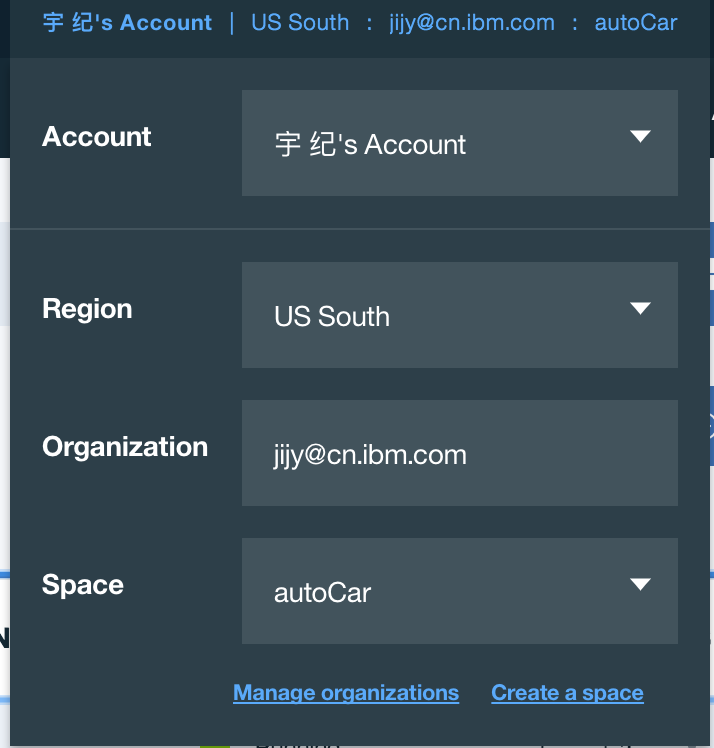
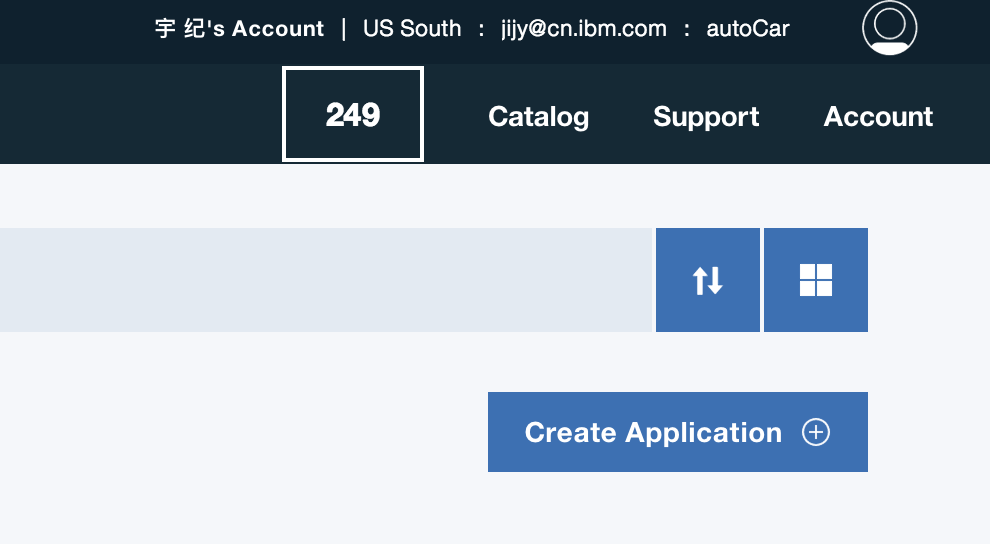
Deploy a Node.js to Bluemix & Edit code in DevOps

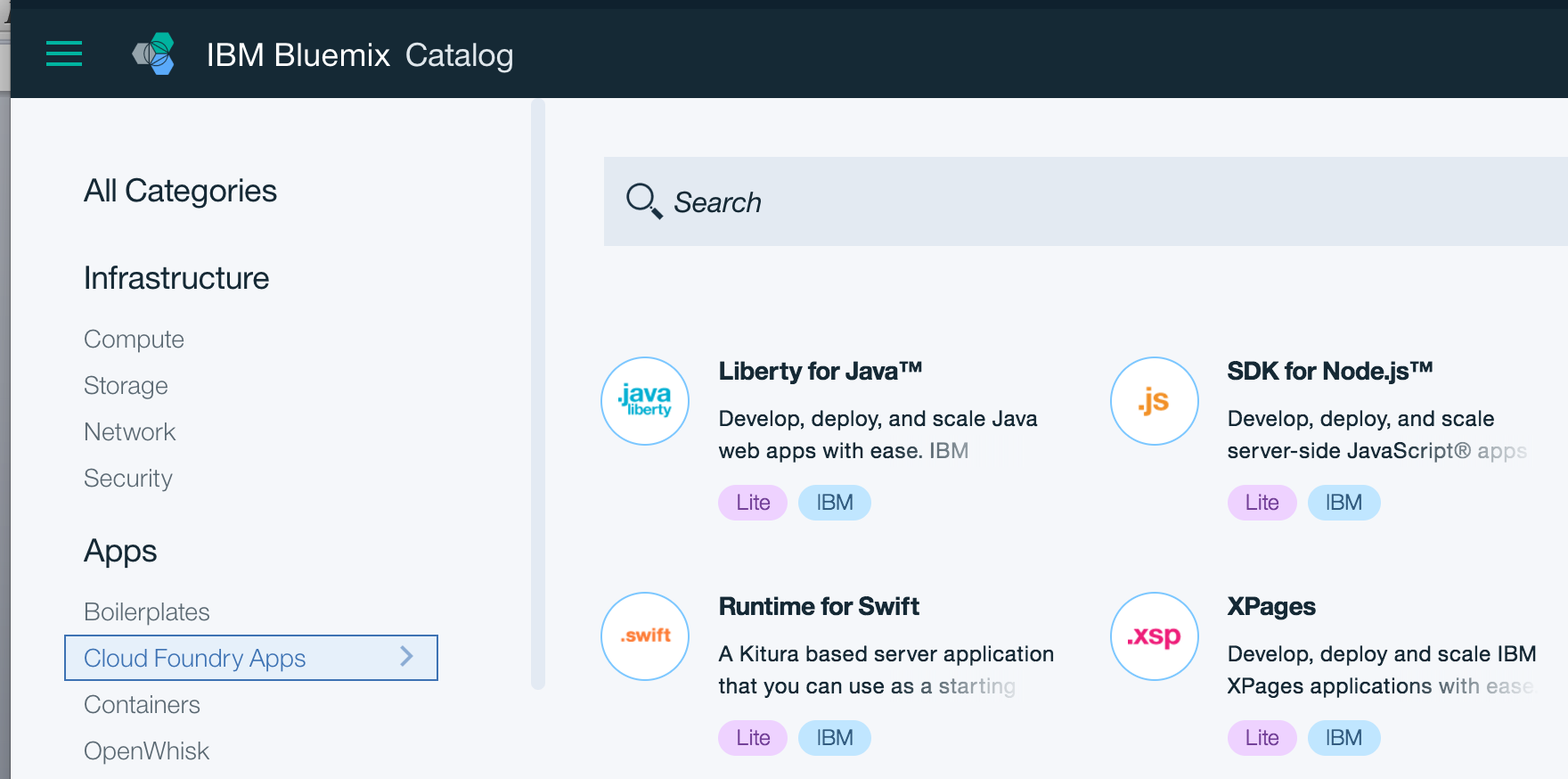
1. First of all, you should have a Bluemix account and a Git account.
2. Login your account and create a workspace. Click the button upper right to scan your profile. Make sure you are in the US South Region.



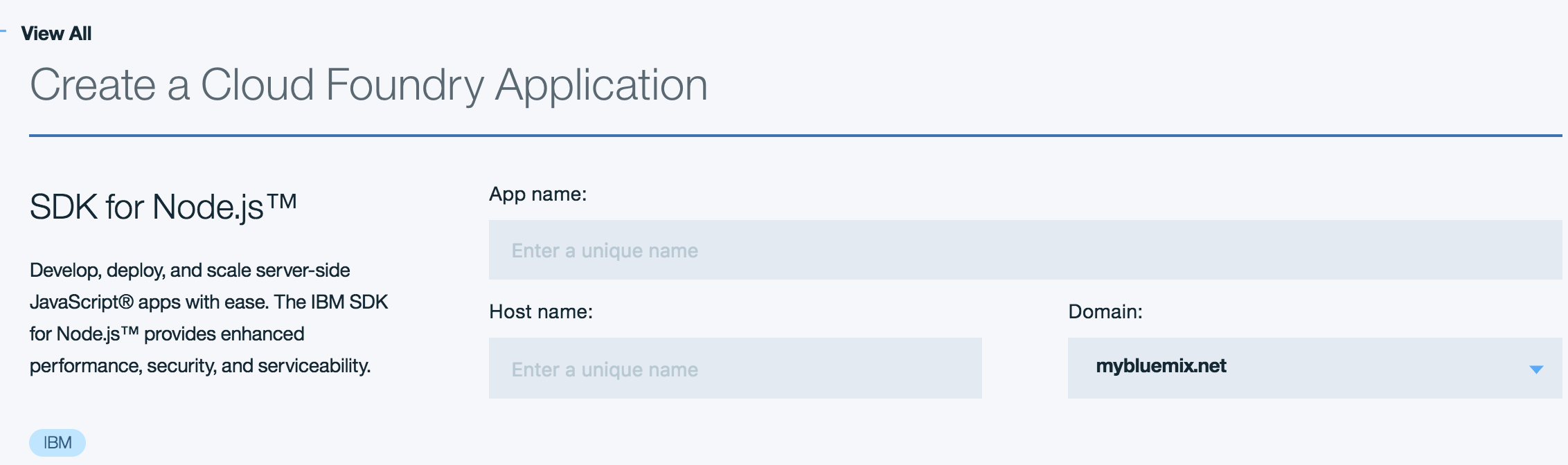
1. Click the “Create Application” button.



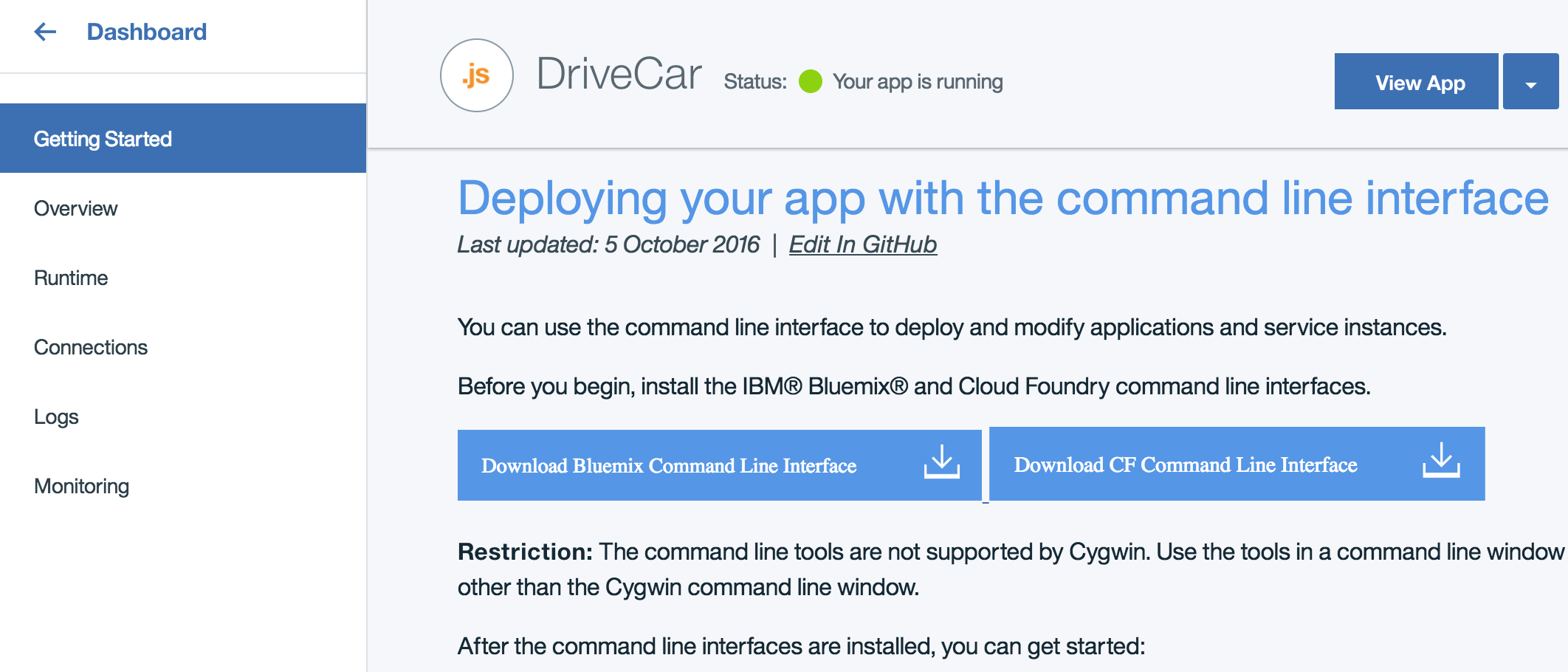
1. Select “Cloud Foundry App” in the left sidebar. Click “SDK for Node.js”.



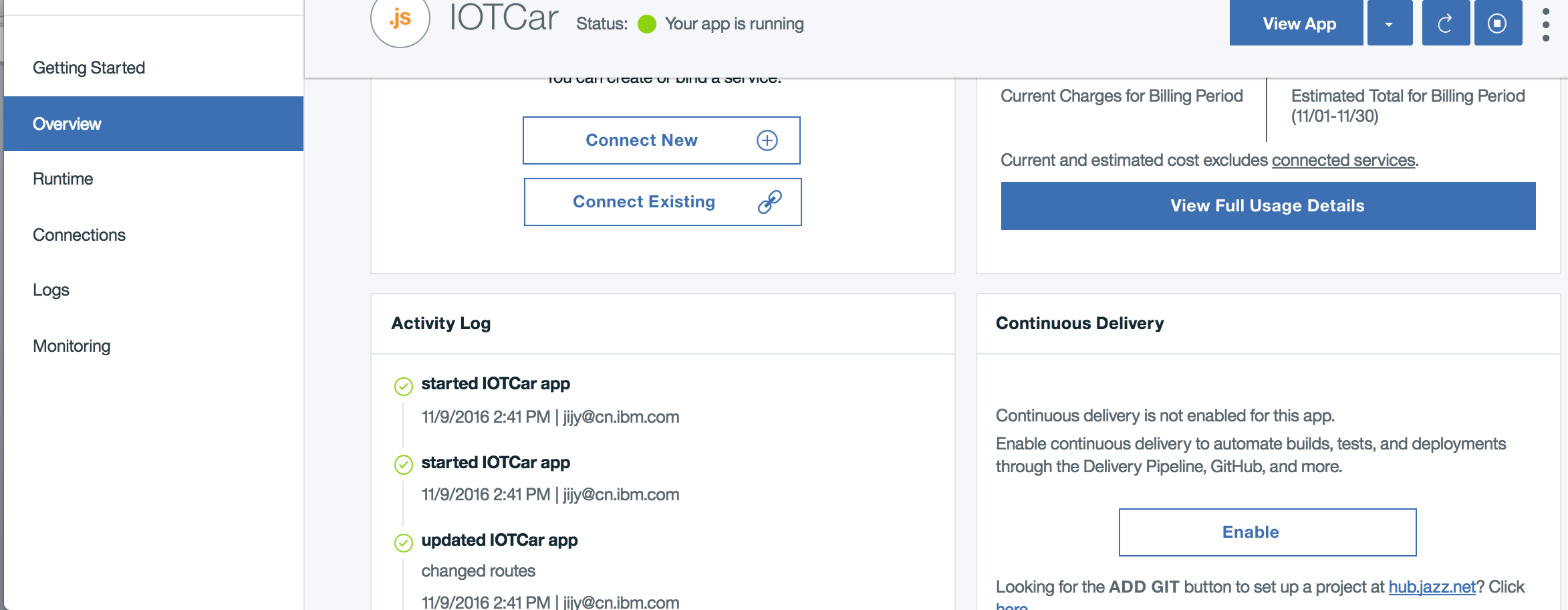
1. Give a unique name for your app. Your app name will generate the host name automatically.



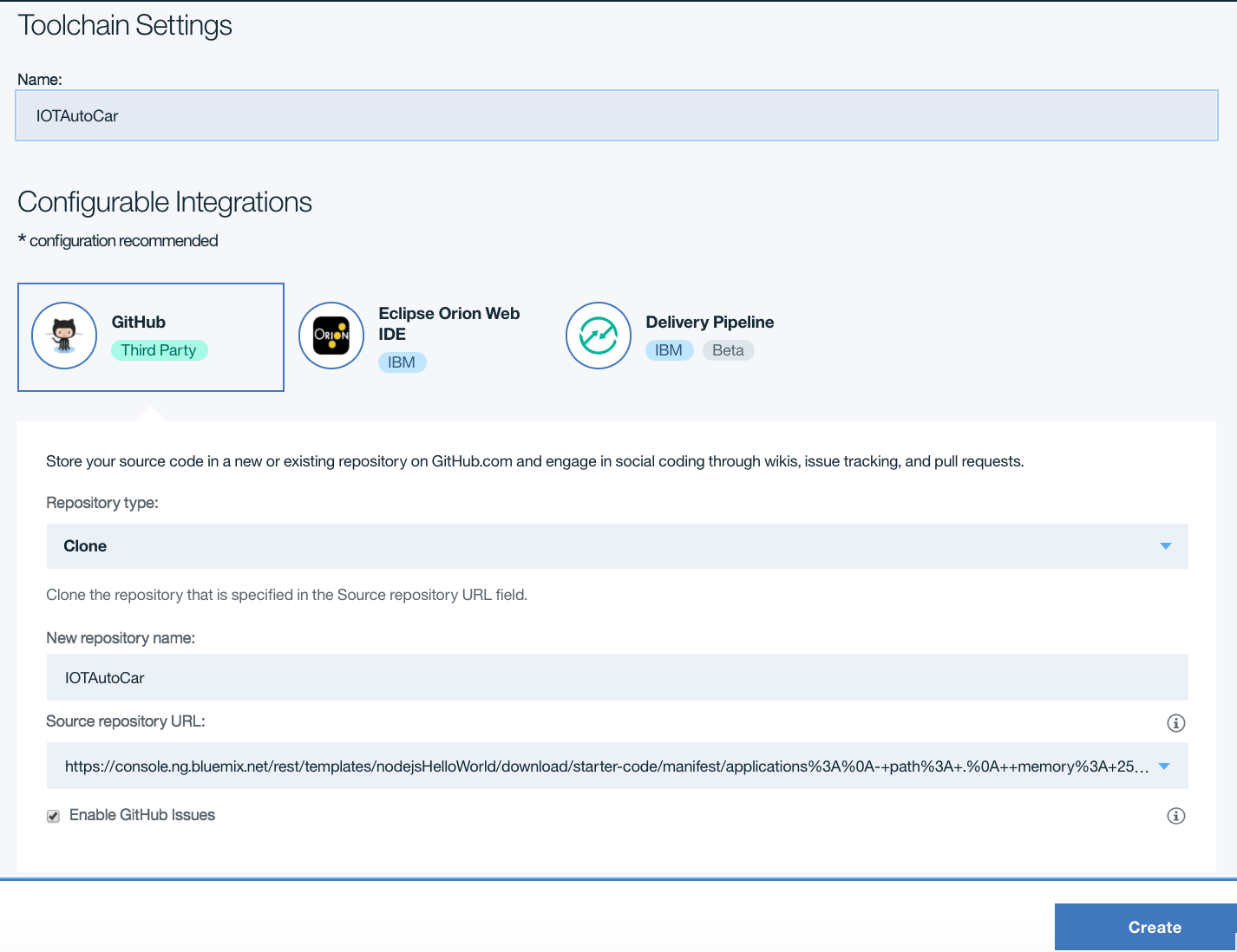
1. Click “Create” button lower right. The page jump to the “getting Started”.



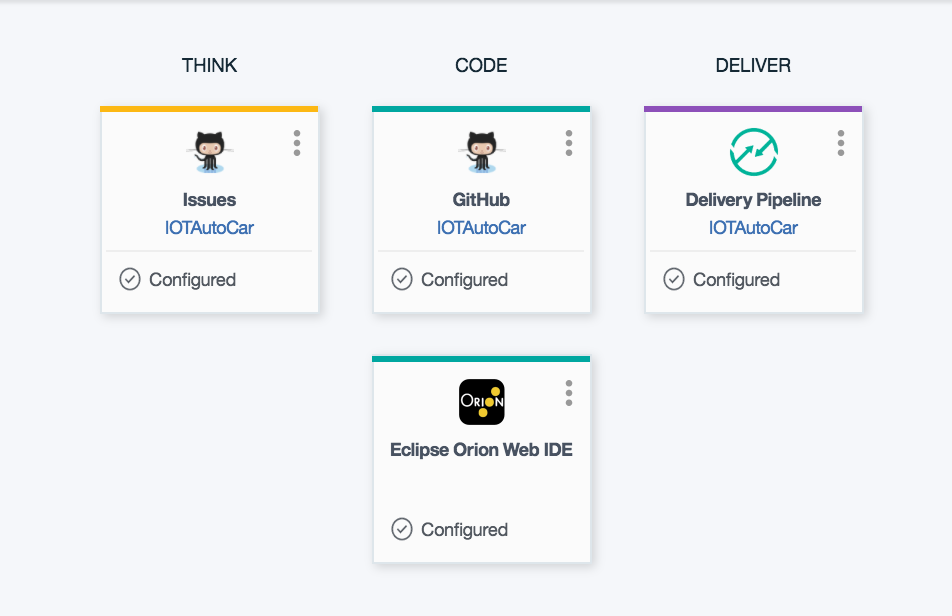
1. Select “Overview” in the left menu. Click the button “Enable” lower right.



1. Select GitHub and give a name for your new repository. Click “Create”.



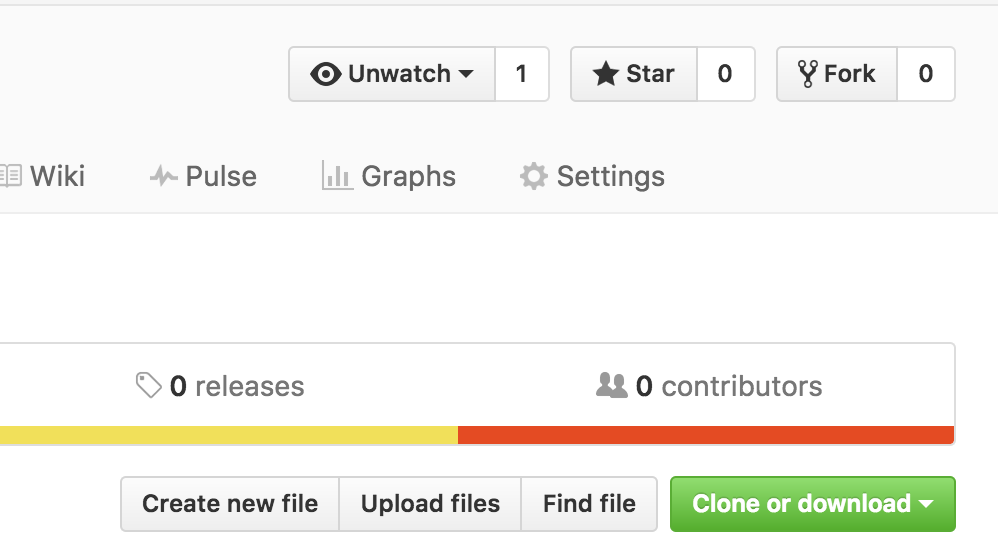
1. Select the second item “CODE” to scan your code on git. At this time, your Node.js program is just empty.



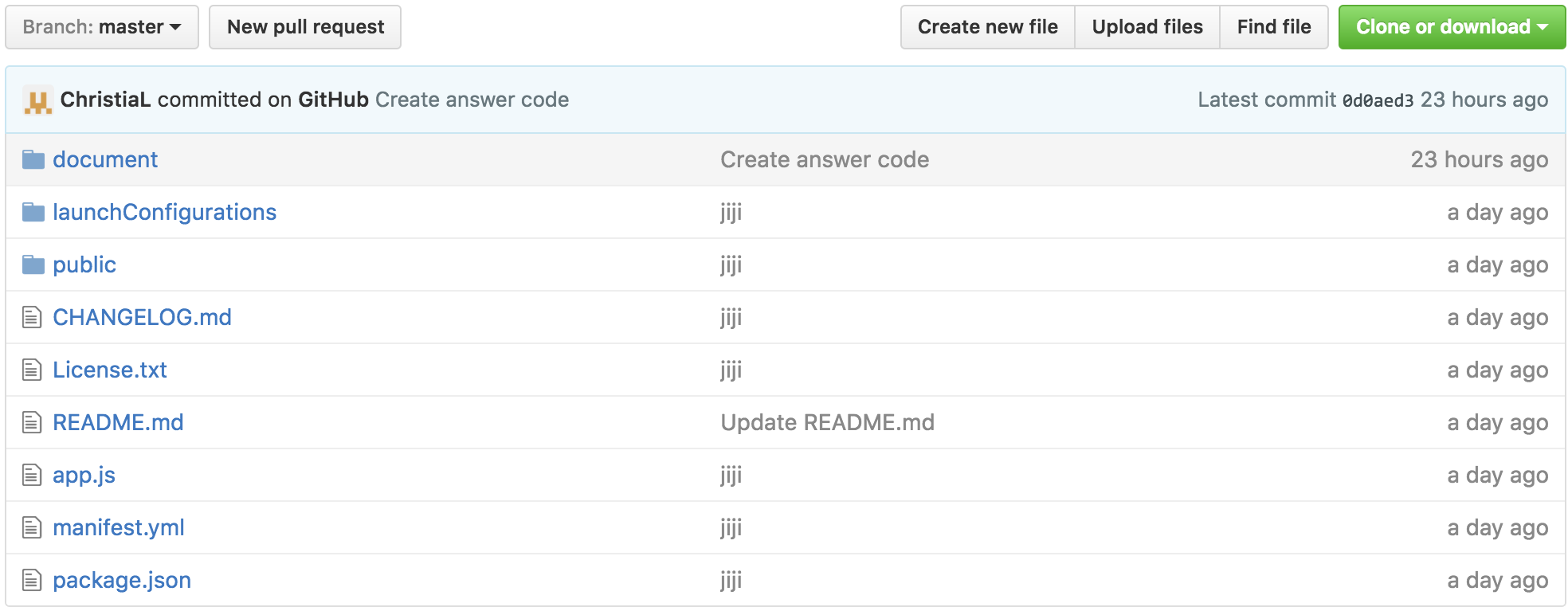
1. Open a new tab with following link.

<https://github.com/ChristiaL/IOTCar>

Click the button “Fork” upper right in this page. Select the repository you created just now to fork the demo code.



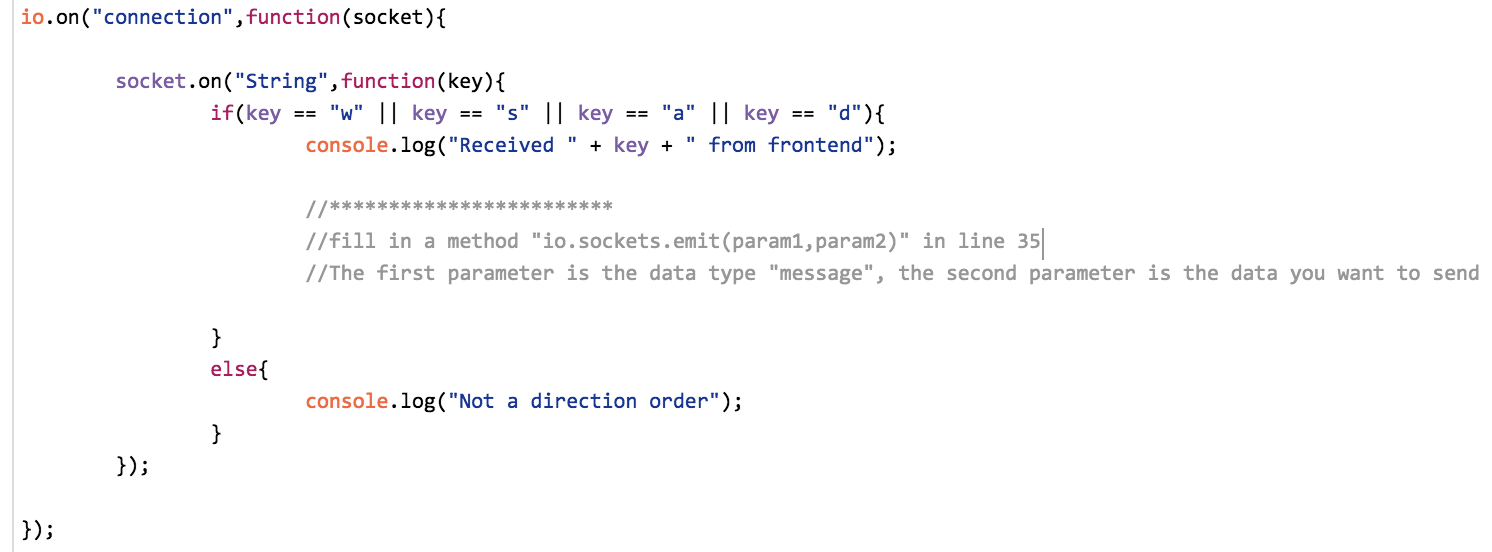
1. Back to your Git page to scan the new demo code. Now the code is missing some key code about sending data with socketIO which needs user to fill in. If you are not good at coding, please open the folder “document” to scan and copy the answer to the code.



1. Open “index.html” under the public folder. Fill in your Bluemix host name(step 5) in line 17. Line 39,50,61,72 need to fill in a common method “socket.emit()”. The first parameter declares the data type is “String”. The second parameter defines the data is “w” or ”s” or ”a” or ”d”(up/down/left/right).



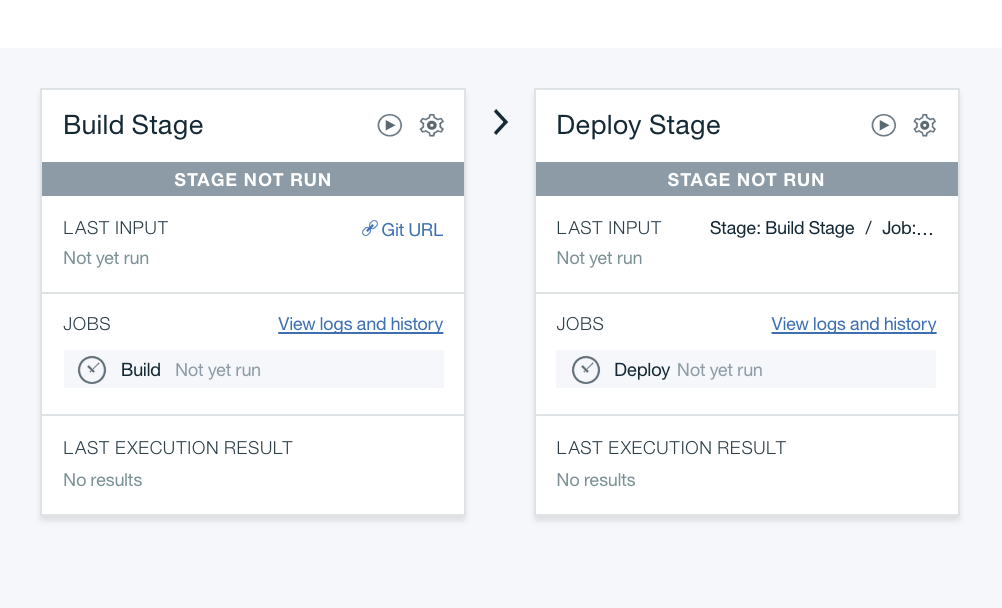
1. Open “app.js” in the root directory. Fill in a method in line 35. The first parameter declares the data type is “message”. The second parameter defines the data is the parameter “key” in the callback function.



1. Open “manifest.yml” in the root directory. Change the name and host name to yours.



1. Back to the DevOps page (step 9). Click the third item(DELIVER). Click the start button upper right in “Build Stage” field. After build success, click the start button upper right in “Deploy Stage”.



1. The last step, after deploy success, open your bluemix link (yourname.mybluemix.net) to scan your new program.

Congratulations! You have successful deploy a node.js program to bluemix!

