

# Mobile Device Movement/Accelerometer Sensor

## Hands-On Lab

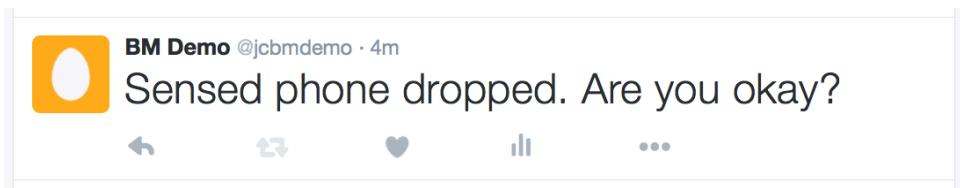
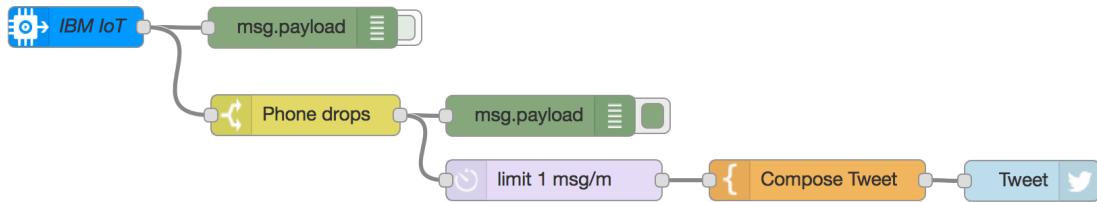
Author: JeanCarl Bisson | [jbisson@us.ibm.com](mailto:jbisson@us.ibm.com) | @dothewww



Topics covered: Node-RED, Watson Internet of Things Platform, and Twitter.

In this lab, we will deploy a web application that measures a mobile device's position, acceleration, and location. The application will send this data to the IBM Watson's Internet of Things Platform service in IBM Bluemix. In the second part of this lab, we will create a simple Node-RED application in Bluemix that listens to these device events and respond by tweeting a message when a sudden acceleration occurs.

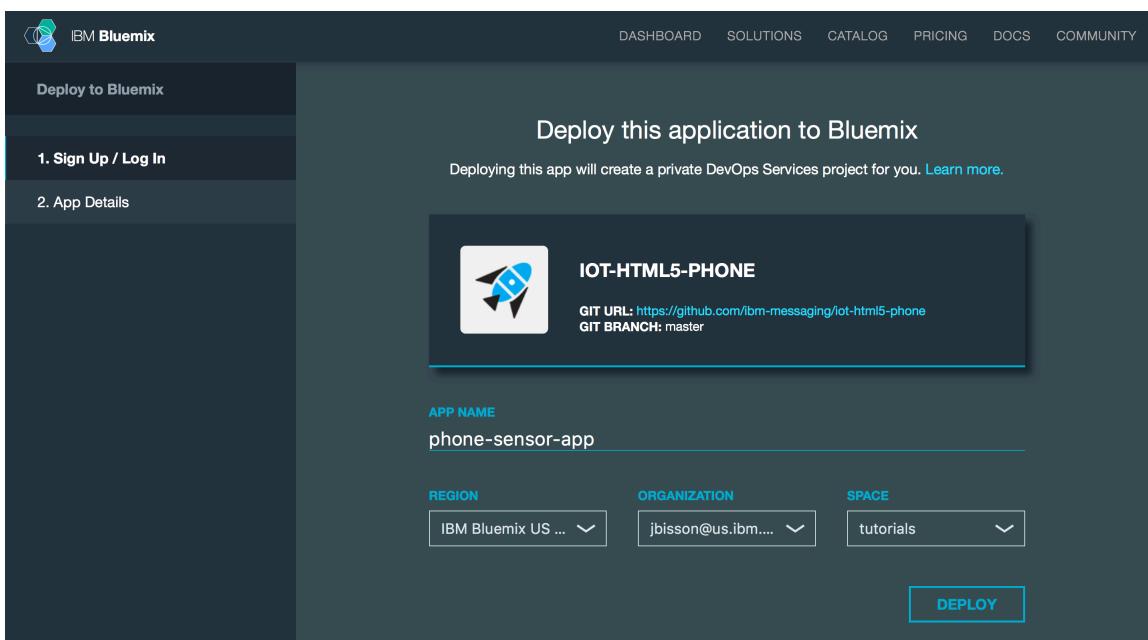
Create a web-based mobile device sensor.....	3
Create an accelerometer sensor/alert system .....	8



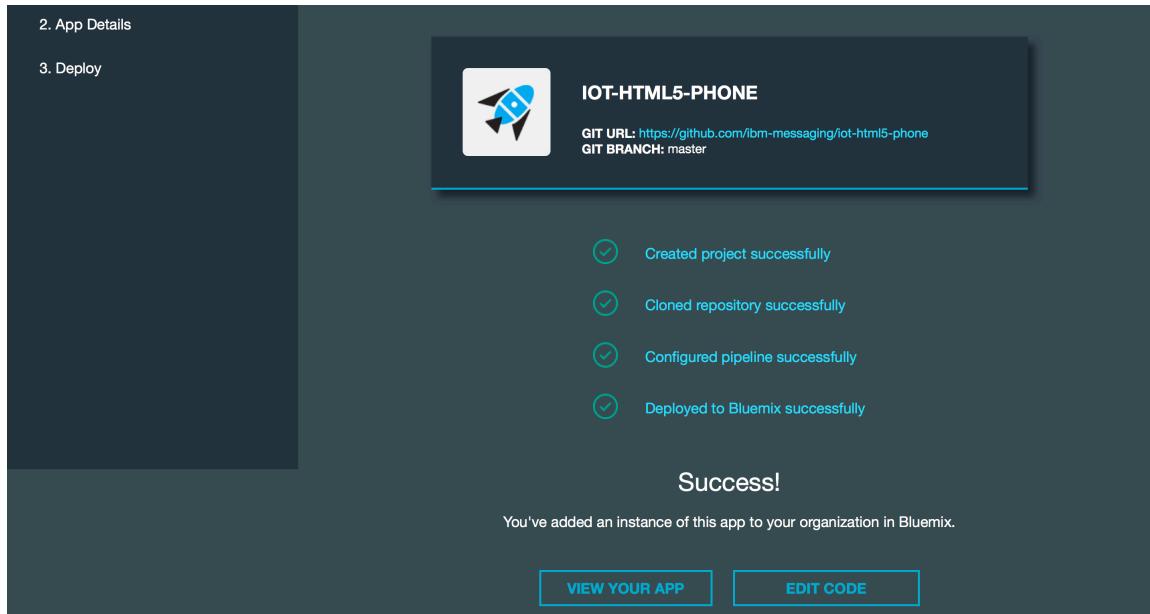
# Create a web-based mobile device sensor

In this section, we will deploy a sample Internet of Things application that reads a mobile device's position, accelerometer and location and sends this data to the IBM Watson IoT Platform service in IBM Bluemix.

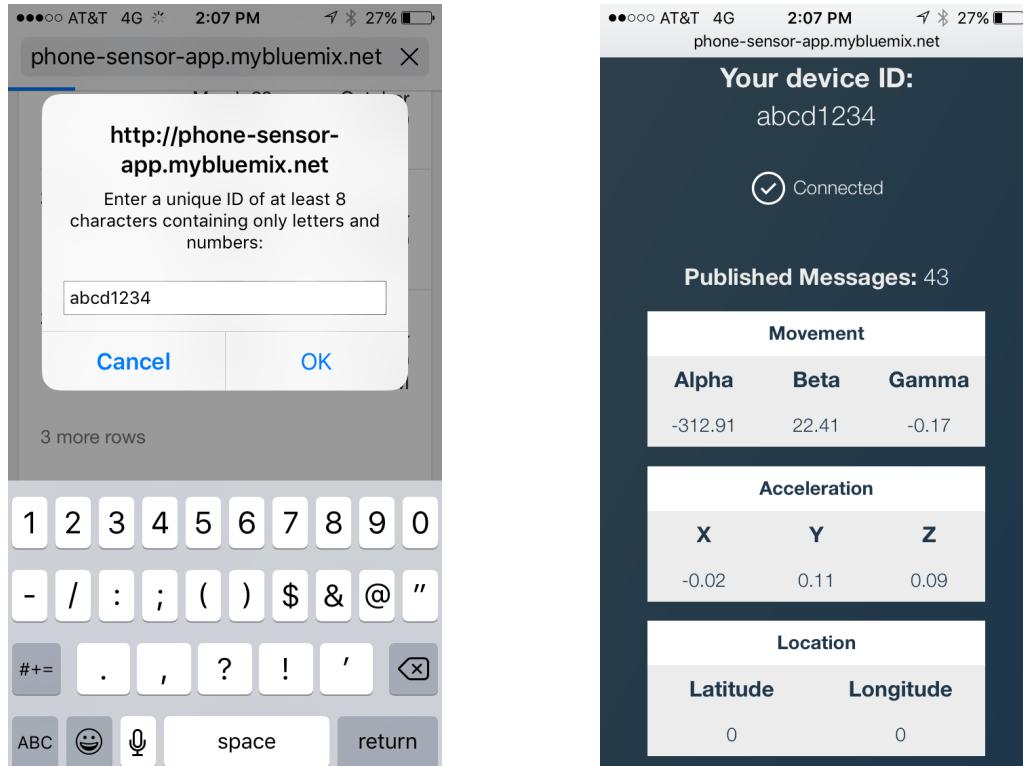
1. Go to GitHub repo <http://ibm.biz/phone-sensor-app>
2. In the Read Me section, click on the  button.
3. Enter a custom app name or use the default given. You can also customize the region, organization, and space the application should be deployed to. Click the Deploy button. Make note of the app name, it will be part of the URL for the application you need to access in the next step.



4. IBM Bluemix will create a new project, clone the repo, and deploy the application to the region and space you specified. When the application has been deployed successfully, you should see two buttons appear at the bottom of the page. Open your application in a browser on your mobile device and visit <http://<app-name>.mybluemix.net> (change <app-name> to whatever app name you chose).



5. Enter a unique ID of at least 8 characters and tap OK. You should see the status change to Connected, and start publishing messages. These messages are sent to the Watson IoT Platform service in Bluemix and contain the movement of the phone and any acceleration the phone may experience.

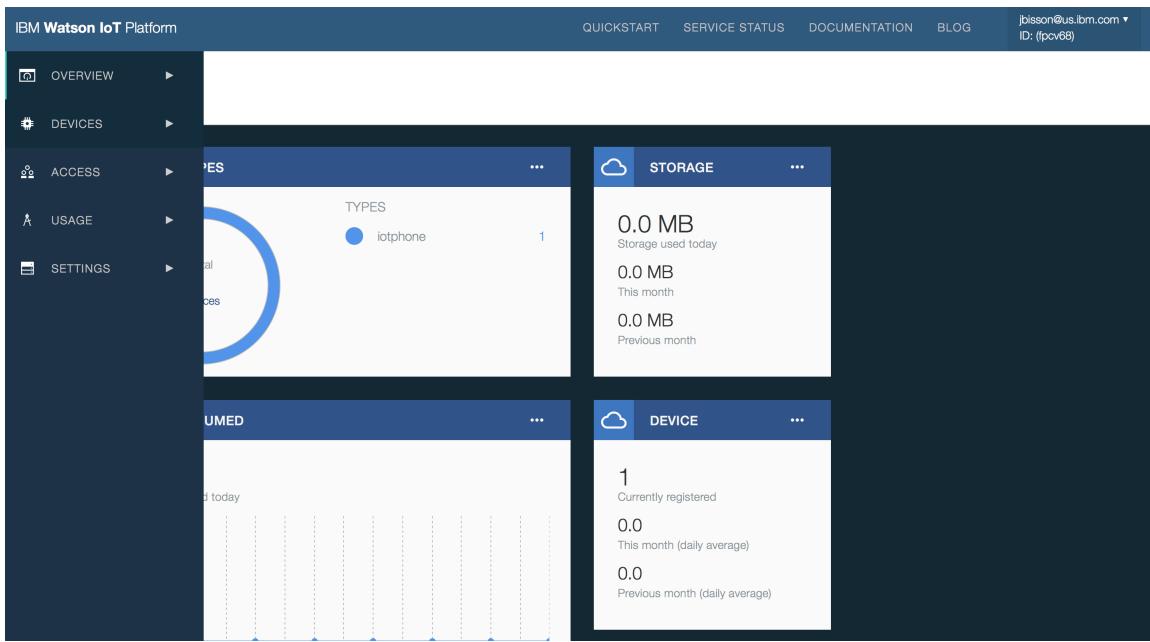


6. Return to the Bluemix dashboard on your computer. Click on the Internet of Things Platform link in the left sidebar. Click on Launch Dashboard link in the first column.

The screenshot shows the IBM Bluemix dashboard with the application name "iot-phone-iotf-service" in the top header. On the left, there is a sidebar with various options like Overview, SDK for Node.js™, Files, Logs, Environment Variables, Start Coding, SERVICES (Cloudant NoSQL DB), and Internet of Things Platform. Under the Internet of Things Platform section, there is a "Launch dashboard" button. The main content area has a title "Hi! Welcome to the Watson IoT Platform" and a sub-section "Connect your devices". It includes a brief description, a "Launch the Watson IoT Platform dashboard" button, and a "Go to docs" button. To the right, there are three sections: "Learn how to build your app", "Learn how to extend your app", and a list of available services (Twilio, Cloudant NoSQL DB, Dash DB, Geospatial Analytics, Time Series Database, IBM Analytics for Hadoop). Each service has a "Find out how" button.

7. This is the IBM Watson IoT Platform dashboard. You can see the connected devices that have been registered through the application you just deployed and ran. Hover over the icons in the left sidebar and click on Devices.

The screenshot shows the IBM Watson IoT Platform Overview dashboard. At the top, there are links for QUICKSTART, SERVICE STATUS, DOCUMENTATION, and BLOG, along with a user profile. The main area features four cards: "DEVICE TYPES" (1 device named "iotphone"), "STORAGE" (0.0 MB used today, month, previous month), "DATA CONSUMED" (0.0 MB consumed today), and "DEVICE" (1 currently registered device). The sidebar on the left has icons for Overview, Device Types, Storage, Data Consumed, and Device.



8. The next screen displays all the devices that have been registered to your instance of the IBM Watson IoT Platform service. You should see the Device ID that you entered on your mobile device. Click on it.

The screenshot shows the "Devices" page in the IBM Watson IoT Platform. The top navigation bar includes links for QUICKSTART, SERVICE STATUS, DOCUMENTATION, BLOG, and user information (jbisson@us.ibm.com \* ID: (fpcv68)).

The main content area is titled "Devices" and features a search bar with filters for "Device ID", "Device Type", "Class ID", "Date Added", and "Location". A "Browse" button is also present. On the right, there is a "+ Add Device" button.

The results table shows one device entry:

Device ID	Device Type	Class ID	Date Added	Location
abcd1234	iotphone	Device	Mar 21, 2016 2:07:44 PM	

9. A dialog displays the recent events that have been sent from that device. Scroll down to the Sensor Information section.

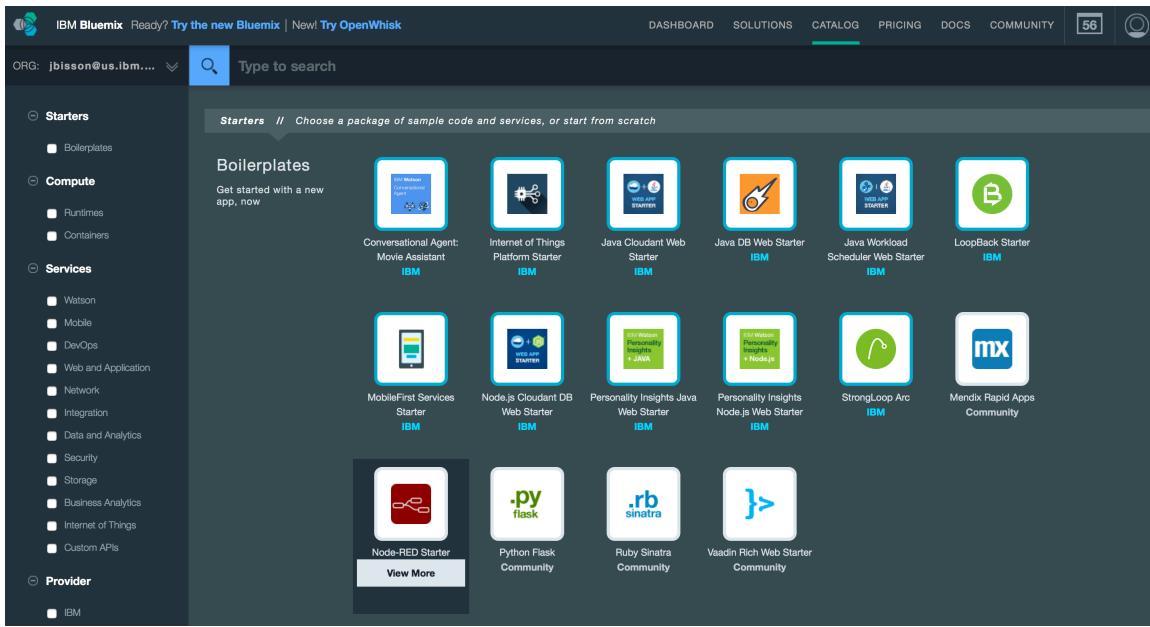
10. The Sensor Information section displays the last data that is submitted to the IoT Platform by the device. As you move the device, the values change.

Event	Datapoint	Value	Time Received
sensorData	d.id	abcd1234	Mar 21, 2016 2:09:28 PM
sensorData	d.ts	1458594568224	Mar 21, 2016 2:09:28 PM
sensorData	d.lat	0	Mar 21, 2016 2:09:28 PM
sensorData	d.lng	0	Mar 21, 2016 2:09:28 PM
sensorData	d.ax	0.01	Mar 21, 2016 2:09:28 PM
sensorData	d/ay	-0.01	Mar 21, 2016 2:09:28 PM
sensorData	d.az	0.06	Mar 21, 2016 2:09:28 PM
sensorData	d.oa	-349.82	Mar 21, 2016 2:09:28 PM
sensorData	d.ob	1.32	Mar 21, 2016 2:09:28 PM
sensorData	d.og	-1.69	Mar 21, 2016 2:09:28 PM

# Create an accelerometer sensor/alert system

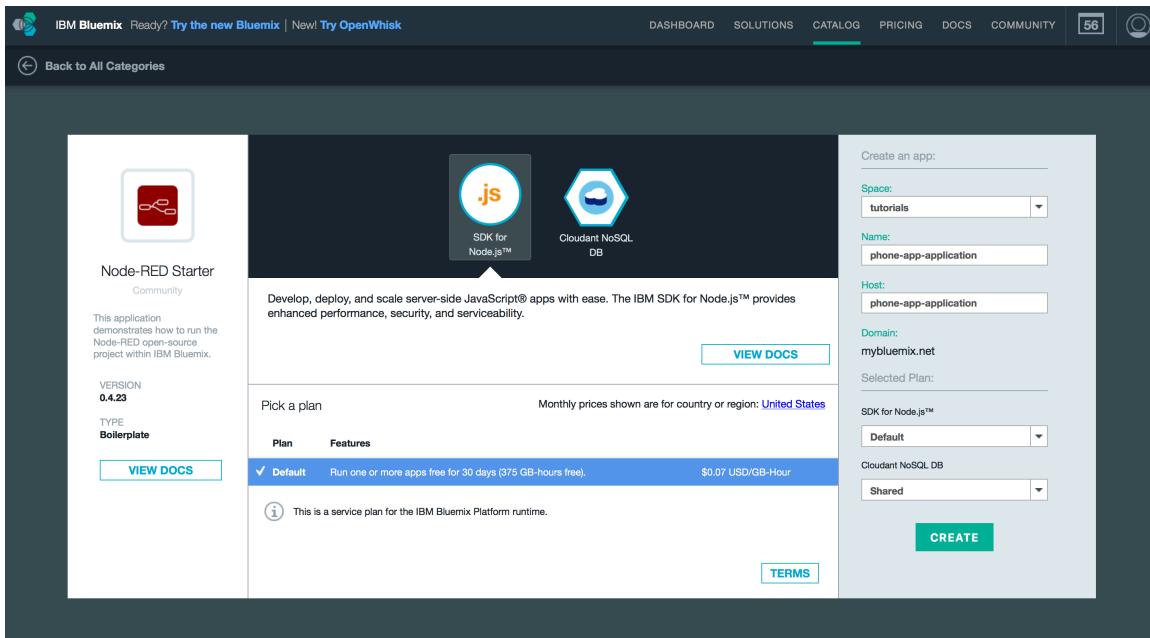
In this section, we will create a simple application that listens to the data the device sends to the Watson IoT platform and respond to a sudden acceleration. This could simulate a sudden fall.

1. In the Bluemix dashboard, click on the Catalog link. Click on the Node-RED Starter tile to create a new Node-RED application.



The screenshot shows the IBM Bluemix Catalog page. On the left, there's a sidebar with categories like Starters, Compute, Services, and Provider. Under Starters, there are sections for Boilerplates, Compute, and Services. Under Services, there are sub-sections for Watson, Mobile, DevOps, Web and Application, Network, Integration, Data and Analytics, Security, Storage, Business Analytics, Internet of Things, and Custom APIs. The main area displays a grid of application starters. One of the starters is the "Node-RED Starter" (Community), which is highlighted with a larger preview window below it. Other starters shown include "Conversational Agent: Movie Assistant", "Internet of Things Platform Starter", "Java Cloudant Web Starter", "Java DB Web Starter", "Java Workload Scheduler Web Starter", "LoopBack Starter", "MobileFirst Services Starter", "Node.js Cloudant DB Web Starter", "Personality Insights Java Web Starter", "Personality Insights Node.js Web Starter", "StrongLoop Arc", and "Mendix Rapid Apps Community".

2. Enter a name for the application. Choose a host name that is unique. Make note of this host name, it will be used in step #7. Click on Create.



The screenshot shows the details for the "Node-RED Starter" application. It includes a summary card with the application icon, name, version (0.4.23), type (Boilerplate), and a "VIEW DOCS" button. Below this, there's a brief description: "This application demonstrates how to run the Node-RED open-source project within IBM Bluemix." It also lists the "SDK for Node.js™" and "Cloudant NoSQL DB" services. A callout box points to the "SDK for Node.js™" service. To the right, there's a "Create an app" form. The form fields are: Space: "tutorials", Name: "phone-app-application", Host: "phone-app-application", Domain: "mybluemix.net", Selected Plan: "Default", and Cloudant NoSQL DB: "Shared". At the bottom right of the form is a "CREATE" button.

3. Wait for the application to finish staging. Click on the Overview link in the left sidebar.

**phone-app-application**

Overview

SDK for Node.js™

Files

Logs

Environment Variables

Start Coding >

SERVICES

Cloudant NoSQL DB

Your application is staging. <http://phone-app-application.mybluemix.net>

Getting Started with:

**Node-RED Flow Editor**

Create IoT apps with Node-RED using the visual flow editor.

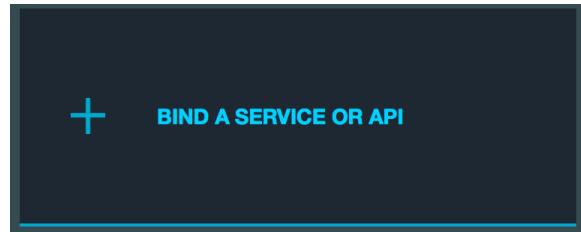
**Start coding with Node-RED**

- 1 After your application has started, click on the **Routes URL** or enter the following URL in a browser:  
<http://<yourhost>.mybluemix.net>
- 2 Click **Go to your Node-RED flow editor**. This opens up a browser-based flow editor that makes it easy to wire together devices, APIs, and online services by using the wide range of nodes included in its palette.

Customizing your Node-RED instance

5 Before you begin, install the Cloud Foundry command line interface.

4. We need to connect our application to the Watson IoT Platform service that was created in the other application. To do this, we “bind the service” to our new application. Both applications will then share access to this service concurrently. The first application sends the data to the service; the second application reads the data from the service. Click on Bind a Service or API.

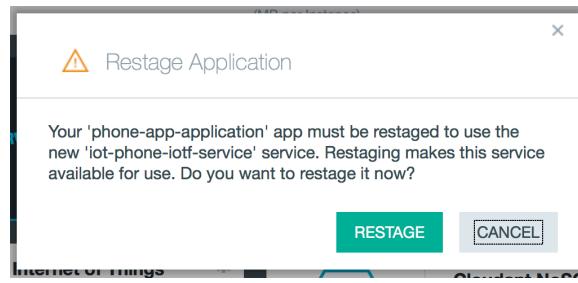


5. Select the IBM Watson IoT Platform service and click Add.

NAME	SERVICE	VERSION
iot-phone-cloudant...	cloudantNoSQLDB	
iotf-service	iotf-service	

ADD CANCEL

6. When prompted that the application must be restaged, click on Restage.



7. When the application has restarted, open a browser tab and visit your application's URL (<http://<host-name-from-step-#2>.mybluemix.net>). This is Node-RED, a graphical programming interface where you can drag and drop nodes (predefined functions) and connect them together to construct a flow (a program of functionality). Click on the red button labeled Go to your Node-RED flow editor.

Node-RED provides a browser-based editor that makes it easy to wire together flows that can be deployed to the runtime in a single-click.

The version running here has been customised for the BlueMix cloud environment.

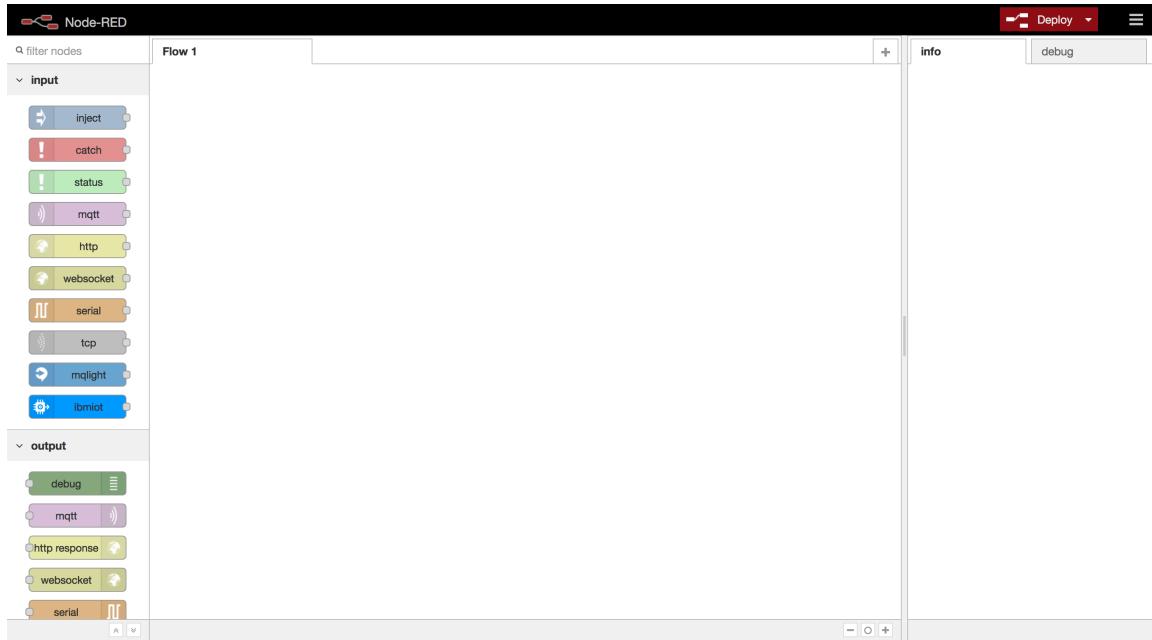
More information about Node-RED, including documentation, can be found at [nodered.org](http://nodered.org).

Go to your Node-RED flow editor

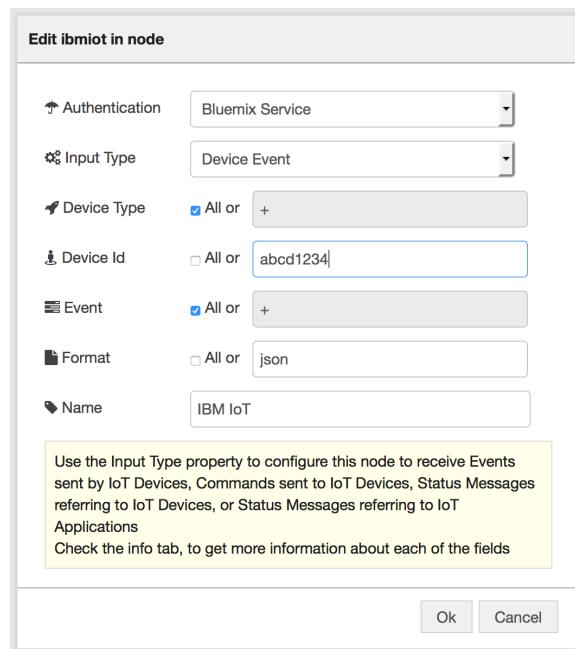
[Learn how to password-protect your instance](#)

[Learn how to customise Node-RED](#)

8. There are four things to note about the Node-RED interface. The palette in the left-sidebar contains nodes that perform pre-defined functionality. Click on each node in the palette and read the details in the info tab in the right sidebar. Drag nodes into the middle canvas and connect the grey nobs together to compose what is called a flow, executed from left to right. In the right sidebar, the info tab explains what the nodes do and their input and outputs; and the debug tab displays data from the debug node. Finally, the red Deploy button in the top right deploys any unsaved nodes and flows.



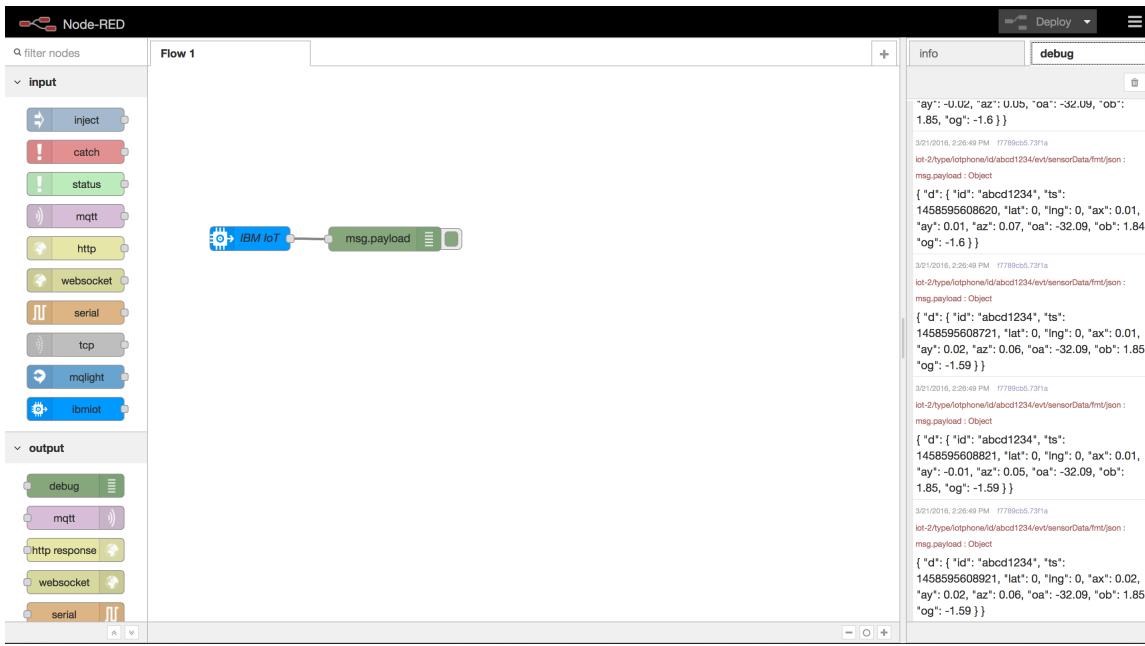
9. Let's create a simple flow that listens to the events from our mobile device and displays the contents. Drag a node and a node to the canvas. Double click on the node and customize the settings as shown below. Use the same Device ID as you had on the mobile device.



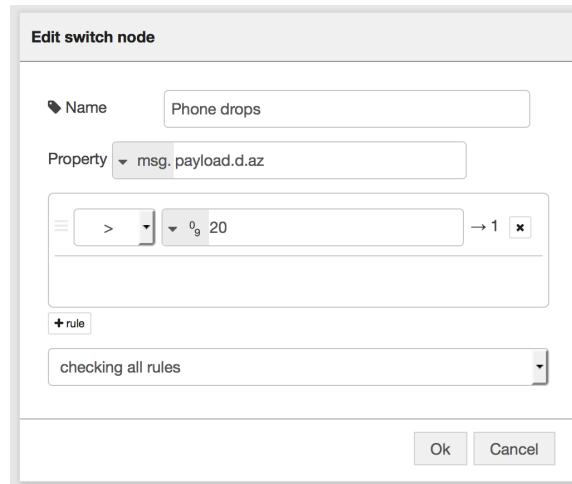
10. The completed flow should look like the following. Click on the  Deploy button in the top-right corner to deploy and save your changes.



11. Click on the debug tab in the right sidebar. The sensor data from the phone should be displayed.

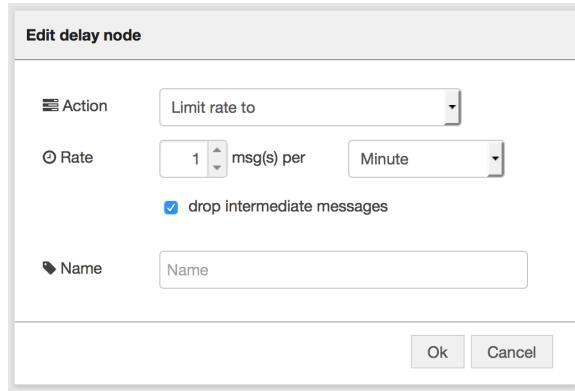


12. Click on the green tab to the right of the debug node in the canvas. This toggles off/on the messages shown in the debug tab. Next, add a  switch statement with the following settings:



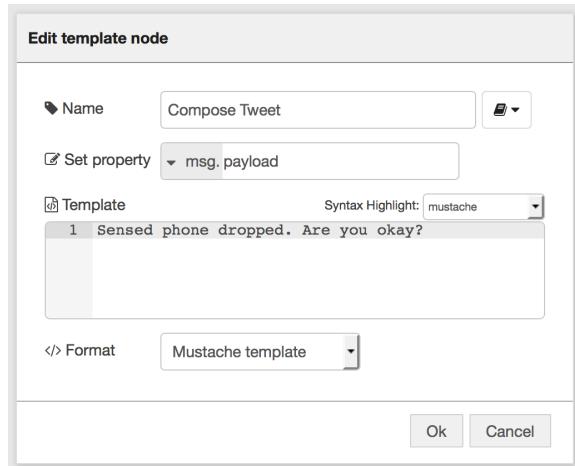
This will filter out any messages where the az value doesn't exceed the value 20. In other words, we only want to proceed with the flow when the value of az exceeds the value 20 (arbitrary value, you can change it if the device is less sensitive).

13. Add a  node with the following settings:



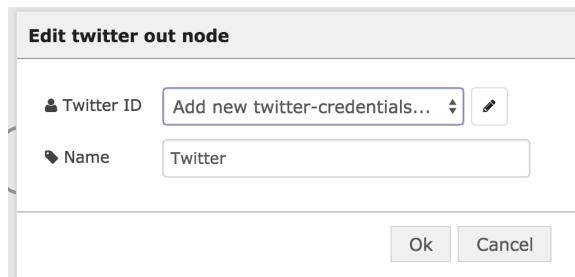
This node limits how often the rest of flow is run. Since the acceleration could exceed this value more than one, without this node, a tweet could be sent several times. With this node, we can limit this flow to send a tweet once per minute.

14. Add a  node with the following settings:



This composes a message that we will tweet.

15. Add a  node. Click on the pencil button and authenticate with Twitter. The account you sign in with will be used to send tweets.





## Authorize Node RED to use your account?



Node RED

nodered.org

Node-RED Twitter node

[Authorize app](#)

[Cancel](#)

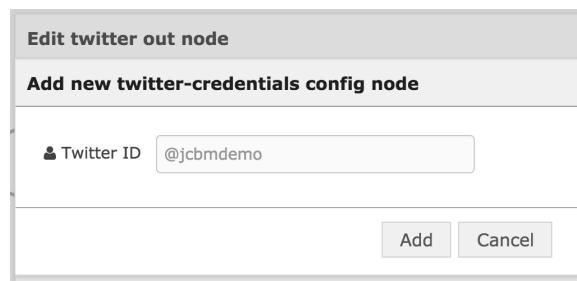
### This application will be able to:

- Read Tweets from your timeline.
- See who you follow, and follow new people.
- Update your profile.
- Post Tweets for you.
- Access your direct messages.

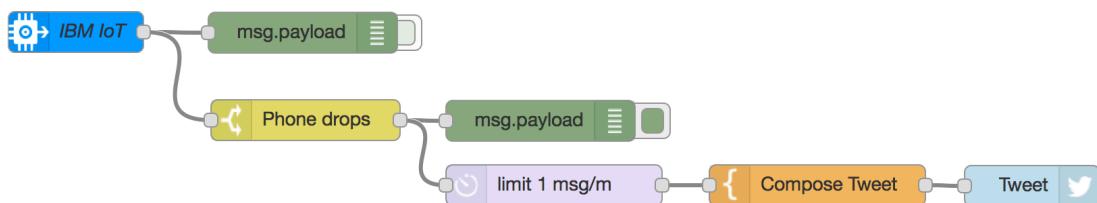
### Will not be able to:

- See your Twitter password.

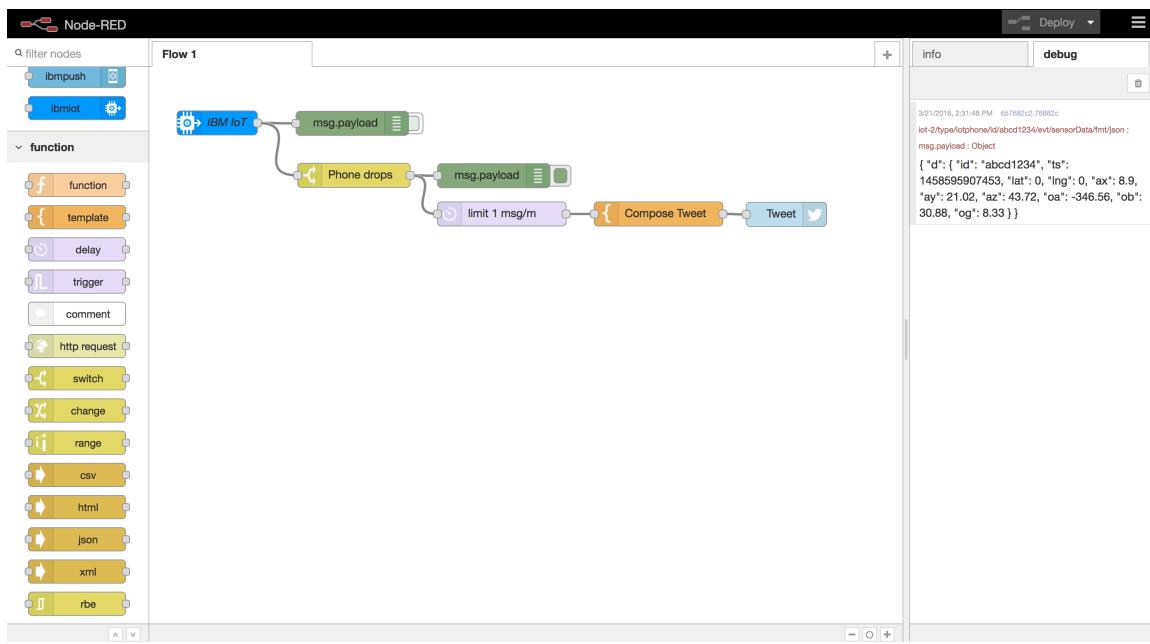
Return back to the node configuration. The settings for the Twitter node should have your username set. Click Add.



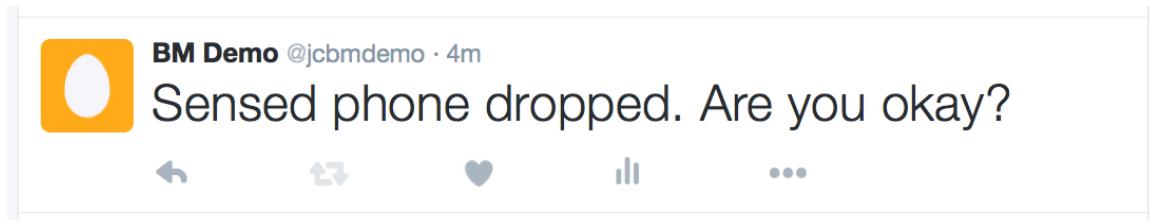
16. Connect the nodes together as follows:



17. Click  Deploy to deploy and save the changes. Give the phone a quick jolt. The debug tab should display the sensor data.



18. Check the Twitter timeline for the tweet.



This lab demonstrates a simple Node-RED application that tweets a message when the mobile device receives a sudden jolt. However, this proof of concept can be extended.

You can change the message to send a Direct Message to a specific Twitter user, say a friend or family member (Hint: read the documentation of the  node to find out how to send a Direct Message). You can also listen for a response from the user (Hint: a  node might help) and if the person has fallen and doesn't respond in a timely fashion, alert someone nearby to check on the owner of the mobile device.

What else you could measure based on the orientation, the movement, or the acceleration of the mobile device?

A digital copy of the completed flow JSON can be found at:  
<http://ibm.biz/lab-mobile-accelerometer-sensor>