

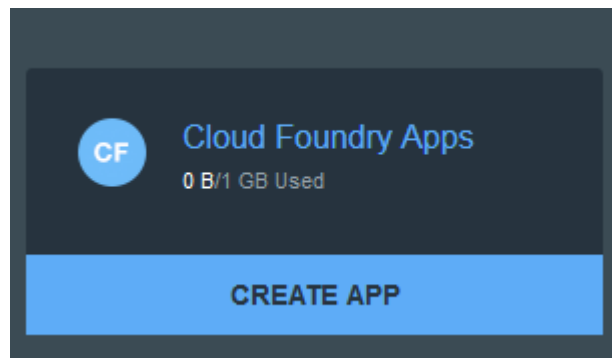
IBM Predictive Analytics Service for Bluemix

General Discussion on Application Development

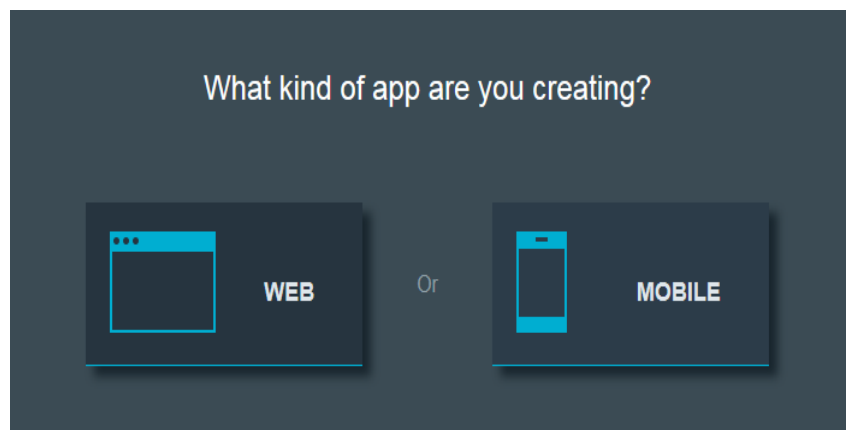
Prepare to Develop the Application

Bluemix makes it very easy to get a new application started. It is worth using the Bluemix 'create an app' at least once to get a good feel for what is involved.

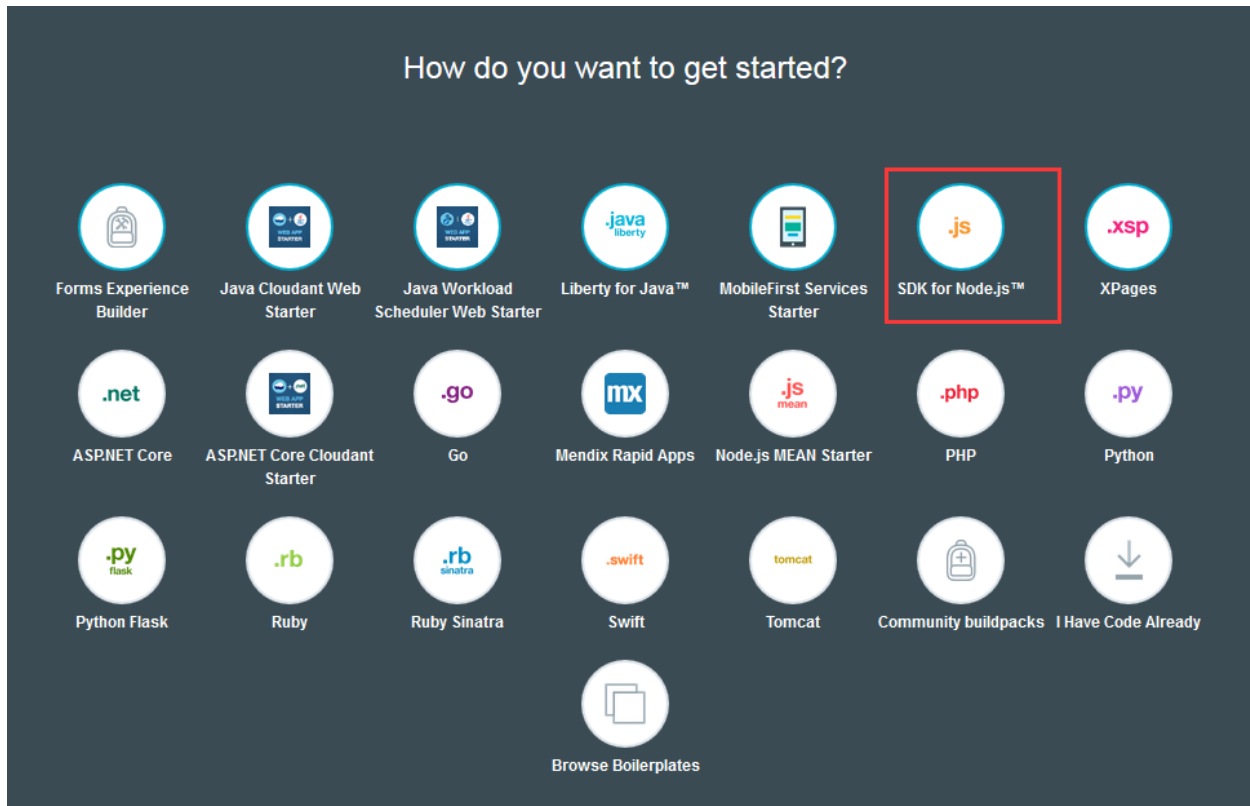
1. Go to the Bluemix web site. We'll be using the 'Development instance' in this example which is at <https://console.stage1.ng.bluemix.net/>
2. Log in using your account details or sign up now for an IBM ID and access to Bluemix, it is free.
3. You may create an 'org' and 'space' for doing this sample development in, the names used are up to you.
4. Once on the dashboard for your development space push the 'Create an App' button



5. In this example we will create a NodeJS application but any language capable of making REST service calls will do.
6. choose "WEB"

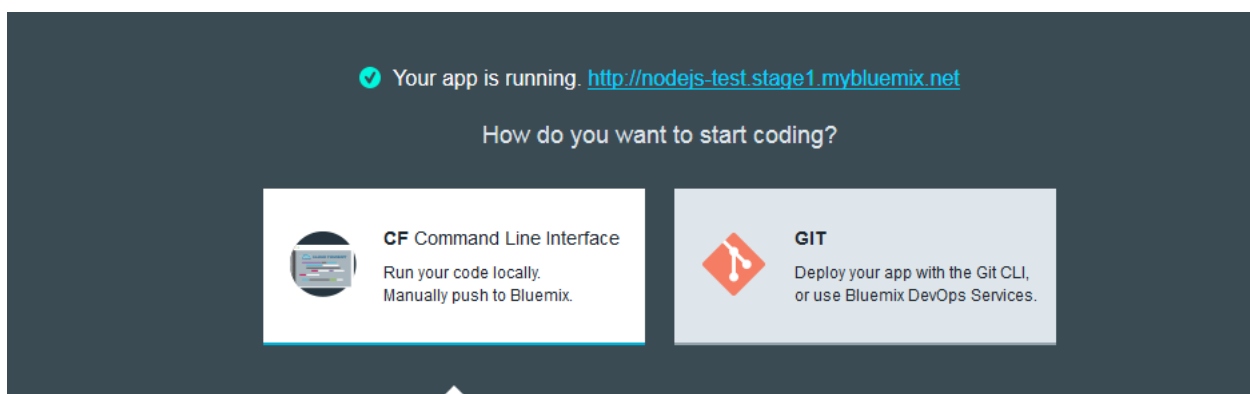


Then choose "NodeJS SDK"



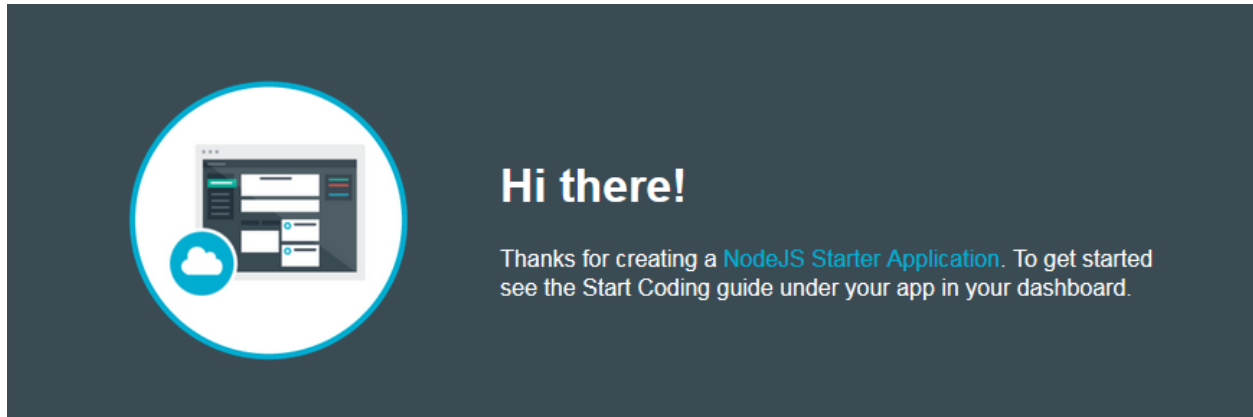
7. Then your application will be there and running. It's a simple "NodeJS Starter Application". If you want more features, you could push your own NodeJS code onto this application by either way:

- a) CF Command Line Interface
- b) GIT

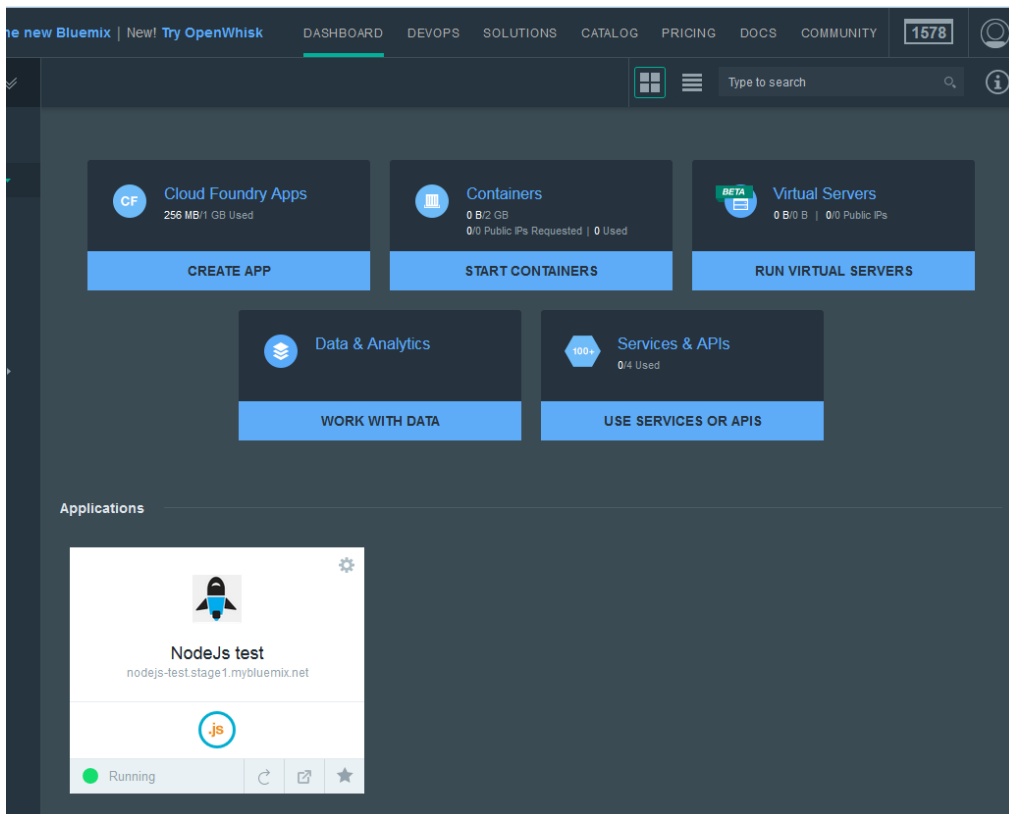


Deploying your app with the command line interface

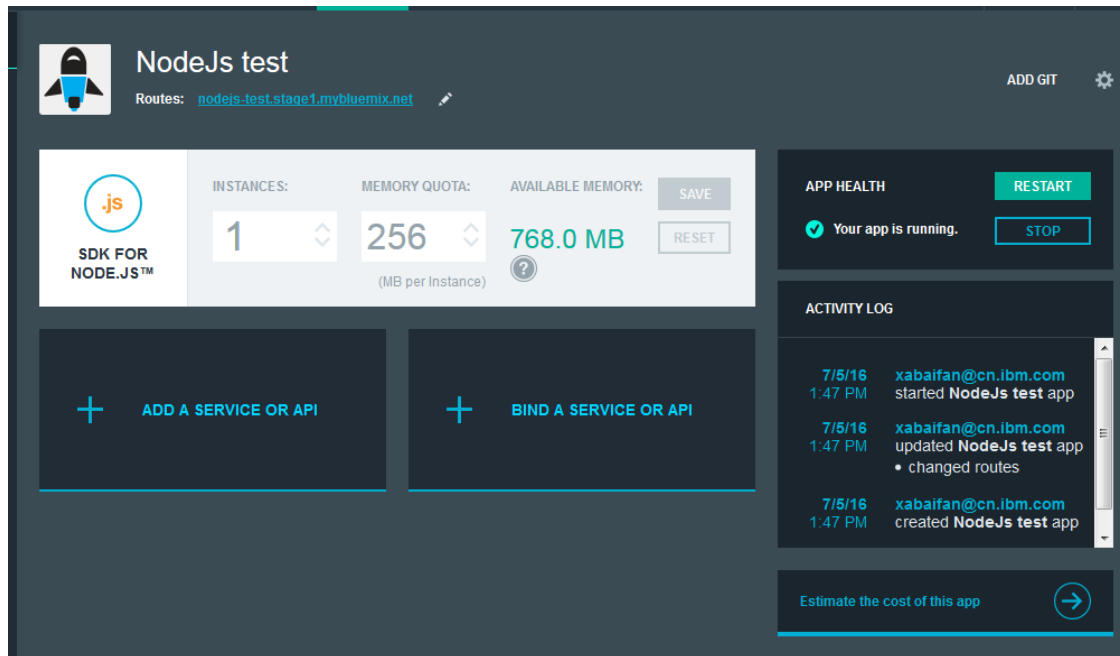
8. If we look into the newly deployed application through the URL (it's beside the "Your app is running"), you could see the welcome page. That means your application is successfully deployed.



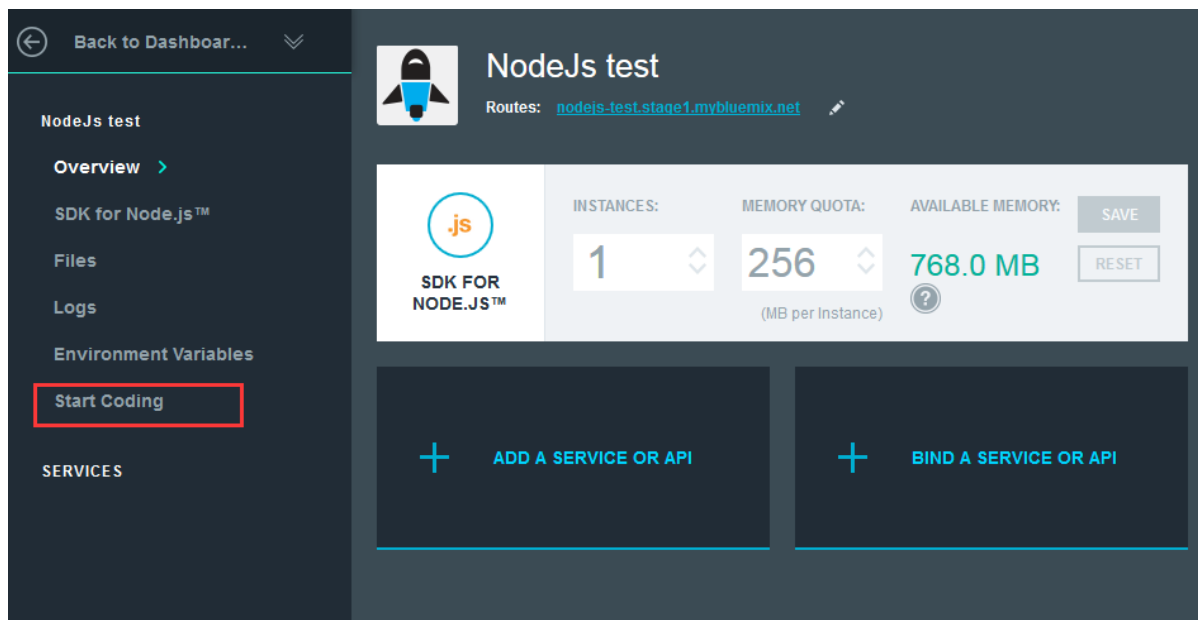
9. A number of things happen when you push 'Create' on your new application. A viable 'shell' application for NodeJS is created and 'pushed' out to Bluemix. The DNS routing entries that let you get to this application are entered. Then the application is started, a basic 'Hello World'. you can see the application in your dashboard.



10. Click on the application you just deployed, you could see more details.



11. The fun begins when we download this application and work with in on our desktop. When we select 'Start coding' on our application we get a help in installing the CF command line tool and downloading our sample application by clicking on the 'Download the starter code'. But before you do that, please install the Bluemix and CF command line interface, and configure their install directory into your Environment Variable "%Path%".



Deploying your app with the command line interface

Last updated: 05 May 2016

You can use the command line interface to deploy and modify applications and service instances.

Before you begin, install the IBM® Bluemix® and Cloud Foundry command line interfaces.

[Download Bluemix Command Line Interface](#)



[Download CF Command Line Interface](#)



Restriction: The command line tools are not supported by Cygwin. Use the tools in a command line window other than the Cygwin command line window.

After the command line interfaces are installed, you can get started:









①

Download your starter code.

[DOWNLOAD STARTER CODE](#)



12. What is in this application bundle when we download it?

Name	Date modified	Type	Size
 public	7/5/2016 3:22 PM	File folder	
 .cfignore	7/5/2016 3:22 PM	CFIGNORE File	1 KB
 .project	7/5/2016 3:22 PM	PROJECT File	1 KB
 app.js	7/5/2016 3:22 PM	JScript Script File	1 KB
 CHANGELOG.md	7/5/2016 3:22 PM	MD File	1 KB
 manifest.yml	7/5/2016 3:22 PM	YML File	1 KB
 package.json	7/5/2016 3:22 PM	JSON File	1 KB
 README.md	7/5/2016 3:22 PM	MD File	1 KB

We get the basic framework for a NodeJS application (app.js and package.json) as well as the manifest.yml used when we 'push' this application up to Bluemix. There is also a 'README.txt' to help you take your next steps in building a NodeJS application.

Push the Modified Application Back to Bluemix

1. The following comments (under 'Download starter code' button in Step 11) shows how to push back the application after you modified.

2. Extract the package to a new directory to set up your development environment.

3. Change to your new directory.

```
cd your_new_directory
```



4. Make changes to your app code as you see fit. We suggest making sure the app runs locally before you deploy it back to IBM® Bluemix®.

One file you should take note of is the manifest.yml file. When deploying your app back to IBM® Bluemix®, this file is used to determine your application's URL, memory allocation, number of instances, and other crucial parameters. You can read more about the manifest file in the Cloud Foundry documentation.

5. Connect to IBM® Bluemix®.

```
bluemix api https://api.stage1.ng.bluemix.net
```



6. Log in to Bluemix.

For more details, please see 'VIEW DOCS'

- 7 Deploy your app to Bluemix. For more information about `cf push` command, see [Uploading your application](#).

```
cf push "NodeJs test"
```



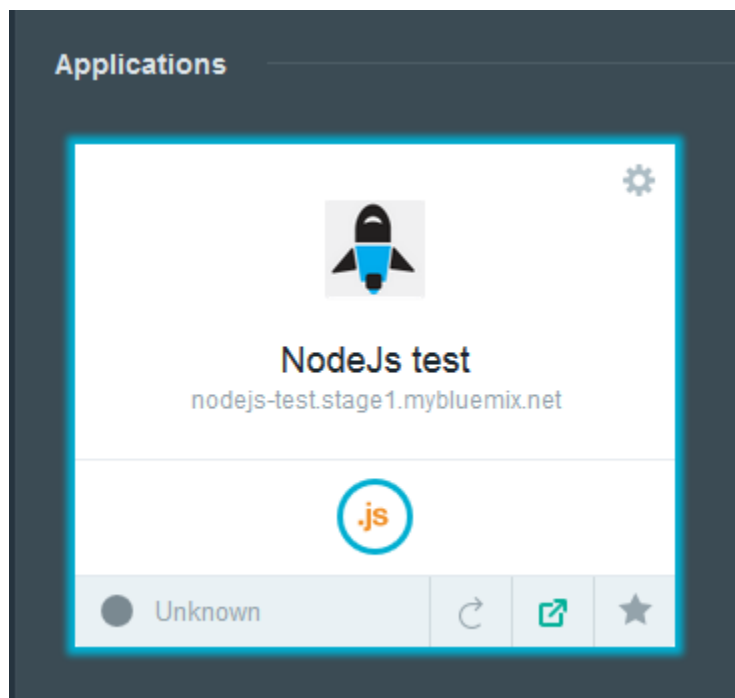
- 8 Access your app by entering the following URL into your browser:

```
nodejs-test.stage1.mybluemix.net
```

[VIEW DOCS](#)[VIEW APP OVERVIEW](#)

The entire bundle is transferred up to Bluemix and you will be notified as it goes through all phases of deployment and finally restarts.

Once restarted the application can be launched right from the Bluemix dashboard by pushing the 'open app in a new page' button.

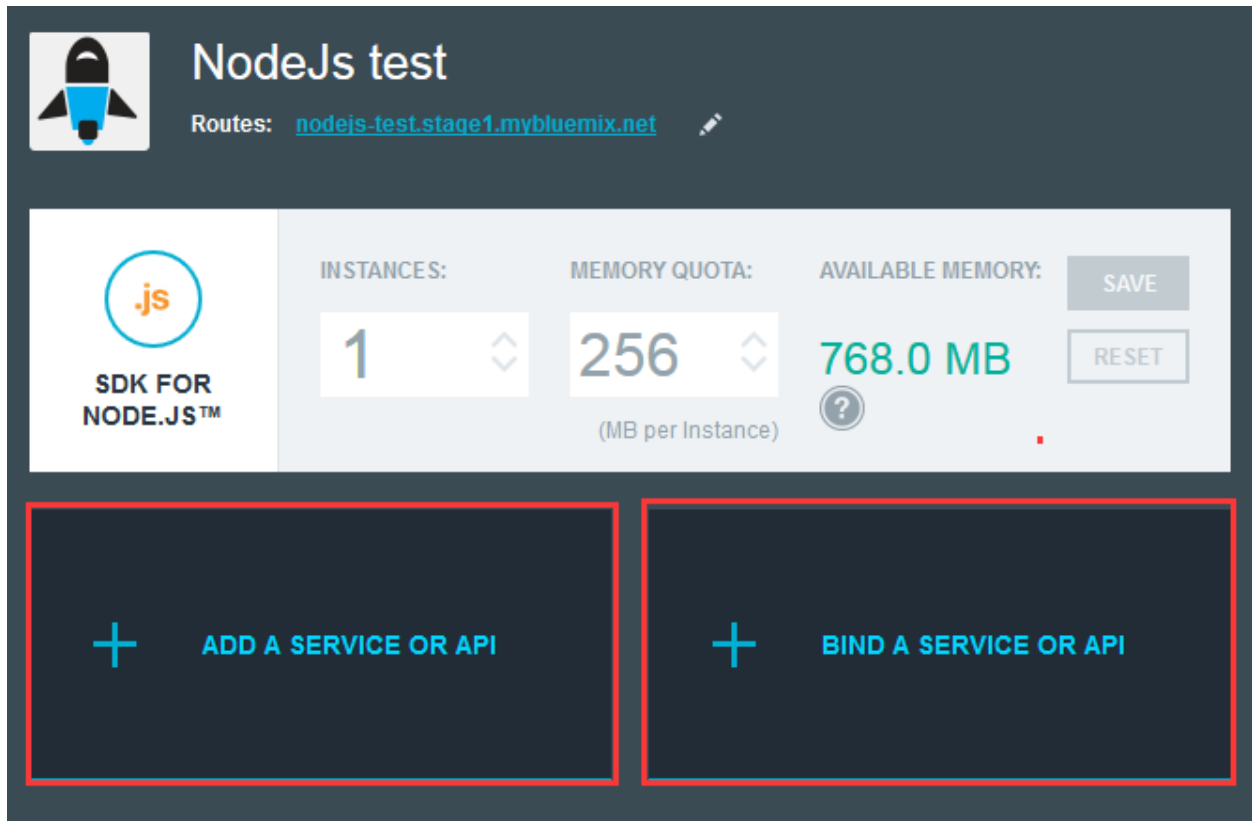


Testing Your Bluemix Application During Development

You may provision an instance of the IBM Predictive Analytics service at any point in time and get the connectivity information you need to test your application from your desktop. You may also 'push' your

application back up to Bluemix at any point in time and test it there. To test your application from your desktop you will need a compatible development environment. In the example we've been looking at here this would mean a NodeJS server install.

Use the "Add/Bind a Service or API" button on your Bluemix application to connect it to your service instance. "Add a Service or API" means you newly add a service and bind it to your application. "Bind a Service or API" means you bind an existing service or API to your application.



The screenshot shows the configuration page for a Node.js application named "NodeJs test". At the top, there is a rocket icon and the application name. Below the name, the routes are listed as nodejs-test.stage1.mybluemix.net. The main configuration area includes a "SDK FOR NODE.JS™" icon, a dropdown for "INSTANCES:" set to "1", a dropdown for "MEMORY QUOTA:" set to "256" (with a note "(MB per Instance)"), and a display for "AVAILABLE MEMORY:" showing "768.0 MB". There are "SAVE" and "RESET" buttons. At the bottom, two buttons are highlighted with red boxes: "+ ADD A SERVICE OR API" and "+ BIND A SERVICE OR API".

Since it's the first time we use it, please choose "Add a Service of API".

In "Data and Analytics" category, find "Predictive Analytics"

Data and Analytics

Essential data services;
limitless possibilities

HELP ME PICK



Apache Spark
IBM



Cloudant NoSQL DB
IBM



dashDB
IBM



DataWorks
IBM



Elasticsearch by
Compose
IBM



Geospatial Analytics
IBM



IBM DataStage on Cloud
IBM



IBM DB2 on Cloud
IBM



IBM Graph
IBM BETA



IBM Master Data
Management on Cloud
IBM



Insights for Twitter
IBM



MongoDB by Compose
IBM



PostgreSQL by
Compose
IBM



Predictive Analytics
IBM



Redis by Compose
IBM



Streaming Analytics
IBM



Weather Company Data
for IBM Bluemix
IBM



ClearDB MySQL
Database
Third Party



ElephantSQL
Third Party



Predictive
Analytics
IBM

PUBLISH DATE
05/26/2016

AUTHOR
IBM SPSS

TYPE
Service

VIEW DOCS

IBM Predictive Analytics is a full-service Bluemix offering that makes it easy for developers and data scientists to work together to integrate predictive capabilities with their applications. Built on IBM's proven SPSS analytics platform, Predictive Analytics allows you to develop applications that make smarter decisions, solve tough problems, and improve user outcomes.

Pick a plan

Monthly prices shown are for country or region: [China](#)

Plan	Features
✓ Free Plan	1 Service instance (2 models maximum) 5,000 Real-time or batch predictions 5 Compute-hours for analysis and model building



The free plan provides you with a single instance of the Predictive Analytics service with a maximum of 2 models, 5,000 predictions per month, and 5 hours of compute time for analysis and model building operations.

Paid Plan	Service instance (20 models per instance) Real-time predictions Batch predictions Analysis and model building compute-hour	\$10.50 USD/instance \$0.53 USD/1,000 Real-time predictions \$0.53 USD/1,000 Batch predictions \$0.47 USD/Compute-hour
-----------	---	---

TERMS

Add Service

Space:
haku_test

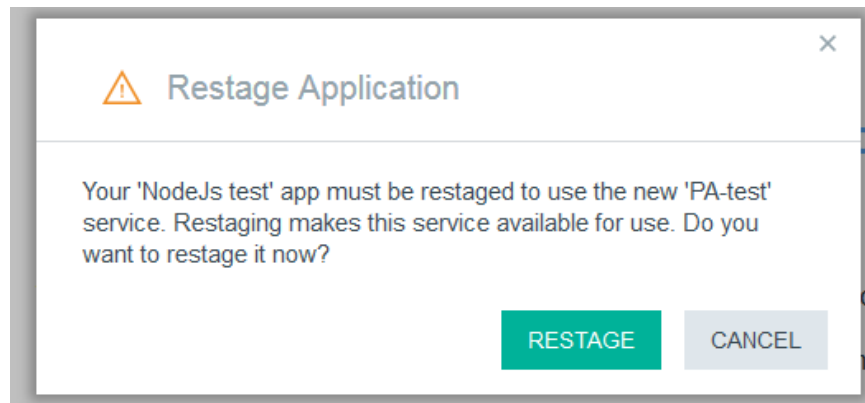
App:
NodeJs test nodejs-test.stage1.my...

Service name:
PA-test

Selected Plan:
Free Plan

CREATE

Set your service name and fee plan, click on "Create". Now the service is created and bound to your application. You may click on the "Restage" button to restage your application and let the bound service work.



Go to application "Overview" page you could see the bound service. Once the application is bound you can click on the "Show Credentials" button on the bound service icon to get the connectivity information you need to test your application from your desktop.

The screenshot displays the IBM Bluemix console interface. At the top, the navigation bar includes links for 'IBM Bluemix', 'Ready? Try the new Bluemix', 'New! Try OpenWhisk', and tabs for 'DASHBOARD', 'DEVOPS', 'SOLUTIONS', 'CATALOG', and 'PRICING'. The main header shows 'NodeJs test' with a rocket icon and the route 'nodejs-test-stage1.mybluemix.net'. The left sidebar contains a menu for 'NodeJs test' with options: 'Overview' (highlighted with a red box), 'SDK for Node.js™', 'Files', 'Logs', 'Environment Variables', 'Start Coding', and 'SERVICES' with 'Predictive Analytics'. The main content area shows the application's status: 'INSTANCES: 1', 'MEMORY QUOTA: 256 (MB per Instance)', and 'AVAILABLE MEMORY: 768.0 MB'. Below this are buttons for 'ADD A SERVICE OR API' and 'BIND A SERVICE OR API'. The 'Predictive Analytics' service is listed with the plan 'PA-test' and 'Free'. A 'Show Credentials' button (highlighted with a red box) is visible, leading to a section titled 'Instantiating Credentials' which displays a JSON configuration for the service.

```
{
  "pm-20": [
    {
      "name": "PA-test",
      "label": "pm-20",
      "plan": "Free",
      "credentials": {
        "access_key": "6h0t5H98wJo6x1EDtg/",
        "url": "https://palbys1.pmservice.i"
      }
    }
  ]
}
```

Modify the NodeJS example app.js file to use either the Bluemix provided environment variables VCAP_APP_HOST, VCAP_APP_PORT and VCAP_SERVICES or values you set to test from your desktop.

Please read the docs under Sample1 and Sample2 folder to know how the Sample1 and Sample2 works.

IBM Predictive Modeling Service for Bluemix - Sample1.doc and IBM Predictive Modeling Service for Bluemix - Sample2.doc