AppDynamics Phonegap Plugin

Introduction

This plugin wraps the AppDynamics SDK for both Android and iOS. This provides javascript functions to call the native SDK.

Installation

The plugin is currently not available via the public plugin repository. The plugin is installed in the usual way.

```
phonegap plugin add <path/to/plugin>
```

As well as installing the plugin the AppDynamics SDK will need to be installed as normal under the platforms of your Phonegap project.

Android

For Android see https://docs.appdynamics.com/display/PRO41/Mobile+Real+User+Monitoring

If you are not using the AppDynamics cloud collector you should use this alternative initialisation code:

```
Instrumentation.start("YOUR KEY", getApplicationContext(), "http://
collector.mycompany.con:7001/", true);
```

Substitute your values in the example above. The final "true" switch is to enable logging; useful for debugging. You can see the logging output with:

```
adb logcat -s chromium,appdplugin,ADInstrumentation
```

Adding the console logging plugin will enable javascript console.log() messages to appear in the logs under the 'chromium' tag.

```
phonegap plugin install cordova-plugin-console
```

iOS

For iOS see https://docs.appdynamics.com/display/PRO41/Mobile+Real+User+Monitoring

If you are not using the AppDynamics cloud collector you should use this alternative initialisation code:

```
[ADEumInstrumentation initWithKey:@"YOUR KEY" collectorUrl:@"http://
collector.mycompany.com:7001" enableLogging:true];
```

Substitute your values in the example above. The final "true" switch is to enable logging; useful for debugging.

Usage

The AppDynamics SDK calls are available via the normal cordova.exec() call from javascript.

```
var win = function(d) {
     console.log('Success');
};
var fail = function(e) {
```

```
      console.log('Error ' + e);
};
cordova.exec(win, fail, 'AppDynamics', 'reportMetricWithName', ['mycount', 42]);
```

reportMetricWithName

This function allows you to send name integer pairs to the collector, these will show up under "Custom Metrics" in the AppDynamics dashboard.

leaveBreadcrumb

This function allows you to record the user's progress through the application. The breadcrumb trail will be included in any crash reports.

startTimerWithName / stopTimerWithName

This function allows you to time any arbitrary block of code. Remember to call the matching stopTimerWithName.

beginCall / endCall

This function allows you to time and count executions of a function. The call to beginCall returns in its callback a unique key which must be used to terminate the matching endCall.

```
var appdkey;
var entrycb = function(d) {
      appdkey = d;
};
var endcb = function(d) {
      console.log('Success');
};
var fail = function(e) {
      console.log('Error ' + e);
};
cordova.exec(entrycb, fail, 'AppDynamics', 'beginCall', ['foo', 'bar']);
// some processing
cordova.exec(endcb, fail, 'AppDynamics', 'endCall', [appdkey]);
```

beginHttpRequest / reportDone / getCorrelationHeaders

This function allows you to time network requests (AJAX) calls to the server side. This will track the response time, execution count and status of the requests.

To have full correlation with a server side application also monitored by an AppDynamics application agent a couple of HTTP request headers must be set before making the call.

```
ADRUM = isAjax:true
ADRUM_1 = isMobile:true
```

If these headers are not set in the request then correlation from the mobile application to the server side will not happen. There is a convenience function available to generate these headers, a JSON object is returned.

```
var appdkey;
var appdheaders;
var requestcb = function(d) { appdkey = d; };
```

```
var headercb = function(d) { appdheaders = d; };
var win = function(d) { sonsole.log('Success'); };
var fail = function(e) { console.log('Error ' + e); };
// get headers
cordova.exec(headercb, fail, 'AppDynamics', 'getCorrelationHeaders', []);
cordova.exec(requestcb, fail, 'AppDynamics', 'beginHttpRequest', [requestURL]);
// make the call
$.ajax({
     url: requestURL,
     headers: appdheaders,
     …
     …
});
// notify call complete
cordova.exec(win, fail, 'AppDynamics', 'reportDone', [appdkey, responseCode,
returnedHeaders]);
```