# iOS MotionDnaSDK API Documentation

## Control

### func start(withDeveloperKey: String)

Starts up the SDK with valid developer key as input. As no configuration is provided in this call, the SDK uses the defaults specified in the configuration table below. A successful or failed authentication will be reported through the reportStatus() callback.

### func start(withDeveloperKey: String, andConfigurations: [String : Any])

Starts up the SDK with valid developer key as input. A dictionary with options from the Configuration table below determine how the SDK is configured on startup. Any options not changed in the dicitonary will maintain their defaults as specified in the table. A successful or failed authentication will be reported through the reportStatus() callback.

### func stop()

Stops the execution of the SDK and frees up memory and any saved estimation state

### func reset()

Equivalent to a stop() followed by a start() while preserving the developer key and any confguration from the initial start() call.

### func pause()

Halts position estimation and collection of sensor data.

### func resume()

Resumes position estimation and collection of sensor data.

### func setBackgroundModeEnabled(\_ state: Bool)

Allow the MotionDnaSDK to run while your app is backgrounded by turning the CLLocationManager on. If you wish to use this, you must enable the `Location updates` `Background Mode` options in XCode and enable `Background Modes` if it is not on.

## Configuration

|  |  |  |  |
| --- | --- | --- | --- |
| **Keyword** | **Options** | **Default** | **Description** |
| model | String (“simple”, “standard”, “headmount”) | “standard” | Choose the type of model to use for motion estimation |
| gps | Bool | true | Use GPS to set an initial position and perform corrections when outdoors |
| corrected\_trajectory | Bool | false | Show path leading up to global position fix and a history the updated path then corrections happen |
| callback | Double | 40ms | Rate (ms) at which estimation is delivered to app layer |
| motion\_source | String (“device”,”external”) | “device” | Determines the source of inertial sensor data used for position estimation |
| logging | Bool | false | Record log file for debugging with Navisens team |

## Change Global Estimation

### func setGlobalPosition(latitude: Double, longitude: Double)

Manually assigns the user’s position in the global frame.

### func setGlobalPosition(latitude: Double, longitude: Double, heading: Double)

Manually assigns the user’s position and heading in the global frame.

### func setGlobalHeading(heading: Double)

Manually assigns the user’s heading in the global frame. Units are in degrees.

## Change Cartesian Estimation

### func setCartesianHeading(heading: Double)

Manually assigns the user’s heading in cartesian space. Parameter units are in degrees and must be contrained to the -180 to +180 frame.

### func setCartesianPosition(x: Double, y: Double)

Manually assigns the user’s cartesian position to the postition (x,y) in cartesian space. Parameter units are in meters.

## External Sensors

### func inputMotion(withTimestamp: Double, roll: Double, pitch: Double, yaw: Double, accX: Double, accY: Double, accZ: Double)

This method must be called at regular intervals with sensor data if you have started the SDK with configuration `motion\_source` set to `external`.

**Parameters**

Timestamp units must be in seconds. Accelerometer x, y, and z should be in G's and Gyro roll, pitch, and yaw unit should be in radians/sec.

### func inputMotion(withTimestamp: Double, roll: Double, pitch: Double, yaw: Double, accX: Double, accY: Double, accZ: Double, heading: Double, headingAccuracy: Double)

This method must be called at regular intervals with sensor data if you have started the SDK with configuration `motion\_source` set to `external`.

**Parameters**

Timestamp units must be in seconds. Accelerometer x, y, and z should be in G's and Gyro roll, pitch, and yaw unit should be in radians/sec. Heading and heading accuracy must be in radians with heading expressed in a 0-2π frame increasing in a clockwise direction.

## SDK Info

### static func sdkBuild() -> String

This static method will return a string representing the current build number (e.g. NAVISENS-R20.08.17.15.56-MASTER).

### static func sdkVersion() -> String

This static method will provide the currention version number of the SDK (e.g. v2.0.0).

## Callbacks (MotionDnaSDKDelegate implementation)

### func receive(motionDna: MotionDna)

Provides a MotionDna object with the current position information and other characteristics of a users motion and status. See Estimation Properties below for further details.

### func report( status: MotionDnaSDK.Status, message: String)

Reports to the developer any changes to the SDK or errors that are noteworthy. Status types are accompanied by a message providing additional details on the event that has occured and any action that should be taken.

## Estimation Properties

These are the values representing the estimation provided

|  |  |  |
| --- | --- | --- |
| **Class** | **Variable/Getter** | **Type** |
| **MotionDna** | attitude | Attitude |
|  | location | Location |
|  | classifiers | Dictionary [String:Classifier] |
|  | timestamp | double |
|  |  |  |
| **Attitude** | euler | Euler |
|  | quaternion | Quaternion |
|  |  |  |
| **Euler** | roll | double |
|  | pitch | double |
|  | yaw | double |
|  |  |  |
| **Quaternion** | x | double |
|  | y | double |
|  | z | double |
|  | w | double |
|  |  |  |
| **Location** | cartesian | CartesianLocation |
|  | global | GlobalLocation |
|  |  |  |
| **CartesianLocation** | x | double |
|  | y | double |
|  | z | double |
|  | heading | double |
|  |  |  |
| **GlobalLocation** | latitude | double |
|  | longitude | double |
|  | altitude | double |
|  | accuracy | enum GlobalLocationAccuracy (LOW, HIGH) |
|  |  |  |
| **Classifier** | prediction.label | String |
|  | prediction.confidence | double |
|  | statistics | Dictionary [String : PredictionStats] |
|  |  |  |
| **PredictionStats** | duration | double |
|  | distance | double |
|  | percentage | double |

## Simultaneous Localization and Mapping (SLAM)

### func recordObservation(withIdentifier: Int, uncertainty: Double)

Input an identifier when visiting and revisiting a landmark in an indoor setting. The ID must be accompanied by an uncertainty value in meters indicating the maximum possible distance the user could currently be from the ID'd landmark. This will provide the SDK with additional information that can enable improved corrections to a users position.

## Logging

### func activeLogFilePath() -> String

If logging is enabled this will return the full path of the nav file generated during the SDK operation.

### func activeLogFilename() -> String

If logging is enabled this will return the simple filename for the nav file generated during the SDK operation.