**Cách cài đặt libtracking**

**I. Android**

**1. Cấu hình cho thư viện**

Thư viện bao gồm:

- File libMesgLog.so nằm trong 2 thư mục armeabi và armeabi-v7a

- File MesgLog.java nằm trong package com.mesg.log là interface được dùng bên java

Các Permissions bên android được thêm vào AndroidManifest.xml bao gồm:

*<uses-permission android:name="android.permission.READ\_EXTERNAL\_STORAGE" />*

*<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />*

*<uses-permission android:name="android.permission.INTERNET" />*

*<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />*

*<uses-permission android:name="android.permission.READ\_PHONE\_STATE"/>*

Các hàm trong file MesgLog.java

*public static native String getLibVersion();*

*public static native String getDeviceID(Context context);*

*public static native boolean sendLogInstall(Context context, String ext);*

*public static native boolean sendLogConfirm(Context context, String ext);*

*public static native boolean sendLogAction(Context context, String isClosed, String ext);*

*public static native boolean sendLogClickBanner(Context context, String app\_id, String currentActivity, String currentAction, String ext);*

**2. Cách sử dụng các hàm bên java**

**a) Hàm sendLogInstall**

Mục đích:

Tracking việc install Apps

Cách sử dụng:

Hàm này được gọi khi bắt đầu khởi động App

Parameters:

|  |  |
| --- | --- |
| Field | Description |
| context | Application context. Sử dụng hàm getApplicationContext() để lấy |
| ext | Field mở rộng  Dữ liệu truyền vào cách nhau bởi dấu **";"**  Ex: 3646;iwida  Nếu không sử dụng thì truyền vào String rỗng ("") |

Response:

- **true**: Nếu send log success

- **false**: Send log failed

**b) Hàm sendLogAction**

Mục đích:

Tracking thời gian sử dụng Apps

Cách sử dụng:

Hàm này được gọi khi khởi động App và kết thúc App

Parameters:

|  |  |
| --- | --- |
| Field | Description |
| context | Application context. Sử dụng hàm getApplicationContext() để lấy |
| isClosed | Khi khởi động App thì truyền vào isClosed = "0"  Khi tắt App thì truyền vào isClosed = "1" |
| ext | Field mở rộng  Dữ liệu truyền vào cách nhau bởi dấu **";"**  Ex: 3646;iwida  Nếu không sử dụng thì truyền vào String rỗng ("") |

Response:

- **true**: Nếu send log success

- **false**: Send log failed

**c) Hàm sendLogClickBanner**

Mục đích:

Tracking việc clicks vào ads ở trong Apps

Cách sử dụng:

Mỗi 1 ads sẽ được cấp 1 app\_id

Khi click vào ads, cần truyền vào activity hiện tại bao gồm package + activity:

Ex: vcc.soha.game.DetailsActivity

currentAction ở đây là click("0") hoặc view ("1")

Parameters:

|  |  |
| --- | --- |
| Field | Description |
| context | Application context. Sử dụng hàm getApplicationContext() để lấy |
| app\_id | App\_id của Ads |
| currentActivity | Activity hiện tại có chứa Ads, bao gồm package name + tên Activity  Ex: vcc.soha.game.DetailsActivity |
| currentAction | Trường hợp click vào Ads thì truyền **"0"**  Trường hợp view Ads thì truyền **"1"** |
| ext | Field mở rộng  Dữ liệu truyền vào cách nhau bởi dấu **";"**  Ex: 3646;iwida  Nếu không sử dụng thì truyền vào String rỗng ("") |

Response:

- **true**: Nếu send log success

- **false**: Send log failed

**d) Hàm getLibVersion**

Mục đích:

Lấy version hiện tại của lib tracking

Cách sử dụng:

Hàm này sử dụng để check version của lib tracking khi cần test

Parameters: None

Response: Version hiện tại của lib tracking

**e) Hàm getDeviceID**

Mục đích:

Lấy thông tin DeviceID

Cách sử dụng:

Hàm này sử dụng để lấy thông tin DeviceID của máy

Parameters:

context: Application context. Sử dụng hàm getApplicationContext() để lấy

Response: Trả về DeviceID của máy

**f) Hàm sendLogConfirm**

Mục đích:

Sử dụng cho các App có bản cài mini, cần update dữ liệu trước khi chạy

Cách sử dụng:

- Với App cài lần đầu tiên và chưa login thì hàm này được gọi khi khởi tạo form login

- Với App đã cài 1 lần và đã login, token login thường sẽ được lưu, do vậy khi update version của App thì khi mở lại App sẽ check việc lưu token. Tại thời điểm này cần check là nếu tồn tại token và version là version mới thì phải gọi hàm  *sendLogConfirm*

Như vậy hàm này được gọi tại 2 thời điểm như ở trên

Parameters:

|  |  |
| --- | --- |
| Field | Description |
| context | Application context. Sử dụng hàm getApplicationContext() để lấy |
| ext | Field mở rộng  Dữ liệu truyền vào cách nhau bởi dấu **";"**  Ex: 3646;iwida  Nếu không sử dụng thì truyền vào String rỗng ("") |

Response:

- **true**: Nếu send log success

- **false**: Send log failed

**II. IOS**

**1. Thư mục chứa library tracking**

Danh sách file của library tracking:

|  |  |  |
| --- | --- | --- |
| Đường dẫn | Danh sách file | Mô tả |
|  | libMesgLog.a | File static library của lib tracking |
| include/MesgLog | MesgLog.h | File chứa các hàm sử dụng cho việc send log lên server |

Các hàm được đưa ra trong lib bao gồm:

*-(void) sendLogClickBanner:(NSString \*)app\_id org:(NSString\*)org cov:(NSInteger)cov;*

*-(void) setExt:(NSString \*)ext;*

*-(NSString \*) getVersion;*

*-(NSString \*) getDeviceID;*

*-(void) sendLogConfirm:(NSString \*)confirmExt;*

**Note**: Framework cần thêm là *Security.framework*

**2. Cách sử dụng**

Library sử dụng dạng single instance, tự động handle việc mở và đóng App

- Khi mở App, sẽ tự động bắn log mở App và log confirm cài App

- Khi đóng App, tự động bắn log đóng App

Việc đóng mở App ở đây là việc Active hoặc không Active App

**a) Hàm sendLogClickBanner**

Mục đích:

Tracking việc clicks vào ads ở trong Apps

Parameters:

|  |  |
| --- | --- |
| Field | Description |
| app\_id | App\_id của Ads |
| org | Controller hiện tại  Ex: vcc.soha.game.DetailsController |
| cov | Trường hợp click vào Ads thì truyền **0**  Trường hợp view Ads thì truyền **1** |

Response: Không trả về kết quả

**b) Hàm setExt**

Mục đích:

Set thêm parameter phục vụ cho việc tracking sau này

Cách sử dụng:

Hàm này được gọi khi khởi tạo instance

Parameters:

|  |  |
| --- | --- |
| Field | Description |
| ext | Field mở rộng  Dữ liệu truyền vào cách nhau bởi dấu **";"**  Ex: 3646;iwida  Nếu không sử dụng thì truyền vào String rỗng ("") |

Response: Không trả về kết quả

**c) Hàm getVersion**

Mục đích:

Lấy thông tin version hiện tại của lib tracking

Cách sử dụng:

Hàm này được sử dụng để check version hiện tại của libtracking trong App

Parameters: None

Response: Version hiện tại của libtracking

**d) Hàm getDeviceID**

Mục đích:

Lấy thông tin DeviceID

Cách sử dụng:

Hàm này sử dụng để lấy thông tin DeviceID của máy

Parameters: None

Response: Trả về DeviceID của máy

**e) Hàm sendLogConfirm**

Mục đích:

Sử dụng cho các App có bản cài mini, cần update dữ liệu trước khi chạy

Cách sử dụng:

- Với App cài lần đầu tiên và chưa login thì hàm này được gọi khi khởi tạo form login

- Với App đã cài 1 lần và đã login, token login thường sẽ được lưu, do vậy khi update version của App thì khi mở lại App sẽ check việc lưu token. Tại thời điểm này cần check là nếu tồn tại token và version là version mới thì phải gọi hàm  *sendLogConfirm*

Như vậy hàm này được gọi tại 2 thời điểm như ở trên

Parameters:

|  |  |
| --- | --- |
| Field | Description |
| confirmExt | Field mở rộng  Dữ liệu truyền vào cách nhau bởi dấu **";"**  Ex: 3646;iwida  Nếu không sử dụng thì truyền vào String rỗng ("") |

Response: không trả về kết quả

**f) Sử dụng**

Setup cho AppDelegate

- import file SohaTracking.h vào AppDelegate.h

- Khởi tạo lib tracking:

*- (BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions*

*{*

*MesgLog\*cpi = [MesgLogshareInstance];*

*[cpi setExt:@"2336;3484"];*

*}*

Log click Ads thì trong chương trình phải chủ động gọi

Ex:

*MesgLog\*cpi = [MesgLogshareInstance];*

*[cpi sendLogClickBanner:@"1235476562" org:@"vcc.soha.ViewController" cov:1 ext:@""];*