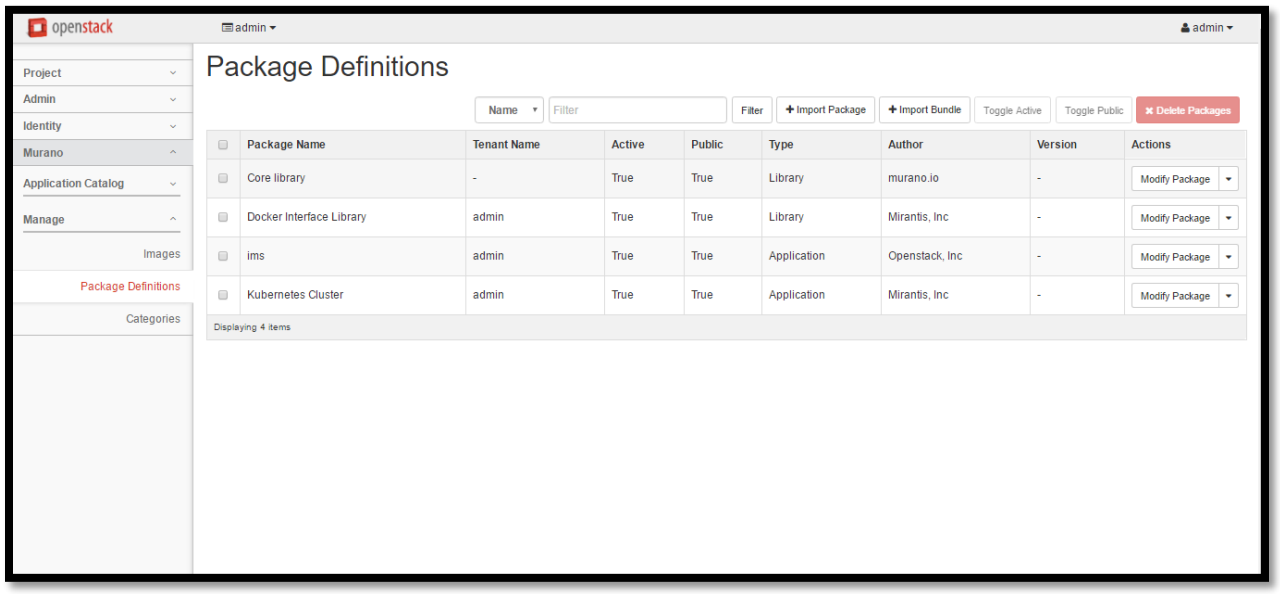
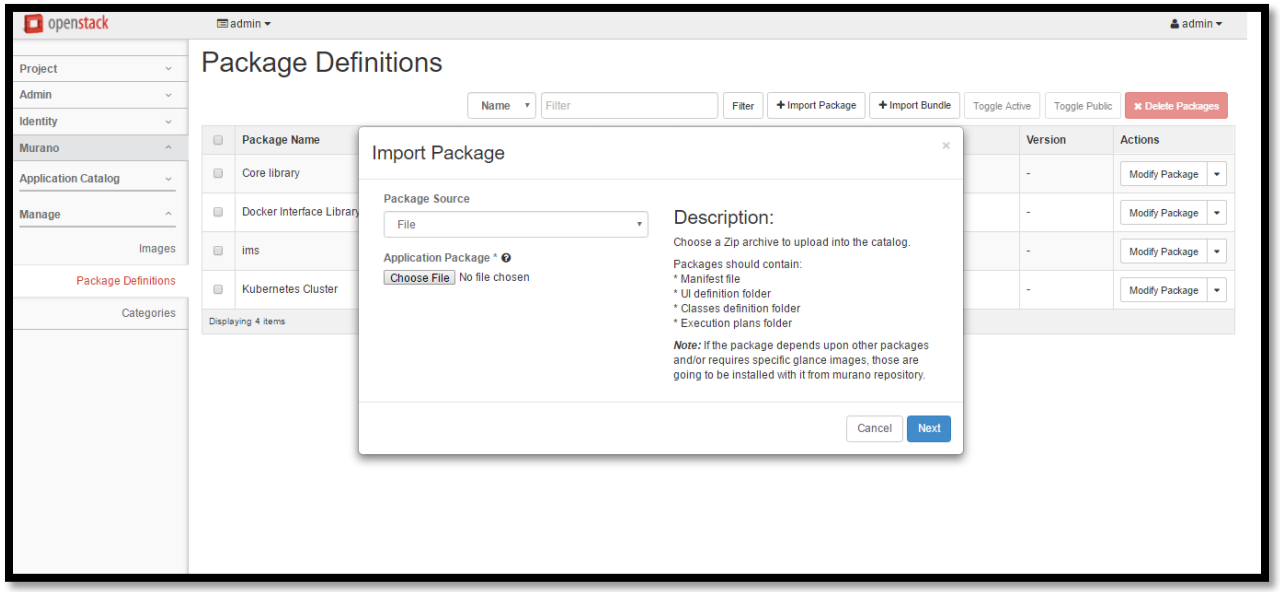
# Procedure

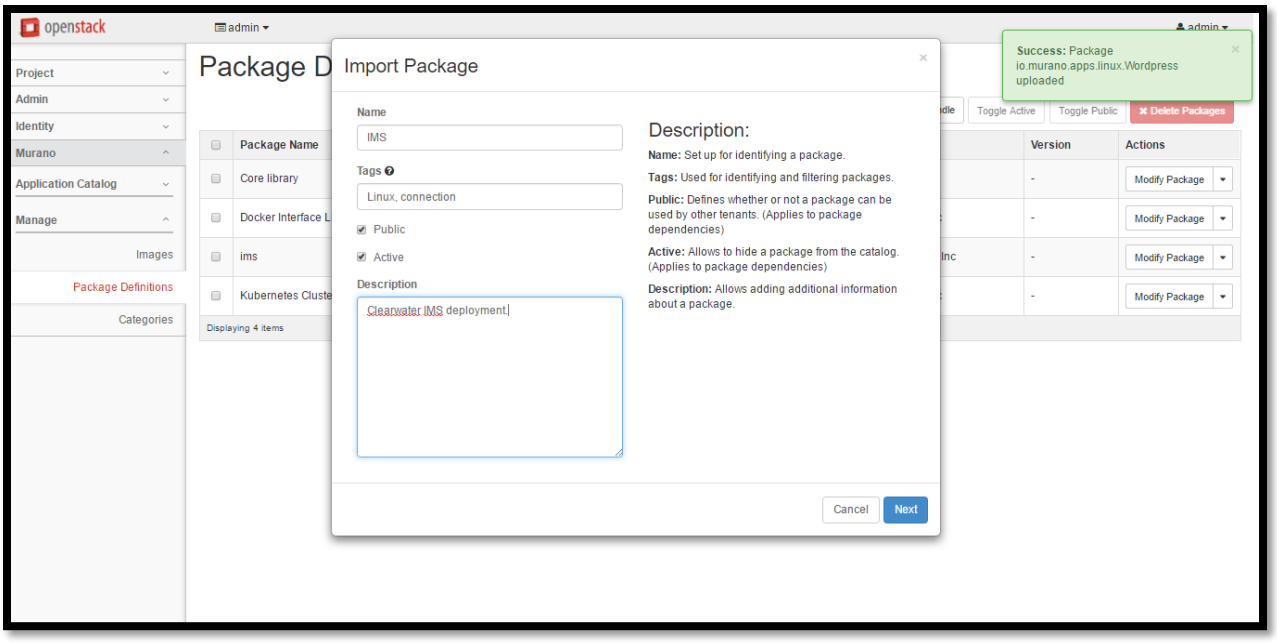
**Step 1:** The figure below shows the Murano component GUI in the Horizon dashboard. The first step is to import the package in to the Murano online catalogue. Click the import package button at the top.



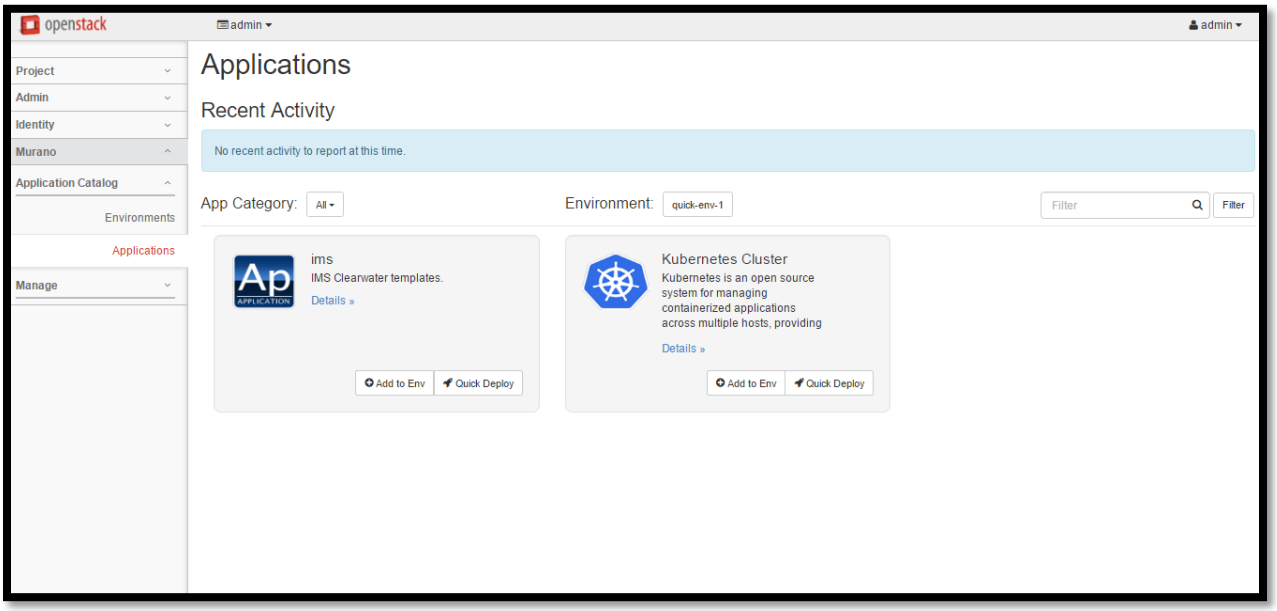

**Step 2:** Choose the package folder from your laptop to upload to Murano and click next.



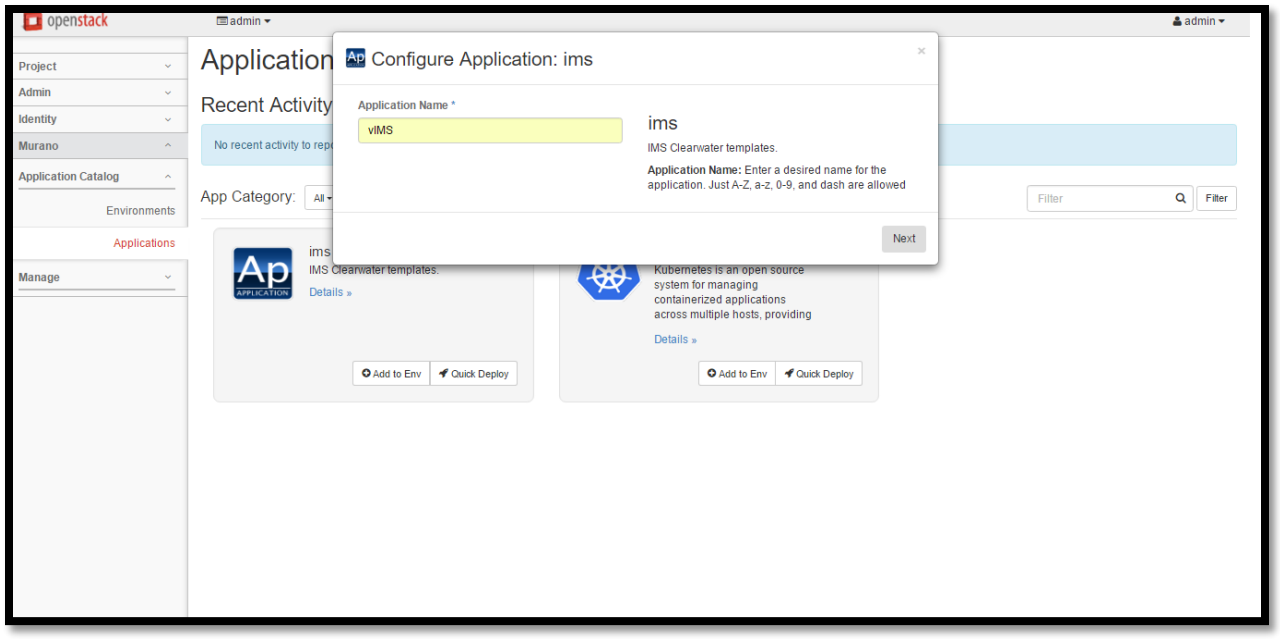

**Step 3:** The page below displays the description of the package as written in the manifest file. Leave it as it is and click next.



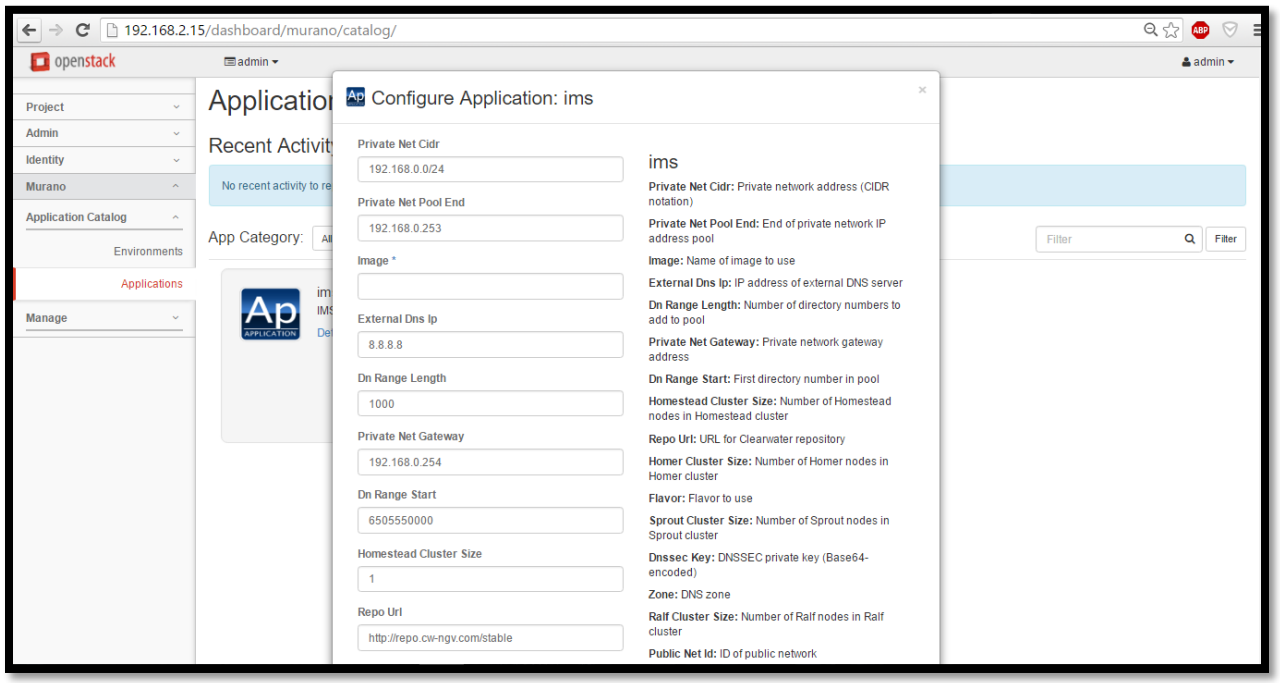

**Step 4:** Click the Application Catalog under the Murano tab on the left, then click applications. You will see your newly created application displayed on the page. Click 'Quick Deploy' on the application.



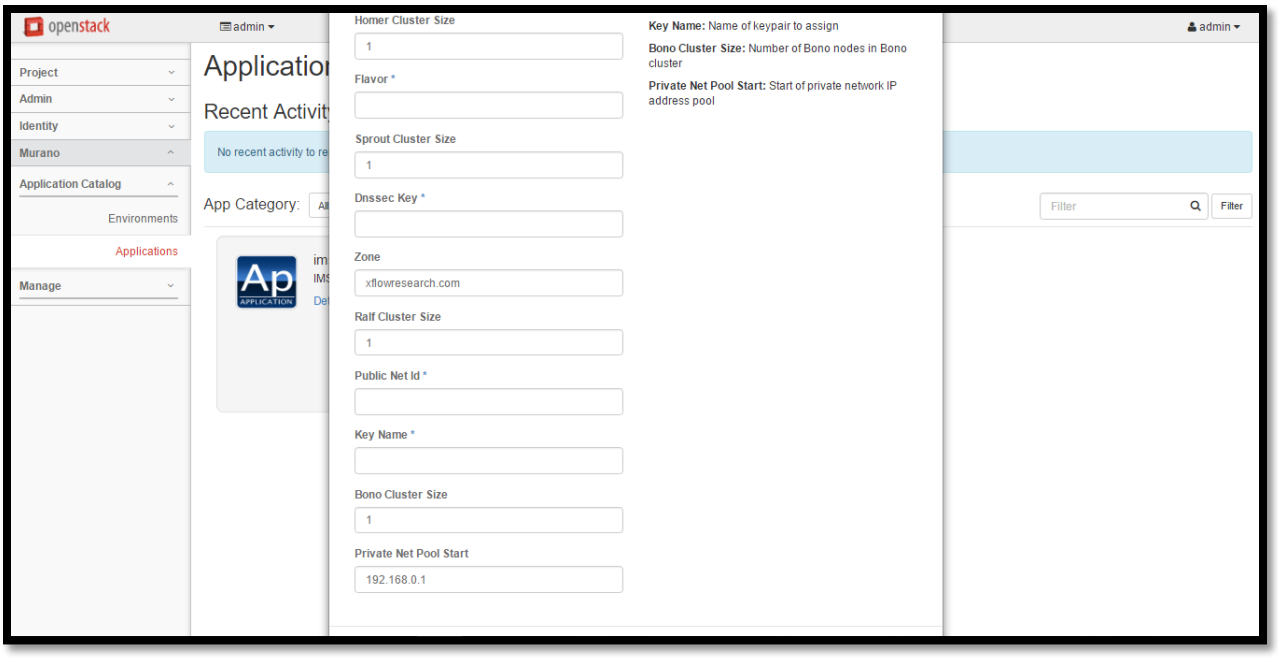

**Step 5:** The new window will open where you need to specify the application name, then click next.



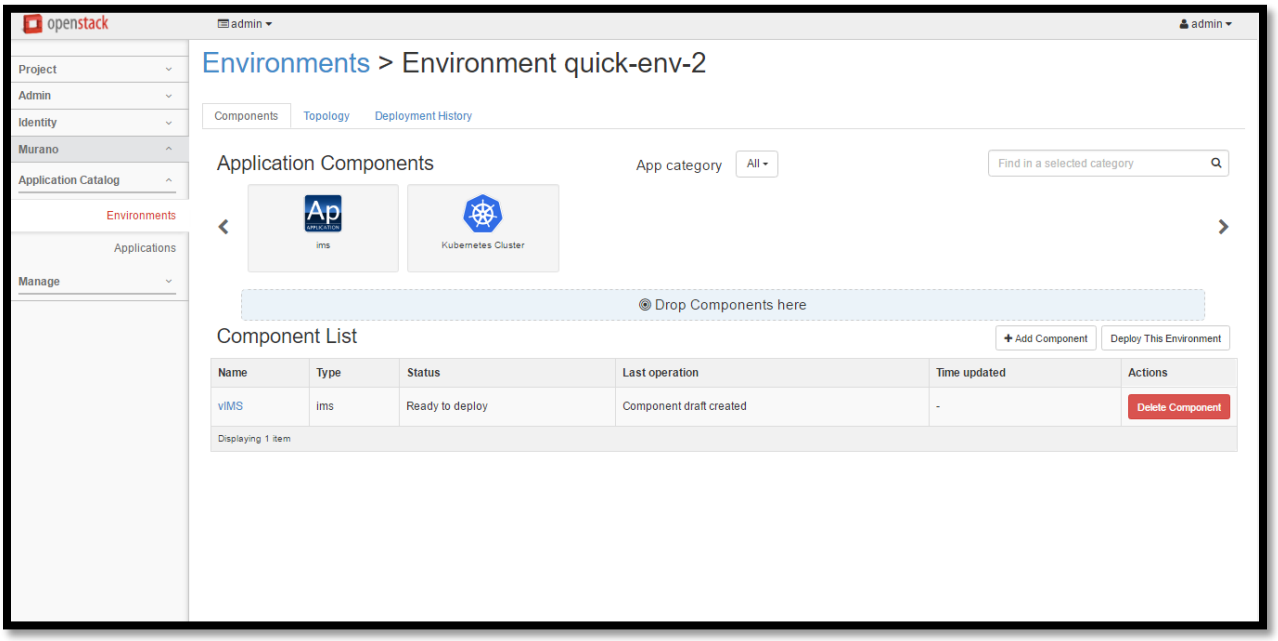

**Step 6:** Next you have to provide the image name. We saved the downloaded image with the name 'IMS', so write IMS.



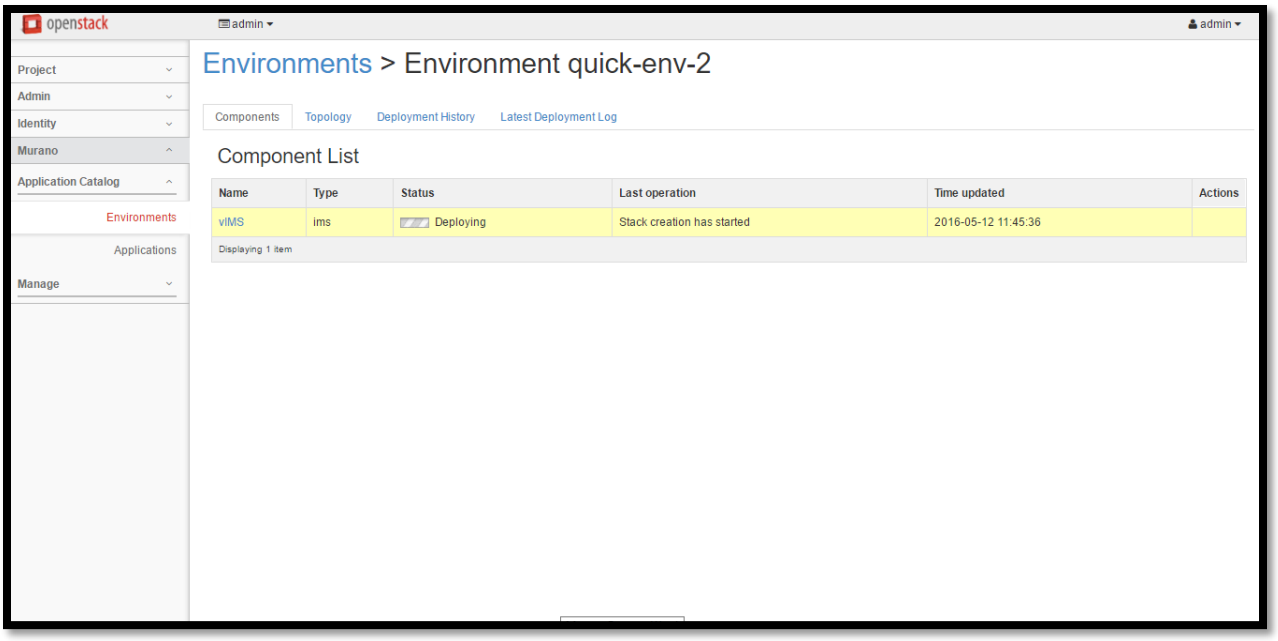

**Step 7:** Write the flavor as 'm1.medium'. Copy and paste the ID of your public network, and write 'secure' as the key name (it is the name of the key pair we created earlier). Click create.



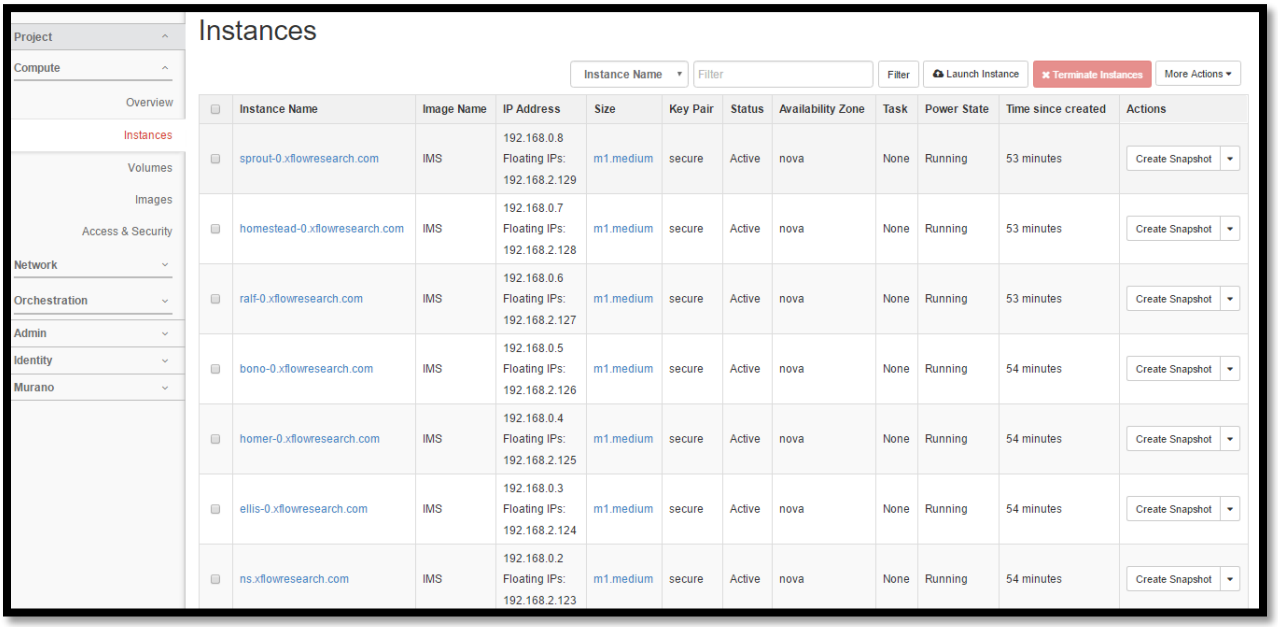

**Step 8:** The environment is created with the application available in the component list. Click 'Deploy The Environment'.




**Step 9:** The page will show that your application deployment has started and the corresponding stack creation is taking place. After a minutes the deployment will be finished.




**Step 10:** All the vIMS nodes can now be seen deployed in the instance tab.




## Auto-Scaling

Auto-scaling is a typical use case for applications that could be scaled horizontally. The Murano application author can implement application auto scaling in two different ways:

- The first approach supported by Murano is to use Heat autoscaling resources, which works perfectly if you can scale your application with the addition of simple Heat resources and even Heat nested stacks.
- The second approach is to use MuranoPL actions to expose specific workflows for the application scaling process. It can write complex, sophisticated scaling scenarios for scaling the application according to the developers need.