iOS SDK

Last Updated: July 16, 2012

Introduction

Placed for Developers provides mobile application developers a simple SDK that allows easy integration with Placed Location Analytics.

It exposes simple public API calls that can be used to register strategic events that occur while an application is being used.

The SDK also collects location and sensor data that is relayed along with the application event data, enabling strategic location-based application usage analytics.
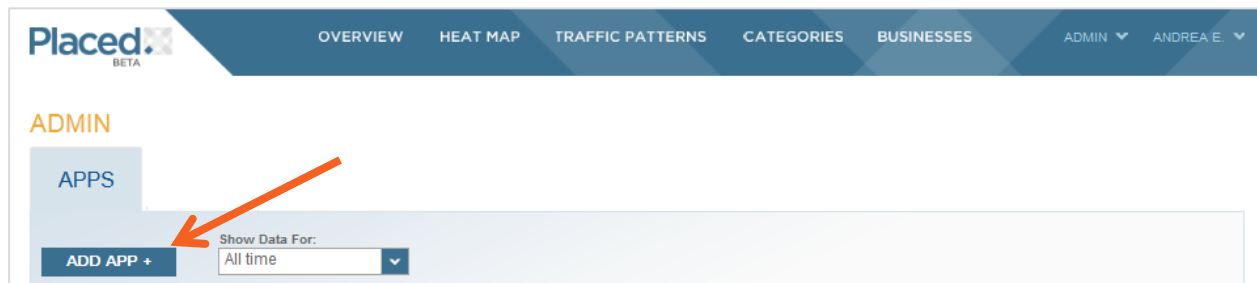
The SDK has been designed for easy set up and integration with both new and existing mobile applications.

Steps to integrate your Application with the Placed SDK

Registration

(If you've already registered an account, skip to the Set Up section of the document)
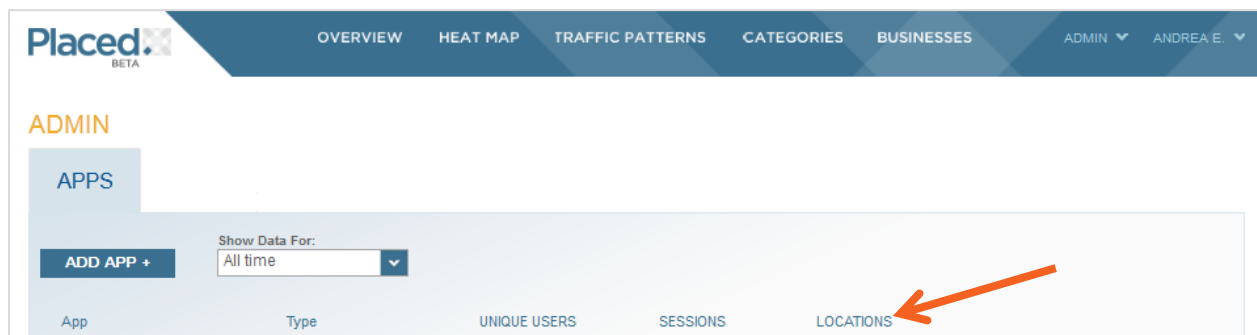
- Create a developer account on Placed.com

  - Visit the *Placed for Developers App Dashboard* to set up your developer account.

- Add your Application

  - Once signed into the *Placed for Developers App Dashboard,* you can add an application

  - Click on "Add App," enter a name for your application and select Android as platform

  - Upon adding your application you will be provided a unique **SDK Application Key**

- Download the App API library

  - Upon adding your application you can download the **placed-sdk-ios.zip**

    - This zip file contains these instructions and the SDK library that you need to reference in your application to enable Placed Location Analytics

Set Up and Configuration

- After adding the Placed SDK files to your project, ensure the following are set up in the Build Phases tab for your Target.

- Link Binary With **Libraries:**

  o CoreData.framework

  o CoreTelephony.framework

  o CoreLocation.framework

  o CoreMotion.framework

  o SystemConfiguration.framework

  o libz.dylib

  o libPlacedAgent.a

- Copy Bundle Resources:

  o PlacedData.mom

- Add "Other Linker Flags":

  o -all_load –ObjC

- Confirm proper implementation by logging into Placed.com and viewing the **Location** count for your application. Note that it takes at least one day for activity to be reflected in the dashboard.

Integration

- **Initialize**

  - #import "PlacedAgent.h"

  - Instantiate the PlacedAgent at the creation of your application.

  - We advise placing this in the ***didFinishLaunchingWithOptions*** *of your AppDelegare.m*

    - *[PlacedAgent initWithAppKey:@"<Application Key>"]*

- **Start**

  - Invoke the StartSession call to indicate the beginning of a session for which you wish to associate subsequent events

  - *[PlacedAgent logStartSession]*

    - If there is an active session, and the application ends the active session and starts a new session within the 'Resume Interval':

      - We will resume the previous session

    - If there is an active session and the App starts a new session without ending the current active session:

      - We will always terminate the previous session and start a new session

    - If there is no active session and the application attempts to invoke any API call:

      - We will automatically start a new session and start data collection

- **Page View**

  - Invoke the PageView call to indicate landing on a new screen/page within the application

  - *[PlacedAgent logPageView:@"<Page Title>"]*

    - Provide the name of the screen/page you wish to track

- **Events**

  - Once you have registered the session start and a landing page, you can proceed to invoke various Custom Events

  - *[PlacedAgent logCustomEvent:@"<Event Title>"]*

    - Provide a meaningful event title that you wish to track and report on

    - Optionally you can provide a list of Key/Value attributes associated with the Event that will be used for more granular analytics

      - *NSArray *attributes = [NSArray arrayWithObject: [PlacedAgent attributeWithName:@"<Attribute Name>" andValue:@"<Attribute Value>"]*

      - *[PlacedAgent logCustomEventWithTitle:@" <Event Title>" andAttributes:attributes]*

  - All events are associated with the most recent PageView invoked

- **Custom Variables**

  - You may choose to register custom variables during the operation of your Application

  - *[PlacedAgent logCustomVariableWithScope:<Scope> andValue:<Variable Value>]*

    - You can specify a free form value and scope it to the User, Session or Page

- **Demographics**

- You may choose to provide demographics including a unique user identifier, age or gender.

- *[PlacedAgent logUniqueId:@"<Unique Id>"]*

    - You can use this to log your application specific user/device unique identifier. We will use this information internally for aggregated reporting only.

- *[PlacedAgent logAge:<Age>]*

    - You can use this to log your user's age. Valid values range from 1 to 115

- *[PlacedAgent logGender:<Gender>]*

    - You can use this to log your user's gender. Valid values include MALE, FEMALE and UNSPECIFIED

- **End**

    - Invoke the EndSession call to indicate the completion of a session

    - We advise invoking end session before the app is closed

    - *[PlacedAgent logEndSession]*

        - This closes the current active sessions and stops the Placed location tracking

Advanced Concepts

- **Logging**

    - You can use this call to enable logging for your debugging

    - *[PlacedAgent setEnableLogging:<NO or YES>]*

        - You may choose to disable logging when releasing your application to the market

- **Data Sync**

    - Data is collected only during an active session, and is synchronized with the Placed servers periodically. This does not block your application's activities.

        - This sync cycle is not configurable.

- **Data Collection Mode**

    - You can configure the mode in which the Placed SDK collects data. We recommend that you keep the setting at DEFAULT. Any changes to a level below DEFAULT will result in lower coverage and accuracy in the analytics.

    - *[PlacedAgent logGender:<DataCollectionMode>]*

    - The modes to select from are as follows:

        - DataCollectionMode.PRECISE

Placed.

- DataCollectionMode.DEFAULT

- DataCollectionMode.CONSERVATIVE

Support

- For further guidance contact support@placed.com