile' is already created in microservices-1 repository. Go check it out and see how it looks here:

<https://github.com/GHCdemo/microservices-1/blob/master/Jenkinsfile>

## Step 6: Shared Pipeline

Let's use a shared pipeline that is created in this repository

<https://github.com/GHCdemo/shared-pipeline> and update our microservices-2 repository. Follow below steps:

- Go to your microservices-2 repository in your github account
- Create a new file called 'Jenkinsfile' (You can use github UI)
- Copy the contents from the below link  
<https://github.com/GHCdemo/Demo-WorkSheets/blob/master/Jenkinsfile-final>
- Commit the newly created Jenkins file
- Go back to Jenkins page and click on Jenkins home.

- Click on 'github-org'.
- On the left side of the organization, click on 'Scan Organization Now' link.
- Refresh your page

You should now see microservices-2 repo in the list. A build should be triggered and the pipeline looks similar to microservice-1 except the Jenkinsfile is using the shared library.

### Step 7: Add a new stage to the shared pipeline

We are going to do Add a new stage called 'Sonar Analysis' to the shared pipeline, after that run your builds again and see if the new stage is picked up. To do that follow the below steps:

- Click on Jenkins home
- Click on 'github-org'
- Click on 'microservices-2'
- Click on 'master'
- Click 'Build Now' from the left side menu

You should see that a new stage called 'Sonar Analysis' is added to the Jenkins pipeline as below

#### Stage View



#### SonarQube Quality Gate

microservices-1 **OK**  
server-side processing: **Success**

You can access sonar analysis from this link:  
<http://<your-environment-ip>:9000/projects>

