# Flight Status sample

This sample demonstrates the use of a secure, cloud-based, commercial RESTful service providing data to an app.

## Description of the sample

This sample displays the status and maps the route of a commercial airline flight.

## Prerequisites

This sample requires the following;

A SharePoint 2013 development environment.

Visual Studio 2012 and Office 2013 Tools for Visual Studio 2012.

A FlightStats developer account and key obtained from <https://developer.flightstats.com>

A Bing Maps developer account and key obtained from <http://www.bingmapsportal.com>

## Key components of the sample

The sample consists of a SharePoint-hosted app that calls to the FlightStats API. After retrieving flight information, the route is plotted on a map using the Bing Maps service.

## Configure the sample

Obtain an application ID and an application Key from <https://developer.flightstats.com> and enter the information into the wingtip.flightstatus.js library in the line shown below:

var appId = "", //Obtained from https://developer.flightstats.com

appKey = "", //Obtained from https://developer.flightstats.com

Obtain a developer key from <http://www.bingmapsportal.com> and enter the information into the App.js library in the line as shown below:

var mapsKey = ""; //Obtained from http://www.bingmapsportal.com

## Run and test the sample

1. Open the FlightStatus.sln solution in Visual Studio 2012.
2. Edit the deployment URL to point to a site in your development environment.
3. Press F5.
4. When the app appears, it will load the airport codes.
5. When ready, select an airport, carrier, and enter a flight number and date.
6. Click the “Get Status” button.

## Troubleshooting

You must have a Bing Maps developer key properly entered into the app.js library. You must have a FlightStats application ID and application Key properly entered into the wingtip.flightstatus.js library.

## Change log

First release June 18, 2013

## Related content

SharePoint 2013 provider-hosted apps