

モバイル Android Campaign Callbacks とその使用方法

概要

WalkMe Mobile SDKバージョン1.6.0以降 WalkMe Mobile SDKはCampaign Callbacksをサポートしています。

コールバックは、WalkMeキャンペーンイベントに登録するためのプログラマティックな方法です。キャンペーンのコールバックインターフェースを実装することで、アプリはコールバックオブジェクトから転送されたデータ（キャンペーンやユーザー情報を含む）を使用でき、またWalkMeとは関係のないアナリティクスやCRMシステムにその情報を渡したり、キャンペーンとのユーザーインタラクションに従ってアプリやユーザー設定を更新したりなど、アプリの用途に応じて利用できます。

ユースケース

キャンペーンのコールバックは、以下のように使用できます：

- 任意の分析システムからオンボーディングを追跡
- WalkMeシャウトアウトとのユーザーエンゲージメントに基づいて CRMで機会を生成 – 例えば「無料のデモに興味がありますか？ はい/いいえ」
- サポートチケットのトピックとワークスルーの使用状況 / 完了を追跡して比較するために既存のサポートシステムと統合

キャンペーンのコールバックを実装するために必要なすべてのステップの説明を以下に示します。

Campaign Callbacksを実装する手順

キャンペーンのコールバックに登録するには、まずこのインターフェースを実装します。

```
/**
 * Interface definition for a callback to be invoked in Campaign actions.
 */
public interface WMCampaignInfoListener {
    /**
     * Called after campaign was dismissed.
     *
     * @param campaignInfo The dismissed campaign info.
     */
    void onCampaignDismissed(WMCampaignInfo campaignInfo);
}
```

インターフェースを実装した後、Campaign Callbackイベントに登録するためにsetCampaignInfoListener方法を使用してください：

```
/**
 * Register a listener to campaign events
 *
 * @param campaignInfoListener The listener
 *
 */
public static void setCampaignInfoListener(WMCampaignInfoListener
campaignInfoListener)
```

コールバックの戻り値オブジェクトはWMCampaignInfoクラスになります。これには次の情報が含まれています：

アイテム	受信元	説明
キャンペーンCTA	getCampaignCta()	アプリユーザーが相互に操作したキャンペーンCTA カスタム以外のすべてのCTAについて、返されたCTA[] abbi://cta、例： abbi://ok または abbi://never
キャンペーンID	getCampaignId()	キャンペーンに割り当てられた一意のID
キャンペーンCTA ID	getCampaignCtaId()	キャンペーンCTAの一意のID
User Data (ユーザーデータ)	getUserData()	campaignsとの相互作用をしたユーザーに関する情報を保持するオブジェクト。以下のオブジェクトの説明を参照してください。
キャンペーンデータ	getCampaignData()	キャンペーンに追加されたデータを使用できるように追加されたプレースホルダーオブジェクト アンケートの送信データは、ここに入力されます。

これは、WMCampaignInfoオブジェクトの構造です：

```
public class WMCampaignInfo {

    public String getCampaignCta()
```

```

public String getCampaignId()

public String getCampaignCtaId()

public WMUserData getUserData()

public Map<String, Object> getCampaignData()
}

```

上記のように、WMCampaignInfoは下記に説明されているオブジェクトWMUserDataが含まれていません：

アイテム	受信元	説明
パブリックユーザー属性	getUserAttributesMap()	ユーザーがキャンペーンと相互作用する時点でアプリユーザーに設定されたすべてのパブリックユーザー属性。リストには常に属性キーとその値が含まれています。例：{ "lead_id" : "12345" , "account_type" : "family"}
プライベートユーザー属性	getPrivateUserAttributes()	ユーザーがキャンペーンと相互作用する時点でアプリユーザーに設定されたすべてのプライベートユーザー属性。リストには常に属性キーとその値が含まれています。例：{ "lead_id" : "12345" , "account_type" : "family"}
ユーザー作成タイムスタンプ	getUserCreationTimestamp()	WalkMeによってユーザーが最初に識別されたタイムスタンプ
セッション期間	getSessionDuration()	ユーザーがcampaignsとの相互作用を行った時点のユーザーセッションの長さ
Androidバージョン	getDeviceVersion()	ユーザーのOSバージョン
デバイス固有のID	getDeviceId()	デバイスの固有のID
デバイスモデル	getDeviceModel()	ユーザーのデバイスモデル
Device Orientation (デバイスの向き)	getDeviceOrientation()	ユーザーがcampaignsと相互作用した時点のデバイスの向き

アプリのバージョン	getAppVersion()	ユーザーが使用しているアプリバージョン
アプリ名	getAppName()	アプリ名
Locale(ロケール)	getLocale()	ユーザーのデバイスロケール
SDKバージョン	getSdkVer()	ユーザーが使用しているアプリと統合されたSDKバージョン
セッションID	getSessionId()	WalkMeによって生成された固有のセッションID
新しいユーザー表示	getIsNewUser()	WalkMeによってユーザーが識別されるのはこれが初ですか (true / false)
プッシュ通知ステータス	getIsPushEnabled()	ユーザーがアプリのプッシュ通知を有効にします (true / false). アプリにプッシュ通知機能がない場合-値は「false」になります。
デバイスタイムゾーン	getTimezone()	デバイス上で設定されているユーザータイムゾーン
ネットワーク	getNetwork()	ユーザーが使用しているネットワークタイプ [WiFi / Cellular / Offline]
ユーザーの緯度	getLocationLat()	位置情報緯度の実装のためのプレースホルダー
ユーザーの経度	getLocationLong()	位置経度を実装できるプレースホルダー
システム名	getSystemName()	ユーザーのOS名-この場合、常に「Android」になります
現在のユーザータイムスタンプ	getTimestamp()	ユーザーの現在のタイムスタンプ

これは、WMUserDataオブジェクトの構造です：

```
public class WMUserData {

    /**
     * User Attributes in current session.
     */
}
```

```
public Map<String, Object> getUserAttributesMap()

/**
 * Private User Attributes in current session.
 */
public Map<String, Object> getPrivateUserAttributes()

/**
 * Returns the User Creation Timestamp for this User.
 *
 * @return long Timestamp.
 */
public long getUserCreationTimestamp()

/**
 * Returns the User Session Duration in second.
 *
 * @return double second.
 */
public double getSessionDuration()

/**
 * Returns the Device Android version.
 *
 * @return String version number.
 */
public String getDeviceVersion()

/**
 * Returns the Device unique id.
 *
 * @return String id.
 */
public String getDeviceId()

/**
 * Returns the Device brand Model.
 *
 * @return String Model.
 */
public String getDeviceModel()
```

```
/**
 * Returns the Device Orientation PRT_REG / LSL.
 *
 * @return String Orientation.
 */
public String getDeviceOrientation()
/**
 * Returns the application version.
 *
 * @return String version.
 */
public String getAppVersion()

/**
 * Returns the application Name.
 *
 * @return String Name.
 */
public String getAppName()

/**
 * Returns the locale language for this instance of the Java Virtual
Machine.
 *
 * @return String Name.
 */
public String getLocale()

/**
 * Returns the walkme sdk version.
 *
 * @return String version.
 */
public String getSdkVer()

/**
 * Returns the Session unique id.
 *
 * @return String id.
 */
public String getSessionId()

/**
```

```
* Returns if the user use the app for the first time.
*
* @return String boolean.
*/
public String getIsNewUser()

/**
 * Returns if Push notification approved for user.
 *
 * @return String boolean.
 */
public String getIsPushEnabled()

/**
 * Returns user device timezone.
 *
 * @return String timezone.
 */
public String getTimezone()

/**
 * Returns user current network ( WIFI / 3G )
 *
 * @return String network.
 */
public String getNetwork()

/**
 * Returns user current location latitude if available
 *
 * @return String latitude.
 */
public String getLocationLat()

/**
 * Returns user current location longitude if available
 *
 * @return String longitude.
 */
public String getLocationLong()

/**
 * Returns user System Name (Android).
```

```
*
* @return String Android.
*/
public String getSystemName()

/**
* Returns the Current Timestamp for this User.
*
* @return String Timestamp.
*/
public String getTimestamp()
}
```