

Push Notifications in Native Windows Phone 8 Applications

fork and edit tutorial (<https://github.ibm.com/MFPSamples/DevCenter/tree/master/tutorials/en/foundation/7.1/notifications/push-notifications-overview/push-notifications-native-windows-phone-8-applications.html>) | [report issue](#)
(<https://github.ibm.com/MFPSamples/DevCenter/issues/new>)

Overview

This tutorial explains how to configure a MobileFirst Native Windows Phone 8 application to support push notifications.

Also mentioned are the addresses and ports that are required for notifications to arrive to the supported Microsoft Push Notification Service vendor (MPNS).

Prerequisite: Make sure that you read the [Configuring a native Windows Phone 8 application with the MobileFirst Platform SDK \(../../hello-world/configuring-a-native-windows-phone-8-application-with-the-mfp-sdk/\)](#) tutorial first.

Setting up the project



To send push notifications to Windows Phone 8 devices, use the Microsoft Push Notifications Service (MPNS).

- Non-authenticated push notification does not require any setup from the developer. Authenticated push notification requires a Windows Phone Dev Center account.
- To use authenticated push, you must use a certificate that is issued by a Microsoft-trusted root certificate authority. *For production, consider using authenticated push notification in order to ensure that the information is not compromised.*

1. Create a MobileFirst project.

Add a MobileFirst Windows Phone 8 native API. The native API project provides the files that are necessary to build a Windows Phone 8 app.

2. Edit the `application-descriptor.xml` file.

Add the **pushSender** element under the `nativeWindows8App` environment (these settings are also editable with the Application Descriptor Editor in Design mode).

- Non-authenticated push

```
<nativeWindowsPhone8App id="AppName" platformVersion="7.0.0.00.20150312-0731"
version="1.0" xmlns="http://www.worklight.com/native-windowsphone8-descriptor">
  <displayName>AppName</displayName>
  <description>AppName</description>
  <pushSender />
</nativeWindowsPhone8App>
```

- Authenticated push

```
<nativeWindowsPhone8App id="AppName" platformVersion="7.0.0.00.20150312-0731"
version="1.0" xmlns="http://www.worklight.com/native-windowsphone8-descriptor">
  <displayName>AppName</displayName>
  <description>AppName</description>
  <pushSender>
    <authenticatedPush serviceName="" keyAlias="" keyAliasPassword=""/>
  </pushSender>
</nativeWindowsPhone8App>
```

- Replace **serviceName** value with the service name.
- Replace **keyAlias** value with the certificate alias.
- Replace **keyAliasPassword** value with the certificate password.

For more information about using the certificate file, see the topic about setting up push notifications for Windows Phone 8, in the user documentation.

3. Edit the `wlclient.properties` file.

Edit the `wlclient.properties` file in your native Windows Phone 8 project and enter appropriate values for the following fields:

- `wlServerHost` - The host name or IP address of the MobileFirst Server instance.
- `wlServerPort` - The port on which MobileFirst Server is listening.
- `wlServerContext` - The context root of your MobileFirst Server instance.
- `wlMPNSServiceName` = Add the MPNS service name for authenticated push.

```
wlServerProtocol = http
wlServerHost =
wlServerPort = 10080
wlServerContext = /EventSourceNotifications/
wlAppId = NativeWP8EventSource
wlAppVersion = 1.0
wlEnvironment = WindowsPhone8native
wlPlatformVersion = 7.0.0.0
#languagePreferences = Add locales in order of preference (e.g. fr, en, pt-BR)
wlMPNSServiceName = Add the MPNS service name for authenticated push.
```

4. **Modify the native Windows Phone 8 project.**

Edit the `Properties\WMApManifest.xml` file and add the following capabilities:

```
<Capability Name="ID_CAP_PUSH_NOTIFICATION" />
<Capability Name="ID_CAP_IDENTITY_DEVICE" />
```

Windows Phone 8 Push Notifications Service

No specific port needs to be open in your server configuration.
MPNS uses regular http or https requests.

Notification Types