プッシュ通知スターターキット

ユーザーマニュアル

この文章に含まれる情報は、公表の日付の時点での考え方を表しています。市場の変化に応える必要があるため、記載されている内容を約束しているわけではありません。この文書の内容は印刷後も正しいとは保障できません。この文章は情報の提供のみを目的としています。

© Copyright 2015 Microsoft Corporation. All rights reserved.

改版: 2015年 1月 5日

初版: 2014年 12月 01日

目次

[STEP 1. Notification Kit について 3](#_Toc408237793)

[1.1 はじめに 4](#_Toc408237794)

[STEP 2. サーバーサイドの構成 6](#_Toc408237795)

[2.1 通知ハブの構成 7](#_Toc408237796)

[2.2 Webサイトの構成 10](#_Toc408237797)

[2.3 Visual Studioからデプロイ（GitHubからデプロイしてない場合） 14](#_Toc408237798)

[2.4 FTPでデプロイ（GitHub、Visual Studioからデプロイしてない場合） 16](#_Toc408237799)

[STEP 3. クライアントサイドの構成 18](#_Toc408237800)

[3.1 iOSの開発 19](#_Toc408237801)

[3.2 Androidの開発 20](#_Toc408237802)

[3.3 Windows Storeの開発 21](#_Toc408237803)

[3.4 Windows Phoneの開発 22](#_Toc408237804)

[STEP 4. Kitを利用する 23](#_Toc408237805)

[4.1 プッシュ通知管理ポータルへのアクセス 24](#_Toc408237806)

[4.2 通知の送信（Send Notification タブ） 25](#_Toc408237807)

[4.3 宛先一覧の操作（Registration Devicesタブ） 27](#_Toc408237808)

[4.4 通知の予約（Schedule Notificationタブ） 28](#_Toc408237809)

|  |
| --- |
| 1. Notification Kit について |
| このSTEPでは、本 Kitの概要について説明します。 |

## はじめに

本KitはWindows, iOS, Androidなどのアプリケーションに対して簡単にプッシュ通知を組み込むためのフレームワークです。アプリケーションの継続率や再帰率を向上させるために重要なプッシュ通知ですが、機能要件によっては実装が複雑で開発・運用コストがかかります。このKitを用いることで開発者は大幅に開発コストを削減して（サーバーサイドを実装することなく、クライアントサイドにいくつかのコードを追加するだけ）、非常に簡単にスケーラブルで柔軟なプッシュ通知を実装することができます。



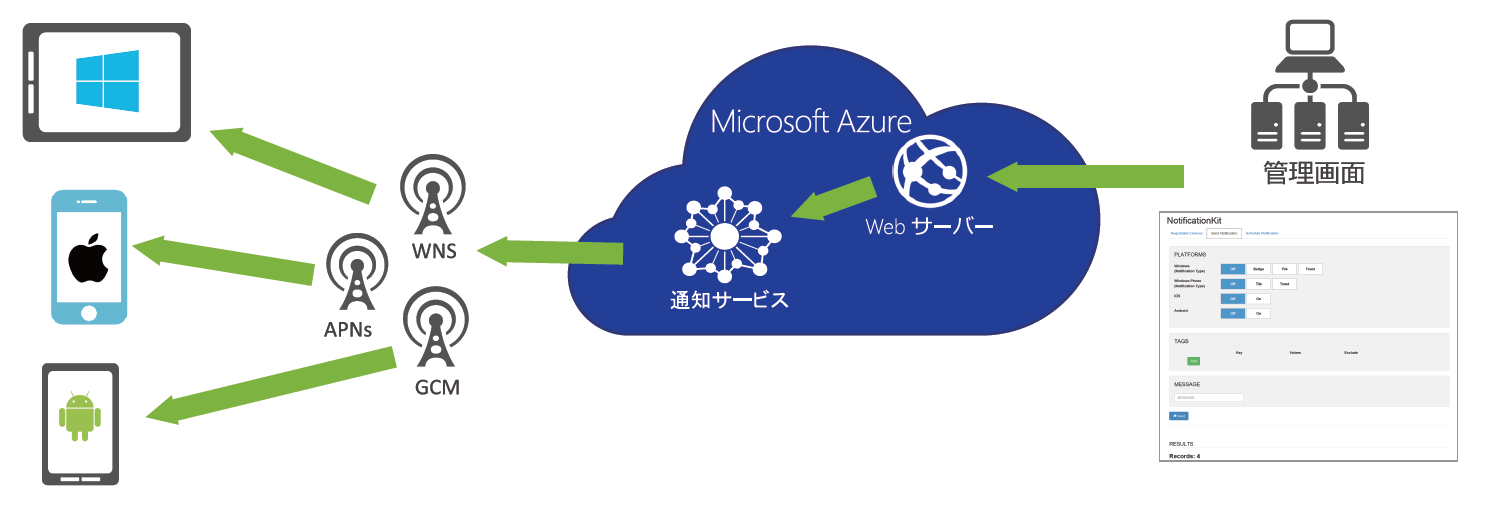
このKitは以下の機能を有します。

* プッシュ通知管理ポータル
* iOS、Android、Windowsに対応したマルチデバイス対応プッシュ通知（リアルタイム大量配信、日時指定によるスケジュール通知、通知先デバイスのグルーピング・フィルタリング）
* 監視（通知の成功、失敗など）
* デバイス登録情報の一覧表示、更新、削除、追加
* デバイス登録情報のインポート・エクスポート（CSV形式）
* 障害発生時の自動復旧（高可用性）

以下のサービスに対応しています。

* Android - Google Cloud Messaging (GCM)
* iOS - Appleプッシュ通知サービス (APNS)
* Windowsストアアプリ - Windows プッシュ通知サービス (WNS)
* Windows Phone - Microsoft プッシュ通知サービス(MPNS)

KitはパブリッククラウドであるMicrosoft Azureのプッシュ通知サービスをベースに実装されています。プッシュ通知サービスにはリアルタイム大量配信、データ管理、サービス監視など便利な機能が予め実装されているので、開発者はプッシュ通知に関わる全ての機能を実装する必要がありません。またKitにはプッシュ通知を管理するためのWebポータルが含まれているので管理用UIを開発する必要もありません。



このKitを利用するためにはAzureアカウントの作成、Webサイトの構成、通知ハブの構成が必要です。全てGUIで簡単に設定できますし、毎月1,000,000プッシュ通知以内であれば継続的に無料で利用することができます。

Azureのアカウントをお持ちでないユーザーは次のサイトを参考に作成してください。尚、Azureには一ヶ月無償の評価版が提供されていますが、本キットが利用するサービスには無償枠が提供されているため枠内であれば（無償期間一ヶ月を超えて）継続して無料で利用できます。

<http://aka.ms/mkaz>

登録が完了すると以下のURLからAzure管理ポータルにアクセスできます。この管理ポータルからAzureの様々なサービスを利用することができます。

<http://manage.windowsazure.com/>



|  |
| --- |
| 1. サーバーサイドの構成 |
| このSTEPでは、Azureのサービスである[通知ハブ]、[Webサイト]を構成してKit本体をデプロイします。 |

## 通知ハブの構成

プッシュ通知の基盤として動作するAzure通知ハブを構成します。

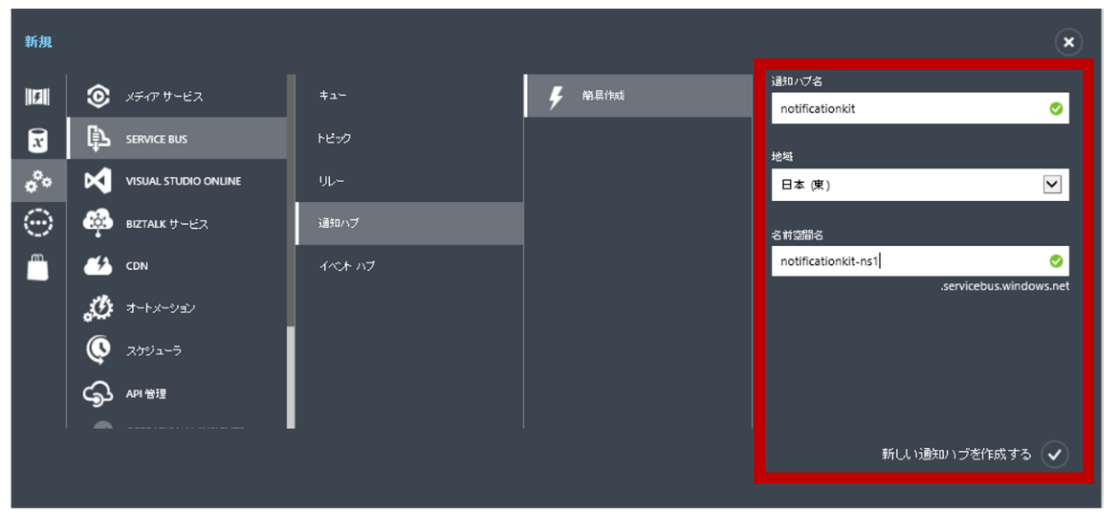
1. Azure管理ポータルにログオンし、画面の左下にある [+新規] をクリックします。

<http://manage.windowsazure.com/>

1. [アプリケーション サービス]、[SERVICE BUS]、[通知ハブ]、[簡易作成] の順にクリックします。



1. 通知ハブ名に任意の名前を入力して、アプリケーションをデプロイする場所（地域）を選択して、右下の [新しい通知ハブを作成する] をクリックします。なお、名前空間名に「指定された名前は使用できません」というエラーが表示された場合は他のユーザーが作成した名前空間と重複しているので別の名前空間を設定します。



1. 前の手順で作成した名前空間をクリックし、上部にある [通知ハブ] タブをクリックして、作成した通知ハブを選択します。







1. [接続情報] タブをクリックして表示された画面から「DefaultListenSharedAccessSignature（クライアントユーザー用）」と、「DefaultFullSharedAccessSignature（管理者用）」の接続文字列をメモ帳などにコピーしておきます。





1. 最後に[スケール]タブをクリックして無料になっていることを確認しておきます。

なお、通知のスケジュール配信やインポートなど高度な機能を利用したい場合は標準（有料）を設定してください。詳細は次のサイトをご確認ください。

**料金詳細**

<http://azure.microsoft.com/ja-jp/pricing/details/notification-hubs/>

## Webサイトの構成

Azure管理ポータルからAzure Webサイトの構成を行います。

1. [コンピューティング]、[Webサイト]、 [カスタム作成] の順にクリックします。



1. Webサイトの名前を入力して、アプリケーションをデプロイする場所（地域）を選択し、データベースを [無料の20MB SQLデータベースの作成]に設定します。またGitHubから直接デプロイする場合は「ソース管理から発行」にチェックをしてください。その後、[→] をクリックします。



1. データベースのログイン名、パスワードを設定して [✔] をクリックします。



1. 「ソース管理から発行」をチェックした場合、次の画面が表示されるので「外部リポジトリ」を選択します。その後、[→] をクリックします。



1. 「リポジトリのURL」に次のアドレスを指定して[✔] をクリックします。

<https://github.com/notificationkit/notificationkit>

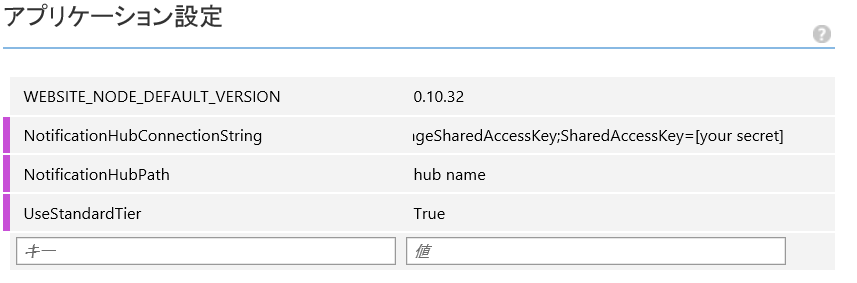


※ 管理ポータルのステータスが「実行中」になっていても実際のデプロイ完了には数分かかる場合があります。

1. 作成したWebサイトの[構成]タブをクリックして[アプリケーション設定]に以下の情報を設定します。（ここではWebサイトと通知ハブの関連付けをしています）



|  |  |
| --- | --- |
| キー | 値 |
| NotificationHubPath | 前のステップで作成した通知ハブの名前 |
| NotificationHubConnectionString | 前のステップでコピーした「DefaultFullSharedAccessSignature（管理者用）」の値 |
| UseScheduleOption（オプション） | スケジュール配信の有効化（有料）  true : 使用する  false（デフォルト）:使用しない  ※使用する場合は通知ハブのスケールを無料ではなく標準にしておく必要があります。 |



1. [スケール]タブをクリックして無料になっていることを確認しておきます。



**Azure Websites の料金詳細**

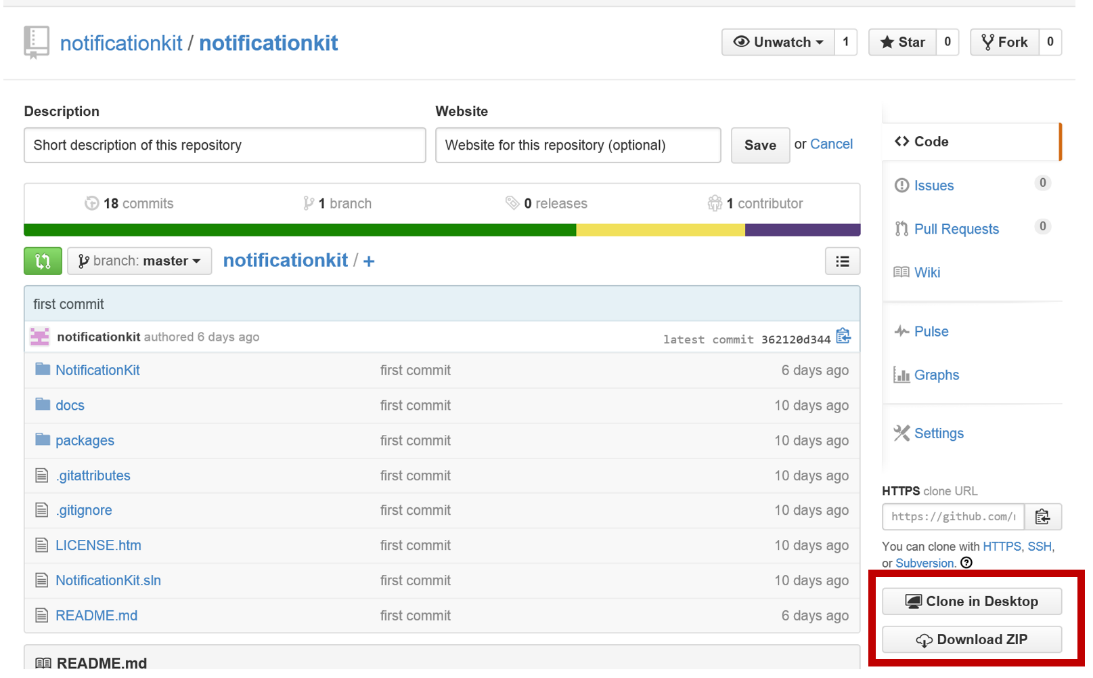
<http://azure.microsoft.com/ja-jp/pricing/details/websites/>

## Visual Studioからデプロイ（GitHubからデプロイしてない場合）

Visual Studio 2013をお持ちであれば、Web Deployで簡単にアプリケーションをAzure Webサイトにデプロイすることができます。

1. GitHubからリポジトリのクローンを作成、もしくはZIPファイルをダウンロードした後、NotificationKitのソリューションファイル（NotificationKit.sln）を開きます。

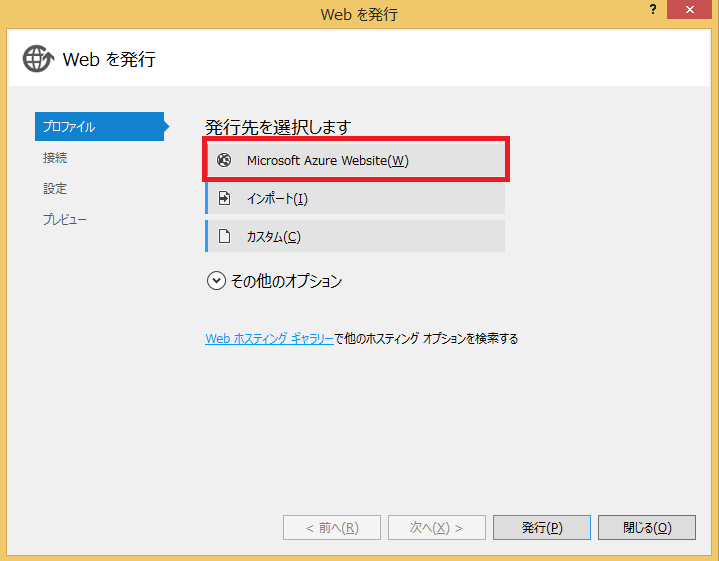
<http://aka.ms/nkit_src>



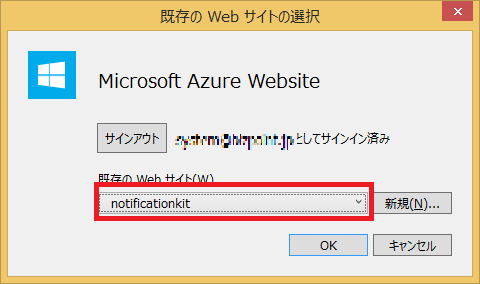
1. ソリューションエクスプローラでプロジェクトを右クリックし、[発行]をクリックします。

****

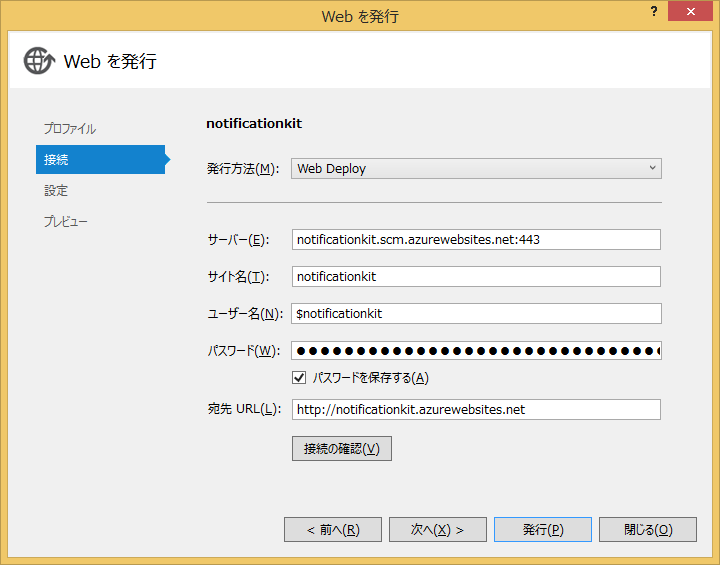
1. [Microsoft Azure Website]をクリックします。



1. サインインし、前の手順で作成したWebサイトを選択し、[OK]をクリックします。



1. [発行]をクリックします。



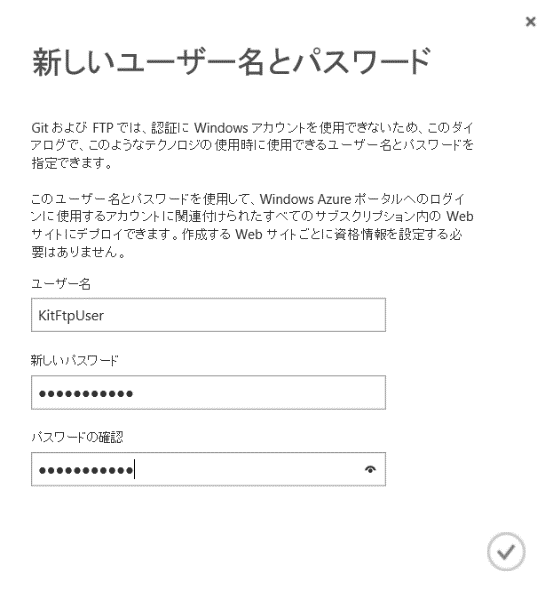
## FTPでデプロイ（GitHub、Visual Studioからデプロイしてない場合）

FTPでKitをAzure Webサイトにデプロイします。

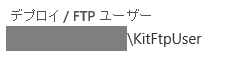
1. Azure 管理ポータルで Web サイトのダッシュボードを開き、 [デプロイ資格情報のリセット]をクリックします。

****

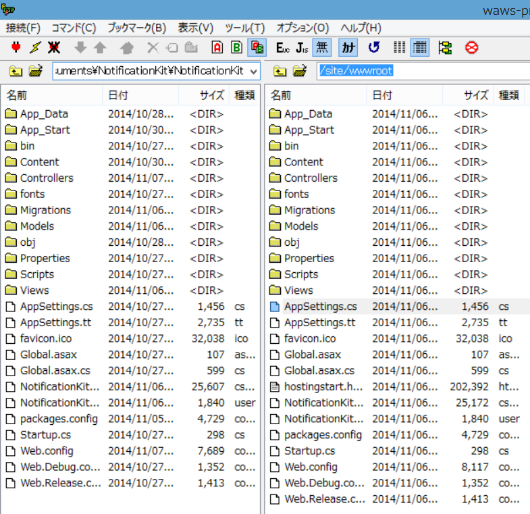
1. 任意のユーザー名とパスワードに変更します。



1. ダッシュボード内に表示される デプロイ/FTPユーザー　にてユーザー名が変更されていることを確認します。



1. FTPツールなどを使ってAzure Web サイトの /site/wwwroot にNotificationKitフォルダの内容をアップロードします。



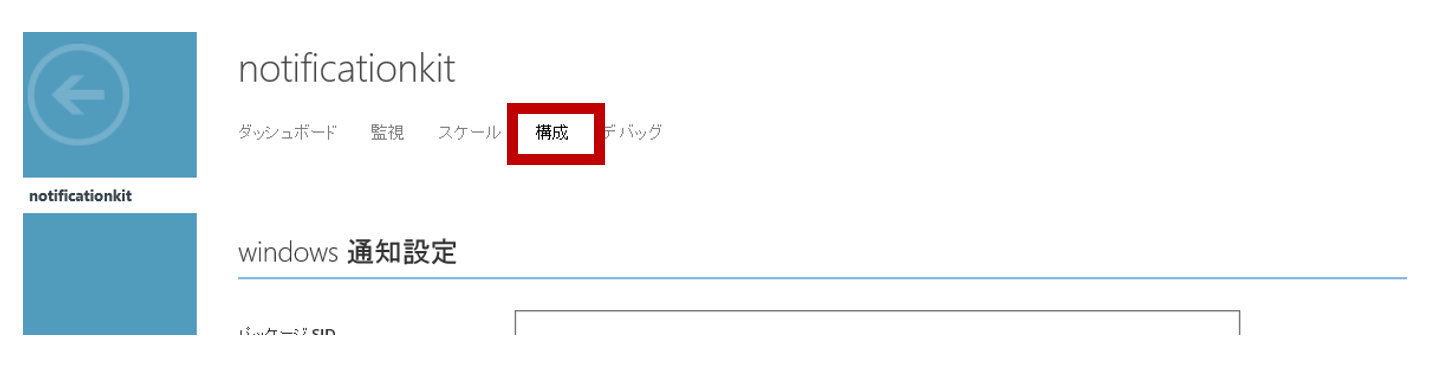
|  |
| --- |
| 1. クライアントサイドの構成 |
| このSTEPでは、本 Kitの概要について説明します。 |

## iOSの開発

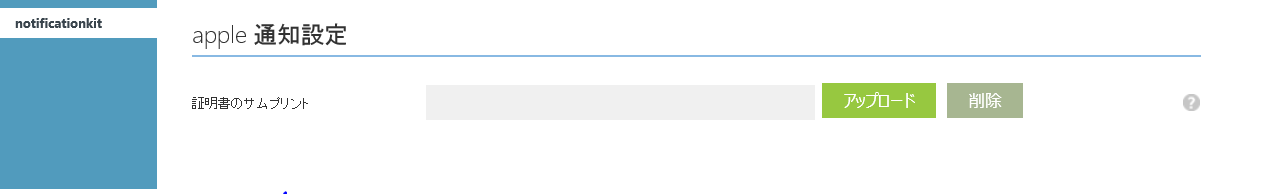
iOSアプリケーションにプッシュ通知を組込む方法は次のサイトを参考にしてください。（なお、前のステップで通知ハブの作成は終わっているため新たに作成する必要はありません）

<http://aka.ms/nkit_ios>

実際に通知を送るには作成した通知ハブの構成タブをに資格情報をセットする必要があります。



iOSではapple通知設定に証明書のサムプリントを設定します。



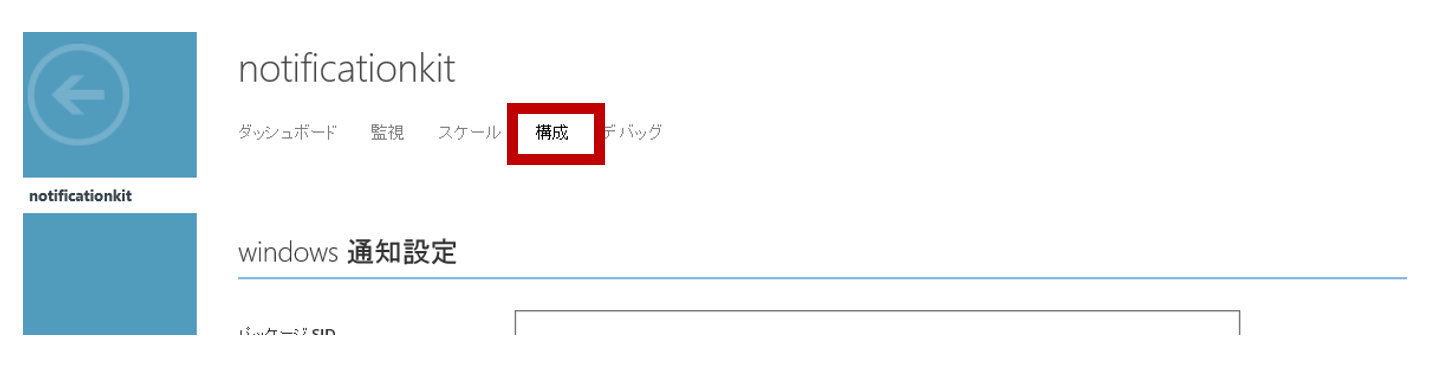
証明書のサムプリントの入手方法は上記サイトを参考にしてください。

## Androidの開発

Androidアプリケーションにプッシュ通知を組込む方法は次のサイトを参考にしてください。（なお、前のステップで通知ハブの作成は終わっているため新たに作成する必要はありません）

<http://aka.ms/nkit_android>

実際に通知を送るには作成した通知ハブの構成タブをに資格情報をセットする必要があります。



Androidではgoogle cloud messaging設定にGCM APIキーを設定します。



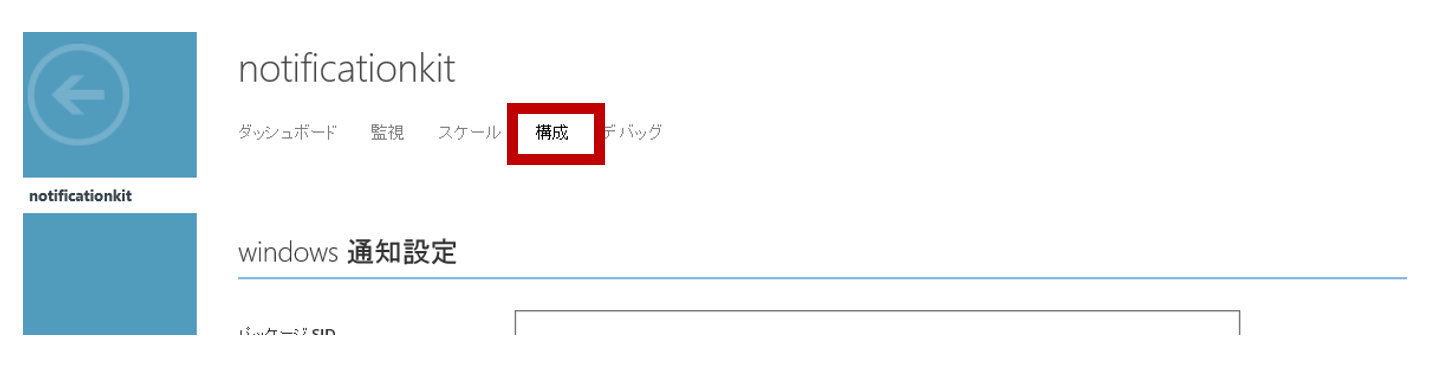
GCM APIキーはGoogle Developer Consoleから取得します。具体的な取得方法は上記サイトを参考にしてください。

## Windows Storeの開発

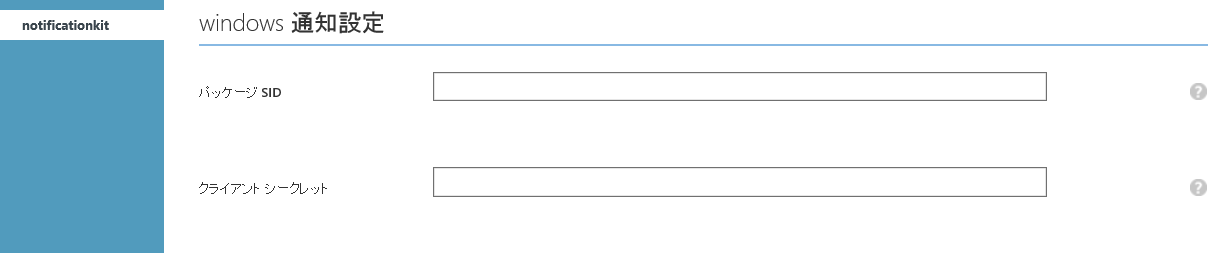
Windows Storeアプリケーションにプッシュ通知を組込む方法は次のサイトを参考にしてください。（なお、前のステップで通知ハブの作成は終わっているため新たに作成する必要はありません）

<http://aka.ms/nkit_ws>

実際に通知を送るには作成した通知ハブの構成タブをに資格情報をセットする必要があります。



Windows StoreではパッケージSIDとクライアントシークレットを設定します。



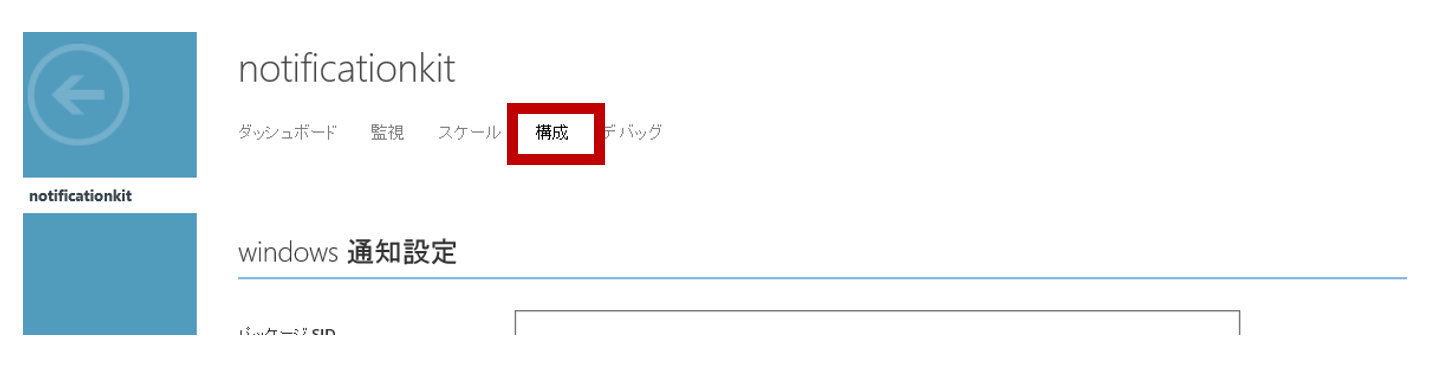
パッケージSIDとクライアントシークレットの入手方法は上記サイトを参考にしてください。

## Windows Phoneの開発

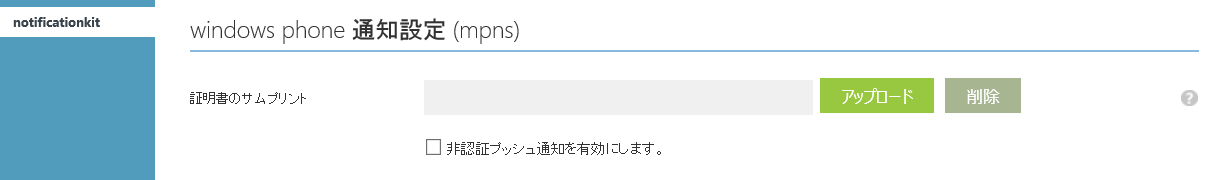
Windows Phoneアプリケーションにプッシュ通知を組込む方法は次のサイトを参考にしてください。（なお、前のステップで通知ハブの作成は終わっているため新たに作成する必要はありません）

<http://aka.ms/nkit_wp>

実際に通知を送るには作成した通知ハブの構成タブをに資格情報をセットする必要があります。



Windows Phoneではパッケージ証明書のサムプリントを設定します。



証明書のサムプリントの入手方法は上記サイトを参考にしてください。

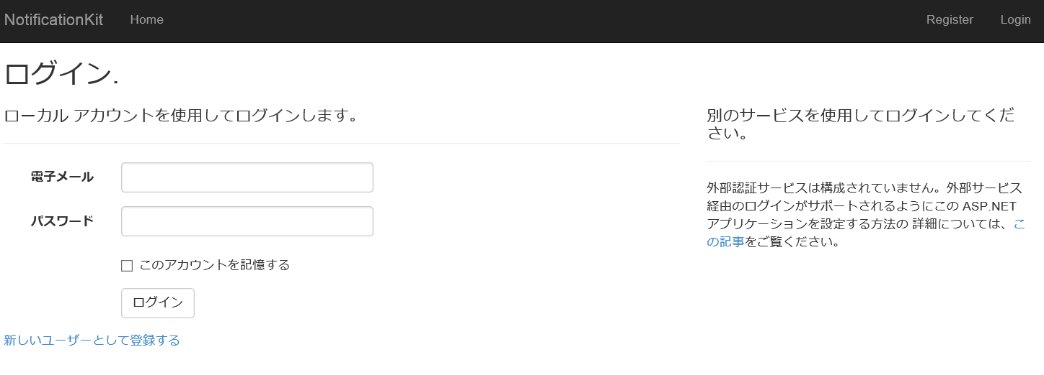
|  |
| --- |
| 1. Kitを利用する |
| このSTEPでは、本 Kitの使い方を説明します。 |

## プッシュ通知管理ポータルへのアクセス

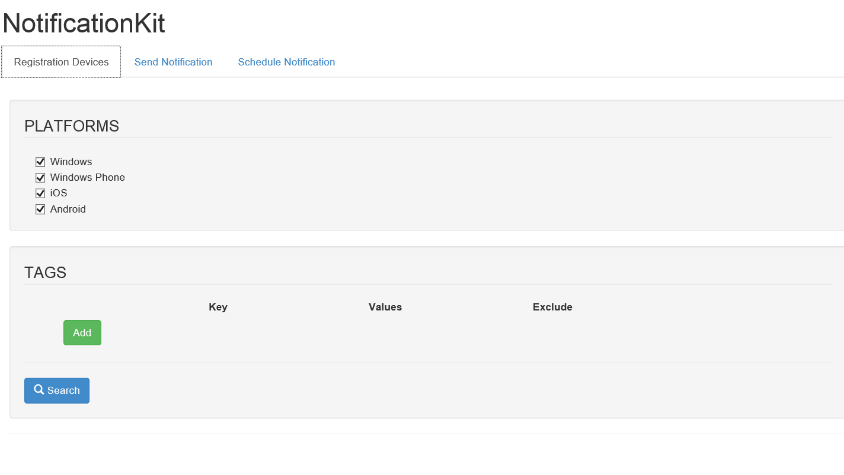
Azure Webサイトへ発行済みのプッシュ通知管理ポータルにアクセスします。

1. 発行先のURLにアクセスするとログイン画面が表示されるので任意のユーザーでログインします。（初回は [新しいユーザーとして登録する] から新しいユーザーの登録が必要です）

http://[指定したWebサイトの名前].azurewebsites.net/



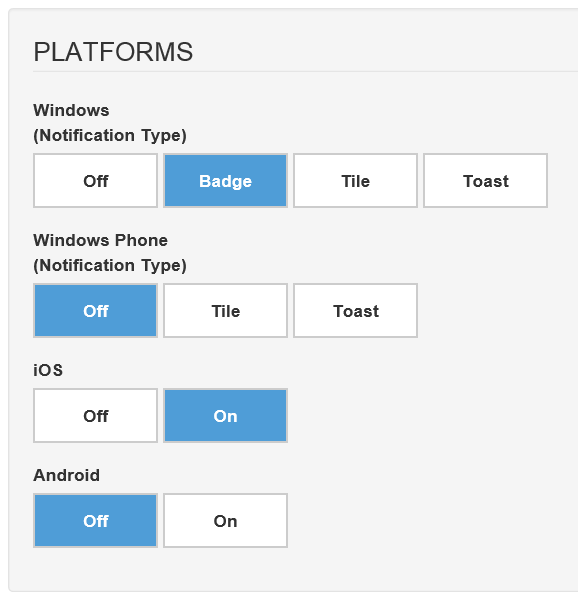
1. プッシュ通知の管理ポータルが表示されます。



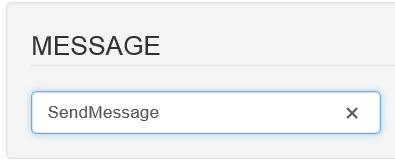
## 通知の送信（Send Notification タブ）

ここではプッシュ通知の管理ポータルからデバイスに対してプッシュ通知を行います。

1. プッシュ通知の送り先になるプラットフォーム（Windows, iOS, Android）と通知メッセージの形式を指定します。



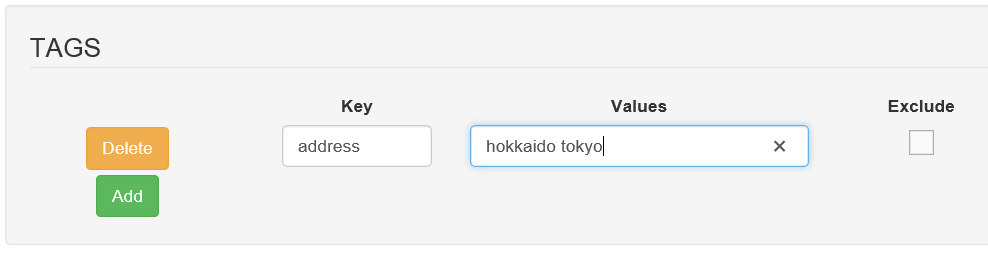
1. 実際に通知するメッセージをMessageに設定します。



1. 宛先を絞り込むにはTagsに値を設定します。KeywordとValueの値を適切に設定して下さい。

tagsに「address:hokkaido」が設定されてユーザーに通知を行う場合、Keyに「address」、 Valueに「hokkaido」を入力します。（tagsは単なる文字列です。）

また、Valueはスペースで区切ることにより複数設定することができます。Notにチェックを付けると指定した条件以外のデータを検索します。Keyを複数設定したい場合は[Add]ボタンを押すと条件項目が追加できます。



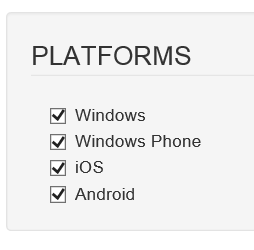
1. Sendボタンを押すと対象のデバイスにメッセージが送信されます。



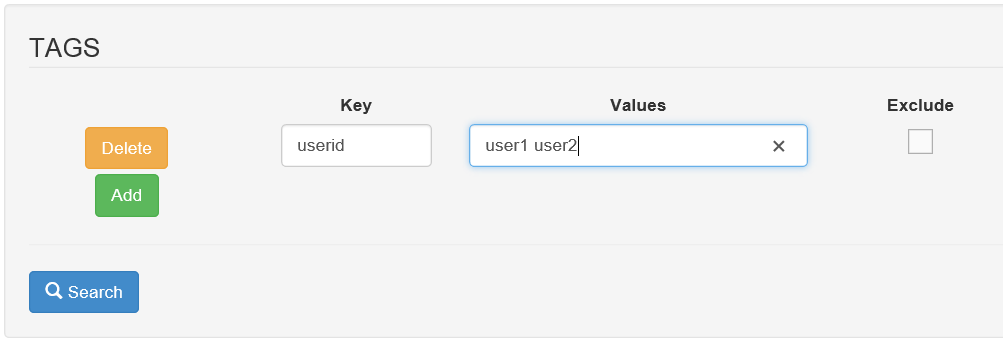
## 宛先一覧の操作（Registration Devicesタブ）

プッシュ通知の宛先データを表示・削除・インポート・エクスポートすることができます。

1. 対象のプラットフォームにチェックします。



1. KeyとValueの値を適切に設定してSearchボタンを押すと宛先を絞り込むことができます。



1. Exportボタンを押すと、検索条件に一致するデータをCSVファイルで出力することができます。
2. Importボタンを押すと、Exportで出力した形式のCSVファイルを取り込み、登録情報の追加、更新ができます。

・追加する場合はRegistrationIdを空白に設定して下さい。

・更新できる項目はHandle、Tagのみです。

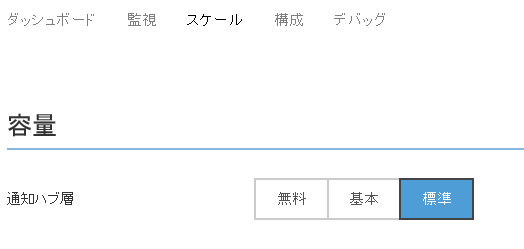
・RegistrationTime、ExpirationTimeはインポートした時刻で再設定されます。

・CSVファイルの編集時にタイトルを削除した場合は、HeaderチェックボックスをOFFにして下さい。

1. 登録情報を削除したい場合は×ボタンをクリックします。

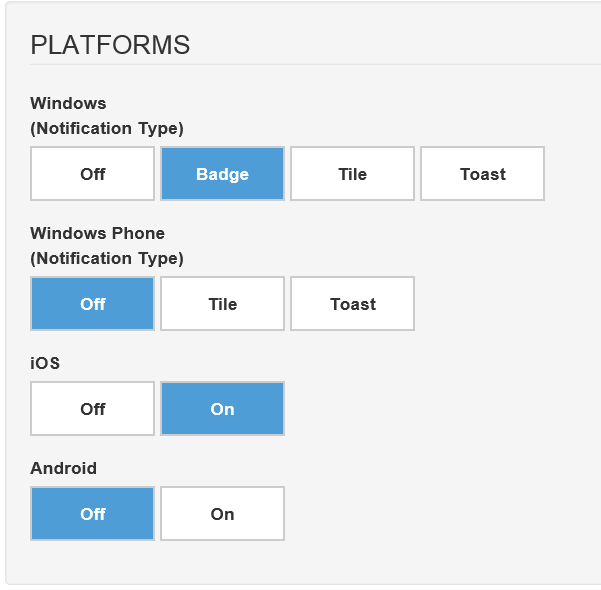
## 通知の予約（Schedule Notificationタブ）

※この機能はAzure 通知ハブが標準（有料）のプランで実行されていて Webサイトの[アプリケーション設定]でUseScheduleOptionをtrueに設定している必要があります



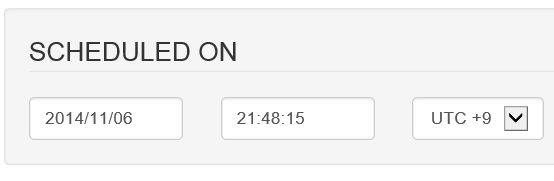
ここでは通知を予約して指定した日時にスケジュール配信する機能をご紹介します。

1. 通知を送信するプラットフォームを選択します。



1. 通知を送信する日時をSCHEDULED ONに設定します。

※日時は現在日時から7日以内にする必要があります。



1. タグ、メッセージを設定してSubmitボタンを押すと通知が予約されます。
2. 正常に予約が完了すると画面下部に予約時の情報が表示されます。予約をキャンセルする場合は × ボタンをクリックしてください。

※すでに送信されている場合はキャンセルできません

