Handling SMS Notifications in Cordova

Overview

SMS notifications are a sub-set of Push Notification, as such make sure to first go through the Push notifications in Cordova (../../) tutorials.
SMS notifications in Cordova applications are supported for iOS and Android.

Prerequisites:

- Make sure you have read the following tutorials:
  - Notifications Overview (../../)
  - Setting up your MobileFirst development environment (../../../installation-configuration/#installing-a-development-environment)
  - Adding the MobileFirst Foundation SDK to iOS applications (../../../application-development/sdk/cordova)
- MobileFirst Server to run locally, or a remotely running MobileFirst Server.
- MobileFirst CLI installed on the developer workstation

Jump to:

- Notifications API
- Using a SMS subscribe servlet
- Sample Application

Notifications API

In SMS notifications, when registering the device, a phone number value is passed.

Register Device

Register the device to the push notifications service.

```
MFPPush.registerNotificationsCallback(notificationReceived);

function registerDevice() {
    var phoneNumber = prompt("Enter Your 10 digit phone number");
    if(phoneNumber != null && phoneNumber!="" && /^\d+$/.test(phoneNumber)) {
        var options = {};
        options.phoneNumber = phoneNumber;
        MFPPush.registerDevice(options,
        function(successResponse) {
            alert("Successfully registered");
            enableButtons();
        }, function(failureResponse) {
            alert("Failed to register");
        });
        return true;
    }

    else {
        alert("Failed to register, You have entered invalid number");
    }
}
```

> You can also register a device using the Push Device Registration (POST) REST API (http://www.ibm.com/support/knowledgecenter/en/SSHS8R_8.0.0/com.ibm.worklight.apiref.doc/rest_runtime/r_restapi_push_device_registration_post.html)

Using a SMS subscribe servlet

REST APIs are used to send notifications to the registered devices. All forms of notifications can be sent: tag & broadcast notifications, and authenticated notifications

To send a notification, a request is made using POST to the REST endpoint: `imfpush/v1/apps/<application-identifier>/messages`.
Example URL:

```
https://myserver.com:443/imfpush/v1/apps/com.sample.sms/messages
```

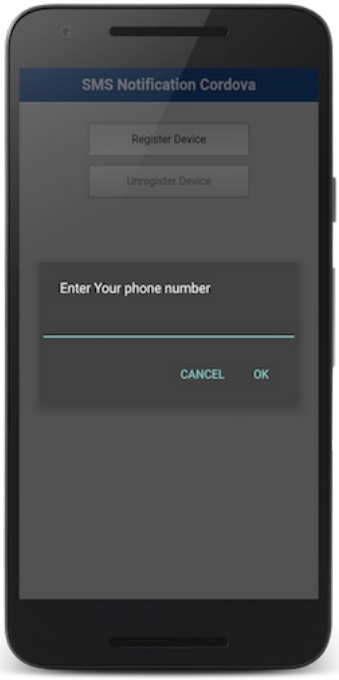
> To review all Push Notifications REST APIs, see the REST API runtime services (https://www.ibm.com/support/knowledgecenter/SSHS8R_8.0.0/com.ibm.worklight.apiref.doc/rest_runtime/c_restapi_runtime.html) topic in the user documentation.

To send a notification, see the sending notifications (../../sending-notifications) tutorial.

Sample application

Click to download (https://github.com/MobileFirst-Platform-Developer-Center/SMSNotificationsSwift/tree/release80) the Cordova project.

Note: The latest version of Google Play Services is required to be installed on any Android device for the sample to run.



Sample usage

Follow the sample's README.md file for instructions.

Last modified on