PRÁCTICA 1.

PROGRAMACIÓN DE DISPOSITIVOS MÓVILES

La finalidad de esta práctica es introducirse en la programación de dispositivos móviles (teléfonos, tabletas, etc.). En concreto, se trabajará sobre el framework de desarrollo de aplicaciones multiplataforma, Apache Cordova. Este framework como veremos, nos permitirá desarrollar una misma aplicación y que esta, puede ser utilizado en las diferentes plataformas disponibles en el mercado (Android, iOS, Windows Phone,…). Se utilizará el entorno de desarrollo Visual Studio 2017 y las herramientas específicas que nos facilitaran el trabajo de creación, pruebas y despliegue desde Visual Studio 2017, mediante Apache Cordova.

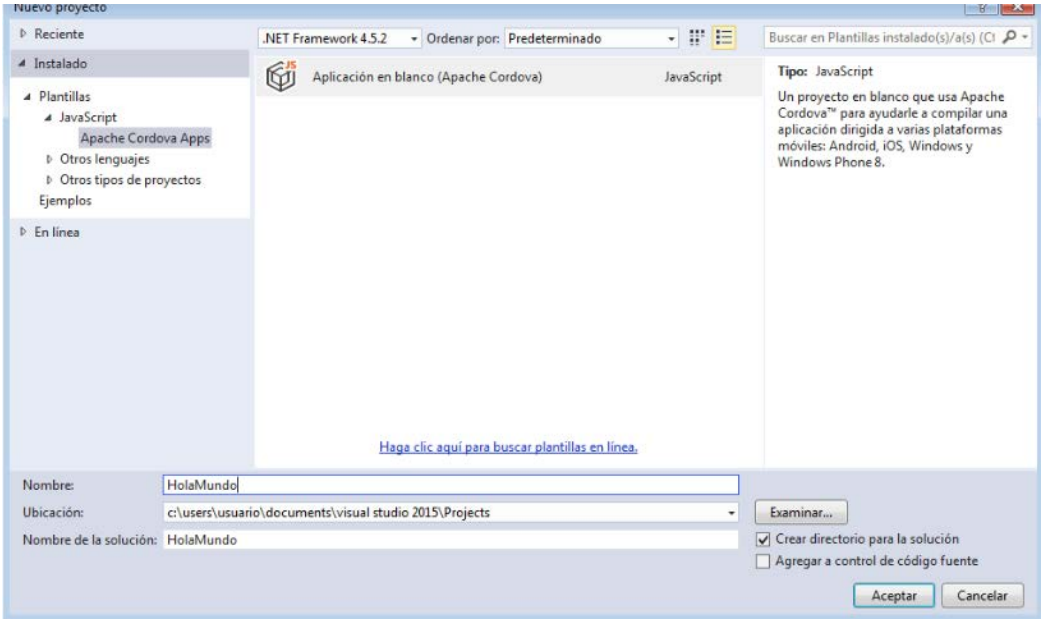
El objetivo de la práctica es familiarizarse con Apache Cordova mediante el entorno de desarrollo Visual Studio 2017. Para ello primero se realizarán unos ejercicios básicos, para comprobar el funcionamiento del entorno y posteriormente se propone un ejercicio, con una parte básica y varios puntos adicionales.

Para la realización de esta, utilizaremos Visual Studio 2017 con las Visual Studio Tools para Apache Cordova. A continuación, se plantearán una serie de ejercicios guiados, para conocer el entorno y realizar unos ejercicios básicos de creación de aplicaciones.

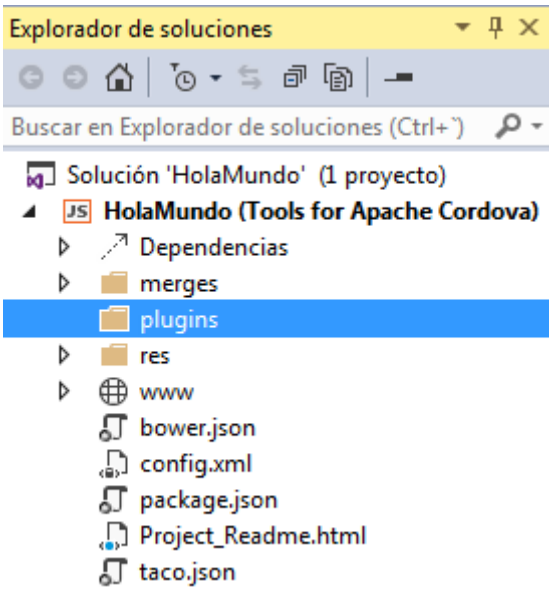
Ejemplo 1. Apache Cordova

Para la creación de la primera aplicación "Hola Mundo" vamos a realizar los siguientes pasos:

- Abra Visual Studio. En la barra de menús, elija **Archivo**, **Nuevo**, **Proyecto**.
- En el cuadro de diálogo **Nuevo proyecto**, en **Plantillas**, elija **JavaScript**, **Aplicaciones Apache Cordova** y luego seleccione la plantilla **Aplicación vacía**.
- Elija **Buscar** para encontrar una ubicación para el proyecto.
- Asigne un nombre a la aplicación y elija **Aceptar**.



- Visual Studio crea el proyecto y abre el Explorador de soluciones en el panel derecho.



El nuevo proyecto Cordova incluye cuatro carpetas de nivel superior:

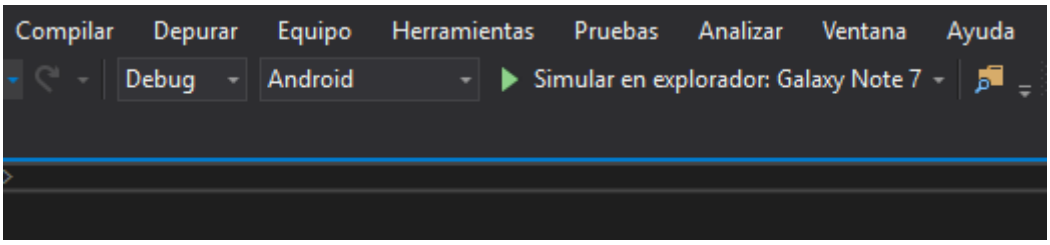
- **merges** se usa para agregar código específico de la plataforma.
- **plugins** se usa para los complementos de Apache Cordova que proporcionan acceso a características de dispositivos nativos.
- **res** se utiliza para activos visuales específicos de la plataforma (iconos y pantallas de presentación), certificados de firma y, si es necesario, archivos de configuración específicos de la plataforma
- **www** es la carpeta que se usa para el código de la aplicación.
  La carpeta **www** contiene otras carpetas:
  - **css** contiene hojas de estilo CSS básicas que se incluyen con la plantilla vacía.
  - **images** es la ubicación sugerida para las imágenes de la aplicación.
  - **scripts** es la ubicación predeterminada de todos los archivos JavaScript o TypeScript.

Además de los archivos CSS y JavaScript, el nuevo proyecto también incluye varios archivos:

- **config.xml**, que contiene opciones de configuración de la aplicación. Puede abrir este archivo desde el Explorador de soluciones en el diseñador de configuración (que es una interfaz de config.xml) o lo puede editar directamente seleccionando **Ver código** en el menú contextual del archivo.
- **taco.json** almacena los metadatos del proyecto que permiten a Visual Studio compilar en sistemas operativos que no son de Windows, como un Mac.
- **www\index.html** es la pantalla principal predeterminada de la aplicación.
- **Project_Readme.html** contiene vínculos a información útil.

Para compilar y ejecutar la aplicación:

1. Elija Android, iOS u otra plataforma, desde la lista **Plataformas de solución**.



2. Seleccione el dispositivo donde emular de la lista derecha.
3. Presione F5 para iniciar la depuración o Mayús+F5 para iniciar sin depurar.
4. Una vez comprobado el correcto funcionamiento, vamos a modificar la aplicación para que muestre el nombre de la asignatura y un *"Hola Mundo"*, tal cual podemos ver a continuación:

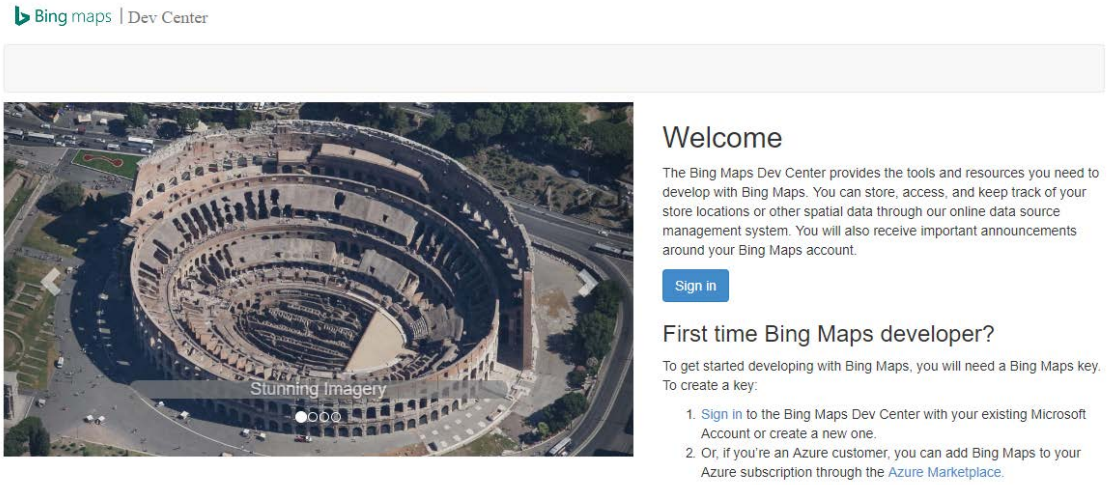

localhost:4400/index.html
APACHE CORDOVA
EL DISPOSITIVO ESTÁ PREPARADO
DISM 2018-2019
HOLA MUNDO

Ejemplo 2. Apache Cordova, Bing Maps

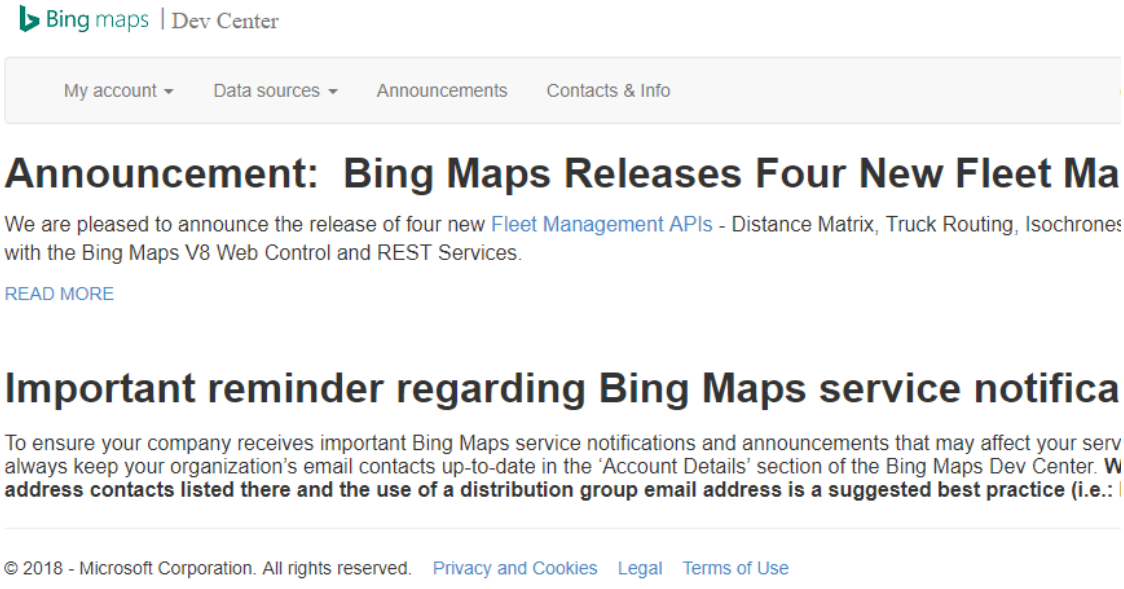
En este ejemplo vamos a trabajar con mapas de Bing (Microsoft). Para poder trabajar con mapas de bing, necesitamos obtener una API key, para ello necesitaremos disponer de una cuenta de Microsoft.

Una vez tengamos registrada la cuenta de Microsoft (Outlook, hotmail,…), accederemos al portal de bing maps:

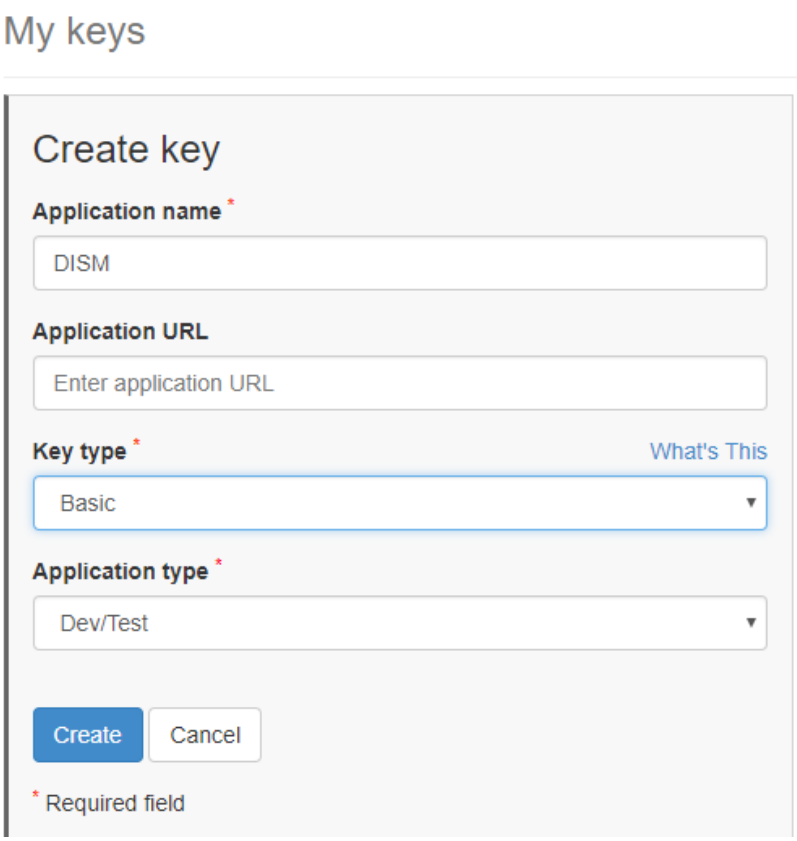
https://www.bingmapsportal.com



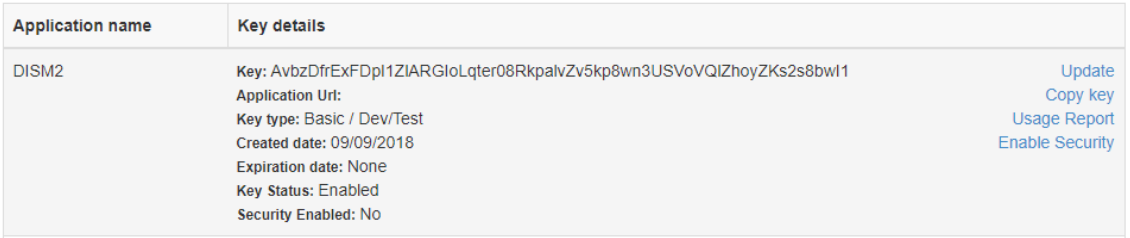
Pulsamos en ***sign in*** e introducimos nuestros datos. Una vez validados nuestros datos, accederemos a la siguiente consola:



Pulsamos en ***My account → My Keys*** y creamos una nueva key asociada a una aplicación:



Una vez se cree la key, veremos los detalles de esta.



La **key** obtenida, será necesaria para poder trabajar con los mapas de bing, como veremos a continuación.

El siguiente paso que vamos a realizar será crear una aplicación que utiliza el servicio de **bing maps** para mostrar un mapa y mostrar un marcador con nuestra ubicación. Para ello utilizaremos el siguiente código en el fichero index.html:

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <meta charset="utf-8" />
    <script type="text/javascript" src="cordova.js"></script>
    <!-- El atributo "async" es necesario para que RequireJS funcione en una aplicación de
Windows. -->
    <script type="text/javascript" data-main="scripts/startup" src="lib/require.2.1.8.js"
async></script>
    <script type='text/javascript'
            src='http://www.bing.com/api/maps/mapcontrol?callback=GetMap'
            async defer></script>
    <script type='text/javascript'>
        function GetMap() {
            var map = new Microsoft.Maps.Map('#myMap', {
                credentials: 'api key bing maps'
            });

            //Request the user's location
            navigator.geolocation.getCurrentPosition(function (position) {
                var loc = new Microsoft.Maps.Location(
                    position.coords.latitude,
                    position.coords.longitude);

                //Add a pushpin at the user's location.
                var pin = new Microsoft.Maps.Pushpin(loc);
                map.entities.push(pin);

                //Center the map on the user's location.
                map.setView({ center: loc, zoom: 15 });
            });
        }
    </script>
</head>
<body>
    <div id="myMap" style="position:relative;width:600px;height:400px;"></div>
</body>

</html>
```

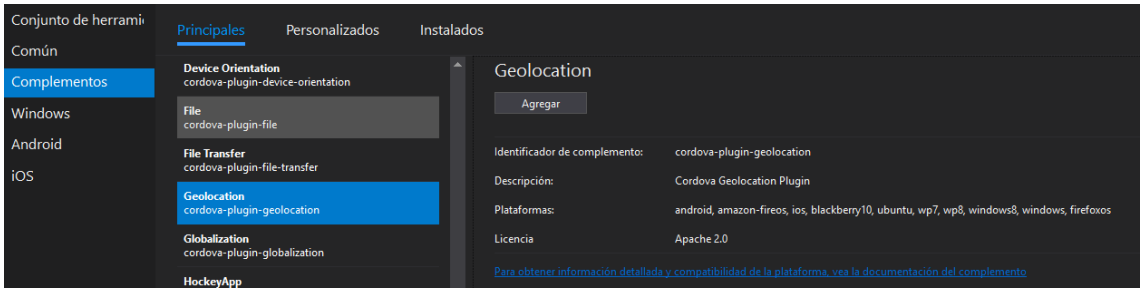
Importante introducir en **credentials**, la key obtenida anteriormente.

Fuente:

https://msdn.microsoft.com/en-us/library/mt712801.aspx

Para el correcto funcionamiento como aplicación apache cordova, deberemos de añadir un plugin para geolocalización:

1. Doble Click en el fichero confix.xml
2. Pulsaremos en Complementos→Principales→Geolocalizacion→Agregar



Como último paso a este ejercicio, vamos a añadir al ejemplo anterior un marcador, de la ubicación actual, haciendo uso de la api de bing maps, para ello se puede consultar el siguiente ejemplo:

https://www.bing.com/api/maps/sdk/mapcontrol/isdk/createpushpinfromimage#JS

Si necesitamos cualquier otro dato de la api de bing maps, en el siguiente enlace podemos encontrar toda la documentación:
https://www.bing.com/api/maps/sdk/mapcontrol/isdk/Overview#JS

Ejemplo 3. Uso de framework responsive

En el siguiente ejemplo vamos a ver cómo utilizar un framework responsive, en nuestro caso vamos a hacer uso del framework jquerymobile https://jquerymobile.com/ .

Este framework se caracteriza por ser muy sencillo de usar, empleando las etiquetas de HTML , para cambiar el aspecto de los componentes de la aplicación y darle un aspecto más agradable y responsive a una página HTML.

Vamos a hacer uso de la versión 1.4.5, en el enlace http://demos.jquerymobile.com/1.4.5/ , podemos ver toda la documentación , así como ejemplos de uso.

En el siguiente ejemplo en concreto, vamos a crear una estructura multipágina, similar a la de la página de ejemplo http://demos.jquerymobile.com/1.4.5/pages-multi-page/ .

Enlaces a Librerías Empleadas:

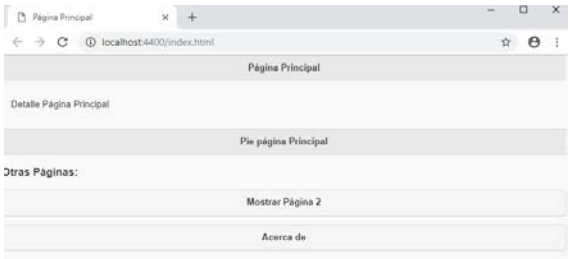
```
    <!-- Apache Cordova-->
    <script type="text/javascript" src="cordova.js"></script>

    <!--JQUERY -->
    <script src="http://code.jquery.com/jquery-1.11.1.min.js"></script>

    <!--JQUERY MOBILE -->
    <link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.css" />

    <script src="http://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
```

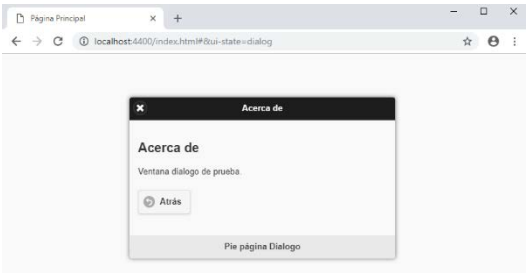
Código Html:

```
<body>
    <div data-role="page">

        <div data-role="header">
            <h1>Página Principal</h1>
        </div>
        <div role="main" class="ui-content">
            <p>Detalle Página Principal</p>
        </div>
        <div data-role="footer">
            <h4>Pie página Principal</h4>
        </div>
        <h3>Otras Páginas:</h3>
        <p><a href="#two" class="ui-btn ui-shadow ui-corner-all">Mostrar Página 2</a></p>
        <p><a href="#popup" class="ui-btn ui-shadow ui-corner-all" data-rel="dialog" data-
transition="pop">Acerca de</a></p>
    </div>
    <div data-role="page" id="two" data-theme="a">
        <div data-role="header">
            <h1>Página 2</h1>
        </div>
        <div role="main" class="ui-content">
            <p>Detalle Página 2</p>
```

```
        </div>
        <p><a href="#one" data-rel="back" class="ui-btn ui-shadow ui-corner-all ui-btn-inline
ui-icon-back ui-btn-icon-left">Atrás</a></p>
        <div data-role="footer">
            <h4>Pie de Página</h4>
        </div>
    </div>
    <div data-role="page" id="popup">
        <div data-role="header" data-theme="b">
            <h1>Acerca de</h1>
        </div>
        <div role="main" class="ui-content">
            <h2>Acerca de</h2>
            <p>Ventana dialogo de prueba.</p>
            <p><a href="#one" data-rel="back" class="ui-btn ui-shadow ui-corner-all ui-btn-
inline ui-icon-back ui-btn-icon-left">Atrás</a></p>
        </div>
        <div data-role="footer">
            <h4>Pie página Dialogo</h4>
        </div>
    </div>

</body>
```

Con esto el resultado final será una aplicación multipágina como esta:





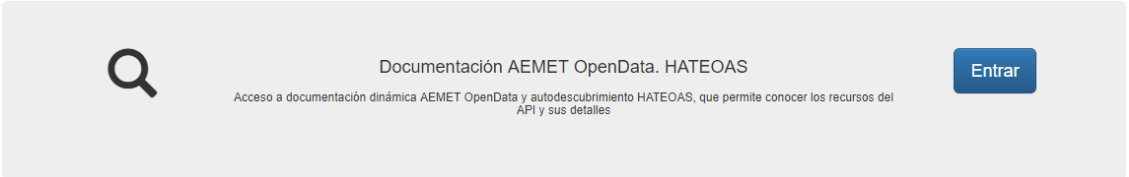
Ejemplo 4. Empleo de datos abiertos Aemet

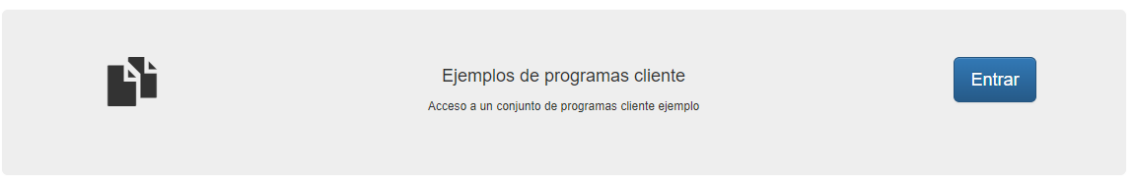
A continuación, vamos a crear una aplicación que haga uso de datos abiertos de una api, en concreto vamos a utilizar los de Aemet:
https://opendata.aemet.es

Al igual que con Bing Maps, necesitamos una key para poder trabajador con los datos abiertos de aemet. Para ello dentro del enlace anterior, pulsaremos en **Obtención de Api Key**.

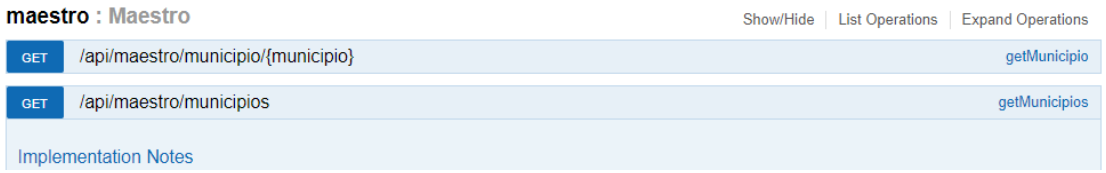
Introducimos un email y seguimos las instrucciones para obtener la key:

Si pulsamos en área desarrolladores, podemos conocer toda la documentación de la api de Aemet y si pulsamos en Ejemplos de programa cliente podremos visualizar una serie de ejemplos para varios lenguajes de programación:

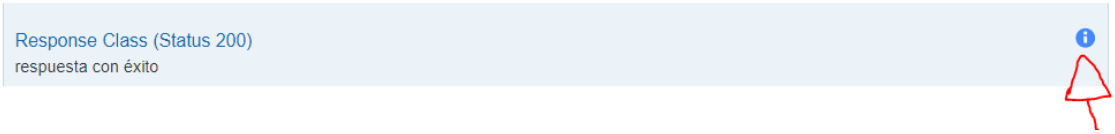




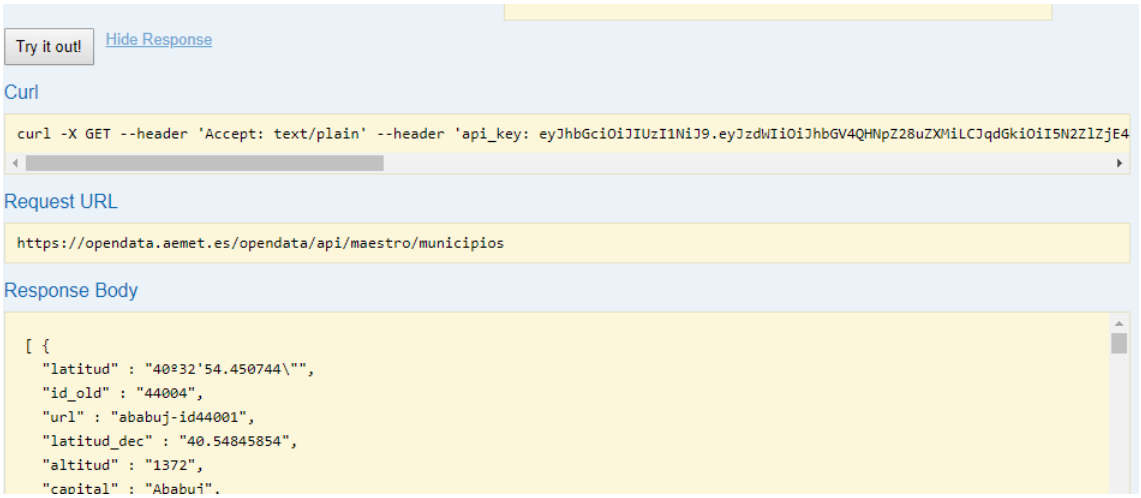
Vamos a realizar una consulta de los municipios de España, los cuales a posterior podremos emplear para determinadas tareas. Para obtener la URL, podemos seguir la documentación, la cual no llevará a la ubicación de los municipios:



Introducimos la api key:



Pulsamos en T**ry it out!** y obtenemos el resultado con la información o una URL donde localizarla:



Una vez que hemos visto como obtener los datos de la API de Aemet, procedemos a realizar el siguiente ejemplo donde mostramos un listado de los municipios en una tabla, mediante la ayuda de los datatables (https://datatables.net/), los cuales nos permiten el mostrar información de tablas de una forma más agradable, que una simple tabla de html.

Enlaces a Librerias Empleadas:

```
     <!-- Apache Cordova-->
    <script type="text/javascript" src="cordova.js"></script>

    <!--JQUERY -->
    <script src="http://code.jquery.com/jquery-1.11.1.min.js"></script>

    <!--JQUERY MOBILE -->
    <link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.css" />
    <script src="http://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>

    <!--JQUERY, Datatabla -->
    <link rel="stylesheet" type="text/css"
href="https://cdn.datatables.net/1.10.19/css/jquery.dataTables.min.css" />

    <script type="text/javascript"

src="https://cdn.datatables.net/1.10.19/js/jquery.dataTables.min.js"></script>
```

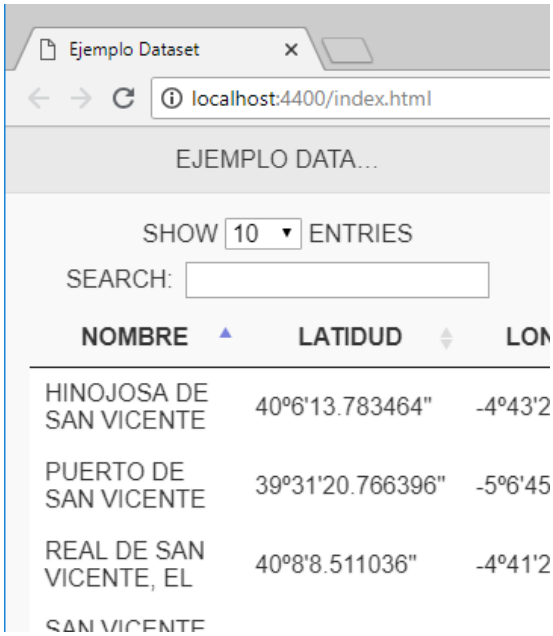
Código Javascript, necesario para el ejemplo:

```
<script>
        var datos;
        var datosfiltrados = [];
        var key = 'key bing maps';
        var settings = {
            "async": true,
            "crossDomain": true,
            "url": "https://opendata.aemet.es/opendata/api/maestro/municipios?api_key=" + key,
            "method": "GET",
            "headers": {
                "cache-control": "no-cache"
            }
        }
        $(document).on("pagecreate", "#index", function (event) {
            InicializarGrid();
        });
        function InicializarGrid() {
            $.ajax(settings).done(function (response) {
                var j = 0;
                //Parseo a objeto para filtrar y meter en datatable
                datos = JSON.parse(response);
                //Filtro los municipios que contengan "san vicente"
                datos.forEach(function (entry) {
                    if (entry.nombre.toLowerCase().indexOf("san vicente") > -1) {
                        datosfiltrados[j] = entry;
                        j = j + 1;
                    }
                });
                tabla = $('#dataGrid').DataTable({
                    "data": datosfiltrados,
                    "columns": [
                        {
                            "data": "nombre"
                        },
                        {
                            "data": "latitud"
                        },
                        {
                            "data": "longitud"
                        }
                    ]
                });
            });
        }

    </script>
```

Html necesario para el ejemplo:

```
<div data-role="page" id="index">
        <div data-role="header">
            <h1>Ejemplo Dataset</h1>
        </div>
        <div role="main" class="ui-content">
            <table id="dataGrid" class="table table-striped table-bordered" style="width:100%">
                <thead>
                    <tr>
                        <th>Nombre</th>
                        <th>Latidud</th>
                        <th>Longitud</th>
                    </tr>
                </thead>
            </table>
        </div>
        <div data-role="footer">
            <h4>DISM 2018-2019</h4>
        </div>

</div>
```

El resultado final será el siguiente:



Ejemplo 5. Empaquetar, firmar y distribuir aplicaciones

En este ejemplo, vamos a ver como empaquetar, firmas y distribuir aplicaciones creadas mediante apache cordova, para Android. Estos pasos son extrapolables a apps Windows y apps los.

El primer paso que vamos a ver, es el de firmar una aplicación, si bien se puede distribuir sin firmar, es recomendable realizar su firma, si queremos poder subirla a la tienda de google play por ejemplo. Para ello es necesario tener Java JDK instalado, podemos comprobarlo, abriendo una ventana de CMD y escribiendo `java –version` y nos indicará en caso afirmativo la versión instalada.

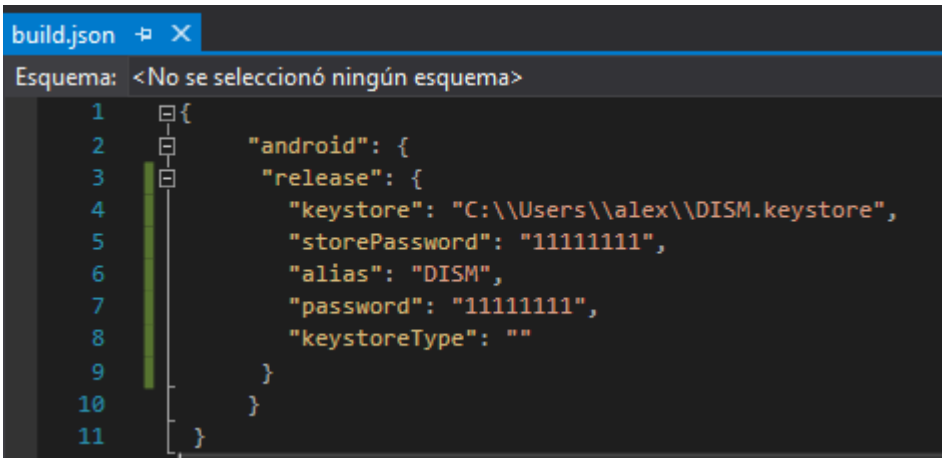
1. Abrimos una consola de CMD con permisos de administrador.
2. Debemos verificar que tenemos la variable `%JAVA_HOME%` correctamente definida, en caso de no tenerla deberemos de añadir al comando siguiente la ruta, hacia nuestra Java instalado en el sistema, Ej.: `C:\Program Files\Java\jdk1.8.0_111\bin`
3. A continuación, ejecutamos la siguiente línea en CMD:

   ```
   keytool -genkeypair -v -keystore dism.keystore -alias DISM -keyalg RSA -keysize 2048 -validity 10000
   ```

Si seguimos el asistente obtendremos el fichero con las keys, en la ubicación indicada por nosotros:

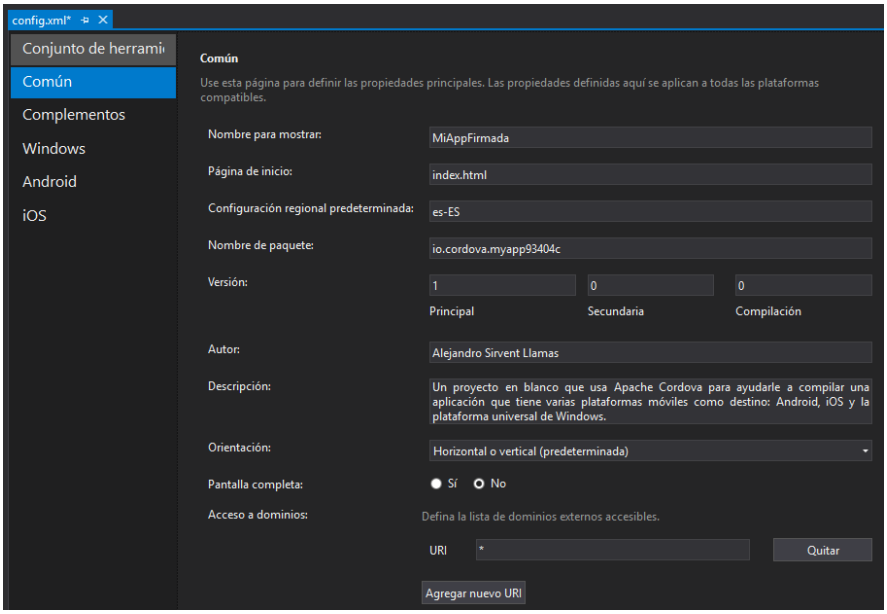
```
C:\Users\alex>keytool -genkeypair -v -keystore DISM.keystore -alias DISM -keyalg RSA -keysize 2048 -validity 10000
Introduzca la contraseña del almacén de claves:
Volver a escribir la contraseña nueva:
¿Cuáles son su nombre y su apellido?
  [Unknown]:  Alejandro Sirvent Llamas
¿Cuál es el nombre de su unidad de organización?
  [Unknown]:  dtic
¿Cuál es el nombre de su organización?
  [Unknown]:  ua
¿Cuál es el nombre de su ciudad o localidad?
  [Unknown]:  San Vicente
¿Cuál es el nombre de su estado o provincia?
  [Unknown]:  Alicante
¿Cuál es el código de país de dos letras de la unidad?
  [Unknown]:  es
¿Es correcto CN=Alejandro Sirvent Llamas, OU=dtic, O=ua, L=San Vicente, ST=Alicante, C=es?
  [no]:  s

Generando par de claves RSA de 2.048 bits para certificado autofirmado (SHA256withRSA) con una validez de 10.000 días
        para: CN=Alejandro Sirvent Llamas, OU=dtic, O=ua, L=San Vicente, ST=Alicante, C=es
Introduzca la contraseña de clave para <DISM>
        (INTRO si es la misma contraseña que la del almacén de claves):
[Almacenando DISM.keystore]

Warning:
El almacén de claves JKS utiliza un formato propietario. Se recomienda migrar a PKCS12, que es un formato estándar del s
ector que utiliza "keytool -importkeystore -srckeystore DISM.keystore -destkeystore DISM.keystore -deststoretype pkcs12"
.
```

Ahora deberemos de ir a Visual Studio , nuestro proyecto y abrir el fichero build.json y cambiarlo por:

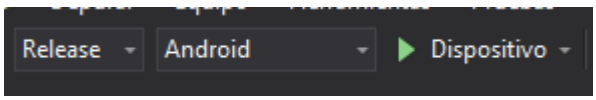


Ya tenemos firmada la aplicación, ahora debemos de empaquetar la aplicación y generar el apk.

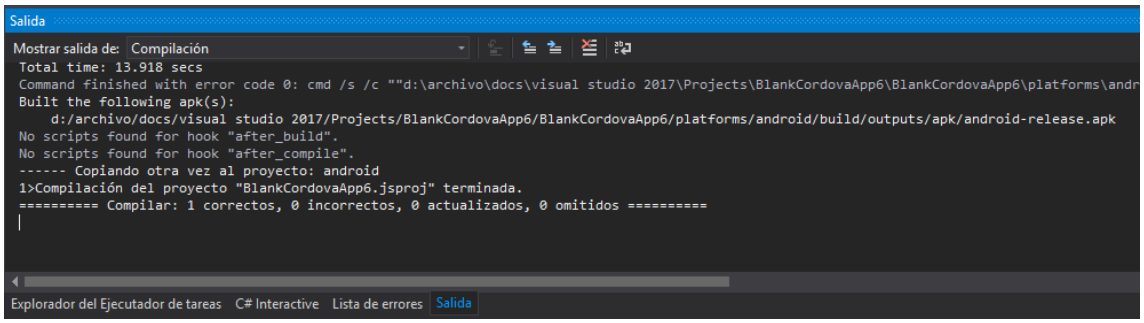
1. Establecemos los valores de la aplicación, haciendo doble click en config.html, dentro de nuestro proyecto de visual studio:



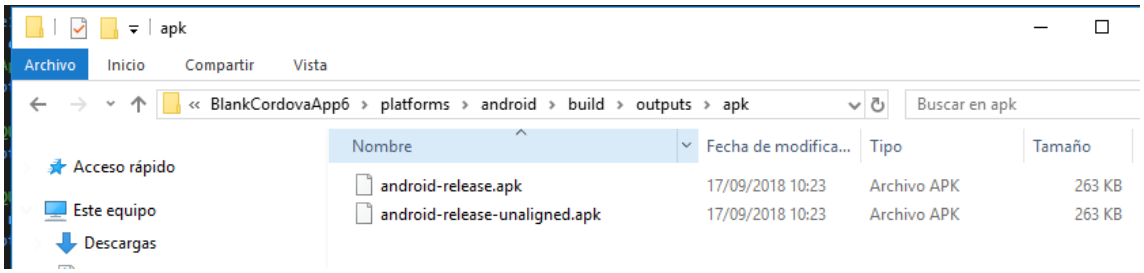
2. A continuación, procedemos a generar el APK, estableciendo las siguientes opciones de compilación y pulsando F6 (Compilar→Compilar Solución):



3. Si todo ha ido correctamente, veremos en Visual Studio →salida la siguiente información:



Donde podemos ver la ubicación del APK:



Én el siguiente enlace podemos ver todo el proceso detallado, para todas las plataformas (Windows,Ios,Android):

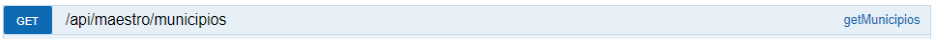
https://docs.microsoft.com/en-us/visualstudio/cross-platform/tools-for-cordova/publishing/publish-to-a-store?view=toolsforcordova-2017

Enunciado Práctica

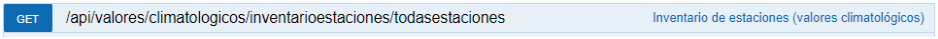
Haciendo uso de los ejemplos guiados anteriores, vamos a proceder a crear una aplicación multipágina, donde hagamos uso de la api de aemet y los mapas de bing para mostrar información meteorológica, mediante un framework responsive tipo jquerymobile.

Crear una aplicación que nos muestre la siguiente estructura de páginas.

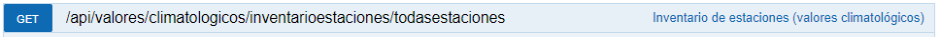
- Página principal.
  - Página Municipios España con más de 50000 habitantes.
    - Tabla con Nombre, Latitud, Longitud y Número de Habitantes.



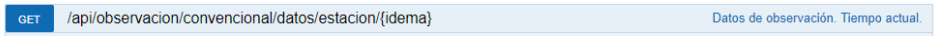
  - Página Estaciones Meteorológicas España.
    - Tabla con Nombre, Latitud, Longitud e Idema.
    - Optativo: Mostrar un botón por cada estación, que al pulsarlo muestre mapa con un marcador (nombre y provincia) con esa estación en una página de diálogo.



  - Mapa Estaciones Meteorológicas España. (Indicando en el marcador, el nombre y la altitud)



  - Datos de observación horarios de las últimas 24 horas de la estación meterológica indicada.



    - Tabla con Idema, Ubicación, fecha-hora, temperatura.
  - Botón Ventana Créditos, con los datos del Alumno (Nombre, DNI, Email, Foto), se mostrará como un dialogo (página emergente).

Se deberá Incluir un botón que nos llevé a la ventana principal desde todas las páginas.

Se valorará:

- El uso de otro framework responsive (bootstrap, ionic, etc…)
- Despliegue de la aplicación en dispositivo móvil.
- Escritura de “código fuente legible”, modularizando lo máximo posible y estructurándolo en las diferentes carpetas existentes en el proyecto.

- Se valorará el trabajo diario del alumno, valorando su trabajo en clase, así como la capacidad de resolución de los problemas que se presentan en la práctica y no sean resueltos por el profesor.

9. Entrega y Corrección

- Se debe entregar, por el Campus Virtual, mediante una entrega de práctica, el Proyecto Visual Studio 2017 creado, comprimido en ZIP, antes del 17-10-2018.
- Se realizará una demo con la aplicación desarrollada al profesor en el laboratorio, el 17-10-2018 y el 18-10-2018, en los turnos asignados al alumno, contestando correctamente a las preguntas realizadas por el profesor.
- La práctica es de carácter individual.
- Si se constata que el código desarrollado no es original (por tanto, es una copia), la práctica se calificará con un 0.