

# Alohar Developer Overview

v1

## 1. Introduction

[Alohar Mobile Inc](#) develops the **Alohar Mobile Location Behavior Platform**, which understands mobile device user's behavior by analyzing location and other mobile sensor data. The platform is composed of a smartphone client SDK (software development kit) for Android and iPhone and several cloud-based services. The Alohar Mobile Location Behavior Platform uses mobile location, motion, wifi and timing data to learn about the mobile device user in real time and over time. Therefore it provides more advanced functionality than a conventional mobile LBS (location based services) platform.

The Alohar Developer API is the programming interface exposed by the SDK. It defines a simple interface for third-party mobile application developers to build a diversity of mobile applications which can offer intelligent functionalities based on mobile user behaviors. Example applications include smart personal assistants, location-based games, mobile health apps, mobile shopping/coupon apps, social networking apps and mobile dating apps.

This document provides a high level overview of the functionalities provided by the Alohar Developer API. The detailed functional specification for Android and iOS is described in a separate document.

Figure 1 illustrates the high level architecture.

### 1.1 Main Functionalities and Benefits

The Alohar SDK provides a rich set of functionalities, accessible through a simple API. By using the easy-to-integrate API, third-party mobile application developers quickly realize the following benefits:

- Automatically understand a user's mobile motion state (e.g. stationary, walking or driving).
- Automatically detect the places (including the name and category) that the user visits
- Get notifications when a user arrives at or departs from a place.
- Automatically get the number of times a user visits a place, and how much time is spent there per visit.
- Minimize power consumption while gathering data from the mobile device.

The beauty of the SDK and API is that the mobile app developer can focus on his application-specific functionalities, rather than low level user location and behavior, and therefore achieve a fast time to market.

The Alohar Mobile Location Behavior Platform differs from conventional geo-fencing systems, which are tedious and difficult to use because they require specific lat/lng boundaries to be defined ahead of time through a manual process. In contrast, the Alohar system accurately

detects the places the user visits without any prior knowledge of the place or the geography. This frees up the developer to spend more time coding his app because the Alohar platform takes care of accurately detecting and reporting the POI data.

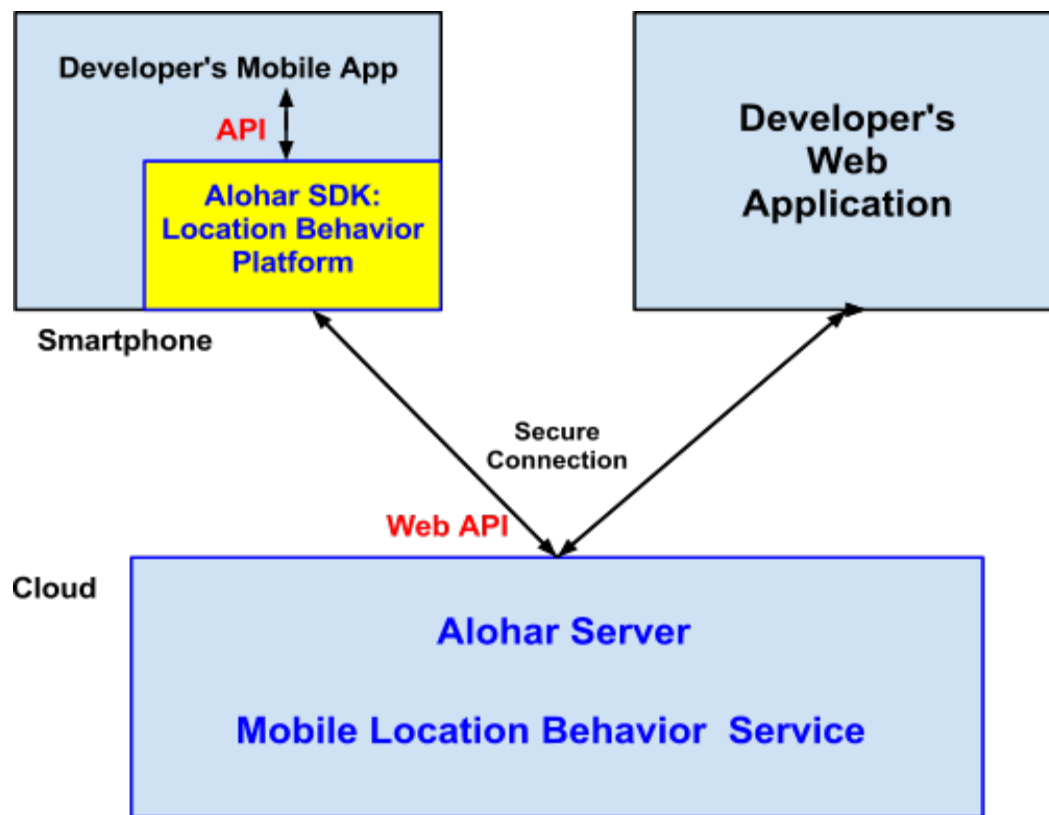


Figure 1

## 1.2 Sample Apps Using API

Many new mobile apps can be created with the SDK and API. As an example, a mobile monopoly game app can use the Alohar Developer API to automatically detect the time a player enters a restaurant, and in real time ask the user whether he wants to “buy” the restaurant using virtual currency. After he buys it, if another game player steps into this restaurant, the API detects this event and notifies the game, which asks the new player to either pay rent or trade properties with the restaurant owner.

Another example is a mobile health app which can use the API to automatically detect how active the mobile device user is everyday. For example, how often the user goes to gym, plays tennis, or how often the user drives his car and dines at junk food restaurants.

A mobile shopping app can use the API to automatically detect the moment the user enters an electronics store, and display the current sales that are running (e.g., 20% off a LED HDTV

- today only). Also, a mobile coupon app can use the API to track which coupons are actually redeemed per store.

## 2. API Functionalities

Once application developers embed the Alohar SDK in their applications and set up some preferences, the SDK works in the background without much additional work from the developers. It can trigger some callback functions when predefined events happen. Or the application can proactively query the SDK for specific information.

The API interface provides two sets of major functionalities:

1. Real-time triggers based on the user's current mobile behavior and the current place.
2. Historical statistics and information based on the past user mobile behavior and places visited.

### 2.1 Real-Time Triggers and Current User State

The application developer can use API functions to retrieve the following mobile user behavior info:

- User's current motion state (i.e., is the user mostly stationary, walking, or driving)
- What place (name and category) the user is currently at
- Whether the user is moving from one place to another place or within the same place.

The application developer can register callback functions to be called when one of the following events occur:

- The user arrives at a place:
  - It can be any place, or a specific place that meets certain conditions.
- The user departs from a place:
  - It can be any place, or a specific place that meets certain conditions.

The developer's application will get a callback when the user arrives or leaves a generic place, or the developer can define the set of interesting places based on:

- The name of the places (such as "Starbucks", or "Bestbuys").
- The category of the places (such as restaurants, or schools, or offices)
- The addresses of the places.

When the callback is triggered, or when the application proactively calls the function to get current place, a suite of information is returned for the place the user visits, including:

- The arrival time at the place.
- The departure time and duration for this visit at the place
- The name and category of the place.
- The list of names of the neighbor places, properly ranked based on the user's preference.

### 2.2 Historical Statistics

The application developer can query for the historical statistics of a user, given a time window and/or a location range.

The developer can query for all the places a user has visited between a start time and an end time, and/or around a specific spatial area. The developer can set up a predefined filter to specify that only a subset of the places are to be returned. For example, the developer can specify that only places with specific name patterns, or in some certain categories (e.g. cafe, restaurant, or electronic stores).

For each specific visit to a place, the timing information can be retrieved. For example, the arrival time and the departure time of the visit.

Queries can be made to get the total time spent at a certain place, such as at Starbucks, or at a gym, over the past month.

In addition, queries can be made to get the time spent in various motion mode, such as stationary, walking or driving.

## **2.3 System Control and Running Mode**

The developer can control the running mode of the Alohar SDK.

The Alohar SDK can run persistently in the background all the time. Alternatively, the developer can control when to start the Alohar background system and when to suspend it.

## **3. Developer API Key Management**

The developer visits a web page at <http://www.alohar.com/developers> (to be launched) to sign up for the Alohar SDK and get an API key to be used with their applications. This secret API key is used to authenticate the specific developer.

## **4. Privacy and User Identity Management**

The Alohar platform provides high security and strong protection of an app user's privacy.

Alohar handles the developer's identity in an anonymous fashion. The developer manages the authentication of their own application end-users. Alohar does not require the identity of the users of the third-party application. Instead, the application developer only provides an anonymous user id to Alohar, which Alohar uses to manage each user's data. This way, Alohar never sees the real identity of the user of the developer's application.

All communication between the Alohar SDK and the Alohar server is encrypted and performed over SSL.

All user data is kept strictly confidential, and is not shared with any other party without the application developer's and the end-user's consent. More information is available in a separate privacy policy.

## **5. Future Functionalities & Requirement Requests**

In the near future, we plan to support web application developers with a RESTful web API. This will allow web developers to access the above functionalities.

We welcome feedback and additional requirements from mobile app developers. For questions, suggestions and requests, please email [developer@alohar.com](mailto:developer@alohar.com) . Thanks!