

Práctica BM009 – Insights for Weather

Noviembre 2015



El servicio Insights for Weather en Bluemix permite consultar la información producida por los cientos de miles de sensores desplegados alrededor del mundo para conocer las condiciones climatológicas actuales y predecir sus cambios en el futuro inmediato.

En esta práctica utilizarás este servicio para crear una aplicación que despliegue las condiciones actuales del sitio donde te encuentras.

En la siguiente URL puedes ver un ejemplo:

<http://clima.mybluemix.net>

Prerrequisitos

1. Cuenta de Bluemix.
2. Conocimientos básicos de Node.js

Preparando el ambiente



1. En el catálogo de servicios, selecciona el runtime SDK for Node.js para crear una nueva aplicación y **asigne un nombre único** en el dominio mybluemix.net.

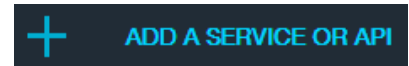
2. Una vez aprovisionada podrás acceder tu nueva aplicación en la URL generada:

<http://<myapp>.mybluemix.net>



Por ahora sólo verás la página por defecto.

3. En la sección **visión general** de la aplicación, dentro del portal de Bluemix, puedes asociar servicios con tu aplicación haciendo click en ADD SERVICE OR API.



4. Selecciona el servicio **Insights for Weather**. Cuando se crea el servicio Bluemix te solicitará reiniciar la aplicación para actualizar las variables de ambiente.

La instancia del servicio que ha sido creada incluye las credenciales requeridas para accederlo desde la aplicación. Vas a necesitar estos datos para desarrollar las llamadas.

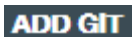
Estos datos están disponibles como variables de ambiente en el contexto de ejecución de tu aplicación.

```
{
  "weatherinsights": [
    {
      "name": "Insights for Weather-mt",
      "label": "weatherinsights",
      "plan": "Free",
      "credentials": {
        "username": "6854d1e9-2cb3-4122...",
        "password": "1L4NHNH...",
        "host": "twcservice.mybluemix.net",
        "port": 443,
        "url": "https://6854d1e9..."
      }
    }
  ]
}
```

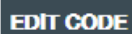


Editando el código

Ve a la sección **visión general** de tu aplicación.



1. Tienes diferentes opciones para editar el código. En esta práctica lo haremos mediante un repositorio GIT, para esto necesitas crearlo dando click en el botón **ADD GIT** de la esquina superior derecha.



2. Una vez creado el repositorio se habilitará el botón **EDIT CODE** por medio del cual podrás acceder al editor en línea de IBM Bluemix DevOps Services. Al dar click el editor se mostrará en una nueva pestaña del navegador.



3. En la nueva pestaña habilita la opción **Live Edit** que encontrarás en la parte superior, de esta manera los cambios que realices en el código se reflejen en la aplicación sin necesidad de tener que desplegarla nuevamente.



4. Selecciona el archivo que quieras modificar y realiza los cambios que necesites. Para que tus cambios se reflejen debes dar click en el botón “reiniciar la aplicación”. A diferencia del despliegue, esta acción no modifica el ambiente aplicativo así que resulta más ágil en tiempo de desarrollo.

app.js

La aplicación recién creada utiliza el framework express y utiliza el documento **app.js** para su ejecución. Si no estás familiarizado con express puedes obtener más información en: <http://expressjs.com>

Selecciona el archivo app.js del directorio raíz para editarlo. Vamos a crear un servicio que obtenga la información del clima y pueda ser llamado desde una página HTML.

Agrega las siguientes dependencias:

```
var https = require('https');
var cfenv = require('cfenv');
var url = require('url');
```

Utilizando el módulo **cfenv** puedes acceder las variables de ambiente de tu aplicación en Bluemix y así obtener las credenciales del servicio Insights for Weather que asociaste:

```
var appEnv = cfenv.getAppEnv();
var vcap_services = JSON.parse(process.env.VCAP_SERVICES);
var wCredentials = vcap_services.weatherinsights[0].credentials;
```

Crea una ruta para el servicio que enviará la información hacia la página HTML

```
app.get('/weather',function(req,res) {
// aquí va el nuevo código
});
```



Ahora invocaremos el servicio Insights for Weather mediante una llamada REST a su API de condiciones actuales:

twc_observations_current

Puedes consultar la descripción completa de las APIs en la siguiente dirección:

<https://twcservice.mybluemix.net/rest-api/>

El siguiente código crea la consulta utilizando las credenciales obtenidas de las variables de ambiente y la posición geográfica para la cual se consultarán las condiciones climatológicas. Se espera que esta posición sea especificada como un parámetro al acceder esta ruta.

Colócalo dentro de la función recién creada.

```
var queryStr=url.parse(req.url,true).query;
var optionsgetmsg = {
  host : wCredentials.host,
  port : 443,
  path : '/api/weather/v2/observations/current?units=m&language=es-MX&geocode='+queryStr.latlon,
  method : 'GET',
  headers : {
    "Authorization" : "Basic " + new Buffer(wCredentials.username + ":" +
wCredentials.password).toString("base64")
  }
};
```

Posteriormente realizamos la llamada REST y la respuesta completa es enviada como respuesta de esta ruta en formato JSON. De esta forma podremos invocarlo como una llamada REST desde la página HTML. Incluye este código también dentro de la función.

```
var reqGet = https.request(optionsgetmsg, function(res1) {
  var cond="";
  res1.on('data', function(d) {
    cond+=d;
  });
  res1.on('end', function() {
    res.json(JSON.parse(cond));
  });
});
reqGet.end();
reqGet.on('error', function(e) {
  console.error(e);
});
```

Página HTML

En app.js se ha declarado el directorio **public** como accesible de forma estática, lo que quiere decir que su contenido podrá ser accedido desde un browser directamente.

Crea un nuevo archivo en este directorio seleccionando File > New > File en el menú del editor. Nómbralo **mapa.html** y selecciónalo para su edición. Esta será la página que desplegará la información.

Agrega la siguiente librería para consultar la información de localización del dispositivo cliente

```
<script src="http://maps.google.com/maps/api/js?sensor=false"></script>
```

y la siguiente para poder realizar llamadas al servidor que hemos creado

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>
```



El siguiente fragmento creará los elementos HTML para desplegar la información.

```
Lugar: <span id="lugar">buscando...</span><br>
<span id="currentConds"></span>
<img id="icon" width="50" height="50" style="{visibility:hidden}">
```

Y este script utiliza la función `navigator.geolocation.getCurrentPosition` para obtener la posición geográfica del navegador y usarla como parámetro para invocar la ruta que creamos en el servidor mediante una llamada REST. Con la información obtenida se modifica el contenido de los elementos HTML de la página, desplegando la información esperada.

```
<script>

function getWeatherInfo(position) {
    var s = document.querySelector('#lugar');
    s.innerHTML = position.coords.latitude+","+position.coords.longitude;

    jQuery.ajax({
        type: "GET",
        url: "weather?latlon="+position.coords.latitude+","+position.coords.longitude,
        contentType: "application/json; charset=utf-8",
        dataType: "json",
        success: function (data) {
            var c=document.querySelector('#currentConds');
            c.innerHTML = data.observation.metric.temp+"°C "
                        + data.observation.sky_cover;
            var icon=document.querySelector('#icon');
            icon.setAttribute("src", "http://clima.mybluemix.net/images/icon" +
data.observation.icon_code + ".png");
        },
        error: function (jqXHR, status1) {
            document.querySelector('#currentConds').innerHTML = "ERROR "+status1;
        }
    });
}

navigator.geolocation.getCurrentPosition(getWeatherInfo, function ()
{document.querySelector('#lugar').innerHTML = "ERROR";});

</script>
```

Da click en el botón de reinicio de la aplicación.



Ahora al acceder la página <http://<myapp>.mybluemix.net/mapa.html> podrás ver información básica del clima en tu localidad.



¡Es tu turno!

La mejor forma de comprobar lo que has aprendido es agregando funcionalidad a la aplicación por ti mismo.

Al consultar el API del servicio Insights for Weather te darás cuenta que provee mucha más información de la que hemos usado, incluyendo índices UV, presión barométrica, la dirección y velocidad del viento, etc. Además es posible consultar predicciones por hora o por día y series históricas.

¿Qué funcionalidad podrías agregar a esta aplicación?

¿Para qué podrías utilizar esta información en otras aplicaciones?

